Team 9 - Check-in

Michelle Vences, Matthew Mohaupt, Waseem Sehwail, Sagar Patel, Jash Patel, Al Abraham

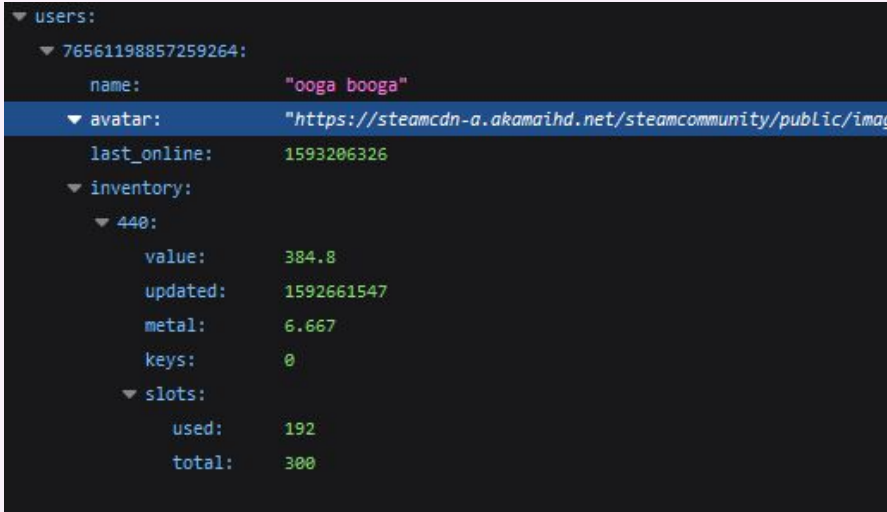
Problem (Big Idea): Do Gamers Actually Play More Games?

This project is geared towards finding if there is there a correlation between the number of games owned by a gamer and the total hours they have played, compared to gamers with fewer games. We would do this using the Steam Web API.

- Hypothesis: If a casual gamer own less games, they might play the games they do own more whereas a gamer who owns many games is not as likely to spend many hours playing certain games

Steam API:

- https://developer.valvesoftware.com/wiki/Steam_Web_API
- The API returns an object containing the named object with the result data
- Arrays are represented as an array with the name of the type of the objects in the array
- Null is represented as JSON's null



GetOwnedGames (v0001)

GetOwnedGames returns a list of games a player owns along with some playtime information, if the profile is publicly visible. Private, friends-only, and other privacy settings are not supported unless you are asking for your own personal details (ie the WebAPI key you are using is linked to the steamid you are requesting).

Example URL: http://api.steampowered.com/IPlayerService/GetOwnedGames/v0001/?key=XXXXXXXXXXXXXXXXX&steamid=76561197960434622&format=json

Arguments

- **steamid**
  - The SteamID of the account.
- **include_appinfo**
  - Include game name and logo information in the output. The default is to return appids only.
- **include_played_free_games**
  - By default, free games like Team Fortress 2 are excluded (as technically everyone owns them). If include_played_free_games is set, they will be returned if the player has played them at some point. This is the same behavior as the games list on the Steam Community.
- **format**
  - Output format. *json* (default), *xml* or *vdf*.
- **appids_filter**
  - You can optionally filter the list to a set of appids. Note that these cannot be passed as a URL parameter, instead you must use the JSON format described in Steam_Web_API#Calling_Service_interfaces. The expected input is an array of integers (in JSON: "appids_filter: [440, 500, 550]")

Result layout

- **game_count** the total number of games the user owns (including free games they've played, if include_played_free_games was passed)
- A **games** array, with the following contents (note that if "include_appinfo" was not passed in the request, only **appid**, **playtime_2weeks**, and **playtime_forever** will be returned):
  - **appid** Unique identifier for the game
  - **name** The name of the game
  - **playtime_2weeks** The total number of minutes played in the last 2 weeks
  - **playtime_forever** The total number of minutes played "on record", since Steam began tracking total playtime in early 2009.
  - **img_icon_url**, **img_logo_url** - these are the filenames of various images for the game. To construct the URL to the image, use this format: http://media.steampowered.com/steamcommunity/public/images/apps/*{appid}*/*{hash}*.jpg. For example, the TF2 logo is returned as "07385eb55b5ba974aebbe74d3c99626bda7920b8", which maps to the URL: [1]
  - **has_community_visible_stats** indicates there is a stats page with achievements or other game stats available for this game. The uniform URL for accessing this data is http://steamcommunity.com/profiles/*{steamid}*/stats/*{appid}*. For example, Robin's TF2 stats can be found at: http://steamcommunity.com/profiles/76561197960435530/stats/440. You may notice that clicking this link will actually redirect to a vanity URL like /id/robinwalker/stats/TF2

Next Steps Data: What information we want to see

Number of games owned: This can be obtained from the game list for each SteamID.

- Playtime metrics: For each game owned by a SteamID, collect the playtime (playtime_forever), which gives the total hours played.

Additional metrics:

- Average playtime per game.
- Maximum playtime in a single game.
- Number of games with zero hours played.

Solution / Analysis

Perform statistical analysis on the collected data to investigate the correlation:

- Plot a scatter plot comparing NumberOfGames vs. TotalPlaytimeHours.
- Calculate the correlation coefficient (e.g., Pearson correlation) to quantify the strength of the relationship.

We'll consider different groups, such as users with fewer games (e.g., <10 games) and users with a larger library (e.g., >50 games), to compare their playtime trends.

Discuss the implications of the results: Do gamers with more games spend more or less time on individual games? Is there a clear correlation?

Github Repository:

https://github.com/cs418-fa24/project-check-in-team-9