

Brisk Recall

Capstone II: FIU Senior Design Project Showcase

Team Members:

Luis Delgado

Ana Oliveira

Gaeldesh Demosthene

Andres Hernandez

Meet The Team

Luis Delgado



Product Owner
Team Lead Developer

Ana Oliveira



Backend/Frontend
Developer

Gaeldesh Demosthene



Backend/Frontend
Developer

Andres Hernandez



Scrum Master
Backend/Frontend
Developer

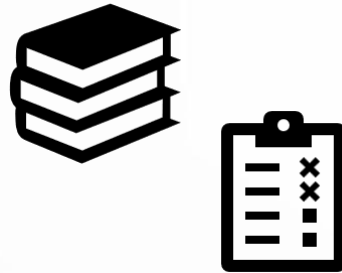


Seyedmasoud Sadjadi
Instructor

Purpose: What is BriskRecall?

Students need an easy way to condense study materials:

- Lecture Notes
- Practice Exams
- Study Guides



How to organize content into an easily accessible format without spending the time to create them from scratch?

BriskRecall is an AI powered tool that aims to aid students by creating flashcards from study materials.

It allows students to save time by doing the hard part, so that they can focus on studying and better prepare for exams.



System Design

Frontend

- ✓ Simple and Functional UI
- ✓ Vite (Tool)
- ✓ React
- ✓ HTML & CSS

Backend

- ✓ Node.js Server
- ✓ CommonJS
- ✓ Local API Endpoint
- ✓ Supabase Database

AI Integration

- ✓ Google Generative AI Library
- ✓ Google Gemini API

Implementation

UI Components

- ✓ Unique URLs
- ✓ Text Input Fields
- ✓ Interactive Buttons
- ✓ List of Components
- ✓ Navigation Bar

Data

- ✓ Login/Logout User Information:
 - ✓ Email Address
 - ✓ Password
- ✓ PDF File Upload
- ✓ Flashcards Content

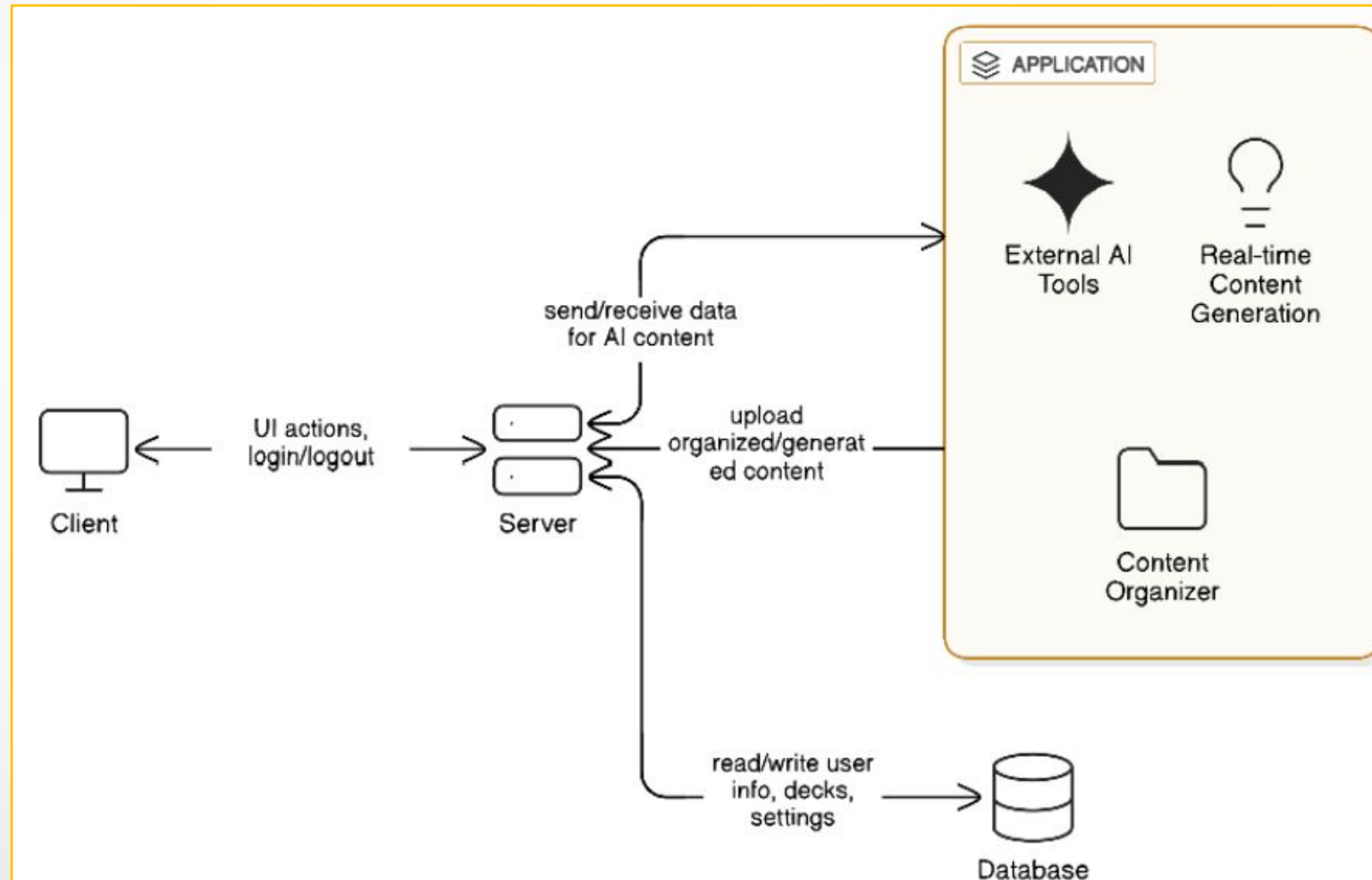
AI Integration

- ✓ Pdf2json Library Processes Data for the AI Model
- ✓ Gemini 2.5 Flash Model Instructed with Flashcards generation

Requirements

- ☐ Install Node.js and Git to clone repository.
 - ☐ Create an account in Supabase, where database is hosted.
 - ☐ Create the necessary tables and rules.
 - ☐ Retrieve Anon API Key and Project URL.
 - ☐ Create an account and generate a Google Generative AI Gemini API Key.
 - ☐ Setup the 2 .env files variables in frontend and backend.
 - ☐ Install dependencies and libraries needed in Frontend and Backend.
 - ☐ Run the project commands to start the frontend (npm run dev) and backend (node server.js)
-
- For a complete guide refer to [Brisk Recall Guide to Test](#).

Architecture Diagram



Summary

- **B**uilds and deploys a **full-stack flashcard learning platform** using **React (frontend), Node.js/Express (backend), and Supabase (PostgreSQL + Auth)** to help students efficiently retain and review key concepts through an interactive, structured system.
- **R**educes manual study preparation by integrating the **Gemini AI API** to analyze uploaded PDFs, extract core topics, and automatically generate structured **question–answer flashcards** from source material.
- **I**mplements a **relational database architecture (users → decks → flashcards)** with **cascading deletes, secure authentication, and real-time CRUD operations**, ensuring data integrity and user-specific customization.
- **S**trengthens learning efficiency through **dynamic UI logic** such as card-flip animations, circular navigation, shuffling, streak tracking, and instant feedback for active recall reinforcement.
- **K**eeps the application **scalable, modular, and maintainable** through clean component design, reusable state management, protected environment variables, and separation of frontend/backend responsibilities