

Steam Library Playlist and Recommendation Generator

1. Project Description

Our project will analyze users' Steam video game libraries and create playlists with songs from their most played games and recommend them new games. We will use [Steam OAuth](#) and [Spotify OAuth](#) for logging in. We will use [Steam's API](#) for getting game information, [Spotify's API](#) for creating playlists, and [ChatGPT's API](#) for recommendations. We will use an SQLite database to manage games played across all users, previously generated playlists, and previous recommendations.

2. Product requirements

- Goal: Create a web-interface for generating game recommendations and Spotify playlists based on someone's Steam games
- Non-goal: Make a playlist out of someone's Steam library
- Non-functional requirement 1: Security
 - Functional requirements:
 - * Use Steam and Spotify OAuth for user verification
 - * Securely store Steam, Spotify, and ChatGPT API keys on local files that are git ignored
- Non-functional requirement 2: Privacy
 - Functional requirements:
 - * Only store/share people's library information/playlists/recommendations if they consent
- Non-functional requirement 3: Repeatability
 - Functional requirements:
 - * Store people's playlists and recommendations in database
 - * Same user will get same playlists/recommendations
- Non-functional requirement 4: Public sharing
 - Functional requirements:
 - * Page listing what playlists/recommendations other users have generated/received
 - * Statistics for most played games across all users

3. Product management

- Theme: Create fun playlists and accurate game recommendations for a Steam gamer
- Epic: Website Beta
- User story 1: As a new user, I want to feel comfortable sharing my Steam and Spotify information on a random site.

- Task 1: use APIs securely for proper OAuths
 - * Ticket 1: Safely store our API keys. We need to store them in a local file and ignore them in .gitignore to ensure we don't push them to the Git repository.
 - * Ticket 2: Implement Steam and Spotify OAuths.
- Task 2: Ask user for permission to store/share their data
 - * Ticket 1: Only record their user stats or share their playlists and recommendations if they allow for it.
- User story 2: I have played a lot of games on Steam and want to listen to a variety of their soundtracks' songs.
 - Task 1: Make playlists based on users' libraries.
 - * Ticket 1: Use Steam API for library information. Get a user's most played games and ask ChatGPT for good songs from those games (the longer they've played the game, the more songs they'll get for that game).
 - * Ticket 2: Create a Spotify playlist with those songs. Gather the songs recommended by ChatGPT and compile them into a Spotify playlist using their API.
 - Task 2: Let users search playlists with songs from certain games.
 - * Ticket 1: Store previous users' playlists in a database. Store the user, the associated games, and the playlist link.
 - * Ticket 2: Display these playlists on a public sharing page.
- User story 3: I have played some games on Steam and want recommendations for new games to play.
 - Task 1: Generate recommendations based on users' libraries.
 - * Ticket 1: Use Steam API for library information. Get a user's most played games and ask ChatGPT for recommendations for similar games.
 - * Ticket 2: Display the recommended games and provide a link to their Steam store page.
 - Task 2: Let users search for most commonly shared games.
 - * Ticket 1: Store previous users' played games in a database. Store the user, their games, and the games' playtime.
 - * Ticket 2: Let users search for game overlaps. For example, a user may search for Game X, and our website will show the user the most commonly owned games by people that played Game X.