Software Requirements Specification

SOFTUNI BLOG

Team FAFNIR :

**Gergana Guncheva**

**Gergana Bozova**

**Madlena Ivanova**

**Version: 4.1 Date:29.03.2019**

**Table of Contents**

[1.Introduction 3](#_Toc4502019)

[1.1.Purpose 3](#_Toc4502020)

[1.2.Scope 3](#_Toc4502021)

[1.3 Definitions, acronyms, and abbreviations 3](#_Toc4502022)

[1.4.Reference 4](#_Toc4502023)

[2. Overview 4](#_Toc4502024)

[3.User characteristics 6](#_Toc4502025)

[4.Functional requirements 7](#_Toc4502026)

[4.1. Use Cases 7](#_Toc4502027)

[4.1.1 Functional requirement 1.1 7](#_Toc4502028)

[4.1.2 Functional requirement 1.2 7](#_Toc4502029)

[4.1.3 Functional requirement 1.3 8](#_Toc4502030)

[4.1.4 Functional requirement 1.4 8](#_Toc4502031)

[4.1.5 Functional requirement 1.5 9](#_Toc4502032)

[4.1.6 Functional requirement 1.6 9](#_Toc4502033)

[4.1.7 Functional requirement 1.7 10](#_Toc4502034)

[4.1.8 Functional requirement 1.8 10](#_Toc4502035)

[4.1.9 Functional requirement 1.9 11](#_Toc4502036)

[4.2 Non-functional requirements 11](#_Toc4502037)

[4.2.1 Usability Requirements 11](#_Toc4502038)

[4.2.3    Compatibility Requirements 12](#_Toc4502039)

# 1.Introduction

Herein, there`s a short overview of the SRS document for the blog system.

## 1.1.Purpose

The purpose of this document is to give a detailed description of the requirements for the “SoftUni Blog” system.It will explain the purpose and features of the system, the interfaces of the system and what the system will do. This document is intended for both the Developers of the system and the Quality Assurance team.

## 1.2.Scope

The system is a basic online blog, which provides functionality, where users can post their own thoughts, their vision on specific problems or things that are to their interest. The system also allows users to edit or delete their posts.

Users can have different access to the system, access as an ordinary user or as an administrator. Registered users can make posts, edit or delete their posts and administrators can monitor for the correctness of the posts on the blog.

The system works on different browsers and needs Internet connection.

## 1.3 Definitions, acronyms, and abbreviations

*Table Definitions:*

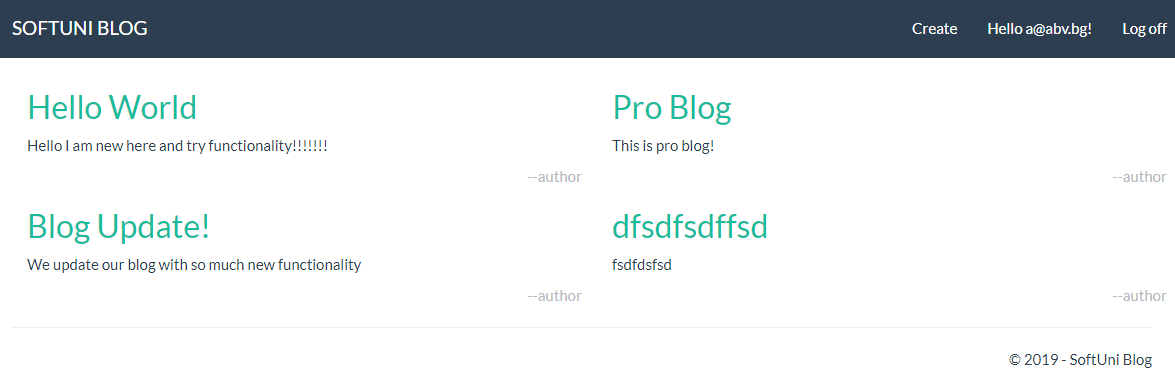
|  |  |
| --- | --- |
| **Term** | **Definition** |
| Blog | A regularly updated web-based system, which displays the posts of individuals or small group with same interests. |
| Admin/Administrator | A person who is given specific permission for managing and controlling the system |
| User | A person who uses functionalities, provided by the system and has some limited permission. |
| Stakeholder | Any person who has interaction with the system who is not a developer. |
| DESC | Description |
| RAT | Rational |
| DEP | Dependency |
| TAG | A unique, persistent identifier contained in a PLanguage statement [2] |
| GIST | A short, simple description of the concept contained in a PLanguage statement [2] |
| SCALE | The scale of measure used by the requirement contained in a PLanguage statement [2] |
| METER | The process or device used to establish location on a SCALE contained in a PLanguage statement [2] |
| MUST | The minimum level required to avoid failure contained in a PLanguage statement [2] |
| PLAN | The level at which good success can be claimed contained in a PLanguage statement [2] |
| WISH | A desirable level of achievement that may not be attainable through available means contained in a PLanguage statement [2] |
| DEFINED | The official definition of a term contained in a PLanguage statement [2] |

## 1.4.Reference

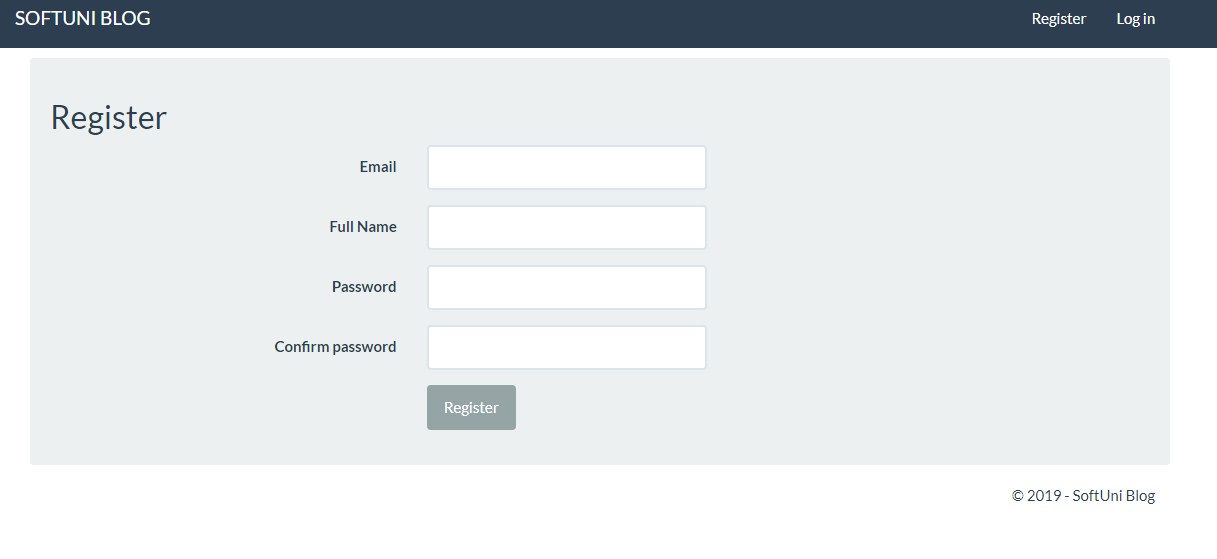
<http://www.softwaretestingstandard.org/> International standards for software testing

# 2. Overview

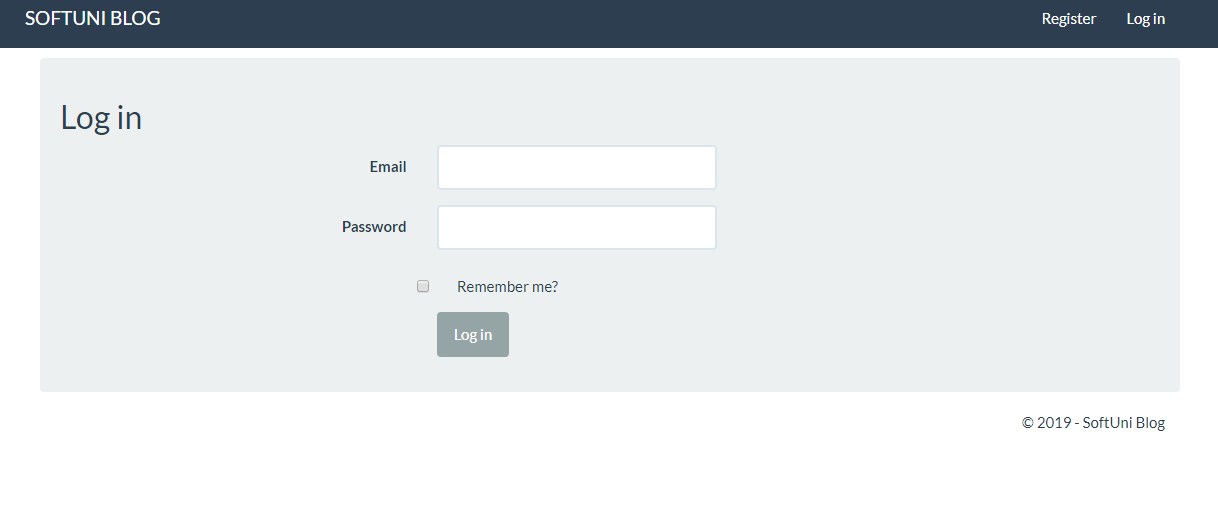
When a user loads the Internet address of the system, he/she should see the home page of the web site:



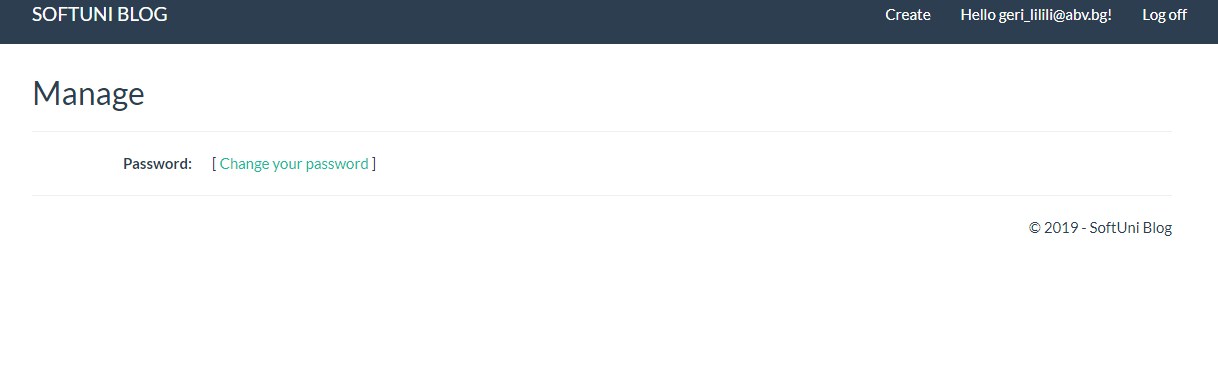
If the user is not registered, he/she must register his/her own profile in the system from the Register menu:



If the user isalready registered, he/she must enter his/her username and password form the Log in menu:



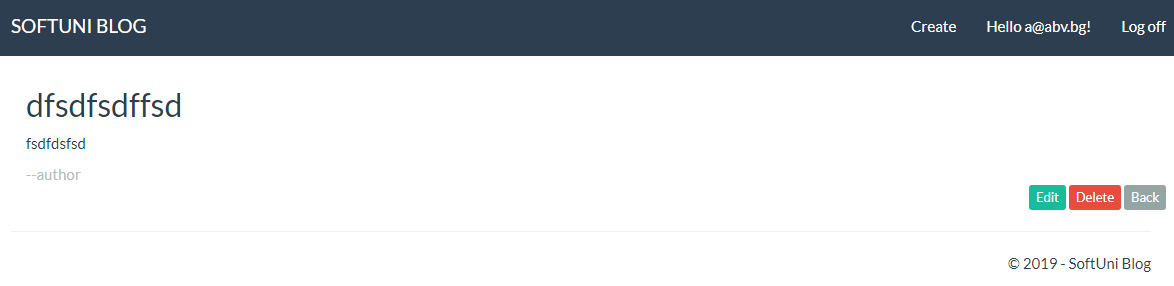
Registered users can manage their profile from special menu, with a functionality for changing password:

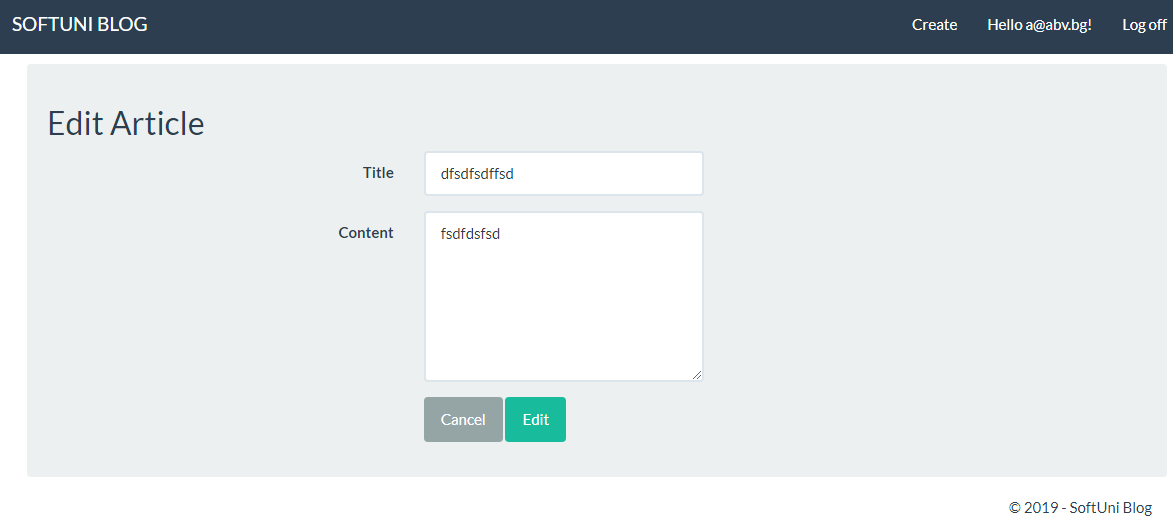


Registered and logged users can create their own articles and publish them to the main page of the blog. They must provide Title and Content for their posts.

### Screenshot_4

Users can edit or delete their posts by clicking on them from the main page:





# 3.User characteristics

There are two types of users that interact with the system: users of the blog and administrators. Each of these two types of users has different use of the system so each of them has their own requirements.

Administrators only interact through the blog portal. They manage the overall system so there is no incorrect information in it.

# 4.Functional requirements

## 4.1. Use Cases

#### 4.1.1 Functional requirement 1.1

**ID: FR1**

TITLE: Home Page Unregistered User

DESC:

GIVEN that a user loads the Blog address,

THEN the user should be able to view the Home Page.

AND Home Page displays Articles with Titles, part of the Body text and Author.

AND The Articles are ordered by Date, latest articles on top of the page.

AND The Home Page should have header with Link Text “Softuni Blog” on the left side, which should lead to the Home Page and “Register” and “Log in” buttons on the right side.

AND The Home Page should have footer with text for the Blog Producer.

RAT: In order unregistered user to see the Home Page

DEP: None

#### 4.1.2 Functional requirement 1.2

**ID: FR2**

TITLE: User Registration

DESC:

GIVEN that an unregistered user loads the Blog address,

WHEN The user clicks on the “Log in” button on the header.

THEN Registration page should load

AND the user must provide e-mail , full name , password and has to confirm the created password in different fields.

AND All fields should be mandatory.

THEN the user should be able to register.

RAT: In order for a user to register on the blog.

DEP: None

#### 4.1.3 Functional requirement 1.3

**ID: FR3**

TITLE: User Log-In

DESC:

GIVEN that a registered user loads the Blog Address

WHEN the user clicks on the ”Log in” button on the header.

THEN Logging in page should load.

AND The log in page should contain “Email” and “Password” fields, “Log in” button and “Remember me” check box.

WHEN user checks the “Remember me” button,

AND the user visits the site again,

THEN the log in credentials should be automatically filled in.

RAT: In order for a user to log in.

DEP: The user is registered

#### 4.1.4 Functional requirement 1.4

**ID: FR4**

TITLE: Home Page Registered User

DESC:

GIVEN that a registered user loads the Blog Address

WHN logging in with valid credentials

THEN the user should be able to view Home Page.

AND Home Page should display Articles with Titles, part of the Body text and Author.

AND The Articles are ordered by Date, latest articles on top of the page.

AND The Home Page should have header with Link Text “Softuni Blog” on the left side,

AND the Link should lead to the Home Page.

AND On the right side of the header there should be “Create” button, which leads to a “Create article” page,

AND “Greetings” link with text “Hello [user email]!”, which leads to “Manage” page, where user can change his/her password,

AND “Log out” button, which should load “Home Page” for unregistered users.

AND The Home Page should have footer with text for the Blog Producer.

RAT: In order for a registered user to see the Home Page

DEP: The user is registered

#### 4.1.5 Functional requirement 1.5

**ID: FR5**

TITLE: Change Password

DESC:

GIVEN that a registered user loads the Blog Address

WHEN logging in with valid credentials

AND The user clicks on the “Greetings” link on the header

THEN the “Manage” page should be loaded.

WHEN The user clicks on “Change your password” button

THEN “Change Password” page should load.

AND On the “Change Password” page there should be 3 mandatory fields:

* “Old Password”
* “New Password”
* “Confirm new Password”

AND “Change password” button.

WHEN the user fills all fields with valid credentials

AND clicks on “Change password” button,

THEN the user should be redirect back to the “Manage” page

AND appropriate message that the password is successfully changed should appear.

RAT: In order registered user to see change his password.

DEP: The user is registered

#### 4.1.6 Functional requirement 1.6

**ID: FR6**

TITLE: Create Article

DESC:

GIVEN that the user is registered and logged in to the blog.

WHEN the user clicks on the “Create” button

THEN the “Create” page should load.

AND “Create” page should contain “Author” and “Content” fields and “Cancel” and “Create” buttons.

AND Both fields should be mandatory.

AND “Cancel” button should lead to the “Home Page” without saving and posting any new article.

AND “Create” button should lead to the “Home Page’, where the new article should be posted.

RAT: In order for a registered user to create article.

DEP: The user is registered

#### 4.1.7 Functional requirement 1.7

**ID: FR7**

TITLE: Article Page

DESC:

GIVEN that the user loads the Home Page,

THEN he/she is able to view all articles.

WHEN the user clicks on each article in order to read the full text.

THEN the “Article” page should contain the Title of the article,

AND the body of the article

AND Author of the article on the bottom left corner.

AND On the bottom right corner of the article there should be three buttons:

* “Edit”- If the user is registered and logged in, he/she should be able to edit his/her own articles. If the user is registered and logged in and tries to edit the article of another author, a proper message that he/she doesn’t have permission to edit articles of other authors should be displayed. If the user is not registered and clicks on “Edit” button, the log in page should be loaded.
* “Delete” - If the user is registered and logged in, he/she should be able to delete his/her own articles. If the user is registered and logged in and tries to delete the article of another author, a proper message that he/she doesn’t have permission to delete articles of other authors should be displayed. If the user is not registered and clicks on “Delete” button, the log in page should be loaded.
* “Back” button, which should lead to the “Home Page”

RAT: In order for a user to read, edit and delete articles.

DEP:

#### 4.1.8 Functional requirement 1.8

**ID: FR8**

TITLE: Edit Article Page

DESC:

GIVEN that the user is registered and logged in to the blog.

THEN the user should be able to edit his own articles.

WHEN The user opens “Article” Page on his/her own article,

AND the user clicks “Edit” button on the bottom of the article.

THEN “Edit Article” page should be loaded with fields “Author” and “Content”, where the user can make changes.

AND Under the fields, there should be two buttons:

* „Cancel”, which loads the “Home Page” without any changes,
* “Edit”, which loads the “Home Page”, where the changes are visible.

RAT: In order for a user to edit articles.

DEP: The user is registered, logged in and has his/her own articles.

#### 4.1.9 Functional requirement 1.9

**ID: FR9**

TITLE: Delete Article Page

DESC:

GIVEN that the user is registered and logged in to the blog.

THEN the user should be able to delete his/her own articles.

WHEN The user opens “Article” Page on his/her own article,

AND the user clicks “Delete” button at the bottom of the article.

THEN “Delete Article” page should be loaded with fields “Author” and “Content”, which are not active for making changes.

AND Under the fields, there should be two buttons:

* „Cancel”, which loads the “Home Page” without any changes,
* “Delete”, which loads the “Home Page”, where the article is not visible anymore.

RAT: In order for a user to delete articles.

DEP: The user is registered, logged in and has his/her own articles.

## 4.2 Non-functional requirements

4.2.1 Usability Requirements   
  
•    Colors that we use on this web system design must be attractive.  
•    Easy navigation is preferable to do any task.  
•    Multiple flows (ways) are preferable to do any task.  
•    Home page should be centralized system (Screen/Window) to navigate to any feature and to get any result.  
•    The facility to return to Home page from any page should be available.  
  
4.2.2    Performance Requirements   
  
•    It has to load, within the industry standard time.  
•    It has to support up to 1 000 concurrent users.  
•    Response time should be 4 seconds

### 4.2.3    Compatibility Requirements

•    It should support all types of hardware versions, operating systems and browsers  
  
4.2.4    Operating Systems   
  
•    Operating systems -Wndows98 and upper Versions (EX: Win98, windows 2000 prof, XP Vista and win NT Server,windows 2000 server, 2003 server and windows 2008 server)  
   
4.2.5    Browsers   
  
•    IE 6.0 and higher versions , Latest versions of Mozilla Firefox, Opera and Google Chrome  
  
4.2.6    Security Requirements  
  
•    Data should be encrypted during Login and while performing publishing.  
•    Server on which database resides must have its own security to prevent unauthorized access. It should be accessed only by the privileged users  
•    The server on which application resides must have its own security. Authorization and Authentication checks should be performed.