**Software Engineering Requirements**

**DRIVEUP**

**Students: Harun Šaranović, Irfan Durić, Ahmed Nurović,**

**Haris Kovačević, Dino Šaciragić**

**Group: 7**

**Course name: Software Engineering**

**Project name: DriveUP**

**Abstract**

For our project we are going to make web application that will be „middle man“ between drivers and customers. It will provide a service to customers that are looking for someone to drive them around the county. It is something like uber driver but our web application will be connection between them. Our main goal is to provide application to people who are going to register as drivers and will use our web application to give services to customers.

**Document Revision History:**

19. March 2019 – Initial Software Requirements version (Abstract, requirements specification)

20. March 2019 – Added functional and non-functional requirements, use cases, realease plan and timeline table

**Services provided to users**

We will have two types of users: drivers and customers who are looking for a driver. Drivers will have to register on application and they will need to put some brief informations about them like their name, phone number, email, what car are they driving, when they are available, how much money are charging, in which cities are they able to go, etc. And second users are customers that are looking for driver. They will have to register and after registration they will be able to search for drivers by categories such as price, city, number of seats etc. Customers will also be able to rate drivers and leave review or comment. Drivers and customers will be able to contract each other by private message or calling number shown in drivers description or through email. There are non – functional system requirements that have to met by users. This will be brief explanation but you will be able to see it in details later on. Minimum operating system should be Windows 7, software requirement for system ( Chrome, Firefox , Visual Studio Code), software requirement for client ( Chrome, Firefox) and, users need to use internet connection to access the system via browser ( internet connection). Our web application will fit well in overall business since a lot of tourists are visiting our country and they need driving service which we will easily provide to them. Our application will also have three different languages ( English, Bosnian and Arabic), and this will attract more users(tourists) since most of them speak Arabic and almost everyone speaks English. Our application will depend on how many users we have, drivers and customers. But we are going to advertise our application and since it will be helpful, easy to use it will attract users.

**System requirements specification**

This Web application is a middleman application which connects the driver to the person that wants to hire them on a daily basis or for a trip around Bosnia and Herzegovina. This application functionalities are : Easy to use, Reliability, User-Friendly, Availability. The Hardware requirements aren't that demanding since it's a web application, therefore the only important requirement for this section would be the version of their web browser. Since most people are using Google Chrome and their web browser are up-to-date with web development coding and not to cause cross-browser compatibility issues we recommend our users to use Google Chrome.

**Minimum hardware requirements :**

1. Windows 7
2. Intel Pentium 4 processor
3. Google Chrome 66.

**Optimal hardware requirements :**

1. Windows 10 or later
2. Intel Pentium dual core or higher
3. Google Chrome 73 (The latest version)

**Functional Requirements**

1. **Register (Must have):** Register button should be located on the home page. When clicked it should navigate to Register page which will have a register form. Anybody can register and they are required to insert their First name, Last name, Contact number, Email, Car they drive, Languages they speak, Locations they go to, number of seats, car specs, dates when they are available to take clients, price and minimum one picture of the car they are offering. After they submit their profile will be reviewed and after it is reviewed it will be posted on our Drivers page.
2. **Login (Must have):** The drivers that are registered will need to log into their accounts in order to receive messages from their clients and to edit the dates they are available or unavailable. They will have a simple form with email and password which they need to insert.
3. **Search (Must have):** The client needs to be able to search and filter to find the specific desires they need. They will pick their preferences and the search will filter the drivers available for their needs. They can filter by dates when they need a driver, by price, by car seats available, languages the driver speaks and location.
4. **Message (Must have):** The client that wants to contact a driver they seem fit must be able to send a direct message to their driver. The average response time of the driver will be shown so that the client knows approximately when to expect an answer. The client and the driver must be able to chat in real time once they have reached contact.
5. **Rating (Must have):** After the client and the driver are finished with the service the client will be available to rate the driver by the driving service, hospitality, work flexibility and overall driving experience. The rating will be done with stars starting from 0 to 5 stars and the client will be available to give half start reviews. After the start review the client will be available to write down any notes they have about the experience.
6. **Profile management (Must have):** The driver needs to be able to edit their profile to change, add or delete anything after they have registered. The driver needs to be able to change the dates when they are available and unavailable.
7. **Driver list (Must have):** The client can navigate to the drivers page form the header menu. In the drivers page all of the available drivers will be listed. The client can open the drivers profile if they are interesting in a specific driver. In the driver list page the client will be able to filter and search for drivers.

|  |  |
| --- | --- |
| Actor | Use Cases |
| * Driver | 1. Log in 2. Post Transport Service 3. Remove/Edit Transport Service 4. Go to ‘’Silent Mode’’ 5. Direct Message 6. Edit Profile 7. Delete Profile |
| * Customer | 1. Log in 2. Lookup for Transport service 3. Search by price, city, number of seats 4. Direct Message 5. Edit Profile 6. Rate Drivers 7. Lookup Driver profile 8. Check for available Drivers 9. Check for available destinations and cities 10. Order Transport service/drivers 11. Check driver ratings 12. See online drivers |
| * Guest | 1. Register as driver or customer 2. Lookup Drivers profile 3. Lookup available drivers/destinations 4. Check drivers rating |

# **3.1 Use Cases**

**3.1.1 Register**

Purpose of this use case is to make account for the user

**ID**: FR1

**Priority**: High

**Pre**-**condition**: Internet

**Post-condition**: Account created

**Basic flow**:

1. User enter homepage url
2. Click on register button in navigation bar
3. Select which type of account, enter information and click register button

**What can go wrong**:

If password does not meet required needs or if the username is already taken, exception message should appear, else than that account is created.

**3.1.2 Login**

Purpose of this use case is to user can login into his/her account after creating it

**ID**: FR2

**Priority**: High

**Pre-condition**: Created account

**Post-condition**: Logged in account

**Basic flow**:

1. User enter homepage url
2. Click on Login button in navigation bar
3. Enter username/email and password, press Log in button

**What can go wrong:**

If user enters wrong username or password, an error message should appear and they should be return to Log in page to retry enter username and password.

**3.1.3 Search**

Purpose of this use case is to describe how user can easily make search for desired options

**ID**: FR3

**Priority**: High

**Pre**-**condition**: enter URL homepage

**Post-condition**: search by desired criteria

**Basic flow**:

1. User types homepage URL
2. User click Search button in navigate bar
3. User can enter in a search bar for a specific destination or driver
4. User can use filter search to specify which car / destination / number of seats / price they search for
5. After typing those information, user click search button
6. Displays offers that matches user criteria

**What can go wrong**: -

**3.1.4 Post Transport Service**

Purpose of this use case is to explain how drivers can post their services

**ID**: FR4

**Priority**: High

**Pre-condition**: Account created as Driver

**Post-condition**: Transport service posted and viewable by customers

**Basic flow**:

1. User types homepage URL
2. User click Log in button in navigate bar, enter username and password and click Log in
3. Click in menu Post Transport Service button
4. Enter required information about service and press Post button

**What can go wrong**: If user does not enter all required information, the system should display warning message and label the field that is not entered.

**3.1.5 Message**

**ID**: FR5

Purpose of this use case is to describe a communication link between driver and customer

**Priority**: High

**Pre-condition**- Account created and logged in as customer/driver

**Post-condition**- Customer and driver are able to communicate

**Basic flow**:

1. User types homepage url
2. User navigates to Log in button and make log into his/her account
3. Click on Message button
4. From there user will be able to see inbox (Previous chat history) as well as make new message to some other registered user

**What can go wrong**: If user get disconnect, message may be not delivered

* + 1. **Rating drivers**

Purpose of this use case is to give option to a previous customer to help future customers to choose right driver for their desires.

**ID**: FR6

**Priority**: Medium

**Pre-condition**: User has to be logged as customer and already done with ordered transport service

**Post-condition**: Customer will rate driver based on his/her satisfaction with driver and make easier decision for future customers

**Basic flow**:

1. User log in as Customer
2. Order a transport service
3. After finishing transport service, log in again and go to that ordered transport service post
4. Click Rate driver button
5. Give your satisfaction with driver by giving starts from 0 to 5 or leave a comment

**What can go wrong:**

If user leave a blank comment, error message should appear and take user back to rate driver page.

* + 1. **Profile Management**

Purpose of this use case is to describe how user can make changes to his/her existing profile

**ID**: FR7

**Priority**: High

**Pre-condition**: User has to be registered and logged in

**Post-condition**: Modify profile information, password, picture…

**Basic flow**:

1. Log into account
2. Go to Profile button, after that go to edit profile
3. Make changes such as edit first name, last name, change picture, change password
4. Click Save button

**What can go wrong**: If user enter information that are same as previous or type password that does not meet requirements, warning message should appear and user should be able to write it again.

* + 1. **See available destinations**

Purpose of this use case is to explain how customer can gather information about destination that they are interested in

**ID**: FR8

**Priority**: High

**Pre-condition**: There is no pre-conditions for this, because it can be done by any actors and actions

**Post-condition**: Potentially customers can take a look for available destinations that drivers offer

**Basic flow**:

1. Go to home URL page
2. Go to Destinations button from navigation bar
3. Check for available dates and destinations that drivers have posted

**3.1.9 Online drivers list**

Purpose of this use case is to explain how user can take a look at online drivers at that moment

**ID**: FR9

**Priority**: Medium

**Pre-condition**: User has to be logged in as customer or driver

**Post-condition**: Check for online drivers and get faster responds

**Basic flow:**

1. Log into account
2. Go to Find Transport button in navigation bar
3. After that, select Online drivers from the menu and check who is online

**What can go wrong:** There shouldn’t be anything that can go wrong, because there is not so much actions from user, this depends on system mostly.

**3.1.10 Drivers list**

Purpose of this use case is to describe how customer can see profiles and offers by a specific drivers

**ID**: FR10

**Priority**: High

**Pre-condition**: -

**Post-condition**: Potential customer can check drivers profiles

**Basic flow:**

1. Go to home page URL
2. Select Drivers from navigation bar
3. See a list of drivers
4. Click driver profile to see more information about driver

**What can go wrong**: Here everything depends on system, so user shouldn’t be able to make anything wrong.

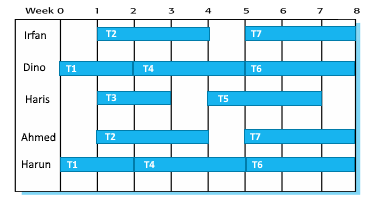
**3.2 Non-Functional Requirements**

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement** | | **Definition** | **More** **Details** |
| NFR1 - Operating system | | Minimum operating system | Windows 7 / 8 / 8.1 |
| NFR2 - Software Interface | | Software requirement for the system | Chrome / Firefox, Visual Studio Code |
| NFR3 - Software Interface | | Software requirement for the client | Chrome / Firefox |
| NFR4 - Communication Interface | Use needs connection to the internet to access the system via browser | | Internet connection |

**3.3. Release Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Requirement** | **Duration** | **Increment** | **Priority** | **Dependencies** | **Release** |
| FR1 | 1 day | 1 | High, must have |  | 1 |
| FR2 | 1 day | 1 | High, must have | FR1 | 1 |
| FR3 | 3 days | 3 | High, must have | FR4 | 1 |
| FR4 | 2 days | 2 | High, must have | FR2 | 1 |
| FR5 | 3 days | 4 | High, must have | FR2 | 1 |
| FR6 | 2 days | 5 | Medium | FR5 | 1 |
| FR7 | 1 day | 2 | High, must have | FR2 | 1 |
| FR8 | 1 day | 3 | High, must have | FR4 | 1 |
| FR9 | 1 day | 6 | Medium | FR2 | 1 |
| FR10 | 2 days | 3 | High, must have | FR1 | 1 |

**3.3.1 Timeline (Activity bar chart, or Gantt chart)**



Progress Report Table

**Meeting no:1**

**Date:14.3.2019**

**Members present: Dino Saciragic,Haris Kovacevic, Ahmed Nurovic, Harun Saranovic ,Irfan Duric**

**Members absent:-**

**Tasks:** **The Requirements Document**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Team member name** | **Task assigned** | **To be completed on** | **Last meeting tasks completed (Yes/No)** | **Signature** |
| Harun Saranovic | Presentation, release plan, timeline and user interface | 20.3.2019 | - |  |
| Ahmed Nurovic | System requirements specification, abstract | 20.3.2019 | - |  |
| Dino Saciragic | Functional Requirements | 20.3.2019 | - |  |
| Haris Kovacevic | Introduction | 20.3.2019 | - |  |
| Irfan Duric | Actor and Use Cases( TABLE ) + Use Cases | 20.3.2019 | - |  |