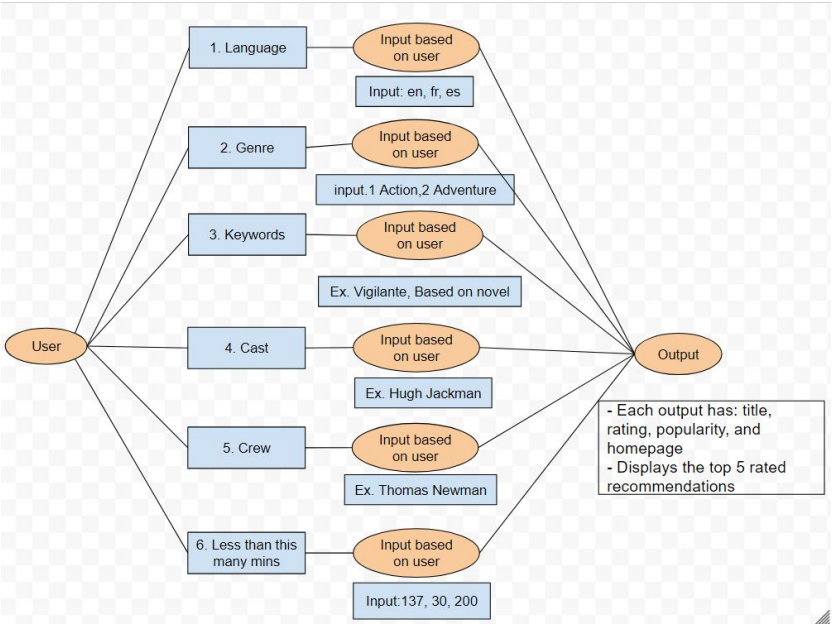
REC 2020 Report

The Design Problem and Solution

The problem that our team was tackling was to create a recommendation search engine that would take user information as an input and output a list of the top 5 movies that the user would potentially enjoy. The stakeholders for our Netflux recommendation search engine are employees and company owners, more specifically the project managers/engineers at Netflux who are maintaining and updating the engine, investors, and partners that are invested in Netlux's growth, and finally, the customers using the Netflux search engine. As Netflux increases in subscribers every day, it is very important that all the subscribers are satisfied with the services Netflux provides. For this reason, we implemented a top 5 movie recommendation engine depending on the user's preferences. The engine takes the user's inputs and filters the database to see what the top 5 movies for their preferences are. This would help the customers find movies favorable to their preferences and as a result, be more satisfied with the services provided by Netflux. The target audience for Netflux would be all age groups because of its large catalog of movies. It can be streamed on all platforms so it should be accessible to everyone.

High-Level Overview of Design Solution

The problem required us to build an application that would give users the ability to sort the top 5 movies based on their input. Our program allowed the users to choose a filter they want to sort the movies based on. Example of this is language, Actor, Genre. Upon selecting one of the options they further imputed what they wanted to choose in that category. This gave them the top 5 movies based on their selection ordered by Rating. As the output, the user will see what they sorted by in one column then the name of the movie, average voting, popularity, and homepage. This way the user can see all the main information about the movie and can directly go to the movie's homepage to learn more information. The average rating is listed from highest to lowest. This can be changed if needed in the future to order based on other selections if needed. In the following diagram, you can see how the user will navigate through the program and how the program is user friendly and provides the best options to get the top 5 movie recommendations.



Design Process, Management process, and Development process

When approaching this project our group went through a design process where we decided what we were going to first decide the inputs and outputs. It was decided that the user would get 6 options to sort the list of movies and as an output, they would be given the important information. As the input, the user had the opportunity to choose between what language the movie was in, what genre the movie should be, some keywords used to describe the movie, a cast member in the movie (i.e Actor), a crew member in the movie (i.e Director) and lastly the time limit for the movie. After the program has sorted and has the recommendations it would then display the information in different columns. The columns are organized to first display the chosen sorting option, the name of the movie, the rating of the movie, the popularity of the movie, and lastly a link to the movie for more information. Our team decided that we would use the first 30 minutes to organize the required material and plan and design what needed to be done. At the same time download the needed programs to run the code. We then gave ourselves one hour for the writing of the report and the presentation. This gave us enough time to then build and test our program to make sure that everything worked. We organized our work and shared the code via GitHub therefore we all had the opportunity to work on the development process and contribute equally. For the development process, we went through our main problem of connecting the CSV file to our python code. We then used different resources linked at the bottom to further learn and guide us in the usage of Python to then successfully build and execute our code.

Additional Features

Our program makes a top 5 list depending on the users' preference for one of the categories in the database(eg.Genres or Languages or Cast). If we had more time to work on this, we would probably make a better search option to filter more than one category from our data. This would fit the customer's preferences better and they would have more than 1 choice for categories. We would also provide the user with the output using a CSV file, this could not be done with the given time constraints. The first mistake we have is that the user is unable to go through the recommendation search engine more than once. They will have to run the program once every time to get a list of movie recommendations. To fix this, we would create a loop to allow users to go through the algorithm as many times as they want. We would also ensure that the variables are overwritable so every time the user runs the algorithm, it can run with no problem.

Core functions of Software/Program's Components (Development perspective)

Our program uses the panda's library. The explode function was used to list-like rows into indexed values. We took the values given in the CSV files and converted them to indexed values. Then the dropna function was used to drop any null values when we created the index values. The loc function was used to group the values by values/conditions. In other words, we located the rows that matched the user input and outputted whatever matched. The unique function was used to see all the distinct values under each column. The sort function was used to sort the final values into ascending order. The apply and eval function was used to use single values and on specific columns.

Installation

Pandas library needs to be installed and python version 3.8. Python compiler needed as input and output is done through the terminal.

References:

Intro to Python:
Python while Loop Statements. (n.d.). Retrieved November 28, 2020, from https://www.tutorialspoint.com/python/python_while_loop.htm
Beginner Guide. (n.d.). Retrieved November 28, 2020, from https://wiki.python.org/moin/BeginnersGuide/Programmers
ABHISHEK TIWARI 13Check out this Author's contributed articles., 13, A., & Check out this Author's contributed articles. (2020, August 05). Taking input in Python. Retrieved November 28, 2020, from https://www.geeksforgeeks.org/taking-input-in-python/
C. (n.d.). Python Cheat Sheet. Retrieved November 28, 2020, from https://blog.finxter.com/python-cheat-sheet/

Research Material:
Gspmoreira. (2019, December 08). Recommender Systems in Python 101. Retrieved November 28, 2020, from https://www.kaggle.com/gspmoreira/recommender-systems-in-python-101
Malik, U. (n.d.). Creating a Simple Recommender System in Python using Pandas. Retrieved November 28, 2020, from https://stackabuse.com/creating-a-simple-recommender-system-in-python-using-pandas/
Real Python. (2020, November 26). Build a Recommendation Engine With Collaborative Filtering. Retrieved November 28, 2020, from https://realpython.com/build-recommendation-engine-collaborative-filtering/

Import Pandas Code:
Data to Fish. (n.d.). Retrieved November 28, 2020, from https://datatofish.com/import-csv-file-python-using-pandas/

Code Fixes Research:
HannanHannan 93022 gold badges1313 silver badges2828 bronze badges, & Jppjpp 127k2424 gold badges163163 silver badges214214 bronze badges. (1967, July 01). Pandas: Sort value and ignore 0. Retrieved November 28, 2020, from https://stackoverflow.com/questions/50593621/pandas-sort-value-and-ignore-0

Pandas Functions:

Python Help. (n.d.). Retrieved November 28, 2020, from https://pandas.pydata.org/