SOFTWARE REQUIREMENTS SPECIFICATION

for

IITI-HUB

By

Ashutosh Sammantaray 210001005
Madhav Kadam 210001027
Mihirkumar Patel 210001050
Sairaj Loke 210001035
Samip Shah 210001061

Organization
ASM Developers

May 9, 2023

Contents

1 Preface

Name	Date	Reason For Changes	Version
Sairaj	15 March	getting requirements approved from customer	v1.0
Madhav	23 March	different levels of access	v1.1
Mihir	30 March	changing db services	v1.2
Ashutosh	2 April	login services and licensing	v2.0
Samip	10 April	cross-platform limits and scope	v2.2

This software requirement specification (SRS) is made for the customer and the developers of the organization to review High-Level features of the project In return for outstanding grades, which is contractually obligated between IIT-Indore and ASM-Developers. This software requirements specification (SRS) is made for the customer and the developers of ASM Developers to review the high-level features of the project. The purpose of this document is to provide a comprehensive overview of the IITI-HUB app that is being developed by our team. This SRS document will be used as a reference throughout the software development process to ensure that the final product meets the customer's requirements and expectations.

The document outlines the project's scope, including the app's intended audience and its features. It also details the external interface requirements and non-functional requirements that need to be considered during the development process.

The SRS document is intended for the customer, who will be reviewing the app's final version, and the development team, who will be responsible for building the app according to the requirements outlined in this document. Any changes to this document will be recorded in the "Reason For Changes" column of the table, along with the corresponding version number.

We hope that this SRS document will provide a clear understanding of the IITI-HUB app's purpose and scope and will serve as a useful reference throughout the development process. IITI-HUB is a customizable app that is being developed for the college community. The app will be used by students, professors, and administrators of the college to provide information and various services to the users. The SRS document will be used as a reference throughout the software development process to ensure that the final product meets the customer's requirements and expectations. The document outlines the project's scope, external interface requirements, non-functional requirements, and other requirements that need to be considered during the development process. The SRS document is intended for the customer and the development team responsible for building the app according to the requirements outlined in this document

2 Introduction

2.1 Purpose

The purpose of the project is to make an app(customizable for each institute) for the college community. It can be used by college students and professors and even by administrations. This app will provide information and regular updates about any technical, cultural, academic and sports events, a platform for student-professor collaboration, and many more things. Thus acting as a bridge between the members of the institute community.

2.2 Intended Audience and Reading Suggestions

This SRS is intended for developers (our team), and the higher authority who is going to review our app. This SRS consists of Project Description, Project scope, User Interfaces, Software interfaces, safety and security requirements, User class and characteristics, environment description, design and implementation constraints, and other small requirements and information.

2.3 Project Scope

The project aims to:

- Build an app/platform for bridging the College community with respect to updates/news
- Provide a solid Platform for Professor-Student collaborations.
- All the above features will be customizable by the concerned Institute using our app.
- Achieving actual implementation of our app in IIT Indore.

3 Overall Description

3.1 Product Perspective

Currently, there is no application software that aggregates the diverse needs of the IIT Indore fraternity. Primarily, we are focused on providing a platform for research collaboration with a secondary focus on event scheduling-level unit features.

3.2 Product Functions

- Communication medium for the organization
- Moderators and authorities will customization options which they can base upon user feedback
- Giving Member, Alumni, Guest mode options for using the app
- Aggregator platform

3.3 User Classes and Characteristics

The product will be used by students, professors, and administration staff. The students can only view project initiatives taken by professors. Professors can edit or delete their project posts once they have met their requirements. Administrators can also add news, make others admin, and add sections for mails.

3.3.1 User Privileges

There will be two modes:
1) Guest mode (for users who don't want to create a profile)
2) User mode In this user will use his Google account to sign in to the app and then there are three possibilities. One can have admin, professor or student privileges. Admin is allowed to add news and entities. Professor is entitled to manage the projects. Student can view all these things with filters and tags enabled.

3.4 Development and Operating Environment

The development will be done using Flutter as a framework and using firebase and firestore for database management, google_Oauth2.0 for authorization and gmail API for mail streamlining. This app will be a cross-platform app that can work on both Android and iOS mobile phones.

3.5 Design and Implementation Constraints

The most challenging part of our app is to make its design in such a way that it can be used by any college and can be made according to their need. Along with that, as it is mostly idealized from a business perspective, so a heavy database with all requirements and necessary security is hard to get. Also, since we are very new to mobile-application development so learning flutter and implementing it in a limited amount of time is much more difficult work.

Apart from that we will also need apache license and verification from google to be deploy a trusted app in the app store in IOS or play store in android.

3.6 Software and Hardware Requirements

Platform: Android and iOS
Architecture: x86_64 or arm
Memory: min 100 MB free storage
Android Version: Above Android 10
iOS version: above iOS 11
Recommended iOS version: iOS 16(latest)
Recommended Android Version: Android 13(latest)
Other Requirements: Working Google account for User mode usage

3.7 User Documentation

We will provide basic directions for using our software in the GitHub repository in the readme markdown file. Along with we may create a visually appealing user manual showing basic functionalities like how to sign up, etc.

3.8 Assumptions and Dependencies

The use cases for our software are mostly focused on businesses and organizations that require team effort and collaboration in some parts of the process. We have made this project keeping in mind the IITI community primarily. However, an effort has been put into making it more general purpose, catering to a wider variety of organizations and institutions.

It has been also taken into account that for general purposes, scalability issues might be faced, for which we have accounted for horizontal scaling.

4 External Interface Requirements

4.1 User Interfaces

We can describe the functioning of the app as, initially, users/authorities will choose how they want their app to function. Our app will have various optional features that they can select and incorporate into their app. after that; they can make their community by adding students, professors, and other authorities.

Our app will consist of different sections for project collaborations with professors, event scheduling for clubs, regular events from different clubs and their upcoming events, etc. Not only that, clubs can send their regular updates through our apps which can be given through the email section. they can now draft mail in our app only.

To enhance user experience, we can add a dark mode theme. Along with that, it will be so user-friendly that it can be used by every age group people so that the whole college community can have access to it

4.2 Software Interfaces

We will primarily provide two major services. The first is email streamlining and the other is research collaboration. For integration with Gmail services, we will be requiring google authentication services. The second major feature is a collaboration forum for organizations. The end-user platform will be hosted cross-platform on android and IOS. The server side will be hosted with a Firestore database on Firebase and use gmail API, for which we will need their cloud services.

5 Nonfunctional Requirements

5.1 Performance Requirements

The services provided are mainly real-time in nature, and as such, performance is a priority for optimal user experiences. Preventing lag in communications and event declaration on the interface. For this, we have implemented lazy-loading strategies wherever possible. The other major requirement is space optimization because of the large-scale storage of images, documents, and other media.

5.2 Safety and Security Requirements

We will ensure that the data of students, as well as of all communities of the college, remains safe and secure and provide proper backup to it. In addition to that, we will make sure that only people from the college community will be able to access materials and references provided on the app. This can be achieved through proper verification through institute passwords and user mail.

5.3 Software Quality Attributes

A major required attribute is flexibility because user feedback is important to any real-time operating software where fixing errors and kerfuffles are top priorities. Apart from that, security is also vital from outsider attacks which might cause leakage of sensitive data, for example, plagiarism of research proposals. We will be taking an incremental approach to the development, for which we have a focus on reusability. Ideally, we want to be able to cater to more general scenarios with customization options hence a need for portability and adaptability.