CSC207 Project Report

Gursewak Sandhu

Demo Link:
https://drive.google.com/file/d/1wd9fRtqjCtMGamw967ZcDv4qBPJFsnfz/view?usp=sharing

User Stories Completed: 4

[DEV-1] parseQuestions.
This user story was created so that users can input their own questions to be used. parseQuestion() method was used to read .txt files and create an object to store them. The information stored includes question number, topic, question hint, answer, and whether the user answered the question already. A Question class was created which stored the information that needed to be hidden such as the answer in a private variable and information that needed to be accessed elsewhere in a public variable such as the question being asked. Outside of basic getter's, getQuestionHint was added which uses a queue to sort through available hints, and check answer was added to compare the given answer to the one stored within the private variable.

[DEV-2] QuestionTracker
This user story was created so player can keep track of their progress so that they can improve how well they do. A QuestionTracker class was a created with a singleton design pattern in mind, therefore this class is only instantiated once and that is done in the AdventureGame class. This class keeps track of all stats such as questions correct, attempted, hints used, etc. It has 2 main public methods, addAttempt() which records whether you got a question right in the QuestionTracker, and createReport() which prints out a report containing all the relevant stats.

[DEV-3] Skip and Hint command
This user story was created so that users that are stuck on a question can progress by skipping or getting a hint. getHint() cycles through a Queue of hints while skip marks a question as answered so the user may continue.

[DEV-4] Text To Speech
This user story was designed with accessibility in mind as users can now hear the room description, question and available commands. The textReader() method reads out any given string allowing for any written text to be made more accessible. This method was implemented using the FreeTTS Library. Room Descriptions and Questions are automatically read while commands are read when player inputs “commands” into the text field.