**Software Requirements Specification**

for

**Accounting System (Module 8),**

**Release 1.0**

**Version 1.0 approved**

**Prepared by: Mohd Mazlan Bin Mohd Nor (177132)**

**Muhammad Nabil Irfan Bin Abdul Malek (178413)**

**Muhamad Adam Fakrullah Bin Nik Mohamad (179660)**

**Haimanul Iman Bin Unvar Rahman (179631)**

**January 10 , 2017**

**Table of Contents**

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

[1.1 Purpose](#_30j0zll) [2](#_30j0zll)

[1.2 Objective](#_5peuhj7q9pwz) [2](#_5peuhj7q9pwz)

[1.3 Scope](#_3znysh7) [3](#_3znysh7)

[1.4 Design and Implementation Constraints](#_2et92p0) [3](#_2et92p0)

[**2.0 Overall Description**](#_tyjcwt)[**4**](#_tyjcwt)

[2.1 Application Perspective](#_3dy6vkm) [4](#_3dy6vkm)

[2.2 User Classes and Characteristics](#_luppe4pu943r) [4](#_luppe4pu943r)

[2.3 Operating Environment](#_4d34og8) [5](#_4d34og8)

[2.4 Assumptions and Dependencies](#_2s8eyo1) [5](#_2s8eyo1)

[**3.0 System Use Case**](#_17dp8vu)[**6**](#_17dp8vu)

[3.1 Create Company Income (UC 1)](#_xeemnh8rneu9) [7](#_xeemnh8rneu9)

[3.2 Edit Company Income (UC 2)](#_pq6zl79ssk7y) [8](#_pq6zl79ssk7y)

[3.3 Delete Company Income (UC 3)](#_mvztk9msucn) [9](#_mvztk9msucn)

[3.4 Create Company Expense (UC 4)](#_i3m5eyftfce0) [10](#_i3m5eyftfce0)

[3.5 Edit Company Expense (UC 5)](#_wexlnj6mss02) [11](#_wexlnj6mss02)

[3.6 Delete Company Expense (UC 6)](#_embwd7o24z4m) [12](#_embwd7o24z4m)

[3.7 Create Company Transfer (UC 7)](#_v4r7rgmfvk16) [13](#_v4r7rgmfvk16)

[3.8 Edit Company Transfer (UC 8)](#_oufxd2rlqj2e) [14](#_oufxd2rlqj2e)

[3.9 Delete Company Transfer (UC 9)](#_qny2k3bpl2fs) [15](#_qny2k3bpl2fs)

[3.10 Invite Other Accountants (UC 10)](#_rv6svevb1m3h) [16](#_rv6svevb1m3h)

[**4.0 Activity Diagram**](#_rp71dy8txp)[**17**](#_rp71dy8txp)

[**5.0 ERD**](#_1mrcu09)[**27**](#_1mrcu09)

[**6.0 User Interface**](#_41mghml)[**28**](#_41mghml)

[**7.0 Non Functional Requirement**](#_3fwokq0)[**29**](#_3fwokq0)

[7.1 Security Requirement](#_4f1mdlm) [29](#_4f1mdlm)

[7.2 Scalability Requirement](#_19c6y18) [29](#_19c6y18)

[7.3 Availability Requirement](#_2u6wntf) [29](#_2u6wntf)

[7.4 Performance Requirement](#_1v1yuxt) [29](#_1v1yuxt)

[**8.0 Architecture Design**](#_vp1emqxcrc4b)[**30**](#_vp1emqxcrc4b)

# **1.0 Introduction**

## **1.1 Purpose**

The purpose of this document is to give a detailed description of the requirements for the Accounting system for module 8. First of all, module 8 is about accountant and task delegation. So, this document will illustrate the purpose and complete declaration for the development of this system. This document will also illustrate the functional and nonfunctional requirements of the Accounting system for module 8.

This Accounting System is a medium for company directors and company secretary, accountants, auditors, accountants delegate and auditors delegate to interact.

## **1.2 Objective**

The objective of this module is to allow user to monitor the cash flow in and out of the company’s account. The user main functionality of this module is managing the expenses and income of the company’s account. This module is to record the details about the cash flow in and out of the company’s account. The accountant and delegate of the company able to keep track the cash flow of the company by using this module.

## 

## **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**

The Internet connection is a constraint for this application. Since the application fetches data from the database over the Internet, it is compulsory that to have an Internet connection for the application to well-functioning.

# 

# 

# **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 8, the system allow accountant or delegate of the company to manage the income and expenses of money in the company. The accountant and delegate of the company will have to add, delete and update the information about income, expenses and transfer transaction. The User can monitor the cash flow of money whether it is in or out and it can be done easily with this module. The income section is where the accountant or delegate of the company will enter the information about the flow in cash into the company’ account and for expense section, it is for the cash out and transfer section is the transfer of money to other company.

## **2.2 User Classes and Characteristics**

This module can only be access by the accountants and delegates for each company that use this system. They have to log in as accountants or delegates to access this module. In this module, they have to access the company database to retrieve the information about the company and then they will create a company’s income or company’s expenses or transfer transaction to another company. They have to enter the details of the company’s income and save it to the system database and it will be verified and validated by the company’s auditor. If it is not correct or there is something wrong with the details of the company’s income, the accountants or delegates have to recheck and edit the details. The accountant also can delete the company’s income that is already submitted into the system. Same goes for company’s expenses and the transfer transaction, they have can create (add), update (edit) and delete the details that have entered in the system.

## **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 System Use Case**

|  |
| --- |
|  |

## **3.1 Create Company Income (UC 1)**

**Flow of Events**

1. Objective - Allow accountants to create company income
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Income Button
      2. The Web Application shows a list of all incomes from every company.
      3. The accountant selects the add new button to add new income.
      4. The accountant fills all the necessary data in the form and clicks on save.
      5. The data will be saved and the list will be updated.
   2. Alternative Flow
      1. The accountant stops filling the form by pressing on the hide button.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully creates an income.
9. Notes/Issues - None

## **3.2 Edit Company Income (UC 2)**

**Flow of Event**

1. Objective - Allow accountants to edit company income
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Income Button
      2. The Web Application shows a list of all incomes from every company.
      3. The accountant can select any income on the list to edit it.
      4. The accountant fills all the necessary data in the form and clicks on save.
      5. The data will be saved and the list will be updated.
   2. Alternative Flow
      1. The accountant stops filling the form by pressing on the hide button.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully edits an income.
9. Notes/Issues - None

## 

## **3.3 Delete Company Income (UC 3)**

**Flow of Event**

1. Objective - Allow accountants to delete company income
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Income Button
      2. The Web Application shows a list of all incomes from every company.
      3. The accountant can select any income on the list.
      4. The accountant selects the delete button to delete the income.
      5. The data will be deleted and the list will be updated.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully deletes an income.
9. Notes/Issues - None

## 

## **3.4 Create Company Expense (UC 4)**

**Flow of Event**

1. Objective - Allow accountants to create company expense
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Expense Button
      2. The Web Application shows a list of all expenses from every company.
      3. The accountant selects the add new button to add new expense.
      4. The accountant fills all the necessary data in the form and clicks on save.
      5. The data will be saved and the list will be updated.
   2. Alternative Flow
      1. The accountant stops filling the form by pressing on the hide button.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully creates an expense.
9. Notes/Issues - None

## 

## **3.5 Edit Company Expense (UC 5)**

**Flow of Event**

1. Objective - Allow accountants to edit company expense
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Expense Button
      2. The Web Application shows a list of all expenses from every company.
      3. The accountant can select any expense on the list to edit it.
      4. The accountant fills all the necessary data in the form and clicks on save.
      5. The data will be saved and the list will be updated.
   2. Alternative Flow
      1. The accountant stops filling the form by pressing on the hide button.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully edits an expense.
9. Notes/Issues - None

## 

## **3.6 Delete Company Expense (UC 6)**

**Flow of Event**

1. Objective - Allow accountants to delete company expense
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Expense Button
      2. The Web Application shows a list of all expenses from every company.
      3. The accountant can select any expense on the list.
      4. The accountant selects the delete button to delete the expense.
      5. The data will be deleted and the list will be updated.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully deletes an expense.
9. Notes/Issues - None

## 

## **3.7 Create Company Transfer (UC 7)**

**Flow of Event**

1. Objective - Allow accountants to create company transfer
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Transfer Button
      2. The Web Application shows a list of all transfers from every company.
      3. The accountant selects the add new button to add new transfer.
      4. The accountant fills all the necessary data in the form and clicks on save.
      5. The data will be saved and the list will be updated.
   2. Alternative Flow
      1. The accountant stops filling the form by pressing on the hide button.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully transfers money.
9. Notes/Issues - None

## 

## **3.8 Edit Company Transfer (UC 8)**

**Flow of Event**

1. Objective - Allow accountants to edit company transfer
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Transfer Button
      2. The Web Application shows a list of all transfers from every company.
      3. The accountant can select any transfer on the list to edit it.
      4. The accountant fills all the necessary data in the form and clicks on save.
      5. The data will be saved and the list will be updated.
   2. Alternative Flow
      1. The accountant stops filling the form by pressing on the hide button.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully edits a transfer.
9. Notes/Issues - None

## **3.9 Delete Company Transfer (UC 9)**

**Flow of Event**

1. Objective - Allow accountants to delete company transfer
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Transfer Button
      2. The Web Application shows a list of all transfers from every company.
      3. The accountant can select any transfer on the list.
      4. The accountant selects the delete button to delete the transfer.
      5. The data will be deleted and the list will be updated.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully deletes a transfer.
9. Notes/Issues - None

## **3.10 Invite Other Accountants (UC 10)**

**Flow of Event**

1. Objective - Allow accountants to invite other accountants to edit data for any company
2. Priority - High
3. Source - Accountant
4. Actors - Accountant
5. Flow of Events -
   1. Basic Flow
      1. The accountant clicks on Income, Expense or Transfer Button.
      2. The Web Application shows a list of all incomes, expenses or transfers from every company according to the button clicked.
      3. The accountant select the Invite button.
      4. The accountant fills in the email address of other accountants to invite them
      5. The other accountants that received the email can edit any data in the database according to the permission given
      6. The data will be saved and the list will be updated.
6. Includes - None
7. Preconditions - Accountant has to be logged in
8. Postconditions - Accountant successfully invite other accountants to edit data.
9. Notes/Issues - None

# **4.0 Activity Diagram**

|  |
| --- |
|  |

**4.1 Create Company Income**

|  |
| --- |
|  |

**4.2 Edit Company Income**

|  |
| --- |
|  |

**4.3 Delete Company Income**

|  |
| --- |
|  |

**4.4 Create Company Expense**

|  |
| --- |
|  |

**4.5 Edit Company Expense**

|  |
| --- |
|  |

**4.6 Delete Company Expense**

|  |
| --- |
|  |

**4.7 Create Company Transfer**

|  |
| --- |
|  |

**4.8 Edit Company Transfer**

|  |
| --- |
|  |

**4.9 Delete Company Transfer**

|  |
| --- |
|  |

**4.10 Invite Other Accountants**

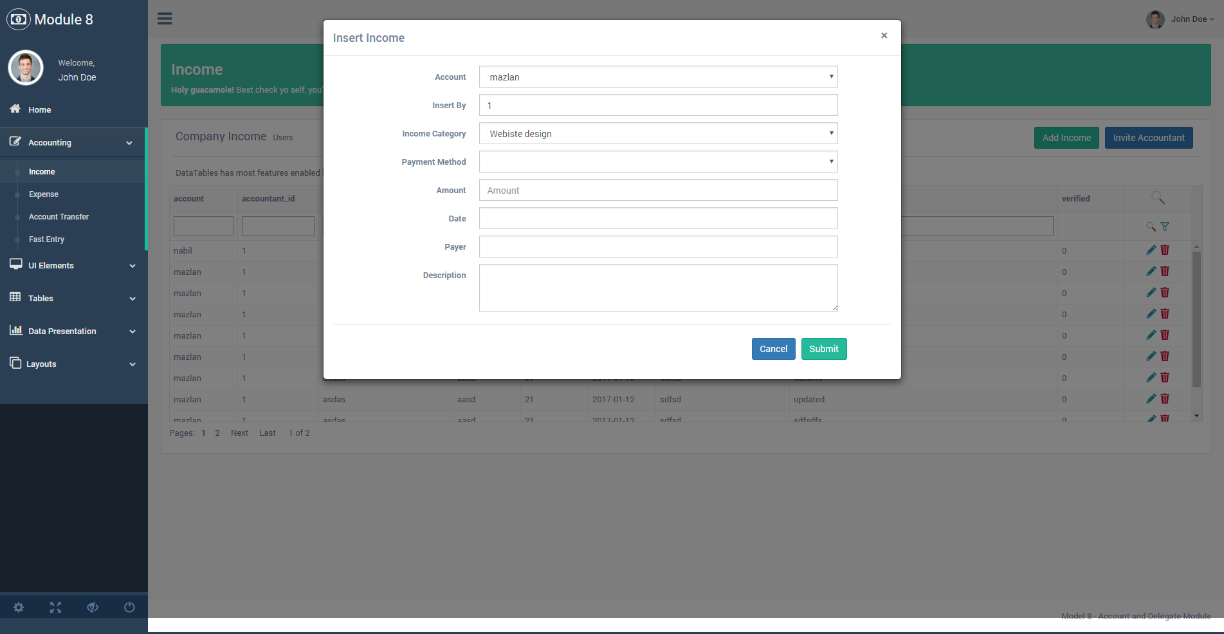
# **5.0 ERD**

|  |
| --- |
|  |

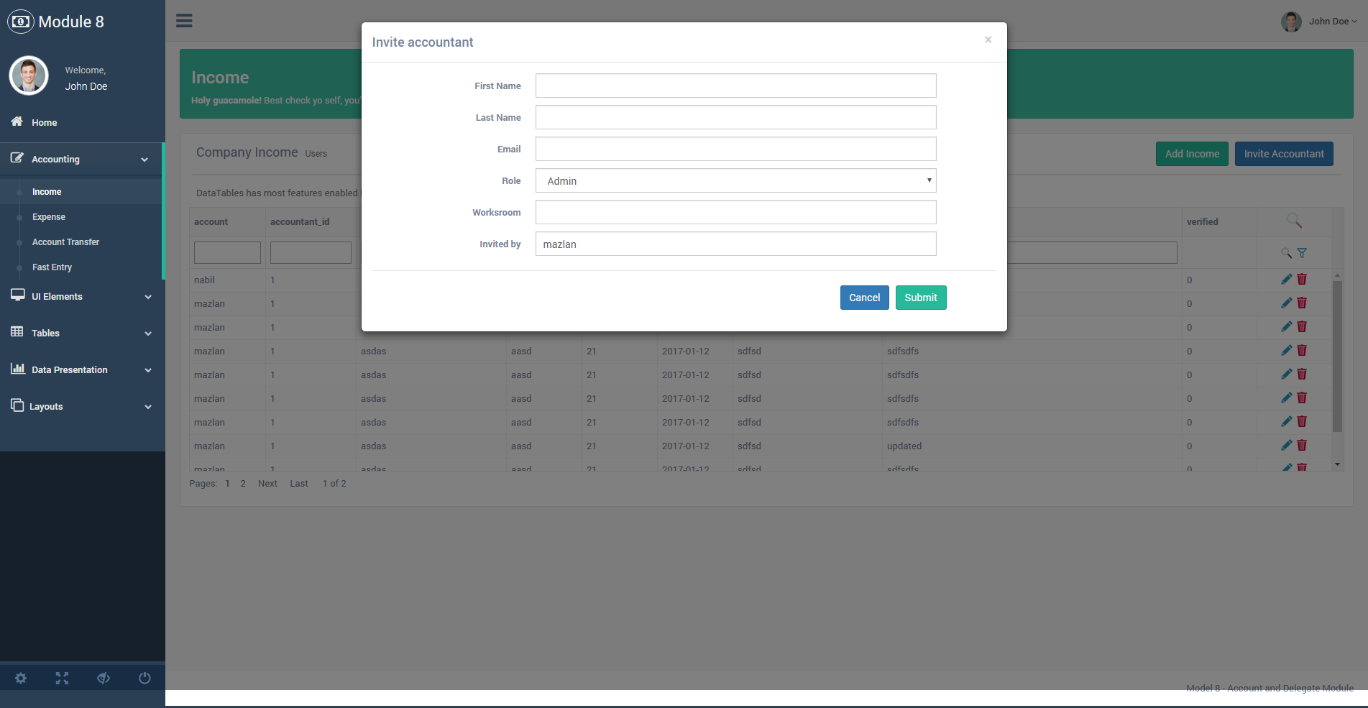
# **6.0 User Interface**

# 

Incomes Interface



Add Income



Invite Other Accountant

# 

Invited Interface Mail

# **7.0 Non Functional Requirement**

## **7.1** **Security Requirement**

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

## **7.2** **Scalability Requirement**

When a lot of accountants use the websites simultaneously during peak operating hours, the accountants 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.

## **7.3** **Availability Requirement**

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%.

## **7.4 Performance Requirement**

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

# **8.0 Architecture Design**

The architecture pattern that we adopt for our module is Model-View-Controller pattern. This pattern is widely used by many mobile app development framework. So, this architecture pattern can be implemented into our system because of its high availability. The reason why we are choosing this architecture is because of its separation of concern. In this pattern, it is clearly a layered architecture and it basically has 3 main layer which is model, view and controller. Each layer play their own role. The model layer often contains the business logic and information about the types of data in the database. The view layer, it is often CSS, JavaScript and HTML with dynamic embedded code and lastly is the controller layer which has various rules and methods for transforming the data moving between the view and the model layer. Each layer is really maintainable, testable, easy to assign separate “roles” and easy to update and enhance layers separately. So, we can develop this website according to our specialization and parallel development can be practice by separate teams. Furthermore, it is also highly reusable for each components of MVC and it can be used by another module of the system.

## 

# 