**COMP 8045**

**Practicum Report**

**Suplus Office Management System**

Sabrina Xue

A00681622

Table of Contents

1. Introduction 4

1.1 Project Description 4

1.2 Current Problems 4

2. Company Background 4

3. Project Objectives 5

4. Solution chosen for this project 7

PHP 7

Bootstrap 7

jQuery 7

JSP 7

CSS 8

Mongo database 8

5. Development Methodology 9

6. Architecture and design 10

Component Diagram: 11

7. Analysis and Design 12

7.1 System Function 12

Manage employee information 12

Manage customer information 12

Manage project information 13

Login: 13

7.2Use Case Diagram 14

7.3 Domain Model Diagram 16

7.4 System Sequence Diagram 17

7.5 Software Design Class Diagram 23

7.6 Database Design 24

MySQL 24

Employee table 25

Customer table 26

Department table 28

Project table 28

Task table 29

NoSQL 30

Collection 30

Post message 31

Post picture 32

Comment 33

8. Access Control 35

Role based access control 35

9. Innovation 35

10. Development and Implementation 36

10.1 Programming Language Chosen 36

10.2 System Perspective 36

Employee module 38

Administrator module 41

Team leader module 49

11. Research 53

11.1 Why NoSQL? 53

11.2 Advantage & Disadvantage 53

11.3 Why Mongo DB? 55

12. Test 56

12.1 User Test Plan 56

Result of user testing 57

Debugging 61

12.2 System Function Test 63

Result of system function testing 63

All users test cases 63

Employee module test cases 64

Admin module test cases 65

Team leader module test cases 66

13. Conclusion 68

14. Future Enhancement 69

15. Appendix 70

A. Work Breakdown 70

B. Test Cases 72

C. Comments of User Testing 101

Reference 102

## Introduction

### 1.1 Project Description

The aim of this practicum project is to develop a system to connect and organize the employees’ information of Suplus Technology, project, and customers. The Suplus Office Management System is designed to be utilized by the employees, administrator, and the team leader in the company. An important aspect of the management system is that it permits its users to communicate efficiently. Therefore a key function of this system is that users can exchange messages, and share pictures. This paper will outline the functions of the system and how it benefits the aforementioned users. Firstly, the system allows the employee to view personal projects and tasks information. Secondly, the system ensures that the administrator is able to organize information of employees, and also customer and project information such as adding, editing and deleting. Thirdly, the system enables the team leader to divide each project into several small tasks and assign these tasks to employees in the software development department. This document includes all the necessary information to explain how the management system operates.

### 1.2 Current Problems

The purpose of this project is to provides solution to the following:

1. How to arrange the system architecture according to the client’s (Suplus Technologies Co. Ltd.) requirement.
2. How to design a user friendly and functional user interface.
3. How to design a social website by using Mongo database.
4. How to design a MySQL database to maintain Suplus software development department employee, project and customer information.

## Company Background

**The industry sponsor** is Beijing Suplus Technologies Co. Ltd.

Suplus Technologies Co. Ltd aims to China Enterprise Communications and application market, dedicated to the development and promotion of new technologies, new products. During the time, Suplus Technologies has helped China Unicom, China Mobile, China Telecom, and Jinjiang Inn to build call center system. Due to Suplus Technologies Co. Ltd is expanding their market to south part of China, there is going to set more branches in these area. Therefore, Suplus Technologies Co. Ltd requires a better functional and more convenient system to manage software development department employees, customers and projects.

## Project Objectives

The objective of the project is to help software development employees of Suplus Technologies Co. Ltd to organize project and task better, and to maintain the pace of the work schedule in order to improve the work efficiency. At the same time, Suplus Office Management System provides a platform that allows employees to exchange information and share their personal information.

Suplus Office Management System consists three actors, and based on these actors the system could divide into three modules, which are software development employee module, administrator module, and team leader module. As mentioned before, each project could divide into several tasks. Employee could check personal projects and tasks. Team leader could divide the project and add tasks for software development employees. In addition, there is a communication page allows employees sharing information.

* The software development employee module includes the following main functions:
* Check personal project/task information
  + Check all personal task and project information
  + Check a specific tasks
* Modify personal information
  + Update personal information such as username, password, status, address, post code, email, phone number
* Communication
  + Post message
  + Post pictures
  + Comment on posted message or picture
* The Administrator module consists the following functions:
* Employee maintain
* Check employee information
* Search a specific employee information
* Add new employee information
* Edit employee information
* Delete employee information
* Customer maintain
* Check the entire customer information
* Search a specific customer information
* Add new customer information
* Edit customer information
* Delete customer information
* Project maintain
* Check the whole project information
* Search specific project information
* Add new project
* Edit project information
* Delete project information
* Modify personal information
  + Update personal information such as username, password, status, address, post code, email, phone number
* Communication
  + Post message
  + Post pictures
  + Comment on posted message or picture
* The team leader module has the following functions:
* Mange project
* Check the entire project information
* Divide a project into small tasks
* Assign a task to employee
* Update personal information
  + Update personal information such as username, password, status, address, post code, email, phone number
* Communication
  + Post message
  + Post pictures
  + Comment on posted message or picture

## Solution chosen for this project

The technologies are used to develop this system are all open source software according to the industry sponsor requirement in order to cut the cost of development. The technologies includes:

* MAMP 3.0.7.3 (Manage apache and MySQL server)
* PHP 5.5 as programming language
* Html CSS
* JavaServer Pages (JSP)
* JQuery
* Bootstrap
* MySQL Server 5.5.38
* Mongo Database
* RoboMongo 0.8.4

### PHP

PHP is a server-side scripting language designed for web development. Since PHP code can mix with HTML code and various engines and web frameworks. Suplus Office Management System is a PHP web-based application.

### Bootstrap

As an extension of Java script, Bootstrap contains HTML and CSS designed templates. It is compatible with most major browsers and supports responsive design. In this system, Bootstrap is most important because almost every page includes bootstrap theme, such as table style, search textbox, and button style.

### jQuery

jQuery's architecture allows developers to create plug-in code to extend its functionality. jQuery library allows the creation of powerful dynamic web pages and web applications. In this system, JQuery combined with Bootstrap to update interface design.

### JSP

JSP helps [software developers](http://en.wikipedia.org/wiki/Software_developer) to create [dynamically generated web pages](http://en.wikipedia.org/wiki/Dynamic_web_page) based on [HTML](http://en.wikipedia.org/wiki/HTML), [XML](http://en.wikipedia.org/wiki/XML), or other document types. It can help to change the presentation of system without learning Java Script. Java Server pages provide better solution to create dynamic and interactive web pages by embedding programming language directly into HTML.

### CSS

CSS is a [style sheet language](http://en.wikipedia.org/wiki/Style_sheet_language) used for describing the [look and formatting](http://en.wikipedia.org/wiki/Presentation_semantics) of a document written in a [markup language](http://en.wikipedia.org/wiki/Markup_language). CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for [web applications](http://en.wikipedia.org/wiki/Web_applications), and user interfaces for many mobile applications. This system is using CSS to create a user-friendly interface, especially on the page layout arrangement.

### Mongo database

Mongo database is based on document to store data, and organize data as collections of documents, which could provide better data model to fit in data structure than relational database. For more detail, please check 11.3 why Mongo DB.

## Development Methodology

The sponsor required to using prototype as the development methodology. Therefore, this project adapted prototype methodology. According to the users requirement, I designed the following versions:

1. Version 1.0

The version 1.0 is a simple mock-ups based on the users’ requirement. This is not the perfect version, but helps to organize and lists the functions of employee, administrator and team leader modules. The functions, such as, update personal information, communication are the major functions for all the users. In addition, there are manage employee, customer, and project information for administrator module, and team leader module includes manage project into tasks and distributes to selected employees. The version 1.0 has been presented to the users of Suplus Technology, and the users have provided suggestions and improvement.

1. Version 1.1

Based on the users’ feedbacks, I updated the version 1.1 to version 1.1. The version 1.1 helped to created the next version.

1. Version 1.2   
   According to the version 1.1, generated version 1.2 with HTML and CSS. This version changed the previous mock-ups into a web system by using HTML and CSS.
2. Version 1.3

The version 1.3 adapted PHP, MySQL and Mongo DB to develop the system. The user testing and the function testing are based on this version. The testers include the employee, administrator, team leader of Suplus Technology, volunteers, and myself. All the testers were asked to provide the test result and suggestions.

1. Version 1.4

According to the testing results and suggestions, version 1.4 debugged and adapted Bootstrap to improve the page layout of user interface. The version 1.4 has been updated to the final version.

The prototype methodology helped me to continue refining the system, and decreased the possibility of missing functions.

## Architecture and design

Server Configuration

* Web server: Apache 2.4.9 or higher
* Server-side scripting language: PHP 5.5.12 or higher
  + Operation system: Operation system Widowns 7/8, UNIX or Linux operating system
  + Database: MySQL Server 5.5.38, Mongo Database V2.6.7
  + IE Browser: Safari, Chrome

Client Configuration

* + Software: Operation system Widowns 7/8, UNIX or Linux operating system
* IE Browser: Safari, Chrome

Suplus Office Management System allow user to access user interface by using web browser through the Internet. User interface allows system users to send SQL command from application server to database server in order to complete tasks.

### Component Diagram:

There are three components, the first one is user interface layer, user login the system by using any web browser, and make sure the internet server MAMP (a solution providing apache, PHP and MySQL server) and Mongo database is turning on. The second layer is coding source, which includes the employee module, admin module, and team leader module. The programming language and technologies adapted are PHP, JQuery, JSP, and Bootstrap. The last component is database, the information are stored in the MySQL server and the NoSQL (Mongo database). The following picture could explain three components.

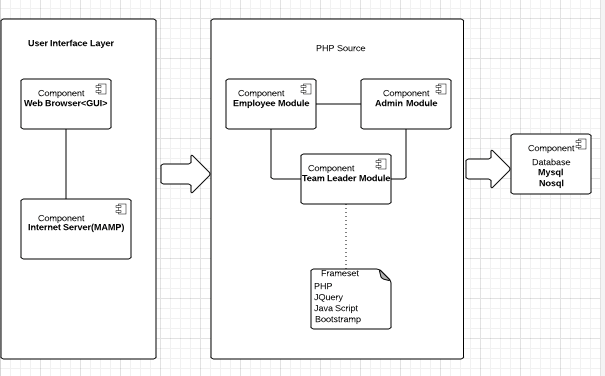


Figure 1 Component Diagram

## Analysis and Design

### 7.1 System Function

Suplus Office Management System has five major functions, which are manage employee information, manage customer information, manage project information, login, and communicate which allows users to share information.

All the users could check personal information, change password, update personal information, and share information, comment other status on communication page.

#### Manage employee information

The purpose of manage employee information is to organized employee personal information and working tasks information. The functions of Employee Information Management include:

* Search employee tasks information

Employee allows checking personal working tasks information.

* Mange personal information

Employee, Administrator and team leader are both the users of this system, they can change their personal information such as address, email and login password.

* Add, Update, Delete employee information

Administrator has right add new employee.

Administrator has right update existing employee information.

Administrator has right delete existing employee.

#### Manage customer information

Suplus Technologies has been worked with many customers, organizations and companies, which are very important clients. Record customer information could help Suplus to maintain product and track feedback better. Therefore, Customer Maintenance is a significant function in the whole system. This function could build a steady and safe customer information database, and easy for administrator to search customer’s information.

* Search customer information

Administrator has right to search any customer information.

* Add, Update, Delete customer information.

Administrator has right add new customer.

Administrator has right update existing customer information.

Administrator has right delete existing customer.

#### Manage project information

Since Suplus has been provided technical support to many customers, it requires a functional management method to organize these projects.

* Create new project  
  Administrator has right to create a new project.
* Divide project into tasks  
  Team leader is required to divide a project into many small tasks.
* Assign tasks to employee/team member
* When tasks are created, team leader requires assigning each task to an employee to work on.

#### Login:

The system includes three different users, which are employee, administrator and team leader.

* Login to system
* Log out

***Communicate***:

Login users (employee, administrator and team leader) could share information.

* Post message

Users could post message on communication page.

* Post picture

Users allow posting picture on communication page.

* Comment

Users could comment any message or picture.

### 7.2Use Case Diagram

The following figure presents the use case diagram, which helps to understand how the system functions work.

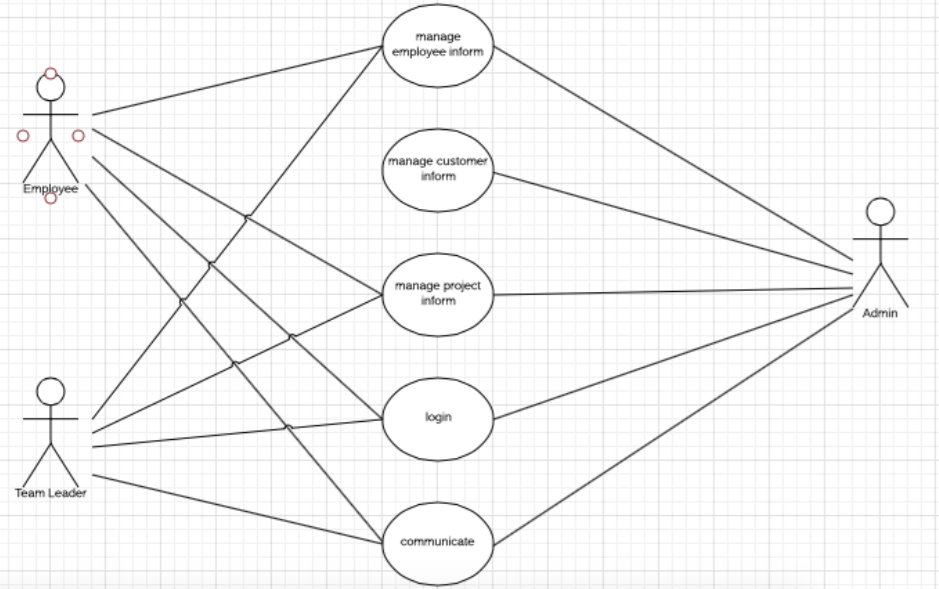


Figure 2 Use Case Diagram

Use Case Description

**Actor: Employee**

Use cases:

1. Login: it makes sure the existing employee could login the system by enter the valid user name and password.
2. Manage project information: After employee login, employee could check personal projects and tasks information
3. Manage employee information: Employee could change their personal information such as password, address, phone number, and email address.
4. Communication: User login as employee has right to post status, share picture, and comment on other users on the communication page.

**Actor: Administrator**

Use cases:

1. Login: When administrator inputs valid user name and password, system should display administrator home page.
2. Manage employee information: Administrator could add new employee, search existing employee information, edit existing employee information, and delete employee information. Administrator could change their personal information such as password, address, phone number, and email address as well.
3. Mange customer information: Administrator is able to create new customer, search existing customer information, edit existing customer information, and delete customer information.
4. Mange project information: Administrator is able to create new project, search existing project information, edit existing project information, and delete project information.
5. Communication: User login as administrator is able to post status, and share picture, and comment on other users on the communication page.

**Actor: Team leader**

Use cases:

1. Login: Team leader could login the system by enter the valid user name and password.
2. Maintain project: Team leader could create tasks to currently project, search existing task information, edit and delete existing task.
3. Manage employee information: Team leader could change their personal information such as password, address, phone number, and email address.
4. Communication: User login as team leader has right to post status, share picture, and comment on other users on the communication page.

### 7.3 Domain Model Diagram

The following domain model diagram demonstrates the association between connectional classes in the Suplus Office Management System.

There are 5 classes, which are employee, admin, team leader, project and task. When the administrator creates a new project, the team leader divides the new project into tasks and passes these tasks to employees. The user as employee has right to check personal project and task information.

In addition, the following figure shows the relationship among these classed. When a customer assigns a contract with company, which means there is a new project. In this diagram, the project as a very important class, and it has a sub child class --task. Meanwhile, there are three roles classes in the diagram, which are employees, the administrator and team leader.

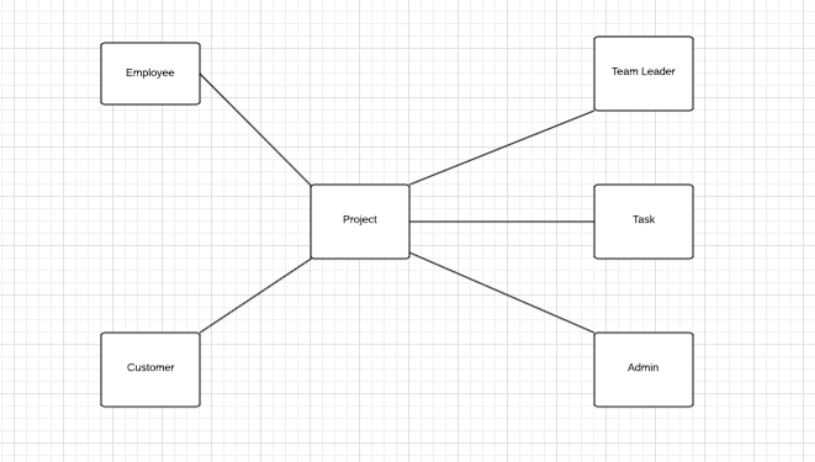


Figure 3 Domain Model Diagram

### 7.4 System Sequence Diagram

The following figures presents sequence diagrams based on use cases diagram.

The first one is sequence diagram for login use case. This diagram represents the login use case for all the users (employee, admin, team leader).

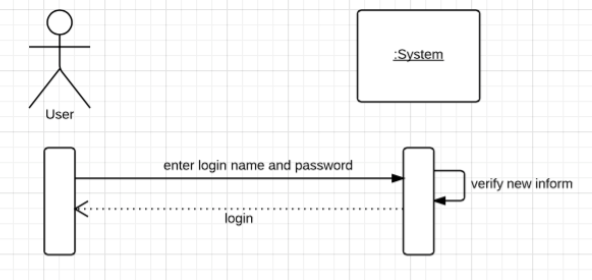
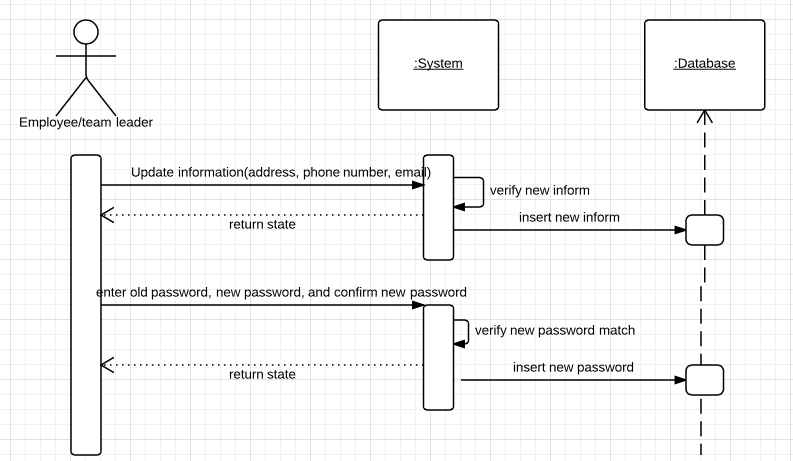
****

Figure 4 login sequence Diagram

Since employee and team leader have similar functions about manage employee information, the next is system sequence diagram for employee and team leader manage employee information.

Figure 5 presents employee and team leader manage employee information sequence diagram.

The following diagram is sequence diagram for administrator managing employee information. Administrator has right to update personal information such as email, address. But most important is could maintain employee information, for example, create new employee, update employee information, and even delete employee information.

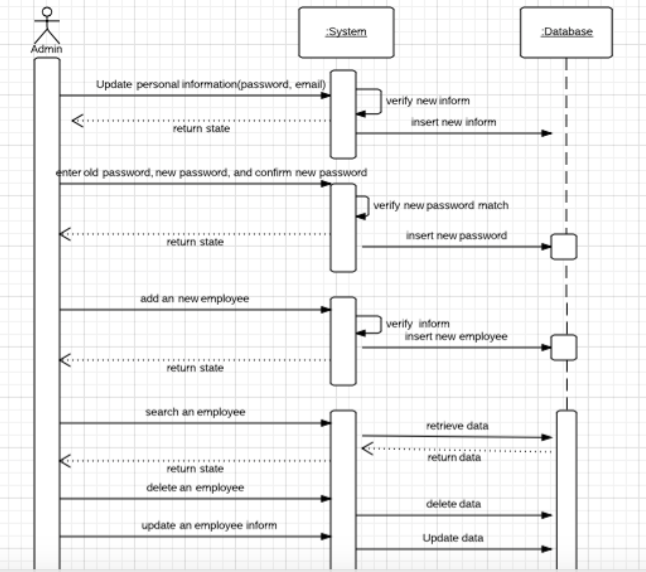


Figure 6 shows admin manage employee information sequence diagram.

The following figure is administrator manage customer information system sequence diagram. Administrator could create new customer. Meanwhile, admin could update existing customer information, and delete customer information as well.

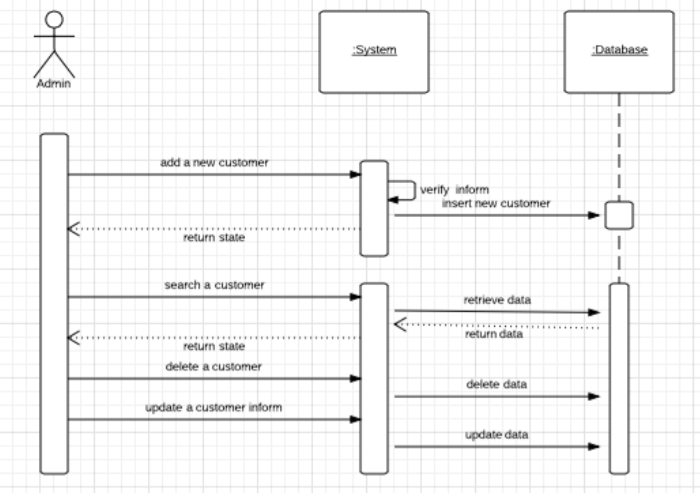


Figure 7 shows admin manage customer information sequence diagram.

There is employee manage project system sequence diagram. Basically, employee has right to search project and task information. After connect to database, webpage should show the searched project and task information.



Figure 8 shows employee manages project sequence diagram.

Admin manage project system sequence diagram:

Admin has right to create new project, search project for detail, update existing project information and delete project.

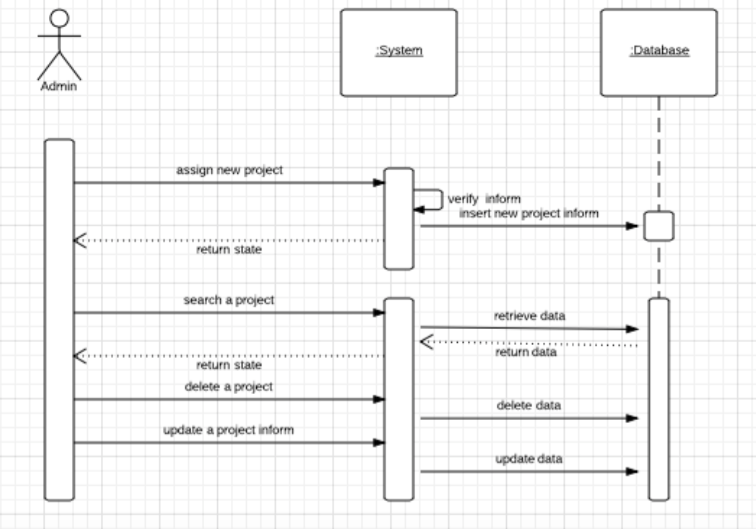


Figure 9 shows admin manage project sequence diagram.

The next one is team leader manage project system sequence diagram. As mentioned above, team leader could divide a project into several small tasks and assign each task to an employee.

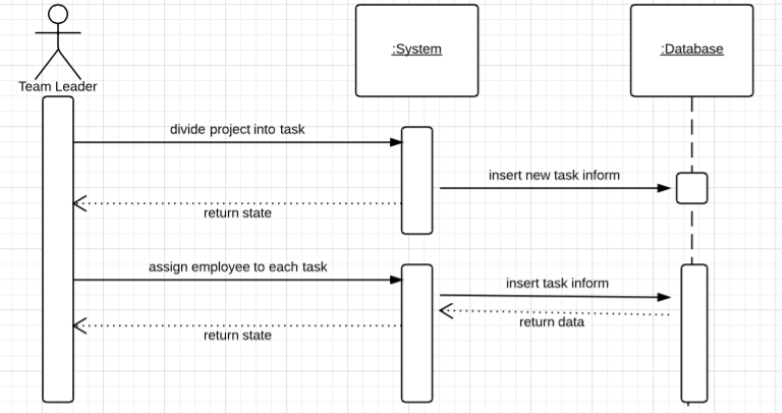


Figure 10 shows team leader manage project sequence diagram

The last system sequence diagram is communicating function for the all users in this system. All the users could post a status, post picture and comment on any status or picture.

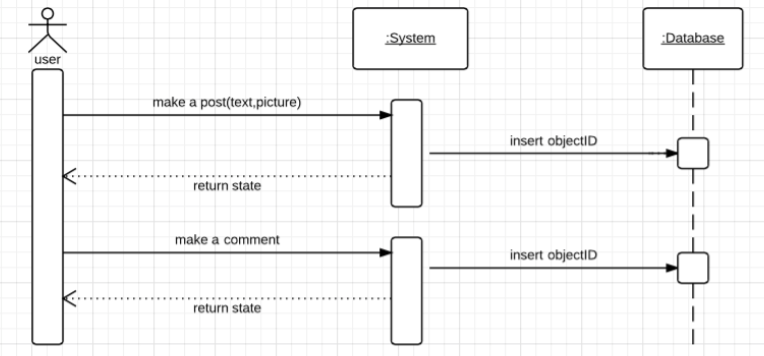


Figure 11 shows communication for the entire users sequence diagram

### 7.5 Software Design Class Diagram

The following diagram is design class diagram of Suplus Office Management System. The system design class diagram is based on domain module class diagram. Generally, each project contains many tasks, so employee working on project and task at the same time. Admin could manage project, employee and customer information. Team leader divide project into tasks and pass them to employees.

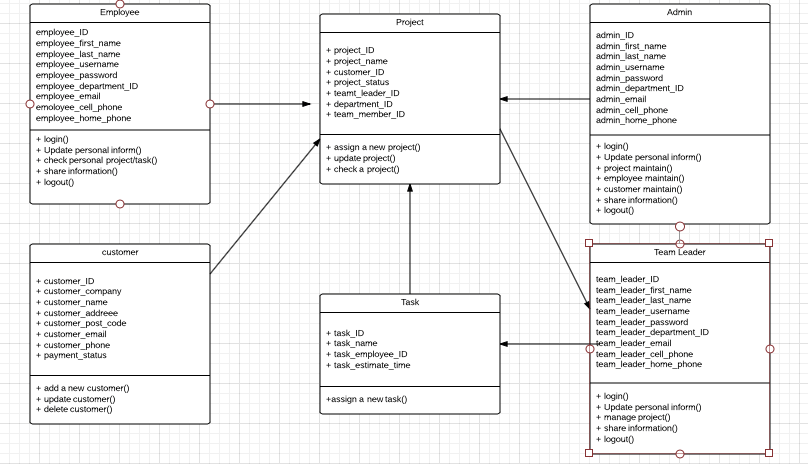


Figure 12 Software Design Class Diagram

### 7.6 Database Design

In this project, there are two database has been adapted, which are MySQL server database and NoSQL Mongo database. MySQL server database helps to manage employee, customer, project, task and department information. Mongo database stores the data from communication page.

### MySQL

MySQL is an open source database system, and inexpensive for developers. Suplus Office Management System is PHP based web system, and it requires compatibility of the chosen database. MySQl support cross platforms, it could run on LINUX, and Windows system. Consider the conditions mentioned above, this system adapted MySQL to manage employees, customers, projects and tasks information.

The following diagram is the relationship diagram of Suplus Office Management System. Database tables include customer, project, task, employee, and department. Employee table includes employee, team leader and admin’s personal information. Customer has customer information. Task is a child table of project.

The design of database is based on domain class diagram, because the classes shown in domain class diagram are significant objects. Suplus Office Management System includes five tables, which are customer, project, task, employee, and department.

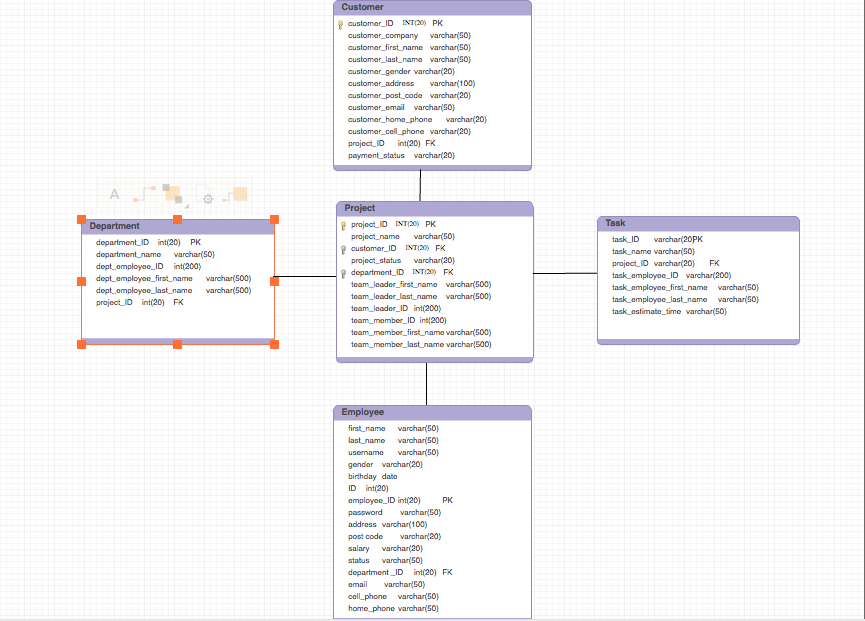


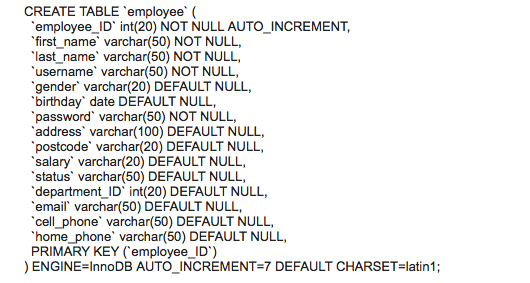
Figure 13 Database E-R Diagram

#### Employee table

Employee table keeps Suplus employee’s information includes employee, administrator, and team leader. Employee table contains

|  |  |  |
| --- | --- | --- |
| Name | Data type | Description |
| First\_Name | varchar | Employee first Name |
| Last\_Name | varchar | Employee last Name |
| Employee\_ID | int | Primary key |
| Gender | varchar | Employee’s gender |
| Birthday | date | Employee’s birthday |
| Address | varchar | Employee’s address |
| Post Code | varchar | Post code |
| Email | varchar | Email address |
| Username | varchar | Login username |
| Password | varchar | Log in password |
| Department\_ID | int | Foreign key |
| Salary | varchar | Employee’s salary |
| Status | Varchar | Single, Married, Divorced |
| Cell\_phone | varchar | Cell phone number |
| Home\_phone | varchar | Home phone number |

Table 1

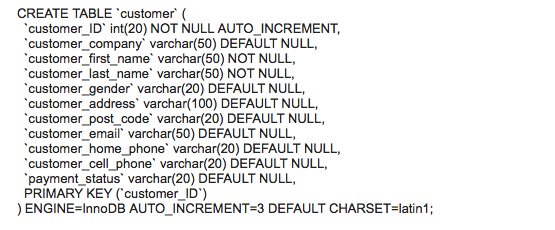


#### Customer table

Customer table includes customer’s information, which are:

|  |  |  |
| --- | --- | --- |
| Name | Data type | Description |
| Customer\_company | varchar | Customer’s company name |
| Customer\_fisrt\_name | varchar | Customer name |
| Customer\_last\_name | varchar | Customer name |
| Customer\_gender | varchar | Gender |
| Customer\_ID | int | Primary key |
| Cusomter\_ home\_phone | varchar | Phone number |
| Cusomter\_ cell\_phone | varchar | Phone number |
| Customer\_address | varchar | Customer’s address |
| Customer\_post\_code | varchar | Customer’s post code |
| Customer\_email | Varchar | Customer’s email address |
| Payment\_status | varchar | Payment asked paid by 3 time, which are when the project begin requires 1/3 of the full payment, when project finished develop before test requires another 1/3, when project finishes requires the last part. |
| Project\_ID | int | Foreign key |

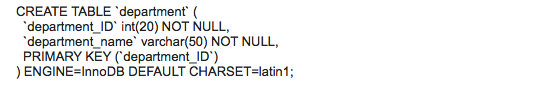
Table 2



#### Department table

Department table has following information:

|  |  |  |
| --- | --- | --- |
| Name | Data type | Description |
| Department\_name | varchar | Name of department |
| Department\_ID | int | Primary key |

Table 3

#### Project table

Project table contains project information.

|  |  |  |
| --- | --- | --- |
| Name | Data type | Description |
| Project\_name | varchar | Name of project |
| Project\_ID | int | Primary key |
| Customer\_ID | int | Foreign key |
| Project\_status | varchar | Beginning, testing, finished |
| Team\_leader\_ID | int | Team leader’s ID |
| Department\_ID | int | Foreign key |

Table 4

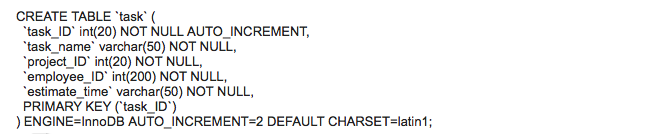
### 

#### Task table

Each project divides into several tasks. Table task stores task information.

|  |  |  |
| --- | --- | --- |
| Name | Data type | Description |
| Task\_name | varchar | Name of task |
| Task\_ID | int | Primary key |
| Project\_ID | int | Foreign key |
| employee­\_ID | int | ID of the employee works on this task |
| estimate\_time | Date | Working Time may costing, calculate by hour |

Table 5

****

### NoSQL

As mentioned before, there is a social media page called communication in this system. Mongo database stored the data came from communication page.

Communication page allow users to post message/picture, and allow users to comment on the post. Each post is considered as a collection, and the post mostly contains same information such as owner, post content, comments. If choosing relational database to store these information, it requires huge amount data mapping and cost more time. However, Mongo DB could fit in data structure without mapping data around, and organized the each data together as a collection. In another word, the each post data of communication page could organize as ID, owner, postedby, reference, image, create date and update date. The following picture shows the data structure.

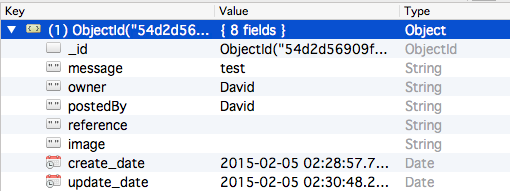


Figure 14

#### Collection

The data has been stored in MongoDB:

ID: As same as relational database, when user save a new data, Mongo database automatically attaches a unique identifier to it.

Owner: This is the person who originally posts the message/picture.

Postedby: This is the user made a comment. It could be same user, because user has right to comment personal posted message.

Reference: When user comments a message/picture, the reference located to the original posted data’s ID.

Image: The image has been inserted.

Created\_date: Date and time of the original post has been creating.

Updated\_date: Date and time of the comment message has been created.

In the following part, screenshots of user testing could explain more of the data design.

#### Post message

In this case, user login as David, so the owner and postedby is both David. David posts a status ”Hello!” in NoSQL database “Hello!” which is been inserted into message for this record. Because this is the original post, the create date and update date are some.



Figure 15

The following picture presents the inserted collection in Mongo database.

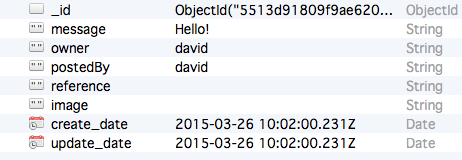


Figure 16

#### Post picture

The following figure is an example of post picture function on communication page.



Figure 17

In this case, user David posts a picture on communication page. The owner and postedby is still David, because he is the person post the picture and this is an original comment. The following picture shows the image has been inserted into Mongo DB after post an image on communication page. In this case, an image has been inserted.

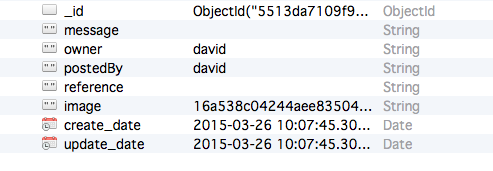


Figure 18

#### Comment

In this case, user David post a message “Good night everyone!” and user Fitz write a comment “Good night! ” under David’s message. 

Figure 19

The next picture demonstrates the original collection in Mongo database. David is the owner of the message and the message postedBy David too. The update\_date is 05:45:55.568Z.

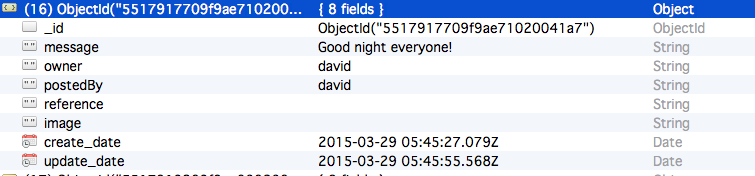


Figure 20

The following picture is the Fitz comment’s collection in Mongo database. Owner is the user who posts the original message, in this case it was David posted the ”Good night everyone! ”.

PostedBy from David change to Fitz, because Fitz commented David’s message.

Update\_date is 05:45:55.566z (figure 21).

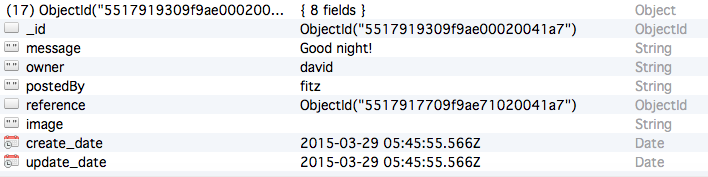
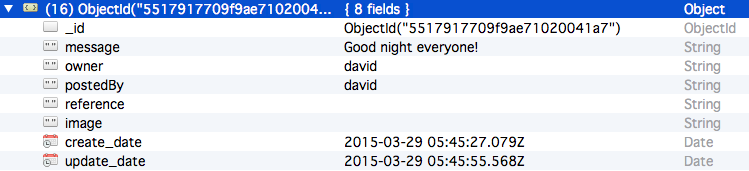


Figure 21

Now move back to David’s original post collection. After Fitz made a commnet on David’s message, update\_date of David’s collction has been changed to Fitz commented time which is same as figure 21 created\_time 05:45:55.566z.

Figure 22

## Access Control

Since Suplus Office Management System is a web application, the security problems should consider as an important part. For example, When administrator login, there are a lots of customer information, which are important values to the company, and the system contains the employee information, project information those are valuable information maybe access by unauthorized users. In order to protect information, Suplus Office Management System builds in role based access control, which could restrict system access to authorize users.

### Role based access control

Role based access control only allows the authorized users to perform the certain operations.

Generally, users are not assigned permissions directly, but their roles could help them to identify themselves and ensure their permissions and duties. Except the all users functions, the system authorizes different functions to different role.

For example, administrator has permission to manage all the employees’ information, customers’ information, and projects’ information. User who login as employee allows checking personal project and tasks information. Users who login as team leader allow managing project.

User requires entering user name and password, which should be match with the stored information in the database. If user tries to login with empty user name and password, the system should warn user with a wrong message. When user login to the system, it is required to enter user name and password. Based on the role access control, user’s interface demonstrates different functions. As the result, role based control access could protect Suplus Office Management System running safely.

## Innovation

As mentioned on system function design, the system includes a social media page called communication. In my design, communication page allows user to post message, picture, and comment the posted information.

This social media page is a PHP based web page and the data has been stored in Mongo DB. In order to improve the design of user interface, adapted Bootstrap as well. For more detail of the Mongo DB store data, please check on 11.1 Why NoSQL.

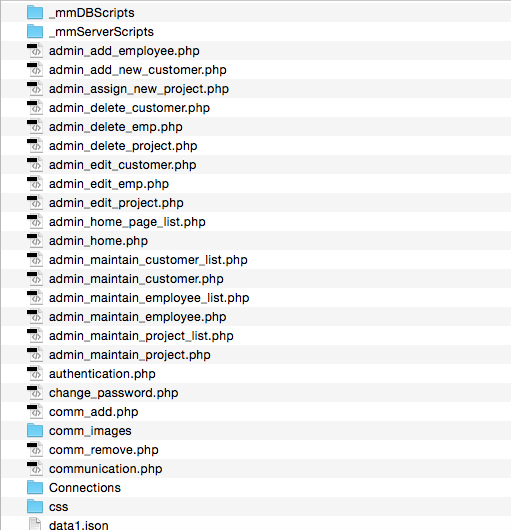
## Development and Implementation

### 10.1 Programming Language Chosen

* PHP
* HTML
* JSP
* jQuery
* CSS
* Bootstrap

### 10.2 System Perspective

The next picture is system design tree map, which includes all web pages has been designed in the system.

****

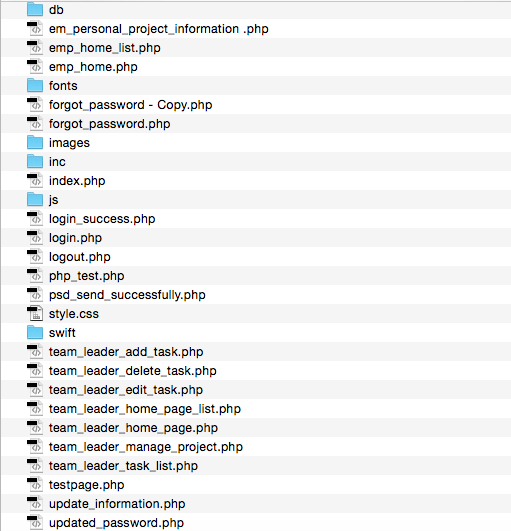
****

Figure 23

#### Employee module

1. Login

When the existing user enters valid user name and password, the employee should login in the system successfully. This is the first page of this manage system, and all the users should start from the login page.

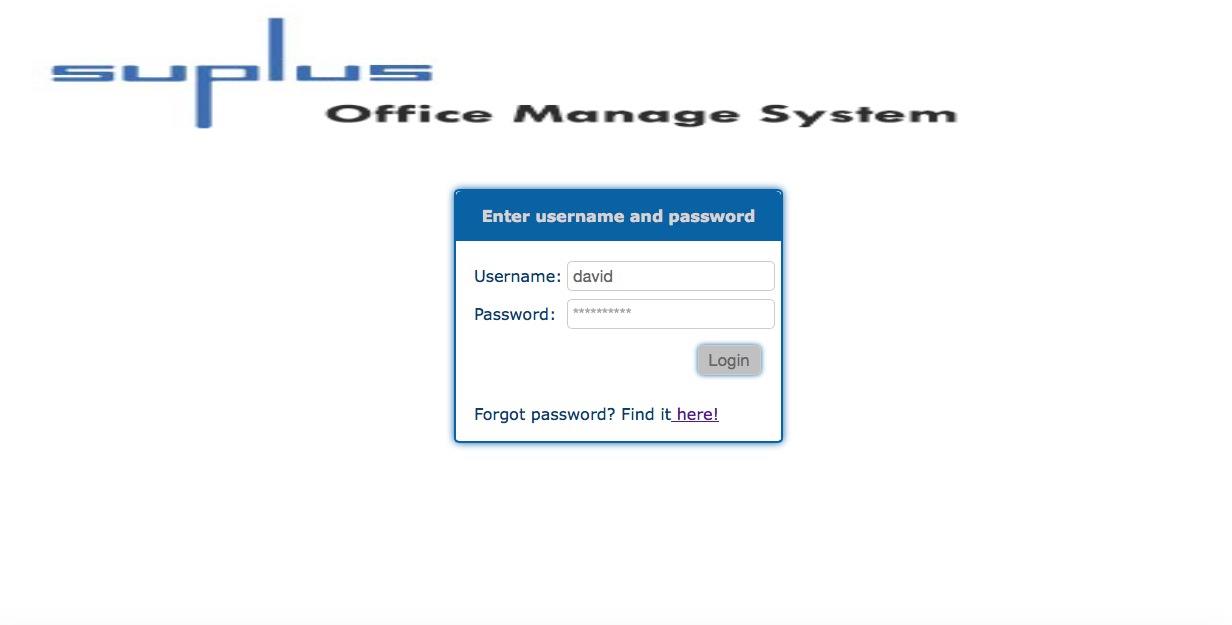


Figure 24

1. Employee home page

After the user login successfully, the system should directly go to the home page of employee, which presents the employee’s functions, and shows all the task information as well. It provides search function to help employee to search task or project information.

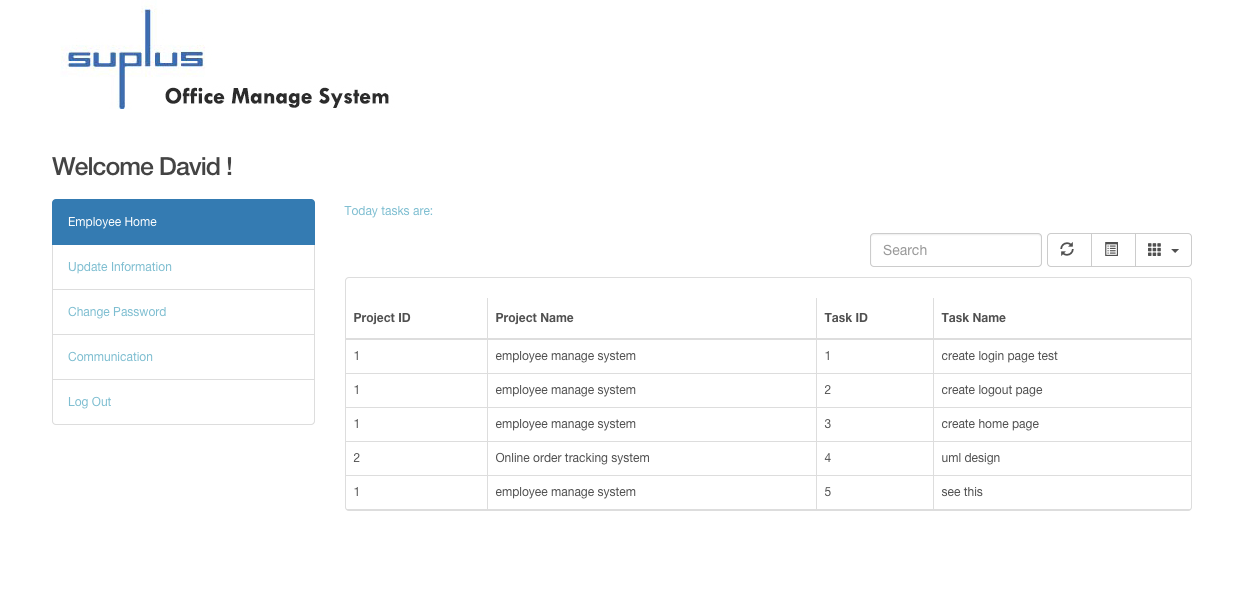


Figure 25

1. Update personal information

After click Update Information from navigation menu, the user could enter the information into the text field, to update personal information.



Figure 26 shows employee update personal information

1. Change password

In addition, the user could change login password. After enter new password and click the update button, the login password should be updated.

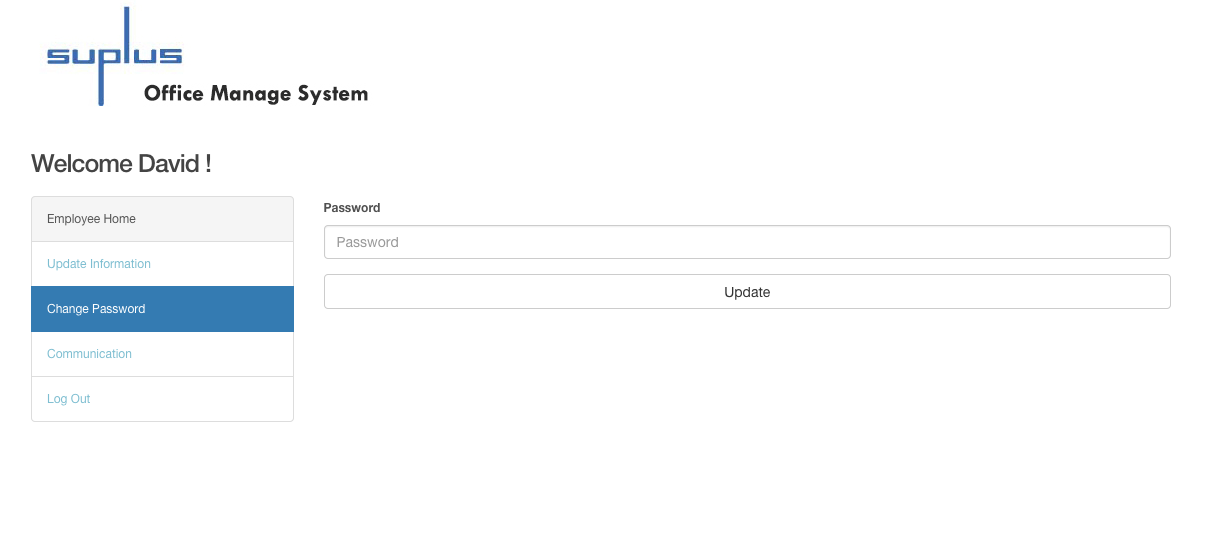


Figure 27 shows change password page

1. Communication

On communication page, the user can post message, share pictures by click post button. At the same time, the user could see other users post as well, and could comment any message or picture.

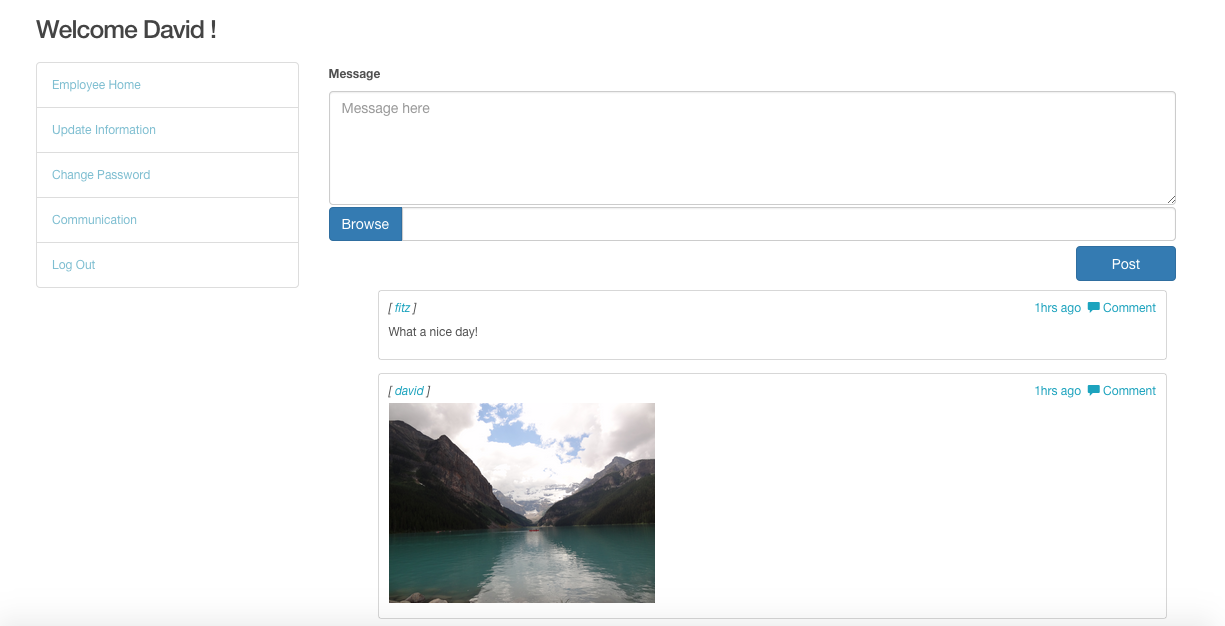


Figure 28 presents Communication page

#### Administrator module

1. Home page

After the user login as administrator, the system redirects to administrator home page. The home page of Admin lists all the projects, team leaders and customers’ information. Also, there is search a menu on the top of this page, could allow the user to search project information.

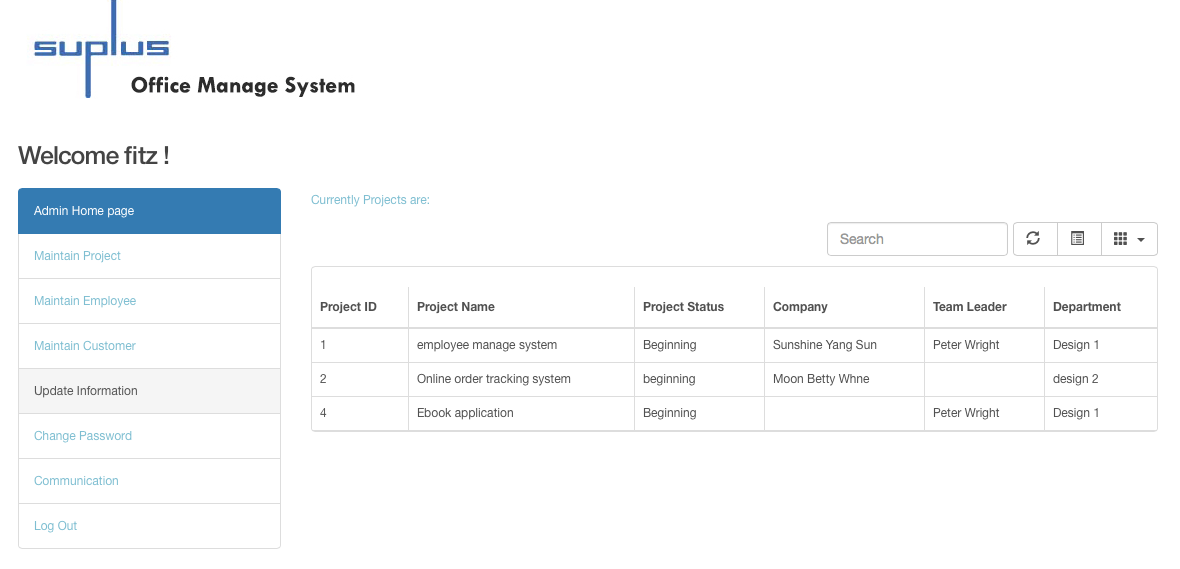


Figure 29 shows admin home page.

1. Maintain employee

Maintain employee function allows the user to search specific employee by enter employee information, create new employee by fill in the information list, edit and delete the existing employee by click the edit and delete button.

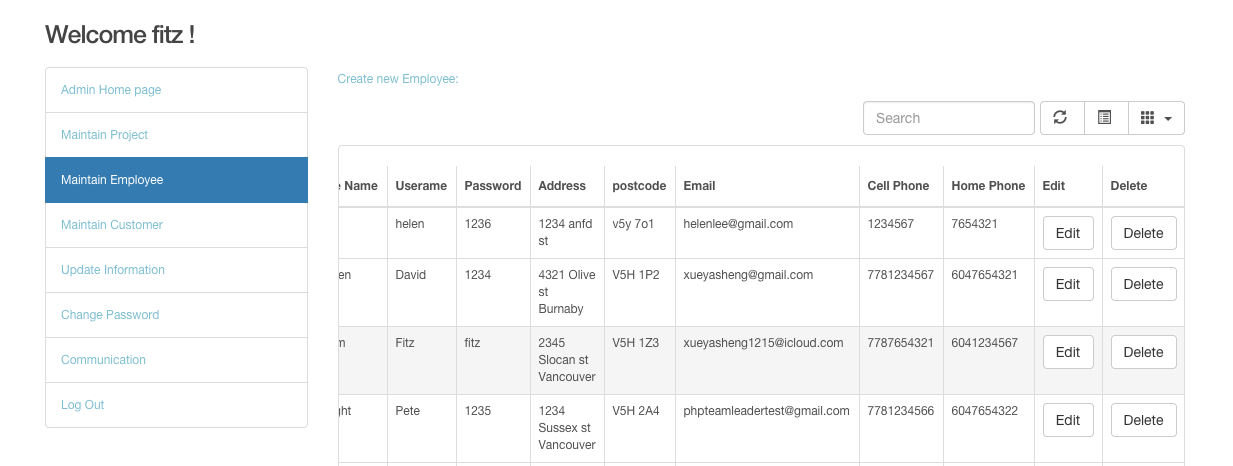


Figure 30 presents maintain employee page.

The following picture show create employee page.

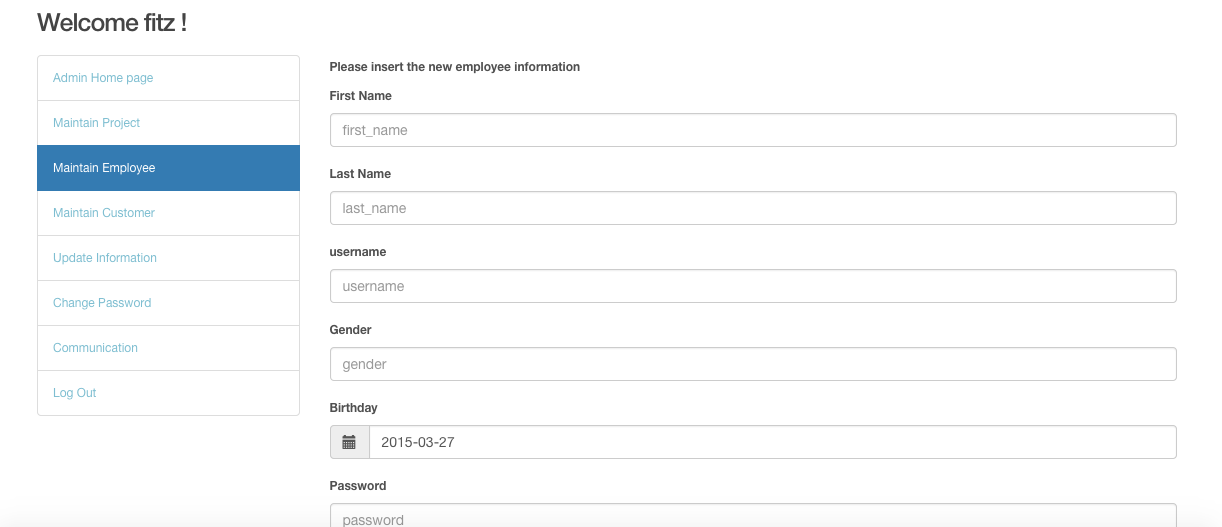


Figure 31

The next figure shows the code of administrator edit employee information.



Figure 32

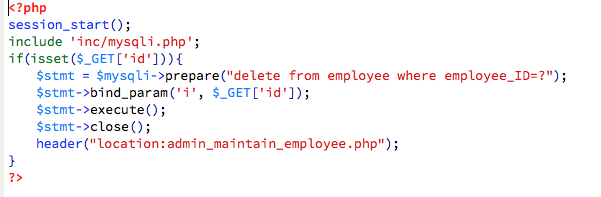
The following figure presents the code of administrator delete employee information. 

Figure 33

1. Maintain Customer

Maintain Customer function allows the user to search customer information, create customer by fill in the list, and update customer information after input the new information in the textboxes and click update information. Meanwhile, administrator could delete the existing customer.

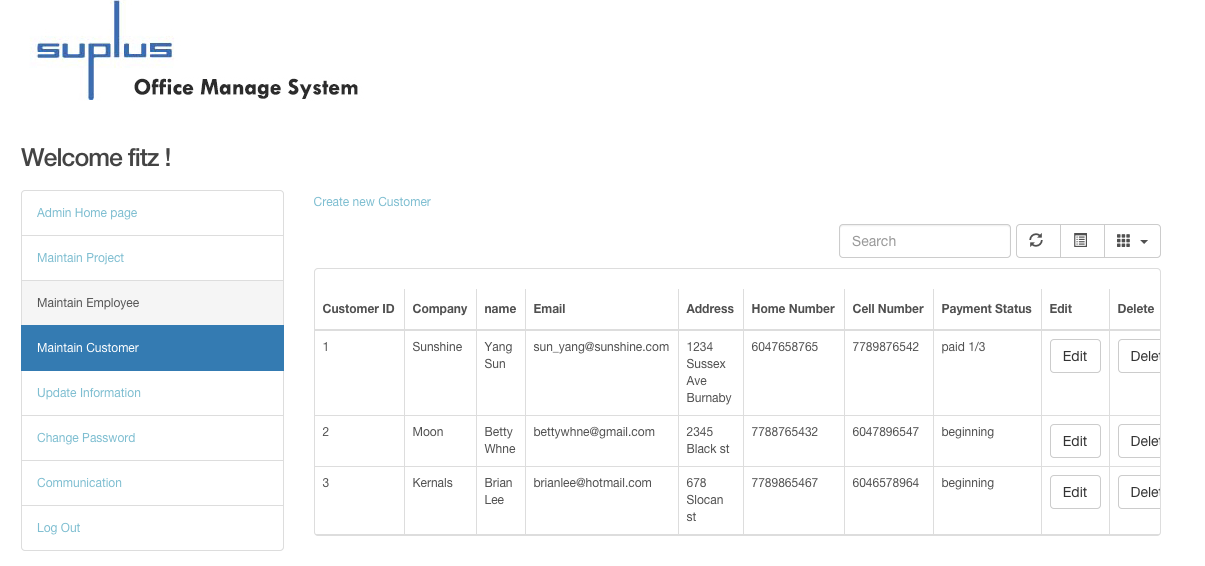


Figure 34 shows maintain customer page.

The next picture is the screenshot of add new customer.

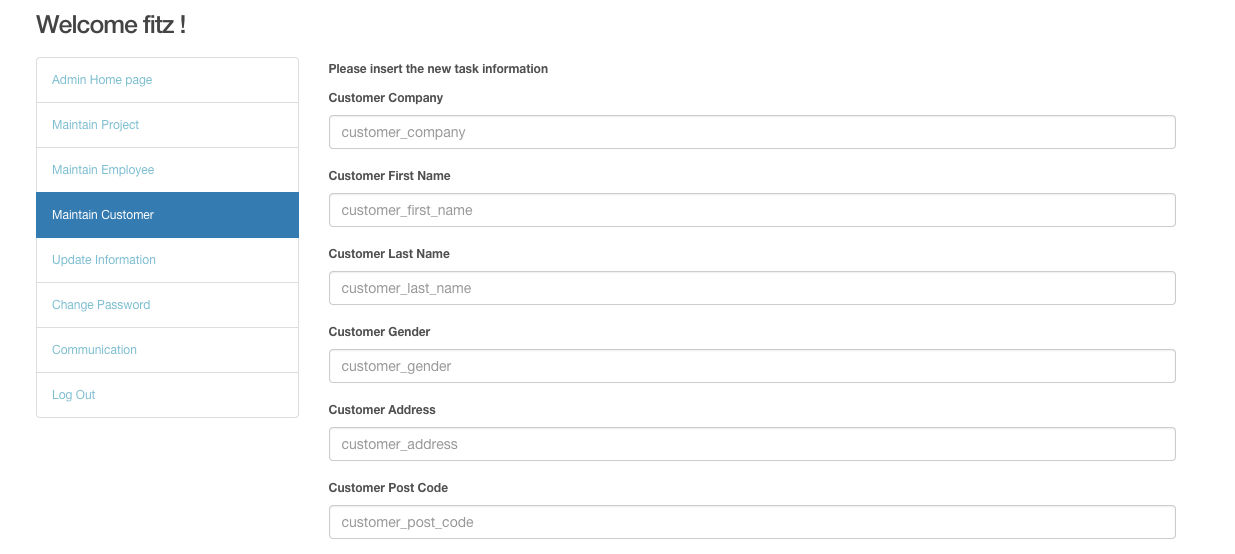


Figure35

The figure 35 presents the code of administrator delete customer information. 

Figure 36

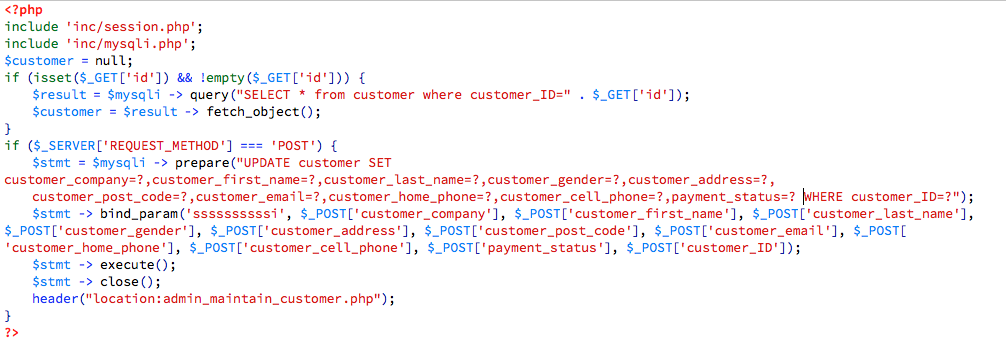
The following figure shows the code of administrator edit customer information. 

Figure 37

1. Maintain Project

Maintain Project function allows the user to search project information, create new project, edit and delete project information.

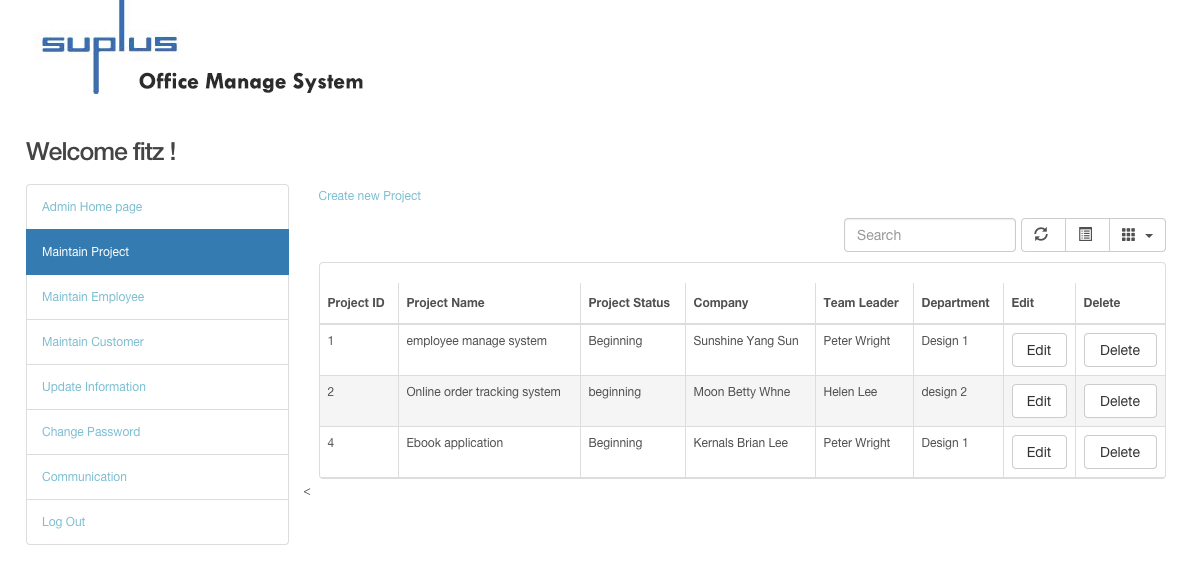


Figure 38 shows administrator maintains project page.

The next picture shows admin add new project page layout.

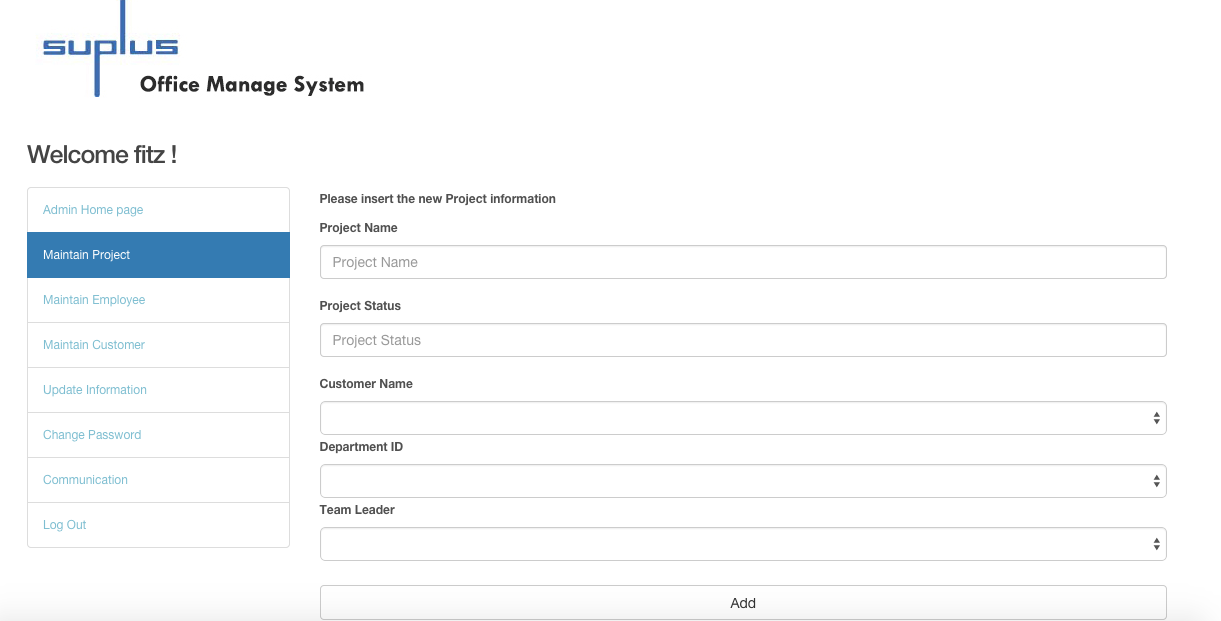


Figure 39

The next figure shows the code of administrator edit project information.

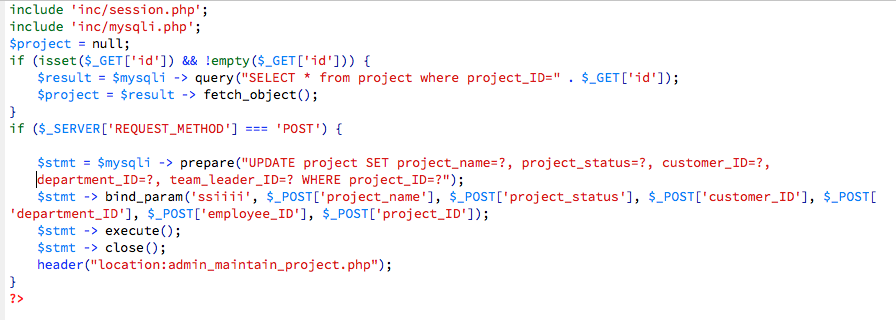


Figure 40

The following figure presets the code of administrator delete project information.



Figure 41

1. Update information

After click Update personal information, the user could enter the information to the textboxes, to update personal information such as address, phone number, email address.

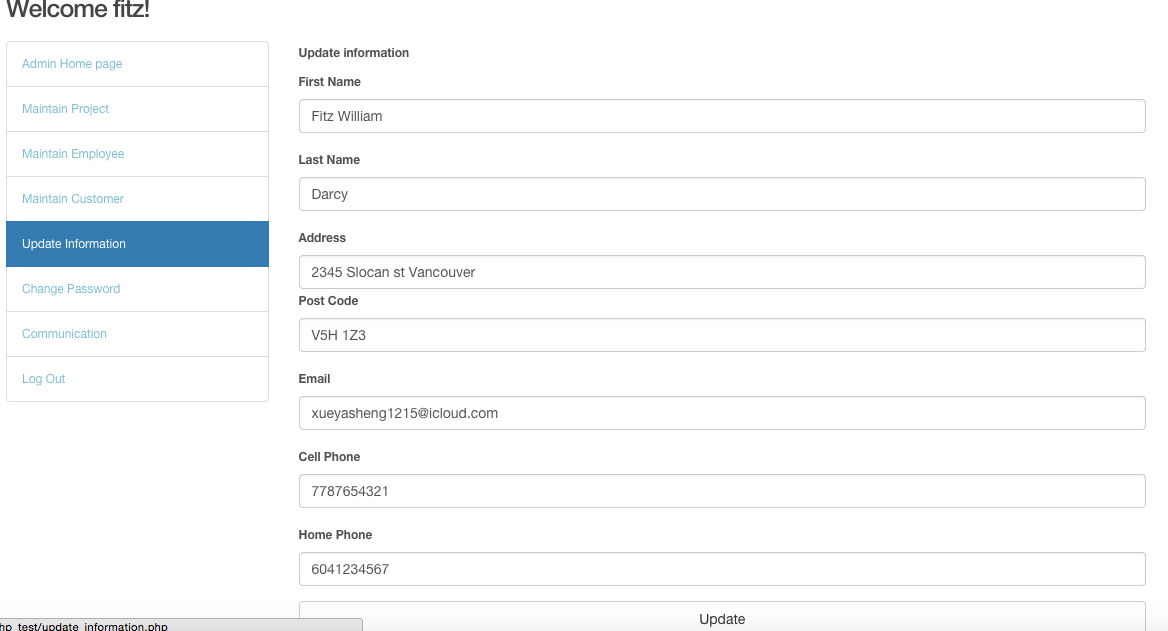


Figure 42

6. Change password

In addition, the user could change login password.

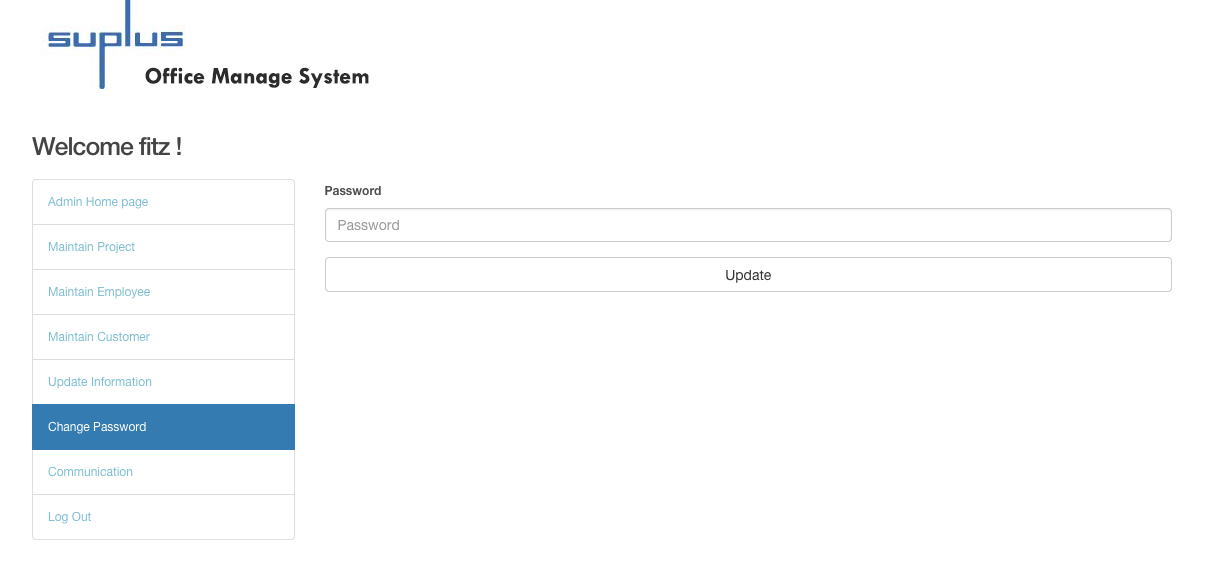


Figure 43 shows update personal information

1. Communication

Administrator as the current user can post messages, share pictures by the post button. At the same time, the user could see other post as well, and could comment other posts.

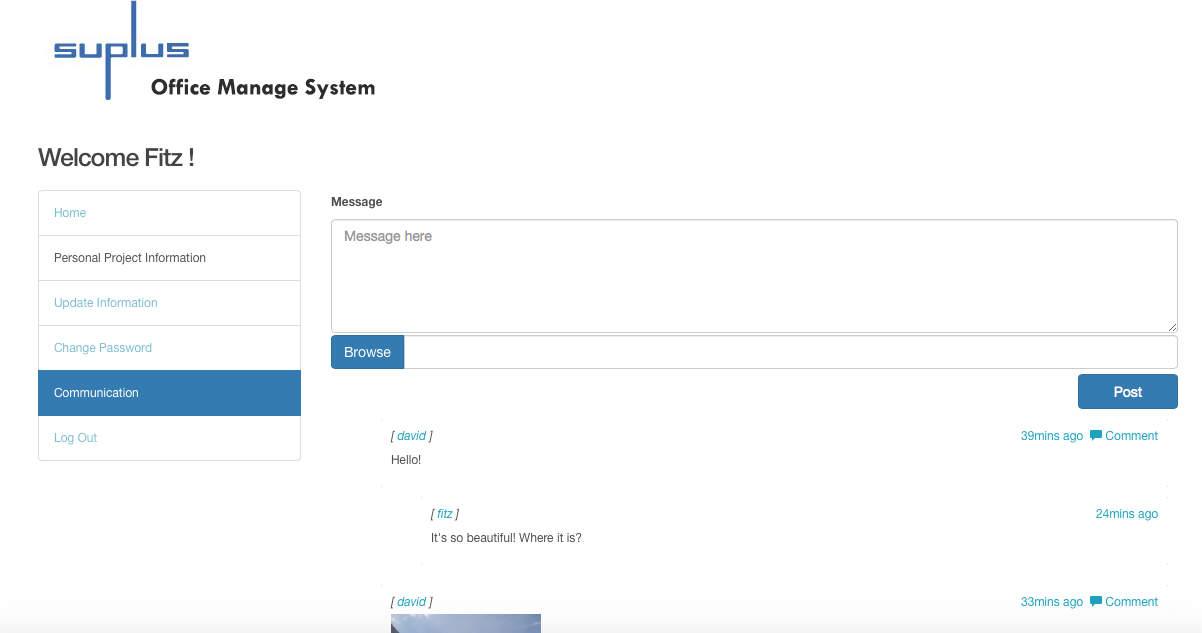


Figure 44

#### Team leader module

1. Home page

When the user login as team leader, the system redirects to team leader home page. The home page lists all working on project and customer information.

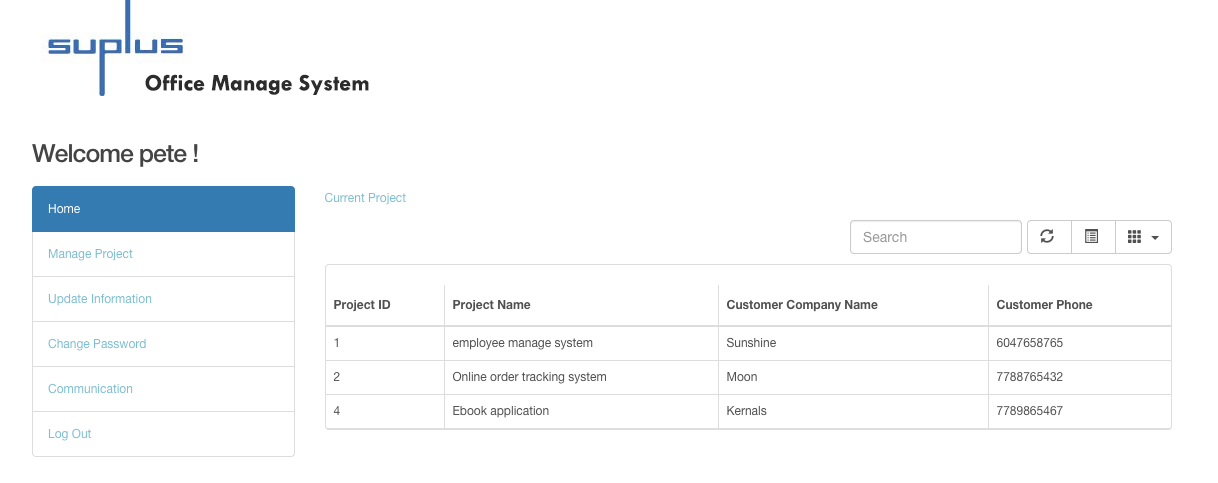


Figure 45 shows team leader home page.

1. Manage project

Manage project page list all the task information. On the top of manage project page, there is a “insert a new task” link, which allows the user to insert new task to a selected project. There is an Edit and Delete button on each task, the user could click to edit or delete the task.

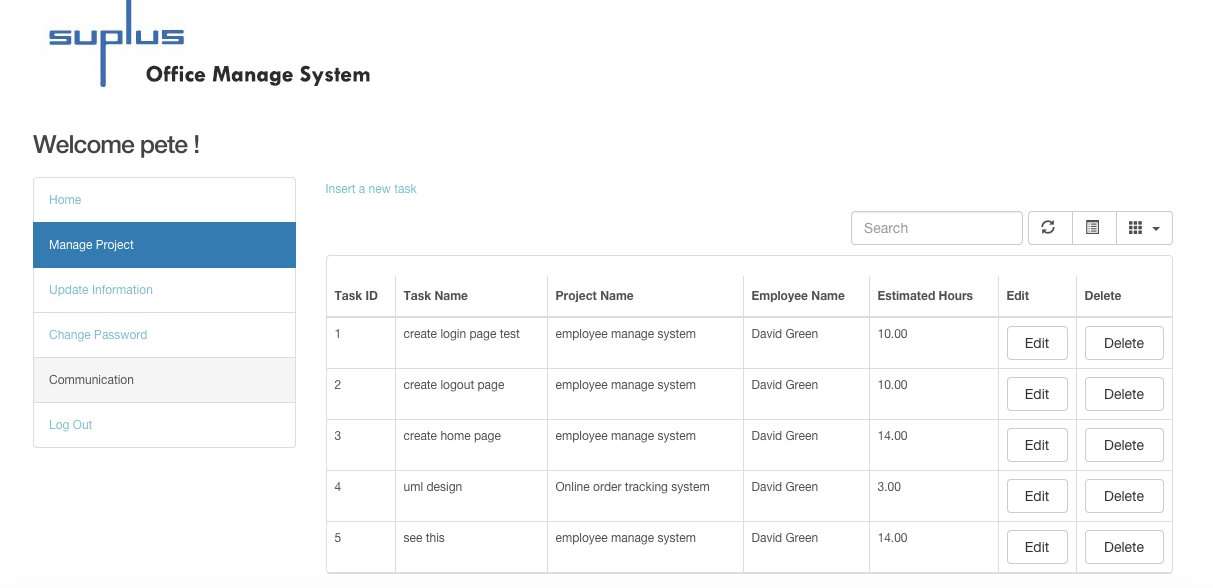


Figure 46

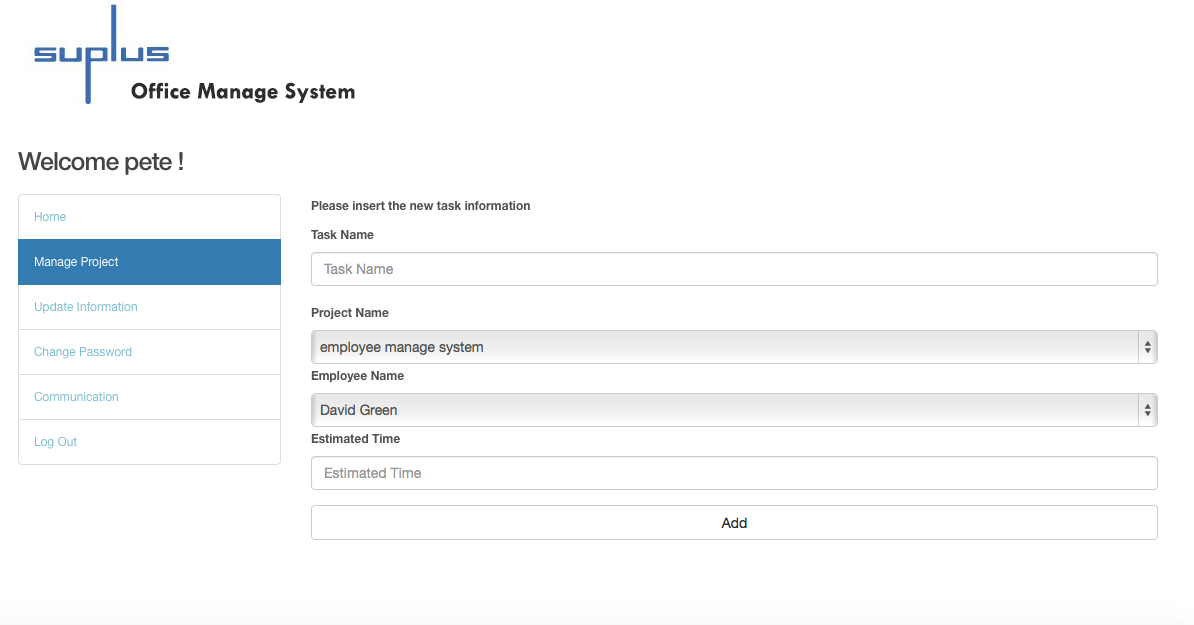
Next picture shows Team leader Insert a new task 

Figure 47

The following picture shows team leader edit the existing task information

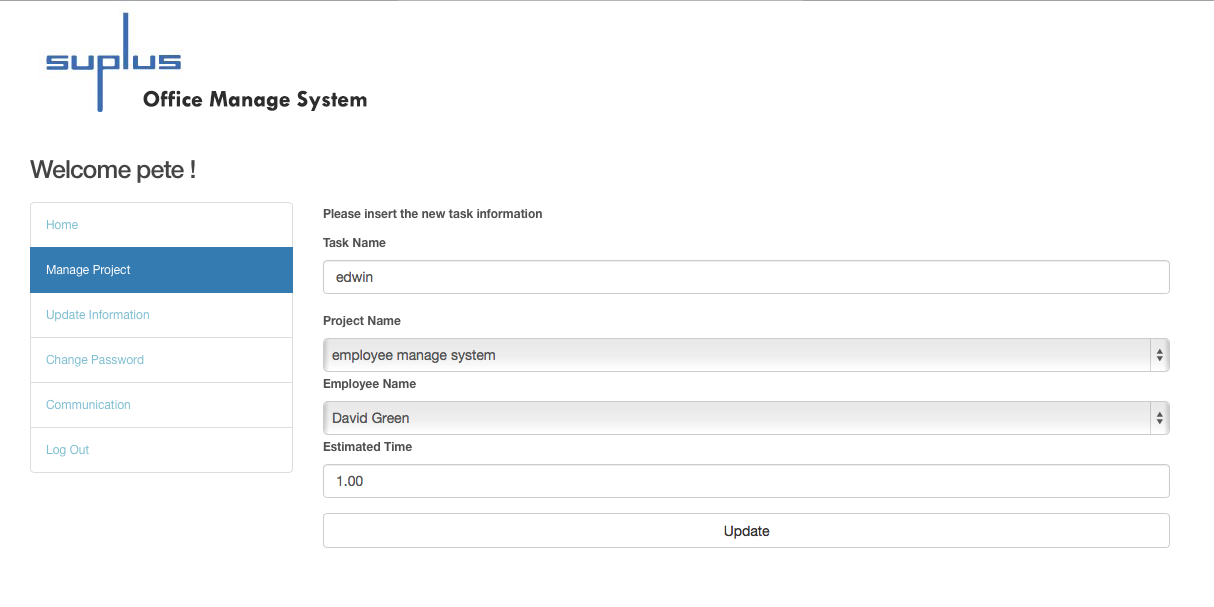


Figure 48

Team leader delete a task:

There is a pop-up window to ask are you sure to delete task?

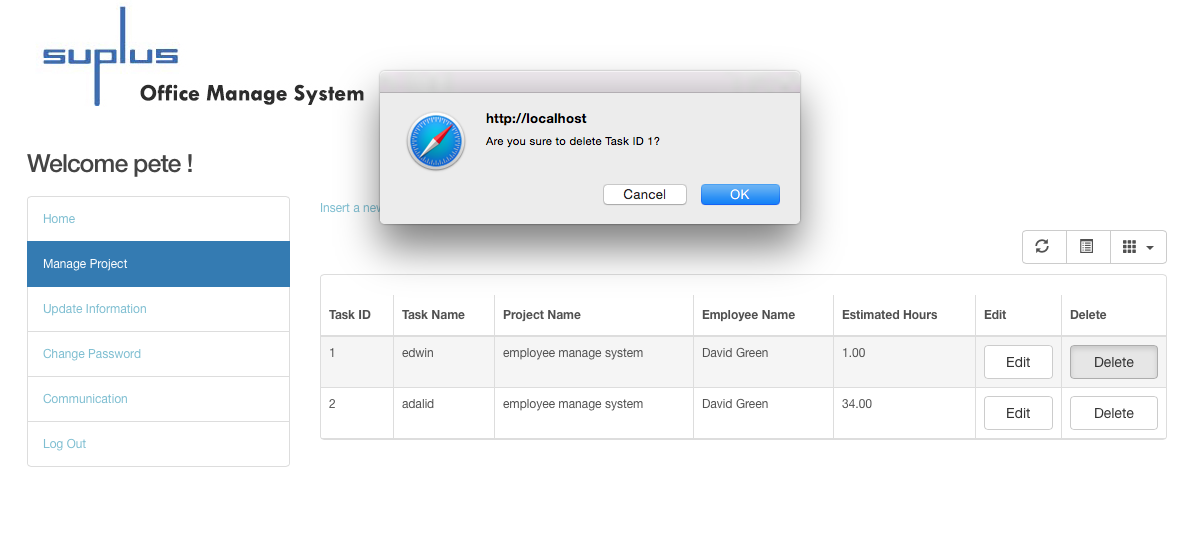


Figure 49

1. Modify personal information

The user could update personal information such as address, phone number, email address.

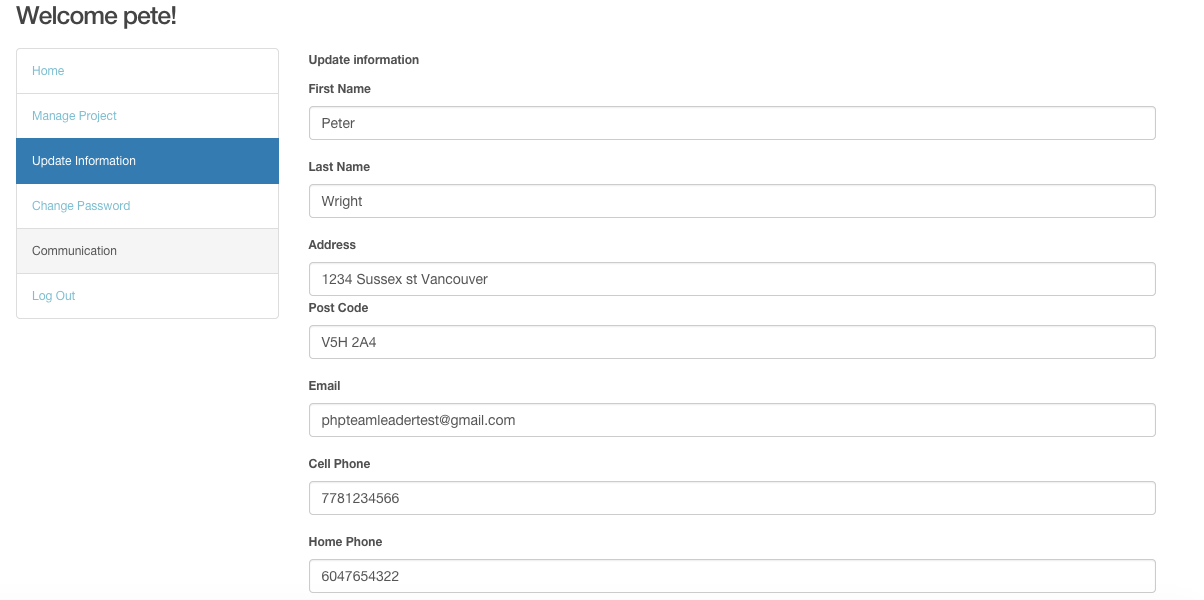


Figure 50 shows administrator update personal information

In addition, the user could change login password.



Figure 51 shows change password

1. Communication

The user could post messages, share pictures by click the post button and comment other’s posts.

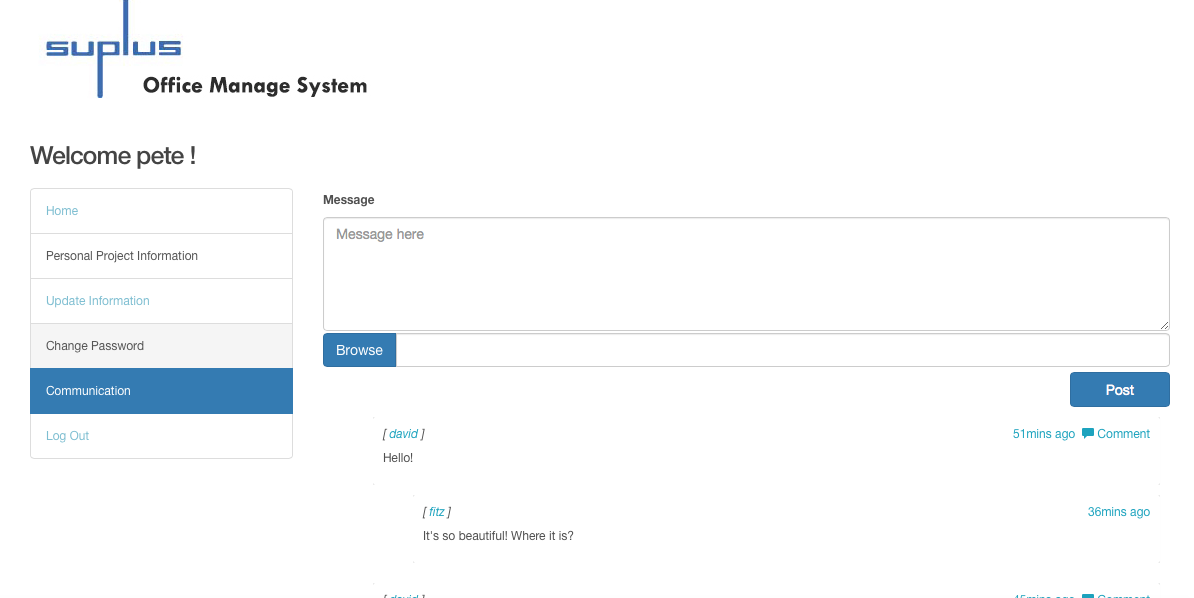


Figure 52

## Research

### 11.1 Why NoSQL?

To explore new technology, this system adapts the NoSQL. Currently, social media websites such as Facebook and Twitter are all conducting NoSQL, which plays a significant role in current database field. Suplus Office Management System adapts Mongo DB to collect the data from communication page, the social media page in this system. The reasons for choosing NoSQL are:

Loading Balance: NoSQL could run over multiple servers, balance the load, and duplicate data to keep the system running in case of hardware failure. As a social website with a huge numbers of data, this system requires a steady database to operate the system.

File storage: NoSQL contains many different types of file storage, such as key-value stores, document database, wide-column stores, and graph database.

The large number of users combined with the dynamic nature of usage patterns is driving the need for more easily scalable database technology. With relational technologies, it is difficult or even impossible to get the dynamic scalability and maintain the performance as users demand. This is the most significant reason for choosing NoSQL when creating communication page as a social media web page (Sadalage & Fowler, 2013).

### 11.2 Advantage & Disadvantage

Compared to NoSQL, relational database is a set of tables containing data fitted into existing categories. However, NoSQL usually handles unstructured and non-defined data. Since it is a very new and young technology, NoSQL still has a few challenges. The next section of the paper outlines the advantages and disadvantages of NoSQL.

**Advantages:**

Processing data is faster than relational database. Usually, relational database requires the same set of ACID (atomicity, consistency, insolation, durability) restraints, but it could decrease the processing speed. But NoSQL does not support ACID, so it increases performances. Besides, NoSQL data model is simpler than relational database. These are the reasons that show NoSQL performs faster than relational database.

Document-based store. NoSQL database stores and organizes data as collections of documents. As mentioned before, the user can customize the document rather than using structured and predefined information in relational database.

Sharding and load balancing. Sharding could store data records across multiple machines when the data requires a huge amount of growth in the future. NoSQL (especially MongoDb) could solve this problem with horizontal scaling, and add more machines to support data growth (would be mentioned on next part).

**Disadvantages:**

1. Complex consistency. Since NoSQL database does not follow the ACID transactions, it may have problem with consistency. Even though inconsistency could improve the performance and scalability, it could cause application and transaction problems, especially in banking management.

2. No joints. NoSQL database does not include joints like relational database does. When the data requires a joint function, it may make multiple quires to join the data manually. It may possibly reduce performance.

3. Absence of standardization. Because NoSQL does not have standard APIs or query language, it could cost more when migration a solution from differ vendors (Leavitt, N, 2010).

Although NoSQL is not a mature technology, it allows data stored over different servers or clusters. It is perfectly for social media page. Therefore, NoSQL technology provides a new environment to develop data and system.

### 11.3 Why Mongo DB?

The data storage model of NoSQL includes document database, and graph database.

Graph database has many advantages on operating graph data, however, it is not a good choice for communication page. Although the system users are able to post information includes picture, the user could post text message as well. Therefore, this system could not use graph database.

For explore and practice new technology and complete the innovation part, this system adapted Mongo DB. This is the most important reason to choosing Mongo DB. Mongo DB is a better solution to this system because Mongo DB is a document-based database. The data came from communication page mostly are unstructured data, therefore, Mongo DB would be a better choice for the system. Mongo DB stores and organize data as collections of documents, which could provide better data model to fit in data structure than relational database. In another word, Mongo DB can save more time for mapping data between data structure than relational database. The post came from communication page could be a simple message, a picture or even an article, which are all collections and are stored in Mongo DB. Therefore, using Mongo DB to store posting data is more efficient.

Another reason to choose Mongo DB is scalbilty. Communication page as a social media web page contains a lot of posting data sets. In this situation, the high query rates and large data set could exceed the storage capacity. However, Mongo DB could provide vertical scaling and sharding to solve this problem.

Vertical scaling approach could add more CUP and storage resources to increase capacity, but it costs more money. Another approach sharding divides the data set and distributes the data to multiple servers or other shards. Shard is an independent database. When user inserts data, application only requires accessing the shard responsible for that record (Francia, 2012). Shard could fix the server capacity problem, and reduce the development budget. Therefore, considering the data type store and scaling, Mongo DB is the best solution for this system.

## Test

I designed 33 tests are based on the key system functions. During the testing, I have completed two categories of tests, which are user testing and system functions testing.

### 12.1 User Test Plan

At first, I tested the employee, administrator, and team leader’s functions based on the test cases I created. Secondly, I invited other three general testers who are not working in computing area and an employee, an administrator and a team leader who are working at Suplus Technology to test the system as well. They were all required to follow test cases I created, and provide feedback for debugging.

The testers also are required to fill in the survey list that could help designer to improve the system. For the detail of the survey, please check Appendix C.

Testers were required to time the duration of tested operation according to the test cases. If the testers take a long time to go through each module, which means this system is hard to follow or understand, and the system is required to fixed.

Scale: The functions are workable or not. The hours the testers spent to operate system according to the test cases.

Test: The tests should under the Windows and Linux system. The testers were invited to test the system functions of a single module (there are three modules: employee, administrator, and team leader). They are all given the existing user name and password to login the system. The duration of operation is timed according to test cases.

Test scenario: The testers have been grouped to test different module. Each module includes two testers. One of them is the working for Suplus Technology, and the other one is a general tester has been invited. The testers were required login the system, and went through all the functions based on the test cases.

Worst case: The testers found functions unworkable. The testers spent more than an hour to complete the entire operation.

Best case: All the functions are workable. The testers spent less than a half hour to complete the entire operation.

Planned: The testers took 20-35 minutes to finish the whole tests. And the system functions are all workable.

#### Result of user testing

For the testers’ survey, please check comment at the back (Appendix C).

**A. All Users Testing**

All of the testers were required to test all the following functions after login in to the system. Half of testers are working for Suplus Technology.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test NO. | Testing | Expected Outcomes | Result | | | | Tester |
|  | Test that the login textboxes could be filled information on the login page. | The textboxes of username and password could enter information. | Passed | | | | \*E  \*A  \*T  \*G |
|  | Test that an error message presents to indicate user or invalid data or empty blank. | * Enter not existing username and password, could not login to system. * Emptied username and password could not login to system. | **Passed** | | | E  A  T  G | |
|  | Test the existing user could login to system by enter the valid username and password. | Input valid username and password, which could be found in database. The user could login to system. | **Passed** | | E  A  T  G | | |
|  | Test that the system could send email with password to binding email address when user clicks the forgot password button on login page. | An email with password sent. | **Passed** | E  A  T  G | | | |
|  | Test that the user could edit personal information on update information page. | The textbox could type in data. And the information has been updated. | **Passed** | E  A  T  G | | | |
|  | Test that the user could change password once have signed in the system. | The user could enter new password on change password page.  After logout, the user could login system by new password. | **Passed** | E  A  T  G | | | |
|  | Test that the user could post message on communication page. | The user login with valid username and password could post message on communication page. | **Passed** | E  A  T  G | | | |
|  | Test that the user could post a picture on communication page. | The user could select a picture from local directory, and the picture could be posted. | **Passed** | E  A  T  G | | | |
|  | Test that the user could post comment on communication page. | The user has posted a comment. | **Passed** | E  A  T  G | | | |

E is an employee working for Suplus Technology. A is the administrator of Suplus, T is the team leader of Suplus. G is the tester has been invited.

**B. Employee Testing**

The testers were required to test all following functions after login in as employees. One of the testers is an employee of Suplus Technology.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test NO. | Testing | Excepted Outcomes | Result | Testers | |
|  | Test that the search textbox on employee home page allow user input information. | Information could be typed into the search textbox. | **Passed** | E  G | |
|  | Test the search function on employee home page. | The data could be found by filling in following search conditions: task ID, task name and project ID. The data could be found. | **Passed** | | E  G |

E is an employee working for Suplus Technology. G is the tester has been invited.

**C. Administrator Testing**

The testers were required to test all the following functions after login in as administrators. One of the testers is an administrator of Suplus Technology.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test NO. | Testing | Excepted Outcomes | Result | Testers | |
|  | Test that the administrator search function on admin home page. | The data could be found by filling in following search conditions: project ID, project name, customer ID. | Passed | A  G | |
|  | Test that the administrator could search employee information on maintain employee page. | The data could be found by filling in following search conditions: employee ID, and employee name. | **Passed** | | A  G |
|  | Test that the user edit employee information function on maintain employee page. | The system could redirect to edit information page, and textbox allows filling in data. The data could be updated. | **Passed** | | A  G |
|  | Test that the “Create new employee” link on maintain employee page. | The system could redirect to create new employee page. | **Passed** | | A  G |
|  | Test that the textbox could insert information on create new employee page. | The textbox could be inserted on create new employee page. | **Passed** | | A  G |
|  | Test that the administrator create new employee function on maintain employee page. | The new employee information has been added. | **Passed** | | A  G |
|  | Test that the administrator delete employee function on maintain employee page. | The employee information has been deleted. | **Passed** | | A  G |
|  | Test that the administrator create new customer function on maintain customer page. | The new customer information has been created. | **Passed** | | A  G |
|  | Test that the user edit customer information function on maintain customer page. | The customer information could be updated. | **Passed** | | A  G |
|  | Test that the user search customer function on maintain customer page. | The data could be found by filling in following search conditions: customer ID, and customer name. | **Passed** | | A  G |
|  | Test that the user could delete customer information on maintain customer page. | The customer information has been deleted. | **Passed** | | A  G |
|  | Test that the select list for department information can be chosen on create new project page. | The select list for department information could allow user to choose the specific department. | **Passed** | | A  G |
|  | Test that the user could create new project. | The new project could be created. | **Passed** | | A  G |
|  | Test that the search project function on maintain project page. | The data could be found by filling in following search conditions: project ID, project name, and customer project status. | **Passed** | | A  G |
|  | Test that the user edit function on maintain project page. | The project information could be updated. | **Passed** | | A  G |
|  | Test that the user could delete project information on maintain project page. | The project information has been deleted. | **Passed** | | A  G |

A is an administrator working for Suplus Technology. G is a tester has been invited.

**D. Team leader Testing**

The testers were required to test all the following functions after login in as team leader. One of the testers is a team leader of Suplus Technology.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test NO. | Testing | Excepted Outcomes | Result | Testers | |
|  | Test that the team leader could search project information on manage project page. | The data could be found by filling in following search conditions: project ID, project name. | Passed | | T  G |
|  | Test that the textbox allow user to enter information and the select list allow user to choose project and employeeon create new task page. | The textbox could be inserted, and project name and employee name select list contain the project information and employee name. | Passed | | T  G |
|  | Test that user could insert new task information on insert new task page. | The new task has been inserted. | Passed | | T  G |
|  | Test that the user could search task information on manage project page. | The data has been found. | Passed | | T  G |
|  | Test that the edit task function allows user to update existing task information. | The task information could be updated. | Passed | | T  G |
|  | Test that the delete task function allows user to delete existing task information on manage project page. | User chooses the information required deleting, and click delete button.  Click ok on the pop-up window that asks are you sure to delete?  Data should be deleted. | Passed | | T  G |

T is the team leader of Suplus Technology. G is the tester has been invited.

### Debugging

During the user testing, it was discovered administrator unable to create a new project, because code was counting the number of records (the number of projects) and prepared to insert new project. However, in database, auto increment is enabled for project\_ID, and the project\_ID is supposed to be created automatically. Therefore, the code of counting projects was commented. The code was changed From Figure A to Figure B.

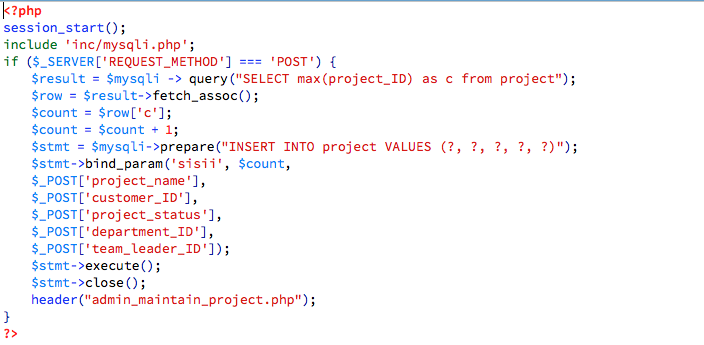


Figure A-1st version of create new project

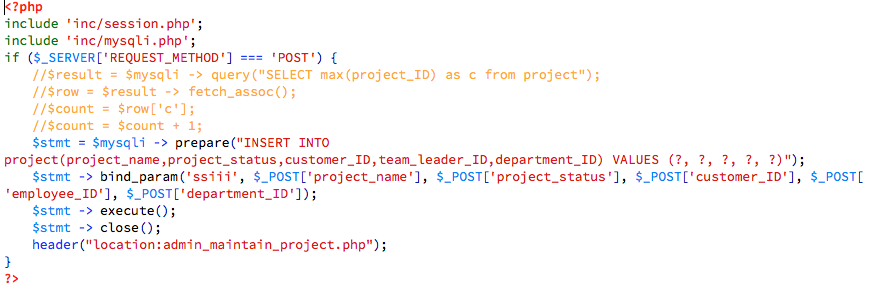


Figure 2- 2nd version of create new project

After changing the code, I have asked the testers of administrator module to test create new project function again. As the result, the project could be created.

In addition, some testers mentioned user interface was not user friendly. To fix this problem, I adapted Bootstrap and JQuery to redesigned the system, such as changing table style, navigation menu, and search menu.

### 12.2 System Function Test

The system function testing of this program consisted of manual testing, and I am the only tester. In the manual test, all the functionality had to be tested. For the detail of test cases, please check Appendix B.

#### Result of system function testing

Based on function design, Suplus Office Management System has been divided into three parts, which are employee module, administrator module, and team leader module. Therefore, this system has been tested by employee module, administrator module, and team leader module.

#### All users test cases

Some basic functions of employee, administrator, and team leader are same, such as login, update personal information, and change password, and post message. Therefore, in this document, some test cases are combined together in following table.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No | Testing | Expected Outcome | Actual Outcome |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Test that the login textboxes could be filled information on the login page. | The textboxes of username and password could enter information. | Passed |
|  | Test that an error message presents to indicate user or invalid data or empty blank. | * Enter not existing username and password, could not login to system. * Emptied username and password could not login to system. | **Passed** |
|  | Test the existing user could login to system by enter the valid username and password. | Input valid username and password, which could be found in database. The user could login to system. | **Passed** |
|  | Test that the system could send email with password to binding email address when user clicks the forgot password button on login page. | An email with password sent. | **Passed** |
|  | Test that the user could edit personal information on update information page. | The textbox could type in data. And the information has been updated. | **Passed** |
|  | Test that the user could change password once have signed in the system. | The user could enter new password on change password page.  After logout, the user could login system by new password. | **Passed** |
|  | Test that the user could post message on communication page. | The user login with valid username and password could post message on communication page. | **Passed** |
|  | Test that the user could post a picture on communication page. | The user could select a picture from local directory, and the picture could be posted. | **Passed** |
|  | Test that the user could post comment on communication page. | The user has posted a comment. | **Passed** |

#### Employee module test cases

In this case, tester should login as employee, and test all the employee functions listed below.

|  |  |  |  |
| --- | --- | --- | --- |
| Test NO. | Testing | Excepted Outcome | Actual Outcome |
|  | Test that the search textbox on employee home page allow user input information. | Information could be typed into the search textbox. | **Passed** |
|  | Test the search function on employee home page. | The data could be found by filling in following search conditions: task ID, task name and project ID. The data could be found. | **Passed** |

#### Admin module test cases

In the following part, tester should login as administrator, and test the following listed functions.

|  |  |  |  |
| --- | --- | --- | --- |
| Test NO. | Testing | Excepted Outcome | Actual Outcome |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Test that the administrator search function on admin home page. | The data could be found by filling in following search conditions: project ID, project name, customer ID. | Passed |
|  | Test that the administrator could search employee information on maintain employee page. | The data could be found by filling in following search conditions: employee ID, and employee name. | **Passed** |
|  | Test that the user edit employee information function on maintain employee page. | The system could redirect to edit information page, and textbox allows filling in data. The data could be updated. | **Passed** |
|  | Test that the “Create new employee” link on maintain employee page. | The system could redirect to create new employee page. | **Passed** |
|  | Test that the textbox could insert information on create new employee page. | The textbox could be inserted on create new employee page. | **Passed** |
|  | Test that the administrator create new employee function on maintain employee page. | The new employee information has been added. | **Passed** |
|  | Test that the administrator delete employee function on maintain employee page. | The employee information has been deleted. | **Passed** |
|  | Test that the administrator create new customer function on maintain customer page. | The new customer information has been created. | **Passed** |
|  | Test that the user edit customer information function on maintain customer page. | The customer information could be updated. | **Passed** |
|  | Test that the user search customer function on maintain customer page. | The data could be found by filling in following search conditions: customer ID, and customer name. | **Passed** |
|  | Test that the user could delete customer information on maintain customer page. | The customer information has been deleted. | **Passed** |
|  | Test that the select list for department information can be chosen on create new project page. | The select list for department information could allow user to choose the specific department. | **Passed** |
|  | Test that the user could create new project. | The new project could be created. | **Passed** |
|  | Test that the search project function on maintain project page. | The data could be found by filling in following search conditions: project ID, project name, and customer project status. | **Passed** |
|  | Test that the user edit function on maintain project page. | The project information could be updated. | **Passed** |
|  | Test that the user could delete project information on maintain project page. | The project information has been deleted. | **Passed** |

#### Team leader module test cases

In this case, tester should login as team leader, and test all the team leader functions listed below.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Test that the team leader could search project information on manage project page. | The data could be found by filling in following search conditions: project ID, project name. | Passed |
|  | Test that the textbox allow user to enter information and the select list allow user to choose project and employeeon create new task page. | The textbox could be inserted, and project name and employee name select list contain the project information and employee name. | **Passed** |
|  | Test that user could insert new task information on insert new task page. | The new task has been inserted. | **Passed** |
|  | Test that the user could search task information on manage project page. | The data has been found. | **Passed** |
|  | Test that the edit task function allows user to update existing task information. | The task information could be updated. | **Passed** |
|  | Test that the delete task function allows user to delete existing task information on manage project page. | User chooses the information required deleting, and click delete button.  Click ok on the pop-up window that asks are you sure to delete?  Data should be deleted. | **Passed** |

## Conclusion

Overall, this project has met all the requirements of Suplus'. According to the requirement of Suplus Technology, Suplus Office Management System adapted prototype development methodology to develop. Meanwhile, this system is using PHP, HTML, and CSS as major programming languages, and adapting JQuery, JSP and Bootstrap to enhance the user interface layout. In addition, there is a social network page maintained by Mongo database to allow all the users to post status (message or pictures), and comments.

In this system, two databases are adapted: MySQL and NoSQL (MongoDB). On the one hand, MySQL database designs and manages employee, customer, project, and task information. On the other hand, for the sake of innovation and experience in advanced technologies, Mongo DB (NoSQL) collects data from the social media website page-communication, which would allow users of this office management system to post messages, pictures, and comment others’ messages and images.

The user interface of the web application is user friendly because this system adapts another new leading language—Bootstrap—which is the most popular language being used to develop responsive mobile project on the Web. During the testing part, using bootstrap fixes simple interface design problems that have been mentioned by most testers.

In the process of developing the system, I practiced and learned a lot of programming languages and challenged myself to use new technologies such as NoSQL and Bootstrap to develop social media web page. In addition, I used popular languages such as PHP, Bootstrap, Java script, and JSP, to develop this system. This practicum is a challenge and also an opportunity to explore and practice new technologies and readily learned programming languages.

Lastly, I conducted user and functional tests to ensure that all the functions designed are working smoothly. In the user testing, participants were asked to follow the task list to provide feedback that includes the results of the test and suggestions for future improvement. In the functional testing, using the test result, the designer debugged and fixed unfeasible functions by changing codes, and using Bootstrap to redesign the user interface. In conclusion, this practicum project perfectly satisfied the users of Suplus Technology.

## Future Enhancement

For future enhancement, a sending report function could be added to the system. In the future, the entire body of employee would be required to fill in a checklist to explain today’s work situation based on achieved tasks. The data generated by the checklist will be turned into a report that will be sent to the customer at the end of every month. This report helps the customers to be aware of the stage that they are at in the project.

In addition, to the targeted users mentioned in this paper, employees in the finance department could use this system. The system could calculate employees’ salary function in the future design. For example, the system could generate employee’s payment that may include bonuses given based on work ethic and performance. Another function is to create a budget sheet for all the departments in the company. The budget sheet could keep track of the income and expenses. Due to the limited amount of time allotted for this project, this practicum version did not include the aforementioned functions.

## Appendix

### Work Breakdown

Based on the users requirement, totally designed five versions for Suplus Office Management System. These versions have been discussed on part 5 Development Methodology.

According to client’s requirement, designed the first mock-up version. This is the step 2 in the following table. The purpose of initial mock-up version is to presenting, and discussing with users of Suplus Technology. And improved the mock-ups based on the users’ suggestions.

During the system design, transformed the latest mock-ups to a HTML and CSS version. Later on, conducted the PHP, MySQL, Mongo DB and Bootstrap to the system. After accomplished the system, firstly, I have tested the system based on the test cases, and invited the general testers and users of Suplus Technology to test the system as well. All the testers were required to providing test results and suggestions.

At the end, according to the test results and suggestions collected from testers, system has been fixed functions and improved the user interface layout. This is the final version of the system.

In order to demonstrate the work I have been done since the last year, I spent almost five months to write report document. This documentation explains the system functions and the design process.

|  |  |
| --- | --- |
| **Components** | **Time Breakdown** |
| **1.Identify basic functional requirement** | |
| Decide the system functions based on the user requirement. | March-April 16, 2014 |
| **2. Develop initial prototype** | |
| Based on the user requirement to create initial mock-up | April 17– May 10, 2014 |
| Improve mock-ups | May 14– June 1, 2014 |
| **3.System design** | |
| Website design/review | June 3– June 5, 2014 |
| Develop RDBMS | June 8- July 6, 2014 |
| Develop employee maintain function | July 10- August 20, 2014 |
| Develop customer maintain function | August 25-September 6, 2014 |
| Develop project maintain function | September 9- October 15, 2014 |
| Develop NoSQL database | October 17- November 5, 2014 |
| Develop communication page | November 8- December 29, 2014 |
| Enhance User Interface design | January 3- January 23, 2015 |
| **4. Test** | |
| Test and debug | January 23-Febuary 20, 2015 |
| **5.Final report** | |
| Documentation | November 20,2014 – March 31, 2015 |

### Test Cases

This section demonstrates the test cases have been conducted, during the system function testing.

A. All user test case

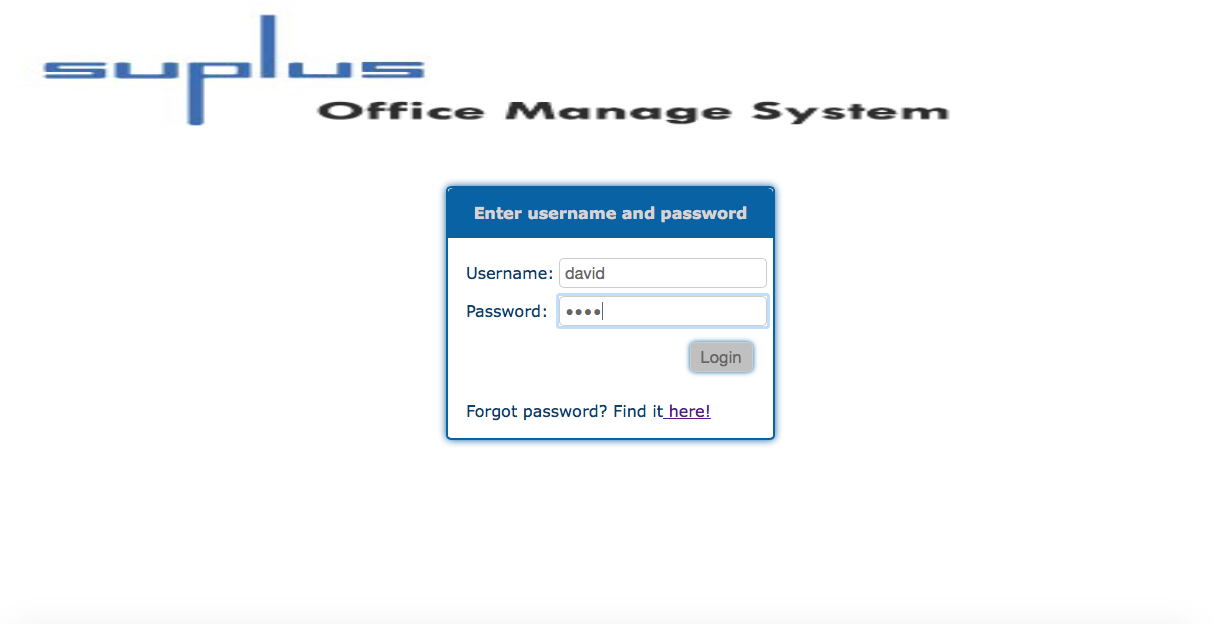
Login

1. Test that the login textboxes could be filled information on the login page.

Test Scenario T1:

1. Enter username into username text field.
2. Enter password into password text field.

Test result:



Login text field could insert information. The test is passed.

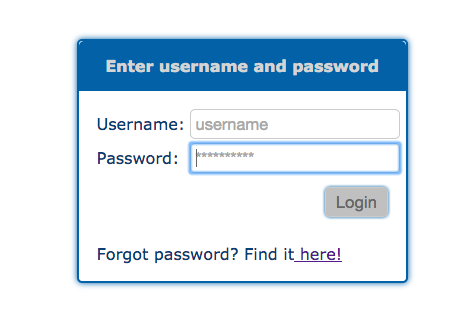
1. Test that an error message presents to indicate user or invalid data or empty blank.

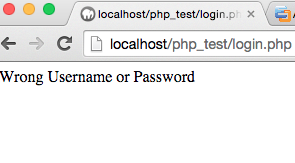
Test Scenario T2:

1. First, do not enter anything in the text filed.
2. Second, input incorrect (does not existing information) in the text filed.
3. Click login button.

Test result:

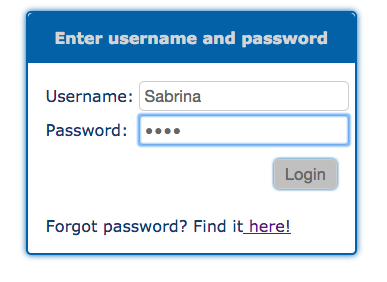
First time:

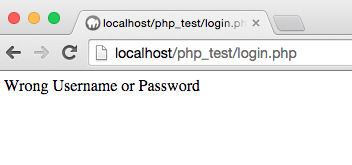




Login system without enters any username or password, could not login to system.

Second time:





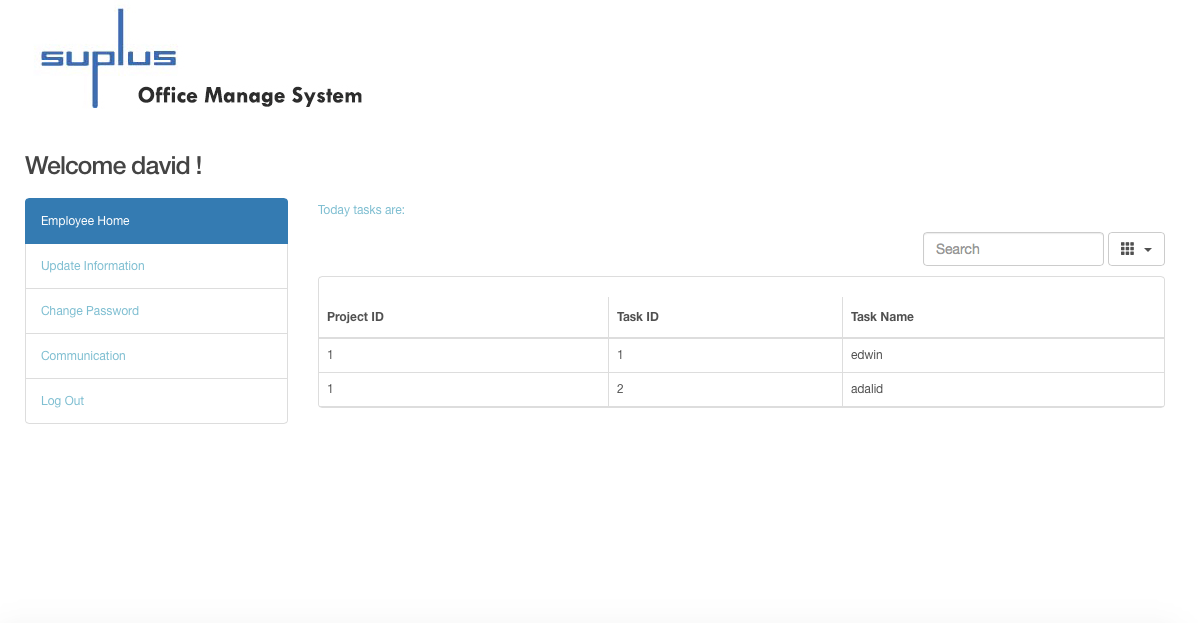
Login system with not existing password and username could not login as well. The test is PASSED.

1. Test the existing user could login to system by enter the valid username and password.

Test Scenario T3:

1. Input correct user name and password. For example, enter user name as David, password as1234.
2. Click login button.

Test result:



When user enter existing login information, user could login to system, the precious picture show after enter David’s username and password, David login to his personal home page. This test passed.

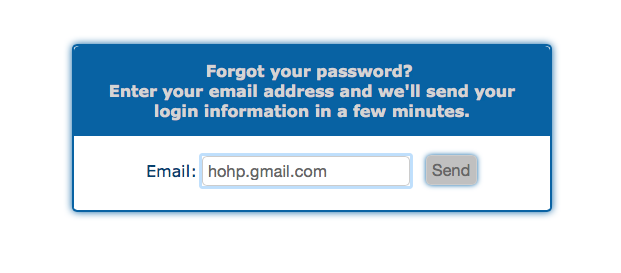
1. Test that the system could send email with password to binding email address when user clicks the forgot password button on login page.

Test Scenario T4:

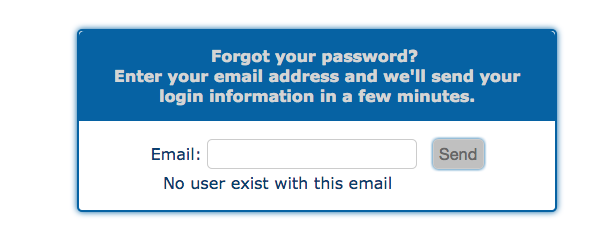
1. Click the forgot password link on the login form.
2. First, input not existed or invalid email address.
3. Second, input existed and valid email address.
4. Click login button.
5. Check email inbox with the new letter with password or not.

Test result:

First time:



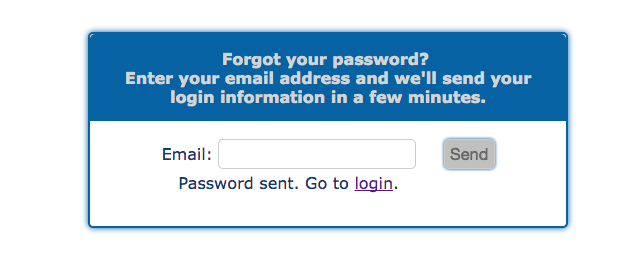
Click send button with percious uncorrect email address, system reminds no exist user.

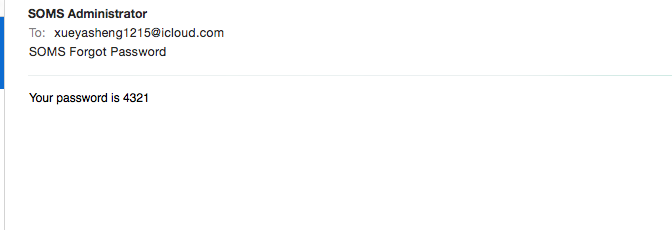


When enter not correct email address, system could not connect database with this invalid information.

Second time:

Enter existing user’s email address, password has been sent already.





Enter the correct email address, and user got an email with password. The test is passed.

Update personal information

1. Test that the user could edit personal information on update information page.

Test Scenario T5:

1. Enter the valid information into the text filed, change postcode from V5H 1P2 to V5H 1P1.
2. Click update button.

Test result:





The information could be updated. The test is passed.

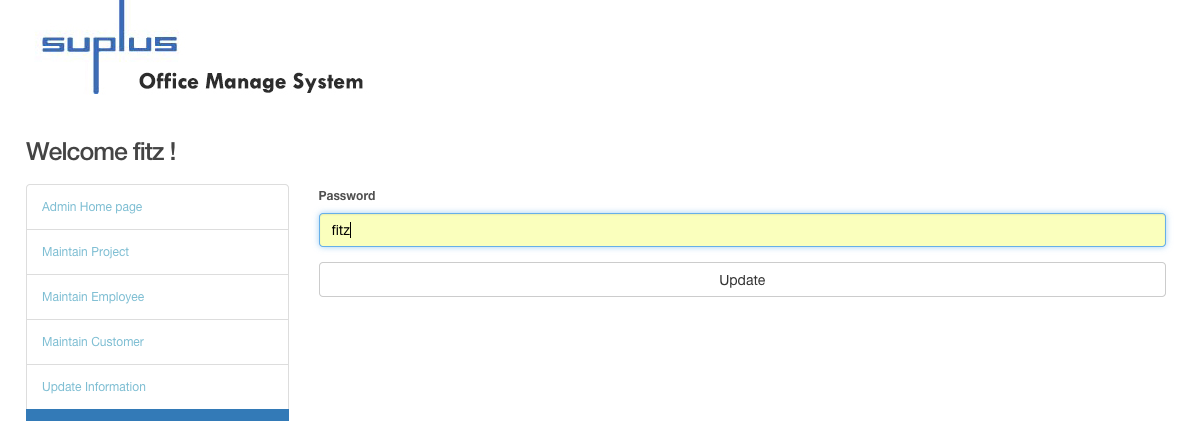
Change password

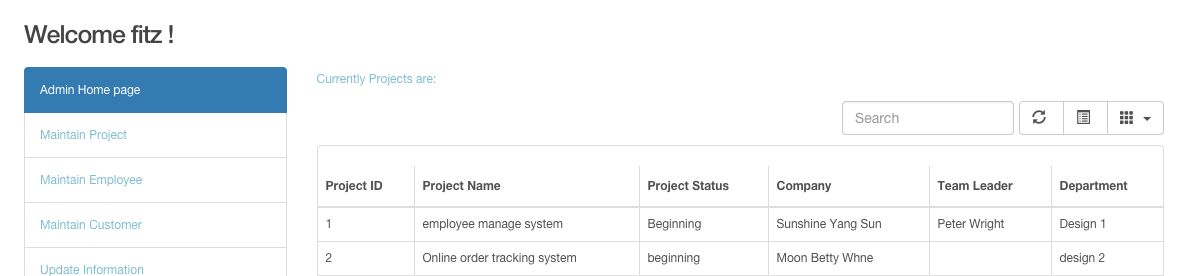
1. Test that the user could change password once have signed in the system.

Test Scenario T6:

1. User login to the system already.
2. Click the change password link on navigation menu.
3. Enter new password, change user Fitz’s password from 4321 to fitz.
4. Click Update.

Test result:





Use new password 4321, user Fitz could login as admin. The test is PASSED.

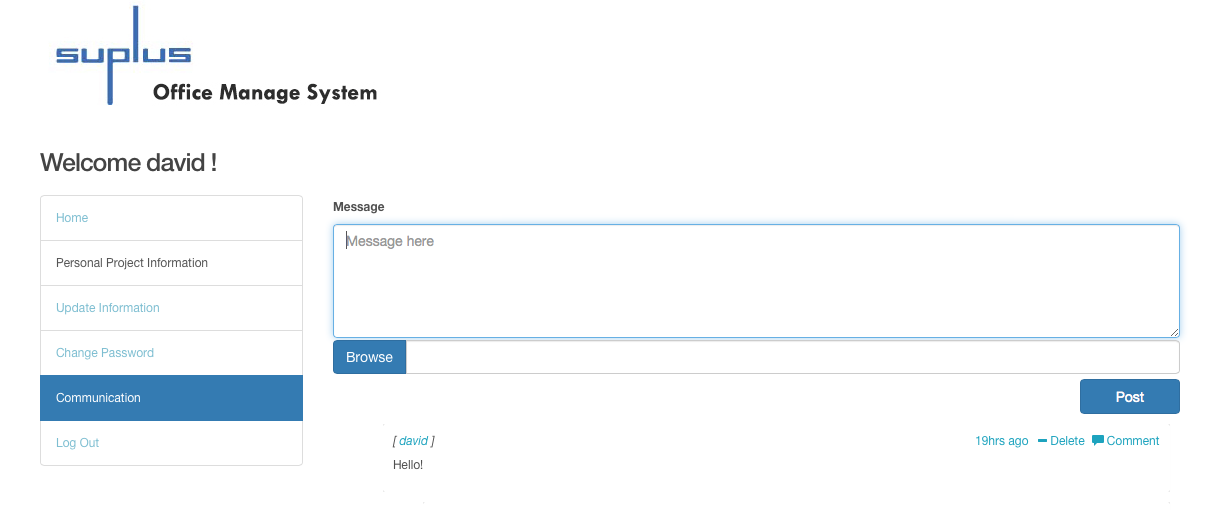
Communication

1. Test that the user could post message on communication page.

Test Scenario T7:

1. Log in system by using user name David, and password 1234.
2. After user login to system, user enters text “Hello!” in the message box.
3. Click post button.

Test result:

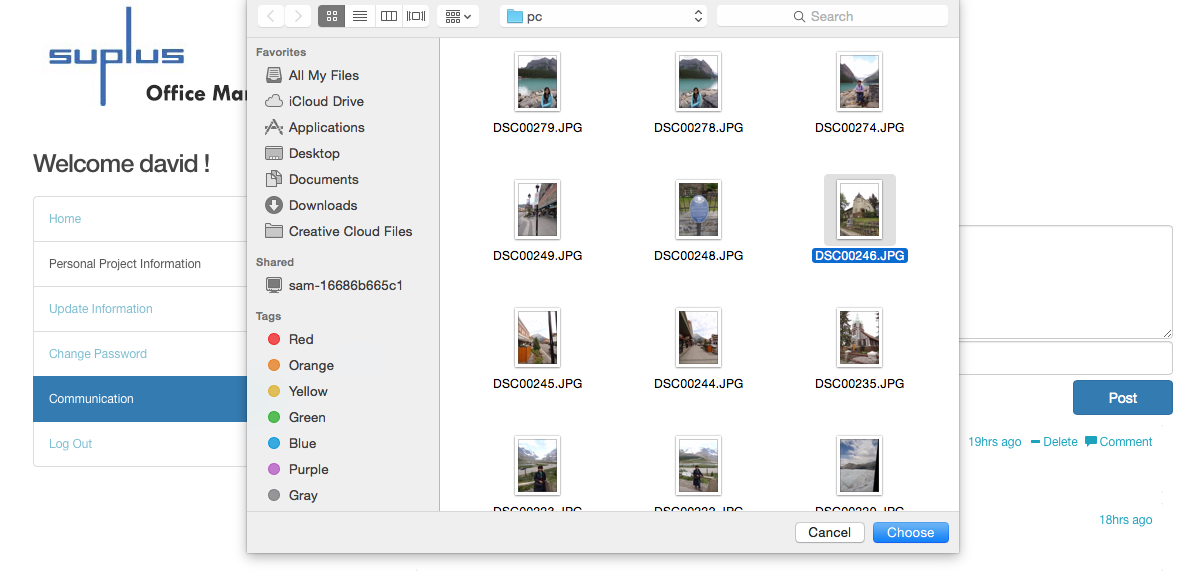


Message posted successfully. The test is PASSED.

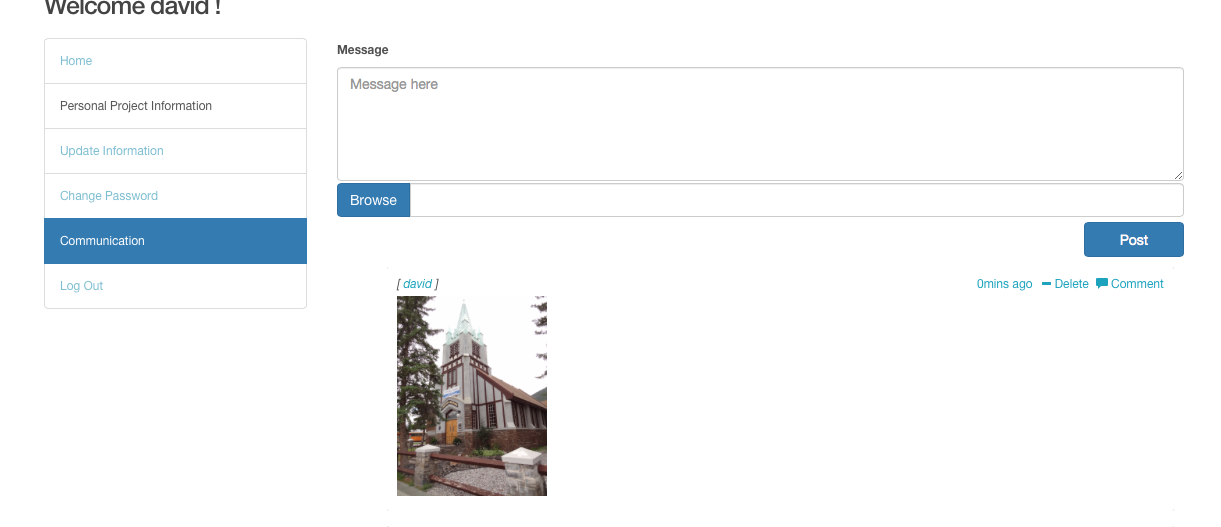
1. Test that the user could post a picture on communication page.

Test Scenario T8:

1. Click Browse button.
2. Choose a picture from the pop-up window.
3. Click button choose.
4. The picture has been chosen, and clicks button post.

Test result:

After click the button choose, the picture direction has been selected, and click post button.



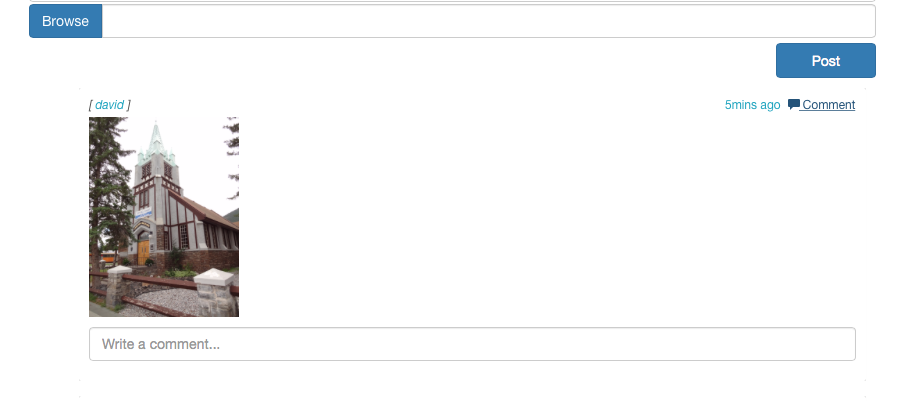
Picture posted successfully, the test is passed.

1. Test that the user could post comment on communication page.

Test Scenario T9:

1. Login as user Pete, and prepare to comment David’s picture.
2. Click comment.
3. Type in a message into the write a comment field.

Test result:





The message from Peter to comment David’s picture has been posted. The test is PASSED.

B. Employee test case:

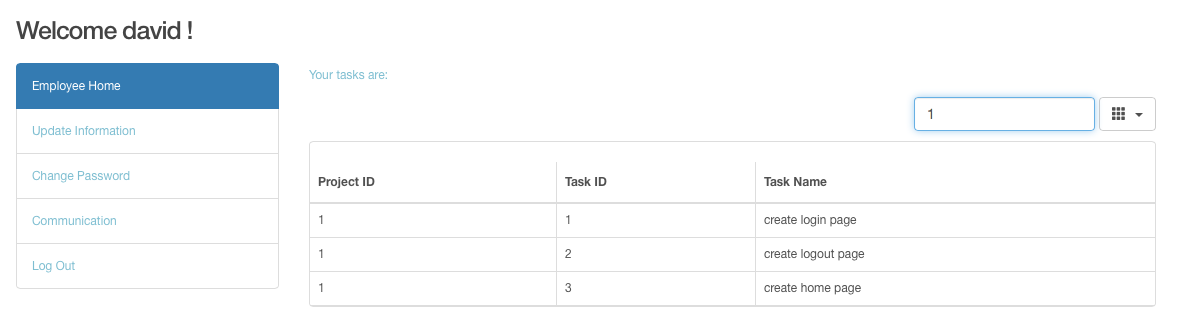
Check personal project information

1. Test that the search textbox on employee home page allow user input information.

Test Scenario T10:

1. Enter any task name, task ID, or project ID.

Test result:



Information could input to the search text field. The test is PASSED.

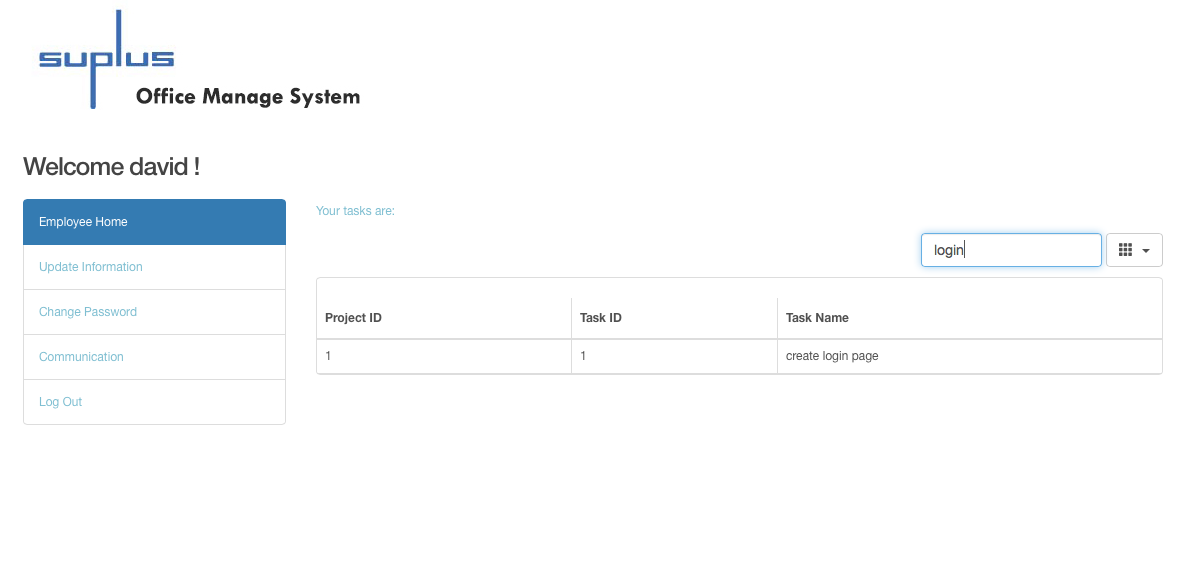
1. Test the search function on employee home page

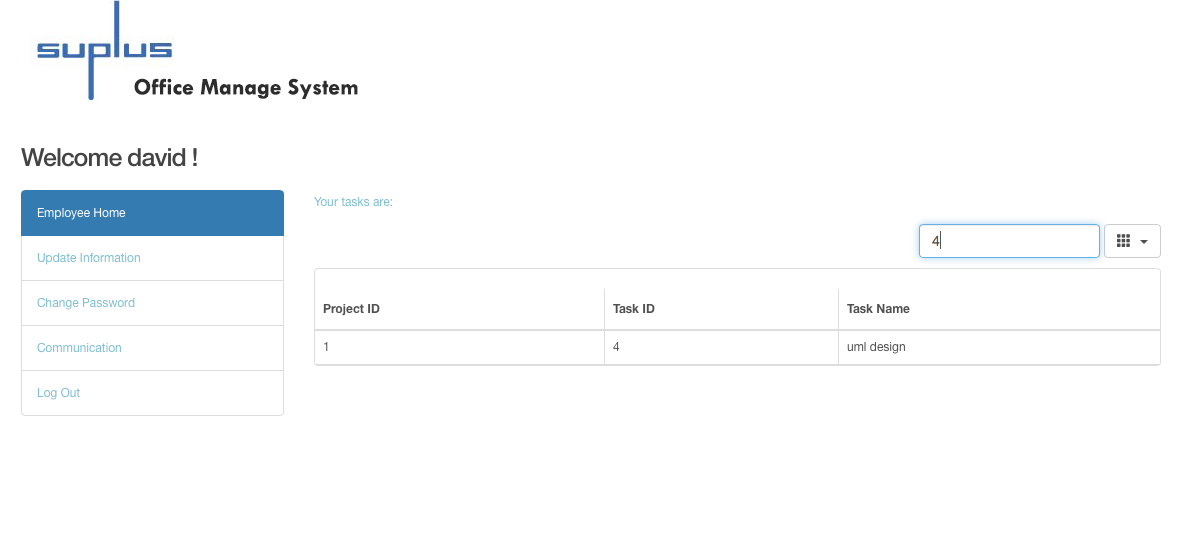
Test Scenario T11

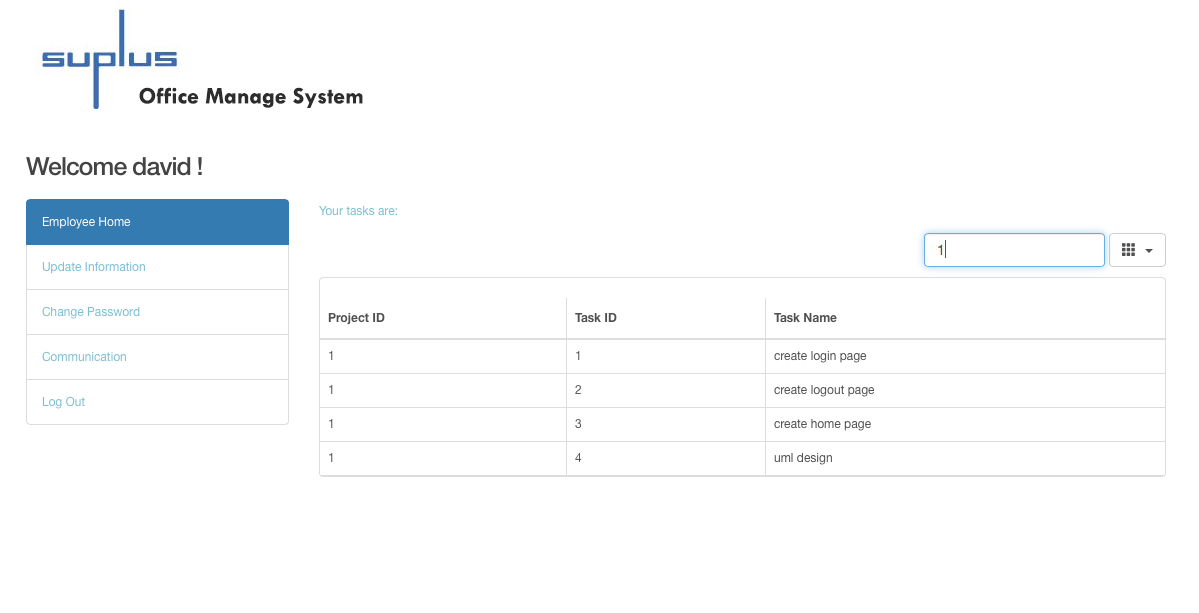
1. Enter valid task name in this case, enter “login” as task name.
2. Enter valid task ID, type in 4 as task ID in the search bar
3. Enter valid project ID, in this case, search related information when project Id is 1
4. Enter not existing information, such as “rest”

Test result:

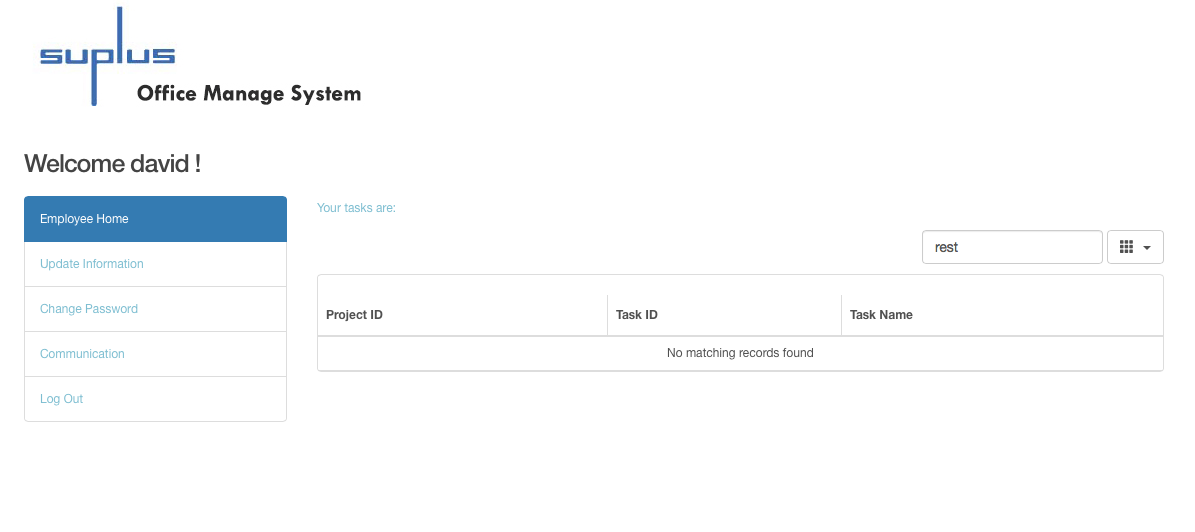
Search by task name login:



Search task ID as 4:

Search project ID as 1: 

Search not existing information “rest”:



System reminds “No matching records found” when search not existing information.

System could search employee’s task ID, task name and project ID. The search function is working. The test is passed.

1. Administrator test case

Home Page

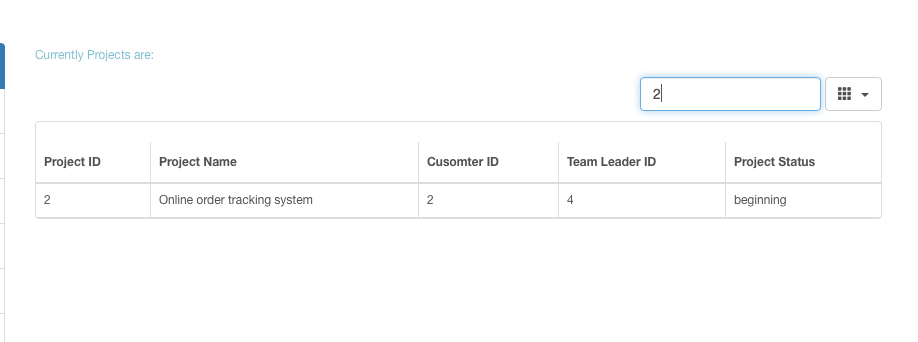
1. Test that the administrator search function on admin home page**.**

Test Scenario T12:

1. Enter valid project ID, input 2 as project ID.
2. Enter valid project name, input existing project name.
3. Enter valid customer ID, type 1 as customer ID.

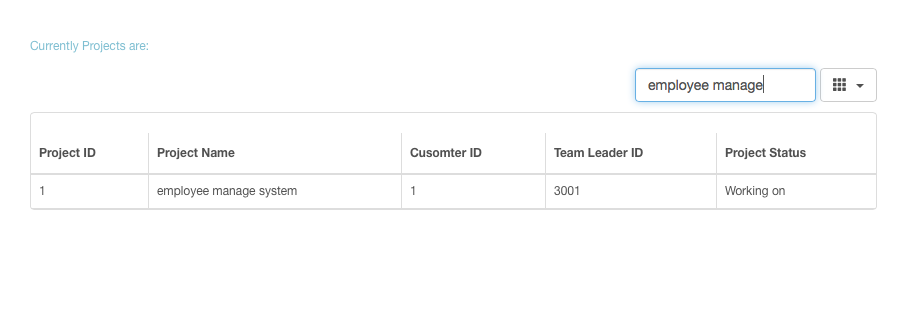
Test result:

Search project information when project ID is 2:



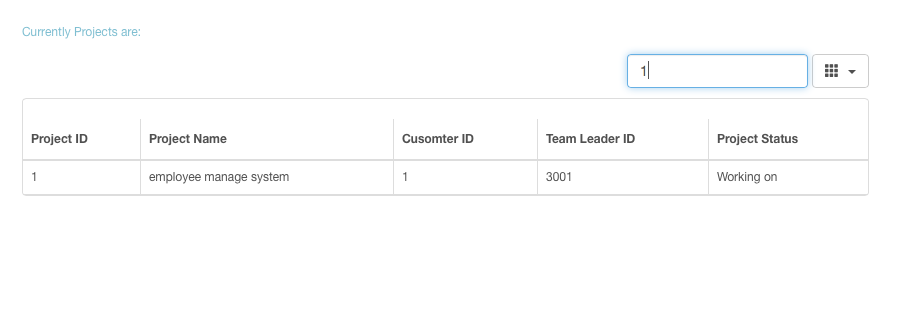
Typed in 2 as project ID, found a record, test is passed.

Search project name as employee manage:



Figures show the existing project could be searched when search by project name.

Search customer information when customer ID is 1:

  
Type 1 as customer ID, found 1 record, the test is passed.

Therefore, administrator could search project related information on the home page.

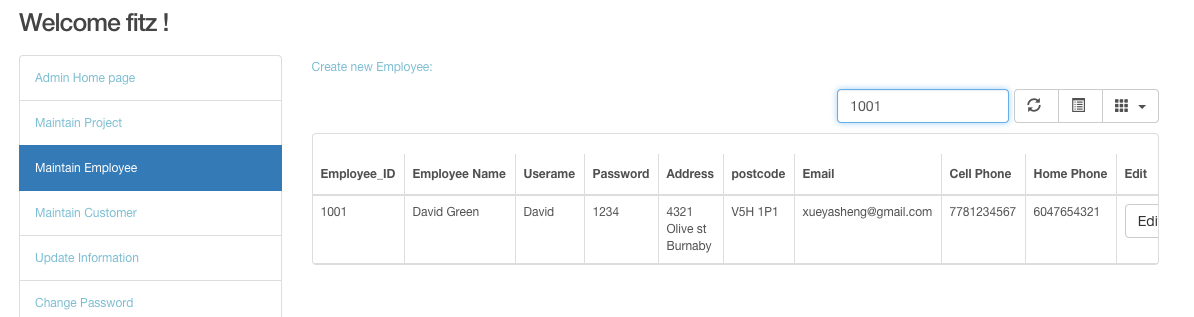
Maintain Employee

1. Test that the administrator could search employee information on maintain employee page.

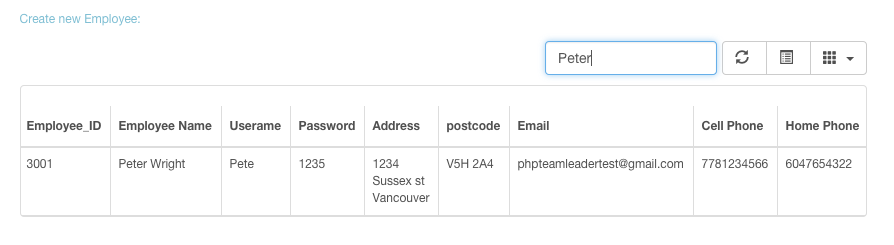
Test Scenario T13:

1. Search employee information by enter Employee ID as 1001.
2. Search employee information by enter Employee name.

Test result:

Search employee who’s ID is 1001: 

Search employee name is Peter:



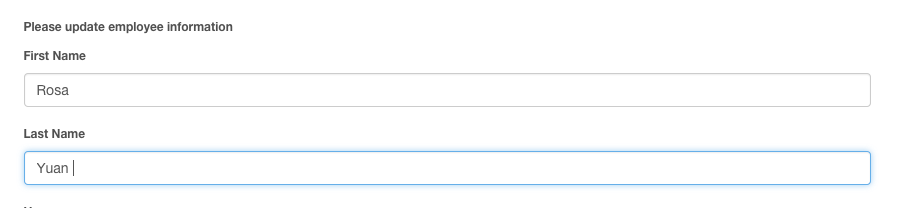
The test is PASSED.

1. Test that user edit employee information function on maintain employee page.

Test Scenario T14:

1. Choose data that employee name is “testdfd”.
2. Click the edit button.
3. Change employee name to “Rosa Yuan”.
4. Click update button.



Insert Rosa Yuan to the name textboxes: 

Test result:



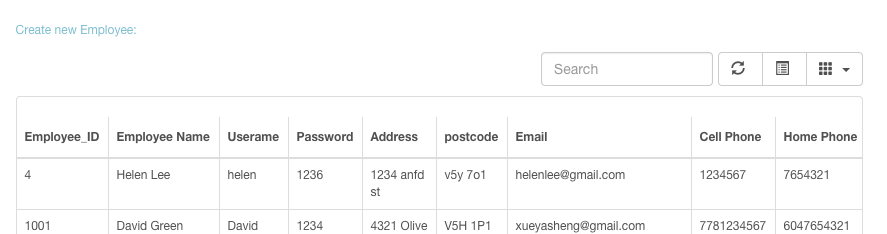
Employee name has been changed. The test is PASSED.

1. Test that the “Create new employee” link on maintain employee page.

Test Scenario T15:

* + - 1. Click add new employee link on Maintain Employee page.

Test result:



After click the link, system went to create employee page.



The link located to add new employee page. The test is PASSED.

1. Test that the textbox could insert information on create new employee page.

Test Scenario T16:

* + - 1. Enter new employee name as “Roman Wang”.

Test result:



The text fields could be inserted. The test is passed.

1. Test that the administrator create new employee function on maintain employee page.

Test Scenario T17:

1. Enter new employee name as “Roman Wang”.
2. Click Add button.

Test result:



Roman has been inserted to employee list on maintain employee page. The test is PASSED.

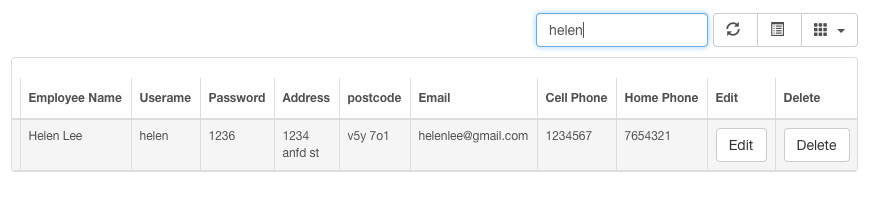
1. Test that the administrator delete employee function on maintain employee page.

Test Scenario T18:

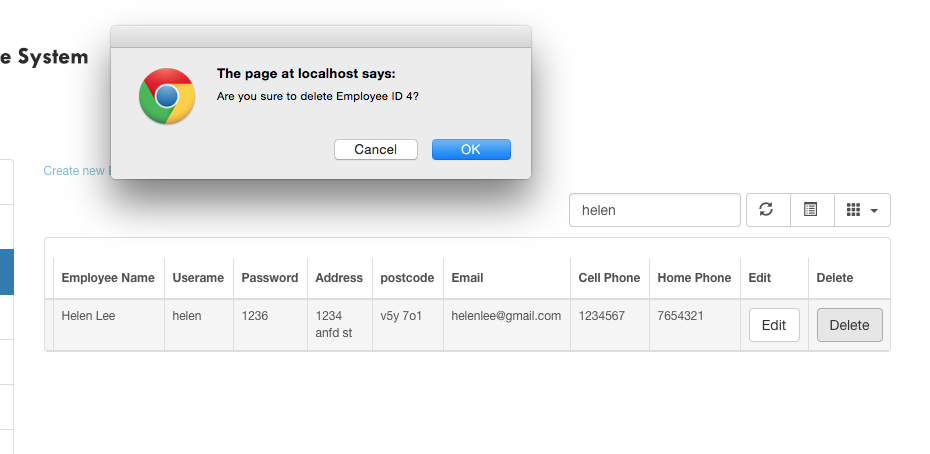
1. Search employee information whose name is Helen.
2. Click delete button.
3. Click ok button on are you sure to delete information pop-up window.

Test result:

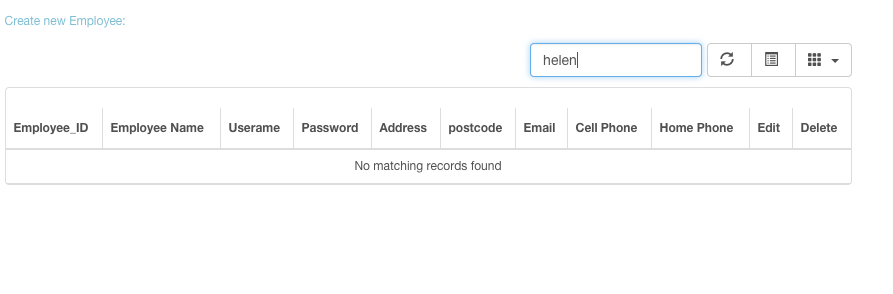
Search Helen and click Delete button:



Click OK button on pop-up window:



Search Helen again, could not found data.



The information has been deleted. The test is PASSED.

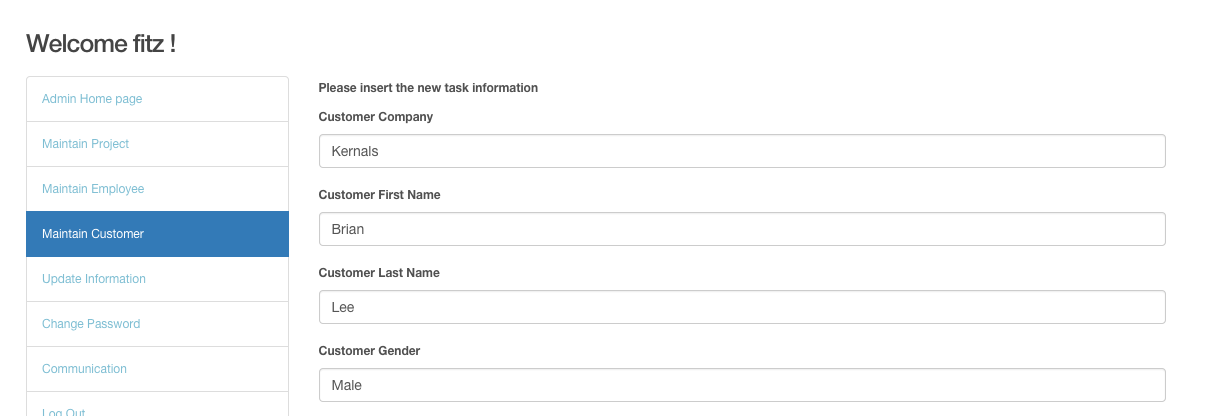
Customer Maintain

1. Test that the administrator create new customer function on maintain customer page.

Test Scenario T19:

* + - 1. Click add new customer link
      2. Insert customer information
      3. Click add button

Test result:



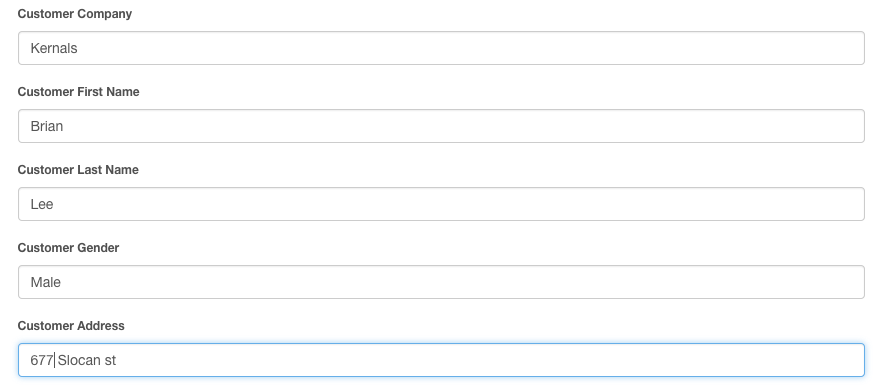
The picture above presents a new customer has been inserted. The test is PASSED.

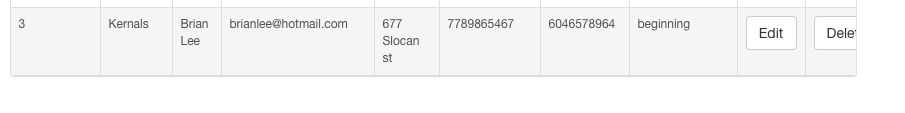
1. Test that the user edit customer information function on maintain customer page.

Test Scenario T20:

1. Choose data customer ID is 3.
2. Click edit button.
3. Change customer address from “678 slocan” to “677 Slocan”.
4. Click update button.

Test result:





Customer’s address has been changed from 678 to 677. The test is PASSED.

1. Test that the user search customer function on maintain customer page.

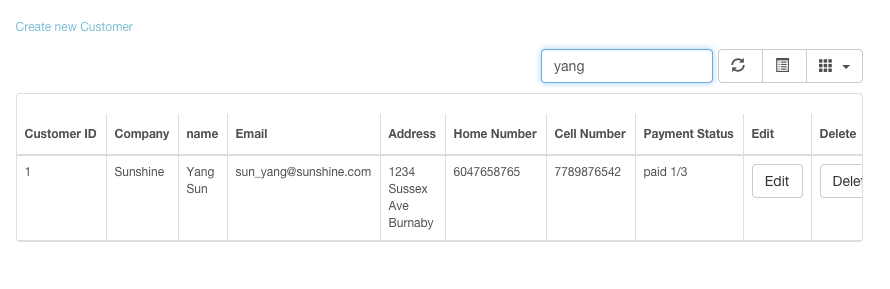
Test Scenario T21:

1) Enter customer name by “Yang” in search menu.

2) Enter customer ID “1” in search menu.

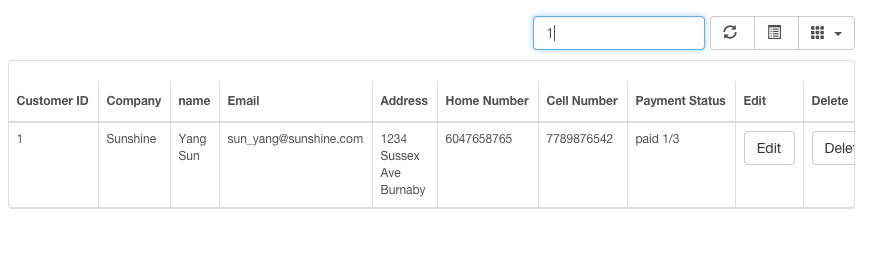
Test result:

Search customer name:



When user enter existing customer name, the data could be found.

Search customer ID:



When user enter customer ID as “1” to search information, the data could be found.

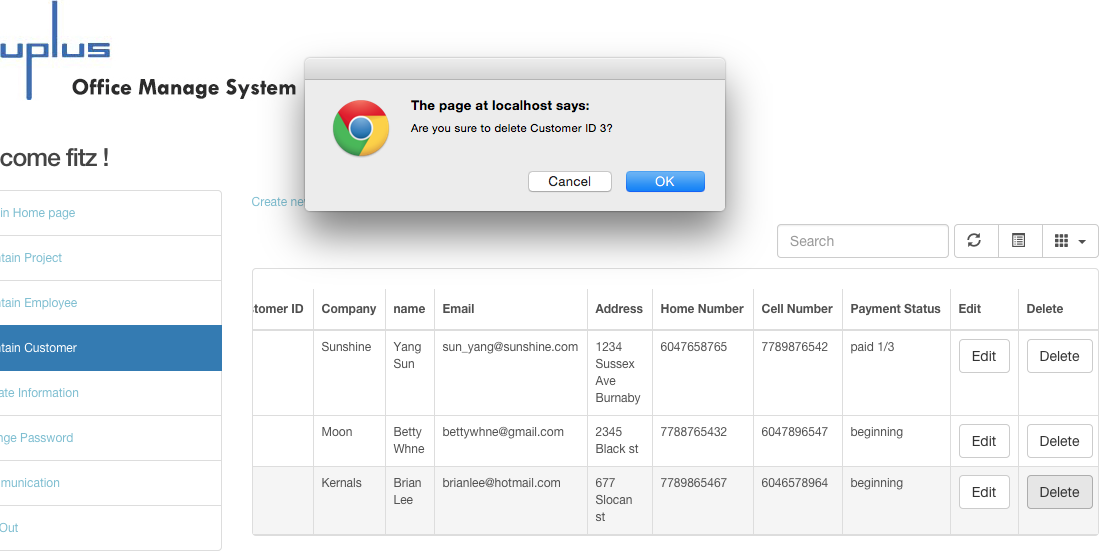
The test is PASSED.

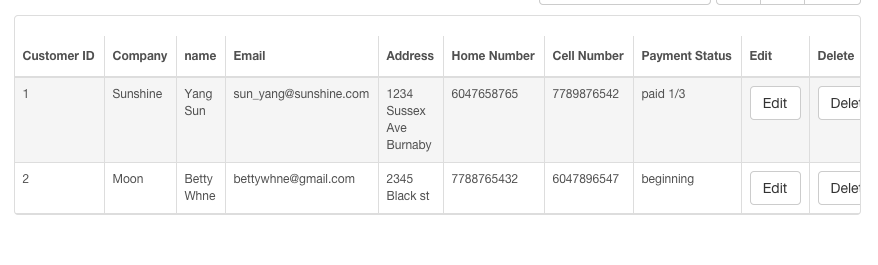
1. Test that the user could delete customer information on maintain customer page.

Test Scenario T22:

1. Search information customer company name is “Kernals”.
2. Click delete button.
3. Click Ok button on pop-up window.

Test result:





Customer information with kernels has been deleted. The test is PASSED.

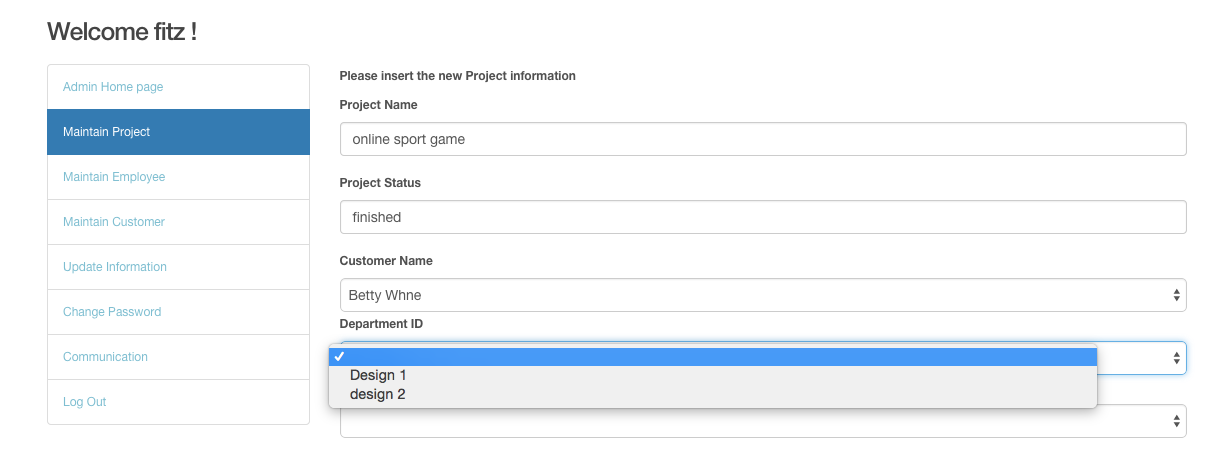
Maintain Project

1. Test that the select list for department information can be chosen on create new project page.

Test Scenario T23:

1. Click create new project.
2. Enter project name as online sport game.
3. Click department select list button.
4. Choose department 2.

Test result:



The select list contains Design 1 and Design 2. The test is PASSED.

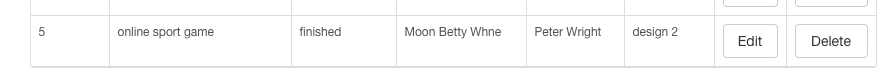
1. Test that the user could create new project.

Test Scenario T24:

1. Insert project name.
2. Choose customer name.
3. Type in project status.
4. Choose department.
5. Choose Team leader.
6. Click add button.

Test result:





The new project added. The test is PASSED.

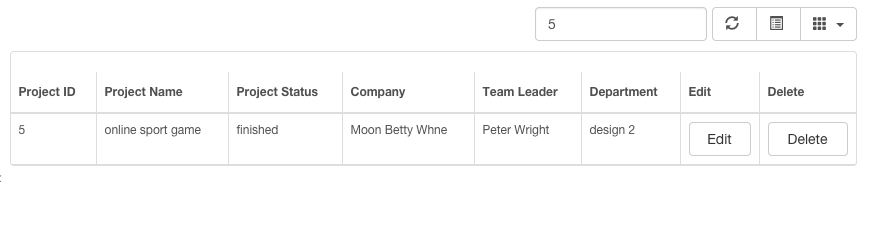
1. Test that the search project function on maintain project page.

Test Scenario T25:

1. Search project information by enter project ID as 5.
2. Search project information by enter project status as fin.
3. Search project information by enter project name as track.

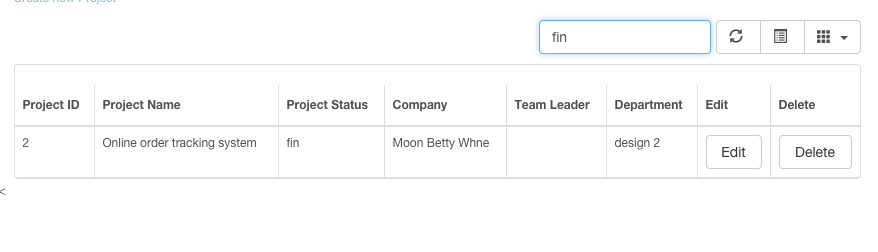
Test result:

Search project ID as 5:



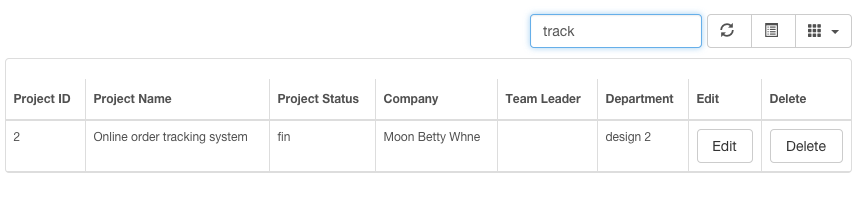
The data could be found.

Search project status as fin:



The data could be found.

Search project name as track:



The data could be found.

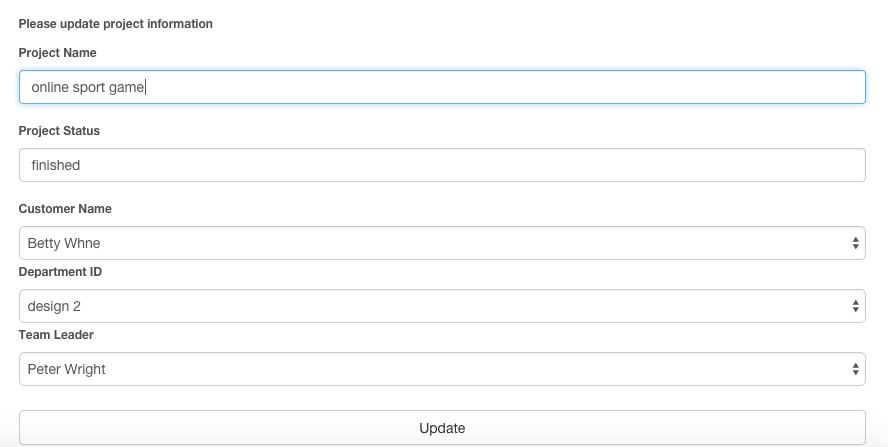
The test is PASSED.

1. Test that the user edit function on maintain project page.

Test Scenario T26:

1. Choose project information whose project ID is 5.
2. Click edit button.
3. Change project status from “finished” to “paid 1/3”.
4. Click update button.

Test result:





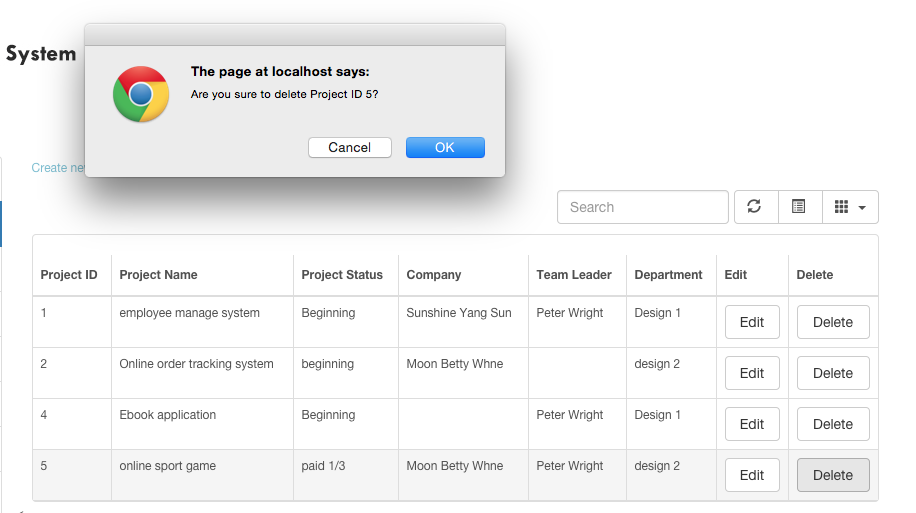
Project status has been changed. The test is PASSED.

1. Test that the user could delete project information on maintain project page.

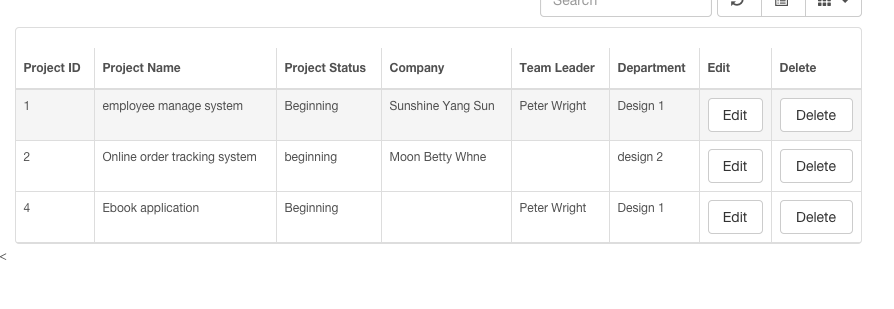
Test Scenario T27:

1. Choose data project ID is 5.
2. Click delete button.
3. Click ok button on pop-up window.

Test result:



After click Ok button:



The NO. 5 project information has been deleted. The test is PASSED.

D. Team leader test case

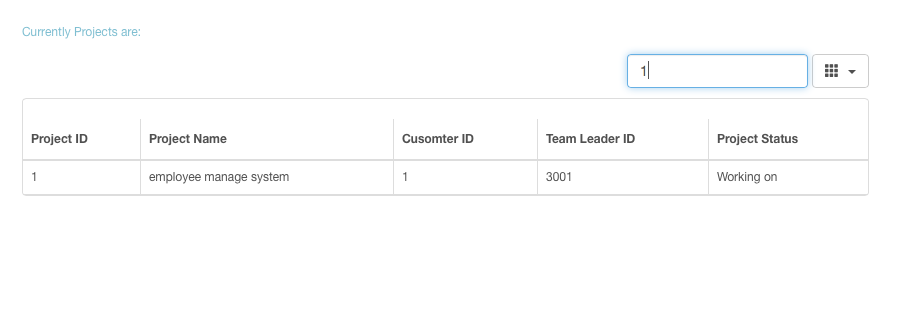
28. Test that the team leader could search project information on manage project page.

Test Scenario T28:

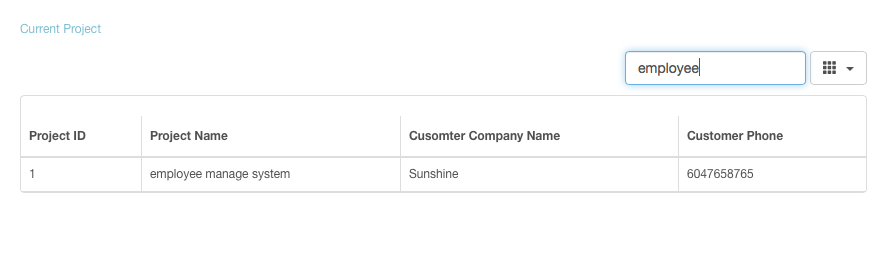
1. Search project information by enter project ID as 1.
2. Search project information by enter keyword employee of project name.

Test result:

Search by project ID:



Search by project name:



When user search project by project ID, there is a record could be found, and search project name by enter the keyword, could found the existing data as well. The test is passed.

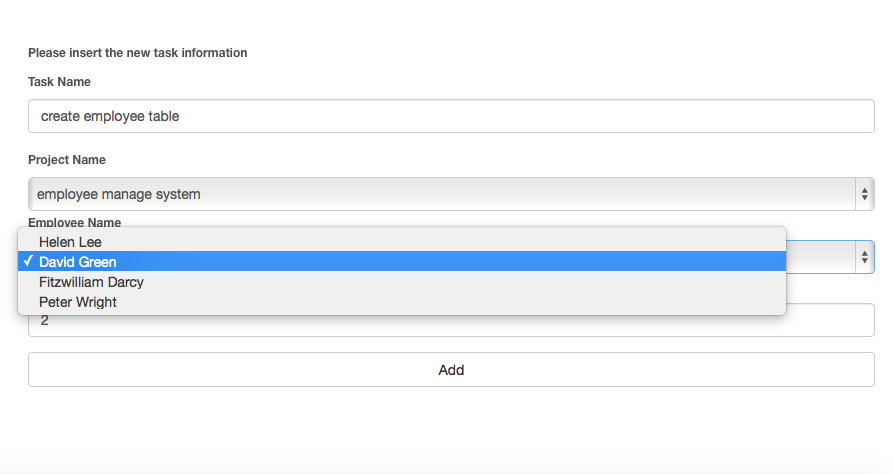
Manage project

29. Test that the textbox allow user to enter information and the select list allow user to choose project and employeeon create new task page.

Test Scenario T29:

1. Click the Insert a new task link on the top of manage project page.
2. Insert the task name as create employee table.
3. Click project name select list button and choose the existing project.
4. Choose employee name.
5. Enter estimate time.

Test result:



The test fields could be inserted, and project name and employee name select list contain the project and employee name. The test is PASSED.

30. Test that user could insert new task information on insert new task page.

Test Scenario T30:

1. Click the Insert a new task link on the top of manage project page.
2. Insert the task name as create employee table.
3. Click project name select list button and choose the existing project.
4. Choose employee name.
5. Enter estimate time.
6. Click add button.

Test result:



The task create employee table has been inserted to database, and presents the system name, employee name, and estimate hour as same as entered. The test is passed.

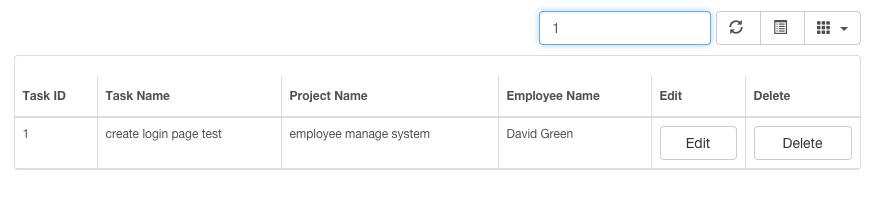
31. Test that the user could search task information on manage project page.

Test Scenario T31:

1. Search task information by enter task ID as 1.
2. Search task information by enter task name as login.
3. Search task information by enter employee name as green.

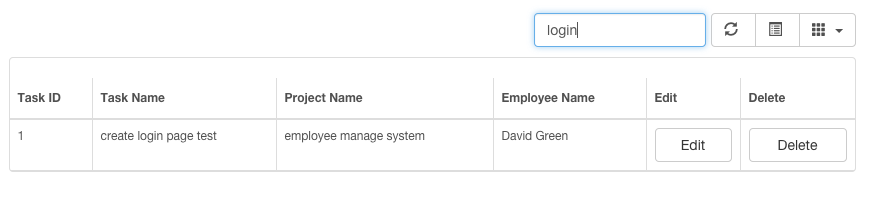
Test result:

Search task information by enter task ID as 1:



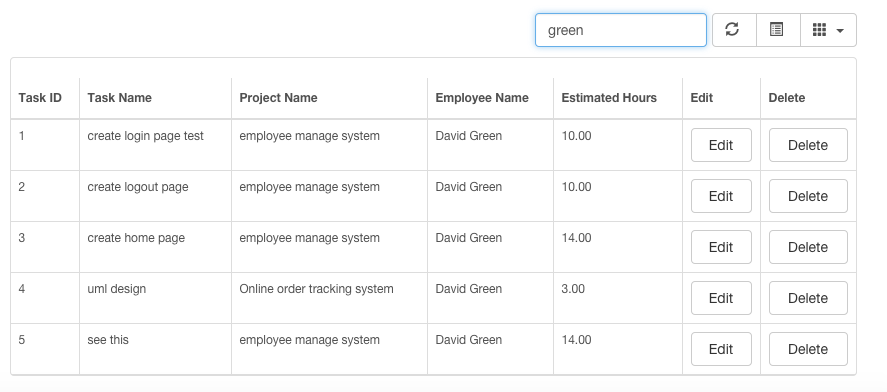
There is a data has been found. The test is passed.

Search task information by enter task name as login:



There is a data has been found. The test is passed.

Search task information by enter employee name as green:



There is a data has been found. The test is passed.

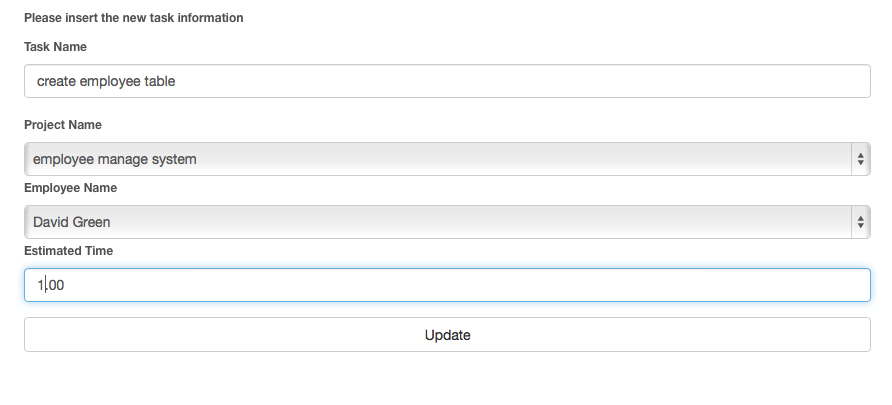
32. Test that the edit task function allows user to update existing task information.

Test Scenario T32:

1. Choose task “create employee table”.
2. Click edit button.
3. Change the estimate time from 2 to 1.
4. Click Update button.
5. Click the Edit button on NO.5 task



Change the estimate time from 2 to 1:



Test result:



The estimate time has been change to 1.0. The test is passed.

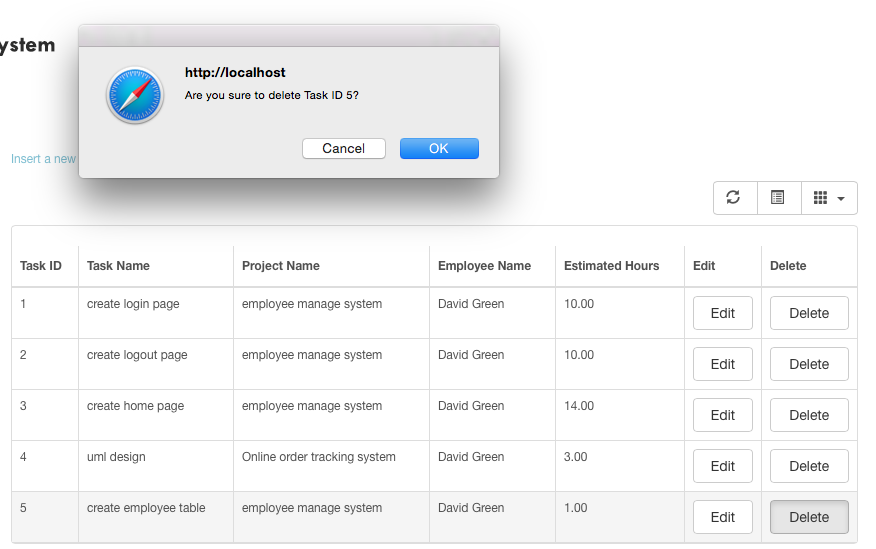
33. Test that the delete task function allows user to delete existing task information on manage project page.

Test Scenario T33:

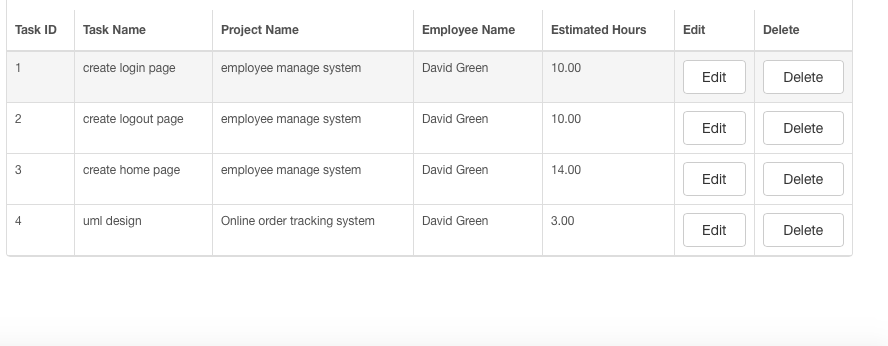
1. Choose task “create employee table”.
2. Click delete button.
3. There is pop-up window to ensure user decided to delete task.
4. Click Ok on the pop-up window.

Click the Delete button on NO.5 task:



Click the Ok on pop-up window: 

The result:



Information has been deleted. The test is PASSED.

### C. Comments of User Testing

For improve system operation, I adapted user tests. The testers were required to operate system by the task lists, which I have mentioned on the user tests. In addition, I conducted a survey as well, and I listed one survey as an example.

Web browser: Google Chrome

Could you select the module you’ve been tested?

B

A. Employee Module B. Administrator Module C. Team Leader Module

How long did it take you to complete all the tasks based on the task list?

A

A. 20min-30 min B. 30min-1hour C. more than 1 hour

Do you have any comment?

No.

## Reference

[Leavitt, N.](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=p_Authors:.QT.Leavitt,%20N..QT.&searchWithin=p_Author_Ids:37284031700&newsearch=true) (2010). Will NoSQL Databases Live Up to Their Promise? *Computer.*

Retrieved from: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=5410700&url=http%3A%2F%2Fieeexplore.ieee.org%2Fstamp%2Fstamp.jsp%3Ftp%3D%26arnumber%3D5410700

Francia, S. (2012). *MongoDB and PHP*.

O’Reilly Media, CA

Sadalage, P., Fowler, M. (2013). *NoSQL distilled: a brief guide to the emerging world of polyglot.*

Person Education. Retrieved from:

https://books.google.ca/books?id=AyY1a6-k3PIC&printsec=frontcover&dq=NoSQL+database&hl=en&sa=X&ei=GsYTVd2YCNCuogShuoKIDQ&ved=0CDAQ6AEwAA#v=onepage&q=NoSQL%20database&f=false