# PROJECT-HUB

**A Mini Project Report Submitted**

**in Partial Fulfillment of the Requirements for the Degree of**

# MASTER OF COMPUTER APPLICATIONS

#### By

#### RASHMI MISHRA

#### (UNIVERSITY ROLLNO: 2300290140137)

#### RASHI VISHNOI

#### (UNIVERSITY ROLLNO: 2300290140136)

#### RAVI SINGH

#### (UNIVERSITY ROLLNO: 2300290140138)

#### RYAN GOYAL

#### (UNIVERSITY ROLLNO: 2300290140151)

**Under the Supervision of**

### Dr. VIPIN KUMAR

**Associate Professor**

**KIET Group of Institutions**



**Submitted to**

## DEPARTMENT OF COMPUTER APPLICATIONS

**KIET Group of Institutions, Ghaziabad Uttar Pradesh-201206**

# 

# 

# DECLARATION

We hereby declare that the work presented in this report entitled **“PROJECT-HUB"**, was carried out by us. We have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute.

We have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. We have used quotation marks to identify verbatim sentences and given credit to the original authors/sources.

We affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, We shall be fully responsible and answerable.

**RASHMI MISHRA** (2300290140137)

**RASHI VISHNOI** (2300290140136)

**RAVI SINGH** (2300290140138)

**RYAN GOYAL** (2300290140151)

**ACKNOWLEDGEMENT**

At the outset, we would like to thank our guide and advisor**, Dr. VIPIN KUMAR Associate Professor,** for giving us an opportunity to work on this challenging topic and providing us ample and valuable guidance throughout the Project.

Without his encouragement and constant guidance, we would not have able to finish this project. He has been always a source of inspiration and motivator for innovative ideas during the entire span of this work.

We are grateful **Dr. Arun Kumar Tripathi, Head Department of Computer Applications, KIET Group of Institutions, Ghaziabad** for providing all the necessary resources

to carry out this Project work.

We will be failing in our duty if we don’t acknowledge the people behind this work to give us moral and psychological support. Our special thanks to our parents for their endless care and constant support.

**RASHMI MISHRA** (2300290140137)

**RASHI VISHNOI** (2300290140136)

**RAVI SINGH** (2300290140138)

**RYAN GOYAL** (2300290140151)

## CERTIFICATE

Certified that **Rashmi Mishra (2300290140137), Rashi Vishnoi (2300290140136), Ravi Singh (2300290140), Ryan Goyal(2300290140151)** have carried out the project work having **“Project-Hub”** for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried outby the student himself/herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

**RASHMI MISHRA** (2300290140137)

**RASHI VISHNOI** (2300290140136)

**RAVI SINGH** (2300290140138)

**RYAN GOYAL** (2300290140151)

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

**Date:**

**Dr. VIPIN KUMAR**

**Associate Professor Department of Computer Applications KIET Group of Institutions, Ghaziabad**

**Signature of Internal Examiner Signature of External Examiner**

**Dr. Arun Kumar Tripathi**

**Head, Department of Computer Application KIET Group of Institutions, Ghaziabad**

# 

# ABSTRACT

The "**Project Hub**" is designed to be the go-to place for all your project needs. Think of it as a one-stop shop where you can plan, manage, and keep track of every project detail without the hassle.

Whether you’re a project manager, a team member, or a stakeholder, this platform brings everyone together on the same page. It has everything you need: tools for planning, assigning tasks, tracking progress, storing documents, and even communicating within the team.

By putting all these features in one place, Project Hub makes it easier for everyone to stay organized, informed, and aligned, ensuring projects run smoothly from start to finish. It’s not just about making work easier—it's about fostering better collaboration and ultimately driving project success

#### TABLE OF CONTENTS

[Declaration ii](#_TOC_250021)

Acknowledge iii

[Certificate iv](#_TOC_250020)

[Abstract v](#_TOC_250019)

Table of content vi

List of tables vii

[List of figures viii](#_TOC_250018)

Chapter 1: Introduction

* 1. Project description 9
  2. [Project Scope 10](#_TOC_250017)
  3. [Future Scope 10](#_TOC_250016)
  4. [Identification of need 11](#_TOC_250015)
  5. [Problem Statement 12](#_TOC_250014)
  6. [Software/Technology used in project 12](#_TOC_250013)
     1. Non-Functional Requirement 14
     2. Functional Requirement 15

Chapter 2: Feasibility Study

* 1. [Introduction 21](#_TOC_250009)
  2. [Main Aspects 23](#_TOC_250008)
     1. [Technical feasibility 24](#_TOC_250007)
     2. Economical Feasibility 24
     3. Operational Feasibility 24
  3. [Benefits 2](#_bookmark0)5
  4. SRS 26

[Chapter 3: Design](#_bookmark1)

* 1. [Introduction 2](#_bookmark2)8
  2. Analysis 29
  3. [SDLC 3](#_bookmark3)1
  4. Soft. Engg. Paradigm 33
  5. Architecture of the system 34
  6. [Control Flow graph 35](#_TOC_250006)

Chapter 4: Report

|  |  |  |
| --- | --- | --- |
| 4.1 Gist |  | 36 |
| 4.2 Some Snippets |  | 37 |
| 4.2.1 | App | 37 |
| 4.2.2 | Firebase | 41 |
| 4.2.3 | Android Studio | 43 |

Chapter 5: Coding 44

Chapter 6: Testing

* 1. [Introduction](#_bookmark4) 75
     1. [Testing Objectives](#_TOC_250005)
     2. [Testing Principles](#_TOC_250004)
  2. [Level of testing 76](#_TOC_250003)
     1. [Unit Testing](#_TOC_250002)
     2. [Integration Testing 77](#_TOC_250001)
     3. [System Testing 77](#_TOC_250000)

Chapter 7: Conclusion & Future scope

* 1. [Conclusion](#_bookmark5) 78
  2. [Future Scope](#_bookmark6) 79
  3. Bibliography 80

## LIST OF FIGURES

Figure 1.1: React Native logo 12

Figure 1.2: MongoDB logo 12

Figure 1.3: JWT logo 13

Figure 1.4: Node.js logo 13

Figure 3.1: Planning step 31

Figure 3.2: Architecture of the system 34

Figure 3.3: Control Flow Diagram 35

Figure 4.1: User Verification Page 37

Figure 4.2: OTP Verification 38

Figure 4.3: Setting profile 39

Figure 4.4: Registered User 40

Figure 4.5: Firebase database 41

Figure 4.6: Firebase Database Authentication 42

Figure 4.7: Android Studio Main Window 43

Figure 5.1: Testing pyramid 77

# CHAPTER 1

**INTRODUCTION**

## PROJECT DESCRIPTION

The "Project Hub" is an easy-to-use platform that acts as the ultimate collection point for all your projects. Suitable for minor projects, this centralized hub ensures that all project-related links and documents are stored in one convenient location, making it incredibly easy to manage and access everything you need.

1. **Central Repository**: The Project Hub forms a central repository for all project links, documents, or information that you might need about a project. No further digging through emails or folders just to find what you may be looking for.

2. **Easy Management**: You would easily manage multiple projects just by having a single view for all of them and then having each project's distinct space within the hub makes everything organized and separate.

3. **Link Inclusion**: You can add links to relevant resources, reference materials, or any other external tools relevant to your project. This way, you can easily access all the necessary tools and information from the hub.

4. **File Hosting**: Uploading and hosting all project documents in one place means that you always have the latest version of the files and it is easy to share the files with the team members or collaborators.

5. **Collaboration**: The Project Hub accommodates multiple users, making collaboration easier. Team members can add and update links and documents so that everyone is on the same page.

6. **Easy Interface**: Anyone can use the Project Hub with minimal effort. It has an intuitive interface, making it easy to navigate and manage your projects without extensive technical knowledge.

Project Hub simplifies project management by centralizing all your projects in one place, ensuring quick access to everything you need.

## PROJECT SCOPE

1. Provide a centralized, streamlined platform for managing and accessing all project-related resources.
2. Consolidate links, documents, and updates in one easy-to-use interface. This simplifies project oversight.
3. Collect all project-related materials in one convenient location, meaning no more searching through various sources.
4. Support users in systematically organizing their projects, providing dedicated spaces for related information and resources.
5. Design for an intuitive and easy navigation experience that allows for quick access and retrieval of needed information.
6. Streamline management of projects, reduce the time and effort it takes to track multiple projects and materials involved.
7. Increase productivity and enhance organization and make collaborating easier.
8. Be an incredibly valuable tool for managing many different projects.

## 1.3 FUTURE SCOPE

1. Use advanced analytics to map project performance, identify trends and provide data-driven recommendations for improvement in project efficiency and success rate.
2. Improved security measures to protect project data: advance encryption, multi-factor authentication, and compliance with all relevant industry standards and regulations.
3. Ensure the platform can be scaled to accommodate larger teams and more complex projects, providing necessary infrastructure and resources to support growth and expansion.
4. That's where AI and machine learning capabilities for predictive analytics with automated assignment of tasks and project management suggestions based on user behavior and project data come into action.

## 1.3 IDENTIFICATION OF NEED

Today, in this collaborative and ever-evolving tech environment, the need for a platform to share and access project resources is very significant. Developers, students, and professionals often require access to various projects, complete with source code and documentation, to learn, draw inspiration, and collaborate.

Key Needs Identified:

1. There is a centralized need for a platform by which people can upload their respective projects. This way, information about the projects being maintained will be stored all together in one place that allows users to find those specific projects that they think they need.
2. Accessibility to Source Code: Users should easily access the source code of projects. The Project Hub enables users to view, download, and contribute to the source codes of projects by linking to GitHub repositories.
3. Documentation Access: Understanding and using a project effectively requires detailed documentation. Providing links to documentation stored on platforms like Google Drive gives users access to the detailed instructions and explanations.
4. Search and Discoverability: The users need a strong search functionality that will enable them to find projects based on particular criteria, such as the type of project, the technology used, or even user ratings. This will enhance the discoverability of relevant projects.
5. Collaboration and Sharing: Collaboration is encouraged by enabling multiple users to contribute to projects, add comments, and provide feedback. .
6. User-Friendly Interface The intuitive and easy-to-use interface ensures that the users upload, search, or find projects with less extensive technical knowledge.

## PROBLEM STATEMENT

There is great demand for a centralized system to efficiently share and recover project resources. In such a case, there exists dispersion in resources, restricted access, and sub-optimal collaboration among different users. Users face several instances of difficulties in locating projects relevant to them, while the documentation tends to be mostly disjointed. The "Project Hub" responds to this demand by providing a central hub where users can upload their projects, thus making the source code available via GitHub and documentation accessible via Google Drive. This approach makes items more discoverable, collaborative, and better managed for resources of the project.

## SOFTWARE/TECHNOLOGY USED IN PROJECT

#### React Native

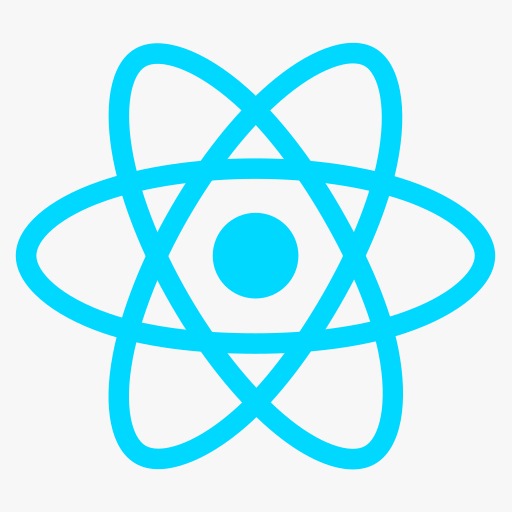


Figure 1.1: React logo

React is a  library for building user interfa**ces** (UIs) on the web. React is a declarative, component based library that allows developers to build reusable UI components and It follows the [Virtual DOM](https://www.geeksforgeeks.org/reactjs-virtual-dom/) (Document Object Model) approach, which optimizes rendering performance by minimizing DOM updates. React is fast and works well with other tools and libraries.

1. **MongoDB**



Figure 1.2: MongoDB logo

MongoDB, the most popular NoSQL database, is an open-source document-oriented database. The term ‘NoSQL’ means ‘non-relational’. It means that MongoDB isn’t based on the table-like relational database structure but provides an altogether different mechanism for storage and retrieval of data. This format of storage is called BSON (similar to JSON format).

#### JSON Web Token (JWT)



Figure 1.3: JWT logo

## JSON Web Tokens (JWT) are compact, URL-safe tokens used for securely transmitting information between parties as a JSON object. They are commonly used for authentication and information exchange.

#### D. Node.js

#### Image result for node.js

#### Figure 1.4: node.js logo

## [Node.js](https://www.geeksforgeeks.org/nodejs/) is an open-source, cross-platform[JavaScript](https://www.geeksforgeeks.org/javascript/) runtime environment that allows developers to run JavaScript code on the server side. Created by Ryan Dahl in 2009, Node.js has revolutionized [server-side programming](https://www.geeksforgeeks.org/server-side-client-side-programming/) by offering an efficient, event-driven, and non-blocking I/O model.

## NON- FUNCTIONAL REQUIREMENTS

The non-functional requirements pertinent to the Project Hub encompass:

1. **Scalability**: The platform must possess the capability to accommodate a growing volume of users, projects, and data while maintaining consistent performance levels.
2. **Performance**: The Project Hub is expected to deliver rapid response times and effective processing, thereby facilitating a seamless and responsive user experience.
3. **Reliability**: The system is required to be trustworthy, exhibiting minimal downtime and demonstrating a swift recovery from any failures or interruptions.
4. **Usability**: The interface designed for users ought to be intuitive, user-oriented, and straightforward to navigate, thereby guaranteeing that individuals with varying levels of technical expertise can utilize the platform efficiently.
5. **Security**: Comprehensive security protocols must be established to safeguard user data and project details against unauthorized access, breaches, and other potential risks.
6. **Compatibility**: The Project Hub must exhibit compatibility with a range of browsers, devices, and operating systems, thereby ensuring that it remains accessible to all users.
7. **Maintainability**: The platform shall be designed so that is easy to maintain and update for use of new features and their bug fixes.
8. Portability: Users shall easily access the Project Hub from different devices and locations.
9. **Interoperability**: The platform shall integrate without any hitch with other tools and systems like the source code on GitHub and the documentation on Google Drive.
10. **Accessibility**: The Project Hub is required to adhere to accessibility standards to guarantee its usability by persons with disabilities.
11. **Data Integrity**: The system must maintain the accuracy and consistency of data, thereby averting any corruption or loss during operational processes.
12. **Scalability of User Access**: The platform ought to accommodate a significant number of simultaneous users without experiencing performance decline.

## FUNCTIONAL REQUIREMENTS

These functional specifications comprise the following parts in the Project Hub:

1. **User registration and authentication**: Users must have the functionality to sign up and login to the portal securely
2. **Project upload and management**: Users should have functionality that lets them upload their projects, which, in turn, should consist of source code associated with GitHub and documentation associated with Google Drive; manage the project (edit, update, delete).
3. **Project Search and Filtering**: Users must be able to search for projects based on various criteria, such as project type, technology used, or user ratings, and then filter the results to easily find relevant projects.
4. **Project Viewing**: Users should be able to access detailed information about each project, including descriptions, links to source code, and related documentation.
5. **User Profile Management**: Users should be able to create and manage their profiles, which will include contact information, bio, and a list of projects uploaded.
6. **Collaboration Tools**: The platform should have features for collaboration, including commenting on projects, rating projects, and giving feedback.
7. **Notification System**: Users should receive notifications of important updates, such as new comments on their projects or updates to followed projects.
8. **Admin Panel**: There should be an admin panel to manage user accounts, monitor usage of the platform, and moderate content.
9. **Data Export**: Users should be able to export their project data for backup or use in other platforms.
10. **Responsive Design**: The platform should be accessible and usable on various devices, including desktops, tablets, and smartphones.
11. **User Support**: Develop a support structure that allows users to get help with using the platform, such as FAQs, instructional documents, and ways of contacting support.
12. **Analytics and Reporting**: Include features to track and record project views, user engagement, and other relevant metrics, thereby helping users understand their projects' impact.

# CHAPTER 2

**FEASBILITY STUDY**

## INTRODUCTION

Feasibility of the system in an important aspect, which is to be considered. The system needs to satisfy the law of economic, which states that the maximum output should be yielded in minimum available resources.

A feasibility analysis evaluates the project’s potential for success; therefore, perceived objectivity is an essential factor in the credibility of the study for potential investors and lending institutions. There are five types of feasibility study separate areas that a feasibility study examines, described below.

1. **Technical Feasibility :**

This assessment focuses on the technical resources available to the organization. It helps organizations determine whether the technical resources meet capacity and whether the technical team is capable of converting the ideas into working systems. Technical feasibility also involves the evaluation of the hardware, software, and other technical requirements of the proposed system. As an exaggerated example, an organization wouldn’t want to try to put Star Trek’s transporters in their building currently, this project is not technically feasible.

1. **Economic Feasibility :**

This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. It also serves as an independent project assessment and enhances project credibility— helping decision-makers determine the positive economic benefits to the organization that the proposed project will provide.

1. **Legal Feasibility :**

This assessment investigates whether any aspect of the proposed project conflicts with legal requirements like zoning laws, data protection acts or social media laws. Let’s say an organization wants to construct a new office building in a specific location. A feasibility study might reveal the organization’s ideal location isn’t zoned for that type of business.That organization has just saved considerable time and effort by learning that their project was not feasible right from the beginning.

1. **Operational Feasibility :**

This assessment involves undertaking a study to analyze and determine whether and how well the organization’s needs can be met by completing the project. Operational feasibility studies also examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development.

**5. Scheduling Feasibility:**

This assessment is the most important for project success; after all, a project will fail if not completed on time. In scheduling feasibility, an organization estimates how much time the project will take to complete.

When these areas have all been examined, the feasibility analysis helps identify any constraints the proposed project may face, including:

* + Internal Project Constraints: Technical, Technology, Budget, Resource, etc.
  + Internal Corporate Constraints: Financial, Marketing, Export, etc.
  + External Constraints: Logistics, Environment, Laws, and Regulations, etc

## MAIN ASPECTS

There are three aspects of feasibility to be considered namely.

1. Technical
2. Operational
3. Economical
4. **TECHNICAL:**

In the technical aspects one may consider the hardware equipment for the

installation of the software. The system being centralized will required very little hardware appliances. Hence this helps the system to work smoothly with limited amount of working capitals.

1. **OPERATIONAL:**

In the operational aspects may think of the benefits of the workload that

many a personal may have to share. This is eased out and the required output may be retrieved in a very short time. Thus there is accuracy in the work on time is also saved there will be very little work that needs to be performed.

1. **ECONOMICAL:**

Economical system is definitely feasible because the software

requirement is less and the operational working for the system requires less number of recruits. This help introduction over-staffing and wastage funds.

We studied on the position to evaluate solution. Most important factors in this study were tending to overlook the confusion inherent in Application Development the constraints and the assumed studies. It can be started that it the feasibility study is to serve as a decision document it must answer three key questions.

1. Is there a new and better way to do the job that will benefit the user?
2. What are the costs and savings of the alternatives?
3. What is recommended?

### Technical feasibility:

This centers on the existing computer system (hardware, software etc.) and to what extent it can support the proposed additional equipment .in this stage of study, we have collected information about technical tools available by which I could decide my system design as the technical requirements.

### Operational Feasibility:

In this stage of study we have checked the staff availability. I concentrate on knowledge of end users that are going to use the system. This is also called as behavioral feasibility in which I have studied on following aspects; people are inherently resistant to change, and computers have been known to facilitate change .An estimate has been made to how strong a reaction the user staff is having toward the development of a computerized system. It is common knowledge that computer installations have something to do with turnover. I had explained that there is need to educate and train the staff on new ways of conducting business.

### 1.2.3 Economical feasibility :

Economic analysis is the most frequently used method for evaluating the effectiveness of candidate system. More commonly known as cost\benefit analysis, the procedure is to determine the benefits and savings that benefits outweigh costs. The decision was to design and implement system because it is for having chanced to be approved. This is an on going effort that improves the accuracy at each phase of the system life cycle.

In developing cost estimates for a system I need to consider several cost elements. Among these is hardware personal facility. Operating and supply costs.

## 1.3 BENEFITS

Benefits of conducting a feasibility study:

* Improves project teams’ focus
* Identifies new opportunities
* Provides valuable information for a “go/no-go” decision
* Narrows the business alternatives
* Identifies a valid reason to undertake the project
* Enhances the success rate by evaluating multiple parameters
* Aids decision-making on the project
* Identifies reasons not to proceed

## SYSTEM REQUIREMENT SPECIFICATION

Any system can be designed after specifies the requirement of the user about that system. For this first of all gathered information from user by the preliminary investigation which is starting investigation about user requirement.

The data that the analysts collect during preliminary investigation are gathered through the various preliminary methods.

1. **Documents Reviewing Organization :**

The analysts conducting the investigation first learn the organization involved in, or affected by the project. Analysts can get some details by examining organization charts and studying written operating procedures.

Collected data is usually of the current operating procedure:

* + The information relating to clients, projects and students and the relationship between them was held manually.
  + Managing of follow-ups was through manual forms.
  + Complaints require another tedious work to maintain and solve.
  + Payments details had to be maintained differently.

1. **Gathering Information By Asking Questions :**

Interviewing is the most commonly used techniques in analysis. It is always necessary first to approach someone and ask them what their problems are, and later to discuss with them the result of your analysis.

1. **Questionnaires**

Questionnaires provide an alternative to interviews for finding out information about a system. Questionnaires are made up of questions about information sought by analyst. The questionnaire is then sent to the user, and the analyst analyzes the replies.

1. **Electronic Data Gathering:**

Electronic communication systems are increasingly being used to gather information. Thus itis possible to use electronic mail to broadcast a question to a number of users in an organization to obtain their viewpoint on a particular issue.

In my project, with the help of Marg software solutions, I have send questionnaire through electronic mail to twenty employees of the company and retrieved the information regarding the problem faced by existing system.

5) **Interviews :**

Interview allows the analysts to learn more about the nature of the project request and reason of submitting it. Interviews should provide details that further explain the project and show whether assistance is merited economically, operationally or technically.

One of the most important points about interviewing is that what question you need to ask.

It is often convenient to make a distinction between three kinds of question that is

* + - Open questions
    - Closed question
    - Probes

Open questions are general question that establish a person’s view point on a particular subject.

Closed questions are specific and usually require a specific answer.

# CHAPTER 3

# DESIGN

## INTRODUCTION

System is created to solve problems. One can think of the systems approach as an organized-way of dealing with a problem. In this dynamic world, the subject system analysis and design, mainly deals with the software development activities.

Since a new system is to be developed, the one most important phases of software development life cycle is system requirement gathering and analysis. Analysis is a detailed study of various operations performed by a system and their relationship within and outside the system. Using the following steps it becomes easy to draw the exact boundary of the new system under consideration.

All procedures, requirements must be analysed and documented in the form of detailed DFDs, logical data structure and miniature specifications.

System analyses also include sub-dividing of complex process involving the entire system identification of data store and manual processes.

## SYSTEM ANALYSIS

System is created to solve problems. One can think of the systems approach as an organized way of dealing with a problem. In this dynamic world, the subject system analysis and design, mainly deals with the software development activities.

Since a new system is to be developed, the one most important phases of software development life cycle is system requirement gathering and analysis. Analysis involves detailed study of the current system, leading to specification of a new system. Analysis is a detailed study of various operations performed by a system and their relationship within and outside the system. Using the following steps it becomes easy to draw the exact boundary of the new system under consideration.

Keeping in view the problems and new requirements, workout the pros and cons area of the system.

All procedures, requirements must be analyzed and documented in the form of detailed DFDs, logical data structure and miniature specifications.

System analyses also include sub-dividing of complex process involving the entire system, identification of data store and manual processes.

System Analysis is conducted with the following steps

* Information gathering
* The tools of structured analysis
* Identification of Need
* System Planning and initial investigation
* Feasibility study
  + Information Gathering:
  + Information about the firm
  + Information about the workflow
  + Various tools used are:
    - Review of literature
    - Procedure
    - Forms

Initial investigation:

* + Problem definition and project initiation
  + Determining the requirements
  + Needs identification
  + Dimension of planning
  + Determination of feasibility
  + Feasibility Analysis:
  + System Performance definition
  + Identification of system objectives
  + Description of outputs

Preliminary Investigation:

* Evaluation of project request is major purpose of preliminary investigation.
* It is the collecting information that helps committee members to evaluate merits of the project request and make judgment about the feasibility of the proposed projects.
* To answer the above questions, system analysts discuss with different category of person to collect facts about their business and their operations.
* When the request is made, the first activity the preliminary investigation begins.
* Preliminary investigation has three parts-
  + 1. Request clarification
    2. Feasibility study
    3. Request approval

Request Clarification:

An information system is intended to meet needs of an organization. Thus the first step is in this phase is to specify these needs and requirements.

* + - * The next step is to determine the requirements met by the system. Many requests from employees and users in the organizations are not clearly defined. Therefore, it become necessary that project request must examine and clarified properly before considering system investigation.
      * Information related to different needs of the System can be obtained by different users of the system. This can be done by reviewing different organization’s documents such
      * as current method of storing sales data, complaint data etc. By observing the onsite activities the analyst can get close information related to real system.

## SDLC

Software Development Life Cycle (SDLC) is a framework that defines the steps involved in the development of software at each phase. It covers the detailed plan for building, deploying and maintaining the software.

SDLC defines the complete cycle of development i.e. all the tasks involved in planning, creating, testing, and deploying a Software Product.

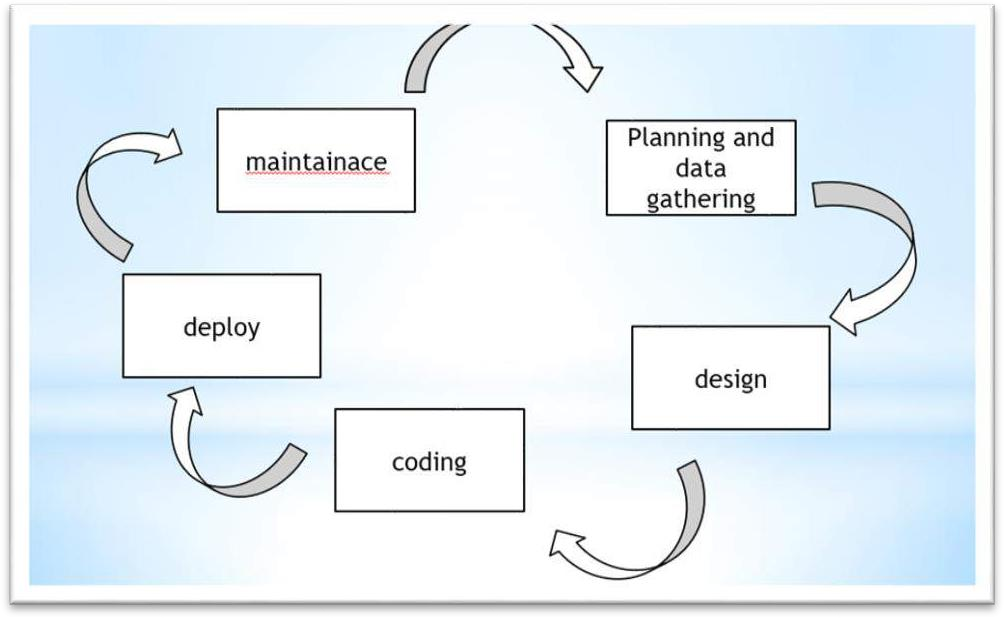


Figure 3.1: Above image depicting the planning step

#### SDLC Phases

**Given below are the various phases:**

* Requirement gathering and analysis
* Design
* Implementation or coding
* Testing
* Deployment
* Maintenance

1. **Requirement Gathering and Analysis:**

During this phase, all the relevant information is collected from the customer to develop a product as per their expectation. Any ambiguities must be resolved in this phase only.

Business analyst and Project Manager set up a meeting with the customer to gather all the information like what the customer wants to build, who will be the end-user, what is the purpose of the product. Before building a product a core understanding or knowledge of the product is very important.

**For Example:** A customer wants to have an application which involves money transactions. In this case, the requirement has to be clear like what kind of transactions will be done, how it will be done, in which currency it will be done, etc.

Once the requirement gathering is done, an analysis is done to check the feasibility of the development of a product. In case of any ambiguity, a call is set up for further discussion.

Once the requirement is clearly understood, the SRS (Software Requirement Specification) document is created. This document should be thoroughly understood by the developers and also should be reviewed by the customer for future reference.

1. **Requirement Gathering and Analysis :**

During this phase, all the relevant information is collected from the customer to develop aproduct as per their expectation. Any ambiguities must be resolved in this phase only.

Business analyst and Project Manager set up a meeting with the customer to gather all the information like what the customer wants to build, who will be the end-user, what is the purpose of the product. Before building a product a core understanding or knowledge of the product is very important.

**For Example:** A customer wants to have an application which involves money transactions. In this case, the requirement has to be clear like what kind of transactions will be done, how it will be done, in which currency it will be done, etc.

Once the requirement gathering is done, an analysis is done to check the feasibility of thedevelopment of a product. In case of any ambiguity, a call is set up for further discussion.

Once the requirement is clearly understood, the SRS (Software Requirement Specification) document is created. This document should be thoroughly understood by the developers and also should be reviewed by the customer for future reference.

1. **Design:**

In this phase, the requirement gathered in the SRS document is used as an input and software architecture that is used for implementing system development is derived.

1. **Implementation or Coding**

Implementation/Coding starts once the developer gets the Design document. The Software design is translated into source code. All the components of the software are implemented in this phase.

1. **Testing:**

Testing starts once the coding is complete and the modules are released for testing. In this phase, the developed software is tested thoroughly and any defects found are assigned to developers to get them fixed.

Retesting, regression testing is done until the point at which the software is as per the

customer’s expectation. Testers refer SRS document to make sure that the software is as

35per the customer’s standard.

**6) Deployment :**

Once the product is tested, it is deployed in the production environment or first [UAT](https://www.softwaretestinghelp.com/what-is-user-acceptance-testing-uat/) [(User](https://www.softwaretestinghelp.com/what-is-user-acceptance-testing-uat/) [Acceptance testing)](https://www.softwaretestinghelp.com/what-is-user-acceptance-testing-uat/) is done depending on the customer expectation.

In the case of UAT, a replica of the production environment is created and the customer along with the developers does the testing. If the customer finds the application as expected, then sign off is provided by the customer to go live.

1. **Maintenance**

After the deployment of a product on the production environment, maintenance of the product i.e., if any issue comes up and needs to be fixed or any enhancement is to be done is taken care by the developers.

## SOFTWARE ENGG. PARADIGM APPLIED

Software engineering is a layered technology. The foundation for software engineering is the process layer. Software engineering processes the glue that holds the technology layers together and enables ratios and timely development of computer software. Process defines a framework for a set of key process areas that must be established for effective delivery of software engineering technology.

Software engineering methods provide the technical how-top’s for building software.

Method compass a broad array of tasks that include requirements analysis, design, program

construction, testing and support. Software engineering tools provide automated or semi- automated support for the process and the methods. When tools are integrated so that information created by one tool can be used by another tool, a system for the support of software development, called computer-aided software engineering is established.

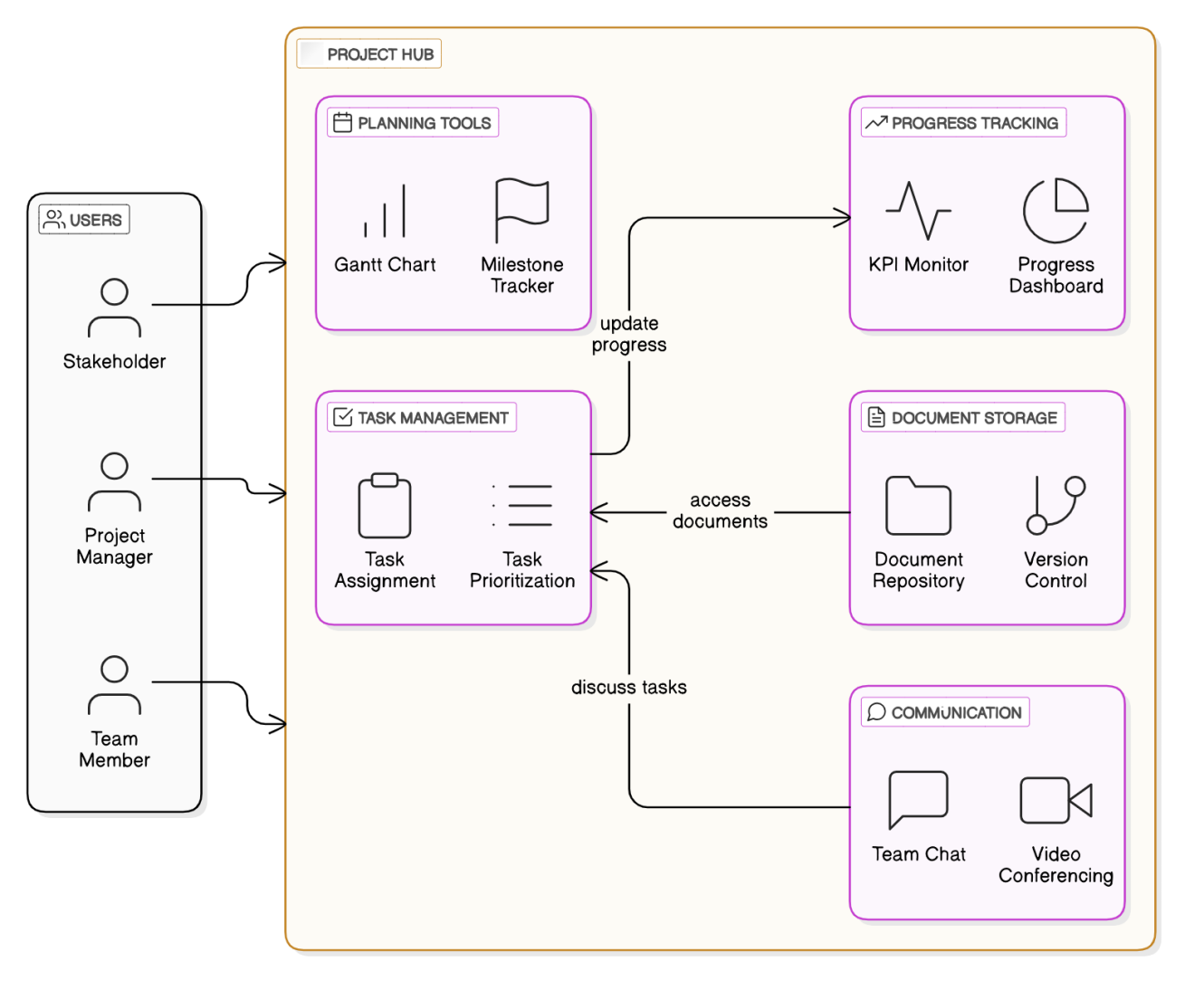
#### The following paradigms are available:

1. The Waterfall Model
2. The Prototyping Model
3. The Spiral model Etc.

## ARCHIETECTURE OF THE SYSTEM

An Architecture of a [system](https://en.wikipedia.org/wiki/System) in which the principal parts or functions are represented by blocks connected by lines that show the relationships of the blocks.

Its for higher level, less detailed descriptions that are intended to clarify overall concepts without concern for the details of implementation.,



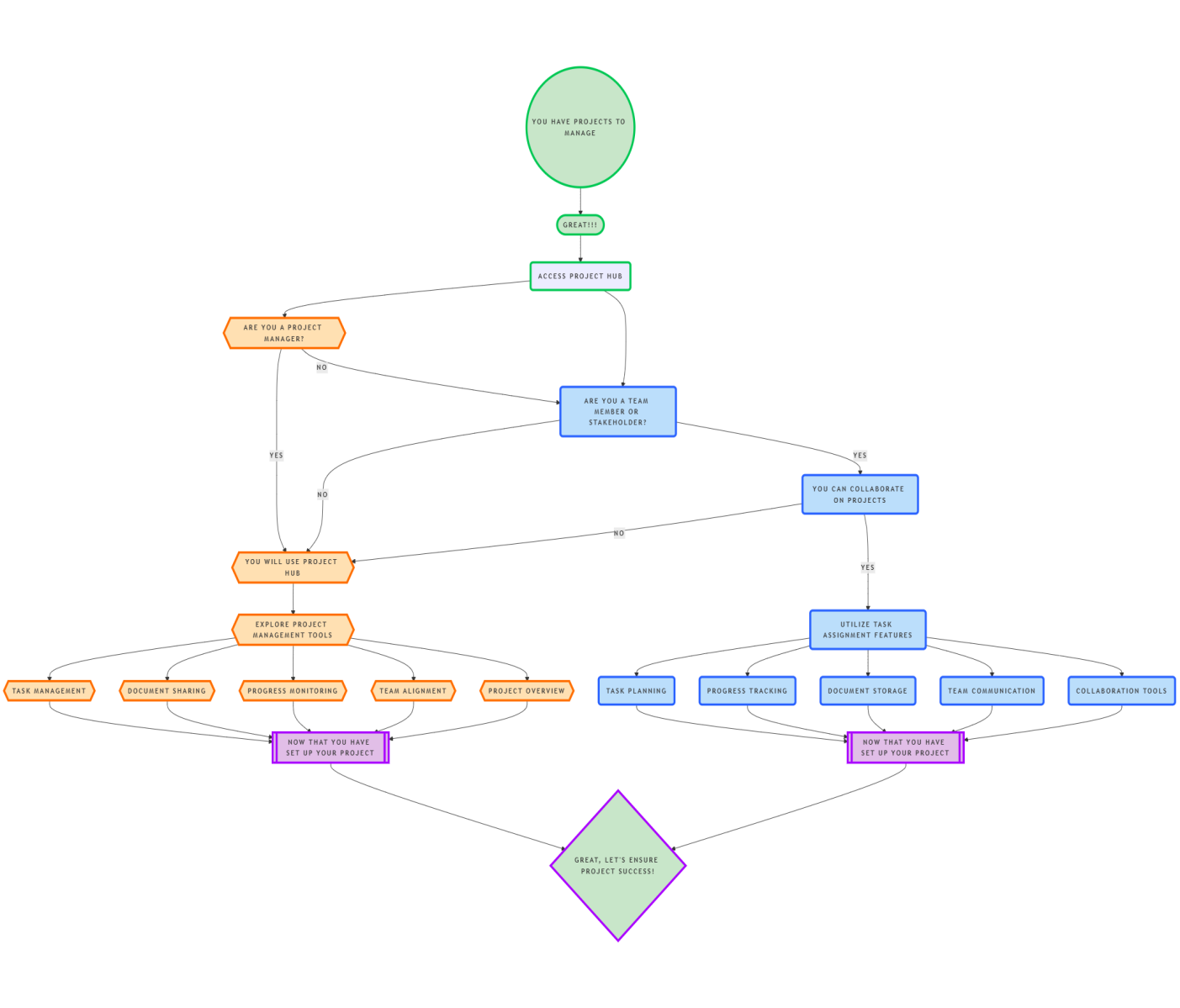
## CONTROL FLOW GRAPH

A Control Flow Graph (CFG) is the graphical representation of control flow or [computation](https://www.geeksforgeeks.org/cyclomatic-complexity/) [during the execution of programs](https://www.geeksforgeeks.org/cyclomatic-complexity/) or applications. Control flow graphs are mostly used in static analysis as well as compiler applications, as they can accurately represent the flow inside of a program unit.

Characteristics of Control Flow Graph:

* Control flow graph is process oriented & directed graph.
* Control flow graph shows all the paths that can be traversed during a program execution.

Edges in CFG portray control flow paths and the nodes in CFG portray basic block



# CHAPTER 4

**REPORT**

## GIST

Project Hub is a comprehensive website dedicated to collecting and showcasing various projects across different domains. Whether you're a student, professional, or hobbyist, Project Hub offers a wide array of projects that you can browse, view, and download with ease. The platform is designed to be user-friendly, ensuring that you can quickly find the projects you need and gain valuable insights into different fields. With Project Hub, accessing and downloading project resources has never been easier.

## SOME SNIPPETS

* + 1. **Web App**

First opening page of our Web Application Project-Hub where user has to verify

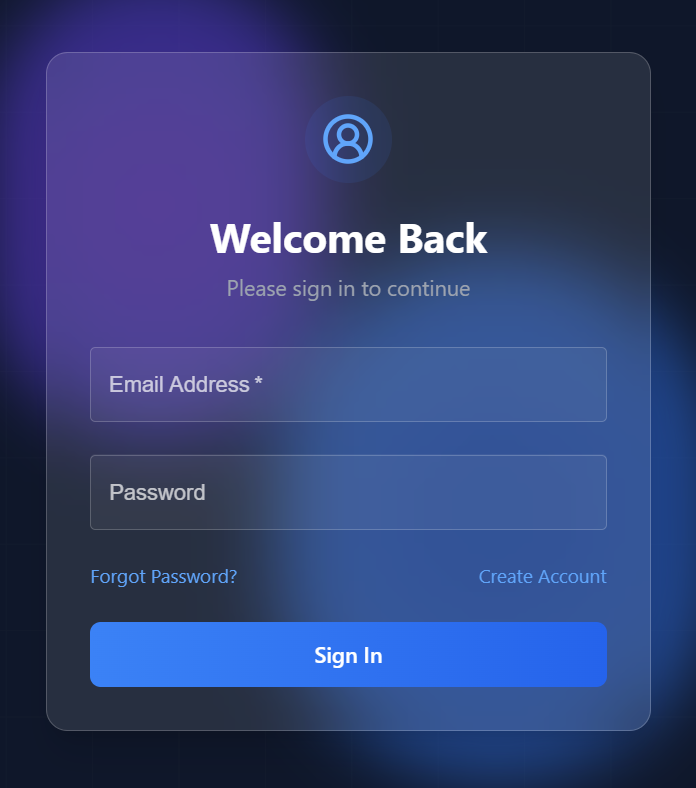
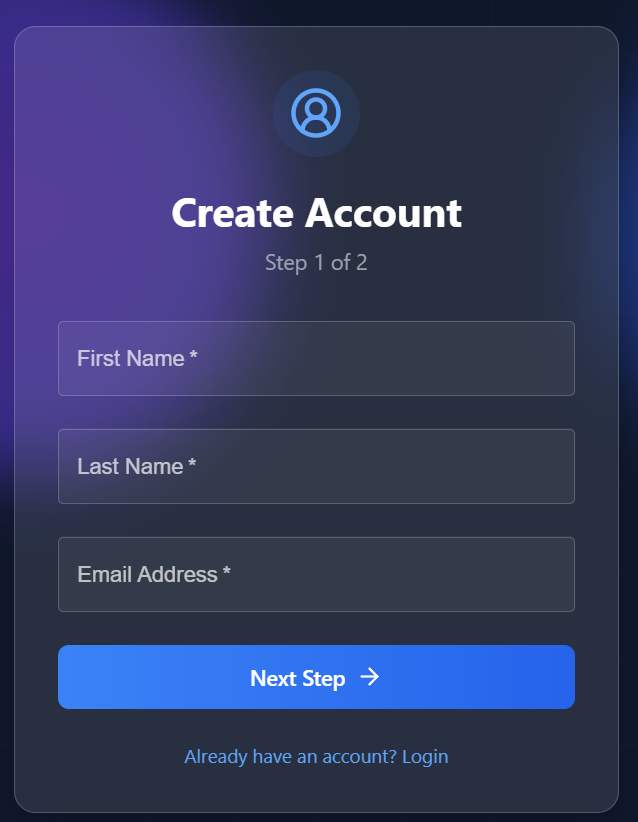


Figure 4.1: Sign in Page

In the given Screenshot we can see that user will see the signup screenshot .



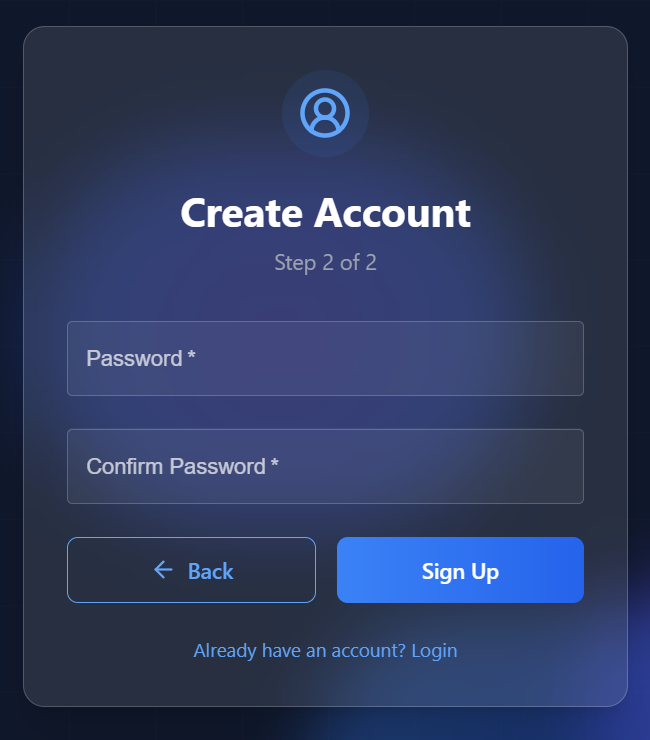


Figure 4.2: User Verification

In the given below screenshot user will setup profile image and its username

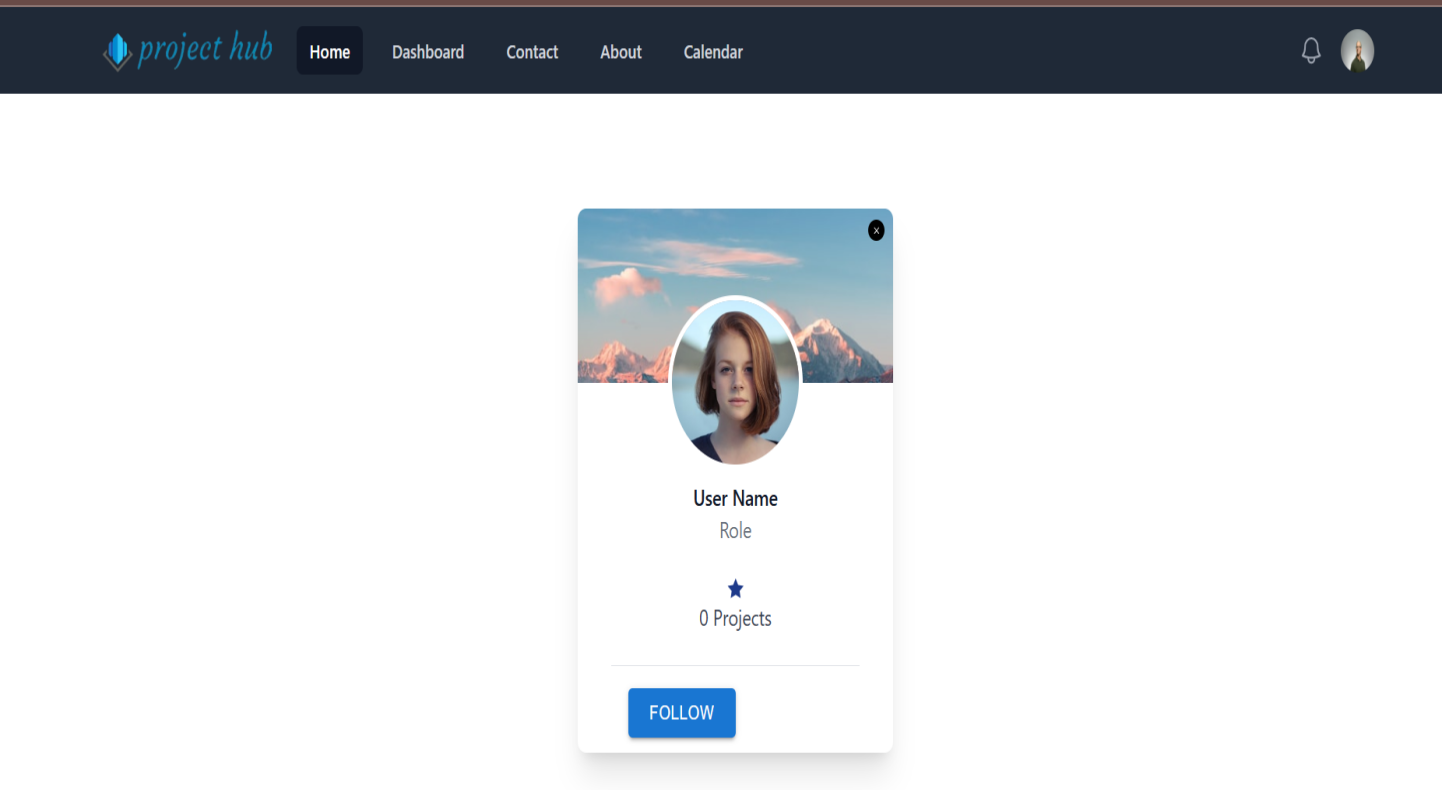


Figure 4.3: Setting profile

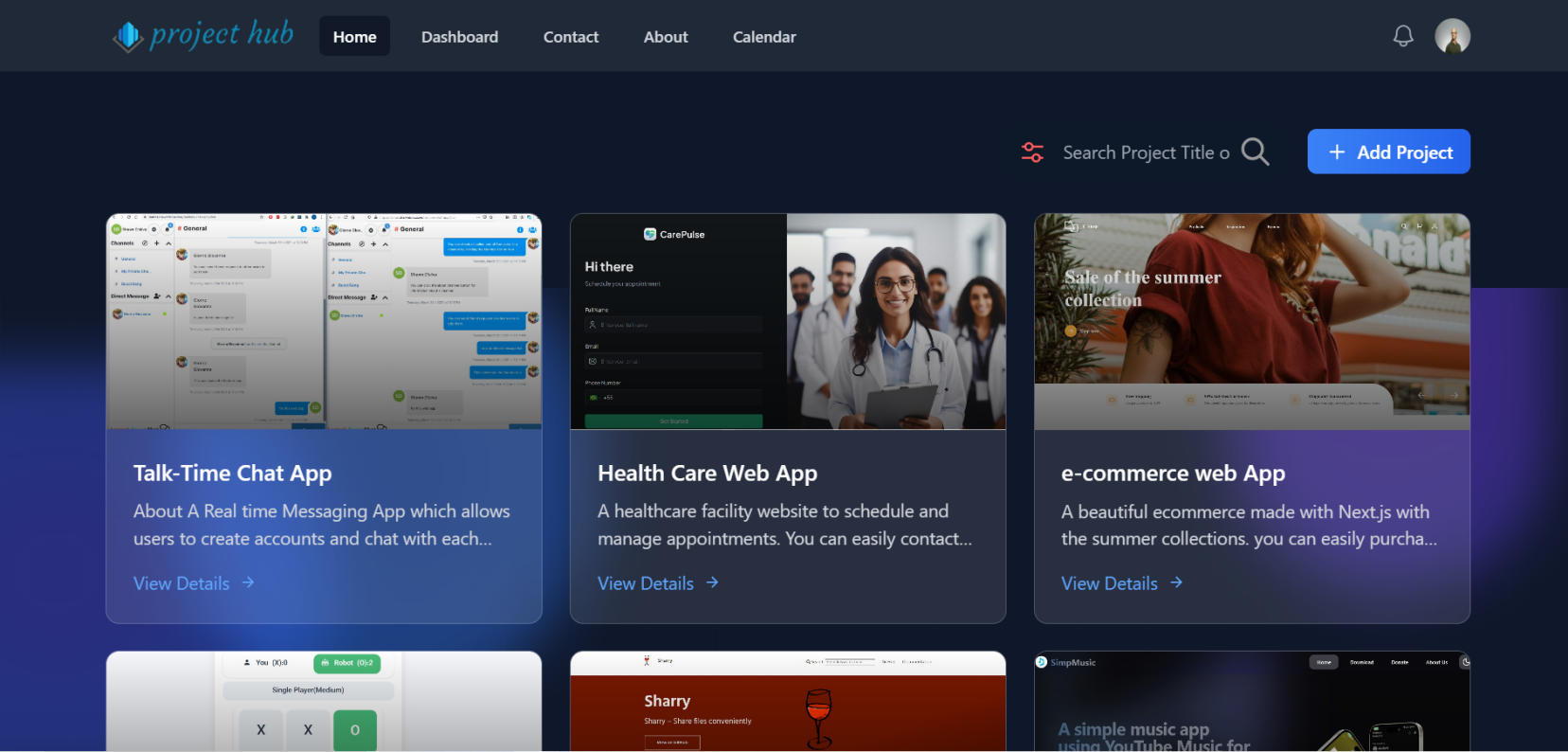


Figure 4.4: Registered User

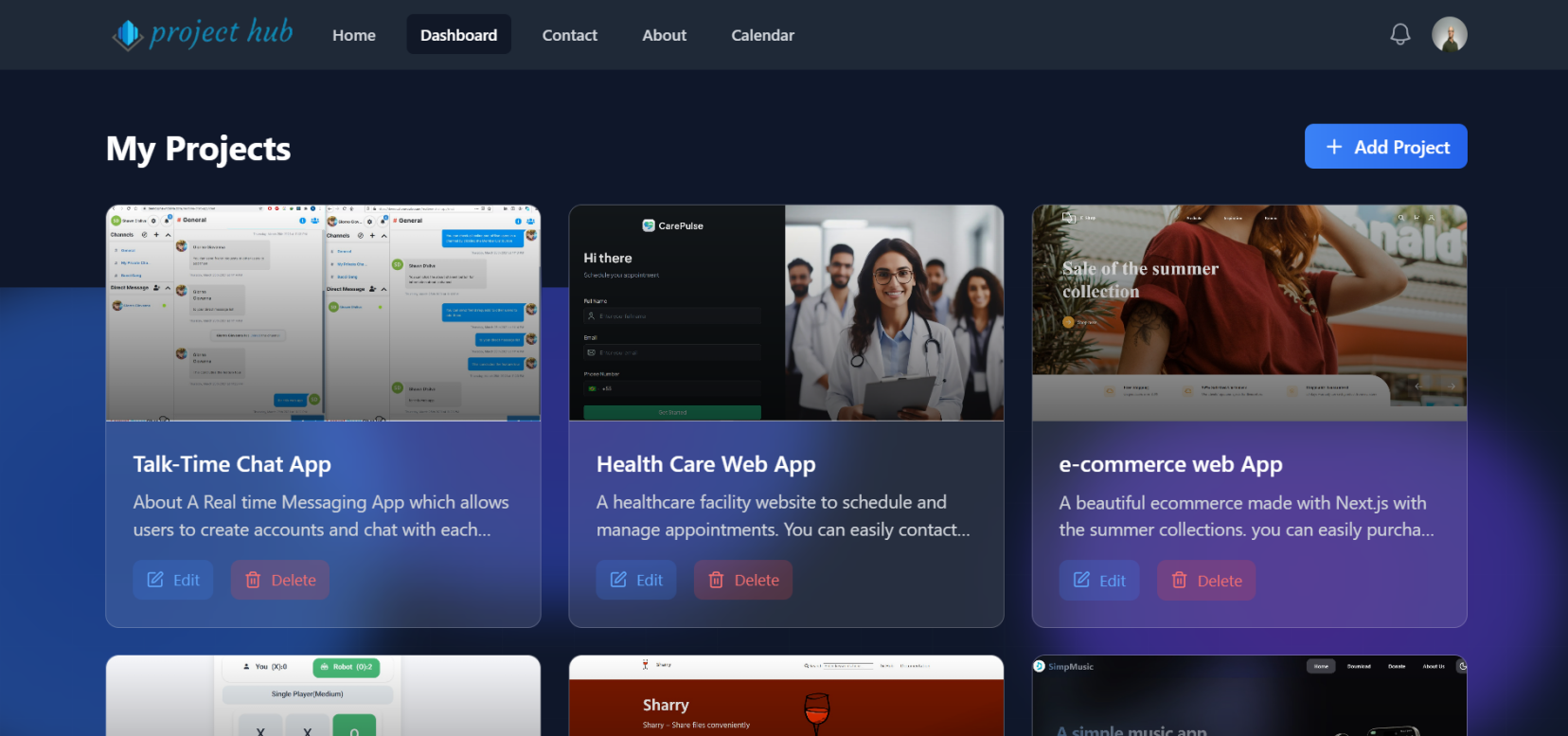


Figure 4.5: DashBoard

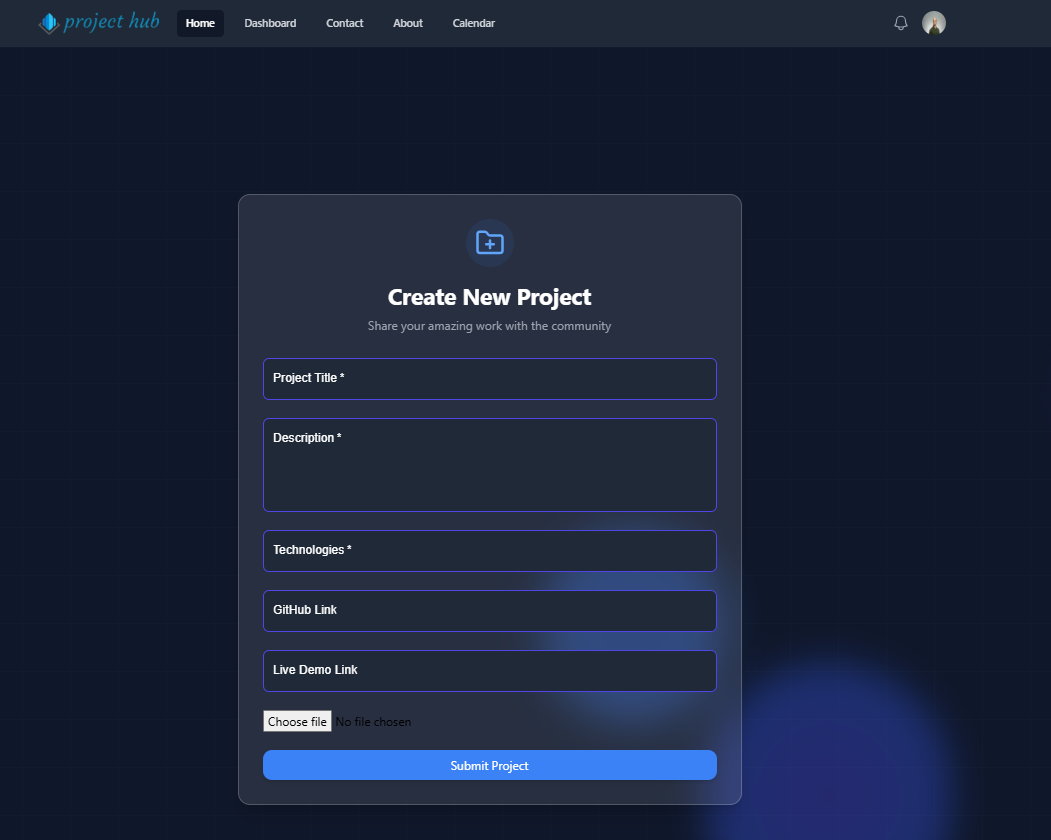


Figure 4.6: Project Addition

## VISUAL STUDIO CODE

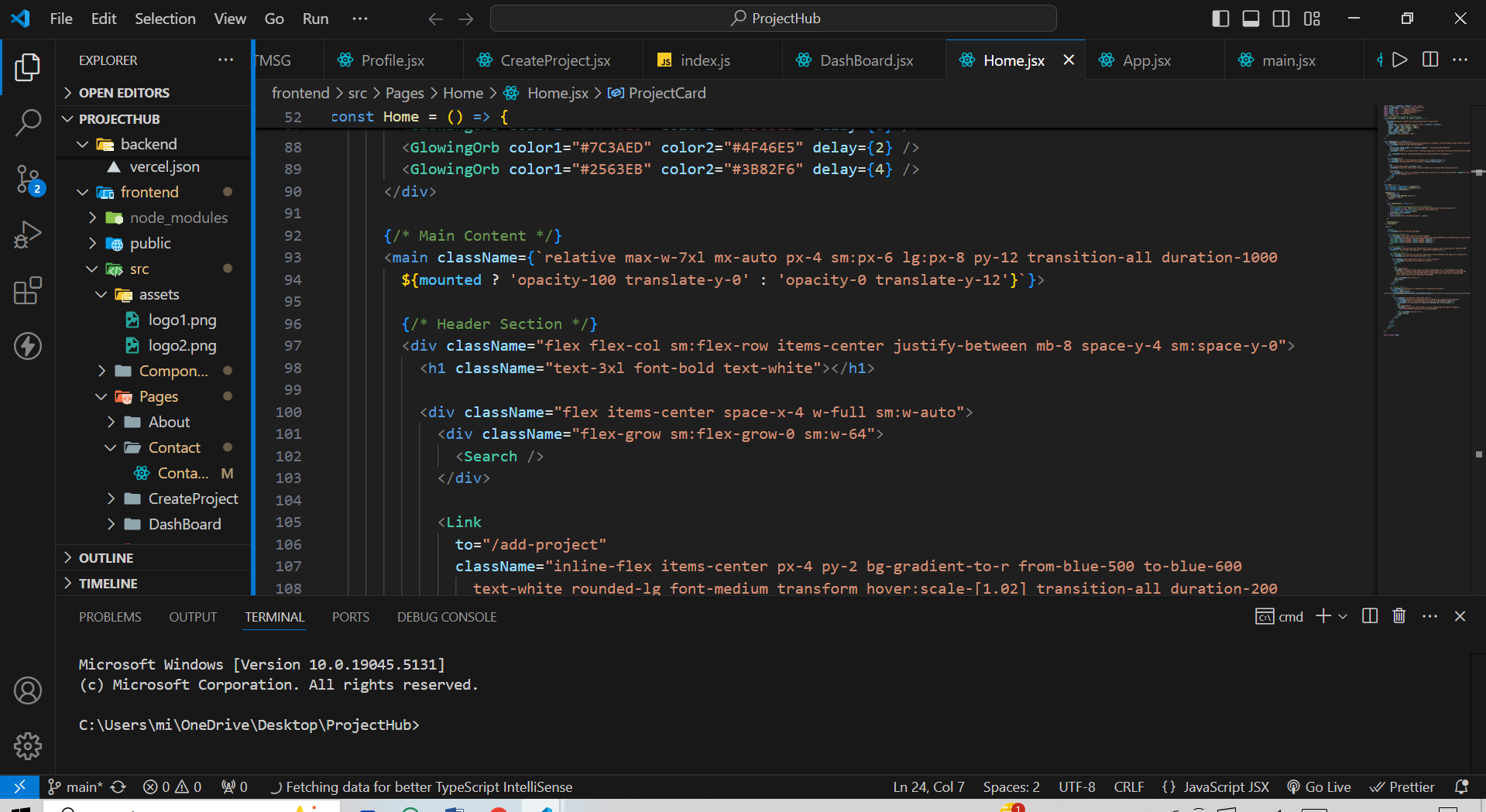


Figure 4.7: VS Code

# CHAPTER 5

**CODING**

This chapter contains some codes of the project. The goal of the coding is to translate the design of the system into code in a given programming language. For a given design, the aim of this phase is to implement the design in the best possible manner. The coding phase affects both testing and maintenance profoundly.

#### Some Codes are as Written below:

**Landing Page**

import React, { useEffect, useState, useCallback } from 'react';

import { Button } from '@mui/material';

import { Code, Laptop, Cloud, Mouse, ArrowDownward } from '@mui/icons-material';

import Storage from "@mui/icons-material/Storage";

// import LogIn from '../LogIn/LogIn';

import { Link } from 'react-router-dom';

const GlowingOrb = ({ delay = 0, color1, color2, size = 200 }) => (

  <div

    className="absolute rounded-full animate-pulse blur-xl opacity-20"

    style={{

      background: `radial-gradient(circle at center, ${color1}, ${color2})`,

      width: `${size}px`,

      height: `${size}px`,

      left: `${Math.random() \* 90}%`,

      top: `${Math.random() \* 90}%`,

      animationDelay: `${delay}s`,

      transform: 'translate(-50%, -50%)',

      transitionDelay:'0.05s'

    }}

  />

);

const FloatingIcon = ({ Icon, position, parallaxStrength = 1, color, delay = 0 }) => {

  const [hover, setHover] = useState(false);

  return (

    <div

      className={`absolute transition-transform duration-500 cursor-pointer

        ${hover ? 'scale-150' : 'scale-100'}`}

      style={{

        ...position,

        transform: `translate(${position.translateX}px, ${position.translateY}px)`,

      }}

      onMouseEnter={() => setHover(true)}

      onMouseLeave={() => setHover(false)}

    >

      <Icon

        className={`w-12 h-12 transition-all duration-500`}

        style={{

          color: color,

          opacity: hover ? 0.4 : 0.2,

          animation: `float ${15 + delay}s ease-in-out infinite`,

          animationDelay: `${delay}s`

        }}

      />

    </div>

  );

};

const ScrollIndicator = () => (

  <div className="absolute bottom-8 left-1/2 transform -translate-x-1/2 animate-bounce">

    <div className="flex flex-col items-center text-white/50">

      <Mouse className="mb-2" />

      <ArrowDownward />

    </div>

  </div>

);

const LandingPage = () => {

  const [mousePosition, setMousePosition] = useState({ x: 0, y: 0 });

  const [scrollPosition, setScrollPosition] = useState(0);

  const [mounted, setMounted] = useState(false);

  const handleMouseMove = useCallback((e) => {

    setMousePosition({

      x: (e.clientX / window.innerWidth - 0.5) \* 40,

      y: (e.clientY / window.innerHeight - 0.5) \* 40,

    });

  }, []);

  const handleScroll = useCallback(() => {

    setScrollPosition(window.scrollY);

  }, []);

  useEffect(() => {

    setMounted(true);

    window.addEventListener('mousemove', handleMouseMove);

    window.addEventListener('scroll', handleScroll);

    return () => {

      window.removeEventListener('mousemove', handleMouseMove);

      window.removeEventListener('scroll', handleScroll);

    };

  }, [handleMouseMove, handleScroll]);

  const parallaxIcons = [

    { Icon: Code, color: '#60A5FA', position: { left: '20%', top: '30%' }, strength: 1.2, delay: 0 },

    { Icon: Laptop, color: '#A78BFA', position: { left: '70%', top: '20%' }, strength: 1.5, delay: 2 },

    { Icon: Storage, color: '#818CF8', position: { left: '85%', top: '60%' }, strength: 2, delay: 4 },

    { Icon: Cloud, color: '#7DD3FC', position: { left: '15%', top: '70%' }, strength: 0.8, delay: 6 },

    { Icon: Cloud, color: '#38BDF8', position: { left: '60%', top: '80%' }, strength: 1.3, delay: 8 }

  ];

  const glowingOrbs = [

    { color1: '#4F46E5', color2: '#2563EB', size: 300, delay: 0 },

    { color1: '#7C3AED', color2: '#4F46E5', size: 250, delay: 2 },

    { color1: '#2563EB', color2: '#3B82F6', size: 200, delay: 4 },

    { color1: '#06B6D4', color2: '#0EA5E9', size: 350, delay: 6 },

  ];

  return (

    <div className="relative min-h-screen bg-gradient-to-b from-slate-900 via-slate-800 to-slate-900 overflow-hidden">

      {/\* Animated Background \*/}

      <div className="absolute inset-0">

        {/\* Grid Pattern \*/}

        <div

          className="absolute inset-0 bg-[linear-gradient(to\_right,#1f2937\_1px,transparent\_1px),linear-gradient(to\_bottom,#1f2937\_1px,transparent\_1px)] bg-[size:4rem\_4rem]"

          style={{

            transform: `translate(${mousePosition.x \* -0.3}px, ${mousePosition.y \* -0.3}px)`,

            transition: 'transform 0.2s ease-out'

          }}

        />

        {/\* Glowing Orbs \*/}

        {glowingOrbs.map((orb, i) => (

          <GlowingOrb key={i} {...orb} />

        ))}

        {/\* Floating Icons with Parallax Effect \*/}

        {parallaxIcons.map((icon, i) => (

          <FloatingIcon

            key={i}

            Icon={icon.Icon}

            position={{

              ...icon.position,

              translateX: mousePosition.x \* icon.strength,

              translateY: mousePosition.y \* icon.strength

            }}

            color={icon.color}

            delay={icon.delay}

          />

        ))}

      </div>

      {/\* Navigation Bar \*/}

      <nav

        className="fixed top-0 w-full px-6 py-4 backdrop-blur-sm z-50 transition-all duration-300"

        style={{

          backgroundColor: `rgba(15, 23, 42, ${Math.min(scrollPosition / 500, 0.9)})`

        }}

      >

        <div className="max-w-7xl mx-auto flex justify-between items-center">

          <span className="text-2xl font-bold bg-gradient-to-r from-blue-400 via-purple-400 to-pink-400 bg-clip-text text-transparent">

            Project Hub

          </span>

          <div className="flex gap-4 items-center">

            {/\* <a href="#features" className="text-white/70 hover:text-white transition-colors">Features</a>

            <a href="#explore" className="text-white/70 hover:text-white transition-colors">Explore</a> \*/}

            <Link to="/login"><Button

              variant="outlined"

              className="bg-white/5 text-white hover:bg-white/10 backdrop-blur-sm border-white/20"

            >

              Login

            </Button>

            </Link>

          </div>

        </div>

      </nav>

      {/\* Hero Section \*/}

      <div className="relative min-h-screen flex items-center justify-center px-6">

        <div className="text-center max-w-4xl mx-auto">

          <h1

            className={`text-5xl md:text-7xl font-bold mb-6 text-white transition-all duration-1000 ${

              mounted ? 'opacity-100 translate-y-0' : 'opacity-0 translate-y-10'

            }`}

          >

            Showcase Your{' '}

            <span className="bg-gradient-to-r from-blue-400 via-purple-400 to-pink-400 bg-clip-text text-transparent animate-gradient">

              Amazing Projects

            </span>

          </h1>

          <p

            className={`text-xl text-gray-300 mb-12 transition-all duration-1000 delay-300 ${

              mounted ? 'opacity-100 translate-y-0' : 'opacity-0 translate-y-10'

            }`}

          >

            Connect with creators and explore diverse projects in our vibrant community.

          </p>

          <div

            className={`flex gap-6 justify-center transition-all duration-1000 delay-500 ${

              mounted ? 'opacity-100 translate-y-0' : 'opacity-0 translate-y-10'

            }`}

          >

            {/\* <Button

              variant="contained"

              className="bg-gradient-to-r from-blue-500 to-purple-500 hover:from-blue-600 hover:to-purple-600

                text-white px-8 py-3 transform hover:scale-105 transition-all duration-300"

            >

              Get Started

            </Button> \*/}

           <Link to="/home"> <Button

              variant="outlined"

              className="border-purple-400 text-purple-400 hover:bg-purple-400/10 px-8 py-3

                transform hover:scale-105 transition-all duration-300"

            >

              Explore Projects

            </Button>

            </Link>

          </div>

        </div>

       <Link to="/home"> <ScrollIndicator /></Link>

      </div>

    </div>

  );

};

export default LandingPage;

App.jsx

import { useEffect, useState } from 'react'

import { Routes, Route, useNavigate} from 'react-router-dom'

import Home from './Pages/Home/Home'

import SignUp from './Pages/SignUp/SignUp'

import LogIn from './Pages/LogIn/LogIn'

import CreateProject from './Pages/CreateProject/CreateProject'

import DashBoard from "./Pages/DashBoard/DashBoard";

import Contact from "./Pages/Contact/Contact";

import About from "./Pages/About/About";

import Profile from './Component/Profile'

import LandingPage from './Pages/LandingPage/LandingPage'

import ProjectDesciption from './Pages/ProjectDesciption/ProjectDesciption'

function App() {

  const navigate = useNavigate();

  useEffect(()=>{

    if(!localStorage.getItem('token')){

      console.log('hello world')

      navigate('/');

      // navigate('/login');

    }else{

      // console.log('bye')

      // navigate('/');

    }

  },[])

  return (

    <>

     <Routes>

      <Route path='/' element={<LandingPage/>}></Route>

      <Route path='/home' element={<Home/>}></Route>

      <Route path='/signup' element={<SignUp/>}></Route>

      <Route path='/login' element={<LogIn/>}></Route>

      <Route path='/add-project' element={<CreateProject/>}></Route>

      <Route path='/dashboard' element={<DashBoard/>}></Route>

      <Route path='/contact' element= {<Contact/>}></Route>

      <Route path='/about' element={<About/>}></Route>

      <Route path ='/profile' element={<Profile/>}></Route>

      <Route path='/projectDescription/:id' element={<ProjectDesciption/>}></Route>

     </Routes>

    </>

  )

}

export default App

SignUp.jsx

import React, { useState, useEffect } from "react";

import TextField from "@mui/material/TextField";

// import { Button } from "@mui/material";

import { Link, useNavigate } from "react-router-dom";

import { CircleUserRound, ArrowRight, ArrowLeft } from "lucide-react";

const GlowingOrb = ({ delay = 0, color1, color2 }) => (

  <div

    className="absolute rounded-full animate-pulse blur-xl opacity-20"

    style={{

      background: `radial-gradient(circle at center, ${color1}, ${color2})`,

      width: `${200 + Math.random() \* 300}px`,

      height: `${200 + Math.random() \* 300}px`,

      left: `${Math.random() \* 100}%`,

      top: `${Math.random() \* 100}%`,

      animationDelay: `${delay}s`,

      transform: "translate(-50%, -50%)",

    }}

  />

);

const SignUp = () => {

  const [step, setStep] = useState(1);

  const [mounted, setMounted] = useState(false);

  const [formData, setFormData] = useState({

    firstName: "",

    lastName: "",

    email: "",

    password: "",

    confirmPassword: "",

  });

  const [passMatch, setPassMatch] = useState("empty");

  const navigate = useNavigate(); // Added useNavigate for navigation after successful signup

  useEffect(() => {

    setMounted(true);

  }, []);

  const handleInputChange = (e) => {

    const { name, value } = e.target;

    setFormData((prev) => ({

      ...prev,

      [name]: value,

    }));

    if (name === "confirmPassword") {

      setPassMatch(value === formData.password);

    }

    if (name === "password") {

      if (formData.confirmPassword) {

        setPassMatch(value === formData.confirmPassword);

      }

    }

  };

  // Updated submitHandler function

  const submitHandler = async (e) => {

    e.preventDefault();

    console.log("Form submitted:", formData);

    if (formData.password !== formData.confirmPassword) {

      alert("Passwords do not match.");

      return;

    }

    const data = {

      firstName: formData.firstName,

      lastName: formData.lastName,

      email: formData.email,

      password: formData.password,

      confirmPass: formData.confirmPassword,

    };

    let response = await fetch("https://project-hub-backend-seven.vercel.app/api/v1/signup", {

      method: "POST",

      headers: {

        "Content-Type": "application/json",

      },

      body: JSON.stringify(data),

    });

    response = await response.json();

    console.log(response);

    if (response.success === "false") {

      alert(response.message);

    } else {

      navigate("/login");

    }

  };

  const nextStep = (e) => {

    e.preventDefault();

    setStep(2);

  };

  const prevStep = () => setStep(1);

  const textFieldStyles = {

    "& .MuiOutlinedInput-root": {

      backgroundColor: "rgba(255, 255, 255, 0.05)",

      "& fieldset": { borderColor: "rgba(255, 255, 255, 0.2)" },

      "&:hover fieldset": { borderColor: "rgba(255, 255, 255, 0.3)" },

      "&.Mui-focused fieldset": { borderColor: "#3B82F6" },

    },

    "& .MuiInputLabel-root": { color: "rgba(255, 255, 255, 0.7)" },

    "& .MuiOutlinedInput-input": { color: "white" },

  };

  return (

    <div className="relative min-h-screen bg-slate-900 flex items-center justify-center p-4 overflow-hidden">

      {/\* Animated Background \*/}

      <div className="absolute inset-0">

        {/\* Grid Pattern \*/}

        <div className="absolute inset-0 bg-[linear-gradient(to\_right,#1f2937\_1px,transparent\_1px),linear-gradient(to\_bottom,#1f2937\_1px,transparent\_1px)] bg-[size:4rem\_4rem] opacity-20" />

        {/\* Glowing Orbs \*/}

        <GlowingOrb color1="#4F46E5" color2="#2563EB" delay={0} />

        <GlowingOrb color1="#7C3AED" color2="#4F46E5" delay={2} />

        <GlowingOrb color1="#2563EB" color2="#3B82F6" delay={4} />

      </div>

      {/\* Sign Up Form \*/}

      <div

        className={`relative w-full max-w-md transform transition-all duration-1000

          ${mounted ? "opacity-100 translate-y-0" : "opacity-0 translate-y-12"}`}

      >

        <div className="bg-white/10 backdrop-blur-lg rounded-2xl shadow-xl overflow-hidden border border-white/20 p-8">

          <div className="text-center mb-8">

            <div className="inline-block p-3 rounded-full bg-blue-500/10 mb-4">

              <CircleUserRound size={40} className="text-blue-400" />

            </div>

            <h2 className="text-3xl font-bold text-white mb-2">Create Account</h2>

            <p className="text-gray-400">Step {step} of 2</p>

          </div>

          <form onSubmit={step === 1 ? nextStep : submitHandler} className="space-y-6">

            {step === 1 ? (

              <>

                <TextField

                  required

                  label="First Name"

                  name="firstName"

                  value={formData.firstName}

                  onChange={handleInputChange}

                  fullWidth

                  variant="outlined"

                  sx={textFieldStyles}

                />

# CHAPTER 6

**TESTING**

## INTRODUCTION

#### Testing Objectives

The following are the testing objectives:

-Testing is a process of executing a program with the intent of finding an error.

-A good test case is one that has a high probability of finding an as-yet-undiscovered error

-successful test is one that uncovers an as yet undiscovered error.

#### Testing Principles

The basic principles that guide software testing are as follows:

* + - * -All tests should be traceable to customer requirements.
      * -Tests should be planned long before testing begins.
      * -The pirate principle applies to software testing.

Pareto principle states that 80 percent of all errors uncovered during testing will likely be traceable to 20 percent of all program components.

Testing should begin “in the small “and progress toward testing “in the large.” Exhaustive testing is not possible.

#### LEVEL OF TESTING

There are different levels of testing

* Unit Testing
* Integration Testing
* System Testing

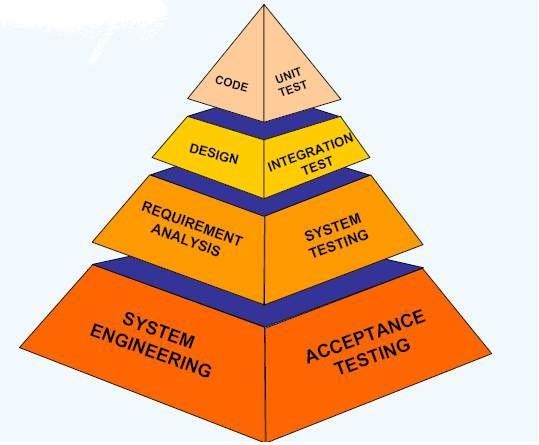


Figure 5.1: Testing pyramid

### Unit testing

Unit testing focuses verification effort on the smallest unit of software design,

the module. The important control parts are tested to uncover with in the boundary of the module. The module interface is tested to ensure that the information properly flows into and out of the program unit and boundary conditions are tested to ensure that the modules operate properly at boundaries established to limit or restrict processing. Test date is provided through testing screens.

### Integration testing

Integrating testing is a systematic technique for constructing Program structure while conducting tests to uncover error associates with interfacing. The objective isto take unit modules and built a program structure that has been directed by design.

* Integration Testing will test whether the modules work well together.
* This will check whether the design is correct.

### System testing

System testing is the process of testing the completed software as a part of the environment it was created for. It is done to ensure that all the requirements specified by the customer are met. System testing involves functional testing and performance testing. System Testing will contain the following testing:

* + - Functional Testing.
    - Performance Testing.
    - Function Testing will test the implementation of the business needs.
    - Performance Testing will test the non-functional requirements of the system like the speed, load etc

# CHAPTER 7

**CONCLUSION AND FUTURE SCOPE**

## CONCLUSION

## The "Project Hub" offers a straightforward and efficient solution for managing small projects. By bringing all project-related materials—documents, links, and resources—into one central location, it makes organizing, accessing, and sharing information much easier. This streamlined approach helps improve productivity, reduces time spent searching for files, and ensures that everyone involved in the project has what they need at their fingertips. With its user-friendly design, the platform is accessible even to those with minimal technical skills, making it an ideal choice for small teams or individual project managers looking for a simple and effective way to stay organized. Overall, the "Project Hub" enhances project management by offering an intuitive, all-in-one solution for smaller-scale projects.

## 7.2 FUTURE SCOPE

**1) Integration with Other Tools**: It could connect with popular project management tools like Trello or Asana, so you can manage everything in one place without switching between apps.

2) **Better Collaboration**: Adding features like live chat, video calls, and project-specific comment sections would help team members work together more effectively.

3) **Mobile App**: A mobile version of the platform would allow users to manage projects from their phones, making it easier to update and access project information on the go.

4) **Reports and Insights**: Adding tools to generate reports or track project progress with visual charts would help project managers keep things organized and spot any potential issues early on.

5) **Handling Larger Projects**: "Project Hub" could be expanded to support bigger and more complex projects, with features like task dependencies, budgeting, and more detailed timelines.

6) **Better Security**: As the platform grows, security becomes more important. Adding extra protection like two-factor authentication and data encryption would keep sensitive project information safe.

## BIBLIOGRAPHY

* <https://www.mindinventory.com/blog/mean-stack-vs-mern-stack/>
* <https://www.mongodb.com/mern-stack>
* https://www.ijert.org/research/performance-optimization-using-mern-stack-on-web-application-IJERTV10IS060239.pdf
* "MERN Stack for Web Development – A Comprehensive Guide" by K. Gopalan, V. Vijayakumar, and R. Vasantha. This paper was presented at the 2020 International Conference on Smart Electronics and Communication (ICOSEC)
* A Secure and Efficient Authentication System Using JSON Web Tokens (JWT)" by A. Kumar, S. Sharma, and R. Singh. This paper discusses the design and implementation of a secure authentication system using JWT and was published in the IEEE Access journal
* Implementing JWT in RESTful Web Services" by R. Gupta, A. Sharma, and N. Kumar. This paper provides a detailed guide on implementing JWT in RESTful web services and was published in the IEEE Journal on Web Services.·
* MERN Stack Full Tutorial & Project by Dave Gray. This video provides a detailed guide on building a MERN stack application from scratch, including setting up the server, database, routes, models, and more
* Node.js and Express.js - Full Course by freeCodeCamp.org. This comprehensive course teaches you how to use Node.js and Express.js to build a complex REST API and a MERN stack application.

## GITHUB PROJECT LINK

https://github.com/ravisingh6565/ProjectHub