**Software Requirements Specification**

**for**

**User Management and Authentication (Module 1),**

**Release 1.0**

**Version 1.0 approved**

Prepared by:

Mu’az Toshihiko Bin Mansor (179117)

Nor Azmira Binti Doman (176553)

Wan Norizzati Binti Wan Hasan (178898)

Syafira Binti Mohd Sofian (179771)

Nurul Afiqah Binti Azman (178688)

**University of Putra Malaysia**

**Disember 22 , 2016**

**Table Of Content**

[**1.0 Introduction**](#_8c50mcvhy5tj)[**2**](#_8c50mcvhy5tj)

[1.1 Purposes](#_dp035m3yrbp0) [2](#_dp035m3yrbp0)

[1.2 Objective](#_wt95277l6zqh) [2](#_wt95277l6zqh)

[1.4 Design and Implementation Constraints](#_xacyuupt5zoc) [3](#_xacyuupt5zoc)

[**2.0 Overall Description**](#_tpwvhh26kf6r)[**3**](#_tpwvhh26kf6r)

[2.1 Application Perspective](#_vj9x0nt41543) [3](#_vj9x0nt41543)

[2.2 User Classes and Characteristics](#_8h7xajuk6au2) [4](#_8h7xajuk6au2)

[2.3 Operating Environment](#_q9lfersbdf83) [4](#_q9lfersbdf83)

[2.4 Assumptions and Dependencies](#_bo75gg4jc7ze) [4](#_bo75gg4jc7ze)

[**3.0 Activity Diagram**](#_6y7albma4gmx)[**5**](#_6y7albma4gmx)

[**4.0 System Use Case**](#_dyorf7iev7y1)[**6**](#_dyorf7iev7y1)

[4.1 Register Account (UC1) Flow of Event](#_ylwreo5cuwh9) [7](#_ylwreo5cuwh9)

[4.2 Login (UC2) Flow of Event](#_weoea3atu38l) [7](#_weoea3atu38l)

[4.3 Update Account (U3) Flow of Event](#_xv2xa4yrm12p) [8](#_xv2xa4yrm12p)

[4.4 Request New Role (U4) Flow of Event](#_2k91ijhfpy5d) [9](#_2k91ijhfpy5d)

[**5.0 User Interface**](#_wqnyzklb6ni9)[**10**](#_wqnyzklb6ni9)

[**6.0 Non Functional Requirement**](#_xff97y8nv8q2)[**11**](#_xff97y8nv8q2)

[6.1 Performance Requirements](#_wal6gwhwgpew) [11](#_wal6gwhwgpew)

[6.2 Security Requirements](#_fr5i0r6tb97q) [11](#_fr5i0r6tb97q)

[6.3 Availability Requirements](#_ivqt91bh66xr) [11](#_ivqt91bh66xr)

[6.4 Scalability Requirements](#_ikm3c5xpjgf3) [11](#_ikm3c5xpjgf3)

[6.5 Usability Requirements](#_goji20e4pqr7) [12](#_goji20e4pqr7)

[**7.0 Architecture Design**](#_7v5zuefyvlc4)[**12**](#_7v5zuefyvlc4)

[**8.0 Entity-Relationship Diagram (ERD)**](#_anr43d6oqsn5)[**14**](#_anr43d6oqsn5)

[**9.0 Integration System**](#_20ilk8lsrit0)[**15**](#_20ilk8lsrit0)

# **1.0 Introduction**

## **1.1 Purposes**

The purpose of this document is to give a detailed description of the requirements for the Accounting system for module 1. First of all, this module 1 is about the user management and authentication. So, this document will illustrate the purpose and complete declaration for the development for this system.This document will illustrates the functional and nonfunctional requirement of Accounting system.

This Accounting System is a medium for company directors and company secretary, accountants, auditors, accountants delegate and auditors delegate to interact. We serve the purpose of simplifying task and also offer productive measure for all users.

## **1.2 Objective**

The objective of this module is to allow user to authenticate into our system. The main functionality of this module is to manage the authorization level of users. This module will determine the level of accessibility of each type of user. User with certain level of authority will only be able to access information or data according to their respective level. User can request a new role by fill out a form to request the role and submit to the admin. Admin will receive the requested form to verify either the user is eligible or not for the role.

**1.3 Scope**

This project will cover web application only. Mobile application is a future improvement. The scope for this system is wide because any qualified accountant and auditor can register into this application.So, company director and company secretary can assign any accountant and auditor to manage their company’s account.

## **1.4 Design and Implementation Constraints**

User must keep their password as confidential. The Internet connection is a constraint for the application. Since the application fetches data from the database over the Internet, it is crucial that there is an Internet connection for the application to function.

# **2.0 Overall Description**

## **2.1 Application Perspective**

The accounting system is a web application that can be used by company (company directors and company secretary), accountants, auditors, accountants delegate and auditors delegate. This system is very flexible so that the user from any related background can register into this system. Basically for module 1, the system allowed user to register and login into this system. User will be categorized by their status which is company director or company secretary or etc. The user will only access certain function in this system by following the status, so they will have different menu in the dashboard. The user able to request more than one roles.

## **2.2 User Classes and Characteristics**

There are eight types of users that interact with the system which are admin, new user, company directors and company secretary, accountants, auditors, accountants delegate and auditors delegate. Each of these six types of users has different use of the system so each of them has their own requirements.

## **2.3 Operating Environment**

This system will operate on almost all modern browser.

## **2.4 Assumptions and Dependencies**

* The Accounting system is available 24 hours a day, 7 days a week.
* The devices should have strong Internet connection and Internet server capabilities.

# **3.0 Activity Diagram**

|  |
| --- |
|  |

Activity diagram above shows the flow for module 1 in this accounting system. Firstly, before user login into the system, the user need to register first.After login, user can edit their information or user also can request role. For edit information, user can update or delete their information while fo request role, user can request any role available and admin will verify before approve the request.

# **4.0 System Use Case**

|  |
| --- |
|  |

## 

## **4.1 Register Account (UC1) Flow of Event**

1. Objective - new user enter the username, password and email to register account.
2. Priority - High
3. Source - Mu’az (Project Manager)
4. Actors - new user
5. Flow of Events -
   1. Basic Flow
      1. Users request to sign-up by clicking on the Register button.
      2. All user are required to fill in the required information to create an account.
      3. A pop-up notification upon successful registration.
6. Includes - None
7. Preconditions - None
8. Postconditions - User successfully create an account
9. Notes/Issues - None

## **4.2 Login (UC2) Flow of Event**

1. Objective - All users able to login into the system
2. Priority - High
3. Source - Mu’az (Project Manager)
4. Actors - All users
5. Flow of Events -
   1. Basic Flow
      1. Users enter their email and password
      2. System validate the entered email and password and logs user into the system

5.1.1 Flow of event 1 - If user entered the wrong email or/and

password, user will not able to log in.There will be a forget password button if the user forgot the password.

1. Includes - None
2. Preconditions - User is registered
3. Postconditions - User is logged into this system
4. Notes/Issues - None

## **4.3 Update Account (U3) Flow of Event**

1. Objective - All users except admin able to edit profile
2. Priority - Medium
3. Source - User
4. Actors - All users except admin
5. Flow of Events -

5.1 Basic Flow

5.1.1 Users click the edit profile button

5.1.2 Users edit their profile

5.1.3 Users save their new information by click the save change

Button.User also can delete their account.

1. Includes - None
2. Preconditions - User is logged into this system
3. Postconditions - User successfully edit their profiles
4. Notes/Issues - None

## 

## **4.4 Request New Role (U4) Flow of Event**

1. Objective - auditor, accountant, auditor delegate and accountant delegate and new user can request other role because a user can have more than one role in one time.
2. Priority - Medium
3. Source - Company Director
4. Actors - new user, auditor, accountant, auditor delegate and accountant delegate and admin
5. Flow of Events -

5.1 Basic Flow

5.1.1 User choose what role to request

5.1.2 Form of the requested role need to be fill out by the user

5.1.3 User submit the form to the admin

5.1.4 Admin receive the request form to verify either the user is eligible or not for the role.

1. Includes - None
2. Preconditions - User is logged into this system
3. Postconditions - The form is sent to admin
4. Notes/Issues - None.

# 

# 

# **5.0 User Interface**

# 

# 

# **6.0 Non Functional Requirement**

## **6.1 Performance Requirements**

The response time dialogs should be in 4 seconds or less and the interface use should be easy to navigate.

## **6.2** **Security Requirements**

The system should be secure and must be encrypted to protect the database and can be changed only by the system administer. User need to be authenticated before having access to any personal data.

## **6.3** **Availability Requirements**

When the database becomes unresponsive due to an error, the system would shut down until the database becomes responsive again. Under normal operation condition, the downtime percentage for the database is a maximum of 10%.

## **6.4** **Scalability Requirements**

When a lot of users use the websites simultaneously during peak operating hours, the users should be able to receive response within 5 seconds. This is because when request load increases, the throughput remains constant but the response time increases linearly. So, the maximum response time during peak hours is 5 seconds.

## 

## 

## **6.5** **Usability Requirements**

The website should be easier to use and matching more closely to user needs and requirements. The website should take less time to accomplish a particular task and more satisfying to use.

# **7.0 Architecture Design**

Client server architecture is a producer-consumer computing architecture where the server acts as the producer and the client as a consumer. Client server architecture works when the client computer sends a resource or process request to the server over the network connection. A server computer can manage several clients simultaneously, whereas one client can be connected to several clients at a time. Simultaneous users can enter the website at the same time without any difficulty.

In this system, we propose the use of client server architecture because this system main operation is done online, main operation is for example the functionality to check trial balance and etc. Server will be the server of this system and client will be all the actor. Each of this actor in this system will interact with the server through a layer of communication. Aside from architecture style, we also propose to use MVC architecture pattern because it is compatible with client server and it provides a separation of concerns. MVC pattern allows the separation of each module so that they are easier to use and would not affect other module greatly.

|  |
| --- |
| Capturewebappmvc.PNG |

When interacting with a Laravel application, a browser sends a request from the user, which is received by a web server and passed on to the Laravel routing engine. The Laravel router receives the request and redirects to the appropriate controller class method based on the routing URL pattern.

Accountant, auditor, accountant delegate, auditor delegate can have more than one role. They can request the new role by fill out a form and submit to the admin for verification. User with certain level of authority will only be able to access information or data according to their respective level. The controller will handle user request using suitable controller whether using HomeController.

Controller will interact with data model to retrieve data. After invoking the model, the controller then renders the final view and returns the complete web page to the user’s browser.

# **8.0 Entity-Relationship Diagram (ERD)**

|  |
| --- |
| erd.png |

Figure above shows the ERD in the database system. There are five entities in this diagram which are roles, user\_roles, users, password\_resets and migration. Roles entity has one to many relationship with user\_role entity and user\_role entity has many to one relationship with users entity. The attributes in roles are role\_id, name and description. User\_role attributes are user\_id, role\_id and user\_role\_id and for users the attributes are user\_id, name, email and password. User entity has one to one relationship with password\_resets entity which has two attributes, token and email and a user is able to reset password. Migration entity has three attributes, migrations\_id, migration and batch. Migration entity exist as a record for example how many times a command had run. It does not has any relationship with others entity.

# **9.0 Integration System**

This user management and authentication module integrated with administrator dashboard module as admin need to approve the user request to have more than one role such as user, an accountant, request to be an auditor. He/she will fill out the form for the request and it is admin’s task to approve it. After admin approve his/her request, user will get notification from admin whether his/her request is approved or not.