# **Personalized Social Interaction with Furhat Robot**

## Overview

This project enables a personalized, multi-user conversational experience using the **Furhat robot**. It leverages:

* **Voice recognition** for user identification
* **Emotion detection** for empathetic responses
* **Dynamic topic selection** for engaging conversations

The robot intelligently switches attention between multiple users, responding based on individual emotions and preferences.

## Features

* **User Identification**  
  Recognizes and greets users based on voice, enabling a personalized experience.
* **Emotion Detection**  
  Detects emotional cues in user speech and responds empathetically using emotion keyword analysis.
* **Dynamic Topic Selection**  
  Suggests topics for discussion and adapts based on user acceptance or rejection.
* **Multi-User Interaction**  
  Seamlessly shifts attention between multiple users during interaction.
* **Godspeed and RoSAS Feedback**  
  Collects standardized Human-Robot Interaction (HRI) feedback to evaluate robot performance.

## Installation

### 1. Requirements

* [Furhat SDK](https://furhatrobotics.com) installed
* IntelliJ IDEA (or other Kotlin-supported IDE)
* Voice recognition model
* Basic knowledge of Kotlin programming

### 2. Setup

**# Clone the tutorial repository**git clone https://github.com/FurhatRobotics/tutorials.git

* Use the audiofeed demo tutorial from the repository. Copy the necessary packages to enable audio recording.
* Ensure your voice model and speaker dataset are in the correct directory and accessible by the Furhat robot.

### 3. Running the Project

* Open the project in IntelliJ or your preferred IDE.
* Use FurhatAudioFeedRecorder to record user and robot audio into .wav files.
* On the Furhat web interface, go to Settings > External Feeds and enable audio feed.
* In the main.kt file:
  + Set the IP address of your robot (ensure it’s on the same network).
  + If necessary, update the IP in the audio streamer configuration using streamer.start().
* Run the main interaction state from the Furhat SDK.

## Usage

### 1. Voice Identification

* On first interaction, the robot will prompt for your name.
* The system uses voice\_model.py to recognize and welcome returning users.

### 2. Emotion Detection

* Speak naturally with the robot.
* It detects keywords linked to emotions like **happiness**, **sadness**, etc., and tailors its responses accordingly.

### 3. Dynamic Topic Recommendation

* The robot suggests random topics.
* If the user accepts, it proceeds with the conversation. If not, it offers another topic.

## Feedback Collection: Godspeed & RoSAS Ratings

This system evaluates the interaction quality using two standardized HRI frameworks:

### Godspeed Attributes

* **Anthropomorphism** – How human-like Furhat appears
* **Animacy** – How lifelike Furhat’s behavior feels
* **Likeability** – Enjoyment of interacting with Furhat
* **Perceived Intelligence** – Smartness and reasoning
* **Safety** – Sense of comfort and safety during interaction

### RoSAS Attributes

* **Perceived Usefulness** – Helpfulness of Furhat’s responses
* **Emotional Engagement** – Emotional connection during interaction
* **Perceived Adaptivity** – Ability to adapt to different users
* **Enjoyment** – Overall fun and engagement level

## Future Improvements

* Integrate facial expression recognition
* Add multilingual support
* Improve topic recommendation using machine learning
* Real-time sentiment tracking for deeper emotion understanding

## Technologies Used

* **Furhat SDK** – Conversational framework
* **Kotlin** – For behavior scripting
* **Python** – For voice and emotion models
* **JetBrains IntelliJ IDEA** – Development environment