

Axter: a healthy chatbot

2772570n Ashwin Nanda

School of Computing Science

Sir Alwyn Williams Building

University of Glasgow

G12 8RZ

A dissertation presented in part fulfillment of the requirements of the Degree of Master of Science at the University of Glasgow

<Date of submission placed here>

**Abstract**

Education Use Consent

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic form.

<**Please note that you are under no obligation to sign this declaration, but doing so would help future students.>**

Name: Signature:

Acknowledgements

<Acknowledgements go here>

Contents

<Update the table of contents by right-clicking on it and selecting Update Field… and then select page numbers only.>

Chapter 1 Introduction <This is Heading 1> 1

1.1 A section <This is style Heading 2> 1

1.1.1 A subsection <This is style Heading 3> 1

Chapter 2 Survey 2

Chapter 3 Further Chapters 3

Chapter 4 Conclusion 4

Chapter 5 References 5

Appendix A <Name of appendix> 1

Appendix B <Another appendix> 2

# Introduction <This is Heading 1>

The web application project aims to create a chatbot-based platform that offers personalized assistance and collects user data to enhance user experience. The primary goal is to provide users with an interactive and user-friendly interface that addresses their queries and requirements efficiently. The project leverages the Django/Python framework for the back-end server, React.js for the front-end user interface, and integrates various APIs and external services to enrich the application's functionality.

The inspiration behind this project stems from the increasing demand for online platforms that offer personalized assistance and recommendations. With the proliferation of internet usage and the continuous growth of online services, users often face challenges in finding relevant information or making informed decisions. The chatbot-based approach offers an intuitive and conversational way for users to interact with the application and obtain tailored suggestions and responses.

The project's core components include the user interface, back-end server, APIs, and database. The user interface is designed using React.js to provide a responsive and dynamic front-end experience. It incorporates modern design principles and interactive elements to ensure an engaging user journey.

The back-end server, built on Django/Python, handles the business logic, user authentication, and database management. It interacts with the database, which stores user profiles, preferences, and historical data to provide personalized recommendations and track user activity.

**Natural Language Processing and Machine Learning Influence**

To enhance the chatbot's capabilities, the project integrates external APIs, such as natural language processing (NLP) services and recommendation engines. These APIs enable the chatbot to understand user queries, extract relevant information, and provide accurate responses and recommendations.

The project also emphasizes user data privacy and security. Measures are implemented to ensure that user data is securely stored and processed in compliance with relevant data protection regulations. Users have control over their personal data and can manage their preferences regarding data tracking and usage.

By combining intelligent chatbot technology, a user-friendly interface, and personalized recommendations, this web application project aims to deliver an enhanced user experience and provide valuable assistance to users in their decision-making process. It strives to create a platform where users can obtain relevant information, engage in meaningful conversations, and receive personalized recommendations to meet their specific needs and preferences.

# Survey

Each new chapter should appear on a new page.

# Further Chapters

<Figure below is in style “figure” which continues to style “figure caption” when you press Enter and then back to “Normal” when you press Enter again.>

Figure 1: Some important shapes.

<If you wanted to show any code fragments, you could use the following style called code, which could then be followed by figure caption..>

*# This is a little bit of Python*

**for** i in range( 10 ):

**for** j in range( 10 ):

**print** i\*j,

**print**

Figure 2: A crucial algorithm for the project.

# Conclusion

Show how you plan to organise your work, identifying intermediate deliverables and dates.

# References

[1] C. Baier and J.-P. Katoen. *Principles of Model Checking*. MIT Press, 2008.

###### <Name of appendix>

<Use Heading 6 for the Appendix heading>

###### <Another appendix>