

Project 1 (Business Requirements): Speedrun Analytics with a Database System

Problem Statement

In the dynamic world of speedrunning, where gamers strive to complete games in the fastest possible times, the community is vibrant and continuously evolving. The primary hub for this community, speedrun.com, hosts an extensive array of data encompassing games, players, records, and more. However, despite the abundance of data, the platform faces significant challenges in data management and analytics. The existing infrastructure lacks the capability to handle complex queries efficiently, perform in-depth analysis, and provide actionable insights into speedrunning trends and patterns. This gap hinders the community's ability to engage deeply with the data, analyze performance trends across different games and categories, and foster a more connected and informed community. There is a pressing need for a structured system that not only aggregates and stores this wealth of data but also organizes it in a way that supports efficient data retrieval, comparison, and advanced analytics.

Objective

The aim of this project is to design and develop a comprehensive database system tailored to the needs of the speedrunning community. This system will consolidate data from speedrun.com, encompassing detailed records of games, players, speedruns, categories, and rankings. By doing so, it will enable users to execute complex queries with ease, perform sophisticated analytics, and gain insights into emerging trends within the speedrunning universe. The ultimate goal is to enhance community engagement through a more interactive and insightful exploration of speedrun data, thereby elevating the overall speedrunning experience.

Database System Rules for Speedrunning Analytics

1. Data Aggregation and Storage

- The system must aggregate data from speedrun.com.
- It should store comprehensive records of games, players, speedruns, categories, and rankings.

2. Data Organization

- Data must be organized to support efficient retrieval.
- The system should allow for easy comparison of records across different games and categories.

3. Query Execution

- The database must handle complex queries efficiently.

- Users should be able to retrieve data without facing performance issues.

4. Data Integrity and Reliability

- Ensure data integrity by accurately reflecting records from speedrun.com.
- Maintain data reliability through regular updates and validation checks.

5. Performance and Scalability

- Design the database for high performance, even with complex queries.
- Ensure the system is scalable to accommodate growing data and user base.

6. Security and Privacy

- Implement security measures to protect user data and privacy.
- Ensure that users can only access data they are permitted to view.

Nouns:

- system
- data
- speedrun.com
- games
- players
- speedruns
- categories
- rankings
- data management
- security
- privacy

Verbs:

- aggregate
- support
- provide
- ensure
- facilitate
- enhance
- design
- enable
- perform
- protect

In-memory key-value storage functionalities

Session Management

- **Functionality:** Store and manage user session data in real time. This includes user authentication tokens, session states, and active time stamps.
- **Benefits:** Rapid access and updates to session data improve the system's response time for user login and ensure security through timely session validation.

Leaderboards

- **Functionality:** Maintain real-time leaderboards for different games and categories. Update these leaderboards as new run data is received.
- **Benefits:** Leaderboards updated in real-time enhance user engagement and community interaction by providing immediate feedback on players' rankings.

Real-Time Notifications

- **Functionality:** Use Redis pub/sub capabilities to handle real-time notifications about upcoming events, new records, or community announcements.
- **Benefits:** Facilitates immediate communication within the community, keeping players informed and engaged with the latest updates and events.

User Preferences and Settings

- **Functionality:** Store user-specific settings and preferences such as display options, favorite players, or preferred categories.
- **Benefits:** Enhances user experience by allowing quick retrieval and application of individual preferences across sessions, improving personalization and interface responsiveness.

Most Viewed Runs

- **Functionality:** Track and store the most viewed runs or popular game analysis in Redis.
- **Benefits:** Quickly provides users with trending content, enhancing engagement and ensuring content relevancy.