**A picture containing logo

Description generated with very high confidence**

**Mentor:**

Dr M N Thippeswamy, Head and Professor

Department Of CSE(AI&ML) and CSE(Cyber Security) Engineering

**Submitted By:**

Anu S M (UG 211)

Akshay S (UG 212)

Venkatesh R (UG 213)

Month 1: Report

**Initiation, Requirement Analysis, Tech Stack**

**Project Title: Making Your WhatsApp Conversations Smarter with Intelligent Reminders**

@IIT Roorkee

**iHUB DivyaSampark**

**1. Initiation**

**Problem Statement**

* The sheer volume of information exchanged on WhatsApp, Messages, and Email poses a unique challenge- Information overload.
* As users engage in diverse conversations, from personal discussions to professional collaborations, the critical details embedded in messages, such as event dates, locations, and times, can easily get lost amidst the digital noise.

**Problem Solution**

Introducing “TimeCraft(Making Whatsapp Conversations Smart with Intelligent Reminders)," a system that extracts dates, times, and locations from WhatsApp messages and automatically sets reminders 24 hours prior.

**Project Scope:**

* **Inclusions:**
  + Develop an app that allows users to log in with their phone numbers, extract event details from WhatsApp messages, store them in a database, and receive timely reminders.
* **Exclusions:**
  + Advanced features such as integrations with other messaging platforms could be considered in future iterations.

**Stakeholders**

Identified Stakeholders: Users (event attendees), Developers, and Administrators.

**2. Requirement Analysis**

**2.1 User Stories:**

Example User Stories:

* + As a user, I want to log in with my phone number to access the app.
  + As a user, I want to view upcoming events and their details.
  + As a user, I want to edit the details of an event.

**2.2 Functional Requirements:**

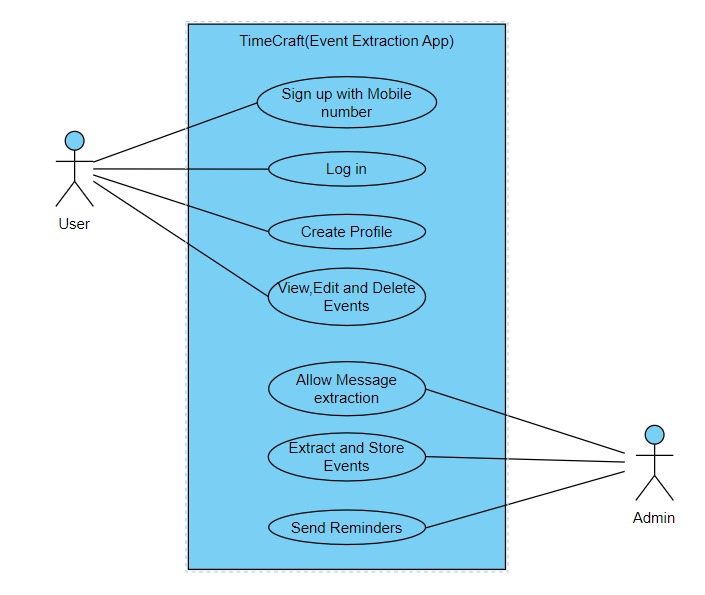
**Key Functionalities:**

* + User authentication via phone number.
  + Message extraction for event details (type, name, date, time).
  + Database storage of events.
  + Viewing and editing events.
  + Reminder notifications.

**2.3 Non-functional Requirements:**

* **Performance:**
  + The app should respond within a reasonable time frame.
* **Security:**
  + User data, including phone numbers and event details, must be securely stored and transmitted.
* **Usability:**
  + The user interface should be intuitive and user-friendly.

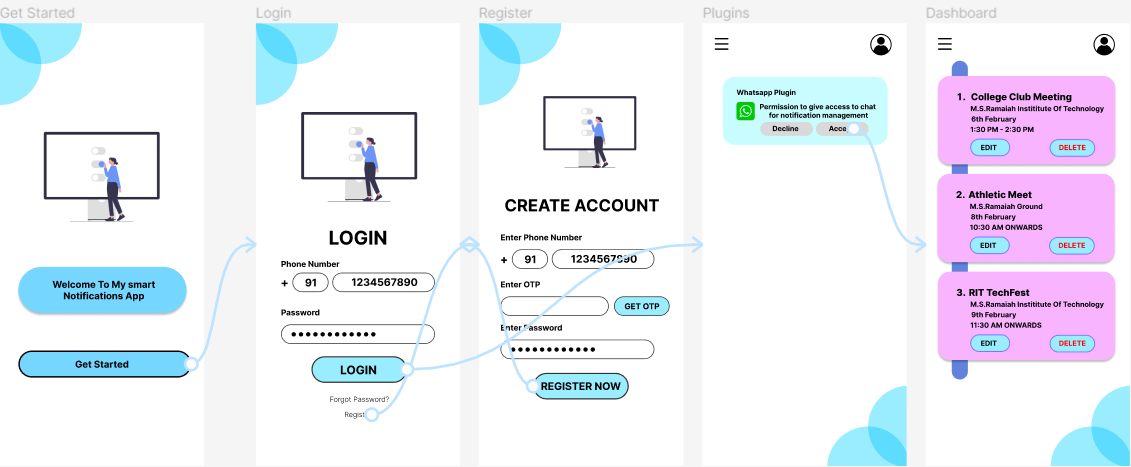
**2.4 Use Case Diagram**



**2.5 Data Requirements:**

* **Entities:**
  + User (phone number, profile information)
  + Event (type, name, date, time)
* **Database Structure:**
  + Users Table (user\_id, phone\_number, ...)
  + Events Table (event\_id, user\_id, type, name, date, time, ...)

**2.6 Sample User Interface of the app**

****

**3. Tech Stack Collection:**

**3.1 Development Frameworks:**

Flutter:

* + Cross-platform framework for mobile app development.

**3.2 Backend Services:**

Firebase:

* + Authentication: Firebase Authentication for phone number authentication.
  + Database: Cloud Firestore for storing user and event data.
  + Cloud Functions: For serverless backend logic.

**3.3 Natural Language Processing (NLP) Model:**

SpaCy (or other suitable NLP library)

**3.4 Notification Service:**

Firebase Cloud Messaging (FCM):

* + For sending reminder notifications.

**3.5 Authentication:**

Firebase Authentication:

* + Phone number authentication.

**3.6 IDE and Version Control:**

IDE:

* + Visual Studio Code.

Version Control:

* + Git for version control.

**3.7 Programming Language:**

Dart:

* + Programming language for Flutter development.