TUNIS BUSINESS SCHOOL
UNIVERSITY OF TUNIS

GameConnect Web API

IT-325 Web Services

Alya Barcous
BA/IT student

Tunis Business School

January 21, 2024

Table of Contents

1. Introduction and Motivation
2. Technologies and Implementation
3. Integration of a Public API
4. Security
5. Strengths and Weaknesses
6. Conclusion

Introduction and Motivation

What is GameConnect?

- **Our Vision:** A centralized hub for Tunisian gamers, providing a virtual space to connect, engage, and elevate the gaming experience
- **Purpose:** Fostering Community - Unite gamers from diverse backgrounds to form a strong, supportive community.
- **Motivation:** Enhanced Gaming Experience - Elevate the gaming journey by providing a platform that goes beyond individual games, emphasizing social connections.

Why Our Gaming Platform?

- **Tunisian Focus:** Tailored for the Tunisian gaming community, celebrating local gaming culture.
- **Inclusivity:** Welcoming players of all levels and interests, from casual gamers to esports enthusiasts.
- **Tech Innovation:** Cutting-edge features and a user-friendly interface for a seamless experience.

Technologies and Implementation

Tunis Business School

Main Technology Stack

- **FastAPI**:
  High-performance Python framework for building APIs with speed and efficiency.

- SQLA **SQLAlchemy**:
  Powerful SQL toolkit and Object-Relational Mapping (ORM) library for database interaction.

- **PostgreSQL**:
  Robust open-source relational database management system.

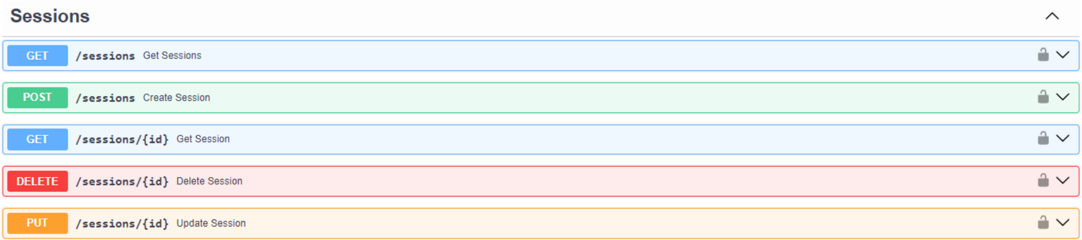
Documentation and testing tools

- **Swagger**:
  Interactive API documentation for FastAPI, allowing developers to explore and test endpoints effortlessly.

- **Insomnia**:
  Feature-rich API client for designing, testing, and debugging APIs.

Key Feature Implementations: Sessions

Session Creation – Easily create and manage gaming sessions, specifying time, availability, and desired number of participants.



Sessions

GET `/sessions` Get Sessions

POST `/sessions` Create Session

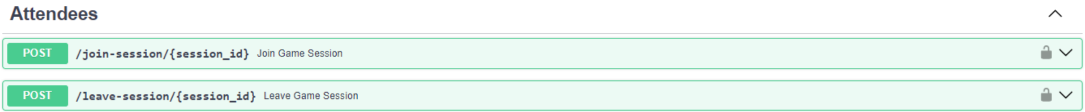
GET `/sessions/{id}` Get Session

DELETE `/sessions/{id}` Delete Session

PUT `/sessions/{id}` Update Session

Key Feature Implementations: Attendees

Community Engagement – Join or leave any session at any time, fostering dynamic community interaction



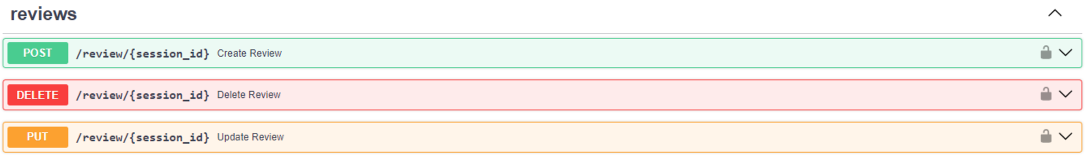
Attendees

POST `/join-session/{session_id}` Join Game Session

POST `/leave-session/{session_id}` Leave Game Session

Key Feature Implementations:Likes and Reviews

User Feedback – Integrated review system for the sessions and the overall user experience.



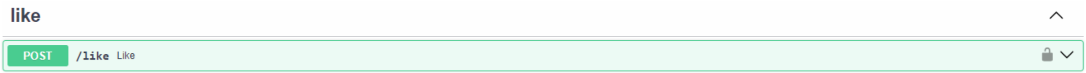
reviews

POST /review/{session_id} Create Review

DELETE /review/{session_id} Delete Review

PUT /review/{session_id} Update Review

User Engagement – Like any session and easily modify your choices as needed.

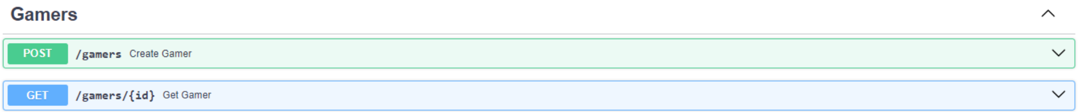


like

POST /like Like

Key Feature Implementations: User Profiles

Gamers – Create and personalize your gamer profile.



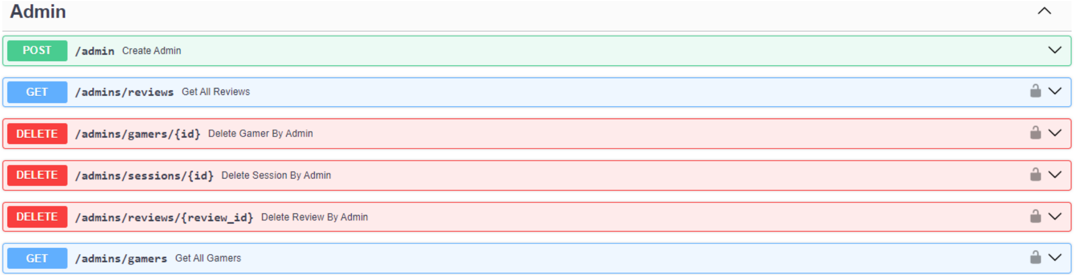
Gamers

POST /gamers Create Gamer

GET /gamers/{id} Get Gamer

Key Feature Implementations: User Profiles

Admins – Oversee and moderate gaming sessions, reviews, and user interactions.



Admin

POST `/admin` Create Admin

GET `/admins/reviews` Get All Reviews

DELETE `/admins/gamers/{id}` Delete Gamer By Admin

DELETE `/admins/sessions/{id}` Delete Session By Admin

DELETE `/admins/reviews/{review_id}` Delete Review By Admin

GET `/admins/gamers` Get All Gamers

Integration of a Public API

Tunis Business School

Discord Bot Integration

- **Bot Overview**
  Seamlessly integrate a Discord bot for community interaction.
- **Key Commands**
  - `!send`: Greeting message to the gaming community.
  - `!joke`: Humorous gaming-related jokes.
  - `!countdown`: Countdown for exciting moments.
  - `!random`: Generate a random number.
  - `!motivate`: Motivational quotes.
- **Implementation**
  - Initialize the bot with a specified prefix and intents.
  - Handle events like successful login.
  - Ensure secure token management.

Security

Security Measures

- **User Roles**
  -Users categorized as either gamers or admins.
  -Distinct privileges based on assigned role.
- **Admin Account Creation**
  -Admin accounts created exclusively with an admin key.
  -Enhances platform security and control.
- **Token-based Authorization (OAuth2)**
  -Utilizing OAuth2 for secure API access.
  -Tokens grant access to specific endpoints based on user role.
  -Ensures precise authorization for gamers and admins.

Strengths and Weaknesses

STRENGHTS

- **Centralized Hub:** the gaming community on one platform.
- **Engagement Features:** Dynamic sessions, reviews, and likes foster active user participation.
- **Security Measures:** Ensures user data protection and privacy.

WEAKNESSES

- **Scalability Planning:** Addressing potential scalability challenges.
- **Dependency on External Platforms:** Reliance on external services may introduce vulnerabilities or disruptions.
- **Feedback Iteration:** Continuous improvement based on user feedback.

Conclusion

Recap and Ideas for Improvement

Recap of Key Points:

- One-stop hub for the gaming community.
- Strong API for easy integration.
- Prioritizing user data security.

Ideas for Improvement:

- **User Profiles and Social Features:** Add user profiles and social functions.
- **Gamification Elements:** Introduce leaderboards or achievements.
- **Recommendation System:** Use AI to suggest gaming sessions based on user preferences.

Thank you for your attention

Alya Barcous
BA/IT student

Tunis Business School

January 21, 2024