Software Requirements Specification

for

CarRental

Version 1.8 approved

Prepared by Piorescu-Oprea Ovidiu-Ilie

Munteanu Stefan

Cristi-Florin Neagoe

**Faculty of Automatics, Computers and Electronics**

06.03.2021

Table of Contents

Table of Contents 2

Revision History 3

1. Introduction 4

1.1 Purpose 4

1.2 Document Conventions 4

1.3 Intended Audience and Reading Suggestions 5

1.4 Product Scope 5

1.5 References 5

2. Overall Description 6

2.1 Product Perspective 6

2.2 Product Functions 6

2.3 User Classes and Characteristics 6

2.4 Operating Environment 6

2.5 Design and Implementation Constraints 7

2.6 User Documentation 7

2.7 Assumptions and Dependencies 7

3. External Interface Requirements 8

3.1 User Interfaces 8

3.2 Hardware Interfaces 9

3.3 Software Interfaces 9

3.4 Communications Interfaces 9

4. System Features 9

4.1 Create Account 9

4.2 Log in 10

4.3 Log out 11

4.4 Add Car 11

4.5 Remove Car 12

4.6 Check Car Availability 12

4.7 Search for Car 12

4.8 Rent a Car 13

4.9 Profile Manage 13

5. Other Nonfunctional Requirements 14

5.1 Performance Requirements 14

5.2 Safety Requirements 14

5.3 Security Requirements 14

5.4 Software Quality Attributes 14

5.5 Business Rules 15

6. Other Requirements 15

Appendix A: Glossary 15

Appendix B: Analysis Models 15

Appendix C: To Be Determined List 15

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Piorescu Oprea Ovidiu Ilie | 05.03.2021 | Adding Purpose (1.1), Adding Document Conventions (1.2), Adding Intended Audience and Reading Suggestions (1.3) | 1.1 |
| Munteanu Stefan | 05.03.2021 | Adding Product Scope (1.4), Adding References (1.5), Adding User Documentation (2.6) | 1.2 |
| Cristi-Florin Neagoe | 06.03.2021 | Adding User Classes and Characteristics (2.3), Adding System Features (4), Adding Operating Environment (2.4) | 1.3 |
| Piorescu Oprea Ovidiu Ilie | 08.03.2021 | Adding Product Perspective (2.1), Adding Product Functions (2.2), Adding Software Interfaces (3.3), Updating Document Conventions (1.2) | 1.4 |
| Munteanu Stefan | 08.03.2021 | Updated Document Conventions (1.2), Updating System Features (4), Adding Assumptions and Dependencies (2.7) | 1.5 |
| Cristi-Florin Neagoe | 09.03.2021 | Updating Product Functions (2.2), Updating User Classes and Characteristics (2.3), Updating References (1.5), Adding Software Quality Attributes (5.4) | 1.6 |
| Piorescu Oprea Ovidiu Ilie | 09.03.2021 | Adding Design and Implementation Constraints (2.5), Adding Performance Requirements (5.1), Adding Security Requirements (5.3) | 1.7 |
| Piorescu Oprea Ovidiu Ilie | 10.03.2021 | Updating Software Interfaces (3.3), Updating Document Conventions (1.2), Adding User Interfaces (3.1), Adding Communications Interfaces (3.4) | 1.8 |

# Introduction

The software described in this document is a car rental service website complete with associated support pages, user interfaces, database management that allows the reservations to be made.

This document seeks to provide the Software Requirements Specifications for the car rental website, and contains an introduction, overall description, specific requirements and non-functional requirements with respect to the current stage of the project.

## Purpose

The purpose of this document is to record the requirements for the design and

development of a car rental service website. It describes requirements for a future, complete version of the project, to be created in the spring of 2021. There may be a need for future updates to the document.

Moreover, this document reflects the current requirements of the project as understood by the project team, presenting an initial description of the various functionalities and services provided by the software. Additionally, it may also serve as a basis for testing by the user.

## Document Conventions

Abbreviations:

|  |  |  |
| --- | --- | --- |
| **No.** | **Abbreviation** | **Definition of abbreviation** |
| 1 | **XAMPP** | Cross-platform web server solution stack package. |
| 2 | **MySQL** | Structured Query Language |
| 3 | **UI** | User Interface |
| 4 | **SRS** | Software Requirement Specification |
| 5 | **OS** | Operating System |
| 6 | **PHP** | Personal Home Page, general-purpose scripting language |
| 7 | **HTML** | Hypertext Markup Language |
| 8 | **PC** | Personal Computer |
| 9 | **IP** | Internet Protocol |
| 10 | **HTTP** | Hypertext Transfer Protocol |
| 11 | **HTTPS** | Hypertext Transfer Protocol Secure |
| 12 | **CSS** | Cascading Style Sheets |

Fonts:

* Body: Arial, size 11
* Title: Times, size 18
* Subtitle: Times, size 14

## Intended Audience and Reading Suggestions

* People in need of transportation, especially tourists that need to visit multiple places in a short amount of time.
* Car enthusiasts, people that want to test the capabilities of a certain type of vehicle before deciding whether to purchase it or not.

We suggest you to read it only if you have some Java/Spring/MySQL knowledge.

## Product Scope

The goal of website is to encourage more people to choose rental services, eventually increasing the profit of the company, improvement of the services, and most importantly helping people with their business. Maintaining a simple and friendly site is deemed of highest importance in order to hold customer attention and guide the viewer to information that will lead to a decision to rent the best vehicle available.

## References

|  |  |  |
| --- | --- | --- |
| No. | Name | Link |
| 1 | Java Tutorials | [*https://www.javatpoint.com/java-tutorial*](https://www.javatpoint.com/java-tutorial) |
| 2 | Hibernate Tutorials | [*https://www.tutorialspoint.com/hibernate/index.htm*](https://www.tutorialspoint.com/hibernate/index.htm) |
| 3 | Spring Tutorials | [*https://www.baeldung.com/spring-tutorial*](https://www.baeldung.com/spring-tutorial) |
| 4 | List of computing and IT abbreviations | [*https://en.wikipedia.org/wiki/List\_of\_computing\_and\_IT\_abbreviations*](https://en.wikipedia.org/wiki/List_of_computing_and_IT_abbreviations) |
| 5 | Web Tutorials | *https://www.w3schools.com/* |
| 6 | Html + CSS + PHP-MySQL Tutorials | *https://marplo.net/* |

# Overall Description

## Product Perspective

This project’s purpose is to facilitate the use of a software system for the automation of the management process of the activity of a car rental service. A DB will be used to store the information about the users, cars and the branches.

This website has a multitude of functionalities. The customer will be able to log in, browse a variety of cars, customize his choices and rent the perfect vehicle. He will also be recommended other vehicles based on his choices. There are also options for newsletter, personal profile and other feedback functionalities. Guests can also view customer-type functionalities of the site, but their abilities are limited.

## Product Functions

* User can create an account
* User/Admin can log in
* User/Admin can log out
* User account can be deleted
* User/Admin can rent a car
* User/Admin can see all the cars he rented
* User/Admin can change his password
* User/Admin can search for a car
* User/Admin can see his profile information
* User/Admin can change his profile information
* The admin can see all the cars that are rented
* The admin can add a new car
* The admin can remove a car
* The admin can update information about a car
* The admin can see the information about the users

## User Classes and Characteristics

This website will be used by three types of users:

* Admin – the administrator will be able to access a database of customers, update certain parts of it and also add car listings for the customers to rent.
* Customer – The users which have logged in are considered customers. They will be able to rent a vehicle, view their rented vehicles, receive recommendations and use various interactive options on the website such as signing up for a newsletter and sending messages to the admin.
* Guest – Users which have not logged in, have limited accessibility to the website. They will need to log in to rent vehicles and use the rest of the services, however they will still be able to browse the website fully.

If it is necessary, the website should undergo design changes to all existing pages to shorten up paragraphs and add bullets and charts where possible to accommodate faster page reading and shorter reader attention span keeping in mind most of the users do not have great technical expertise and are looking for a simple and well-structured car presentation in order to rent it easier.

## Operating Environment

This software being developed will be compatible with machines running on Windows, Android & IOS. Website will run on Apache Server.

Development will be done using IntelliJ Idea, MySQL, Spring Framework and Thymeleaf. The website will be written using HTML, CSS and Java.

Google Chrome will be the main browser used to test the website, however it will be tested on Mozilla Firefox, Microsoft Edge, Opera as well.

All databases and functional components will be stored on a personal computer for testing purposes. Databases and servers are handled by IntelliJ Idea & MySQL Server.

The website will work on almost any machine, so the minimum requirements are trivial. The synchronization procedures will be written to interface with Windows operating systems. There are no software components that are in conflict with this project, the website can be tested as is.

## Design and Implementation Constraints

The employer must provide the following tools: MySQL Community Server 8.0.23, IntelliJ Idea 2.2, Microsoft Teams.

We have chosen the above IDE because this is the one, we are most familiar with.

## User Documentation

Customers with account will receive a user manual for the application. Support (Email) will be provided in case it’s necessary.

## Assumptions and Dependencies

The requirements needed to run the application:

* An internet connection with a speed of minimum 1MB/s
* A device (Phone – Android / IOS, PC) with a browser like Chrome, Mozilla Firefox, Opera, Internet Explorer etc.

# External Interface Requirements

## User Interfaces

The website will feature a home page which contains a summary of all the features, a simple, interactive presentation of some featured cars, statistics, stocks and sales information.

It will also feature an “about us” page containing information about the site owners and a car listing page which will contain the database of all available cars, complete with detailed information about all of their features and prices, including current availability.

There will also be a FAQ page with frequently asked questions about the car rental services and a “contact us” page containing contact information. There may also be buttons which contain links to social media.

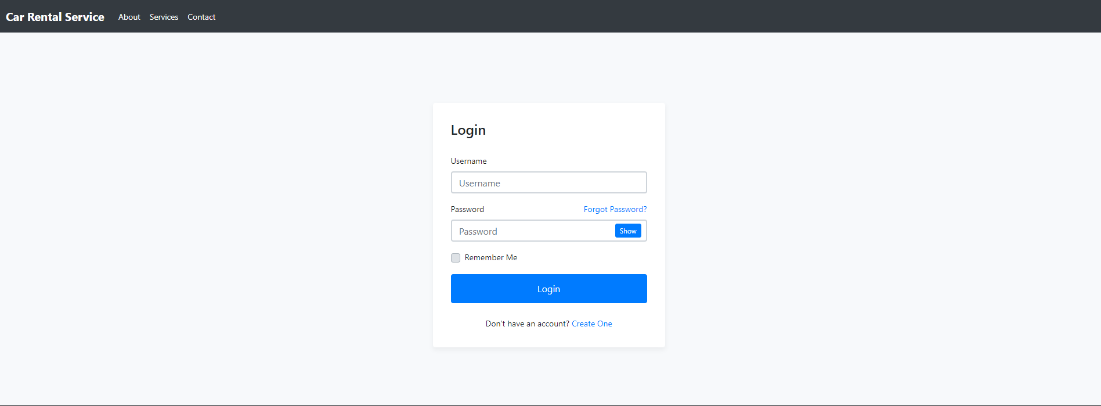
The user has the ability to log in and rent a car. There is a search bar which can easily find a specific car, or it can be selected from the car listing page. Users also have access to their personal profile where they can see the bookings they have made, and can be suggested cars similar to the ones they’ve booked.

Satisfied customers, a newsletter, and a special admin login button are also available.

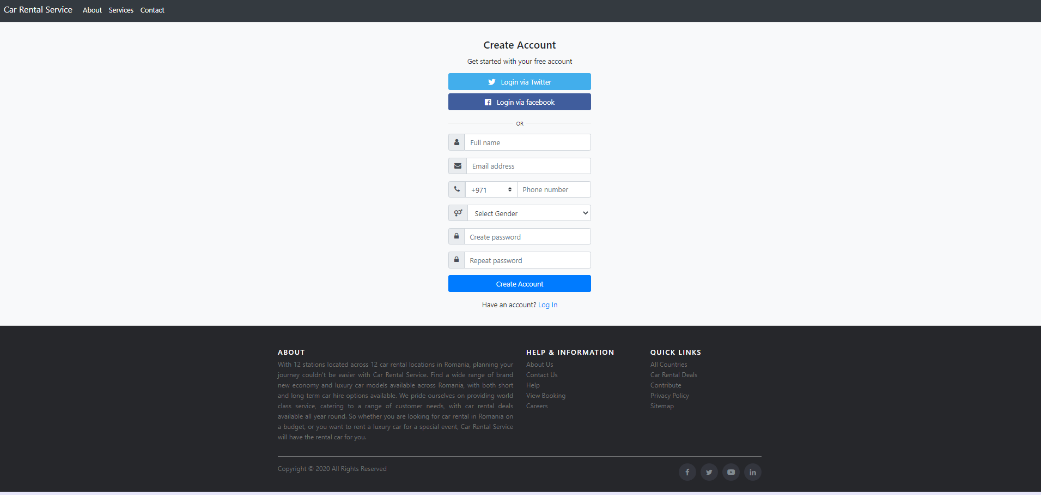
The website should work and be tested on Microsoft Edge, Firefox, Google Chrome.

Some screenshots with an early version model have been attached.

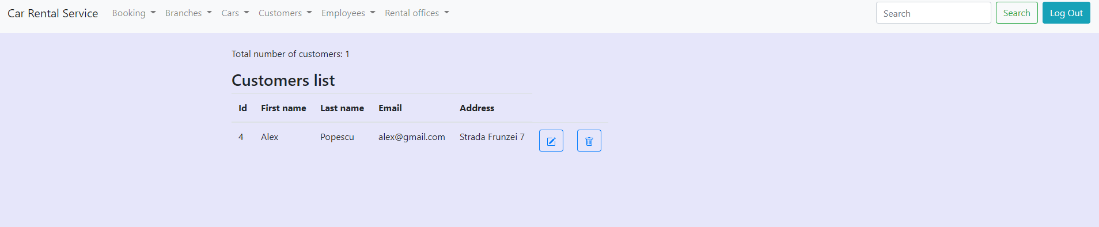
Login Page:



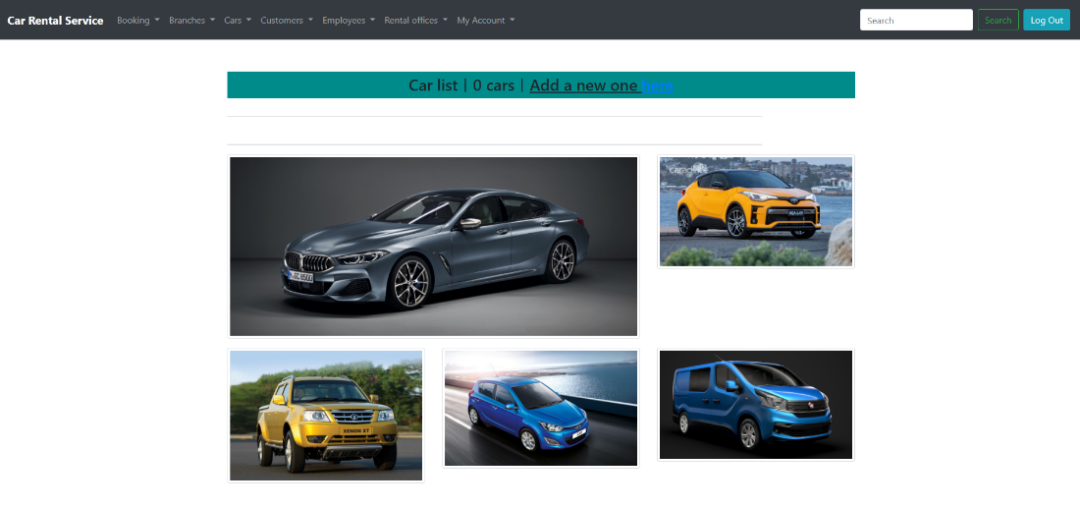
Registration Page:



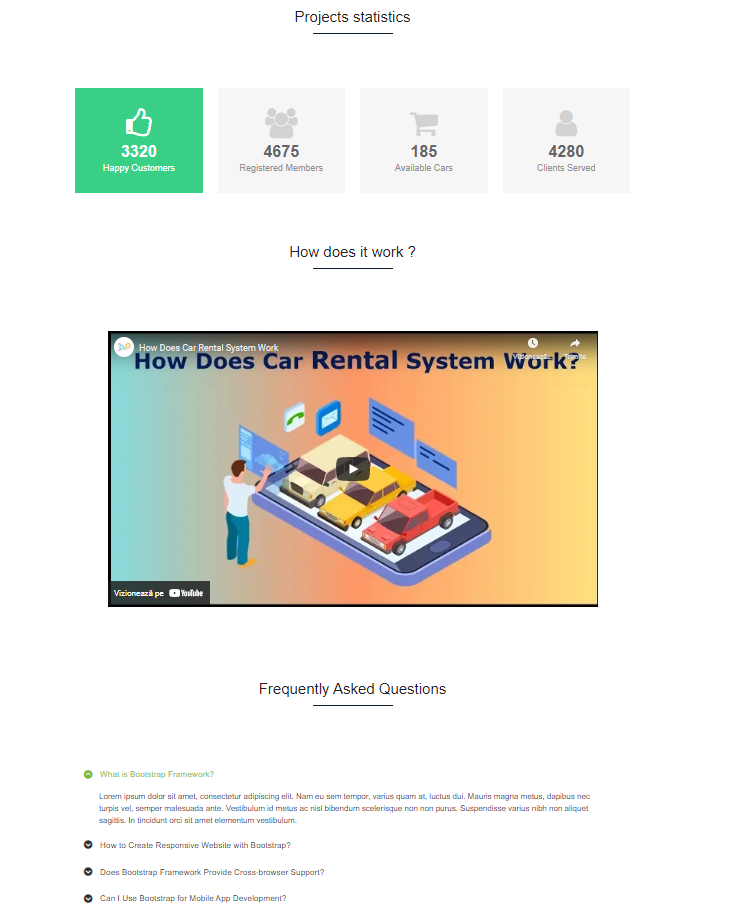
Customers List Page:



Car List Page:



Home Page:



## Hardware Interfaces

Not applicable.

## Software Interfaces

Information is read from the DB.

The software application is directly linked to the DB in order to get information about the users, cars and branches.

The web interface sends information directly to the DB, so it can make changes to it.

## Communications Interfaces

Having that it’s a web application, the communication between the client and server is made through HTTP/HTTPS. The web address is set by the IP address and the default port (80 for HTTP, 443 for HTTPS).

# System Features

## Create User Account

**4.1.1 Description and Priority**

A user should be able to access the website through either a browser or similar service. The website should be free to access. When a new/updated version of the website is released, the user should check for these manually, however the newsletter will notify of the changes. Given that a user has accessed the website, then he should be able to register through the login/register service.

**4.1.2 Stimulus/Response Sequences**

The Input (to create an account):

- Full Name

- Email

- Password

- Phone Number

- Gender

- Date of Birth

The output will be:

- Account has been registered successfully.

- Error messages: “Email already exists”, “Password too low”, “Phone number already exists”.

- We will also check the name. Users must be 18.

**4.1.3 Functional Requirements**

REQ-1: One textbox must be available in order to input the full name

REQ-2: One textbox must be available in order to input an email address

REQ-3: One textbox must be available in order to input a phone number

REQ-4: One drop-down menu must be available in order to select the gender

REQ-5: One form must be available in order to input date of birth

REQ-6: When the “Create account” button will be pressed, the information is read from the interface

REQ-7: It should be checked if the email address already exists in the DB

REQ-8: If it exists in the DB, an account cannot be created

REQ-9: If it doesn’t exist in the DB, then the email address will be stored in the DB and an account can be created

REQ-10: It should be checked if the phone number already exists in the DB

REQ-11: If it exists in the DB, an account cannot be created

REQ-12: If it doesn’t exist in the DB, then the phone number will be stored in the DB and an account can be created

## User Log in

* + 1. **Description and Priority**

Given that a user has registered, then the user should be able to log in to the web application. The log-in information will be stored on the browser and in the future the user can opt to be logged in automatically. Given that a user has registered, then the user should be able to retrieve his/her password by e-mail.

**4.1.2 Stimulus/Response Sequences**

The input (to log in):

- Email Address

- Password

The output will be:

- Access Permitted

- Access Rejected

**4.1.3 Functional Requirements**

REQ-13: One textbox must be available in order to input the email address

REQ-14: One textbox must be available in order to input the password

REQ-15: The email address will be read from DB to check if it coincides with the input introduced

REQ-16: The password will be read from the DB to check if it coincides with the one that has been introduced

REQ-17: If they match then the access will be set to true

REQ-18: If they are different then the access will be set to false

REQ-19: It should be checked if the textboxes are filled

REQ-20: If they are, then access will be granted

REQ-21: If they aren’t, then the access will be set to false (access rejected)

REQ-22: If the password or username were forgotten, the user can click on “Forgot Password?” button, then the user will be redirected to another page

REQ-23: Two textboxes must be available in order to input the new password (“new password” and “confirm password”)

REQ-24: One textbox must be available in order to input the email address

REQ-25: It should be checked if the email address already exists in the DB

REQ-26: If it exists in the DB, then the new password will be stored in the DB and a confirmation mail should be received

REQ-27: If it doesn’t exist in the DB, an exception is thrown.

## User Log out

* + 1. **Description and Priority**

A user must be able to sign out and log in at any time. Registration process should prompt the user accordingly, depending on the requirement.

Low priority.

**4.1.2 Stimulus/Response Sequences**

The user can press the “Log out” button and then he will be redirected to Homepage.

**4.1.3 Functional Requirements**

REQ-28: After pressing the “Log out” button the user will be redirected to the Homepage.

## Add cars by Admin

* + 1. **Description and Priority**

The admins will be able to add new cars.

High priority.

**4.1.2 Stimulus/Response Sequences**

The input for add a car:

* Name of the car
* Brand
* Year

The output will be:

* Added car
* Invalid car (if the car already exists in the database)

**4.1.3 Functional Requirements**

REQ-29: A textbox must be available in order to input the name of the car

REQ-30: A textbox must be available in order to input the brand

REQ-31: A textbox must be available in order to input the year

REQ-32: If one of the textboxes is empty, an error will be shown

REQ-33: If the car already exists in the DB, an error will be displayed

REQ-34: If it doesn’t exist in the DB, then the new car will be added to the DB

## Remove cars by Admin

* + 1. **Description and Priority**

The admins will be able to remove cars from the website.

Medium priority.

**4.1.2 Stimulus/Response Sequences**

If the admins find a car that does not exist in the stock anymore, they will be able to remove that car from the website by pressing the “Delete” button.

**4.1.3 Functional Requirements**

REQ-35: The admin should be able to press the “Delete” button

REQ-36: An image will pop up, asking for confirmation

REQ-37: If “Yes” is pressed, the car will be deleted from the DB and the website

REQ-38: If “No” is pressed, the DB and website will remain unchanged and nothing is going to happen

## Car availability check

* + 1. **Description and Priority**

The users will be able to check if a car is available.

Low priority.

**4.1.2 Stimulus/Response Sequences**

The user should be able to check if a car is available by pressing the “Rent” button. Then a message will be displayed: if the car is available - “In stock”, otherwise - “Out of stock”.

**4.1.3 Functional Requirements**

REQ-39: If the “Rent” button is pressed then the number of that specific car available in the DB should be compared to 0

REQ-40: If it is 0, the car is out of stock

REQ-41: If it is bigger than 0, then the car is in stock

## Car search

* + 1. **Description and Priority**

The users will be able to search for a car on the website.

Low priority.

**4.1.2 Stimulus/Response Sequences**

The user should be able to search a car by name in the Search bar. Once he hits enter, the search will begin.

**4.1.3 Functional Requirements**

REQ-42: The car is being searched by name which is introduced in the textbox

REQ-43: The enter button is pressed and the search begins

REQ-44: It’s checked to see if the car is available in the DB

REQ-45: If the car is available, it’s going to show up on the screen

REQ-46: If the car is not available, an error message will appear

## Car renting

* + 1. **Description and Priority**

The users will be able to rent a car on the website.

High priority.

**4.1.2 Stimulus/Response Sequences**

The user should click on the “Rent” button, if the car is available, then it can be rented.

**4.1.3 Functional Requirements**

REQ-47: If the “Rent” button is clicked and the car is in stock , the user will be able to complete some textboxes

REQ-48: Four textboxes will appear, asking for the user’s City, Street, Number, Zip Code, the date and hour on which the car shall be picked up

REQ-49: After the information are completed, the user will be scheduled at the given time and date

REQ-50: The DB will be modified accordingly when the car will be picked up

## Profile Manage

* + 1. **Description and Priority**

The users can access the section “My profile” where it will display information about them and they will be able to change some of the details.

Medium priority.

**4.1.2 Stimulus/Response Sequences**

After logging in, the users can change some details about them, like : their email address, password.

**4.1.3 Functional Requirements**

REQ-51: A textbox for each field will show up to input the information that the user wants to change

REQ-52: If some changes are made, the button “Save” must be pressed

REQ-53: After the changes, the new fields will be updated in the DB

# Other Nonfunctional Requirements



## Performance Requirements

The application should be able to load in less than a second.

Minimum system requirements in order to access the website through a browser (Google Chrome, Mozilla Firefox, Opera, Internet Explorer):

* + Intel Pentium 4 / AMD Athlon 64 processor or later that’s SSE2 capable
  + 250 MB of hard drive space (Mozilla Firefox needs 200 MB)
  + 512 MB of RAM (Mozilla Firefox needs 512 MB)
  + Android 5.0 Lollipop or higher or iOS 8.0 or higher for mobile devices

## Safety Requirements

Not applicable.

## Security Requirements

User registration is used in order to help better security and privacy. The login information is securely stored in the MySQL database. User passwords are encrypted. A “forgot password” system is available, and the user profile is also private and may not be viewed by others.

## Software Quality Attributes

Quality:

* The web application shall contain simple, clean and easy to use interfaces
* The web application shall contain well-placed styles and layouts that fit the chosen theme.
* The web application shall provide the user with all available options at all times to satisfy quality assurance.
* Add a blank line between if, else, for, do and while statements and their body code
* Use meaningful names for query variables
* A method should have a single purpose
* Braces are used with if, else, for, do and while statements even when the body contains

Performance:

* The web application shall contain smooth animations in drop-down menus, sliders and other related features.
* The web application shall contain fast execution times for all features.
* The web application shall manage user login support in a quick and correct manner.

## Business Rules

* The web application shall be easy to use by all employees including sales representatives and managers.
* The web application shall be available in several languages.
* The web application shall allow several sales to be made at the same time without downgrading performance.
* The web application representatives shall provide quick customer support for all users that have submitted a ticket.
* The web application FAQ and future updates shall be done based on the user feedback.

# Other Requirements

Not applicable.

Appendix A: Glossary

Not applicable.

Appendix B: Analysis Models

Not applicable.

Appendix C: To Be Determined List

Not applicable.