

## Manual – Furhat Question-Response Script

The following steps explain the process of how to edit and use the Furhat script template to customise Furhat's responses based on your research needs and list of optional questions that you want participants to be able to ask:

- 1) It is recommended to use *IntelliJ IDEA* to open and work on the script.

Here you can find a download link to the software:

<https://www.jetbrains.com/idea/download/?section=windows>

You can get a free license for educational purposes:

<https://www.jetbrains.com/community/education/#students>

- 2) Open the file called *"empty-template.kt"* in IntelliJ IDEA to view and edit the script.
- 3) Change *speechRate* (recommended value: 1.0) and *introTimeoutInSeconds* (recommended value: 10) to your preferred tempo of speech and time the intro is supposed to take.

```
7     val emptyConversation = Conversation(  
8         speechRate = 0.0,  
9         introSpeech = utterance{+""},  
10        introTimeoutInSeconds = 0,
```

- 4) Fill in the blank in *introSpeech = utterance {+" "}* to tell Furhat how to welcome the participant. You may also add a gesture with the *Gesture.* command and more than one sentence using *+*. See example below:

```
7     val myConversation = Conversation(  
8         speechRate = 1.0,  
9         introSpeech = utterance{ +"Good day, how good to see you!"  
10            +Gestures.Smile  
11            +"Today I will help you answer your questions about your..." },  
12        introTimeoutInSeconds = 10,
```

- 5) Fill in the blank in *areYouReadyQuestion = utterance{+" "}* to indicate what Furhat should ask to get to know if the participant is ready to begin asking questions. For instance:

```
13        areYouReadyQuestion = utterance{ +"Are you ready to begin?"},
```

- 6) Fill in the blank in `clarifyQuestion = utterance{+" "}` to define what Furhat should ask when unsure what the participant said. For instance:

```
14 clarifyQuestion = utterance{ +"I didn't hear that. Could you please repeat it?"},
```

- 7) Fill in the blank in `infoQuestion = utterance{+" "}` to tell Furhat how to indicate that the participant can start asking questions to *ChatGPT*. For instance:

```
15 infoQuestion = utterance{ +"You can ask your question, I will be happy to answer it."},
```

- 8) Fill in the blank in `continueQuestion = utterance{+" "}` to define how Furhat should ask the participant if he wants him to continue with the answer from ChatGPT in case it is a long answer. For instance:

```
16 continueQuestion = utterance{ +"Would you like to know more?"},
```

- 9) Fill in the blank in `noResponse = utterance{+" "}` to indicate what Furhat should say when the participant is not responding. For instance:

```
17 noResponse = utterance{ +"I didn't hear you, can you repeat that one time"},
```

- 10) Specify the response given by *ChatGPT* in case the participant wants to know something besides the standard list of questions by filling in the blank in `chatGptInstructions = " "`. For instance:

```
chatGptInstructions = "Only search for information on the following websites:... Keep the answer concise and easy."
```

- 11) Fill in the blank in `chatGptThinking = utterance{+" "}` to tell Furhat what to say while ChatGPT is searching for results. For instance:

```
19 chatGptThinking = utterance{ +"Let me think"},
```

- 12) Fill in the blank in `chatGptOtherQuestion = utterance{+" "}` to define how Furhat should ask the participant if he has another question unrelated to the standard questions. For instance:

```
chatGptOtherQuestion = utterance{ +"Do you have any other questions? Just ask or just say no."},
```

- 13) Adjust the listening time of Furhat by changing the value of `listenTimeoutInSeconds` (recommended value: 3)

```
21 listenTimeoutInSeconds = 3,
```

- 14) The dialogue with *id* = "0" is intended as an introduction (number is interchangeable; used to identify where to go next). Fill in the blank in *question* = *utterance*{+" " } to let Furhat start the conversation and ask the participant if they are ready. For instance:

```
{+"Before you will see a list of standard questions. Take your time to read the list carefully. If you are ready say I am ready. Otherwise say I am not ready."},
```

Then add answer options to the question below. For instance:

```
23     Dialogue(  
24         id = "0",  
25         question = utterance{+"Before you will see a list of standard questions. T  
26         answers = listOf(  
27             Answer("I'm done", utterance{+"Perfect"}, "1"),  
28             Answer("Yes", utterance{+"Perfect"}, "1"),  
29             Answer("I'm not ready", utterance{+"It's okay, take your time"}, "0"),  
30             Answer("No", utterance{+"It's okay, take your time"}, "0"),  
31         )  
32     ),
```

If the participant answers "I'm done" or "Yes" Furhat will continue with the dialogue with *id* = "1".

- 15) The dialogue with *id* = "1" has the purpose of giving instructions to the participant to choose a question from the list or alternatively ask a question to ChatGPT. Fill in the blank in *question* = *utterance*{+" " } with this instruction. For instance:

```
{+"You can name a number of a standard question and I will answer it. If you have questions outside the standard questions, say: other question."},
```

You can specify your answers based on your list of questions below. At the end of each bracket change the number to indicate which dialogue should follow next. For instance:

```
),  
Answer("1", utterance{+"The most common complaints of osteoarthritis are pain, stiffness, swelling and difficulty moving the joints. "}, followUps = "3"),  
Answer("2", utterance{+"Osteoarthritis has no clear cause. However, you can pay attention to your lifestyle. This way you can prevent the symptoms from worsening."}, followUps = "3"),  
Answer("3", utterance{+"The most common complaints of osteoarthritis are pain, stiffness, swelling and difficulty moving the joints."}, followUps = "3"),
```

Fill in the blank in *userInput* and *followUps* to tell Furhat after what response it should initiate a *ChatGPT* search. For instance:

```
Answer(  
    userInput = "Other question",  
    response = utterance{+"You have chosen a different question."},  
    followUps = "chatGPT",  
    redirect = "3"  
)
```

## Script Template - Empty

```
package furhatos.app.template.flow.main
import furhatos.flow.kotlin.utterance
import fur hats.gestures.Gestures

val emptyConversation = Conversation(
    speechRate = 0.0,
    introSpeech = utterance{+""},
    introTimeoutInSeconds = 0,
    areYouReadyQuestion = utterance{+""},
    clarifyQuestion = utterance{+""},
    infoQuestion = utterance{+""},
    continueQuestion = utterance{+""},
    noResponse = utterance{+""},
    chatGptInstructions = "",
    chatGptThinking = utterance{+""},
    chatGptOtherQuestion = utterance{+""},
    listenTimeoutInSeconds = 0,
    dialogues = listOf(
        Dialogue(
            id = "0",
            question = utterance{+""},
            answers = listOf(
                Answer(" ", utterance{+""}, followUps = "1"),
                Answer(" ", utterance{+""}, followUps = "1"),
                Answer(" ", utterance{+""}, followUps = "0"),
                Answer(" ", utterance{+""}, followUps = "0"),
            )
        ),
        Dialogue(
            id = "1",
            question = utterance{+""},
            answers = listOf(
                Answer(
                    userInput = "",
                    response = utterance { +""},
                    followUps = "chatGPT"
                    redirect = "2"
                )
            ),
        Answer("1", utterance{+""}, followUps = "2"),
        Answer("2", utterance{+""}, followUps = "2"),
        Answer("3", utterance{+""}, followUps = "2"),
        Answer("4", utterance{+""}, followUps = "2"),
        Answer("5", utterance{+""}, followUps = "2"),
        Answer("6", utterance{+""}, followUps = "2"),
        Answer("7", utterance{+""}, followUps = "2"),
        Answer("8", utterance{+""}, followUps = "2"),
    )
)
```