

Index

No.	Title	Page No.
1.	Preface	2
2.	Acknowledgment	3
3.	Project Profile	4
4.	Introduction Of Project	5
5.	System Requirement	6
6.	Front-End & Back-End	7
7.	System Analysis	11
8.	Feasibility Study	12
9.	Functional Requirement	13
10.	Non-Functional Requirement	14
11.	Project Scheduling	15
12.	Data Flow Diagram	17
13.	Flow Chart	19
14.	ER-Diagram	26
15.	Data Directory	27
16.	Screenshot	29
17.	System Testing	34
18.	Implementation and Maintenance	35
19.	Limitation of the Project	37
20.	Bibliography	38

Preface

Theory is essential in any subject, but without practical application, it becomes ineffective, especially for computer science students. A computer student cannot become a proficient technologist without a hands-on understanding of their field. This project has provided a golden opportunity for us to bridge the gap between theoretical knowledge and practical implementation.

In today's competitive market, it is crucial to design and implement strategies that align with current trends and demands. As the saying goes, "Experience is the best teacher," and this project has been an invaluable experience, allowing us to learn about real-world applications in a meaningful way.

The project training in the B.C.A curriculum exposes students to the real world. The primary aim of this training is to provide practical experience and foster a better understanding of real-world challenges. This opportunity has not only enhanced our technical skills but also helped us develop our talents and qualities.

By applying theoretical knowledge in practical scenarios, we have gained valuable insights into its real-world implications. This project has been a significant part of our academic journey, preparing us for the challenges of the professional field.

Acknowledgement

We would like to express our sincere gratitude to Prof. Thakkur Jitu Sir, the Project Coordinator at Smt. J. J. Kundalia Commerce College, for his invaluable support and guidance throughout the entire project development process. His constant encouragement, meticulous attention to detail, and insightful feedback were pivotal in making this project a success. His dedication and unwavering commitment to our progress were instrumental in our learning and in overcoming challenges during the development.

We also extend our heartfelt thanks to the faculties who guided us and contributed significantly to the development of our project website. Their expertise, timely assistance, and constructive suggestions made it much easier to navigate through the complexities of the project. Their encouragement and willingness to share knowledge helped in honing our technical and problem-solving skills, which were crucial for the project's completion.

We are truly grateful for the resources and support provided by the college, which helped us undertake this project and apply theoretical knowledge in a practical context. The opportunity to work on this project has greatly enriched our learning experience and has prepared us for future challenges in our careers.

This project, developed by Chudasama Vishal, Hadani Vishnu, and Shah Rajkumar, is a result of collaborative effort, and we acknowledge the significant contributions of everyone involved. We are deeply appreciative of Prof. Thakkur Jitu's dedication, the faculty's guidance, and the collective effort that made this project possible.

Project Profile

Project Name	: Keep Notes – Store Your Notes Securely
Front-End	: JSP, JQuery, Ajax, HTML, CSS, JavaScript
Back-End	: Java Spring, Hibernate
Documentation Tool	: Microsoft Word 2021
Operating System	: Windows 11
Web Server	: Apache MySQL
Web Browser	: Google Chrome
Editor IDE	: Eclipse
Guided By	: Thakkur Jitu Sir
Submitted To	: Smt. J. J. Kundalia Commerce College
Project Duration	: 1.5 Months
Academic Year	: 2024-25
Team Members	: 1. Chudasama Vishal B. 2. Hadani Vishnu R. 3. Lakhani Dixit B.

Introduction of Project

➤ Introduction:

The project, "Keep Notes - Store Your Notes Securely," is a web-based application designed to allow users to manage their notes efficiently. The system ensures user data is stored securely while providing a seamless and user-friendly interface for managing notes. The application supports various features such as creating, editing, deleting, pinning, and un-pinning notes, along with setting custom colors for better

➤ User Objectives:

- The Enable users to securely create and manage their personal accounts.
- Allow users to write, save, edit, and delete notes with ease.
- Implement features for pinning important notes and un-pinning them as needed.
- Provide options to set custom colors for notes to enhance organization.
- Ensure data security and privacy using robust backend technologies

System Requirement

➤ Hardware Requirements:

- Processor: Dual-core CPU (Intel Core i3 or equivalent)
- RAM: 4GB (8GB recommended for smoother performance)
- Storage: 20GB free space for tools, frameworks, and project files
- OS: Windows 10/11, macOS, or a Linux distribution (Ubuntu, Fedora, etc.)

➤ Software Requirements:

- IDE: Eclipse IDE or IntelliJ IDEA (both run better with 8GB RAM)
- Web Server: Apache Tomcat (Version – 9.0)
- Database: MySQL (bundled with XAMPP)
- Java Version: JDK 17

Front-End Tools

➤ JSP (JavaServer Pages)

JSP is a technology used to create dynamic web content by embedding Java code directly within HTML pages. It simplifies the development of server-side views for web applications.

Key Features:

- **Dynamic Content Rendering:** Allows developers to embed Java logic within HTML for generating dynamic pages.
- **Tag Libraries:** Facilitates reusability with predefined tags and custom tag libraries.
- **Integration:** Works seamlessly with servlets and frameworks like Spring.
- **Scalability:** Suitable for building scalable web applications when combined with MVC frameworks.

In project, JSP was used for the view layer to present dynamic content to users, while integrating with JavaScript, jQuery, and AJAX for enhanced interactivity.

➤ JavaScript

JavaScript is a versatile, client-side scripting language used to create dynamic and interactive features on web pages. It enables seamless interaction between the user and the application.

- **Why Used JavaScript ?**

In this project, JavaScript was used as the foundation for implementing jQuery and AJAX functionalities. It enabled efficient

API calls, enhanced user interactions, and handled dynamic content updates, ensuring a smooth and responsive user experience

➤ JQuery

jQuery is a fast and lightweight JavaScript library that simplifies HTML document manipulation, event handling, and AJAX interactions. It is widely used for creating interactive and dynamic web pages.

- Why Used jQuery ?

In this project, jQuery was utilized to simplify DOM manipulations and streamline the integration of AJAX calls for real-time communication with the server, improving the user experience.

➤ Ajax: -

Ajax (Asynchronous JavaScript and XML) is a web development technique that allows data to be exchanged with the server asynchronously, enabling parts of a web page to update without refreshing the entire page.

- Why Used Ajax ?

AJAX was employed to call APIs dynamically via jQuery, allowing seamless server interactions and updating specific sections of the web page, such as form submissions or data display, without page reloads.

Back-End Tools

➤ Java Spring Framework

The Spring Framework is a powerful, open-source framework for building enterprise-level Java applications. It provides comprehensive infrastructure support to develop robust and maintainable web and backend applications.

Key Features:

- **Dependency Injection (DI):** Simplifies application development by managing dependencies and promoting loose coupling.
- **Spring Web MVC:** A robust Model-View-Controller (MVC) architecture for building scalable web applications.
- **Integration Support:** Easily integrates with various technologies like Hibernate, JPA, and third-party APIs.
- **Modular Architecture:** Provides flexibility to use only the required modules (e.g., Spring Web MVC without using Spring Boot).

In project, Spring Web MVC was used for managing HTTP requests, routing, and controlling the flow of the application.

➤ **Hibernate Framework**

Hibernate is an advanced Object-Relational Mapping (ORM) framework that simplifies interaction with relational databases by mapping Java objects to database tables.

Key Features:

- ORM: Eliminates the need for complex SQL queries by handling database operations through Java objects.
- HQL (Hibernate Query Language): A database-independent query language that allows for seamless switching between databases.
- Automatic Table Generation: Can generate database schemas directly from entity classes.
- Lazy/Eager Loading: Efficient data fetching mechanisms to optimize performance.

In project, Hibernate was used to manage database interactions, ensuring a seamless mapping of database tables to Java classes while abstracting the complexity of SQL.

System Analysis

System analysis is an essential part of any project management. It involves understanding and specifying in detail what an information system should do. Surprisingly, it's an area that often receives far too little attention.

➤ Scope of the System:

The application is targeted toward individuals looking for a secure and efficient way to manage personal or professional notes. It offers an intuitive interface and advanced note-management features, making it suitable for students, professionals, and general users. The system is accessible via any modern web browser and ensures responsive design for better usability.

➤ Problem Statement

In the digital age, users often struggle to organize their notes effectively while ensuring data security. Existing solutions may lack certain functionalities, such as categorization, color coding, or secure access. This project aims to address these challenges by providing a secure, feature-rich platform that simplifies note management.

Feasibility Study

The feasibility study evaluates the project's practicality and ensures its successful implementation by analyzing key aspects like technical, economic, and operational feasibility, along with project scheduling.

➤ **Technical Feasibility:**

The project uses robust technologies like Java Spring, Hibernate, JSP, jQuery, and AJAX, ensuring scalability, security, and compatibility. The chosen tools provide a reliable foundation for the application's performance and long-term maintainability.

➤ **Economic Feasibility:**

The project minimizes costs by utilizing open-source tools and frameworks, making it a budget-friendly solution while ensuring high-quality results. The investment in development time and resources is justified by the value the system provides to users.

➤ **Operational Feasibility:**

The system's intuitive interface and user-friendly features ensure it can be easily adopted and operated by users. The application's responsive design further enhances its usability.

➤ **Project Schedule:**

The project was developed over a planned timeline, divided into key phases: requirement analysis, system design, development, testing, and deployment. Adhering to this schedule ensured the timely delivery of a functional and efficient system.

Functional Requirements

- **User Account Management:**

Users can register, log in, and manage their accounts securely.

- **Note Management:**

- Create new notes with a title and content.
- Save and edit existing notes.
- Delete unwanted notes permanently.
- Pin and un-pin important notes for easy access.
- Set and modify colors for notes for better visual organization.

- **Security:**

Secure user authentication and data encryption.

Non-Functional Requirements

Non-functional requirements define the criteria that the system must meet in terms of performance, usability, reliability, and other quality attributes that ensure the system operates efficiently and effectively.

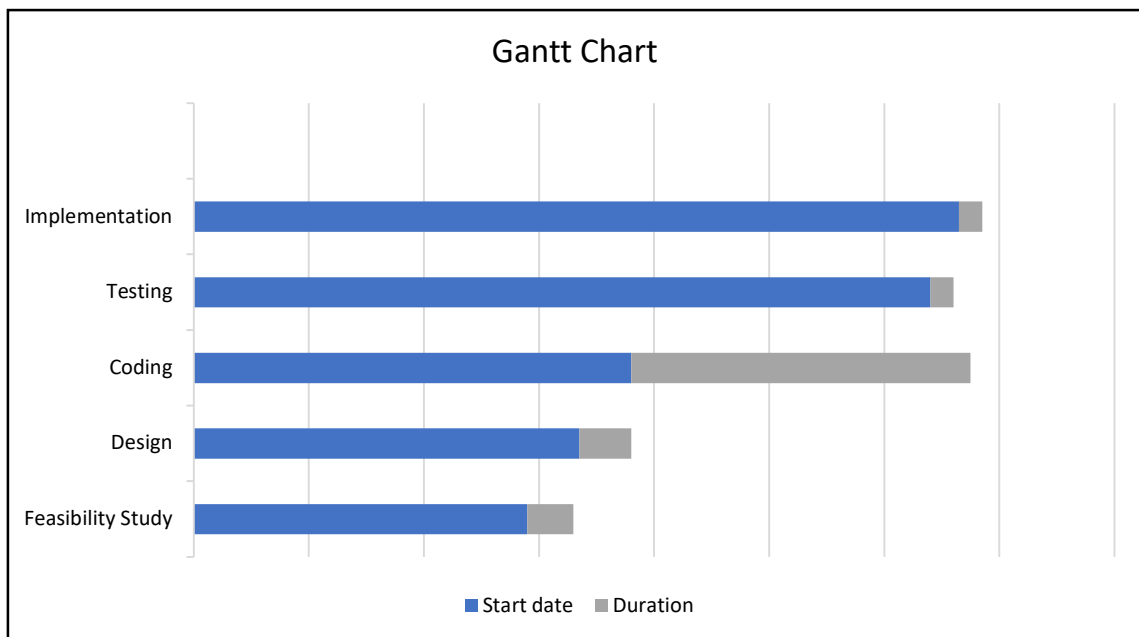
- **Performance:** The system must be responsive, ensuring minimal loading times and quick interactions, especially when creating, editing, or saving notes. It should support multiple users simultaneously without significant slowdowns.
- **Usability:** The user interface should be intuitive and easy to navigate for users of all skill levels. Clear instructions and well-placed buttons will allow users to quickly understand and utilize the system's features.
- **Scalability:** As the number of users grows, the system should be able to scale efficiently. The application should handle an increasing number of notes and users without compromising performance.
- **Reliability:** The system must be highly reliable, with minimal downtime. Data should be consistently stored and retrieved without errors. The system should also have backup mechanisms to avoid data loss.
- **Maintainability:** The system should be designed for easy maintenance, with well-documented code and a modular architecture. This will allow developers to implement updates, fixes, and new features efficiently without disrupting the system's functionality.
- **Compatibility:** The system should be compatible across different browsers (Chrome, Firefox, Edge, etc.) and devices (desktop, mobile). It should offer a consistent experience on all platforms.

Project Scheduling

➤ Gantt Chart: -

A Gantt chart is a type of bar chart, developed by Henry Gantt, which illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements comprise the work breakdown structure of the projects.

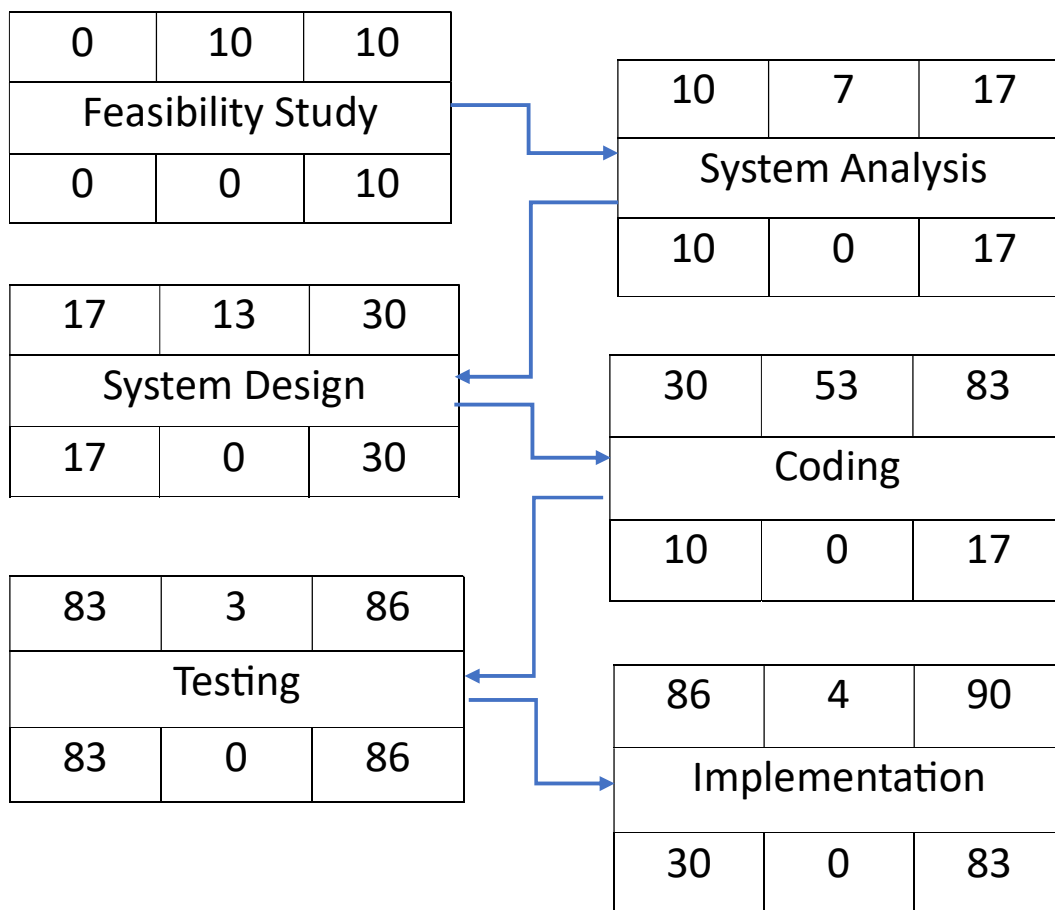
Some Gantt charts also show current schedule using percent- complete shadings. Gantt chart is also known as a Timeline chart.



➤ Pert Chart: -

A PERT chart is a project management tool used to schedule, organize, and coordinate tasks within a project. PERT stands for Program Evaluation Review Technique, a methodology developed by the U.S. Navy in the 1950s to manage the Polaris submarine missile program.

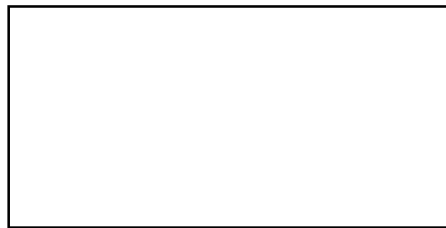
Early Start	Duration	Early Finish
Activity		
Late Start	Stack	Late Finish



Data-Flow Diagram

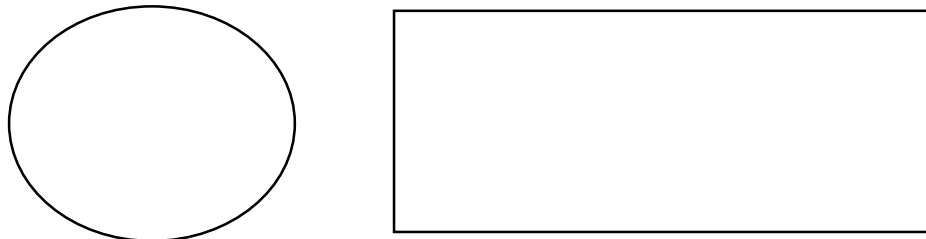
A Data Flow Diagram (DFD) is a graphical representation of the flow of data through an information system, illustrating its process aspects. It serves as a preliminary step to provide an overview of the system's functionality before detailed design. The DFD components include:

➤ **External Entity (Terminator):**



An external entity refers to a person, an institution or a department that is outside of the system being modelled. It is a thing where data originates (data source) and is usually a final destination (data sink). As such an entity is often referred to as a terminator.

➤ **Process (Presentation): -**



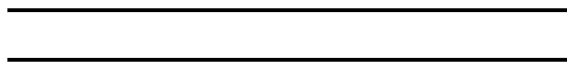
A Process is represented graphically as a circle Rectangle with rounded edge.

➤ **Data Flow: -**



It is used to describe the more event of data or physical material from one part of the system to another part.

➤ **Data Store: -**



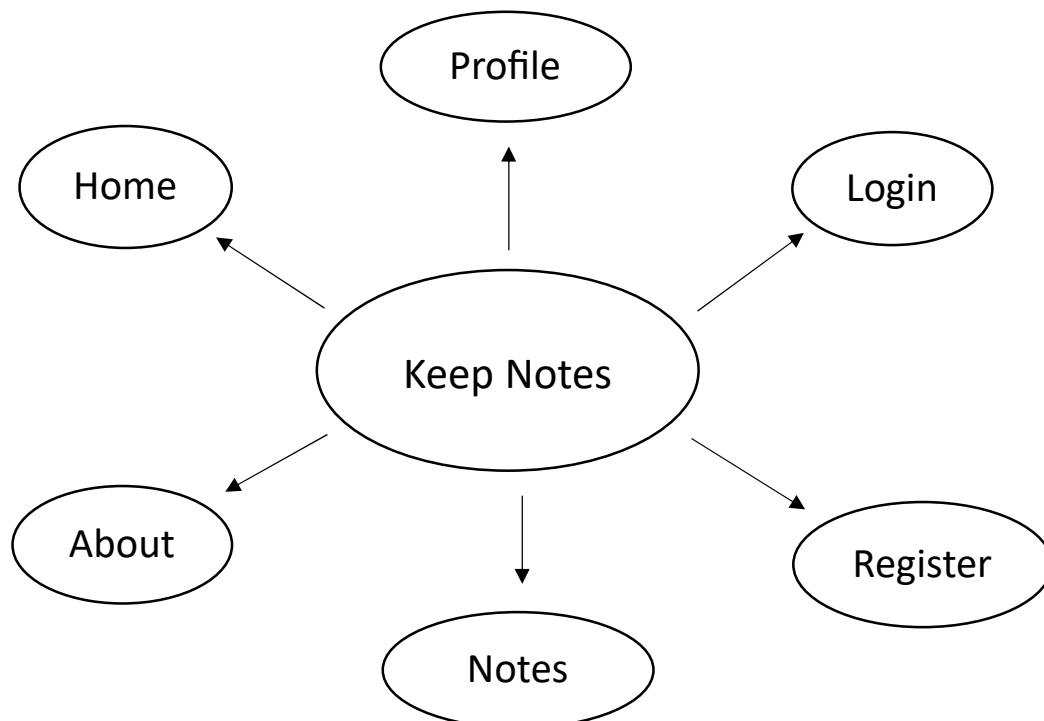
It is used to model a collection of data packets at rest
(1) Files (2) Tapes (3) Punched cards etc.

➤ **Output: -**



This box represents data production during human interaction computer output data generated by system.

➤ **Context Level Diagram: -**



Flow Chart

A flowchart is a visual representation of a process or workflow, commonly used in various fields such as software development, business analysis, and project management. It uses standardized symbols and connectors to depict the sequence of steps and decisions involved in completing a task or achieving a goal. Flowcharts are invaluable tools for clarifying complex processes, identifying bottlenecks, and improving efficiency.

The key elements of a flowchart include:

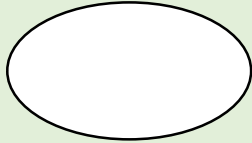


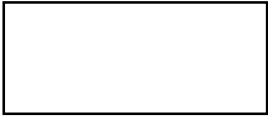
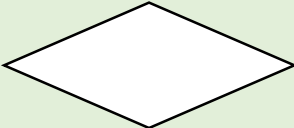
- **Symbols:** Each step or action in a process is represented by a specific symbol, such as a rectangle for a process step, a diamond for a decision point, or an oval for the start and end points of the process.
- **Arrows and Lines:** These connectors show the flow of control or data between the symbols, indicating the sequence of actions or decisions.
- **Decision Points:** Flowcharts use diamonds to denote decision points where different actions or paths can be taken based on conditions or inputs.

Flowcharts offer several benefits:

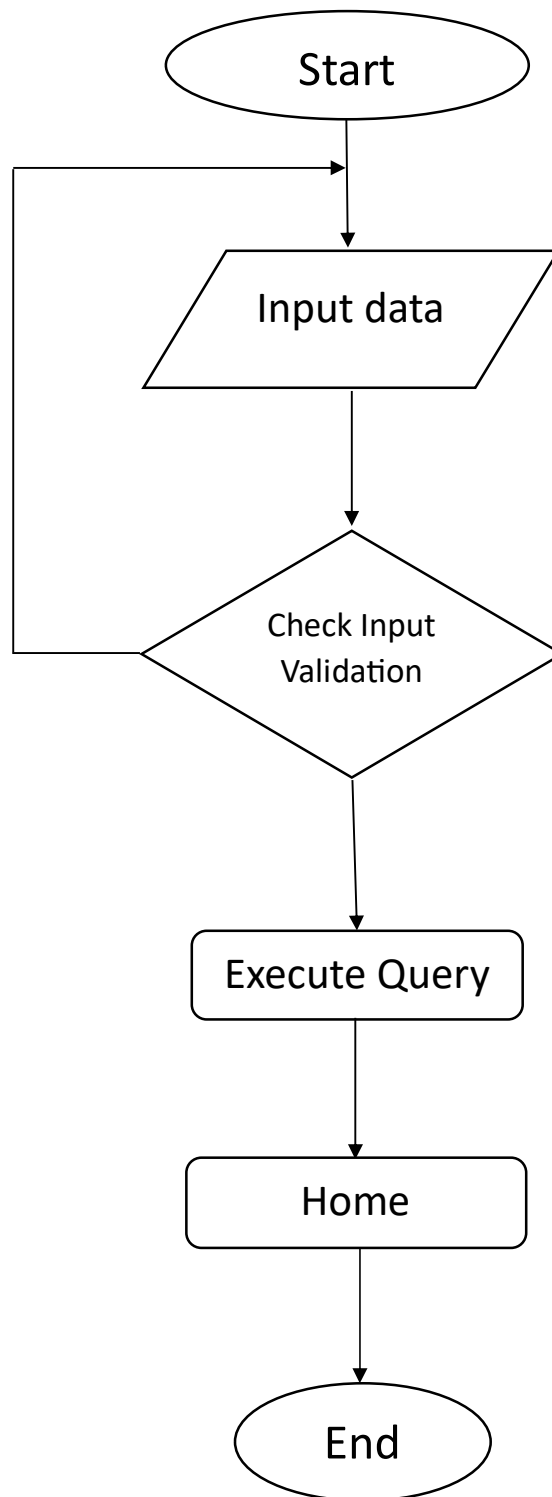
- **Clarity and Understanding:** They provide a clear visual representation of complex processes, making it easier for stakeholders to understand how tasks are performed and how data flows within a system.
- **Communication Tool:** Flowcharts serve as effective communication tools, enabling teams to discuss, analyze, and improve processes collaboratively.

- **Process Improvement:** By visually mapping out processes, flowcharts facilitate identification of inefficiencies, redundancies, or opportunities for optimization, leading to improved workflows and resource utilization.

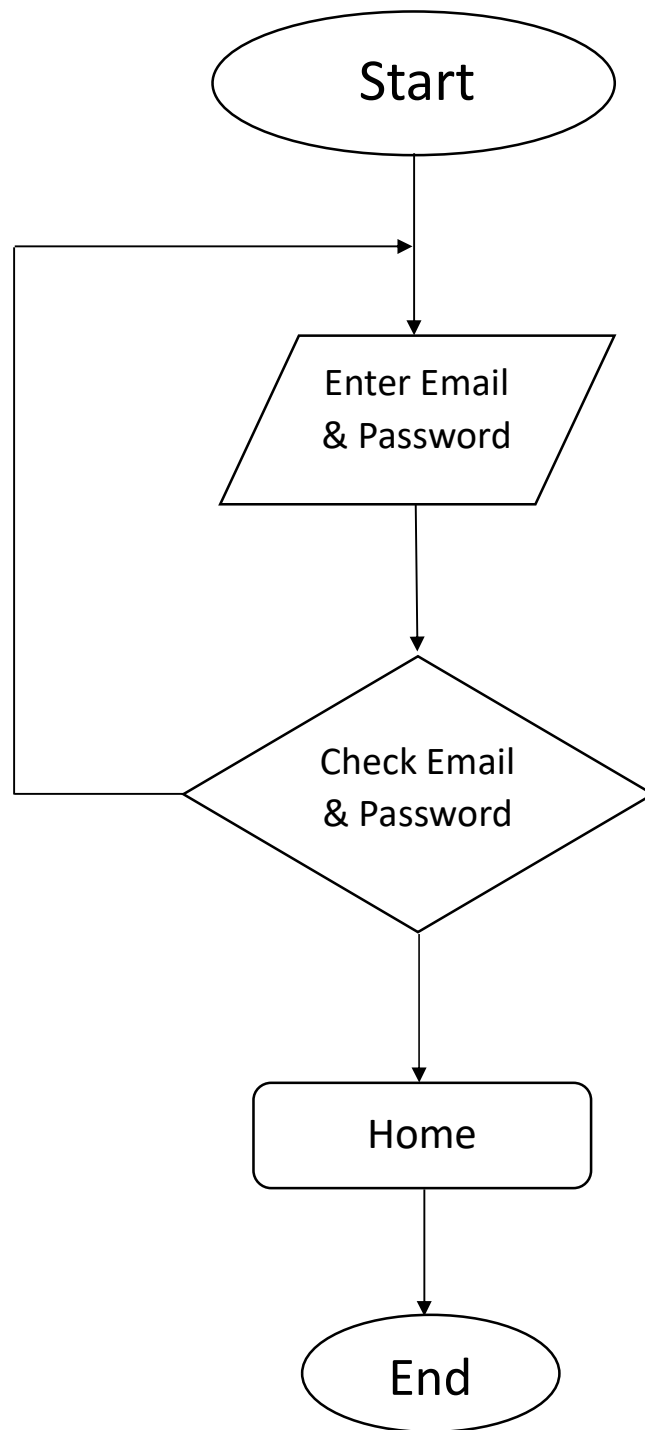
➤ **Program Flowchart Shapes: -**

Symbol	Name	Function
	Start/end	An Oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Input/Output	A parallelogram represents input or output
	Process	A rectangle represents a process
	Decision	A diamond indicates a decision

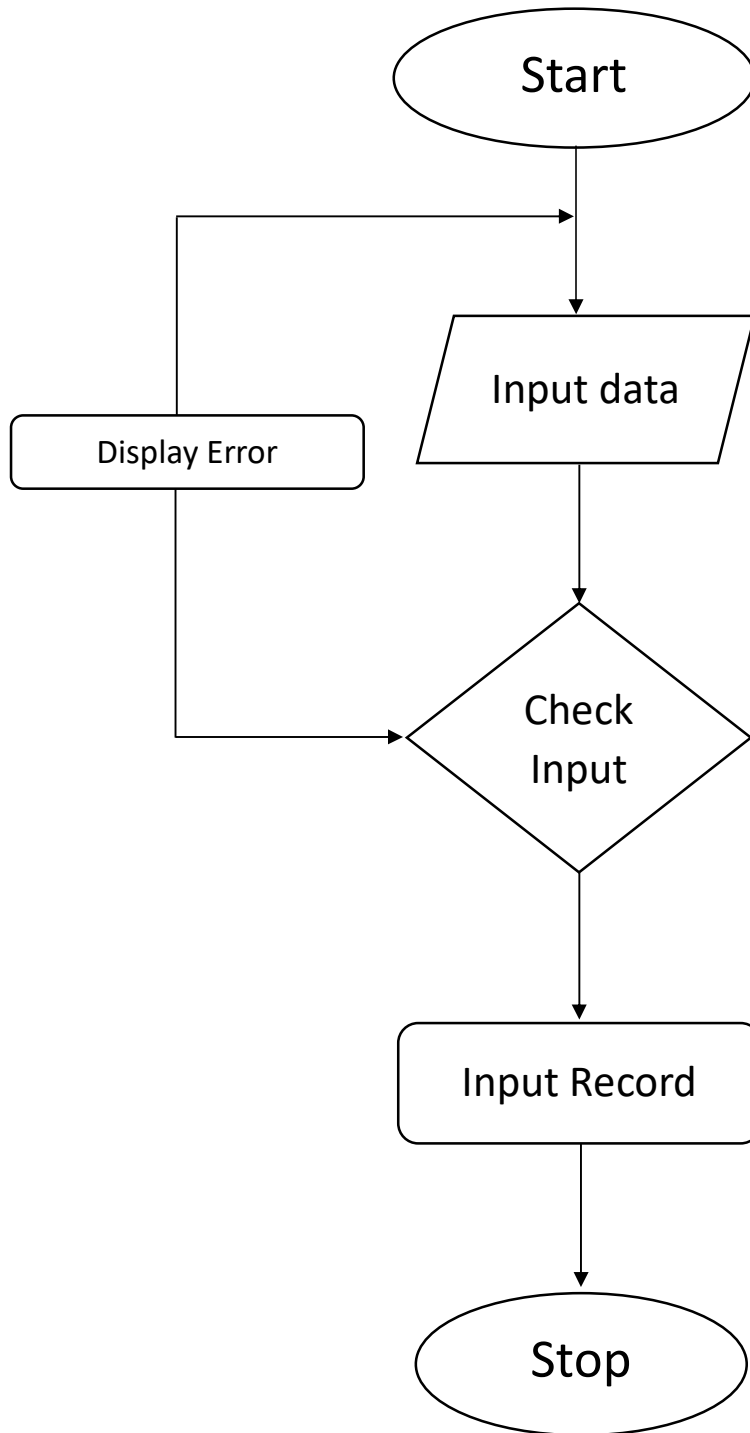
➤ Sign up:



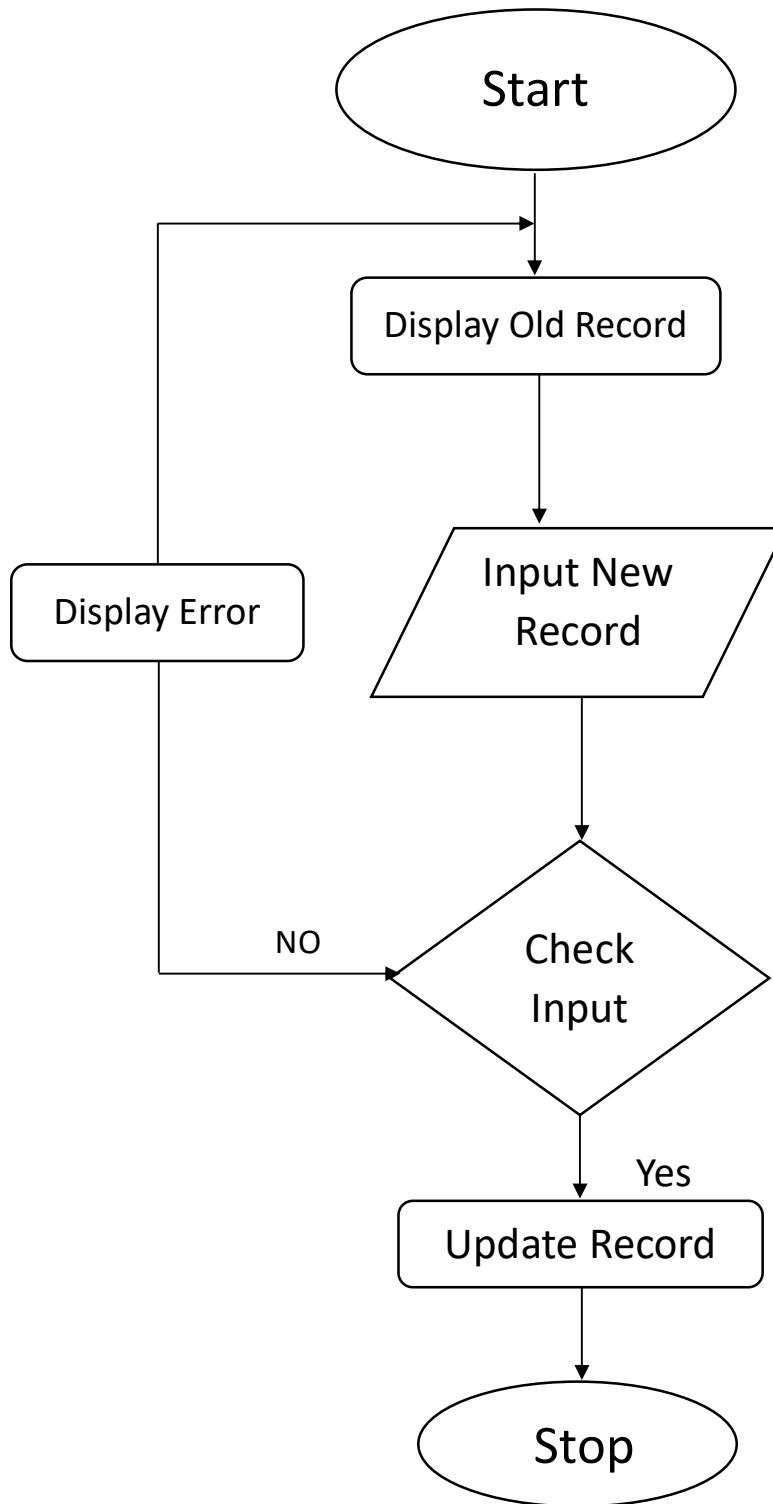
➤ **User and Admin Login:**



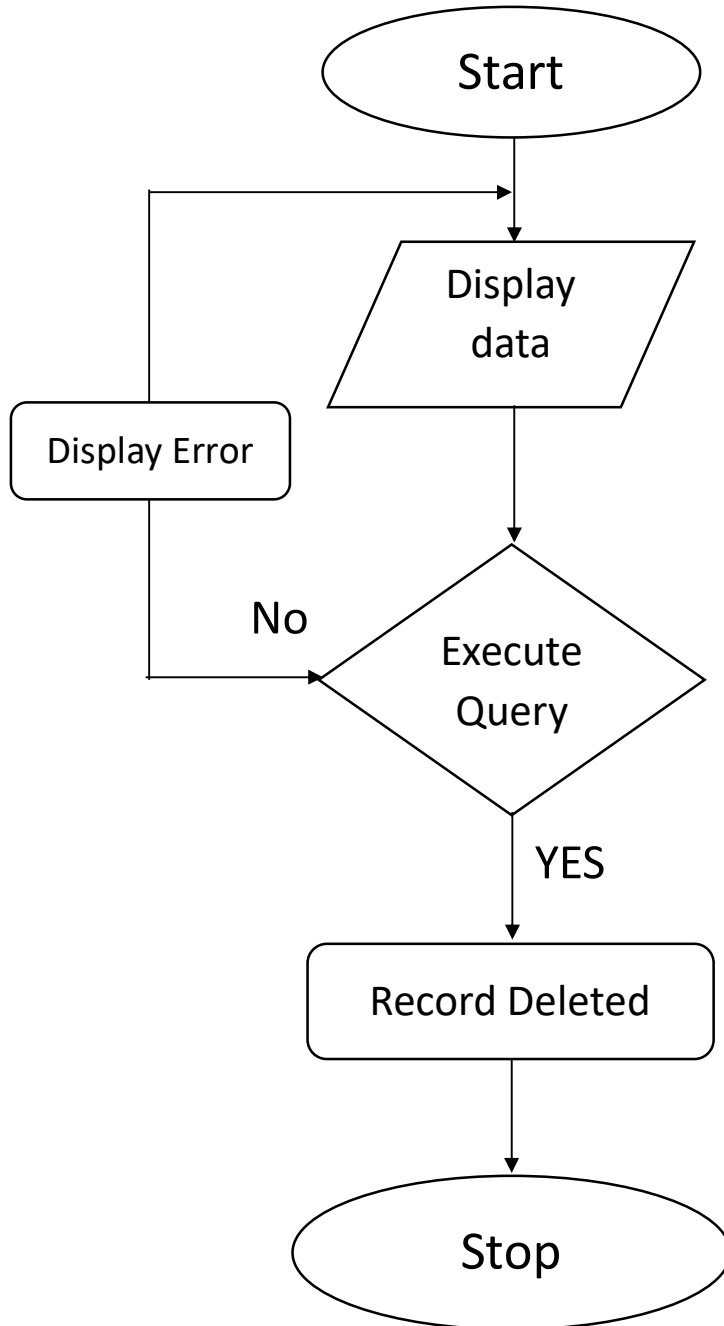
➤ **Insert Record: -**



➤ **Update Record: -**



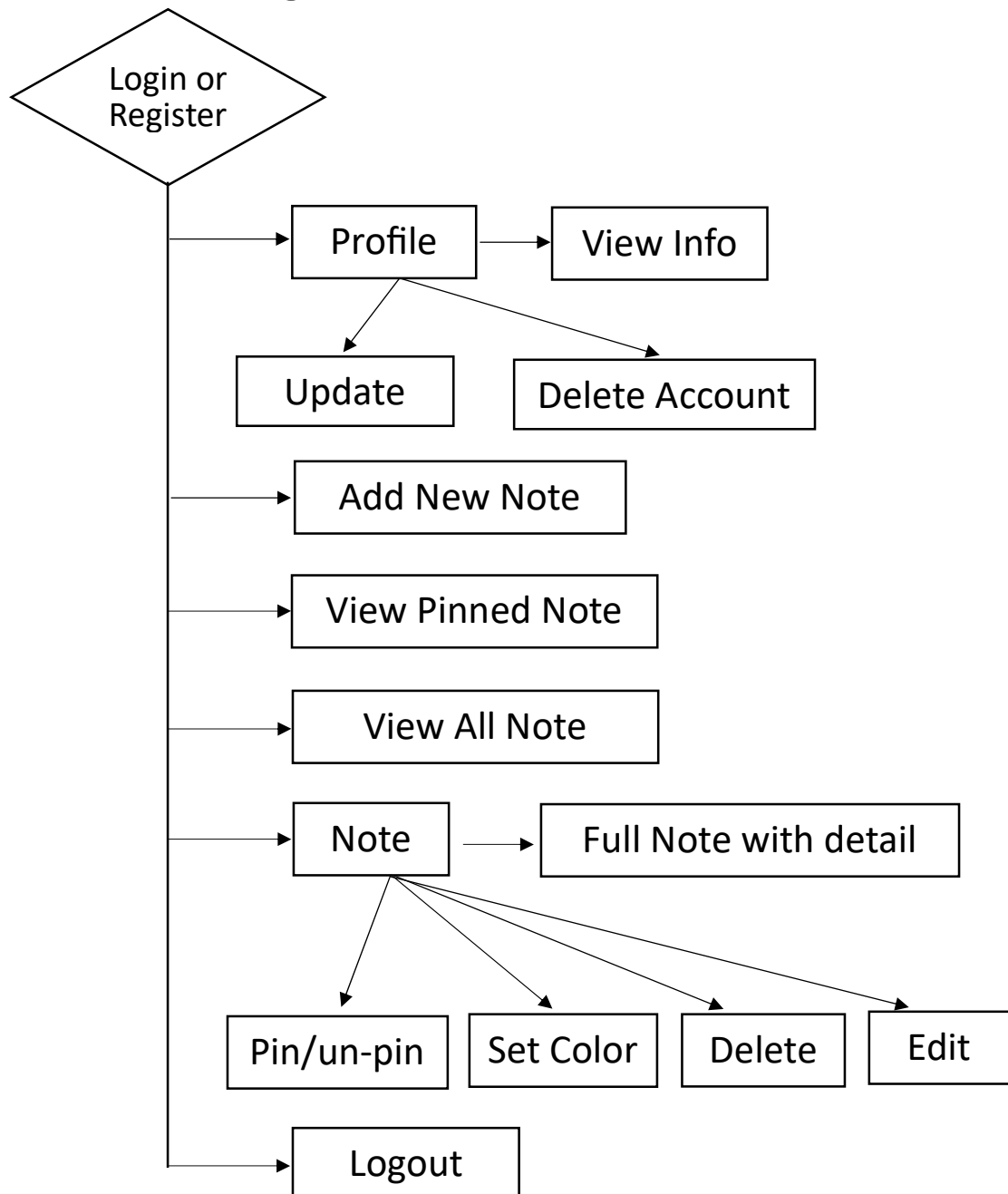
➤ **Delete record:**



E-R Diagram

E-R Diagram is one of the better ways to communicate Different of a component of a system. They are also too easy to Understand by everyone. They offer an overview of the entire System.

❖ User E-R Diagram



Data Directory

A data dictionary is a catalog-a-repository-of the elements in a system as the name suggests, these elements centre on data and the way they are structured to meet users' requirements and organization needs in a data dictionary you will find a list of all elements composing the data flowing through a system.

Database Name: keep_notes_db

No.	Table Name	Description
1	users	Store the info about Users
2	notes	Store the all notes with information

Tables Information: -**❖ users table:**

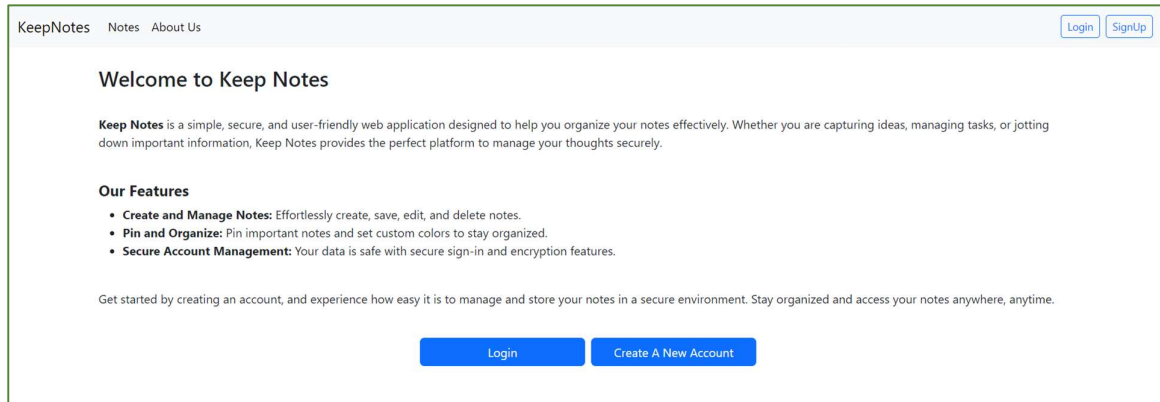
Key	Name	Type	Size	Constraints
1	id	bigint	20	PRIMARY KEY, AUTO_INCREMENT
2	first_name	varchar	255	NOT NULL
3	last_name	varchar	255	NOT NULL
4	email	varchar	255	Unique index, NOT NULL
5	password	varchar	255	NOT NULL
6	phone_number	number	15	NOT NULL
7	create_date	datetime	6	NOT NULL
8	update_date	datetime	6	NOT NULL

➤ notes table : -

Key	Name	Type	Size	Constraints
1	id	bigint	20	PRIMARY KEY, AUTO_INCREMENT
2	user_id	bigint	20	FOREIGN KEY, NOT NULL
3	title	varchar	255	None
4	description	varchar	255	None
5	pinned	bit	1	DEFAULT 0
6	bg_color	varchar	255	--
7	created_date	datetime	6	NOT NULL
8	updated_date	datetime	6	NOT NULL

Screenshots

Index page:



The screenshot shows the index page of the 'Keep Notes' application. At the top, there is a navigation bar with 'KeepNotes', 'Notes', and 'About Us' links, and 'Login' and 'SignUp' buttons. The main content area has a heading 'Welcome to Keep Notes' followed by a paragraph describing the application. Below this is a section titled 'Our Features' with three bullet points: 'Create and Manage Notes', 'Pin and Organize', and 'Secure Account Management'. At the bottom, there is a paragraph encouraging users to create an account and two buttons: 'Login' and 'Create A New Account'.

KeepNotes Notes About Us Login SignUp

Welcome to Keep Notes

Keep Notes is a simple, secure, and user-friendly web application designed to help you organize your notes effectively. Whether you are capturing ideas, managing tasks, or jotting down important information, Keep Notes provides the perfect platform to manage your thoughts securely.

Our Features

- **Create and Manage Notes:** Effortlessly create, save, edit, and delete notes.
- **Pin and Organize:** Pin important notes and set custom colors to stay organized.
- **Secure Account Management:** Your data is safe with secure sign-in and encryption features.

Get started by creating an account, and experience how easy it is to manage and store your notes in a secure environment. Stay organized and access your notes anywhere, anytime.

Login Create A New Account

User Register page:



The screenshot shows the user registration page of the 'Keep Notes' application. It features a navigation bar at the top with 'KeepNotes', 'Notes', and 'About Us' links, and 'Login' and 'SignUp' buttons. The main heading is 'Register Here'. Below this, there are input fields for 'First Name', 'Last Name', 'Email', 'Password', 'Confirm Password', and 'Phone Number'. A 'Submit' button is located at the bottom left of the form area.

KeepNotes Notes About Us Login SignUp

Register Here

First Name

Last Name

Email

Password

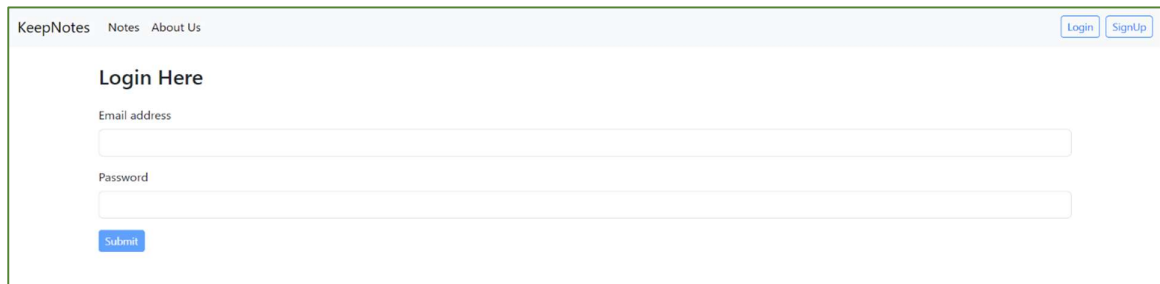
Confirm Password

Phone Number 

Submit

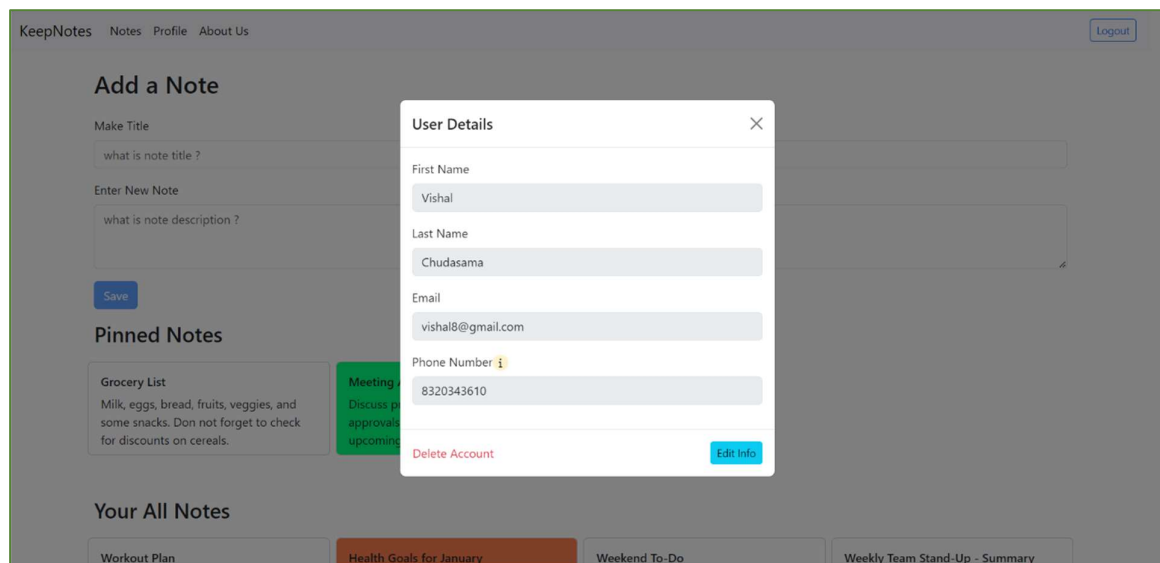
Keep Notes – Store Your Notes Securely

User Login page:



The screenshot shows the 'Login Here' page of the 'KeepNotes' application. The header includes 'KeepNotes', 'Notes', and 'About Us' links, along with 'Login' and 'SignUp' buttons. The main form has two input fields: 'Email address' and 'Password', followed by a 'Submit' button.

User Profile:



The screenshot shows the 'User Profile' page of the 'KeepNotes' application. The header includes 'KeepNotes', 'Notes', 'Profile', and 'About Us' links, along with a 'Logout' button. The main content area is divided into three sections: 'Add a Note' with input fields for title and description, a 'Save' button, and a 'Pinned Notes' section showing a 'Grocery List' and a 'Meeting' note. Below these is a 'Your All Notes' section with a list of notes: 'Workout Plan', 'Health Goals for January', 'Weekend To-Do', and 'Weekly Team Stand-Up - Summary'. A 'User Details' modal is open, displaying fields for 'First Name' (Vishal), 'Last Name' (Chudasama), 'Email' (vishal8@gmail.com), and 'Phone Number' (8320343610), along with 'Delete Account' and 'Edit Info' buttons.

Keep Notes – Store Your Notes Securely

Notes Home page:

KeepNotes

Notes

Profile

About Us

Logout

Add a Note

Make Title

what is note title ?

Enter New Note

what is note description ?

Save

Pinned Notes

Grocery List

Milk, eggs, bread, fruits, veggies, and some snacks. Don not forget to check for discounts on cereals.

Meeting Agenda

Discuss project milestones, budget approvals, team feedback, and upcoming deadlines.

Your All Notes

Workout Plan

Morning run: 3km, strength training: legs and core, evening: stretching and yoga.

Health Goals for January

Begin each day with 10 minutes of meditation to improve focus and reduce stress. Commit to drinking at least 2 liters of water daily and tracking meals for balanced nutrition.

Weekend To-Do

Clean the house, finish the pending assignments, and plan a day trip to the mountains.

Weekly Team Stand-Up - Summary Report

Date: January 4, 2025

Summary of Today Stand-Up Meeting:

1. ****Key Achievements:****

- The marketing team successfully launched the social media campaign for the new product line, achieving a 25% increase in

Book Recommendation

"Atomic Habits" by James Clear. A practical guide to building good habits and breaking bad ones.

Travel Packing List

Clothes, toiletries, charger, passport, tickets, sunscreen, and travel guide.

Recipe Notes

Spaghetti Aglio e Olio: Olive oil, garlic, chili flakes, parsley, and parmesan cheese. Quick and delicious!

Project Alpha - Status Update

Date: January 3, 2025

Today meeting focused on the progress of Project Alpha. The main points discussed were:

1. ****Progress Update:****

- The development team has completed the initial implementation of the

Project Outline

Research the topic, prepare a draft, review with the team, and finalize the presentation.

College Project Timeline

Finalize the project topic by the end of this week, ensuring it aligns with the syllabus. Divide tasks among team members and create a shared timeline for progress tracking. Complete

Dinner Ideas

Plan a simple dinner: grilled chicken, roasted vegetables, and mashed potatoes. Keep it healthy but satisfying, and maybe try a new sauce recipe.

Home Renovation Plan

Start with repainting the living room walls, focusing on lighter shades for a fresh look. Replace the old curtains and consider adding some indoor plants for a cozy vibe. Check

Buy Essentials

Toothpaste, shampoo, face wash, and detergent. Restock pantry basics too.

Office Presentation Prep

Finish the slides by tonight, focusing on clear visuals and concise points. Review the data accuracy and rehearse twice to ensure smooth delivery. Check the projector and audio setup in

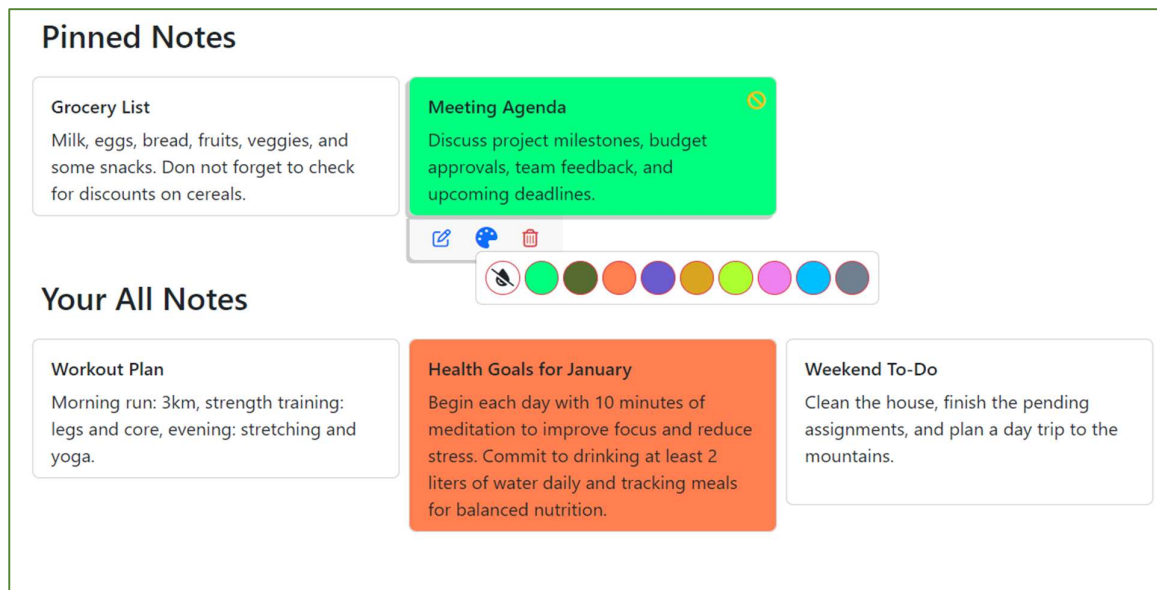
Personal Development Goals for 2025

1. Skill Building: Enroll in a professional certification course related to software development and dedicate 5 hours per week to learning.

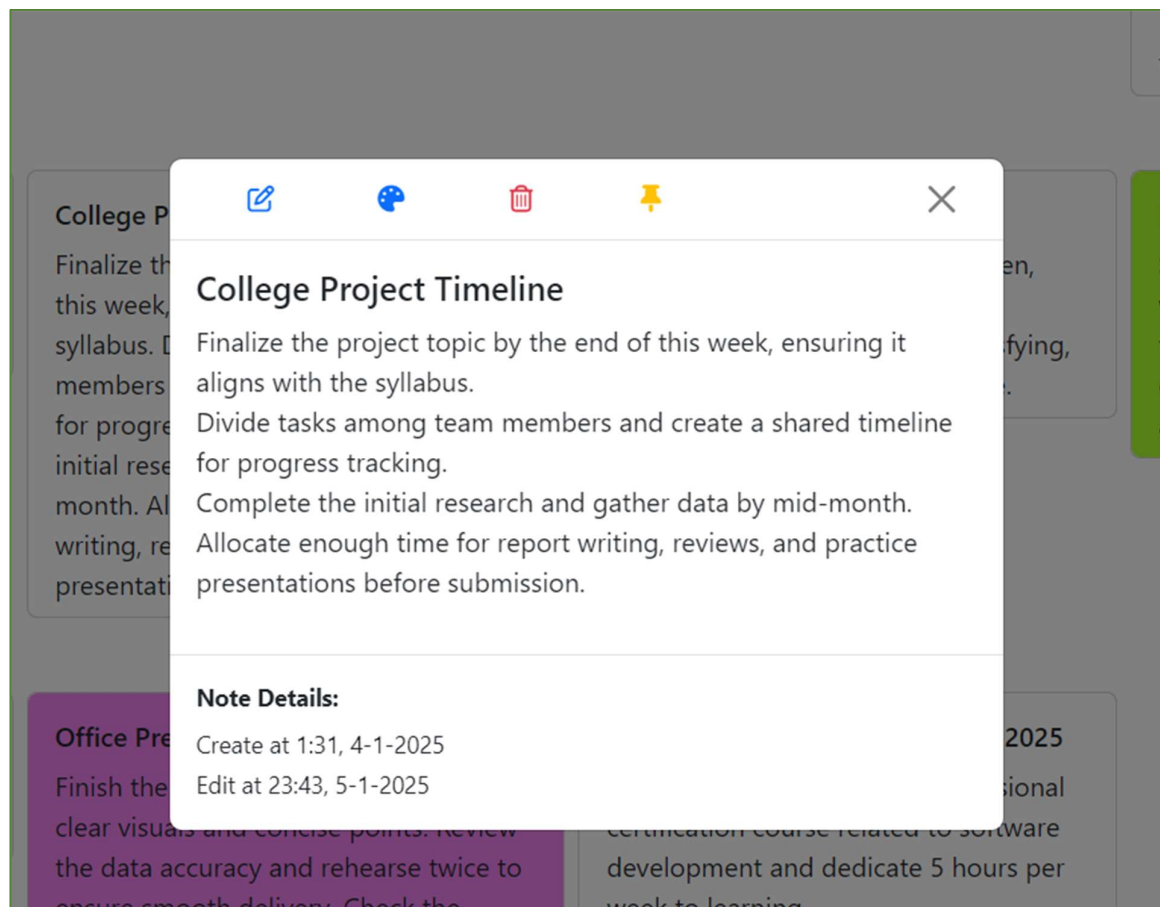
2. Fitness: Set a goal of running 5 kilometers non-stop

Keep Notes – Store Your Notes Securely

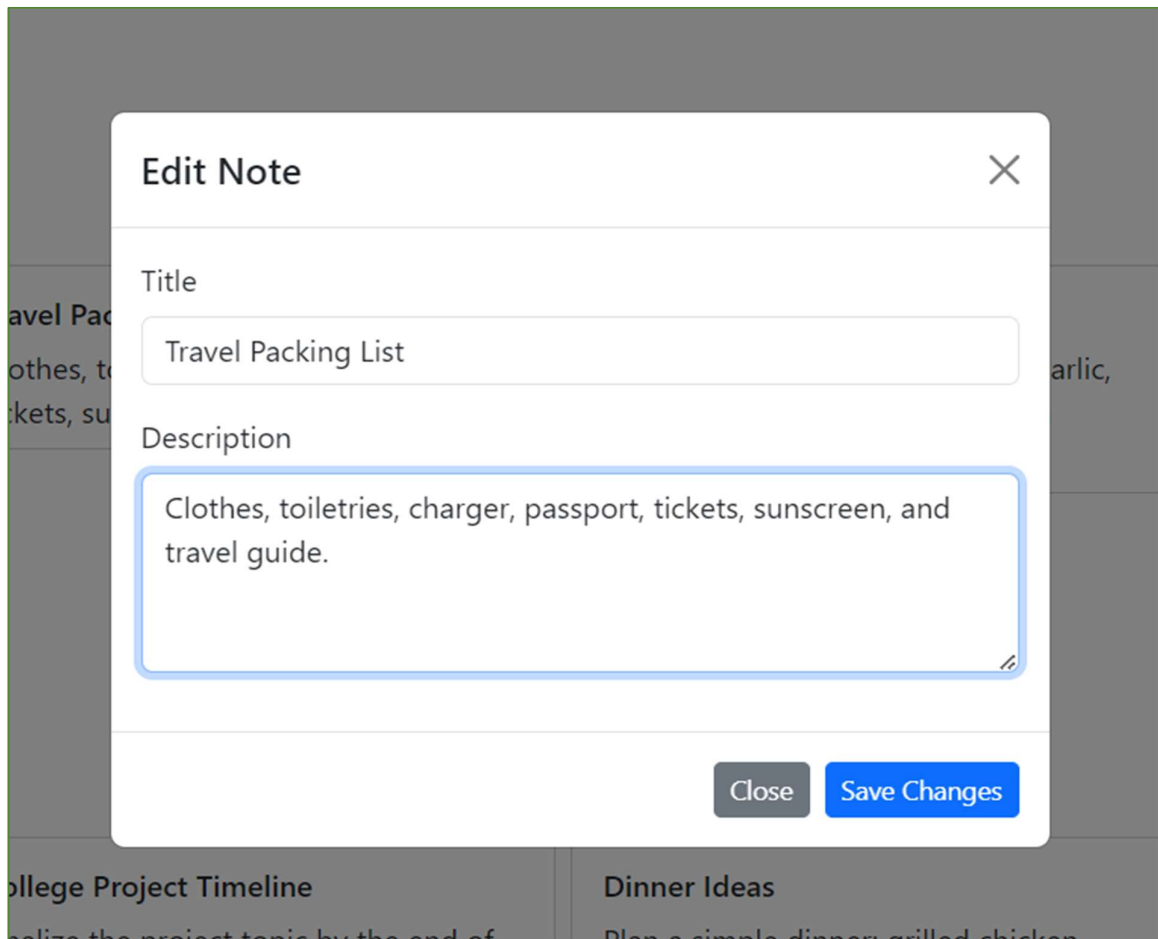
Set Note Colour or another Icons:



Open Full Note with Detail:



Edite Note:



Edit Note ✕

Title

Travel Packing List

Description

Clothes, toiletries, charger, passport, tickets, sunscreen, and travel guide.

Close Save Changes

System Testing

The System testing evaluates the entire application to ensure it meets functional and non-functional requirements. Key aspects include:

- **Functional Testing:** Verified essential features like user authentication, and CRUD operations for notes, ensuring proper functionality.
- **Usability Testing:** Tested the user interface for responsiveness, intuitive navigation, and meaningful error messages.
- **Compatibility Testing:** Ensured the system works seamlessly across popular browsers (Chrome, Firefox, Edge) and operating systems (Windows, macOS).
- **Performance Testing:** Assessed system behavior under load and measured response times for actions like saving or deleting notes.
- **Security Testing:** Validated against vulnerabilities such as SQL injection and XSS, with secure session management.
- **Database Testing:** Checked data integrity and performance of database operations during concurrent usage.
- **Regression Testing:** Ensured no existing features were affected by bug fixes or updates.

Implementation & Maintenance

➤ Implementation:

The project was implemented using the Spring Web MVC framework, Hibernate ORM for database interactions, and JSP for the view layer. The following steps outline the implementation process:

- **Requirement Analysis:** The functional and non-functional requirements of the system were gathered and analyzed.
- **System Design:** The MVC architecture was adopted, with the application divided into Controller, Service, Repository, and View layers.
- **Database Setup:** MySQL was used to design and create tables for managing users and notes. Hibernate was configured to map these tables to Java entities.
- **Frontend Development:** JSP, Bootstrap, and jQuery were utilized for creating user-friendly and responsive web pages. AJAX was used for seamless data interaction.
- **Backend Development:** Business logic and request handling were implemented using Spring controllers and services.
- **Testing:** Comprehensive testing was conducted for functionality, usability, and performance using both manual and automated methods.

➤ **Maintenance:**

After Regular maintenance is critical for the smooth functioning and reliability of the application. The key activities include:

- **Bug Fixes:** Identifying and resolving any issues reported by users during system usage.
- **Database Management:** Ensuring data integrity and performing routine database backups to prevent data loss.
- **Feature Updates:** Incorporating new features or modifying existing ones based on user feedback and evolving requirements.
- **System Monitoring:** Monitoring server performance and optimizing resource usage to ensure scalability and reliability.
- **Security Updates:** Regularly updating dependencies and implementing enhanced security measures to protect against vulnerabilities.
- **Documentation:** Keeping all project documentation up-to-date to assist in further development and maintenance.

Limitation of the Project

➤ **Single User Login at a Time**

The system currently does not support multiple concurrent logins for the same user account, limiting usability in shared or collaborative scenarios.

➤ **No Real-Time Synchronization**

Notes are not synchronized in real-time across multiple devices, making it less suitable for users who need immediate updates.

➤ **Limited Security Features**

While basic authentication is implemented, advanced security measures like two-factor authentication or encryption for stored notes are not included.

➤ **Static User Roles**

The system does not include role-based access controls or support for hierarchical permissions, which may limit its scalability for larger teams.

➤ **Dependency on Internet Connection**

The application requires an active internet connection to function, which might be a limitation for users in areas with unstable connectivity.

➤ **Minimal Search Capabilities**

The search functionality is basic and does not support advanced filtering or tagging, which could make it harder for users to locate specific notes in large datasets.

Bibliography

Maven Repository

The website <https://mvnrepository.com> was used to download the required JAR files for Spring, Hibernate, and other dependencies.

Spring Framework Documentation

<https://spring.io> The official Spring Framework documentation provided detailed insights into configuring Spring MVC and integrating it with Hibernate for backend development.

Google Keep

<https://keep.google.com> served as inspiration for the project design and functionality, offering a reference for creating a similar notes-keeping system.

Apache Tomcat

<https://tomcat.apache.org> website provided resources for setting up and configuring the Tomcat server for deploying and testing the application.

Bootstrap Documentation

<https://getbootstrap.com/docs/5.3> The official Bootstrap documentation helped in designing a visually appealing and responsive user interface.

Thank You