

CPP PROJECT- GROUP 16

GAME – LEGEND OF JAGGER

PROJECT DESCRIPTION:

In this project, we have made a game named Legend of jagger using SDL library in C++ programming language.

Github Repo Link: https://github.com/vatsal-dhama/legend_of_jagger

Team Details:

Our team consisting of 5 members together contributed in this project. Name of our team members are as follows:

IMT2020039 – Anshul Jindal

IMT2020501 – Shreyansh Rajeeva Rai

IMT2020029 – Vatsal Dhama

IMT2020114 – Puram Rahul Kumar Reddy

IMT2020082 - Prudhvi Nath Reddy Sagili

Contributions:

→ **(Shreyansh and Vatsal)**

1. **Map Design :** Designed a vast world map of 100x100 tiles using open-source tilesets. This map had the following designed elements:
 - a. Grassland biome
 - b. Desert biome
 - c. Dark biome
 - d. Across the world bridge
 - e. King Albert's Castle
 - f. Dark King Zekrom's Castle

2. **Collision avoidance** : Thought of an algorithm for collision avoidance which has the time complexity of O(1).
 - a. This was possible by displaying the map as a png.
 - b. Then Tiled software was used to create a mask of the map with areas of collision marked. This masked image was exported as an 1D array.
 - c. A 1D array to 2D array conversion function was created which converted our generated 1D array of our masked image to 2D array. This provides the programmer an easier way to query the 2D array.
 - d. Now before initializing a movement in any direction, this 2D array is queried for checking whether collision is there.
 - e. This 2D array can also be used in future to query for special events to be triggered at a specific tile.

3. **Basic I/O** :
 - a. Used the SDL_GetKeyboardState functionality to identify the inputs.
 - b. Then the output movement was done in a way that the map was made to move its position instead of the player position.
 - c. Basic terminal prompts created for entering the player name and displaying score and dialogues.

4. **Sprite animations** : Added moving animations for players for all directions using the open-source sprite sheets.

5. **Tweaking animations and framerate** : Used functions to control global framerate and the individual sprite animation framerate

6. **Random fight event generator** : Created a random function which initiates a fight sequence on walking a certain amount of steps
 - a. If the player is not walking then the random function will not initiate a fight.

7. **Map Interactions** : Made certain points in the game where the player needs to go and interact with for progressing in the story.
 - a. Like there is this legendary sword “Zankyo” which the player needs to find and pickup in order to fight the dark lord Zankyo

8. **Game story design** : Designed a linear rpg story for completing the game.

- ***(Anshul Jindal)***
- **Animations:** Added Background, Hero and Enemy attack animations using the open-source sprite sheets. As the game proceeds, 3 different enemy sprites (forest, desert and dark) are rendered depending on position of the hero on the map.
- **Fighting Arena:** New window creation for the fighting arena. Game.init() executes this functionality.
- **Fight Initiating:** Hero attacks when key 1 or 2 pressed. This scenario has been implemented using SDL_GetKeyboardState functionality to identify the inputs from the keyboard. Enemy counter attacks immediately after attack of the hero.

→ **(Rahul)**

→ **Scoreboard:** Rendered health bars of hero and enemy and text on the battle zone and decreased health on particular type of attack and rendered instructions for the attacks of the hero and closing the battle zone when war ends and printed the total score of the game

→ **(Prudhvi)**

→ **Music:** Added main bgm when game starts and added different bgm when fight starts Different types of sounds for different type of attacks

Instructions to run the game:-

Step1 :- open terminal and run the following command:-

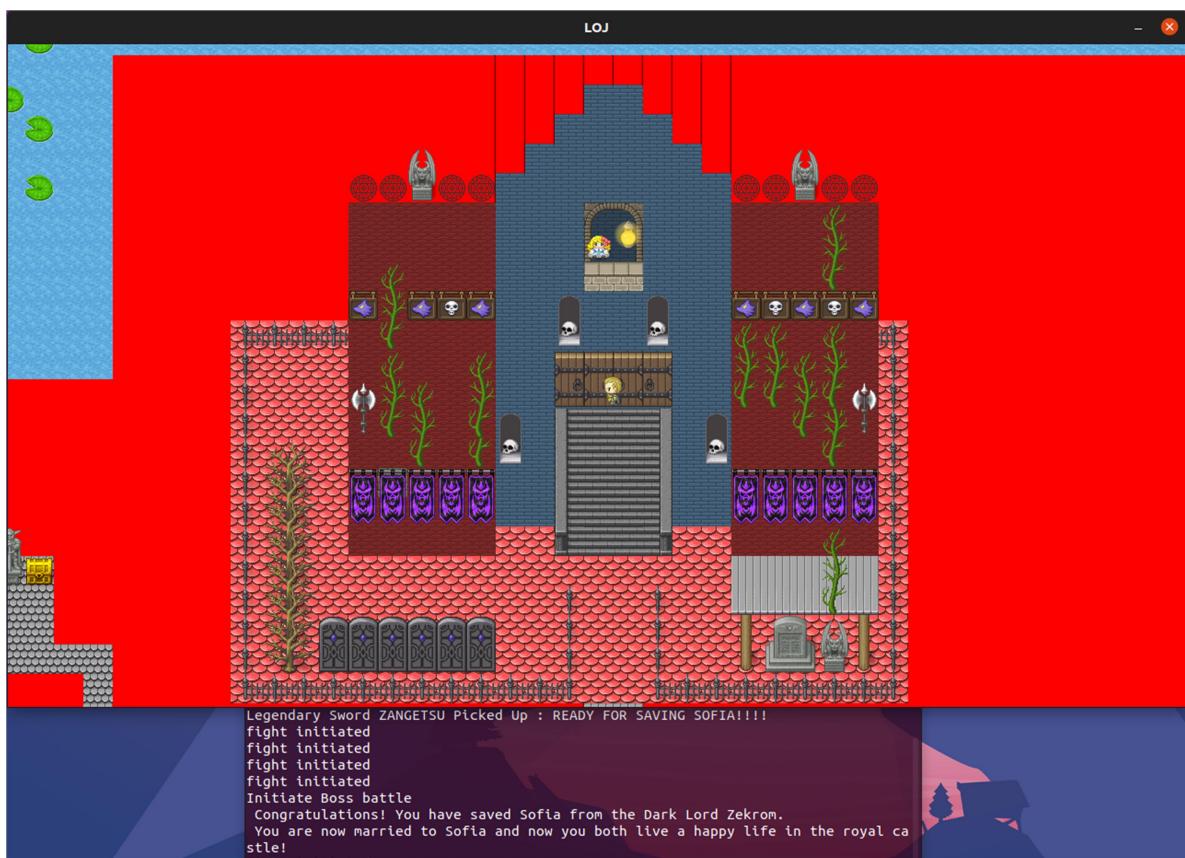
"make all". (Make sure you have all SDL modules installed in your machine)

Step2 :- Put "./main" command to start the game.

Screenshots of the game:







Instructions to run the game:

- Use the arrow keys for moving across the map
- While in a fight scene you need to press the
 - key 1 to perform attack 1
 - key 2 to perform attack 2
- The game window is designed such that the terminal is visible which will be used to check for ingame dialogue updates and display useful hints to progress in the game story.

END