**Software Requirements Specification**

**For**

**Module 7: Auditor and Delegate**

**Version 1.0**

**Prepared by : 1. Nurul Ain Binti Mohamed Nasir 162356**

**2. Nurfarah Lyana Binti Zulkiply 179865**

**3. Nur Shuhada’ Binti Harris Rizal 179729**

**4. Atiqah Adlina Binti Mohd Naziran 180502**

**5. Fatin Delailah Binti Adnan 179845**

**Table of Contents**

**Table of Contents……………………………………………………………………….….....2**

**1.0** **Introduction………………………………………………………………………........3-4**

1.1 Purposes……………..……………..……………………………………………….....3

1.2 Objectives……………………...……………………………………………………....3

1.3 Scope……………………………………………………………………….………….4

**2.0** **Overall Description………………………………………………………………….....4-6**

2.1 User Classes and Characteristics…………………………………………….…….…..4

2.2 Operating Environment…………………………………………………………........5

2.3 Design and Implementation Constraints…………………………………………......5

2.4 Assumptions and Dependencies………………………………………………..........6

**3.0** **External Interface Requirements……………………………………………….……6-8**

3.1 User Interfaces……………………………………………………………..…….....6-8

**4.0** **System Use Cases…………………………………………………………………..........9**

**5.0 Functional Requirements………………………………………………………............10**

**6.0** **Architecture Design…………………………………………………………………......10**

**7.0** **Entity-Relationship Diagram (ERD)……………………………………..………........11**

**8.0 Integrated System…………………………………………………………………...****12-13**

**1.0 INTRODUCTION**

**1.1 Purposes**

The focus of this project is to create an accounting system. We have been assigned to do the ‘Auditor and Delegate’ module. Auditing is the process of reviewing and investigating any aspect of a business, whether financial or nonfinancial. Auditors are fully trained to spot areas of needed improvement, potential dangers and incidents of unethical conduct in their area of expertise. Audits can disrupt the normal flow of business in a company, but the ability to spot and address potential weaknesses can outweigh any temporary losses of productivity. Among the range of issues audits can review accounting audits.

In this project, auditors play role in audit the account until it is approved by them. Until then, there will be feedback among accountant and auditor to correct the account (by accountant) where each feedback with corrected account will have the version until the account is approved and cannot be edited or changed anymore.

## 1.2 Objectives.

1.2.1 To audit the account.

1.2.2 To invite other auditor to audit the account.

1.2.3 To make sure to be the last person to verify the account.

1.2.4 To make sure the version are provided to the corrected account before being approved.

1.2.5 To make sure the approved account cannot be changed or edit anymore.

**1.3 Scope.**

The module is to be use in any company that have finance and accounting department in Malaysia. The target user of our module is the end-user. This project is mainly emphasized on developing a system for auditing process. It is also have other features that support the auditing process.

The auditor and delegate module consist several databases which include three database which includes the auditor database, audit database, and account database.

# **2.0 OVERALL DESCRIPTION**

## 2.1 User Classes and Characteristics

This table shows user levels and their characteristics in Auditor and Delegate module.

|  |  |  |
| --- | --- | --- |
|  | User Classes | Characteristics |
| I. | Auditor | Auditor plays an important role because auditor audits the account of the company. Auditor can invite other auditors to audit the account. Besides, auditor also can send feedback to the accountant that in charge on the account if there are any mistakes in the accounts. Auditors also are the one who approve the accounts after there is no mistake in the account. |
| II. | Accountant | Accountant sends the account that need to be audited to the auditor. So, auditor receives the account. Each time if there is any mistakes in account, auditor will informed it to the accountant to correct it through feedback. Where each feedback with corrected accounts from accountant will have the version. |

## 2.2 Operating Environment

2.2.1 Software

i) Command Prompt

Also known as cmd.exe or cmd (after its executable file name), is the command-line interpreter on Windows operating systems.

ii) WAMP

WAMP cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP programming language.

iii)Sublime Text

Sublime Text is a proprietary cross-platform source code editor with a Python application programming interface (API). It natively supports many programming languages and markup languages, and its functionality can be extended by users with plugins, typically community-built and maintained under free-software licenses.

2.2.2 Hardware

i) Laptop

Laptop is a small, portable personal computer. Easy to carry that make laptop as the hardware use to develop DGD.

## 2.3 Design and Implementation Constraints

There are several constraints in the Auditor and Delegate module. Firstly, this module only can support up to 200GB. The application only supports 1 language which is English. Thus those who are not native English speakers will find it hard to interact with the system interface.

## 2.4 Assumptions and Dependencies

There are a few problem that we faced during developing this module which are the difficulties to integrate with other module as some of the information need by us are dependent to the other modules.

# **3.0 EXTERNAL INTERFACE REQUIREMENTS**

**3.1 User Interface Design**

3.1.1 UI design for Auditor Module

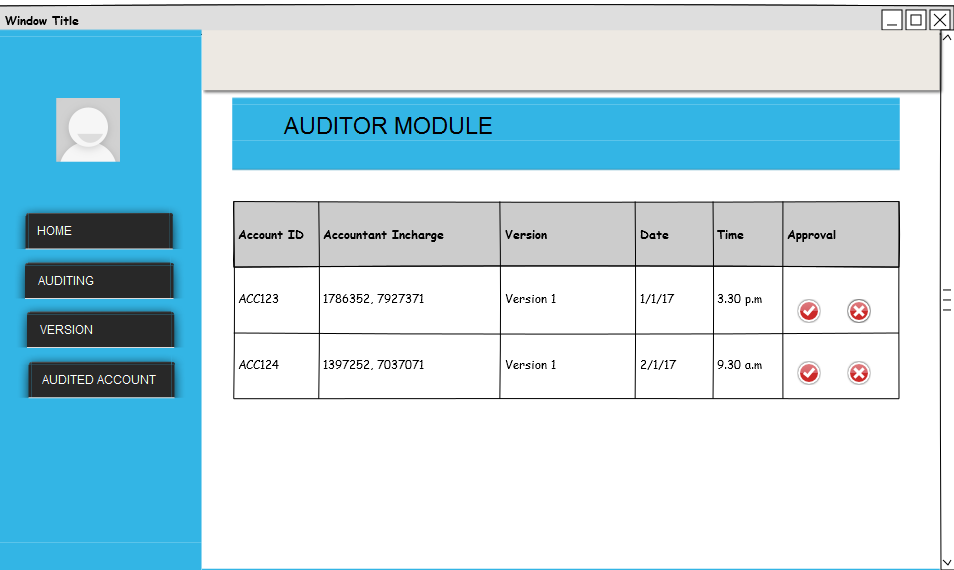


Diagram 1 shows the interface design of the auditor’s view once he log in

## 3.1.2 UI design for Audit account

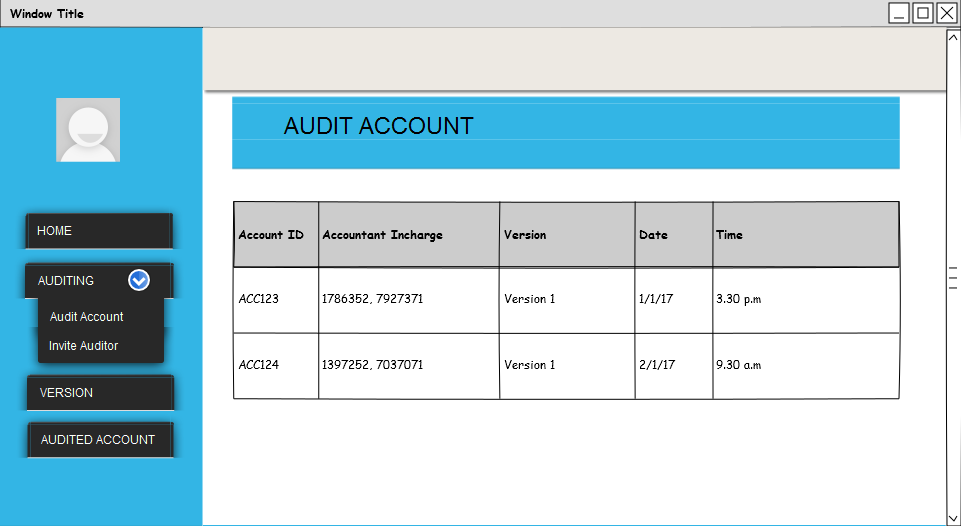


Diagram 2 shows the design of the view of accounts that needs to be audit

3.1.3 UI design for Invite other auditor

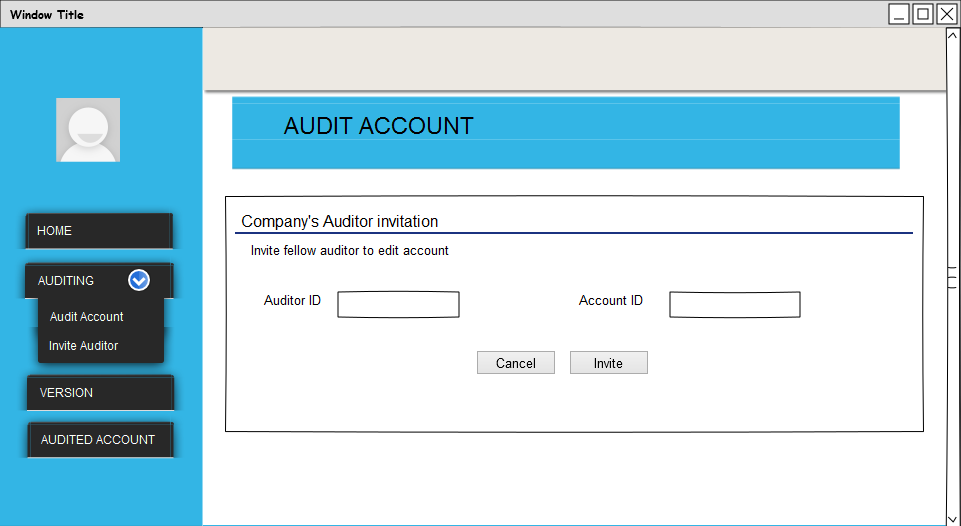


Diagram 3 shows the design view of invite other auditor to audit account.

3.1.4 UI design for Account version view

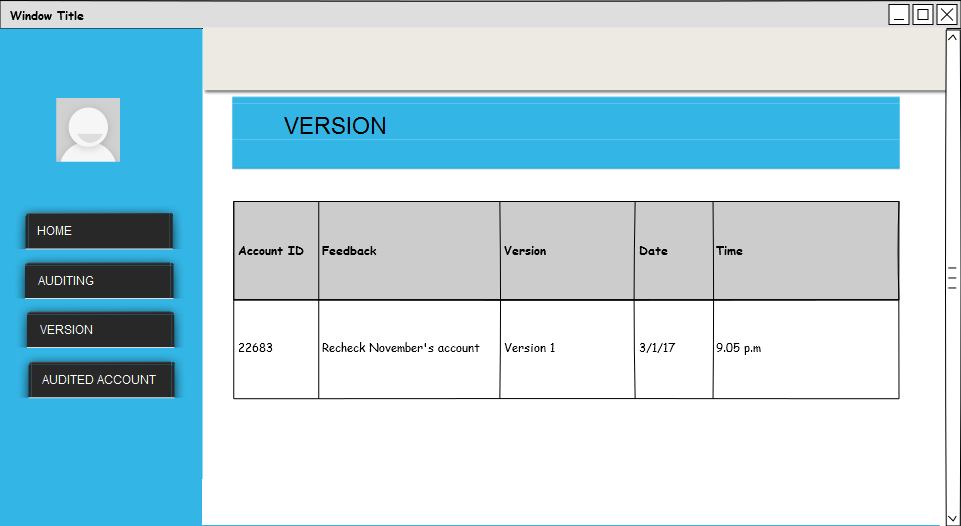
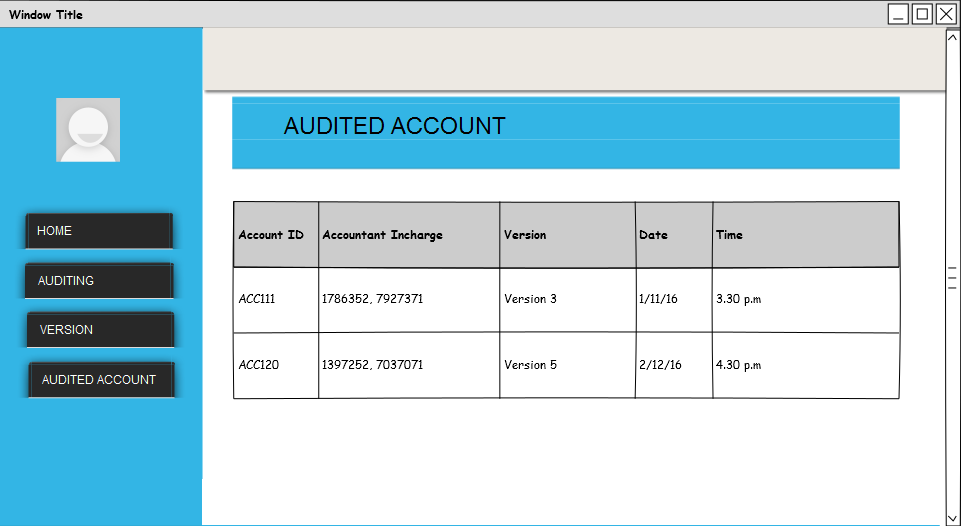


Diagram 4 shows the design view of version for corrected accounts before audit

3.1.5 UI design for audited account



## Diagram 5 shows the design for audited account

# **4.0 SYSTEM USE CASE**

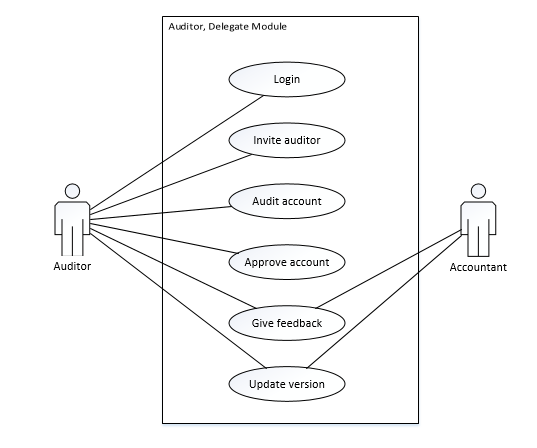


Diagram 6. Use case for Auditor and Delegate module.

This is the use case for Auditor and Delegate Module which have 6 use case which are login, invite auditor, audit account, approve account, give feedback and update version. The actor involve is the Auditor. The Auditor can invite other auditor to audit the account. Auditor also needs to audit the account. After auditing, the Auditor also approves the account after there are no mistakes in account. Auditor gives feedback to the accountant if there is any wrong in the account until the account is approved. And Auditor also needs to update the version of the account each time the feedback is occur between accountant and auditor to correct the mistakes in the account.

**5.0 FUNCTIONAL REQUIREMENTS**

5.1 The auditor shall be able to invite other auditor to audit the account.

5.2 The auditor shall be able to audit the account themselves.

5.3 The auditor shall be able to verify or approve the account.

5.4 The auditor shall be able to give feedback to the account.

5.5 The auditor shall be able to update the account.

**6.0 ARCHITECTURE DESIGN**

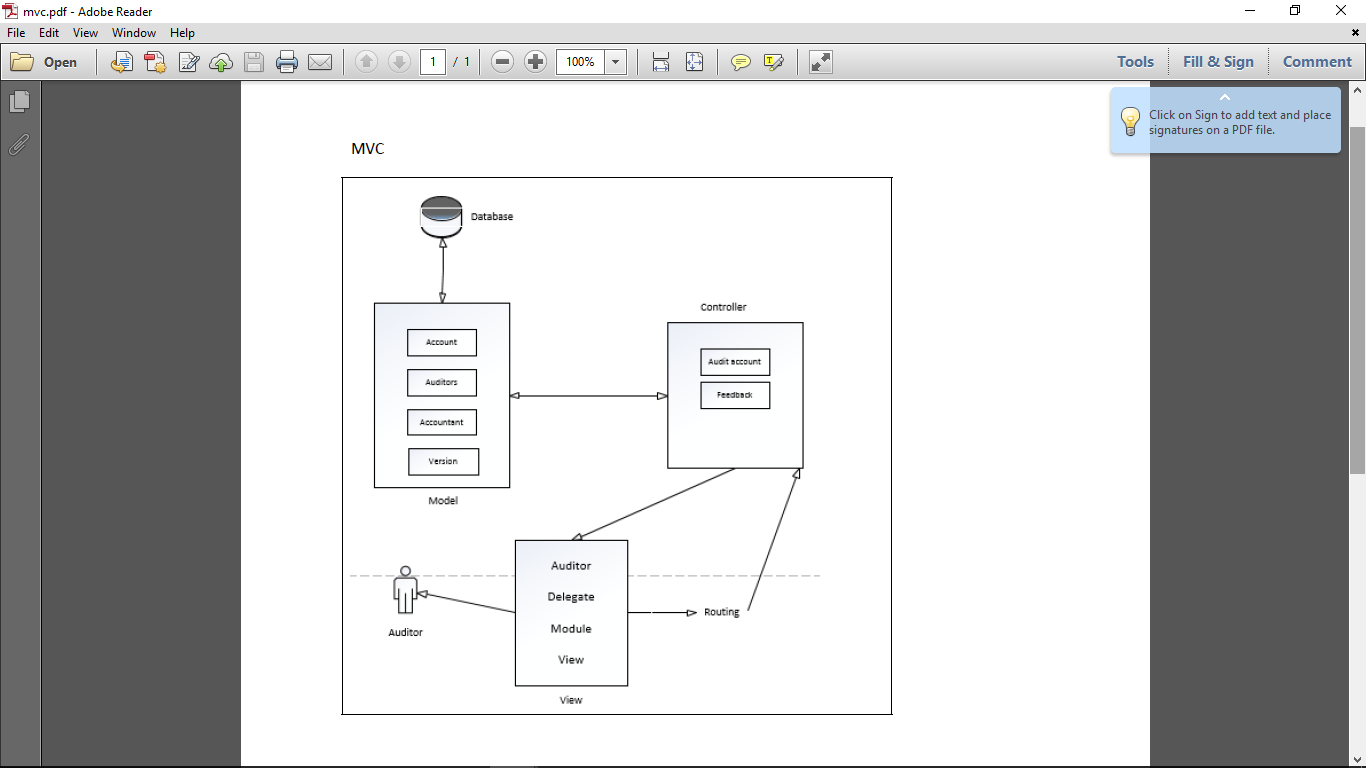


Diagram 7 shows the architecture diagram of the auditor and delegate module

Diagram above shows the architecture diagram of this module where we are using Model View Controller (MVC) as we are using Laravel Framework.

**7.0 ENTITY-RELATIONSHIP DIAGRAM (ERD)**

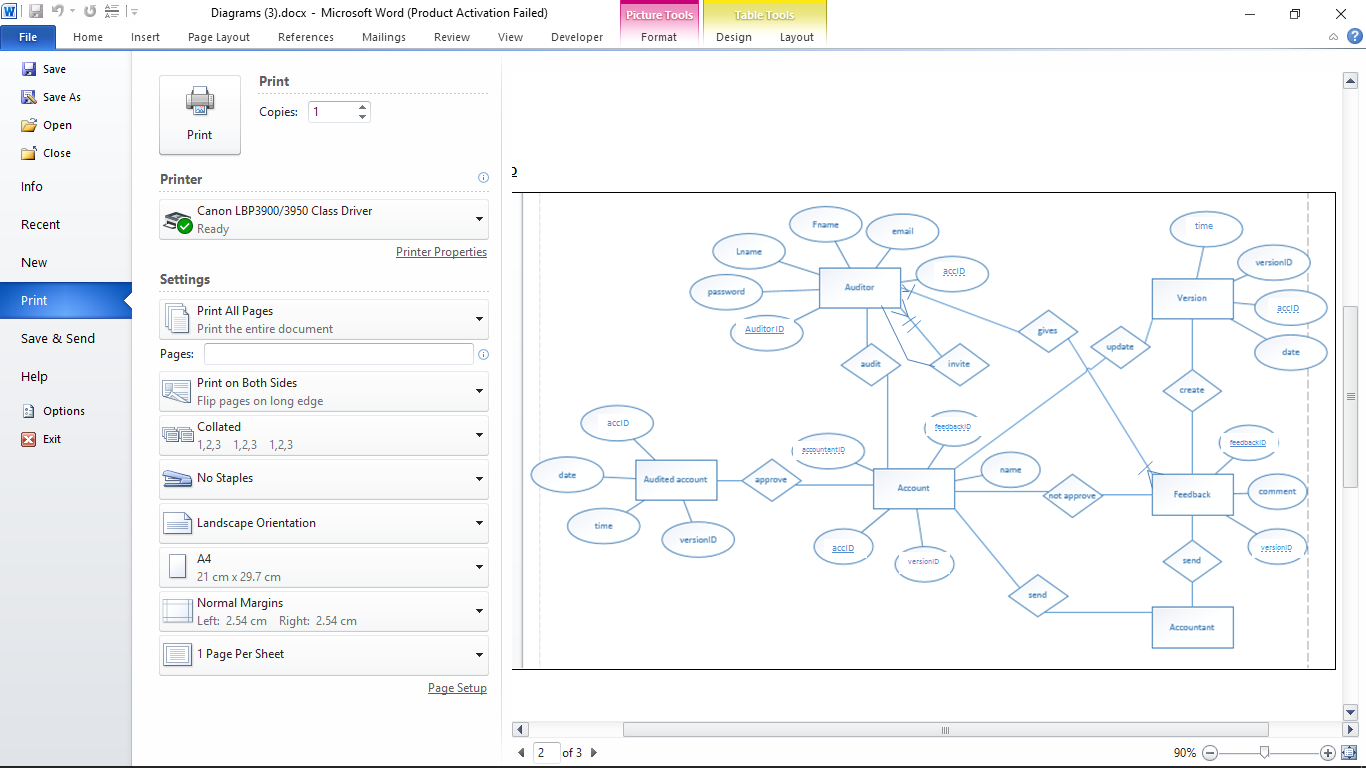


Diagram 8 shows ERD diagram for Auditor and Delegate Module.

This diagram shows in Auditor and Delegate module have four(6) Entities which includes:

1. Audited Account.
2. Auditor
3. Account.
4. Accountant.
5. Feedback
6. Version.

**8.0 INTEGRATED SYSTEM**

There are few interface has been created for this module. Here are few of these interfaces:

8.1 Auditors module

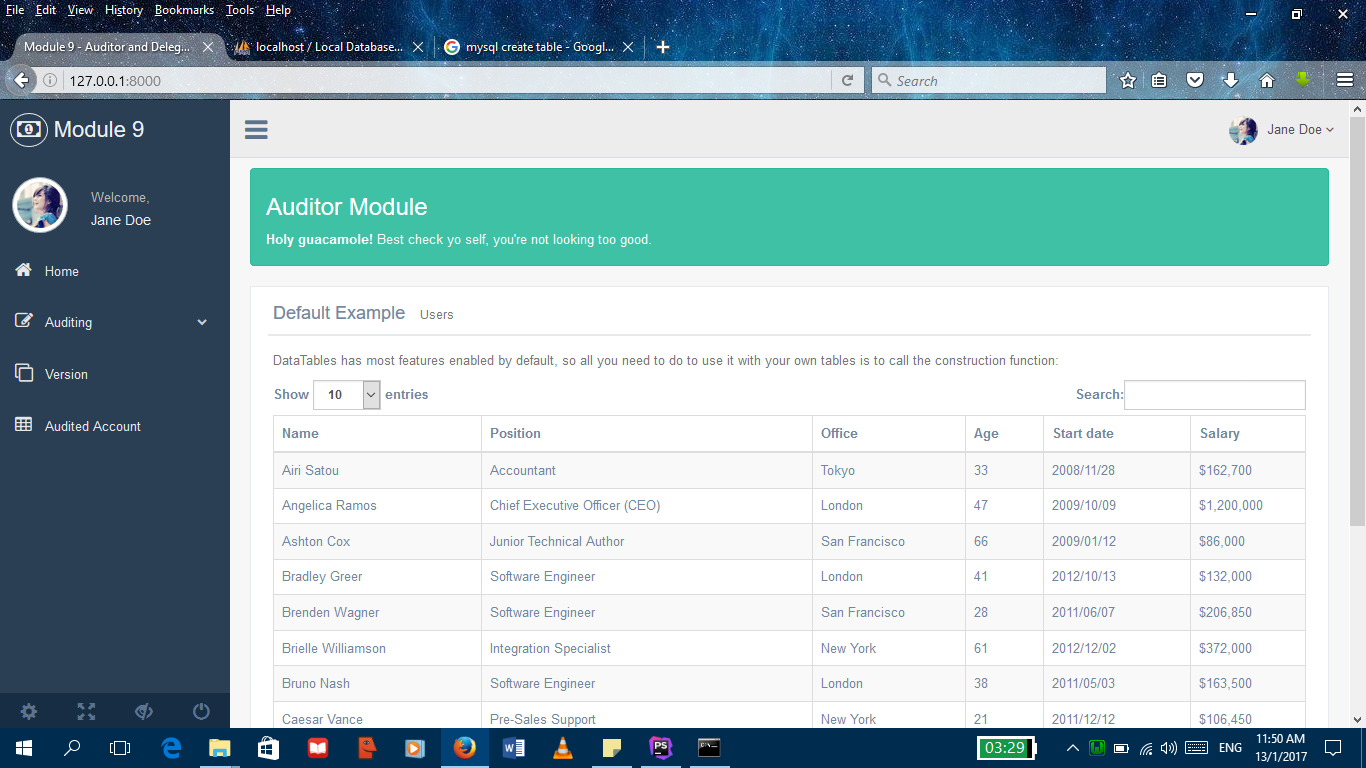


Diagram 9 shows the auditor module for auditor view after log in

8.2 Audit Account

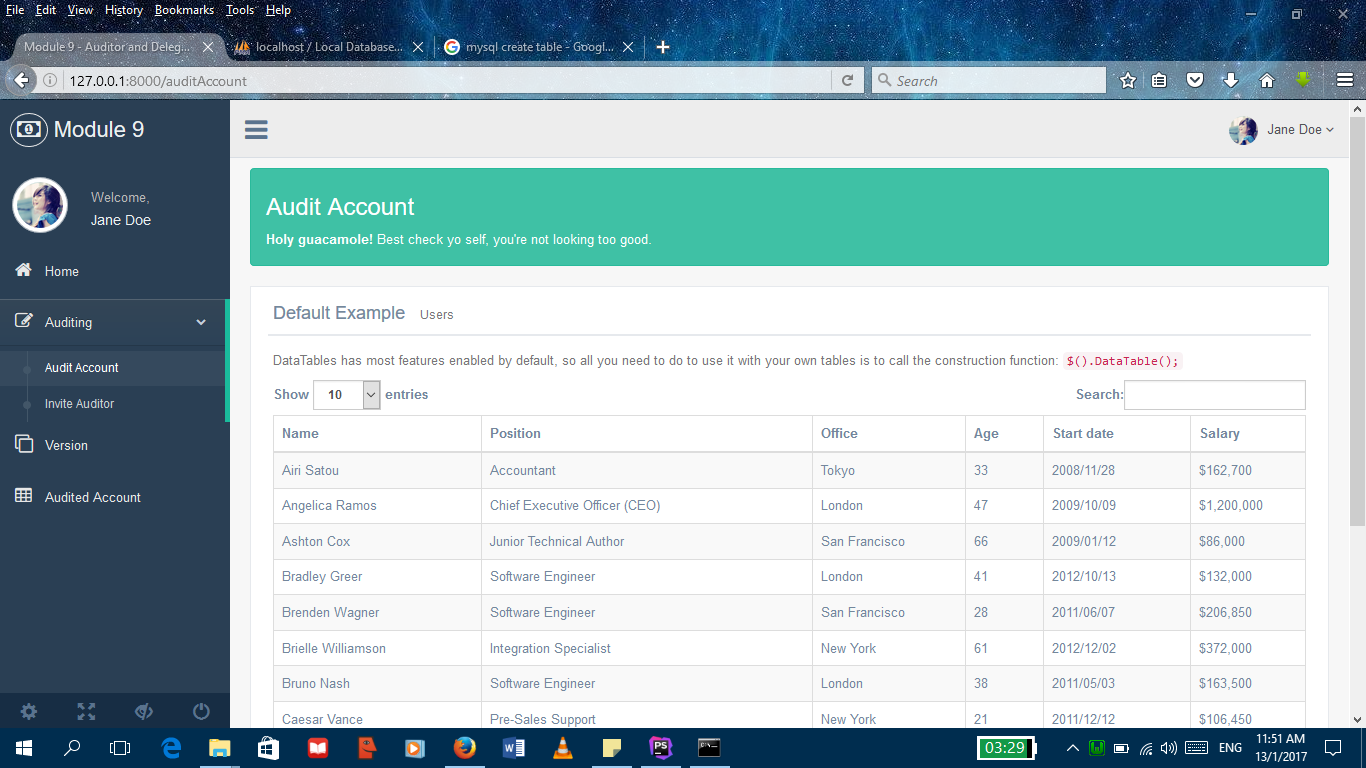


Diagram 10 shows the account that need to be audited

8.3 Invite Auditor

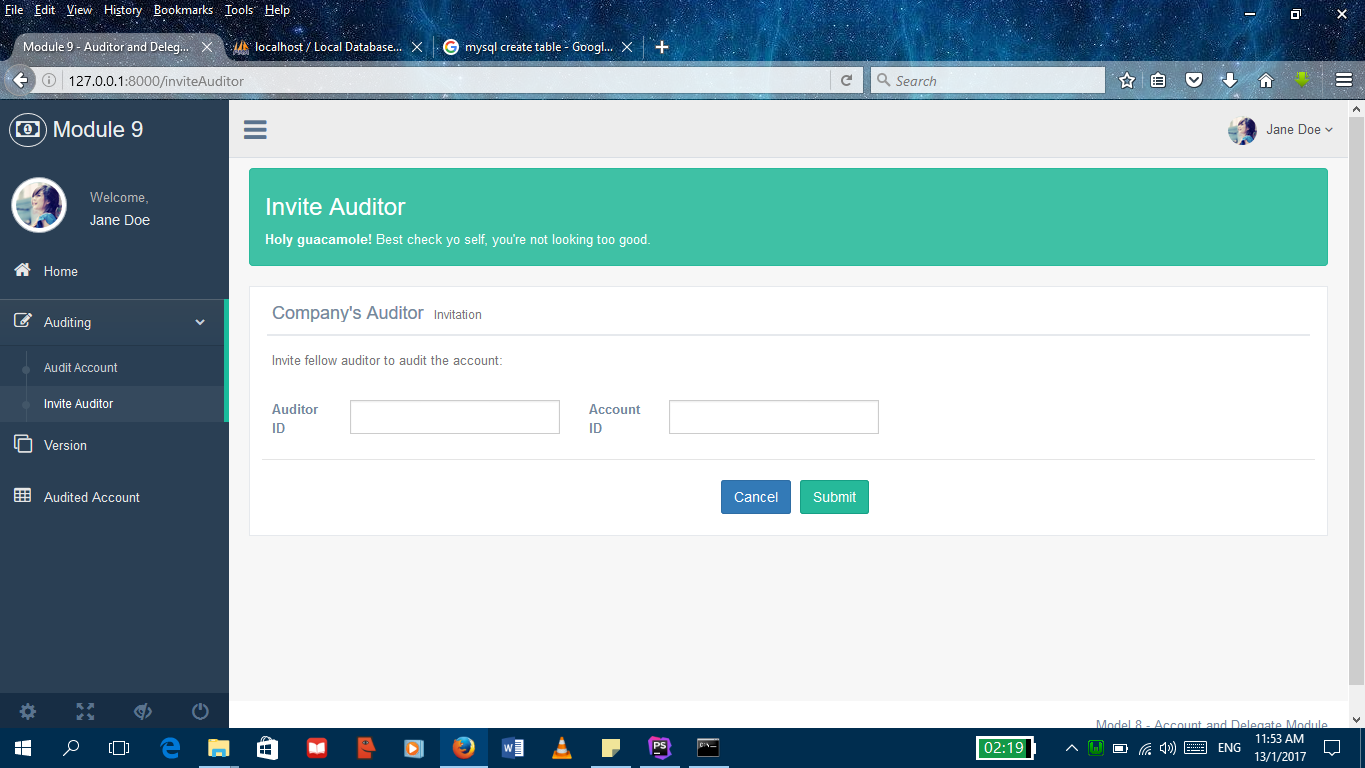


Diagram 11 shows the interface to invite other auditor to audit the account