**Game Specification Form** **Student ID: _____** **Level 3/4**

| Marking Criteria | Describe how your game matches the criteria |
|---|---|
| **Game design (10%)** | |
| Game Goals: | Find all treasures in each level and kill the enemies. |
| Game Type: | Top-down 2D action game. |
| **Core development (30%)** | |
| Game scene (visual representation [2D, 2.5D or 3D], internal data structure): | Top-down 2D levels, each level is a different scene in Unity and is loaded when you walk up to a door. A separate Game Manager scene which contains the player and other code needed to run across all levels. |
| Game flow / game progression (e.g., navigation, screen scrolling, levels): | Player walks around the levels and uses doors to move between levels. The camera follows the player around the levels and moves to the average position of the player and enemies during combat. When walking up/down stairs the player position moves up/down inside the level accordingly. |
| Game interaction (e.g., action detection and response generation): | WASD/arrow keys move the player around. Escape pauses the game, allowing the player to see controls or quit. Spacebar toggles the flashlight on/off. Press R to eat an apple. Aim with mouse and left click to sing sword, damaging enemies that touch the sword, or right click to throw a book, dealing damage on impact. |
| Game object (e.g., use of sprite, 3D objects, animation, multimedia): | 2D animated sprites for character and enemies. Player animations include walking, throwing, swinging sword, holding the flashlight, and dying. Enemies have attack and idle animations. |
| **Game mechanics (30%)** | |
| Game rules / logics: | Player can attack with sword or throw a book projectile to deal damage to enemies. The flashlight is used to navigate dark levels. The player can eat apples Enemies will attack the player when you get close by, and drop apples/books when you kill them. Apples restore health points when you eat them. |
| Game challenges: | Player must find all treasures and kill all enemies before they are allowed to proceed to the next level. Enemies get harder as the game progresses, with a boss battle at the end. |
| **Good use of game engine (15%)** | |
| Choice (pyGame, Unity): | Unity |
| User input (keyboard, mouse, joystick): | Keyboard and mouse controls used. WASD/arrow for movement. Escape for pausing. Spacebar toggles the flashlight on/off. R to eat an apple. Aim with mouse and left/ right click to attack. |
| Game object interaction (e.g., event triggering, collision detection): | Collision detection between all objects in the game, player boundaries in the levels, books bounce off the walls, allowing you to attack enemies around corners. Player coming close to enemies triggers combat, showing the health bar and moving the camera. |
| Incorporate multimedia content: | Sound effects |
| Other features used (e.g., asset, incorporation of external libraries): | 2D assets used, credits in the CREDITS.md file. |
| **Demonstrate creativity (15%)** | |
| Game economy (e.g., support to game type, game feedback, game difficulty): | Limited ammunition for books, more found by killing enemies or exploring levels. Apples increase player health and can also be found by killing enemies and within levels. |
| Advanced Interaction (e.g., game physics, | Bees follow the player and swarm around the player |

| | |
|---|---|
| object tracking, steering behaviour): | uniformly. |
| **Game optimisation and configurability (50%) [For Level 4 Students Only]** | |
| Include optimisation to enhance game performance (e.g., game related functions, game scene and objects, interaction, rendering, media content): | |
| Make the game flexible to support making changes (e.g., game scene and objects, game flow / progression): | |