Die approbierte Originalversion dieser Diplom-/Masterarbeit ist an der Hauptbibliothek der Technischen Universität Wien aufgestellt (http://www.ub.tuwien.ac.at).

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology (http://www.ub.tuwien.ac.at/englweb/).

FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

An online wayfinding aid for the Vienna University of Technology

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Wirtschaftsinformatik

eingereicht von

Johannes Harms
Matrikelnummer 0426887

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung
Betreuer: Thomas Grechenig
Mitwirkung: Christoph Wimmer

Wien, TT.MM.JJJJ

(Unterschrift Verfasser) (Unterschrift Betreuer)

Erklärung / Declaration

Johannes Harms,
Josefstädter Straße 14,
1080 Wien, Österreich

“Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.”

I hereby declare that I am the sole author of this thesis, that I have completely indicated all sources and help used, and that all parts of this work - including tables, maps and figures - if taken from other works or from the internet, whether copied literally or by sense, have been labelled including a citation of the source.

Ort, Datum

Unterschrift

Danksagung / Thanks

Ich möchte meinem Betreuer *Prof. Thomas Grechenig* danken, der mich schon bei meiner Bachelor-Arbeit betreut hat, der mir das Thema zu dieser Arbeit empfohlen hat, und an dessen Institut ich an spannenden Projekten mitarbeiten kann. Mein besonderer Dank geht auch an die betreuenden Assistenten *Dipl.Ing. Christoph Wimmer* und *Monika Suppersberger, Bakk.techn.* für ihre Unterstützung, für gute Ratschläge und für die harte Arbeit, für das Feedback und für richtungsweisende Gespräche.

Ich bin dankbar für *Prof. Georg Gartner*s Unterstützung beim Thema Kartographie und Wegfindehilfen, und für das Angebot für zukünftige Zusammenarbeit. Gleichermaßen möchte ich *Prof. Wolfgang Zagler* für Ratschlag und Hilfe zum Thema Accessibility danken.

Mein Dank geht auch an meine *Freunde und Kollegen* für das Mitmachen bei Interviews und Test Sessions. Und - last but not least - möchte ich meinen Eltern und Geschwister danken, fürs Bei-mir-Sein, für Hilfe und Unterstützung zu allen Zeiten.

An online wayfinding aid for the Vienna University of Technology

Abstract

Wayfinding aids communicate route and survey knowledge to help the user reach a distant goal. This knowledge can be objectified and transmitted using different media, e.g. through maps or verbal instructions. This work investigates web applications as technical medium for wayfinding aids: The focus of this work is the design of a new online wayfinding aid for students, teachers, and visitors at the Vienna University of Technology.

One challenge with indoor route descriptions is that they will often include three-dimensional instructions such as when to change into a particular floor. These instructions can be described verbally in a comprehensible way, but are difficult to depict graphically together with a street map. As a solution, this work proposes to place two-dimensional floorplan images over a regular street map, and to add additional, verbal instructions concerning the floor levels.

Usability Engineering methods have been employed in this work. It is structured by an iterative design process where prototyping and evaluation are used repeatedly during the design phase. The process kept a focus on users by involving them in interviews and usability tests.

This work results in a prototype of a campus wayfinding aid built as a web application. The prototype features an interactive, two-dimensional street map with floorplans, and photos, text, and a hypertext structure. The interaction design of the prototype offers several different ways of accessing the system: navigation on the map, textual search, hyperlinks from other systems, and browsing in the hypertext structure.

In summary, the work at hand provides an overview on relevant wayfinding theory and employs usability engineering methods for the construction of a prototype. It thereby offers a detailed specification of the content, appearance and interaction of an online campus wayfinding aid.

Keywords: Online wayfinding aid, campus map, indoor navigation. Usability engineering, UCD, HCI, wireframe, mockup, sketch, design, prototype, usability testing, evaluation.

Kurzfassung

Wegfindehilfen kommunizieren Routen- und Umgebungswissen, um so den Benutzer beim Erreichen eines entfernten Ziels zu unterstützen. Dieses Wissen kann mit Hilfe verschiedener Medien objektifiziert und vermittelt werden, z.B. mit Landkarten oder verbalen Anweisungen. Diese Arbeit untersucht interaktive Web-Applikationen als technisches Medium für Wegfindehilfen: Im Zentrum der Arbeit steht eine neu zu konzipierende, online Wegfindehilfe für Studenten, Professoren und Besucher an der Technischen Universität Wien.

Eine Schwierigkeit bei Routenbeschreibungen in Innenräumen ist, dass sie oft dreidimensionale Anweisungen enthalten, zum Beispiel für das Wechseln in den richtigen Stock. Diese Anweisungen können verbal verständlich wiedergegeben werden, lassen sich jedoch graphisch schlecht in Kombination mit einer Straßenkarte darstellen. Als Lösung werden zwei-dimensionale Stockwerkspläne über eine reguläre Straßenkarte gelegt, wobei verbale Anweisungen bezüglich der Stockwerke hinzugefügt werden.

Usability Engineering Methoden kamen in dieser Arbeit zum Einsatz. Sie wird von einem iterativen Designprozess strukturiert, in dem Prototypen und Evaluation wiederholt während der Design-Phase verwendet wurden. Der Prozess orientierte sich am Benutzer, der in Interviews and Usability Tests eingebunden wurde.

Das Ergebnis dieser Arbeit ist ein Prototyp einer Campus-Wegfindehilfe, der als Web-Applikation umgesetzt wurde. Der Prototyp beinhaltet eine interaktive, zweidimensionale, Nord-orientierte Karte mit Straßen- und Stockwerksplänen, und außerdem Fotos, Text, und eine Hypertext-Struktur. Das Interaktionsdesign des Prototypen bietet mehrere Möglichkeiten, auf das System zuzugreifen: Navigation in der Karte, textuelle Suche, Hyperlinks von anderen Systemen, and das Browsen in der Hypertext-Struktur.

Zusammenfassend bietet die vorliegende Arbeit einen Überblick über relevante Wegfindungs-Theorie und verwendet Usability Engineering Methoden für den Bau eines Prototypen. Sie liefert so eine detaillierte Spezifikation für Inhalt, Aussehen und Interaktion einer online Campus-Wegfindehilfe.

Schlüsselwörter: Online Wegfindehilfe, Campus Karte, Gebäude-Navigation, Innenraum-Navigation. Usability engineering, UCD, HCI, Wireframe, Mockup, Entwurf, Design, Prototyp, Usability Test, Evaluierung.

Executive summary[1]

Recommendations for building a wayfinding system for the Vienna University of Technology:

1. Consider the prototype:
The result of this work is a prototype for a wayfinding system: It tells people where to find rooms or persons. Have a look at the screenshots and explanations in chapter 6.

2. Save money or create more value:
Indoor wayfinding software is complex and expensive to build: Another much simpler campus map[2] took more than 400 working days to complete! – Look for off-the-shelf software before building your own, continue to cooperate with wegweiser[3] or try to buy their maps. If you are building your own system, try to sell it to others.

3. Get cartographic know-how:
If you decide to roll your own system: Making good maps is a science of its own – try to cooperate with the Cartographic Institute of the Vienna University of Technology.

3. Work with the real estate management team
and with the TISS project's people for the room reservation system: They have the best data about buildings and rooms. But don't simply copy architectural plans as a wayfinding aid.

4. Don't forget accessibility
and try to cooperate with the Institute of Integrated Study at Vienna University of Technology for this purpose.

5. Keep the doors open for mobile wayfinding and internationalization:
Even if these features are not required now, they may be in the future.

Other recommendations:

1. *Redundancy* is necessary: Provide wayfinding information using multiple media: Textual instructions, indoor and outdoor maps, and photos or drawings. And allow for multiple ways of accessing the system: Links from other systems, search, and hypertext structure.

2. *Don't go into dynamic routing* if you have to build it yourself - it's a lot of work and not so necessary for the user.

3. *Being able to rotate the map* into the direction of a building's entrance is useful but technically difficult.

[1] Although unusual for an academic work, an executive summary has been added to this thesis because it is likely to feed into a follow-up project. It will help project managers understand the recommendations and implications of this work quickly.
[2] The University of Wisconsin's campus map, see `http://www.uwosh.edu/map/campusmap.htm`, June 2010
[3] The website `http://wegweiser.ac.at` has been providing campus maps up until now.

Contents

1 Introduction **7**

2 Wayfinding **9**

2.1 Definitions of wayfinding . . . 9

2.2 Spatial memory and cognitive maps . . . 11

2.3 The orientation of maps . . . 15

2.4 Indoor wayfinding . . . 16

2.5 The medium used for wayfinding aids . . . 17

2.6 The necessary amount of detail in wayfinding aids . . . 19

2.7 Route versus survey knowledge in wayfinding aids . . . 21

2.8 Landmarks . . . 25

3 Usability engineering **28**

3.1 Usability . . . 28

3.2 Accessibility . . . 30

3.3 Usability engineering . . . 32

3.4 Benefits of usability engineering . . . 34

3.5 Discount usability engineering . . . 34

3.6 The usability engineering lifecycle . . . 36

3.7 User involvement . . . 39

3.8 Qualitative research . . . 40

3.9 Qualitative interviews . . . 42

3.10 Personas and storyboards . . . 44

3.11 Wireframes, mockups and prototypes . . . 47

3.12 Usability evaluation . . . 47

4 Analysis phase **51**

4.1 A survey of existing solutions . . . 53

4.1.1 Evaluation criteria . . . 54

4.1.2 Evaluation . . . 55
4.1.3 Conclusions drawn from the evaluation . . . 56
4.2 Interviews . . . 60
4.2.1 A-priori assumptions before starting the interviews . . . 61
4.2.2 Planning the interviews . . . 61
4.2.3 Conclusions drawn from the interviews . . . 63
4.3 Personas . . . 65
4.4 Use cases . . . 67
4.5 Data needed for the wayfinding aid . . . 68
4.6 Integration with other systems . . . 70
4.7 Conclusions drawn from the analysis phase . . . 70

5 Design phase **72**
5.1 Storyboards . . . 72
5.2 Wireframes . . . 74
5.3 Floorplan overlays . . . 80
5.4 Designing the hypertext . . . 82
5.4.1 A motivation for dimensional data analysis . . . 82
5.4.2 Data dimensions of the indoor wayfinding scenario . . . 83
5.5 Map design . . . 86
5.6 The search function . . . 91
5.7 Designing interaction with rapid prototyping tools . . . 94
5.8 Designing interaction between map and sidebar . . . 96
5.9 Usability evaluation . . . 98

6 Presentation phase **100**
6.1 Users and use cases . . . 100
6.2 Interaction and content . . . 100
6.3 Technical aspects . . . 103
6.4 Future options . . . 105

7 Conclusions and future work **109**

7.1 Results . . . 109

7.2 Difficulties and lessons learned . . . 110

7.3 Future work, technically . . . 111

7.3.1 Requirements for a geographic software architecture . . . 112

7.3.2 Implications of different software architectures . . . 114

7.3.3 A survey of existing software products . . . 117

7.4 Future work, scientifically . . . 118

8 Appendix: Evaluated wayfinding systems **119**

8.1 The wegweiser.ac.at website . . . 119

8.2 University of Graz . . . 121

8.3 University of Innsbruck's "Student City Map" . . . 125

8.4 University of Karlsruhe's campus map . . . 127

8.5 University of Karlsruhe's 3D campus . . . 129

8.6 Technical University of München's roomfinder . . . 130

8.7 Herold.at's Strassentour . . . 132

9 Appendix: Personas **135**

10 Appendix: Storyboards **137**

11 List of Figures **140**

12 List of Abbreviations **142**

13 References **145**

1 Introduction

The purpose of this work is to develop a vision for a new *online campus wayfinding aid* for Vienna University of Technology. This includes the examination of theoretical backgrounds and the construction of a prototype. For a quick view on the results of this work, please refer to the executive summary at the beginning of this work, and to the conclusions in chapter 7 at the end of this work.

The scope of this work comprises a theoretical part about wayfinding, where relavant topics and related research are presented, and another theoretical part describing the background and methods of usability engineering. The scope of this work further includes a practical part, where a prototype for an online wayfinding application was created. Both the development process and the results are documented in this thesis. The prototype is limited in that it does not include maps and floorplans for every building and room and in that it is only installed on a development computer and does not need to be running on the university's servers. This work is further limited by what theory and methods are used: It does not contain the economic view on the situation, e.g. buying decisions or cost estimates. Nor does it contain the organizational aspect of how to manage the wayfinding software project within the university's institutions, practices and regulations.

The theoretical foundations and practical disciplines employed in this work are threefold:

1. *The geographic and cognitive theory of wayfinding aids*: Wayfinding relates to the cognitive act and physical movement a person needs to perform to get from one place to another. Wayfinding can be supported by a whole range of information sources: In addition to the immediate perception of the environment at hand, these may be a verbal description, the memory of a place, a paper map or an interactive, electronic device. Correspondingly, there are many different options on how wayfinding information can be provided, and using a visual map offered through the internet is just one way to help people find places. The theoretical part on wayfinding (chapter 2) is meant to serve as foundation for future work and shed light unto related aspects.

2. *A user-driven software engineering process*: It is crucial for an information system such as the wayfinding aid proposed in this thesis to provide the user with an easy, efficient and pleasant interaction. To reach this goal, a user-driven software engineering process shall be employed. Specifically, interviews with future users shall provide the developer with a vision of what is expected from the system, and user testing with early prototypes shall provide valuable feedback needed for the further design of the system. A user-driven software engineering process and its methods are first described theoretically (chapter 3) and then applied practically (chapters 4 to 6).

3. *The technology needed for web-based mapping applications*: For the actual construction of the envisioned system, a lot of technical knowledge is required. This includes programming languages and techniques such as HTML, JavaScript, Ajax, and SQL and software products and frameworks such as web servers, web application frameworks, search engines, WFS/WMS Servers and GIS software for both the desktop and the web. The prototype built in this work does feature a reasonable software architecture, but its primary focus is to demonstrate and explore possible user interfaces, as opposed to demonstrating technical feasibility, performance, or software architecture.

The structure of this work consists of two parts, one laying the theoretical foundation for the other: Sections two and three form the theoretical part describing wayfinding and usability engineering, thus laying a foundation for the practical part in sections four, five and six. These three chapters reflect the development phases of the software engineering process used in this work: Analysis, iterative design and implementation, and finally presentation of the resulting prototype.

The results of this work can be found in the following places:

- The executive summary at the beginning of this work takes a practical perspective, providing a quick overview of this work and suggestions for a follow-up project.
- The abstract at the beginning of this work takes a scientific perspective, summing up the purpose, methods and results of this work.
- The resulting prototype is described using text and screenshot graphics in chapter 6, ‘Presentation phase’, page 100.
- The conclusions drawn from this work and suggestions for future work are presented in chapter 7, ‘Conclusions and future work’, page 109.

2 Wayfinding

Wayfinding is an everyday activity for most of us because it is the activity we need to perform when we want to get from one place to another: While we are moving towards a goal, we need to observe our environment and use the knowledge we have about the world around us in order to recognize places. At certain points along the route, we need to decide which of several paths we want to pursue. We sometimes need to orient and re-orient ourselves to know where we are. When we are faced with unknown terrain, it can happen to us that we get lost. Some of the readers will know from their own experience that getting completely lost in unknown terrain can be very inconvenient and troublesome: "The very word "lost" in our language means much more than simple geographical uncertainty; it carries overtones of utter disaster" [Lynch, 1960, p.4]. For this reason (and for more practical reasons in everyday situations), humanity has a long tradition of making and using wayfinding aids such as maps, signs or verbal route descriptions.

Wayfinding research is concerned with how people navigate through space and orient themselves [Raubal and Winter, 2002, ch. 2.1]. The activity of wayfinding relates to a "purposeful, directed, and motivated *movement*" [Raubal and Winter, 2002, ch. 2.1] to a distant [Allen, 1999] *goal*. Wayfinding can be defined as the *cognitive act* necessary for a person to perform the aforementioned movement, but some authors also include the physical movement into the definition. The following section 2.1 takes a closer look at how wayfinding can be defined.

Wayfinding aids can use a whole range of different media as information source: In addition to our immediate perception of the environment at hand, possible media include a verbal description given to us, the memory of a place (possibly in the form of a 'mental map' we have in our head), a map we have once seen and are now remembering, or a map we are carring around on paper or using an electronic device. Using these media, two types of information can be communicated to the user: Survey information describes the surrounding environment whereas route information describes the segments and decision points that make up a route. There are thus many different options on how wayfinding information can be provided, and using a visual map offered through the internet is just one way to help people find places.

Wayfinding aids currently offered by universities or other institutions are in no way homogeneous, but a great variety of different designs and concepts can be found. To help gain a better understanding of these differences, a theoretical foundation is necessary. For this reason, theoretical issues related to wayfinding will be presented in this section.

2.1 Definitions of wayfinding

The word 'wayfinding' has first been coined by the urban planner Kevin A. Lynch. In his book 'The image of the city' [Lynch, 1960], he speaks of the different 'wayfinding devices' that help us not to get

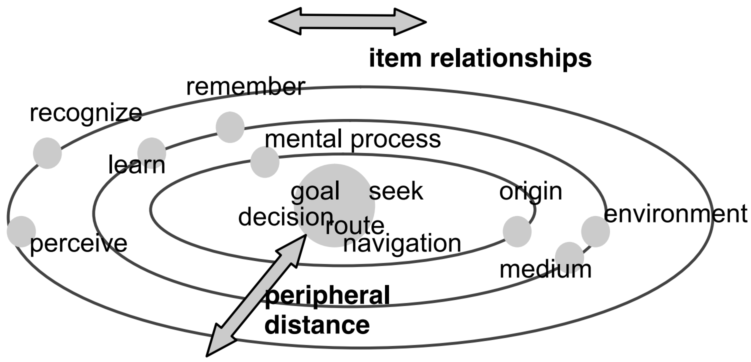


Figure 1: Defining wayfinding as a radial category. Taken from [Hochmair, 2002, fig.11]

lost in a city: maps, street numbers, route signs, bus placards, etc. Since its first use by Lynch, the word wayfinding has been increasingly used in the literature of diverse disciplines such as geography and environmental psychology [Allen, 1999].

In [Hochmair, 2002], the author provides an overview of current wayfinding definitions. Because these definitions are very different from each other but include recurring terms that are used by most definitions, it is suggested that the term 'wayfinding' is a *radial category*: "The terms that are more often used in the definitions define a central case of wayfinding (a kind of prototypical wayfinding), whereas more seldom mentioned features are extensions of the central meaning". A visualization of wayfinding seen as a radial category can be found in figure 1, 'Defining wayfinding'. From this figure, the central terms defining wayfinding can be seen to be decision, goal, seek, route and navigation. From these more central concepts of the radial category, the author derives a set of axioms as minimum requirements to call an activity 'wayfinding'. These axioms are described both formally using the Haskell programming language and informally using natural language as follows:

- Wayfinding includes *spatial decisions*. At certain decision points along the way, the wayfinder needs to choose from a given set of paths in a network.

- Wayfinding involves a *goal*. A wayfinder has reached his or her goal when the actual perceived position equals the goal position.

- A wayfinder tries to *reach that goal*. Activities with a different intention may not be called wayfinding.

- Wayfinding (here understood as comprising both mental decisions and physical movement) has *no impact upon the environment*. While this axiom may not be true for a real person, the axioms have been formulated with an agent-based simulation in mind. In this simulation, the agent does not change his environment by any of his wayfinding activities.

- Wayfinding *actions are ordered*. Examples for wayfinding actions are the decision to take a certain path or the movement along a chosen path. Permutating a sequence of actions can lead to an invalid sequence with non-connected locations.

A similar attempt to formally define the term wayfinding is found in [Pontikakis, 2006], where the view on wayfinding is extended to include all three phases of a trip (conception, planning, and execution). In addition to that, the business aspect of ticket purchase in public transport is also considered. An agent-based simulation of urban wayfinding including walking and public transport is then presented. Through the implementation of the simulation by a Haskell program, this work also contains a formalization of the wayfinding domain.

Starting again from the intentionally fuzzy definition of wayfinding as a radial category, we can reach a more precise understanding by contrasting wayfinding to other activities that are different from wayfinding:

- Wayfinding primarily means the *cognitive act* (when and while a person tries to find his or her way from one place to another), not the physical movement [Cheung, 2006]. In other definitions [Hochmair, 2002], wayfinding does include the physical movement, but it always also includes the cognitive act.

- Wayfinding is different from *route finding* [Cheung, 2006]: Route finding is about choosing the best route from A to B. Wayfinding, on the other hand, is about the activities a person needs to perform while following a given route.

- Wayfinding is different from *navigating* [Hochmair, 2002, p.30]: The difference lies in the way the surrounding environment is seen. Human movement in a continuous, open space involves navigation. Wayfinding, on the other hand, involves selecting paths from an existing network and thus requires an environment with discrete paths to choose from.

- Wayfinding relates to a *distant goal* that cannot be directly perceived [Allen, 1999]. It follows that simple pointing and gestures are often not satisfactory to communicate wayfinding information because of the amount of route or survey information that needs to be communicated.

2.2 Spatial memory and cognitive maps

Since the activity of wayfinding is strongly related to mental activities (see section 2.1, 'Definitions of wayfinding', page 9 for how the mental activities are included into the definition of wayfinding), the examination of these cognitive aspects plays an important role. In [Montello, 2002], a historical overview of cognitive research related to map design is given: Since the twentieth century, a lot of work has been done trying to understand human cognition in order to improve the design of maps and other wayfindinding devices. The author describes the relation between human cognition and maps as follows: “Cartographers have long realized that maps do not present the world directly and transparently. Maps *re*-present the world by providing versions of truth for human minds to apprehend. In turn, minds represent the world too, internally as ‘cognitive maps’ ”. It follows that if we want to communicate wayfinding information, we actually want to change and shape and enrich the cognitive map within another person’s mind. In his book ‘The image of the city’ [Lynch, 1960], Lynch also stresses the importance of cognitive maps, i.e. of the mental image of the world around us that we each hold as an individual: “In the process of way-finding, the strategic link is the environmental image, the generalized mental picture of the exterior physical world that is held by an individual. This image is the product both of immediate sensation and of the memory of past experience, and it is used to interpret information and to guide action. The need to recognize and pattern our surroundings is so crucial, and has such long roots in the past, that his image has wide practical and emotional importance to the individual.” [Lynch, 1960, p.4]

Maps are amongst the most important wayfinding devices. For their design, cognitive cartography applies the theories and methods of psychology because it sees maps as cognitive devices providing input to mental worlds. Cartography can thus be understood “as a discipline that attempts to pass along the cartographer’s conception of the world to the mind of the map reader via the symbolic medium of the map” [Montello, 2002]. This conception of cartography led to the the *communication model*, which in its simplest form portrays maps as information channels that transmit information from a source to a recipient (similar to the model of technical communication developed by Shannon and Weaver). A simple and a more complex graphical representation of this model are shown in figure 2, ‘The Communication model in map design’, page 13. The more complex version elaborates on what happens within the boxes of the simpler model, namely the encoding and decoding of geographic information using maps as a language. In the more complex model, it is furthermore recognized that the process of encoding and decoding is influenced by the sender’s and receiver’s prior knowledge. The resulting process of how a map can be used to communicate spatial information can be summed up like this [Koláčcný, 1969]:

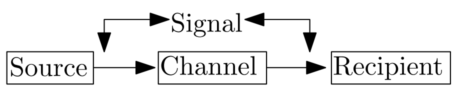
1. Selective observation of the reality U_1 by the cartographer, with the intention of creating cartographic information I_c.

2. As a result of the observation, the cartographer now possesses the selective information I_s as a multidimensional, intellectual model of reality.

3. The cartographer transforms his selective model of reality into cartographic language L, which results in a two-dimensional, intellectual model of the cartographic information I_c.

4. The cartographer objectifies his two-dimensional model e.g by drawing a map.

a) A simple communication model



b) A more elaborated communication model

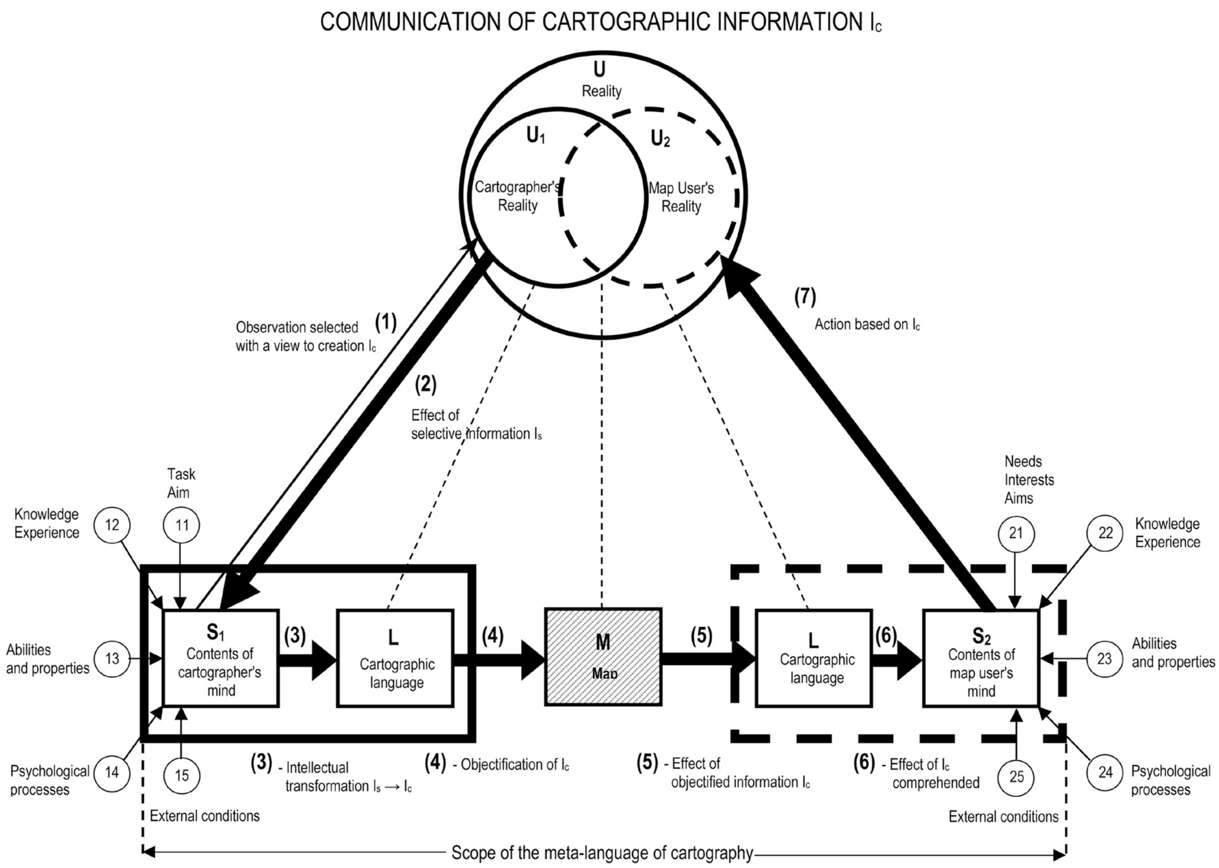


Figure 2: The communication model in map design: (a) A simple framework for the communication model, and (b) a more elaborated version taken from [Koláčcný, 1969].

5. The map user works with the map by reading the objectified cartographic language.

6. Comprehending the cartographic language changes the map user's mind: He creates a multidimensional model of reality and experiences this new reality.

7. The map user acts upon his widened experience of reality $U_{2.}$

Maps are just one possible way of objectifying cartographic information, and other wayfinding devices than maps have been examined with regard to human cognition, e.g. verbal route instructions [Allen, 2000; Tom and Denis, 2003; Richter and Duckham, 2008] or tactile [Heuten et al., 2008] or auditive [Dodiya and Alexandrov, 2008] wayfinding aids. From an urban planning [Lynch, 1960] or architectural [Hölscher et al., 2005] perspective, the environment itself can be understood as a cognitive device providing input to mental worlds - in this context, Lynch speaks of the 'legibility' of a city.

Since, according to the communication model, wayfinding devices are meant to communicate spatial information, the question of how spatial knowledge is learned and acquired is of central importance. According to [Cheung, 2006] and following the view expressed in [Norman, 2002], *knowledge in the world* is sensored, interpreted and added to prior known *knowledge in the head*. Knowledge in the world is not only gained through direct interaction with the environment, but through different medial representations as well, such as maps, pictures or drawings. A similar view is expressed by [Lynch, 1960, p.4]: The mental image that we hold of our environment is the result of both our *memory of the past* and of *immediate sensation*.

The metaphor of the mental or cognitive map is just one amongst several different metaphors for how spatial knowledge is stored. This is due to the *character of spatial knowledge*:

- Spatial knowledge includes *landmark, route, and survey knowledge*. This knowledge was originally thought to be acquired sequentially, but it was later shown that this is not strictly the case, see [Hirtle and Sorrows, 1998] for an overview. Landmarks are still thought to be learned early in the process of spatial knowledge acquisition [Richter, 2007]. (See section 2.8, 'Landmarks', page 25 for more about the importance of landmarks in wayfinding).

- Spatial knowledge is *multimedial*. It may include knowledge chunks such as images of a place, the general layout of streets, knowledge about the color codings of subway lines, and so on [Hirtle and Sorrows, 1998]. The same work also contains implications from this finding for the design of wayfinding systems.

- Spatial knowledge includes *visual and somatosensorical data*. This is due to how spatial knowledge is learned: Since the immediate sensation provides multimodal input (visual and somatosensorical), neuro-psychological findings suggest that the resulting knowledge is multimodal in nature, too [Engel et al., 2005, ch.2.1].

- Spatial knowledge can be characterized as being *distorted, fragmented, and incomplete.* It may contain different, even contradicting representations. When spatial knowledge is recalled, missing pieces are filled in based on assumptions that are likely to be true. [Barkowsky, 2001]

- It is stored in a *hierarchical* manner with different resolutions [Barkowsky, 2001].

Because of these characteristics of spatial knowledge, scientific literature has come up with a whole range of different metaphors for spatial knowledge: Amongst these metaphors are the 'cognitive collage', the 'cognitive atlas', and the 'spatial mental model' conception. See [Engel et al., 2005, ch. 2.1] for an overview and [Barkowsky, 2001] for a more in-depth discussion.

2.3 The orientation of maps

The orientation of (both physical and mental) maps is of great practical importance for the task of wayfinding and for the construction of wayfinding devices because people have difficulties to align the orientation of their mental map with a physical map or other wayfinding device and with the environment [e.g. see Cheung, 2006, ch 7.1].

In [Kelly and McNamara, 2008], an overview on spatial memory and spatial orientation is given: A distinction is drawn between longterm spatial memory and shortterm sensorimotor working memory. One difference between the two types of memory is their respective spatial orientation. *Longterm memory* contains what can be described using the cognitive map metaphor, a rather static image of the environment. It allows the navigator to plan routes and recognize previously encountered environments. The orientation of longterm memory is relative to a few reference directions that are chosen during learning. This choice of reference system depends upon the body position during learning and upon the environment's structure. *Shortterm sensorimotor memory* on the other hand is used to coordinate immediate action within the environment such as negotiating obstacles, intercepting moving objects or steering a straight course towards a goal. It uses a bodycentric perspective that is continuously updated while a person moves. Because the continuous updating of object locations is cognitively demanding, only a limited number of objects can be traced in shortterm memory.

From these theoretical findings about the orientation of mental maps and from practical experiments, the following *practical considerations* concerning the orientation of maps have been made:

- The mental map a person keeps in mind is typically north-orientated for outdoor maps (probably because this eases comparison with street maps), but not so much for indoor maps. [Gartner and Radoczky, 2006]

- People tend to turn paper maps to avoid mental rotation. [Cheung, 2006, ch 7.1] and [Gartner and Radoczky, 2006].

- Electronic, interactive wayfinding aids are generally more efficient if they provide ego-centric instead of north-oriented maps [Gartner and Radoczky, 2006].
- When re-orientation is necessary in an urban environment, such as when a person comes out of a subway station, photos work better than maps [Ishikawa and Yamazaki, 2009].
- If verbal instructions are given instead of a map, they work better in relative than in absolute frames of reference, e.g. 'turn to the left' is preferred over 'turn to the west' [Ishikawa and Kiyomoto, 2008].

2.4 Indoor wayfinding

Indoor wayfinding is in several ways different from wayfinding in outdoor situations such as open spaces, urban spaces, road networks or other spaces that are otherwise often considered in wayfinding research. Exemplary situations where indoor wayfinding aids may be useful include big public buildings, airports or shopping malls. A historical overview on research specific to *indoor* wayfinding is given in [Hölscher et al., 2005, ch2]. The bottom line is that the special characteristics of indoor space inevitably also lead to special requirements for indoor wayfinding systems:

- *Cellular space:* While both indoor and outdoor space can be described using euclidean coordinates, another system is often used to refer to indoor places: Using the notion of symbolic space or cellular space, one may refer to an indoor place using a room number, thus identifiying a 'cell' of indoor space by providing the cell's symbolic name [Li, 2008]. This observation is of particular importance for the design of an online campus wayfinding aid since users may want to search for a location using a room number or the name of a lecture hall.
- *Constraints by architectural elements:* Indoor space is different in that it is to a high degree constrained by architectural elements such as floors, stairs, and walls. Within these contraints however, a pedestrian is able to roam freely. This is in contrast to how public or private transport networks are often modelled: In these networks, vehicles cannot roam freely within certain boundaries, but are strictly bound to following the network's paths. [Stoffel et al., 2007, ch1]
- *Modelling of indoor space:* If an electronic system is to provide dynamically calculated wayfinding advice for indoor situations, it must have an understanding of the indoor space's structure. Hence, a data model suited to describe indoor space is needed [Li, 2008]. There are several options how indoor space may be modeled [Stoffel et al., 2007, ch1], [Becker and Dürr, 2005]: (a) Cognitive models try to capture human thinking and are therefore suited for the generation of route descriptions. (b) Topological models like Region Adjacancy Graphs, Cell and Portal Graphs, and hierarchical models abstract away from geometric details and only preserve topological relationships. (c) Floor plans on the other hand document the detailed structure of a

building in terms of its boundaries. From these boundaries, topological models may be derived in an automatic process [Stoffel et al., 2008, 2007].

- *3D - Aspect:* Indoor space has the 3D aspect of multiple interconnected floor levels. The importance of this aspect is highlighted by the work of [Cheung, 2006], where the author found that the main difficulty of test users in a wayfinding experiment was the interpretation of three-dimensional routes represented on a two-dimensional map. The author concludes that verbal instructions work better than maps for three dimensional movements.

- *Indoor wayfinding strategies:* Several strategies or heuristics for dealing with a building's 3D structure are employed by test users in experiments [Hölscher et al., 2005]: (a) central point strategy: sticking to central, known points and corridors. (b) direction strategy: go into the right direction, then into the right floor. (c) floor strategy: go into the right floor, then into the right direction. For the design of an indoor wayfinding system, it may be helpful to find out which indoor wayfinding strategy is best suited given a building's structure. The wayfinding information offered by a wayfinding aid should then be structured accordingly.

- *The influence of architecture on indoor wayfinding:* As already described in section 2.2, 'Spatial memory and cognitive maps', page 11 in the context of spatial cognition, architecture influences wayfinding - in fact, a whole city [Lynch, 1960] or the architecture of a building [Hölscher et al., 2005] can be understood as a wayfinding aid. On the other hand, architecture can also hinder spatial learning and wayfinding. For example, [Abu-Ghazzeh, 1996] describes how too much similarity between buildings and hallways can hinder orientation.

- *Indoor geolocation:* Locating devices in indoor situations does not work well using outdoor geolocation techniques (e.g. using a global navigation satellite system such as the GPS 'Global Positioning System' or by triangulating the signals of a mobile phone network). Alternative techniques are therefore used in indoor situations. An overview of such techniques is given in [Mautz, 2009; Cope and Jorgenson, 2009b]. Examples of how indoor wireless LAN based geopositioning is used to provide a wayfinding aid can be found in [Cope and Jorgenson, 2009a; Linge, 2009].

2.5 The medium used for wayfinding aids

In addition to knowledge that can be derived by the wayfinder by directly observing the environment, different media and representations may be used to communicate wayfinding information. Maps are widely used to support wayfinding. Arguably, they are most suited to help with wayfinding because maps "are still the best form of giving an overview of an area", [Gartner and Radoczky, 2006, p.8]. But maps are not the only representation suited as a wayfinding aid: Verbal (acoustical or textual) representations can also be used to describe a location or route [Giudice et al., 2007]. In fact, it has been shown in [Cheung, 2006] that textual route descriptions may in some situations be superior compared to visual maps. Other possible media not mentioned so far include virtual environments, annotated images, recorded video, navigation arrows as on a GPS device.

Considering the *medium* used to represent and communicate a route or location, wayfinding aids can, for the purpose of this work, be classified as follows (compare with [Gartner and Radoczky, 2006]):

- *In-place wayfinding aids*:
  Knowledge in the world, including architecture, landmarks, signs, the affordance of things.

- *Visual wayfinding aids*:
  Including photos, drawings, street maps, topographic maps or satellite images

- *Verbal wayfinding aids*:
  Including text in written or oral form

- *Other wayfinding aids*:
  Including multimedia, animation and augmented reality

In [Cheung, 2006], three wayfinding aids, each using a different medium, have been compared in a practical experiment, see figure 3, 'Different media used as wayfinding aid'. In the experiment, thirty participants were to follow the instructions given by their wayfinding aid system running on a PDA. The first group of participants were using an *interactive map*, the second group of participants were using *photos* that had been annotated with arrows and verbal instructions including street names, and the third group of participants were given a *verbal route description*. In the study, none of the groups performed better than the others in all situations, but the overall performance of the group given verbal instructions was significantly better. The author concludes that different media perform better for different environmental conditions and different wayfinding tasks. Despite the lack of a thorough comparison including different environmental conditions and wayfinding tasks, the following observations relating to the environment and wayfinding task have been made:

- The performance of the verbal system was significantly better than that of the other two systems, namely an interactive map and annotated photos [Cheung, 2006, ch. 7.3].

- "Map-based and image-based systems performed better in navigating the participants through open space or places that are lean in landmarks" [Cheung, 2006, ch. 8], the reason for this being that maps or other detailed representations provide a lot of survey knowledge, while verbal instructions need to rely upon landmarks to describe the route.

- "The main difficulty for wayfinders using the map system is the interpretation of three dimensional routes represented on a two dimensional map" [Cheung, 2006, ch. 7.1]. This problem is especially relevant for this work since the campus of Vienna Technical University includes indoor navigation in multifloor buildings.

	Advantages	Disadvantages
Map-based System	• Rich in information	• Requires initialization • Requires longer cognition time • Problem in displaying three dimensional information in its current form
Image-based System	• Realistic description of spatial features • Spatial configuration is embedded • Geometric instructions can be embed into instruction arrows	• Vulnerable to landmark changes • Lack of visual depth as one of the main problems • Obstructions of foreground objects • Difficulty in maintaining a flow between images
Verbal-based System	• Abstract, Direct Instruction • No dimensional barriers • Can be geometry driven or task driven	• Efficiency plummets in open space or lean environment

Figure 3: Different media used as wayfinding aid: A comparison. Taken from [Cheung, 2006, Table 2]

Another study [Ortag, 2005] comparing spoken instructions with maps came to the same conclusion that the groups using the two media performed similarly well.

The inclusion of different media into a wayfinding aid is beneficial, and the redundancy that is thereby introduced is necessary for a good wayfinding aid [Hirtle and Sorrows, 1998; Hirtle, 2000]. In the before-mentioned work, the author demonstrates how different media can be used at once through a prototype that includes maps, floor plans, photographs and verbal route descriptions. A screenshot of this system is shown in figure 4, 'Different media combined in one wayfinding aid'.

2.6 The necessary amount of detail in wayfinding aids

A wayfinding aid needs to describe a place or a route. The designer of a wayfinding aid is therefore concerned with the same questions a cartographer would typically ask when making a map: Who is the intended audience? What data needs to be visualized or otherwise expressed? How can we represent qualitative, quantitative or hierarchical data? How much abstraction is needed for the specific purpose at hand? In [Krygier and Wood, 2005], these questions are elaborated from a mapmaker's perspective in a very visual, easy-to-understand way with a lot of examples. Amongst these questions, one with major importance is about the necessary amount of detail that needs to be included in a wayfinding aid. The process of "*abstraction from unnecessary detail* to concentrate on essential information" [Meilinger et al., 2006, emphasis mine] is called map schematization. As can be seen in figure 5, 'Map schematization', it may include straightening lines, simplifying angles, distorting proportions and omitting or re-arranging objects. Examples for highly schematized maps found in everday life

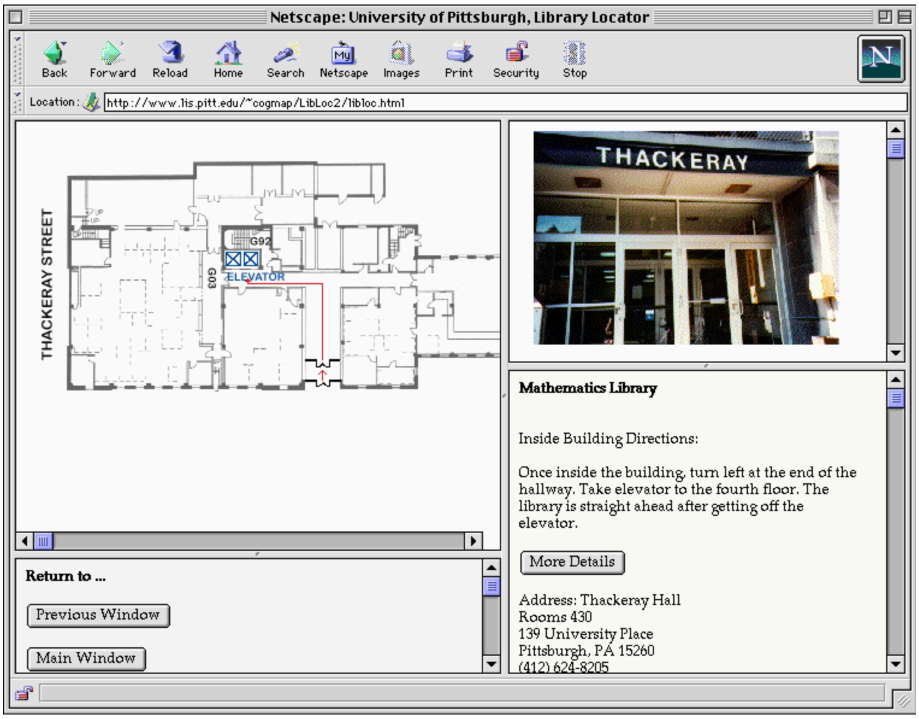


Figure 4: Different media combined in one wayfinding aid: The prototype features maps and floor plans, photographs, and verbal route descriptions. Taken from [Hirtle, 2000, fig 4].

include city maps (usually featuring a simplified street graph) and public transport maps (featuring a simplified version of the transport network).

The advantage or motivation for map schematization is described in [Klippel et al., 2005, abstract]: Instead of conveying every detail of the environment to the user, thus burdening him with the task of deriving the information of interest, map schematization looks for more abstract, cognitive ways of directly communicating this information.

In [Gartner and Radoczky, 2006, p.4], the necessary amount of detail to be included in a map that is to be used for a wayfinding task is examined in an experiment. The authors find that schematic, highly abstracted maps work as well as more detailled street maps. The authors of [Meilinger et al., 2006] come to a similar conclusion: Unambiguous turning information in a highly schematized map is more useful than the survey knowledge offered by a more detailed, less schematized map.

Too much abstraction can also lead to bad wayfinding performance: The authors of [Gartner and Radoczky, 2006, p.8] find that schematic maps provide a good topographic overview but often lack details necessary to stay on the right track. The authors recommend to enrich a simplified, schematic map with certain features that are essential for the wayfinding task at hand. Features of this kind may include street names and landmarks such as distinctive buildings or monuments.

In [Berendt et al., 1998], the problem of using a schematized map is demonstrated in an example,

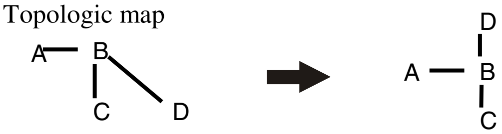
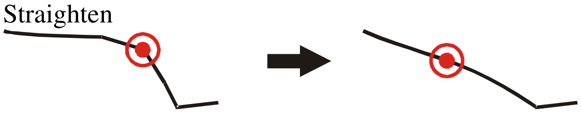
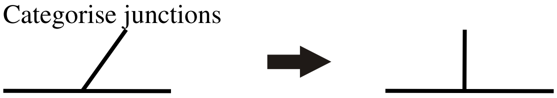
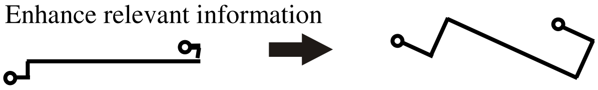
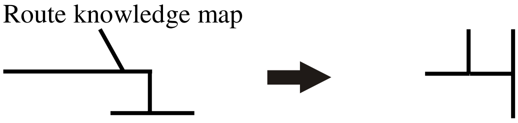
Schematisation principle		Survey information	Route information
Topologic map		incorrect	incorrect
Straighten		rather correct	correct
Categorise junctions		rather incorrect	correct
Enhance relevant information		start & go-al correct	correct
Route knowledge map		incorrect	correct

Figure 5: Examples of map schematization. Taken from [Meilinger et al., 2006, Table 1]

where a schematized map is used together with a more detailed map in order to solve a given wayfinding task. In the example, the information contained in the two maps needs to be combined by the user in order to solve the task. The combination of information is hindered by the fact that it is not always clear if some information is contained within the schematized map. E.g. 'Have the distances been preserved in the schematic map? Have directions and angles been preserved? Have some details been left out? If yes, what has been left out and what has been preserved?'

Schematization can not only be applied to maps, but to other forms of wayfinding aids as well: In the same way as with maps, some information is simplified and some is left out in order to only provide the information necessary for the task at hand. For example, a complex sequence of turns may in some cases be replaced by one simple instruction saying "traverse this area" [Hirtle, 2000]. The same instruction may be annotated to a map using one large arrow as shown in figure 6, 'Route schematization'. In a similar way, schematization is applied to verbal route descriptions in [Srinivas and Hirtle, 2008]: Depending upon the user's prior familiarity with a given route segment, more or less information can be omitted in the schematization process.

2.7 Route versus survey knowledge in wayfinding aids

Wayfinding can be supported through two types of information [Meilinger et al., 2006]: Survey information and route information. *Survey information* can be communicated through topographic maps or other "realistic" representations of the environment of a certain place (e.g. a hiking map

Figure 6: Route schematization of complex route instructions: Instead of many complicated turns, one large arrow indicates that this area needs to be crossed. Taken from [Hirtle, 2000, fig 5].

or an orthographic aerial photo). These representations typically distort angles and lengths as little as possible and contain a lot of detailed information. As described in the previous section, some of the information found in a detailed map can be omitted for a given wayfinding task. If however the resulting map is to be used for a wayfinding task, the map must still provide correct *route information*.

Richter and Klippel provide a definition of route information [Richter and Klippel, 2004] as follows: Route information provides instructions on how to get from A to B, i.e. a route description is a "task-oriented specifications of the actions to be carried out to reach the destination". A route description can be abstract in the sense that it is independent of the medium (e.g. map, verbal instructions, gestures) used to later communicate it, and can later be externalized to fit the modality of the chosen medium. In [Srinivas and Hirtle, 2008], a refined and more formal definition is given. According to this definition, a route consists of an origin and a destination, connected by a series of route segments and *decision points*. At each decision point, the wayfinder needs to be provided with information to support his decision making. In [Rehrl et al., 2009], the type of information that needs to be given has been classified into 'action schemes', see figure 8, 'Route modelling using decision points and motion concepts', page 24.

The type of information that should best be provided at each decision point is subject to a vivid debate. This information can be provided as turning information or with the help of landmarks.

Turning information has been closely examined in [Klippel, 2003]: Route descriptions are understood as a combination of the elements of one basic set of instructions such as 'turn left', 'turn right', 'turn half left', etc. (with the latter instruction meaning a turn to the left in a 45 degree angle). For this basic set of instructions, the author coined the term *wayfinding choremes*. The cognitive, functional model underlying this set of instructions is that of an 8-directional star, see figure 7, 'Route descriptions using prototypical, 8-directional turning concepts', page 23. Through an empirical study,

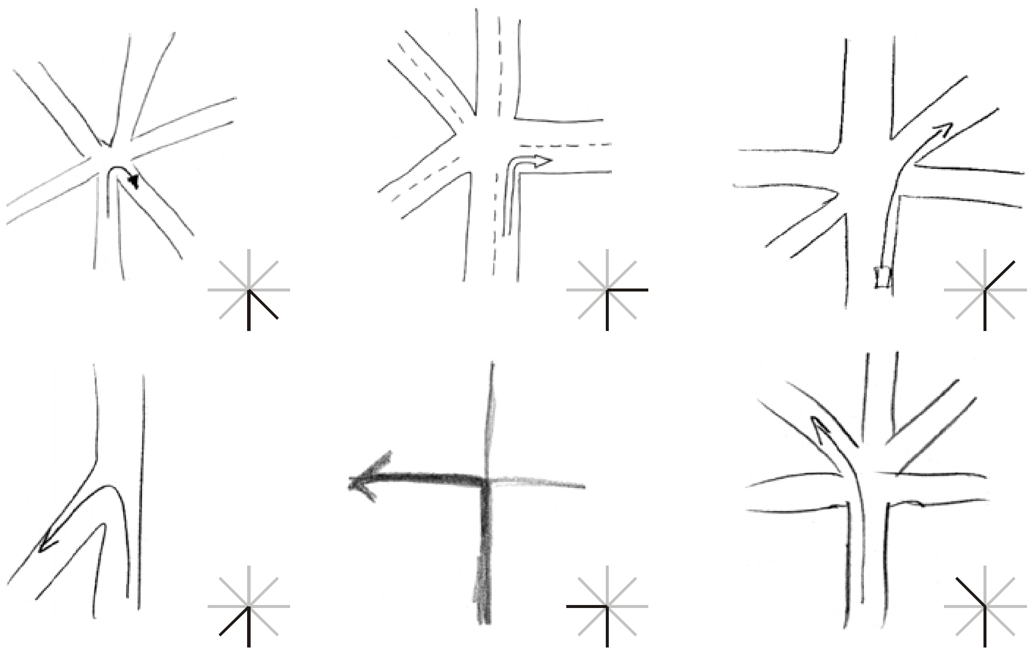


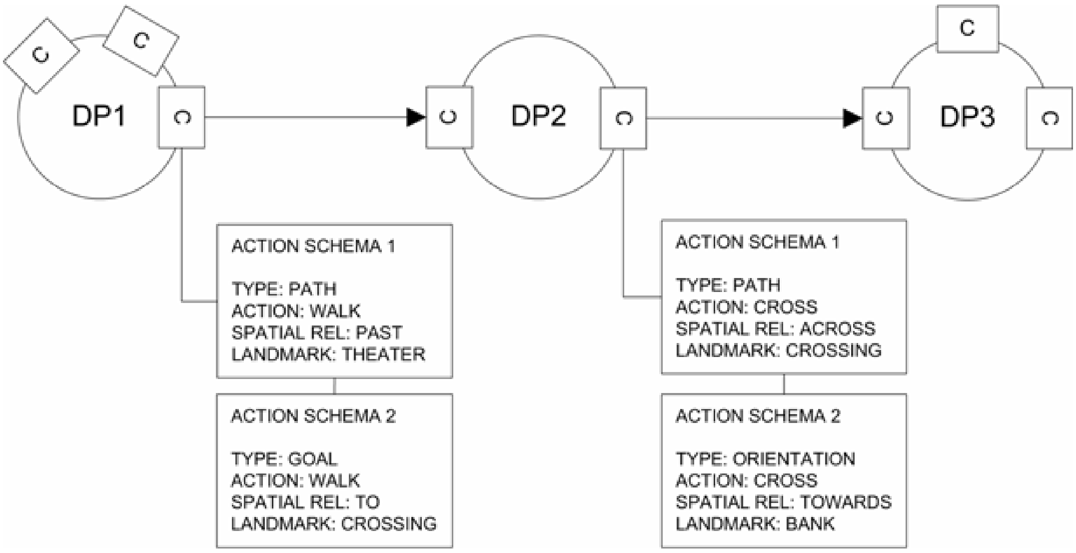
Figure 7: Route descriptions using prototypical turning concepts (handdrawn) with their according wayfinding choremes (computer-drawn, bold lines) within the 8-directional star model. Taken from [Klippel, 2003, fig.6]

the author found that this 8-directional star *functional* model is constantly employed by people - even when the *structural* model of the real-world situation is different, as for example in the situation of a 5-directional star crossing, see figure 7, 'Route descriptions using prototypical, 8-directional turning concepts', page 23. Continuing from these findings, [Klippel et al., 2005] suggest to schematize route instructions by replacing exact and unsimplified turning information by the corresponding abstractions from amongst the wayfinding choremes.

Landmarks are often used instead of turning information in everyday language (see section 2.8, 'Landmarks', page 25): An empirical study by [Rehrl et al., 2009] examined the type of instructions given in real-world situations. For this purpose, the authors classified more than 7000 verbal instructions to find often used combinations of direction and motion concepts. The thirteen most popular instructions are shown in figure 8, 'Route modelling using decision points and motion concepts', page 24, suggesting that people most often refer to landmarks when giving route instructions. The authors are aware of the fact that this set of instructions may not be complete. It may however serve as a reasonable starting point because it is founded in empirical observation, and the authors plan [Gartner et al., 2009] to further extend it into a formal, systematic model that is not limited to describing the test routes. In a second step it is planned to use this model in a prototypical navigation device.

A route description often does not need to describe every detail to the user. Some information can be simplified or left out. This process of simplification is called *abstraction* and has been described in

a) Formalization of a route description: Decision points and action schemes



b) Most frequent combinations of spatial and motion concepts

NO	**VERB**	**RELATION**	**USEAGE**	**LANDMARK**	**FREQ**
1	WALK	TOWARDS	ORIENTATION	Yes	213
2	WALK	ALONG	ORIENTATION	Yes	145
3	TURN	INTO	PATH	Yes	108
4	TURN	TOWARDS	ORIENTATION	Yes	94
5	WALK	TO	GOAL	Yes	94
6	WALK	INTO	PATH	Yes	94
7	COME	TO	GOAL	Yes	85
8	WALK	PAST	PATH	Yes	68
9	CROSS	ACROSS	PATH	Yes	68
10	WALK	AHEAD	ORIENTATION	No	60
11	LEAD	TO	GOAL	Yes	58
12	WALK	THROUGH	PATH	Yes	51
13	WALK	ON	TOPOLOGY	Yes	49

Figure 8: Route modelling using decision points and motion concepts: (a) Route modelling: At each decision point *DP*, the wayfinder is faced with a set of route choices *C*. Following the instructions specified as an ordered set of *action schemes* leads to the next decision point. (b) Combinations of motion and spatial relation concepts that most frequently used in route descriptions. Taken from [Rehrl et al., 2009, tbl.3 and fig.7]

the previous section 2.6, 'The necessary amount of detail in wayfinding aids', page 19.

A lot of work has been done to improve the *automatic generation of route descriptions*. In [Richter and Klippel, 2004], the choice of the best route description is presented as an optimization problem, with the goal of minimizing the number of distinct parts needed to describe the route. Other authors have chosen other optimization goals: In [Srinivas and Hirtle, 2008], the approach is to generate good route descriptions by adapting the route segmentation to the knowledge of the wayfinder. In [Richter and Duckham, 2008], the aim is to find easy-to-describe routes using path selection with simplicity as criterion. The use of landmarks in general and their use for route descriptions in particular is treated in the following section 2.8, 'Landmarks', page 25.

2.8 Landmarks

Landmarks play an important role in wayfinding because they provide a means for "locating oneself and establishing goals" [Sorrows and Hirtle, 1999]. The key characteristic of a landmark is its *singularity* regarding the landmark's physical characteristics and its *memorability* [Lynch, 1960, p.78]. Put more simply, "everything that stands out of the background may serve as landmark" [Richter, 2007, ch.2]. In [Sorrows and Hirtle, 1999], the authors further elaborate the characteristics of landmarks:

Singularity: One characteristic that can make an entity a landmark is its singularity, i.e. its sharp contrast with its surroundings. For example, a building may stand out because of its size, style or age.

Prominence: The structural prominence of a spatial location may make it suitable for the use as a landmark. This might be a building that stands in a central location connecting many roads and that is visible from a wide area. Distant, but prominent locations such as a mountain range that can be seen from everywhere in a city may also serve as a landmark.

Significance: An entity may be a landmark because of its content, meaning, use or cultural significance. For example, even a rather normal-looking building in a non-prominent location may still serve as landmark if it is well-known for historical reasons.

Prototypicality: Entities may be used as landmark because of how typically they represent a category. For example, one may refer to a church building as 'the town church' if the particular church building meets the expectation of what a typical church building looks like and how it is located in the town's center.

From these characteristics, Sorrows and Hirtle derive three types of landmarks:

- *Visual* landmarks stand out because of their visual singularity.

- *Structural* landmarks stand out because of their prominent location within the structure of the environment.

- *Cognitive* landmarks stand out because of their significance to wayfinders or because of their prototypicality (how well they represent a certain category).

In [Richter, 2007, ch2], another categorization of landmarks is developed using the landmark's position relative to the route as criterion, see figure 9, 'A classification of landmarks', page 27. In addition to this categorization, an overview of landmarks definitions is also given, and further distinctions of landmarks are introduced in both [Richter, 2007, ch2] and [Gartner and Radoczky, 2006]:

- *Local / distant / global landmarks*: Landmarks may have different positions relative to a place or route [Richter, 2007, ch.2]. As can be seen in figure 9, 'A classification of landmarks', page 27, landmarks can be categorized using this criterion: Local landmarks are only useful for their immediate environment. Distant and structural landmarks are landmarks that are useful despite the fact that they are located in some distance to the place or route described. Global landmarks are out of the wayfindinder's navigational space, but still provide orientation.
- *Abstraction* of a landmark's physical shape: In many cases, the representation of a landmark's entity in the real world is abstracted to a single point in space. In [Richter, 2007, ch.2], the definition of a landmark is extended to also allow lines and areas.
- *Emotional landmarks*: An entity may serve as landmark if there is an emotional or otherwise personal connection between the entity and the wayfinder. [Gartner and Radoczky, 2006]
- *Active and passive landmarks*: Landmarks are usually passive towards the wayfinder. An electronic device that actively communicates with the wayfinder using wireless technology [Gartner and Radoczky, 2006] or by producing sound [Baus et al., 2007] may also be used as a landmark. For this situation, Gartner and Radoczky coined the term 'active' landmark.

In [Richter, 2007], the importance of landmarks for wayfinding tasks is stressed: Landmarks are learned early in the process of spatial knowledge acquisition (see section 2.2, 'Spatial memory and cognitive maps', page 11 for more about the learning of spatial knowledge). Because of this, chances are that wayfinders will still recognize landmarks on lesser-known terrain. Landmarks also play an important role for the organization of spatial knowledge: Evidence suggests that spatial knowledge is organized hierarchically around landmarks that serve as anchor points. Besides their importance for the acquisition and organization of spatial knowledge, landmarks are of major importance for the communication of routes and directions:

Several studies show how the use of landmarks (as opposed to wayfinding systems that only provide survey knowledge or only provide distance-to-turn information) improves both vehicle [May and Ross, 2006] and pedestrian [Tom and Denis, 2003; Ross et al., 2004] navigation. An empirical study [Rehrl et al., 2009] shows that when people give route instructions, they most often refer to a landmark. In

Local, path and route	**Distant, Structural, environmental**	**Global Landmarks**
egocentric references	edges	cardinal directions
landmarks at decision points	districts	global landmarks
landmarks between decision points	slant	
distant landmarks		
linear and areal landmarks		
path annotations		

Figure 9: A classification of landmarks into local, distant, and global landmarks. The categorizing criterion is the landmark's position relative to the route. Figure redrawn from [Richter, 2007, tbl.1]

[Fontaine and Denis, 1999], the use of landmarks in the indoor, underground situation of an urban public transport network is examined. The authors conclude that landmarks (including signs) are preferred for underground situations. In [Omer and Goldblatt, 2007], the aspect of inter-visibility of landmarks is examined. The authors' hypothesis is that landmarks with an overlapping field of view ease wayfinding. This hypothesis is confirmed through an experiment using a virtual 3D environment.

A lot of effort has gone into the automatic generation of landmarks (i.e. choosing real-world entities to be used as landmark within a route description for a given wayfinding task). The above quoted categorization of landmarks into visual, structural and cognitive landmarks [Sorrows and Hirtle, 1999] has been used to automatically calculate the landmark saliency of an object: An object's landmark saliency measures how well the object is suited for being used as a landmark [Raubal and Winter, 2002].

Several theoretical aspects related to wayfinding have now been considered. The following section describes an approach to software engineering that focuses upon usability. Taken together, wayfinding theory as described in this section and usability engineering as described in the following chapter serve as a foundation for the practical part of this work.

3 Usability engineering

Usability engineering methods are applied in this work with the goal of building a wayfinding system that is 'usable', i.e. that is easy to learn and efficient to use. This user-centred approach to geographic hypermedia systems is in line with what is recommended in [Wealands, 2006]. This chapter gives a closer definition of usability and describes the methods of usability engineering. The application of these methods and principles can be found in the practical part of this work: analysis phase (chapter 4), design phase (chapter 5), and presentation phase (chapter 6).

3.1 Usability

Usability measures the "effectiveness, efficiency and satisfaction with which specified users achieve specified goals in particular environments" (ISO 9241). The term 'usability' is now used instead of the term 'user-friendliness' because users don't need machines to be friendly, but easy to learn and efficient to use [Nielsen, 1993]. The professional field dealing with usability is known under names like CHI (computer-human-interaction), HCI (human-computer-interaction), UCD (user-centered design), UID (user interface design), and UXD (user experience design). Jakob Nielsen's book "Usability Engineering" [Nielsen, 1993] further elaborates the definition of usability:

Usability contributes to system acceptibility: See figure 10, 'Usability and other attributes of system acceptibility', page 29 for how usability is just one of several other factors contributing to the acceptance of a system. Other factors include: A system's utility, i.e. whether the system's functionality helps with what the user wants to do. Cost, reliability, compatibility with existing systems, and social acceptance also determine a system's acceptibility. In a similar way, usability can be seen as one part of the wider fields of product design, industrial design, and user expericence design (the last term includes all aspects of owning and using a product).

Usability is a quantitative measure: Usability is more than a subjective, fuzzy concept capsulating the personal liking or disliking of a system or product. Instead, usability comprises several characteristics that can each be measured quantitatively:

- Learnability: How easy it is to get started. (E.g.: How long does it take an average user to learn how to use a set of actions relevant for a specific task)
- Efficiency: How much productivity is possible for an experienced user. (E.g. How long does it take to carry out a benchmark task). This notion of efficiency includes the term 'effectiveness' from the ISO 9241 definition: A system is *effective* if a given task can be solved using the system. A system is *efficient* if the same effect can be accomplished without too much time, cost, or effort.

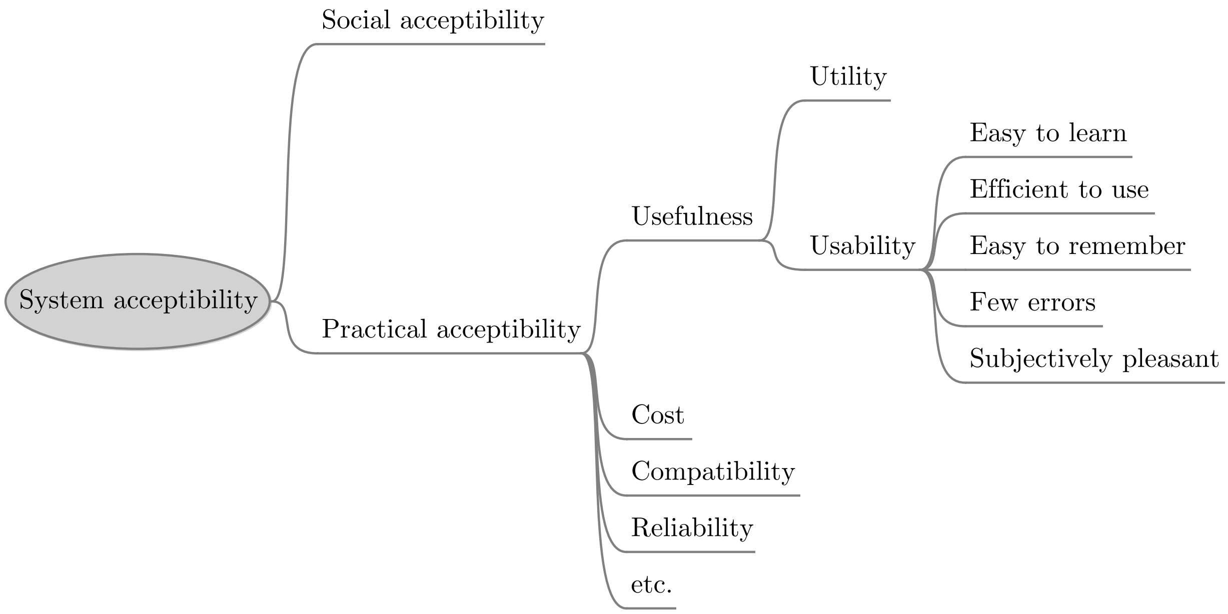


Figure 10: A model of the different attributes of system acceptibility, including the system's usability. Taken from [Nielsen, 1993, fig 1].

- Memorability: How easy it is to remember how the system works. (E.g. How well do users maintain their knowledge after a few hours, days, weeks...)
- Errors: How many (few) errors the user makes. (E.g. The numbers of errors made, classified by severeness)
- Satisfaction: How subjectively pleasant the system is to the user. (E.g. How positive is the feedback from interviews and questionnaires)

Usability depends upon the user's characteristics: A system that may be well-suited for one user (one group of users) may have very bad usability for another user (group of users). This is because users have different levels of experience with the system, with computers in general, and with the task domain. Other influencing factors include age, gender, memory, learning style and a user's personal attitude towards the system.

Usability needs Usability Engineering: The goal of creating a usable product can be reached using the systematic methods of usability engineering. These methods and practices can be combined into a development process. See the following section for more about usability engineering.

3.2 Accessibility

Accessibility is "the usability of a product, service, environment or facility by people with the widest range of capabilities" [International Organization for Standardization, 2003]. This definition relates accessibility with usability, but places an emphasis on people's different capabilities or disabilities. A definition and classification of disability has been defined in 1980 by the World Health Organization's[1] ICIDH classification: *Impairments* relate to body structures or functions, *disabilities* relate to activities a person may not be able to perform, and *handicaps* relate to participation in society. The more recent ICF classification shifts focus from cause to impact, places more emphasis on the social aspects of disability, and also includes contextual factors of a person's environment. The classification is available for free through the ICF browser application[2]. Clicking through the hierarchical structure offered by this application is a good way to get a more concrete understanding of the classification's terms.

Three aspects of an online wayfinding aid need to be accessible if the system is to be useful for people with disabilities:

- *Web Accessibility*: The internet site providing an online wayfinding needs to be technically accessible to people with different capabilities (e.g. sight, hand movement) and in different contexts (e.g. using special webbrowsers or text-to-speech engines). The W3C's Web Accessibility Initiative (WAI)[3] provides a wealth of information to this topic including the 'Web Content Accessibility Guidelines' (WCAG)[4] checklist.

- *Accessibility of the wayfinding description*: The route or location descriptions need to be useful for people with disabilities: 1.) Places, landmarks and routes must be described in an appropriate way for people with perceptional disabilities. E.g. purely visual landmarks make a route description useless for blind people. 2.) The routes must be chosen in an appropriate way for people with limited mobility. E.g. a route for people in wheelchairs must not make use of stairs.

- *Accessibility of the building itself*: In Austria, ÖNORM B 1600 defines what is needed to make a building accessible. An example of a campus map that includes accessibility information and a feedback mechanism for inaccessible places is the University of Montana's Access Map[5], see also figure 11, 'Accessibility: The University of Montana's access map', page 31.

[1] `http://www.who.int/`, May 2010
[2] `http://apps.who.int/classifications/icfbrowser/`, May 2010
[3] `http://www.w3.org/WAI/`, May 2010
[4] `http://www.w3.org/TR/WCAG10/`, May 2010
[5] `http://www.umt.edu/accessmap/`, May 2010

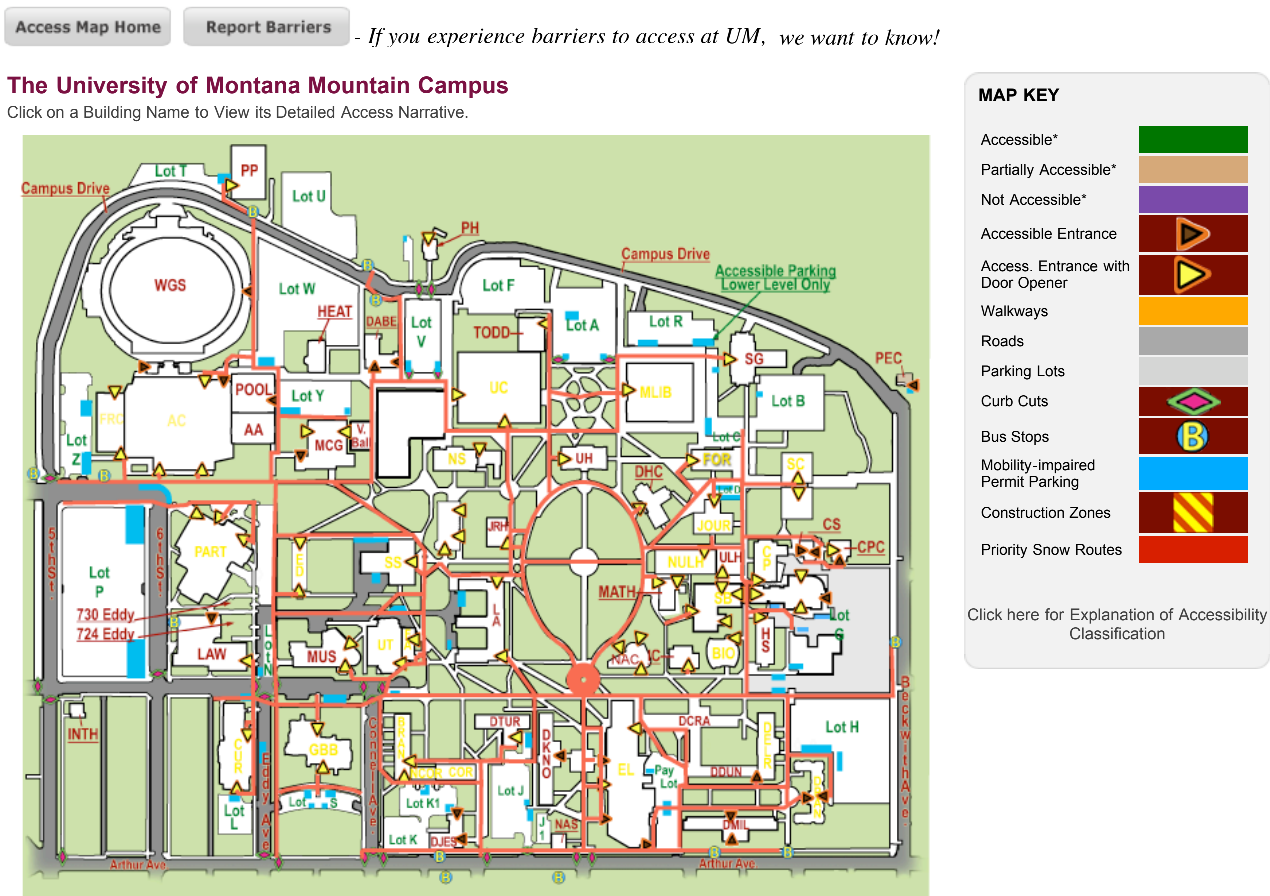


Figure 11: Accessibility: The University of Montana's access map provides accessibility information for the university's campus. The website also includes a feedback function for inaccessible places. Screenshot taken from `http://www.umt.edu/accessmap/`, May 2010.

3.3 Usability engineering

Usability engineering provides a systematic framework of methods to achieve good usability during product development [Nielsen, 1993]. It is a set of activities that take place throughout the whole lifecycle of a product: Significant activities happen at early stages before the user interface has even been designed (e.g. trying to understand the user's tasks), and other activities happen later in the product's lifecycle (e.g. evaluating the usability of a finished product). A very creative introduction to the topic of usability engineering is 'The Fable of the User-Centred-Designer' [Travis, 2009], where a young man discovers three principles of user centered design: Early and continual focus on users and their tasks, empirical measurement of user behaviour, and iterative design.

One goal and challenge of usability engineering is to *bridge the gap* between the computer's capabilities and the user's needs, or, put differently, to bridge the gap between the designer and the user (see figure 12, 'Usability engineering and bridging the gap', page 33 for an illustration). This mental gap between designer and user can be very challenging if the user's tasks and context differ a lot from the designer's world. In the case of the work at hand, the width of the gap between designer and user corresponds to level 2 in figure 12: The designer (and author of this work) designs a product that he understands (the wayfinding system), but this product will be used by many other people (e.g. students, teachers, people familiar or unfamiliar with the campus).

As described in [Mayhew, 1999, ch. 1], usability engineering draws from several other disciplines:

- *Cognitive psychology* is the study of human perception and cognition. In this work, cognitive psychology is of special importance to help understand how human wayfinding and orientation works.

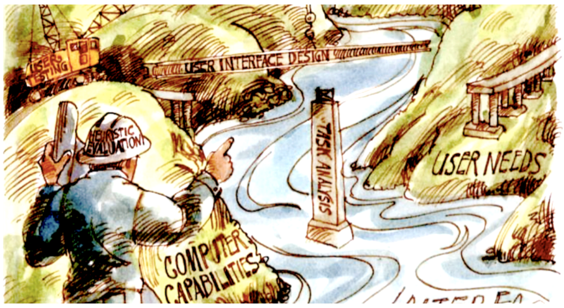
- *Ethnographic methods* can be used to better understand the user and their requirements. In this work, the ethnographical method of qualitative interviews has been used for task analysis (i.e. to analyze the user's approach to finding a room on campus).

- *Software engineering* is a structured approach to developing software. Usability engineering integrates into the software development process.

In correspondence to the many influencing disciplines, usability engineering consists of a wealth of different methods. An overview of methods can be seen in figure 13, 'Discount usability engineering', page 35 and in figure 16, 'Nielsen's usability engineering lifecycle', page 38. A classification of usability engineering methods is given in [Seffah and Metzker, 2009]. A more detailed and in-depth description of each of the methods can be found in textbooks such as [Nielsen, 1993; Shneiderman, 1997; Mayhew, 1999; Benyon et al., 2006; Herczeg, 2006].

Bridging the gap between

- ***computer capabilities and user needs***
- ***designer and user***



Level 1: ***The designer is the user.***

When you're designing something that only you will use.

(e.g. a geeky program used by yourself and a few other geeks)

Level 2: ***The Designer Understands the Product***

(e.g. when designing a mobile phone)

But be careful!! Design Team ≠ Typical Users

Designers...

know too much *about the product*

are too skilled *in using computers*

care too much *about their own baby*

Level 3: ***Designing for a Foreign Domain***

e.g. oil company geophysicists integrating drilling data and seismological data to determine the best place to drill for oil.

highly specialized ***users***

who perform narrow ***tasks*** *that*

depend on expert domain ***knowledge***

within a ***context*** *you can't even envision.*

image sources:
Book cover of J.Nielsen., Usability Engineering. Morgan Kaufmann, 1993
http://cybernetnews.com/wp-content/uploads/2008/06/linux-geek.jpg
http://www.newlinuxuser.com/wp-content/uploads/2008/10/tux.jpg
http://commons.wikimedia.org/wiki/File:Phone-nokia-6300.svg
http://upload.wikimedia.org/wikipedia/commons/f/fa/Oil_platform_in_the_North_Sea.jpg

Figure 12: Usability engineering and "Bridging the gap": The goal of usability engineering is to bridge the gap between the user's needs and the computer's capabilities, and to bridge the gap between the designer and the user [Nielsen, 1993, cover], [Nielsen, 2008a].

3.4 Benefits of usability engineering

The benefits of usability engineering can be described in different ways, depending upon how its effect is measured:

In a survey made by Nielsen [Nielsen, 2008b], the metric used to measure success are key performance indicators such as conversion rates, traffic numbers, user performance, or target feature usage. Using this metric, the study concludes that usability has a high return on investment (ROI) of 83% (as in Jan 2008). This figure is substantially smaller than it was in 2003. The study argues that this is due to the fact that "we've already cashed in the easy, huge advances, and so new projects will realize less spectacular gains".

In other case studies, the success of usability engineering is measured financially. In these studies, it is often difficult to prove that the success can be fully attributed to user interface improvements [Nielsen, 1993, p.2]. Despite this difficulty, many such case studies exist. The value propositions made by these studies include reducing development costs, increasing sales revenue, and improving the user's effectiveness - see [Marcus, 2002] for a list of case studies and statistics. A framework for cost-justifying usability engineering efforts is given in [Mayhew, 1999, ch. 20].

3.5 Discount usability engineering

Despite the promising ROI figures suggesting heavy use of usability engineering, developers and software managers can be intimidated by the cost associated with usability efforts. In this case, it can be argued that some usability work is better than none, and that using simplified usability methods that may not provide the best results is better than using no methods at all.

One such approach is called "discount usability engineering" [Nielsen, 1993, ch. 1.4], [Nielsen, 1994]. It suggests the use of four techniques, namely user and task observation, scenarios, simplified thinking aloud, and heuristic evaluation. An exemplary calculation for a medium-sized software project (see part *a* of figure 13, 'Discount usability engineering', page 35) shows that costs can be cut in half if cheaper methods are used. Another approach with a similar goal, but more focused upon qualitative research than on the usability engineering lifecycle as a whole, is 'Rapid Ethnography' [Millen, 2000]. A list of discount methods is provided on the usabilitynet website[1], a project sponsored by the European Union, in the form of a table of usability engineering methods. The table indicates for each method whether it is suited for situations with limited time, resources, skills, and limited access to users. (See figure 13, 'Discount usability engineering', page 35).

The idea of discount usability engineering is of great value to the work at hand, because limited time and resources - about one year and one person, parttime - make it necessary to gain good results quickly (instead of trying to gain the best results with an immense effort).

[1] `usabilitynet.org`, Mar 2010

a) An exemplary calculation of discount usability engineering

Original usability cost estimate	$128,330
Scenario developed as paper mockup instead of on videotape	- $2,160
Prototyping done with free hypertext package	- $16,000
All user testing done with 3 subjects instead of 5	- $11,520
Thinking aloud studies analyzed by taking notes instead of by video taping	- $5,520
Special video laboratory not needed	- $17,600
Only 2 focus groups instead of 3 for market research	- $2,000
Only 1 focus group instead of 3 for accept analysis	- $4,000
Questionnaires only used in feedback phase, not after prototype testing	- $7,200
Usability expert brought in for heuristic evaluation	+ $3,000
Cost for "discount usability engineering" project	**$65,330**

b) Usability methods suitable for use with limited ressources

Planning & Feasibility	**Requirements**	**Design**	**Implementation**	**Test & Measure**	**Post Release**
Getting Started LTS, NU	User surveys NU	Design Guidelines NU	Style guides LTS, NU	Diagnostic evaluation LTS	Post release testing
Stakeholder meeting LTS, NU	Interviews	Paper prototyping LTS, NU	Rapid prototyping LT	Performance testing	Subjective assessment LTS
Analyse context LTS, NU	Contextual inquiry	Heuristic evaluation NU		Subjective evaluation LTS	User sureys
ISO 13407	User Observation	Parallel design NU		Heuristic evaluation NU	Remote evaluation
Planning NU	Context LTS, NU	Storyboarding NU		Critical incident technique	
Competitor analysis NU	Focus Groups	Evaluate prototype		Pleasure	
	Brainstorming NU	Wizard of Oz NU			
	Evaluating existing systems LTS	Interface design patterns NU			
	Card Sorting				
	Affinity diagramming LTS, NU				
	Scenarios of use LTS, NU				
	Task analysis NU				
	Requirements meeting LTS, NU				

NU: No users – this method also works if direct access to users is not available.
LT: Limited time – this method can be employed if only limited time is available.
LTS: Limited time and skills – this method can be employed if only limited time and skills are available.

Figure 13: Discount usability engineering methods. (a) An exemplary calculation on how costs can be saved using 'discount' usability engineering methods in a medium-scale software project. Taken from [Nielsen, 1994, tbl. 1]. (b) An overview of methods of usability engineering. The methods are annotated according to how well they lend themselves for situations without direct contact to users and where limited time and resources are available. Taken from `usabilitynet.org`, Mar 2010.

3.6 The usability engineering lifecycle

When usability engineering methods are applied in the field of software engineering, the methods and processes of the two fields need to be integrated. Historically, there have been three approaches on how this integration can be accomplished [Mayhew, 1999, ch. 1]: One approach is to simply add usability experts to an existing team. Alternatively, selected usability methods can be added to an existing software development process. A third approach is to fully re-design the whole development process. This is the approach taken in [Mayhew, 1999] and followed in this work:

Mayhew's 'usability engineering lifecycle' is a development process that integrates usability engineering and object-oriented software engineering tasks. The process consists of three main steps or phases. (See figure 14, 'Mayhew's usability engineering lifecycle', page 37)

- *Analysis phase*: Understand the user and his tasks.
- *Design phase*: Iteratively create a concept or guideline, design and implement prototypes and later on the real system, and evaluate what has been accomplished.
- *Installation phase*: The product installed is the result of the last iteration.

It is an iterative process, which means that during the design phase, the sequence 'conceptualize-design-prototype-evaluate' is looped several times, each time gaining one level of detail. This is opposed to linear development processes such as the waterfall model, where a sequence like 'analyse-design-implement' is executed once only. See figure 15, 'Iterative development processes', page 38 for a graphic illustrating the iterative nature of the usability engineering lifecycle.

The methods of usability engineering: An overview of usability engineering methods for each step in the development process can be seen in figure 13, 'Discount usability engineering', page 35 and figure 14, 'Mayhew's usability engineering lifecycle', page 37. These methods are discussed in a detailed way, including both practical aspects and the scientific background, in textbooks such as[Benyon et al., 2006; Shneiderman, 1997; Mayhew, 1999; Nielsen, 1993], so that within this work and in the following sections, only a brief description of methods and aspects relevant to the work at hand shall be given.

Application in this work: In this work, a modified version of Mayhew's usability engineering lifecycle is used. Because of the project's limited scope and resources, the process has been simplified and only discount methods (see previous section) have been used. Another difference is that the last phase is not the installation of a finished product, but the description and demonstration of a prototype. The resulting process is shown in figure 21, 'This work's iterative development process', page 52 and explained in the containing chapter 4, 'Analysis phase', page 51.

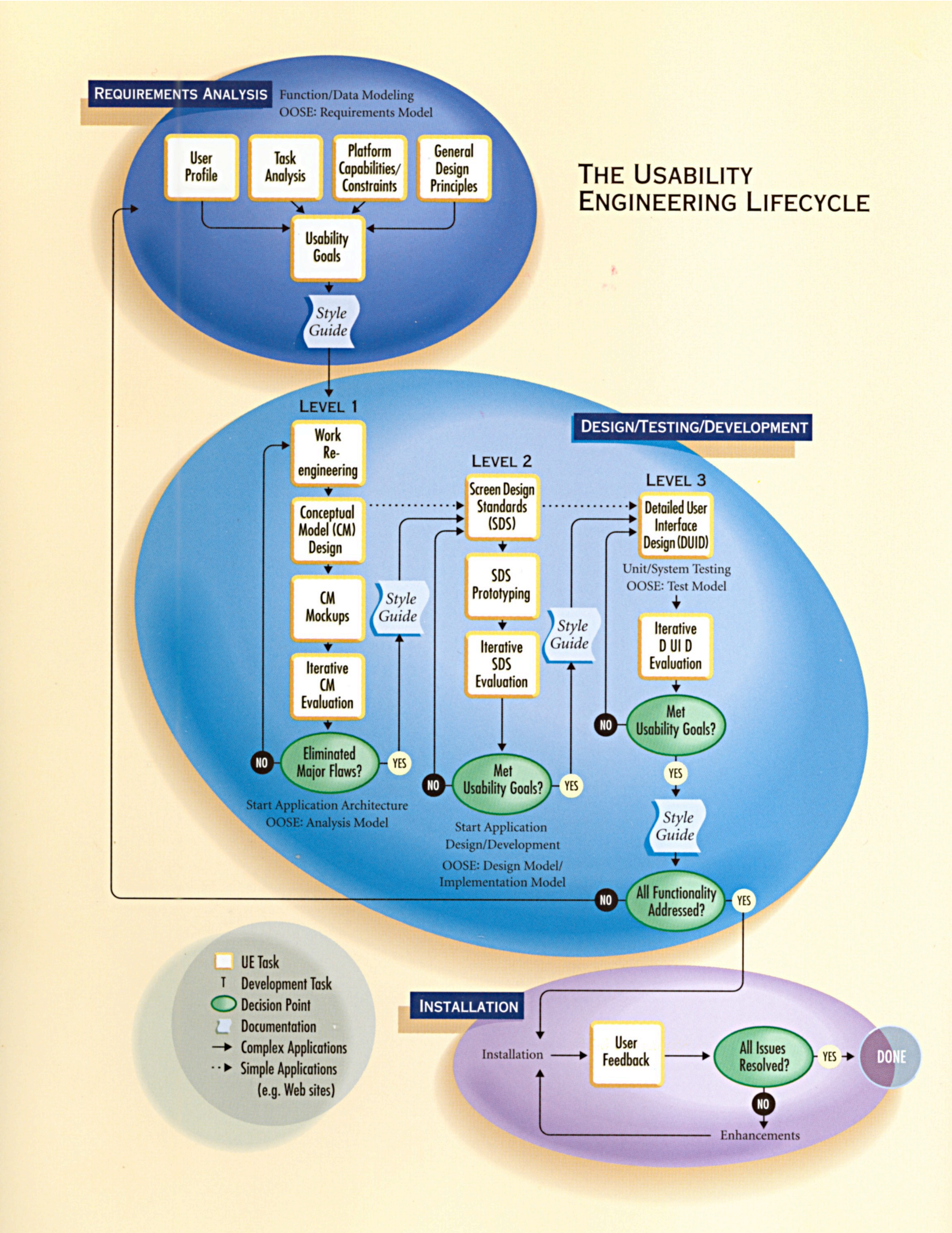


Figure 14: A usability engineering process as described in Deborah Mayhew's book "The usability engineering lifecycle" [Mayhew, 1999].

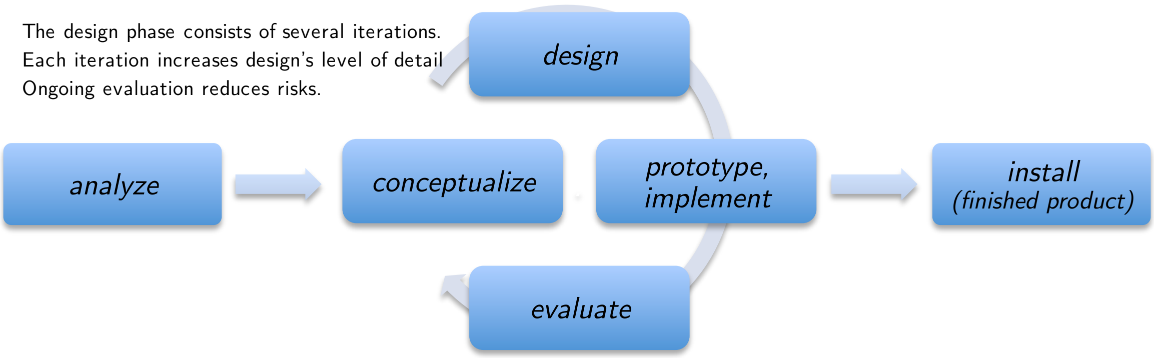


Figure 15: Iterative development processes: Where iteration takes place in an iterative development process such as Mayhew's usability engineering lifecycle or the development process used in this work.

1. **Know the user**
   a. Individual user characteristics
   b. The user's current and desired tasks
   c. Functional analysis
   d. The evolution of the user and the job
2. **Competitive analysis**
3. **Setting usability goals**
   a. Financial impact analysis
4. **Parallel design**
5. **Participatory design**
6. **Coordinated design of the total interface**
7. **Apply guidelines and heuristic analysis**
8. **Prototyping**
9. **Empirical testing**
10. **Iterative design**
    a. Capture design rationale
11. **Collect feedback from field use**

Figure 16: A usability engineering process as described in Jakob Nielsen's book "Usability Engineering" [Nielsen, 1993, table 7].

3.7 User involvement

Involving 'the user' (the people who are later on going to use the system that is now in construction) into the development of a software product is a widely accepted principle in usability engineering. There is however no clear definition of user involvement, but the term is associated with a 'focus on users', 'consulting end-users', and 'participation of users' [Kujala, 2003].

Different approaches to user involvement: An overview of user involvement in software engineering is given in [Kujala, 2003]: The paper names benefits and challenges for four different approaches to user involvement: User centered design, Participatory design, Ethnography, and Contextual design (see figure 17, 'Approaches to user involvement', page 40 for an overview). The way in which these approaches are practically applied can be very different from one organization to the other, depending on the organization's culture, preferences and strategies [Iivari, 2006].

Different levels of user involvement: The actual level of involvement can be very different. It may range from 'tester' (the user is only involved for testing), to 'informant' (the user is allowed to comment to designers), up to full participation (the user is an active member of the design team). [Kujala, 2003; Druin, 2002].

Benefits of user involvement: In a more general way, the benefits of usability engineering have already been described in the previous section 3.4, 'Benefits of usability engineering', page 34. The following list names benefits of user involvement (thus benefits of a specific subset of usability engineering activities) [Kujala, 2003]:

- *More accurate requirements.* Avoiding costly features that the user does not want or cannot use leads to an improved quality of the system and to higher acceptance.

- *A better understanding of the user's actual tasks.* (Especially through the contextual design approach).

- *Discovering tacit knowledge* from well-learned tasks and everyday self-evidences that are difficult to articulate. (Especially through field studies).

- *Increased participation in decision-making* within the organization. (Through participatory design)

When to involve users during software development: In [Kujala, 2003, p.1] it is argued that "user involvement is most efficient and influential in the early stages of system development as the cost involved in making changes increases during system development". Some particular methods however are specifically targeted at later stages of the development process, such as the evaluation of a software product through end user testing. The author concludes that different methods are best used at different steps of the development process. This is in line with comprehensive textbooks on usabil-

	User-centred design	Participatory design	Ethnography	Contextual design
Emphasis	Usability	Democratic participation	Social aspects of work	Context of work
Typical methods	Task analysis, Prototyping, Usability Evaluations	Workshops, Prototyping	Observation, field studies, video-analysis	Contextual inquiry, Prototyping

Figure 17: Different approaches to user involvement and their respective methods. Taken from [Kujala, 2003, table 1].

ity engineering that describe methods of user involvement for each step in the development process [Benyon et al., 2006; Mayhew, 1999].

Application in this work: In this work, users have been involved in interviews early on in the development process to learn about their perceived tasks and needs, and in test sessions during design and development. (See the practical part of this work: analysis phase in chapter 4, design phase in chapter 5, and presentation phase in chapter 6).

3.8 Qualitative research

Qualitative research is - as the name implies - research that is based on qualitative data. Qualitative data are mostly verbalizations or other non-numeric symbolizations, for example: Interview texts, observation protocols, photos, etc. To analyze this kind of data, qualitative research has developed special interpretative methods. An overview of qualitative research can for example be found in [Bortz and Döring, 2006, ch 5]. Qualitative research is opposed to quantitative methods, where numerical representations of data are used.

The usefulness of qualitative research: Qualitative research is useful for HCI design (human-computer-interaction design) and system engineering during all phases of development, from the gathering of requirements to product evaluation and iterative design [Millen, 2000]. Qualitative methods enable a researcher to discover the user's tacit knowledge about automated activities: "In well-learned tasks, much of the relevant knowledge is no longer consciously available for the person and everyday self-evidences are difficult to articulate. Thus, the type and level of user involvement need to be carefully considered. A promising approach is to perform field studies, whereby qualitative methods are used to study users and their activities in their own environment. Users do not need to explicitly articulate their needs, but the underlying problems and possibilities are understood by studying the future context of use." [Kujala, 2003, p.1]

The diversity of methods of qualitative research: Despite the fact that qualitative research has been used for software engineering purposes for more than twenty years, there is no 'one way' of doing it.

Instead, different theoretical foundations and different purposes are found, as described in [Dittrich et al., 2007]:

- Qualitative research with a *positivistic underpinning* seeks to find objective, truthful statements about the world. In this case, qualitative methods are employed with a similar purpose as in quantitative research, e.g. for subsequent statistical analysis or to test a research hypothesis.

- *Ethnographically inspired research* seeks to understand the culture, language and thinking of a group of people. This approach is widely recognized [Hughes et al., 1994] for the design of CSCW (Computer supported cooperative work) systems.

- Some *pragmatic 'quick and dirty' approaches* to qualitative research have been criticized in that they lack a firm theoretical foundation [Dittrich et al., 2007] and underestimate the theoretical grounding and analytic work needed to produce good ethnographic results [Eriksson et al., 2009, ch.4]. Nevertheless, short product cycles and limited resources lead to the necessity of producing results quickly, and HCI methods are still useful even if they are applied by people with no HCI experience [Eriksson et al., 2009]. One approach titled 'Rapid Ethnography' [Millen, 2000] trying to gain ethnographic results quicker is based on the three ideas of (a) narrowing the field research before entering the field, (b) using multiple interactive observation methods to discover exceptional and useful user behavior, and (c) the use of computerized iterative data analysis methods.

The problem with a-priori assumptions in qualitative research: One practical problem is the perspective that the researcher should start observations with the least possible a-priori assumptions. This is formulated in [Froschauer and Lueger, 1992, p.36] as follows: "Ethnography starts with a conscious attitude of almost complete ignorance. 'I don't know how the people of Cushing, Wisconsin, understand their world. That remains to be discovered.'". The practical application of qualitative research in the field of usability engineering shows however that starting without any a-priori assumptions at all is impossible. E.g. before an interview can be designed with the aim of characterizing personas, the researcher must have determined user categories [Mayhew, 1999, p.39]. By doing so, a-priori assumptions about why some people shall be interviewed and others not have already been met. Since it is impossible to start observations without prior assumptions, the approach taken in this work is to try to be conscious about the assumptions made.

Application in this work: Qualitative interviews have been carried through during the analysis phase of this work (see section 4.2, 'Interviews', page 60). Another area where qualitative methods have been used is for in-depth evaluation of a relatively small number of wayfinding systems (see section 4.1, 'A survey of existing solutions', page 53).

3.9 Qualitative interviews

Qualitative interviews belong to the methods of qualitative research. Their output is verbal or textual. This is in contrast to questionnaire-style interviews where the interviewee's answers directly translate to a numeric value (e.g.: 1 - I agree. 2 - I am not sure. And 3 - I disagree) that can be evaluated in a quantitative way. The perspective taken in qualitative interviews is also very different: "The primary means for data collection is to talk with people and to get them to tell their stories, as opposed to answering your questions." [Beebe, 2001, p.3].

A typical qualitative interview is characterized in [Hopf, 1978, p.99] as follows:

- It is conducted by the researchers themselves
- It does not have a lot of (pre-defined) structure
- It is led by loose hypotheses
- It serves for the exploration of an area of research that has not yet been extensively covered by science

Explorative versus focused interviews: Qualitative interviews differ in how much the focus and structure of the interview has been predefined. Explorative interviews allow more freedom, while focused interviews tend to stick to a predefined structure. The difference between the two forms of interview and the trade-off between their particular advantages and disadvantages is described at length in [Hopf, 1978, p.101-107]: Sticking too close to pre-defined points hinders spontaneity, blocks answers and cuts off topics, and thus limits the width of the interview. On the other hand, sticking to the points will make the conversation stay close to the interviewer's focus of interest.

Preparing questions for an interview: For a focused interview, questions need to be designed beforehand. To help with this task, a systematic framework of different types of questions can be useful. Such a framework is given in [Froschauer and Lueger, 1992, p.46-48], and the following types of questions are identified:

- *Open/closed*: Open questions leave a wide array of possible answers, not just yes or no. For a qualitative interview, open questions are preferred over closed questions.
- *Immanent/exament*: Immanent questions lead the interviewee to telling more about the same or a similar subject. Exmanent questions lead to a new topic. In qualitative interviews, immanent questions are used first to get the interviewee to tell as much as possible about things that matter to him. In a later phase, exmanent questions may be asked in order to lead the talk to topics that have so far been left out.

- *The type of question*: direct, indirect, relational, recursive. Direct questions directly ask for facts, e.g. "Can you describe the cooperation with your colleagues?". Indirect questions only relate indirectly to the facts, e.g. "If your relationship with your colleagues was an image, what would it look like?". Relational questions ask about a relation between different facts, e.g. "How would you compare the cooperation in your department to the cooperation in the company as a whole?". Recursive questions bring back the same topic but vary the question, e.g. "How do you think that others think about this?"

- *The question's content*: social, factual, temporal. A question may be about social relationships, about factual matters, or about temporal development.

A practical guideline for the interviewer's attitude is given in [Froschauer and Lueger, 1992]:

- *Learning*: The interviewing person must see themselves as learning from their interview partner.

- *Interest and curiosity*: Expect new and interesting things to be learned, try to examine them from the perspective of the interview partner.

- *Avoid quick judging*: Stay open instead, the thinking of the person interviewed is of primary interest.

- *The person interviewed is always right*: This emphasizes the subjectivity of the material. Even contradictions can make sense from the perspective of the person interviewed.

- *Listen*: Allow a broad range of answers, listen well, ask when things are unclear

- *Don't take things for granted*: Many terms are used with a specific meaning within a particular social context.

Alternatives to qualitative interviews include other qualitative methods such as field research and other interactive or non-interactive observational methods. The author of [Filstead, 1971] argues that participant observation ("that method in which the observer participates in the daily life of the people under study") delivers the most complete information, because this method is able to uncover aspects that remain hidden in interviews (e.g. special meanings given to specific words, matters the interviewee does not want to talk about, or things people see through distorting lenses).

Application of qualitative interviews in this work: Qualitative interviews have been employed in the analysis phase of this work, see chapter 4 on page 51.

3.10 Personas and storyboards

Personas are fictional characters that capture the characteristics of a group of users [Cooper, 2004]. Storyboards are fictional stories presented like a comic book that capture how the envisioned system could be used, i.e. who uses it in which context [Van der Lelie, 2006]. Both methods have in common that they bring more life into the otherwise rather dry descriptions of users and tasks. The benefit of using these methods is the ability to communicate user and task characteristics within different parts of the team and to focus the design upon the user and work context [Pruitt and Grudin, 2003; Haesen et al., 2009; Van der Lelie, 2006].

Personas are described using text and image in a way that creates a lively image as if it were of a real person. Although the description is fictional, it is based upon knowledge about the users of a system. The benefit of personas is that they replace the very general and fuzzy term ‘user’ with the goals, capabilities, likes and dislikes of a particular person: Instead of stretching the term ‘user’ into whatever direction seems convenient to the designer, the designer will now have to stretch and design for the target persona [Cooper, 2004]. A practical example of what may be included into a persona definition is shown in figure 18, ‘Exemplary persona definition’, page 45. This example of a very comprehensive persona description is taken from [Pruitt and Grudin, 2003], a paper that describes how personas have been used by Microsoft in one large project (several hundred people working on MSN Explorer) and one very large project (several thousand people working on Microsoft Windows). Difficulties with using personas and practical advice to overcoming these difficulties is also given.

Personas versus user profiles: Both methods are used to describe the characteristics of the users of a system. The difference between personas and user profiles is how these characteristics are communicated: In the case of user profiles, it is through a plain list of statistically proven facts. In the case of personas, it is through fictional characters based upon the same data. Both personas and user profiles are created during the analysis phase of a user-centered software development process, and the results are used to guide the design process. The advantage of personas in this context is that they enable designers to imagine the person described, and to extrapolate other behavior from the partial knowledge they have: “If team members are told, ‘Market research shows that 20% of our target users have bought cell phones’, it may not help them much. If told ‘Alan has bought a cell phone’ and Alan is a familiar Persona, they can immediately begin extrapolating how this could affect behavior. They can create scenarios. We do this kind of extrapolation all the time, we are skilled at it – not perfect, but very skilled” [Pruitt and Grudin, 2003].

Personas need ethnographic research: In order for personas to be believable characters, they need to be based upon real data from ethnographic research, both qualitative and quantitative [Pruitt and Grudin, 2003]. Purely fictional personas that are not based upon real data have been reported to be abused by design firms to justify whatever decision made [Loo, 2010].

Just a few personas are enough: Creating only a few personas should be sufficient for most purposes: When personas were used to design Microsoft Windows, the work of 22 people during three months

Exemplary table of contents of a persona definition:

Overview – Alan Waters (Business Owner)

Get to know Alan, his business, and family.

A Day in the Life

Follow Alan through a typical day.

Work Activities

Look at Alan's job description and role at work.

Household and Leisure Activities

Get information about what Alan does when he's not at work.

Goals, Fears, and Aspirations

Understand the concerns Alan has about his life, career, and business.

Computer Skills, Knowledge, and Abilities

Learn about Alan's computer experience.

Market Size and Influence

Understand the impact people like Alan have on our business.

Demographic Attributes

Read key demographic information about Alan and his family.

Technology Attributes

Get a sense of what Alan does with technology.

Technology Attitudes

Review Alan's perspective on technology, past and future.

Communicating

Learn how Alan keeps in touch with people.

International Considerations

Find out what Alan is like outside the U.S.

Quotes

Hear what Alan has to say.

References

See source materials for this document.

Figure 18: Exemplary table of contents of a persona definition, illustrating the kind of information that may be used to describe a persona. Taken from [Pruitt and Grudin, 2003, fig.1].

boiled down to a total of six personas [Pruitt and Grudin, 2003]. Creating too many personas will make it difficult to focus design upon the personas' characteristics. To help designers focus even more clearly, anti-personas can convey the kind of person that the system shall be specifically not designed for.

Storyboards are small episodes of how the envisioned system could be used. This is communicated using a combination of text and images resulting in a comic-style sequence of narrative elements. Exemplary storyboards can be found in [Haesen et al., 2009] and in the practical part of this work. Photos can also be used instead of drawings, as demonstrated in [Pedell and Vetere, 2005].

Storyboards versus scenarios and use cases: The common purpose of storyboards, scenarios and use cases is to capture the user's tasks or activities. In contrast to the often purely technical use cases, scenarios and storyboards also consider the context of use, i.e. social interactions, physical environments and personal motivations [Pedell and Vetere, 2005]. Scenarios communicate tasks and context by telling little episodes using text only, while storyboards combine text and images. All three methods can be used at different phases in a user-centered software engineering process: If used to describe the current situation, the methods belong to the analysis phase. If used to describe a future situation (e.g. how the user will accomplish tasks using the envisioned system), the methods belong to the beginning of the design phase, where work re-engineering takes place. (See section 3.6, 'The usability engineering lifecycle', page 36).

Storyboards and personas combined: Storyboards sometimes lack actors with a real personality (the same is true for scenarios and use cases). Personas can be used to bring more realistic characters into these otherwise rather abstract and dry stories [Pruitt and Grudin, 2003].

Software support for creating storyboards: Storyboards can be drawn by hand or using standard drawing software, but special software also exists. There are a lot of products made for the film industry (where storyboarding is also used), but some products are targeted at software design. In [Haesen et al., 2009], the authors describe a custom-made tool that supports combining a scenario (purely textual) with a storyboard (image sequence with text annotations) and metadata (other textual annotations).

Application in this work: In this work, two personas were created (see section 4.3, 'Personas', page 65). To formulate the persona descriptions, insight gained from interviews was used. An emphasis was laid upon those user characteristics that were likely to influence future use. The personas were used later on to guide the design process. Because of time constraints, not every need formulated in the persona definitions could be accounted for when building the prototype, e.g. accessibility and internationalization had to be left for future work. Storyboards have been used at the beginning of the design phase with the intention of trying to imagine the future use of the university's wayfinding system. In a similar way to anti-personas, anti-storyboards have been created to illustrate competing ways of getting wayfinding information. (See section 5.1, 'Storyboards', page 72.)

3.11 Wireframes, mockups and prototypes

The terms wireframes, mockups and prototypes are not very clear and are sometimes used interchangeably. The common motivation is that it is important to explore many ideas and possible solutions during the design process. In order for these ideas to be communicated and evaluated, they need to be embodied into some kind of prototype, ranging from simple hand-drawings to physical objects or interactive computer applications.

Low-fidelity prototyping and wireframe sketches: Wireframes are low-fidelity mockups of the envisioned system. The user interface is represented through simple black-and-white drawings including text and the outlines of buttons and windows etc (thus the name 'wireframe'). Wireframes can be drawn by hand or using software such as drawing programs, presentation software, or even spreadsheets for form-based applications. Special software also exists for this purpose. The activity of creating low-fidelity mockups takes place at the beginning of the design phase (see section 3.6, 'The usability engineering lifecycle', page 36).

High-fidelity prototyping: When the general concept has been proven using low-fidelity prototypes, the design needs to be further elaborated: User interaction and detailed screen designs can be tested using high-fidelity prototypes. These prototypes can be graphically very accurate and may feature complex interaction where necessary, but they will typically lack application logic. E.g. forms submitted will not be processed, or the data returned will come from a simple textfile instead of a full-blown database system. The activity of creating high-fidelity mockups takes place during later parts of the design phase. (See section 3.6, 'The usability engineering lifecycle', page 36).

Early prototyping: Late design changes are costly. Prototypes allow to test ideas early on in the development process. At that stage, all it takes to change the design is to quickly make another protoype. This reduces the risk of developing a product that in the end is not usable.

Quick and cheap prototyping: In order for prototyping to be effective, it needs to be quick and cheap. This allows for many ideas to be tested, changed and improved, or rejected. In many cases, very simple, cheap and quick paper prototypes will be sufficient [Mayhew, 1999, ch.9], especially in earlier phases of product development (see figure 19, 'Cheap prototyping at the beginning of the design process' and [Buxton, 2007, fig. 51]).

Application in this work: Prototyping has been extensively used in this work - in fact, the main deliverable of this work is a prototype demonstrating a possible wayfinding interface. The creation of this protype is described in the practical part of this work (chapters 4 to 6).

3.12 Usability evaluation

Usability evaluation comprises a set of methods to measure the usability of a product or prototype. It has a different focus than other test methods that are aimed at finding technical errors ('bugs'):

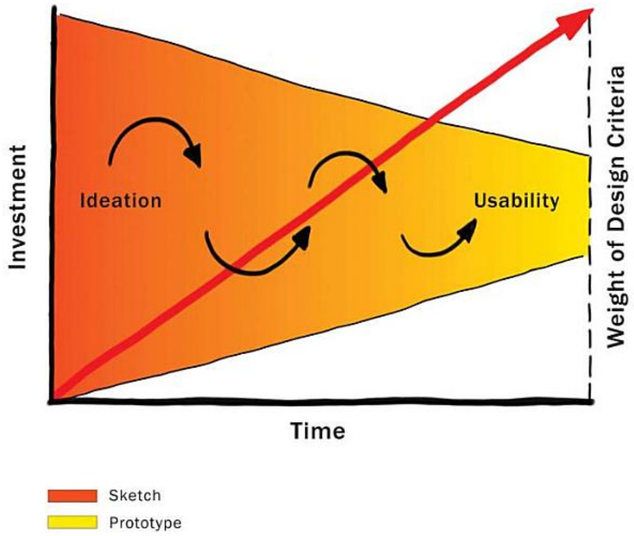


Figure 19: Cheap prototyping at the beginning of the design process: The design process begins with ideation and sketching of very simple (and inexpensive) prototypes. The process then narrows down into more and more specific (and expensive) prototypes. Graphic taken from [Buxton, 2007, fig. 51].

While bugs are likely to reduce the usability of a product, the definition of usability goes much further to include learnability, efficiency, memorability, errors made by the user, and user satisfaction (see section 3.1, page 28 for a definition of usability). All of these factors contributing to usability are covered by usability evaluation.

Motivations for usability evaluation: "Designers can become so entranced with their creations that they may fail to evaluate them adequately. Experienced designers have attained the wisdom and humility to know that extensive testing is a necessity" [Shneiderman, 1997]. This necessity to evaluate the usability of a product or design has to do with differences between designers and users (see figure 12, 'Usability engineering and bridging the gap', page 33 for an illustration) and between different groups of users: The same design won't work well for everybody, and what works for the designer will not necessarily work for the users. Designing a product that does fit the target users (and having to re-design it later on) can be very costly. To reduce this risk, designs need to be prototyped and evaluated.

Usability evaluation within the development process: Usability evaluation takes place at all stages of the software engineering process. It ranges from the evaluation of early design ideas to the evaluation of a finished product. E.g. Mayhew's usability engineering lifecycle places evaluation steps at the end of every iteration (see section 3.6, 'The usability engineering lifecycle', page 36). A more detailed classification of usability tests and when they are best used during the development process is found in [Rubin, 1994]:

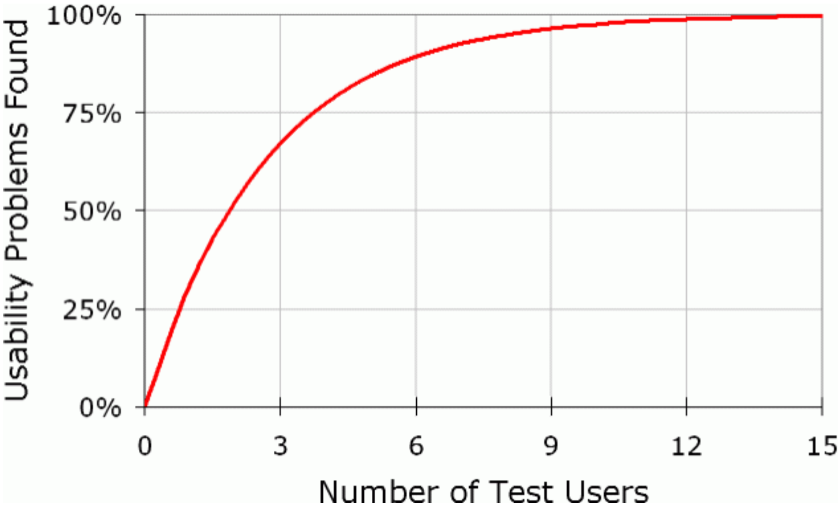


Figure 20: How many test users in a usability test? [Nielsen, 2000] recommends testing with a maximum of five users because: "As you add more and more users, you learn less and less because you will keep seeing the same things again and again". Figure taken from [Nielsen, 2000].

- *Exploratory testing*: This type of testing seeks qualitative insight and can typically inform the designer for basic design decisions early on during the product development process.
- *Assessment testing*: In this type of test, the user is asked to complete a specific task (an assessment) using the software that is being developed. Assessment testing is typically used later on during the development process, when the design has been specified using detailed prototypes.
- *Comparison testing*: Different design options can be compared by testing both variants with users. This type of testing can be applied at all stages of the development process.
- *Validation testing*: Usability measures can be included into requirements specifications and legal contracts. Validation testing is used to find out if a software product meets these requirements. This typically happens very late during the development process.

Methods of usability evaluation: There are many different methods available for usability evaluation. Some of them don't involve end users (e.g. heuristic evaluation or cognitive walkthroughs where a usability expert uses his experience together with guidelines and usability patterns). Other methods do involve end users (e.g. usability tests, interviews, surveys, beta testing). Amongst these latter methods, only usability tests directly observe the end user during his work with the product. A motivation for observing the end user with the product is that it may not suffice to ask for people's opinion – they don't always know what is achievable. (A related saying attributed to Henry Ford goes: "If I had asked my customers what they wanted, they would have asked for a faster horse" [Travis, 2009]).

Methods of observation: There exist different methods of observing the user's behavior in a usability test: E.g. videotaping, screen recording, eye tracking, and the thinking-aloud-method, which is recommended as a low-cost approach [Nielsen, 1993] and has been used in this work. In thinking-aloud tests, the participants are asked to speak out loud what they are thinking while executing the tasks given to them, e.g. motives, expectations, surprises, contentment or discontentment. A more in-depth description of the different methods of usability evaluation and a practical guideline is given in textbooks such as [Rubin, 1994].

Low-cost usability evaluation: In situations with limited time and budget, it is better to do some usability testing than to do none. This is in line with Nielsen's 'discount usability engineering' approach (see section 3.5, page 34) where the author recommends heuristic evaluation and the 'speaking aloud' method for usability tests. Also, just a few test persons will suffice in many cases: Nielsen recommends to test with a maximum of five users (see figure 20, 'How many test users in a usability test?', page 49), except for situations with several highly distinct groups of users. It is reported that three test persons find about 50% of usability errors, four to five persons find about 80%, and ten persons find about 90% of errors. [Dumas and Redish, 1999].

Application in this work: This work used interviews to evaluate early ideas and thinking-aloud tests to evaluate wireframes and prototypes. (See section 5.9, 'Usability evaluation', page 98).

4 Analysis phase

The previous chapters about wayfinding (chapter 2) and usability (chapter 3) lay a theoretical foundation that the following practical part builds upon. The latter is structured in three phases: An analysis phase (this chapter), a design phase (chapter 5) where a prototype is iteratively created, and a final phase (chapter 6) where the resulting prototype is presented. This structure corresponds with a modified version of Mayhew's usability engineering lifecycle: An iterative, user-centered development process using discount methods only, as depicted in figure 21, page 52.

The *methods* employed in the practical part of this work have also been described theoretically. The corresponding parts have been cross-referenced, but the following shall provide an additional overview:

This work's development process

- A simplified version of the process in section 3.6, 'The usability engineering lifecycle', page 36.

The analysis phase: To understand the user and his tasks.

- A survey of existing solutions: See section 3.8, 'Qualitative research', page 40 for the corresponding theory.
- Qualitative interviews: See section 3.8, 'Qualitative research', page 40 and section 3.9, 'Qualitative interviews', page 42.
- Personas and Use Cases: See section 4.3, 'Personas', page 65 and section 4.3, 'Personas', page 65.

The design phase: To iteratively create a concept or guideline, design and implement a prototype, and evaluate what has been accomplished.

- Storyboards: See section 3.10, 'Personas and storyboards', page 44 for the theoretical background.
- Wireframes: See section 3.11, 'Wireframes, mockups and prototypes', page 47.
- Combining media and ways of accessing the system: See section 2.5, 'The medium used for wayfinding aids', page 17.
- Map design: See chapter 2, 'Wayfinding', page 9, but in particular the sections about map orientation (section 2.3, page 15), indoor wayfinding (section 2.4, page 16), map abstraction (section 2.6, page 19), and landmarks (section 2.8, page 25).
- Evaluation: See section 5.9, 'Usability evaluation', page 98.

The presentation phase: To deliver the results.

- The presentation phase simply describes the prototype that results from the last iteration of the design phase.

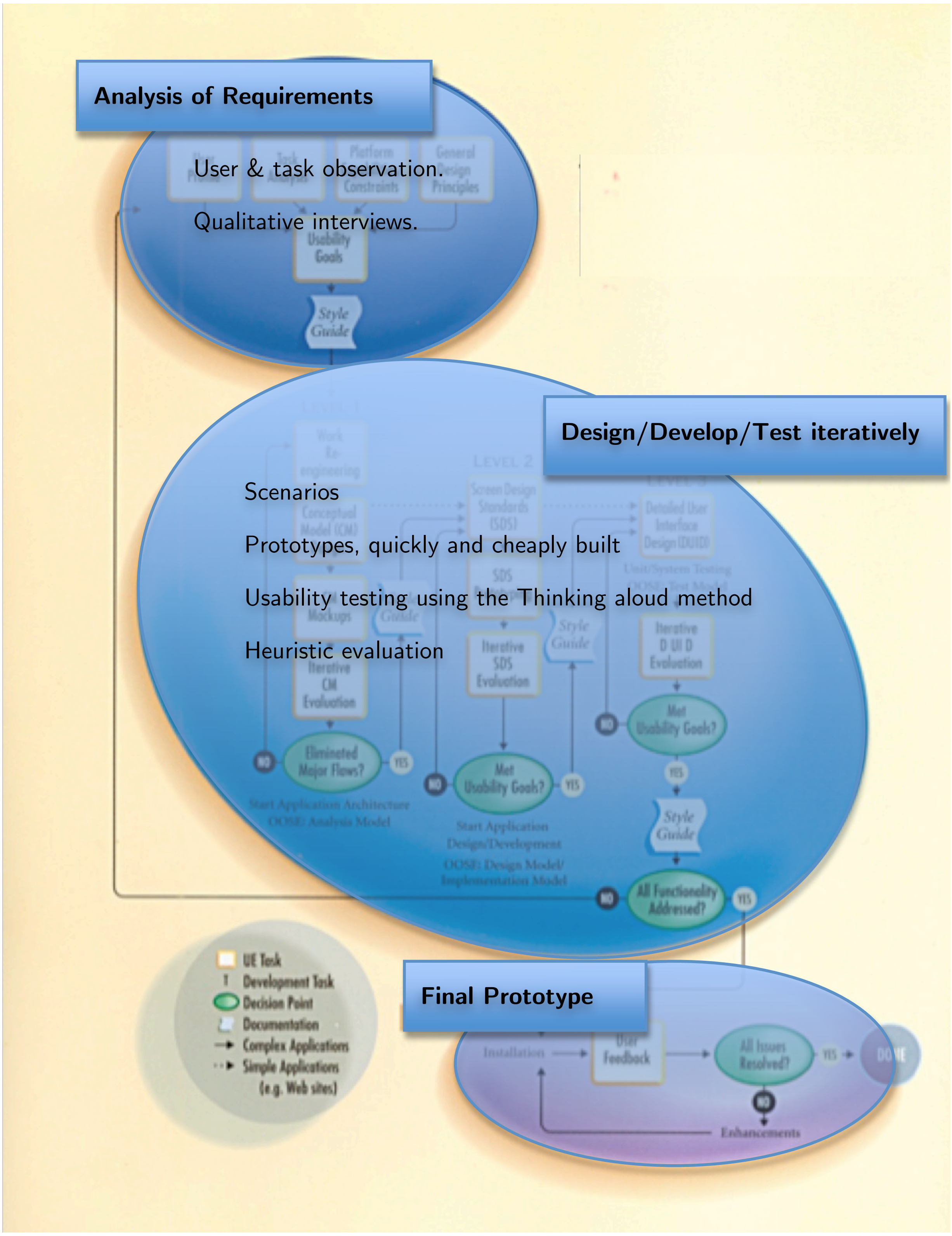


Figure 21: This work's iterative development process is an adapted version of Mayhew's usability engineering lifecycle, using discount methods only. In line with the scope of this work, the output of the process is just a prototype and not an installed product.

The goal of the analysis phase in the usability engineering lifecycle is to better understand the users and their context:

- *The users*: Who are the users? What characteristics are relevant to how they will be using the system? – In this work, interviews were performed to learn more about the users, see section 4.2, 'Interviews', page 60. The results are presented in section 4.3, 'Personas', page 65.
- *The tasks*: How do the users think about the tasks to be done, and what words do they use to describe them? How do they currently organize their work? – Interviews were performed to learn more about the tasks from the users's perspective. The results are presented in section 4.4, 'Use cases', page 67.
- *The platform*: This phase also considers platform constraints and general design principles. – In this case, we considered the data needed for the wayfinding application (see section 4.5 on page 68) and the integration with other systems (see section 4.6 on page 70).

Before any of the above analysis tasks were performed, existing wayfinding aids were compared in order to learn about these solutions and to understand design possibilities for a future system:

4.1 A survey of existing solutions

This section presents a survey of existing wayfinding aids. The intent of this competitive analysis is to evaluate existing solutions before creating a similar system for the Vienna University of Technology.

The perspective taken in the survey is that of a user comparing services that are offered to him; not that of a company seeking to buy a software solution: The survey compares the user interfaces that are provided by universities or other companies, and does not primarily look at the software products used to realize the services.

The methods used for the survey are qualitative because we take an in-depth look at a relatively small number of seven wayfinding systems. This is in contrast to quantitative research, where a large numbers of systems would have been compared in a more shallow way because each feature would have been reduced to a numerical representation. (See section 3.8, 'Qualitative research', page 40 for more about this topic). When the survey is being referred to as an 'evaluation', this uses the general meaning of the word and is not to imply that the methods of usability evaluation (e.g. usability testing with end users, see section 3.12 on page 47) have been employed.

The approach taken was to first identify evaluation criteria, and then use these criteria to describe each system (see chapter 8, 'Appendix: Evaluated wayfinding systems', page 119). Conclusions have then been drawn by comparing these descriptions.

The results gained from the survey are descriptive, not prescriptive or normative: The aim of the evaluation is not to find out which system is in some way better than the others, but rather to reveal a space of possible future designs. The observations made and conclusions drawn from the survey are presented in the following sections.

4.1.1 Evaluation criteria

A set of evaluation criteria is used to compare existing wayfinding aids. The criteria presented here have been identified by looking at differences between a first set of wayfinding systems and using the theory described in chapter 2, 'Wayfinding', page 9. The primary intent was to use these criteria to describe the systems chosen for evaluation. But the evaluation criteria can also be understood as design decisions that will have to be decided upon if a new wayfinding aid is to be built.

Since in contrast to other approaches, *online* wayfinding aids make use of internet technologies to provide their service, the evaluation criteria must include two aspects: The *geographic aspect* is concerned with the information we need to communicate to help another person find their way. The *technical aspect* is, in the case of online wayfinding aids, concerned with how the information is presented to the user using internet technologies. The following list includes evaluation criteria of both the geographic and the technical aspects:

- *Medium*: This criterium subsumes two aspects. Firstly, what technical medium is used to communicate wayfinding information to the user? E.g. Paper, an online or offline desktop application, a web application, or a mobile application. In most of the applications evaluated, this will be an online application (it needs an internet connection to work) built using web technologies (HTML, JavaScript, Flash etc). Secondly, what medium is used to communicate the geographic content? E.g. Are maps, photos or verbal instructions used to communicate the wayfinding information? See section 2.5, 'The medium used for wayfinding aids', page 17 for more about the role of the medium used in wayfinding aids.

- *Level of map abstraction and schematization*: How much detail is included in the map? See section 2.6, 'The necessary amount of detail in wayfinding aids', page 19 for more about map schematization.

- *Route versus survey knowledge*: Does the system primarily convey route or survey knowledge? Are routes dynamically calculated for the user? See section 2.7, 'Route versus survey knowledge in wayfinding aids', page 21.

- *Landmarks*: How many landmarks are included in the map? What kinds of landmarks? See section 2.8, 'Landmarks', page 25 for more about the role of landmarks in wayfinding systems.

- *Map orientation*: How are the maps oriented? E.g. North-oriented, in the direction of the route, or - as possible for mobile mapping - in the user's direction. See section 2.3, 'The orientation of maps', page 15 for the role of North orientation of maps.

- *Indoor space*: Does the system display indoor space? How does it handle the 3D aspect of several indoor floor levels? Does the system promote a specific indoor wayfinding strategy (central point, direction, or floor strategy)? See section 2.4, 'Indoor wayfinding', page 16 for the special requirements related with indoor wayfinding.

- *Search style*: What styles of interaction are offered to the user for searching for a specific room or location? E.g. browsing through hypertext or a menu structure, or a textual search for location names, or navigating through a map or virtual world
- *Navigation style*: If the user can navigate through some kind of virtual world (e.g. through panoramas or on an interactive map), what interaction style is used by the system? This can be none (if navigating is not possible such as with a basic, static image), hyperlinks, 2D zoom and pan for maps, zoom and pan for interactive panoramas, or full 3D interaction.
- *Internet Technology*: E.g. HTML, JavaScript, Flash, Java Applets, Quicktime Video

4.1.2 Evaluation

Seven online wayfinding applications have been chosen for an in-depth comparison. This choice has been motivated by the following considerations:

- The applications present *interesting concepts* in the author's opinion.
- The applications are all *different from each other* in some ways and thus show different design possibilities.
- Most of the applications are used for *on-campus or indoor* wayfinding.
- Most of the applications are *online, web applications* since '*online* wayfinding aids' are the focus of this work. (Online: they need an internet connection to work. Web application: they are built using web technology such as HTML, JavaScript, Flash or Quicktime)
- One of the applications (see section 8.1, 'The wegweiser.ac.at website', page 119) is *related to the Vienna University of Technology* in that it is the online wayfinding aid that is currently used.

A detailed description of each one of the evaluated systems can be found near the end of this work in the appendix in chapter 8, 'Appendix: Evaluated wayfinding systems', page 119, but the following list shall provide an additional overview by highlighting interesting features:

- *Wegweiser.ac.at*: This is the system that is currently in use at the Vienna University of Technology. It features static graphics that show several floor layers at once arranged in a 3D perspective. (See section 8.1, 'The wegweiser.ac.at website', page 119 in the appendix for a more detailed description).

- *University of Graz*: The system offered by the University of Graz (re-)uses very detailed real estate management plans as a wayfinding aid. (See section 8.2, 'University of Graz', page 121).

- *University of Innsbruck*: Besides a street map to show each of the building's positions, the University of Innsbruck's "student city map" features interactive 3D panoramic images. (See section 8.3, 'University of Innsbruck's "Student City Map"', page 125).

- *University of Karlsruhe, 2D maps*: The wayfinding aid offered by the University of Karlsruhe is particularly simple: It consists of one static image showing the campus and its buildings. Since the whole plan would be too big to show at once, the image can be zoomed and panned as needed. (See section 8.4, 'University of Karlsruhe's campus map', page 127).

- *University of Karlsruhe, 3D campus model*: A 3D model of the University of Karlsruhe's campus also exists. The user can fly around the virtual campus using the Google Earth desktop software. (See section 8.5, 'University of Karlsruhe's 3D campus', page 129).

- *TU München's "roomfinder"*: This wayfinding aid allows users to share links to any of the maps with others. The maps are simple static images that cannot be zoomed or panned, but the system provides hyperlinks to maps with different scales. (See section 8.6, 'Technical University of München's roomfinder', page 130).

- *Herold's Strassentour*: In addition to an interactive street map, this system features inter-linked panoramas for selected indoor locations. The user can zoom and pan around inside one panoramic image, and navigate to the next panoramic spot by clicking on an arrow symbol embedded into the panorama. (See section 8.7, 'Herold.at's Strassentour', page 132).

As mentioned before, the detailed evaluation of each one of these systems has been moved to chapter 8, 'Appendix: Evaluated wayfinding systems', page 119. The following presents the results gained:

4.1.3 Conclusions drawn from the evaluation

The author's general impression is that the systems evaluated in the course of this work each offer a very different experience. This should be evident from the screenshots included in this work (see chapter 8, 'Appendix: Evaluated wayfinding systems', page 119), but also from the comparison table (figure 22, 'Evaluation of existing solutions', page 57). This section presents the conclusions drawn from the survey:

Software development: In-house, no off-the-shelf software. All of the systems evaluated have - to the author's knowledge - been developed specifically as an online wayfinding aid. The authors impression

Title:	wegweiser.ac.at	Univ. of Graz	Univ. of Innsbruck	Univ. of Karlsruhe 2D campus map	Univ. of Karlsruhe 3D Google Earth	TU München's roomfinder	herold.at
Summary:	floor layers in 3D perspective	real estate management plans	interactive panorama	interactive map	full 3D	hyperlinks, maps in different scales	interactive interconnected panoramas
Medium	online, web application, 3D layered map	online, web application, map	online, web application, map & pano	online, web application, map	online, desktop application, 3D world	online, web application, map	online, web application, map & pano
Abstraction	high	low	medium	medium	none	different, high	map:medium panos: realistic
Route vs Survey	route	survey	survey	survey	survey	survey	both
Landmarks	yes, local	no	yes, one global	yes, local	photo-realistic	yes, local	few
Orientation	route, fixed	random, fixed	map: north, fixed panos: variable	north, fixed	variable	north, fixed	map: north, fixed panos: variable
Indoor	yes, 3D floor layers	yes, 1 map per floor	no	no	no	yes, 1 map per floor	interactive panoramas
Search styles	menu textual search	menu textual search	menu	menu	navigating	textual search	menu textual search navigating
Navigation	none	none	pan & zoom map and panos	pan & zoom	full 3D	none	pan & zoom for map and panos, hyperlinks from one pano to the next
Internet Techn.	HTML	HTML	Flash, QuickTime	Flash	Google Earth	HTML	HTML, Flash

Figure 22: Evaluation of existing solutions: A summary table showing all the systems that have been evaluated.

is that in-house development has been preferred over off-the-shelf software in most cases. Existing off-the-shelf software for indoor wayfinding includes but is not limited to Abysseo[1], here2there[2], zetek[3], micello[4], and Gebäudenavigation[5]. A more complete survey from a buyer's perspective, comparing free and commercial products, considering the product's software architecture and how it would integrate into the existing IT landscape, is left for future work.

Technical medium: Mostly web applications. The technical medium used for the wayfinding aids evaluated in this work is in all cases - except for University of Karlsruhe's 3D campus in Google Earth - a web application. The reason for this certainly has to do with the systems that have been chosen for evaluation since the focus of this work are *online* wayfinding aids.

Medium for geographic content: Mostly maps. From a geographic viewpoint, the medium used to communicate wayfinding information is in most systems a simple, two-dimensional map. Panoramic photos and a full 3D virtual environment are also found. None of the systems made use of verbal route descriptions beyond the provision of a room's address (usually a combination of the room number, floor number, and the building's address).

Abstraction: Rather different. The level of abstraction applied to the maps that were used in the evaluated systems varied widely, ranging from high levels of abstraction such as in Wegweiser (section 8.1) and TU München (section 8.6) to very detailed architectural plans such as in Uni Graz (section 8.2).

Route versus Survey knowledge: Mostly survey. Most of the systems evaluated provide survey knowledge through a map. One exception to this are Wegweiser's maps (section 8.1) because they indicate a route to get from a building's entrance to a certain room and thus provides route knowledge. Another exception are the panoramas used in Herold's system (section 8.7) because arrow symbols act as hyperlinks, guiding the user from one panorama to the next along a given route.

Landmarks: Generally few. Most systems do include information that can be used as landmarks (e.g. if a building has a special shape, this shape is visible on a floor plan, and can thus serve as landmark). But the systems seem to have a different focus: Only few landmarks are included, and often, the landmark information is only implicitely represented (like in the example of a building's special shape in a floor plan, or special buildings found in photorealistic representations like panoramic images). UniInnsbruck's (section 8.3) interactive map is special in that it features one global landmark, namely the big mountain range "Nordkette" that can be seen from everywhere in the city.

[1] http://www.absysseo.com/, June 2010
[2] http://www.here2theresoftware.com/, May 2010
[3] http://www.zetek.com/solutions_wayfinding.php, May 2010
[4] http://micello.net/, June 2010
[5] http://www.gebaeudenavigation.de, May 2010

Orientation: Mostly facing North. Most of the systems (except Wegweiser, section 8.1, and UniGraz, section 8.2) use North-oriented maps both for indoor and outdoor situations. In some cases, the North orientation has been slightly adapted so the building's walls align horizontally and vertically. The maps' orientation is fixed and cannot be changed: None of the systems (except for Uni Karlsruhe's 3D campus, section 8.5) allow the user to rotate a map. In the authors opinion, one reason for this may be technical difficulties, and another reason that the systems evaluated are not primarily meant for mobile use, and that being able to rotate a map to match the actual viewing direction is especially useful in a mobile situation.

Dealing with Indoor space: Different approaches. Not all of the systems evaluated show indoor space. Those that do use one of three approaches that are quite different from each other: The two universities in Graz and Karlsruhe (section 8.2 and 8.4) offer one map per floor level. Wegweiser (section 8.1) combines all floor plans into one graphic. Herold (section 8.7) uses interactive panoramas for both indoor and outdoor situations. All but one of the systems (section 8.7) that deal with indoor space provide a textual search for indoor cellular space names - this means that the user can search for a room's name or number.

Indoor wayfinding strategy: Floor-first strategy. The preferred indoor wayfinding strategy is to first go into the right floor, and then into the right direction (see section 2.4 for more about indoor wayfinding strategies). In most systems dealing with indoor space, the 3D aspect of how a wayfinder gets into the right floor is neglected. From this neglection we can deduce a floor-first-strategy, because the systems imply the user has to first make it into the right floor before he or she can even start to make use of the wayfinding instructions. In the wegweiser system (section 8.1), routes are drawn in 3D and give direct evidence that in this case, too, the floor-first strategy is preferred.

Visualizing the three dimensions: Mostly 2D projections. Most of the systems evaluated do not visualize all three dimensions, but provide two-dimensional plans each for a different floors level instead. Two systems however had a different way of dealing with three-dimensionality: Wegweiser's maps (section 8.1) show several floor plans at once in a 3d perspective. And University of Karlsruhe's 3D campus (section 8.5) uses Google Earth to create a full 3D experience, but this system does not include indoor space.

Search style: Three possibilities that can be combined. The systems evaluated each offer a subset of three possible search styles: The first possibility is a menu structure or hypertext that allows the user to search for a location's name within the hypertext. The second possibility is a textual search box that allows the user to enter a location name and hit the search button. And the third possibility is that the user is in some cases able to search for a location by navigating within the virtual representation e.g. by zooming and panning an interactive map. It is interesting to note that one system (Herold, section 8.7) combines all three search styles.

Navigation style: Mostly pan-and-zoom. Some systems allow the user to navigate within the virtual representation, e.g. by zooming and panning a map. The interaction style employed by those systems is very similar: A user may use drag-and-drop mouse gestures in order to pan either a map, a panorama or the viewport of a full 3D application. A user may re-center the map by double-clicking on it. And a user may zoom in and out using either a slider control or the mouse wheel. This interaction style is employed by all systems. The only exception to this observation is found in Herold (section 8.7), where the user can also navigate from one panorama to another by following hyperlinks.

Internet technology: Mostly HTML and JavaScript. Most systems are built using HTML and JavaScript, but some systems additionally employ Flash or Quicktime.

The evaluation of wayfinding aids that was carried through in this work showed a great variety of possible solutions. A number of relevant design decisions were shown in this section. Taken together, these decisions span up a space of possible future solutions. In order to know which of all the possibilities best suits Vienna University of Technology's situation, the author suggests and uses a user-centric approach for both map design and software design.

4.2 Interviews

Qualitative interviews were performed during the analysis phase of this work. The purpose of the interviews was to better understand the users and their wayfinding behavior on campus:

- *The user*: To better understand who the users are, and which characteristics are relevant to how they will be using the wayfinding system.
- *The tasks*: How do the users currently accomplish the task of finding an unknown place? How can they imagine using the wayfinding system in the future?
- *Wayfinding cognition*: What do the users' mental maps look like? What orientation do they use to describe a route? Are there differences between indoor and outdoor situations?
- *Best practices*: What works well and what does not with the current wegweiser[1] system. How much abstraction is needed, and what should be featured on the maps?

Qualitative methods were used to design and conduct the interviews: Instead of a large number of relatively shallow answers that can be numerically evaluated, the author strived for an in-depth look with a relatively small number of interviews. The type of the interviews was somewhere between free and focused: While the author did prepare some questions of interest, he did not always stick to the pre-defined structure and encouraged the interviewees to tell experiences of their own by posing open,

[1] `http://www.wegweiser.ac.at/`, Sept 2009

immanent questions. (See section 3.9, 'Qualitative interviews', page 42 for more about different types of interviews). The following subsections describe the assumptions made prior to interviewing. The interview's design is then explained, and conclusions drawn from the interviews are presented.

4.2.1 A-priori assumptions before starting the interviews

As explained in the theoretical part of this work (see section 3.8, 'Qualitative research', page 40), the perspective taken in qualitative research is to start observations and interviews with the least possible a-priori assumptions. Since this is not always possible in practice, the author wants to be conscious about the assumptions he is making before making any observations. He was aware that some of these assumptions may have to be revised, but found most of them confirmed during the course of examinations.

- *Wayfinding*: The whole point of this work is about the creation of a wayfinding system. This means that the primary goal of the resulting application will be to enable people to better find their way on campus.

- *User categories*: The primary users will be: Students who are new to the campus. Students who know the campus. Teachers and other university staff. Visitors. Interviews were performed with two teachers and four students who either knew or did not know the university's campus.

- *Use Cases*: The primary use cases of the wayfinding system will possibly include: searching for a place, printing/copying/sending/sharing the wayfinding information, linking/publishing a place. This assumption will guide the design of the qualitative interviews. The interviews are likely to refine the before-mentioned use cases. (And they did in fact change the use cases: E.g. none of the students was interested in publishing wayfinding information. See subsection 4.2.3, 'Conclusions drawn from the interviews', page 63)

- *Web application*: The primary focus will be a web application. Not a desktop application that needs to be downloaded and installed. Not an offline application, but one that needs an internet connection to work. Not a mobile application made for use with cellphones or PDAs. But an application built using web technology and accessible through a webbrowser. This choice has to do with how the task for this thesis has been assigned. Nevertheless, mobile wayfinding has been considered in the survey (section 4.1, 'A survey of existing solutions', page 53), in the interviews (this section), and during the design phase (chapter 5, 'Design phase', page 72).

4.2.2 Planning the interviews

The following guideline of questions and topics was prepared before conducting the interviews. Two indirect questions were used to simulate certain situations in a similar way to role plays. These questions also included observational elements through tasks that the interviewee had to perform (e.g. to draw a map on a sheet of paper while explaining a route). The atmosphere was quite different depending upon the interviewee's role (student or teacher). Additional topics were often discussed ad-hoc.

- *Introduction*: After saying hello, I introduce to the topic of the interview, namely how to find lecture halls or other unknown places on campus. I tell the story of how my thesis may lead into a follow-up project for creating an online wayfinding aid. I explain that I have prepared some questions but that we don't have to strictly stick to them. I encourage to answer the questions with whatever comes to your mind, by telling about your experience, and by saying whatever matters to you. I ask for permission to record the talk because I would otherwise have to concentrate on taking notes. The recordings will not be published, and the results from the interviews will be anonymized.

- *Searching for a place*: Imagine you have to find a lecture hall where you have never been before. E.g. let's say lecture hall 18. What would you normally do? – This question is interesting for how users approach a wayfinding task: Do they call somebody? Try to look it up on the internet? Print a map? Write down some notes on a piece of paper?

- *Sharing wayfinding information*: What is a place you know well? Ok, imagine a friend of yours calls you, saying he needs to find this place but he forgot to look up where it is. That friend is me now. Can you please explain to me how to find this place? – This question is interesting for how wayfinding information is shared verbally. Also, what information is needed for textual descriptions?

- *Publishing wayfinding information*: Have you ever had to publish how to find a place, let's say at your institute's website, or in an email or Word-file sent to many people? How did you do it? How could this ideally work?

- *Wishes and dreams*: A new website for campus maps shall be created. What would be important to you, what would be your suggestions? What would be the worst error this project could make?

- *Mental map*: What is a place you are well familiar with on campus? Can you draw a map for me and give me instructions so I can find it as well? – This questions provides the user's mental map and the map's orientation. Does the user provide route or survey knowledge? What is his/her indoor wayfinding strategy? How big is the level of abstraction?

- *Mobile Wayfinding aids*: Do you have a PDA or a smartphone? Do you use it with maps? What works well, what doesn't? Can you show it to me?

- *Existing maps*: This is a map from the wegweiser system. Can you read it and explain the route to me? How do you like the 3D perspective? What do you like about the maps, and what don't you like? Similar questions about the real estate management's architectural floorplans and their new campus overview maps.

- *Some closed questions*: Some very quick yes/no a-lot/not-at-all questions: How do you like the wegweiser system? Would you welcome a new system? How often do you look up unknown places? When a new system is installed, would you mind if everything works differently? – I never posed these questions because they were redundant given what the users had been telling me before.

- *Thank you and good bye*: I thank for the interviewee's time and effort. I explain again how I will use the information and what may come out of my thesis.

4.2.3 Conclusions drawn from the interviews

A total of six interviews were conducted with representatives of different groups of users: Two students who knew the campus, two students unfamiliar with it, and two teachers were also interviewed. The interviews led to a wealth of information of more than 100 minutes of speech or 10.000 words. The talks were recorded and transcribed. Statements associated with common themes were then collected. All of this work has been performed in German, and only the results from the last step were translated to English. The conclusions drawn from the interviews guided future design and particularly fed into the persona and use case definitions (see the following section 4.3 and section 4.4). The results are presented in the following:

- *The purpose of the envisioned system*: To find one's way quickly. To know where a lecture hall is. To know where to find a person. A similar experience online (using the envisioned wayfinding system), mobile (if a mobile wayfinding aid is provided) and locally (using signs and maps attached to buildings or signposts).

- *Alternatives to using the system*: Asking friends e.g on the phone. Going to a place together with colleagues. Asking the doorman. Using wayfinding information on the institute's homepage.

- *Ways of accessing the system*: Current wayfinding systems have been accessed by the interviewees using the following methods: Often by following links from other systems (TUWIS lecture details, TUWIS agenda/calendar, the library's homepage). But also directly, by going to the wayfinding system's website.

- *Memorizing wayfinding information*: For partly-known places, it suffices to memorize or write down the place's address, usually consisting of the building's name, the floornumber, and the room's number or the lecture hall's name. For lesser-known places, printing a map or drawing a map by hand (e.g. if no printer is available) is an option.

- *Sharing wayfinding information*: Wayfinding information is primarily shared with others verbally. The verbal instructions given basically consist of a place's address as explained above. For lesser-known places, meeting beforehand and going to the place together is an option.

- *Publishing wayfinding information*: None of the students had to publish wayfinding information to a wider audience. One teacher doubted there was an actual need for a usecase such as 'publishing wayfinding information' using the new system, but another teacher specifically asked for the possibility to share links pointing to a wayfinding description.

- *Mobile devices*: Current usage of wayfinding information on mobile devices did not go beyond adding an address to an existing date in a calendar on the mobile device. This was mostly due to unsuited mobile hardware. The idea of mobile wayfinding was however appealing to most of the interviewees. From this, the author concludes that greater availability of advanced mobile devices will foster demand for mobile wayfinding.

- *Map orientation*: In order to gain insight into the orientation of the interviewee's mental map, they were asked to draw instructions on a sheet of paper to explain how the interviewer would find his way to a certain place. Most of the drawings focused on indoor maps and used the perspective of a person entering the building as orientation, i.e. the maps were not necessarily North-oriented. One drawing however used North-orientation despite the fact that a person entering the building looks southwards. Since the drawing included a lot of (outdoor) streets, the observation made in [Gartner and Radoczky, 2006] and described in section 2.3, 'The orientation of maps', page 15 of this work is confirmed, namely that mental maps of indoor places rather use the building's entrance as orientation, while mental maps of outdoor and street situations tend to be North-oriented. The implication for an indoor wayfinding system is that offering North-oriented maps will improve finding the building, while offering route-oriented maps will improve indoor navigation. A hybrid solution offering both orientations may be a viable solution, but many interviewees had difficulties matching two maps with different orientations. The author concludes that each solution is a trade-off and that it is difficult to give a clear recommendation.

- *Route versus survey knowledge*: The interviewees were asked to explain verbally and by drawing how to get to a certain place. The resulting instructions always indicated the route to take and only included the least possible survey knowledge.

- *The necessary amount of detail*: When explaining how to get to a certain place, the interviewees could include just as many details as were necessary for me (the interviewer) to understand the instructions. In contrast to this situation, a wayfinding system will have to be understood by a large group of heterogeneous people. Hence the amount of detail given by the interviewers is not representative for the necessary amount of detail of wayfinding system's maps or instructions. In order to gain more representative results, another approach was chosen instead: The interviewees were asked which one of three maps with different levels of details they preferred. The general result is that the interviewees preferred many details and landmarks for street maps, but rather abstracted and schematic representation for indoor space. They particularly disliked and had problems with detailed architectural plans.

- *Labels and Landmarks*: Labels that should be contained in a map: Street names, Building names, room numbers (but just for the one room of interest). Landmarks mentioned in route descriptions given by the interviewees included the opera house, the Karlsplatz fountain, public transport stations, cafés, bigger streets, outstanding buildings (e.g. "the owl building" = the Technical University's library building featuring a big owl on one corner), statues, and other university buildings or lecture halls.

- *Three-dimensional indoor space*: Some interviewees disliked three-dimensional representations of different floorlevels in one graphic. One person had difficulties interpreting the stack of floor layers correctly. The other interviewees were able to interpret the graphic correctly.

4.3 Personas

There is no single best user interface for everybody because different users have different requirements. The aim of creating personas is to characterize these differences. This will enable the designer to make informed design decisions [Mayhew, 1999, p.35]. Personas can also help discussing what "the user" wants, because they fill this vague term with specific personalities, skills and preferences.

The approach taken to creating personas: As described in the previous section, interviews have been conducted to learn about different user requirements. From these interviews and using the theoretical foundation of this work, relevant user characteristics were identified and combined to form a persona. The results of this work are shown in figure 23, 'Personas: An overview', page 66. Starting from these sets of properties, the author invented a story and added a name and a face. Please note that personas are not real people, but invented characters that are meant to help the design process. Any resemblance of the character description to a real person's life is by accident. The photos used are licensed under creative commons license. The detailed persona descriptions can be found in chapter 9, 'Appendix: Personas', page 135.

Personas (an overview)

Personas have been created by combining user characteristics into a person with a name, a face and a story:
↓ User characteristics Personas →

	Sun-Li (Lilly)	**Peter**
Campus knowledge	✓ Knows the campus well	✗ Does not know the campus
Knowledge of academic terms	✓ Lives in the academic world	✗ New to the academic world
Computer skills	✓ Good computer skills	✗ Very limited computer skills
Language skills	✗ Bad German, perfect English	✓ Native German speaker
Experience with wayfinding systems	✓ Some experience with the previous system	✗ No experience with electronic wayfinding systems
Frequency of use	✓ Often, but mainly in the beginning of a term	✗ Will be using the system only a few times
Disabilities	✓ Knee injury, problems with stairs	✓ Limited sight

Figure 23: Personas: An overview and a comparison of the two personas Peter and Lilly that have been created to help designing the wayfinding system. The table shows how relevant user characteristics have been combined to invent a character.

The following user characteristics were identified:

- *Knowing the campus* or being new to the campus. Judging from the interviews, this makes the biggest difference, because for a person who is well-acquainted with the campus, simple address-style instructions are sufficient, whereas people who are new to the campus need more detailed instructions, possibly including a map.
- *Knowledge of the Austrian academic system* . E.g. knowing what is meant by 'lecture hall' or 'conference room'.
- *Computer skills.*
- *Language skills.* German and English are the university's main languages. Wayfinding instructions are likely to be provided in these two languages.
- *Experience with wayfinding systems*, e.g. with the previous wegweiser system.
- *Frequency of use* of the wayfinding system: How often and how regular.
- *Disabilities.* E.g. bad vision or limited mobility.

Starting from the above list, two personas have been designed that give a name and a face to the otherwise rather abstract user characteristics. (See figure 23, 'Personas: An overview', page 66 for a comparison). *Li-Sun (Lilly)* knows the campus well and is well-acquainted with the Austrian academic system. She has excellent computer skills. Her German is rather poor because she speaks English most of the time. She uses the wayfinding system frequently when a new term starts. Due to an injury, she had difficulties climbing stairs last year. See figure 49, 'Persona: Lilly', page 135 for a closer description. *Peter*: Peter is new to the university's campus and does not know the Austrian academic system and the sub-language that comes with it. He has poor computer skills. German is his native language. He is using the wayfinding system for the first time, and won't need to use it more than a few times. He has limited sight. See figure 50, 'Persona: Peter', page 136 for a closer description.

4.4 Use cases

Before the interviews were conducted, the author had already made assumptions about possible future use cases. (This was needed to design good questions for the interviews, see subsection 4.2.1, 'A-priori assumptions before starting the interviews', page 61). Most of these assumptions were confirmed by the interviews. We conclude with the following list of use cases:

- *To follow a link* from another system that leads to a wayfinding description.
- *To search for a place* (or a person's office) by going directly to the wayfinding website. The place's name (the person's name) is known.
- *To read and understand* the wayfinding instructions. This includes knowing how to get to the building and to understand the indoor route within the building.
- *To write down or print* wayfinding instructions. This includes adding the address to an electronic calendar, possibly on a mobile phone, but also to simply write down instructions on a piece of paper.
- *To share wayfinding instructions* with others. This does not have to be done using the system (e.g. calling a person and explainding the route is an alternative). But the option to copy a link to the current location was specifically requested by one interviewee.

The overall purpose of these use cases is to find one's way quickly to a lecture hall or to another place or person. The navigation itself (i.e. walking around town and on campus) has not been considered a use case because the wayfinding system is not involved anymore. This would be different with a mobile wayfinding system.

4.5 Data needed for the wayfinding aid

This section contains a list of data that is useful for providing a campus wayfinding aid.

Indispensable data: Using this data, basic wayfinding instructions can be provided. A search function and a hierarchical structure to browse entities can also be created. Without this data, a wayfinding aid cannot be created.

- *Important entities*: About 17 buildings. Over 50 lecture halls and more than 150 conference rooms or other rooms of interest. The offices of over 120 institutes belonging to 8 faculties. Several teachers per institute (no exact figure available) making several hundred teachers and their office rooms. This data is available from the Vienna University of Technology's real estate management. It is currently (as of May 2010) being restructured for the TISS project[1], and a database dump from this project has been used to create the prototype. This large amount of data makes it necessary to build the system using a lot of automatization.
- *Name and description for each entity*: For each entity, a name and a description should be provided. Building names should be user-friendly in that they reflect how people actually call the building instead of the somewhat cryptic codes used by real estate management. This data is partly available by the real estate management and the TISS project, but needs a lot of editing.
- *Textual wayfinding instructions*: A textual description how to find each entity. Possibly including public transport options. This data is currently not available.

Map data: If the wayfinding aid is to include an interactive map, the following data will be needed:

- *Street map*: A street map as base layer will help the user identity a building's location. Many internet mapping providers allow to use their maps for free (e.g. Google, Yahoo, Microsoft). The OpenStreetMap project[2] goes one step further and also offers all of the underlying map data for free.
- *Floor plans*: Floor plans showing indoor locations (i.e. rooms) can be added to the map as overlays. Using this approach, only one floor level can be shown at a time. (Note that this can actually be an advantage, as interviews showed problems reading three-dimensional plans: Some interviewees disliked the 3D representation, and one had problems interpreting it correctly. See subsection 4.2.3, 'Conclusions drawn from the interviews', page 63). Floor plan drawings currently only exist as architectural plans. The approach taken in this work was to re-draw and simplify these plans to better suit the needs of a wayfinding aid.

[1] `https://tiss.tuwien.ac.at/`, Sept 2009
[2] `http://www.openstreetmap.org/`, July 2010

- *Locations for each entity*: In order for an entity to be shown on the map, the system must know its location. The location could be specified in 2D coordinates with a reference to a floorplan map layer as third coordinate. This data is, to the author's knowledge, currently not available.
- *Paths or routes on the map and on the floorplan*: A graphical representation of the route instructions can be provided in the form of a line leading into a building, then icons for stairs that need to be taken into a specific floor. From there, the line continues to show the indoor route to a specific room. This data is, to the author's knowledge, not available.

Floor plans: While drawing floorplans by hand is possible, automatic derivation from the university's real estate management would be interesting because they use and possess recent and well-maintained data. (Note however that standard architectural plans should not be used for wayfinding aids, as could very clearly be seen from the interviews. Instead, wayfinding plans should be much simpler and abstract away architectural details). Either way, whether drawn manually or automatically, the following data will be needed – and most of this data is available in the real estate management's architectural plans:

- Building shapes, courtyards
- Colored building sections
- Indoor shapes of rooms and hallways
- Stairs and elevators
- Landmarks such as cafés, lecture halls, toilets

Landmarks: The wayfinding system should include landmarks to aid orientation. Judging from the interviews, this should include public transport stations, cafés and supermarkets, the nearby St.Charle's church and the fountain in front of it, the opera house, and the library building's façade featuring an owl. The wegweiser system[1] that is currently in use at the Vienna Technical University features many additional landmarks. Further investigation for the kind of landmarks needed are left for future work.

Other media: For each entity, information can be provided using other media than text. Photographs or drawings to help recognize a building would be particularly useful. No such data is available to the author's knowledge.

[1] `http://www.wegweiser.ac.at/`, Sept 2009

4.6 Integration with other systems

Integration with other systems is essential for the wayfinding aid in construction. This is true for several reasons:

Integration through hyperlinks from other systems: A primary way of accessing the wayfinding aid will be other system's hyperlinks pointing to a specific location in the wayfinding aid. This is at the same time the simplest form of integration, because any other system may 'integrate' by simply linking to a place in the wayfinding system.

Integration through hyperlinks to other systems: In a similar way, the wayfinding system may link to information in other systems. E.g. when displaying a room, the wayfinding system could provide a link to the university's room reservation system, or a link to the lecture timetable of a lecture hall.

Data-Integration for information about people, organizations and rooms: Other existing applications such as the University's whitepages know *who* (which person or organization) has his/her office *where* (in which room). This existing information would better be re-used than re-created redundantly.

Data-Integration for automated map generation:

- For easier maintenance: As already mentioned in the above section: If the maps are automatically generated from the University's real estate management data, chances are they will be more up-to-date.
- Because of the many entities: As described above, wayfinding instructions should be provided for several hundred buildings, rooms, and persons. Creating all these maps by hand would mean a lot of work.
- Because of different zoom levels: This work proposes to add indoor floorplans as map overlays to an interactive, zoomable streetmap. In this case, floorplans must be rendered in different resolutions (otherwise, text loses readability when zooming in and out of the map)

Integration with geographic hypermedia systems: To create the proposed wayfinding system, a technical solution for interactive maps is needed. For this purpose, existing geographic hypermedia systems are better re-used than created from scratch. E.g. to create the prototype featured in this work, the author made use of the Google Maps Javascript framework[1].

4.7 Conclusions drawn from the analysis phase

This section sums up the most important conclusions that were drawn from the analysis phase. These considerations are at the same time the starting point for the next phase in the development process.

[1] `http://code.google.com/apis/maps/`, June 2010

The users: The most important distinction is between users who know the campus and users who don't know the campus. (Other distinctions are presented in section 4.3, 'Personas', page 65).

The tasks: To find a place quickly (a lecture hall or another room). By following links, by searching, by reading a hypertext structure, by looking at maps. To write down, print, or share this information. (See section 4.4, 'Use cases', page 67).

The content: Lecture halls, conference rooms, offices of institutes and teachers. Textual and visual route descriptions, not just survey knowledge. Map orientation is a trade-off between indoor, route-oriented and outdoor (North-oriented) maps. (See section 4.5, 'Data needed for the wayfinding aid', page 68 and subsection 4.2.3, 'Conclusions drawn from the interviews', page 63).

The technical aspect: The system includes a lot of content, which means it will have to some extent be automatically generated. For this, integration with other systems will be necessary, namely with the real estate management software and with the whitepages application. (See section 4.6, 'Integration with other systems', page 70).

5 Design phase

The design phase is where 'the actual work' is done. This is not to say that the other phases are less important, easier or even shorter, but the design phase is where step by step, the desired product is formed. In the case of this thesis, the result of the design phase's last iteration is the finished prototype. Because this result is extensively described in the last phase (chapter 6, 'Presentation phase', page 100), this chapter will instead focus on the work in progress: Reading this section, one can watch the prototype develop from vague ideas to a rather detailed specification. Many wireframes have been drawn and a total of six prototypes have been built, formulating different aspects and gaining more and more detail as the work progresses. Design possibilities and decisions made are documented through text and graphics.

5.1 Storyboards – imagining future use

Storyboards are small episodes about the user's tasks. Each story is told using a combination of pictures and text as in comic books (see section 3.10, 'Personas and storyboards', page 44. In this work, storyboards have been used with the intention of trying to imagine future use of the university's wayfinding system. This means that the stories told do not describe the current situation, but already include ideas about the envisioned system – the storyboards already contain first steps of system design and work re-engineering. "Anti-storyboards" have also been created in this work, illustrating ways of getting wayfinding information without using the wayfinding system. The author came up with this idea so he could tell more about the context of use. This section only includes one exemplary storyboard, the rest of the storyboards are shown in chapter 10, 'Appendix: Storyboards', page 137.

The first two storyboards, figure 24, 'Storyboard: Lilly and Peter', page 73, illustrate two typical ways how the personas Peter and Lilly would use the wayfinding system: Lilly, as a research-assistant, knows how to find wayfinding information by following links from other systems. She is comfortable sharing a link with others via email because she has done the same thing through Google maps already. Peter who does not know the university first needs to make his way through Google and different links on the university's website in order to get to the wayfinding information.

The storyboard shown in the appendix in figure 52, 'Storyboard: Looking for a teacher', page 138 shows how people and places are connected through the task of searching someone's office. It also illustrates the option to print the wayfinding instructions (instead of memorizing them, or writing them down into a calendar or unto a sheet of paper).

The "anti-storyboards" in the appendix in figure 53, 'Anti-Storyboard: Ways of not using the system', page 139 show alternatives to using the wayfinding system: Calling friends, asking people or asking at an info point, meeting beforehand with colleagues, or making an educated guess and relying on local signage are popular options, judging from the interviews.

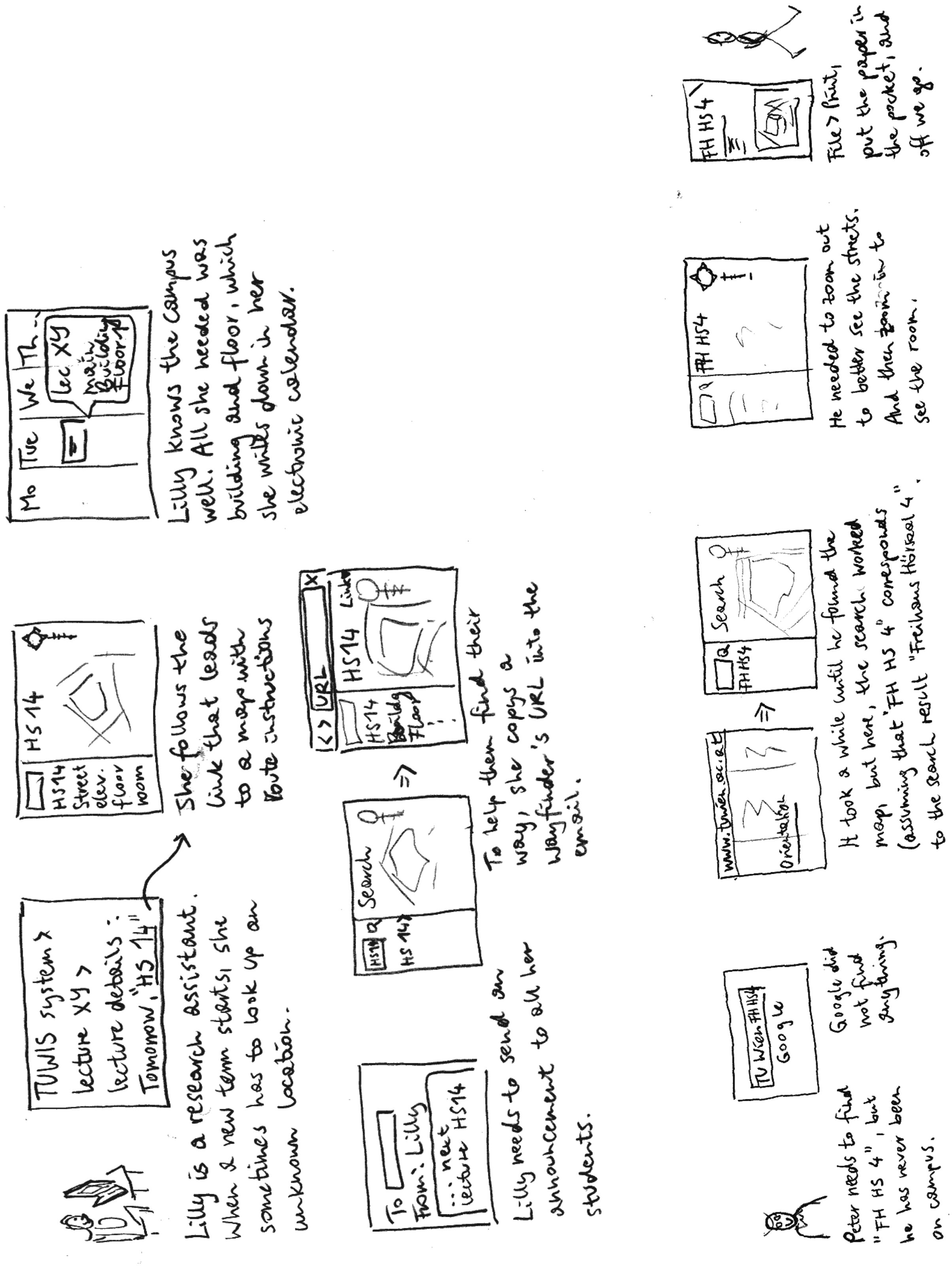


Figure 24: Storyboard: Lilly follows a link from the TUWIS system and shares a link with others. Peter needs to find the wayfinding website using Google, and then continues using the search-function. More storyboards are included in chapter 10, 'Appendix: Storyboards', page 137.

5.2 Wireframe sketches and quick prototypes – capturing the initial idea

Wireframe sketches have been created to explore different possibilities for the envisioned systems. Most of the figures in this section have first been drawn by hand using pen and paper. Only at a later stage, a drawing program was used to further develop ideas and to get a more exact impression of what the resulting product would look like. In the author's experience, this allows for quicker prototyping compared to using the computer from the beginning. Paper prototypes (see figure 25, 'Exemplary paper mockup', page 75 for an example) provide the same advantage: They are easy to create and manipulate and can – despite their simplicity – still give a good impression of the future system.

Within the process of designing wireframes, the following points have been considered:

Four ways of accessing the system: Results from the interviews and the survey of existing solutions (see chapter 4) show that there are four possible ways of accessing wayfinding information. The system can be designed in a way that allows for all four possibilities, as shown in figure 26, 'Wireframe: Four ways of accessing the system', page 77:

- *Hyperlinks from other systems*: Links from other systems or webpages lead directly to the display of a location.
- *Search*: The wayfinding system allows the user to find places by entering text into a search box. (See section 5.6, 'The search function', page 91 for more about the design of the search function).
- *Navigation on the map*: The wayfinding system allows the user to navigate directly within the map's virtual world by zooming and panning around.
- *Hypertext*: The wayfinding system offers a hypertext structure, e.g. lists of lecture halls and other places. The hypertext contains verbal route instructions and images. (See section 5.4, 'Designing the hypertext', page 82)

Four types of media: The wayfinding system includes four different types of media to communicate wayfinding information (see figure 27, 'Wireframe: Four types of media', page 78). The redundancy that is thereby introduced is not only positive, but necessary for a good wayfinding aid (see section 2.5, 'The medium used for wayfinding aids', page 17).

- *Map*: An interactive street map with overlays for university buildings and floor plans.
- *Pictures*: Photographs or drawings to help people visually recognize a building.
- *Textual description*: A textual "how to get there" wayfinding description including the address of a place and possibly public transport options.
- *Hypertext structure*: Browsing through a directory-structure reflecting real-world containment relationships: E.g. the university's campus contains buildings and people, a building contains lecture halls etc. (See figure 31, 'A proposed hypertext structure', page 85)

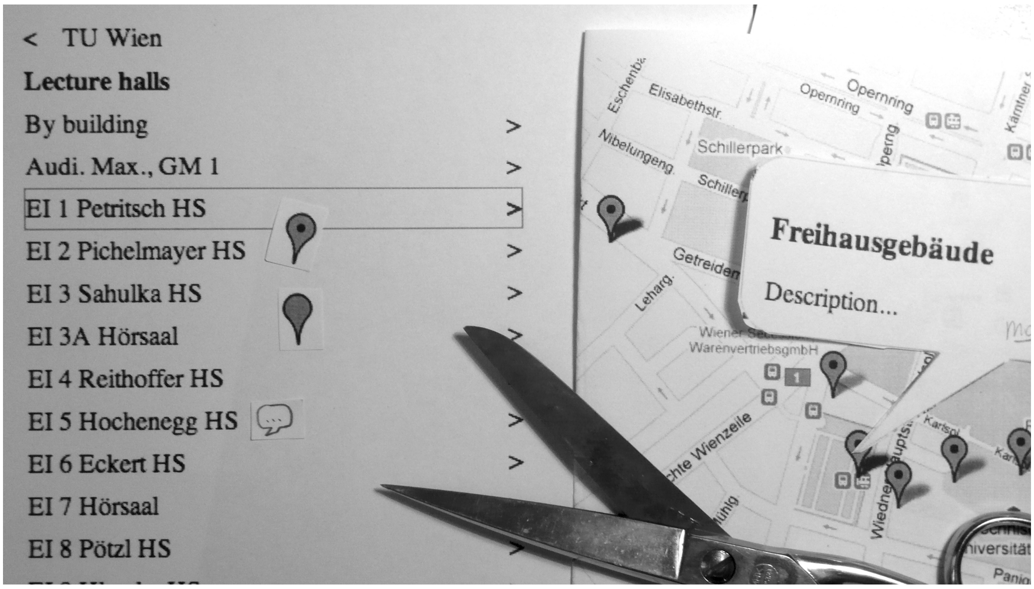


Figure 25: Exemplary paper mockup: Using pen, paper and scissors to try out variants of a prototype is faster than programming. This makes it easier to reject bad ideas and to start again from scratch.

Combining the four ways of access and the four type of media: The ways of accessing the system and the different types of media need to be integrated into one coherent user interface. One possibility is to split the screen into a sidebar and a map. (See figure 28, 'Wireframe: Combining ways of accessing the system and media', page 79).

- *Split screen layout*: Splitting the screen as described above works well for large desktop screens, but the small screens of mobile devices need a different solution. A mobile wayfinding system could allow the user to switch between the map and the hypertext view, or include the hypertext into the map in the form of popup bubbles, as shown in figure 28, 'Wireframe: Combining ways of accessing the system and media', page 79.
- *Coordinating map and sidebar*: When the user navigates the map, the sidebar's content should change accordingly, and vice versa. This concept is further elaborated in section 5.8, 'Designing interaction between map and sidebar', page 96.

Dealing with the 3D aspect: Indoor situations contain the 3D aspect of several floor layers. The three dimensions are difficult to describe visually (e.g. on a map), but easy to describe verbally (it usually suffices to tell somebody to 'go into this or that floor'). For this reason, the prototype will not visually represent the 3D aspect. Instead, 2D floorplan images will be used as an overlay over a regular streetmap, and instructions about the right floor level will be given in the verbal route descriptions. (As shown in figure 27, 'Wireframe: Four types of media', page 78)

Street maps: Results from the interviews show that a street map is a useful addition to the campus maps: Students who were unfamiliar with the campus had been using Google Maps in addition to the campus maps because they first needed to see where the building was and how to get there e.g. using public transport. The idea embodied in the wireframe sketches is to combine a regular street map with the floor plans.

Map orientation: There is typically a conflict between the mental map's orientation for indoor and outdoor situations: Outdoor maps are North-oriented, whereas indoor maps often use the route's direction as orientation (see section 2.3, 'The orientation of maps', page 15). Showing two maps with different orientations (one for indoor, one for outdoor) is problematic, judging from the interviews. The prototypes therefore simply use North-orientation for both indoor and outdoor situations.

Map navigation: The user interaction design for navigating an interactive map is very similar throughout all major online mapping providers (e.g. Google, Microsoft, Yahoo) and for all the systems evaluated in section 4.1, 'A survey of existing solutions', page 53. Basically, the map can be panned using mouse drag actions, re-centered by doubleclicking, and zoomed using the mousewheel or by clicking on +/- buttons of a map navigation interface element. The same interaction design will be used for the project at hand.

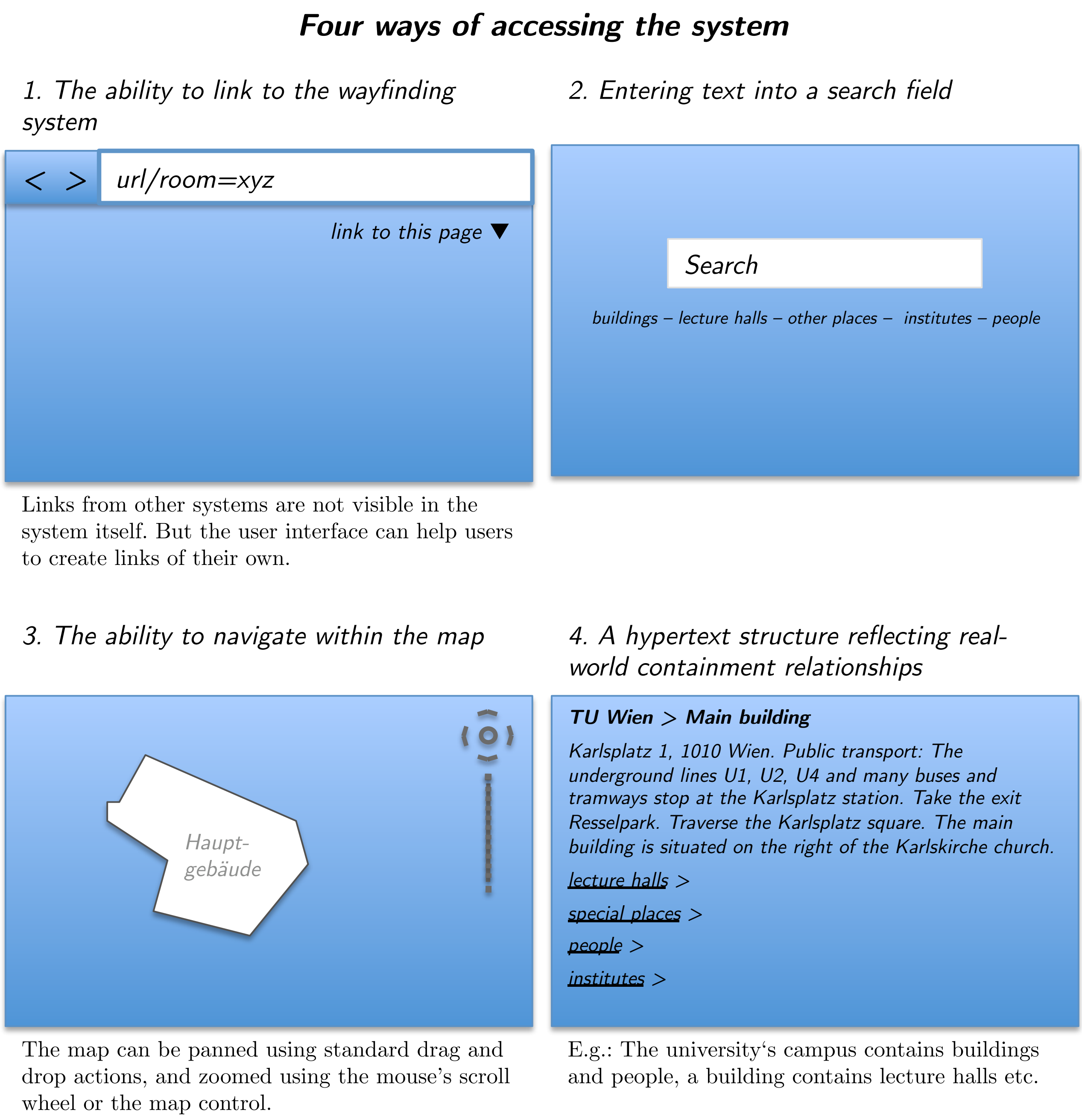
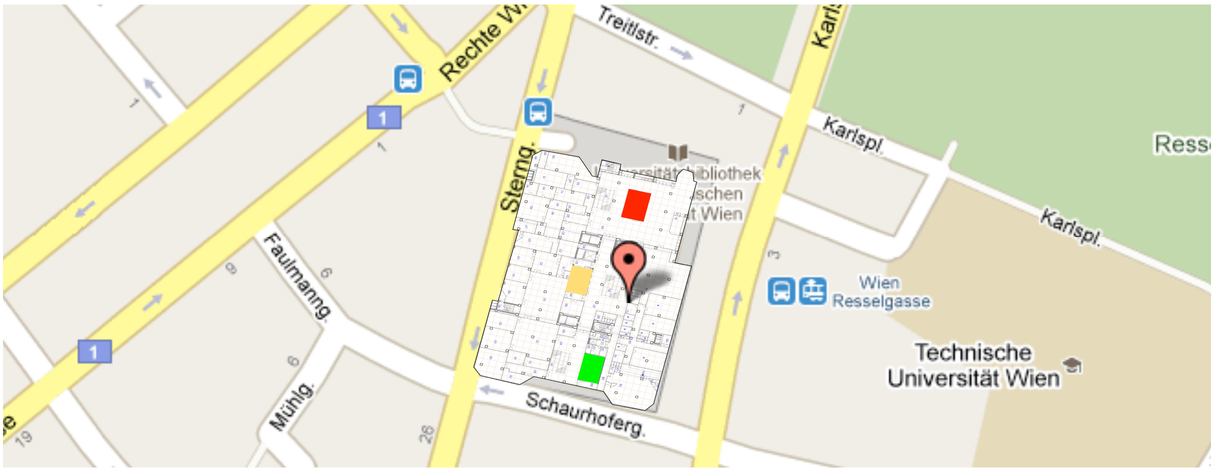


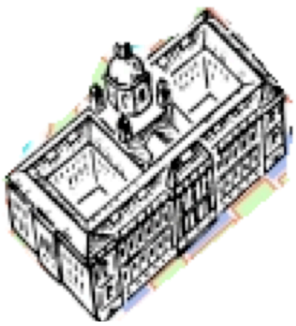
Figure 26: Wireframe: Four ways of accessing the system: Incoming links from other systems, textual search, navigation within a map, and browsing of a hypertext structure.

Four different types of media

1. An interactive street map with overlays for university buildings and floor plans.



2. Photographs or drawings to help people visually recognize a building



3. A textual "how to get there" wayfinding description including the address of a place and possibly public transport options

E.g.: Lecture hall HS 1. Main building, Karlsplatz 1, 1010 Wien. Public transport: The underground lines U1, U2, U4 and many buses and tramways stop at the Karlsplatz station. Take the exit Resselpark. Traverse the Karlsplatz square. The main building is situated on the right of the Karlskirche church.

4. A hypertext structure reflecting real-world containment relationships

E.g.: The university's campus contains buildings and people, a building contains lecture halls etc.

Figure 27: Wireframe: Four types of media are included into the wayfinding system.

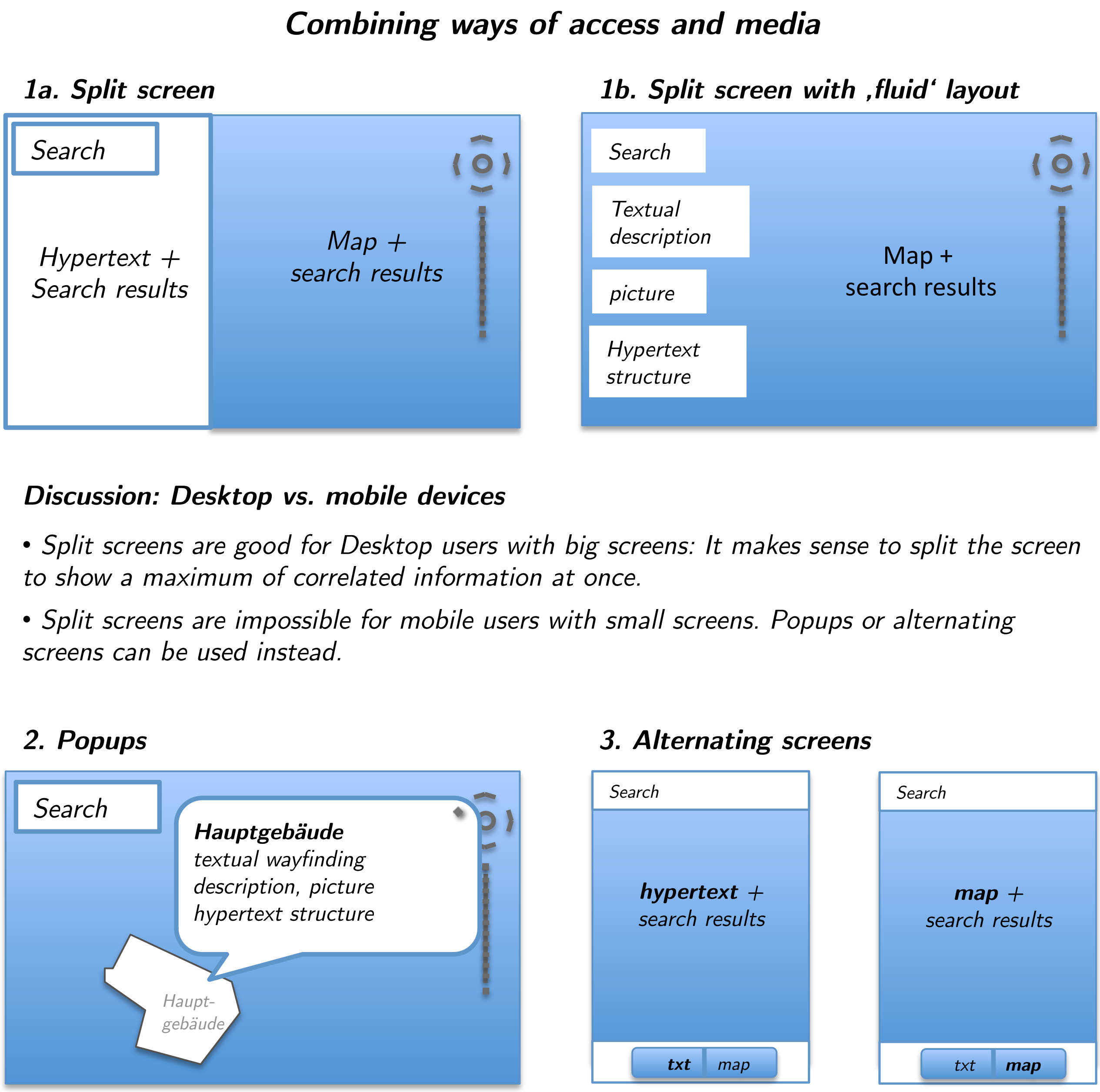


Figure 28: Wireframe: Combining the four ways of accessing the system and the four types of media: Split screens are good for large displays, whereas the small screens of mobile devices require a solution using popups or alternating screens.

5.3 Floorplan overlays – a prototype to evaluate the technical feasibility

A technical prototype (see part a of figure 29, 'Technical feasibility prototype', page 81) was built early on during the design phase to ensure that the system specified through wireframe sketches can be built with a reasonable effort. The prototype demonstrates the technical possibility of combining a streetmap with floorplans by overlaying images over Google Maps. Instead of using images, one could also use vector objects as overlays. This is demonstrated by the University of Wisconsin Oshkosh's campus map[1], which is shown in figure 29, part b.

- The Google Maps API[2] is used to load a map layer of Google's street maps
- Floor Plan images and placemarks are defined in KML[3] files. The KML files can be written by hand or created using the Google Earth[4] desktop software.
- The kml files are dynamically loaded and parsed by the webbrowser. The floorplan and placemarks are then rendered on the map.

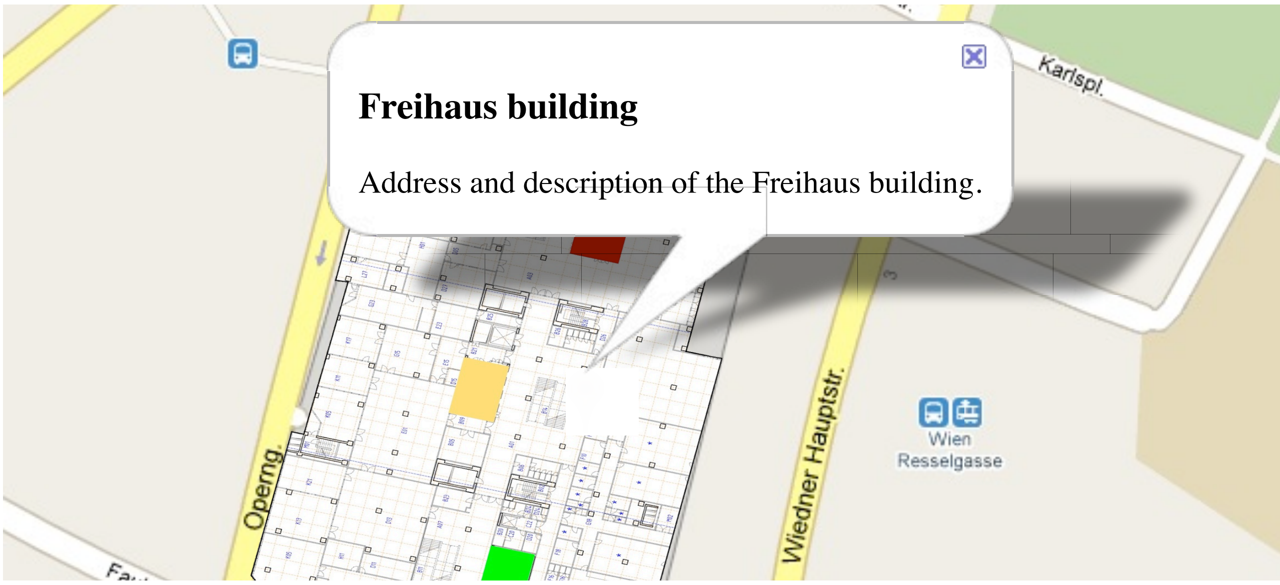
Google Maps and alternatives: The prototype uses the Google Maps API. The author is aware of alternatives to Google maps such as OpenStreetMap, but Google maps has been chosen because of its familiarity to the author and the (quite) well-documented public API. The version 3 of the Google Maps API is currently in active development and has only recently replaced the older version 2 (as of May 2010).

Sourcecode: A few lines of javascript code suffice to load an interactive street map. For the parsing of KML data, the Javascript library GeoXml3[5] was used. Newer versions of the Google Maps API natively support loading KML overlays.

The authoring of geographic data in kml: As mentioned above, the floorplans and placemarks have been defined in kml files, an XML dialect to describe geographic data. Google Earth can be used as a convenient authoring tool because in Google Earth, ground image overlays and placemarks can be exported to kml. The GeoXML3 script does not support rotating ground overlay images. A small software called WorldFileTool[6] was used to rotate the images so they don't have to be rotated in GeoXML3 anymore. Note that when using a more recent version of Google's API, a google server will parse the kml file, rotate the images, and return pre-rendered tiles to be laid over the street map.

[1] `http://www.uwosh.edu/map/campusmap.htm`, June 2010
[2] `http://code.google.com/apis/maps/documentation/v3/`, June 2010
[3] `http://code.google.com/apis/kml/`, June 2010
[4] `http://earth.google.com/`, June 2010
[5] `http://code.google.com/p/geoxml3/`, Feb 2010
[6] `http://gis.hsr.ch/wiki/Worldfiletool`, Feb 2010

a) Overlaying images over Google Maps



b) Overlaying vector objects over Google Maps

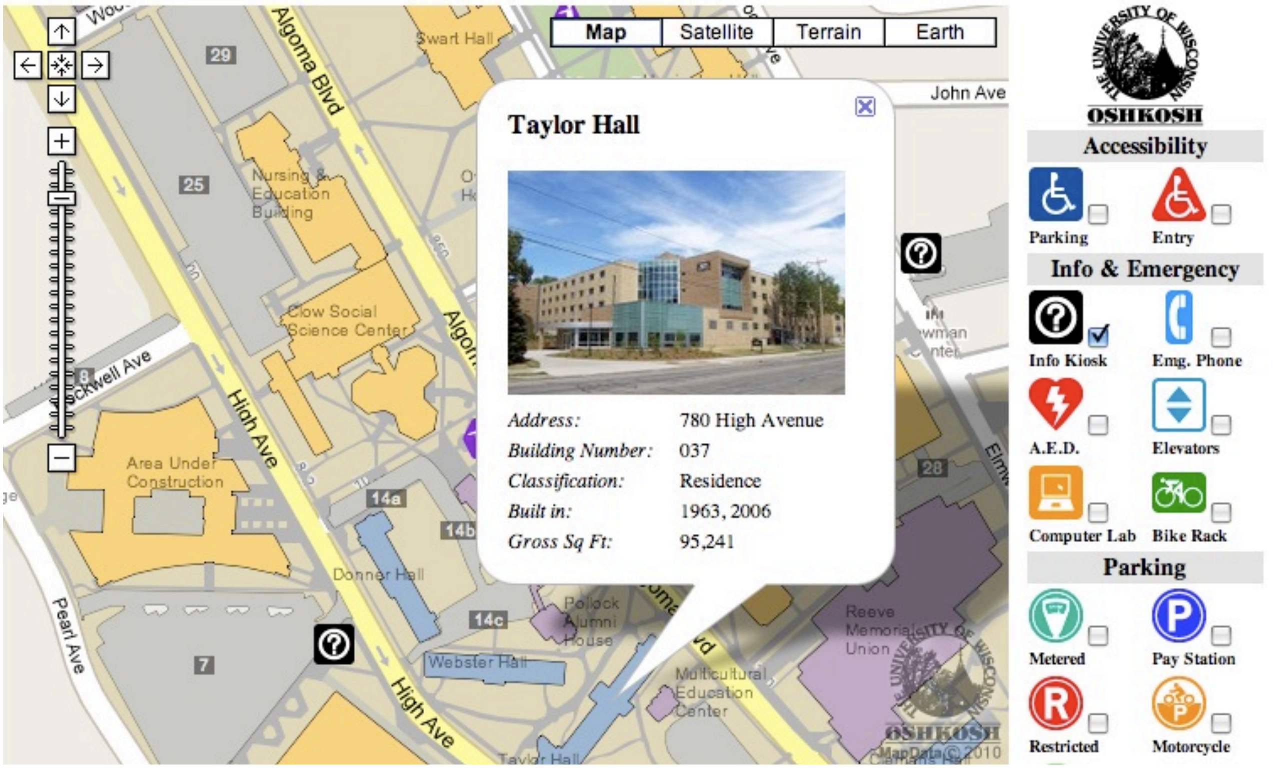


Figure 29: Technical feasibility prototype: (a) A prototype demonstrating the technical feasibility of overlaying floorplan images over Google's street map. (b) The possibility of overlaying vector objects is demonstrated by the University of Wisconsin Oshkosh's campus map.

5.4 Designing the hypertext – dimensional data analysis for a better understanding of the data structure

Viewing the domain of the envisioned wayfinding application as *multi-dimensional* data helped better understand the structure of the data and how it can be navigated. In the following, the problem and motivation for taking this approach are explained, and the dimensional analysis technique is then applied.

5.4.1 A motivation for dimensional data analysis

In the course of this work, while drawing wireframe sketches (see section 5.2, 'Wireframes', page 74 and figure 31, 'A proposed hypertext structure', page 85) to find out what a hypertext structure of buildings, floor levels and room types might possibly look like, the complexity of how the data can be navigated became apparent. The idea embodied in the wireframe sketch was that each navigation step down into the structure would 'drill down' and further constrain the resulting set of data: E.g. by choosing a building, only rooms within this building would be shown. Or by choosing a roomtype, only rooms with this type would be shown.

Since in the real world, buildings, floors and rooms have hierarchical, tree-like containment relationships, a tree-like data model seemed a good choice. But a tree-like data model encourages one single way of drilling down (namely along the tree-like structure of data), whereas the proposed hypertext structure should allow more freedom: The user should be able to constrain the resulting data as he/she desires at any place, in a similar manner as in typical 'OLAP' online analytical processing applications where the user can drill down (or up) along any dimension in any situation. E.g. the user should be able to first constrain the data by building and then by roomtype, or reverse the procedure as he/she wishes and first constrain by roomtype.

This freedom to drill up and down as the user wishes leads to the before-mentioned complexity: Restricting navigation to the tree-like data structure does not provide enough freedom, while simply allowing all constraints to be made at all times does not produce the desired result either: Some contraints (e.g. room type) can be made at any place within the data structure, but other contraints can only be made under certain conditions. For example, it only makes sense to choose the floor level after the building has been specified. Or, to name another example, only rooms with room type 'office' may be constrained by faculty, because lecture halls are not associated with any specific faculty.

One way to better understand the inter-dependence of constraints (and thus the availability of navigation steps that further constrain the resulting data) is by means of dimensional data analysis. This technique is employed in the requirements gathering phase for data warehouse and OLAP applications. A good introduction can be found in the 'Data warehousing fundamentals' textbook by Ponniah [Ponniah, 2001].

5.4.2 Data dimensions of the indoor wayfinding scenario

The dimensional data analysis technique shall be employed for the project at hand to analyze the kind of data we are dealing with, and to understand how the data can be navigated. The result of such an analysis can be presented as a table with one column per dimension, as can be seen in figure 30, 'Dimensional data analysis', page 84. For each dimension (i.e. in each of the table's columns), the hierarchies are listed starting with the biggest granularity. E.g. the dimension 'location' has a hierarchy of building, floor and room, and the dimension 'time' has a hierarchy of year, month and day.

Not all data dimensions identified are within the scope of the system. The time and event dimension, for example, are clearly out-of-scope because events (such as lectures or exams) will be managed using other applications. The integration with these other applications will not go beyond the possibility to link to rooms and buildings in the wayfinding system (see the following section 4.6, 'Integration with other systems', page 70 for more information). The organizational / personal dimension is of greater importance because it is closely related to the common wayfinding task of finding someone's (or some organization's) office.

As discussed in figure 30, 'Dimensional data analysis', page 84, not all possible ways of drilling down along a dimension make sense for a given a wayfinding task. For example, the possibility of first specifying the building, then the floor, and then the room's name is only useful for people who already know where the room is - these people either won't need the wayfinding application at all, or could have found the room quicker by simply entering the room's name into a search field.

Dimensional data analysis

1. Useful ways of navigating the data:

Note the mismatch with the tree model of real-world containment relationships, e.g. university → lecturehalls → by building.

university >
- lecture halls >
  - by building >
- other rooms >
  - by building >
- buildings >
  - lecture halls >
- faculties >
  - institutes >
  - teachers >
- teachers >
  - by faculty >

2. Some navigation paths are useless:

•/floor:1st
Useless from a data perspective: It does not make sense to specify the floor unless the building has been specified, because floor has a weak identity dependent upon the containing building.

•/building:mainbuilding/floor:1st/room:hall1
Useless from a usability perspective: If a user knows all of this information, he or she won't need further wayfinding instructions!

3. Dimensional data analysis:

Dimensions →	**Location**	**Room Type**	**Organization**	**Time**	**Event**
Hierarchies ↓	Building	Room Type	Faculty	Year	
	Floor		Institute	Month	
	Room		Teacher	Day	
			Information from the university's whitepages	*Time is neglected for the geographic aspects. Events (e.g. lectures) are stored in other systems such as TUWIS and possibly comprise several dimensions.*	

Figure 30: The real-world containment relationships (e.g. campus → building → lecturehall) form a tree model. But some useful ways of navigating and constraining the data (e.g. campus → lecturehalls → by building) mismatch with the tree model's structure (1), and simply allowing every sequence of constraints does not produce the desired result either (2). Looking at data in a multi-dimensional way by means of dimensions and hierarchies brings clarity into the situation (3): Each dimension can be constrained independently, but only along the dimension's hierarchy. When the hierarchy contains a weak entity (for example: floor), a constraint must first be set upon the parent, containing entity (in the example: building).

Proposed hypertext structure

Lecture halls >

By building >

[All building] >

[All lecture halls within the building]

[All lecture halls]

Conference rooms >

By Building >

[All buildings] >

[All conference rooms in the building]

[All conference rooms]

People >

Professors >

By faculty >

[All faculties] >

[All professors within the faculty]

[All professors]

Other persons >

Link to the Whitepages

Organizations >

Institutes >

By faculty >

[All faculties] >

[All institutes of the faculty]

[All institutes]

Other organizations >

[All other organizations]

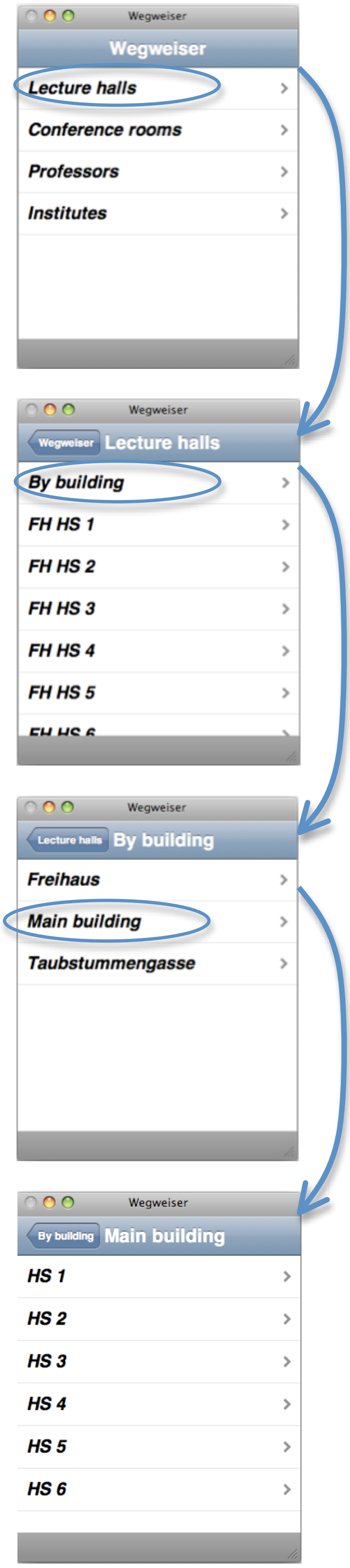


Figure 31: A proposed hypertext structure that allows the user to browse for a specific location. An exemplary implementation built as an iPhone application is shown in the right-hand column. To understand the data structure and rules that generate this hypertext, the dimensional data analysis technique was employed.

5.5 Map design – communicating directions

The process and the result of how floorplans were designed is shown in figure 32, 'The evolution of map Design', page 88. The proposed design is to use 2D floorplans that are laid over a regular, North-oriented street map. This approach has the following advantages and disadvantages:

– It cannot display several floor layers at once. This can be problematic for routes that need to be described in several floors. In these cases, different floor layers need to be shown within the same map.

– The map's North-orientation can mismatch with the orientation of the user's mental indoor maps.

+ It avoids the difficulty that some people have trying to read 3D plans. Verbal instructions are used instead to indicate the right floor level.

+ The map's North-orientation typically matches the orientation of the user's mental map of outdoor situations.

+ The streetmap is very helpful to people who do not know the campus.

+ The map can be zoomed and panned as usual for internet mapping software.

+ Apart from the sidebar that would need to be modified, any mobile client could use the application without further modification

Dealing with the 3D-aspect: As described above, the fact that 2D maps are used to communicate 3D routes can lead to problems. Whereas in most cases, displaying one single floor level will be sufficient, the university's main building for example is more complex: A typical route requires to first find the right staircase (i.e. a route at ground floor level), and then to proceed from there to the right room (i.e. a route at a higher floor level). In such cases, it is necessary to draw several floor plans in one graphic and to add textual annotations, as shown in figure 32, 'The evolution of map Design', page 88.

Amount of detail / level of abstraction: The plans are drawn in a very simplified and abstracted way, similar to the 'wegweiser' maps shown in section 8.1. This omission of unnecessary details is beneficial to wayfinding aids (see section 2.6, 'The necessary amount of detail in wayfinding aids', page 19 for more about this topic).

Façades and landmarks: Drawings of distinct façades can serve as landmarks and help orientation. The author first tried adding the façades directly unto the map as shown in figure 33, 'Façades', page 89. Usability testing showed that the perspective of these façade drawings was difficult to understand. Photos of buildings were added to the sidebar and to the info bubbles instead. Well-known cafés and info points have also been added as landmarks. (See section 2.8, page 25 for more theory about landmarks).

The design of map icons: It is difficult to communicate up/down directions (whether the user should go up or down the stairs) using staircase icons on a 2D map because the up/down direction conflicts with the North/South semantics of the rest of the map. The author therefore used the same generic staircase icons for routes that go up or down a staircase. Elevator icons could not be designed in a gender-neutral way because it made the users think of toilets. (See figure 34, 'Icon Design', page 90).

The difficulty with map design: During his work, the author found it very difficult to design good maps. The main reason for this certainly lies in his lack of experience: cartography and map design is a science of its own, but the author's background is computer science. A meeting with Prof. Gartner[1] provided helpful insights and confirmed the need for further collaboration. Despite the need for more professional map design, floorplans were designed and tested and worked sufficiently well.

Technical problems with interactive maps: Despite the fact that the Google Maps API[2] chosen for prototyping supports a lot of custom interaction, there is no simple way to author and render interactive maps:

- As for the authoring, some kind of authoring environment is required. In this work, Google Earth was used to edit KML files. The ability to edit both the overlay and the underlying map would be beneficial. Otherwise the overlays can happen to cut off underlying text, as can be seen in figure 33, 'Façades', page 89. Also, more control over the street maps would be useful: During this work, Google suddenly changed their maps by adding pseudo-3D building shapes to their maps, and the map overlays had to be adapted to the new situation (see figure 33, 'Façades', page 89).

- As for the rendering, different versions of the overlay are required for each zoom level because of font sizes and line widths. This works well with Google's KML rendering, but Google Earth's authoring tools are limited to rather simple KML shapes. The author used a vector graphic program to author more complex shapes. These shapes were converted to a raster image format, and did not scale lines and fonts nicely.

- As for the interaction, some of the map objects should react to mouse-click or mouse-over event, display popup info-bubbles, or interact with the applications's sidebar. This kind of custom interaction does not work with objects rendered using Google's KML overlays.

The choice of a software architecture that fulfills all of these requirements is left for future work. (See subsection 7.3.1, 'Requirements for a geographic software architecture', page 112 for a more detailed discussion of the requirements and possibilities).

[1] Prof. Georg Gartner is the head of the Cartographic Institute of the Vienna University of Technology. `http://cartography.tuwien.ac.at/`, June 2010
[2] `http://code.google.com/apis/maps/documentation/javascript/`, June 2010

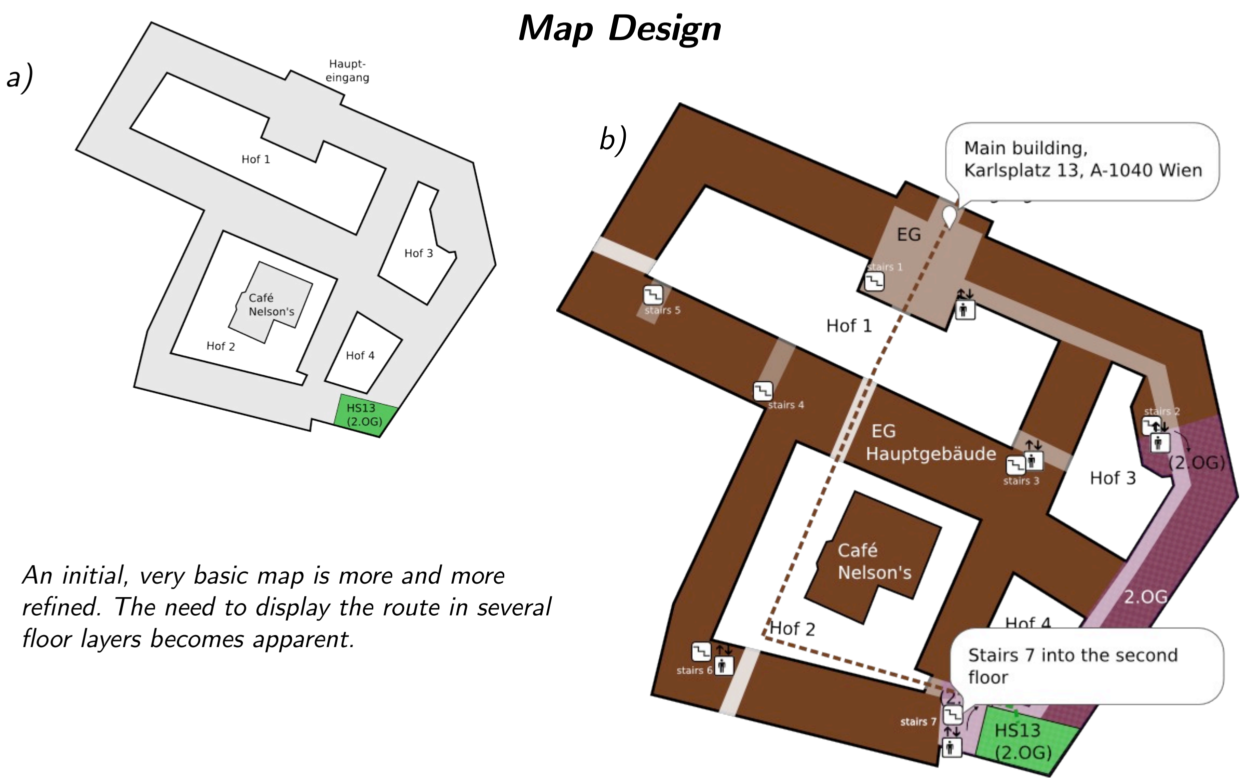


Figure 32: The evolution of map Design: (a) The process of designing maps started with an initial, very basic map. (b) During the development process, more and more features were added, and the need to display several floor layers in one map (here: ground floor and second floor) became apparent.

a) Façades as part of the overlays

b) Façades added by Google. Mismatch between map and floorplans.

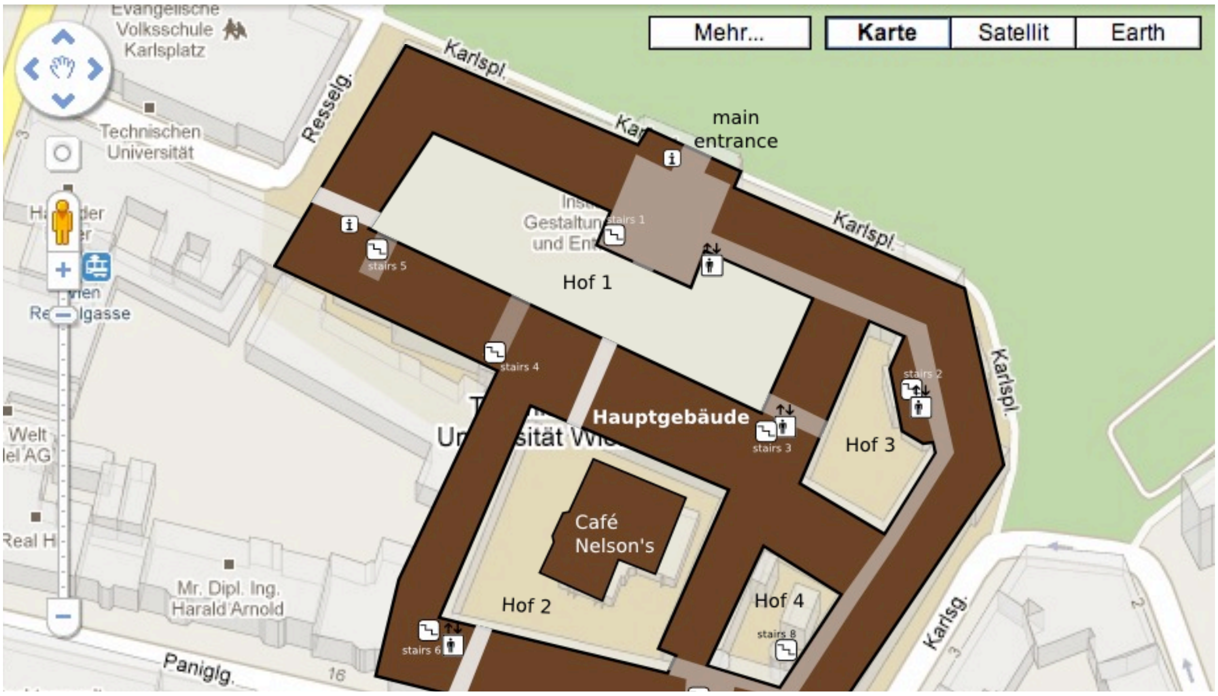
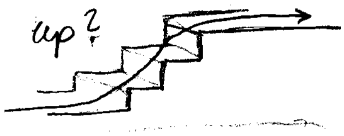


Figure 33: (a) Building façades and distinct building shapes can serve as landmarks. Adding the façades directly to the maps as shown in this figure was not easily understood by users because of perspective problems. (b) During the prototype's development, Google changed their maps in June 2010, adding very simple building shapes. Unfortunately, Google's shapes mismatch with the more detailed floorplans.

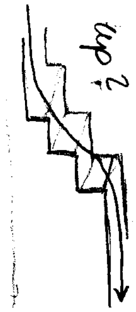
Icon Design

a) Staircase icons: *Portraying up and down in staircase icons on a 2D map is difficult:*

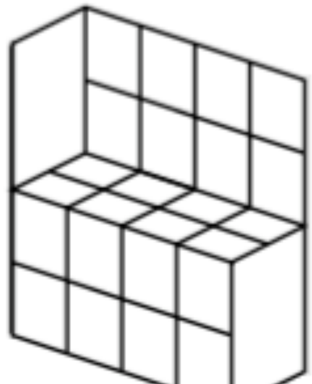
The simplest case: The path's direction matches the reading-direction. The stairs obviously leads up:



A more complex case: The path leads southwards The stairs are meant to lead up. This symbol does not work:

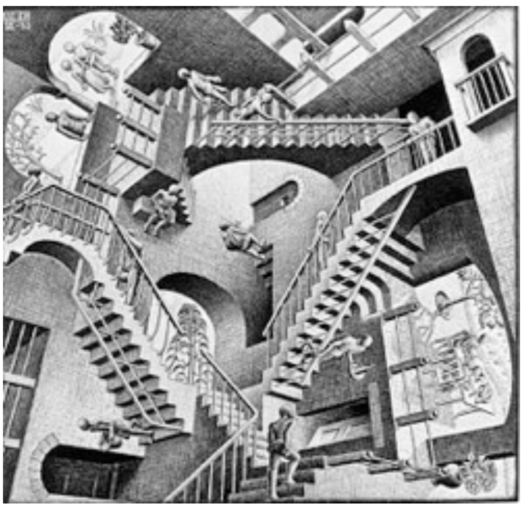


Making the stairs "look more 3D" does not help either because of the multistable perception effect:

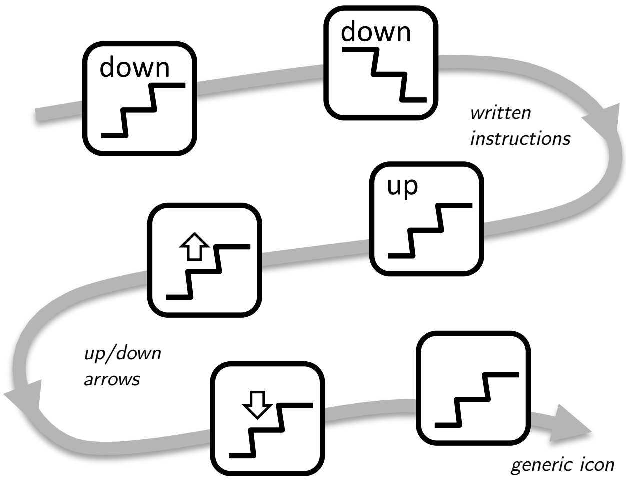


graphic from http://en.wikipedia.org/wiki/Multistable_perception

The problem is wonderfully illustrated by M.C. Escher's "Relativity" (Graphic from http://en.wikipedia.org/wiki/Relativity_(M._C._Escher)



Written instructions and arrows within the staircase icon are possible solutions, but up/down may still conflict with the route's direction:



...because of these problems, the author resigned from trying to indicate an up/down direction and uses a generic staircase icon.

b) Elevator icons:

The first elevator icon that the author tried was confused with a toilet symbol. One or three persons in the elevator and abandoning gender neutrality worked better:

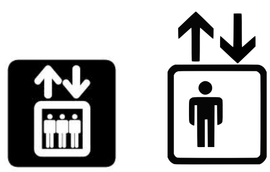


Figure 34: Icon Design: Staircase icons and why it is difficult to indicate a stairs up/down direction on a two-dimensional map. Elevator icons and why they should avoid gender neutrality.

5.6 The search function – dealing with abbreviations and numbers

The general design of the search function as proposed in this work is to list search results in the application's sidebar, and to display corresponding markers on the map, as shown in figure 35, 'Wireframe design for search results', page 92. If there is only one search result, it can directly be shown with more detail.

The search algorithm: A lot of work has gone into improving the search algorithm, i.e. how the search string is matched to search results. The motivation was to improve the existing situation, because judging from the interviews and from the author's own experience, the search algorithm implemented in the wegweiser.ac.at site[1] was not very convenient to use. The reason why is that the search did not deal with aliases and numbers properly. A future system should do better:

- *Aliases*: The search function should account for different naming conventions, aliases, abbreviations, and different use of whitespace: For example, one and the same lecture hall can have very different names such as 'FHHS8', 'FH HS 8', 'Freihaus HS 8', 'FH Hörsaal 8', or 'Nöbauer Hörsaal'. Put more generally: A room's name may include different versions of the building's name (FH, Freihaus), different versions of the room's type (HS, Hörsaal), different names for the same room (HS8 but also Nöbauer HS), and different spacing conventions.
- *Numbers*: The search function should not treat numbers and text the same way: For a purely textual search, it makes perfect sense to match each word of the search string to rooms by looking at whether the word is contained in the room's name. For numbers however, this algorithm would lead to strange results: The search string 'hs 1' would also match 'hs 13' and 'hs 21'. The search should therefore treat numbers differently, and only match numbers if every digit matches.

The search algorithm proposed in this work and implemented in the prototype can deal with aliases and numbers properly. It splits both the search string and the entities' names into tokens, and transforms each token into a canonical form. A search string matches if all of its tokens are found amongst an entity's tokens. The algorithm has been implemented in JavaScript. To make the printing less verbose, it is listed in pseudocode, see figure 36, 'The search algorithm's pseudocode', page 93.

[1] `http://wegweiser.ac.at` have so far been providing a wayfinding aid for Vienna University of Technology.

Figure 35: Wireframe design for showing search results: The results are listed in the sidebar, and corresponding markers are added to the map. If there is only one search result, it can be shown with more detail.

```
Search algorithm:  Given a search string, find and display all matching entities:

 1. Function tokenize:
    a) Expand abbreviations such as 'g.'  for 'Gasse' at the end of words (before
       the dot character '.'  is replaced by whitespace in the next step)
    b) Replace special characters by whitespace
    c) Separate numbers from characters, e.g.  hs3 becomes hs 3
    d) Split combined words, e.g.  FHHS becomes FH HS
    e) Remove multiple whitespace, trim leading and trailing whitespace
    f) Tokenize (Create an array of tokens by splitting the string at whitespace)
    g) Use canonical word forms (e.g.  Hörsaal becomes HS, Freihaus becomes FH)
 2. Tokenize each of the entities' names (for efficiency, do this only once and
    re-use the result)
 3. Tokenize the search string
 4. The matching function:
    a) If each search string token is found amongst the entity's tokens, then:
    b) Add the entity to the search results
 5. Display the search results:
    a) Create a sidebar entry for each search result
    b) Create a marker on the map for each search result
```

Figure 36: The search algorithm's pseudocode: Names are tokenized in a way that compensates different whitespace naming conventions. Each token is translated into a canonical form, thus effectively dealing with aliases and abbreviations. The whole tokens are then compared (using `==` operators, and not the `String.contains`-function), with the effect that numbers only match if every digit matches.

5.7 Designing interaction – the difficulty with rapid prototyping tools

It is important for a design project to use quick and cheap prototyping in order to explore and evaluate many different ideas. This is described in the theoretical part (see section 3.11, 'Wireframes, mockups and prototypes', page 47) and has been extensively put to practice in this work. One difficulty the author encountered in this context was how to prototype complex interaction:

Complexity because of the interaction style, e.g. zooming, panning, hovering: A lot of interface design can be prototyped quickly and cheaply using hand-drawn sketches or presentation software. E.g. For the panning of a map, a paper prototype may be a viable solution (e.g. see figure 25, 'Exemplary paper mockup', page 75), but it can never fully transport the feel of an electronic, interactive, pannable, and zoomable map. The same difficulty applies to complex interaction with selections and highlights: None of the before-mentioned, quick and cheap methods allow to experience quick interaction sequences such as 'I hover the mouse cursor over this text, then this marker is highlighted on the map. I hover around and then select the room HS 13, which at the same time opens up its infoWindow on the map'.

Complexity because of the amount of data: While complex interactions can to some extent be prototyped using simple methods, these methods come to their limit with bigger amounts of interactive objects (e.g. 15 buildings, 50 rooms). Prototyping the experience of scrolling through a list of fifty items and quickly clicking through a hierarchical structure that filters the list using different criteria will require some amount of automation.

The author's need to prototype complex interactions with many objects required better tools. Different prototyping tools were tried but none could cover every need:

- *Pencil and paper, handdrawn sketches*: These are, in the author's experience, the most efficient tools. This works well for screen designs and for prototyping interaction by telling a story (e.g. 'the user clicks on this button, then this screen opens up...'). The resulting drawing certainly allows to document an idea, but is very far away from the final experience of the envisioned software.

- *Graphics or presentation software*: It certainly takes more time to use graphics software than to draw things by hand, but the software allows a prototype to be more exact: Questions such as 'how much space will the sidebar really take?' can very well be answered using these tools. They have also been used extensively in this work to present wireframe sketches in a pleasant and easy-to-read way within this thesis. The drawback of graphics or presentation software is the poor support for interaction.

- *Paper prototyping*: In contrast to handdrawn sketches, the paper is used more physically in this method: It may be cut into pieces to form user interface elements, moved around to simulate the panning of a map, and layered to simulate overlapping windows. This method was used to prototype the interaction between map and sidebar. (See figure 25, 'Exemplary paper mockup', page 75). The drawback of paper prototyping has to do with the physical limits of papers: It cannot be zoomed, it could be scrolled by moving a piece of paper around, but this movement usually messes up the composition of all the other parts, and every single object state (higlighted, selected...) needs a separate piece of paper.

- *Adobe Flash*[1] allows complex interactions that can be programmed using the ActionScript programming language[2]. Despite prior experience in Flash Programming, the author found prototyping with Flash to be as much work as using HTML and JavaScript.
- *Microsoft SketchFlow*[3] is part of the 'Microsoft Expression Studio' software suite. The product is specifically meant for creating interactive prototypes. The author found the tool very convenient for rapid prototyping of simple interactions, but more complex interactions such as zooming and panning a map were either buggy or required programming and ended the quick and easy experience. In the author's opinion, products such as SketchFlow have a great potential.
- *HTML and JavaScript* was the solution that was finally used. The author found that JavaScript mapping API's such as the Google Maps API[4] used in the prototype allow for easy embedding of interactive maps. The JQuery[5] JavaScript framework allows for easy event handling and manipulation of the HTML document object model (this worked very well for the sidebar and the search function). With rising support of the HTML5 web-storage API[6] by all major browsers, client-side databases can be used to prototype with larger amounts of data.

Looking back, the choice for HTML and Javascript was certainly reasonable. It allowed to prototype complex interaction with many objects, and to integrate many aspects into one prototype, as required for this thesis. But despite the advantages and the successful application within this work, prototyping in HTML and Javascript sometimes rather felt like the tedious programming of a final product instead of the quick prototyping that was originally wanted. The author concludes with the recommendation to keep an eye on software products that are specifically made for prototyping rich interaction.

[1] `http://www.adobe.com/products/flash/`, May 2010
[2] `http://www.adobe.com/devnet/actionscript/`, May 2010
[3] `http://www.microsoft.com/Expression/products/Sketchflow_Overview.aspx`, May 2010
[4] `http://code.google.com/apis/maps/documentation/v3/`, May 2010
[5] `http://jquery.com/`, May 2010
[6] `http://www.w3.org/TR/webstorage/`, May 2010

5.8 Designing interaction – coordinating map and sidebar

This section describes the interaction between map and sidebar: When the user navigates the map, the sidebar's content should change accordingly, and vice versa. Put more generally, when the user interacts with any one of the four ways of accessing the system (i.e.: a link into the system, searching, navigating the map, or browsing hypertext), this will change the content of the four types of media (map, photographs, textual description, and the hypertext structure).

The sidebar entries' functions: The area on the left side of the map consists of 'sidebar entry' elements that are vertically stacked one above the other. These sidebar entries fulfill a set of different functions. Please note that these functions are not disjunct: E.g. sidebar entries representing a building are used for navigation but are also associated with a marker on the map. See figure 37, 'Sidebar entries', page 97 for an illustration.

- Sidebar entries for navigating the hypertext: going back, or drilling further in.
- Sidebar entries that function as page titles or headings.
- Sidebar entries that display information (e.g. a textual route description, a photo...)
- Sidebar entries that are associated with a marker on the map.

The sidebar entries' states: The sidebar entries can be highlighted and selected (seefigure 37, 'Sidebar entries', page 97). The interaction design to toggle between the states is as follows:

- *On mouse over: Highlight*: Highlighting occurs when the user moves the mouse over the sidebar entry, or when the user moves the mouse over the corresponding map marker.
- *On mouse out: Don't highlight*: When user moves the mouse away from the sidebar entry or marker, the entry loses its highlighted state.
- *On click: Select*: Only sidebar entries that are associated with a marker can be selected. Selection occurs when the user clicks on a sidebar entry or clicks on a marker, thereby opening the info bubble that belongs to the marker.
- *On info bubble close: Deselect*: When the user closes a marker's info bubble, the associated sidebar entry is deselected.
- *On click: Navigate*: Some sidebar entries are associated with a target URL. When the user clicks on a sidebar entry, the system navigates to this URL. Sidebar entries with a marker are first selected, and only navigate to the URL upon the second click.

Redundant navigation: Initially, only the hypertext structure could be navigated, and the map displayed corresponding contents. It was found through user testing that users expected to be able to navigate using the objects on the map as well, e.g. by following links within the info bubbles.

Figure 37: Sidebar entries: Sidebar entries may function as links for navigating the hypertext (marked with '<' or '>'), as page titles or headings (marked with bold letters), and as containers for the display text or photos. Their interaction states can be highlighted and selected. Sidebar entries may be associated with a marker on the map. The associated marker changes when the sidebar entry is highlighted. The marker's info bubble is shown when the sidebar entry is selected.

5.9 Usability evaluation

Usability testing has been performed for every major feature developed in the prototype:

Interviews: This work's interviews contained some elements of explorative usability evaluation: Interviewees were observed performing tasks, e.g. reading a map and describing a route, or drawing a route description as a map on a piece of paper. Since the interviews took place early on during the analysis phase (see section 4.2, page 60), only existing designs could be tested, e.g. the 'wegweiser' maps[1] that are currently used.

Early user testing: Early wireframes and mockups were tested by showing print-outs to friends and colleagues, asking them to carry out a small task. E.g. 'What do you see on this plan?', and 'How would you make your way to this lecture hall? Can you describe it to me?...'

Late user testing: Late design ideas were tested using assessment testing and the thinking-aloud method (see section 3.12, page 47): Users were assigned a specific task and were asked to speak their thoughts out loud while completing the task using a relatively detailed prototype.

The tests were rather informal, using friends or colleagues as test persons. The type of test was mostly explorative, looking for usability problems in a qualitative way (e.g. performance measures were not conducted). Some *interesting insights* gained from usability tests shall be listed in the following:

- Map design: Users generally understood the concept of overlaying a street map with a 2D floorplan. Adding semi-transparent building façades did not work well because of perspective problems. Adding photos to the sidebar and to the info bubbles worked better. (See figure 33, 'Façades', page 89)

- Problems with the geographic software stack: The overlays cut off some of the underlying text in the map. There can be a mismatch between Google's data and the very detailed real estate management plans. (See subsection 7.3.1, 'Requirements for a geographic software architecture', page 112).

- Info bubbles with redundant navigation links: The info bubbles that are shown on the map should include the same links as in the sidebar, because some users forgot about the sidebar and only looked at the map. Scrolling within the info bubble should be avoided if possible.

- Marker and sidebar entry interaction: Clicking the marker should toogle the info bubble's visibility instead of just showing the info bubble. (See section 5.8, 'Designing interaction between map and sidebar', page 96).

[1] `http://wegweiser.ac.at`, May 2010

- Icon design: A first version of elevator symbols looked like toilet signs. Staircase symbols on a 2D map cannot portrait up or down directions. Icons were too small in the beginning. (See figure 34, 'Icon Design', page 90).

- Software design: The hypertext structure worked very well. The need for a titlebar became apparent because one user was not aware she had already found what she was looking for.

- The search function: Users generally found what they were looking for. When searching for rooms, the search results should also include the building's name.

- The link-to-this-page function: The textfield containing the current URL should be able to show the entire URL.

Most of the insights gained from usability testing could directly be used for further design and development. The problems with the geographic software stack could not be resolved but had to be left for future work. The final prototype is described in the next chapter.

6 Presentation phase

This section corresponds to the last phase of the development process used in this work. In Mayhew's original development process (see section 3.6, 'The usability engineering lifecycle', page 36), this phase is titled 'installation phase' and refers to the installation of a finished product – in the case of this work however, the finished product is a prototype that needs to be presented and described instead of installed and shipped. For this reason, the largest part of this chapter describes the prototype's design and not its installation.

6.1 Users and use cases

Users: The primary user categories are people who are familiar with or new to the university's campus. Other distinctions are: Teachers versus students versus visitors, knowledge of academic terms, computer skills, language skills, experience with other wayfinding systems, frequency of use, and disabilities. (See section 4.3, 'Personas', page 65 in the analysis phase of this work).

Use cases: The prototype implements the following use cases: To search for a place by going directly to the wayfinding website, or alternatively: to follow a link from another system that leads to a wayfinding description. To read and understand the wayfinding instructions. To write down or print wayfinding instructions. To share wayfinding instructions with others by sharing a link to the wayfinding application. (See section 4.4, 'Use cases', page 67 in the analysis phase of this work)

6.2 Interaction and content

The data offered by the system: The prototype offers information about the university's buildings and rooms. Information about people (and their office locations) has not been implemented in the prototype.

- *The vast amount of data*: Providing wayfinding information for every building, lecture hall, conference room, and every teacher's office means the system has to deal with a very large amount of data (about 17 buildings, more than 200 lecture halls or conference rooms, several hundred offices). The prototype only features a few exemplary buildings and lecture halls.
- *Important data entities*: The prototype offers the following data: The main entities are buildings and rooms. For each entity, a name, an address, and a textual route description is given. A standard streetmap as base layer for the map. Floorplan overlays. Routes for every entity to be added to the floorplan overlay. The geographic position of every entity.
- *Data model*: The prototype uses a very simple relational data model consisting of two tables only, one for buildings and one for rooms.

Redundancy in interaction and content: The wayfinding system features redundant information and interaction possibilities as shown in figure 39, 'The prototype's general layout', page 106. This redundancy is positive and necessary, as described by literature (see section 2.5, 'The medium used for wayfinding aids', page 17) and confirmed through usability testing (see section 5.9, 'Usability evaluation', page 98).

- *Interaction: Four ways of accessing the system*: Hyperlinks from other systems, textual search, navigation on the map, and browsing in a hypertext structure. These different ways of access provide partly redundant ways of navigating within the system.
- *Content: Four types of media*: Redundancy is also added through the inclusion of multiple media with overlapping content: Maps, pictures, textual descriptions, and a hypertext structure have been used to communicate wayfinding information.
- *Combining ways of accessing the system and the different media types into one coherent layout*: The prototype features a split screen with a sidebar on the left and a large map on the right side. The sidebar contains the search function, the hypertext structure, textual wayfinding instructions, and photos. The map contains streets and buildings as well as floorplans. The info bubbles on the map have a similar content and functionality as the hypertext structure in the sidebar. When the user navigates in the sidebar, the map's content changes accordingly, and vice versa.

The map: The prototype features an interactive street map with 2D floor plan overlays. See section 5.5, page 86 for the process of map design, and figure 40, 'The prototype's map design', page 107 for a screenshot of the prototype's maps.

- *2D zoomable maps and floorplan overlays*: The way the prototype solves the problem of communicating 3D routes on a 2D display is this: Instead of showing several floor layers in a 3D perspective, the system shows only one floor layer that is laid over an interactive, zoomable, 2D streetmap. In some cases and for complex routes, different floor layers need to be shown in the same map (see figure 40).
- *Map orientation*: The prototype's maps are North-oriented. The choice of map orientation is generally difficult: Outdoor maps and street maps are best North-oriented, but indoor maps are best oriented in the direction of the route. Providing two maps with different orientation is confusing for many people, judging from the interviews (see subsection 4.2.3, 'Conclusions drawn from the interviews', page 63). Being able to rotate the map would be helpful, but may be technically difficult.
- *Route versus survey knowledge*: Both route and survey knowledge are provided by the prototype. Route knowledge is provided by the textual instructions and by the route drawn on the map. Survey knowledge is provided by the map.

- *Landmarks*: Photos are used in the prototype to communicate what a building's façade looks like. The prototype's maps include landmarks such as cafés and stores. A study to find a set of popular landmarks around the university's campus is left for future work.
- *Map navigation*: The map features the following interaction possibilities: It can be panned and zoomed using drag and scroll mouse actions. Doubleclicking the map re-centers the map. Pushpin-style markers on the map represent interactive objects. Clicking on a marker opens an info bubble that offers the same navigational options as the sidebar.
- *Map design*: A lot of map abstraction has been applied to architectural plans in order to create simple floorplans that better suit the needs of a wayfinding aid. The color scheme was chosen to match the colors used in Google Maps, with light colors used for passages and dark colors used for inaccessible areas. Texture was added for the colorblind. Icons are used to mark staircases, elevators, and info points. Map markers are used for interactive objects.
- *Map authoring*: The author used the vector graphics program Inkscape[1] for drawing the floorplans. The resulting graphics have been geo-referenced using the Google Earth[2] desktop software and saved using the KML[3] data format. Once the KML files are publicly accessible through HTTP, they can be rendered using the Google Maps Javascript API[4].

The hypertext structure:

- *Content: Combining text and photos*: The hypertext combines textual wayfinding descriptions and photos into a hierarchical structure of pages. Search results are also shown as a hypertext page.
- *Interaction: Drilling down and filtering data*: The hypertext structure allows the user to navigate the data structure of buildings and rooms by 'drilling down' and filtering the data in a similar way to OLAP (online analytical processing) applications.
- *Coordination with the map*: When the hypertext content in the sidebar changes, the map's content changes accordingly. Some of the hypertext's elements are associated with a marker on the map. Highlighting the hypertext element also highlights the marker, and selecting the hypertext element selects the marker and shows the associated info bubble on the map.

The search function: The search function is designed to allow a wide range of different naming conventions for buildings and rooms. Search results are displayed in the sidebar and in the map.

[1] `http://www.inkscape.org/`, June 2010
[2] `http://earth.google.com/`, June 2010
[3] `http://code.google.com/apis/kml/`, June 2010
[4] `http://code.google.com/apis/maps/documentation/v3/`, June 2010

- *Search algorithm*: The search algorithm can deal with aliases, abbreviations, spacing, numbers and different naming conventions in a smart way. (See section 5.6, 'The search function', page 91 for a more detailed description)
- *Search results*: The list of search results is shown in the sidebar. Each search result is also associated with a marker on the map. If there is only one search result, there is no need to display a list, but the system directly jumps to the according entity.

Saving and sharing wayfinding information: The prototype allows to save and share wayfinding information by copying a URL and by printing wayfinding instructions.

- *Sharing links*: The user can copy the URL from the browser's address field or from a textfield in a small dropdown menu.
- *Printing wayfinding instructions*: The ability to print the page is foreseen but not fully implemented because of technical difficulties.

6.3 Technical aspects

Web application: The prototype is built as a web application, i.e. it needs an internet connection to work, it is built using web technologies such as HTML and Javascript, and it is meant to be used with desktop or laptop computers.

Software architecture: The technical architecture needed to implement a final system that runs on the university's servers has been left for future work. Nevertheless, the necessity to design and evaluate user interaction required programming and a software architecture suited for rapid prototyping:

- The prototype should be *easy to build and easy to change*, allowing for big changes and making it possible to start again.
- The prototype required the ability to *specify user interaction* in a detailed manner. This requirement cannot be covered by static wireframe sketches, but needs some kind of programming environment.
- The prototype required the ability to *test with real data*. E.g. the number of buildings and lecture halls available in the prototype should closely match the real situation - it makes a big difference to the user whether he/she has to scroll through 15 or 50 items in a list!

The software architecture that was used in the prototype is shown in figure 38, 'The prototype's software architecture'. It has the following characteristics:

- *No application server*: All of the prototype's application logic is written in Javascript and executed within the client's webbrowser. Even the database resides on the client side.

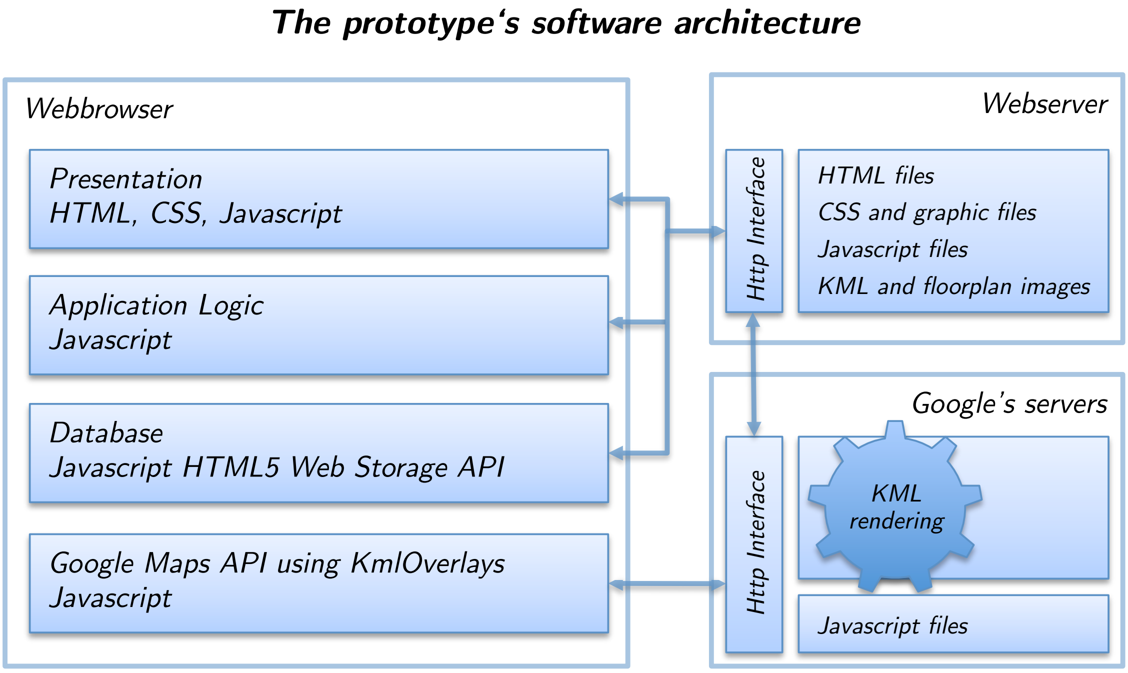


Figure 38: The prototype's software architecture: All application logic is on the client, even the database. Google's servers are used to render KML floorplan overlays.

- *Web technologies*: Html and Javascript were chosen to create the prototype. The author had tried different products aimed at interactive media creation but finally settled with HTML and Javascript because of his prior experience and because of difficulties with the other products (see section 5.7, 'Designing interaction with rapid prototyping tools', page 94). The specific software products, standards and technologies used are HTML, CSS, Javascript, Ajax, JQuery, Html 5 web storage API, KML and the Google Maps Javascript API.
- *Client-side database*: A popular client-side database is the Google Gears[1] browser plugin, but its development is likely to be discontinued[2] by Google in favor of the HTML5 web-storage API[3], which is already supported by most modern browsers (including Internet Explorer, Firefox, Safari and Opera). On top of the web-storage API, the prototype uses a small SQL engine called DOM-SQL[4] that provides a client-side SQL database that works in most modern browsers without the need for a special plugin. DOM-SQL did not seem ready for production use but was well-suited for prototyping because it allowed for easy deployment and testing without the need to setup a special database server.

[1] `http://gears.google.com/`, May 2010
[2] `http://googleenterprise.blogspot.com/2010/04/laying-foundation-for-new-google-docs.html`, May 2010
[3] `http://www.w3.org/TR/webstorage/`, May 2010
[4] `http://dom-storage-query-language.googlecode.com/`, May 2010

- *Map rendering*: The prototype makes heavy use of Google's javascript mapping APIs. The map's base layer is Google's streetmap. Overlays are drawn using the Inkscape vector graphics program. The resulting graphics are geo-referenced using the Google Earth desktop software, exported to the KML format, and rendered on the map using Google's javascript API.
- *Javascript, Html and CSS bundled as components*: The sidebar's entries (i.e. hypertext elements shown in the sidebar on the left side of the map) feature rather complex interaction. The corresponding Html, Css and Javascript code has been factored out into a component using the JQuery UI Widget framework.

The prototype's installation: To install and run the prototype, make its files accessible through a webserver and open the index.html file in a modern webbrowser. Both the javascript and the kml files contain absolute URLs that need to be changed to match the webserver's server name.

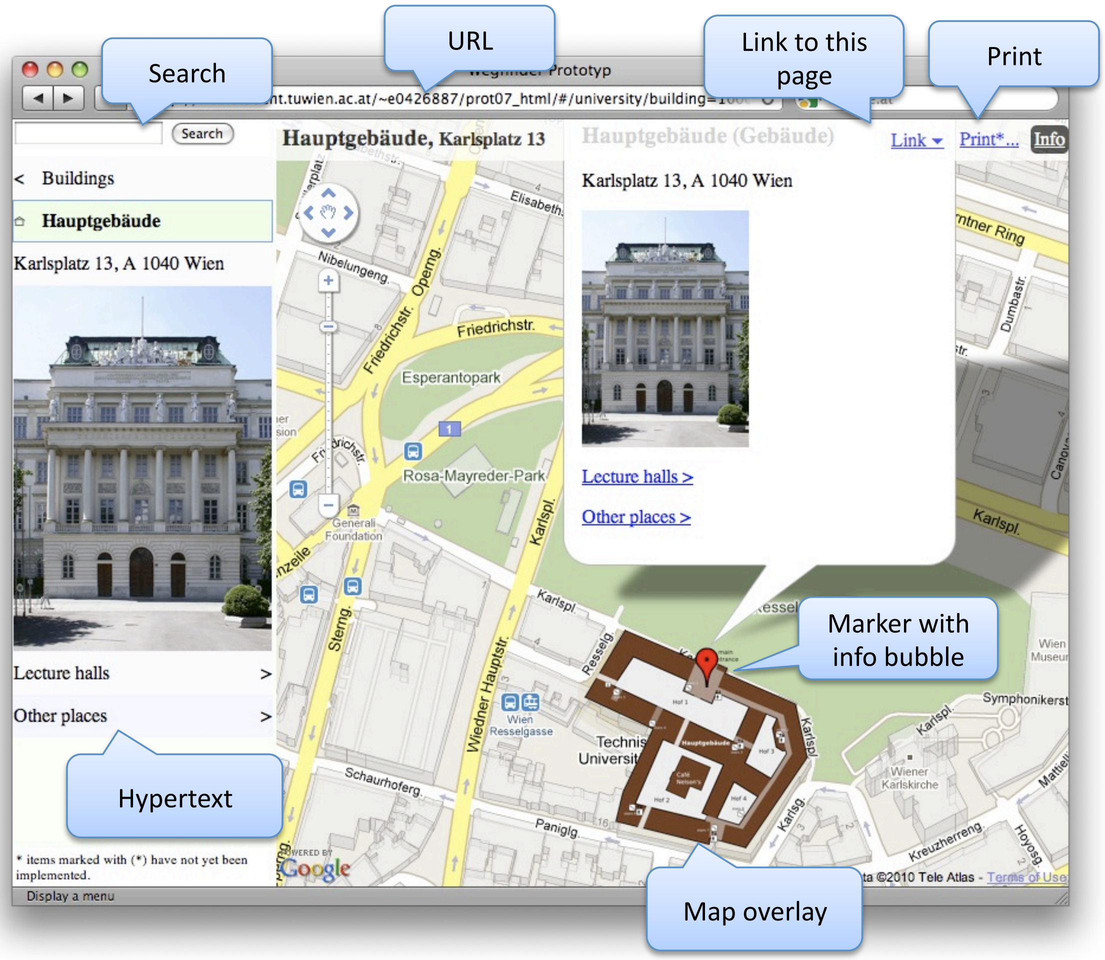
6.4 Future options

Mobile wayfinding: Although outside the original scope of the thesis, the prototype can be adapted for use with mobile devices such as PDAs and modern cell phones. Suggestions for how this can be accomplished are presented in the design phase, section 5.2, 'Wireframes', page 74.

- *Suited for mobile wayfinding*: The prototype's design can be adapted for mobile wayfinding. Instead of a split screen with sidebar and map next to each other, a mobile application should either show the sidebar's content or the map.
- *Suited for location based services*: Since standard 2D-maps are used, it is easy to display the user's location on the map.
- *Suited for rotating map and floorplans*: Because only 2D maps are used in the protoype, they could easily rotate into the user's current direction of view. This is not possible with the prototype's implementation because Google Maps currently lacks this feature.

Integration with other systems: The prototype is open for integration with other systems: Any system can offer a hyperlink to a place description in the wayfinding aid, and links to other systems can be included into the prototype. Further integration had to be left for future work, e.g. integration with a common database about buildings and rooms, or integration with other systems for automatic map generation.

The prototype's general layout



Four ways of accessing the system:

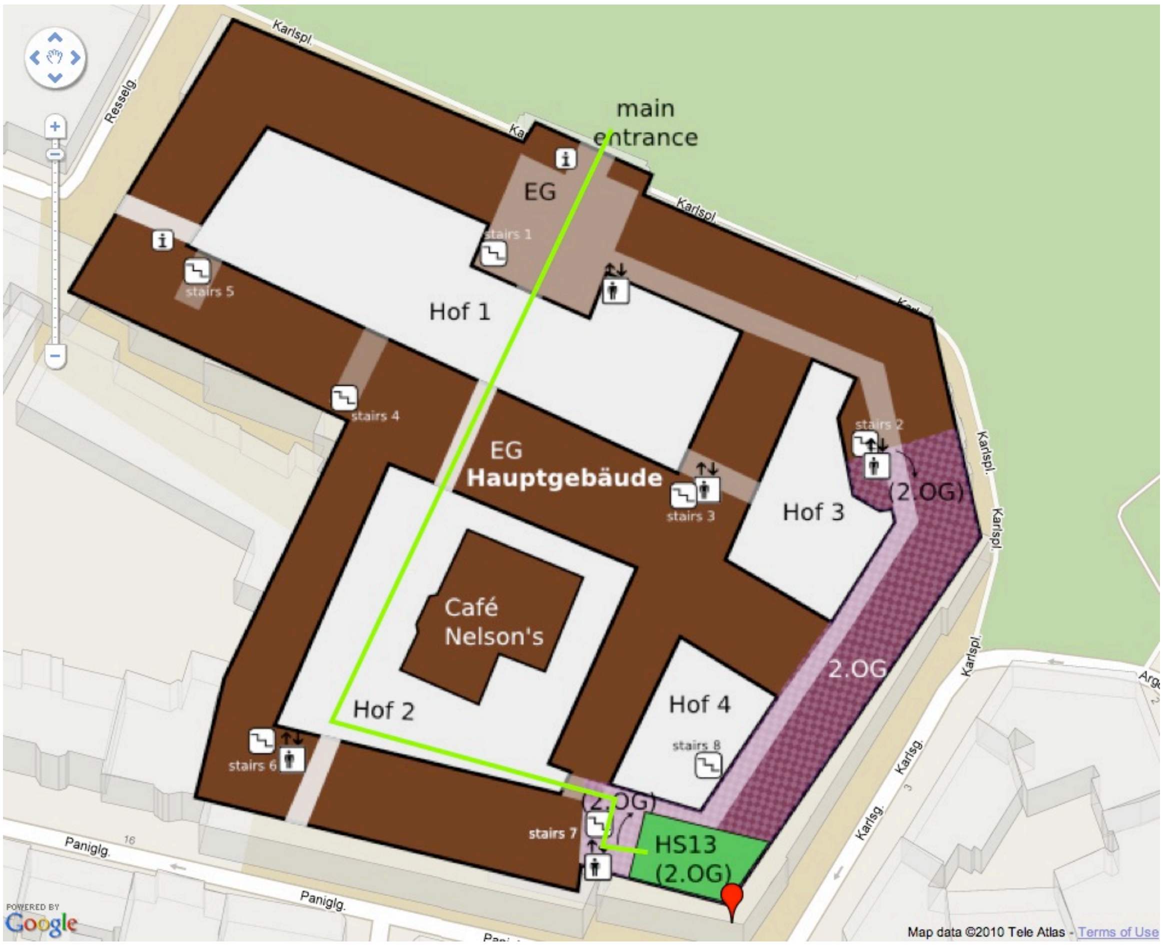
- URLs and Hyperlinks
- Textual search
- Navigation on the map
- Browsing a hypertext structure

Four types of media

- Map
- Pictures
- Textual descriptions
- Hypertext structure

Figure 39: The prototype's general layout.

Map design

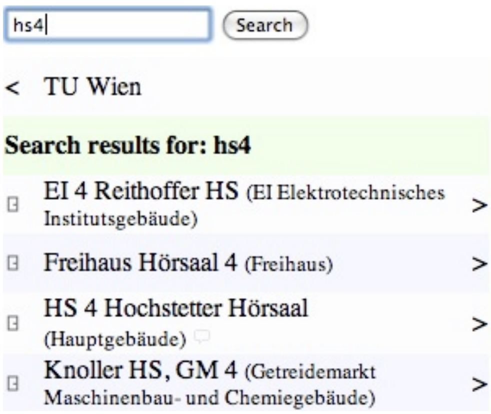


Characteristics of the prototype's map design:

- 2D-floorplans laid over a streetmap
- North orientation for both indoor and outdoor situations
- Map navigation using drag and scroll actions as usual for internet mapping software
- A lot of abstraction has been applied to simplify architectural plans into floorplans suitable as wayfinding aid
- The maps have been authored using Inkscape and Google Earth

Figure 40: The prototype's map design: Dealing with three-dimensional routes displayed on a 2D screen. Other aspects include orientation, navigation within the map, map abstraction and map authoring.

The search function



The search algorithm features a clever way of dealing with aliases, abbreviations, spacing, numbers and different naming conventions of rooms and buildings.

Search results are listed within the hypertext structure and shown as markers on the map.

The 'link to this page' function

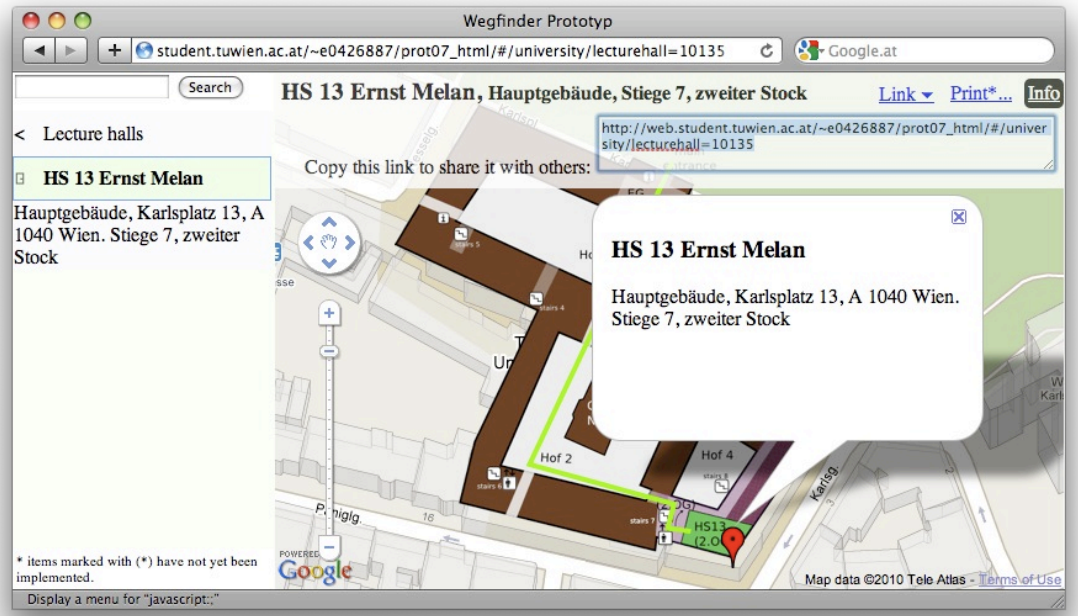


Figure 41: The prototype's search and 'link to this page' functions.

7 Conclusions and future work

This section sums up the results of this work, reflects on the methods that were used, and provides directions for future work.

7.1 Results

The results of this work meet up with the purpose and scope formulated in the introduction: Wayfinding theory and usability engineering methods were used to create a prototype for a new online wayfinding aid for the Vienna University of Technology's campus (see chapter 6 for a description of the prototype):

Chapters two and three contain the theoretical part of this work: Wayfinding theory is elaborated in chapter 2 and usability engineering methods are presented in chapter 3. Chapters four to six contain the practical part of this work, where a prototype for the online wayfinding aid was constructed. These three chapters reflect the three phases of the software development process that was used: The examination of existing solutions, users, tasks and data form the analysis phase presented in chapter 4. Iterative design, prototyping, implementation and evaluation make up the design phase in chapter 5. The resulting prototype's characteristics are presented in chapter 6.

The methods used in this work proved to be effective and efficient in most cases:

- *Usability engineering*: The iterative, systematic, user-centered approach to software development provided a helpful framework to structure and plan this work. Using 'discount' methods helped gain good results with a reasonable amount of time and effort.
- *User involvement*: Involving the user through interviews and usability testing gave helpful insights throughout the whole development process.
- *Personas and storyboards*: The author found personas and storyboards to be a pleasant and creative way of formulating user and task characteristics and the envisioned context of use. The benefit of personas and storyboards for communicating these facts to other team members was less relevant in the case of this work because the author worked on his own most of the time.
- *Prototyping*: Prototyping was extensively put to practice and allowed to explore different ideas before investing too much work into the implementation. Paper prototypes and wireframes made using presentation software worked well. The use of web technologies to prototype more complex interactions proved to be effective, but better rapid prototyping tools would be helpful.
- *Web-based mapping techniques*: Using web-based mapping software allowed to prototype complex interactions and to integrate different aspects (e.g. hypertext, photos, maps, search, links) into one prototype. The Google Maps Javascript API provided an easy, well-documented software

stack, but limited the prototype to features available through Google's API. Other geographic software architectures can be chosen for the final system. The options are presented in this chapter, but the choice is left for future work.

- *Usability evaluation*: User testing using the thinking-aloud method as recommended by Nielsen's 'discount usability engineering' approach [Nielsen, 1993] worked very well for the purpose of this work.

The limitations of this work: A lot of future work relates to the fact that the scope of work is limited by the theory and the methods that were used and by the fact that a prototype instead of a final system was to be constructed:

- The prototype is limited in that it does not include maps and floorplans for every building and room and in that it is only installed on a development computer and does not need to be running on the university's servers.
- The prototype's focus was upon usability. The software architecture chosen is suited for the exploration and evaluation of design ideas, but a different software architecture should be chosen for the final system.
- This work is further limited by what theory and methods were used: Wayfinding theory and usability engineering methods are this work's foundations. The economic view (e.g. buying decisions or cost estimates) is not included. Nor does this work contain the organizational aspect of how to plan and manage the future wayfinding software project.

A quick overview of recommendations for a follow-up project is given in the executive summary at the beginning of this work and is further elaborated in the following section. Future work from a scientific perspective is outlined at the end of this work in section 7.4, page 118.

7.2 Difficulties and lessons learned

This section sums up difficulties encountered during this work. This may be particularly interesting for a follow-up project.

Limited access to users: During the analysis phase of this work, the author was eager to make interviews with a wide range of people, including teachers and university staff. The author found these people's availability to be limited, but managed to arrange two interviews (see section 4.2, 'Interviews', page 60). Later on during the design phase, the author resigned from testing with teachers, but tested with two categories of users only, those familiar or unfamiliar with the university's campus (see section 5.9, 'Usability evaluation', page 98).

Lack of suited tools for rapid prototyping: The author had wished to find a software tool for the rapid prototyping of complex interactions (e.g. an interactive map) with large amounts of data (e.g. around 50 lecture halls). None of the tools that were tried fulfilled all needs. The author resorted to programming the prototype using HTML and JavaScript. (See section 5.7, 'Designing interaction with rapid prototyping tools', page 94)

Lack of cartographic knowledge: Making good maps is a science of its own. The floorplans featured in the prototype worked sufficiently well, despite the author's lack of prior cartographic experience. Co-operating with more experienced people, e.g. from the Cartographic Institute of the Vienna University of Technology, would very likely benefit the project.

Open questions concerning map orientation and the 3D aspect: There is no simple answer to the question how to orient a mixed indoor and outdoor map and how to present the three-dimensional floor levels on a two-dimensional computer display (see section 2.3, 'The orientation of maps', page 15 and section 2.4, 'Indoor wayfinding', page 16). The proposed solution of using 2D floorplans that are laid over a regular, North-oriented street map (see section 5.5, 'Map design', page 86) works well judging from usability tests, but further evaluation and research is needed (as recommended in section 7.4, 'Future work, scientifically', page 118).

Technical difficulties with mapping software: This work's approach to combine a streetmap with two-dimensional floorplan overlays led to problems: In some cases, the overlay cut off underlying text, in other cases there was a mismatch between the overlay and the less detailed street map. At one moment, the mapping provider changed the street map's look, which required all overlays to be adjusted. Different approaches to overcoming these difficulties are discussed in a detailed way in the following section 7.3, 'Future work, technically'.

7.3 Future work, technically

This section describes the future work necessary for a follow-up project that implements the university's campus wayfinding aid.

Organizational aspects: Define the follow-up project's organizational structures necessary for the wayfinding aid's implementation within the university's institutions, practices and regulations. Define the project's key dimensions time, money and resources.

Considering mobile wayfinding and local signage: The scope of this thesis was limited to online wayfinding aids, but mobile wayfinding and local signage (using physical signs or digital devices) could also be considered for the final system.

Integration with other systems: This includes integration through hyperlinks from and to other systems, data integration for shared information about buildings and rooms and persons (e.g. integrating with the university's whitepages application), and integration with the real estate management's system to allow for automatic map generation. (See section 4.6, 'Integration with other systems', page 70)

Accessibility: This comprises web accessibility, acessibility of the route descriptions and of the building itself. (See section 3.2, 'Accessibility', page 30). The author recommends to seek cooperation with Prof. Zagler from the Institute of Integrated Study at Vienna University of Technology[1].

Cartography: Cartographic knowhow is needed for more professional map design and to examine what landmarks should best be featured on the maps. The author recommends to seek help for the technical questions and to try to cooperate with Prof. Gartner from the Cartographic Institute at the Vienna University of Technology[2].

Geographic software: The choice of software architecture and software products has big consequences for the wayfinding project. More details concerning this point are given in the rest of this section.

7.3.1 Requirements for a geographic software architecture

Since implementing every part of a geographic software project from scratch is too much work for most projects, existing products must be chosen. This choice will often determine the resulting software architecture because the software products are meant to work together in a specific way. A group of software products that together implement a geographic software system for collecting, storing and sharing geographic information are referred to as GIS infrastructures, spatial data infrastructures, geographic software stacks, or geostacks [Haklay et al., 2008; Turner, 2006].

The prototype built in this work makes heavy use of Google's geostack: Maps are authored in KML using Google Earth, rendered on Google's servers, and displayed using the Google Maps Javascript API in the client's webbrowser. (The prototype's architecture is described at length in chapter 6, 'Presentation phase', page 100). This software architecture was a reasonable choice for the prototype because it allowed to try different designs without too much effort. But it also has some disadvantages, and the final system should consider other options as well.

The following list contains different requirements for choosing a geographic software architecture and a stack of software products:

Authoring - the need for a good authoring environment: Some kind of authoring environment is needed because copy-pasting latitude-longitude-values by hand proved to be a pain even for a small prototype

[1] `http://www.is.tuwien.ac.at/`, July 2010
[2] `http://cartography.tuwien.ac.at`, July 2010

with only a few buildings and rooms. In this work, the Google Earth Desktop software[1] was used to author KML data, which is rendered by Google's servers using the KmlOverlay object provided by Google's API. In other setups, it may be advisable to use a more professional GIS or architectural software stack.

Rendering - the need for separately rendered zoom levels: Interaction with the proposed map includes zooming. The map tiles of zoomable maps are usually rendered separately for each zoom level. The reason is that objects would otherwise quickly be too big to be convenient or too small to see: This applies to font sizes, icons, and line thickness. The conclusion is that the floorplan overlays need to be rendered in different zoom levels as well.

Three-dimensional data: Because indoor routes can span different floor levels, the software architecture must provide a way of dealing with three-dimensional data. The approach taken in the prototype is to represent the different floors using map layers: Floorplan layers are laid over a regular street map. The following point formulates requirements for the rendering of the map and its overlays.

Rendering - the need to render the streetmap together with the overlay: While adding overlays to an existing map is easy and works well in many situations, it may also lead to problems: E.g. when a floorplan is laid over a street map, it may happen that building's shape shown on the streetmap includes less detail and therefore does not exactly match the floorplan (this is certainly the case with Google's street maps which for the moment being[2] do not include detailed building shapes). Another problem is that the overlay may partly hide text labels of the underlying street map. A third problem is that the company providing the street map may from one day to another change the maps - then the overlays don't fit anymore (this did happen during development!). The only solution to these mismatches is to render the streetmap together with the overlay.

Rendering and interaction - the need for rotatable maps: Being able to rotate the maps is technically difficult but would benefit the users. Three directions are of particular interest for map rotation: North orientation for outdoor maps. Route orientation for indoor maps. And the user's view direction as orientation for maps on mobile devices. If the maps are to be rotated, the geostack should be able to render the maps accordingly (otherwise, the text would be rotated as well which would make it hard to read).

Interaction - the need to make some map objects interactive: Although all of the rendering work could be performed on a server, some knowledge about individual map objects still needs to be transported to the client for the user to interact with them. E.g. when using Google's KmlOverlay object, Google's

[1] `http://earth.google.com/`, June 2010
[2] July 2010

server returns a snippet of Javascript code that adds the overlay tiles (rendered by Google and loaded from Google's servers) but also enables the interaction with objects shown on the tiles.

Rendering - the need for internationalization: If possible, the geostack should be able to render the maps in different languages.

Dynamic routing: The prototype that results from this work does not dynamically calculate routes. A wayfinding system with dynamic routes needs a more complex representation of floorplans: Not just drawings, but a topological model of interconnected rooms. Topological models of indoor space are discussed in [Kuipers et al., 2003; Remolina and Kuipers, 2004; Li and Lee, 2008]. If the envisioned campus wayfinding system is to provide dynamic routes, the chosen geographic software stack should have this feature.

Location based services for mobile wayfinding: Interviews showed that students are interested in mobile wayfinding. This was true even when they lacked suited mobile devices. If the campus wayfinding aid is to be used in mobile situations, it should be aware of the user's current location. The geographic software stack should therefore allow for location-based services. Software architectures for location based services have also been examined at Vienna University of Technology [Gruber, 2010; Simon, 2008].

Integration with signage systems: The maps provided by the wayfinding system could also be used for signage within the university's buildings. This may include physical signage, digital signage on non-interactive screen, and digital signage on interactive information kiosks.

7.3.2 Implications of different software architectures

This section presents different software architectures and their implications. The focus of the following considerations is upon map rendering and whether it should take place at the client or server side. The way that floorplan overlays and the underlying streetmap are combined is also considered.

Drawing maps by hand: Drawing maps or map overlays by hand is a simple but viable solution that has partly been used in the prototype resulting from this work. It has the following advantages and disadvantages:

+ Graphical possibilities: Since any drawing software can be used for map authoring, the graphical possibilities are unlimited.

+/- Effort: This is an easy solution that does not need a lot of effort to setup. Maintaining a few hundred hand-drawn maps may however prove difficult.

- Interaction: Since the maps and overlays are only drawings, they cannot feature interaction.

- Rendering of zoom levels: Zooming in and out of the drawing will lessen the maps readability because fonts and line widths will be too big to be convenient or too small to see.
- Mismatch between map and overlay: If the drawings are used as map overlays there may be a mismatch with the underlying map.

Parsing and rendering map overlays in the browser: Another possibility is to parse and render geographic data on the client-side in the web browser. One example is the University of Wisconsin's campus map[1] that loads an xml file and generates vector overlays on the map using custom Javascript. Another similar approach is the GeoXML3 parser project[2], and OpenLayer's[3] vector layers implement the same software architecture. The advantages and disadvantages of this approach are:

+ Rendering of different zoom levels: Since the map overlay is rendered in the client, the rendering can be adjusted to suit different zoom levels. E.g. font sizes and line thickness can be adjusted to ensure good visibility at all zoom levels.
+ Interaction: Because the map overlay's objects are rendered in the browser, it is relatively easy to make them interactive. E.g. clicking on a building may show an info bubble, and hovering the mouse over a staircase icon may show a little tooltip.

+/- Effort: Writing a custom client-side parser for geographic data is a lot of effort. The OpenLayers vector layer implementation can parse and render data on the client.

- Mismatch between overlay and map: Because the overlay and the map are rendered separately, the map overlay may conflict with the underlying map. E.g. an overlay may cut off text labels of the underlying map.
- Limited graphical possibilities: The graphical representation of objects rendered on the client side is rather limited. Modern browser's support for SVG[4], the HTML canvas element[5], the WebGL standard[6], and for CSS-based transformations[7] are pushing the limits of what can be accomplished on the client side.

Rendering map and overlays separately on the server: Map overlays may be rendered on a server. The pre-rendered tiles are then added as an overlay to the map on the client side. This is the architecture chosen by Google's implementation of KmlOverlays and used in this work's prototype. OpenLayers's raster layers use the same architecture.

[1] `http://www.uwosh.edu/map/campusmap.htm`
[2] `http://code.google.com/p/geoxml3/`, Feb 2010
[3] `http://www.openlayers.com/`, July 2010
[4] `http://www.w3.org/Graphics/SVG/`, July 2010
[5] `http://www.whatwg.org/specs/web-apps/current-work/multipage/the-canvas-element.html`, July 2010
[6] `http://www.khronos.org/webgl/`, July 2010
[7] `http://www.w3.org/TR/2009/WD-css3-2d-transforms-20091201/`, July 2010

+ Rendering of different zoom levels: Google's servers render different versions for each zoom level. This means that font-sizes and line widths can keep good visibility no matter what zoom level. The same effect can be accomplished using another rendering server, e.g. a server with a Web Map Service[1] interface that allows to load the overlay as an OpenLayers layer.

+/- Effort: It is very easy to use Google's KmlOverlays because Google's servers take care of the rendering for you. Alternatively, another rendering server can be set up, but this requires the installation and configuration of a whole stack of software products.

+/- Interaction: Google's Kml Overlays allow for complex interaction with the features rendered on a map overlay: In addition to the map tile graphics for the overlay, Google's servers also return Javascript code to enable interaction with the overlay's features. The same effect can be accomplished in OpenLayers by adding a transparent vector layer with a 'SelectFeatureControl'[2] to make some map regions interactive.

\- Mismatch between overlay and map: As with the previous option described above, the fact that the overlay and the map are rendered separately can cause problems.

Rendering map and overlays together at the server: This option requires the ability to manipulate the map and to render custom tiles. This can be accomplished by licensing map data and mapping software or by using free data and tools e.g. from the OpenStreetMap project[3].

+ No mismatch between overlay and map: Because the overlay (e.g. the floorplan) is rendered together with the map, there is no mismatch between the two: The floorplan's data may correct inaccurate building shapes where necessary, and add objects to the map's different layers. The problem of map overlays cutting off text labels of the underlying map can thereby be avoided.

+ Rendering: Since custom rendering is used for every part of the map, everything can be styled to fit the zoom level and map orientation that is needed.

+/- Interaction: Information about interactive map regions needs to be sent to the client for this purpose. This can be accomplished using a transparent vector layer as described above.

\- Effort: Custom map rendering is a lot of effort and requires setting up a whole stack of geographic software components.

The above list shows that the last option (rendering map and overlays together on the server) has many advantages. The author recommends a technical feasibility prototype to see if this option can be realized with reasonable effort. One possible way to start would be to download the OpenStreetMap

[1] `http://www.opengeospatial.org/standards/wms`, July 2010
[2] `http://openlayers.org/dev/examples/select-feature.html`, July 2010
[3] `http://www.openstreetmap.org/`, July 2010

data and load it into PostGis[1] using the OsmInABox[2] project. This has the disadvantage that updating OpenStreetMap data is difficult, but the advantage that the PostGis database can then be modified using standard GIS software to add floorplan data. GeoServer[3] can then be used to serve a raster layer and a transparent vector layer for interaction with map features to the OpenLayers[4] web client.

7.3.3 A survey of existing software products

The complexity and the effort associated with creating an online wayfinding system has become apparent from the above considerations. The scope of work is further widened if mobile wayfinding, local signage and electronic information kiosks are to be integrated with the wayfinding system.

The large amount of work necessary for the creation of the university's wayfinding aid makes it advisable from an economic perspective to consider buying off-the-shelf software, or to try to sell the product to others if a custom product needs to be built. The author recommends starting with a survey of existing solutions from a buyer's perspective. (Note that the survey included in section 4.1, page 53 of this work is different in that it takes a usability perspective). The future survey should consider the following points:

- Price and effort for the software's implementation, installation and maintenance.
- The above list of requirements for a software architecture with the corresponding implications.
- Standards compliance and the ability to integrate with other systems.
- Integrated solutions for indoor wayfinding. These may include but should not be limited to Abysseo[5], here2there[6], zetek[7], micello[8], and Gebäudenavigation[9]. A more comprehensive list of products is needed.
- Popular stacks of geographic software such as Esri[10], OpenStreetMap[11] and its variants for indoor wayfinding[12] and increased standards compliance[13], OpenGeo[14], GeoServer[15], UMN Map

[1] `http://postgis.refractions.net/`, July 2010
[2] `http://dev.ifs.hsr.ch/osminabox/`, July 2010
[3] `http://geoserver.org/`, July 2010
[4] `http://www.openlayers.com/`, July 2010
[5] `http://www.absysseo.com/`, July 2010
[6] `http://www.here2theresoftware.com/`, July 2010
[7] `http://www.zetek.com/solutions_wayfinding.php`, July 2010
[8] `http://micello.net/`, July 2010
[9] `http://www.gebaeudenavigation.de`, July 2010
[10] `http://www.esri.com/`, July 2010
[11] `http://www.openstreetmap.org/`, July 2010
[12] `http://geoparden.wordpress.com/`, July 2010
[13] `http://dev.ifs.hsr.ch/osminabox/`, July 2010
[14] `http://workshops.opengeo.org/stack-intro/`, July 2010
[15] `http://geoserver.org/`, July 2010

Server[1], Deegree[2], OpenTripPlanner[3], and the Google Maps API Family[4]. Attempts of adding a third spatial dimension exist within the Postgis[5], OGC[6], CityGML[7], OpenFloorPlan[8], deegree[9] and Geoserver[10] communities.

7.4 Future work, scientifically

This work revealed two topics of primary interest for the envisioned wayfinding aid that need further scientific investigation:

Communicating 3D route descriptions: Indoor routes typically contain the three-dimensional aspect of different floor levels. Communicating these 3D routes using a 2D medium is difficult. The interviews performed in this work showed that many users disliked 3D representations of the different floor levels and partly had problems interpreting them correctly. Using 2D floorplan overlays over a regular street map is proposed as a solution to the problem. An empirical study and usability evaluation is needed to better prove the usefulness of this approach.

Map orientation: The best orientation of maps with indoor route descriptions is an open question. North-orientation best suits the user's mental map for outdoor situations. Using the route's direction or the direction of a building's entrance best suits the user's mental indoor map. Mobile wayfinding applications may also orient the map in the user's current view direction. The conflict between these different map orientations is not easily resolved. The provision of several maps with different orientations (e.g. one North-oriented street map and a route-oriented indoor map) was problematic because users had difficulties matching maps with different orientations. Another approach is to provide interactive, rotatable maps. A survey of different approaches to the problem of map orientation and an evaluation of their usefulness is needed to guide the design of indoor wayfinding systems.

[1] http://mapserver.org/, July 2010
[2] http://www.deegree.org/, July 2010
[3] http://opentripplanner.org/, July 2010
[4] http://code.google.com/apis/maps/index.html, July 2010
[5] http://postgis3d.blogspot.com/, July 2010
[6] http://www.ogcnetwork.net/node/624, July 2010
[7] http://www.citygml.org/, July 2010
[8] http://www.cmu.edu/silicon-valley/research/floor-plan/index.html, July 2010
[9] http://wiki.deegree.org/deegreeWiki/deegree3/deegree3D/, July 2010
[10] http://geoserver.org/display/GEOS/Multidimensional+WCS, July 2010

8 Appendix: Evaluated wayfinding systems

This appendix is dedicated to a more detailed description of each one of the wayfinding aids that have been chosen for evaluation in section 4.1. Each description is structured in the same way, mimicking the structure of the list of evaluation criteria in subsection 4.1.1. Taken together, the systems described in this section reveal a space of possible future designs.

8.1 The wegweiser.ac.at website

The website "wegweiser.ac.at"[1], also available in English[2], provides online campus maps for most Austrian universities, but it also features other information such as a list of studies offered by each Austrian university.

In the past time, the Vienna University of Technology has been relying upon this website (which is maintained by an external service provider 'unikat'[3]) in order to provide an online wayfinding aid. In the course of the University's TISS project[4], the campus maps - amongst many other systems - shall be integrated into one homogeneous set of applications. The current work of evaluating existing wayfinding systems is situated in the context of these efforts.

In the following, the wegweiser.ac.at wayfinding aid will be described using the evaluation criteria identified in subsection 4.1.1.

Medium: From a technical point of view, the wegweiser.ac.at website is an online web application (i.e. it needs an interconnection to work and it is built using web technology). From a geographical point of view, the medium used by the wegweiser.ac.at website are maps and very short verbal instructions (including the address and name of the building, the staircase or escalator number, the floor and the room number).

Level of map abstraction and schematization: The maps offered by the wegweiser.ac.at website are highly simplified as can be seen in figure 42c and d: Only the most important streets bear names, and only buildings belonging to the Vienna University of Technology are shown. The floor plan as well is kept very simple: Doors (except for the buildings entrances), windows and detailed contours are not shown.

Route versus survey knowledge: Some survey knowledge is provided by the system, not very detailed but just enough to locate a room or building within its environment. The simplified street graph as shown in figure 42c places a building into the context of its surrounding streets. Through the inclusion of street names for a building's entrances as shown in figure 42d, this context is preserved for the floor plan as well. Although the system does not dynamically calculate routes for the user,

[1] `http://www.wegweiser.ac.at/`, Sept 2009
[2] `http://www.studyguide.at`, Sept 2009
[3] `http://www.unikat.at/`, Sept 2009
[4] `https://tiss.tuwien.ac.at/`, Sept 2009

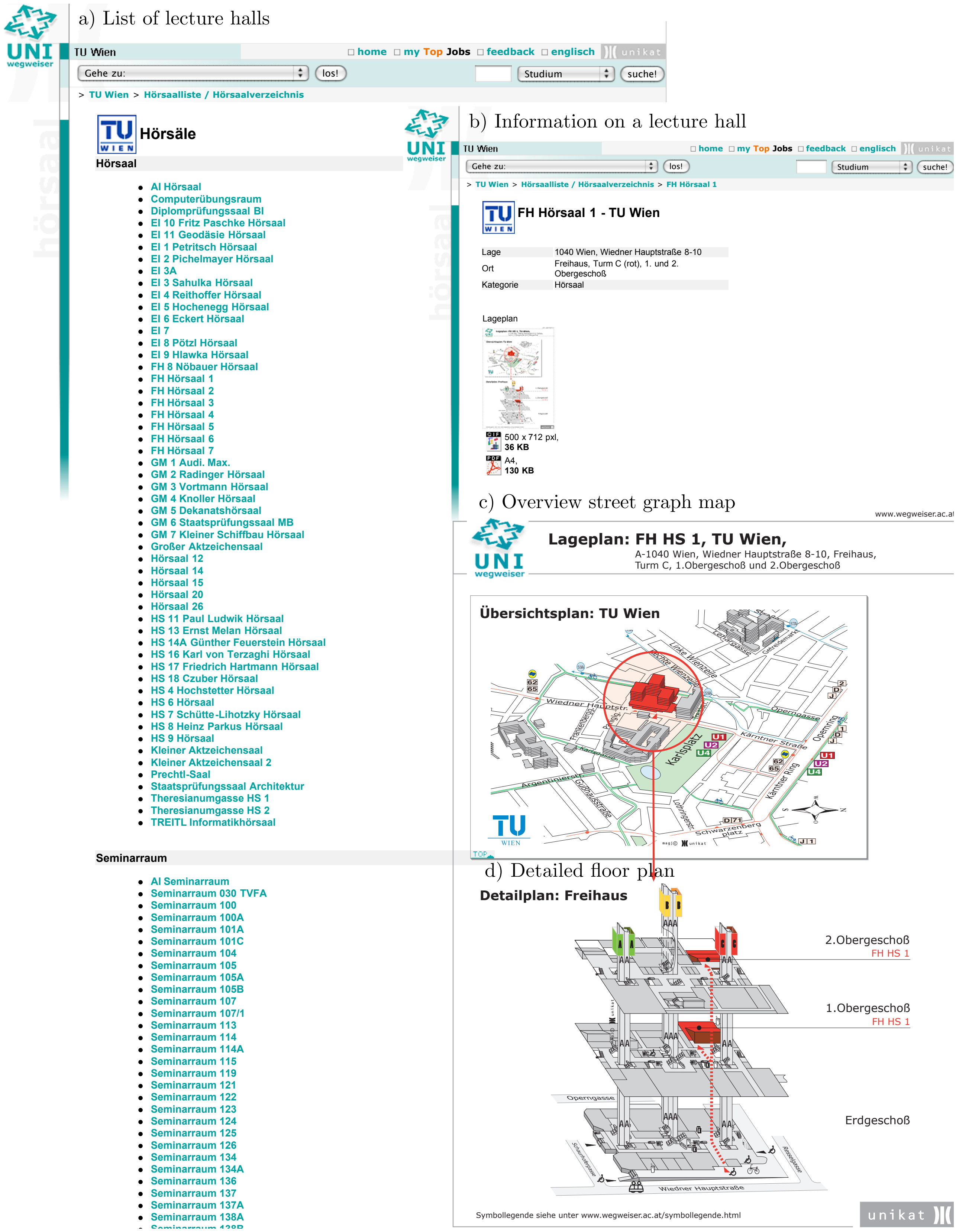


Figure 42: The www.wegweiser.ac.at website

it does provide some route knowledge: On the detailed floor plan (figure 42d), red dots lead from the building's entrance to the elevator and from there to the room in question. Also, the textual description (figure 42b and c) of a room does provide some kind of route information: A user is able to use this information step by step, e.g. by looking for the right building “Freihaus” first, then for the right tower “Turm C” within this building, then the right floor “1. Obergeschoß” and so on.

Landmarks: The maps offered by the wegweiser.ac.at system do not include objects just for the sake of their use as landmark. However, some of the objects that are included on the maps such as escalators, entrances or the dining hall are also useful as a landmark.

Map orientation: None of the overview street maps (as seen in figure 42c) are North-oriented, but the same westward orientation is used for all of the overview maps. The reason for this is not apparent to the author. The detailed floor plans however (as seen in figure 42d) each use a different orientation: the detailed maps show the same perspective as seen when entering the building through the main entrance.

Indoor space: As shown in figure 42d, the maps display indoor space in an interesting 3D perspective including several floor layers. The red dotted path that indicates a route on the map also promotes a specific indoor wayfinding strategy: First choose a suited entrance, then go to the nearest elevator and get to the right floor, and then go to the right place within that floor.

Search style: As shown in figure 42a, the wegweiser.ac.at website can be accessed through both a menu structure and a textual search allowing to find a lecture hall by name. Through the menu structure, the user first selects “TU Wien” as his university, and then follows a link to get to the list of lecture halls that is shown in figure 42a. Instead of reading through the whole page, the webbrowser's built-in search function may then be used to find the appropriate room within the list of lecture halls.

The textual search offered by the website will in most cases speed up the process because it will narrow down the results. The search algorithm is however rather too tolerant, e.g. for the keyword ”hs1” it will not only include rooms named “Hörsaal 1”, “FH Hörsaal 1” or “HS 1”, but also “Hörsaal 12” or “HS 18”. A more intelligent search algorithm would make more precise matches and would thus make searching more efficient for the user.

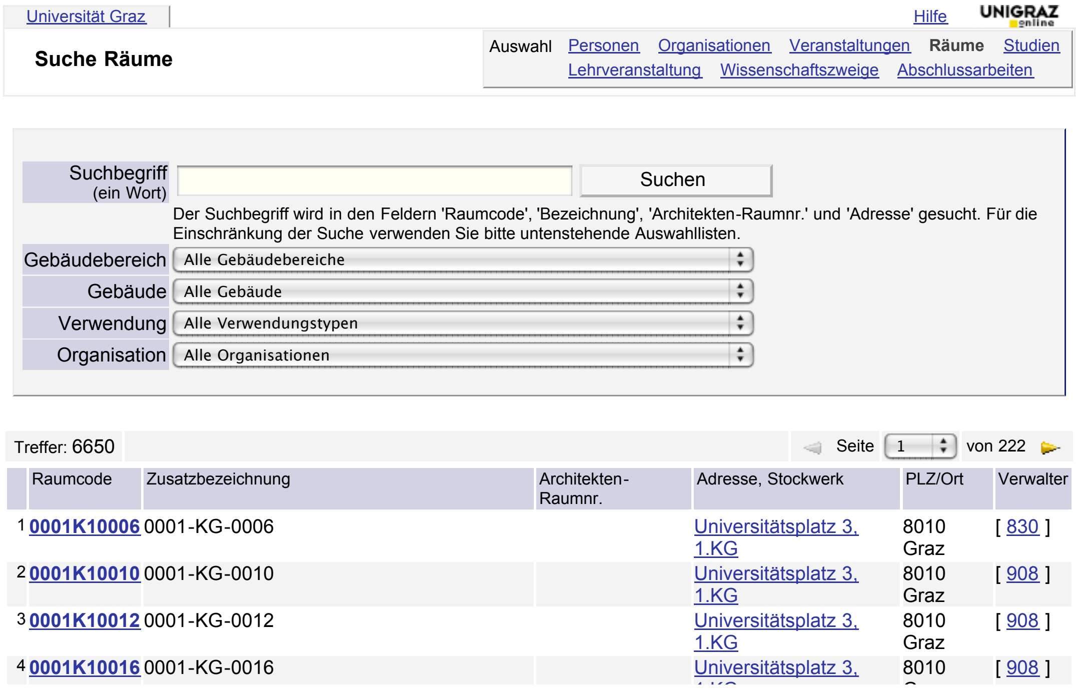
Navigation style: The maps offered by wegweiser.ac.at are static images. Thus, once the user has navigated through the website's hypertext and now displays a location's map, no further navigation (e.g. by panning and zooming the map or by following hyperlinks) is possible.

Internet technology: The website is made of HTML with some basic JavaScript functionality. The maps can be displayed and downloaded both as GIF images and as PDF files for higher quality.

8.2 University of Graz

The wayfinding system offered by the University of Graz, Austria (see figure 43) provides much more than just wayfinding information. The system can show maps in different levels of detail and of

a) Searching for rooms



b) Displaying a room on the floor plan

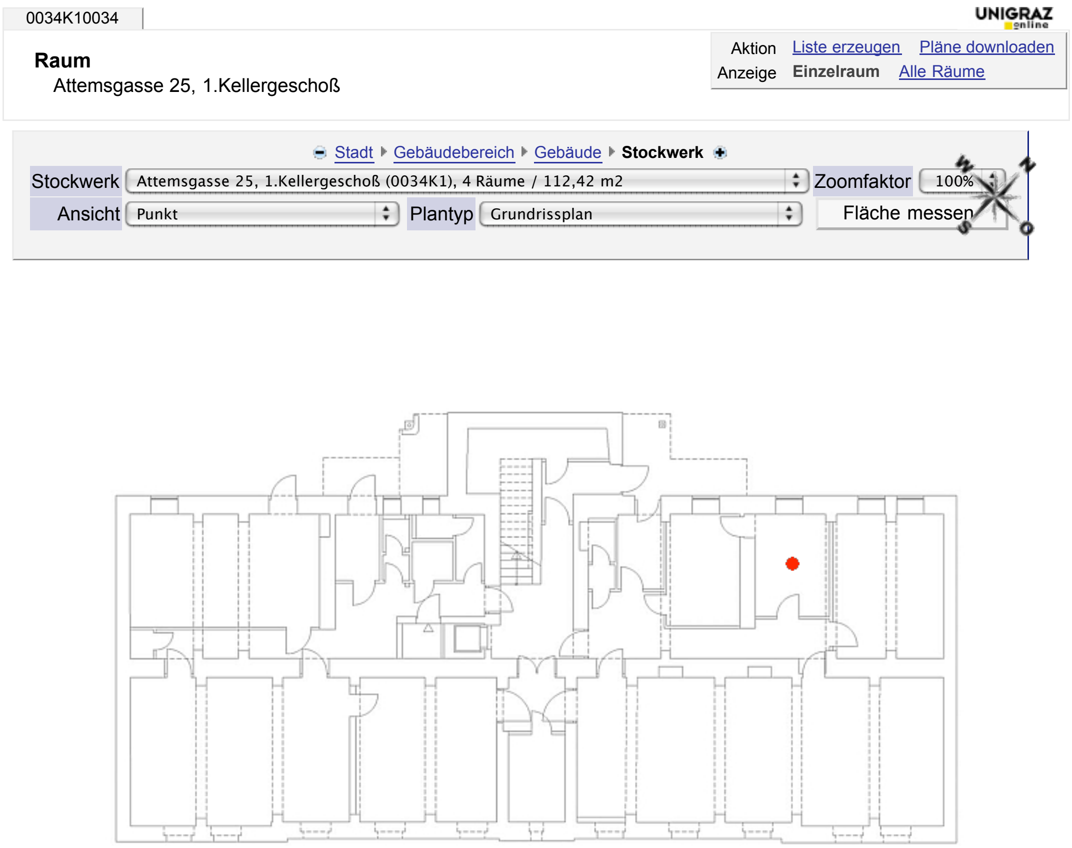


Figure 43: The online application offered by the University of Graz

different types (a 'normal' floor plan and two types of fire escape plans). And for each room, it can display the room number, the organizational unit who is primarily using the room, a schedule of lectures or other events booked for this room, the cleaning schedule, and the room's size. The system seems to be primarily made for a real-estate management type of usecase. As a first-time user with a simple question in mind (how do I find a specific hall?), the author found the amount of information and the many options overwhelming.

Medium: A web application using maps to communicate wayfinding information.

Level of map abstraction and schematization: The wayfinding system of the university of Graz offers maps in different levels of detail: The most coarse level is just one simplified street map with highlighted university buildings provided as a static image. The next level consists of one map per building and shows the building with the streets around it. Unfortunately, these maps have abstracted away a lot of things that would be useful for wayfinding (street names, landmarks, names of buildings). Instead, they show detailed contours not only of the building in question but also of the buildings around it. The finest level consists of a set of almost 300 floor plans such as the one shown in figure 43b. As described above, these plans exist in different versions and a lot of additional information can be laid over these plans.

As a note of interest, the Vienna University of Technology is in possession of similar plans (see figure 44) to those used for the wayfinding aid at the university in Graz. Those plans are primarily used for real estate management, but some of the overview plans (similar to those shown in figure 44a) have also been used in the university's "whitepages" phonebook web application[1].

Route versus survey knowledge: The system does not provide route information, but provides survey knowledge instead: Neither are routes dynamically calculated for the user, nor are pre-defined routes shown on a map, nor are routes described by other means such as a textual description. Instead, the system provides survey knowledge; this means that the user can see where a room is located *relative to other* rooms or places. Unfortunately, street names - and thus an important aspect of survey knowledge - are missing on most of the plans.

Landmarks: The maps used in the system in Graz do not include landmarks objects at all. Rather, the maps are similar to architectural plans, as can be seen in figure 43b.

Map orientation: Map orientation is different for most of the floor plans. The city map and the maps showing a building with its surroundings are mostly North-oriented.

Indoor space: Indoor space is shown using different maps for each floor level. The system does not promote any specific indoor wayfinding strategy.

Search style: The main style of interaction when searching for a specific room is text-based search for room names and browsing through a menu structure. The application's entry point is a flexible search form that stays on top of the screen during almost all of the possible interactions with the

[1] `http://whitepages.tuwien.ac.at/`, Sept 2009

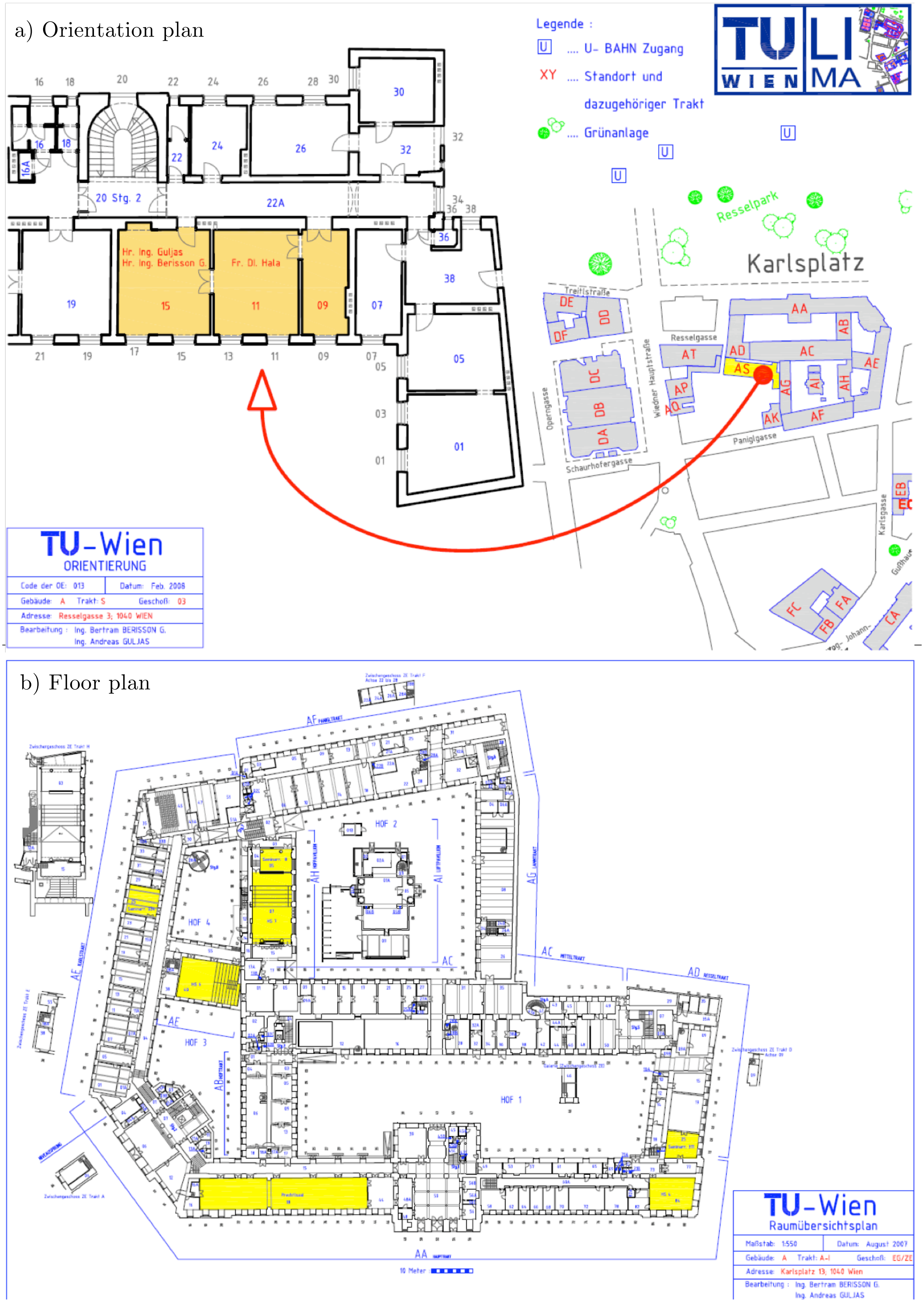


Figure 44: The plans used by Vienna University of Technology's real estate management

system. As can be seen in figure 43a and b, the form changes dynamically depending on what is currently displayed. Using the form, the user can drill up and down in the level of detail (e.g. drill up from a floor plan to the map showing the building and its surroundings) or select other maps of the same level of detail. Searching for rooms by zooming and panning or otherwise navigating on a map is not possible.

Navigation style: The application is built using static images. Zooming and panning or otherwise navigating within the maps is not possible.

Internet technology: A web application using HTML and Javascript. The maps are provided as JPEG images. A transparent GIF image is laid over the maps, maybe to hinder all-too-easy downloading.

8.3 University of Innsbruck's "Student City Map"

The "Student City Map" offered by the University of Innsbruck, Austria, combines an interactive city map with information on where the university's buildings are located. The system does not show indoor space (e.g. floor plans of individual rooms), but provides links to the institutions using a building for further information.

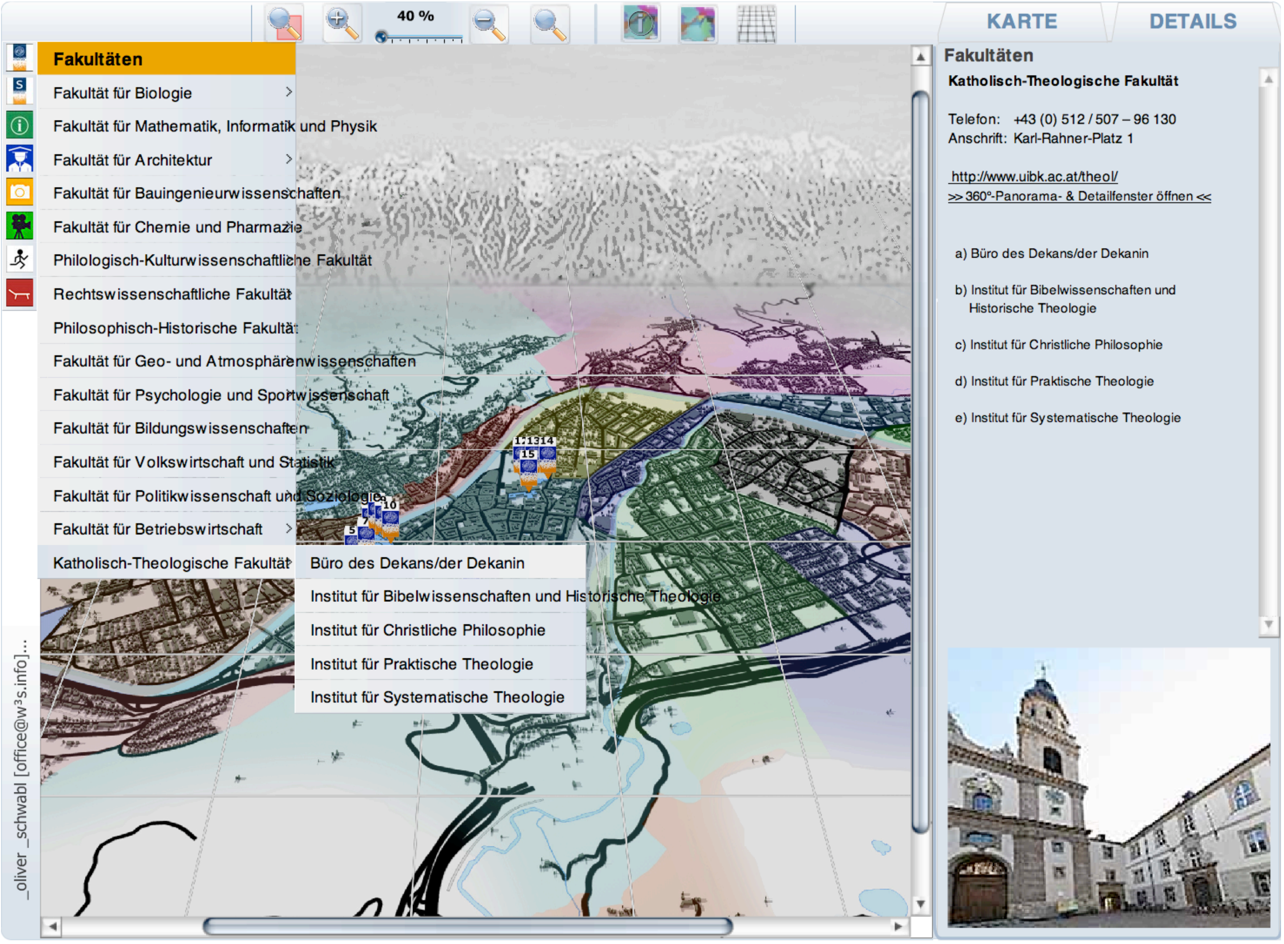
Medium: A web application using an interactive map and interactive panorama images to communicate wayfinding information.

Level of map abstraction and schematization: As can be seen in figure 45a, the city map's street graph reminds of an arial photograph, preserving lines and angles and distances. Nevertheless, the map does not show a lot of details: Only the rough shapes of building blocks can be guessed from it. The colors on the map are not natural, instead color is used to indicate the different city quarters. Street names are not shown on the map.

Route versus survey knowledge: The interactive map (see figure 45a) provides survey knowledge and does in no way indicate a specific route to be followed. The system does however link to the different institution's websites (as can be seen in figure 45a and b), and in most cases, these institutions will provide some kind of "how to get to our place" route information. For some buildings, a 3D interactive panorama providing additional survey knowledge is available, as can be seen in figure 45b.

Landmarks: The interactive map (figure 45a) features one very prominent landmark, which is the mountain range "Nordkette" on the northern edge of the city of Innsbruck. These mountains are easy to recognize and can be seen from everywhere in the city, making them a perfect 'global' landmark. Apart from this, the map does not feature other landmarks such as prominent buildings or well-known places. For some university buildings however, a photograph is availabe (clicking on one of buildings on the map will open the details-pane on the right hand side, showing a photograph of that building. See figure 45a). In this way, the building itself can become a landmark for the user, since he or she will recognize it by looking for its visually prominent features.

a) Interactive "Student City Map"



b) Interactive Panorama

Figure 45: The “Student City Map” offered by the University of Innsbruck

Map orientation: The interactive map is North-oriented. The 3D panorama images open with a prominent view on the building not matter what orientation.

Indoor space: The system does not show indoor space, nor does it verbally or otherwise describe individual rooms and floors.

Search style: The system offers a menu structure with a list of faculties and institutions (figure 45a). Alternatively, the user can search by navigating the map using pan-and-zoom actions and selecting buildings shown on the map until the right building is found. Also, from a list with street names, a street can be selected and displayed on the map. The reverse process of finding out a street name by choosing it on the map is possible for buildings (because the building's address is displayed), but not for streets in general.

Navigation style: The interactive map offers panning, zooming and selecting objects as main means of interaction. In the interactive panorama, the user is also able to pan and zoom, but is otherwise more limited: The user cannot select objects in the panorama (e.g. buildings, institutions) for more information. Instead, the infopane on the right-hand side of the panorama (see figure 45b) always stays the same. It is also not possible to select an object from the info pane on the right-hand side of the panorama and have the panorama display that object by rotating and zooming in the right direction. Also, as usual for panoramic images, the user cannot change his viewpoint (e.g. 'walk around').

Internet technology: The interactive map has been realized using Adobe Flash. The flash application dynamically loads information from a server as needed. The 3D panoramic images are provided as QuickTime movies embedded into an HTML page.

8.4 University of Karlsruhe's campus map

The university of Karlsruhe, Germany offers two kinds of wayfinding information systems. One is an interactive campus plan[1] consisting of one map graphic that can interactively be zoomed and panned as shown in figure 46a. The other consists of 3D buildings modelled in Google Earth. Both systems shall be discussed in the course of this work, starting with the interactive campus map in this subsection. The 3D buildings are described in the following section 8.5.

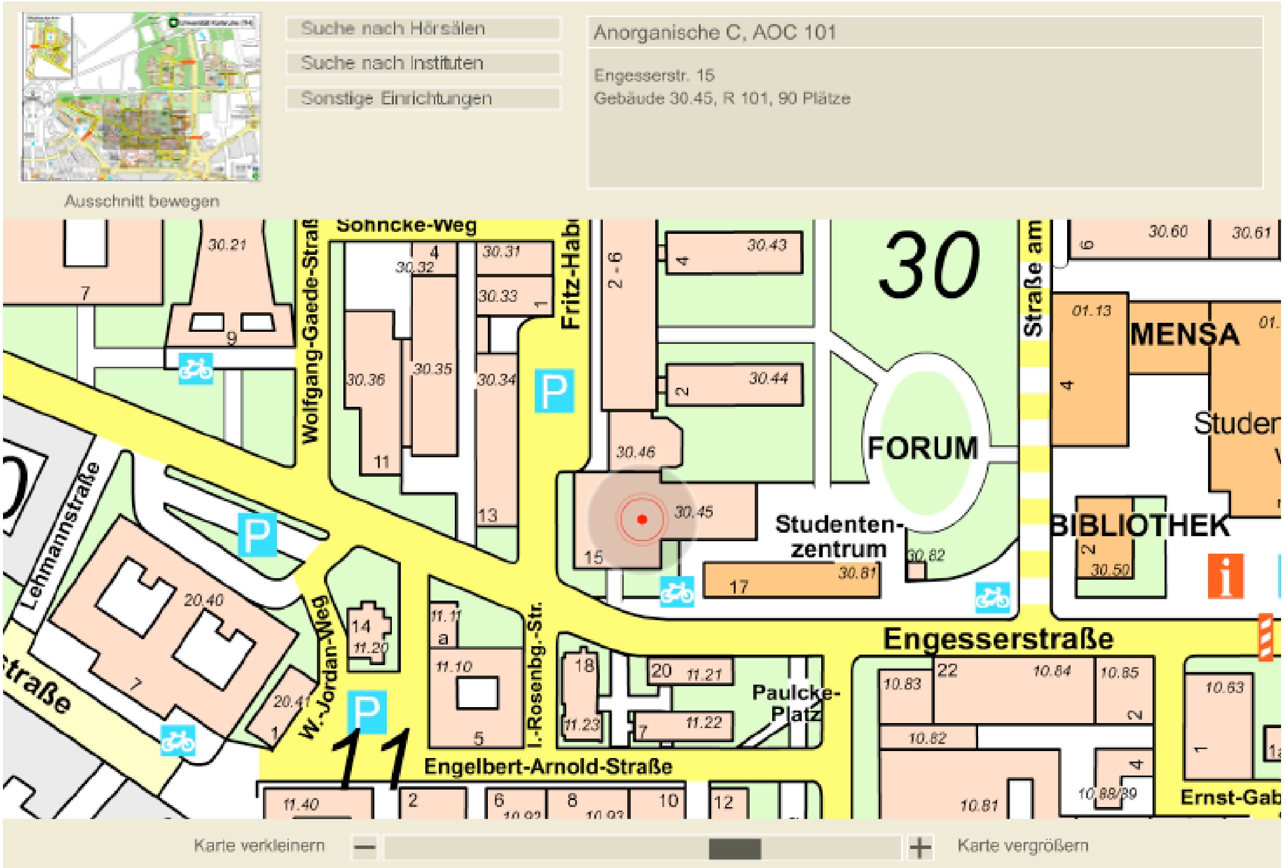
Medium: A web application using an interactive map to communicate wayfinding information.

Level of map abstraction and schematization: The natural, photo-realistic appearance of buildings and streets has been abstracted away. But angles and distances have been preserved. The map includes street names for the major roads on campus and the building's names.

Route versus survey knowledge: The system communicates survey knowledge. Routes are neither dynamically calculated for the user nor does the system display any pre-made routes.

[1] `http://www.uni-karlsruhe.de/info/campusplan/`, Sept 2009

a) Interactive campus map



b) 3D campus in Google Earth

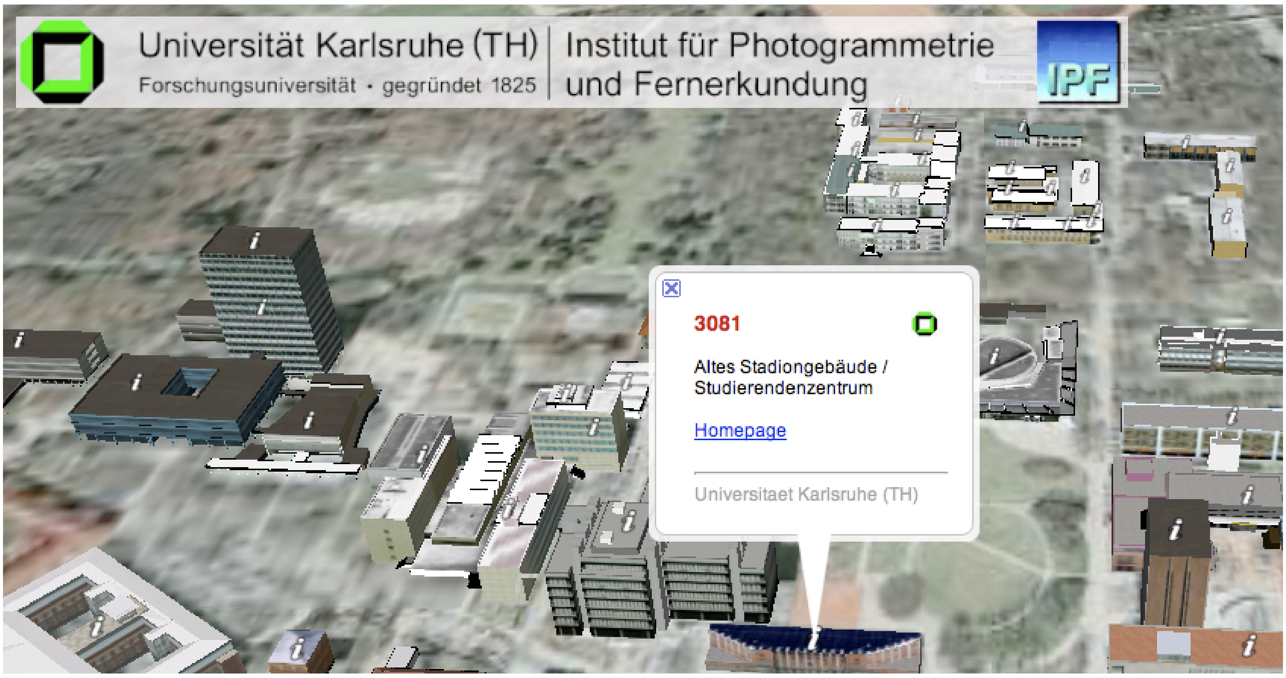


Figure 46: The University of Karlsruhe offers (a) an interactive campus map and (b) 3D-buildings in Google Earth.

Landmarks: The map used in the system includes diverse objects that may serve the user as local landmarks. Examples for such objects are an info-point building, gates to the campus area, parking places, and buildings with a special shape such as an oval-shaped 'forum'.

Map orientation: The map is roughly North-oriented.

Indoor space: The campus plan does not show indoor space. The user can however choose lecture halls from a list and a red dot on the map will highlight the according building.

Search style: The application has only one style of search interaction, and that is by choosing an item from one of three lists. Note that even though the interactive map can be navigated using zoom and pan actions, this cannot count as a means of searching for rooms: The reason for this is that it is impossible to select a building on the map and find out which rooms are contained within this building. A textual search for buildings, rooms, or institutions is not provided.

Navigation style: The map can be navigated using zoom and pan actions. Panning is however implemented in quite an unusual way: Dragging and dropping the map does not work. Instead, clicking in the upper area of the map will scroll the map down, clicking in the left area of the map will scroll the map to the right and so on. An additional tool is provided that shows where the current viewport is on the whole map, as can be seen in figure 46a in the upper-left corner. Using this tool, the user can scrool the map by dragging with his mouse the gray box representing the viewport.

8.5 University of Karlsruhe's 3D campus

In addition to the interactive campus map described in the previous section 8.4, the University of Karlsruhe have also created a 3D model of their buildings[1] and have published them using Google Earth. A screenshot of this system in action can be seen in figure 46b.

Medium: The University of Karlsruhe's 3D campus uses Google Earth (an online Desktop application) to provide photorealistic 3D models.

Level of map abstraction and schematization: The 3D model has not undergone much schematization at all - the aim has been on the contrary to create a realistic representation of the campus's buildings.

Route versus survey knowledge: Although Google Earth offers the possibility to dynamically calculate routes, this is not very helpful for on-campus navigation (the reason for this being that Google Earth does not know about the minor roads on campus or the building's names). Because of this limitation, the 3D buildings in Google Earth primarily provide survey knowledge.

Landmarks: Since the 3D models are a realistic representation of the actual campus, chances are that visually outstanding objects will be recognized and used as landmark in both the virtual and the real world. No special highlight has been placed on buildings with a high landmark saliency.

Map orientation: The 3d buildings and the map (or rather: arial photograph) that they have been placed upon can be freely rotated and can be viewed from an arbitrary angle in 3D space.

[1] `http://www.ipf.uni-karlsruhe.de/campus3d.html`, Sept 2009

Indoor space: Indoor space has not been modeled in this application. Technically, the system could provide information on the rooms contained within a building as hypertext within the bubbles (see figure 46b) that appear when the user clicks on an info-icon, but it does not.

Search style: Browsing the virtual world offered by the 3D building models. The ‘Fly to’ textual search offered by Google Earth does not work for the university buildings.

Navigation style: Zooming and panning, flying around in the virtual world.

Internet technology: The Google Earth desktop application loads data from Google’s and the University of Karlsruhe’s servers as needed.

8.6 Technical University of München’s roomfinder

The Technical University of München (TU München) have developed a wayfinding aid called ‘roomfinder’[1]. It allows a user to search for a room by entering the room’s number or the name of an institution. The according floor plan is then shown with a red dotted cross (see figure 47b) marking where the room is on the plan. The plans generated by the roomfinder system can not only be linked using a URL but can directly be embedded into websites. Another way of using the maps is provided in the form of a download link to a .kml file that opens in Google Earth. In Google Earth, the map is shown as an overlay image as shown in figure 47c.

Medium: TU München’s roomfinder is a web application. It uses maps (floor plans and campus maps with buildings and streets) to provide wayfinding information.

Level of abstraction and map schematization: Maps are not only provided in different scales (the user can choose from a list), but there are also different types of maps (overview maps with buildings and streets, but also floor plans). Thus the level of abstraction and schematization that has been applied is different from one map to the other. The floor plans use simplified shapes, but the basic proportions have been preserved. Some of the floor plans feature a highly abstracted topological graph that shows how the different parts of a building are interconnected (see figure 47b).

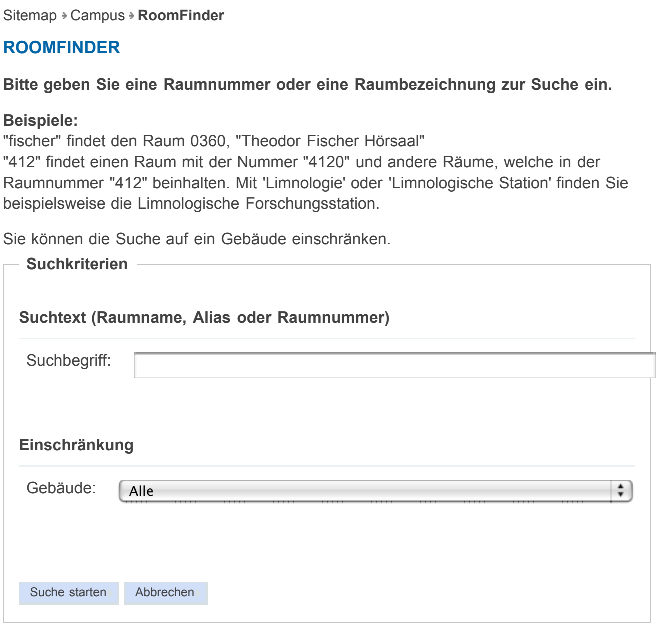
Route versus survey knowledge: For a given room, the system displays a map with the room’s surroundings, thereby providing survey knowledge. It does not dynamically calculate routes for the user. On some maps however, the main routes leading through a complex building are shown (see figure 47b), revealing the building’s indoor topology.

Landmarks: The maps used by the TU München’s wayfinding system are poor in landmarks: The rough shapes of buildings are visible in most cases, and can sometimes be used as landmark. Some maps contain street names and underground stations.

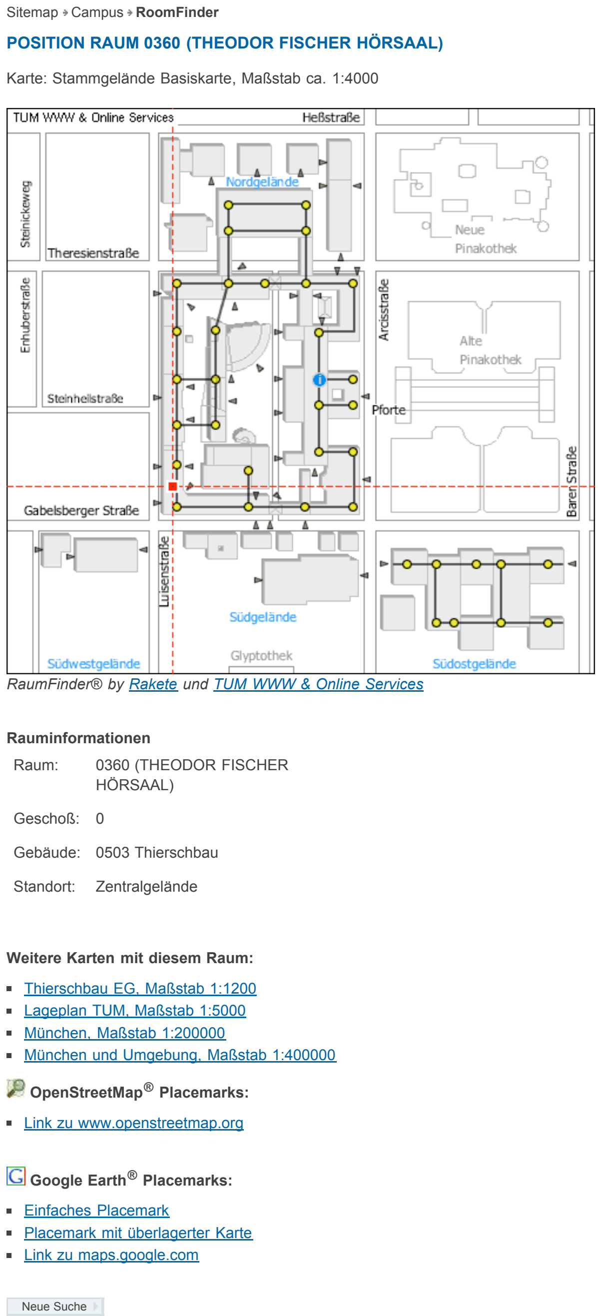
Map orientation: The maps orientation cannot be changed because the maps are provided as static images. The indoor maps are roughly North-oriented, but the orientation has been manipulated so

[1] `http://portal.mytum.de/campus/roomfinder/`, Sept 2009

a) The roomfinder's search form



b) The roomfinder displaying a room's location



c) The roomfinder's map displayed in Google Earth

Figure 47: The Technical University of München's roomfinder application.

that the rectangular grid of rooms and their walls is displayed horizontally and vertically on the map. Outdoor maps showing a larger street network are North-orientated.

Indoor space: Indoor space is shown using floor plans. The system uses one plan per floor and does not combine several floors on one map.

Search style: A search form (see figure 47a) is provided at the roomfinder application's entry point. In this form, the name of a room or institution can be entered, and search results - each linking to a map - are then displayed in a list. This is the main style of interaction used in the application for searching. Furthermore, roomfinder maps are found embedded or linked in websites of the University's institutions.

Navigation style: The maps are provided as static images without the possibility to zoom and pan or otherwise navigate within a map. Using hyperlinks to other maps, the user can show maps of the same region but with a different scale.

Internet technology: The application is built using HTML and JavaScript. The animated red dotted cross highlighting a room's location on a map is made using a GIF animation.

8.7 Herold.at's Strassentour

Herold Business Data GmbH[1] is an Austrian information provider and has been maintaining both printed and electronic phonebooks and directories for years. In June 2009, the company started a new service called street tour, "Herold Strassentour" [2]. As in Google's Street view, the user can walk along a series of panoramic images and thus immerge into a virtual 3D world. A screenshot of herold's street tour in action can be seen in figure 48a. Here, the user can 'walk along the street' by either clicking on one of the dots on the map or by clicking on the green arrows in the panorama images. Some shops are highlighted in the panorama and can be entered by clicking on it.

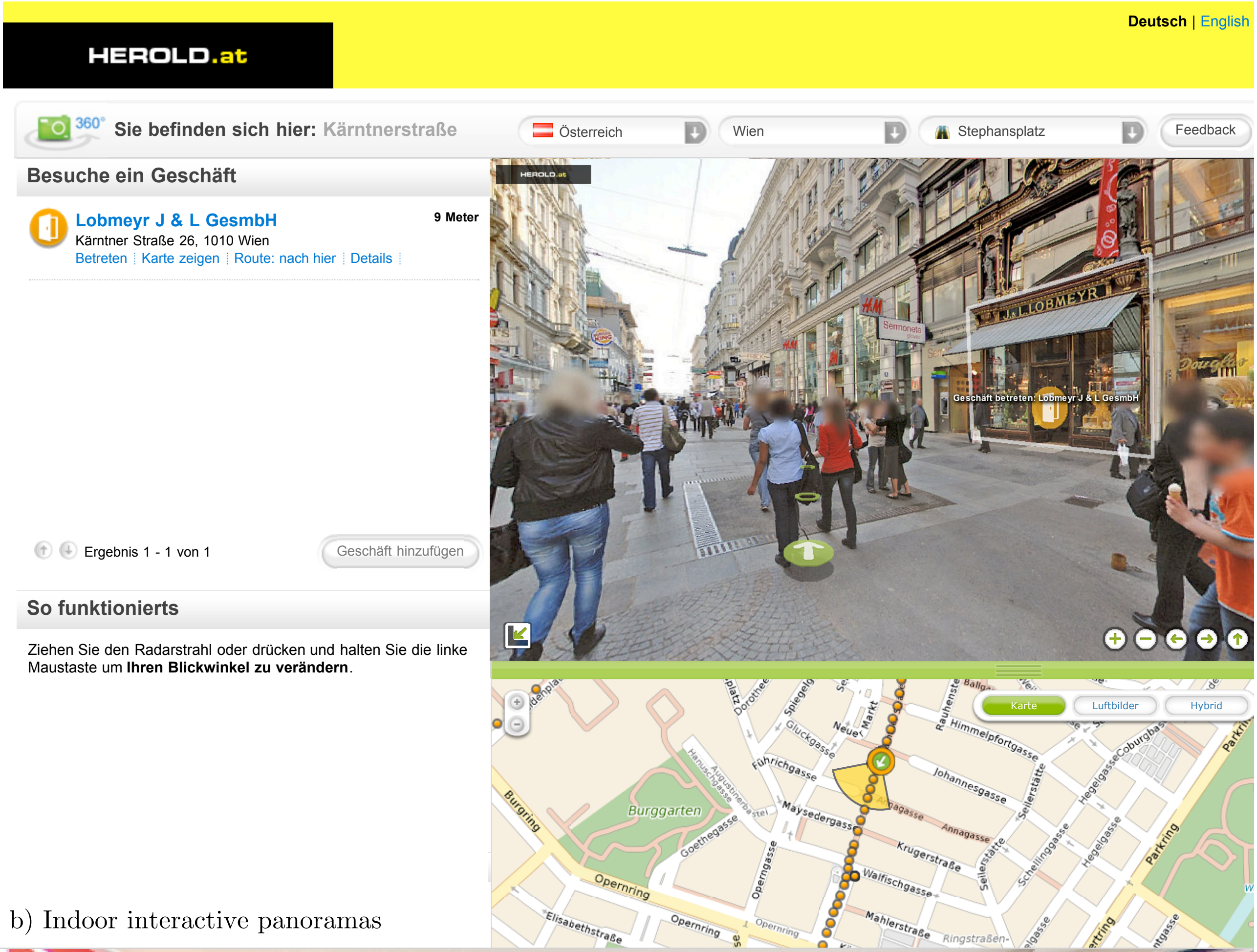
In contrast to Google's street view, Herold's street tour places its focus on shopping and also includes indoor space of shops and shopping malls. This makes the application an interesting example for other indoor wayfinding systems such as the wayfinding system to be built for the Vienna University of Technology.

Medium: A web application featuring a map and several series of inter-connected panorama images.

Level of abstraction and map schematization: As can be seen in figure 48a, the maps used by herold offer a fairly detailed street graph where angles and distances have been preserved. Street names and some additional names are provided as well. However the focus of this evaluation is not the (rather usual) street map, but the interactive panoramas. The panoramas offer a photo-realistic view. No information has been schematized, simplified, or otherwise left out.

[1] `http://www.herold.at/`, Sept 2009
[2] `http://strassentour.herold.at/`, Sept 2009

a) Street tour with interactive panoramas and a map



b) Indoor interactive panoramas

Figure 48: Herold.at offers interactive, navigatable panoramas for both indoor and outdoor locations.

Route versus survey knowledge: Both route and survey knowledge is offered by the sytem: Since by navigating from one panorama to the next, the user can only follow given routes, route knowledge is provided by the system. For outdoor situations, the route is displayed on a street map (see figure 48a), and for indoor situations, a small overview map is displayed (see figure 48b). The routes are in this case pre-defined and not dynamically calculated for the user. On his virtual walk along this route, the user learns a lot about the environment and thus gains some survey knowlege, too.

Landmarks: Because of the photo-realistic representation offered by panoramic images, natural objects that visually stand out as a landmark will be recognized as such in the virtual world as well.

Map orientation: The map (as in figure 48a) is North-oriented. The overview map provided for indoor situations is West-oriented. In the panoramas, the user is free to turn around in any direction. The current direction of view is indicated both on the street map and in the small overview map.

Indoor space: Indoor space is displayed as panoramic images in very much the same way as outdoor space. The 3D aspect of several inter-connected floor levels is handled in several cases for shops with several floors. In this case, the user is able to go up into the next floor by following an arrow pointing up a stair, and find the way back down by following another arrow that - unfortunately and sometimes to the author's confusion - points up, too.

Search style: Many different search styles are supported: Shops can be found using the herold.at phonebook's textual search. They can be found by zooming and panning the street map (the street map shows a door symbol for shops with indoor panoramas). Also, a list of shops nearby the user's current location on the streetmap is displayed in the left area of the screen as seen in figure 48a. From within the panoramic images, shops can be found by clicking on the entrance of a shop (the panoramas highlight with a white frame all shop entrances that can be entered).

Navigation style: Interactive panorama images where the user can zoom and pan around. Arrows displayed in the panorama act as hyperlinks to the next panorama. By using these hyperlinks, the user can navigate in the virtual world without leaving the panoramic images. The author found this type of navigation confusing in some cases, because he lost the context and sense of direction when clicking from one panorama to the next. Smaller transitions from one panorama to the next should help with this problem.

Internet technology: The website makes heavy use of HTML, JavaScript (including the Yahoo User Interface library) and Adobe Flash for the display of panorama images.

9 Appendix: Personas

Li-Sun (Lilly)

Li Sun is a young research assistant at Vienna University of Technology. Originally from China, she has been living in Vienna for almost four years now. She has in the meantime become used to simply being called Lilly because the people around her were always having difficulties pronouncing and remembering her name. When she was new to Vienna, she found it rather difficult to establish contact with local people, so many of her friends come from different countries (her boyfriend is from Israel). Lilly likes this international flair but she is a little embarrassed about her bad German – she speaks English perfectly well, though. Despite the fact that Lilly still needs to finish her master's thesis in mechanical engineering, she is already involved in two research projects and is about to hold her first lecture in an introductory subject. For her future career, she would like to accept the PhD position offered to her, and stay at university to do further research if possible.

Needless to say, as a mechanical engineer Li-Sun is used to working with the computer, doing research on the internet, using web applications and even some programming if required to solve a mathematical problem.

To manage her many tasks and meetings, Lilly has an electronic calendar where she sometimes also writes down the location of an unknown place. She has been using the wegweiser wayfinding system a lot, mainly when a new term started and lectures took place in new locations. When last year, she had a knee injury from a skiing accident, Lilly realized how different the world must be for people in wheelchairs (luckily, she did not need one, but she did have a hard time climbing stairs!).

Figure 49: The persona 'Lilly' is a chinese research assistant who is familiar with the university's. Photos under creative commons license, taken from http://www.flickr.com/photos/angelatchou/3581516921/ and http://www.flickr.com/photos/angelatchou/3581516921/

Peter

Peter is a forty-six year old father, as he would present himself. He likes to speak about his family first because "that's what really matters in life", as he likes to put it. His recent divorce obviously has not left him untouched, but has only strengthened the relationship with his two children, Anne and David.

Peter has been in charge of maintaining industrial sewing machines for an automobile supplier during the past twelve years. During this time, he has more and more grown into managing all kinds of related purchases for his department. With his many contacts and firm practical background, Peter is now considering to start his own business. He hopes to gain the necessary know-how in business administration and better computer skills by taking courses twice a week in the evening at a conference room in one of Vienna University of Technology's buildings.

One difficulty with computers and "yet another reason" why Peter has disliked computers in the past is that with his strong glasses, he is having a hard time to read the small fonts on the flickering computer screens. Luckily, the new laptop he bought really cheap from a friend has a much better screen. While training for his computer exam, Peter accidentally found out how to increase the size in his web browser, which works great for some sites, but messes things up with others. Too bad his Word program does not have the same feature.

For his first course, Peter had to call the instructor twice because he could not find the place. Unfortunately, the location won't stay the same but change every few weeks, but the instructor said he should look for that campus map on the internet which supposedly is "very easy to use", a phrase Peter has heard all too often during his computer courses.

Figure 50: The persona 'Peter' is an Austrian office worker who is new to the university's campus. Photos under creative commons license, taken from `http://www.flickr.com/photos/rberteig/170091952/` and `http://www.flickr.com/photos/caseywest/4604030891/`

10 Appendix: Storyboards

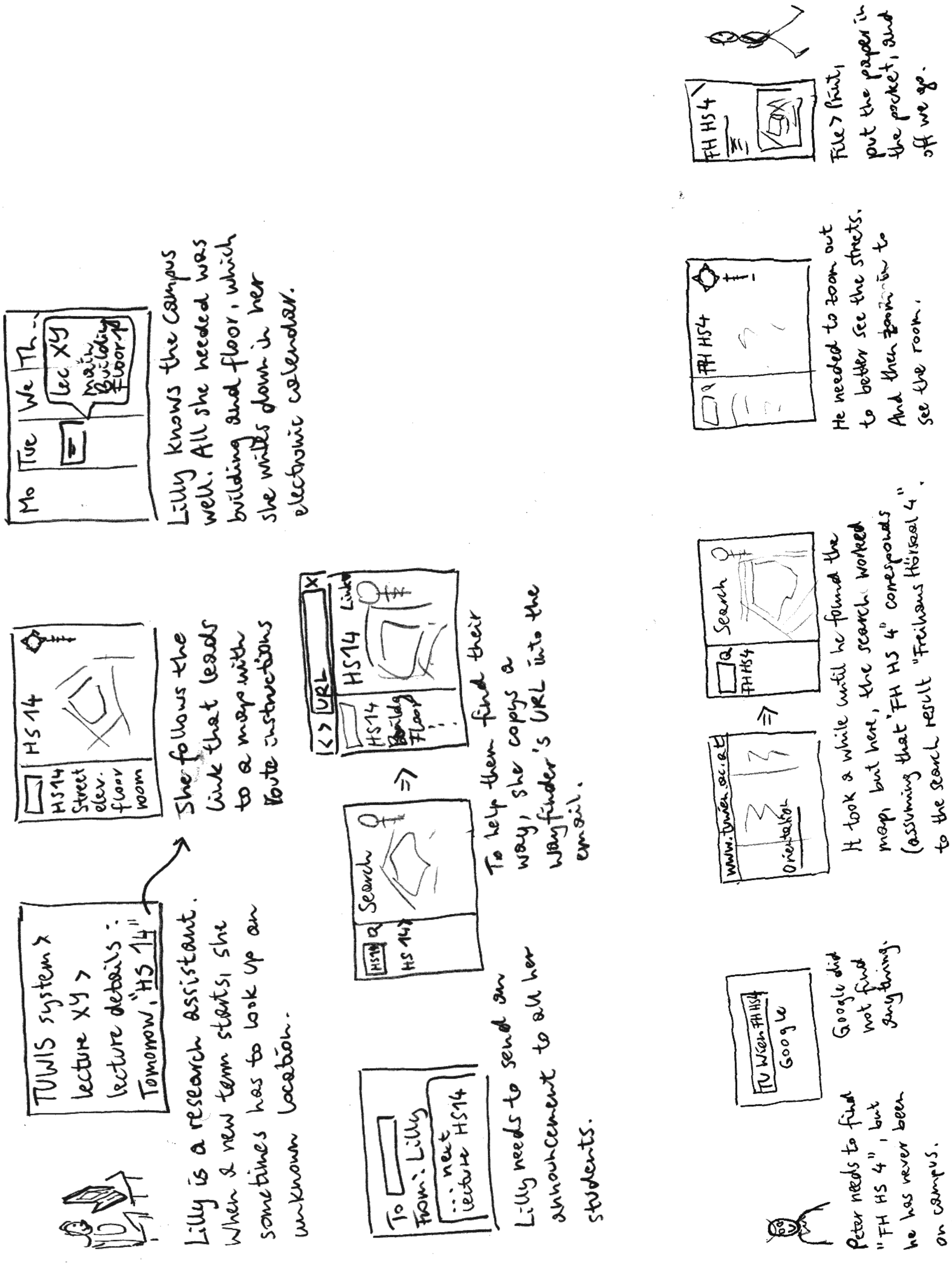


Figure 51: Storyboard: Lilly follows a link from the TUWIS system and shares a link with others. Peter needs to find the wayfinding website using Google, and then continues using the search-function.

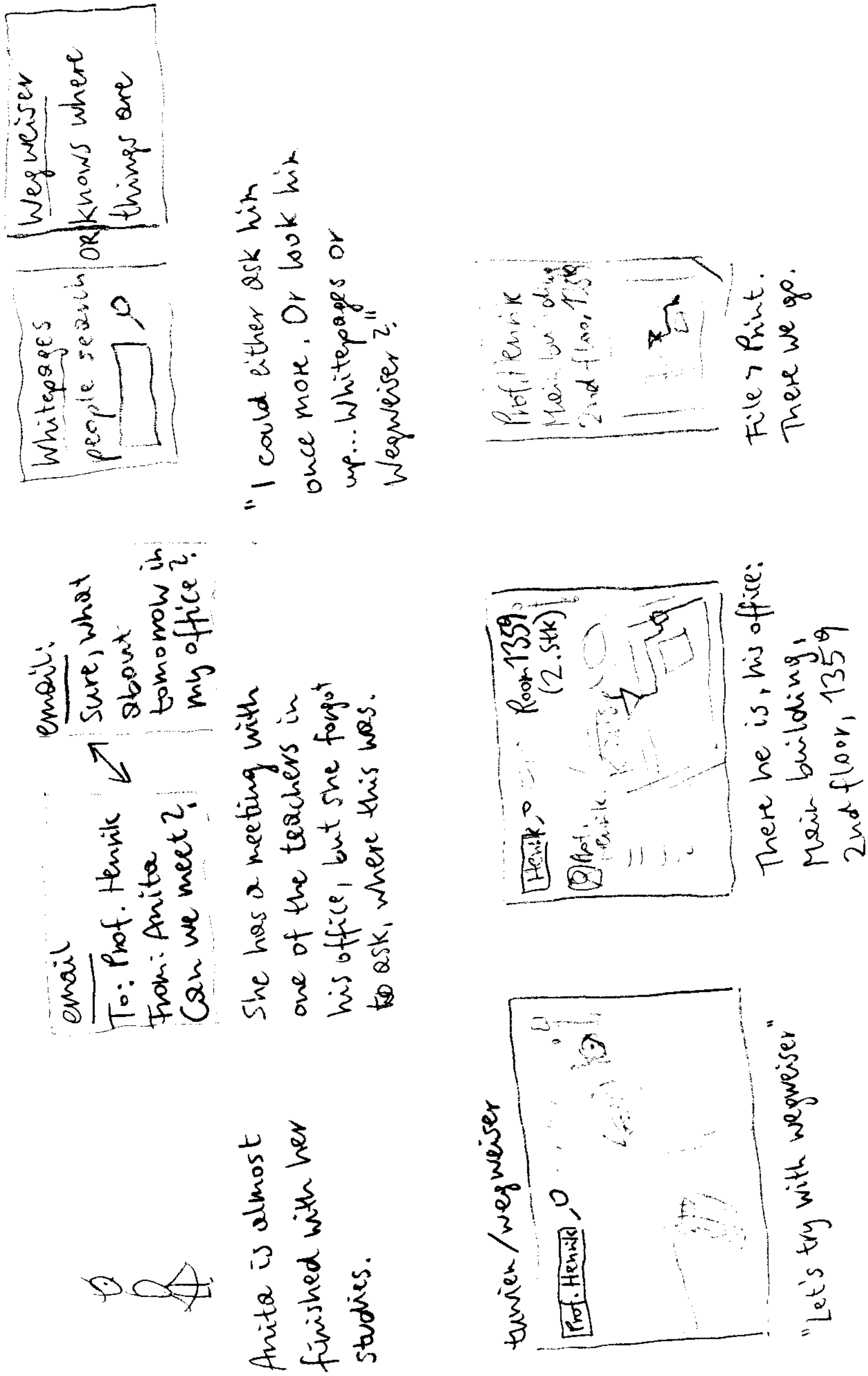


Figure 52: Storyboard: Looking for a teacher's office.

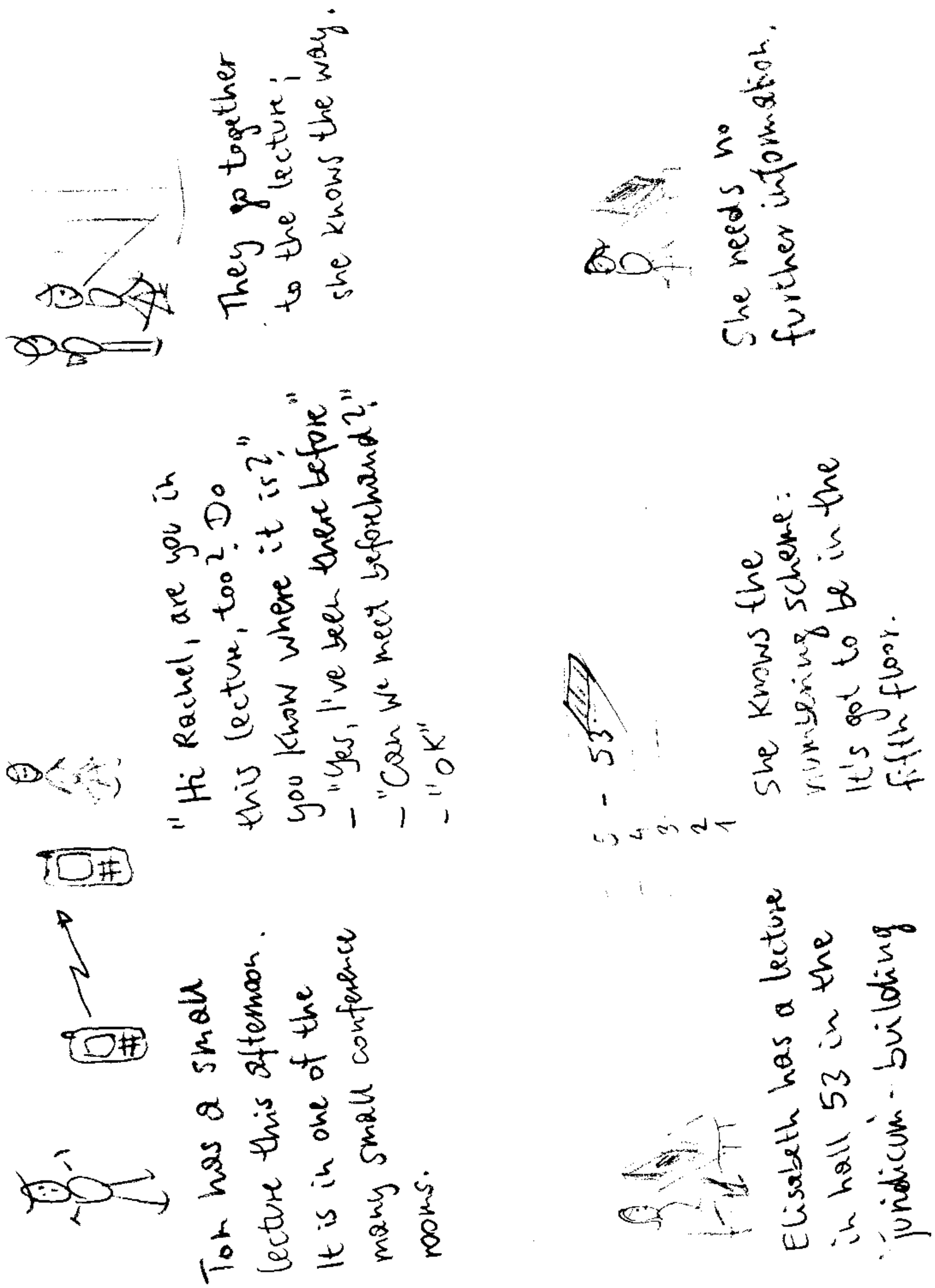


Figure 53: Anti-Storyboard: Ways of not using the system, or competing ways of getting wayfinding information: Calling friends or asking people, and making educated guesses about where a place should be located.

11 List of Figures

1 Defining wayfinding . . . 10
2 The Communication model in map design . . . 13
3 Different media used as wayfinding aid . . . 19
4 Different media combined in one wayfinding aid . . . 20
5 Map schematization . . . 21
6 Route schematization . . . 22
7 Route descriptions using prototypical, 8-directional turning concepts . . . 23
8 Route modelling using decision points and motion concepts . . . 24
9 A classification of landmarks . . . 27
10 Usability and other attributes of system acceptibility . . . 29
11 Accessibility: The University of Montana's access map . . . 31
12 Usability engineering and bridging the gap . . . 33
13 Discount usability engineering . . . 35
14 Mayhew's usability engineering lifecycle . . . 37
15 Iterative development processes . . . 38
16 Nielsen's usability engineering lifecycle . . . 38
17 Approaches to user involvement . . . 40
18 Exemplary persona definition . . . 45
19 Cheap prototyping at the beginning of the design process . . . 48
20 How many test users in a usability test? . . . 49
21 This work's iterative development process . . . 52
22 Evaluation of existing solutions . . . 57
23 Personas: An overview . . . 66
24 Storyboard: Lilly and Peter . . . 73
25 Exemplary paper mockup . . . 75
26 Wireframe: Four ways of accessing the system . . . 77
27 Wireframe: Four types of media . . . 78
28 Wireframe: Combining ways of accessing the system and media . . . 79
29 Technical feasibility prototype . . . 81
30 Dimensional data analysis . . . 84
31 A proposed hypertext structure . . . 85
32 The evolution of map Design . . . 88
33 Façades . . . 89
34 Icon Design . . . 90

35 Wireframe design for search results . . . 92
36 The search algorithm's pseudocode . . . 93
37 Sidebar entries . . . 97
38 The prototype's software architecture . . . 104
39 The prototype's general layout . . . 106
40 The prototype's map design . . . 107
41 The prototype's search and 'link to this page' functions . . . 108
42 Wegweiser.ac.at . . . 120
43 University of Graz . . . 122
44 Real estate management plans . . . 124
45 University of Innsbruck . . . 126
46 University of Karlsruhe . . . 128
47 TU München . . . 131
48 Herold.at . . . 133
49 Persona: Lilly . . . 135
50 Persona: Peter . . . 136
51 Storyboard: Lilly and Peter . . . 137
52 Storyboard: Looking for a teacher . . . 138
53 Anti-Storyboard: Ways of not using the system . . . 139

12 List of Abbreviations

2D	two-dimensional
3D	three-dimensional
Ajax	Asynchronous JavaScript and XML, a web development technique enabling a webpage to load data asynchronously from the server
API	Application programming interface
CHI	Computer-human interaction
CSCW	Computer supported cooperative work
CSS	Cascading Style Sheets, a language used to describe the look of documents such as webpages
EG	Erdgeschoß, the german word and abbreviation for ground floor
EI	Elektrotechnisches Institutsgebäude, a building within the Vienna University of Technology's campus
Flash	Adobe Flash, a software product and platform for multi-medial user interfaces
Geostack	Geographic software stack
GIF	Graphics Interchange Format, a bitmap image format
GIS	Geographic information system, a computer system for location-based data
GPS	Global Positioning System, a satellite system that globally provides time and location information
HCI	Human-computer interaction
Hof	Hof is the german word for courtyard
HS	Hörsaal, the german word and abbreviation for lecture hall
HTML	Hypertext Markup Language, a publishing language for the World Wide Web
HTTP	Hypertext Transfer Protocol, a networking protocol used in the World Wide Web
ICF	International Classification of Functioning, Disability and Health: The World Health Organization's classification of functioning and disability

ICIDH	International Classification of Impairments, Disabilities, and Handicaps: The World Health Organization's former classification of functioning and disabilities
iPhone	The iPhone is a mobile phone and internet and multimedia device created by Apple Inc.
ISO	International Organization for Standardization
Java	Java is a programming language for cross-platform software development
JavaScript	JavaScript is a scripting language supported by most web browsers
JPEG	Joint Photographic Experts Group, and a bitmap image format named after the group
JQuery	JQuery is a JavaScript framework for web development
KML	Keyhole markup language, an XML-based file format for geographic data that was adopted by Google
ÖNORM	Österreichische Norm, a standard published by the Austrian Standards Institute
OGC	Open Geospatial Consortium
OG	Obergeschoß, the german word and abbreviation for upper floor
OLAP	Online analytical processing applications support viewing and analyzing multi-dimensional data, todo
OOSE	Object-oriented software engineering
OpenLayers	OpenLayers is a JavaScript library for interactive maps
OSM	Open Street Map, a project for collaborative mapping over the internet
PDA	Personal digital assistant, a pocket-sized computer
PDF	Portable Document Format, an open format for two-dimensional documents
PostGIS	PostGIS is a relational database system with extensions for geographic data
Quicktime	Quicktime is a multimedia framework created by Apple Inc.
ROI	Return on investment, the amount of money gained relative to the amount of money invested
SketchFlow	SketchFlow is a software product for rapid prototyping by Microsoft

SQL	Structured Query Language, a declarative query language for relational database systems
SVG	Scalable Vector Graphics, an open standard for vector graphics
TISS	TU Wien Informations-Systeme und Services, a project at the Vienna University of Technology to renew their software infrastructure
TU Wien	Technische Universität Wien, the german name for the Vienna Technical University
TUWIS	TU Wien Informationssystem, an information system of the Vienna University of Technology
UCD	User-centered design
UE	Usability evaluation
UID	User interface design
UI	User interface
URL	Uniform resource locator, an identifier for webpages and other resources that specifies where it is available and how it can be retrieved
UXD	User experience design
W3C	World Wide Web Consortium
WAI	Web Accessibility Initiative, the World Wide Web Consortium's efforts to make the web accessible to people with different abilities (todo)
WCAG	Web Content Accessibility Guidelines, a checklist by the World Wide Web Consortium to verify a website's accessibility
WebGL	WebGL is a specification for 3D-graphics rendered inside an HTML webpage
WFS	Web Feature Service, an OGC standard for a web service interface to allow querying of geographic features
WHO	World Health Organization
WMS	Web Map Service, an OGC standard for a web service interface to allow querying of map images
XML	Extensible Markup Language, an open standard for a data format to encode hierarchical data

13 References

Abu-Ghazzeh, T. M. (1996). Movement and wayfinding in the King Saud University built environment: A look at freshman orientation and environmental information. *Journal of Environmental Psychology*, 16(4):303–318.

Allen, G. (1999). *Spatial abilities, cognitive maps, and wayfinding.* Johns Hopkins University Press, Baltimore.

Allen, G. L. (2000). Principles and practices for communicating route knowledge. *Applied Cognitive Psychology*, 14(4):333–359.

Barkowsky, T. (2001). Mental processing of geographic knowledge. *Spatial Information Theory*, pages 371–386. URL: `http://dx.doi.org/10.1007/3-540-45424-1_25`, retrieved 2010/09/10.

Baus, J., Wasinger, R., Aslan, I., Krüger, A., Maier, A., and Schwartz, T. (2007). Auditory perceptible landmarks in mobile navigation. In *IUI '07: Proceedings of the 12th international conference on Intelligent user interfaces*, pages 302–304, New York. ACM.

Becker, C. and Dürr, F. (2005). On location models for ubiquitous computing. *Personal Ubiquitous Comput.*, 9(1):20–31.

Beebe, J. (2001). *Rapid assessment process: An introduction.* AltaMira Press.

Benyon, D., Turner, P., and Turner, S. (2006). *Designing interactive Systems.* Addison-Wesley.

Berendt, B., Rauh, R., and Barkowsky, T. (1998). Spatial thinking with geographic maps: An empirical study. *Herausforderungen an die Wissensorganisation: Visualisierung, multimediale Dokumente, Internetstrukturen*, pages 63–73.

Bortz, J. and Döring, N. (2006). *Forschungsmethoden und Evaluation für Human- und Sozialwissenschaftler.* Springer Berlin Heidelberg, 4. überarbeitete Auflage edition.

Buxton, B. (2007). *Sketching user experiences: Getting the design right and the right design.* Morgan Kaufmann.

Cheung, A. (2006). Representational issues in interactive wayfinding systems: Navigating the Auckland University campus. *Web and Wireless Geographical Information Systems*, pages 90–101.

Cooper, A. (2004). *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity (2nd Edition).* Pearson Higher Education.

Cope, A. and Jorgenson, M. (2009a). Interactive maps: Report from Cardiff University. Technical report, Cardiff University. URL: `http://www.ja.net/development/network-access.html`, retrieved 2010/09/10.

Cope, A. and Jorgenson, M. (2009b). Overview of location technologies. Technical report, Cardiff University. URL: `http://www.ja.net/development/network-access.html`, retrieved 2010/09/10.

Cova, T. J., Miller, H. J., Beard, K., Frank, A. U., and Goodchild, M. F., editors (2008). *Geographic Information Science, 5th International Conference, GIScience 2008, Park City, UT, USA, September 23-26, 2008. Proceedings*, volume 5266 of *Lecture Notes in Computer Science*. Springer.

Dittrich, Y., John, M., Singer, J., and Tessem, B. (2007). For the special issue on qualitative software engineering research. *Information and Software Technology*, 49(6):531 – 539. Qualitative Software Engineering Research.

Dodiya, J. and Alexandrov, V. N. (2008). Use of auditory cues for wayfinding assistance in virtual environment: music aids route decision. In Feiner, S., Thalmann, D., Guitton, P., Fröhlich, B., Kruijff, E., and Hachet, M., editors, *VRST*, pages 171–174. ACM.

Druin, A. (2002). The role of children in the design of new technology. *Behaviour and Information Technology*, 21(1):1–25.

Dumas, J. and Redish, J. (1999). *A practical guide to usability testing*. Intellect Ltd.

Engel, D., Bertel, S., and Barkowsky, T. (2005). Spatial principles in control of focus in reasoning with mental representations, images, and diagrams. *Spatial Cognition IV. Reasoning, Action, and Interaction*, pages 181–203.

Eriksson, E., Cajander, Å., and Gulliksen, J. (2009). Hello world! –experiencing usability methods without usability expertise. *Human-Computer Interaction –INTERACT 2009*, pages 550–565.

Filstead, W. J. (1971). *Qualitative Methodology: Firsthand Involvement with the Social World*. Markham Publishing.

Fontaine, S. and Denis, M. (1999). The production of route instructions in underground and urban environments. *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*, pages 747–747.

Froschauer, U. and Lueger, M. (1992). *Das qualitative Interview: zur Analyse sozialer Systeme*. WUV, Univ.-Verl.

Gartner, G., Haosheng, H., Ortag, F., and Millonig, A. (2009). Strategies of context-based and semantic adaptive route communication. In Tomko, M. and Richter, K.-F., editors, *AGILE Workshop on Adaptation in Spatial Communication*, number 019-05/2009 in Report Series of the Transregional Collaborative Research Center SFB/TR 8 Spatial Cognition, Universität Bremen.

Gartner, G. and Radoczky, V. (2006). About the role of cartographic presentation for wayfinding. *Geographic Hypermedia*, pages 381–398.

Giudice, N., Bakdash, J., and Legge, G. (2007). Wayfinding with words: spatial learning and navigation using dynamically updated verbal descriptions. *Psychological Research*, 71(3):347–358.

Gruber, W. (2010). Entwurf eines SOA-Konzepts für Location-Based Services. Master's thesis, Vienna University of Technology.

Haesen, M., Luyten, K., and Coninx, K. (2009). Get your requirements straight: Storyboarding revisited. *Human-Computer Interaction –INTERACT 2009*, pages 546–549.

Haklay, M., Singleton, A., and Parker, C. (2008). Web mapping 2.0: The neogeography of the geoweb. *Geography Compass*, 2(6):2011–2039.

Herczeg, M. (2006). *Interaktionsdesign. Gestaltung interaktiver und multimedialer Systeme*. Oldenburg Verlag München.

Heuten, W., Henze, N., Boll, S., and Pielot, M. (2008). Tactile wayfinder: a non-visual support system for wayfinding. In Gulz, A., Magnusson, C., Malmborg, L., Eftring, H., Jönsson, B., and Tollmar, K., editors, *NordiCHI*, volume 358 of *ACM International Conference Proceeding Series*, pages 172–181. ACM.

Hirtle, S. (2000). The use of maps, images and gestures for navigation. *Spatial Cognition II*, pages 31–40.

Hirtle, S. C. and Sorrows, M. E. (1998). Designing a multi-modal tool for locating buildings on a college campus. *Journal of Environmental Psychology*, 18(3):265–276.

Hochmair, H. (2002). *The wayfinding metaphor. Comparing the semantics of wayfinding in the physical world and the WWW*. PhD thesis, Technische Universität Wien, Fakultät für Mathematik und Geoinformation, Institut E127 für Geoinformation und Kartographie.

Hölscher, C., Meilinger, T., Vrachliotis, G., Brösamle, M., and Knauff, M. (2005). Finding the way inside: Linking architectural design analysis and cognitive processes. *Spatial Cognition IV. Reasoning, Action, and Interaction*, pages 1–23.

Hopf, C. (1978). Die Pseudo-Exploration. Überlegungen zur Technik qualifizierter Interviews in der Sozialforschung. *Zeitschrift für Soziologie*, 7:97–115.

Hughes, J., King, V., Rodden, T., and Andersen, H. (1994). Moving out from the control room: ethnography in system design. In *CSCW '94: Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 429–439, New York. ACM.

Iivari, N. (2006). 'Representing the User' in software development – a cultural analysis of usability work in the product development context. *Interacting with Computers*, 18(4):635 – 664. Special Theme Papers from Special Editorial Board Members (contains Regular Papers).

International Organization for Standardization (2003). ISO/TS 16071:2003. URL: `http://www.iso.org/iso/catalogue_detail.htm?csnumber=30858`, retrieved 2010/09/10.

Ishikawa, T. and Kiyomoto, M. (2008). Turn to the left or to the west: Verbal navigational directions in relative and absolute frames of reference. In Cova et al. [2008], pages 119–132.

Ishikawa, T. and Yamazaki, T. (2009). Showing where to go by maps or pictures: An empirical case study at subway exits. *Spatial Information Theory*, pages 330–341.

Kelly, J. and McNamara, T. (2008). Spatial memory and spatial orientation. *Spatial Cognition VI. Learning, Reasoning, and Talking about Space*, pages 22–38.

Klippel, A. (2003). Wayfinding choremes. *Lecture notes in computer science*, pages 301–315.

Klippel, A., Richter, K., Barkowsky, T., and Freksa, C. (2005). The cognitive reality of schematic maps. *Map-based Mobile Services-Theories, Methods and Implementations*, pages 57–74.

Koláčcný, A. (1969). Cartographic information - a fundamental concept and term in modern cartography. *The Cartographic Journal*, 6(1):47–49(3).

Krygier, J. and Wood, D. (2005). *Making Maps. A Visual Guide to Map Design for GIS*. The Guilford Press.

Kuipers, B., Tecuci, D. G., and Stankiewicz, B. J. (2003). The Skeleton In The Cognitive Map: A Computational and Empirical Exploration. *Environment and Behavior*, 35(1):81–106.

Kujala, S. (2003). User involvement: a review of the benefits and challenges. *Behaviour and Information Technology*, 22:1–16.

Li, D. and Lee, D. L. (2008). A topology-based semantic location model for indoor applications. In *GIS '08: Proceedings of the 16th ACM SIGSPATIAL international conference on Advances in geographic information systems*, pages 1–10, New York. ACM.

Li, K. (2008). Indoor Space: A New Notion of Space. In *Proceedings of the 8th International Symposium on Web and Wireless Geographical Information Systems*, pages 1–3. Springer-Verlag Berlin, Heidelberg.

Linge, N. (2009). Interactive maps: Report from the University of Salford. Technical report, University of Salford. URL: `http://www.ja.net/development/network-access.html`, retrieved 2010/09/10.

Loo, T. (2010). Digital design agencies: Taking the U out of UCD. URL: http://www.foolproof.co.uk/taking-the-u-out-of-ucd/, retrieved 2010/06/14.

Lynch, K. (1960). *The image of the city*. MIT press.

Marcus, A. (2002). The ROI of Usability. URL: http://www.upassoc.org/usability_resources/usability_in_the_real_world/roi_of_usability.html, retrieved 2010/03/01.

Mautz, R. (2009). Overview of current indoor positioning systems. *Geodesy and Cartography*, 35(1):18–22.

May, A. and Ross, T. (2006). Presence and quality of navigational landmarks: effect on driver performance and implications for design. *Human factors*, 48(2):346.

Mayhew, D. J. (1999). *The Usability engineering lifecycle: A practitioner's handbook for user interface design*. Morgan Kaufmann.

Meilinger, T., Hölscher, C., Büchner, S. J., and Brösamle, M. (2006). How much information do you need? schematic maps in wayfinding and self localisation. In Barkowsky, T., Knauff, M., Ligozat, G., and Montello, D. R., editors, *Spatial Cognition*, volume 4387 of *Lecture Notes in Computer Science*, pages 381–400. Springer.

Millen, D. R. (2000). Rapid ethnography: time deepening strategies for hci field research. In *DIS '00: Proceedings of the 3rd conference on Designing interactive systems*, pages 280–286, New York, NY, USA. ACM.

Montello, D. (2002). Cognitive map-design research in the twentieth century: Theoretical and empirical approaches. *Cartography and Geographic Information Science*, 29(3):283–304.

Nielsen, J. (1993). *Usability engineering*. Morgan Kaufmann.

Nielsen, J. (1994). Guerrilla HCI: Using discount usability engineering to penetrate the intimidation barrier. URL: http://www.useit.com/papers/guerrilla_hci.html, retrieved 2010/09/10.

Nielsen, J. (2000). Why you only need to test with 5 users. URL: http://www.useit.com/alertbox/20000319.html, retrieved 11/07/2010.

Nielsen, J. (2008a). Bridging the designer–user gap. URL: http://www.useit.com/alertbox/designer-user-differences.html, retrieved 2010/03/10.

Nielsen, J. (2008b). Usability ROI Declining, But Still Strong. URL: http://www.useit.com/alertbox/roi.html, retrieved 2010/03/01.

Norman, D. (2002). *The design of everyday things.* Basic Books New York.

Omer, I. and Goldblatt, R. (2007). The implications of inter-visibility between landmarks on wayfinding performance: An investigation using a virtual urban environment. *Computers, Environment and Urban Systems*, 31(5):520–534.

Ortag, F. (2005). Sprachausgabe vs. Kartendarstellung in der Fußgängernavigation. Master's thesis, Vienna University of Technology.

Pedell, S. and Vetere, F. (2005). Visualizing use context with picture scenarios in the design process. In *MobileHCI '05: Proceedings of the 7th international conference on Human computer interaction with mobile devices & services*, pages 271–274, New York. ACM.

Ponniah, P. (2001). *Data warehousing fundamentals: a comprehensive guide for IT professionals.* Wiley-Interscience.

Pontikakis, E. (2006). *Wayfinding in GIS: formalization of basic needs of a passenger when using public transportation.* PhD thesis, Technische Universität Wien, Fakultät für Mathematik und Geoinformation, Institut E127 für Geoinformation und Kartographie.

Pruitt, J. and Grudin, J. (2003). Personas: practice and theory. In *DUX '03: Proceedings of the 2003 conference on Designing for user experiences*, pages 1–15, New York. ACM.

Raubal, M. and Winter, S. (2002). Enriching wayfinding instructions with local landmarks. *Lecture Notes in Computer Science*, 2478:243–259.

Rehrl, K., Leitinger, S., Gartner, G., and Ortag, F. (2009). An analysis of direction and motion concepts in verbal descriptions of route choices. *Spatial Information Theory*, pages 471–488.

Remolina, E. and Kuipers, B. (2004). Towards a general theory of topological maps. *Artificial Intelligence*, 152(1):47–104.

Richter, K. and Klippel, A. (2004). A model for context-specific route directions. In *Spatial Cognition IV. Reasoning, Action, and Interaction: International Conference Spatial Cognition*, pages 11–13. Springer.

Richter, K.-F. (2007). A uniform handling of different landmark types in route directions. In Winter, S., Duckham, M., Kulik, L., and Kuipers, B., editors, *COSIT*, volume 4736 of *Lecture Notes in Computer Science*, pages 373–389. Springer.

Richter, K.-F. and Duckham, M. (2008). Simplest instructions: Finding easy-to-describe routes for navigation. In Cova et al. [2008], pages 274–289.

Ross, T., May, A., and Thompson, S. (2004). The use of landmarks in pedestrian navigation instructions and the effects of context. *Mobile Human-Computer Interaction –MobileHCI 2004*, pages 300–304.

Rubin, J. (1994). *Handbook of usability testing*. Wiley New York.

Seffah, A. and Metzker, E. (2009). Usability engineering methods plethora. *Adoption-centric Usability Engineering*, pages 15–33.

Shneiderman, B. (1997). *Designing the user interface: strategies for effective human-computer interaction*. Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.

Simon, D. I. R. (2008). *Mobilizing the Geospatial Web: A Framework and Conceptual Model for Spatially Aware Mobile Web Applications*. PhD thesis, Vienna University of Technology.

Sorrows, M. and Hirtle, S. (1999). The nature of landmarks for real and electronic spaces. *Spatial Information Theory. Cognitive and Computational Foundations of Geographic Information Science*, pages 748–748.

Srinivas, S. and Hirtle, S. (2008). Knowledge based schematization of route directions. *Spatial Cognition V Reasoning, Action, Interaction*, pages 346–364.

Stoffel, E., Lorenz, B., and Ohlbach, H. (2007). Towards a Semantic Spatial Model for Pedestrian Indoor Navigation. *Lecture Notes in Computer Science*, 4802:328.

Stoffel, E.-P., Schoder, K., and Ohlbach, H. J. (2008). Applying hierarchical graphs to pedestrian indoor navigation. In Aref, W. G., Mokbel, M. F., and Schneider, M., editors, *GIS*, page 54. ACM.

Tom, A. and Denis, M. (2003). Referring to landmark or street information in route directions: What difference does it make? *Spatial Information Theory*, pages 362–374.

Travis, D. (2009). The Fable of the User-Centred-Designer. URL: `http://www.userfocus.co.uk/pdf/fable.pdf`, retrieved 2010/05/28.

Turner, A. (2006). *Introduction to neogeography*. O'Reilly Media, Inc.

Van der Lelie, C. (2006). The value of storyboards in the product design process. *Personal and Ubiquitous Computing*, 10(2):159–162.

Wealands, K. (2006). In pursuit of usefulness: Resetting the design focus for mobile geographic hypermedia systems. *Geographic Hypermedia*, pages 137–153. URL: `http://dx.doi.org/10.1007/978-3-540-34238-0_8`.