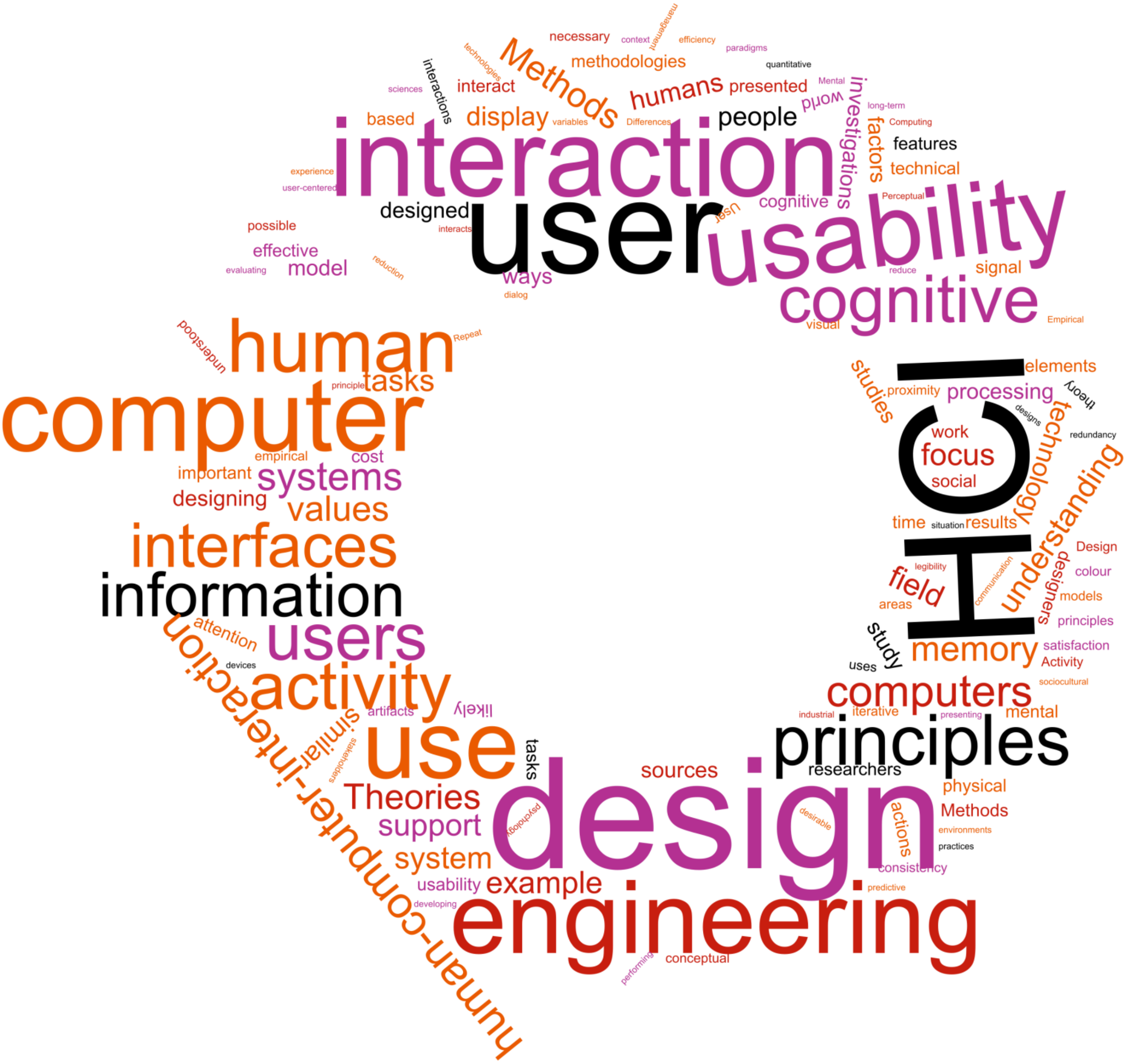
Mensch Computer Interaktion – Designspezifikation Meilenstein 2

Team

Lukas Gobelet	*11118011*
Joey Gobelet	*11118012*
Erik Gödel	*?*
Jonas Faßbender	*?*
Hamza ?	*?*



Einleitung

Dieses Dokument gliedert sich nach den Meilensteinen des Praktikums. Die jeweiligen Kapitel der Meilensteine stellen den Fortschritt und die Entwicklung des Teams im Laufe des Semesters dar. Ab Meilenstein 2 werden die vorhandenen Artefakte auf Basis von Feedback und Evaluationen überarbeitet. Da es sich beim Rapid Prototyping um eine Darstellung des aktuellen Konzepts handelt ist es nicht notwendig den jeweiligen Prototyp (Meilenstein 1: Story Board, Meilenstein 2: Wireframes, Paperprototyp) zum folgenden Meilenstein zu überarbeiten. Die Überarbeitung dieser Artefakte wird in Form des nächsten Prototypens dargestellt. D.h. die Überarbeitung des Storyboards zeigt sich in den Wireframes und dem Paperprototypen. Die Überarbeitung der Wireframes und des Paperprototypen zeigt sich im funktionalen Prototyp.

Was in den jeweiligen Abschnitten zu erbringen ist wird immer über *[Platzhalter]* gekennzeichnet. Hierbei gibt es 3 Arten von Kennzeichnungen:

[(Überarbeitete/Finale) Artefakt/e]

Platzhalter für das jeweilige Artefakt welches es anzufertigen bzw. zu überarbeiten gilt z.B. Personas.

[Erläuterung]

Platzhalter für eine Erläuterung des jeweiligen Artefakts. Hierbei ist keine Erläuterung der Methode oder Technik gemeint sondern vielmehr eine Begründung von Entscheidungen. Warum wurden ausgerechnet diese Personas erstellt? Warum sind Elemente im Wireframe wie im Artefakt ersichtlich angeordnet?
Dieser Punkt ist essenziell für das Bewertungskriterium „Dokumentation“. Der wichtigste Anhaltspunkt ist hierbei das Feedback seitens der Testnutzer als auch der Praktikumsbetreuer. Sie sollten stets versuchen ihre Entscheidungen durch Nutzer oder Quellen zu stützen. (Bspw. Style Guides, MCI Vorlesung etc.) Entsprechend verwendete Quellen sind sowohl in der Erläuterung als auch im Quellenverzeichnis zu dokumentieren.

[Überarbeitung]

Ab dem 2. Meilenstein dokumentieren Sie Veränderungen an einzelnen Artefakten. Dies kann stichpunktartig erfolgen. Achten Sie jedoch darauf, dass sich diese Änderungen auch in der Erläuterung wiederfinden. Bspw.:

-zusätzliche Persona Max aufgenommen, wir hatten die Perspektive dieses Nutzers übersehen

-Repositionierung des Login, 2 Tester hatten Probleme diesen Button zu finden, sie hätten ihn an einer anderen Position erwartet

Im Laufe des Projekts mag ein Artefakt so solide sein, dass eine Überarbeitung nicht mehr notwendig ist. In diesem Fall kopieren Sie einfach Artefakt und Erläuterung aus dem vorherigen Abschnitt in diesen und notieren „Keine Überarbeitung erfolgt“..

Szenario

Nach Ihrem erfolgreichen Studium sind Sie Mitarbeiter/-in der Softwareschmiede "Best Practice GmbH". Ihr Unternehmen konzentriert sich auf maßgeschneiderte Softwarelösungen und die Digitalisierung von Arbeitsprozessen.

Das neueste Projekt, welches gleichzeitig Ihren Einstieg in den Geschäftsalltag darstellt, erhielt Ihr Unternehmen durch die "TH-Köln".

Am Campus Gummersbach ist es mit entsprechenden Berechtigungen möglich, Transponder zum Öffnen von Räumen auszuleihen. Der Prozess zum Ausleihen, dem Prüfen von Berechtigungen, als auch dem Verleihen von Berechtigungen erfolgt dabei rein auf ausgedruckten Listen.

Die Raumverantwortlichen übermitteln hierzu Berechtigungen an die Pforte. Hier führen die Mitarbeiter/-innen an der Pforte eine Liste mit allen Schlüsseln und den dazugehörigen Personen, die die Berechtigung zum Ausleihen eines Transponders besitzen. Ein Transponder kann dabei mehrere Räume öffnen. Möchte eine Person nun einen Transponder ausleihen, ist es durch die Mitarbeiter/-innen an der Pforte notwendig die Person und seine entsprechende Berechtigung in den entsprechenden Listen zu überprüfen. Besitzt diese Person die entsprechende Berechtigung trägt er sich mit Datum, Uhrzeit und Namen in eine Verleihliste ein.

Die aktuelle Handhabung ist sehr Pflege intensiv und widerspricht den Leitkriterien guter Usability: Effektivität, Effizienz und Zufriedenheit.

Ihre Aufgabe ist es den Transponderverleih der TH Köln am Campus Gummersbach zu optimieren. Hierzu entwickeln Sie im Laufe des Projekts ein Konzept welches sie prototypisch implementieren und testen.

Meilenstein 2

Nutzermodellierung

Stakeholder Analyse

[Überarbeitetes Artefakt/e]

[TODO Erik: add introduction with reference to the interview, when, who, where was the interview]

Gatekeeper

He/she lends the transponders to the lenders.

For this they need the authorization lists from the room staff (the person in charge of rooms), to know to whom they can lend transponders.

They are also holding a register about who lends which transponder and the lending time.

He is interested in automating the authorization lists, because it takes way too long to look through and manage them per hand.

Persons in charge of rooms

They grant the lenders the timely limited permissions for the transponders.

They must give these to the gatekeeper.

They are interested in automating the system of delivering the lists to the gatekeeper.

Lenders

The lender can be any person. He can lend the transponder for which he has the authorization granted.

While he has the transponder he also has the alone responsibility for it and he must sign this with his name when lending a transponder at the gatekeeper.

To lend a transponder he must give his name and the transponder- or room number to the gatekeeper.

The lender is just interested in a fast and smooth lending process without having to give much data.

Central office in cologne

The office in cologne is responsible for maintenance of the transponders.

To this counts the programming, repairing and fabrication of the transponders.

[Erläuterung]

[Überarbeitung]

User Profiles

[Überarbeitetes Artefakt/e]

The user profiles introduce the users of the transponder-lending-system.

They state which relationship the users have to the product, which problems the system tries to solve and what the motivation for the system for every type of user is.

Gatekeeper

Motivation:

The current solution for the lending of transponders is associated with a very high administration effort.

Huge list must be looked up and the synchronisation for new permissions works very badly.

So, the gatekeeper has a hard time managing all these lists and the process occupies accordingly a lot of time.

The gatekeepers are hoping for a new solution, that makes all this easier, so they can concentrate on more important aspects of their work and don't need to struggle anymore with this huge amount of paperwork.

Persons in charge of rooms

These are professors and staff of the TH Köln at the Campus Gummersbach, who oversee rooms and can give permissions for transponders that open their rooms to other people.

Characteristics:

They are very busy and don't have much time.

Motivation:

They want to give and remove permissions for transponders in an easy and quick way. Permissions should be removed automatically after certain amount of time (e.g. half a year).

Lenders

There are three different types of lenders: 1. students, 2. professors and scientific staff and 3. external lenders.

All lenders are responsible for any loss or break of the transponder. There are differences how these user types use the system and which role the lending system has for them. These differences are depicted below.

Type A students: These are the primary users of the system. They usually identify themselves when they want to lend a transponder with their multicard. When they want to view their permissions for the lending of transponders they can log in to the system with their gmid-account (or campus-it account).

Type B professors and scientific staff: These are secondary users of the system. They use the same interface as the student, but they may have additional rights and options for using the system.

Type C external lenders: These are all the lenders that don't fit in the above categories. So that can be cleaning staff who need to access the rooms for cleaning or a craftsman who need to repair something. They don't use the interface for viewing their permissions for lending transponders. They either lend transponders ad hoc, e.g. a craftsman who needs to repair something just need the transponder one time and then no more, or they have a long-term permission, e.g. cleaning staff needs to access rooms for the time they are doing this job.

System Administrator

The system administrator is an expert user and it's a user that wasn't there beforehand but is needed for the system to work correctly. He/she is responsible for the data maintenance like adding, changing and removing transponders and users. He/she has a deep understanding of the lending-system and is also responsible for any exceptions and questions people have. Although he/she is an expert, they wish for an interface that is practical and can get their job done quickly, because they also have lots of other tasks in their job.

[Erläuterung]

The user profiles should have given us an overview over the users and a short introduction to each user and their characteristics (when important) and the motives.

As you can see, we did not mention the Central office in cologne from the *Steakholder Analysis* as a user. This is because the central office is no user and is therefore an anti-user. Our system only covers the lending process and the managing of permissions for transponders and not the maintenance and production of the transponders.

We also added an additional user that didn't appeared in the *Steakholder Analysis*, the system administrator. We need the administrator to keep the system intact. The system administrator doesn't use the product like all the users. The system doesn't simplify the work or is practical for the administrator. The administrator is rather part of the system itself. The administrator task is it to for example to add transponders to the system or to let a person in charge of rooms to be responsible for more rooms and therefore more transponders.
To accomplish these tasks, the administrator could tinker around with the systems databases itself because he/she is an expert. But that would be very inconvenient for the admin. Also, the system would be unstable and there is the chance to break something in the system. Therefore it would be helpful if the system would take the needs and goals of the system administrator into account and could provide interfaces (and logic) that helps the administrator. But these things would come with an extra cost for the design and development, so this is also a question of the budget.

[Überarbeitung]

- added introduction
- divided lenders in different types
- added system administrator
- added conclusion

Personas

[Überarbeitetes Artefakt/e]

"A persona, […] in user-centred design and marketing is a fictional character created to represent a user type…" (Wikipedia: Persona [user experience]).

Personas are depicted as individuals, but they serve as archetypes and therefore represent a class or a type of users with special needs and interaction with the product (c.f. Alan Cooper, Robert Reimann & David Cronin 2010, S. 105).

The personas are going to help us to understand the users that are using the system and to create the system for the user's needs and goals. To understand the user is critical for the design choices we will make in the design process later.

Here are our personas:

- Andreas Fischer (gatekeeper)
- Prof Dr. Sieglebud (person in charge of rooms)
- Sandra Meyer (master-student)

Andreas Fischer (gatekeeper)

Andreas Fischer works as a gatekeeper at the TH Köln in Gummersbach. In general, he really likes his job. He has a lot to do with people and there is always something going on. But he sometimes struggles with the paperwork he needs to do during his work day. Especially the lending of transponders causes him headaches sometimes. The permission lists need frequent editorial work, and this is very time consuming and error-prone. Also, when there is somebody who wants to lend a transponder, which happens quite often a day, he need to search in these huge lists and this is causing him stress, because people can be quite impatient sometimes. He works at the faculty for computer and engineering science and in general the building is technically very well equipped, so he wonders why there is no technical system that helps him with his problem. Luckily, he heard about a student project, that tries to come up with a design-solution for his problem. He hopes that the project will move in the right direction and that there will be something soon, that supports him at work, so he can concentrate on more important aspects of his work.

Andreas wants:

- to look up permissions for transponders and the availability of transponders uncomplicated and quickly
- he doesn’t want to manage the permission lists by hand
- he wants a system that is less error-prone and let him quickly lend transponders to people

Prof. Dr. Siegelbud

Prof. Dr. Siegelbud is a professor for communication and network technology at the TH Köln in Gummersbach. This semester he supervises the master-project of some students. The students need access to the laboratories, so he needs to give them permissions to lend the corresponding transponders that can open the rooms. So, he needs to go down to the gatekeeper, where he needs to authorize and give the gatekeeper the list of students, that need access to the laboratories. He is very busy with other things, so this is quite inconvenient for him. His to-do-list has already stacked up higher than he would normally allow it. It would be much easier if he just could give the permissions straight from his desk.

Prof. Dr. Siegelbud wants:

- to give permissions for transponders (rooms) to people right at his desk (or anywhere)
- to look up the permissions he gave (also at his desk or anywhere)
- to remove permissions in the same easy manner

Sandra Meyer (master-student)

Sandra Meyer is a master-student and currently in her third semester. For her master-project about IT-Security she needs to access the laboratories several times a week. She has spoken with her supervisor Prof. Dr. Siegelbud, who already gave her access. Every time she goes in the laboratories she goes to the gatekeeper in the lobby. There she waits for the gatekeeper to check her permissions and give her the transponder, that let her open the room. Quite often the transponder was already lent, which is impractical for her because the laboratories are in the opposite direction of the lobby, so it always takes her a while to get her there. The lending of transponders also takes a while. She feels sorry for the gatekeeper, because he needs to work through some confusing lists, but she also wishes sometimes, that the process would be faster. It's not much time the process takes, but it is stacking up and she really could use this time for her studies.

Sandra wants:

- that the process of lending a transponder is quicker
- that she can look beforehand whether a transponder is lent or not

[Erläuterung]

Although there are more than 3 user types, we decided to create only 3 personas, on which we will focus, and which are the most important user types for the system.

The persona, that represents the gatekeeper is Andreas Fischer.

The persons in charge of rooms are represented by the professor Dr. Siegelbud.

And the lenders are represented by Susi Meyer the master-student.

For the lenders we chose to only represent the most important of the three lender-types (students, professors and scientific staff, external, c.f. User Profiles – Lenders), because we identified the students as the primary users of the system and professors and scientific staff as secondary users, because they use the interface, that we will design, just like the students, but they have further needs, so there are some more requirements.
The external lenders would belong to *Served Personas*, which are personas-types that don't use the system directly but are affected by the system (c.f. Alan Cooper, Robert Reimann & David Cronin 2010, S. 105), because they don't interact with the system directly unlike the other lenders, who have an interface for seeing their permissions, and because they are only affected by it during the lending process.
Therefore, we created only one persona 'Susi Meyer' because as a *Primary Persona*, it's the user type we should concentrate on the most.

We also excluded the administrator for the personas, because we decided that we won't focus on him by now, due to time constraints. We know that the administrator should not be neglected, because he/she plays a crucial role for the maintenance of the system and will probably work the most with the system, but as an expert the system administrator will be able to do his/her work without well-designed interfaces specially designed for him. A well-designed interface for the administrator would be very nice, but the admin is not that high in our priorities, to justify the costs. But if enough budget is left, we will definitely create an interface for the admin.

We tried to tie the stories of the personas a bit together, so they are all part of a big picture. So, the professor Siegelbud is the supervisor for Sabine Meyer's master project, and he needs to grant her access to the laboratories. Sabine Meyer show empathy with the gatekeeper because of the complex non-automated system the gatekeeper must deal with, but also shows her dissatisfaction with the current solution that takes up a lot of her valuable time.

[Überarbeitung]

- added introduction
- added explanation

User Stories

[Überarbeitetes Artefakt/e]

A user story is a description of one or more features of a software system, written in the perspective of the user. They are part of the definition of requirements.

They help us to know, what the system should be capable of. they are an excellent method to define our requirements in our user-centred design process, because they also state why a specific function is important for users and what the goal is the user wants to achieve.

Here we created user stories for our three personas, we identified and who are most important to us. The personas we created have special needs and tasks they need to do during their day. Now here is described which needs the system must satisfy and what the users should be able to do to achieve their goals.

Gatekeeper

1. As a gatekeeper, I can check the availability of a specific transponders, so that I can give the person, who wants to lend the transponder, information, so they know whether it is already lent or not.

2. As a gatekeeper, I can check whether a person has a permission to lend a certain transponder or not, in an easy and uncomplicated way, so that the process is quick, the person doesn't need to wait long and can get the transponder and I can go on with my work.

3. As a gatekeeper, I can lend transponders to people without much administrative effort, so I don't have to manage huge lists and the persons who want the transponders are happy.

4. As a gatekeeper, I can view a list of all transponders with their status and can sort and filter them by their status, lending time and other data, so that I can create an individual overview of all transponders, which let me find the information I need.

5. As a gatekeeper, I get a message when a transponder is missing for too long so that I am informed about it and can perform the necessary actions, like contacting the lender or inform the authority.

Persons in charge of rooms

1. As a person in charge of rooms, I can give/remove permissions to other people for lending transponders to open the rooms I oversee, in an easy and quick way, so I don't have to go anywhere to do this and persons I trust can open the rooms, I am responsible for.
2. As a person in charge of a rooms, I can also give/remove permissions to a group of people that share common traits like students studying *this* in *that* semester or students attending a special lecture or course, so that I don’t have to grant permissions for every single person of a group which would be annoying and time-consuming.
3. As a person in charge of rooms, I can view a list of persons to whom I granted a permission, so that I have an overview which persons can access the rooms, I am responsible for.
4. As a person in charge of rooms, I can view details for each permission I gave, like which rooms can be opened by the transponder, when a permission was granted or when it will expire, so that I can use this information to plan and to reason my decisions.
5. As a person in charge of rooms, I can sort and filter the list of persons (cf. 3.) by room number, times I granted the permission or person groups (and more), so that I can create my individual overview and can find and access the information needed quickly.
6. As a person in charge of rooms, I can view open permission-requests which I can then either accept or deny or ignore in which case it is denied automatically after a specific amount of time, so that I can react on permission-requests which is one of my responsibilities as a person in charge of a rooms.
7. As a person in charge of rooms, I can view the rooms and the corresponding transponders I have the responsibility for, so I know what I am responsible for.

Lenders

1. As a lender, I can lend a transponder at the gatekeeper’s office without much effort, so I can open rooms with it.
2. As a lender, I can look beforehand whether a transponder is already lent or not, so I save time and I only lend a transponder if it is available.
3. As a lender, I can view a list of all permission for lending transponders, so that I have an overview which transponder I can lend which helps me to plan my actions.
4. As a lender, I can view details for each permission, that is granted to me, with information like which rooms I can open with the transponder, when the permission was granted, who

granted the permission, when it will expire, so that I can use this information to plan, e.g. I know that the permission will expire in two weeks and I can renew the permission.

5. As a lender I can filter and sort the list of my permission for lending transponders by the permission data (like grant-time, expire-time, etc.), so that I can create an individual overview and can quickly find the information needed.
6. As a lender I can enter a room number or a transponder number and the system will then tell me if I have a permission for the transponder or not, so that I can quickly check whether I have a permission for a specific transponder / room, so that I can ask for a permission if I haven't one already.
7. As a lender I can ask for a permission for a transponder, so that I have the chance to lend transponders.
8. As I lender I get a message (email) when a request for a permission was accepted or denied, I got a new permission, or a permission was removed, so that I am informed, and I always know what changed, so that I can adjust to the situation.
9. As a lender I want that changes of my permissions are reflected in the application (e.g. a short message when I launch the app and I got a new permission), so that I directly know what changed, so that I don't need to actively look for the change.

All Users

1. As a user I can login with my user-id and password in the system, so that I am authorized with my authentication to do the things I am allowed to do with the system.

[Erläuterung]

Here we can see which role every type of user has and what functionality they need in the system, to achieve their goals.

The three user types all have a different view on the data the system manages. What attracts attention is that all the users have options to sort and filter the data. For the gatekeepers and the persons in charge of rooms, this makes sense because they need to manage huge lists and need to find information as efficiently as possible. But for the lender's it is questionable that it is a necessary option to filter and sort the permissions. It depends on how much permissions a lender acquires and how many rooms there are and how the lenders use this service overall. If a lender only has less than six permissions, it would not be necessary, to add these features. We don't have answers on these questions yet and so our decision was to think ahead and add the filter-, and sort-option for now.

What the user stories don't show is the context and in which relation the user stories stand to each other. But this problem is solved by the next step the *Hierarchical Task Analysis*, that will show of which smaller tasks a feature is composed.

But neither the user stories, nor the *Hierarchical Task Analysis* show how the users interact with each other. Therefore, I will briefly point it out at this point:

The lenders can ask the persons in charge for permissions to lend a transponder. This option is integrated in the user interface.
When a request was sent, the persons in charge of rooms then get a message in their application, on which they can respond to. They can either accept the request or refuse it. Or they can ignore it in which case it will be denied automatically after a specified time. When the person in charge of room responded on the request, the lender gets an email that states whether their request was accepted or denied. The lenders also gets emails if any other changes of their permissions happened. These changes should also be shown in the app.
With the permissions the lender can know lend the transponders at the gatekeeper's office.

[Überarbeitung]

- added user stories for gatekeepers 4-5
- added user stories for persons in charge of rooms 2-7
- added user stories for lenders 3-8
- added "All-Users"-section and user story for login
- added introduction
- added conclusion

Top Level User Tasks

[Überarbeitetes Artefakt/e]

The Top Level User Tasks should give us a comprehensive insight into the main tasks our users will try to fulfil when using our application. Each Top Level User Task represents the root of a task tree in the Hierarchical Task Analysis. All tasks combined should make every User Story possible to accomplish.

0. lend transponder
1. give/remove permission
2. verify lender's permission
3. check for available transponder
4. check if I have permission to room/transponder
5. ask for permission

[Erläuterung]

This list is a very raw representation of our users' tasks, but gives us a great starting point for our Hierarchical Task Analysis.

[Überarbeitung]

- added description
- added conclusion
- added tasks 3 – 5

Hierarchical Task Analysis

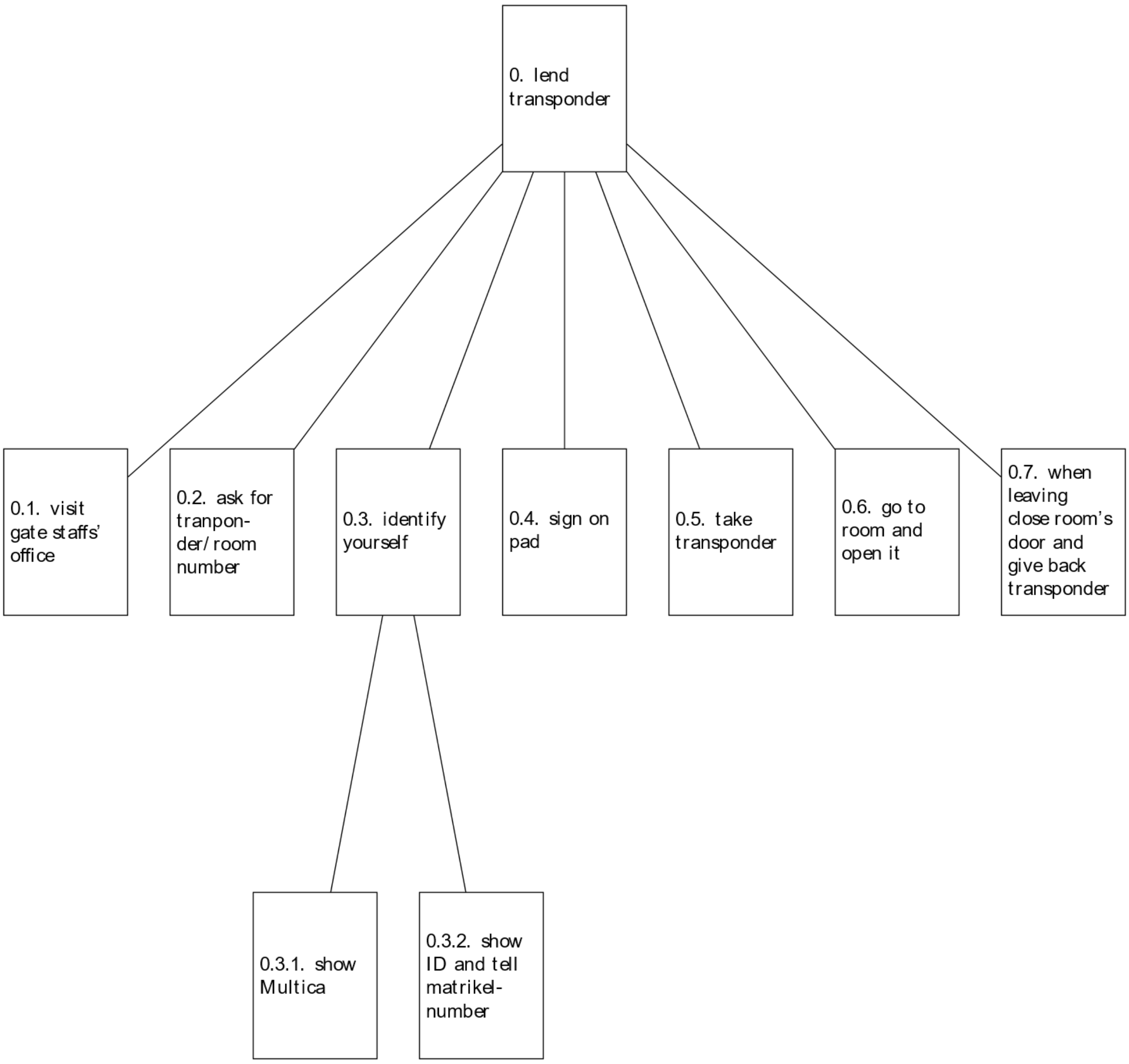
[Überarbeitetes Artefakt/e]

Our Top Level User Tasks decomposed in smaller, hierarchical tasks/subtasks and actions a user will perform using our application. Furthermore, it allows us to specify plans, which helps us to model state (e.g. transponder not available, lender has no permission), conditions (e.g. only if lender has permission he can lend a transponder, else the gatekeeper resents the request) and decisions (e.g. lender decides to use his/her Multica as identification), which should make it easier to implement the different tasks/subtasks in a safe and useful manner later in the development process.

0. lend transponder

Plan 0.3.1: do 0.1-0.2. If you choose to identify with your Multica do 0.3.1 and continue with 0.4.

Plan 0.3.2: do 0.1-0.2. If you choose to identify with your ID do 0.3.2 and continue with 0.4.

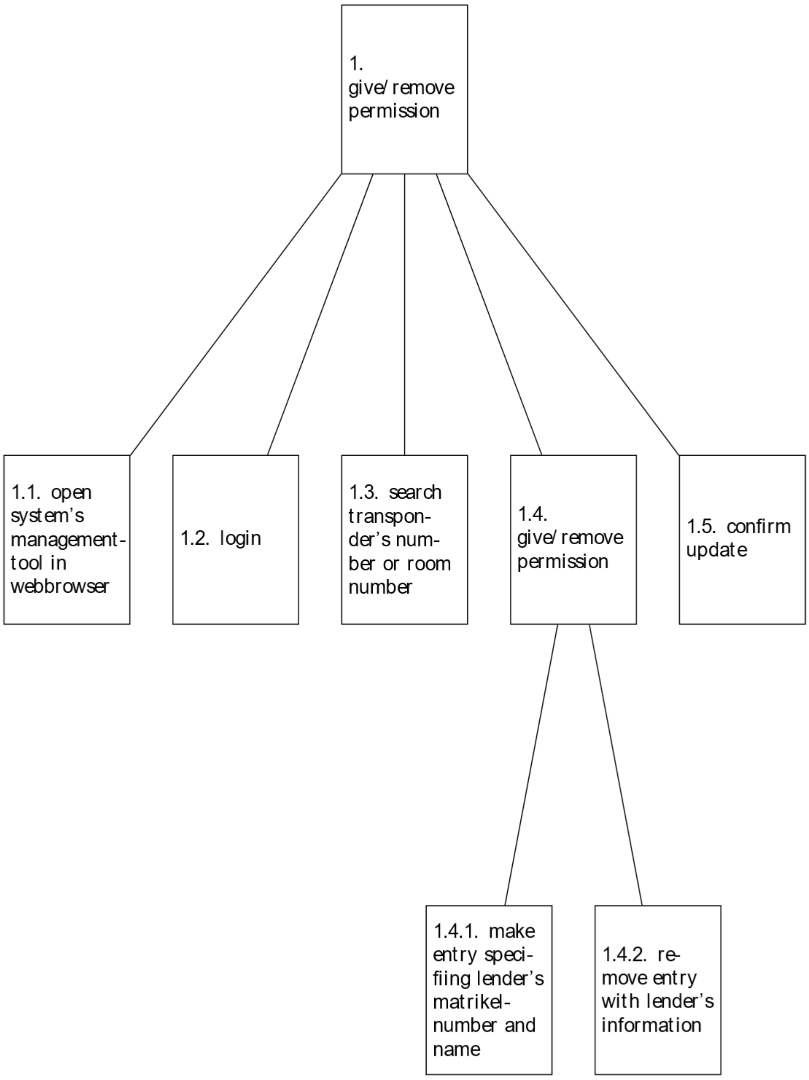

0. lend transponder
0.1. visit gate staffs' office
0.2. ask for tranpon-der/ room number
0.3. identify yourself
0.4. sign on pad
0.5. take transponder
0.6. go to room and open it
0.7. when leaving close room's door and give back transponder
0.3.1. show Multica
0.3.2. show ID and tell matrikel-number

The main task a lender does. He/she wants to use this system to gain access to a room he/she has permission to use.

1. give / remove permission

Plan 1.4.1: do 1.1-1.3. If you want to give permission do 1.4.1 and continue with 1.5.

Plan 1.4.2: do 1.1-1.3. If you want to remove permission do 1.4.2 and continue with 1.5.


1. give/ remove permission
1.1. open system's management-tool in webbrowser
1.2. login
1.3. search transponder's number or room number
1.4. give/ remove permission
1.5. confirm update
1.4.1. make entry specifiing lender's matrikel-number and name
1.4.2. remove entry with lender's information

If a person in charge of a room or transponder wants to give/take permission to/from a lender he/she has to complete this task.

2. verify lender’s permission

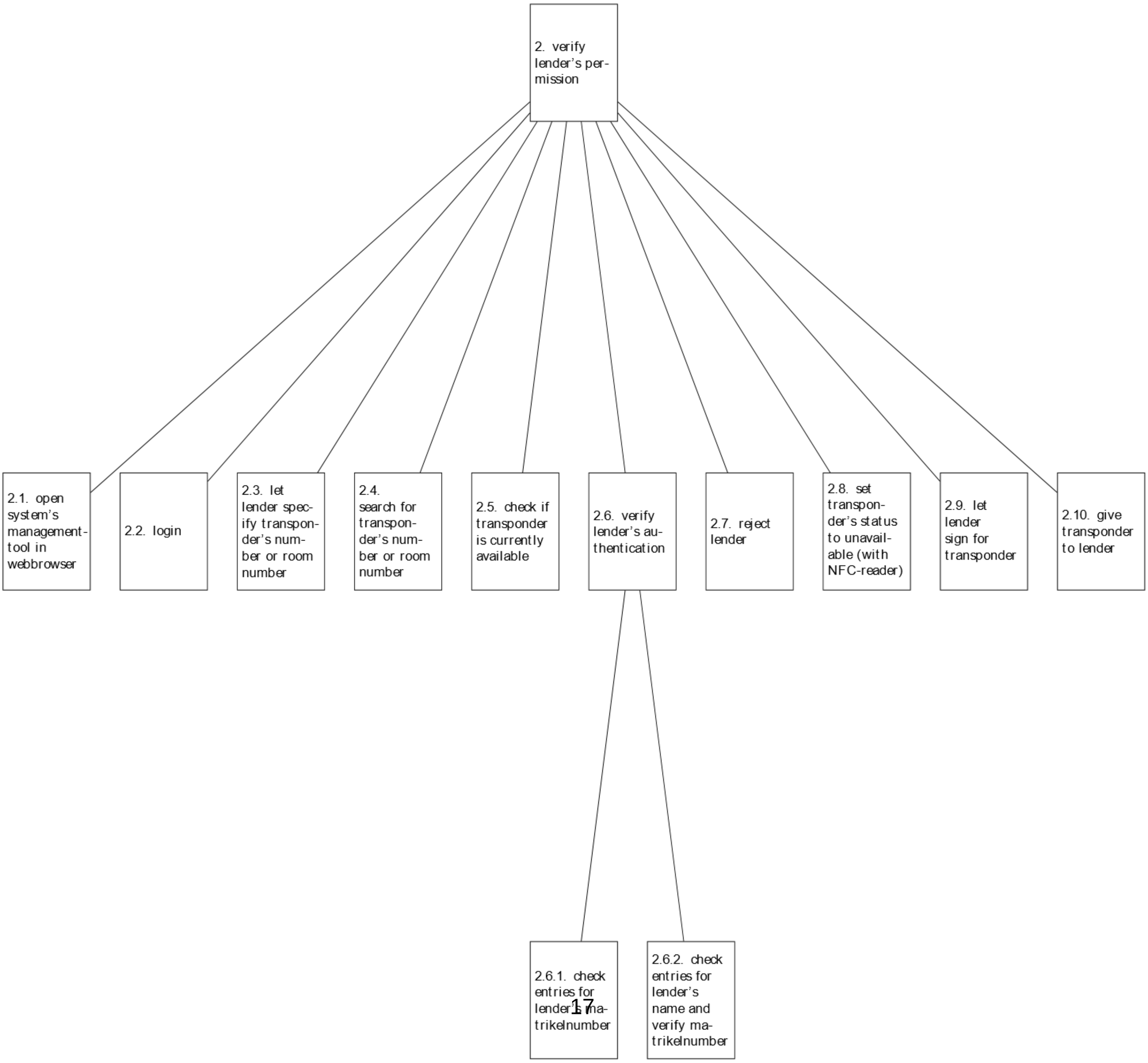
Plan 2.5.1: do 2.1-2.5. If transponder is available continue with 2.6.

Plan 2.5.2: do 2.1-2.5. If transponder is not available do 2.7.

Plan 2.6.1: do 2.1-2.5. If the lender uses his Multica as identification do 2.6.1 and continue with 2.8.

Plan 2.6.2: do 2.1-2.5. If the lender uses his ID as identification do 2.6.2 and continue with 2.8.

This task is done by a member of the gatekeeping staff. It basically runs parallel to the


2. verify lender’s permission
2.1. open system’s management-tool in webbrowser
2.2. login
2.3. let lender specify transponder’s number or room number
2.4. search for transponder’s number or room number
2.5. check if transponder is currently available
2.6. verify lender’s authentication
2.7. reject lender
2.8. set transponder’s status to unavailable (with NFC-reader)
2.9. let lender sign for transponder
2.10. give transponder to lender
2.6.1. check entries for lender’s matrikelnumber
2.6.2. check entries for lender’s name and verify matrikelnumber

lender's task "0. lend transponder".

Since one of our main goals is to keep the university's property safe, this task of verification is utterly important. We can not allow that somebody permissionless, unverified or without the system's/the person in charge of the room's knowledge gains access to a room.

This is the reason why every lending process gets protocolled, so we can always keep track of who has and has had a transponder. Now, if some damage gets reported, we always know who had access during the approximate time period the damage was done.

We want to automate every task as much as possible and make the task for every user as seamless as possible, too, so we thought it would be easiest if we would introduce NFC (Near-Field Communication) to this particular task.

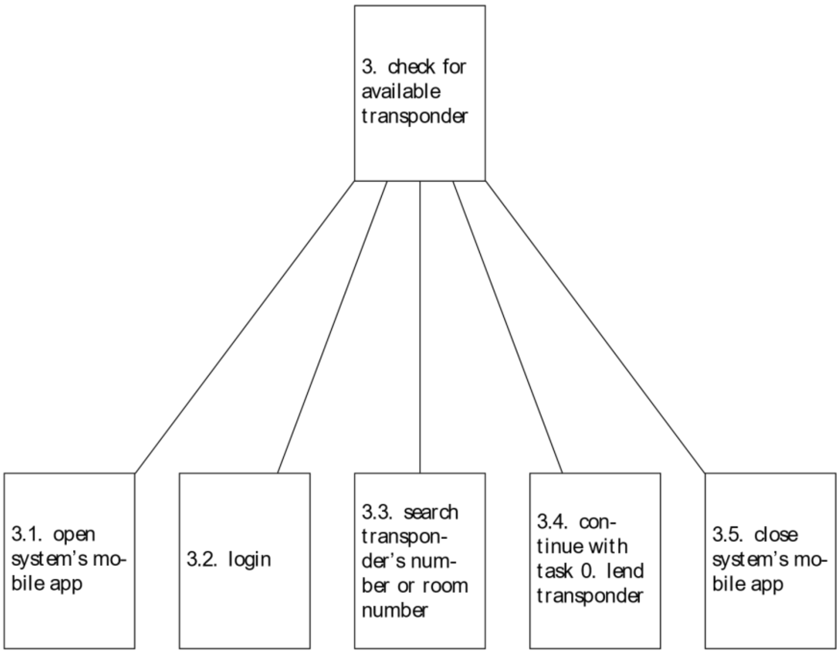
We want a NFC-reading device connected to the workstation the gatekeeper uses, which we can access with our application running on this workstation. Now, if the gatekeeper hands out the transponder, he/she just has to hold it against the NFC-reading device and our app knows which transponder was just lent. If the lender used his Multica to identify himself he used the same NFC-reading device, so our app can combine these two information and we know which user lent the transponder. The gatekeeper doesn't has to make interactions with our application interface that could lead to errors, for example giving out a different transponder than the one selected in the application.

If the lender used his ID for identification, the gatekeeper presses a button on his interface which tells the app who is the lender. After that he holds the transponder against the NFC-reading device and the app knows which transponder was lent.

We think this is easier, faster and safer than the gatekeeper having to always manually specify the transponder in his application interface opened in his browser.

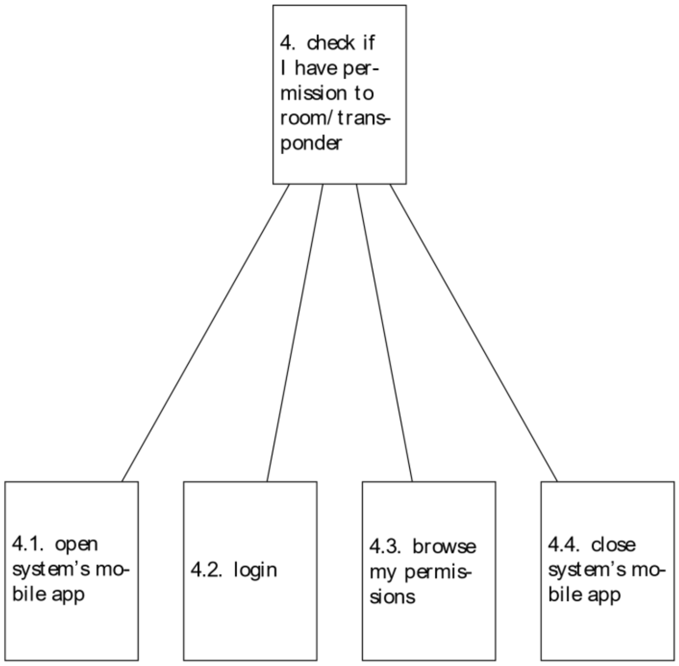
3. check for available transponder

Plan 3.4: do 3.1-3.3. If transponder is available continue with 3.4, else continue with 3.5.


3. check for available transponder
3.1. open system's mobile app
3.2. login
3.3. search transponder's number or room number
3.4. continue with task 0. lend transponder
3.5. close system's mobile app

A lender can check if his desired transponder or a transponder to his desired room is available using our mobile application. The lender can look up his permissions and filter them. Since our system knows all the time if a transponder is available or not, it can give the lender this information.

4. Check if I have a permission to room/transponder

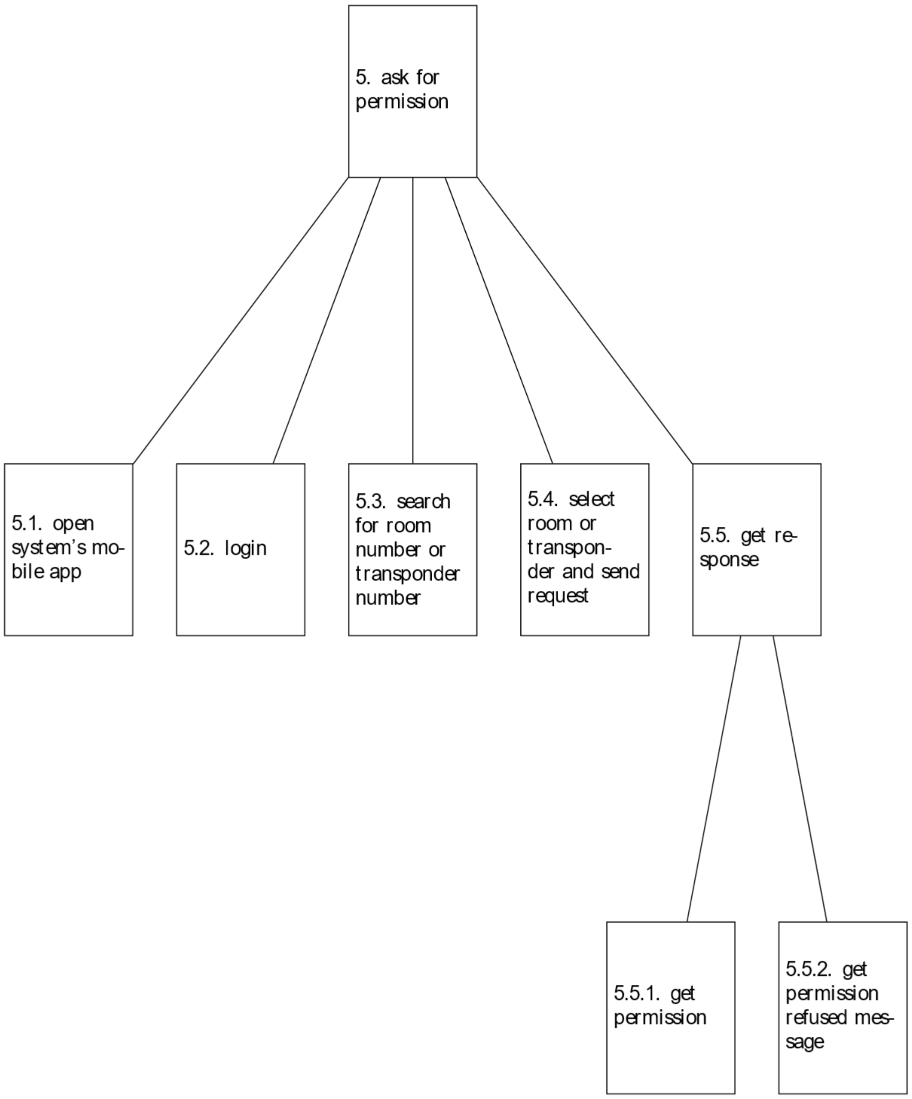

4. check if I have permission to room/transponder
4.1. open system's mobile app
4.2. login
4.3. browse my permissions
4.4. close system's mobile app

In case a lender wants to check if he has permission to a room or transponder. Furthermore he can see when his permission expires.

5. Ask for permission

Plan 5.5.1: do 5.1-5.4. If permission is granted continue with 5.5.1.

Plan 5.5.2: do 5.1-5.4. If permission is refused continue with 5.5.2.


5. ask for permission
5.1. open system's mobile app
5.2. login
5.3. search for room number or transponder number
5.4. select room or transponder and send request
5.5. get response
5.5.1. get permission
5.5.2. get permission refused message

A lender can use our application for making permission requests. That means he/she can look up a room or transponder and ask for a permission, which the person in charge of this room or transponder can either grant or refuse. Either way the lender gets informed by our application about whether the request is granted or refused.

[Erläuterung]

The Hierarchical Task Analysis gave us a great entry point and some insights to how our application could be structured (it's components, interfaces, hardware, technologies, etc.).

[Überarbeitung]

- changed from text representation to block diagram
- added tasks 3 – 5
- added description
- added conclusion

Rapid Prototyping

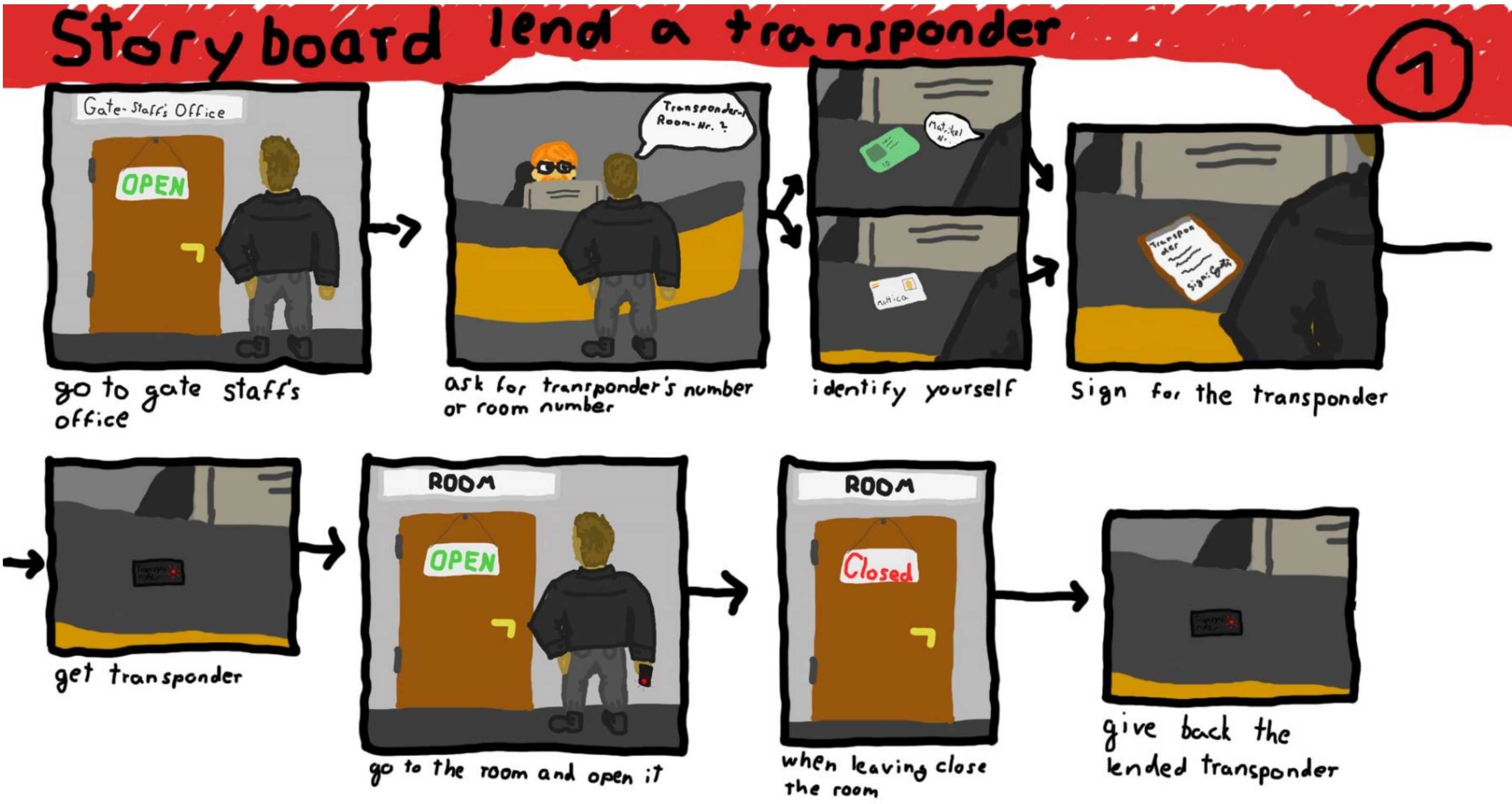
Story Board

[Artefakt/e]

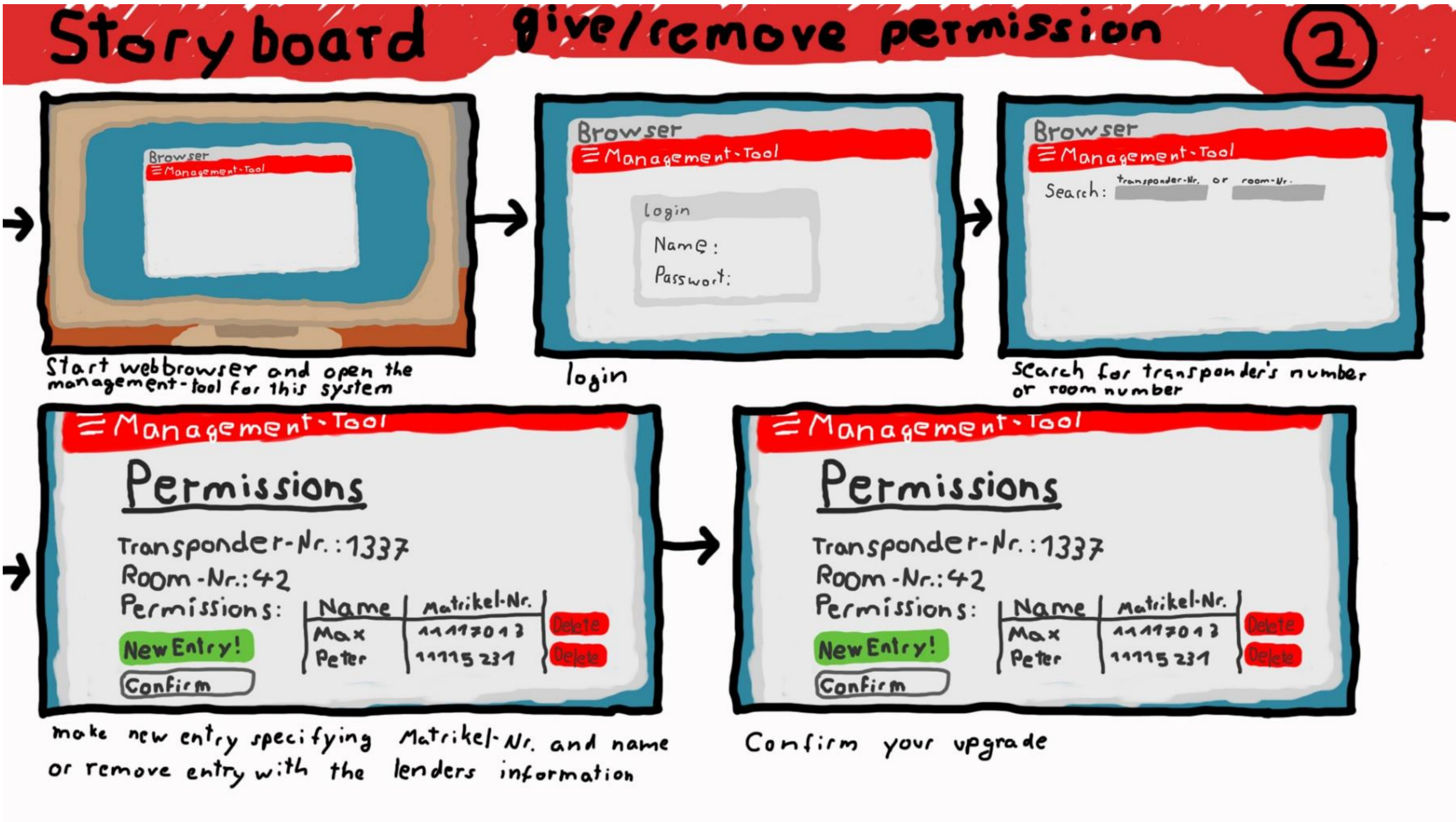
We created story boards for the first three top level user tasks of the HTA. The story boards present the results of the HTA visual appealing. They show sketches on how we imagine the process to happen and which interaction is involved.

They give also a first idea how we want to design the interfaces and what is in these aspects important for us.

0. lend a transponder


Storyboard lend a transponder
1
Gate-Staff's Office
OPEN
go to gate staff's office
Transponder-/Room-Nr. ?
ask for transponder's number or room number
Matrikel Nr.
identify yourself
Sign for the transponder
get transponder
ROOM
OPEN
go to the room and open it
ROOM
Closed
when leaving close the room
give back the lended transponder

1. give / remove permission



Start webbrowser and open the management-tool for this system

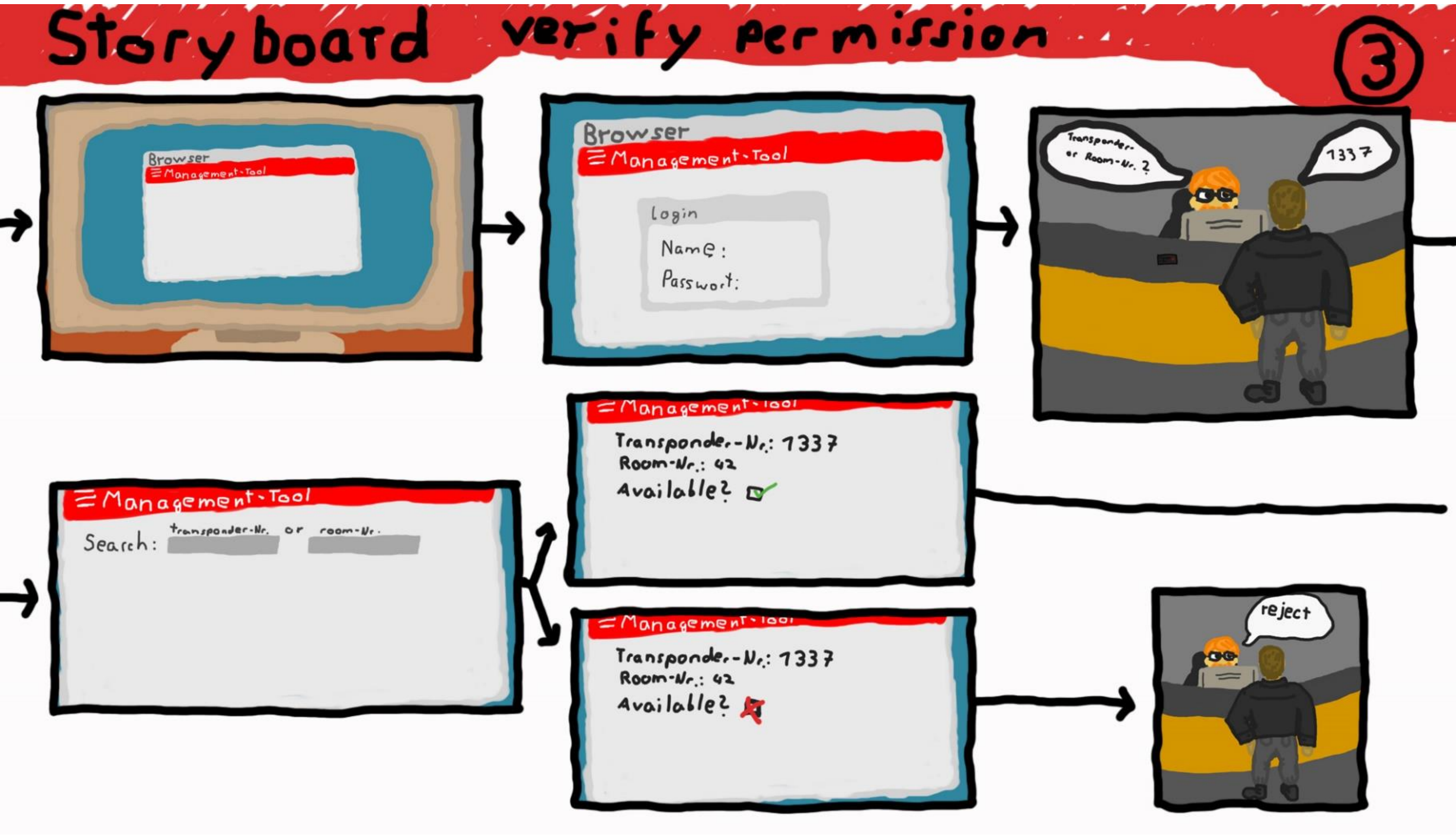
login

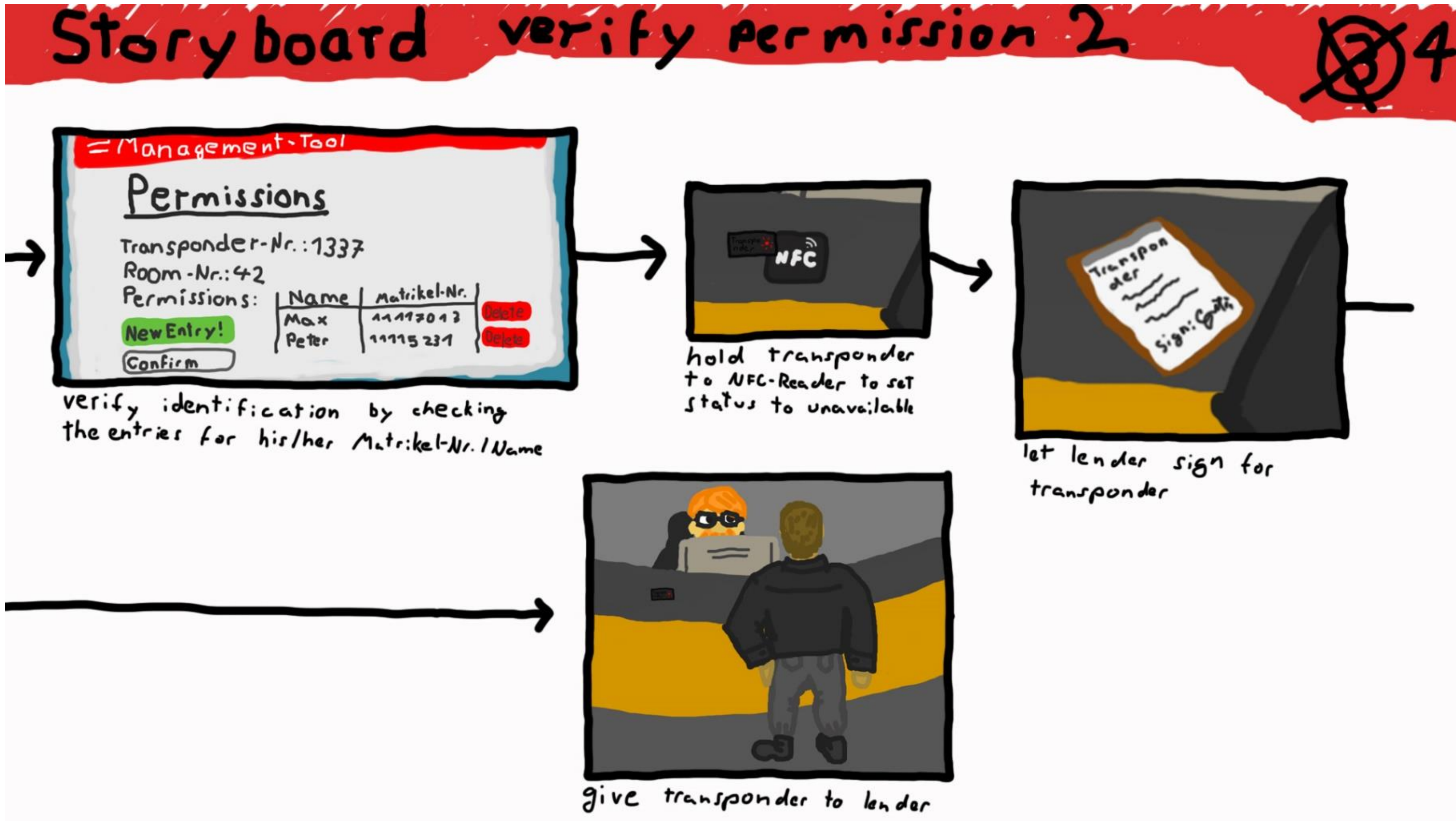
Search for transponder's number or room number

make new entry specifying Matrikel-Nr. and name or remove entry with the lenders information

Confirm your upgrade

2. verify lender’s permission




Storyboard verify permission 2
~~3~~ 4
Management-Tool
Permissions
Transponder-Nr.: 1337
Room-Nr.: 42
Permissions:
Name | Matrikel-Nr.
Max | 11117013
Peter | 11115231
Delete
Delete
New Entry!
Confirm
verify identification by checking the entries for his/her Matrikel-Nr. / Name
NFC
hold transponder to NFC-Reader to set status to unavailable
Transponder
Sign:
let lender sign for transponder
give transponder to lender

[Erläuterung]

Wireframes

[Artefakt/e]

[Erläuterung]

Paperprototyping

[Artefakt/e]

[Erläuterung]

Ergebnisse des Paperprototypings

[Artefakt/e]

[Erläuterung]

Quellenverzeichnis

Literatur: [Autor; Tite;, Verlag/Veröffentlichung]

Alan Cooper, Robert Reimann, David Cronin; About Face. Interface und Interaction Design, 1. Aufl. – Heidelberg [u.a.] : mitp, 2010

Web: [Autor/HomePage; URL: ; Stand: Datum]

Wikipedia: Persona (user experience);
https://en.wikipedia.org/wiki/Persona_(user_experience); cited 23.05.2018