**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”. It allows the player to participate in a chase that presents a multitude of choices and realities. The game allows the player to take specific actions that will help him navigate through various paths, each with a different outcome. It aims to entertain the user while providing some challenging thinking tasks. The game is produced in C and utilizes conditional branching and user interactivity as the primary means of advancing in the game. This report studies the sets of motivation, their types and functions, the motivation design of the implementation, and potential future modifications to increase user participation.

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

## **Introduction**

Every decision bears grave consequences. Shadows of the Forsaken blends horror and adventure to immerse players in a chilling, text-based thriller. Shadows of the Forsaken" is a text-based horror adventure game where every choice you make could lead to your doom. As you explore the underbelly of the forest, it soon becomes evident that a sinister force is drawing you towards a mansion. No matter how much you try to escape, you will find yourself in this abandoned cursed place, as if the forest itself has a mind of its own. The console commands are basic, but with every input, gruesome discoveries await. A story that is going in both directions at once, between hope and despair, asking whether escape is possible, anyway.

## **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.

1. ***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 between start game, credit information, and quit options.
2. **Story Modules:** Each scene or decision point is implemented as a separate function, allowing dynamic branching..
3. **Dynamic UI:** The menu interface is dynamically created using text file inputs, enabling easy updates and modifications to the game content.
4. ***Dynamic Storytelling & Decision Trees:***
5. ***Input Validation & Error Handling:***
6. ***Game Flow & Exit Handling:***

## **Ⅲ. Game Design**

The first screen you arrive at is a welcome screen, allowing you either to start a new game, view credits, or exit. The players enter their choices numbers to interact with the game.

1. ***Main Menu***

Players get options to either start the game,  view credits, or exit.

1. ***New Game***

Players choose a new game which includes a prologue that sets up all story elements, and then they choose from different game areas, which might include a silent room, echoing hall and hidden doorway etc.

1. ***Credits***

This section shows the team members who were involved in developing and designing the game.

## **Ⅳ. Code Architecture**

***A. Core Components:***

1. **Main Menu:** Gives options such as New Game, Credits & Exit. It also reacts to the user's input.
2. **Game Logic:** Implements functions to direct the player down various story lines influenced by their decisions.
3. **Getting User Input:** The get\_choice() function handles the input from the player.
4. **Game Events are Modularized:** Each game event new\_game(), abandoned\_mansion() are modularized in functions for better readability.
5. **Handling Game Over:** The function game\_over() is called when the player loses, for restarting the game.
6. **Credits:** The function credits() shows a list of contributors and allows the user to go back to the main .

***B. Flow of Execution:***

1. **Main Menu:** User selects an option.
2. **Game Play:** 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.

***C. Future Improvements:***

1. Include the functionality of creating and searching player profiles in the system.
2. Add some random events to increase diversity and interest in the game..
3. Improve graphics design and text integration to increase storytelling.
4. This approach offers modularity, efficient debugging, and scalable expansion.

**V. Flowchart**

-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**ⅤI. Result**

**--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**ⅥI. 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 each imperceptible entity 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.