|  |
| --- |
| **FPT- aptech computer education** |
| eProject Report |
| Nexus SERVICE MARKETING SYSTEM |
|  |
| |  |  | | --- | --- | | **T1708A\_GroupII** | | | **Group Member** | Hoàng Sang - D00541  Đào Tuấn Tú - D00552  Vũ Trí Thành -D00554  Nguyễn Đại Dương – D00618  Phạm Thị Thanh Tươi - D00556 | | **Instructor** | Đặng Kim Thi | |
|  |

|  |
| --- |
| - Hanoi, 3/2019 - |

INDEX

[1. Acknowledgements 4](#_Toc296936987)

[2. Introduction 4](#_Toc296936988)

[3. Problem Definition 4](#_Toc296936989)

[3.1. Problem Abstraction 4](#_Toc296936990)

[3.2. The Current System 4](#_Toc296936991)

[3.3. The Proposed System 4](#_Toc296936992)

Non-Financial :

Financial :

[3.4. Boundaries of the System 4](#_Toc296936993)

[3.5. Development Environment 4](#_Toc296936994)

[4. Requirements and Business Flows 5](#_Toc296936995)

[4.1. Customer Requirements 5](#_Toc296936996)

[4.2. Activity Diagram 5](#_Toc296936997)

[4.3. Use Case Diagram 5](#_Toc296936998)

[5. Design 7](#_Toc296937001)

[5.1. System Architecture 7](#_Toc296937002)

[5.2. Class Diagram 7](#_Toc296937003)

[5.3. Class Diagram Explanation 7](#_Toc296937004)

[5.7. Entity Relationship Diagram 7](#_Toc296937008)

[5.8. Database Design 7](#_Toc296937009)

[6. System Prototype 9](#_Toc296937012)

[7. Management and Project Planning 10](#_Toc296937013)

[7.1. Management Approach 10](#_Toc296937014)

[7.2. Project Plan 10](#_Toc296937015)

[7.3. Task Sheet 10](#_Toc296937016)

[8. Checklists 11](#_Toc296937018)

[8.1. Check List of Validation 11](#_Toc296937019)

[8.2. Submission Checklist 11](#_Toc296937020)

[9. Screenshots 11](#_Toc296937021)

[10. User Guide 12](#_Toc296937023)

# Acknowledgements

# Introduction

Report on the system to build the process of NEXUS marketing services

# Problem Definition

## Problem Abstraction

Nexus Communication System is an Organization system that provides two services like the tele-communications and Internet Services. It is one of the major vendors at the local place. It has now extended its service all over the territory due to the increase in demand for the internet service.

They want you to research and develop the Nexus system, creating a strategy to solve current problems. In addition, they also want you to propose solutions to use the company that will have a competitive advantage in this fiercely competitive market.

## The Current System

Nexus Communication System, has expanded their service the territory and has even started the retail outlets at each place, so as to increase the comfort level of the customers where the customers can make the payment, also they can place an order or enquire for a new connection. They also recently started a bulk and corporate schemes, so as to retain the customers where they can avail a discount as per the option applied

Presently Nexus :

1. The details of the customers is maintained, including the payments based on the plan and equipment as chosen by the customer.
   * The customer is provided with some unique 10 digit account id so as to track the connection details easily.
   * The details like the address proofs, application form and the receipts are to be filed along with the customer’s records based on the year and area wise.
2. the details of the product manufacturing vendors are maintained along with the orders placed, payment lists, name and contact details are maintained.
3. They also maintain the list of the products that are being provided at the time of providing the connection based on the plan chosen (as they provide different plans), also the details of the payments (security deposit, monthly rentals including the paid amount for the last month and the amount due, it also includes the charges for any replacement done where the equipment has been spoiled by the customer and as well the discounts, etc.).

## The Proposed System

In order to avoid the slipups of the records and to bring the automation in the process, they want an online application to be build up which will update and maintains the records automatically. Also they want the separate login for the Admin to enter the details of the employee, Stock, vendors, retail shops.

They also want to log in separately for employees to place orders, check the status of orders, as well as invoices and maintenance of contact details, and they can also have a clear idea of ​​plans.

The advanced search option should be included for checking the status of the order or the status of the connection with the options being the unique id as per the requirement, name on which the order is placed or the connection is taken, type of connection, date or period of application or received the connection, contact number which is provided at the time of applying for the connection.

Also the customer from his end can check for his/her contact details, place an order, also they can keep track of the status of the order placed, check their bills, etc. They can check the status of the order placed through the unique order id and the status of the connection through their unique account id.

They want a database to hold all the details of the equipments, vendors, retail shops, employers and customers, and the connections provided and that of the applied (order) details, so that they can automate the system.

**►Non-Financial:**

* The database that holds the details of the customers, employees, the various connection or plan details that they provide, the connections assigned, order details, retail show room details, product or equipment details, billing details and the payment details with respect to the billed amount.
* The employee details must include the admin, accounts department people, technical people, and the employees in the retail outlet. Each of the employees are to be provided with the user credentials, so as to login into the application for performing the respective functionalities
* Through the application one should be able to place and order and check for its status. (Note: the members or the employees of the retail outlet can only place the orders)
* Through the employees of the retail outlet login, one should be able to keep track of the orders placed till date, status of the orders (like whether feasible or not, whether the customer is provided with the connection or not), connection details (like the details of the connections provided till date), the till date billing details with respect to the connections assigned, the till date payment details with respect to the bills generated.
* The details of the bills will be maintained by the accounts department. And the details of the payments will be maintained by the employees of the retail outlets and by the accounts department (if the payment is done at the office with in the city).
* The details of the charges that are levied for the connection or a plan, the new plans along with the detailed descriptions of each are to be maintained (insert, updated, deleted and searched) by the admin
* Through the Technical people login one should be able to keep track of the orders, update the orders with the status (like whether the place is feasible or not, whether provided with the connection or not), create new connections (based on the orders received, once the area is feasible, the customers will be provided with the connection), making a connection inactive (both temporarily, or permanently). Also the technical people will maintain the details of the products or the equipments.
* Based on the plan chosen by the customer, the customer will be charged for the connection.
* The system should be updated with the details from time to time along with the new queries and the new products

**►Financial:**

The bill is to be generated by the accounts department and should be updated in the application so that one can access the bill through the website. The payment details should be clearly mentioned by the system indicating the amount paid and due amount. And the details of the charges levied for a particular plan, the details of the plans are to be maintained by the admin.

## Boundaries of the System

## Development Environment

**Software**

* Visual Studio
* Postman
* Microsoft SQL Server 2017

# Requirements and Business Flows

## Customer Requirements

Our client wants the application to perform the following tasks :

● Customers can:

1. Visit website

2. Register or login to website.

3. View product.

4. view history order.

5. update profile.

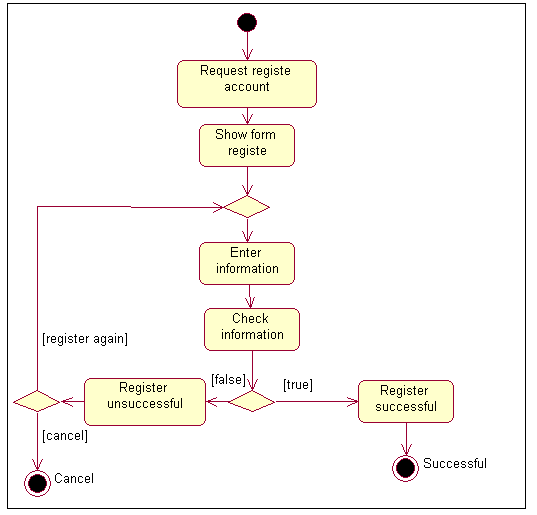
6. View about us

● Admin can manage:

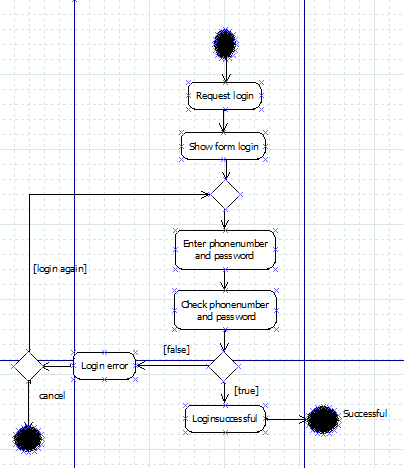
1. Details of retail stores
2. Details of the employees
3. Details of customers, vendors are needed to be updated.
4. Details of the orders generated
5. Details of the products
6. Details of the materials provided with the customer
7. Details of the feedback

Activity Diagram

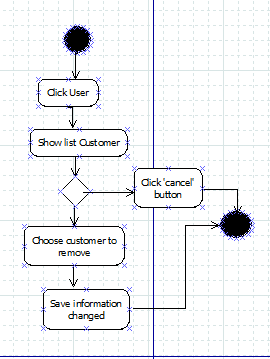
* Activity diagram “ Register”



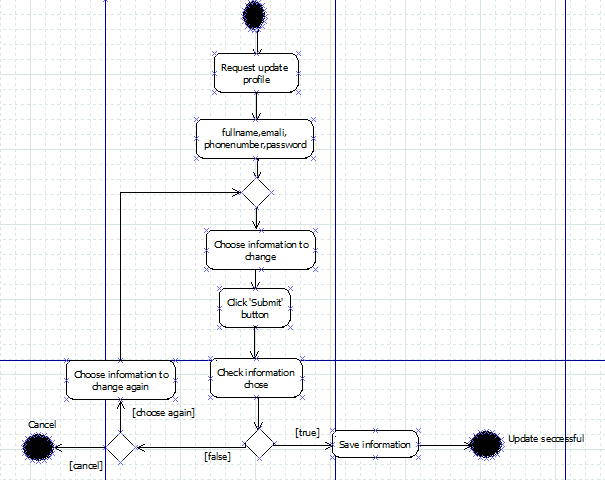
* Activity diagram “Login”



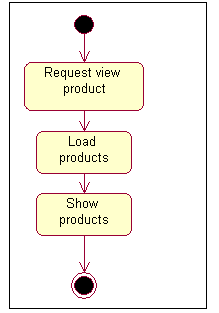
* Activity diagram “Manage customer “



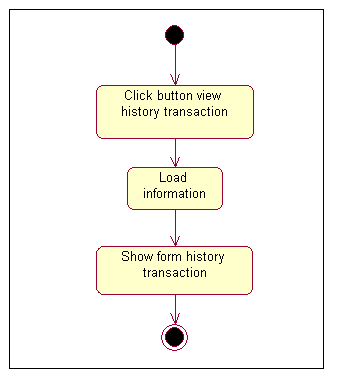
* Activity diagram “ Update profile ”



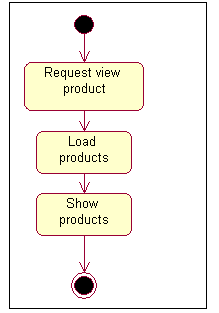
* Activity diagram “view produce”



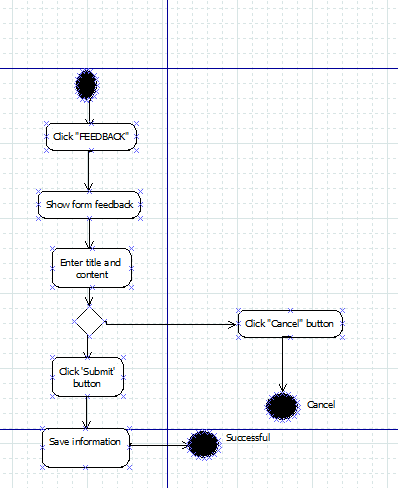
* Activity diagram “View history transaction ”



* Activity diagram “view produce”



* Activity diagram “ Send feedback“



## Use Case Specification

|  |  |
| --- | --- |
| **Use Case Name** | Register |
| **Actors** | Customer |
| **Brief Description** | - Use case allows the customer register to system |
| **Preconditions** | - The person has not already been registered with System |
| **Basic Flow of Events** | 1. The use case begins when customer click on “register “ button in website.  2.The system show register form contains  Fields:   * PhoneNumber * Password. * FullName. * Email   3. Customer enter information into form.  4.The system checks the validity of the information.  5. Save information of customer to database |
| **Alternative Flows** | A1.**Exist use name:**  If in step 4 of the basic flow customer enter a usename existed in the database then:   1. System display error message. 2. The use case resumes at step 3 of Basic Flow enter other username.   A2.**Email not valid:**  If in step 4 of the basic flow customer enter a usename existed in the database then:   1. System display error message. 2. The use case resumes at step 3 of Basic Flow re-enter email address.   A3.(**Full name, password) is null:**  If in step 4 of the basic flow customer not enter (Full name, password)then**:**   1. The use case resumes at step 3 of Basic Flow enter requiment information. |
| **Exit Options** | - Customer click on “Cancel” button |
| **Special Requirements** |  |
| **Post Condition** | - Customer have a new account in system. |

|  |  |
| --- | --- |
| **Use Case Name** | **Login** |
| **Actors** | Customer, admin. |
| **Brief Description** | -Actors use this use case to login to system |
| **Preconditions** |  |
| **Basic Flow of Events** | 1. Actors click on “Login “ button in website.  2.The system show” login” form contains  Fields :   * PhoneNumber. * Password.   3. Actors enter phonenumber and password into form.  4.The system checks the validity of username and password .  5.System logs the Actors into the system. |
| **Alternative Flows** | If in the step 4 Basic Flow,Customer enter invaild user name and/or password then:  1.System display error message.  2.The use case resumes at step 3 of Basic Flow. |
| **Exit Options** | - Actors click on “Cancel” button |
| **Special Requirements** |  |
| **Post Condition** | - Actors login to system |

|  |  |
| --- | --- |
| **Use Case Name** | **Update profile** |
| **Actors** | Customer,admin |
| **Brief Description** | -Customer use this use case to update personal information to system |
| **Preconditions** | The Customer must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. The use case begins when customer click on “Update information” menu on the website.  2.The system load personal information of customer.  3.The system display information into ” Update” form contains Fields:   * FullName. * Email * PhoneNumber * Password   4. Customer change information in form.  5.Customer click on “submit” button.  6.The system checks the validity of the information.  7.System save information to data base.  8. System display successful message. |
| **Alternative Flows** | If in the step 6 Basic Flow,Customer enter invaild infromation:  1.System display error message.  2.The use case resumes at step 4 of Basic Flow. |
| **Exit Options** | - Customer click on “Cancel” button |
| **Special Requirements** |  |
| **Post Condition** | - Personal information of customer is updated to data base. |

|  |  |
| --- | --- |
| **Use Case Name** | View history transaction |
| **Actors** | Customer |
| **Brief Description** | - Use case allows the customer view history transaction |
| **Preconditions** | -The Customer must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. The use case begins when customer click on “View history transaction “ menu in website.  2.System load transactions make by customer .  3.System display result to customer. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** | - History transaction of customer is displayed. |

|  |  |
| --- | --- |
| **Use Case Name** | View Product |
| **Actors** | Customer |
| **Brief Description** | - Use case allows the customer view list products on the website. |
| **Preconditions** |  |
| **Basic Flow of Events** | 1. The use case begins when customer click on “Product“ menu in the website.  2.System load products in database .  3.System display result to customer. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** | - list products are displayed. |

|  |  |
| --- | --- |
| **Use Case Name** | Rating products. |
| **Actors** | Customer |
| **Brief Description** | - Use case allows the customer rate a product. |
| **Preconditions** | -The Customer must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. This use case begin when customer click on star . 2. System save rating value into database.   3.System display rating product. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** | - Rating of customer saved to database. |

|  |  |
| --- | --- |
| **Use Case Name** | Send feedback |
| **Actors** | Customer |
| **Brief Description** | - Use case allows the customer send a feedback. |
| **Preconditions** | -The Customer must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. This use case begin when customer click on “send feedback” menu. 2. System display a form contain :  * Title. * Content.  1. Customer enter information into form. 2. Customer click on “Send” button. 3. System save information to database. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** | -A feedBack of customer is sent. |

|  |  |
| --- | --- |
| **Use Case Name** | Process orders |
| **Actors** | Dealer manager |
| **Brief Description** | - Use case allows the dealer process orders |
| **Preconditions** | - Dealer manager must to login into system |
| **Basic Flow of Events** | 1. The system display order list .  2. Actor click on “detail “ button.  3.The system show detail order .  4. Actor change status of order .  5.Actor click on “Save” button.  6. System save status of order to database  7.System display successful message. |
| **Alternative Flows** | . |
| **Exit Options** | - Dealer manager click on “Cancel” button |
| **Special Requirements** |  |
| **Post Condition** | - The order process and save into database |

|  |  |
| --- | --- |
| **Use Case Name** | Manage Policy |
| **Actors** | Admin |
| **Brief Description** | - Use case allows the Admin manage policy contain: taxes and discount |
| **Preconditions** | -The actor must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. This use case begin when actors click on “**Manage policy**” menu. 2. System display list all policy 3. Admin enter taxes policy, discount policy in to textbox. Then click update button. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** | - Policy is added and updated |

|  |  |
| --- | --- |
| **Use Case Name** | Manage customer |
| **Actors** | Admin |
| **Brief Description** | - Use case allows the Actors manage customer contain: remove customer. |
| **Preconditions** | -The actor must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. This use case begin when actors click on “manage customer” menu. 2. System display list all customer in database and button “remove customer”. 3. Actor chose customer want to remove then click on “Remove” button. 4. System remove customer from database. 5. System display success message. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** | - A customer is removed |

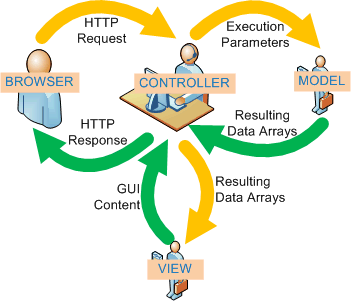
|  |  |
| --- | --- |
| **Use Case Name** | View rejection product |
| **Actors** | Admin |
| **Brief Description** | - Use case allows the Actors view rejection product. |
| **Preconditions** | -The actor must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. This use case begin when actors click on “rejection product” menu. 2. System display list all rejection product in database and button “remove”. 3. Actor chose rejection product want to remove then click on “Remove” buttion. 4. System remove rejection product from database. 5. System display success message. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** |  |

|  |  |
| --- | --- |
| **Use Case Name** | View feedback |
| **Actors** | Admin |
| **Brief Description** | - Use case allows the admin view feedback of customer. |
| **Preconditions** | -The actor must be logged into system before this use case begins. |
| **Basic Flow of Events** | 1. This use case begin when actors click on “view feedback” menu. 2. System display list all feedback. 3. Actor choose feedback want to view detail or remove feedback 4. System display success message. |
| **Alternative Flows** |  |
| **Exit Options** |  |
| **Special Requirements** |  |
| **Post Condition** |  |

# Design

## System Architecture

Website bases on MVC (Model-View-Controller) architecture



Web-Based Distributed MVC Architecture of the Project

## Class Diagram

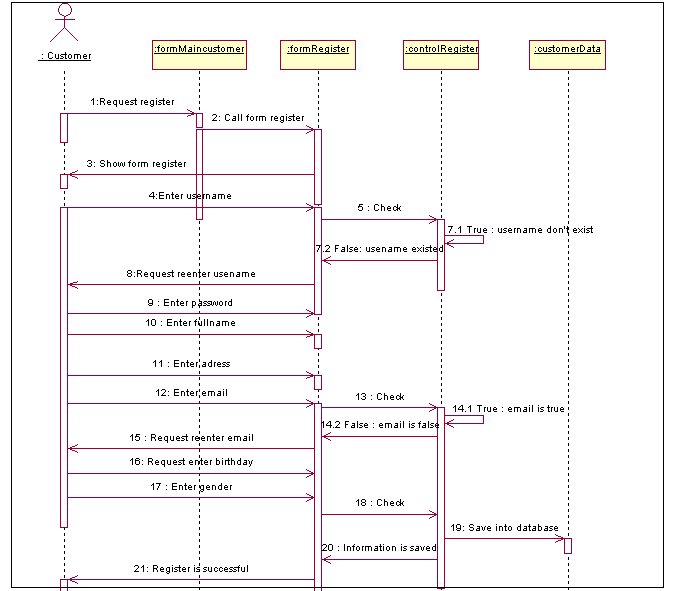
*<Provide class diagrams for the project>*

## Class Diagram Explanation

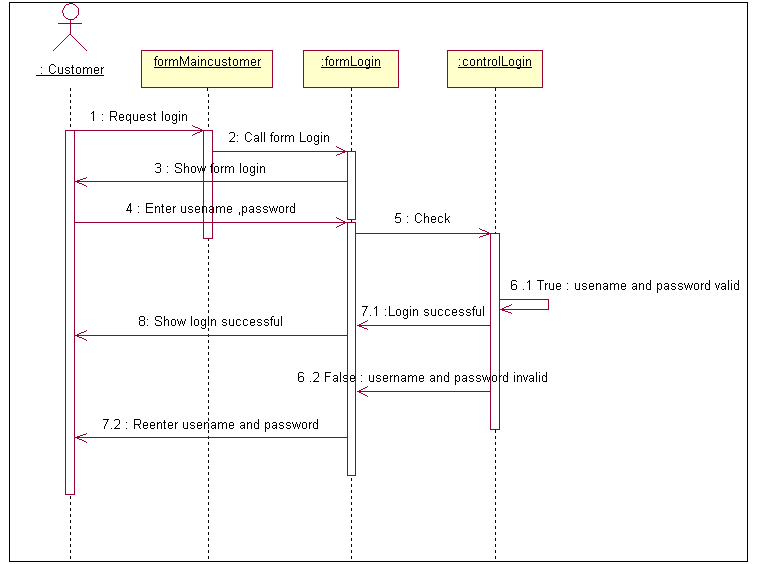
*<Provide brief explanation about the class diagram above. You do not need to explain “obvious” parts of your class diagram. For example, I know what a “Login” class is. Don’t say “The login class was created to store login information.”>*

## Entity Relationship Diagram

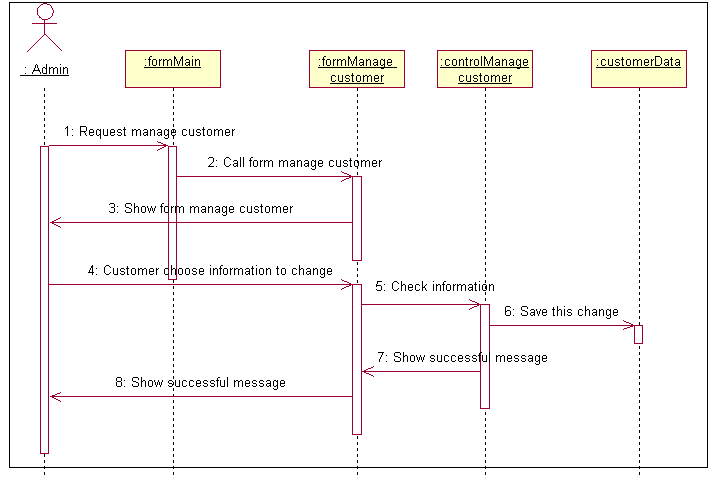
* Sequence diagram “ Register ”



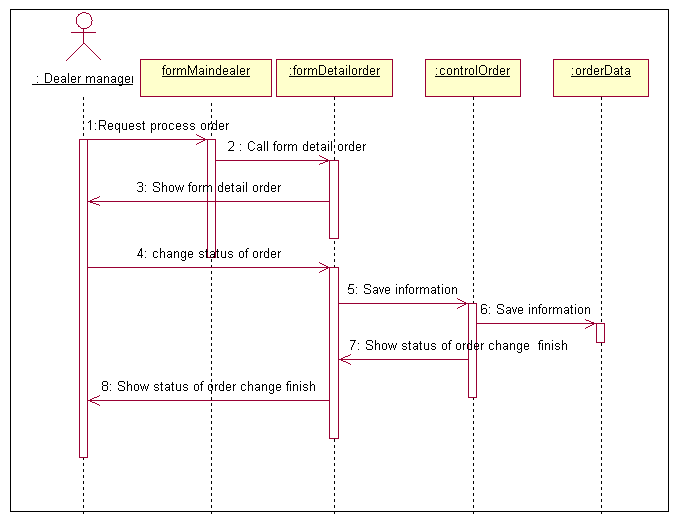
* Sequence diagram “Login”



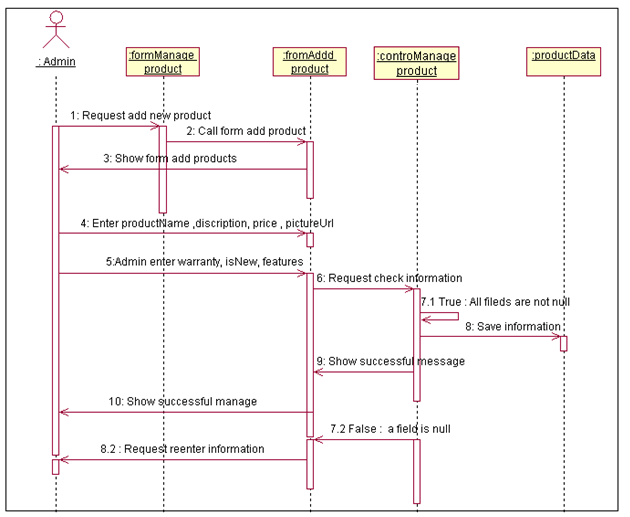
* Sequence diagram ”Manage customers “



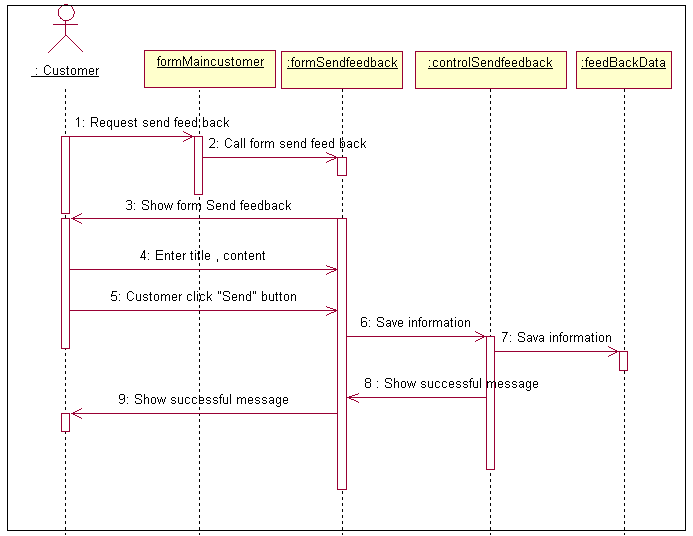
* Sequence diagram “Process order”



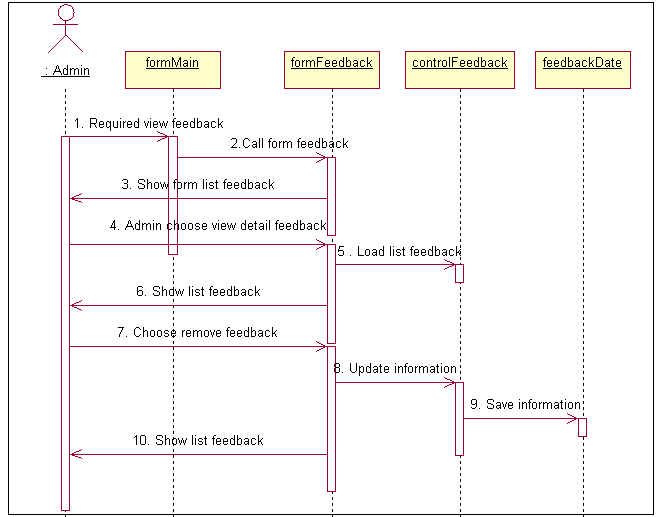
* Sequence diagram “Add product “



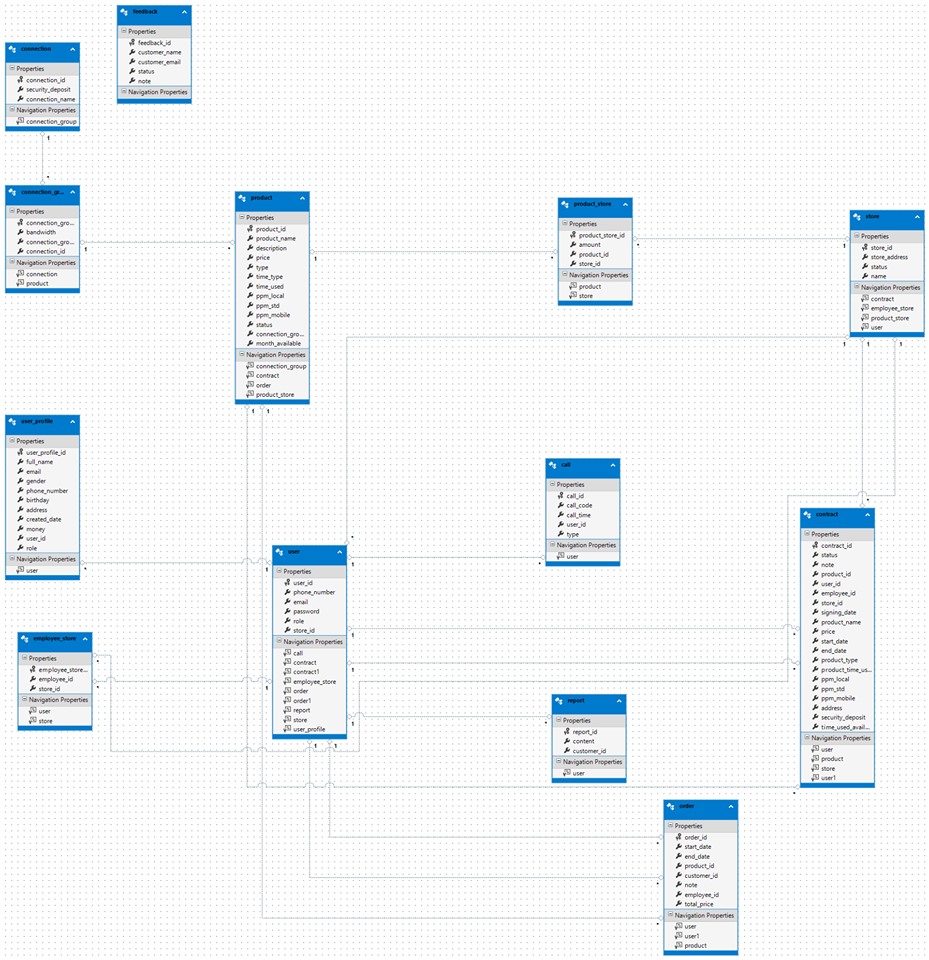
* Sequence diagram “Send feedback “



* Sequence diagram “ View feedback”



## Entity Relationship Diagram



## Database Design

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. **Table :products** | | | | |
| **Field Name** | **Data Type** | **Null** | **Key** | **Description** |
| productId | Int | No | PK | ID of product |
| productName | nvarchar(255) | No |  | Name of product |
| discription | nvarchar(255) | No |  | Description of product |
| price | Decimal | No |  | Price of product |
| type | int | No |  | Type of product |
| TimeType | Int | No |  | TimeType of product |
| TimeUsed | Int | No |  | Time Used of product |
| PpmLocal | decimal | yes |  | PpmLocal of product |
| PpmStd | decimal | yes |  | PpmStd of produce |
| TimeUser | decimal | yes |  | TimeUser of product |
| Status | Int | No |  | Status of product |
| Role | int | no |  | Role of product |
| MonthAvailable | int | no |  | MonthAvailable of product |
| ConnectionGroupId | Int | no |  | ConnectionGroupId of product |

|  |
| --- |
| **Table:User\_profile** |
| **Field Name** | **Data Type** | **Null** | **Key** | **Description** |
| userprofileId | Int | No | PK | userprofileId of User\_profile |
| PhoneNumber | String | No |  | PhoneNumber of User\_profile |
| FullName | String | No |  | Full name of User\_profile |
| BirthDay | datetime | no |  | Birthday of User\_profile |
| Gender | Bit | no |  | Gender of c User\_profile |
| money | int | No |  | money of User\_profile |
| Address | nvarchar(255) | No |  | Addess of  User\_profile |
| email | nvarchar(255) | No |  | Email of User\_profile |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table : Store** | | | | |
| **Field Name** | **Data Type** | **Null** | **Key** | **Description** |
| StoreId | Int | No | PK | Store of Store |
| Address | nvarchar(255) | No |  | Address of Store |
| Status | Int | Yes |  | Status of Store |
| Name | nvarchar(255) | No |  | Name of Store |

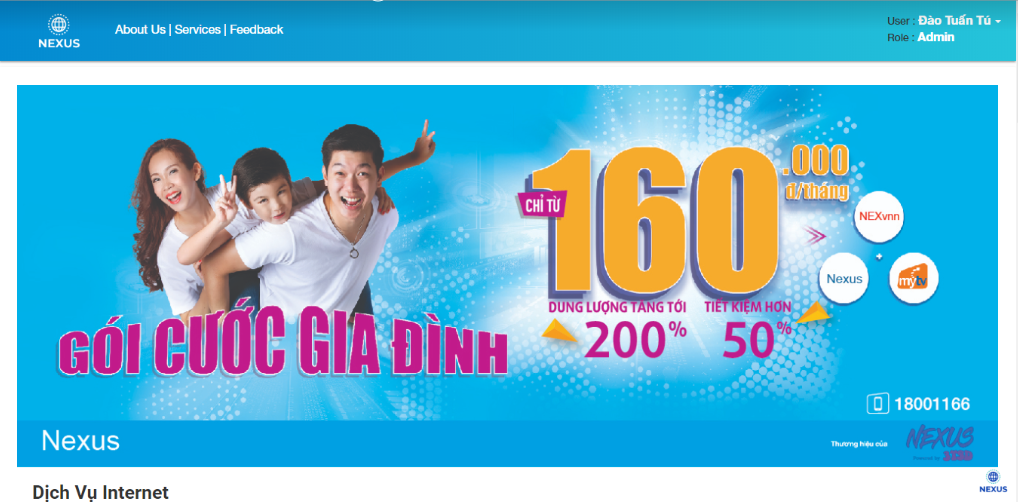
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table : employee\_store** | | | | |
| **Field Name** | **Data Type** | **Null** | **Key** | **Description** |
| employee\_store\_Id | Int | No | PK | employee\_store\_Id of employee\_store |
| employee\_id | int | no |  | employee\_id of employee\_store |
| Store\_Id | int | No | FK | Store\_Id of store who send  employee\_store |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table : Feedback** | | | | |
| **Field Name** | **Data Type** | **Null** | **Key** | **Description** |
| feedbackId | Int | No | PK | FeedbackId of feedback |
| CustomerId | Int | No | FK | CustomerId of customer who send feedback |
| Note | Nvarchar(max) | No |  | Note of customer |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table : Contract** | | | | |
| **Field Name** | **Data Type** | **Null** | **Key** | **Description** |
| contractId | Int | No | PK | contractId ofContract |
| Status | Int | Yes |  | Status of Contract |
| productId | Int | No | FK | productId of Contract |
| employeeId | Int | No | FK | employeeId of  employee\_store who send Contract |
| UseriId | Int | No | FK | UseriId of User who send  Contract |
| StoreId | Int | Yes | FK | StoreId of Store who send Contract |
| Signing\_data | Int | Yes |  | Signing\_data of Contract |
| productName | nvarchar(255) | No |  | productName of Contract |
| price | float | Yes |  | price of Contract |
| Time\_usre | nvarchar(255) | No |  | Time\_usre of Contract |
| Start\_data | nvarchar(255) | No |  | Start\_data of Contract |
| Product\_type | nvarchar(255) | No |  | Product\_type of Contract |
| Address | nvarchar(255) | No |  | Addess of Contract |
| Secunty\_deport | nvarchar(255) | No |  | Secunty\_deport of Contract |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table :User** | | | | |
| **Field Name** | **Data Type** | **Null** | **Key** | **Description** |
| userId | Int | No | PK | ID of User |
| PhoneNumber | nvarchar(255) | No |  | PhoneNumber of User |
| Password | nvarchar(255) | No |  | Password of User |
| FullName | nvarchar(255) | No |  | Full name of User |
| UserName | nvarchar(255) | No |  | UserName of User |
| BirthDay | Int | Yes |  | Birthday of User |
| Gender | Int | Yes |  | Gender of c User |
| Role | Int | No |  | Role of User |
| Money | Int | Yes | PK | Money of User |
| Token | nvarchar(255) | No |  | Token of User |
| Address | nvarchar(255) | No |  | Addess of User |
| email | nvarchar(255) | No |  | Email of User |

# System Prototype

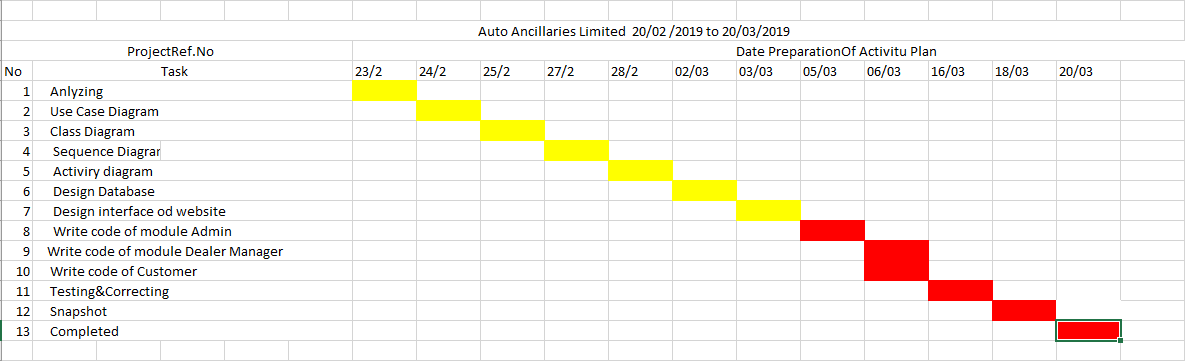
**

# Management and Project Planning

## Management Approach

Group leader will make a list of tasks, and assign tasks for each member of group. If member has any issues about assigned tasks, group will have a meeting to discuss and assign tasks again. After members submit assigned tasks to SVN server, group leader will check each task and give any issues or bugs which member must fix or discuss again with group

## Project Plan



## Task Sheet

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project Ref. No.:** | | **Date of Preparation of Activity Plan:** | | | |
| **No** | **Task** | **Actual Start Date** | **Actual Days** | **Team Member Names** | **Status** |
| 1 | Analyzing | 02-23-2019 | 1 | Hoang Sang | Completed |
| 2 | design of the project | 02-24-2019 | 1 | Hoang Sang | Completed |
| 3 | Use Case Diagram | 02-25-2019 | 2 | Dao Tuan Tu | Completed |
| 4 | Class Diagram | 02-027-2019 | 3 | Vu Tri Thanh | Completed |
| 5 | Sequence Diagram | 02-028-2019 | 2 | Hoang Sang | Completed |
| 6 | Activity diagram | 03-02-2019 | 3 | Vu Tri Thanh | Completed |
| 7 | Design Database | 03-03-2019 | 3 | Hoang Sang | Completed |
| 8 | Snapshot | 03-05-2019 | 2 | Pham Thi Thanh Tuoi | Completed |
| 9 | Write code of module Admin | 03-06-2019 | 8 | Hoang Sang | Completed |
| 10 | Write code of module Dealer Manager | 04-06-2019 | 8 | Dao Tuan Tu | Completed |
| 11 | Write code of Customer | 04-14-2019 | 4 | Dao Tuan Tu | Completed |
| 12 | Testing & Correcting | 04-18-2019 | 6 | Pham Thi Thanh Tuoi | Completed |

# Checklists

## Check List of Validation

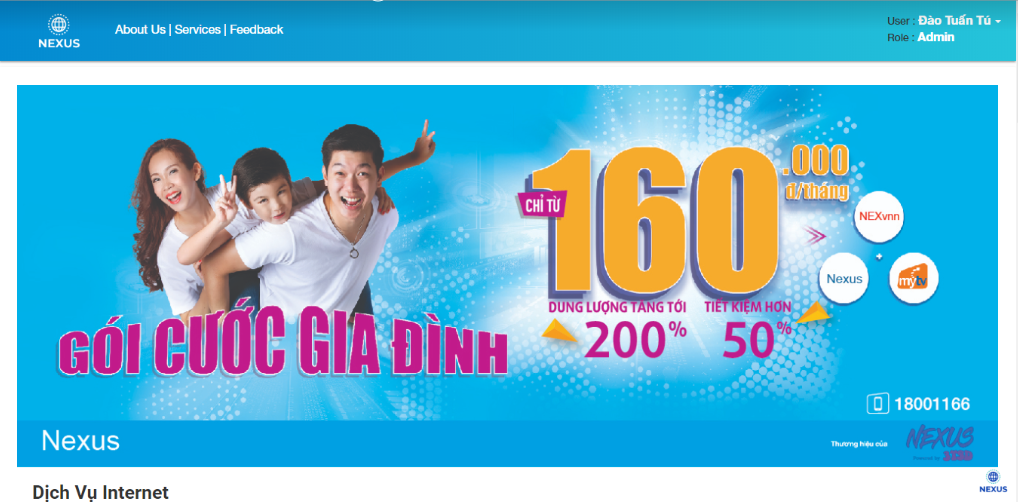
|  |  |
| --- | --- |
| **Checklist Option** | **Validated** |
| Can a new user who gets registered, enter the web site after logging in? | Yes |
| Do all the links navigate to the correct web page? | Yes |
| Does the web site’s functionality resolve the client problem and satisfy his need? | Yes |
| Has the hardware and software been correctly chosen? | yes |

## Submission Checklist

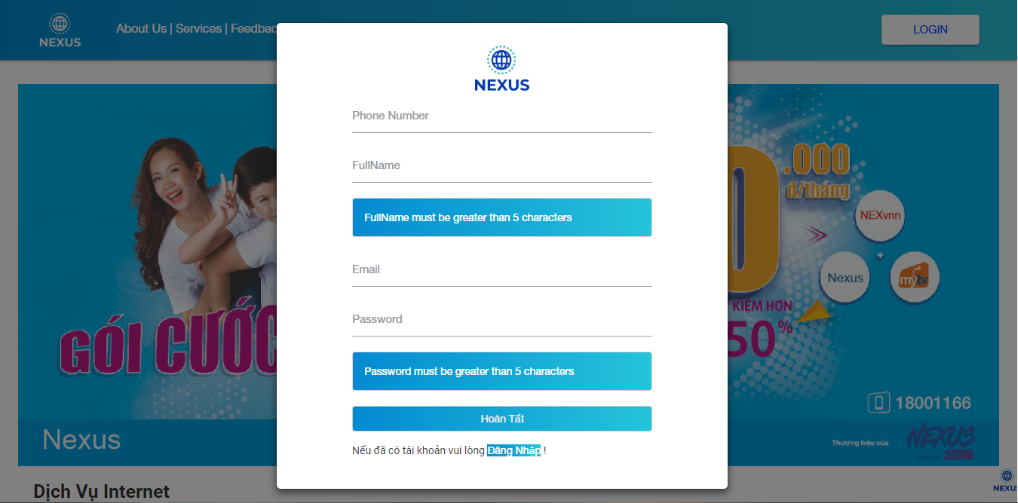
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1. **Sr.No** | **Particulars** | **Yes** | **No** | **NA** | **Comments** |
| 1 | Are the users able to enter the web site after validation is performed on the UserID and password? | Yes |  |  |  |
| 2 | Are the users able to enter the web site after getting registed? | Yes |  |  |  |
| 3 | Do all the web page contents are devoid of spelling mistakes? | Yes |  |  |  |
| 4 | Is the user able to post entry to the web site | Yes |  |  |  |
| 5 | Do all user and visitor able to view forum |  | No |  |  |
| 6 | Admin able manage activity of forum | Yes |  |  |  |
| 7 | Is the web site user- friendly | Yes |  |  |  |

# Screenshots

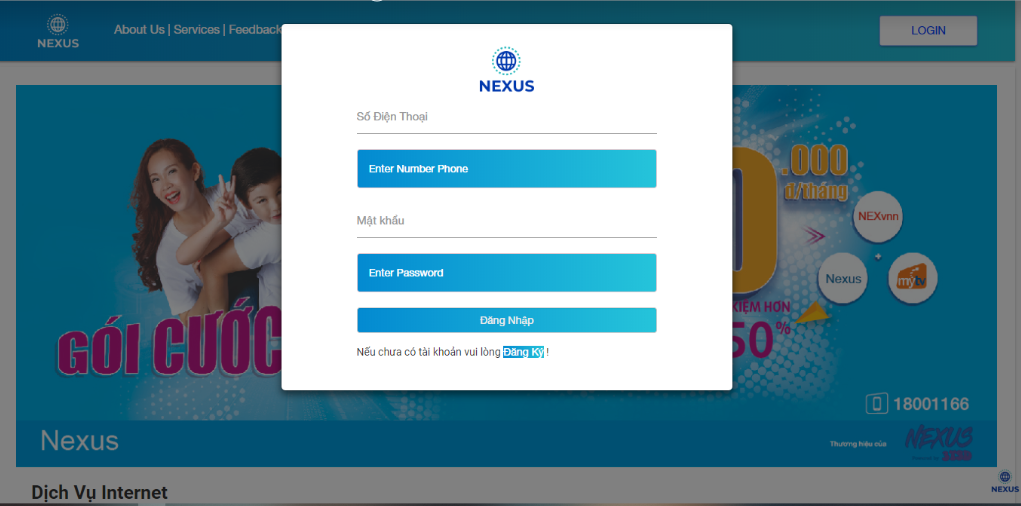
Main Customer



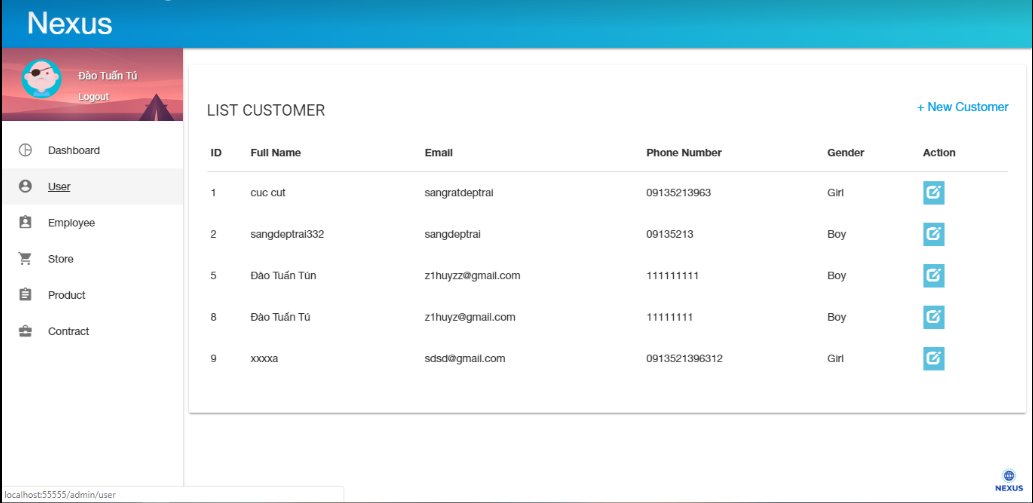
Registration



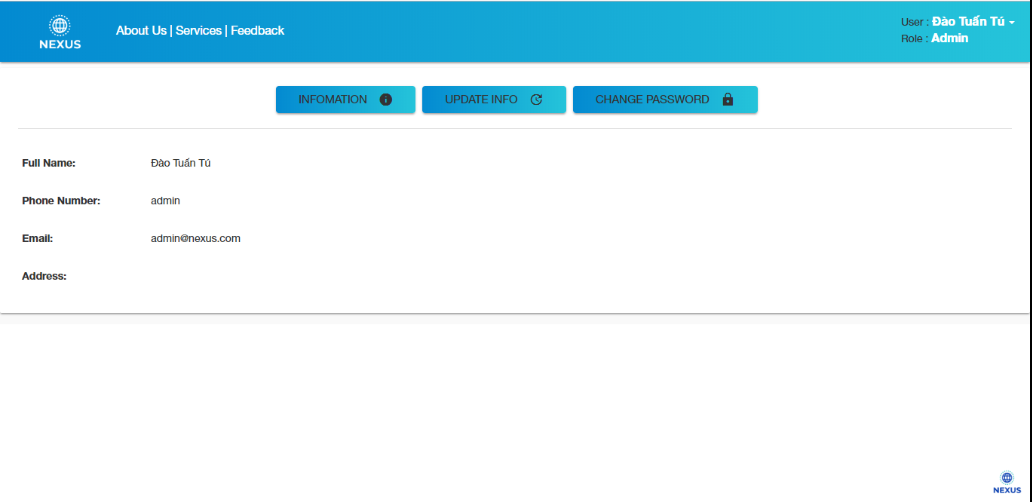
Login



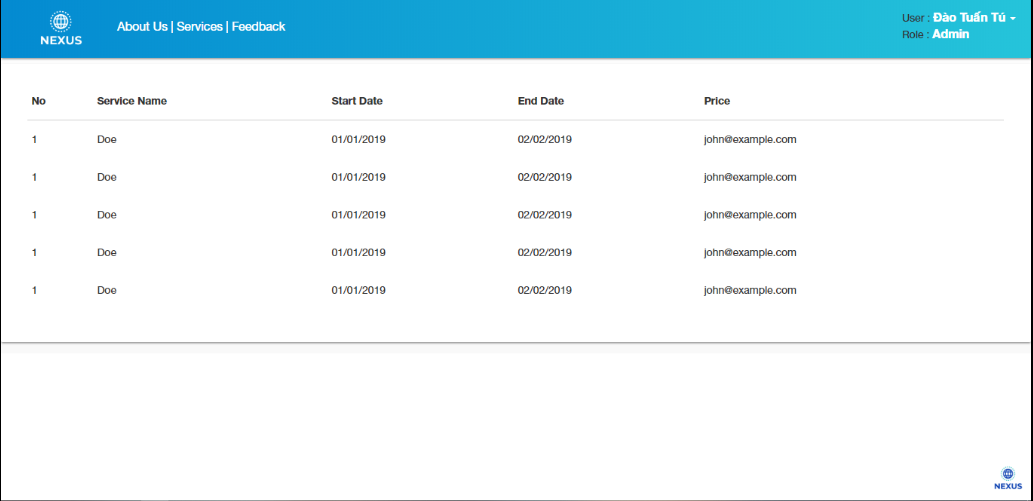
List Customer



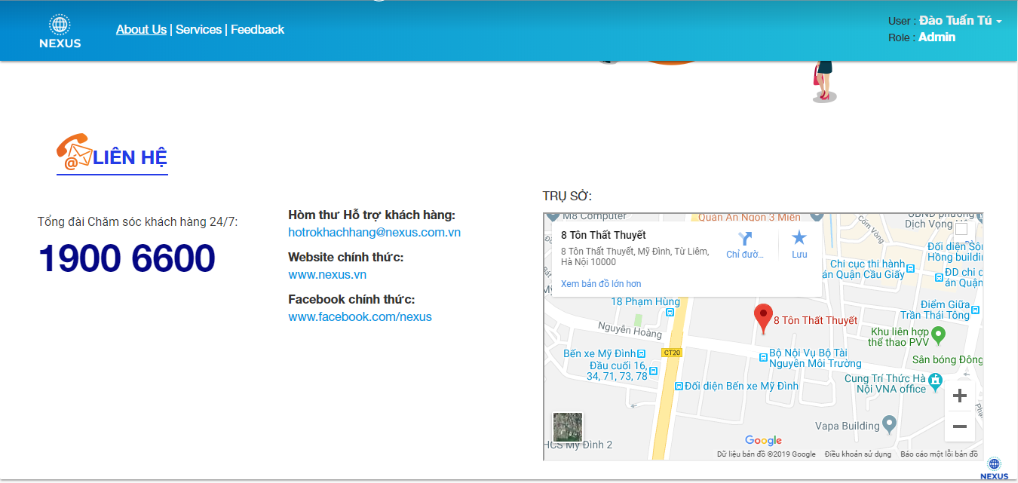
User- Profile



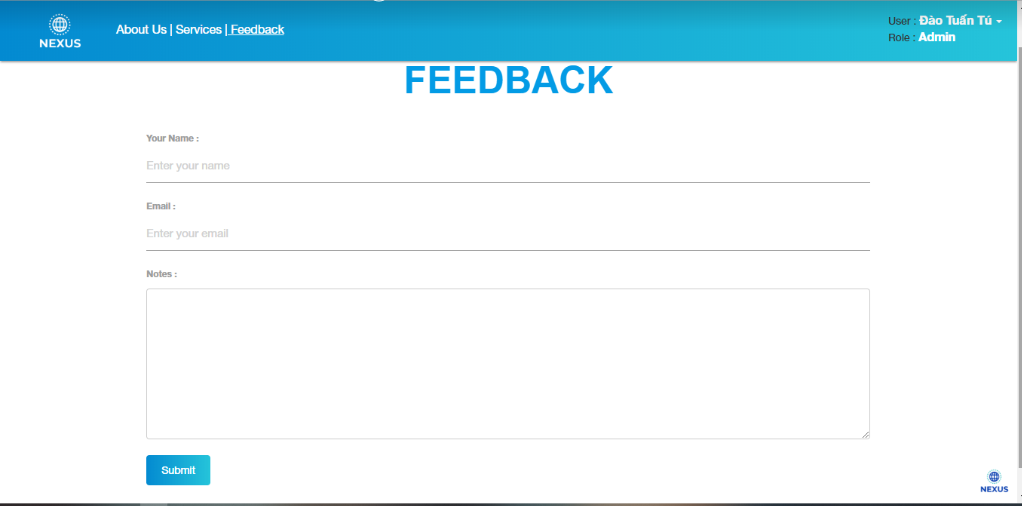
History



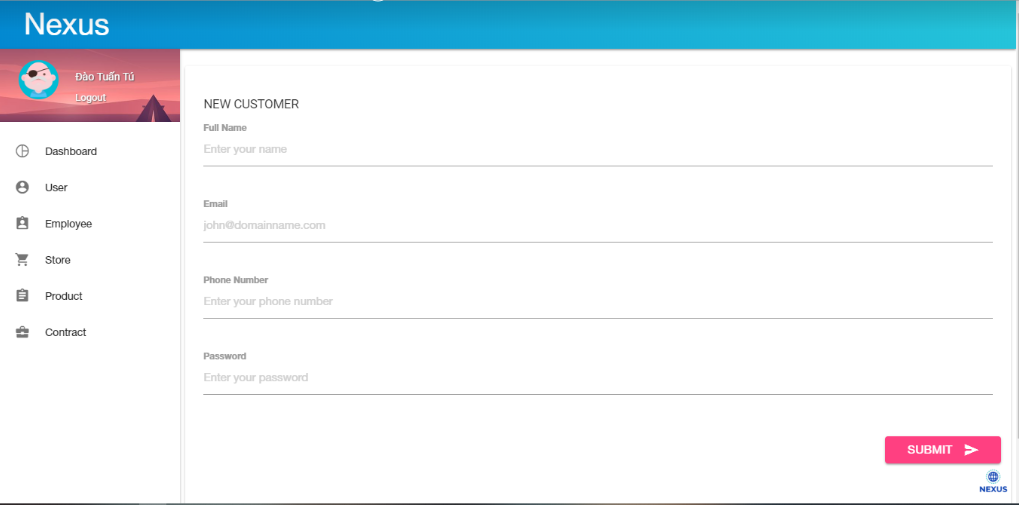
About-us



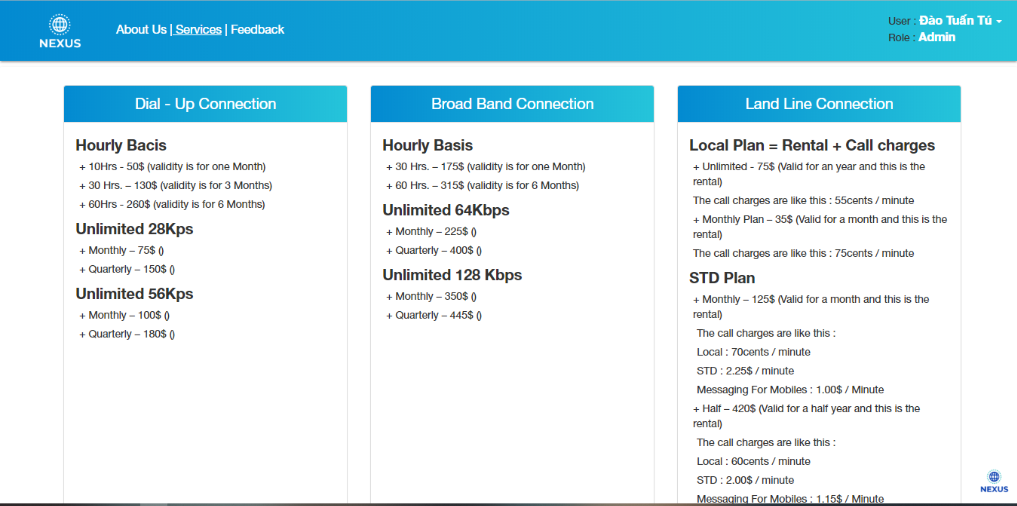
FeedBack



New Customer(admim)



Product



# Coding Convention

# User Guide