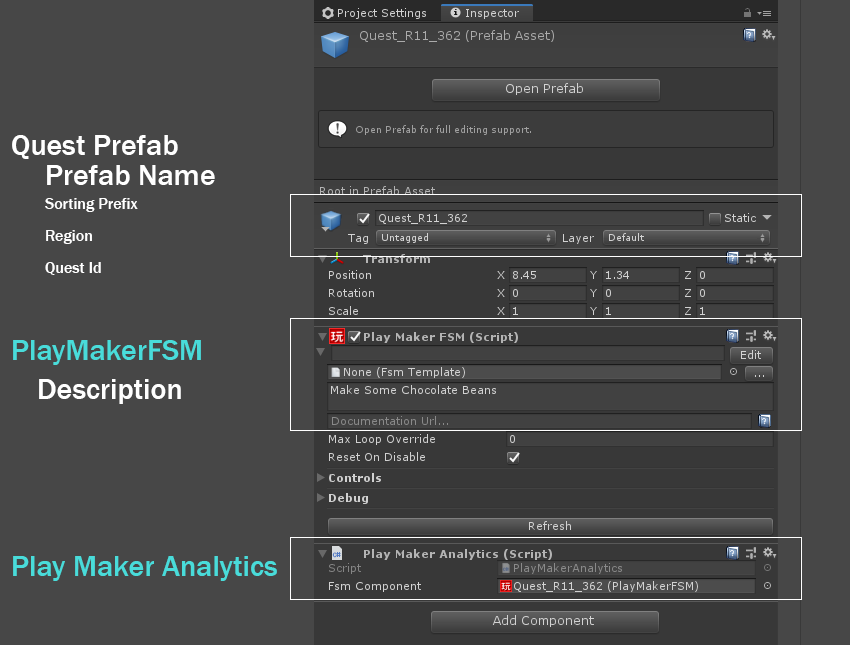
### Quest Sequence Instruction

### Quest Prefab Creation

Quest Prefab are initially empty game objects with 2 required components

* Play Maker FSM
* Play Maker Analytics

The quest prefab object needs to be configured with the following naming convention.  
  
**Quest\_<RegionNumber>\_<QuestID>**  
  
Play Maker FSM needs the description updated with either the quest title or quest description from the dialog reference sheet. This will help other developers quickly identify quests inside Unity.  
  
The Quest prefabs also require a separate analytics component. No configuration of this component is required.



### General Quest Flow

### 

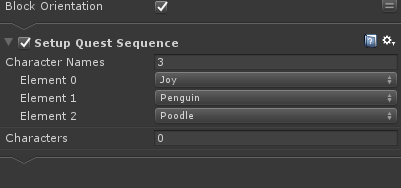
### Setup Quest Action

Use SetQuestSequence to initialize the 2D character portraits that will be used for the dialogue during the quest sequence.

In the CharacterNames field, type in the number of characters that will talk during the sequence. **ONLY 3 ARE SUPPORTED. ANY MORE WILL THROW AN ERROR.** You should notice that new fields will be created with the names “Element 0”, “Element 1” etc. Select the desired character in the dropdown menu for each field.

Element 0 is the left portrait position, Element 1 is the right-most position, and Element 2 is the middle-right position.

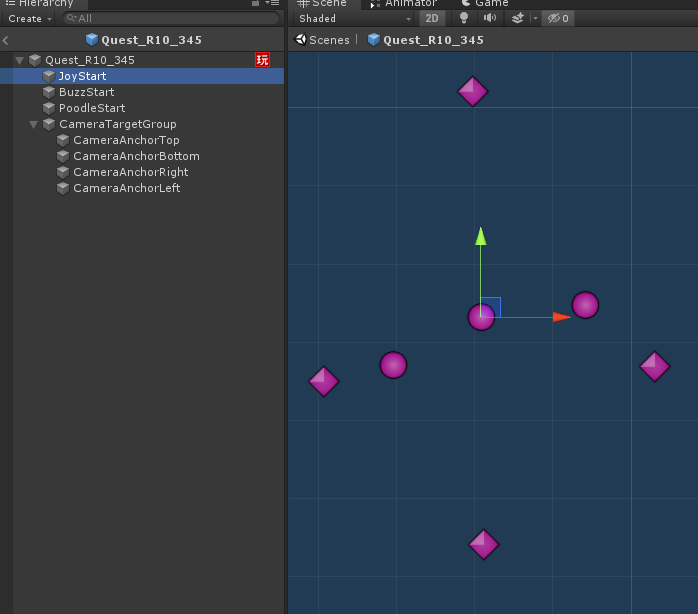
The second “Characters” field is no longer needed. It only exists for backwards compatibility so that old quests can reference character portrait prefabs directly.



### Create Character Start Position GameObjects

Add empty child objects to the quest for each character that will appear during the sequence. These objects will function as locators for each of the characters during the quest. We typically name these objects“JoyStart”, “ScoopsStart”, etc. for clarity’s sake, but any name can be used for these.

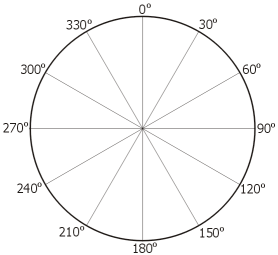
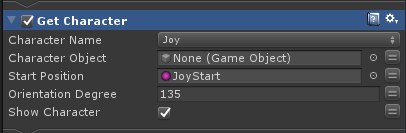
Typically it will be necessary to open up the map scene and drag the quest prefab into it in order to figure out where exactly the characters should be placed. Make sure that any quest prefabs added to the scene do not have any local / worldspace position offsets and are not nested underneath other scene objects in the hierarchy.  
  
Once you find the right placement, make sure you apply your changes to the quest prefab.

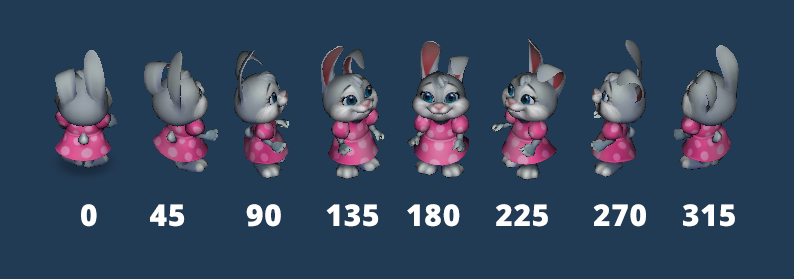


### Setup Character Position and Orientations

Use GetCharacter action to initialize a character that will appear in the quest sequence.

* For the “Character Name” field, choose the appropriate character from the dropdown menu.
* The “CharacterObject” field is optional. It saves the desired map character into a game object variable so that it can be used in quests that are more complex.
* For “Start Position”. Drag in the character locator child object directly into the field.
* “Orientation Degree” dictates where the character is looking in terms of degrees. Use the circle below for reference. Degree 0 will make characters face away from the camera while Degree 180 will make characters face towards the camera. Can also input negative values e.g. -135
* The “ShowCharacter” option determines if a character is visible or not at the start of the quest and will be checked by default.





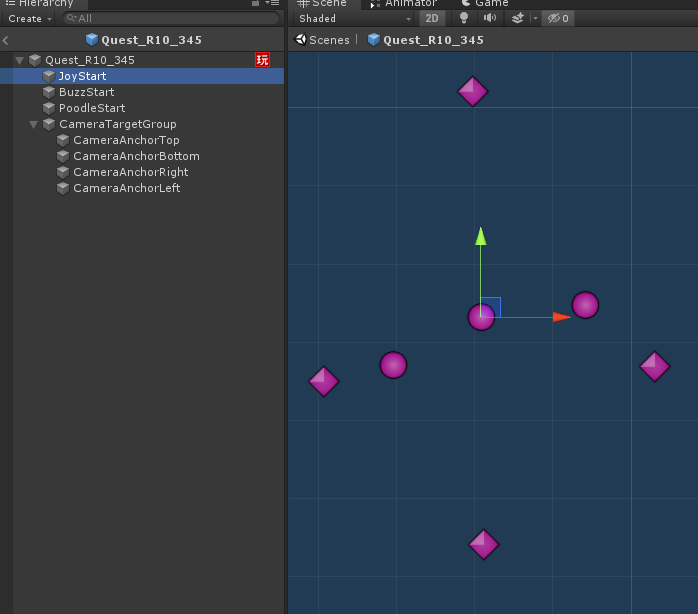
### Quest Camera Setup

#### Create Camera Group Locators

Create an empty child object in the quest prefab and call it “CameraTargetGroup”. Create four more child objects under “CameraTargetGroup” and name them

* CameraAnchorTop
* CameraAnchorBottom
* CameraAnchorRight
* CameraAnchorLeft

These game objects will determine the bounds for our camera during the quest. In the same way that the character locators are setup, open up the map scene and place the main target group on the desired centerpoint of the camera. The anchors then can be adjusted individually to adjust zoom levels and camera positioning at different orientations (See [Testing Camera Settings](#_4hntyk3e3jur)). The closer the anchors are positioned to each other the closer the camera will zoom into the scene.



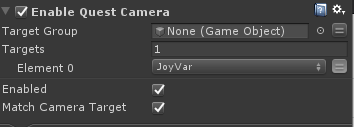
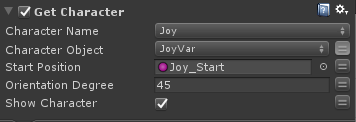
Character locators should always be inside the bounds of our camera anchors.

#### Enable Quest Camera

The Enable Quest Camera action will switch the control of the camera from the player controlled map camera to the sequence controlled cinematic camera.

Target Group - This field can be assigned the target group game object directly or use an FSM variable for the target group. Each child object of this target group object will be given influence over the camera zoom (orthographic size) and position.

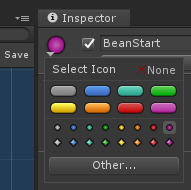
Targets - Can be used to give individual camera targets. This is maintained for backwards compatibility with previous quest sequences. It can also be used to track characters. During a Get Character action at the start of a quest you store a reference to the character using an FSM variable. See quest 585:



Enabled - This will toggle the cinematic camera On / Off. This is used in the finish quest sequence stage to revert the camera control back to the player.  
  
Match Camera Target - This will update the player controlled map camera to match the cinematic camera’s view. Enable this for a seamless transition from the cinematic camera to the map camera.

Note there is a 1 second transition when the camera view switches between map / quest camera. A **Wait** action can be used during this transition so that the sequence will not continue until after the transition is completed.

#### Assign Icons to Locators



Navigate to the inspector window while a character locator or camera anchor locator is selected. Open the icon menu and select the color purple.

* Use circle icons for character locators
* Use diamond icons for camera locators.

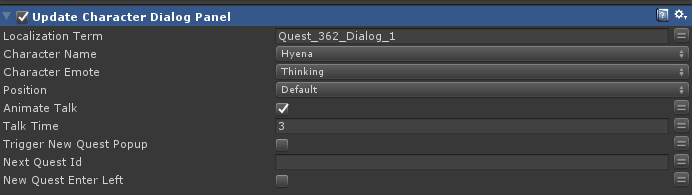
Once updated, you should notice that the empty game objects are now visible in the scene view with the chosen icons.

Gizmos must also be enabled in your scene view for the icons to display.

Start on the right and move left one color for every additional locator. E.g: 1st locator is purple, 2nd is red, 3rd is orange, 4th is yellow.

### Dialog Panel

#### Update Character Dialog Panel



The dialog panel can be opened using the update character dialog panel action. Each line of dialog will need a separate Update Character DIalog Panel action.

* Localization Term - This is a key that maps the dialog on this action to the localization sheet. Keys for each line can be found on the dialog sheet.
* Character Name - This is the character that will be speaking this dialog line. Note that most characters are not referenced directly by name. [(Character Name Reference)](#_5dues8g5imfl)
* Character Emote - This displays what character emotion portrait to display on the dialog panel. ([Character Emotion Reference](https://drive.google.com/open?id=16f3JUc9SUnMw01lHOLW4dWq2JxUZUjsO))
* Position - Use this to override position information on the dialog panel. In most cases the character positioning is already defined in the Setup Quest Sequence action and the default value can be used.
* Animate Talk - Flag this to play a talking animation on the selected 3D character while their dialog is active.
* Talk Time - The amount of time (in seconds) the character’s talk animation will play before resuming their idle animation.

#### Hide Dialog Panel

This action will temporarily close the dialog panel until another Update action refreshes the panel.. Note that certain actions (such as Select Quest Item or Open Reward Popup) will automatically hide the dialog panel.

#### Adding A Fourth Character

Some quests require more than 3 characters to speak. A new character can be added to the panel by selecting the new character’s name and position spot that the character will occupy on the panel. Note that no more than 3 characters can appear on the panel at once.

This can also be done by using the Update Dialog Characters action while the panel is hidden. See [Update Dialog Characters](#_hjjwxbv01e9s)

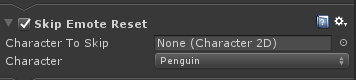
#### Skip Emote Reset

This action is used after a dialog panel. It skips a character’s emote reset, making their current emote persist.

Ex: In the dialog panel right before this action, Scoops has the Messy emote applied and we want his Messy emote to persist (not get reset back to his neutral emote) when the next character talks, we would assign Penguin to Character.

It will get overridden the next time the character talks.

* Character To Skip - In for backwards compatibility, don’t use
* Character - The character that you want to skip the emote reset

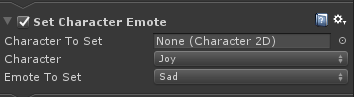


#### Set Character Emote

Sets the emote of a character. The important thing about this action is that it can be used before the character has dialog.

Ex: If Joy is playing her worried animation but she isn’t the first one to speak, you can use this action to set her to Sad even, even though she’s not the character in focus. It also skips the emote reset when a character finishes talking, so the Character will hold the emote until they have dialog or you set it to something else.

* Character To Set - In for backwards compatibility, don’t use
* Character - The character you want to set the emote of
* Emote To Set - The emote you want the character to have

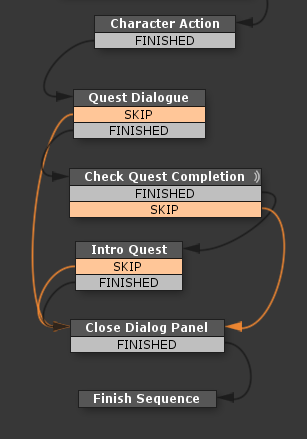


#### Close Dialog Panel

This action is used at the end of a sequence to close and reset the dialog panel. This will get the panel ready for another quest sequence. If the dialog panel needs to be removed temporarily, use the hide dialog panel action instead.

### Dialog Skip

Any state that contains dialog actions needs a skip transition in addition to a finished transition. New transitions can be added to a state by right clicking the target states and selecting:

**Add Transition > Custom Events > SKIP**  
  
Use caution when adding additional actions to dialog states as skip will immediately transition to the target state, bypassing any actions that have not yet been activated.  


Skip transitions should not bypass:

* Custom Prop Selection States
* Progression Prop Update States
* Chapter Reward States
* UI Popups

### Select Quest Item (Custom Props)

Create 2 FSM variables to use with the select quest item action.

QuestItem - FSM String

QuestItemConfig - FSM Int

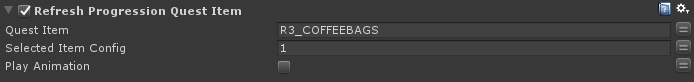
QuestItem should be assigned the prop name for the custom prop that is rewarded from the quest sequence. The prop ids for each quest can be found on the [Zone Tracking Checklist](https://drive.google.com/a/redemptiongames.com/open?id=11s_yFelyoM6K1742-3QYPdpO9BC3loor5kUIkDEvmpU). FSM Variable values can be assigned under the variable tab of the Play Maker editor.

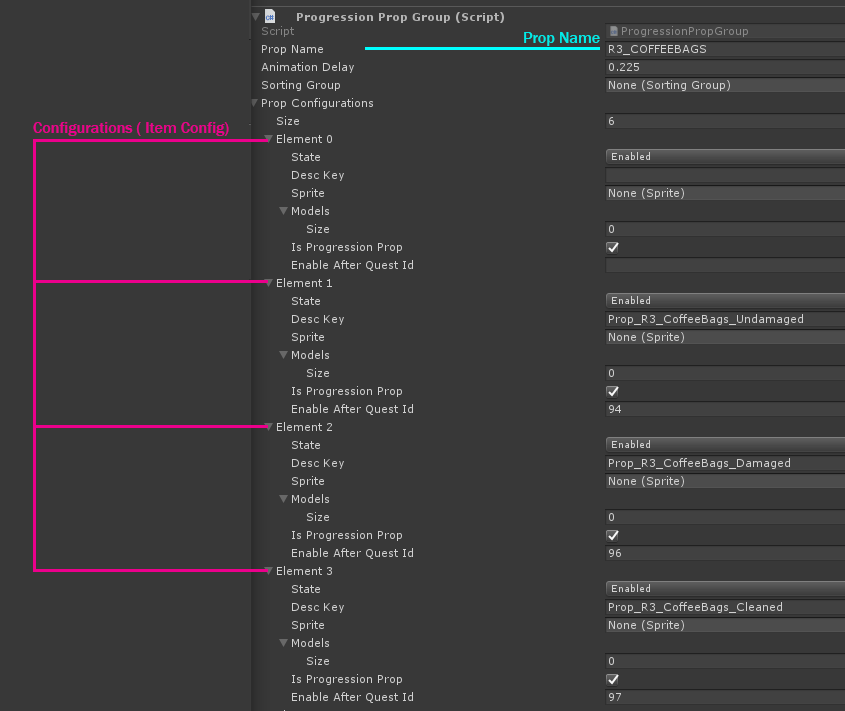
QuestItemConfig is an FSM Int Variable. The value for QuestItemConfig will be updated when the player closes the item selection screen. It does not need a default value.

These two variables should be assigned to the select quest item action. Selected Item Config MUST be assigned an FSM variable for later use with the Finish Quest action.

Select Index Override can be used if to highlight a specific configuration when the item select panel is opened.

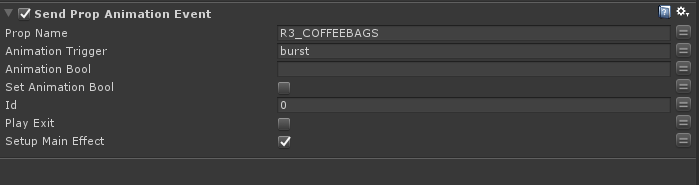
### Refresh Progression Quest Item

  
  
Progression props are map props that are not customized by the player but can update through quest progression. This is typically things like trash/clutter but could also include trees, furniture, particle effects etc.  
  
Refresh Progression Quest Item updates a progression prop to a specific state. This action needs to be configured with the prop name and updated configuration. The config state and prop id can be found on the [Zone Tracking Checklist](https://drive.google.com/a/redemptiongames.com/open?id=11s_yFelyoM6K1742-3QYPdpO9BC3loor5kUIkDEvmpU) and also on the progression prop group prefab that is updating.

The play animation flag will automatically trigger an enter animation (if the prop has one).  


### Prop Animation

Some props have custom animation that can be activated with a prop animation event. Information for prop animation needed on a quest can be found on the [Zone Tracking Checklist.](https://drive.google.com/a/redemptiongames.com/open?id=11s_yFelyoM6K1742-3QYPdpO9BC3loor5kUIkDEvmpU) Send prop animation event can be used for both Custom and Progression Props.



* Prop Name - Prop Name from Custom/Progression Prop.
* Animation Trigger / Animation Bool - Name of animation parameter
* Set Animation Bool - Set specified boolean parameter to true/false
* Id - Index of prop to animation - List can be found under the prop group component.   
   Prop Configurations > Model
* Play Exit - Automatically send trigger “exit”.
* Setup Main Effect - Toggle for spawning prop’s effect prefab during the animation.

#### Add Progression Prop Jingle Action

A separate action is needed to play the Progression Prop Jingle Music, due to how the Refresh Progression Quest Item and Send Prop Animation Event actions can be used. This action can be used in conjunction with a Wait action to sync the music with animation.

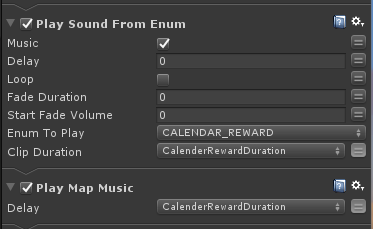
### Audio In Sequences

#### Play Sound From Enum

Note: In regular quests, you will not be calling audio.

Allows you to call sounds/music during the quest sequence. There are some caveats with sound, so make sure that you play through the quest with audio turned on to check that it’s working correctly.

* Music: Only one music track is allowed to play at a time. If Music is true, it will fade out any current music track, then play the music you have specified via the Enum. Non-music tracks however don’t have this restriction. If False (unticked) it will play the specified sound without affecting other sounds
  + It is not recommended to call multiple Music plays at once, because you will have to add a delay. And because it uses a callback for the delay, and only one music track can play at once, it sometimes won’t play.
* Delay: How long to delay playing the sound. Positive values only.
* Loop: If true, the audio will loop. If false, it will play once
* Fade Duration: How long it should take the sound to fade in. Positive values only.
* Start Fade Volume: The volume level that the audio is at when it starts to fade in. Positive values only.
* Enum To Play: The audio key for the audio clip to play. What key relates to what clip can be found in the AudioManager prefab. Music and non-music clips are seperated, with music clips being located in the
* Clip Duration: The length of the clip. Click the “=” and set an FSM variable to it. The action will set the value. This is useful for if you want to play a sound right after another sound. You can set the “Delay” of the next sound to the duration of the previous sound.



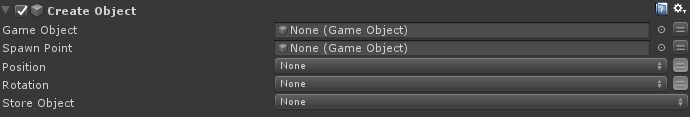
#### Play Map Music

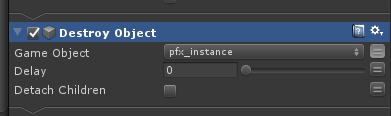
This will play the map music. Make sure to call it if you have played a sound and checked the “Music” box. Progression Prop Jingle will automatically play the map music after playing the jingle, so you don’t have to worry about calling this action after that one.

* Delay: How long it should wait before playing the map music

### Adding Sequence VFX

Effects on props are triggered from prop animations using the action [Send Prop Animation Event](#_mx7aqh2m1exp), however there are some sequences where special VFX are needed. These effects can be added to the sequence in two ways. The first method is to nest the effect prefab inside the effect prefab (See Unity Documentation on Nested Prefabs).

  
  
The effect can then be activated / deactivated using the activate game object action. This is a native playmaker action that can be found under Actions > GameObject.   
  
  
  
Effects can also be instantiated into the sequence prefab using Play Maker’s [Create Object action](https://hutonggames.fogbugz.com/default.asp?W61).

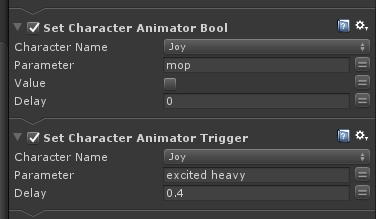


For the most part VFX objects should destroy themselves after a few seconds of being created. However, if Create Object is used to create a vfx object, this may not always be the case. It is good practice to cleanup vfx objects used in the quest by using the DestroyObject action.

Any custom effects needed for a sequence are listed in the [Zone Tracking Checklist](https://drive.google.com/a/redemptiongames.com/open?id=11s_yFelyoM6K1742-3QYPdpO9BC3loor5kUIkDEvmpU).

### Character Animation Actions

#### Set Character Animator Trigger / Boolean



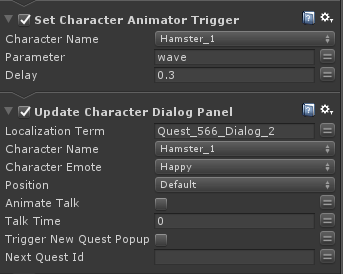
These two actions are used to activate animations on specific characters. Assign a character name and parameter name. The [animation reference sheet](https://docs.google.com/spreadsheets/d/10EJsUlUJKB5MPrwOIJPRsXCut8ZLNUJogLO8LzJS42Y/edit#gid=0) contains information for what parameters transition to specific animations.

The Delay field will delay the animation by the specified time in seconds.  
  
Looping animations use Bool parameters to control when to enter/exit looping animation states. Make sure that you turn them off again at the end of a quest, taking into account if player Skip.

Trigger parameters are used for single use animations (Animation plays and then transitions back to idle)

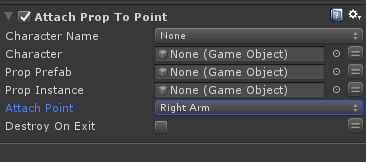
Locomotion animations (Walking / Running etc) are not controlled by trigger/bool parameters. See [Character Movement Actions](#_jt1bhhjgs61h) for controlling locomotion.

Set Character Animator Trigger / Boolean During Quest Dialog



To have a character animate during dialog, turn off Animate Talk and set Talk Time to 0. Set a slight delay for the animation so that the 2D character changing doesn’t distract the player from the 3D character animating.

#### Attach Character Props



Some character animations automatically attach props to the characters for use in the animation (Such as Joy’s phone, or sweep animations). Other animations may need a prop attached through the quest sequence. This can be done by using the Attach Prop to Point action.  
  
Character Name / Character - The character that will use the prop. The character name can be selected from the list or a character stored on an FSM variable can be used.

Prop Prefab - This is the prefab that will be instantiated on the character. This should be assigned a prefab from the project panel.  
  
Prop Instance (Optional) - This can be assigned an FSM GameObject to store the instance of the prop that is attached to the character.

Attach Point - This where the prop can be attached. Props will typically have a suffix indicating what attach point is needed. The attach points on each character are:

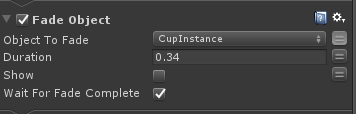
* Right arm
* Left arm
* Head
* Root - The character’s root is near where the character’s feet touch the ground plane.

Destroy on exit - This will remove the prop from the character when the sequence exits the current state.

#### Fade Object

This action can be used to fade out or fade in 3D models (such as character props or 3D map props). It can sometimes cause things behind the object to become see-through because of a rendering issue, so make sure to test it. This is good to use when characters are holding props so that the object doesn’t just pop out of existence.

* Object To Fade: the object you want to affect
* Duration: how long the fade will take
* Show: if true, the object will fade in. If false, the object will fade out
* Wait For Fade Complete: if true, it won’t exit the state until fade is complete

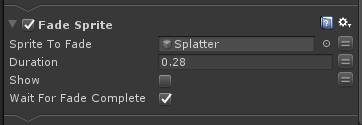


#### Fade Sprite

Used to fade out or fade in sprite objects. This will not work on 3D models. This is used most commonly for quest props, objects contained within the quest prefab. More often than not, the map props will have an exit animation so this will not need to be called.

Ex: there is a mess on the floor, and the characters are cleaning it up. Instead of making it instantly disappear you can call the fade sprite.

* Sprite To Fade: the gameobject to fade. It will also get any children objects that have sprites and fade them as well. So if an object has multiple children sprites on it, they’ll all fade
* Duration: how long the fade will take
* Show: if true, will fade the sprite in. If false, will fade the sprite out
* Wait For Fade Complete: if true, it won’t exit the state until fade is complete



#### Set character animation layer weights

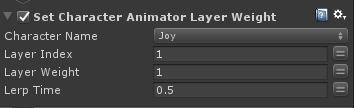
Layer weights are set on characters to modify animations.

On Joy layer 1 makes her eyes look up towards the camera, layer 2 for Joy makes her left arm stick out

Those are normally the only two you would be using in sequences.

* Character Name - which character you want to set the layer weight on
* Layer Index - the layer you want to affect
* Layer Weight - the final value you want the layer weight to be

Lerp Time - the time you want the transition to take



Joy Layer Weight

* 1 - looking up. For when she’s talking to the player
* 2 - right arm extended to hold things

Scoops Layer Weight

* 1 - sitting. He has other animations he can trigger
* 2 - holding something in front with both arms

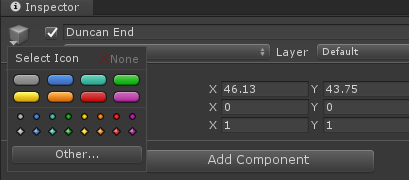
Buzz Layer Weight

* 1 - holding something in front with both arms
* 2 - he moves his upper body for dancing. Can walk.

### Character Movement Actions

In the quest prefab, create a gameobject. It will serve as the waypoint for the character.

Change the icon so it can be differentiated from the starting positions. This is done by clicking on the cube on the top left of the inspector. Circle icons are used for character locators. Blue circles are typically used as character locators on map props, so another color is recommended.

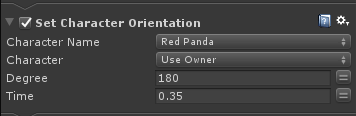


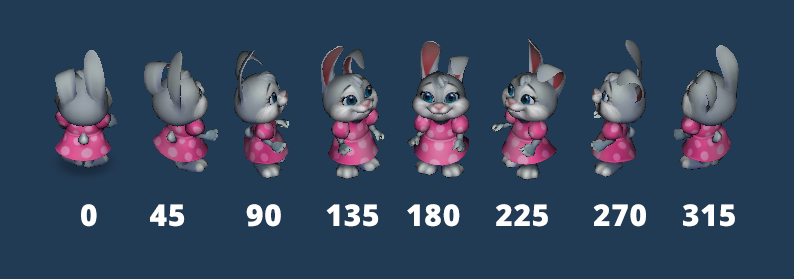
Drag in “PolyNav\_Workspace” into the scene view or load just that scene. Position your waypoint so that it isn’t in any of the boxes. If it’s inside a box the character won’t be able to navigate to it.

We usually separate movement into its own state so we can clearly tell when characters are moving. Use the action MoveCharacterToPosition if that suffices. MoveAndOrientCharacter has more options.

#### Set Character Orientation

* Character Name - set to the character you want to orient
* Character - deprecated, do not use
* Degree - what the character is going to orient themselves towards
* Time - how long the lerp is going to take. 0.35 is a good starting point.

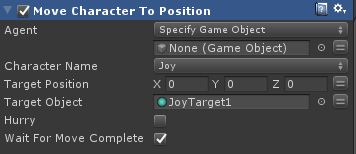




Set Character Orientation can be used before/during the startup of any character movement to orient them towards the direction they will be going. This makes their movement look better as they aren’t gliding towards their target orientation as long.

#### Move Character To Position

* Change “Agent” to “Specify Game Object”. “Use Owner” is used outside of quest sequences.
* Set “Character Name” to the character that you want to move.
* “Target Object” - either drag the waypoint gameobject to the field, or click the “=”, create a variable for the waypoint, and then assign the variable the waypoint (in the Variables tab of PlayMaker).
* “Hurry” should be checked if you want the character to run. If left unchecked, they will walk.
* “Wait For Move Complete” should be checked if you want the move action to be completed before going to the next action.

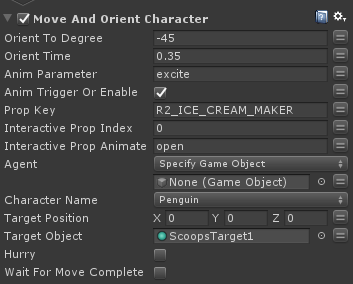


#### Move And Orient Character

This action is similar to Move Character To Position but has many optional fields.

Agent, Character Name, and Target Object are all handled the same as Move Character To Position. Everything listed below happens once the character reaches their destination.

* “Orient To Degree” is the degree that the character will orient to after completing the move.
* “Orient Time” is how long the orientation will take
* “Anim Parameter” is the animation trigger or bool that the character will call after completing the move.
* “Anim trigger or enable” should be checked if you want the animation trigger called or the animation bool set to true. If unchecked, the “Anim Parameter” will be assumed a boolean and be attempted to set to false.
* “Prop Key” is if you want the character to call the Interactive Prop script attached to a prop. An example of this is the sorting for chairs when characters sit on them. The “Prop Key” is the “Prop Name” on the Custom on Progression prop script. An example would be “R10\_BARSTOOLS”.
* “Interactive Prop Index” is the locater index of the prop. You have to look at a prop group to see what it is.
* “Interactive Prop Animate” is the animation trigger sent to the prop that corresponds to the prop key and “Interactive Prop Index”



#### Set Character Speed

This action lets you modify the speed of a character. This action can be called anytime. Before a character moves or while a character moves. The character speed will reset to default at the end of the quest so you don’t have to worry about setting it back.

* “Character Name” is the character you want to apply the change to
* “Walk Speed” is the speed when they are not in a “Hurry”
* “Run Speed” is the speed when they are in a “Hurry”
* “Hurry” makes the character “Hurry” when ticked

Joy has a default walk speed of 1. All other characters it’s 0.8. All default run speeds are 3.

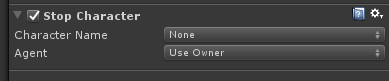


#### Stop Character

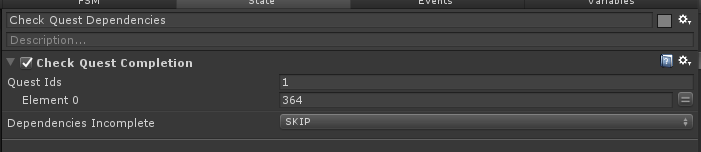
This action stops all current character movement and animations. The character will reset to its standing idle state.

CharacterName is the character you wish to stop.

Agent is an optional parameter that can be used to stop a character game object instead of using a character name. It is normally used for idle map character behavior instead of quest sequences.



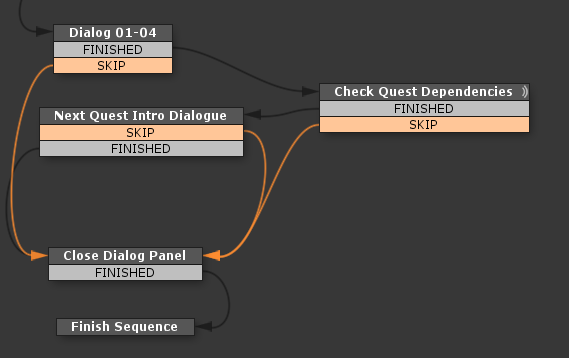
### Check Quest Completion



When a single quest has multiple dependencies, a quest completion check is needed before advancing on to that quest’s intro sequence states. Typically this is when quest lines branch and then converge back to a single quest line.   
  
The quest dependency check needs to be configured with:

Quest Ids - Amount of quests ids to check for completion.

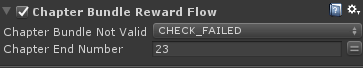
Dependencies Incomplete - FSM event / transition to use should the check fail.

The dependency incomplete event is raised when any quest listed fails a completion check. Typically the skip transition is used to bypass any quest intro states.  
  


### Chapter Reward Flow

Chapter reward screens are needed for the final quest of a chapter. These are indicated on the [Zone Tracking Checklist](https://drive.google.com/a/redemptiongames.com/open?id=11s_yFelyoM6K1742-3QYPdpO9BC3loor5kUIkDEvmpU). These are the steps needed to add the chapter reward screens:

* Create a new state labeled “Chapter Reward Flow”
* Add the transitions “FINISHED” and “CHECK\_FAILED”.
* Right click on “CHECK\_FAILED” > Link Color > Orange
* Set “Chapter Bundle Not Valid” to “CHECK\_FAILED”
* “Chapter End Number” has to be filled
* The dialog panel will be hidden by this action so it doesn’t need to be hidden beforehand

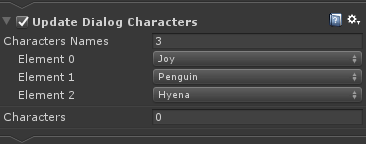


Additional steps are required for chapter end sequences. The next chapter’s quest introductions need to be split on to a new chapter intro prefab.

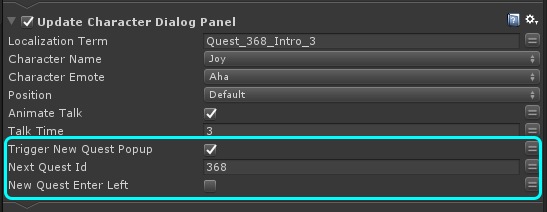
### Update Dialog Characters

This will reset and assign new dialog characters. 1-3 can be assigned. The positioning is the same as in the Setup Quest Sequence action. 0 - Left, 1 - Right, 2 - Middle. There cannot be more than 3 characters.

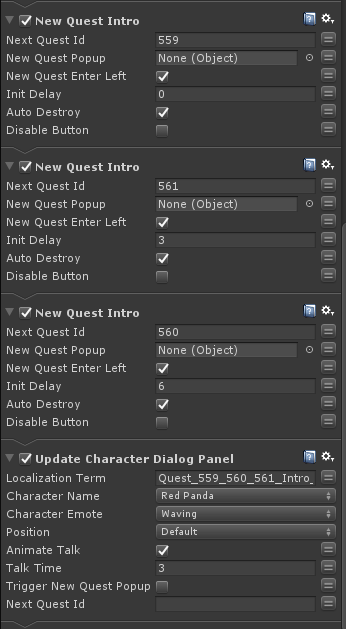
* “Character Names” - set the amount of characters you want. Up to 3. Then assign which characters you want
* “Characters” is the old style of referencing characters using variables. It is in for backwards compatibility and shouldn’t be used.
* It is best to use Update Dialog Characters while the dialog panel is hidden. Otherwise there can be a brief flicker when the characters update.



### New Quest Indicators



New Quest Indicators are needed for each new quest that is available after the current quest is completed. These indicators can be setup to appear alongside character dialog. Use the Trigger New Quest Popup flag on the Update Character Dialog Panel Action. This needs to be configured with the new quest id which can be found on the region layout reference sheet.



If there are more popups needed than available dialog, the new quest intro action can be used to open a popup. Multiple popups can be queued with delays (increments of 3) to stagger when popups appear. Popups opened with this method should use the auto destroy flag. Take into account if the 2D character saying the dialog is on the left or right side of the screen and enable New Quest Enter Left if needed.

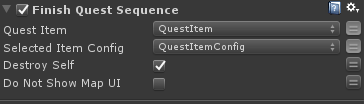
### Ending A Quest Sequence

1. Reset character animations
2. Disable sequence effects
3. [Close Dialog Panel](#_5ydtlx14ie3x)
4. [Disable Cinematic Camera and Update Map Camera](#_81cb6liwq773)
5. Finish Quest Sequence - Configure with Quest Item and Quest Item Config Variables

#### Reset Character Animations

Make sure to reset the characters back to their normal idle state at the end of a quest. If a looping animation is running on a character, make sure it is disabled before the quest ends. Props should also be removed from characters before exiting a sequence. The remove props action can be used to remove any props that are attached to a character by playmaker or by animation events.

#### Finish Quest Sequence



The Finish quest sequence action completes the quest and will add the completed quest to Player Data. Typically this is the last action used in the sequence and is also used to remove the sequence from the scene. It is also used to add any player selected items to the player’s inventory. If a player exits the game before this action is completed, the quest is not completed in their player data and will still be visible in the task panel when the player relaunches the game.

Quest Item - This is the custom prop is awarded from the quest. Use the same FSM variable from [Select Quest Item](#_4vat14xmpj2y). This property **must** be left empty if no quest item is selected in this sequence.  
  
Selected Item Config - This is the prop configuration that the user selected in the [Select Quest Item](#_4vat14xmpj2y) Action. This property must also be left empty if Select Quest Item was not used.

Destroy Self - Removes the Sequence From the Scene

Do Not Show Map UI - Does not reactivate the map UI on quest finish. Sometimes used when transitioning from a Quest to FTUE sequence.

Finish Quest No Exit is a similar action that will complete the current quest but remain in the sequence if additional actions are needed.

### Quest Polish

### 

#### Character Staging

It is best to try to get all characters to face towards the camera whenever possible. Most scenes will involve characters talking to each other. While it's ok to have the characters face each other directly, it tends to look better when their faces face slightly towards the camera.

When more than two characters are present in a scene,setting them up in a half circle formation tends to work for most quests. In some cases it will be necessary to turn characters mid-dialogue if the focus of their conversation changes. For example, characters may begin talking with each other and then address the player directly. In that case they will probably have to turn to face directly towards the camera.

Having characters face their back towards the camera should be avoided, but there may be some situations where these rules are broken.



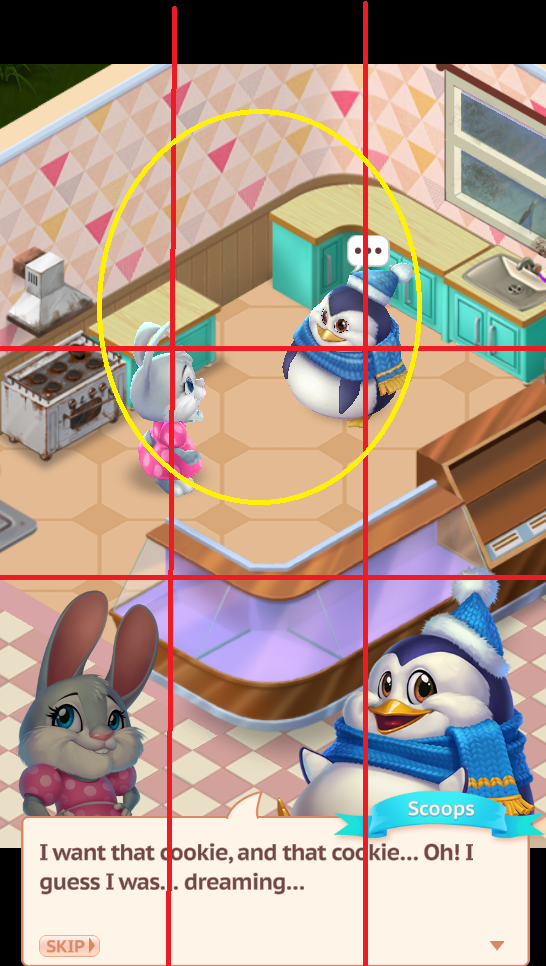
#### Camera Framing

In general the camera should always be centered where the action of the quest is at any given time. The idea is to guide the player’s attention towards the most important information in the scene, whether it is an animation sequence or changes to the props/map.

Characters should almost always be placed near the center of the camera. However, we need to reserve some space for the dialogue portraits, which in landscape mode can cover up the bottom half of the screen.

The safest placement for the characters is near the top-middle area of the screen. This will prevent 3D characters from overlapping with their 2D counterparts.

The picture below provides an example for quest 5, where Scoops breaks out of the fridge. The 2D portraits cover the bottom third of the screen, while the 3D models of Joy, Scoops and the space where the fridge used to be cover the top-middle third of the camera.

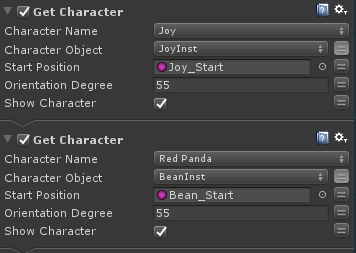


Placing characters at the very edges of the screen should be avoided. If it is not possible to fit the entire scene in one camera shot then a second camera should be used by using the SwitchCameraTargets action.

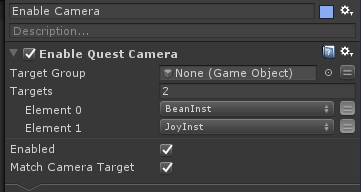
#### 

#### Characters Walk & Talk

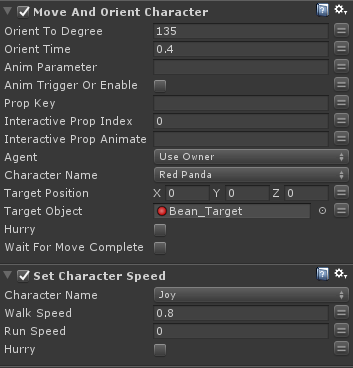
Reference taken from Quest 584



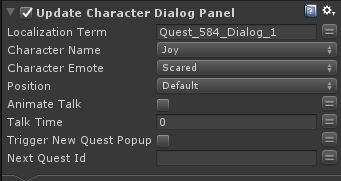
Spawn the characters normally. Have them face the direction they will be walking in. Create an FSM variable to store the Character Object. This is needed for the camera.



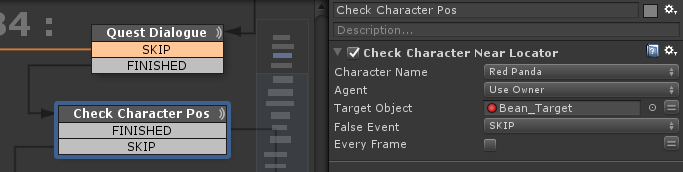
Set the camera Targets to the FSM character variables. Don’t have a wait after enabling the quest camera. This is so the characters will already be walking when the camera focuses on them and it will look more natural.



Give them a target, and make sure it doesn’t overlap with any waypoints that other map characters can spawn on. Wait For Move Complete should be disabled. Joy needs her walk speed set to 0.8 if you want her to walk the same speed as the other characters.



Animate Talk and Talk Time should be disabled and set to 0.



If there is dialog after they arrive at their destination, you want to wait until they get to their destination or fade to black and teleport them to their destination. Check Character Near Locator will check if they are at their destination. For a continuous check, enable Every Frame and don’t give it a False Event. For a one-time check, Give False Event and event, and untick Every Frame.

#### Animation Timing

When creating sequences that involve a lot of animations, it may be tempting to animate multiple events at the same time during the quest. However, it is important to remember that players can only pay attention to one thing at a time.

In general, a quest will read the best when events are staggered one after another. For example, most of the quests for our shop entrances follow a pattern where the prop animates, the camera pans towards the characters, and the characters do a celebration animation.

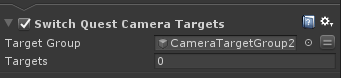
All of those events could occur at the same time, but in that scenario it is very likely that a player will not know what to pay attention to and will probably miss out on the prop animations.

A better flow for this quest would be to hang on for a couple of seconds after the player has selected the prop and allow the prop animations to play out. Then, forcing the camera to wait for the animation to finish before it pans over towards the characters. The characters in turn should wait for the camera to stop moving before starting their celebratory animations.

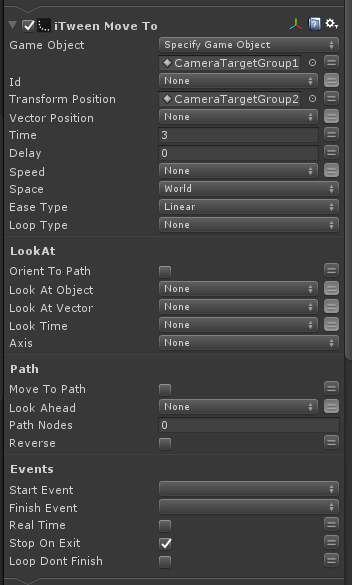
#### Switching Cameras

Sometimes you will need more than one camera for a quest. Duplicate the CameraTargetGroup gameobject, update the CameraAnchors with a new icon, and position it where you want.

Then use Switch Quest Camera Target. Drag the new CameraTargetGroup to “Target Group”. It has the same 1 second transition as when the quest camera is enabled. Can also use Targets the same as Enable Quest Camera.

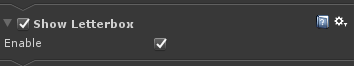


#### Switching Cameras - iTween



If the set time of 1 second is too quick, iTween can be used. Set the settings as shown above. Adjust the Time and Ease Type to fit the quest. Game Object is the Camera Target Group that is active. The Transform Position is the target that it will move to. This only moves the parent so it will not zoom in and out.

#### Letterbox



The letterbox can be enabled or disabled manually. It will come up automatically with dialog. Don’t disable it during dialog. Can be used when it feels odd to not have a letterbox. One example of where to put it is during a camera transition to the characters. Don’t enable it when a custom prop is being selected however, as the prop selection UI is there.

### Testing Sequences

### 

Sequences can be tested by starting the game and then advancing to the indicated quest ID using the debug menu. Quests can be launched normally from the quest panel or Quest prefabs can be dragged into the scene from the project window. Dragging quest prefabs into the scene window or using the Quest Sequence Debug Tool will not add the completed quest to Player data, allowing for repeat viewing.

#### Quest Sequence Tool

The sequence tool can be found in the top menu under Redemption > Quest Sequence Tool.  
To test a sequence you must first configure the map in the correct state. Use the reach tool to set the map state for the indicated quest id. Then select which quest prefab that will be tested.  
  
Any edits made to the FSM prefab will be saved when the quest is finished. This can be a useful way to test animation timing or character / camera positioning. Be aware that it will save the prefab in the state that it is in at the end of the quest so use care when using actions to alter locators or effects nested in the prefab. Their final state will be saved when using the tool.

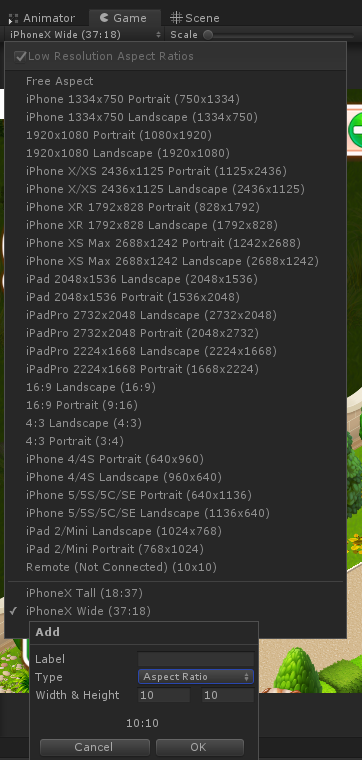
Reset Player will reset your progress back to the start of the game.

#### Testing Camera Settings

Camera settings need to be tested across different orientations and aspect ratios to ensure that all important quest content stays on screen. The camera zoom / positioning also may need to be adjusted to keep map characters and/or props from being blocked by the dialog panel.

These are the aspect ratios used for testing.

* 3:4(Portrait)
* 4:3(Landscape)
* 9:16 (Portrait)
* 16:9 (Landscape)
* 18:37 (portrait)
* 37:18 (Landscape)



Aspect ratios can be changed from the top left dropdown of the game view panel. New aspect ratio settings can also be added from this panel as well.

### Reference Documents

* [Zone Tracking Checklist](https://drive.google.com/a/redemptiongames.com/open?id=11s_yFelyoM6K1742-3QYPdpO9BC3loor5kUIkDEvmpU)
* [Dialog For Brunette Games](https://docs.google.com/spreadsheets/d/1EBk8njuYrKTyjXdgqqh5S8b1DmSkrIV054qXusIM0cA/edit?usp=sharing)
* [Character Animation Reference Doc](https://docs.google.com/spreadsheets/d/10EJsUlUJKB5MPrwOIJPRsXCut8ZLNUJogLO8LzJS42Y/edit?usp=sharing)
* [Character Emote Reference Sheet](https://drive.google.com/open?id=16f3JUc9SUnMw01lHOLW4dWq2JxUZUjsO)
* [Sequence Checklist](https://docs.google.com/spreadsheets/d/1RHpDgQ4zRi2rYkckHJbmIX1jlVxaOSMEG-_zrB8eAa8/edit?usp=sharing)
* [Play Maker Manual](https://hutonggames.fogbugz.com/)

### Character Name Reference

|  |  |
| --- | --- |
| Penguin | Scoops |
| Hyena | Buzz |
| Red Panda | Bean |
| Sugar Glider | Spree |
| Lemur | Coco |
| Parrot | Willie |
| Chicken | Wings |
| Poodle | Cherry |
| Bear | Bubba |
| Hamster\_1 | Aggy |