Dat220 - Problem Analysis

April 25, 2025

1 Introduction

The project involves developing a gaming website with database integration to manage user data, game scores, live chat, and other features. The goal is to create a functional application that supports user interaction.

2 Problem Understanding

A key challenge in the project is designing a database that efficiently handles different data types and their relationships. The main challenges include:

- Securing user data and access rights
- Efficient storage and retrieval of game scores
- Implementation of a dynamic live chat
- Managing user feedback and there opinions.
- Storing and handling files, comments, and notes
- Maintaining performance and scalability

3 Identification of Entities and Relationships

To meet the requirements, the database must include the following entities:

3.1 Main Entities

- **User**: Stores private information such as username, email, and password.
- **Game Hi-Scores**: Stores scores and game history.
- **User Notes**: Personal notes for users.
- **Live Chat**: Allows users to communicate in real-time.

- **Feedback**: Feedback on the website - only visible to admin
- **Files**: Upload and sharing of game-related files.
- **Comments**: Comments on profiles, game results, or posts.
- **Sessions**: Using count function to see how many active sessions there is on the website at that time.

3.2 Relationships

- A **user** can have multiple **game scores**, **notes**, **comments**, and **files**.
- **Comments** and **files** can be linked to specific **users**.
- **Live chat** must support communication between multiple **users** in real-time.

4 Potential Issues and Solutions

4.1 Security

Problem: User data must be secured to protect privacy.
Solution: Implement password hashing and access control.

4.2 Data Integrity

Problem: Ensure that data such as game scores and comments remain consistent.
Solution: Use database constraints like primary and foreign keys.

4.3 Performance and Scalability

Problem: Live chat and large amounts of data can impact performance.
Solution: Use indexing and optimized SQL queries.

5 Conclusion

To develop a successful web application, the database must be designed with a clear structure that efficiently manages relationships between entities. By addressing potential problems early, the system can be both functional and scalable.