Software Requirement Specification(SRS)

1. Introduction:

1.1 Purpose of this Document:

- Finding a project in an interested domain to work on, has become a tedious process for students and project seekers.

- Building a capable and innovative team for an ideated project is also a mind-numbing task for researchers, faculties and companies.

- FinderX is a minimal, user-friendly solution that bridges the gap between these two parties by offering them a platform for seeking and communicating with each other

1.2 Scope of this document

To bring the process of making a profile, building a resume, applying for a job/project, to your fingertips while also making it unchallenging and effortless

- To help companies and project leaders with trouble-free single swipe accepting/offering mechanism, allowing them to choose more team members efficiently

- To promote small scale projects by leveling them with large scale ones on the same platform, making it easier for them to build a team

To promote prospective employees and project seekers by giving them access to top projects and the opportunity to join them.

1.3 Overview

- User accounts: Students and teachers will need to create accounts in order to use the app. Make sure to include fields for relevant information, such as name, email address, and school affiliation.

- Assignment management: Teachers should be able to view, grade, and provide feedback on student projects. This feature should allow teachers to access submitted project files, grade them on a rubric or scale, and provide comments.

- Communication tools: The app should provide communication tools that allow students and teachers to collaborate on projects. These tools could include a messaging system, discussion forums, or video conferencing capabilities.

- Notifications: The app should send notifications to students and teachers when new assignments are posted, deadlines are approaching, or grades have been posted.

Overall, the goal of the app should be to streamline the project management process for students and teachers, making it easier to collaborate, communicate, and track progress. By including features that address the needs of both students and teachers, you can create an app that is valuable to both groups and improves the overall project experience.

2. General description:

The software application is designed to help students and teachers collaborate on academic projects. The app typically provides tools for project creation, assignment management, communication, notifications, and reporting.

Students can use the app to create and submit projects to their teachers. The app allows them to upload project files, add descriptions, and set deadlines. Teachers can then access the submitted projects, grade them on a rubric or scale, and provide comments.

The app also provides communication tools that allow students and teachers to collaborate on projects. These tools may include messaging systems, discussion forums, or video conferencing capabilities. Notifications are sent to students and teachers when new assignments are posted, deadlines are approaching, or grades have been posted.

Reporting and analytics features are also provided, allowing teachers to track student progress and identify areas where students may be struggling. The app may provide charts and graphs that show student performance over time, as well as tools for identifying common errors or areas for improvement.

Overall, a student-teacher project app aims to streamline the project management process for students and teachers, making it easier to collaborate, communicate, and track progress. By including features that address the needs of both students and teachers, the app can improve the overall project experience and help students achieve academic success.

3. Functional Requirements:

1. Role Selection Page

The user can select their role, i.e. Teacher or Student, in turn representing their purpose of using the application. This allows them to have two separate individual interfaces which are appropriate for each of them. This page is the first one that pops up once the app has been downloaded.

2. Sign Up/Sign In Page

The SignIn page allows a registered user to login to the app with the registered email and verified password. The Sign-Up page helps a user to create an account to login to the app. The new user has to use a valid email id to register as a verification email is sent to the user's mail.

3. Create Profile/Create Project Page

This Page allows the user to create a profile for themselves(if they have selected Student as their role),or the project(if their role is a teacher). The Student Profile is similar to a brief resume that takes inputs allowing the user to describe themselves and their personal project interests. The Create Profile Page allows the teacher to enter the details of the project that he/she is interested to work on.

4. Home Page

The Home Page for the STUDENT ROLE includes a list of Teachers that have been uploaded on the database, a filter option to filter by search, and access to a navbar which contains buttons to open notifications, wishlist, profile page. This page also allows the user to access the Messages Section and also sign out. The Home Page for the TEACHER ROLE contains a similar outline or layout, but instead of a list of Teachers, the teacher will be able to view a list of Students. Each role gets to expand on one particular project/student by clicking on the button to access the detailed view.

5. Profile Page

The profile page displays the details of the project or the user, in case of teacher and student role respectively- Profile Picture, Background Picture, a short description, Phone Number, Email ID, Skills entered during creation, Project Interests, Branch for the student and similarly these details along with the Project Details like Project Status, Project Description for the teacher

6. Messages Page

This page contains the list of people in contact with, or groups that have been created for chat purposes.

7. Chat room Page

This page resembles the basic chat-room features- Input field to type the message out, a Send button that allows the user to send the typed message, a text bubble to represent messages sent by the user and other text bubbles to represent messages sent by other users.

8. Notifications Page

The notifications page displays the list of notifications- when someone clicks on connect on our profile/project.

9. Wishlist Page

This page displays all the students that have been wishlisted(if the user role is TEACHER) and the list of all the projects that have been wishlisted(if the role is STUDENT).

10. Expanded Profile Page

When the user clicks on the button on their homepage, this page pops up which contains a brief description of the Student/Project the user chose to expand. This page contains “connect” and “message” buttons that allow the user to do the respective.

5. Performance Requirements:

- Responsiveness: The app should respond quickly and smoothly to user interactions, such as clicking on a button or scrolling through a page.

- Load Time: The app should load quickly, allowing users to access the content and features without any significant delay.

- Scalability: The app should be designed to scale as the user base grows. It should be able to handle an increasing number of users and data without any significant drop in performance.
- Security: The app should be designed to protect user data and prevent unauthorized access. It should have security measures in place to ensure the confidentiality, integrity, and availability of data.

- Compatibility: The app should be compatible with different devices, browsers, and operating systems. It should be optimized for performance on different platforms.

- Speed: The app should be designed to respond quickly to user inputs and load content and features without any significant delay.

6. Design Constraints:

- Data Security Constraints: The project application should have data security constraints in place to protect sensitive information about students and teachers. It should comply with data privacy laws and regulations.

- Communication Constraints: The application should allow seamless communication between students and teachers. It should have the capability to send messages, share files, and receive feedback.

- Technical Constraints: The application should be compatible with different devices, browsers, and operating systems to ensure that it can be accessed by all users. It should also be optimized for performance and speed.

- Accessibility Constraints: The application should be designed to meet accessibility standards to ensure that it can be used by students and teachers with disabilities.

- Curriculum Constraints: The application should align with the curriculum and learning objectives to ensure that it is relevant and useful for students and teachers.

- Language Constraints: The application should be designed to support multiple languages to cater to a diverse user base.

7. Non-functional attributes:

Non-functional attributes, also known as quality attributes, are the characteristics of a software application that describe how well it performs certain functions or meets certain requirements. Here are some non-functional attributes that are important for a student-teacher project app:

Usability: The app should be easy to use and intuitive for both students and teachers. It should have a user-friendly interface, with clear instructions and easy-to-find features.

Performance: The app should perform well, with fast loading times and minimal lag or delay. It should also be able to handle a large number of users and projects without crashing or slowing down.

Security: The app should be secure, with features such as encryption, password protection, and user authentication. It should also comply with relevant data privacy laws and regulations.

Reliability: The app should be reliable and available to users at all times. It should have backup and recovery mechanisms in case of system failures or data loss.

Scalability: The app should be able to scale up or down to accommodate changes in user demand. It should also be able to integrate with other software systems and platforms.

Maintainability: The app should be easy to maintain and update, with clear documentation and modular design. It should also be flexible enough to accommodate changes in user requirements and feedback.

Accessibility: The app should be accessible to users with different abilities, such as those who use screen readers or other assistive technologies.

Overall, these non-functional attributes are important for ensuring that the student-teacher project app is reliable, secure, user-friendly, and accessible to all users.

8. Preliminary,Schedule and Budget:

- Complete within 2 months of period from 01/2/2023 to 01/4/2023
- Budget of $500 for server and database purchasing.