

Customised

LEARNING ASSESSMENT

GENERATION

Project Client - Mr. Arjun Rajasekar (RCTS)

Team 20
Ankith Pai (2022113012)
Archisha Panda (2022111019)
Gargi Shroff (2022114009)
Prakhar Jain (2022115006)

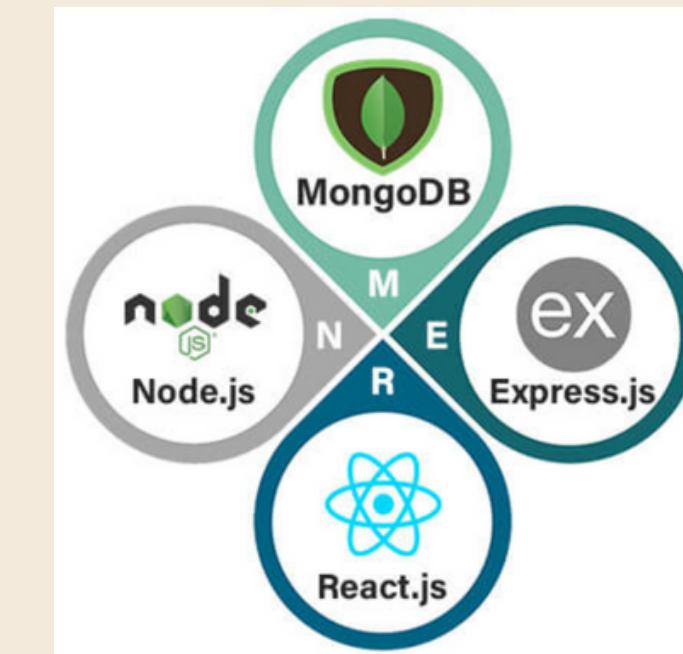
Design and Analysis of Software Systems

Problem Statement

- The new National Education Policy 2020 emphasizes a well-rounded education that goes beyond traditional textbooks and exams.
- It suggests teaching methods should consider students' backgrounds, interests, and practical application of knowledge, along with the subject matter. Assessments should test real-life application of concepts, not just memorization.
- However, creating such assessments manually takes a lot of time and effort for teachers. So, there's a need for automated software that can generate custom assessments tailored to each student's needs and abilities.

Approach Towards The Problem

- With the increasing need for individualistic attention to student learning, we make the use of advancements in technology to assist teachers in these tasks.
- We leverage Large Language Models (LLMs) alongside the MERN Stack to aid teachers in creating customized assessments tailored to judge comprehension levels of individual students.



Tech Stack Used

Front End Design

- React(antd)
- Javascript

Back End Design

- Python
- Flask
- MongoDB

Testing Methods

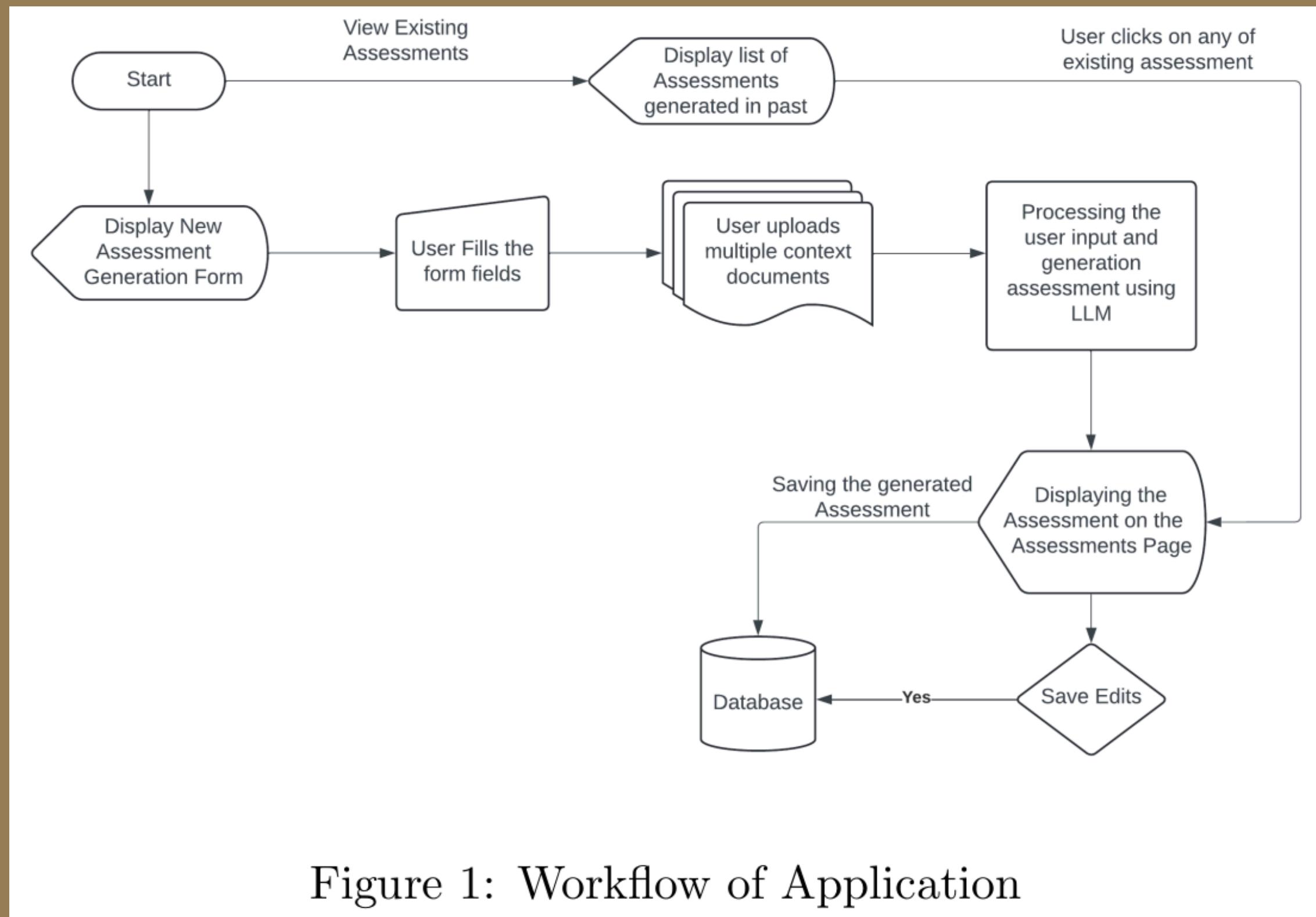
- PyTest
- Manual User Testing

Large Language Model

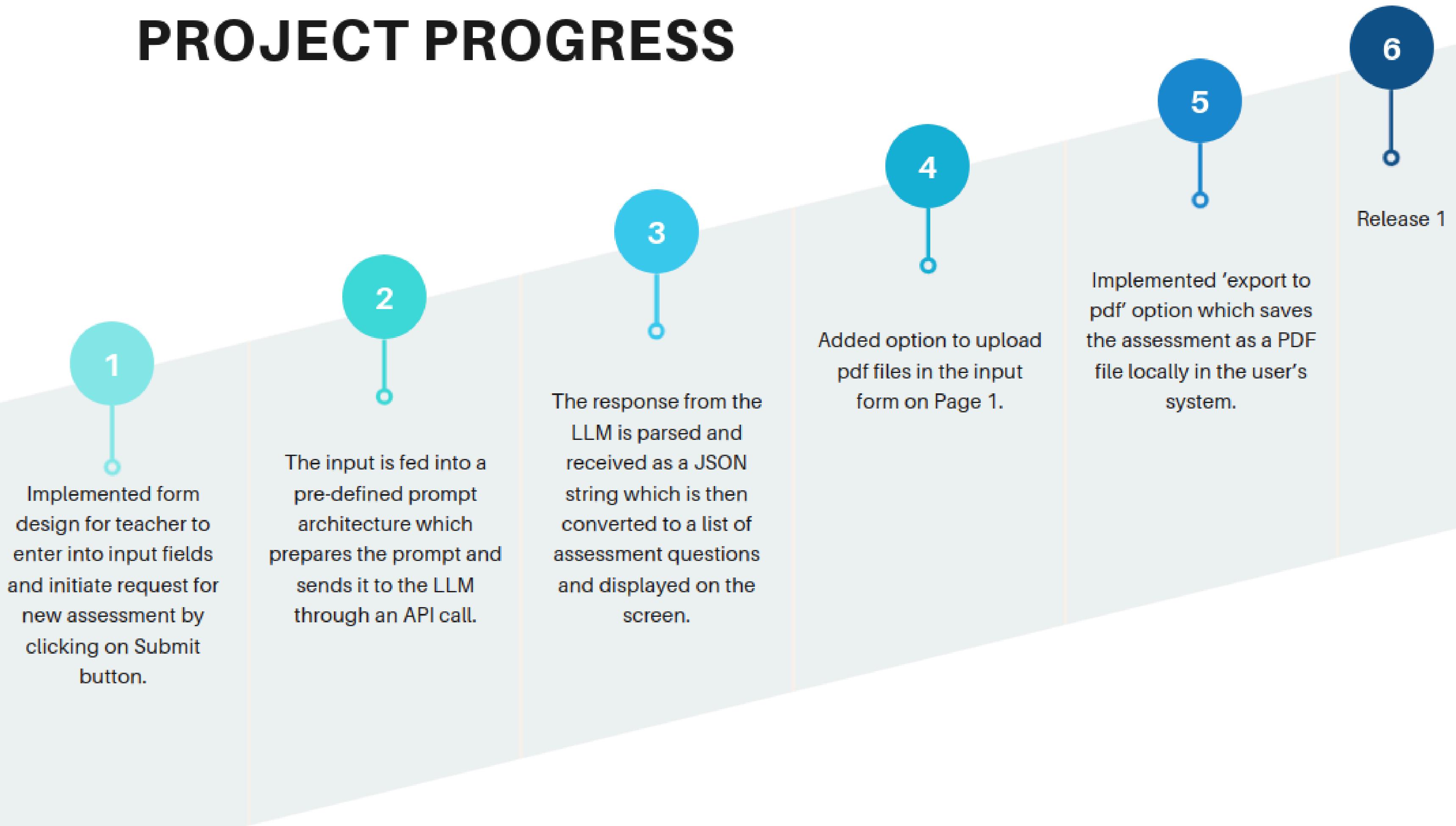
- LlaMA
- Hugging Face

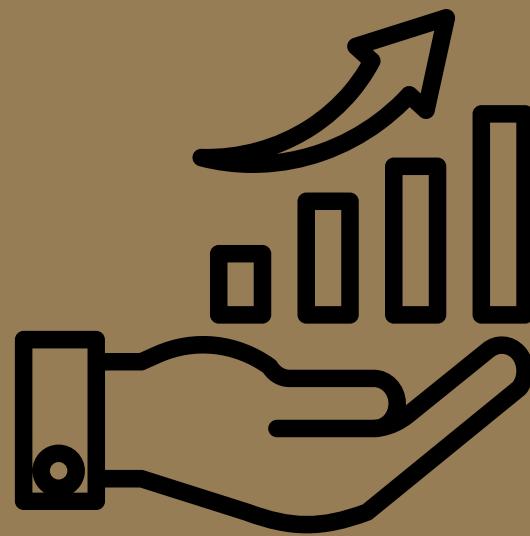
Quick Overview

Flow of Application For User



PROJECT PROGRESS





Release 2

Export as DOCX
Functionality

Explore LM
Studio for LLM
Related Issues

Parsing the
PDFs for
Assessment
Generation

Editing and
Save
Functionality
for Assessments
Page

Implementing
the History
Page with
Delete, Sort and
Search

Generate Assessment Page

Navigation Bar

Compulsory Input Fields

Optional Input Field

Submit Button

Option to upload multiple PDFs

Topic:

Type of Assessment:

Number of Questions:

Contextual Keywords:

Upload PDF:

Generate New Assessment View Existing Assessments

New Assessment

Enter the main topic or subject of the Assessment to be Generated. (NOTE : The character limit is 125)

Choose the type of questions required in the Assessment from the options provided.

Enter the number of questions in the Assessment (NOTE : The maximum limit is 20)

In this field you can enter the contextual keywords i.e., specific concepts which are to be integrated with the main topic for Assessment Generation

You can upload file(s) for providing the content for Assessment Generation

Upload

Submit

Assessments Page

Question Number

Editing Correct Answer

The below assessment has 1 mcqs question(s) on the topic 'World War 2'.
Context keywords: Economic Crisis.
No context PDF files were uploaded.

Assessment Questions

Question 1

What was the main cause of the economic crisis that contributed to the start of World War 2? [🔗](#)

Options

A The Great Depression [🔗](#)

B World War 1 Reparations [🔗](#)

C Stock Market Crash of 1930 [🔗](#)

D All of the above [🔗](#)

Summary of User Input

Edit the question

Editing individual options

Generate a pdf of the assessment

Generate a editable docx for the assignment

[↓ Export to PDF](#)

[↓ Export to DOCX](#)

[Save Assessment](#)

Save Assessment either as copy or overwrite existing version

History Page

User can sort the assessments according to date of creation

User can search for an assessment according to topic

Delete assessment

Clickable cards which redirect to Assessments Page

Choose number of records per page

List of Already Generated Assessments

Assessment Title	Last modified	Question Type
Assessment on World War 2	2024-04-22 02:51:23	mcqs
Assessment on psychology	2024-04-21 00:07:58	Long Answer
Assessment on Maths	2024-04-21 00:05:39	mcqs
Assessment on Industrial revolution	2024-04-18 10:13:26	long answer
Assessment on thermodynamics	2024-04-17 20:16:07	mcqs
Assessment on thermodynamics	2024-04-17 19:59:34	mcqs
Assessment on Software design	2024-04-17 17:33:37	mcqs
Assessment on Design and analysis of software systems	2024-04-17 15:14:37	mcqs
Assessment on quantum computing	2024-04-16 22:57:43	Long Answer
Assessment on quantum computing	2024-04-16 17:41:17	Short Answer

MongoDB Atlas Database

The screenshot shows the MongoDB Atlas Data Services interface. On the left, a sidebar lists various project components: Overview, DEPLOYMENT, Database (selected), Data Lake, SERVICES, Device Sync, Triggers, Data API, Data Federation, Atlas Search, Stream Processing, Migration, SECURITY, Quickstart, Backup, Database Access, Network Access, and Advanced. The main area displays the 'AssessmentsGenerated' database, specifically the 'assessments' collection. The top navigation bar includes 'Atlas', 'Archisha's O...', 'Access Manager', 'Billing', 'All Clusters', 'Get Help', and 'Archisha'. The 'Data Services' tab is active. A search bar at the top right says 'Search Namespaces'. Below it, a 'Find' tab is selected, followed by 'Indexes', 'Schema Anti-Patterns (0)', 'Aggregation', and 'Search Indexes'. An 'INSERT DOCUMENT' button is located in the top right of this section. A 'Filter' button and a query input field ('Type a query: { field: 'value' }') are also present. The results section shows two documents with their _id, user_input, questions, and last_modified fields. Navigation buttons for 'PREVIOUS' and 'NEXT' are at the bottom, along with a message icon with a '1' notification.

AssessmentsGenerated.assessments

STORAGE SIZE: 100KB LOGICAL DATA SIZE: 113.78KB TOTAL DOCUMENTS: 66 INDEXES TOTAL SIZE: 36KB

Find **Indexes** **Schema Anti-Patterns (0)** **Aggregation** **Search Indexes**

INSERT DOCUMENT

Filter Type a query: { field: 'value' } **Reset** **Apply** **Options ▾**

QUERY RESULTS: 1-20 OF MANY

```
_id: ObjectId('6618dfd3b9e15348567e5e50')
▶ user_input : Object
▶ questions : Array (2)
last_modified : "2024-04-12 12:46:35"
```

```
_id: ObjectId('661903884abcd5e248a15d1f')
▶ user_input : Object
▶ questions : Array (3)
last_modified : "2024-04-12 15:18:56"
```

PREVIOUS **NEXT**

1-20 of many results

Goto System Status: All Good

1

Testing

Test Planner and Tracker								
Test No. ID	Related Use case	Pre-conditions	Test Description (steps)	20 Archisha Panda, Ankith Pai, Prakhar Jain				
				Expected Outcome	R1 Outcome (color code cell background)	Comments (if test case failed)	R2 Outcome (color code cell background)	Comments (if test case failed)
10	Viewing Generated Assessment	1. User has submitted the form 2. User is on View Assessment page	User has submitted the form and has been redirected to View Assessment Page.	The generated assessment is displayed as list of questions. User is given an option to edit the output or simply export as pdf/docx	Red	Assessment is displayed. Edit option not implemented. Export pdf works.	Green	
11	Editing the Assessment Generated	1. User has submitted the form 2. User is on View Assessment page 3. Test 10 completed	The assessment generated is displayed. The user wishes to change a particular question. They click on the 'Edit Question/Option' button present near the question.	The user is prompted to enter the changes they wish to make. The change should reflect in the assessment displayed.	Red	Not implemented	Green	
12	Editing the Assessment Generated	1. User has submitted the form 2. Assessment has been generated. 3. User is on View Assessment Page	User clicks on the Save button and then clicks overwrite in the pop up.	The Assessment is overwritten and saved. No new entry is made in the history page.	Red	Not implemented	Green	
13	Editing the Assessment Generated	1. User has submitted the form 2. Assessment has been generated. 3. User is on View Assessment Page	User clicks on the Save button and then clicks save as copy in the pop up.	The Assessment is copied and saved as a new assessment. A new entry is visible in the history page.	Red	Not implemented	Green	
14	Viewing a List of previously generated assessments	1. User is on History page	User has navigated to History page through the navigation bar.	The page displays all the previously saved assessments with their name and time of generation.	Red	Not implemented	Green	

- For the front end, on recommendation of the client, the tests are manual and user tests.
- For the back end, testing is complete using PyTest. All the modules namely, assessment, exceptions, llm_interface and userinput have been unit tested.
- Integration testing has been done verifying module interaction and interface efficiency.

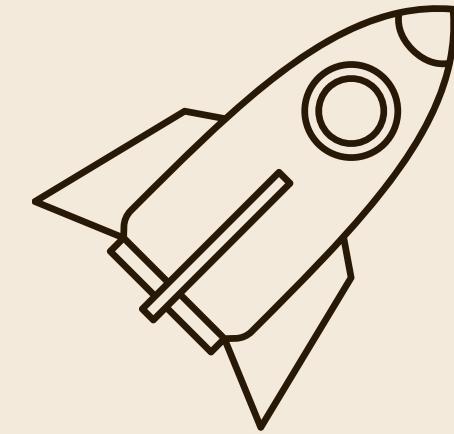
```
● ankith@AnkithLaptop:~/mount/gitstuff/iiith/dass-project-spring-2024-team-20/src$ py -m pytest
=====
test session starts =====
platform linux -- Python 3.11.6, pytest-7.4.0, pluggy-1.2.0
rootdir: /home/ankith/mount/gitstuff/iiith/dass-project-spring-2024-team-20/src
plugins: anyio-4.2.0
collected 29 items

tests/test_assessment.py .....
tests/test_exceptions.py .....
tests/test_llm_interface.py .
tests/test_userinput.py ...

[ 65%]
[ 86%]
[ 89%]
[100%]

===== 29 passed in 2.78s =====
```

DEMONSTRATION



[Link to the video](#)

Design Qualities Of Our System

● ABSTRACTION AND INFORMATION HIDING

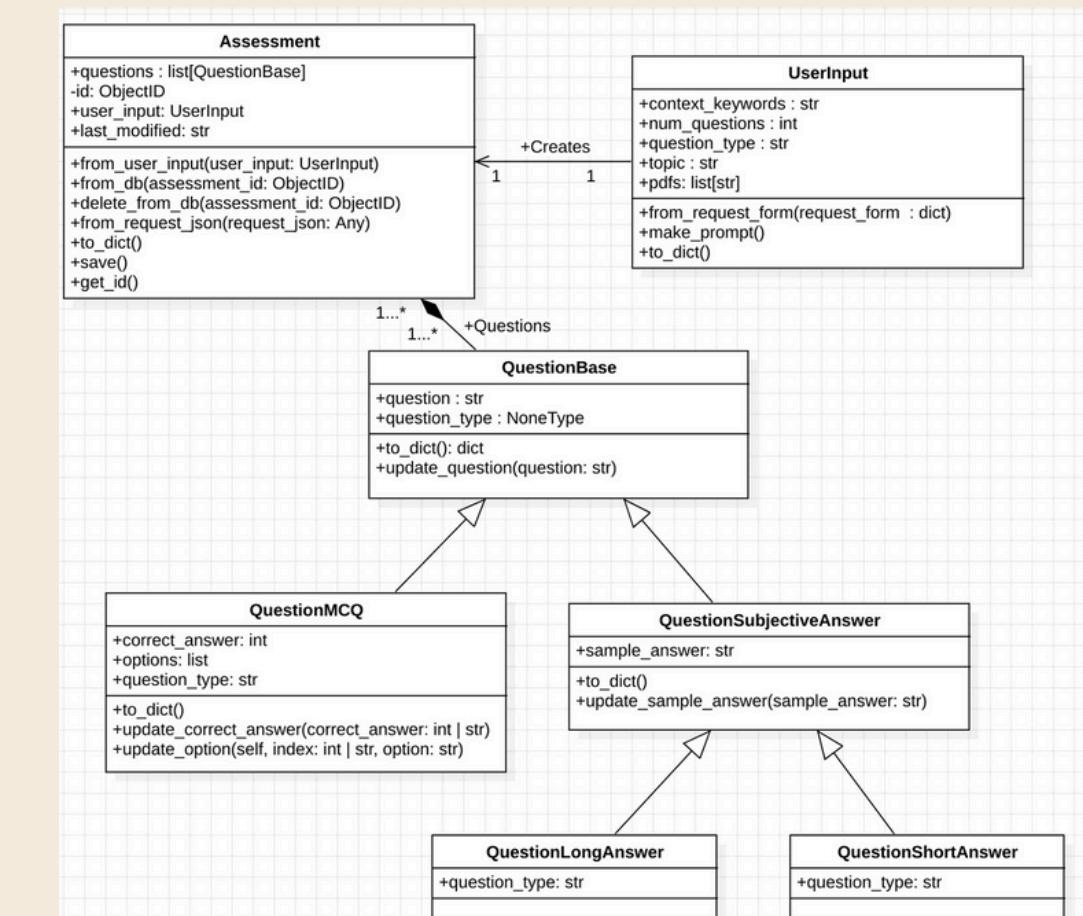
Our code provides a common controlled interface for all database related operations. In addition, data and procedure control is seen through for instance QuestionBase abstracting all single question operations in Assessments.

● SEPARATION OF CONCERNS

The three basic functionalities of our application have been split into several modules, each taking care of a single feature. As an example user input handled generating new assessment, assessments class oversaw editing functionalities, LLM interface dealt with API calls to LLM and so on.

● LOW COUPLING, HIGH COHESION

Methods modifying similar objects have been encapsulated into a single class. Only necessary modules have been associated together ensuring automatic transmission of any changes avoiding any inconsistencies.



Advantages Beyond Design...

Huge Scope for Extensibility

Simple, Self-Explanatory UI

Modifiable Access Control to MongoDB Database

Backend implemented on Flask and gunicorn

Low Demand Of Resources and Installations

Easily Adaptable To Change Of LLM/ Database



Learning Outcomes

- **Technical Skills Development:** MERN Stack, Flask development, LLM Interfacing, Collaboration on github
- **Project Management Skills:** Building on Project management methodologies like Agile and Scrum that includes frequent team meets and multiple iterations of waterfall model.
- **Domain Knowledge:** Gained insights into technological advancements in the Education domain
- **User Interface (UI) Design:** Understanding requirements and preferences to create intuitive and user-friendly interfaces.
- **Documentation Skills:** Writing clear and comprehensive documentation for the software to aid in its maintenance and future development.

Conclusion



The product will be integrated with the Customized Content Generation Application, ultimately leading to a complete system which given a topic and context will generate relevant content used for teaching along with assessments to test the learning of students.

Individual Contributions

Ankith Pai

- Worked on the design and implementation of various backend classes
- Worked on integration of the LLM into the backend
- Implemented question editability (frontend+backend)
- Implemented the two ways of save assessment
- Wrote some backend unit tests
- Worked on the documentation including setup guide

Archisha Panda

- Worked on the front end design for the web application.
- Parsing the output generated by the LLM and displaying it on the Assessment Display Page, Navigation Bar.
- Implementing the Export to PDF option on the Assessment Display Page.
- Implemented the History Page with API calls and storage in MongoDB backend
- Added pagination to History Page
- Worked on the documentation

Prakhar Jain

- Worked on JSON handling of LLM output in backend
- Worked on PDF upload handling
- Implemented DOCX export and improved PDF export
- Wrote some backend unit tests
- Worked on the documentation
- Researched on LM Studio

Gargi Shroff

- Worked on the front end design for the web application.
- Designing the Input Form and adding various constrains on the input fields, Navigation Bar.
- Displaying the summary of user input on the Assessment Display Page.
- Added Search and Sort functionality to History Page
- Researched on LM Studio
- Worked on the documentation



Thank You

Team 20

Design and Analysis of Software Systems