A
Project Report
on
**Integrated Supply Chain Solutions All Under One
Roof**

Developed at

Samyak Infotech Pvt. Ltd
905-908, Abhijeet 1 Mithakhali Six Roads Elish Bridge,
Ahmedabad, Gujarat, India 380006

Developed by

Raval Kruti – Department of IT, DD University

Guided By

Internal Guide :   **External Guide:**
Prof. Sandeep R. Suthar   **Mr. Sandeep Das**
Department of Information Technology   **Project Manager**
Faculty of Technology   **Samyak Infotech**
DD University   **Pvt. Ltd**

Department of Information Technology
Faculty of Technology, Dharmsinh Desai University
College Road, Nadiad-387001
April-2017

Table of Contents

ABSTRACT

The purpose of this document is to capture, in natural language and at a functional level, the description and requirements of Integrated Supply Chain Solutions for Network Global Logistics(NGL), a premier provider of end to end supply chain solutions to the Aerospace, eCommerce and Retail, Food and Grocery, High Tech, Industrial and Automotive, Life-sciences and Healthcare, Medical Equipment, Telecom and Entertainment. The focus here is to develop a courier management system. This is a functional description of those features required to address current courier management requirements. A short discussion accompanies each requirement, to add the background and framework necessary to explain the functionality. It also describes nonfunctional requirements and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

This document is meant for users, developers, project managers, testers and documentation writers. The document aims to explain in an easy manner, the basic Idea behind the Integrated Supply Chain Solution System and how the developers aim to achieve their goals. It also aims to introduce to the users the main features of the Integrated Supply Chain Solution Management System where different functionalities can be edited or added.

This system will be an internal courier management system for NetLink wishing to manage their supply chain management needs with the help of Electronic Data Interchange(EDI). More specifically to design and develop a simple and intuitive system which shall cater the various needs such as customer management, order tracking, inventory management, report generation, billing, rate calculation, etc. The system shall provide features to the agent based on the customer/client roles like frequent caller, first time caller, call back and others to manage, view and insert the required information.

Supply Chain Management System is a web based system which aims to provide a host of features including Electronic Data Interchange(EDI). Thus, our project is a self contained database project that works on any operating system with a modern web browser.

COMPANY PROFILE

Samyak Infotech Pvt. Ltd

Samyak Infotech Private Limited is an IT company based in Ahmedabad, India specializing in providing Customized Integrated Software Solutions. Imagined, fashioned and accomplished by a group of self-motivated IT professionals with ability to innovate and a strong desire to excel, Samyak Infotech Pvt Ltd is a rapidly growing IT company with extensive experience in designing and developing cutting edge IT solutions for clients representing a broad cross-section of businesses.

What makes Samyak Infotech Distinguishing?

Samyak Infotech has been pretty distinguishing in comparison to its rivals due to its ability to develop user-friendly solutions for complex IT requirements. Be it about the productivity or efficacy, we have always been emphasizing on building the right architecture for the product

This is the reason that our applications have been the best in terms of installation and recycle. Working with Fortune 100 clients, we have managed to be able in understanding the right ways of addressing the comparatively challenging demands.

- Hold the specialization in developing
- The multitier architecture applications
- Application with simplest UI for service processing
- Applications for the budget-friendly devices

Services:

Samyak Infotech Provide range of software solutions and services meeting national & international standards. There are various Services in various technologies.

- Software Development
- Frontend Coding
- Design
- Mobile application

Portfolio:

Samyak is a leading Web Design and development company, build website using wide areas of technologies Microsoft Technologies, Open Source, Java, Responsive website layout.

It has 3 categories for portfolio. 1) Web Application 2) Desktop Application 3) Mobile Application In which we have following projects Library Management System, Component Management and Information System , Terminal Automation Software , Enterprise Application , Samyak Store , Asset Management System , Free SMS Software , Weighbridge Software , Document Management System (Online / Offline) , GPS Tracking System , Phone Number Extrator , Contact Manager , Sugar Mill , Word Translator , Math Formulas and Tricks etc.

Why Samyak?

Samyak Infotech is Established in 1998. Perfection, Completeness and Wholeness is always in action & attitude at Samyak. This is the way of life and atmosphere, in which we are working. Samyak has started operations in the year 1987 as Process Control Instruments Manufacturing Company. With continuous development and foresight, Samyak ventured into IT solutions provider company, in the year 1998. Over 15+ years of industrial presence, in spite of stiff competition, speaks for itself about the quality of service and expertise in providing most suitable solution to client needs. The execution of government-funded projects by Samyak speaks for its trust and quality assurance at state and national level. From hardware to Software, Embedded solutions to ERP. Samyak has proved itself a trustworthy companion by accepting challenging roles and executing them successfully.

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

CSR – Cooperate social responsibility
KSMS – Known Shipper Management System
FR8 - Freight
PA – Priority Air
FRG –FedEx heavy ground service
FRA – FedEx heavy Air Service
SDC – Same Day City
PR - Priority
ST - Standard
OD Pair – Zip code OA/OB/OC/OD Pair
NCS - Network Courier
DSA – Deliver same address
HAD – Hold at Delivery
DSR – Direct signature require
ASR - Adult Signature Require
H3P – Hold at Third party
CPU – Customer Pickup, its service
QPT -Quoted Pickup Time
QDT -Quoted Delivery Time
PU-SCH Pickup Schedule
DU-SCH Delivery Schedule
GUI -Graphical User Interface
HTML -Hypertext Markup Language
HTTP- Hypertext Transfer Protocol
IIS -Internet Information Services
JSON -JavaScript Object Notation
SQL -Structured Query Language
UI -User Interface
URL -Uniform Resource Locator
WWW -World Wide Web
XML- Extensible Markup Language
OS -Operating System

1.0   INTRODUCTION

1.1 PROJECT DETAILS

The purpose of this document is to capture, in natural language and at a functional level, the description and requirements of Integrated Supply Chain Solutions for Network Global Logistics (NGL), a premier provider of end to end supply chain solutions to the Aerospace, eCommerce and Retail, Food and Grocery, High Tech, Industrial and Automotive, Life-sciences and Healthcare, Medical Equipment, Telecom and Entertainment. The focus here is to develop a courier management system. This is a functional description of those features required to address current courier management requirements. A short discussion accompanies each requirement, to add the background and framework necessary to explain the functionality. It also describes nonfunctional requirements and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

1.2 PURPOSE

The main purpose of a global supply-chain network is to maximize a firm's competitive advantage. The typical activities performed by suppliers as part of a global supply-chain network are maintain inventory, transport goods and process orders.The basic purpose of logistics cost analysis is to provide managers with reliable information that will enable a better allocation of resources to be achieved. The purpose of logistics and supply chain management is concerned to meet customer service requirements in the most cost-effective way with the most accurate and meaningful data possible.

1.3 SCOPE

This system will be an internal courier management system for NetLink wishing to manage their supply chain management needs with the help of Electronic Data Interchange(EDI). More specifically to design and develop a simple and intuitive system which shall cater the various needs such as customer management, order tracking, inventory management, report generation, billing, rate calculation, etc. The system shall provide features to the agent based on the customer/client roles like frequent caller, first time caller, call back and others to manage, view and insert the required information..

Additional some unique features like Quote Generation and 3rd Party Courier Relations will be also included to cater specific needs of the SCM system. The Order details will also be mailed to the required mail id and live tracking is also available with this SCM system.

Thus the ideal goal of this system is to provide an easy Supply Chain Management from anywhere at any time.

1.4   OBJECTIVE

The ERP System is an integrated solution for the automation of various academic institutions. This ERP system is specifically for our college, aiming to replace completely current manual processing of an internal result management. Using technologies like Hibernate and JSP MVC for data and business logic layer while, JQuery and Bootstrap for front-end development.. Our software has different modules for different tasks and thus facilitate computerized process for result management, attendance management, faculty allocation, attendance and academic monitoring and many more.

Online Result Management system is a web based system which aims to provide a web application to college and its student to easily access and manage results. Thus, our project is a self contained database project that works on any operating system with a modern web browser.

1.5   TECHNOLOGY AND LITERATURE REVIEW

AngularJS:

AngularJS, commonly referred to as Angular, is an open-source web application framework maintained by Google and a community of individual developers and corporations to address many of the challenges encountered in developing single-page applications. Its goal is to simplify both development and testing of such applications by providing a framework for client-side model–view–controller (MVC) architecture, along with components commonly used in rich internet applications.
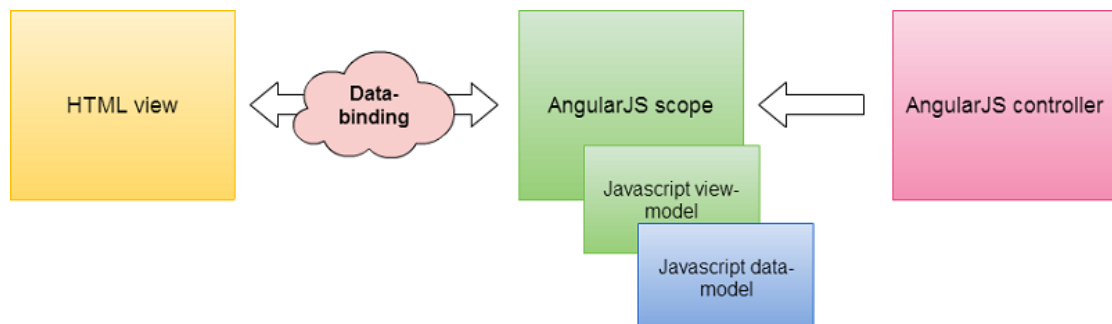
Figure 1.1   How AngularJS works (2)

The library works by first reading the HTML page, which has embedded into it additional custom tag attributes. Those attributes are interpreted as directives telling Angular to bind input or output parts of the page to a model that is represented by standard JavaScript variables. The values of those JavaScript variables can be manually set within the code, or retrieved from static or dynamic JSON resources.

AngularJS is built around the belief that declarative programming should be used for building user interfaces and connecting software components, while imperative programmingis better suited to defining an application's business logic. The framework adapts and extends traditional HTML to present dynamic content through two-way data-binding that allows for the automatic synchronization of models and

views. As a result, AngularJS de-emphasizes DOM manipulation with the goal of improving testability and performance.

AngularJS's design goals include:

- Decouple DOM manipulation from application logic. This difficulty is dramatically affected by the way the code is structured.
- Decouple the client side of an application from the server side. This allows development work to progress in parallel, and allows for reuse of both sides.
- Provide structure for the journey of building an application: from designing the UI, through writing the business logic, to testing.

Angular implements the MVC pattern to separate presentation, data, and logic components. Using dependency injection, Angular brings traditionally server-side services, such as view-dependent controllers, to client-side web applications. Consequently, much of the burden on the server can be reduced.

Scope:

Angular uses the term "Scope" to mean something different than what it usually means in computer science. In Angular, "scope" is a certain kind of object that itself can be in scope or out of scope in any given part of the program, following the usual rules of variable scope in JavaScript like any other object. When the term "scope" is used below, it means the Angular scope object and not the variable scope.

Bootstraper: The tasks performed by the AngularJS bootstrapper occur in three phases after the DOM has been loaded:

1. Creation of a new Injector
2. Compilation of the directives that decorate the DOM
3. Linking of all directives to scope

Two Way Data Binding:
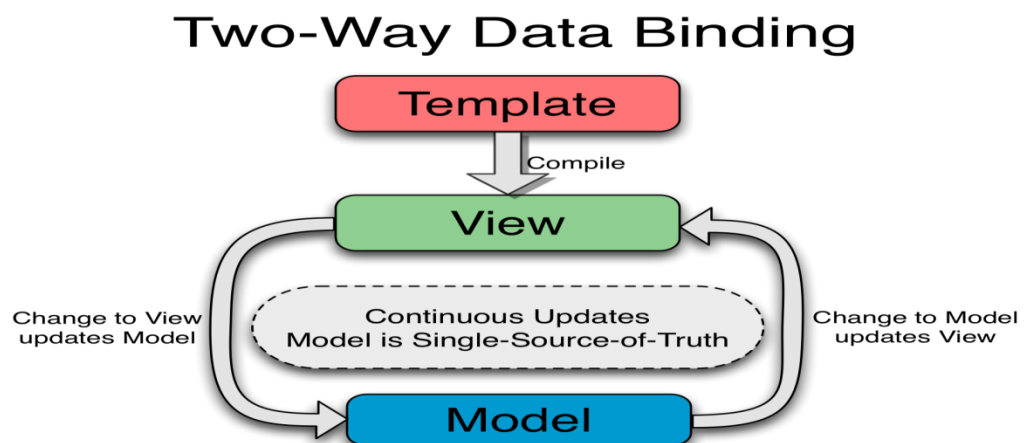
Figure 1.2   Two Way Data Binding in AngularJS

AngularJS' two-way data binding is its most notable feature, and it reduces the amount of code written by relieving the server backend of templating responsibilities. Instead, templates are rendered in plain HTML according to data contained in a scope

defined in the model. The $scope service in Angular detects changes to the model section and modifies HTML expressions in the view via a controller. Likewise, any alterations to the view are reflected in the model. This circumvents the need to actively manipulate the DOM and encourages bootstrapping and rapid prototyping of web applications. AngularJS detects changes in models by comparing the current values with values stored earlier in a process of dirty-checking.

WCF

Windows Communication Foundation (WCF) is a framework for building service-oriented applications. Using WCF, you can send data as asynchronous messages from one service endpoint to another. A service endpoint can be part of a continuously available service hosted by IIS, or it can be a service hosted in an application.

ASP.NET MVC

The Model-View-Controller (MVC) architectural pattern separates an application into three main components: the model, the view, and the controller. The ASP.NET MVC framework provides an alternative to the ASP.NET Web Forms pattern for creating Web applications. The ASP.NET MVC framework is a lightweight, highly testable presentation framework that (as with Web Forms-based applications) is integrated with existing ASP.NET features, such as master pages and membership-based authentication. The MVC framework is defined in the System.Web.Mvc assembly.

Bootstrap

Bootstrap is a free and open-source front-end web framework for designing websites and web applications. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions. Unlike many web frameworks, it concerns itself with front-end development only.

jQuery

jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML. jQuery's syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, themeable widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and Web applications.

Microsoft SQL Server

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which mayr un either on the same computer or on another computer across a network (including the Internet).

2.0   PROJECT MANAGEMENT

2.1   FEASIBILITY STUDY

2.1.1   Technical Feasibility

Amidst, technical analysis evaluates technical merits of the system at the same time collection additional information about performance, reliability, maintainability and productivity. In some cases, an analysis step also includes research and design.

The project being developed uses AngularJs Technology which are available freely on the internet. As AngularJs Technology is platform independent technology it is compatible with other systems with different platform. Project is going to use AngularJs Technology with mySQL in backend.

Technical Requirement	How Accomplished?
Language used for coding	AngularJs, Javascript,asp.net
Data Server	mySQL
Documentation Tools	Microsoft Office Tools
Development Environment	Windows 10

Table 2.1 Technical Feasibility

Since the development tools and support for any operating system is easily available the development of this project is a viable option. Also the preliminary designs and techniques suggest this project could be implanted using the current set of resources by the team to which the development was assigned.

2.1.2   Time Schedule Feasibility

Time schedule plays a vital role in client's project. If the project is not delivered at due time then it can cause a project failure. Hence before undertaking particular project high concentration should be focused on the time management by project manager.

It should also be taken care that the staff, which is related with the project, should be able to complete the technical tasks in given schedule. If the current staff is not sufficient in completing the project tasks, project manager should allot more technical persons.

2.1.3   Operational Feasibility

Operational Feasibility helps assess whether the application can sustain itself during the operational phase of the application. The major aspects to be considered
- Performance: The performance of the system must be on par with what is specified and should not degrade over a period of time.

- Control: The current operational mode provides effective control for security.
- Efficiency: The current operational mode provides maximum use of available resources.
- Services: The current operational mode provides reliable services including flexibility & expandability
- There is no equipment to buy at all – as the system already exists and is previously implemented in Silverlight,only the cost for development is applicable.

2.1.4 Implementation Feasibility

Implementation feasibility is concerned with specifying external resources and software's that will successfully satisfy the user requirements. We have given more importance to external resources and configuration of the system rather than the actual map of the hardware. we have tried toutilise third party services as well where ever applicable so that cost is reduced. A proper implementation is essential to provide a reliable system meet the requirement of the organization.

2.1.5 Economic Feasiblity

Because this project is based on extension i.e. whole of the system is currently working under an existing system (i.e silverlight). Now to extend the system into a computerized one, the initial investment of hardware costs and network setup is not so much. But on measuring the cost effectiveness of the whole system, it will be proved very soon that in long term this system will be economically beneficial to the NGL as Silverlight is now shutting down it enhances the compatibility with all browsers and extends the application shelf life, Since the development tools and resources are easily available and the ROI (Return over Investment) for this application is greater than the effort put in it can be affirmed that the application is economically feasible.

2.2 PROJECT PLANNING

2.2.1 Project Development Approach and Justification

The waterfall model is a sequential (non-iterative) design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall)   through   the   phases   of   conception,   initiation, analysis, design, construction, testing, production/implementation and maintenance.

The waterfall development model originates in the manufacturing and construction industries: highly structured physical environments in which after-the-fact changes are prohibitively costly, if not impossible. Because a time when no formal software development methodologies existed, this hardware-oriented model was simply adapted for software development.
In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.
Waterfall model is the earliest SDLC approach that was used for software development.

The waterfall Model illustrates the software development process in a linear sequential flow; hence it is also referred to as a linear-sequential life cycle model. This means that any phase in the development process begins only if the previous phase is complete. In waterfall model phases do not overlap.

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially. Following is a diagrammatic representation of different phases of waterfall model.
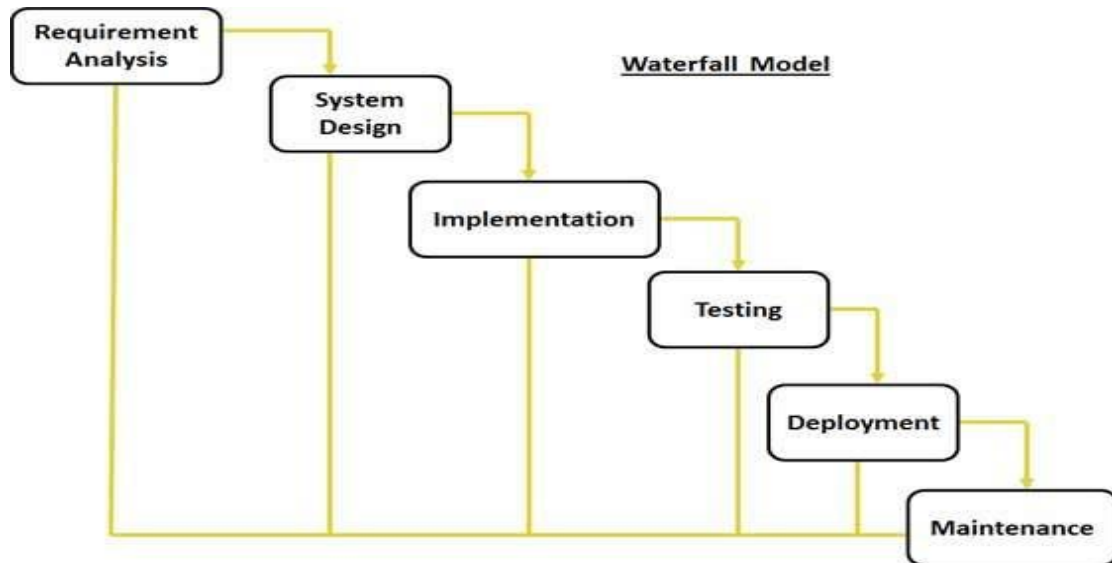
Figure 2.1   Waterfall Model (1)

Why Waterfall Model?

- This model is simple and easy to understand and use.
- This model is useful as the project has been migrated from Microsoft Silverlight to AngularJS. So, the requirements are the same and no further modification is required.
- It is easy to manage due to the rigidity of the model – each phase has specific deliverables and a review process.
- In this model phases are processed and completed one at a time. Phases do not overlap.
- Waterfall model works well for smaller projects where requirements are very well understood.

2.2.2   Milestones and Deliverables

Management needs information. As software is intangible, this information can only be provided as documents that describe that state of the software being developed. Without this information, it is impossible to judge progress and cost estimates and schedules cannot be updated. When planning a project series of milestones are established.

Milestones:

- Milestone is an end-point of the software process activity.
- At each milestone there should be formal output, such as report, that can be represented to the management. The weekly report is submitted to project guide, which include day to day work report.
- Milestone represents the end of the distinct, logical stage in the project.

Deliverables:

- Deliverables is a project report that is delivered to the administrator of the project.
- Deliverables are delivered to the administrators of our organization at the end of the some major project phase such as specification, design, etc.
- Deliverables are usually milestone
- Milestones may be internal project results that are used by the project manager to check progress but which are not delivered to the administrator

2.2.3   Roles and Responsibilities

Name	Role				
	Analysis	Designing	Coding	Testing	Documentation
Kruti Raval	✔	✔	✔	✔	✔
Priyanka Trivedi	✔	✔	✔	✔	✔

Table 2.1   Roles and Responsibilities

2.2.4   Group Dependencies

To keep a project smooth-going a proper team structure has to be maintained. This project involved mixed control team structure.

A mixed-control team organization attempts to combine the benefits of centralized and decentralized control, while minimizing or avoiding their disadvantages.

Rather than treating all members the same, as in a decentralized organization, or treating single individual as the chief, as in a centralized organization, the mixed organization differentiates the engineers into senior and junior engineers. Each senior engineer leads a group of junior engineers and reports, in its turn, to a project manager. Control is vested in the project manager and senior programmers, while communication is decentralized among each set of individuals, peers, and their immediate supervisors.

2.2.5 Project Scheduling Chart

Name	December			January					February				March			
	12	19	26	2	9	16	23	30	6	13	20	27	6	13	20	27
Problem Definition	▓	▓	▓													
Requirement Gathering			▓	▓	▓											
Design				▓	▓	▓	▓	▓								
Coding						▓	▓	▓	▓	▓	▓	▓	▓			
Testing											▓	▓		▓	▓	
Integration													▓	▓		
Documentation						▓	▓	▓	▓	▓	▓	▓	▓	▓	▓	▓

Table 2.2   Project Scheduling Chart

3.0 SYSTEM REQUIREMENTS STUDY

3.1 STUDY OF CURRENT SYSTEM

Currently, there is no ERP system available to maintain result and do corresponding analysis for colleges. Though, many are available for schools but they cannot be used by colleges as they are not customizable.

3.2 PROBLEMS AND WEAKNESSES OF CURRENT SYSTEM

If we consider the current scenario, our college is not using any ERP System to maintain result or for faculty allocation. And manual result processing system as many problems as it is highly error prone and hard to use as lots of paperwork is involved. While there are many ERP solutions currently available, they are not much configurable as being hard to customize and doesn't cater to specific needs of our institution.

3.3 USER CHARACTERISTICS

The system provider services based on the role of user. Any user can use the web based application to access the system and they can use it anywhere given that their computing device is connected to the internet. They do not need technical expertise or possess any specific education level other than basic knowledge of computing device use. User can be Admin, CSR, Supervisor, Customer, Courier Vendor, Courier Agent or Airline Company.

3.4 HARDWARE AND SOFTWARE REQUIREMENTS

Client Side :
1. The client system shall operate with the modern Web browsers like Microsoft Internet Explorer versions 8.0 or above, Mozilla Firefox 1.5 or above, Google Chrome and Safari. But using Mozilla Firefox or Google Chrome stays highly recommended.
2. The client system's web browser shall not disable cookies or JavaScript.

Server Side :
1. The server system must be running Apache Tomcat Web Server.
2. The server system shall use MySQL server for data storage.
3. The server system must have at least 4GB of RAM.
4. The server system must be running on 64-bit microprocessor.
5. The server system must have at least 16GB hard disk.

3.5 CONSTRAINTS

3.5.1 Regulatory Policies

The policies which are designed for proper usage of system are as follows :
1. Any user will not be able modify inactive/delete customer details.
2. Any user will not be able modify inactive/delete courier details.

3. Any user will not be able modify inactive/delete airline details.
4. All user would rightly exercise their powers and would not leak their credentials to any other user of the system.
5. Admin would rightly exercise his rights, for identification and allotment for different user roles.

3.5.2 Hardware Limitations

Since neither the mobile application nor the web portal have any designated hardware, it does not have any direct hardware interfaces. The hardware connection to the database server is managed by the underlying operating system on the mobile phone and the web server.

3.5.3 Interfaces to Other Applications

There is no application using this application as an interface so-far.

3.5.4 Parallel Operations

There are no parallel operations as such executing during the operation of the current application

3.5.5 Reliability Requirements

The reliability requirements for appropriate usage of system are as follows :
1. The system shall meet or exceed 99.99% uptime.
2. The system shall not be unavailable more than 1 hour per 1000 hours of operation.
3. Less than 20 seconds shall be needed to restart the system after a failure 95% of the time.

3.5.6 Criticality of the Application

Risk Category	Application Task
High Risk	• Internet Facing Applications • Storing personal data • Storing sensitive information of customers and various stakeholders
Low Risk	• Internal Supporting Applications facing internet • Standalone applications (e.g. Batch Applications)

Table 3.1   Application Risk Analysis

3.5.7 Safety and Security Consideration

The Safety considerations are :
1. The responsibility of the material to be shared lies with administrators of each college and they will be responsible for the material uploaded by them.

2. The database may get crashed at any certain time due to operating system failure. Therefore, it is required to take the database backup at regular intervals so that the database is not lost.

The Security considerations are :
1. Customer should not be able to view the details of any other customer than their own.
2. Courier Vendor should not be able to view the details of any other Courier Vendor than their own.
3. Courier Agent should not be able to view the details of any other Courier Agent than their own.
4. Airline Company should not be able to view the details of any other Airline Company than their own.
5. All users should be authenticated before accessing the system.
6. All users must be provided access to system strictly as per the role.
7. The system shall transfer/save data securely.
8. The system shall be protected against various security vulnerabilities
9. The system's database server shall be accessed by authenticated admin only.

3.6 ASSUMPTIONS AND DEPENDENCIES

The Assumptions are :

1. All customers would be associated with one of the NGL registered companies, i.e. NGL, NGW, NGB, NGC, NGT and FDX.
2. All users will access our system only via internet and use a modern web browser.
3. There won't be major change of responsibility for the given role.

The External Dependencies are :

1. CheersLookup Search would give accurate distance measurement for given zip code to correctly calculate STEM mileage.
2. It will also verify address against entered zip code by the user.
3. Airline Schedule available via 3rd party service would be as accurate as possible

4.0 SYSTEM ANALYSIS

4.1 REQUIREMENTS OF NEW SYSTEM

4.1.1 Use-case Diagram

Figure 4.1   Usecase Diagram For Customer Creation

Figure 4.2   Usecase Diagram For Customer Modification

Figure 4.3   Usecase Diagram For Courier Creation

Figure 4.4   Usecase Diagram For Courier Modification

Figure 4.5   Usecase Diagram For Airport Creation

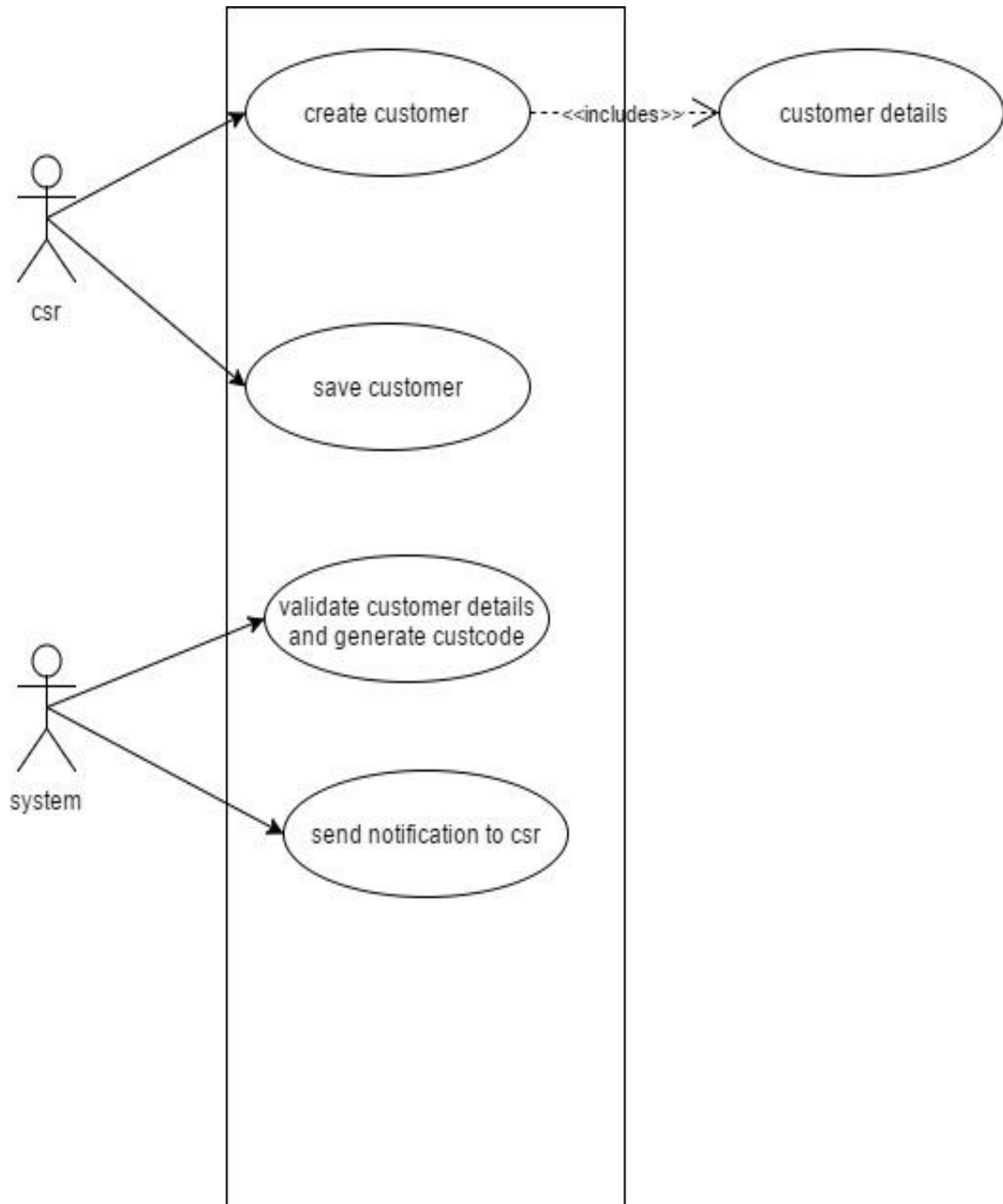Figure 4.6   Usecase Diagram For Airport Modification

Figure 4.7   Usecase Diagram For Airline Creation
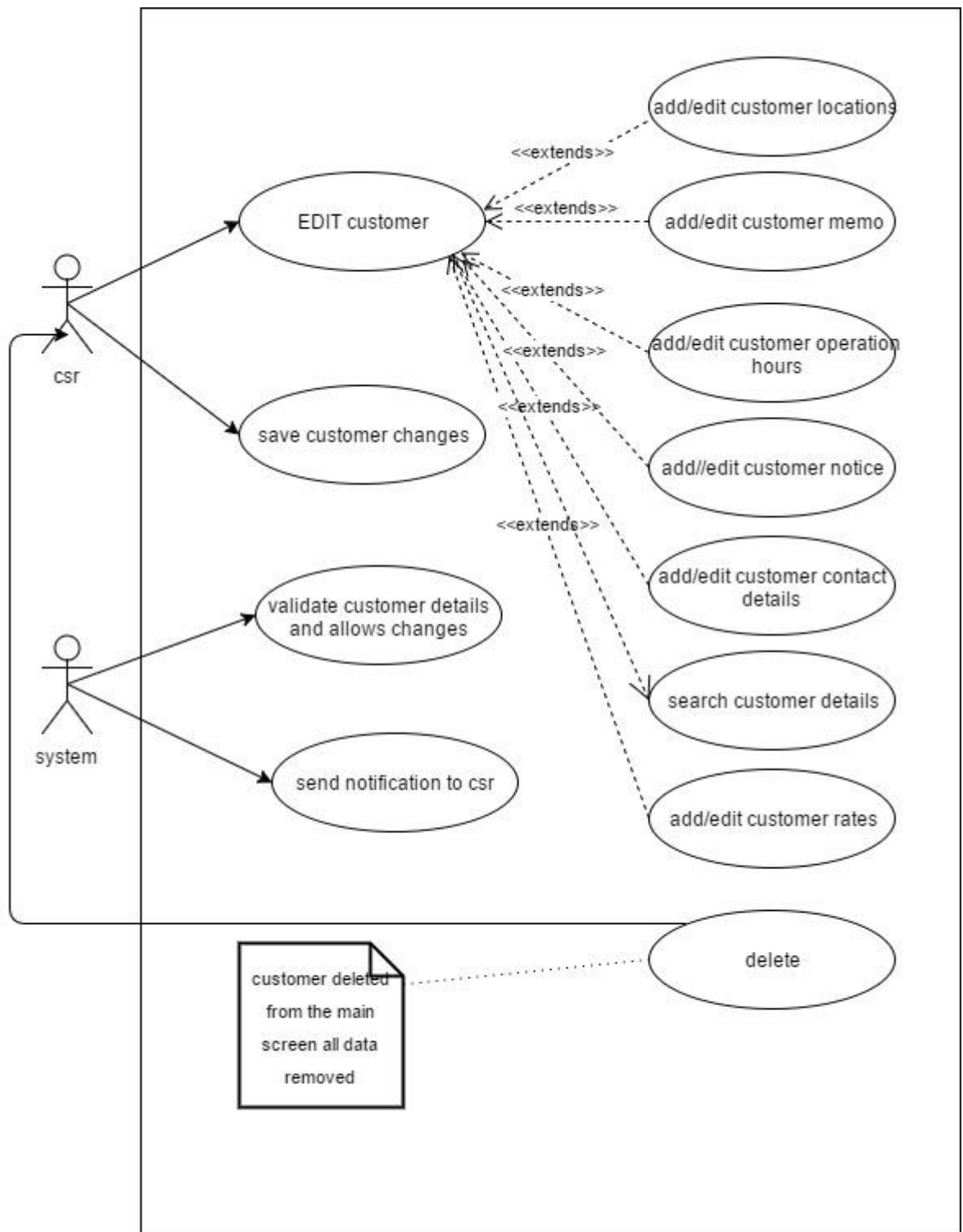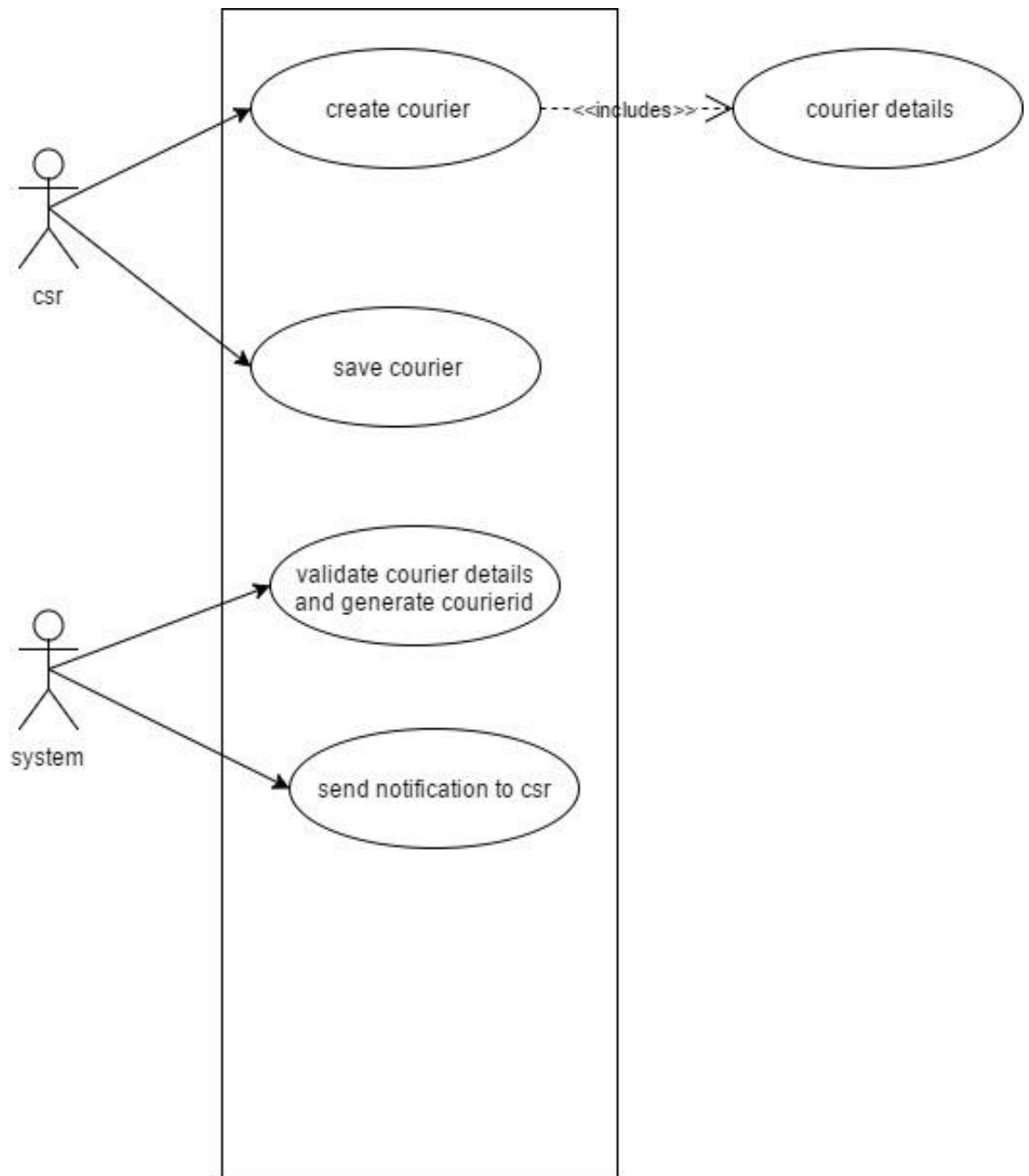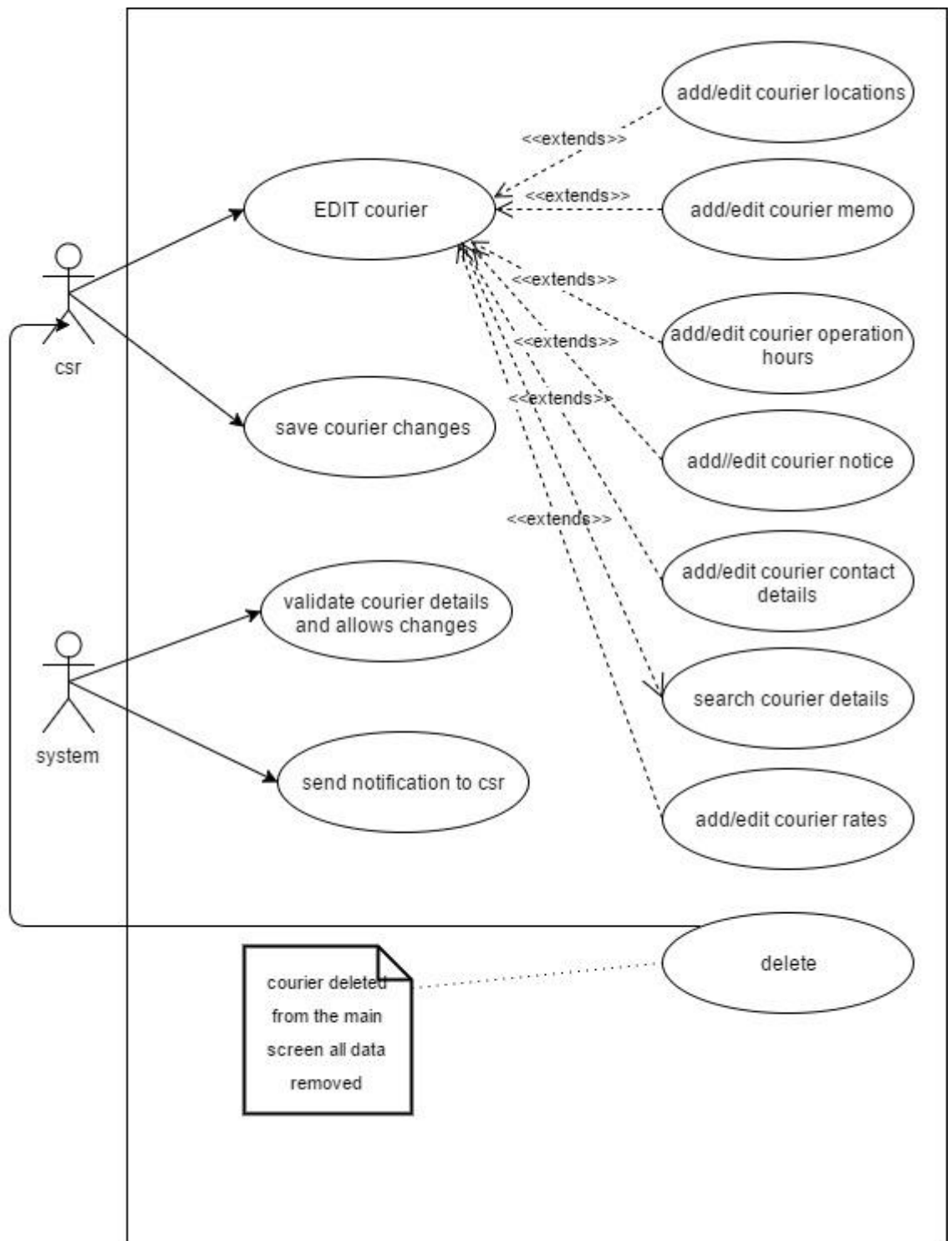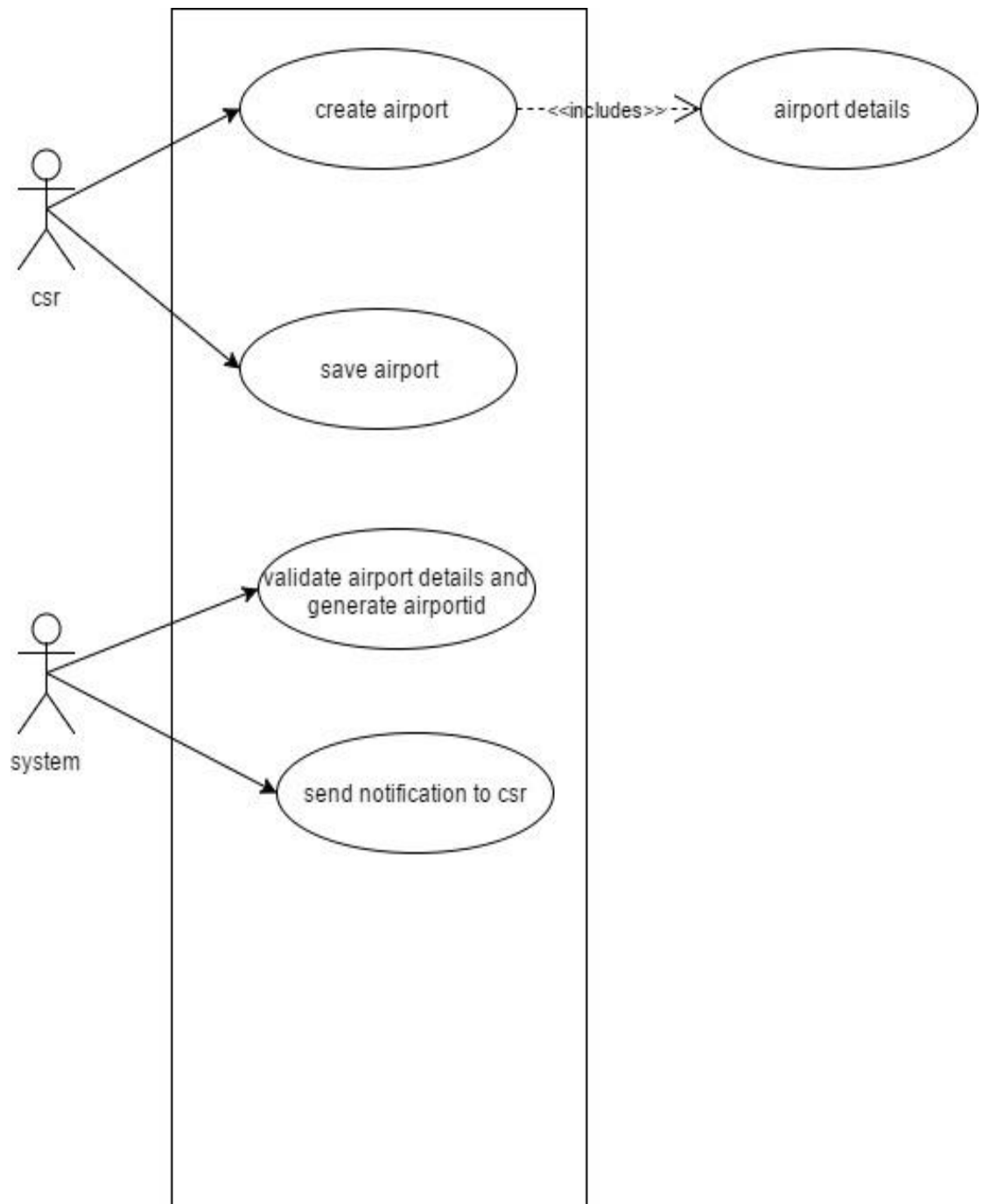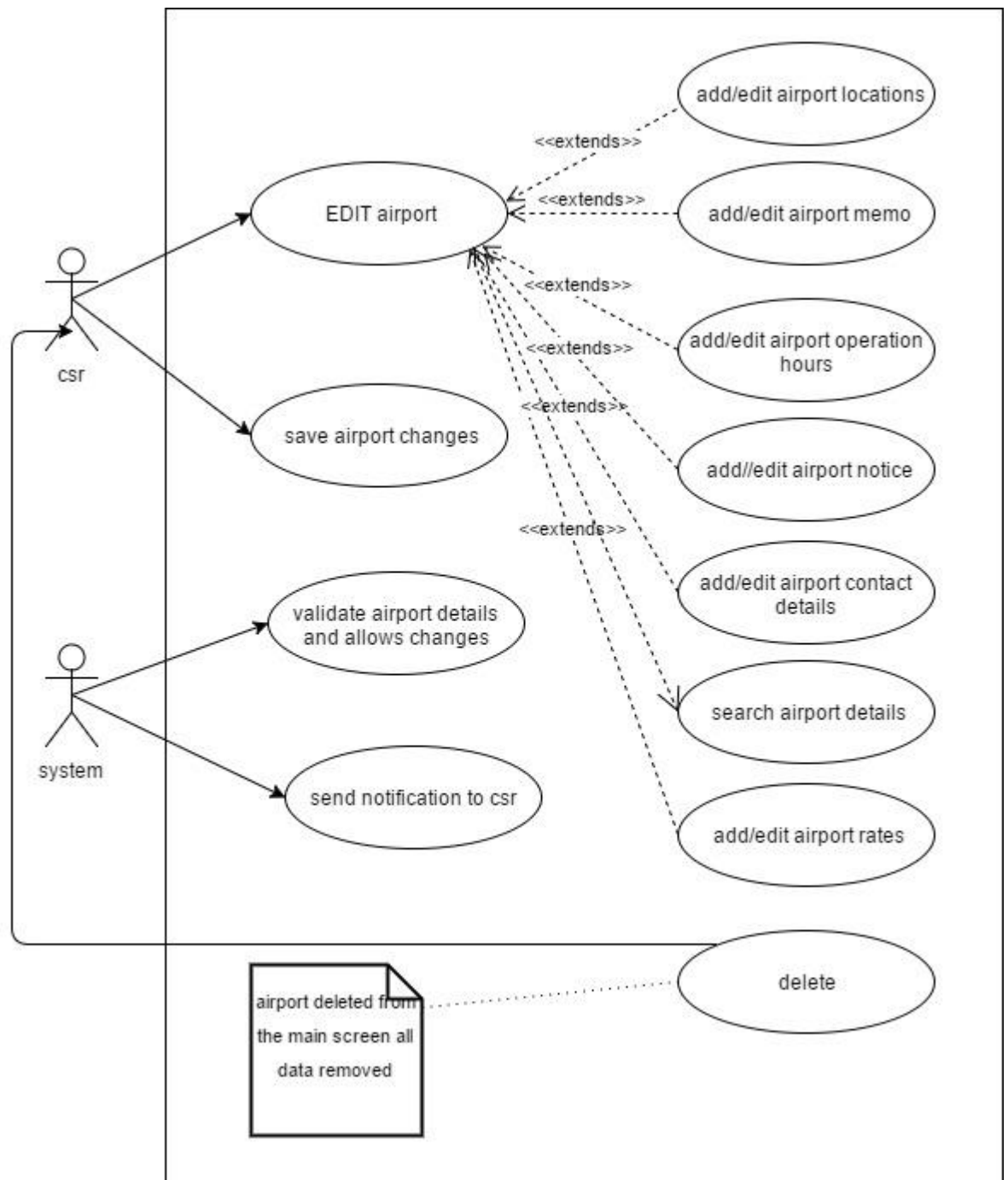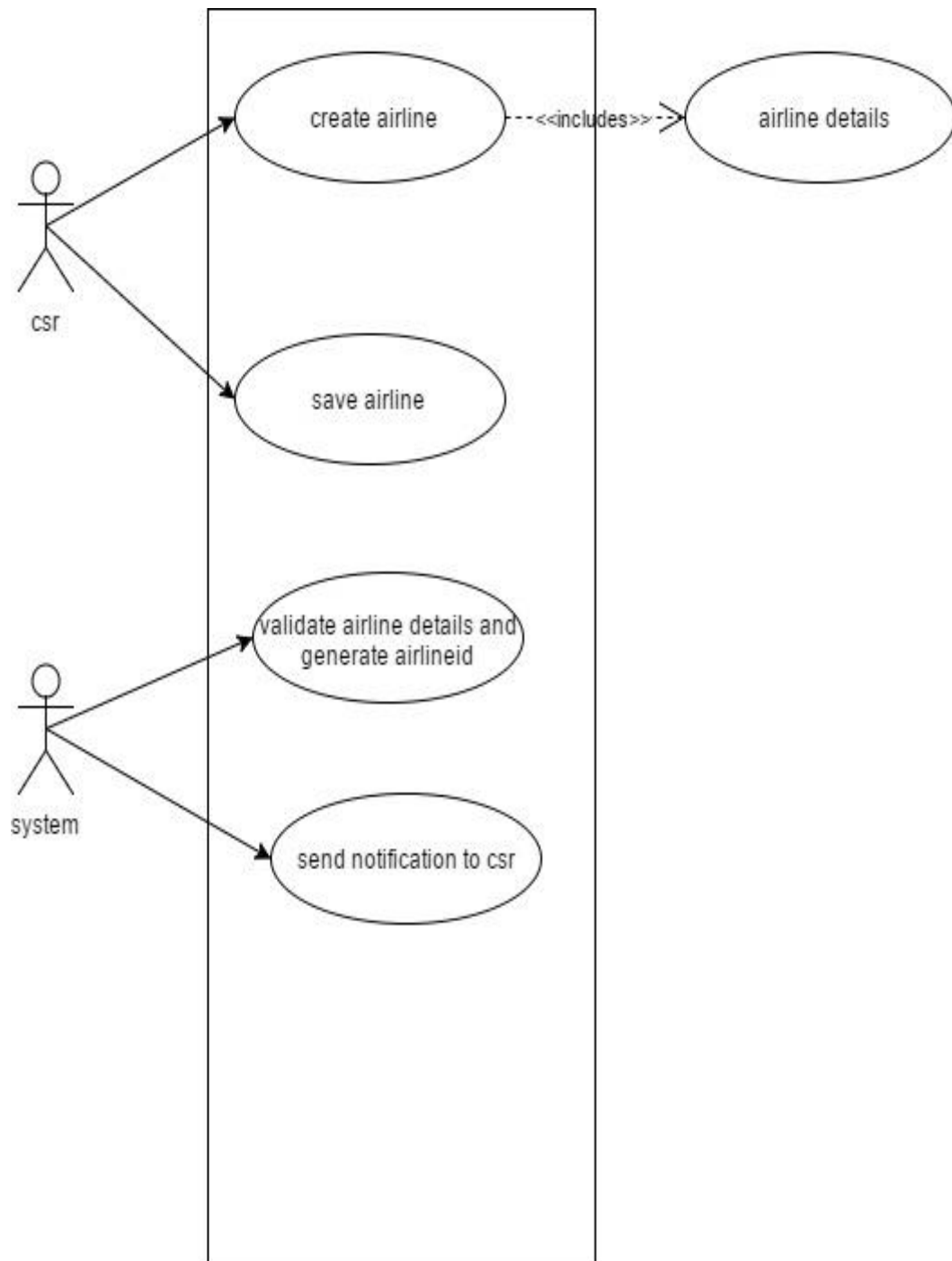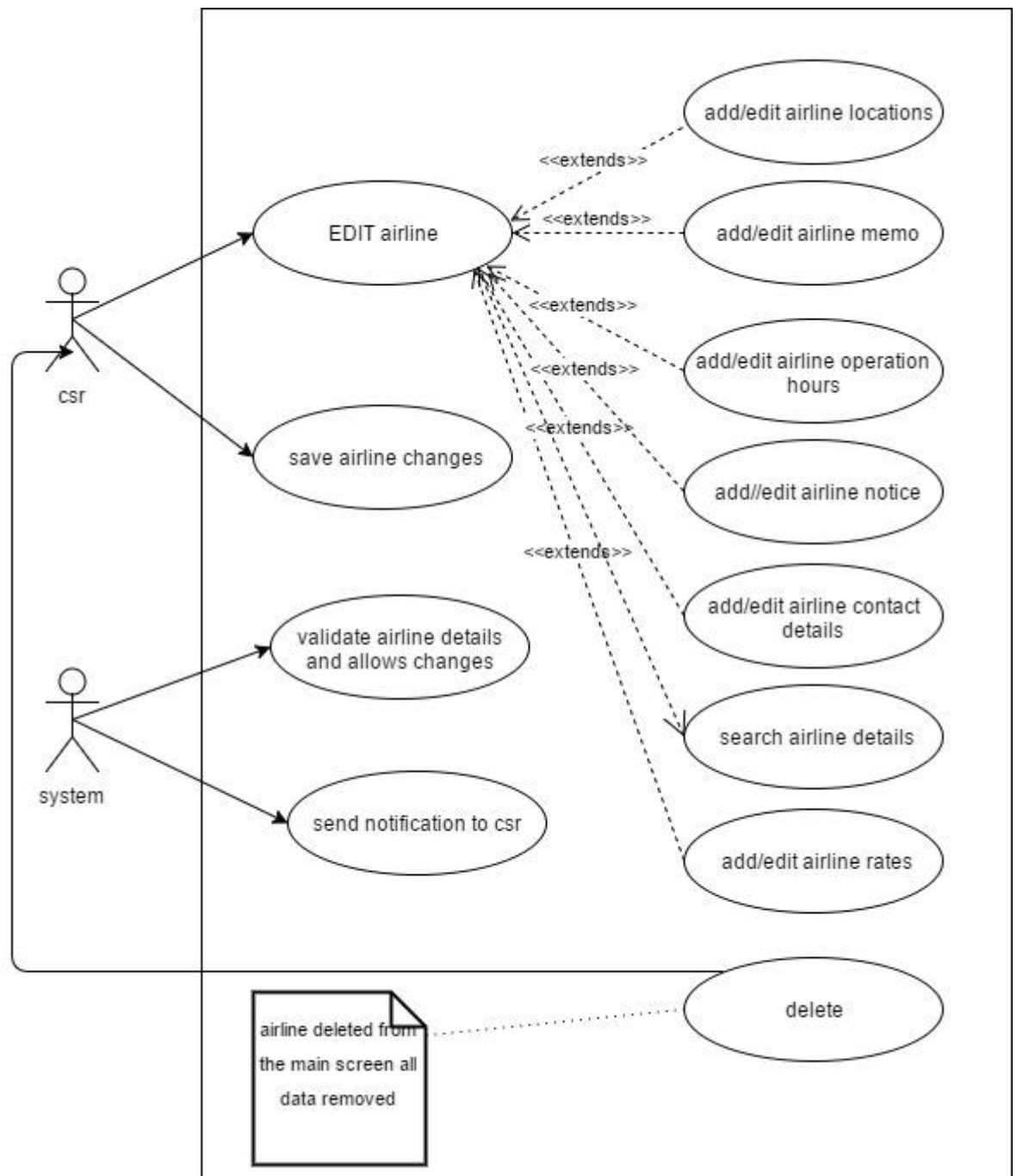
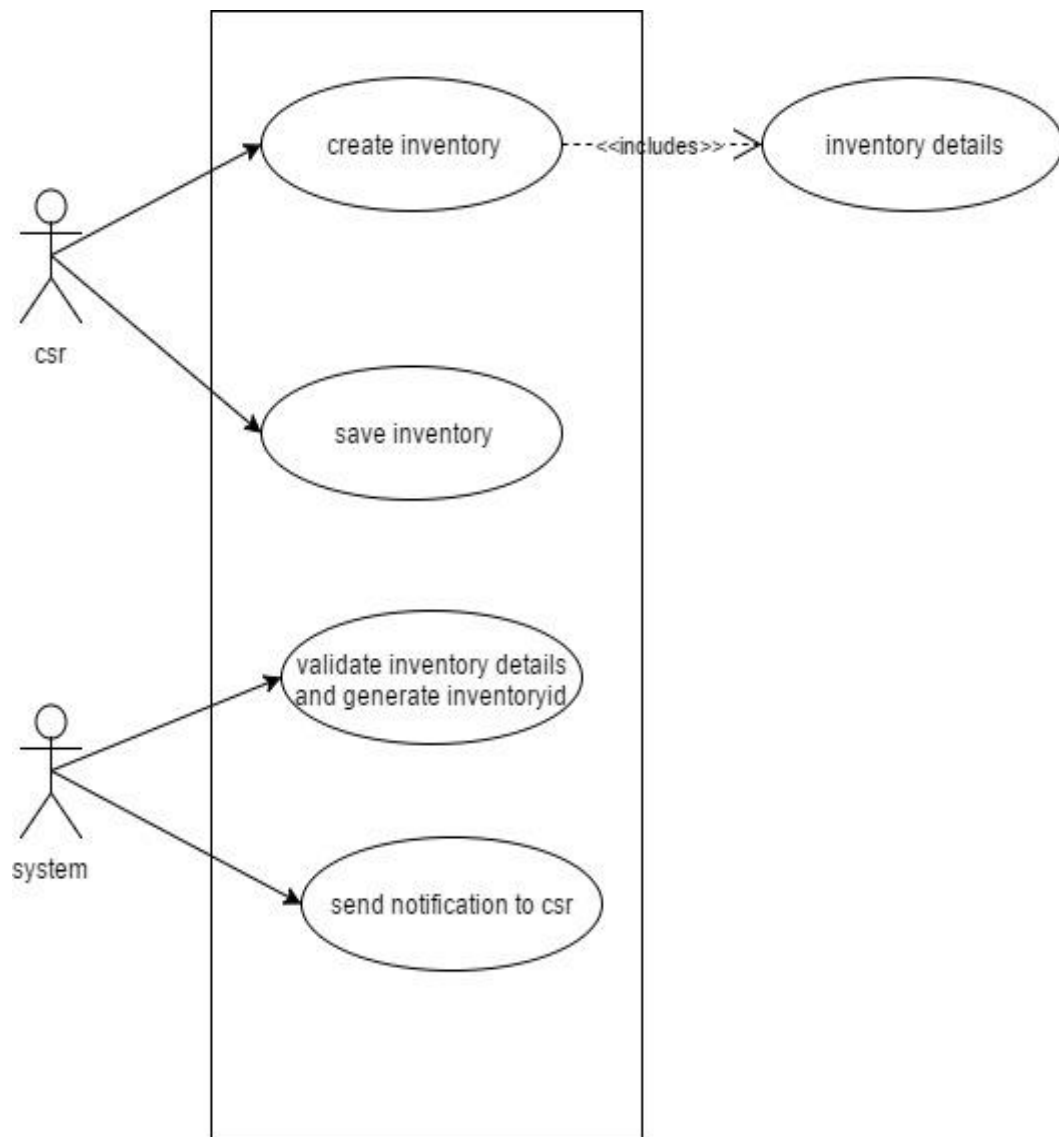Figure 4.8   Usecase Diagram For Airline Modification

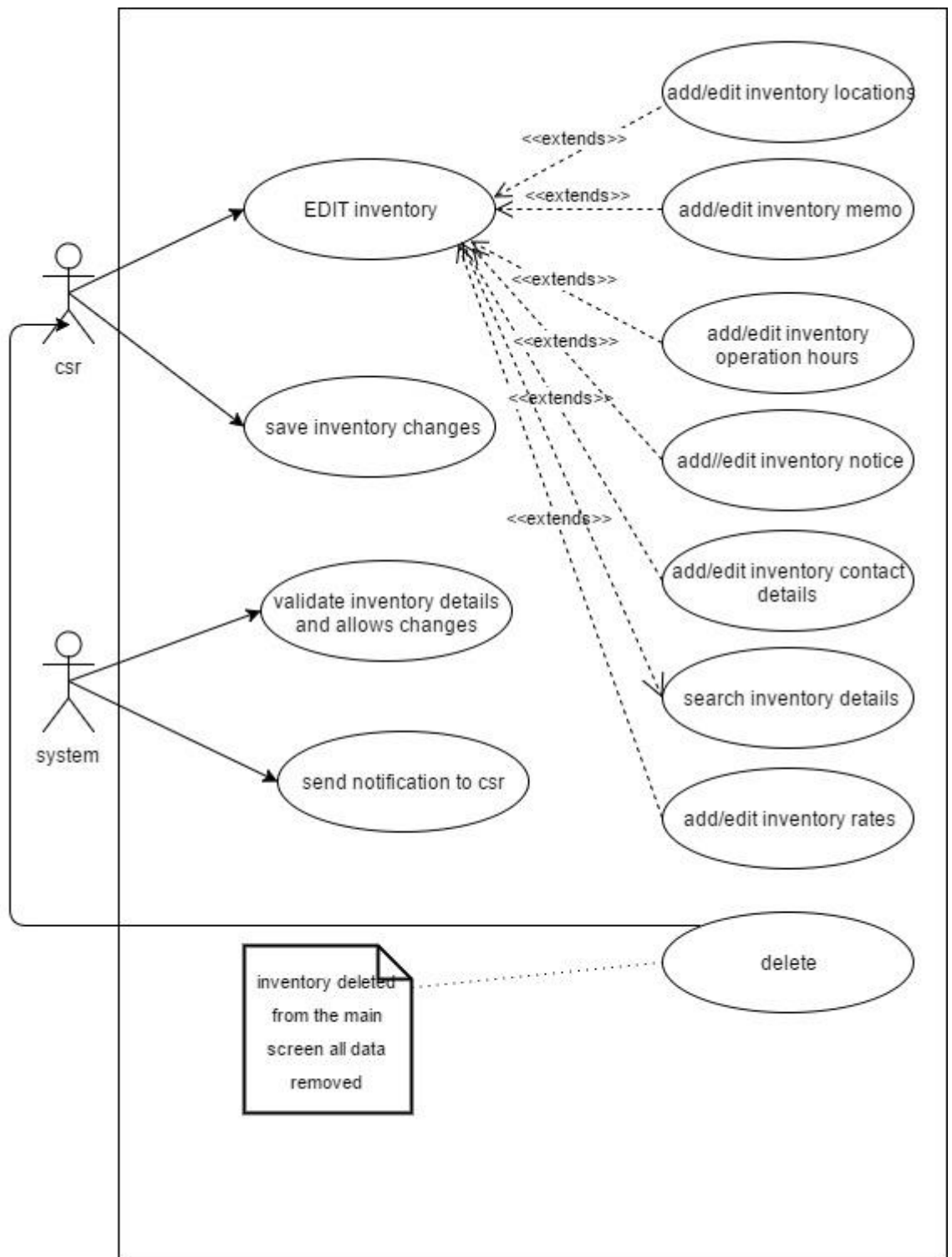Figure 4.9   Usecase Diagram For Inventory Creation

Figure 4.10 Usecase Diagram For Inventory Modification

4.1.2 System Requirements

1. The system will be accessible through a web browser and an internet connection.
2. The system shall interface with other modules developed by our peers.
3. All web presentation code shall conform to the XHTML Transitional 1.0 standard.
4. All code shall be written in AngularJS, .NET MVC framework with WebAPI method and WCF services using various last and industry standard frameworks and designs patterns.
5. The client component of the system needs to be accessible on multiple platforms and from remote locations.
6. The server must be capable of storing a large amount of data.

User Requirements

User must be aware that system works properly with full availability, reliability, security and safety. The user responsibilities are as follows:

- Should know how to use software
- Should adhere to guidelines and prescribed standards
- Must have a valid NGL login ID and password.

Described below are the common functional requirements which are implemented in all the screens of courier, customer, airline, airport and inventory.

Functional requirements:

R1 grid databind on valid custcode entered:
Purpose: When a person logs in with a valid username password and enters a valid custcode, then the all the data of the respective customer must bind with the verified custcode and retrieve all the data attached to the respective custcode.
 Input*:* User has to enter a valid custcode
Output: Then the details of the user are stored into the database. If the user doesn't enter valid custcode then no relevant record data will be available int the grid.if user enters zipcode or city,state,location then all the records relevant to this search will be displayed in the grid.
 Data elements:
#Nataccountid
custcode
customer name
zipcode
state
city

R2 add new record:
Purpose: This is implemented to enable user to either create a new customer record or to add data to existing customer record. A valid user account must be used.
Pre-condition: User must have a valid username and password to login first.
Input: (add button click)

Output: blank datafields will be displayed in addmode with some fields having its default values if any where the customer can enter the relevant and valid data to add to its database which on save button click will be saved in the database.
Data elements:
custcode
record status(=active)

R3 edit existing record:
Purpose: To make changes to the existing data in the database of existing customer related to the order .
Pre-condition: User must have a valid custcode and account in NGL connect and some pre-existing data added to the database.
Input: custcode,recordstatus and the respective screenid for example if customer is using the customer documents screen to upload any documents then the data sent as input will be(custcode,recordstatus,custdocid) .
Output: respective data of the customer with their respective custdocid in edit mode which can be edited by the customer.
Data elements:
custcode
screened
record status(=active)

R3 trans existing record:
Purpose: To save changes to the existing data in the database of existing customer related to the order or save data of a new customer.
Pre-condition: User must have a valid custcode and account in NGL connect .
Input: custcode,recordstatus and the respective screenid for example if customer is using the customer documents screen to upload any documents then the data sent as input will be(custcode,recordstatus,custdocid) .
Output: record saved successfully message.

R3.1 save in addmode:
Input: custcode,recordstatus.(here respective screenID is not sent as the new record will call a stored procedure on the server side which will generate a new screenID for the new record generated.)
Output: record saved successfully message.
Data elements:
custcode
recordstatus

R3.2 save in editmode:
Input: custcode,recordstatus and the respective screenid for example if customer is using the customer documents screen to upload any documents then the data sent as input will be(custcode,recordstatus,custdocid) .
Output: record saved successfully message.
Data elements:
custcode
screenid
record status(=active)

R4 common validation functions in all screens:
Purpose: check for the validation conditions on the respective fields and then let data allow or display error message as per the data.
Pre-condition: user logged in.

R4.1 E-mail validation function:
Purpose: To check if the customer has entered a valid email address or not .
Pre-condition: User must have a valid custcode and account in NGL connect and enter data in e-mail field.
Input: EmailID.
Output: works normally if valid email id is entered else show error message invalid email address.
Data elements:
custcode
screenid
emailed
record status(=active)

R4.2 FAX validation function:
Purpose: To check if the customer has entered a valid fax address or not .
Pre-condition: User must have a valid custcode and account in NGL connect and enter data in fax field.
Input: fax.
Output: works normally if valid fax is entered else show error message invalid fax address.
Data elements:
custcode
screenid
fax field
record status(=active)

R4.3 FTP validation function:
Purpose: To check if the customer has entered a valid ftp address or not .
Pre-condition: User must have a valid custcode and account in NGL connect and enter data in ftp field.
Input:ftp adress.
Output: works normally if valid ftp is entered else show error message invalid ftp address.
Data elements:
custcode
screenid
ftp address
record status(=active)

R4.4 phone number validation function:
Purpose: To check if the customer has entered a valid phone number address or not .
Pre-condition: User must have a valid custcode and account in NGL connect and enter data in phone number field.
Input: phone number.

Output: works normally if valid phone number is entered else show error message invalid phone number .
Data elements:
custcode
phone number
screened
record status(=active)

R5 delete existing record:
Purpose: To delete existing record in the database of existing customer related to the order or delete data of a new customer.
Pre-condition: User must have a valid custcode and account in NGL connect .
Input: custcode,recordstatus and the respective screenid for example if customer is using the customer documents screen to upload any documents then the data sent as input will be(custcode,recordstatus,custdocid) (for delete record validations will not be checked).
Output: ask the user "are you sure you want to delete this record?" confirmation.if user clicks on yes then delete the record and show success message "record deleted successfully".
Data elements:
custcode
screenid
record status(=active)

R6 Record status changed:
Purpose: This is a functionality of the divex grid used in all the screens where there atr three kinds of record status Active, Deleted and All on change of which this method will be called and respectively data will be bind to grid.
Pre-condition: User must be logged in with a valid custcode.
Input: (custcode,recordstatus,screenID)

R6.1 record status active:
Purpose: if the record status sent in the request is active then the user will getall the active records bind in the grid.
Pre-condition: user enters custcode .
Input: user choose Active radiobutton in the grid(though bydefault also the value of record status will be Active only)
Output: display all the current active records of the customer in the divex grid.
Data elements:
custcode
screenid
record status(=active)

R6.2 record status deleted:
Purpose: if the record status sent in the request is deleted then the user will get all the deleted or inactive records bind in the grid.
Pre-condition: user enters custcode .
Input: user choose deleted radio button in the grid.
Output: display all the current deleted records of the customer in the devex grid.
Data elements:

custcode
screenid
record status(=deleted)

R7 Record status deleted:
Purpose: if the record status sent in the request is deleted then the user will get all the deleted and active records bind in the grid.
Pre-condition: user enters custcode .
Input: user choose ALL radio button in the grid.
Output: display all the current active and deleted records of the customer in the divex grid.
Data elements:
custcode
screenid
record status(=All)

R8 check validTo is smaller then validFrom:
Purpose: to check if the valid to field entered in the data is greater then today's date and also smaller than validfrom.
Pre-condition: user enters custcode.
Input: validTo and validfrom(optional)
Output: works normally if entered valid to date is smaller then valid from messageor if only valid to is entered,but if valid to date entered is smaller than todays date then display error message"valid to date must be greater than today's date". And if the entered valid to date is greater than valid from date then display error message "valid to date must be smaller than valid from date"
Data elements:
custcode
screenid
record status(=All)
validto
validfrom

R9 check required field validation:
Purpose: to check if the data is entered in the field which are set as required and also display error message if empty.
Pre-condition: user enters custcode.
Input: Enter fields details.
Output: on save button click works normally if entered data in the required field,but displays error message "XYZ field required" if required field is left empty.

R10 searchzipcode():
Purpose: to bind city, state and country from the entered zipcode.
Pre-condition: user enters custcode.
Input: enter a valid zipcode/postalcode in zipcode field.
Output: bind the state ,city and country fields of the respective zipcode.
Data elements:
custcode
screenid
record status(=All)

zipcode
state
country
city

Non-Functional requirements:

Safety Requirements: By incorporating a robust and proven DBMS (PostgreSQL) into the system, reliable performance and integrity of data is ensured. There must be a power back up for server requirements.

Performance Requirements: The product shall be based on web or mobile application and has to be run from a web server. The product shall take initial load time depending on the internet connection strength which also depends on from which device the product is run. The performance shall depend upon hardware components and connection strength of the client/customer. System can withstand many number of clients accessing the desired service.

Security Requirements: The system shall use secure sockets in all transactions that include any confidential client's information. The system shall automatically log out all clients after a period of inactivity. The system shall not leave any cookies on the client's computer containing the user's password and any of the user's confidential information. The customer's web browser shall never display a customer's password. The system's back-end servers shall only be accessible to authenticated administrator. This system maintains the secrecy of user details. This is maintained by using user id's and passwords. Data confidentiality can be provided in such a way that functionalities of one user cannot be used by other user. The system should not accept forge request is the main security requirement of this project
Software Quality Attributes

Functionality*:* The ability of the system to do the work for which it was intended.

Performance*:* The response time, utilization, and throughput behaviour of the system. Not to be confused with human performance or system delivery time.

Security*:* A measure of system's ability to resist unauthorized attempts at usage or behaviour modification, while still providing service to legitimate users.

Availability*:* The measure of time that the system is up and running correctly; the length of time between failures and the length of time needed to resume operation after a failure.

Usability: The ease of use and of training the end users of the system. Sub qualities: learnability, efficiency, affect, helpfulness, control.

Interoperability: The ability of two or more systems to cooperate at runtime.

Modifiability: The ease with which a software system can accommodate changes to its Software.

Portability: The ability of a system to run under different computing environments. The Environment types can be either hardware or software, but is usually a combination of the Two.

Reusability: The degree to which existing applications can be reused in new applications

Integration: The ability to make the separately developed components of the system work correctly together.

Testability: The ease with which software can be made to demonstrate its faults.

4.2 FEATURES OF NEW SYSTEM

Customer Creation
A customer can be shipper/consignee who wants to do business with NGL. To process various deals of customer, CSR has to enter details pertaining to that customer NGL will ask CSR to create based on details entered. It is a high priority and frequently used feature as it is the base of order processing flow.

Feature
Request Customer Creation: The application shall allow the CSR to take the customer creation request and enter customer details.
Save Customer Details: The system shall allow the creation of new customer entered by CSR after validating mandatory fields and then automatically generate customer code.

Customer Modification
A customer can be shipper/consignee who wants to do business with NGL. To process various deals of customer, CSR has to enter details pertaining to that customer. NGL will allow CSR to check for existing customer details. System will also allow CSR to delete the existing customer details if that customer is no longer in Business with NGL. It is a high priority and moderately used feature as it is the base of order processing flow.

Feature
Search Customer: The system shall allow CSR to search customer based on various Customer Details such as Customer Code, Customer Name, Associated Company, National Account, City, State, Zip-Code and Logistics/Non-Logistics and display list of selected records as per selection criteria entered by CSR or if only one customer is matched then it will direct open Customer Master Screen.
Edit Customer Details: The system shall allow CSR to update customer details.
Save Customer Details: The system shall allow CSR to save edited customer details after validating mandatory fields.
Delete Customer Record: The system shall allow CSR to delete selected customer record after prompting with confirmation message.

Courier Creation
A courier can be vendor/agent who wants to do business with NGL. NGL will identify various couriers based on quotation and rates offered by couriers and do

business with selective courier vendors according to location/area. Hence based on NGL decision, CSR will enter the courier details in the system and the system will automatically generate Courier Id. It is a high priority and frequently used feature as it is the base of order processing flow.

Feature
Request Courier Creation : The application shall allow the CSR to take the Courier creation request as per business deal and enter Courier Vendor details.
Save Courier Details : The system shall allow the creation of new customer entered by CSR after validating mandatory fields and then automatically generate Courier Id.

Courier Modification
A courier can be vendor/agent who wants to do business with NGL. To process various orders/deals placed by customer, CSR has to select various couriers based on pickup and delivery locations. NGL system will allow CSR to check for courier details and modify them. System will also allow CSR to delete the existing courier details if that vendor/agent is no longer in Business with NGL. It is a high priority and moderately used feature as it is the base of order processing flow.

Feature
Search Courier: The system shall allow CSR to search courier based on various Courier Details such as Courier ID, Courier Name, Vendor, Courier Type, Airport, City State, Zipcode, Courier Status and display list of selected records as per selection criteria entered by CSR or if only one customer is matched then it will direct open Customer Master Screen.
Edit Courier Details: The system shall allow CSR to update Courier details.
Save Courier Details: The system shall allow CSR to save edited Courier details after validating mandatory fields.
Delete Courier Record: The system shall allow CSR to delete selected Courier record after prompting with confirmation message.

Airline/Airport Creation
NGL does business with various Airlines based upon shipper/consignee, pickup/drop location. Based on selected Airlines, NGL will use various Airports and associate itinerary accordingly. It is a high priority and frequently used feature as it is the base of order processing flow.

Feature
Request Airline Creation : The application shall allow the CSR to take the airline creation request and enter airline details.
Save Airline Details : The system shall allow the creation of new airline details entered by CSR after validating mandatory fields

Airline/Airport Modification
NGL does business with various Airlines based upon shipper/consignee, pickup/drop location. Based on selected Airlines, NGL will use various Airports and associate itinerary accordingly. NGL will allow CSR to check for existing airline details. System will also allow CSR to delete the existing airline details if that airline is no longer in Business with NGL. It is a high priority and moderately used feature as it is the base of order processing flow.

Feature

Search Airline: The system shall allow CSR to search airline based on various Airline Details such as Airline Id, Airline Name, OAG ID and display list of selected records as per selection criteria entered by CSR or if only one customer is matched then it will direct open Customer Master Screen.

Edit Airline Details: The system shall allow CSR to update customer details.

Save Airline Details: The system shall allow CSR to save edited customer details after validating mandatory fields.

Delete Airline Record: The system shall allow CSR to delete selected airline record after prompting with confirmation message.

4.3 SYSTEM ACTIVITY

Customer Activity:
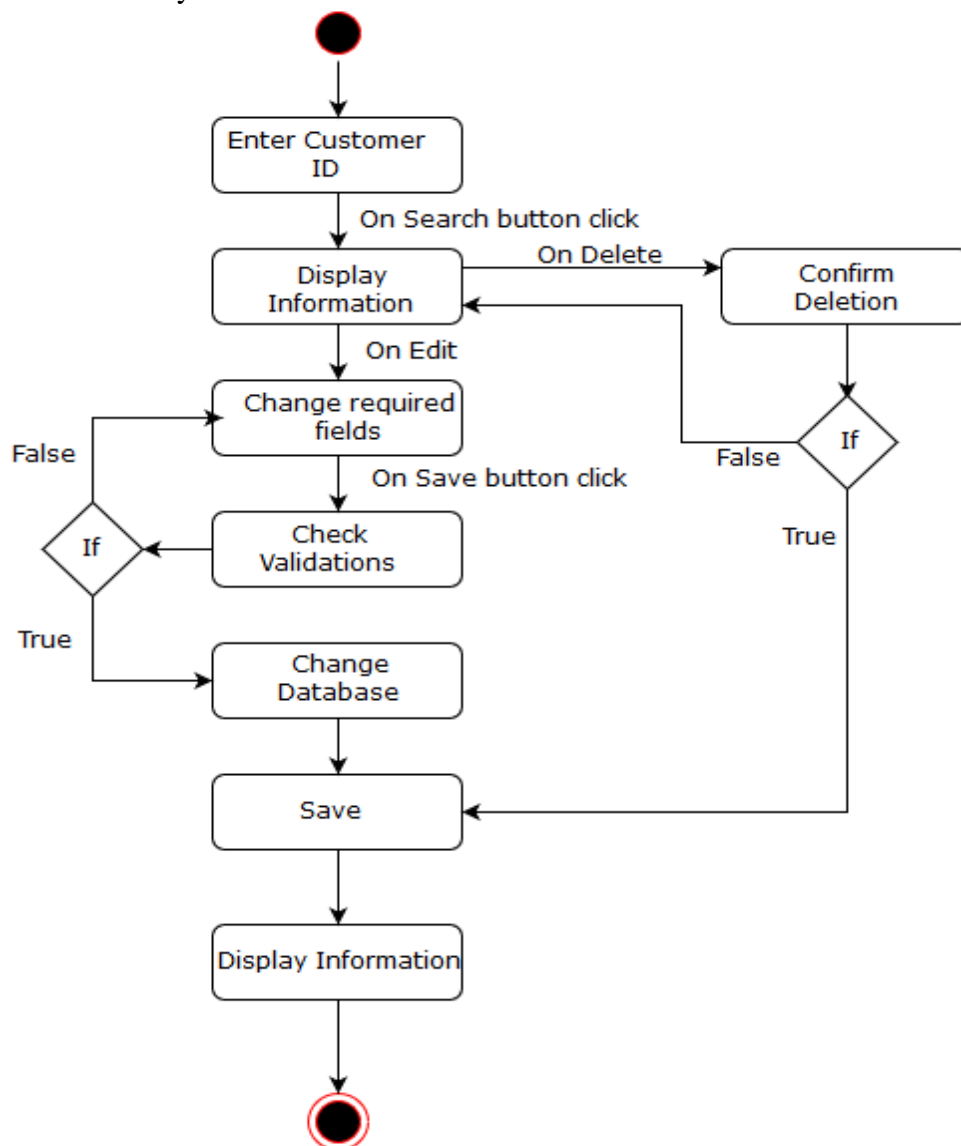
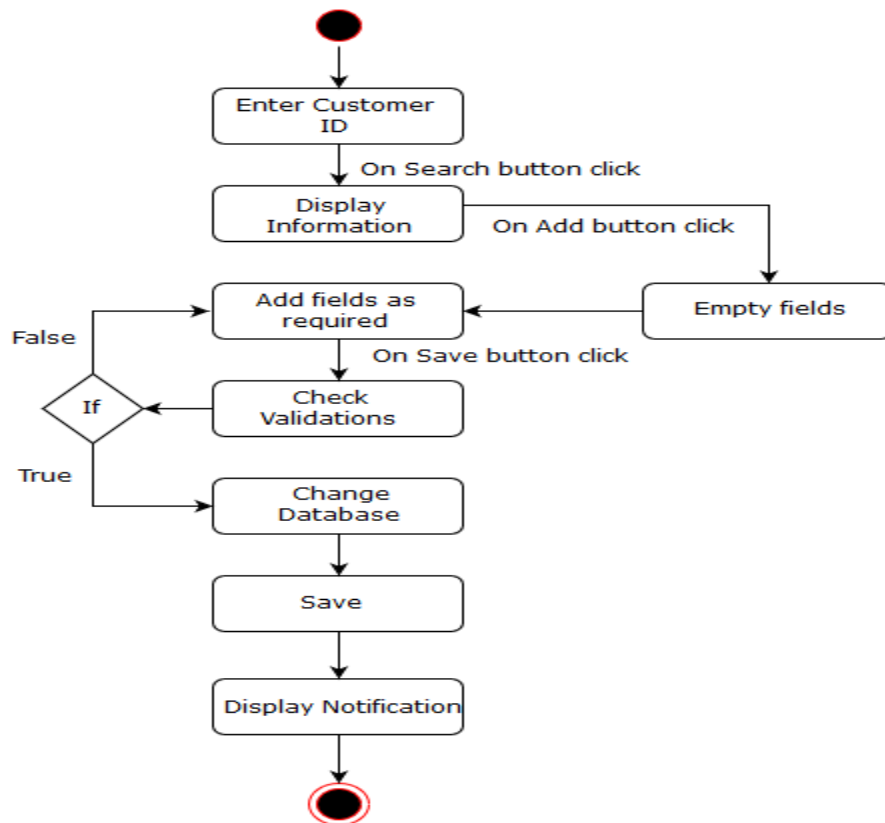Figure 4.11 Customer Modification Activity Diagram

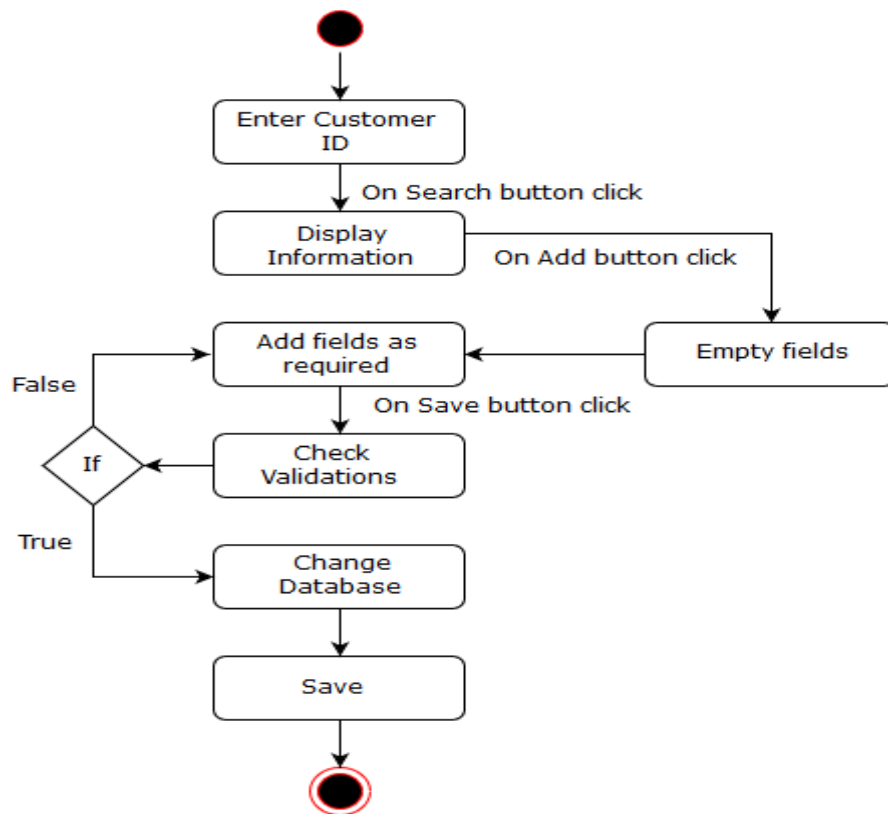Figure 4.12 Customer Creation Activity Diagram

Figure 4.13 New Customer Add Activity Diagram

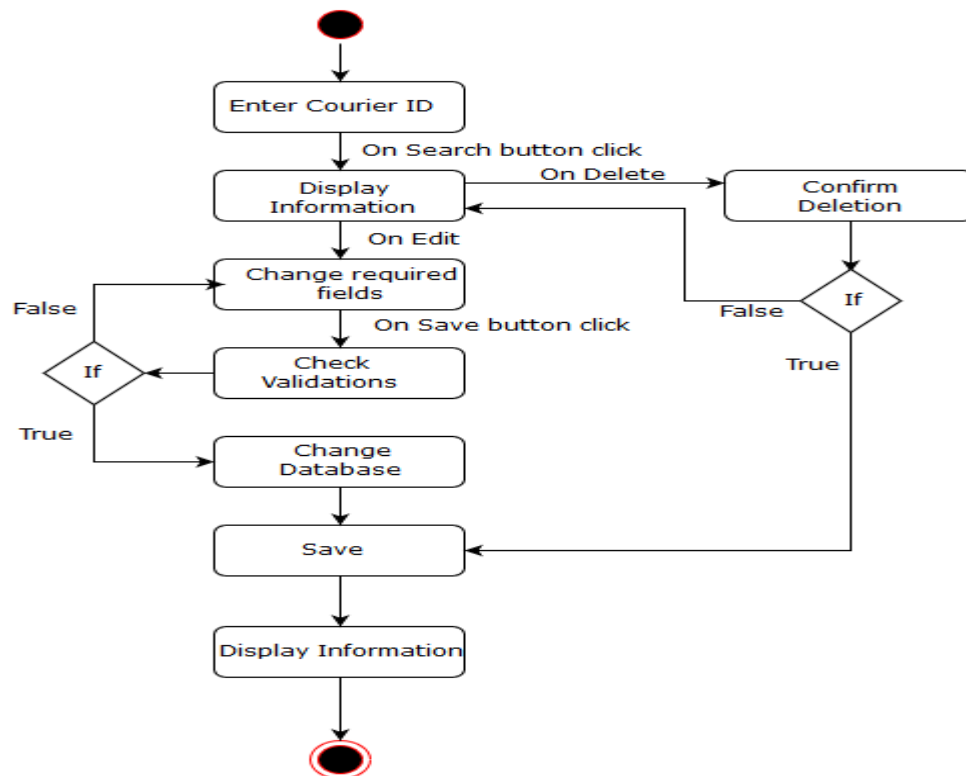Courier Activity:

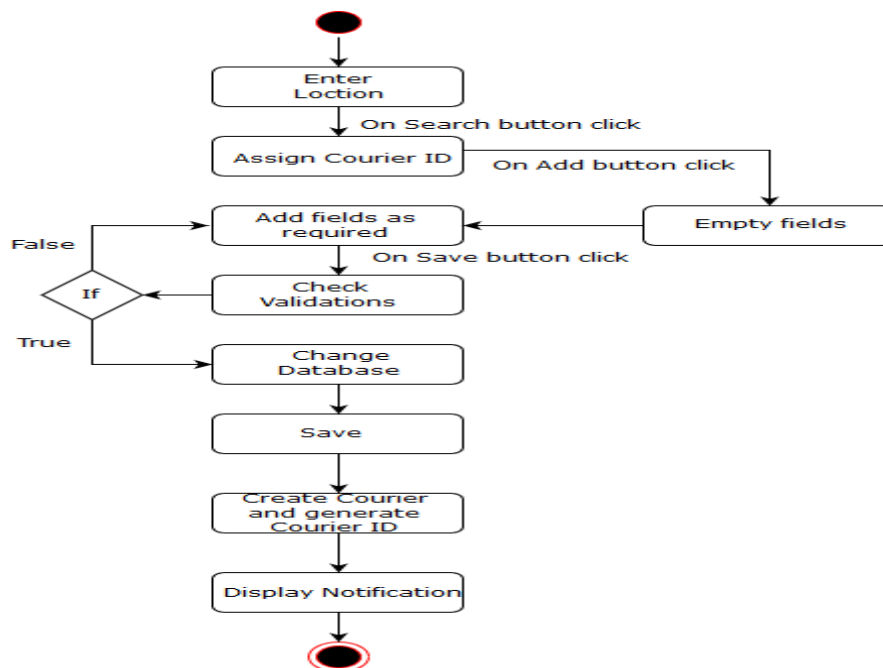Figure 4.14 Courier Modification Activity Diagram

Figure 4.15 Courier Creation Activity Diagram
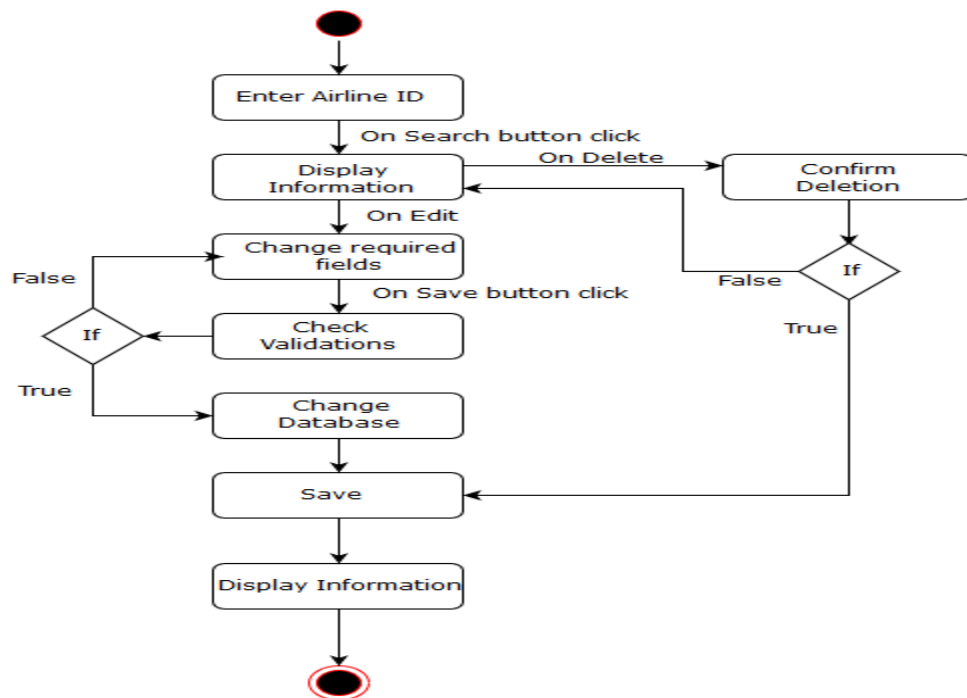
Airline/Airport Activity:

Figure 4.16 Airline Modification Activity Diagram
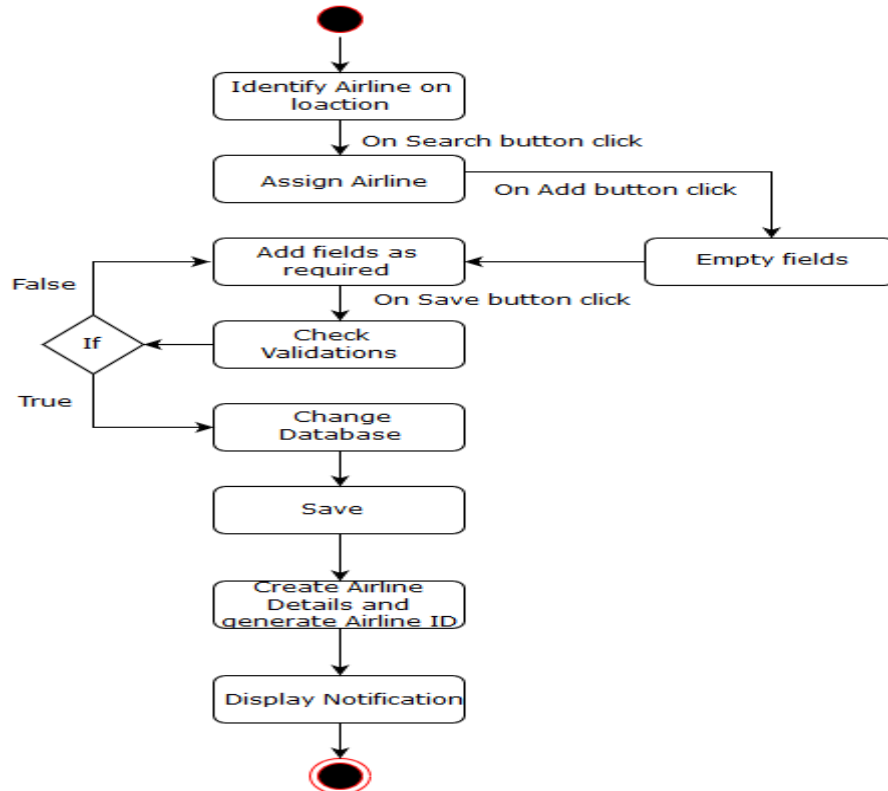

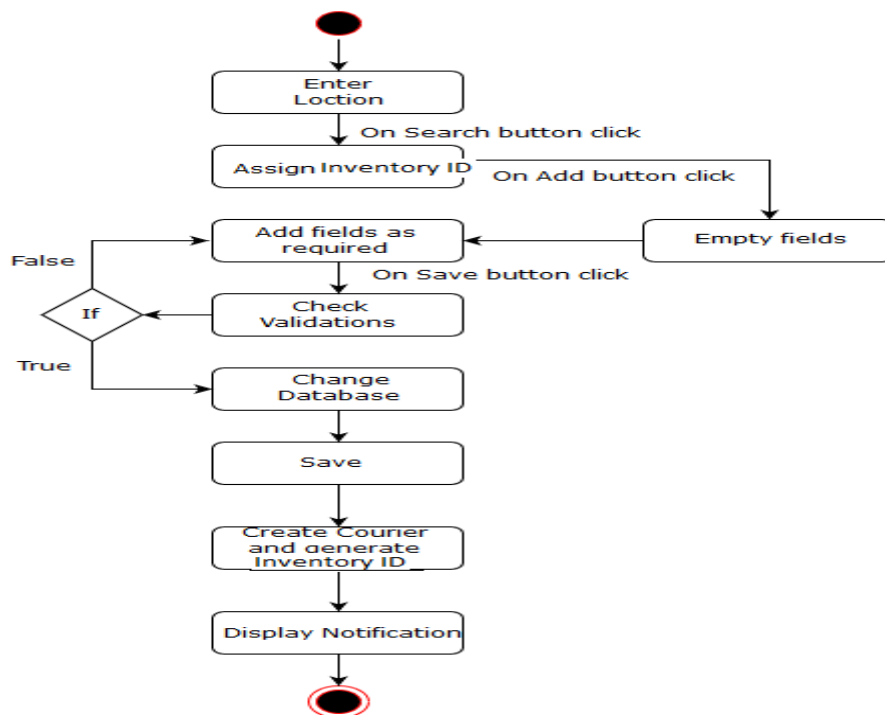Figure 4.17 Airline Creation Activity Diagram

Inventory Activity:

Figure 4.18 Inventory Creation Activity Diagram

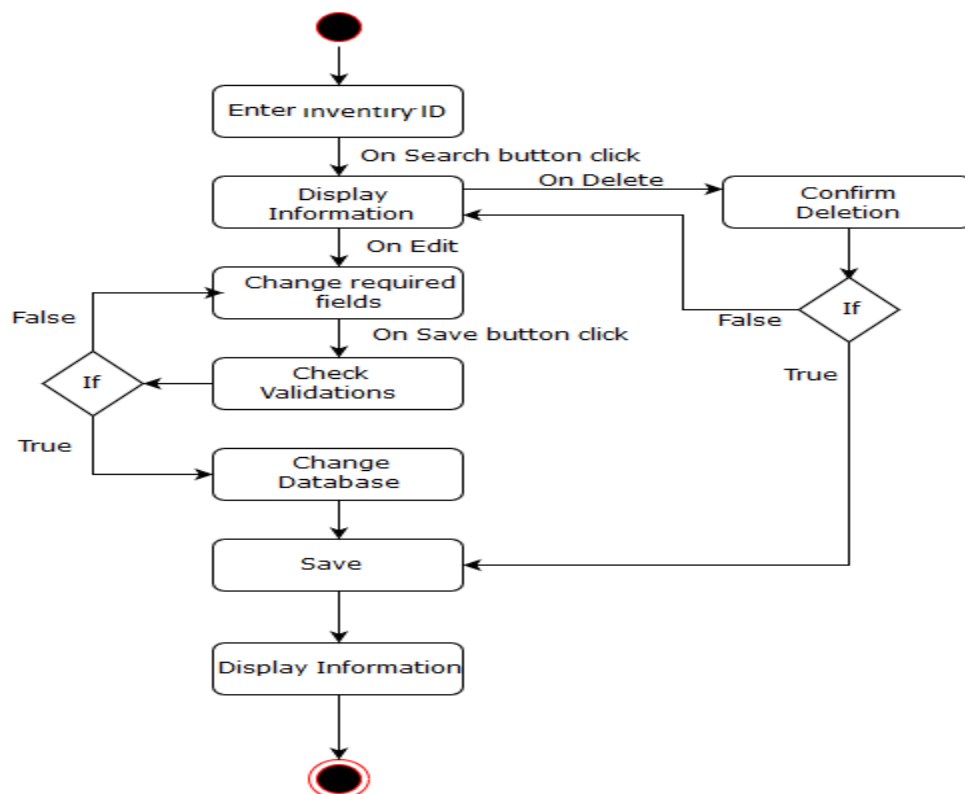
Figure 4.19 Inventory Modification Activity Diagram
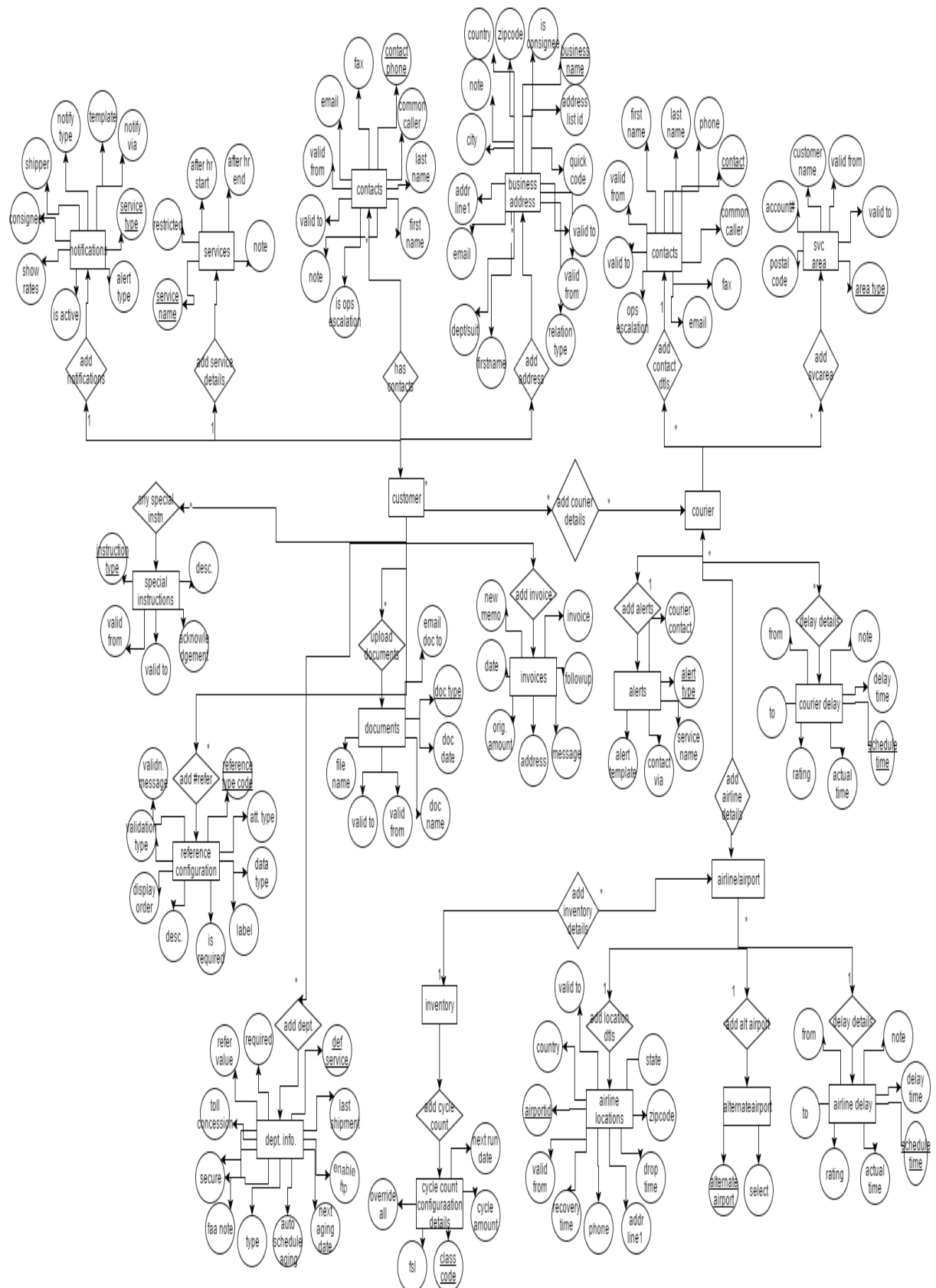
4.4 DATA MODELING – ER DIAGRAM

Figure 4.20 ER Diagram

5.0   SYSTEM DESIGN

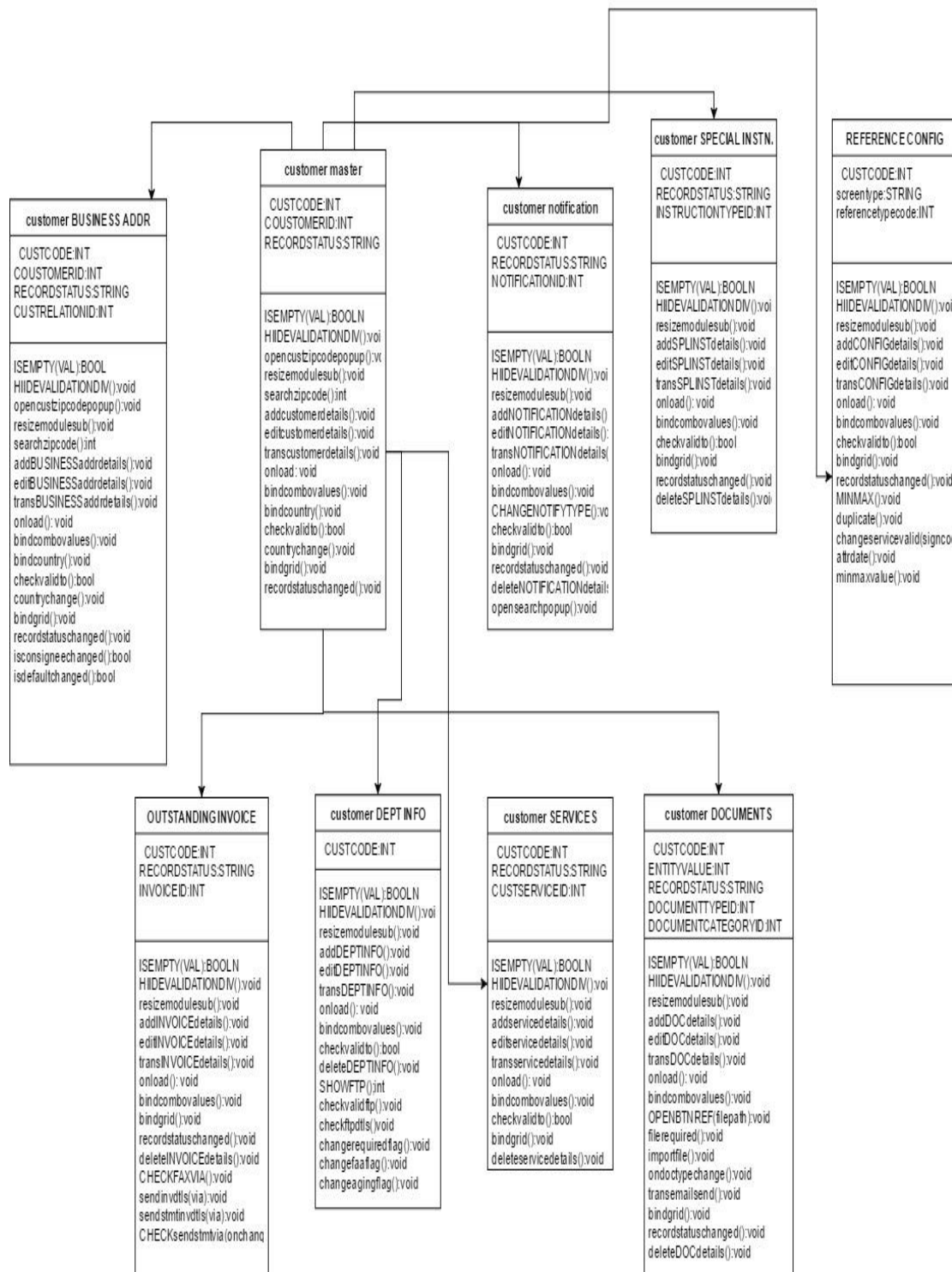## 5.1   SYSTEM ARCHITECTURE DESIGN

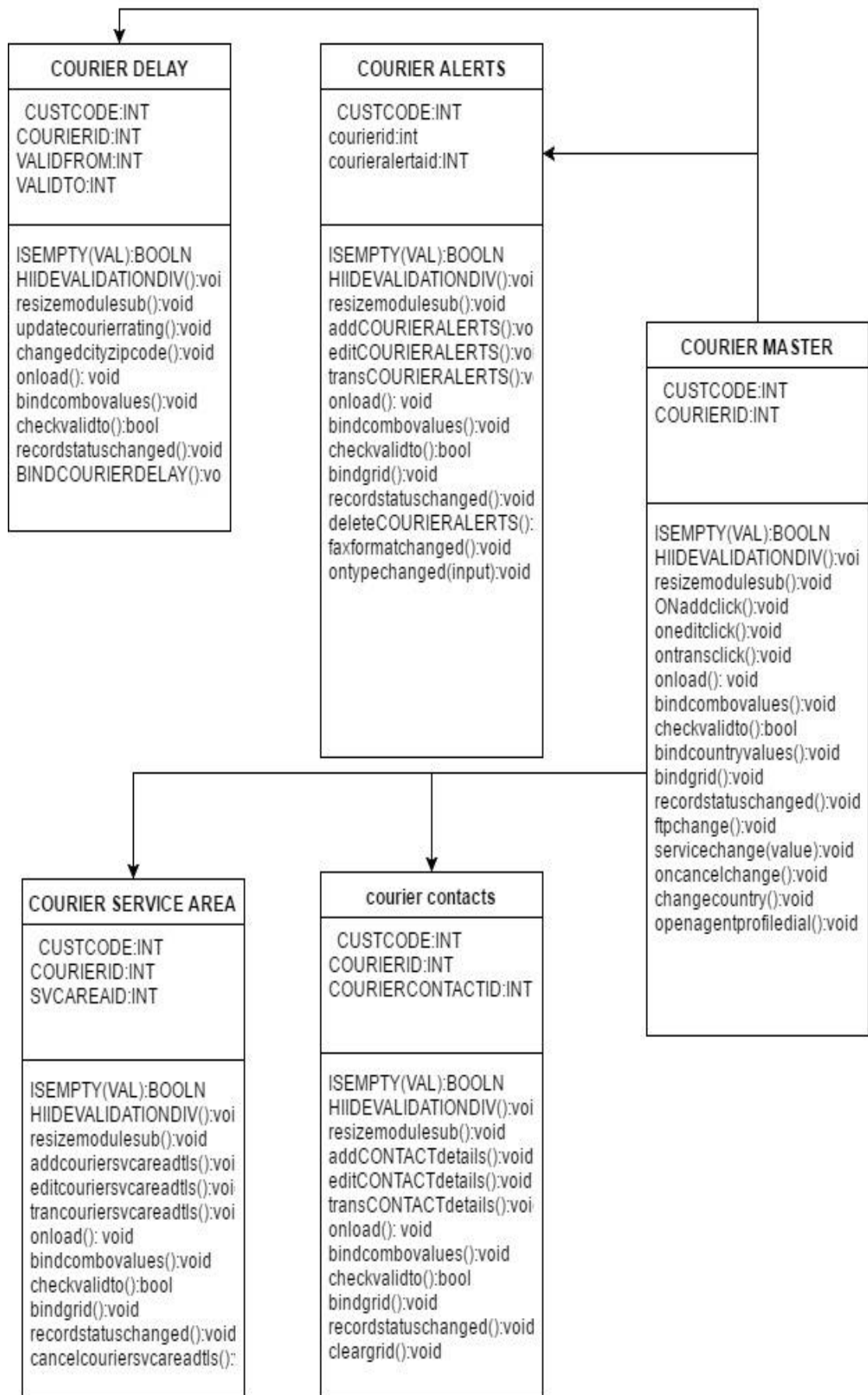### 5.1.1   Class Diagram

Figure 5.1   Class Diagram For Customer

COURIER DELAY

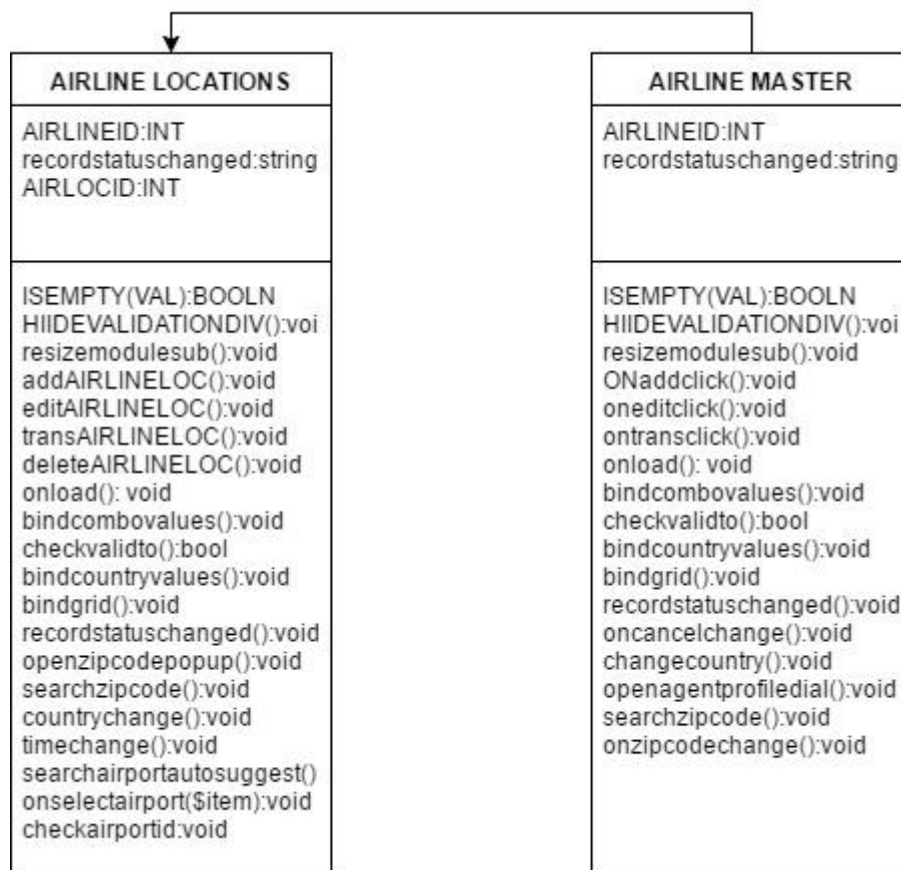CUSTCODE:INT
COURIERID:INT
VALIDFROM:INT
VALIDTO:INT

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
updatecourierrating():void
changedcityzipcode():void
onload(): void
bindcombovalues():void
checkvalidto():bool
recordstatuschanged():void
BINDCOURIERDELAY():vo

COURIER ALERTS

CUSTCODE:INT
courierid:int
courieralertaid:INT

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
addCOURIERALERTS():vo
editCOURIERALERTS():vo
transCOURIERALERTS():v
onload(): void
bindcombovalues():void
checkvalidto():bool
bindgrid():void
recordstatuschanged():void
deleteCOURIERALERTS():
faxformatchanged():void
ontypechanged(input):void

COURIER MASTER

CUSTCODE:INT
COURIERID:INT

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
ONaddclick():void
oneditclick():void
ontransclick():void
onload(): void
bindcombovalues():void
checkvalidto():bool
bindcountryvalues():void
bindgrid():void
recordstatuschanged():void
ftpchange():void
servicechange(value):void
oncancelchange():void
changecountry():void
openagentprofiledial():void

COURIER SERVICE AREA

CUSTCODE:INT
COURIERID:INT
SVCAREAID:INT

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
addcouriersvcareadtls():voi
editcouriersvcareadtls():voi
trancouriersvcareadtls():voi
onload(): void
bindcombovalues():void
checkvalidto():bool
bindgrid():void
recordstatuschanged():void
cancelcouriersvcareadtls():

courier contacts

CUSTCODE:INT
COURIERID:INT
COURIERCONTACTID:INT

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
addCONTACTdetails():void
editCONTACTdetails():void
transCONTACTdetails():voi
onload(): void
bindcombovalues():void
checkvalidto():bool
bindgrid():void
recordstatuschanged():void
cleargrid():void

Figure 5.2   Class Diagram For Courier

AIRLINE LOCATIONS

AIRLINEID:INT
recordstatuschanged:string
AIRLOCID:INT

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
addAIRLINELOC():void
editAIRLINELOC():void
transAIRLINELOC():void
deleteAIRLINELOC():void
onload(): void
bindcombovalues():void
checkvalidto():bool
bindcountryvalues():void
bindgrid():void
recordstatuschanged():void
openzipcodepopup():void
searchzipcode():void
countrychange():void
timechange():void
searchairportautosuggest()
onselectairport($item):void
checkairportid:void

AIRLINE MASTER

AIRLINEID:INT
recordstatuschanged:string

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
ONaddclick():void
oneditclick():void
ontransclick():void
onload(): void
bindcombovalues():void
checkvalidto():bool
bindcountryvalues():void
bindgrid():void
recordstatuschanged():void
oncancelchange():void
changecountry():void
openagentprofiledial():void
searchzipcode():void
onzipcodechange():void

Figure 5.3   Class Diagram For Airline

ALTERNATE AIRPORT

AIRPORTID:INT
recordstatuschanged:string
ALTAIRPORTID:INT

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
addaltairport():void
editaltairport():void
transaltairport():void
deletealtairport():void
onload(): void
bindcombovalues():void
checkvalidto():bool
bindgrid():void
recordstatuschanged():void
strstartwith(str,prefix):void
searchairportautosuggest()
onselectairport($item):void
checkairportid():void

AIRPORT MASTER

AIRPORTID:INT
recordstatuschanged:string

ISEMPTY(VAL):BOOLN
HIIDEVALIDATIONDIV():voi
resizemodulesub():void
ONaddclick():void
oneditclick():void
ontransclick():void
onload(): void
bindcombovalues():void
checkvalidto():bool
bindcountryvalues():void
bindgrid():void
recordstatuschanged():void
oncancelchange():void
changecountry():void
searchzipcode():void
onzipcodechange():void
ONAIRPORTSEARCH():voi

Figure 5.4   Class Diagram For Airport

Figure 5.5   Class Diagram For Inventory

5.1.2   Sequence Diagrams

Fig 5.2.1 sequence diagram for customer creation

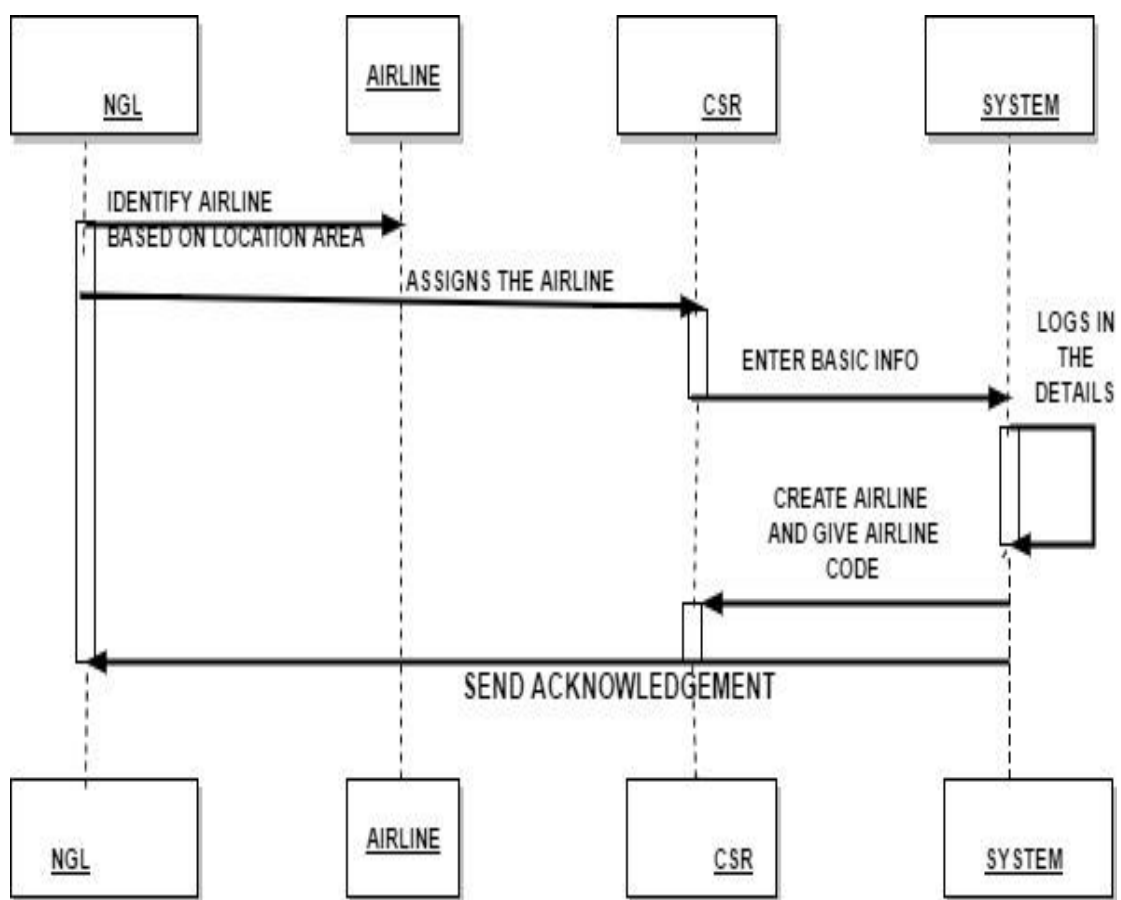Figure 5.6   sequence diagram for courier creation

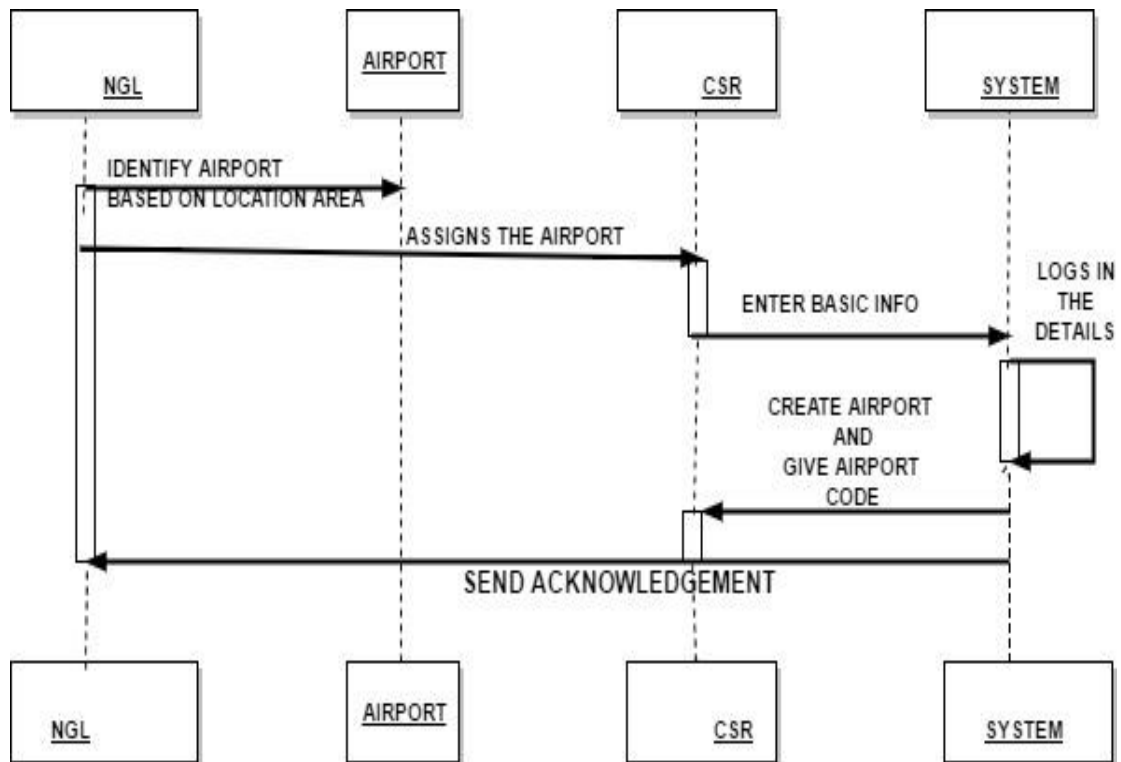Figure 5.7 sequence diagram for airline creation

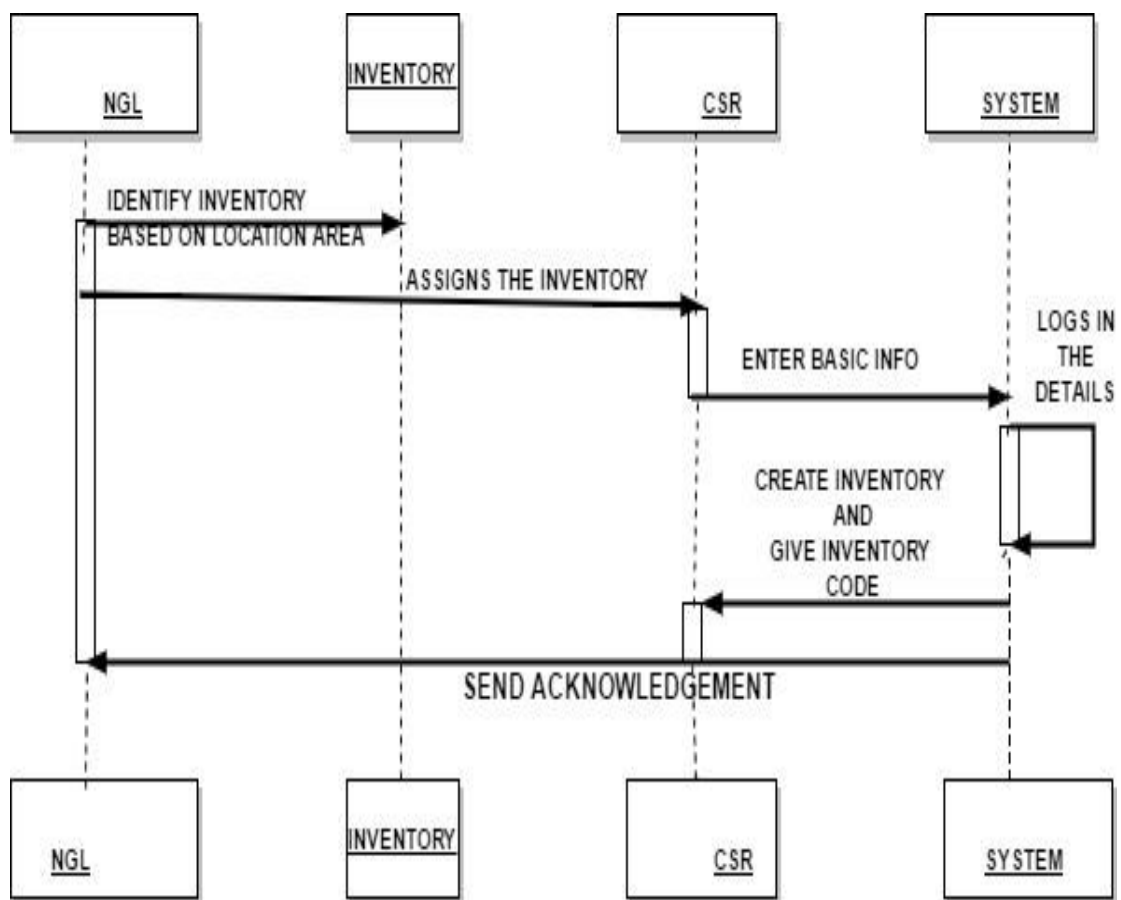Figure 5.8 sequence diagram for airport creation
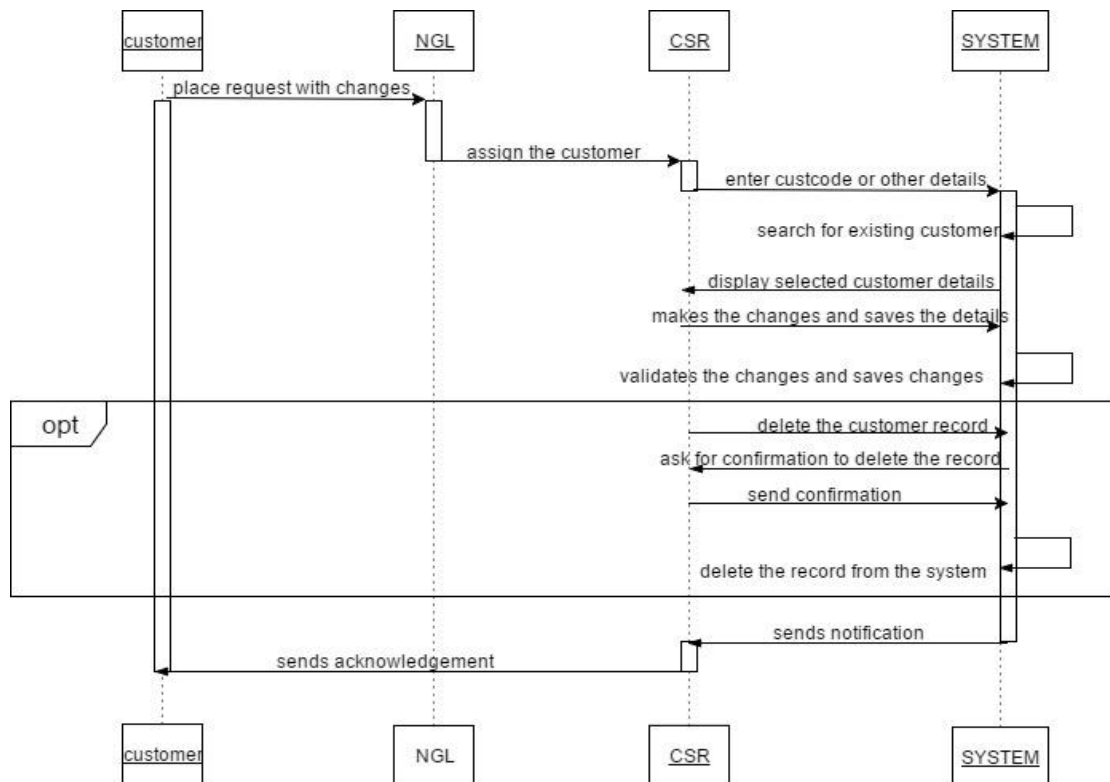

Figure 5.9 sequence diagram for inventory creation

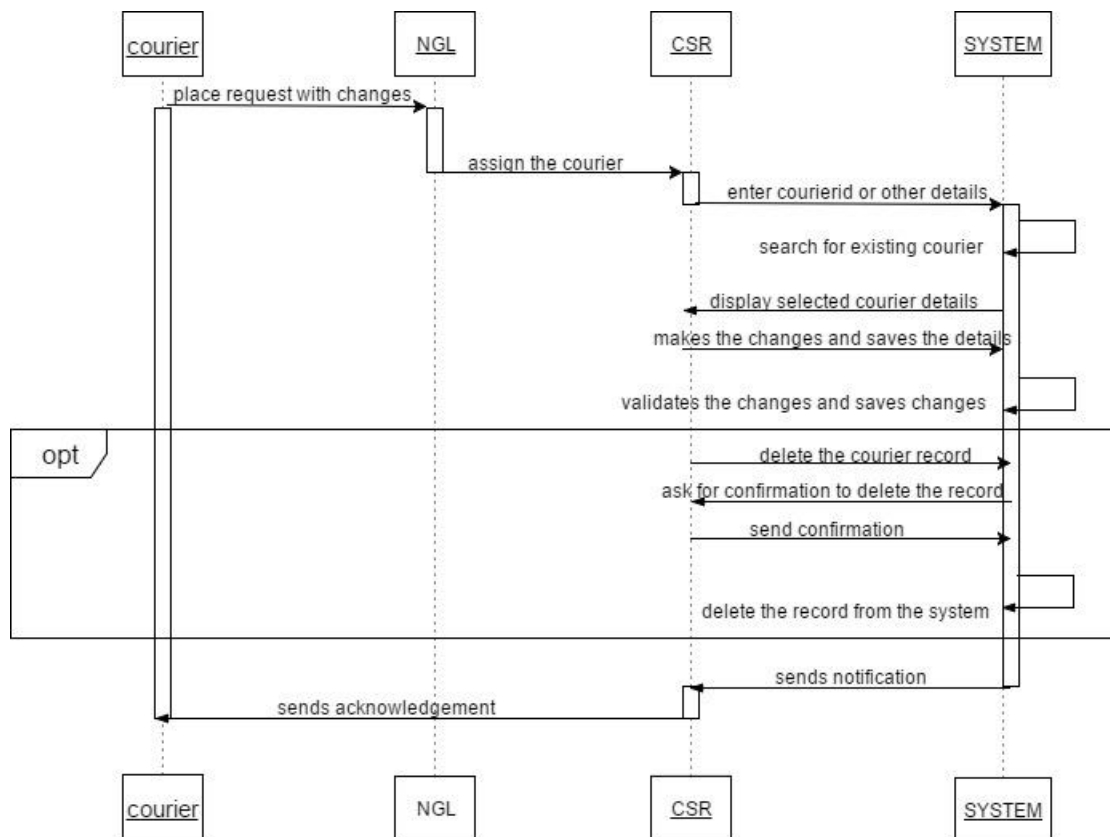Figure 5.10   sequence diagram for customer modification
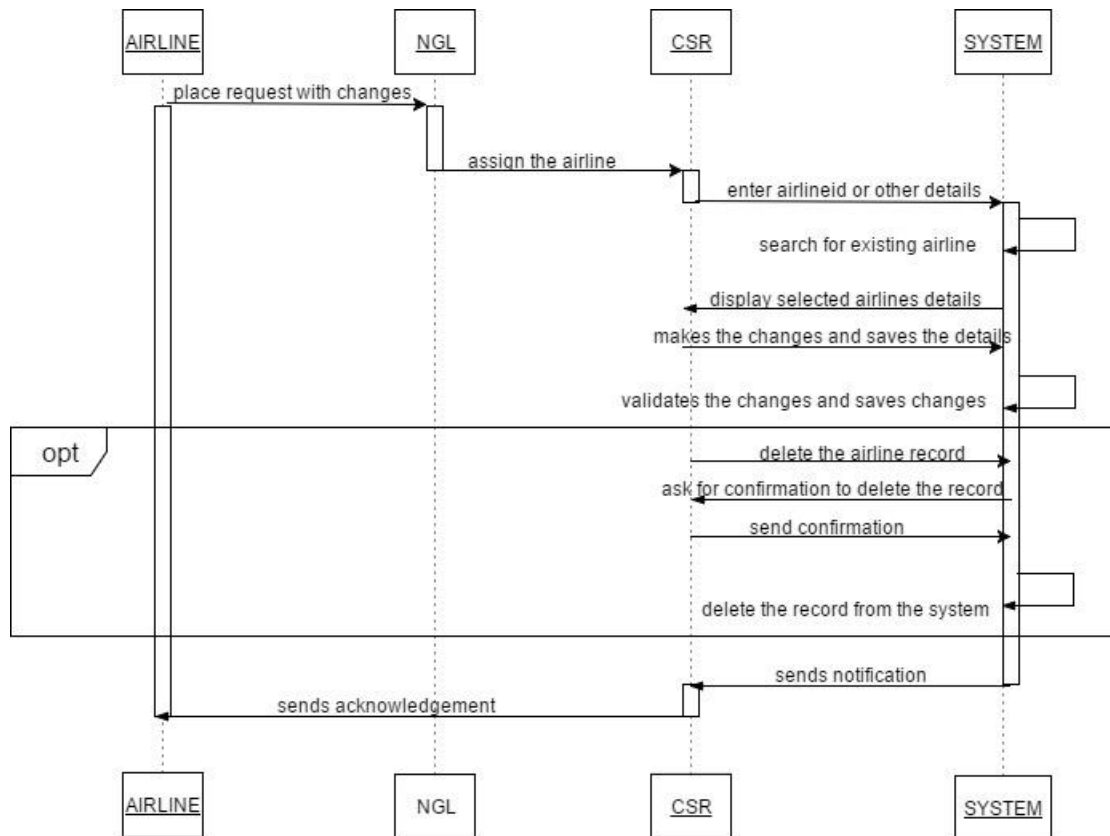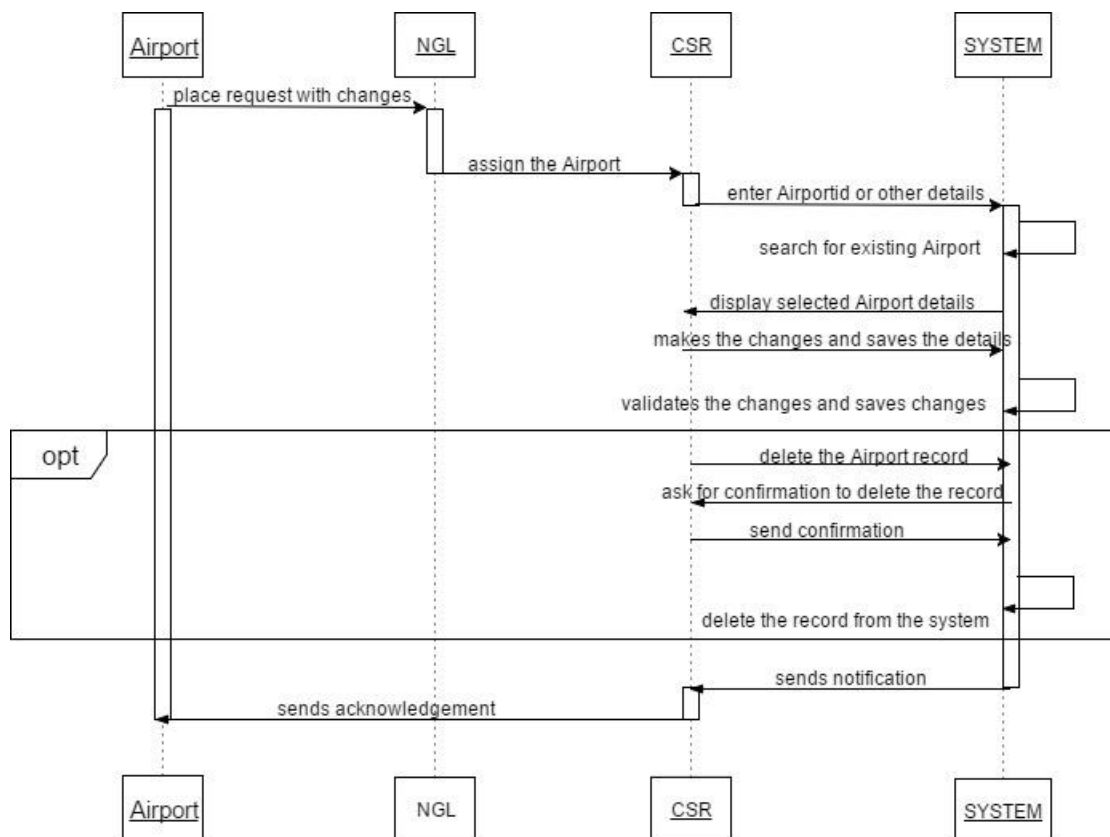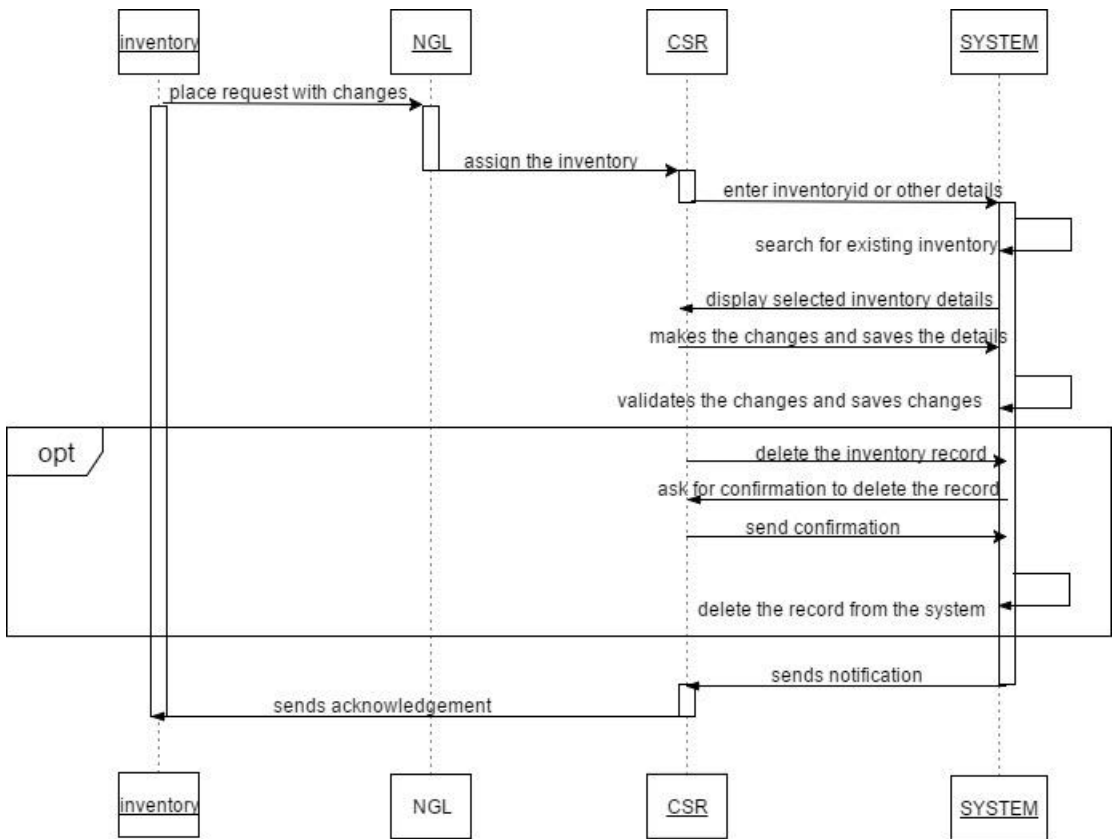

Figure 5.11   sequence diagram for courier modification

Figure 5.12   sequence diagram for airline modification

Figure 5.13   sequence diagram for airport modification

Figure 5.14   sequence diagram for inventory modification

5.1.3   Component Diagram
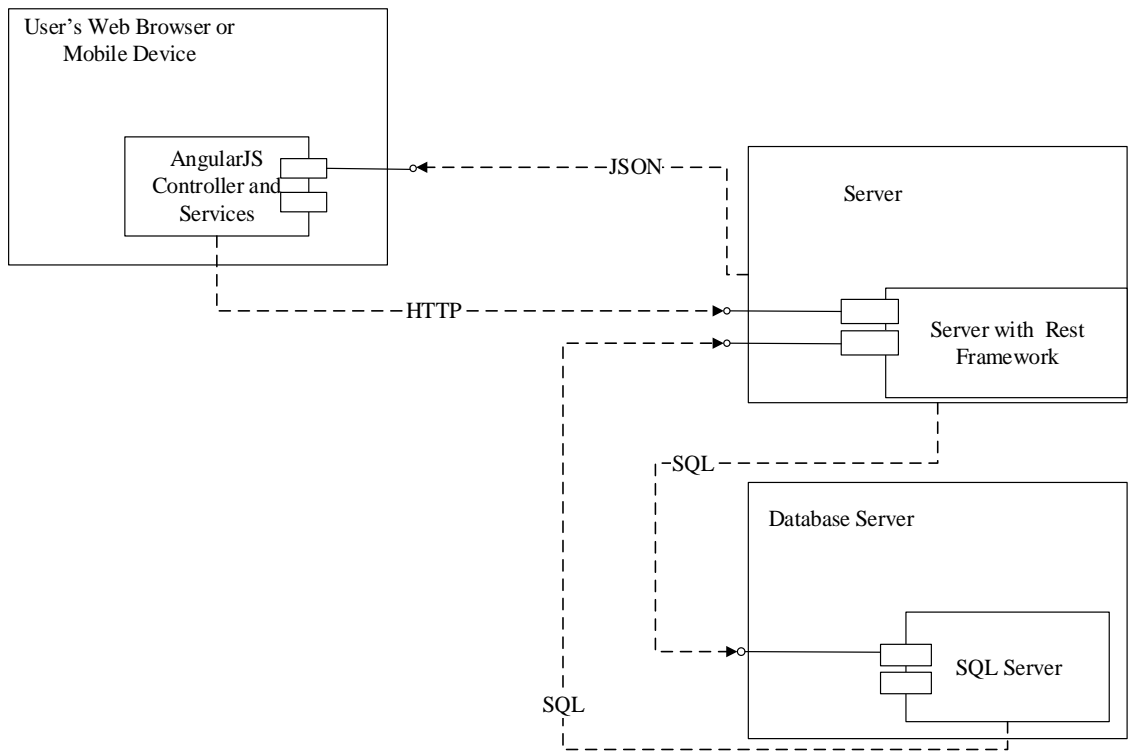
Figure 5.15   Component Diagram

5.1.4   Deployment Diagram
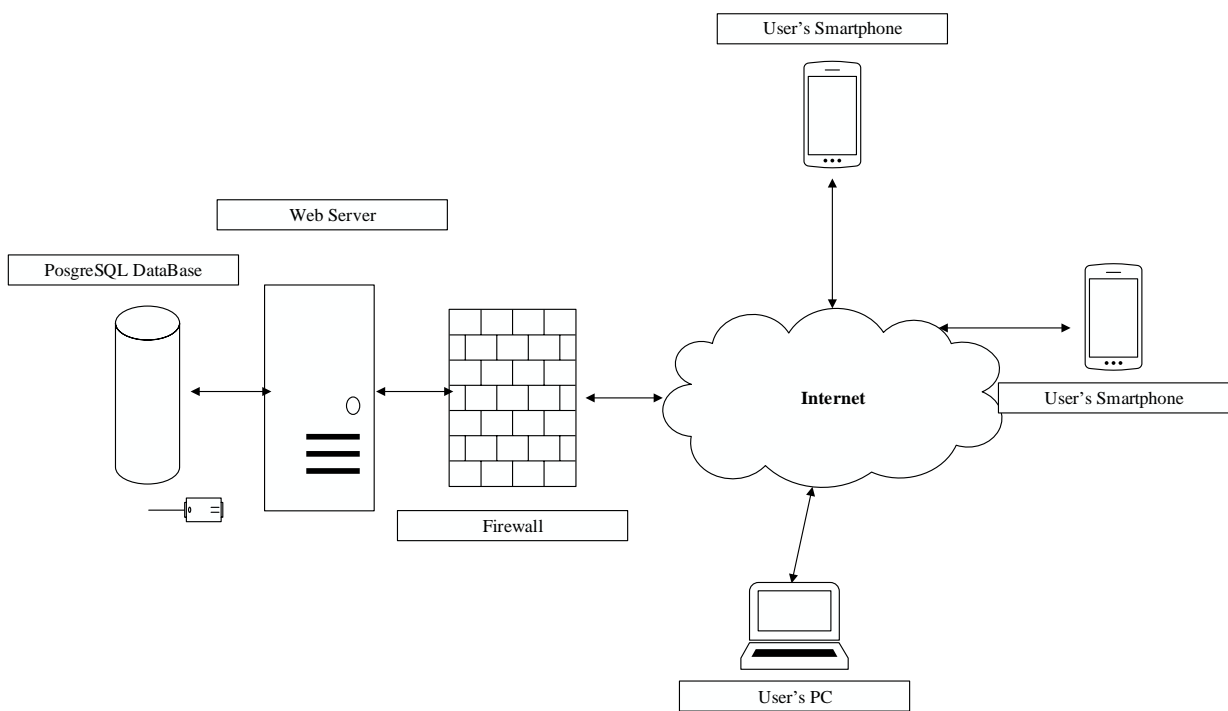
Figure 5.16   Deployment Diagram

5.1.5   State Transition Diagram
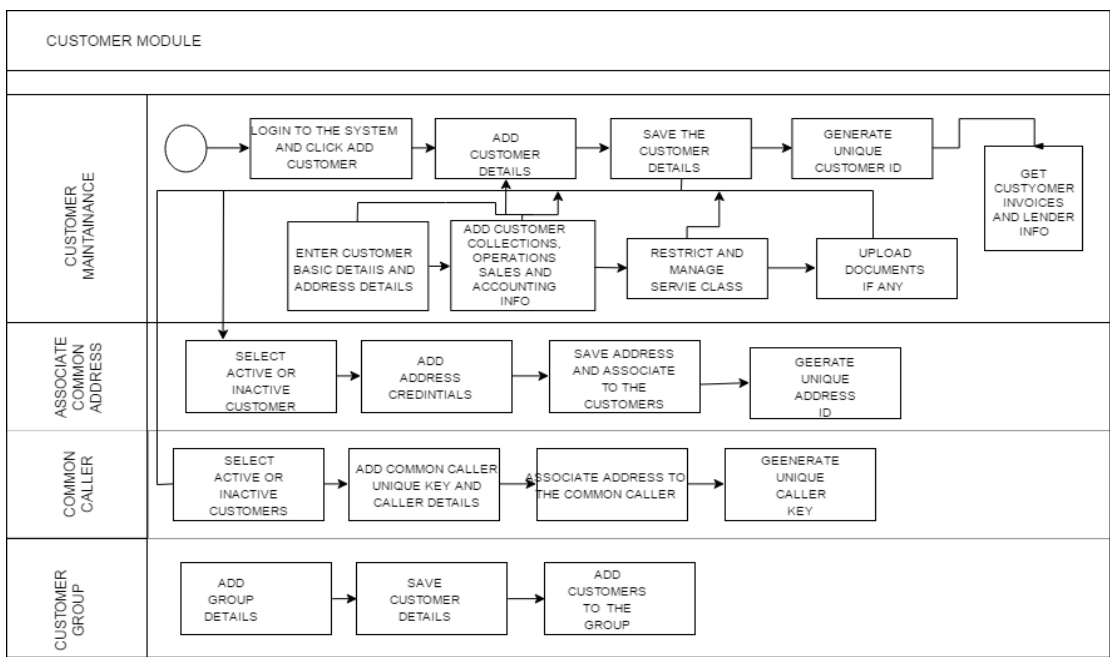
Fig 5.1.5 State Transition of customer events

5.2   DATABASE DESIGN AND/OR DATA STRUCTURE DESIGN

5.2.1   Data Dictionary

Customer:

Field Name	Data Type	Description	Requird
CustCode	varchar(20) NOT NULL,	The external customer identifier	true
NatAccountId	int NOT NULL,	National Customer Account ID	true
CompanyId	varchar(3) NOT NULL,	NCS=Network Courier FDX=FedEx	true
StationId	varchar(6) NOT NULL,	1he controlling station which will receive revenue from the customer	true
OrganizationId	int NULL,	The organization which this customer ts associated with	
ArNum	varchar(12) NULL,	The a/r account number in chart of accounts that thus client should be assigned to	
Name	varchar(50) NOT NULL,	The name of the Customer	true
CsrUserId	varchar(20) NULL,	The customer's customer service agent	
AcctCreateddt	smalldatetime NOT NULL,	The date the account was created	true
ReferTypeId	varchar(3) NULL,	The format of the reference number (required of' RefNumRequired)	
DefShipAddr	int NULL,	Default shipper address (appears when account is selected in order entry)	
ClassificationId	int NULL,	Classifies a customer, see cust_cIass table.	

StatusVia	char(1) NOT NULL,	Shipment status received by 'F'ax, 'E'mail, '\/'oice	true
StatusAddress	varchar(255) NULL,	Fax number. Email address, Phone number	
LastInvoiceDt	smalldatetime NULL,	Last date of invoice, if null then never invoiced.	
LastCallerName	varchar(50) NULL,	The name of the last caller that used this account.	
LastCallerPhone	varchar(25) NULL,	The phone number of the last caller that used this account.	
LastCallerFax	varchar(25) NULL,	The fax number of the last caller that used this account.	
Passcode	varchar(30) NULL,	The password as defined by a customer contact.	
PasscodeContact	varchar(50) NULL,	The account authorized defines the password.	
NextCrReview	smalldatetime NOT NULL,	The next account credit review. (*see note below)	true
LastActivationDttm	smalldatetime NULL,	Last Activativity date time.	
InvInfo	varchar(80) NULL,	information to prim on invoice	
RebatePer	smallint NULL,	Rebate percentage	
PayApplyType	char(1) NULL,	Payment Apply Method.	
EnteredBy	int NOT NULL,	The user who created the record	
EnteredOn	datetime NOT NULL,	The date/time the record was created.	true
InsertTzId	varchar(3) NOT NULL,	The timezone of the database server.	True
ModifiedBy	int NULL,	The user who has Modify	

		Record	
ModifiedOn	datetime NULL,	The date/time the record was Modified.	
ValidFrom	datetime NULL	The date/time from where record will Active.	

<center>Table 5.1   Customer Table</center>

Description:

Customer's details are stored in this table. This table contain basic information like of new or existing customer registered.

Indexes:

- CustCode (Primary key)

Table 5.4: Customer services

Field Name	Data Type	Description	Is Right Required
CustService Id	Int Not Null	The Unique Customer Service Identifier	true
CustCode	varchar(20) NOT NULL,	The unique customer identifier.	true
ServiceId	varchar(6) NOT NULL,	The unique service identifier	true
PodAlert	varchar(1) NOT NULL DEFAULT ('Y'),	Y=notify with POD, N=notify only if problems or late shipment.	true
PodAlertHo w	varchar(1) NOT NULL DEFAULT ('V'),	lf pod_aIer1=Y notify with POD information via voice(V). Email(B), fax(F)	true
PodAlertAd dr	varchar(255) NULL,	Phone/fax number or email address to send POD alert.	
Restricted	Varchar(1) NOT NULL	Y=Yes, N=No	true

	DEFAULT ('A'),	Default (No).	
PodNote	varchar(255) NULL,	Note: which appears on the POD call back window.	
ConseeConf	varchar(1) NOT NULL DEFAULT ('N')	Y-confirm delivery with consignee, N=no confirm	true

Table 5.2   Customer services

Description:

This table contains the information about the services required by the customer.

Indexes:

- CustServiceId (Primary key)
- CustCode   (Foreign key) has relation with Customer table

Table 5.6: Customer Special Instructions

Field Name	Data Type	Description	Is Right Required
CustCode	varchar(20) NOT NULL,	The external customer identifier	true
SeqNo	int NOT NULL,	The sequence number of the instruction.	true
InstructionTypeId	int NOT NULL,	The type of instruction.	true
Description	varchar(255) NULL,	Description of the instruction given.	
EnteredBy	varchar(50) NOT NULL,	The CSR agent name,	true
EnteredOn	datetime NULL,	Date and time of record entered.	
ModifiedBy	varchar(50) NULL,	The name of agent who last modified the record.	
ModifiedOn	datetime NULL,	Date and time of record last modified.	

			Is Right Required
ValidFrom	smalldatetime NOT NULL,	The valid from date above today's date.	true
ValidTo	smalldatetime NULL,	The date upto which the order will be valid.	

Table 5.3   Customer Special Instructions

Description:

This table contains the information about any special instructions that are needed to be given on any order .

Indexes:

- InstructionTypeId   (Primary key)
- CustCode   (Foreign key) has relation with Customer table

Table 5.7: Customer Documents

Field Name	Data Type	Description	Is Right Required
CustDocId	int IDENTITY(1,1) NOT NULL,	The unique identity number of doc.	true
CustCode	varchar(20) NOT NULL,	The unique custcode of each document.	true
Name	varchar(50) NOT NULL,	Name of customer.	true
FileName	varchar(255) NOT NULL,	Filename being uploaded by the customer.	true
DocDate	smalldatetime NULL,	Date on which the doc was uploaded.	
EnteredBy	varchar(50) NOT NULL,	The name of the agent who is entering the info.	true
EnteredOn	datetime NULL,	Date and time on which entered.	
ModifiedBy	varchar(50) NULL,	Name of agent who modified the record.	
ModifiedOn	datetime NULL,	Date and time of record last modified .	

Table 5.4   Customer Documents

Description: This table contains the information about the coustomer documents uploaded on the server.

Indexes:

- CustDocId   (Primary key)
- CustCode   (Foreign key) has relation with Customer table

Courier database:
Table 5.8: Courier Master

Field Name	Data Type	Description	Is Right Required
CourierKey	char(10) NOT NULL,	Unique courier key. *see note	true
VendorKey	char(10) NULL,	The courier vendor identifier	
CourierId	int IDENTITY(30001,1) NOT NULL,	The unique courier identifier.	true
Name	varchar(30) NOT NULL,	the name of the courier.	true
AddrLine 1	varchar(255) NULL,	First line of the address.	
AddrLine 2	varchar(255) NULL,	Second line of the address.	
City	varchar(50) NOT NULL,	City name	true
StateAbbr	char(2) NOT NULL,	The state.	true
ZipCode	char(10) NOT NULL,	The zip code. "INTL" for international address.	true
CourierType	char(1) NOT NULL DEFAULT ('2'),	1=Employee,2=Agent, 3-Contractor	true
EnteredB	varchar(50) NOT NULL DEFAULT	The user who	true

y	(user_name()),	created the record	
EnteredOn	datetime NOT NULL DEFAULT (getdate()),	The date/time the record was created.	true
EnteredTzId	char(3) NOT NULL DEFAULT ('PDT'),	Since the insert date/time is created by the database, this is the timezone of the database.	true
ModifiedBy	varchar(50) NULL,	The user who has Modify Record	
ModifiedOn	datetime NULL,	The date/time the record was Modified	
ValidFrom	smalldatetime NULL,	The date/time from where record will Active.	
ValidTo		smalldatetime NULL,	
CountryId	char(3) NOT NULL DEFAULT ('USA'),	The country identifier.	true

Table 5.5   Courier master

Description:

This table contains the information about the services required by the courier master.

Indexes:

- CourierId (Primary key)

TABLE 5.10:Courier Alerts

Field Name	Data Type	Description	Is Required	Right
AlertId	int(4)	Unique Alert identifier.		
AlertTypeId	int(4)	Alert Type for pickup, Delivery, Job		

AlertCodeId	int(4)	Unique Identifier for pickup, Delivery, Job	
CourierId	int(4)	Unique identification number for courier.	
AlertStatus	int(4)	The status of the alert.	
AlertUser	nvarchar(20)NULL	User who alerted dispatcher, Customer, driver , courier	
AlertWho	nvarchar(50)NULL	Name or description (dispatcher) of person alerted.	
	int(4)NULL	The communication type used to alert the person.	
AlertVia	nvarchar(255) NULL	The key for the method used to alert (alert_how). Phone #, Email address, fax #, etc.	
AlertAccepted	datetime(8)NULL	Date/time the courier (agent), driver, shipper, consignee accepted the alert.	
Mileage	decimal(21,2) NULL	Milage calculated as per the standard method.	
AttemptCount	smallint(2)NULL	Counter for no. of attempts made for delivery	
WaitTime	smallint(2)NULL	Amount of time customer had to wait before the delivery of package.	
Pieces	smallint(2)NULL	No.of pieces in courier.	
Weight	decimal(10,2) NULL	Weight of courier	
AlertDttm	datetime(8)NULL	Date and time courier (agent), driver, shipper, consignee was alerted. (local time at dispatch center).	
EnteredTzId	char(3) NOT NULL	Since the insert date/time is created by the database, this is	true

	DEFAULT ('PDT'),	the timezone of the database.	
EnteredBy	varchar(20) NOT NULL DEFAULT (user_name()),	The user who created the record	true
EnteredOn	datetime NOT NULL DEFAULT (getdate()),	The date/time the record was created.	true
ModifiedBy	varchar(20) NULL,	The user who has Modify Record	
ModifiedOn	smalldatetime NULL,	The date/time the record was Modified.	
ValidFrom	datetime NULL,	The date/time from where record will Active.	
ValidTo	datetime NULL,	The date/time of record when record will expire.	

Description:

This table contains the information about the services required by the courier to add customer alerts information.

Indexes:

- AlertId (Primary key)
- courierid (Foreign key) has relation with courier table

Table 5.11: AlernateAirport

Field Name	Data Type	Description	Is Right Required
AirportId	char(4) NOT NULL,	Unique identification number for airport	true
AltAirportId	char(4) NOT NULL,	Id for alternate airport in case of emergency.	true

Selected	char(1) NOT NULL DEFAULT ('N'),	Is the item selected for special delivery.	true
EnteredBy	varchar(50) NULL,	The user who created the record	
EnteredOn	datetime NULL,	The date/time the record was created.	

Description:

This table contains the information about the services required by alternate airport table of airport module for the customer.

Indexes:

- AltAirportId (Primary key)
- Airportid   (Foreign key) has relation with Customer table

Table 5.12: Cycle Count Configuration

Field Name	Data Type	Description	Is Required	Right
CustCode	varchar(20) NOT NULL	Customer for which cycle count is created.	true	
FSLId	bigint(8) NOT NULL	FSL for which cycle count is configured.	true	
ClassCode	int(4) NOT NULL	Identify Part characteristics.	true	
CountPerctg	int(4) NULL	Cycle Count on Percentage of Parts.		
CountMovement	int(4) NULL	Identify Movement of parts.		
CCFrequency	int(4) NULL	How frequantly cycle Count can be generated.		
FreqDuration	int(4) NULL	Duration for configure Cycle Count frequantly.		
NextRunDate	datetime(8) NULL	Next run date		
EnteredBy	nvarchar(50) NULL	The user who created		

		the record	
EnteredOn	datetime(8) NOT NULL DEFAULT (getdate()),	The date/time the record was created.	true
ModifiedBy	nvarchar(50) NOT NULL	The user who has Modify Record	true
ModifiedOn	datetime(8) NOT NULL DEFAULT (getdate()),	The date/time the record was Modified.	true
ValidFrom	datetime(8) NOT NULL DEFAULT (getdate()),	Record valid from date	true
ValidTo	datetime(8) NULL	Record valid to date	

Description:
This table contains the information about the services required by the cycle count configuration screen of the customer.
Indexes:
- FSLId (Primary key)
- Inventoryid   (Foreign key) has relation with Customer table

Table 5.13: Customer department info

Field Name	Data Type	Description	Is Right Required
AgingReportType	int NOT NULL,	Unique AgingReportType number.	true
AgingReportEmail	int NOT NULL,	The master billing cycle AgingReportEmail.	true
CutOffAmount	int NULL,	Cutoff amount to clear the billing cycle.	
NextAgingDate	varchar(20) NOT NULL,	Customer NextAgingDate identifier date.	true
LastShipDt	smalldatetime NOT NULL,	The date of the invoice (aging starts?)	true
defSVCList	smalldatetime NOT NULL,	The date that the list contains true	
FTPURL	decimal(16,2) NOT NULL	The url for ftp upload.	true

Field Name	Data Type	Description	Is Right Required
	DEFAULT ((0)),		
FTPUserName	decimal(16,2) NOT NULL DEFAULT ((0)),	The username for ftp upload.	true
FTPPassword	decimal(16,2) NOT NULL DEFAULT ((0)),	The password for ftp upload.	true
FTPCodeId	smalldatetime NULL,	The codeid for ftp upload.	
EnteredBy	int NOT NULL,	The user who created the record	true
EnteredOn	datetime NULL,	The date/time the record was created.	
ModifiedBy	int NULL,	The user who has Modify Record	
ModifiedOn	datetime NULL,	The date/time the record was Modified.	
ValidFrom	datetime NULL,	The date/time from where record will Active.	
ValidTo	datetime NULL	The date/time of record when record will expire.	

Description:
This table contains the information about the services required by the customer department info. Screen to add customer details.
Indexes:
- AgingReportType (Primary key)
- CustCode   (Foreign key) has relation with Customer table

Table 5.14: Airline Locations

Field Name	Data Type	Description	Is Right Required
AirlineId	int NOT NULL,	The unique airline identifier.	true

AirportId	char(4) NOT NULL,	Unique airport identifier.	true
LocationType	int NOT NULL,	Type of location where the parsel is to be delivered.	true
Location	varchar(100) NULL,	Location address of delivery.	
Phone	varchar(25) NULL,	Customer contact phone	
Contact	varchar(50) NULL,	Customer contact phone	
Hours	varchar(200) NULL,	Hours required for delivery.	
Note	varchar(200) NULL,	Any note attached along with the delivery	
Alertnote	varchar(255) NULL,	Any alert note attached like handle with care along with the delivery	
Tracing	varchar(25) NULL,	Tracing of driver date and time of delivery.	
Fax	varchar(25) NULL,	Fax address if any.	
Droptime	int NULL,	Time at which courier is dropped.	
Rectime	int NULL,	Time at which courier is recovered from airport.	
AddrLine1	varchar(30) NULL,	Address of customer.	
AddrLine2	varchar(30) NULL,	Address of customer.	
CityName	varchar(30) NULL,	City name.	
StateAbbr	char(2) NULL,	Abbravation for state.	
ZipCode	char(10) NULL,	City zipcode/postalcode.	
EnteredBy	varchar(50) NOT NULL DEFAULT (user_name()),	The date/time the record was created.	true
EnteredOn	datetime NOT NULL DEFAULT	Since the insert date/time is created by the database, this is	true

	(getdate()),	the timezone of the database.	
EnteredTzId	char(3) NOT NULL DEFAULT ('PDT'),	Since the insert date/time is created by the database, this is the timezone of the database.	true
ModifiedBy	varchar(50) NULL,	The user who has Modify Record	
ModifiedOn	datetime NULL,	The date/time the record was Modified.	
ValidFrom	smalldatetime NULL,	The date/time from where record will Active.	

Description:This table contains the information about the services required by the airline locations screen in airline module of customer.

Indexes:

- AirlinelocID (Primary key)
- AirlineID (Foreign key) has relation with Customer table

6.0   IMPLEMENTATION PLANNING

6.1   IMPLEMENTATION ENVIRONMENT

The .NET Framework is a technology that supports building and running the next generation of applications and XML Web services. The .NET Framework is designed to fulfill the following objectives:

1. To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
2. To provide a code-execution environment that minimizes software deployment and versioning conflicts.
3. To provide a code-execution environment that promotes safe execution of code, including code created by an unknown or semi-trusted third party.
4. To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
5. To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.
6. To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code.

The .NET Framework consists of the common language runtime and the .NET Framework class library. The common language runtime is the foundation of the .NET Framework. You can think of the runtime as an agent that manages code at execution time, providing core services such as memory management, thread management, and remoting, while also enforcing strict type safety and other forms of code accuracy that promote security and robustness. In fact, the concept of code management is a fundamental principle of the runtime. Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged code. The class library is a comprehensive, object-oriented collection of reusable types that you can use to develop applications ranging from traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services.For example, ASP.NET hosts the runtime to provide a scalable, server-side environment for managed code. ASP.NET works directly with the runtime to enable ASP.NET applications and XML Web services, both of which are discussed later in this topic.

AngularJS is an open-source web application framework maintained by Google and a community of individual developers to address many of the challenges encountered in developing single-page applications. The library works by first reading the HTML page, which has embedded into it additional custom tag attributes. Those attributes are interpreted as directives telling Angular to bind input or output parts of the page to a model that is represented by standard JavaScript variables. The values of those

JavaScript variables can be manually set within the code, or retrieved from static or dynamic JSON resources.

NGL connect is GUI based and multiple-user system so multiple users can access the Application paralleled and through the internet.

6.2 PROGRAM/MODULES SPECIFICATION

6.3 CODING STANDARDS

The following is the Development checklist and coding conventions followed while developing the application:

Completeness	
1	Are the naming conventions followed for all the variables, classes, properties and methods according to style guide?
2	Is the scope of the variables well-defined? i.e. at application,page,services and procedure level?
3	Are the code commenting conventions followed properly? E.g. proper commenting for properties, methods and class overview.
4	Is the code a complete and exact implementation of the functionality as documented in the Functional Specifications?
5	Are there any unreferenced or undefined variables, constants, or data types?
6	Does the DoU and Functional specs both referred to while coding.
7	Has any assumptions been made? If so, are they notified?
8	Are there any open issues or things pending for the page? If so, are they notified?
Consistency	
1	Are there common objects (eg.Data Object, Paging Object,Layout object, Popup, controls etc) used throughout?
2	Is the Navigation Interface consistent for all screens?
3	Is a proper Stylesheet followed for background,alignment of controls and the overall look and feel of all the screens? Prefer should be light colors.
4	Is the code logically consistent with the Functional Specifications?
5	All labels are coming from Resource files, no label are hardcoded. If any are they mentioned?
6	We have to add Refresh Button as per other buttons

Correctness	
1	Does the code conform to specified standards?
2	Are all comments complete and accurate,corresponding to the logic being described?
3	Are all proper input and output parameters used in the code?
4	Are there any general SQL statements being used? E.g. select\insert\update\delete in the code
5	Password TextBox always set in PassWord Mode
Robustness	
1	Does the code protect against detectable runtime errors like range array index values, division by zero, out of range variable values, and stack overflow?
2	Are error-handling routines present.? Is proper Redirection provided on page crash?
Traceability	
1	Does the code identify each function name uniquely ?
2	Has the coding methodology as decided been followed?
3	Has the code been put under VSS?
4	Does the code contains a revision history of all code modifications and the reason for them ? E.g. Commenting while checking in and out of VSS.
5	Does the code contains proper commenting for tracking the changes done in it?
6	All code respective Sps/DDL/DML are added into VSS.
Understandability	
1	Is ambiguous or unnecessarily complex coding used? If so, are they clearly commented ?
2	Were consistent formatting techniques (e.g., indentation, use of white space) used to enhance clarity?
3	In Memo screen, we have to give message for set Enable SilverLight Storage check box in Browser.
Reusability	
1	Are the procedures,functions/routines,etc so developed that they could be re-used? i.e. A true modular approach followed?
2	Are the function libraries used? i.e. all common, reusable and customised functions placed in a common file, to be included whenever the function needs to be called?

Testability	
1	Are there any misleading or empty links on the page?
2	Has the code been reviewed and verified that there is no hard coding for paths for image, files and any variables?
3	Is the display/tab order for all fields/columns/combos checked?
4	Has the code been debugged for syntax or any other runtime errors?
5	Have all the test cases for functionality, security etc. completed and filled?
6	Is compatibility with old data tested?
7	Has the code been tested for data containing special characters and HTML tags?
8	Is the code tested for browser(IE, FireFox, Safari) compatibility?
9	Are all pages Title and labels are grammatically correct sentences ?
10	As per sql Injection Issue below operators are not allowed in Name field.(For ex.&,%)

Table 6.1   Development Checklist

7.0 TESTING

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. Testing is a vital part during the course of software development. Testing helps us understand the flaws in the system and hence enhance the system for a better user experience. In the course of software development testing usually occupies anywhere between 20-40 % of the effort and resources. Software testing is both a discipline and a process. Though software testing is part of the software development process, it should not be considered part of software development. It is a separate discipline from software development. Software development is the process of coding functionality to meet defined end- user needs. Software testing is an iterative process of both validating functionality, and, even more important, attempting to break the software. The iterative process of software testing consists of

- Designing tests
- Executing tests
- Identifying problems

7.1 TESTING PLAN

We follow the following unit testing guidelines followed while testing the application:

1	Spelling mistake in sentences
2	Are all pages Title and labels are grammatically correct sentences ? And is it coming from resources?
3	Alignment of page is proper else while development send change request to Designer.
4	Consistent theme of all pages including pop-up.
5	Check any control is in page, which we are not using.
6	Check any font is in bold which is not consistent
7	Check in all page same font family is using.
8	Is header and footer is correct on Page and on popup if requires.
9	Is any menu or any mouse hover disturb design or it is dancing.
10	Is any javascript error is coming in browser?
11	Is the code tested for browser(IE, FireFox, Safari) compatibility?
12	Is the display/tab order for all fields/columns/combos checked? It should not go to labels.
13	Max length is set to all textbox wherever requires.
14	Is Validation checked to respective fields and proper message displaying?
15	Are all messages displaying from resources?
16	Are there any misleading or empty links on the page?
17	Has the code been reviewed and verified that there is no hard coding for paths for image, files and any variables?
18	Has the code been tested for data containing special characters (') and HTML tags (<input />) ?
19	Is any hidden field displaying on UI?
20	Check space between two word in validation summary.
21	Are fields reset if pages is already opened and user tries to open same page using

	Navigation menu.
22	Popup should open in center or consistent within the system.
23	Include common regular expression like email, currency, website, phone, date time etc with Mask if require. e.g. Currency should be seprate by comma if it is thoushand
24	In Copy/paste also validation should work proper.
25	While delete there should be confirmation message including grid.
26	On page load proper focus on field should happen
27	Default date 01/01/0001 should be handled properly.
28	In search on page load are you displaying sorting on specific fields?
29	In search text box, if we click enter then search should perform and button should highlighted as default button
30	ALL DROP DOWN list MUST have this functionality to select value by typing characters in for the value to select. User does not have to mouse click to select value.
31	Any time If we perform or press Search, old search result should be reset
32	Whatever value we are inserting it should be in capital, so in fetch we can see in consistent way.
33	Every page respective to DML should have audit trail.
34	Is page slow, if yes then inform the project manager.
35	Check fault in every service Call back if fault in not null then display fault message in Message-box.

Table 7.1   Unit Testing Guidelines
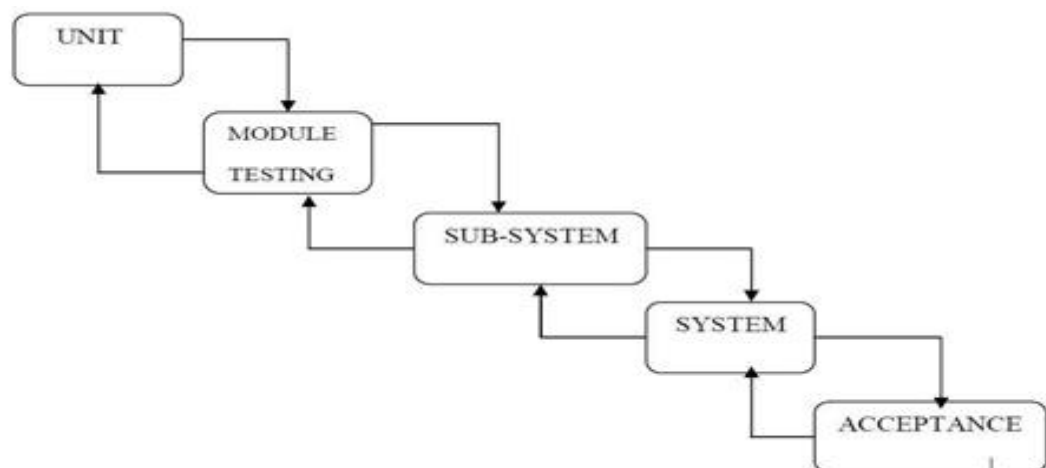
Figure 7.1 Test Plan

7.2   TESTING STRATEGY

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

7.3   TESTING METHODS

White Box Testing:

Throughout the development phase of the project life cycle, white box testing was carried out on a continuous basis. Screen messages were added to appear at several points so it was always clear at which point the code was while the application was being run.

Module Testing:

In a system each module in developed individually and each module is tested separately and the result is integrated. We have tested each small module like a face detection, switching on the screen and switching it off again.

Integration Testing:

It is clear that certain errors, which are related to the integration of different program modules, cannot be detected by unit testing. Such errors only are detected by an integrated test. The process by which individual modules are put together to realize major sub sections and functions of a program is known as a system integration. When tests are performed which exercises interfaces among modules this is known as integration. The number of instructions coded and tested or the number of functions or modules implemented and tested often measures the progress of IT.

Regression Testing:

After we made some changes in one module, we had to check whether older modules were working perfectly or not.

Install / uninstall Testing:

At the end of project, to create the setup of the project we tried to run on browser. It

works perfectly.

7.4 TEST CASES

Purpose
    The test case serves multiple purposes:
- To check the functionalities of the application
- To make sure the application functions smoothly under all cases possible
- The test cases are designed to encompass all usages requirements of the application

Test Cases with Required Input and Expected Output
A typical test case for customer master screen is shown as follows and similar kind of regerous testing has been carried out throughout the project for all the screens made by us.

Test Case ID: cust_1	Test Designed by: Kruti,Priyanka.
Test Priority (Low/Medium/High): Med	Test Designed date: 15/03/2017
Module Name: Customer master Module	Test Executed by: Kruti,Priyanka.
Test Title: Verify file types	Test Execution date: 15/03/2017
Description: Test customer master.	
Pre conditions: User must be have a valid login ID and password of NGL connect.	

Test Case Id	Test Summary	Test Description	Test Steps	Expected Result	Actual Result	Status
GUI CHARECTERISTICS						
BASIC INFORMATION						
CUST_001	Customer Code	Nil	Nil	Search text box	As Expected	Pass
CUST_002	Fed Ex Code	Nil	Nil	Text Type = String; Minimum=1; Maximum =20; Not Null = False;		Pass
CUST_004	Name	Nil	Nil	Text Type = String; Minimum=1;	As Expected	

				Maximum =100; Not Null = True;		
CUST_0 05	Compan y	Nil	Nil	Combo box; Not Null = True;	As Expect ed	Pass
	Known Shipper	Nil	Nil	Check box; By default this check box shows disabled. Check box will get enable when particular customer has created 4 job.		
CUST_0 07	National Account	Nil	Nil	Combo Box Not Null = False;	As Expect ed	Pass
CUST_0 08	Station	Nil	Nil	Combo Box Not Null = False ;		Pass
CUST_0 10	Valid From	Nil	Nil	Date Picker, Date Time Not Null = True;	As Expect ed	
CUST_0 11	Valid To	Nil	Nil	Disable Date Picker,	As Expect ed	Pass
CUST_0 12	Collectio n	Nil	Nil	If this check box is checked to true then The flag collection will be updated to 'Y' in Data Base.		Pass
CUST_1 4	First Name	Nil	Nil	Text Type = String; Minimum=1; Maximum =50; Not Null = True;	As Expect ed	Fail
CUST_1 5	Last Name	Nil	Nil	Text Type = String; Minimum=1; Maximum =50; Not Null = False;	As Expect ed	Fail

CUST_1 6	Contact Title	Nil	Nil	Text Type = String; Minimum=1; Maximum =20; Not Null = False;	As Expect ed	Fail
CUST_1 7	Phone	Nil	Nil	Masked text box Minimum = 1; Maximum = 25; Not Null = False;	As Expect ed	Fail
CUST_1 8	Email	Nil	Nil	Text Type = String; Minimum=0 Maximum =255; Not Null = False;	As Expect ed	Fail
CUST_1 9	FAX	Nil	Nil	Masked text box Minimum = 1; Maximum = 25; Not Null = False;	As Expect ed	Fail
CUST_2 0	Country	Nil	Nil	Combo Box Not Null = True;	As Expect ed	
CUST_2 1	ZipCode	Nil	Nil	Search text box; Text Type = String; Minimum=1; Maximum =10; Not Null = True;	As Expect ed	
CUST_2 2	City	Nil	Nil	Combo box; Not Null = True;	As Expect ed	
CUST_2 3	State	Nil	Nil	Combo Box Not Null = True;	As Expect ed	
CUST_2 7	Fax	Nil	Nil	Masked text box Not Null = False;	As Expect ed	
CUST_3 0	Status	Nil	Nil	Combo Box Not Null = True;	As Expect ed	Pass
CUST_3 1	Account Open	Nil	Nil	Date Picker, Date Time Not Null = False;	As Expect ed	Pass

CUST_34	Last Payment	Nil	Nil	Disable Date Picker,	As Expected	Pass
CUST_36	Base TTC	Nil	Nil	Text box max=4 Not Null =False ;	As Expected	Pass
CUST_37	Average balance	Nil	Nil	It will display average balance of the customer ,disable	As Expected	Pass
CUST_41	Send Statement	Nil	Nil	Combo Box Not Null = False;	As Expected	
CUST_44	Inv Cycle	Nil	Nil	Combo Box Not Null = True;	As Expected	Pass
CUST_46	Auto Verify	Nil	Nil	if this check box is checked to true then The flag Original QDT will be updated to 'Y' in Data Base.	As Expected	Pass
CUST_47	Inv Break	Nil	Nil	Combo Box Not Null = False;	As Expected	Pass
CUST_49	Pay Term	Nil	Nil	It accept only numeric value Not Null=False;	As Expected	Fail
CUST_52	Bill Group	Nil	Nil	Combo Box Not Null = False;	As Expected	Fail
CUST_53	Last Invoice	Nil	Nil	Date Picker, Date Time Not Null = False;	As Expected	Fail
CUST_54	Next Review	Nil	Nil	Date Picker, Date Time Not Null = False;	As Expected	Fail
CUST_55	TargetId	Nil	Nil	It accept only numeric value Not Null=False;	As Expected	
CUST_60	Proj Rev Date	Nil	Nil	Date Picker, Date Time Not Null = False;	As Expected	

CUST_61	Next Credit Date	Nil	Nil	Date Picker, Date Time Not Null = False;	As Expected	
CUST_64	No of Last Order	Nil	Nil	It accept only numeric value, Minimum=0; Maximum =6; Not Null=False;	As Expected	Fail
CUST_65	Invoice Print	Nil	Nil	Check Box; This check box is default checked.	As Expected	

Table 7.2   Test Cases

Validations						
Basic Information						
CUST_66	Name	Enter numeric only	Test data:123	System should accept entered value.		As Expected	Pass
		Enter characters only	Test data:NGI	System should accept entered value.		As Expected	Pass
		Enter alphanumeric only	Test data:NGI 12	System should accept entered value.		As Expected	Pass
		Enter special characters only	Test data:$% #@	System should accept entered value.		As Expected	Pass
		Leave it Blank	Nil	System Should show validation Message for entering Mandatory field.		As Expected	Pass

CUST_ 68	Nation al Accou nt	Select any National Account from the combo box		It should allow to select any one National Account from the Combo box.		As Expected	Pass
		Leave the Combo box Blank.		It will not display any validatio n message		As Expected	Pass
CUST_ 70	Valid From	Enter Valid From Date Null	Nil	System should give Validation message.		As Expected	Pass
		Enter Valid From Date Grater then Valid to Date.	Nil	System should give appropria te validation message.		Valid To date is always display disbale	Query
		Enter Valid From Date Grater then current date.	Nil	It should accept and save record & status should not be as a inactive.		As Expected	Pass
CUST_ 74	Phone	Enter numeric only	Test data:12 3	System should accept entered value.		As Expected	Pass

	Collection Details						
CUST_ 87	Status	Leave Field Blank	Nil	System generates validation message for entering mandatory fields.		As Expected	Pass
CUST_ 88	Send Statement	Enter numeric only	Test data:123	System should accept entered value.	Send Statement is Combo Box		Pass
		Enter characters only	Test data:NG l	System should accept entered value.	Send Statement is Combo Box		Pass
		Enter alphanumeric only	Test data:NG l12	System should accept entered value.	Send Statement is Combo Box		Pass
		Enter special characters only	Test data:$% #@	System should accept entered value.	Send Statement is Combo Box		Pass
		Enter more than 10 mix characters	Nil	System should not allow user to enter more than 10 mix characters.	Send Statement is Combo Box		Pass
		Leave Field Blank	Nil	System generates validation message for entering mandatory fields.		As Expected	Pass

Accounting Information							
CUST_ 90	Inv Cycle	Leave Field Blank	Nil	System generates validation message for entering mandatory fields.		As Expected	Pass
CUST_ 95	Auto Verify	Check the check box	Nil	Account will be auto verified		As Expected	Pass
		Dont check the check box	Nil	Account verification will not done		As Expected	Pass
CUST_ 97	Bill Group	Select value from the combo & save it	Nil	Selected value from the combo will display as bill group		As Expected	Pass
		Leave it blank	Nil	It will not give any validating message.		As Expected	Pass
Sales Details							
CUST_ 99	Rv to Date	Enter numeric only	Test data:12 3	System should accept accept entered value.		As Expected	Pass
		Enter characters only	Test data:NG l	System should not accept entered value.		As Expected	Pass
		Enter alphanumeric only	Test data:NG l12	System should not accept entered value.		As Expected	Pass

		Enter special characters only	Test data:$%#@	System should not accept entered value.		It isn't allow entered special characters value for RvToDate text box field	Fail
		Leave Field Blank	Nil	System will not give any validation message .		As Expected	Pass
CUST_101	Projected Revenue	Enter numeric only	Test data:123	System should accept entered value.		As Expected	Pass
		Enter characters only	Test data:NGl	System should not accept entered value.		As Expected	Pass
		Enter alphanumeric only	Test data:NGl12	System should not accept entered value.		As Expected	Pass
		Leave Field Blank	Nil	System will not give any validation message .		As Expected	Pass
CUST_106	Account number validation	Click on Add button.	Nil	It should display message "Account number will be generated automatically"		As Expected	Pass

CUST_107	Account number validation for FDX company	Click on Add button and select the FDX as company Id	Nil	It should not display the message "Account number will be generated automatically"		As Expected	Pass
CUST_108	Select company as a FDX	Nil	Nil	Customer code & FedEx Code should display a mandatory			Query
CUST_109	Enter FedEx Code but not enter customer code , click on save button	Nil	Nil	Validation message should display for the customer code	remove FedEx Code		pass
CUST_110	Enter cust code more than 9 digit & click on save button	Nil	Nil	Validation should be display "Cust code should be 9 digits"	Message Display as "Account # should be 9 digits."	As Expected	pass
CUST_111	Enter cust code less than 9 digit & click on save button	Nil	Nil	Validation should be display "Cust code should be 9 digits"	Message Display as "Account # should be 9 digits."	As Expected	pass

CUST_ 112	Enter customer code but not enter FedEx code & click on save button	Nil	Nil	Validation message should display "Fed Ex Code Required"	remove FedEx Code		Pass
CUST_ 113	Enter valid FedEx code & Customer code ,Click on save button	Nil	Nil	Message should display "Record saved successfully"	remove FedEx Code		pass

Table 7.3   Test Cases Validation

BUSINESS RULES				
R – 001	Add Customer	Called when Add Button Clicked	As Expected	Pass
		All the fields becomes enable but valid to ,last payment,G/L DB Dept,Last invoice ,Next review field should disable Verify after clicking on add button only cancel, save buttons enabled. Verify Add button gets disabled after clicking.	As Expected	Pass

	Nil	Verify following fields are mandatory and are displayed with red color border. Basic Information → Name → Company → Valid From Corporate Contact → First Name Corporate Address → Address Line 1 → City → State → Zip Code Collection Details → Status Accounting Information → Inv Cycle	Testcase should be updated for SIC Code and It isn't mandotory field	Query
R – 002	Edit Action	Called when Edit Button Clicked	As Expected	Pass
	Editing Customer record	Verify clicking edit opens the record in edit mode.	As Expected	Pass
		Verify user can update fields. Verify fields shows updated value after saving the record. Save/cancel buttons get disabled and add,edit, delete buttons get enabled.	As Expected	Pass
	Cancel Action	It Called when Cancel Button Clicked. On Click on Cancel Button it it Open the screen of view.	As Expected	Pass
	Customer Basic Info search event	System will open the search dialog screen.	As Expected	Pass
		System will display result accordingly. If there is a above 20000 records then it will display message.	As Expected	Pass
		System should bind a record.	As Expected	Pass

		System should display result accordingly.	It is display only 25 records on this screen	Fail
		System should display the result as per value entered in to search criteria.	As Expected	Pass
		System should display result accordingly.	It is display only 25 records on this screen for entered zip code no	Fail
		If there is no records then system should display message that "Record not found".	As Expected	Pass
	Delete	System should Populate confirmation message with Yes and No option before deleting the record. If user select Yes, system should soft delete the Record.	As Expected	Pass
R – 003	Saving record for Add operation	Verify record saved successfully message is displayed. Verify all fields gets disabled after saving record. Verify Save,Cancel buttons are disabled and Add,Edit,Delete buttons are enabled. Verify correct data is displayed in respective field.	As Expected	Pass
R – 004	Saving record for Edit operation	Verify record saved successfully message is displayed. Verify all fields gets disabled after saving record. Verify Save,Cancel buttons are disabled and Add,Edit,Delete buttons are enabled. Verify correct data is displayed in respective field.	As Expected	Pass

R – 005	Company Combo	User should select any company from below list for New customer. FDX, GRD, NCS, NGB, NGC, NGT and NGW	As Expected	Pass
R – 006	Revenue Type	Revenue charge share link display on Customer Master screen after user bind revenue type customer.		Query
R – 007	Collection Details	User should not allow to enter data in Last Payment Filed.	As Expected	Pass
		User should not allow to enter data in Average Balance Filed.	As Expected	Pass
		User should not allow to enter data in Last Invoice Filed.	As Expected	Pass
		User should not allow to enter data in Next Review Date Filed.	As Expected	Pass
R – 008	Logistic Account	Logistic Check box is disable in Edit Mode After Create Work Order of Customer.		Pending
		Logistic Check box is enable in Edit Mode Before Create Work Order of Customer.		Pending
R – 009	Base TTC	Base TTC Numeric text box have 30 as it's default value.	As Expected	Pass
		Base TTC Numeric text box allow to save with '9999' and it is Base TTC text box's max value. Base TTC Numeric text box is not allow to enter Greater than 9999.	As Expected	Pass
R – 010	B/U Combo	User should select any business Unit from below for New customer. Fedex and NGL	As Expected	Pass
R – 011	Add Mode	City, State and Country bind on Screen based on Zip Code.	As Expected	Pass

		City, State and Country bind on Screen based on Zip Code.	As Expected	Pass
	Edit Mode			
R – 012	Customer Search	There is a single record based on Enter value then that particular customer data bind on Customer Master screen.	As Expected	Pass
		There are multiple customer with enter value than that customer list display in Customer search popup and user should select customer from that list.	As Expected	Pass
		There is no customer with enter value than it should display message "No Record found."	It is no popup message for no customer record found	Fail
R – 013	Customer Master screen	Customer bind on Customer Master screen with Edit Mode. In Edit Mode Company Field is Disable.	As Expected	Pass
		In Add mode Company Combo should enable.	As Expected	Pass
		Account# filed is disable. After save New customer Account# is auto generated for Non Fedex Company.	As Expected	Pass
		Account# filed is Editable and user should enter 9 digit Fedex Account#.	As Expected	Pass
EXTRA CRITERIAS				
Cust_E_ 001	Add customer	It should clear all previous search record when we click on ADD other wise it miss guide user.	As Expected	Pass

Table 7.4 Test Cases Business Rules

8.0 USER MANUAL

The User manual acts an interface between the developer and the user to help the user use the application to suit his needs and requirements. The manual shows the look and feel of the application and pictorially guides the user for the normal course of operation of the application.

This software module will provide a good graphical interface for the user who can operate on the system performing the required tasks such as viewing, adding, updating and deleting the details of various operations master screen. To facilitate this tasks, following interface screen are design in web portal:

1. Customer Master Screens
    a. Customer Search
    b. Customer Master
    c. Business Address
    d. Contacts
    e. Services
    f. Department Info
    g. Documents
    h. Special Instruction
    i. Notification
    j. Cust 3P Relation
    k. Operation Hours
    l. Reference Configuration
    m. Outstanding Invoice

2. Courier Master Screens
    a. Courier Search
    b. Courier Master
    c. Contacts
    d. Service Area
    e. Special Service
    f. Alerts
    g. Courier Vehicles
    h. Courier Delay
    i. Courier Driver

3. Airline Master Screens
    a. Airline Search
    b. Airline Master
    c. Contacts
    d. Notices
    e. Locations
    f. Airline Delay

4. Airport Master Screens
    a. Airport Search
    b. Alternate Airports

5. Inventory Profile Screens
    a. Inventory Profile Search
    b. Cycle Count Configuration

All the above screens are developed by keeping following UX best practices in mind :

● User shall have the ability to access the system from many platforms and locations without installing any software locally.

● The user interface must be designed to be responsive in nature.

Login:

User can login to NGL Connect using valid User Id and password.

Figure 8.1   Log In

Operations:

Once the user is logged in there are a set of operations that one can perform in various sections.

Figure 8.2   Operations

Customer Master Screens:

All the information about different Customers of Network Global Logistics as well the customers of FedEx are displayed according to search criteria and one can manipulate them according to the need.

Figure 8.3   Customer Search

Services:

The main objective of this screen is to provide the user an gui to enter customer services required for the delivery of courier detailed information for the customer service registeration.

Figure 8.4   Services

Department Info:

The main objective of this screen is to provide the user an gui to enter customer details of department information which is required for the delivery of courier.here fields like toll concession etc,are required for the user to enter.

Figure 8.5   Department Info

Documents:

The main objective of this screen is to provide the user an gui to enter customer document details and provide te user wint an interfact to upload documents to the database.

Figure 8.6   Documents

Special Instruction:

The main objective of this screen is to provide an interface for customer Special Instructions for customer.

Figure 8.7   Special Instructions

Reference Configuration:

If the customer has got any reference message and refer value to be given to the recipient then this screen provides a GUI for the customer to enter such details.

Figure 8.8   Reference Configuration

Outstanding Invoice:

If the customer has got pending payment then the details of the customer will be saved in Outstanding Invoice section of the database and the details ate displayed here.

Figure 8.9   Outstanding Invoice 1

Figure 8.10   Outstanding Invoice 2

Courier Master Screens:

All the information about different Couriers of Network Global Logistics are displayed according to search criteria and one can manipulate them according to the need.

Figure 8.11   Courier Master Screens

Service Area:

On load get all the record of Courier Service Area. We will show on grid.
If we select Area type airport than Airport search textbox is visible.
If we select Area type Route Master than RouteWorkId Combo is visible.
If we select Area type zip/postal Code than zip/postal Code search textbox is visible.

Figure 8.12   Service Area 1

Figure 8.13   Service Area 2

Figure 8.14   Service Area 3

Alerts:

For customers who want to attach alert template along with the sent parcel then thy can add the required template to the customer from this screen it provides a GUI for the user to attach alert templates to the order.

Figure 8.15   Alerts 1

Figure 8.16   Alerts

Courier Delay:

On load get all the record of delayed job within From date to To date. By default it will display last 30 days records in grid
OnSerch-User can show all the records by entering From date to To date.
OnReset-Whenever user click on reset button it will display last 30 days records in grid.
OnUpdateRating-User can also update the courier rating.

Figure 8.17   Courier Delay

Airline Master Screens:

All the information about different Airlines that are used by Network Global Logistics are displayed according to search criteria and one can manipulate them according to the need.

Figure 8.18   Airline Search

Locations:

This screen provides the GUI for the customer to enter the airline location details,airport ,city,state,etc it contains all the nformation about the airline location for courier delivery.

Figure 8.19   Locations 1

Figure 8.20   Locations 2

Airport Master Screens:

All the information about different Airports that are used by Network Global Logistics are displayed according to search criteria and one can manipulate them according to the need.

Alternate Airports:
Provides alternate airport for existing airport.

Figure 8.21   Alternate Airports 1

Figure 8.22   Alternate Airports 2

Figure 8.23   Alternate Airports 3

Figure 8.24   Alternate Airports 4

Figure 8.25   Alternate Airports 5

Inventory Profile Screens:

All the information about different Inventories that are used by Network Global Logistics are displayed according to search criteria and one can manipulate them according to the need.

Cycle Count Configuration:

This screen provides the GUI to the end user to to enter the cycle count for inventory cycle ans has fields for Fsl, Classcode, Frequency,cycle amount, etc.

Figure 8.26   Cycle Count Configuration

9.0 LIMITATION AND FUTURE ENHANCEMENT

LIMITATIONS

- System compulsorily needs computers for communication. Internet connection becomes compulsory for a user to interact because it is a web based application.
- Location base for internet and agent is necessary.
- This is web application so user must be on desktop computer to access this website.
- Angularjs is very heavy weight ang its execution is also complex.
- Mysql cannot deal withhetrogenous and multidimentional databases.

FUTURE ENHANCEMENTS

- Converting the application to Angular 3 so to increase the compatibility for all the browsers.
- UI design can be improved in such a way so that application can be opened in mobile browsers also.
- Android and iOS apps can be prepared.

10.0 CONCLUSION AND DISCUSSION

10.1 CONCLUSIONS AND FUTURE ENHANCEMENT

This report has discussed the development of the Result Module with the main objective to replace the current manual result processing system. This development leads to an error-free and efficient Online Result System which would act as a beneficial tool for its users.

The Management system developed would be a web based system which aims to provide a web application to college and its student to easily access and manage results. This will make the ERP System a self-contained database project that works on any operating system with a modern web browser with many features such as result management, attendance management, faculty allocation, attendance and academic monitoring which makes it an efficient ERP system.

The Management system developed are able to save the world's leading businesses large amounts of money, time and effort by creating an effective supply chain. One will be better able to appreciate that the high quality of the product and its value for money are not only a result of high quality design and engineering, but also a direct result of lean production, just-in-time methods and premium supply chain management.

10.2 DISCUSSION

10.2.1 Self-Analysis of Project Viabilities

According to us, this projected is completed with the primary functionalities as specified earlier but then again there is lot more than this which can be done. The project is well capable to handle the given job for some particular task but not all of them. So then it is a challenge to further develop it in to a well flagged software as it was challenge to develop up to this very stage.

10.2.2 Problem Encountered and Possible Solutions

There were many problems encountered during the design and the development phase of the project.

- First I faced many problems in AngularJS and SQL Server 2008 as both concepts were new to me. By making demo application and using online material to study them and with help of our project guide we learned it.
- Moreover the lack of practical know-how of these aspects did cause a few hindrances in early stages of training.
- The problem to maintain threshold.
- The problem to maintain back end service.

- In project we were using Jquery and Bootstrap along with C# . As we were not aware of them, we faced some difficulty while learning for first time. But referring online blogs on going through some tutorials we understood all the concepts.
- We were not aware of how to deploy on server so we faced some difficulty during deploying project on live project but later with the help and guidance of team members it became easier to cope up with these concepts and applications.

10.2.3 Summary of Project work

we have completed our project work using software engineering and system analysis and design approach and have done work with planned scheduling related with time constraints and result oriented progress in project development.

REFERENCES

WEB REFERENCES:

- https://docs.angularjs.org/
- www.stackoverflow.com
- www.tutorialspoint.com
- www.w3schools.com
- www.wikipedia.org

BIBLIOGRAPHY:

- Ng-Book - the Complete Book on Angularjs - Ari Lerner
- "Head First JavaScript" by Michael Morrison
- Head First HTML with CSS & XHTML

EXPERIENCE

Firstly working in Samyak Infotech has been an endeavour of extreme pleasure and in depth knowledge imbibing. To be a bit more precise about the learning during the project work we would like to mention the technical knowledge gained about AngularJs programming, Data Binding,web app development frameworks and many more such technical aspects.

The greatest learning curve during these months was learning about the work culture in software industries, the interpersonal skills gained through the team work and efficient work patterns to resist against stress and work pressure yet achieve the targets given.

The firm has not only helped us develop as programmer but as an individual by letting us realize our potential and believing in our capabilities and hence boosting our confidence.

This project will always remain an integral part of our resume and we hope this wonderful experience here would provide a boost in our career as well.