ITC 6040 Informatics Capstone

Professor: Xiaomu Zhou

Signature Assignment 1: Final Project Report

Sponsors: Dr. Carl Zangerl, Dr. Youngbok Ryu

Members: Ananya Kakani
Chen Yang
Jhanvi Patel
Sotheavatey Bong
Titiksha Ghosh
Vivek Vaghasiya

Date: March 26th, 2024

Project Statement

In collaboration with Northeastern University's Lab for Inclusive Entrepreneurship (NL4IE), our project focuses on developing a GenAI Chatbot specifically designed to serve small and diverse business owners. Navigating the procurement process can be challenging and confusing for small and minority-owned businesses often due to limited access to information and a lack of awareness of bidding/contracting opportunities. With NL4IE's sponsorship, our initiative champions economic inclusivity, particularly benefiting small minority businesses and disadvantaged entrepreneurs. Our overarching goal is to simplify the procurement process for business owners seeking to become vendors for higher education institutions by employing advanced technology within our chatbot. Through transparency, accessibility, and collaboration, we aim to drive positive change and foster equitable opportunities within the academic procurement landscape.

Sponsor's Background Information

Our project receives sponsorship from Northeastern University's Lab for Inclusive Entrepreneurship (NL4IE), with support and strategic guidance from Dr. Carl Zangerl and Dr. Youngbok Ryu (Zangerl, 2024; NL4IE's Website, 2024). Dr. Zangerl serves as NL4IE's Communication Lead, while Dr. Ryu oversees research and initiatives related to supplier diversity.

NL4IE, located at Northeastern University, holds prestigious recognition as a University Center by the U.S. Economic Development Administration, affirming its role in advancing inclusive entrepreneurial practices (NL4IE's Website, 2024). One of the primary research areas of the Lab for Inclusive Entrepreneurship involves discovering methods to encourage connections between procurement teams in higher education and diverse small businesses.

The lab is dedicated to tackling technical and managerial obstacles faced by small businesses, with a particular emphasis on those owned by socially and economically disadvantaged entrepreneurs.

Support from NL4IE extends to diverse small enterprises, with a specific focus on businesses in the Portland-Boston-Providence region (Zangerl, 2024). NL4IE's strategy involves integrating research expertise, practical learning opportunities, and community partnerships to empower a diverse range of small business owners with essential skills. This assistance is geared toward fostering growth and enabling these businesses to expand into broader markets, including academic institutions (NL4IE's Website, 2024).

Project Objectives

Current Procurement Methods:

The current procurement methods involve a cumbersome process of web surfing and navigating through multiple links to gather specific information. This process typically includes finding contact details, understanding requirements, identifying necessary certifications, and exploring potential benefits associated with procurement opportunities.

Challenges with Current Methods:

The existing approach is time-consuming, inefficient, and prone to errors. Vendors may struggle to locate accurate and up-to-date information, leading to frustration and confusion. Moreover, the decentralized nature of information across various websites complicates the procurement process further.

Introduction of Diversity Supplier Chatbot:

To address these challenges, the Diversity Supplier Chatbot offers a centralized solution. It

aggregates and organizes relevant procurement information, providing vendors with a streamlined and user-friendly interface to access essential details.

Project requirement

Our sponsors have helped us pinpoint the main goal of our chatbot project: to create a prototype that gives helpful information about Northeastern University's procurement process. We've narrowed our focus to ensure the chatbot provides clear and detailed responses tailored to the university's needs. This means users can expect accurate and useful information whenever they interact with the chatbot.

Challenges Faced

Throughout the project, we encountered several challenges that impeded our progress. Initially, missing data files hindered the full connectivity of our chatbot, despite efforts to retrieve them from previous team members. Unfortunately, we were unable to secure the necessary files, prompting us to embark on developing new code. Additionally, expired OpenAI API keys necessitated the creation of new ones, while the absence of an existing database led us to build one from scratch using the Pinecone API. Despite these obstacles, our determination to overcome these challenges fueled our progress, allowing us to optimize the chatbot's functionality and enhance user experience.

Research and Technical Methods

Our research and technical methods involved a comprehensive approach, including web scraping to access diverse online data sources and coding to extract information from databases. We focused on structured data storage for efficient retrieval and analysis, emphasizing the importance of staying updated with technology. Sponsor engagement played a crucial role, providing varied insights and ensuring alignment with project goals.

We adapted to the unique challenges of each webpage, addressing copyright concerns and refining our approach for accurate data acquisition, showcasing our commitment to meticulous practices. Below are the more particular areas we focused our qualitative research on.

User Experience Testing:

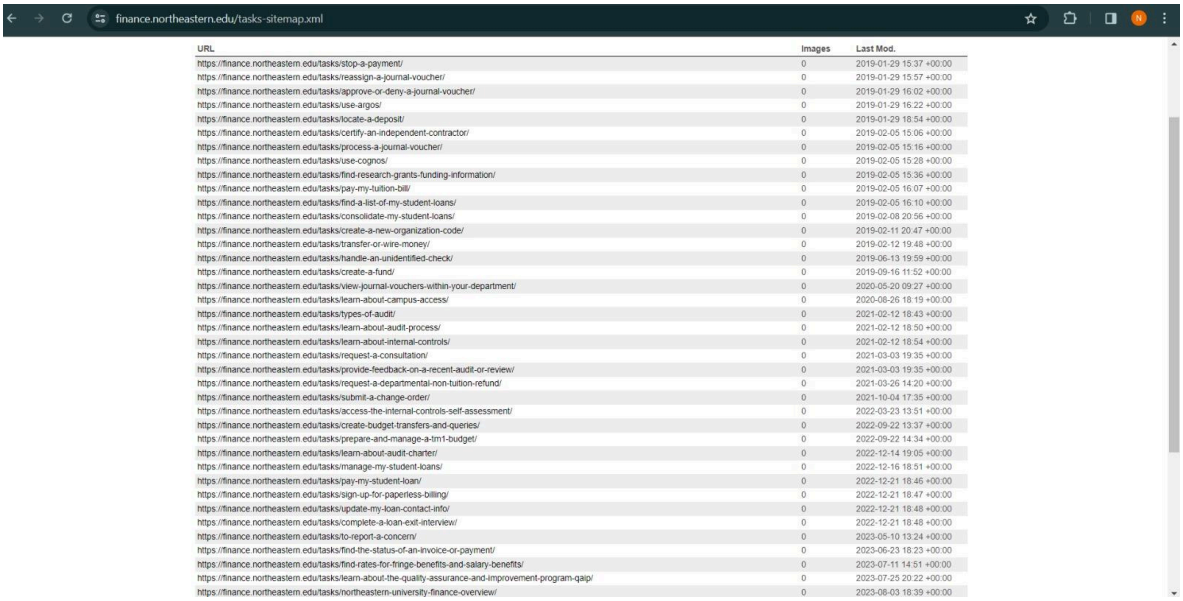
We conducted user experience testing by interacting with vendors to interact with our chatbot prototype and provide feedback based on their questions and interactions. This testing aimed to evaluate the usability, effectiveness, and user satisfaction of the chatbot in addressing their procurement-related queries and needs.

Content Analysis:

We performed content analysis to assess the relevance, accuracy, and completeness of the information provided by the chatbot in response to vendors' inquiries. This analysis helped ensure that the chatbot delivers accurate and helpful responses to enhance the user experience.

Web Scraping:

We utilized web scraping techniques to gather relevant data and information from procurement websites, supplier databases, and other online sources. For this process we scrapped more than 50 web pages. This data collection method enabled us to enrich the chatbot's knowledge base with up-to-date and comprehensive content, enhancing its effectiveness in supporting vendors' procurement efforts.



URL	Images	Last Mod.
https://finance.northeastern.edu/tasks/stop-a-payment/	0	2019-01-29 15:37 +00:00
https://finance.northeastern.edu/tasks/reassign-a-journal-voucher/	0	2019-01-29 15:57 +00:00
https://finance.northeastern.edu/tasks/approve-or-deny-a-journal-voucher/	0	2019-01-29 16:02 +00:00
https://finance.northeastern.edu/tasks/use-argos/	0	2019-01-29 16:22 +00:00
https://finance.northeastern.edu/tasks/locate-a-deposit/	0	2019-01-29 18:54 +00:00
https://finance.northeastern.edu/tasks/certify-an-independent-contractor/	0	2019-02-05 15:06 +00:00
https://finance.northeastern.edu/tasks/process-a-journal-voucher/	0	2019-02-05 15:16 +00:00
https://finance.northeastern.edu/tasks/use-cognos/	0	2019-02-05 15:28 +00:00
https://finance.northeastern.edu/tasks/find-research-grants-funding-information/	0	2019-02-05 15:36 +00:00
https://finance.northeastern.edu/tasks/pay-my-tuition-bill/	0	2019-02-05 16:07 +00:00
https://finance.northeastern.edu/tasks/find-a-list-of-my-student-loans/	0	2019-02-05 16:10 +00:00
https://finance.northeastern.edu/tasks/consolidate-my-student-loans/	0	2019-02-08 20:56 +00:00
https://finance.northeastern.edu/tasks/create-a-new-organization-code/	0	2019-02-11 20:47 +00:00
https://finance.northeastern.edu/tasks/transfer-or-wire-money/	0	2019-02-12 19:48 +00:00
https://finance.northeastern.edu/tasks/handle-an-unidentified-check/	0	2019-06-13 19:59 +00:00
https://finance.northeastern.edu/tasks/create-a-fund/	0	2019-09-16 11:52 +00:00
https://finance.northeastern.edu/tasks/view-journal-vouchers-within-your-department/	0	2020-05-20 09:27 +00:00
https://finance.northeastern.edu/tasks/learn-about-campus-access/	0	2020-08-26 18:19 +00:00
https://finance.northeastern.edu/tasks/types-of-audit/	0	2021-02-12 18:43 +00:00
https://finance.northeastern.edu/tasks/learn-about-audit-process/	0	2021-02-12 18:50 +00:00
https://finance.northeastern.edu/tasks/learn-about-internal-controls/	0	2021-02-12 18:54 +00:00
https://finance.northeastern.edu/tasks/request-a-consultation/	0	2021-03-03 19:35 +00:00
https://finance.northeastern.edu/tasks/provide-feedback-on-a-recent-audit-or-review/	0	2021-03-03 19:35 +00:00
https://finance.northeastern.edu/tasks/request-a-departmental-non-tuition-refund/	0	2021-03-26 14:20 +00:00
https://finance.northeastern.edu/tasks/submit-a-change-order/	0	2021-10-04 17:35 +00:00
https://finance.northeastern.edu/tasks/access-the-internal-controls-self-assessment/	0	2022-03-23 13:51 +00:00
https://finance.northeastern.edu/tasks/create-budget-transfers-and-queries/	0	2022-09-22 13:37 +00:00
https://finance.northeastern.edu/tasks/prepare-and-manage-a-tm1-budget/	0	2022-09-22 14:34 +00:00
https://finance.northeastern.edu/tasks/learn-about-audit-charter/	0	2022-12-14 19:05 +00:00
https://finance.northeastern.edu/tasks/manage-my-student-loans/	0	2022-12-16 18:51 +00:00
https://finance.northeastern.edu/tasks/pay-my-student-loan/	0	2022-12-21 18:46 +00:00
https://finance.northeastern.edu/tasks/sign-up-for-paperless-billing/	0	2022-12-21 18:47 +00:00
https://finance.northeastern.edu/tasks/update-my-loan-contact-info/	0	2022-12-21 18:48 +00:00
https://finance.northeastern.edu/tasks/complete-a-loan-exit-interview/	0	2022-12-21 18:48 +00:00
https://finance.northeastern.edu/tasks/to-report-a-concern/	0	2023-05-10 13:24 +00:00
https://finance.northeastern.edu/tasks/find-the-status-of-an-invoice-or-payment/	0	2023-06-23 18:23 +00:00
https://finance.northeastern.edu/tasks/find-rates-for-fringe-benefits-and-salary-benefits/	0	2023-07-11 14:51 +00:00
https://finance.northeastern.edu/tasks/learn-about-the-quality-assurance-and-improvement-program-qaip/	0	2023-07-25 20:22 +00:00
https://finance.northeastern.edu/tasks/northeastern-university-finance-overview/	0	2023-08-03 18:39 +00:00

Python Script Code:

We developed Python scripts to automate the web scraping process and extract specific data elements required for the chatbot's knowledge base. These scripts efficiently collected and processed information from various online sources, contributing to the robustness and accuracy of the chatbot's content.

Data Collection:

We collected data from multiple sources, including web scraping, user interactions with the chatbot, and vendor feedback surveys. This data collection process helped us gather insights into vendors' needs, preferences, and challenges in navigating procurement processes, informing iterative improvements to the chatbot's functionality and content.

Chatbot Workflow:

User Input:

Vendors interact with the chatbot by entering their queries or requirements.

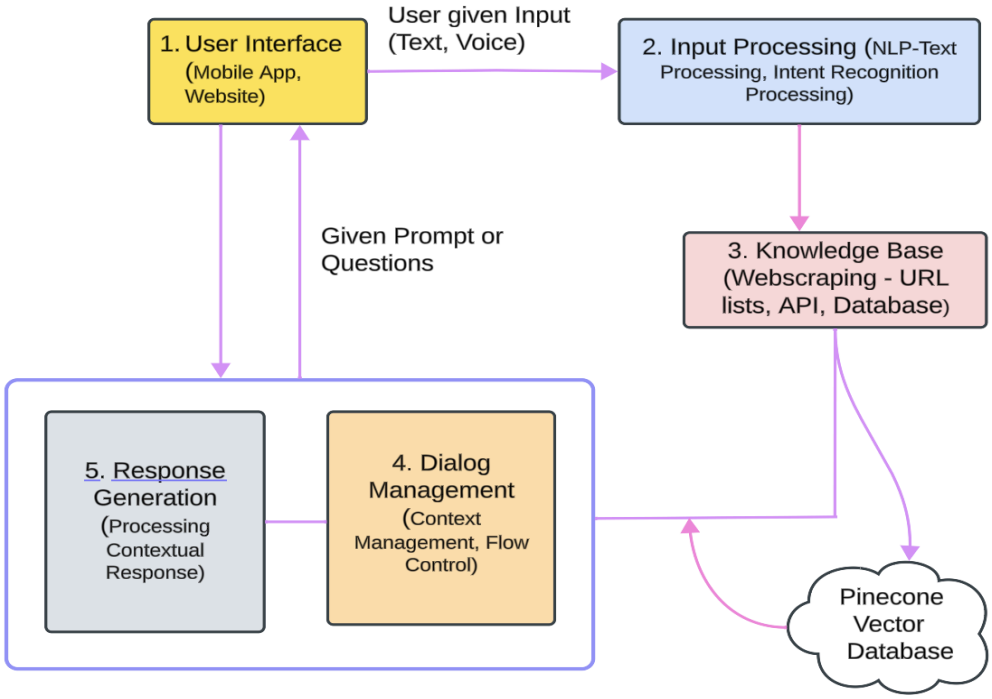
Query Processing:

The chatbot processes the user's input using a knowledge base, potentially powered by OpenAI API. This enables the chatbot to understand and interpret the user's intent effectively.

Generating Replies:

The chatbot composes replies to users based on the processed queries. To ensure accuracy and comprehensiveness, it internally scrapes data from targeted websites, gathering pertinent information related to the user's inquiry.

Chatbot Working Architecture:



As mentioned in the above picture, The user initiates the interaction by submitting a query via the user interface, which can be accessed through either a mobile phone or a web application. The query is then processed by the input processor in the second block, which identifies the user's intent from the question. Subsequently, the query is forwarded to the knowledge base in the third block, where it is matched with corresponding data stored in the Pinecone database. This data primarily consists of information scraped from university websites ensuring that the chatbot has access to accurate and relevant content. Finally, in the last block, the response generation process takes place. The chatbot utilizes the retrieved data to generate an appropriate response tailored to the user's request, thereby providing a seamless and efficient user experience.

Key Features of the Chatbot

Centralization:

The chatbot consolidates information from disparate sources, eliminating the need for vendors to navigate multiple websites.

Accessibility:

By offering an easy-to-use interface, the chatbot simplifies the procurement process, making critical information readily accessible to vendors.

Efficiency:

By automating the data retrieval process through web scraping, the chatbot enhances efficiency, enabling vendors to obtain information quickly and reliably.

Overall, the Diversity Supplier Chatbot revolutionizes procurement by leveraging AI technology to streamline information access, enhance user experience, and empower vendors in their procurement endeavors.

Detailed of Tools and Technologies Implemented in the Project

Front-end:

HTML:

Building upon the groundwork laid by our predecessors, we implemented the chatbot's user interface using HTML. With the structure already defined, our focus was on integrating interactive components like the university selection dropdown menu and category buttons. While the foundation was set, we faced the challenge of ensuring seamless user interactions and intuitive navigation within the existing framework. Despite the constraints, we successfully incorporated the necessary elements, contributing to the overall functionality

of the chatbot interface.

CSS:

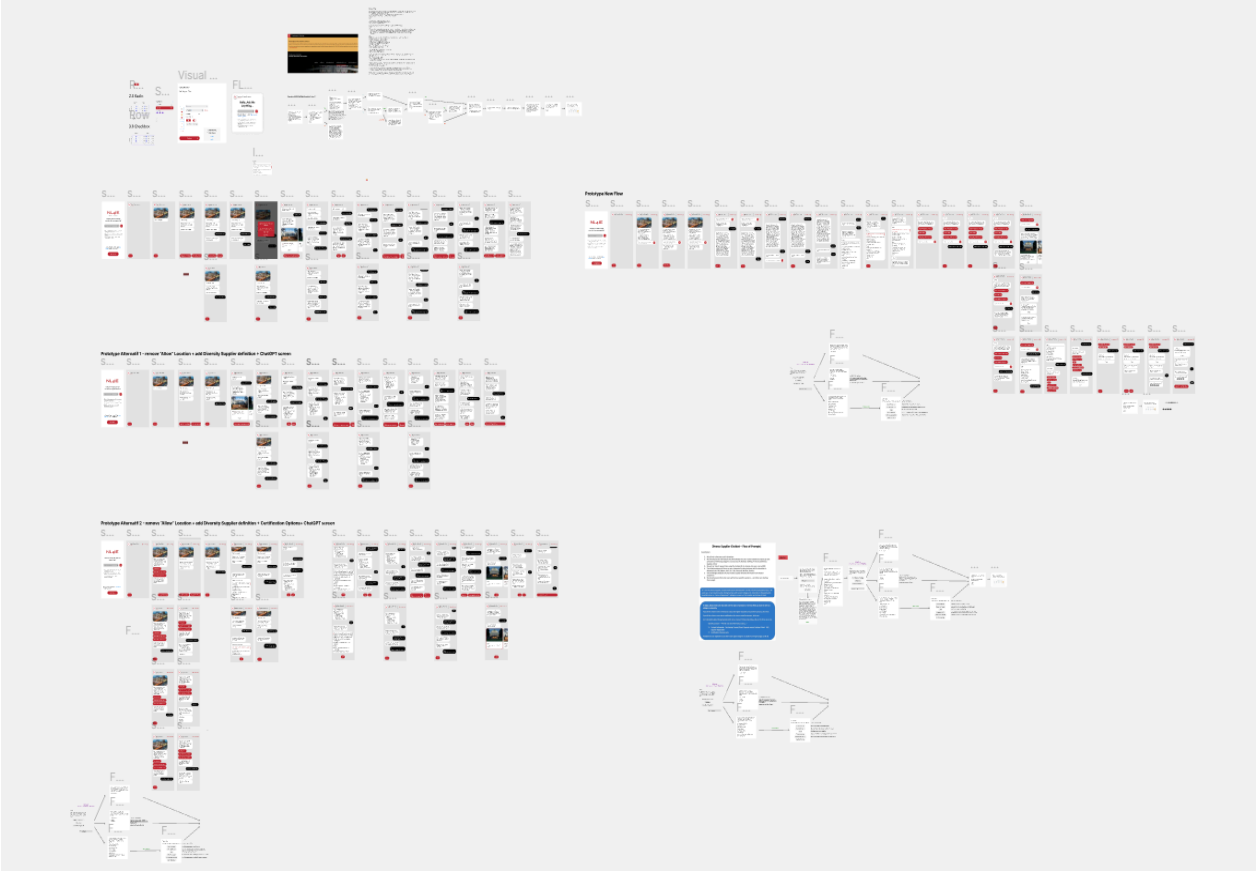
Drawing inspiration from the established design guidelines, we applied CSS styles to enhance the visual appeal of the chatbot interface. Working within the parameters set by the previous team, we fine-tuned the colors, fonts, and layout to align with Northeastern University's branding. Balancing aesthetic preferences with accessibility requirements presented its challenges, but our collaborative efforts resulted in a visually cohesive and user-friendly interface that maintained continuity with the existing design.

JavaScript:

Incorporating JavaScript for client-side development, we focused on enhancing the chatbot's interactivity and functionality. Leveraging existing event listeners and API communication structures, we refined the user experience by handling user interactions and dynamic UI updates. Our contributions in JavaScript coding enabled seamless communication with the backend server, ensuring real-time rendering of responses and efficient processing of user queries.

Figma UI/UX Design:

While the initial UI/UX design was crafted by the previous team using Figma, our role involved implementing design updates and optimizations based on feedback and evolving project requirements. Collaborating closely with the design team, we translated design mockups into functional UI components, ensuring consistency and coherence with the established visual identity. Despite inheriting the design framework, our input and refinements played a crucial role in enhancing the overall user experience of the chatbot interface.

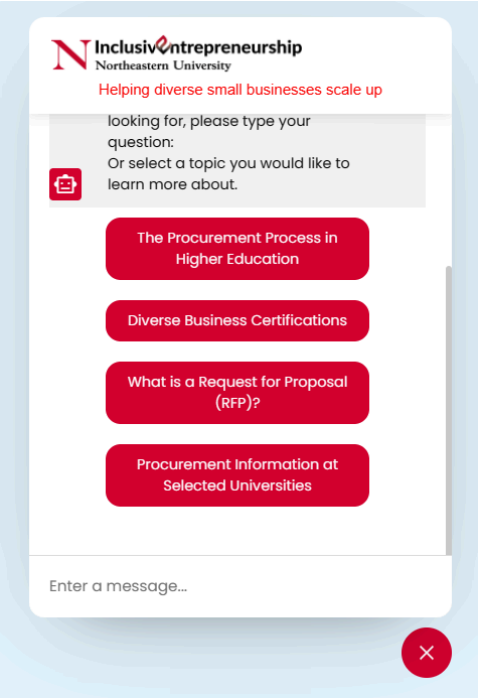


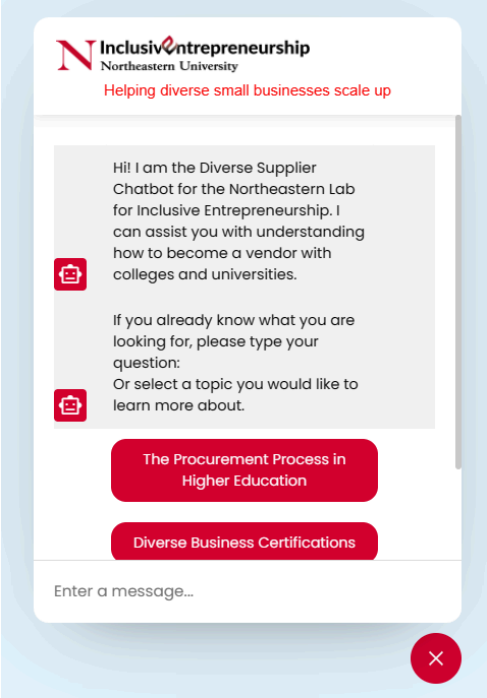
Link:

https://www.figma.com/file/2lCZLs01AoJY75i5v6tR6D/NL4IE-Chatbot-XN-Project?type=design&node-id=0-1&mode=design#-1

Initial UI Testing:

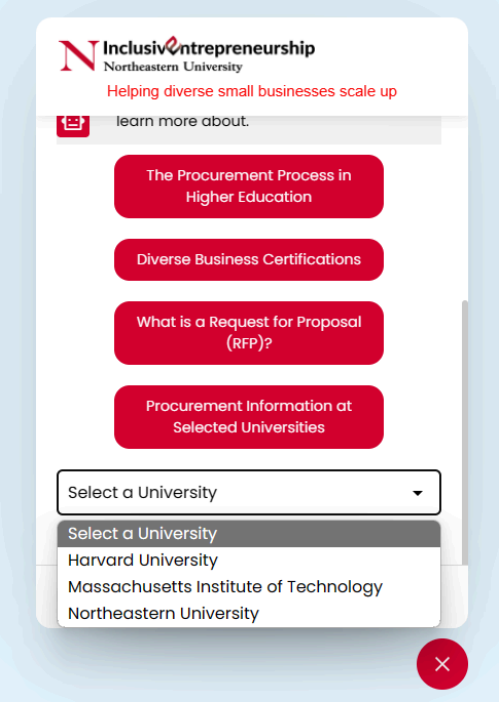
During the initial testing phase, we ensured that the user interface (UI) design, based on the FIGMA prototypes, was intuitive and user centric. Feedback from Dr. Z prompted adjustments to improve user understanding such as updating the welcome message and relabeling buttons for clarity. Despite occasional issues with response accuracy, the UI effectively facilitated user inputs and displayed chatbot responses accurately.

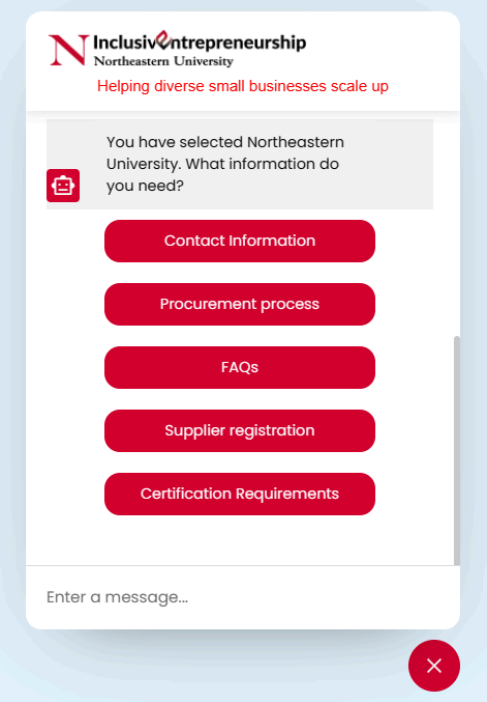




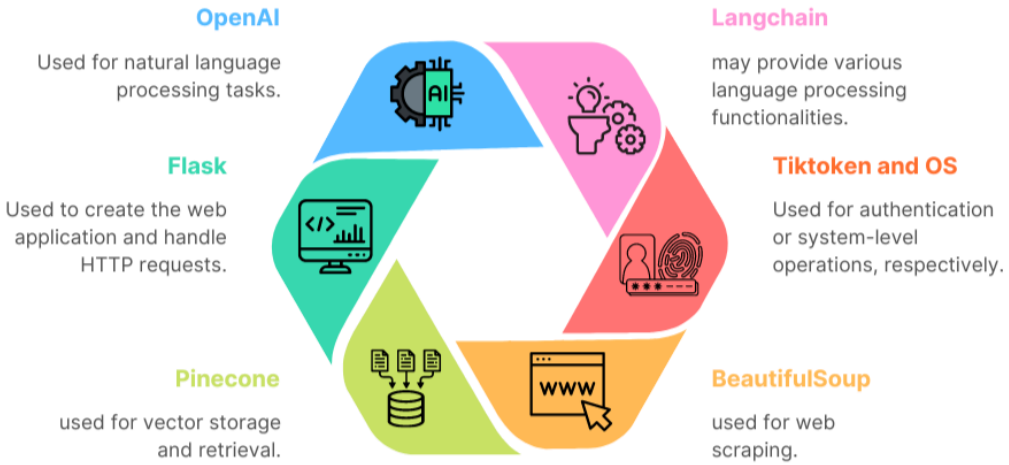
Final UI Testing:

Following iterations based on feedback from stakeholders, the final UI interface was refined and optimized. The chatbot interface features a simple yet impactful dropdown menu for university selection, allowing users to navigate personalized procurement processes seamlessly. This user-centric design approach ensures inclusivity and ease of use for diverse businesses engaging with Northeastern University.





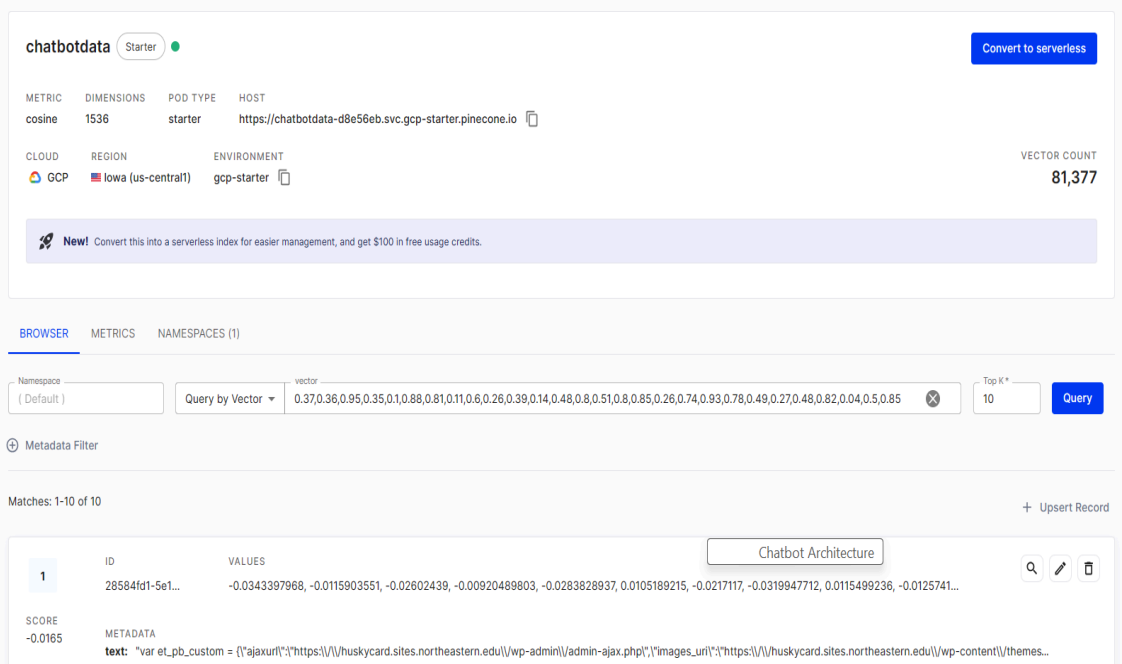
Middleware:



1. **OpenAI:**
   - Description: OpenAI offers powerful natural language processing (NLP) models, like GPT-4, for various text-based tasks.
   - Usage: We utilize OpenAI's GPT-4 model for understanding user queries and

generating responses in a conversational manner.

2. **Langchain:**
    - Description: Langchain is a Python library for NLP tasks, including text embedding, document retrieval, and question-answering.
    - Usage: It's used to split documents, perform text embeddings, and set up a question-answering chain for our chatbot.
3. **TikToken:**
    - Description: TikToken is a Python library for accessing TikTok's API and interacting with its features programmatically.
    - Usage: While not directly used in our project, it could integrate TikTok functionality into our chatbot.
4. **Beautiful Soup:**
    - Description: Beautiful Soup is a Python library for web scraping and extracting data from HTML/XML documents.
    - Usage: We use it to scrape content from web pages, extract text data, and store it in our system.
5. **Pinecone:**
    - Description: Pinecone is a vector database for storing high-dimensional vectors and performing similarity searches.
    - Usage: Pinecone is used to store document embeddings and efficiently retrieve relevant documents based on user queries.



1. **Flask:**
    - Description: Flask is a lightweight web framework for building web applications and APIs in Python.
    - Usage: It's used to develop the backend of our chatbot, handling HTTP requests, processing data, and generating responses.

Backend Development:

1. **Data Collection:**
    - Python scripts utilize the requests library to fetch web pages from university websites.
    - BeautifulSoup parser HTML content to extract relevant data.
2. **Data Processing:**

- Raw data undergoes cleaning and normalization steps to ensure consistency.
- OpenAI's text-embedding-ada-002 model transforms textual information into vector embeddings, capturing semantic meaning numerically.

3. **Data Storage:**
    - Vector embeddings are stored in a Pinecone database, facilitating efficient similarity searches.
    - Pinecone enables high-speed retrieval of information based on vector proximity.
4. **Query Processing:**
    - User input is converted into a query embedding using OpenAI's model.
    - The query embedding is matched against the Pinecone database to retrieve relevant information.
5. **API Layer:**
    - Developed with Flask, it manages RESTful API endpoints for communication between frontend and backend.
    - Receives requests from the frontend, interacts with the backend to fetch responses, and delivers them back to the frontend.
6. **Connecting to Middleware:**
    - Acts as an additional processing layer, performing tasks like data validation, request routing, and interaction orchestration with other systems.
    - Flask API: Our project utilizes middleware to facilitate communication between the front-end and back-end components. Leveraging Flask, a Python-based web framework, we develop RESTful APIs to interact with the Pinecone database. These APIs enable seamless integration and data retrieval from Pinecone, ensuring efficient handling of user queries and responses.

Below is the code snippet of flask utilization:

api.py > ask_question

```python
from flask import Flask, request, jsonify
from query import initialize_services, load_documents, initialize_qa_chain
from flask_cors import CORS
from flask_executor import Executor

app = Flask(__name__)
CORS(app)
executor = Executor(app)

embeddings = initialize_services()
docs = load_documents()
qa_with_sources = initialize_qa_chain(embeddings, docs)

@app.route('/ask', methods=['POST'])
def ask_question():
    try:
        data = request.get_json()
        question = data.get("question")
        if not question:
            return jsonify({"error": "No question provided"}), 400

        future = executor.submit(qa_with_sources, {"query": question})
        result = future.result()

        if 'result' in result:
            response = result['result']
            return jsonify({"response": response})
        else:
            return jsonify({"error": "No response from chatbot"}), 404

    except Exception as e:
        return jsonify({"error": str(e)}), 500

if __name__ == '__main__':
    app.run(debug=False)

```

7. **Integration:**

- Seamless alignment of APIs, data formats, web scraping, data processing, and storage across front-end, middleware, and backend components ensures a highly responsive, accurate, and up to date chatbot.
- Implementation of robust error handling mechanisms to gracefully manage integration issues.
- Complementary integration ensures enhanced efficiency and user experience within the existing ecosystem.

Hosting: https://diversitysupplychatbot.great-site.net/

Important Points for the Next Team

First and foremost, the next team should set clear objectives and understand that the primary goal of this project is to create a Diversity Supplier Chatbot that can serve as a supporting tool to assist with the procurement process with the University. It is crucial to maintain a focused approach to ensure that they can further enhance the working prototype, which can provide relevant responses and deliver knowledgeable output. We suggest the next team study the problem at hand, perform thorough research on the current procurement, and understand the challenges faced by small and diverse businesses to develop a product that meets the needs of the end users.

Data Collection and Relevance

While working on and developing new code for our chatbot, it is essential to select and use only relevant data. Therefore, it requires us to engage in careful observation and analysis to determine the data's usefulness. Our focus is on scraping only the most useful websites to avoid overwhelming the chatbot with irrelevant information. The reason is that our team agreed to prioritize high-quality output rather than scraping a large amount of unrelated and irrelevant information. This precision has allowed our team to build a chatbot that can provide reasonably good responses that generate satisfactory feedback to the user. In addition, we also set up a GitHub repository for our data collection and for our team members' collaboration, which helps us increase workload transparency, avoid duplicated work, and allows the team to work more efficiently.

APIs and Database Usage

We continuously update and manage OpenAI API keys regularly to ensure there is a proper connection to the database. We also monitor the data usage so it will not exceed our budget. Transitioning to the new and more efficient Pinecone database was a strategic choice; however, we always consider whether to possibly increase the database's usage as necessary. We have to ensure that the database is properly connected and integrated with our chatbot system.

Web Scraping Practices

We learn to use BeautifulSoup and other necessary tools for web scraping wisely. We only target university websites that provide the most relevant information on the procurement process. We strictly maintain a balance between comprehensive data collection and data relevance. We solely utilize information that is publicly accessible to ensure we steer clear of any ethical concerns.

User-Centric Response

Another key focus that the next team needs to know is how to think using a user-centric focus. Our end-users are small and diverse businesses, so we have to ensure that our generated responses are appropriate, easy to understand, and relevant to their queries. Hence, we work on refining language use and avoiding technical jargon; we also create different buttons for users to access information. The quality of responses, ease of access, including their simplicity and accuracy, is extremely important for the usability and effectiveness of our chatbot.

For the current prototype, we concentrated exclusively on Northeastern University, but

moving forward, we believe the next team has the opportunity to expand the scope significantly. By incorporating data from additional universities, the chatbot will be able to provide a broader range of relevant information catering to a more diverse audience. This expansion requires careful selection of sources to ensure the information remains pertinent and useful. Furthermore, the team should focus on optimizing the chatbot's functionality ensuring that it operates more smoothly and efficiently. This includes refining the algorithms responsible for parsing and responding to user queries. In crafting responses, we strongly advise for even greater clarity, relevance, and conciseness, thereby enhancing user satisfaction and engagement. This iterative process of expansion, optimization, and refinement will contribute to the development of a more robust and versatile Diversity Supplier Chatbot.

What Could Be Done Differently

As a team, we are glad to contribute to this project and be a part of such a rewarding initiative; however, if our team is given a chance to restart the project, we will try to give the live functionality and make a few changes to how we approach some of the challenges.

Changes can be made as following:

- **User Interface:** It's critical to make sure the UI works well and is accessible across a range of screens and devices. Media queries and adaptable layouts have shown to be successful, therefore they will remain top priorities. More actionable insights into user wants and pain points may be obtained by incorporating more direct and interactive user feedback mechanisms, such as in-app feedback features or more captivating user surveys.
- **Data Collection Approach:** Previously, we relied on the prior teams to work with

us in the initial weeks to discuss the project and try to understand their methodology. Later on in the weeks, we realized that this was not a good course of action. After we got going with our own strategy, we gathered a variety of information and used diverse methods, like qualitative research analysis, to gather information.

- **Time Management:** Incorporate more time for team members to learn the new system. This might include formal training sessions or self-study time. Identify potential risks and challenges that could delay the project and develop contingency plans for these risks. Use feedback from the testing phases and early stages of the migration to make iterative improvements. This can help in optimizing the process and managing time more effectively.
- **Documentation:** Ensure that documentation is updated or created for the new system. This is crucial for maintenance and future development efforts.
- **Database Management:** Managing vector-based data by switching from Pinecone to alternatives like Milvus or Weaviate is a big decision that might affect our system's cost-effectiveness and performance. Depending on the particular use cases and scalability requirements of your project, each of these systems has special features and benefits that might better meet the Chatbot’s needs. While abandoning Pinecone may result in lower expenses, we should take into account the operational difficulties involved in managing alternatives and self-hosting. At a premium, managed services provide convenience.
- **Collaboration and Research Sharing:** To accelerate the development of our GenAI Chatbot, we would benefit more if we prioritize the approach of collaborating with the broader research community and sharing our findings.

- **Exploration of New Technologies and Frameworks:** As artificial intelligence (AI) grows, new frameworks and technologies are always being developed. Therefore, we can restart the development process by incorporating cutting-edge research and techniques in a more exploratory manner.

References

- Accenture. (n.d.). Accenture Supplier Inclusion & Sustainability. Retrieved from https://www.accenture.com/us-en/about/company/accenture-supplier-inclusion-sustainability
- IBM.(n.d.). Supplier Portal. Retrieved from https://www.ibm.com/mysupport/s/topic/0TO500000002XcVGAU/supplier-portal?language=en_US
- Zangerl, C. Z., Grippa, F., Ryu, Y., & Singh, S. (2022). Promoting Supplier Diversity in Higher Education: Barriers and Opportunities. Lab for Inclusive Entrepreneurship
- *beautifulsoup4*. (2024, January 17). PyPI. https://pypi.org/project/beautifulsoup4/
- Gcu. (2021, May 20). Qualitative vs. Quantitative Research: What's the Difference? *GCU*. https://www.gcu.edu/blog/doctoral-journey/qualitative-vs-quantitative-research-whats-difference
- *Overview*. (n.d.). Pinecone. https://docs.pinecone.io/docs/overview?utm_term=pinecone%20db&utm_campaign=Brand+-+US/Canada&utm_source=adwords&utm_medium=ppc&hsa_acc=3111363649&hsa_cam=16223687665&hsa_grp=133738612775&hsa_ad=582256510975&hsa_src=g&hsa_tgt=kwd-1628011569784&hsa_kw=pinecone%20db&hsa_mt=p&hsa_net=adwords&hsa_ver=3&gad_source=1&gclid=CjwKCAiAopuvBhBCEiwAm8jaMTDqgeopz0H9UrZTFdZFe2yjEU2QKXBCriKfhmQQ-yndwM_cMzgPoRoCfHYQAvD_BwE
- *Supplier diversity in higher education*. (n.d.).

https://nl4ie.sites.northeastern.edu/supplier-diversity/