COMP2160 Game Development Task 2 –   
Implemented Features

|  |  |
| --- | --- |
| **Feature** | **Implemented** |
| 1. Physics-based player movement | 1. The avatar is represented using one of the avatar models provided on iLearn, with appropriate collider and textures. 2. The player moves in 3D using the WASD keys to move up, down, left, right, accelerating to a designer-specified maximum speed 3. The avatar should turn to face the direction they are moving in. 4. If the avatar is on the ground and the movement keys are released, the avatar should come to a rapid stop. 5. Movement controls still work if the avatar is off the ground. 6. If the avatar is off the ground and the movement keys are released, the player should continue moving at the same horizontal speed. |
| 1. Jumping | 1. If the avatar is on the ground and not inside a teleporter, and the player presses the Space key, the avatar should initiate a jump. 2. Nothing should happen if the avatar is in the air when the player presses Space. 3. The jump should move upwards to a designer-specified maximum height. 4. The avatar should be affected by gravity, a constant force in the world’s downward direction. 5. When the avatar hits the ground or a platform from above, they should stop falling. 6. The avatar’s downward velocity should be limited to a designer-specified maximum value. (used drag on the rigid-body) |
| 1. Trampolines | 1. If the avatar collides with a trampoline from above, an upwards impulse will be added to them. 2. The player’s movement controls operate as normal when on a trampoline. |
| 1. Checkpoints | 1. Checkpoints are represented by a coloured space the size of the avatar, distinct from the teleporter. 2. If the avatar is destroyed, they will restart the level at the last checkpoint they collided with. 3. The avatar can move through checkpoints without obstruction. |
| 1. Lasers | 1. If the avatar collides with a laser, they are destroyed and must restart the level from the beginning or at the last checkpoint reached. |
| 1. Switches | 1. If the avatar collides with a switch, it will turn on or off an associated laser. 2. The switch should change colour to represent if it is on or off. |
| 1. Moving Platforms – Vertical | 1. Vertical Moving Platforms move along their y-axis only. 2. Moving Platforms move at a designer specified speed. 3. Different platforms may move at different speeds. 4. If the player collides with a Vertical Moving Platform, they should move up and down with it, but still be able to use their movement controls and jump. |
| 1. Moving Platforms – Horizontal (hard) | 1. Horizontal Moving Platforms move along the x-axis and/or z-axis only. 2. Moving Platforms move at a designer specified speed. 3. Different platforms may move at different speeds. 4. If the player collides with a Horizontal Moving Platform, they should move horizontally with it, but still be able to use their movement controls and jump. |
| 1. Camera control | 1. The camera moves to keep the player avatar in the centre of the screen. 2. Moving the mouse horizontally rotates the camera left/right around the avatar. 3. Moving the mouse vertically rotates the camera up/down around the avatar. |
| 1. Scene Management | 1. Each level contains a teleporter, represented as a region of designer-specified coloured space the size of the avatar distinct from the checkpoints. 2. The avatar can move through the teleporter without obstruction. 3. If the player presses Space while the avatar is inside the teleporter, they are teleported to the beginning of a different level. |
| 1. Analytics | 1. The start of a level. 2. Whenever the player avatar touches a checkpoint. 3. Whenever the player avatar dies. 4. Whenever the player uses a teleporter. |