KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
AN AUTONOMOUS INSTITUTION - ACCREDITED BY NAAC WITH 'A' GRADE
Narayanaguda, Hyderabad.

THE SMART HABIT TRACKER

BY:
HIMABINDHU 22BD1A663P
PHANIDATTA 22BD1A662V
NISHANTH 22BD1A663Y
SAKETH 22BD1A663V
AKSHITHA 22BD1A663R

FRONTEND: React.js for a dynamic and responsive user interface.

BACKEND: Node.js and Express for a robust and scalable server-side application.

DATABASE: MongoDB for flexible and efficient data storage.Authentication: Secure user authentication and authorization using JWT (JSON Web Tokens).

NOTIFICATIONS: Integration with push notification services for real-time reminders.

CONTENTS

- **Problem Statement**
- **Feature List**
- **Use cases/Work flow**
- **DB Schema**
- **UI Screens(mock up)**
- **Middleware**

*PROBLEM STATEMENT:

The Smart Habit Tracker is an innovative web application designed to help users build and maintain positive habits while breaking free from negative ones. Our mission is to empower individuals to take control of their daily routines, achieve their personal goals, and enhance their overall well-being through a user-friendly and data-driven platform.

***FEATURE LIST:**

1.*INTUITIVE INTERFACE*: A clean and simple design that makes tracking habits effortless and enjoyable.

2.CUSTOMIZABLE HABITS: Users can create and personalize their own habits, setting specific goals and timeframes.

3.PROGRESS VISUALIZATION: Graphs and charts that provide visual feedback on progress, helping users stay motivated.

4.REMINDERS AND NOTIFICATIONS: Timely reminders to keep users on track and ensure consistency.

5.COMMUNITY SUPPORT: Social features that allow users to share their progress, motivate friends, and participate in group challenges.

6.DATA ANALYTICS: Insightful analytics that highlight trends and patterns, offering a deeper understanding of one's habits.

7.GAMIFICATION: Reward systems and achievement badges to make habit tracking fun and engaging.

*USECASES/WORKFLOW:

USECASES:

Use Case 1:

User logs in and views today's habits.

Use Case 2:

User creates a new habit and sets reminders.

Workflow:

1. User logs in →
2. Dashboard shows today's habits →
3. User clicks 'Add Habit' →
4. User fills habit details →
5. User sets reminders →
6. Habit is saved and added to the list.

*DB SCHEMA:

Entities:

User

Habit

Activity

Relationships:

Each User can have multiple Habits.

Each Habit can have multiple Activities.

*UI SCREENS:

1.Login Screen:

->Username/Email input field.

->Password input field.

->Login button.

->Sign-up link.

2.Dashboard with Habit List:

->Header with user profile picture and username.

->List of habits with completion status and options to mark completion.

->Add habit button.

3.Habit Creation Form:

->Form fields for habit name, description, frequency, and start date.

->Save and cancel buttons.

4.Habit Detail View:

->Header with habit name and navigation back button.

->Habit details such as description, frequency, and start date.

->Calendar or timeline view to track habit completion.

->Mark completion button.

*MIDDLEWARE:

1.Designing the Middleware Architecture:

->Identify the main functionalities required for your habit tracker: user authentication, habit management, and notifications.

->Design the structure of middleware components to handle these functionalities in a modular and scalable manner.

2.Setting up the Development Environment:

->Choose a suitable programming language and framework for middleware development.

For example, Node.js with ->Express.js is commonly used for building web application backends.->Install necessary tools such as Node.js and npm.

3.Writing Code for Middleware Components.

4.Integrating Middleware with the Database and Frontend:

->Connect your middleware with the database by configuring database connection and models.

->Define routes and API endpoints for frontend communication, and integrate middleware functions accordingly.

THANK YOU