UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Course: Systems Analysis.

Teacher: Carlos Sierra.

Technical report

Student: Jaider Santiago Avila Robles – 20231020200

Student: Alejandro Mauricio Junco Oviedo – 20231020129

Bogotá, D.C.

October 4, 2024

Video Game Review Platform with AI-Based Chatbot

1. **General Representation of the System**
   The system is an online platform where users can explore video game reviews and receive personalized recommendations through an AI-based chatbot. The chatbot gathers all the information from previously uploaded PDF documents.

    1. **Website Access**: Users access the platform via a web browser.
    2. **Available Options**:
        - **Explore Reviews and Recommendations**: Users can browse reviews organized by categories or directly search for video games.
        - **Interact with the Chatbot**: The chatbot asks users about their video game preferences and offers recommendations based on their input.
    3. **User Actions**:
        - Users can:
            - Read additional reviews.
            - Filter searches by category or return to the homepage.
            - Ask the chatbot for more recommendations or save their preferences.
    4. **Session End**: Users can leave the site or make new inquiries with the chatbot.

2. **System Sensitivity Analysis**
   This analysis explores how different parameters affect system performance:

    - **Concurrent Users**: The system's ability to handle multiple concurrent users is limited by server resources and backend efficiency. As the number of users increases, response times may rise.
    - **Chatbot Accuracy**: The chatbot's effectiveness depends on the quantity and quality of training data. With more reviews and queries, the chatbot improves in its recommendation accuracy. However, if the dataset is insufficient or biased, the chatbot's performance may be affected.
    - **Review Volume**: As the number of reviews increases, query time may be impacted. However, the system is designed to be scalable, allowing it to process a large volume of documents without significant performance

degradation. By using optimized techniques for reading and extracting data from PDFs, the chatbot can continue providing fast and accurate responses, even with a high volume of documents.

3. **System Complexity Analysis**
   The system is composed of various interconnected components, adding complexity to its design and development:
   - **System Components**: The system includes a frontend, backend, database, and AI-based chatbot. Each component must be efficiently integrated for optimal performance. The interaction between these modules adds complexity due to the need for user authentication, session management, and data persistence.
   - **Interactions**: Each component must manage concurrent requests. For instance, the backend should process multiple requests simultaneously, while the chatbot must handle real-time queries. As the number of users increases, the number of connections and data processed grows exponentially.
   - **Chatbot Training**: Training and optimizing the language model used by the chatbot adds a significant layer of complexity. This process involves tuning hyperparameters, improving the quality of training data, and continuously iterating to enhance the chatbot's accuracy and efficiency.
4. **Emergent Behaviors**
   During the system's development and implementation, unanticipated or emergent behaviors may arise, which need to be identified and managed:
   - **Chatbot Errors**: The chatbot may misinterpret ambiguous queries or provide out-of-context answers. This stems from the inherently unpredictable nature of natural language interactions.
   - **Review Biases**: Users may leave subjective or biased reviews, negatively impacting the accuracy of recommendations. The chatbot must have mechanisms to recognize these biases and minimize their influence on decision-making.
   - **Undesired Interactions**: Users may interact with the system in unforeseen ways, such as submitting unrelated queries or spam. These undesired interactions can cause performance issues or affect the user experience.