DS 7330 PROJECT FINAL RESEARCH PAPER

1. PROBLEM

Many video game enthusiasts face difficulty in choosing the right game specific to their preferences due to the overwhelming number of options available. Many issues are discouraging potential gamers from exploring new titles, often leading to hesitation in purchasing new games, driven by the fear of spending $70 on games they might not enjoy. (*Digital Trends* 2022).

Our project, Gamerpedia aims to address this issue by providing a comprehensive database that helps users select videogames based on their preferences. The database will feature two main functionalities:

1. Recommendation Engine: A recommendation system that suggests video games based on preselected parameters and ranks them using ChatGPT.
2. Game Encyclopedia: An information hub that provides users with key details about video games, including predicted playtime, news, popularity, release trailers, playthrough videos and top streamers.

Both features are designed to operate independently, ensuring users can either find game recommendations or access detailed information about their specific game options.

2. RELATED WORK

Our projects took inspiration from previous works done by various companies and individuals. Here is a comprehensive list of each of them.

- **Quantic foundry**: It is a web page that takes 3 of the user's preferred games as input and suggests a list of various games the user might be interested in. It also suggests the rating of each game (rated 1-15). Each item is attached to a list that redirects the user to a website that discusses extensively about the game. The suggestion varies based on the options the user selects like platforms, type of game, and release year. (Quantic Foundry, 2018)
- **IGN:** This is a webpage that displays the latest content about various media forms. The media forms include video games, animations, movies, comic books, etc. The contents can be guides, walkthroughs, news and reviews. This will be a good inspiration for showing the latest content of the game our user selects. (ign.com, 1998).
- **YouTube:** YouTube is a video database that contains and displays a huge amount of video content from different creators. It is popular as its user friendly and it's the number one source for free video content in the world. We plan on using it to display the top gameplays of the games selected by the user. (*YouTube* 2005).
- **Twitch:** This is a free interactive livestreaming service that shows various streaming contents from various streamers. It is the number one site to access streaming content around the world. We plan on using it to display to video game streamers of the game selected by the users. (*Twitch* 2002).
- **Games Finder:** This website is self-proclaimed as the number one source of self-curated video game recommendations. It recommends a list of video game recommendations, their ranking, where to buy them and their available platforms based on a video game the user selects, this will be a great inspiration for our video game recommendation feature. (GamesFinder, 2013).

3. SOLUTION METHODOLOGY

After our project proposal, we made some changes to the proposal due to lack of information on the technology, or the method not being related to what we are working on in class. A major change includes replacing the retrieval of game data via API integration for using datasets hosted on Kaggle. This was decided due to previous methodology not having enough information for us to tailor it to our specific app. Also, the fact that the new method relates to the related topic of Database Management Systems helps. We also scrapped the idea of integrating a cloud platform for our app, instead, choosing to host all the data and resources locally. This was done because the utilized data was small enough to be run locally, also this method being free helps a lot. We will still need to host the app on a cloud platform after the app has been completed,

Now that all our current updates have been discussed, here is our methodology for creating the videogame database

1. **Web Development**: Our team decided on using a web framework as the best platform for hosting the game database app. This choice was made to provide accessibility of the app to global users. Also, it helps provide the user with a sense of security when putting private information into the app. The app uses Python Django as its web framework engine, which is a Python tool that utilizes Python functions and READ/WRITE requests to generate impressive websites. The website can be hosted locally or globally via a web framework. For the app's front-end design, the app uses HTML and CSS to create an optimized, visually pleasing, and user-friendly user interface. While in the backend, the app utilizes JavaScript and Python functions to run the app behind the scenes. Django engines are also used for login and sign-in functions.

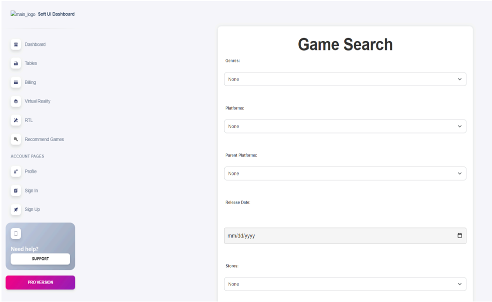
2. **Database Schema Design:** A database schema was designed in order to create our database for the app. A total of 9 files were downloaded from Kaggle. Each contains various datasets related to games. Like the games, sales, reviews, tags, scores, etc. A model of the database tables was created using 'MySQL Workbench', where similar tables and attributes were created based on the downloaded files and their columns. After the model was created, an SQL script was generated using forward engineering, where the database was declared, followed by the tables and their attributes, each with its respective datatypes. The files were uploaded to their respective tables using the 'LOAD DATA INFILE' SQL function. After all the files were uploaded to the database. These tables were combined into a new table using the 'INSERT INTO' SQL function. The data was optimized by removing duplicate row and filling the null values. The resulting table was downloaded into the app's local directory.

3. **Recommendation Algorithm Implementation:** An algorithm function was generated based on ranking on the order of importance using ChatGPT API. Due to the vast amount of missing values present within our dataset, the recommendation algorithm randomly selected game names that have corresponding columns similar to the ones selected by the user via the option menu. The total games randomly selected by the algorithm, was limited to 10 games. Then ChatGPT ranks the games based on the metacritic score available within its database. The top 5 games are selected to be recommended to the user. This algorithm provides the user an opportunity to explore new games, even if their preferences are fixed.

4. **Generating Game Encyclopedia via API Integration:** This last methodology involves redirecting the user to an information hub that provides them with key details about the user-selected game, including predicted playtime, news, popularity, release trailers, playthrough videos, and top streamers. This involves querying the game specific link for each user-selected game via API integration, then redirecting the user to the link. The API tools that were utilized for the apps are IGDB API which provides the cover image for each game, and Giant Bomb API which provides the necessary link needed to redirect the user to the GiantBomb site.

4. RESULTS

1. **Web Development:** The web app utilized python Django for its web framework. The user interface is optimized and visually pleasing. It utilizes the use of whitespace effectively with a corporate theme. Also, the navigation is foolproof where the features can easily be navigated to. The app also uses security features like login and sign-up tools to optimize user security. A page was created, where the user can select his preferences, based on different tags like platform, genre, multiplayer options, price, etc. Lastly, the GET requests work perfectly with no errors.



Game Search

Genres: None

Platforms: None

Parent Platforms: None

Release Date: mm/dd/yyyy

Stores: None

a. *Fig 1: Screenshot of the User Interface*

2. **Database Schema Design:** The database query worked perfectly. The tables were successfully merged into a new table. The resulting product was a table with 39 attributes (columns) and over 1.17 million Tuples (rows). The record was saved into a new file in the database. The size of the file was over 162 MB (166 KB).
3. **Recommendation Algorithm:** Through the use of tools such as Python and Django, A successful algorithm function was created that allowed for a decision to be made on which game to suggest for the user. For any preferences chosen by the user that yields more than one suggestion, the algorithm then kicks in to make a choice based on ChatGPT rankings,
4. **Game Encyclopedia:** Once again, tools such as Python and Django were utilized in making our Game Encyclopedia. These tools, coupled with GiantBomb's API helped to provide more information for the games that are recommended. We used IGDB API for to provide the cover URL for each recommended game. When a game is suggested on our app, you are able to click the provided link that will redirect you to GiantBomb's own page that they have for the suggested game.

5. ANALYSIS

Looking at the database from the combined files, we were able to extract some useful insight into the data that would be useful for our app. We mainly focused on 11 main attributes that we used for our app; Release_Year, platform, genre, Game_Rating, metacritic_score, UserScore, Total_critics, Total_Users, Player_info, Console, and Review. Due to page restraint we will look at some analysis.

EDA Analysis: There are over 1.17M games in the dataset, with a standard deviation of 364K games. We can gain some insights from Figures 2-5 below. There is a steady increase in the amount of games distributed from 2000 to 2018. Highlighting the rise in investment in videogames by the industry. In terns of platforms, PC has the highest fanbase, followed by playstation, and lastly Xbox. The most popular genres of videogames are the action, adventure and Role Playing (RPG) genre, reflecting a strong market demand. Lastly looking at the relationship between user and metacritic score, there seems to be a weak positive correlation between the metacritic and the user scores with some notable outliers.

Missing Value Analysis: The heatmap from Figure 6 below shows that there is a substantial amount of missing values in many columns. Which is an after effect of joining different videogame related databases together. The yellow color signifies missing values. From the heatmap below, there are significant amount of missing values in most of the columns, except Release_Year, Total_critics, and Total_Users.

Analysis of user and critics distribution: Both the users and critics have similar distribution patterns from Figure 7 below. With the patterns skewed to the right. Most of the games in the database have little to no reviews, while a couple of them have extensive attention from users and critics alike.

Correlation Analysis: From the correlation Matrix in Figure 8 below, we can see the videogames with higher user/metacritic scores tends to have more reviews, showcasing their popularity and quality. Also there is a weak correlation between user and metacritic scores, highlighting a difference in perspective among those communities.

Time-Based Trend Analysis: From the time graph in figure 9 below, we can see a steady growth in reviews increased engagement among users and critics, highlighting the the importance of reviews in the videogame ecosystem. The steady rise in user reviews suggests that the user's voices are becoming more influential, thus influencing game development and marketing strategies. Lastly, the trend reflects the growing popularity and investment of videogames in the industry with more titles being released and reviewed as time and resourced invested improves.

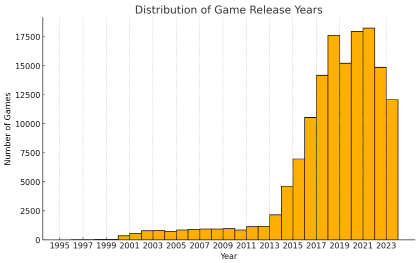


Fig 2: Distribution of videogame release years

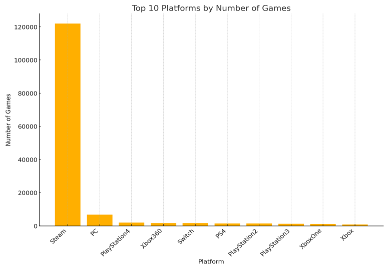


Fig 3: Top platforms by number of videogames

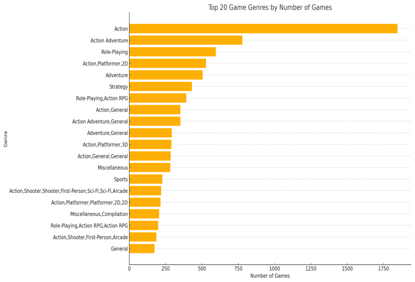
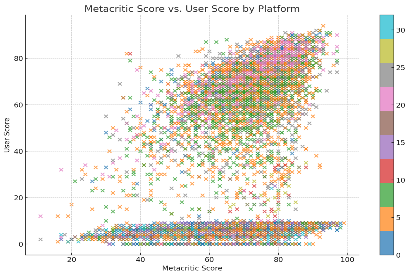
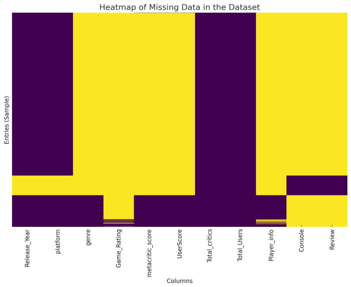


Fig 4: Top Genres by number of videogames





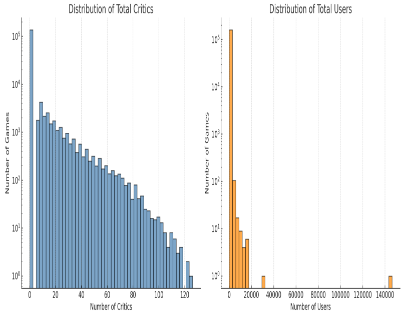
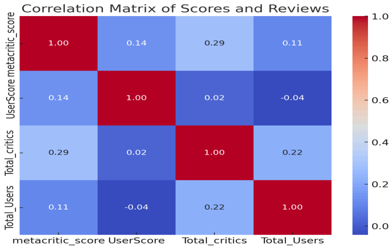


Fig 5: Metacritic score vs User Score by platform   *Fig 6: Missing value heatmap*   *Fig 7: Histograms of User count and Critic count*



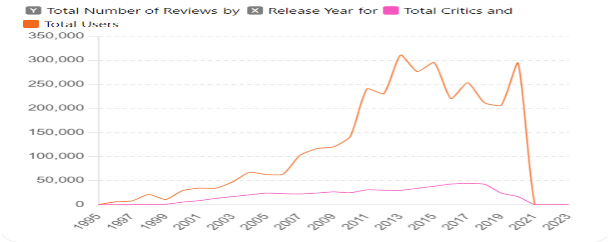


Fig 8: Correlation matrix of scores and reviews

Fig 9: Trends of total reviews over time (year)

These analyses provides useful insights unto the gaming community, showcasing valuable trends, user engagement and industry dynamic. The results of these analyses can inform us how to better improve our algorithm to better recommend games that are geared towards users based on their preferences, utilising options like scores, genres, platforms, reviews.

6. CONCLUSION

In conclusion, the Gamerpedia app aims to resolve the indecisiveness and confusion beginner gamers face when choosing a game that matches their preferences and interests. By integrating a game recommendation feature and an encyclopedia feature, we aim to create a web framework that enhances the overall gamer experience for all user by recommending games based on their preferences and providing detailed resources for each game.

Our methodology includes creating a user-friendly, website to host our app using Django, creating an optimized game database using database schema design from MySQL, utilizing an efficient, optimized recommendation algorithm with the aid of ChatGPT API, and utilizing API to provide resources on each game.

Overall, this will allow Gamerpedia to speed up gaming discovery and selection, enabling gamers to find a game perfectly suited to their unique taste. Using advanced recommendation algorithms and nested information about video games, our platform will assist gamers in finding and selecting games tailored to their individual preferences. At this stage, we look forward to a further enhancement of our tool and releasing it publicly for the gaming community.

7. CITATIONS

Quantic Foundry. (2018, January 15). *Video game recommendation engine*. Quantic Foundry. https://apps.quanticfoundry.com/recommendations/gamerprofile/videogame/

Recommending games like your favourites. Games Finder. (2013, April 14). https://gameslikefinder.com/

Twitch. (2002, July 5). https://www.twitch.tv/

Video game news, reviews, and walkthroughs. IGN. (1998, December 3). https://www.ign.com/

YouTube. (2005, April 28). YouTube. https://www.youtube.com/

Owning a video game is much harder than it should be in 2022. (2022, August 11). Digital Trends. https://www.digitaltrends.com/gaming/game-ownership-digital-media-players-weigh