GenAI Python Project

June 25th 2023

**Overall Objective Description:** Create a web-interfacing application that connects to a custom LLM model and allows users to input new truth data to improve quality of output.

**Budget:** XXX USD

**Deliverable timeframe:** 2 weeks

**Features to be developed:**

1. **Phase I**
   1. **Web interface to interact with GenerativeAI**
      1. Brief Description:
         1. Super simplified, production-ready web application that interfaces between user and LLM application. Preferably built in Python but open to hearing alternative choices. Need something up and running very quickly.
      2. How it works:
         1. User types question to the LLM application.
         2. The LLM application will output a result and list the documents which it referenced for this answer.
         3. Sources are a mix of PDF names or URLs.
         4. The web app should work on top windows and Mac browsers and be mobile-friendly, but no need for an app.
         5. Want to know the difference in cost, pros/cons of a small bare bones sight vs. one that could scale easier.
         6. Website needs an email, first and last name and username before using the tool.
         7. Needs to collect credit card information before proceeding.
         8. I need to be trained on how to connect the GenerativeAI to the website.
   2. **Enhanced Supervised Learning Algorithm**
      1. Brief Description:
         1. The user(s) supervise the algorithm. With the click of a button one can challenge the answer and can upload a PDF or URL.
      2. How it works:
         1. When the user prompts the Bot, the Bot returns a result AND four sources that support its response.
         2. The user sees what the four sources are and can click on a button to challenge a single source. That source is flagged and the user uploads either a new PDF or a new URL. This new source is now added to the VectorDB.
         3. Whenever a user inputs a similar prompt, five sources will appear. Of the 5, one will be the flagged source and another will be the user provided source. This new user now that has the option to select which of the two sources is MOST accurate, but not knowing which is flagged and which is not.
         4. This process happens three times until one of the sources (new vs. flagged) is chosen three times. Does not have to be consecutive and no time limit on when the two sources are available.
         5. IF the new source is chosen three times by other users, then that user is sent a message thanking them for their contribution and is awarded extra tokens to interact with GenerativeAI. All others who contributed to verifying the source, regardless of which option they picked, also get awarded extra tokens but fewer than the user who suggested the new source originally.
         6. This old source is now deleted from the VectorDB and the new source stays.
   3. **Data Scrapper that keeps information up-to-date.**
      1. Brief Description:
         1. Web scrapper that continuously monitors the web for added information
      2. How it works: Maybe I just scrape data and add to VectorDB. Not sure I need a scrapper or can simply build my own.
   4. **Prompt support (not included)**
      1. Brief Description:
         1. Supports user with more specific prompts. Learns from other prompts or ask key questions.
         2. Should be generated by a human SME and categorized/vectorized
2. **Phase II**
   1. TBD

Contractors:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Contractor | Country | Approximate hourly rate | ML/LLM Experience? | Backend/Frontend, Python? | Sophisticated Algorithms? | Other Notes |
| Insoftex | Ukranian |  | Yes | Yes |  |  |
| Serokell | Estonia |  | Yes | No? |  | MVP |
| Geek Solutions | Estonia |  | Yes, but no mention of LLM | Yes |  | Idea validation submission |