Dharmsinh Desai University
Nadiad

Faculty of Technology

Department Of Computer Engineering

B.Tech. CE Semester- VII

Subject : System Design Practice

Project Title : Online Eatery Ordering System

Prepared by:

Manthan Doshi (ID :106017)

Anand Naik (ID :102068)

Guided by : Prof. M. T. Mehta

DHARMSINH DESAI UNIVERSITY

NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project carried out in the subject of Software Design Practice, entitled "**Online Eatery Ordering System**" and presented in this report is a bona fide report of work done by **Manthan Doshi** and **Anand Naik** of Department of Computer Engineering, semester VII.

Prof. M. T. Mehta,	Prof. C. K. Bhensdadia,
Project guide	Head
Department of Computer Engineering	Department of Computer Engineering
Date:	Date:

CONTENTS

ABSTRACT

Online Eatery Ordering System(OEOS) is an android based application intended to meet new era of Internet Shopping. This application is designed in a way that users with android enabled devices can place order for food items from Eatery Shop, through internet without having to visit the shop. The order is processed by server, which then sends appropriate response to client. This document is meant to delineate the features of OEOS, so as to serve as a guide to the developers on one hand and as software validation document for the prospective client on the other.

1. INTRODUCTION

1.1 Problem Details:

Online Eatery Ordering System (OEOS) provides an easy way through an Android application to order food from Online Eatery to registered users. Customer can easily navigate through the menu and view different food items.

1.2 Purpose:

The OEOS is an application that allows customers to place order online by selecting from available items and allows manger/administrator to view and manipulate orders received and maintained in database. Eliminating the need of customers to go to eatery personally to place an order. Maintain consistency among different access modes, e.g. by phone, by web, at the door-delivery desk and across different outlets of eatery.

1.3 Scope:

We describe what features are in the scope of the software and what are not, for the software to be developed.

In Scope:

a. To view food-items menu online through Android application.
b. To order food-items online through Android application.
c. To provide client for registering at server.
d. To provide client for giving his/her feedback about service.
e. To view all orders placed by android application.
f. To process ordered items at server and maintaining them in database.
g. To provide manager at server, GUI to view, analyze and manipulate orders.
h. To provide manger at server, GUI to generate reports.

Not in Scope:

a. To find nearest centers of Eatery.
b. To verify client's order based on his location.

1.4 Platform:

Android

Windows

1.5 Technology Used:

Android

Java 2 SE

JSP

Servlets

1.6 Tools used:

Eclipse IDE

ADT

NetBeans IDE

1.7 Software Development Model:

Iterative Waterfall Model

1.8 User Characteristics:

- Android application can be used with basic knowledge of operating Android based device.
- Server application can be used with basic knowledge of using a web browser.

2 SOFTWARE REQUIRE SPECIFICATION

2.1 Functional Requirements

User Class : Customer.

R1 User Registration.

Input: User provides email, phone no, address.

Processing: Application stores the user details.

Output: Application directs to Home page.

R2 Login.

Input: User provides phone number as id.

Processing: Application validates the user.

R3 View Menu.

Input: User selects "View Menu" option.

Output: Application displays the menu.

R4 Place Order.

R4.1 Select Category.

Input: User selects an eatery category.

Output: Application displays items in that category.

R4.2 Select Items.

Input: User selects items.

Processing: Selected item is added to current order.

Output: Selected Items are displayed checked.

R4.3 Select Quantity.

Input: User specifies required quantity.

Processing: Selected quantity of current item is added to current order.

R4.4 Modify Order.

R4.4.1 Add Items

Input: User selects more items to be ordered.

Output: Application displays modified order.

R4.4.2 Remove Items

Input: User selects items to be removed.

Output: Application displays modified order.

R4.5 Select Delivery Address.

Input: User selects either registered address or specifies new address.

Processing: Application incorporates specified address in Order.

R4.6 Confirm Order.

Input: User selects "Order" Option to proceed to place order.

Processing: Application requests order to manager and waits for response.

R5 Feedback.

Input: User provides his feedback.

Processing: Application forwards feedback to manager.

User Class : Manager.

R1 Process Order.

R1.1 Accept Order.

Description: Manager selects 'Accept Order' option upon receiving notification for new order. Current Order is added to pool of orders and is high-lighted.

R1.1.1 Generate Order Receipt.

Input: Manger selects "Print Order" Option.

Output: System prints the received order for cook.

R1.1.2 Reply with estimated delivery time to customer.

Input: Manager specifies estimated time of delivery.

Processing: System replies to customer's application.

R1.2 Reject Order.

Input: Manager cancels order and records the reason.

Processing: Current Order in pool is de-high-lighted and message is sent to customer.

R2 Generate Bill.

Input: Manager selects "Print Bill" option for delivery.

Output: System generates printed receipt of bill.

R3 Generate Report.

Input: Manager specifies report query.

Output: System generates corresponding report.

2.2 Non Functional Requirements

- **Performance**
  - Response time of the OEOS should be less than 2 second most of the time. Response time refers to the waiting time while the system accesses, queries and retrieves the information from the databases (DB-user, DB-schedule etc) (A local copy of database is maintained as DB-schedule to reduce this access time).
  - OEOS shall be able to handle at least 1000 transactions/inquiries per second.
  - OEOS shall show no visible deterioration in response time as the number of customer requests increases.

- **Reliability**
  - OEOS shall be available 24 hours a day, 7 days a week.
  - OEOS shall always provide real time information about Eatery processing.
  - OEOS shall be robust enough to have a high degree of fault tolerance. The system should not crash in case of invalid input and shall identify the invalid input and produce a suitable error message.
  - OEOS shall be able to recover from hardware failures, power failures and other natural catastrophes and rollback the databases to their most recent valid state.

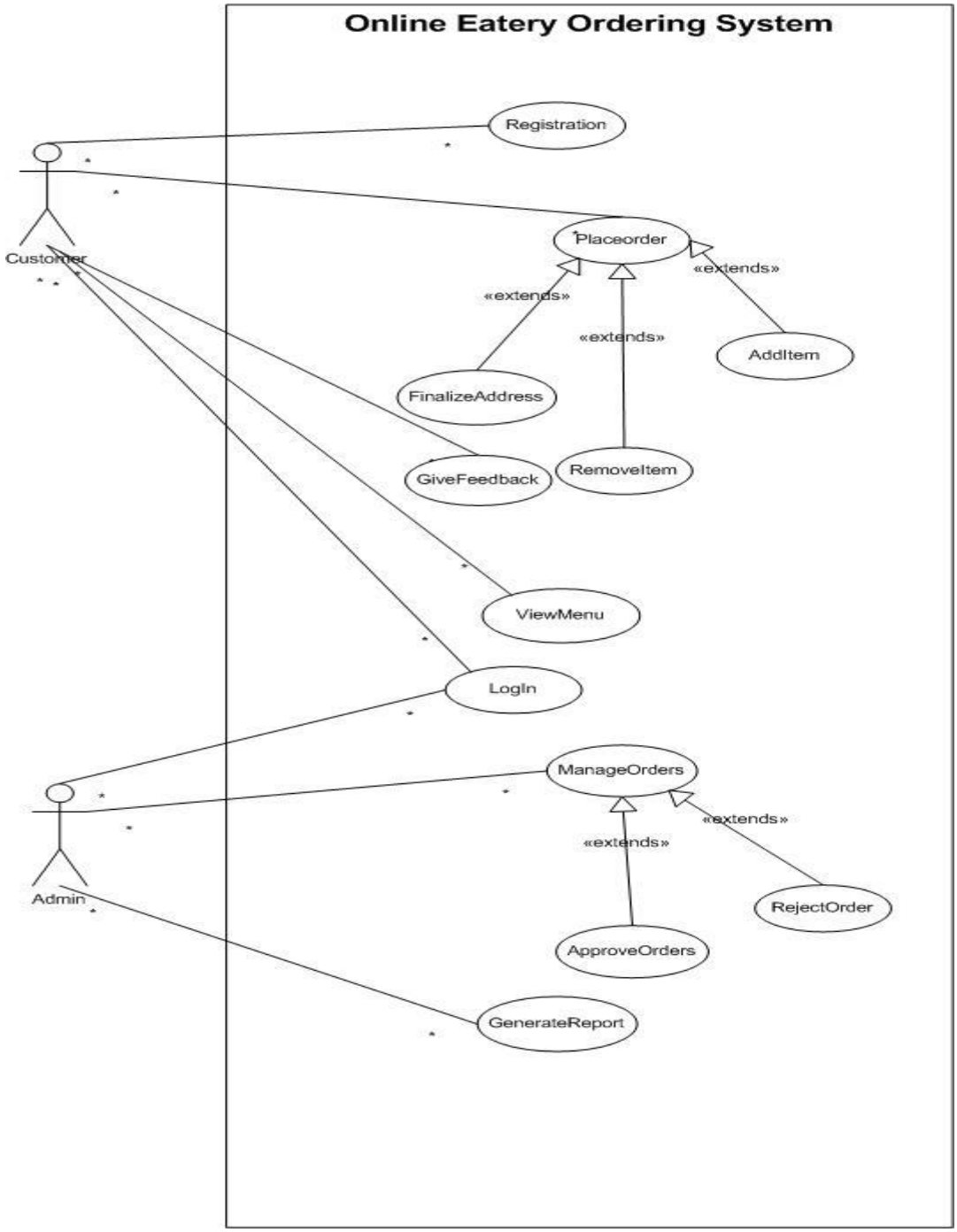
- **Usability**
  - OEOS shall provide a easy-to-use graphical interface similar to other existing systems so that the users do not have to learn a new style of interaction.
  - The web interface shall be intuitive and easily navigable. Users shall be able to understand the menus and options provided by OEOS.
  - Any notification or error messages generated by OEOS shall be clear, succinct, polite and free of jargon.

- **Integrity**
  - Only system administer has the right to change system parameters, such as pricing policy etc. The system should be secure and must use encryption to protect the databases.
  - Users need to be authenticated before having access to any data of Eatery.

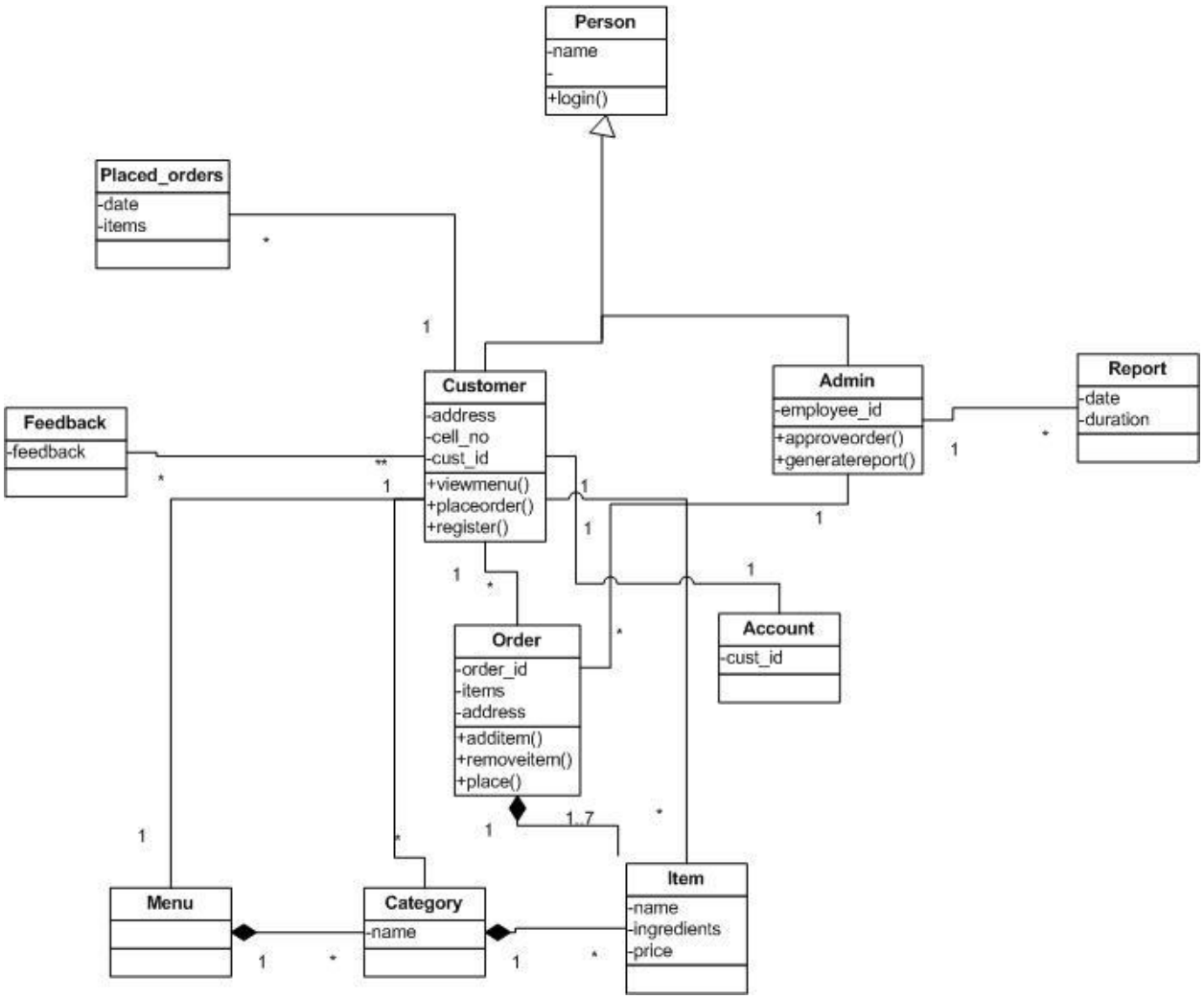
- **Interoperability**
  - OEOS shall minimize the effort required to couple it to another system.

3 SYSTEM DESIGN

3.1 Use Case Diagram

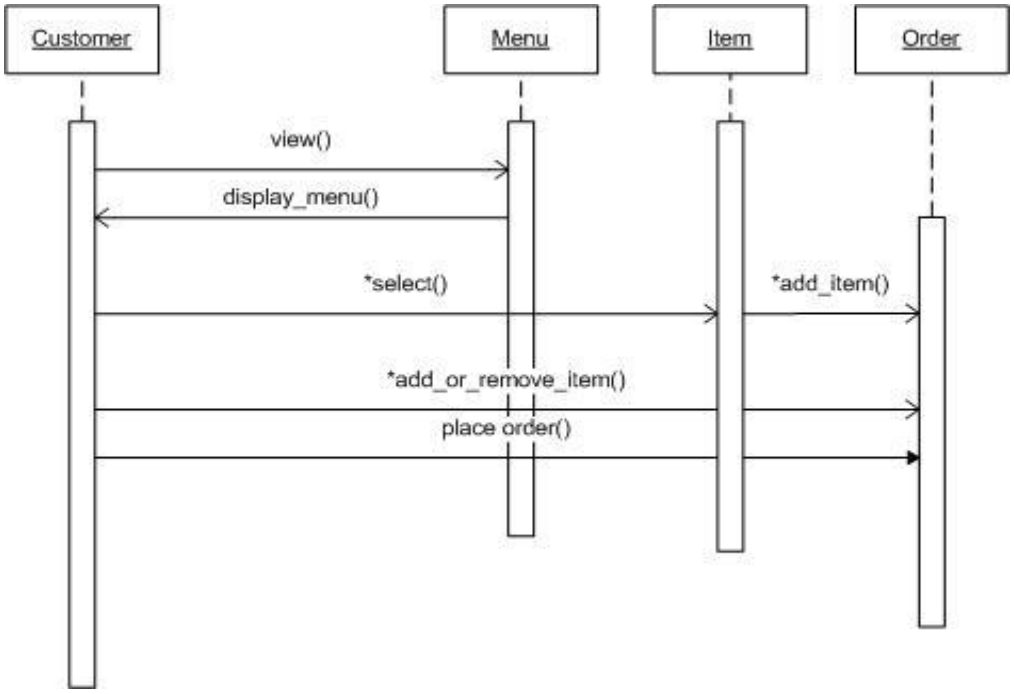


3.2 Class Diagram

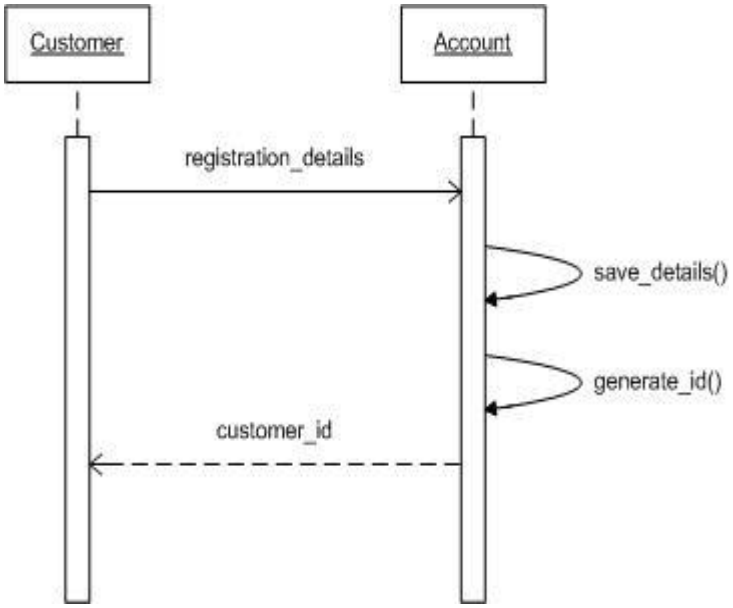


3.3 Sequence Diagrams

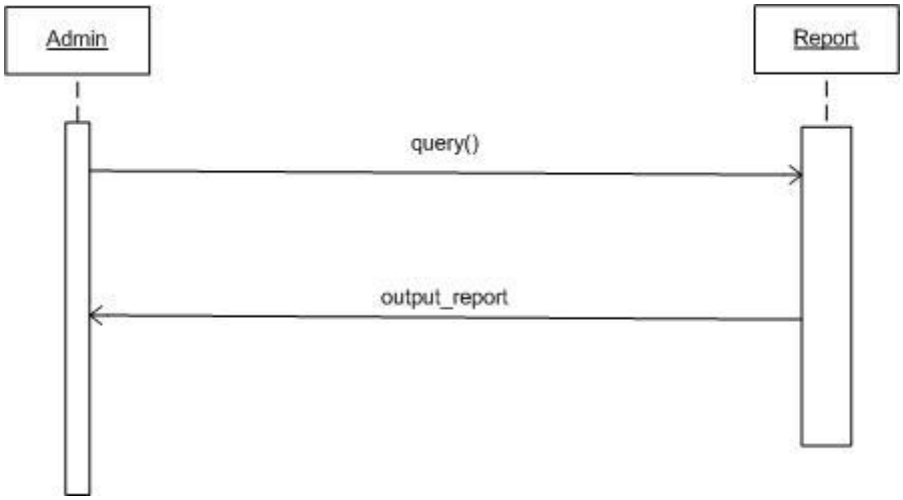
1.) Sequence Diagram for 'PlaceOrder' usecase:



2.) Sequence Diagram for 'Registration' usecase:

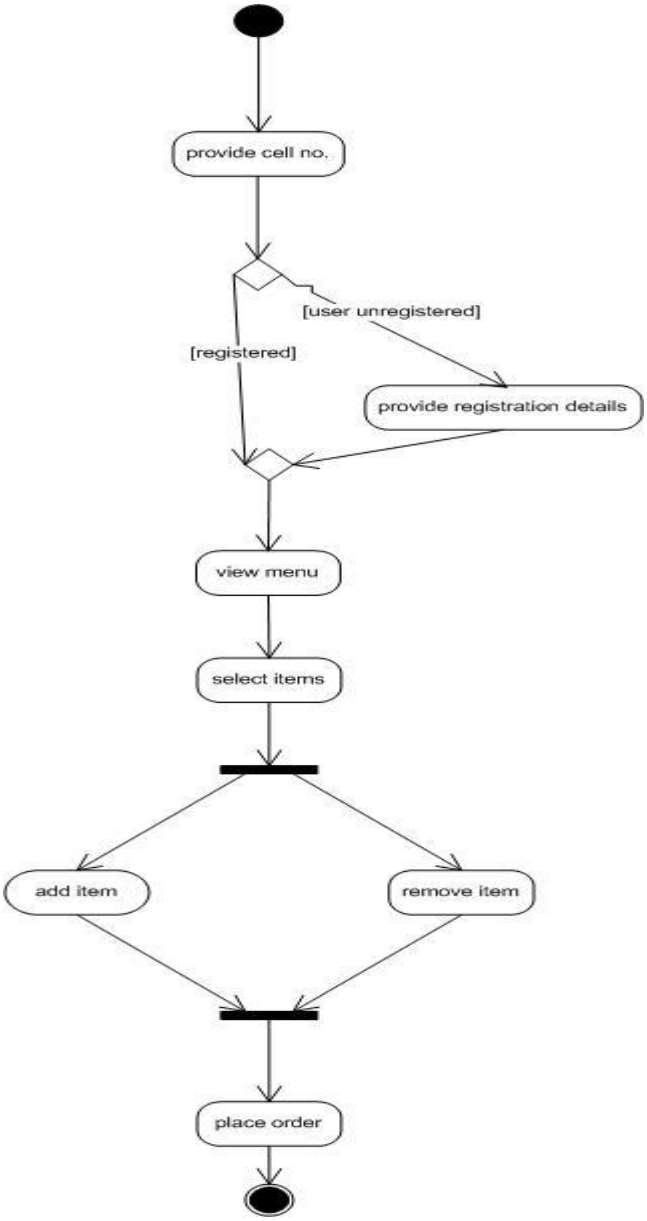


3.) Sequence Diagram for ‘Generate Report’ usecase (Server side):

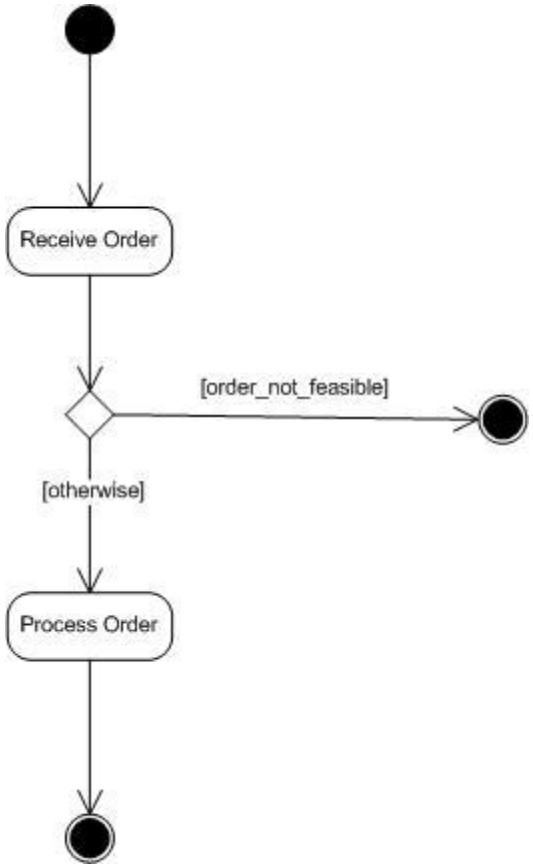


3.4 Activity Diagrams

1.) Activity Diagram for OEOS Android application:

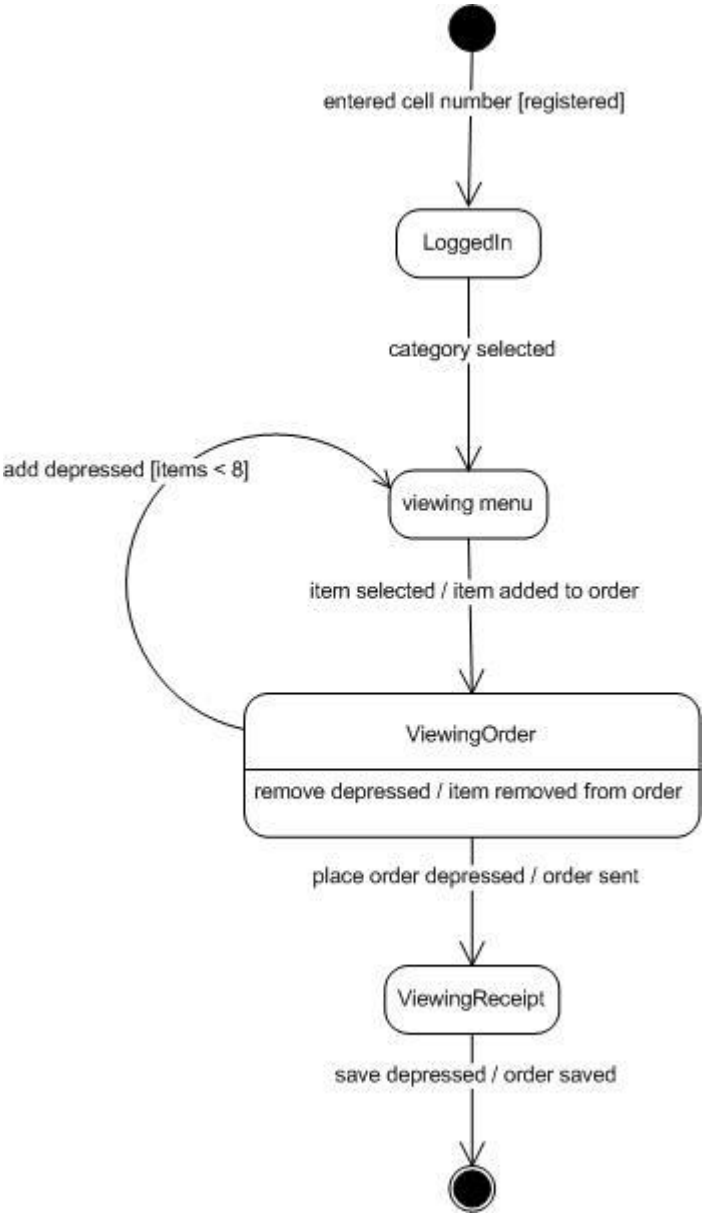


2.) Activity Diagram for ‘Approve Order’ usecase (Server side):



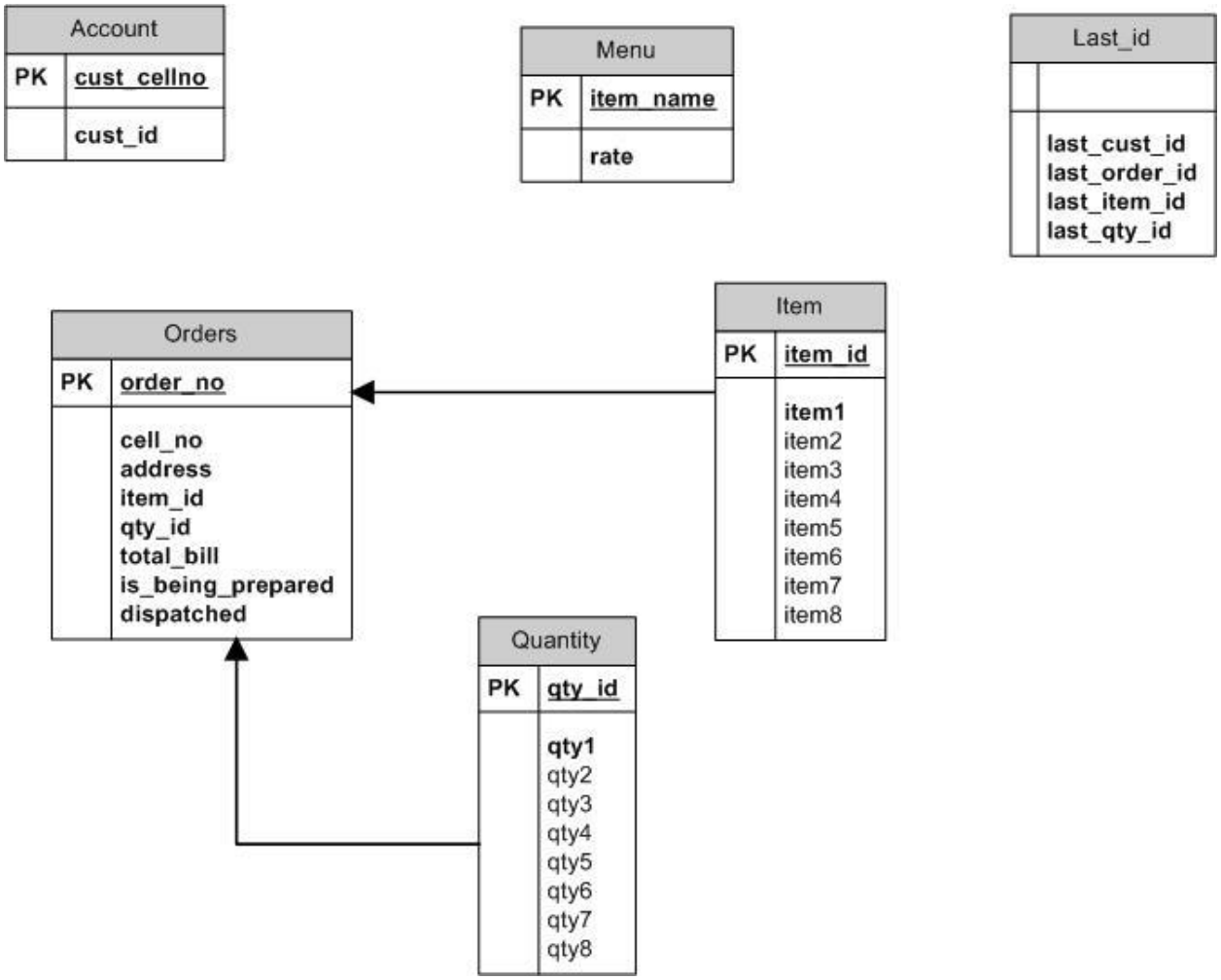
3.5 State-Chart Diagram

State-chart diagram for ‘Customer’ object



3.6 E-R Diagram

E-R Diagram for the Server side Database:



3.7 Data Dictionary

3.7.1 Orders

Attributes	Types	Constraints
Order_no	Varchar(10)	Primary key
Cell_no	Decimal(12,2)	Not Null
Address	Varchar(50)	Not Null
Item_id	Decimal(5,2)	Not Null
Qty_id	Decimal(5,2)	Not Null
Total_bill	Decimal(5,2)	Not Null
Is_being_prepared	Decimal(1,2)	Not Null
Dispatched	Decimal(1,2)	Not Null

3.7.2 Item

Attributes	Types	Constraints
Item_id	Decimal(5,2)	Primary Key
Item1	Varchar(10)	Not Null
Item2	Varchar(10)	
Item3	Varchar(10)	
Item4	Varchar(10)	
Item5	Varchar(10)	
Item6	Varchar(10)	
Item7	Varchar(10)	
Item8	Varchar(10)	

3.7.3 Quantity

Attributes	Types	Constraints
Qty_id	Decimal(5,2)	Primary Key
Qty1	Varchar(10)	Not Null
Qty2	Varchar(10)	
Qty3	Varchar(10)	
Qty4	Varchar(10)	
Qty5	Varchar(10)	
Qty6	Varchar(10)	
Qty7	Varchar(10)	
Qty8	Varchar(10)	

3.7.4 Account

Attributes	Types	Constraints
Cust_cellno	Varchar(12)	Primary key
Cust_id	Decimal(10,2)	Not Null

3.7.5 Menu

Attributes	Types	Constraints
Item_name	Varchar(20)	Primary key
Rate	Decimal(4,2)	Not Null

3.7.6 Last_id

Attributes	Types	Constraints
Last_cust_id	Decimal(4,2)	Not Null
Last_order_id	Decimal(12,2)	Not Null
Last_item_id	Decimal(5,2)	Not Null
Last_qty_id	Decimal(5,2)	Not Null

4 IMPLEMENTATION

MODULES

Some of the main modules of the applications are described below.

Android Application Module:

1 Login.

2 Registration.

3 Place Order.

4 Feedback.

5 Display Menu.

Manager Application Module:

6 Login.

7 Validate Order.

8 Generate Report.

1. **Login Module:**
    - User can log-in using his mobile number into the Android application and place an order.

2. **Registration Module:**
    - If a user is not registered, on submitting his mobile number he will be redirected to the registration page. On this page, he will be required to fill his details (name, mobile number, address) and submit.

3. **Place Order Module:**
    - Using this module user can place a new order.
    - User can add and remove food items in his order.
    - User can also specify delivery address for his order.

4. **Feedback Module:**

- User can submit his feedback about the food or the application through this module.

5. **Display Menu Module:**
    - User is shown the menu that is available at the eatery, also various categories that are available in the menu.

6. **Login Module:**
    - Administrator can log-in using appropriate credentials to access database of orders received.

7. **Validate Order Module:**
    - Administrator can either accept or reject order based on currently accepted orders.

8. **Generate Report Module:**
    - Administrator can generate report of orders received and total sales done by specifying start and end dates.

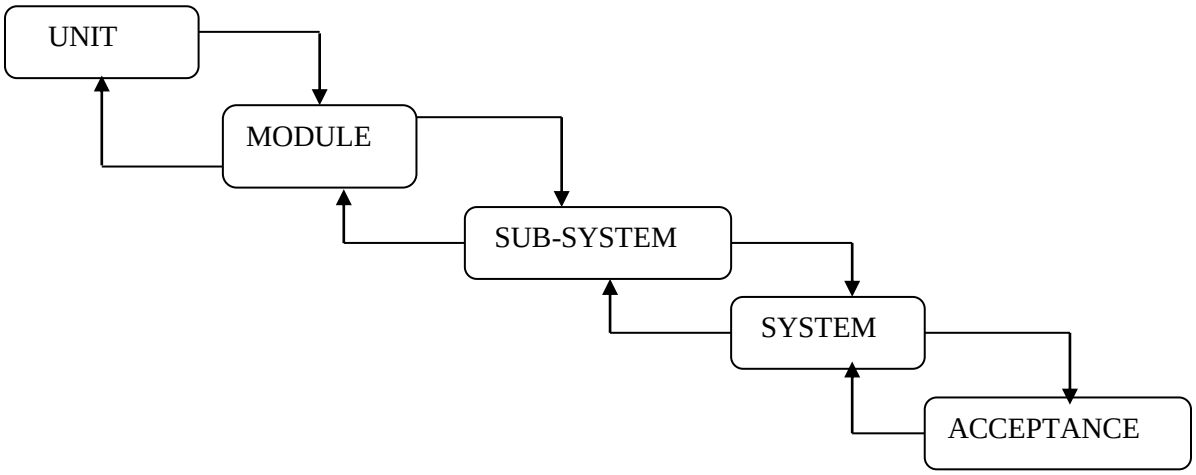
5 TESTING

Testing is the process carried out on software to detect the differences between its behavior and the desired behavior as stipulated by the requirements specifications. Testing is advantageous in several ways. Firstly, the defects found help in the error correction, testing gives an idea as to how reliable the software is. Secondly, over time, the record of defects found reveals the most common kind of defects, which can be used for developing appropriate preventive measures such as training, proper design and reviewing.

5.1 Testing Plans

The testing sub-process includes the following activities in a phase dependent manner:

a) Create Test Plans.
b) Create Test Specifications.
c) Review Test Plans and Test Specifications.
d) Conduct tests according to the Test Specifications, and log the defects.
e) Fix defects, if any.
f) When defects are fixed continue from activity.



Testing Plan

5.2 Testing Strategies

The development process repeats this testing sub-process a number of times for the following phases.

a) Unit Testing.
b) Integration Testing

Unit Testing tests a unit of code (module or program) after coding of that unit is completed. Integration Testing tests whether the various programs that make up a system, interface with each other as desired, fit together and whether the interfaces between the programs are correct. System Testing ensures that the system meets its stated design specifications. Acceptance Testing is testing by the users to ascertain whether the system developed is a correct implementation of the Software Requirements Specification.

Testing is carried out in such a hierarchical manner to ensure that each component is correct and the assembly/combination of components is correct. Merely testing a whole system at the end would most likely throw up errors in components that would be very costly to trace and fix.

We have performed both Unit Testing and System Testing to detect and fix errors. A brief description of both is given below.

Unit Testing

Objective:

The objective of Unit Testing is to test a unit of code (program or set of programs) using the Unit Test Specifications, after coding is completed. Since the testing will depend on the completeness and correctness of test specifications, it is important to subject these to quality and verification reviews.

Input: Unit Test Specifications

Testing Process:

- Checking for availability of Code Walk-through reports which have documented the existence of and conformance to coding standards.
- Review of Unit Test Specifications.

  Verify the Unit Test Specifications confirm to the program specifications.

  Verify that all boundary and null data conditions are included.

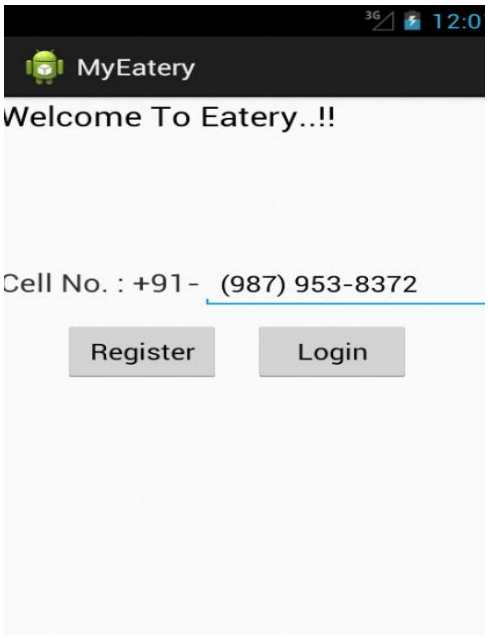
5.3 Test Cases:

Srno.	Module	Test Case Name	Expected result	Actual result	Pass/Fail
1	Login (User /Admin)	Valid CellNo or Username and Password	Successful Login	Successful Login	Pass
		Invalid CellNo or Username or Password	Invalid CellNo error or Invalid Admin Credentials error	Invalid CellNo error or Invalid Admin Credentials error	Pass
2	Registration	All the details are valid	Redirected to Home Page	Redirected to Home Page	Pass
		One or more invalid detail(s)	Error message as pop-up	Error message as pop-up	Pass
3	Place Order	Valid items and quantities ordered with total items less than 8	Order request sent to server	Order request sent to server	Pass
4	Feedback	Message of length less than 80	Feedback submitted and app redirected to home page	Feedback submitted and app redirected to home page	Pass
5	Display Menu	Category selected from available list	Corresponding list of items displayed	Corresponding list of items displayed	Pass
6	Validate Order	'Accept' Order selected	Order showed to be in process and recorded	Order showed to be in process and recorded	Pass
		'Reject' Order selected	Order not recorded and message sent to android application	Order not recorded and message sent to android application	Pass
7	Generate Report	Providing valid start and end dates	Corresponding Report Displayed	Corresponding Report Displayed	Pass

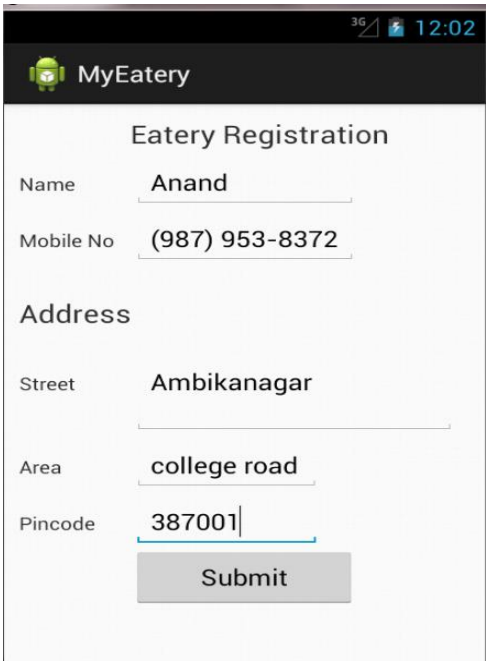
6 SCREEN SHOTS

Android Application:

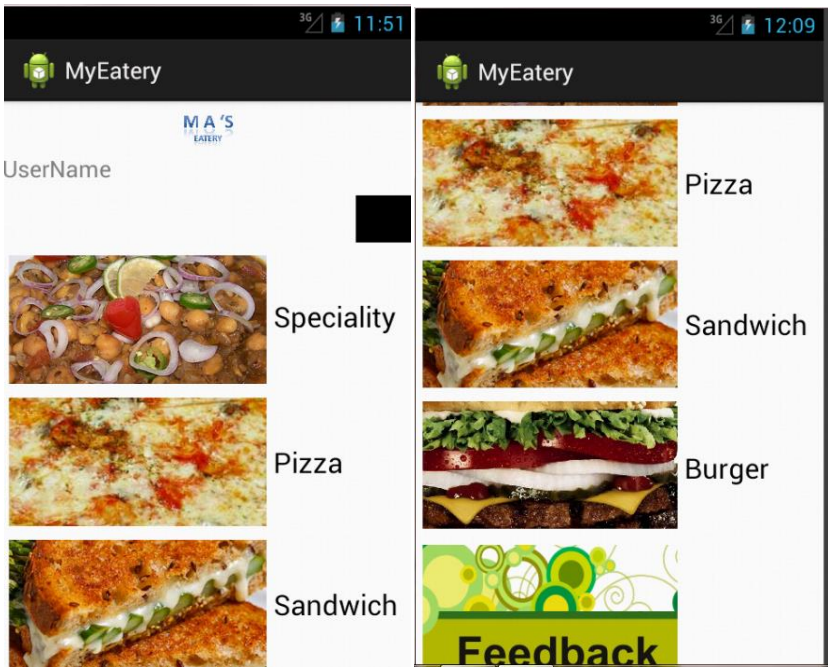
1.) Login Activity



2.) Register Activity

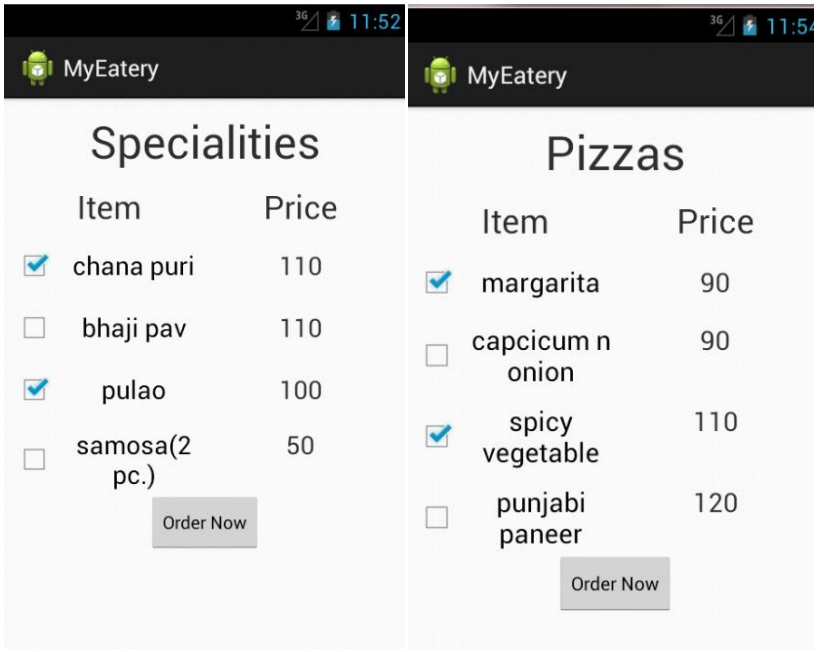


3.) Home Activity

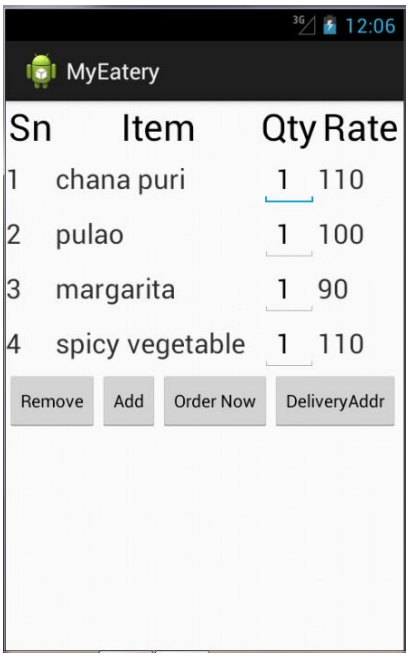


4.) Menu Activity

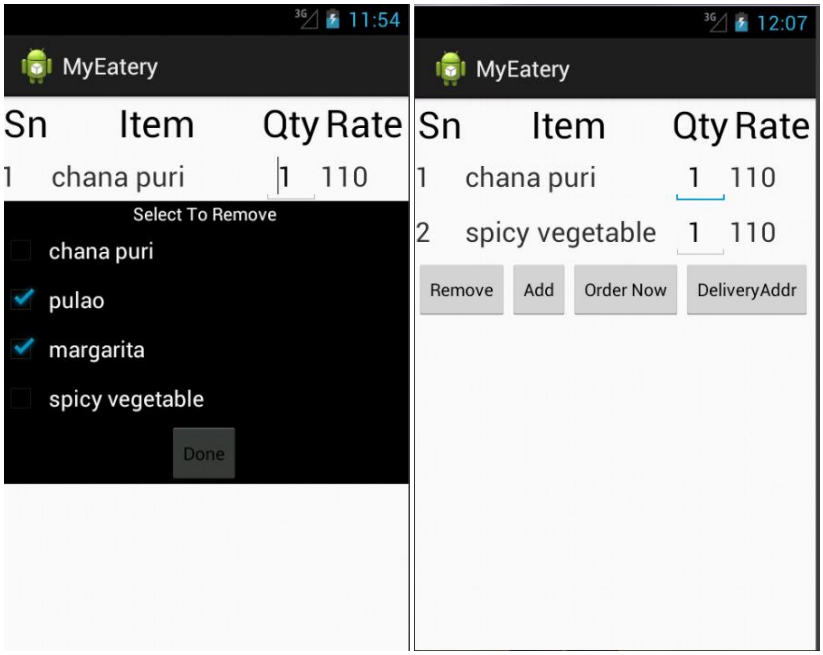
5.) Category Activity



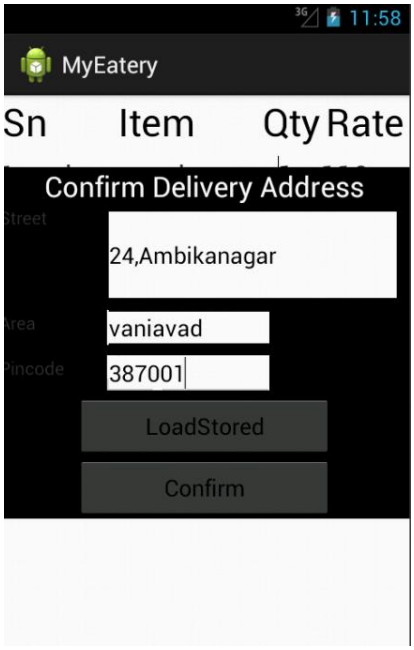
6.) Place Order Activity



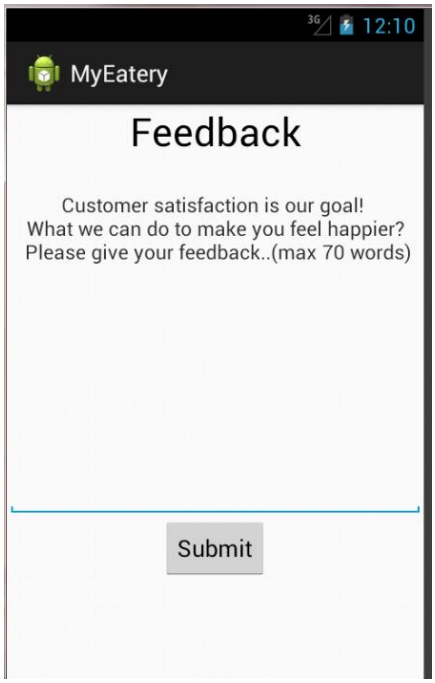
7.) Remove Action Pop-up



8.) Confirm Address Pop-up

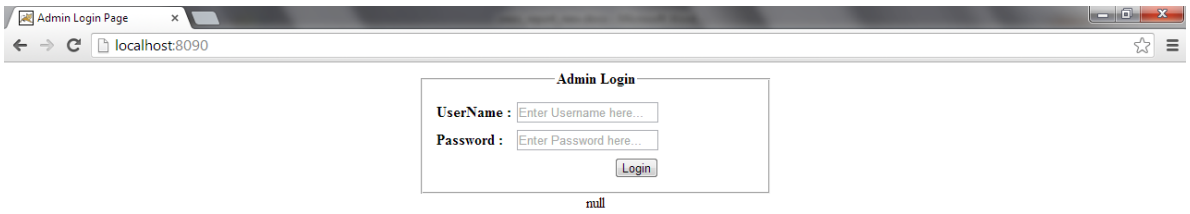


9.) Feedback Activity

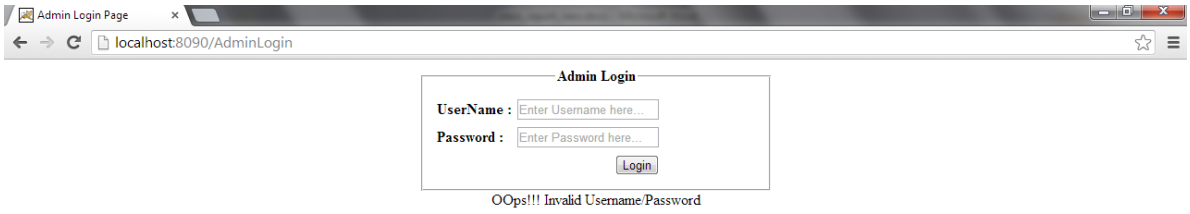


Manager Application:

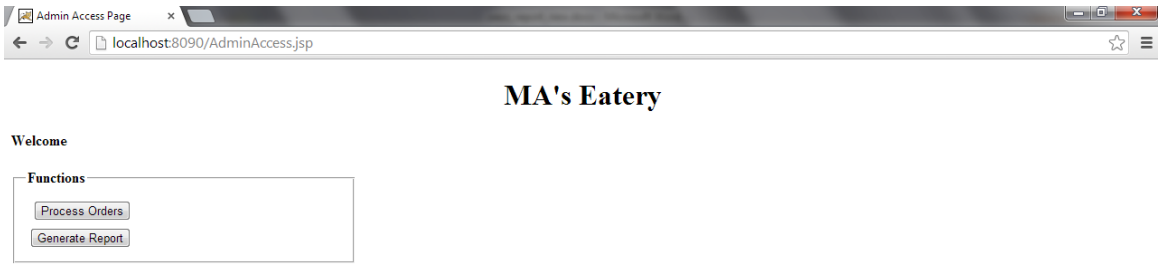
1.) Admin Login



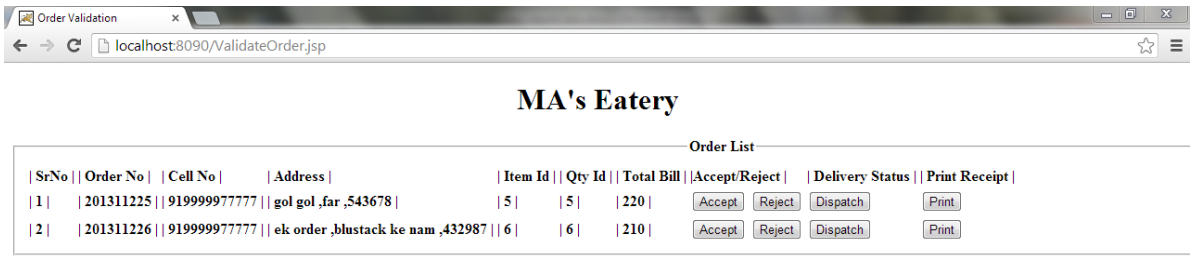
2.) Invalid Login Credentials



3.) Admin Access



4.) Validate Order Action



7 CONCLUSION

"Time neither wait nor return". Thus coping up with the constantly evolving era is a challenging task. In today's era of mobile and technology, every minute a new application, a new technology is being developed. To add to the luxuries of a man, our application gives android device users, tool to place order of food items, online.

Though a simple application, the main purpose of ordering online is well satisfied by this application. Other modifications can be made to improve features of app. We look forward to implementing more features to our app.

8 FUTURE EXTENSION

Limitation

- Not possible to modify menu from server side. Menu modification requires reinstallation of android application.
- An order can have at most eight items at a time.
- Report can be generated for predefined period only.

Future enhancements

- GUI can be improved.
- Location based search and user authentication can be implemented.
- Menu modification from server side can be implemented.
- Report generation can be made more interactive.
- More functionalities can be added on server side.

9 BIBLIOGRAPHY

Websites

- developer.android.com
- www.stackoverflow.com
- www.javapoint.com
- www.jspjdbc.blogspot.in

Books

- Head First JSP and Servlets
- Beginning Android Application Development - Wrox Application