Independent University **Bangladesh (IUB)**

IUB Academic Repository

Internship Reports Summer 2023

2023-10-05

An Undergraduate Internship/Project on QuantumSync CRM

Rafi, Mahabub Hasan

Independent University, Bangladesh

https://ar.iub.edu.bd/handle/11348/678

Downloaded from IUB Academic Repository



An Undergraduate Internship/Project on QuantumSync CRM

By

Mahabub Hasan Rafi

Student ID: **1830235**

Summer, 2023

Supervisor:
Mr. Sajed Imtenanul Haque

Lecturer

Department of Computer Science & Engineering

Independent University, Bangladesh

October 05, 2023

Dissertation submitted in partial fulfillment for the degree of
Bachelor of Science in Computer Science

Department of Computer Science & Engineering

Independent University, Bangladesh



An Undergraduate Internship/Project on QuantumSync CRM

By

Mahabub Hasan Rafi

Student ID: **1830235**

Summer, 2023

Consent from Supervisor

The student modified the internship final report as per the recommendations made by his/her academic supervisor and/or panel members during and/or before final viva, and the department can use this version for archiving as well as the OBE course material for CSE499.

This internship report is checked with Turnitin and plagiarism checker, and the score is:

Turnitin Score (%): 14%

Sajed

Mr. Sajed Imtenanul Haque

Lecturer | CSE

Department of Computer Science & Engineering

Independent University, Bangladesh

Attestation

I, Mahabub Hasan Rafi, affirm that this report has been diligently prepared by me, as an essential component of the fulfillment process to acquire the Computer Science and Engineering Degree from Independent University, Bangladesh.
Under the direction of Mr. Sajed Imtenanul Haque, the report has been completed. I vouch for the accuracy of all the work given here, which is based on the expertise I gained at my internship. If any information required, contact my organizational supervisor, Mr. MD Manzurul Karim, from Fronture Technologies Limited.

Mahabub Hasan 18.10.2023

Signature Date

Mahabub Hasan Rafi

Name

Acknowledgement

I would like to start by expressing my utmost gratitude to the Almighty Allah for guiding me to this far with his blessings and mercy. Working as an intern at Fronture Technologies Ltd. has been a great opportunity.

The experienced professionals at Fronture Technologies have provided me with unwavering support and motivation throughout my journey, making it an enriching experience. I want to express my gratitude to everyone on the Fronture Technologies team. This report has been greatly influenced by their readiness to impart their knowledge and valuable time.

I want to express my gratitude to Mr. Sajed Imtenanul Haque, my academic supervisor at Independent University in Bangladesh's Department of Computer Science and Engineering, for his invaluable advice, support, and unwavering encouragement throughout my internship and report completion.

Last but not least, I want to acknowledge the sacrifices, well wishes, and unending support from my respected faculty members, family, relatives, and friends. Their moral support, insightful advice, and encouragement have been a constant source of inspiration.

Letter of Transmittal

October 05, 2023
Mr. Sajed Imtenanul Haque
Lecturer
Department of Computer Science and Engineering
Independent University, Bangladesh

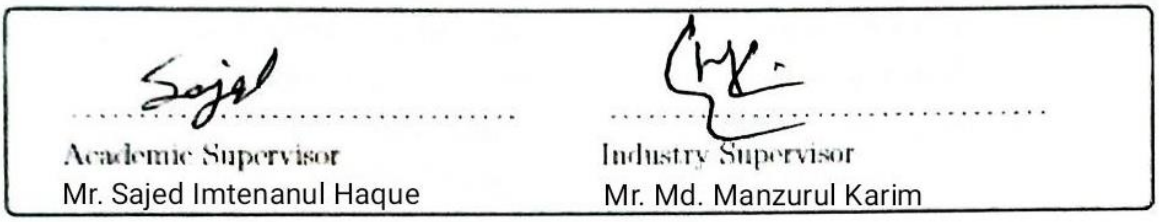
Subject: Letter of Submission for Internship Report, Summer-2023

Dear Sir,
With due respect to state that, I, Mahabub Hasan Rafi, from the Internship Program Summer – 2023, would like to submit my internship report. I'm pleased to inform you that my internship program and the associated report have been completed successfully. From July 3, 2023, until the present, I worked as an intern at Fronture Technologies Limited.
The experiences and tasks I undertook during my internship at Fronture Technologies are reflected in this report. My main goal during this time was to expose myself to the company's recommended technology fields. This included learning about software development methods and techniques, documentation, and research and development. Instead of merely concentrating on the finished result, I tried to comprehend the software development process. During my time at Fronture Technologies Limited, I had the opportunity to acquire and apply new skills and technologies. The company houses a small team of skilled software professionals who actively learn, collaborate and innovate together.
I extend my heartfelt gratitude to you for your unwavering guidance and support throughout this journey. I sincerely hope that this report satisfies all standards and meets expectations.

Sincerely,
Mahabub Hasan
ID: 1830235
Email: 1830235@.iub.edu.bd

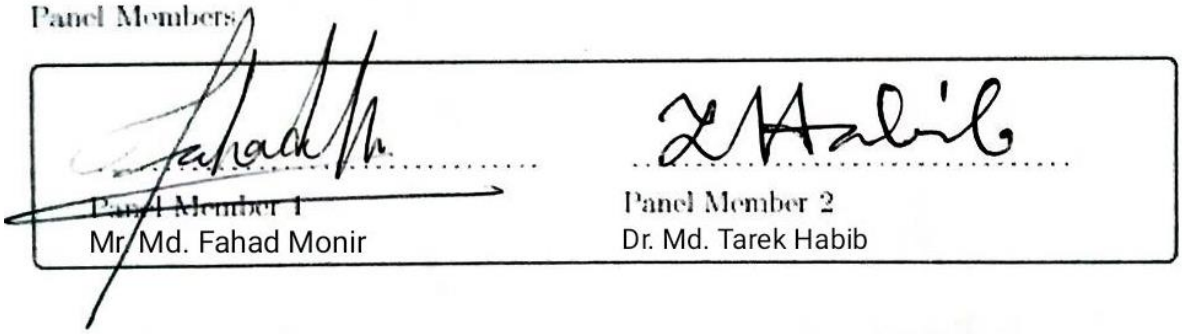
Evaluation Committee

Supervision Panel



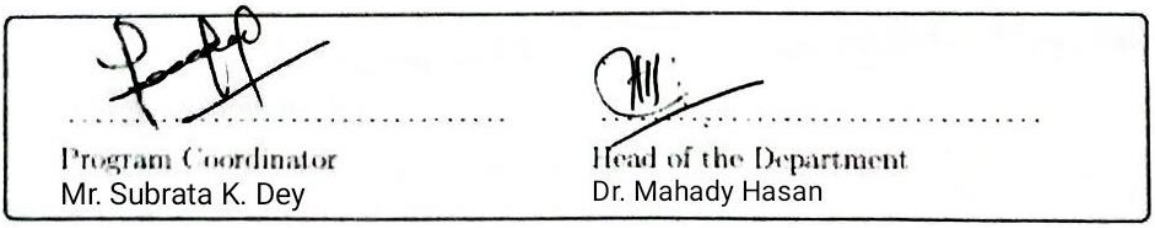
Academic Supervisor	Industry Supervisor
Mr. Sajed Imtenanul Haque	Mr. Md. Manzurul Karim

Panel Members



Panel Member 1	Panel Member 2
Mr. Md. Fahad Monir	Dr. Md. Tarek Habib

Office Use



Program Coordinator	Head of the Department
Mr. Subrata K. Dey	Dr. Mahady Hasan

Abstract

A Customer Relationship Management (CRM) system is a software solution that enables businesses to manage interactions with their customer, streamline processes, and improve customer engagement. To run a business efficiently, CRM is very good software solution. CRM provides enhanced customer relationship, improved customer service, efficient sales processes, shop management, effective marketing campaigns, better communication, data-driven decision making, increased sales and revenue, workflow automation, cross-selling and upselling, scalability, customer insights etc. Overall, a well-implemented CRM system contributes to building strong customer relationships, optimizing business processes, and increasing overall efficiency and profitability. It's a useful tool that equips companies to deliver outstanding client experiences and maintain competitiveness in today's dynamic market environment. There are many CRM systems available in the market ex. Salesforce Sales Cloud, HubSpot Sales Hub, ClickUp etc. But these systems do not fully cover the business requirements of a business operation. And some of the business organizations do not want to use other software which have subscription fees monthly or yearly. They want their own system. Considering the situation stated above, we have built a custom CRM system for an electronics devices and components selling and supply service. Just like mentioned earlier the company did not find the perfect CRM for their business in the market. They wanted something very easy to understand and operate. Not something that is overwhelmingly congested with features. So, we have built an easy to operate CRM for the company using React, Express and MongoDB. Discussion about the system will be stated in the coming section of this report.

Keywords: CRM, React, Express, MongoDB,

Contents

Attestation i

Acknowledgement ii

Letter of Transmittal iii

Evaluation Committee iv

Abstract v

1 **Introduction** 1

1.1 Overview/Background of the Work 1

1.2 Objectives 1

1.3 Scopes 2

2 **Literature Review** 3

2.1 Relationship with Undergraduate Studies 3

2.2 Related works 4

3 Project Management & Financing 5

3.1 Work Breakdown Structure 5

3.2 Process/Activity wise Time Distribution 6

3.3 Gantt Chart 6

3.4 Process/Activity wise Resource Allocation 6

3.5 Estimated Costing 7

4 **Methodology** 8

5 Body of the Project 9

5.1 Work Description 9

5.2 Requirement Analysis 9

5.3 System Analysis 15

5.3.1 Six Element Analysis 15

5.3.2 Feasibility Analysis 16

5.3.3 Problem Solution Analysis 16

5.3.4 Effect and Constraints Analysis 18

5.4 System Design 19
5.5 Implementation 22
5.6 Testing 27

6 **Results & Analysis** 29

7 **Project as Engineering Problem Analysis** 31
7.1 Sustainability of the Project/Work 31
7.2 Social and Environmental Effects and Analysis 31
7.3 Addressing Ethics and Ethical Issues 32

8 **Lesson Learned** 33
8.1 Problems Faced During this Period 33
8.2 Solution of those Problems 33

9 **Future Work & Conclusion** 34
9.1 Future Works 34
9.2 Conclusion 34

Bibliography 35

List of Figures

Figure 1: SalesForce Sales Cloud 4
Figure 2: ClickUp 4
Figure 3: WBS of QuantumSync 5
Figure 4: QuantumSync Gantt Chart 6
Figure 5: Rich Picture of QuantumSync 9
Figure 6: QuantumSync Use Case Diagram 19
Figure 7: Super Admin Activity Diagram 20
Figure 8: System Architecture 21
Figure 9: Login Page 22
Figure 10: Sign-up Page 22
Figure 11: Add Customer Modal 23
Figure 12: Place Order Modal 23
Figure 13: Add Supplier Modal 23
Figure 14: Add Product Modal 24
Figure 15: Add Product Category Modal 24
Figure 16: Add Product Type Modal 24
Figure 17: Super Admin Dashboard 25
Figure 18: Customer Table 25
Figure 19: Order Table 25
Figure 20: Supplier Table 26
Figure 21: Product, Product Category, Product Type Table 26
Figure 1: SalesForce Sales Cloud

List of Tables

Table 1: Project Time Allocation....6
Table 2: Resource Allocation Table....6
Table 3: Functional Requirement 1 - Sign Up....10
Table 4: Functional Requirement 2 – Login....10
Table 5: Functional Requirement 3 - Reset Password....10
Table 6: Functional Requirement 4 - Add Customer....10
Table 7: Functional Requirement 5 - Update Customer....11
Table 8:Functional Requirement 6 - Delete Customer....11
Table 9: Functional Requirement 7 - Order from customer....11
Table 10: Functional Requirement 8 - Add Supplier....12
Table 11: Functional Requirement 9 - Update Supplier....12
Table 12: Functional Requirement 10 - Delete Supplier....12
Table 13: Functional Requirement 11 - Order via email....13
Table 14: Functional Requirement 12 - Add Product....13
Table 15: Functional Requirement 13 - Update Product....13
Table 16: Functional Requirement 14 - Delete Product....14
Table 17: Test Case Table....27
Table 18: Test Case Result Table....28

Chapter 1

Introduction

1.1 Overview/Background of the Work

As technology is advancing in this fast-growing world, business operations are also adapting new technologies to gain more flexibility and workability. Customer Relationship Management (CRM) system have become crucial tools for optimizing procedures, enhancing customer engagement, and managing customer interactions. A CRM system acts as a central hub that enables companies to develop solid client relationships, company processes, and ultimately lead to increased productivity and profitability. We have built a CRM for a company (QuantumSync). The custom CRM system is developed using modern technologies such as React, Express, and MongoDB. These technologies were chosen for their ability to create a responsive user interface, facilitate seamless communication between the frontend and backend, and provide efficient data storage and retrieval capabilities. The project follows MVC architecture. The system will mainly help the company to manage their customers, products, suppliers, shops, sales information etc.

1.2 Objectives

The system aims to offer an intuitive user interface, efficient customer data management, streamlined sales processes, supplier management and product and shop management. The Focuses are:

- Developing a web-based application
- Building a user-friendly system that is easy understand and operate
- Building APIs to execute operations
- Integrating authentication module
- Developing customer management
- Developing supplier management
- Developing product management
- Developing order management
- Chart generation

1.3 Scopes

The Scopes of the project:

- Authentication: User can Sign-Up to the system and log into the system. User can reset their password.
- Customer management: Super admin can create, update, delete, sort, search customer. Also can create customer order, order invoice.
- Product Management: Admin can create, update, and delete, sort, search product, product category and product type information.
- Supplier management: Super admin can create, update, delete, sort, search supplier.
- Generate order charts

Chapter 2

Literature Review

2.1 Relationship with Undergraduate Studies

All of my courses and professors have taught me great principles that have benefited my professional life and, I am confident, will also benefit me in my future problems. List of courses that helped in the development process:

CSE 203, Data Structures: This is the course that helped me with the concepts of various data structures and where to apply them.
CSE 213, Object Oriented Programming: Data is represented as an object in most of the developing industry. This course has gave idea that how to approach a project in an object oriented way.
CSE 303, Database Management: I have learnt how to design and plan a project and how to deal with databases in this course.
CSE 307, System Analysis and Design: This course gives an overview of Used Case Diagram, Used Case Scenario, SDLCs and how to adapt each one of them to the project. Different UML diagrams have been taught in this course which helped me to analyze the project and build the project report.
CSE 309, Web Application and Internet: Web apps development was covered in this course. It covered a number of crucial technologies that are in great demand in the sector, including HTML, CSS, and JavaScript. The technologies and techniques covered in this course make it easier to learn React and Express.
CSE 451, Software Engineering: This course was one of the most important courses which gives an overview over different aspect building a software. SDLCs, WBS, Testing was taught in this course.

2.2 Related works

There are two examples I would like to talk about. (All concise lists taken from https://www.g2.com/categories/crm)

1. Salesforce Sales Cloud: Salesforce Sales Cloud is the complete platform for Sellers, Sales Leaders, and Sales Operations, to grow sales and increase productivity. (https://www.salesforce.com/products/sales-cloud/)

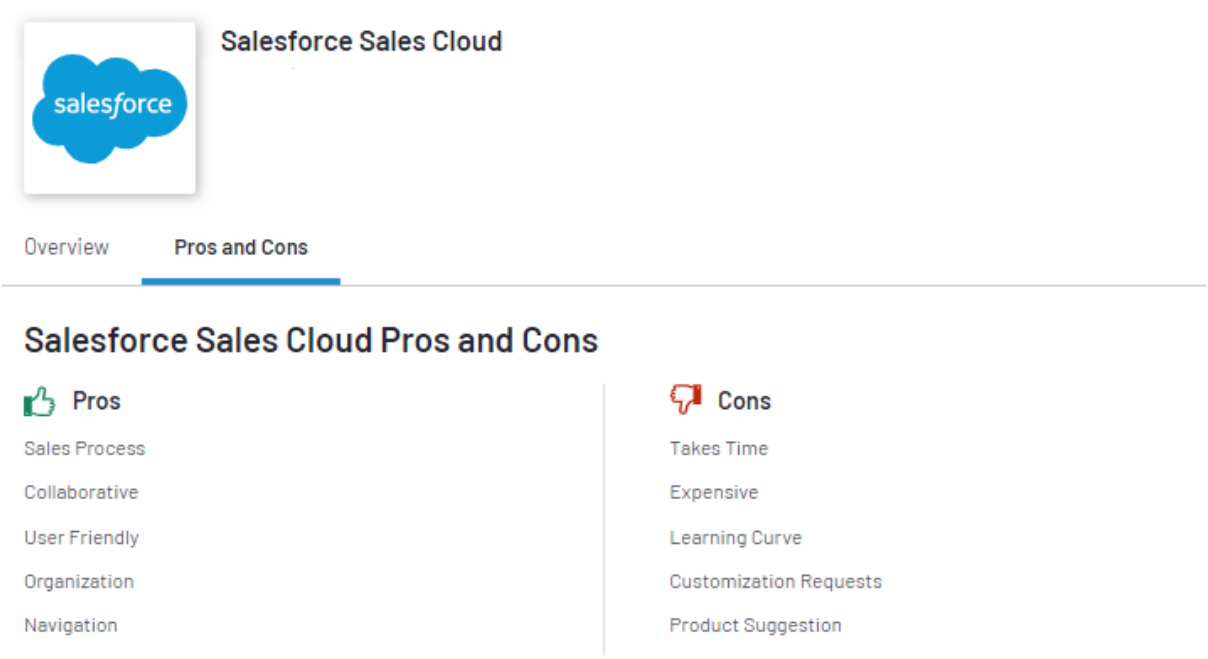


Figure 1: SalesForce Sales Cloud

2. ClickUp: The productivity platform ClickUp is all-inclusive. Using tasks, docs, chat, goals, whiteboards, and other tools, teams congregate there to plan, organize, and collaborate on work. (https://clickup.com/features)

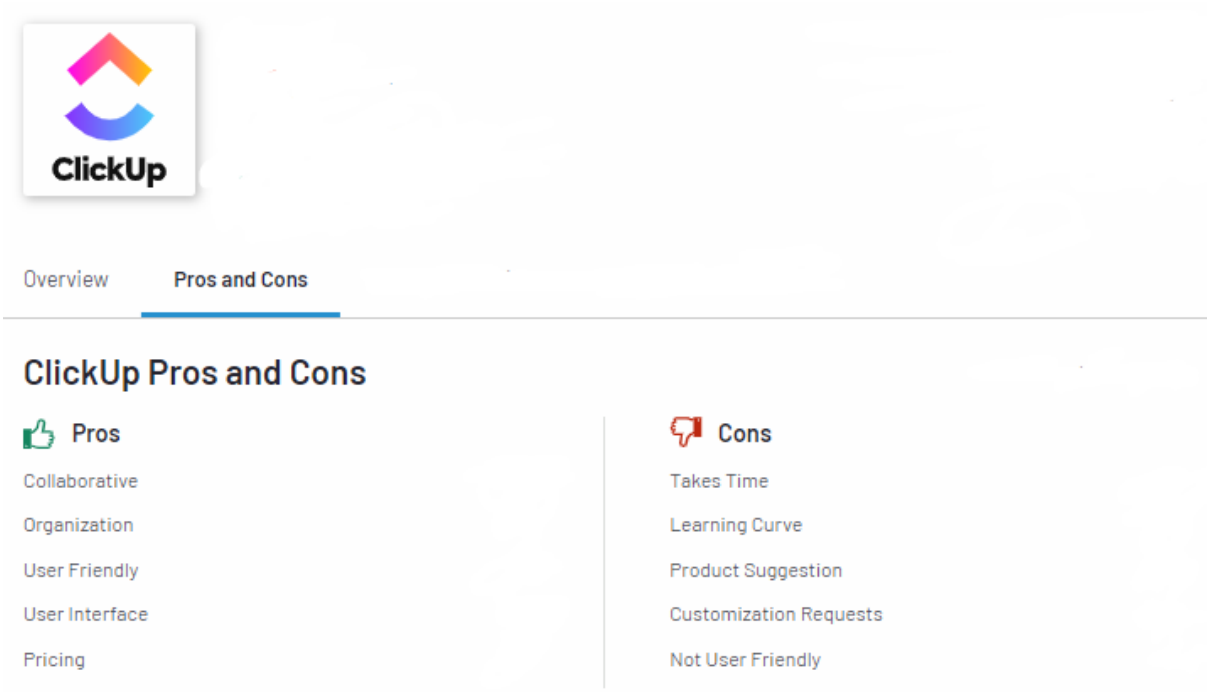


Figure 2: ClickUp

Chapter 3

Project Management & Financing

3.1 Work Breakdown Structure

A WBS is a hierarchical structure which demonstrates the division of a project into smaller parts. In order to organize our work on QuantumSync, we have established a WBS. The WBS offers a visual depiction of all scopes, risks, communications, responsibilities, costs, and guarantees that crucial deliveries won't be missed. We've used the top-down approach in this case.

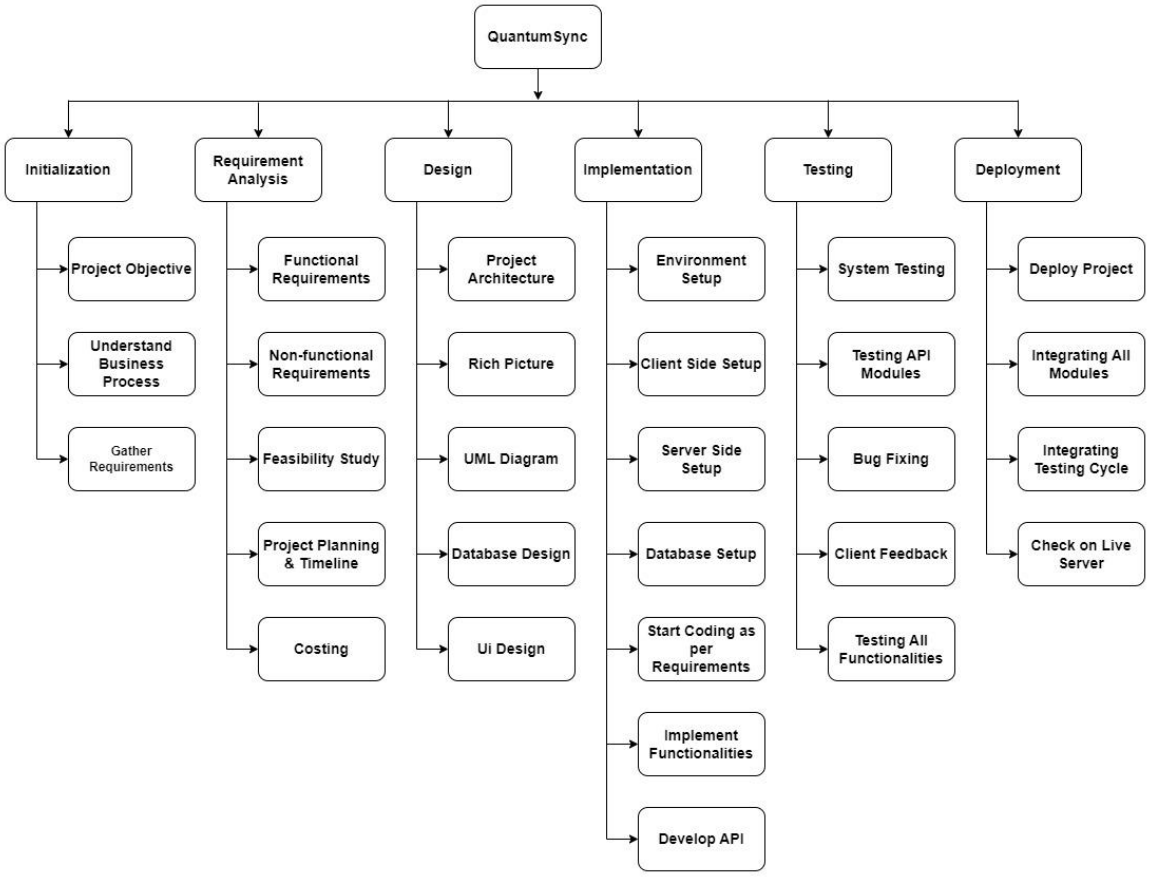


Figure 3: WBS of QuantumSync

3.2 Process/Activity wise Time Distribution

For each part shown in the WBS Diagram above, we allocated a certain amount of time, as required to complete the tasks. The Table below shows them in detail.

Task	Days	Work Percentage (%)
Initialization	6	6.38
Requirement Analysis	8	8.51
Design	14	14.89
Implementation	42	44.68
Testing	14	14.89
Deployment	10	10.65
Total =	**94**	**100**

Table 1: Project Time Allocation

3.3 Gantt Chart

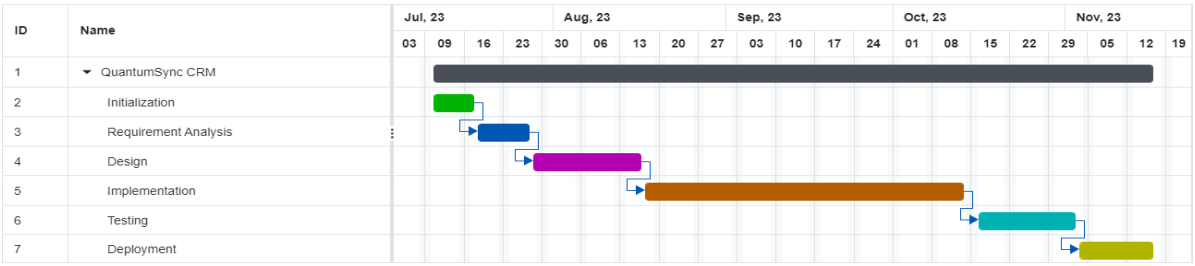


Figure 4: QuantumSync Gantt Chart

3.4 Process/Activity wise Resource Allocation

Resource Type	Name	Description
Hardware	Desktops/Laptops	Platform needed to work
	Routers	For internet access
	USB Drives	Offline data transfer
Network	Internet	Connection to the web
	Hosting	Making website available via World Wide Web
	Domain	Website identification string
Human	Project Manager	Organizes and allocates work
	UI/UX Designer	Designs website interface
	Developers	Code and develop the project
	DevOps	Optimizes the flow of data and automates infrastructure
	Tester	Tests the entire project
Software	Visual Studio Code	Code editor and development environment
	Windows, MacOS, Linux	Operating systems preferred by developers
	Node Runtime	Node runtime to run project

Table 2: Resource Allocation Table

3.5 Estimated Costing

It would naturally have a charge as it is software. It would also require expenses to maintain as it is a web application that is constantly in use. So, estimates of these costs are given below:

Item	Cost (USD)
Development process	9500$
Server cost per year	2200$
Maintenance cost per year	500$

Chapter 4

Methodology

A software development life cycle (SDLC) is a procedure that enables a corporation to quickly and effectively deliver a software product. Every modern software development firm has implemented the concept of task and process breakdown. Phase separation improves project management and system design for that particular system. The seven phases that make up a software development life cycle (SDLC) are mostly as follows:

1. Planning
2. Analysis
3. Design
4. Development
5. Testing
6. Implementation
7. Maintenance

The most commonly used methodologies in the software development firms are:

- Agile Methodology
- Waterfall Methodology
- DevOps
- Spiral
- Iterative
- V-Model

The Software Development Life Cycle (SDLC) Methodology that we will use in our project is the Agile Methodology since it divides the phases into many smaller phases, allowing the tasks to be completed more quickly and effectively. This allows the jobs to be completed whenever they are needed, customer feedback is gathered thereafter, and if changes are necessary, they will be made quickly because all the features don't need to be designed or executed sequentially. As a result, the project may be better controlled and managed by the developers and project managers.

Chapter 5

Body of the Project

5.1 Work Description

QuantumSync is a web-based CRM. The client will mainly control their electronics components and tools sell and supply operations through this system. The system will have authentication, customer, supplier, product, shop, order generation, report generation module. React.js, Tailwind, DaisyUI will be used for frontend. Backend will be crafted by Express.js. Firebase will be used for authentication module. We will use MongoDB as database to maintain a document-based data transaction model. As there will be lots of products, and we need to load those products into the system. JSON based data transaction will be an easier method to load a lot of products. We will follow MVC architecture to build this project. As a member of the development for the project, I had contributed to both the front end and the backend of the application

5.2 Requirement Analysis

5.2.1 Rich Picture

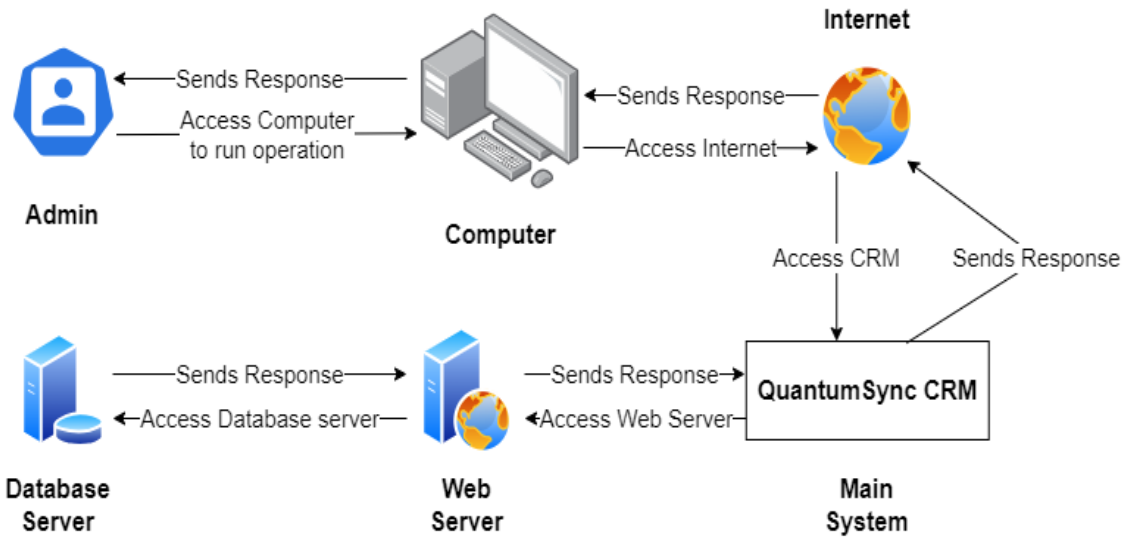


Figure 5: Rich Picture of QuantumSync

5.2.2 Functional and Non-Functional Requirements

Functional Requirements:

Function: Sign Up		
Input: • Enter name, email, password, confirm password • Click "Sign Up" button	**Process:** • Call firebase signup module to create a user and call a server API to store user information	**Output:** • User will be created and user will be redirected to dashboard
Pre-condition: There cannot be any duplicate user.		
Post-condition: User will get a successful message for account creation		

Table 3: Functional Requirement 1 - Sign Up

Function: Login		
Input: • Enter email, password • Click "Login" button	**Process:** • Call firebase login module to initiate login	**Output:** • User will be redirected to dashboard
Pre-condition: User should be registered.		
Post-condition: User should redirect to dashboard.		

Table 4: Functional Requirement 2 – Login

Function: Reset Password		
Input: • Click "Forget Password" • Enter email in the pop-up • Click "Submit" button	**Process:** • Call firebase reset password module • In the pop-up modal write email address and click 'submit' button	**Output:** • A password reset email will be send to the users email address.
Pre-condition: User should be registered.		
Post-condition: Password reset email will be send to user		

Table 5: Functional Requirement 3 - Reset Password

Function: Add Customer		
Input: • Enter name, email, phone, street address, city, state, zip code • Click "Add" button	**Process:** • Go to Customers page • Click "Add a customer" button • Fill up the pop-up customer form • Click "Add" button	**Output:** • Customer data will be added into database by following no duplication constraint.
Pre-condition: User must be super admin and logged in.		
Post-condition: Customer information will be added to database and will appear in table in the system.		

Table 6: Functional Requirement 4 - Add Customer

Function: Update Customer		
Input: • Enter (only required fields) name, email, phone, street address, city, state, zip code • Click “Update” button	**Process:** • Go to Customers page • Click “Update” button for that specific customer • Change the desired fields in the pop-up customer update form • Click “update” button	**Output:** • Updated customer data will be added into database.
Pre-condition: User must be super admin and logged in.		
Post-condition: Updated Customer information will be added to database and will appear in table in the system.		

Table 7: Functional Requirement 5 - Update Customer

Function: Delete Customer		
Input: • Click “Delete” button	**Process:** • Go to Customers page • Click “Delete” button for that specific customer to delete • Click “Ok” in pop-up to confirm the delete process or “cancel” to abort delete	**Output:** • Customer data will be removed from database.
Pre-condition: User must be super admin and logged in.		
Post-condition: Customer information will be removed from database and will disappear from table in the system.		

Table 8: Functional Requirement 6 - Delete Customer

Function: Take Order from Customer		
Input: • Enter customer name, email, phone, address • Enter product name, quantity, and price. • Click “Place Order” button	**Process:** • Go to Customers page • Go to “Take Order” tab • Fill the form to take order from a customer and then click “Place Order”.	**Output:** • Customer order data will be added to database.
Pre-condition: User must be super admin and logged in.		
Post-condition: Customer order information will be added to database.		

Table 9: Functional Requirement 7 - Order from customer

Function: Add Supplier		
Input: • Enter supplier name, email, phone, street address, city, state, zip code • Click “Add” button	**Process:** • Go to Supplier page • Click “Add Supplier” button • Fill the form to add supplier information. • Click “Add” button	**Output:** • Supplier data will be added into database by following no duplication constraint.
Pre-condition: User must be super admin and logged in.		
Post-condition: Supplier information will be added to database.		

Table 10: Functional Requirement 8 - Add Supplier

Function: Update Supplier		
Input: • Enter (only required fields) name, email, phone, street address, city, state, zip code • Click “Update” button	**Process:** • Go to Supplier page • Click “Update” button for that specific supplier • Change the desired fields in the pop-up customer update form • Click “update” button	**Output:** • Updated supplier data will be added into database.
Pre-condition: User must be super admin and logged in.		
Post-condition: Updated supplier information will be added to database.		

Table 11: Functional Requirement 9 - Update Supplier

Function: Delete supplier		
Input: • Click “Delete” button	**Process:** • Go to supplier page • Click “Delete” button for that specific supplier to delete • Click “Ok” in pop-up to confirm the delete process or “cancel” to abort delete	**Output:** • Supplier data will be removed from database.
Pre-condition: User must be super admin and logged in.		
Post-condition: Supplier information will be removed from database and will disappear from table in the system.		

Table 12: Functional Requirement 10 - Delete Supplier

Function: Change Order Status		
Input: • Click “Pending” or “complete” or “Cancelled” button	**Process:** • Go to customer’s page • Click “Place Order” tab • Click “Pending” or “complete” or “Cancelled” button for a specific order	**Output:** • Customer order status will be changed to either “Pending” or “complete” or “Cancelled” .
Pre-condition: User must be super admin and logged in.		
Post-condition: Customer order status will be changed.		

Table 13: Functional Requirement 11 - Order via email

Function: Add Product		
Input: • Enter product image, product name, quantity, product category, brand name, price, product type, discount • Click “Add” button	**Process:** • Go to product page • Click “Add Product” button • Fill the form to add product information. • Click “Add” button	**Output:** • Product data will be added into database by following no duplication constraint.
Pre-condition: User must be super admin and logged in.		
Post-condition: Product information will be added to database.		

Table 14: Functional Requirement 12 - Add Product

Function: Update product		
Input: • Enter (only required fields) product image, product name, quantity, product category, brand name, price, product type, discount • Click “Update” button	**Process:** • Go to Product page • Click “Update” button for that specific product • Change the desired fields in the pop-up product update form • Click “update” button	**Output:** • Updated product data will be added into database.
Pre-condition: User must be super admin and logged in.		
Post-condition: Updated product information will be added to database and will appear in table in the system.		

Table 15: Functional Requirement 13 - Update Product

Function: Delete Product		
Input: • Click “Delete” button	**Process:** • Go to product page • Click “Delete” button for that specific product to delete • Click “Ok” in pop-up to confirm the delete process or “cancel” to abort delete	**Output:** • Product data will be removed from database.
Pre-condition: User must be super admin and logged in.		
Post-condition: Product information will be removed from database and will disappear from table in the system.		

Table 16: Functional Requirement 14 - Delete Product

Non-functional Requirements:

Non-functional requirements are the features of the system that specify its properties, including usability, availability, security, compatibility, the system's response time, and many more. They act as the system's blueprint, enabling the system to carry out each activity better and with greater user friendliness.

Performance: When it comes to user satisfaction, performance is crucial. As a result, the system was created to operate flawlessly when users engage with it, allowing them to navigate without difficulty and comprehend its functionalities.

Security: Users' top concern when using the internet is security, and the system has good security measures built in from the beginning to prevent users' sensitive information from being released online.

Usability: Usability is defined as responsiveness and user satisfaction with comprehensive and simple navigation throughout. The system has been created and developed for this aim to ensure a user-friendly UI/UX.

Availability: The system has been designed to remain operational at all times, even though it is essential for an online-based system to be accessible to users. Everything depends on the work done to keep it up to date and make it accessible to users without any lags or functional issues.

Scalability: The system is designed in such a way so that it can adapt any kind changes in the system. Every module is controlled separately. It can control large volume of data. System is accessible from any platform.

5.3 System Analysis

5.3.1 Six Element Analysis

Process	Human	Non-computing hardware	Computing hardware	Software	Database	Network and communication
Sign Up	User enters user information to create account	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Login	user enters credentials into login screen	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Reset password	User resets password via email	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Add customer	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Update customer	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Delete customer	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Take order	Super Admin	Pen, paper	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Change order Status	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Add supplier	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Update Supplier	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Delete supplier	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Add product	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Update product	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet
Delete product	Super Admin	N/A	Desktop/Laptop, keyboard, mouse	Web Browser	MongoDB	Internet

5.3.2 Feasibility Analysis

A feasibility analysis is done to determine whether a project is feasible and deserving of further consideration. This is known as a feasibility study. The analysis's main goal is to finish the design, plan, and strategy. Assumptions, restrictions, options, and strategies can all be verified using this technique. Technical, operational, and economic feasibility are the three categories of feasibility.

Technical Feasibility: A system's technical feasibility is assessed by looking at the software, hardware, and other technological criteria. This part discusses how we plan to deliver a good or service to the customer. It helps businesses determine whether the technical staff is capable of turning ideas into workable solutions and whether the resources available are sufficient for their needs.
QuantumSync is created utilizing Express framework which is built on node.js architecture and React library. These technologies are quite common in current industry and are widely used in many businesses.

Operational Feasibility: A measure of a proposed system's ability to address problems, take advantage of opportunities discovered during scope definition, and adhere to requirements discovered during the requirements analysis stage of system development is called operational feasibility.
It is quite easy to operate QuantumSync because of the way it has been created. It's a carefully designed system. The organization's overall business plan is supported by this system. Users don't need a lot of technical expertise to operate this system. Every instruction is clear to the users. All of the users' needs should be able to be satisfied by this system, we anticipate.

Economic feasibility: This economic feasibility research includes a cost/benefits analysis of the project, which enables businesses to weigh the costs and advantages of the project before allocating resources. Additionally, it serves as a third-party project evaluation and promotes project credibility. This helps decision-makers evaluate the project's financial advantages for the business.
By evaluating the overall application system, we came to learn that the cost to conduct a full system investigation is possible.

5.3.3 Problem Solution Analysis

Understanding our difficulties and how to address them is the process of problem solution analysis. When a problem is solved, a solution is found that satisfies the requirements and limitations of the issue. Even while the goal of the system may be perceived as improving current systems or taking advantage of market opportunities, a lot of what is done in designing and creating information systems is to solve problems.

Problem:
The client needs a web-based CRM system to manage their electronics components and tools selling and supply operations. They require authentication, customer, supplier, product, shop, order generation, and report generation modules.

Solution:
To address the client's needs, we have developed QuantumSync, a web-based CRM system with the following components and technologies:

Authentication Module (Firebase): We identified the need for secure user authentication and chose Firebase for its robust authentication capabilities. This solution ensures that only authorized personnel can access and manage the system.

Frontend Technology (React.js, Tailwind, and DaisyUI): We utilized React.js for the frontend, offering a responsive and dynamic user interface. Tailwind and DaisyUI streamline the development process, providing a clean and visually appealing design.

Backend Technology (Express.js): For the backend, we selected Express.js, a fast and scalable Node.js framework. This ensures efficient data handling and processing for a smooth user experience.

Database (MongoDB): MongoDB, a document-based database, was chosen to manage the extensive product catalog efficiently. JSON-based data transactions simplify the process of loading a large number of products into the system.

We have implemented key modules, including:

Customer Module: Enables the management of customer data and interactions.
Supplier Module: Facilitates supplier relationship management.
Product Module: Manages the product catalog and inventory.
Order Generation Module: Streamlines the order creation process.
Report Generation Module: Allows for data analysis and reporting.

MVC Architecture: We have adopted the MVC (Model-View-Controller) architecture to ensure a well-organized and maintainable codebase. This separation of concerns enhances code readability and scalability.

5.3.4 Effect and Constraints Analysis

Each project has constraints and risks that must be controlled if it is to be long-term successful. Time, cost, and scope are the three fundamental project management constraints. Frequently referred to as the "three limitations" or the "project management triangle." Increasing the project's scope, for instance, will probably demand more time and resources, whilst reducing the project's timeline can lower expenses while expanding the project's scope.

Constraint 1: Time

Time is a crucial factor in any project development. In our project, all employees worked regularly and giving their daily update every day. This kept our project on track and no such delays were reported.

Constraint 2: Cost

A project's budget includes both fixed and variable costs, such as those for materials, permits, work, and the financial impact of the project team members. The budget was already estimated because our project has already through multiple evaluations.

Constraint 3: Scope

The project's limits are its scope. It includes objectives that the project and the organization must meet. Not only will you find deliverables in scope, but also the methods used to create them. There was no backtracking in our project because the scopes were already established at the beginning.

5.4 System Design

5.4.1 UML Diagrams

Use case Diagram

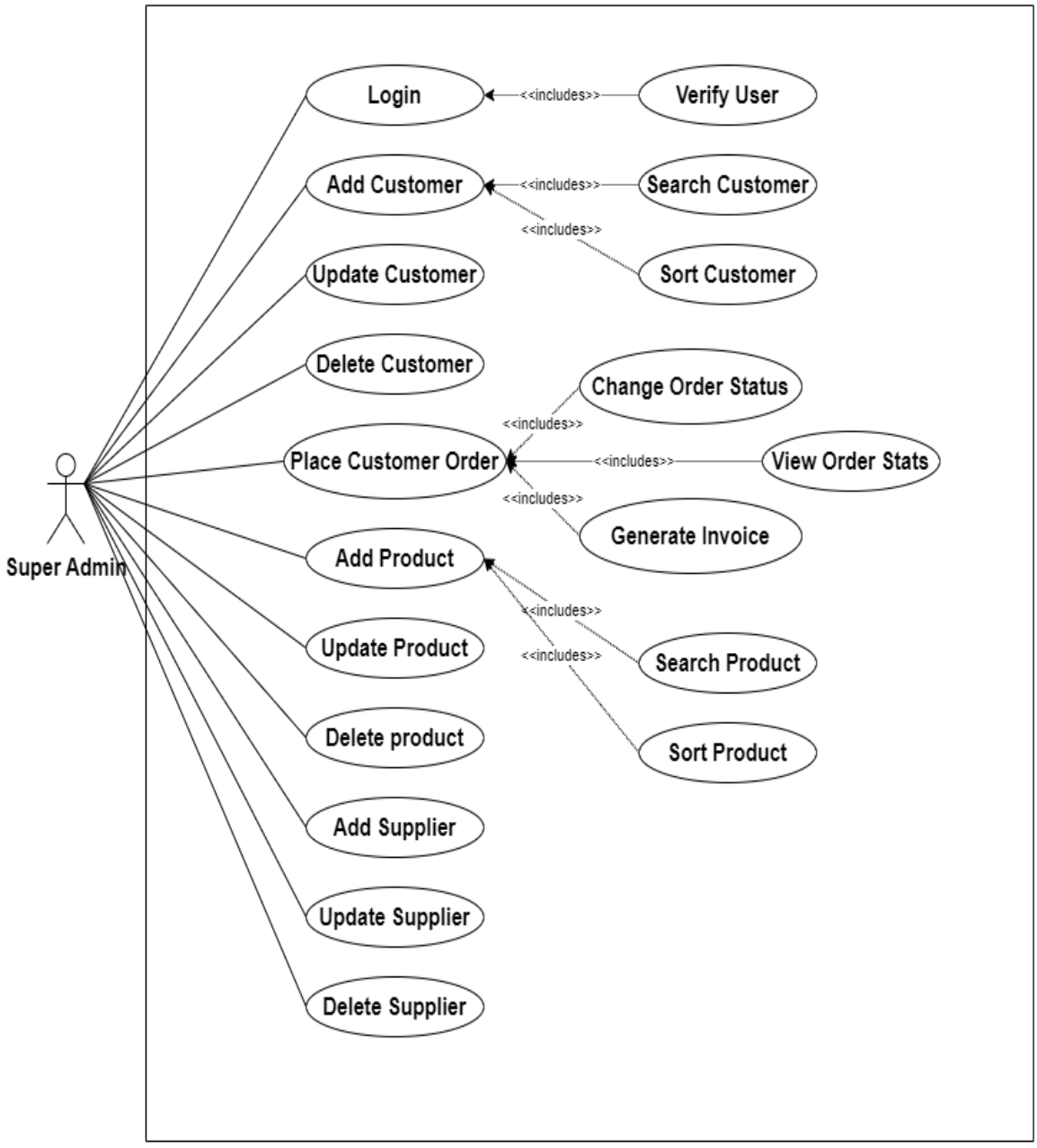


Figure 6: QuantumSync Use Case Diagram

Activity Diagram of Super Admin

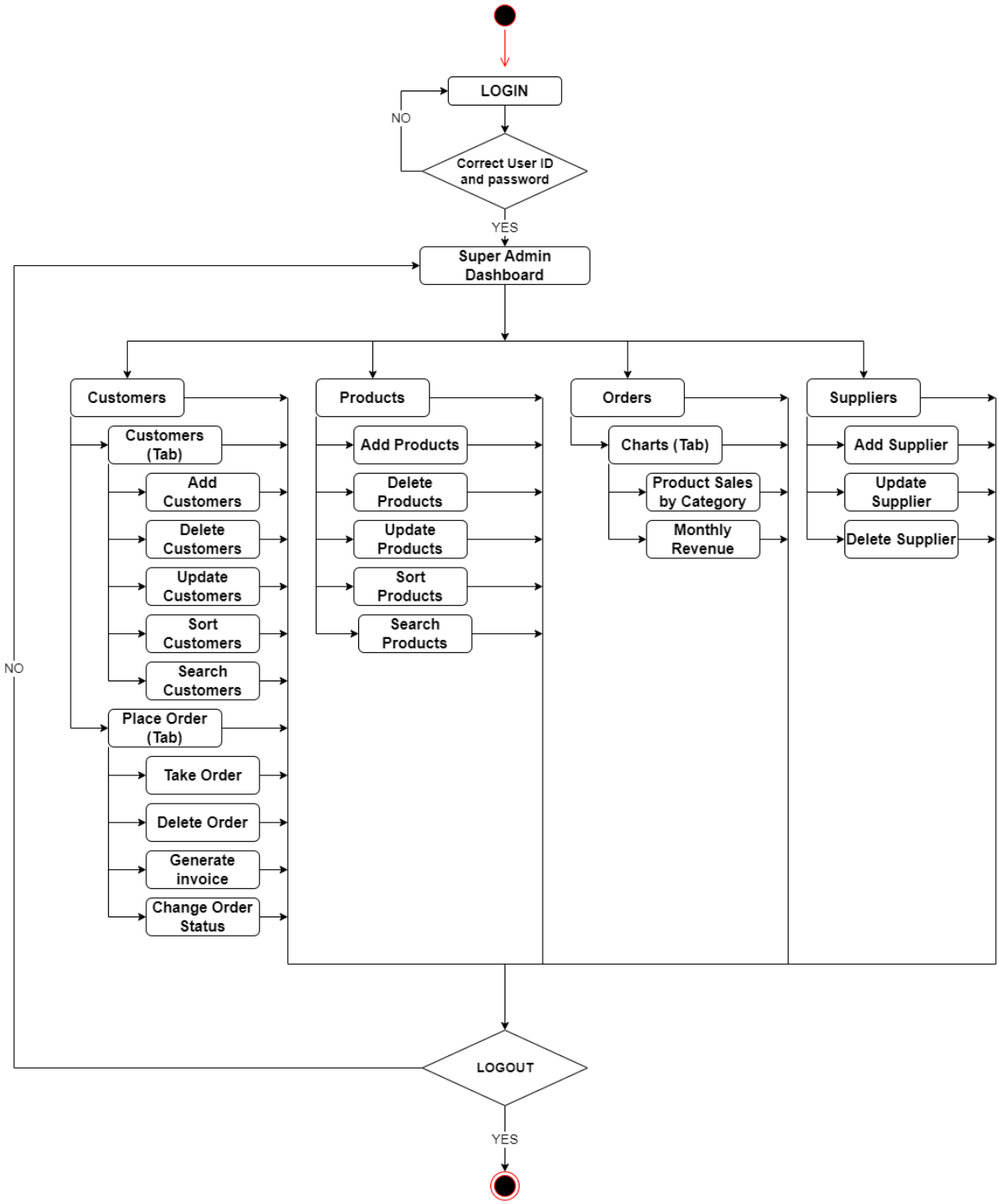


Figure 7: Super Admin Activity Diagram

5.4.2 Architecture

We followed 3 tier architecture to build the system. The three-tier architecture is a way of organizing a software application into three main parts, each responsible for specific tasks. Let's break down each tier:

1. **Client Tier:** This is the part of the software that the user interacts with directly. It's like the "front end" of the application that we see and use. It includes the user interface (UI), which can be a website, mobile app, or any interface that users interact with. The client tier communicates with the logic tier to request and display data or perform actions.
2. **Logic Tier:** This is like the "brain" of the application. It handles the core functionality and processing. It contains the business logic, which defines how the application should work.
3. **Data Tier:** This is where the application's data is stored, the database tier. It's often referred to as the "back end" of the application. It's responsible for managing and storing data. The data tier is accessed by the logic tier when the application needs to read or write data. It provides a way to store and retrieve information efficiently.

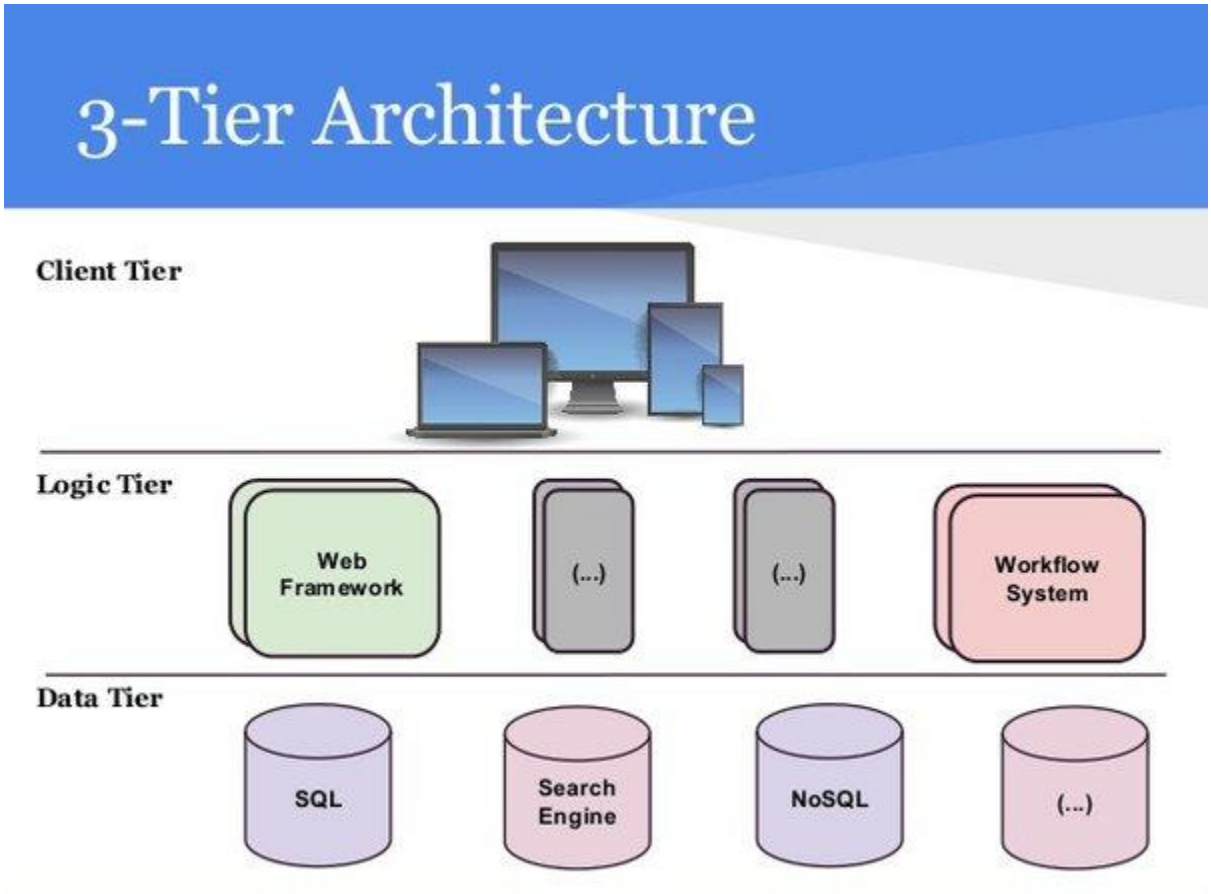


Figure 8: System Architecture

5.5 Implementation

5.6 Testing

5.6.1 Input

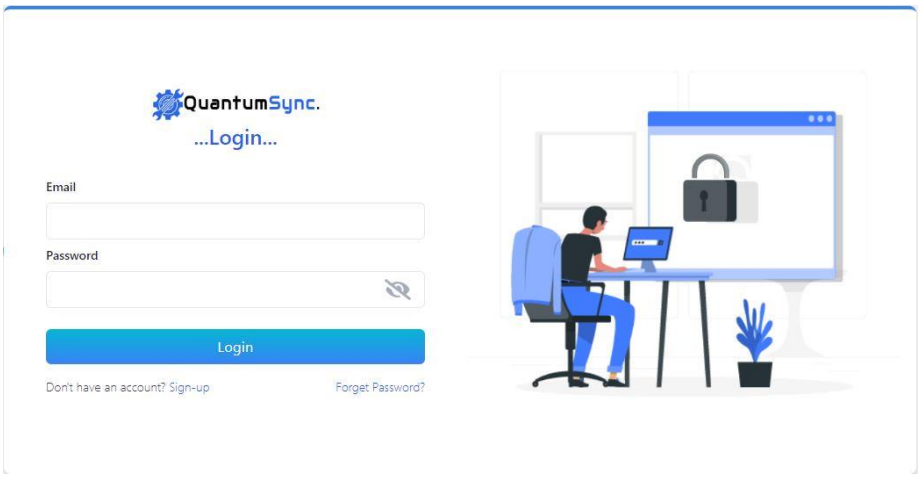


Figure 9: Login Page

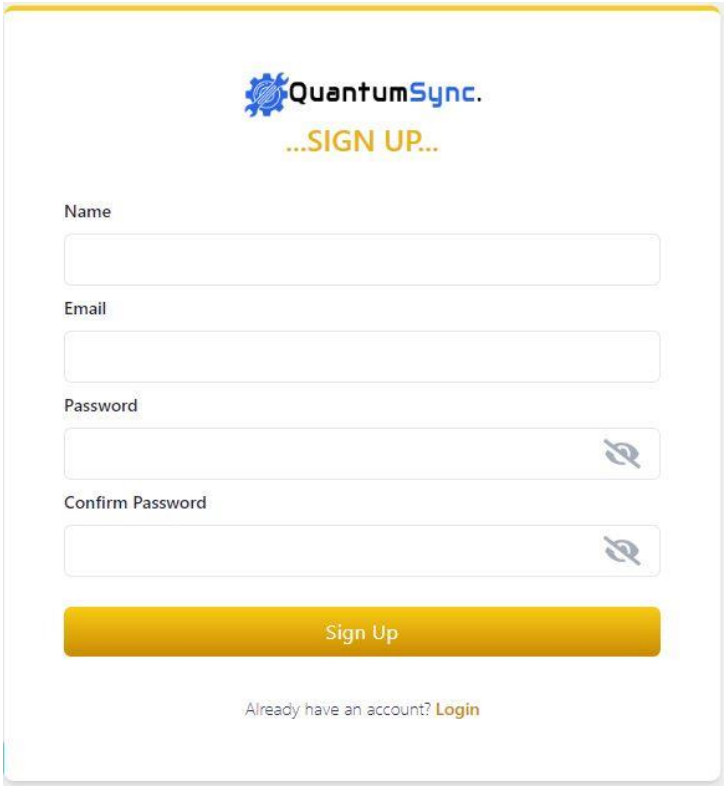


Figure 10: Sign-up Page

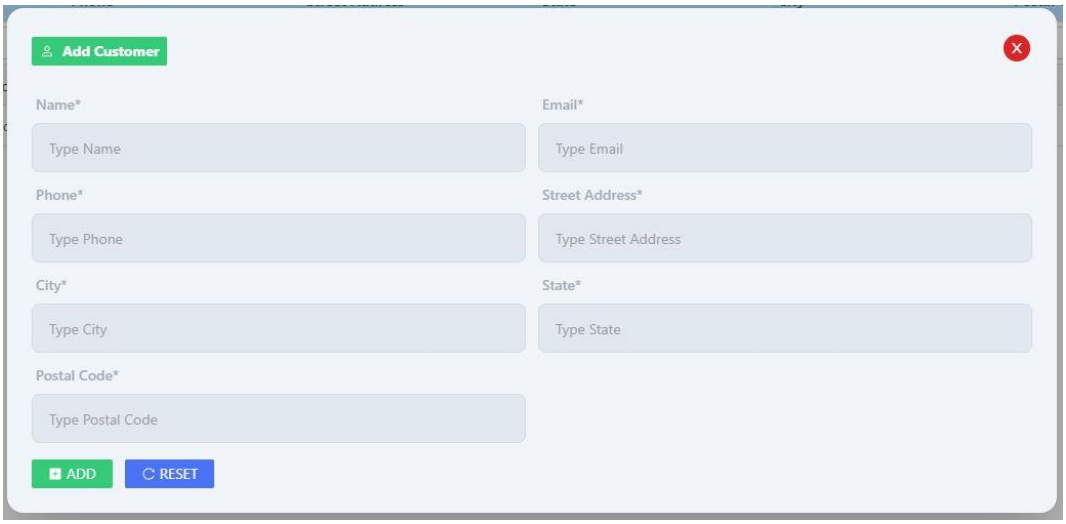


Figure 11: Add Customer Modal

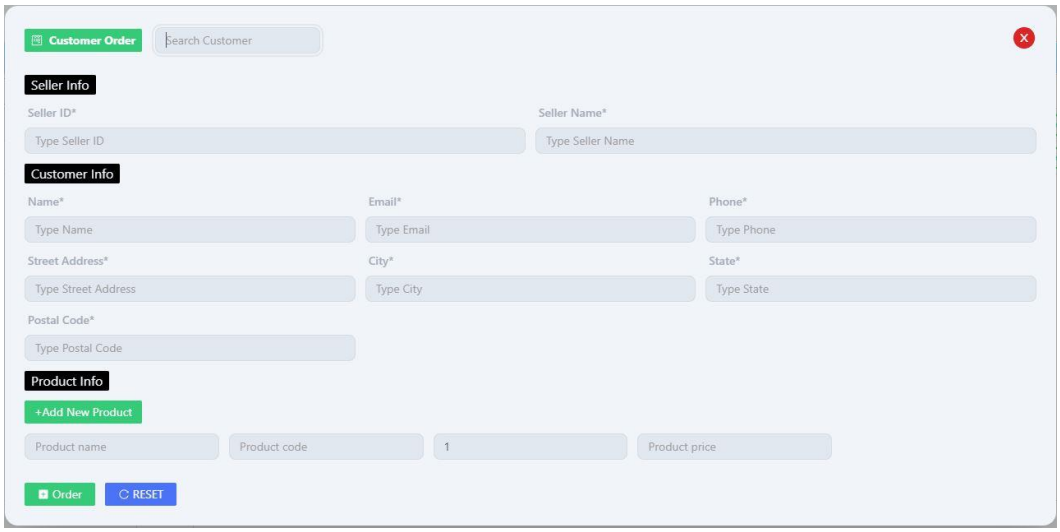


Figure 12: Place Order Modal



Figure 13: Add Supplier Modal

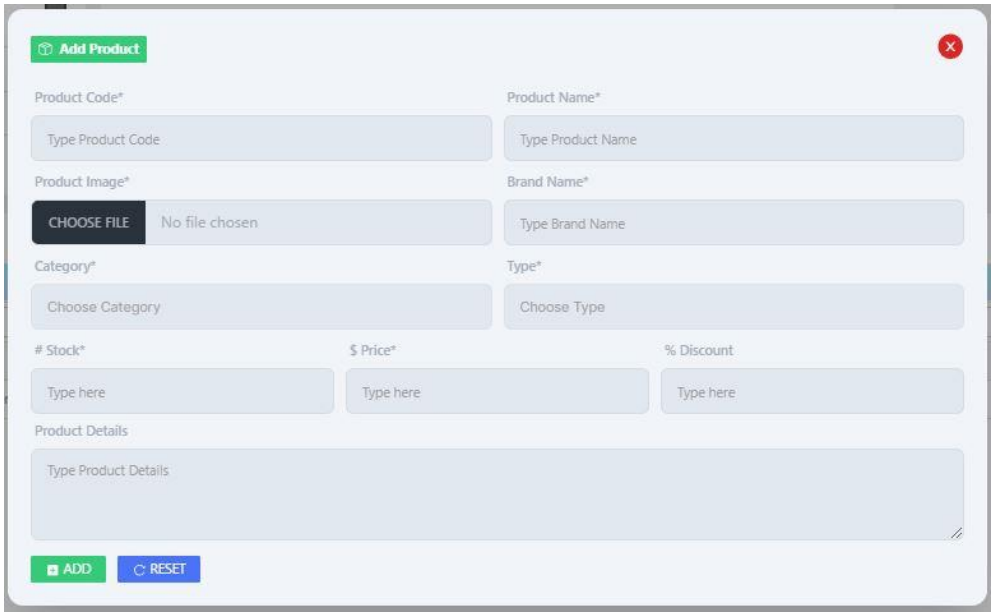


Figure 14: Add Product Modal

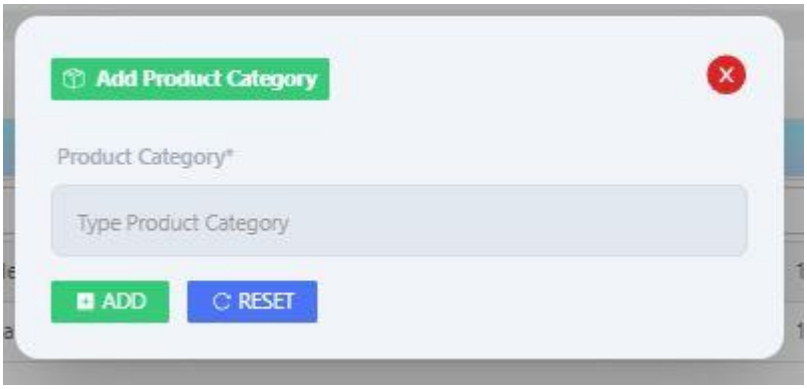


Figure 15: Add Product Category Modal



Figure 16: Add Product Type Modal

5.6.2 Output

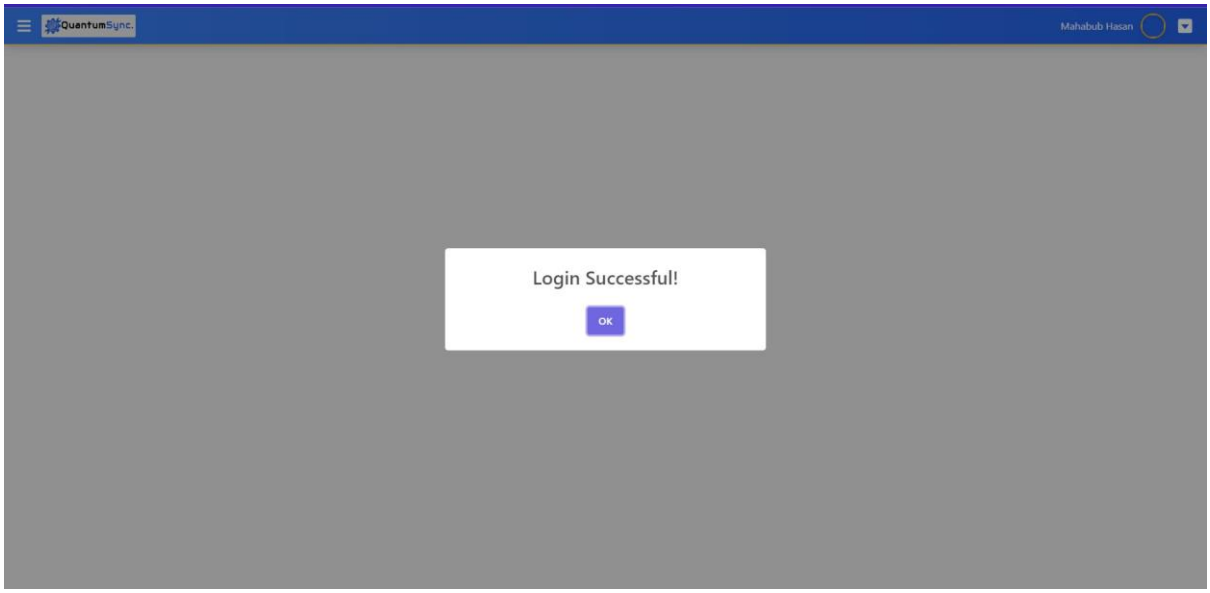


Figure 17: Super Admin Dashboard

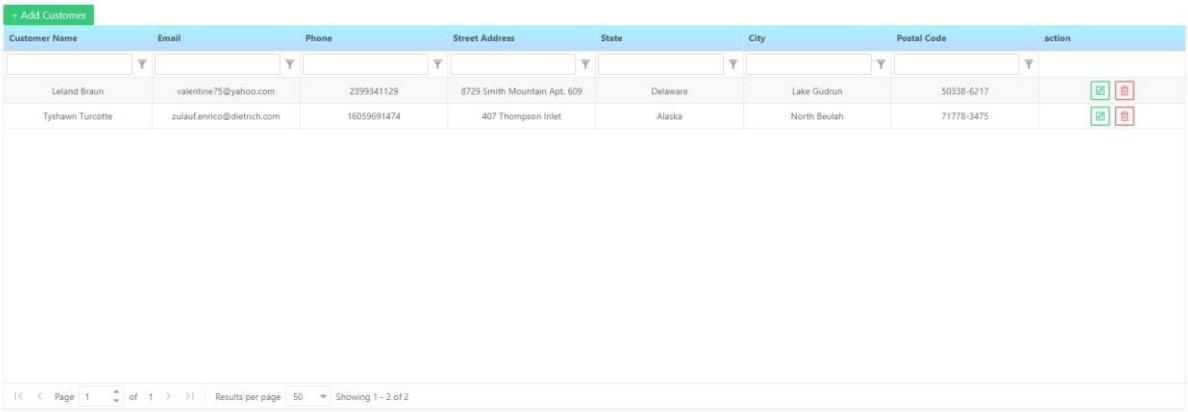


Figure 18: Customer Table

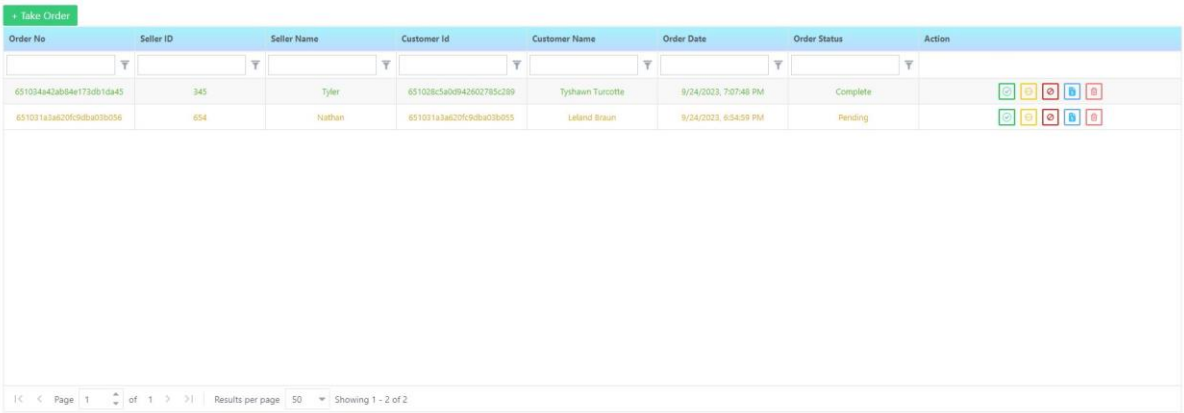


Figure 19: Order Table

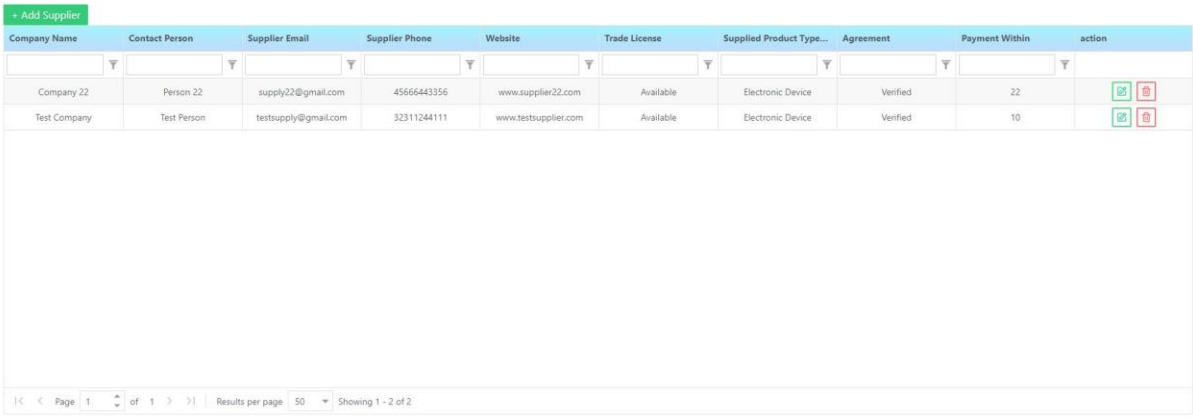


Figure 20: Supplier Table

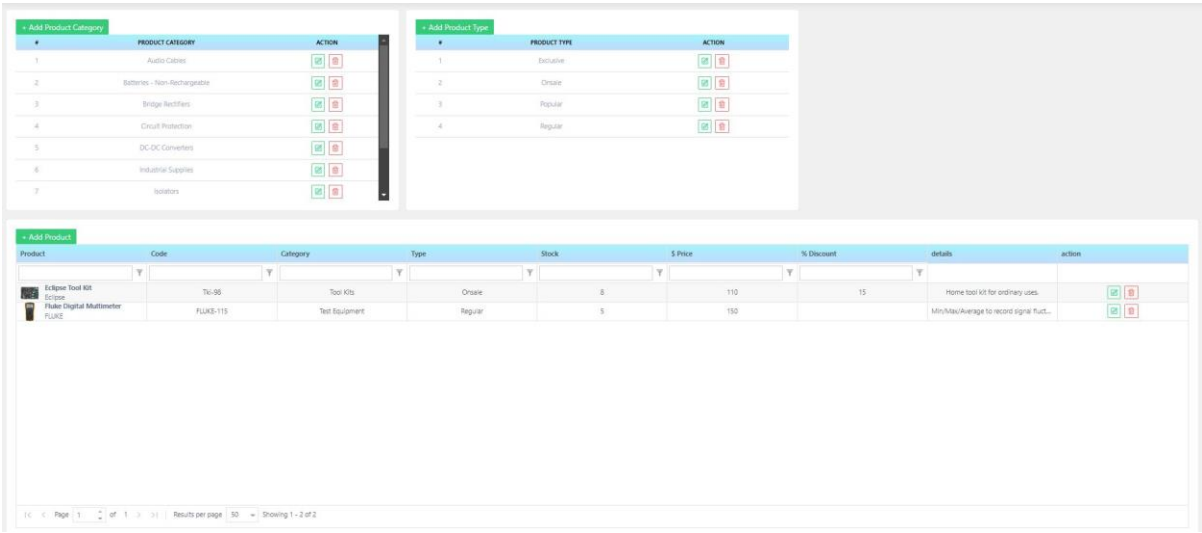


Figure 21: Product, Product Category, Product Type Table

5.6.3 Designing Test Cases

Test Case ID	Test Case Description	Preconditions	Test Steps	Expected Result	Post-conditions
1	Valid Login	1. The application is running. 2. The user has a valid account.	1. Go to login page. 2. Enter a valid email and password. 3. Click "Login" .	Login successful and user redirected to the dashboard.	The user is logged in.
2	Invalid Email or Password	1. The application is running. 2. The user has an invalid email or password.	1. Go to login page. 2. Enter an invalid email or password. 3. Click "Login".	An error message is displayed indicating that the login credentials are invalid.	Login not successful.
3	Empty email or Password Fields	1. The application is running. 2. The user leaves either the email or password field empty.	1. Go to login page. 2. Leave either the email or password field empty. 3. Click "Login".	An error message is pop-up indicating that the email and password fields are required.	Login not successful.
4	Forgot Password	1. The application is running.	1. Go to login page.	A password reset email will be send to the user's email address with password reset instruction.	Login not successful but user has initiated the password reset process.

Table 17: Test Case Table

5.6.4 Test Results

Test Case ID	Result
1	Log in success. User redirects to dashboard.
2	The system displays an error message indicating that the login credentials are invalid. The user is not logged in.
3	Error message will be pop-up indicating that username and password fields are required. The user is not logged in.
4	A password reset email sent to the user's email address with instructions on how to reset the password. The user is not logged in but has initiated the password reset process.

Table 18: Test Case Result Table

Chapter 6

Results & Analysis

The overall work was difficult and sometimes confusing for a new developer as myself. However, as I was only implementing some of the features of the system rather than the whole thing, it was more bearable.
Meetings with the external supervisor led me to believe trial and error was the way to go. The main goal of the project was to build a CRM that is easy to navigate and operate.
Analysis of the system:

Authentication:

- **Login:** System has a secure login system with proper error handling functions.
- **Sign-up:** System has a sign up module for new user.
- **Forget Password:** User can reset their password using reset password feature.

Customer Module:

- **Add Customer:** Admin will be able to add customer data. System will not allow duplicate customer data.
- **Update Customer:** Customer data can be updated. During the update, data duplication is not allowed.
- **Delete Customer:** Customer data can be deleted. Confirmation pop-up for deletion will appear to prevent unexpected data loss.
- **Search & Sorting:** Search and sorting data feature has been added to table.
- **Place Order:** Admin can take inputs to place a customer order.
- **Order Status:** Order status can be changed to “Pending” or “Complete” or “Cancelled”.
- **Invoice:** System can generate invoice pdf for each and every order with proper calculation.

Supplier Module:

- **Add Supplier:** Admin will be able to add supplier data. System will not allow duplicate supplier data.
- **Update Supplier:** Supplier data can be updated. During the update, data duplication is not allowed.
- **Delete Supplier:** Supplier data can be deleted. Confirmation pop-up for deletion will appear to prevent unexpected data loss.
- **Search & Sorting:** Search and sorting data feature has been added to table.

Product Module:

- **Add Product:** Admin will be able to add product data. System will not allow duplication.
- **Update Product:** Product data can be updated. During the update, data duplication is not allowed.
- **Delete Product:** Product data can be deleted. Confirmation pop-up for deletion will appear to prevent unexpected data loss.
- **Search & Sorting (for product only):** Search and sorting data feature has been added to table.
- **Add Product Category:** Product category can be added without data duplication.
- **Update Product Category:** Product category can be updated without data duplication.
- **Delete Product Category:** Product category data can be deleted. Confirmation pop-up for deletion will appear to prevent unexpected data loss.
- **Add Product Type:** Product type can be added without data duplication.
- **Update Product Type:** Product type can be updated without data duplication.
- **Delete Product Type:** Product type data can be deleted. Confirmation pop-up for deletion will appear to prevent unexpected data loss.

Order Chart Module:

- **Sales by Product Category Yearly:** System will generate chart for total sales of the product by category for every year.
- **Monthly Revenue Chart:** System will generate a monthly revenue chart for every year.

Chapter 7

Project as Engineering Problem Analysis

7.1 Sustainability of the Project/Work

The sustainability of the project depends on its ability to change and advance along with changing business demands and technology developments. Sustainability is ensured through regular maintenance of the client side, server side, database and efficient use of development tools. We've optimized both frontend and backend development by minimizing repetitive code, optimizing scripts, and adhering to coding standards. This ensures smooth performance across all devices. QuantumSync is being kept up-to-date and in line with its specifications through routine maintenance. The system will be maintained and supported by a professional team. Furthermore, using open-source technologies promotes long-term sustainability by reducing dependency on proprietary software and associated costs.

7.2 Social and Environmental Effects and Analysis

Social Effects: QuantumSync is a web-based CRM system that seeks to automate and simplify the majority of the manual processes in the business. The system's social impact is primarily positive, as it enhances customer relationships and communication. System will reduce manual processes. Which ensures faster business operations and better customer service.

Environmental Effects: As a matter of the environment, this system does not do any harm to it. The system eliminates the requirement for paper records and allows all other actions to be carried out without the use of physical copies. As a result, the immediate area will continue to be environmentally friendly.

7.3 Addressing Ethics and Ethical Issues

One of the most crucial assets of any business or organization is data. With the growth of technology, there has been a significant increase in hacking, cyber-crime, and data theft. Ethical considerations in this project include data privacy and security. QuantumSync will ensure that customer data is handled with the utmost care, complying with data protection regulations.
Regular security audits and encryption measures can help safeguard sensitive information. Client side, server side and databases will be maintained separately. There's a lower likelihood of data corruption. In this project JWT verification process has been applied to verify bearer token for the authentic users. Data transactions using API call will be controlled by token verification. Additionally, ethical use of customer data for marketing campaigns should be transparent and consent-based to maintain trust.

Chapter 8

Lesson Learned

8.1 Problems Faced During this Period

Since this internship was a new experience for me, I encountered a few difficulties during my time as an intern at this company. However, with observation and understanding of the corporate world, I was eventually able to overcome these difficulties, and my coworkers and project manager served as an inspiration to me throughout my learning period.
I would like to mention some of the problems that occurred during my internship period:

Understanding the Methodology:
Understanding how the software development life cycle works, is used in actual projects was interesting to me and much more challenging to absorb and apply. For me, it was crucial and challenging to understand the technique because everyone was tasked with figuring out how to improve the system's development process faster.

Adapting to New Technology:
To do the project, I had to learn new technologies like Node, React, Express, MongoDB, Axios, Tanstack Query, React Data Grids etc. Beside that I had to also learn the MVC architecture. These technologies are not quite easy to understand over a short period of time.

8.2 Solution of those Problems

The keys to overcoming the problem were research and development. After examining numerous websites with comparable subject matter, online forums, and tutorials, I was finally able to understand SDLC, MVC, Project structures and related technologies. The project manager helped us to understand how the software development process works.
Learning is a lifelong process, and occasionally the odds are stacked against you. It is our responsibility as engineers to pick up the slack in these situations and manage it through learning. Which, I believe, is what I succeeded in doing.

Chapter 9

Future Work & Conclusion

9.1 Future Works

Several opportunities for further development and improvement of our project become apparent when thinking about the future. Future projects could include the following:

- **Advanced Features Integration:** We can enhance the CRM system by integrating sophisticated features like chatbots, predictive analytics, and customer insights.
- **AI Integration:** AI will be able to detect product selling pattern, customer interactions, and product insights. Moreover, AI will be helpful to take necessary business decisions.
- **Mobile Application:** Developing a mobile application version of the CRM system to provide convenience and accessibility for user's on-the-go.
- **Scalability:** As QuantumSync's business expands or diversifies, the CRM system must be scalable and simple to use.
- **Enhanced Security:** Constantly improving security protocols to keep up with the rapid evolution of cyber threats and guarantee the security of critical customer data.
- **Feedback Mechanism:** Implementing a feedback mechanism will allow users and stakeholders to provide input that will help shape future changes.

9.2 Conclusion

In conclusion, this report and project on a customer relationship management system and most importantly this internship program was a very educational experience that taught me a lot and helped me grow as a technical person.
I have learned the importance of leveraging modern technologies like React, Express, and MongoDB to create a responsive, efficient, and user-friendly software solution.
I did face many hardships but I made sure to adapt to the new environment and make a better version of myself.
Fronture technologies has made me learn and motivate myself, which will forever live within me. I'd like to express my gratitude to the team and everyone at the office for letting me have the opportunity to be amongst them.

Bibliography

1. OpenJS Foundation (2017). *Express - Node.js web application framework*. [online] Expressjs.com. Available at: https://expressjs.com/.

2. reactjs.org. (n.d.). React – A JavaScript library for building user interfaces. [online] Available at: https://legacy.reactjs.org/.

3. www.mongodb.com. (n.d.). MongoDB Node Driver — Node.js. [online] Available at: https://www.mongodb.com/docs/drivers/node/current/.

4. tanstack.com. (n.d.). TanStack Query | React Query, Solid Query, Svelte Query, Vue Query. [online] Available at: https://tanstack.com/query/latest.

5. Axios (2023). Getting Started | Axios Docs. [online] axios-http.com. Available at: https://axios-http.com/docs/intro.

6. Martin, R.C. (2018). Clean Architecture: a craftsman's guide to software structure and design. Boston Columbus Indianapolis New York San Francisco Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montreal Toronto Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo: Prentice Hall.

7. Martin, R.C. (2014). Agile software development, principles, patterns, and practices. Essex: Pearson.