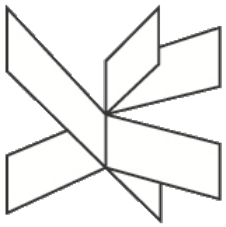

VIA University College

Project Report

Surfing-couch

Rosalie Peree (258377) Grégoire Cartier (258378)

Supervisor & Academic advisor: Kasper & Jakob

2017 (Autumn semester)

Contents

1 Abstract

2 Introduction

2.1 Background Description

2.2 Problem Description

2.3 Limitations and Delimitations

3 Methods

The following table presents our choice of model and methods for the problems that we listed in our project description (see Appendix #1):

What *Problem*	**Why** *Why study this problem*	**Which** *Which models/theories did you use to solve the problem?*
How can we develop a social application which is fun to use and rewards the user for helping travelers in their area?	How to make the application interesting for our potential users?	We have to pay attention to all the following problems to be able to develop an application that will address all of them as good as possible.
What is important for travelers?	As our application is mainly targeted to travelers, it is interesting to know what is important for them, so that we cant try to implement those things in the final application.	We conducted a survey via Google Forms to gather people's opinion on that subject.
What is important for hosts?	As we want hosts to offer services to travelers, we have to meet with what they want to encourage them to do so.	We conducted a survey via Google Forms to gather potential user's opinion on that subject.
How can we allow users to create and find trips in their area and offer a service to travelers?	This problem is the main problem for a couch-surfing-like application. We have to localize the user to be able to provide a service for them in a specific area.	We have to set up a way for the users to enter their city/location and look for hosts in cities they want to visit.
How can we make the service secure for users?	Users don't want their personal data to be forwarded to strangers.	No personal data is displayed on the application for users to see. We implemented an internal messaging system so people can communicate directly on the application.
Can we add reviews and comments about travelers and hosts?	If you have to host someone or sleep/shower at a stranger's place, safety is very important. Users need a way to see if someone behaved correctly or not before their stay.	We implemented a system of user reviews - with grades - so people can report how they experience was with that specific user - hosts and travelers.

What Problem	**Why** Why study this problem	**Which** Which models/theories did you use to solve the problem?
How can we define the point-value of the provided services?	As the system is based on a reward, all services must have a fixed value.	We defined a value for each service - sleep, shower and laundry. The points are given to the host once the booking is completed.
How can we exchange the points for gifts?	The system is based on rewarding hosts with gifts for providing a service.	We set up an online shop where users can exchange their points for goods.
How will the user interface be implemented?	The user interface has to be easy to get and use.	The interface will be an Android application.
What kind of tool will be used for this implementation?	The tools used to develop can make the process of developing the application more or less easy.	We decided to develop the application using Android Studio, as it is the platform we are both most familiar with.
How will the interface be user-friendly?	The interface has to be easy to use and understand for any user.	The interface is simple, with buttons and labels that are easy to understand.
How is the social interaction going to take place?	The point of the assignment is to enable a social interaction.	In our application, the social interaction will take place in the form of direct messaging between two users, a common chat for all the users and the possibility of reviewing experience with other users.
What kind of database will be implemented?	We need a database that is free and easy to integrate in Android.	The chosen database is Firebase for our project.

4 Requirements

4.1 List of Requirements

4.1.1 Functional Requirements

4.1.2 Non-Functional Requirements

5 Analysis

6 Design

6.1 Architecture

6.2 Technologies

6.3 Design Patterns, Class Diagram, Sequence Diagrams

7 Implementation

8 Test

8.1 Test Specifications

8.2 White Box Testing

9 Results & Discussion

10 Conclusion