

Movies Hangman Report

By Maurely Acosta & Nicolette Celli

INDEX

TABLE OF CONTENTS	PAGE NUMBER
INTRODUCTION	3
PROBLEM STATEMENT	3
METHODOLOGY	4
RESULTS	5
ANALYSIS	6
REFERENCES	7

1. INTRODUCTION

This project consisted of programming a multi-player game with a chat room using Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). The game that was implemented was Hangman but with a twist. A SQLite movie database from Kaggle.com was used to generate random movie titles for the players to guess.

Additional studying involved further knowledge in python (version 3) and network programming. In order to meet these requirements the official python documentation was read and online tutorials were used.

The final product for this project includes a Hangman game with a graphical user interface (GUI) using the TkInter package. To implement the multiplayer functionality and chat room, TCP and UDP was implemented. Lastly, a SQLite movies database was used to generate the movie titles.

2. PROBLEM STATEMENT

The project involves creating a multiplayer game with a chatroom. The game should have centralized server to manage the multi-user/multi-player environment using TCP and UDP. The game should be able to support at least 4-10 users with a recommended of 10 users with a GUI. For extra credit add SSL or encryption in the chat room communication.

3. METHODOLOGY

The project was tackled by first programming the business logic of the Movies Hangman while simultaneously creating the GUI while also testing for any bugs. Once that portion of the project was completed, the next step was to create a chat room program.

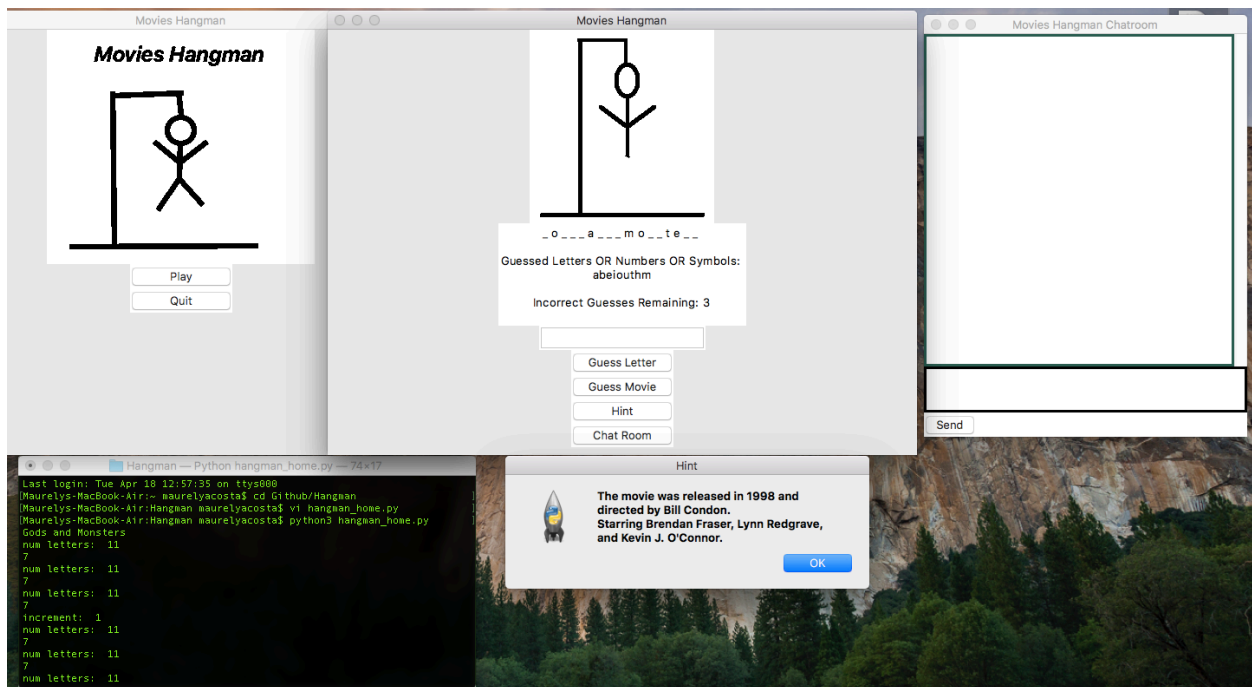
The chatroom was initially created using terminal input using TCP where more than one user could message one another from different computers. After, the chatroom functionality was migrated to part of the Hangman GUI.

The multiplayer functionality was done last using TCP. Since, Movies Hangman is not normally played in groups. It was established that there would be a limit of 2 groups since there are not so many letters in the alphabet to guess and the movie titles only contained a small portion of distinct letters from the alphabet.

4. RESULTS

The project was tested on the class server: cnt4713.cs.fiu.edu, and a Macintosh and Windows computer.

Below is the picture of the Movies Hangman on a Macintosh computer:



5. ANALYSIS

The project was rewarding since there was autonomy in picking the game to implement as well as whom to partner with. It also further knowledge in network programming and the python (version 3) language.

The approach to the project was to gradually built to the most difficult requirements. There was issues with some test cases of the Movies Hangman's business logic. The most difficult issue we encountered was implementing the multiplayer functionality since it required integrating the flow between the Movies Hangman and the chat room and by the time we started to implement it there was little time to improve on the multi-player functionality.

REFERENCES

(n.d.). Retrieved April 18, 2017, from <https://www.kaggle.com/deepmatrix/imdb-5000-movie-dataset>

Kurose, James F., and Keith W. Ross. Computer networking: A Top-Down Approach. Boston: Pearson, 2013. Print.

Postel, J., and J. Reynolds. "FILE TRANSFER PROTOCOL (FTP)." Internet Engineering Task Force (IETF). Internet Engineering Task Force (IETF), n.d. Web. 12 Mar. 2017.

Tutorialspoint.com. "Python Networking Programming." www.tutorialspoint.com. N.p., n.d. Web. 12 Mar. 2017.