SAPIENZA
UNIVERSITÀ DI ROMA

Master of Science in Engineering

in Computer Science

a.y. 2018/2019 | 2nd semester

Human Computer Interaction

Submitted to:

Tiziana Catarci

Valeria Mirabella

Presented By :

Sinibaldi Daniele

Trapasso Alessandro

Index

1. Abstract

This project is about the development of a mobile app called WIMSapp (Where Is My Seat application) whose main purpose is to provide an easy way for the students of finding places where they can study in their free time.

Actually, it isn't unusual for a student that arrives in a new university or a new building of his own university to remain without a quiet place where he can study only because he doesn't know the new university or buildings where he is.

At the same time, even if a student knows very well his own university and he knows where he can find libraries or study rooms, it isn't unusual that he can find these well known places already full of other people.

Because of these and other more reasons we thought about a solution for these poor students: and so WIMSapp was born.

Our app has got a simple solution for the first problem: indeed, it contains a database full of study rooms and libraries, divided on the base of where they are located. Rooms are divided with respect to the different buildings where they are and the users can check which rooms are near him through a textual menu divided on the base of locations or through a map that shows him the buildings. Moreover, for each room there are information about its opening and closing times, about how many seats it offers and about its location. For each room WIMSapp offers also some directions to reach it.

As it was said, even if a student knows where he can go to study, it isn't sure that he can find a seat in his favorite location: for this reason our app offers a user report based system to share information about how many seats are available in the different rooms and libraries; thanks to this system a student can have an idea of how much probably he will find a seat where he would like, simply by selecting the room he was thinking about and by checking the room color on the app: every room will have a color (such as green, yellow or red) on the base of its state of occupation.

Finally, another important aspect is that we developed the idea of this application by considering the user centered design approach, by focusing on the students as the main target users.

2. Introduction

2.1 Overview

The project has as main target one kind of user: the student. This kind of user is discussed in section 2.2, where some possible and representative scenarios and the analysis of the requirements are shown.
In section 2.3 a competitor analysis is presented, and this is very useful to know the main aspects of the market where the final product will operate.
Then, in chapter 3 there is the development of the work plan adopted, divided in phases in which some prototypes are designed, implemented and finally evaluated, by repeating these stages in an iterative way until the final product is reached.
In chapter 4 the design of the possible behaviors and actions to be done by the users is discussed thanks to the use of some task modelling tools like HTA and STN.
In chapter 5 the implementation of the first prototype from the initial design is shown. In the same chapter there is the application of an expert-based evaluation technique to the prototype, done to come up with the second prototype.
In chapter 6 the implementation of the second prototype after the expert-based evaluation is shown. In the same chapter there is the application of a user-based evaluation technique to the prototype, done to come up with the third prototype.
In chapter 7 there is the implementation of the third prototype after the user-based evaluation. As before, in the same chapter there is the application of another user-based evaluation technique to the prototype, done to obtain the final product.
In chapter 8 there is the presentation of the final product and its features.
In chapter 9 the conclusion of the work and the possible future work are presented.

2.2 The student

Here below there is the presentation of a general user profile and some possible personas and scenarios that could be of interest for this application.

2.2.1 User profile, personas and scenarios

User profile:

Age: 18 – 30

Gender: 58% male – 42% female

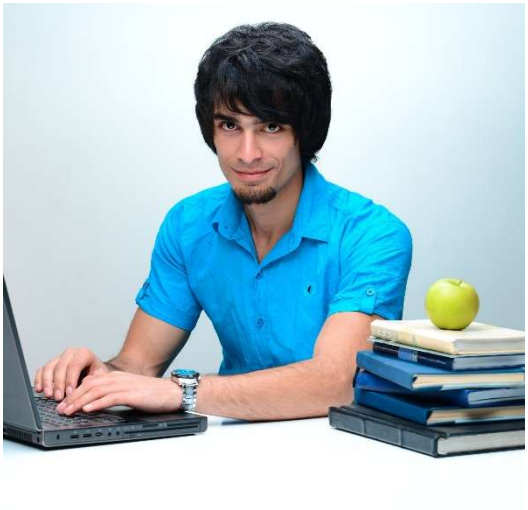
Job title: university student (Bachelor, Master, PhD or higher)

expertises: smartphone, pc, internet

Location: Italy, Rome

Income: 0€ - 1000€

Persona: Mark



Mark is an Erasmus student from England that will remain in Italy for some months and he arrived to University Sapienza of Rome only few time ago. He doesn't know well the new university and its buildings yet. He usually uses a mobile 4G internet connection to browse internet on his phone and to use the applications he installed on it.

Scenario:

Mark has got a couple of hours between two lessons and he would like to study a bit but he has got a problem: he would like to find a quiet place to study but he doesn't know well the place where he is and he hasn't got enough time to go around by searching an empty study room because the time he has got before the next lesson is limited. So, he decides to use the WIMSapp he installed on his smartphone the day before for curiosity and he finds the perfect study room for him on the app map, just near him.

Persona: Paola

Paola is a university student from the University of Rome La Sapienza that is at her first academic year. She comes from Rome. She doesn't know well the university and its buildings yet because she is starting her new university study course for the first time. She usually uses a mobile 3G internet connection to browse internet on her phone and to use the applications she installed on it.

Scenario:

Paola has a free afternoon and she is at the university because she had lessons in the morning. She would like to study during this her free time but she has got a problem: she would like to find a quiet place to study but she doesn't know well the place where she is and she doesn't want to go around blindly by searching an empty study room. So, she decides to use the WIMSapp that a friend suggested her and she finds the perfect study room for her on the app by searching in the list of the available study rooms and libraries that are in the same building where she is.

Persona: Antonio

Antonio is a university student from the University of Rome La Sapienza that is attempting the second year of his master degree. He comes from a city a bit far from Rome. He knows very well his university and its buildings because he attempts lessons in the same building since two years ago. He usually uses a mobile 4G internet connection to browse internet on his phone and to use the applications he installed on it.

Scenario

Antonio has a lesson in the afternoon and because of the fact he comes from a city a bit far from Rome, he decides to go to Rome already in the morning to be directly at Rome when he will have to go to his lesson. So, he needs a quiet place because he decided to study while he is waiting for his afternoon lesson. He knows very well his university and he has got a favorite library to study but he doesn't know if he will find a free seat in this library because there are always a lot of people there. So, he decides to use the WIMSapp he installed on his smartphone some months ago and that he tried several times yet, and he discovers that luckily this day there are a lot of free seats in his favorite library.

2.2.2 Requirement Analysis

We asked to some students to answer to a questionnaire to collect some information which can have helped us in a better refinement of some aspects of our application. By spreading this questionnaire also on some socials, we succeeded in collecting information from about 60 students.

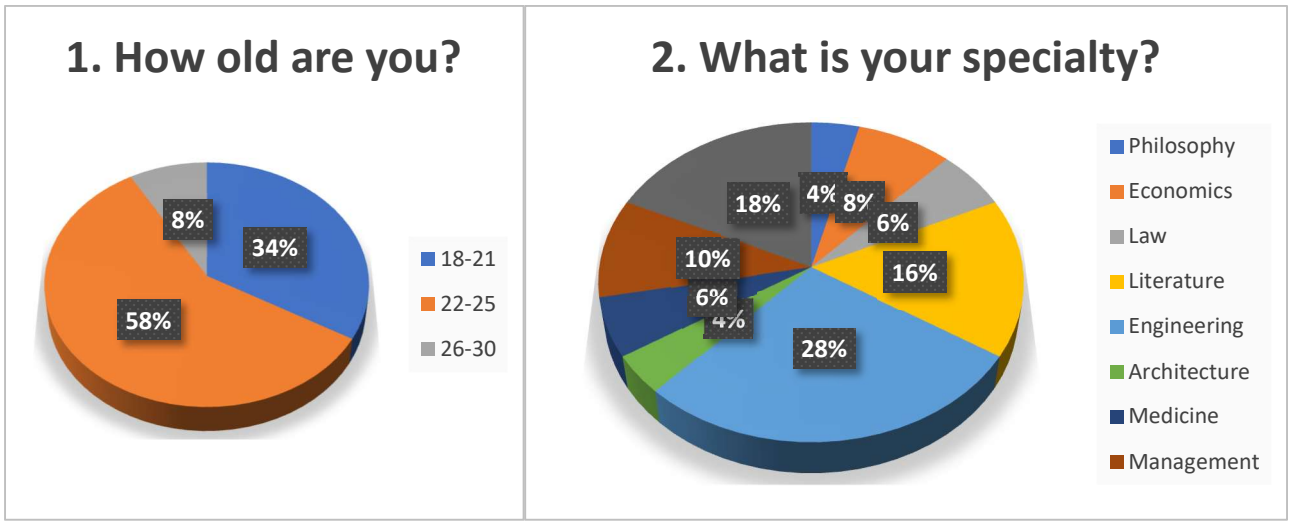
Questionnaire

The questionnaire that we gave to these students was the following one:

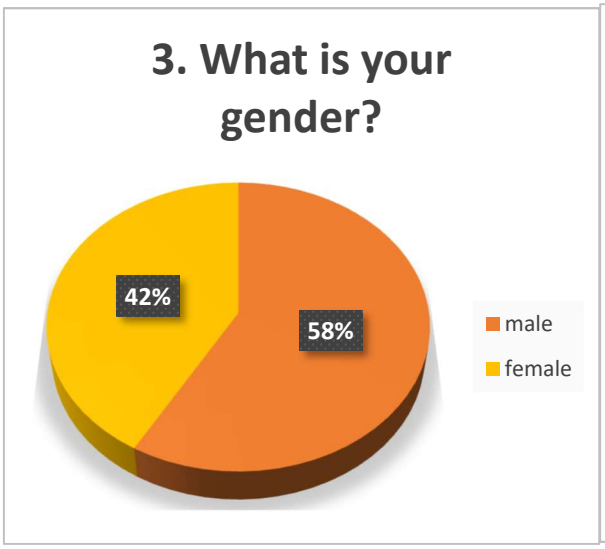
N	Question	Possible answers
1	How old are you?	A number for the age
2	What is your specialty?	Open answer
3	What is your gender?	Male / Female / Other
4	What is your level of education?	Bachelor / Master / PhD or higher
5	How often do you have troubles in finding a place to study?	Never / Sometimes / Always / Often
6	Would you use an app to find study rooms and to monitor their occupation status?	I don't think so / Maybe no / Maybe yes / Sure

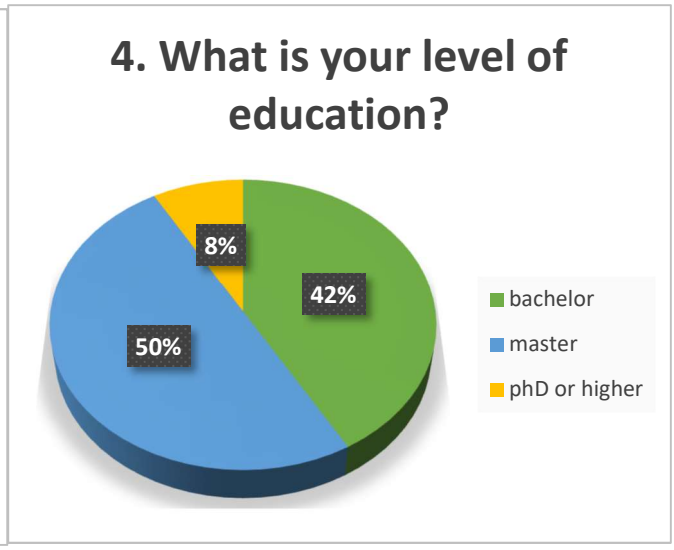
7	Would you prefer search a room through a menu, a map or both?	Menu / Map / Both
8	Do you think it would be useful if the rooms were divided with respect to their locations?	Yes /Probably yes / Probably no / No
9	Do you think that would be useful to know for each room on which floor of its building it is?	Yes /Probably yes / Probably no / No
10	Do you think that would be useful to know for each room its opening and closing times?	Yes /Probably yes / Probably no / No
11	Do you think that would be useful to know for each room how many seats it has?	Yes /Probably yes / Probably no / No
12	Would you help other students like you by reporting the occupation status of a room if you were in it?	Yes /Probably yes / Probably no / No
13	Would you help the app developers by reporting new rooms which aren't in the app database yet?	Yes /Probably yes / Probably no / No
14	Would you use an app which doesn't allow you to delete your account?	Yes /Probably yes / Probably no / No
15	Do you think it would be useful a home menu that allows you to reach all the important parts of the app in a simple way?	Yes /Probably yes / Probably no / No
16	Do you think it is important to log out from an app installed on your device instead of remaining logged in?	Yes /Probably yes / Probably no / No
17	Do you think it would be easier for you to report an occupation status through colors (green, yellow, red) or through a percentage?	Colors / Percentage / It's the same

Results analysis

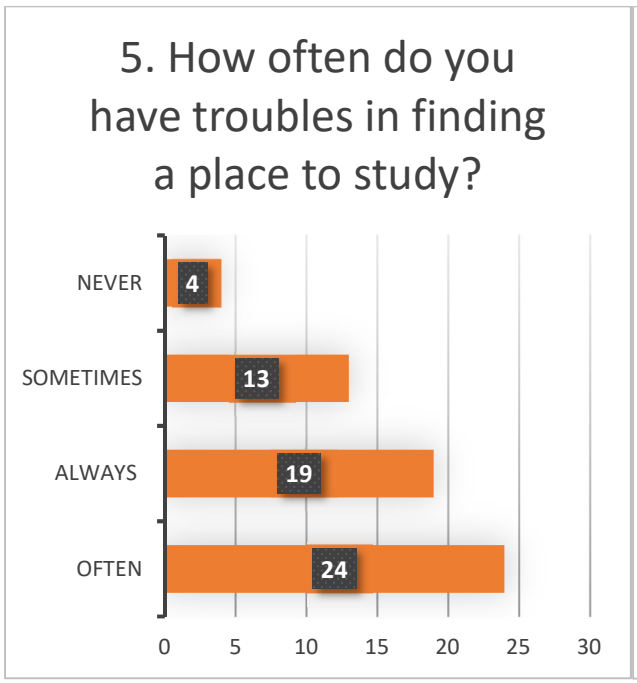


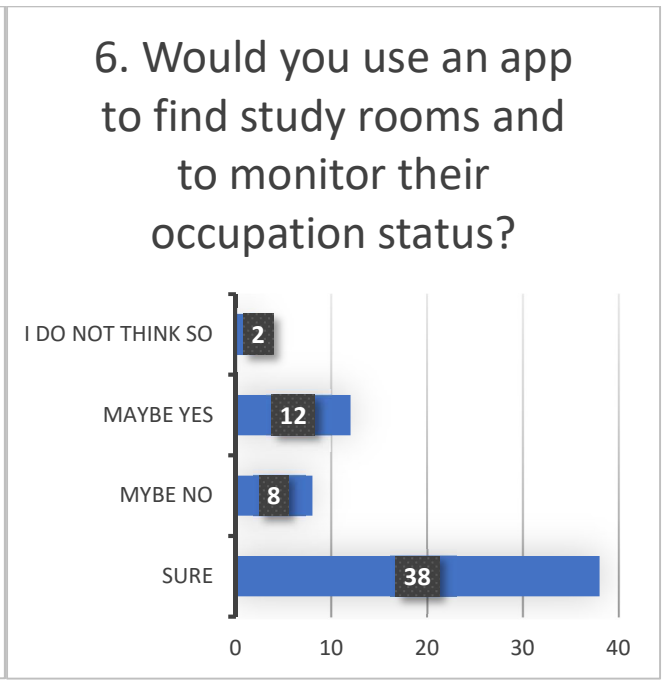
1. As you can see the majority of the students who answered were aged between 22 and 25.
2. The majority of the students that answered to the questionnaire studies engineering, and students from almost all the types of specialty answered to the questionnaire.



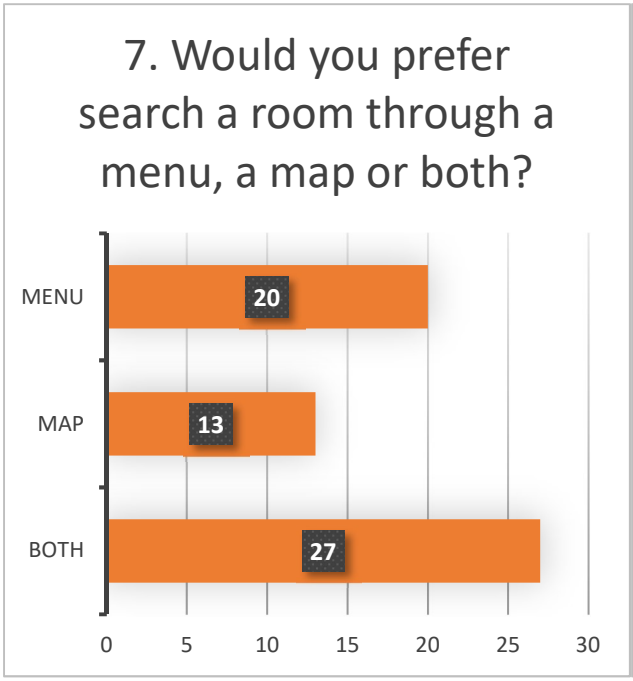


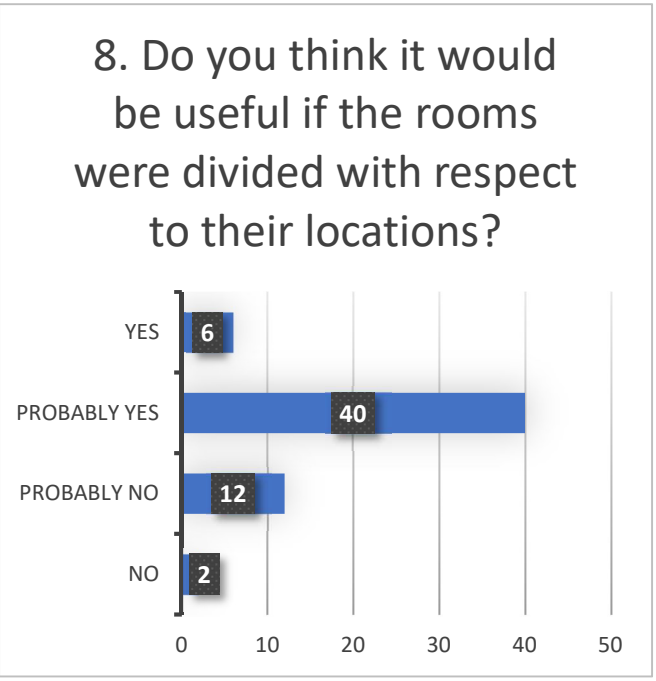
3. 58% of the participants that answered this questionnaire were males which is over a half of the total amount. The rest 42% were females.
4. Most of the students who answered to the questionnaire come from a master degree.



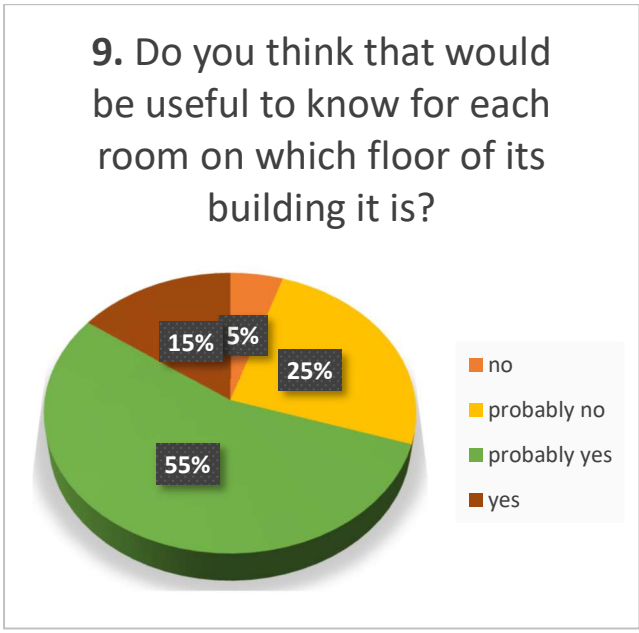


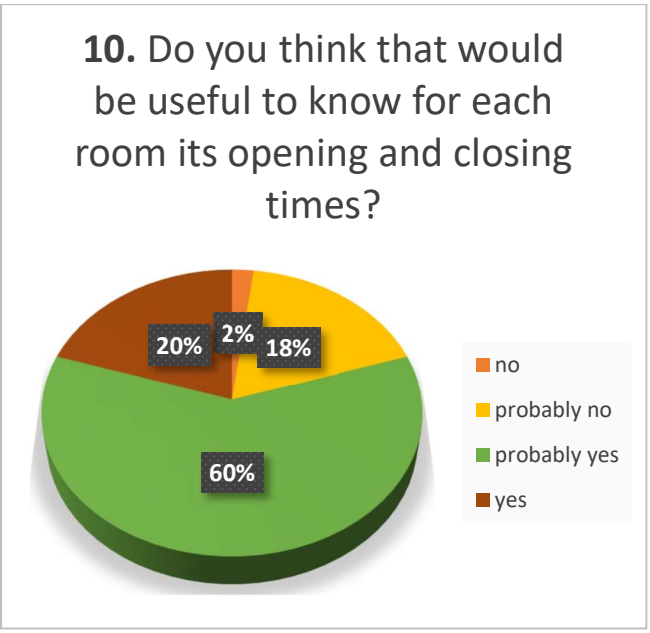
5. The results show us that the majority of students have troubles in finding a place to study.
6. The majority of students would like to have an app to find study rooms and to monitor their occupation status.



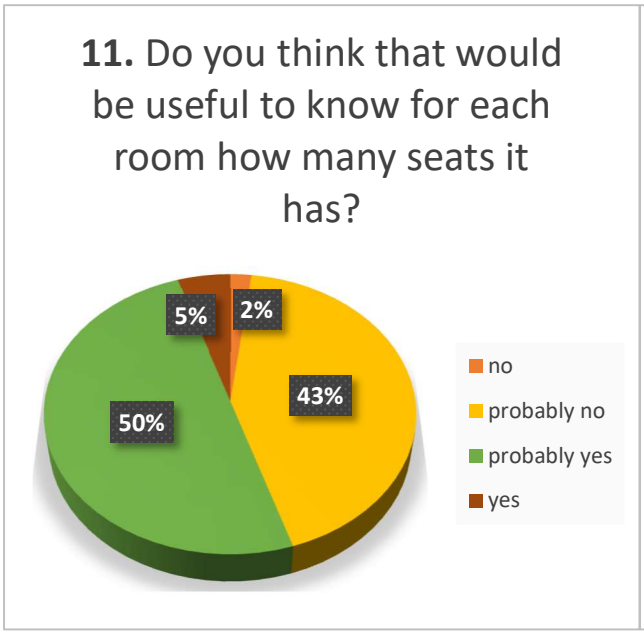


7. The majority of interviewed students would like both a menu and a map to find rooms.
8. The biggest part of students think that it would be useful if the rooms were divided with respect to their locations.

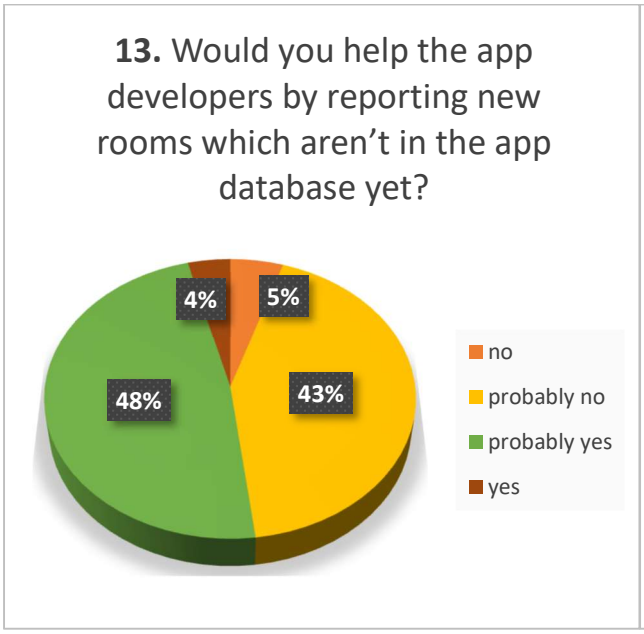




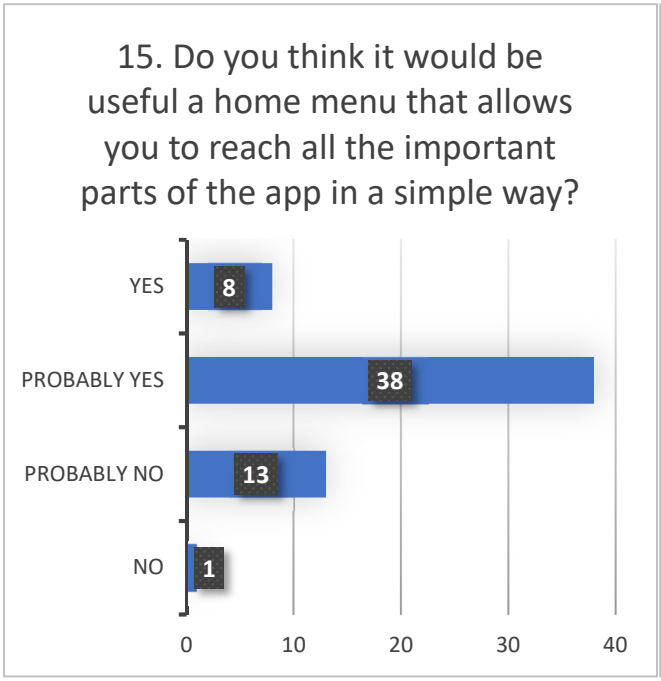
9. Students think that the floor of the room is a useful information.
10. Students think that the opening and closing times of a room are important information.

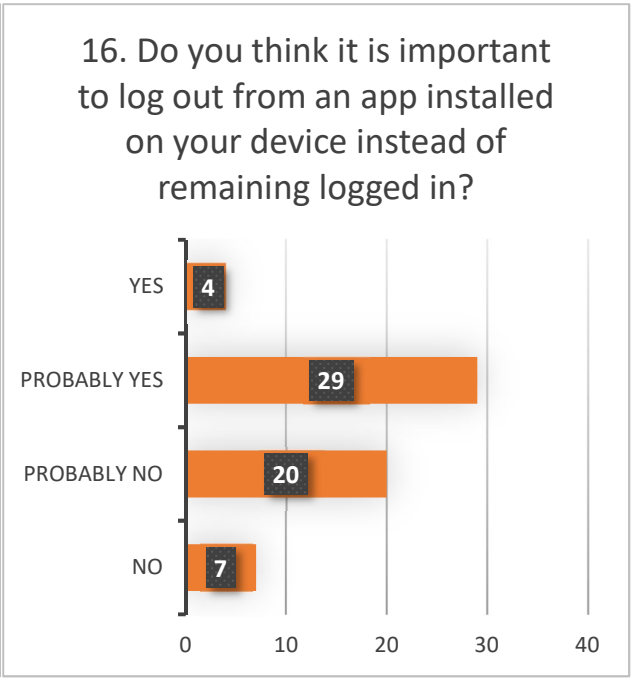


11. More than one half of the students think that it would be useful to know how may seats a room has got.
12. If there will be the occasion, the majority of students will help each other by reporting the occupation status of rooms where they are.

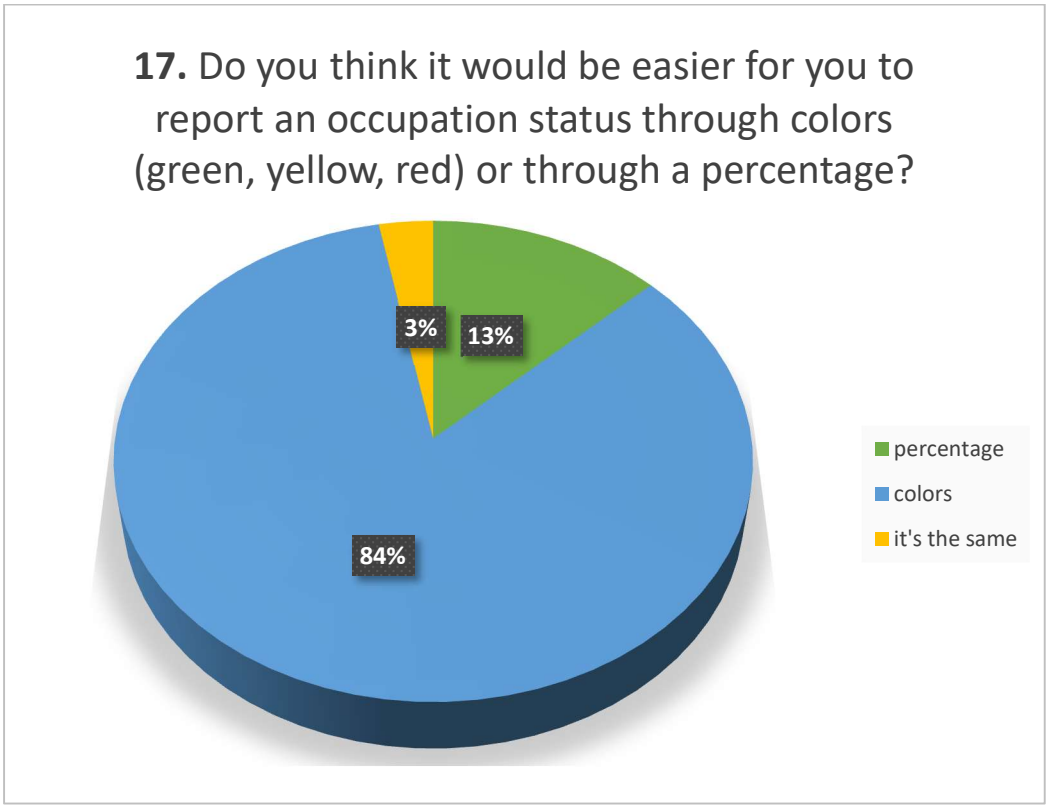


13. Slightly more than one half of the students would help the app developers by reporting new rooms which aren't in the app database yet.
14. The very majority of interviewed students wouldn't use an app that doesn't allow you to delete your account.





15. The majority of interviewed students would appreciate a practical home menu that allows to reach all the important parts of the application in a simple way.
16. More than one half of the students think that is important to log out from an app instead of remaining logged in.



17. The majority of interviewed students think that it would be easier to report the occupation status of a room through the use of colors (green, yellow, red).

Conclusions

Our target users are university students from different faculties and whose 58% is male and whose 42% is female. The majority of them comes from a master degree. They often have problems in finding study rooms and they would use an app to find them and to monitor their occupation status. Our app should have both a textual menu and a map and in the textual menu the rooms should be divided with respect to their locations. The information about a room that should be present are its floor, its opening and closing times and the number of its seats. The students would help each other by reporting the occupation status of the rooms where they are if they would have this possibility, so our app should offer this possibility. Our app should offer the possibility to report new rooms that aren't present in its database. Our app should offer the possibility to log out (because students think that this is an important feature) and to delete the user's account (since the majority of students wouldn't use it otherwise). WIMSapp should have a home menu that allows the user to reach all the most important parts of the app. Finally, our app should give to its users the possibility to report the occupation status of a room through the use of colors (green, yellow, red), since the major part of the students think that this is the easiest way to do it.

2.3 Competitor analysis

We searched eventual competitors for our app and we found only one which can be considered as a main competitor for WIMSapp:

The new unofficial version of Infostud developed by some students of the university of Rome La Sapienza.

There are many differences between WIMSapp and this alternative solution:

- in the unofficial Infostud the management of the rooms is only a marginal functionality and there isn't so much attention in developing it as in WIMSapp (in which the aim purpose is the focus on the rooms). Indeed, some aspects of the rooms functionality of the unofficial Infostud are still in a developing phase (such as the possibility to understand how much a room is full thanks to the use of Bluetooth);
- the rooms aren't divided with respect to their locations, but they are simply listed on the base of how much they are near to the user. This behavior is fine if the user is searching a room near him, but it becomes something bad when the user wants to check the status of a room far from him (in this case he is forced to use the search function of the app);
- the occupation status of a room is checked also through the use of the official timetables of the university. This is a great idea, but the problem is that these timetables are often untrustable;
- the check of the occupation status of the rooms (that is still a sperimental feature) is done through the use of Bluetooth, and this can lead to a lack of precision since the Bluetooth has got a range of about 20/30 m and the people who are in a room could be seen as they are in the room next if there is one;
- the unofficial Infostud shows a map with the location of the room only at the end of the search, as part of the information about the room. So, it isn't possible to use a map to see all the rooms which are near the same place;
- it hasn't got a system of favorite rooms;

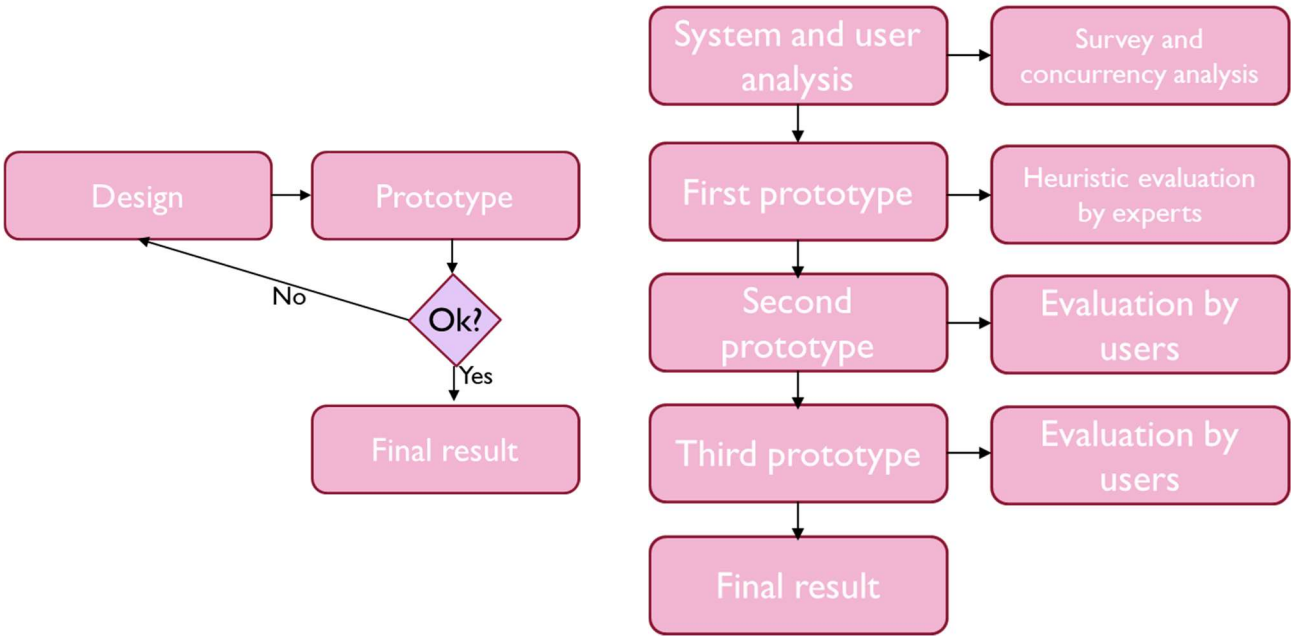
• it doesn’t give to the users the possibility to report new rooms or problems about existing rooms.

3. Work Plan

To work as a team, we took advantage by several tools and communication means. To share the same software and code and to manage their versioning we used GitHub. To deploy the backend, we relied on the Heroku platform. To keep the contacts between the members of the group we continued to meet often in real life and, when it wasn't possible, we communicated through phone. After the collection of the Requirements, the analysis of the data and the analysis of the competitors, we developed a first prototype through the use of mockups. This first prototype was then evaluated through an expert-based technique to come up with its possible defects. These defects were corrected and the second prototype was developed as an interactive interface made thanks to Android Studio. The second prototype was analyzed through a user-based technique to come up with its defects. Another time, these defects were corrected and the third prototype was developed as an interactive interface as its predecessor. Finally, the third prototype was evaluated through another user-based technique and, in this way, after the correction of its defects, the final application was obtained. For the final application also a backend was developed, in such a way to obtain a final product really working. So, the final product is an Android app really working.
As shown, the developing work went through three predominant phases:

- Requirements collection;
- Analysis and design (HTA and STN);
- Prototyping (mockup, first prototype, second prototype, third prototype).

Here below there is a schematization of this process:



4. Design and Analysis

4.1 Definition

4.1.1 Hierarchical Task Analysis (HTA)

Hierarchical Task Analysis (HTA) is a task description methodology. HTA is used to produce a complete description of tasks in a hierarchical structure of goals, sub-goals, operations and plans.

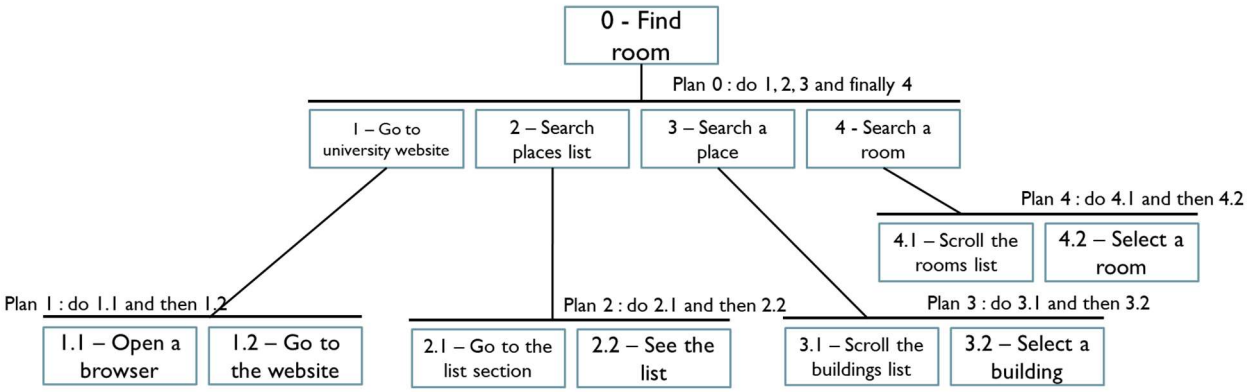
4.1.2 State Transition Network (STN)

A State Transition Network (STN) is a schema that is developed from a set of data and graphs the flow of data from particular data points (called states or nodes) to the next ones.

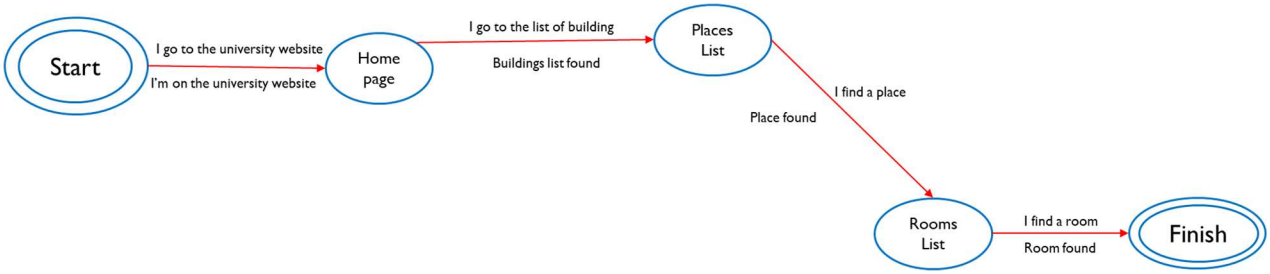
4.2 HTA & STN

After the requirements analysis step, we prepared some HTA and STN schemas based on the data collected during this previous stage. We observed the way in which the potential users of our app solved the problems of finding a room and checking if it is free or not and we "represented" these observed behaviors in the following HTA and STN schemas.

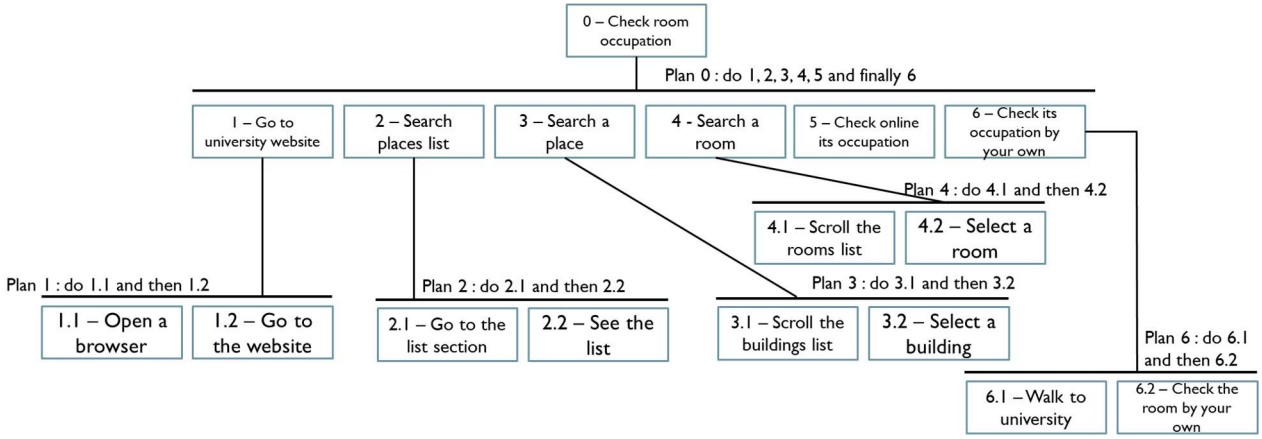
HTA for "Find room"



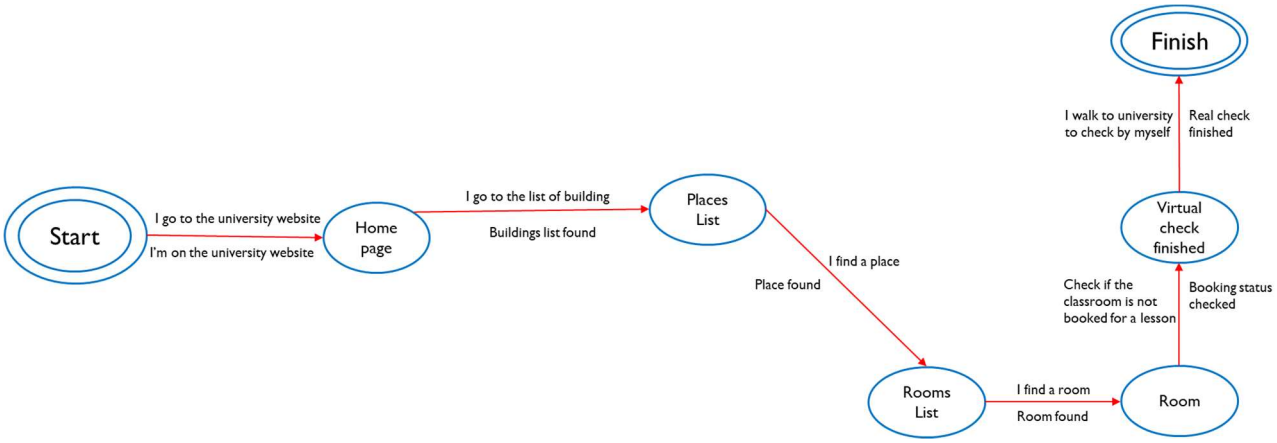
STN for "Find room"



HTA for "Check room occupation"



STN for "Check room occupation"



5. Prototype 1

After the design and analysis step, we continued the work process by developing the first prototype of the application.

5.1 Prototype description

Our first prototype was realized through mock-up. Its main characteristics were:

- it had an essential graphic user interface, to avoid a too heavy design that, even if fancy, wouldn't be easy to use;
- we chose to use familiar icons, in such a way that the user can take advantage of his knowledge of other applications that use similar icons and that the user is facilitated in his utilization of the app by recognizing familiar icons instead of recall the meaning of icons that he never saw before;
- it presents the possibility to check and report a room occupation status by following simple and linear activity flows;
- it offers also the possibility to report a new room by following another simple activity flow;
- there is an intuitive, and so easy to take advantage by, division of the sections of the application;
- it offers a simple and intuitive home menu as starting point for the navigation of the application.

5.2 Expert-based evaluation technique

5.2.1 Definition of heuristic evaluation

A heuristic evaluation is a usability inspection methodology for computer software that helps to identify usability problems in the user interface design. It specifically involves evaluators examining the interface and judging its compliance with recognized usability principles (the "heuristics"). These evaluation methods are now widely taught and practiced in the new media sector, where UIs are often designed in a short space of time on a budget that may restrict the amount of money available to provide for other types of interface testing.

The main goal of heuristic evaluations is to identify any problem associated with the design of user interfaces. Usability consultant Jakob Nielsen developed this method on the basis of several years of experience in teaching and consulting about usability engineering.

The heuristics as published in Nielsen's book Usability Engineering are as follows.

5.2.1.1 Heuristics

1. Visibility of system status:

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

2. Match between system and the real world:

The system should speak the user's language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

3. User control and freedom:

Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

4 Consistency and standards:

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

5 Error prevention:

Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

6 Recognition rather than recall:

Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

7 Flexibility and efficiency of use:

Accelerators — unseen by the novice user — may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

8 Aesthetic and minimalist design:

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

9 Help users recognize, diagnose, and recover from errors:

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

10 Help and documentation:

Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

5.2.1.2 Severity scale

0 = I don't agree that this is a usability problem at all

1 = Cosmetic problem only

2 = Minor usability problem

3 = Major usability problem

4 = Usability catastrophe

5.2.2 Heuristic evaluation

So, the evaluation of the first prototype was a heuristic evaluation made by experts.

For this step the evaluation was made by our professor Valeria Mirabella.

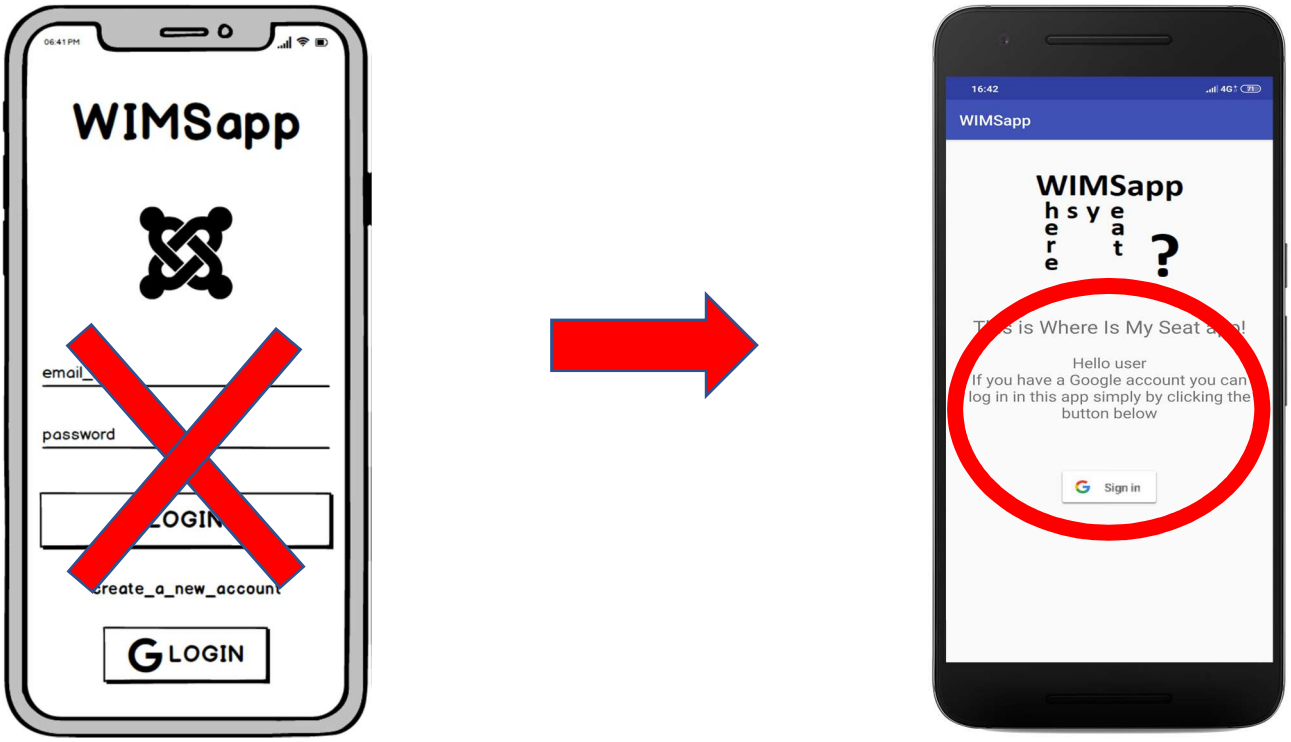
Here below there is her heuristic evaluation of our project:

Page	Heuristic violated	Severity	Description / Comment
1	Flexibility and efficiency of use	2	Avoid asking unnecessary data. If Google account is mandatory you can skip login registration
2	Recognition rather than recall	2	Write something that can help uses in call to mind application's goals. You could anticipate some contents of the following page.
4	Flexibility and efficiency of use	3	Think about search functions
10	Match between the system and the real world	3	POI based on rooms can be overlaid
7	Match between the system and the real world	3	Give more information about room availability
5/6	Flexibility and efficiency of use	3	Think about accelerator for experienced users

5.2.2.1 Correction of detected defects

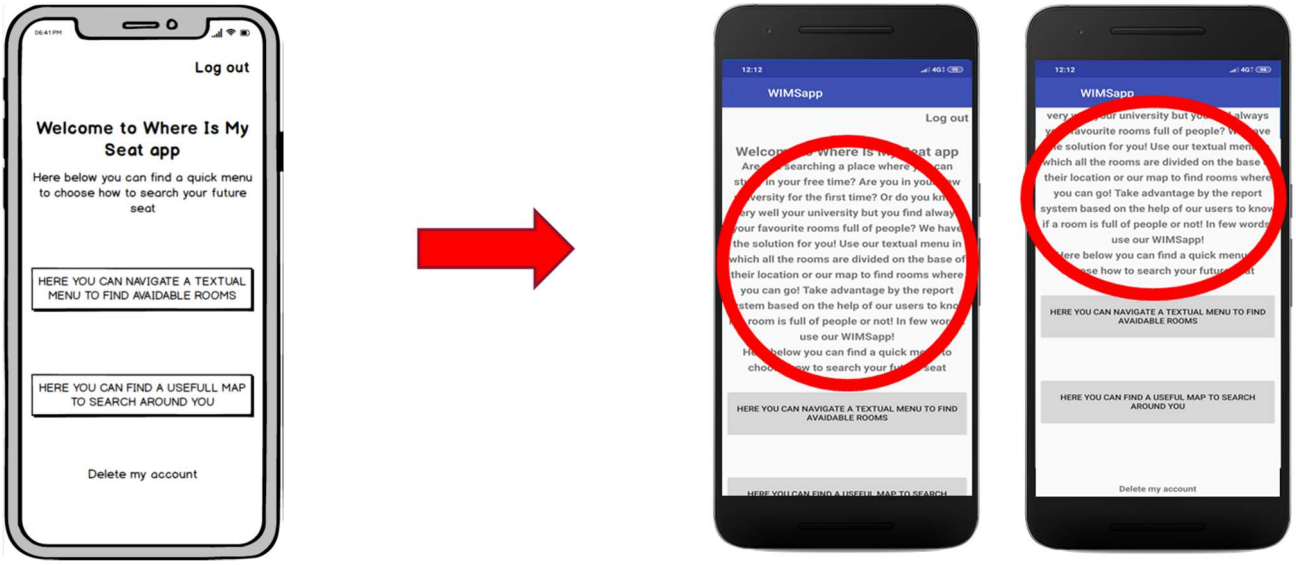
The first suggestion for our application was: "Avoid asking unnecessary data. If Google account is mandatory you can skip login registration".

We corrected this defect by deleting the autonomous login system present in the first prototype and leaving only the login through Google account, as shown here below:



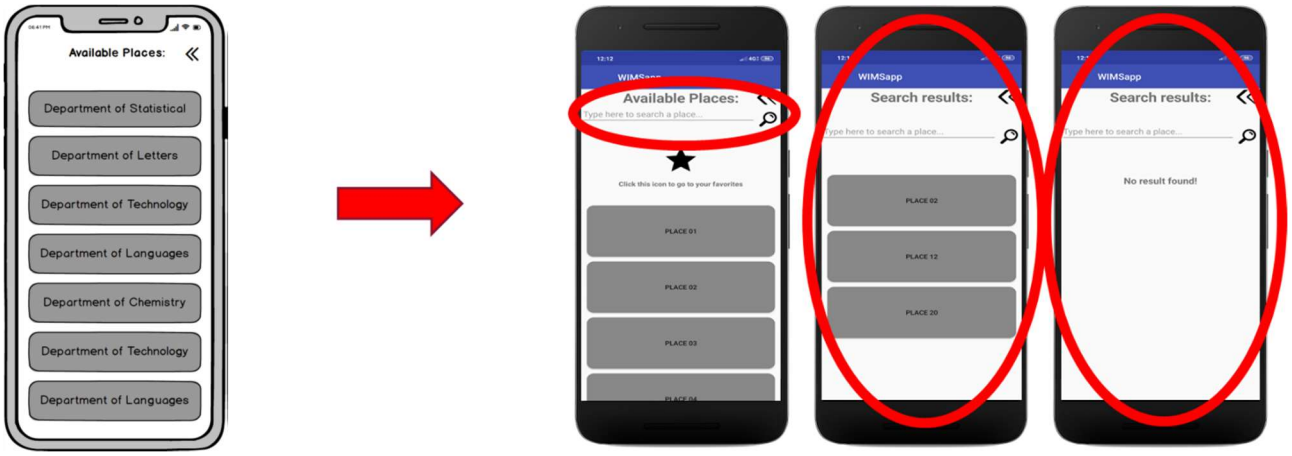
"Write something that can help uses in call to mind application's goals. You could anticipate some contents of the following page."

We corrected this aspect by adding to the homepage a small description of the application goals and some anticipations of the contents of the other parts of the application:



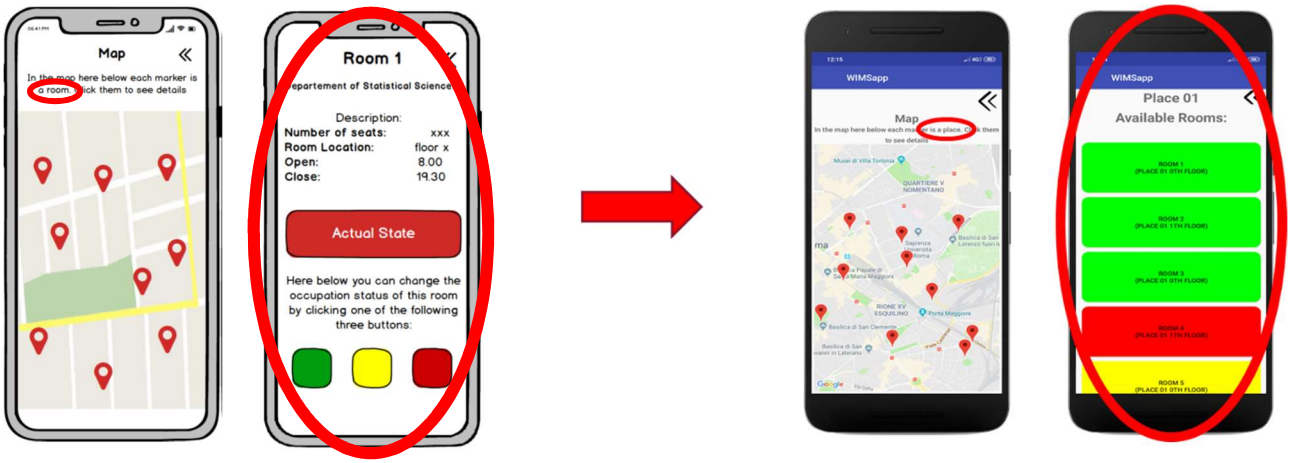
"Think about search functions"

To supply to this lack, we added in the page of the available places a search bar to search among all the places that are in the database of the application:



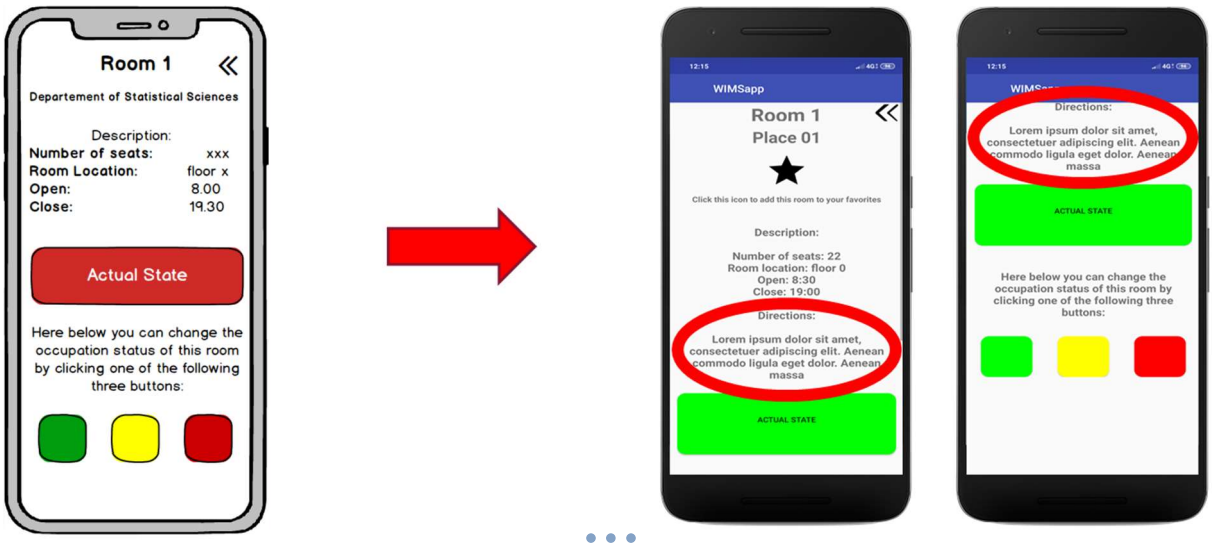
"POI based on rooms can be overlaid"

So, we based the map pointers not on rooms, but on places, as shown here below:



"Give more information about room availability"

To correct this defect, we added some directions to reach the room to the room information which can be seen in the page of the room occupation status report:



“Think about accelerator for experienced users”

So, we added to the room occupation status report page a clickable icon to add the room to a group of favorite rooms and to the page of available rooms another clickable icon which guide you to the list of these favorite rooms:

6. Prototype 2

And so, after the correction of the defects of the first prototype was finished, we obtained our second prototype.

6.1 Prototype description

Our second prototype was realized through Android Studio. It was essentially an interactive interface working on Android devices but it still didn't "really work", in the sense that it still hadn't got a working backend.
Its main differences with respect to the previous prototype were essentially the ones linked to the corrections described at the end of the past chapter.

6.2 User-based evaluation technique

6.2.1 Definition of think aloud

A think aloud is an evaluation technique based on the help of some potential users of the final product. Its main characteristics are:

- User observed performing task
- User asked to describe what he is doing and why, what he thinks is happening etc.
- Advantages
  - simplicity - requires little expertise
  - can provide useful insight
  - can show how system is actually used
- Disadvantages
  - subjective
  - selective
  - act of describing may alter task performance

6.2.2 Think aloud

So, we chose to evaluate the second prototype thanks to the help of some users through the think aloud technique.

8 people were interviewed and they were asked to perform the 3 main tasks that WIMSapp had at that point of the development:

- Check the current occupation status of a room;
- Report the current occupation status of a room;
- Report a new room.

There was no intromission by the evaluator during the evaluation.

The major problems found by the users during this evaluation were two:

- The absence of a menu usable in all the parts of the application (except in the login page)
- The impossibility to understand how much old the reports about the occupation status of the rooms were and so how much they were trustable

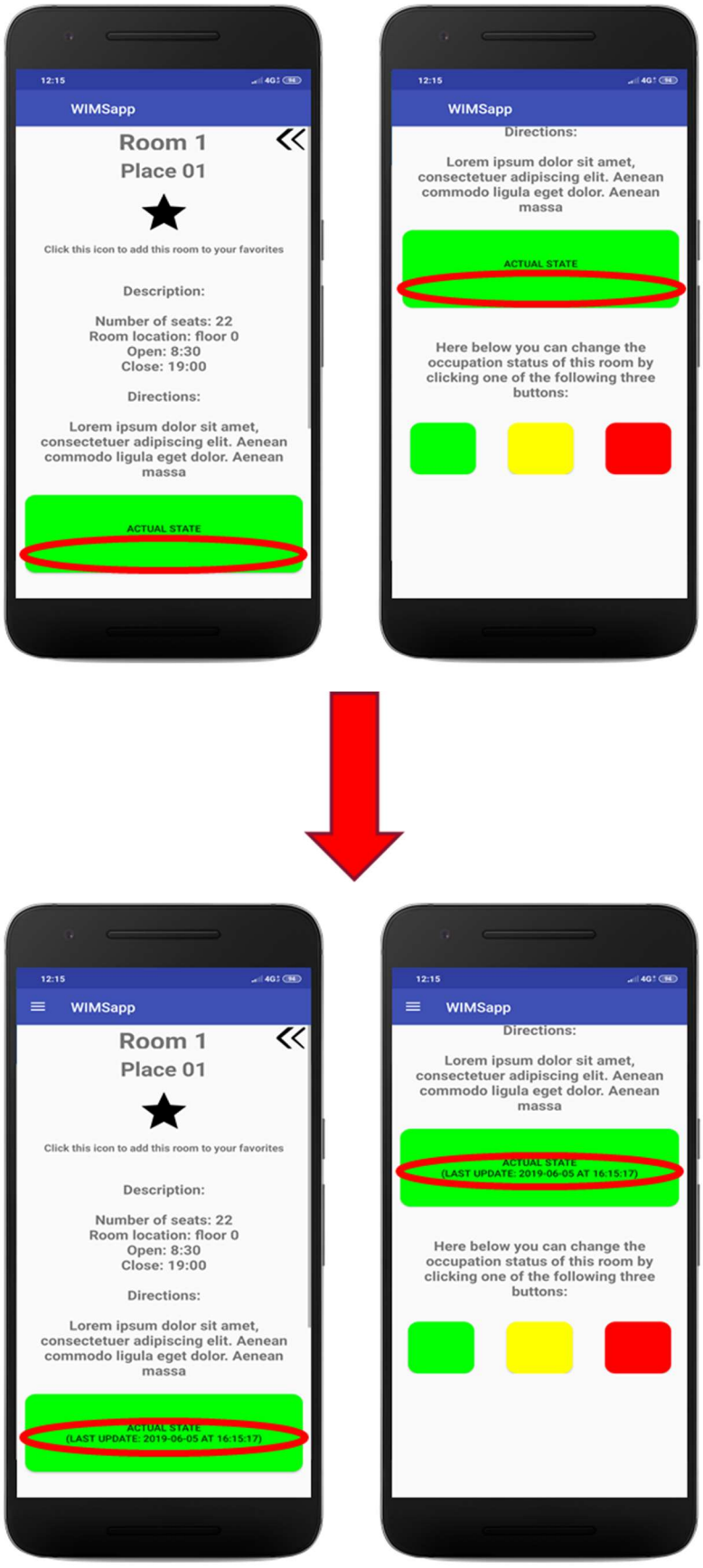
6.2.2.1 Correction of detected defects

The first problem detected thanks to the users' help was: "The absence of a menu usable in all the parts of the application (except in the login page)".

We corrected this defect by adding a slide menu which can be used in all the pages of the application with the only exception of the login page, as shown here below:

The other defect detected was: "The impossibility to understand how much old the reports about the occupation status of the rooms were and so how much they were trustable"

We solved this problem by adding under the occupation status report of a room the indication of which were the date and the time in which this report was done, as shown here below:


12:15
WIMSapp
Room 1
Place 01
Click this icon to add this room to your favorites
Description:
Number of seats: 22
Room location: floor 0
Open: 8:30
Close: 19:00
Directions:
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa
ACTUAL STATE
12:15
WIMSapp
Directions:
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa
ACTUAL STATE
Here below you can change the occupation status of this room by clicking one of the following three buttons:
12:15
WIMSapp
Room 1
Place 01
Click this icon to add this room to your favorites
Description:
Number of seats: 22
Room location: floor 0
Open: 8:30
Close: 19:00
Directions:
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa
ACTUAL STATE
(LAST UPDATE: 2019-06-05 AT 16:15:17)
12:15
WIMSapp
Directions:
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa
ACTUAL STATE
(LAST UPDATE: 2019-06-05 AT 16:15:17)
Here below you can change the occupation status of this room by clicking one of the following three buttons:

7. Prototype 3

And so, after the correction of the defects of the second prototype was finished, we obtained our third prototype.

7.1 Prototype description

Also our third prototype was realized through Android Studio. Even in this case, it was essentially an interactive interface working on Android devices but it still didn't "really work", in the sense that it still hadn't got a working backend.
Also for this prototype, its main differences with respect to the previous one were essentially the ones linked to the corrections described at the end of the past chapter.

7.2 User-based evaluation technique – cooperative evaluation

7.2.1 Definition of cooperative evaluation

A cooperative evaluation is a variation on the think aloud technique and so it is an evaluation technique based on the help of some potential users of the final product too. Its main characteristics are:

- User collaborates in evaluation
- Both user and evaluator can ask each other questions throughout
- Additional advantages
    - Less constrained and easier to use
    - User is encouraged to criticize the system
    - Clarification is possible

7.2.2 Cooperative evaluation

So, in this new case we chose to evaluate the third prototype through Cooperative Evaluation, this variation on the think aloud technique.

8 people were interviewed and they were asked to perform the 3 main tasks that WIMSapp had at that point of the development (the same 3 tasks of previous prototype):

- Check the current occupation status of a room;
- Report the current occupation status of a room;

- Report a new room.

As told in the previous paragraph, during this evaluation both user and evaluator could have asked each other questions, with some advantages as the possibility for the evaluator to encourage the users to criticize the application.

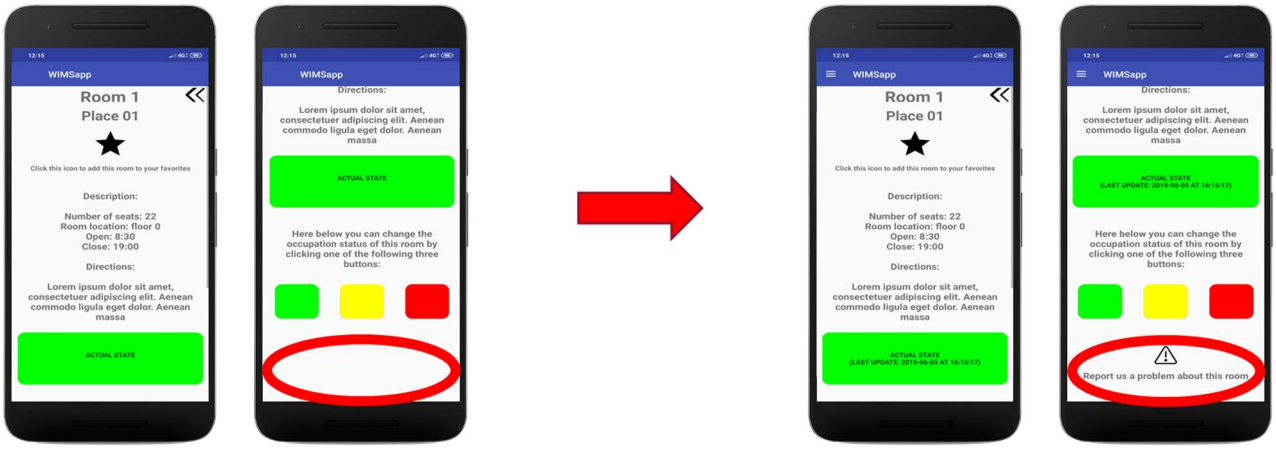
The major problems found by the users during this evaluation were two:

- The impossibility to report possible problems about existing rooms
- The absence of buttons to log out and to delete the account also in the slide menu

7.2.2.1 Correction of detected defects

The first problem detected thanks to this other evaluation technique was: “The impossibility to report possible problems about existing rooms”.

We corrected this defect by adding to the room info page a clickable icon that links this page to a report page in which the user can report problems about the existing room that he was watching, as shown here below:



“The absence of buttons to log out and to delete the account also in the slide menu”.

We solved this problem simply by adding to the slide menu a couple of buttons to log out and to delete the user’s account, as shown here below:

7.3 User-based evaluation technique – controlled experiment

7.3.1 Definition of controlled experiment

A controlled experiment is an experiment in which everything is held constant except for one variable. Usually a set of data is taken for a control group, which is commonly the normal or usual state, and one or more other groups are examined, where all conditions are identical to the control group and each other except this one variable. Sometimes it's necessary to change more than one variable, but all the experimental conditions will be controlled so that only the variables being examined change and the amount or way they change is measured.

Controlled experiments are considered to be the most rigorous of empirical methods capable of providing empirical evidence to support a particular claim or hypothesis.

7.3.2 Controlled experiment

Since we had some doubts about the way in which we implemented the occupation report system and we wanted to be sure that the interface we implemented was the best possible one, we decided to implement also an alternative interface to manage this particular aspect of the user interaction

with our product and we ran a controlled experiment to understand which of the two alternative interfaces was the best one.

7.3.2.1 Participants

We searched, and we chose some participants for our controlled experiment in such a way that the sample which we obtained was as coherent to our supposed users' population as possible.

To choose the most appropriate participants we relied on the data collected thanks to our questionnaires distributed during the requirements analysis step and we tried to recreate a sample population which reflected as much as possible those data.

We searched among university students (which would have been our future users), among them we asked to our friends and not (luckily, we met some kind and disponible people) and finally we found 15 participants for our experiment (at that time none of them had never watched our application before):

- 9 of them were aged between 22 and 25 (we considered that from the questionnaires the 58% of our potential users had this age), 5 of them were aged between 18 and 21 (as the 34% of our users as the results of our questionnaires said) and the remaining one was 26 years old (indeed by the questionnaires results the 8% of our user was aged between 26 and 30);
- 6 of them were from an engineering faculty, 3 of them were from medicine, other 2 of them were from literature, while the others were everyone from a different faculty, such as management, economics, law and philosophy; even in this case we succeeded more or less in reflecting the data collected with our questionnaires;
- 9 of them were males while the other 6 were females (58% male and 42% female from the questionnaires results);
- 8 of them were from a master degree, 6 of them were from a bachelor degree and the remaining one was from a PhD; even in this case we succeeded more or less in reflecting the data collected through our questionnaires (50% from a master degree, 42% from a bachelor degree and 8% from a PhD or higher).

7.3.2.2 Variables

As the controlled experiment technique wants, we divided the experiment variables in two groups:

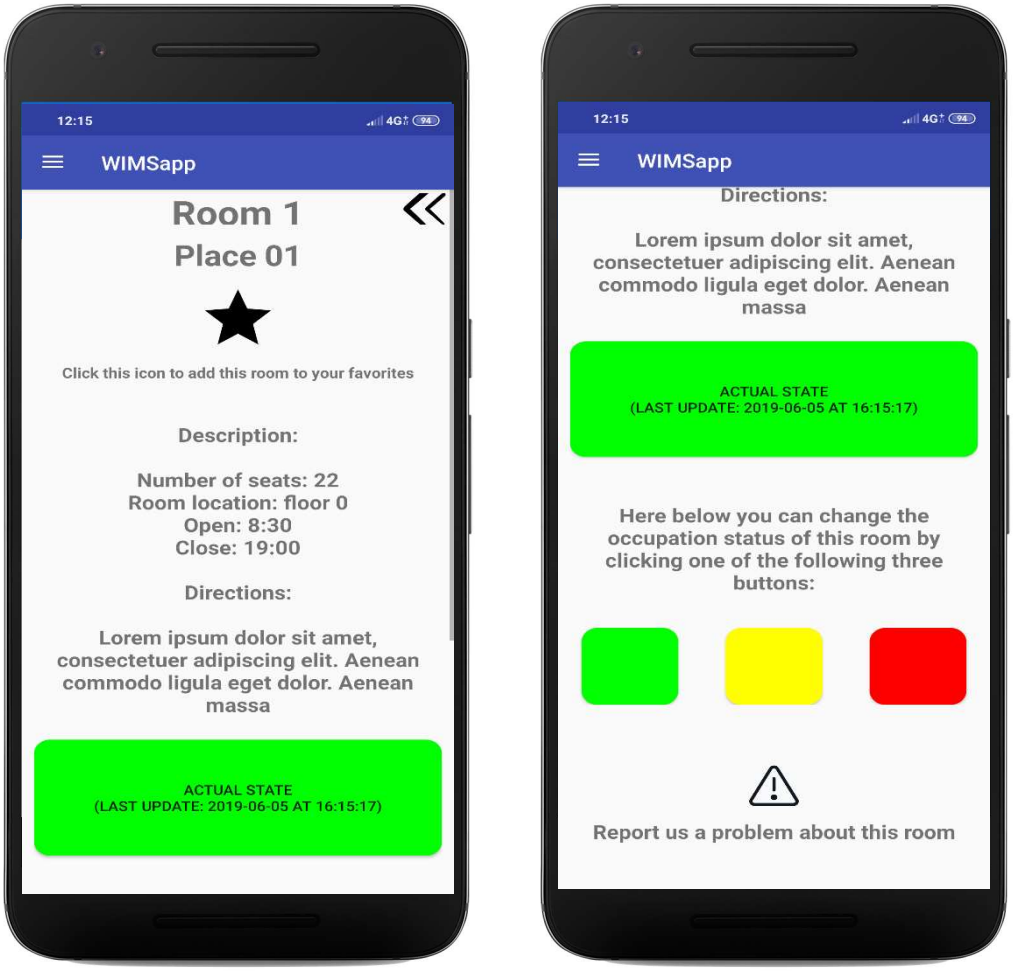
- The independent variables, which are the ones manipulated by us in order to set properly the experiment;
- The dependent variables, which are the ones measured and interpreted to get the results of the experiment; these dependent variables should change only on the base of some changes in the independent variables and they should be free from noises of other factors, if the experiment is well set.

7.3.2.2.1 Independent variables

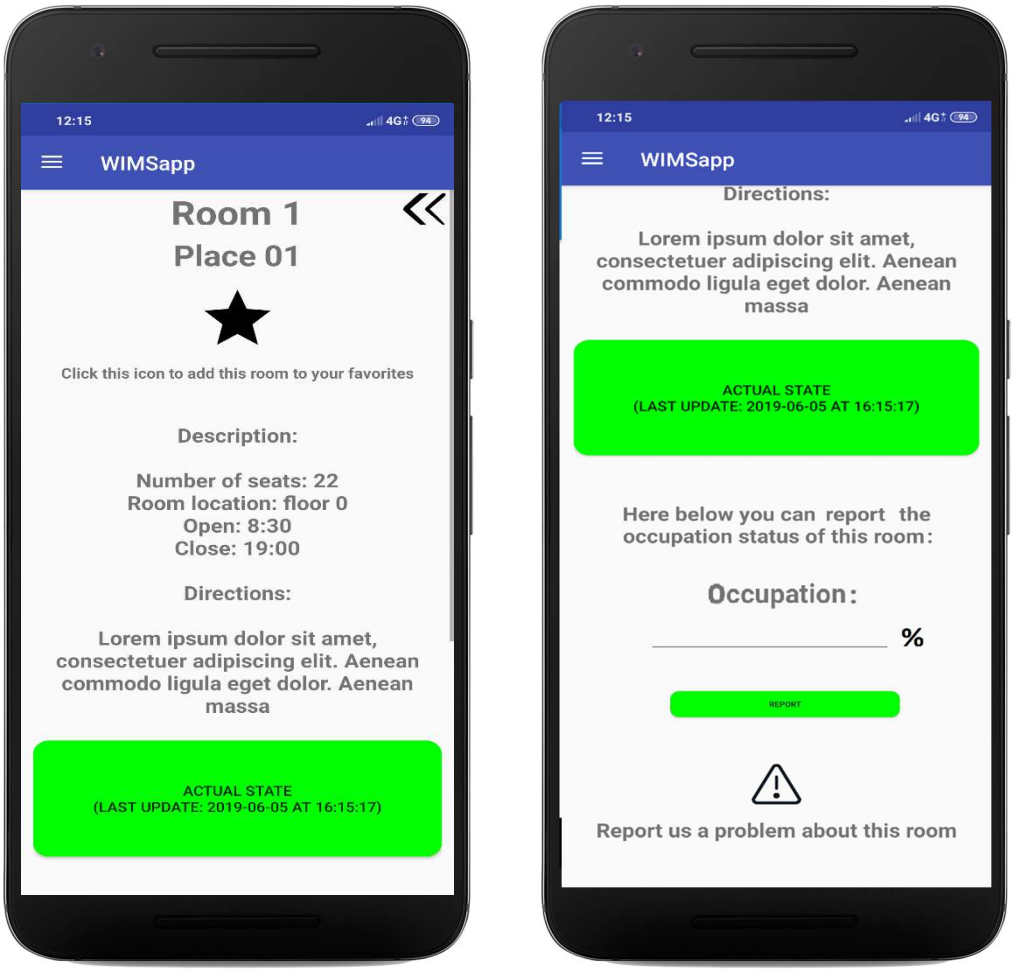
As independent variables we chose to use the two different interfaces which we developed to support the occupation report system.

These different interfaces are the Interface 1, which is the first interface we developed in order to fulfil this purpose, and the Interface 2, which is the alternative one that we thought in order to test if it could be better than the first one or if our first solution was the best possible.

These two interfaces are shown here below.



Interface 1: through this interface the user can report the occupation status of a room by simply clicking one of the three colored buttons (green, yellow and red), consistently with how much he thinks that the room is full of people. The current occupation color of the room will change on the base of the color of this user's report.



Interface 2: through this interface the user can report the occupation status of a room by inserting the occupation percentage of the room where he is, consistently with how much he thinks that the room is full of people, and by clicking the report button. The current occupation color of the room will change coherently with the reported percentage (the occupation color of the room will become green if the percentage is between 0% and 33%, it will become yellow if the percentage is between 34% and 66% or it will become red if the percentage is between 67% and 100%).

7.3.2.2.2 Dependent variables

We decided to rely only on one dependent variable, which is the one that we thought being the most appropriate one: the time used by the users to complete the task of reporting the occupation status of a room.

So, we asked to our sample of users to complete this task of reporting the occupation status of a room and we registered the times which they took to complete this task through the two different interfaces.

The only thing which changed between a test and another one was the interface used by the current user, so we were sure that the dependent variable (the time used to complete the task) varied on the base of only changes of the independent variable (the interface used).

7.3.2.3 Hypotheses

A prediction of the outcome of the experiment. This prediction is stated in terms of Independent Variables and Dependent Variables; what it is to show is that a variation in the Independent Variables will cause an effect on the Dependent Variables.

In the experiment, we would like to prove that our prediction is correct i.e. to confirm our hypothesis. This is done by disproving/rejecting the null hypothesis (a null hypothesis states that a variation in the Independent Variables does not make any differences in the Dependent Variables). Moreover, to disprove this null hypothesis and to demonstrate that our hypothesis is true, it is necessary as first thing that the data collected during the experiment have a statistical relevance (this demonstrates that the null hypothesis is false), and then, if the first thing is true, it is necessary to observe the differences among the means of the data collected and eventually show that these differences prove our hypothesis.

7.3.2.3.1 Null hypotheses

Both the interfaces offer the same performance to the users and so there are no differences among the usages of these two different designs by the users (the times used by the users to complete the task of reporting the occupation status of a room will be more or less the same ones with both the interfaces).

7.3.2.3.2 Our hypotheses

The interface 1 offers better performances with respect to the interface 2 since the first one has a simpler and more intuitive design and interaction. This means that the times taken by the users to complete the task of reporting the occupation status of a room will be shorter by using the interface 1 with respect to those times taken with the interface 2.

7.3.2.4 How the experiment ran

There are two different techniques to run a controlled experiment: the Between Groups experiment and the Within Groups one.

In the Between Groups experiments the sampled users are divided in groups and users who are from the same group test only one interface.

In the Within Groups experiments each sampled user test every interface available.

Since usually to run a Between Groups controlled experiments it is necessary to have available a great number of users and we found only 15 testers for the experiment, we decided to run a Within Groups controlled experiment.

Because of this decision, we had to face an important issue: the learnability effect. This problem states that a user who used an interface to complete a determined task will learn something about the system in use while completing this task, and so if the same user is asked to complete the same task but through a different interface, this second time he will be advantaged by his past experience and he will complete the task in a easier way and with less time. But in this way, the results of the experiment could be slightly bias, and they can be a bit untrustable.

This issue can be completely overcome by relying on a Between Groups controlled experiments, but since this kind of experiment requires a great number of testers it isn't always possible to run it (as in our case).

So, since we decided to run a Within Group controlled experiment because of the number of testers who we had available, we overcome the learnability effect issue by relying on another additional technique: we divided our sampled users in two groups and we asked to the members of the first group to test both the interfaces by testing before the interface 1 and then the interface 2, and we asked to the members of the second group to do exactly the opposite, i.e. to test before the interface 2 and then the interface 1.

In this way, we mitigated the effect of the learnability effect issue in our Within Groups controlled experiment.

So, we divided our sampled users and we asked them to test both our interfaces in the way that is described here above, we asked them to complete the task of reporting the occupation status of a room and we took the times they needed to complete this task with each of the two interfaces.

7.3.2.5 ANOVA single factor and results

After the needed data were collected, we ran an ANOVA (analysis of variance) single factor statistical analysis on them. ANOVA single factor is a kind of statistical analysis used to understand if some data are statistical relevant or not and to derive from these data some useful results.

The data we collected, and their ANOVA single factor analysis are shown here below:

Interface 1	Interface 2
50	55
47	50
52	58
43	45
51	57
40	44
54	58
45	51
48	50
44	46
52	56
46	51
56	59
42	45
48	53

Analisi varianza: ad un fattore

RIEPILOGO

Gruppi	*Conteggio*	*Somma*	*Media*	*Varianza*
Colonna 1	15	718	47,86667	21,40952
Colonna 2	15	778	51,86667	27,12381

ANALISI VARIANZA

Origine della variazione	*SQ*	*gdl*	*MQ*	*F*	*Valore di significatività*	*F crit*
Tra gruppi	120	1	120	4,945055	0,034407333	4,195972
In gruppi	679,46667	28	24,26667			
Totale	799,46667	29				

As we can see, F > F crit (i.e. 4.945055 > 4.195972).

Therefore, we can reject the null hypothesis. This means that our collected data have statistical relevance and that the means of the two populations examined (the times taken to complete the task of reporting the occupation status of a room through interface 1 and the times taken to perform the same task but through interface 2) are not equal.

But did this difference/effect happen by chance or was it caused by the Independent Variables?

Usually, the chosen threshold for the significance level is of 0.05 (as in this case).

Since here the significance value is of 0.034407333 and *significance value < significance level threshold* (i.e. 0.034407333 < 0.05), we can say that this is a significant result and that it was caused by the Independent Variables (as we wanted).

Since the null hypothesis is rejected and these results are proved to be statistically significant, we know that to use the interface 1 or the interface 2 isn't the same thing.

So, we can see the means of these two populations and we can say that, since the mean obtained with interface 1 is less than the mean obtained with interface 2 (47.86667 < 51.86667) and we are talking about times taken to complete a task, the interface 1 is better than the interface 2.

So, we were lucky and we confirmed our hypothesis, i.e.:

"The interface 1 offers better performances with respect to the interface 2 since the first one has a simpler and more intuitive design and interaction. This means that the times taken by the users to complete the task of reporting the occupation status of a room are shorter by using the interface 1 with respect to those times taken with the interface 2".

For these reasons, we decided to keep the interface 1 and to discard the alternative interface 2.

8. Final Product

8.1 Overview

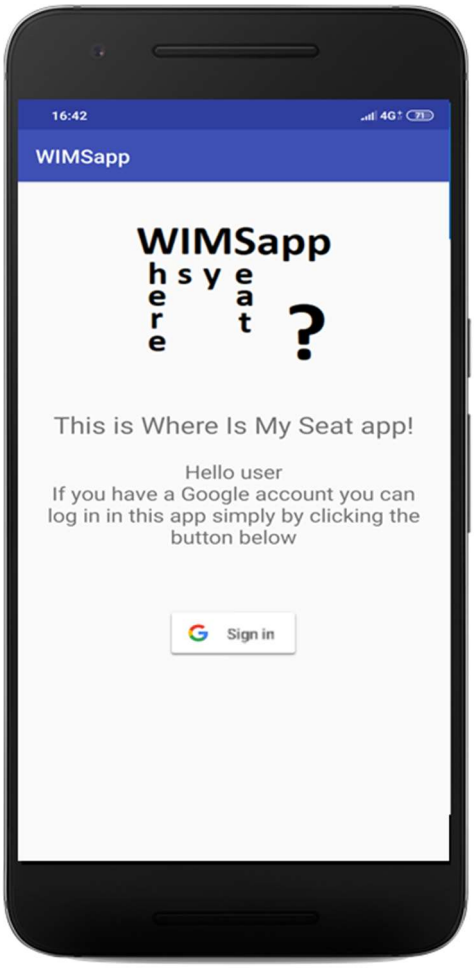
The final application was realized through Android Studio by modifying directly the third prototype and by implementing, in addition to a reactive interface, also a working backend.

In this way, after the correction of all the defects of the third prototype, a working final application was obtained.

In few words, this application is a “really working” Android app with both a front and a back end.

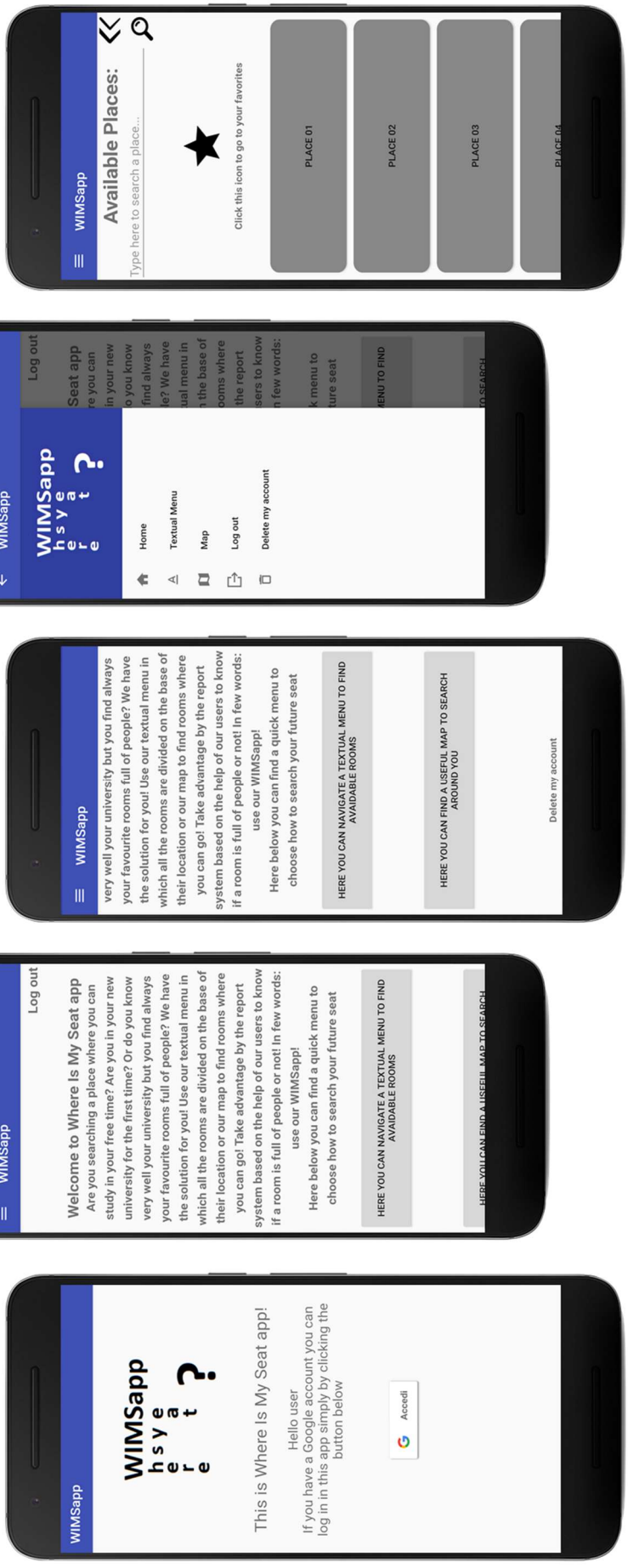
Features:

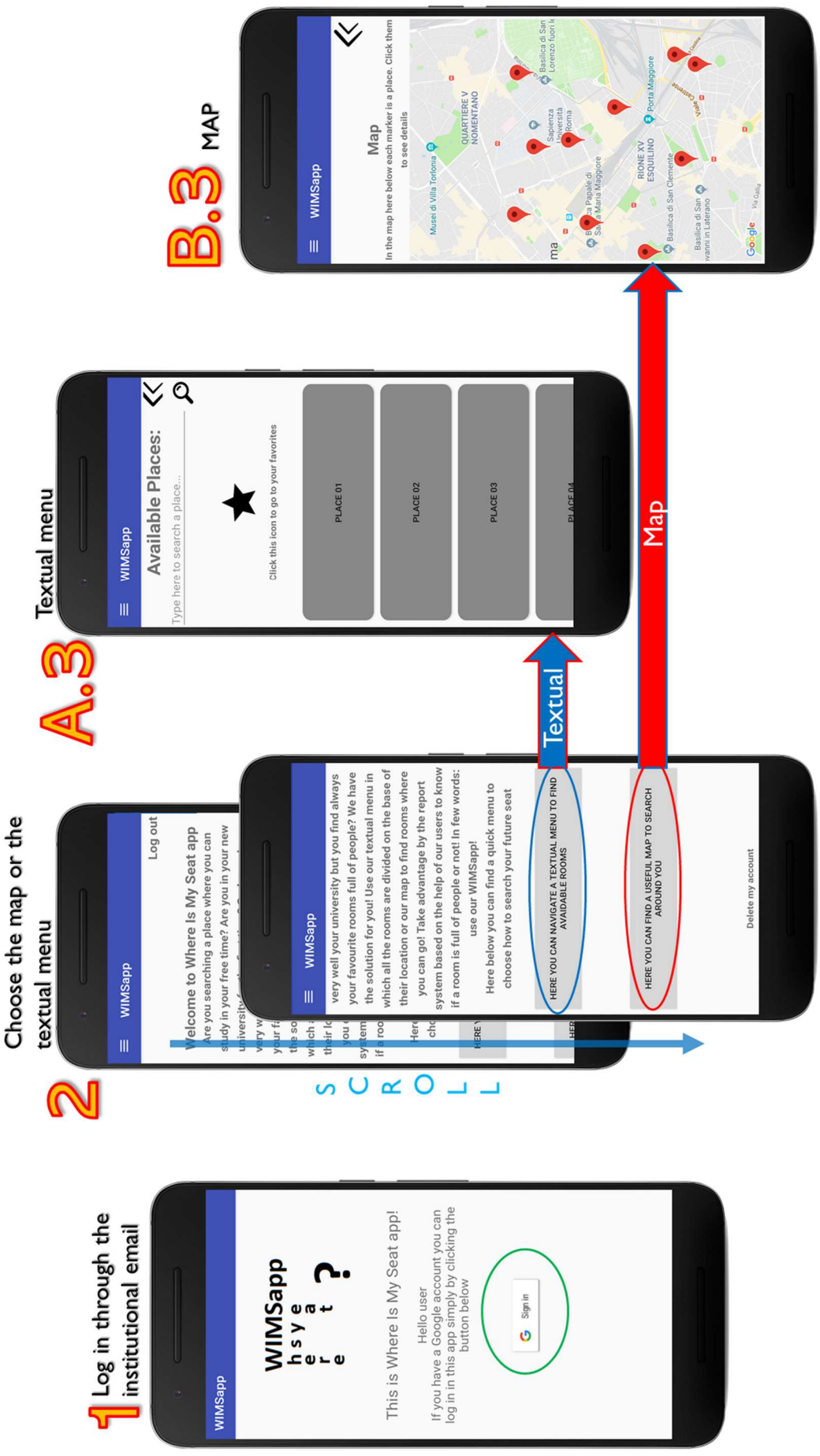
- Google login
- Useful home with quick menu
- Slide menu
- Textual menu with rooms divided with respect to their locations
- Map
- Room details
- Favorites features
- Possibility of checking rooms occupation status
- Possibility of reporting rooms occupation status
- Possibility of reporting new rooms
- Possibility of reporting problems about existing rooms

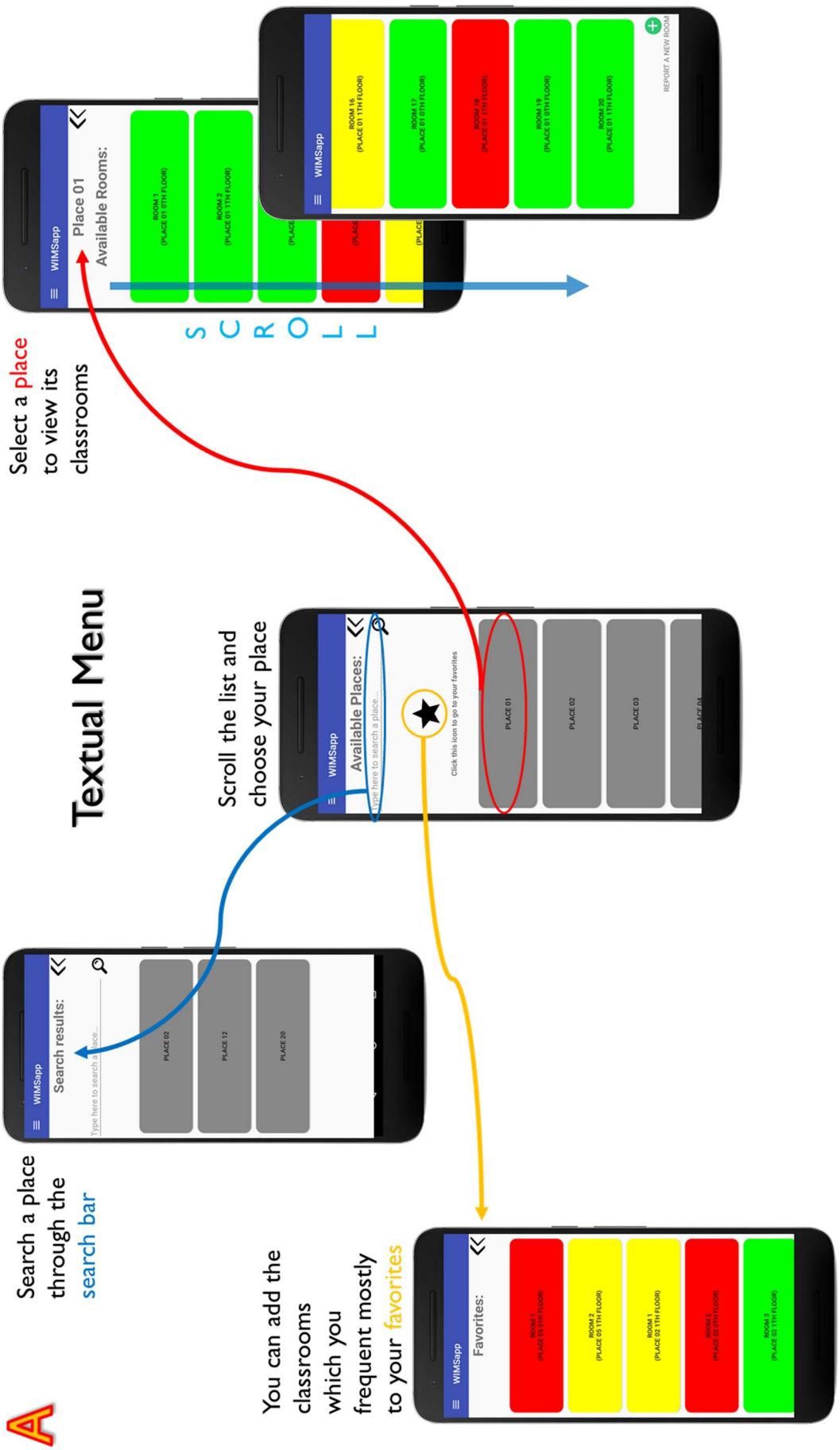


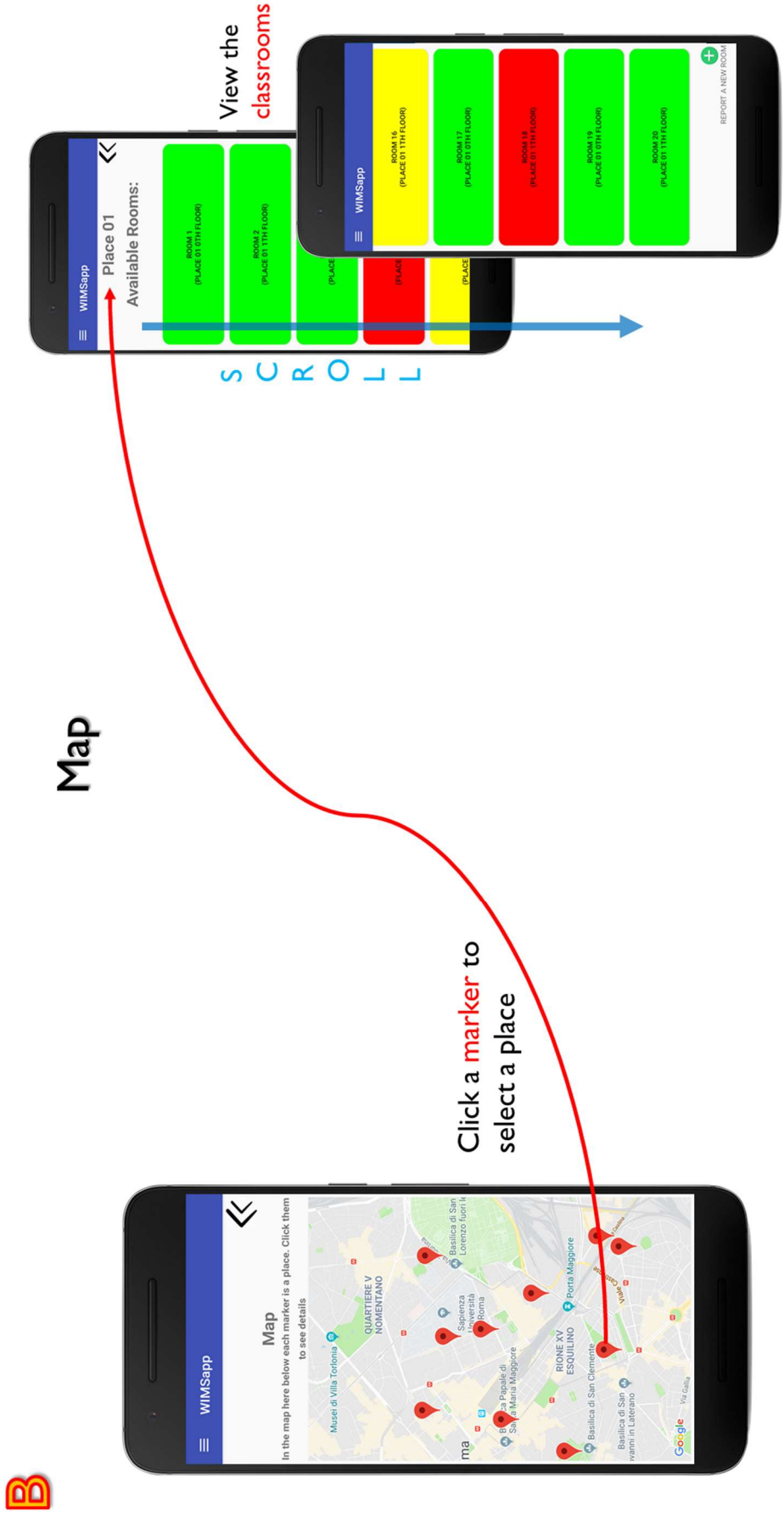
8.2 Story Board

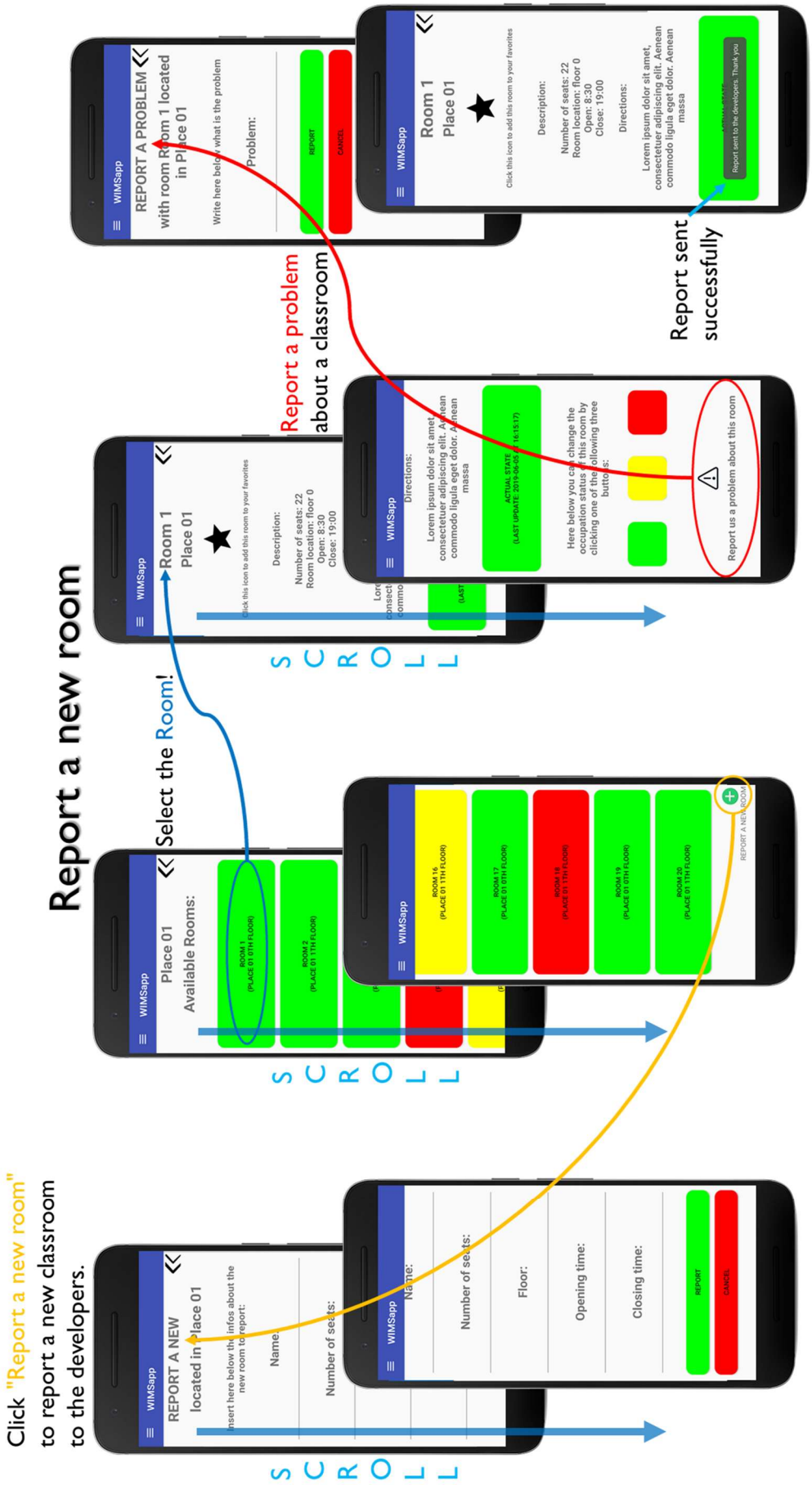
In the next pages there is the story board that shows all the aspects and the functionalities of our WIMSapp.


WIMSapp
WIMSapp hsyeeret?
This is Where Is My Seat app!
Hello user
If you have a Google account you can log in in this app simply by clicking the button below
Accedi
WIMSapp
Log out
Welcome to Where Is My Seat app
Are you searching a place where you can study in your free time? Are you in your new university for the first time? Or do you know very well your university but you find always your favourite rooms full of people? We have the solution for you! Use our textual menu in which all the rooms are divided on the base of their location or our map to find rooms where you can go! Take advantage by the report system based on the help of our users to know if a room is full of people or not! In few words: use our WIMSapp!
Here below you can find a quick menu to choose how to search your future seat
HERE YOU CAN NAVIGATE A TEXTUAL MENU TO FIND AVAIDABLE ROOMS
HERE YOU CAN FIND A USEFUL MAP TO SEARCH AROUND YOU
Delete my account
WIMSapp
WIMSapp hsyeeret?
Home
Textual Menu
Map
Log out
Delete my account
Available Places:
Type here to search a place...
Click this icon to go to your favorites
PLACE 01
PLACE 02
PLACE 03


1 Log in through the institutional email
WIMSapp
WIMSapp here?
This is Where Is My Seat app!
Hello user
If you have a Google account you can log in in this app simply by clicking the button below
Sign in
2 Choose the map or the textual menu
SCROLL
A.3 Textual menu
WIMSapp
Log out
Welcome to Where Is My Seat app
Are you searching a place where you can study in your free time? Are you in your new
WIMSapp
Available Places:
Type here to search a place...
Click this icon to go to your favorites
PLACE 01
PLACE 02
PLACE 03
PLACE 04
very well your university but you find always your favourite rooms full of people? We have the solution for you! Use our textual menu in which all the rooms are divided on the base of their location or our map to find rooms where you can go! Take advantage by the report system based on the help of our users to know if a room is full of people or not! In few words: use our WIMSapp!
Here below you can find a quick menu to choose how to search your future seat
HERE YOU CAN NAVIGATE A TEXTUAL MENU TO FIND AVAIDABLE ROOMS
HERE YOU CAN FIND A USEFUL MAP TO SEARCH AROUND YOU
Delete my account
Textual
Map
B.3 MAP
WIMSapp
Map
In the map here below each marker is a place. Click them to see details
Musei di Villa Torlonia
QUARTIERE V NOMENTANO
Basilica di San Lorenzo fuori le
Sapienza Università di Roma
Porta Maggiore
RIONE XV ESQUILINO
Basilica di San Clemente
Basilica di San Giovanni in Laterano
Via Gallia
Google


A
Search a place through the search bar
Textual Menu
Scroll the list and choose your place
Select a place to view its classrooms
You can add the classrooms which you frequent mostly to your favorites
WIMSapp
Search results:
Type here to search a place...
PLACE 02
PLACE 12
PLACE 20
Available Places:
Click this icon to go to your favorites
PLACE 01
PLACE 02
PLACE 03
Place 01
Available Rooms:
ROOM 1 (PLACE 01 0TH FLOOR)
ROOM 2 (PLACE 01 1TH FLOOR)
ROOM 16 (PLACE 01 1TH FLOOR)
ROOM 17 (PLACE 01 0TH FLOOR)
ROOM 18 (PLACE 01 1TH FLOOR)
ROOM 19 (PLACE 01 0TH FLOOR)
ROOM 20 (PLACE 01 1TH FLOOR)
REPORT A NEW ROOM
Favorites:
ROOM 1 (PLACE 05 0TH FLOOR)
ROOM 2 (PLACE 05 1TH FLOOR)
ROOM 1 (PLACE 02 1TH FLOOR)
ROOM 2 (PLACE 03 0TH FLOOR)
ROOM 3 (PLACE 02 1TH FLOOR)
SCROLL


B
Map
Click a marker to select a place
View the classrooms
SCROLL
WIMSapp
Map
In the map here below each marker is a place. Click them to see details
Place 01
Available Rooms:
ROOM 1 (PLACE 01 0TH FLOOR)
ROOM 2 (PLACE 01 1TH FLOOR)
ROOM 16 (PLACE 01 1TH FLOOR)
ROOM 17 (PLACE 01 0TH FLOOR)
ROOM 18 (PLACE 01 1TH FLOOR)
ROOM 19 (PLACE 01 0TH FLOOR)
ROOM 20 (PLACE 01 1TH FLOOR)
REPORT A NEW ROOM


Report a new room
Click "Report a new room" to report a new classroom to the developers.
Select the Room!
Report a problem about a classroom
Report sent successfully
SCROLL
WIMSapp
REPORT A NEW room located in Place 01
Insert here below the infos about the new room to report:
Name:
Number of seats:
Floor:
Opening time:
Closing time:
REPORT
CANCEL
Place 01
Available Rooms:
ROOM 1 (PLACE 01 0TH FLOOR)
ROOM 2 (PLACE 01 1TH FLOOR)
ROOM 16 (PLACE 01 1TH FLOOR)
ROOM 17 (PLACE 01 0TH FLOOR)
ROOM 18 (PLACE 01 1TH FLOOR)
ROOM 19 (PLACE 01 0TH FLOOR)
ROOM 20 (PLACE 01 1TH FLOOR)
REPORT A NEW ROOM
Room 1
Place 01
Click this icon to add this room to your favorites
Description:
Number of seats: 22
Room location: floor 0
Open: 8:30
Close: 19:00
Directions:
Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa
ACTUAL STATE (LAST UPDATE: 2019-06-05 16:15:17)
Here below you can change the occupation status of this room by clicking one of the following three buttons:
Report us a problem about this room
REPORT A PROBLEM with room Room 1 located in Place 01
Write here below what is the problem
Problem:
Report sent to the developers. Thank you

9. Conclusion and Future Work

9.1 Conclusion

We succeeded in finishing this WIMSapp and it is ready to be used by needful students. We always followed User Centered Design.

9.2 Future Work

We may try to spread WIMSapp all over the world by adding to it new rooms, places and universities.

In this way, we could try to help as needful students as possible!