### Create Node Graph

|  |  |
| --- | --- |
| Use Case Name | Create Node Graph |
| Description | A developer creates a new node graph to add nodes. |
| Actor(s) | Developer |
| Precondition(s) | None |
| Postcondition(s) | A new node graph is added to the project. |
| Trigger | Developer selects 'Graph.' |
| Main Flow | 1. Developer right-clicks in the project folder. 2. Hovers over 'Create.'  3. Sub-menu opens up. 3. Selects 'Graph.' 4. System creates a new node graph and adds it to the project. |
| Alternative Flow | If the developer clicks out of the pop-up menu before selecting 'Graph,' then no new node graph is created, and the action is aborted. |

### Delete Node Graph

|  |  |
| --- | --- |
| Use Case Name | Delete Node Graph |
| Description | A developer deletes an existing node graph from the project folder. |
| Actor(s) | Developer |
| Precondition(s) | * A node graph exists in the project folder. |
| Postcondition(s) | The selected node graph is removed from the project. |
| Trigger | Developer selects ‘Delete’ in the pop-up menu. |
| Main Flow | 1. Developer right-clicks the Node Graph in the Project Folder.  2. Pop-up menu opens. 3. Select “Delete” from the menu. 4. System Deletes the Node Graph |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the node graph remains in the project. |

### Open Node Graph

|  |  |
| --- | --- |
| Use Case Name | Open Node Graph |
| Description | A developer opens an existing node graph to add or edit nodes. |
| Actor(s) | Developer |
| Precondition(s) | * A node graph exists in the project folder. |
| Postcondition(s) | The node graph opens in the workspace, ready for editing. |
| Trigger | Developer double-clicks the node graph in the Projects folder in Unity. |
| Main Flow | 1. Developer double-clicks the node graph in the project folder. 2. System opens the node graph in the workspace. |
| Alternative Flow | If the node graph fails to open due to a system error, then an error message is displayed, and the graph is not opened. |

### Create Start Node

|  |  |
| --- | --- |
| Use Case Name | Create Start Node |
| Description | A developer creates a Start node, allowing the game to understand where the node graph starts. |
| Actor(s) | Developer |
| Precondition(s) | * The node graph must be open. |
| Postcondition(s) | A new start node is added to the node graph. |
| Trigger | Developer selects “Start Node” inside the pop-up menu |
| Main Flow | 1. Developer right-clicks inside the open node graph.  2. Pop-up menu opens. 3. Select ‘Start Node' from the menu. 4. System creates a Start Node. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Delete Start Node

|  |  |
| --- | --- |
| Use Case Name | Delete Start Node |
| Description | A developer deletes a start node from the Node Graph. |
| Actor(s) | Developer |
| Precondition(s) | * The node graph must be open. * There must be a pre-existing Start Node |
| Postcondition(s) | The start node gets deleted. |
| Trigger | The developer selects 'Remove' from the pop-up menu. |
| Main Flow | 1. Developer right clicks on a Start Node 2. Pop-up menu opens. 3. Press the “Remove” from the menu 4. The system deletes the start node. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Create Dialogue Node

|  |  |
| --- | --- |
| Use Case Name | Delete Dialogue Node |
| Description | A developer creates a dialogue node, allowing the editing of dialogue, background, characters, music, and effects. |
| Actor(s) | Developer |
| Precondition(s) | * The node graph must be open. |
| Postcondition(s) | A new dialogue node is added to the node graph. |
| Trigger | The developer selects 'Dialogue Node’ from the pop-up menu. |
| Main Flow | 1. Developer right-clicks inside the open node graph.  2. Pop-up menu opens. 3. Select 'Dialogue Node' from the menu. 4. System creates a dialogue node. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Delete Dialogue Node

|  |  |
| --- | --- |
| Use Case Name | Delete Dialogue Node |
| Description | A developer deletes a dialogue node from the Node Graph. |
| Actor(s) | Developer |
| Precondition(s) | * The node graph must be open. * There must be a pre-existing Dialogue Node in the Node Graph |
| Postcondition(s) | A new dialogue node is deleted in the node graph. |
| Trigger | The developer selects 'Remove' from the pop-up menu. |
| Main Flow | 1. Developer right clicks on a Dialogue Node 2. Pop-up menu opens. 3. Press the “Remove” from the menu 4. The system deletes the dialogue node. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Create End Node

|  |  |
| --- | --- |
| Use Case Name | Create End Node |
| Description | A developer creates an end node, allowing that signifies the end of the graph and allows the developer to also connect it to the next scene. |
| Actor(s) | Developer |
| Precondition(s) | * The node graph must be open. |
| Postcondition(s) | A new exit node is added to the node graph. |
| Trigger | The developer selects the ‘End Node’ from the pop-up menu. |
| Main Flow | 1. Developer right-clicks inside the open node graph. 2. Pop-up menu opens. 3. Select ‘End Node' from the pop-up menu. 4. System creates a dialogue node. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Delete End Node

|  |  |
| --- | --- |
| Use Case Name | Delete End Node |
| Description | A developer deletes an end node from the Node Graph. |
| Actor(s) | Developer |
| Precondition(s) | * The node graph must be open. * There must be a pre-existing End Node in the Node Graph. |
| Postcondition(s) | The developer selects the End Node from the pop-up menu. |
| Trigger | The developer selects 'Remove' from the pop-up menu. |
| Main Flow | 1. Developer right clicks on an End Node 2. Pop-up menu opens. 3. Press the “Remove” from the menu 4. The system deletes the dialogue node. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Create Scene

|  |  |
| --- | --- |
| Use Case Name | Create Scene |
| Description | A developer adds a new scene, which will be displayed during gameplay. |
| Actor(s) | Developer |
| Precondition(s) | None |
| Postcondition(s) | A new scene is added to the project. |
| Trigger | Developer right-clicks in the project folder, hovers over 'Create,' then selects 'Scene.' |
| Main Flow | 1. Developer right-clicks in the project folder. 2. Hovers over 'Create.' 3. Selects 'Scene.' 4. System adds the scene. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Delete Scene

|  |  |
| --- | --- |
| Use Case Name | Delete Scene |
| Description | A developer deletes an existing scene from the project folder. |
| Actor(s) | Developer |
| Precondition(s) | * A scene exists in the project folder. |
| Postcondition(s) | The selected scene is removed from the project. |
| Trigger | Developer selects the scene to delete, clicks on it, and presses the 'Delete' button. |
| Main Flow | 1. Developer right clicks the scene they want to delete. 2. Pop-up menu opens. 3. Developer selects ‘Delete’ from the pop-up menu. 3. The Unity removes the scene from the project. |
| Alternative Flow | If the developer clicks out of the pop-up menu, then the action is aborted. |

### Modify Speaker Name

|  |  |
| --- | --- |
| Use Case Name | Modify Speaker Name |
| Description | A developer modifies the name of the speaker in the dialogue box through the use of the Node System. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | Updated Speaker name displays in the dialogue box UI. |
| Trigger | Updated dialogue line displays in the dialogue box UI. |
| Main Flow | 1. The developer accesses the node graph  2. Selects dialogue node with Speaker Name to modify.  3. Clicks on text field and edits Speaker Name.  4. System saves modified line. |
| Alternative Flow | None |

### Modify Dialogue Line

|  |  |
| --- | --- |
| Use Case Name | Modify Dialogue Line |
| Description | A developer modifies a line of dialogue within the node system, updating the text displayed in the dialogue box. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | The system saves the modified Dialogue Line within the node. |
| Trigger | User clicks on the dialogue text box and types inside it. |
| Main Flow | 1. Developer accesses the node graph.  2. Selects dialogue node with line to modify. 3. Clicks on text field and edits dialogue. 4. System saves modified line. |
| Alternative Flow | None |

### Add Sprite

|  |  |
| --- | --- |
| Use Case Name | Modify Speaker Name |
| Description | A developer adds a sprite or an image to the dialogue node, associating an image with the dialogue node. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | Specified sprite displays alongside the dialogue line in the UI. |
| Trigger | User clicks “Add Sprite” in dialogue node. |
| Main Flow | 1. Developer accesses node graph. 2. Opens dialogue node. 3. Selects “Add Sprite” and chooses file. 4. System associates selected sprite with dialogue node. |
| Alternative Flow | The developer drags the desired sprite from Project Folder directly into the dialogue node. |

### Add Background Image

|  |  |
| --- | --- |
| Use Case Name | Add Background Image |
| Description | A developer adds a background image to the node, setting the visual background of the scene for the dialogue. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | Selected background image displays in the scene. |
| Trigger | User clicks “Add Background” in dialogue node. |
| Main Flow | 1. Developer accesses node graph. 2. Opens dialogue node. 3. Clicks “Add Background” and chooses file. 4. System displays chosen background image within the scene when triggered. |
| Alternative Flow | Developer drags the desired background image from Project Folder directly into the dialogue node. |

### Add Camera Effect

|  |  |
| --- | --- |
| Use Case Name | Add Camera Effect |
| Description | A developer adds a camera effect prefab to the dialogue node, enabling visual camera effects during the dialogue. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | Selected camera effect is triggered during the dialogue when the node is executed. |
| Trigger | User selects “Add Camera Effect” in dialogue node. |
| Main Flow | 1. Developer accesses node graph. 2. Opens dialogue node. 3. Selects “Add Camera Effect” and chooses a prefab. 4. System links the camera effect to the node for execution during dialogue. |
| Alternative Flow | Developer drags the desired camera effect prefab from Project Folder directly into the dialogue node. |

### Add Visual FX

|  |  |
| --- | --- |
| Use Case Name | Add Visual FX |
| Description | A developer adds a visual FX prefab to the dialogue node, associating visual effects with the dialogue. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | Visual FX activates when the dialogue node is executed. |
| Trigger | User clicks “Add Visual FX” in dialogue node. |
| Main Flow | 1. Developer accesses node graph. 2. Opens dialogue node. 3. Selects “Add Visual FX” and chooses prefab. 4. System links FX prefab to node, to activate with the dialogue. |
| Alternative Flow | Developer drags the desired visual FX prefab from Project Folder directly into the dialogue node. |

### Add Sound Effects

|  |  |
| --- | --- |
| Use Case Name | Add Sound Effects |
| Description | A developer adds a sound effect to the dialogue node, associating audio effects with the dialogue line. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | Selected sound effects play during the dialogue. |
| Trigger | User clicks “Add Sound Effect” in dialogue node. |
| Main Flow | 1. Developer accesses node graph. 2. Opens dialogue node. 3. Selects “Add Sound Effect” and chooses sound file. 4. System links sound effect to the node, to play during the dialogue. |
| Alternative Flow | Developer drags the desired sound effect from Project Folder directly into the dialogue node. |

### Add Music Clips

|  |  |
| --- | --- |
| Use Case Name | Add Music Clips |
| Description | A developer adds a music clip to the dialogue node, associating background music with the dialogue. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * Dialogue Node Exists |
| Postcondition(s) | Selected music clip plays as background music during the dialogue. |
| Trigger | User clicks “Add Music Clip” in dialogue node. |
| Main Flow | 1. Developer accesses node graph. 2. Opens dialogue node. 3. Selects “Add Music Clip” and chooses music file. 4. System links music clip to node for background playback during the dialogue. |
| Alternative Flow | Developer drags the desired music clip from Project Folder directly into the dialogue node. |

### Specify Next Scene

|  |  |
| --- | --- |
| Use Case Name | Specify Next Scene |
| Description | A developer specifies the next scene to be loaded when the End Node is Activated. |
| Actor(s) | Developer |
| Precondition(s) | * Node Graph Exists * Node Graph Window is Opened * End Node Exists |
| Postcondition(s) | Specified scene is loaded when the End Node is triggered. |
| Trigger | User clicks on “Next Scene” field in the End Node. |
| Main Flow | 1. Developer accesses the node graph.  2. selects the End Node.  3. Clicks on “Next Scene” field and inputs the name of the desired next scene.  4. System saves the specified next scene in the node. |
| Alternative Flow | None |

### Save Game State

|  |  |
| --- | --- |
| Use Case Name | Save Game State |
| Description | A tester clicks the Save button, which opens a Save Slot Panel where they can select a slot to save the current game state. |
| Actor(s) | Tester |
| Precondition(s) | * Game is running * Save functionality is available in the UI |
| Postcondition(s) | Game state is saved to the chosen save slot. |
| Trigger | Tester clicks the Save button in the UI. |
| Main Flow | 1. Tester clicks on the Save button. 2. System opens the Save Slot Panel. 3. Tester selects a slot or creates a new slot. 4. System saves the current game state to the specified slot. |
| Alternative Flow | If the save slot already contains data, the system overwrites the save existing save data. |

### Load Game State

|  |  |
| --- | --- |
| Use Case Name | Load Game State |
| Description | A tester clicks the Load button, which opens a Load Slot Panel where they can select a previously saved slot to load the game state. |
| Actor(s) | Tester |
| Precondition(s) | * Saved game slots exist |
| Postcondition(s) | Game state is restored from the chosen save slot. |
| Trigger | Tester clicks the Load button in the UI. |
| Main Flow | 1. Tester clicks on the Load button. 2. System opens the Load Slot Panel. 3. Tester selects a slot to load. 4. System loads the saved game state from the specified slot. |
| Alternative Flow | None |