Contents

[**1. INTRODUCTION** 2](#_Toc185293570)

[**1.2 System Scope** 2](#_Toc185293571)

[**1.3 Goals of the Team** 2](#_Toc185293572)

[**1.4 Process Model** 3](#_Toc185293573)

[**2.2 Literature Survey and Technical Research** 4](#_Toc185293574)

[**3 DESCRIPTION** 5](#_Toc185293575)

[**3.1 Features of Habit Tracker Website** 6](#_Toc185293576)

[**3.2 Environment** 7](#_Toc185293577)

[**4. REQUIREMENTS** 8](#_Toc185293578)

[**4.1 Functional Requirements** 8](#_Toc185293579)

[**4.1.1 Menu Requirements** 8](#_Toc185293580)

[**4.1.2 Habit Tracking Functions** 8](#_Toc185293581)

[**4.2 Structural Requirements** 8](#_Toc185293582)

[**4.2.1 User Interface (UI)** 8](#_Toc185293583)

[**4.2.2 Notifications** 9](#_Toc185293584)

[**4.2.3 Database** 9](#_Toc185293585)

[**4.3 Performance Requirements** 9](#_Toc185293586)

[**4.4 Software Requirements** 10](#_Toc185293587)

[**4.5 Hardware Requirements** 10](#_Toc185293588)

### 

### **1. INTRODUCTION**

The first and essential step in the development of the Habit Tracker Website is the requirement analysis phase. This phase focuses on identifying and documenting the objectives, user needs, and system specifications that guide the design and implementation process.

This report provides a detailed definition of the problem statement and conducts an in-depth analysis of the functional and technical requirements. It also includes system models to represent the planned structure and operation of the website, along with management plans to ensure organized and efficient project execution.

**1.1 System Purpose**

The purpose of this project is to develop a Habit Tracker Website that enables users to build and maintain positive habits efficiently.

### **1.2 System Scope**

The scope of the Habit Tracker Website project is limited to the following:

* Providing a user-friendly platform for creating, tracking, and managing habits.
* Analyzing user requirements to ensure the system aligns with user needs.
* Designing an responsive interface for both desktop and mobile devices.
* Implementing habit tracking features, including progress monitoring and streak tracking.
* Supporting data visualization through charts, calendars, and analytics tools.
* Ensuring secure user authentication and data storage.

### 

### **1.3 Goals of the Team**

Our goals are:

* Learning how to manage a project.
* Using documentation efficiently.
* Using time efficiently.
* Learning teamwork.
* Producing a satisfactory software.

### **1.4 Process Model**

For the **Habit Tracker Website** project, we will use the **Agile Scrum** model. Scrum is an iterative approach where the project is divided into small, manageable parts called sprints. Each sprint focuses on delivering a specific feature, with regular feedback to improve the product.

**2. RESEARCH**

**2.1 Market Research**

The habit tracker market is growing rapidly, with various applications already available to help users build and maintain habits. Many popular apps offer basic features such as habit creation, reminders, and progress tracking. Our **Habit Tracker Website** aims to provide a similar solution for users. By studying successful habit tracker applications, we aim to understand the best practices and features that users value, while also identifying areas for innovation. This project will guide us in developing a user-friendly platform with features that cater to the needs of a wide audience.

In this project, we will review existing habit tracker applications, analyze their key features.

### **2.2 Literature Survey and Technical Research**

In this section, we will review existing studies and research to understand the best methods and tools for creating a habit tracker website. This will help us build a platform that is easy to use, effective, and can grow with more features in the future.

#### **Literature Survey**

Research shows that tracking habits can help people stay motivated and build good routines. Techniques like setting small goals, gamification (turning tasks into games), and habit stacking (combining habits together) are common strategies used in successful habit trackers. We plan to include similar features to keep users engaged and help them stick to their goals.

#### **Technical Research**

For the technical part of our project, we will use the following technologies:

* **Frontend:** React to build the user interface, and **Tailwind CSS** to make the website look good on all devices.
* **Backend:** A simple server using **Node.js** to handle user requests and manage data.
* **Database:** **MongoDB** to store user data and progress.
* **Authentication:** **Firebase Authentication** to make it easy and secure for users to sign up and log in.

This research helps us understand how to build a reliable habit tracker, ensuring the website is secure, fast, and easy to use. We aim to create a product that is simple for users while also being technically strong behind the scenes.

### **3 DESCRIPTION**

### **3.1 Features of Habit Tracker Website**

The **Habit Tracker Website** will include several key features to help users track and manage their habits. The properties and functionalities of these features are outlined below:

#### **3.1.1 Habit Creation**

* Users can create new habits with specific names and descriptions.
* Each habit will have a start date, frequency, and target goal.
* Reminders can be set to notify users to complete their habit.
* A visual progress tracker will show the percentage of completion over time.

#### **3.1.2 Habit Progress Tracking**

* Users can mark daily habit completions with a simple “Yes” or “No.”
* The website will display the total number of completed days, missed days, and streaks.
* A calendar view will show progress for the entire month.
* A detailed progress graph will display long-term trends for each habit.

#### **3.1.3 Social Features**

* Users can share their progress with friends or on social media.
* A leaderboard will display the top users based on consistency and streaks

#### **3.1.4 Habit Reminders**

* Users will receive daily notifications reminding them to complete their habits.
* The reminder time can be customized based on user preferences.
* A progress notification will also alert users when they reach a milestone or streak.

#### **3.1.5 Analytics and Reports**

* Users will receive weekly or monthly reports on their habit progress.
* Reports will include details on the total completion rate, frequency, and consistency of habits.
* Insights will be provided to help users understand their most successful habits and areas for improvement.

#### **3.1.6 User Profiles**

* Each user will have a profile where they can track all their habits and goals.
* The profile will display an overall success rate, streaks, and progress on each habit.
* Users can modify their profile settings, including username and email preferences.

### **3.2 Environment**

* The website will be designed to work on desktop and mobile devices.
* The responsive layout will adjust based on screen size for optimal viewing.
* Dark and light mode options will be available to users for a personalized experience.

# **4. REQUIREMENTS**

## **4.1 Functional Requirements**

### **4.1.1 Menu Requirements**

* The website should allow users to create, delete, and edit habits.
* Users can set frequency (daily, weekly, monthly) for each habit.
* A clear and simple navigation menu should allow users to view their habits, progress, and settings.

### **4.1.2 Habit Tracking Functions**

* The website should track each habit's progress and mark them as completed or missed.
* The system will provide a progress bar or checklist to indicate the completion of each habit.
* Users should be able to set reminders for each habit and receive notifications.

## **4.2 Structural Requirements**

### **4.2.1 User Interface (UI)**

* The design must be user-friendly and mobile responsive.
* The layout should display habits, progress, and reminders in a clear and concise manner.
* Color schemes and fonts must be consistent throughout the website.

### **4.2.2 Notifications**

* Users must receive timely reminders based on the habits they have set.
* Notifications must be customizable by time, frequency, and method (e.g., push notifications, email).

### **4.2.3 Database**

* The website will use a cloud-based database to store user data, including habits, progress, and reminders.
* Data must be secured and encrypted, with secure login and authentication for users.

## **4.3 Performance Requirements**

* The website must load within 2 seconds for smooth user experience.
* The website should be able to handle up to 100 concurrent users without significant performance degradation.

## **4.4 Software Requirements**

* The website will be developed using React for the frontend and Node.js for the backend.
* The website will be compatible with both iOS and Android devices.
* The system will integrate with cloud services (e.g., MongoDB) for data storage.

## **4.5 Hardware Requirements**

* The website should work on mobile devices (smartphones and tablets) with a minimum of 2GB RAM and 2GHz processing speed.
* The website should be compatible with the latest versions of Android and iOS.

## **5. System Analysis and Modeling**

**5.1 System Overview**

The Habit Tracker Website follows a client-server architecture where users interact with the frontend interface, and the backend processes requests, stores data, and manages authentication. The system ensures secure, scalable, and efficient habit tracking with real-time updates.

**5.2 Use Case Diagram**

A Use Case Diagram helps visualize user interactions with the system.

Actors:

* User: Registers, logs in, creates/manages habits, views progress, and sets reminders.
* System: Sends notifications, tracks progress, and generates reports.

Use Cases:

1. Register/Login – Users sign up and authenticate via Firebase.
2. Create Habit – Users add new habits with details like name, frequency, and goals.
3. Track Progress – Users mark habits as completed or missed.
4. Receive Notifications – System sends reminders for pending habits.
5. View Reports – Users analyze progress using charts and graphs.

**5.3 Data Flow Diagram (DFD)**

**5.3.1 Level 0 - Context Diagram**

At Level 0, the system is represented as a single process interacting with external entities.

Entities & Data Flow:

* User → Habit Tracker System: Inputs habit details, marks progress.
* System → Database: Stores and retrieves user habits, progress.
* System → User: Sends habit reminders, reports.

**Fig: Context Diagram (DFD)**

**5.3.2 Level 1 - Detailed DFD**

At Level 1, the system is broken into multiple processes that show the internal flow of data.

Processes in Habit Tracker System:

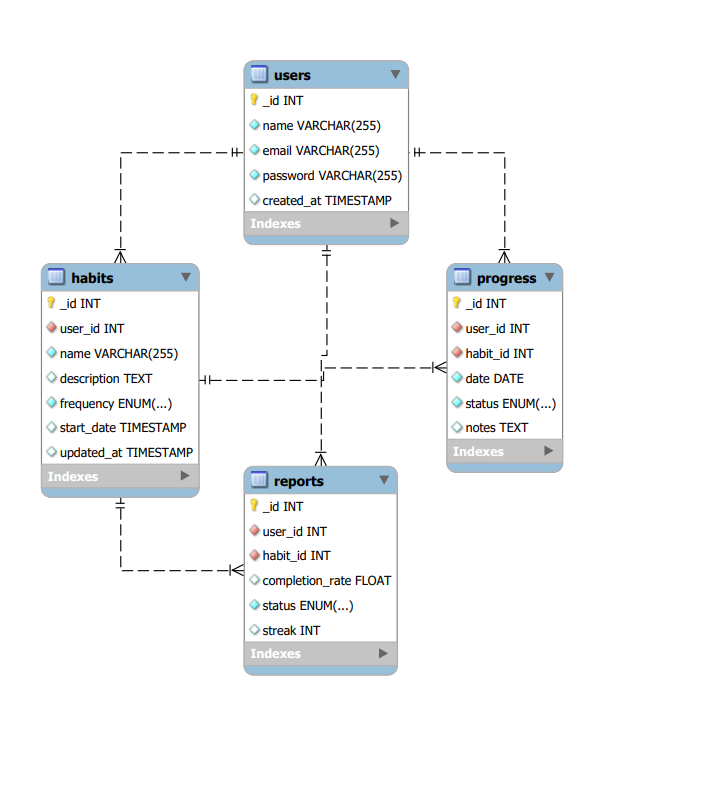
1. User Authentication (Login/Register)
   * User submits credentials → System verifies and grants access
2. Habit Management
   * User creates a habit → System saves habit in the database
   * User updates/deletes a habit → Database reflects changes
3. Habit Tracking
   * User marks habit as completed → System updates progress
4. Notifications
   * System checks scheduled reminders → Sends notifications
5. Reports & Analytics
   * System calculates statistics → Displays reports

**Fig: Detailed DFD**

**5.4 Entity-Relationship Diagram (ERD)**

An ERD represents how data is stored in MongoDB.

Entities & Relationships:

* User (UserID, Name, Email, Password, CreatedAt)
* Habit (HabitID, UserID, HabitName, Frequency, StartDate, Progress, Reminders)
* ****Progress (ProgressID, HabitID, Date, Status)

**Fig: Entity-Relationship Diagram (ERD)**

**5.5 State Diagram**

A State Diagram shows different states of a habit in the system.

Habit State Transitions:

1. Created → Pending → Completed → Archived
2. Missed habits → Notification sent → Reattempt or Drop

**Fig: State Diagram**

Final Thoughts

* This section makes your project documentation stronger by showing how the system operates.
* Adding visuals (Use Case, DFD, ERD, State Diagram) will improve clarity.