Digital Workshop

Build your first social robot interaction!

Recap steps to get the SDK and blockly up and running

1. Install the SDK (skip his step if you already installed it)

- Follow the written instructions from your teacher or at docs.furhat.io
- Look at the instructional movie if you need extra help

2. Open the SDK launcher application and run the Virtual Furhat

- Select ‘Launch SDK’ to run the Virtual Furhat
- (Optional) If your computer is struggling to run the Virtual Furhat, you can close the Furhat Head animation, the Furhat emulator will still run

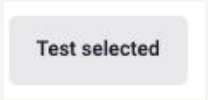
3. Start the blockly tool

- Select ‘Launch blockly’
- The blockly interface will open in a browser tab

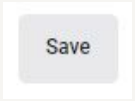
You are ready to start coding!

Key buttons to use / keep in mind in Blockly

These are your start & stop buttons to try your interaction



Select a specific state, or a specific part of the interaction to test.



This button exports your flow to the downloads catalogue in your computer.

Load a previously saved flow.

This clears everything on your canvas - use with care! (or in emergency, click Ctrl Z to get it back)

Blockly update with Furhat Platform 2.0+

Note!

Blockly received a significant update in August 2021 when version 2.0 of the Furhat Platform was released.

See summary of improvements →

This workshop was created with the old Beta version* of blockly - some new and improved features have been added!

Summary of blockly changes from Beta version*:
- New State block that can inherit other states (replaces flow-init state)
- Autosaving the interaction
- Replaced import/export/load/save with load/save
- 'Test Selected' can now be used on entire states
- Attend block now has options for selecting a user
- User triggers now have options for selecting a user
- New Trigger for user smiling
- New settings block for number of users and size of interaction space
- New value block for number of users
- New condition block for evaluating number of users
- Removed unused blocks (edited)

*** Can be downloaded from furhat.io/downloads**

Workshop tasks

The task

Create a interaction with a character that...
... greets you
... ask how you are
... asks your name

Does it sound too simple? Don't worry!

Interactions that are very simple for humans

are the most difficult to program well!

Create a interaction with a character that...

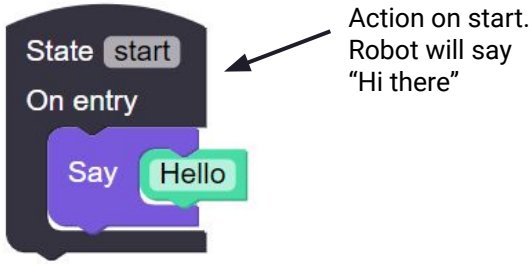
Step 1 - Greets you

Step 2 - Asks how you are

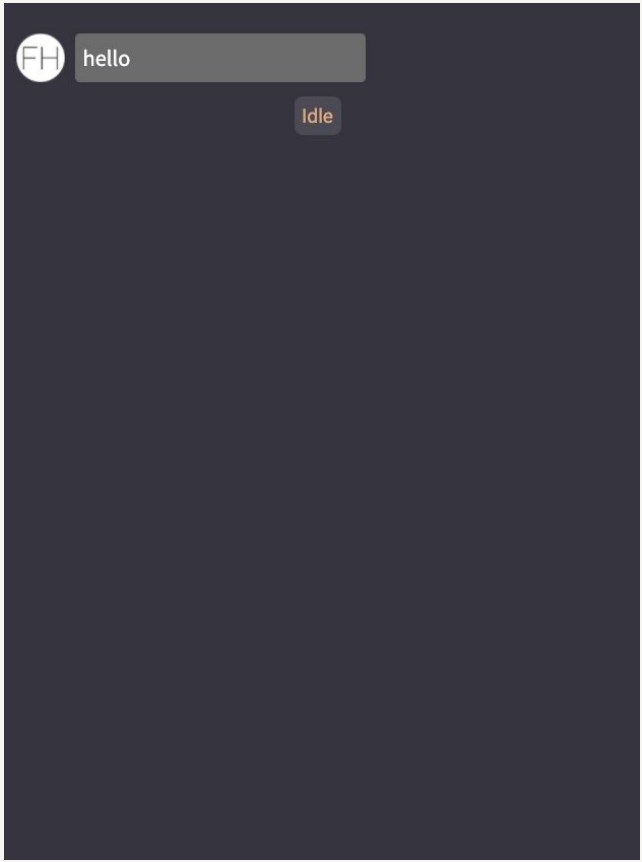
Step 3 - Asks your name

Intro - Step 1

Add a say action on start.



Action on start.
Robot will say
“Hi there”



Create a interaction with a character that...

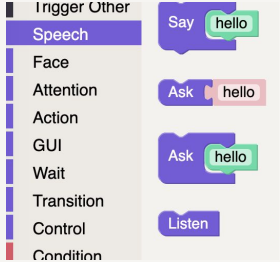
Step 1 - Greets you

Step 2 - Asks how you are

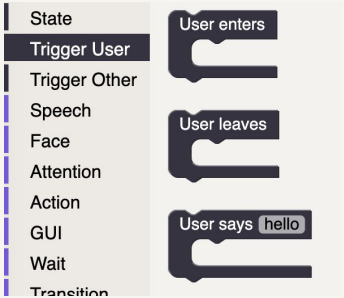
Step 3 - Asks your name

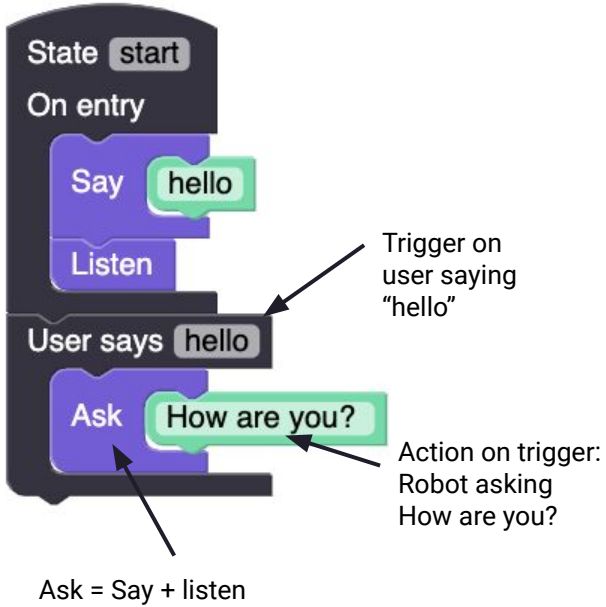
Intro - Step 2a

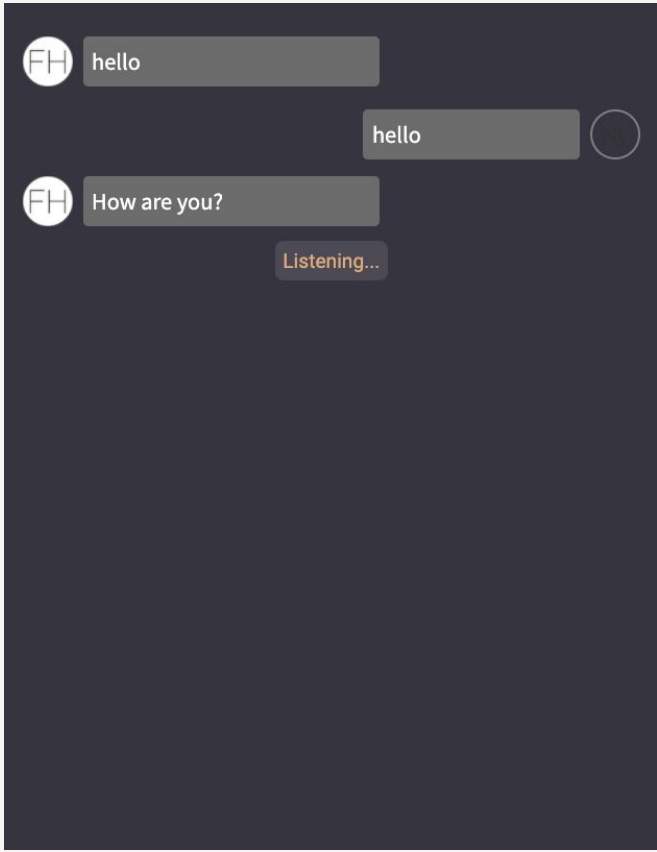
Add a listen - so the robot is listening for a reply.



Add a trigger that will trigger when user speaks.



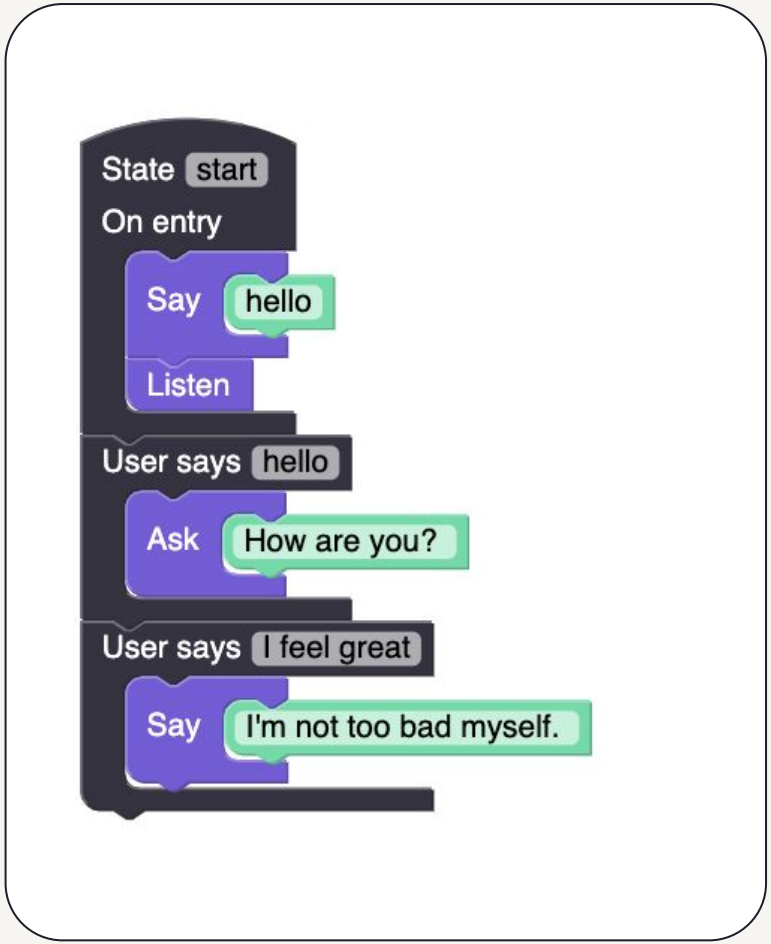


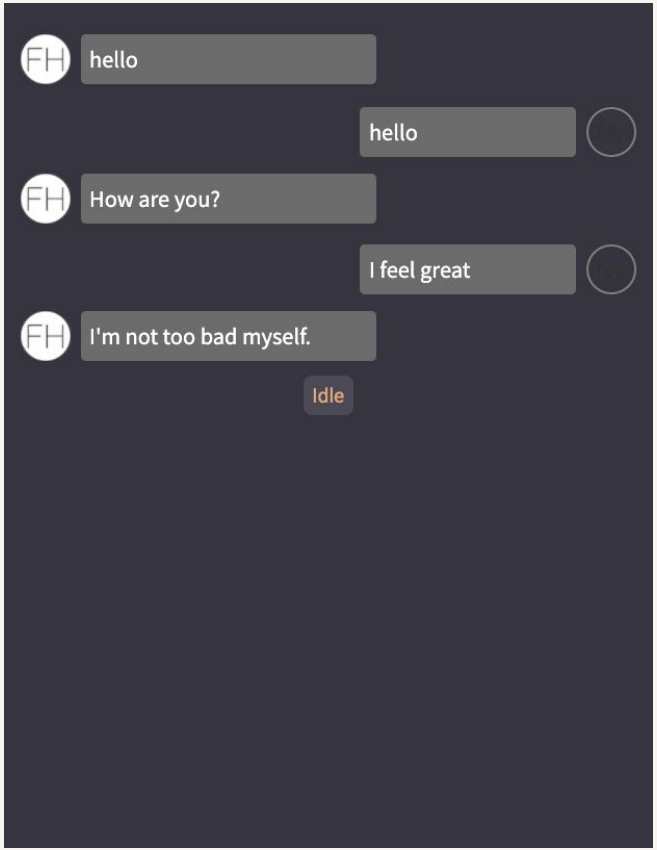


Intro - Step 2b

Add a trigger for user saying “I feel great”

Add a response from the robot:
“I’m not too bad myself”





Create a interaction with a character that...

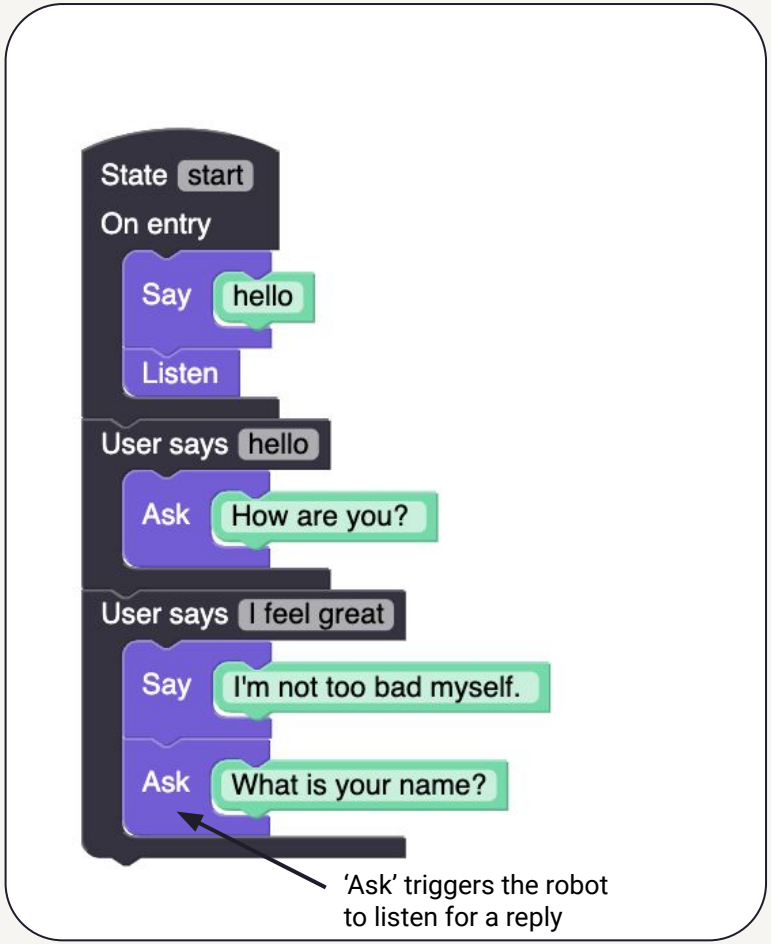
Step 1 - Greets you

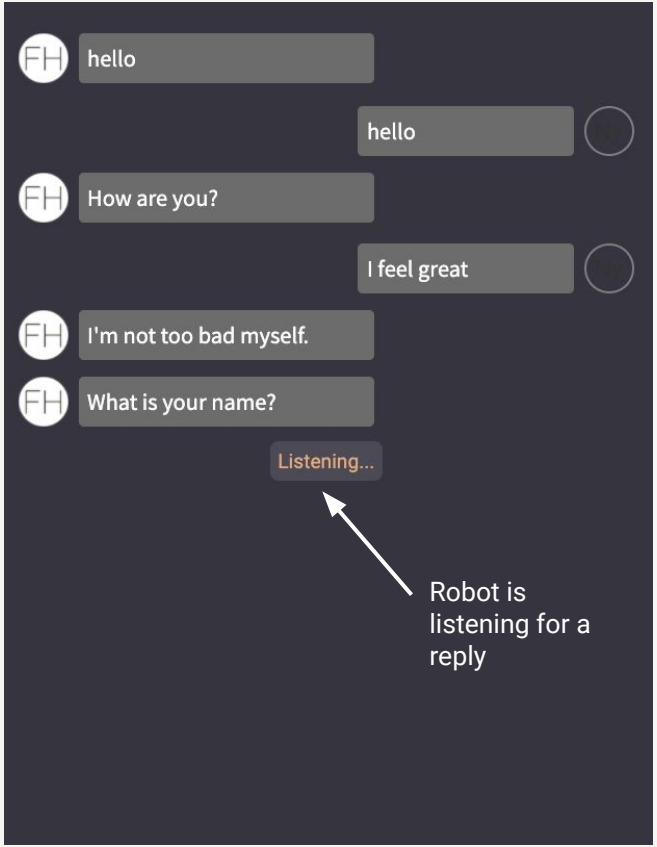
Step 2 - Asks how you are

Step 3 - Asks your name

Intro - Step 3a

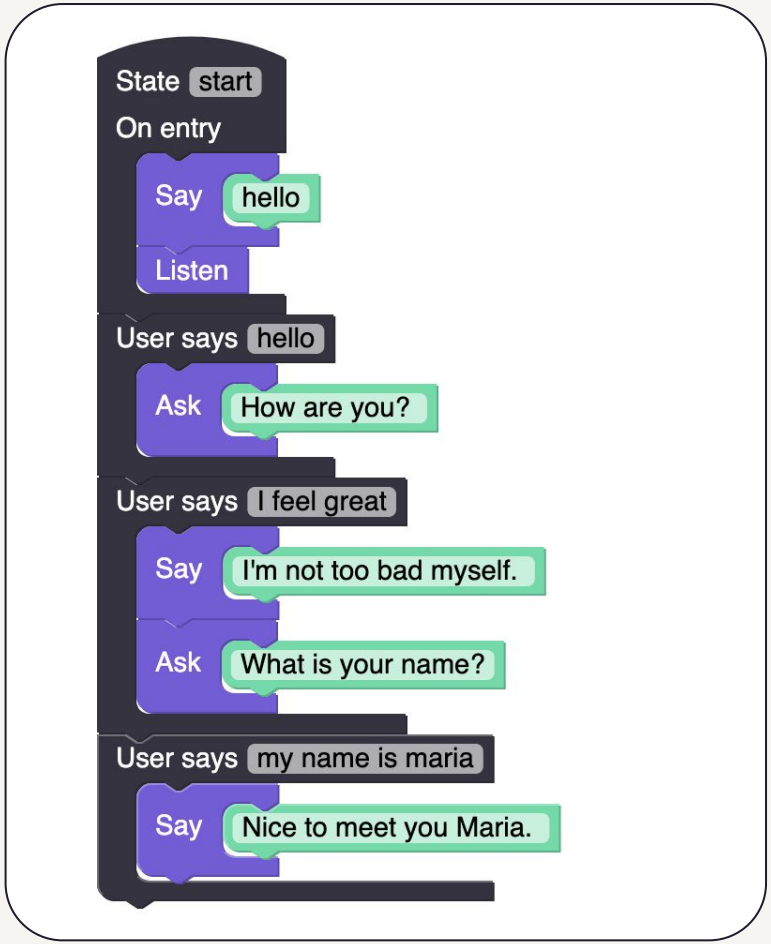
Add action where the robot ask for the name of the user

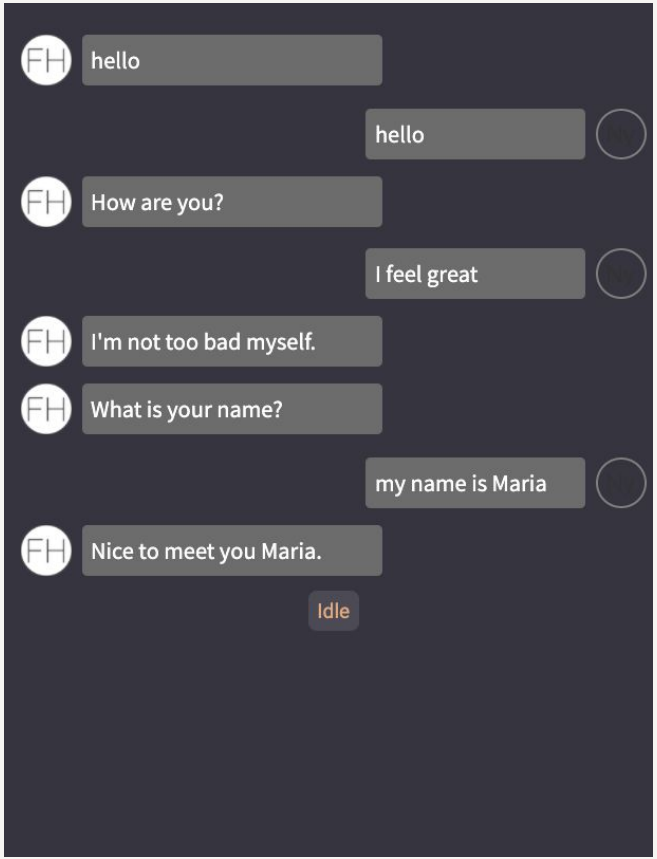




Intro - Step 3b

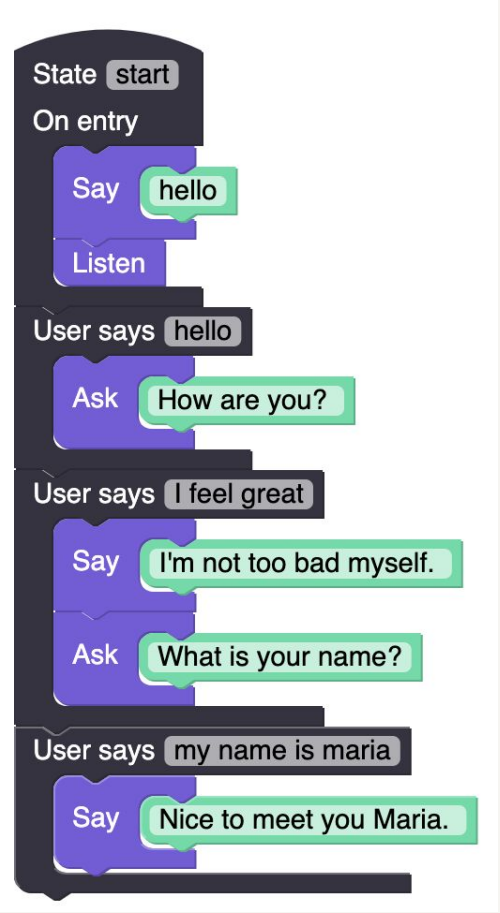
Add a trigger for the user saying their name.





Good job!

You now have created an interaction that...
... greets you
... ask how you are
... asks your name

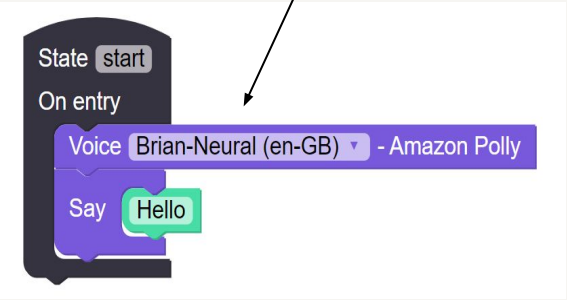


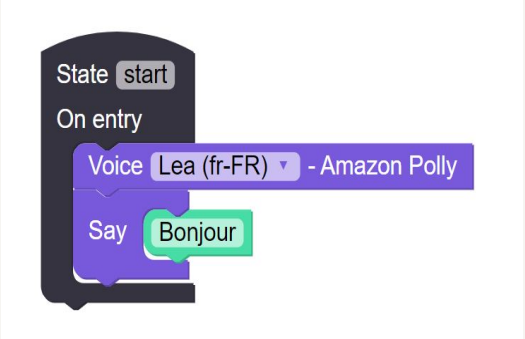
Of course not!

What are my options now?

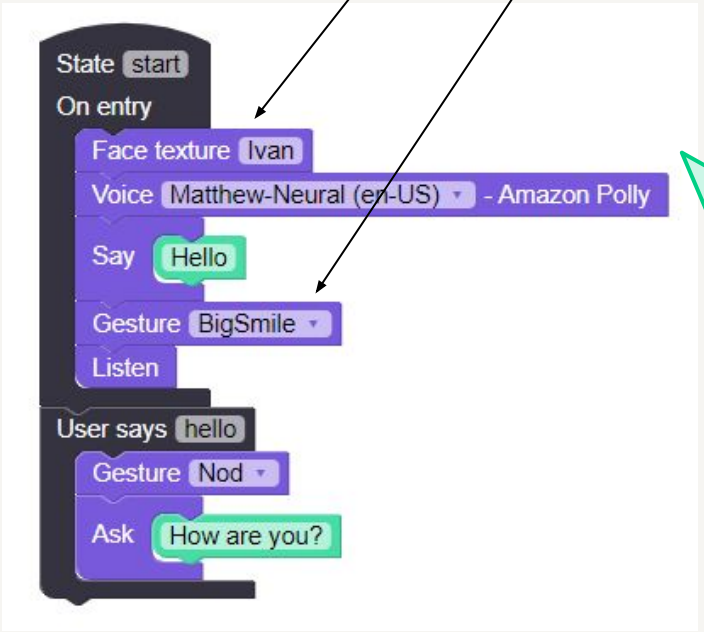
- **You can try to build your flow in another language**
- **You can add face gestures to your robot**
- **You can change what the robot / user says**
- **You can continue on to the next steps, and make your interaction more *robust* and more *delighting*.**

By adding a 'Voice' block (found in Speech folder), you can choose the voice and language you prefer. But remember to also change the spoken text to your new language





You can add a new face look and different gestures by adding a 'Face texture' or 'Gesture' block (found in Face folder) - try it out on your virtual Furhat and see how it looks!



For Face textures, you have to type in one of these options yourself:

- Elsa
- Fedora
- Ivan
- Fred
- Geremy
- Marty
- Rene
- Ted
- Ursula

Making the interaction more **robust**

Right now we have an interaction that works but will fail often.

... It only responds to exact phrases
... There is no *flow* – the user can say anything anytime
... All *actions* and *triggers* are in the same place (this become messy and difficult to manage as our interactions grows)
... There is no error handling

So in the following steps we will:

Step 4 - Break up the interaction into multiple states with transitions in between

Step 5 - Change our triggers to respond to *intents* rather than exact phrases

Step 6 - Add error handling when user is silent or says something we did not understand

Making the interaction more **robust**

Step 4 - Break up the interaction into multiple states

Step 5 - Change our triggers to respond to intents rather than exact phrases

Step 6 - Add error handling when user is silent or says something else

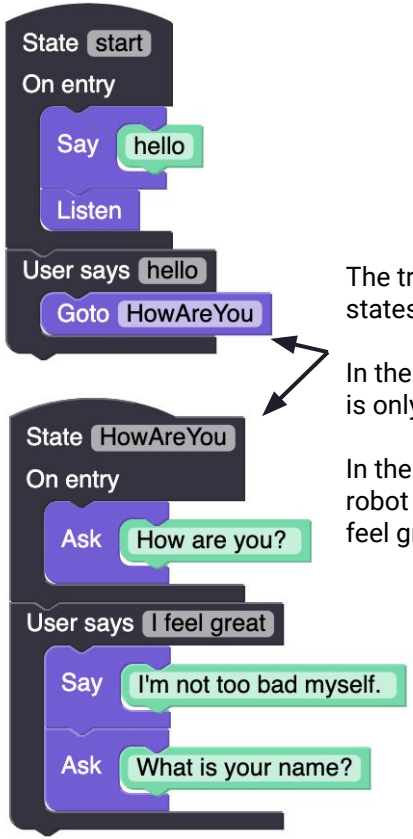
Robust - step 4a

Create a new state and name it: HowAreYou

Create a goto transition to the new state and place it in the action block in the user trigger “hello”.

Move the ask action (“How are you”) on the trigger to the ‘On entry’ of the new state.

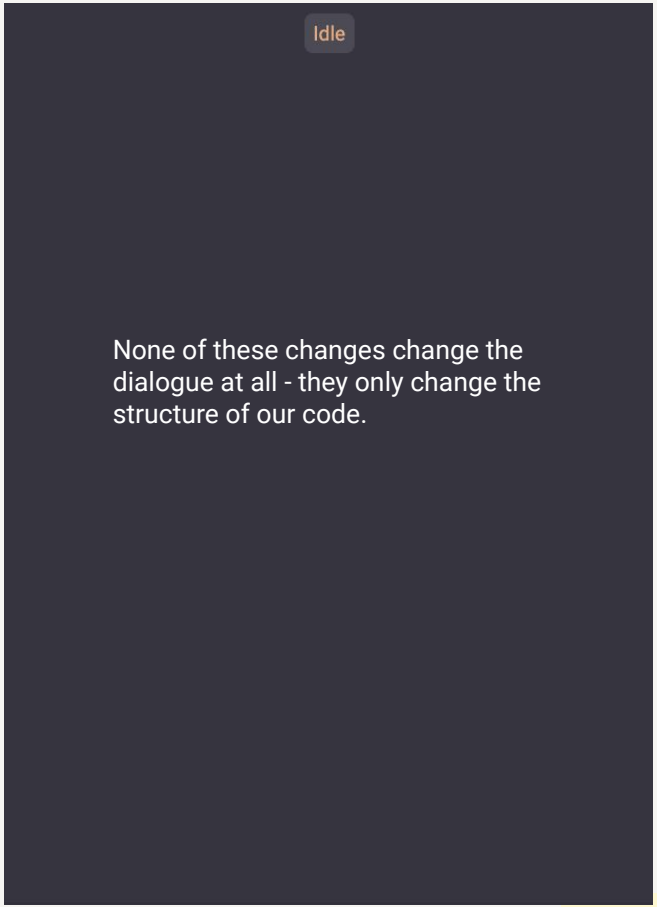
Move all the other triggers to the new state.



The transition binds the states into a dialogue *flow*.

In the start state - the robot is only listening for “hello”

In the HowAreYou state the robot is only listening for “I feel great”



None of these changes change the dialogue at all - they only change the structure of our code.

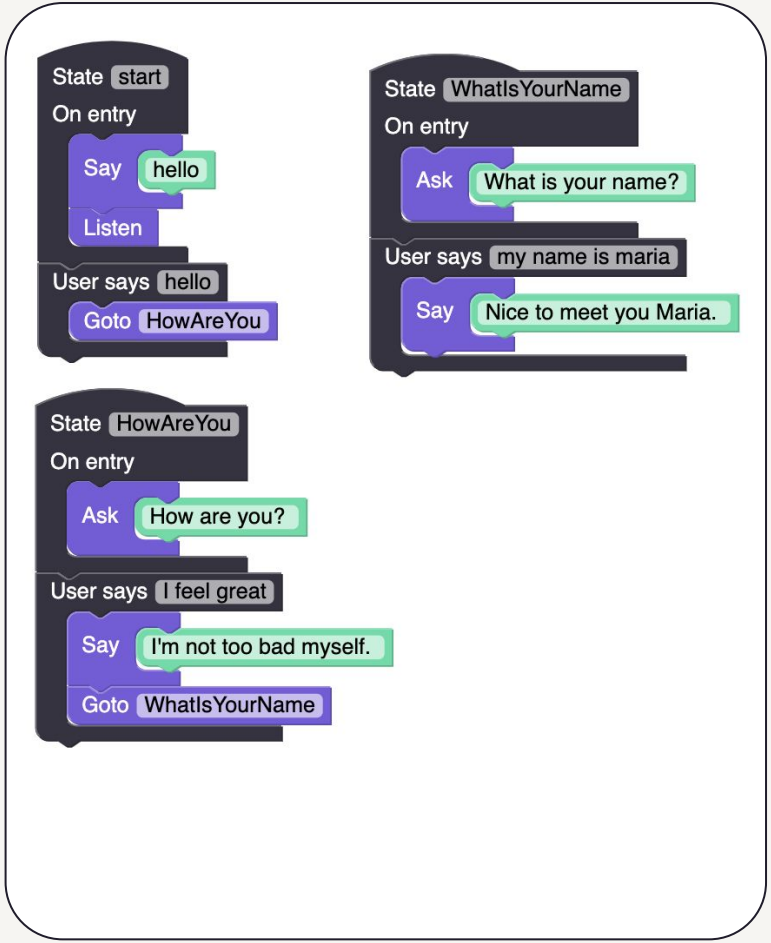
Robust - step 4b

Create a new state and name it: WhatIsYourName

Create a goto transition to the new state and place it in the action block in the user trigger “I feel great”.

Move the ask action (“what is your name”) on the trigger to the ‘On entry’ of the new state.

Move the “my name is maria”-trigger to the new state.



Idle

None of these changes change the dialogue at all - they only change the structure of our code.

Making the interaction more **robust**

Step 4 - Break up the interaction into multiple states

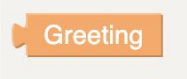
Step 5 - Change our triggers to respond to intents rather than exact phrases

Step 6 - Add error handling when user is silent or says something else

Robust - step 5a

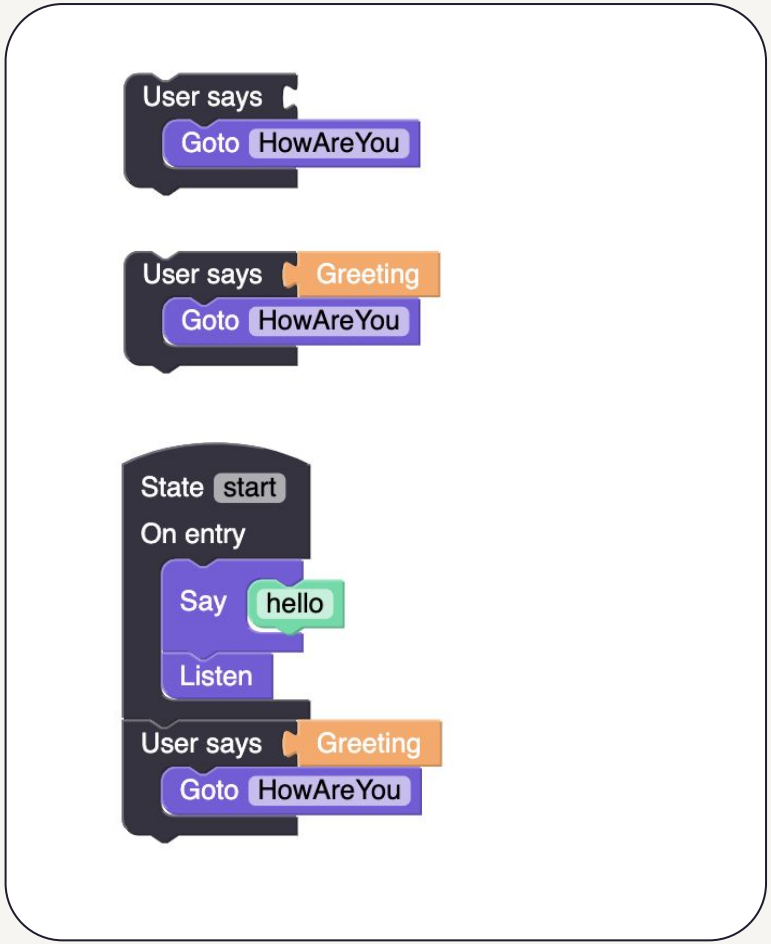
Change the user trigger “Hello” to a trigger with a slot for an intent.

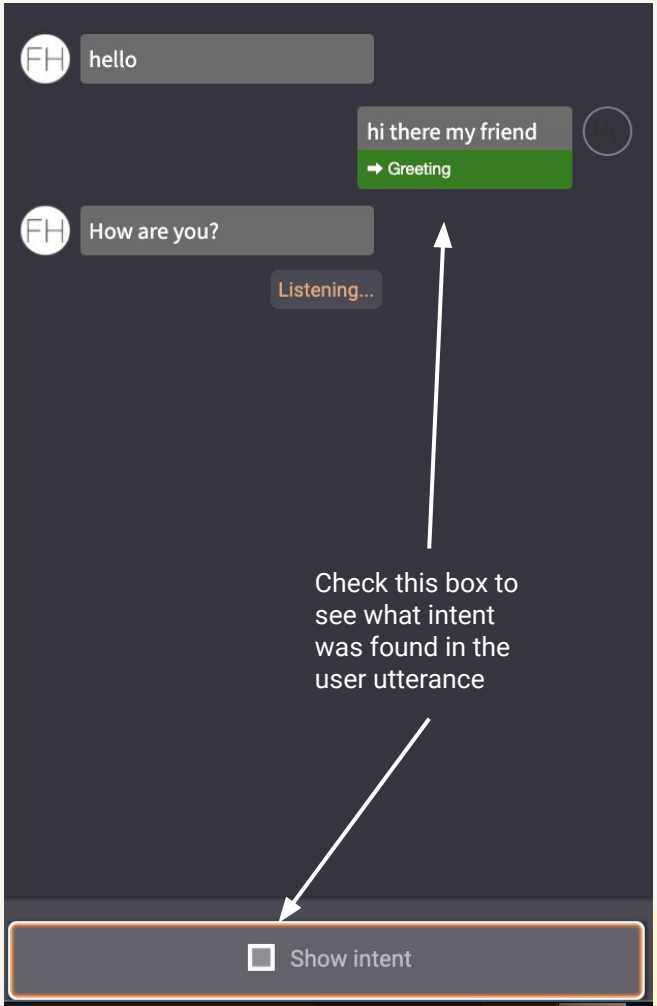
Add a built-in *intent* for greetings.



Now it will not only trigger on the exact phrase “hello”, but instead it will trigger on many different greetings and it is more tolerant for other words before and after.

Try out the interaction a couple of times and see what other greetings works.



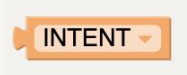


Robust - step 5b

Change the user trigger “I feel great” to a trigger with a slot for an intent.

This time we are going to define our own intent instead of using a pre-defined intent.

Attach an intent variable.



We have not yet defined our intent - so we will have to wait before we can point to our intent.

Robust - step 5c

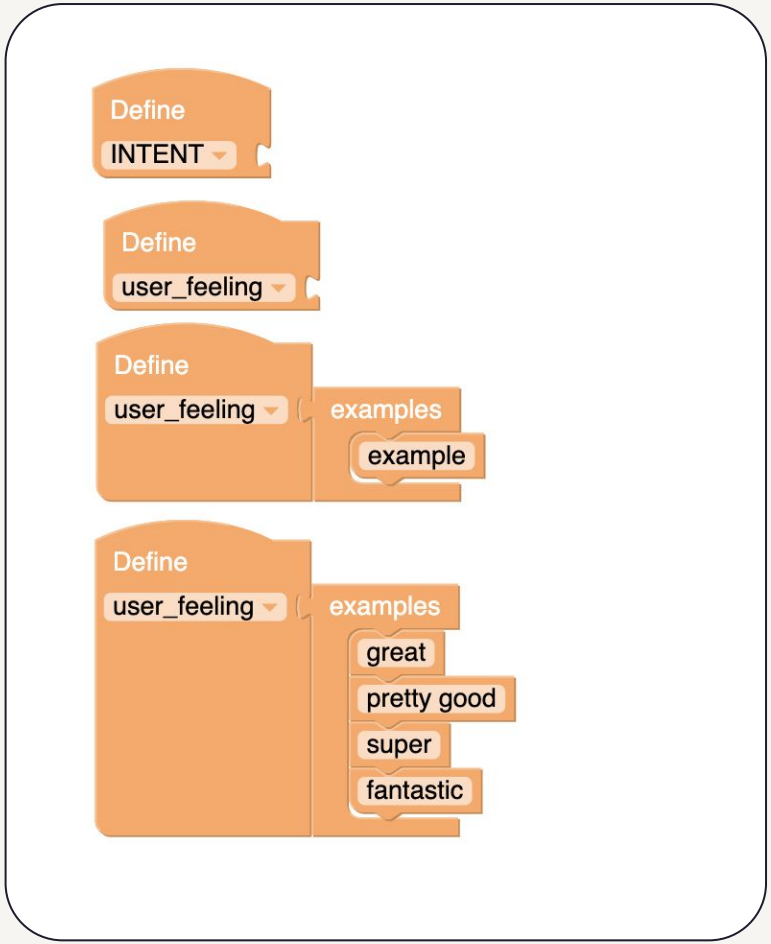
Add a box to ‘Define intent’

Name your intent: user_feeling

Add block of examples to the intent definition.

Add more examples of responses from the user.

The robot will use the examples to learn what phrases to listen for and associate with our intent.



Robust - step 5d

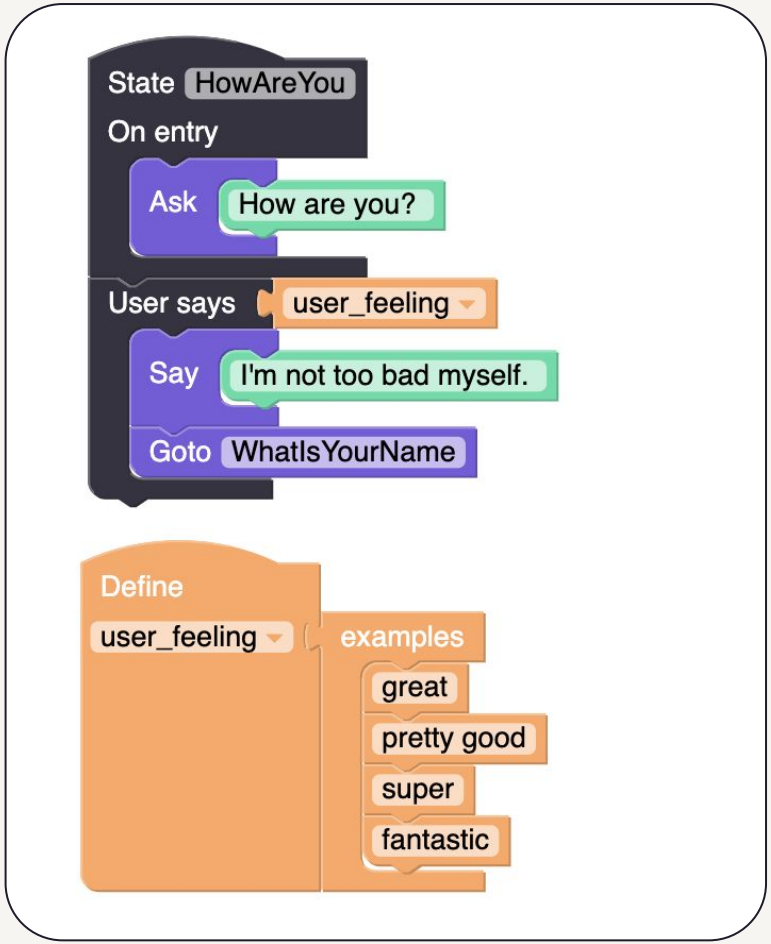
Change the variable of our intent to our newly created ‘user_feeling’-intent.

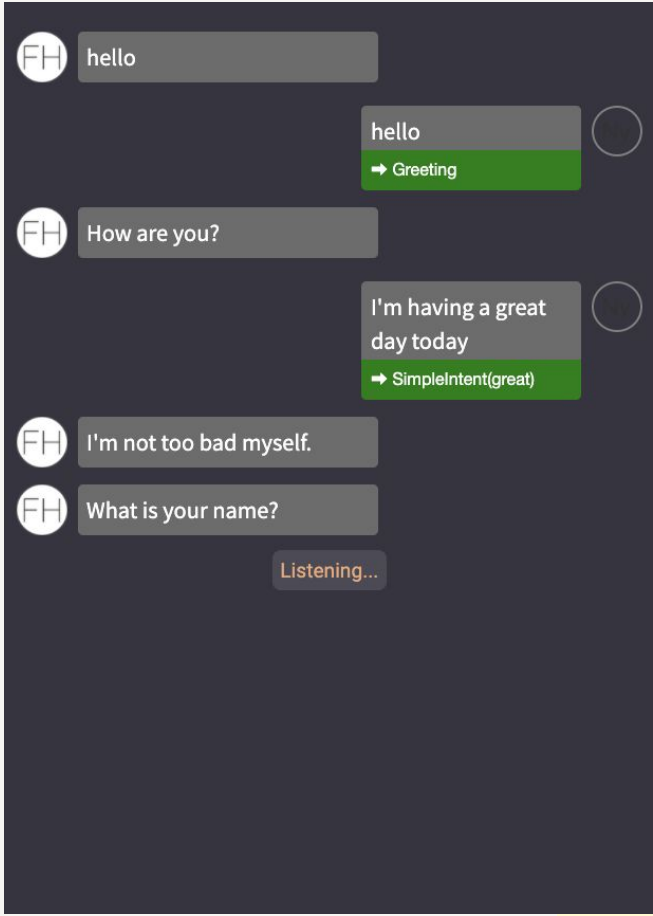
In our intent we used single keywords. This will make this intent trigger very easy - as long as the user utterance contains any of the word it will match to this intent.

So we can say things like:
“I’m having a **great** day today”

If we add longer phrases, we can make sure to only match this intent when the words are uttered in a context. E.g
“having a ___ day”
“I feel ____”
“I’m _____”

Try out the interaction and test adding different examples and see how the intent will match.

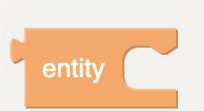




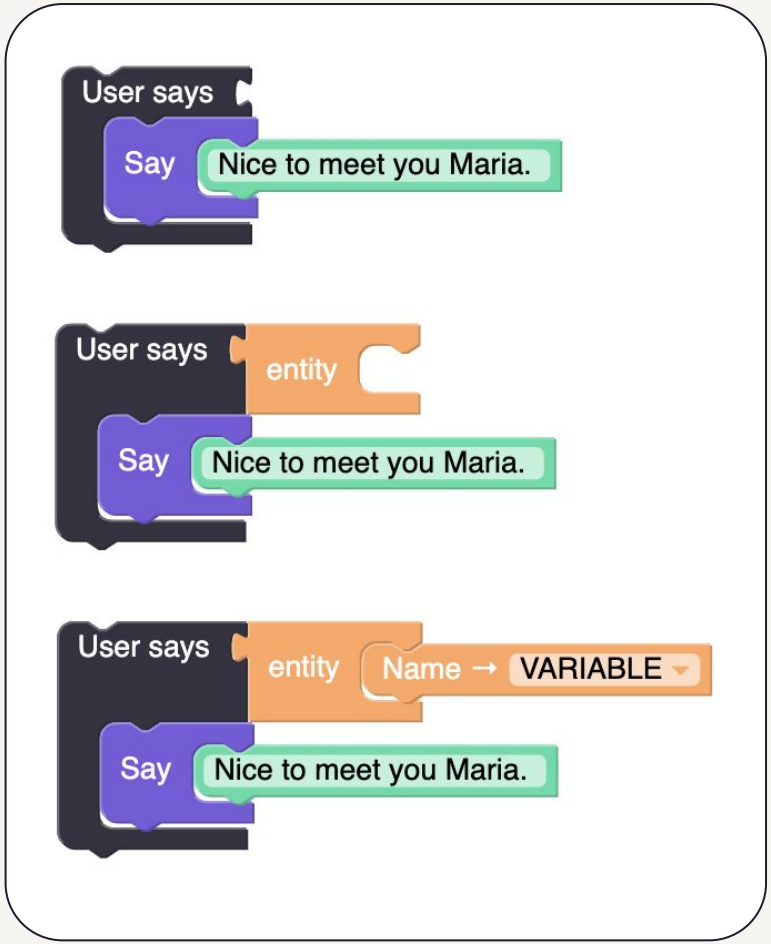
Robust - step 5e

Change the user trigger “my name is maria” to use a trigger where we can attach an *intent*.

We want to add an intent of a user saying a name. A name can be defined as an *entity*.



And we can use the pre-made name entity that contains a list of common english names.



Robust - step 5f

Whatever name the user says*, we can store it on a variable and use it later to repeat the name back to the user.

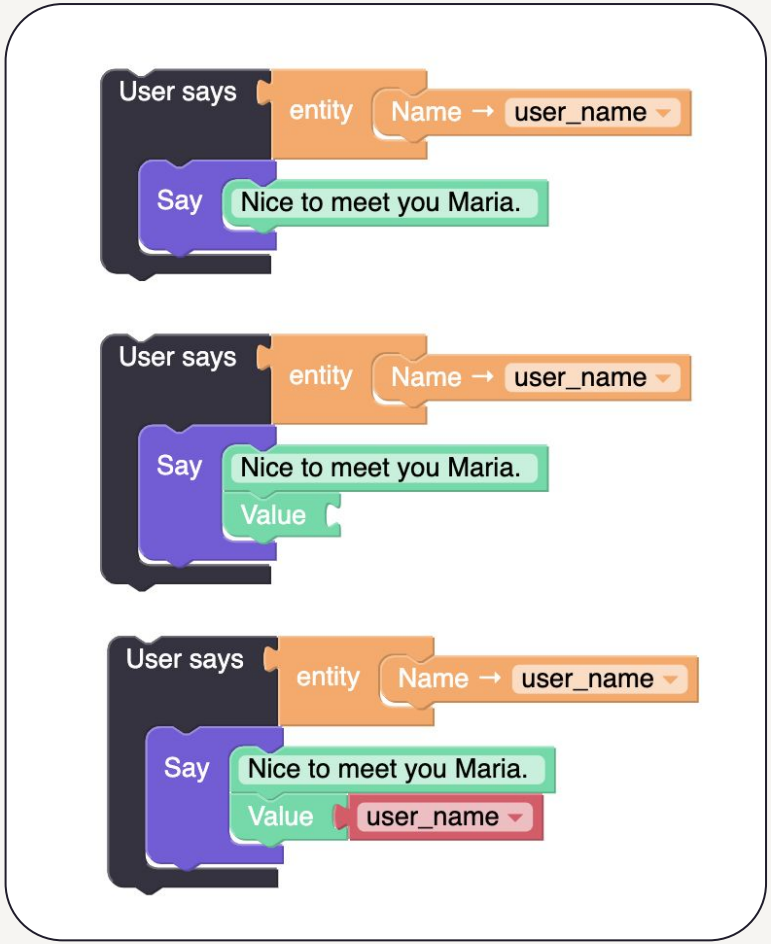
Give the the name entity variable the name: ‘user_name’

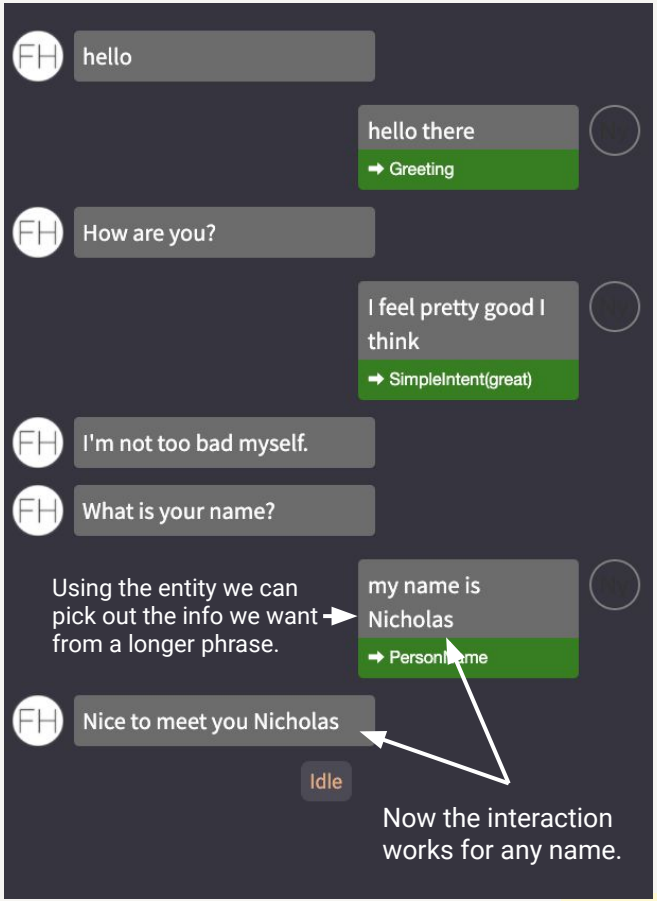
Remove ‘maria’ from “Nice to meet you Maria”, and add an empty ‘value’ block instead.

Add a variable block and attach it to the value block.

Select ‘user_name’ as the variable.

*Note that not all names are covered in the list of names defined in the entity





Making the interaction more **robust**

Step 4 - Break up the interaction into multiple states

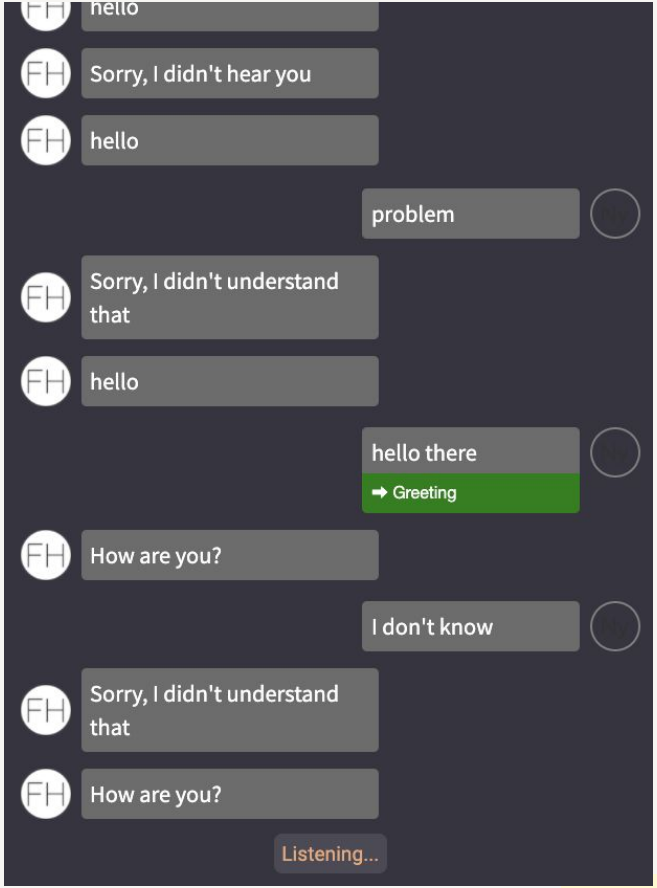
Step 5 - Change our triggers to respond to intents rather than exact phrases

Step 6 - Add error handling when user is silent or says something else

Robust - step 6

When testing the interaction you have probably heard a lot of “Sorry, I didn’t hear you” and “Sorry I didn’t understand that” and the robot repeating this mechanically. You feel like you are stuck in a loop (you are!) and it’s not a nice experience.

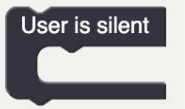
Let’s make sure that never happens in our interaction.



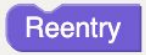
Robust - step 6a

When user is silent in the start of the interaction - instead of bothering the user with questions - we will hand over the initiative to the user and let the robot be quiet instead (but still listening).

Add a trigger for ‘user is silent’ to the start state.

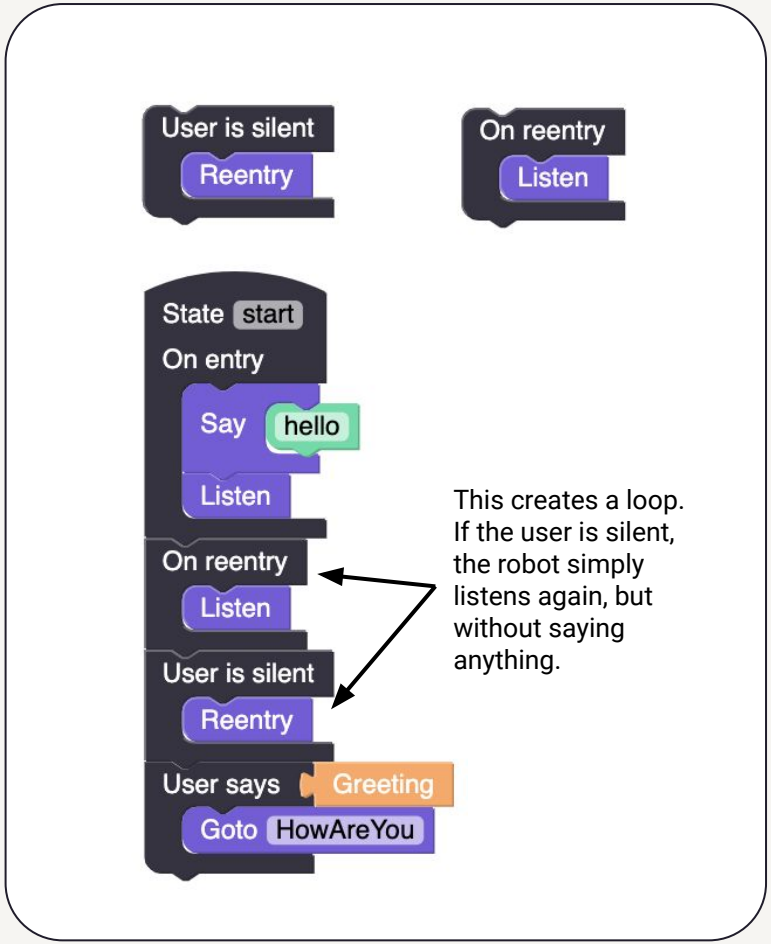


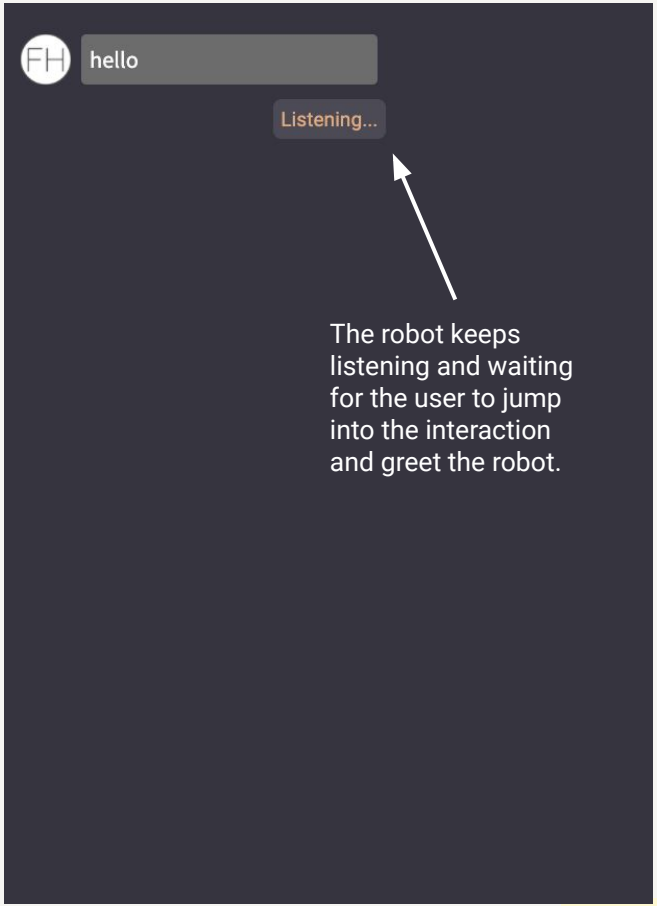
As the action - do a transition of type ‘reentry’



Add an other trigger ‘on reentry’

Just add a listen.

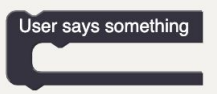




Robust - step 6b

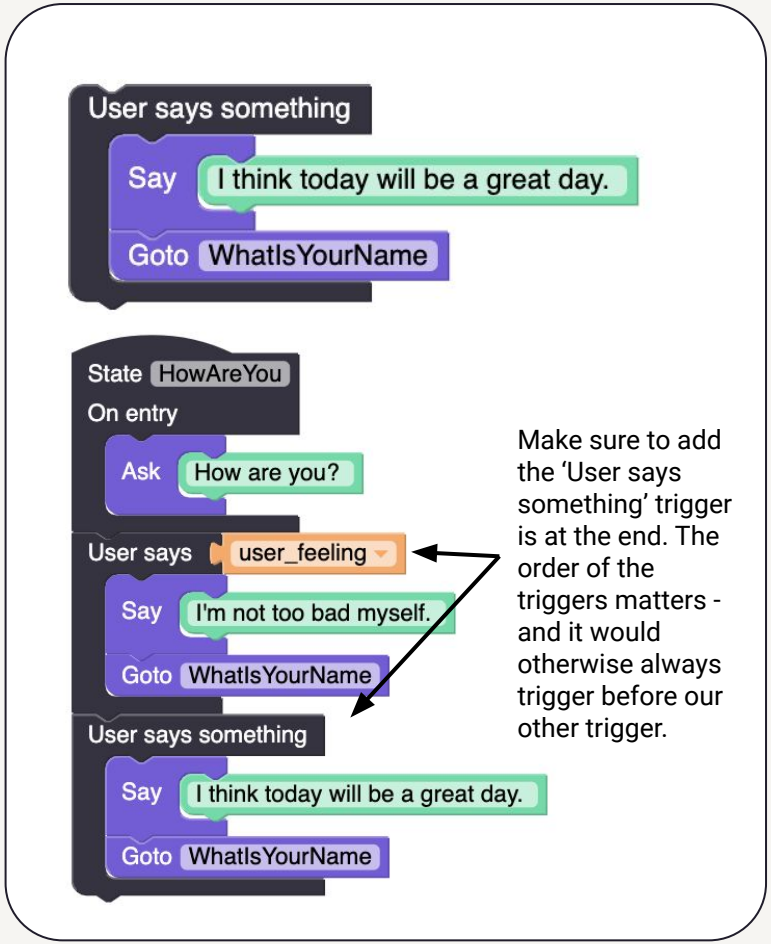
We ask the user how they are feeling - but it does not really matter what they answer. So we can make sure that no matter what they say we will make sure to keep the interaction moving and not let them get stuck.

Add a trigger for ‘user says something’ to the HowAreYou state.

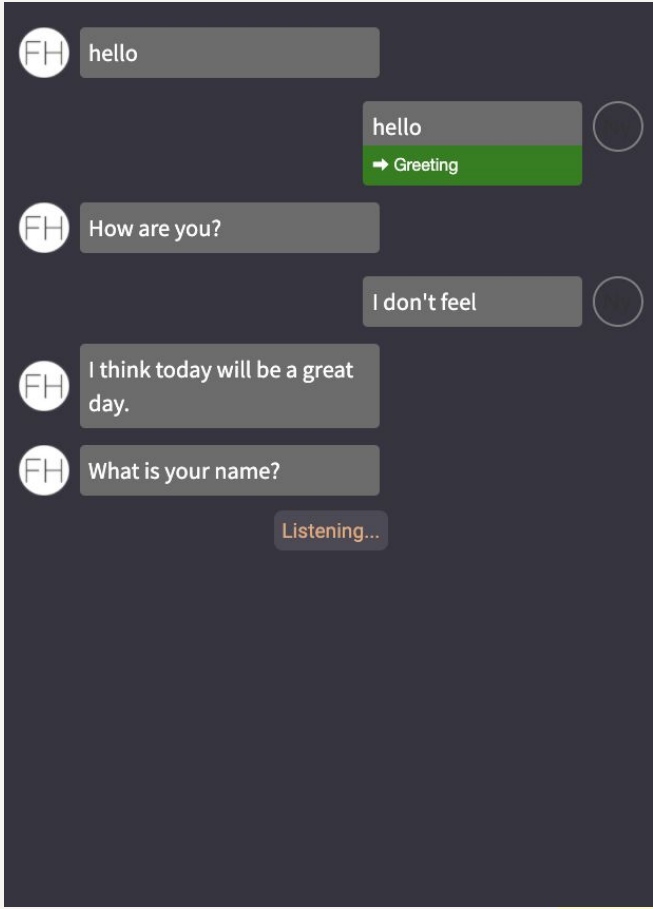


Say something that won’t sound strange regardless of what the user said.

Add a transition to the next state - WhatIsYourName



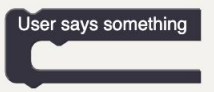
Make sure to add the ‘User says something’ trigger is at the end. The order of the triggers matters - and it would otherwise always trigger before our other trigger.



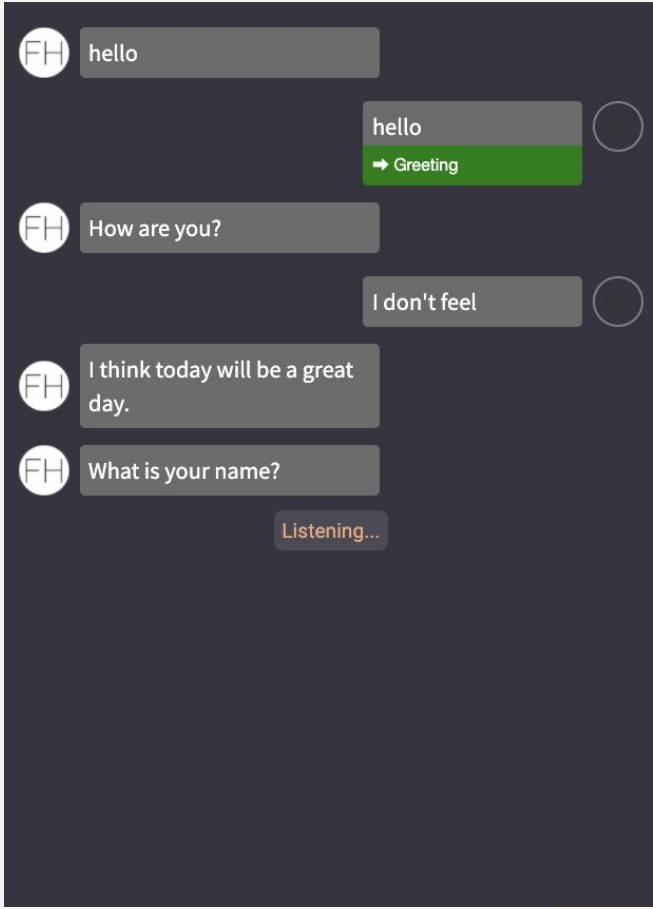
Robust - step 6c

The name list is not complete - so there is a chance the users says a name we don't understand.

We'll add a 'user says something' to catch user names we don't know to the WhatIsYourNameState



Add a nice error handler that will make the user smile instead of feeling frustrated.



Nice job!

You now have created an interaction that...
... has a good flow
... is dynamic
... handles common errors

Are we done now?

Of course not!

We need to make the interaction more natural and delighting.

Making it more **delightful**

Right now we have an interaction that is robust but boring and unengaging...

... it does not include any facial gestures
... it does not try to look at the user
... it does not have any character traits or funny quirks

So in the following steps we will:

Step 7 - Add smiles and other facial expressions

Step 8 - Add actions to *attend* the user and respond when the user enters

Step 9 - Give it some character

Making the interaction more **delightful**

Step 7 - Add smiles and other facial expressions

Step 8 - Add actions to attend the user and respond when the user enters

Step 9 - Give it some character

Delightful - step 7

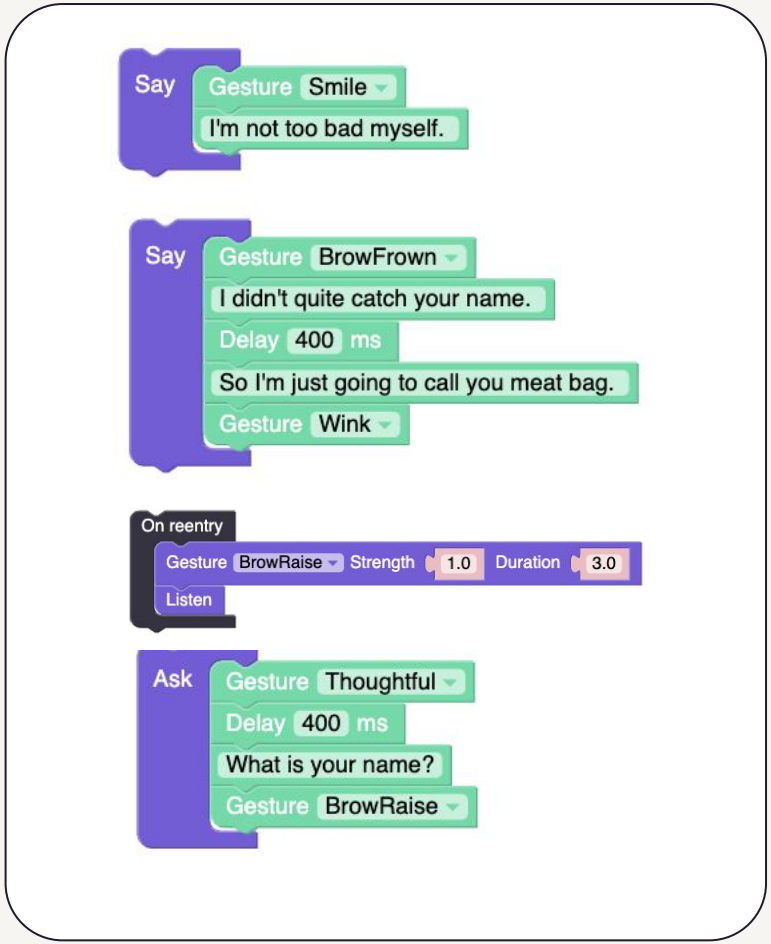
Add smiles after user has said something.

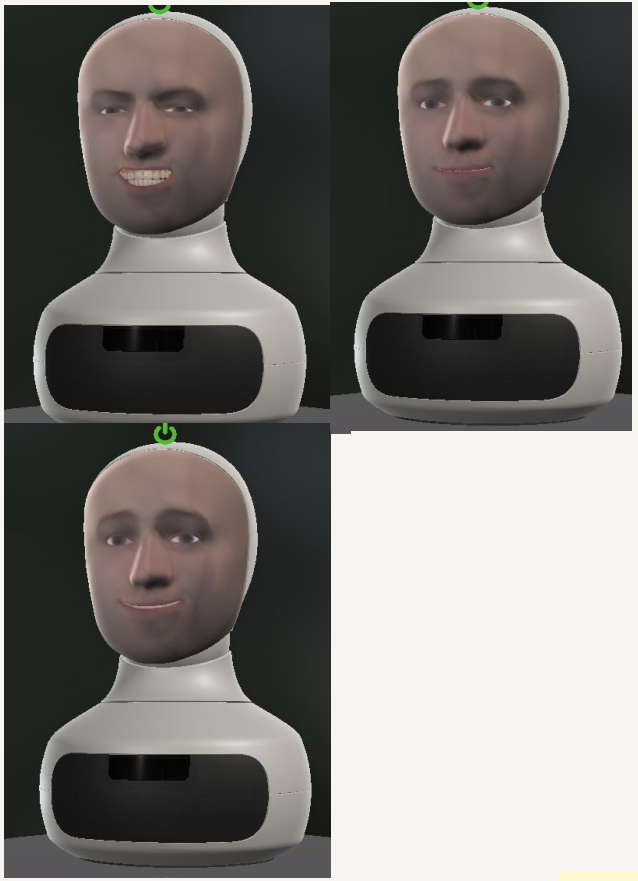
Add raise eyebrows after robot asked a question.

The green block in the ‘utterance’ category can only be placed in a say/ask block.

Use the purple gesture block to add gestures at other locations - and use the options to change the prominence and speed.

Add delays to make sure the robot has time to complete the gesture. And add delays to add natural pauses in the robots speech.





Making the interaction more **delightful**

Step 7 - Add smiles and other facial expressions

Step 8 - Add actions to attend the user and respond when the user enters

Step 9 - Give it some character

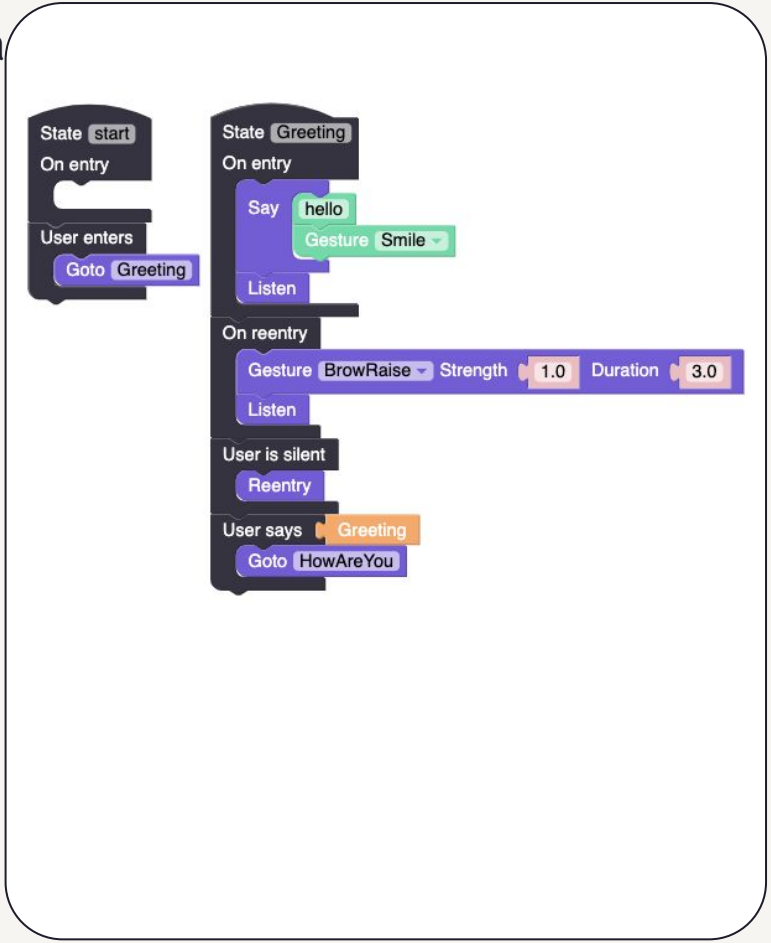
Delightful - step 8a

Currently the interaction starts as soon as we press ‘play’. In a real scenario the interaction wouldn’t start unless there was a user present.

Create a new state and name it Greeting. Move all the greeting to the new state, so our start state is completely empty.

Add a ‘user enters’ trigger to the start state.

Add a transition to the greeting state.



Delightful - step 8b

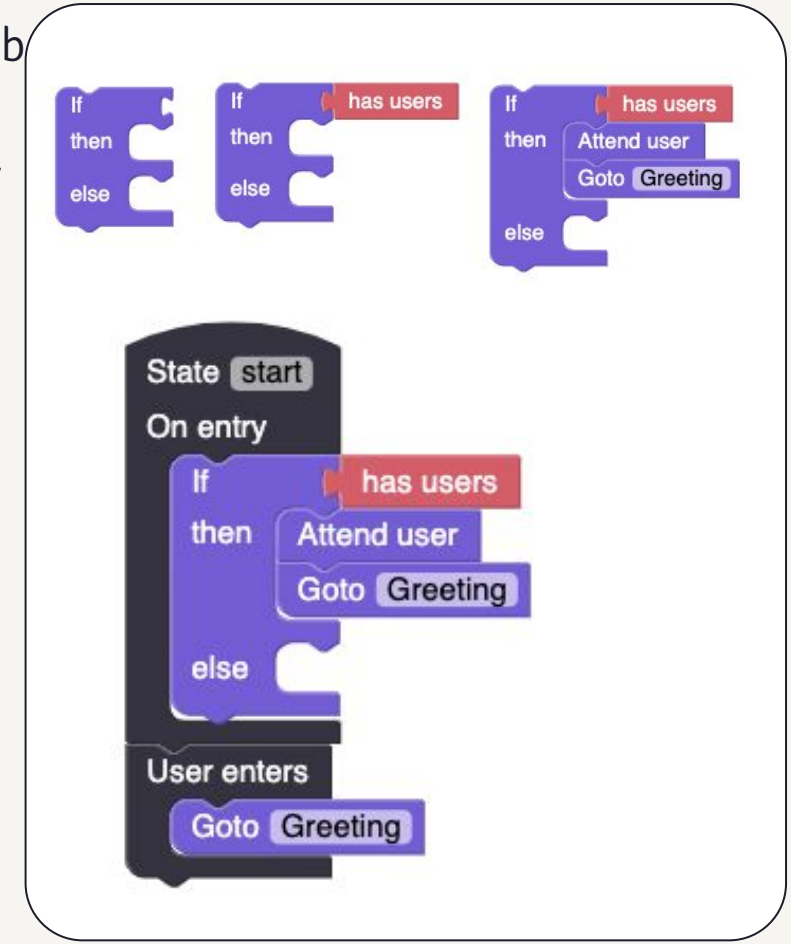
When the interaction starts - we need to do a check if we already have a user present.

Add a control block of the type if-then-else to the ‘on entry’ of our start state.

Add ‘has users’ as the if-condition

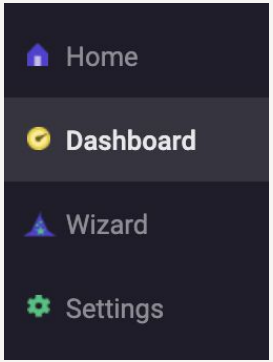
On true / then: add an ‘attend user’ block and a transition to the greeting state

On false / else: do nothing



Delightful - step 8c

Test the interaction by adding a virtual user in the dashboard of the web interface.



Making the interaction more **delightful**

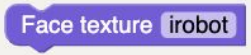
Step 7 - Add smiles and other facial expressions

Step 8 - Add actions to attend the user and respond when the user enters

Step 9 - Give it some character

Delightful - step 9a

Select face for the character in the interaction.

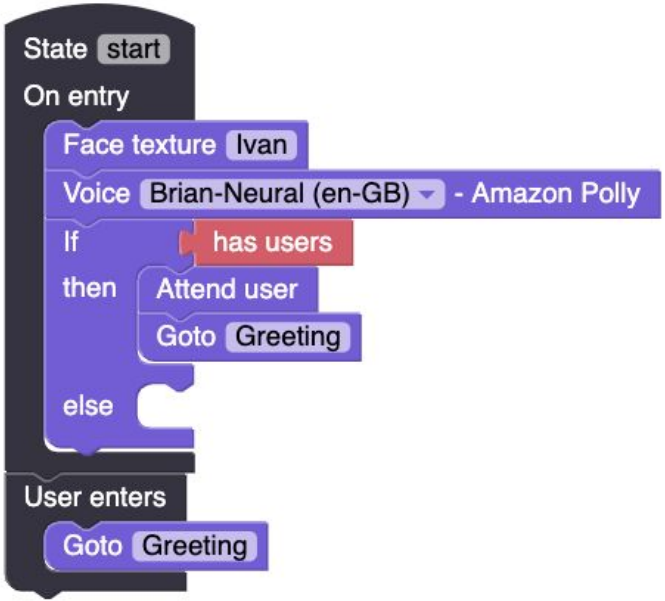


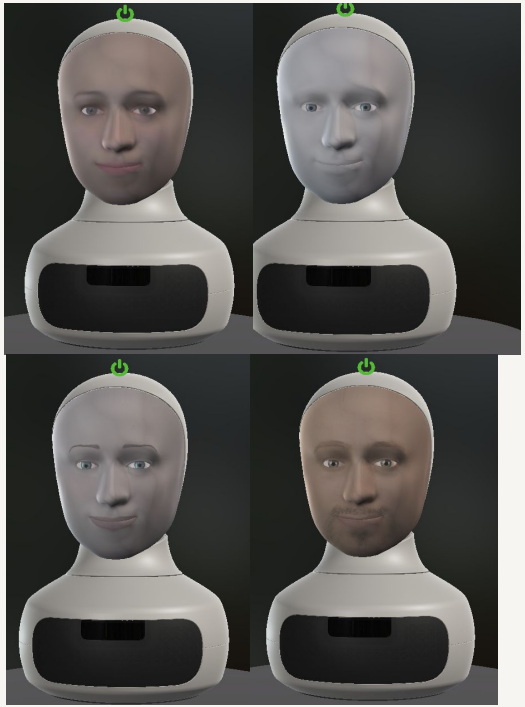
Try different faces from the home tab of the web interface.

Select a voice for the robot.

Select voices and try them out using the home tab and the ‘settings - voice’-tab of the web interface.

Add them to the ‘on entry’ of our start state.



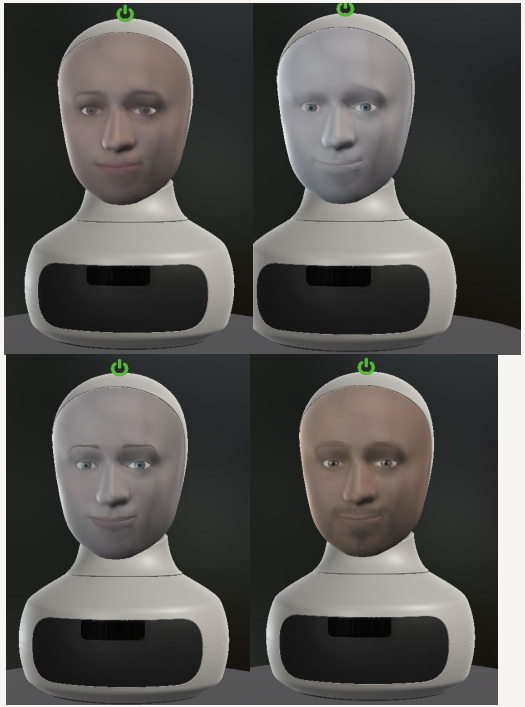


Delightful - step 9b

Change the script to try to give the robot a character that fits well with the face and voice.

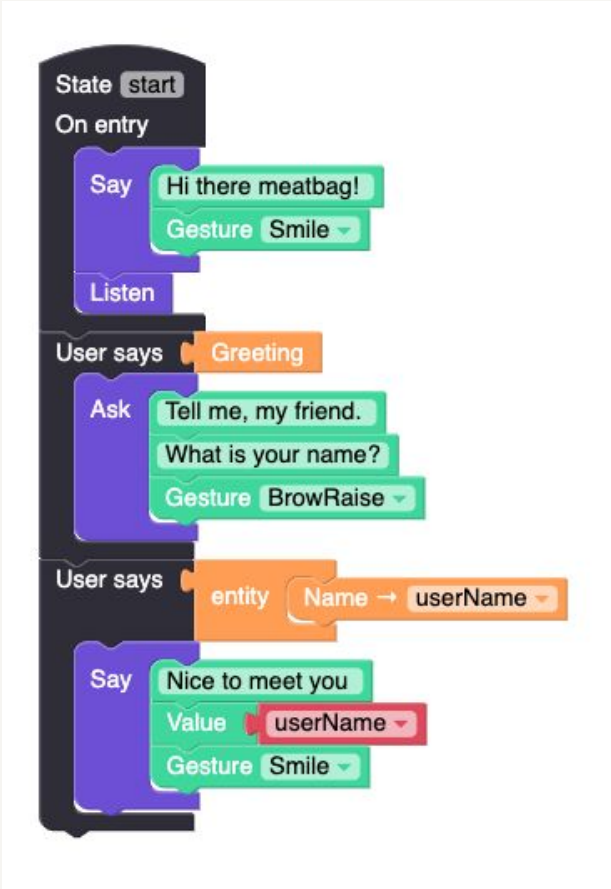
The voice Brian-neural has a nice dry english that fits well with sarcasm.

Together with the 'Ivan' face we can create a sarcastic robot that dislikes humans.



If you want to keep going

- **Robot initiative**
- **No natural start of interaction**
- **No natural end**
- **No error handling**



Nicely done!

You now have created an interaction that...
... uses multi-modal communication (uses facial expressions and voice)
... has social awareness (reacts to users)
... has a character

For further reading - if you want :)

A couple of things to consider

Initiative

- **How is the interaction initiated?**
- **Who carries the initiative in the conversation – or is it a mixed initiative?**

Natural and delighting

- **How to make the robot “natural”**
- **How to make the UX delightful**

Error handling and recovery

- **What happens when user is silent?**
- **Avoiding robot “death”**

A couple of things to consider

Initiative

- **How is the interaction initiated?**
- **Who carries the initiative in the conversation – or is it a mixed initiative?**

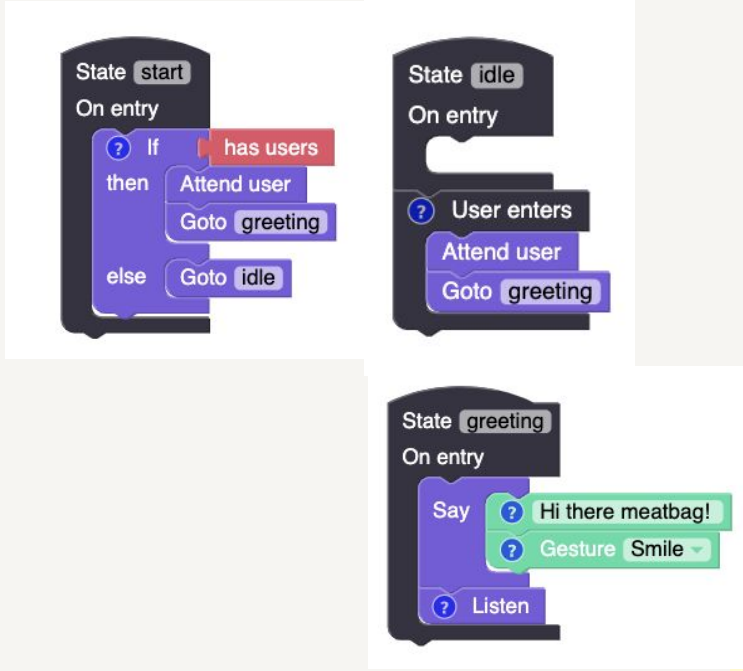
Natural and delighting

- How to make the robot “natural”
- How to make the UX delightful

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

Start simple with the robot taking initiative on user enter.



A couple of things to consider

Initiative

- **How is the interaction initiated?**
- **Who carries the initiative in the conversation – or is it a mixed initiative?**

Natural and delighting

- How to make the robot “natural”
- How to make the UX delightful

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

Make sure to have a end state where the interaction will have a natural ending. And not just leave the robot doing a zombie stare.

A couple of things to consider

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

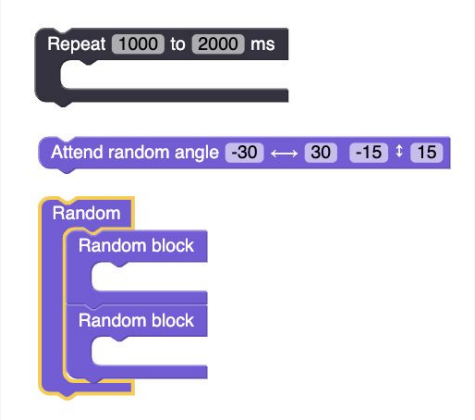
Natural and delighting

- **How to make the robot “natural”**
- **How to make the UX delightful**

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

Add random movement and behaviour using random blocks and repeating triggers



A couple of things to consider

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

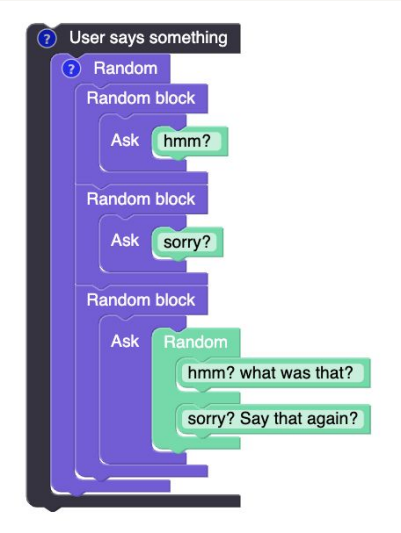
Natural and delighting

- **How to make the robot “natural”**
- **How to make the UX delightful**

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

Add variation to the robots speech using random blocks. Always use variation on phrases that might get repeated!



A couple of things to consider

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

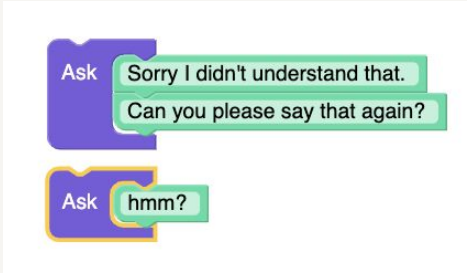
Natural and delighting

- **How to make the robot “natural”**
- **How to make the UX delightful**

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

Avoid mechanical dialogue and keep it short!



A couple of things to consider

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

Natural and delighting

- **How to make the robot “natural”**
- **How to make the UX delightful**

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

Add pauses and facial expressions

A couple of things to consider

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

Natural and delighting

- **How to make the robot “natural”**
- **How to make the UX delightful**

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

The robot always have character. If you don’t give the robot a character it’s going to get the default character - that is bland and boring!

A couple of things to consider

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

Natural and delighting

- **How to make the robot “natural”**
- **How to make the UX delightful**

Error handling and recovery

- What happens when user is silent?
- Avoiding robot “death”

Add something that might surprise the user a bit.

Getting them to react and perhaps getting a little smile can do wonders for the engagement!

A couple of things to consider

Note! This block has been replaced in Blockly update!

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

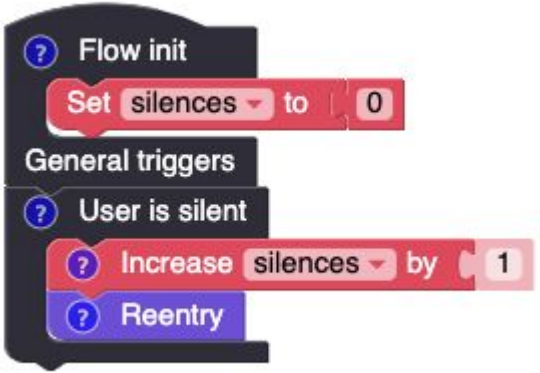
Natural and delighting

- How to make the robot “natural”
- How to make the UX delightful

Error handling and recovery

- **What happens when user is silent?**
- **Avoiding robot “death”**

Use the built-in state ‘Flow init’ to set global commands and global behaviour.



A couple of things to consider

Initiative

- How is the interaction initiated?
- Who carries the initiative in the conversation – or is it a mixed initiative?

Natural and delighting

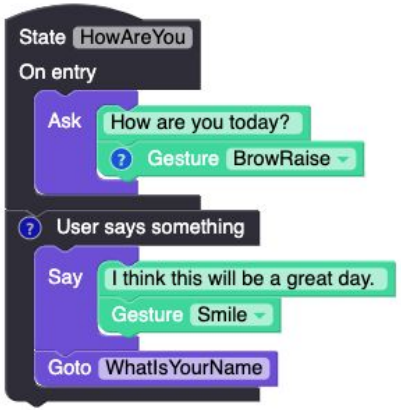
- How to make the robot “natural”
- How to make the UX delightful

Error handling and recovery

- **What happens when user is silent?**
- **Avoiding robot “death”**

Never get stuck. Always keep the interaction moving using smart dialogue and the trigger ‘User says something’.

This is especially true for dialogue turns that are not required to complete the task.



Tips and tricks

Tips and tricks

- **Use** Test selected **to try out how it sounds.**
- **Use ‘main’ in web interface to find a face, and use ‘settings’ - ‘voice’ to find a voice**
- **Use ‘call’ transitions and immediate ‘terminate’ to create functions**