**Assignment 3**

Contents

[Task 1: 2](#_Toc533240133)

[Analysis: 2](#_Toc533240134)

[\_Use-case Diagram: 2](#_Toc533240135)

[Design: 3](#_Toc533240136)

[\_ERD: 3](#_Toc533240137)

[Class diagram(Functional) 4](#_Toc533240138)

[Flowchart: 4](#_Toc533240139)

[Gui design: 5](#_Toc533240140)

[Implementation: 6](#_Toc533240141)

[DBMS: 6](#_Toc533240142)

[\_Selected Programming language: 7](#_Toc533240143)

[Technology: 9](#_Toc533240144)

[\_IDE 9](#_Toc533240145)

[\_Evaluation: 10](#_Toc533240146)

[User Guide 10](#_Toc533240147)

[Test Case: 14](#_Toc533240148)

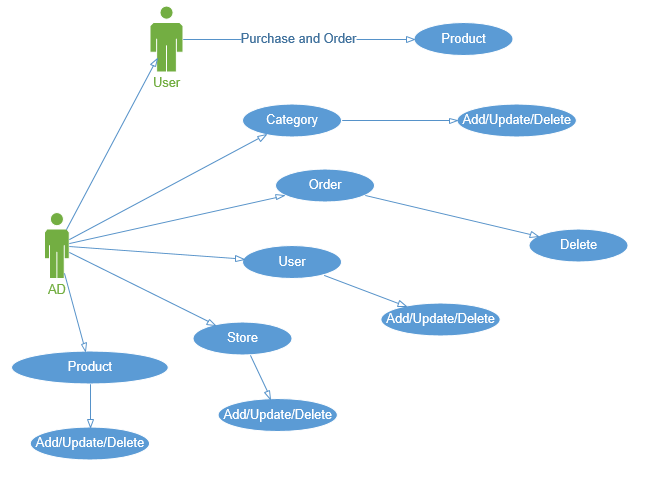
[Questionnaire and Peer feedback evaluation 16](#_Toc533240149)

[Task 2: 17](#_Toc533240150)

# Task 1:

## Analysis:

### \_Use-case Diagram:



\_Use-case Description:

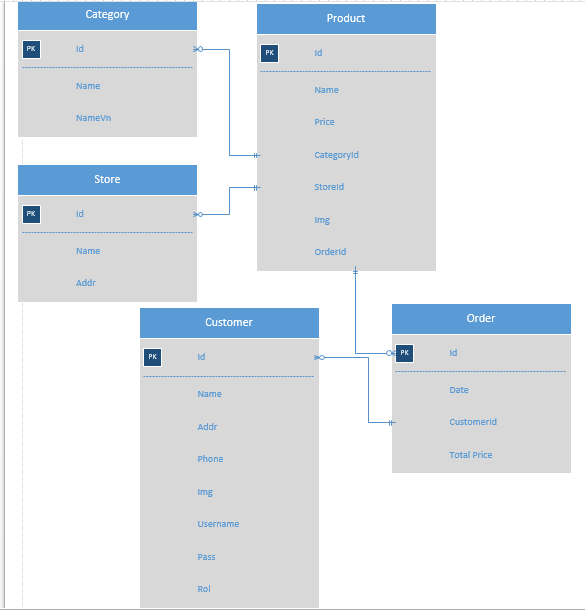
+Customers can order through the ordering system with different types. From there the system will execute the customer orders through the information that the customer has just logged in. Ultimately, the system will transmit the order data including product information and customer information about the database.

+As for the administrator, they will have the full right to use the database system in the project and have the right to decide whether to update or delete any data in the database. In the use-case diagram, tables like Order, Product, Category, or even their customers can all hold information.

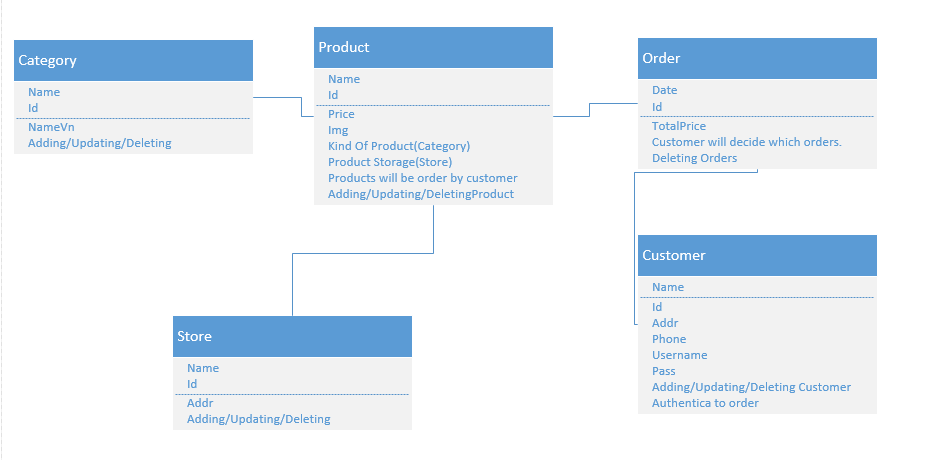
+And there is only a small note in the administration that they cannot edit or add any orders into the database. Since the orders are intentional from the customer and will be accepted by the system if they agree to the bar math. Thus, the Order table is set for a function that deletes or delays the order.

## Design:

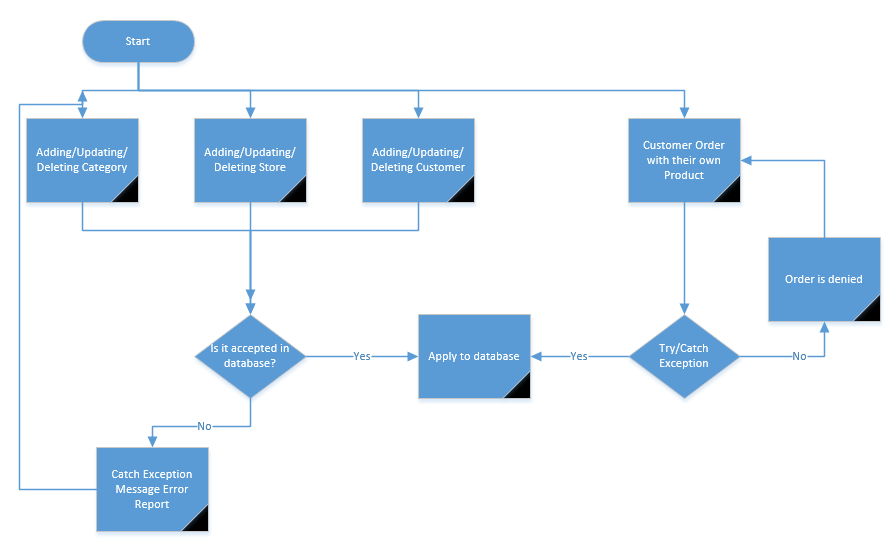
### \_ERD:



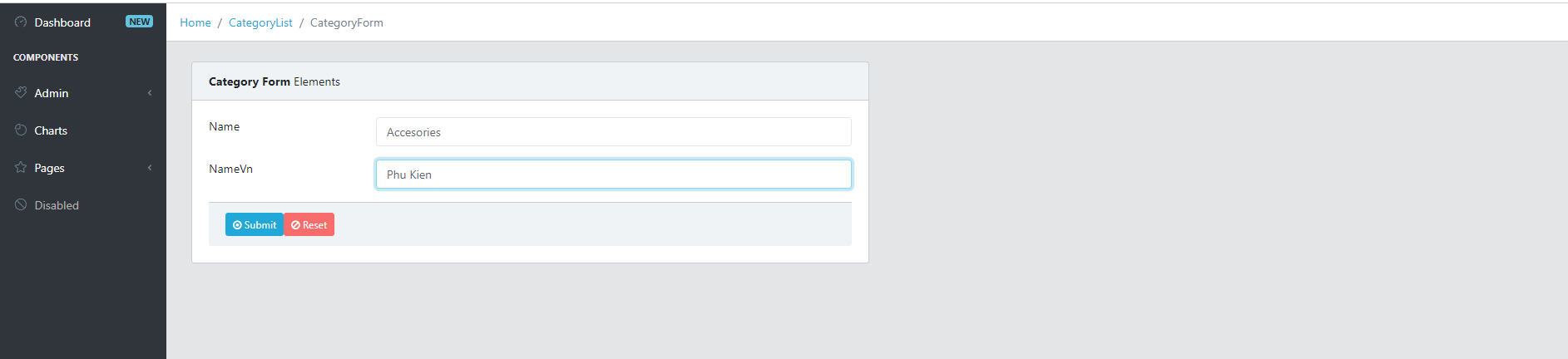
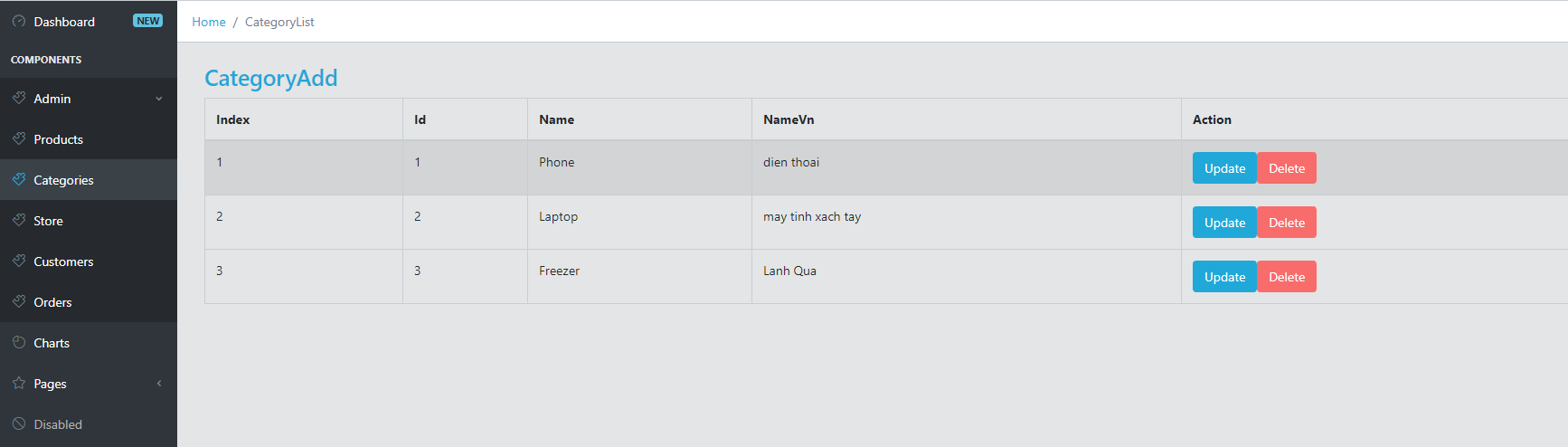
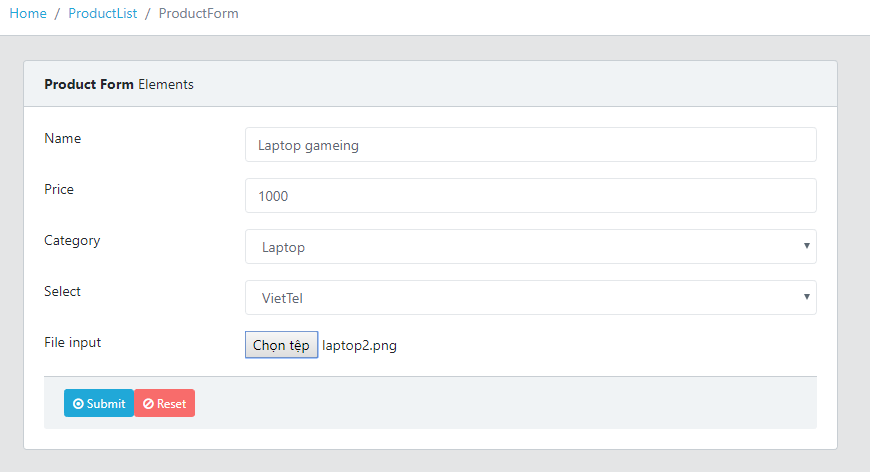
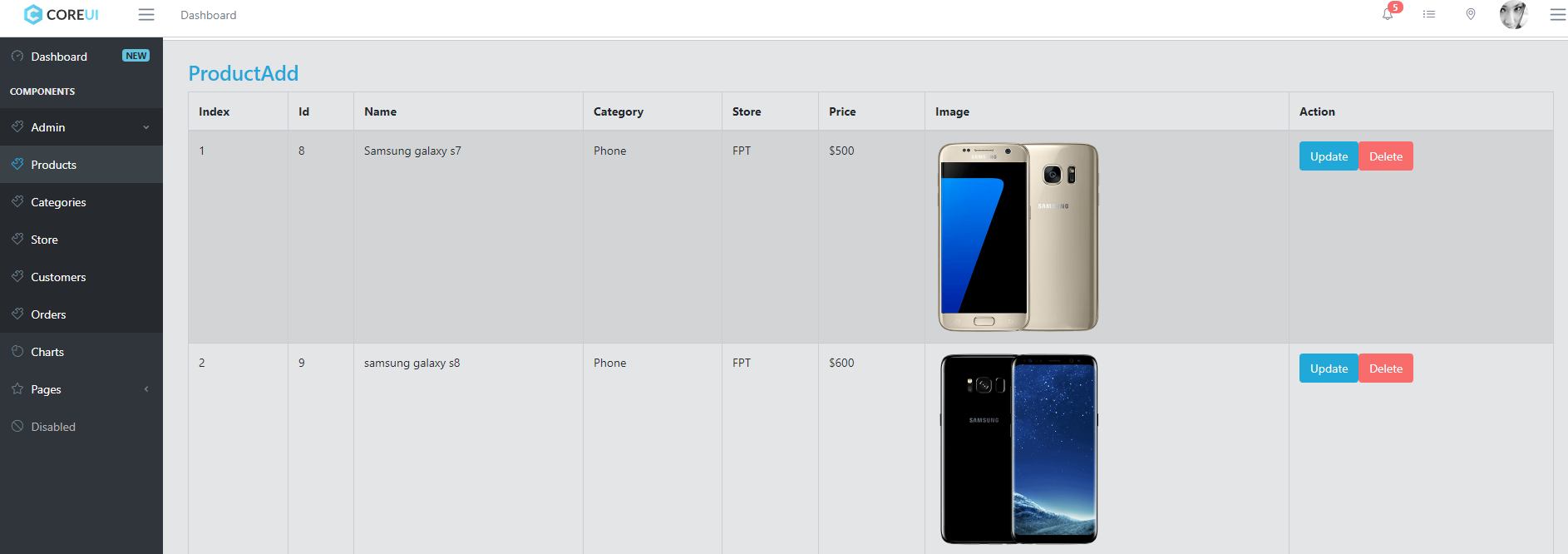
### Class diagram(Functional)



### Flowchart:



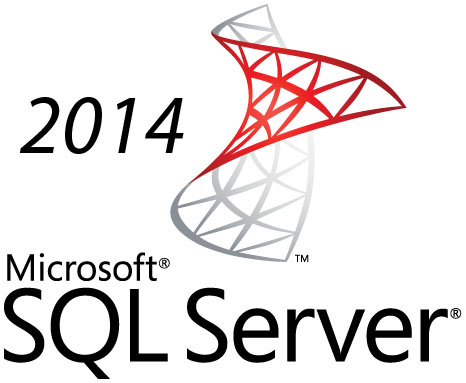
## Gui design:



## Implementation:

### DBMS:

Sql server 2014



### \_Selected Programming language:

C#



Javscript



Html

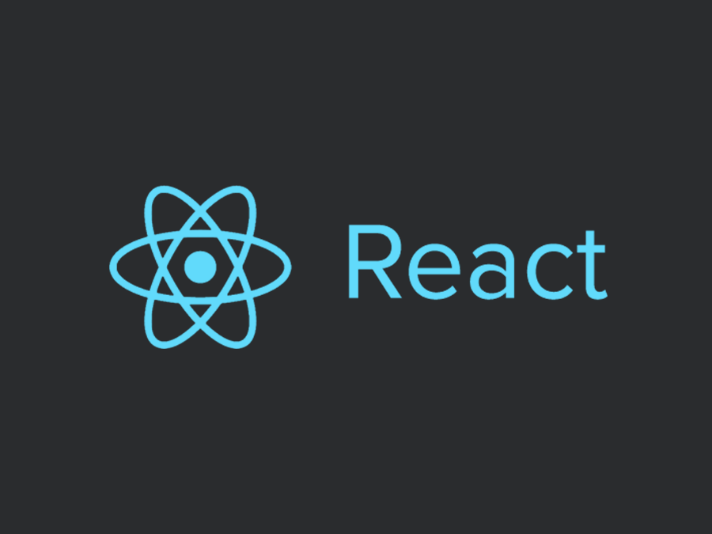


Css



### Technology:

ReactJS(Front end)

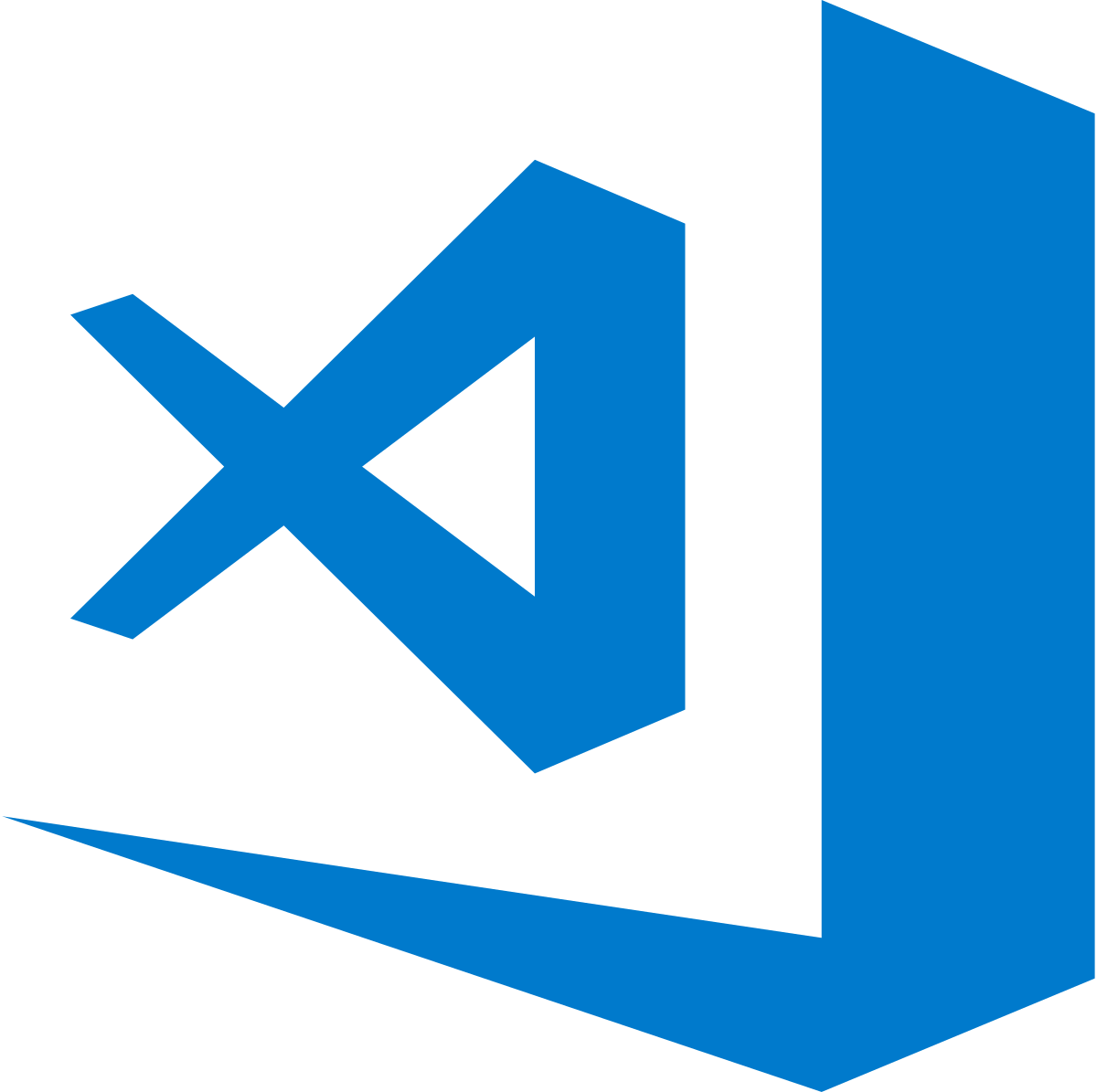


Microsoft web api mvc (Back end)



### \_IDE

Visual Studio Code



Visual studio 2017:



## \_Evaluation:

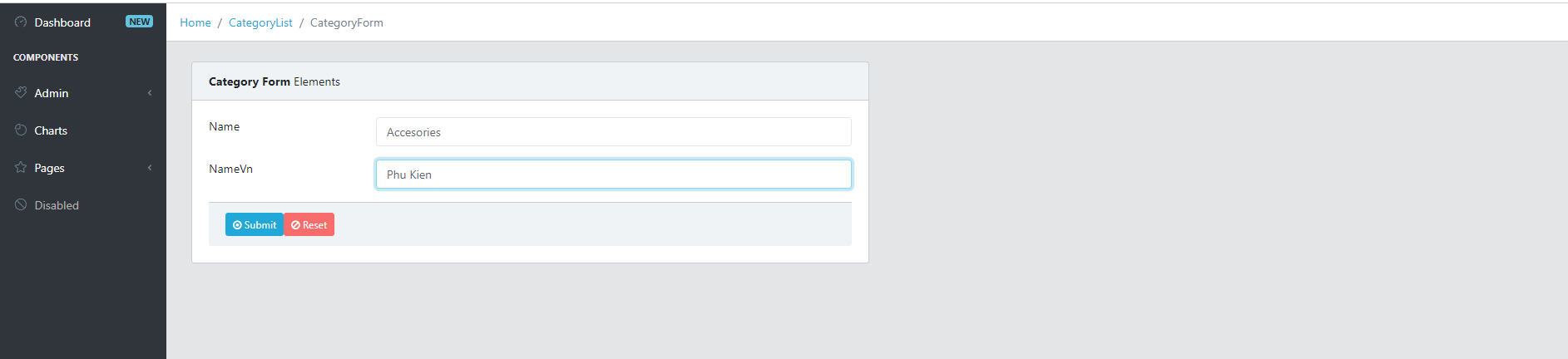
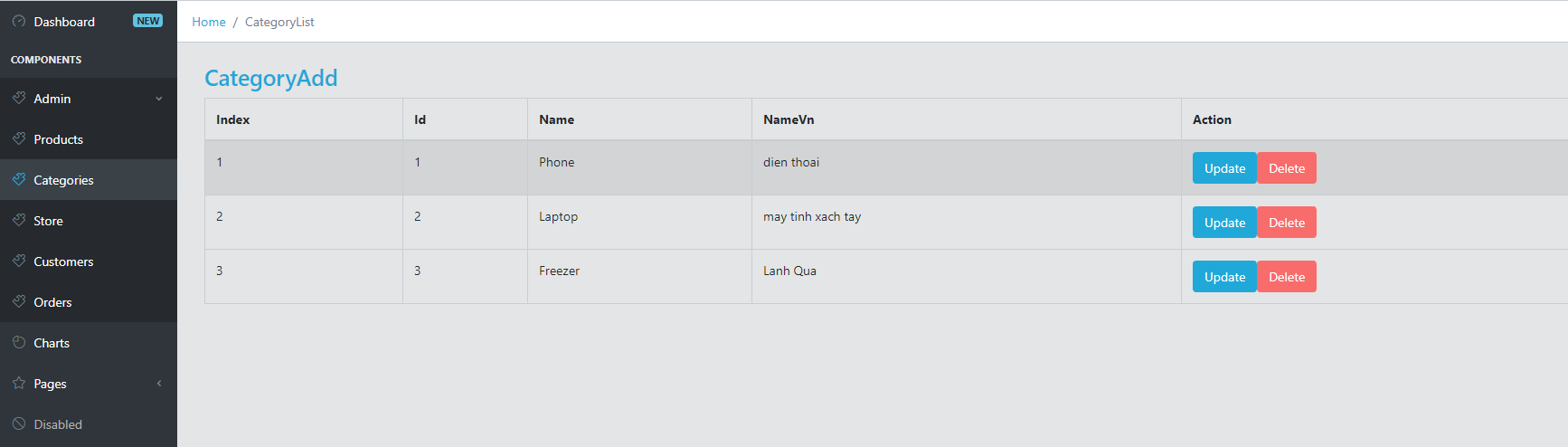
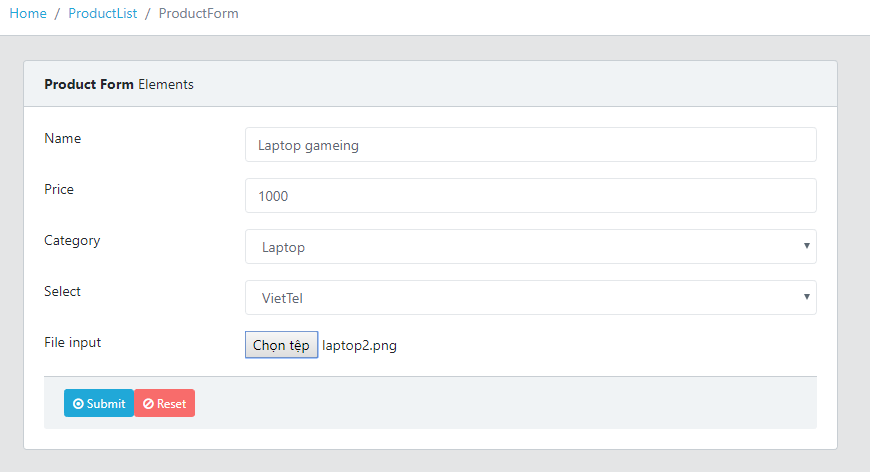
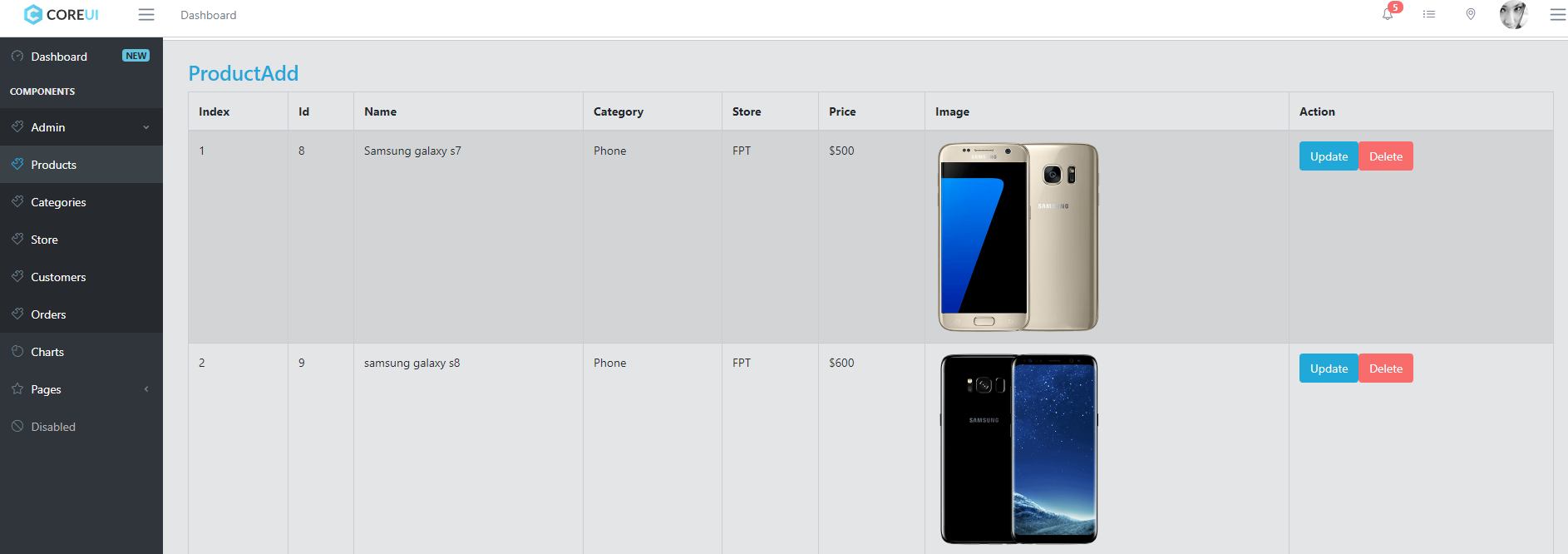
### User Guide

For take after the association's need. The association has many stores wherever all through the country. They give magnificence things. Since they have variety of things and amounts of stores. They might need to tie together their store systems to have the ability to record purchases, make reports for the things sold step by step/step by step (or any predefined date go).

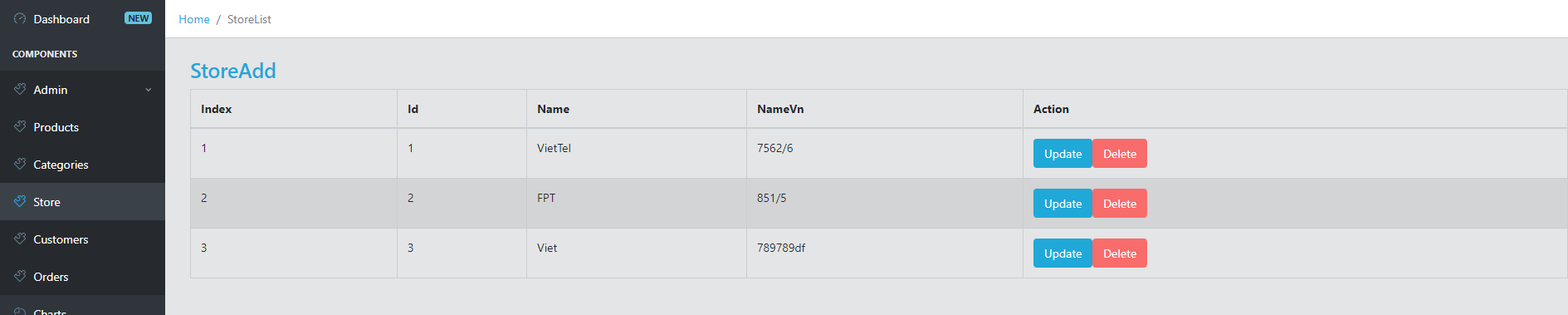
There are two essential sorts of customers

Arrangements staff: they can record demands and enroll portions.

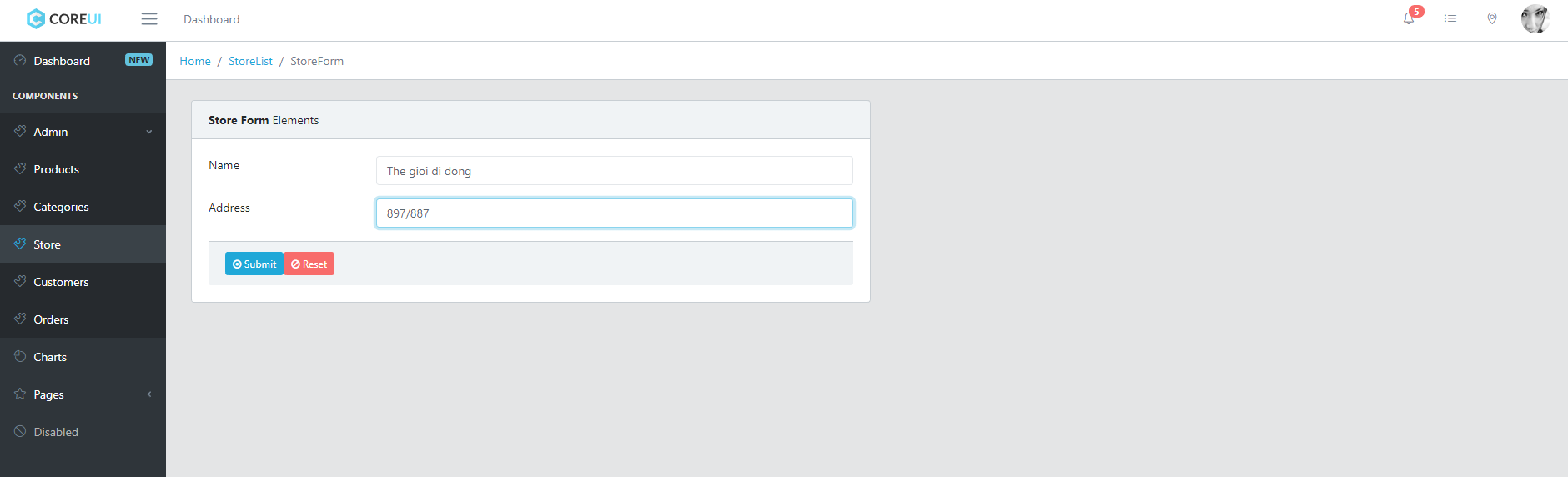
National director: They may be arranged at any store and they may have the ability to make expounds on consistently/step by step (or from any predefined date go) concerning trades held and the things sold, for a specific staff or for all staff, for a specific store, or for all stores. We was influenced a business to web by scatter structure and response's association as essential, administer system. By then we do the guide document empower customer to can use convincing.



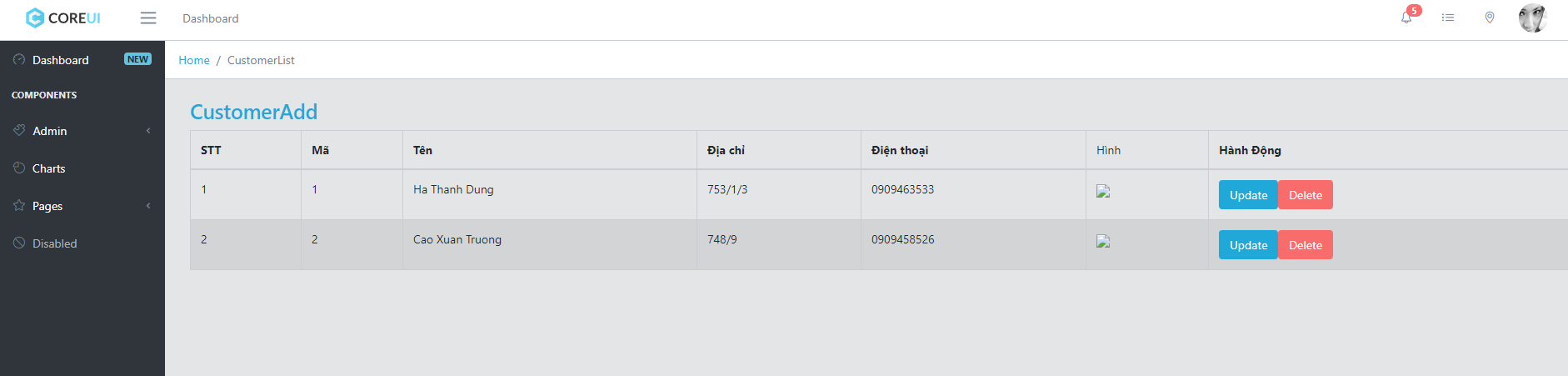
First is manage category, same as product we can add new category after we add new category will have new category in category bar.



Second is manage supplier, we can add new supplier after we add new supplier will have new supplier in supplier bar



This is where you can see all order that the store had.



Revenue chart:



### Test Case:

1. Extent of The Test:

1.1 The site produces for cosmetic

1.2 The reports: Design archive, Test record, Practice code, Test code

2. Testing Environment:

2.1 Testing locales: All controller organizer, element envelope, session organizer

2.2 Required equipment and firmware setup: Yes

2.3 Participating associations: cosmetic

2.4 Manpower prerequisites: Coding group, Analysis group, Testing group, Design group, Planer.

2.5 Preparation and preparing expected of the test group: No

3. Test Detail (For Each Test):

3.1 Test ID: testing full capacity

3.2 Test objective: Staff, Manager, arrange, report.

3.3 Cross-reference to important outline record and the necessity report: Coding Document, Design Document

3.4 Test prerequisites:

Check and confirm email

Request for client

Oversee account

Report check kind things bought

4. Test Schedule (For Each Test or Test Group) including time gauges for the accompany:

4.1 Preparation: 1 week

4.2 Testing: 1 weeks

4.3 Error rectification: 1 weeks

4.4 Regression test: 1 week

5.Resuft testing for utilized

5.1 Interface be agreeable

5.2 Enough necessity

5.3 The site stacking easily.

5.4 The site make some intriguing for guest

5.5 Checked framework security it was fine

Test Result Evaluation:

A webpage, also created as website, is an aggregation of related website pages, including blended media content, usually identified with a run of the mill region name, and appropriated on no short of what one web server. A webpage may be accessible by methods for an open Internet Protocol (IP) compose, for instance, the Internet, or a private neighborhood (LAN), by referencing a uniform resource locator (URL) that recognizes the site. All straightforwardly accessible destinations everything considered constitute the World Wide Web, while private locales are regularly a bit of an intranet.

Pages, which are the building squares of destinations, are reports, generally made in plain substance sprinkled with outlining rules of Hypertext Markup Language (HTML, XHTML). They may combine segments from various locales with fitting markup catches. Site pages are gotten to and transported with the Hypertext Transfer Protocol (HTTP), which may then again use encryption (HTTP Secure, HTTPS) to give security and assurance to the customer. The customer's application, much of the time a web program, renders the page content as demonstrated by its HTML markup rules onto a show terminal.

Application outline

For take after the association's essential. The association has many stores wherever all through the country. They give gloriousness things. Since they have grouping of things and amounts of stores. They might need to unite their store structures remembering the true objective to have the ability to record purchases, make reports for the things sold each day/week after week (or any foreordained date expand). There are two essential sorts of customers. Arrangements staff: they can record demands and enroll portions.

National executive: They may be arranged at any store and they may have the ability to make gives a record of step by step/step by step (or from any predefined date run) as for trades held and the things sold, for a specific staff or for all staff, for a specific store, or for all stores. We was influenced a business to web by fitting system and response's association as need, direct structure. By then we do the guide document empower customer to can use fruitful.

In this site have two essential sort is: Sales staff, National overseer

In sort staff have a couple of limits:

- Receive email demand's customer and certify email.

- Check things in data

- Order and check portion for customer

In sort National Manager have a couple of limits:

- Manage account staff.

- Check organize each staff.

- Report reliably, reliably to check kind things.

Possible improvements

Some features and detail have to improve:

Improve interface more friendly.

* Advanced search engine
* Fast download speeds
* Compatible with difference modern.
* High security.
* Add more validation.
* **Testing plan used**

## Questionnaire and Peer feedback evaluation

|  |  |  |
| --- | --- | --- |
| Question | Answer | Feedback evaluation |
| Does my application make you happy about UX / UI, speed or not? | On the speed side I feel the web is running fast without loading the page. As for the UI, you need to improve the product list and the order page as it makes the user feel friendly. | The React Library as well as the web api provide developers with a fast web development tool as well as improved interface that is the core of web development. |
| What do you see about my website function? | The functionality of the web is relatively adequate to meet the needs of today's users. But need to upgrade some more features posted available as ordered with product prices and shipping prices. | In summary, it is necessary to change some functions in the shopping cart and in the database there must be a database of shipping price management to serve the customers want to deliver home. |
| About the look and feel of the admin section, how do you feel about my app? | I feel you have met the requirements of the user when listing the details of the whole table as well as allowing them to manipulate the database through the web application. Also, I think you should offer the super administrator privileges to the administrator. | Here, in addition to providing a large distribution system in the system, we must replace the function in the user's orders. We only allow administrators to change the order's status and prevent them from deleting orders. |
| What do you think about my website which is divided into 2 part front-end and back-end? | I feel that you are splitting it out because it is easier to manage. | In this section, the need to improve is to learn the latest knowledge and update information from the reactor to upgrade the reactor application. |

Future Work: The application can develop several other functions such as intelligent search with the product that users want to find. Next, is the use of nodejs support for real-time processing and application for online customer support. In addition, customers do not need to log in to pay for the shopping cart instead of providing personal information to keep the system for potential customers.

Interface needs to change constantly to meet the expectations of customers and users. Improved UX / UI is also considered as an important step to reach and attract customers.

Develop a system for capturing customer feedback on web services as well as improvements to improve service quality.

\_Conclusion:

In conclusion, the website has provided the necessary functions to help customers make their requests easily. But through that the website also needs to be upgraded to provide more protection in the near future.

# Task 2:

\_The first problem is finding the api call to get data about in microsoft web api. The default design of microsoft's api is to return the html data to the user to retrieve the data. But it drained the computer.

Improvement: Method of improvement: We have to customize the system to return the data of the web api from xml to json. Since the format of JSON is literally (lightweight), it is easy to send as the data format used by Any programming language.

\_The second problem is finding the library to write the front end. Instead of using the angular, I used the reactor for web application development. Because Reactjs is extremely efficient: Reactjs creates for itself the virtual DOM - where the components actually exist on it. This will greatly improve performance. Reactjs also calculates which changes need to update the DOM Len and only execute them. This helps Reactjs avoid the need to manipulate the DOM at multiple costs. We can write a simple example of ReactJS as follows.

\_The authenticate function retrieved data from the api encountered a serious error and I detected the patch and corrected it. To be precise, it is a must to completely change the authenticate structure of the present. Because in the web's controller, the return data is any data type except for the data by object. Returning customer information to a data type is an object that causes the application to crash in the authenticate section.

\_In future to improve the quality of web applications written by reacting or deploying the application to the server we have to maximize the amount of code that we have used. To be more precise, we have to pack the code into a file and just a pure html page can deploy the application to the host.

\_And I was thinking of using web pack to improve my project: what is the title of the web pack, in short, Web pack is a tool to encapsulate all js, css files (including scss, sass,) . The encapsulation is not all messy, but it is wrapped around the project structure, from one module to the other. The web pack also has a lot of useful functions, such as optimize or options to run on different environments (dev or production), watch file, ... I've been browsing the web pack using a plugin that compresses a js file from 6mb to 76 kb. With the contribution of a huge dev community, web pack is a great tool for dev js, and using web packs for large projects is great.

Having said that, web pack cannot fail to mention code splitting, this is my favorite feature in the web pack, I will give a simple this, in a project, the import of many libraries at the same time. It is very normal if you have an A.js import C library file and there is also a B.js file importing C library, so when the web pack builds, it will be a file that contains code 2 times the C library. , so this can reduce performance  
There are 3 approaches to split current code:  
 \_Entry point: Manually divide by configuring file entries, files to start the application, and from which the web pack operates  
 \_Prevent duplication: Use split Chunks to remove duplicate bundles and split them into chunks, simply understanding the web pack will automatically find the import libraries for different files, when your logic needs it. Separate files to use  
 \_Dynamic Imports: Divide code through functions called in modules.

\_Babel is a JavaScript or JavaScript transpiler code conversion engine, used for the purpose of converting JavaScript code written using the new ECMAScript standard to older versions of earlier versions.  
  
\_Reading the above paragraph will no less you will ask yourself what ECMAScript is, is not the JavaScript language already have rules for programming already?

References:

<https://redux.js.org/>

<https://reactjs.org/>

<https://github.com/axios/axios>

<https://getbootstrap.com/>

<https://cdnjs.com/libraries/font-awesome>

<https://github.com/jerairrest/react-chartjs-2>

<https://viblo.asia/>

<https://stackoverflow.com/>

<https://www.youtube.com/watch?v=99vHH_6F0Ko&t=2823s>

https://blog.thanhtung.website/react-native-build-deploy-faster/?fbclid=IwAR38ZBNgWUad3hVzWRrjySeetkKxEVuIJi2FECwDp4ZPPc47FhwhgUHIlv0