For our Week 3 challenge, we undertook the task of enhancing our gaming event management system by introducing a real-time chat feature.

- Database Enhancement:

We introduced two primary types of chat functionalities.
One-on-One Chats: Enabling two players to interact privately.
Group Chats: We allowed players to create or join chat 'rooms' or 'channels.'
These channels can be general or themed around quests, item trading, etc.
We took the initiative to implement the @mention Feature. This allowed players to mention others using the '@' symbol, triggering a unique notification for the mentioned user.
To accommodate these features, we expanded our database schema by:
Creating tables to store messages, associating them with their respective sender and chat session.
Modifying some existing tables to relate player details with chat logs.
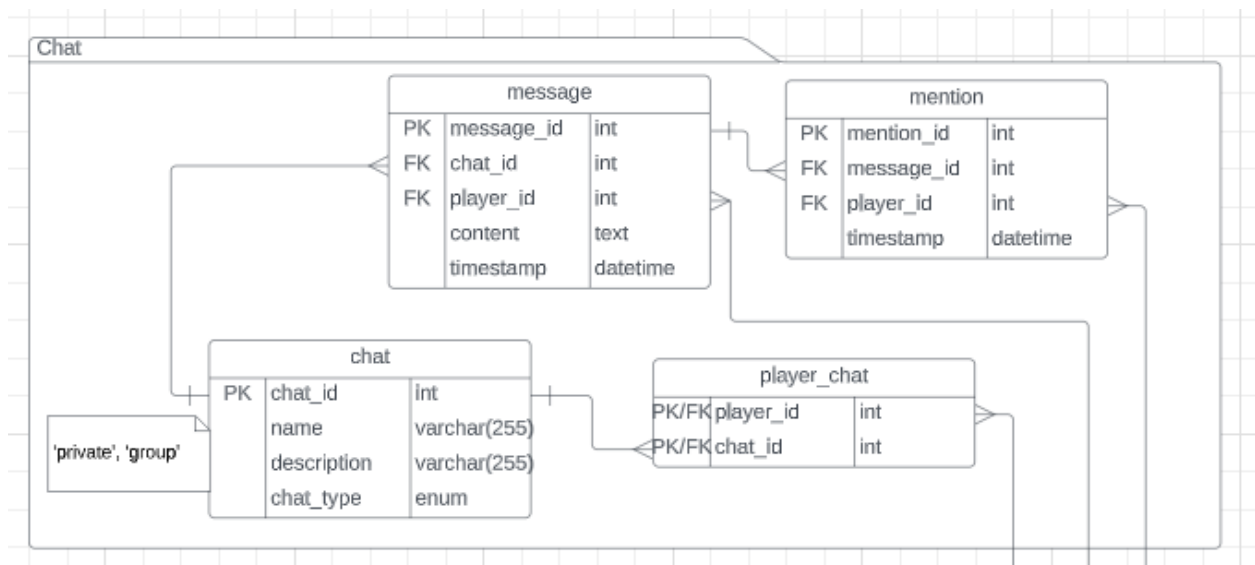
- **Chat Functionality Implementation:**

Using SQL triggers, we ensured that players were promptly notified of new messages or when they were @mentioned.
We utilized stored procedures to simplify complex queries like fetching a specific number of latest messages.
Every message stored in the database was timestamped to preserve chronology.

hey @Alice	Send

message_id	chat_id	player_id	content	timestamp
24	2	4	@Alice	2023-10-20 14:52:32
25	2	4	hey @Alice	2023-10-20 14:55:41
26	2	1	@Alice	2023-10-20 14:56:40

mention_id	message_id	player_id	timestamp
10	10	10000	2023-10-20 11:50:39
11	24	10000	2023-10-20 14:52:32
12	25	10000	2023-10-20 14:55:41
13	26	10000	2023-10-20 14:56:40

*a message that contain a @mention is stored both in the message table and mention table

- Programming Enhancements:

Our team took extra precautions with Data Validation, ensuring every piece of data was verified before being saved into the database.
An Error Handling mechanism was established, ensuring smooth operations even during network glitches or temporary database inaccessibility.
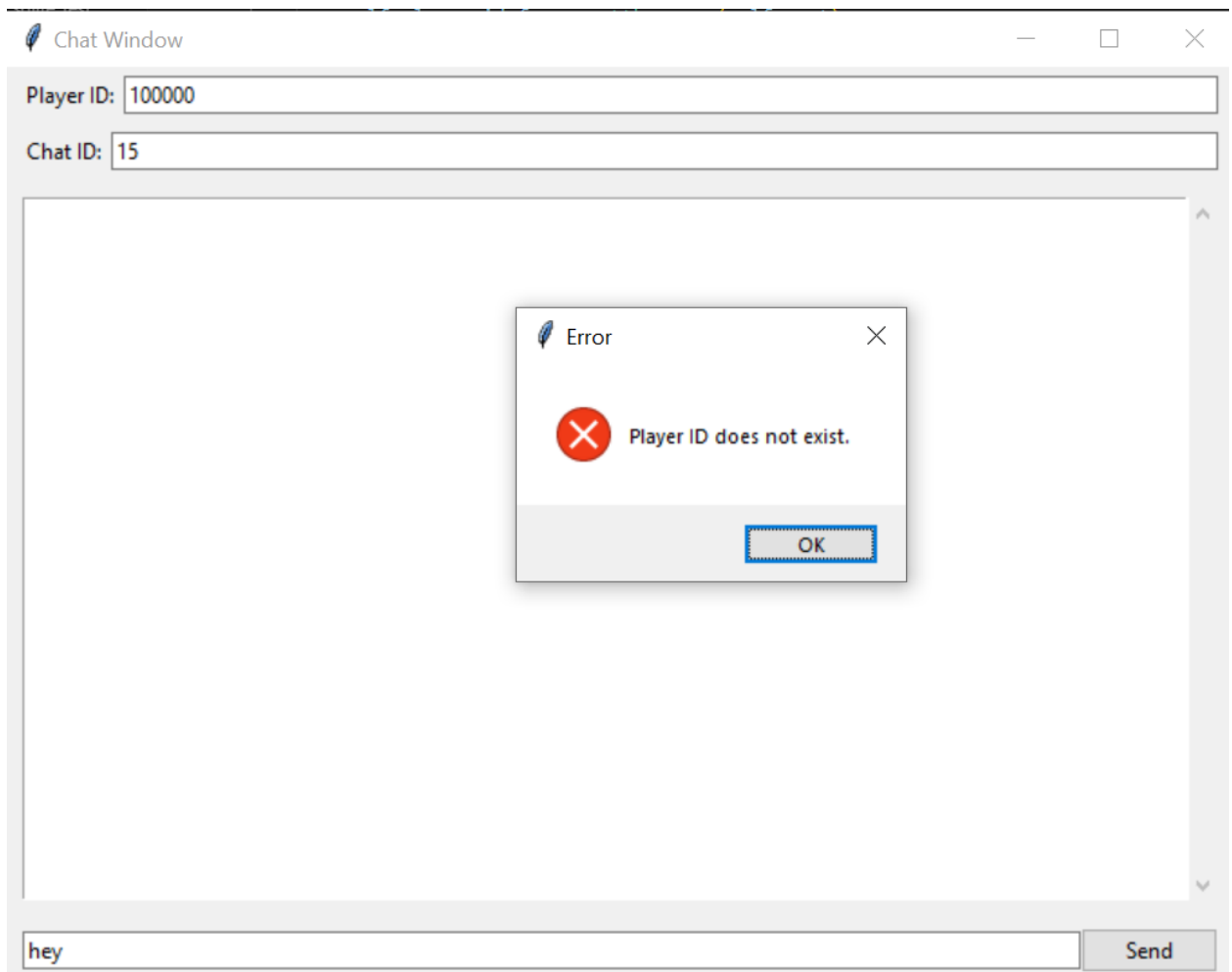
- Client Interface Creation:

To enhance user experience, we designed a simplistic yet efficient front-end interface, offering players a seamless medium to send and receive messages. The integration of the database acted as a "message-broker," recording each communication efficiently.

In culmination, Week 3 saw us evolving our system, making it not just more interactive and user-centric, but also more intricate in terms of database operations and application logic.