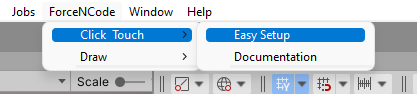
**Force Click Touch Documentation**

**Initialization:**

****

1. Find ForceNCode Bar on the Top Navigation Bar (File, Edit, Assets… etc.)
2. You should see a section as “Click & Touch”, if not please be sure your project has no error so that it can compile at least once after importing this asset.
3. Press Easy Setup in the “Click & Touch” Section

What does Easy Setup do?

1. Checks previous(old) Click Controllers in the hierarchy and removes them.
2. Finds out any camera in the hiearchy(checks main camera first).
3. Adds 2 Click Controllers to the GameObject that has camera component. One is for Mobile Devices and touch checks, other is for desktop, PC, Standalone, Editor checks

**4-** Setup is completed.

**Overview & Simple Usage:**

In Runtime you will notice that one of the controller(according to device is Mobile or Desktop) is destroyed which is the intended action.

And will create a Singleton called “ClickEventsController” for being able to reach it in any state of your scene.

**Important: This asset works with raycasts so that to get a click reaction raycast should hit a collider, which has a layer different than IgnoreRaycast (Layer : 2)**

Example Method Usage:

ClickEventsController.Instance.GetInitalPosition();

Example Event Usage:

ClickEventsController.Instance.OnClickDown += YourMethodWhileUserClickedOrTouchedScreen;

ClickEventsController.Instance.OnClickHold += YourMethodWhileUserHoldingClickOrTouch;

**Methods & Classes:**

**ClickController:**

This class is inherited from ClickEventsController please use the fields, methods and events in ClickEventsController such as mentioned in Overview & Simple Usage Part.

**ClickEventsController:**

**Fields:**

**Camera ThisCamera :** Returns Camera component that ClickController is attached to.

**float SwipeDistanceRequired:** Swipe Distance for firing swipe events.

**bool IsTouch:** Returns Camera component that ClickController is attached to.

**bool ShowRay:** Returns Camera component that ClickController is attached to.

**float MouseMoveOffset:** Returns Camera component that ClickController is attached to.

**Events:**

**OnClickDown:** Fires when click down.

**OnClickUp:** Fires when click up.

**OnClickUpNoSwiped:** Fires when click up but not swiped.

**OnClickHold:** Fires when click holded.

**OnClickStationary:** Fires when click holded but not moved.

**OnClickMoving:** Fires when click holded but and moved according to MouseMoveOffset.

**OnInitialPositionSet:** Fires when initial click position set. Can be reach via InitialPosition.

**OnSwipeRight:** Fires when swiped right. According to SwipeDistanceRequired.

**OnSwipeLeft:** Fires when swiped left. According to SwipeDistanceRequired.

**OnSwipeUp:** Fires when swiped up. According to SwipeDistanceRequired.

**OnSwipeDown:** Fires when swiped down. According to SwipeDistanceRequired.

**Methods:**

**Vector3 GetInitialPosition():** Get Initial Position Raw(without making any Screen conversions). You might get wrong value if don’t use it after click ended. (Recommended Usage On Events: **OnInitialPositionSet**)

**Vector3 GetInitialWorldPosition():** Get Initial Position after converting position via raycast hit. You might get wrong value if don’t use it after click ended. (Recommended Usage On Events: **OnInitialPositionSet**)

**Vector3 GetHoldPosition():** Get Hold Position Raw(without making any Screen conversions). You might get wrong value if don’t use it after click ended. (Recommended Usage On Events: **OnInitialPositionSet**)

**Vector3 GetHoldWorldPosition():** Get Hold Position after converting position via raycast hit. You might get wrong value if don’t use it after click ended. (Recommended Usage On Events: **OnInitialPositionSet**)

**Vector3 GetEndPosition():** Get End Position Raw(without making any Screen conversions). You might get wrong value if don’t use it after click ended. (Recommended Usage On Events: **OnClickUp**)

**Vector3 GetEndWorldPosition():** Get End Position after converting position via raycast hit. You might get wrong value if don’t use it after click ended. (Recommended Usage On Events: **OnClickUp**)

**Vector3 GetRayPoint(Vector3 Point):** Converts screenpoint to ray point then gets it.

**Vector3 GetInitialToEndDirection():** Gets the DIRECTION vector from intial click point to end click point

**Vector3 GetInitialToEndWorld ():** Gets the vector from intial click point to end click point and converts it to World Position

**Vector3 GetInitialToHoldWorldDirection():** Gets the DIRECTION vector from intial click point to holding click point and converts it to World Position

**Vector3 GetInitialToHoldDirection():** Gets the DIRECTION vector from intial click point to holding click point raw.

**Vector3 GetInitialToHoldWorld ():** Gets the vector from intial click point to holding click point and converts it to World Position