

CCPS 406 Project: Text Adventure

Architecture Diagram

Group members:

- Aden Hersi (Aden.hersi@ryerson.ca)
- Charif Sukkar (charif.sukkar@ryerson.ca)
- Ervin Demnushaj (ervin.demnushaj@ryerson.ca)
- Chris Fontein (cfontein@ryerson.ca)
- Jeessoo Kim (j17kim@ryerson.ca)

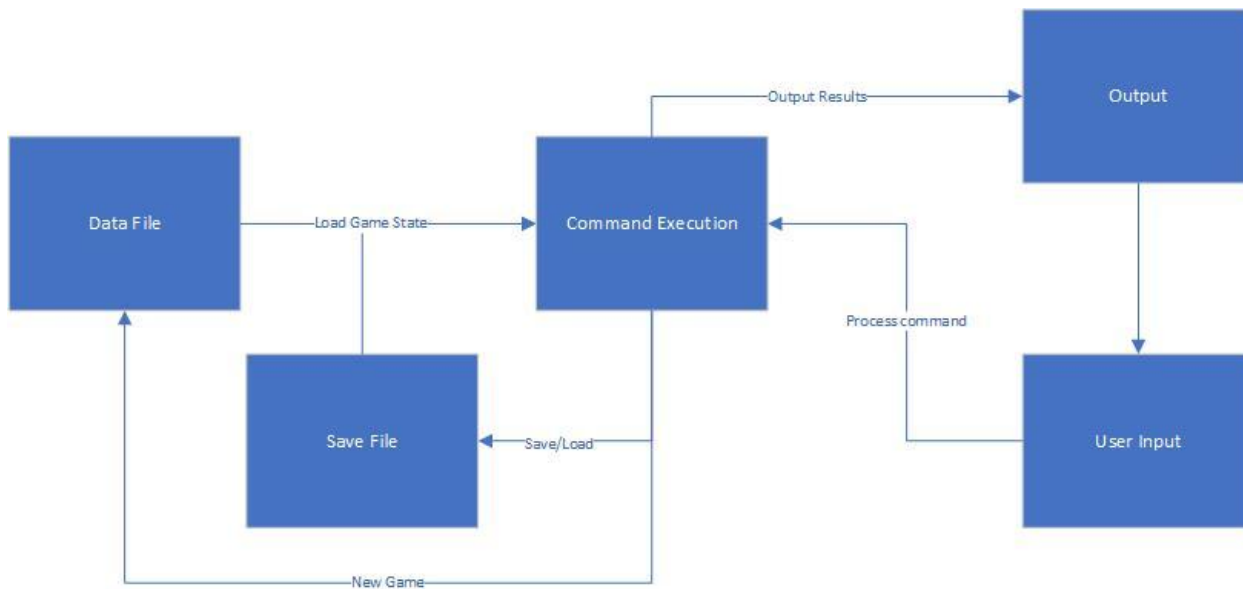
Instructor: Ilkka Kokkarinen

Subject: CCPS 406 Introduction to Software Engineering

School: Toronto Metropolitan University

Date: May 22, 2022

[Figure1] Architecture diagram for the entire system



The architecture diagram [Figure 1] shows the outline of the system and the connections between the parts of the system. This is to show high-level system descriptions.

The responsibilities of each part are as follows.

- 1) Data File (Read only)
This is default data. Data file is read by Command execution initially. It is used to set the state of a new game. Loaded games will read from the save file.
File format: YAML
- 2) Save File (Read/Write)
Save file is a written out version of the data files based on the current state of the game world. When a save command is given then it is written/overwritten. It can then be read to recreate the state of the game at save.
File format:YAML
- 3) Command Execution
This part controls the Interaction between user interface (user input and output) and the data.The command execution is reading the data from the data files, either data or save files, and the user inputs from the standard input. Then, depending on a scenario or algorithms/randomness, this will create the output dynamically.
Programming language: Python
- 4) User Input
User input is the standard input where a user types in their command as a player in the game. At the same time, User input is considering the output from the command execution which makes the game more dynamic depending on both user input and program.
- 5) Output
Output is the standard output where a user can read the command execution and also decide what to do as a player. This output is to print the output of command execution. Accordingly, the output is to provide a user/a player the current status of the game and to ask the next action to take.