SRS FOR FOOD DELIVER APPLICATION

BY

VARSHINI M

RUPALI DEBNATH

JALADANKI DIVYA

DOMMARAJU DIVYA SREE

TABLE OF CONTENT

6.2.
Use Case Diagram:
6.3.
Sequence Diagram
6.4.
Activity Diagram
6.5.
State chart Diagram
6.6.
Object Diagram

7. Implementation Planning

7.1.
Implementation Environment:
7.2.
Program/Modules Specification
7.2.1.
Security Features
7.2.2.
Coding Standards

8. Testing

8.1.
Testing Plan
8.2.
Testing strategy
8.3.
Testing Methods
8.4.
Test Cases
8.4.1.
Purpose
8.4.2.
Required Input
8.4.3.
Expected Result

9. Limitation and Future Enhancement:

10. Conclusion and Discussion:

Introduction :

This Software Requirement Specification document provides a complete description of all the functionalities and the specifications of the Food Delivery. The developers and the testers can use
this document as a reference for developing the design and test plan documents.

1. Purpose:

The Software Requirements Specification (SRS) will provide a detailed description of the requirements for the Food Delivery (FD). The clear understanding of the FD and its' functionality
will allow for the correct software to be developed for the end user and will be used for the development of the future stages of the project. This SRS will provide the foundation for the project.
From this SRS, the FD can be designed, constructed, and finally tested.
This SRS will be used by the software engineers constructing the FD and the Food Delivery receiver
end users. The software engineers will use the SRS to fully understand the expectations of this FD
to construct the appropriate software. The Food Delivery end users will be able to use this SRS as a
"test" to see if the software engineers will be constructing the system to their expectations. If it is
not to their expectations the end users can specify how it is not to their liking and the software
engineers will change the SRS to fit the end users' needs.

1.1. Definitions, Acronyms, and Abbreviations SRS :

• Software Requirements Specification
• FD – Food Delivery
• Subjective satisfaction – The overall satisfaction of the system End users – The people who will be actually using the system.

1.2. Scope

The software product to be produced is a Food Delivery System which will automate the major hotel
operations. The first subsystem is an Ordering and Delivery System to keep track of reservations and
room availability. The second subsystem is the Tracking and Selling Food System that charges the customer.

2. The Overall Description

Describes the general factors that affect the product and its requirements. This section does not state
specific requirements. Instead it provides a background for those requirements, which are defined in
section 3, and makes them easier to understand.

2.1. Product Perspective

The Food Delivery is an independent stand–alone system. It is totally self- contained.

2.1.1. System interface

The application runs in the latest version of Chrome or Firefox browser on Windows, Linux and
Mac and on smartphones android V8.0+.

2.1.2. User Interfaces

The User Interface Screens are described in table given below.

2.1.3. Software Interfaces

The system will store all data dividing into 4 parts as per the character of database which are Login,
Ordering, Customer Record and payment and customer details. The system shall permit only Management team and database administrator to edit the database and the system will allows customer to cancel the Orders. System will have order detail database which store all the information
of order.
Ordering database store the order information and payment database shows all the payment whichever is due from the customer and customer database provide all the information of customers.
The system shall interface with an Oracle or Access database

2.1.3. Hardware Interfaces

The system shall run on a Microsoft Windows, Apple i0S and Android based system.

2.1.4. Communication Interfaces

The system shall be a standalone product that does not require any communication interfaces.

2.2. Product Functions

Ordering and Delivery System:
• Allows for typing in customer information
• Has a default price rate that is adjustable
• Includes a description field for the changed rate
• When a customer order, the order number will be changed to occupied in the database
• Ability to modify an order
• When no orders are available and a customer would like to extend their order their information will be placed in a database and when there are order available the first customer on the list will have the order
• When a customer order is fulfilled the amount owed is displayed
• Records payment
• Allows for space to write customer's feedback

2.3. User Characteristics

There are two types:-
• Admin
• User

Admin

• Manage whole Application.

• Manage system database

• Develop, implement, and manage Food items.

• Resolve user complaints and answer user's questions regarding Food items.

User

• User means that use our Application.

• User can observe whole Application (only allowed items for user).

• View available Food item.

2.4. Constraints

The project should be completed within specified time period including Planning, Designing, Development, Testing and Deployment.
The project should be completed within specified budget. All the Entry and Exit criteria of all the
stages should be met.
The product should be user-friendly, reliable and should maintain the industry standards without
compromising the quality.
The system architecture and design should be open and in a standard way such that additional functionalities can be added later without much effort.
Company will provide only the software; it is the responsibility of the client company to set up
hardware for running the application.
System should sync frequently to backup server in order to avoid the data loss during failure, so it
can be recovered.

2.5. Assumption and Dependencies

It is assumed that system developed will work perfectly that's going to be developed under the
Windows OS, and Apache Server with Mongo DB database.
If in case of any difficulties, SRS should be flexible enough to change accordingly.

3. Specific Requirement

This section contains all the software requirements at a level of detail, that when combined with the
system context diagram, use cases, and use case descriptions, is sufficient to enable designers to
design a system to satisfy those requirements, and testers to test that the system satisfies those requirements.

3.1. External Interface Requirements

The Food Delivery will use the standard input/output devices for a personal computer. This includes the following:

- Keyboard
- Monitor
- Mouse
- Printer

3.2. Functional Requirements

Functional requirements define the fundamental actions that system must perform.
The functional requirements for the system are divided into various main categories,
Registration,
Logging in, payment, ordering, etc.

3.2.1. Registration

The Customer should be able to register with their details.
The system should record following customer details into member database: **name, email, password, order, address**
The system shall send verification message to email.

3.2.2. Logging In

The system should verify the customer email & password against the member database when logging
in.
After login, member should be directed to Home screen with the menu card with the price range.

3.2.3. Ordering

The system should enable customer to check for availability of Food items.
The system should display rate for all Food items.
The system should allow customer to confirm or cancel the order.
The system should record order details into database.

3.2.4. Manager Access

The system should generate financial and customer report for manager.
The system should enable manager full modification access to customer, ordering, order information.

3.2.5. Payment Management System

The system should allow customer to pay bill via online using credit or debit card.

3.3. Performance Requirements

Performance requirements define acceptable response times for system functionality.
The load time for user interface screens shall take no longer than two seconds.
The log in information shall be verified within five seconds.

3.4. Logical Database Requirements

The logical database requirement include the retention of the following data elements. This list is

not a complete list and is designed as a starting point for development.

Booking/Reservation System
• Customer first name
• Customer last name
• Customer address
• Customer phone number
• Customer feedback
• Payment received (yes/no)
• Payment type
• Total Bill

Food Services:
• Meal
• Meal item
• Meal order
• Meal payment (Credit/Check/Cash)

3.5. Software System Attributes

3.5.1. Reliability

Specify the factors required to establish the required reliability of the software system at time of
delivery.

3.5.2. Availability

The system shall be available during normal hotel operating hours. And everyone can access it for
ordering food.

3.5.3. Security

Customer Service Representatives and Managers will be able to log in to the Food Delivery System.
Customer Service Representatives will have access to the Ordering and Food subsystems. Managers
will have access to the Management subsystem as well as the Ordering and Food subsystems. Access .

3.5.4. Maintainability

The Food Delivery System is being developed in JavaScript. It is an object oriented programming
language and shall be easy to maintain.

4.System Analysis

Under system analysis we will analyses the system operations, their features, functionalities, and
other parameters of measures.

4.1. Study of current system:

The study of current system for Food Delivery system is fully based on paper work. Food Delivery
has to manage all records of customers and orders in papers.

4.2. Problem and Weaknesses of Current System:

Paperwork is tedious job. There is delay and the problem of human error in ordering system and
providing services

4.3. Requirements of New System:

4.3.1. User Requirements:

There should be software which allocate orders automatically and maintain records of customers and
generate bill automatically for customers.

4.3.2. System Requirements:

There should be database backup of food delivery system. There should be Java supported framework for the system. Operating system should be Windows XP or higher version of Windows.

4.4. Feasibility Study:

Does the system contribute to the overall objectives of the organization?
Can the system be implemented using the current technology? Can the system be integrated with
other system which is already in place?

4.5. Requirements Validation:

It is concerned with showing that the requirements actually define the system which the customer
wants.

Features Of New System:

Easy management of ordering and other processes.
Saves time
Can give accurate information about orders when required
Billing becomes easy
Customer information

4.6. Data Dictionar

Field name	Data type	Size	Constrain
User name	varchar	12	Not null
password	Varchar	50	Not null
Id	Numeric	20	Primary key

Table 4.2 Registration Table

Field name	Data type	Size	Constrains
Id	Numeric	-	Primary key
Fame	Varchar	20	Not null
Lame	Varchar	20	Not null
Country	Varchar	30	Not null
City	Varchar	10	Not null
State	Numeric	10	Not null
Gender	Char	15	Not null
Email	Varchar	30	Not null
password	Varchar	100	Not null

Table 4.3 Add Categories

Field name	Data type	Size	Constrains
Catenae	Varchar2	30	Not null
Id	Varcahr2	30	Not null
Cate_code	Varchr2	10	-

Field name	Data type	Size	Constrains
Item name	Char	30	Not null
Item Id	Numeric	10	Not null
Item price	Numeric	-	Not null
Image	Varchar2	100	Not null
descriptions	Char	100	Not null

Table 4.5 Make Order

Field name	Data type	Size	Constrains
Field name	Numeric	5	Primary key
Order no	Numeric	5	Foreign key
Product_id	Numeric	10	Not null
Order price	Numeric	30	Not null
Total amount	Numeric	20	Not null

Table 4.6 Admin

Field name	Data type	Size	Constrains
Admin_id	Varchar2	50	Primary key
Admin_pass	Varchar2	20	Not null
Admin_email	Varchar	10	Not null

Table 4.7 Make Payment

Field name	Data type	Size	Constrains
Payment type	Varchar	10	Not null
Payment_ammount	numeric	15	Not null

5. System Design

5.1. System Application Design

From first form of login for administrator to corresponding other forms.

5.1.1. Method Pseudo code

Algorithm or flow-chart is generated according to the data and the task to be performed.

5.2. Database Design

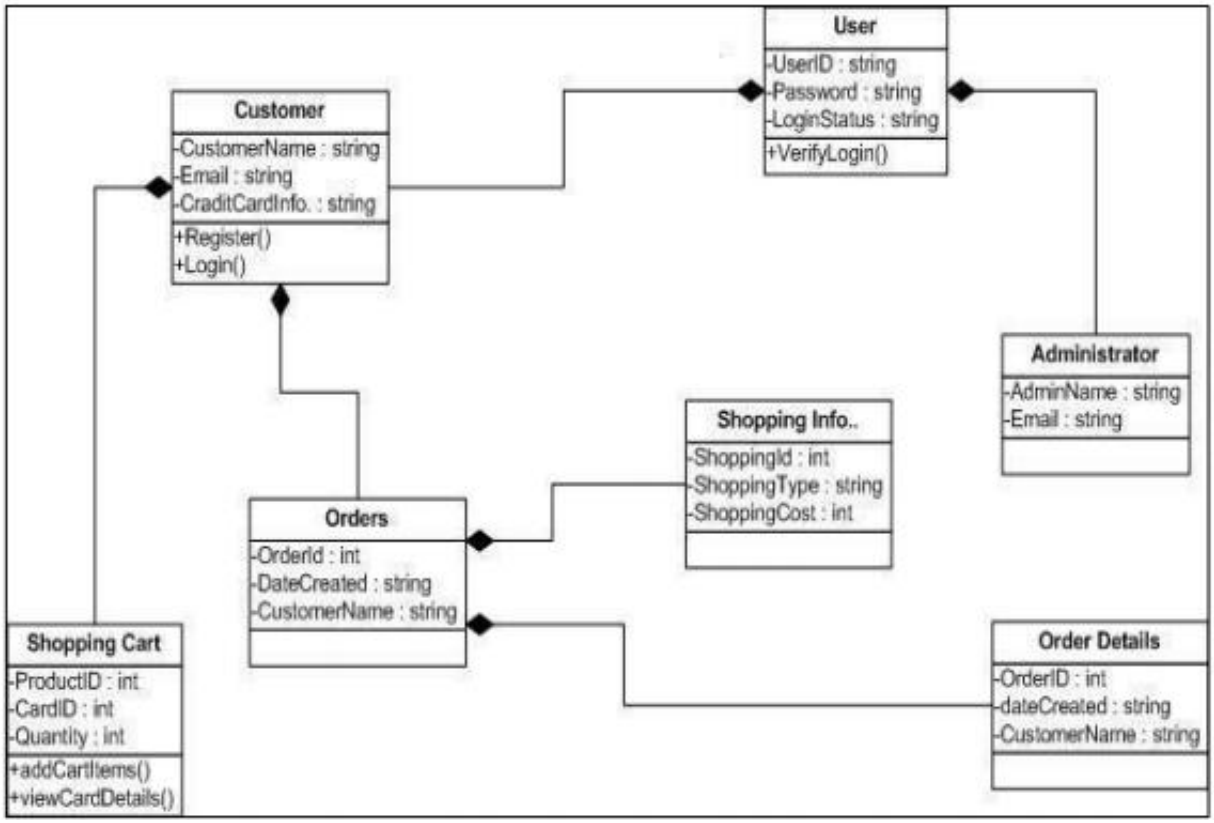
Relational Database is maintained as we know the data required is the same for all.

5.2.1. Table and Relationship

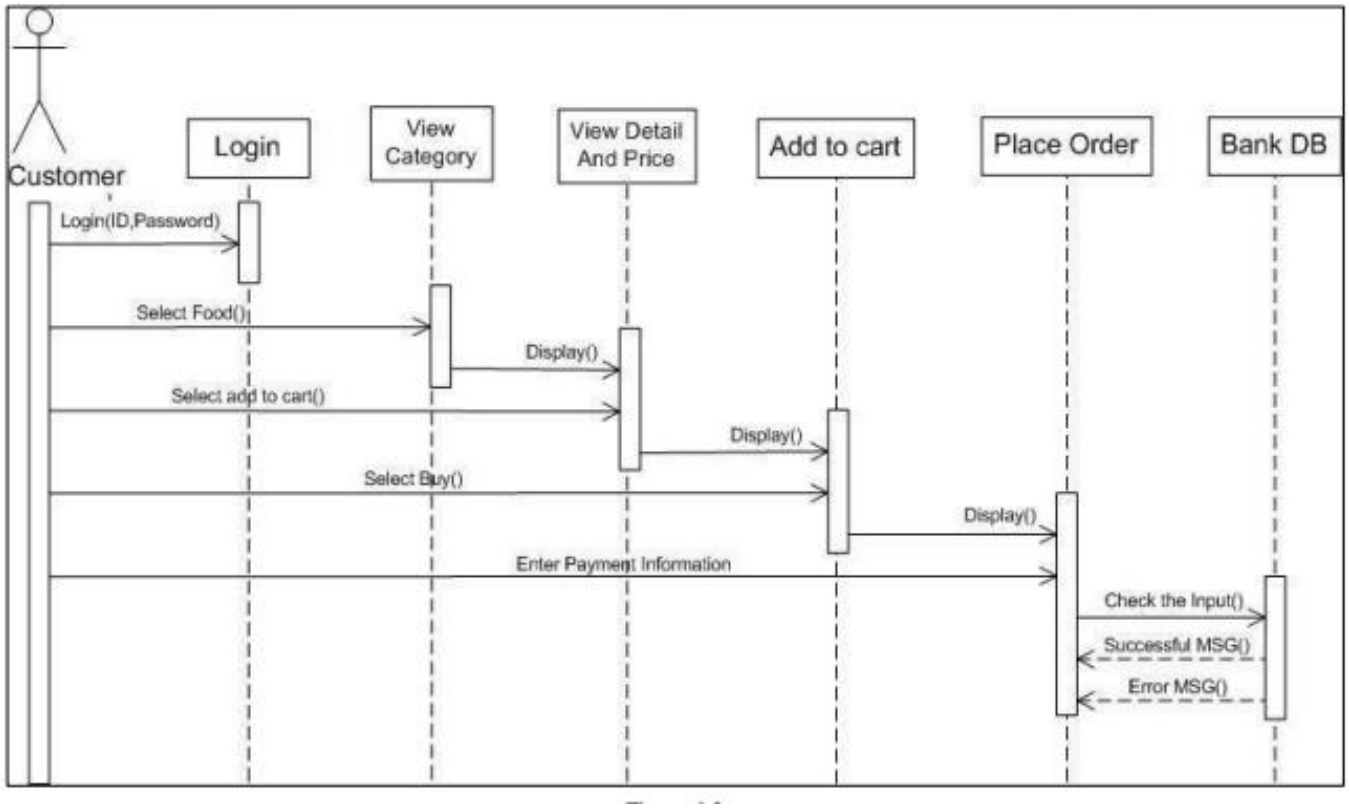
There are 3 tables maintained in MS Access:
• **Order**: Maintains list of all orders of the hotel
• **Customer**: Information about customer
• **Admin**: Administrator login

6. Diagrams

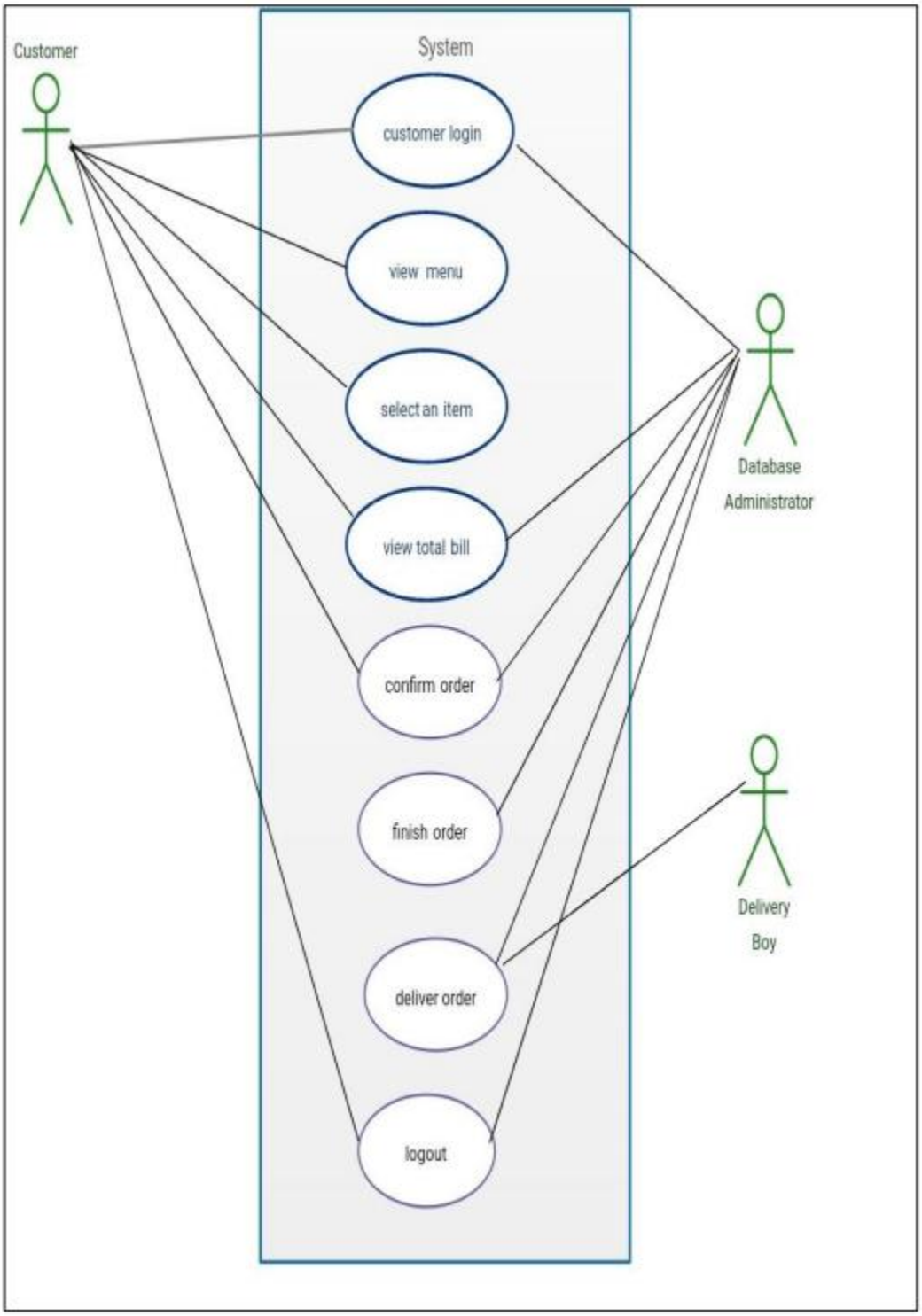
6.1. Class Diagram



6.2. Sequence Diagram



6.3. Use Case Diagram:



6.4. Activity Diagram

Login & Registration:-

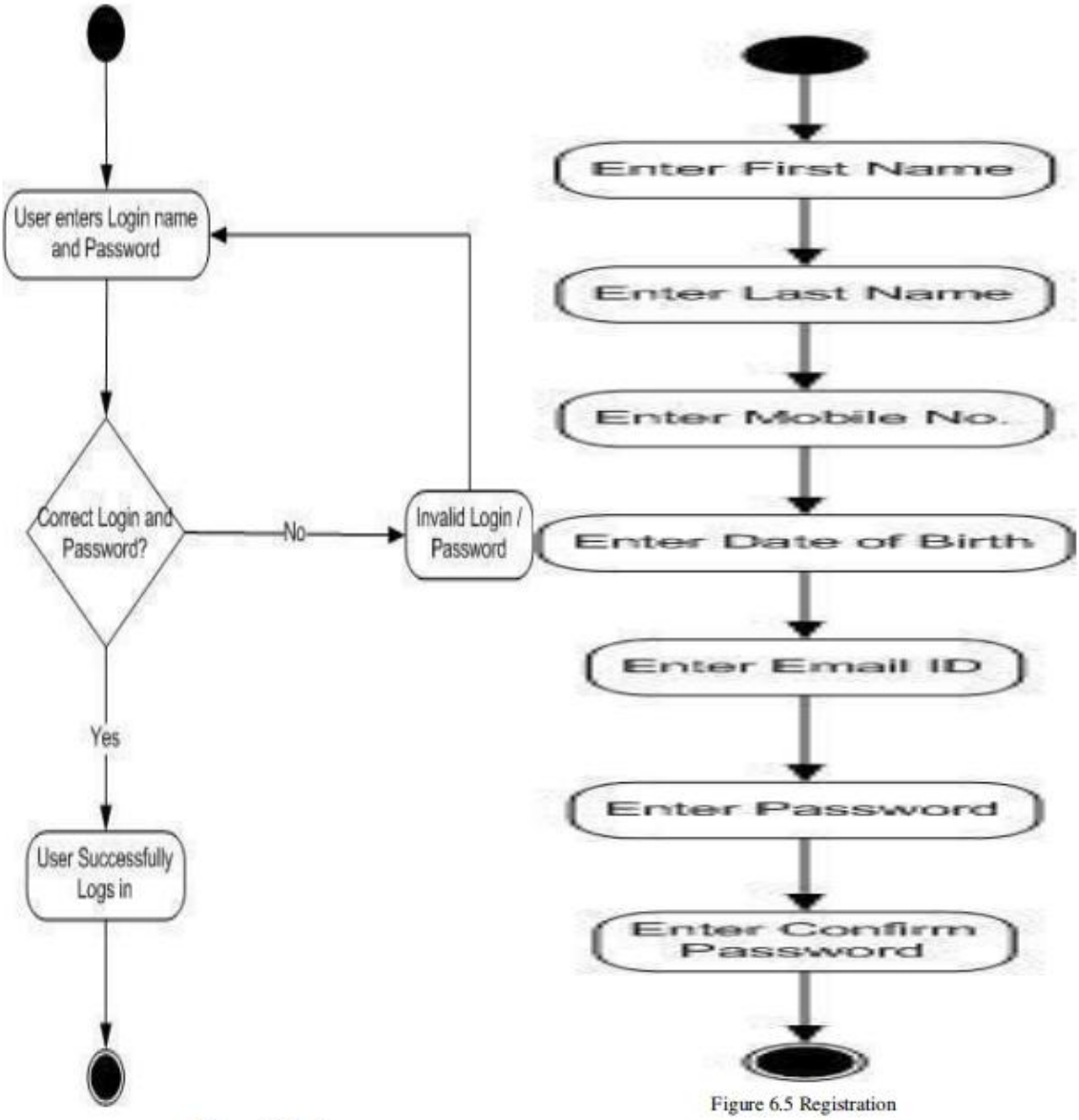
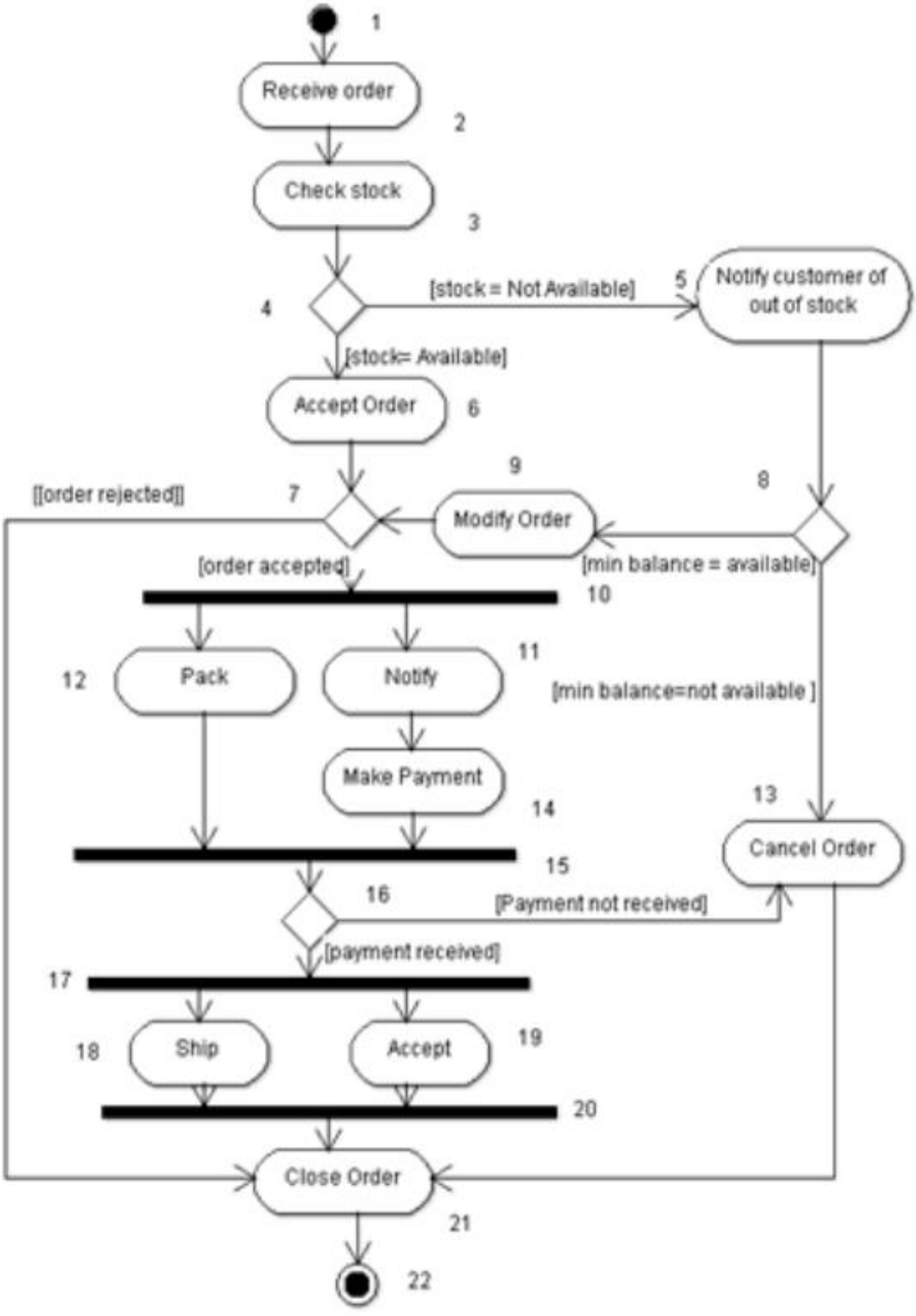
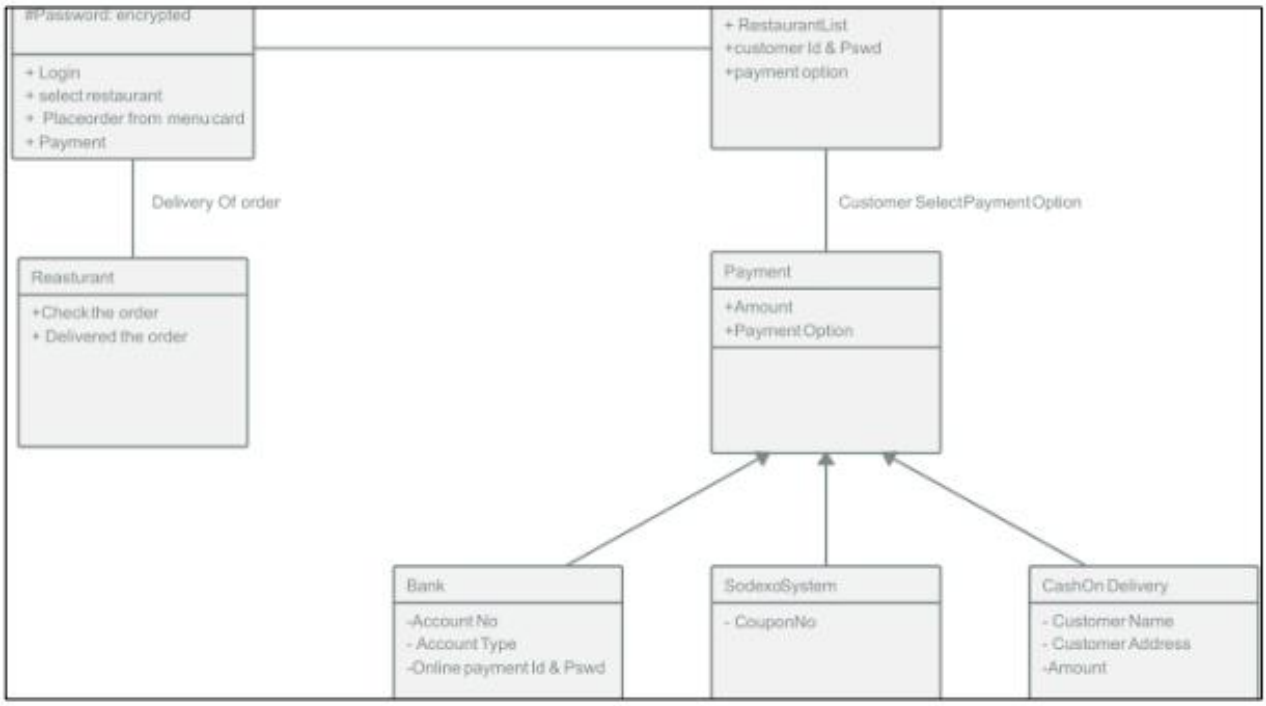


Figure 6.5 Registration

6.5. State chart Diagram:



6.6. Object Diagram:



7. Implementation Planning

7.1. Implementation Environment:

Implementations environment is in Java and android software is made in VS code IDE. VS code and
android studio provides a user friendly environment for project development.

7.2. Program/Modules Specification:

The Food Delivery software is aimed towards the customers that want to computerize their management system. It is user-friendly and easy to learn.
Functions:
• Login for authorized users
• Entry of customer details
• Manage the billing including usage of other services of the order

.

7.2.1. Security Features:

There is one module for administrator login. First of all administrator has to login to software with
his/her user-id and password

7.2.2. Coding Standards

The system is developed in Java which is an Object-Oriented language. Proper validation has been done for various inputs and outputs of the system. Care has been taken about proper indentation of the code

8. Testing

8.1. Testing Plan

First inspection is performed followed by white box testing which is applied by the programmer.

8.2. Testing Strategy

If it is white box testing then all statements are checked whether they are logically correct.

8.3. Testing Methods

Two methods are used: Boundary value checking, Equivalence class partitioning

8.4. Test Cases

Test Case 1: Testing login module

Expected response:

➢ Error message should be displayed if entered USERID and/or password is not found in database of administrator.

System response:

➢ "Error: No match for USERID and/or password." Test Case 2: Testing Check in module "CHECK IN"

Expected Response:

➢ Error message should be displayed if room is not available for requested period.

System Response:

➢ "The following errors were found:
➢ Items is not available for requested period."

8.4.1. Purpose

Checking whether the performance is up to the mark is very important.

8.4.2. Required Input

First condition to check is to provide valid input which should give valid output

8.4.3. Expected Result

Through expected results only we can check whether the output is correct.

9. Limitation and Future Enhancement:

Limitation: The limitation of our present software is that this is desktop based software and order
placing and billing is computer generated so there might occur system errors.

Future Enhancement: We are going to make online web based application for this system so
customer can automatically find menu and book their orders in advance.

10. Conclusion and Discussion:

Self-Analysis of Project Viabilities

Review the software often, to check how it's working.

Problem Encountered and Possible Solutions

If any problem occurs, then solve it.

Summary of Project work

Planning and design of code is very important while building large software.