Software Engineering

2022./2023.

Smart Home

Documentation, Rev. *<1 or 2>*

Group: *CHATEAUfort*

Leader: *Alexia Baudrand*

Date: *18. 11. 2022.*

Teacher: *Nikolina Frid*

Contents

1. Documentation change log

Continuous updating

Rev.	Change description	Authors	Date
0.1	Functional requirements, Other requirements	Alexia Baudrand, Paul Guyard, Jaques-Erwann Lostec, Roméo Tavernier, Apolline Savi	23/10
0.2	Use cases	Alexia Baudrand, Paul Guyard, Jaques-Erwann Lostec	12/11
0.5	Functionnal requierements, Use cases, project assignment requirement	Alexia Baudrand	16/11

Continued on next page

Continued from previous page

Rev.	**Change description**	**Authors**	**Date**
0.6	Sequence diagram	Apolline SAVI	17/11
0.8	Use case diagram	Jaques-Erwann Lostec	18/11
0.10	Database diagram	Paul Guyard	18/11
0.11	Class diagram	Roméo Tavernier	18/11
0.12.1			
0.12.2			
1.0			
1.1			
1.2			
1.3			
1.5			
1.5.1			
2.0			

There must be major revisions of documents 1.0 and 2.0 at the end of the first and second cycles. Between these revisions, there may be minor revisions depending on how the document is updated. It is expected that after each significant change (addition, modification, removal of parts of the text and accompanying graphic content) of the document, this will be recorded as a revision. For example, revisions within the first cycle will be labeled 0.1, 0.2,…, 0.9, 0.10, 0.11 .. until the final revision of the first cycle 1.0. In the second cycle, revisions 1.1, 1.2, etc. are continued.

2. Project assignment description

part of the 1st revision

Describe user requirements in detail based on the project assignment. Describe the goal of the project task as clearly as possible, elaborate the problem of the task, add new aspects of the problem and potential solutions. A minimum of 3, preferably 4-5 pages of description is expected. Topics that need to be further elaborated in this chapter are :

- *potential benefit of this project.*
- *existing similar solutions (explore and briefly describe the differences in relation to the given task). Add images that represent similar solutions .*
- *a set of users who might be interested in the solution achieved .*
- *the ability to customize the solution*
- *scope of the terms of reference*
- *possible upgrades.*

For help, see the references in the "References" section and, if necessary, consult online content that offers good guidance in this regard.

The main goal of the smart home project is to implement a mobile application able to control various appliances and meters remotely. To achieve that goal, we used several software engineering tools. This project is designed to help connected homeowners to handle their devices. The possibility to manage the house completely from a single app would be a considerable game changer as well as a huge gain of time. First, we made a quick survey of what the expected finished product should enable, allowing us to come up with a list user requirement allowing to overcome the problems we might have encountered along the way.

In that kind of software, the first requirement is to allow the user to create an account. To ensure that only one account is created per person, we chose to define each new account by the connected homeowner email address. To provide the best possible security a password was necessary. Allow anybody to have access to your home 's appliances would be a clear security breach. As such, the first user requirement is to Register into our application. To get a realistic application, several information must be entered before an account is created, like Name, Surname, username, email address and the password. To ensure no mistakes are made while creating the password, asking for a confirmation seen like the easiest way. Several accounts linked to the same email address would be problematic, to prevent it we created a database tasked with saving each account to prevent anyone from doing so. When registered the connected homeowners have to login into the app, our second user requirement. As soon as logged in, the home management must be the focus of the app. To produce a clear interface, we chose to break the devices into their concerned virtual home. Creating several virtual homes can become a game changer, being able to start the heating devices before coming home from the holidays for example is going to add comfort to the connected homeowners. Each home will be name to convenience making the app fully adaptable to the situation. Once the virtual homes created, they are useless without devices. To avoid mixed up the devices will be placed in their respective home. The focus of the app being to ease the household management, adding new devices need to be intuitive and fast. Scan a QR-code placed on the appliances allows that. Once those devices added they become remotely controllable, making the need to be on site to handle them obsolete. With the rise of energy prices, being able to monitor live and have access to a retrospective of the home and each devices consumption helpful. Being able to spot a malfunction or a leak remotely allows connected homeowners to handle the problem more efficiently. To that end, each device once connected has an info page, in which basic data is available as well as a support system. The ability to avoid spending hours on the phone wit a repair shop to try finding the problem seemed appealing. To that end, contacting the closest repair shop never have been that easy, a button in each device info page has been created to that end. Phone call being time consuming, emails are easier to handle on both ends. The app will allow to send an email using the login email address with all the necessary data already filled. That way everybody can tend to those problems and no expertise is needed from the homeowner.

The potential benefit from this project would be an easily managed household and the ability to detect and deal with dysfunctions while being on site or remotely, the connected homeowners would become completely autonomous. Being intuitive, the project would benefit everybody having a Wi-Fi connection and devices recent enough to be considered as connected.

Similar solutions, applications are already exiting nowadays, for example some brands have developed a connected range of items with an app to control them remotely like Phillips with their smart light bulbs, or the smart speaker from Sonos. With the same goal, brands created connected outlet allowing to switch on or off the plugged I device, like Meross.

Figure 2.1: Philips version of a connected home

Figure 2.2: Meross version of a connected home

This kind of app always have the potential to evolve. For example develop the software in different languages making it available worldwide. We could also implement a voice command mode making the research of a precise information easier. In order to be more inclusive, we could build a version with an audio-description mode allowing partially or completely blind to use the app.

3. Software specification

3.1 Functional requirements

Stakeholders:

1. Connected Home Owners
2. Systems Administrators
3. Repair shop

Actors and their functional requirements:

1. Connected Home Owners can:
   (a) Register
   (b) Login
   (c) View list of all the virtual Homes
   (d) View list of all the Devices
   (e) Add Home
   (f) Add Device
   (g) Delete Home
   (h) Delete Device
   (i) View Device info
   (j) Monitoring Device consumption
   (k) Monitoring House consumption
   (l) Remote Control
   (m) Send an Email to repair shop
   (n) View a list of all working Devices in real time
   (o) Logout

2. Systems Administrators can:
   (a) Enter informations about devices

3. Devices (passive actor):

   (a) Be control remotely (Appliances)
   (b) Measure consumption (Meters)

4. Firebase (passive actor):
   (a) Stock connected home owner profile data
   (b) Stock list of homes
   (c) Stock list of devices
   (d) Stock device status
   (e) Stock meters data

5. Email server (passive actor):
   (a) Stock email address of the connected home owner
   (b) Stock previous emails
   (c) Stock email address of the repair shop

3.1.1 Use cases

part of the 1st revision

Use case description

UC1 - Register

- **Main participant:** Connected Home Owners
- **Goal:** Gain an access to the application
- **Participants:** Database, email server
- **Prerequisites:** None
- **Description of the basic course:**
  1. The connected home owners input name, surname, username, email and password
  2. The system stores the new accounts into the database
  3. The newly registered connected home owners are redirected to the login page
- **Description of possible deviations:**
  - An account is already registered with the input email address
    - 1.a The system invites the connected home owners to try again with a different email address
    - 1.b The system redirects the connected home owners to the login page
  - The connected home owners fail to confirm the password to create the account
    - 2.a The system invites the connected home owners to try again
  - The connected home owners register an invalid email address
    - 3.a The systems invites the connected home owners to try with a different one

UC2 - Login

- **Main participant:** Connected Home Owners
- **Goal:** Access the application
- **Participants:** Database
- **Prerequisites:** None
- **Description of the basic course:**
  1. The connected home owners enter username and password

2. The system verifies in the database the submitted informations
3. The connected home owners log in and gets access to the application content

- **Description of possible deviations:**
  - The system is unable to connect with the database
    1.a The system sends an error messages advising to try again later
  - The system sends a notification for invalid informations input
    2.a The connected home owners modify the invalid information
    2.b The connected home owners choose to reset his login informations via an automatic email
  - The connected home owners don't have a account
    3.a The system redirects the connected home owners to the register page (UC1)

UC3 - View a list of all the virtual home

- **Main participant:** Connected Home Owners
- **Goal:** : Being able to see all the virtual home connected in an account
- **Participants:** Database
- **Prerequisites:** None
- **Description of the basic course:**

1. The systems shows a list of all the virtual homes created
2. The system proposes to create a new virtual home

- **Description of possible deviations:**
  - There isn't any virtual home:
    1.a Create a new virtual home
  - All the virtual homes aren't listed:
    2.a Check the WIFI router in the house to make sure the virtual home is reachable
    2.b Reboot the WIFI setup and the app

UC4 - View a list of all the devices

- **Main participant:** Connected Home Owners
- **Goal:** : Being able to see all the device in one virtual home at the same time

- **Participants:** Database, Devices
- **Prerequisites:** View a list of all the virtual homes (UC3)
- **Description of the basic course:**

  1. The system shows a list of all the appliances and meters installed in the designated home
  2. The system allows to add new devices

- **Description of possible deviations:**

  – Nothing is appearing on the screen:

    1.a Add devices to the virtual home

  – All the devices aren't showing up in the list :

    2.a Make sure to be in the right home

    2.b Install the appliances in the designated home

    2.c Check the WIFI router in the house to make sure the devices are connected and reachable

    2.d Reboot the WIFI setup and the app

UC5 - Add Home

- **Main participant:** Connected Home Owners
- **Goal:** Add one or several virtual home linked to the connected home owners account
- **Participants:** Database
- **Prerequisites:** View a list of all the virtual homes (UC3)
- **Description of the basic course:**
  1. The connected home owners select the add home option
  2. They enter the name and location of the virtual home
  3. They save the new home into the database
- **Description of possible deviations:**

  – Error in the location

    1.a The connected home owners change the location with a viable one

    2.b The connected home owners cancel the virtual home creation

  – The connected home owners leave without saving

    2.a The system suggests to save the changes

    2.b The connected home owners exit without saving, no virtual home is created

UC6 - Add device

- **Main participant:** Connected Home Owners
- **Goal:** Add one or several devices
- **Participants:** Database, Devices
- **Prerequisites:** View a list of all the virtual homes (UC3), View a list of all the devices (UC4)
- **Description of the basic course:**
  1. The home owners select the home to which the device is going to be be add
  2. Click on the add option to open the QR code scanner
  3. Scan the code placed on the new device to set it up
  4. Verify the data
  5. Choose a name for the device
  6. Add to the virtual home
- **Description of possible deviations:**
  - The home doesn't exist:

    1.a The connected home owners modify the invalid information and choose a valid house

    1.b The connected home owners create a new one
  - The Qr code isn't viable:

    2.a The connected home owners try to scan it again

    2.b The connected home owners choose to file by hand a form to add the new device

UC7 - Delete Home

- **Main participant:** Connected Home Owners
- **Goal:** Remove a virtual home from the application
- **Participants:** Database
- **Prerequisites:** View a list of all the virtual homes (UC3)
- **Description of the basic course:**
  1. Connected home owners select the delete home option
  2. The system asks for a confirmation before the deletion of the home
- **Description of possible deviations:**

  - Connected home owners get a notification warning that this action is irreversible
    1. The connected home owner proceed with the deletion
    2. They cancel the deletion

UC8 - Delete Device

- **Main participant:** Connected Home Owners
- **Goal:** Remove one or several devices from a virtual home
- **Participants:** Database, Devices
- **Prerequisites:** View a list of all the virtual homes (UC3), View a list of all the devices (UC4)
- **Description of the basic course:**
  1. The connected home owners select the device to delete
  2. The connected home owners click on the chosen device
  3. The connected home owners choose the delete option
  4. The system asks for a confirmation before the deletion of the device
- **Description of possible deviations:**
  - The connected home owners get a notification warning saying that this action is irreversible

    1.a The connected home owners proceed with the deletion

    1.b The connected home owners cancel deletion

UC9 - View device info

- **Main participant:** Connected Home Owners
- **Goal:** Give basic data on each device such as the name, ref, date of purchase, and basic functioning methods and the contact of the repair shop.
- **Participants:** Database, Device
- **Prerequisites:** View a list of all the devices (UC4)
- **Description of the basic course:**
  1. The connected home owners select a device
  2. The connected home owners have access to basic data about the device
  3. The connected home owners can monitor the device consummation
  4. The connected home owners monitor the state of the device (on/off, settings)
  5. The connected home owners can contact the repair shop by email if needed a button for that purpose

- **Description of possible deviations:**
  - The device doesn't exist:
    - 1.a The connected home owners modify the invalid information and choose a valid device
    - 1.b They create a new one

UC10 - Monitoring device consumption

- **Main participant:** Connected Home Owners
- **Goal:** : Monitor the consumption of each device in water, electricity and gas
- **Participants:** Database, Devices, Email server
- **Prerequisites:** View a list of all the devices (UC4)
- **Description of the basic course:**
  1. The connected home owners select the device under review
  2. They select the type of consumption that they want to check (electricity, gas, water)
- **Description of possible deviations:**
  - The device doesn't exist:
    - 1.a The system send you a warning message
    - 1.b The connected home owners modify the invalid information and choose a valid device
  - The consumption of the device is higher than expected:
    - 2.a The connected home owners check the use time of the device in order to rule out the overuse as a reason for the high consumption
    - 2.b The connected home owners suspect a malfunction and send an email to the repair shop

UC11 - Monitoring house consumption

- **Main participant:** Connected Home Owners
- **Goal:** : Monitor the energy consumption of the virtual home
- **Participants:** Database, Devices
- **Prerequisites:** View a list of all the virtual homes (UC3)
- **Description of the basic course:**
  1. The connected home owners select the house in which consumption they are interested in
  2. They select the type of consumption that they wants to check (electricity, gas, water) for the house as well as the time frame

- **Description of possible deviations:**
  - The consumption of the house is higher than expected:
    - 1.a The connected home owners check the time frame to be sure of an over-consumption
    - 1.b They check each device to find the one working erratically
    - 1.c They contact the repair shop via email
  - The time frame selected isn't available
    - 2.a The system has already deleted the data because of the data turnover
    - 2.b The data isn't available yet because the selected time frame hasn't pass yet

UC12 - Control device remotely

- **Main participant:** Connected Home Owners
- **Goal:** : Be able to use a device wherever the connected home owner is
- **Participants:** Database, Devices
- **Prerequisites:** View a list of all the devices (UC4)
- **Description of the basic course:**
  1. The connected home owners select a device
  2. they choose to turn it on or off, or even more advanced setting like choosing the temperature or the program of the appliance
- **Description of possible deviations:**
  - The device is already being used in a program (dishwasher, washing machine etc...)
    - 1.a The system sends a notification, giving the time at which device will become available
  - The system can't seem to connect with the device
    - 2.a Check the internet connection of the phone
    - 2.b Check the working status of the WiFi router in the house

UC13 - Send an email to the repair shop

- **Main participant:** Connected Home Owners
- **Goal:** : Repair a non-working appliance
- **Participants:** Database, Devices, Email server
- **Prerequisites:**View a list of all the virtual homes (UC3), View a list of all the devices (UC4), View device info (UC9)

- **Description of the basic course:**
  1. The connected home owners select the " Contact a repair shop" button in the device info
  2. The system automatically uses the owners email address to send the email
  3. They just have to explain the problem encountered so that the repair shop can solve it as soon as possible
- **Description of possible deviations:**
  - The repair shop is closed :

    1.a For a short period: the problem will be treated as soon as possible

    1.b For a longer period (more than a week): The email will be transferred to an other repair shop in the area
  - The device is not fixable :

    2.a The connected home owners have to replace it

<u>**UC14 - View a list of all working devices in real time**</u>

- **Main participant:** Connected Home Owners
- **Goal:** : Being able to see all the devices turned on in a selected virtual home in live
- **Participants:** Database, Devices
- **Prerequisites:** View a list of all the virtual homes (UC3), View a list of all the devices (UC4)
- **Description of the basic course:**
  1. The systems shows a list of all the devices being use in the house
  2. The systems allows to make sure no unwanted devices stays home when the connected home owners leave the house for example
- **Description of possible deviations:**
  - There isn't any devices working:

    1.a The systems sends a notification informing the connected home owners
  - All the working devices aren't listed:

    2.a Check the WIFI router in the house to make sure the devices are connected and reachable

2.b Reboot the WIFI setup and the app

UC15 - Logout

- **Main participant:** Connected Home Owners
- **Goal:** : Logging out of the application
- **Participants:** Database, Devices
- **Prerequisites:** None
- **Description of the basic course:**

  1. The connected home owners goes into his account settings
  2. The systems allows you to disconnect from the account by using the disconnect button

- **Description of possible deviations:**

  - None

Use case diagrams

Show the relationship of actors and patterns of use with the appropriate UML diagram. It is not necessary to draw everything on one diagram. Model by levels of abstraction and sets of related functionalities.

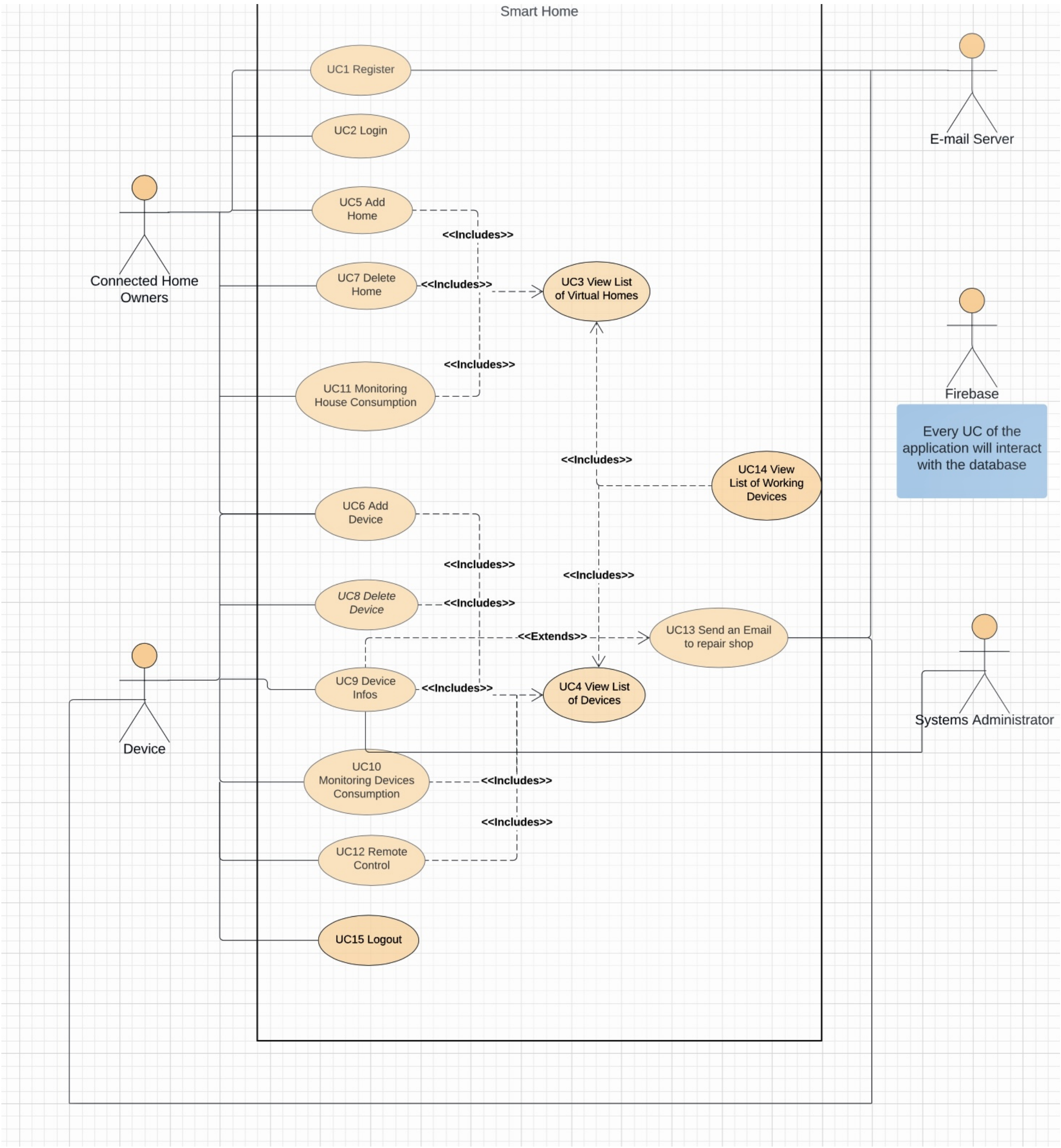

Figure 3.1: Use Case Diagram

3.1.2 Sequence diagrams

part of the 1st revision

Draw sequence diagrams that model the most important parts of the system (max. 4 diagrams). If in doubt about the choice, clarify with the assistant. Write a detailed description of the diagram next to each diagram.

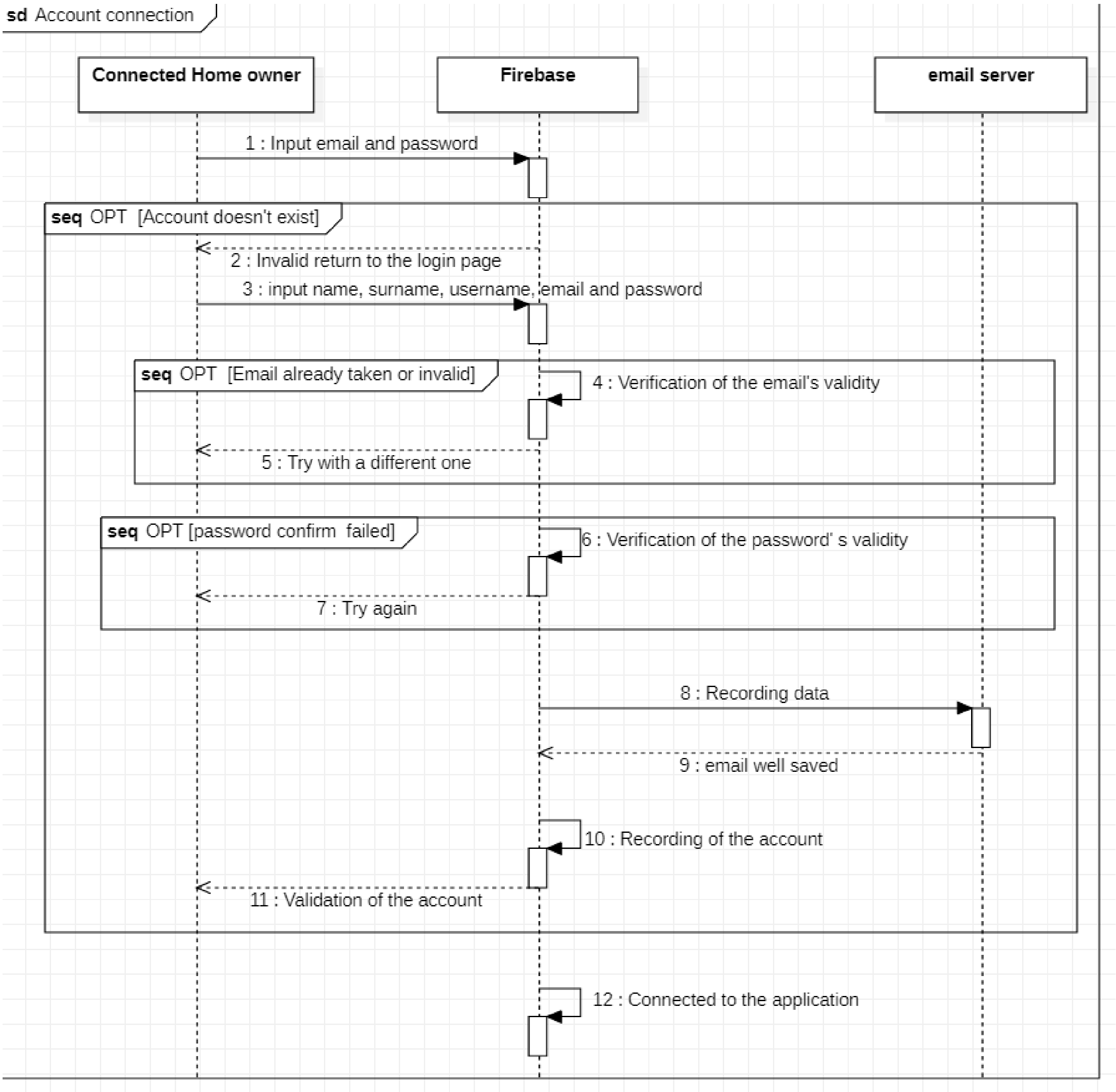
sd Account connection
Connected Home owner
Firebase
email server
1 : Input email and password
seq OPT [Account doesn't exist]
2 : Invalid return to the login page
3 : input name, surname, username, email and password
seq OPT [Email already taken or invalid]
4 : Verification of the email's validity
5 : Try with a different one
seq OPT [password confirm failed]
6 : Verification of the password' s validity
7 : Try again
8 : Recording data
9 : email well saved
10 : Recording of the account
11 : Validation of the account
12 : Connected to the application

Figure 3.2: Creation of a new account - Sequence Diagram

To enter in the application the connected home owners have to input their email and their password to have access at their account if they don't have one they have to create it by putting their name, surname, username, email and password. When they want to connect themselves the connected home owners can meet several problems like an email already taken or an invalid one. In this case they must try with a different one. Another problem that can happen it's if the password confirm failed. For this it is necessary to try again or to click on the button “forgotten password” that send them an email in which they can establish again the password. If everything is okay, they can continue and have access to the application.

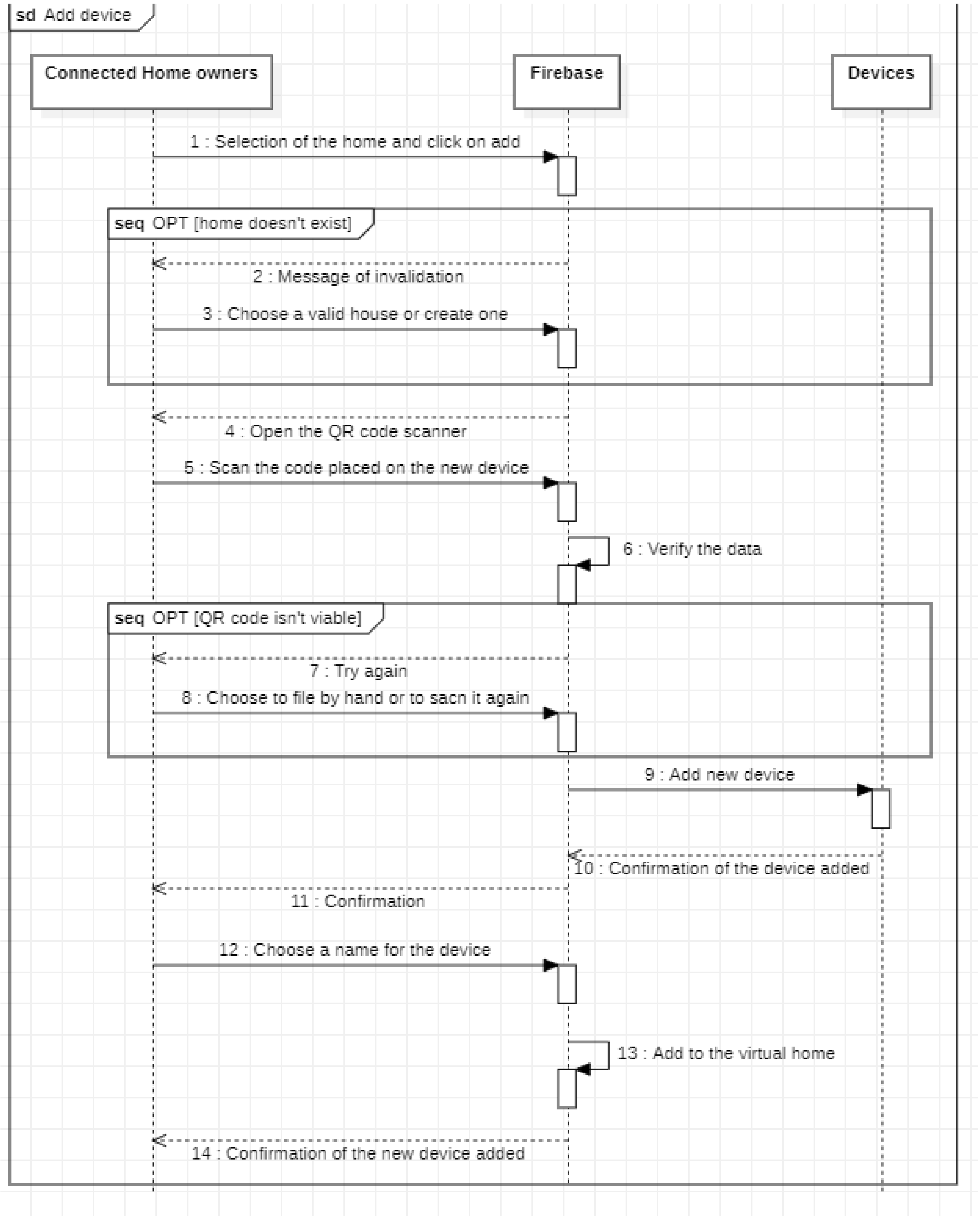
sd Add device
Connected Home owners
Firebase
Devices
1 : Selection of the home and click on add
seq OPT [home doesn't exist]
2 : Message of invalidation
3 : Choose a valid house or create one
4 : Open the QR code scanner
5 : Scan the code placed on the new device
6 : Verify the data
seq OPT [QR code isn't viable]
7 : Try again
8 : Choose to file by hand or to sacn it again
9 : Add new device
10 : Confirmation of the device added
11 : Confirmation
12 : Choose a name for the device
13 : Add to the virtual home
14 : Confirmation of the new device added

Figure 3.3: Add new device - Sequence Diagram

In the application the connected home owners can add a new device on the list to do this they have to select the home to which the device is going to be add. If the home doesn't exist, they try again until they choose a valid house or create one. After that they click on the add option to show a QR code scanner and must scan the code placed on the new device to set it up. If the QR code isn't available, they can file by hand or scan it again. The firebase will verify the data and after this operation the connected home owners can choose a name for the device and begin to use it.

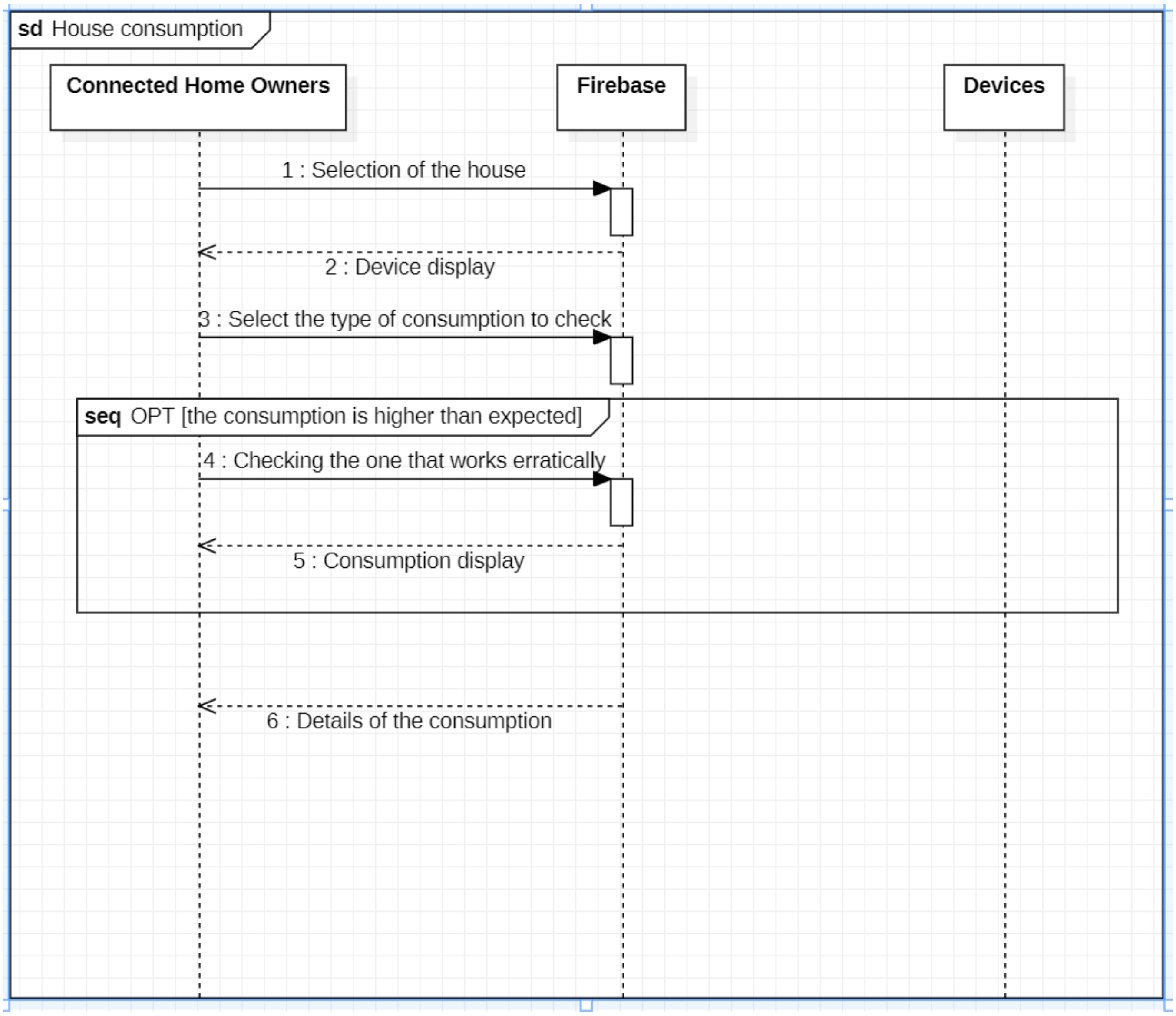
sd House consumption
Connected Home Owners
Firebase
Devices
1 : Selection of the house
2 : Device display
3 : Select the type of consumption to check
seq OPT [the consumption is higher than expected]
4 : Checking the one that works erratically
5 : Consumption display
6 : Details of the consumption

Figure 3.4: Display consumption - Sequence Diagram

Through the application the connected home owners can visualize the house consumption for this they have to select the house then select the type of consumption to check and after that a detailed list will appear with all the details of the device consumption.

3.2 Other requirements

Our application can be download using IOS or Android. It will be usable on mobile phones, tablets and computers. Our application will be in English to be understandable by everyone with a logical conception in order to be easily manipulated. We implemented a chain of command giving the opportunity for everyone to use their devices without loosing the control of the house despite being off-site.

4. System architecture and design

part of 1st revision

It is necessary to describe the architecture style and identify: subsystems, mapping to the work platform, data warehouses, network protocols, global control flow and hardware-software requirements. Elaborate by points and accompany with appropriate sketches:

- *choice of architecture basedname on the design principles shown in the lectures (explain why you chose such an architecture)*
- *organization of the system with the highest level of abstraction (eg client-server, database, file system, graphical interface)*
- *application organization (eg frontend and backend layers, MVC architecture)*

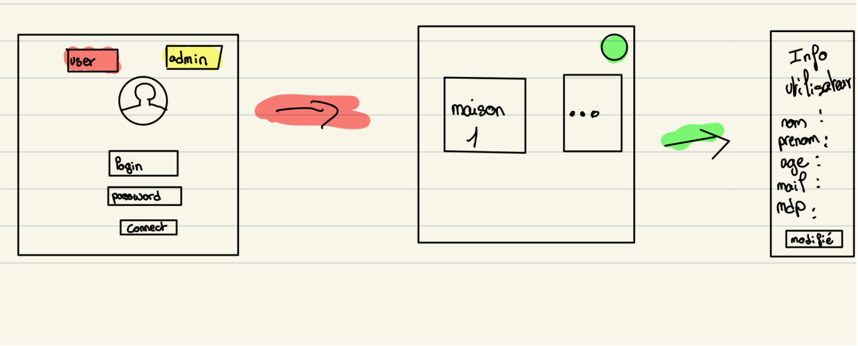

Figure 4.1: App utilisation(1)

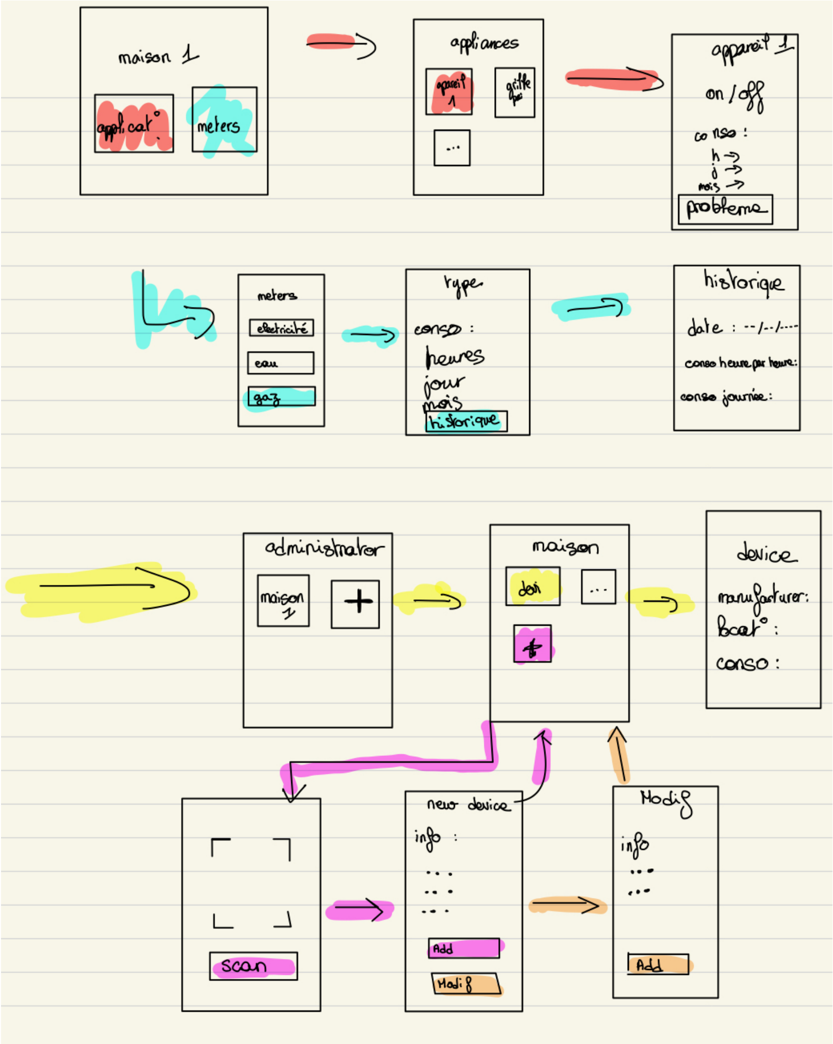

Figure 4.2: App utilisation(2)

These two pictures explain how the user will see the app and the link between each features. For example, when the user press the green bouton (in the first picture) he will can see account information. In the second picture it's the same functioning. The red way show the application way (put on or off devices) and the blue way show information about homes or devices (consumption of each device, make a report). The yellow part is about the administrator view (who can add or delete homes and devices.

4.1 Database

part of 1st revision

We use Vertabelo for the database diagram to create the drawing .

4.1.1 Table description

Each table must be described according to the template. On the left is the exact name of the variable in the database, in the middle is the data type, and on the right is the description of the variable. Highlight the primary key in light green and the foreign key in blue

type of house		
id	INT	id of the house to recognize the different house. the id is unique
place	VARCHAR	the adress of the house
size	VARCHAR	the size of the house to estimate the consumption
device	VARCHAR	the lsit of all the devices link to the house
user id	int	the id of the user of the house

type of device		
id	INT	id of the device to recognize the different device.
electrical consumption	VARCHAR	the consumption of the device and other information to use the device
state	VARCHAR	if the device is activate or not
name	VARCHAR	the name of the device
house id	int	the id of the house where is the device

type of meters		
id	INT	id of the meter to recognize the different meter. the id is unique to a type of meter
type	VARCHAR	the type of meter
device id	int	the id of the device affiliated to the meter

admin		
id	INT	id of the admin
custome user	VARCHAR	the capacity to add or delete a user
id of the house	int	the id of the house

custom user		
id	INT	id of the user
email	VARCHAR	the email of the user
password	VARCHAR	the password they have use to register
phone number	int	the phone number of the user
age	int	the age of the user
name	VARCHAR	the name of the user
surname	VARCHAR	the surname of the user
SAV id	int	the id of the SAV
house id	int	the id of the house

SAV		
id	INT	id of the SAV
email	VARCHAR	the email of the SAV

4.1.2 Database diagram

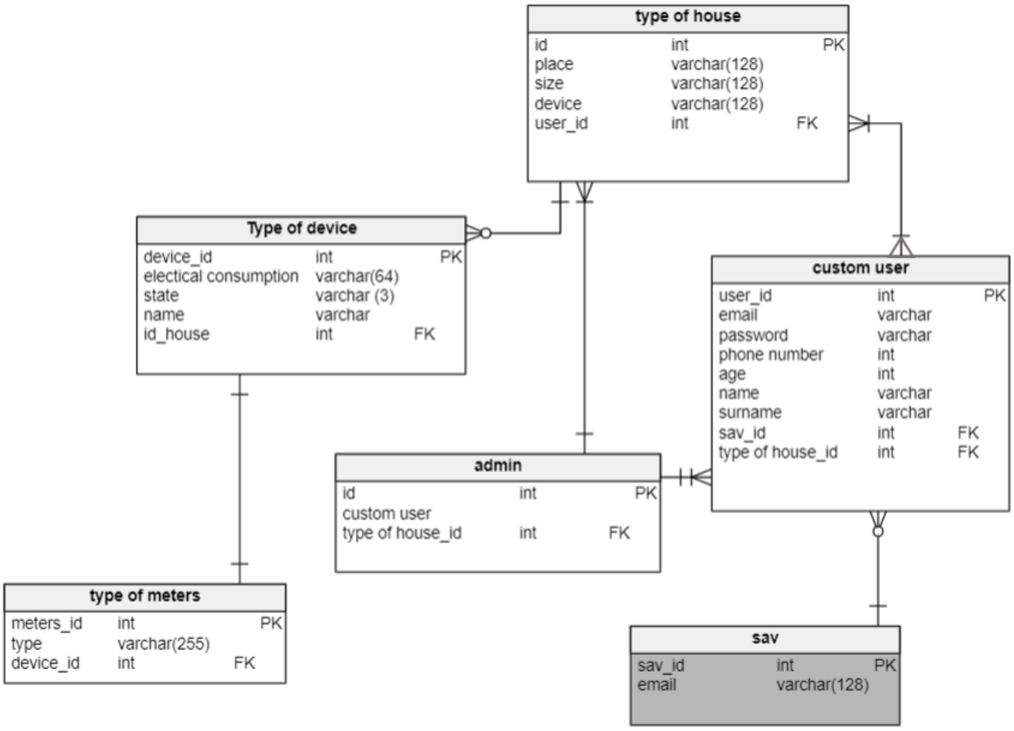

Figure 4.3: database diagram

4.2 Class diagram

A class diagram with a corresponding description must be attached. For clarity, it is possible to break the diagram into several, but they must be grouped according to similar levels of abstraction and related functionalities.

part of the 1st revision

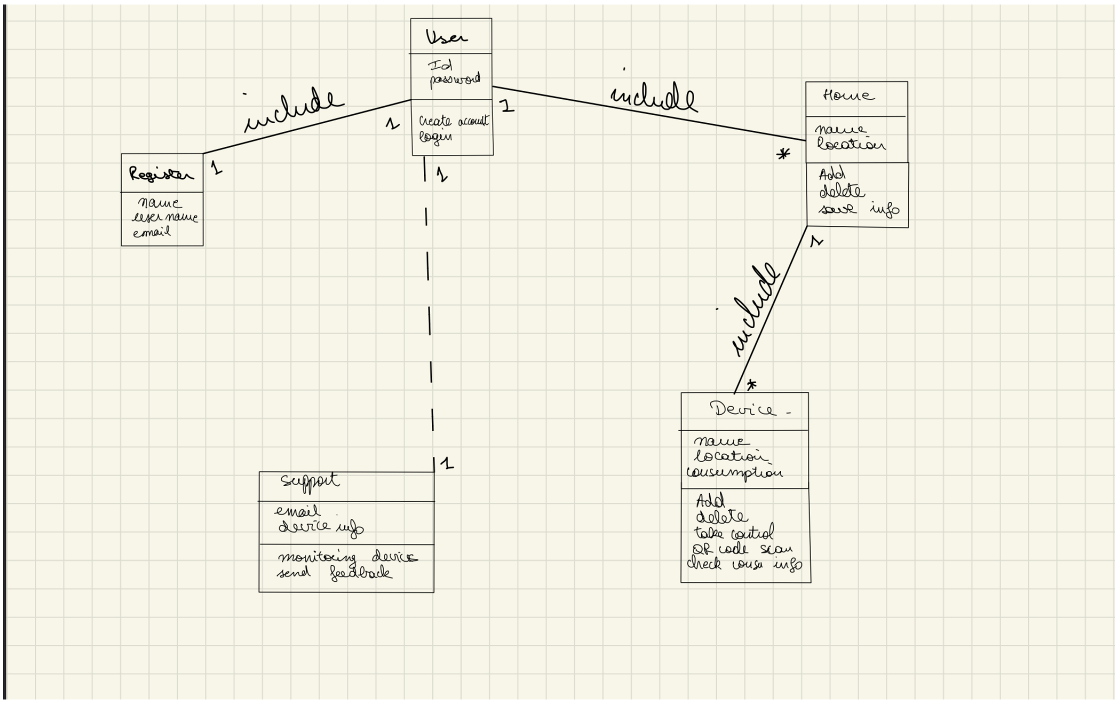

Figure 4.4: Class Diagram

The class diagram is use to explain the relationship between each things in the application. For example the users must create an account in the app(register). They need to log in to put new home. They need to choose a home to add, delete or see device information. They need to be log in to send email to the support, etc.

At the first submission of the project, it is necessary to attach a fully developed class diagram related to the ***generic functionality*** *of the system. Other functionalities should be conceptually elaborated in a diagram with the following components: class names, method names and types of access methods (eg public, protected), class attribute names, relationships and relationships between classes.*

part of the 2nd revision

During the second submission of the project, the class diagram and descriptions must correspond to the actual state of implementation.

4.3 State machine diagram

part of the 2nd revision

It is necessary to attach a state diagram and describe it. One state diagram showing the ***significant part of the functionality*** *of the system is sufficient. For example, user interface states and the flow of use of some key functionality are a significant part of the system, and registration and login are not.*

4.4 Activity diagram

part of the 2nd revision

It is necessary to enclose a attach of activities with a corresponding description. The activity diagram should show a significant part of the system.

4.5 Component diagram

part of the 2nd revision

A component diagram with the accompanying description must be attached. The component diagram should show the structure of the whole application.

5. Implementation and user interface

5.1 Tools and technologies

part of the 2nd revision

List in detail all technologies and tools used in the development of documentation and applications. Briefly describe them, and state their meaning and place of application. For each of the listed tools and technology it is necessary to ***specify internet link*** *where you can download or learn more about them.*

5.2 Software testing

part of the 2nd revision

In this chapter it is necessary to describe the implementation of testing of implemented functionalities at the level of components and at the level of the whole system with the presentation of selected test cases. Students should examine core functionality and boundary conditions.

5.2.1 Component testing

*It is necessary to conduct unit testing on classes that implement basic functionalities. Develop a **minimum 6 test cases** in which regular cases, boundary conditions, and exception throwing will be examined. It is also desirable to create a test case that uses functionalities that are not implemented. It is necessary to enclose the source code of all exam cases and a presentation of the results of the exam in the development environment (passing / failing the exam).*

5.2.2 System testing

The system test should be performed and described using the Selenium footnote `https://www.seleniumhq.org/` *framework. Develop a **minimum 4 test cases** that will examine regular cases, boundary conditions, and call functionality that is not implemented / cause an error to see how the system responds when something is not fully realized. The test case should consist of an input (eg username and password), the expected output or result, the test step and the output or result obtained.*

The creation of test cases using the Selenium framework can be performed using one of the following two tools:

- *browser extension **Selenium IDE** - recording user actions for automatic exam repetition*
- ***Selenium WebDriver** - support for writing exams in Java, PHP languages using a special programming interface*

Details on the use of the Selenium tool will be presented in a special lecture during the semester.

5.3 Deployment diagram

part of the 2nd revision

*You need to insert a **specification** layout diagram and describe it. It is possible to insert an instance layout diagram instead of a specification layout diagram, provided that this diagram better describes some important part of the system.*

5.4 Deployment instructions

part of the 2nd revision

In this chapter it is necessary to give instructions for deployment of the realized application. For example, for web applications, describe the process by which the source code leads to a fully set up database and server that responds to user queries. For a mobile application, the process by which the application is built and placed on one of the stores. For a desktop application, the process by which an application is installed on a computer. If mobile and desktop applications communicate with the server and / or database, describe the procedure for setting them up. When creating instructions, it is recommended that ***highlight installation steps using hints*** *and use* ***screenshots*** *as much as possible to make the instructions clear and easy to follow.*

The completed application must be running on a publicly available server. Students are encouraged to use one of the following free services: Amazon AWS, Microsoft Azure or Heroku. Mobile apps should be released on the F-Droid, Google Play or Amazon App Store.

6. Conclusion and future work

part of the 2nd revision

In this chapter it is necessary to write a review of the time of project assignment, what technical challenges have been identified, whether they have been solved or how they could be solved, what knowledge was acquired during project development, what knowledge would be especially needed for faster and better project implementation. and what would be the prospects for continuing work in the project team.

It is necessary to accurately list the functionalities that are not implemented in the realized application.

References

Continuous updating

List all references and literature that helped in the realization of the project.

1. Oblikovanje programske potpore, FER ZEMRIS, `http://www.fer.hr/predmet/opp`

2. I. Sommerville, ”Software engineering”, 8th ed, Addison Wesley, 2007.

3. T.C.Lethbridge, R.Langaniere, ”Object-Oriented Software Engineering”, 2nd ed. McGraw-Hill, 2005.

4. I. Marsic, Software engineering book“, Department of Electrical and Computer Engineering, Rutgers University, `http://www.ece.rutgers.edu/~marsic/books/SE`

5. The Unified Modeling Language, `https://www.uml-diagrams.org/`

6. Astah Community, `http://astah.net/editions/uml-new`

Index of figures

Appendix: Group activity

Meeting log

Continuous updating

It's necessary to frequently update the meeting log according to the template.

1. meeting
    - Date: in this format: November 18, 2022
    - Attendees: J.Doe, ...
    - Meeting subjects:
        - subject description
        - subject description
2. meeting
    - Date: in this format: November 18, 2022
    - Attendees: J.Doe, ...
    - Meeting subjects:
        - subject description
        - subject description

Activity table

Continuous updating

Note: Activity contributions should be entered as hours contributed by person and activity.

	Team lead	Team member	Ime Prezime	Team member	Team member	Team member	Team member
Project management							
Project task description							
Functional requirements							
Individual patterns description							
Patterns diagram							
Sequence diagram							
Other requirements description							
System architecture and design							
Database							
Class diagram							
State diagram							
Activity diagram							
Components diagram							
Used technologies and tools							
Solution testing							
Layout diagram							
Deployment instructions							
Meeting log							

Continued on next page

Continued from previous page

	Team lead	**Team member**	**Ime Prezime**	**Team member**	**Team member**	**Team member**	**Team member**
Conclusion and future work							
References							
Additional task examples							
Welcome page creation							
Database creation							
Connecting to the database							
back end							

Change log diagrams

part of the second revision

Import generated change log diagrams from GitLab to this chapter. Diagrams can be reached at GitLab at Repository/Contributors.