Oasis Wellbeing

Type: Software Engineering

Project Description

Oasis is an interpersonal journaling application that uses the power of AI to link, connect, acquiesce, and remember your mental recollections and thoughts. Think of a guide that not only analyzes entries but provides a response that is much more palatable than Chat GPT.

Oasis also provides the ability to give the AI context by allowing users to upload historical journaling content. Leveraging the content, the AI can create a "history" with that user improving the responses given on future queries or journal entries.

For reference:

- Slideshow Pitch video
- Slideshow Pitch

Fall 2023 Project Goals

The Oasis project is ambitious and, as a result, the Spark! Team will be focused on creating an initial, working prototype which lays the groundwork for future features. In particular, the system will be focused on:

- User authentication and authorization
- User Historical upload
- Data storage implementation (relational, vector, file system)
- Main User Interface for the application
- About and Signup pages
- Adapt UX from prior work to be applicable to current system

Approach

- Identify an architecture for the main system
    - Typical web application architecture should be sufficient
    - User information / system information should leverage a typical relational database or firebase if using generic user information and authentication

- Identify LLM options
    - Review the available LLM options that allow for vertical training (wellness / spiritual guidance) and then per user training. Review options with Spark! Staff and Instructors
    - Based on LLM review, identify vector databases with good integration with the LLM options, understand the data storage / goal of the vector database, propose vector database architecture, review with Spark! Staff and instructors
    - Implement API access to proposed LLM
    - Implement vector database integration with LLM
    - Implement vector data loading from user upload
    - Implement administrative user interface to “prompt engineer” the LLM for our use case
- Adapt UX to new application approach
    - The requirements and goals of the system have changed since the UI/UX work was implemented. Identify what designs can be salvaged and what needs to be addressed. The UI/UX can be found on Figma.
- Identify storage & consumption options for user data upload
    - Identify file types / formats for data upload, near term and long term
    - Design solution for data storage: e.g. file system, blob, s3, etc
    - Implement data upload and ingestion which may include data transformation
    - See Technical Considerations
- Implement user authentication and authorization
- Implement new user signup page
- Implement main UI page for journaling
- Implement profile page for users (likely to include historical journal upload)
- Implement marketing page for site
- Implement about page

Technical Considerations

- The Web User Interface may not be the only interface over time. As a result, the system should have a clear boundary between the UI layer and the functionality layer (i.e. API)
- User data upload should be considered highly sensitive, consider encryption in transit and encryption at rest
- All pages in the site should indicate “For Demonstration Purposes Only”

Expected Final Deliverable

- Working website
- Signup and normal user access
- Administrative user access featuring prompt engineering
- User data upload and storage

Useful Reading

- WTF Is a Vector Database: A Beginner's Guide! - DEV Community
- Evolution of Vector DB
- Use Cases
- Understand Vector Database in 5 minutes
- Graph databasing vs Vector Databasing
- Top ten Vector DB's: Vector Databases ranked
- Islamicly AI, chat bot that can answer any question based on the Qur'an

Additional Information

Contact Information

Role	First Name	Last Name	Email
Client - Organization Name			
Spark! Support	Langdon	White	langd0n@bu.edu
Spark! Program Lead	Thao	Nguyen	tpnguyen@bu.edu
Spark! EIR			
Class Instructor	Nagendra Vashist	Mishr	nagendra@bu.edu
Class Instructor	Matthew	Zhang	matzh@bu.edu
PM	Abhishek	Malakar	amalakar@bu.edu
Team Lead [if applicable]			
Teammate	Erwin	Pimentel	erwinp@bu.edu
Teammate	Harshitha	Tumkur Kailasa Murthy	harshutk@bu.edu
Teammate	Vishwas	Bhaktavatsala	vishwasb@bu.edu

Meeting Notes

Meeting 1: Kickoff Meeting, Date - 4th October 2023

Attendance: Erwin, Harshitha, Vishwas

Minutes of the Meeting

Date: 4th October 2023
Time: 4:00pm - 5:00pm
Attendees: Abhishek Malakar, Erwin, Harshitha, Vishwas, Dakota Jackson, Langdon White, Ian Saucy

1. Introductions

All participants introduced themselves.

2. Overview of the Application

The app will function as a journal entry system with AI support.
The chat history will enable the bot to form a reflection of the individual user by analyzing typing style, speech patterns, traumas, etc.

3. Platform Discussion

Initial presentation showed a web application, but there were indications of a mobile implementation.
Clarification sought on the type of application. The consensus was for both mobile and web platforms, but with a primary focus on functionality.

4. LLm Implementation

Discussion on whether the project should be a full-stack implementation or if AI/ML would be the primary contribution.
Consideration of using existing LLM models. The team will research and select an appropriate LLM for the project and work on building the architecture.

5. Project Timeline

The Minimum Viable Product (MVP) will be a chatbot application that includes:
User login and account creation features.
Infrastructure around the application.
A system where one or multiple LLMs can be integrated.
Development from scratch to a fully functional website with a chat interface.

A feature for users to enter data, preferably in markdown or PDF format but currently limited to a file upload.

6. Data Input

Emphasis on file upload functionality, followed by conversion of the file to data.
Input via markdown or PDF is ideal.

7. Priorities

User profile creation and robust security architecture.
Signup and login page that redirects users to a dashboard upon successful login.
A cohesive UI system (Reference Rosebud UI).
A file upload system that lets users upload their thoughts.

8. Additional Features (Good to Haves)

The system should allow users to upload thoughts in various formats, including text, audio, and through Optical Character Recognition (OCR).
Integration of chatbot functionality with open-source platforms.
The user interface can be modeled similarly to the "Rosebud" interface.

Next Steps:
Begin research on the most suitable LLM for the project. (Future scope)
Design the initial layout for the user interface.
Determine the tools and technologies for the file upload system.

Next Meeting: TBD