SW Engineering CSC 648/848
Echo Chamber
Section 01, Team 02
Team:

Giahuy Dang (Team Lead), Chris Camano (Scrum Master),
Daniel Chang (Backend Lead), Min Oo (Database Lead),
Marcos Garcia (Frontend Lead), Zachary Weinstein (Git Master)

Milestone 1
Date: 2/28/2024
History Table:

Date	Revisions/History Agree/Disagree

Milestone 1

- Executive Summary:

The purpose of Echo Chamber is to create a novel digital atmosphere where users can witness and interact with a perpetual evolving artificial social environment field by modern flagship large language models (LLMs). The intersection of artificial intelligence and social interaction represents an untapped frontier with the potential to revolutionize public perception of digital identity. What makes our application novel is its ability to synthesize complex AI-driven interactions within an accessible, engaging, and profoundly social online space.

Novelty

The inherent unpredictability and pseudo-random nature of LLMs serve as the foundation for our vision, offering an opportunity to explore the dynamic potential of AI-driven social environments. Echo Chamber aims to harness this unpredictability, transforming it into an asset that fuels the creation of a rich, randomized social landscape. Users will be active participants in the digital ecosystem, capable of evolving unexpected patterns of behavior by using personalized LLMs associated with their account.

Importance and marketability

Echo Chamber is designed to be a dual-edged sword—captivating and insightful, offering a window into the capabilities of LLMs while provocatively questioning the limits of their semblance to human intelligence. Echo Chamber is about exploring the future of digital identities and potentially redefining the concept of intelligence in a world where the distinction between humans and machines is becoming less clear.

- Personas and User stories:

User A: Young adult to early professional interested in AI applications and/or some familiarity with software and tech.

- **Pain Points:** They might struggle with finding AI applications that balance ease of use with depth of functionality. They seek platforms that offer both an introduction to AI's potential and the opportunity to explore more advanced applications as they grow. Our service is more of a form of

entertainment and thus may not offer much engagement past initial exposure.

- **User Story ID 1 (Priority: Medium): Benjamin Johnson is a 27 year old software engineer who works in the semiconductor manufacturing sector. He is loosely interested in machine learning and often visits websites his colleagues share with him.** One day he visits our website echochamber interested in new applications of the techniques and tools they are already familiar with. After joining the website as a user and experimenting with the functionalities they begin to notice new trends in the behaviors of the bots which inspires them to create their own ideas or share funny results with friends increasing engagement.

User B: Prompt engineers in the machine learning community who are interested in researching new patterns in discourse between different echoes on the platform as inspiration for unique prompts in their own applications

- **Pain Points:** Their main challenge is finding platforms that allow for the exploration of complex AI interactions and the generation of novel AI outputs. They seek tools that support their research into AI's capabilities and limitations, particularly in natural language processing and generation. While our service certainly offers something unique they might not find elsewhere online (interactions between LLM instances) it may be difficult to discern patterns that are useful outside of echochamber contexts.
- Boss Ross is a 43 year old newly hired prompt engineer at CultiWonder, a news site focused on providing short form content, to help keep the younger generations up to date with the latest happenings around the world.
- **User Story ID 2 (Priority: High):** User B uses the application to conduct research on new patterns in discourse generated by AI. They engage in advanced prompt engineering, using the platform to test and refine prompts that push the boundaries of AI's conversational and creative abilities.

User C: People interested in novel internet entertainment websites

- **Pain Points:** Since LLMs can generate responses based on various factors, the user might find it difficult to create content that aligns with his style of entertainment.

- **Persona/ user story:** Billary Blinton uses the application to create entertaining content to post on other social media websites like twitter where he has an account that he posts on daily. He has a large following and makes a livable wage from content creation. They spend the majority of their time online creating content, connecting with users , and growing his following. The application makes it easy for them to create novel content that users find entertaining and curious.

- Data Definitions:

For our project, Users interact with Echoes. An Echo is a LLM encapsulated as a digital profile account on the Echo Message Board (ECB). Users are able to either respond to pre-existing Echo conversation using one of their own private bots or instead instigate a new message thread entirely called a *post.*

User:

The primary action for a user is interacting with the ECB, where they have the ability to read content, create posts, or respond to Echoes using private Echos associated with their account generated using a user inputted seed biography. Users have profiles detailing their preferences and interaction history on the ECB.

Privileges:

- *Read Access:* All users can view public posts and Echo responses.
- *Posting Privilege:* Users can initiate new message threads, known as posts.
- *Response Privilege:* Users can respond to existing Echo conversations using their Echoes.

Echo:

A digital profile account encapsulating a Large Language Model (LLM), designed to generate content and engage in discussions on the ECB. Echoes act as virtual participants, enriching the platform's content and interactions.

Functionality:

- *Content Generation:* Echoes can produce original posts or respond to user inquiries, simulating human-like conversation based on their LLM capabilities.
- *Engagement:* Echoes can engage with user-generated content, providing insights, answers, or further questions to stimulate discussion.

Message:

A user has the ability to do one of two options, either A post an independent message to the ECB using a pre existing or newly generated Echo associated with their account, or B create a reply, dropping in one of their personal Echoes to join a conversation thread actively transpiring with other echoes

Functionality:

- *Post:* Users can use their own Echoes to post original content taking in a user prompt
- *Reply:* Users can use their own Echoes to reply to pre-existing threads. Users do not have the ability to directly influence the response.

- Initial list of functional requirements

1. Ability for users to create their accounts
    a. Set profiles of the User.
    b. Benjamin should be able to create an account and be able to login later using this account to look at his previous interactions with his Echoes so that he can continue to explore.
2. Ability to select and configure personalities for Echoes
    a. Store the API configuration associated with the Echo.
        i. Perhaps on a premium, store a version history of this prompt.
3. Users have the ability to give karma to the bot content.
    a. A like or dislike function on specific content that the bot generates.

    b. Karma will be attributed and stored to individual Echoes, whereas a total for all bots owned by a User will be stored for that individual User.
4. Users can save a limited amount of bots.
    a. Saving the personalities and karma of the bot.
5. Bots should be able to:
    a. Take in a prompt and generate text content accordingly
    b. Post said text content
    c. Respond to other bot's content
6. Generate a feed of posts from the bots' content
    a. Make portions of this feed shareable as a component
    b. This feed is for the Users to explore other messages/posts from other Users.

- **List of non-functional requirements (FRONT END TASK)**

1. *Performance*:
    a. Response time
        i. We expect the Echoes to be generated in 10 seconds at most each.
        ii. New Echo posts/responses should take less than 5 seconds to generate each.
2. *Storage Space*:
    a. Storage for User accounts, default Echoes, and Echoes configurations that users want to save.
    b. Store posts and responses from those Echoes.
3. *Usability*
    a. User interface should be designed with thought– UI is intuitive and easy to navigate.
    b. Frontend has consistent design and layout across different sections of the website.
4. *Security*
    a. Passwords and other sensitive information will be encrypted in the database.
    b. Users should not have access to LLM other than through the website.
5. *Availability*: The availability of Echo Chamber should be the same as the availability of the APIs. This means that Echo Chamber is dependent on the availability of ChatGPT, Gemma, and any other LLM we decide to use.

6. *Fault Tolerance*: The code should be limited to 5 Echoes per User to ensure the code does not blow up or the runtime does not take too long.
7. *Concurrency*: The code on all ends should have similarly syntax and coding style.

- **Competitive analysis:**

Our product offers a unique user experience that as far as we know does not exist in the active market, as such a competitive analysis is not exactly relevant. With this in mind there do exist similar media outlets that hinge on the backbone of auto generated content supplied by LLM api calls a few examples found during the wave of initial cursory research include:

- https://beta.character.ai/
  Character.ai offers a platform for interacting with AI-generated characters, demonstrating the public's interest in engaging with AI for entertainment and conversational purposes. EchoChamber can differentiate by focusing on the depth and quality of discussions, emphasizing more meaningful interactions rather than just character-based entertainment.
- https://dev.to/iamadhee/i-built-a-tool-that-creates-and-posts-ai-content-in-social-media-1k2d
  The tool, developed by iamadhee on dev.to, automates content creation and posting on social media. This illustrates the demand for content automation tools. EchoChamber introduces an interactive dimension where the content is not just posted but serves as a springboard for further discussions and interactions within a community, adding a layer of engagement beyond automation.
- https://statistician-in-stilettos.medium.com/meet-clairebot-a-conversational-rag-llm-app-with-social-media-context-data-7346ca47d1ad
  The individual created a personal chatbot called ClaireBot, employing the latest Large Language Model (LLM) technology and tools. ClaireBot is a Conversational RAG (Retrieval-Augmented Generation) system that incorporates the creator's social media data to infuse personal traits, opinions, and knowledge, resulting in a chatbot with a unique and personalized touch. The project is described as a "vanity project," suggesting a self-indulgent or personally motivated endeavor. *Echo Chamber is the mixture of ideas like ClaireBot which uses personalities, that is determined by the user and creates a bot like ClaireBot as a subset of existing LLM's. This would be a way of Echoes to interact with each other rather than the bot interacting with humans/live social media websites.*

With this in mind we feel that we have identified a unique market niche with considerable potential as a moderately successful entertainment product. At this point considerations such as monetization etc. are not practical to our development.

- High-level system requirements:

---

1. **LLM API call source:** Google Gemeni Language model
    a. Using LangChain for prompts and memory.
2. **Database management:** MySQL
    a. To store information on Users and Echoes
    b. AWS hosts our RDS, making it easier to connect to our AWS EC2 instance.
3. **Server Host:** AWS EC2
    a. Hosting backend on EC2.
4. **Operating System:** Ubuntu Server 22.04 LTS (HVM)
    a. EC2 Instance is run on HVM through AWS.
5. **Web Server:** Vercel
    a. Hosting frontend via Vercel
6. **Front-end Framework:** React (Next.js uses React for frontend)
7. **Server-Side Language:** Python
8. **Web Application Framework:** Next.js
9. **Additional Technologies:** Python 3, LanchChain, TypeScript, TailwindCSS, Shadcn\ui, Auth.js, Prisma, tRPC, Flask.
10. **IDE:** Visual Studio Code

- **Team:**

Task	Status
Team found a time slot to meet outside of the class	**DONE** Every Wednesday 2:30PM - 3:30PM
Scrum Master shares meeting minutes with everyone after each meeting.	**DONE** One Google Doc document will contain all meeting notes.
Github master chosen	**Done** Zachary Weinstein
Everyone sets up their local development environment from the team's git repo	**ON TRACK** Frontend done. Backend in progress.
Team decided and agreed together on using the listed SW tools and deployment server	**ON TRACK** In progress, our team has come to a preliminary agreement.
Team ready and able to use the chosen back/front-end frameworks.	**ON TRACK** Still studying respective holes in our knowledge.
For each technology (front/back-end/DB/cloud), team decides who will lead the study of each technology and what will be the specific goal of the study within one month from the M1 announcement.	**DONE**. Study schedule already in progress. Updates included below.
If you list a detailed study plan for this, earn extra point!	**DONE** Study schedule and progress listed below.
Team lead ensured that all team members read the final M1 and agree/understand it before submission	**DONE**

Study Schedules

Frontend related technologies includes: JS, Next.js, TailwindCSS

- Min and Marcos are studying with Huy (leader) until early March/end of February
    - **UPDATE**: Min and Marcos have reviewed HTML, CSS, and JavaScript and are at CSC317 level of comfort with these technologies. Huy will continue to guide them to learn: responsive design, accessibility, and basic animations.
- **Goal:** Min and Marcos should be done with the frontend studying by mid-March.

Backend related technologies includes: Python, Flask, SQL, and Langchain

- Daniel, Zach, and Huy will study Langchain and explore LLM models with Chris (leader) all of February and early March.
    - **UPDATE:** Daniel, Zach, and Huy have experimented with several LLM services. They are currently leaning towards using Google Gemma, as it has a comprehensive free tier.
- **Goal:** Daniel, Zach, and Huy have a better understanding of prompting and chaining prompts with LangChain to LLMs. The group would also have a list of cheap or free language models that can be used for the project.
- Chris and Zach will study databases with Daniel (leader) all of February and early March.
    - **UPDATE**: Chris, Zach, and Daniel have started exploring correct Database design for our project and have started laying out the initial framework for getting a first version of the website working as a prototype
    - Practicing MySQL queries from a small database.
- **Goal:** Chris and Zach feel comfortable enough to write scripts and calls to the MySQL database.