

Camelot Testing Environment: Known Bugs Report

The purpose of this documentation is to keep track of all known bugs that our team has encountered in our building of the testing environment. This list will be separated into three sections:

- **Visual Bugs:** These bugs are visual in nature (clipping, camera angles, etc), they do not break the Camelot engine but do present an annoyance for the viewer.
- **Camelot Bugs:** These bugs are something that Camelot should be doing, but cannot. These bugs are usually caught by the TEC, in which a popup is created.
- **Unity Bugs:** These bugs extend beyond Camelot and are due to the interaction between Camelot and it's engine. These tend to be the worst due to these bugs often causing crashes in camelot. These errors tend to be stack errors and are not outputted by Camelot but are still present in the Gameplay logs.

Visual Bugs

What is happening: Once a character is given a beard it cannot be removed, even when swapping hairstyles, the hair itself changes but the beard stays on.

What should be happening: Once a character is given a beard, and then reassigned another hair style the beard should be removed and should be the same entity as the hairstyle it comes with.

Associated Command/s: SetHairStyle(entity, Short_Beard)

Steps to Reproduce:

Create a character, and give it a beard with SetHairSyle(entity, Short_Beard)

Changing it to any other hairstyle will result in the hair being changed but the beard staying

What is happening: There is a lot of clipping of items when placing an item in a character's hand.

What should be happening:

There shouldn't be clipping or minimal clipping

Associated Command/s:

SetPosition(item, character)

Steps to Reproduce:

Create a character, and create an item of choice and place that in the character's hand.

What is happening:

There is a lot of clipping, withdrawing an item into a character's hand.

What should be happening:

There shouldn't be clipping or minimal clipping

Associated Command/s:

SetPosition(item, character)

Pocket(character,item)

Draw(character, item)

Steps to Reproduce:

Running the commands above will see clips

Some other visual bugs that are minor but noteworthy:

- In the CastleBedroom table, when the character is walking around, the character walks through the chairs.
- In the City, BrownHouseDoor, has a hinge on the door and on the wall of the door, and when opening and closing, they lap on top of each other.
- In the Dungeon, when walking into the Cell, if the camera is in track mode, it will temporarily spaz out.
- In Dungeon, when the Cell door is opened Camelot waits for a long time before sending out the input causing a character to stay outside even when the cell door itself is open.
- When in CameraMode focus, due to the nature of the camera, if anything is in front of the focused character it blocks it, and when the character moves towards the camera, it clips through the character and allows you to see its mouth.
- Items in hand increase in size when placed on a surface.

Camelot Bugs

What is happening: The Reset command resets Camelot, but does not allow for the camera to be used again, and any attempt to do so causes a crash.

What should be happening:

Reset should, reset camelot and allow the user to redo their commands without having to restart.

Associated Command/s:

Reset()

Steps to Reproduce:

Running Reset() presents you with the Camelot loading screen, but cannot be removed.

What is happening: DisableEffect does not work on specific effects. The default value that clears out all the effects does work, however if you try to disable a specific effect it does not. So DisableEffect(entity), works but clears out everything but specific ie Blood DisableEffect(entity, "Blood") does not work.

What should be happening: DisableEffect should work on specific effects.

Associated Command/s: EnableEffect(entity, "Blood")

DisableEffect(entity,"Blood")

Steps to Reproduce:

Give any entity an effect, or multiple ones. EnableEffect(entity, effect)

Trying to get rid of just one specific one does not work with DisableEffect(entity, effects)

DisableEffect(entity) does work however

What is happening: On Draw command a sword draw sound is played.

What should be happening: If a non-sword item is drawn, don't play the sound

Associated Command/s: Draw(Character, Item)

Steps to Reproduce:

Create a character, and an item, give the item to the character, pocket the item, and have the character draw the item.

What is happening: HayPiles as typed in the documentation is not a location the character can walk to.

What should be happening: HayPiles should be a registered entity or location

Associated Command/s: WalkTo(entity, Farm.HayPiles)

Steps to Reproduce:

Create a character, Load them into the farm with SetPosition()

Making them walk to the hay piles location will cause an error saying the location or entity does not exist.

What is happening: If a character dies and is revived, he cannot die again.

What should be happening:

Death should be death.

Associated Command/s:

Die(character)

Revive(character)

Die(character)

Steps to Reproduce:

Following the above steps will cause a crash on the 2nd die.

We also found a lot of bugs regarding where characters were not able to walk to certain locations due to the path being blocked, or the place not existing when it should.

All of these bugs spit out the error of **failed WalkTo(character,**

LocationName.SpecificLocation) “The path to the target is blocked” or “The entity does not exist”.

Here is a list of the location bugs:

As of 11/29/2021, any instance of **Behind**, causes an error with the Camera and cannot be resolved without removal of the tag.

Camp: The path to the **Plant** is blocked, and puts out an error.

Farm: Haypiles exists in the documentation but not in Camelot. The pathway to the **Horse** is blocked and puts out an error.

Blacksmith: Backdoor exists in documentation, but does not exist in Camelot causing an error.

CastleBedroom: Pathway to **Table** and the **Closet** is blocked and puts out an error.

Courtyard: Pathway to **Horse** is blocked and puts out an error.

AlchemyShop: Attempting to go **Behind**, will cause a stack error in unity, and **is removed from the CSV's due to the crashes and will need to be reinstated when fixed.**

City: Pathway to **BrownHouseDoor** is blocked, though this error did not come up every time, it has come up in testing.

Cottage: The pathway to the **Fireplace** is blocked, along with the **Table**.

DiningRoom: The path to the **DiningTable** is blocked if approached from specific angles, and the path to **DiningTable.FrontRight** is also blocked, and both put out an error.

Dungeon: The pathway to the **Table** is blocked and puts out an error.

Library: The pathway to **Table** is blocked, and the pathway to **Fireplace** is also blocked, both out an error.

Ruins: The pathway to **DirtPile** is blocked, and puts out an error.

Tavern: The pathway to **Table** is blocked and so is the pathway to **RoundTable**, both put out an error.

Unity Bugs

What is happening:

When zooming in on an item, it crashes the second time you do it.

What should be happening:

Camera focus on the item should work or ignore if it's called on the same location.

Associated Command/s:

```
Draw(BobB,Apple)
SetCameraFocus(Apple)
Pocket(BobB,Apple)
SetPosition(Bag,BobB)
SetCameraFocus(Bag)
```

Steps to Reproduce:

Using a created character BobB, if you use the commands provided, it crashes when the camera is focused on Bag, even when the character is holding it.

What is happening:

Setting the SetCameraMode or SetCameraFocus to the same thing as it current is causing a Camelot crash

What should be happening:

If the new camera location is the same as the old one, don't do anything

Associated Command/s:

```
SetCameraMode, SetCameraFocus
```

Steps to Reproduce: Using the same entity/camera mode twice will cause a crash.