**ABSTRACT**

The concept of shopping in malls is undergoing a dramatic change. A few years back, people just used to visit a lot of stores for their needs, but off late every shopping mall seems to be overflowing with people of different ages from children to adults. With globalization at its peak and paving a way for the various brands, people now feel that they can get anything that they want in these shopping malls be it electronic goods, apparels, consumer durables, groceries and vegetables.

For the past few years researchers have been working to establish a relationship between people shopping in malls and there shopping behaviors. The reason for conducting such a research is to find out the customer convents way of shopping that they can do a lot of shopping in the malls. But yet, not many people will be comfortable in shopping with traditional methods. In the mall there are commonly two types of items that people will see. The first type consists of items that need not to be chosen from so many choices but are fixed and basic stuff. The other type is where people wish to give more time while shopping the items of their choice. Traditionally people have to choose their self and need to take care of all the things like size, design, color, etc. The best example for the second type is clothes. It is clear that one always spends more time in this type of shopping. Our Web portal helps the user by managing the fixed items section by just collecting the list of items needed and then it passes the data to the supporting staff members. Thus while the customer is busy in the clothing or other such section the supporting staff will collect all the required items from the ordered list and will keep it ready for billing at the counter.

This software system is an online Shopping Mall BOT system known as “Shoppy” BOT that helps in managing various shopping operations effectively. The details include finding an item, suggesting similar items, finding an existing item using image processing, showing current offers, adding them to bill, etc. The project is an online web portal that is operated with voice commands and helps the customer to enhance the shopping experience.

**TABLE OF CONTENTS:**

Abstract 1

Table of Contents 2

List of Figures 4

List of Tables 4

|  |  |  |
| --- | --- | --- |
| 1 | **Introduction**  1.1 Introduction  1.2 Scope  1.3 Project summary and Purpose  1.4 Overview of the project | 5  5  5  6 |
| 2 | **Technology and Literature Review**  2.1 About Tools and Technology  2.2 Brief History of Work Done | 7  8 |
| 3 | **System Requirements Study**  3.1 User Characteristics  3.2 Hardware and Software Requirements  3.3 Constraints  3.3.1 Hardware Limitations  3.3.2 Interfaces to Other Applications  3.3.3 Parallel Operations  3.3.4 Reliability Requirements  3.3.5 Criticality of the Application  3.3.6 Safety and Security Consideration  3.4 Assumptions and Dependencies | 13  13  13  13  14  14  14  14  14  14 |
| 4 | **System Analysis**   * 1. Study of Current System   2. Requirements of New System   4.2.1 User Requirements  4.2.2 System Requirements  4.3 Features of New System  4.4 Class Diagram   * 1. System Activity-Use case diagram   4.6 Sequence Diagram | 15  15  15  15  16  17  18  19 |
| 5 | **System Design**   * 1. System Application Design   5.1.1 Method Pseudo code   * 1. Database Design   2. ER Diagram   5.4 Activity Diagram  a.) For Customer   1. For Shoppy | 20  20  21  24  25  25  26 |
| 6 | **System Testing**  6.1 Test cases | 27 |
| 7 | **Conclusion** | 29 |
| 8 | **Bibliography** | 30 |

**List of Figures**

**Figure NO.** **Name** **Page No**.

4.4 Class Diagram 25

4.5 Use case Diagram 26

4.6 Sequence Diagram 27

5.3 Entity Relationship Diagram 31

5.4(a) Activity Diagram 32

5.4(b) Activity Diagram 33

**List of Tables**

**Table NO.** **Name** **Page No**.

**5.1** Items 21

**5.2**  Orders 22

**5.3** Offers 22

**5.4** Staff 23

**1 INTRODUCTION**

**1.1 INTRODUCTION**

The Shopping Mall BOT is a voice command driven system which will help the customer in shopping. The system takes care of basic operations that will ease the shopping experience of a customer. It can give you direction of an item, suggest you similar items, add items to your bill, etc. From now on the customer will not have to carry a shopping cart full of items and roaming around the mall, instead he/she only has to go to the section where they have to choose the things by themselves and for items that are fixed, just add them to bill via BOT and you will get them ready on the counter.

**1.2 SCOPE**

**1.2.1 CURRENT SCOPE**

With this BOT system the customer can find items, add them into list, change quantity of item, remove an item from list, cancel the list through voice command. They can also get information on current offers through our system.

**1.2.2 FUTURE SCOPE**

* Multiple language support
* Multiple browser support
* Give suggestion through image input
* Suggestion on frequently bought together items

**1.3 PROJECT SUMMARY AND PURPOSE**

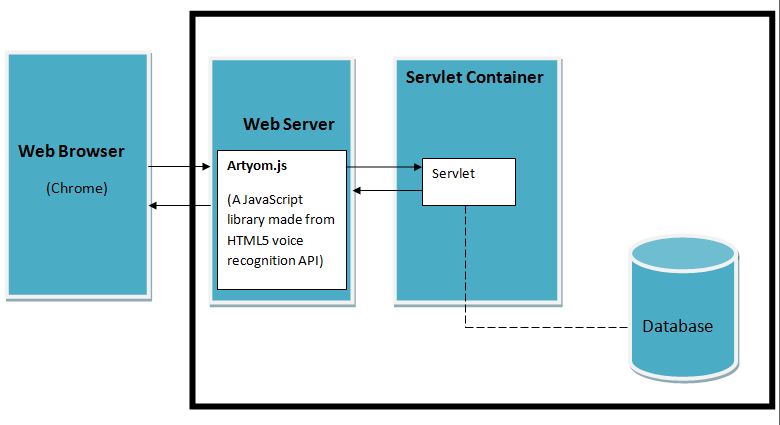
**1.3.1 PROJECT SUMMARY**

* Customer can find items
* Instead of searching them get them ready to collect at counter
* Add items to list
* Calculate total amount
* Can handle other shopping related operations, for example adding, removing, changing, etc.

**1.3.2 PURPOSE**

The main purpose of shopping mall BOT is to enhance the shopping experience. The BOT will take care of your shopping list and will guide you in shopping. The main advantage is, it is a voice command driven system so the customer just have to use the given commands and you can see the results in your screen. You just have to make a list and the BOT will send your order to the staff member, so you can spend more time on clothing section or in your favourite sections while the order is being ready on counter.

**1.4 Overview of the project**



System architecture is the conceptual model that defines the structure, behaviour, and views of a system. It comprises of system components, the externally visible properties of those components, and the relationships between them. It can provide a plan from which products can be developed, that will work together to implement the overall system.

The architecture is based on many components which are mentioned in the architecture diagram as follows:

1. **Browser**

This component of the system is used is by the client to access the web portal. It uses *HTTP* request/response architecture to send and receive data.

1. **Web Server**

This component contains all JSP, HTML, JS, CSS, etc files which will be invoked by the client as a *HTTP* request to server.

1. **Servlet Container**

This component **contains** all class files and interfaces, i.e. known business logic of the application.

1. **Database**

This manages the application’s database by creating, updating and modifying the database whenever the user changes the task list information in Show-Task view. Whenever the user uses the application for the first time, the database is created, and further-on when the user enters new task information, the database is updated by checking the integrity of the information before storing the data.

**2 TECHNOLOGIES AND LITERATURE REVIEW**

**2.1 ABOUT TOOLS AND TECHNOLOGY**

**2.1.1 Web Speech API**

This specification defines a JavaScript API to enable web developers to incorporate speech recognition and synthesis into their web pages. It enables developers to use scripting to generate text-to-speech output and to use speech recognition as an input for forms, continuous dictation and control. The JavaScript API allows web pages to control activation and timing and to handle results and alternatives.

**Features:**

* Speech-input and text-to-speech output.
* Can support both server-based and client-based/embedded recognition and synthesis.
* Speech recognition results are provided to the web page as a list of hypotheses.

**2.1.2 Java Technology**

Java has strong support for web development. While Java on the desktop, with the notable exception of Eclipse RCP based application was never a huge success, Java is frequently used at the server side.

A Java web application is a collection of dynamic resources (such as Servlets, Java Server Pages, Java classes and jars) and static resources (HTML pages and pictures). A Java web application can be deployed as a WAR (Web Archive) file. A WAR file is a zip file which contains the complete content of the corresponding web application.

**Features:**

* Portable.
* Platform independent.
* Secured.
* Robust.
* Architecture neutral.
* Dynamic.

**2.1.3 AJAX**

AJAX stands for **A**synchronous **J**avaScript **and** **X**ML. In a nutshell, it is the use of the XMLHttpRequest object to communicate with servers. It can send and receive information in various formats, including JSON, XML, HTML, and text files. AJAX’s most appealing characteristic is its "asynchronous" nature, which means it can communicate with the server, exchange data, and update the page without having to refresh the page.

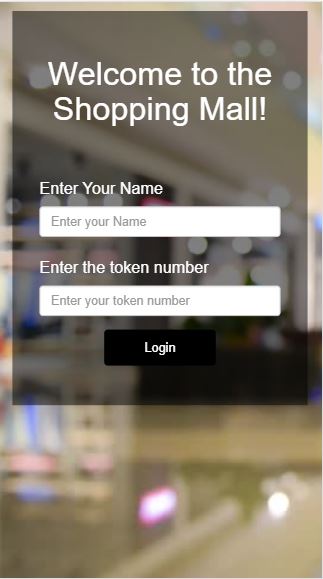
The two major features of AJAX allow you to do the following:

**Features:**

* Make requests to the server without reloading the page
* Receive and work with data from the server

**2.2 BRIEF HISTORY OF WORK DONE**

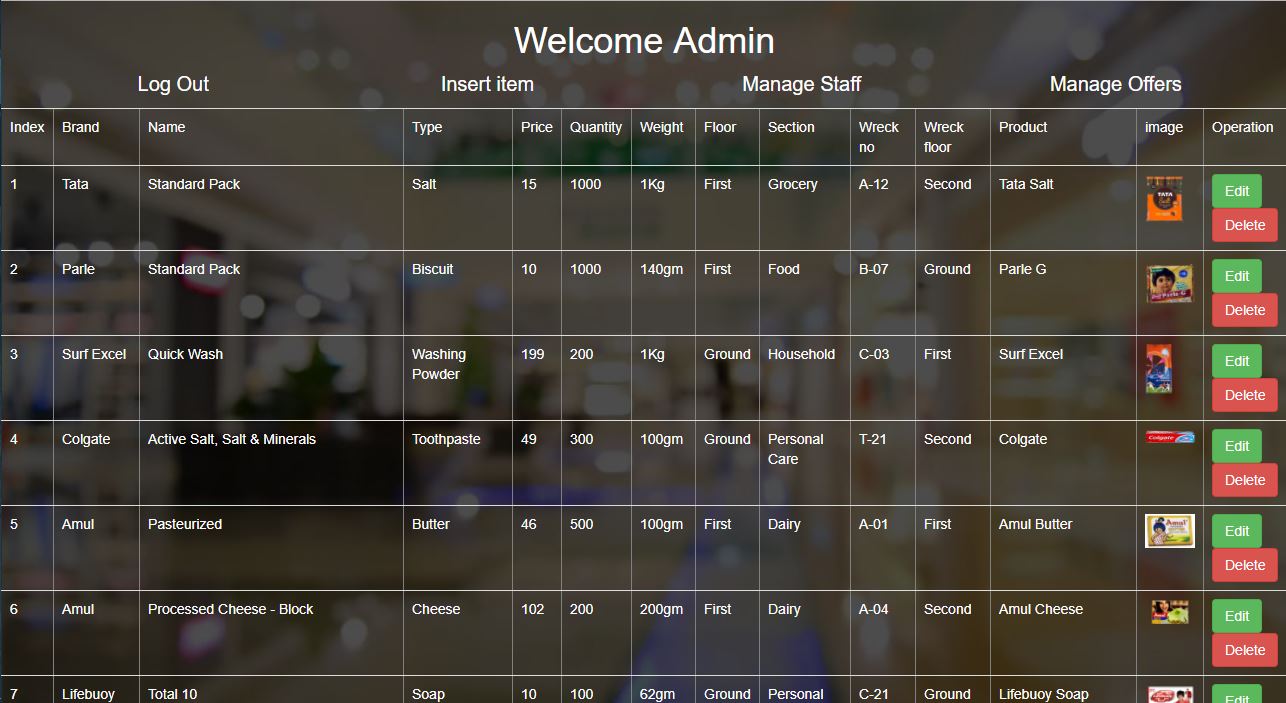
**Login Page (Login.jsp)**

****

The login page is common for all of our users, admin and staff. The admin will enter the unique password and username that is unique and confidential, staff members will enter their name and their unique id as token. Customers have to add the token number which will be given at the entrance of mall.

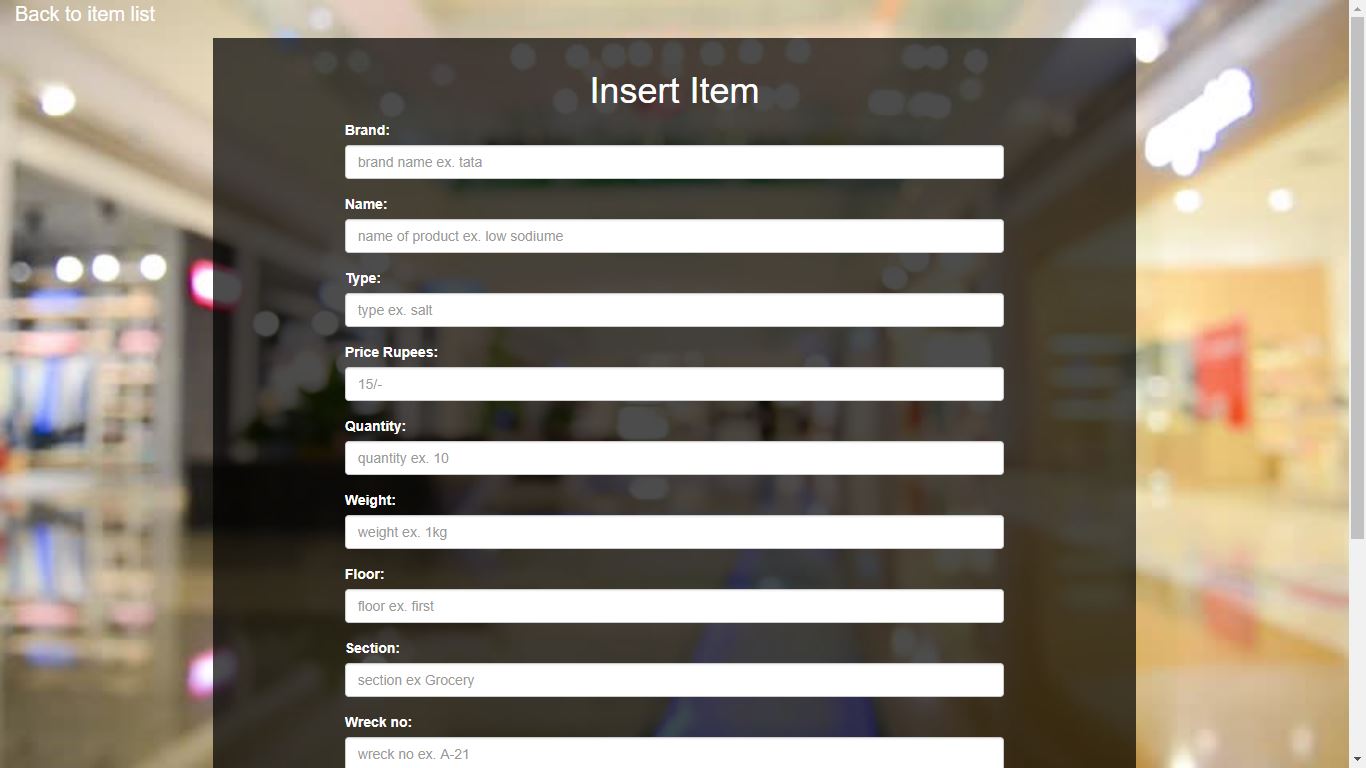
**Admin Side Pages**

**Admin.jsp**

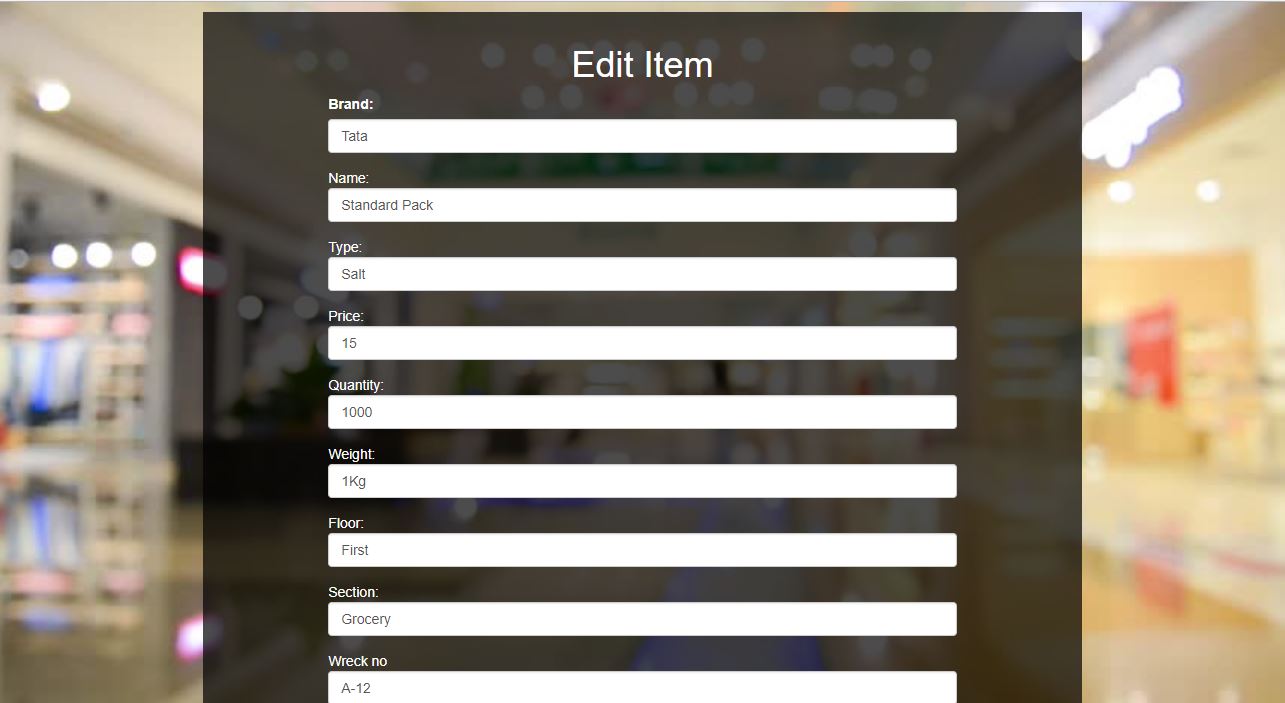
****

Admin page manages tasks like adding offer, managing items and staff operations.

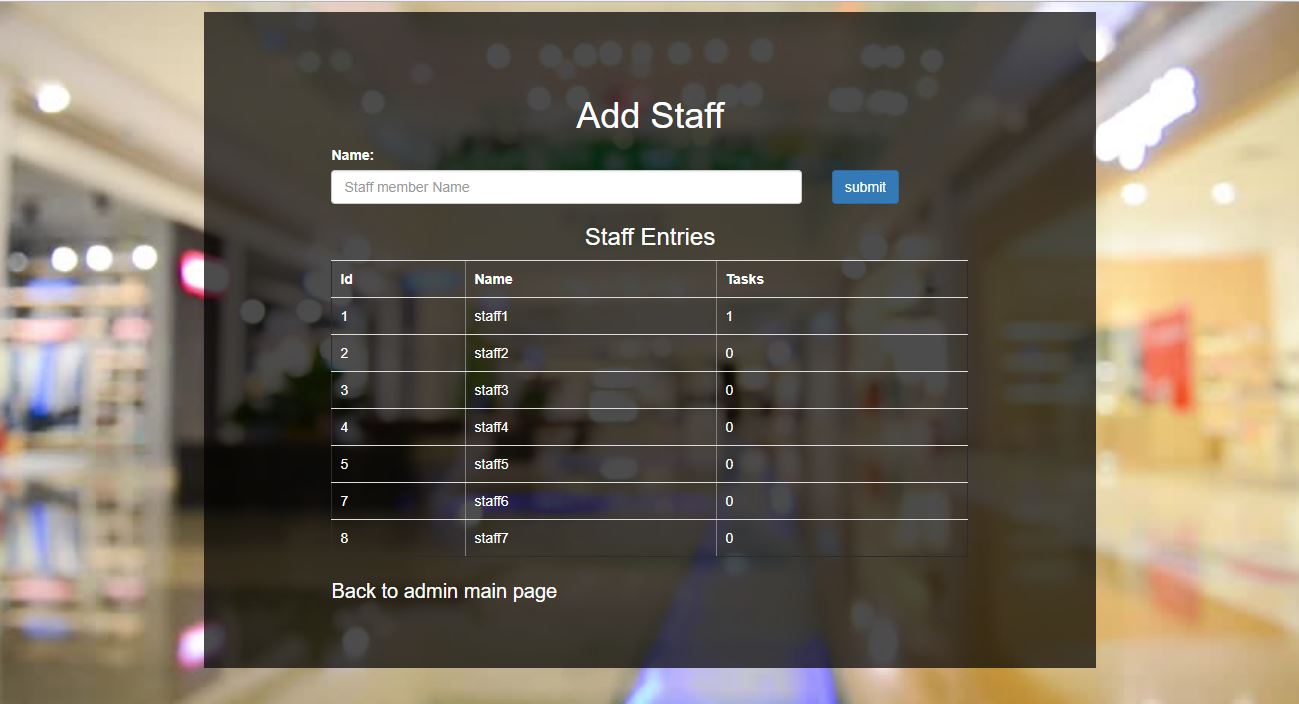
**Insert.jsp (To insert an item)**

****

**Edit.jsp(To edit)**

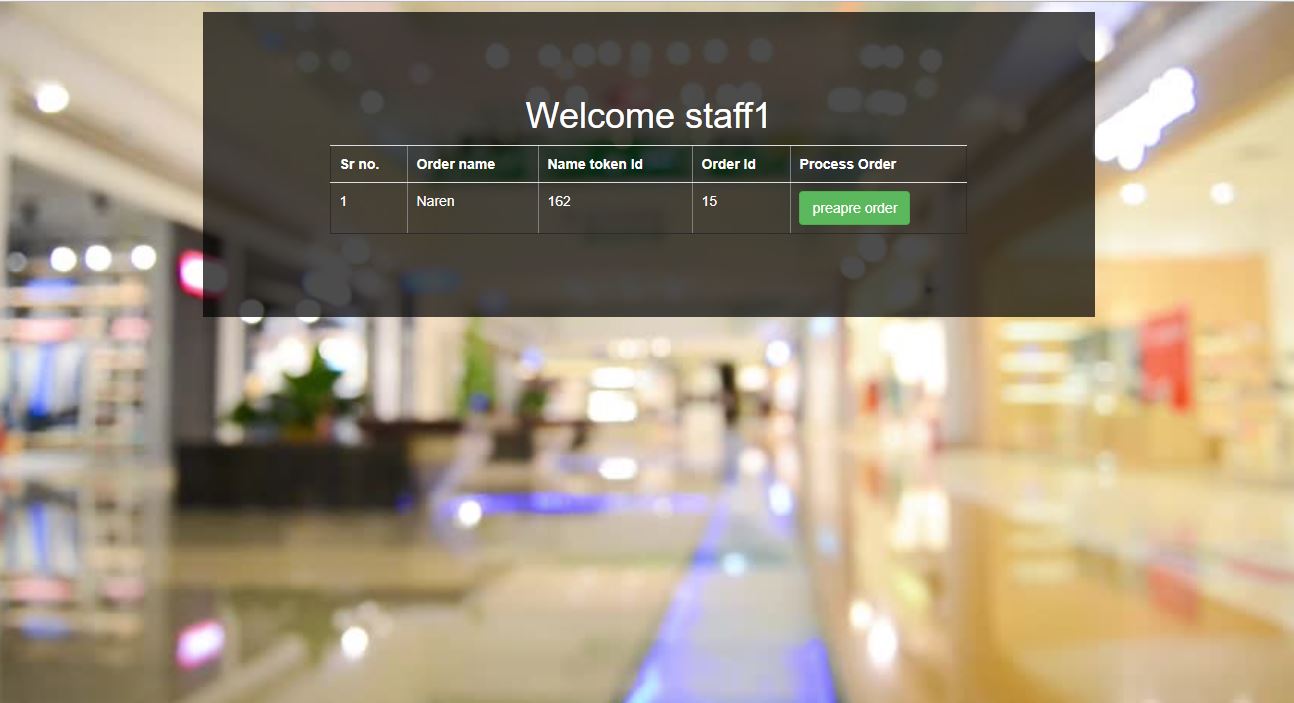
****

**ManageStaff.jsp (To see staff activity and staff members)**

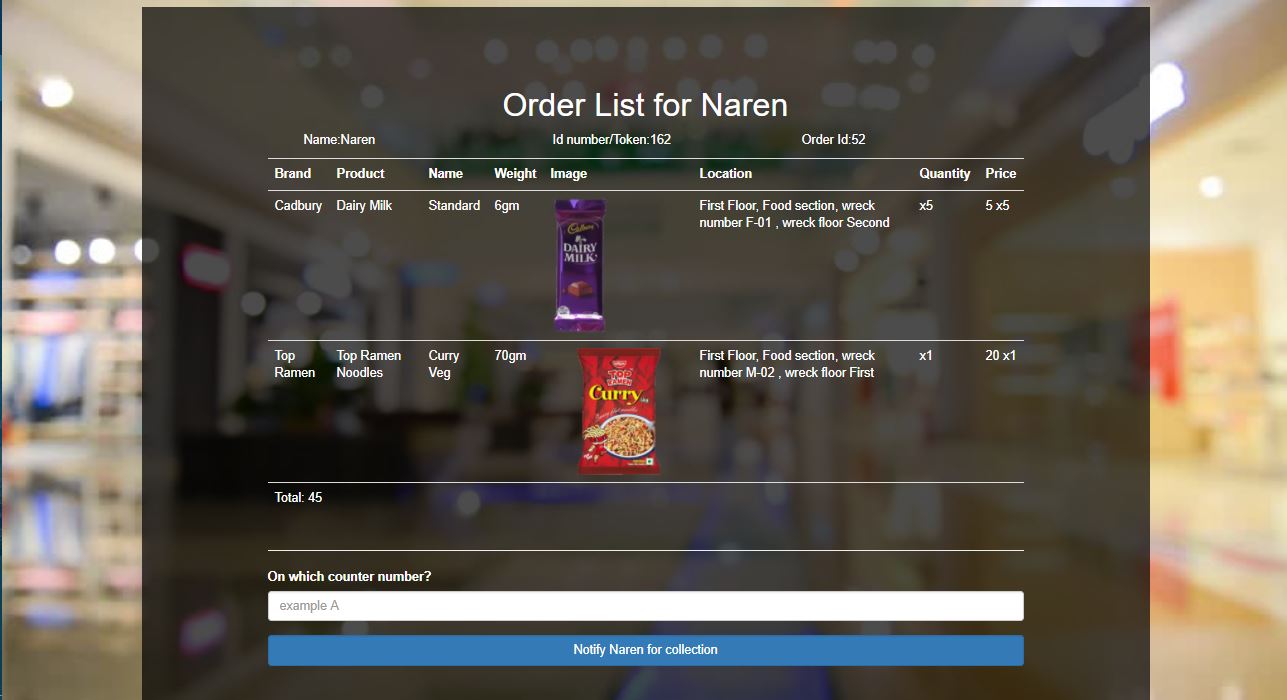
****

**Staff Side Pages**

**Staff.jsp (To manage orders)**

****

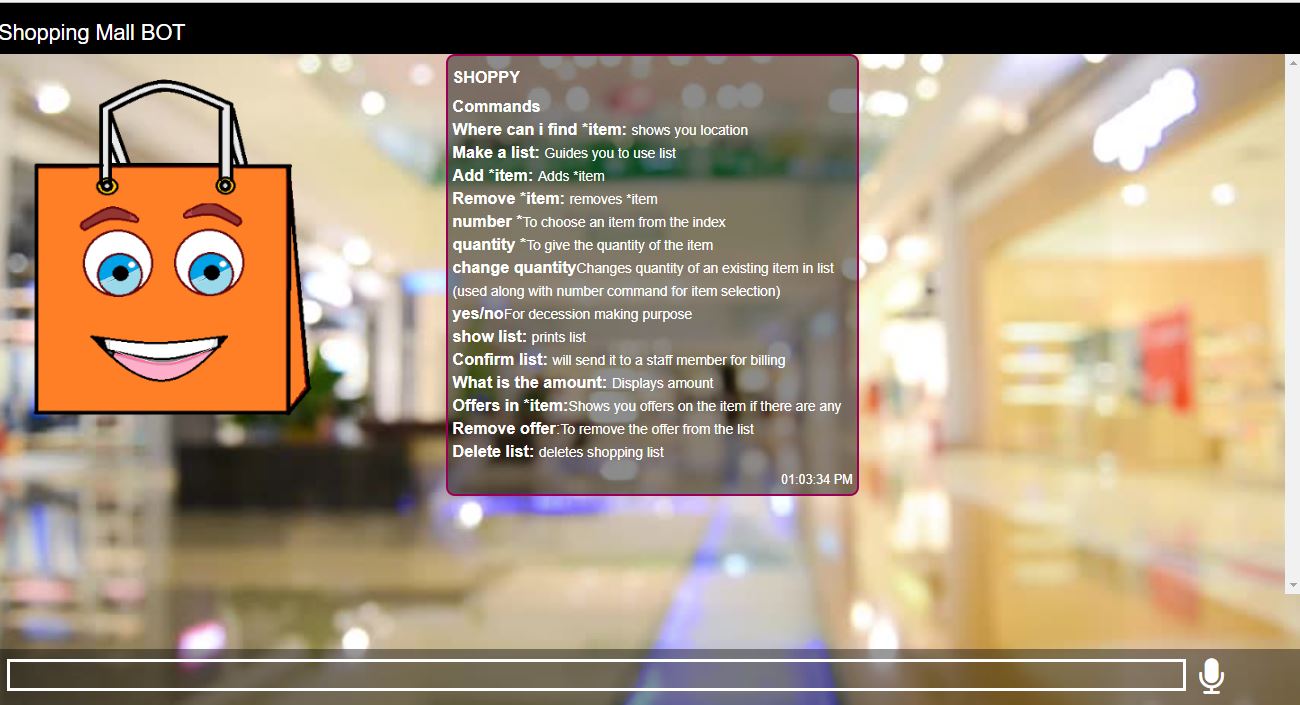
**CollectOrder.jsp (To prepare order and get it ready for customers)**

****

**Main page (mainPage.jsp)**

This is the main page and the only page the user/customer is going to interact with the system. The Shoppy BOT will help you in shopping and will manage your shopping list and will help you to increase your shopping experience. Below the bag is list of commands are given. Through which customer can interact with Shoppy. The commands are shown in seconds figure. The history of queries and results is saved as a chat format between the user and Shoppy.

We have added list of voice commands. The customer only has to touch the microphone icon and a sound will be generated which indicates that now system is ready to listen. The commands and their meanings are as follows:



|  |  |
| --- | --- |
| **Command** | **Result** |
| Where can I find \*item | Shows you location of the \*item |
| Make a list | Starts storing items in a list |
| Add \*item | Adds \*item to the list |
| Remove item | Removes \*item from the list |
| Show list | It displays/prints the list |
| Confirm List | The list is sent to the staff member for collecting the specified items and keeping it ready on the counter |
| What is the amount | Informs the total bill amount |
| Delete list | Deletes list |
| Offers in \*item | Shows you available offers in a particular item |
| Yes/no | Used in decision making |
| Quantity \* | To specify the quantity of item while adding |
| Change quantity | This is used along with number command |
| Number \* | To select an item from index number |
| Log Out | To exit the session |

**3. System Requirements Study**

**3.1. User Characteristics**

We have identified four potential classifications of users of our system:

* Customer:  More often than not, software is designed for a customer. The customer is the end user of the system.
* Search BOT:  The BOT helps the customer in shopping by doing some basic operations like searching an item, giving suggestion for similar items, showing current offers, etc.
* Staff:  These users will receive the order from the user through the system and will add those things to their billing list and will keep those things ready on the counter.
* System Administrator: Generally they are less in number. This user is the most important for the system. The admin is responsible for security and maintenance of the system.

**3.2 Hardware and Software Requirements**

* Hardware requirements:
  + Smart phone running Android/ IOS / Windows/ etc
  + Supporting voice recognition
  + Able to connect to Internet
* Software requirements:
  + Voice recognition enabled
  + Google Chrome Browser installed
  + Java Supportive
  + JavaScript Enabled
  1. **CONSTRAINTS**

**3.3.1 HARDWARE LIMITATIONS**

Most devices offer relatively less accurate voice recognition services. In those cases our Shoppy BOT cannot give the customer intended result. The most important requirement is good internet connection, without that user cannot use the system since it is a web portal. The system only works in devices supporting java technology.

**3.3.2 INTERFACES TO OTHER APPLICATIONS**

Application uses external JavaScript library that is made from Speech recognition API and Speech synthesis API service. The API will convert the voice to speech in a form of JS object which contains identified speech and its hypothesis.

.

**3.3.3 RELIABILITY REQUIREMENTS**

Reliability requirements of the system are one of the prime ones in the list. The system is needed to be highly reliable in terms of performance and capable of delivering robust performance. If the Voice recognition works at all times and accurately, the application is said to be reliable.

**3.3.4 PERFORMANCE REQUIREMENTS**

Application should help the customer to store items in list, find location of item accurately, suggesting similar items and offers. Adding items in bill and storing items in list should be done accurately.

**3.3.5 CRITICALITY OF THE APPLICATION**

Application will need internet connectivity continuously because it is a web portal. Also customer’s device should support Voice input and permit as well. Google Chrome is also needed to be installed in the user’s device and the latest version is preferable.

**3.3.6 SAFETY AND SECURITY CONSIDERATIONS**

Application should not crash anywhere in between taking voice input. Input should be properly taken. Since it works using Google API this is secure.

**3.4 ASSUMPTIONS AND DEPENDENCIES**

**ASSUMPTIONS**

* User is the person having enough knowledge for their own device’s operation.
* Chrome is installed and voice input permission is given.
* The user has entered a valid token number that was given at the entrance.
* The device is a java supported.

**DEPENDENCIES**

* The Internet connectivity of the phone has to be on the whole time for using the application.
* The Google API/Artyom.js is working all the time and is accessible.

**4. SYSTEM ANALYSIS**

**4.1 STUDY OF CURRENT SYSTEM**

No similar system exists in present.

**4.2 REQUIREMENTS OF NEW SYSTEM**

**4.2.1 USER REQUIREMENTS**

The requirements of the users as have been identified are as follows:

* Users are required to use the OTP/Token, with the help of which they will be able to enable “Shoppy- The Shopping Mall BOT”.
* Users are required to enter the product details through voice input, to find the product using BOT.
  + 1. **SYSTEM REQUIREMENTS**

**Mobile Requirements and Permissions**

* User’s mobile must support ‘Google ***Voice Recognition Service***’ and give permission to application to use it.
* User’s mobile must support ‘Google ***Voice Synthesis Service***’ and give permission to application to use it.
* User’s mobile must contain “***Google Chrome***” Web Browser.

**Used API/ External Library:**

* **Artyom.js**

**ABOUT API:**

Artyom is a Robust Wrapper of the Google Chrome Speech Synthesis and Speech Recognition that allows you to create a virtual assistant. Create awesome stuff with Artyom; build your own Siri, Google Now or Cortana within your web application.

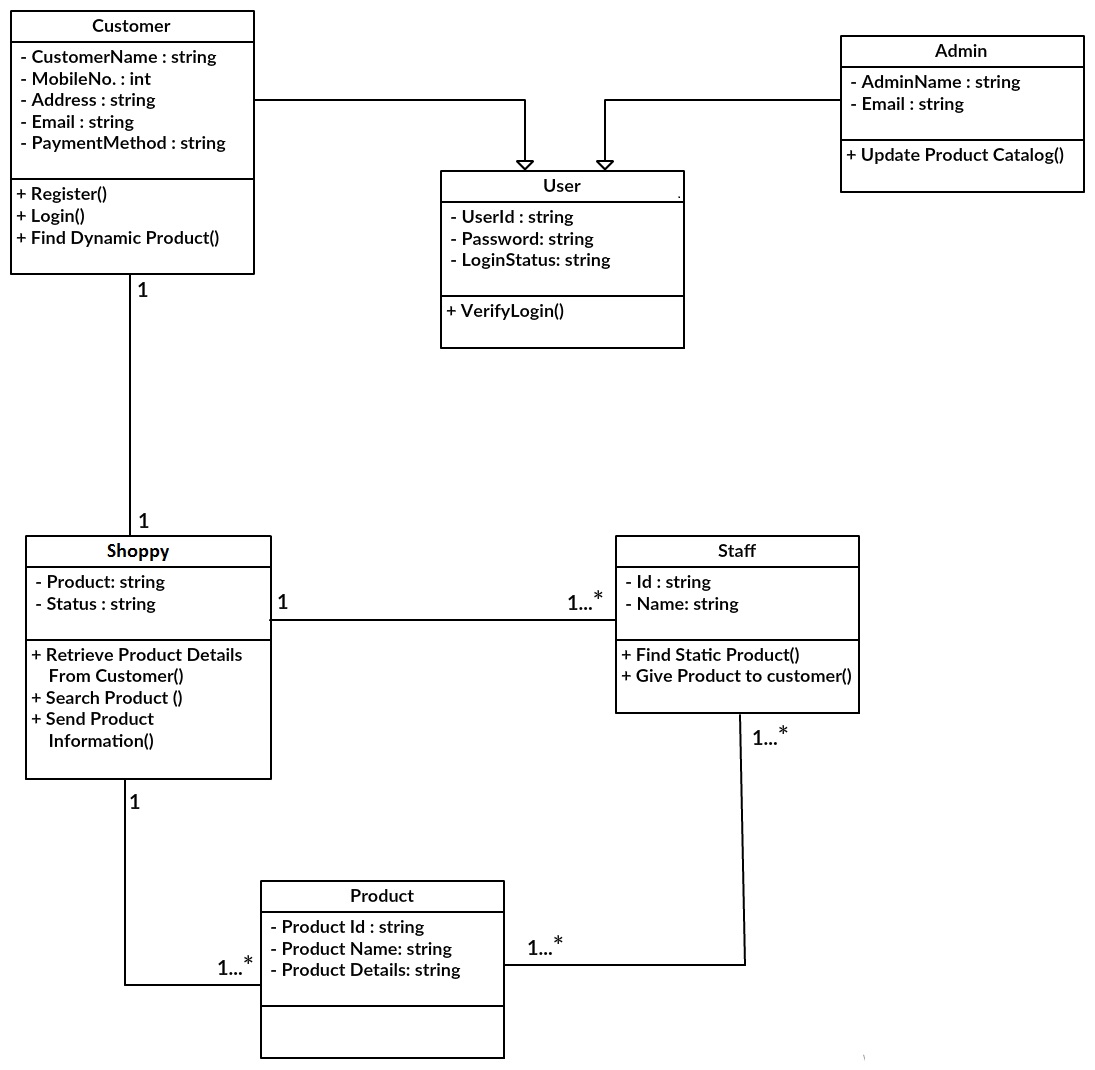
The official APIS Speech Synthesis and Speech Recognition are actually supported only for Google Chrome (which uses ***Google Recognition and Synthesis services***). Artyom works in Android Chrome Too.

**4.3 FEATURES OF NEW SYSTEM**

The new proposed BOT system of helping the users to find some product in the shopping mall, gives the following features:

* After using the BOT for searching, the user will be notified about the product information (including the product location within the mall).
* The BOT will also notify the staff members in case about the product information, in case the customer searched for a static product. To help the customers by getting product and keeping them ready on the billing counter.
* The BOT will also give information about Special offers and Discounts going on certain products.

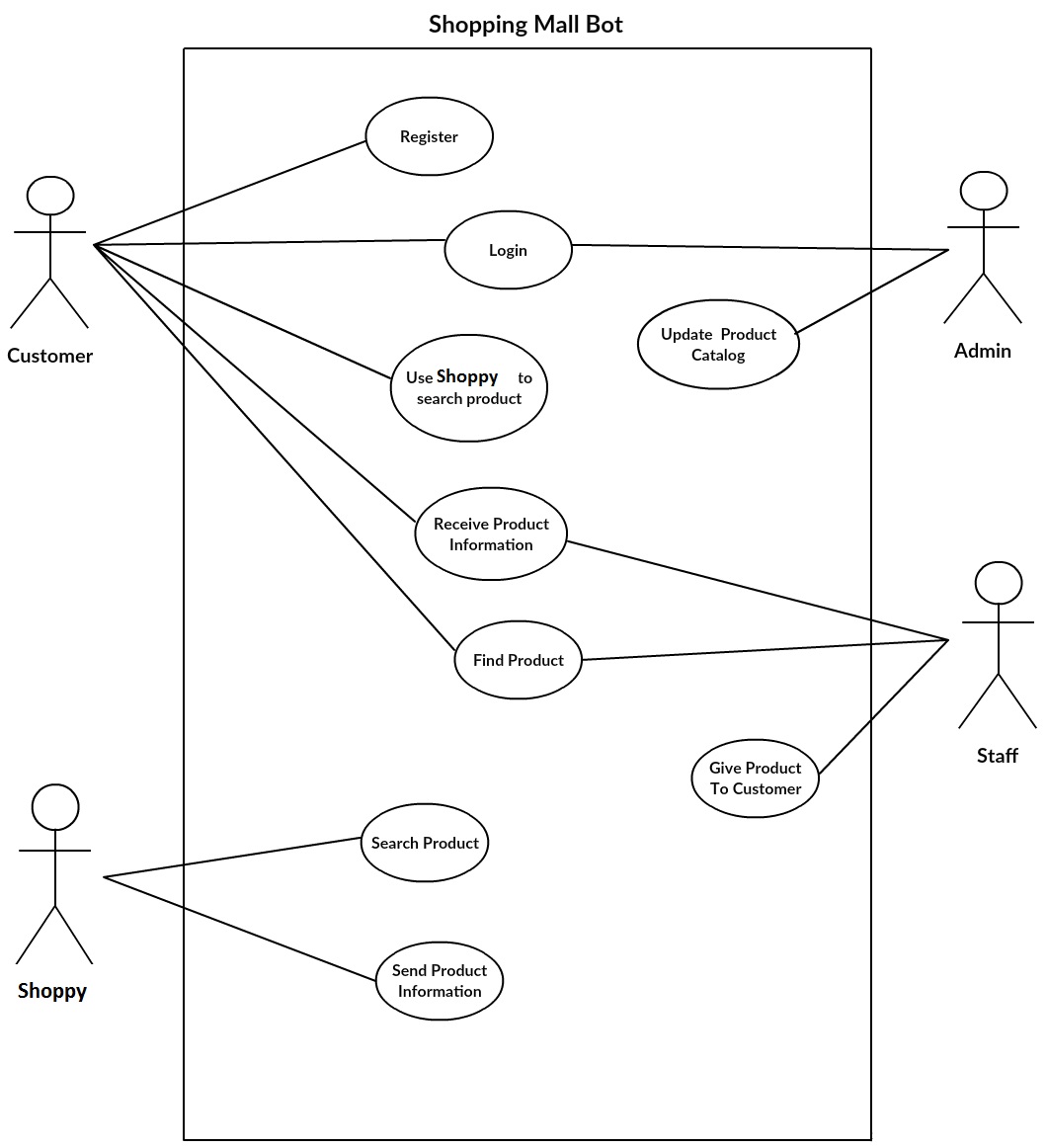
**4.4 CLASS DIAGRAM**

****

**Figure 4.4 Class Diagram**

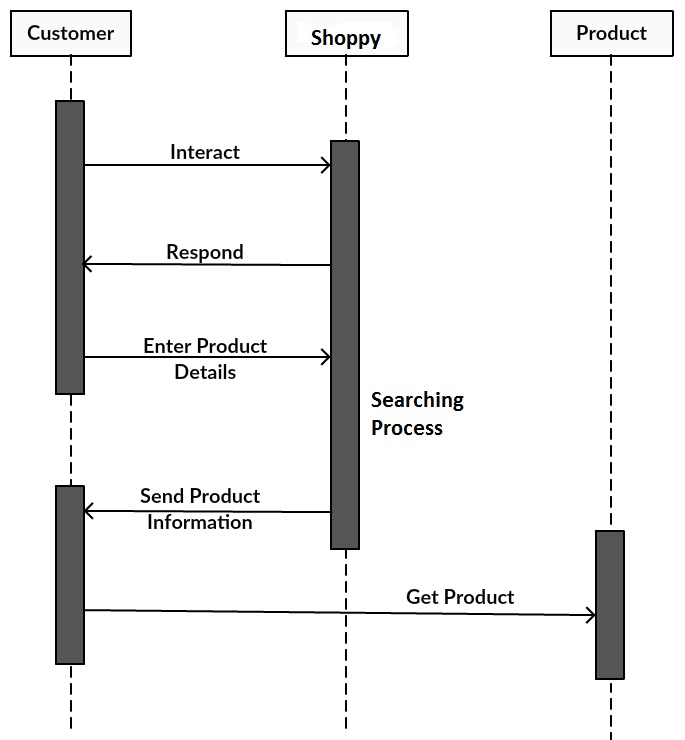
**4.5 SYSTEM ACTIVITY**

**USE CASE DIAGRAM**

****

**Figure 4.5 Use Case Diagram**

**4.6 SEQUENCE DIAGRAM**

****

**Figure 4.6. Sequence Diagram**

**5. SYSTEM DESIGN**

**5.1 SYSTEM APPLICATION DESIGN**

There are two types of product in mall **Static and Dynamic Products**. Our focus is on the first category.

**Static** products are basically a group of products which are used in daily life such as food, bakery, dairy, stationary, general life use products, etc. which does not require customer choice to choose the product.

**Dynamic** products are basically a group of products which are used in daily life for style, comfort, etc. such as cloths, etc. require customer choice to choose the product.

**5.1.1 METHOD PSEUDO CODE**

* **“Enter Product Details” Activity**

This activity allows the user to tell about the type of product to the “Shoppy-The Shopping Mall BOT”, which is to be searched. The details include like product name, product brand, etc. Once the details are entered with voice input, it is checked whether the product exist in shopping mall or not.

**ALGORITHM 1**

Algorithm: **find Product item.**

//Purpose: To check what product is needed by customer and check whether that product exist within shopping mall or not.

//Input: product name, product brand, etc.

//Output: If successful product found, go to “Send product information” activity.

STEP 1: Let the user enter the product details via voice input

STEP 2: Retrieve the details, which are entered by user.

STEP 3: Verify and Check whether the given product according to details, is present in the shopping mall.

if (found ())

Then go to “Send product information” activity.

else

Generate “Product not found” Message.

**2. “Send Product Information” Activity**

This activity allows the “Shoppy-The Shopping Mall BOT” to tell about the information of the product(such as product location details, product cost ,etc.) to the users which have entered product details , so that they can find product easily and add to shopping list.

In case of static product, staffs are also notified about the product information which the customer wants to purchase.

**ALGORITHM 2**

Algorithm: **Send Product Information.**

//Purpose: To send information about product to the customer, so that they can find product.

//Input: Item name or brand.

//Output: Send product information to customer and staff.

STEP 1: Retrieve the information about the matched product from the product database.

STEP 2: Send the product information to the user.

STEP 3: Send product information to customer as well as staff.

**5.2 DATABASE DESIGN**

**DATA DICTIONARY**

**Table 1:** items (product information)

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Null** | **Description** |
| Id | INT(11) | Not null | Indicates id of product |
| Brand | Varchar(50) | Not null | Indicates brand name |
| Name | Varchar(100) | Not null | Indicates product name |
| Type | Varchar(50) | Not null | Indicates type of the item |
| Price | INT(11) | Not Null | Price of the item |
| Quantity | INT(11) | Not Null | Available quantity |
| Weight | Varchar(20) | Not null | Weight of the product |
| Floor | Varchar(50) | Not null | On which floor the item is |
| Section | Varchar(10) | Not null | Indicates section |
| Wreck\_no | Varchar(40) | Not null | Wreck number of the product |
| Wreck\_floor | Varchar(10) | Not null | Floor number in wreck of item |
| Product | Varchar(150) | Not null | Abstract name of product |
| Image | TEXT | Not null | Source string of image |

**Table 2:** Offers (Offers information)

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Null** | **Description** |
| id | INT(11) | Not null | Id of image |
| Products | Varchar(245) | Not null | Number of products specified in offer |
| Text | TEXT | Not null | Description of product |
| Ids | TEXT | Not null | Ids of products |
| Price | INT(11) | Not null | Total price of the offer |
| Max\_quantity\_allowed | INT(11) | Not null | Maximum quantity allowed per customer |

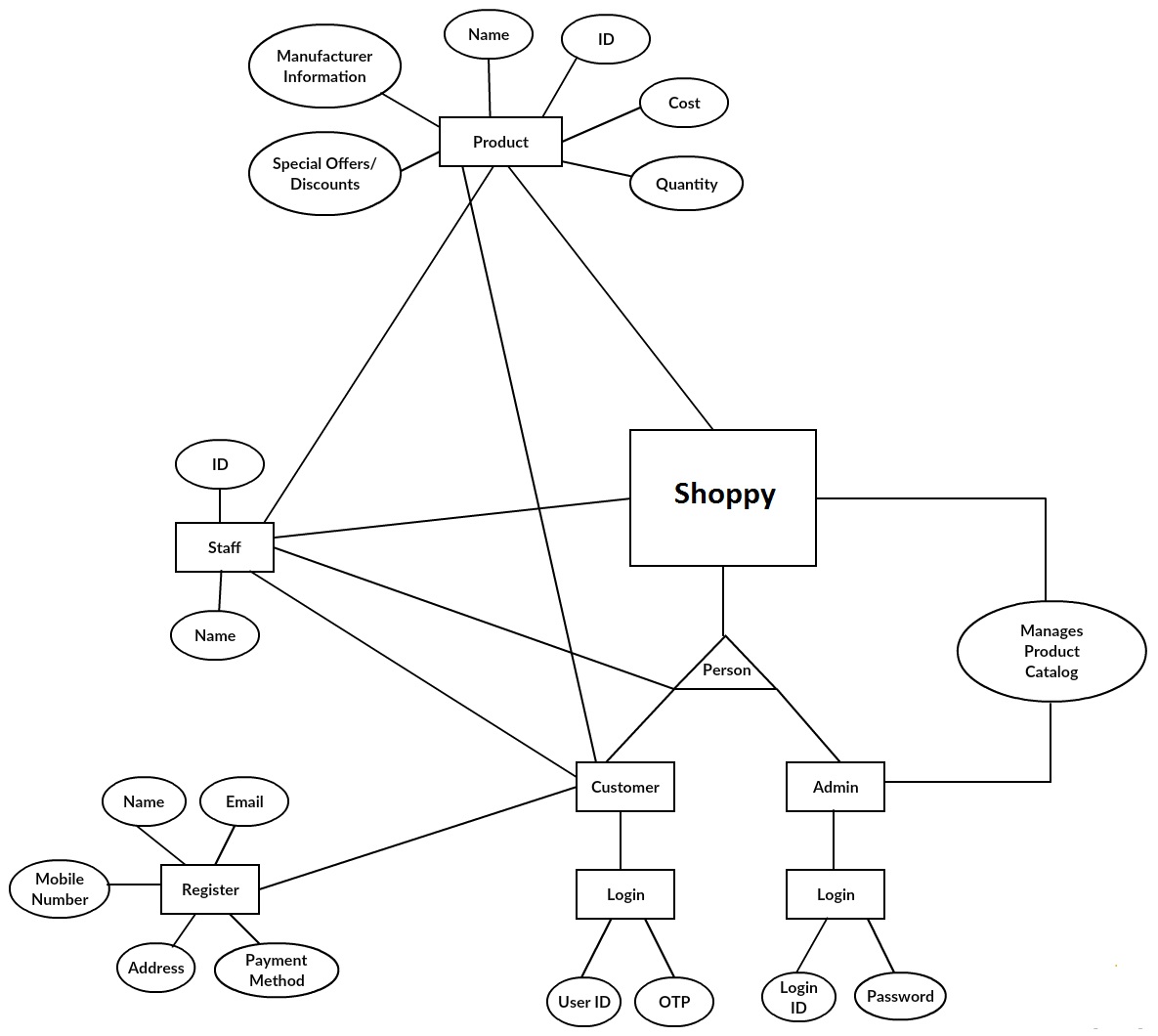
**Table 3:** Orders (Orders of the customer)

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Null** | **Description** |
| Id | INT(11) | Not null | Id of the customer |
| Staff\_id | INT(11) | Not null | Unique id of staff |
| Order\_name | Varchar(45) | Not null | Customer Name |
| Token\_id | TINYTEXT | Not null | Token id of customer |
| Items | TEXT | Not null | List of items separated |
| Message | TEXT | Not null | Description of the offer |
| Completed | TINYINT(4) | Not null | Boolean variable |

**Table 4:** Staff (Staff table)

|  |  |  |  |
| --- | --- | --- | --- |
| **Attributes** | **Data Types** | **Null** | **Description** |
| Id | INT(11) | Not null | Id of the customer |
| Name | Varchar(45) | Not null | Unique id of staff |
| Tasks | INT(11) | Not null | Number of tasks given to staff |

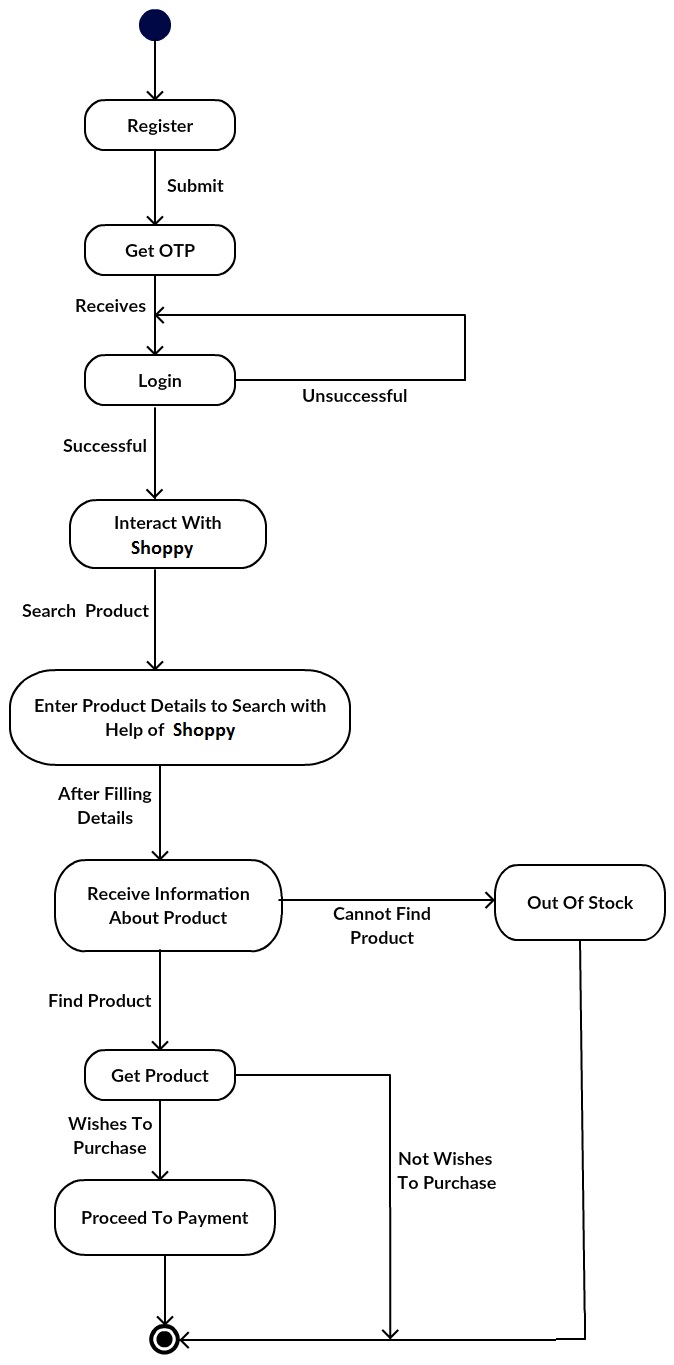
**5.3 ENTITY RELATIONSHIP DIAGRAM**

****

**Figure 5.3 ER Diagram**

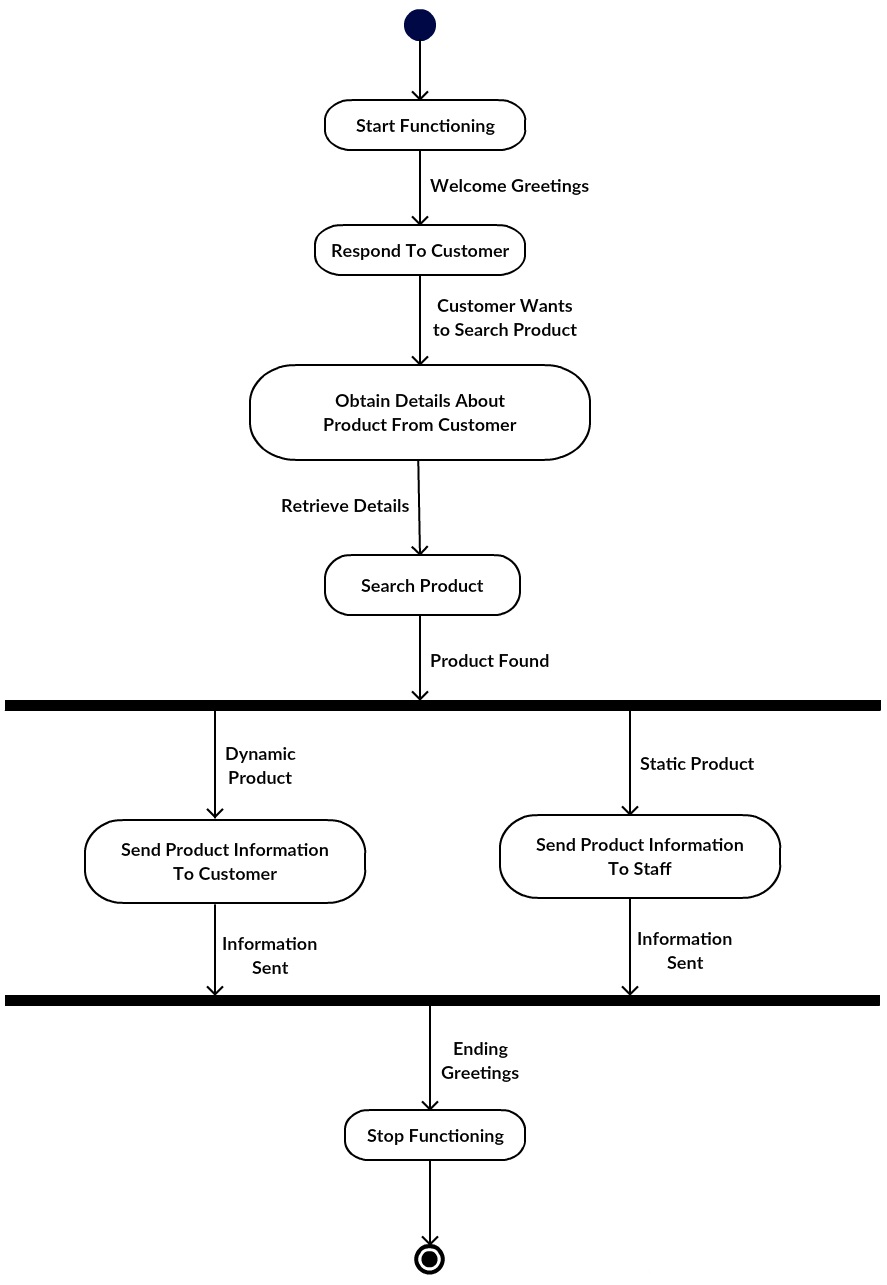
**5.4. Activity Diagram**

**5.4.(a). Customer**

****

**Figure 5.4.(a). Activity Diagram**

**5.4.(b). Shoppy-The Shopping Mall BOT**

****

**Figure 5.4. (b). Activity Diagram**

**6 SYSTEM TESTING**

**6.1 TEST CASES**

**Test Case 1: Find item**

1. Open the Shopping Mall BOT web portal and login

2. Press the microphone button and use command “**Where can I find item\_name**”

**OR**

2. If you want to find the item using image then upload the image using the camera button

**Result:** Shoppy BOT will search the item in database and will give you its location if that item is available and in the second case it will give you similar items as result.

**Test Case 2: Add an Item**

1. Open the Shopping Mall BOT web portal and login

2. Press microphone button and use command “**add item\_name**”

**Result:** Shoppy BOT will search the availability of the item and if it is found then it will add that item in the list

**Test Case 3: Remove an Item**

1. Open the Shopping Mall BOT web portal and login

2. Press microphone button and use command “**remove item\_name**”

**Result:** Shoppy BOT will remove the item from your shopping list

**Test Case 4: Display the list**

1. Open the Shopping Mall BOT web portal and login

2. After adding items in your list press microphone button and use command “**show list**”.

**Result:** Shoppy BOT will display the list items if there is at least one item added.

**Test Case 5: Confirm the list**

1. Open the Shopping Mall BOT web portal and login

2. After the list is created press microphone button and use command “**confirm list**”

**Result:** If the list is not empty it will be sent to the staff member and the member will keep the things ready on the counter.

**Test Case 6: Know total amount**

1. Open the Shopping Mall BOT web portal and login

2. After creating the list press microphone button and use command “**what is the amount**”

**Result:** Shoppy will calculate and give you the total amount

**Test Case 7: Delete the list**

1. Open the Shopping Mall BOT web portal and login

2. If the list is created but now the user wants to delete it than press microphone button and use command “**delete list**”

**Result:** Shoppy will delete the list.

**Test Case 8: Showing offers**

1. Search the offer using “**Offers in \***”

2. The server will fetch offers and will display the result

**Result:** Shoppy will display offers

**Test Case 9: To change the quantity**

1. Use “**change quantity**” command

2. Shoppy will display the list, choose the item by using “**number**”command

3.Use “**quantity**” command and change the quantity

**Result:** Shoppy will change the quantity of the item

**7 CONCLUSION**

* The goal was to achieve a Shopping BOT Assistant which helps customer in shopping in mall by guiding them and handling basic tasks. This hence proved out to be extremely useful for the customers.
* During the period of the project we came to know the practical aspects of Java Web Development that sound our knowledge in Java and Advanced Java Topics, JavaScript, SQLite Database Connectivity and Google API Services. Thus, it improved the skills in various fields and the eagerness to learn more.

**8. BIBLIOGRAPHY**

**Reference site:**

* [**https://sdkcarlos.github.io/sites/artyom.html**](https://sdkcarlos.github.io/sites/artyom.html)
* [**https://www.java.com/en/**](https://www.java.com/en/)
* [**https://stackoverflow.com**](https://stackoverflow.com)
* [**https://github.com/**](https://github.com/)

**Reference Books:**

* Object Oriented Modelling and Design with UML (second edition)

By Michael Blaha and James Rambaugh

* Software Engineering by Roger S. Pressman