RFID BASED AUTOMATED VEHCILE SURVIELLANCE SYSTEM

BY

NC Shoohira Aftab

NC Mariam Zia

PC Amna Khalid

NC Mannal Niazi

Submitted to the faculty of Computer Science, Military College of Signals, National University of Sciences and Technology, Rawalpindi in partial fulfillment for the requirements of a B.S degree in Computer Science

July 2010

Table of Contents

Abstract

RFID Based Automated Vehicle Surveillance System

Changes in the scale and pace of development of recent science and emerging technologies and the growing need to improve performance efficiency and standard of services and security combine to present a series of challenges and opportunities for exponentially growing organizations. The flexibility and power of wireless technologies like RFID offer solution to many of these growing business needs and demands. The chief objective of this system is to provide a flexible RFID technology based system that has an automatic identification method which can be used in sensitive organizations to provide a solution for the problems regarding the authentication and active monitoring of vehicle movement within the organization. The utility for such a system is to gauge and monitor the location and movement of automobiles. This system of actively monitoring, checking, tracking and reporting of vehicles within the organization bounds is completely automated. The ability to perform all these operations greatly expedites process that would otherwise cost considerable time and resources. With ability to save time and man power this software is a system that sensitive organizations will long leap benefit from.

Acknowledgements

All praise and acknowledgement be to Allah Almighty who is always there guiding us in the right direction and helping us in the best manner possible.

We thank our supervisor Maj. Dr. Asif Masood who has always been very cooperative and full of insight in all matters relating to the project or otherwise. He also deserves our hearty acknowledgement for his patience and optimism in coming up with solutions to what seemed like dead ends to us.

We also thank the entire computer science department including the faculty and administration, especially the helpful instructors for providing moral and technical support, cooperation and guidance to us without which we would not have made it this far.

It was a great experience to work alongside so many intelligent and helpful colleagues and friends who deserve our heartiest acknowledgement and thanks for their useful suggestions and creative ideas.

Above all, we are deeply thankful to our beloved parents for their patience, support and prayers that have made us who we are today.

1 Introduction

1.1 Introduction

This chapter throws light on the different aspects of the data capture technologies that are currently prevailing, world-wide. A broad brush regarding the extensive work, which has been conducted in the field of RFID, has been given in this chapter.

1.2 Radio Frequency Identification Concept

RFID is an automated data capture technology that can be used to electronically identify, track, and store information about groups of products, individual items, or product components. The technology consists of three key concepts, RFID tags, RFID readers and a data collection, distribution, and management system.

RFID tags contain information about a product or an identification number that corresponds to information that is stored in a database. The tags can be located inside or on the surface of the product or item. RFID readers interrogate or send signals to the tags and receive the responses. These responses are transferred to the data collection system. Lastly, data collection systems consist of computers running data processing software, which typically are networked with a larger information management system. RFID technology relies on the transfer of packets of information through radio waves or electromagnetic waves [1]. A typical RFID system is shown in Figure 1.1.

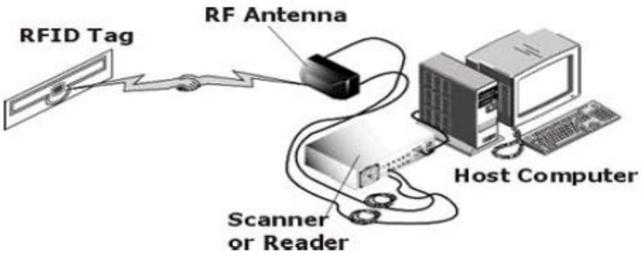


Figure 1.1: A typical RFID System

It has been the exponential growth in information and communications technologies coupled with the expansion of global production and trade that has resulted in RFID technology becoming useful for managing and tracking large shipments and product sales, and as a means of identification for security purposes and supply chain management. In future years the new worldwide setting is likely to transform the conventional ways of managing sensitive organizations, augmenting access to primary sources, such as the actual run time data generated by the vehicle movement itself enabling active monitoring.

It is still too early to anticipate the full impact of RFID technology as a solution to traffic problems because of its possible diverse applications. Our project will highlight the prospects for using it in this field and present a practical and useful application of this technology.

1.3 Objective

Our objective is to provide automatic authentication to the vehicle, actively monitoring the visitors, maintaining record of vehicle movement and providing assistance/information of record to the security officer through user friendly environment. Development of system which is robustness against tempering and misuse of RFID tags.

1.4 RFID Based Automated Vehicle Surveillance system Concept

Managing a busy parking lot can pose significant challenges, especially to a sensitive organization that also owns some of the vehicles in the premises. The area has to be secure, with barrier-enforced entrances and exits. It should be an automated efficient monitoring system that provides accurate authentication, active monitoring and vehicle tracking. Providing different queries regarding vehicle record, validation of the registered user and easy in-and-out access for drivers.

Purpose of the project is to develop a complete RFID vehicle tracking for the organization's parking lot and fleet of vehicles. Each exit and entrance points has gate equipped with RFID readers. Each vehicle has an RFID tag, which is applied on the windscreen.

The software will be based on the RFID information that we will get from different locations. RFID readers will be installed in such a way that it can cover all possible paths for vehicle movement. The information from the RFID readers will be transmitted to database through wireless LAN. For the limitation of time and cost RFID readers will not be installed physically. These will be simulated and will work for any scenario of vehicle movement. System will allow user to make different scenarios in order to test the system. All other modules will be designed such that it can be easily integrated with actual network of RFID readers.

As the vehicles move inside an organization, database will keep updating its record in real time. Vehicle Record will have two different categories registered vehicle and visiting vehicle. Registered vehicles are given their registration RFID tags which are authenticated at the time they enter or leave the premises of organization. Record contains the vehicle number its color, type and the user registration identification number. Visiting vehicles are the non registered vehicles. At the time of entrance, unregistered vehicles will be issued an RFID tag containing user identification number and a vehicle identification number. The possible route for the visiting vehicle is allotted and any deviation from the expected path if will be easily detectable.

1.5 Organization of Report

This project report has been divided into seven chapters. Chapter 1 gives an introduction to the technology used and to the RFID based Automated Vehicle Surveillance System. Chapter 2 gives the literature review. Chapter 3 is based on the detailed analysis of system requirements. Chapter 4 describes the system design and architecture and explains the way project is organized. Chapter 5 describes the system development with all the details of the system functions and explains the way they have been implemented.

2 Literature Review

2.1 Introduction

This chapter provides the details about the technology, its working principles and limitations.

2.2 RFID Technology

Short for radio frequency identification, RFID is a dedicated Short Range Communication (DSRC) technology. The term RFID is used to describe various technologies that use radio waves to automatically identify people or objects. RFID technology is similar to the bar code identification however one big difference between RFID and bar code technology is that RFID does not rely on the line-of-sight reading that bar code scanning requires to work.

2.3 Basic RFID System

It consists of three components:

a. An antenna or coil
b. A transceiver (with decoder)
c. A transponder (RF tag) electronically programmed with unique information

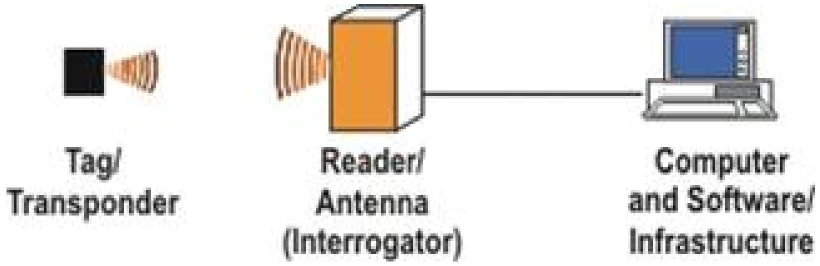


Figure 2.1: Basic RFID System

The antenna emits radio signals to activate the tag and to read and write data to it.

The reader emits radio waves in ranges of anywhere from one inch to 100 feet or more, depending upon its power output and the radio frequency used. When an RFID tag passes through the electromagnetic zone, it detects the reader's activation signal.

The reader decodes the data encoded in the tag's integrated circuit (silicon chip) and the data is passed to the host computer for processing.

2.4 RFID Tag

RFID tags two parts. One is an integrated circuit for storing and processing information, modulating and demodulating a radio-frequency (RF) signal, and other specialized functions. The second is an antenna for receiving and transmitting the signal.

2.4.1 Types of RFID Tags

Following are the type of tags.

1. Passive Tags
2. Active Tags
3. Zombie Tags
4. Semi Passive Tags

2.4.1.1 Passive Tags

Passive tags Use the radio frequency from the reader to transmit their signal. Passive RFIDs are tags that do not require a power source to operate. Instead, the reader for the RFID is operated on a voltage source. With this voltage source, the reader is able to construct a magnetic field when it senses a tag is near. From this, a current is induced through the magnetic field, which allows the tag to send its signal to the reader

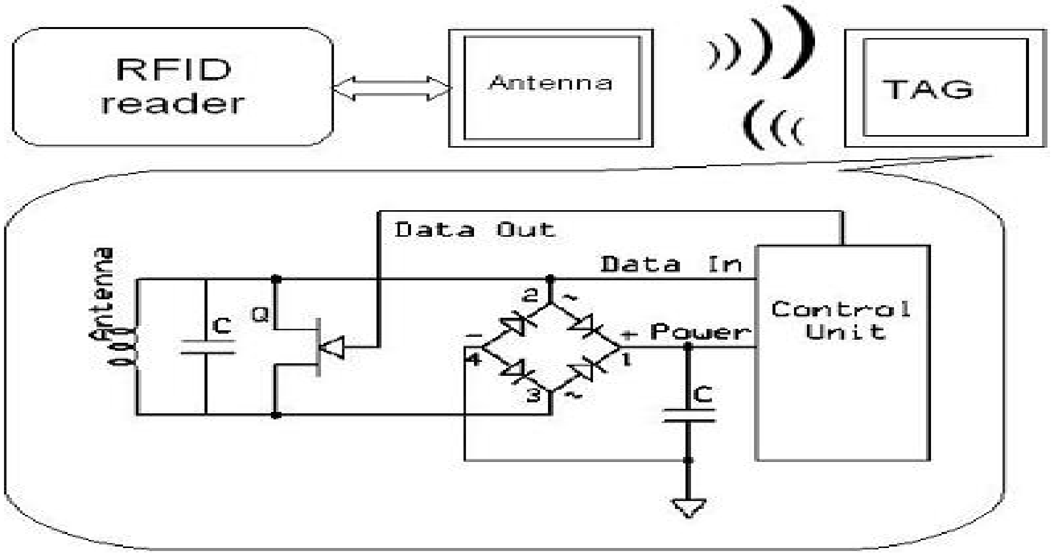


Figure 2.2: Internal Structure of Passive RFID tags.

Passive RFID tags vary in how they communicate data to RFID readers and how they receive power from the RFID reader's inductive or electromagnetic field. This is commonly performed via two basic methods:

Load modulation and inductive coupling in the near field—

In this approach the RFID reader provides a short-range alternating current magnetic field that the passive RFID tag uses for both power and as a communication medium. Via a technique known as inductive (or near-field) coupling' this magnetic field induces a voltage in the antenna coil of the RFID tag, which in turn powers the tag. The tag transmits its information to the RFID reader by taking advantage of the fact that each time the tag draws energy from the RFID reader's magnetic field, the RFID reader itself can detect a corresponding voltage drop across its antenna leads. Capitalizing on this phenomenon, the tag can communicate binary information to the reader by switching ON and OFF a load resistor to perform load modulation. When the tag performs load modulation, the RFID reader detects this action as amplitude modulation of the signal voltage at the reader's antenna. Load modulation and inductive coupling can be found among passive RFID tags using frequencies from 125 to 135 kHz and 13.56 MHz.

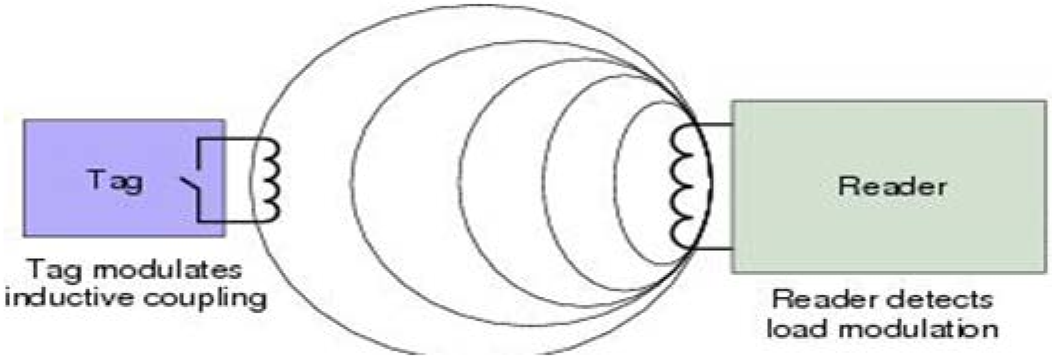

Tag
Tag modulates inductive coupling
Reader
Reader detects load modulation

Figure 2.3: Passive Tag Load Modulation

Backscatter modulation and electromagnetic coupling in the far field

In this approach, the RFID reader provides a medium-range electromagnetic field that the passive RFID tag uses for both power and a communication medium. Via a technique known as electromagnetic (or far-field) coupling, the passive RFID tag draws energy from the electromagnetic field of the RFID reader. However, the energy contained in the incoming electromagnetic field is partially reflected back to the RFID reader by the passive tag antenna. The precise characteristics of this reflection depend on the load (resistance) connected to the antenna. The tag varies the size of the load that is placed in parallel with the antenna in order to apply amplitude modulation to the reflected electromagnetic waves, thereby enabling it to communicate information payloads back to the RFID reader via backscatter modulation. Tags using backscatter modulation and electromagnetic coupling typically provide longer range than inductively coupled tags, and can be found most commonly among passive RFID tags operating at 868 MHz and higher frequencies

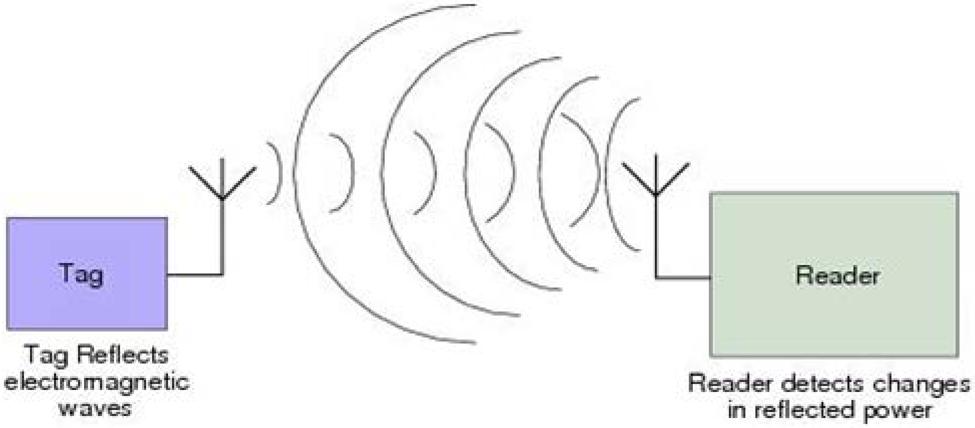

Tag
Tag Reflects electromagnetic waves
Reader
Reader detects changes in reflected power

Figure 2.4: Passive Tag Backscatter Modulation

The reading and writing depend on the chosen radio frequency and the antenna design/size. Passive tags have a variety of ranges depending on the antenna incorporated into the tag. Some have a read distance ranging from about 11 cm (with near-field) and up to approximately 10 meters (with far-field)

The lack of an onboard power supply means that the device can be quite small. Commercially available products exist that can be embedded in a sticker, or under the skin in the case of low frequency (Low FID) RFID tags.

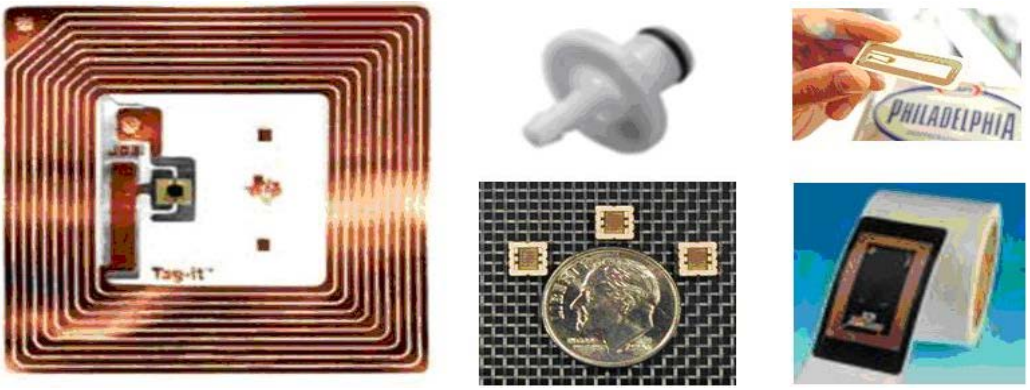


Figure 2.5: Low frequency RFID tags

2.4.1.2 Active Tags

An active RFID is similar to a passive RFID tag. It has similar circuitry and an antenna to that of a passive RFID. The main difference between the two is that the active version has a battery incorporated to its circuitry. This battery can be the sole or partial source of the tag's power supply.

The advantages of an active RFID include a longer range reading distance. The power supply can amplify the signal and transmit it to farther distances. In addition, the power source to an active RFID can also power other sensors on the RFID, enabling it to send certain signals under certain circumstances.

2.4.1.3 Zombie Tags

A tag that can be temporarily deactivated when it leaves the store is called Zombie tag. The process would work like this: you bring your purchase up to the register, the RFID scanner reads the item, you pay for it and as you leave the store, you pass a special device that sends a signal to the RFID tag to "die." That is, it is no longer readable. The "zombie" element comes in when you bring an item back to the store. A special device especially made for that kind of tag "re-animates" the RFID tag, allowing the item to reenter the supply chain

2.4.1.4 Semi Passive Tags

Semi-passive RFID tags overcome two key disadvantages of pure passive RFID tag designs:

a. The lack of a continuous source of power for onboard telemetry and sensor asset monitoring circuits.
b. Short range.

Semi-passive tags differ from passive tags in that they use an onboard battery to provide power to communication and ancillary support circuits, such as temperature and shock monitoring. It is interesting to note that although they employ an onboard power source, semi-passive RFID tags do not use it to directly generate RF electromagnetic energy. Rather, these tags typically make use of backscatter modulation and reflect electromagnetic energy from the RFID reader to generate a tag response similar to that of standard passive tags. The onboard battery is used only to provide power for telemetry and backscatter enabling circuits on the tag, not to generate RF energy directly.

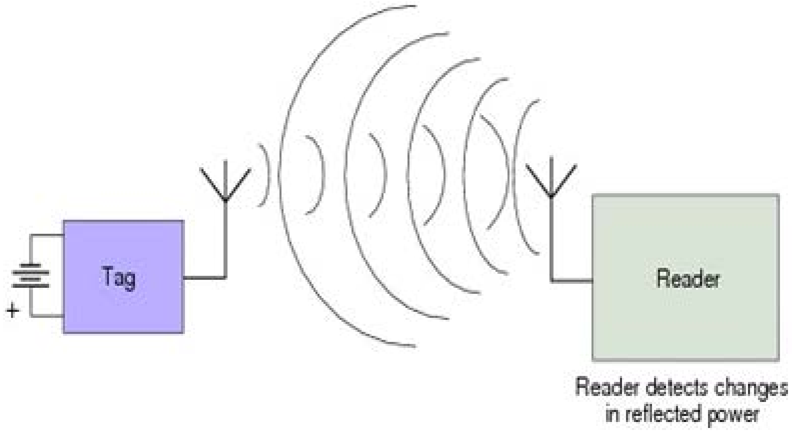


Figure 2.6: Back scatters Modulation in Semi-Passive RFID Tags

Semi-passive RFID tags can have a range of up to 30 meters with onboard lithium cell batteries lasting several years. Range is vastly improved over conventional passive RFID tags primarily because of the use of a backscatter-optimized antenna in the semi-passive design. Unlike a conventional backscatter-modulated passive RFID tag, the antenna contained in a semi-passive tag is dedicated to backscatter modulation and there is no dependence on the semi-passive RFID tag antenna to be a reliable conduit of power for the tag. Therefore, the semi-passive tag antenna can be optimized to make most efficient use of the backscatter technique and provide far better performance than purely passive RFID tag antenna designs.

Figure 2.7: Semi-Passive RFID Tags

2.5 RFID Reader

An RFID reader is a device that is used to interrogate an RFID tag. The reader has an antenna that emits radio waves; the tag responds by sending back its data. A number of factors can affect the distance at which a tag can be read (the read range). The frequency used for identification, the antenna gain, the orientation and polarization of the reader antenna and the transponder antenna, as well as the placement of the tag on the object to be identified will all have an impact on the RFID system's read range

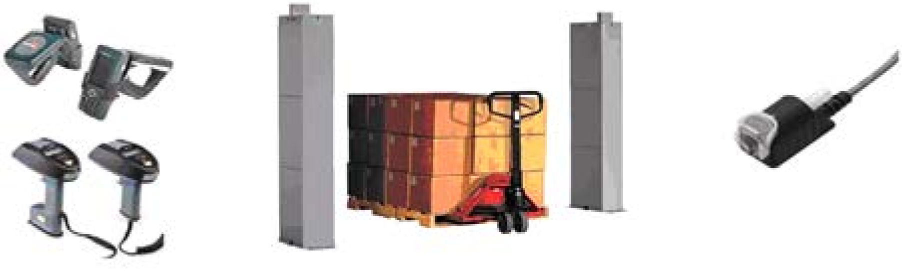


Figure 2.8: Some RFID readers

2.6 Common Problems With RFID

Some common problems with RFID are reader collision and tag collision. Collision, in the context of RFID tag reads, occurs when either several tags or several readers are simultaneously present in the same field.

Tag collision occurs when a reader attempts to simultaneously read several tags. Reader collision occurs when two or more readers are simultaneously present in the same field several solutions have been found such as using Aloha, Slotted Aloha, Framed slotted aloha, Q algorithm and CDMA based algorithm.

2.7 Applications

2.7.1 Asset Tracking

Asset tracking is one of the most common uses of RFID. Companies can put RFID tags on assets that are lost or stolen often, that are underutilized or that are just hard to locate at the time they are needed. Just about every type of RFID system is used for asset management. NYK Logistics, a third-party logistics provider based in Secaucus, N.J., needed to track containers at its Long Beach, Calif., distribution center. It chose a real-time locating system that uses active RFID beacons to locate container to within 10 feet.

2.7.2 Manufacturing

RFID has been used in manufacturing plants for more than a decade. It's used to track parts and work in process and to reduce defects, increase throughput and manage the production of different versions of the same product.

2.7.3 Supply Chain Management

RFID technology has been used in closed loop supply chains or to automate parts of the supply chain within a company's control for years.

As standards emerge, companies are increasingly turning to RFID to track shipments among supply chain partners.

2.7.4 Retailing

Retailers such as Best Buy, Metro, Target, Tesco and Wal-Mart are in the forefront of RFID adoption. These retailers are currently focused on improving supply chain efficiency and making sure product is on the shelf when customers want to buy it.

2.7.5 Payment System

RFID is all the rage in the supply chain world, but the technology is also catching on as a convenient payment mechanism. One of the most popular uses of RFID today is to pay for road tolls without stopping. These active systems have caught on in many countries, and quick service restaurants are experimenting with using the same active RFID tags to pay for meals at drive-through windows.

2.7.6 Security and Access Control

RFID has long been used as an electronic key to control who has access to office buildings or areas within office buildings. The first access control systems used low-frequency RFID tags. Recently, vendors have introduced 13.56 MHz systems that offer longer read range. The advantage of RFID is it is convenient (an employee can hold up a badge to unlock a door, rather than looking for a key or swiping a magnetic stripe card) and because there is no contact between the card and reader, there is less wear and tear, and therefore less maintenance.

2.8 How to Choose the Right System?

Different RFID system is suitable according to practical situation. To design and evaluate an RFID system consideration to the following factors.

The transceiving range between Transponder and Reader

One to One / Short range – Door watching / Pets / Anti-Counterfeit / M-Commerce

a. The Reader recognizes one object per time
b. Range - Several to 15cm
c. Frequency – 125 KHz (LF) and 13.56MHz (HF)

One to One / Medium range – WIP / Pallet Tracing / Library / EAS

a. The Reader recognizes one object per time

b. Range - Up to 120cm
c. Frequency - 13.56MHz (HF) and 915MHz (UHF)

One to One / Long Range – Park Lot Tracking / Container Tracking / Highway Tolling System

a. The Reader recognizes one object per time
b. Range – several to 10m
c. Frequency - 915MHz (UHF) and 2.45GHz (Micro Wave)

One too Many – Logistics Application

a. The reader need to recognize more than one entity per time
b. Range – Up to 5m
c. Frequency – 13.56MHz (HF), 915MHz (UHF) and 2.45GHz (Micro Wave)

2.9 Suggested System

SYNO Tag provides Passive RFID readers with ranges:

2.9.1 5-12 meters

2.9.1.1 Available Frequencies

ISM 902MHz~928MHzor 920MHz~925MHz

Customized: 860MHz~960MHz

2.9.1.2 Read Write Performance

Read Distance: 5~12 Meters (depends on tag and environment)

Read Speed: < 10 ms per 64 bits (single card)

Successful Read: Buzzer + LED

Write Distance: 3~6 Meters (depends on tag and environment)

Write Speed: < 30 ms per 8 bits (single card)

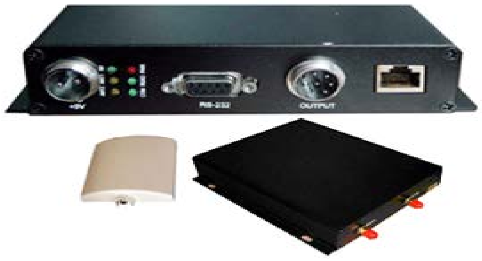


Figure 2.9 Different RFID Reader's interfacing mediums with PC

2.9.2 8 – 15 meters

2.9.2.1 Available Frequencies

ISM 902MHz~928MHz or 920MHz~925MHz

Customized: 860MHz~960MHz

2.9.3 Read Write Performance

Read Distance: 8~15 Meters (depends on tag and environment)

Read Speed: < 10 ms per 64 bits (single card)

Successful Read: Buzzer

Write Distance: 4~7 Meters (depends on tag and environment)

Write Speed: < 30 ms per 8 bits (single card)

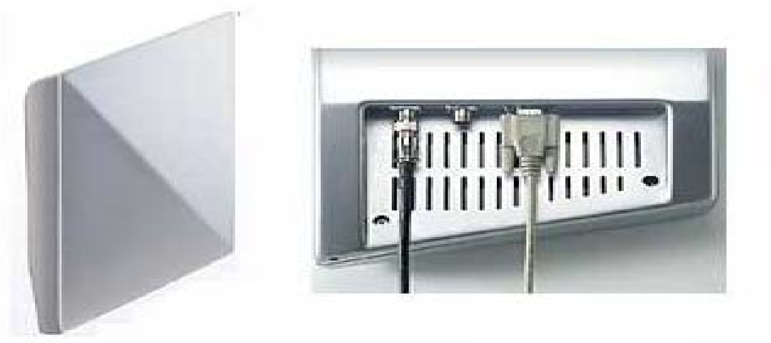


Figure 2.10: Serial Port connector

3 Requirement Specification

3.1 Introduction

Requirements analysis encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, such as beneficiaries or users. Systematic requirements analysis is also known as requirements engineering. Requirements analysis of the RFID based Automated Vehicle Monitoring And Surveillance System is done in this chapter..

3.2 Existing System

The present system used authentication of vehicle and monitoring uses conventional ways of operation. Authentication and tracking system has been manual, consuming a lot of manpower, time and money. Moreover the existing system is not accurate and efficient.

3.3 Problems in the Existing System

The existing security/checkup systems in Pakistan have many problems.

a. There is no check on the vehicle once it is entered in the organization.
b. All the checking is done manually so lot of manpower is required for this purpose.
c. If any suspicious activity is notice by the gatekeeper or any other guard it take time to inform the security officer of the organization.
d. It takes long time to authenticate the vehicle owner at the entrance. And if a visitor wants to take his car inside the organization it take even more time to allow him to enter or not.

e. Tempering with the car sticker is quite easy.

3.4 Requirements Consolidated

Based on the problems of the current system and security situation of Pakistan the requirement for our system was consolidated. It was desired that the system should provide complete tracking and active monitoring under all conditions throughout organization. It should be able to check the path violation. It should also keep a record of vehicle. The system should be able to authenticate the registered user. The system should also provide tag entry for the visitor along with allowed path entry. There is a need of a system that provides efficient tracking of vehicles based on the run time data generated by the vehicles. To detect for the vehicles that entered the prohibited area should be reported immediately (with in 1 sec) to security officer.

3.5 Specific Requirements

Specific requirements of a system are a comprehensive description of the intended purpose and environment for software under development. They fully describe what the software will do and how it will be expected to perform. System features like interfaces, functions and performance are catered in this portion.

3.6 External Interfaces

The input to the system comes in two ways. The first is the data that lies within the system. This is designed to be entered into the system manually and includes the cars and tags identification numbers. Only the clients will be allowed to enter this data.

The other part of input to the system is the information sent by the readers. This is given to the system to update its database about the flow and control. Both these information sections combine to produce information needed to monitor the traffic in organization. This is also used to actively monitor the system by checking whether a vehicle is on allowed path/location or not. The same information pool is used to generate other information like entry/exit time, vehicle detail and path violation history of any vehicle. The output of the system is provided through a user interface that provides consolidated information regarding vehicle records.

3.7 Functions

The system is required to perform these operations. The data is entered into the system, manually form the clients and automatically from the readers. The system offers an operation that checks whether the vehicle is on allowed path or not. This all information is used to generate the desired results needed by the security officer. The calculation process is internal to the system. Calculations should be done both on the server and the client(s) when requested for them.

3.8 Data Flow Diagram

3.8.1 RFID reader

RFID reader reads data as tag crosses its interrogation zone, this data through LAN is sent to the server. Server receives the data and updates the RFID READER table; the tag id is added in the appropriate row. Following is the data flow diagram for how data flows when read from reader

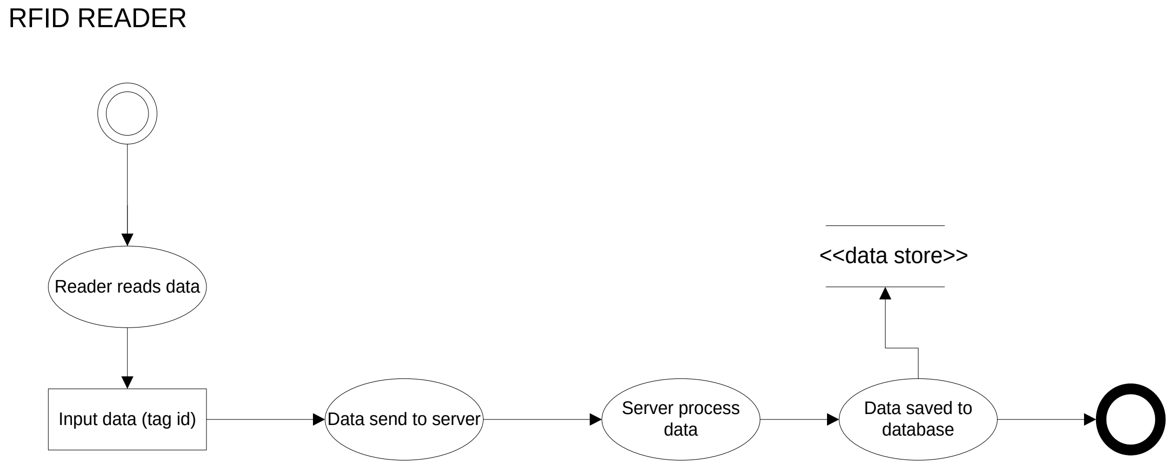


Figure 3.1: Data flow diagram for data for RFID reader

3.8.2 Client (Gatekeeper/ Administrator)

For adding a new user either to the list of registered users or visitors client adds information of the user and request server to retrieve available tags from database and assign it to that user. Client then send this information to the server. Server saves it to the database and sends a confirmation message to the client

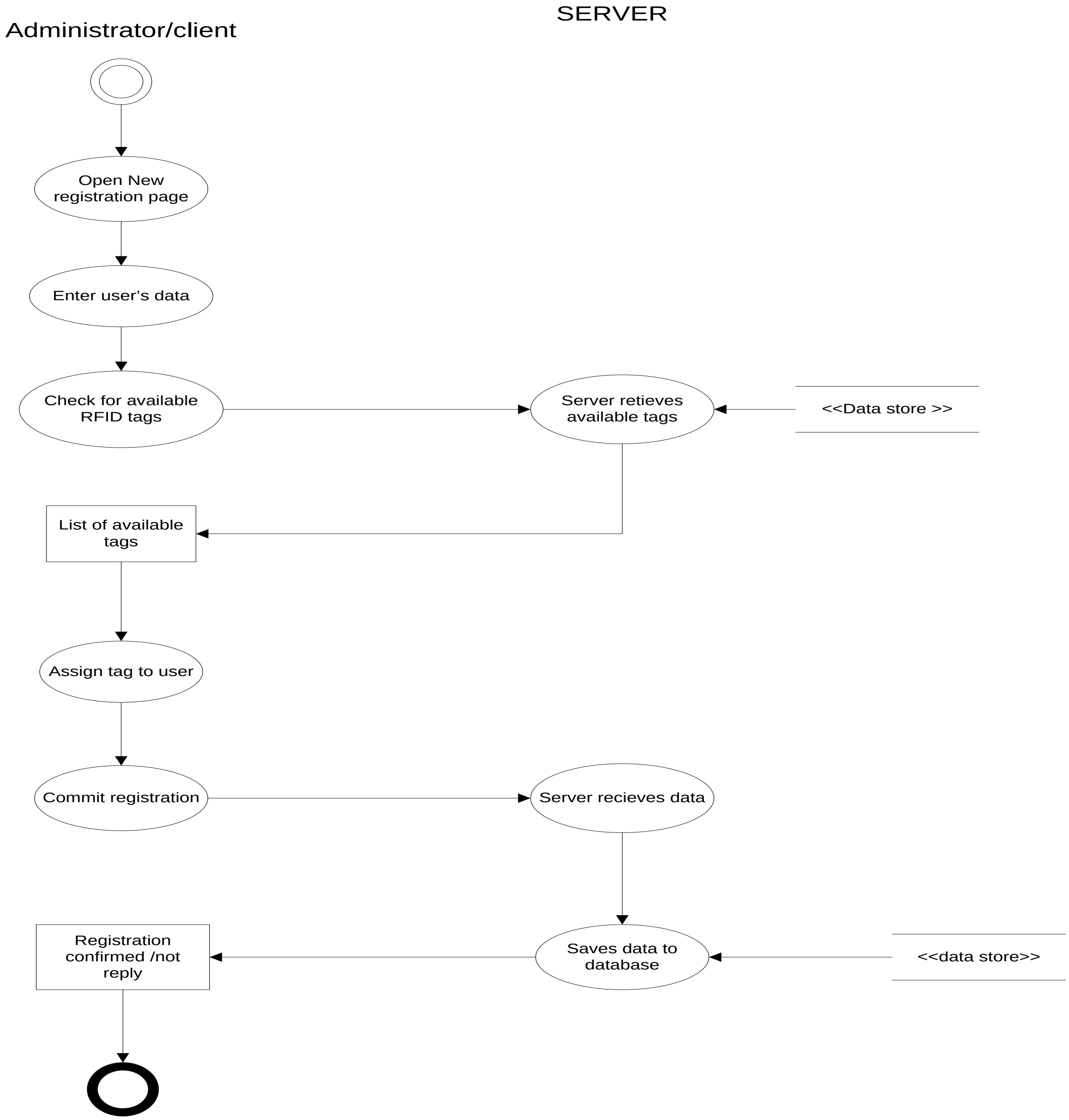


Figure 3.2: Data flow diagram for data new registration

3.8.3 Query from clients

Administrator and security officer can query the database. Dataflow diagram is shown below

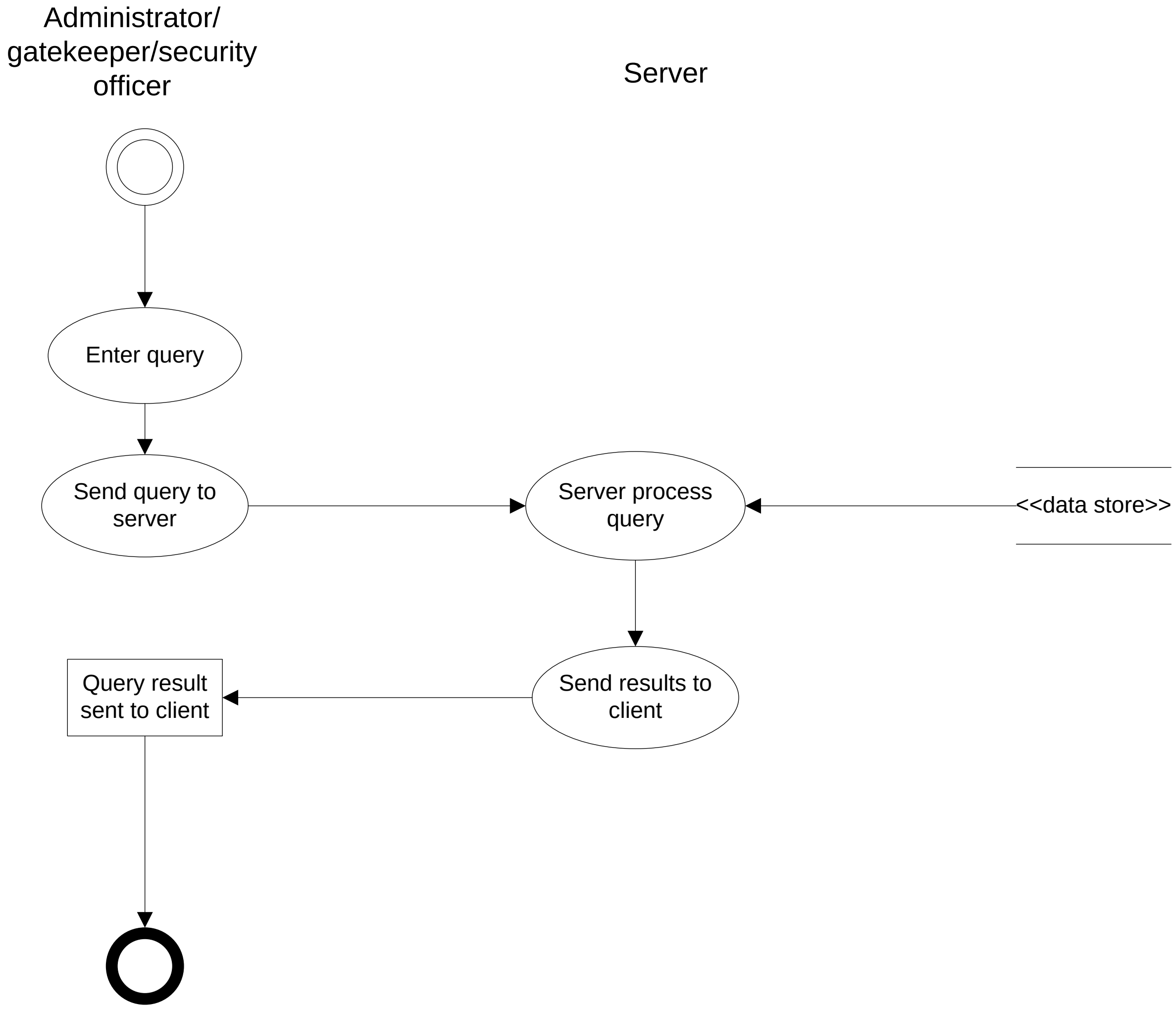


Figure 3.3: Data flow diagram for query

3.9 Use case

A use case in software engineering and systems engineering is a description of a system's behavior as it responds to a request that originates from outside of that system. In other words, a use case describes "who" can do "what" with the system in question. The use case technique is used to capture a system's behavioral requirements by detailing scenario-driven threads through the requirements. Use case for RFID Based Automated Vehicle Monitoring and Surveillance system is shown in figure 3.1

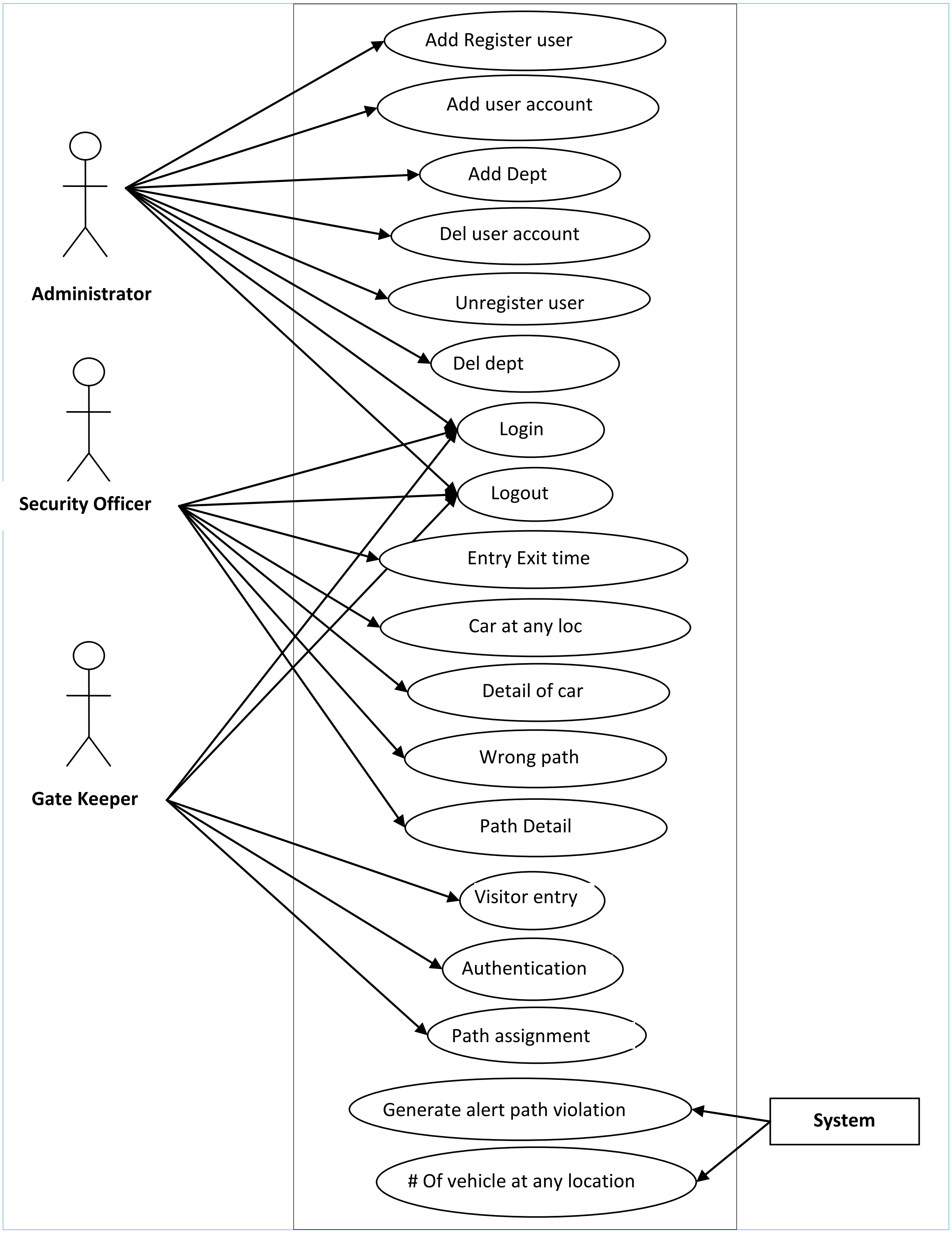


Figure 3.1: Use case Diagram

3.9.1 Use case: Add Register user

Scope: Register a new vehicle holder.

Level: User goal

Primary actor: Administrator.

Stakeholder and interest

Administrator: to register a vehicle holder so that he is authenticated when at gate.

Vehicle holder: to get an RFID tag so that on entering the organization he is authenticated and allowed to enter the organization premises.

Precondition

1. Administrator has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee

1. A new vehicle holder is registered to the system.
2. Database has been updated.

Main success scenario

1. Administrator enter a name
2. Administrator enter NIC number
3. Administrator browse picture.
4. Administrator select department.
5. Administrator select status.
6. Administrator enter vehicle Registration number.
7. Administrator select model of vehicle.
8. Administrator select type of vehicle.
9. Administrator select color of vehicle.
10. Administrator read a new RFID tag with RFID reader.
11. Administrator press register button.

12. System will check that NIC number and Registration number pair is unique
13. System makes an entry in database.
14. System will show a message record insert on administrator screen.

Alternate Scenario

12a NIC number and Registration number pair is unique system will display an error.

3.9.2 Use case: Add user account

Scope: To Add a new user account in the system.

Level: User goal

Primary actor: Administrator.

Stakeholder and interest:

Administrator: create a new user account for security officer and gatekeeper so that they can use that username and password to use the system.

Security Officer: to get a username and password so that using that can connect to the system.

Gatekeeper: to get a username and password so that using that can connect to the system.

Precondition:

1. Administrator has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee:

1. a new user account has been created
2. Database has been updated.

Main success scenario:

1. Administrator enters a username.
2. Administrator enters a password.
3. Administrator retype password.
4. Administrator select access rights.
5. Administrator press create account button.
6. System checks that username is unique.
7. System checks that Password entered in password and re-write password field are same.
8. System will make a new entry in database.
9. System display message that record is inserted.

Alternate Scenario:

1. Username is not unique. System will display an error message.
2. Password entered in password and re-write password field are not same. . System will display an error message.

3.9.3 Use case: Add Dept

Scope: to add a new department in a database.

Level: User goal

Primary actor: Administrator.

Stakeholder and interest:

Administrator: enter a new department so that in selecting vehicle holder department that department can be selected.

Precondition:

1. Administrator has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee:

A new department has been entered in the system database.

Main success scenario:

1. Administrator enters a new department name.
2. Administrator press add department button.
3. System checks that a unique department has been entered.
4. System makes an entry in the database.

Alternate Scenario:

Department name is not unique. System will display an error message**.**

3.9.4 Use case: Del user account

Scope: to delete a user account from a system

Level: User goal

Primary actor: Administrator.

Stakeholder and interest:

Administrator: to delete a user account so that security officer gatekeeper cannot use that account for using the system.

Precondition:

1. Administrator has been logged on.
2. Administrator selects the delete button.
3. System is connected to server via LAN
4. RMI is running.
5. System is up and running.

Success guarantee:

User account has been deleted from the system.

Main success scenario:

1. Administrator select user account.
2. Administrator press delete button.
3. System will delete the user account from the database.

3.9.5 Use case: Unregister user

Scope: To unregister a registered user so that his tag can't be used for authentication, and entering in the organization.

Level: User goal

Primary actor: Administrator.

Stakeholder and interest:

Administrator: to unregister a user from system.

Precondition:

1. Administrator has been logged on.
2. Administrator selects the delete button.
3. System is connected to server via LAN
4. RMI is running.

5. System is up and running.

Success guarantee:

Vehicle holder entry has been deleted from the system.

Main success scenario:

1. Administrator select user NIC number.
2. Administrator press delete button.
3. System will delete the user account from the database.

3.9.6 Use case: Del dept

Scope: to delete a department name from the system

Level: User goal

Primary actor: Administrator.

Stakeholder and interest:

Administrator: to delete a department name from the system, so that it does not appear in the department list.

Precondition:

1. Administrator has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee: department name is deleted from the system.

Main success scenario:

1. Administrator select user department name.
2. Administrator press delete button.
3. System will delete the user account from the database.

3.9.7 Use case: Login

Scope: to login the system in order to use the system functionalities.

Level: User goal

Primary actor:

1. Administrator.
2. Security Officer.
3. Gatekeeper.

Stakeholder and interest:

Administrator: to login the system to register vehicle holder, add department, add user account, delete registered user, delete department, and delete user account.

Security Officer: to login the system to see different information regarding the vehicles

Gatekeeper: to login the system in order to authenticate the registered user and add the information regarding visitor.

Precondition:

1. System is connected to server via LAN
2. RMI is running.
3. System is up and running.

Success guarantee:

User is login to system and respective window is opened according to its access rights.

Main success scenario:

1. User enters username.
2. User enter password.
3. User select login as option according to its access rights.
4. User press login button.
5. System checks the login and respective password from the system database.
6. System login in the user and open the window according to its access rights.

Alternate Scenario:

Username and respective password does not match. System will display error message.

3.9.8 Use case: logout

Scope: to log out the user from system.

Level: User goal

Primary actor:

1. Administrator.
2. Security Officer.
3. Gatekeeper.

Stakeholder and interest:

Administrator: to logout the system so that some other user login with another username and password

Security Officer: to logout the system so that some other user login with another username and password

Gatekeeper: to logout the system so that some other user login with another username and password

Precondition:

1. System is connected to server via LAN
2. RMI is running.
3. System is up and running.

Success guarantee:

User is logout and welcome window is displayed

Main success scenario:

1. User selects logout options.
2. System will logout the user and welcome screen will be displayed.

3.9.9 Use case: Entry Exit time

Scope: to see the entry ext time of ant vehicle

Level: User goal

Primary actor: Security Officer.

Stakeholder and interest:

Security Officer: to check the entry exit time of any vehicle.

Precondition:

1. Security Officer has been logged on.
2. System is connected to server via LAN

3. RMI is running.
4. System is up and running.

Success guarantee:

Entry exit time of the selected vehicle is shown to the security officer

Main success scenario:

1. Security officer select vehicle from the vehicle list.
2. Security officer press the entry/ exit button
3. System will show the entry ext time of the user

3.9.10 Use case: Car at any location

Scope: to see the cars at any location

Level: User goal

Primary actor: Security Officer.

Stakeholder and interest:

Security Officer: to see the car at any location.

Precondition:

1. Security Officer has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee:

Car at any location is shown to the security officer.

Main success scenario:

1. Security officer select location from the location list.
2. Security officer press the reader button.
3. System will show the car registration number that is present at that location.

3.9.11 Use case: Detail of car

Scope: detail of selected cars are shown

Level: User goal

Primary actor: security officer

Stakeholder and interest:

Security Officer: check the detail of car.

Precondition:

1. Security officer has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee: color, model, type of car is shown to the user.

Main success scenario:

1. Security officer select vehicle registration number from the vehicle registration list.
2. Security officer press the show car detail button.
3. System will show the car registration number color, model, type of vehicle.

3.9.12 Use case: Wrong path

Scope: wrong path of the entire vehicle can be seen

Level: User goal

Primary actor: Security Officer.

Stakeholder and interest:

Security Officer: to check which cars have taken wrong path, or enter in the prohibited area

Precondition:

1. Security Officer has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee:

Wrong path of all the vehicles are shown to the Security Officer.

Main success scenario:

1. Security Officer presses the wrong path option.
2. System will show the wrong path of the entire vehicle.

3.9.13 Use case: Path Detail

Scope: see the detail path of any registration number.

Level: User goal

Primary actor: Security Officer.

Stakeholder and interest:

Security Officer: to see the path detail of any car. So that the rout of that car is shown to the Security Officer

Precondition:

1. Security officer has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee:

Path detail of the selected car is shown to the Security Officer

Main success scenario:

1. Security Officer select registration number from registration number list.
2. Security Officer presses the show detail button.
3. System will show the detail of that selected registration number.

3.9.14 Use case: Visitor entry

Scope: enter the visitor information to the system database

Level: User goal

Primary actor: Gatekeeper.

Stakeholder and interest:

Gatekeeper: to enter a visitor in the database of the system.

Precondition:

1. Gatekeeper has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee: visitor information is added to the system.

Main success scenario:

1. Gatekeeper enters NIC number of the visitor.
2. Gatekeeper enter name of the visitor.
3. Gatekeeper enter Registration number.
4. Gatekeeper will enter the destination for the visitor
5. Gatekeeper will assign a tag.
6. Gatekeeper press the allow button.
7. System will add the information in the database.

3.9.15 Use case: Authentication

Scope: authenticate the registered vehicle holder.

Level: User goal

Primary actor: Gatekeeper.

Stakeholder and interest:

Gatekeeper: to allow the registered user enter the premises of organization.

Precondition:

1. Gatekeeper has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee: registered user is allowed to enter the organization

Main success scenario:

1. Gatekeeper will read the tag.
2. System will enter information in database.
3. Gatekeeper will allow the user to enter in the organization.

3.9.16 Use case: Path assignment

Scope: to add a new path in the system

Level: User goal

Primary actor: administrator.

Stakeholder and interest:

Administrator: add anew path in the system.

Precondition:

1. Administrator has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee: a new path is entered in the organization

Main success scenario:

1. Administrator will select the source
2. Administrator will select the destination.
3. Administrator will select the readers between the source and destination.
4. System will check for unique path.

5. System will enter the path in the database.

Alternate Scenario:

4. If the source and destination is not unique system will show error message.

3.9.17 Use case: Generate alert path violation

Scope: alert generation when a vehicle enters in prohibited area.

Level: User goal

Primary actor: System.

Stakeholder and interest:

Security Officer: path violation alert will be shown to the security officer.

Precondition:

1. Administrator has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee: alert is shown to the security officer.

Main success scenario:

1. Reader will read the tag.
2. System will enter the tag information.
3. System will check that reader is not allowed to that tag and alert will be generated.

3.9.18 Use case: No. of vehicle at any location

Scope: To see the number of car at any location in the organization.

Level: User goal

Primary actor: Security Officer

Stakeholder and interest:

Security Officer: see number of car at any location.

Precondition:

1. Security officer has been logged on.
2. System is connected to server via LAN
3. RMI is running.
4. System is up and running.

Success guarantee:

Number of vehicle is shown to the security officer.

Main success scenario:

Number of vehicle will be shown to the security officer at any time

3.10 Performance Requirements

The system is subject to some static and dynamic numerical requirements placed on the system and on readers' and clients' interaction with the system. The system's scope, presently, is a sensitive organization. The numbers of clients are presently limited to the number of Reader on the road. Dynamic Performance Requirements, such as response time of transactions, are dependent on the system's design, development and evaluation. These requirements can be accurately measured after development during testing/evaluation.

3.11 Design Constraints

The database design should adhere to the rules of a relational database for compliance with the international standards of operation. It should avoid redundancy at all costs and ensure data integrity. There are no design constraints for the system. It is subject to the developer’s choice.

Because of the limited hardware available, the simulation part of the system is developed in order to provide a graphical view of the system to enhance understanding and interaction with the system.

3.12 Standards Compliance

The software system should be developed in compliance with internationally accepted ISO15693 standard for development of RFID applications. ISO 15693 is a standard for Vicinity Cards, i.e. cards which can be read from a greater distance as compared to Proximity cards. The database for the system will follow the relational database rules.

3.13 Reliability

The system’s reliability is important for its functional working and availability. It should be reliable enough to work well with both the system’s data and the input data. Its reliability is most important in case of its output where accurate results and comparisons are required. It is required and expected to be available to the users all the time. Any chances of unavailability or malfunctioning should be minimized. This involves operations at the client as well as the server end.

3.14 Security

The system is limited by a few security constraints. The cars and tags definition and manipulation will be done with security checks. Also, only the authorized clients (terminals) would be given access to the server operations. System data should be critically checked to maintain data integrity. System backups should be taken manually, initially.

3.15 Maintainability

The system is intended to be an easy-to-maintain one. The data input is done, partially manually and partially through readers input. Both ways, the information should go into the system's database and recorded and maintained there. Its operations are consistent and will need infrequent maintenance, when needed.

3.16 System Mode

The system should be designed to operate partially in manual and partially in automated mode. Part of input, car information coming from the readers, is automatic. Rest of the input, coming from the gatekeeper and the security officer are manual. Using the given information, the system should generate desired outputs.

3.17 Objects and External Features

The main objects that the system should interact with are the RFID readers and the vehicles. These and other related entities should be explained and modeled in the system's design. There are no external features that interact with the system. It is a standalone system that interacts only with its own application components and objects.

3.18 Response

The system response time should be very less since the organizations security at stake. It should acknowledge the information input to it by the readers. This will also confirm the successful input of the data and its updating. Outputs should be generated based on the input sent by the reader and the gatekeeper.

4 System Design and Architecture

4.1 Introduction

The software design and architecture of a program or computing system is the structure or structures of the system, which comprise software components, the externally visible properties of those components, and the relationships between them. This chapter covers the layout of our project. All resources available, the components designed and their link with each other is explained here in detail.

4.2 System Components and Layout

Our project was divided into components for its design namely RFID Tags, RFID Readers, Server and Clients. Figure 4.1 explains the layout of the system components and their relation with each other. It also shows the working of our system when all components are put to work together.

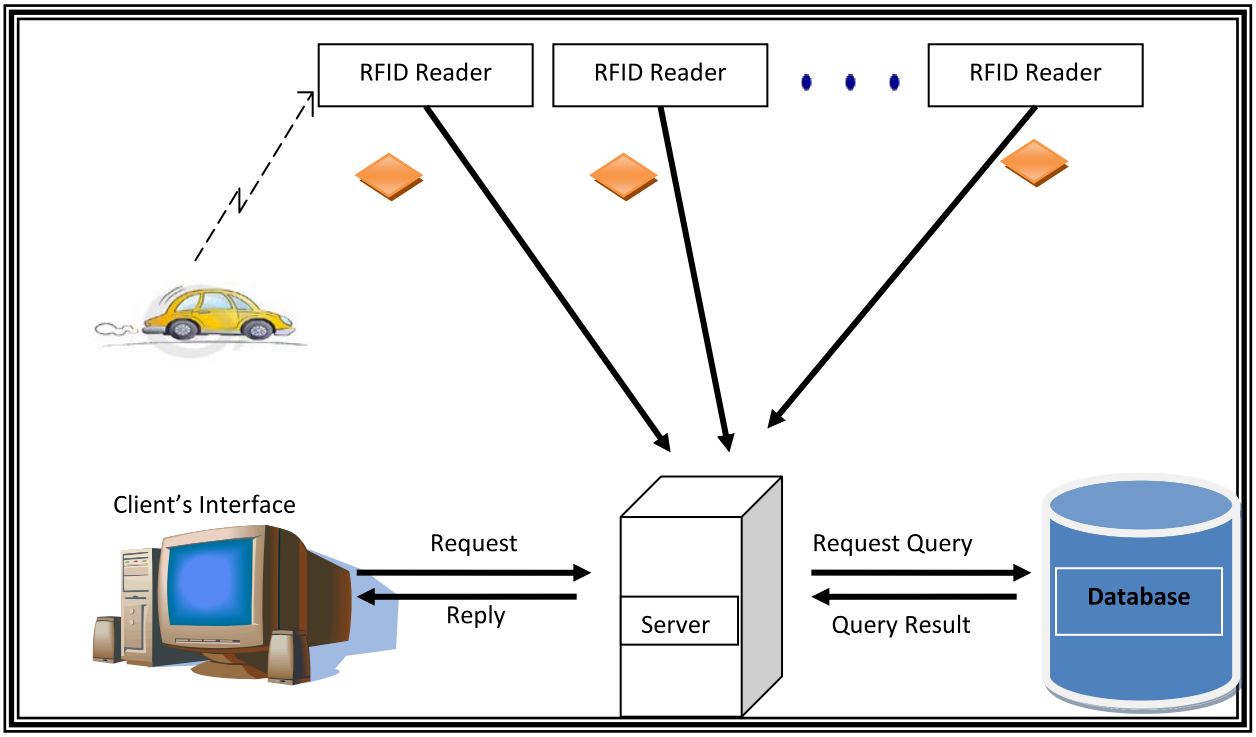


Figure 4.1: RFID Surveillance System Layout

4.3 System Design

RFID surveillance system is designed as independent software that works at the application layer. At the implementation layer, it is integrated with the hardware layer to establish communication with the RFID layer. Figure 4.2 explains system design where the RFID layer handles connection between the system software and the hardware exchanging information with the tags and reader.

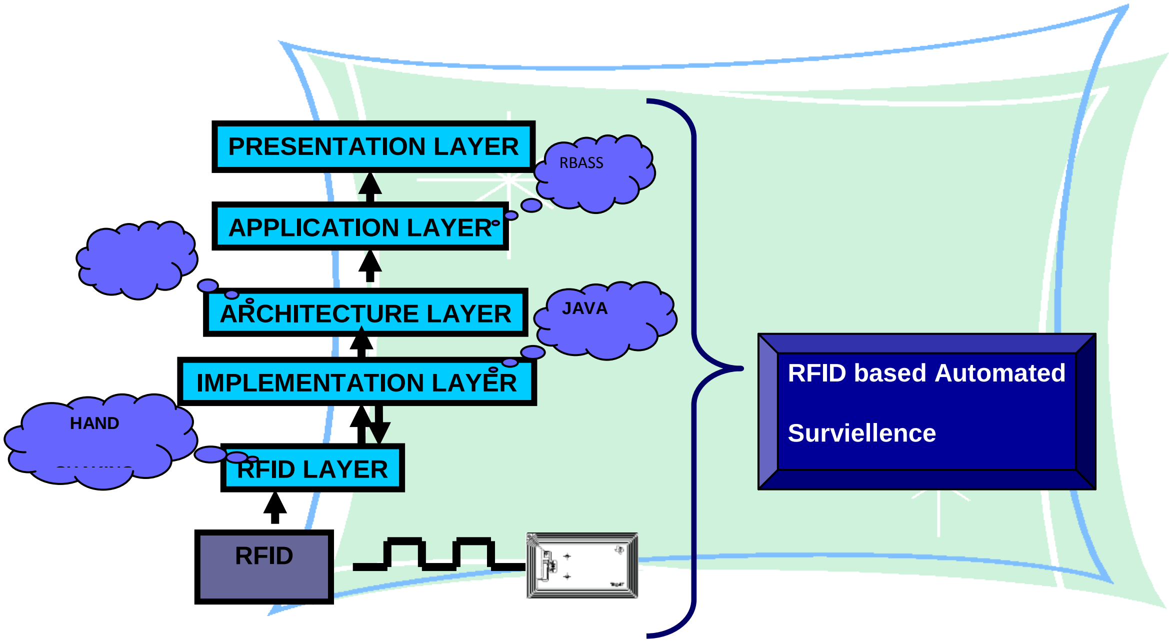


Figure 4.2: System Design

4.4 System Architecture

RFID Based Automated Surveillance System acts as an independent software integrated with RFID. It uses REMOTE METHOD INVOCATION technique for a distributed design of the system and is built on CLIENT SERVER PARADIGM. Two servers and one client are developed to perform and access operations respectively. The comprehensive system architecture is shown in Figure 4.3.

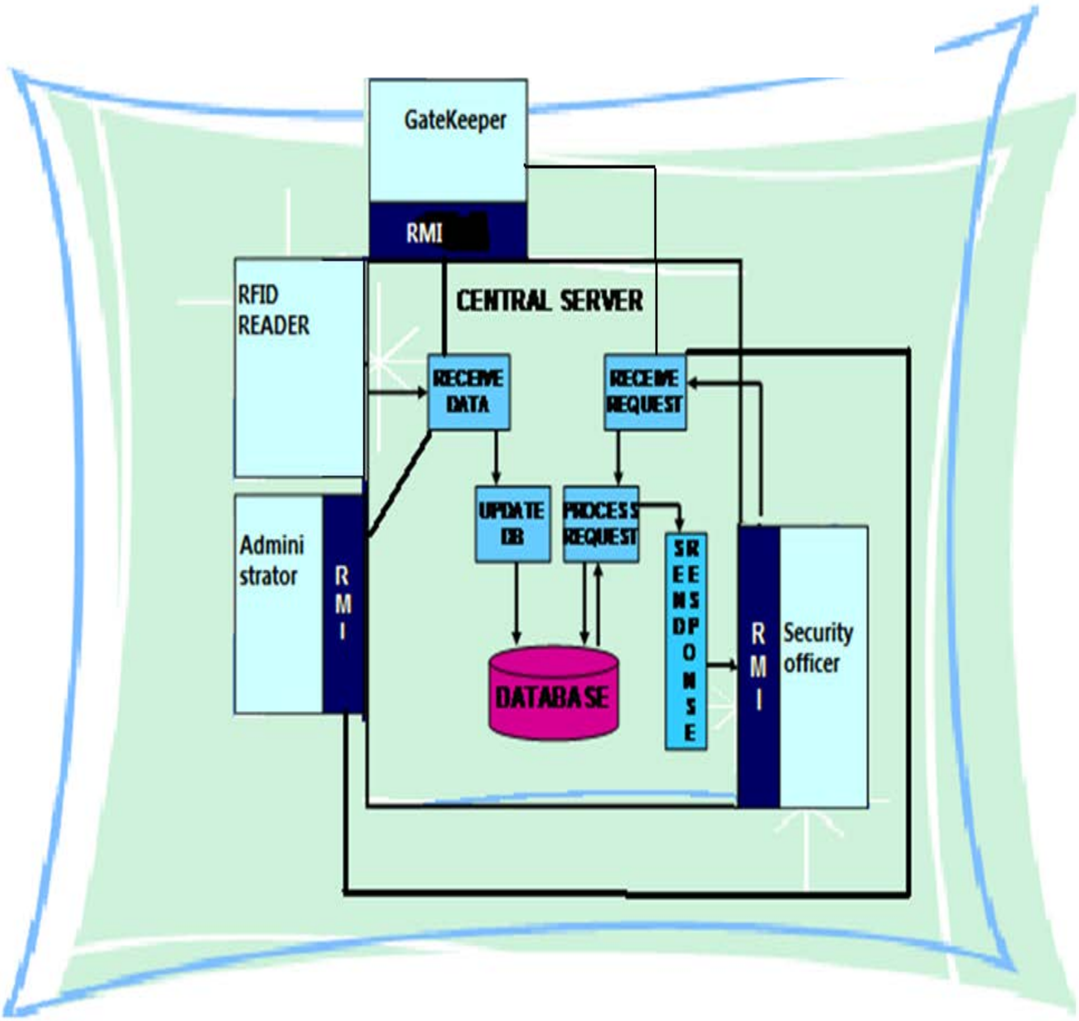


Figure 4.3: System Architecture

4.5 Remote Method Invocation

RMI applications often comprise two separate programs, a server and a client. A typical server program creates some remote objects, makes references to these objects accessible, and waits for clients to invoke methods on these objects. A typical client program obtains a remote reference to one or more remote objects on a server and then invokes methods on them. RMI provides the mechanism by which the server and the client communicate and pass information back and forth. Such an application is sometimes referred to as a distributed object application.

Distributed object applications need to, firstly locate remote objects. Applications can use various mechanisms to obtain references to remote objects. For example, an application can register its remote objects with RMI's simple naming facility, the RMI registry. Alternatively, an application can pass and return remote object references as part of other remote invocations. Secondly, they need to communicate with remote objects. Details of communication between remote objects are handled by RMI. To the programmer, remote communication looks similar to regular Java method invocations. And lastly, they have to load class definitions for objects that are passed around. Because RMI enables objects to be passed back and forth, it provides mechanisms for loading an object's class definitions as well as for transmitting an object's data.

Figure 4.4 depicts an RMI distributed application that uses the RMI registry to obtain a reference to a remote object. The server calls the registry to associate (or bind) a name with a remote object. The client looks up the remote object by its name in the server's registry and then invokes a method on it. It also shows that the RMI system uses an existing web server to load class definitions, from server to client and from client to server, for objects when needed.

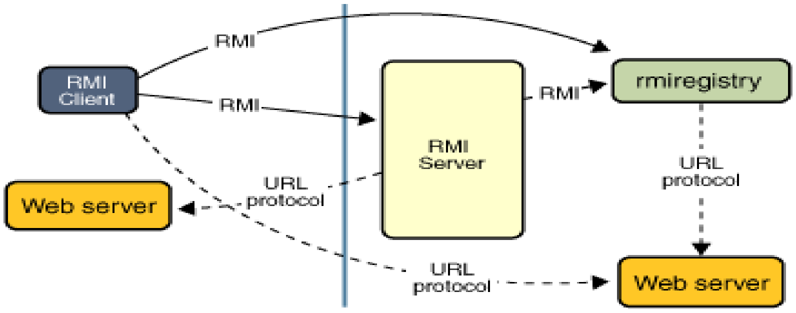


Figure 4.4: Remote Method Invocation

4.5.1 Advantages of RMI

One of the central and unique features of RMI is its ability to download the definition of an object's class if the class is not defined in the receiver's Java virtual machine. All of the types and behavior of an object, previously available only in a single Java virtual machine can be transmitted to another, possibly remote, Java virtual machine. RMI passes objects by their actual classes, so the behavior of the objects is not changed when they are sent to another Java virtual machine. This capability enables new types and behaviors to be introduced into a remote Java virtual machine, thus dynamically extending the behavior of an application.

4.5.2 Remote Interfaces, Objects, and Methods

Like any other Java application, a distributed application built by using Java RMI is made up of interfaces and classes. The interfaces declare methods. The classes implement the methods declared in the interfaces and, perhaps, declare additional methods as well. In a distributed application, some implementations might reside in some Java virtual machines but not others. Objects with methods that can be invoked across Java virtual machines are called remote objects.

An object becomes remote by implementing a remote interface, which has a remote interface extends the interface java.rmi.Remote and each method of the interface declares java.rmi.RemoteException in its throws clause, in addition to any application-specific exceptions.

RMI treats a remote object differently from a non-remote object when the object is passed from one Java virtual machine to another Java virtual machine. Rather than making a copy of the implementation object in the receiving Java virtual machine, RMI passes a remote stub for a remote object. The stub acts as the local representative, or proxy, for the remote object and basically is, to the client, the remote reference. The client invokes a method on the local stub, which is responsible for carrying out the method invocation on the remote object.

A stub for a remote object implements the same set of remote interfaces that the remote object implements. This property enables a stub to be cast to any of the interfaces that the remote object implements. However, only those methods defined in a remote interface are available to be called from the receiving Java virtual machine.

4.6 System Server

This is the main server where the database of the entire system is maintained. It receives traffic information from the RFID readers, gatekeeper and administrator and automatically updates the. It receives and processes requests from the CLIENTS (security officer and administrator) and responds to them, thus providing them its services. It also used to establish communication between the server and the RFID equipment. It provides communication with the readers.

4.7 System Clients

This is the part of the application that sends vehicle information to the CENTRAL SERVER to register vehicle on the System. It requests the services of the CENTRAL SERVER to perform the available functions. These operations are performed through a connection with the CENTRAL SERVER'S database. There are four different clients.

4.7.1 RFID Reader

All the readers connected to the wireless LAN connect to server and receives data read from readers. This data is stored in the database at the server, there are multiple readers installed at path ways to provide active monitoring and to check for any path violation.

4.7.2 Gatekeeper

Gatekeeper authenticates vehicles entering in the organization and assigns tags to the visitors and assigns them paths. Gatekeeper first connects to RFID reader and displays the authentication data of the tag or otherwise if the tag read is not already added to the system it takes information (such as NIC , name ,Reno of car etc) assigns it paths that vehicle is allowed to follow and adds it to the system.

4.7.3 Administrator

Administrator adds registered users to the system and assigns them paths. It also makes new account for new gatekeepers or security officers. Administrator can query any record or any detail of cars in the system .It can view situation of any location the number of cars at any place and other time related queries.

4.7.4 Security Officer

Security offices have a view to see all the readers and cars within range of one reader. And path violation by a car shows security officer an alert. Security Officer can query any record or any detail of cars in the system .It can view situation of any location the number of cars at any place and other time related queries

4.8 Database Design

Relational Model is today the primary data model for commercial data-processing applications. The knowledgebase, as shown in Figure 5.5, has been designed for minimizing the response time and to maximize the throughput and the operational speed so as to update the DB swiftly and avoid functional problems.

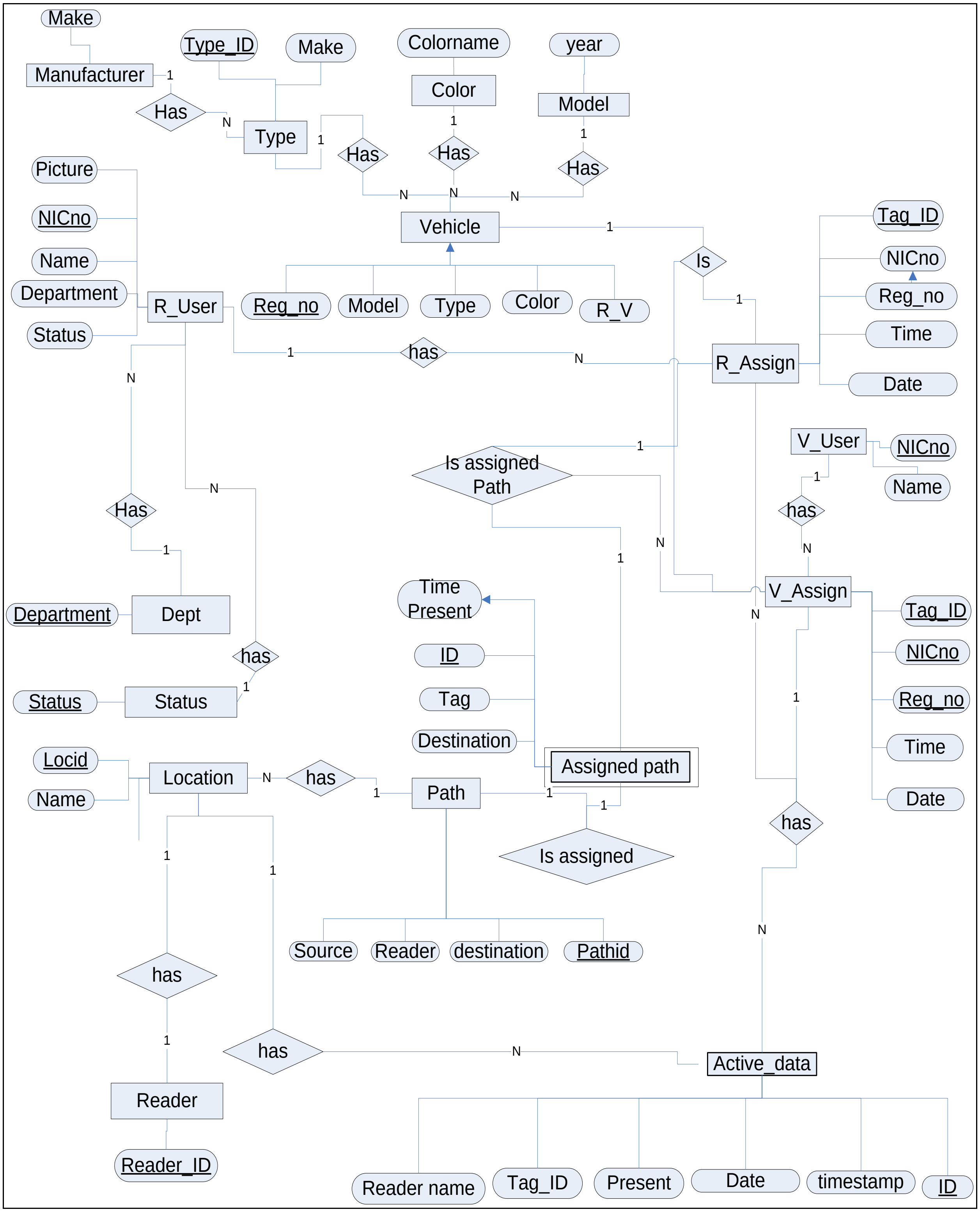


Figure 5.5: Database Design Diagram

4.9 UML Design

Modeling has been an essential part of engineering, art and construction for centuries. Complex software designs that are difficult to describe textually can readily be conveyed through diagrams. Modeling provides three key benefits: visualization, complexity management and clear communication. UML, the Unified Modeling Language, is a visual language for specifying, constructing, and documenting the artifacts of systems. UML was approved by the OMG as a standard in 1997. Over the past few years there have been minor modifications made to the language.

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. It is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software

Using the UML modeling, a class diagram has been developed for the system, the class diagram is shown in Figure

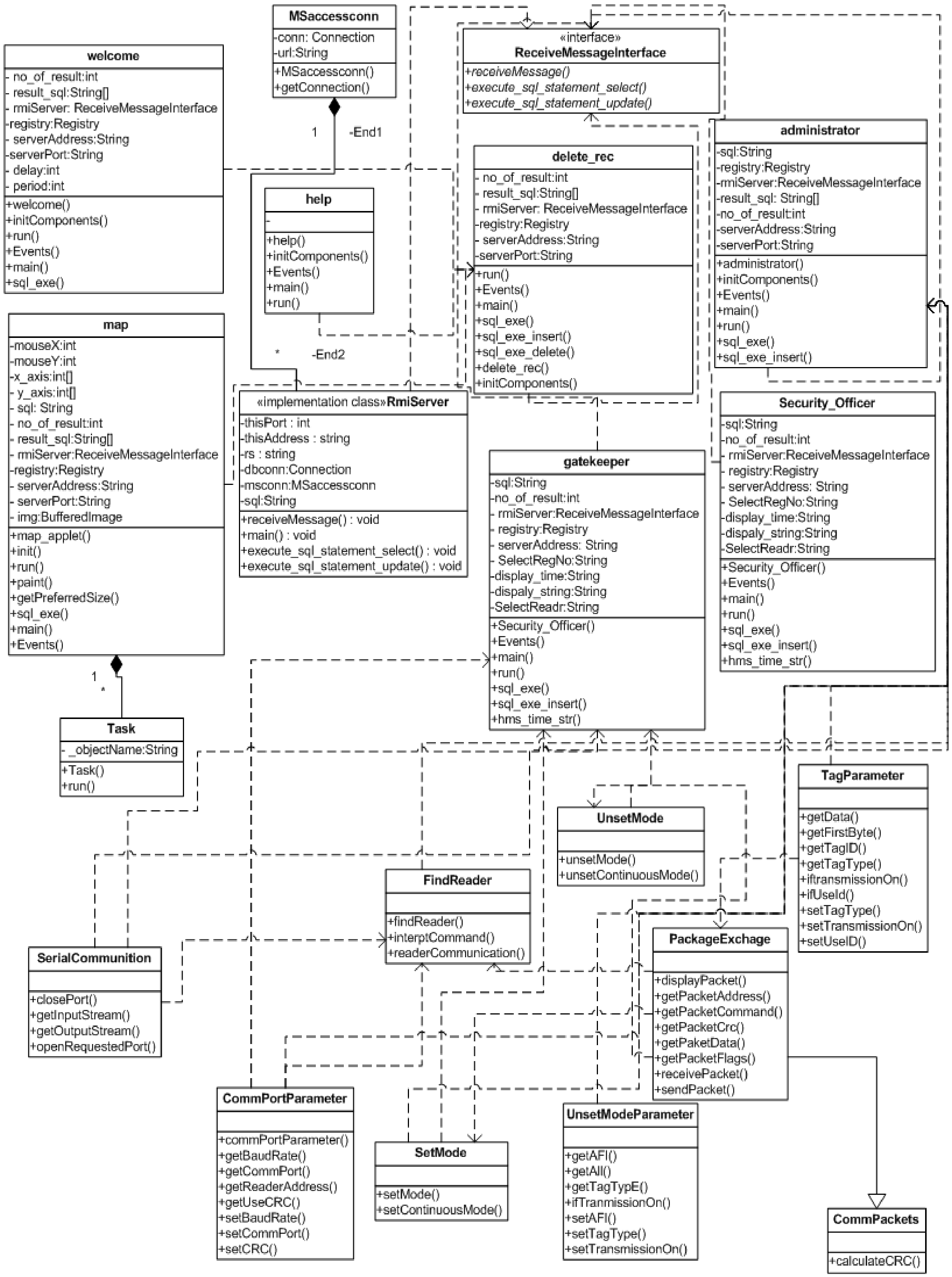


Figure 5.6 UML Class Diagram

5 System Implementation

5.1 Introduction

This chapter gives a closer look at the detailed aspects of system implementation. The chapter starts with a discussion about the software and hardware used and its configuration. Next, the system modules developed are explained.

5.2 Tools and Platforms

The system is developed using the following languages: Java, SQL. The software tools used for system development are JDK 1.6.0, Netbeans 6.1, MySQL.

5.3 RFID Reader

Because of limited resources, Right Tag USB Hand-Held RFID Reader which is used in this system is shown in Figure 5.1.

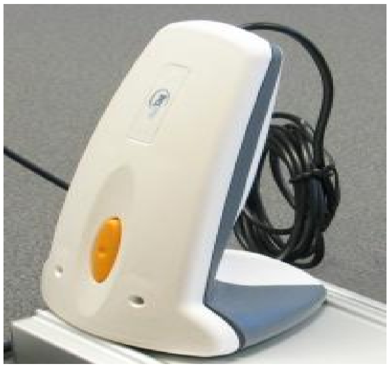
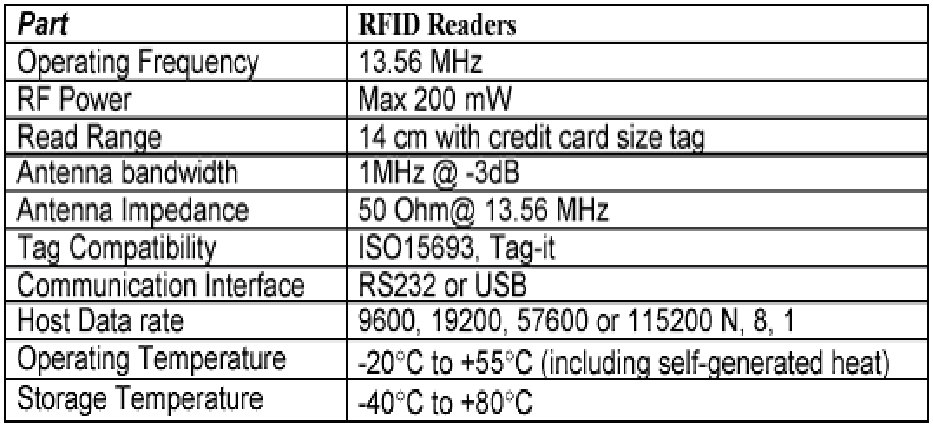


Figure 5.1: Right Tag Reader

This reader is used for collecting the runtime data from the traffic. In real time such readers are to be deployed along road sides. The data sheet of the used reader is given in Figure 5.2.



Part	RFID Readers
Operating Frequency	13.56 MHz
RF Power	Max 200 mW
Read Range	14 cm with credit card size tag
Antenna bandwidth	1MHz @ -3dB
Antenna Impedance	50 Ohm@ 13.56 MHz
Tag Compatibility	ISO15693, Tag-it
Communication Interface	RS232 or USB
Host Data rate	9600, 19200, 57600 or 115200 N, 8, 1
Operating Temperature	-20°C to +55°C (including self-generated heat)
Storage Temperature	-40°C to +80°C

Figure 5.2: Right Tag Reader Data Sheet

5.4 Memory Organization of RFID Tags

User data is read and stored in a 256-bit nonvolatile user memory that is organized in 64 blocks. Each block with 32 bit is user programmable and can be locked individually to protect data from modification. Once set, the lock bit cannot be reset. The user memory is field programmable per block. Two levels of block locking are supported: Individual block locking by the user (U) or individual block locking of factory programmed data (F) during manufacturing. Bit 2 of the “Block Security Status” byte defined in ISO 15693-3 is used to store the Factory Lock Status of the Block. Factory Block locking irreversibly protects the locked data from any further reprogramming. A factory-programmed block contains the IC reference and the physical memory info (Block size and Number of Blocks). This is shown in Figure 5.3

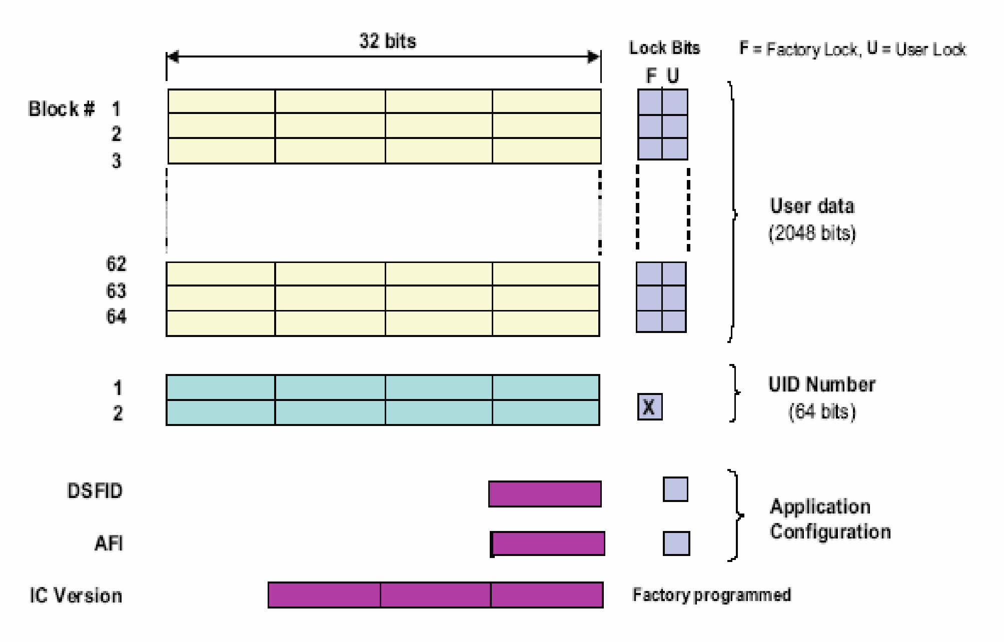


Figure 5.3 Memory Organization of Tag-It HF-I plus Transponder

5.5 RFID Reader Implementation

RFID reader has been implemented using java API for communication that is javax.com . Classes ComPortIdrntifier and SerialPort have been used. The Communication.jar package handles all the serial port access, protocol command composing and decomposing and interpreting the protocol command bytes for user convenience on standard output device. The package consists of 3 source files:

GeneralParameters.java: contains a class that keeps track of the general parameters that may be included from command line for every application. The class has methods to parse the command line to find suitable parameters and also methods to retrieve these parameters later. The general parameters are: COM port name, baud rate, whether the protocol uses addressed mode and whether CRC is used.

SerialCommunication.java: contains a class that opens and configures the port and assigns InputStream and OutputStream for further data exchange. It has methods to get the InputStream and OutputStream.

PacketExchange.java: contains classes that handle the binary protocol: defines command types and their names, has method for CRC calculation, has methods for packet composing, sending, receiving and decomposing, has methods to get the properties of last data packet and also has method for interpreting the last communication packet on the standard output

5.5.1 Protocol Description

5.5.1.1 Data exchange

Normally, data exchange between RF-ID Reader and PC is initiated by PC. (There are few exceptions: Reader sends a "welcome message" after power-up and can be configured to initiate some transfers itself).

Normally, the data exchange is as follows:

1. PC sends a request to the Reader.
2. Reader processes the request and does the desired actions.
3. Reader sends a response to indicate if the actions were successful.

5.5.1.2 Data fields

The protocol format is raw binary. Every REQUEST / RESPONSE starts with a 0x02 character, which indicates the start of data packet. The possible fields for requests and responses are:

1. 0x02 (mandatory)
2. Flags (1 byte, mandatory)
3. Reader address (1 byte, optional)
4. Command (1 byte, mandatory)
5. Length of data n (1 byte, mandatory)
6. Data (n bytes, n may also be 0)
7. CRC (2 bytes, optional)

5.5.1.3 Flags

The Flags byte indicates the presence of Reader address byte (ie if the command is addressed to a specific reader or any reader listening to the command), the presence of CRC bytes.

Binary format of Flags byte: 00000CBA

Bit A: set, if CRC bytes are present in the request / response.

Bit B: set, if the command is addressed to a specific reader (ie address byte is present in the request / response).

Bit C: Error flag, currently only the Reader can set this flag to indicate that something went wrong when processing the request from PC.

5.5.1.4 Command byte

Selects the purpose of the message and will determine the layout of data field in the request / response. A response by the Reader is always with the same command byte than it was in the request.

5.5.1.5 CRC

If the CRC is present in the request, it will also be included in the response. The CRC is calculated on all the bytes in the request / response excluding the 0x02 character at the beginning and the CRC itself. The 16-bit CRC is written MSB first into the message.

5.5.1.6 Setting the readers mode

To set the readers mode COMMAND BYTE is set to 0x05 . For command byte 0x05 the request packet format is

Request: 0x02 0x00 0x05 0x01 0xXX

XX is the new mode (1 byte). Bits of it have the following meaning: 0GFEDCBA

Bit A: Send TAG ID automatically - if set, the Reader always sends the TAG ID after the TAG has been read by pressing the pushbutton of the Reader.

Bit B: Send TAG Data automatically - if set, the Reader always reads and sends all the TAG data after the TAG has been read.
Bit C: Continuous read mode - if set, the Reader is in continuous read mode and sends all TAG IDs / Data whenever a TAG comes present in the Reader's antenna field.
Bit D: Disable button - if set, the pushbutton of the Reader is disabled.
Bit E: If set, the reader identifies all tags in the field when the pushbutton is pressed. If cleared, only the first tag will be identified.
Bit F: If set, the transmitter remains ON after pressing the pushbutton, otherwise it will be switched OFF.
Bit G: If set, the reader will send tag ID in protocol format (bits A and B have no meaning in such case).

5.5.1.7 Setting the reader to continuous mode

For RFID to work first there is an initial handshaking between the reader and and the PC . PC sends a packet data to reader with command byte set to 0x01 and data of zero bytes . Format for request packet for command bte 0x01 is
Request: 0x02 0x00 0x01 0x00

So reader receives this gets the version and senf a response back with command byte set to 0x01 along with the version number . the response packet format is:
Response: 0x02 0x00 0x01 0xnn 0xXX 0xYY 0xZZ 0x0D 0x0A (name of reader) 0x0D 0x0A (Version string) 0x0D 0x0A
0xXX, 0xYY and 0xZZ select the version of firmware. Interpret these bytes as follows: translate them into decimal numbers and show as "XX.YY.ZZ".

Next to set the reader in continuous mode PC creates a packet with command byte 0x05 that is used to set working mode of the reader .
the request packet is

Request: 0x02 0x00 0x05 0x01 0xXX
XX is the new mode (1 byte). Bits of it have the following meaning: 0GFEDCBA

Bit C and G is set to 1 so that the reader works in continuous mode and G is set to 1 so that so that data is received in protocol format .

Reader receives this packet sets the mode to continuous mode and sends a packet back to computer with command byte set to 0x05 indicating that continuous mode is set . after that if any tag comes in the range of reader its tag ID is sent to the computer

5.6 Client Implementation

Client application is implemented in Java .Functions for database connection, updating the database and for manipulating the data are called here. Which in turn connect to server and receive data from server through RMI (Remote Method Invocation).

5.6.1 Front End Development

The client is that application module of the system which is designed to be implemented on terminals of all clients. The user interface is designed and developed to be user friendly so that it is easy to operate keeping in view the nominal computer literacy level of the expected user. A GUI is developed that provides all the operations of login to the system and establish a connection between client and server.

Depending on the access right the respective window is shown. The data is received from GUI (Graphical User Interface), then from function call send to the client. All client uses RMI to contact server for querying the database

5.6.2 Back End Development

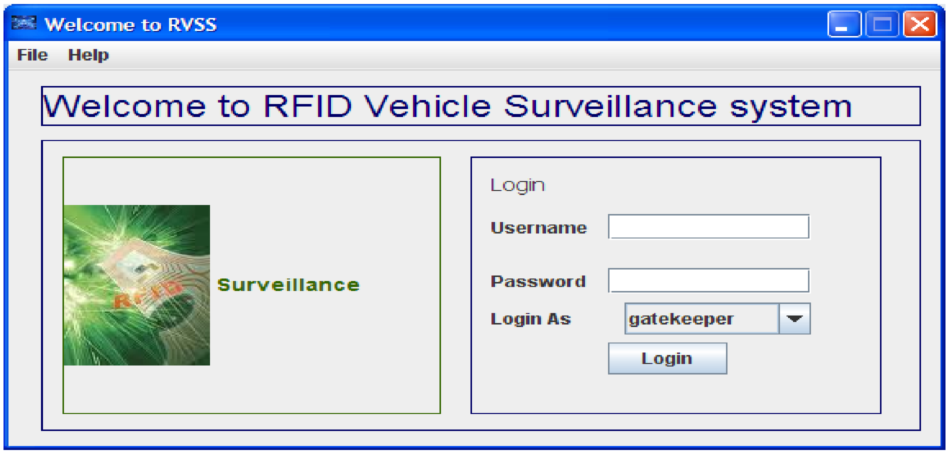
The client back end is used to connect to the central server. It takes input from its user and queries the data base by using the functions of the RMI. The back end is used to retrieve the result of the login request if the user has a valid username password and access then he is allowed to login to the system. And establish a connection between client and server.

5.7 Interface

A user friendly graphical user interface is being developed.

5.7.1 Welcome

Welcome is the first screen which appears when the application starts. It allows to login to the system. The login is based on the type of user i.e., Administrator, gatekeeper, Security Officer. On providing correct login and respective password the window according to the user opens. For example when administrator logins the administrator window opens, similarly the others. A little introduction of the system is also displayed in Welcome window.



Welcome to RVSS

Welcome to RFID Vehicle Surveillance system

Login As: gatekeeper

Figure 5.4 Welcome screen

5.7.2 Administrator

When a user login as an administrator, Administrator window is displayed.

Options available in that window are,

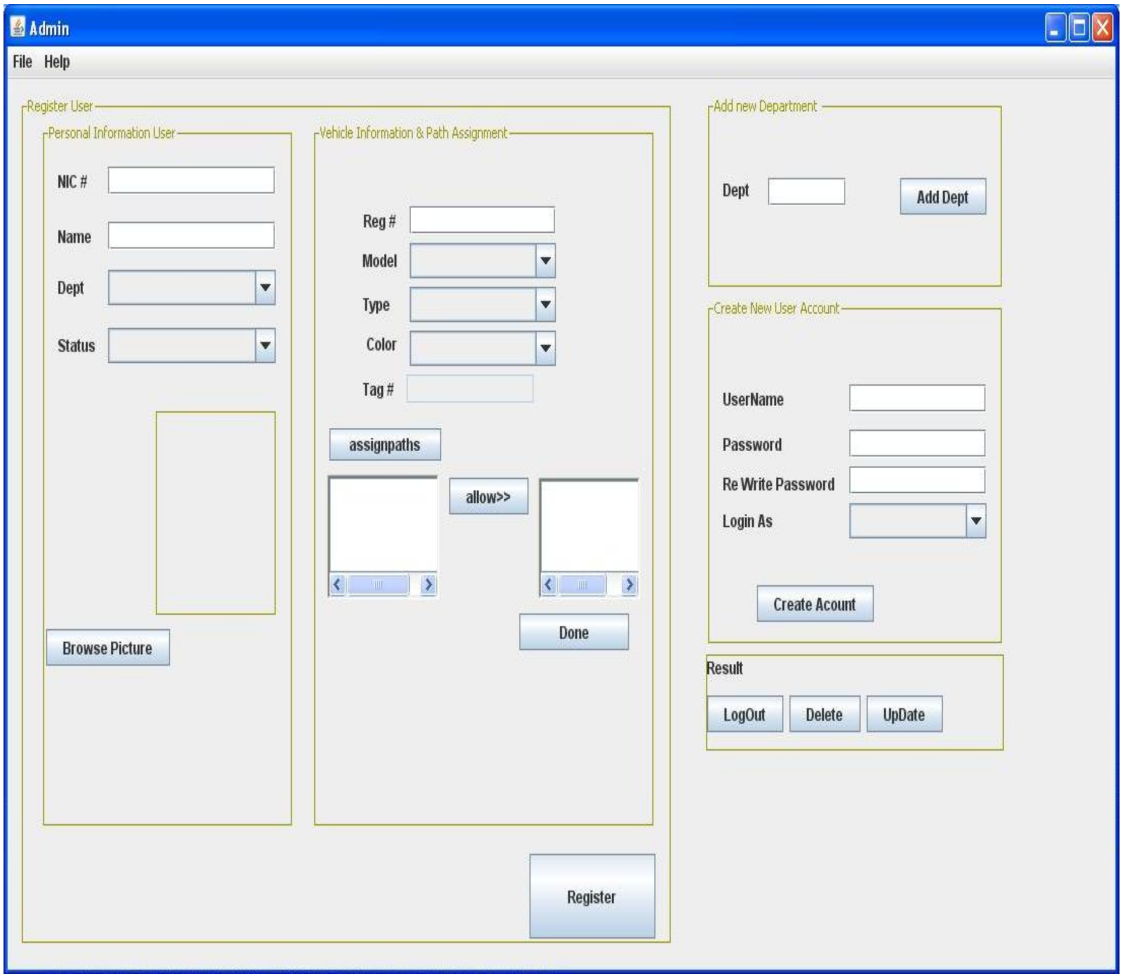


Figure 5.5 Administrator screen

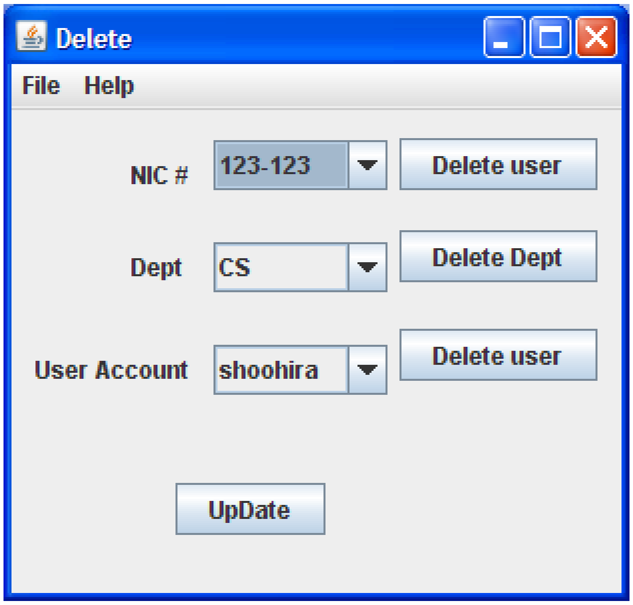
Add a new registered user: NIC number, name, picture, department, status, Vehicle registration number, type, model, color, Assigned tag are taken as input and then Add register user add this information in the database.

Add a new Department: Name of Department is taken as an input and Add Dept button add this information in the database.

Add a new Type of Vehicle: Type of Vehicle is taken as an input, and Add type Button Add this information in the database.

Add a new User account: Username, Password and login as is taken as an input and Create account button adds this information in the database.

Delete: When user press delete button following options are shown.



NIC # 123-123

Dept CS

User Account shoohira

Figure 5.6 Delete screen

Delete a user account: Select user name and then press delete user account button. User account will be deleted and on pressing update button available user account will be shown.

Delete a registered user: Select NIC of registered user and then press delete user button. Registered user will be deleted and on pressing update button available registered user NIC will be shown.

Delete a Department: Select Department name and then press delete Department button. Department name will be deleted and on pressing update button available department name will be shown.

5.7.3 Security Officer

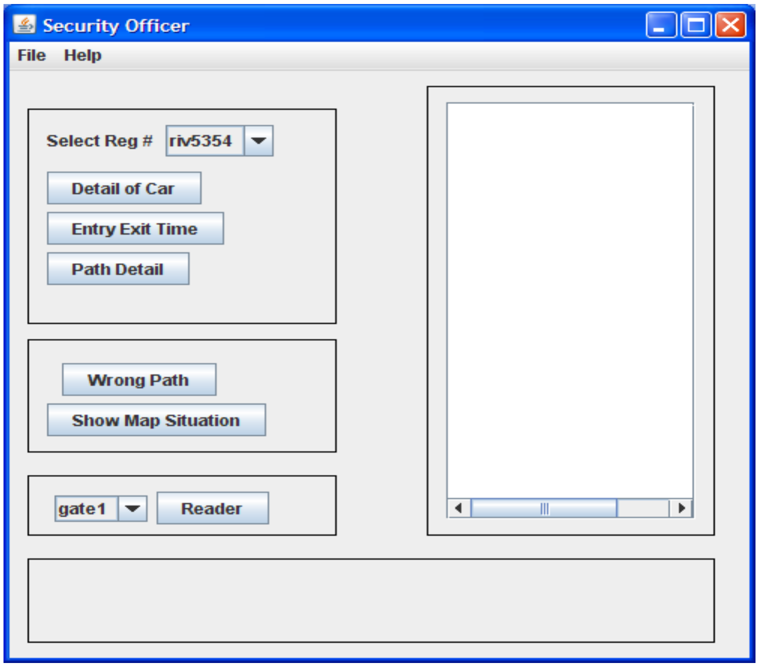


Figure 5.6 Security Officer Screen

Security Officer has following options.

Detail of car: Detail of the selected vehicle is shown. Color, model, type owner and status as visitor or registered is shown.

Entry exit time: Entry exit time of the selected car is shown.

Path detail: Detail path is shown. Track of the selected vehicle as from which readers car has passed is shown.

Wrong Path: Detail of wrong path that a particular vehicle has taken is shown.

Reader situation: Detail information about a reader is shown. As at any time which cars are within selected reader range are present.

Show map situation: Map situation button shows the current situation of readers. It shows how many vehicles in each reader range are present.

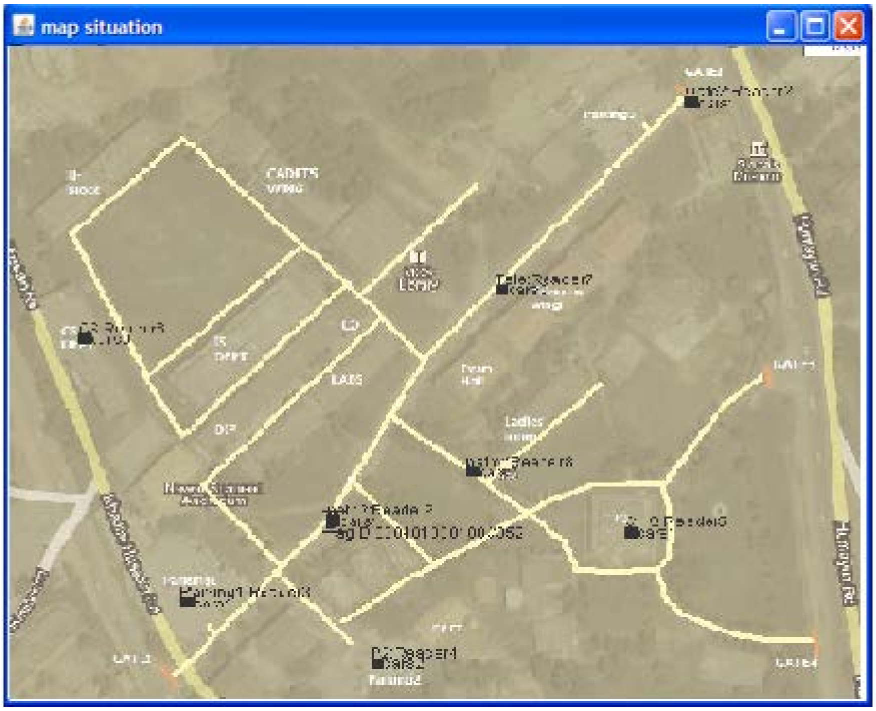


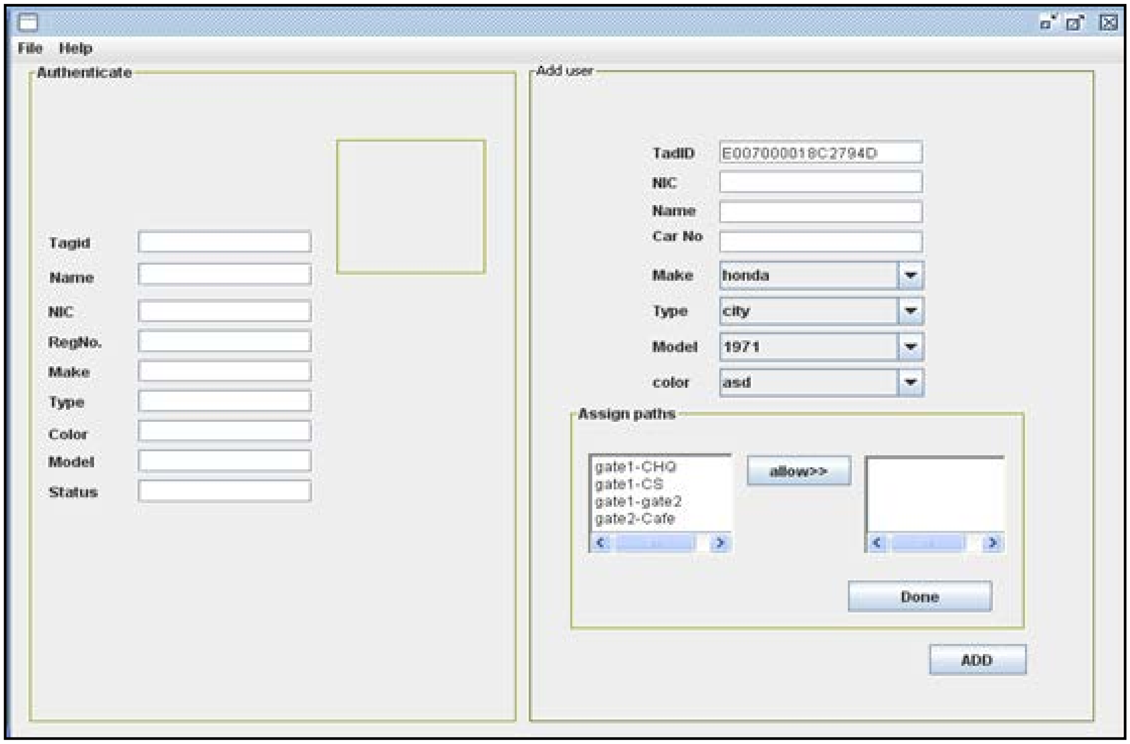
Figure 5.6 Data Display for active monitoring

5.7.4 Gatekeeper

This is the application module of the system installed for gatekeeper. It has an RFID reader attached to it for reading tag entering into the organization where this system is installed. It works to authenticate the registered users (who have a tags assigned to them) and assigns tag to visitors i.e. unregistered users.

5.7.4.1 Front End Development

The user interface is designed and developed to be user friendly so that it is easy to operate keeping in view the nominal computer literacy level of the expected user. A GUI is developed that provides allows gatekeeper to view the data of registered cars and vehicle owners so that they can be authenticated against their saved data in the database .It also has the option to add a new car by adding required data to the fields. Figure 6.7 shows the front end of the gatekeeper client application.



TadID E007000018C2794D

Make honda

Type city

Model 1971

color asd

gate1-CHQ
gate1-CS
gate1-gate2
gate2-Cafe

Figure 5.7: Gatekeeper Interface

5.7.4.2 Back End Development

The client back end is used to connect to the server. It takes input from its user and updates the data base by using the functions of the Server. It sends the tag data read to the server to checks it record in the database .if it's already stored server send all the data saved across that tag for authentication. If that tag is not already added to the system it allows the gatekeeper to enter data against the tag to add it to the system. The back end is used to retrieve the result of operations performed by the server on the request of the client The RFID reader was installed using its driver and configured for the system. The implementation was carried out using serial comm. ports for which the external package javax.comm was used

6 Simulation

6.1 Introduction

Simulation is the imitation of some real thing, state of affairs, or process. The act of simulating something generally entails representing certain key characteristics or behaviors of a system. FID Based Automated Vehicle monitoring and surveillance system is based on the setup of warless LAN through which all the wireless RFID readers are connected to the server and send updates to the server for any data read . this requires series of long range RFID readers to be installed along strategic positions that can grapple on any path violations to allow for active monitoring. Due to limited resources large number of RFID readers could not be made available. The hardware available was a single RFID reader with very small range. Therefore a separate module of simulation has been developed to show the basic features of how the system will work.

6.2 Implementation

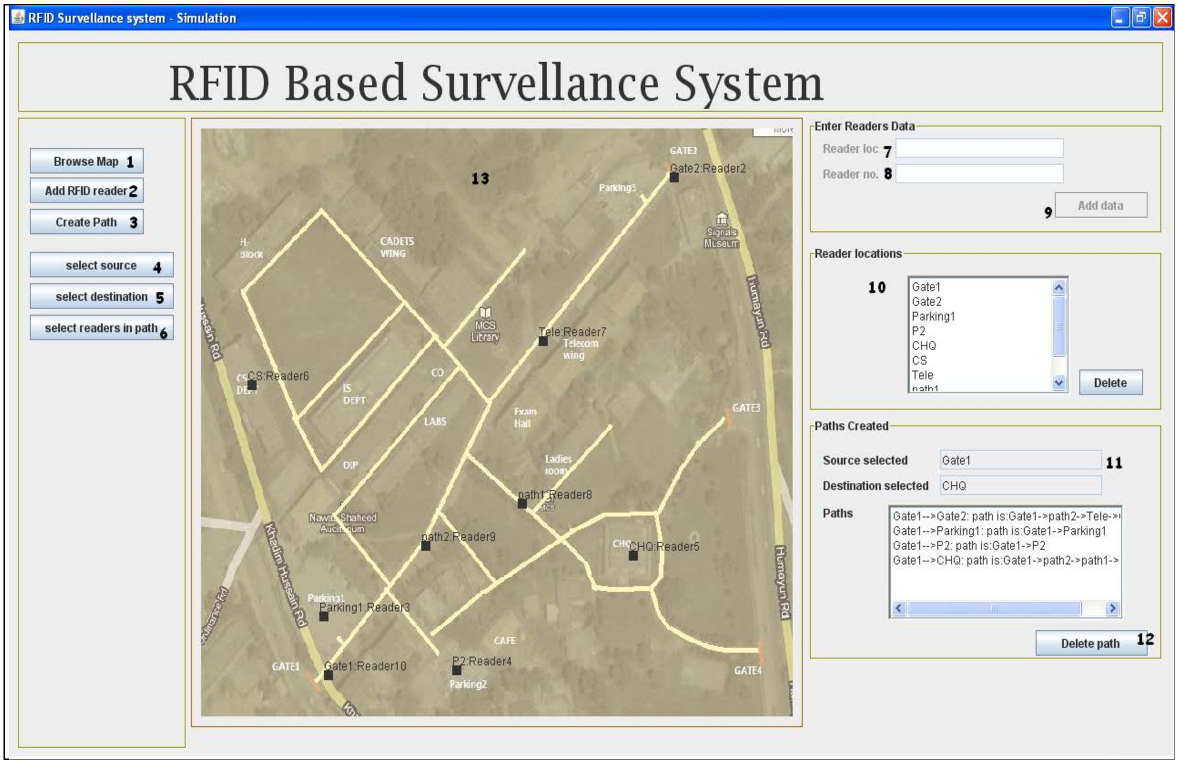
Simulation has been implemented using java with net beans IDE. Client. The RFID reader was installed using its driver and configured for the system. The implementation was carried out using serial comm. ports for which the external package javax.comm was used. Simulation has been implemented in two parts first part is for READER INSTALLATION AND PATH CREATION and second part is the actual SYSTEM SIMULATION.

6.3 Reader Installation and Path Creation

This module has been developed to allow the administration of the organization to install the readers at specific locations on the map through clicking on those locations .This basically adds organizations information (location and paths) to system to perform its key functions as active monitoring and authentication

6.3.1 Front end

Front end has been developed as a user friendly interface to add the readers at point clicks, deleting readers, creating paths on point clicks and deleting them. As a path is created or reader is installed it is displayed to provide feedback so user can verify for its correction.



Gate1
Gate2
Parking1
P2
CHQ
CS
Tele
path1

Source selected: Gate1
Destination selected: CHQ

```
Gate1-->Gate2: path is:Gate1->path2->Tele->
Gate1-->Parking1: path is:Gate1->Parking1
Gate1-->P2: path is:Gate1->P2
Gate1-->CHQ: path is:Gate1->path2->path1->
```

Figure 6.1 Adding readers and creating paths

Browse map (1) allows to add any map to the working area (13) on clicking ADD RFID READER (2) working area is activated ,as a mouse is clicked at any point a reader is plotted and READER LOC (7) and READER NO(8) fields are activated to allow enter data to these fields. After ADD DATA (9) is clicked it's added to the sends the update to server and then the reader is displayed in READER LOCATIONS (10). After placing all the readers, for path creation CREATE PATH (3) button is to be clicked that enables the buttons SELECT SOURCE, SELECT DESTINATION, AND SELECT READERS IN PATH. On clicking the select source button and clicking on the reader (that is to be taken as a source) this information is displayed in SOURCE SELECTED field (11).After selecting source and destination SELECT READERS IN PATH (6) is to be clicked .Clicking on readers the makes the path creates a path stores it to database and displays it in PATHS in box (11).

6.3.2 Back end

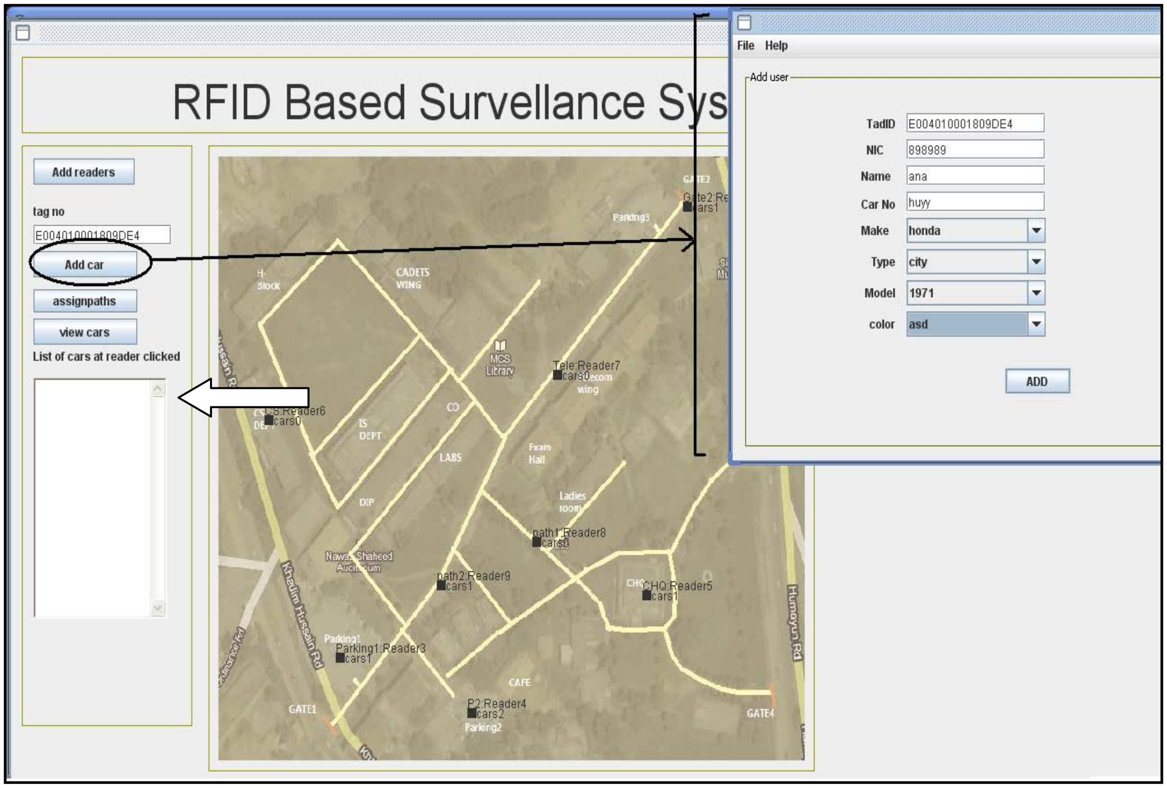
This module adds the basic data for the system required for its working because the paths are the basic information needed to provide active monitoring. It connects to database at its back end. All the readers location and their data added is updates in the database and so for the paths created .it checks that each reader is assigned a name or not.

6.4 System simulation

This portion of simulation imitates what actual system will do and is connected to a reader .As only one reader is available so it is programmed to act as multiple readers. In addition to working with reader it also works with simulated data where it creates threads for each tag entering in the organization .It generates alert for all path violations which in actual system is sent to the security officer

6.4.1 Front end

It’s an interface that displays a general view of what all is happening in the system .It shows all the readers’ number of cars at one reader it displays the registration number of the car read at that reader. It also has an interface to connect to the reader and make it act as any of the reader plotted at any location. It also allows for adding a user to the system and assigning paths. For any path violation either from data coming for the connected reader or simulates data it displays and alert showing details of the car that violated the path .On clicking the reader a list of cars near that reader is displayed.

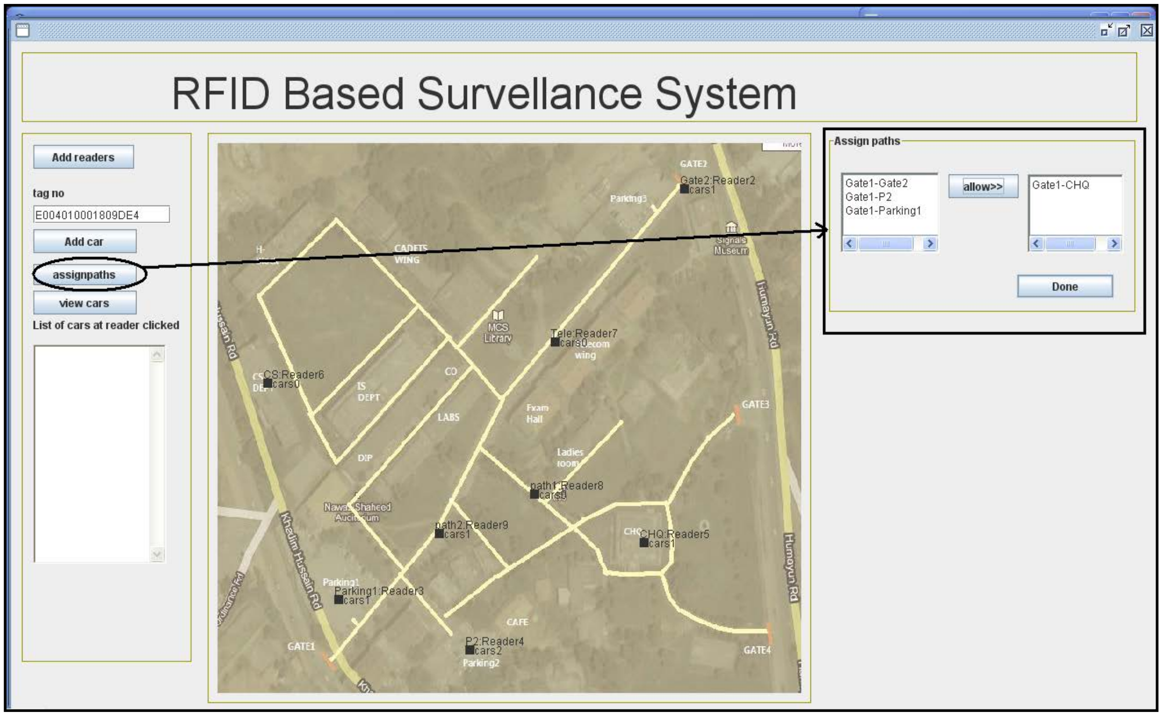


tag no E004010001809DE4

TadID E004010001809DE4
NIC 898989
Name ana
Car No huyy
Make honda
Type city
Model 1971
color asd

Figure 6.2 Adding a new car

ADD CAR button opens up a window to add a new car to the and entering it information its is connected to reader and takes tagged from as value read from reader .On clicking on a reader on the map the list of cars registration number is displayed in box market with arrow.


RFID Based Survellance System

tag no: E004010001809DE4

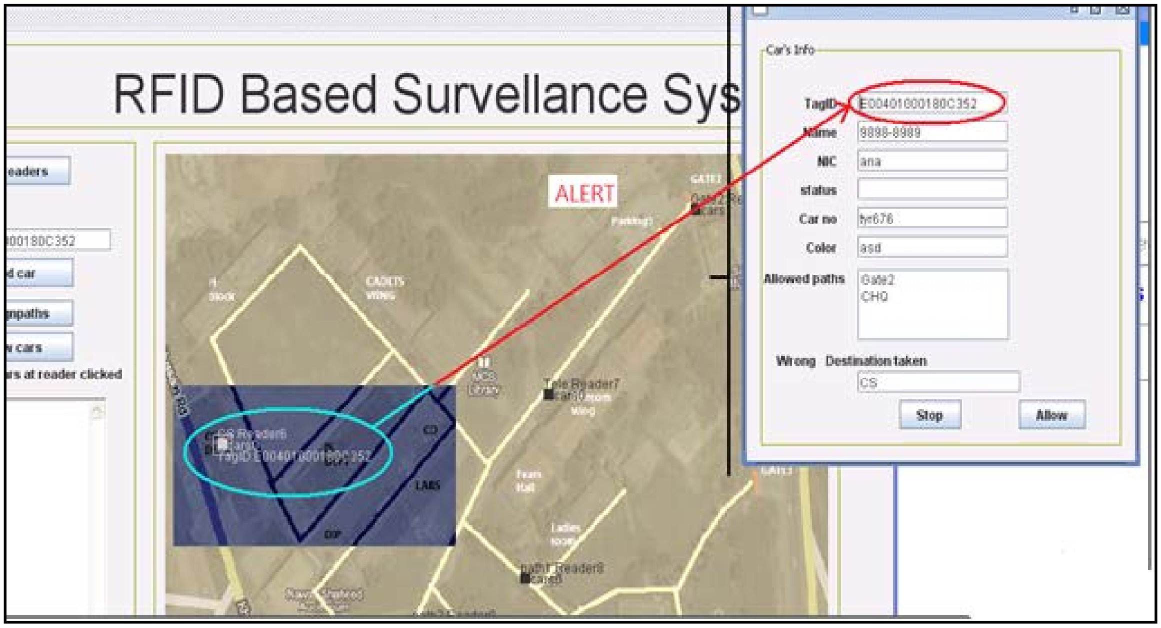
Assign paths

Gate1-Gate2
Gate1-P2
Gate1-Parking1

Gate1-CHQ

Figure 6.3 Assign Paths

After adding the user a path is assigend to it . here is a list of all the path available selecting the paths and and doing allow opertaion adds it to the lis of allowed paths and clicking on DONE assocites these paths with this tag id . Ny variation from this path reports an alert. As shown in figure

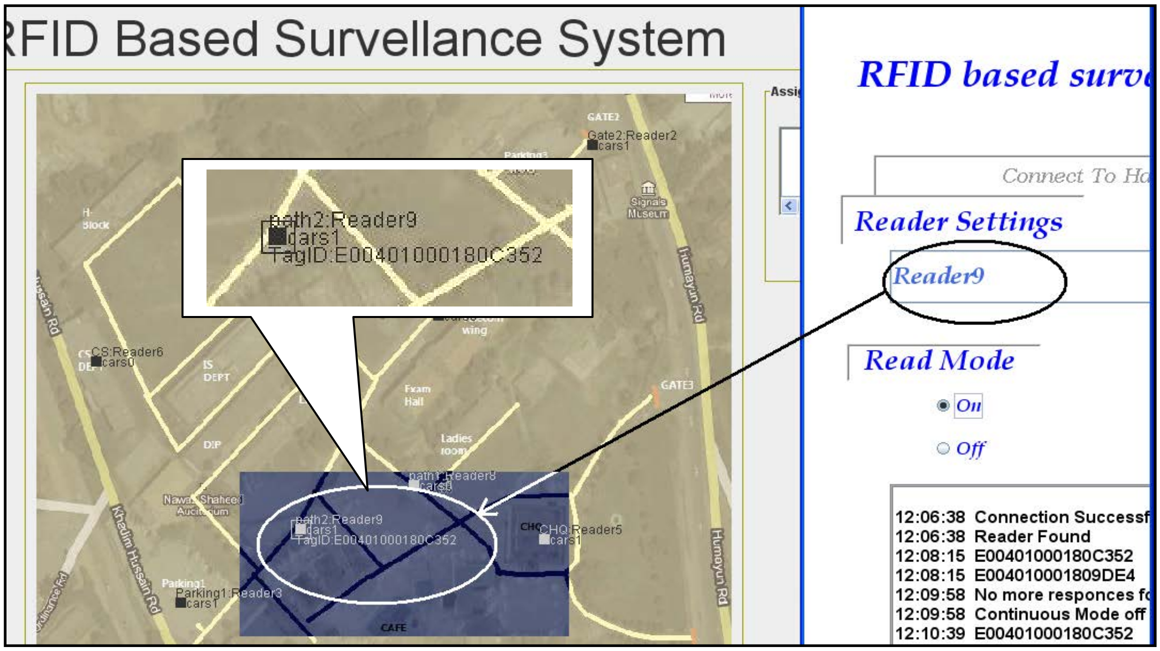


```
00180C352
```

Car's Info

TagID: E00401000180C352
Name: 9898-8989
NIC: ana
status:
Car no: hr676
Color: asd
Allowed paths: Gate2
CHQ
Wrong Destination taken: CS

Figure 6.4 Alert generation



Read Mode: ● On ○ Off

```
12:06:38  Connection Successf
12:06:38  Reader Found
12:08:15  E00401000180C352
12:08:15  E004010001809DE4
12:09:58  No more responces fo
12:09:58  Continuous Mode off
12:10:39  E00401000180C352
```

Figure 6.5 Tag ID display and counter updates

The reader is connected to reader 9 the location is named as path2 .as the tagged is read at this reader its tagid is displayed. Total number of cars at that reader is also updated.

6.4.2 Backend

At back end this module connects to the reader and database. It connects to the reader for the reader number with which it connects it checks from the database at which reader location it has to connect to and then displays the data read at from the reader and updates the counter at that reader. For simulated data it initializes a thread for each car entering the premises of organization and displays alerts on screen from this data

7 Testing

7.1 Introduction

Software testing is one element of a broader topic that is often referred to 'Verification and Validation' (V&V). Verification refers to the set of activities that ensure that software correctly implements a specific function. Validation refers to the different set of activities that ensures that the software that has been built is traceable to customer requirements. Figure 7.1 explains the testing process.

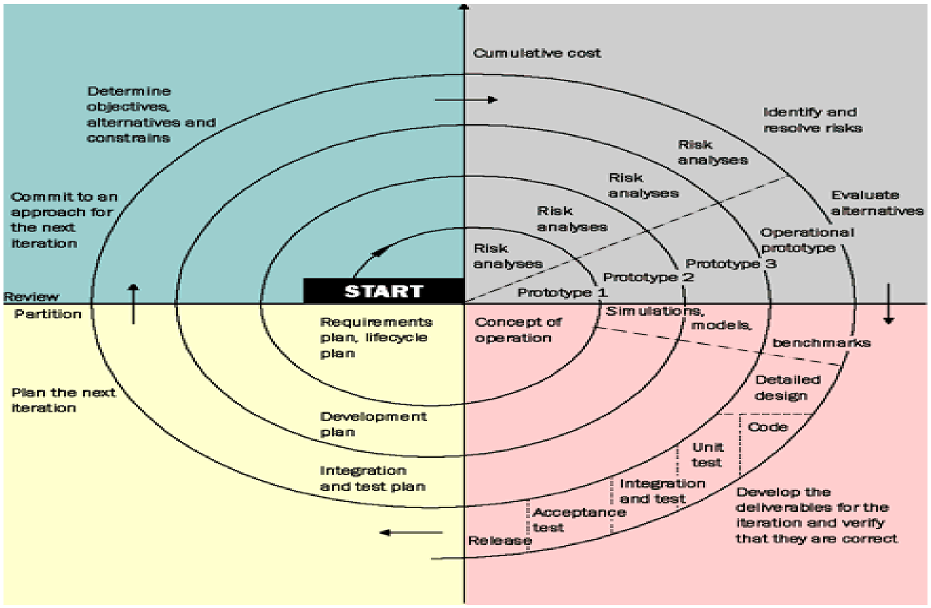


Figure 7.1: Software Testing Process

7.2 Validation and Verification

Validation and Verification is intended to be a systematic and technical evaluation of the system and its processes. To effectively deal with the increased complexity and functionality, systems need practical techniques that can help improve software quality using the validation and verification process. RFID Based Automated Vehicle Surveillance system was tested for validation by giving it different inputs and getting the desired outputs. In verification testing it was assured that software meets all functional, behavioral, and performance requirements.

7.3 Unit Testing

In computer programming, unit testing is a procedure used to validate that individual units of source code are working properly. A unit is the smallest testable part of an application. Unit testing concentrates on each unit of the software as implemented in source code. The goal of unit testing is to isolate each part of the program and show that the individual parts are functioning properly. In our system each component as developed is individually tested so as to check for possible errors that could occur.

7.4 Integration Testing

Integration testing is the phase of software testing in which individual software modules are combined and tested as a group. It follows unit testing and precedes system testing. In integration testing focus is on design and the construction of software architecture. We have structured the classes as per the user requirement so the extensibility of the software is guaranteed. The system is constructed and tested ensuring conformity with the basic objectives of the software testing strategy. The design is fully based on user specification.

7.5 Black Box Testing

Black box testing takes an external perspective of the test object. These tests can be functional or non-functional, though usually functional. The test designer selects valid and invalid input and determines the correct output. There is no knowledge of the test object's internal structure. The requirements established as part of software requirements analysis, are validated against the software that has been constructed. System has been fully validated as per meeting user requirement. It provides final assurance that the software meets all functional, behavioural, and performance requirements. Black box testing techniques have been used exclusively during validation.

7.6 Yellow Box Testing

Yellow box testing is checking against the warning messages and alerts. It helps answer weather the system is properly throwing the warning messages and alerts or not? In the developed system, system shows a warning in case

1. Invalid username or password at the time to log in.
2. Field left empty at the time of registering a user.
3. A vehicle violates a path (this is the result of the active monitoring; active monitoring is one of the system features for reference see Appendix Section B)

7.7 Red Box Testing

Red Box Testing is the user acceptance testing and is defined as the protocol testing in case of any protocol based systems. In the developed system the RFID communication protocol is tested against all its required functions and it was concluded that the protocol well performs all the required functions.

7.8 System Testing

In software testing the software and other system elements are tested as a whole. System testing of software or hardware is testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing takes, as its input, all of the integrated software components that have successfully passed integration testing and also the software system itself integrated with any applicable hardware system(s). The purpose of integration testing is to detect any inconsistencies between the software units that are integrated together. System testing is a more limiting type of testing; it seeks to detect defects within the system as a whole. For the developed system, it has been verified that all elements connect together properly and that the overall system function/performance is achieved.

8 Appendix

SECTION A

Relational Database Table Design Standards and Principles

Database Developers Network Neal Bratschun, 8 February 2003

ddn@ddn.org.af / ddn@neepjpmu.org nwb@strategicimpactconsulting.com

Table Naming Standards

1. The table name must clearly identify the one type of data the table contains.
2. Table names should usually be plural.
3. Table names are the only database objects not prefixed.
4. Only the first latter of each word in a table name should be capitalized.
5. Functional table names (for many-to-many relationships) should be a combination of
6. the two tables referenced. For example, SubprojectsDocuments indexes the many-tomany
7. relationship between Subprojects and Documents.
8. Lookup tables should be grouped together. They can be grouped at the end by
9. starting the name with the letter "z". Lookup table names may be singular.
10. Developer's tables begin with "zz". (Example: zzChangeLog)
11. FIELD STANDARDS
12. Field names should be descriptive and clearly understandable by someone who looks
13. at them for the first time.
14. Field names should contain letter, number, and underscore characters ("_") only.

15. The primary key field should always end with “ID”. If the field is not autonumber,
16. all relationships to that field must have referential intergrity with cascade update.
17. A field referencing a primary key field in another table should have exactly the same
18. name as the field it is referencing.
19. If multiple fields in a table reference the same primary key in another table, the field
20. names should be followed by a suffix. (Example: BankID_from BankID_to)
21. Abbreviations for field names should be avoided. An exception is when the table
22. name contains or more words, the initials of each word may be used, for any field
23. except the Primary Key field, and the description field. For example, if the table
24. name is “FacilitatingPartners”, a field may be named “FPName”. The Primary Key
25. would be “FaciliatingPartnerID”.
26. Fields with Dari text have a name ending with “_dari”
27. Composite keys should never be used. If necessary, use a composite index instead.
28. Field names should be unique throughout the database.
29. NORMALIZATION PRINCIPLES
30. A field is a fact about a particular subject (the field contain one piece of data).
31. Fields in a database should be fully functionally dependent on the primary key.
32. All non-key fields should be mutually independent. For example, a calculated field
33. should not be stored. (Note: sometimes this rule must be violated for performance
34. reasons.)

35. JOIN PROPERTIES
36. Inner Join – contains matching records from both tables
37. Left Join – contains all the records from the left table, and matching from the right
38. Right Join – contains all the records from the right table, and matching from the left

Join Types

1. One-to-one
2. One-to-many
3. Many-to-many – requires an intermediate table

SECTION B

Software Requirement Specification Document (SRS)

Introduction

Purpose

Managing a busy parking lot can pose significant challenges, especially to a sensitive organization that also owns some of the vehicles in the premises. The area has to be secure, with barrier-enforced entrances and exits. It should be an automated efficient monitoring system that provides accurate authentication, active monitoring and vehicle tracking. Providing different queries regarding vehicle record, validation of the registered user and easy in-and-out access for drivers.

Document Conventions

1. When writing this document it was inherited that all requirements have the same priority.
2. First there is presented an overall view about RFID Based Vehicle Surveillance System and then all features and functions are analyzed in detail.

Intended Audience and Reading Suggestions

This requirement document contains general information about RFID Based Vehicle Surveillance System, main classes and use cases, functions, features and special technologies. It describes in detail all that RFID Based Vehicle Surveillance System needs to work properly and with safety.

Developers: in order to be sure they are developing the right project that fulfills requirements provided in this document.

Testers: in order to have an exact list of the features and functions that have to respond according to requirements and provided diagrams.

Users: in order to get familiar with the idea of the project and suggest other features that would make it even more functional.

Documentation writers: to know what features and in what way they have to explain. What security technologies are required, how the system will response in each user's action etc.

System administrators, security officer and gatekeeper: in order to know exactly what they have to expect from the system, right inputs and outputs and response in error situations.

Project Scope

In this project we will be developing interface of RFID reader and computer system. Designing of the RFID reader installation at different places within the MCS. Designing and implementing of the Database with a large number of transactions. Designing different queries and graphical display for tracking record.

Overall Description

At entrance of organization RFID readers read data from the RFID tag that will be installed on the vehicle and will sends the data to the server via LAN.

1. If the data matches with any of the entry in database vehicle is authenticated. Vehicle will be allowed to enter the organization.
2. In case of visitors, gatekeeper will issue a tag to that vehicle and will enter relevant information about the visitor in database. Based on purpose of visit, an allowed path will be assigned to the vehicle. As vehicle enters in the organization RFID reader will track the vehicle and send information to Central Server. Any deviation from assigned path will result in a generation of alert signal to security officer.
3. As vehicle passes through different locations RFID reader (installed to cover all possible paths) reads the tag and send information to central server. Server updates tracking record of vehicle in the database.
4. The architecture used for this system is client server Architecture. Client will send queries to server .e.g. Client at entrance will send query for authentication or new entry. Security officer (client) can query for viewing any record.

Product Perspective

The software will be based on the RFID information that we will get from different locations. RFID readers will be installed in such a way that it can cover all possible paths for vehicle movement. The information from the RFID readers will be transmitted to database through wireless LAN. For the limitation of time and cost RFID readers will not be installed physically. These will be simulated and will work for any scenario of vehicle movement. System will allow user to make different scenarios in order to test the system. All other modules will be designed such that it can be easily integrated with actual network of RFID readers.

Product Features

Our objective is to provide automatic authentication to the vehicle.

1. Registering the authorized users
2. Assigning and monitoring the visitor tags
3. Maintaining record of vehicle movement and providing
4. Information of record to the security officer through user friendly interface.

User Classes and Characteristics

Administrator: Administrator will maintain and manage the database on central server. Administrator has full access to the database and can request any query. Administrator can register users, delete users, assigns login and password, access rights, monitors tracking record, add paths, change paths etc.

Security officer: Security officer can only query record required for keeping security checks such as taking a record of cars in a particular parking. Or tracking the movement of a car.

Gatekeeper: Gatekeeper checks whether that the data read from reader matches the data entered for the particular RFID tag. If the data matches only then client will be allowed to enter.

RFID reader: RFID readers send signals through wireless connection. As tag crosses the interrogation zone of RFID reader it sends an update to central server that updates the record of tag.

Operating Environment

The operating system used for this Software is Windows XP. Hardware includes RFID Reader that connects to the system through USB Port.

Design and Implementation Constraints

As the available RFID reader are limited and of limited range so we cannot deploy this system in real time but this software will be capable of running in the real environment.

User Documentation

1. User manual
2. Project report

Assumptions and Dependencies

RFID readers are enough to provide full coverage of the area under consideration.

Server is up and running.

RFID readers are working and transmitting data properly.

All clients are connected to the Server.

External Interface Requirements

The RFID Card Reader is available in LF or HF version with its USB connection. By connecting the USB interface for data communication and power supply, it can read and write the tags with computers for information collection and data transfer. It is connected to USB serial port because it is speedy. The data is written in the form of binary form of the ASCII values and is read as ASCII values of the binary data. The hardware is made to responding to tag events and errors, updating databases and other applications, sending processed information to systems through connectors.

The EPC and ISO have standardized the first two layers of the communication protocol stack between the readers and the tags. These two layers include the local wireless communication that occurs between a reader and the tags within its read field. The first layer standard is the physical, which describes the specific radio frequencies and whether tags and readers are communicating in half or full duplex mode. The second layer, referred to as the data link layer, has been standardized based on a slotted Aloha scheme. Middleware standards have been defined to support temporary collection of event data for filtering and consolidating the EPC data coming from the readers. This standard is called Application Level Events or ALE. Another communication standard has been defined for the readers in terms of how they capture and communicate event data from tags and sensors - called the Reader Protocol (RP).

User Interfaces

There are three different users. As they turn on the system a log in interface appears. Users enter login id and password. System checks the database for these entries and provides the appropriate user interface.

For administrator the window has created user, register user delete user, tag entry, query time, query path and generate path option. For security officer there will be query time query path and generate path option and At gatekeepers interface as a car enters information from the server is sent to gatekeeper for the tag read info is displayed along with the picture. Gatekeeper authenticates and allows the car to enter. Gatekeepers interface has an additional option for visitor's entry. For this option a new form window opens that takes the information of the car owner has an assign path option and a tag id is assigned to it.

Hardware and Software Interfaces

The RFID Card Reader is available in LF or HF version with its USB connection. By connecting the USB interface for data communication and power supply, it can read and write the tags with computers for information collection and data transfer. It is connected to USB serial port because it is speedy. The data is written in the form of binary form of the ASCII values and is read as ASCII values of the binary data. The hardware is made to responding to tag events and errors, updating databases and other applications, sending processed information to systems through connectors.

The EPC and ISO have standardized the first two layers of the communication protocol stack between the readers and the tags. These two layers include the local wireless communication that occurs between a reader and the tags within its read field. The first layer standard is the physical, which describes the specific radio frequencies and whether tags and readers are communicating in half or full duplex mode. The second layer, referred to as the data link layer, has been standardized based on a slotted Aloha scheme. Middleware standards have been defined to support temporary collection of event data for filtering and consolidating the EPC data coming from the readers. This standard is called Application Level Events or ALE. Another communication standard has been defined for the readers in terms of how they capture and communicate event data from tags and sensors - called the Reader Protocol(RP).

Communications Interfaces

All the RFID readers, clients and server must be connected to LAN.

Other Nonfunctional Requirements

Software Quality Attributes

The system is subjected to following quality attributes.

1. Reliability: The system should provide backup and should have failure management.
2. Availability: The system should at least be available within the office hours
3. Extendibility: The system should be able to incorporate extensions i.e. increase in vehicle tag, parking lots, new paths.

Response time: The system should have a response time of few seconds.

9 Bibliography

1. A Basic Introduction to RFID Technology And Its Use In The Supply Chain by R. K. LARAN, January 2004.
2. RFID A Week Long Survey On The Technology And Its Potential, Harnessing Technology Project, Interaction Design Institute Ivera by Mario Chiesa, Ryan Genz, Kim Mingo and Jason Tester.
3. TAGS, YOU'RE IT by Bushell and Sue, Australia, March, 2004.
4. Making sense of RFID by Laura, Y. Smart, October 2004.
5. Introduction to RFID Technology Published by IEEE CS and IEEE ComSoc
6. Radio Frequency Identification Technology by IEE 2005, July 2005.
7. http://www.gis.com
8. http://www.arcdeveloper.net
9. 'Mastering ArcGIS' by Maribeth Price South Dakota School of Mines and Technology.
10. 'Geographical Information System Based Tracking and Plotting (GTrap)' by BESE-9
11. http://java.sun.com/docs/books/tutorial/rmi/overview.html
12. Mastering RMI: Developing Enterprise Applications In Java and EJB by Rickard Oberg
13. Distributed Computing Principles And Applications by M.L. Liu California Polytechnic State University, San Luis Obispo.
14. http://www-306.ibm.com/software/rational/uml/
15. http://atlas.kennesaw.edu/~dbraun/csis4650/A&D/UML_tutorial/what_ is_ uml.htm
16. http://www-306.ibm.com/software/awdtools/developer/rose/index.html
17. Java Tutorials, http://java.sun.com
18. Tag-it HF-I Plus Transponder Inlays, Refernce Guide, Literature Number:SCBU004 December 2005, Texas Instruments