**International University of Sarajevo**

Computer science and engineering

Software engineering

Sarajevo, 20.3.2017.

**REQUIRMENTS DOCUMENT**

# KoomDriving

**Students**:   
Adem Dinarević

Ahmed Aletić

Aziz Boudela

**CONTENT**

* Preface
* Introduction
* Glossary
* User requirements definition
* System requirements specification
  + Functional requirements
  + Non-functional requirements
* System evolution
* References

**Preface**

This document is intended for system engineers, system test engineers, system maintenance engineers, managers and end users or system customers of the web application KoomDriving. This is the first volume of the document, with further changes being announced in later versions of this document.

**Introduction**

In Bosnia and Herzegovina there is a real need for a car school software. In our research before starting with this project, we were getting in touch with some of the major car schools in Sarajevo and found an interesting fact that they do not have a system which could make their life easier when it comes to filing and other documentation of their candidates. Because of that we decided to do this application which can be used in real life to help managers and instructors of those car schools, as well as future candidates (people who want to learn how to drive and to enroll to those car schools) to easier organize their job. That was our main goal when we choose to do this software.

It is going to be an ASP.NET application which is going to be implemented with C#. We found that to be the best platform due to its prevalence in todays IT as well as to make it easier for the users not to install certain programs which will work just on one platform (i.e. Windows or Mac). So it is the most universal platform that we could think of (i.e. a web application).

When the home page is opened the application will offer certain options for the users (tabs in the header of the page). It is going to have home, about us, instructors, register and login options. Like mentioned there is going to be a register and login option which is going to be connected to a database. So when a client wants to join he can easily register and open his client profile. Also instructors will have the possibility to login and to manage their instructor profiles. The clients will be able to see the schedule of the instructors so they can book a class with them, and instructors are going to have an option to see information about the candidates, their scores from driving exams and to add finished classes with them.

The application is going to include some constraints, like security requirements, so that it is not possible for other people use other peoples profiles. If there is time, some additional security features can be implemented, like e-mail confirmation etc. Confidentiality is also important as the passwords are going to be encrypted so the system administrator can not see the clients’ passwords.

Later in this document there are going to be detailed lists of functional and non-functional requirements as well as system evolution, references etc. The requirements are sorted according to priority list.

**Glossary**

|  |  |
| --- | --- |
| **1.3.2 Acronyms and Abbreviations Acronym** | **Meaning** |
| .NET | Microsoft framework with which Windows, Mac, web application etc. are made |
| ASP | Active Server Pages |
| ASP.NET | Web framework with .NET |

**User requirement definition**

Like previously mentioned the users of the system are instructors and candidates. They will have their profiles and they will be able to access and manipulate certain data.

Both of the users will be able to access the main page with the header which has tabs (Home, About us, Instructors, Register and Login). In those pages they will be able to see certain content which is going to be described with details in the functional requirements part.

The candidates will have the option for both login and register. If they did not register before they will have to register first before they can login to their system. When they register, there should be an e-mail verification of their profile, as of the non-functional requirements regarding security. When they register successfully they will be able to login with their e-mail. When they get to their profile, there are going to be certain options for them. One of them is the edit profile options in which they will be able to add certain information about themselves (name, surname, age, change password etc.). Another options is going to be to see the instructors schedule. If an instructors is available in a certain period of time, they can click reserve button so they can have a class with that instructor. They will be one field with the number of driving lesson they have to finish in order to go to the final driving session in front of the commission of the Ministry for traffic of the Canton (administrative region of Bosnia and Herzegovina). The instructors will be able to add each class when they finish to the students profile. The number of classes before getting to the commission is 35.

When it comes to the instructors they are not going to need to register, because their profiles will be already there ready, prepared from the system administrator. So they will just login with their username to their profile. They will have the option to see all the candidates and their info. They will also have an option to make their own schedule during the week and make it possible for the candidates to book his/her instructions. When they finish a class with one candidate they can add +1 to the counter of the candidates “finishes lessons” label.

When it comes to some other non-functional requirements regarding reliability, performance, response time etc. with the previously mentioned security requirements, that is going to be discussed in the non-functional requirement part of the document.

**System requirements speciation**

**Functional requirements**

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Definition** | **More details** |
| FR1.: Candidate register | Candidate can register by entering his mail and password. | A confirmation mail is sent afterwards. |
| FR2.: Candidate login | Candidate enters his e-mail and password after registering. | He is redirected to the candidate profile. |
| FR3. Instructor login | Instructor enters his username and password. | Instructors do not have to register to page. |
| FR4.: Home page | First page that opens when entering the site. | Contains greeting message with a header with other tabs. |
| FR5.: About us tab | Text information about the driving school. |  |
| FR6.: Instructors tab | List of all instructors in the car school. | Beside the instructors there is also an option “see more” for his education, licence and car. |
| FR7.: Register tab | Has a form with an input box for creating a user with e-mail and password. | An confirmation e-mail is sent after completion. |
| FR8.: Login tab | Has two textboxes, e-mail and password. | After logging in, a new page is opened for the user. |
| FR9.: Store data | Information about the clients and instructors | It is going to be stored in a database with all of their usernames, passwords etc. Certain information can be added afterwards (e.g. the candidate can add some information about himself) which is also going to be stored in a different database |
| FR10.: Retrieving data | Instructors will be able to retrieve data from their candidates | Some of the information include the number of finished classes as well as their scores on previous driving exams. |
| FR11.: E-mail confirmation | Users will receive an e-mail that they successfully registered to the site | It is going to include a greeting message, his username and password. |

**Non-functional requirements**

|  |  |  |
| --- | --- | --- |
| **Requirement No.** | **Definition** | **More details** |
| NFR1. | Product requirements | Program will have all three main requirements, efficiency, reliability, portability. |
| NFR2. | Efficiency | App. will be very efficient and will save time and less workers will be needed to manage system. |
| NFR3. | Reliability | All the actions that user wants at wanted time will create desired output. |
| NFR4. | Portability | Since app is .NET it is very easily accessible and it is very easily installed to any computer. |
| NFR5. | External requirements | Are going to include interoperability and legislative requirements at most, but others too. |
| NFR6. | Interoperability | Application will be able to use and exchange information’s in system |
| NFR7. | Legislative | Will provide all necessary safety and security. |
| NFR8. | Privacy | Privacy will guaranty that candidates and other staff will have separate accessibility to system. |
| NFR9. | Safety | Safety is one of the most important parts of every app, all information of this app will meet high standards of security and safety. |
| NFR10. | Implementation | Implementation of the application will have high performance and will be very user friendly. |
| NFR11. | Organisational requirements | Organization is the main feature of app. so it will meet all organizational requirements. |
| NFR12. | Standard | The standard will guaranty the user that application will be well build, useable, safe. |
| NFR13. | Implementation | Implementation of the application will have high performance and will be very user friendly. |

**System evolution**

If the conditions are ideal and we do not face certain problems while developing the software we will be able to finish all features of the software on desired time. But we predict that they are going to be certain changes to the project, mainly due to the time constraints that we have. Our deadline is till end of May or beginning of June which is a limited number of time to implement all the features that are mentioned in the document. We predict that the car school owners will also change their mind with some of the features, adding or removing certain options that are mentioned in this document. When it comes to the hardware, we think it is not going to represent an issue, due to the fact that we are using cutting edge web application development techniques which are following the latest trends in the industry, so any changes in e.g. browser compatibility are not going to be a problem.

**References**

* I Sommerville, Software Engineering*, 9*th ed, Addison-Wesley, 2007.
* *Software Requirements Engineering.* This collection of papers on requirements engineering includes several relevant articles such as 'Recommended Practice for Software Requirements Specification', a discussion of the IEEE standard for requirements documents. (R. H. Thayer and M. Dorfman (eds.), 1997, IEEE Computer Society Press.)    

**Extras**

When it comes to the organization of work in the project, we decided that Adem Dinarević is going to the backend of the program, creating the database and connecting it to the page. Ahmed and Aziz are responsible for the front end and dealing with the design and front-end part of the page, object oriented design and interface.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Team member** | **Task** | **Deadline** | **Completed** | **Signature** |
| Adem Dinarević | Introduction, System evolution, user requirements definition, functional requirements |  |  |  |
| Aziz Boudela | Non-functional requirements |  |  |  |
| Ahmed Aletić | Functional requirements |  |  |  |

**Organization of Requirements Document**