

Group: A

Subject/Block: ITCC 14 B

Proposal: 1

Context

Game enthusiasts often struggle to keep track of new and upcoming video game releases due to information being scattered across numerous websites, social media platforms, and news outlets. This fragmentation makes it difficult to find accurate, consolidated, and timely information. The Video Game Bulletin (VGB) project proposes a centralized, web-based platform to serve as a streamlined and reliable hub for all video game and console release information, simplifying the discovery process for gamers.

Problems/Needs Analysis

Currently, there is no single, dedicated system that provides a comprehensive and interactive calendar for game releases. Gamers must manually browse various sources, which is inefficient and often leads to them missing out on announcements or release date changes. This creates a clear need for an integrated platform where all information on game releases—including details, platforms, and system specifications—can be stored, searched, and updated consistently, providing a one-stop resource for the gaming community.

Solution

We propose the development of a RESTful API called the **Video Game Bulletin (VGB) API**. This API will power a web application and provide structured endpoints for all core functionalities. The technical solution will involve a full-stack JavaScript framework (such as Next.js) for the front-end and a Firebase database for the back-end to ensure real-time updates and scalability.

Key API endpoints will include:

- **/releases (GET)**: To retrieve the list of upcoming and newly released games for the main calendar.
- **/games (POST)**: (Admin-only) To allow an administrator to add new game information to the database.
- **/games/{id} (GET)**: To retrieve detailed information for a specific game, including description, platforms, and specifications.
- **/games/{id} (PUT)**: (Admin-only) To allow an administrator to modify existing game information in the database.
- **/games/{id} (DELETE)**: (Admin-only) To allow an administrator to remove game information from the database.
- **/users (POST)**: To handle new user registration and create a profile.
- **/users/{userId}/favorites (POST)**: To allow a registered user to add a game to their personal favorites list.
- **/games/{id}/comments (POST)**: To allow a registered user to post a comment on a specific game's page.