

AI-powered MCQ Generation

Vishnu Prasad C P

Roll No: 60

Reg.No: KTE23MCA-2060

Guided By: Dr. John C John

Department of Computer Applications

Rajiv Gandhi Institute Of Technology, Kottayam

10 October 2024

Contents

- Introduction
- Scope of the project
- Relevance of the project
- Requirement analysis
- Development methodology
- Design
- Project progress
- Project plan
- Conclusion
- Git history

Introduction

- Provide a free educational tool that enhances study experiences for students preparing for entrance exams.
- Designed to facilitate the creation and practice of (MCQs).
- Question setters can easily add questions including options and answers, or let AI generate them.
- It includes a feature for test takers to create personalized question banks.

Scope of the project

- Automation of MCQs, options and solutions generation using NLP, providing a valuable tool for both question setters and test takers.
- Allows test takers to solve MCQs, track their progress, and report errors.
- Moderators can correct questions and the options and add them to the global pool.
- The application also have the ability to categorize questions by subject.
- Allows test takers to generate personalized MCQs from their study materials and add these questions to their local pool.

Relevance of the project

- Free resource for students preparing for entrance exams.
- Provides AI-driven generation of MCQ options and solutions.
- Moderators are responsible for ensuring the authenticity and accuracy of questions and answers.
- The automated categorization of questions by subject.
- PDF content extraction and generate personalized MCQs.

Requirement Analysis

Existing System:

- Manual MCQ Creation.
- Manual Subject Categorization.
- No PDF Integration.
- Traditional Methods.

Proposed System:

- Automatically/Manually.
- Automatic Subject Categorization.
- PDF to MCQ Conversion.
- NLP techniques.

Requirement Analysis (cont'd)

Software Requirements:

- Python Flask for the backend.
- HTML, CSS, and Bootstrap for frontend.
- NLTK, spaCY, Transformers, PyPDF2
- MySQL for database management.
- Development in PyCharm.
- Version control with GitHub.

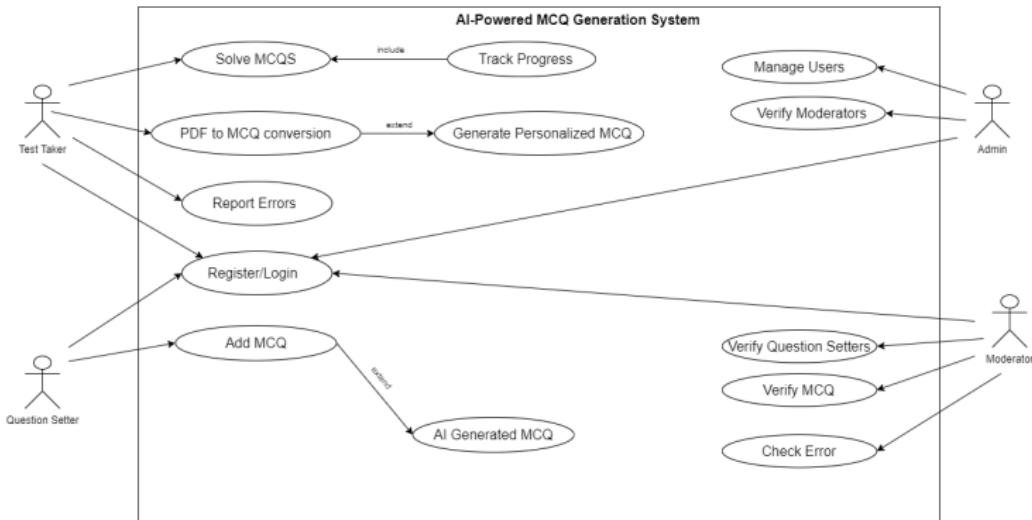
Hardware Requirements:

- High-performance PCs or laptops.
- At least 8GB of RAM.
- Multi-core processors.
- Sufficient storage.

Development Methodology

- This project follows agile methodology
- Authentications:
 - Moderator registration and login.
 - Question Setter registration and login.
 - Test takers registration and login.
- Application-specific:
 - Question setters can log in, and add MCQ questions including options and answers.
 - The same thing can be done by the AI models.
 - Moderators can log in, verify questions, options, and answers, and add them to the global pool.
 - Test takers can log in, solve MCQ questions, view solutions, and scores, and track progress. Additionally, they can input text in various formats including PDF to generate questions with the help of AI models and add these questions to their local pool.

Design (Usecase Diagram)



Design (WireFrame)

REGISTER

Name	Age
<input type="text"/>	<input type="text"/>
Email	Mobile
<input type="text"/>	<input type="text"/>
Role	Password
<input type="text"/>	<input type="text"/>
Subject	Qualification
<input type="text"/>	<input type="text"/>
University	Qualification Certificate
<input type="button" value="Register"/>	

Already have an account? [Login](#)

figure 1.1: Registration Page

LOGIN

Email
<input type="text"/>
Password
<input type="text"/>
<input type="button" value="Login"/>

figure 1.2: Login Page

Design (WireFrame)



figure 1.3: Moderator Home Page



figure 1.4: Question-Setter Home Page

Project progress

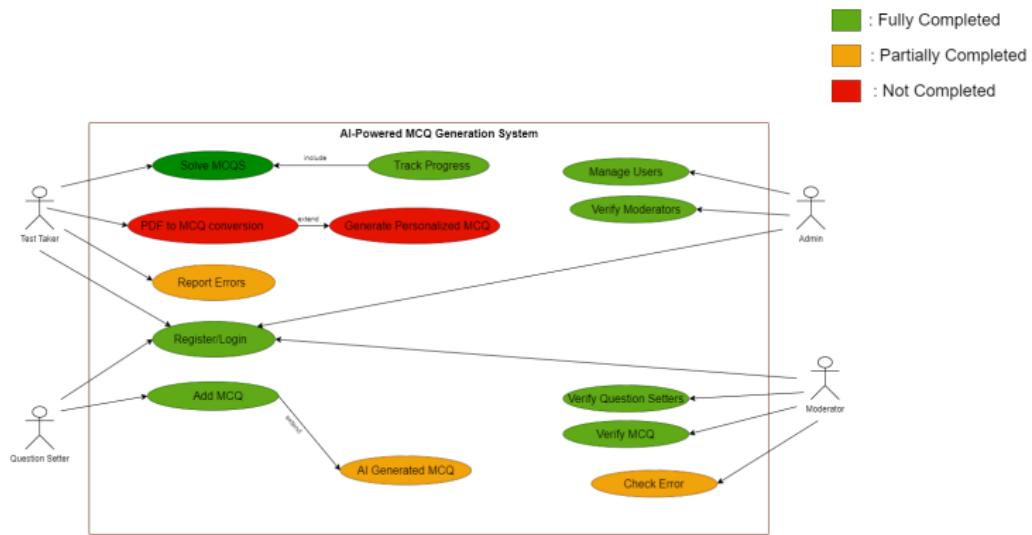
Current Status

Tasks	Status
Database design and Connection	Completed.
Authentication	Authentication for Admin, Moderator, Question Setter and Test Taker has completed.
Admin Module	Dashboard. Features like Verify Moderators and Manage all users are completed.
Moderator Module	Dashboard Created. Features like Verify question setters and Verify questions are added.
Question Setter Module	Dashboard Created. Features of manually adding MCQ has completed.
Test Taker Module	Dashboard Created.

Results

Tasks	Percentage
Register/Login	90%
Manage Users	100%
Verify Moderators and Question Setters	100%
Add MCQ	60%
Verify MCQ	90%
View MCQ	100%
Report Errors	30%
Solve MCQ	75%
pdf to mcq	30%
Total	77%

Work Progress



Pending Works

- Error reporting and Verifying Feature
- Automating MCQs options and solution generation.
- Solving MCQ and Tracking Progress.
- PDF content extraction and generate personalized MCQs.

Project Plan

Task	Status	Remarks
Analysing Problem	Completed	
Diagrams design	Completed	
Database and forms design	Completed	
Coding	In Progress	started by 05/08/2024
40 percentage of Project completion	Completed	
60 percentage of Project completion	Completed	
AI part	Yet to start	Planning to complete by 03/10/2024
100 percentage of Project completion	Yet to start	Planning to complete by 23/10/2024

Implementation

- **Frontend:** Built with HTML, CSS, and Bootstrap, featuring role-based dashboards for Admins, Moderators, Question Setters, and Test Takers.
- **Backend:** Developed using Python Flask for user authentication, question management, error reporting, and AI integration; MySQL is used for database management.
- **Database:** MySQL stores user data, MCQs, progress tracking, and error reports; SQL queries manage data retrieval and updates.
- **AI and NLP:** Integrated google.generativeai, NLTK, and spaCy for automated MCQ generation from text and PDFs using PyPDF2.
- **Testing:** Conducted using PyCharm for unit testing, with version control via GitHub to manage development progress and bug fixes.

Conclusion

- Offers students a personalized and dynamic study resource.
- Demonstrates AI's potential in education, benefiting various subjects and educational levels.
- Future plans include integrating an advanced image-to-text translation feature.

Git History

Commits on Sep 2, 2024	
manage user feature is added Vishnuprasadcp committed 1 hour ago	ba240c3 ⌂ ⌂
added moderator verification Vishnuprasadcp committed 1 hour ago	6af5c7d ⌂ ⌂
Commits on Sep 1, 2024	
added some moderator function Vishnuprasadcp committed 10 hours ago	ee1d250 ⌂ ⌂
added some moderator functions Vishnuprasadcp committed yesterday	890950a ⌂ ⌂
Commits on Aug 31, 2024	
third commit Vishnuprasadcp committed yesterday	c1d00f0 ⌂ ⌂
added admin functions Vishnuprasadcp committed yesterday	6432b3a ⌂ ⌂
Commits on Aug 16, 2024	
initial commit Vishnuprasadcp committed 2 weeks ago	d94aac6 ⌂ ⌂



References

- **Natural Language Toolkit (nltk)**, *Human language data toolkit*. [Online]. Available: <https://nltk.org/>
- **spaCy**, *Advanced NLP library*. [Online]. Available: <https://spacy.io/>
- **Hugging Face Transformers**, *State-of-the-art NLP models*. [Online]. Available: <https://huggingface.co/transformers/>
- **Python Flask**, *Lightweight web framework*. [Online]. Available: <https://flask.palletsprojects.com/>
- **MySQL**, *Relational database system*. [Online]. Available: <https://www.mysql.com/>
- **PyPDF2**, *PDF file manipulator*. [Online]. Available: <https://pypi.org/project/PyPDF2/>

Thank you!