

## Table of Contents

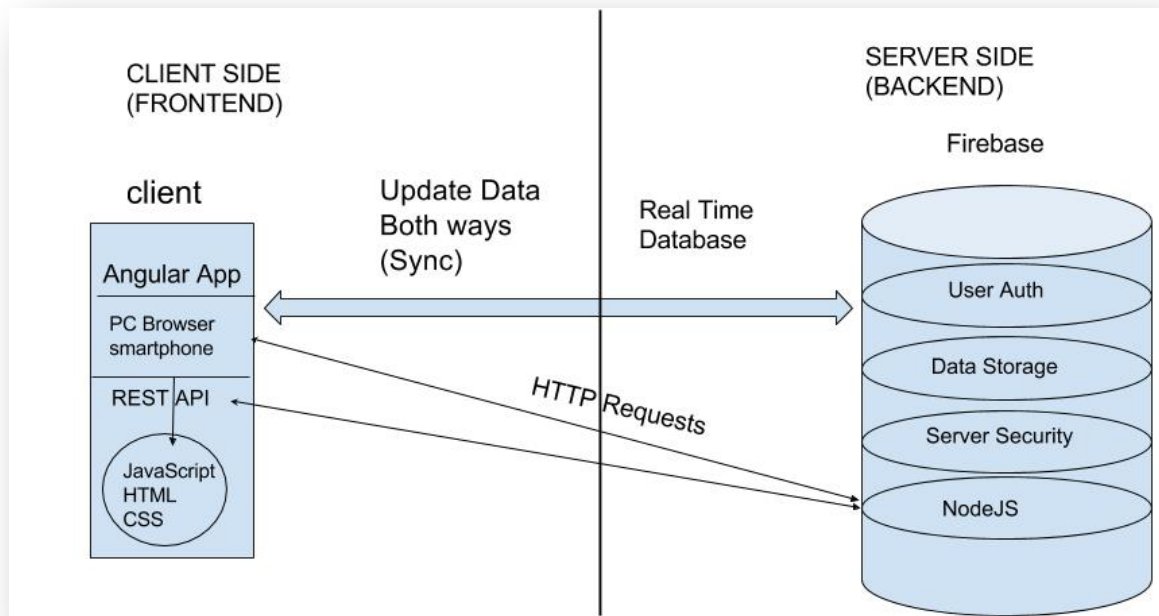
<b>5 Requirements Specification Document .....</b>	<b>2</b>
5.1 Introduction.....	2
5.2 Functional Requirements .....	3
5.2.1 User Registration .....	3
5.2.2 Tutor Profile .....	4
5.2.3 The Dashboard .....	5
5.2.4 The Main Page.....	5
5.2.5 User Types and Roles .....	6
5.3 Performance Requirements.....	9
5.3.1 Authentication Requests Response Time .....	9
5.3.2 Notifications Requests Response Time.....	10
5.4 Environmental Requirements .....	10
5.4.1 Development Environment Requirements .....	10
5.4.2 Execution Environment Requirements .....	11

## 5 Requirements Specification Document

### 5.1 Introduction

The School-Tutoring web application will provide an elevated environment for the user experience of both students and tutors to collaborate with each other in the context of schoolwork help. The application will allow students and tutors to register for different types of accounts. The student account will allow the user to post new jobs for tutor's help. The jobs will be listed under schools' major categories, such as Math, CMSI, Economy and more. Then the student will set a due date, upload documents, and set a price for the job. The tutor account on the other hand will allow the tutor to promote him or herself by listing their experience and credibility. Then, the tutor will be able to see a list of jobs posted by students. The tutor will be able to review them and propose to do the job for the student. The student will have the ability to accept, decline or counter the tutor's proposal. During this process students and tutors can communicate through live chat, and private messages to reach agreements. When the job gets assigned the tutor will work on it, and after finishing the job the tutor will submit it back to the student. The student can check the added notes or solutions, and then the payment is released. The student after releasing the payment will be able to post a review and rate the overall work of the tutor, which will be visible to all users.

The following is a high-level UML diagram.



UML System High Level Diagram

The remainder of this document is structured as follows. Section 5.2 contains functional requirements, with five subsections. Subsection 5.2.1 contains user registration requirements. Subsection 5.2.2 contains tutor profile requirements. Subsection 5.2.3 contains the dashboard requirements. Subsection 5.2.4 contains the main page requirements. Subsection 5.2.5 contains user types and roles requirements. Section 5.3 contains operational requirements with two subsections. Subsection 5.3.1 contains authentication response time requirements. Subsection 5.3.2 contains notifications response time requirements. Section 5.4 contains environmental requirements with two subsections. Subsection 5.4.1 contains development environment requirements, and subsection 5.4.2 contains execution environment requirements.

## **5.2 Functional Requirements**

The School-Tutoring Web Application once completed shall provide all of the following requirements included in each subsection. The School-Tutoring Web Application is divided into five main subsections. The user registration subsection will examine and provide the necessary requirements for each user prior and during the registration process. The tutor profile subsection is important for emphasizing the key elements each tutor type account shall have, and this will provide the core structure for most of the system overall functionalities to work. The dashboard subsection is where the user spends most of his or her time, as it contains references to their messages, posts, history of posts, and settings. The main page subsection is where all the posted jobs, and tutor reviews are shown. It serves as the landing page after signing in. The user types and roles subsection is there to explain the 'shall' requirements for each type of user, and therefore, will provide extensive functional requirements to be implemented.

### **5.2.1 User Registration**

The users of the School-Tutoring Web Application will be able to navigate to a registration page where they will be able to choose their user's account type and provide the necessary information for creating a new user account.

#### ***5.2.1.1 Student Account Registration Type***

The user shall be able to choose to register for a student account.

#### ***5.2.1.2 Student Account Registration Email***

The student shall provide his/her name and email address for a student account.

#### ***5.2.1.3 Student Account Registration Password***

The student shall provide an 8-character password before registering for a student account.

#### ***5.2.1.4 Tutor Account Registration Type***

The user shall be able to choose to register for a tutor account.

#### ***5.2.1.5 Tutor Account Registration Email***

The tutor shall provide his/her name and email address for a tutor account.

#### ***5.2.1.6 Tutor Account Registration Password***

The tutor shall provide an 8-character password before registering for a tutor account.

#### ***5.2.1.7 Tutor Account Validation***

The tutor shall go through a validation process after providing the above requirements.

#### ***5.2.1.8 Tutor Account Terms and Conditions***

The tutor shall be able to agree on the terms and conditions prior to submitting his or her validation request.

#### ***5.2.1.9 Tutor Account Approval and Rejection***

The tutor shall be approved by the admin or rejected prior to using the application.

### **5.2.2 Tutor Profile**

The tutor after being approved for an account should go to create a profile page. The content of the profile page will be used as the primary indicator of why a student should hire a certain tutor.

#### ***5.2.2.1 Tutor Profile Biography***

The tutor shall be able to provide a fully detailed biography.

#### ***5.2.2.2 Tutor Expertise***

The tutor shall be able to provide his/her field of expertise.

#### ***5.2.2.3 Tutor Experience***

The tutor shall be able to provide his/her years of experience.

#### ***5.2.2.4 Tutor Education***

The tutor shall be able to provide his/her education.

#### ***5.2.2.5 Tutor Profile Validation***

The tutor shall be able to submit the profile for admin review before publishing.

### **5.2.3 The Dashboard**

The dashboard page will be the user's primary page to edit their profile, access the messages page, check the progress of their current posts, review their history of posts, and edit the general settings of their account.

#### ***5.2.3.1 Dashboard Access***

The tutor user type and the student user type shall be able to access the dashboard.

#### ***5.2.3.2 Dashboard Side Navigation Bar***

The side navigation bar shall contain references to the profile page, messages page, active posts page, history page and the settings page.

#### ***5.2.3.3 Dashboard Side Navigation Bar***

The side navigation bar should contain the user's profile image.

#### ***5.2.3.4 Dashboard Side Navigation Bar***

The side navigation bar should contain the user's name and email on top.

#### ***5.2.3.5 Dashboard Main Container***

The main container shall display the content of the referenced pages in the side navigation bar.

### **5.2.4 The Main Page**

The main page is the website's primary page after the user is registered. This page is where the new posts are posted and listed. Also, this page will contain the list of the best tutors and their star reviews.

#### ***5.2.4.1 Main Page Posting Area***

The main page shall contain a posting area.

#### ***5.2.4.2 Main Page Posting Area visibility***

The main page shall contain a posting area that is only visible to students.

#### ***5.2.4.3 Main Page New Posts List Area***

The main page shall contain a list area for the new posts.

#### ***5.2.4.4 Category of The New Post***

The posting area shall allow the student to choose which category the desired help falls into.

#### ***5.2.4.5 Title of The New Post***

The posting area shall allow the student to specify a title for the new post.

#### ***5.2.4.6 Due Date of The New Post***

The posting area shall allow the student to specify the due date for the new post, based on a time zone.

#### ***5.2.4.7 Content of The New Post***

The posting area shall allow the student to write the question of the content of the new post.

#### ***5.2.4.8 Content upload of The New Post***

The posting area shall allow the student to upload documents from their computer.

#### ***5.2.4.9 Price of The New Post***

The posting area shall allow the student to specify the amount of money they are willing to pay.

#### ***5.2.4.10 Main Page New Posts Visibility***

The main page shall allow students and tutors to see the list of all the new posts.

#### ***5.2.4.11 New Post Proposal Option***

The main page shall allow only tutors to see and use the “propose” button next to a new post.

#### ***5.2.4.12 New Post Review Option***

The main page shall allow only tutors to see and use the “review” button next to a new post.

#### ***5.2.4.13 New Post Review***

The main page shall allow tutors to review the post in the same page.

#### ***5.2.4.14 New Post Page.***

The main page shall allow tutors to go to the post’s own page for more information.

#### ***5.2.4.15 Main Page Best Tutors of The Month.***

The main page will show a list of the best tutors in this month.

#### ***5.2.4.16 Main Page Best Tutors of All Time***

The main page will show a list of the best tutors in all times.

#### ***5.2.4.17 Main Page Tutor’s Highest Grosser***

The main page will show a list of the highest grosser tutors of all times.

### **5.2.5 User Types and Roles**

The School-Tutoring Web Application will have three types of users with different privileges and Roles for each type.

#### ***5.2.5.1 Admin’s User Management***

The admin shall be able to add, edit, and remove users.

#### ***5.2.5.2 Admin’s Posts Management***

The admin shall be able to edit and delete students’ posts.

#### ***5.2.5.3 Admin's Tutor Submissions Management***

The admin shall be able to manage tutor's submissions.

#### ***5.2.5.4 Admin Blocking Authority***

The admin shall be able to block violators from using the application.

#### ***5.2.5.5 Admin Messages Monitoring***

The admin shall be able to monitor messages between students and tutors.

#### ***5.2.5.6 Admin Money Refund Authority***

The admin shall be able to refund money to students when submissions go wrong.

#### ***5.2.5.7 Admin Tutor Account Validation***

The admin shall be able to validate tutor accounts for legitimacy.

#### ***5.2.5.8 Student Registration***

The student shall be able to register for a free account.

#### ***5.2.5.9 Student Dashboard Access***

The student shall be able to access the dashboard.

#### ***5.2.5.10 Student New Post***

The student shall be able to post a new job.

#### ***5.2.5.11 Student New Post Details***

The student shall be able to add details to a new job.

#### ***5.2.5.12 Student New Post Edits and Delete Options***

The student shall be able to edit and delete his or her own posts.

#### ***5.2.5.13 Student Proposal Review***

The student shall be able to review the submitted proposals by tutors.

#### ***5.2.5.14 Student Proposal Decision***

The student shall be able to accept or reject the submitted proposals by tutors.

#### ***5.2.5.15 Student Payment Release***

The student shall be able to submit payment after receiving the submission.

#### ***5.2.5.16 Student Messages***

The student shall be able to message available tutors.

#### ***5.2.5.17 Student Messaging History***

The student shall be able to see the history of their messages.

#### ***5.2.5.18 Student Posts History***

The student shall be able to see his or her own posts history.

#### ***5.2.5.19 Student Notification***

The student shall be able to see new notifications and respond to them.

#### ***5.2.5.20 Student Profile***

The student should be able to create a brief profile that contains their essential information.

#### ***5.2.5.21 Student Settings***

The student shall be able to edit their account settings.

#### ***5.2.5.22 Student Login and Logout***

The student shall be able to login and logout of the system.

#### ***5.2.5.23 Tutor Registration***

The tutor shall be able to register for a free account.

#### ***5.2.5.24 Tutor Login and Logout***

The tutor shall be able to login and logout of the system.

#### ***5.2.5.25 Tutor Registration***

The tutor shall be able to access the dashboard.

#### ***5.2.5.26 Tutor New Post Proposal***

The tutor shall be able to propose to the student to do his or her new job.

#### ***5.2.5.27 Tutor New Post Proposal Details***

The tutor shall be able to add details description to his or her proposal for a new job.

#### ***5.2.5.28 Tutor New Post Proposal Edits and Delete Options***

The tutor shall be able to edit and delete his or her own proposal.

#### ***5.2.5.29 Tutor Help Submission***

The tutor shall be able to submit his or her helpful notes to student posts.

#### ***5.2.5.30 Tutor Payment Receiving***

The tutor shall be able to receive payment after the student accepts the submitted answer.

#### ***5.2.5.31 Tutor Messages***

The tutor shall be able to message available students.

#### ***5.2.5.32 Student Messaging History***

The student shall be able to see the history of their messages.

#### ***5.2.5.33 Tutor Posts History***

The tutor shall be able to see his or her own-posted answers' history.

#### ***5.2.5.34 Tutor Notification***

The tutor shall be able to see new notifications and respond to them.



#### **5.2.5.35 Tutor Profile**

The tutor should be able to create a full profile that contains their essential information provided earlier.

#### **5.2.5.36 Tutor Settings**

The tutor shall be able to edit their account settings.

### **5.3 Performance Requirements**

The following performance requirements will provide the response time for the major tasks the School-Tutoring web application performs. The response time for each requirement has been obtained by actually performing a request call for each one from the client side to the server-side. Then, the process is repeated multiple times to record the average run-time for the following requirements. Therefore, the application shall maintain the same recorded values when deployed. Also, the application shall maintain the same run-time average when scaled for larger number of users.

#### **5.3.1 Authentication Requests Response Time**

This performance subdivision will show the average run-time for various authentication requests between the UI and the Firebase Server, and are listed as performance requirements.

##### **5.3.1.1 New User Signup**

The signup request for a new user shall return the first result in an average ranging from 0.44 seconds to 0.55 seconds.

##### **5.3.1.2 User Logging-In**

The login request for a new user shall return the first result in an average ranging from 0.69 seconds to 0.9 seconds.

##### **5.3.1.3 Main Page Navigation**

The navigation between the main pages shall return the first result in an average ranging from 0.19 seconds to 0.23 seconds.

##### **5.3.1.4 Dashboard Page Navigation**

The navigation between the pages in the dashboard shall return the first result in an average ranging from 0.20 seconds to 0.27 seconds.

##### **5.3.1.5 User Logout**

The user logout request shall return the first successful result in an average ranging from 0.072 seconds to 0.1 seconds.

#### ***5.3.1.6 User Logout Redirect To Login Page***

The redirect call to “login” page after logging out shall return the first successful result in an average ranging from 0.092 seconds to 0.123 seconds.

#### ***5.3.1.6 User Logout Redirect To Login Page***

The redirect call to “login” page after logging out shall return the first successful result in an average ranging from 0.092 seconds to 0.123 seconds.

### **5.3.2 Notifications Requests Response Time**

This performance subdivision will show the average run-time (performance) to notify users about new messages, payment notifications, and so on.  
(This subdivision will be implemented later.)

#### ***5.3.2.1 New Message Notification***

New message notifications shall be in real-time with delays no more than 0.025 seconds.

#### ***5.3.2.2 New Proposal Request Notification***

New proposal request notifications shall be in real-time with delays no more than 0.15 seconds.

#### ***5.3.2.3 New Payment Notification***

New payment submission notification shall be quick, and the delay time should not be more than 5 seconds from verifying the payment.

## **5.4 Environmental Requirements**

This section will cover all the environmental requirements in terms of the following two subsections, the development environment and the execution environment.

### **5.4.1 Development Environment Requirements**

This subsection includes the necessary software and hardware requirements for the development of the School-Tutoring web application.

#### ***5.4.1.1 Development Computer***

The school-Tutoring web application requires a development computer capable of running at least 3 software development application at one time.

The following are the minimum hardware requirements for the development computer:

Category	Requirement
Processor	Pentium II 266Mhz or any similar CPU
Hard Drive	Space 512MB
RAM	256MB
Display	800x600, 256 colors
Sound Card	Optional

#### **5.4.1.2 Local-host Python HTTP Server**

The python Simple HTTP Server will help running the development on the local computer. With the help of this HTTP server one can turn any directory in the system into a web server directory. The only thing required with this is to have Python installed.

#### **5.4.1.3 Sublime Text Editor**

To develop and write code for the School-Tutoring web application we will need to use Sublime Text, which is a cross-platform source code editor.

#### **5.4.1.4 Firebase Database & Server**

For the Tutoring Web application the data being store in a Firebase database, and the server side processes are all, also based on Firebase Google Cloud services, therefore using Firebase for real-time data connections and transfer for this application is essential.

#### **5.4.1.5 AngularJS**

The Tutoring application will require the usage of a structural framework for dynamic web applications, such as AngularJS. It is important because it will allow us to use HTML as our own template language and it will allow us to extend HTML's syntax to express our application's components clearly and succinctly. Angular's data binding and dependency injection eliminate much of the code that we would otherwise have to write. Therefore, Angular is the required JavaScript framework for the School-Tutoring web application.

#### **5.4.1.6 Angular-UI**

To develop the application faster and easier we will use the Angular-UI modules.

##### **5.4.1.1 Angular Material**

For the Tutoring app we will use Angular Material Design, which is a specification for a unified system of visual, motion, and interaction design that adapts across different devices and different screen sizes developed by Google.

## **5.4.2 Execution Environment Requirements**

This subsection includes the necessary software and hardware requirements for the execution requirements of the School-Tutoring web application.

#### **5.4.2.1 Computer Browser**

To run the application one will need any new browser with a release date starting at least from the year of 2009, such as Google Chrome, Mozilla Firefox, Safari, Windows Explorer or the new Windows Edge.

#### **5.4.2.2 Smartphones**

The Tutoring application will run on any smart phone, as the design of the front-end will be responsive on any screen size.

#### **5.4.2.3 Server**

The application will be hosted on a Firebase server, and the server shall be able to provide the needed minimum requirements to run such an application and accommodate as many requests from users as possible..