Super Dude

HTML5 Game Project Documentation

Anuj Rathee

Project Overview:

Introduction

This document provides a comprehensive overview of the HTML5 game project developed using the phaser framework. The game titled “Super Dude” is a multi-level adventure game like Super Mario where players navigate through the map to collect various items and avoid enemies. The game includes interactive elements and is optimized for both keyboard and touch screen controls.

Game Features

1. Clear Plot: The game start with the player being spawned on the first level of the game and his task is to reach to the final castle and clear the stage there is two stages in the game with each having multiple enemies and traps. The game ends when the player loses all the life in the health bar or complete all the level in the game.

2. Collectable Items: There are multiple collectable items in the game while some of the are visible, some are hidden inside the platforms.

3. Interactive Environment: The game area includes moving enemies, bombs and platforms.

4. Multiple Maps: The game consist of two different level with each having a different layout.

5. Controls: The player can be controlled through keyboard and has buttons for movement for touch devices.

6. Physics Engine: The game incorporates a arcade physics engine from phaser for realistic movements and interactions

7. Enemies: Various enemies are present in the game across the stages.

8. Audio Effects: The game includes sound effects when for different actions like jump, collect coins, clear stage, game over.

9. Touch screen support: The game is fully playable on touch devices with an intuitive input system.

Tools and Technologies Used:

Phaser Framework

Phaser is an HTML5 game framework that allows for the creation of interactive games. It was chosen for its robust feature set, ease of integration with other tools, and active community support. Phaser provides a comprehensive suite of features including a physics engine, asset management, and support for both 2D and 3D graphics. This made it an ideal choice for developing a feature-rich and engaging game.

Development Environment

IDE: Visual Studio Code

Web Server: Live Server

Version Control: Git and GitHub

Sound: Audacity for sound editing

Points Justification:

Feature evaluation and requested Points:

Feature	**Max points**
Well written PDF report	3
Application is responsive and can be used on both desktop and mobile environment.	4
Application works on Firefox, Safari, Edge and Chrome	3
The application has clear directory structure, and everything is organized well	2
There is a clear plot in the game. It has a start and end.	3

The player has a health bar. When all lives are lost, the player is taken to the game over scene.	3
There is different background for each stage of the game	2
The health bar and score display follow the player and remain visible throughout the game.	2
There are different (more than 1) objects to collect	2
There are moving parts in the game area (other than the player and enemies, so e.g. some floors fall apart)	3
There is more than one map	3
Gamer needs to use keyboard to meaningfully control the player character	2
Game uses physics engine, so that there are falling parts / enemies / players	2
There are enemies that can hurt the player	3
There are enemies that the player can kill and some that they cannot, such as birds and cannons. The player can kill spiders by jumping on their heads.	3
There are some hidden items (Stars) which the player can collect by finding their location and popping them out like super Mario	3
There is music and sound effects when player shoots/jumps or anything like that	3
The game includes smooth transitions for each level. For example, Upon a game over, you can restart by clicking the screen on a keyboard device or touching the screen on a touch device. Transitions are also provided when you clear all the stages.	3
Gamer can play the game with touch screen (this is in addition to general points, as here there needs to be input system build for touch screen)	3