UNIVERSITY OF MUMBAI

A PROJECT REPORT ON

Scalable RFID Using Cloud Computing

Submitted towards the Partial Fulfillment of the Requirements for the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

By

HARSH DEV(808)

SOHAN KUNKERKAR(823)

DEVEN SAWANT(844)

Under The Guidance Of

Prof. Ankush Hutke

and

Prof. Nilesh Patil

DEPARTMENT OF INFORMATION TECHNOLOGY

MCT's RAJIV GANDHI INSTITUTE OF TECHNOLOGY

VERSOVA, ANDHERI (W), MUMBAI- 400053

2011-2012

MCT's RAJIV GANDHI INSTITUTE OF TECHNOLOGY

Juhu Versova Link Road,

Versova, Andheri (W), Mumbai 400 053.

CERTIFICATE

This is to certify that the project titled **"Scalable RFID Using Cloud Computing "**has been successfully and satisfactorily submitted by **Harsh Dev,Sohan Kunkerkar,**and **Deven Sawant** student of Information Technology as prescribed by University of Mumbai in partial fulfillment of the requirement of Semester VIII for the Degree in **Bachelor of Engineering in Information Technology** during the year 2011-2012.

Internal Guide
(Prof.Ankush Hutke)

Co-Guide
(Prof. Nilesh Patil)

H.O.D
(Prof. S. K. Sabnis)

Principal
(Dr. U.V. Bhosle)

MCT's RAJIV GANDHI INSTITUTE OF TECHNOLOGY

Juhu Versova Link Road,

Versova, Andheri (W), Mumbai 400 053.

CERTIFICATE

This is to certify that the project titled **"Scalable RFID Using Cloud Computing "**has been successfully and satisfactorily submitted by **Harsh Dev,Sohan Kunkerkar,**and **Deven Sawant** student of Information Technology as prescribed by University of Mumbai in partial fulfillment of the requirement of Semester VIII for the Degree in **Bachelor of Engineering in Information Technology** during the year 2011-2012.

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

This endeavour had the benefit of important people whom we wish to acknowledge and thanks in large measures. We hereby take the opportunity to thank all those who have contributed in the successful completion of this project report. We are deeply indebted to our project guide Prof.Ankush Hutke without whose valuable guidance, effort encouragement this project report would not have been completed. We also express our sincere thanks to all the staff members for giving us the project idea and guiding us at the various phases of the project.

We are highly obliged to thank our guide and Head of Department Prof.S.K.Sabnis for his effort and encouragement that boosted our morale and helped us gain experience.This project would be incomplete without thanking all those who contributed directly or indirectly in giving this project a decent shape.

Lastly we acknowledge the many authors whose work has been quoted in this project.

Thanking You,

Harsh Dev

Sohan Kunkerkar

Deven Sawant

ABSTRACT

RFID is a leading edge technology with some bleed points. Some of the issues in RFID are preventing it from gaining everyday acceptance. Such performance issues associated with RFID systems are limited computational capacity, poor resources and inefficient data management. Hence there is a demanding urge to address these issues in the light of some mechanism which can make the technology excel.

Cloud Computing is a paradigm shift which is gaining momentum with the promise of dynamically scalable and often virtualized resources as a service over the network. Motivated from the innovations offered by cloud computing, the primary focus of this research paper is to propose an architecture framework for the existing RFID systems melded with the Cloud computing paradigm in order to improve the scalability and boost the performance of RFID systems.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

RFID technology is being adopted widely for various applications worldwide and at the same time the technology has certain issues which are hindering its exponential growth. Worldwide, researches are being carried out to address the issues with this technology like low computational capabilities of RFID, collision in RFID systems and expansion of RFID systems for huge organizations. The problem is strictly formulated based upon keeping three futuristic scenarios concerns in mind. When it comes to computational capability, the RFID tags start behaving like the main character of movie Ghajni. The main lead of Ghajni tried to solve the mystery of his girlfriend's murder inspire of amnesia and his memory got rebooted after every 15 minutes. Similar is the case with RFID passive tags in the computing world. An RFID tag may effectively reboot itself even more than once per second and then lie abeyant indefinitely. RFID tags are dependent for power on occasional, salvaged, electromagnetic energy from an RFID reader waiting for the next RFID reader to come along. Needless to say, RFID tags are less computational capable.

In addition to that, the existing RFID systems generate terabytes of data which has to be stored, processed and mined before its usage. This characteristic of the existing RFID systems may even lead to enhanced problems even if the computational and data production capability is improved in futuristic RFID systems. Which means that, even if the computational capabilities in the existing RFID systems is improved in future , then it would lead to fast identification which will directly result in generation of more data the same unit time (compared to the existing systems) which has to be stored, processed and mined. This further directly implies the requirement of more resources, services and of course infrastructure. It strictly depicts a strong urge for more scalable RFID systems to support this innovation of increasing computational capabilities in these systems. Across the globe, extensive research is being carried to minimize the collision in RFID systems.

The futuristic RFID systems will be the ones with less number of collisions. Reduction of collisions in such systems would be resulting in fast identification. Fast identification means greater number of tags recognized same unit of time (as compared with existing RFID systems). This directly implies more data production capabilities. Hence, this scenario as well creates a stage for the demand of scalable

RFID systems to support the ongoing inventions.RFID is still in its growth phase and is being adopted by a huge number of organizations. Expanded deployments of RFID systems in big organizations are taking place as a result of adoption of the RFID technology. As a result, more infrastructure, cost, services, applications are required. The problem of scalability is even more intense in such a situation.

All the above depicted scenarios point towards one common concern in RFID systems and that is the scalability. For an RFID to perform more computations, efficiently deal with more data or to be widely adopted, it becomes quite important to design ideas leading to the futuristic scalable RFID systems. In the context of RFID technology, when a single reader is able to identify large number of tags and is very well able to communicate with them, it may be referred as a scalable RFID system. But this often does not happen in the real time scenario due to various practical constraints like collisions during the communication that takes place between the tags and the reader, lack of computational power of RFID tags, bulk of data generated by RFID systems. The above discussed scenarios support this view even more strongly keeping in view the futuristic RFID systems. In other words, there is an alarming need to revive the existing architecture of the existing RFID systems in a way that the new architecture is capable of surviving scalability.

Chapter 2

Project Scope

This Project is giving an innovative solution to scale the capacity of RFID tags in terms of memory.The Project is developed in the hopes of making RFID more efficient and scalable by using Cloud Computing Paradigm.The proposed project is using cloud computing to make information generated by RFID tag scalable.It is a prototype that we have created for RFID which can be used in any application domain.This Project Model can also helpful in medical sector where drugs are manufactured in bulk.

We have taken Drugs Authentication system application to explain the Scalabilty concept for RFID. Many methods had been invented to make RFID scalable which were hardware based, but not efficient enough to tackle the RFID problem.This Project is proposing a software solution for Scalability of RFID which is efficient enough to cope with current RFID memory problem.This Project Model can easily be implemented in Manufacturing Sectors as well as in Inventory control where accuracy and speed is required.

Chapter 3

Literature Survey

One of the biggest advantages of using Radio Frequency Identification in industry and logistics today is the visibility it can provide into the process. With the use of cloud computing, that visibility can be achieved with greater flexibility and lower cost.

3.1 Past History of the Project

RFID (Radio Frequency Identification) is a technology which has been around since early 1900's. It has grown over the years from then. As per the currents scenario, RFID technology is seen heavily influencing our daily lifestyles to an extent that it is woven into our clothes and it has penetrated into our skin. When it comes to tracking objects, RFID has become the prime choice. Fully automated homes are no more a dream. What makes this technology so important would be appreciated by looking at the numerous bonuses it offers over the other existing identification technologies in the market. But even the roses are with thorns and so is the case with this technology. Before actually coming to the issues faced by the technology and the mechanism applied to the issues faced, a brief understanding of the technologies themselves becomes mandatory and the same has been presented in the subsequent paragraphs.
An RFID system can be classified mainly according to the physical components it is composed of, frequency and data. The physical components of an RFID system are primarily numerous tags and readers. The RFID tag related factors are power source it has, environment in which it operates, the antenna on the tag for communication with the reader, its standards, memory, logic applied on the chip and application methods of the tag. The RFID tags depend on power source which may be battery in case of active tags and reader in case of passive tags.
An RFID tag is also associated with the environment in which it operates where the temperature range and the humidity range matters. Another component of an RFID tag is the antenna where its shape and material comes into picture. RFID tags have EPC and ISO standards associated with them. RFID tag has its small memory and logic is also associated along with and RFID chip which may be finite state or microprocessor or none. The application methods of an RFID tag are attached,

removable, embedded or conveyed.
The RFID reader factors include its antenna, polarization, protocol, interface and portability. The antenna for communication in case of RFID reader may be internal or external and its ports may be single or multiple. Polarization of an RFID reader may be linear or circular. Single or multiple protocols may be used. Ethernet, serial, Wi-Fi, USBor other interfaces are used in an RFID reader. Regarding portability, it may be fixed or hand held. Apart from the physical components inside an RFID system, the RFID system may also be viewed from the frequency perspective. Write here

3.2 Problem Definition

3.2.1 Limited Computational Capacity of RFID Tags

When it comes to computational capability, the RFID tags start behaving like the main character of movie Ghajni. The main lead of Ghajni tried to solve the mystery of his girlfriend.s murder inspire of amnesia and his memory got rebooted after every 15 minutes. Similar is the case with RFID passive tags in the computing world. An RFID tag may effectively reboot itself even more than once per second and then lie abeyant indefinitely. RFID tags are dependent for power on occasional, salvaged, electromagnetic energy from an RFID reader waiting for the next RFID reader to come along. Needless to say, RFID tags are less computational capable. In addition to that, the existing RFID systems generate terabytes of data which has to be stored, processed and mined before its usage. This characteristic of the existing RFID systems may even lead to enhanced problems even if the computational and data production capability is improved in futuristic RFID systems. Which means that, even if the computational capabilities in the existing RFID systems is improved in future , then it would lead to fast identification which will directly result in generation of more data the same unit time (compared to the existing systems) which has to be stored, processed and mined. This further directly implies the requirement of more resources, services and of course infrastructure. It strictly depicts a strong urge for more scalable RFID systems to support this innovation of increasing computational capabilities in these systems.

3.2.2 Collisions in RFID Systems

Across the globe, extensive research is being carried to minimize the collision in RFID systems. The futuristic RFID systems will be the ones with less number of collisions. Reduction of collisions in such systems would be resulting in fast identification. Fast identification means greater number of tags recognized same unit of time (as compared with existing RFID systems). This directly implies more data production capabilities. Hence, this scenario as well creates a stage for the demand of scalable RFID systems to support the ongoing inventions.

3.2.3 Adoption of RFID Technology in the industry

RFID is still in its growth phase and is being adopted by a huge number of organizations. Expanded deployments of RFID systems in big organizations are taking place as a result of adoption of the RFID technology. As a result, more infrastructure, cost, services, applications are required. The problem of scalability is even more intense in such a situation.All the above depicted scenarios point towards one common concern in RFID systems and that is the scalability. For an RFID to perform more computations, efficiently deal with more data or to be widely adopted, it becomes quite important to design ideas leading to the futuristic scalable RFID systems. In the context of RFID technology, when a single reader is able to identify large number of tags and is very well able to communicate with them, it may be referred as a scalable RFID system.

But this often does not happen in the real time scenario due to various practical constraints like collisions during the communication that takes place between the tags and the reader, lack of computational power of RFID tags, bulk of data generated by RFID systems. The above discussed scenarios support this view even more strongly keeping in view the futuristic RFID systems. In other words,there is an alarming need to revive the existing architecture of the existing RFID systems in a way that the new architecture is capable of surviving scalability.

Chapter 4

System Analysis

Systems analysis is the study of sets of interacting entities, including computer systems analysis.

4.1 Existing System

RFID provides more data at the edge today than ever before, and the amount of information available will grow with increased volumes, read points and tagging locations. Managing this vast amount of information requires a simple way to collect, filter and transfer and transport data. In addition, the system should be configured to provide a seamless interface to current and future business systems, as well as an upgrade path to support multi-site applications.There are many solutions available to increase the information storing capacity of RFID system which includes Advanced RFID Tags which can hold more data as comapared to normal barcode tag,but just by adding extra hardware facility to the given RFID really doesn't make any sense in scalabilty.Also giving hardware solution for scalabilty of RFID system can increase the cost of RFID.
Hence there is need to have software solution for scalabilty of RFID which is effcient enough to tackle all the problems.

4.2 Proposed System

One of the biggest advantages of using Radio Frequency Identification in industry and logistics today is the visibility it can provide into the process. With the use of cloud computing, that visibility can be achieved with greater flexibility and lower cost.
RFID with cloud computing leverages shared IT infrastructure and standard software to collect and present tracking data without having to develop, maintain and most importantly, finance a redundant and load balanced infrastructure internally. Cloud applications also provide visibility and access anytime, anywhere, and on any web-enabled device.We referred to the following papers while working on the project

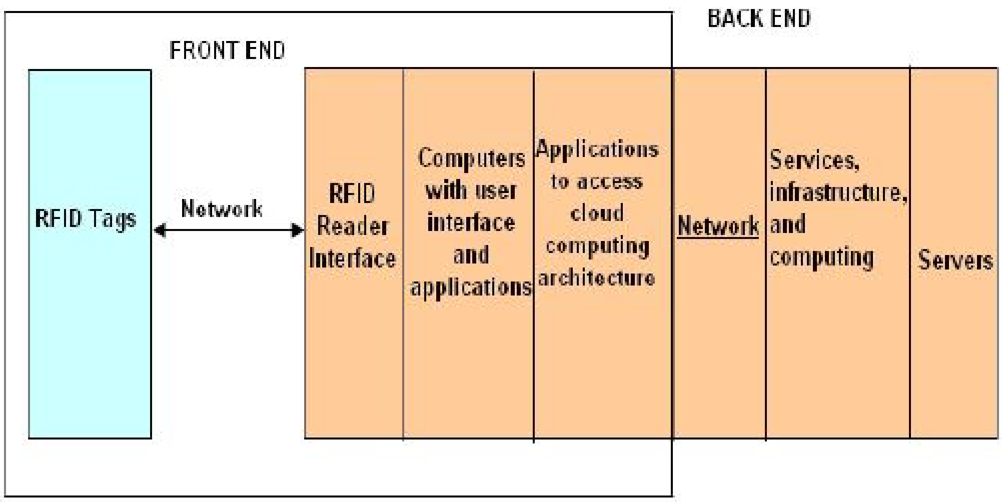


Figure 4.1: Architecture Framework.

to seek help and learn from their conclusions,to make certain improvisations wherever possible.
The main focus of the project is to attach a mechanism called cloud computing to the existing RFID technology and thereby to propose a framework of the architecture for the same. This would further show a way to the futuristic scalable RFID systems. Figure represents the proposed cloud computing architecture for scalable RFID.

The Front End:

The front end part of the proposed architecture is composed up of the networked RFID Tags and Readers, computers with RFID data storage and processing capabilities (limited) and lastly applications which provide access to the cloud computing system. The working of the front end starts up with the initial generation of RFID data as a result of communication between the RFID tags and the RFID reader over the network.This data is generated as and when an RFID tag appears in the interrogation zone of the RFID reader.
It is assumed here that the RFID system under consideration is the futuristic RFID system which is generating even more data than the existing RFID systems as a result of reduced number of collisions, enhanced computational capabilities and expanded infrastructure with even wider adoption of the technology. The data captured by the RFID reader is now sent to the data processing system composedof user interface and specific application to access the back end of the architecture. Now this huge data generated by this RFID system would be filtered, stored, semantically inferred before its actual usage.
Now for all these purposes a cloud computing back end would be used which is connected to the front end via a network. The RFID data would actually be fed as an input to the back end of This proposed cloud computing architecture for scalable RFID in order to enhance the use of RFID data generated for greater purposes.

The Back End:
The back end of the proposed architecture is composed up of infrastructure (computers, data storage systems etc.), applications, services and servers. Here, on the cloud, the RFID data is semantically filtered according to a specific application of need, use its relative platform and infrastructure over the network and then stored on a specific server. Middleware and protocols are used in the cloud computing system here. A server administers the system and traffic.
The middleware and protocols allow the networked resources to communicate to each other. Here, in the proposed architecture, the cloud computing systems need a lot of storage space which it requires to keep all its RFID systems clients' information stored. It makes a copy of all the RFID information and stores it. The copies enable the central server to access backup machines to retrieve data that otherwise would be unreachable. Now as a result of using cloud computing, the huge amount of data which was lost earlier at very early stages as a result of lack of storage and because of bearing low priority in the semantic inference may now be preserved and processed on the cloud to draw more intense conclusions. Moreover, it motivates more efficient utilization of the existing resources rather than demanding a need to incorporate new infrastructure and services.Apart from this data being processed according to specific need and scalable according, the back end of the proposed architecture also provides a way to extend the possibility of further extending the framework of infrastructure and scalable computational capabilities for RFID systems.

4.3 Feasibility Study

4.3.1 Operational Feasibility

The Project does not involve much hardware installations.We are planning to use RFID reader with Serial port connectivity and RFID tags most probably smart cards ,As far as application is concern,it is user-friendly,developed in asp.net technology,For back end we are using Amazon public cloud.We have taken Drugs Authentication System application where in we are not only scaling the database with the help of cloud but also showing Authentication of drugs whose concept is based on HP cloud.

4.3.2 Technical Feasibility

RFID plays an important role in Drugs Authentication System.Here RFID reader will scan smart cards(RFID Tags)and code will be display on windows application The interfacing between the physical hardware and the database can be easily done using c sharp and asp.net.Communication with the server becomes better with this environment. The database that is required holds with itself, the key details about the products. Majority of these details are in the text format.Therefore, Sql Server is good option as it is fast and compatible with asp.net application.

4.3.3 Economic Feasibility

This system is financially feasible.It's development can be done at a price affordable by Manufacturing industries who wish to make use of this technology.The prototype we have constructed involved an investment of approximately Rs. 3000. The split up is mentioned as follows:

Table 4.1: Cost Split-up

Sr.No.	Component	Cost(in Rs.)
1	RFID Reader	800
2	RFID Tag	210(30x7)
3	RFID Adaptor	150
4	Amazon Cloud Subscription	1000

4.4 Project Estimation

Software project estimation is a form of problem solving. Today, software is the most expensive element of virtually all computer based systems. For complex, custom systems, a large cost estimation error can make the difference between profit and loss. Cost overrun can be disastrous for the developer. Hence viable approaches to software project estimation are used.

4.4.1 Cost Estimation

LOC- Function Point based estimation using COCOMO model
FP based Estimation
FP Based estimation focuses on the information domain values such as inputs , outputs ,inquiries, files and external interfaces for class. for the purpose of this estimate, the complexity weighting factor is assumed to be average. The below mentioned table presents the results of this estimates. Count-Total = 423
The estimated FP is derived using the formula:
FP=COUNT-TOTAL*[0.65+0.01 x ?Fi]
Fi = 34
Complexity adjustment factor [0.65+0.01*34] =0.99 = 1
Finally the estimated of FP is derived:
FP=count-total*[0.65=0.01*?Fi]
FP=423*1
FP=423
Considering our average productivity is 141 FP/pm and labor rate is assumed to be INR 200 per month,the cost per FP is approximately INR 1.5, making the project cost INR 5000.

Table 4.2: FP-Based Estimation Result Table

Information Domain Value	Count	Weight	FP-Count
Number of Inputs	20	5	100
Number of Outputs	15	3	45
Number of Inquiries	12	4	48
Number of Files	18	10	180
Number of External Inter-faces	10	5	50

Table 4.3: Weighting Factors Table

Factor	Value
Backup and recovery	4
Data communication	2
Distributed processing	0
Performance critical	4
Existing operating environment	3
On-line data entry	0
Input transactions over multiple screens	0
Master files update online	3
Information values complex	3
Code design for reuse	4
Conversion/installation in design	3
Multiple installations	3
Applications designed for change	5

4.4.2 Effort Estimation

Effort estimation is required to and the number of people required to complete the project over the duration of the project. We are using an estimation model that uses empirically derived formulas to predict effort as function of LOC or FP i.e. the COCOMO model.We have used function points(FP) as a sizing option as a part of model hierarchy.

4.4.3 Time Estimation

Scheduling is an inexact process in that it tries to predict the future. While it is not possible to know with certainty how long a project will take, there are techniques that can increase your likelihood of being close. If you are close in your planning and estimating, you can manage the project to achieve the schedule by accelerating some efforts or modifying approaches to meet required deadlines.

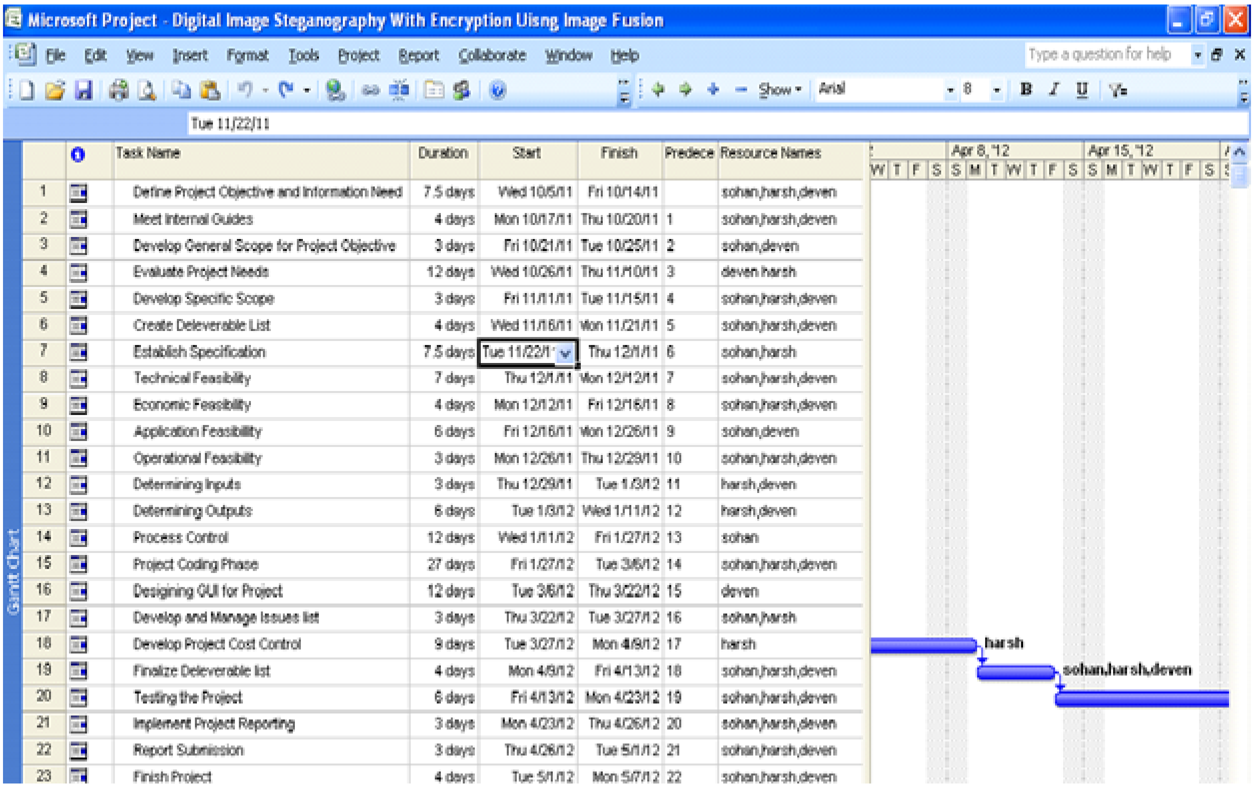
One key ingredient in the scheduling process is experience in the project area; another is experience with scheduling in general. In every industry area there will be a body of knowledge that associates the accomplishment of known work efforts with time duration. In some industries, there are books recording industry standards for use by cost and schedule estimators. Interviewing those who have had experience with similar projects is the best way to determine how long things will really take.

When preparing a schedule estimate, consider that transition between activities often takes time. Organizations or resources outside your direct control may not share your sense of schedule urgency, and their work may take longer to complete. Beware of all external dependency relationships. Uncertain resources of talent, equipment, or data will likely result in extending the project schedule.

Experience teaches that things usually take longer than we think they will, and that giving away schedule margin in the planning phase is a sure way to ensure a highly stressed project effort. People tend to be optimistic in estimating schedules and, on an average, estimated only 80 of the time is actually required. Failure to meet schedule goals is most often due to un- realistic deadlines, passive project execution, unforeseen problems, or things overlooked in the plan.

4.5 Project Planning

Project planning is part of project management, which relates to the use of schedules such as Gantt charts to plan and subsequently report progress within the project environment.
Initially, the project scope is defined and the appropriate methods for completing the project are determined. Following this step, the durations for the various tasks necessary to complete the work are listed and grouped into a work breakdown structure. The logical dependencies between tasks are defined using an activity network diagram that enables identification of the critical path. Float or slack time in the schedule can be calculated using project management software.
Then the necessary resources can be estimated and costs for each activity can be allocated to each resource, giving the total project cost. At this stage, the project plan may be optimized to achieve the appropriate balance between resource usage and project duration to comply with the project objectives. Once established and agreed, the plan becomes what is known as the baseline. Progress will be measured against the baseline throughout the life of the project. Analyzing progress compared to the baseline is known as earned value management.



	Task Name	Duration	Start	Finish	Predece	Resource Names
1	Define Project Objective and Information Need	7.5 days	Wed 10/5/11	Fri 10/14/11		sohan,harsh,deven
2	Meet Internal Guides	4 days	Mon 10/17/11	Thu 10/20/11	1	sohan,harsh,deven
3	Develop General Scope for Project Objective	3 days	Fri 10/21/11	Tue 10/25/11	2	sohan,deven
4	Evaluate Project Needs	12 days	Wed 10/26/11	Thu 11/10/11	3	deven harsh
5	Develop Specific Scope	3 days	Fri 11/11/11	Tue 11/15/11	4	sohan,harsh,deven
6	Create Deleverable List	4 days	Wed 11/16/11	Mon 11/21/11	5	sohan,harsh,deven
7	Establish Specification	7.5 days	Tue 11/22/11	Thu 12/1/11	6	sohan,harsh
8	Technical Feasibility	7 days	Thu 12/1/11	Mon 12/12/11	7	sohan,harsh,deven
9	Economic Feasibility	4 days	Mon 12/12/11	Fri 12/16/11	8	sohan,harsh,deven
10	Application Feasibility	6 days	Fri 12/16/11	Mon 12/26/11	9	sohan,deven
11	Operational Feasibility	3 days	Mon 12/26/11	Thu 12/29/11	10	sohan,harsh,deven
12	Determining Inputs	3 days	Thu 12/29/11	Tue 1/3/12	11	harsh,deven
13	Determining Outputs	6 days	Tue 1/3/12	Wed 1/11/12	12	harsh,deven
14	Process Control	12 days	Wed 1/11/12	Fri 1/27/12	13	sohan
15	Project Coding Phase	27 days	Fri 1/27/12	Tue 3/6/12	14	sohan,harsh,deven
16	Desigining GUI for Project	12 days	Tue 3/6/12	Thu 3/22/12	15	deven
17	Develop and Manage Issues list	3 days	Thu 3/22/12	Tue 3/27/12	16	sohan,harsh
18	Develop Project Cost Control	9 days	Tue 3/27/12	Mon 4/9/12	17	harsh
19	Finalize Deleverable list	4 days	Mon 4/9/12	Fri 4/13/12	18	sohan,harsh,deven
20	Testing the Project	6 days	Fri 4/13/12	Mon 4/23/12	19	sohan,harsh,deven
21	Implement Project Reporting	3 days	Mon 4/23/12	Thu 4/26/12	20	sohan,harsh,deven
22	Report Submission	3 days	Thu 4/26/12	Tue 5/1/12	21	sohan,harsh,deven
23	Finish Project	4 days	Tue 5/1/12	Mon 5/7/12	22	sohan,harsh,deven

Figure 4.2: Project Plan.

4.5.1 Gantt Chart

A Gantt chart is a type of bar chart that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project. Some Gantt charts also show the dependency (i.e. precedence network) relationships between activities.
Although a Gantt chart is useful and valuable for small projects that fit on a single sheet or screen, they can become quite unwieldy for projects with more than about 30 activities. Larger Gantt charts may not be suitable for most computer displays. A related criticism is that Gantt charts communicate relatively little information per unit area of display. That is, projects are often considerably more complex than can be communicated effectively with a Gantt chart.
Gantt charts only represent part of the triple constraints (cost, time and scope) of projects, because they focus primarily on schedule management. Moreover, Gantt charts do not represent the size of a project or the relative size of work elements, therefore the magnitude of a behind-schedule condition is easily miscommunicated. If two projects are the same number of days behind schedule, the larger project has a larger impact on resource utilization, yet the Gantt does not represent this difference.

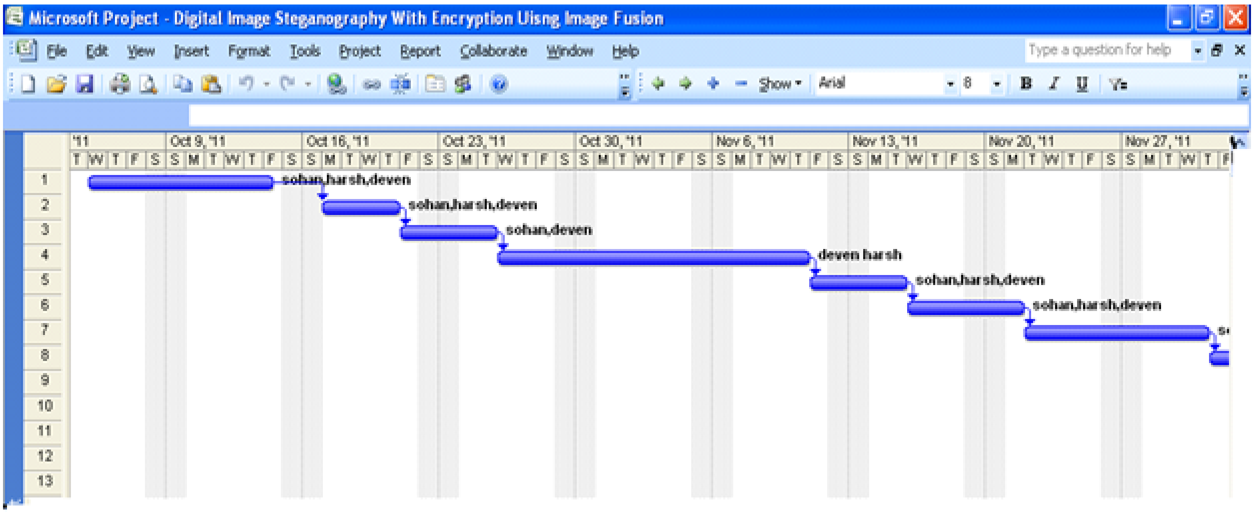


Figure 4.3: Gantt Chart(1/5).

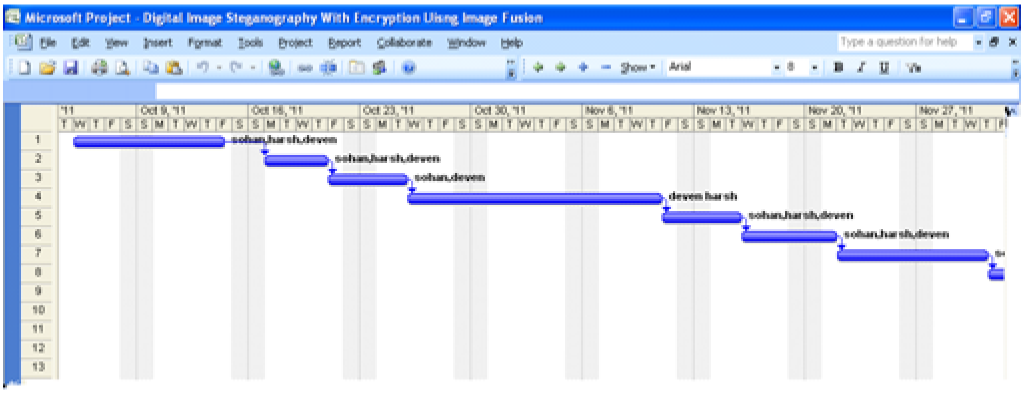


Figure 4.4: Gantt Chart(2/5)

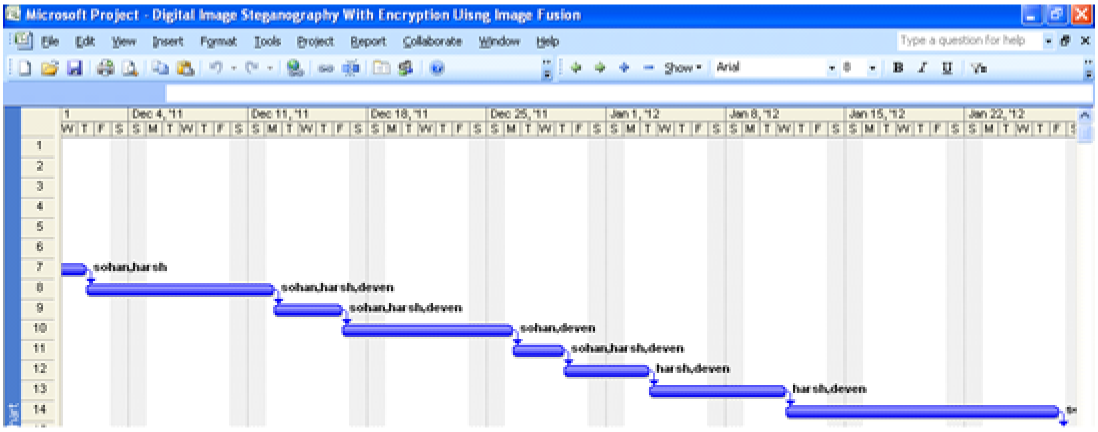


Figure 4.5: Gantt Chart(3/5).

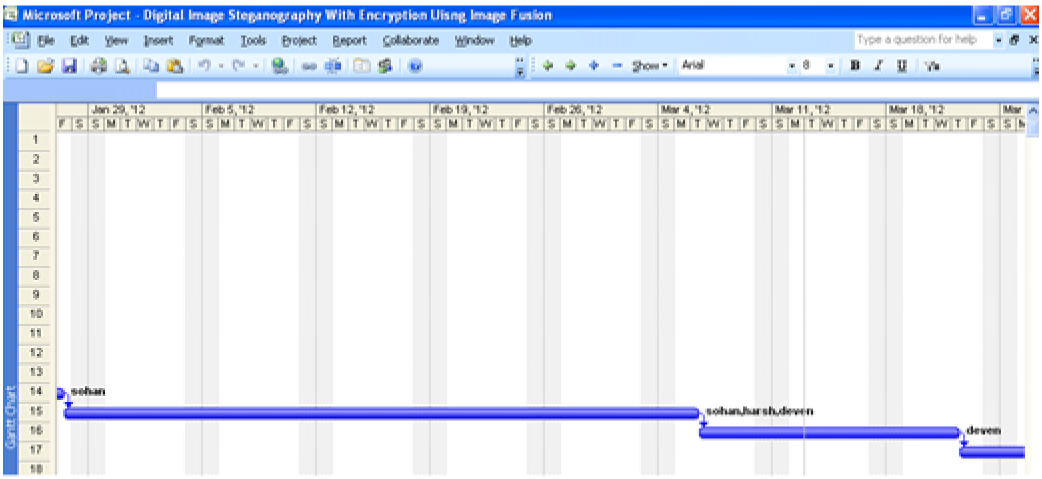


Figure 4.6: Gantt Chart(4/5).

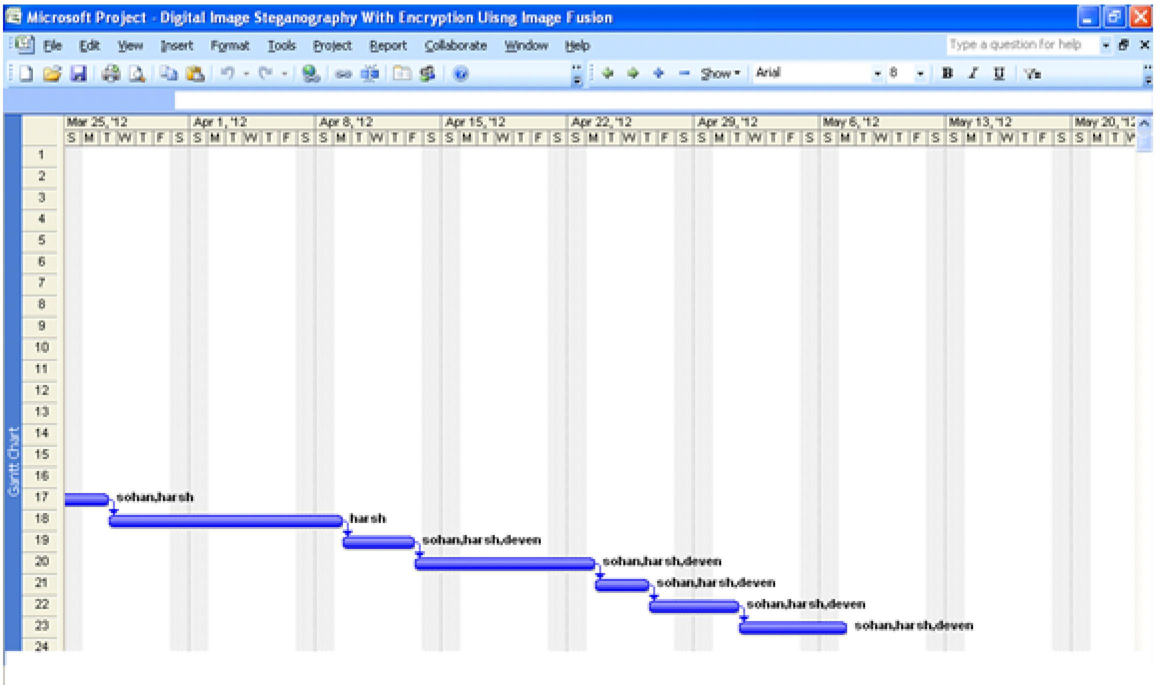


Figure 4.7: Gantt Chart(5/5).

4.5.2 Work BreakDown Structure

A work breakdown structure (WBS) in project management and systems engineering, is a tool used to define and group a project's discrete work elements (or tasks) in a way that helps organize and define the total work scope of the project.
A work breakdown structure element may be a product, data, a service, or any combination. A WBS also provides the necessary framework for detailed cost estimating and control along with providing guidance for schedule development and control. Additionally the WBS is a dynamic tool and can be revised and updated as needed by the project manager.
The Work Breakdown Structure is a tree structure, which shows a subdivision of effort required to achieve an objective; for example a program, project, and contract. In a project or contract, the WBS is developed by starting with the end objective and successively subdividing it into manageable components in terms of size, duration, and responsibility (e.g., systems, subsystems, components, tasks, subtasks, and work packages) which include all steps necessary to achieve the objective.

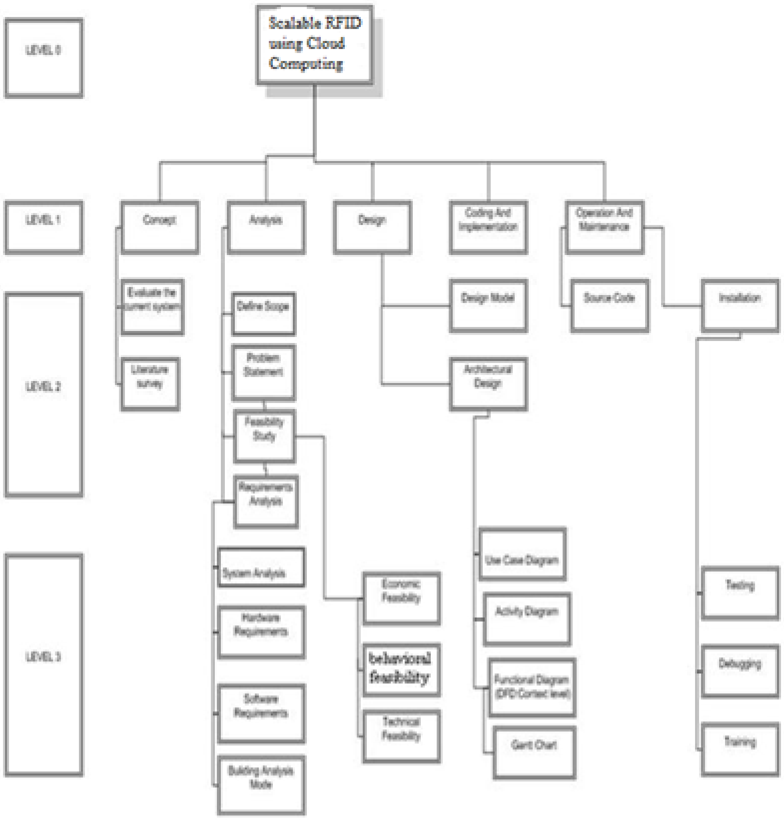


Figure 4.8: Work Breakdown Structure for Scalable RFID using Cloud Computing.

4.6 Software Requirement Specification(SRS)

4.6.1 Introduction

4.5.1.1 Purpose

The purpose of the project is to create a standard technique to make RFID system scalable using Cloud Computing Paradigm.In this project we have created one application which is deployed on Amazon cloud and through RFID reader product will be register on cloud database.Customer,then can buy product and also authenticate it using RFID product code.Hence instead of hardware solution we are providing hassle free software solution which is effcient and cheap. This Project is giving an innovative solution to scale the capacity of RFID tags in terms of memory.The Project is developed in the hopes of making RFID more efficient and scalable by using Cloud Computing Paradigm.The proposed project is using cloud computing to make information generated by RFID tag scalable.It is a prototype that we have created for RFID which can be used in any application domain. This Project Model can also helpful in medical sector where drugs are manufactured in bulk.

We have taken Drugs Authentication system application to explain the Scalabilty concept for RFID. Many methods had been invented to make RFID scalable which were hardware based, but not efficient enough to tackle the RFID problem.This Project is proposing a software solution for Scalability of RFID which is efficient enough to cope with current RFID memory problem.This Project Model can easily be implemented in Manufacturing Sectors as well as in Inventory control where accuracy and speed is required.

4.5.1.4 Overview

SRS will provide the description of Model or prototype i.e Scalable RFID Using Cloud Computing. It will give brief idea about the need of Scalability in the RFID system.RFID System users frequently need to store, send, or receive large amount of information. The most flexible way to do this is to put all the data onto Cloud.That will result in efficient usage of memory and resources.

4.6.2 Positioning

4.5.2.1 Business Opportunity

The Model has a wide variety of application in the field of business. Information can be scalable using Cloud with Pay as per you use Model which is anytime cheaper solution than hardware solution.Thus the space required to store product details is more as compared to earlier method.It can be used in various Manufacturing inudustries ,Retail industries,and IT etc.

4.5.2.2 Problem Statement

The table below shows the our Problem Statement.

The problem is	Sending the large size secret message can be read by other person
Affects	Sending the secret message on the network can be read by other person.The person can interchange the message and send wrong message to receiver or attack on secret information.
The impact of which is	May affect the growth of business, misguidance to others.
The successful solution should	The successful solution should be use Steganography. Which is helpful to sender for secure transmission of message by using encryption method which is included into the Steganography.

Table 4.4: Problem Statement

4.5.2.3 Product Positioning Statement

For	Any one can use this Model, who wants to scale RFID system
That	Scale the data generated by RFID tag using Cloud Computing Paradigm.
Unlike	Other RFID System that uses Hardware solution to scale the memory of RFID Tag,still unable to resolve the problem.
Our prototype	provides a means to scale RFID System using Cloud Computing.

Table 4.5: Product Positioning Statement

4.6.3 Product Overview

4.5.4.1 Product Perspective

It becomes quite important to design ideas leading to the futuristic scalable RFID systems. In the context of RFID technology, when a single reader is able to identify large number of tags and is very well able to communicate with them, it may be referred as a scalable RFID system. But this often does not happen in the real time scenario due to various practical constraints like collisions during the communication that takes place between the tags and the reader, lack of computational power of RFID tags, bulk of data generated by RFID systems. To support this view even more strongly keeping in view the futuristic RFID systems,there is an alarming need to revive the existing architecture of the existing RFID systems in a way that the new architecture is capable of surviving scalability.This project proposed architecture which solves the above problem with the help of Cloud computing.

4.6.4 Constraints

4.5.5.2 Constraints

The problems faced with the Model are:

1.It stores data on cloud hence security constraints is always their..

2.Public cloud subscription is costly.

3.Required thorough knowledge about Cloud computing and RFID interfacing with Cloud

4.7 Other Product Requirements

4.5.6.1 Environmental Requirements

Reliability
The system should be reliable. The information shown should be accurate.

Availability
The system should be up all the time.

Maintainability
The system should be easy to maintain the files should be stored and maintained so that they are available whenever needed.

Portability
The system is portable as it is implemented using .NET framework.

4.5.6.2 System Requirements

A. Hardware Requirements(Minimum)

Following are the hardware specifications needed for implementation of the project:
1. RFID Reader and RFID Tags(smart Cards).

RFID Reader

Description :- 125 KHz RFID readerwith serial and wiegand26 output format.

Features :-

- Easy interface of computer serial terminal through DB9 connector or direct interface to micro-controller via on board connectors
- Onboard buzzer and led for indicating card detection
- Onboard switch for selecting serial wiiegand26 output format or RS232 output
- TTL RS232 signal are available on male header and 10 pin FRC box header

Figure 4.9: RFID Reader.

Technical specifications :-

- Supply voltage 9V to 12 v DC
- Opearating current 50 mA
- Opearating frequency 125KHz
- Read distance 5 to 10 cm

Output data format

- Data baud rate -9600 bps
- Data bit- 8 bit
- Parity check - None
- Stop bit - 1

Testing of RFID card reader:-

- Connect RFID card reader to serial port and to the supply
- In Hyper terminal use setting as baud rate -9600 bps,parity check - None RTS and CTS check on None for that
- Then bring tag to sensor
- You will get tag ID reading in hyperterminal window.

RFID Tags

Description :
This is a basic RFID tag card/ Keyfob uesd for presence sensing , acess control etc. work in the 125 KHz RF range. these tags come with a uinque 32 bit ID and are not re-programmble. card/Keyfob is blank, smooth and middly flexible.
Features :-

- EM4001 ISO based RFID IC

Figure 4.10: RFID Tag.

- 125KHZ carrier
- 2kbps ASK
- Manchester encoding
- 32 bit unique ID
- 64 bit data stream

2. RFID serial port connector.
3. RFID adapter,
4. RAM: atleast 1 GB,
5. The output device as Visual display unit, input device as keyboard, mouse etc.

B. Software Requirements(Minimum)

Following are the hardware specifications needed for implementation of the project:

- Operating system: WINDOWS XP.
- .NET Framework
- Visual Studio-2008 Software
- SQL Server-2005-2008

Chapter 5

System Design

5.1 Preliminary System Design

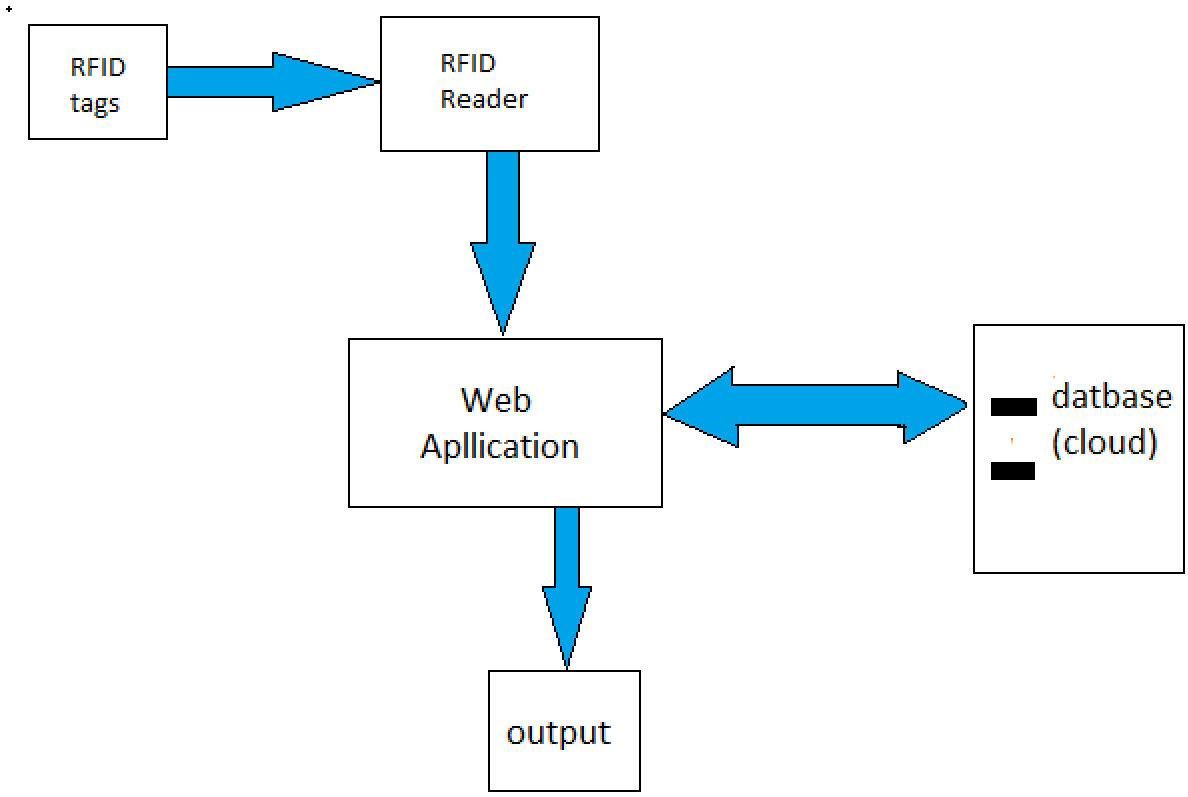


Figure 5.1: Block Diagram

5.2 Use-Case Diagram

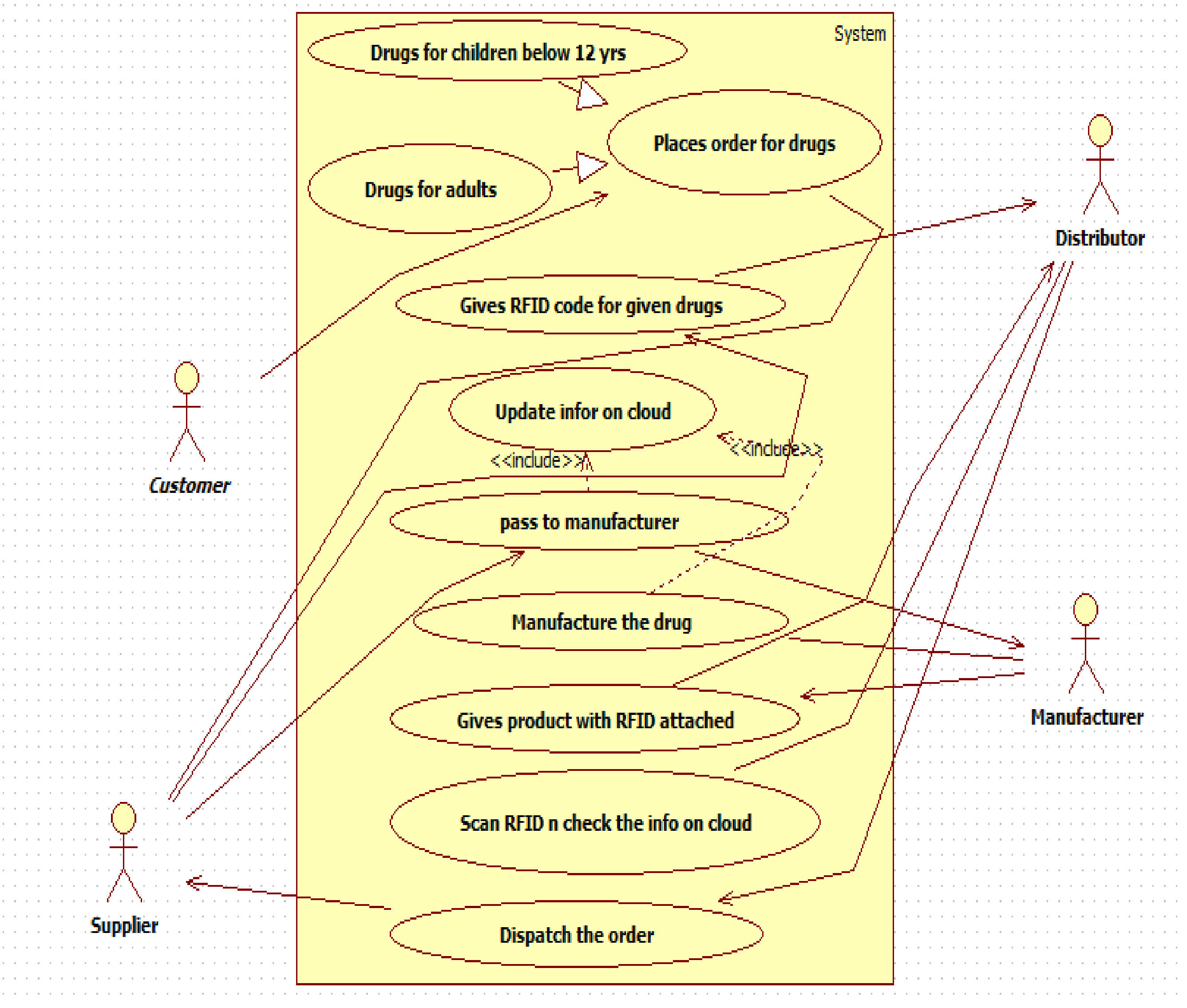


Figure 5.2: Usecase[1/2]

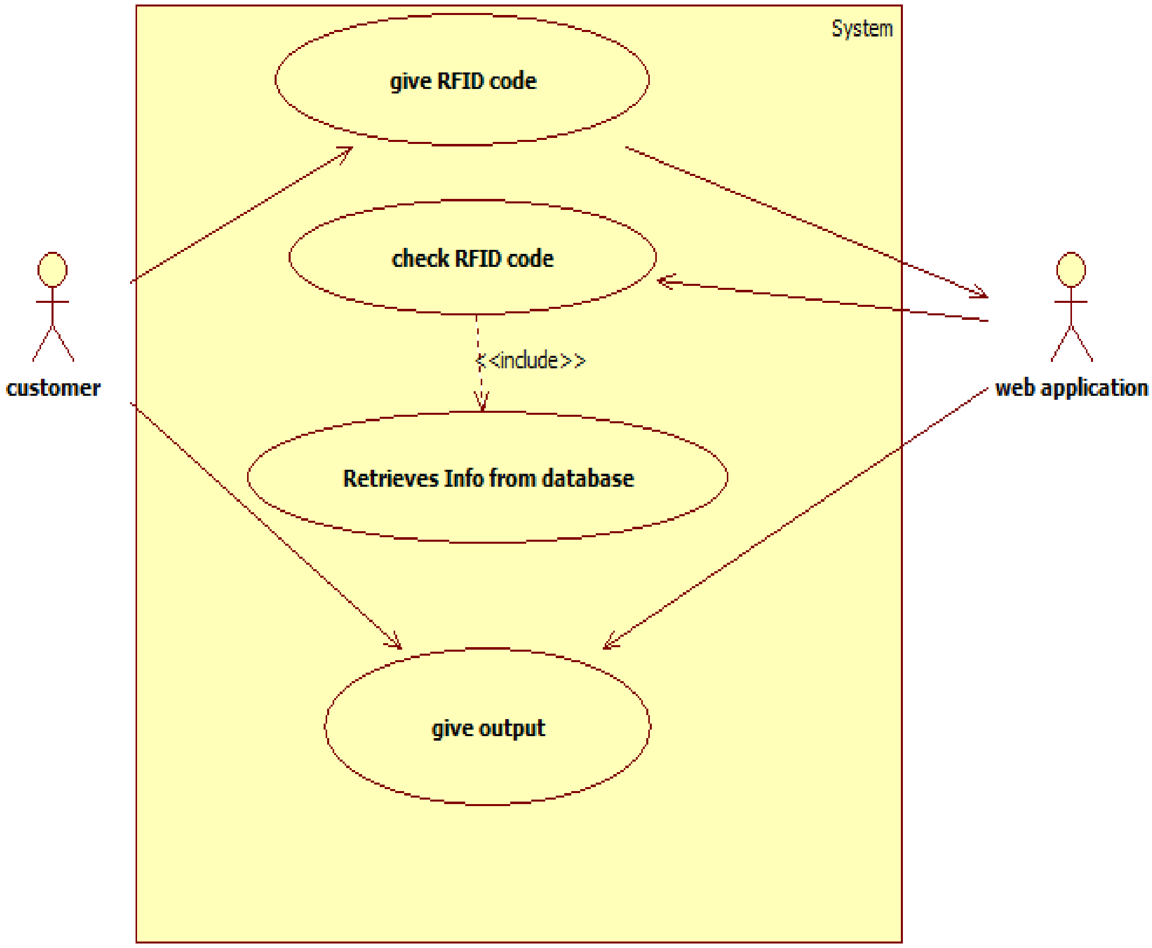


Figure 5.3: Usecase[2/2]

5.3 Database Design

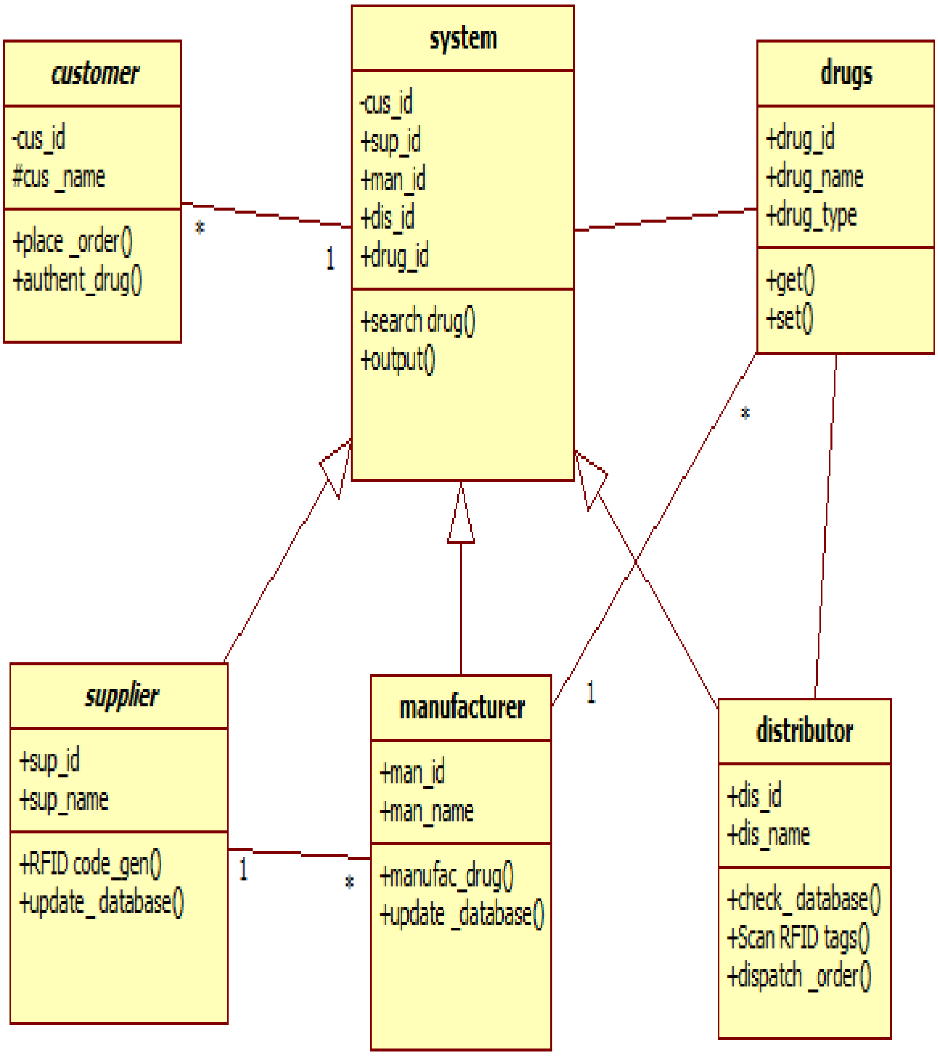


Figure 5.4: Class diagram

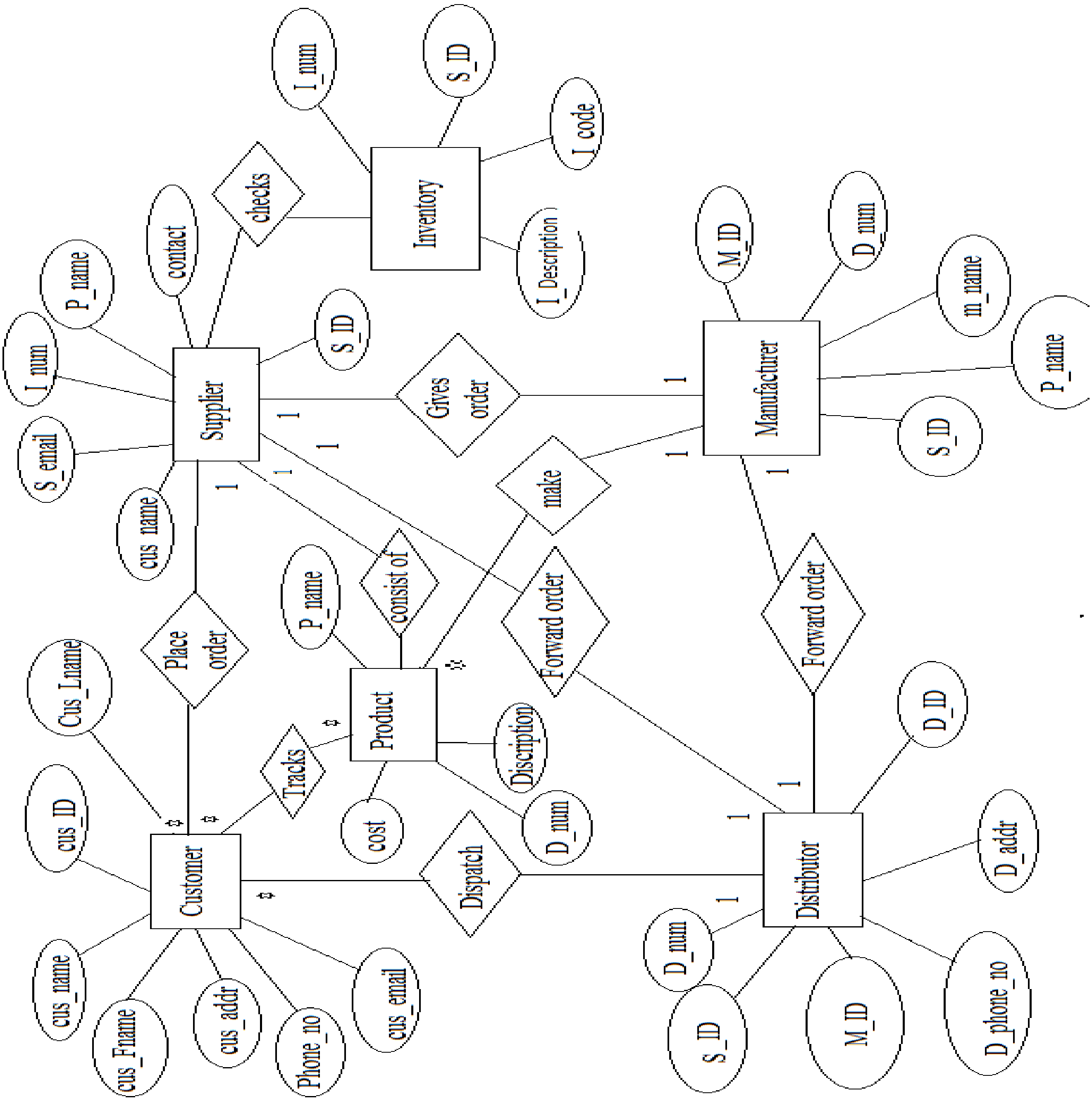


Figure 5.5: ER Diagram

5.4 Activity Diagram

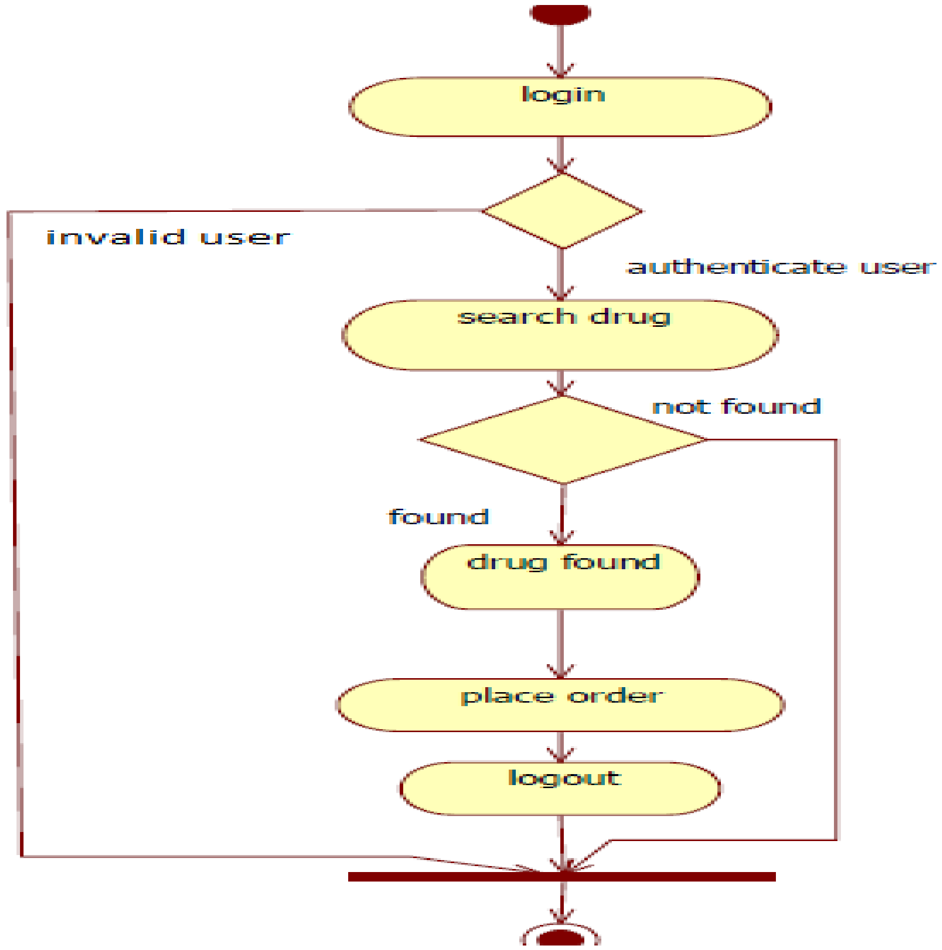


Figure 5.6: Activity Diagram[1/2]

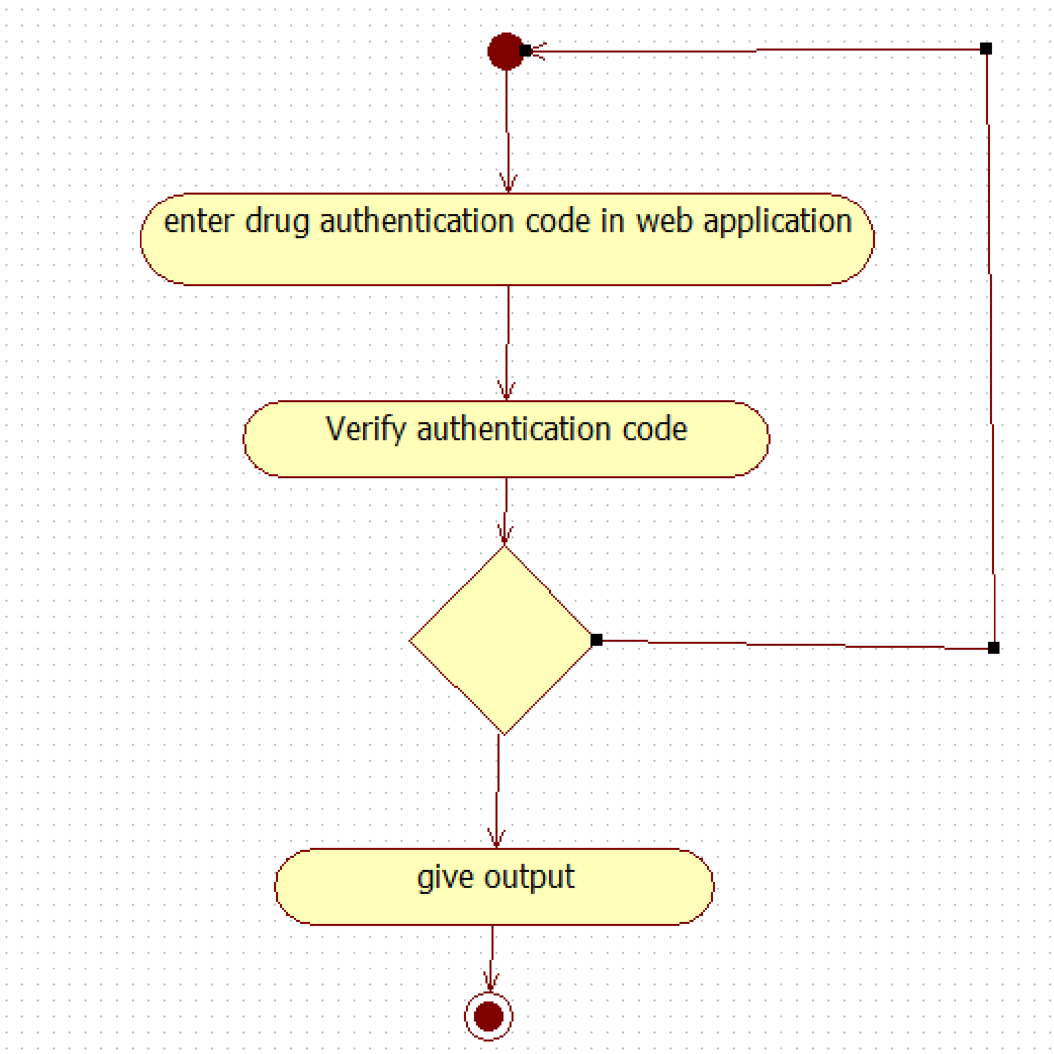


Figure 5.7: Activity Diagram[2/2]

5.5 Sequence Diagram

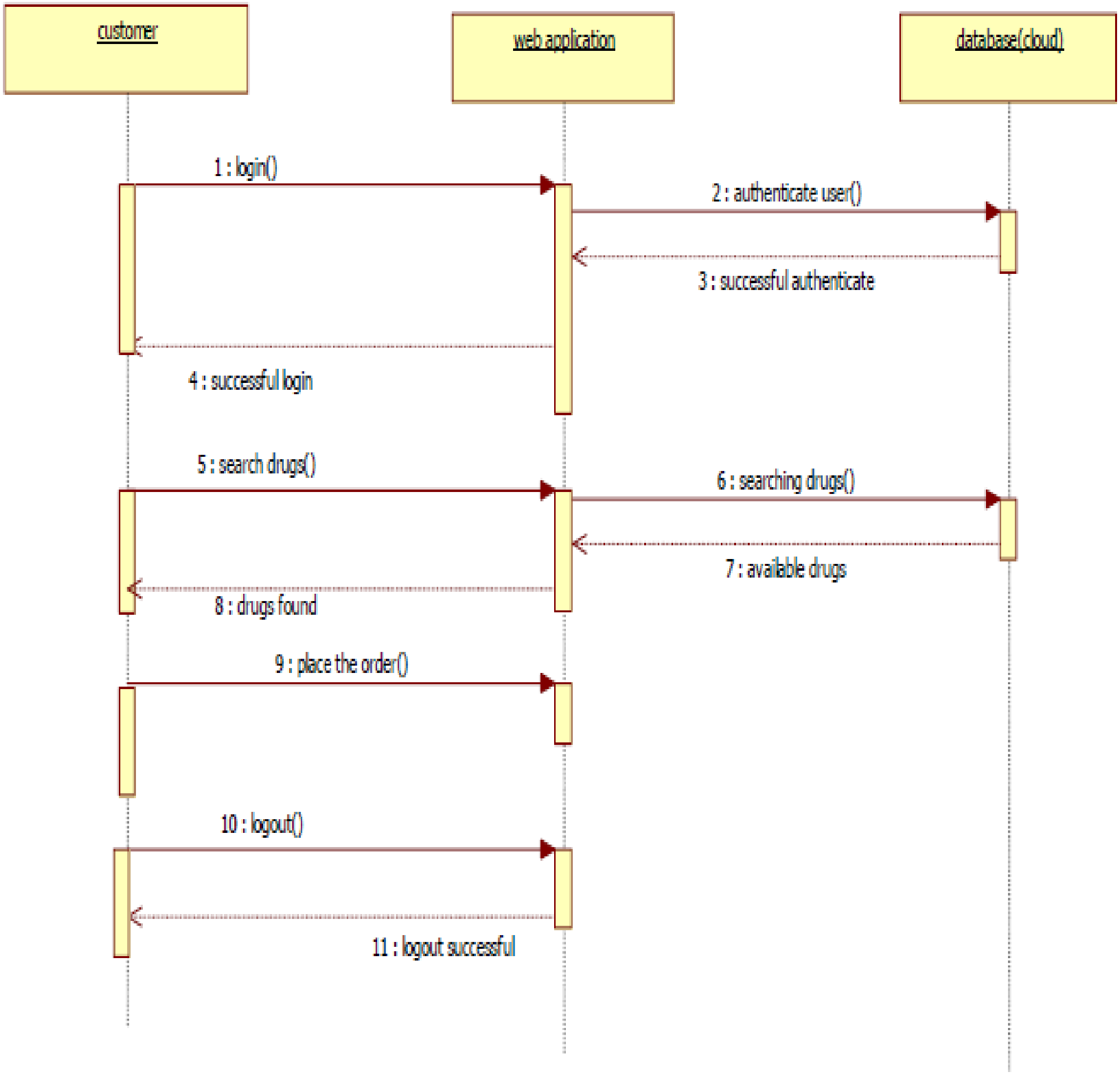


Figure 5.8: Sequence Diagram[1/2]

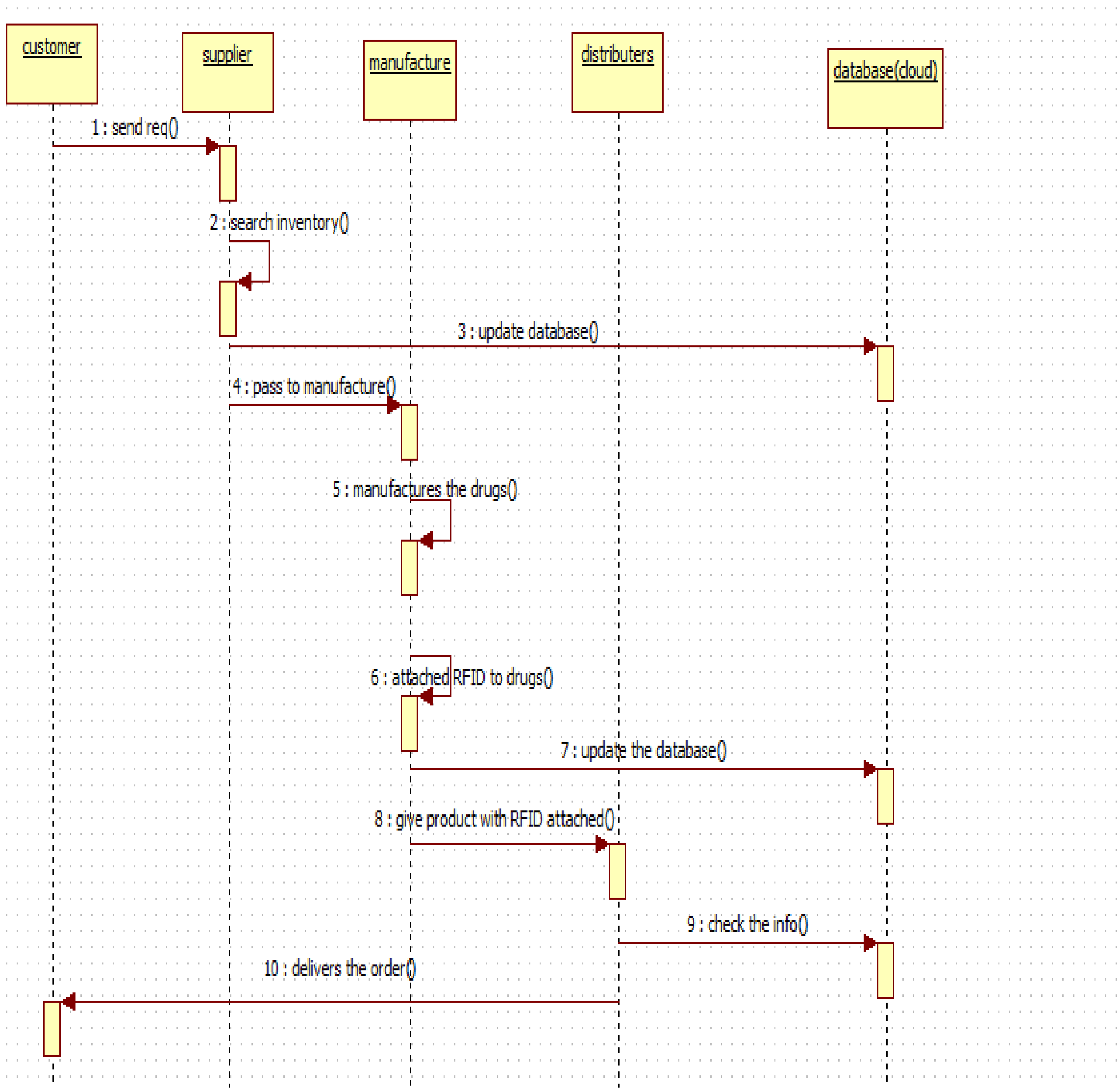


Figure 5.9: Sequence Diagram[2/2]

5.6 Collaboration Diagram

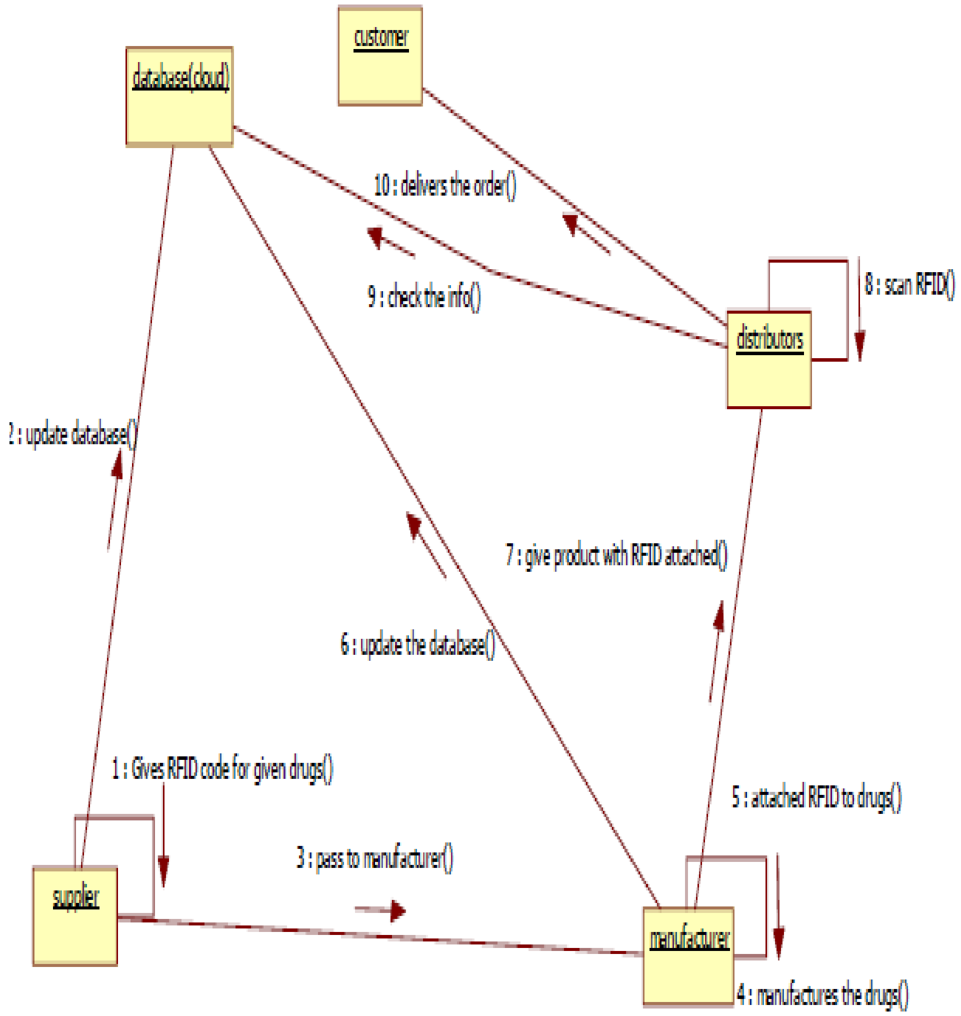


Figure 5.10: Collaboration Diagram[1/2]

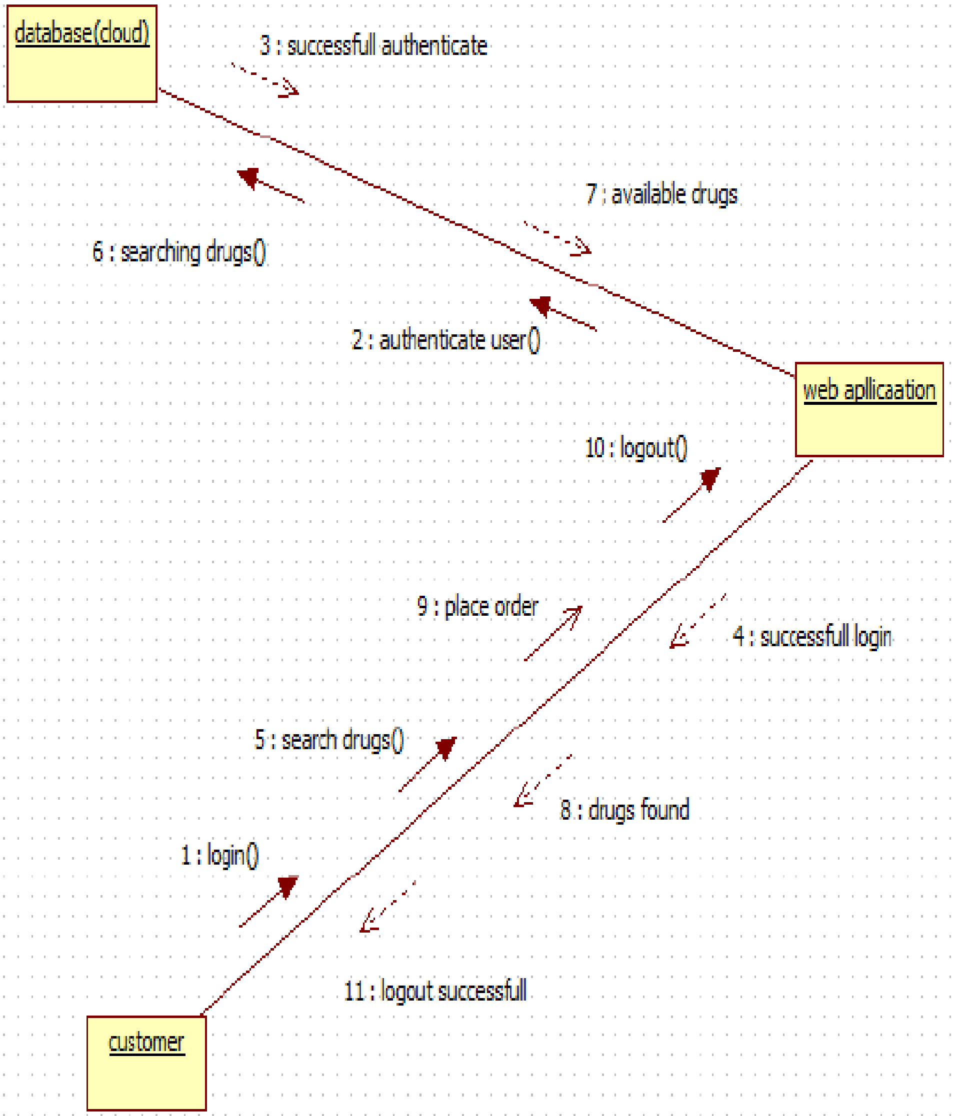


Figure 5.11: Collaboration Diagram[2/2]

5.7 Data-Flow Diagram

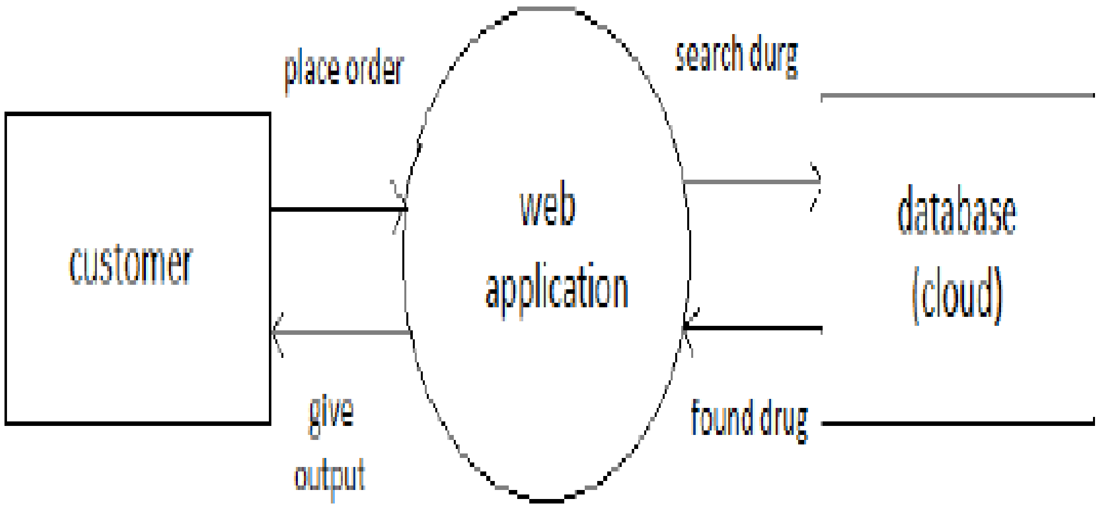


Figure 5.12: Dataflow Diagram[1/2]

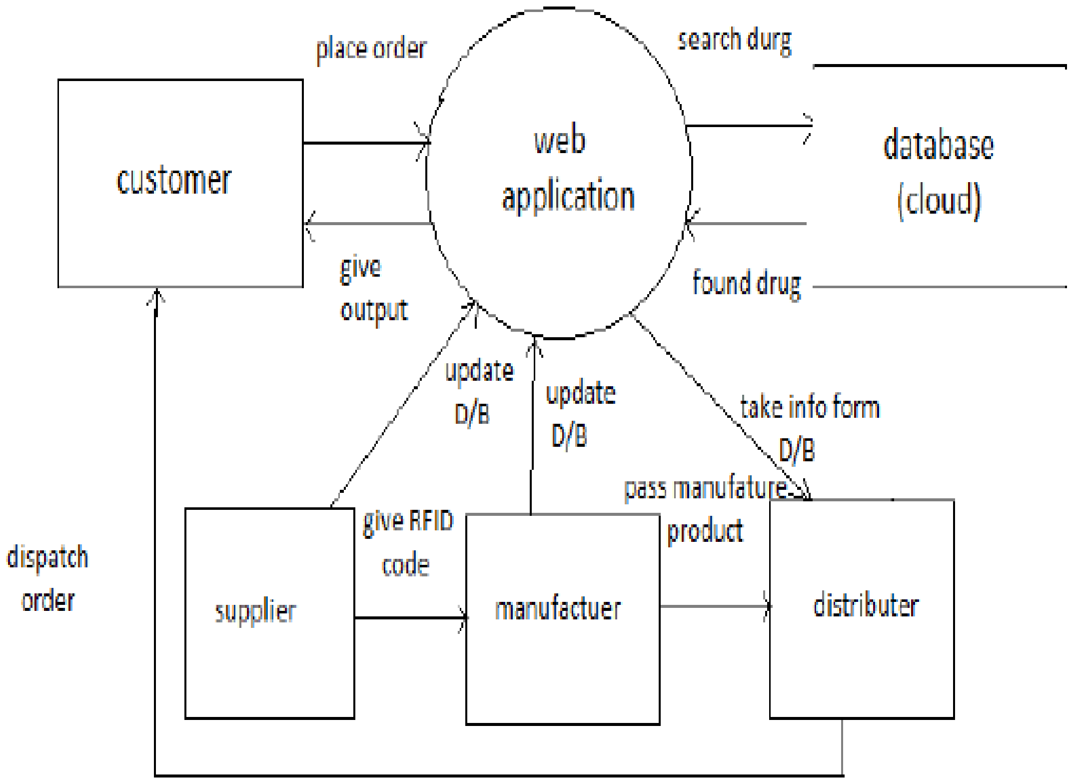


Figure 5.13: Dataflow Diagram[2/2]

Chapter 6

Implementation

6.1 Screenshots

We have designed a graphical user interface in asp.net and SQL Server as the back end. The interface establishes a connection with the database. This is the same database that holds all the records of products already available in the Inventory and also those that are continuously being added or subtracted from the existing stock.

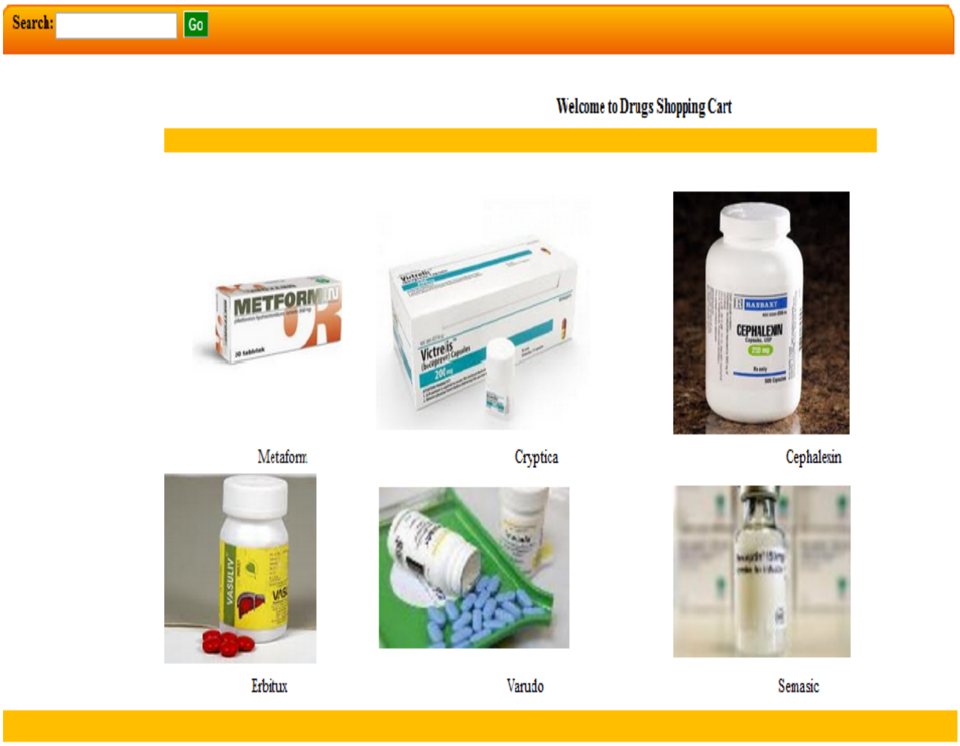


Figure 6.1: Homepage

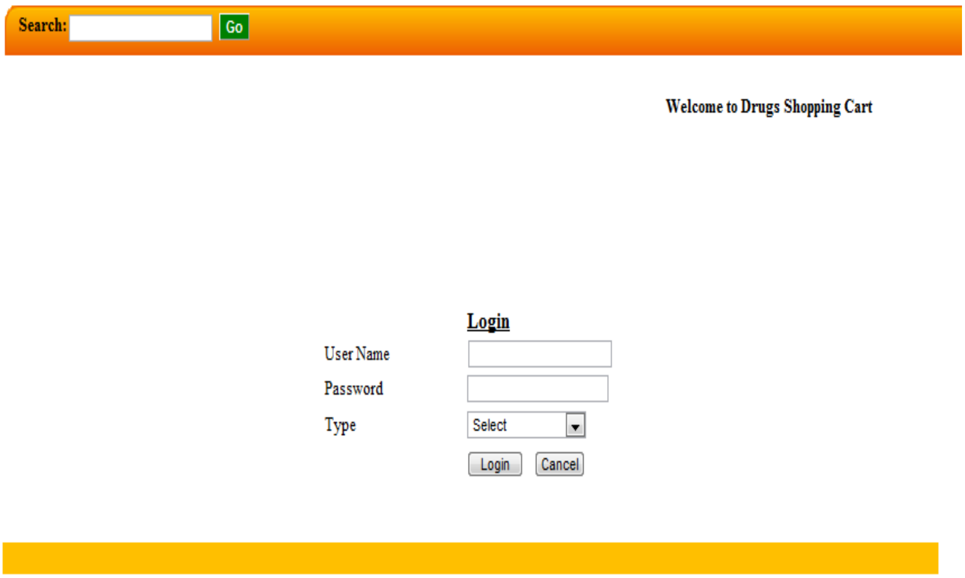


Figure 6.2: Login Page

Figure 6.3: Customer Page

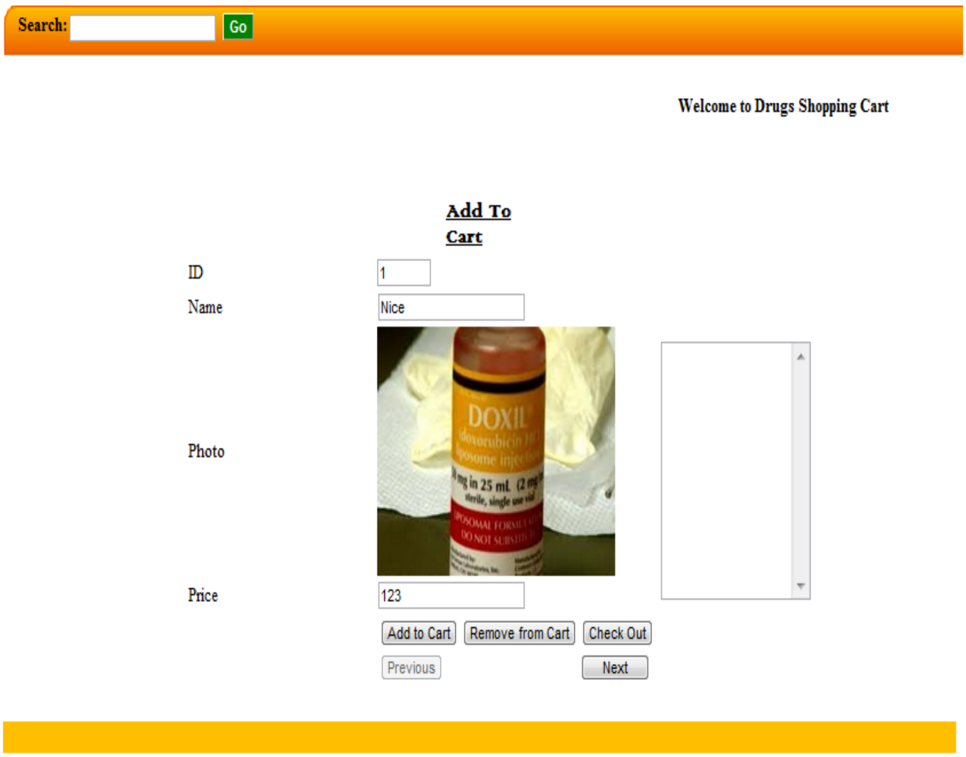
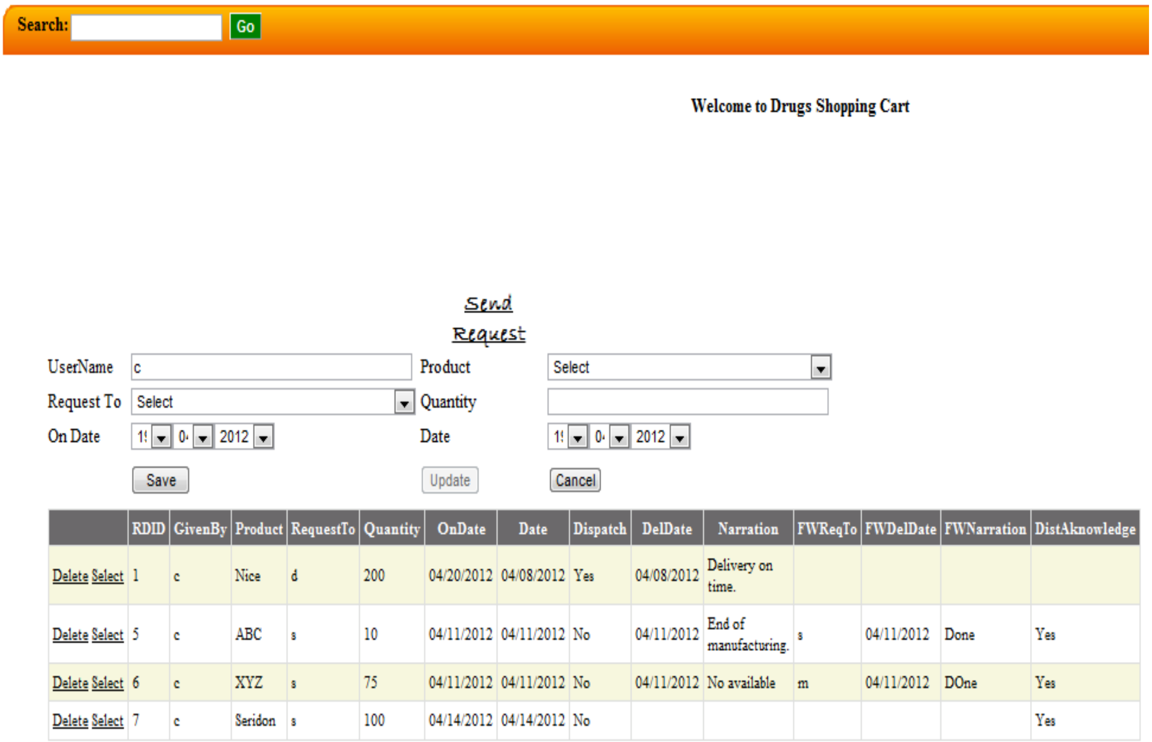


Figure 6.4: Buy Product Page



	RDID	GivenBy	Product	RequestTo	Quantity	OnDate	Date	Dispatch	DelDate	Narration	FWReqTo	FWDelDate	FWNarration	DistAknowledge
Delete Select	1	c	Nice	d	200	04/20/2012	04/08/2012	Yes	04/08/2012	Delivery on time.				
Delete Select	5	c	ABC	s	10	04/11/2012	04/11/2012	No	04/11/2012	End of manufacturing.	s	04/11/2012	Done	Yes
Delete Select	6	c	XYZ	s	75	04/11/2012	04/11/2012	No	04/11/2012	No available	m	04/11/2012	DOne	Yes
Delete Select	7	c	Seridon	s	100	04/14/2012	04/14/2012	No						Yes

Figure 6.5: Customer Request Page

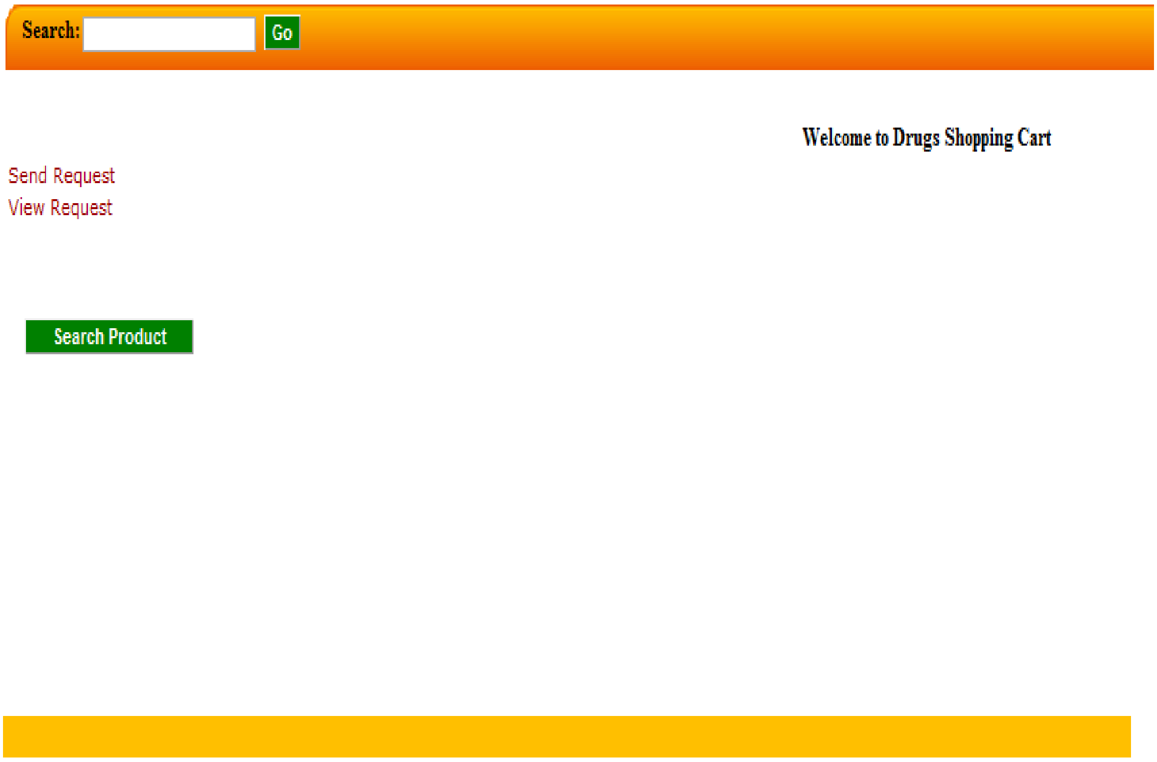


Figure 6.6: Supplier Home Page

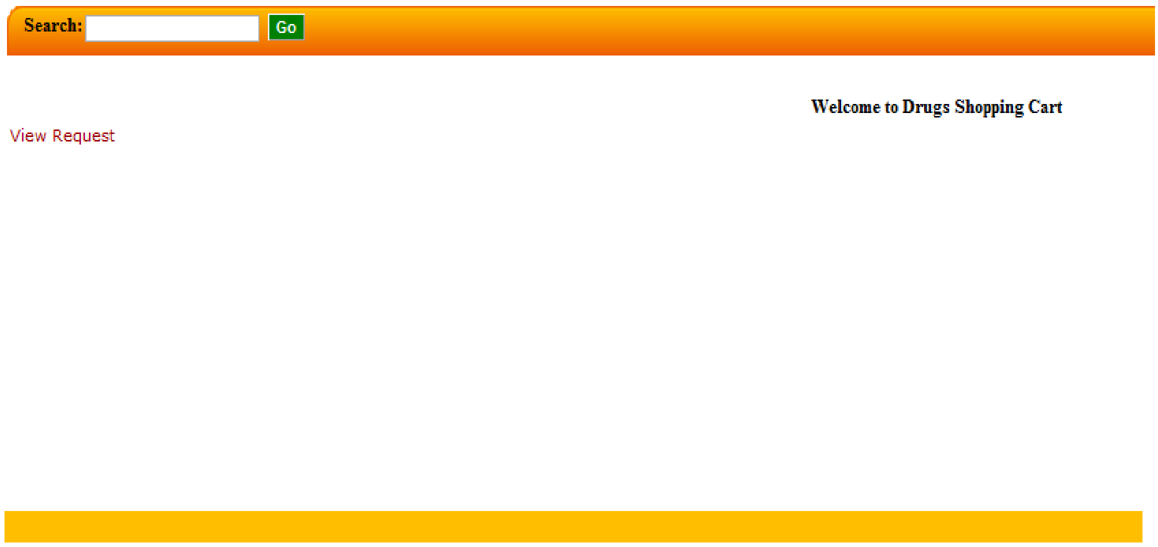


Figure 6.7: Distributor Home Page

Figure 6.8: Manufacturer Home Page

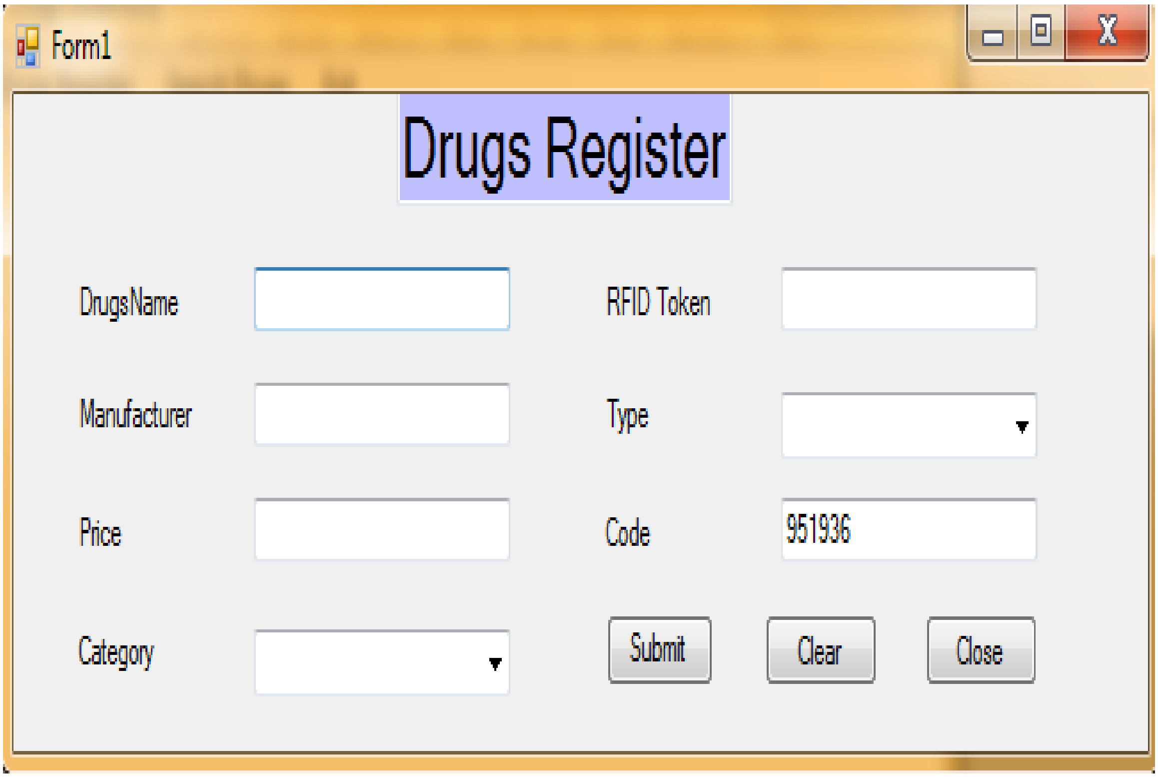


Figure 6.9: Application for registration of product

Figure 6.10: Application for searching of product

6.2 Component Diagram

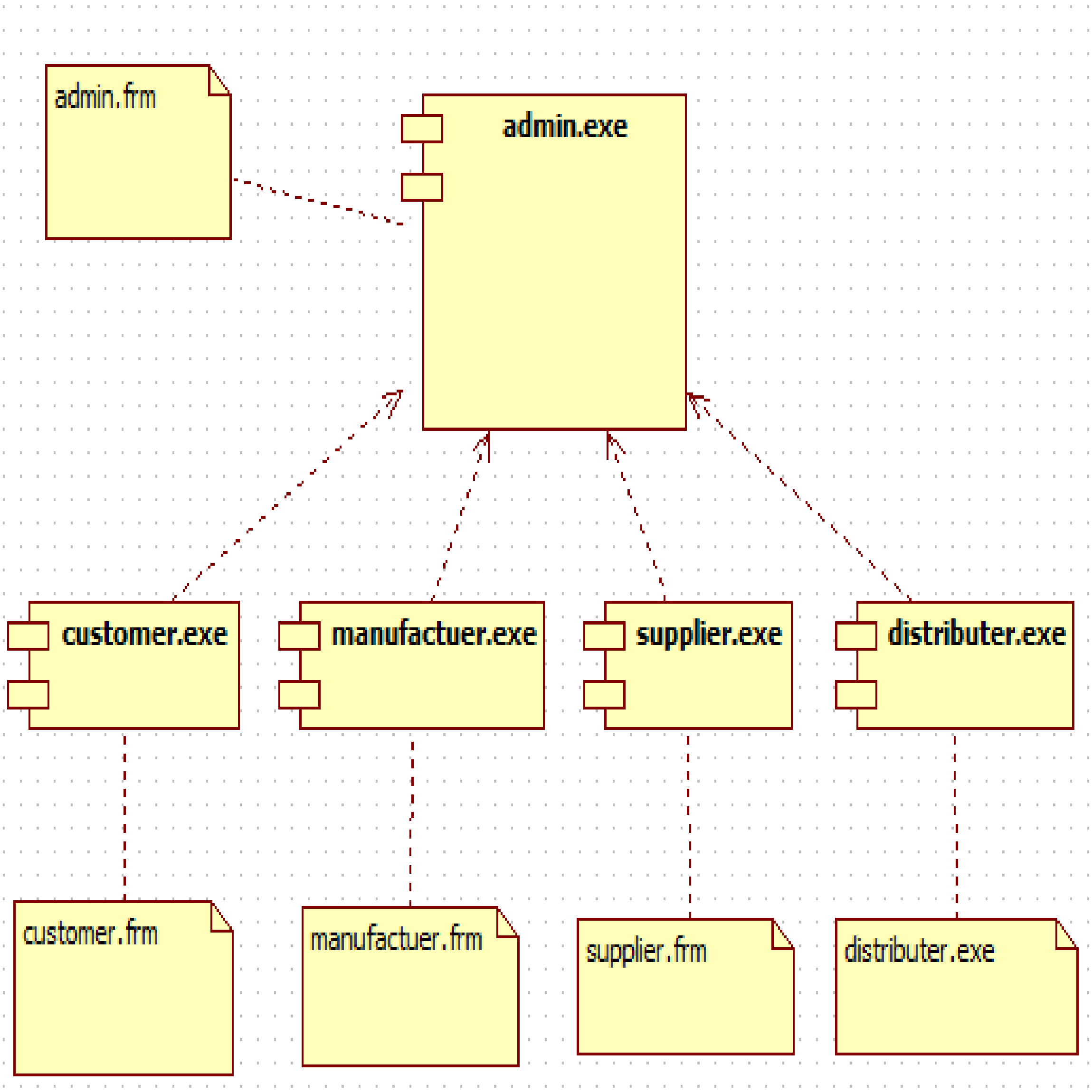


Figure 6.11: Component Diagram

6.3 Deployment Diagram

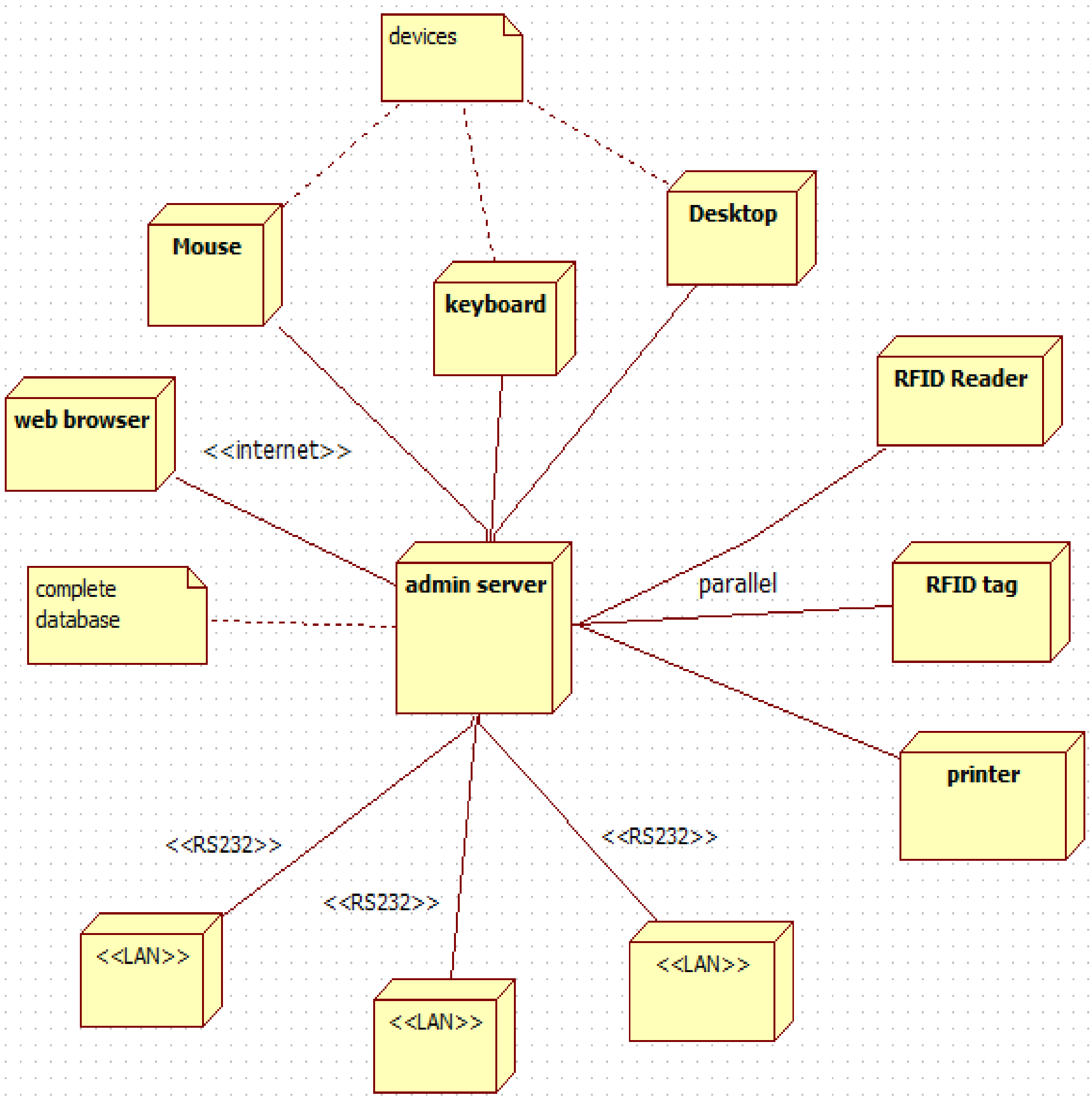


Figure 6.12: Deployment Diagram

Chapter 7

Testing and Debugging

System testing provides the file assurance that software once validated must be combined with all other system elements. System testing verifies whether all elements have been combined properly and that overall system function and performance is achieved.

Characteristics of a Good Test:
- Tests are likely to catch bugs
- No redundancy
- Not too simple or too complex

7.1 Types of Testing

7.1.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software designs the module. To check whether each module in the software works properly so that it gives desired outputs to the given inputs. All validations and conditions are tested in the module level in the unit test.

Control paths are tested to ensure the information properly flows into, and output of the program unit and out of the program unit under test. Boundary conditions are tested to ensure that the modules operate at boundaries. All independent paths through the control structure ensure that all statements in a module have been executed at least once.

7.1.2 Integration Testing

The major concerns of integration testing are developing an incremental strategy that will limit the complexity of entire actions among components as they are added to the system. Developing a component as they are added to the system, developing an implementation and integration schedules that will make the modules available when needed, and designing test cases that will demonstrate the viability of the evolving system. Though each program works individually they should work after linking them together. This is also referred to as interfacing. Data may be lost across interface and one module can have adverse effect on another.

Subroutines after linking may not do the desired function expected by the main routine. Integration testing is a systematic technique for constructing program structure while at the same time conducting tests to uncover errors associated with the interface. In the testing, the programs are constructed and tested in small segments.

7.1.3 System Testing

When a system is developed, it is hoped that it performs properly. In practice however some errors always occur. The main purpose of testing and information system is to find the errors and correct them. A successful test is one which finds an error.

The main objectives of system testing are ◊ To ensure during operation the system will perform as per specifications.
◊ To make sure that the system meets users requirements during operation.
◊ To verify that the controls incorporated in the system function as intended.
◊ To see that when correct inputs are fed to the system the outputs are correct.
◊ To make sure that during operation incorrect input and output will be deleted.

The scope of a system test should include both manual operations and computerized. Operations system testing is a comprehensive evaluation of the programs, manual procedures, computer operations and controls. System testing is the process of checking if the developed system is working according to the original objectives and requirements. All testing needs to be conducted in accordance to the test conditions specified earlier.

Acceptance Testing:

Acceptance testing involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies the requirements.

7.1.4 Functioanl Testing

The objective of functional testing is to validate the software behaviour against the functional requirements and specifications. A functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behaviours and outputs (Everett and McLeod, 2007). Functional requirements are defined in terms of calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish (Malan and Bredemeyer, 1999).
Therefore, functional testing has been conducted for the proposed scheme to evaluate the compliance with its specified requirements. In order to proceed with the functional testing, several test steps are constructed based on the use cases in the system design. The functional test of the proposed scheme will be partitioned into

three parts which are encoding and the decoding process as well as the multi-layered embedding.

7.1.5 White Box Testing

White box testing, clear box testing, glass box testing or structural testing is used in computer programming, software engineering and software testing to check that outputs of a program, given certain inputs, conform to the struc- tural specification of the program. The term white box indicates that testing is done with knowledge of the code used to execute certain functionality. For this reason, a programmer is usually required to perform white box tests. A complementary technique, black box testing, performs testing based on previ- ously understood requirements, without knowledge of how the code executes.

7.1.6 Black Box Testing

Black box testing, concrete box or functional testing is used in computer programming, software engineering and software testing to check that out- puts of a program, given certain inputs, conform to the functional specifica- tion of the program. The term black box indicates that the internal imple- mentation of the program being executed is not examined by the tester. For this reason, black box testing is not carried out by the programmer. In most real-world engineering rms, one group does design work while a separate group does testing.

Test Cases of GUI Module:-

Test Case Id: 01

Test Objective: To test the GUI of the system

Item No	*Test Condition*	*Operator action*	*Input specifiaction*	*Output Specification*	*Pass/ Fail*
1	Successful Reception of RFID Tag	RFID reader reads the RFID Tag.	RFID Tag	Retrieves data of respective RFID Tag	Pass
2	Successful reception of data.	Retrieves data from database of respective RFID Tag.	None	Display product details of respective RFID Tag and send microprocessor	Pass Pass

Chapter 8

Installation and Maintainance

Installation and Maintainance is one the important task in project development.

8.1 Installation

The prototype we have designed is a standalone System that would be powered by cloud computing paradigm , and shall remain independent of any further manual pro- gramming or updations on the device-side. The installation process is simple and fast.

It involves the following steps:

- Connect a RFID reader to power supply to RFID Tags.
- Run the asp.net application in Visual Studio 2005-2008.
- Link it to pre-existing database of all the enlisted products.
- Scan and connect the device to the database server via Serial port connection.
- After step 4, verify that the COM Port that connects the device is the same as mentioned in the VB application. If not recofigure it with the new value of the COM Port.
- A red LED blinking on the RFID Reader indicates that the device is unable to draw the desired power from the power source.
- Once steps 1-7 are successfully carried out, the device is ready for use.

8.2 Maintainance

1. We are making use of RFID Tags for the simple reason that they can be used and reused repeatedly. As and when a new product has stocked up in the Inventory, a tag must be affixed to it and an update should be made in the database with the new Tag ID associated with the product.
2. This is the database that holds all the records and is linked directly to the inventory.
3. A regular maintenance of this database ensures smooth and uninterrupted working of the entire system.

Chapter 9

Conclusion

This project proposes a cloud computing architecture framework which provides a direction towards the scalable RFID systems. It involves the commitment of the existing cloud computing systems to enhance the computational capabilities and wide adoption of the RFID systems on the cloud. Here, in the proposed architecture framework, virtualized resources as services over the network are melded with the RFID technology with limited resources. It may act as a prototype for the futuristic more scalable RFID systems since it evolves itself under the light of the cloud computing systems.

Chapter 10

Future Scope

When working to meet RFID System, it is important to implement a scalable solution that not only satisfies today's needs, but also allows for future growth.The future seems pretty bright for the prototype that we have tried to implement and as far as modifications are concerned, it wouldn't be wrong if one says that they are limitless. As of now, we are using Cloud computing paradigm for the scalability of RFID system.Deploying databse and application on cloud will not only save the storage space but also gives efficient usage of available resources. Also extra features may be added to it, for example, Using AI in Cloud which can administered the entire database without actual human intervention.so it is quite effcient in manufacturing industries where workforce is quite large, as and when needed.
RFID technology being reliable and robust, can be applied is several real world scenarios to improve the productivity of industries using RFID system. More aspects may be investigated and incorporated before actual system implementation.

Chapter 11

References

[1] Keeping It Real: Protecting the worlds poor from fake drugs,,International Policy Network, May 2009.

[2] H Mark Sehmer and Nathan Pleshek , The Cloud and RFID; Making Deployments Seamless, International Journal of Computer Science(IJCSS), Volume (3)

[3] Cloud-based Multi-purpose Asset Tracking Systems for Business, Industry, Government,Education, and HealthCare ;http://www.queraltinc.com

[4]Ankita Satija and Zachariah Thomas ,HP Launches Cloud based Global Authentication Service to Fight Counterfeit Drugs

[5] Daniel Owunwanne and Rajni Goel:Radio Frequency Identification (RFID) Technology: Gaining A Competitive Value Through Cloud Computing;International Journal of Management and Information Systems Fourth Quarter 2010 Volume 14, Number 5

[6] Chetna Dabas and J.P.Gupta:A Cloud Computing Architecture Framework for Scalable, RFID Proceedings of the International Multiconference of Engineers and Computer Scientists 2010 Vol-I,IMECS 2010,March 17-19,2010,Hong Kong

[7] Johnson D, Kiran Murari, Murthy Raju, Suseendran RB, Yogesh Girikumar,Eucalyptus Beginners Guide - UEC Edition (Ubuntu Server 10.04 - Lucid Lynx),ICT Services Technology Asoption, May 10, 2010

[8] The 10th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2010). May 17-20, Melbourne, Victoria, Australia.
http://www.manjrasoft.com/ccgrid[2010]

[9] IEEE Cloud 2010: The 3rd International Conference on Cloud Computing.
http://www.thecloudcomputing.org/2010/cfp.html]

[10]AWS: "Amazon Elastic Compute Cloud Getting Started Guide API Version.pdf" 2011-12-15

[11]CLOUDCOMP 2009: First International Conference on Cloud Computing October 19 - 21, 2009, Munich, Germany [http://www.cloudcomp.eu/—http://www.cloudcomp.eu/]

[12] Manish Bhuptani : "RFID Field Guide: Deploying Radio Frequency Identi

cation Systems by"

[13]Steven Sheppard: "RFID: Radio Frequency Identi cation (McGraw-Hill Networking Profes- sional)"

[14] http://www.google.com

[15] http://www.wikipedia.com

Chapter 12

Research Paper on Project

A Cloud Computing Architecture Framework for Scalable RFID

Abstract

RFID is a leading edge technology with some bleed points. Some of the issues in RFID are preventing it from gaining everyday acceptance. Such performance issues associated with RFID systems are limited computational capacity, poor resources and inefficient data management. Hence there is a demanding urge to address these issues in the light of some mechanism which can make the technology excel.Cloud computing is a paradigm shift which is gaining momentum with the promise of dynamically scalable and often virtualized resources as a service over the network. Motivated from the innovations offered by cloud computing, the primary focus of this research paper is to propose an architecture framework for the existing RFID systems melded with the Cloud computing paradigm in order to improve the scalability and boost the performance of RFID systems.

Keywords– Architecture, Cloud Computing, RFID, Scalability.

I. INTRODUCTION

RFID (Radio Frequency Identification) is a technology which has been around since early 1900s. It has grown over the years from then. As per the currents scenario, RFID technology is seen heavily influencing our daily lifestyles to an extent that it is woven into our clothes and it has penetrated into our skin. When it comes to tracking objects, RFID has become the prime choice. Fully automated homes are no more a dream. What makes this technology so important would be appreciated by looking at the numerous bonuses it offers over the other existing identification technologies in the market. But even the roses are with thorns and so is the case with this technology. Before actually coming to the issues faced by the technology and the mechanism applied to the issues faced, a brief understanding of the technologies themselves becomes mandatory and the same has been presented in the subsequent paragraphs. An RFID system can be classified mainly according to the physical components it is composed of, frequency and data. The physical components of an RFID system are primarily numerous tags and readers. The RFID tag related factors are power source it has, environment in which it operates, the antenna on the tag for communication with the reader, its standards, memory, logic applied on the chip and application methods of the tag. The RFID tags depend on power source which may be battery in case of active tags and reader in case of passive tags. An RFID tag is also associated with the environment in which it operates where the temperature range and the humidity range matters. Another component of an RFID tag is the antenna where its shape and material comes into picture. RFID tags have EPC and ISO standards associated with them. RFID tag has its small memory and logic is also associated along with and RFID chip which may be finite state or microprocessor or none. The application methods of an RFID tag are attached, removable, embedded or conveyed. The RFID reader factors include its antenna, polarization, protocol, interface and portability. The antenna for communication in case of RFID reader may be internal or external and its ports may be single or multiple. Polarization of an RFID reader may be linear or circular. Single or multiple protocols may be used. Ethernet, serial, Wi-Fi, USB or other interfaces are used in an RFID reader. Regarding portability, it may be fixed or handheld. Apart from the physical components inside an RFID system, the RFID system may also be viewed from the frequency perspective. The frequency may further be classified according to the signal distance, signal range, reader to tag, tag to reader, coupling. The signal distance includes the read range and the write range. The signal range here in case of RFID systems points towards the various frequency bands i.e. LF, HF, UHF and Microwave. Reader to tag frequency may be single frequency or multi-frequency. Tag to reader frequency may be sub harmonic, harmonic, corresponding or an-harmonic. Load modulation, surface acoustic wave or backscatters are coupling used in RFID systems. The data sub classification is in terms of the security associated with RFID systems, multi tag read co-ordination and processing. Public algorithm, proprietary algorithm or none are applied for security in RFID systems. The multi-tag read co-ordination techniques used in the latest RFID systems include SDMA, TDMA, FDMA, CDMA. The processing part is composed of the middleware which further has its own architecture which may be single-tier or multi-tier and its location may be reader or the server. The cloud computing term refers to computational resources (computing) made accessible as scalable, on-demand services over a network (the cloud) [10]. The concept of cloud computing may be well understood by using an example. Suppose a server room executive of a large academic institute requires installing the right hardware and software on the personal computers of all the faculty of the institute as per the requirements of their respective allocated courses. This would further require the purchasing of various softwares or software licenses to give the faculty the tools they require for their respective courses. Whenever there would be a new hire in the institute, more software need to be purchased, it must be made sure that the current software license allows another user and which may not be an easy task for a large organization. If cloud computing is used in such a scenario, instead of installing a suite of software for each computer, the server room executive need to load only one application. That application would allow the faculty to log into a Web-based service which hosts all the programs the user would need for his or her job. Remote machines owned by another organization would run everything from word processing to complex data analysis programs. The cloud computing may be broadly categorized as software-as-a-service, platform-as-a-service and infrastructure-as-a-service. First, software-as-a-service (SaaS) is also known as application-as-a-service. It includes the process of any application being delivered over the platform of the Web to an end user, typically leveraging the application through a browser. It is based on the more traditional timesharing model where many users shared one application and one computer. Second, Platform-as-a-service is a complete platform, including application development, interface development, database development, storage, and testing, delivered through a remotely hosted platform to subscribers. Based upon the traditional timesharing model, modern platform-as-service providers provide the ability to create enterprise-class applications for use locally or on-demand for a small subscription price or for free. Third, the infrastructure-as-a-service, (IaaS) is what makes basic computational resources like disk space, storage, and servers available as on-demand services. Instead of using physical machines, IaaS customers get access to virtual servers on which they deploy their own software, generally from the operating system on up. It leads to cost savings and risk reduction since a big amount of capital expenditures are eliminated which were required in the deployment of infrastructure or large-scale applications in-house [10].

II. Problem Formulation

RFID technology is being adopted widely for various applications worldwide and at the same time the technology has certain issues which are hindering its exponential growth. Worldwide, researches are being carried out to address the issues with this technology like low computational capabilities of RFID, collision in RFID systems and expansion of RFID systems for huge organizations to name a few [2,9]. The problem is strictly formulated based upon keeping three futuristic scenarios concerns in mind.
A. Limited Computational Capacity of RFID Tags
When it comes to computational capability, the RFID tags start behaving like the main character of movie Ghajni. The main lead of Ghajni tried to solve the mystery of his girlfriends murder inspire of amnesia and his memory got rebooted after every 15 minutes. Similar is the case with RFID passive tags in the computing world [3]. An RFID tag may effectively reboot itself even more than once per second and then lie abeyant indefinitely. RFID tags are dependent for power on occasional, salvaged, electromagnetic energy from an RFID reader waiting for the next RFID reader to come along. Needless to say, RFID tags are less computational capable. In addition to that, the existing RFID systems generate terabytes of data which has to be stored, processed and mined before its usage [7]. This characteristic of the existing RFID systems may even lead to enhanced problems even if the computational and data production capability is improved in futuristic RFID systems. Which means that, even if the computational capabilities in the existing RFID systems is improved in future , then it would lead to fast identification which will directly result in generation of more data the same unit time (compared to the existing systems) which has to be stored, processed and mined [8]. This further directly implies the requirement of more resources, services and of course infrastructure. It strictly depicts a strong urge for more scalable RFID systems to support this innovation of increasing computational capabilities in these systems.
B. Collisions in RFID Systems
Across the globe, extensive research is being carried to minimize the collision in RFID systems [4], [6], and [9]. The futuristic RFID systems will be the ones with less number of collisions. Reduction of collisions in such systems would be resulting in fast identification. Fast identification means greater number of tags recognized same unit of time (as compared with existing RFID systems). This directly implies more data production capabilities. Hence, this scenario as well creates a stage for the demand of scalable RFID systems to support the ongoing inventions.
C. Adoption of RFID Technology in the industry
RFID is still in its growth phase and is being adopted by a huge number of organizations [2], and [11]. Expanded deployments of RFID systems in big organizations are taking place as a result of adoption of the RFID technology. As a result, more infrastructure, cost, services, applications are required. The problem of scalability is even more intense in such a situation. All the above depicted scenarios point towards one common concern in RFID systems and that is the scalability. For an RFID to perform more computations, efficiently deal with more data or to be widely adopted, it becomes quite important to design ideas leading to the futuristic scalable RFID systems. In the context of RFID technology, when a single reader is able to identify large number of tags and is very well able to communicate with them, it may be referred as a scalable RFID system. But this often does not happen in the real time scenario due to various practical constraints like collisions during the communication that takes place between the tags and the reader, lack of computational power of RFID tags, bulk of data generated by RFID systems. The above discussed scenarios support this view even more strongly keeping in view the futuristic RFID systems. In other words, there is an alarming need to revive the existing architecture of the existing RFID systems in a way that the new architecture is capable of surviving scalability.

III. Problem Solution

THE PROPOSED ARCHITECTURE
The main focus of the paper is to attach a mechanism called cloud computing to the existing RFID technology and thereby to propose a framework of the architecture for the same. This would further show a way to the futuristic scalable RFID systems. Fig 1 represents the proposed cloud computing architecture for scalable RFID..

The Front End: The front end part of the proposed architecture is composed up of the networked RFID Tags and Readers, computers with RFID data storage and processing capabilities (limited) and lastly applications which provide access to the cloud computing system. The working of the front end starts up with the initial generation of RFID data as a result of communication between the RFID tags and the RFID reader over the network. This data is generated as and when an RFID tag appears in the interrogation zone of the RFID reader. It is assumed here that the RFID system under consideration is the futuristic RFID system which is generating even more data than the existing RFID systems as a result of reduced number of collisions, enhanced computational capabilities and expanded infrastructure with even wider adoption of the technology. The data captured by the RFID reader is now sent to the data processing system composed of user interface and specific application to access the back end of the architecture. Now this huge data generated by this RFID system would be filtered, stored, semantically inferred before its actual usage. Now for all these purposes a cloud computing back end would be used which is connected to the front end via a network. The RFID data would actually be fed as an input to the back end of This proposed cloud computing architecture for scalable RFID in order to enhance the use of RFID data generated for greater purposes.
The Back End:
The back end of the proposed architecture is composed up of infrastructure (computers, data storage systems etc.), applications, services and servers. Here, on the cloud, the RFID data is semantically filtered according to a specific application of need, use its relative platform and infrastructure over the network and then stored on a specific server. Middleware and protocols are used in the cloud computing system here. A server administers the system and traffic. The middleware and protocols allow the networked resources to communicate to each other. Here, in the proposed architecture, the cloud computing systems need a lot of storage space which it requires to keep all its RFID systems clients' information stored. It makes a copy of all the RFID information and stores it. The copies enable the central server to access backup machines to retrieve data that otherwise would be unreachable. Now as a result of using cloud computing, the huge amount of data which was lost earlier at very early stages as a result of lack of storage and because of bearing low priority in the semantic inference may now be preserved and processed on the cloud to draw more intense conclusions. Moreover, it motivates more efficient utilization of the existing resources rather than demanding a need to incorporate new infrastructure and services. Apart from this data being processed according to specific need and scalable according, the back end of the proposed architecture also provides a way to extend the possibility of further extending the framework of infrastructure and scalable computational capabilities for RFID systems

IV. CONCLUSIONS
This paper proposes a cloud computing architecture framework which provides a direction towards the scalable RFID systems. It involves the commitment of the existing cloud computing systems to enhance the computational capabilities and wide adoption of the RFID systems on the cloud. Here, in the proposed architecture framework, virtualized resources as services over the network are melded with the RFID technology with limited resources. It may act as a prototype for the futuristic more scalable RFID systems since it evolves itself under the light of the cloud computing systems.

V. LIMITATIONS

More aspects may be investigated and incorporated before actual system implementation.

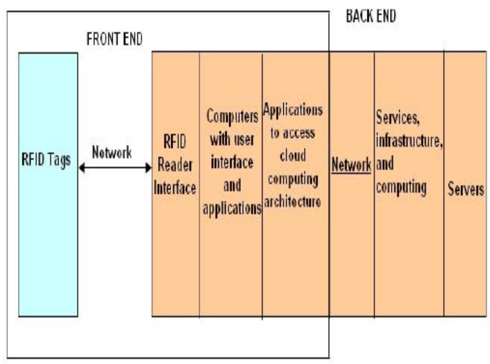


Figure 12.1: Architecture for Scalable RFID