**Requirements Document for Student Education Management Application**

|  |  |
| --- | --- |
| **1 Introduction** |  |
| 1.1 Background |  |
| **2 Requirement Identification** |  |
| 2.1. Functional Requirements |  |
| 2.2. Non-Functional Requirements |  |
| **3 User Personas** |  |
| **4 Use Cases** |  |
| **5 User Stories and Acceptance Criteria** |  |
| **6 Constraints and Assumption** |  |

**Table of Contents**

**1 Introduction**

**1.1 Background**

The Student Education Management Application is designed to assist school students in managing their educational activities, including homework, performance tracking, and communication with teachers and parents. The primary goal of the application is to provide a user-friendly, secure, and engaging platform to enhance the learning and management experience for students.

**2 Requirement Identification**

**2.1 Functional Requirements**

* **User Authentication**
* Users (students, parents, teachers) can register with valid credentials.
* Authentication mechanisms (e.g., email verification, two-factor authentication, OTP) ensure secure access.
* **User Profiles**
* Users can create and manage their profiles with personal information.
* Students can specify their grade level and school information.
* **Homework Management**
* **Task Creation**: Students can create tasks for their homework, specifying the subject, due date, and description.
* **Task Tracking**: Students can mark tasks as completed, and the system should update their progress.
* **Task Reminders**: Students receive notifications or reminders about upcoming homework deadlines.
* **Attachments**: Students can attach files or links related to their homework tasks.
* **Exam Management**
* **Exam Creation:** Teachers can create MCQ exams, specifying the subject due date, and description.
* **Exam Report Generation:** The report of the exam should be generated automatically.
* **Ensuring Fairness**: A system that prevents the students from switching to new tabs or minimizing the app during the exams.
* **Performance Tracking**
* **Grades Management**: Students can enter their grades for assignments, quizzes, and exams.
* **Grade Analytics**: The system generates visual reports (charts/graphs) to display the student's performance trends.
* **Goal Setting**: Students can set academic goals, and the system provides progress updates.
* **Feedback Collection**: Teachers can provide feedback on assignments, and students can view this feedback.
* **Attendance Tracking**: Students can mark their attendance for classes.
* **Communication**
* **Messaging System**: Students, teachers, and parents can send and receive messages within the app.
* **Announcements**: Teachers can post announcements or important information.
* **Parent-Teacher Communication**: Parents can schedule meetings with teachers and receive academic updates.
* **Chatbot Support**: An AI-powered chatbot can answer common queries related to the education system.
* **Discussion Forums**: Students can participate in subject-specific discussion forums.
* **Resources and Materials**
* Teachers can upload learning materials, such as study guides, notes, learning videos and reference materials.
* Students can access and download these resources.
* **Progress Reports**
* Automated generation and distribution of progress reports to parents.
* Performance summaries and teacher feedback.
* **Live Classroom**
* Teachers can schedule live class and students can participate

**2.2 Non-Functional Requirements**

* **Security**
  + Data encryption during transmission and storage.
  + Role-based access control (students, parents, teachers).
  + Regular security audits and updates.
* **Usability**
  + Intuitive and user-friendly interface.
  + Accessibility features for users with disabilities.
  + User should be able to access essential features offline
* **Performance**
  + Low latency and fast response times.
  + Scalability to accommodate increasing user numbers.
* **Reliability**
  + Backup and recovery mechanisms to prevent data loss.
* **Availability**
* The App should be available 24/7
* **Privacy**
  + Compliance with data privacy regulations (e.g., GDPR, COPPA).
  + Transparent data usage policies.
* **Integration**
  + Compatibility with various operating systems (iOS, Android, web).
  + Integration with popular education platforms (e.g., Google Classroom).

**3 User Personas**

Student Persona 1:

* Name: Sajib
* Age: 13
* Grade: 8th
* Tech-savviness: Beginner
* Learning needs: Sajib needs an app that is easy to use and navigate. He wants to be able to use the app to view his homework assignments, track his progress, and communicate with his teachers.

Student Persona 2:

* Name: Ratul
* Age: 16
* Grade: 11th
* Tech-savviness: Advanced
* Learning needs: Ratul is highly motivated and wants to excel in school. He needs a tool to help him track his progress over time and identify areas where he can improve. He needs the resources and materials to improve in his study.

Student Persona 3:

* Name: Fahim
* Age: 17
* Grade: 12th
* Tech-savviness: Intermediate
* Learning needs: Struggles with time management. Enjoys interactive learning experiences. Wants to collaborate with classmates on projects.

**4 Use Cases**

**Use Case 1:** **Student Adds Homework Task**

* **Actor:** Student
* **Description:** Student adds a new homework task.
* **Preconditions:** Student is logged in.

**Main Scenario:**

1. Student logs in.
2. Student navigates to the homework management section.
3. Student selects "Add New Homework."
4. Student fills in assignment details (subject, description, due date etc).
5. Student saves the assignment.

**Alternate Scenario:**

* If the student cancels the action, the assignment is not saved.

**Use Case 2: Teacher Sends Announcement**

* **Actor:** Teacher
* **Description:** The teacher generates and sends a progress report to a student's parent.
* **Preconditions:** Teacher is logged in.

**Main Scenario:**

1. Teacher selects "Send Announcement."
2. Teacher enters the announcement message.
3. Teacher selects the class or group to send the announcement to.
4. Teacher clicks "Send."

**Alternate Scenario:**

* If the teacher encounters an issue while sending announcement, they can contact technical support.

**Use Case 3: Teacher Sends Progress Report**

* **Actors**: Teacher
* **Description:** The teacher generates and sends a progress report to a student's parent.
* **Preconditions:** Teacher is logged in

**Main Scenario:**

* Teacher selects the student and generates a progress report.
* Teacher reviews and adds comments to the report.
* Teacher sends the report to the student's parent.

**Alternate Scenario:**

* If the teacher encounters an issue while generating the report, they can contact technical support.

**5 User Stories**

**User Story 1: Student Adds Homework**

**As a student, I want to be able to add homework assignments so that I can keep track of my assignments and due dates.**

**Acceptance Criteria:**

* I can log in with my credentials.
* I can navigate to the homework management section.
* I can select "Add New Homework."
* I can fill in assignment details (subject, description, due date).
* I can save the assignment.
* The assignment is displayed in my homework list.

**User Story 2: Teacher Sends Progress Report**

**As a teacher, I want to generate and send progress reports to students' parents so that they can stay informed about their child's performance.**

**Acceptance Criteria:**

* I can log in with my teacher credentials.
* I can select a student and generate a progress report.
* I can review and add comments to the report.
* I can send the report to the student's parent.
* The parent receives the progress report in their inbox.

**6 Constraints and Assumptions**

**Constraints**

1. **Technical Constraints**
   * The application must be compatible with iOS, Android, and web platforms.
   * Integration with external apps may depend on their APIs.
2. **Legal Constraints**
   * The application must comply with data privacy regulations (e.g., GDPR, COPPA).
   * User data should be stored securely and used only for authorized purposes.

**Assumptions**

* Users have access to smartphones or computers with an internet connection.
* Users are willing to adapt to the new system.
* The development team has the necessary skills and resources to implement the desired features.