Software Requirements Specifications

For

KMIT CLUBS HUB

Version 1.0 approved

Prepared By-Team-549

KMIT

30-07-2025

# Table of Contents

[Table of Contents ii](#_Toc6145)

[Revision History ii](#_Toc6146)

[1. Introduction 1](#_Toc6147)

[1.1 Purpose 1](#_Toc6148)

[1.2 Document Conventions 1](#_Toc6149)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc6150)

[1.4 Product Scope 1](#_Toc6151)

[1.5 References 1](#_Toc6152)

[2. Overall Description2](#_Toc6153)

[2.1 Product Perspective 2](#_Toc6154)

[2.2 Product Functions 2](#_Toc6155)

[2.3 User Classes and Characteristics 2](#_Toc6156)

[2.4 Operating Environment 2](#_Toc6157)

[2.5 Design and Implementation Constraints 2](#_Toc6158)

[2.6 User Documentation 2](#_Toc6159)

[2.7 Assumptions and Dependencies 3](#_Toc6160)

[3. External Interface Requirements 3](#_Toc6161)

[3.1 User Interfaces 3](#_Toc6162)

[3.2 Hardware Interfaces 3](#_Toc6163)

[3.3 Software Interfaces 3](#_Toc6164)

[3.4 Communications Interfaces 3](#_Toc6165)

[4. System Features 4](#_Toc6166)

[4.1 System Feature 1 4](#_Toc6167)

[4.2 System Feature 2 (and so on) 4](#_Toc6168)

[5. Other Nonfunctional Requirements 4](#_Toc6169)

[5.1 Performance Requirements 4](#_Toc6170)

[5.2 Safety Requirements 5](#_Toc6171)

[5.3 Security Requirements 5](#_Toc6172)

[5.4 Software Quality Attributes 5](#_Toc6173)

[5.5 Business Rules 5](#_Toc6174)

[6. Other Requirements 5](#_Toc6175)

[Appendix A: Glossary 5](#_Toc6176)

[Appendix B: Analysis Models 5](#_Toc6177)

[Appendix C: To Be Determined List 6](#_Toc6178)

# 

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Change** | **Version** |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

**1. Introduction**

**1.1 Purpose**

KMIT CLUBS HUB is a web-based application designed to centralize and streamline the management of student clubs at KMIT. The system enables students to explore, join, and engage with clubs while providing faculty and administrators with tools for managing memberships, events, reports, and documentation.

**1.2 Document Conventions**

* Functional requirements are labeled as **REQ-XX**.
* Nonfunctional requirements are detailed in **Section 5**.

**1.3 Intended Audience and Reading Suggestions**

This document is intended for:

* Developers and engineers implementing the system
* Project managers and system architects
* UI/UX designers
* Faculty coordinators and student leaders
* System administrators

**1.4 Product Scope**

This product aims to unify the handling of club activities, member data, event planning, and approval processes. It provides a platform for:

* Club discovery and membership applications
* Event scheduling and approvals
* Documentation storage (minutes, reports, media)
* Real-time updates for audits and admin dashboards

**1.5 References**

* IEEE SRS Format Guidelines
* KMIT Accreditation and Club Management Policies

**2. Overall Description**

**2.1 Product Perspective**

KMIT CLUBS HUB is a standalone platform that may integrate with institutional systems in the future. It serves students, faculty, and admins with distinct roles and views.

**2.2 Product Functions**

* REQ-1: Student registration and login
* REQ-2: Explore and join clubs
* REQ-3: Club creation and member management
* REQ-4: Event planning and calendar view
* REQ-5: Document/photo upload for events and meetings
* REQ-6: Role-based dashboards for students, club heads, faculty, and admins
* REQ-7: Approval workflows for events and budgets
* REQ-8: Reporting and export for audits

**2.3 User Classes and Characteristics**

* **Students**: General users who can join clubs and register for events
* **Club Heads**: Manage club details, events, and members
* **Faculty Coordinators**: Oversee club operations, verify activities
* **Admins**: Approve events, monitor activity, and access reports

**2.4 Operating Environment**

* Platforms: Web-based (Chrome, Firefox), Android-friendly
* Backend: Node.js with Express
* Frontend: React/Vite
* Database: MongoDB
* Hosting: Vercel or Heroku

**2.5 Design and Implementation Constraints**

* Cloud storage integration (e.g., Firebase or Cloudinary)
* Secure login (JWT/Auth)
* Responsive design for mobile and desktop

**2.6 User Documentation**

* User manual
* On-screen tooltips/help
* Admin training video (optional)

**2.7 Assumptions and Dependencies**

* Reliable internet connection
* Active participation by clubs and faculty
* Administrative access for system oversight

**3 External Interface Requirements**

**3.1 User Interfaces**

* Clean and responsive web UI
* Dashboards tailored to user roles
* Calendar/event views
* Document upload interface

**3.2 Hardware Interfaces**

* Standard laptop or mobile device with browser support

**3.3 Software Interfaces**

* MongoDB for database operations
* Firebase/Cloudinary for media uploads
* RESTful API for communication

**3.4 Communications Interfaces**

* HTTPS for secure transmission
* Email notification system for approvals and alerts

**4. System Features**

**4.1 Club Registration and Discovery**

* REQ-1: Students can browse and apply to clubs
* REQ-2: Club profiles display description, photos, events

**4.2 Member and Role Management**

* REQ-3: Club leaders manage members and assign roles
* REQ-4: Faculty oversee and validate member changes

**4.3 Event Creation and Approvals**

* REQ-5: Events can be created with poster uploads
* REQ-6: Faculty/admins can approve or reject events

**4.4 Document Archiving**

* REQ-7: Upload meeting minutes, reports, photos
* REQ-8: Filter by club/date/year

**4.5 Admin and Audit Tools**

* REQ-9: Generate PDF reports for inspections
* REQ-10: Dashboard showing club activity summaries

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

* Should load pages within 2 seconds under average load
* Handle 100+ concurrent users

**5.2 Safety Requirements**

* Regular database backups
* Graceful error handling for failed uploads or submissions

**5.3 Security Requirements**

* Role-based access
* Encrypted data transmission
* Secure login using JWT

**5.4 Software Quality Attributes**

* **Usability:** Minimal training needed
* **Maintainability:** Modular and scalable codebase
* **Reliability:** 99% uptime target

**5.5 Business Rules**

* Only authorized faculty/admins can approve events
* Events must be submitted 3 days prior for approval
* Each club must submit one annual report

**6. Other Requirements**

* Potential integration with KMIT’s central login or ERP
* Exportable data formats (CSV/PDF) for audit purposes

**7. Appendices**

* **Appendix A: Glossary**
* **REQ**: Requirement
* **JWT**: JSON Web Token
* **ERP**: Enterprise Resource Planning
* **UI**: User Interface

**Appendix B: Analysis Models**

* Use case diagrams
* Class diagrams
* Activity flow for event approval

**Appendix C: To Be Determined List**

* Selection of hosting provider
* Design of analytics dashboard
* Notification (email/SMS) system provider