Diseño y arquitectura de software

Proyecto Final

Integrantes

Abraham Alejandro Sáenz Tirado A00823861 **ITC**

12 abril 2023

Create a list of the main functional requirements of each system.

- Identify at least 10 requirements for the whole architecture.
    1. The system must allow users to create tweets with a maximum length of 300 characters.
    2. The system must allow users to reply to tweets.
    3. The system must display tweets with a comment as a thread.
    4. The system must implement scrollable answers for a tweet thread.
    5. The system must display the recent 10 tweets on the user's home dashboard.
    6. The dashboard must display both tweets and threads.
    7. Every action triggered by Twitter reloaded must be registered as a new event at the event dashboard.
    8. The system must save the following information for each event: type of action, user who did the action, and timestamp.
    9. The system must be able to generate reports based on the stored event data.
    10. The reports must answer specific questions such as the user who registered the most events, the most commented tweet of the day, and the number of users who opened the application during the day.

- Include in this list as well the requirements that may not impact functionality but are required in order to glue the system together.
    1. A database to store tweets, threads, and user information.
    2. A user authentication system to manage user accounts and sessions.
    3. A data processing system to handle event data, generate reports, and store them in a database.

Create a list of the architecture characteristics for each system.

- Identify at least 5 architecture characteristics as a whole.
    1. Scalability: The system should be scalable to handle many tweets and replies, and to support the increasing number of users.
    2. Real-time processing: The system should be able to process events in real-time and register them on the event dashboard.
    3. Data storage: The system needs to have a reliable and scalable data storage solution to store tweets, replies, and event data.
    4. Modular design: The system should be designed in a modular way to make it easy to maintain, modify and extend in the future.
    5. User authentication and authorization: The system needs to have a robust user authentication and authorization mechanism to ensure that only authorized users can create tweets, reply to tweets, and access the event dashboard.

- The architecture characteristics are divided into explicit and implicit.

Explicit architecture characteristics:

1. Maximum character limit of 300 for tweet creation.
2. Ability to reply to tweets.
3. Display of tweets with comments as a thread.
4. Implementation of scrollable answers for threads.
5. Display of recent 10 tweets on the user's home dashboard.
6. Inclusion of tweets and threads on the user's home dashboard.

Implicit architecture characteristics:

1. A user authentication system to keep track of the user who performed the actions.
2. A database to store event data including action type, user, and timestamp.
3. An event dashboard interface to display registered events.
4. Logic to analyze event data and generate reports for questions such as user with the most events, most commented tweet, and number of users who opened the application during the day.
5. Implementation of appropriate security measures to ensure user data privacy and prevent unauthorized access to the system.

- You add a justification about why you sorted the characteristics that way.
    - The reason for sorting the characteristics into explicit and implicit is to differentiate between the requirements that are explicitly mentioned in the project specifications and the requirements that are not explicitly stated but are necessary to fulfill the stated requirements.
    - The explicit characteristics are the ones that are specifically mentioned in the project specifications, and are essential to meet the project's objectives.
    - The implicit characteristics, on the other hand, are requirements that are not explicitly stated but are necessary to fulfill the explicit requirements

Create one Sequence Diagram about the information flow for the whole architecture.