|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test NO. | Testing | Expected Outcomes | Result | | | | Tester |
|  | Test if information could be filled in the login textboxes on the login page. | Information could be entered in the textboxes of username and password. | **Passed** | | | | E  A  T  G |
|  | Test if an error message is present to indicate invalid data, or empty blank. | * Enter not existing username and password—fail to login to system. * Empty username and password—fail to login to system. | **Passed** | | | E  A  T  G | |
|  | Test if the existing user could login to system by entering the valid username and password. | The user could login to system. | **Passed** | | E  A  T  G | | |
|  | Test if 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 could be sent. | **Passed** | E  A  T  G | | | |
|  | Test if the user could edit personal information on update information page. | The information could be updated. | **Passed** | E  A  T  G | | | |
|  | Test if the user could change password once signed in the system. | The user could enter the new password on change password page.  After logout, the user could login system with the new password. | **Passed** | E  A  T  G | | | |
|  | Test if the user could post message on communication page. | The user who logins with valid username and password could post message on communication page. | **Passed** | E  A  T  G | | | |
|  | Test if the user could post a picture on communication page. | The user could select a picture from local directory, and the user could post a picture. | **Passed** | E  A  T  G | | | |
|  | Test if the user could post comment on communication page. | The user could post a comment. | **Passed** | E  A  T  G | | | |

|  |  |  |
| --- | --- | --- |
|  | Test if the search textbox on employee home page allows the user to input the information. | The search textbox could allow the user to insert the information. |
|  | Test the user search function on employee home page. | The data could be found by filling in following search conditions: task ID, task’s name and project ID. |

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

|  |  |  |
| --- | --- | --- |
|  | Test if the user could search for project’s information on manage project page. | The data could be found by filling in the following search conditions: project ID and project’s name. |
|  | Test if the textboxes allow the user to enter information and if the select lists allow the user to choose project and employeeon insert new task page. | The textbox could be inserted, and the project name and employee name select lists would include the projects information and employees’ name. |
|  | Test the user insert new task function on insert new task page. | The new task could be inserted. |
|  | Test if the user could search for task’s information on manage project page. | The data could be found by filling in the following search conditions: task ID, task’s name and employee’s name. |
|  | Test the user edit task function on manage project page. | The task could be updated. |
|  | Test the user delete task function on manage project page. | The task could be deleted. |

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.

P 31 Each post is considered as a collection, and the post mostly contains same information such as owner, post content, comments.

### Project Description

The aim of this practicum project is to develop a system to manage employees in the software development department, organize Suplus Technologies project information, and manage 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 a management system is that it permits its users to communicate efficiently, therefore a key function of this system is that users can exchange messages, post status, 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 task information. Secondly, the system ensures that the administrator is able to organize employees in the software development department, and also customer and project information such as adding, editing and deleting customer information. 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. P1

## 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 of the employees in the software development. At the same time, Suplus Office Management System provides a platform that allows employees to exchange information and share their personal interests. P1

## Conclusion

Overall, this project has met all the requirements of my sponsor. Suplus Office Management System uses prototype development methodology to make the system organized and functional. This system is using PHP, HTML, CSS as major programming languages, and adapting JQuery, JSP and Bootstrap to enhance the user interface layout. Meanwhile, there is a social network page maintained by mango database to allow all the users to post status (message or pictures), and comments.

In this system, two databases are adapted: MySQL and NoSQL (MangoDB). 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, MongooDB (NoSQL) collects data from the social media website page-communication, which would allow users of this office management system to post status, pictures, and comment others’ status 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, Java script, JSP, and bootstrap 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, the designer conducted user and function tests to ensure that all the functions designed are working smoothly. In the user tests, 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 function tests, using the test result, the designer debugged and fixed unfeasible functions by changing codes, and using Bootstrap to redesign user interface. In conclusion, this practicum project perfectly satisfied the functions required by the Suplus Company.

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

https://books.google.ca/books?id=Bj--vAEPQDUC&printsec=frontcover&dq=Mongo+database&hl=en&sa=X&ei=56EXVbfoEsqpogTh04HoBg&ved=0CEAQ6AEwAg#v=onepage&q=Mongo%20database&f=false

## 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, this system could be used by employees in the finance department. 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.

## Innovations and Chosen Technology

### 10.1 Technology Chosen

### Why chosen Mongo database?

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

Loading Balance: could run over multiple servers, balance the load, and duplicate data to keep the system running in case of hardware failure.

File storage: Instead of storing a file in a single document, Mongo database divides a file into parts or chunks, and stores each of those chunks as a separate document.

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 website. As mentioned before, relational database requires stored data to be structured and predefined. Thus, it is difficult to insert an image or manage image data for relational database. In response to this difficulty, NoSQL database is adapted in order to store and manage image data. It is really easy to insert an image document into MongoDB because NoSQL is a document-based database.

Compared to relational database, NoSQL database is a set of tables containing data fitted into existing categories. Nosql database 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 (Leavitt, 2010).

**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 perform 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.

**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 vendor.

### Although NoSQL is not a mature technology, it could help increase the performance because the data is not in a structured form. The technology provides a new environment to develop data and system.

### PHP

PHP code can be simply mixed with [HTML](http://en.wikipedia.org/wiki/HTML) code, or it can be used in combination with various [engines](http://en.wikipedia.org/wiki/Web_template_system) and [web frameworks](http://en.wikipedia.org/wiki/Web_framework). PHP code is usually processed by a PHP [interpreter](http://en.wikipedia.org/wiki/Interpreter_(computing)), which is usually implemented as a web server's native [module](http://en.wikipedia.org/wiki/Plugin_(computing)) or a [Common Gateway Interface](http://en.wikipedia.org/wiki/Common_Gateway_Interface) (CGI). After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page. For example, PHP code can generate a web page's HTML code, an image, or some other data. PHP has also evolved to include a [command-line interface](http://en.wikipedia.org/wiki/Command-line_interface) (CLI) capability and can be used in [standalone](http://en.wikipedia.org/wiki/Computer_software) [graphical applications](http://en.wikipedia.org/wiki/Graphical_user_interface).

### 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.

As mentioned above, the user interface should be kept as simple, basic, and functional as possible. This website should have a consistent style so that it would not puzzle the user. The purpose of this project is to help improve employees’ performance and organization when they are working on projects.

For a better user-friendly and functional interface design, the system adapts the CSS, PHP, JQuery, JSP, and Bootstrap. The original version of the website, which only includes CSS and HTML, looks too simple and it is not user-friendly. By using JQuery, JSP, and bootstrap, the interface in the new version has been improved.

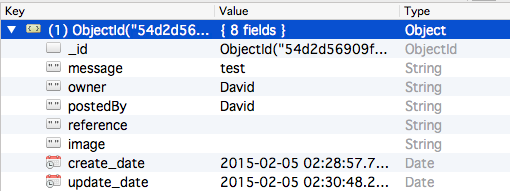
### NoSQL

As mentioned before, there is a social network page called communication in this system. Mango database maintain the data of communication page.

The reason of choosing Mongo database manages communication data is mango database based on document to store data. NoSql database stores and organize data as collections of documents, which could provide the data model better fit in data structure requirement than relational database, which means saving time to mapping data between data structure and relational database. Communication page includes users’ (the employees are involved in this system) huge amount posts, which could be a simple message, a picture or even an article. NoSQL database provides schema-less data storage, in this situation, using NoSQL database store and manage posting data is more efficient than relational database. (Sadalage & Fowler, 2013).

The more detailed information of choosing NoSQL and mongo database, please check 11.Research.

Communication page allow user to post status/picture, and allow other user to comment it. Simply consider each post as a collection, and the post mostly contains same information such as owner, what the post it is (message or picture), any user commented it or not. If choosing relational database to store these information, it requires huge amount data mapping and cost more time. However, NoSQL database could fit in data structure without mapping data around, and organized the each post data (colletion) together and store data by a default structure. In another word, the each post data volume of communication page could organize as ID, owner, postedby, reference, image, create date and update date. The following picture shows the data structure.



ID: As same as relational database, when user save a new data, Mongo database automatically attaches a unique identifier to it. Each collection has its personal and unique ID. ID helps to identify each single post.

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

Postedby: This is the user commented an original post (message/picture), It could be same user, because user has right to comment personal posted message.

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

Image: For here to store 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.

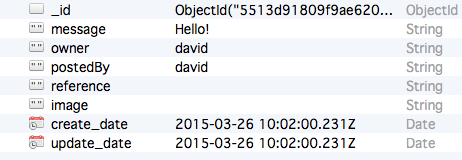
In the following part, screenshots of user testing could explain more of the data design.

#### 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.



The following picture presents the inserted collection in mongo database.

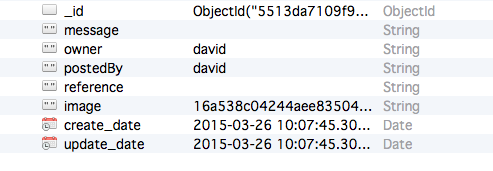


#### Post picture

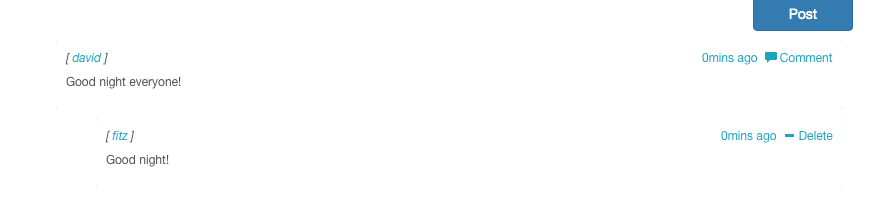
In this case, user David posts a picture on communication page.



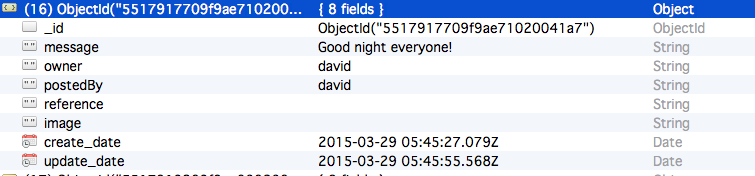
The owner and postedby is still David, because he is the person post the picture and this is an original comment. The following picture shows the image ID after post an image on communication page. In this case, an image has been inserted.



#### Comment

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

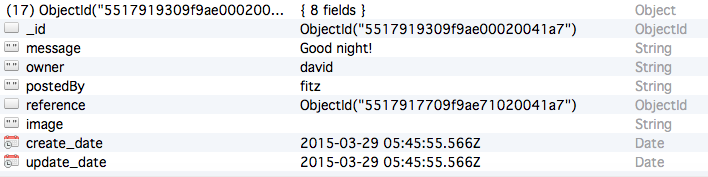
The next picture demonstrates the original post data. David is the owner of the message and the message postedBy David too. The update\_date is 05:45:55.568Z.



The following picture is the Fitz comment’s data 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 changed from



After Fitz made a commnet on David’s message, update\_date of David’s original message has been changed to Fitz commented time.

Small question:

In this system, most information has been presented in table, so it was difficult to demonstrate all the information in the page nicely. P52 11.1

When user click browse button, system should open a window allow user to choose picture from local direction.

Comment should be post when user hits enter from keyboard.

Words like using and adapting

Scale: The hours the tester spent to operate employee module by task list.

Test: The 5 testers were invited to test employee functions. They are all given the existing user name and password to login the system. The duration of operation is timed according to task list has been mentioned above.

Test scenario: The testers login as employee account, and go through all the functions provided by the system.

Worst case: The testers spent more than an hour to complete the entire employee module.

Best case: All the functions are workable. The testers spent less than a half hour to completed employee module.

Planned: The testers took 20-55 minutes to finish employee module.

All user test case

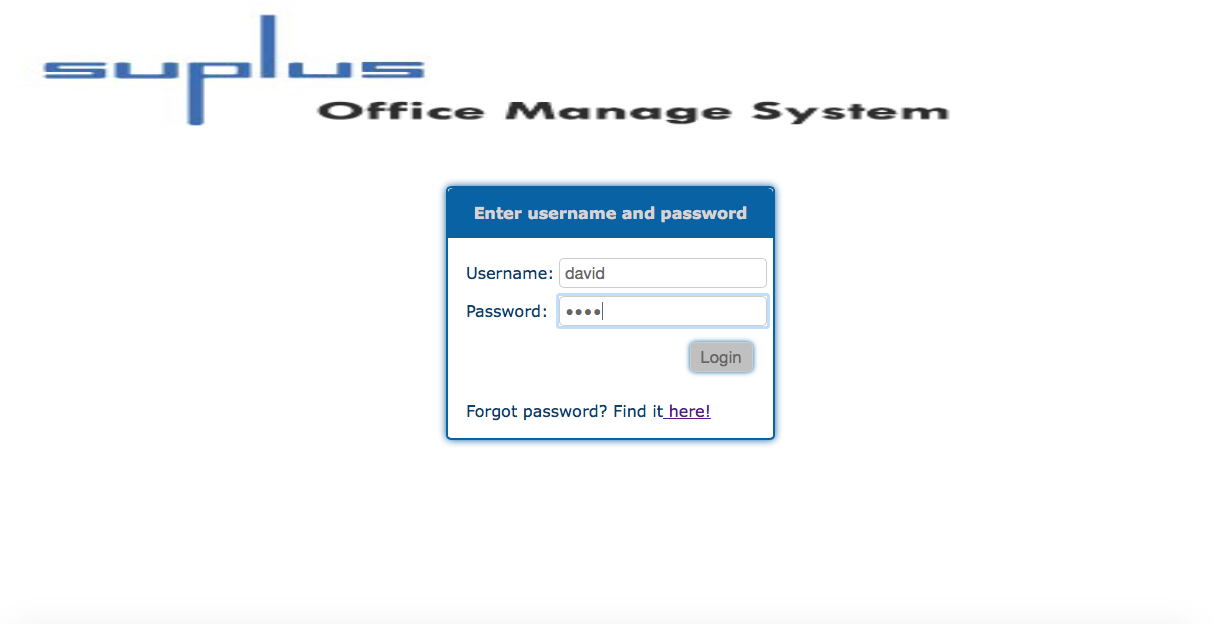
Login

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.

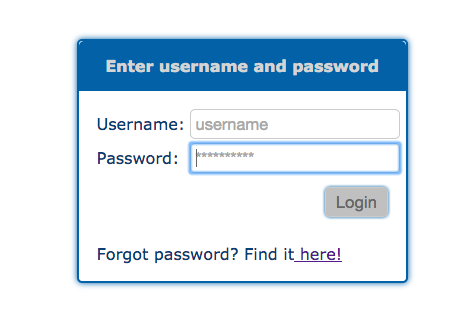
1. Test that an error message presents to indicate user or invalid data or empty blank.

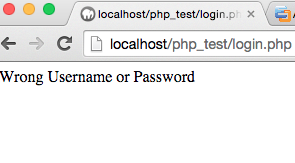
Test Scenario T2:

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:

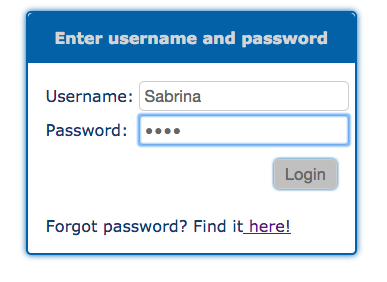
First time:

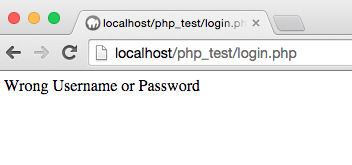




Login system without enters any username or password, could not login to system.

Second time:





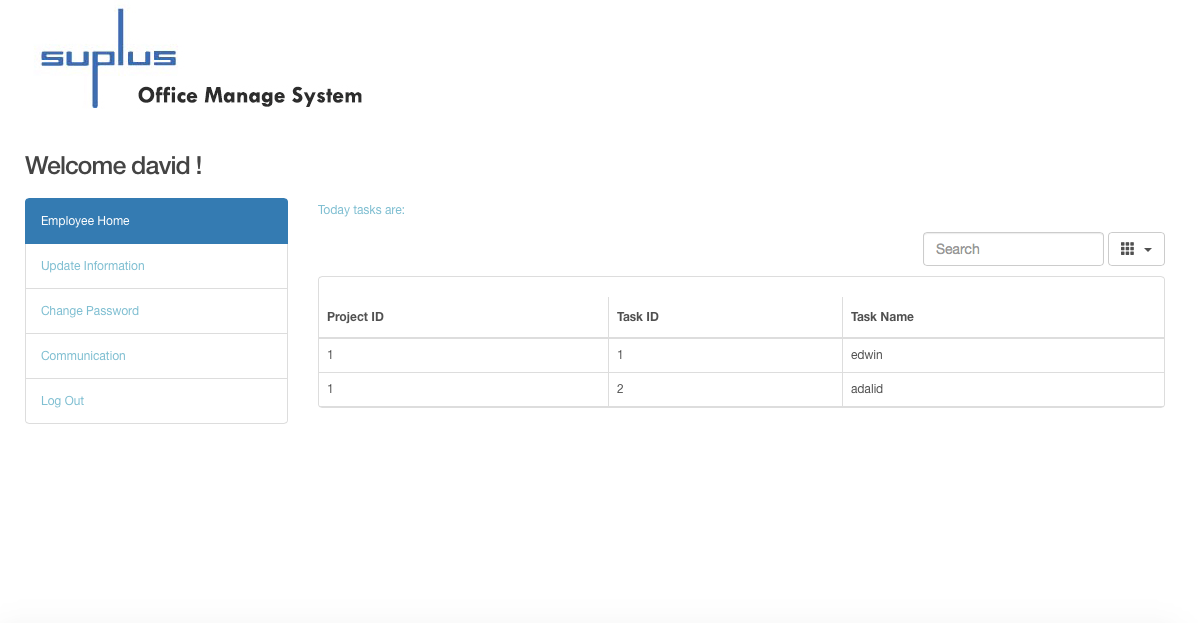
Login system with not existing password and username could not login as well. The test is PASSED.

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.

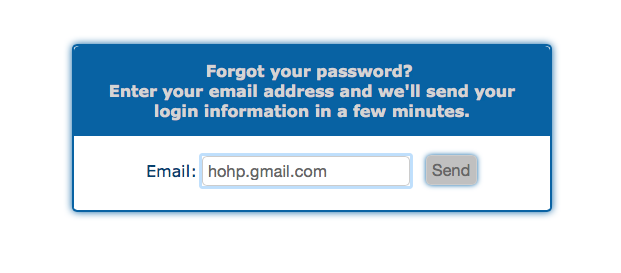
1. Test that the system could send email with password to binding email address when user clicks the forgot password button on login page.

Test Scenario T4:

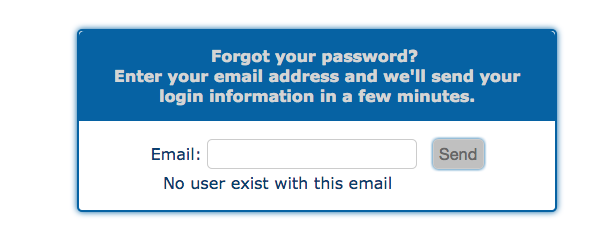
1. Click the forgot password link on the login form.
2. First, input not existed or invalid email address.
3. Second, input existed and valid email address.
4. Click login button.
5. Check email inbox with the new letter with password or not.

Test result:

First time:



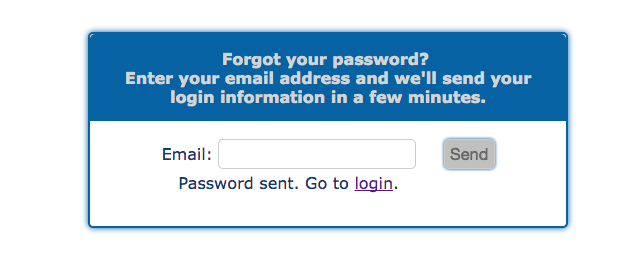
Click send button with percious uncorrect email address, system reminds no exist user.

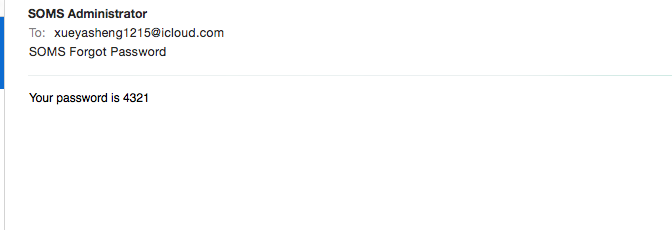


When enter not correct email address, system could not connect database with this invalid information.

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

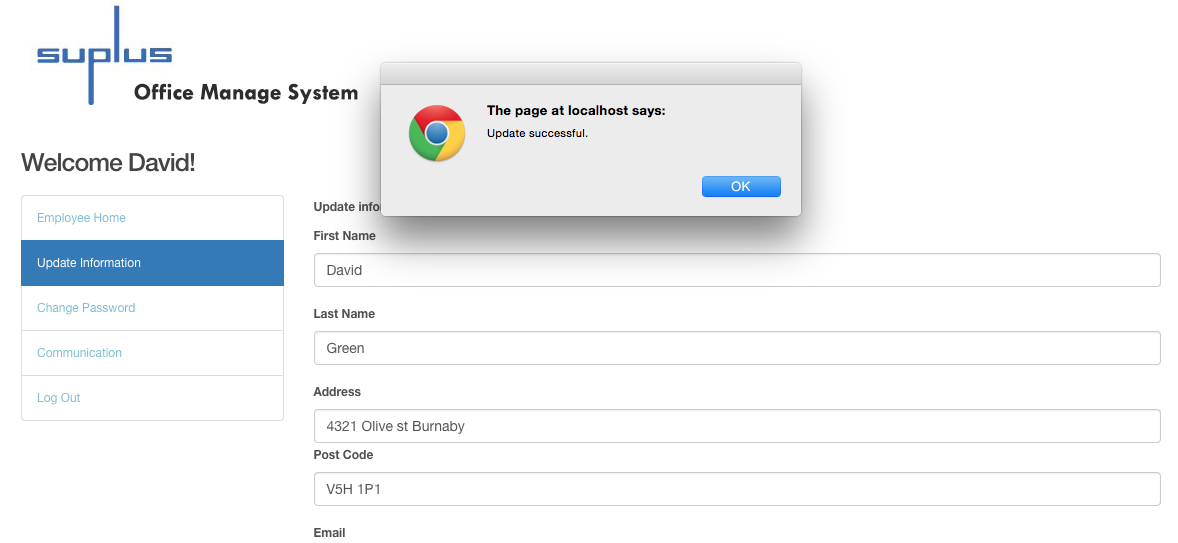
1. Test that the user could edit personal information on update information page.

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.

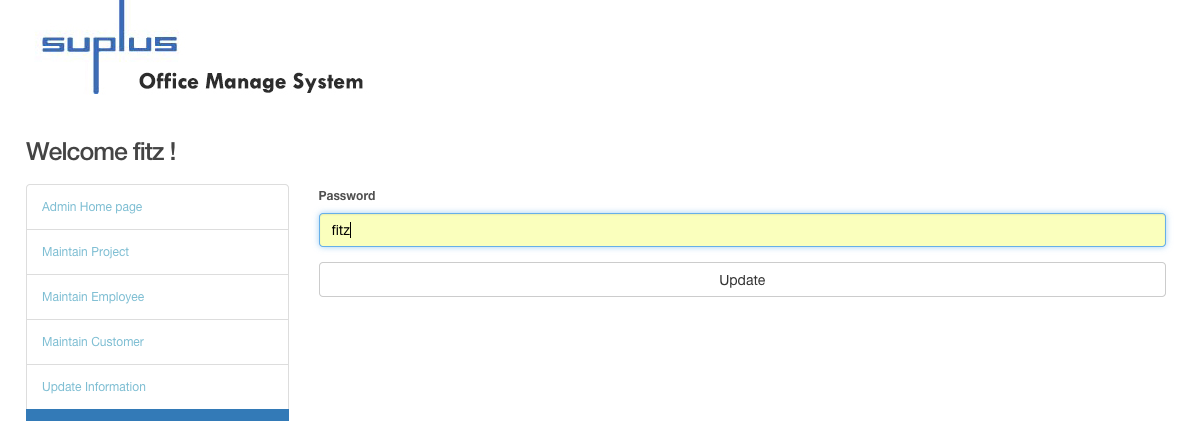
Change password

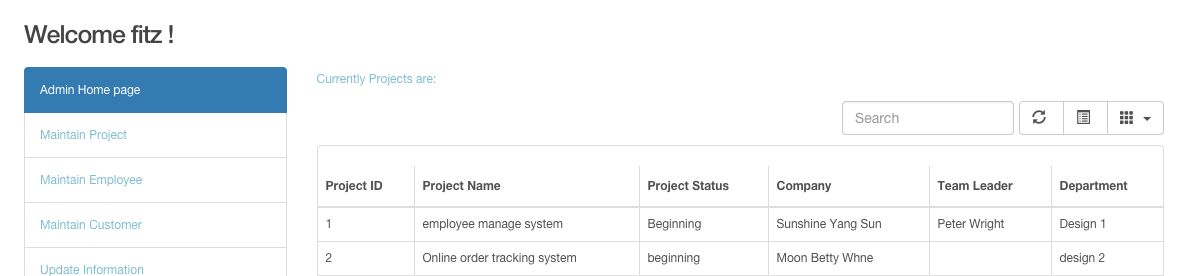
1. Test that the user could change password once have signed in the system.

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.

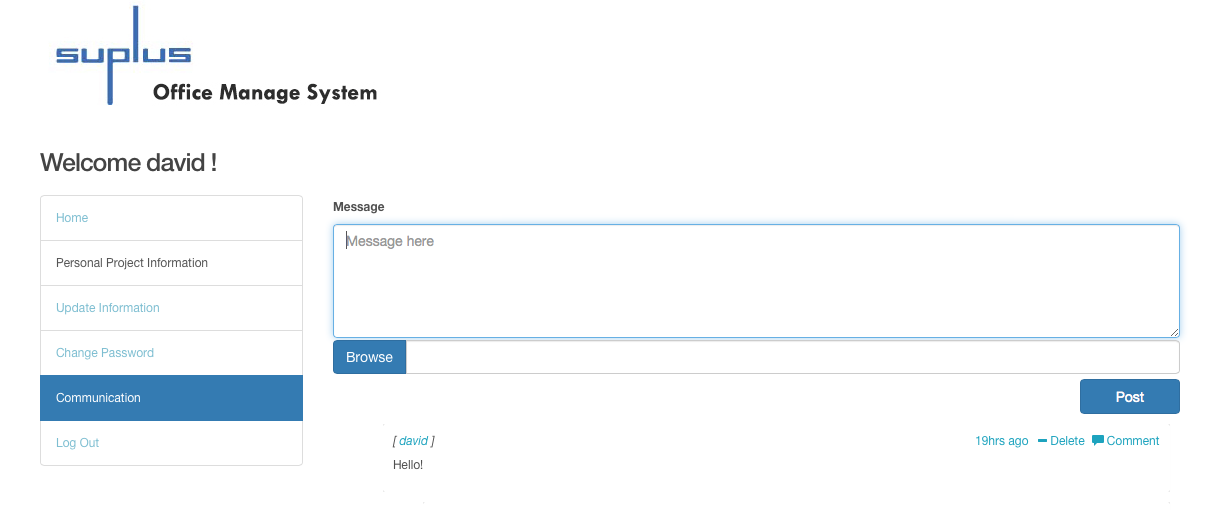
Communication

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:

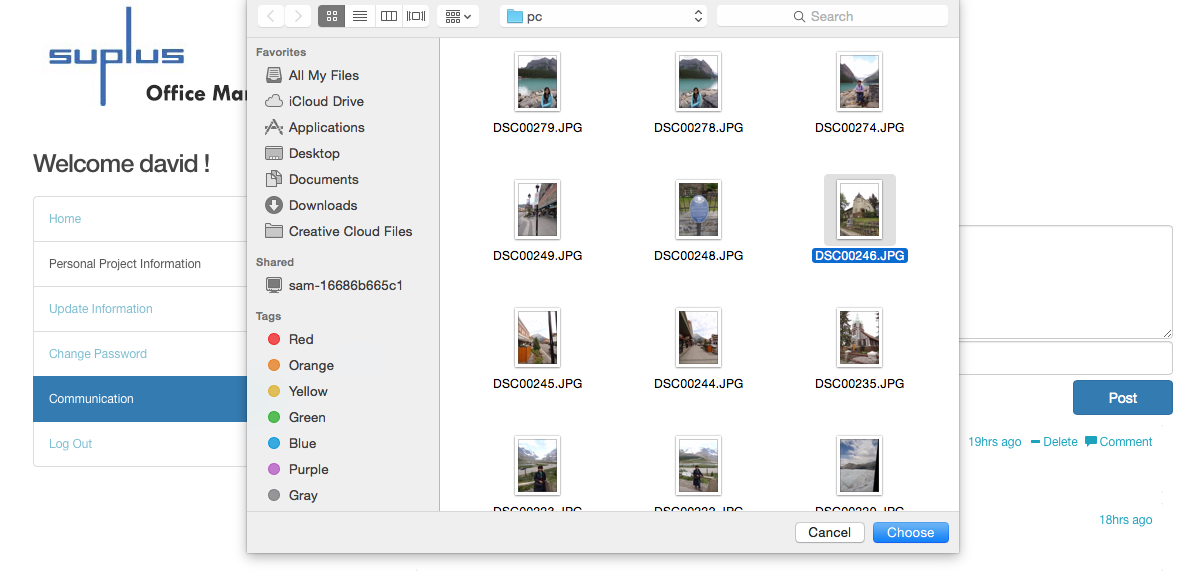


Message posted successfully. The test is PASSED.

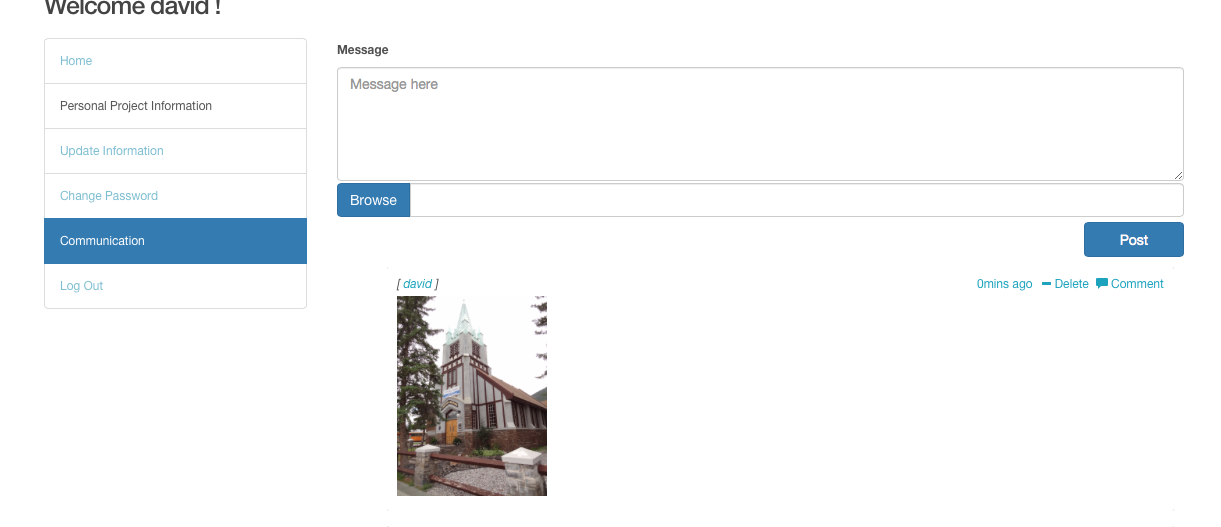
1. Test that the user could post a picture on communication page.

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.



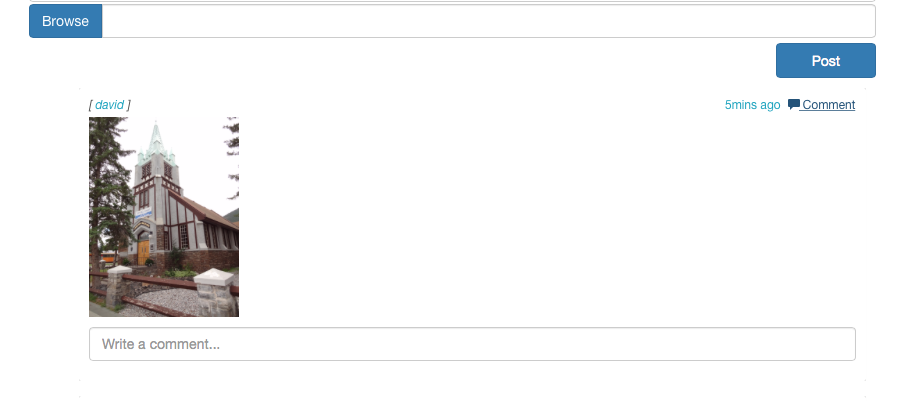
Picture posted successfully, the test is passed.

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 Module

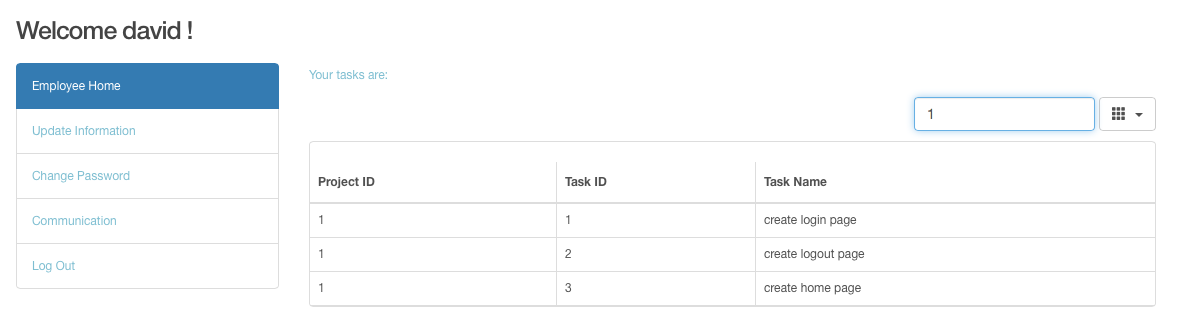
Check personal project information

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.

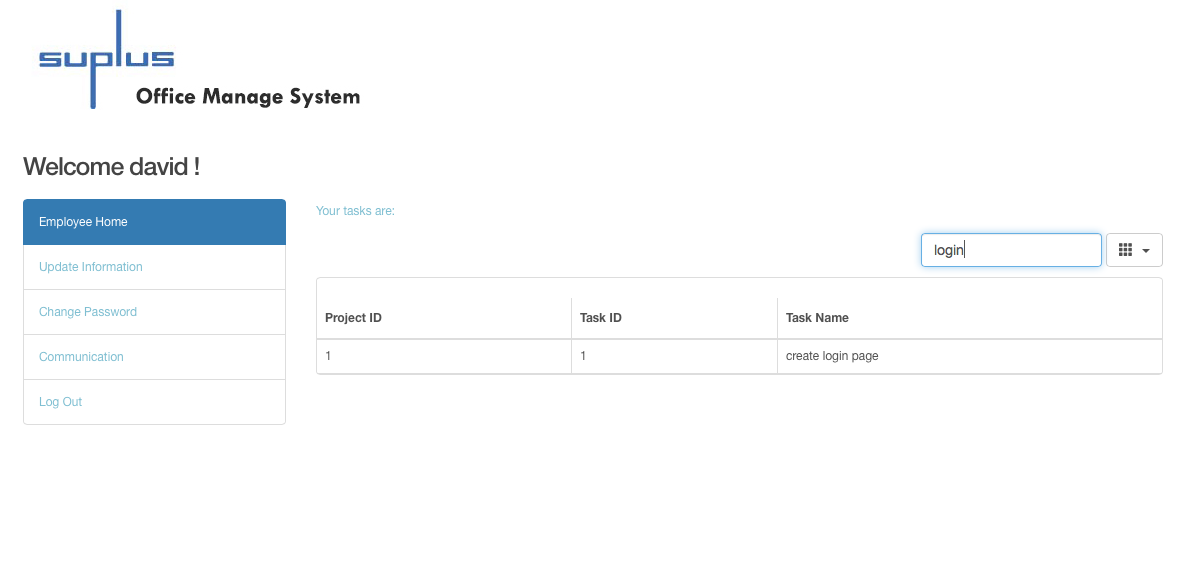
1. Test the search function on employee home page

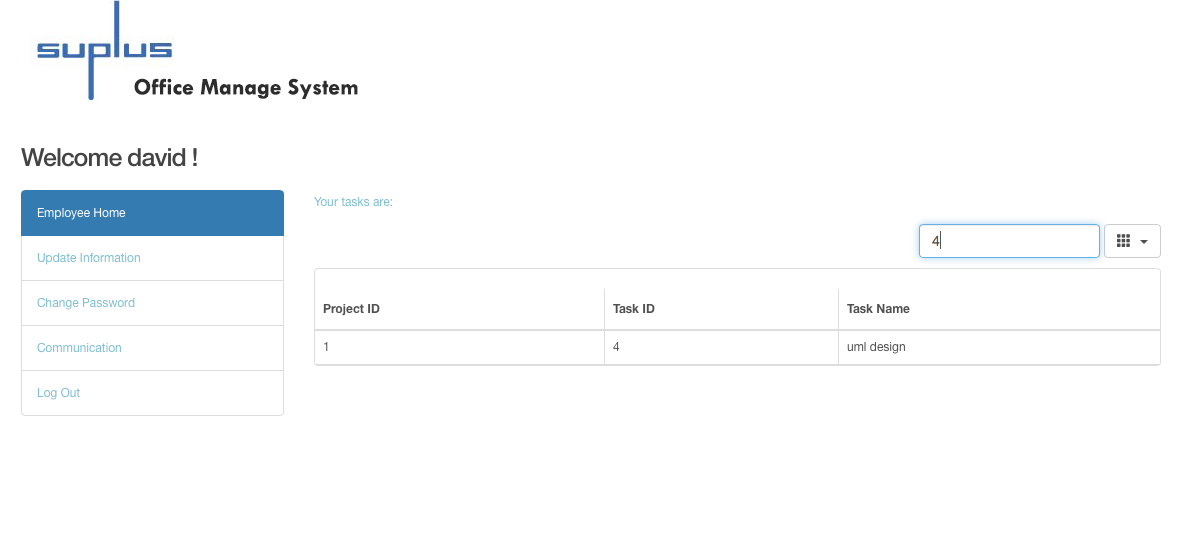
Test Scenario T11

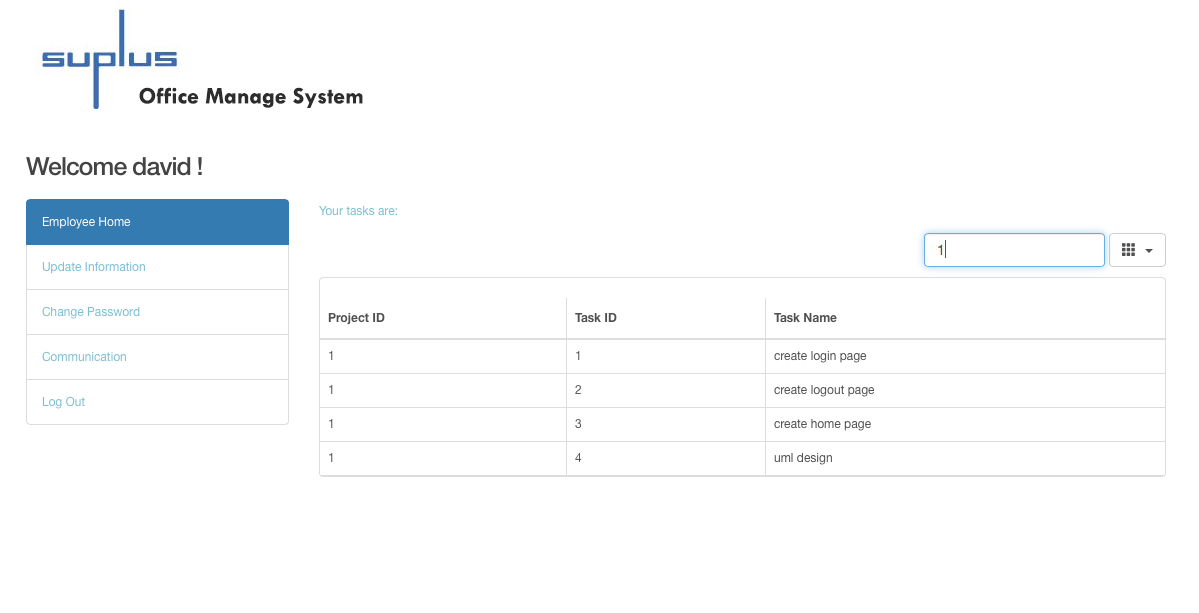
1. Enter valid task name in this case, enter “login” as task name.
2. Enter valid task ID, type in 4 as task ID in the search bar
3. Enter valid project ID, in this case, search related information when project Id is 1
4. Enter not existing information, such as “rest”

Test result:

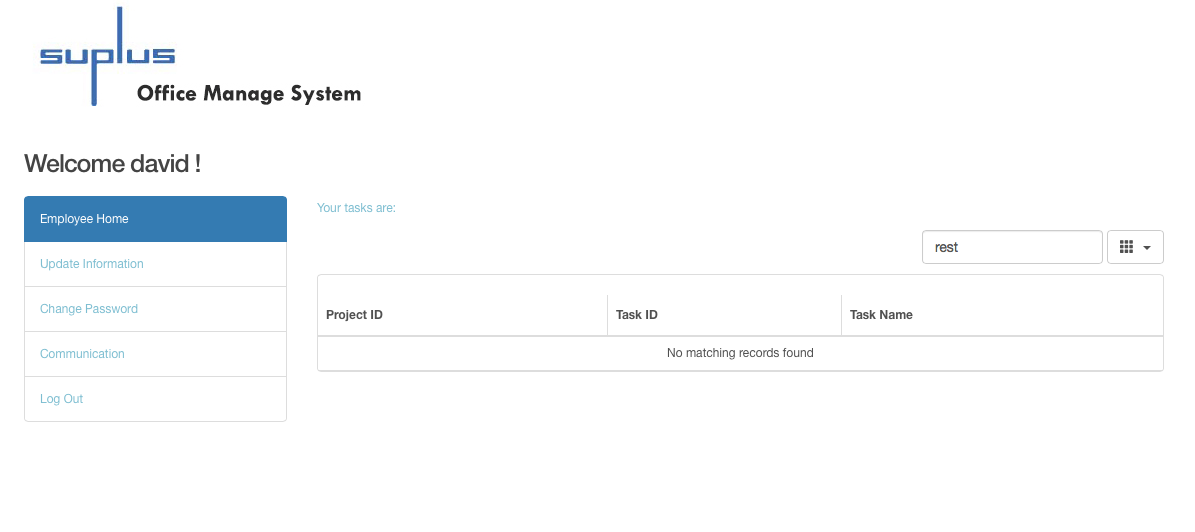
Search by task name login:



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 Module

Home Page

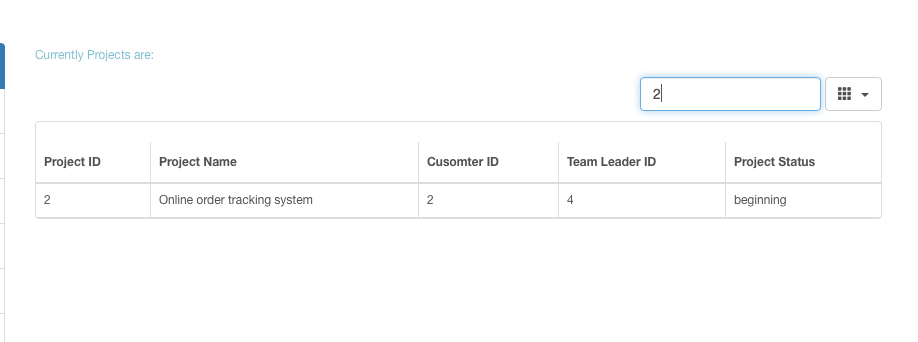
1. Test that the administrator search function on admin home page**.**

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.

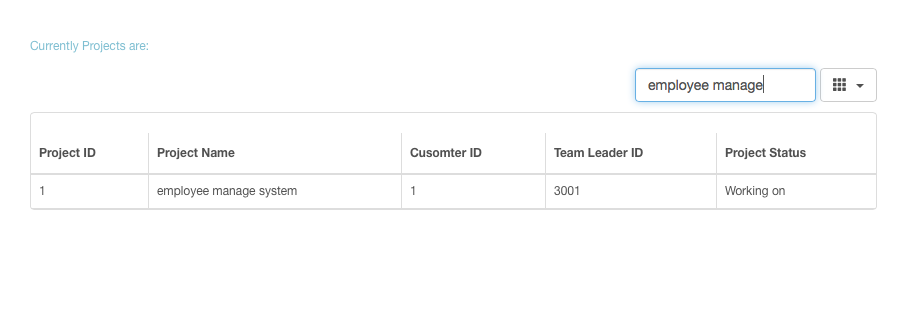
Test result:

Search project information when project ID is 2:



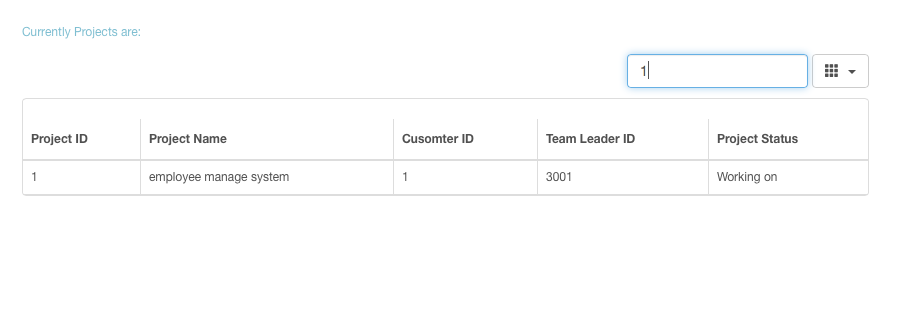
Typed in 2 as project ID, found a record, test is passed.

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.

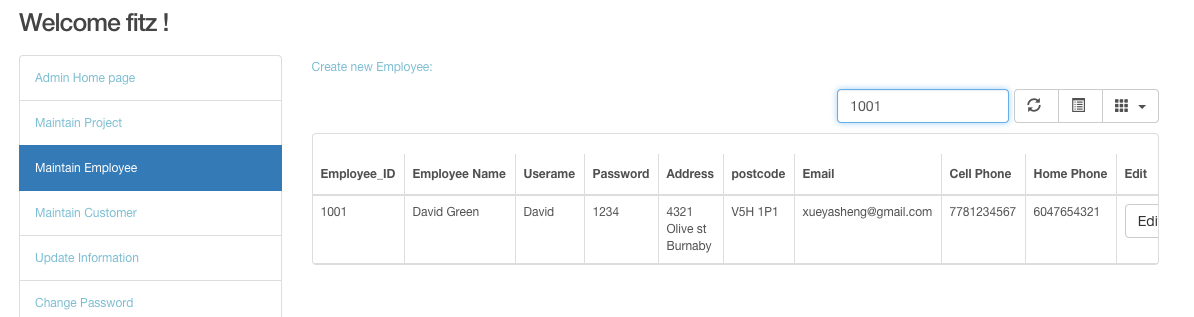
Maintain Employee

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

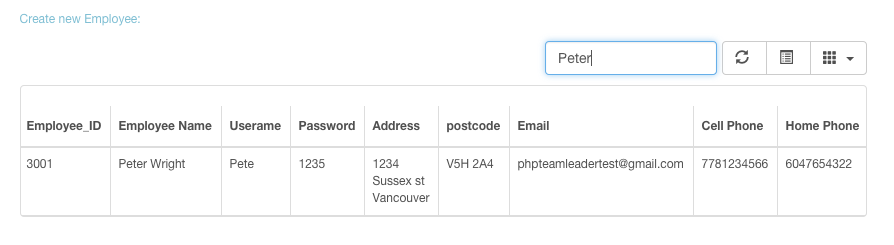
Test Scenario T13:

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:



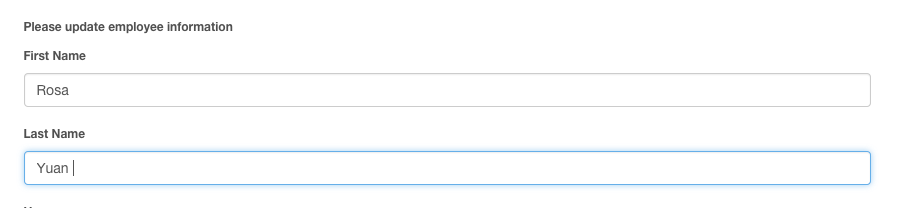
The test is PASSED.

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:



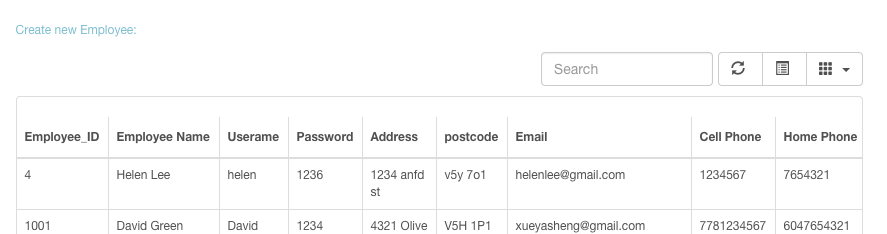
Employee name has been changed. The test is PASSED.

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

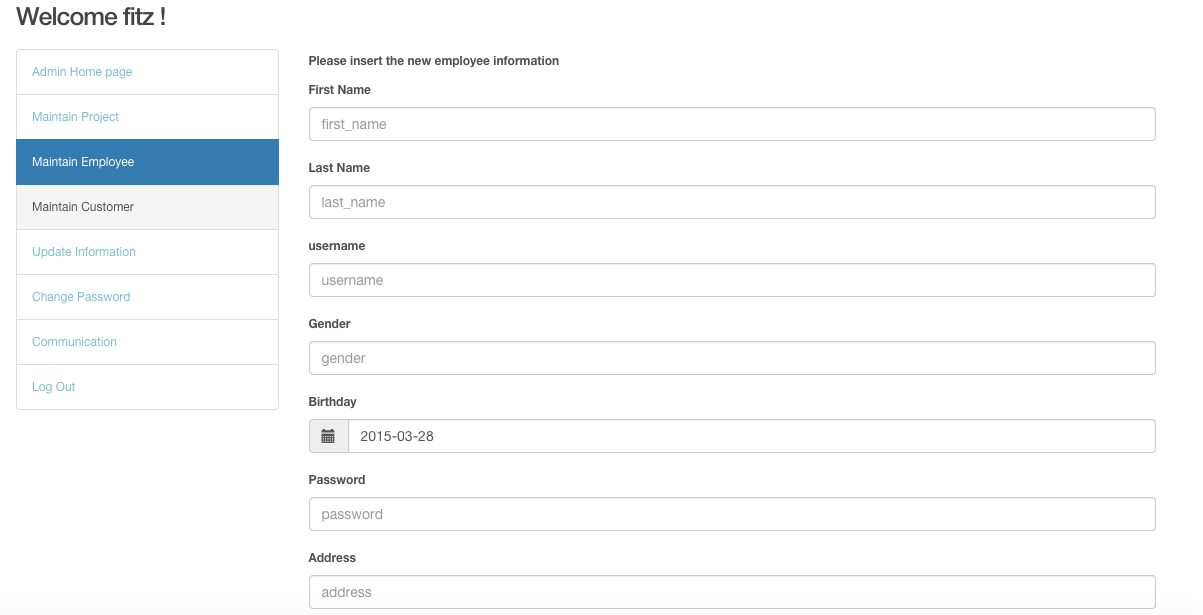
Test Scenario T15:

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

Test result:



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



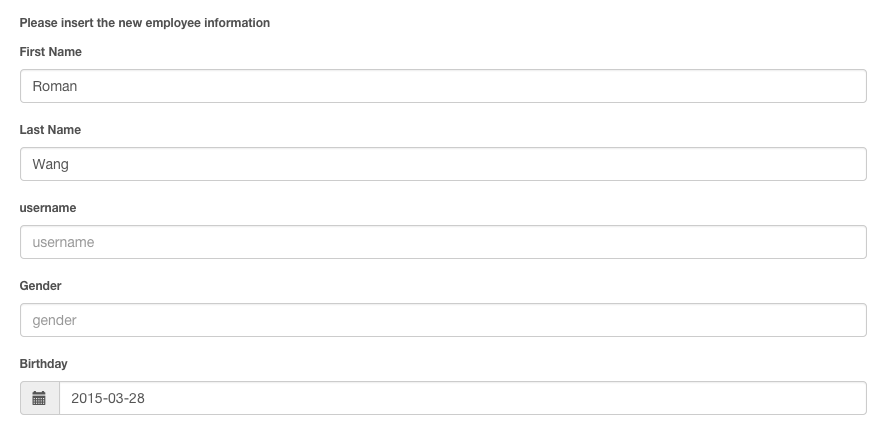
The link located to add new employee page. The test is PASSED.

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.

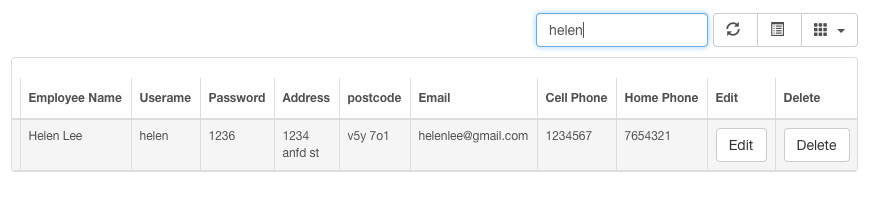
1. Test that the administrator delete employee function on maintain employee page.

Test Scenario T18:

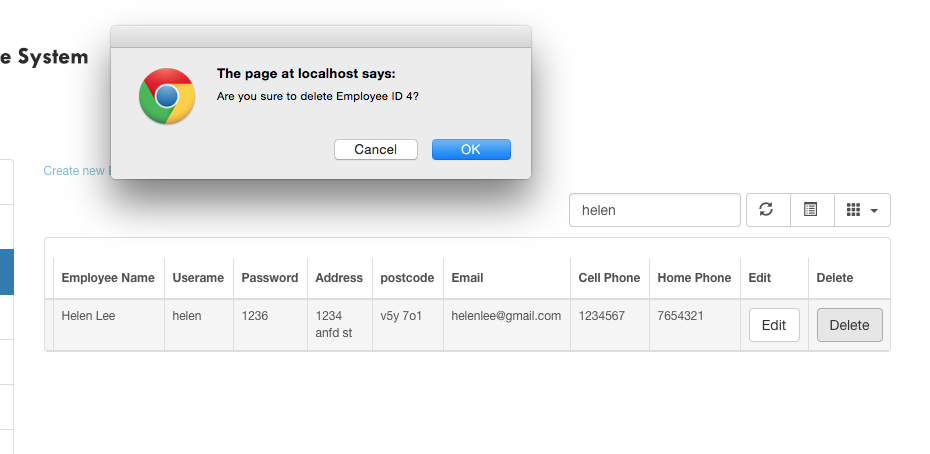
1. Search employee information whose name is Helen.
2. Click delete button.
3. Click ok button on are you sure to delete information pop-up window.

Test result:

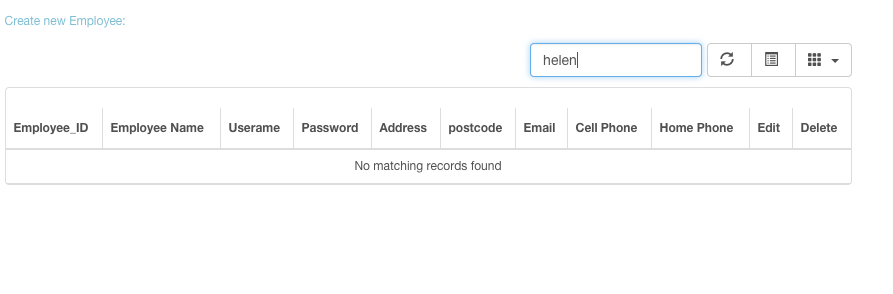
Search Helen and click Delete button:



Click OK button on pop-up window:



Search Helen again, could not found data.



The information has been deleted. The test is PASSED.

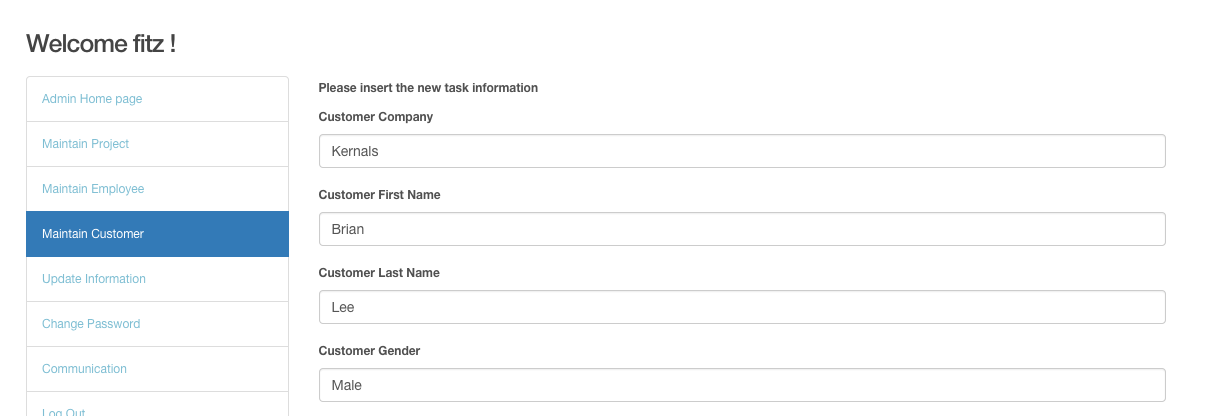
Customer Maintain

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:



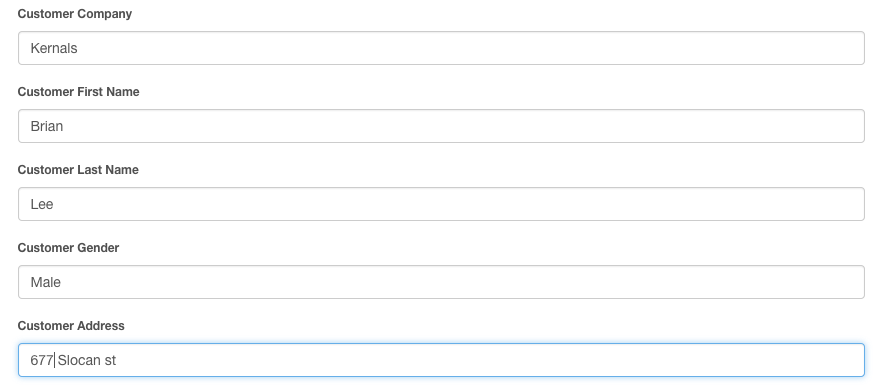
The picture above presents a new customer has been inserted. The test is PASSED.

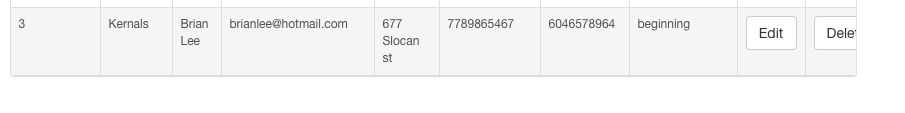
1. Test that the user edit customer information function on maintain customer page.

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.

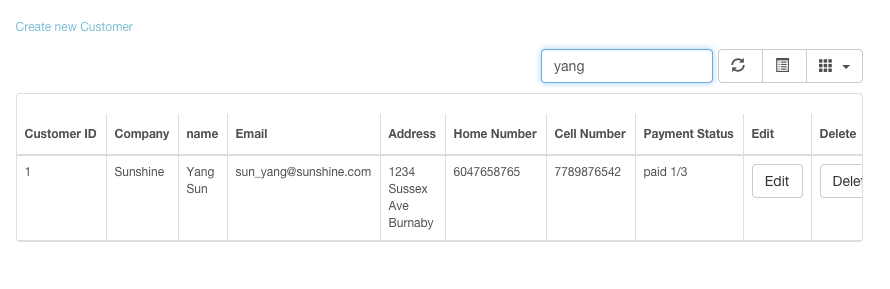
Test Scenario T21:

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

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

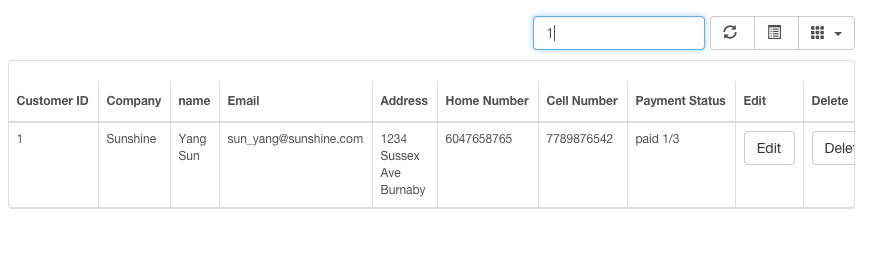
Test result:

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.

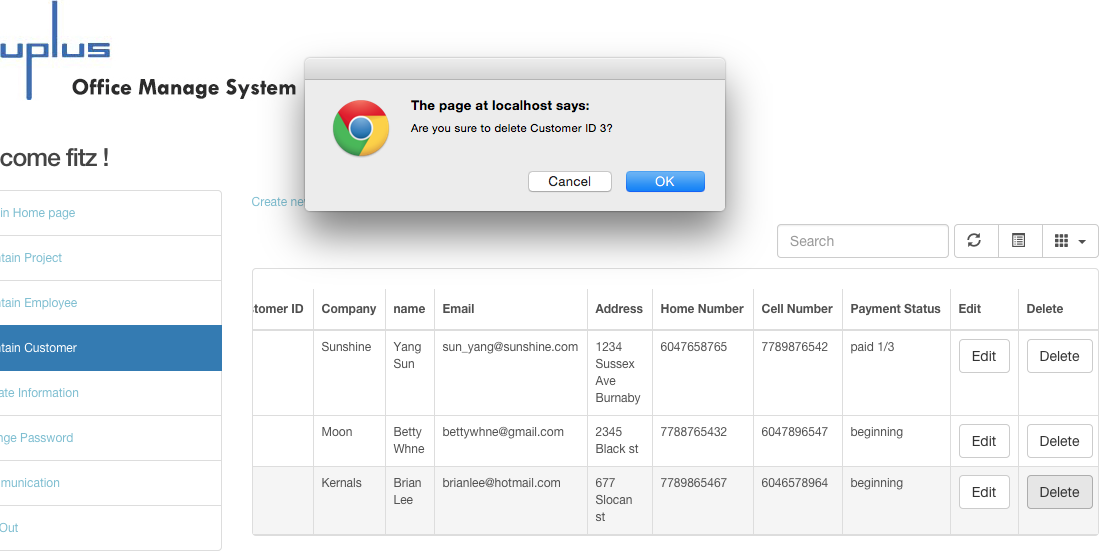
The test is PASSED.

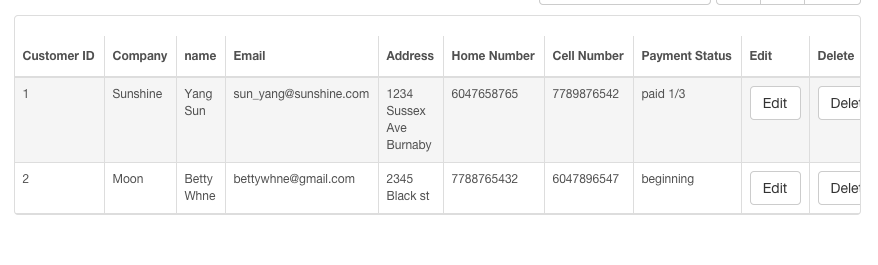
1. Test that the user could delete customer information on maintain customer page.

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.

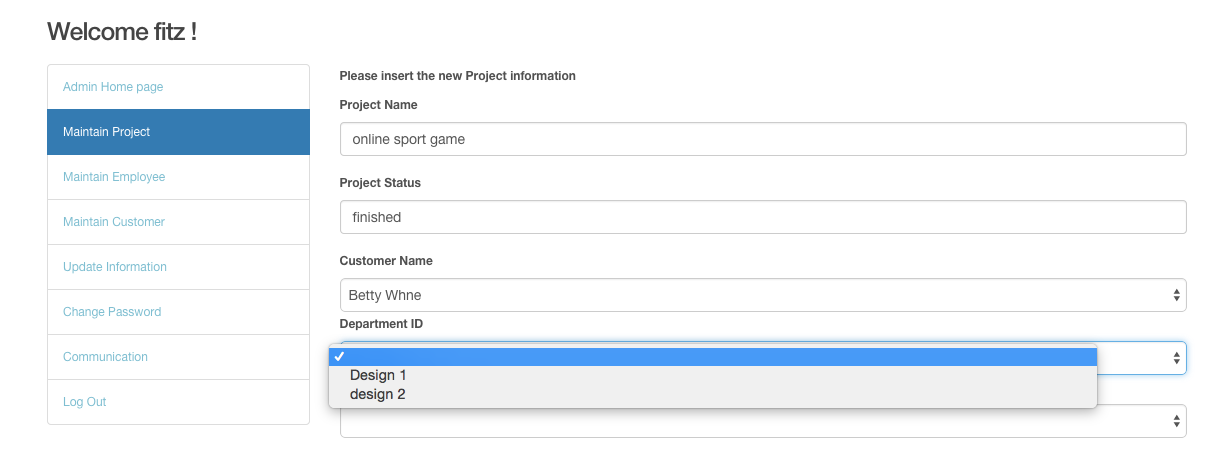
Maintain Project

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.

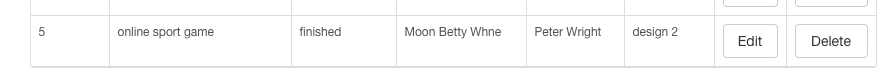
1. Test that the user could create new project.

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.

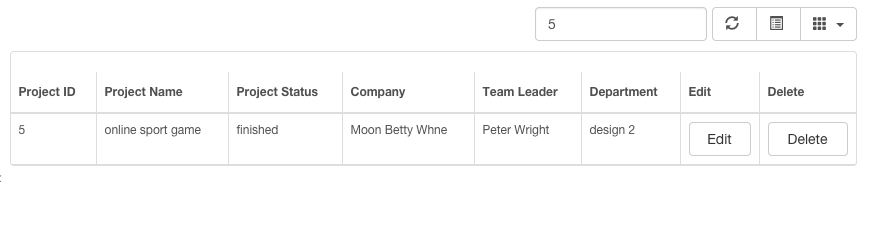
1. Test that the search project function on maintain project page.

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.

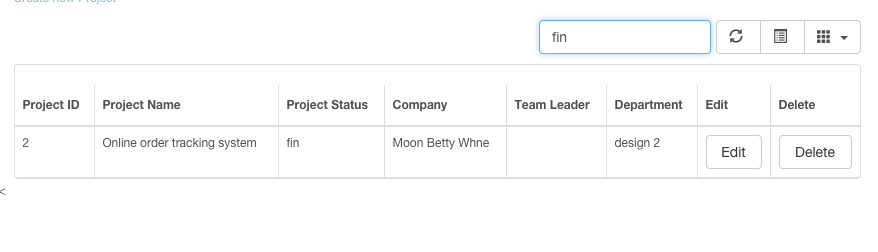
Test result:

Search project ID as 5:



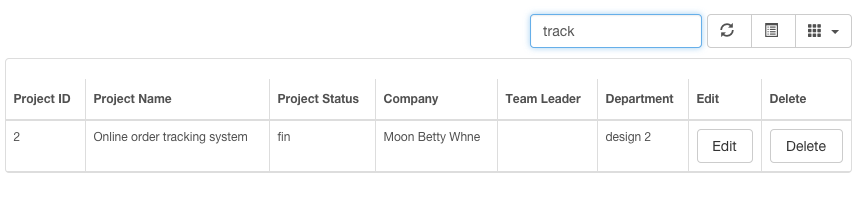
The data could be found.

Search project status as fin:



The data could be found.

Search project name as track:



The data could be found.

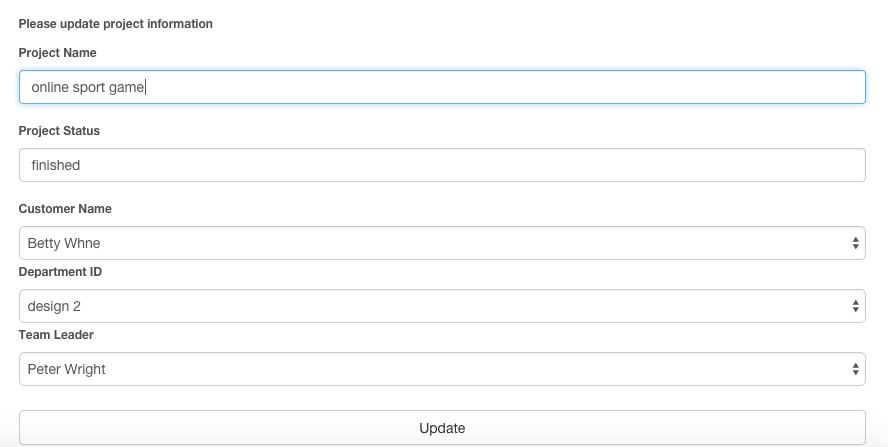
The test is PASSED.

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:





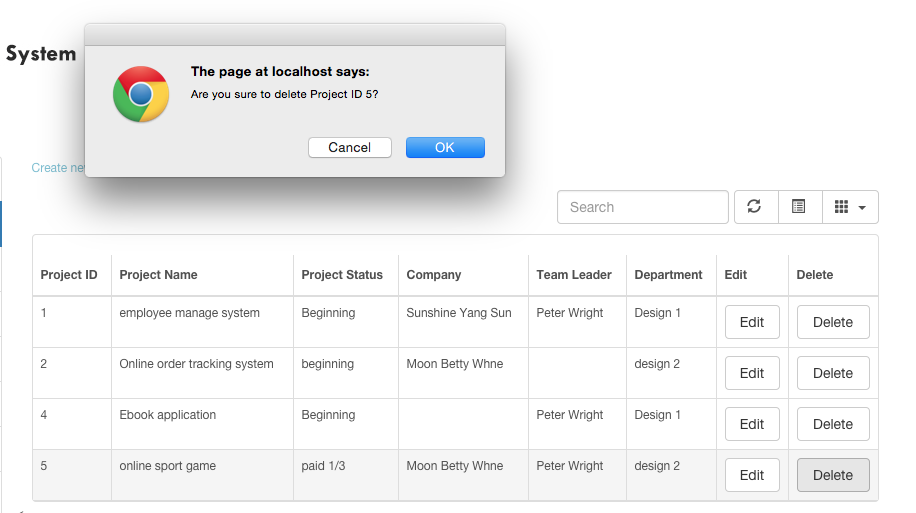
Project status has been changed. The test is PASSED.

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

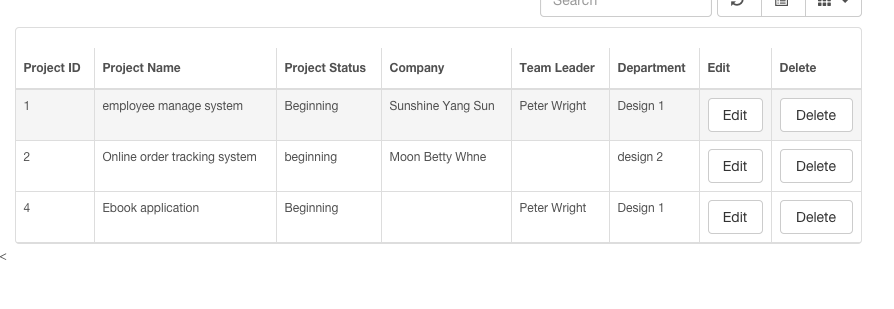
Test Scenario T27:

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 module

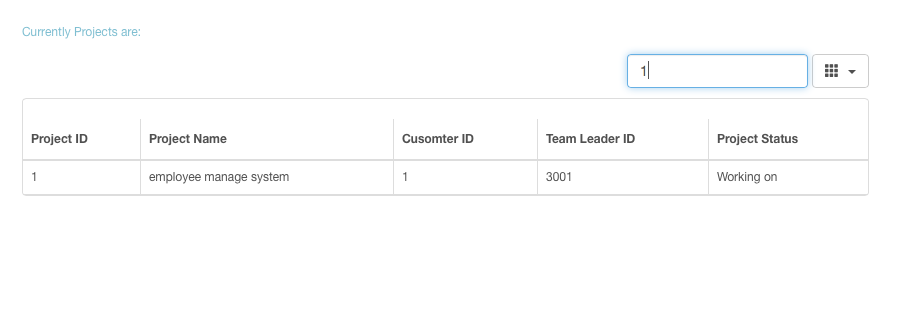
28. Test that the team leader could search project information on manage project page.

Test Scenario T28:

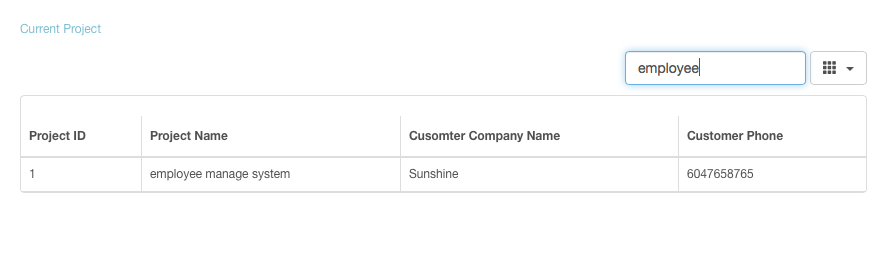
1. Search project information by enter project ID as 1.
2. Search project information by enter keyword employee of project name.

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.

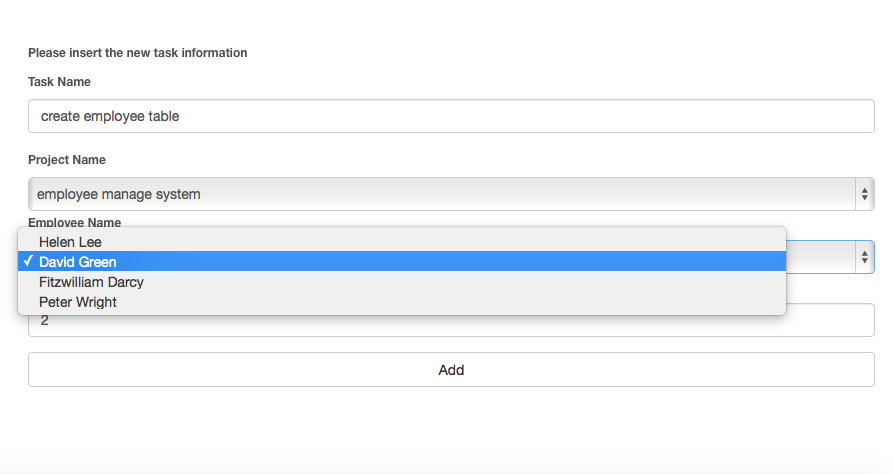
Manage project

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.

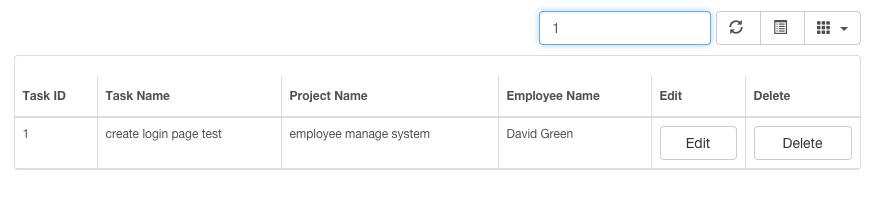
31. Test that the user could search task information on manage project page.

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.

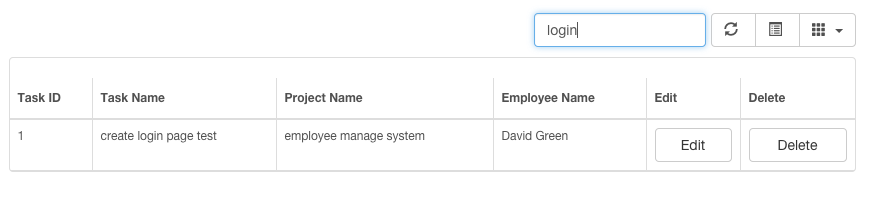
Test result:

Search task information by enter task ID as 1:



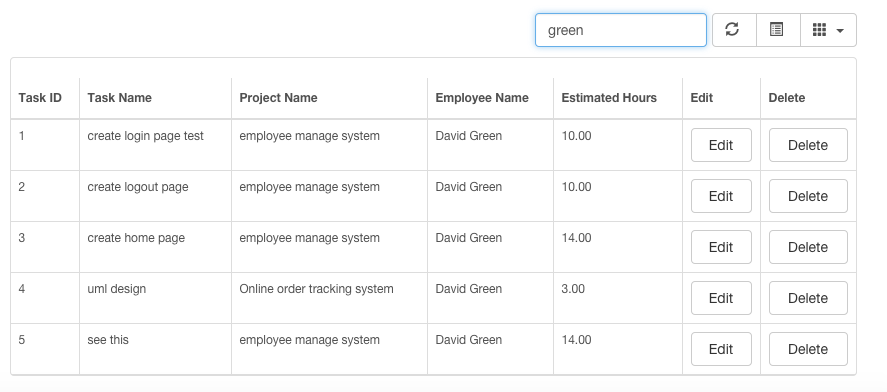
There is a data has been found. The test is passed.

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.

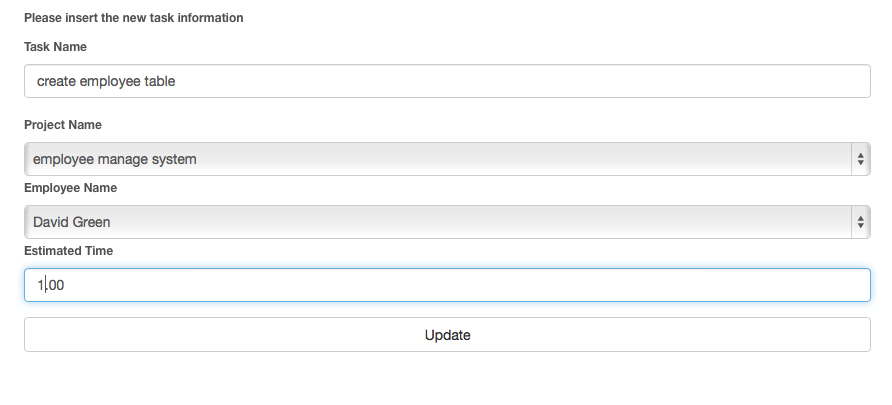
32. Test that the edit task function allows user to update existing task information.

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.

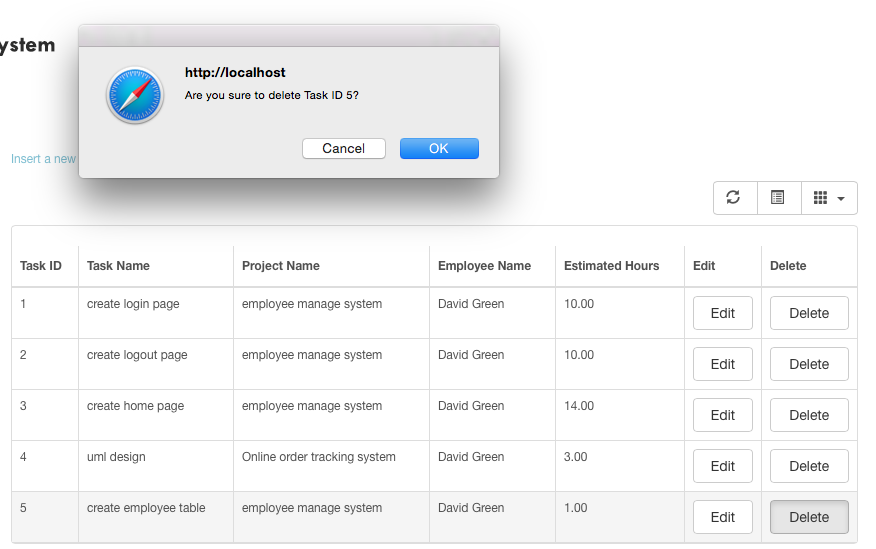
33. Test that the delete task function allows user to delete existing task information on manage project page.

Test Scenario T33:

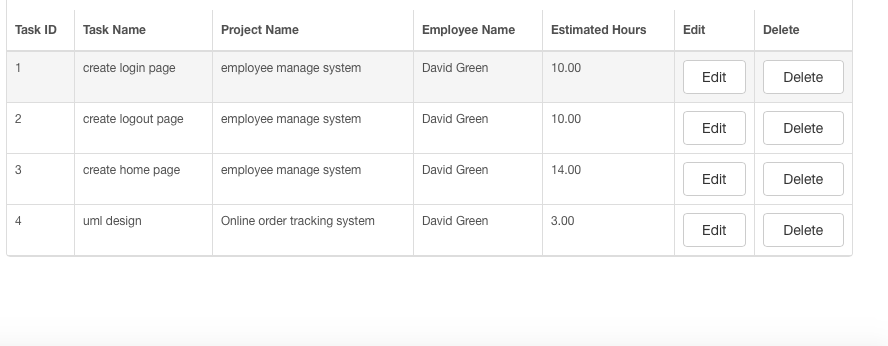
1. Choose task “create employee table”.
2. Click delete button.
3. There is pop-up window to ensure user decided to delete task.
4. Click Ok on the pop-up window.

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.