

Internship Report
on
Task Management Tool

Project report submitted in partial fulfillment of the requirement for the degree
of Bachelor of Technology

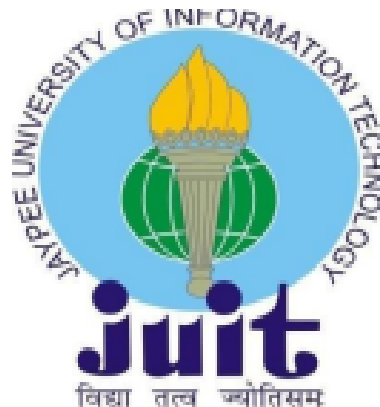
in

Computer Science and Engineering

By

Muskan Thakur (181208)

to



Department of Computer Science & Engineering and Information Technology
Jaypee University of Information Technology Waknaghat, Solan-173234,
Himachal Pradesh

DECLARATION

I hereby declare that this project has been carried out at Paymentus Corporation, Delhi from Feb, 2021 to June, 2021. I also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

Ms. Himanshi Chawla

Software Developer

Paymentus Corporation, Delhi

A handwritten signature in black ink that reads "Muskan". The signature is written in a cursive style with a large initial 'M' and a long horizontal stroke.

Submitted by:

Muskan Thakur 181208

Computer Science & Engineering Department

Jaypee University of Information Technology

CERTIFICATE

This is to certify that the work which is being presented in the “Internship report” is in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science And Engineering and submitted to the Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by Muskan Thakur(181208) during the period from February 2022 to June 2022 under the supervision of Ms Himanshi Chawla, Paymentus Corporation, Delhi.

Muskan Thakur (181208)

The above statement made is correct to the best of my knowledge.

Himanshi Chawla

Ms. Himanshi Chawla

Software Developer

Paymentus Corporation, Delhi

ACKNOWLEDGEMENT

First and foremost, I want to express my heartiest gratitude to Ms. Himanshi Chawla who mentored me and assisted me throughout my internship period. Her endless patience, scholarly guidance, continual encouragement and constructive criticism at all stages have made it possible to complete this internship successfully. .

I also would like to thank all the people that worked along with me at Paxcom with their patience and openness. I offer Jaypee University of Technology my deepest gratitude for giving such a unique chance for students to widen their understanding of how the actual world in the field of computer engineering operates.

I would also generously welcome each of those individuals who have helped me straightforwardly or in a roundabout way to make this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

Muskan Thakur

TABLE OF CONTENT

			Page No
		Declaration	I
		Certificate	II
		Acknowledgment	III
		Abstract	IV
1.	Introduction		
	1.1	Introduction	7
	1.2	Problem Statement	8
	1.3	Internship Objectives	8
	1.4	Limitation of Report	9
	1.5	Source of Data Collection	9
2.	Company Overview		
	2.1	About the Organization	10
	2.2	Paymentus	12
3.	System Development		
	3.1	Tools & Technologies Used	13
	3.2	Feasibility Study	31
	3.3	Requirement for Hardware and Software	32

4.	Activity Analysis		
	4.1	Roles and Responsibility	33
	4.2	Description of Project	34
	4.3	Why Task Management Software	34
	4.4	Features of Software	35
	4.5	Overview of Functionalities	36
	4.6	Team	37
	4.7	Project	38
	4.8	User Interface Design	39
	4.9	Code Snippets	44
4.	Conclusion		
	4.1	Conclusion	48
5.	References		50

LIST OF FIGURES

Fig No.	Figure Name	Page No.
1	Organization Services	11
2.	Organization Overview	11
3.	Package.lock.json File	16
4.	Package.json File	16
5.	How Redux Works	18
6.	States and Actions in Redux Store	18
7.	Single Thread with Event Loop	23
8.	Testing in Postman	25
9.	Authorization Header in Postman	25
10.	Git Workflow	27
11.	Local and Remote Repository	27
12.	Login Page	39
13	Sign up Page	40
14	Tasks SideBar	41
15	User Profile	43
16	App Component	44
17	Folder Structure Backend	44
18	Protected Routes	45
19	Storing applications state in Redux store	45
20	List of routes	46

21	Folder Structure Frontend	46
22	Components	47

ABSTRACT

The internship report covers both academic and non-academic initiatives done during the internship. With an emphasis on the intern's accomplishments, the goal of this paper is to identify and describe the research undertaken, tasks completed, and experience gained.

Project management is a way to end a life cycle of a project in small steps. Multiple repetitions or incremental stages leading to project completion form a repetitive or dynamic life cycle. Repetition methods are often used in software development projects to increase speed and flexibility as they allow a person to change as he travels rather than following a line path. A repetitive or quick approach aims to provide benefits throughout the process rather than just the conclusion. In a dynamic endeavor, trust, flexibility, empowerment, and cooperation must be fundamental values and values. Scrum is a core framework used by many fast-moving teams, and consists of four main processes and functions, running plans, daily meeting, demo sprint and sprint viewing. Customer happiness is one of the most important parts of a business today. Because of its simplicity and flexibility, rapid project management has made this much easier.

The projects undoubtedly aided by deepening practical knowledge. The research and development initiatives were very beneficial in broadening perspectives on various web application development frameworks.

Chapter 01:- INTRODUCTION

1.1 Introduction

I worked at Paymentus India Pvt Ltd at the Delhi branch. I worked with a team and was able to grasp their organizational structure and pattern. I also learnt soft skills like working with a team and communicating within a corporate environment. I was able to properly comprehend several coding paradigms employed by a corporation to develop its application. I gained a comprehensive understanding of some of the company's current and prospective products.

The company's working culture is fantastic. Working there has been a great experience for me. Paxcom has a good work-life balance. Despite the fact that I didn't get to spend much time in the firm office and had to work from home due to the coronavirus pandemic, I enjoyed the company's office meetings. Even though I worked from home, communication with my team was excellent thanks to numerous meeting platforms such as Zoom meetings and Google Meets.

My internship has a lot of things I like about it: the experience/information I've gained, the balance of fun and serious events, my manager and coworkers, and, interestingly, the group of people I interacted with during my mid-day breaks. In any case, if I had to put it on something, what I enjoy best is how this opportunity genuinely opened the door to my profession. The most impressive aspect of this temporary job experience is that it successfully surmounted any barriers between acquiring new skills and putting them into practice.

1.2 Problem Statement

Project managers can pick from a variety of project management approaches to guide their efforts. There are several options available on the market. However, all of the solutions on the market have some flaws and features that aren't used by everyone. To address these issues, an application was developed that is simple to use and allows easy collaboration among teams without compromising with their work.

1.3 Internship Objectives

- Internship is often thought of as a way for college students to gain experience in a particular field.
- Internship training, on the other hand, can help many people gain real knowledge and improve their skills.
- The purpose of this position should be to emphasize your academic ability and your desire to learn more.
- Internships are used in a variety of fields, including construction, engineering, health, economics, and marketing. Internships are designed to give people practical work experience.
- Internships are a great way to develop one's skills and abilities.

1.4 Limitation of Report

There were several limits in acquiring, processing, and representing information. Regardless of the limitations, a thorough report is produced. The following are the report's overall limitations:

- Because some information was assumed to be confidential, there were some restrictions on disclosing it. Some material in the study was withheld due to organizational confidentiality.
- Some of the secondary sources' data was not organized consistently.

1.5 Source of Data Collection

Data gathering processes were followed to make the report as comprehensive as attainable. Information for this research was acquired from both primary and secondary sources.

1.5.1 Primary Source : Several face-to-face interviews with officials from various departments were conducted for primary data. They provided extremely valuable information for this report.

1.5.2 Secondary Source : Several face-to-face interviews with officials from various departments were conducted

Chapter 02:- COMPANY OVERVIEW

2.1 About the Organization

Paxcom is a SAAS-based platform that offers ecommerce automation solutions, including the creation and management of estores (or marketplaces), extensive data analysis, exponential revenue development, managed warehouses and supply chains, and customer satisfaction. Paxcom's services and technologies can be adjusted to meet any demand, whether it's troubleshooting or a comprehensive strategy. Everything Paxcom produces is driven by smart data analysis and analytics, which allows for informed decisions and the refinement of product plans.

- **Channel intelligence:** Paxcom's dynamic KPI dashboards provide on-demand complete analytics with insights into sales, competition, category, and promotion. Paxcom's patented technology can collect and analyze publically available data across different platforms and geographies, allowing users to make better educated and qualified decisions.
- **Strategy & Execution:** Paxcom's team of eCommerce professionals, data scientists, business & data analysts, content writers, and creative designers can assist one with product pricing, promotion, and placement on top eCommerce sites.
- **Campaign & AMS Management:** Paxcom's dedicated staff can help brands design, execute, analyze, and optimize search and display (CPC and CPM) campaigns. Paxcom has a lot of experience with Amazon AMS promotions, Flipkart PLA ads, and other materials that can support one's customers throughout their buying process.
- **Omni-channel Solutions:** Paxcom simplifies warehouse management and order fulfillment from beginning to end. To help one track inventory across channels, including eCommerce, physical stores, and distributors, Paxcom solution effortlessly interacts with popular logistics providers and marketplaces like Amazon and Flipkart.
- **Marketplace Seller Solutions:** Order fulfillment, inventory management, and account reconciliation are all improved using Paxcom. Its extensive digital features can help one automatically reconcile sales and refunds, as well as identify additional charges, which is important for payment analysis, consolidation, and dispute resolution.

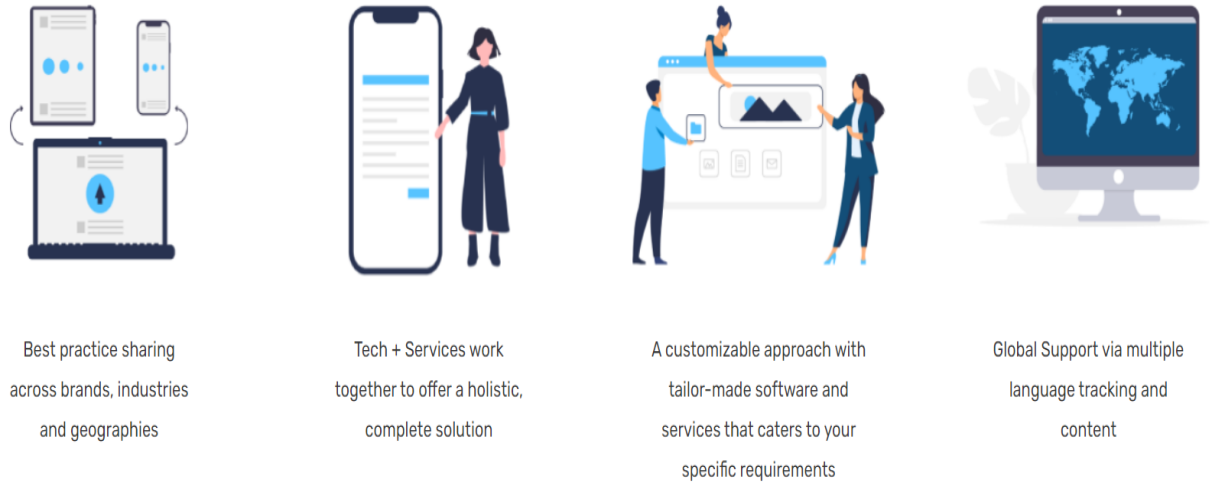


Fig 1. Organization Services



Fig 2. Organization Overview

2.2 Paymentus

Paymentus was founded in 2004 as a result of a desire to change the way bills are paid. Paymentus has grown to become the premier paperless electronic billing and payment solution on the market, with over 1,300 clients including some of North America's major billers.

We understand that in order to maintain our solutions current and relevant, we need employees that have the know-how, motivation, and aptitude for creating a fantastic client experience. Our dedicated, innovative staff transformed an idea into a secure, SAAS-based Customer Engagement and Payment Platform, allowing direct-bill companies to create a consistent customer experience while increasing adoption of cost-effective electronic billing and payment services.

Paymentus has continually strived to produce better, faster, more secure, and cost-effective billing and payment solutions, having been named one of Deloitte's fastest growing North American companies in 2011, 2013, 2014, and 2016. In both solutions and service, we are always looking for ways to provide more value to our customers.

It's what has propelled us forward in the last decade. When our clients prosper, we succeed. They prosper when their client ties are strengthened and their customers use these cost-cutting technological services in large numbers.

Chapter 03:- SYSTEM DEVELOPMENT

3.1 Tools & Technologies Used

3.1.1) ReactJs

ReactJS is a JavaScript toolkit for creating reusable, visually appealing, fast, and flexible visual components. It is an open source, front-end library that works specifically with the application overview. Developed and managed by Facebook, and later used in popular programs such as WhatsApp and Instagram.

The main goal of ReactJS is to create User Interfaces (UI) which increases the speed of the application. Uses a visual DOM (JavaScript object) to speed up the process. In JavaScript, the visual DOM is faster than the normal DOM. ReactJS can be applied to client and server, as well as other components. It uses component and data patterns to improve readability while also helping with app optimization.

ReactJS is best known for its high performance. This feature sets us apart from other segments in the market today. This is because it controls the visual DOM. DOM is a cross-platform computer programming API for HTML, XML, and XHTML. DOM contains only internal memory. As a result, we did not write directly to DOM when creating the component. Instead, we are creating real-time components that will be converted to DOM, resulting in smoother and faster performance.

3.1.2) Express

Express is a Node.js web application framework that provides a full set of features for developing online and mobile apps. It makes it easy to quickly construct Node-based Web apps. The Express framework's key features are given below.

- Allows middlewares to respond to incoming HTTP requests.
- A routing table is defined for doing various actions based on HTTP Method and URL.
- Allows you to dynamically render HTML pages by passing variables to templates.

Web frameworks are often defined using the terms "thoughtful" and "unintentional."

Ideas frames have strong emotions about the “right” way to do the job. They usually help with rapid growth in a particular area as the proper way to accomplish something is often well understood and well written (fixing problems of some kind). They can, however, be less flexible when faced with problems outside of their core area, as they have fewer opportunities for components and processes.

Non-discriminatory structures, on the other hand, have very limited limits on how you link components to achieve a goal, or what components should be used.

They make it easier for engineers to use the best work tools, but at the expense of having to track those parts yourself.

Express is unprejudiced. In the request handling chain, you can put practically any compatible middleware in almost any sequence you want. The software can be organized in a single file or several files, with any directory structure. You may occasionally feel as if you have too many options!

3.1.3) NPM

Npm is two things: first, it's an online repository for open-source Node.js projects; second, it's a command-line interface for dealing with that repository, which aids with package installation, version control, and dependency management. There are several Node.js modules and apps available on npm, with more being updated on a daily basis. In addition to simple downloads, npm also handles versioning, allowing you to specify any exact package version or need a version higher or lower than what you require. Frequently, you'll discover that a library is only compatible with another library's major release. Or a bug in the most recent release of a library is causing a problem. Specifying an explicit library version also aids in keeping everyone on the same exact version of a package, ensuring that the entire team runs the same version until the package is updated. Versioning is quite useful in all of these situations, and npm adheres to the semantic versioning (semver) standard. Two files are generated after we write the command “npm install”

3.1.3.1)package.json

- It contains the project's basic information.
- It is required for all projects.
- It keeps track of crucial project information.
- It includes information such as the author's name, description, script, and dependencies.

3.1.3.2)package.lock.json

- It specifies the exact tree that was created in order for later installations to have the same tree.
- It is generated automatically when npm updates the node modules tree or package.json.
- It makes it possible for future developers to add the same dependencies to the project.
- It contains the project's name, dependencies, and locked version.

```

    "animate.css": "^4.1.1",
    "axios": "^0.21.1",
    "bootstrap": "^4.6.1",
    "bootstrap-icons": "^1.8.1",
    "crypto-js": "^4.0.0",
    "date-fns": "^2.24.0",
    "dompurify": "^2.3.6",
    "js-cookie": "^2.2.1",
    "jsonwebtoken": "^8.5.1",
    "lodash": "^4.17.21",
    "make-server": "^1.0.0",
    "material-icons": "^0.7.0",
    "node-sass": "^6.0.1",
    "quill-image-compress": "^1.2.15",
    "quill-mention": "^3.0.4",
    "rc-color-picker": "^1.2.6",
    "re-resizable": "^6.9.1",
    "react": "^17.0.1",
    "react-bootstrap": "^1.5.2",
    "react-color": "^2.19.3",
    "react-dom": "^17.0.1",
    "react-draggable": "^4.4.4",
    "react-file-icon": "^1.1.0",
    "react-google-login": "^5.2.2",
    "react-pro-sidebar": "^0.6.0",
    "react-quill": "^1.3.5",
    "react-redux": "^7.2.2",
    "react-router": "^5.2.0",
    "react-router-dom": "^5.2.0",
    "react-scripts": "4.0.3",
    "react-toggle-button": "^2.2.0",
    "redux": "^4.0.5",
    "redux-persist": "^6.0.0",

```

Fig 3. Package.lock.json File

```

"bootstrap": "^4.6.1",
"bootstrap-icons": "^1.8.1",
"crypto-js": "^4.0.0",
"date-fns": "^2.24.0",
"dompurify": "^2.3.6",
"js-cookie": "^2.2.1",
"jsonwebtoken": "^8.5.1",
"lodash": "^4.17.21",
"make-server": "^1.0.0",
"material-icons": "^0.7.0",
"node-sass": "^5.0.0",
"quill-image-compress": "^1.2.15",
"quill-mention": "^3.0.4",
"rc-color-picker": "^1.2.6",
"re-resizable": "^6.9.1",
"react": "^17.0.1",
"react-bootstrap": "^1.5.2",
"react-color": "^2.19.3",
"react-dom": "^17.0.1",
"react-draggable": "^4.4.4",
"react-file-icon": "^1.1.0",
"react-google-login": "^5.2.2",
"react-pro-sidebar": "^0.6.0",
"react-quill": "^1.3.5",
"react-redux": "^7.2.2",
"react-router": "^5.2.0",
"react-router-dom": "^5.2.0",
"react-scripts": "4.0.3",
"react-toggle-button": "^2.2.0",
"redux": "^4.0.5",
"redux-persist": "^6.0.0",

```

Fig 4. Package.json File

3.1.4) Redux

- Predictable

Redux enables you to create applications that act reliably across several environments (client, server, and native) and are simple to test.

- Centralized

The option to centralize your application's information and logic allows for powerful features like undo/redo, state persistence, and more.

- Debuggable

The Redux DevTools make it simple to see when, when, why, and how the state of your app changed. The architecture of Redux allows you to log changes, employ "time-travel debugging," and even send full error reports to a server.

- Flexible

Redux is a flexible UI framework that works with any UI layer and has a big ecosystem of addons to meet your specific needs.

Redux is basically a store for storing the state of your app's variables. Redux establishes a set of processes and procedures for interacting with the store, ensuring that components do not change or read the store at random. The bank is similar. It does not imply that just because you have money in the bank, you can walk in at any time, open the vault, and withdraw cash. To withdraw money, you must follow certain processes.

In a nutshell, Redux is a means to manage "state," or a cache or storage that can be accessed in a systematic manner by all components. It can only be accessible via a "Reducer" and "Actions."

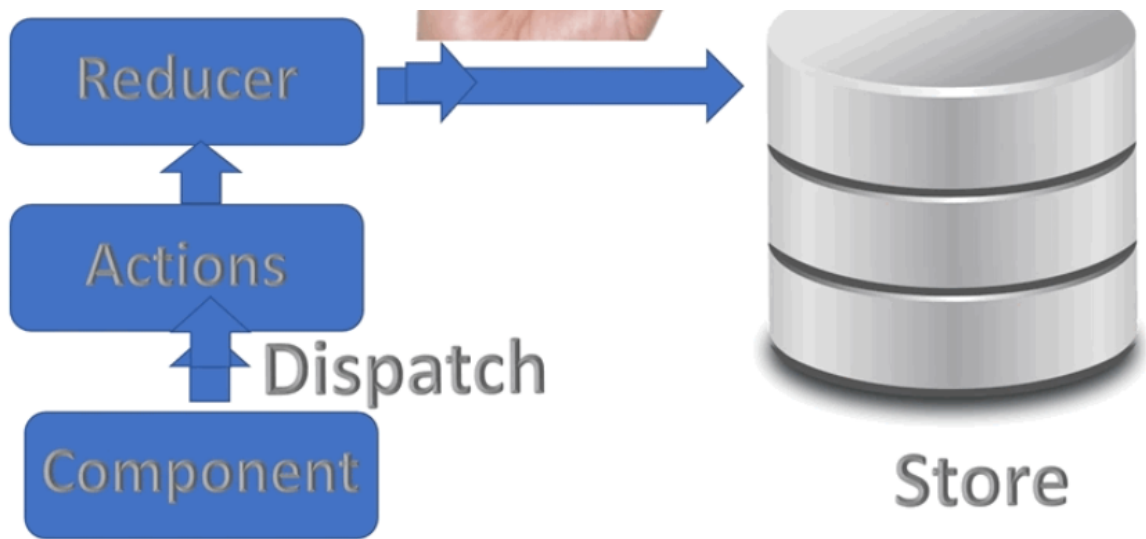


Fig 5. How Redux Works

SET_CURR_USER	+30:00.37
RESET_STORE	+30:00.00
persist/PERSIST	+00:00.00
persist/REHYDRATE	+30:00.22
SHOW_LOADER	+30:28.41
HIDE_LOADER	+30:00.03
ALERT_NOTIFICATION	+30:00.01
ALERT_NOTIFICATION	+30:03.03
SHOW_LOADER	+30:12.79
HIDE_LOADER	+30:00.03
ALERT_NOTIFICATION	+30:00.00
ALERT_NOTIFICATION	+30:03.02

Tree Chart Raw

- ▶ **auth** (pin): { isAuthenticated: false, signupSuccess: false, passwordUpdateSuccess: false, ... }
- ▶ **company** (pin): { userReqBody: [], confirmCompany: false, organization: [] }
- ▶ **loader** (pin): { loading: false }
- allQueues** (pin): []
- companyUsers** (pin): []
- allProjects** (pin): []
- ▶ **projectForm** (pin): { privacyList: [], view: [], status: [] }
- ▶ **team** (pin): { projectMembers: [], searchText: "", roleId: "" }
- ▶ **queueProjects** (pin): { projects: [], searchText: "" }
- ▶ **storeActivityLog** (pin): { activityData: [], users: [], actionTypes: [] }
- ▶ **tasks** (pin): { allTasks: [], taskLogs: [], cardCounts: {}, ... }
- myBodyConfig** (pin): []
- ▶ **comments** (pin): { allComments: [], allReplies: [], isReply: false, ... }
- workflow** (pin): {}

ALERT_NOTIFICATION (13)

Fig 6. States and Actions in Redux Store

3.1.5) Html

The acronym for Hypertext Markup Language (Html) is Hypertext Markup Language. HTML is a markup language that specifies how a document will be displayed on a computer screen. HTML pages can include audio, moving visuals, lists, real data, and java documents, and they can be created as simple text or complex multimedia. HTML pages are shown by web browsers, which are applications that can travel across a network and display a variety of data. The most extensively used web publication format is HTML. Allows the author to include not just text, but also text titles, lists, and tables, as well as still images, video, and audio in their writing. The student's PC can be used to retrieve information. HTML web pages can also be used to enter data and as the front end of a commercial transaction.

3.1.6) JSX

React understands that rendering logic is closely tied to other UI logic like event handling, state changes over time, and data preparation for display.

Rather than artificially splitting technology by putting markup and code in different files, React separates concerns through loosely linked pieces called "components" that include both. We'll go back to components later, but if you're still afraid to utilize JS for markup, this lecture could change your mind.

Although React does not need the usage of JSX, most people find it handy as a visual assistance when working with UI in JavaScript code. React may now provide more detailed error and warning messages.

3.1.7) CSS

HTML is used to display a document that has been created using the tag language. Cascading style sheets allow you to change the design of multiple web pages at once, saving you time and effort. External style sheets are piled on top of one other. CSS style sheet files are used to define the layouts, textures, and changes in how different devices and screen sizes are displayed on websites. The style description is usually stored in external.css files. CSS is separated into three groups:

Inline CSS is a type of CSS that is used in the body section and is associated with the inline CSS element. Within the HTML tag, the style property is utilized to offer this type of style.

When a single HTML document requires a unique style, internal CSS (also known as featured CSS) is employed. CSS rules must be specified in the HTML file's header section, which is CSS integrated within the HTML page.

External CSS: This folder provides a separate CSS file that only uses tag symbols to store style assets. A separate file is used to store the CSS assets. To link a CSS extension to an HTML text, use the link tag. It specifies that the style must be specified only once per item and that it will be applied to web pages.

3.1.8) MongoDB

MongoDB is an open source document source that stores data using a flexible schema and built on a horizontal scale-out structure. MongoDB, launched in 2007, has a large community of engineers worldwide.

Instead of row tables or columns similar to SQL domains, each item on the MongoDB website is a document rendered in BSON, a dual representation of data. Apps may receive this data in the JSON format.

3.1.9) Why MongoDB

MongoDB is built on a scale-out structure, developed by the developers' reputation of all kinds of smart systems with flexible data schemes. As a document database, MongoDB makes it easy for developers to store organized and unstructured data. Saves documents in a format similar to JSON. Because developers do not have to worry about standardizing data, many current programming languages automatically convert this format into native objects. MongoDB can also manage large volumes of data and scale them up or down to manage them. MongoDB is designed for developers of online and commercial applications that need to innovate quickly while still growing well.

3.1.10) Javascript

JavaScript is a portable scripting language. JavaScript is a scripting language for building in-app apps. JavaScript is a platform that is open source. A tiny server connection is JavaScript. Because no such functionality is available. JavaScript cannot be utilized for communication. When the user submits the form, the JavaScript HTML code integration is completed, and all entries are enabled, they are posted to the Web server. Curly-bracket syntax, strong typing, and object-oriented orientation are all features. The web application's language is JavaScript. JavaScript is the most widely used programming language on the planet. Event-driven, functional, and essential programming paradigms are used in JavaScript.

3.1.11) Why Javascript?

JavaScript is used because it works on a wide range of platforms, including Windows, Mac OS X, Linux, Raspberry Pi, and more. To get started, one only needs a small amount of money. JavaScript can be used in a variety of applications and can mix the function well with other languages. It includes a number of libraries that make it simple for users to navigate. Server load is reduced while using JS since the burden on the website server is lessened on the client side. API routing, fast-paced expansion, internal usage, developing various domains, ease of implementation, regulation, tools, and so on are all basic concepts in JavaScript. It has the ability to design complex user interfaces.

3.1.12) Nodejs

The word "Node.js" does not relate to a single file in this context; rather, it is an extension of the JavaScript code file extension. Node.js is a cross-platform, open-source JavaScript environment that runs behind the V8 engine and enables us to use JavaScript code without the need for a web browser. Node.js allows developers to construct command-line tools and server-side scripts that generate dynamic web page content before delivering it to a user's browser through JavaScript. Node.js allows developers to construct command-line tools and server-side scripts that generate dynamic web page content before delivering it to a user's browser through JavaScript. Node.js allows developers to construct command-line tools and server-side scripts that generate dynamic web page content before delivering it to a user's browser through JavaScript. As a result, Node.js symbolizes the "JavaScript Everywhere paradigm," which entails creating online applications using a single programming language. It is a fantastic alternative for constructing real-time data storage applications because it supports numerous server-side languages and client-side text using a non-blocking I/O mechanism. Node.js is a powerful and fast programming language. It's a fantastic option for making messaging or chat apps. It also helps with the building of huge apps and e-commerce sites that require quick processing. Because of its single-threaded nature, Node.js is better suited for nonrestrictive, event-driven servers. Node.js is a programming language for creating ordinary web pages as well as background API services. It's designed for push-based, real-time thinking. As a result, communicating with members of the development team as well as stakeholders is crucial. Operational requirements, in general, characterize how well a system operates in specific conditions.

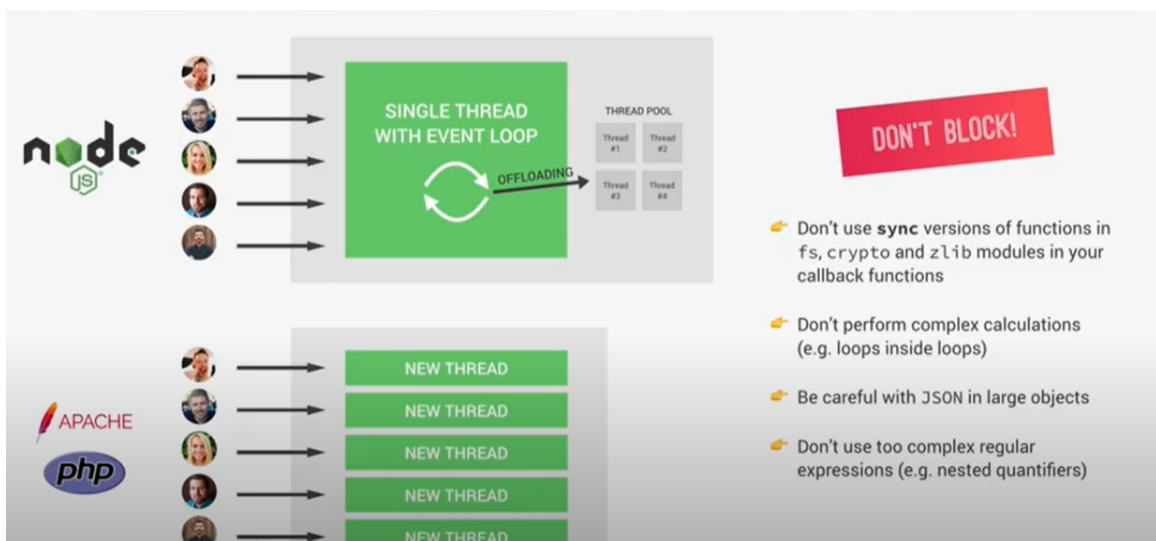


Fig 7. Single Thread with Event Loop

3.1.13) Postman

One of the most widely used software testing tools for API testing is Postman. Developers may simply create, test, share, and document APIs with this tool. Postman is a stand-alone API (Application Programming Interface) testing platform for creating, testing, designing, modifying, and documenting APIs. It's a straightforward GUI for sending and viewing HTTP requests and answers.

For testing purposes, Postman eliminates the requirement to develop HTTP client network code. Instead, we create groups of tests and let Postman interact with the API.

Almost any functionality that a developer would require is integrated in this programme. This tool can make GET, POST, PUT, and PATCH HTTP queries, as well as translate the API into code for languages like JavaScript and Python.

3.1.14) Why Postman?

Postman is built around a set of powerful technologies that are extremely easy to use. Postman has become a useful tool for more than 8 million users.

Postman is used for the following reasons:

- Postman may be used from anywhere after installing it on the smartphone and logging into the account.
- Postman allows you to construct collections for your API calls. Each set can generate several requests and subfolders. It will help with test suite organization.
- To test checkpoints, every API call should contain verification of a successful HTTP response status.
- Automation Testing-Tests can be run in numerous repetitions or iterations using the Collection Runner or Newman, saving time for repeated tests.

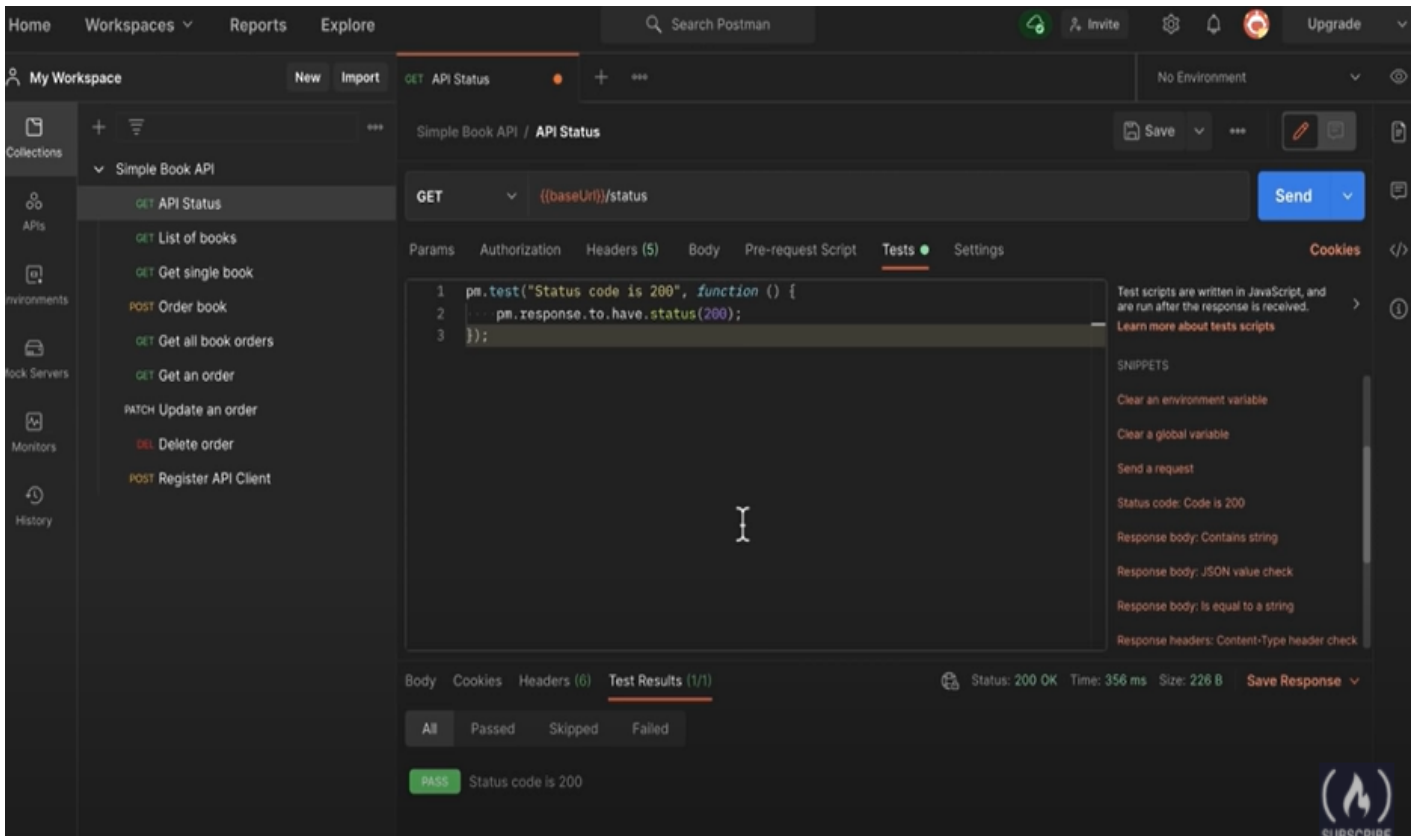


Fig 8. Testing in Postman

<input checked="" type="checkbox"/>	Postman-Token	<calculated when request is sent>	
<input checked="" type="checkbox"/>	Content-Type	application/json	
<input checked="" type="checkbox"/>	Content-Length	<calculated when request is sent>	
<input checked="" type="checkbox"/>	Host	<calculated when request is sent>	
<input checked="" type="checkbox"/>	User-Agent	PostmanRuntime/7.29.0	
<input checked="" type="checkbox"/>	Accept	*/*	
<input checked="" type="checkbox"/>	Accept-Encoding	gzip, deflate, br	
<input checked="" type="checkbox"/>	Connection	keep-alive	
<input checked="" type="checkbox"/>	Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjEwIiwiaWF0IjoiMTYxMjM0NTY3In0.eyJpZCI6IjEwIiwiaWF0IjoiMTYxMjM0NTY3In0.	
	Key	Value	Description

Fig 9. Authorization Header in Postman

3.1.15) Git

Git is a widely used version control system. Git records the changes you make to files so you can see what you did and go back to previous versions if needed. Git also facilitates collaboration by allowing minority changes to be integrated into a single source.

Git is a locally installed app. Your computer stores your information and its history. You can also use web hosts to save a copy of files and their update history (like GitHub or Bitbucket). If you have a central site where you can publish your changes and get changes from others, you can easily share with other developers.

3.1.15.1) Git Repository

Git repository contains all project files and all review history (or repo for short). You will be prompted by Git to convert the standard collection of files (such as a web root folder) into a repository. This creates an `.git` sub-directory containing all the Git metadata needed to track changes.

Files and folders that begin with a period (`.`) are hidden on Unix-based operating systems like macOS, so you won't see the `.git` folder in the macOS Finder unless you enable hidden files, but it's there! Some coding editors may be

3.1.15.2) Stage and Commit Files

Think of Git as a tool for tracking file changes. So, how can we tell Git to track our changes? An obligation is a file record or a set of edited files.

Before we can commit, we must inform Git of the files we intend to create. This is called a stage, and it uses an add-on command. Why are we forced to do this? Why not just create a file? Imagine working with two files, but only one complete one. You should not be forced to tie both files; only the righteous should give themselves up.

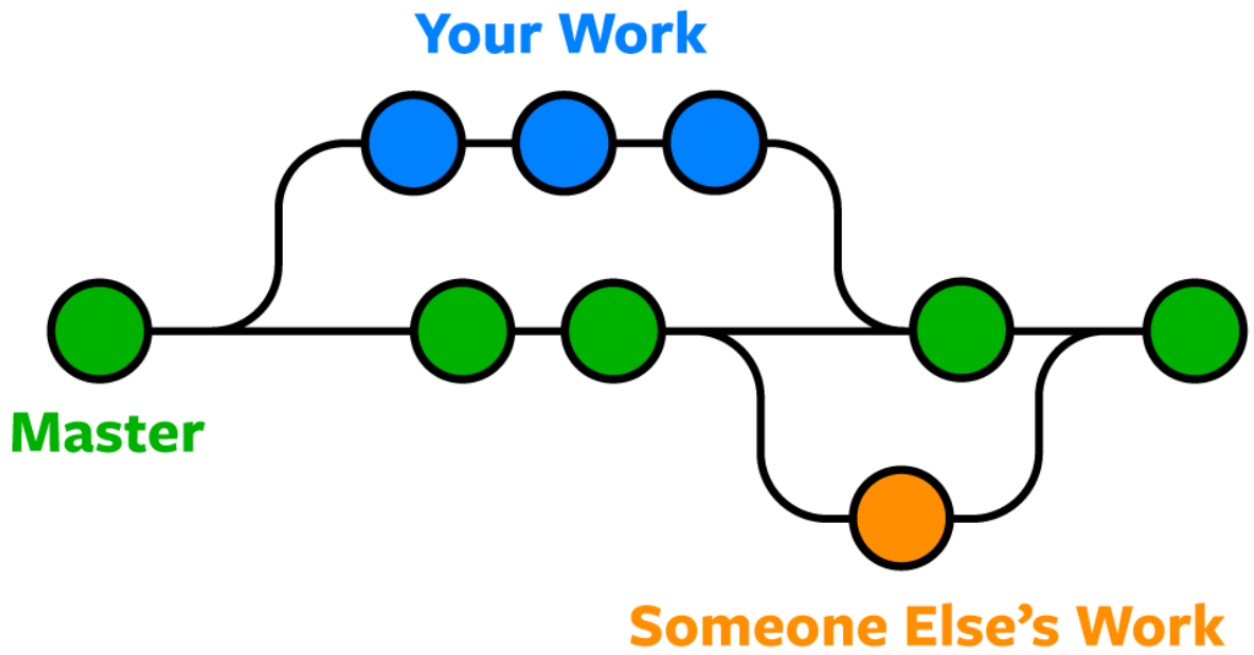


Fig 10. Git Workflow

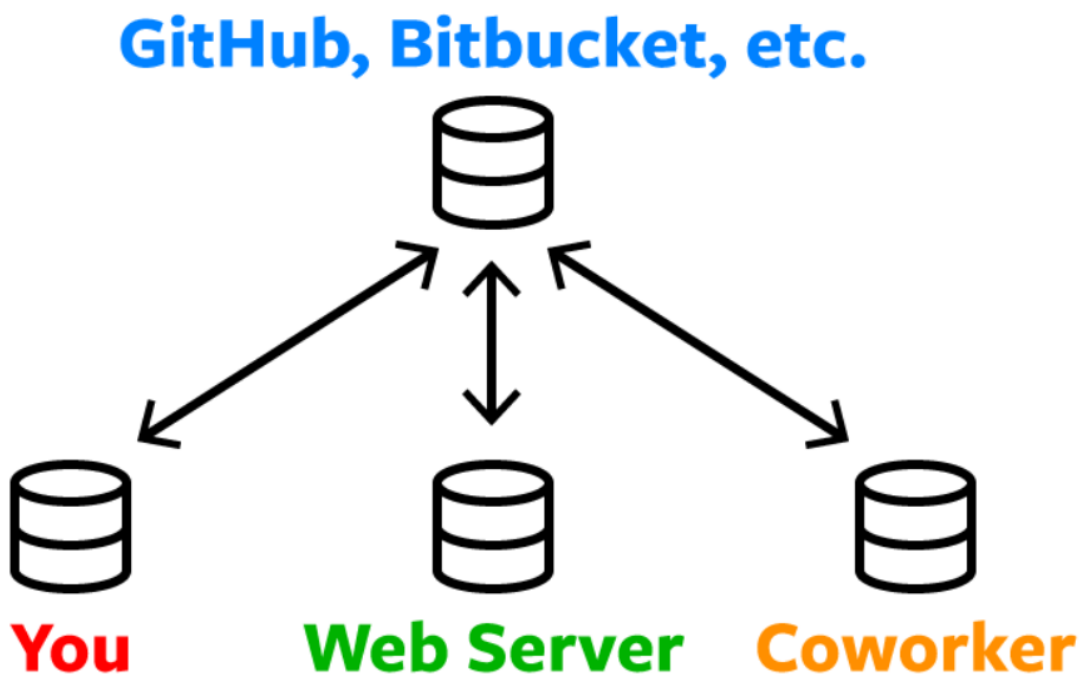


Fig 11. Local and Remote Repository

3.1.16) GitLab

GitLab is a github-like service that allows businesses to manage git repositories internally. It's a self-hosted Git-repository management solution that keeps user code private while allowing for easy deployment of updates. Gitlab is a service that allows you to access Git repositories from anywhere. The services offer extra capabilities to help manage the software development lifecycle in addition to hosting your code. These extra features include coordinating code sharing amongst multiple individuals, bug tracking, wiki space, and other social coding' tools.

3.1.16) Why GitLab ?

- GitLab offers free hosting for your (private) software projects.
- GitLab is a Git repository management tool.
- GitLab provides public and private repositories, as well as issue tracking and wikis, for free.
- GitLab is a user-friendly web interface layer built on top of Git that accelerates Git work.
- GitLab has its own Continuous Integration (CI) system for managing projects, as well as a user interface and other GitLab features.
- Users can use GitLab Community Edition to find out where their code is hosted on GitLab servers.
- GitLab offers an unlimited number of free private and public repositories.
- Instead of sharing the entire project, the Snippet section allows you to share a small portion of it.

3.1.17) MySQL

MySQL is an outstanding web-based management system (RDBMS) used by small and large businesses. MySQL AB, a Swedish company, is responsible for its development, marketing, and support. MySQL is becoming increasingly popular for many reasons.

3.1.18) Why MySQL

- MySQL is a free and open source website. As a result, there is no cost to use it.
- MySQL is a very powerful software in itself. It can handle many of the skills found in the most expensive and sophisticated website packages.
- MySQL uses the standard implementation of the widely used SQL data language.
- MySQL works with a variety of applications and languages, including PHP, PERL, C, C ++, JAVA, and more.
- MySQL is a fast web site that manages large data sets successfully.
- MySQL is best suited for PHP, the most common language for web development.
- MySQL is capable of handling up to 50 million rows of data on a table. The default table file size limit is 4GB, but you can extend the theoretical limit of 8 million gigabytes (as long as your operating system can support it) (TB).
- MySQL is flexible. Thanks to the open source GPL license, MySQL software can be customized to meet individual needs.

3.1.19) TypeScript

TypeScript is a tightly typed and open source Object Oriented programming language. It's a Strict Superset of JavaScript, which means that anything that can be done in JavaScript can also be done in TypeScript, with the option of adding more capabilities (every existing JavaScript Code is a valid TypeScript Code). It's easy to integrate TypeScript code into JavaScript projects because it's transformed to JavaScript code. It might result in more resilient software for a large-scale project, while still being deployable where a standard JavaScript application would run. It will not ensure that your software is bug-free. However, it can help you avoid a lot of typing errors. Together with Clever IntelliSense.

3.1.20) Why Typescript ?

- JavaScript was designed to be a lightweight, easy-to-learn language that focused mostly on simple DOM manipulations, but standards evolved with time, and TypeScript was born to provide expanded functionality to JavaScript.
- Support for Classes and Objects is also one of the key reasons for its growing popularity, as it makes OOPS principles easier to grasp and use than native JavaScript's conventional prototype-based implementation.
- TypeScript's inclusion in popular JavaScript frameworks like Angular has further boosted its popularity.
- The essential building parts of your application are taken from JavaScript by Typescript. For the purpose of execution, all TypeScript code is translated to its JavaScript equivalent.

3.2) Feasibility Study

Feasibility studies assess the organization's performance, ability to meet user needs, and efficiency of resources. The purpose of research is whether it is possible to get an idea of the extent of the problem rather than solve it.

- **Economic feasibility**

The construction of this app is inexpensive. The organization did not have to spend a lot of money to set up a similar system that already existed. What remains to be done is to create an environment for development that is closely monitored. If we achieve this, we may be able to increase the usability of connected services. Even after development, the organization will not be able to invest extra money in it. As a result, the system is profitable.

- **Technical feasibility**

We can say with certainty that it is technically possible as gathering the necessary resources for system development and maintenance will be easier. All the resources needed to develop and maintain the system are accessible to the company, and we use them.

- **Probability feasibility**

We can state with certainty The availability of a product refers to its capacity to function. Some items may perform admirably throughout creation and usage, yet they may fail in use. It necessitates doing background checks on the necessary persons as well as testing their technical expertise. Data, updated information, and reports for generations are accurate and timely in the current system.

3.3) Requirement for Hardware and Software

System Configurations

Specification of software requirements may be adjusted at the end of the analysis function. As part of system engineering, comprehensive information definition, detailed performance description, system ethical representation, performance indicator and design issues, appropriate verification method, and other information related requirements are used to refine the function and performance assigned to it.

Software. Software Requirements:

- Operating system : Versions of Windows 7, 10, and above are supported..
 - Coding Language : JavaScript
 - Front-End : Visual Studio Code.
 - Data Base : MYSQL Workbench
-
- Hardware Requirements:
 - System : Pentium IV 2.4 GHz.
 - Hard Disk : 1TB
 - Ram : 4GB.

Chapter 04:- ACTIVITY ANALYSIS

4.1 Role and Responsibility

Various platforms and programming languages were learned during the internship term. The goal was to create a completely functional web-based system employing various tools and frameworks, such as the MERN stack.

To achieve that goal, this internship needed substantial background study on the MERN stack prior to understanding the system's actual requirements. The study was required not only to comprehend the issue under analysis, but also to develop solutions to existing difficulties, and putting the study's results into practice was an additional hurdle. Other key tasks carried out throughout the internship included actual implementations and team talks to examine the problem and come up with the best solution. Regular meetings with the mentor and team discussions helped me broaden my understanding of the existing system and problem context.

Software development refers to the process of generating software programmes as a whole, encompassing all stages of the system development life cycle. I studied and implemented the MERN stack and Typescript during my internship. And created a personal project using MERN, as well as working on a live project for the company that used technologies such as express, NodeJS, ReactJS, Kendo React, and MySQL.

4.2 Description of Project

During the internship, I learnt about many platforms and computer languages. The purpose was to develop an orderly method for recognising, monitoring, and controlling your and your team's activities. Task management entails:

- Keeping Track of task progress
- Assigning work within team
- Fixing deadlines
- Managing Time and work schedules

Effective task management assists people or organizations in achieving their goals by taking into account task planning, task prioritization, work automations, dependencies, and budgets. Task management is usually one of the essential features of good project management software.

4.3 Why Task Management Software

- It aids in the organization of tasks.
- It allows a team to work together more successfully.
- It aids in work prioritization and the project can be viewed from several viewpoints
- This project includes comprehensive integrations.
- Tracking Task Progress makes it simple to monitor where everyone is in the project and how an individual and their team are progressing toward their objectives.
- In the case that assistance is required responsive support is provided, the task management solution, provides speedy email responses
- It can be used by both novice and experienced project managers. It can be used by both code and non-code teams, and collaboration between departments is possible.

4.4 Features of Software

There are a number of features in the software like, setting up target and goals, notepad, dashboard, integrations. Few of the features are briefed below

- **Recurring Tasks :** Recurring Tasks can also be set up for projects. One may also adjust how often each Recurring Task appears by changing the time interval between them.
- **Priorities:** Priorities help you prioritize tasks when working on projects. Using a simple color-coded method, the team will have no issue selecting which tasks to prioritize and which to de-emphasize.
- **Reminders:** This project management software includes sophisticated Reminders that will keep track of one's deliverables at all times. For added convenience, one may choose where and when you receive these reminders.
- **Notifications:** It features a variety of notifications to keep users informed about the progress of their task. Users can also choose how frequently they get them.
- **View:** Allow the management tool to adapt to oneself rather than the other way around. This software provides users with multiple views to manage tasks in the way they want: Board view, List view and Dashboard view
- **Assigned Comments:** One can add comments to tasks using this project management software.

4.5 Overview of Functionalities

- Users will be able to safely log into the system due to Login. Only Google accounts will be given SSO-based login. Options for forgetting password and signing up in the login page.
- Homepage - Workspace of any user, it will allow customization based on widgets of 3 different types-task list, project list, user list
- Teams are found at the top of the programme, followed by projects, which are then followed by tasks, and finally sub-tasks.
- Users complete tasks, which can then be transferred/assigned to multiple teams, projects, and users.
- There is no owner for this collection of jobs.
- Concept-tasks will be assigned to the default assignee, after which another user may assign it to themselves, or the Team owner may allocate it to a user.
- Users can then begin working on the task.
- Teams can assist with task prioritization, distribution, and assignment among teams that share the effort.
- A task can be added to a Team by email, or it can be created/moved from other Teams in the organization.
- Admins can create and remove teams.
- There is no limit to the number of team owners that Admin can add or remove from the team.
- In the team Assignment, team owners can create projects.
- Users/Members who have access/have been added to the team can view all of the records and claim ownership/assign themselves to the ones they wish. (Assign yourself/rename your team) Project managers of projects within the teams will be managed by the Admin Team Owner/The person with "Manage" authorization privileges.
- The premise is the same as having a team, with the team managing many projects for various types of jobs.

4.6 Team

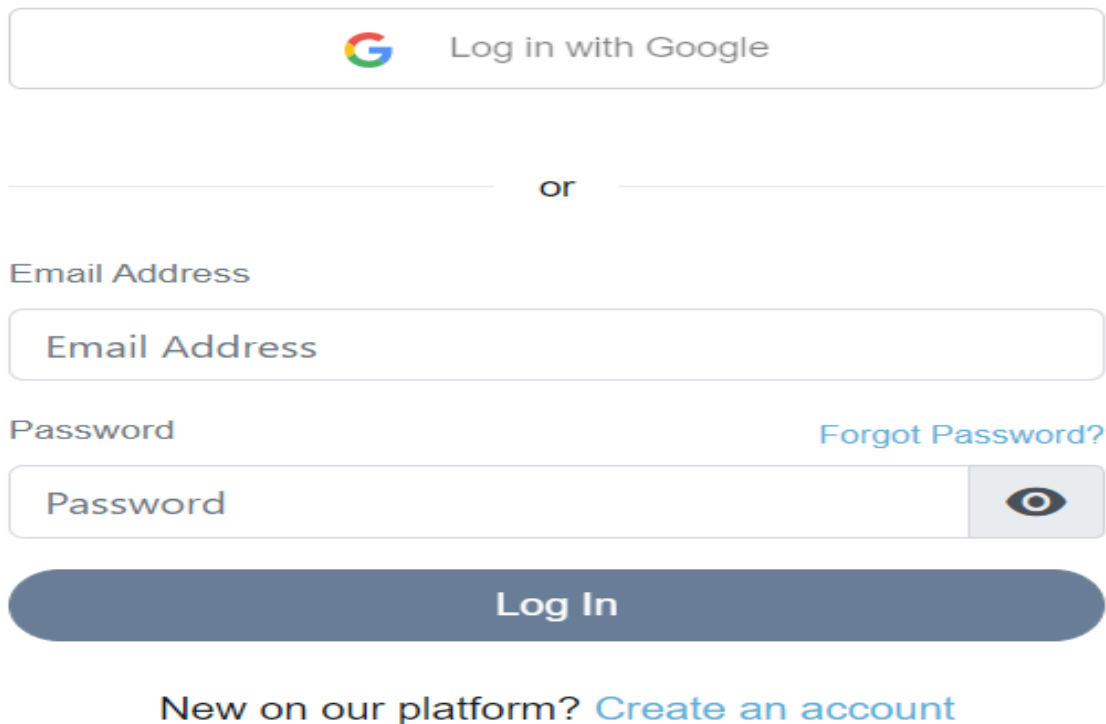
- A team lead can work on forming a team for the organization; a team can have numerous projects, and each Team will have a default project generated when it is formed. A team's members can collaborate and create projects to work on together.
- Each organization's teams should be divided into 4-5 major categories in the ideal situation (HR, Development, QA, Operations, Project management, Accounts etc.)
- According to the needs of the organization, the team lead can form multiple teams.
- If a user is joined to a project, he or she will automatically become a member of the task.
- When a user joins a team, he or she can give tasks to any project in the team, but he or she cannot directly access the other projects unless he or she is a member of the project.
- If you don't want to see a particular team on your teams list, use the move button to shift it to the secondary list.
- On this dashboard, you can see the overall number of teams as well as the total number of projects.
- This dashboard has a search feature that allows the user to quickly find the team he or she is looking for.
- A user can mark a team as primary or secondary based on the required level from the team dashboard.

4.7 Projects

- Each Team can have an unlimited number of projects, and each team member can create one.
- project according to their specifications
- This dashboard displays all of the active projects.
- This dashboard allows users to add, edit, and delete projects.
- A project can have several tasks, and each project member can view and edit all of the project's tasks.
- Each project has a Project Code that is used to identify the project's tasks. If a task is created under a project, the project code will appear in the task's header.
- A project can have several users; there is no limit to how many users can be in a project.
- The project owner has the ability to add new participants to the project.
- The user can also see all of the projects that are currently active or inactive.
- The search bar allows you to look up any project in the list.

4.8 User Interface Design

4.8.1 Login



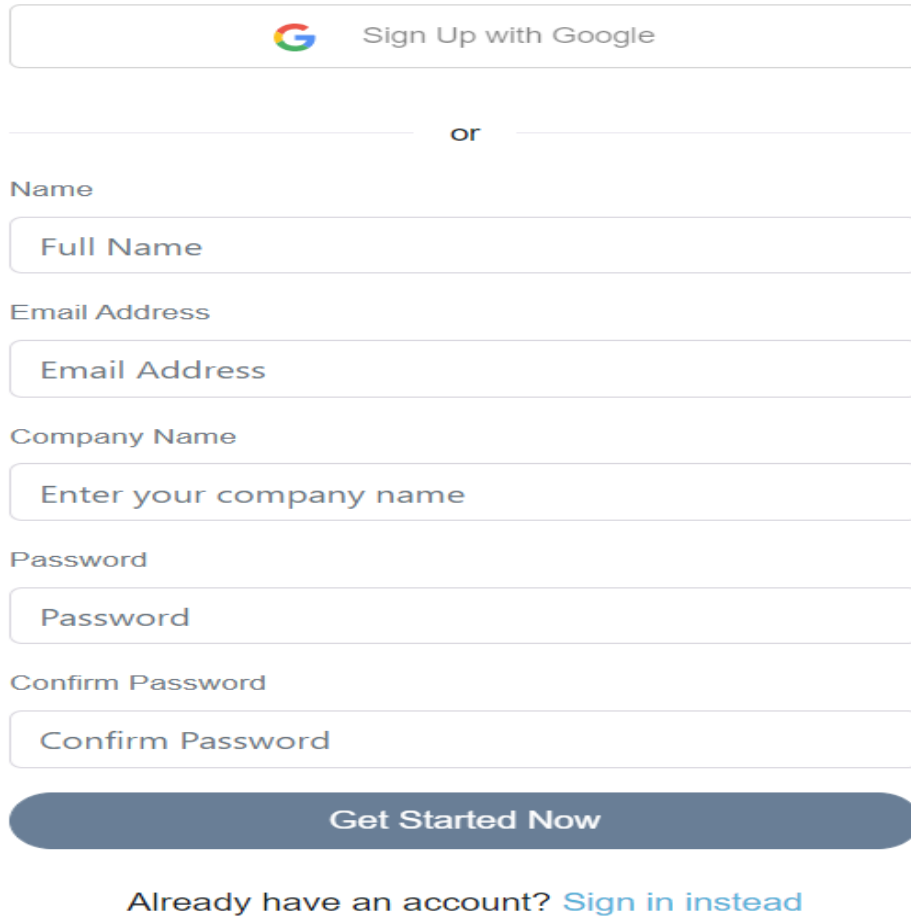
The image shows a login page with the following elements:

- A button with the Google logo and the text "Log in with Google".
- A horizontal line with the word "or" centered below it.
- A label "Email Address" above a text input field containing the placeholder text "Email Address".
- A label "Password" above a text input field containing the placeholder text "Password".
- A link "Forgot Password?" in blue text to the right of the password input field.
- A toggle button with an eye icon to the right of the password input field.
- A large, dark blue button with the text "Log In" centered on it.
- A link "New on our platform? Create an account" in blue text below the "Log In" button.

Fig 12. Login page

The Login Page is a page that everyone who visits the site will see. Registered users can utilize the sign-in page to login and access all of the services that their account enables them to. They will be logged in if they enter their username and password and click to submit the verified user's details. An error warning will show if they are incorrect.

4.8.2 Sign up



The image shows a sign-up form with the following elements:

- A button with the Google logo and the text "Sign Up with Google".
- A horizontal line with the word "or" in the center.
- A label "Name" above a text input field containing "Full Name".
- A label "Email Address" above a text input field containing "Email Address".
- A label "Company Name" above a text input field containing "Enter your company name".
- A label "Password" above a text input field containing "Password".
- A label "Confirm Password" above a text input field containing "Confirm Password".
- A dark blue button with rounded corners containing the text "Get Started Now".
- Text below the button: "Already have an account? [Sign in instead](#)".

Fig 13. Sign up Page

- If the name of the organization is known, anyone can join up for it immediately through SSO. The user must also provide the correct name of the organization. After signing up, the user must get permission from the organization's admin to utilize the platform.
- If the organization's name is incorrect or unknown: Following signup, the user will be informed that the organization does not exist in the system and that a new one will be created. The user will be redirected to the application's platform, where they will be granted admin access.

4.8.3 Task Sidebar

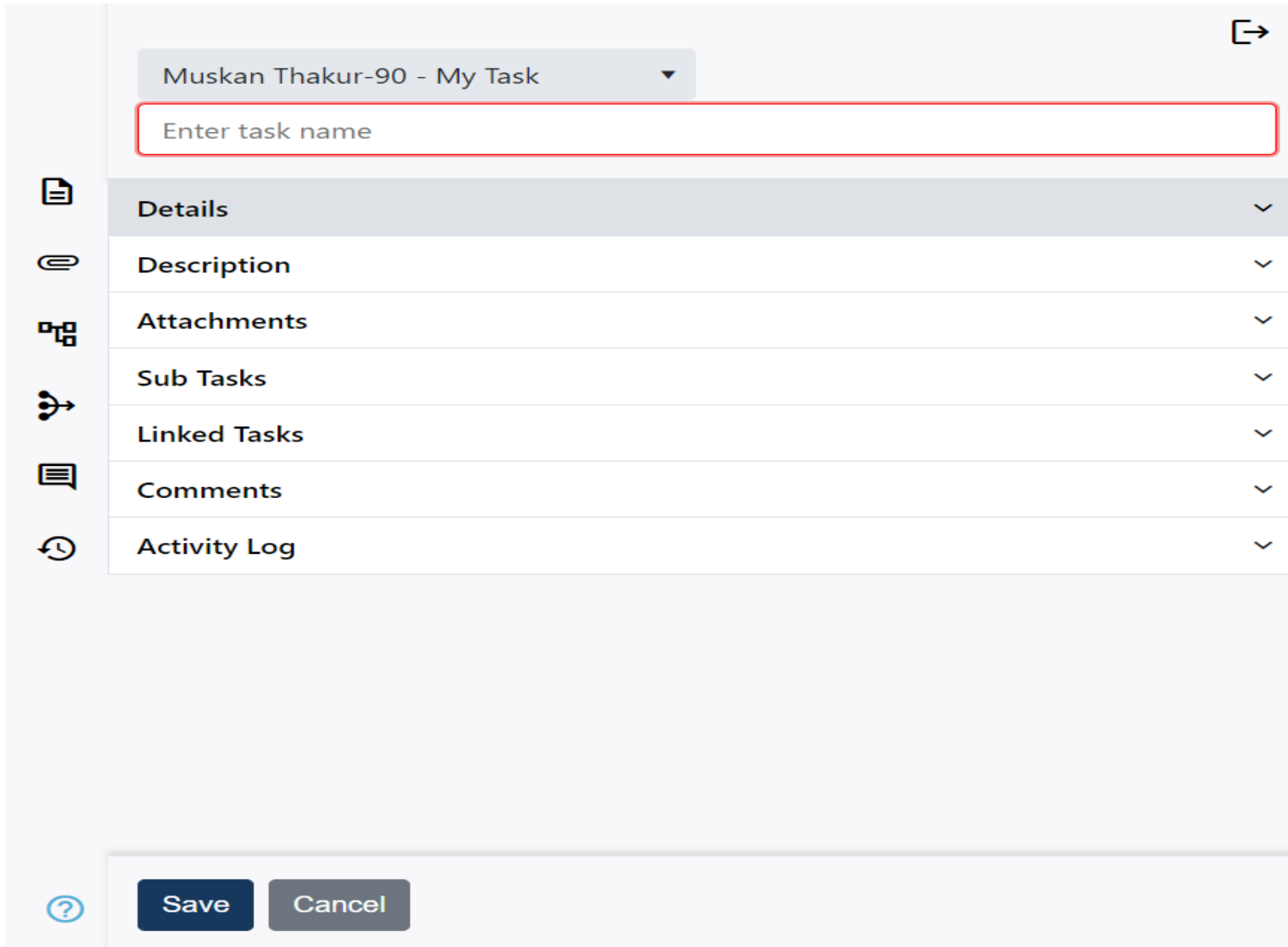



Fig 14. Task Assigning TaskSidebar

- My task is located on the home page's left side. It displays the tasks that the user has been allocated.
- The user's personal tasks are also included.
- This area also allows the user to assign tasks. On the top, you'll find the Help section and the Archive part.
- A new dialogue box appears after clicking Add task. The user can delegate tasks on his or her own behalf or on behalf of the teams to which he belongs. Tasks can be assigned to individuals, projects, or teams.
- The task can be delegated or passed on by the user/assigner.

- The user/assigner can opt to delegate a job, which will be assigned back to the assigner once the assignee has completed it (task will return). This also means that once the work is delegated, it will be locked for the assigner.
- The work can also be passed on to the assignee by the assigner. This prevents the work from appearing in his task portal and notifying him when it is completed.
- The user should give the task a name, select the current stage from the three options (To-Do, In Progress, and Completed), and select the task's priority from the options available (Low, Medium, High and Highest). The task's due date should also be specified; otherwise, the default due date will be used.
- We also have the option of adding tags to the task, which is not required. Tags are a means to filter a task so that it may be distinguished from others based on the department, priority, and so on.
- A single job might have many tags applied to it.
- In the description box, we can give a brief description of the task and include any attachments that are related to the task completion. The 'Sub Tasks' and 'Linked Tasks' features are still being worked on. We can leave comments on the task and also include media from the machines in the comment box.
- We have many alternatives in the upper right corner of the dialogue box, such as bookmarking this task, copying the task to the clipboard, deleting and archiving the task from the settings, the Help option, the maximize option, and the Collapse option.

4.8.4 User Profile

Profile Preferences My Project [↗](#)



[Upload an Image](#)
[Remove photo](#)

Display Name ▾

My Task Code

Email

Job Title

Preferences

<input type="checkbox"/> Donot Disturb	<input type="checkbox"/> OFF
<input checked="" type="checkbox"/> Global Notifications	<input checked="" type="checkbox"/> ON
<input checked="" type="checkbox"/> Email Notifications	<input checked="" type="checkbox"/> ON

Fig . 15 User Profile

- A user's display name, job title, and profile picture can all be changed.
- My task code is an acronym that will be used as my personal task code's initials.
- The toggle buttons can be used to set preferences.
- Only users with Admin access will be able to see the Admin Console.

4.9 Code Snippets

```

const Home = lazy(() => import('../Home/Home'));
const Tasks = lazy(() => import('../Tasks/Tasks'));
const Login = lazy(() => import('../Login/Login'));
const Signup = lazy(() => import('../SignUp/Signup'));
const ForgotPassword = lazy(() => import('../ForgotPassword/ForgotPassword'));
const Project = lazy(() => import('../Projects/Project/Project'));
const Loader = lazy(() => import('../Loader/Loader'));
const ResetPassword = lazy(() => import('../ResetPassword/ResetPassword'));
const Queue = lazy(() => import('../Queues/Queue/Queue'));
const Workflow = lazy(() => import('../Workflow/Workflow'));
const ActivityDigest = lazy(() => import('../ActivityDigest/ActivityDigest'));
const Inbox = lazy(() => import('../Inbox/Inbox'));
const Calendar = lazy(() => import('../Calendar/Calendar'));
const HelpSupport = lazy(() => import('../HelpSupport/HelpSupport'));
const ManageUsers = lazy(() => import('../Admin/ManageUsers/ManageUsers'));
const RolePermissions = lazy(() => import('../Admin/RolePermissions/RolePermissions'));
const ManageOrganization = lazy(() => import('../Admin/ManageOrganization/ManageOrganization'));
const ActivityLog = lazy(() => import('../Admin/ActivityLog/ActivityLog'));
const ViewOrganisation = lazy(() => import('../SuperAdmin/ViewOrganisation'));

```

Fig 16. App Component (container for all components in the application)

```

> config
> controllers
> helpers
> middlewares
> node_modules
> Notification
> routes
> services
> utils
> workers
≡ combined.log
≡ error.log
JS index.js
{} package-lock.json
{} package.json

```

Fig 17. Folder Structure Backend

```

return (
  <Suspense fallback={renderLoader}>
    <div id="layout-wrapper">
      <div className="sidenav-overlay"></div>
      <Router>
        {auth.isAuthenticated ? (
          <React.Fragment>
            <SideBar sendCollapse={getCollapse} />
            <Header
              getCollapse={getCollapse}
              sideCollapse={sideCollapse} />
          </React.Fragment>
        ) : (
          ...
        )}
      </Router>
      <Switch>
        {Object.values(route.PRIVATE_ROUTE).map((route, index) => {
          const PrivateComponent = getComponent(route.COMPONENT)
          return (<PrivateRoute
            exact
            path={route.ROUTER_PATH}
            key={index}
            isAuthenticated={auth.isAuthenticated}
            renderer={(props) => {
              return (
                <div className="main-content">
                  <Breadcrumbs
                    {...props}
                    route={route}
                  />
                </div>
              )
            }}
          />
        )}
      </Switch>
    </div>
  </Suspense>
)

```

Fig 18. Protected Routes (checks if user is authorized to visit a route)

```

    ))
  }
  {Object.values(route.ROUTE).map((route, index) => {
    const RouteComponent = getComponent(route.COMPONENT)
    return (
      <Route key={index} path={route.ROUTER_PATH} component={RouteComponent} />
    )
  })
}
</Switch>
</Router>
{showAlert &&
  <Notify message={alertMsg} icon={alertIcon} onClicknotif = {onClickNotification}/>
  {loading && <Loader />}
}
</div>
</Suspense>
)
}
/**
 * merges ReduxStore with props
 * @param {*} state
 * @returns {state as props}
 */
function mapStateToProps(state) {
  return {
    auth: state.auth,
    loading: state.loader.loading
  };
}
export default connect(mapStateToProps)(App);

```

Fig 19. Storing applications state in Redux store


```
const ApiRoute = () => {  
  const GET_REQUEST = [  
    "/user/getUserDetails",  
    "/user/getAllUserOfACompany",  
    "/user/getNotificationCount",  
    "/queue/getAllQueues",  
    "/queue/getUserQueues",  
    "/queue/getQueueProjects",  
    "/admin/getOrganization",  
    "/admin/getActivityLog",  
    "/admin/getFiterData",  
    "/project/getCompanyProjects",  
    "/project/getProjectSelects",  
    "/project/getUserProjects",
```

Fig 20. List of routes (in the backend which maps a given request to a controller and an action.)

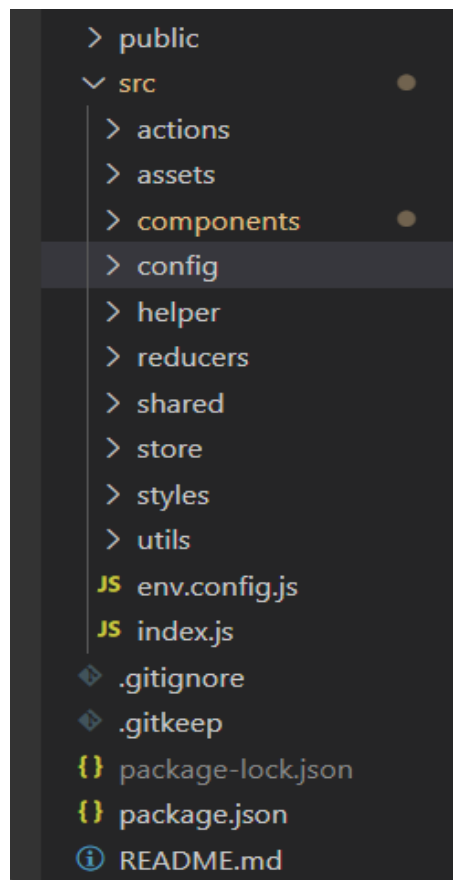


Fig 21. Folder Structure Frontend

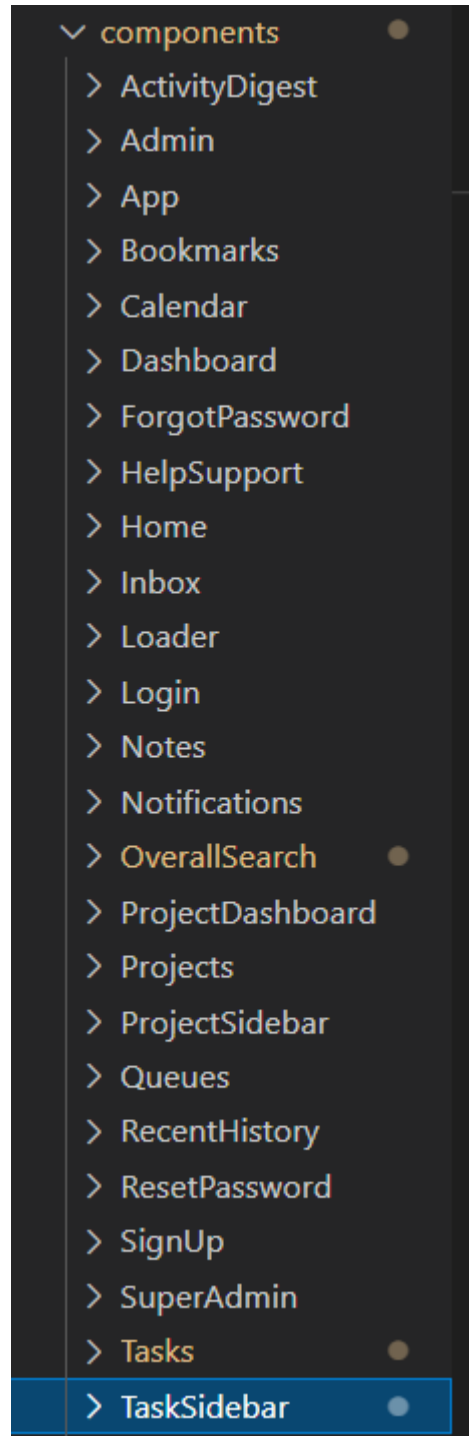


Fig 22.Components

Chapter 05:- CONCLUSION

5.1 Conclusion

I had been exposed to programmer working life after completing my industrial training. During my internship, I learned more about the concept of a programmer and how to prepare myself to be a responsible and inventive programmer in the future. During my training, I realized that observation is a critical component in determining the fundamental cause of an issue. Not only for my project, but also for everyday activities. Throughout my project, I collaborate with my colleagues and operators to identify issues. Furthermore, the project indirectly assists me in learning independently, self-discipline, being considerate/patient, self-confidence, taking initiative, and problem-solving abilities.

Furthermore, when interacting with others, my communication abilities have improved. When I made mistakes during my training, I received criticism and advice from engineers. However, their suggestions are helpful in helping me adjust and prevent repeating the same mistakes. Apart from that, I had improved my programming skills by participating in numerous programmes. This also helps me improve my web building skills. To summarize, the actions I learnt throughout industrial training will help me handle obstacles in the workplace in the future. During my industrial training, I discovered that numerous factors are critical:

Critical Thinking : We must assess our difficulties and assignments in order to organize our chores and assignments, as well as design a viable solution to the problem. We'd have to devise a contingency plan for the solution so that we're ready for any unanticipated circumstances.

Time Management: Because overall programmers are always working against a tight deadline and a jam-packed schedule, effective time management will help them avoid missing deadlines. Effective time management allows us to complete our assignments quickly and on time. Scheduling saves time and allows us to plan ahead, allowing us to acquire more.

Team Interactions: Teamwork is critical in the workplace for building a great organization. Teamwork is also necessary for the organization's overall goals to be met. As a result, communication and sharing are critical in the workplace. As a result, instead of working alone, we should respect each other at work and collaborate as a team. This is because achieving our goals is easier when we work together as a team rather than working alone.

REFERENCES

1. Bozikovic, H., Stula, M. (2018). Web design Past, present and future. 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO).
2. Carter, B. (2014). HTML Architecture, a Novel Development System (HANDS): An Approach for Web Development. 2014
3. Sterling, A. (2019). NodeJS and Angular Tools for JSON-LD. 2019 IEEE 13th 22
4. What is Git, Why Should You Use it? (2021)
5. Erin Gilliam Haije.(2022). Best Project Management Software and Overview.
6. Shahriar Shovon.(2018). Testing REST API Using Postman.
7. Yogesh Baiskar, Priyas Paulzagade, Krutik Koradia, Pramod Ingole, Dhiraj Shirbhate.(2022). A Full-Stack Development.
8. Sanja Delcev, Drazen Draskovic.(2018).Modern JavaScript frameworks: A Survey Study.
9. G. Bierman, M. Abadi, Mads Torgersen.(2014).Understanding TypeScript
10. Paxcom.net
- 11 .Paymentus.com
12. Gitlab.com

Muskan Thakur 181208

ORIGINALITY REPORT

7%

SIMILARITY INDEX

3%

INTERNET SOURCES

1%

PUBLICATIONS

6%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to University of Greenwich

Student Paper

1%

2

Submitted to New York College in Athens,
Greece

Student Paper

1%

3

Submitted to Ghana Technology University
College

Student Paper

1%

4

www.javatpoint.com

Internet Source

1%

5

Submitted to Kyambogo University

Student Paper

1%

6

Submitted to Softwarica College Of IT & E-
Commerce

Student Paper

1%

7

Submitted to Middlesex University

Student Paper

<1%

8

Submitted to University of Wolverhampton

Student Paper

<1%

9	Submitted to Birkbeck College Student Paper	<1 %
10	Submitted to Vietnamese-German University Student Paper	<1 %
11	www.anblicks.com Internet Source	<1 %
12	Submitted to Informatics Education Limited Student Paper	<1 %
13	library.binus.ac.id Internet Source	<1 %
14	Www.tutorialspoint.com Internet Source	<1 %
15	"Enterprise and Organizational Modeling and Simulation", Springer Science and Business Media LLC, 2019 Publication	<1 %

Exclude quotes On

Exclude bibliography On

Exclude matches < 5 words