

# **REQUIREMENT TRACKING MANAGEMENT SYSTEM**

**BY**

**Md Anisur Rahaman**

**ID: 162-15-7734**

This Report Presented in Partial Fulfillment of the Requirements for the  
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

**Md. Zahid Hasan**

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised By

**Ms. Moushumi Zaman Bonny**

Sr. Lecturer

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

**DHAKA, BANGLADESH**

**OCTOBER 2020**

## APPROVAL

This Project titled “**Requirement Tracking Management System**”, submitted by **Md. Anisur Rahaman**, ID No: **162-15-7734** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 7 October, 2020

### BOARD OF EXAMINERS



---

**Dr. Syed Akhter Hossain**  
**Professor and Head**

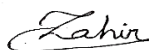
Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University



---

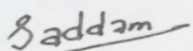
**Nazmun Nessa Moon**  
**Assistant Professor**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University



**Gazi Zahirul Islam**  
**Assistant Professor**

Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University



---

**Dr. Md. Saddam Hossain**  
**Assistant Professor**

Department of Computer Science and Engineering  
United International University

**Chairman**

**Internal Examiner**

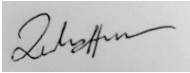
**Internal Examiner**

**External Examiner**

## DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Md. Jahid Hasan, Assistant Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

### Supervised by:

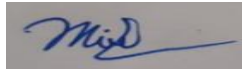


---

### **Md. Jahid Hasan**

Assistant Professor  
Department of CSE  
Daffodil International University

### Co-Supervised by:

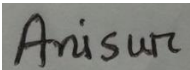


---

### **Ms. Moushumi Zaman Bonny**

Sr. Lecturer  
Department of CSE  
Daffodil International University

### Submitted by:



---

### **Md Anisur Rahaman**

ID: -162-15-7734  
Department of CSE  
Daffodil International University

## ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project successfully.

We really grateful and wish our profound our indebtedness to **Md. Zahid Hasan, Assistant Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Management System* ” to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Dr. Syed Akhter Hossain, Professor and Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

## **ABSTRACT**

The Software Requirements are description of features and functionalities of the target system. Requirements can express the expectation of the users from the software product. Requirements are definitely from client's point of view. The process of collecting requirements from client is called Requirement Engineering. Requirement analysis is an utmost process for the success of a software project. This is a process where requirements are being documented, actionable, measurable, testable, traceable, helps to identify business opportunities. Basically, requirement analysis involves all the tasks that are being conducted to identify the needs of different clients. Versioning of requirements is an important part of requirement engineering. Requirement's version is a process to record the changes of requirements and label the history with a version number, it is able to help the user to track the change of a requirements. From here he can compare the current requirements with newer requirements. In our country, most of the software firms use Microsoft word files to document or to keep track on the requirements. Sometimes they use handwritten papers also. This is not safe as well. The files can be missed or lost any time and this way is not helpful for comparison between older and newer requirements for version control. Version control is a system that records changes to a file or set of files over time so that we can recall the specific version later if we need. It enables multiple people to simultaneously work on a single project. Version control is important for code, files and assets that multiple team members will collaborate on. This will help a firm to keep track of changes, and keep every team member working off the latest version. So, according to these functionalities, we are proposing a management system that can work on requirement and requirement tracking.

## TABLE OF CONTENTS

<b>CONTENTS</b>	<b>PAGE</b>
Board of examiners.....	i
Declaration.....	ii
Acknowledgements.....	iii
Abstract.....	iv
<b>CHAPTER 1: INTRODUCTION</b>	<b>1-3</b>
1.1 Introduction.....	1
1.2 Motivation .....	1
1.3 Objectives .....	2
1.4 Report Layout.....	2
<b>CHAPTER 2: BACKGROUND</b>	<b>4-12</b>
2.1 Introduction.....	4
2.2 Background.....	6
2.3 Related work.....	12
2.4 Challenge.....	12

## **CHAPTER 3: REQUIREMENT SPECIFICATION**

3.1 Proposed System.....	13
3.2 Summary of Project Achievement.....	13
3.2.1 Working Procedure.....	13
3.2.2 Working Models and Diagrams.....	15
Activity Diagram.....	15
ER Diagram.....	17
Data flow Diagram.....	18
3.2.3 Tools.....	22

## **CHAPTER 4: DESIGN SPECIFICATION**

4.1 Front-end Design.....	23
4.2 Back-end Design.....	25
4.3 Interaction Design and UX.....	26

4.4 Implementation Requirements.....	27
<b>CHAPTER 5: IMPLEMENTATION AND TESTING</b>	<b>28-30</b>
3.2 Applications.....	28
5.3 Testing Implementation.....	29
	<b>31-31</b>
<b>CHAPTER 6: CONCLUSION AND FUTURE SCOPE</b>	
6.1 Future work.....	31
6.2 Conclusion .....	32
	<b>33</b>
<b>APPENDIX .....</b>	
	<b>34</b>
<b>REFERENCES.....</b>	
	<b>35</b>
<b>PLAGIARISM.....</b>	
Plagiarism.....	35



## LIST OF FIGURES

<b>FIGURES</b>	<b>PAGE NO</b>
Fig 2.1: Requirement specification types	4
Fig 2.2: Requirement Analysis	5
Fig 2.3: Requirement Tracing Attributes	5
Fig 2.4: Traceability Matrix Structure	7
Fig 3.2.2: Activity Diagram	16
Fig 3.2.2: Entity Relationship Diagram	17
Fig 3.2.2: Zero Level DFD	18
Fig 3.2.2: First Level DFD	20
Fig 3.2.2: Second Level DFD	21
4.1 Front-end Design	23
4.1 Front-end Design	24
4.1 Front-end Design	24
4.2 Back-end Design	25
4.2 Back-end Design	26

# CHAPTER 1

## INTRODUCTION

### 1.1 Introduction

As we live in a world where new tools and IT practices that can have full of our attention. Just as there is pressure to own the newest smart phone or car, we are always trying to keep up with the new technique in IT.

A **Requirement** is a capability to which a project outcome (product or service) should confirm or need to reach the goal.

**Requirement Traceability** refers that to the ability to describe and follow the lifetime of any requirement to reach the goal, in both forward and backward direction. This kind of traceability is called *Bidirectional Requirement Traceability*. The whole process can be said as *Requirement Tracking*.

**Requirements Management** is the process of documenting, analyzing, tracing, prioritizing and agreeing on requirements and controlling change and communicating to relevant stakeholders.

### 1.2 Motivation

The purpose of developing this project is to solve the problem of many IT company of our country which are doing so many projects. They need to keep track on them and requirements of that projects in order to develop the future version. Though whole list of requirements are needed to develop the future version and other activities of projects. This system can help the IT companies to keep track on the projects details along with all requirements in an automated way. They don't have to use Microsoft word or any other hand written paper for the projects.

For *software development*, change tracking and version handling is important. If the developer uses MS word, in that case change tracking is almost impossible. Only 'Manual tracking of change' has to be done, which can be entered through manual comments. Version Handling is an issue to follow in requirement management. Test cases with requirement id can't be maintain in MS word. There is a lot of manual work that is done that is done in that case of an MS word, this can not only impact the resource utilization but also time consuming.

### 1.3 Objectives

**Requirement Traceability Matrix (RTM)** is a document that maps and traces user's requirements with test cases. The main purpose of Requirement Traceability Matrix is to validate that all requirements are checked via test cases such that no functionality is unchecked during Software Testing.

Now the question is, how to make sure that any requirement is not left out of the testing cycle? Answer is, by a simple way is to trace the requirement with it's corresponding test cases. This is termed as, *Requirement Traceability Matrix*. This is typically a worksheet that contains the requirements with its all possible scenarios. This is helpful indeed for the testing team and manage the projects. [1]

### 1.4 Report Layout

#### **Chapter 1: Introduction**

Motivation, objectives and the expected outcome of the project have been discussed in this chapter. The report layout has been mentioned in the last part of this chapter.

#### **Chapter 2: Background**

Background circumstances of our project have been discussed here. Related work, comparing with other systems, the scope of the problem and challenges of the project are explained here.

### **Chapter 3: Requirement Specification**

This chapter discusses about the requirements. Business process modeling, the requirement collection and analysis, the use case model of the project with description, the logical relational database model and the design requirements are the topics of this chapter.

### **Chapter 4: Design Specification**

This chapter consists of all the designs of the project. These are Front-end design, back-end design, Interaction design and UX and the implementation requirements

### **Chapter 5: Implementation and Testing**

Implementation of database, front-end designs and interactions, testing implementation, test results and reports are the contents of this chapter.

### **Chapter 6: Conclusion and Future Scope**

It is the last chapter where conclusion and future scope of the project have been discussed.

## CHAPTER 2

### BACKGROUND

#### 2.1 Introduction

**Software Requirements specification** defines how the intended **software** interact with hardware, external interface, speed of operation, response time of system, portability of software across various platforms, maintainability, speed to recovery after crashing, Security, Quality, Limitations etc. [2]



Fig- 2.1 Specification types

It can be a document that captures complete description about how the system is expected to be executed and perform. It is actually signed off at the end of requirement engineering phase.

**Requirement Traceability** refers to the ability to describe and follow the life of a requirement in both forward and backward direction (i.e. from its origins, through its development and specification, to its subsequent deployment and use, and through all periods of on-going refinement and iteration in any of these phases).

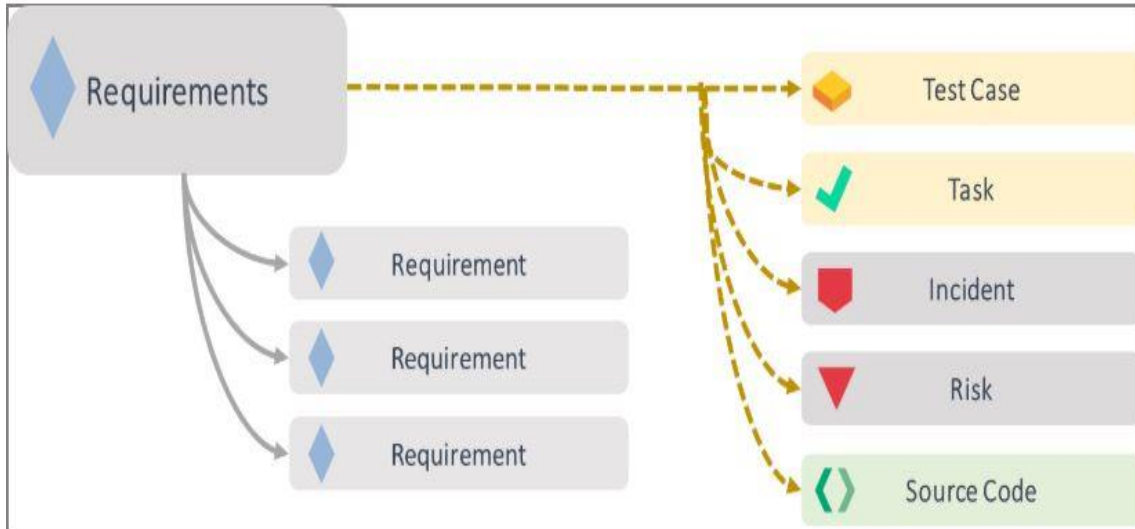


Fig -2.2 Requirement



Fig-2.2 Requirement Management

Performing a **Requirements Traceability Analysis** is an important part of the software engineering process as it ensures that all of the requirements have been adequately considered during each phase of the project, and that there aren't any scope 'holes' in the developed system due to missed requirements. The activity also ensures that all of the requirements are internally consistent with each other and support the overarching business drivers, goals and objectives.[1]

## 2.2 Background

For the success of a project, it is utmost **important** to analyze project requirement when they are gathered as well as throughout the lifecycle of the project. Requirement analysis helps to keep the requirements in line with the need to the business.

For completing each stage of a project, it must go through the requirements. It can help definitely to develop the future version of that project also. In that case, if a developer couldn't manage or track the sequence of requirements, there is a big chance to fall the project down. Traceability helps to measure a team's success effectively, by letting check it the most important business needs have been met. Traceability Matrices make it easy to track the requirements, codes, test cases and release plans for update version of the on-going projects.

This project will keep track on every single requirement and make sure that the requirement is fulfilled or not. A chief technical officer (CTO) can control the whole system with project details. Though **Storing Codes** is also a big part of requirement engineering, here is a good way to store every project's source code safe.

By using this **system**, there is no need to make any hand written document which is sometimes risky to keep secrecy about any project / requirement / traceability matrix. This management system ensures adequate documentation for further uses.

**Version Handling** is a perennial challenge when tracking delivery against a requirement. By the time, the development team subscribe to a requirement and put it through a delivery sprint, the Product Owner could have changed the requirement completely, for valid reason of course. So, the development team need to be able to look back at the end of the sprint and say with confidence that the change can be achieved Versioning can be a great assistance here. [3][4]

A good **TM** (*Traceability Matrix*) will let us check quickly whether all documentation is in place for a requirement – be it user stories, design documents, wireframes, technical specifications, test cases. ReQtest, for instance, allows us to attach

files to a requirement, which we can use to link design documents, wireframes, technical specifications etc.

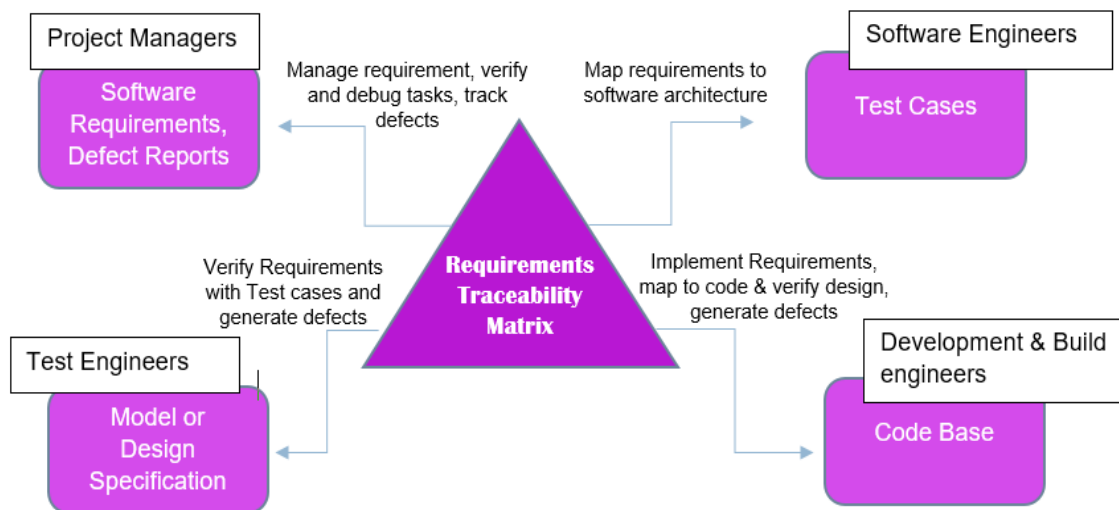


Fig-2.3 Traceability Matrix

We have studied some of requirement tracking / management systems. Such as,

RQM,



Reqheap,

Osrmt,

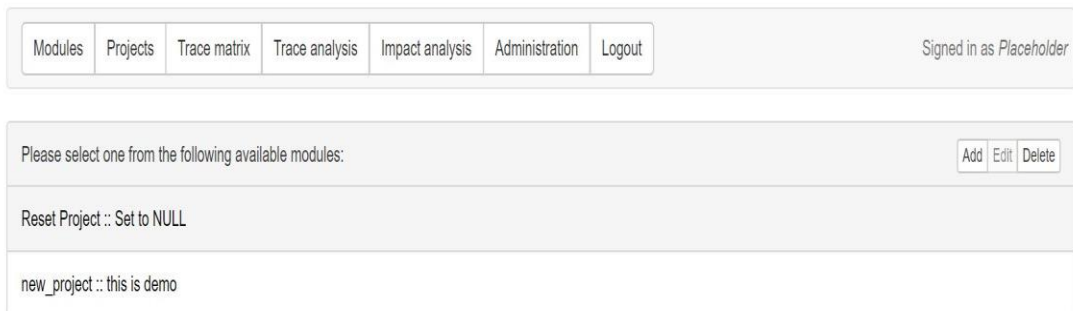
NagioSQL,

Rmtoo.

We have gathered some basic knowledge from these systems and intended to do something like that from the root.

Here are some snaps from the systems which we have studied –

**(i).**



The screenshot shows a web application interface. At the top, there is a navigation menu with buttons for 'Modules', 'Projects', 'Trace matrix', 'Trace analysis', 'Impact analysis', 'Administration', and 'Logout'. To the right of the menu, it says 'Signed in as Placeholder'. Below the menu, there is a section titled 'Please select one from the following available modules:' with 'Add', 'Edit', and 'Delete' buttons. Below this, there is a table with two rows of data:

Reset Project :: Set to NULL
new_project :: this is demo

Table 2.1 Placeholder

**(ii).**

Modules Projects Trace matrix Trace analysis Impact analysis Administration Logout Signed in as Placeholder

Gates Add Edit Delete

Credentials Add Edit Delete

Configuration settings

Table 2.2 Administration

(iii).

Fruit Snaccs Inc.

**Deliverables**

Create Task

Title	Due Date	Completion	Description
Deliverable Task 1	12/05/19	63%	Create Deliverable Display Page
Deliverable Task 2	12/10/19	23%	Create Deliverables Creation Page
Deliverable Task 3	12/30/19	90%	Create Deliverables Editing Page
Deliverable Task 4	12/01/19	57%	Creating A Single Deliverables Display Page
Deliverable Task 5	12/10/19	25%	Task Display for Deliverables

Table 2.3 Deliverables

(iv).

### Related Requirements

Title	Date Raised	Date Assigned	Severity	Priority	Description
Issues 1	6/10/19	12/05/19	High	High	Create Issues
Issues 2	6/10/19	12/01/19	Medium	Low	Create Issues Creation Page
Issues 3	6/10/19	12/30/19	Medium	Low	Create Issues Editing Page
Issues 4	6/10/19	12/01/19	High	Medium	Creating A Single Issues Display f
Issues 5	6/10/19	12/10/19	Medium	Low	Task Display for Issues

Table 2.4 Database

(v).

Business Requirement #	Technical Requirement #	Test Case ID
B1	T94	1
B2	T95	3
B3	T96	3
B4	T97	4

*Requirement Traceability Matrix*

Table 2.5 Database 2

(vii).

Req No	Req Desc	Testcase ID	Status
123	Login to the application	TC01,TC02,TC03	TC01-Pass TC02-Pass
345	Ticket Creation	TC04,TC05,TC06, TC07,TC08,TC09 TC010	TC04-Pass TC05-Pass TC06-Pass TC06-Fail TC07-No Run
456	Search Ticket	TC011,TC012, TC013,TC014	TC011-Pass TC012-Fail TC013-Pass TC014-No Run

Table 2.6 Database 3

(viii).

The screenshot shows a Google Sheet with the following data:

	A	B	C	D	E	F	G
1		Important Business Requirement 1	Important Business Requirement 2	Important Business Requirement 3	Important Business Requirement 4	Important Business Requirement 5	
2	System Requirement 1	X		X			
3	System Requirement 2		X				
4	System Requirement 3						
5	System Requirement 4		X				
6	System Requirement 5			X			
7	System Requirement 6				X		
8	System Requirement 7				X		
9	System Requirement 8				X		
10	System Requirement 9				X		
11	System Requirement 10				X		
12	System Requirement 11				X		
13							
14							
15							

Table 2.7 Database 4

From these open source management systems, we've generated some idea and wished to build a system based on Requirement Tracking & Management System, which can be more helpful for our country's developing farms and QA analyzer.

### **2.3 Related work**

There are some websites which provides the address of requirement management in Bangladesh . Some others have very limited features. Some platforms for finding requirement management in Bangladesh.

These websites are basically common platforms to find community center. They are not managed by any specific one community center.

### **2.4 Challenges**

Some challenges we have faced are mentioned below:

- Firebase Fire store database was a new tool for us. As we have used this database in our project, so we needed to learn a lot about this database.
- Beautiful design and unique features.
- Developing user friendly interface.
- Handling two parts, common users and admin in one app
- Ensuring data security
- Blocking unauthorized users from reading, writing and deleting data from database.

## **CHAPTER 3**

### **REQUIREMENT SPECIFICATION**

#### **3.1 Proposed System**

From our background studies and knowledge, we are proposing a system that work on, “Requirement Management & Requirement Traceability Matrix”. From the login management to matrix management, it can handle all related thing with projects.

It can also handle –

Login management of system users.

Management of role of System users Management traceability of requirements and matrix.

- Test cases handling.
- Economy handling.

#### **3.2 Summary of Project Achievement**

We have made a very simple and user-friendly management system that can be operated by its system users, also very much helpful to the requirement analysis.

##### **3.2.1 Working Procedure**

There are some stages in this system. Here they are –

- Sign IN / UP

- Different view for different system user
- Storing project details
- Requirement gathering
- Matrix manipulation

## **Sign IN / UP**

From the very beginning of using this system, the system has a registration process, which can be called as ‘Sign IN / UP’ procedure. By this system, an employee is going to be registered as a System User. In this part, user need to fill some form –

### For Sign IN –

1. Username
2. Password
3. User Type

### For Sign UP –

- |             |              |
|-------------|--------------|
| 1. Name     | 4. Phone No. |
| 2. Username | 5. Password  |
| 3. Email    | 6. User Type |

## **Different view for different system user**

The system is divided into some particular parts. Such as, if a Chief Technical Officer Sign Into this system, he / she can see the whole related features with traceability matrix. Project allocation, requirement gathering and matrix manipulation handling.

If others system users, like project manager, team leader and software engineer signed in here, there are something different interface would come with some different features and shortage of technical procedure. For example, a project manager can divide the tasks and add any new project. But software engineers are not able to add new task here. [5]

## **Storing Project**

Related things are being stored in an automated way through this system. Related requirements, the system user working details with the particular project, requirement analysis and matrix handling are stored through this system.

## **Requirement Gathering**

Gathering requirement is an easy process through this management system. According to project id, list of requirements will be stored in database.

## **Matrix Manipulation**

A Requirement Traceability Matrix (RTM) can be generated here automatically, which can carry project id, requirement id, requirements description, source code storage link and assigned employee. This can help the developer team for further uses.

### **3.2.2 Working Model & Diagrams**

#### **Activity Diagram**



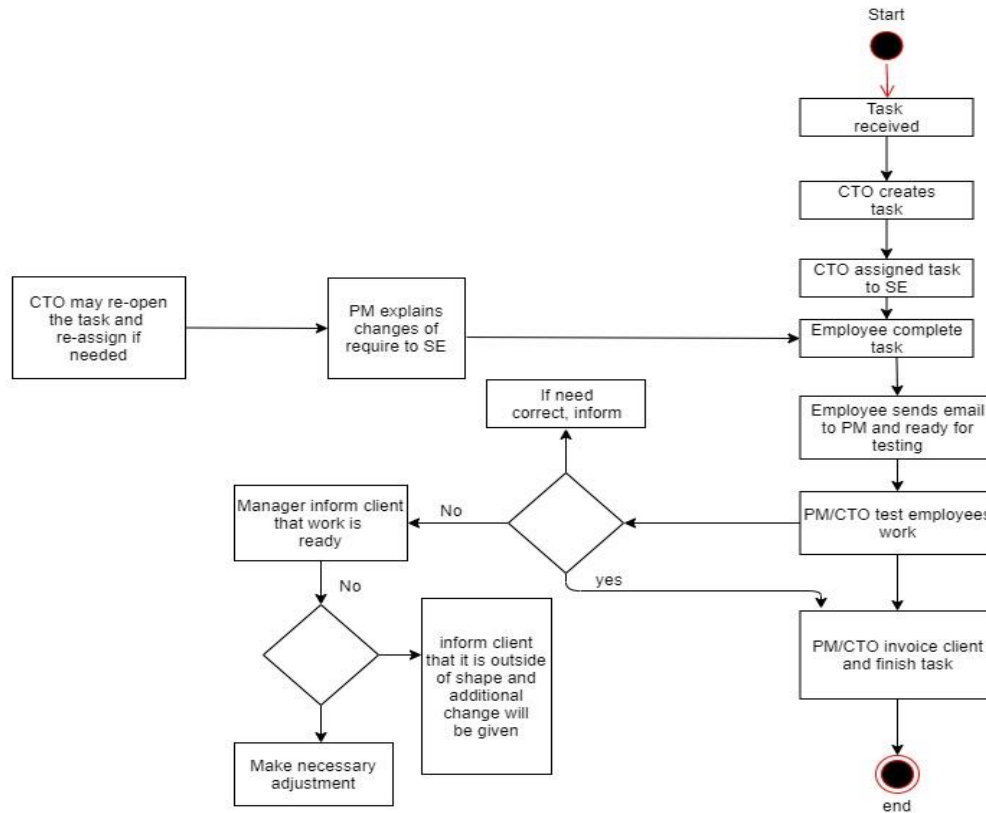


Fig-3.2.2 Activity Diagram

An ‘Activity Diagram’ can demonstrate the logic of an algorithm / system. From the starting to the end of the whole process, this diagram can express that how the system works. It is like a model of the target software which carries architecture elements, such as method, function, operation etc. There can be so many steps, such that receiving task, allocation or assigning, re-opening or re-assigning, project manager or controller can test the completed project with test cases.

If there need to change anything of related projects, the controller or project manager can change it and explains it to software engineers. After testing and checking projects in some stages, project manager can inform the client that the work is ready. [6]

After that, if the work has any problem, necessary adjustment or additional change will be given. At last, Controller or Project Manager will invoice the client and finish the task.

## Entity Relationship (ER) Diagram

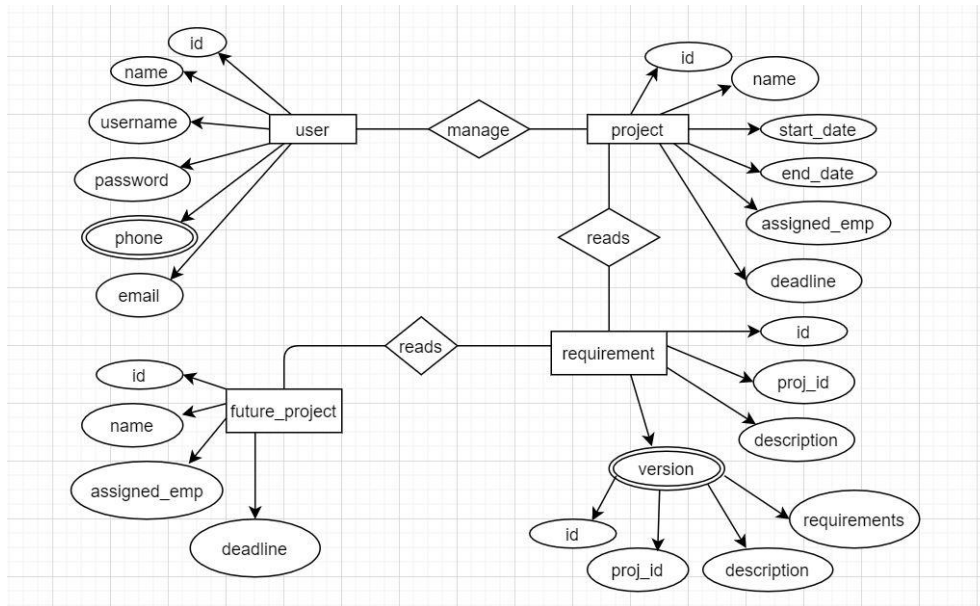


Fig-3.2.2 Entity Relationship (ER) Diagram

Here is the 'Entity Relationship' diagram to express the structure of the database. There can be some tables of system users, running projects, requirements, future projects and version. Each of tables has particular entity to store project information.

These tables are being used to project manager, team leader and software engineer. The controller would have some extra tables to have a better control over the system.

A good visual representation of the system is expressed here. User, Project, Requirement, Version, Future Project, these tables are able to represent of the logical structure of the database and relationship between this table entities.

Here the table 'user' contains the data of the system users. A table, 'project' contains the details of a project including the id number. The relation between USER and PROJECT is 'manage'. It means, an employee / system user can manage the information about projects and can forward to the next stage.

Every project must have list of requirements. So, the tables 'project' and 'requirements' can be managed by 'read' relation. Each requirement would have particular id, related project id, description of the requirement and which version it is running to.

A table named version can be separated here including a particular id, project id, description and list of requirements.

Another table for future project, is the list of upcoming projects connected requirement table also. This can read requirements too. This is a basic structure diagram for the system.

### Data Flow Diagram(DFD)

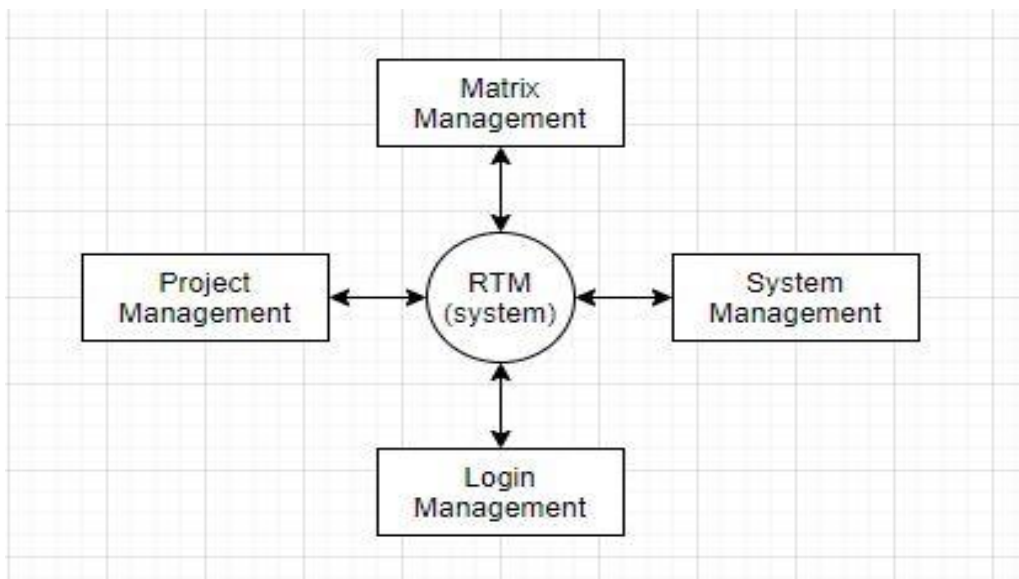


Fig-3.2.2 Zero Level DFD

Data flow diagram is the way of representing Flow of Data of a process or an information system. Zero Level Data Flow Diagram expresses the basic and fundamental job of a system. From this Zero Level DFD, it expresses that our system basically works with these four tasks. They are,

- ✓ Login Management – This can handle the multi user login into this system. Though here are various types of system user.
- ✓ System Management – From a task receiving to task delivery, this whole process can be managed by this system.
- ✓ Project Management – Handling running projects, pending projects, future project that has to be done.
- ✓ Matrix Management – Project related information can be stored by a matrix, that can be said ‘Requirement Traceability Matrix’. This matrix consist of project id, related requirement id, the link where the code is stored and employee under whom the project is done.

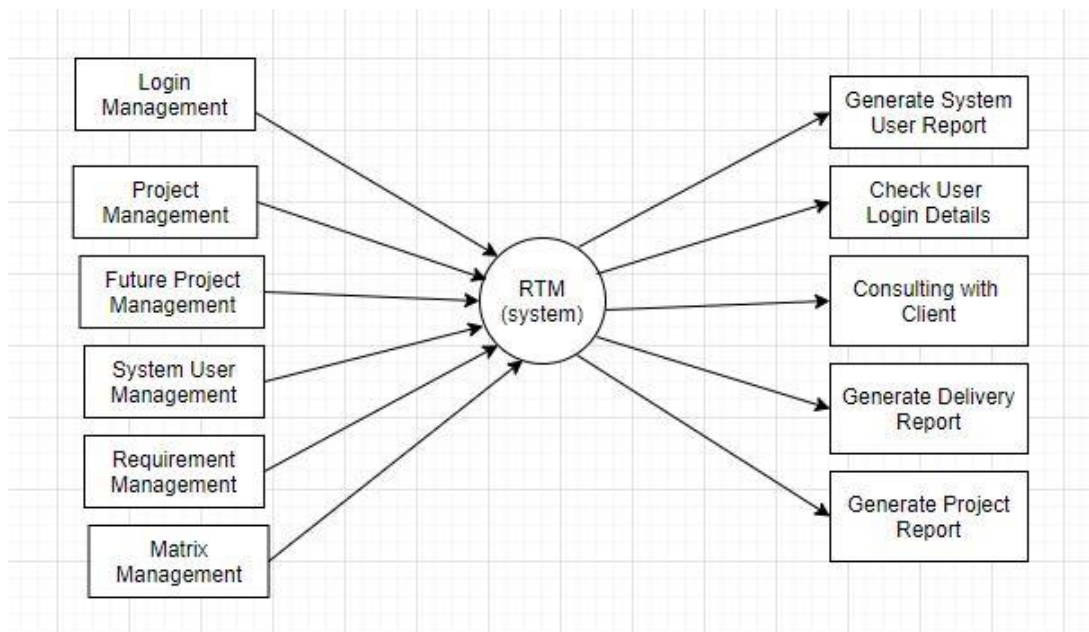
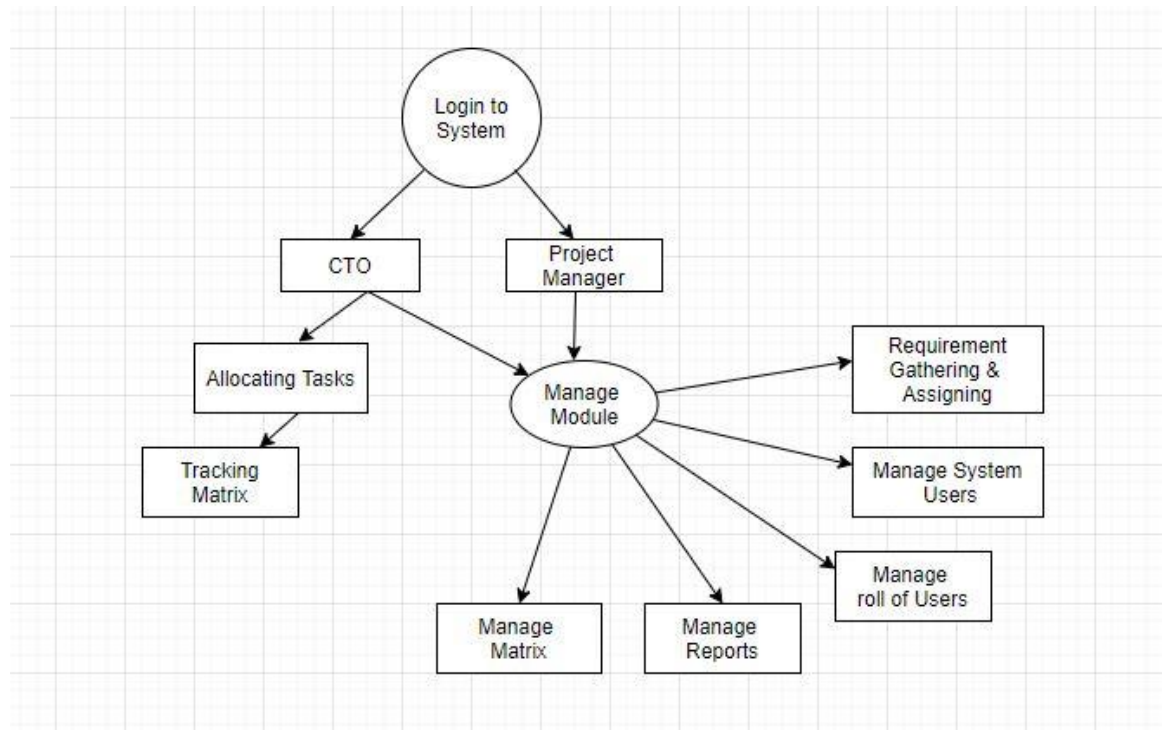


Fig-3.2.2 First Level DFD

First Level Data Flow Diagram is an extension of zero level data flow diagram. It expresses that the total tasks of the project or the system. This First Level DFD consist of some management, checking & generating tools.

These management and generating tools are –

1. Login Management
  2. Project Management
  3. Future Project Management
  4. System User Management
  5. Requirement Management
  6. Matrix Management
  7. Generate System User Reports
  8. Check User Login Details
  9. Consulting with Client
  10. Generating Delivery Report
  11. Generate Project Report
7. These tools are connected each other.
8. Controller has access to all of these tools and functions.



9. Fig-3.2.2 Second Level DFD

10. Second Level DFD expresses the full process of the system. From login to the system, two different interfaces will come onto the screen. The controller, who can have the access to all these related functions. The Project Manager / Team Leader / Employee can have some limitations on their task and responsibilities.
11. Project Allocation, related requirement gathering, manage roll of system users and manage matrix, the controller can handle all these functionalities.
12. The project manager and other system users can have some manage modules. Requirement Gathering & Assigning, Manage matrixes and reports etc.

### 3.2.3 Tools

We have used some developing tools here to develop this system. They are –

- Language – HTML5, CSS3, PHP, js, ajax and some other languages.
- Server – XAMPP
- IDE – Sublime Text

# CHAPTER 4

## DESIGN SPECIFICATION

### 4.1 Front-end Design



Fig: 4.1 Home page



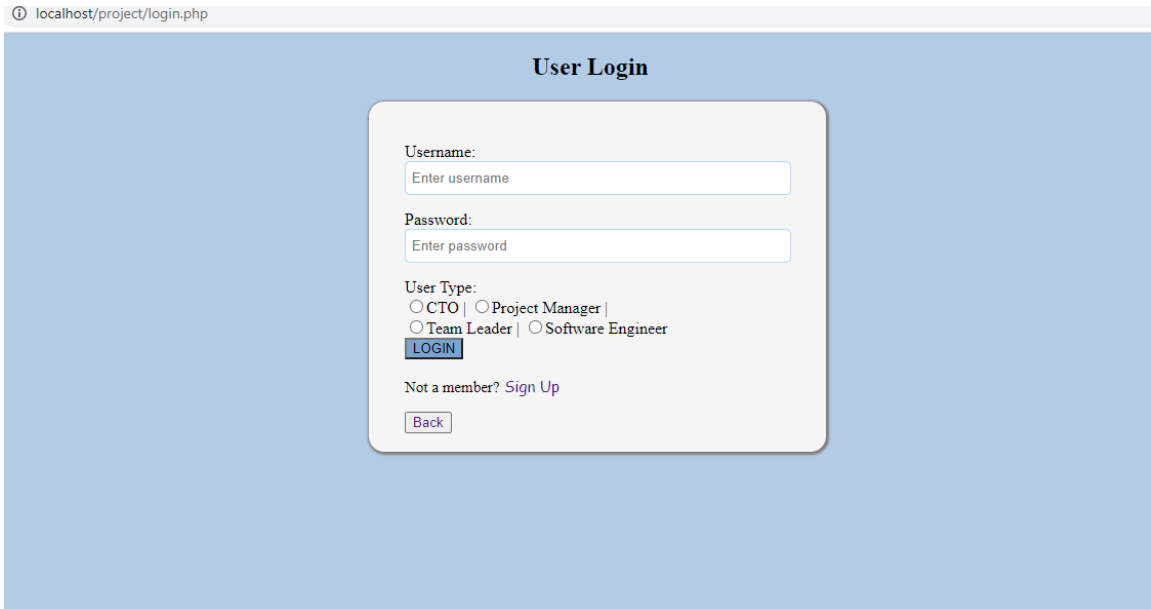


Fig: 4.2 Log In Page

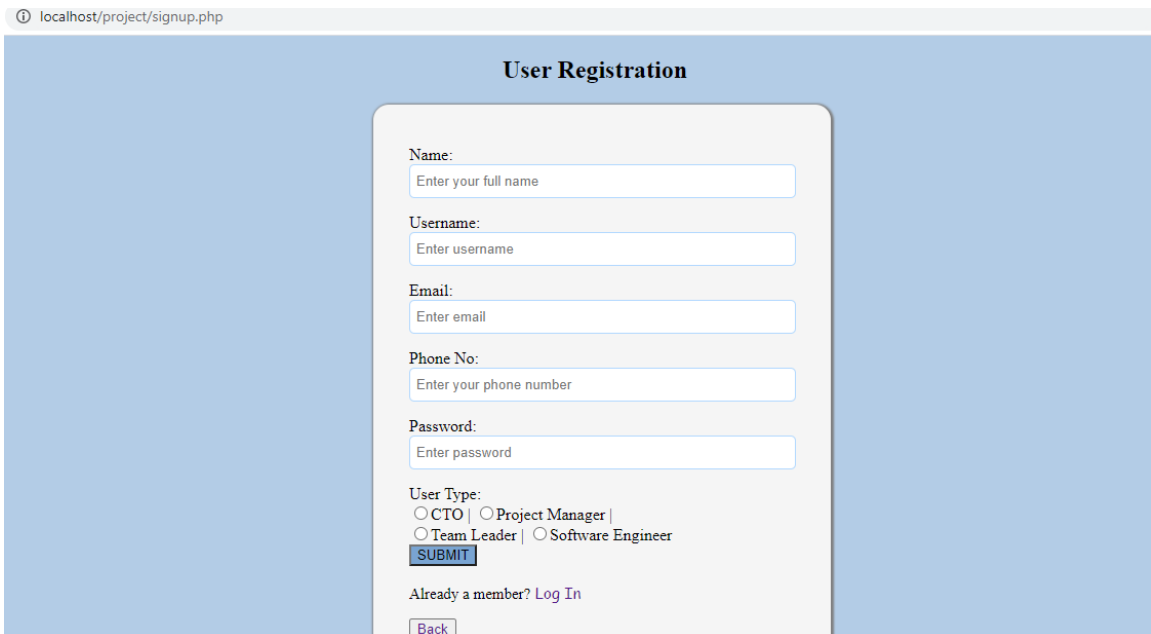
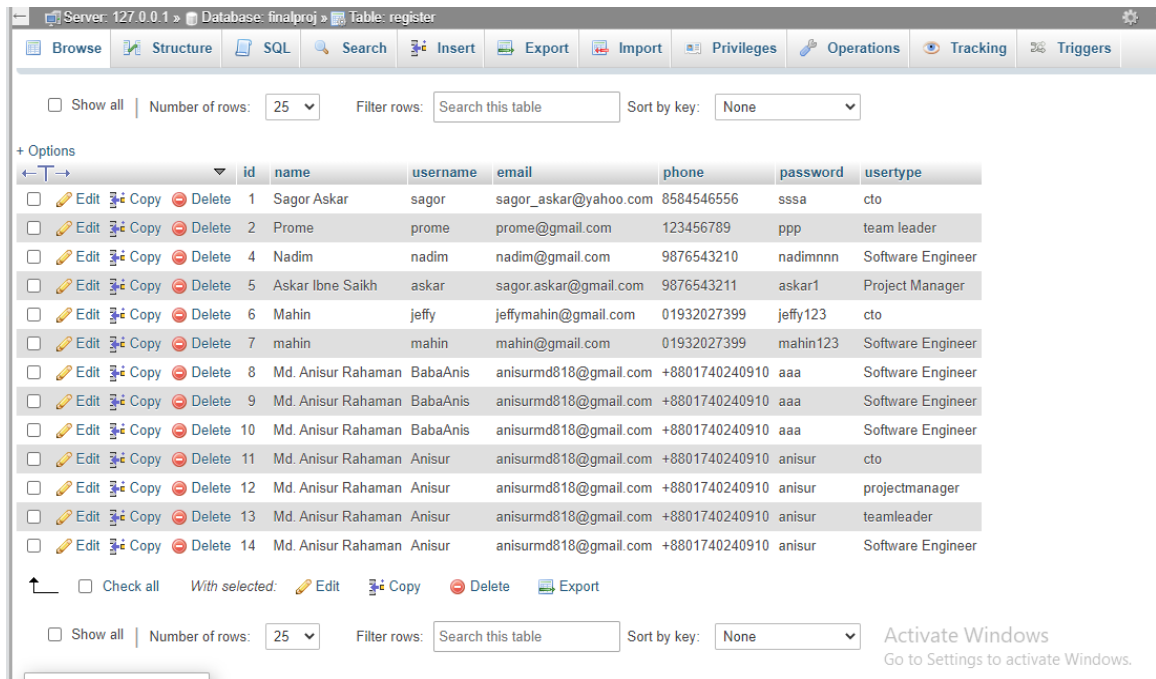


Fig: 4.3 User Registration Page

## 4.2 Back-end Design



Server: 127.0.0.1 » Database: finalproj » Table: register

Browser Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

	id	name	username	email	phone	password	usertype
<input type="checkbox"/>	1	Sagor Askar	sagor	sagor_askar@yahoo.com	8584546556	sssa	cto
<input type="checkbox"/>	2	Prome	prome	prome@gmail.com	123456789	ppp	team leader
<input type="checkbox"/>	4	Nadim	nadim	nadim@gmail.com	9876543210	nadimnn	Software Engineer
<input type="checkbox"/>	5	Askar Ibne Saikh	askar	sagor_askar@gmail.com	9876543211	askar1	Project Manager
<input type="checkbox"/>	6	Mahin	jeffy	jeffymahin@gmail.com	01932027399	jeffy123	cto
<input type="checkbox"/>	7	mahin	mahin	mahin@gmail.com	01932027399	mahin123	Software Engineer
<input type="checkbox"/>	8	Md. Anisur Rahaman	BabaAnis	anisurmd818@gmail.com	+8801740240910	aaa	Software Engineer
<input type="checkbox"/>	9	Md. Anisur Rahaman	BabaAnis	anisurmd818@gmail.com	+8801740240910	aaa	Software Engineer
<input type="checkbox"/>	10	Md. Anisur Rahaman	BabaAnis	anisurmd818@gmail.com	+8801740240910	aaa	Software Engineer
<input type="checkbox"/>	11	Md. Anisur Rahaman	Anisur	anisurmd818@gmail.com	+8801740240910	anisur	cto
<input type="checkbox"/>	12	Md. Anisur Rahaman	Anisur	anisurmd818@gmail.com	+8801740240910	anisur	projectmanager
<input type="checkbox"/>	13	Md. Anisur Rahaman	Anisur	anisurmd818@gmail.com	+8801740240910	anisur	teamleader
<input type="checkbox"/>	14	Md. Anisur Rahaman	Anisur	anisurmd818@gmail.com	+8801740240910	anisur	Software Engineer

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Activate Windows  
Go to Settings to activate Windows.

Fig: 4.4 Username List

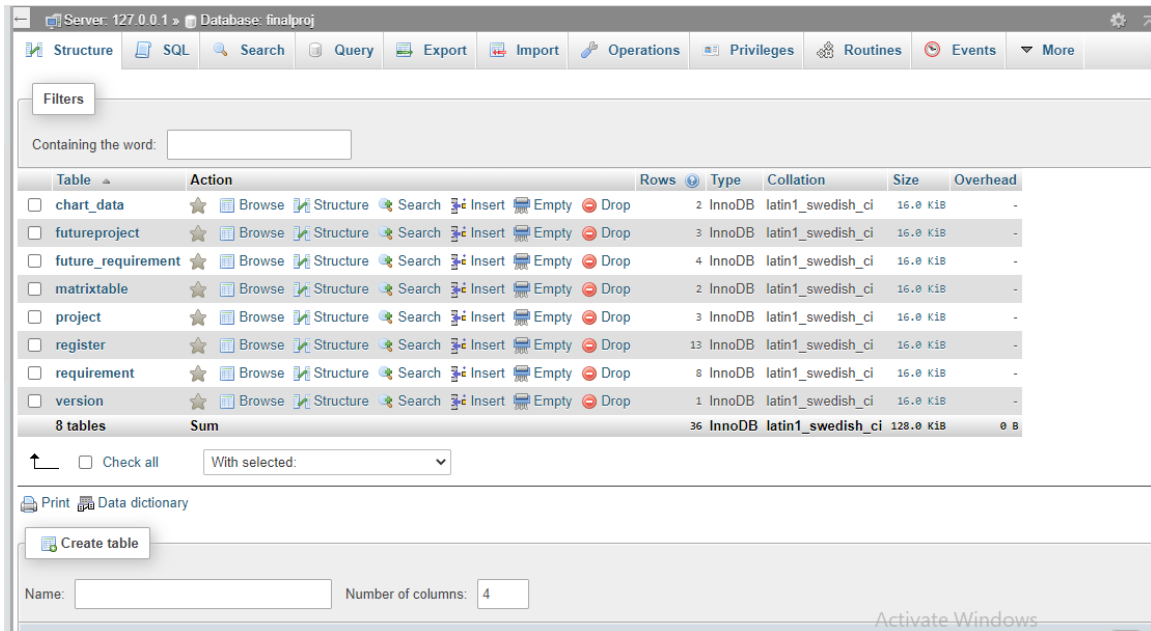


Fig: 4.5 Database List

### 4.3 Interaction Design and UX

Sequence diagrams represent objects participating horizontally and vertically in interactions.

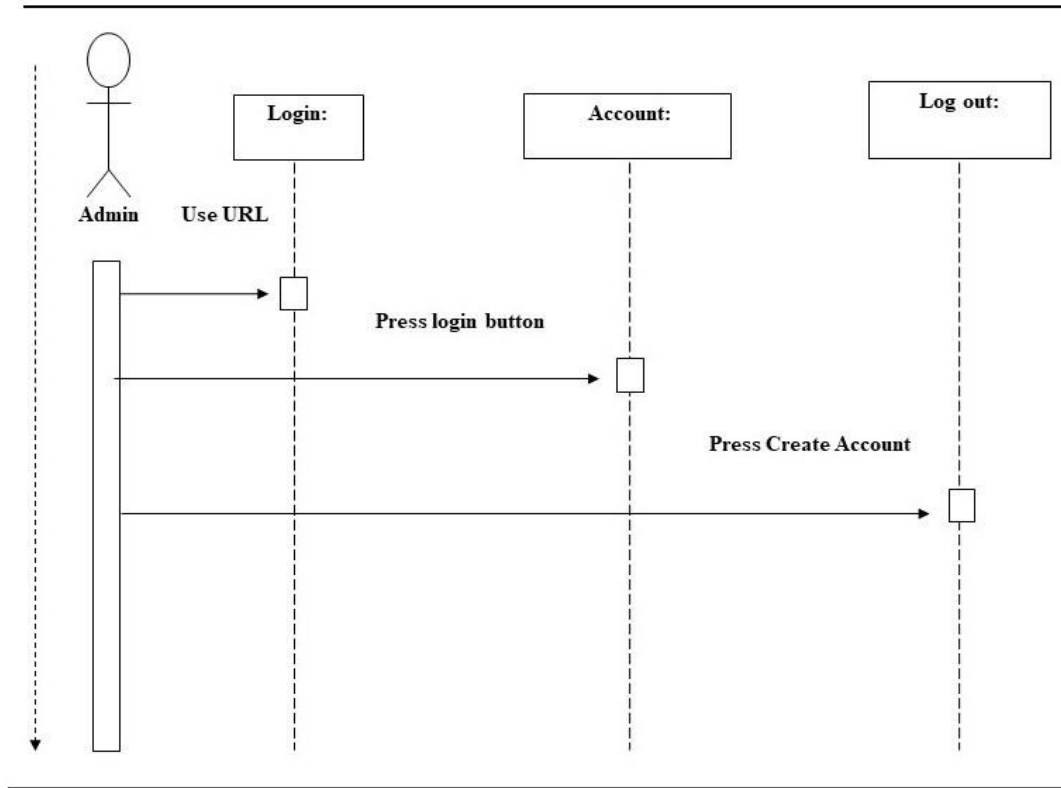


Figure 4.6: Admin Login Interaction Design

## 4.4 Implementation Requirements

**Purpose:** The common things for Equip this documents is to made a generally insights into the analysts and Necessity for this Subsist systems or situations and for determined the operations Specialty of the whole systems.

**Scopes:** The common Documents are plays a Necessary Introduction in the developmental process life cycle and its impact to describe the completed Necessity from the same system. It is Intended for users by the developer and will be the basics concept during the testing phases. Any kind of changes made to the Necessity include the futures will have to go with the help of formal change approvals process.

## Chapter 5

### IMPLEMENTATION AND TESTING

#### 5.1 Application

In a software development project, Requirement Traceability Matrix (RTM) is a document which is used to validate that all the requirements are being tested or covered in testing phase. It a tabular format as it holds multiple relationships between requirements are project details.

The main agenda of every tester should be to understand the client's requirement and make sure that the output product should be defect-free and user friendly. To achieve this goal, every QA should understand the requirement thoroughly and create positive and negative test cases.

The traceability matrix is typically a worksheet that contains the requirements with all possible test cases and their current state, i.e. if they have been passed or failed. This would help the testing team to understand the level of testing activities done for the specific product.

This system is applicable for our country, where very few developing teams are using this kind of management systems. It will help the developer team to go through according to the needs of the client and make them sure that all requirements are being tested. For this system, there is no need to put this data in any other format, like Microsoft word or hand notes. This management system can keep a good track on every single information of the project and related requirements which are being tested.

## 5.2 Testing Implementation

This system is also can be applicable to track all requirements and whether or not they are being met by the process and design of the running project, also assist in the creation in the creation of the Project Plan Tasks, Deliverable Documents etc. Though it holds the list of requirements, it can define what the client wants. In order to change anything, it can analyze the impact of change.

It can resolve the conflicting issues by talking to the stakeholders, in order to change anything based of priority of the requirements. By this system, less development rework and higher level of satisfaction from stakeholders can come. So, this can be a good option for project development procedure.

It can be application for the QA team as it helps in analyzing the impact on the QA team's work with respect to re-working on the test cases if there is any defect. This system can show overall execution status with a focus on the requirements. It can do for future project also.

This system can manage the solution scope for future projects or version handling of the running or completed projects. If any change is required, it can evaluate faster. It can reduce project risk. It means, proper handling of related issues will help the team to make the project completed.

Though this proposed system will work with requirements, it can help to promote the consistence between requirements and also help to see at a glance that what works has been completed and what are pending.

It can deal with every single requirement of the project, and make sure that the work has done or not. It means, no requirements are being overlooked during design and implementation.

Though this system handles login management, system management, project management and also matrix management, this can be a good option for a developing team or QA team.

## Chapter 6

### CONCLUSION AND FUTURE SCOPE

#### 6.1 Future Work

We have completed this project with the base of Requirement Management System. In future, we are going to work on these topics –

More User Friendly

Develop the Matrix

Data Security

We are working to make this system more user friendly. There are three different view after login / registration process. One is for controller, one is for project manager and team leaders and other one is for software engineers. All necessary actions can be done by this procedure. We will work to make it more smooth and reliable, also perfect according to the needs. It is a management system also. So we will make it hundred percent perfect and eligible to take care of the projects and system users.

Though the main feature of this system is to make a perfect *Traceability Matrix*, which can contain whole related functionalities of the project and requirements. We will work it to make it more strengthen and secured. If so, this can be more flexible for the users. It will help the users to find all related information about any project, and make sure that every single requirement are tested with case that make the project complete. In future we will work on it.



## 6.2 Conclusion

We will develop the data security system. The term data encryption, Password encryption and some other security can be added in future. Though we know, requirements are concerned with that the system should do and security requirements are concerned with what the system should not do. For further development of this system, we will keep it in mind too that which requirement should be prioritized.

In this is system, security controls are perceived to limit functionality or interfere with usability. Another thing is that, the users who help define the system are not typically the abusers from whom the system must be protected. We are following this rules here. Space for expressing and perceiving requirement analysis are being increased in future.

## **APPENDIX**

We began our journey to build this project from Spring 2020. For better understanding, we did a lot of research on requirement in Bangladesh. We collected information from many websites. Then we took decision to use XAMPP as our database. For this, we need edto study a lot on this topic as XAMPP was a very new database system for us.. We've tried to build a user-friendly requirement tracking management system. It can also be a communication medium between users and authority' officials. I hope this application will be very helpful for the users. We will keep upgrading our system adding new facilities and features.

## REFERENCE

1. RQM - <https://github.com/AdrianSimionov/RQM>, last accessed on 18 Mar, 2020.
2. Software Requirement Management System - <https://github.com/besher-alrez/Software-Requirement-Management-System> , last accessed on 25 Mar, 2020.
3. NagioSql - <https://github.com/LUCKFBO/nagiosql>, last accessed on 01 Apr, 2020.
4. An effective Requirement Engineering Process Model for Software Development & Requirement Management - [https://www.researchgate.net/publication/224202533\\_An\\_Effective\\_Requirement\\_Engineering\\_Process\\_Model\\_for\\_Software\\_Development\\_and\\_Requirements\\_Management](https://www.researchgate.net/publication/224202533_An_Effective_Requirement_Engineering_Process_Model_for_Software_Development_and_Requirements_Management), last accessed on 10 Apr, 2020.
5. Analysis of Requirement Engineering Processes, Tools / Techniques and Methodologies - [https://shop.tarjomeplus.com/UploadFileEn/TPLUS\\_EN\\_3992.pdf](https://shop.tarjomeplus.com/UploadFileEn/TPLUS_EN_3992.pdf), last accessed on 10 June, 2020
6. Successful requirement elicitation by combining requirement engineering techniques - <https://ieeexplore.ieee.org/abstract/document/4664355>, last accessed on 18 May, 2020
7. Use case diagram tutorial, <https://creatly.com/blog/diagrams/use-case-diagram-tutorial/amp/> , last accessed on 14 May, 2020

# PLAGIARISM

