HOMECARESERVICEFORSENIORS

# 21IT902–ADVANCEDAPPLICATIONDEVELOPMENT

***Submittedby***

|  |  |
| --- | --- |
| **MOHAMMED SHARAFATH S** | **(727722EUCD023)** |
| **ABHISHECK P O** | **(727722EUCD001)** |
| **HARI GOVINDH** | **(727722EUCD016)** |

***in partial fulfilment for the award of the degree of***

**BACHELOR OF ENGINEERING**

**IN**

COMPUTER SCIENCE AND DESIGN

# SRIKRISHNACOLLEGEOFENGINEERINGANDTECHNOLOGY

**(AnAutonomousInstitution,AffiliatedtoAnnaUniversityChennai-600025)**

MARCH2024



**BONAFIDECERTIFICATE**

Certified that this project report titled **“HOME CARE SERVICE FOR SENIORS”** is thebonafideworkof**SAHANADEVIP(727721EUIT136),SUBAHARININ(727721EUIT161),**

**SWETHAS(727721EUIT168)**whocarriedouttheprojectworkundermysupervision.

# SIGNATURE SIGNATURE

Dr.N.SUSILA Ms.R.JANANI

# HEADOFTHEDEPARTMENT SUPERVISOR

Professor AssistantProfessor

InformationTechnology, InformationTechnology,

SriKrishnaCollegeofEngineeringand Sri Krishna College of Engineering andTechnology, Technology,

Kuniyamuthur,Coimbatore–641008. Kuniyamuthur,Coimbatore–641008**.**

SubmittedfortheProjectviva-voceexaminationheldon

INTERNALEXAMINER EXTERNALEXAMINER

# ACKNOWLEDGEMENT

At this juncture, we take the opportunity to convey our sincere thanks andgratitudetothemanagementofthecollegeforprovidingallthefacilitiestous.

Wewishtoconveyourgratitudetoourcollegeprincipal**,Dr.S.Sophia**forforwarding usto do our project and offering adequate duration to complete ourproject.

We would like to express our grateful thanks to **Dr. N. Susila** Head of thedepartment, Department of Information Technology for her encouragement andvaluableguidancetothisproject.

Weextendmygratitudetoourbelovedguide**Ms.R.Janani,ME.**,**MBA.,(PhD).,**AssistantProfessor,Department of Information Technology for herconstantsupportandimmensehelpatallstagesoftheproject.

# ABSTRACT

**MY TUTOR: An Innovative Self-Learning Application**

In the evolving landscape of education, self-learning has become increasingly prevalent as students seek to acquire knowledge and skills independently. However, this mode of learning presents significant challenges, including a lack of interactive feedback, engagement issues, and limited access to collaborative tools. Addressing these challenges, MY TUTOR emerges as a pioneering solution designed to enhance the self-learning experience.

MY TUTOR is a comprehensive educational application that integrates advanced technologies to support self-learners in their educational pursuits. The application offers a suite of features aimed at improving learning efficiency and engagement. Key functionalities include summarization tools that provide concise overviews of complex topics, interactive quizzes that adapt to the user's knowledge level, and customizable flashcards for effective memorization and review.

The core of MY TUTOR's functionality lies in its use of the Gemini API, which processes uploaded content to generate detailed summaries and facilitate interactive learning experiences. This integration ensures that users receive accurate and relevant content tailored to their study needs. The application’s user-friendly interface, built using React for the frontend and Python with Django for the backend, ensures a seamless and responsive experience.

Beyond its core features, MY TUTOR addresses the inherent challenges of self-learning by offering personalized feedback and progress tracking. The interactive quizzes provide immediate insights into a user’s understanding, while flashcards aid in quick revision and retention of key concepts. The application is designed to be intuitive and engaging, making it easier for students to stay motivated and effectively manage their study sessions.

Looking ahead, MY TUTOR plans to further enhance its capabilities with the addition of audio-to-text functionality. This feature will enable users to practice verbal skills by converting spoken responses into text, providing real-time feedback on pronunciation and fluency. This advancement will enrich the self-learning experience by addressing both written and spoken aspects of education.

# TABLE OF CONTENTS

|  |  |  |
| --- | --- | --- |
| **CHAPTER** | **TITLE** | **PAGE**  **No.** |
|  | **ACKNOWLEDGEMENTABSTRACT**  **LIST OF TABLES LIST OF FIGURES**  **LIST OF ABBREVIATIONS** | **iiiivviiviiiix** |
| **1** | **INTRODUCTION** | **1** |
|  | 1.1 OVERVIEW | 1 |
|  | 1.2 COMPONENTS OF SYSTEM | 3 |
|  | 1.3 ADVANCED TECHNOLOGIES | 4 |
|  | 1.4 GLOBAL PERSPECTIVES | 6 |
| **2** | **SYSTEM ANALYSIS** | **7** |
|  | 2.1 EXISTING SYSTEM | 7 |
|  | 2.1.1 DRAWBACKS | 8 |
|  | 2.2 PROBLEM DEFINITION | 8 |
|  | 2.3 PROPOSED SYSTEM | 9 |
|  | 2.3.1 ADVANTAGES | 9 |
| **3** | **SYSTEM REQUIREMENTS** | **10** |
|  | 3.1 HARDWARE REQUIREMENTS | 10 |
|  | 3.2 SOFTWARE REQUIREMENTS | 10 |
|  | 3.3 SOFTWARE DESCRIPTION | 10 |
|  | 3.3.1 FRONTEND | 10 |
|  | 3.3.2 BACKEND | 13 |

1. SYSTEM DESIGN 15
   1. [MODULE DESCRIPTION 15](#_TOC_250016)
      1. [PROVIDERS MANAGEMENT 15](#_TOC_250015)
      2. [SERVICE MANAGEMENT 16](#_TOC_250014)
      3. [BOOKING MANAGEMENT 17](#_TOC_250013)
      4. [USER PROFILE MANAGEMENT 18](#_TOC_250012)
   2. [USE CASE DIAGRAM 19](#_TOC_250011)
   3. [SEQUENCE DIAGRAM 19](#_TOC_250010)
   4. [DATA FLOW DIAGRAM 20](#_TOC_250009)
2. TESTING 21
   1. [UNIT TESTING 21](#_TOC_250008)
   2. [INTEGRATION TESTING 21](#_TOC_250007)
   3. [SECURITY AND AUTHENTICATION](#_TOC_250006)
   4. [TEST CASES 23](#_TOC_250005)
      1. [TESTCASE I 23](#_TOC_250004)
      2. [TESTCASE II 24](#_TOC_250003)
3. CONCLUSION AND FUTUREWORK 25
   1. [CONCLUSION 25](#_TOC_250002)
   2. [FUTUREWORK 25](#_TOC_250001)
4. APPENDICES 26

APPENDIX I

SOURCECODE 26

APPENDIX II

SCREENSHOTS 32

[REFERENCES 39](#_TOC_250000)

**LISTOFTABLES**

|  |  |  |
| --- | --- | --- |
| **TABLENO.** | **TITLE** | **PAGE** |
|  |  | **No.** |
| 4.1 | ProvidersManagement | 15 |
| 4.2 | ServiceManagement | 16 |
| 4.3 | BookingsManagement | 17 |
| 4.4 | UserProfileManagement | 18 |

# LISTOFFIGURES

|  |  |  |
| --- | --- | --- |
| **FIGURENo.** | **TITLE** | **PAGE** |
|  |  | **No.** |
| 3.1 | VSCodeLogo | 10 |
| 4.1 | UseCaseDiagram | 17 |
| 4.2 | SequenceDiagram | 18 |
| 4.3 | DataFlowDiagram | 18 |
| 5.1 | Storingthetokeninlocalstorage | 22 |
| 5.2 | Authenticatingtheuserusing |  |
|  | bearertoken | 22 |
| 5.3 | TestcaseI | 23 |
| 5.4 | TestcaseII | 24 |
| A.2.1 | Signup | 32 |
| A.2.2 | Login | 32 |
| A.2.3 | Landingpage | 33 |
| A.2.4 | Landing page(Dark Theme) | 33 |
| A.2.5 | Recent Files | 34 |
| A.2.6 | Summary | 34 |
| A.2.7 | Flashcards | 35 |
| A.2.8 | Quiz | 35 |
| A.2.9 | Features Section | 36 |
| A.2.10 | About Section | 36 |
|  |  |  |
|  |  |  |
|  |  |  |

**LIST OF ABBREVIATIONS**

|  |  |  |
| --- | --- | --- |
| **S.No** | **ABBREVIATIONS** | **EXPANSION** |
| 1 | ADL | ActivitiesofDailyLiving |
| 2 | RAM | RandomAccessMemory |
| 3 | GB | GigaBytes |
| 4 | VS | VisualStudio |
| 5 | OS | OperatingSystem |
| 6 | HTTP | HyperTextTransferProtocol |
| 7 | JPA | JavaPersistenceAPI |
| 8 | API | ApplicationProgramming |
|  |  | Interface |
| 9 | JDBC | JavaDatabaseConnectivity |
| 10 | SQL | SequentialQueryLanguage |
| 11 | UI | UserInterface |
| 12 | DOM | DocumentObjectModel |
| 13 | JSX | JavaScriptXML |
| 14 | JWT | JSONWebToken |
| 15 | UML | UnifiedModellingLanguage |
| 16 | DFD | DataFlowDiagram |
| 17 | FAQ | FrequentlyAskedQuestions |

# OVERVIEW

**CHAPTER 1 INTRODUCTION**

As self-directed learning gains popularity, it presents a transformative

opportunity for individuals to pursue knowledge and skills independently. However, this approach introduces several challenges that traditional educational resources often fail to address effectively. Self-learners frequently encounter issues such as a lack of interactive engagement, insufficient feedback, and difficulties in maintaining motivation. These challenges can lead to gaps in understanding and hinder progress, highlighting the need for a more integrated and supportive solution.

To address these needs, the MY TUTOR project has been developed to provide a comprehensive and user-centric app designed specifically for self-learners. This innovative application aims to revolutionize the self-learning experience by offering a centralized platform that combines several advanced educational tools into one cohesive system. The primary goal of MY TUTOR is to bridge the gap between traditional educational methods and the evolving needs of modern learners, providing a seamless and engaging learning experience.

At the heart of MY TUTOR are its core features: summarization, interactive quizzes, and customizable flashcards. The summarization tool is designed to distill complex topics into clear and concise summaries, allowing users to quickly grasp essential concepts and review critical information without feeling overwhelmed. This feature is especially valuable for learners who need to process large amounts of material efficiently, providing a streamlined approach to understanding key points.

The interactive quizzes feature of MY TUTOR offers a dynamic way to assess and reinforce learning. These quizzes are tailored to the user’s knowledge level and adapt to their progress, providing immediate feedback to help identify strengths and areas for improvement. This real-time feedback mechanism ensures that learners can continuously track their understanding and adjust their study strategies accordingly, maintaining engagement and motivation throughout their learning journey experience for users around the world.

In addition to summarization and quizzes, MY TUTOR includes customizable flashcards, which serve as a flexible tool for memorization and review. Users can create and personalize flashcards based on their study needs, allowing for targeted revision and ongoing reinforcement of key concepts. This feature supports a personalized learning approach, enabling users to focus on areas where they need the most practice.

By integrating these features into a single platform, MY TUTOR addresses several critical challenges faced by self-learners. It enhances learning efficiency by streamlining study sessions and providing interactive content that keeps users engaged. The app’s personalized feedback and adaptive quizzes ensure that learners receive tailored support and can track their progress effectively. Moreover, the customizable flashcards offer a convenient way to reinforce knowledge and prepare for assessments.

In summary, MY TUTOR is poised to transform the landscape of self-directed learning by offering a robust and integrated platform that meets the diverse needs of modern learners. Through its innovative features and user-centric design, MY TUTOR aims to enhance educational outcomes, support effective study practices, and promote a more engaging and personalized learning

# COMPONENTS OF SYSTEM

MY TUTOR consists of several key components, each contributing to the overall functionality and effectiveness of the application. These components are carefully designed to meet the diverse needs of self-learners and ensure a seamless user experience.

**Home:** The Home component serves as the central hub of the application. It provides users with an overview of their study progress, quick access to different features, and a summary of recent activities. The Home page is designed to be user-friendly, allowing users to easily navigate between different sections of the app.

**Sign In:** The Sign In component is crucial for user authentication and personalization. It allows users to log in to their accounts or create new ones, ensuring that their data and progress are securely stored. The Sign In feature includes options for both traditional email/password authentication and social media logins, offering flexibility and convenience.

**Summary:** This component provides users with concise and clear summaries of their study materials. By distilling complex topics into manageable chunks, the Summary feature helps learners quickly understand and retain essential information. This component is particularly useful for reviewing large volumes of material in a short period.

**Flashcard:** The Flashcard component is designed to enhance memorization and revision. Users can create custom flashcards with key terms and concepts, which can be reviewed at any time. The interactive nature of flashcards makes them an effective tool for reinforcing learning and testing knowledge on the go.

**Quiz:** The Quiz component offers interactive assessments to evaluate users’ understanding of the study material. Quizzes are designed to adapt to the user’s level of knowledge, providing a personalized learning experience. Immediate feedback helps users identify areas for improvement and track their progress.

**Recent Files:** The Recent Files component allows users to manage and access recently viewed or created documents and study materials. This feature helps users quickly locate important files and continue their studies without having to navigate through multiple folders.

Each component of MY TUTOR is integrated to work seamlessly with the others, creating a cohesive learning experience that addresses the various needs of self-learners.

# ADVANCED TECHNOLOGIES

MY TUTOR integrates several cutting-edge technologies to deliver its features and ensure an optimal user experience. Among these technologies, the Gemini API plays a pivotal role in processing uploaded files, enhancing the application's capabilities.

**Gemini API Integration:**

A central component of MY TUTOR's functionality is its use of the Gemini API for processing uploaded files. The Gemini API is a powerful tool designed to handle various types of data inputs, including text documents, spreadsheets, and other file formats commonly used in educational contexts. Here’s how the Gemini API contributes to MY TUTOR:

**File Parsing and Analysis:**

The Gemini API enables MY TUTOR to efficiently parse and analyze the contents of uploaded files. This includes extracting relevant information, organizing data, and converting it into a format suitable for summarization and other educational purposes. By leveraging advanced algorithms, the API ensures that the data is processed accurately and promptly.

**Text Summarization:**

One of the key features of MY TUTOR is its ability to provide concise summaries of complex texts. The Gemini API supports this functionality by analyzing the content of uploaded documents and generating summaries that highlight key points and essential information. This helps users quickly grasp important concepts without having to sift through lengthy documents.

**Data Extraction:**

For files containing structured data, such as spreadsheets, the Gemini API facilitates the extraction and organization of data. This enables MY TUTOR to present information in a user-friendly format, such as interactive tables or charts, enhancing the learning experience by making data more accessible and understandable.

**Compatibility and Flexibility:**

The Gemini API is designed to handle a wide range of file formats and types. This compatibility ensures that users can upload and process various types of educational materials, including text files, PDFs, and more. The flexibility of the API allows MY TUTOR to accommodate diverse user needs and preferences.

**Efficiency and Scalability:**

With the Gemini API, MY TUTOR can process large volumes of data efficiently. The API’s scalable architecture ensures that the application can handle an increasing number of users and file uploads without compromising performance. This scalability is crucial for maintaining a smooth and responsive user experience as the app grows

**Responsive Design:**

MY TUTOR is designed with responsive design principles to ensure that the application is accessible and functional across various devices and screen sizes. Whether users are accessing the app on a desktop, tablet, or smartphone, the responsive design ensures a seamless and intuitive user experience.

By integrating these advanced technologies, including the Gemini API, MY TUTOR offers a sophisticated and efficient learning platform that meets the diverse needs of self-learners and enhances their educational experience.

# GLOBALPERSPECTIVES

MY TUTOR is positioned to address educational challenges on a global scale by aligning with international trends in e-learning, self-directed learning, and technological advancements. Here’s how MY TUTOR fits into the global educational landscape:

**1. Bridging the Education Gap**

In many regions worldwide, access to quality education remains a significant challenge. MY TUTOR helps bridge this gap by providing a platform that offers educational resources and tools to individuals regardless of their geographical location. By enabling self-learning through summarization, quizzes, and flashcards, MY TUTOR empowers users who may not have access to traditional educational resources or formal instruction.

**2. Supporting Lifelong Learning**

The concept of lifelong learning is increasingly emphasized across the globe. As the job market evolves and new skills are required, individuals are seeking flexible and efficient ways to continue their education. MY TUTOR caters to this need by offering a platform that supports ongoing learning. Users can revisit and update their knowledge, explore new subjects, and engage with interactive content, aligning with the global push for continuous personal and professional development.

**3. Enhancing Self-Learning Experiences**

Self-learning is gaining popularity as a method of education due to its flexibility and accessibility. MY TUTOR enhances self-learning experiences by providing tools that make studying more engaging and effective. Through features like interactive quizzes and customizable flashcards, MY TUTOR supports diverse learning styles and helps users stay motivated and on track with their educational goals.

**4. Leveraging Technological Advancements**

The integration of advanced technologies such as the Gemini API and Natural Language Processing (NLP) reflects a global trend towards leveraging technology to improve educational outcomes. By incorporating these technologies, MY TUTOR stays at the forefront of educational innovation, offering users sophisticated tools for processing information and personalizing their learning experience. This approach aligns with the global shift towards technology-driven education solutions.

**5. Addressing Diverse Educational Needs**

Educational needs vary widely across different cultures and regions. MY TUTOR is designed to be adaptable and responsive to diverse user requirements. The ability to process various types of educational materials and provide personalized feedback ensures that the platform can cater to users from different educational backgrounds and levels. This adaptability helps MY TUTOR serve a global audience, offering tailored solutions that meet the specific needs of individual learners.

**6. Promoting Inclusivity and Accessibility**

Inclusivity and accessibility are critical considerations in the global educational landscape. MY TUTOR is committed to providing an inclusive learning environment by ensuring that its features are accessible to all users, including those with disabilities. The platform’s responsive design and user-friendly interface contribute to its accessibility, allowing users with diverse needs to benefit from its educational tools.

**7. Encouraging Collaboration and Community**

While MY TUTOR primarily focuses on self-learning, it also fosters a sense of community and collaboration through its features. Users can share their progress, compare quiz results, and engage with others through the platform. This aspect of MY TUTOR supports the global trend towards collaborative learning and peer interaction, even in a self-directed learning environment.

In summary, MY TUTOR aligns with global educational trends and needs by offering a flexible, technology-driven learning platform that supports lifelong learning, addresses diverse educational needs, and promotes inclusivity. By bridging educational gaps and enhancing self-learning experiences, MY TUTOR contributes to the   
educationon a global scale.

# 2.1 EXISTING SYSTEM

**CHAPTER2**

**SYSTEM ANALYSIS**

Overview

Before the advent of sophisticated educational technology platforms like MY TUTOR, self-learners primarily relied on traditional methods of study which included textbooks, handwritten notes, and offline study aids. These methods, while foundational, have several limitations in addressing the dynamic needs of modern learners.

**Traditional Learning Methods**

Traditional learning systems involve the use of physical resources such as textbooks and printed notes. These methods are often static, providing a fixed amount of information that may not be updated or tailored to individual learning needs. Students typically engage in self-study by reading and reviewing these materials, but this approach lacks interactive elements and immediate feedback mechanisms.

**Digital Learning Tools**

With the rise of digital technology, various educational tools have been developed, including online courses, educational apps, and e-books. While these tools offer more flexibility compared to traditional methods, they often suffer from issues such as lack of integration, limited interactivity, and the absence of personalized learning paths. Many digital platforms provide generic content without considering individual learning preferences or progress.

**Limitations of Existing Systems**

Lack of Personalization: Existing systems often fail to adapt to individual learning styles and preferences. Users receive the same content regardless of their prior knowledge or learning pace.

**Limited Interactivity:**

Many digital learning tools lack interactive features that engage users and enhance their learning experience. Traditional methods and even some digital tools do not offer real-time feedback or adaptive learning experiences.

**Static Content:**

Traditional educational resources are static and may become outdated over time. Users cannot easily update or customize the content according to their evolving needs.

**Fragmented Experience:**

Users often have to switch between multiple platforms or tools to get a comprehensive learning experience. This fragmented approach can lead to inefficiencies and a lack of coherence in the learning process.

# DRAWBACKS

Identified Issues in Existing Systems

**1. Limited Feedback Mechanisms**

Existing systems generally provide minimal feedback to learners, which is crucial for understanding and improving performance. Without immediate feedback, learners may not be aware of their mistakes or areas that need improvement, hindering their learning progress.

**2. Low Engagement and Motivation**

Traditional methods and many digital tools lack engaging features that can maintain a learner's interest. The absence of interactive elements such as quizzes, games, or dynamic content often results in lower motivation and less effective learning outcomes.

**3. Inefficient Study Processes**

Manual methods of studying, such as note-taking and reviewing textbooks, can be time-consuming and inefficient. Learners often spend excessive time organizing and summarizing information, which could be streamlined with the right tools.

**4. Accessibility Issues**

Many existing educational resources are not accessible to all learners, particularly those with disabilities or those who lack access to the latest technology. This can create barriers to effective learning and limit opportunities for a diverse range of users.

**5. Lack of Integration**

Educational tools are often not integrated with one another, leading to a disjointed learning experience. For example, users may need to use separate tools for summarization, quizzes, and flashcards, which can be cumbersome and inefficient.

# PROBLEMDEFINITION

Current Challenges in Self-Learning

**1. Fragmented Learning Experience**

The current educational landscape is characterized by a fragmented approach to self-learning, where students must piece together various resources and tools to create a comprehensive study plan. This fragmentation can lead to inefficiencies and a lack of coherence in the learning process.

**2. Inadequate Feedback and Assessment**

Without access to immediate feedback and assessment tools, learners struggle to gauge their understanding and progress. The absence of real-time evaluations can result in prolonged gaps in knowledge and hinder overall learning effectiveness.

**3. Limited Engagement and Motivation**

Traditional and many digital learning methods fail to keep learners engaged and motivated. The lack of interactive and dynamic content often leads to disengagement and decreased motivation to continue studying.

**4. Personalization Deficit**

Existing systems often provide a one-size-fits-all approach to education, which does not cater to the individual needs and learning styles of users. This lack of personalization can make it challenging for learners to effectively grasp and retain information.

**5. Accessibility and Inclusivity Issues**

Many educational tools and resources are not accessible to all learners, particularly those with disabilities or those who lack advanced technological resources. This creates barriers to learning and limits the effectiveness of educational initiatives.

# 2.3 Proposed System

Overview of MY TUTOR

MY TUTOR is designed to address the limitations of existing educational systems by offering a comprehensive, integrated platform that enhances the self-learning experience. The system provides tools and features that cater to diverse learning needs and preferences, ensuring a more effective and engaging study process.

**1. Integrated Learning Platform**

MY TUTOR combines summarization, quizzes, flashcards, and other educational tools into a single platform. This integration streamlines the learning process, making it easier for users to access and manage their study materials.

**2. Personalized Learning Experience**

The platform uses advanced technologies to tailor the learning experience to individual needs. Users receive personalized content and feedback based on their progress and performance, ensuring a more effective and relevant study experience.

**3. Interactive Features**

MY TUTOR incorporates interactive elements such as quizzes and customizable flashcards to enhance engagement and motivation. These features make learning more dynamic and enjoyable, helping users stay committed to their educational goals.

**4. Real-Time Feedback and Assessment**

Immediate feedback on quizzes and assessments allows users to quickly identify and address areas of improvement. This real-time feedback mechanism supports continuous learning and helps users track their progress effectively.

**5. Accessibility and Inclusivity**

MY TUTOR is designed to be accessible to a wide range of users, including those with disabilities. The platform’s responsive design and user-friendly interface ensure that all learners can benefit from its features, regardless of their technological resources.

**2.3.1 Advantages**

Benefits of Implementing MY TUTOR

**1. Enhanced Learning Efficiency**

By integrating various educational tools into a single platform, MY TUTOR improves learning efficiency. Users can quickly access and utilize different features, saving time and streamlining their study process.

**2. Increased Engagement**

The interactive features of MY TUTOR, including quizzes and flashcards, help maintain user engagement and motivation. These tools make learning more enjoyable and effective, leading to better educational outcomes.

**3. Personalized Learning Paths**

MY TUTOR’s use of advanced technologies to personalize the learning experience ensures that users receive content and feedback tailored to their individual needs. This personalized approach enhances the effectiveness of the learning process and supports better knowledge retention.

**4. Real-Time Feedback**

Immediate feedback on assessments allows users to quickly identify and address their weaknesses. This real-time feedback mechanism supports continuous improvement and helps users achieve their learning goals more effectively.

**5. Accessible and Inclusive Design**

The platform’s focus on accessibility ensures that all users, including those with disabilities, can benefit from its features. MY TUTOR’s inclusive design promotes equal learning opportunities for a diverse range of users.

**6. Flexibility and Adaptability**

MY TUTOR’s flexible and adaptable features cater to various learning styles and preferences. Users can customize their study materials and learning paths to suit their individual needs, making the platform a versatile tool for self-learners.

**7. Global Applicability**

The platform’s ability to cater to diverse educational needs and preferences makes it suitable for users worldwide. MY TUTOR’s global applicability ensures that learners from different regions and backgrounds can benefit from its features and resources.

# CHAPTER 3

# SYSTEM REQUIREMENTS

# HARDWAREREQUIREMENTS

**Server Requirements:**

CPU: Quad-core processor or higher (e.g., Intel Core i5 or AMD Ryzen 5)

RAM: 8 GB or more

Storage: SSD with at least 100 GB of available space

Network: Reliable internet connection with minimum bandwidth of 1 Mbps

Operating System: Linux (Ubuntu or CentOS preferred) or Windows Server 2019

**Client Requirements:**

CPU: Dual-core processor (e.g., Intel Core i3 or AMD Ryzen 3)

RAM: 4 GB or more

Storage: Minimum 20 GB of free space

Network: Stable internet connection with at least 512 Kbps download speed

Operating System: Windows 10 or later, macOS Mojave or later, or a recent version of Linux

**Additional Hardware Considerations**

Backup Storage: External or cloud-based backup solutions to ensure data redundancy and recovery.

Development Environment: Development machines with higher specifications may be required for testing and development purposes, such as 16 GB RAM and multi-core processors

# SOFTWARE REQUIREMENTS

**Operating System:**

Linux (e.g., Ubuntu 20.04 LTS) or Windows Server 2019 for server environments.

**Database**:

MySQL: For managing relational data, ensuring robust data storage and retrieval capabilities.

Django: For managing database schemas.

**Backend Frameworks and Libraries:**

Python: Primary programming language for server-side logic and API development.

FastAPI: For building high-performance RESTful APIs with Python, ensuring rapid and efficient communication between the frontend and backend.

Spring Boot: For building and managing microservices, enhancing modularity and scalability of the backend system.

**Web Server:**

Nginx or Apache: For handling HTTP requests and serving the application to users efficiently.

**Containerization:**

Docker: For creating, deploying, and running applications in containers, ensuring consistency across different environments.

Client-Side Software

**Frontend Framework:**

React: JavaScript library for building dynamic and responsive user interfaces, ensuring a seamless user experience.

Web Browser:

Modern web browsers such as Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge for accessing the MY TUTOR application.

**Development Tools**

**IDE/Editor:**

VS Code: Recommended for coding in JavaScript (React) and Python.

**Version Control:**

Git: For source code management and collaboration.

# SOFTWARE DESCRIPTION

Overview

This section provides detailed descriptions of the software components used in the MY TUTOR project, including frontend and backend technologies, and their specific roles in the system.

# FRONTEND

**React**

Purpose: React is used to build the user interface of MY TUTOR. It allows for the creation of dynamic and interactive web pages by leveraging reusable components.

Features: Component-Based Architecture: Facilitates the development of modular and maintainable code by breaking down the user interface into smaller, reusable components.

Virtual DOM: Enhances performance by minimizing direct manipulation of the real DOM, resulting in faster rendering and updates.

State Management: Handles complex state management within the application using hooks or state management libraries like Redux.

Key Advantages:

Responsiveness: Ensures that the application is responsive and works well on various devices, including desktops, tablets, and smartphones.

Developer Efficiency: Streamlines the development process with tools and libraries that improve productivity and code maintainability.

# BACKEND

**Python**

Purpose: Python serves as the primary programming language for backend development, providing a flexible and powerful environment for server-side logic.

Features: Ease of Learning and Use: Python’s simplicity and readability make it an ideal choice for backend development.

Rich Ecosystem: Extensive libraries and frameworks available for various backend needs.

**FastAPI**

Purpose: FastAPI is used to build and manage RESTful APIs, enabling efficient communication between the frontend and backend.

Features: Performance: Provides high-performance API endpoints with minimal latency.

Automatic Documentation: Generates interactive API documentation automatically, enhancing developer productivity.

Spring Boot

Purpose: Spring Boot is utilized for developing microservices that handle specific functionalities within the backend.

Features: Microservice Architecture: Facilitates the creation of modular and scalable backend systems.

Integration: Seamlessly integrates with other Java-based technologies and services.

**Django**

Purpose: Django is used as an ORM for interacting with the MySQL database, managing database schemas, and handling data migrations.

Features: Built-in Admin Interface: Provides a ready-to-use interface for managing database records and user interactions.

Robust Security: Includes built-in security features to protect against common vulnerabilities.

MySQL

Purpose: MySQL is the relational database management system used to store and manage data for the MY TUTOR application.

Features: Reliability: Known for its robustness and reliability in handling large volumes of data.

Scalability: Supports scaling to accommodate growing data needs.

# CHAPTER 4

# SYSTEM DESIGN

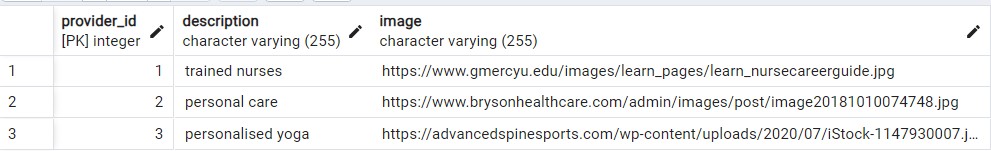
# MODULEDESCRIPTION

* + - Providersmanagement
    - Servicemanagement
    - Bookingmanagement
    - Userprofilemanagement

# PROVIDERSMANAGEMENT

Theadminmanagesserviceprovidersbyadding,editing,anddeletingtheirprofiles.Thisincludesmanagingqualifications,availability,andcontactinformation. This ensures that users have access to accurate and up-to-dateinformation aboutservice providers.

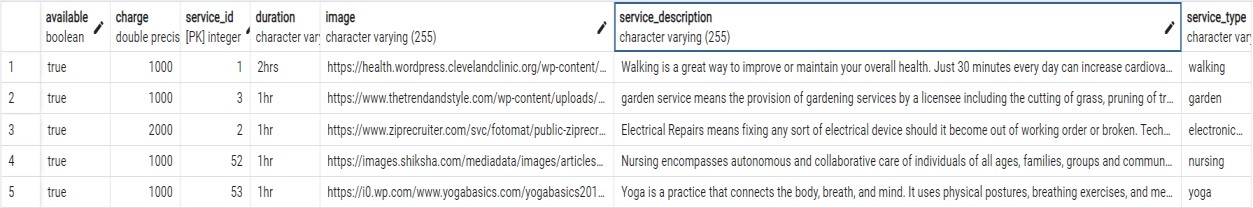
# Table4.1.ProvidersManagement



# SERVICEMANAGEMENT

The service management system organizes and controls various aspects ofservices to ensure efficient delivery and customer satisfaction. These aspectsinclude specifying the service type, describing the service, setting the charge,defining the duration, and indicating service availability. Efficient managementoftheseaspectshelpsensuresmoothservicedelivery,meetscustomerexpectations,and maintainscompetitivepricing.

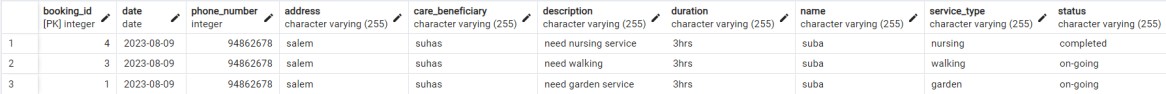
# Table4.2.ServiceManagement



# BOOKINGMANAGEMENT

In the booking management, various details are managed to facilitate thescheduling and tracking of services. These details include the user's name andmobile number for identification, service type, date and timings for scheduling.Additionally,informationaboutthecarebeneficiary,address,andbookingstatus(which is managed by the admin) is also tracked. Efficient management of thesedetails helps ensure that bookings are accurately scheduled, communicated, andexecuted,leading to improvedcustomerserviceandsatisfaction.

# Table4.3BookingManagement



# USERPROFILEMANAGEMENT

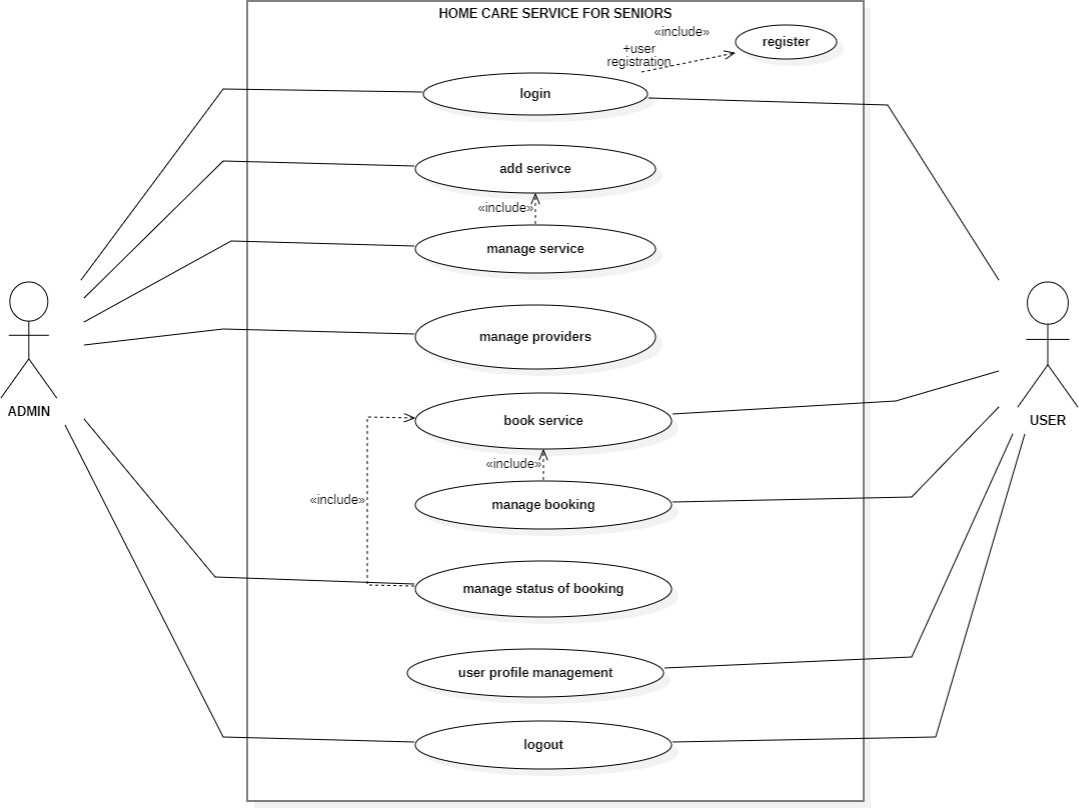
Intheuserprofilemanagementsystem,userscanmanagetheirdetailssuchas their name, email ID, and mobile number. However, while users can updatetheirmobilenumber,theycannotedittheirnameandemailID.Thissetupensuresthat critical user information remains accurate and secure, while still allowinguserssome controlover theirprofiles.

# Table4.4.UserProfileManagement



# USECASEDIAGRAM

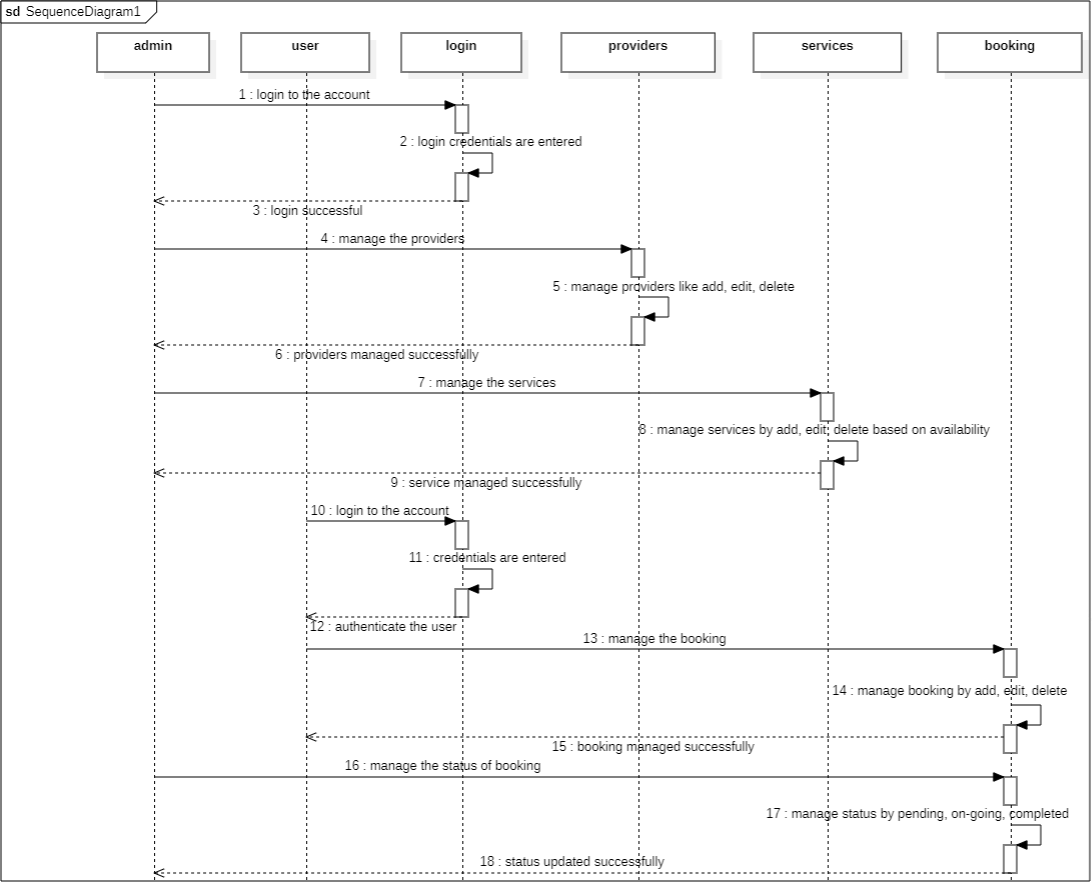
A use case diagram is a visual representation in UML (Unified ModellingLanguage)thatillustratestheinteractionsbetweenactorsandasystemorsoftwareapplication.Itisusedtodepictthevariouswaysusersorexternalentitiescaninteractwiththesystemandthespecificfunctionalitiesorusecasesitoffers.



# Fig.4.1.UseCaseDiagram

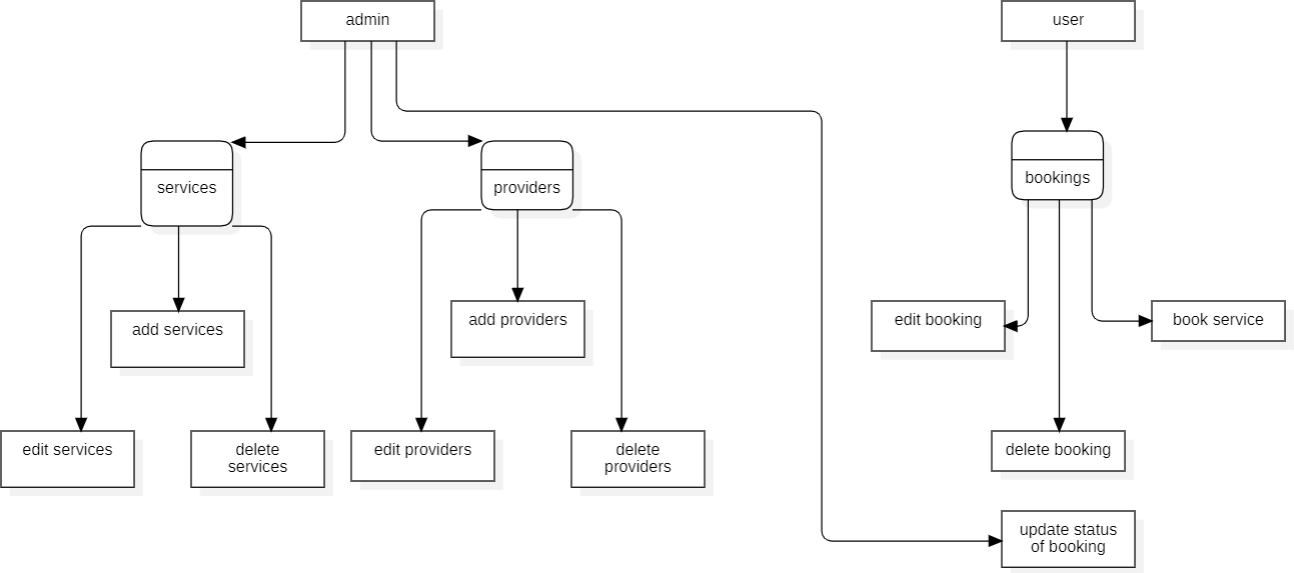
# SEQUENCEDIAGRAM

Asequencediagramisavisualrepresentationusedinsoftwareengineeringto illustrate the interactions and communication between different objects orcomponentsinasystemoveraspecificperiodoftime.Itshowsthechronologicalorder of messages or method calls exchanged between these entities, helping todepict the dynamic behaviour of a system or a particular scenario. In essence, itprovides a time-ordered view of how various parts of a system collaborate toachieveaparticulartaskorfunctionality.



# Fig.4.2.SequenceDiagram

# DATAFLOW DIAGRAM

A Data Flow Diagram (DFD) is a visual representation that illustrates theflowofdatawithinasystemorprocess.Itusessymbolstodepictprocesses,datastores,dataflow,andexternalentities.DFDhelpstoanalyse,design,anddocumentinformationsystems,showinghowdataisinput,processedandoutput,whileemphasizingtheinteractionsbetweendifferentelementsinthesystem.

# Fig.4.3.DataFlowDiagram

# CHAPTER 6

**CONCLUSION AND FUTUREWORK**

# CONCLUSION

MY TUTOR is a cutting-edge educational application designed to enhance the self-learning experience for students. Through a suite of innovative features, including summarization tools, interactive quizzes, and customizable flashcards, MY TUTOR addresses key challenges faced by self-learners, such as lack of immediate feedback, engagement issues, and limited interaction.

**Key Achievements:**

Enhanced Learning Efficiency: MY TUTOR’s summarization capabilities allow students to quickly grasp and review complex topics, making study sessions more effective. The interactive quizzes and flashcards provide a streamlined approach to reinforcing learning and tracking progress.

Increased Engagement: The application’s interactive elements, such as quizzes and flashcards, help maintain student motivation and engagement. The integration of these tools ensures that students stay actively involved in their learning journey.

Personalized Feedback: Immediate feedback from quizzes and progress tracking features help students identify their strengths and areas needing improvement. This personalized approach guides students towards achieving their educational goals more effectively.

Technological Integration: The use of advanced technologies such as the Gemini API for file processing enhances the app's capabilities, ensuring efficient content handling and user-friendly interactions.

Overall, MY TUTOR successfully provides a comprehensive platform for self-learners, improving their study habits and overall educational experience. The application combines innovative technology with user-centric design to deliver an effective learning tool that addresses common challenges in self-education.

# FUTUREWORK

As technology and user needs evolve, there are several opportunities for enhancing MY TUTOR to provide even more value to its users. One of the primary areas for future development includes:

**Audio-to-Text Functionality for Practice Recitation:**

**Objective:** Implementing audio-to-text functionality will allow users to practice recitation by converting their spoken responses into text. This feature will enable students to improve their pronunciation, fluency, and overall verbal skills.

**Features:**

Speech Recognition: Utilize advanced speech recognition technology to transcribe audio input into text accurately.

Real-Time Feedback: Provide immediate feedback on pronunciation and clarity to help students refine their speaking skills.

Practice Mode: Offer a practice mode where users can record their recitations and compare their spoken text with the original content to assess their performance.

Progress Tracking: Implement a system to track improvements in speech accuracy and fluency over time, offering personalized recommendations for further practice.

Integration: This feature will be integrated seamlessly with existing functionalities, such as quizzes and flashcards, to offer a holistic learning experience that includes both written and spoken practice.

By incorporating audio-to-text functionalities, MY TUTOR will expand its capabilities and offer a more comprehensive tool for language learning and verbal practice. This enhancement will address additional aspects of self-learning, further supporting students in achieving their educational objectives.

Other Future Enhancements:

Enhanced AI Features: Explore the integration of more advanced AI algorithms to provide personalized learning paths and adaptive content based on user performance.

Mobile Application: Develop a mobile version of MY TUTOR to offer greater accessibility and convenience for users on the go.

Collaborative Learning Tools: Introduce features that enable collaboration and interaction among users, such as study groups or peer review systems.

These future enhancements aim to build upon MY TUTOR’s current success, ensuring that it continues to meet the evolving needs of self-learners and provides a state-of-the-arteducational experience

# APPENDIX I

# SOURCE CODE

**Backend.py**

# CHAPTER 7

# APPENDICES

from fastapi import FastAPI, File, UploadFile, HTTPException, Request, Form

from fastapi.responses import FileResponse

from fastapi.middleware.cors import CORSMiddleware

import os

from io import BytesIO

from mimetypes import guess\_type

from pymongo import MongoClient

from gridfs import GridFS

from motor.motor\_asyncio import AsyncIOMotorClient

from bson import ObjectId

from fastapi.responses import JSONResponse, StreamingResponse

import pypandoc

from docx import Document

from pypdf import PdfReader

from pptx import Presentation

from fpdf import FPDF

import random

import string

import json

from pymongo.errors import PyMongoError

# from moviepy.editor import VideoFileClip

# from quiz\_generator import export\_quiz

# from flash\_card\_generator import export\_flashcards

# from summary\_generator import export\_summary

from Summary.export\_summary import export\_summary

from Quiz.export\_quiz import export\_quiz

from FlashCards.export\_flashcards import export\_flashcards

from gemini import prompt\_everyting

# from speech\_to\_text import get\_audio

app = FastAPI()

app.add\_middleware(

    CORSMiddleware,

    allow\_origins=["\*"],

    allow\_credentials=True,

    allow\_methods=["\*"],

    allow\_headers=["\*"],

)

# Initialize the MongoDB client

client = MongoClient('mongodb://localhost:27017/')

db = client['MYTUTOR']

fs = GridFS(db)

metadata\_collection = db["metadata"]  # Collection for storing metadata

@app.post("/uploadtodb")

async def upload\_file(file: UploadFile = File(...), responseData: str = Form(...)):

    try:

        # Read the file contents

        contents = await file.read()

        # Parse the responseData from JSON string

        try:

            response\_data = json.loads(responseData)

        except json.JSONDecodeError:

            raise HTTPException(status\_code=400, detail="Invalid JSON for response data.")

        # Store the file in GridFS

        file\_id = fs.put(contents, filename=file.filename)

        # Store the response\_data in a separate MongoDB collection

        metadata\_collection.insert\_one({

            "file\_id": str(file\_id),

            "response\_data": response\_data

        })

        # Return the response with file info and response data

        return JSONResponse(content={"file\_id": str(file\_id), "filename": file.filename, "response\_data": response\_data})

    except PyMongoError as e:

        raise HTTPException(status\_code=500, detail=f"Database error: {str(e)}")

    except Exception as e:

        raise HTTPException(status\_code=500, detail=f"File upload failed: {str(e)}")

@app.get("/files/{file\_id}")

async def get\_file(file\_id: str):

    try:

        gridout = await fs.get(ObjectId(file\_id))

        headers = {

            'Content-Disposition': f'attachment; filename="{gridout.filename}"'

        }

        return StreamingResponse(gridout, headers=headers)

    except Exception as e:

        raise HTTPException(status\_code=404, detail="File not found")

@app.get("/recentfiles")

async def get\_recent\_files():

    try:

        files\_collection = db['fs.files']

        # Find the most recent files

        cursor = files\_collection.find({}, {"\_id": 1, "filename": 1}).sort("\_id", -1).limit(10)

        files = [file for file in cursor]

        # Fetch the file metadata

        file\_details = []

        for file in files:

            file\_id = file['\_id']

            file\_details.append({

                'id': f"{file\_id}",

                'filename': file['filename'],

                'file\_url': f"/view/{file\_id}"  # URL to view the file

            })

        return JSONResponse(content=file\_details, status\_code=200)

    except Exception as e:

        print(f"Error: {str(e)}")

        raise HTTPException(status\_code=500, detail="Internal Server Error")

@app.get("/view/{file\_id}")

async def view\_file(file\_id: str):

    try:

        file\_id = ObjectId(file\_id)

        grid\_out = fs.get(file\_id)

        # Get the file MIME type

        mime\_type, \_ = guess\_type(grid\_out.filename)

        if mime\_type is None:

            mime\_type = "application/octet-stream"  # Default MIME type

        # Stream the file to the client

        return StreamingResponse(BytesIO(grid\_out.read()), media\_type=mime\_type)

    except Exception as e:

        print(f"Error: {str(e)}")

        raise HTTPException(status\_code=404, detail="File not found")

@app.get("/filemetadata/{file\_id}")

async def get\_file\_metadata(file\_id: str):

    try:

        # Ensure that file\_id is valid

        if not ObjectId.is\_valid(file\_id):

            raise HTTPException(status\_code=400, detail="Invalid file ID format.")

        # Fetch file metadata from the database

        metadata = metadata\_collection.find\_one({"file\_id": file\_id})

        if metadata:

            return JSONResponse(content={

                "file\_id": metadata.get("file\_id"),

                "filename": metadata.get("filename"),

                "file\_url": metadata.get("file\_url"),

                "response\_data": metadata.get("response\_data", {})  # Ensure response\_data is included

            })

        else:

            raise HTTPException(status\_code=404, detail="File metadata not found.")

    except Exception as e:

        raise HTTPException(status\_code=500, detail=f"Error retrieving file metadata: {str(e)}")

def handle\_pdf(file\_path):

    reader = PdfReader(file\_path)

    number\_of\_pages = len(reader.pages)

    s = ""

    for i in range(number\_of\_pages):

        page = reader.pages[i]

        text = page.extract\_text()

        s += text

    return s

def handle\_txt(file\_path):

    with open(file\_path, "r") as f:

        s = f.read()

    return s

def handle\_docx(file\_path):

    d = Document(file\_path)

    s = ""

    for paragraph in d.paragraphs:

        s += paragraph.text + "\n"

    return s

# def handle\_mp3(file\_path):

#     s = get\_audio(file\_path)

#     return s

# def handle\_mp4(file\_path):

#     video = VideoFileClip(file\_path)

#     audio = video.audio

#     audio\_file\_path = "file.mp3"

#     audio.write\_audiofile(audio\_file\_path)

#     audio.close()

#     video.close()

#     return handle\_mp3(audio\_file\_path)

def handle\_pptx(file\_path):

    p = Presentation(file\_path)

    s = ""

    for slide in p.slides:

        for shape in slide.shapes:

            if hasattr(shape, "text"):

                s += shape.text + "\n"

    return s

@app.get("/")

async def hello\_world():

    return {"message": "Hello, World!"}

@app.post("/upload")

async def upload(file: UploadFile = File(...)):

    name = file.filename

    extension = name.split(".")[-1]

    file\_path = f"file.{extension}"

    with open(file\_path, "wb") as buffer:

        buffer.write(file.file.read())

    if extension == "pdf":

        s = handle\_pdf(file\_path)

    elif extension == "txt":

        s = handle\_txt(file\_path)

    elif extension == "docx":

        s = handle\_docx(file\_path)

    # elif extension == "mp3":

    #     s = handle\_mp3(file\_path)

    # elif extension == "mp4":

    #     s = handle\_mp4(file\_path)

    elif extension == "pptx":

        s = handle\_pptx(file\_path)

    else:

        raise HTTPException(status\_code=400, detail="Unsupported file type")

    response = prompt\_everyting(s)

    return response

@app.post("/export")

async def export(request: Request):

    req = await request.json()

    selected = req.get("selected")

    data = req.get("data")

    filename = ""

    if selected == 0:

        filename = "Summary.docx"

        export\_summary(data, filename)

    elif selected == 1:

        filename = "Flashcards.docx"

        export\_flashcards(data, filename)

    else:

        filename = "Quiz.docx"

        export\_quiz(data, filename)

    return FileResponse(path=filename, filename=filename, media\_type='application/vnd.openxmlformats-officedocument.wordprocessingml.document')

if \_\_name\_\_ == "\_\_main\_\_":

    import uvicorn

    uvicorn.run(app, host="0.0.0.0", port=5000)

# Home.jsx

import React, { useState } from 'react';

import { useNavigate } from 'react-router-dom';

import FileUpload from '../Frontend/FileUpload/FileUpload';

import work from './undraw\_work\_from\_anywhere\_re\_s2i6.svg';

import education from './undraw\_education\_f8ru.svg';

import SocialLinks from './SocialLinks';

import { useLocation } from 'react-router-dom';

import './home.css'; // Adjust the path if necessary

const Home = ({ setfunc }) => {

  const [isSidebarOpen, setSidebarOpen] = useState(false);

  const [isDarkTheme, setDarkTheme] = useState(false);

   const [hoveredIndex, setHoveredIndex] = useState(null);

  const navigate = useNavigate();

  const location = useLocation();

  const { decode,name } = location.state || {};

  const toggleSidebar = () => {

    setSidebarOpen(!isSidebarOpen);

  };

  const handleThemeToggle = () => {

    // console.log(signupData);

    setDarkTheme(!isDarkTheme);

    document.body.classList.toggle('dark-theme', !isDarkTheme);

  };

  return (

    <div className={`container ${isDarkTheme ? 'dark-theme' : ''}`}>

      <header className="header">

        <h1 className="logo">MY <span style={{"color":"#F6B17A"}}>TUTOR</span></h1>

        <button className="sidebar-toggle-btn" onClick={toggleSidebar}>

          ☰

        </button>

        <nav className={`nav ${isSidebarOpen ? 'sidebar active' : 'sidebar'}`}>

          <span className="close-btn" onClick={toggleSidebar}>×</span>

          <a href="#features" className="nav-link">Features</a>

          <a href="#testimonials" className="nav-link">Testimonials</a>

          <a href="#contact" className="nav-link">Contact</a>

          <button onClick={() => navigate("/recentfiles")} className="recent-files">Recent Files</button>

        </nav>

        <nav className="desktop-nav">

          <a href="#features" className="nav-link">Features</a>

          <a href="#testimonials" className="nav-link">Testimonials</a>

          <a href="#contact" className="nav-link">Contact</a>

          <button onClick={() => navigate("/recentfiles")} className="recent-files">Recent Files</button>

        </nav>

        <label className="switch">

          <input type="checkbox" className="input" onChange={handleThemeToggle} />

          <span className="slider"></span>

          <span className="moon"><svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 384 512"><path d="m223.5 32c-123.5 0-223.5 100.3-223.5 224s100 224 223.5 224c60.6 0 115.5-24.2 155.8-63.4 5-4.9 6.3-12.5 3.1-18.7s-10.1-9.7-17-8.5c-9.8 1.7-19.8 2.6-30.1 2.6-96.9 0-175.5-78.8-175.5-176 0-65.8 36-123.1 89.3-153.3 6.1-3.5 9.2-10.5 7.7-17.3s-7.3-11.9-14.3-12.5c-6.3-.5-12.6-.8-19-.8z"></path></svg></span>

          <span className="sun"><svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 24 24"><g fill="#ffd43b"><circle r="5" cy="12" cx="12"></circle><path d="m21 13h-1a1 1 0 0 1 0-2h1a1 1 0 0 1 0 2zm-17 0h-1a1 1 0 0 1 0-2h1a1 1 0 0 1 0 2zm13.66-5.66a1 1 0 0 1 -.66-.29 1 1 0 0 1 0-1.41l.71-.71a1 1 0 1 1 1.41 1.41l-.71.71a1 1 0 0 1 -.75.29zm-12.02 12.02a1 1 0 0 1 -.71-.29 1 1 0 0 1 0-1.41l.71-.66a1 1 0 0 1 1.41 1.41l-.71.71a1 1 0 0 1 -.7.24zm6.36-14.36a1 1 0 0 1 -1-1v-1a1 1 0 0 1 2 0v1a1 1 0 0 1 -1 1zm0 17a1 1 0 0 1 -1-1v-1a1 1 0 0 1 2 0v1a1 1 0 0 1 -1 1zm-5.66-14.66a1 1 0 0 1 -.7-.29l-.71-.71a1 1 0 0 1 1.41-1.41l.71.71a1 1 0 0 1 0 1.41 1 1 0 0 1 -.71.29zm12.02 12.02a1 1 0 0 1 -.7-.29l-.66-.71a1 1 0 0 1 1.36-1.36l.71.71a1 1 0 0 1 0 1.41 1 1 0 0 1 -.71.24z"></path></g></svg></span>

        </label>

      </header>

      <hr></hr>

      <section className="hero-section" id="home">

      <img src={education} alt="learning" className='page\_image'/>

      <div className='text-slide-in'>

        <h2 className="headline">

        Welcome to MY TUTOR <span style={{ color: "#E79E4F",fontWeight: "bolder" }}>{decode ? decode.name : name ? name : "User"}</span>, Your Ultimate Learning Companion

        </h2>

        <p className="sub-headline">

            Dive into a world of personalized learning with MY TUTOR, where advanced AI technology adapts to your unique study needs.

            {/\* Whether you're preparing for exams, mastering new skills, or seeking additional support, MY TUTOR is here to guide you

            every step of the way with tailored resources, interactive lessons, and real-time feedback. \*/}

        </p>

        {/\* {signupData && (

        <div>

          <h2>Welcome, {signupData.name}!</h2>

          <p>Username: {signupData.username}</p>

          <p>Email: {signupData.email}</p>

        </div>

      )} \*/}

        <FileUpload setfunc={setfunc} />

      </div>

      </section>

      <hr></hr>

      <section className="features-section" id="features">

        <div className='features-content'>

        <h3 className="features-title">Features</h3>

        <div className="features-list">

          <div className="feature-item">

            <h3>Smart Flashcards</h3>

            <p>Automatically generated flashcards from your study materials.</p>

            <button className="cta-button" onClick={() => navigate("/flashcards")}>Generate Flashcards</button>

          </div>

          <div className="feature-item">

            <h3>Personalized Quiz</h3>

            <p>Quizzes tailored to your learning progress and needs.</p>

            <button className="cta-button" onClick={() => navigate("/quiz")}>Take Quiz</button>

          </div>

          <div className="feature-item">

            <h3>Short Summarizing</h3>

            <p>Summarize your entire study material in brief within seconds.</p>

            <button className="cta-button" onClick={() => navigate("/summarization")}>Get Summarized</button>

          </div>

        </div>

        </div>

        <img src={work} alt="learning" className='page\_image'/>

      </section>

      <hr></hr>

      <section className="testimonials-section" id="testimonials">

        <h3 className="testimonials-title">What our users say</h3>

        <div className="testimonial">

          <p>"<span style={{"fontWeight":"bolder"}}>MYTUTOR</span> has revolutionized the way I study. Highly recommended!"</p>

          <p>- Rajinikanth</p>

        </div>

        <div className="testimonial">

          <p>"The personalized quizzes are a game changer. I feel more prepared than ever."</p>

          <p>- Virat Kohli</p>

        </div>

      </section>

      <hr></hr>

      <footer className="site-footer" id="contact">

            <div className="footer-container">

                <div className="footer-row">

                    <div className="footer-col">

                        <h6>About</h6>

                        <p className="text-justify">

                        MyTutor is an innovative self-learning app designed to empower individuals with flexible, on-demand educational resources. Tailored for students of all ages, it provides a wide range of interactive lessons, quizzes, and practice exercises across various subjects. Users can learn at their own pace, track their progress, and receive personalized recommendations based on their learning needs and goals. With a user-friendly interface and diverse content, MyTutor supports self-directed learning and helps users build knowledge and skills effectively. Whether for academic enhancement or personal growth, MyTutor makes self-paced education accessible and engaging.

                        </p>

                    </div>

                    <div className="footer-col">

                        <h6>Quick Links</h6>

                        <ul className="footer-links">

                            <li><a href="#home">Home</a></li>

                            <li><a href="#features">Features</a></li>

                            <li><a href="#testimonials">Testimonals</a></li>

                            <li><a href="#contact">Contact</a></li>

                        </ul>

                    </div>

                    <div className="footer-col">

                        <h6>Social Links</h6>

                        <SocialLinks/>

                    </div>

                </div>

                <hr />

            </div>

            <div className="footer-container">

                <div className="footer-row">

                    <div className="footer-col">

                        <p className="copyright-text">

                            Copyright &copy; 2024 All Rights Reserved by <a href="#" style={{"fontWeight":"800"}}>MYTUTOR</a>.

                        </p>

                    </div>

                </div>

            </div>

        </footer>

    </div>

  );

};

export default Home;

# APPENDIX II

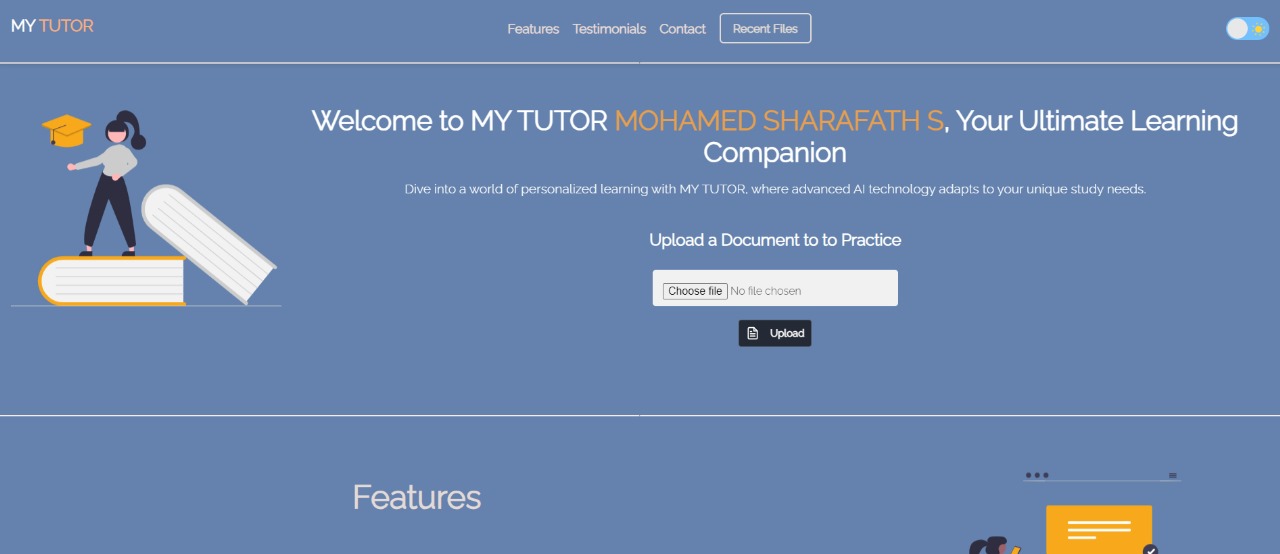
# SCREENSHOTS

****

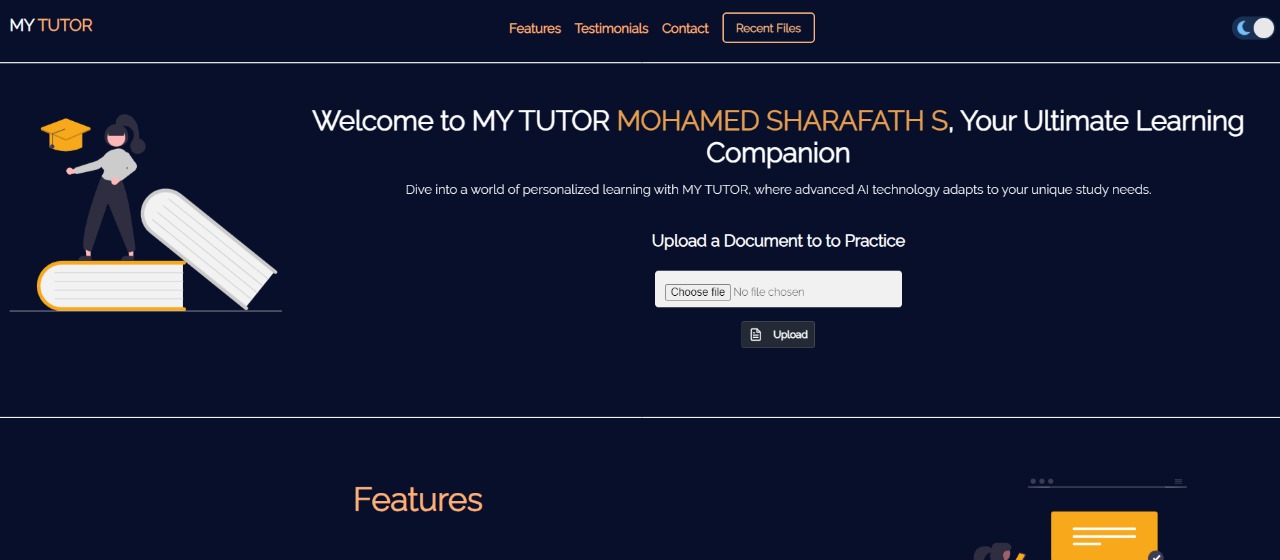
**Fig.A.2.1Signuppage**

****

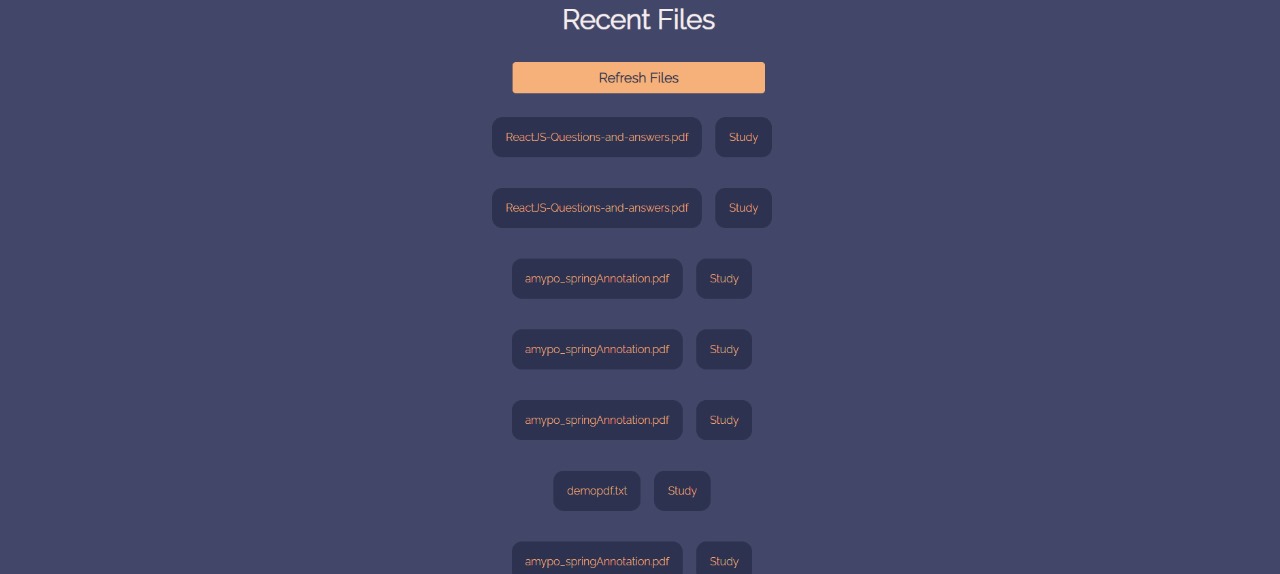
# Fig.A.2.2.Login



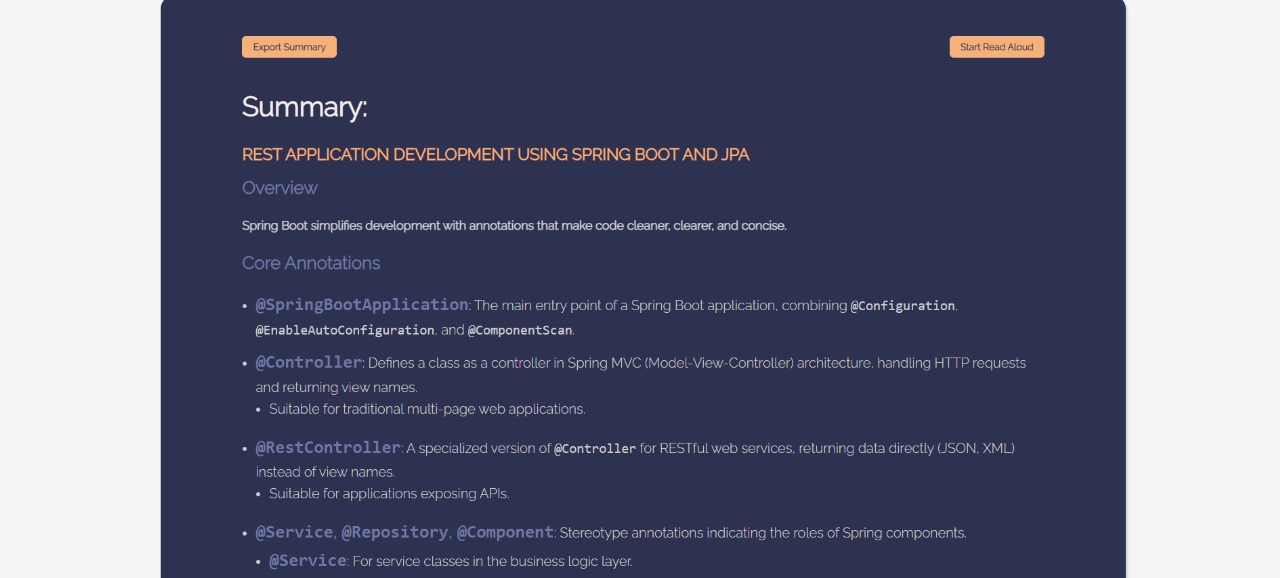
**Fig.A.2.3.Landing Page**

****

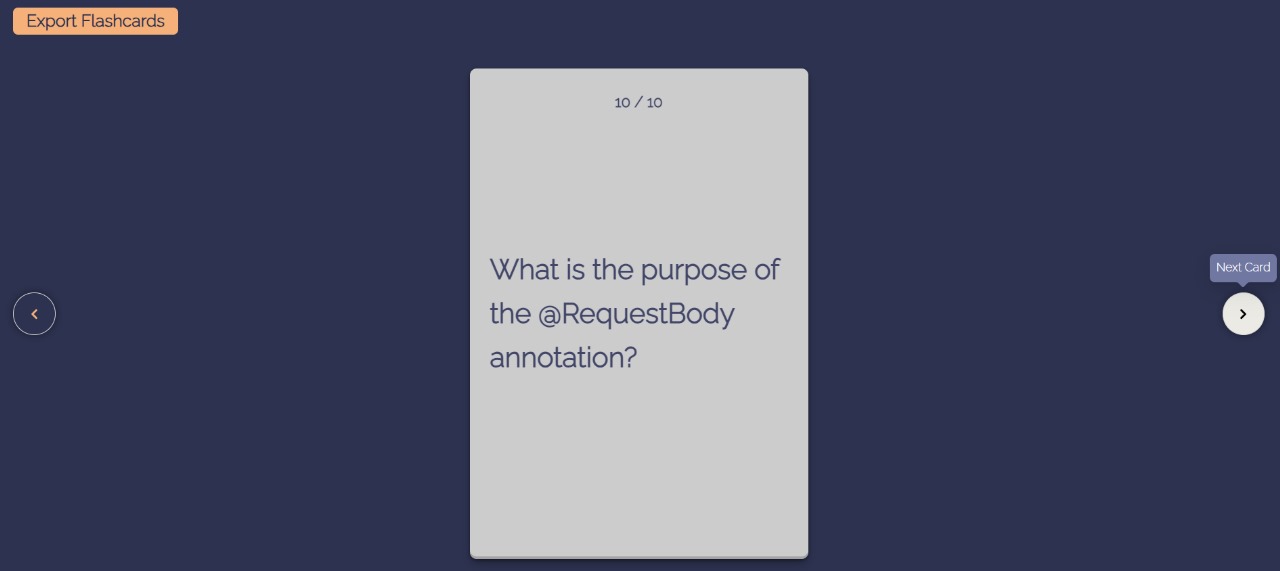
# Fig.A.2.4.Landing Page(Dark Theme)



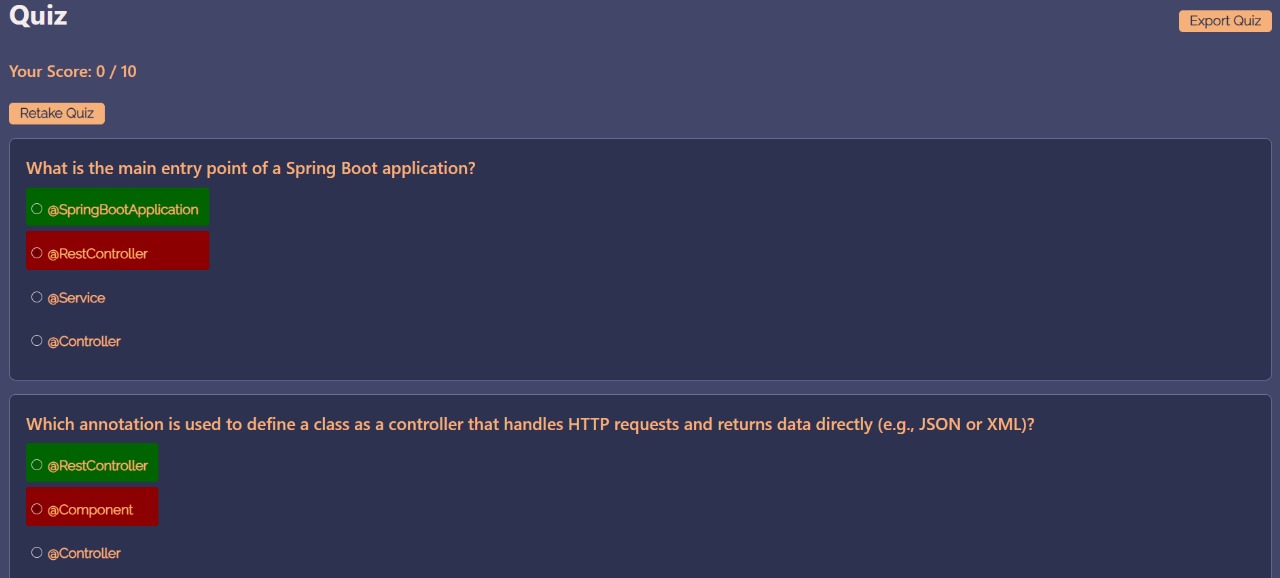
**Fig.A.2.5.RecentFiles**

****

# Fig.A.2.6.Summary



**Fig.A.2.7.Flashcards**

****

# Fig.A.2.8. Quiz



**Fig.A.2.9. Features Section**



# Fig.A.2.10.About section

# REFERENCES

**Webreferences:**

1. PostgreSQLOfficialDocumentation:<https://www.postgresql.org/docs/>
2. ReactOfficialDocumentation:<https://reactjs.org/docs/getting-started.html>
3. SwaggerOfficialDocumentation:<https://swagger.io/tools/swagger-ui/>
4. Java Extensions for Visual Studio Code Official Documentation:<https://code.visualstudio.com/docs/java/extensions>

# Bookreferences:

1. AmandaDavis(2022),“TheUltimateGuidetoHomeCareforSeniors:

EverythingYouNeedtoKnow”,SeniorCarePublications

1. EmilyWilson(2022),“TheCompleteGuidetoHomeCareforSeniors”,

ElderlyCareBooks

1. JenniferBrown(2021),“AginginPlace:A ComprehensiveGuidetoHomeCareServices”,Senior CarePublishers
2. JessicaAdams(2022), “HomeCareSolutionsfor Seniors:ACompleteHandbook”,Caregiver Press
3. KimberlyJohnson(2021),“HomeCareforSeniors:APracticalGuide”,

SeniorCarePublishers

1. LauraMiller(2021),“HomeCareServicesforSeniors:AComprehensiveManual”,CaregiverResources
2. RachelSmith(2021),“SeniorHomeCare:AStep-by-StepGuidetoCaringforYourLoved OneatHome”,Family CareGuides
3. SarahThompson(2022),“CaringforAgingParents:APracticalGuidetoHomeCareServices”,AgingWell Books