FlashMind

AI Powered Flashcards Generator

Application Development and Emerging Technologies (ADET)

Submitted by:

Villegas, Jesalle Hilary D.

BSIT 3-2

Submitted to:

Prof. Rosicar Escober

I.   INTRODUCTION

Project Overview

FlashMind is an online study app designed to help students and learners retain information more effectively. The app allows users to create and review flashcards with the help of artificial intelligence. By processing notes provided by the user, the AI generates flashcards in second that will help the user to effectively understand information.

Problem Statement

Many students struggle with memorization and efficient study habits. Traditional flashcards can be time-consuming to create, and users may not always know the best way to structure their learning. This app solves these problems by using AI to generate flashcards and help memorization, making learning more effective and less stressful.

II.   NOTABLE USEFULNESS

Key Features

- **Flashcard Creation:** Users can manually create flashcards with text.
- **AI-Generated Flashcards:** AI generates flashcard based on uploaded text or pdf file.
- **AI-Powered Summarization:** Extract key points from large text and create flashcards.
- **Voice Input & Text-to-Speech:** Users can speak to create flashcards, and AI can read flashcards aloud.

Value Proposition

FlashMind helps people by making studying faster, easier, and more efficient. Instead of spending hours making flashcards, users can focus on learning while AI handles the rest.

III. **INNOVATION**

Unique Aspects

- AI-powered flashcard generation saves time.

- Voice input makes flashcard creation easier.

- Text-to-speech helps auditory learners.

- Using React.js creates a responsive and visually appealing user interface that works on different devices.

- Using Node.js with Express.js seamlessly communicate the app and the server.

Emerging Technology Used

- Artificial Intelligence for generating and summarizing content.

- Speech Recognition for voice input.

- React.js for Frontend Development

- Node.js + Express.js for Backend Development

IV.   BENEFICIARIES

Target Audience

- Students of all ages

- Professionals preparing for exams or certifications

- Anyone who wants to improve memory and learning skills

Impact

- Saves time by automating flashcard creation.

- Improves memory with AI-powered study techniques.

- Makes learning more engaging with voice and text-to-speech features.

V.   TIMELINE

Project Phases

- **Phase 1:** Research & Planning (1 Week) – Define features and design app flow.

- **Phase 2:** Design & Prototyping (1 Week) – Create wireframes and initial layout.

- **Phase 3:** Backend Development (1 Week) – Set up APIs and data storage.

- **Phase 4:** Frontend Development (2 Weeks) – Implement UI and integrate APIs.

- **Phase 5:** Testing & Debugging (1 Week) – Identify and fix bugs, improve performance.

- **Phase 6:** Deployment & Submission (1 Week) – Final testing, documentation, and launch.

Milestones

- **Week 1:** Complete research and application requirements.

- **Week 2-3:** Finalize design and deliver prototype.

- **Week 4-5:** Core system features fully developed.

- **Week 6:** Finish Testing and polish the application.

- **Week 7:** Deploy FlashMind and submit documentation.

VI. BUDGET

Estimated Costs

- Frontend & Backend Development: Free (using React.js, Node.js, Express.js)

- AI Integration: Free (Cohere)

- Hosting: Free (Netlify or Render)

- Miscellaneous: None

Total Budget:

Estimated Total: PHP 0

VII. TECHNOLOGY STACK

Tools and Technologies

- Frontend: React.js

- Backend: Node.js + Express.js

- AI Tools: Cohere

- Speech Recognition & TTS: Google Cloud Speech-to-Text and Text-to-Speech

Justification

- React.js provides a fast and efficient user interface.

- Node.js + Express.js offer a strong backend for AI processing and API integration.

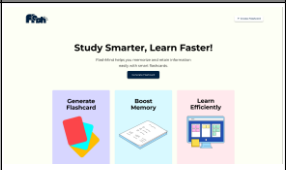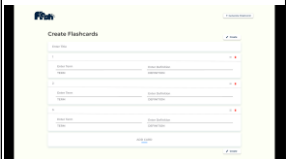- Cohere reduce costs while providing AI-powered features.

VIII. CONCLUSION

Summary

FlashMind AI is an AI-powered study tool that helps users create and review flashcards easily. By automating flashcard generation, the app makes learning more effective and stress-free. It is designed for students, professionals, and anyone looking to improve their memory and study habits.

Next Steps: Finish planning and start developing.

IX. REQUIREMENTS LIST

Project Name: FlashMind - Flashcard Generator App

Project Owner: Jesalle Hilary D. Villegas | 3-2

Comprehensive List of Requirements

	Requirement	Steps	Input Fields and Files	Output Display and Report	Acceptance Criteria	Remarks	UI/UX
1	Onboarding	1. Open FlashMind	None	**Output Display:** 1. Button to go to Generate Flashcard Page 2. Button to go to Create Flashcard Page **Output Report:** 1. Button to go to Create Flashcard Page 2. Button to go to Create Flashcard Page	1. When the "Generate Flashcards" button is clicked, the app should navigate to the Generate page. 2. When the "Create" page button is clicked, the app should navigate to the Create page.		

2	Create Flashcard	1. Open FlashMind
		2. Go to Flashcard Creation Page
			Input fields:
			1. Text Input
			  a.Term
			  b. Definition
				Output Display:
				1. Input fields for creating flashcard
				2. "Create" button
				3. Buttons to Move/Delete input
				4. Button to go to Generate Flashcard Page
				
				Output Report:
				1. "Create" button
				2. Option to Move/Delete input
				3. Button to go to Generate Flashcard Page
					1. When the "Generate Flashcards" button is clicked, the app should navigate to the Generate page.
					
					2. All input fields must be required.
					
					3. When the move icon is clicked, the user should be able to drag and reposition the flashcard.
					
					4. When the delete icon is clicked:
					-If more than 3 flashcards are present, a confirmation message should appear, and upon confirmation, the selected flashcard should be deleted.
					-If only 3 flashcards remain, display a message: "At least 3 flashcards are required."
					
					5.When the "Add Card" button is clicked, the app should validate that the existing flashcard fields are not empty. If valid, allow the addition of a new card.
					
					6.When the "Create" button is clicked:
					-Check if all required fields are filled.
					-If all fields are valid, proceed to the View page.
					-If any field is empty, display an appropriate error message		

3	Generate Flashcard	1. Open FlashMind 2. Go to AI-Generated Flashcard Page 3. Input text or upload document	Input fields: 1. Text/Voice Input 2. PDF Upload	**Output Display:** 1. Text box for notes input 2. Button to upload notes 3. Button for voice input. 4. Generate button to process inputs. 5. Button to go to Create Flashcard Page **Output Report:** 1. Button to upload notes 2. Button for voice input. 3. Generate button to process inputs. 4. Button to go to Create Flashcard Page	1. When the "Create" page button is clicked, the app should navigate to the Create page. 2. The text area should allow users to input or paste notes. Placeholder text should display: "Insert notes to generate flashcards…" 3. When the "Upload File" button is clicked, the user should be able to select a file from their device. Accepted formats must include .pdf, .txt, or other supported document types. The content of the uploaded file should populate the input area for review or immediate generation. 4. When the submit icon button (right corner of text area) is clicked: -If the input field is empty, display an error message prompting the user to enter content. -If content is present, the app should process the input and navigate to the Flashcard Viewer page with generated flashcards		
4	View Flashcard	1. Open FlashMind 2. Create or Generate Flashcard 3. View Flashcard	Input fields: 1. Text Input	**Output Display:** 1. Display Created or AI-Generated Flashcards (temporary session-based) 2. Display Term and Definition 3. Option to Edit Flashcards 4. Option to Text-to-Speech 5. Button for Prev and Next **Output Report:** 1. Display Term and Definition 2. Option to Edit Flashcards 3. Option to Text-to-Speech 4. Button for Prev and Next 3. Button to go to Generate Flashcard Page	1. When the "Generate Flashcards" button is clicked, the app should navigate to the Generate page. 2. The flashcard should display the term on the front and description on the back. There should be a Flashcard Title displayed above the flashcard 3. Users should be able to navigate between flashcards using the left (◄) and right (►) arrow buttons. The flashcard count (e.g., 1/5) should be shown below the flashcard. 4. Clicking the pencil icon should enable editing of the current flashcard's content. 5. Clicking the speaker icon should trigger a text-to-speech feature that reads aloud the flashcard content.		