Syed Amaan (085)
Muhammad Daniyal (053)
Malik Siawish (025)

Computer Programming

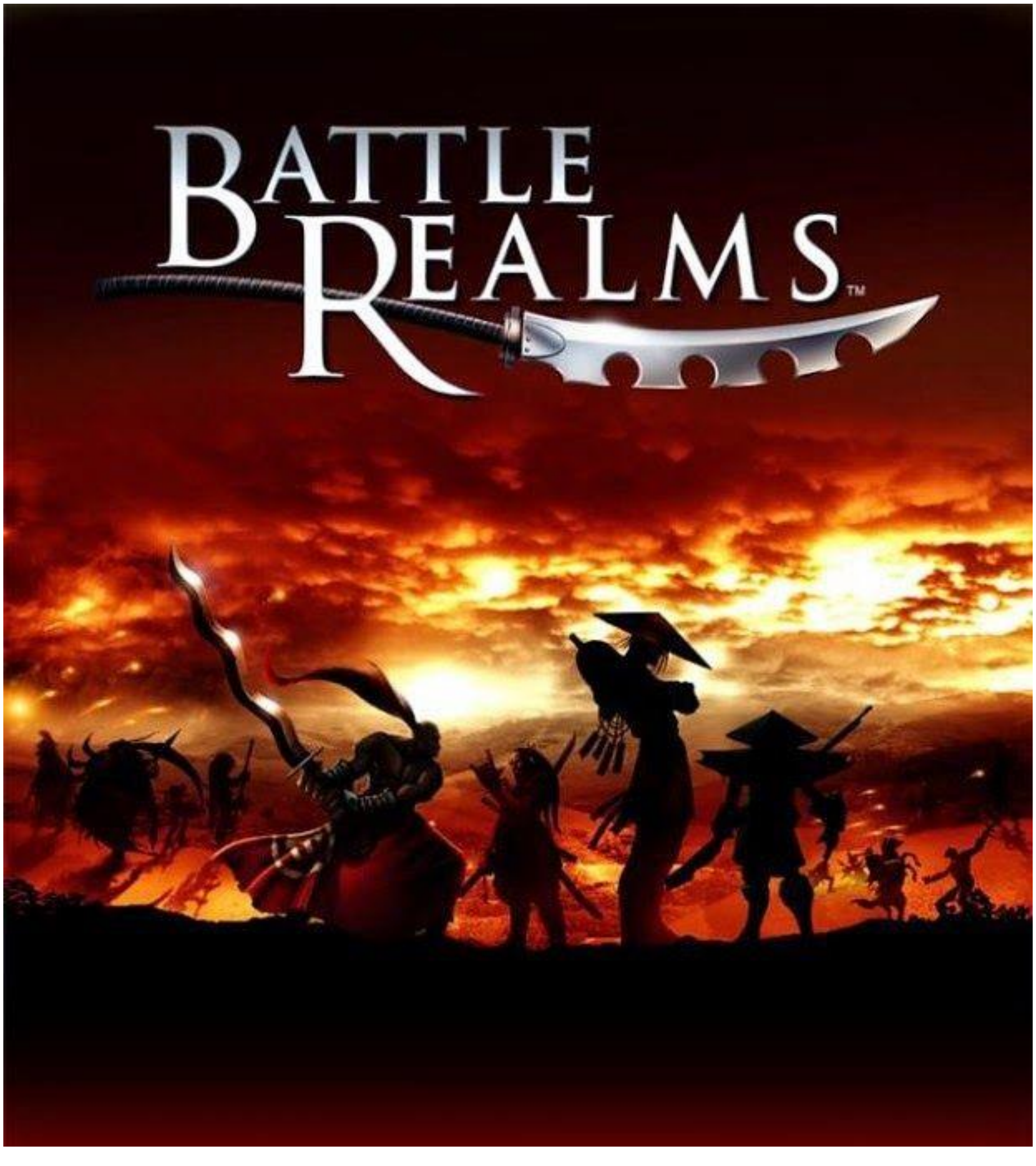
Dr. Raja Suleiman

Bahria University

Final Project

Battle Realms

January 2, 2024


BATTLE REALMS™

Battle Realms

Introduction:

The Battle Realms project is a text-based adventure game implemented in computer programming. The game is designed to provide an interactive and engaging experience for players through a series of missions, combat phases, character progression, and storytelling elements.

Algorithm:

Step 1: Initialization

1.1. Initialize variables: Cchoice, qchoice, qcount, randcount.

1.2. Create structures:

Player with attributes: name, clas, health, xp, level, activity, h_full, loot.

Quest with attributes: info, Difficulty, XPreward, Treasure, enemy.

1.3. Seed the random number generator using srand(time(0)).

1.4. Collect player information:

Get player's name.

Choose a class (Sage, Phoenix, or Omen).

Initialize player attributes (xp, health, level, activity, loot).

Step 2: Game Loop

2.1. Display introductory storyline based on the player's chosen class.

2.2. Display character information (name, class, xp, health, level, activity, loot).

2.3. Display the game menu:

Begin Adventure (mission).

Exit the game.

2.4. Accept the player's choice from the menu.

2.5. Execute corresponding actions based on the player's choice:

If Begin Adventure is chosen:

Proceed to the mission.

If Exit the game is chosen:

Display a farewell message and terminate the program.

Step 3: Mission Initialization

3.1. Prepare quest data based on the player's chosen class:

Define quests for Sage, Phoenix, and Omen classes.

3.2. Display available quests based on the player's class.

3.3. Accept the player's choice of quest.

3.4. Fetch details of the selected quest.

Step 4: Combat Phase

4.1. Initialize combat variables:

Set enemy health based on quest difficulty.

Set maximum enemy health.

Initialize player's strike, enemy hit, and random count.

4.2. Start combat loop until player or enemy health reaches zero or the enemy escapes:

Display player and enemy health.

Present attack options for the player.

Accept the player's attack choice.

4.3. Calculate and handle damage:

Apply player's attack damage to the enemy.

If the enemy doesn't escape, calculate and apply enemy's attack damage to the player.

Update player and enemy health accordingly.

4.4. Check for victory or defeat

If enemy health reaches zero or the enemy escapes:

Grant XP reward, update player's loot, and display victory message.

Check for level up conditions and handle level progression.

If player health reaches zero:

Deduct XP, handle potential XP loss, update player's activity, and display defeat message.

Step 5: Level Up

5.1. If the player's XP reaches a certain threshold:

Increment player's level.

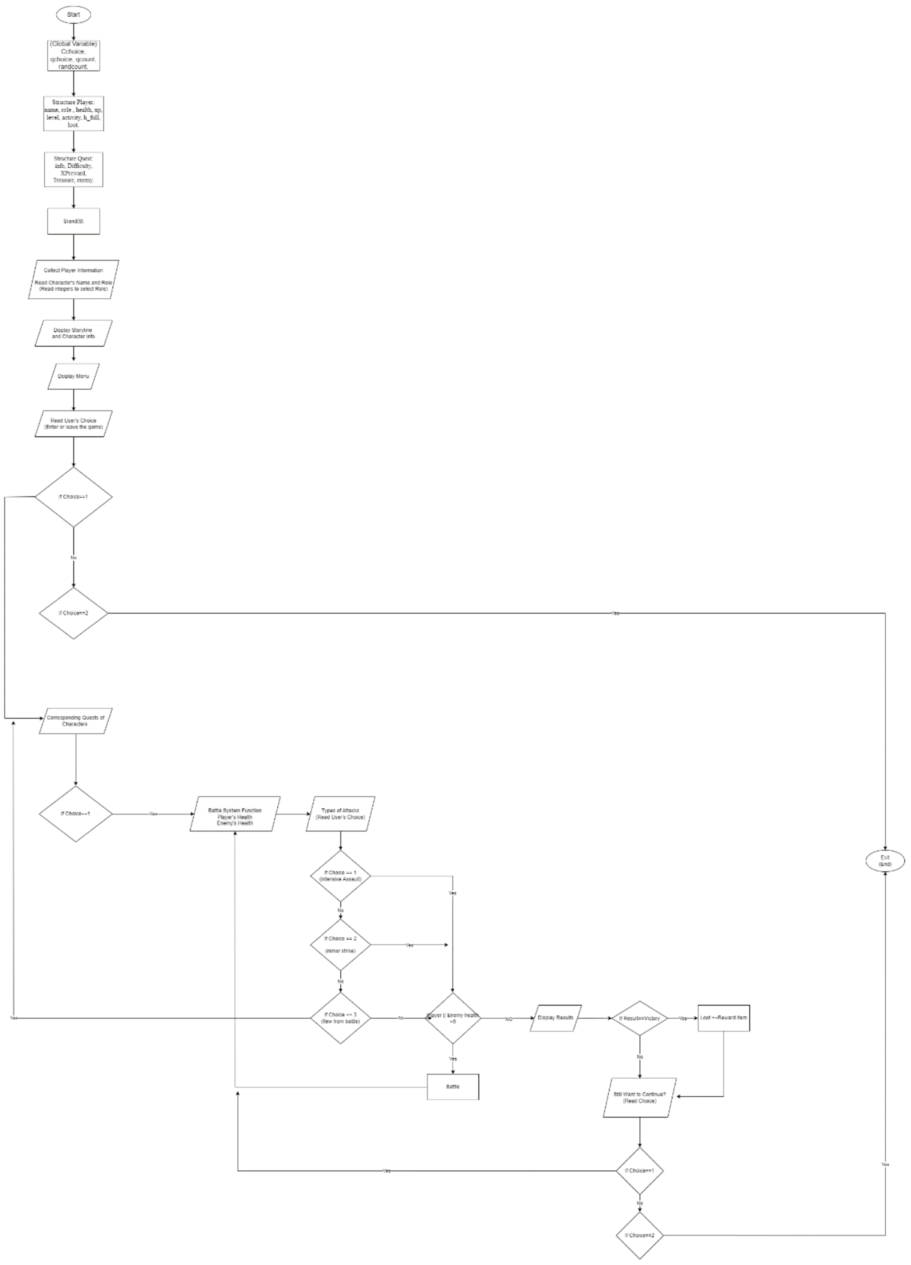
Reset XP.

Increase player's maximum health.

Step 6: Storyline Display

6.1. Display introductory storyline based on the player's class at the beginning of the game.

FlowChart:


Start
(Global Variable) Cchoice, qchoice, qcount, randcount.
Structure Player: name, role , health, xp, level, activity, h_full, loot.
Structure Quest: info, Difficulty, XPreward, Treasure, enemy.
Srand(0)
Collect Player Information
Read Character's Name and Role (Read integers to select Role)
Display Storyline and Character Info
Display Menu
Read User's Choice (Enter or leave the game)
If Choice==1
No
If Choice==2
Yes
Corresponding Quests of Characters
If Choice==1
Yes
Battle System Function Player's Health Enemy's Health
Types of Attacks (Read User's Choice)
If Choice == 1 (intensive Assault)
Yes
No
If Choice == 2 (minor strike)
Yes
No
If Choice == 3 (flew from battle)
Yes
No
Player || Enemy health >0
NO
Yes
Battle
Display Results
If Result==Victory
Yes
Loot +=Reward Item
No
Still Want to Continue? (Read Choice)
If Choice==1
Yes
No
If Choice==2
Yes
Exit (End)

Team Collaboration

The development of the Battle Realms project involved a dedicated team of three members, each contributing their skills and expertise to different aspects of the project.

Team Members:

1. Syed Amaan:

- Role: Lead Programmer
- Responsibilities:
- Implemented the core game mechanics, including combat systems, character progression, and quest interactions.
- Handled the overall game logic and algorithm design.
- Managed code integration and debugging.

2. Muhammad Daniyal:

- Role: Story Writer
- Responsibilities:
- Crafted engaging storylines and narratives based on player choices and character classes.
- Collaborated with the programming team to integrate storytelling elements into the game.

3. Malik Saiwish:

- Role: Quality Assurance and Tester
- Responsibilities:
- Conducted rigorous testing and debugging of the game, identifying and reporting bugs, glitches, and gameplay issues.
- Provided valuable feedback to enhance gameplay mechanics, balance, and overall user experience.
- Assisted in refining game features to ensure smooth gameplay and eliminate errors.

Collaboration Highlights:

- **Regular Meetings:** The team held regular meetings to discuss progress, assign tasks, and address challenges encountered during development.

- **Communication Channels:** Utilized communication tools to facilitate seamless collaboration and information sharing among team members.

- **Task Allocation:** Each member had distinct roles and responsibilities, contributing expertise in their respective areas while collaborating on shared project goals.

- **Peer Review and Feedback:** Conducted peer reviews and provided constructive feedback to enhance different aspects of the game, ensuring a cohesive and polished final product.

- **Adaptability and Support:** Flexibility and mutual support among team members allowed for adaptability in addressing unforeseen challenges and making necessary adjustments to meet project goals and deadlines.

Conclusion

The Battle Realms project benefited from the combined efforts and synergy of a dedicated team. The collaboration and coordinated contributions of each team member played a vital role in the successful development and realization of the game, resulting in an engaging and immersive gaming experience for the players.

GitHub: https://github.com/Daniya1Q/ProgrammingProject