	MOBDEVE Major Course Output
	BSCS-ST^1, BSIT^2, BS<specialization>$^{\text{<superscript>}}$
	intelliTest
	Natividad, Josh Austin Mikhail | Razon, Luis Miguel Antonio

Description

intelliTest is an application that can auto-generate holistic mock exams that follow a structured format, such as DepEd's K-12 KPUP (Knowledge, Process, Understanding, Product/Performance) format, to prepare students for their actual exams. The application can take in the user's notes, lecture slides, readings, and other forms of content, which are then analyzed and processed by OpenAI's LLMs to generate the user's mock exam that follows the structured format. The LLM's output will be processed and parsed by the application to provide a user-friendly form layout for answering these mock exams.

When a student generates a mock exam, these would contain exam types such as simple types of Remembering and Understanding (Knowledge) of Multiple Choice, True or False, Matching Type, or Fill in the blanks, to complex types such as Problem Solving, Programming, Concept Mapping, and Essays.

Aside from generating mock exams, the app can also auto-grade these mock exams so that students can receive immediate feedback and learn from their mistakes. The simple exam types can be auto-graded due to their linearity, while the LLM can evaluate the complex exam types to provide feedback and grades using its own generated rubrics that the student can also use as a guide for answering the mock exam.

Services / APIs (are these valid)

Camera

- To take pictures of notes and convert them to text, handled by Google ML Kit on Android

Google ML Kit on Android

Text recognition v2

- To recognize and extract text from images using OCR to be formatted in a readable and understandable format by the OpenAI API

Web Server

- To handle communication between OpenAI API and the user's application
- To handle document and image parsing to be understandable by OpenAI's LLMs
- To communicate with the PocketBase database instance storing the following:
    - User Profile Information (email, grade level)
    - User's generated mock exams
    - User's exam results with corrections

OpenAI API (Inside the Web Server)

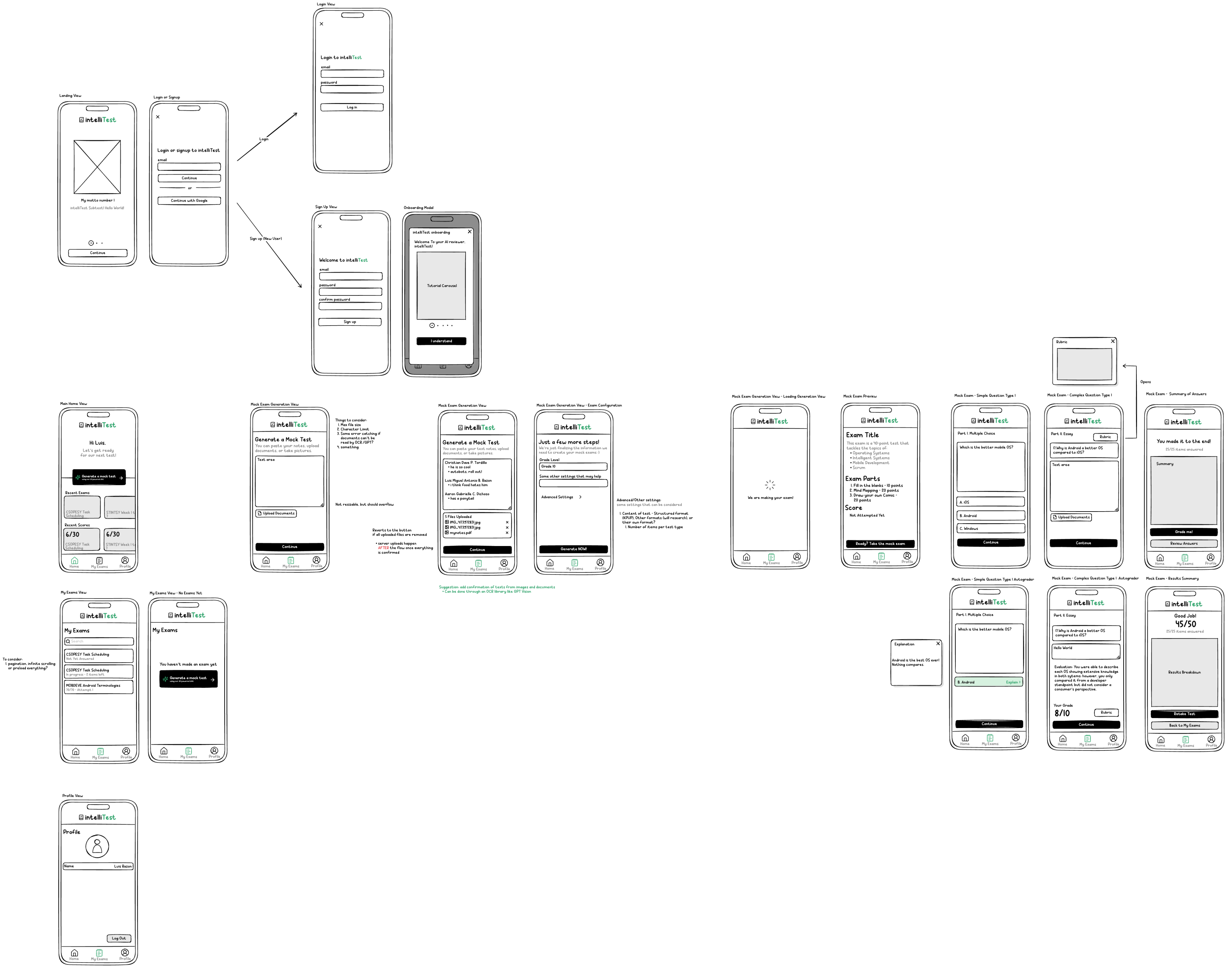
- Used to generate mock exams in a JSON format that will be parsed and sent to the Web Server, which in turn will send it to the appropriate user as a response.
- Used to handle the grading of complex exam types, given a rubric or answer key (similarity and contextual checking)
    - Provides feedback including the grading of the question, and comments based on the generated rubric/criteria.

Functions

Function	Description
Register	The user must first register through their Google account using Google Auth. The name and email will only be collected to serve as identification of the user.
Log-in	The user must log-in through Google before accessing their account and the rest of intelliTest's features. The same button for registering will automatically turn to the
Onboarding	A modal with a simple onboarding guide on how to use intelliTest will be shown to the users. This view can also be inherited by other types of modals such as app updates, announcements, and more.
View Main Screen (Home)	The user can view a bottom navigation menu that leads to the following pages: "Exam Generation", "My Exams", and "Profile/Settings". The page layout contains a Call to Action button to create a mock exam, a carousel containing cards listing the generated mock exams containing the status of the mock exam. The user can then click on these cards where it will lead to its appropriate exam screens, allowing the user to check their answers or answer the exam.
View "My Exams" Screen	The user can view the list of mock exams that they have generated. It will show the name of the mock test and the status of the exam. The statuses of each exam are as follows: 1. The exam has not yet ever been answered 2. The exam is in progress, which means that the user has not finished the exam yet or is in the process of retaking it. This will show how many items the user has left unanswered. 3. The exam has been completed by the user. This will show the score and the number of times the user has taken the exam. In the case that the user has not generated a mock exam yet, it will instead display a Call to Action button to create a mock exam.
View "Profile/Settings" Screen	The user can see the details of their profile, such as their name and profile picture associated with their Google Accounts. The user will also be able to log-out of their accounts from this page.
View "Exam Generation" Screen	The user can input their notes in this screen through a text box or file upload. The user would also be able to select some settings that would enable the app to generate a mock exam that is better suited for their needs. Once the user has prompted the app to generate the mock exam, it will then proceed to the "Answer Generated Exam" Screen.
View "Answer Generated Exam" Screen	The user can see the mock exam that was generated based on their inputs. The user would also be able to answer the mock exam in this page and submit it once they are done.
View "Check Answers" Screen	The user can see their answers and the score that the auto-grader gave. They would also see a tally of their total score over the maximum score that they could get. The user would also have the option to retake the exam in this screen.
Input Notes	The user would be able to input their notes in multiple formats:

	1. Entering text through the integrated text box 2. Document Files, such as PDFs, docx, and pptx files 3. Uploading Images of their notes
Generate Exam	Given the notes that the user has submitted, a mock exam will be generated and sent back to the user, ready to answer.
Answer Exam	The user is able to answer the generated mock exam through the app. They would be provided with the appropriate tools in order to answer the mock exam (e.g. Radio Buttons for Multiple Choice types and Text Box for Essay types).
Submit Exam	After the user is finished answering the mock exam, they would be able to submit it which will allow the app to send it to the server for auto-grading and evaluation.
Review Exam Scores	The user can view an un-editable version of their mock exam after auto-grading is complete where they can view their answers and its corresponding grades.
Auto-grading System	The app will auto-grade the answers of the user to the mock exam by comparing it with a generated answer key for simple exam type portions while utilizing OpenAI to auto-grade complex exam type portions that cannot be compared with an answer key.
Log-out	The app will allow the user to log-out of their accounts.

intelliTest – Low Fidelity Wireframes



Things to consider:
1. Max file size
2. Character Limit
3. Some error catching if documents can't be read by OCR/GPT?
4. something

Not resizable, but should overflow

Reverts to the button if all uploaded files are removed

- server uploads happen AFTER the flow once everything is confirmed

Suggestion: add confirmation of texts from images and documents
- Can be done through an OCR library like GPT Vision

Advanced/Other settings:
some settings that can be considered

1. Content of test - Structured format (KPUP), Other formats (will research), or their own format?
   1. Number of items per test type

To consider:
1. pagination, infinite scrolling or preload everything?