**Event Management App System Design Document**

# **Introduction:**

### **Purpose:**

The Purpose of the event planning and management app is to streamline and enhance the process of organizing, managing, and participating in events within the college community. The application aims to provide an eﬃcient and user-friendly platform for both event organizers and attendees, facilitating seamless communication, coordination, and execution of various events, ranging from small club meetings to

large-scale campus-wide functions.

### **Scope:**

Scope of the app encompasses the functionalities and features that will be included in the application. The scope of the event planning and management app includes, but is not limited to:

### **Deﬁnitions, Acronyms, and Abbreviations:**

* Event:
  + An organized activity, gathering, or occasion planned and executed by the clubs within the college.
  + Acronym/Abbreviation: EVNT
* Management:
  + The creation, adjustment, and optimization of event schedules to ensure eﬃcient utilization of resources and a seamless experience for participants and organization.
  + Acronym/Abbreviation: MNGT
* Clubs:
  + Student organizations formed around common interests, hobbies, or activities within the college community.
  + Acronym/Abbreviation: CLBS
* Students:
  + Individuals enrolled in the college who are potential participants, attended, or organizers of events.
  + Acronym/Abbreviation: STDNS
* Organizers:
  + Individuals or groups responsible for planning, coordinating, and executing events.
  + Acronym/Abbreviation: ORG

## **Reference:**

* IEEE std 1016-2009 SDD Document.

**Overview**

This document provides an overview of the system architecture, user interface design, data design, and deployment design for the Event Management App. It outlines the various screens and functionalities of the application, serving as a guide for developers, designers, and stakeholders involved in the project.

**System Overview:**

* System Context:
  + The Event Management App is a comprehensive platform designed to facilitate the organization and participation in events. It serves as a centralized hub for users to create, manage, and attend various types of events, enhancing efficiency and engagement within the event ecosystem.
* System Functionality:
  + Users can create new events by providing essential details such as titles, descriptions, and event categories.
  + Event organizers can manage their events, edit event details, and delete events as needed.
  + Users have access to an event dashboard that provides an overview of upcoming and past events, enabling efficient event management.
  + The app provides event analytics, offering insights into event performance and engagement metrics to improve planning.
  + Attendees can view comprehensive event details such as schedules, speakers, and venue information to plan their attendance effectively.
  + User profiles, passwords, and privacy settings can be managed through the account settings screen.
  + Notifications keep users informed about important updates and event-related information, with options to manage preferences.
  + Attendee lists for events are accessible, allowing users to search attendees and send messages as required.
  + Users can provide valuable feedback on events through ratings, comments, and feedback submission, aiding organizers in improving future events and enhancing attendee satisfaction.

### User Characteristics:

* + Intended Users: This application is mainly directed towards students and the leads of various clubs. For scalability purposes, this application will later be extended to the college administration.
  + Prerequisites for Users: Users must be students of the Indian Institute of Information Technology, Kottayam. They must be well-versed with using a mobile application. Other than this, the application is simple to use and does not have any more prerequisites.
  + User Interface Requirements: The student user interface as well as professional user interface will feature a login page. After logging into the application, the student will be directed to a customized homepage that features their preferred clubs and club aﬃliations. For club leads, the homepage will display past conducted events, events scheduled by other clubs, etc.
* Constraints:
  + The Event Management App's performance may vary based on the hardware capabilities of the user's device, affecting the speed and smoothness of interactions.
  + Network connectivity is required for users to access detailed event information, register for events, and save changes to event data in the Firebase database.

### Assumptions and Dependencies:

* + Device connectivity and compatibility: The application assumes that the intended users have a compatible device to run the app on and an internet connection.
  + User engagement: The application should be utilized by the students and club leads to facilitate event management, attendance tracking and other facilities.
  + College support: The application must be encouraged by the college administration.
  + Event venue availability: The availability and allocation of event venues must be accurately reﬂected in the app. Any changes should be promptly updated. The app depends on these timely updates.
  + Database Integration: The application depends on its integration with the college's existing database or authentication system for user veriﬁcation and access control.
  + User training and support: Adequate training and support mechanisms to ensure that users (students and club leads) are familiar with the app's functionalities and can troubleshoot common issues.

**Architectural Design:**

* Overview:
  + This section describes the overall architecture of the Event Management App, including hardware and software components, interfaces, and data flow between them.
* Hardware/Software Mapping:
  + Hardware: User devices such as smartphones, tablets, or computers.
  + Software: React Native for frontend development, MongoDB for backend database management.
* Data Architecture:
  + User-created events and related data are stored in MongoDB.
  + User information, profiles, and authentication details are also managed within MongoDB.

* Access Control and Security:
  + MongoDB Atlas provides secure user authentication and access control to ensure data security.

**User Interface Design:**

* Overview:
  + The user interface design focuses on simplicity and intuitive interaction for event creation, management, and attendance.
  + We will be using Figma for the creation of the user Interface Design.

* Interface Design:
  + UI elements are designed using Figma, following modern design principles and guidelines for mobile applications.

* Interface Features:
  + Welcome Screen: Options for login, signup, and password recovery.
  + Event Creation: Form for creating new events with titles, descriptions, and categories.
  + Event Details: Comprehensive view of event information with edit and delete options.
  + Event Registration: Registration for events after viewing details.
  + Event Dashboard: Overview of user's events, upcoming and past.
  + Event Analytics: Insights into event performance and engagement.
  + Account Settings: Profile management, password change, and privacy settings.
  + Notifications: Management of event-related notifications.
  + Event Search: Search for events based on criteria such as date, location, and keywords.
  + Event Details (Attendees): Detailed information for attendees, including schedules and speakers.
  + Attendee List: View of attendees for specific events with search and messaging options.
  + Event Feedback: Provide feedback on events with ratings and comments.

* Interface Identification and Description:
  + Main Menu: Central hub for accessing all app functionalities.
  + Event Cards: Display of events with essential details for quick viewing.
  + Edit Forms: User-friendly forms for editing event details.
  + Notification Centre: Inbox for managing event notifications.
  + Analytics Dashboard: Visual representation of event data for analysis.

**Data Design:**

* Overview:
  + This section describes the data design of the Event Management App, including the data used by the system, a data dictionary, and information about data storage, access, and manipulation.
* Data Description:
  + React Native frontend components: UI elements, navigation, and event management logic.
  + MongoDB backend components: Database management, user authentication, and event data storage.
* Data Storage:
  + Events and related data are stored in MongoDB, ensuring scalability and reliability.
  + User profiles and authentication information are securely stored within MongoDB, following best practices for data security.

**Component Design:**

* Component Description:
  + This section provides a detailed description of the components that make up the Event Management App. The system consists of the following components.
* User Interface Component:
  + Handles user interactions, navigation, and event creation/editing.

* Database Component:
  + Manages data storage, retrieval, and manipulation within MongoDB.

* Authentication Component
  + Provides user authentication and access control to app features.
* Component Functionality:
  + Each component performs specific tasks related to its functionality, ensuring modularity and scalability.

* Interface Description:
  + Components communicate through well-defined interfaces, adhering to established protocols for data exchange.
* Event Creation and Management:
  + Allows users to create, edit, and delete events with ease.

* Event Registration and Attendance:
  + Enables users to register for events and view their attendance status.

* Messaging and Notifications:
  + Facilitates communication between event organizers and attendees through messaging features and event notifications.
* Component Implementation:
  + React Native components are implemented using JavaScript and JSX for UI rendering and logic.
  + MongoDB Atlas is used for backend data storage and management, with Node.js for server-side logic.
* Software Infrastructure:
  + Database management system: MongoDB Atlas
  + Programming languages: JavaScript, JSX
  + Frontend: React Native, Figma for UI design

**Deployment Design:**

* Overview:
  + The deployment design outlines the process of distributing the Event Management App to end-users and managing server-side infrastructure.
* Security Infrastructure:
  + To ensure the security of the Event Management App, the following measures have been implemented:
  + User Authentication: Secure user authentication using MongoDB Atlas.
  + Data Encryption: Encryption of sensitive user data in transit and at rest.
  + Access Controls: Role-based access controls to limit user permissions.
  + Secure Communication: Use of HTTPS for secure data transmission between client and server.
* Deployment Strategy:
  + The app will be deployed to app stores for iOS and Android platforms.
  + MongoDB Atlas will host the backend database, ensuring scalability and reliability.
  + Continuous monitoring and updates will be performed to maintain app security and performance.