

Abstract

Industrialisation has proved to be one of the biggest factor for globalisation all over the world. The industries are growing at a faster rate, globalisation has surely increased. Due to globalisation, these industries are achieving business growth in no matter of time. The sellers are producing goods in one country and selling the same in other country, thus by making huge profits. As huge amounts of cargo is shipped everyday by different sellers, today there is no efficient system existing to track the cargo. Thus to help the seller, we are proposing a Cargo Tracing and Business Analysis System. The system basically uses RFID tags and a database. We propose method in which there will be two main checkpoints having both check-in and check-out processes. The RFID tags are then attached to cargo container. It can be placed in a some type of casing so that it does not get damaged. The intermediate checkpoints could vary depending on the type of cargo and number of halts of the ship. The RFID scanners at the check-out areas facilitate step by step tracing and monitoring of cargo. With this tracing system, the seller can also analyze the business growth on various parameters. We also propose the growth analysis of seller using data mining algorithms to help the seller in shipping the cargo based on the demand as well as the profit earned from them.

Keywords- RFID (Radio Frequency Identification) Tag, RFID Scanner/Reader, Arduino Board, Esp-WIFI Module.

Contents

1	INTRODUCTION	1
1.1	Problem Definition	1
1.2	Motivation	1
1.3	Existing System	1
1.4	Proposed System	2
1.5	Scope of the Project	2
1.6	Salient Contribution	3
1.7	Organization of Synopsis	3
2	LITERATURE OVERVIEW	4
3	SOFTWARE PROJECT MANAGEMENT PLAN	5
3.1	Introduction	5
3.1.1	Project Overview	5
3.1.2	Project Deliverables	6
3.2	Project Organisation	7
3.2.1	Software Process Model	7
3.2.2	Roles and Responsibilities	8
3.2.3	Tools and Techniques	8
3.3	Project Management Plan	9
3.3.1	Tasks	9
3.3.1.1	Task 1 - Requirement analysis	9
3.3.1.2	Task 2 - Software requirement specification	9
3.3.1.3	Task 3 - Software design document	10
3.3.1.4	Task 4 - System Test Document	10
3.3.1.5	Task 5 - Coding and Hardware Integration	11
3.3.2	Risk Table	12
3.3.3	Risk Template-1	13
3.3.4	Risk Template-2	13
3.3.5	Risk Template-3	14
3.3.6	Risk Template-4	15
3.4	TimeTable	16

4	SOFTWARE REQUIREMENTS SPECIFICATION	18
4.1	Introduction	18
4.1.1	Purpose	18
4.1.2	Project Scope	18
4.2	Specific Requirements	19
4.2.1	External Interface Requirements	19
4.2.1.1	User Interface	19
4.2.2	Hardware Interface	19
4.2.3	Software Interface	20
4.2.4	Communication Protocols	20
4.2.5	Software Product Features	21
4.2.5.1	Login	21
4.2.5.2	Register	21
4.2.5.3	Filter	21
4.2.5.4	View Timeline	21
4.2.5.5	Scanning	21
4.2.5.6	Erasing	22
4.2.6	Software System Attributes	22
4.2.6.1	Reliability	22
4.2.6.2	Availability	22
4.2.6.3	Security	22
4.2.6.4	Maintainability	23
4.2.6.5	Portability	23
4.2.6.6	Performance	23
4.2.7	Database Requirements	23
5	SOFTWARE DESIGN DOCUMENT	24
5.1	Introduction	24
5.1.1	Design Overview	24
5.1.2	Design Requirement Traceability Matrix	24
5.2	System Architecture	25
5.2.1	Chosen Architecture : Event Driven Architecture and Client Server Architecture	25
5.3	System Interface Description	26
5.3.1	Hardware Interface	26
5.3.2	Software Interface	27
5.4	Detailed Description Of Components	27
5.4.1	Server Connection	27
5.4.2	Verification	27
5.4.3	RFID tags	28
5.4.4	RFID Reader	28
5.4.5	Wifi Module	29

5.4.6	Arduino	29
5.5	User Interface	29
5.5.1	Description	29
5.5.2	Images	30
6	SOFTWARE TEST DOCUMENT	33
6.1	Introduction	33
6.1.1	System Overview	33
6.1.2	Test Approach	33
6.1.2.1	Unit Testing	33
6.1.2.2	System Testing	33
6.1.2.3	Performance Testing	34
6.1.2.4	Functional Testing	34
6.2	Test Plan	34
6.2.1	Features to be tested	34
6.2.2	Features not to be tested	34
6.2.3	Testing Tools and Environment	35
6.3	Test Cases	35
6.3.1	Test Case: Registration	35
6.3.1.1	Purpose	35
6.3.1.2	Inputs	35
6.3.1.3	Expected Output and Pass/Fail Criteria	35
6.3.1.4	Test Procedure	35
6.3.2	Test Case: Login	36
6.3.2.1	Purpose	36
6.3.2.2	Inputs	36
6.3.2.3	Expected Output and Pass/Fail Criteria	36
6.3.2.4	Test Procedure	36
6.3.3	Test Case: Shipment Booking	36
6.3.3.1	Purpose	36
6.3.3.2	Inputs	36
6.3.3.3	Expected Output and Pass/Fail Criteria	36
6.3.3.4	Test Procedure	37
6.3.4	Test Case: Tracing	37
6.3.4.1	Purpose	37
6.3.4.2	Inputs	37
6.3.4.3	Expected Output and Pass/Fail Criteria	37
6.3.4.4	Test Procedure	37
6.3.5	Test Case: RFID Scanning	38
6.3.5.1	Purpose	38
6.3.5.2	Inputs	38
6.3.5.3	Expected Output and Pass/Fail Criteria	38

6.3.5.4	Test Procedure	38
6.3.6	Test Case: RFID Erasing	38
6.3.6.1	Purpose	38
6.3.6.2	Inputs	39
6.3.6.3	Expected Output and Pass/Fail Criteria	39
6.3.6.4	Test Procedure	39
6.3.7	Test Case: Business Analyzer	39
6.3.7.1	Purpose	39
6.3.7.2	Inputs	39
6.3.7.3	Expected Output and Pass/Fail Criteria	39
6.3.7.4	Test Procedure	40
6.3.8	Test Case Table	41
7	CONCLUSION	45
8	REFERENCES	46

Chapter 1 INTRODUCTION

1.1 Problem Definition

Due to the growing trade , industrialization and globalization, large scale shipping of cargo is done , owing to increased inter country trade. As the amount of cargo, number of sellers is increasing day by day, there is also a need to update the existing system which has many discrepancies. So there is definitely a need for a System to track these shipments, prevent their loss as well as provide an automated analysis to the seller which may help them to improve their business and enhance their profits manifold.

1.2 Motivation

As we see nowadays, we have a lot of items being imported into different countries. Whether be it clothes for fashion or be it the latest technology devices, all of them are imported. With this, the cargo is being shipped everyday to different countries. The ports are busy places where a lot of different kinds of cargo loading and unloading takes place with the help of modern engineering devices. But it maybe possible that some cargo may be mishandled due to human errors. So this gave us motivation to make a system where there is efficient tracing of the cargo so that the shipments reach in time where they are intended to. With more research we came up with the idea to create a system where the seller would be benefited using extra provisions of getting business insights from our system.

1.3 Existing System

As such there isn't a well administered, automated cargo tracing system in the world right now. Currently we are using Barcodes and Barcode scanners. The major drawbacks of these are the usage of tremendous quantities of a treasured resource "paper". Along with this, specialized staff is required as the barcode is not scanned without a certain proximity. Also damaged, folded barcode may or may not be often passed without proper authentication. The existing system does not have the following features:

- It does not notify the user, when there is a breach in the declared route of the cargo.
- Does not notify, even when the barcode is damaged hampering tracking.
- Cannot provide sellers specialized business guidance.

1.4 Proposed System

As we described some of the disadvantages in the existing system stated above, we propose a system wherein the seller could track all the shipments efficiently. The system would be consisting of the website portal wherein the seller need to register along with the details of its own. After registration seller would be authorised to use the system. The cargo shipment will be booked by the seller at the nearest area of our working. The RFID tags will be assigned to them and the load will be distributed using threshold weight for one container and also the cargo type. The tags will be well protected. Now, these tags will be assigned to unique sellers who are using the system. As the cargo passes the initial checkpoint wherein the cargo will be loaded, the system will inform the seller about the same with date and time of the check-in of cargo on the website portal. As the cargo is in transit, it passes through various checkpoints and the same updation will be notified to the user on the portal. If at all there is some mishandling at any point in journey, the scanners located at the checkpoints will help to identify those by giving some error messages. At the last checkpoint, the sellers shipment is finally off-loaded safely and then the RFID tags are removed for their reuse. The system also provides the sellers with some additional functionalities. It also helps the seller to analyze the business growth using the past data and also predict the growth for the next quarters. The system will also give suggestions about the cargo type which the seller sends to particular region. So using this he could successfully analyze the business policies and make profits.

1.5 Scope of the Project

The proposed Cargo Tracing and Business Analysis system will help the seller of the goods to trace the shipments made to different locations. As due to human errors, sometimes the shipments are mishandled and reach at wrong destination. This problem is solved by our proposed system. The project aims to provide services only to those sellers who are selling their goods to other countries. Also by analyzing the business patterns and parameters, the system aids in predicting the overall business growth.

1.6 Salient Contribution

As pointed out that the existing system used for cargo tracing was not efficient we contribute to the existing system by upgrading the system from paper barcodes to new technology of RFID. We also contribute to the existing system by giving the sellers a help in analysing his business and see the growth using our system.

1.7 Organization of Synopsis

Chapter 1- This chapter provides us information of the project in detailed manner. It gives insights about the problem definition, the motivation why we took this as a project, the existing and proposed system with scope and organization of synopsis

Chapter 2- The chapter introduces us about the literature survey performed at the start of the project.

Chapter 3- The chapter tells about the Software Project Management Plan. It covers various topics such as Project Deliverables, the process model used, various tasks that are being carried out along with the Risk Templates, and the timetable of our project.

Chapter 4- The chapter tells about the Software Requirements Specification. It mainly covers the Requirements gathered such as functional and non functional requirements, various interfaces used, various product features and the attributes which are identified for our system.

Chapter 5- The chapter tells about the Software Design Document. It says about the system architecture used for the system, detailed analysis about the system components, and the user interface developed.

Chapter 6- The chapter tells about the Software Test Document. It tells us about the test approach being used for testing, the test plan designed, and the various test cases mapped with their inputs and expected output.

Chapter 7- It concludes the whole synopsis.

Chapter 2 LITERATURE OVERVIEW

The literature survey we conducted was from these technical papers choosen [1] [2] [3]. RFID-based Tracking and Tracing Information Platform of Logistics technical paper speaks about the how the radio frequency identification technology could be used to find the effective solution for tracing activities. A platform for tracing is proposed by the author. It is divided into three parts. They are Edge Server and the underlying hardware module, Enterprise Server data center module and the Enterprise application module. Edge Server and the underlying hardware module include RFID tags system, reading and writing sensor and antenna network, and Edge Server middleware. Enterprise Server is equivalent to data collection and exchange center to be used in the system. The business module consists of 3 groups. Manufacturers, the third party logistics and the retailers. Author has designed the system and specified the system architecture. The detailed design of the system using MVC design patterns in J2EE platform uses the SSH (Struts + Spring +Hibernate) framework for building the platform. Another paper which was survyed was the Comparative analysis of Linear Regression and Support Vector Regression. It compares both the algorithms using the training dataset along with the classifier. The experiments perfomed using both of these algorithms were evaluated on basis of various parameters. Time taken to build model for SVR is relatively low as compared to the Linear Regression model. Although model time is good for SVR, the mean absolute error are produced less in Linear Regression. This makes the Linear regression model achieve a good amount of accuracy. So we choose the Linear Regression model for our system.

Chapter 3 SOFTWARE PROJECT MANAGEMENT PLAN

3.1 Introduction

3.1.1 Project Overview

The Software Project Management Plan Document gives details about the project deliverables, the organisation of the project and the project plan which will tell us how the project will be moving forward. The Project organisation chapter would consist of the software process model which we will be using for the whole system. It also consists of the Roles and Responsibilities along with the tools and Techniques which will be using for the project. The Management plan would be defining various tasks which need to be carried out. At last, there will be risk identification and categorization along with the impact which those risks will be having on the project. There will be risk sheets prepared with all the details of the risks.

3.1.2 Project Deliverables

Deliverables	Description	Delivery Date
Software Project Management Plan	A complete formal project plan, including technical and managerial processes that will be implemented in the development and delivery of the system	30/09/2018
Software Requirements Specification	A formal document detailing the functional and non-functional requirements of the system	30/09/2018
Software Design Specification	A formal document detailing the component designs as well as the relationships among components	05/10/2018
Software Test Document	Formal documentation detailing scenarios that must be followed in order to ensure that the product software is satisfactorily tested.	14/10/2018
Implementation of frontend and hardware components.	Software files to form the backbone of the system and their integration into the hardware used	14/02/2019
Final Presentation	A demonstration of the product software and a presentation of the project experience	March 2019

3.2 Project Organisation

3.2.1 Software Process Model

The process model used is Prototype Process Model. Prototype is a working model of software with some limited functionality. The prototype does not always hold the exact logic used in the actual software application and is an extra effort to be considered under effort estimation. Prototyping is used to allow the users evaluate developer proposals and try them out before implementation. It also helps understand the requirements which are user specific and may not have been considered by the developer during product design.

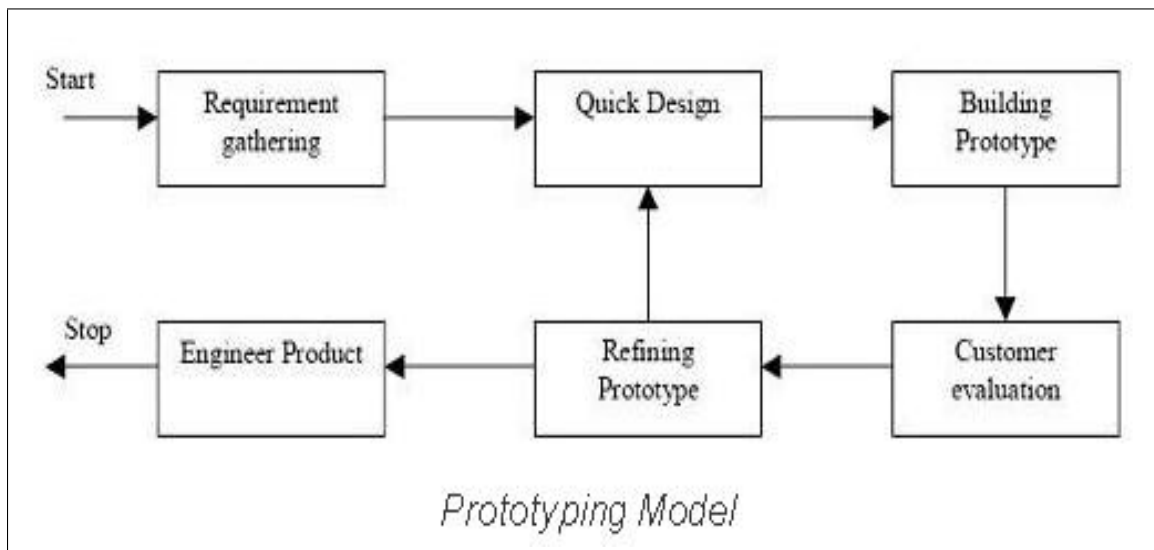


Figure 3.1: Prototype Model Diagram

3.2.2 Roles and Responsibilities

Roles	Responsibilities
Project Co-ordinator	Motivate the team members to perform their tasks in a organized manner Help the team in allocating deadlines
Project Guide	Works with the team to help formulate the application strategy Approves the project documents Helps the team in analysing the project from every perspective and set the goals
Project Manager	Plans, organizes, and coordinates the team project Schedules and prepares team meetings Resolves conflicts Works as a link between team members Monitors and reports the weekly status of the team Ensures that project deliverables are met
Application Designer	Designs a web application to the problem statement that satisfies the requirements Assists the Technical Writers in documenting the design
Application Developer	Develops the android application Determines the data needs for the solution Determines what hardware and tools are necessary Fixes bugs found by the Testers
Database Developer	Develops and populates Databases Ensures proper operation and interaction with entire system application
Tester	Tests all the application modules
Technical writer	Coordinates the project documents and their review by all team members Collects, proofreads, and integrates document parts Generates the final version of all the documents

3.2.3 Tools and Techniques

- LATEX for SRS, SPMP, SDD, STD.
- Gantt project for planning and to prepare the time-line chart.
- Rational rose for UML diagrams.
- Microsoft powerpoint for presentation for the users and project personnel.
- Arduino IDE(v1.8.8) for Arduino Code.
- Tex Editor - Overleaf
- Sublime Editor

3.3 Project Management Plan

3.3.1 Tasks

3.3.1.1 Task 1 - Requirement analysis

Description

Definition of the different requirements which will help the users get a good gist of the project. It provides the basic understanding of the problem and nature of the solution.

Deliverables and milestones

The task provides a list of the various requirements and their analysis for paving the path of design phase.

Resources needed

Effort, time and knowledge about the software.

Dependencies and constraints

The requirements must be documented, testable and related to the needs and defined to a level sufficient for system design.

Risks and contingencies

If the team does not have knowledge about the software, then it can gather the information by communicating with the experts in that field.

3.3.1.2 Task 2 - Software requirement specification

Description

Description of the behaviour of the system to be developed and the features in the scope of the project.

Deliverables and milestones

SRS delineates the features of the project and serves as a guide to the developers.

Resources needed

Latex

Dependencies and constraints

SRS should be documented in a way understandable to other developers to identify the aspects of the system.

Risks and contingencies

There is a high amount of risk if the SRS is not well documented as the features of the system will not be clear.

3.3.1.3 Task 3 - Software design document**Description**

The structure of the software to satisfy the requirements. It specifies the software structure, components, interfaces and data necessary for implementation.

Deliverables and milestones

Architecture design, data design, interface and procedural design.

Resources needed

Latex for documentation and IBM Rational Rose for designing purposes

Dependencies and constraints

SDD is developed according to the SRS, so the SRS should provide an entire overview of the system

Risks and contingencies

Risk is involved if the design does not follow the requirements . The design can be revised by proper communication among the development team.

3.3.1.4 Task 4 - System Test Document**Description**

Specifies the approach that ensures that the features are adequately tested.

Deliverables and milestones

The document includes all the test cases with results done after finishing the development.

Resources needed

Latex and software test plan

Dependencies and constraints

STD should give entire description about features to be tested, amount of testing in order to save time of the testing team

Risks and contingencies

The risk is when the STD does not cover the entire system as this might cause major problem in the future which can be avoided by developing test cases for entire section wise coverage.

3.3.1.5 Task 5 - Coding and Hardware Integration**Description**

Actual programming and functionalities of the application

Deliverables and milestones

The different modules and components of the system.

Resources needed

Arduino, RFID tags and sensors.

Dependencies and constraints

Coding phase depends on the SRS and SDD and should be flexible.

Risks and contingencies

Developers may have insufficient amount of knowledge.

3.3.2 Risk Table

Risks	Category	Probability	Impact	Preventive measures
Server Crash	TI	10%	1	Maintain a distributed server system
Computer crash	TI	20%	3	Powerful computers capable of handling high load
Late delivery	BU	30%	2	Implementation of basic functionality first and parallelism in work
Deviation from Software Engineering Standards	PI	50%	2	Proper design standards and principles must be followed
Poor Quality Documentation	BU	50%	2	Proper understanding of the requirements
Lack of Database Stability	TI	40%	2	Update DB Structure as the traffic grows
Software failure	TI	20%	1	Maximize portability
Staff is inexperienced	ST	40%	3	Self-learning using various resources providing correct knowledge
No internet Connection	TI	10%	1	Maintain a backup hotspot/tethering service
Conflict with other traffic	TI	10%	1	Shield the high frequency signals from external noise
Failure of Scanner	TI	30%	2	Facilities of updating the database manually.
Damage of RFID	TI	20%	2	Attach the RFIDs in such a way they aren't easily accessible.

Impact Values:

- 1 – Catastrophic
- 2 – Critical
- 3 - Marginal
- 4 - Negligible

3.3.3 Risk Template-1

Risk information sheet			
Risk ID: 1	Date: September 30, 2019	Probability: 40%	Impact:2
Description: The database maintained may not be stable which may lead to Database Instability.			
Refinement/Context: Sub-condition 1: The information gathered was misinterpreted.			
Mitigation/Monitoring: 1. Re-gather the information from the user. 2. Understand with modules are improper and correct them.			
Management/Contingency plan/Trigger: Contact the team leader and make a new increment with all the respective changes needed.			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	

3.3.4 Risk Template-2

Risk information sheet			
Risk ID:2	Date:September 30, 2019	Probability:10%	Impact:1
Description: The internet connection fails.			
Refinement/Context: Sub-condition 1: Connection fails due to fault at the ISP. Sub-condition 2: There is some lose contact of cables to the modem.			
Mitigation/Monitoring: 1. Contact the ISP provider and resolve the issues. 2. Fix the cable properly			
Management/Contingency plan/Trigger: Provide some alternative solution for internet like hotspot. Or make a provision for offline data storage.			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	


3.3.5 Risk Template-3

Risk information sheet			
Risk ID:3	Date:September 30, 2019	Probability:30%	Impact:2
Description: The scanners fail to read the RFIDs.			
Refinement/Context: Sub-condition 1: RFID / Scanner interface is tampered. Sub-condition 2: Incompatible scanner and RFIDs			
Mitigation/Monitoring: 1. Make sure the interface is clean. Check if the RFID is not damaged. 2. Try scanning the same RFID on another sensor or scan another RFID on the sensor.			
Management/Contingency plan/Trigger: Get technical assistance as soon as possible. Find out the problem location (Scanner or RFID) and take actions accordingly			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	

3.3.6 Risk Template-4

Risk information sheet			
Risk ID:4	Date:September 30, 2019	Probability:20%	Impact:2
Description: RFID tag gets damaged.			
Refinement/Context: Sub-condition 1: The RFID tag is missing from a cargo when it reaches a scanner. Sub-condition 2: The tag is damaged due to weather conditions / mishandling, etc.			
Mitigation/Monitoring: 1. . Try to identify the cargo with the marking on it. 2. Wait for the user to raise a ticket when he notices that one of his cargo containers has not reached the desired destination yet. The cargo can then be linked with the ticket and the right identity of it can be found. A new RFID can be associated to it at this point and the tracing can continue as before.			
Management/Contingency plan/Trigger: Have spare RFIDs at all counters.			
Current status: Mitigation steps have been initialized.			
Originator:		Assigned:	

3.4 TimeTable



Name	Begin date	End date
☐ • Initialisation	4/16/18	4/16/18
• Problem Definition	4/16/18	4/16/18
☐ • Planning	9/24/18	10/19/18
• Project Plan (SPMP)	9/24/18	9/30/18
• Requirements (SRS)	10/1/18	10/7/18
• Design (SDD)	10/8/18	10/11/18
• Test Case Design (STD)	10/12/18	10/19/18
☐ • Implementing the base	10/20/18	1/11/19
• UI development	10/20/18	11/10/18
• Configuration of hardware	11/11/18	12/10/18
• Linking database	12/11/18	1/11/19
☐ • Implementing the logic	1/12/19	3/12/19
• UI phase 2 development	1/12/19	1/31/19
• Implement DM algorithms	2/1/19	3/2/19
• Integreate modules to the mai...	3/3/19	3/12/19
☐ • Final Documentation	3/13/19	4/11/19
• Technical Paper	3/13/19	3/22/19
• Testing	3/23/19	4/11/19
• Demonstration	4/12/19	5/11/19

Figure 3.2: Gantt Chart

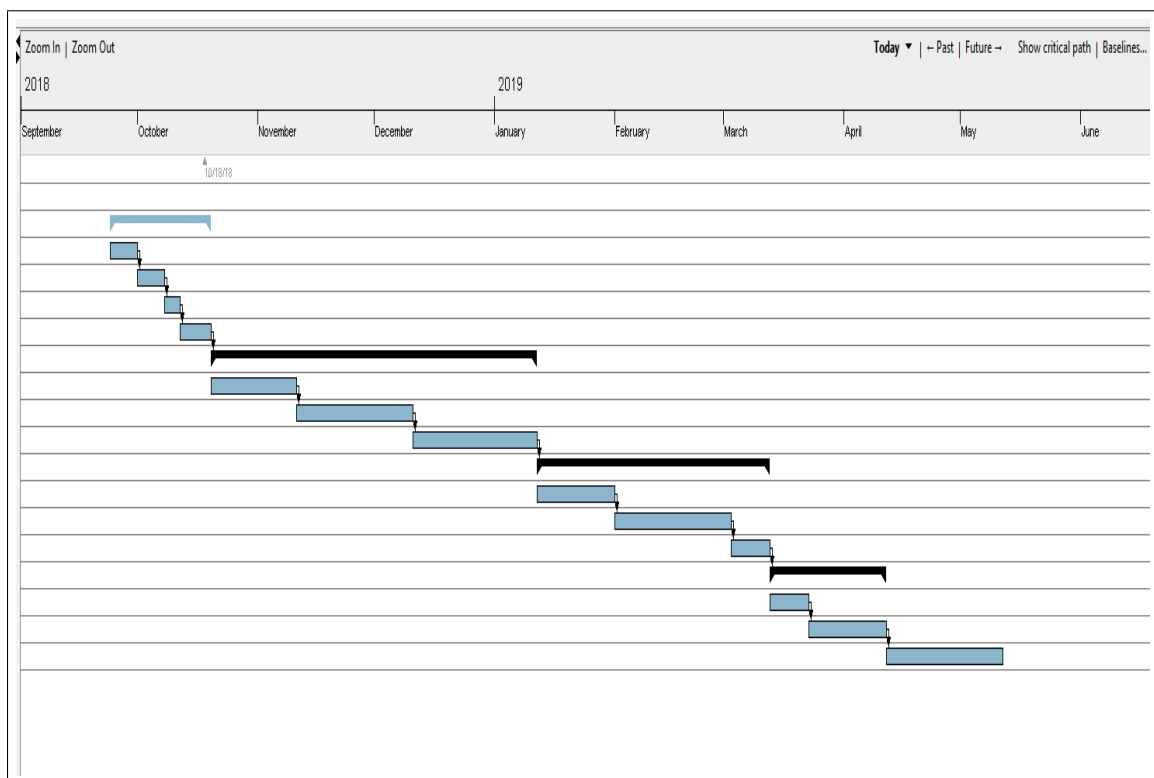


Figure 3.3: Gantt Chart

Chapter 4 SOFTWARE REQUIREMENTS SPECIFICATION

4.1 Introduction

4.1.1 Purpose

Industrialization has proved to be one of the biggest factors for globalization all over the world. Due to globalization, these industries are achieving business growth at a very rapid rate. The sellers are producing goods in one country and selling the same in other, thus making huge profits. There is no system to keep a track of the goods being transported. Thus to help the seller, we propose a Cargo Tracing and Business Analysis System which helps in Goods Tracing for the seller. We also propose to make a business analysis model which will study the trends or patterns in the past records and predict the appropriate market place for the seller.

4.1.2 Project Scope

Due to human errors, at times the shipments are mishandled and reach wrong destinations. The proposed Cargo Tracing and Business Analysis system will help the seller to track the shipments made to different locations and thus using this system the errors can be handled with the help of pre-defined triggers to send notifications when a particular cargo which is expected to arrive at a checkpoint, does not arrive. Along with this, future market place suggestions will also be provided to the seller based on analysis of current trends based on previous records containing details about Shipment details, Source, Destination, Region type, Season, Date/time and so on. Furthermore, the system also helps the shipment industry to manage orders based on the profit related to particular season, destination and cargo type.

4.2 Specific Requirements

The following Sections consists the requirement which are necessary for the proposed system to function at its optimal limits without any hindrances.

4.2.1 External Interface Requirements

This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces which provides basic knowledge of the system.

4.2.1.1 User Interface

- The Cargo Tracing and Business Analysis System helps the seller of the goods to trace the cargo. For this, the seller needs to register on the website provided. First there will be a registration process which is to be followed by the seller. The registration form includes the following fields-
 - Name
 - Address
 - Contact Number (Personal and Office)
 - Country
 - Seller Registration Number
 - Email-id
 - Username
 - Password
- After successful registration, the seller will be able to log-in to the system. All the sellers who are successfully registered will be assigned a unique id which they can see in their profile. This unique id will help the seller to trace their respective cargo. As the cargo passes the checkpoints, the seller would be notified about it on the timeline of the cargo shipment. This will also be supported by notifying the user about the date and time on which the shipment has cleared that checkpoint. Thus, the user interface is simple, compact and reliable.

4.2.2 Hardware Interface

The proposed system consists of RFID tags, the scanner as the hardware for the system. The following are the hardware required for the system-

1. Rc522 RFID Scanner
2. EspNode MCU 8826 Wifi Module
3. Arduino Board or any other Uno Board
4. Any RFID Tag

The Wifi Module used in the system will be able to send data to the database of the system where all the values will get stored. The RFID tags are independent of the scanner used, so any type of the scanner can be used at different checkpoints as based on availability.

4.2.3 Software Interface

Each and every RFID tags will have their unique hex number pre included with them. The tags will generate the binary files when they are scanned by the scanner. There will be a main code written in the Arduino/ Uno Board that establishes connection and control with all the hardware modules mentioned. Both the Rc522 scanner and the Wifi module are connected with arduino with the drivers provided. These interfaces will be used for about the scanners and RFID tags. Now, the web portal will be designed by using the following interfaces.

- Server side
 - An database server and web server which will accept all requests from the client and forward it accordingly. A database will be hosted centrally using MySQL.
- Client side
 - An OS which is capable of running a modern web browser which handles requests and responses of the communication.
 - A web browser with the Javascript and PHP technologies enabled.
 - Android Version of Jellybean(4.1) and above would be recommended for the mobile devices.

4.2.4 Communication Protocols

As mentioned above, the scanner will be using the Wifi Module for communication between the database and the RFID tags when scanned successfully. HTTP, FTP are the communication protocols used in the system for the web portal. To connect to back-end processing with PHP JSON is useful.

4.2.5 Software Product Features

The proposed system will provide the seller with own portal to track the shipment made by the seller. The dashboard will help the seller to track all their shipment activities on one screen. This dashboard will keep updating the seller regarding all the activities at the checkpoints during the whole process. The following are the functional requirements gathered for the proposed system.

4.2.5.1 Login

The system will help the seller to login to the system to be able to trace the cargo shipment as and when required. For successful login to the portal, the seller has to enter the correct credentials for the username and password otherwise the system will throw error.

4.2.5.2 Register

Any seller from any country can register itself to the portal of Cargo Tracing. For this, registration process would require personal details like name, number, address, email, passport number to be entered by the seller. On successful registration, the seller would then be able to use our system. All the personal details are kept secured.

4.2.5.3 Filter

As there can be number of shipments made by the seller, the seller may need to segregate out some of the shipments made. To make this happen, the seller would have an option to filter the shipments based on the date of shipment, the month of shipment or the year of shipment.

4.2.5.4 View Timeline

The seller would be able to trace the shipment based on a timeline provided on the dashboard. The timeline specifies the shipment details such as the date of scanning of shipment at a particular time, the arrival time, the departure of the shipment and so on.

4.2.5.5 Scanning

At the checkpoints, the cargo shipment with RFID would be scanned at that location. The RFID scanner present at the checkpoint will trigger the database of the system after successful scanning of RFID stating the name of the checkpoint and the location of the same, the arrival date of the shipment and the arrival time of the shipment. The shipment would be then transferred for the further processing either to next checkpoint or it will be unloaded.

4.2.5.6 Erasing

When the shipment passes through the last checkpoint and the cargo is unloaded, the RFID tags will be erased of all the data which it collected in the whole journey from source of shipment to the destination of the shipment. These tags will be erased programmatically and all the tags will be reused again for other shipments.

4.2.6 Software System Attributes

The following are the system attributes:

4.2.6.1 Reliability

The final System will be a reliable one as it will pass through a series of testing procedures regarding the working of whole system. The trust factor of the seller relies on giving the perfect tracing of the cargo shipment along with the business analysis of the seller with high precision. The mean time of failure is very large as the deployment of web portal will be good enough to handle large requests. The probability of the system unavailability largely depends on the scanners used for RFID tags. The weather conditions mainly at port may cause hindrance in scanning and may make system unavailable for some time. Thus the rate of failure occurrence of the system is quite small.

4.2.6.2 Availability

Some times due to load on the server which is in process may malfunction. This may cause non-availability of the system. To recover from this inconsistent state, there will be checkpoints saved of the previous working and stable version of the system which will help in successful recovery of the system. The recovery phase of the system will not take long duration, so maximum of half hour would be required to recover the system to it's normal state. The scanners will be available whenever required till the continuous supply of electricity is provided.

4.2.6.3 Security

This feature is important aspect of our system as the system will be used world-wide. When the seller enters critical information of identity such as the Passport number, the seller license number, these values will not be directly stored in database but with the help of cryptographic algorithms such as SHA or MD5 will be applied on those values. Every care would be taken to protect the RFID from being forged or stolen. This can be implemented

by reviewing the RFID data and logs at each checkpoint.

4.2.6.4 Maintainability

The maintenance of the system is quite easy work. The web portal for the seller is easier to maintain. Any upgradation to the portal can be easily managed as there are no complex structures, API's used nor any kind of third-party services. The CSS, Javascript, PHP codes, are all developed by developers from the team so the code is easier to maintain and well readable. The system will be stable when new releases or the version of the system are rolled out.

4.2.6.5 Portability

Requirements do change with the time based on stakeholders decision. So to meet the needs of stakeholders, there will be changes made to the system. The newly developed system will easily adapt to the current working environment and there will be some functionality added to the system which will be used for reverse compatibility.

4.2.6.6 Performance

The performance of the system or to be specific the web portal will depend on internet speed. The performance of the RFID tags will depend on how well the tags are kept secured. As the weather conditions change from country to country at a particular time, the tags might get damaged. Moreover, the scanner should be placed at a location where they do not catch any moisture or not get damaged while the containers are transported. Any mishandling of scanners or tags might result in degradation of the entire system.

4.2.7 Database Requirements

The Web Portal will maintain a database which will contain all the details of the seller such as the number, name, address. The passport number and other confidential details will be kept secured. The other details of shipment such as the shipment number, the source, the destination, and also the intermediate checkpoints will also be stored to help the seller in tracing of shipment. The Authority to access the database will be given only to the administrator who will maintain the web portal. There will be another attributes which will be stored in database to keep track of the RFID unique hex values along with the other details such as timestamp and so on.

Chapter 5 SOFTWARE DESIGN DOCUMENT

5.1 Introduction

5.1.1 Design Overview

This document presents the Software Design Document for the Cargo Tracing and Business Analysis project. This document gives insights about overview of the document. The requirement traceability matrix gives mapping about the requirements identified and the components in which they are fulfilled. With extensive research we chose Event driven architecture and Client Server Architecture as the architectures of our system. The system interface that is the hardware and software interfaces are mentioned in the document. Various components are described to a good extent which will help in making the system working. The document provides the description of user interface along with some of the images of the UI developed.

5.1.2 Design Requirement Traceability Matrix

Requirements \ Components	Web Portal	RFID Scanner	Database
Trace Cargo	X	X	X
Business Analysis	X		X
Payment	X		
Login	X		X
Scanning RFID		X	X
Updation at Checkpoint			X
Register Seller	X		
Erasing RFID Data			X
View TimeLine	X		

5.2 System Architecture

5.2.1 Chosen Architecture : Event Driven Architecture and Client Server Architecture

An event-driven application is a computer program that is written to respond to actions generated by the user or the system. In a computing context, an event is any identifiable occurrence that has significance for system hardware or software. As such, events include both user-generated actions like mouse clicks and keystrokes and system-generated events such as program loading or IDs passing through scanners. Event-driven programming separates event-processing logic from the rest of a program's code.

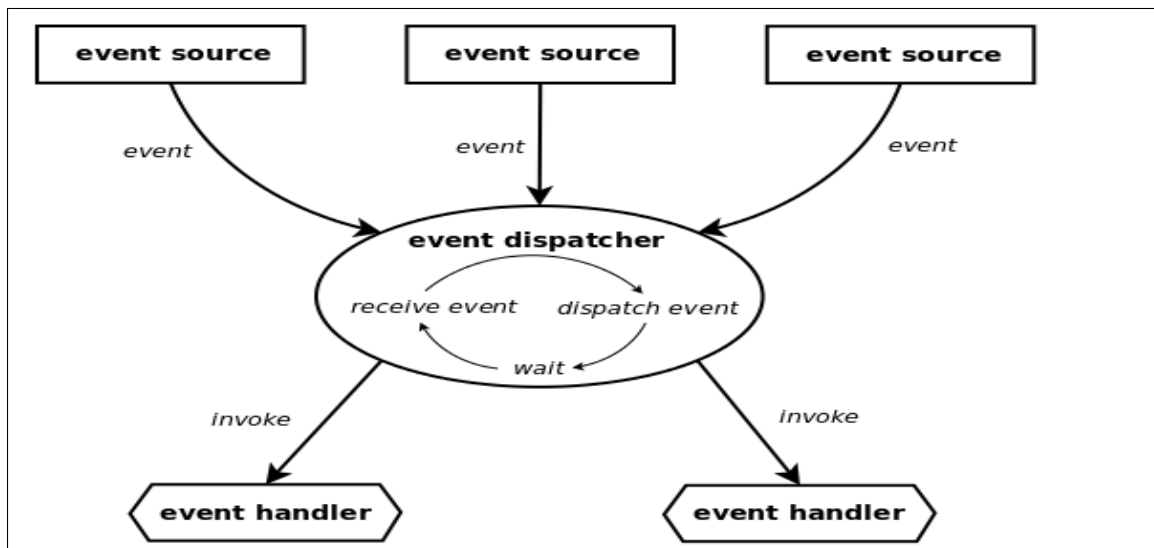


Figure 5.1: Event Driven Architecture

Client-server architecture, architecture of a computer network in which many clients (remote processors) request and receive service from a centralized server (host computer). Client computers provide an interface to allow a computer user to request services of the server and to display the results the server returns. Servers wait for requests to arrive from clients and then respond to them. Ideally, a server provides a standardized transparent interface to clients so that clients need not be aware of the specifics of the system (i.e., the hardware and software) that is providing the service. Clients are often situated at workstations or on personal computers, while servers are located elsewhere on the network, usually on more powerful machines.

This computing model is especially effective when clients and the server each have distinct tasks that they routinely perform. In our Cargo Tracing system, for example, a client

computer can be running an application program for entering all the shipment related information while the server computer is running another program that manages the database in which the information is permanently stored. Many clients can access the server's information simultaneously, and, at the same time, a client computer can perform other tasks, such as sending e-mail. Because both client and server computers are considered intelligent devices, the client-server model is completely different from the old "mainframe" model, in which a centralized mainframe computer performed all the tasks for its associated "dumb" terminals.

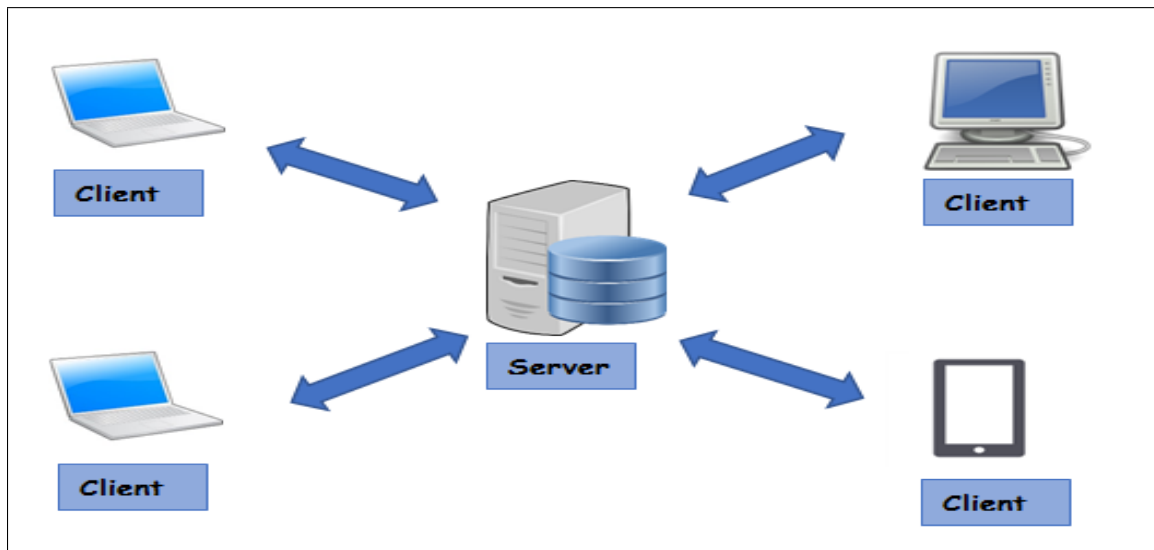


Figure 5.2: Client Server Architecture

5.3 System Interface Description

5.3.1 Hardware Interface

- The mandatory hardware requirement is a computer or an android device with basic android version of jellybean. This low specification is mainly due to the simplicity of design.
- Internet connection is a necessity.
- Decent RAM size and storage space required.
- RFID
- RFID Scanners

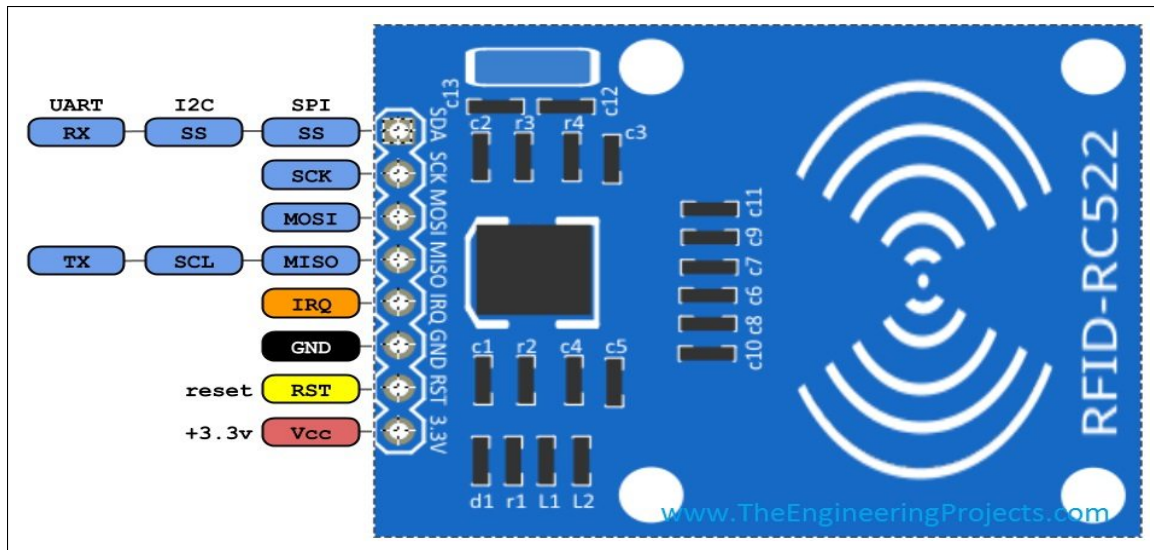


Figure 5.3: RC552 Scanner

- Aurdino Board

5.3.2 Software Interface

- A web browser with minimum compatibility of JavaScript.
- An android device with minimum android version of jellybean.

5.4 Detailed Description Of Components

5.4.1 Server Connection

This component consists of php files which are downloaded locally on the hardware along with some on the server. These files help to connect with the sever and pass on the data on the server side from the client side in the form of files like text or xml. Thse files are interpreted by the server side and response in the form of JSON is given back to the client by the php files. These files also verify the data sent. Server side scripting and validation is also used.

5.4.2 Verification

Although there are php files to verify before sending the data, the software must contain verifying files for the data to be sent and received. Moreover the device must make sure that the data is sent by a human and not by any illegal practice. Basic validation also must be done separately of every data.

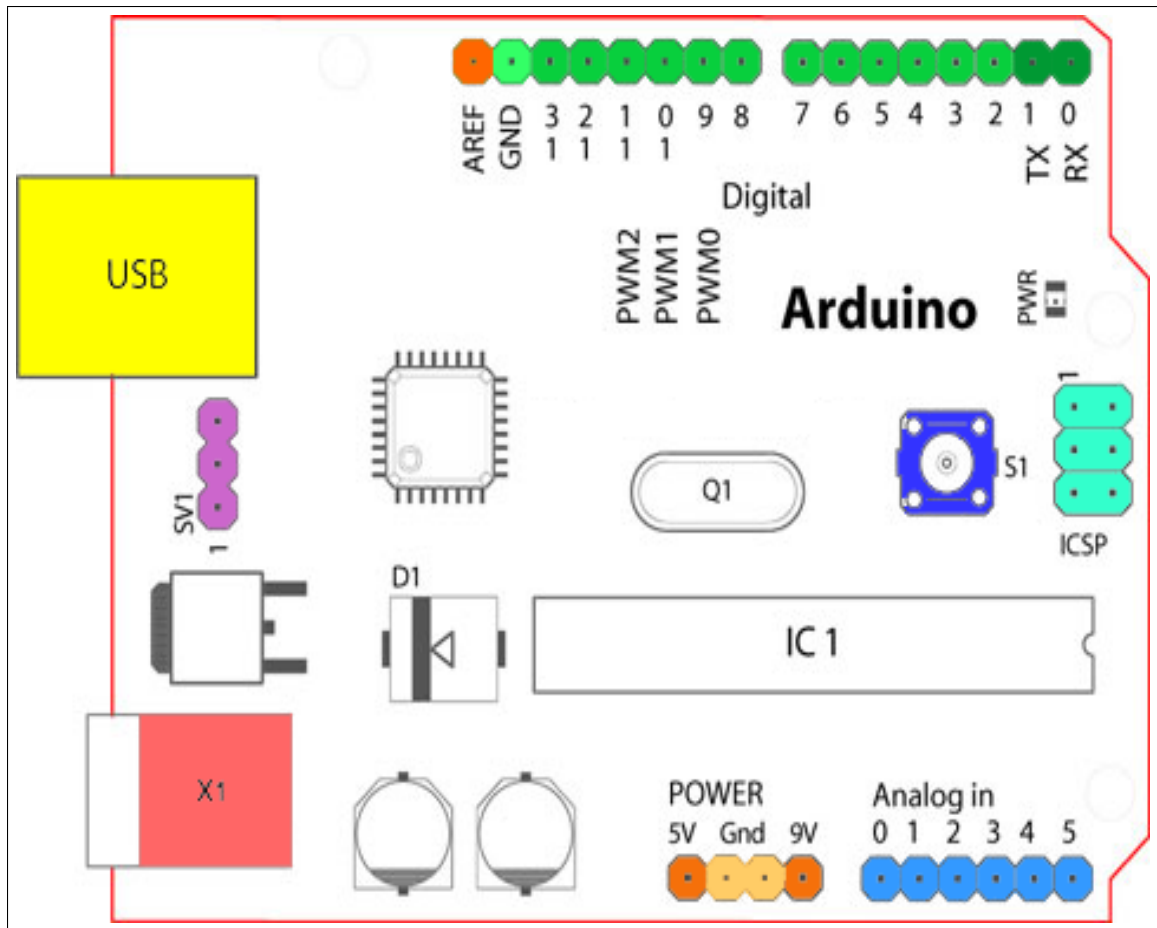


Figure 5.4: Arduino Board- MEGA

5.4.3 RFID tags

RFID tagging is an ID system that uses small radio frequency identification devices for identification and tracking purposes. An RFID tagging system includes the tag itself, a read/write device, and a host system application for data collection, processing, and transmission. An RFID tag (sometimes called an RFID transponder) consists of a chip, some memory, and an antenna. RFID tags in the project will be used to track shipments, assigning them user data, unique identity to each shipment for scanning and tracking purposes.

5.4.4 RFID Reader

For interacting purpose of RFID tags, we have developed our own RFID scanning mechanism with hardware: RC522 - RFID Reader / Writer 13.56MHz includes a 13.56MHz RF reader cum writer module that uses an RC522 IC and two S50 RFID cards. The MF RC522 is a highly integrated transmission module for contact-less communication at 13.56

MHz. RC522 supports ISO 14443A/MIFARE mode. RC522 - RFID Reader features an outstanding modulation and demodulation algorithm to serve effortless RF communication at 13.56 MHz.

5.4.5 Wifi Module

For interaction purpose with our website we are using ESP MCU Node8266 module basically, NodeMCU is an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module

5.4.6 Arduino

Arduino is a basic micro-controller used to handle all the hardware components present in the system, main underlying code of basic functionality of the system like scanning of ID tags, writing of ID tags, communication with wifi module to send or receive data to/from a centralized database of a website. Arduino is an open-source computer hardware and software company, project and user community that designs and manufactures single-board micro-controllers and micro-controller kits for building digital devices and interactive objects that can sense and control objects in the physical and digital world. The micro-controllers are typically programmed using a dialect of features from the programming languages C and C++.

5.5 User Interface

5.5.1 Description

The User will have an option initially to sign up or to log in in an existing account. Further once the user has logged in to his account they can track their shipments on the website. The user gets an option to know the possible prediction of transaction by providing basic details of the transaction. The user gets an option to manage the account where he can change his account type, edit his information, manage his security options and many more.

5.5.2 Images

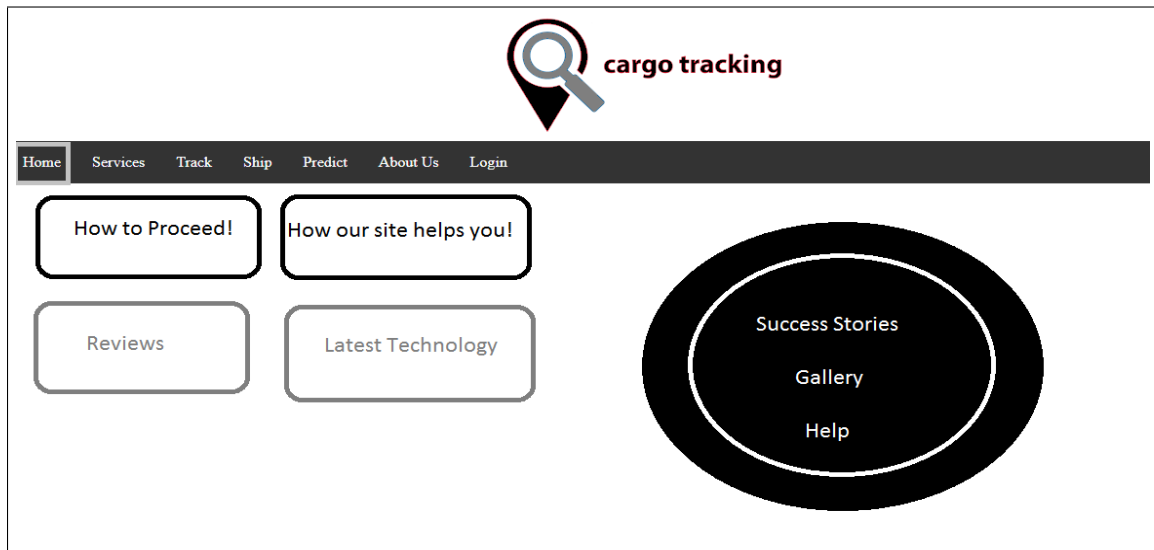


Figure 5.5: Home Page

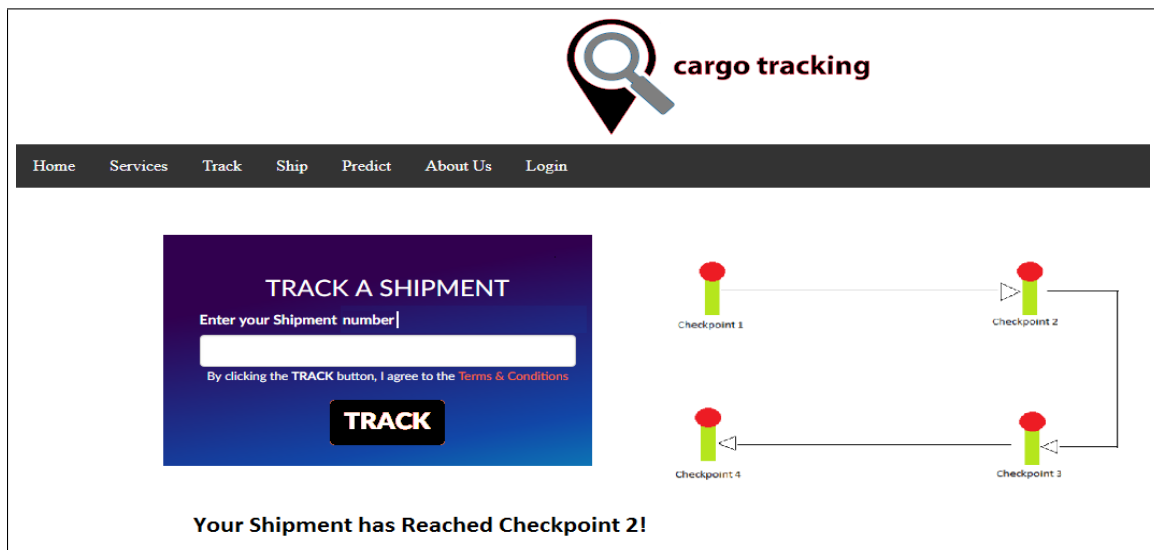


Figure 5.6: Tracking Page

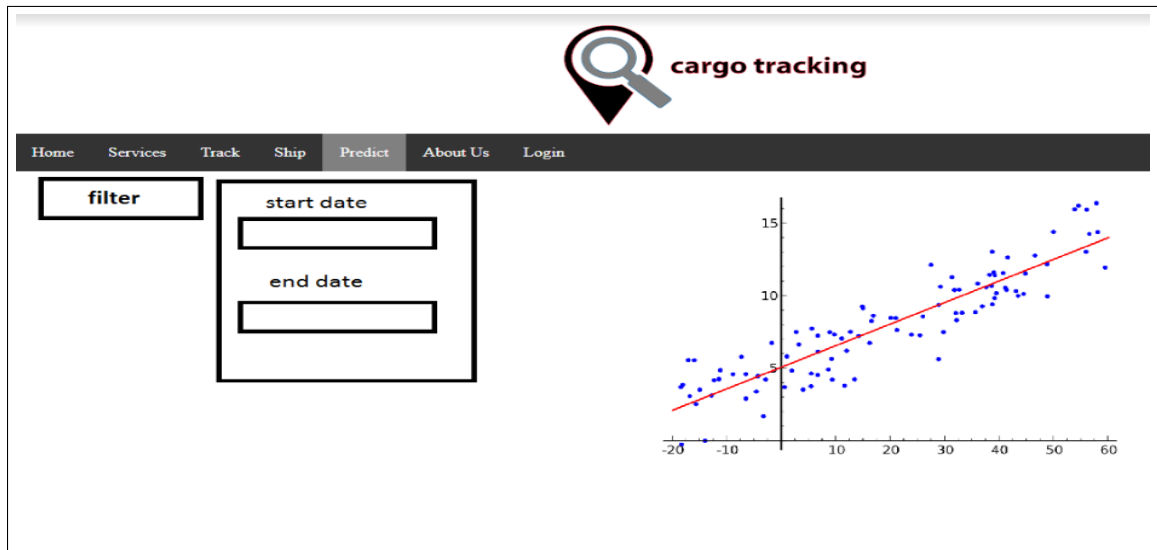


Figure 5.7: Business Analysis

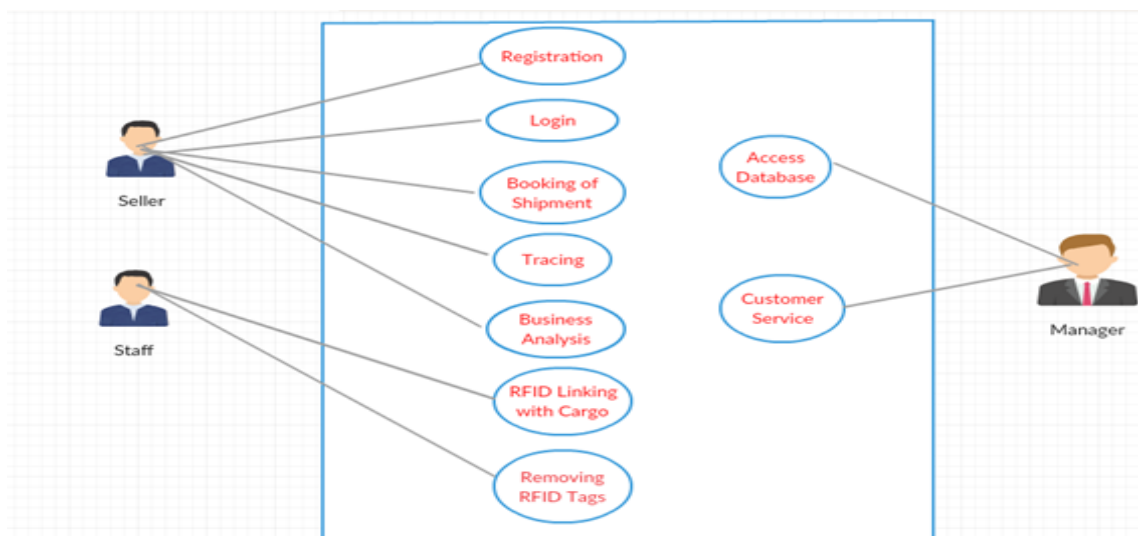


Figure 5.8: Use Case Diagram

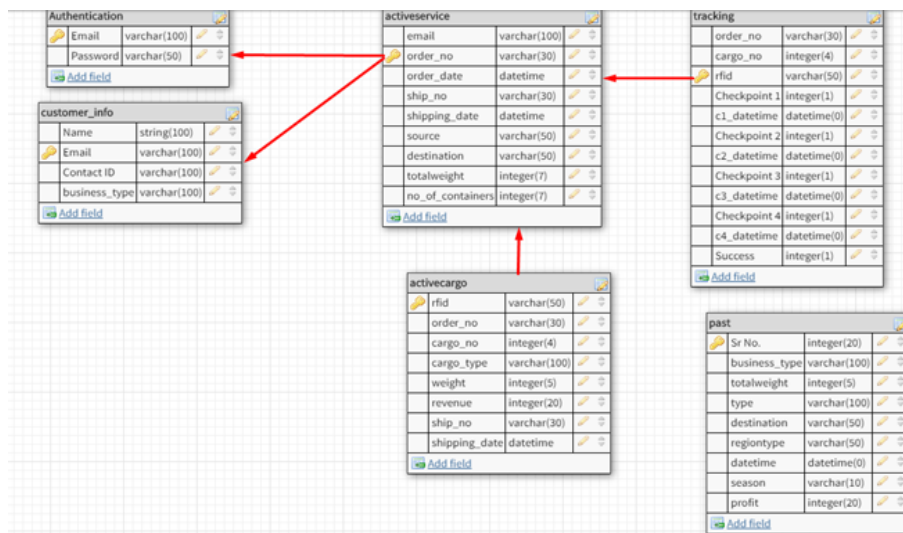


Figure 5.9: Database Schema Diagram

Chapter 6 SOFTWARE TEST DOCUMENT

6.1 Introduction

6.1.1 System Overview

The goal of the system is to manage shipments of the customers by providing them tracking of their goods using RFIDs and also giving business recommendation to cargo suppliers based on their cargo shipment order. The system has to be up and running at all times to ensure proper updation of checkpoints and tracking. Staff should be trained enough to be able to use the scanners and update the database by scanning the RFIDs. The scanners should have uninterrupted internet connection to not cause any delay. RFIDs should be cleared from database once a particular journey is over. To ensure smooth functioning of the whole system, it is really necessary for these conditions to be met at all times.

6.1.2 Test Approach

6.1.2.1 Unit Testing

Unit testing is a method of testing that verifies the individual units of source code are working properly. The goal of unit testing is to isolate each part of the program and show that the individual parts are correct. Unit testing is simplified when a component with high cohesion is designed. When only one function is addressed by a component, the number of test cases is reduced and errors can be more easily predicted and uncovered. Here, we can test the working on scanners, mobile app connectivity, server based notifications all separately.

6.1.2.2 System Testing

Once the entire system has been built then it has to be tested against the Software Requirement Specification and System Specification to check if it delivers the features required. System testing can involve a number of specialist types of test to see if all the functional and non-functional requirements have been met. System testing can be applied to our project when the prototype of RFID + Scanners is generated along with a sample dataset.

6.1.2.3 Performance Testing

The system should meet the performance requirements as mentioned in the SRS. The performance will be evaluated based on the response time of the GUI and the database commands. The system will also be tested based on the load (number of active users at a time) on the servers and the ability of the system to handle database stability when various transactions are taking place simultaneously.

6.1.2.4 Functional Testing

The functional requirements specified in the SRS must be met at all costs. Tests can be applied to check if the tasks supposed to be fulfilled by the system are being successfully executed. A sample cargo order can be used as it can be simulated such that the system thinks it's travelling. The end point can be studied to ensure that the functional requirements are met.

6.2 Test Plan

Each module can be tested separately to check if it works. Functional Testing, Unit testing can all be done as soon as development of a particular module is completed. The product can be tested completely when the first prototype is ready which will also verify the integration aspect of the whole system, data flow and usability.

6.2.1 Features to be tested

- Login
- Registration
- Place an order for shipment / Book shipment.
- Trace cargo using mobile app / website
- Scan RFIDs at different places using scanners as checkpoints
- Business suggestion

6.2.2 Features not to be tested

- Reset Password: Is not a mandatory functionality of the system. Can be skipped for demonstration purposes.
- Checking if all RFIDs are unique: All RFIDs are unique.

6.2.3 Testing Tools and Environment

The project consists of various modules both at the software and hardware part. Each module can be tested as and when the development is complete. Well prepared documentations of the hardware which will be used will make the testing process easier as most of the steps and working will already be described. The goal is to go step by step, gradually, making sure that the work done is correct.

6.3 Test Cases

6.3.1 Test Case: Registration

6.3.1.1 Purpose

Our system will help the sellers from all over the world to register themselves to our system. Therefore, it is necessary that the registration feature works on all kinds of devices and perform its desired functionality. As the seller tries registering, the system should not be in ambiguous state i.e. it should either register a particular seller successfully or it should display an error.

6.3.1.2 Inputs

Name, Address, Email, Phone, Seller Registration number, Passport number, DOB, password.

6.3.1.3 Expected Output and Pass/Fail Criteria

If all the input details are valid then the seller's details must be stored in the database and the seller must be directed to Login page to log to the system. If the details are not valid the web portal must show where the error have occurred along with the error fields highlighted. The test case will be passed only when all the values entered in registration form will be valid and successfully submitted. Otherwise, the test case will fail on invalid input values.

6.3.1.4 Test Procedure

Verify whether the user has input text values with no special characters and blank spaces wherever required. Also verify the code for database connectivity so that successful registration adds the seller details to the database. Check the password size with necessary security requirements.

6.3.2 Test Case: Login

6.3.2.1 Purpose

To verify the seller with correct login credentials so that seller could be authenticated successfully to use our system.

6.3.2.2 Inputs

Email, Password.

6.3.2.3 Expected Output and Pass/Fail Criteria

If the email id and password both are correct and matching with the values stored in database, the seller would be redirected to the dashboard of our system from where the seller can manage all the activities provided by the system. The test case will be passed if the values are matched with the database values and it will fail if the values are not matching the values.

6.3.2.4 Test Procedure

Check if the length of the password matches with the standard length specified. Also see whether the email id is of correct pattern.

6.3.3 Test Case: Shipment Booking

6.3.3.1 Purpose

When seller has registered to our system successfully, seller is authorised to use our tracing facility by booking the cargo on web portal. The portal would ask some details about the cargo like the quantity in tons, the type of cargo whether it is fragile glass or clothing or food kind. It is necessary to book the cargo to trace it using our system.

6.3.3.2 Inputs

Type of Cargo, Quantity, Date of Shipping, Time of shipping, source, destination.

6.3.3.3 Expected Output and Pass/Fail Criteria

If the date is available for booking along with the time slot, the cargo will be successfully booked for that seller. The test case will be passed if all the values are valid and then only the shipment will be booked successfully. The Fail Criteria will be when the seller enters insufficient details or the seller will try to book when the dates are not available.

6.3.3.4 Test Procedure

Check the maximum and minimum values of all the details to be entered whether they pass the test or not. It is necessary to check the database for the correct booking so that there is no inconsistency in the field values.

6.3.4 Test Case: Tracing

6.3.4.1 Purpose

Tracing is the main part of the system so it is necessary to make sure that the seller gets the correct details of cargo whereabouts. The updates when the shipment reaches from one checkpoint to another will be notified. It will make the seller to know the exact details of the shipment where it has arrived.

6.3.4.2 Inputs

Source, destination, date, time.

6.3.4.3 Expected Output and Pass/Fail Criteria

The system should give correct results when the shipment is in transit. Whenever the RFID is scanned properly the database will be updated and then the seller will be able to see the shipment tracing on web portal with the details such as checkpoint place, time and date. The test case will be passed when the seller will be notified about the correct details of tracing along with the date and time. On the other hand, the test case will fail when the seller will not be able to see the tracing details or incorrect details.

6.3.4.4 Test Procedure

Check if the scanning and updating of database is done properly. Check the values fetched from database and show it to particular seller. Make sure that if load comes to the database fetching values and matching to every seller is correctly done.

6.3.5 Test Case: RFID Scanning

6.3.5.1 Purpose

When the shipment arrives at the checkpoints, it is required that the RFID attached to the cargo are successfully scanned at that location. When these RFID are successfully scanned, they will trigger the database for updation of checkpoint with date and time. If it is not scanned properly or there is malfunction in scanning, the database trigger will not happen and the seller would not be able to trace the cargo at that particular location.

6.3.5.2 Inputs

RFID Tag

6.3.5.3 Expected Output and Pass/Fail Criteria

After scanning the RFID tag successfully, the database will be triggered setting the values of the checkpoint location along with the date and time of the scanning. These values will be set in the database. These will be used to provide tracing to the seller on the web portal provided. The test case will be passed when the RFID is scanned successfully and failed when scanning is failed.

6.3.5.4 Test Procedure

RFID scanning will be done manually by the staff members present at the checkpoints. The RFID will be scanned by trying at different distances from the scanner. They will also be checked for scanning at different angles from the scanner. Suppose if the RFID is broken during the transit of cargo, the broken RFID will also be tested for scanning. They will be checked if they are scanned or not.

6.3.6 Test Case: RFID Erasing

6.3.6.1 Purpose

After a particular cargo arrives at the destination intended, the RFID will be collected to reuse them for other cargo shipments. These RFID need to be erased of the information collected during the whole journey. The values from the database also need to be made invalid for that particular erased RFID.

6.3.6.2 Inputs

RFID tag

6.3.6.3 Expected Output and Pass/Fail Criteria

The RFID information will be erased successfully and can be used for other tracing process. The values from the database would be made invalid after the RFID's are erased. The new RFID will not contain any information and can be reused. The testing will be passed when the information is totally erased with all the information contained in them and it will fail when either some information is present in tag or nothing is erased.

6.3.6.4 Test Procedure

Collect the RFID tags which are arrived at the destination location. These RFID tags will be then erased by the staff members present at that location. There will be manual testing of erasing and scanning done to ensure that erasing and feeding of information is done successfully without any error in the process.

6.3.7 Test Case: Business Analyzer

6.3.7.1 Purpose

After a particular cargo arrives at the destination intended, the RFID will be collected to reuse them for other cargo shipments. These RFID need to be erased of the information collected during the whole journey. The values from the database also need to be made invalid for that particular erased RFID.

6.3.7.2 Inputs

All the attributes which are needed for analysis for ourselves along with the shipment source, destination, quantity, type of the shipment, etc.

6.3.7.3 Expected Output and Pass/Fail Criteria

The analysis will be done on the values provided by database. After successful completion of the deductions and calculations on the values, the output will be provided as a graphical analysis. The graphs should be properly plotted with a good amount of accuracy. The test

cases will be passed when the accuracy will be above the specified standard limit and the graphs show the correct results made by analysis. The cases will fail when the accuracy will be less.

6.3.7.4 Test Procedure

Give the model constructed different amount of values such as first give the model 60% of the values and measure the accuracy. For next iteration, provide the model with 70% of data and again measure the accuracy. Consider also the cases where there will be missing values in the dataset.

6.3.8 Test Case Table

ID	Module	Input/Test Case	Expected Output	Actual Output	Result
1	Registration	Seller doesn't enter field values according to conditions	Display condition messages	-	-
2	Registration	Seller forgets to enter the required field values	Display required field values	-	-
3	Registration	Seller uses same password as username	Display message saying username and password cannot be same	-	-
4	Registration	Seller enters all the details	Successful Registration message	-	-
5	Login	Seller enters username which doesn't exist	Display message to enter proper username	-	-
6	Login	Seller enters wrong password	Display invalid password message	-	-
7	Login	Seller enters correct details	Successful login and seller will be redirected to dashboard	-	-

ID	Module	Input/Test Case	Expected Output	Actual Output	Result
8	Login	Seller checks the remember me option	Successful login and credentials will be saved	-	-
9	Shipment Booking	Seller enters quantity of a particular shipment type as zero	Display error message saying quantity should at least be one	-	-
10	Shipment Booking	Seller enters same destination and source	Display message indicating source and destination should not be same	-	-
11	Shipment Booking	Seller tries to book shipment for passed date	Notification should be made regarding the same	-	-
12	Shipment Booking	Seller enters all details correctly	User will be redirected to Payment page	-	-
13	Tracing	Two or more shipments arrive at same time	Display shipment numbers along with the date and time	-	-
14	Tracing	Updation error in database	Display message indicating error at back end	-	-

ID	Module	Input/Test Case	Expected Output	Actual Output	Result
15	Tracing	RFID broken during the transit	Inform user about the problem and suggest methods to resolve the issue	-	-
16	Tracing	RFID scanned correctly	Seller will be able to trace shipment	-	-
17	RFID Scanning	A broken or damaged RFID	Display error message saying RFID damaged	-	-
18	RFID Scanning	Scanner not working properly	Display error message saying can't scan RFID	-	-
19	RFID Scanning	Scanner scans RFID	Display message saying RFID scanning successful	-	-
20	RFID Erasing	RFID not placed properly for erasing	Display error message RFID not placed properly	-	-
21	RFID Erasing	RFID placed properly for erasing	Erase all the data stored with only RFID number remaining	-	-

ID	Module	Input/Test Case	Expected Output	Actual Output	Result
22	Business Analyzer	Data from database or archived data	Calculating results and displaying in graphical format	-	-
23	Business Analyzer	Missing attributes provided as input	Display error message for missing attribute value or show nothing	-	-

Chapter 7 CONCLUSION

As all the necessary documents and the work has been presented, the system will be surely helpful for the sellers. We have adopted a prototype model for this project, so there will be continuous improvement of this system by taking the reviews from the sellers and by making inspections from the industry experts. This system will indeed help the sellers to trace their cargo shipments and analyze the business growth using this system.

Chapter 8 REFERENCES

- [1] Wei He, Eng Leong Tan, Eng Wah Lee, and TY Li. A solution for integrated track and trace in supply chain based on rfid & gps. In *Emerging Technologies & Factory Automation, 2009. ETFA 2009. IEEE Conference on*, pages 1–6. IEEE, 2009.
- [2] S Kavitha, S Varuna, and R Ramya. A comparative analysis on linear regression and support vector regression. In *Green Engineering and Technologies (IC-GET), 2016 Online International Conference on*, pages 1–5. IEEE, 2016.
- [3] Min Li and Yang Xiao. Rfid-based tracking and tracing information platform of logistics. In *Conference Anthology, IEEE*, pages 1–4. IEEE, 2013.