| Identifier | TC-1 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | Verify that the player can move left by pressing the A key. |
| Pre-condition(s) | The user has clicked on NEW GAME |
| Input data | Press the W key |
| Detailed steps | Press the W key |
| Expected result(s) | The character moves to the left |
| Post condition(s) | None |
| Actual result(s) | The character moves to the left. |
| Test Case Results | The character has successfully moved to the left. |

| Identifier | TC-2 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | Verify that the player can move right by pressing the D key. |
| Pre-condition(s) | The user has clicked on NEW GAME |
| Input data | Press the D key |
| Detailed steps | Press the D key |
| Expected result(s) | The character moves to the right |
| Post condition(s) | None |
| Actual result(s) | The character moves to the right. |
| Test Case Results | The character has successfully moved to the right. |

| Identifier | TC-3 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | Verify that the player can jump up by pressing the A key. |
| Pre-condition(s) | The user has clicked on NEW GAME |
| Input data | Press the A key |
| Detailed steps | Press the A key |
| Expected result(s) | The character jumps up |
| Post condition(s) | None |
| Actual result(s) | The character jumps up. |
| Test Case Results | The character has successfully jumped up |

| Identifier | TC-4 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | Verify that the player can move down by pressing the S key. |
| Pre-condition(s) | The user has clicked on NEW GAME, player is not on the ground |
| Input data | Press the S key |
| Detailed steps | Press the S key |
| Expected result(s) | The character moves towards the ground |
| Post condition(s) | None |
| Actual result(s) | The character moves towards the ground |
| Test Case Results | The character has successfully moved down.. |

| Identifier | TC-5 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | The gravity inverts |
| Pre-condition(s) | The game is being played |
| Input data | Press the I key |
| Detailed steps | Press the I key |
| Expected result(s) | The character moves to the ceiling if they are on the ground, or towards the ground if they are on the ceiling |
| Post condition(s) | The character “inverts” its path of run. |
| Actual result(s) | The character moves to the ceiling if they are on the ground, or towards the ground if they are on the ceiling |
| Test Case Results | The gravity has successfully been inverted. |

| Identifier | TC-6 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | Verify that the player can avoid obstacles by jumping or moving left/right. |
| Pre-condition(s) | The game is being played |
| Input data | Any movement key |
| Detailed steps | Press any movement key depending on the situation |
| Expected result(s) | The character avoids the obstacle by jumping or moving left/right. |
| Post condition(s) | The character avoids the obstacle by jumping or moving left/right. |
| Actual result(s) | The character avoids the obstacle by jumping or moving left/right. |
| Test Case Results | The character has successfully avoided the obstacle |

| Identifier | TC-7 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | The character can jump using their “grapplehook” |
| Pre-condition(s) | The game is being played |
| Input data | O key |
| Detailed steps | Press the O key to use grapplehook on the ceiling |
| Expected result(s) | The character can deploy the grapplehook and use it to jump |
| Post condition(s) | none |
| Actual result(s) | The character deploys the grapplehook and uses it to jump successfully |
| Test Case Results | Successful |

| Identifier | TC-8 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | The “TimeStop” powerup is working properly |
| Pre-condition(s) | The game is being played |
| Input data | K key |
| Detailed steps | Press the K key to use the TimeStop powerup |
| Expected result(s) | The gameplay slows down signifying that timestop has been activated |
| Post condition(s) | none |
| Actual result(s) | The gameplay slows down as expected |
| Test Case Results | Successful |

| Identifier | TC-9 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | The player attacks the enemy when P key is pressed |
| Pre-condition(s) | The game is being played |
| Input data | P key |
| Detailed steps | Press the P key to attack |
| Expected result(s) | The character swings their sword and the attack deals damage to enemies |
| Post condition(s) | none |
| Actual result(s) | The sword deals damage |
| Test Case Results | Successful |

| Identifier | TC-9 |
| --- | --- |
| Related requirement(s) | Player control functionality |
| Short description | The game ends once the player dies |
| Pre-condition(s) | The game is being played |
| Input data | None |
| Detailed steps | If the player dies the game ends and you are taken to the menu |
| Expected result(s) | Goes to main menu upon character death |
| Post condition(s) | none |
| Actual result(s) | Screen goes to main menu when the character dies |
| Test Case Results | Successful |

| Identifier | TC-10 |
| --- | --- |
| Related requirement(s) | Game control functionality |
| Short description | Verify that the game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times. |
| Pre-condition(s) | The game is being played |
| Input data | None |
| Detailed steps | Verify that the game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times. Press all keys and try all possible scenarios to determine if the game is working properly |
| Expected result(s) | The game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times |
| Post condition(s) | none |
| Actual result(s) | The game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times |
| Test Case Results | Successful |

| Identifier | TC-11 |
| --- | --- |
| Related requirement(s) | Game control functionality |
| Short description | Verify that the game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times. |
| Pre-condition(s) | The game is being played |
| Input data | None |
| Detailed steps | Verify that the game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times. Press all keys and try all possible scenarios to determine if the game is working properly |
| Expected result(s) | The game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times |
| Post condition(s) | none |
| Actual result(s) | The game plays background music and sound effects (e.g., sword swinging, obstacle collision) at appropriate times |
| Test Case Results | Successful |

| Identifier | TC-12 |
| --- | --- |
| Related requirement(s) | Game control functionality |
| Short description | Interface Functionality Verification |
| Pre-condition(s) | The game is being played |
| Input data | Click on all clickable buttons |
| Detailed steps | 1. Click/tap on each interactive element of the interface, including buttons, menus, and icons. 2. Verify that each element responds visually to user input, such as changing color or displaying feedback. 3. Test the functionality of interactive elements by performing actions like starting the game, pausing, restarting, or accessing settings. 4. Navigate through different screens or menus within the game interface, such as the main menu, level selection, and settings. |
| Expected result(s) | Each interface element responds promptly to user input without delays or unresponsive behavior.  Visual feedback confirms user actions, such as highlighting buttons upon hover or click.  All interactive elements perform their intended functions accurately, leading to the expected outcomes.  Navigation between different screens or menus is smooth and seamless. |
| Post condition(s) | Players can navigate through the game interface effectively and interact with all elements without encountering any issues. |
| Actual result(s) | Players can navigate through the game interface effectively and interact with all elements without encountering any issues. |
| Test Case Results | Successful |