Ride Sharing Website

COMPX374-20B (HAM) Software Engineering Industry Project

Deliverable I

Software Requirement Specification

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

12th August 2020

Team Members:

Aaron Win

Lysa Phan

Victor Yao

Ryan Good

Table of Contents

1. Introduction

1.1 Purpose 2

1.2 Scope 2

1.3 Definitions, acronyms, and abbreviations 2

1.4 Overview 3

2. Overall description

2.1 Product perspective 4

2.2 Product functions 4

2.3 User characteristics 4

2.4 Constraints 4

2.5 Assumptions and dependencies 5

3. Specific requirement

3.1 External Interface 6

3.2 Functional Requirements 6

3.3 Performance Requirements 8

3.4 Logical Database Requirements 8

Appendices 9

1. Introduction

1.1 Purpose

This report aims to present detailed requirements of the Ride Sharing website. It contains all functions, specifications and constraints of the website. This documentation is written for both the software development team and the clients. It will also be presented to the convenors of the course Software Engineering Industry Project as a part of required deliverable for this course for evaluation purpose only.

1.2 Scope

At this stage, the official name of this website has not yet been decided. For ease of reference, the website for this project will be referred as “Ride Sharing website” throughout the process of designing and implementing, unless otherwise notified by the client. The Ride Sharing website will be designed to resolve current issues of public and private transportation. The target users of this website are university students who have difficulty to make effective use of public transport and people who wish to share the cost of petrol and maintenance for their vehicles. In addition, this system will make positive impacts on the environment such as reducing traffic congestion and air pollution. This web application is created for people who wish to share a ride with each other. All verified users can view or post a request to look for other people who desire to travel to the same destination at the same time. They can make an arrangement to travel in the same vehicle instead of using public transport or using multiple private vehicles. For this version, the web page will not include the payment functionality due to legal reasons. The users need to arrange the payment by themselves to share the cost of car ownerships.

1.3 Definitions, Acronyms and Abbreviations

Term	Definition, Acronyms & Abbreviation
Software Requirements Specification	A documentation that gives a full description of the functions and constraints of a system.
Member	Verified user of the website who has provided their legal identification.

Time Frame	The period of time that the user is able to leave the start location.

1.4 Overview

There are two sections in the remainder of this report, which are Overall Description and Specific Requirement. Section 2, Overall Description, aims to give an overall description which will explain the general purpose and functionality of the web app. This will give a clear overview and understanding of the project as a whole. Section 3, Specific Requirement, is specifically written for the developers. It describes the functionality of the web page in details and how each part of the web application should operate.

2. Overall Description

2.1 Product perspective

The primary goal of this web application is to let users find other people to share rides with. In specific, users can visit the website to view requests from other people or post an offer to share a ride with someone who has the same travel destination.

2.2 Product functions

- User can log in via Facebook to become a member.
- Member can act as a passenger or a driver (given a valid driver's license) for a ride share.
- Member can post an offer where they act as the driver who wishes to use their own vehicle to carry other passengers.
- Member who wants to become passengers can negotiate and/or accept the available offers.
- Member can post a request to look for a ride as a passenger if they cannot find an available offer that suits them.
- Member can exchange details once a match is made, in order to discuss the details and times of the ride.
- Member can give a rating to other members who they shared the ride with.

2.3 User Characteristics

The users of the web application are students who attend The University of Waikato. This user group is expected to be familiar with the concept of ride sharing and the general use of the web application or other similar tools. For future development, the user group can expand to people who are student from different schools or people who are not a student.

2.4 Constraints

- Time Frame - The web application will need to be completed by November 6, 2020. This includes all the stages of creating this system being done, which are requirement, design, implementation and testing.
- Web based - The application will be designed primarily to be used on a PC or Mobile Phone.
- User group - The target users at this stage are only students in the University of Waikato.

2.5 Assumptions and Dependences

- Members will be able to access the web page from wherever they are, as long as they have an internet connection.
- Members will keep the web page open while they intend to use its functionality.

Apportioning of Requirements:

Expansion of users allowed to join and use the web application.

3. Specific Requirements

3.1 External Interface

- **Facebook verification**
  Allows the user to log into the web page by using third party which is a trusted and secure service.

- **Google Map API**
  An in-built map that will allow users to see the general location of the people in their ride, so they can see how close they are to being picked up, for example. API data will allow the web app to calculate the relative distances between people.

- **Select time frame to find matches**
  The user will be able to choose a timeframe that suits their plans for that journey, meaning that the matches they find will get them where they want to go on time.

- **Database**
  The database will store data pertaining to users and offers to be used to facilitate users creating offers and joining them, plus other features like user profiles.

3.2 Functional Requirements

Step #1: A wellbeing verification page where users confirm their wellbeing through a tick box before entering the application.

Step #2: Facebook Verification: User clicks on “Facebook” button which will redirect them to the Facebook verification page. After the user enters their Facebook username and password, they will be redirected back to the web page to proceed the next step. 1. Create a user database and store the user's name.

Step #3:

- After returning from the Facebook verification page as a verified user, it takes members 6 inputs to complete a request:

1. Select the role: driver/passenger/both
2. Select a date & time frame xx:xx-xx:xx (Validity check on date and time frame is valid)
3. Starting place/destination (Address → geographical coordinates → Google Map interface)
4. Party number (Validity check on number<5 to have at least one seat left)
5. Contact number

- Inputs 1-5 are stored in the user's database.
- "Make request" button, to the match result page. Store the request time in the database

Step #4: Going through a list of users who already made the request, comparing the five inputs they made in the previous step, comparing inputs 1-4 in the previous step:

1. At least one driver in one ride.
2. On the same date, having a common period of time in their time frame selection, store the common timeframe in the database for the result page.
   When 0 matches, such as a person's timeframe does not overlap with anyone:
   Option 1: The information of the request would be kept in the database waiting for users to match.
   Option 2: User going back to Step #3 to make a new request, his previous data would be replaced.
3. Using coordinates to get the distance between the users and stored as a variable that will be displayed on the match result page.
4. Calculate total people and store in the database, the number must be less than 5.
5. Check that users have not already made a match with others. If already with others, check if there are still any seats in the ride.
6. Write JavaScript to pick out a list of users who fulfil the criteria listed above.
7. Pass these users info onto the next page in the application.

Step #5: Results Page:

- Display a list of users, ordered by
- Variables in the user's info (from user's database):

  #2.1 name

#3.1 role

#4.2 common timeframe

#3.3 starting place/destination

#3.4 party number

#3.5 contact number

#4.4 Number of people already in one ride (could be more than two parties already)

- Click on the "Accept" button. Update the total passenger number **#4.4** (#number of people in the party + 1) (#id of the accepted parties)

Step #6: When user's timeframe expired (real time > #3.2), delete user's variables.

3.3 Performance Requirements

In the project, the product is a web-based app. Therefore, it can support many terminals and simultaneous users at one time. The information being handled will be data pertaining to the offers (Timeframe, Location, Vehicle capacity). 95% of the interactions with the database shall be processed in 1 s.

3.4 Logical Database Requirements

A table containing the data pertaining to a specific Ride Share offer shall be created when a new offer is created. This table will store the Start and Expiry Times for the offer, the general location of the organiser/driver, and the capacity of their vehicle. It also stores the user's information such as their name, email address, phone number and their rating.

Appendices

#5 Meeting with Convenors: The software development team and the convenors discussed Software Development methodologies and the team then decided to use SCRUM for the project. The team also gave an update on their process.

No.	Date & Time	Duration	Attendees	Agenda/Note
1	28th July 2020 @11.38am	33 minutes	Victor Yao Aaron Win Lysa Phan	• Introduction • Preparing questions for client • Clarification of requirements • Paper prototype
2	29th July 2020 @1pm	45 minutes	Victor Yao Aaron Win Lysa Phan Ryan Good *Rohesia*	• Introduce the group to the client • Clarify some requirements • Show the paper prototype to the client
3	30th July 2020 @1.50pm	40 minutes	Victor Yao Aaron Win Lysa Phan Ryan Good	• Discuss software to use • Visit links from client
4	31st July 2020 @2.15pm	90 minutes	Victor Yao Aaron Win Ryan Good	• Sort out requirements
5	3rd Aug 2020 @11.00am	20 minutes	Victor Yao Aaron Win Lysa Phan Ryan Good *Convenors*	• Decide on Software Development methodology • Take notes for next meeting with Lecturers
6	4th Aug 2020 @2.30pm	30 minutes	Victor Yao Aaron Win Lysa Phan Ryan Good	• Decided to use the SCRUM software methodology
7	5th Aug 2020 @1.00pm	30 minutes	Victor Yao Aaron Win Lysa Phan Ryan Good *Rohesia*	• Go over requirements (so far) with client

8	5th Aug 2020 @1.35pm	50 minutes	Victor Yao Aaron Win Lysa Phan Ryan Good	• Using Scrum to produce the backlog and sprint • Design the web pages • Note: Read section 5 in the requirement template on Moodle