|  |
| --- |
| **FPT-aptech computer education** |
| eProject Document |
| Mantech Helpdesk |
|  |
| |  |  | | --- | --- | | **Group 1** | | | **Group Member** | Nguyen Xuan Minh, C00174, 3391  Bui Huy Phong, C00119, 0141  Nguyen Thanh Nam, C00078, 0192  Nguyen Duc Viet, C00167 , 0204  Dao Anh Toan, , 0159 | | **Instructor** | Nguyen Ngoc Tu | | **Batch** | C0812I | | **Semester** | 4 | |
|  |

|  |
| --- |
| - Hanoi, 08/2011 - |

# Introduction

1. Mantech Limited wants to implement Customer Helpdesk Management for Technical Support. It is encountered that there are too many requests. The registration of these complaints/queries is manual which involves filling of complaint forms and maintaining records on paper.
2. For the past few months they have been facing problems with this process. Due to the problems they have felt the need for a change in their system. It has been observed that there are a lot of Transactions that are happening daily which if automated would help them to serve their staff better and faster.
3. For this purpose, they have approached you to develop a system. They would like you to study the problem at hand, and evolve a strategy for solving the current problem.

# 2. Problem Definition

## The Proposed System

1. After conducting a deep study of the existing method of lodging complaints and keeping the documents in place, we have suggested a solution to the management.
2. We have suggested them to have a web-site, wherein the employees can lodge the complaints regarding any hardware or software problems in their machine.
3. All these complaints will be stored in a database and the administrator can see the complaints registered on daily basis and can allocate a technician to solve the complaint.
4. After solving the complaint/problem, technician will have to enter the details on the site and close the complaint.
5. Employees can lodge new complaints, can see the status of their old complaints on the site. Similarly administrator can also see the status of all the complaints on the site and can also change the priority of the complaints. Different kind of reports can be generated by the administrator.

## Boundaries of the System

* Create a website using JSF and Microsoft SQL server 2005.
* The website will have a web page which has links or option Home, My account, Departments management, Complaints management etc.
* Login page allow user login into the system
* Complaints management page allow user list all complaints, view complaint, sort etc.
* Departments management allow user to insert, disable and update department information etc.
* My account page allow user to change account information like full name, password etc.

## Development Environment

* Hardware and OS Requirements:
  + Pentium IV 2.0 GHz or later
  + 512MB of RAM or later
  + HDD: 4GB or later
  + Windows 2000 or later
* Software Requirements:
  + Internet Explorer 8.0 or other
  + GlassFish server 3.x or later
  + SQL Server 2005 or later

# Requirements and Business Flow

## Customer Requirement Specification

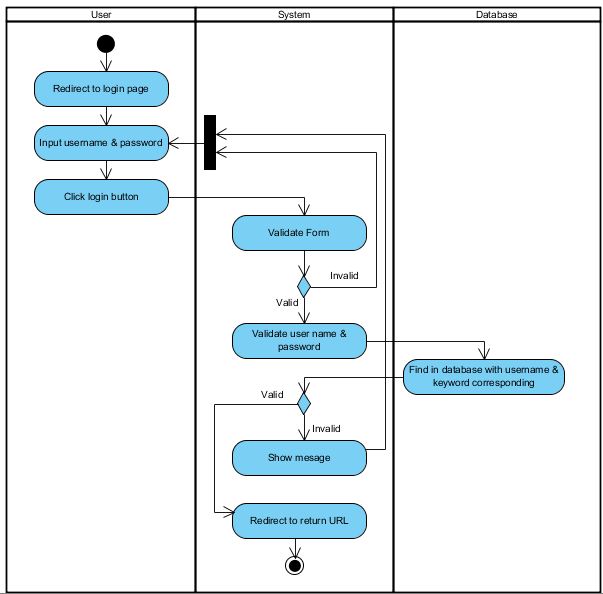
The system needs to be developed based on following requirements:

* A home page with suitable logo and a punch line. The page will provide interface for signing in to the account and register the complaint.
* There should be 2 different types of users for the site.

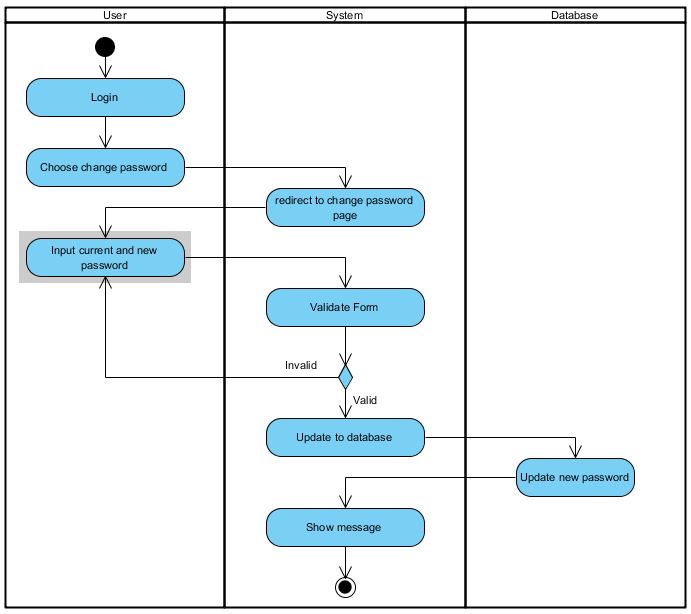
1. Administrator:
   1. Administrator will be creating the accounts for all the employees.
   2. Receive all the complaints lodged in the database.
   3. Sort the complaints based on the department.
   4. Assign the task to a technician based on complaint id
   5. Generate the reports:
   6. Daily, Weekly, and Monthly reports. These reports can be
      1. Summary report: The summary report will include the total no. of complaints logged in the particular period (department wise separated), status.
      2. Detailed report: This will include
         1. Complaint No
         2. Complaint category
         3. Date/Time of lodging complaint
         4. Date/Time of closing complaint
         5. Department of the user
         6. Name of the technician
         7. Actual time taken to solve the problem
         8. Employee no/name who registered the complaint.
   7. Status Report of the pending complaints.
      1. Based on number of days.
      2. Based upon the priority of the complaints.
   8. Department wise Report: This will include
      1. Complaint category
      2. No/Name of the employee
      3. Date/Time of lodging complaint
      4. Date/Time of closing complaint.
      5. Name of technician
      6. Actual time taken to solve the problem
   9. Technician wise report: This will include
      1. Complaint category
      2. No/Name of the employee
      3. Department of employee who registered the complaint
      4. Date/time of lodging complaint
      5. Date/Time of closing complaint.
      6. Actual time taken for solving the problem
   10. Complaint category wise report: This will include
       1. No/Name of the employee who registered the complaint
       2. Department of employee who registered the complaint
       3. Date/ Time of registering the complaint
       4. Date/Time of closing complaint
       5. Name of technician
       6. Actual time taken to solve the problem.
2. Registered User: A registered user is an employee for whom an account is created by administrator. With this account he will be able to :
   1. Lodge a new complaint: The complaint should be mailed at [complaint@helpdesk.com](mailto:complaint@helpdesk.com) with very short description of the complaint in subject line of the mail.
   2. View complaint log: The user will be able to see the previous complaint information like date/time of complaint, status (pending/closed), date/time of closing the complaint, actual time taken to solve the query, name of technician who solved the complaint.
   3. Re-send the complaint if it is pending for more than 2 days.
   4. Change the password for their account.

* In addition to the above mentioned, the site also provides knowledge base portal where technical articles are published every month. For time being you can include at least 5 articles. These articles can be on latest updates in technology or on new release of software versions or some tips n tricks for some programming languages etc. Also reader can rate the article.
* You can also include the Frequently Asked Questions(FAQs) which will help the user to access the site functions

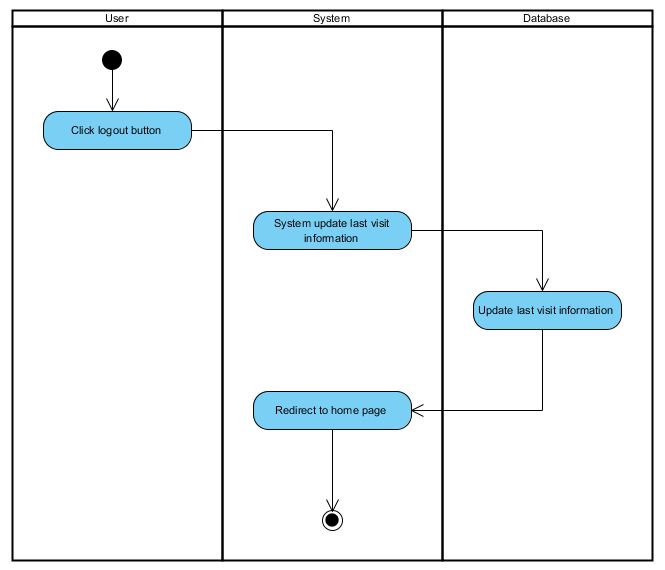
## Activity Diagram



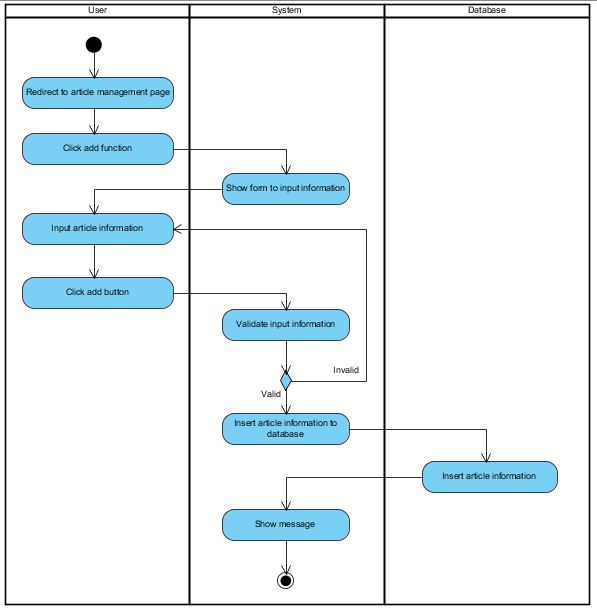
Admin, Employee login



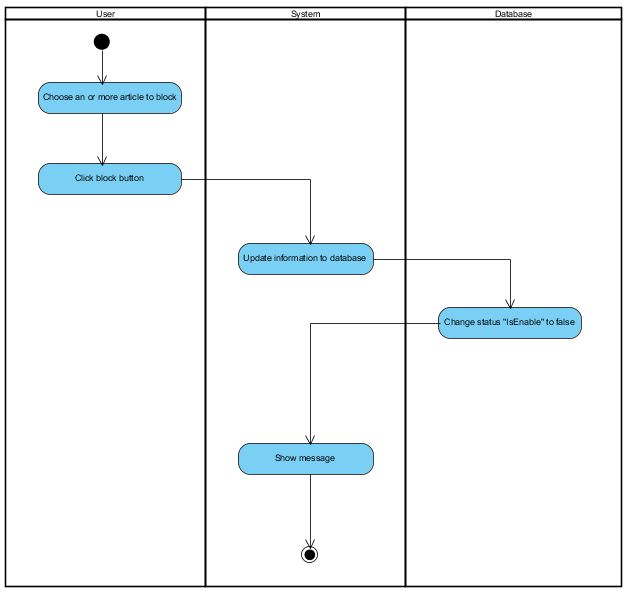
Admin, Employee change password



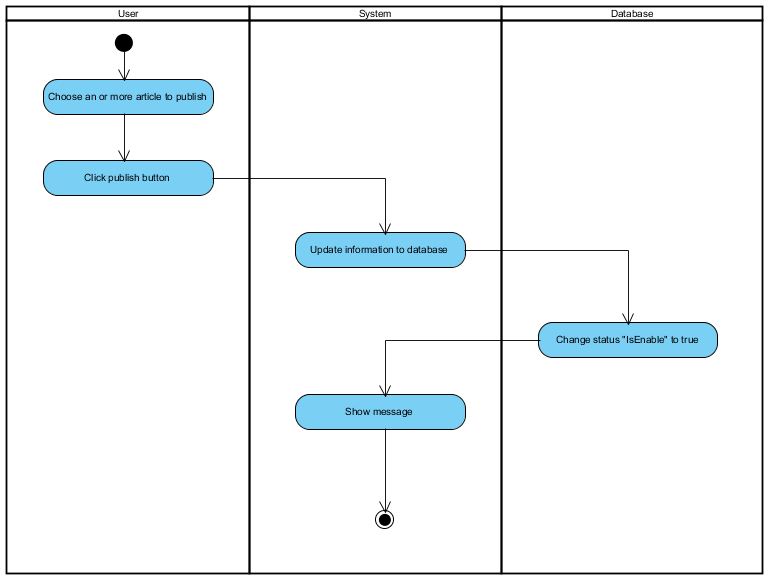
Admin, Employee logout



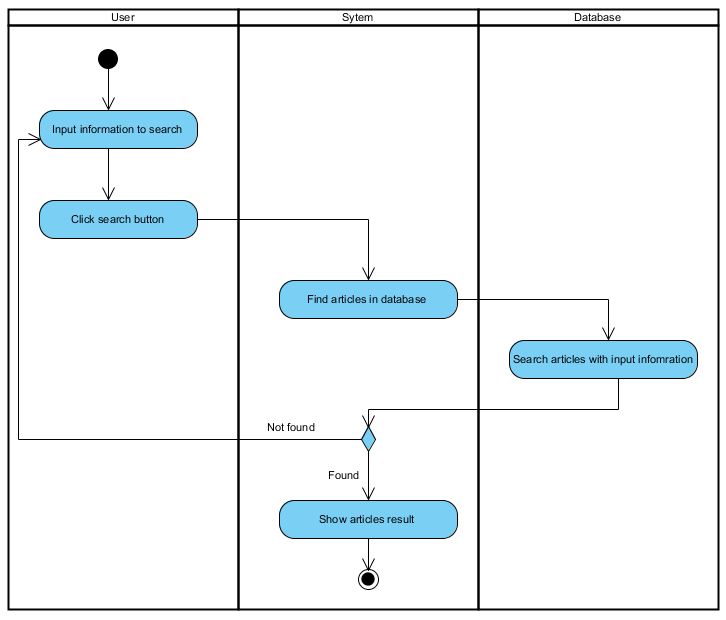
Admin add new Article



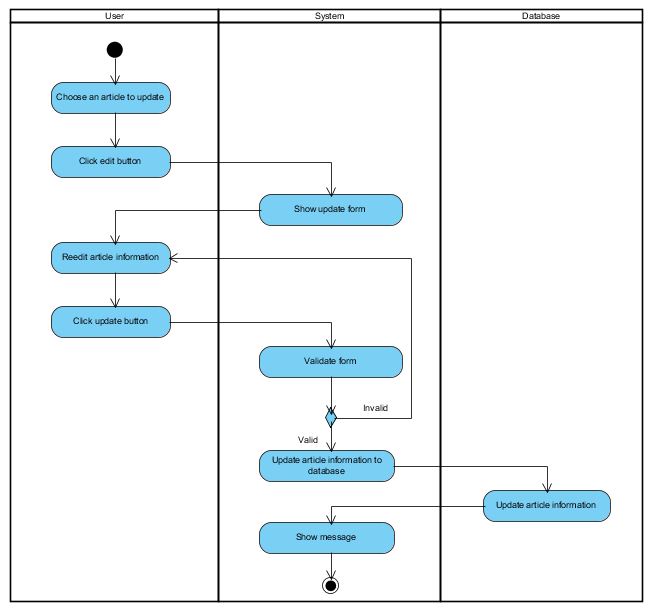
Admin block Article



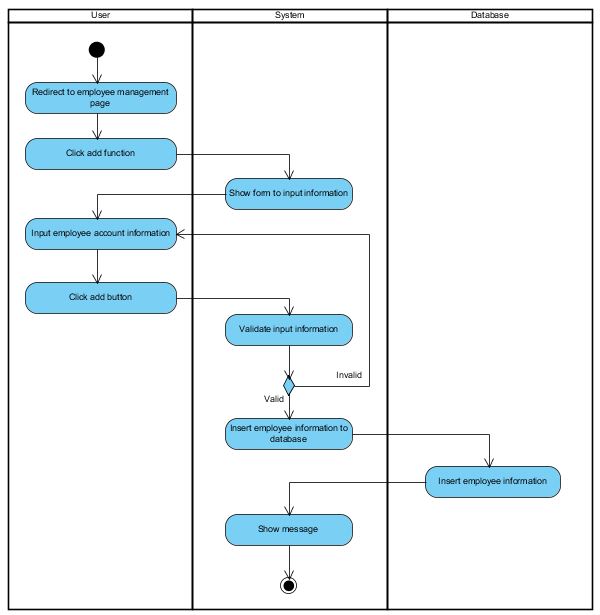
Admin public Article



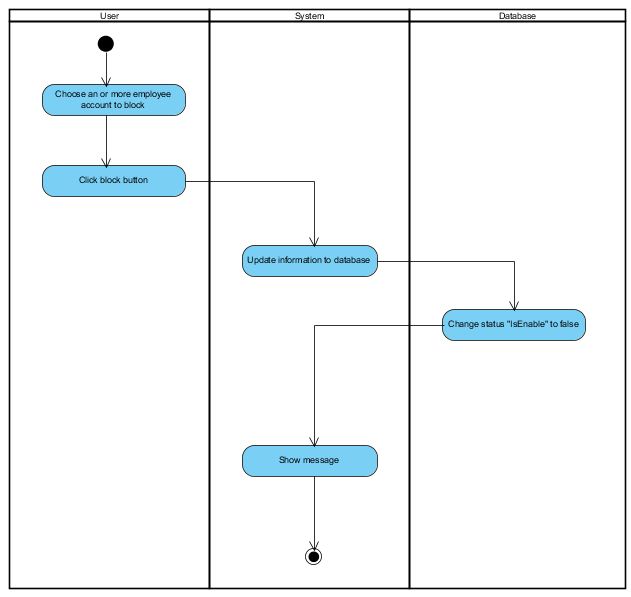
Admin search Article



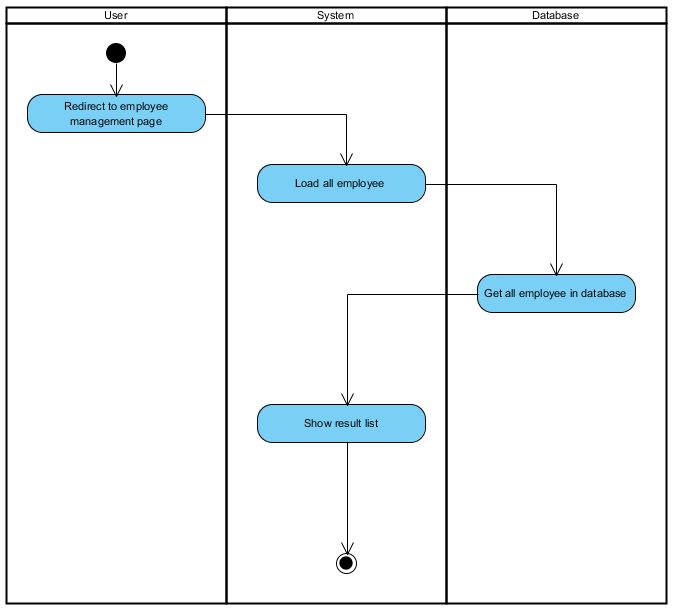
Admin update Article



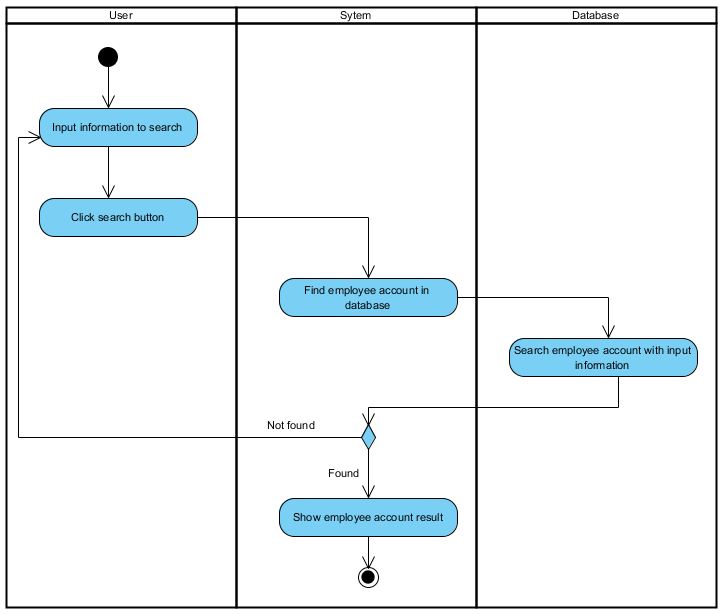
Admin add new Employee account



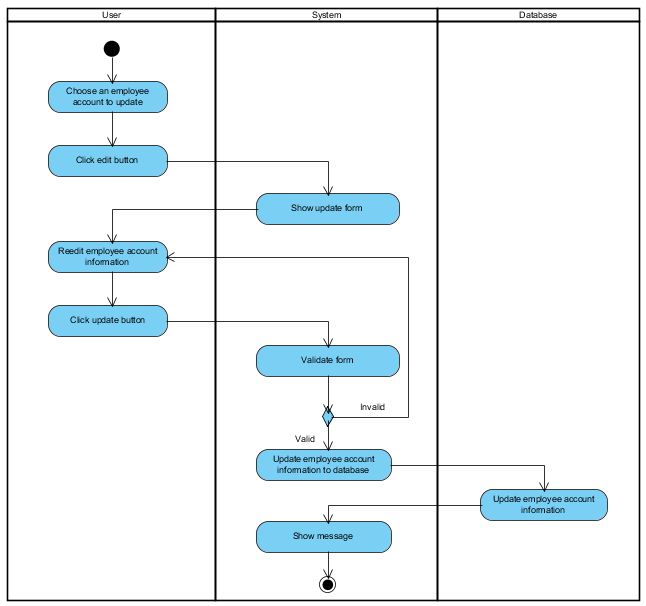
Admin block Employee account



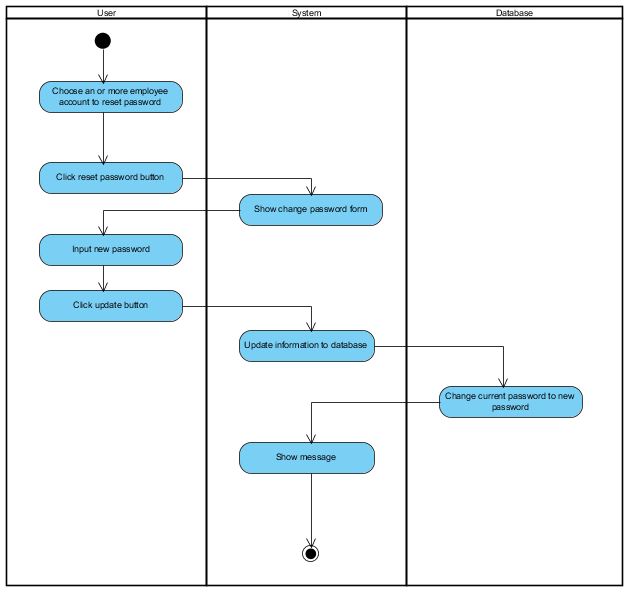
Admin list Employee account



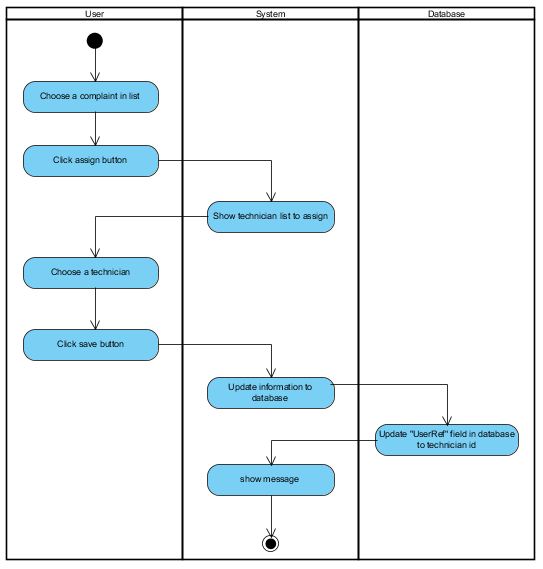
Admin search Employee account



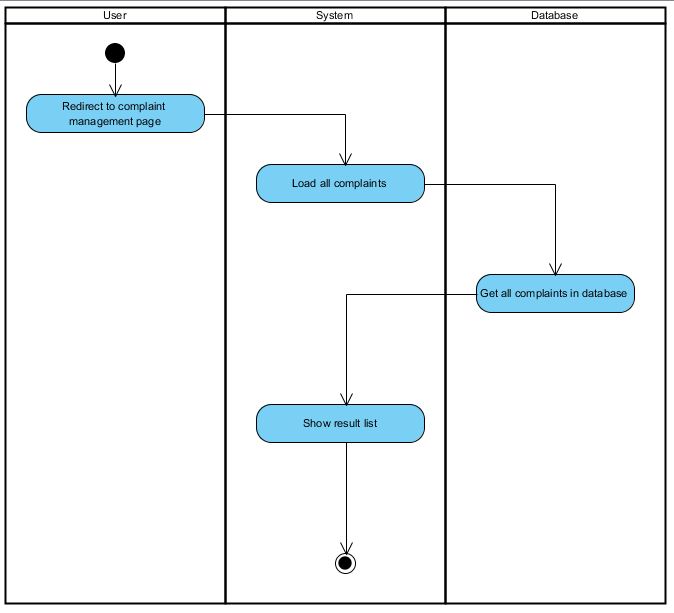
Admin update Employee account



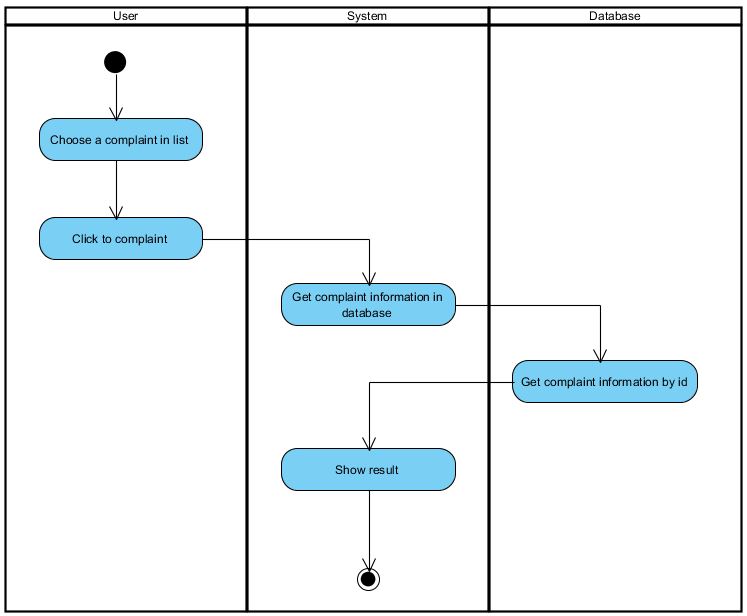
Admin reset Employee account password



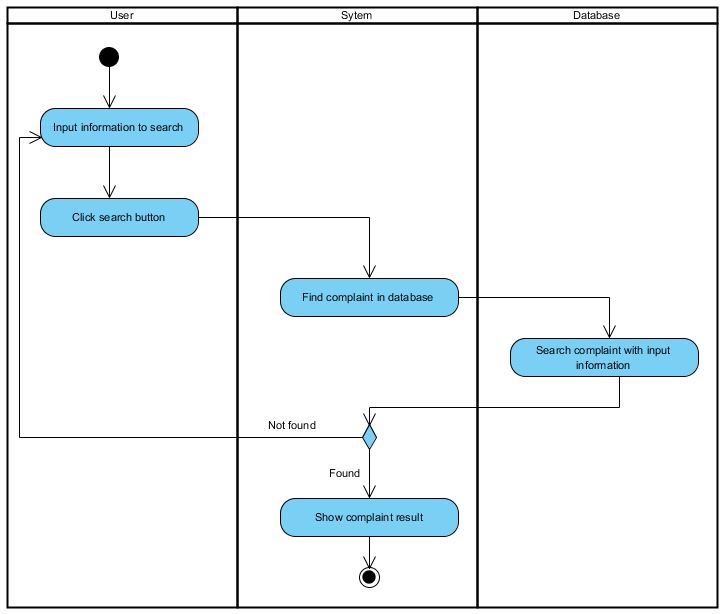
Admin assign the task to a technician



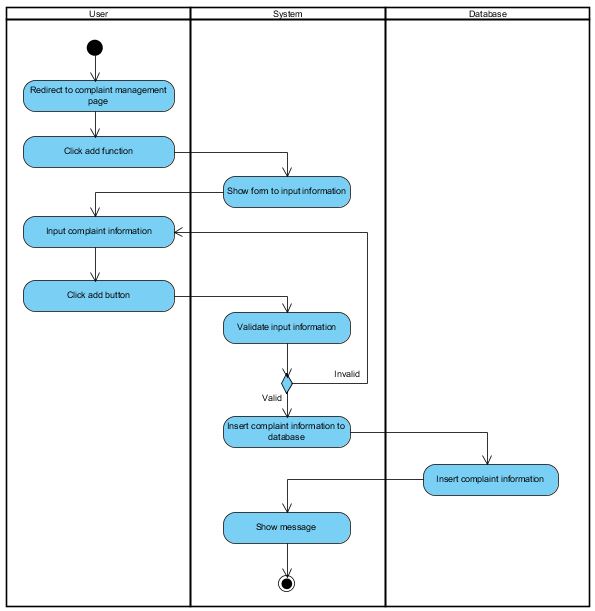
Admin list all complaints



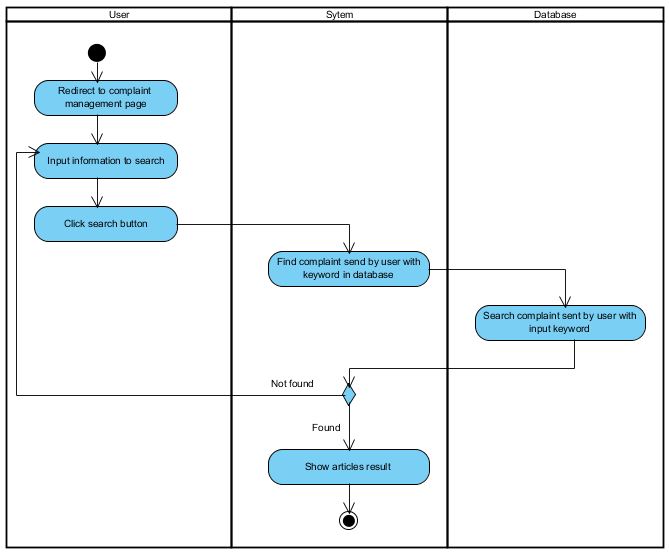
Admin read a complaint



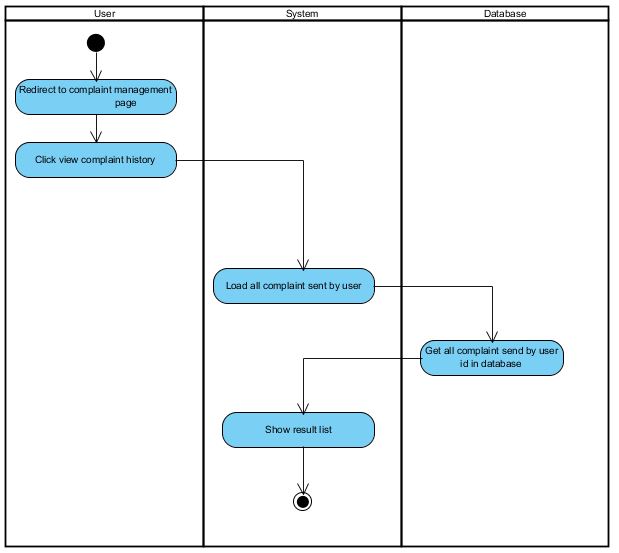
Admin search complaints



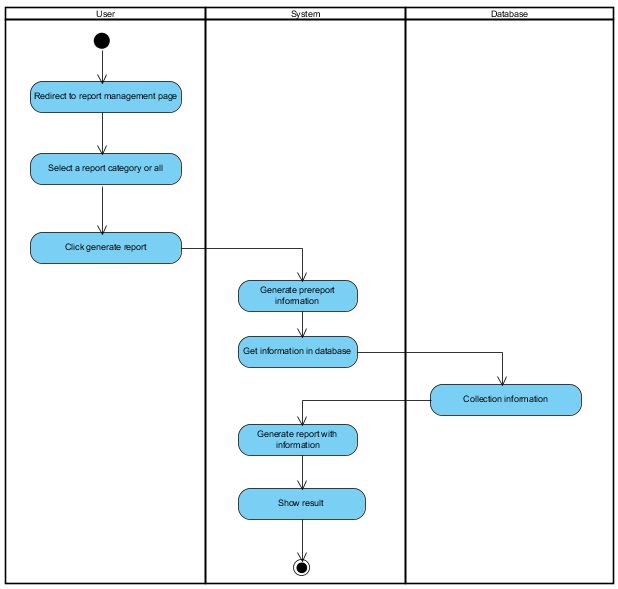
Employee create new complaint



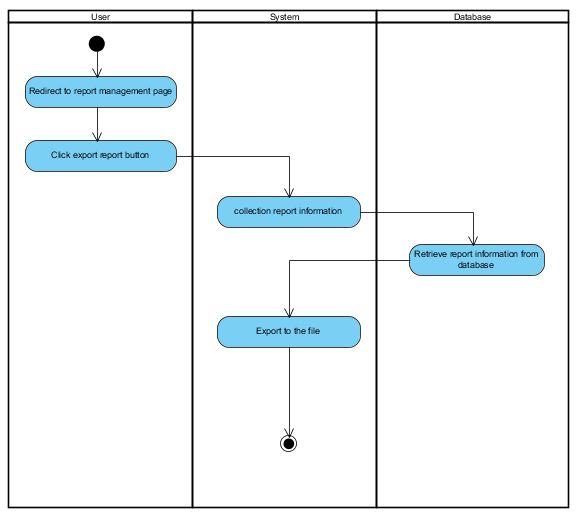
Employee search complaints log



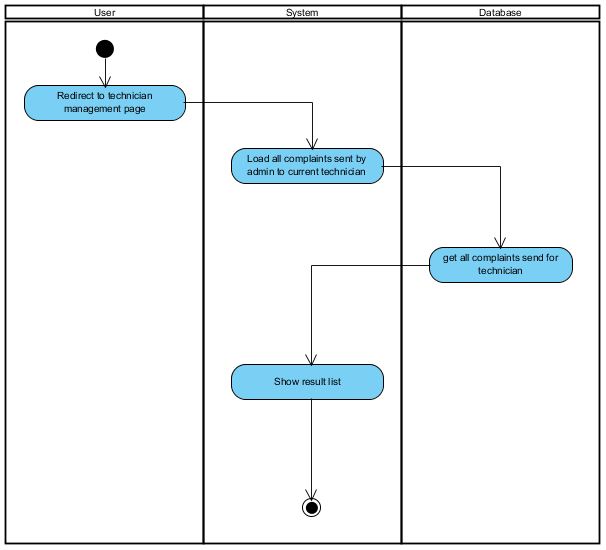
Employee view complaints log



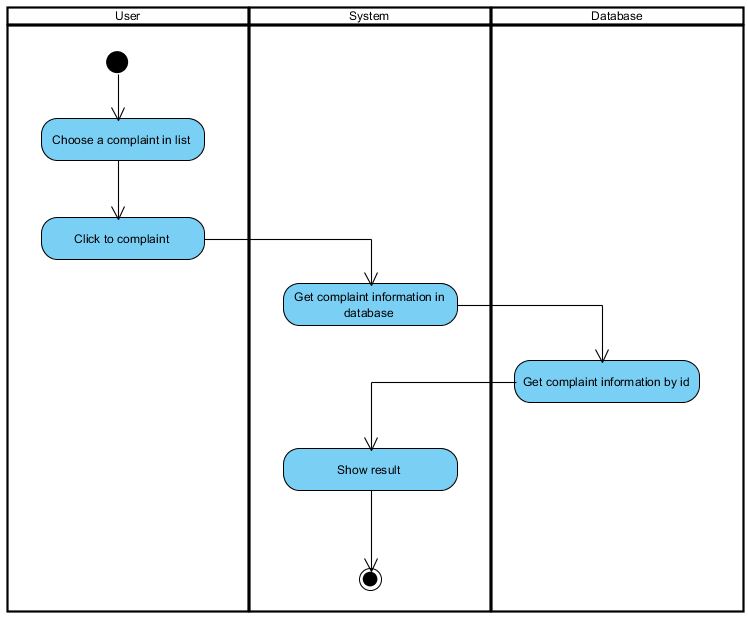
Admin generate the report



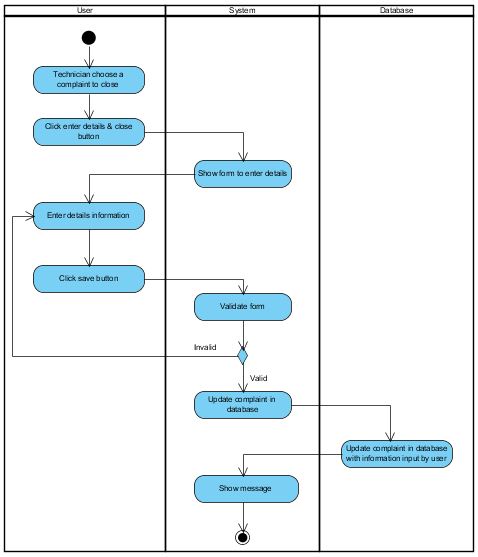
Admin export the report



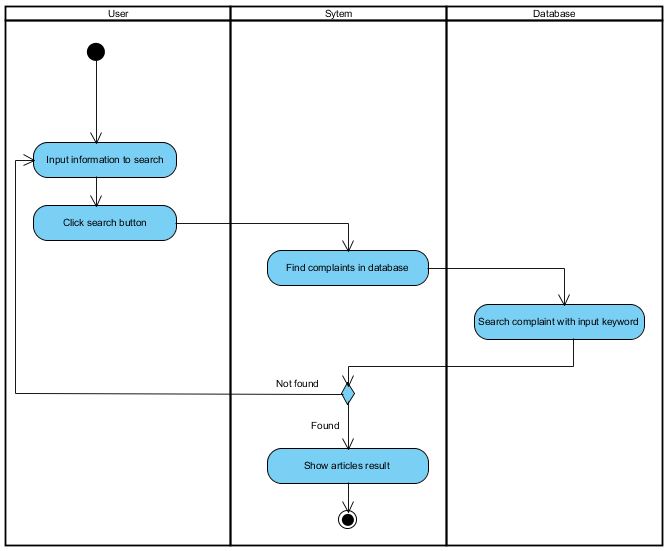
Technician list all complaint send by admin



Technician read a complaint

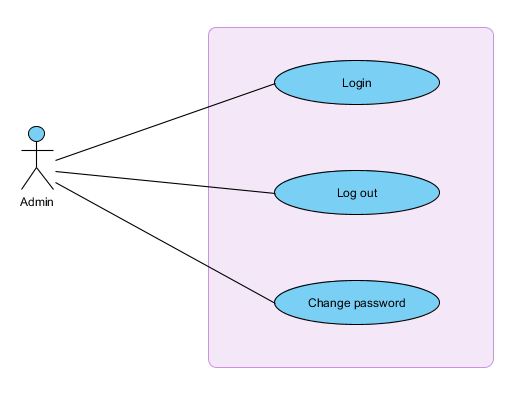


Technician close a complaint

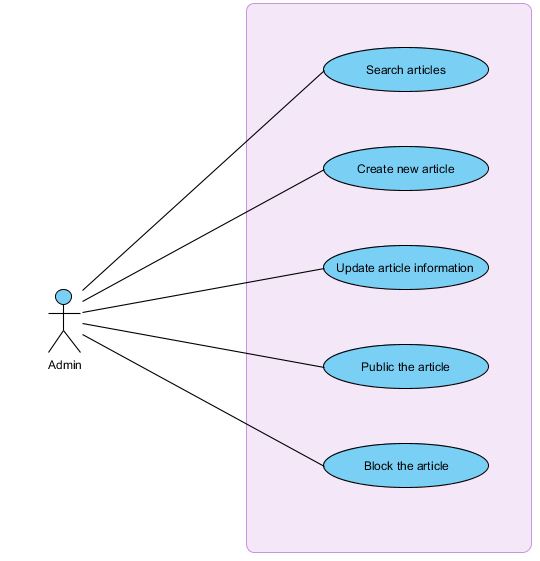


Technician search complaints information

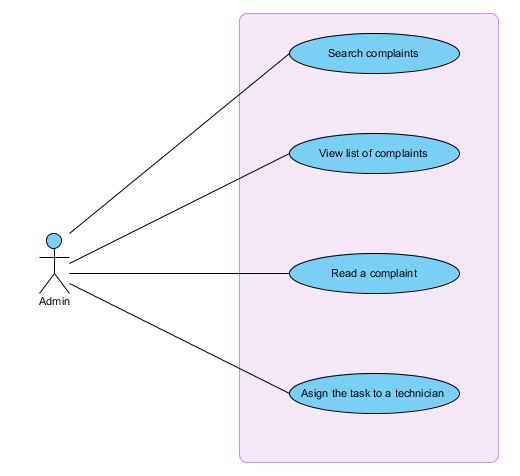
## Use Case Diagram



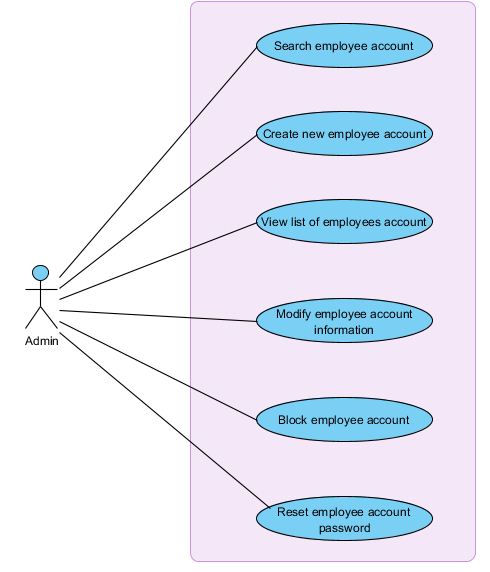
Admin with account



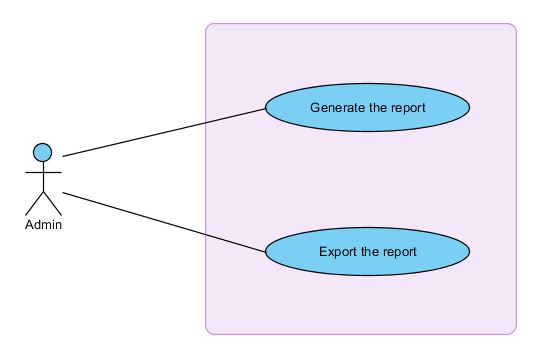
Admin with article



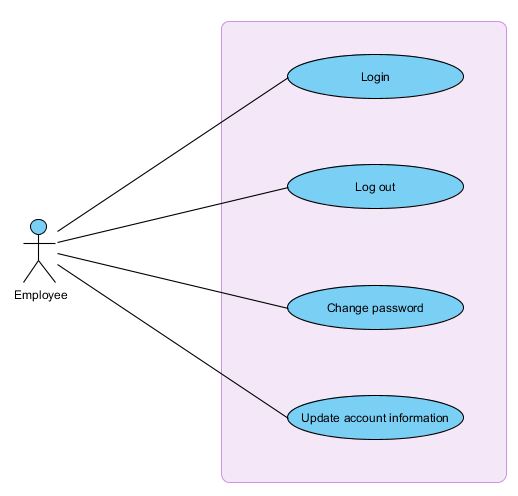
Admin with complaint



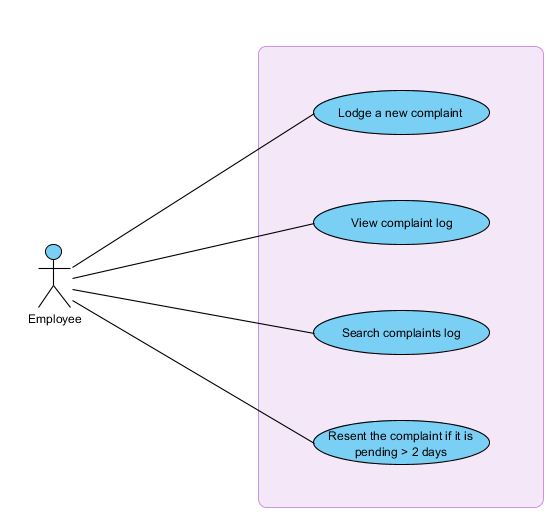
Admin with employee account



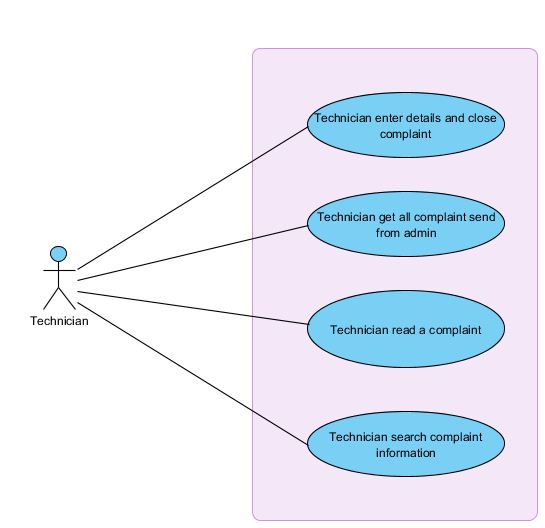
Admin with report



Employee with account



Employee with complaint



Technician with complaint

## Use Case Specification

|  |  |  |
| --- | --- | --- |
| USE CASE # | UC001 | |
| USE CASE Name | Login | |
| ACTOR | Admin, Employee | |
| Purpose (1 phrase) | Allow users login into the system | |
| Overview and scope | User redirect to login page | |
| Level | Primary | |
| Preconditions | User Provides username and password to login Form | |
| Post conditions in words | System redirect to return URL or home page. | |
| **Trigger** | User click Login button | |
| **Included Use Cases** | None | |
| **Extended Use Cases** | None | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** |
| 1. User redirect to Login page. |  |
| 2. User provides username and password to Login Form. |  |
| 3. User click Login button |  |
|  | 4. System validate user login |
|  | 5. System redirect to return URL or home page if return URL is empty. |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** |
|  |  |
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | **Actions** |
| 3a. Form is invalid | User may reenter information missing or wrong. |
|  | 4a. System fail to validate user account | User must reenter username and password |
| **Developer** |  | |
| **Creation date and last modified date** |  | |

|  |  |  |
| --- | --- | --- |
| USE CASE # | UC002 | |
| USE CASE Name | Create new employee | |
| ACTOR | Admin | |
| Purpose (1 phrase) | Allow admin add new account for employee | |
| Overview and scope | User redirect to login page | |
| Level | Primary | |
| Preconditions |  | |
| Post conditions in words | System redirect to return URL or home page. | |
| **Trigger** | User click Submit button | |
| **Included Use Cases** | None | |
| **Extended Use Cases** | None | |
| ***MAIN S\UCCESSFUL SCENARIO*** | **Actor Action** | **System Action** |
| 1. Admin redirect to Add new Employee page |  |
| 2. Admin input information of employee |  |
| 4. Admin click Submit button |  |
|  | 5. System validate text required |
|  | 6. System add new record to database |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** |
|  |  |
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | **Actions** |
| 4a. Form is invalid | Admin may reenter information missing or wrong. |
| **Developer** |  | |
| **Creation date and last modified date** |  | |

|  |  |  |
| --- | --- | --- |
| USE CASE # | UC003 | |
| USE CASE Name | View list employee account | |
| ACTOR | Admin | |
| Purpose (1 phrase) | Allow admin view list employee in system | |
| Overview and scope |  | |
| Level | Primary | |
| Preconditions |  | |
| Post conditions in words | System redirect to return URL or home page. | |
| **Trigger** | User click Submit button | |
| **Included Use Cases** | None | |
| **Extended Use Cases** | None | |
| ***MAIN \UCCESSFUL SCENARIO*** | **Actor Action** | **System Action** |
| 1. Admin redirect to List Employee page |  |
|  | 2. System validate user login |
|  | 3. System load all account of employee to the list. |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** |
| 2a. Admin is in role of admin |  |
|  |  |
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | **Actions** |
| 2a. System fails to authorize Admin account | System redirect to Login page and admin need to log into the system |
| **Developer** |  | |
| **Creation date and last modified date** |  | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC004 | | |
| USE CASE Name | Search employee | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Search employee account information | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin enters keyword to search | | |
| Post conditions in words | System display details information about employee account contain keyword. | | |
| **Trigger** | Admin clicks **Search** button | | |
| **Included Use Cases** | None | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Employee Management page. |  | |
|  | 2. System validate user login. | |
| 3.Admin click Search button |  | |
|  | 4. System search employees account in database and return records corresponding. | |
|  | 5. System display result | |
|  |  |  | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. Admin is in role of admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize Admin account | | System redirect to Login page and admin need to log into the system |
|
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC005 | | |
| USE CASE Name | Update employee account information | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Allow update employee account information existing in the system | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin need login before update an employee account 2. Admin choose an employee account want to edit the information in the list of employees account in the system. | | |
| Post conditions in words | System display employee account is updated successfully to the database. | | |
| **Trigger** | Admin click Edit button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Employee Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of employees account, Admin chooses an employee account want to edit. |  | |
| 4. Admin click Edit button |  | |
|  | 5. System show Edit form for employee account | |
| 6. Admin modify employee account information like full name etc. |  | |
| 7. Admin click Update button |  | |
|  | 8. System will update employee account information to the database | |
|  |  | 9. System displays a message to Admin. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. Admin is in role of admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize Admin account | | System redirect to Login page and admin need to log into the system |
| 7a. Information in Form is invalid | | User need to enter information missing. |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC006 | | |
| USE CASE Name | Block employee account | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | block an employee account existing in the database | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin need login before disable a bouquet 2. Admin choose an employee account want to block in the list of employees account in the system. | | |
| Post conditions in words | System display employee account is blocked successfully. | | |
| **Trigger** | Admin click Disable button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Employee Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of employees account, admin chooses an employee account want to disable. |  | |
| 4. Admin click Disable button |  | |
|  | 5. System show confirm allow disable or not | |
| 6. Admin click Yes button to continue, No to cancel |  | |
|  | 7. If admin chooses “Yes”, system will update employee account information to the database with “IsEnable” attribute equal FALSE. | |
|  |  | 8. System displays a message to admin. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in role of admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account | | System redirect to Login page and admin need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC007 | | |
| USE CASE Name | Reset employee account password | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | reset an employee account password existing in the database | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin need login before disable a bouquet 2. Admin choose an employee account want to reset password in the list of employees account in the system. | | |
| Post conditions in words | System display reset employee account password successfully. | | |
| **Trigger** | Admin click Reset Password button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Employee Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of employees account, admin chooses an employee account want to reset password. |  | |
| 4. Admin click Reset Password button |  | |
|  | 5. System show form for input new password | |
| 6. Admin input new password and click Save button to continue |  | |
|  | 7. system update new employee account password to the. | |
|  |  | 8. System displays a message to admin. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in role of admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account | | System redirect to Login page and admin need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC008 | | |
| USE CASE Name | Search complaints | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Search complaints information | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin enters keyword to search | | |
| Post conditions in words | System display details information about complaints contain keyword. | | |
| **Trigger** | Admin clicks **Search** button | | |
| **Included Use Cases** | None | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Complaints Management page |  | |
|  | 2. System validate user login | |
| 3.Admin click Search button |  | |
|  | 4. System search complaints in database and return records corresponding. | |
|  | 5. System display result | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in Admin role |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account. | | System redirect to Login page and admin need to log into the system |
|
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC009 | | |
| USE CASE Name | View list of complaints | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | View all complaints in system | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin need login into the system 2. Admin click View Complaint List | | |
| Post conditions in words | System displays the list of complaints for admin. | | |
| **Trigger** | User clicks **View Complaint List** button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Complaints Management page |  | |
|  | 2. System validate user login | |
| 3. User click View Complaints List button |  | |
|  | 5. System get all complaints in to admin | |
|  | 6. System display result | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in Admin role |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account. | | System redirect to Login page and admin need to log into the system |
|
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC010 | | |
| USE CASE Name | Read a complaint | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Allow admin read complaint details information | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User need login before read a complaint 2. Users choose a complaint want to read details in the system. | | |
| Post conditions in words | System display details of the complaint. | | |
| **Trigger** | User click View button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Complaints Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of complaints, admin chooses a complaint want to view details. |  | |
| 4. User click View button |  | |
|  | 5. System show complaint details from database | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in Admin role |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account. | | System redirect to Login page and admin need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |
| --- | --- | --- |
| USE CASE # | UC011 | |
| USE CASE Name | Assign complain to technical | |
| ACTOR | Admin | |
| Purpose (1 phrase) | Allow admin assign the task to a technician based on complaint id | |
| Overview and scope |  | |
| Level | Primary | |
| Preconditions |  | |
| Post conditions in words |  | |
| **Trigger** | User click Assign To button | |
| **Included Use Cases** | Login | |
| **Extended Use Cases** | None | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** |
| 1. Admin redirect to Article Management page |  |
|  | 2. System authorization the user |
| 3. Admin choose complain in the list |  |
| 4. Admin click Assign To button |  |
| 5. Admin choose technical from combo box and department name, click Save button |  |
|  | 6. System update complaint information with UserRef property equal technician based ID |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** |
| 2a. admin is in Admin role |  |
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | **Actions** |
| 2a. System fails to authorize admin account. | System redirect to Login page and admin need to log into the system |
| **Developer** |  | |
| **Creation date and last modified date** |  | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC012 | | |
| USE CASE Name | Search articles | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Search articles information | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin enters keyword to search | | |
| Post conditions in words | System display details information about articles contain keyword. | | |
| **Trigger** | Admin clicks **Search** button | | |
| **Included Use Cases** | None | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Article Management page |  | |
|  | 2. System authorization the user | |
| 3.Admin click Search button |  | |
|  | 4. System search articles in database and return records corresponding. | |
|  | 5. System display result | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in Admin role |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account. | | System redirect to Login page and admin need to log into the system |
|
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |
| --- | --- | --- |
| USE CASE # | UC013 | |
| USE CASE Name | Create new article | |
| ACTOR | Admin | |
| Purpose (1 phrase) | Allow admin add new article | |
| Overview and scope | User redirect to login page | |
| Level | Primary | |
| Preconditions |  | |
| Post conditions in words | System redirect to articles list | |
| **Trigger** | User click Add New Article button | |
| **Included Use Cases** | None | |
| **Extended Use Cases** | None | |
| ***MAIN S\UCCESSFUL SCENARIO*** | **Actor Action** | **System Action** |
| 1. Admin redirect to Article Management page |  |
|  | 2. System authorization the user |
| 3. Admin click Add New Article button |  |
| 4. Admin input information of article |  |
| 5. Admin click Save button |  |
|  | 6. System validate text required |
|  | 7. System add new record to database |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** |
| 2a. admin is in Admin role |  |
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | **Actions** |
| 2a. System fails to authorize admin account. | System redirect to Login page and admin need to log into the system |
| 5a. Form is invalid | Admin may reenter information missing or wrong. |
| **Developer** |  | |
| **Creation date and last modified date** |  | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC014 | | |
| USE CASE Name | Update article information | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Allow update article information existing in the system | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin need login before update an employee account 2. Admin choose an article want to edit the information in the list of articles in the system. | | |
| Post conditions in words | System display article is updated successfully to the database. | | |
| **Trigger** | Admin click Edit button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Article Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of articles, Admin chooses an article want to edit. |  | |
| 4. Admin click Edit button |  | |
|  | 5. System show Edit form for article | |
| 6. Admin modify article information like title, etc. |  | |
| 7. Admin click Save button |  | |
|  | 8. System will update article information to the database | |
|  |  | 9. System displays a message to admin. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. Admin is in role of admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize Admin account | | System redirect to Login page and admin need to log into the system |
| 7a. Information in Form is invalid | | User need to enter information missing. |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC015 | | |
| USE CASE Name | Public the article | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Public an article existing in the database | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin need login before public an article 2. Admin choose an article want to public in the list of articles in the system. | | |
| Post conditions in words | System display article is public successfully. | | |
| **Trigger** | Admin click Edit button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Article Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of articles, admin chooses an article want to public. |  | |
| 4. Admin click Edit button |  | |
|  | 5. System show edit form for admin | |
| 6. Admin choose IsEnable status to true if admin want to public and click Save button |  | |
|  | 7. System update article information to the database with “IsEnable” attribute equal TRUE. | |
|  |  | 8. System displays a message to admin. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in role of admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account | | System redirect to Login page and admin need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC016 | | |
| USE CASE Name | Block the article | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Block an article existing in the database | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. Admin need login before public an article 2. Admin choose an article want to block in the list of articles in the system. | | |
| Post conditions in words | System display article is blocked successfully. | | |
| **Trigger** | Admin click Edit button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. Admin redirect to Article Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of articles, admin chooses an article want to public. |  | |
| 4. Admin click Edit button |  | |
|  | 5. System show edit form for admin | |
| 6. Admin choose IsEnable status to false if admin want to block and click Save button |  | |
|  | 7. System updates article information to the database with “IsEnable” attribute equal FALSE. | |
|  |  | 8. System displays a message to admin. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in role of admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account | | System redirect to Login page and admin need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC017 | | |
| USE CASE Name | Generate the report | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Generate the report information | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User need login before generate the report | | |
| Post conditions in words | System display report generated from system to admin | | |
| **Trigger** | User click Generate Report button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Report Management page. |  | |
|  | 2. System validate user login. | |
| 3. User click Generate Report button |  | |
|  | 4. System will get information in database and generate the report to admin | |
|  |  | 6. System displays the report | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in role of Admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account | | System redirect to Login page and admin need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC018 | | |
| USE CASE Name | Export the report | | |
| ACTOR | Admin | | |
| Purpose (1 phrase) | Export the report information | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User need login before export the report | | |
| Post conditions in words | System export report generated from system to file | | |
| **Trigger** | User click Export Report button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Report Management page. |  | |
|  | 2. System validate user login. | |
| 3. User click Export Report button |  | |
|  | 4. System will get information in database and generate the report and export to the file | |
|  |  | 6. System export the report to a file like pdf, xml, etc. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. admin is in role of Admin |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account | | System redirect to Login page and admin need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC019 | | |
| USE CASE Name | Update account information | | |
| ACTOR | Admin, Employee | | |
| Purpose (1 phrase) | Allow update account information | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User need login before update account information 2. User choose edit the account information in menu | | |
| Post conditions in words | System display account information is updated successfully to the database. | | |
| **Trigger** | User click Edit button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to My Account page. |  | |
|  | 2. System validate user login. | |
| 3. User chooses edit function button in the menu. |  | |
|  | 4. System show Edit form for account information. | |
| 5. User modify account information like full name, address, etc. |  | |
| 6. User click Save button |  | |
|  | 7. System will update account information to the database | |
|  |  | 8. System displays a message to user. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. User is in role of admin, employee |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize user account | | System redirect to Login page and user need to log into the system |
| 6a. Information in Form is invalid | | User need to enter information missing. |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC020 | | |
| USE CASE Name | Technician enter details and close complaint | | |
| ACTOR | Employee | | |
| Purpose (1 phrase) | Allow enter details for complaint and close complaint | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User need login before enter information for complaint 2. User choose Close Complaint in action menu | | |
| Post conditions in words | System complaint is closed successfully and information saved in database | | |
| **Trigger** | User click Close Complaint button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Complaint Management page. |  | |
|  | 2. System validate user login. | |
| 3. User chooses Close complaint function button in the action menu. |  | |
|  | 4. System show form for enter details | |
| 5. User enters details of complaint. |  | |
| 6. User click Save button |  | |
|  | 7. System will save user entered information to the database | |
|  |  | 8. System displays a message to user. | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. User is in role of employee |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize user account | | System redirect to Login page and user need to log into the system |
| 6a. Information in Form is invalid | | User need to enter information missing. |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC021 | | |
| USE CASE Name | Technician get all the complaints send from admin | | |
| ACTOR | Employee | | |
| Purpose (1 phrase) | View all complaints send from admin | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User need login into the system 2. User click View Complaint List | | |
| Post conditions in words | System displays the list of complaints for admin. | | |
| **Trigger** | User clicks **View Complaint List** button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Technician Management page |  | |
|  | 2. System validate user login | |
| 3. User click View Complaints List button |  | |
|  | 5. System get all the complaints send from admin | |
|  | 6. System display result | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. user is in Employee role |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize admin account. | | System redirect to Login page and admin need to log into the system |
|
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC022 | | |
| USE CASE Name | Technician read a complaint | | |
| ACTOR | Employee | | |
| Purpose (1 phrase) | Allow technician read complaint details information send from admin | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User need login before read a complaint 2. Users choose a complaint want to read details in the system. | | |
| Post conditions in words | System display details of the complaint. | | |
| **Trigger** | User click View button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Technician Management page. |  | |
|  | 2. System validate user login. | |
| 3. In the list of complaints, user chooses a complaint want to view details. |  | |
| 4. User click View button |  | |
|  | 5. System show complaint details from database | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. user is in Employee role |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize user account. | | System redirect to Login page and user need to log into the system |
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| USE CASE # | UC023 | | |
| USE CASE Name | Technician search complaints information | | |
| ACTOR | Employee | | |
| Purpose (1 phrase) | Search complaints information send from admin | | |
| Overview and scope |  | | |
| Level | Primary | | |
| Preconditions | 1. User enters keyword to search | | |
| Post conditions in words | System display details information about complaints send from admin to user contain keyword. | | |
| **Trigger** | User clicks **Search** button | | |
| **Included Use Cases** | Login | | |
| **Extended Use Cases** | None | | |
| ***MAIN SUCCESSFUL SCENARIO*** | **Actor Action** | **System Action** | |
| 1. User redirect to Technician Management page. |  | |
|  | 2. System validate user login. | |
| 3.User click Search button |  | |
|  | 4. System search all complaint sends from admin to user in database and return records corresponding. | |
|  | 5. System display result | |
| ***OTHER SUCCESSFUL SCENARIOS*** | **Step** | **Branching Action** | |
| 2a. User is in Employee role |  | |
|
|
|
|
|
| ***UNSUCCESSFUL SCENARIOS*** | **Conditions** | | **Actions** |
| 2a. System fails to authorize user account. | | System redirect to Login page and user need to log into the system |
|
| **Developer** |  | | |
| **Creation date and last modified date** |  | | |

# Design

## System Architecture

Controller

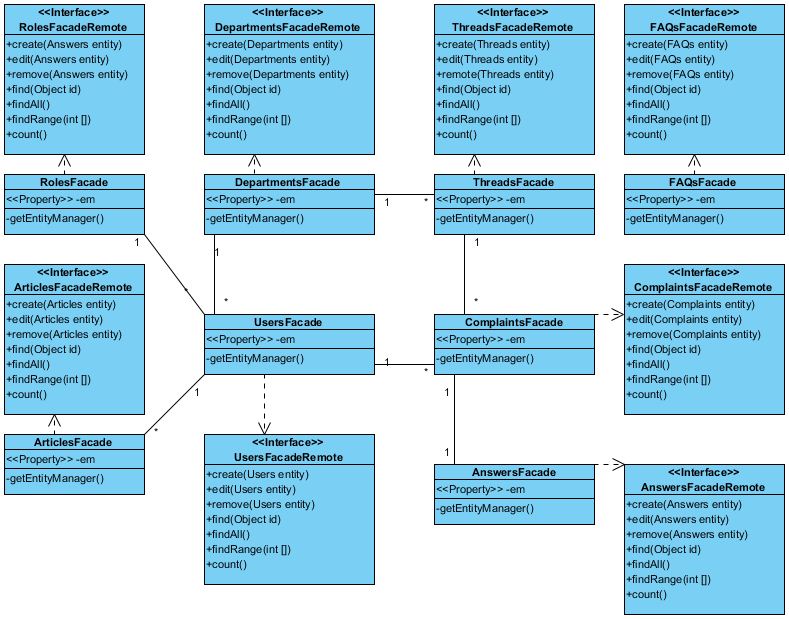
View

Model

Model – View – Controller architecture

## D:\Class\SEM IV\E-Project\c0812i-semiv\class_diagram\entity.JPGClass Diagram

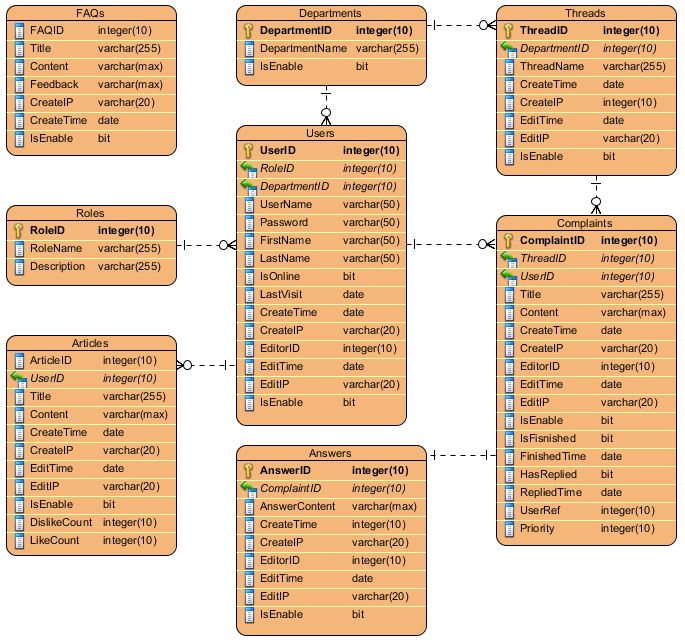
Entities Class Diagram



Facade Class Diagram

## Sequence Diagram

## Entity Relationship Diagram

**

## Database Design

|  |  |  |
| --- | --- | --- |
| **Table: Roles** | | |
| **Column** | **Data Type** | **Description** |
| RoleID | Int | PK |
| RoleName | Nvarchar(50) | Not null |
| Description | Nvarchar(max) | Allow null |

|  |  |  |
| --- | --- | --- |
| **Table: Departments** | | |
| **Column** | **Data Type** | **Description** |
| DepartmentID | Int | PK |
| DepartmentName | Nvarchar(200) | Unique, Not null |
| IsEnable | Bit | Default 1, not null |

|  |  |  |
| --- | --- | --- |
| **Table: Users** | | |
| **Column** | **Data Type** | **Description** |
| UserID | Int | PK |
| RoleID | Int | FK, Not null |
| DepartmentID | Int | FK, Not null |
| Username | Nvarchar(50) | Unique, not null |
| Password | Nvarchar(50) | Not null |
| FirstName | Nvarchar(50) | Not null |
| LastName | Nvarchar(50) | Not null |
| IsOnline | Bit | Default 0 |
| LastVisit | Datetime | Default GetDate(), Not null |
| CreateTime | Datetime | Default GetDate(), Not null |
| CreateIP | Nvarchar(20) | Not null |
| EditorID | Int | FK |
| EditTime | Date | Default GetDate() |
| EditIP | Nvarchar(20) |  |
| IsEnable | Bit | Default 1, not null |

|  |  |  |
| --- | --- | --- |
| **Table: Threads** | | |
| **Column** | **Data Type** | **Description** |
| ThreadID | Int | PK |
| DepartmentID | Int | FK, not null |
| ThreadName | Nvarchar(256) | Not null |
| CreateTime | Datetime | Not null |
| CreateIP | Nvarchar(20) | Not null |
| EditTime | Datetime | Default GetDate() |
| EditIP | Nvarchar(20) |  |
| IsEnable | Bit | Default 1, not null |

|  |  |  |
| --- | --- | --- |
| **Table: Complaints** | | |
| **Column** | **Data Type** | **Description** |
| ComplaintID | Int | PK |
| ThreadID | Int | FK, not null |
| UserID | Int | FK, not null |
| Title | Nvarchar(256) | Not null |
| ComplaintContent | Nvarchar(max) | Not null |
| CreateTime | Datetime | Not null |
| CreateIP | Nvarchar(20) | Not null |
| EditorID | Int | FK |
| EditIP | Nvarchar(20) |  |
| IsEnable | Bit | Default 1, not null |
| IsFinished | Bit | Default 0, not null |
| FinishedTime | Datetime |  |
| HasReplied | Bit | Default 0, not null |
| RepliedTime | Datetime |  |
| UserRef | Int | FK |
| Priority | Int | Not null |

|  |  |  |
| --- | --- | --- |
| **Table: Answers** | | |
| **Column** | **Data Type** | **Description** |
| AnswerID | Int | PK |
| ComplaintID | Int | FK, not null |
| AnswerContent | Nvarchar(max) | Not null |
| CreateTime | Datetime | Not null |
| CreateIP | Nvarchar(20) | Not null |
| EditorID | Int |  |
| EditTime | Datetime | Default GetDate() |
| EditIP | Nvarchar(20) |  |
| IsEnable | Bit | Default 0, not null |

|  |  |  |
| --- | --- | --- |
| **Table: Articles** | | |
| **Column** | **Data Type** | **Description** |
| ArticleID | Int | PK |
| UserID | Int | FK, not null |
| Title | Nvarchar(500) | Not null |
| Content | Nvarchar(max) | Not null |
| CreateTime | Datetime | Not null |
| CreateIP | Nvarchar(20) | Not null |
| EditTime | Datetime | Default GetDate() |
| EditIP | Nvarchar(20) |  |
| IsEnable | Bit | Default 1, not null |
| DislikeCount | Int | Not null |
| LikeCount | Int | Not null |

|  |  |  |
| --- | --- | --- |
| **Table: FAQs** | | |
| **Column** | **Data Type** | **Description** |
| FAQID | Int | PK |
| Title | Nvarchar(500) | Not null |
| Content | Nvarchar(max) | Not null |
| Feedback | Nvarchar(max) |  |
| CreateIP | Nvarchar(20) | Not null |
| CreateTime | Datetime | Not null |
| IsEnable | Bit | Default 1, not null |

# Management and Project Planning

## Management Approach

The group members using Google Code to manage the project code, and TortoiseSVN tool.

## Project Plan

## Task Sheet

[Task Sheet](TaskSheet.xls)

# Checklists

## Check List of Validation

## Submission Checklist

# Screenshots

# Coding Convention