From Concept to Code: Shadows of the Forsaken A Developer's Perspective

Course Code & Section: CSE 115.2
Project Group 1
Names & IDs
Md. Fahim Hossain, Samia Zaman, Sidratul Muntahara Audhira, Md. Lutful Alam and Riazul Jannat
2512847642, 2511026642, 2513009642, 2412032042, 2512244042

***Abstract*—**The purpose of the project is to design and implement a text-based adventure game titled "Shadows of the Forsaken". The game allows the player to participate in a chase that presents a multitude of choices and realities. It enables the player to take specific actions, helping him to navigate through various paths, each with a different outcome. The core purpose of the game is to entertain the user while providing him with different onerous and contemplative tasks. The game is built in C and utilizes conditional branching with user interactivity as the primary means of advancement within the game.This report examines various motivation sets, including their categories and roles, the design of motivation for implementation, and possible future adjustments to gamification to enhance user engagement..

Keywords: text-based adventure game, game design, interactive narrative, C programming, user choice.

I. Introduction

Every decision bears grave consequences. "Shadows of the Forsaken," a blend of horror and adventure, immerses players in a chilling, text-based thriller where every choice could lead to their doom. As you explore the hidden depths of the woods, it soon becomes evident that a sinister force is drawing you towards a dreadful mansion. No matter how much you try to escape, you will find yourself in this abandoned cursed place, as if the forest has a mind. The console commands are basic, but with every input, gruesome discoveries await, reflecting a story that unfolds in two simultaneous directions, navigating between hope and despair, pondering if escape is even attainable.

II. Methodology

This text-based interactive horror game is structured around a modular, function-driven methodology showing structured storytelling, efficient code organization and smooth user interaction.

A. Modular Programming Structure:

The game itself is separated out into a function based architecture, where each major component—menu navigation, story progression, input handling— is wrapped in a function. It helps with code reusability, readability, and scalability.

1. **Menu:** Manages navigation
2. on between start game, credit information, and quit options.
3. **Story Modules:** Each scene or decision point is implemented as a separate function, allowing dynamic branching.
4. **Dynamic Interface:** The menu interface is dynamically created using text file inputs, enabling easy updates and modifications to the game content.

B. Dynamic Storytelling & Decision Trees:

There are branching paths in the game (if-else logic) that determine the direction the story takes according to player choices

1. **Key aspects include**:
   The handleDecision() method also recursively prompts players until a valid selection is entered, ensuring seamless gameplay.
2. **Progression by Way of the State:**
   Every choice leads into another event; no one story act follows another in a linear manner.It compels players to take multiple playthroughs just to see what happens when you make different decisions.

C. Checking Input & Handling Errors:

A dedicated function (get_choice) ensures that players only enter valid numeric inputs, reducing the risk of unexpected behavior. The implementation includes:

1. **Input Buffer Clearing:** Eliminates unwanted residual input.
2. **Recursion for Invalid Inputs:** Ensures users remain within the expected selection

range.

D. Game Flow & Exit Handling:

1. The game continues to execute in a loop until some exit condition is true.
2. The exit routine contains a confirmation prompt that allows you to avoid inadvertently terminating your game.
3. If the player chooses to restart, the program seamlessly returns to the main menu without requiring a manual restart

Ⅲ. Game Design & Code Architecture

A. Core Components:

1. **Game Logic:** Implements functions to direct the player down various storylines influenced by their decisions.
2. **Getting User Input:** The get_choice() function handles the input from the player.
   Definite input leads the player to a definite direction. Such as-
   - **i. Start New Game*:***
     Leads the player to a prologue that sets up all story elements, allowing them to choose from different game areas, which might include a silent room, echoing hall, hidden doorway etc. and start with the game.
   - **ii. Credit**:
     Directs the player to a section showing the team members involved in developing and designing the game
   - **iii. Exit:**
     The player is enabled to exit the game screen.

   The players are made to enter such different choices throughout the storyline, sewing up an interactive ambiance, to keep the gamer engaged to the game.
3. **Game Events are Modularized:** Each game event new_game(), abandoned_mansion() are modularized in functions for better readability.
4. **Handling Game Over:** The function game_over() is called when the player loses, for restarting the game.

B. Flow of Execution:

1. **Main Menu:** User selects an option.
2. **Gameplay StartsAll:** The choices you make determine the path of your story.
3. **Conditional Paths:** Functions are executed depending on decisions.
4. **Game Over or Progress:** Player loses or progresses further into the narrative.

IV. Future Improvements

1. Introducing player profiles.
2. Feature of saving the game and loading it afterwards.
3. Add some random events to increase diversity and interest in the game..
4. Improve graphics design (GUI) to experience the game world.
5. Inventory system for collectables
6. Multiple endings .

V. Conclusion

“Shadows of the Forsaken” is not just a game, but a journey into terror, where each step unveils mint trepidation. This text-based dreadful adventure plunges players into an accursed forest, where each decision leads them deeper into a world filled with fear. Each boundary, the consequences resulting from each preference, and the glimpse of the imperceptible entities keep the players in a rush of excitement blended with anxiety. “Shadows of the Forsaken” corroborates that fear doesn’t lie in the dark only, but also aligns with the narratives through the choices we make, consciously , taking us deeper into the experience.

VI. References

- C Functions. (n.d.). Retrieved 2024 from GEEKSFORGEEKS: https://www.geeksforgeeks.org/C-FUNCTIONS/?REF=SHM
- C stdlib. (n.d.). Retrieved 2024 from W3Schools: https://www.w3schools.com/c/c_ref_stdlib.php
- GEEKSFORGEEKS. (n.d.). From getchar function in C: https://www.geeksforgeeks.org/getchar-function-in-c/
- W3Schools. (n.d.). Retrieved 2024 from "C Reference": https://www.w3schools.com/c/index.php
- W3Schools. (n.d.). Retrieved 2024 from C string: https://www.w3schools.com/c/c_ref_string.php