Victor Zuniga Salgado

CS499

Professor Bryant

11/29/24

Enhancement Three: Databases

For Enhancement Three: Databases I chose to use a text-based game titled “Battle at Mystery Manor Text Adventure Game” that I created for my final project in my IT-140 Intro to Scripting class. IT-140 was the first class that I took after transferring to Southern New Hampshire University. Prior to this, I had not been in college for two years. There was a lot of information to take in, but I enjoyed what I learned as time went on. Python was the first programming language that I got to use, and it’s remained my favorite programming language this whole time. The text-based game is an adventure-based game that contains different rooms that each contain an item. Players will need to collect every single item prior to reaching the final room to defeat the boss and win the game.

The ”Battle at Mystery Manor Text Adventure Game” is an excellent artifact to include in my ePortfolio because it highlights several key aspects of software development that showcases my creativity, problem-solving abilities, and a great understanding of programming concepts. The artifact demonstrates proficiency in Python such as the use of functions, dictionaries, loops, and conditionals. Demonstrates the ability to build a user-focused application with an emphasis in engagement and usability. It also demonstrates my ability to think creatively and solve problems effectively by designing a room-based game world that has specific rules and objectives.

Enhancements were made to the original artifact to demonstrate a well-rounded use of a database. Json was imported to provide functionality for reading and writing high scores to a JSON file which would act as a simple database. JSON stands for JavaScript Object Notation. It stores data as text and is supported by most programming languages. Global variables include the use of a (highscore\_file) which specifies the name of the JSON file used to store high scores. High score management was implemented through the introduction of three new defined functions. First, the (load\_highscores()) def function reads high scores from the “highscores.json” file but returns an empty list if the file doesn’t exist. Next, the (save\_highscore(player\_name, score)) def function adds a new score, sorts the list by the number of moves, and writes the top 10 entries to the “highscores.json” file. Finally, the (display\_highscores()) will load and print the high score leaderboard, displaying up to 10 scores. Users will need to input “highscores” as their action to view the high scores when they run the code. Highscores are saved when the player wins the game. It will prompt them to enter their name and then their player’s score (moves made) will be saved and displayed alongside the top 10 scores. Modular design of the distinct classes (Room, RoomGraph) reduce the likelihood of introducing vulnerabilities due to complex, interdependent code. It ensures that potential vulnerabilities are easier to detect and patch. It also helps with applying security controls like data validation. The (Room) class tracks an item’s history using a deque, which provides an audit trail. It can be used to trace activities and identify malicious or unintended behavior patterns.

The course outcomes that the enhancement aligned with were: Employ strategies for building collaborative environments that enable diverse audiences to support organizational decision-making in the field of computer science. Demonstrate an ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals. Develop a security mindset that anticipates adversarial exploits in software architecture and designs to expose potential vulnerabilities, mitigate design flaws, and ensure privacy and enhanced security of data and resources. Design, develop, and deliver professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts. After reviewing the enhancements that were made I’m certain that the course outcomes that were planned were met.

Enhancing and modifying this artifact with a database was the most challenging enhancement out of the three. When I initially made the artifact in my IT-140 class I didn’t think at time to implement a database to save the players score as I didn’t know much about databases at the time. Implementing the correct defined functions took a while, as there were times where the game didn’t prompt me to enter my name to save my score. Another challenger that I encountered was locating where the “highscores.db” and “highscores.json” file were saved in my computer. I will include a screenshot of where mine were saved, so that others could use it in case their having issues. It was interesting to see how much enhancements can be made to an artifact as more knowledge and skills are gained. The enhancements made laid the foundation necessary to further improve on the artifact as one may choose to do so.

A screenshot of a computer

Description automatically generated