CS 410W Lab 1

Descriptive Document

Kobe Franssen

17/02/2025

Table of Contents

1 Introduction 3

2 CueCode Product Description 4

2.1 Key Product features and capabilities 4

2.2 Major Components 5

3 Identification of Case Study 6

4 X Product Prototype Description ..... **Error! Bookmark not defined.**

4.1 Prototype Architecture ..... **Error! Bookmark not defined.**

4.2 Prototype Features and Capabilities ..... **Error! Bookmark not defined.**

4.3 Prototype Development Challenges ..... **Error! Bookmark not defined.**

5 Glossary 6

6 List Of Tables 7

6.1.1 Competition Matrix 7

7 List Of Figures 8

7.1.1 CueCode Logo 8

7.1.2 Conceptual Diagram 8

7.1.3 Conceptual Diagram with NLP 9

7.1.4 Process Flowchart 9

7.1.5 Process Flowchart with OpenAPI specification 10

7.1.6 Translation from Natural Language to REST API 10

7.1.7 Major Functional Components 11

7.1.8 Major Functional Components with Translation 12

7.1.9 Major Functional Components with Customer Interaction 13

7.1.10 Overview of Major Functional Components 13

8 Works Cited 14

Table of Figures

Figure 1 8

Figure 2 9

Figure 3 9

Figure 4 10

Figure 5 10

Figure 6 11

Figure 7 11

Figure 8 12

Figure 9 13

Figure 10 13

1 Introduction

With the recent outbreak in popularity of Artificial Intelligence, a lot of different solutions are being released and updated daily but a lack of experience in the employee market and a lot of money in the future of Artificial Intelligence (AI) (University, 2024). AI is a term which has been around for a long time (Tableau, 2023), but with the sudden boom little were prepared or even understand how AI works. With AI there are many complexities and possibilities, it becomes very difficult for a single organization to affectively tackle the use of AI as a tool within a short time span without needing to hire a skilled and therefore expensive workforce.

AI is currently used to automate and simplify front facing applications such as chat bots or even complex tasks such as autonomous driving (Parlament, 2020). The most common service AI could replace with most to all businesses is a chat bot, as they currently are a tedious service to provide due to the 24 hour uptime requirement and possible delays and queues for customers. Many 3rd party services which provide human chatbots on websites have already adopted AI into their systems, by for example first allowing the individual to interact with the AI but still have the option to be transferred to a representative. In this case the AI is capable of returning information which can already be provided on the website and unable to complete business logic which the representative could. Example of this would be the ability to book an appointment with natural language with no delays and an interactive experience (Luminita Nicolescu, 2022).

This is where CueCode steps in and provides a service to go from natural language to business actions such as making appointments or obtaining user specific information. A plug and play application which requires little to no changes in the business logic and infrastructure. Allowing businesses to quickly pick up on Artificial Intelligence and its benefits.

2 CueCode Product Description

With the rapidly evolving landscape of artificial intelligence and software development, enterprises and Software-as-a-Service (Saas) providers encounter significant challenges when integrating large language models (LLMs) into their operation. Even though LLMs are remarkably good at producing language that seems human, the existing solutions does not have the means to integrate consistent business logic and human judgment into the API message production process. For businesses that depend on precise and secure API call execution for data entry and action triggering, this gap creates a high-risk environment.

Further, getting an LLM to produce Web API payloads and validating them is a specialized task requiring skills that many Web and fullstack developers do not possess. Since these developers are those most often building business applications, a solution for turning natural language into REST API payloads should take into consideration how easy it will be for developers with other skillsets to use the tool. CueCode gives a good foundation for Web and fullstack developers to turn natural language into REST API payloads.

2.1 Key Product features and capabilities

CueCode is an innovative framework and service designed to implement Natural Language Processing (NLP) capabilities, enabling the understanding and processing of natural language inputs. The system will offer a user-friendly interface through API client libraries, making it accessible for developers to integrate into their projects. Additionally, CueCode will provide a developer portal web application where users can upload OpenAPI specifications for their APIs and configure their CueCode service, streamlining the setup process.

The significance of CueCode lies in its unique approach to transforming natural language into Web API payloads, a functionality that is currently unparalleled for arbitrary REST APIs. This innovative solution is designed to work seamlessly with any REST API defined using an OpenAPI specification, offering unprecedented versatility and adaptability. By enabling non-technical staff and customers to

interact with complex APIs using simple language, CueCode dramatically improves access to technology and enhances the overall user experience.

One of the key accomplishments of CueCode is its ability to increase efficiency in REST API interactions. By allowing quick and accurate API calls based on simple language inputs, organizations can respond to customer requests more rapidly, leading to enhanced service levels. This improved efficiency not only streamlines operations but also allows client service representatives to focus more on customer engagement rather than getting bogged down in technical details, ultimately resulting in higher customer satisfaction.

CueCode addresses a significant problem in the tech industry by making NLP and Large Language Model (LLM) generation capabilities accessible to non-specialist developers. It applies these advanced technologies to the specific challenge of converting natural language into REST API payloads. By abstracting the technical complexity involved in generating API payloads, CueCode effectively bridges the gap between human input and the technical execution of requests via REST API calls. This solution empowers a wider range of users to leverage powerful API functionalities without requiring in-depth technical knowledge, thereby democratizing access to advanced technological capabilities.

2.2 Major Components

CueCode will require hardware capable of running the following systems separately (at most one at a time):

- Ollama 3.1, 70 billion parameter model
- Spacy.io (CPU or GPU)

Ollama 3.1 can be self hosted or used with an active subscription from 3rd party provider. Spacy.io on the other hand will need to be ran on hosted / rented servers as they will be processing all the data with confidentiality.

3 Identification of Case Study

The target audience for CueCode are Frontend Devs as the main thing they would have to incorporate is CueCode into their chatbot, upload their API specification and provide an API key with sufficient authentication for the actions they want CueCode to complete. To illustrate this, throughout our presentations we reference to a Frontend developer named Steve which wants to integrate text-to-API features into his Hospital website. While Patricia, a customer, will be using CueCode on the Hospitals website to make an appointment.

4 Glossary

API Payload (informal): Information that is sent together with an API request or response. This data, which can be organized in JSON or XML forms, usually includes the details needed by the client to comprehend the answer or by the server to carry out an action.

CueCode Developer Portal: A web-based platform that allows easy API creation with NLP-generated requests and gives developers access to CueCode's tools, API configuration, and integration workflow management.

HTTP Header: Additional metadata, such as the content type, authentication information, or caching instructions, are transmitted with HTTP requests and answers. Headers give context, which improves communication.

HTTP (Hypertext Transfer Protocol): The protocol that specifies the format and transmission of messages between web clients and servers. The type of request is determined by the HTTP methods (GET, POST, etc.).

Representational State Transfer (REST): A set of design guidelines for networked apps that use stateless, cacheable, and consistent HTTP processes to facilitate interaction. Through the use of common HTTP techniques, REST allows clients to communicate with servers by modifying resources that match an expected structure.

URL (Uniform Resource Locator): A web address that indicates where a resource is located on the internet. Protocol (such as HTTP/HTTPS), domain, and resource path are all included in URLs. They are necessary in order to access and consult internet resources.

5 List Of Tables

5.1.1 Competition Matrix

Feature	CueCode	OpenAI Functions	Google Natural Language API	Spacy.io	LangChain	GenKit	Phone AI
Entity recognition	✓		✓	✓		**P**	✓
Plug and Play	✓				**P**	**P**	**P**

Retrieval Augmented Generation	✓	✓			✓	✓	
API call generation as a service	✓	**P**	**P**		**P**		**P**

Competition matrix, showing features partially or fully implemented in CueCode's Competitors. Demonstrates CueCodes complete feature set as a framework for turning natural language into REST API payloads.

6 List Of Figures

6.1.1 CueCode Logo

Figure 1

6.1.2 Conceptual Diagram

Conceptual diagram of turning natural language into Web API payloads, with the NLP component left a mystery.

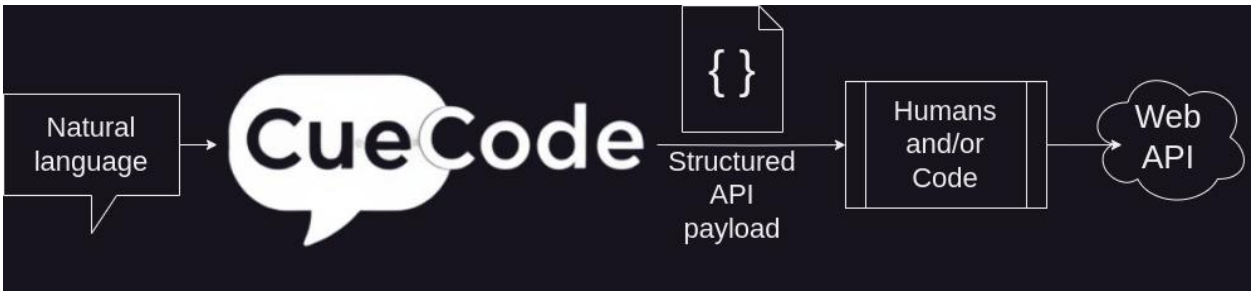

Figure 2

6.1.3 Conceptual Diagram with NLP

Conceptual diagram of turning natural language into Web API payloads, with CueCode shown as the NLP component.

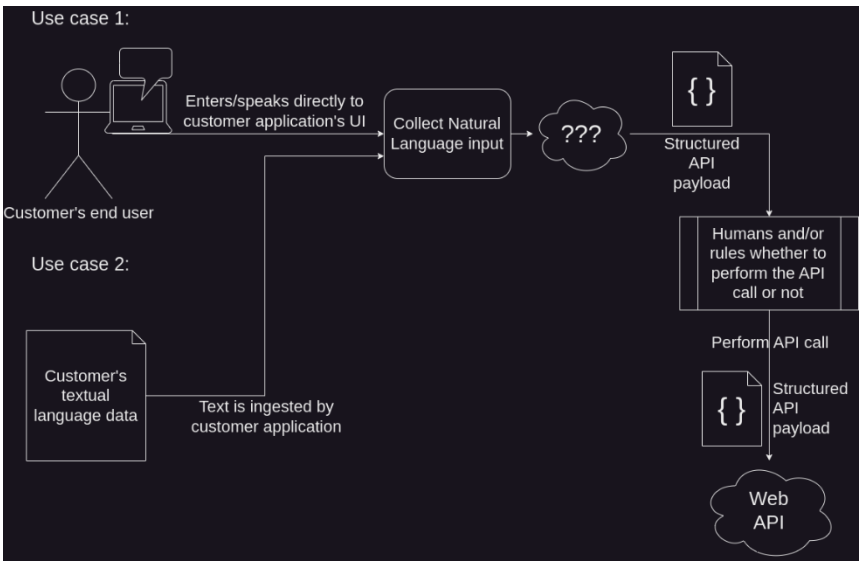

Figure 3

6.1.4 Process Flowchart

Current process flowchart for engineering REST API generation with OpenAPI specifications, showing the two example customer use cases from other slides. Major system components

involved at each process step are labeled with icons.

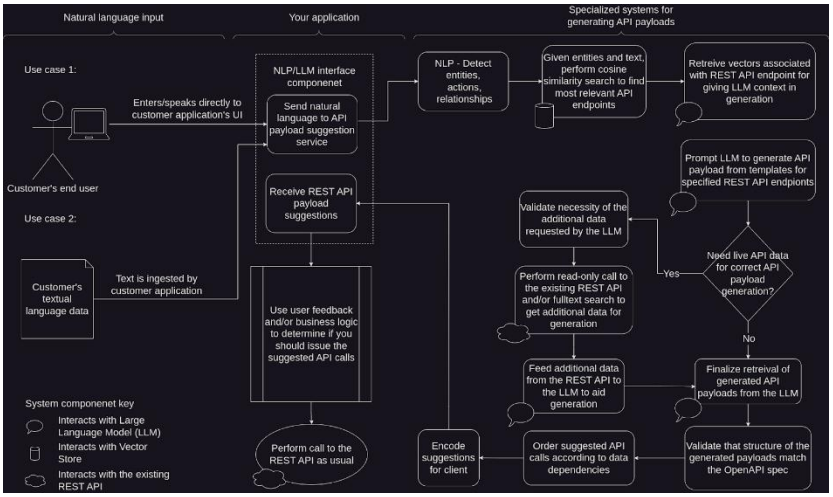

Figure 4

6.1.5 Process Flowchart with OpenAPI specification

The process flowchart for configuring CueCode with an OpenAPI specification.

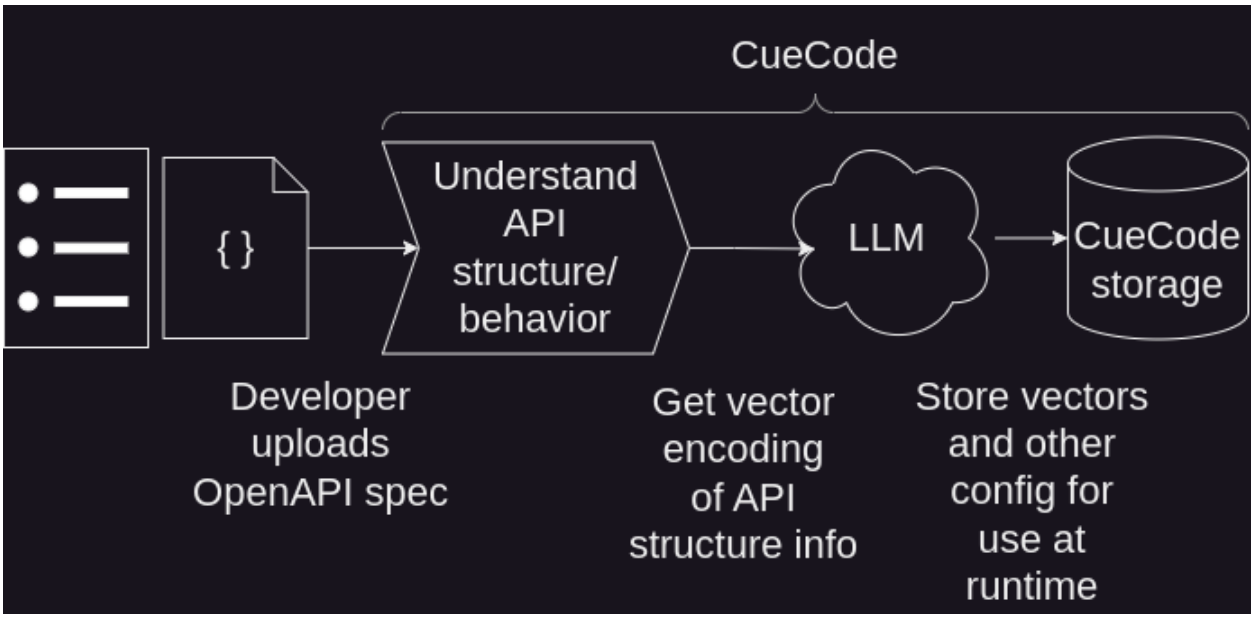

Figure 5

6.1.6 Translation from Natural Language to REST API

The process of turning natural language to REST API calls when using CueCode. Diagram includes validation and the two example use cases.

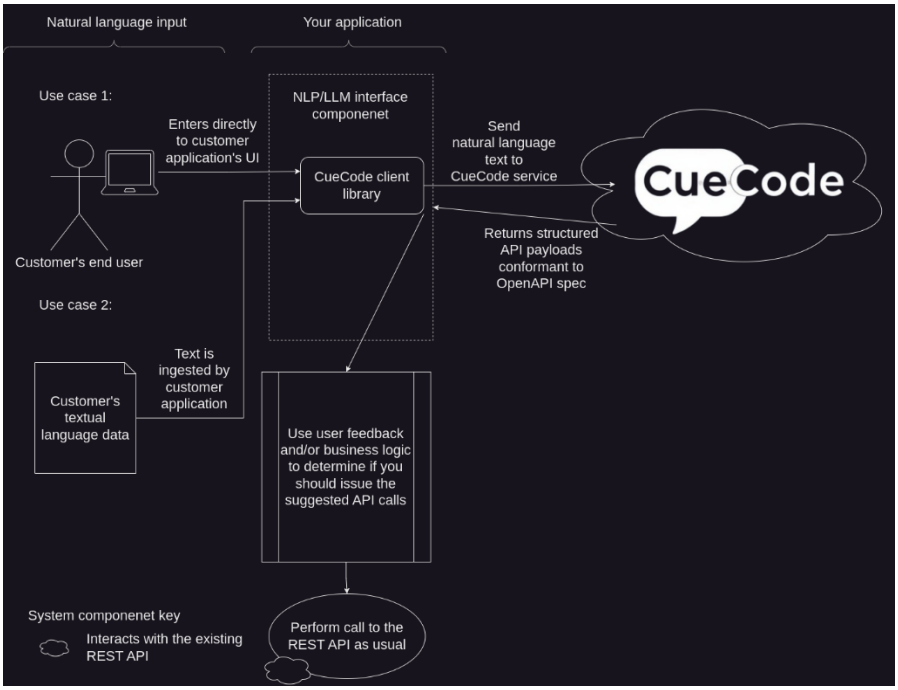

Figure 6

6.1.7 Major Functional Components

Major functional components diagram, focusing on customer interactions and components involved in the Developer Portal and CueCode configuration process.

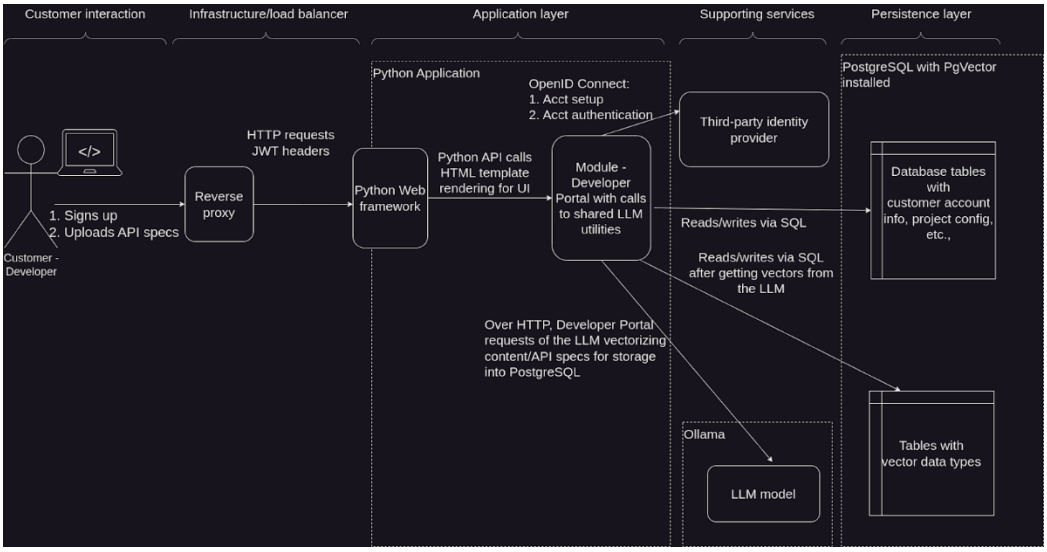

Figure 7

6.1.8 Major Functional Components with Translation

Major functional components diagram, showing how the customer's application can use CueCode to turn natural language to REST API payloads, perform validation of the payloads, then issue the suggested REST API calls.

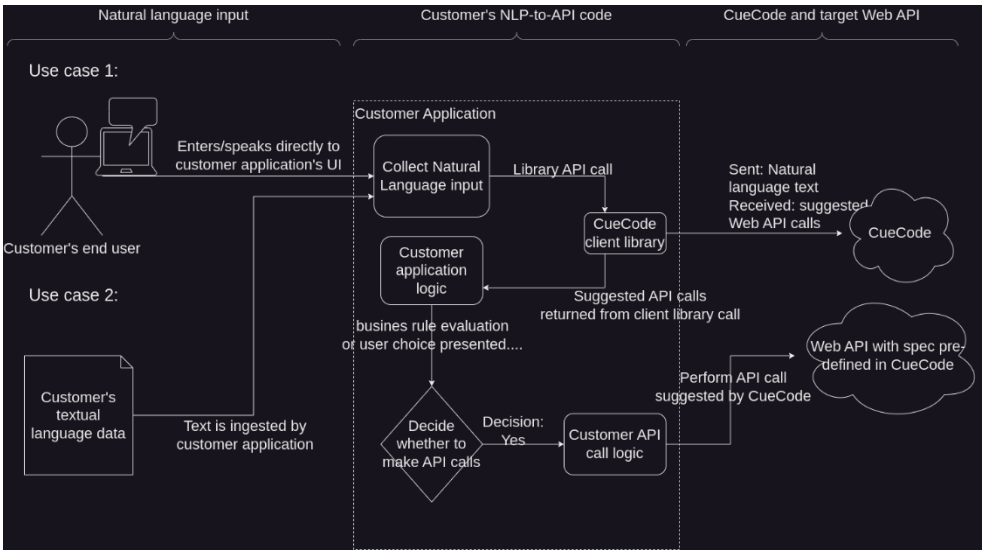

Figure 8

6.1.9 Major Functional Components with Customer Interaction

Major functional components diagram, focusing on the customer's application interacting with CueCode and components involved in the API payload generation process.

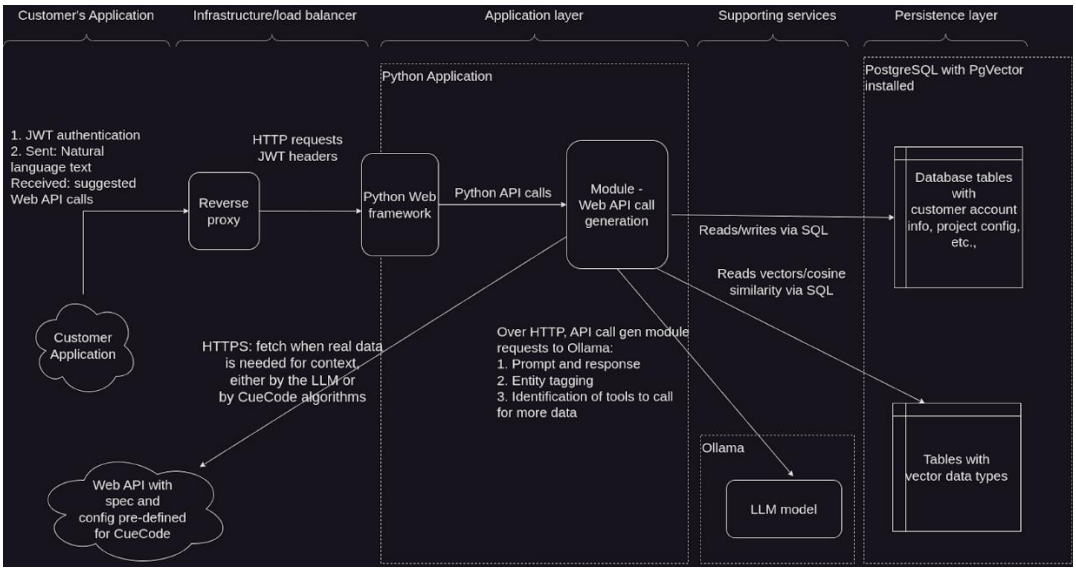

Figure 9

6.1.10 Overview of Major Functional Components

An overview of all major functional components involved.

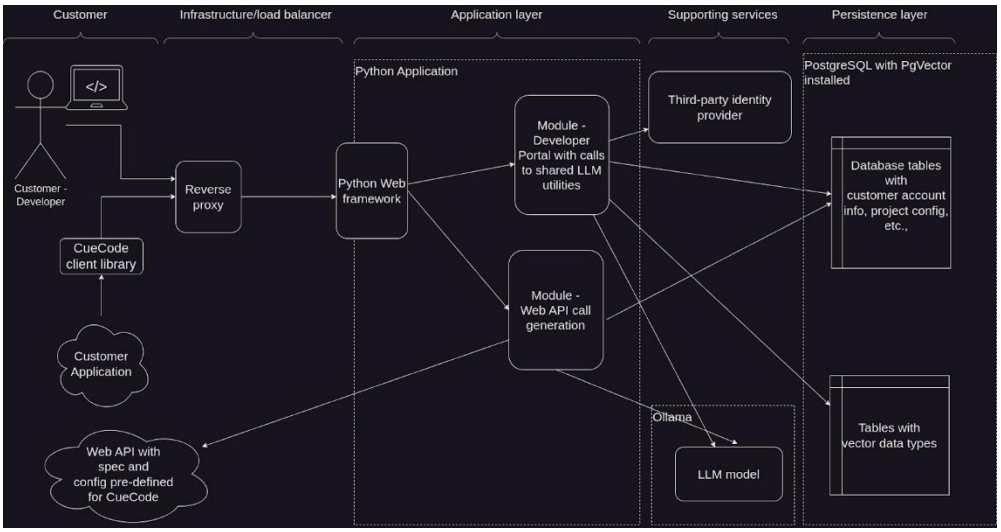

Figure 10

7 Works Cited

Luminita Nicolescu, M. T. (2022). Human-computer interaction in customer service: The experience with AI-Chatbots. *Electronics*, 1579. Retrieved from Electronics.

Parlament, E. (2020). *What is artificial intelligence and how is it used?* Retrieved from europarl europa: https://www.europarl.europa.eu/topics/en/article/20200827STO85804/what-is-artificial-intelligence-and-how-is-it-used

Tableau. (2023). *What is the history of artificial intelligence*. Retrieved from Tableau: https://www.tableau.com/data-insights/ai/history

University, N. (2024). *131 AI statistics and Trends*. Retrieved from National University: https://www.nu.edu/blog/ai-statistics-trends/

About continuous integration with GitHub Actions. (n.d.). GitHub Docs. Retrieved October 22, 2024, from https://docs.github.com/en/actions/about-github-actions/about-continuous-integration-with-github-actions

About Git. (n.d.). GitHub Docs. Retrieved October 22, 2024, from https://docs.github.com/en/get-started/using-git/about-git

Against LLM maximalism · Explosion. (2023, May 18). https://explosion.ai/blog/explosion.ai

AppDirect | Developer Portal. (2024). Appdirect.com. https://developer.appdirect.com/

Au-Yeung, J. (2020, March 2). Best practices for REST API design. Stack Overflow Blog. https://stackoverflow.blog/2020/03/02/best-practices-for-rest-api-design/

Baker, S. (2024). *Paragonsean/ChatBotAsync* [Python]. https://github.com/paragonsean/ChatBotAsync (Original work published 2024)

Cloud Natural Language. (n.d.). Google Cloud. Retrieved September 26, 2024, from https://cloud.google.com/natural-language

Evaluation | Genkit. (n.d.). Firebase. Retrieved September 14, 2024, from https://firebase.google.com/docs/genkit/evaluation

Firebase Genkit. (n.d.). Retrieved September 14, 2024, from https://firebase.google.com/docs/genkit

Function Calling. (n.d.). Retrieved September 14, 2024, from https://platform.openai.com/docs/guides/function-calling

HTTP headers - HTTP | MDN. (n.d.). Developer.mozilla.org. https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers

Learn Data with Mark (Director). (2023, July 26). *Returning consistent/valid JSON with OpenAI/GPT* [Video recording]. https://www.youtube.com/watch?v=lJJkBaO15Po

Lorica, B. (2024, January 25). *Expanding AI Horizons: The Rise of Function Calling in LLMs*. Gradient Flow. https://gradientflow.com/expanding-ai-horizons-the-rise-of-function-calling-in-llms/

Merritt, R. (2023, November 15). *What Is Retrieval-Augmented Generation aka RAG?* NVIDIA Blog. https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/

Microsoft/prompt-engine. (2024). [TypeScript]. Microsoft. https://github.com/microsoft/prompt-engine (Original work published 2022)

Natural Language Processing [NLP] Market Size | Growth, 2032. (n.d.). Retrieved September 14, 2024, from https://www.fortunebusinessinsights.com/industry-reports/natural-language-processing-nlp-market-101933

OpenAI Platform. (n.d.-a). Retrieved September 10, 2024, from https://platform.openai.com

OpenAI Platform. (n.d.-b). Retrieved October 24, 2024, from https://platform.openai.com

OpenAPI Specification—Version 3.1.0 | Swagger. (n.d.). Retrieved September 10, 2024, from https://swagger.io/specification/

OpenAPITools/openapi-generator. (2024). [Java]. OpenAPI Tools. https://github.com/OpenAPITools/openapi-generator (Original work published 2018)

piembsystech. (2023, October 2). *Dynamic Binding in Python Language*. PiEmbSysTech. https://piembsystech.com/dynamic-binding-in-python-language/

Scarpati, J. (n.d.). What is a URL (Uniform Resource Locator)? SearchNetworking. https://www.techtarget.com/searchnetworking/definition/URL

SpaCy · Industrial-strength Natural Language Processing in Python. (n.d.). Retrieved September 26, 2024, from https://spacy.io/

Stanfordnlp/dspy. (2024). [Python]. Stanford NLP. https://github.com/stanfordnlp/dspy (Original work published 2023)

Su, Y., Awadallah, A. H., Khabsa, M., Pantel, P., Gamon, M., & Encarnacion, M. (2017). Building Natural Language Interfaces to Web APIs. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 177–186. https://doi.org/10.1145/3132847.3133009

Tool/function calling | LangChain. (n.d.). Retrieved September 14, 2024, from https://python.langchain.com/v0.1/docs/modules/model_io/chat/function_calling/

Tutorial: ChatGPT Over Your Data. (2023, February 6). LangChain Blog. https://blog.langchain.dev/tutorial-chatgpt-over-your-data/

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2023). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models* (arXiv:2201.11903). arXiv. http://arxiv.org/abs/2201.11903

What Is NLP (Natural Language Processing)? | IBM. (2021, September 23). https://www.ibm.com/topics/natural-language-processing

What is Representational State Transfer (Rest) API? - Ampcontrol. (2024). Ampcontrol.io. https://www.ampcontrol.io/ev-terminology/what-is-rest-api

Why Visual Studio Code? (n.d.). Retrieved October 22, 2024, from https://code.visualstudio.com/docs/editor/whyvscode

W3Schools. (n.d.). HTTP Methods GET vs POST. W3schools.com. https://www.w3schools.com/tags/ref_httpmethods.asp

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2023). *ReAct: Synergizing Reasoning and Acting in Language Models* (arXiv:2210.03629). arXiv. http://arxiv.org/abs/2210.03629

Zafin, E. at. (2023, August 15). Bridging the Gap: Exploring use of Natural Language to interact with Complex Systems. *Engineering at Zafin*. https://medium.com/engineering-zafin/bridging-the-gap-exploring-using-natural-language-to-interact-with-complex-systems-11c1b056cc19