Software Requirements Specification

for

Oline Shop Application

**Version 1.0 approved**

**Prepared by:**

**Saraçi Sotiraq**

**Faculty of Automatics, Computer Science and Electronics**

**11.11.2022**

**Table of Contents**

[1. Introduction 1](#_Toc98407442)

[1.1 Purpose 1](#_Toc98407443)

[1.2 Document Conventions 1](#_Toc98407444)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc98407445)

[1.4 Product Scope 2](#_Toc98407446)

[1.5 References 2](#_Toc98407447)

[2. Overall Description 2](#_Toc98407448)

[2.1 Product Perspective 2](#_Toc98407449)

[2.2 Product Functions 3](#_Toc98407450)

[2.3 User Classes and Characteristics 3](#_Toc98407451)

[2.4 Operating Environment 3](#_Toc98407452)

[2.5 Design and Implementation Constraints 3](#_Toc98407453)

[2.6 User Documentation 4](#_Toc98407454)

[2.7 Assumptions and Dependencies 4](#_Toc98407455)

[3. External Interface Requirements 4](#_Toc98407456)

[3.1 User Interfaces 4](#_Toc98407457)

[3.2 Hardware Interfaces 4](#_Toc98407458)

[3.3 Software Interfaces 4](#_Toc98407459)

[3.4 Communications Interfaces 5](#_Toc98407460)

[4. System Features 5](#_Toc98407461)

[4.1 Registration 5](#_Toc98407462)

[4.2 Login 5](#_Toc98407463)

[4.3 Multiple user/admin login 6](#_Toc98407464)

[4.4 Users 6](#_Toc98407465)

[4.5 Administrators 7](#_Toc98407466)

[4.6 Event booking 7](#_Toc98407467)

[4.7 Event booking (administrators) 8](#_Toc98407468)

[4.8 Event booking (non-members) 8](#_Toc98407469)

[4.9 Event canceling 9](#_Toc98407470)

[4.10 Event canceling (non-members) 9](#_Toc98407471)

[4.11 Event canceling (administrators) 10](#_Toc98407472)

[4.12 Event canceling (Admins to members & non-members) 10](#_Toc98407473)

[4.13 Resource expense report 11](#_Toc98407474)

[4.14 Resource earnings report 11](#_Toc98407475)

[4.15 Blocking events by reports 11](#_Toc98407476)

[5. Other Nonfunctional Requirements 12](#_Toc98407477)

[5.1 Performance Requirements 12](#_Toc98407478)

[5.2 Safety Requirements 12](#_Toc98407479)

[5.3 Security Requirements 12](#_Toc98407480)

[5.4 Software Quality Attributes 12](#_Toc98407481)

[5.5 Business Rules 12](#_Toc98407482)

[6. Other Requirements 13](#_Toc98407483)

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Online Shop | 11.11.2022 | Work started | 1.0 |
|  |  |  |  |

# Introduction

The introduction of this Software Requirements Specification (SRS) provides an overview of the entire SRS with purpose, scope, definitions, acronyms, abbreviations, references, and overview of the SRS. The purpose of this document is to give an insight of the complete **Online Shop Web Application** by defining the application statements in detail. The detailed requirements of the **Online Shop** are provided in this document.

## Purpose

The scope of this web application is to develop software system to manage the activity of a club for art enthusiasts.

## Document Conventions

|  |  |
| --- | --- |
| **Definitions, Acronyms, and Abbreviations** | **Explication** |
| OSWA | Online Shop Web Application |
| OS | Online Shop |
| FAQ | Frequently asked questions |
| FTP | File Transfer Protocol |

Also, when you read this manual, certain words are represented in different fonts, typefaces, sizes, and weights. This highlighting is systematic; different words are represented in the same style to indicate their inclusion in a specific category. The types of words that are represented this way include the following:

|  |  |
| --- | --- |
| Bold | Titles |
| Baskerville Old Face | SRS Font |
| 12 | Font size |

## Intended Audience and Reading Suggestions

This software requirement specification (SRS) document is written for general audience, this document is intended for individuals directly involved in the development of web application technologies. This includes software developers, project consultants, team managers and users. This document may not be read sequentially; users are encouraged to jump to any section they find relevant. Below is a brief overview of each part of the document.

1. Introduction
2. Overall Description
3. External Interface Requirements
4. System Features
5. Other Nonfunctional Requirements

## Product Scope

The scope of OSWA is an online e-commerce platform where users can browse and buy products. In this case products are divided in two main categories which are: clothes and shoes. The system includes a user interface (web navigator or application), one server of databases which will contain information about club activity and an application server.

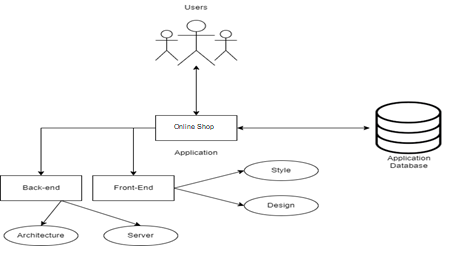
## References

* [Visual Studio Code](https://code.visualstudio.com/docs)
* [Microsoft Docs - Visual Studio Community 2019](https://docs.microsoft.com/en-us/visualstudio/windows/?view=vs-2022)
* [Microsoft Docs - ASP.Net](https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-6.0)
* [Microsoft Docs - SQL Server](https://docs.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver15)
* [Angular](https://angular.io/)
* [W3Schools](https://www.w3schools.com/)
* [YouTube - SuperSimpleDev Tutorials](https://www.youtube.com/channel/UCB6dvaWu0N8uVq2yKsZ5s5g)
* [C Coding Style](http://caxapa.ru/thumbs/468328/misra-c-2004.pdf)

# Overall Description

## Product Perspective

This document contains the statement of OSWA, an WA which provides registration and login to its users. In real time users can register, login, and effectuate transactions within the application. This WA is designed with ASP.Net framework (back-end), Angular (front-end) and a database which will hold all the data that is necessary for the OS. These components are interconnected as below:



## Product Functions

Members will be able to register online with a login name and password. This information is verified with the information in the database and the member is appended.

Members will be able to add, modify or delete the information provided by them when they register. The members will be able to browse through different item’s categories, items, add them to cart, buy cart items, introduce their billing information, get receipt about their transactions, and leave their reviews about the experience on the site.

## User Classes and Characteristics

* Administrator
* Will be able to supervise member’s activities.
* Will be able to add, edit, or modify items, categories, etc.
* All user characteristics.
* User
* Will be able to register and login.
* Will be able to browse through items and categories.
* Will be able to select items and adding them to cart.
* Will be able to buy the cart.
* Will be able to introduce billing details (ex: credit card details).
* Will be able to get a receipt for his/her transactions.
* Will be able to leave a reviews about his experience on site.

## Operating Environment

* Hardware platform: PC, Laptop & Mobile Phone.
* Operating systems: Windows (XP 2005-10-14 or higher), Linux, Mac, Android 4.0 or higher & iOS 5.0 or higher.
* Software platforms: Web Browser (Google Version 4-60 and 71 or higher, Mozilla Version 3.6 or higher, Opera Version 15 or higher, etc.).
* FTP.
* Cloud database.

## Design and Implementation Constraints

* Regulatory policies: none if not indicated.
* Hardware limitations: deadlines for milestones as following

|  |  |
| --- | --- |
| **Milestone** | **Deadline** |
| First presentation | 12.11.2022 |
| Second Presentation | 10.12.2022 |
| Final Presentation | 21.01.2023 |

* Technologies used: ASP.net Framework, Angular, JavaScript, databases.
* Language requirements: C#, JavaScript, SQL
* Programming standards: OOP
* Design conventions: see references C-Coding Style.
* Microsoft Visual Studio Community 2019
* Microsoft SQL Server Management Studio 2019
* Visual Studio Code

## User Documentation

All the details for this application are presented here.

## Assumptions and Dependencies

Not applied

# External Interface Requirements

## User Interfaces

OSAW will start with a homepage where it is displayed the functionality of the whole system and it will have different options. 2 of the most important options on the homepage will be sign-up and login. After a user has been registered in the webpage, he will be registered automatically in the system database. An admin can also confirm a registration. After logging in, the user will have a set of options which are described here. An admin will be added manually from the database and only he, has the access to add new admins.

Text

Description automatically generatedText

Description automatically generated with medium confidence

Diagram

Description automatically generated

## Hardware Interfaces

There are not any hardware interfaces.

## Software Interfaces

The server is connected permanently with the database. There will be implemented functions that will modify the database. These functions will be displayed as options for the user.

Functions prototypes are:

• private void Insert (object sender, EventArgs e)

• private void Update(object sender, EventArgs e)

• private void Delete(object sender, EventArgs e)

• private void DisplayData().

## Communications Interfaces

* FTP
* E-mail
* Cloud

# System Features

## Registration

**4.1.1 Description and Priority**

Allows new users to create an account on the webpage.

Priority: high

**4.1.2** **Stimulus/Response Sequences**

Input: The user enters his credentials (email/username/password/confirm password) and submits them to the system.

Output: System sends a confirmation that new account is created in the system.

**4.1.3** **Functional Requirements**

* User completes the fields of registration.
* User submits the information presented.
* If credentials are not written within the rules presented, the user will be requested to modify his data.
* User data is stored in the respective database.

## Login

**4.2.1 Description and Priority**

Allows users access their account on the webpage.

Priority: high

**4.2.2** **Stimulus/Response Sequences**

Input: The administrator enters his credentials (username/email/password) and submits them to the system.

Output: After logging, the application will send a response if the administrator has logged in successfully.

**4.2.3** **Functional Requirements**

* User completes the fields of registration.
* User submits the information presented.
* If credentials are not correct, he will be requested to add his data again.
* If the user doesn’t remember the credentials, he will have to recover them.
* Recovering credentials will be done by contacting the administrator.
* If the credentials presented are verified by the database, the user has access to his account.

## Multiple user/admin login

**4.1.1 Description and Priority**

Allows many more users to connect at the webpage at the same time.

Priority: high

**4.1.2** **Stimulus/Response Sequences**

Input: Multiple users can log in at the same time.

Output: System returns number of users connected in the moment.

**4.1.3** **Functional Requirements**

* Many users/admins can connect at the same time
* Count the number of active users at a time in a textbox at the bottom right of the screen.

## Users

**4.1.1 Description and Priority**

User account is the basic account which can interact in the webpage. His roles are described as below.

Priority: high

**4.1.2 Stimulus/Response Sequences**

Inputs: The list of options that user has (creating events, modifying them, inviting members, etc.)

Outputs: The system returns executed options.

**4.1.3 Functional Requirements**

* Can register.
* Can login.
* Can browse the website.
* Can effectuate transactions within the website.

## Administrators

**4.1.1 Description and Priority**

Administrator account can be added by the owner. His role is to supervise all user activities and ensure the correctness of operations on the webpage.

Priority: high

**4.1.2 Stimulus/Response Sequences**

Inputs: The list of options that user has (creating events, modifying them, inviting members, admin panel, etc.)

Outputs: The system returns executed options.

**4.1.3 Functional Requirements**

* All user functionalities.
* Can manage payments.
* Can be contacted from the users if they encounter problems or need help.
* Can generate or edit products.

## Product

**4.1.1 Description and Priority**

Products are divided in categories which makes easier finding them and adapting to user requests

Priority: high

**4.1.2 Stimulus/Response Sequences**

Input: Admin creates product / User selects product

Output: Product created / Product selected

**4.1.3 Functional Requirements**

* Products will have details (name, category, price, etc)
* Product can be added to cart.
* Products can be added to favorites

## Cart

**4.1.1 Description and Priority**

Products that an user would like to buy are added at a cart which then proceeds to the payment.

Priority: high

**4.1.2 Stimulus/Response Sequences**

Input: User selects product

Output: Product added to cart

**4.1.3 Functional Requirements**

* Products will have details (name, category, price, etc)
* It should display quantity.
* It should display price.

# Other Nonfunctional Requirements

## Performance Requirements

|  |  |
| --- | --- |
| **Requirement** | **Specification** |
| Database | Database size must be greater than 250MB |
| Server | Server size must be greater than 250MB |
| Internet connection | Ping less than 30 |
| Windows | Version XP or higher |

## Safety Requirements

Not applied.

## Security Requirements

* Users log in using a unique account and password. Only persons authorized by  
  the owners association can log in to the application.
* The password will be encrypted in the database.

## Software Quality Attributes

1. Each instruction is written on a line.
2. Each comment will be placed on a separate line.
3. Each variable must have a suggestive name.
4. Each variable declaration contains only one variable. A statement that int a, b; is not used.
5. All statements must have open and closed accolades.
6. Each method must be written one per line.
7. All variables will be declared before use.
8. Variables will have the appropriate date type so that there are no compilation errors.
9. Variable names longer than 40 characters will not be used.

10. Only one space (line) is allowed between instructions to write an orderly, organized code  
in logical sections.

## Business Rules

The system must have at least one administrator account to access and interact with the application.

# Other Requirements

**Appendix A: Glossary  
Appendix B: Analysis models  
Appendix C: List to be determined**