Software Requirements Specification

for

EduBridge

Version 1.0 approved

**Prepared by :**

**Kashish Tyagi - 2100290120094**

**Kaushiki Verma – 2100290120095**

**Dewansh Gupta -2100290120070**

**Department of Computer Science,**

**KIET Group of Institutions, Delhi-NCR**

**March 21, 2024**

**Table of Contents**

**Table of Contents………………………………………………………………………………... ii**

**Revision History…………………………………………………………………………………. ii**

**1. Introduction………………………………………………………………………………….. 3**

1.1 Purpose………………………………………………………………………………………………... 3

1.2 Document Conventions………………………………………………………………………………..3

1.3 Intended Audience…………………………………………………………………………………….3

1.4 Product Scope…………………………………………………………………………………………3

**2. Overall Description………………………………………………………………………….. 4**

2.1 Product Perspective……………………………………………………………………………………4

2.2 Product Functions……………………………………………………………………………………..4

2.3 User Classes and Characteristics………………………………………………………………………4

2.4 Operating Environment………………………………………………………………………………..4

2.5 Design and Implementation Constraints………………………………………………………………4

2.6 User Documentation…………………………………………………………………………………...4

2.7 Assumptions and Dependencies……………………………………………………………………….4

**3. External Interface Requirements…………………………………………………………...5**

3.1 Login & Registration………………………………………………………………………………….5

3.2 DashBoard…………………………………………………………………………………………….. 5

3.3 Activity Feed………………………………………………………………………………………….. 5

3.4 Collaboration Tools……………………………………………………………………………………5

3.2 Software Interfaces…………………………………………………………………………………………………… 5

3.3 Communication Interface……………………………………………………………………………...6

3.4 External Integration…………………………………………………………………………………… 6

**4. System Features……………………………………………………………………………...6**

**5. Other Nonfunctional Requirements………………………………………………………... 7**

5.1 Performance Requirements…………………………………………………………………………… 7

5.2 Security Requirements……………………………………………………………………………...… 7

5.3 Usability Requirements……………………………………………………………………………….. 7

5.4 Scalability & Maintainability ………………………………………………………………………….7

**Appendix A: Glossary ……………………………………………………………………………8**

**ER Diagram………………………………………………………………………………………9**

**Appendix B: Analysis Models …………………………………………………………………...9**

**Revision History**

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

# 1. Introduction

In the rapidly evolving landscape of education, the dispersed nature of academic activities poses a significant challenge. "Edubridge" emerges as a potential solution, aiming to streamline access to vital information about educational engagements. This proposal outlines a centralized platform designed to address the existing fragmentation, promoting collaboration and efficiency within the academic community.

## Purpose

The purpose of this document is to present a comprehensive understanding of the Edubridge project, elucidating its objectives, methodologies, and the technology stack employed. By delineating the problem statement and proposed solution, this document serves as a guide for stakeholders involved in the development and implementation of Edubridge.

## Document Conventions

The document was created based on the IEEE Template for system requirement specification document.

## Intended Audience and Reading Suggestions

* + - Universities
    - Students
    - Mentors
    - Faculties
    - College Societies

## Product Scope

Edubridge addresses the fragmentation in educational activities by offering a unified platform. The scope encompasses students, mentors, and educational institutions, aiming to revolutionize the academic engagement experience. This document outlines the specific objectives and features that Edubridge will incorporate, providing a clear roadmap for the project's development and deployment..

# Overall Description

## Product Perspective

Edubridge exists as a standalone platform, designed to bridge the gap in the current educational landscape. It interfaces with external systems such as Firebase for backend services and utilizes Android Studio for app development. The platform, while self-contained, can integrate with educational institutions' systems for a seamless user experience.

## Product Functions

* **Comprehensive Database:** Edubridge will host a centralized database encompassing academic events, seminars, competitions, projects, and announcements.
* **User-Friendly Interface:** The platform will provide an intuitive and user-friendly interface for easy navigation and interaction.
* **Personalized Recommendations:** Edubridge will employ algorithms to offer personalized recommendations based on user preferences and engagement history.
* **Collaboration Tools:** The platform will incorporate collaboration features, facilitating communication and interaction between students and mentors.

## User Classes and Characteristics

* **Students:** Primary users, seeking information on academic activities, events, and opportunities.
* **Mentors:** Individuals guiding and supporting students, utilizing the platform to disseminate information and engage with their mentees.
* **Educational Institutions**: Entities benefiting from streamlined management of academic activities and enhanced student engagement.

.

## Operating Environment

Edubridge operates on the Android platform, utilizing Java and XML for programming and Firebase for backend services. It is compatible with devices running Android OS, ensuring accessibility for a broad user base..

## Design and Implementation Constraints

* **Technology Stack:** The platform is constrained by the chosen tech stack, including Java, XML, Firebase, and Android Studio.
* **Device Compatibility:** Edubridge is designed for Android devices, limiting its accessibility on other operating systems.
* **Security:** The web app should be designed with robust security features to protect against potential security threats, including unauthorized access, data breaches, and other vulnerabilities.

* **Scalability:** The app should be designed to handle a potentially large volume of data and user traffic, and should be scalable to accommodate future growth and changes in demand.

* **User experience:** The web app should be designed with a user-friendly interface that allows users to easily navigate and interact with the app. The design should be consistent throughout the app and should follow best practices for usability and accessibility.

* **Performance:** The app should be designed for optimal performance, with efficient and optimized code that can handle multiple requests and queries simultaneously without slowing down or crashing. The app should also be optimized for quick load times and smooth user experience.

* **Integration:** The app should be designed to integrate with other systems and services, such as authentication systems, payment gateways, and shipping services.

* **Compliance:** The app should comply with relevant laws and regulations, such as data privacy laws, consumer protection laws, and intellectual property laws. As well as the rules and regulations of the Institutions. The app should also comply with industry standards and best practices for security, performance, and user experience.

## User Documentation

* **Overview:** Provide an overview of the app and its purpose, explaining how it will help to prevent the sale of counterfeit clothes.

* **System requirements:** List the system requirements needed to access and use the app, including compatible devices and operating systems.

* **User classes:** Explain the different user classes and their roles in the system, including administrators, companies, and customers.

* **Features:** Users can explore a comprehensive database of academic activities, receive personalized recommendations, and utilize collaboration tools. The app promotes student engagement, fosters community building, and supports personal growth through active involvement in clubs and organizations.

* **Troubleshooting:** The user documentation includes troubleshooting tips for common issues, ensuring users can address challenges related to navigation, access, or collaboration effectively.

* **Security:** Edubridge prioritizes user data protection and employs Firebase for secure backend services. Details on data protection measures and the prevention of fraudulent activities are outlined in the documentation.

* **Technical information:** Users can find technical details about Edubridge, including the programming languages (Java, XML), development environment (Android Studio), backend services (Firebase), and database (Firebase Realtime DB). This section aims to provide insights into the technology stack used..

* **Contact information:** For technical support or customer service, users can find contact information, including email addresses and phone numbers, ensuring prompt assistance during operational hours.

## Assumptions and Dependencies

**Assumptions:**

* Edubridge assumes active collaboration with educators for content curation, relying on their expertise to identify and update key learning materials.
* The platform assumes iterative user feedback for continuous improvement, expecting users to actively participate in shaping the platform's features and functionality.
* Edubridge assumes the availability and functionality of Firebase services for backend operations and push notifications.

**Dependencies:**

* Edubridge depends on users having access to Android devices running the Android OS, limiting its compatibility to this platform.
* The platform's effectiveness depends on the accuracy and completeness of the curated content, relying on educators to ensure relevant and up-to-date information.
* Edubridge's success is contingent on the willingness of users to actively engage with the platform, contributing to a vibrant and collaborative educational community.

# External Interface Requirements

**3.1 Login and Registration:**

* Users access Edubridge through a secure login process. Registration includes basic information, ensuring a personalized experience.
* The interface provides a seamless experience for first-time users, guiding them through the registration process.

**3.2 Dashboard:**

* Upon login, users are presented with an intuitive dashboard displaying personalized recommendations, upcoming events, and recent announcements.
* The dashboard serves as a central hub for navigating various features and activities.

**3.3 Activity Feed**:

* Users can explore a dynamic activity feed showcasing current and upcoming academic events, projects, and announcements.
* The interface allows for easy interaction, with options to like, comment, and share activities.

**3.4 Collaboration Tools:**

* Dedicated spaces for students and mentors to collaborate, featuring discussion forums, project spaces, and communication channels.
* The interface fosters seamless interaction and engagement within the academic community.

**3.5 Software Interfaces**

**3.5.1 Android OS:**

* Edubridge is designed to operate on devices running the Android operating system (OS).
* Compatibility ensures a wide user base, facilitating access for students, mentors, and educational institutions.

**3.5.2 Java and XML:**

* The platform utilizes Java and XML for programming, ensuring robust functionality and a user-friendly interface.
* These programming languages contribute to the overall performance and responsiveness of Edubridge.

**3.5.3 Firebase Backend:**

* Edubridge integrates with Firebase for backend services, including real-time database functionality and cloud messaging for push notifications.
* Seamless integration with Firebase ensures reliable and secure data management and communication.

**3.6 Communication Interfaces**

**3.6.1 Push Notifications:**

* Edubridge employs Firebase Cloud Messaging for push notifications, alerting users about upcoming events, announcements, and personalized recommendations.
* The communication interface enhances user engagement and ensures timely updates.

**3.6.2 In-App Messaging:**

* The platform features in-app messaging for real-time communication between users, fostering collaboration and interaction.
* Users can communicate within dedicated spaces for projects, events, and mentor-student interactions.

**3.7 External Integration (Future Consideration):**

* While currently self-contained, Edubridge remains adaptable for potential integration with external educational systems, expanding communication possibilities.
* The platform is designed with flexibility to accommodate future communication interfaces, aligning with evolving technological landscapes.

# Other Nonfunctional Requirements

## 4.1 Performance Requirements

* Edubridge aims for near-instantaneous response times for user interactions, ensuring a seamless and engaging user experience.
* The platform should load activities, recommendations, and collaboration tools promptly to enhance user satisfaction.
* The system should handle an increasing number of users, activities, and interactions without compromising performance.
* Edubridge aims for scalability to accommodate growth and increased usage over time.

## 4.2 Security Requirements

* Edubridge employs secure login mechanisms, protecting user accounts from unauthorized access.
* User authentication ensures the confidentiality and integrity of personal information
* All communication between the app and the backend (Firebase) should be encrypted to safeguard sensitive information.
* Data encryption ensures that user data is secure during transmission and storage.

## 4.3 Usability Requirements

* Edubridge's user interface should be intuitive, allowing users to navigate the platform effortlessly.
* The design aims to minimize the learning curve for new users, ensuring accessibility for individuals with varying levels of technical expertise.
* The platform should be accessible to users with disabilities, adhering to accessibility standards.
* Edubridge aims to provide an inclusive user experience, accommodating diverse user needs.

## 4.4 Scalability & Maintainability

* Edubridge should support seamless updates and upgrades to introduce new features and improvements.
* The platform aims to evolve over time, adapting to changing educational landscapes and user requirements.
* Comprehensive documentation should be maintained for system administrators and developers, facilitating system maintenance and troubleshooting.
* Documentation ensures ease of system management and enhances the platform's overall maintainability

**Appendix A: Glossary**

**EduBridge: Our Software’s name**

**AWS: Amazon Web Services**

**IEEE Template: SRS Document creation template**

**SRS: Software Requirement Specification**

**UI: User interface**

**UX: User Experience**

**Firebase: Backend service for realtime database functionality**

**Android Studio: App development Enviornment**

**Level 0 DFD: Data flow diagram of level 0**

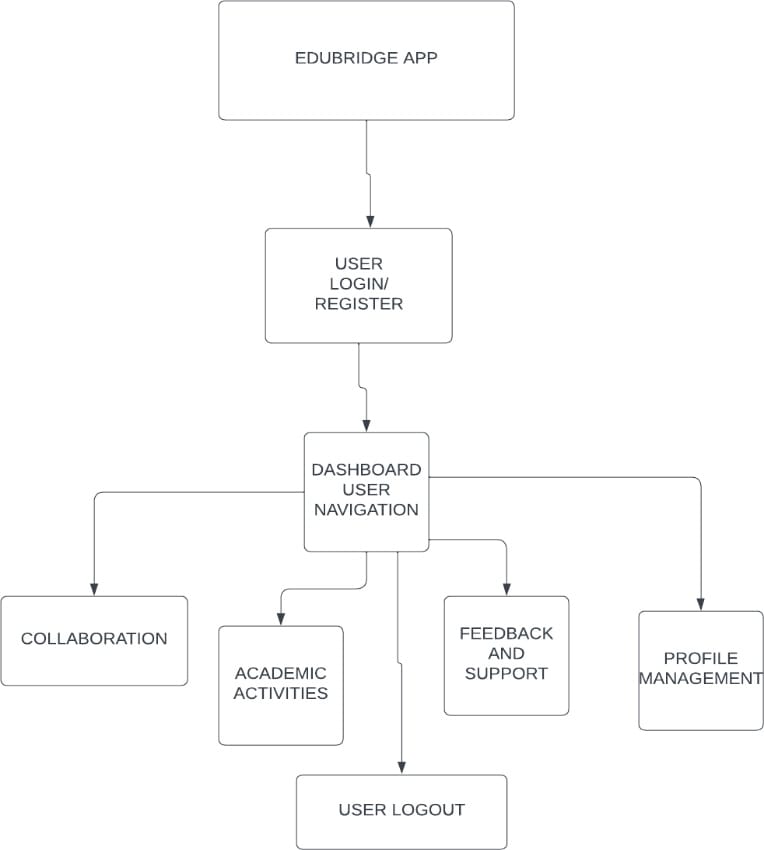
**Level 1 DFD: Data flow diagram of level 1**

**Level 2 DFD: Data flow diagram of level 2**

**ER Diagram: Entity Relationship diagram**

**Appendix B: Analysis Models**

Flow Diagram



ER Diagram

