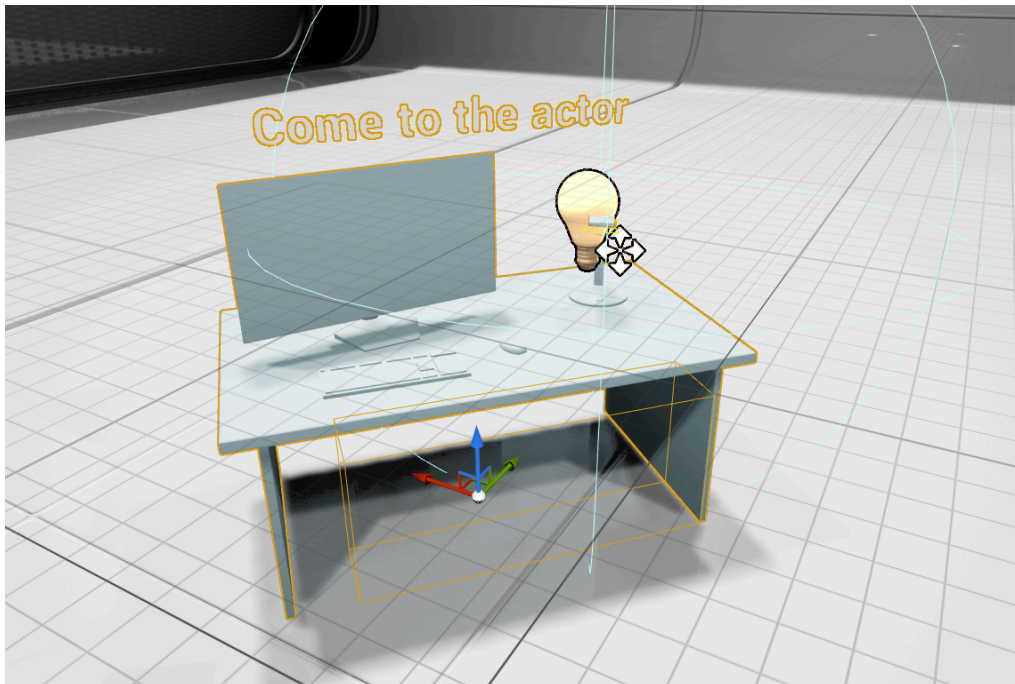




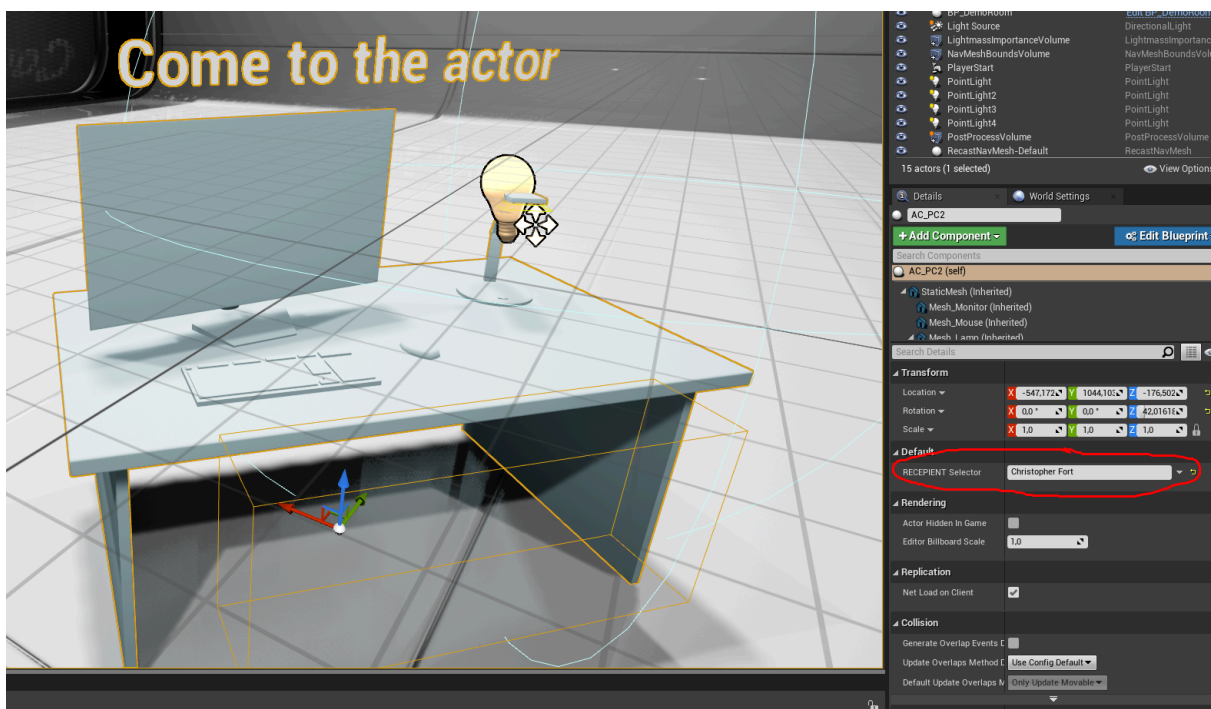
Lite Widget GrMail

How to use

Step 1: Position the actor at the level where you want to install the mail widget



Step 2: Create a variable in it, I named it "RECEPIENT Selector". For convenience, I made it "Instance Editable"

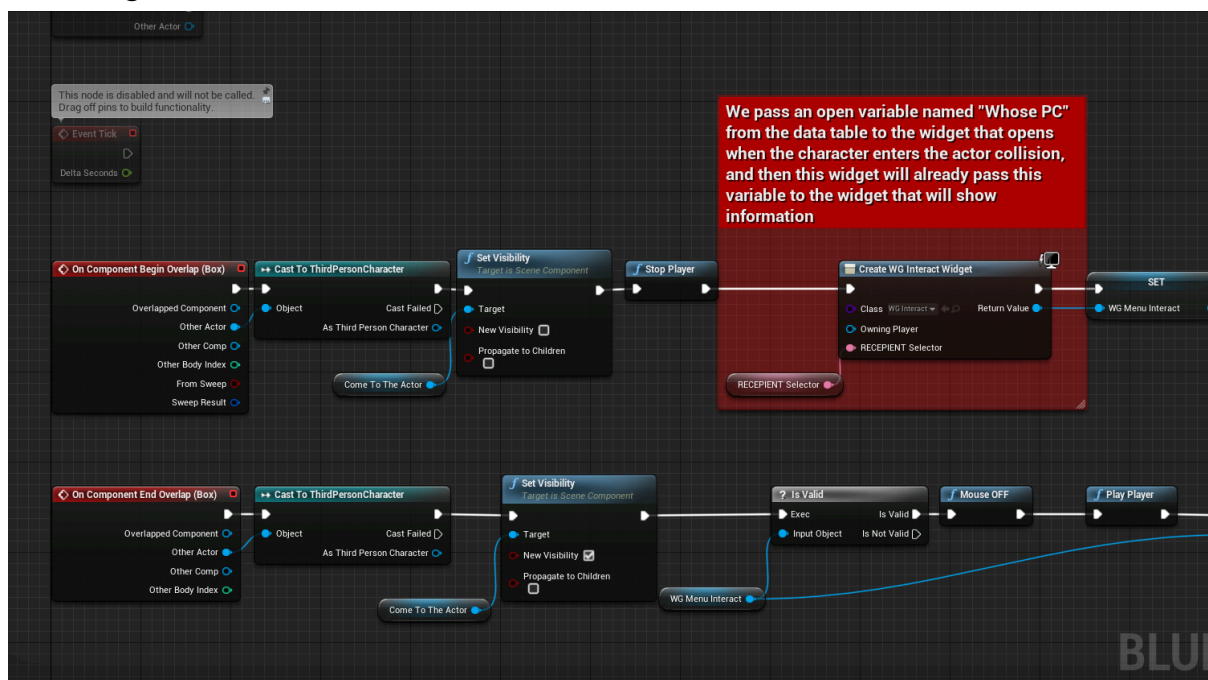


Step 3: Fill in the "DT_Mail" data sheet with your details. Please note that the "ID" must be individual for each entry (1). "Whose PC" we must specify this value in the variable of our actor (2)

Row Name	ID	Type_of_Letter	Whose_PC	RECEIVED	SENDER	RECIPIENT	SUBJECT
NewRow	0	inbox	Engineer Brandon	03/01/2023	Archie Joe <joearch@gmail.com>	Brandon Joe <drandon782joe@gmail.com>	Brother read!
NewRow_0	1	sent	Engineer Brandon	03/01/2023	Archie Joe <joearch@gmail.com>	Brandon Joe <drandon782joe@gmail.com>	OK
NewRow_1	2	inbox	Christopher Fort	01/18/2023	Shop "MegaBRE" <mail@megabre.com>	Christopher Fort <christ1@gmail.com>	Read this very important
NewRow_2	3	draft	Christopher Fort	01/18/2023	Shop "MegaBRE" <mail@megabre.com>	Christopher Fort <christ1@gmail.com>	Cancel subscription
NewRow_3	4	sent	Christopher Fort	01/19/2023	Director Kristi Kup <Kricput@gmail.com>	Christopher Fort <christ1@gmail.com>	Cancel subscription
NewRow_4	5	inbox	Christopher Fort	01/19/2023	Director Kristi Kup <Kricput@gmail.com>	Christopher Fort <christ1@gmail.com>	WHAT?
NewRow_5	6	sent	Christopher Fort	01/19/2023	Director Kristi Kup <Kricput@gmail.com>	Christopher Fort <christ1@gmail.com>	Excuse me
NewRow_6	7	inbox	Engineer Brandon	07/03/2023	Archie Joe <joearch@gmail.com>	Brandon Joe <drandon782joe@gmail.com>	Hou did not forget?
NewRow_7	8	sent	Engineer Brandon	02/26/2023	Jessica Mert <gessmert@gmail.com>	Brandon Joe <drandon782joe@gmail.com>	Reminder

After filling in the table, I selected the actor at the level and in the Settings on the "RECEPIENT Selector" wrote one of the selected "Whose_PC".

Step 4: In the "DemoLevel" actor, you can see a blueprint in which when a character enters a collision, a menu widget is created to select actions, but if you do not need this, you can directly open the "WG_Mail" widget. Don't forget to pass the value of the RECEPIENT Selector variable to the widget!!!



The diagram illustrates a visual programming workflow. It begins with a 'SET' node (TEXT) connected to a 'Create WB Button Widget' node. The 'Create WB Button Widget' node has inputs for 'Class' (WB Button), 'Owning Player', and 'Button ID'. Its 'Return Value' output is connected to a 'SET' node (Widget, Target). This 'SET' node is then connected to a 'Textbutton' node (Target is WB Button). The 'Textbutton' node's 'Target' input is connected to the 'RECEIVED' node, which is circled in red. The 'RECEIVED' node is also connected to the 'Type Of Letter' input of a 'Switch on' node (Selection). The 'Switch on' node has a 'Selection' input and a 'Type Of Letter' output.

The screenshot shows a Node-RED workflow for handling a 'Break STRUCT_WIND' event. The workflow consists of the following nodes and connections:

- Break STRUCT_WIND** (Function Block):
 - Input: **STRUCT WIND** (blue circle)
 - Outputs: **ID** (green circle), **Type Of Letter** (green circle), **Whose PC** (pink circle), **RESEIVED** (pink circle, circled in red), **SENDER** (pink circle), **RECIPIENT** (pink circle), **SUBJECT** (pink circle), **TEXT** (pink circle).
- GetText (Text)** (Function Node):
 - Label: *Target is Text*
 - Input: **RESEIVED Text** (blue circle, circled in red)
 - Output: **Return Value** (pink circle)
- AND** (Logic Node):
 - Label: **AND** Add pin +
 - Inputs: **RESEIVED** (pink circle) and **Return Value** (pink circle)
- Branch** (Logic Node):
 - Label: **Branch**
 - Inputs: **Condition** (red circle)
 - Outputs: **True** (white circle) and **False** (white circle)
- Equality Comparison** (Function Node):
 - Label: **==**
 - Inputs: **ID** (green circle) and **Return Value** (pink circle)
 - Output: **Condition** (red circle)

The workflow logic is as follows:

- When the **STRUCT WIND** event occurs, the **Break STRUCT_WIND** node is triggered.
- The **RESEIVED** output of the **Break STRUCT_WIND** node is passed to the **GetText (Text)** function node.
- The **GetText (Text)** function node returns a value to the **Return Value** output.
- The **RESEIVED** output and the **Return Value** are combined in the **AND** node.
- The output of the **AND** node is passed to the **Branch** node.
- The **Branch** node evaluates the condition and routes the flow to either the **True** or **False** output.