**E-Mandi**

**Software Requirements Specification**

**Members:** (M.Lokesh Babu, Vaibhav Agarwal)

**College Name**: Keshav Memorial Institute of Technology(KMIT)

**Department**: Information Technology

**State**: Andhra Pradesh

**Our Plan:**

* Registration for users.
* Online maintenance of prices of vegetables on daily basis.
* Online booking from people.

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <30-01-2013> | 1.0 | Synopsis | <KMIT -Wakebusters> |

**Table of Contents**

**Description**

**1.0 Introduction**

1.1 Purpose

1.2 Scope

1.3 Definition, Acronyms, and Abbreviations

1.4 Tools Used

1.5 References

1.6 Technologies to be used

1.7 Overview

**Page No.**

**2.0 Overall Description**

2.1 Product Perspective

2.2 Software Interface

2.3 Hardware Interface

2.4 Communication Interface

2.5 Constraints

2.6 ER Diagram

2.7 Use Case Model Description

2.8 Architecture Design

2.9 Database Design

**3.0 Specific Requirements**

3.1 Use Case Reports

3.2 Activity Diagrams

3.3 Sequence Diagrams

3.4 Supplementary Requirements

**Software Requirements Specification**

1. **Introduction:**

It is a electronic vegetable market making the vegetable market more convenient for the use of civilian and even to keep the transparency in the whole market system from retailer to the whole seller.

**1.1 Purpose:**

The main objective of this project is to build a website which will help civilian, retailer, whole seller and even the framer to get the best from his inputs. With the help of this a farmer will be able to know the best value for his vegetable and will not be fooled by the marketers.

It will help in keeping the transparency between the whole seller and retailer and also the selection for civilian for his requirement become easy. So this will help in eradicating black marketing and inflation.

* 1. **Scope:**

The Scope of the E-Mandi includes:

* There are six different kinds of users – **Admin, Farmer, Whole Seller, Retailer,** **Civilian, Computer Professional.**
* All the users can have their own profile.
* People can search for different bazaars and enquire the prices of vegetables.
* People can register in the site and can avail online transaction for selling and buying vegetables in bulks.
* Feedback or complaint facilities can also be used by people.
* Website will be available in regional languages.
* People have facility of booking vegetables for commercial use on a large scale online through bigger marketers.
* Help section for those who are unable to understand the website or any of its part.

**1.3 Definitions, Acronyms, and Abbreviations:**

* HTML (Hyper Text Markup Language): It is used to create static web pages.
* JSP (Java Server Pages): It is used to create dynamic web content.
* J2EE (Java 2 Enterprise Edition): It is a programming platform, belonging to the Java platform, which is used for developing and running distributed java applications.
* WASCE (WebSphere Application Server Community Edition): It is an application server that runs and supports the J2EE and the web service applications.
* WSAD (WebSphere Studio Application Developer ): It is a designer toolkit which is designed to develop more complex projects by providing a complete dynamic web service.
* DB2 (IBM Database 2): It is a database management system that provides a flexible and efficient database platform to raise a strong "on demand" business applications.
* HTTP (Hyper Text Transfer Protocol): It is a transaction oriented client/ server protocol between a web browser and a web server.
* XML (Extensible Markup Language): It is a markup language that was designed to transport and store data.
* Ajax (Asynchronous Java Script and XML): It is a technique used in java script to create dynamic web pages.
* Web 2.0: It is commonly associated with web applications which facilitate interactive information sharing, interoperability, user-centered design and collaboration on the World Wide Web.

**1.4 Tools Used**

Application architecture – JAVA, J2EE

**JAVA**

Java is an object-oriented programming language developed by Sun Microsystems a company best known for its high end UNIX workstations. Java language was designed to be small, simple, and portable across platforms, operating systems, both at the source and at the binary level, which means that Java programs (applet and application) can run on any machine that has the Java virtual machine (JVM) installed.

**J2EE**

**Java Platform, Enterprise Edition** or **Java EE** is a widely used platform for server programming in the Java programming language. The Java platform (Enterprise Edition) differs from the Java Standard Edition Platform (Java SE) in that it adds libraries which provide functionality to deploy fault-tolerant, distributed, multi-tier Java software, based largely on modular components running on an application server.

**Web server – WASCE**

**WebSphere Application Server Community Edition** (from now on WASCE) is a free, certified Java EE 5 server for building and managing Java applications. It is IBM's supported distribution of Apache Geronimo that uses Tomcat for servlet container and Axis 2 for web services. Over 15

WASCE developers are committers in the Apache Geronimo project.

**Development tool –RAD**

**IBM Rational Application Developer for WebSphere Software (RAD)** is an integrated development environment (IDE), made by IBM's Rational Software division, for visually designing, constructing, testing, and deploying Web services, portals, and Java (J2EE) applications.

**Database platform – DB2**

**DB2 Database** is the database management system that delivers a flexible and cost effective

database platform to build robust on demand business applications and supports the J2EE and web services standards.

**Design tool – Rational Software Modeler**

**IBM Rational Software Modeler**, (RSM) made by IBM's Rational Software division, is a Uni- fied Modeling Language UML 2.0-based visual modeling and design tool. Rational Software Modeler is built on the Eclipse open-source software framework and includes

capabilities focused on visual modeling and model-driven development (MDD) with the UML for creating resilient, thought-out applications and web services.

**1.5 References**

* Object Oriented Modeling and Design with UML-Michael Blaha, James Rambaugh.
* Software Engineering, Seventh Edition, Ian Sommerville.
* Java - [www.sun.com](http://www.sun.com)
* Wikipedia - [*www.wikipedia.com*](http://www.wikipedia.com)
* Database Management Systems - Navathe.
* Complete Reference - J2EE – Keogh

**1.6 Technologies to be used**

* DB2: Relational Database Management System.
* RAD: Rational Application Developer.
* WASCE: Websphere Application Server Community Edition.
* Rational Software Modeler.
  1. Overview:

**Existing System:**

* Registration for users

**Drawbacks:**

* No web camera interaction
* Remote area users and people who doesn't have knowledge of internet cannot use the system

**Proposed System:**

* Registration for farmers, whole sellers
* Computer professional - help the farmers who doesn't have knowledge of internet to use the system.

**2. Overall Description:**

1. Product Perspective:

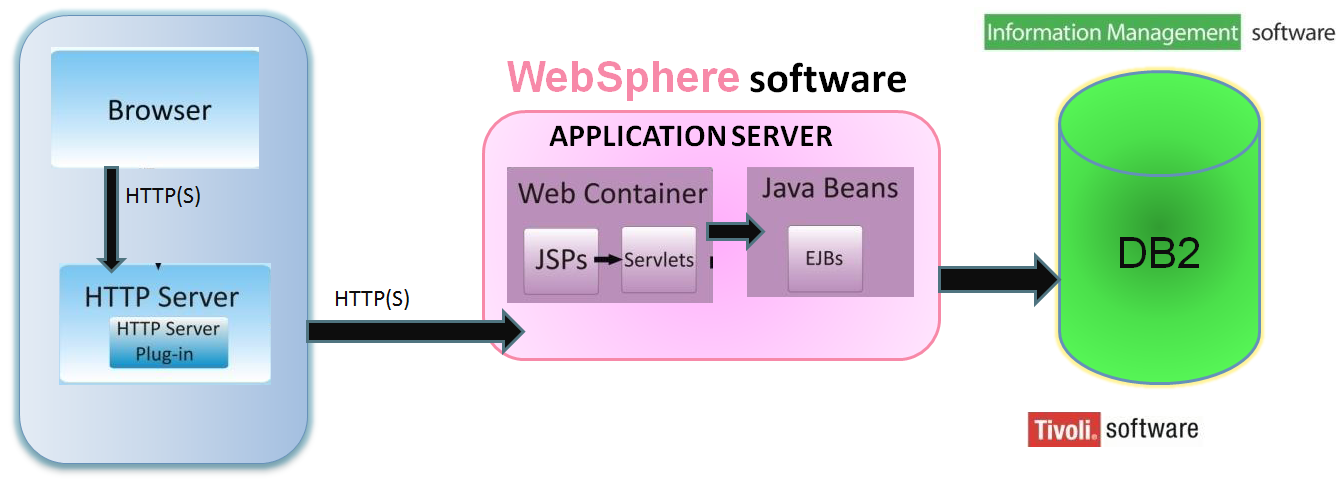


Fig 2.1: Product Perspective

2.2 SoftwareInterface

**Client on Internet**

Web Browser, Operating System (any)

**Client on Intranet** Web Browser, Operating System (any) **Web Server**

WASCE, Operating System (any)

**Data Base Server**

DB2, Operating System (any)

**Development End**

RAD (J2EE, Java, Java Bean, Servlets, HTML, XML, AJAX), DB2, OS (Windows), WebSphere(Web Server)

2.3 Hardware Interface

**Mini mum Requi rements:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Client Side** | | | |
|  | **Processor** | **RAM** | **Disk Space** |
| Internet Explorer - 6 | Intel Pentium III or AMD -  800 MHz | 128 MB | 100 MB |

|  |  |  |  |
| --- | --- | --- | --- |
| **Server Side** | | | |
|  | **Processor** | **RAM** | **Disk Space** |
| RAD | Intel Pentium III or AMD -  800 MHz | 1 GB | 3.5 GB |
| DB2 - 9.5 | 256 MB | 500 MB  (Excluding Data  Size) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Server Side** | | | |
|  | **Processor** | **RAM** | **Disk Space** |
| RAD | Intel Pentium III or AMD -  800 MHz | 1 GB | 3.5 GB |
| DB2 - 9.5 | 256 MB | 500 MB  (Excluding Data  Size) |

2.4 Communication Interface

* Client (customer) on Internet will be using HTTP/HTTPS protocol.
* Client (system user) on Internet will be using HTTP/HTTPS protocol.

2.5 Constraints

* Login and password is used for the identification of users.
* Only registered users and civilians will be authorized to use the services.

2.6ER Diagram

2.7 Architecture Design:

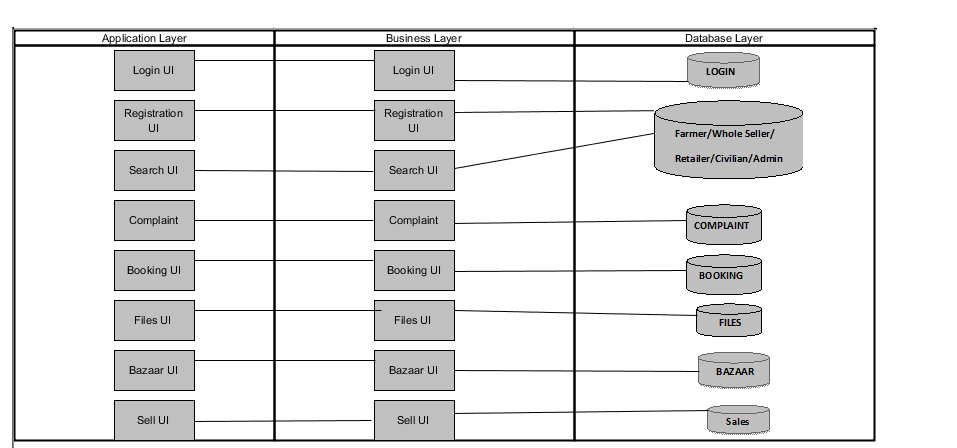


Fig: Architecture Diagram

2.8 Use Case Model Discription:

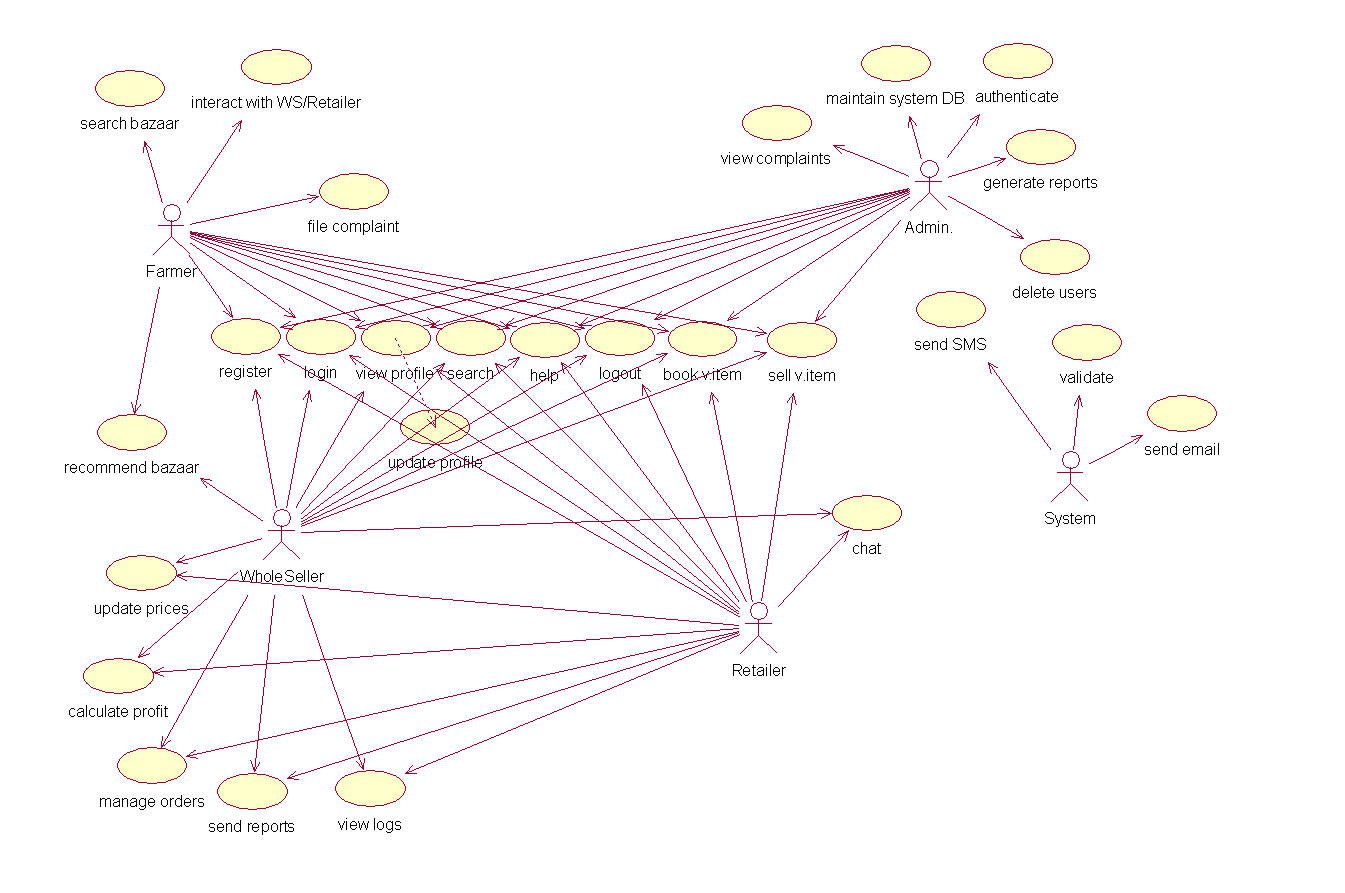


Fig: Use Case Model Discription

**Farmer:**

Farmer can search for bazaar and make online appointments. They also can view their profile, update their profile. They have the provision of selling and buying vegetables. Farmer can also register, complaint on any WS/Retailer.

**Whole Seller or Retailer:**

Along with the farmer privileges a Whole Seller or Retailer can recommend bazaar to a farmer. He/she can update the prices of his/her vegetables and calculate the profit accordingly. He can manage orders so that he/she can sell the vegetables to only those whom they consider. He/she can send reports of feedback to farmer for future transactions and to develop relationships. He/she can chat directly with administrator for suggestions or in report of a complaint.

**Admin:**

Admin has the authority to add/delete users, grant permission to WS and Retailer, to generate and view reports. He also views the complaints of farmers, civilians and takes necessary actions. He maintains the database, authenticates Whole Sellers and Retailers and ensures that the data is consistent.

**System:**

Takes the responsibility of sending sms and email to the users when registered and involving in any transaction related to money. It also checks for the validity of users.

2.9 Database Design:



Fig: Database Design

**Specific Requirements**

3.1 Use Case Reports:

**Farmer use-case report:**



Fig: Use case diagram for farmer

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| **Login** | The farmer has to login in order to do following things. |
| **View profile** | Every registered farmer has his/her own profile containing personal details. |
| **Update profile** | The farmer has the option to update his/her own profile. |
| **Select WS/Retailer** | The farmer can select WS/Retailer based on various criteria. |
| **File complaint** | The farmer can file complaint on Whole Seller or Retailer to admin. |
| **Send Message** | The farmer can send private message to Whole Seller and Retailer. |
| **Receive mail** | The farmer can receive mail. |
| **Search** | The farmer can search for a particular item in website by entering the keyword. |
| **Sell** | The farmer can sell vegetables in bulk to his requested Whole Seller or Retailer. |
| Book | The farmer can book for vegetables in bulk from selected Whole Seller or Retailer. |

**Whole Seller use-case report:**

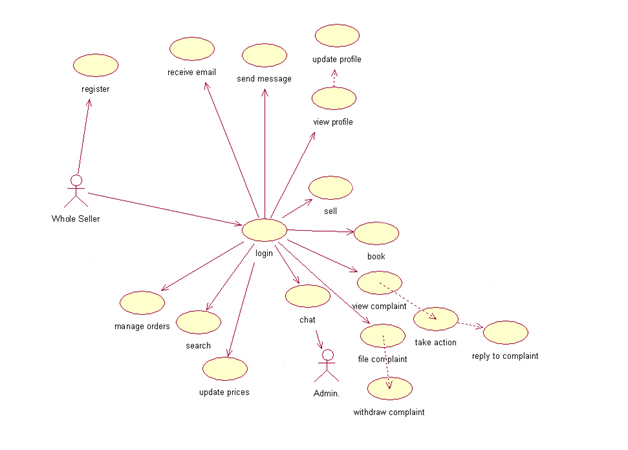


Fig: Use case diagram for Whole Seller

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| **Login** | The WS has to Login in order to start begin his work. |
| **View profile** | Every registered WS has his/her own profile containing personal and professional de-  tails. |
| **Update profile** | The WS has the option to update his/her own profile. |
| **Update prices** | The WS can update the prices of his/her vegetables from time to time. |
| **Chat** | The WS can chat with admin. in order to keep track of his fees/commission. |
| **Acknowledge a request (manage orders)** | The WS accept/rejects the request of farmer for vegetables. |
| **Calculate profits** | The WS can calculate profits on the prices of his/her vegetables. |
| **File complaint** | The WS can file complaint on other Whole Seller or Retailer to admin. |
| **View complaints** | The WS views the feedback of the admin to the complaints issued to a WS. |
| **Sell** | The WS can sell vegetables in bulk to his requested Whole Seller or Retailer. |
| Book | The WS can book for vegetables in bulk from selected Whole Seller or Retailer. |
| View logs | The WS can view his/her logs . |
| **Send message** | The WS can send private messages to farmers and admin. |
| **Receive mail** | The WS can receive mail. |
| **Search** | The WS can search for a particular item in website by entering the keyword. |

**Retailer use-case report:**

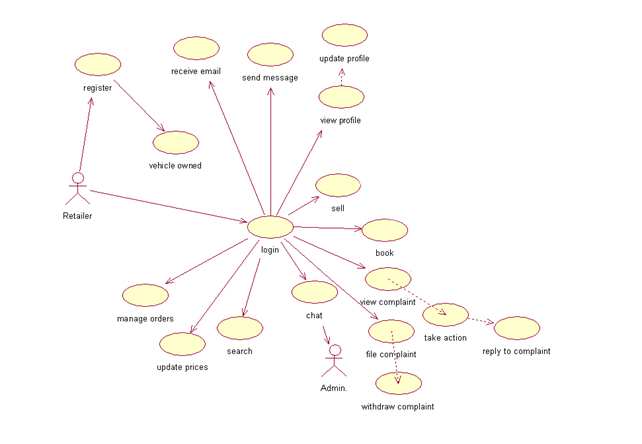


Fig: Use case diagram for Retailer

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| **Login** | The Retailer has to Login in order to start begin his work. |
| **View profile** | Every registered Retailer has his/her own profile containing personal and professional de-  tails. |
| **Update profile** | The Retailer has the option to update his/her own profile. |
| **Update prices** | The Retailer can update the prices of his/her vegetables from time to time. |
| **Chat** | The Retailer can chat with admin. in order to keep track of his fees/commission. |
| **Acknowledge a request (manage orders)** | The Retailer accept/rejects the request of farmer for vegetables. |
| **Calculate profits** | The Retailer can calculate profits on the prices of his/her vegetables. |
| **File complaint** | The Retailer can file complaint on other Whole Seller or Retailer to admin. |
| **View complaints** | The Retailer views the feedback of the admin to the complaints issued to a WS. |
| **Sell** | The Retailer can sell vegetables in bulk to his requested Whole Seller or Retailer. |
| Book | The Retailer can book for vegetables in bulk from selected Whole Seller or Retailer. |
| View logs | The Retailer can view his/her logs . |
| **Send message** | The Retailer can send private messages to farmers and admin. |
| **Receive mail** | The Retailer can receive mail. |
| **Search** | The Retailer can search for a particular item in website by entering the keyword. |

**Admin use-case report:**

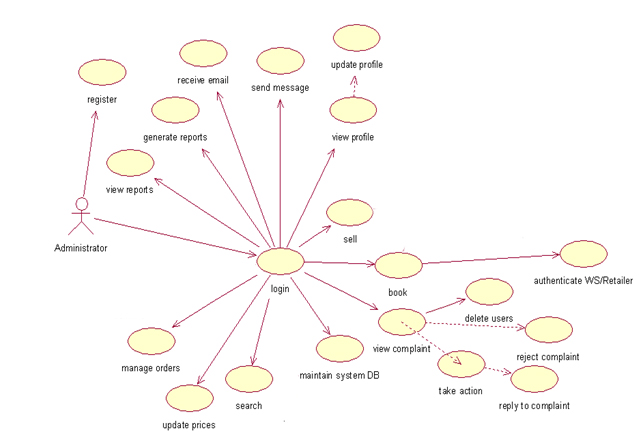


Fig: Use case diagram for Admin

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| **Login** | The admin has to Login in order to start begin his work. |
| **View profile** | Admin has his/her own profile containing personal and professional details. |
| **Update profile** | The admin has the option to update his/her own profile. |
| **View logs and reports** | The admin can view logs and reports sent by WS/Retailer. |
| **Generate reports** | The admin can generate reports. |
| **Authenticate WS/Retailer** | The admin will authenticate WS and Retailer. |
| **View complaint** | The admin views the complaint sent by other users. |
| **Take action** | The admin can take necessary actions . |
| **Take backup(maintain system DB)** | The admin can take backup of the system and maintain database. |
| **Delete users** | The admin is given the option to delete the users . |
| **Send mail** | The admin can send mail. |
| **Receive message** | The admin can receive private messages from WS and Retailers. |
| **Reject complaint** | The admin can reject any complaint sent by farmer, Ws and Retailer. |
| **Search** | The admin can search for a particular item in website by entering the keyword. |
| **Sell** | The admin can sell vegetables in bulk to his requested Whole Seller or Retailer. |
| Book | The admin can book for vegetables in bulk from selected Whole Seller or Retailer.   |  |  |  |  |  | | --- | --- | --- | --- | --- | |  |  |  |  |  | |  |  |  |  |  | |
| **Update prices** | The admin can update the prices of his/her vegetables from time to time. |
| **Calculate fees/commission** | The admin can calculate fees/commission for a particular WS and Retailer. |

**System use-case report:**

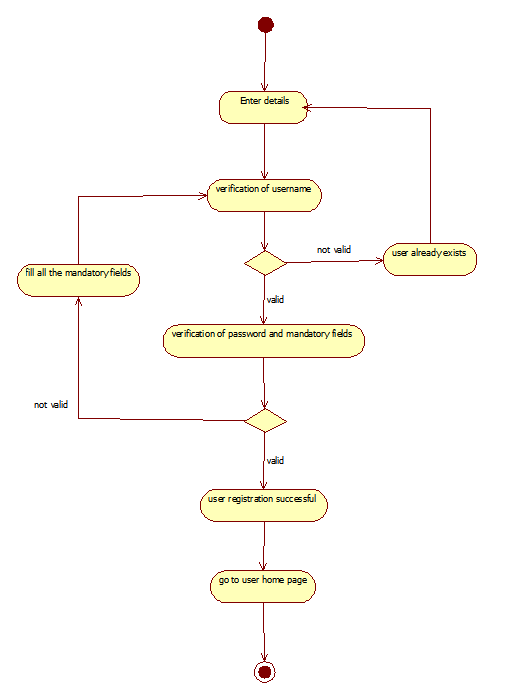
****

Fig: Use case diagram for System

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| **Send sms** | The system can send sms to those who got registered, booked or sold vegetable as  Acknowledgement and subscribed for price updates. |
| **Validate** | The system can validate the registration process there by detecting fraud WS or Retailers. |
| **Send email** | The system can send mail to those who got registered, booked or sold vegetable as  Acknowledgement and subscribed for price updates. |

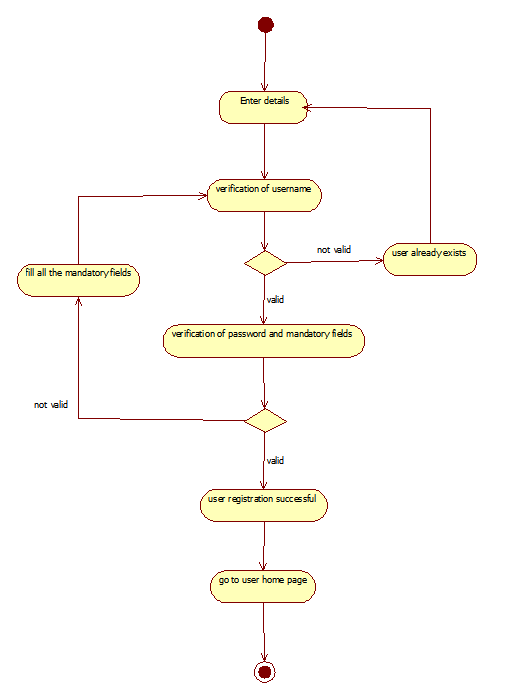
3.2 Activity Diagrams

**User Registration Activity**

Initially user is made to fill all mandatory fields filled in registration form. Once the user clicks submit, the username is verified. If the username is already present, then the user is again taken back, so that he can change the username. If the username is not present then it checks for password and remaining mandatory fields. If any of the mandatory field is left empty or filled incorrect, then the user is informed to enter the correct values. Once all these verifications are succeeded, then the registration is done.**** Fig: Activity Diagram Representing User Registration

**Whole Seller and Retailer Registration Activity:**

The Whole Seller and Retailer needs to fill all mandatory fields filled in registration form. Once they click submit, the username is verified. If the username is already present, then they are again taken back, so that they can change the username. If the username is present then it checks for password and remaining mandatory fields. If any of the mandatory field is left empty or filled incorrect, then they are informed to enter the correct values. These details are sent to admin for verification and the admin verifies the data and approves registration and grants privileges.

Fig: Activity Diagram Representing WS/Retailer Registration

**User Login Activity**

User is made to enter the username and password and then entered values are verified. If it is a valid username and password, then the user is logged in, or else they are asked to re enter the correct values.

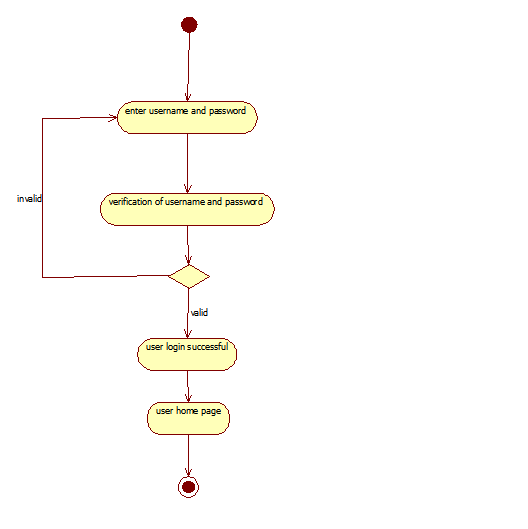


Fig: Activity Diagram Representing User Login

**Lodging a Complaint Activity:**

The user enters the complaint text and sends it to admin and the system generated Complaint- ID is returned to user.

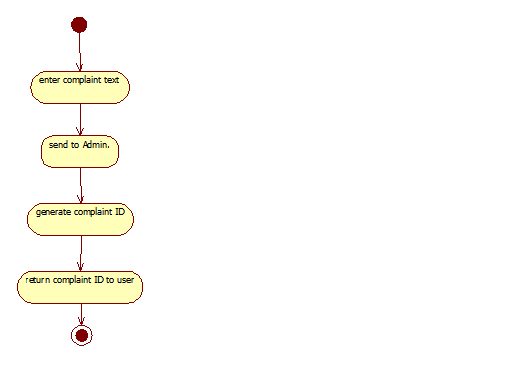
****

Fig: Activity Diagram for Lodging Complaint

**Replying a Complaint Activity:**

The admin views the complaint and then he/she may reply to the user or he/she may select the respective Whole Seller or Retailer and forwards the complaint to him/her.

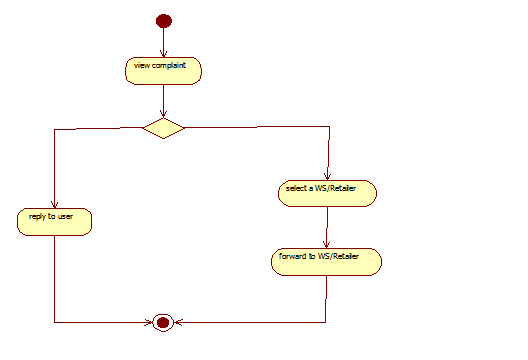


Fig: Activity Diagram for Representing Reply To Complaint

**Selling a vegetable Activity:**

The user can sell a vegetable. After opening the sell vegetable page he/she has to select a vegetable, amount of vegetable i.e., the quantity and the unit which determines the way vegetable is arranged for disposal. Further he/she has to select a WS/Retailer by specifying their ID. This makes his/her job of selling vegetable complete.

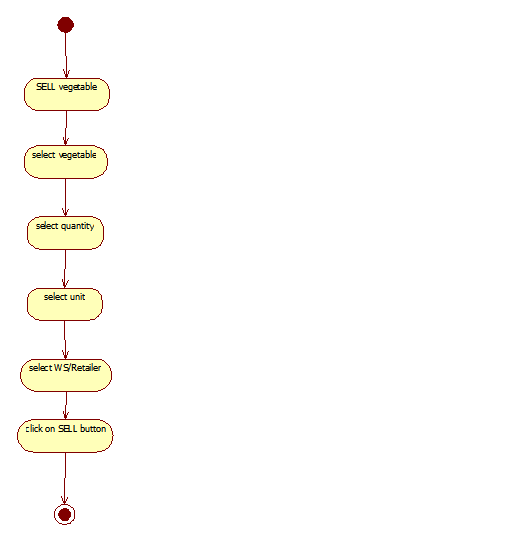
****

Fig: Activity Diagram for Representing Selling a vegetable

**Booking a vegetable Activity**

The user can book a vegetable. After opening the book vegetable page he/she has to select a vegetable, amount of vegetable i.e., the quantity and the unit which determines the way vegetable is arranged for disposal. Further he/she has to select a WS/Retailer by specifying their ID. This makes his/her job of booking vegetable complete.

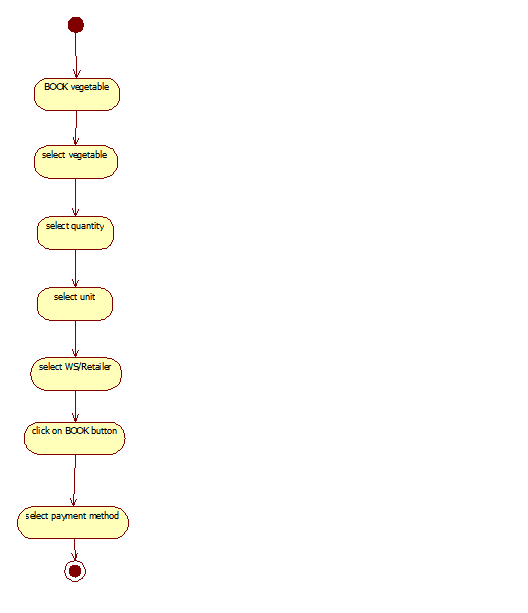


Fig: Activity Diagram for Representing Booking a vegetable

enter Booking ID

select from booked orders

click on CANCEL booking

Fig: Activity Diagram for Representing Cancel-Booking a vegetable

**Search nearest bazaar Activity:**

The user can find out the nearest bazaar of his/her address. He/she can also find the nearest bazaar related to location of his/her choice by selecting the locality.

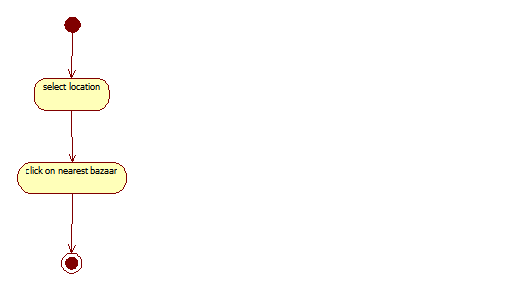
****

Fig: Activity Diagram for Representing searching nearest bazaar

**View price difference Activity**

The user can find the differences between two bazaars and compare the prices so that he/she can know the best price for the vegetable. The user has to select bazaar-1 and bazaar-2 and clicking on submit will serve his/her purpose.

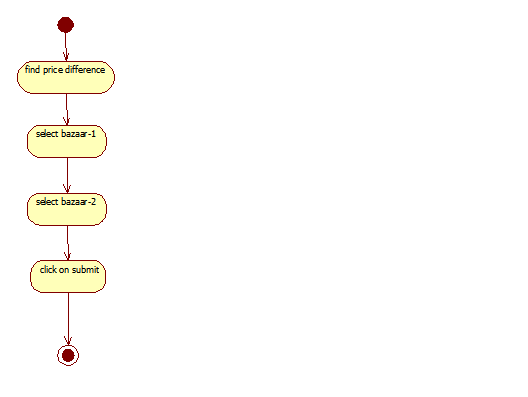
****

Fig: Activity Diagram Representing price-difference

**3.3 Sequence Diagrams**

**3.3.1 Users Registration Sequence Diagram**

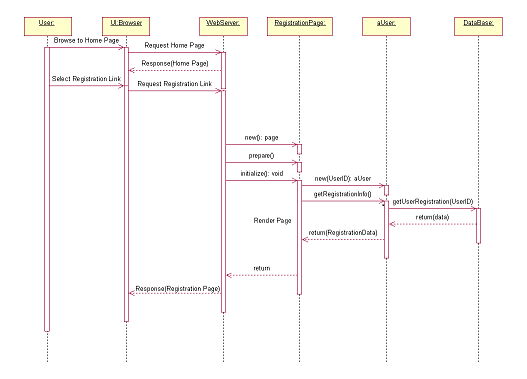


Fig 3.15: Sequence Diagram Representing Registration Process

**3.3.1 Users Login Sequence Diagram**

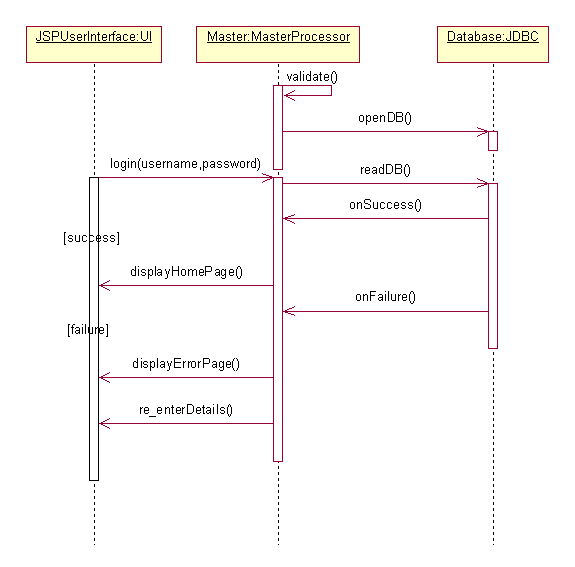


Fig 3.15: Sequence Diagram Representing Login Process

**3.3.1 Booking Sequence Diagram**

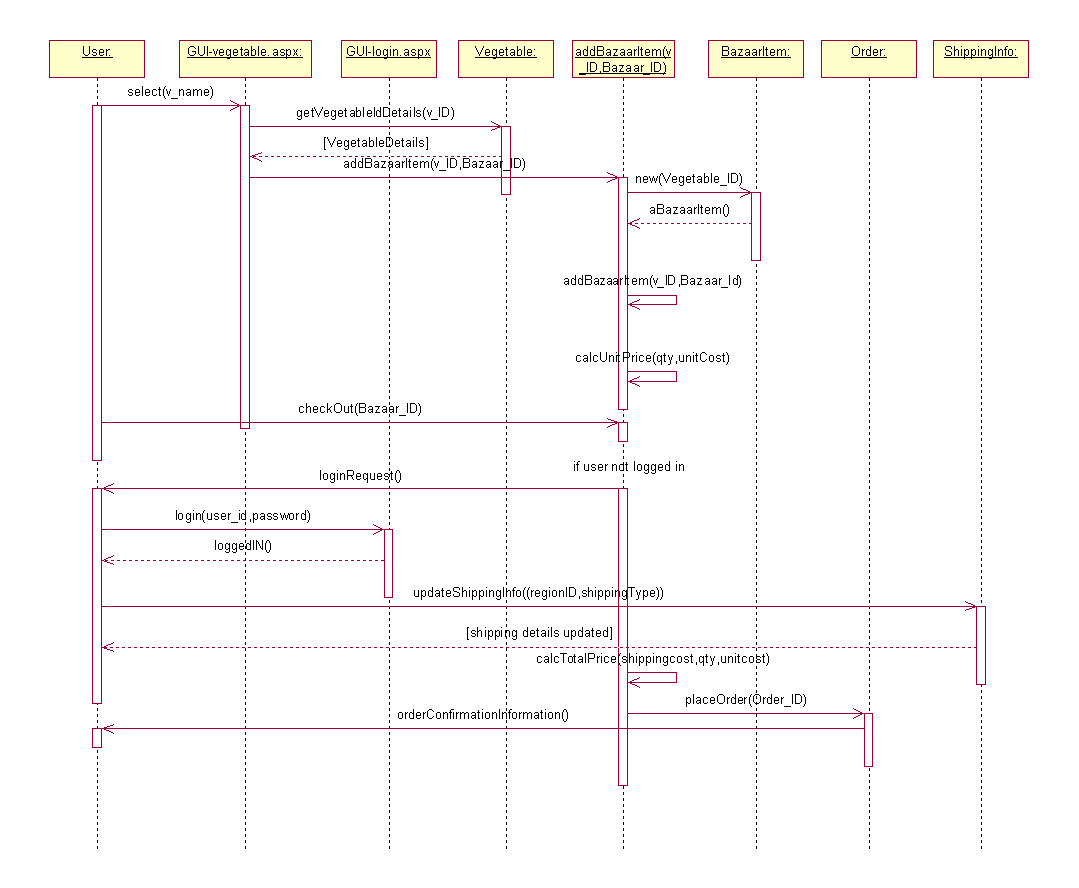


Fig 3.15: Sequence Diagram Representing Booking Process

**3.3.1 Sales Sequence Diagram**

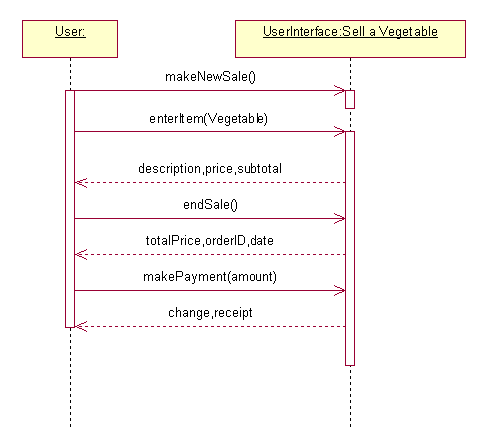


Fig 3.15: Sequence Diagram Representing Sales Process

**3.3.1 Search Sequence Diagram**

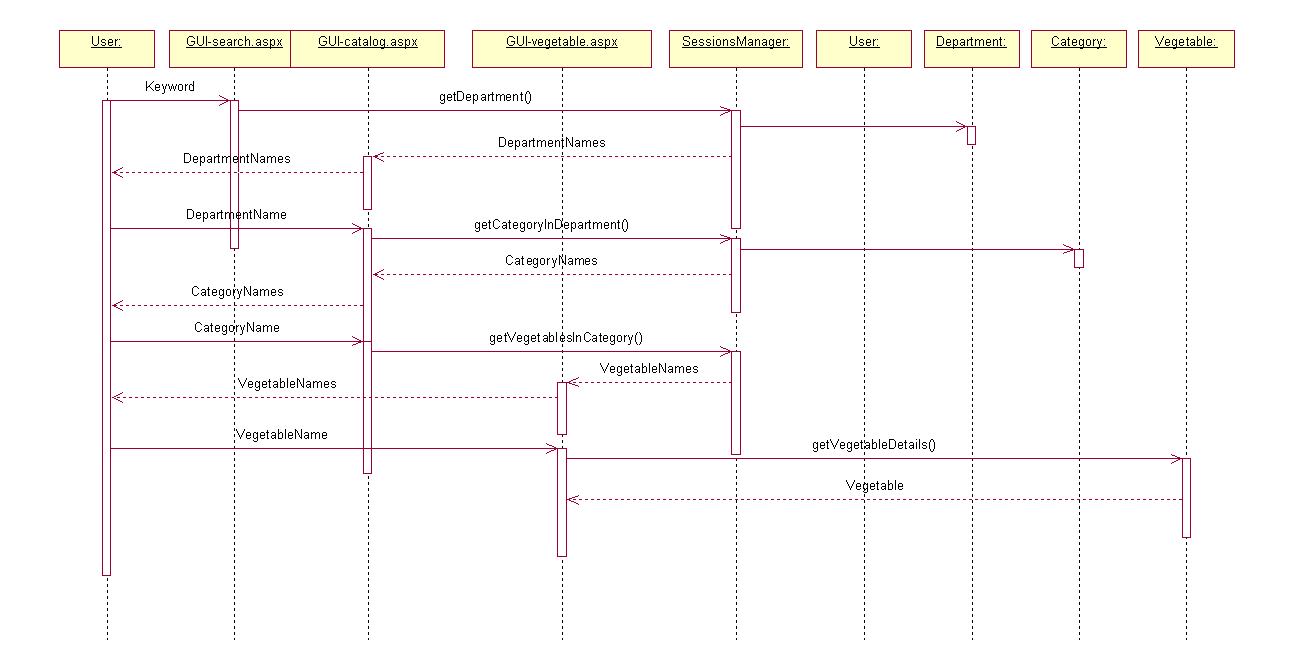
****

Fig 3.15: Sequence Diagram Representing Search Process

**3.3.1 Complaint Sequence Diagram**

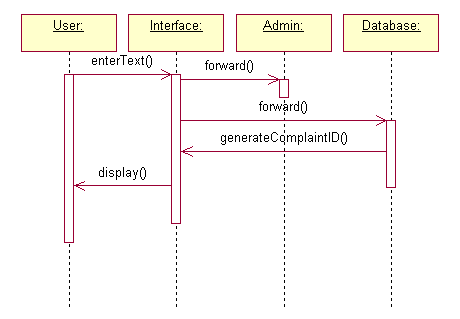


Fig 3.15: Sequence Diagram Representing Complaint Process

**4.Supporting Information**

4.0Index

A

Abbreviations · 5

Acronyms · 5

Administrator · 16

AJAX · 7, 13

Analysis · 3 Design · 18

Address

Area

Authenticate

Action

B

Buy

Bazaar

Book

Backup

C

Client on Internet · 8

Client on Intranet · 8

Communication · 11

Complaints · 5

Constraints · 11

Construction · 4

Civilian

Contact

Calculate

Commission

Chat

Cancel

D

Data Base · 15

DB2 · 5, 6

Definitions · 5

Development End · 8

Drawbacks · 8

DOB

Delete

Difference

E

Elaboration · 4

ER Diagram · 12

Existing System · 8

E-Mandi

Experience

E-mail

Established

F

Farmer

Feedback

Fees

File

G

Gender

Generate

H

Hardware · 10

HTML · 8

HTTP · 5,8

HTTPS · 8

Help

I

Inception · 3

Interface · 9,10,11

Introduction · 4

Item

Interact

J

J2EE · 5, 13

Java · 6

JSP · 5,6

L

Login

Logout

Logs

List

M

Methodology · 4

Message

Manage

N

Name

O

Operating System · 6, 8

Our Plan· 8

Overall · 9

Overview · 8

Order

P

Prescription · 4

Product Perspective · 9

Proposed System · 8

Purpose · 4

Price

Password

Profile

Profit

Q

Quantity

Qualification

R

RAD · 6

References · 7

Reports · 4

Rational Unified Process · 3

Retailer

Rating

Registration

Reply

Reject

Recommend

Requirement

S

Scope · 4

Sequence · 30

Software · 9

Sell

SMS

Search

Select

System

Send

T

Transition· 4

Technologies · 7

U

Use-case · 13, 16

UML · 5

Unit

Username

User

Update

V

Vegetable

View

Vehicle

Validate

W

WASCE · 5,

Whole Seller

Withdraw