

EduGenius Software Requirements Specification For EduGenius

Version 1.0

Team Members:

PERERA S.A.I.M.	210471F
PRABASHWARA D.G.H.	219483T
PRANAVAN S.	210491P

Group ID:	21
Project ID:	22
Mentor:	Dr. Chathuranga Hettiarachchi

Revision History

Date	Version	Description	Author
17/07/2024	1.0	Initial Version	Group 21

Table of Contents

1. Introduction	4
1.1. Purpose	4
1.2. Scope	4
1.3. Definitions, Acronyms, and Abbreviations	4
1.4. References	4
1.5. Overview	4
2. Overall Description	5
2.1. Product Perspective	5
2.2. Product Functions	5
2.3. User Characteristics	5
2.4. Constraints	6
2.5. Assumptions and Dependencies	6
3. Specific Requirements	6
3.1. Functionality	8
3.2. Usability	8
3.3. Reliability	9
3.4. Performance and Security	9
3.5. Supportability	10
3.6. Design Constraints	10
3.7. On-line User Documentation and Help System Requirements	11
3.8. Purchased Components	11
3.9. Interfaces	15
3.10. Database Requirements	15
3.11. Licensing, Legal, Copyright, and Other Notices	15
3.12. Applicable Standards	16
4. Supporting Information	16
4.1. Index	16
5. References	17

EduGenius	Version: 1.0
Software Requirements Specification	Date: 17th July 2024

Software Requirements Specification

1. Introduction

1.1. Purpose

This Software Requirements Specification (SRS) outlines the functional requirements, non-functional requirements of a mathematics tutoring application called **EduGenius**. This document also gives a thorough explanation about the design constraints of the system and supplementary information in order to design and deploy the application. It serves as a guideline for developers, project managers, testers, and other stakeholders involved in the software development lifecycle.

1.2. Scope

EduGenius is a web application created for providing interactive guidance through O/L math problems based on exam point of view. This application uses data science technologies such as Retrieval Augmented Generation, Large Language Models and Object Character Recognition to achieve its intended purpose.

Students can either upload their answer for the math problem given by the system as an image or draw their answer on the canvas. Their response is evaluated by the system then feedback and explanation will be given. Further, students can receive tips during the attempt of the questions. Finally, students can ask questions from the system clarifying their doubts regarding the elaboration given by the system.

1.3. Definitions, Acronyms, and Abbreviations

The definitions, acronyms and abbreviations are added under Index of supporting information(section 4).

1.4. References

The references are added under the reference section of supporting information (section 4).

1.5. Overview

The remaining part of this document will provide information about the functional, non-functional requirements, product's function, design constraints, interfaces, standards followed by the system and other requirements (Database requirements etc.)

Under the overall description (section 2) section high-level view of the functions, user characteristics, design constraints and requirements subsets will be provided.

Under specific requirements section (section 3), functional requirements, non-functional requirements, design constraints, interfaces, application standards and other specific requirements will be discussed in detail. Additionally, license and copyright information will be provided there.

The later part of this document (section 4) will provide supplementary information such as appendix and index.

2. Overall Description

2.1. Product Perspective

The EduGenius Mathematics Tutoring Application is a new product developed to provide an enhanced and personalized learning experience for students preparing for Ordinary Level (O/L). It leverages advanced educational technologies, including Retrieval-Augmented Generation (RAG), to dynamically generate contextually relevant guidance and explanations. EduGenius offers a comprehensive and interactive tutoring experience by delivering real-time feedback, and performance tracking. This ensures that students receive effective and engaging support throughout their math learning journey.

2.2. Product Functions

2.2.1. User

- Register/Create a new account
- Login to the account
- Update user profile information
- Provide mathematical problems and hints for the problem if user is stuck on the problem
- Take user's answers for the problem and give feedback
- Provide answers for problem related questions by user
- Get user feedback on tutoring
- Logout from the account

2.2.2. Administrator

- Adding more content
- Manage user accounts
- Generate reports

2.3. User Characteristics

2.3.1. Characteristics

'Users' are the set of people(O/L students) who have created accounts on the EduGenius system with the purpose of using the system to get tutoring on mathematics problems. They shall share their user experience inorder to improve the system.

2.3.2. Requirements:

Users should be able to create a new account in the system, to login to the account, to update the profile information, to get mathematical problems, to get feedback on the answers, to ask questions about the problem and to get marks for the answer.

2.4. Constraints

2.4.1. Regulatory Policies

The EduGenius application will comply with the General Data Protection Regulation and the regional regulations that govern their respective data protection policies. Additional features will be implemented to enforce the regulations regarding the prohibited contents.

2.4.2. Language requirements

The EduGenius application will only support the English language and will use unicode character encodings.

2.4.3. Communication Protocols

HTTPS: The EduGenius application will use HTTPS protocol to view web pages.

Web Sockets: The EduGenius application will use Web Sockets for real-time, bi-directional instant messaging for tutoring sessions.

2.4.4. Security Considerations

User Identity Verification - Registering/Logging in with Authentication link sent to the email entered by the user upon registration.

2.4.5. Design conventions

Modular Design: The design of the EduGenius application primarily follows modular design and implementations. Modules for authentication and context generation will be implemented individually and integrated together.

Naming Conventions: Snake case is used to name variables, functions, classes, and files of the project.

Code Documentation: Code will adhere to the naming conventions and will be commented consistently. Class-level, file-level documentation are expected to be there. Documentations should include the respective versions of themselves. (i.e. Documentation versions should follow the project version)

External Dependencies: Gemini APIs will be used as the LLM for this project while Wolfram Alpha Show Steps API will be used as a math solver tool. Vision models will be used for image recognition.

2.5. Assumptions and Dependencies

- There should be a seamless internet connection.
- Users have installed any web browser to open the links. (No built-in features provided for that)

3. Specific Requirements

3.1. Functionality

- 3.1.1. Users should be able to register to the system using his/her email address. Users who wish to access tutoring sessions of the EduGenius, first have to

register to the system. To register to the system, the first user has to go to the registration page. Then they have to provide the email address. After submitting it the system will send a token to that email address. The user has to click on the link that is provided in the email to verify the email. After that the system will verify the email address and redirect the user to the password page to create the password. Users have to provide the password and confirmation password and then submit the form. After that the system will create the user instance with the email address and the password and will log in the user to the platform.

- 3.1.2. Users should be able to login to the platform and should be able to recover passwords.

Users who already have user accounts of the platform have to first go to the login page to login to the system. In there users can provide the email address and the password and press submit. If the system manages to find a user instance with that email address, the system will check whether the password associated with the user instance and the password provided by the user is the same. If the passwords are the same user will be logged in to the platform. If user passwords do not match, then the system will show an error message to the user indicating an incorrect password. If the system fails to find a user instance associated with the provided email address in the database, then the user will be redirected to the register page along with the error message indicating that an account associated with the email address does not exist.

In case the user forgot the password to the account, the user can select the "forgot password" link to recover the password. After clicking the link the user will be redirected to the password recovery page, where the user has to provide the email and click send message. After that the system will first check if there exists an user instance with the provided email address. If not, the user will show an error message and will redirect to the register page. If a corresponding user instance is found, the system will send an email to the user email address with a token, while informing the user that email has been sent. User has to go to that email and click on the link that is provided. After that the system will redirect the user to the password resetting page where the user has to provide a new password and confirmation password and submit. At that point the system will change the password of the user instance password and log in the user with a success message indicating success password change.

- 3.1.3. Users should be able to select math problems and answer them while having tips.

Users should be able to choose a math problem from the provided problems. Then the answer for the math problem can be provided using the canvas, by uploading an image or by providing a text. If a user is not able to solve the problem, they can ask for a tip from the system. System will provide a tip to help the user to solve the problem.

- 3.1.4. Users should be able to get feedback from the system and marks for the answers
If the answers are provided for a selected math problem, the system will generate positive feedback if the answer is correct, and if the answer is incorrect, the system will generate feedback explaining the errors made in the answer. The marks will be awarded for correct answers and correct steps.

- 3.1.5. Users should be able answer questions to clarify issues related to the math problem

Once the feedback is provided by the system, the user is able to ask any questions related to the math problem. If the questions are related to the problem, they will be answered by the system, otherwise the system will inform

the user that the question is not regarding the math problem.

3.1.6. User should be able to provide a feedback on the tutoring

Once the user has done answering the math problem and asking questions regarding the math problem, the user should be able to give feedback on his experience on that specific tutoring session. It is not mandatory for users to provide feedback. When the user provides feedback, the system will record it along with the tutoring session data.

3.1.7. Users should be able to view the past tutoring sessions

To view all the previous tutoring sessions the user had, the user has to visit the history page. The system will provide a list of all past tutoring sessions with the obtained marks in that page. Users can select one of those sessions to view the interactions. Then the system will redirect the user to that tutoring session page which contains all the interactions.

3.1.8. Users should be able to view and update their personal informations

Inorder to view or change the user informations, user has to first login to the system and navigate to the profile page. There all the information will be displayed. Users will be able to edit the personal information except the email address by clicking the edit button. Once the information is changed as needed, users have to press the save button to save. At that point the system will update the user instance and reload the page

3.1.9. User should be able to use the math solving tool

To use the math solving tool users first have to login to the system. Users provide the math problem he/she needs to be solved as a text. The system will take user problems and will send them to the math solving api and get results. Those results will be presented to the user.

3.2. Usability

3.2.1. Usability Requirement-1: Training Time

A normal user, with basic computer literacy, should be able to learn to navigate and use the primary functions of the application within 30 minutes of initial use. A power user, experienced with similar software, should be able to master the advanced features within 15 minutes of initial use.

3.2.2. Usability Requirement-2: Task Times

- Creating and logging into the system will take no longer than 5 minutes.
- Selecting and attempting a question will take no longer than 1 minute from the dashboard.
- Submitting the responses and getting the evaluation will take no longer than 3 minutes.
- Accessing the explanation after submitting the response will take no longer than 1 minute.
- Receiving hints during an attempt at a math problem will take no longer than 1 minute.

3.2.3. Usability Requirement-3: Consistent and Simple User Interface

- User interface should be consistent and simple, avoiding multiple colors.
- Users should be able to easily navigate through UI and use the system effortlessly.

3.3. Reliability

3.3.1. Reliability Requirement-1 : Availability

The automatic tutoring application must be available for use 90% of the time, ensuring minimal downtime and high availability for users. Scheduled maintenance should be limited to non-peak hours, with advance notice provided to users. In the event of degraded mode operations, essential functionalities should remain accessible.

3.3.2. Reliability Requirement-2 : MTBF

The application should have a Mean Time Between Failures (MTBF) of at least 3,000 hours. This high MTBF ensures that the system operates reliably for extended periods without experiencing failures.

3.3.3. Reliability Requirement-3 : MTTR

In case of a system failure, the Mean Time To Repair (MTTR) must be less than 7 hours. This ensures that the system is restored to full functionality quickly, minimizing disruption for users.

3.3.4. Reliability Requirement-4 : Accuracy

The system's outputs, including answers to questions and retrieval of information, must be accurate and precise, adhering to educational standards. The system should maintain an accuracy rate of at least 75% for responses based on the context provided.

3.3.5. Reliability Requirement-5 : Maximum Defect Rate

The application should have a maximum bug rate of 1bug per thousand lines of code (bugs/KLOC). This low defect rate ensures high code quality and reliable system performance.

3.3.6. Reliability Requirement-5 : Defect Rate Categorization

- Critical Bugs: Complete loss of operation of the system or inability to use the basic features such as attempting a quiz or getting tutoring from the system.
- Significant Bugs: Partial loss of functionality or performance degradation.
- Minor Bugs: Minor issues that do not significantly impact usability. For example, animation on the user interface is not rendering.

3.4. Performance and Security

3.4.1. Performance Requirement-1: Response Time

Submitting the responses and getting the evaluation will take no longer than 3 minutes.

3.4.2. Performance Requirement-1: Capacity

System should be able to handle 1000 users concurrently.

3.4.3. Performance Requirement-1: Degradation Modes

In case of partial system failures or high load conditions, the system should degrade gracefully, maintaining core tutoring features. Non-essential services,

such as performance tracking may be temporarily disabled to ensure core functionality.

3.5. Supportability

3.5.1. Supportability Requirement -1: Coding Standards

The application must adhere to established coding standards to ensure consistency, readability, and maintainability of the codebase.

- **Commenting and Documentation:** Each function and class should have clear, concise comments explaining its purpose and usage.
- **Code Reviews:** All changes to the codebase must undergo peer reviews to ensure quality and adherence to standards.

3.5.2. Supportability Requirement -2: Naming Conventions

Consistent naming conventions must be used throughout the codebase. This includes:

- **Variables and Functions:** Use snake case (e.g. `calculate_score`).

3.6. Design Constraints

EduGenius aims to provide an engaging and interactive learning experience to students by adhering to several design constraints to ensure effective and efficient development.

3.6.1. Standards Compliance

The platform must comply with educational technology standards and data privacy regulations, including GDPR for data protection, COPPA for children's online privacy, and FERPA for student data privacy.

3.6.2. Software Languages

Development will primarily be done using Python for backend functionalities and frameworks such as Streamlit will be used for building the web application.

3.6.3. Development Tools

The project will use Git for version control, GitHub for code repository, and Docker for containerization.

3.6.4. Architectural Constraints

The system will be designed using a microservices architecture to ensure scalability, maintainability, and flexibility.

3.6.5. Third Party Components

The project will use third-party APIs for integrating external services such as Large Language Models (LLMs), complex math solving tools, vision models for image recognition, tracing tools for evaluating LLM outputs, user feedback

evaluation tools, and payment gateways for subscription services.

3.6.6. Hardware Limitations

The platform will be designed to be compatible with a wide range of devices, including desktops, laptops, tablets, and smartphones, to ensure accessibility for all users without requiring high-end hardware specifications.

3.7. On-line User Documentation and Help System Requirements

3.7.1. Interactive Help

An integrated help system offering brief, context-specific tips and explanations displayed when users click on specific interface elements like expanders.

3.7.2. Support Portal

Users will have access to report issues and troubles they are facing while using the application, directly to the developers.

3.7.3. Help Notices

Regular help notices and updates will inform users about new features, maintenance schedules, and other important information.

3.8. Purchased Components

EduGenius will use several third-party purchased components to enhance the functionality and user experience of the platform. These components include:

3.8.1. Large Language Models (LLMs) : Gemini-Pro, Gemini-Flash

EduGenius will leverage LLMs to provide advanced natural language processing capabilities for text analysis, question answering, and content generation.

3.8.2. Math Solving Tools : Wolfram Alpha Step-by-Step API

Application is integrated with Wolfram Alpha's math solving API to provide step-by-step solutions to complex math problems.

3.8.3. LLM Output Tracing Tool and User Feedback Evaluation Tool : Langfuse

EduGenius will use Langfuse to trace the outputs of LLMs and evaluate user feedback to improve the quality of responses.

3.8.4. Email APIs : Brevo

The platform will use Brevo's email APIs for sending notifications, updates, resetting passwords, and other email-related functionalities.

3.8.5. Payment Gateway : Stripe

EduGenius will integrate Stripe's payment gateway for handling subscription services, payments, and transactions on the platform.

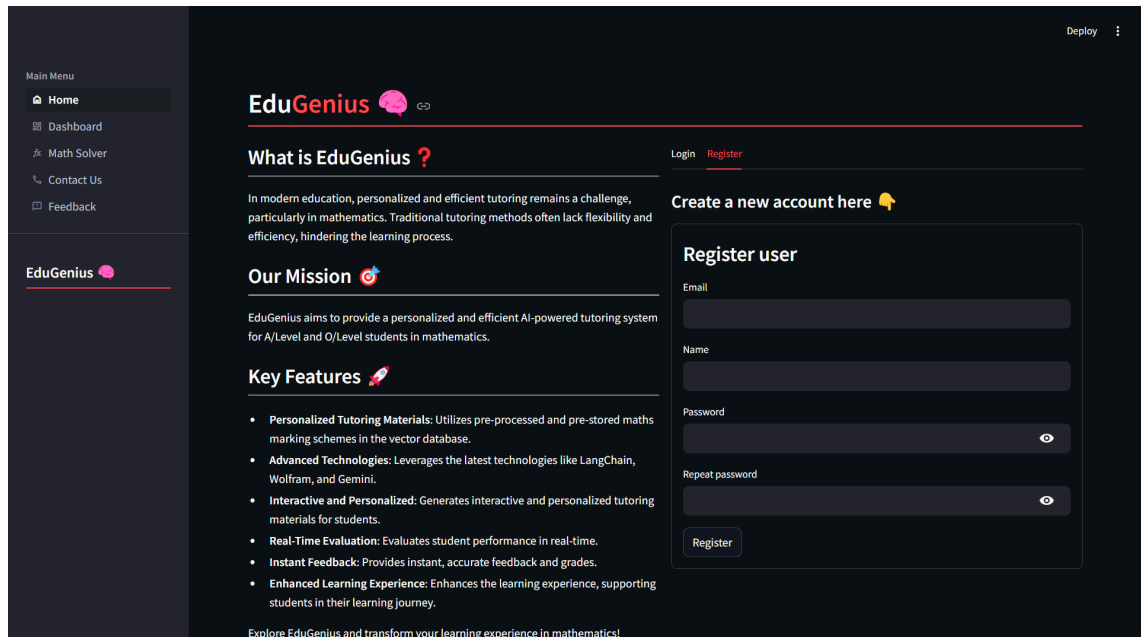
3.9. Interfaces

The EduGenius platform will support various interfaces to ensure a seamless and functional user experience, as well as efficient communication and data handling.

3.9.1. User Interfaces

- Login Page

- Functionalities : User authentication and access control.
- Elements : Two forms for login and registration, brief description about the system.



- Dashboard

- Functionalities : Users can do math problems or obtain hints for them interactively. Users are able to upload handwritten answers or write their answers on the provided canvas.
- Elements : Select boxes to select preferred year and a preferred math problem, buttons to obtain hints and proceed with the provided answer and navigation button to EduGenius Math Solver if the user has a separate math problem.

Home

Dashboard

Contact Us

Feedback

Tools

Math Solver

Research Helper

View less

EduGenius

Welcome to EduGenius

Let's get started!

Select a paper

2018 OLevel

Select a question

Part A Problem (20)

20. Based on the information given in the Venn diagram, write the set $A' \cup B'$ in terms of its elements.

Want a hint?

It is worth giving it a try

✓ Here what we found for you!

Hint 1

Hint 1

Start by identifying the elements of A' (A complement). A complement includes all the elements that are **not** in set A .

Hint 2

Upload your answer. Let's see how you did!

Drag and drop file here

Limit 200MB per file • PNG, JPG, JPEG

Browse files

Don't have a piece of paper? Write here!

Proceed

- Result and Performance Analysis
 - Functionalities : Display quiz results and performance analysis, suggestions for the student to improve his answers, clean explanations for the problem and even similar math problems with answers to practice.
 - Elements : Multiple expanders containing each section and like dislike buttons to obtain user satisfaction level with the provided answers.

Home

Dashboard

Contact Us

Feedback

Tools

Math Solver

Research Helper

View less

EduGenius

Results

Correctness

Partially Correct

Marks

1

Explanation

Explanation

The problem requires finding the elements of the set $A' \cup B'$, which represents the union of the complements of sets A and B .

Understanding the Concepts:

- Complement ($'$):** The complement of a set contains all elements in the universal set that are **NOT** in the original set. In this case, the universal set is $\{1, 2, 3, 4, 5\}$.
- Union (\cup):** The union of two sets combines all elements from both sets.

Solving the Problem:

- Find A' (Complement of A):** A' includes all elements *not* in A . Looking at the Venn diagram:
 $A = \{1, 2, 4\}$
 $A' = \{3, 5\}$
- Find B' (Complement of B):** Similarly:
 $B = \{1, 2, 3\}$
 $B' = \{4, 5\}$
- Find $A' \cup B'$ (Union of A' and B'):** Combine the elements of A' and B' :
 $A' \cup B' = \{3, 4, 5\}$

Improvements

Improvements

The student correctly identified element '4' which belongs to B' and is part of the solution. However, they missed including element '5' which is present in both A' and B' , and therefore also belongs to their union. The student should review the concept of the union of sets to ensure all elements from both sets are included.

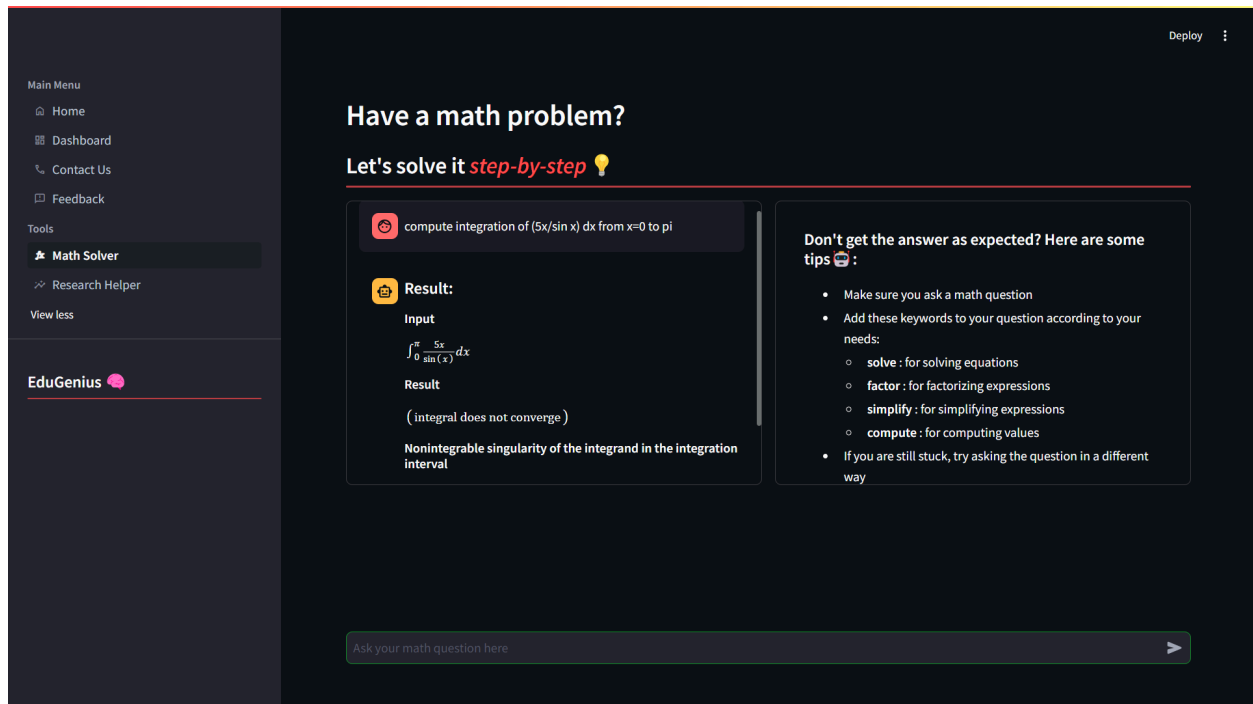
Similar Problems

Confidential

©EduGenius, 2024

Page 13

- EduGenius Math Solver
 - Functionalities : Users can ask their math problems in the input prompt.
 - Elements : Input field to enter the math problem user has, a chat interface between the user and the bot, and some guidelines for obtaining better results from the API.



3.9.2. Software Interfaces

The EduGenius platform requires robust software interfaces to ensure secure server connectivity, seamless integration with external services, and compatibility with major web browsers.

- Server Connectivity :Secure connections to backend servers using HTTPS and Web Socket protocols
- Web Services/Plugins: Integration with external services like Wolfram, Gemini, Langfuse, and data handling and storage services like MongoDB
- Supported Browsers: Google Chrome (Version 80 and above), Mozilla Firefox (Version 75 and above) , Microsoft Edge (Version 80 and above), Safari (Version 13 and above)

3.9.3. Hardware Interfaces

For optimal performance and user experience on the EduGenius platform, the following minimum client-side hardware prerequisites are recommended. Please note that these specifications are not strict minimum requirements, but rather to ensure the best possible interaction with the platform.

- Processor : Dual-core processor, 2.0 GHz or higher
- Memory: 4GB or higher, DDR3 or DDR4

- Storage: Minimum 500 MB of free space for browser cache and temporary files.
- Network: Stable internet connection, Ethernet or Wi-Fi (802.11n or higher)
- Peripheral Devices: Input devices (keyboard, mouse)

3.9.4. Communication Interfaces

EduGenius utilizes various communication interfaces to enable efficient data exchange and real-time interactions within the platform.

- HTTP Requests: Asynchronous HTTP/HTTPS protocol requests over the internet for data retrieval and submission, such as fetching quiz questions, and submitting answers.
- WebSocket: For real-time updates and notifications, ensuring seamless interacting and communication within the platform.

3.10. Database Requirements

For the EduGenius project, several database requirements are essential to support its functionality and manage data effectively. Here are the key database requirements:

3.10.1. User Account Management:

- MongoDB: Utilize MongoDB, a NoSQL database, to store and manage user accounts. MongoDB's flexibility in handling unstructured data and scalability makes it suitable for managing user profiles, authentication details, and preferences.

3.10.2. Content Storage:

- ChromaDB: Employ ChromaDB, a vector database system, to store data sources related to mathematical problems, quizzes, and educational content. ChromaDB's capability to handle high-dimensional data and perform similarity searches makes it ideal for storing diverse educational materials.

3.11. Licensing, Legal, Copyright, and Other Notices

For the EduGenius project, intending to use the MIT License is a strategic choice for ensuring flexibility and openness in software distribution. The MIT License is permissive, allowing users to freely use, modify, and distribute the software, including commercially, without imposing many restrictions beyond requiring attribution and including the original copyright and license notice in any copies or substantial portions of the software.

Under the MIT License:

- Legal Disclaimers: It disclaims warranties, limiting liability for the software's use "as is" without warranties of any kind.
- Patent Notices: There are no explicit patent grants, meaning users must negotiate separately for patent rights.
- Trademark Compliance: It doesn't mandate compliance with trademarks, but users must not use the project's name or trademarks in ways that imply endorsement.

Usage Restrictions: Users can modify the software and integrate it into proprietary software, but they must include the original license and copyright notice. They cannot hold the project or its

contributors liable for any issues arising from the software's use.

Copyright Notices: The license requires maintaining the copyright notice in all copies or substantial portions of the software.

By adopting the MIT License, EduGenius ensures that its software remains accessible, fosters community contributions, and enables widespread adoption while safeguarding contributors' and users' rights within the scope defined by the license.

3.12. Applicable Standards



- **Unicode Standard:** The application should implement the Unicode Standard to support internationalization, ensuring that text and data are consistently represented and manipulated across different languages and scripts.

4. Supporting Information

4.1. Index

- **LLMs** : Large language Models
- **RAG** : Retrieval-Augmented Generation
- **OTP** (One time password): A security measure commonly used in authentication systems to provide a temporary code for accessing a service or account. In the context of this document OTP refers to mobile OTPs: sending a One Time Password Via SMS to a Mobile Phone.
- **REQ:** Requirement
- **USB** : Universal Serial Bus
- **API** : Application Programming Interface
- **SDK** : Software Development Kit
- **HTTP/S** : HyperText Transfer Protocol
- **OCR:** Object Character Recognition
- **O/L:** General Certificate of Education Ordinary Level Examination
- **UI:** User Interface
- **MTTR:** Mean Time To Repair
- **MTBF:** Mean Time Between Failures
- **bugs/KLOC:** Defects per thousand lines of code

5. References

- [1] "Introduction |  LangChain." <https://python.langchain.com/v0.2/docs/introduction/> (accessed Jul. 07, 2024).
- [2] "Chroma Docs." <https://docs.trychroma.com/> (accessed Jul. 07, 2024).
- [3] "What is MongoDB? - MongoDB Manual v7.0." <https://www.mongodb.com/docs/manual/> (accessed Jul. 08, 2024).
- [4] "Gemini API Developer Docs and API reference," Google for Developers. <https://ai.google.dev/gemini-api/docs> (accessed Jul. 08, 2024).
- [5] "Get started with LangSmith |  LangSmith." <https://docs.smith.langchain.com/> (accessed Jul. 08, 2024).
- [6] "Langfuse documentation - langfuse." <https://langfuse.com/docs> (accessed Jul. 08, 2024).
- [7] "Get started with Streamlit - Streamlit Docs." <https://docs.streamlit.io/get-started> (accessed Jul. 08, 2024).
- [8] "Wolfram|Alpha Show Steps API: Reference & Documentation." <https://products.wolframalpha.com/show-steps-api/documentation> (accessed Jul. 09, 2024).
- [9] Modran, H., Bogdan, I. C., Ursuțiu, D., Samoila, C., and Modran, P. L., "LLM Intelligent Agent Tutoring in Higher Education Courses using a RAG Approach," Preprints, 2024. [Online]. Available: <https://doi.org/10.20944/preprints202407.0519.v1> (accessed Jul. 10, 2024).
- [10] "Build a Q&A App with Multi-Modal RAG using Gemini Pro | Google Codelabs," Google Codelabs. <https://codelabs.developers.google.com/multimodal-rag-gemini#4> (accessed Jul. 10, 2024).