Chapter 6

Building Chatbots III

Rasa (https://rasa.com/)

- Rasa is an **open-source framework** for building conversational AI applications, including chatbots and voice assistants.
- It provides tools for natural language understanding (NLU), dialogue management, and the creation of complex, contextual conversations.
- Rasa consists of two main components:
  - **Rasa NLU** (Natural Language Understanding)
    - responsible for understanding the user's input.
    - extracts **intents** (what the user wants) and **entities** (specific details in the input).
  - **Rasa Core**
    - handles dialogue management, **making decisions** on
      - what the chatbot should do
      - what the chatbot say next

      based on the conversation's context, using machine learning models and rules.

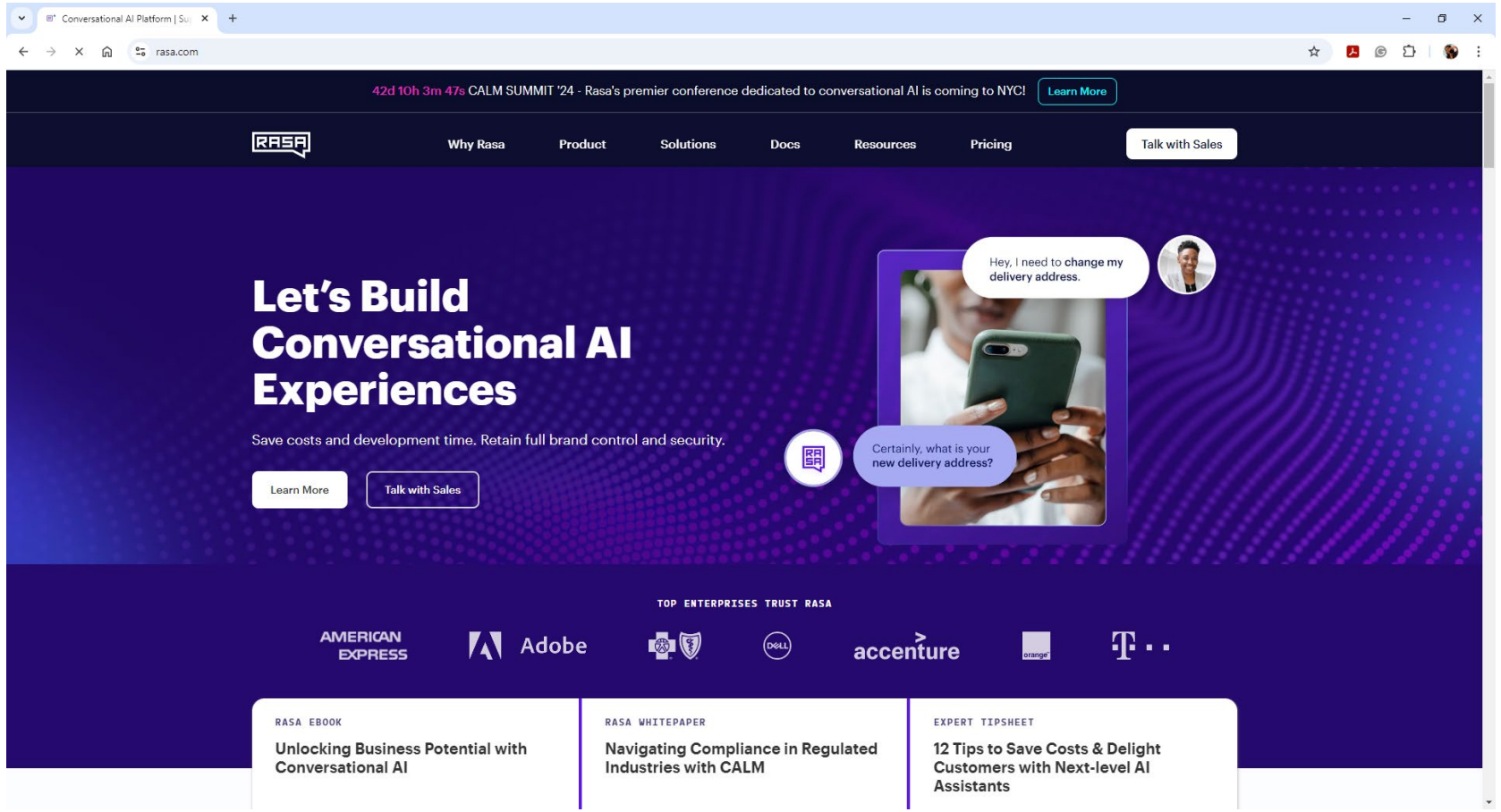
rasa

- main files involved in handling dialogue management:
  - nlu.yml: used to define the training data for Rasa NLU. กำหนดข้อมูลการฝึกสอน
    - This data helps the system learn how to extract intents and entities from user input.
    - Intents represent what the user wants
    - Entities refer to specific information within the input.
  - rules.yml: defines specific rules for conversation paths กำหนดกฎเฉพาะสำหรับเส้นทางการสนทนา
    - can guide the bot's behavior for certain inputs or conditions.
  - stories.yml: contains training data for Rasa Core, ข้อมูลการฝึกสำหรับ Rasa Core (กำหนดเรื่องราวการสนทนา)
    - Define different conversational paths or stories. โมเดลจะเรียนรู้จากตัวอย่างเหล่านี้
    - Each story represents an example of how a conversation could unfold.
  - config.yml: configures the Rasa NLU pipeline and the policies for Rasa Core,
    - manage how the bot predicts the next action based on user input and conversation history.
  - credentials.yml: a configuration file in Rasa
    - used to define the credentials for various messaging platforms and APIs that the Rasa chatbot can connect to.
    - contains the authentication details necessary to integrate the chatbot with external services such as Slack, Facebook Messenger, Twilio, or custom webhooks.
  - domain.yml: the configuration file for Rasa Core. การตั้งค่าสำหรับ Rasa Core เช่น กำหนดเจตนา (intents), การตอบกลับ (responses)
    - Define intents, entities, slots, responses, and actions that the chatbot can perform.
    - Essential for determining how the dialogue is managed.
  - endpoints.yml: specifies the endpoints for external services,
    - such as the Rasa action server or messaging platforms.
  - actions.py: contains custom actions that your bot can perform,
    - such as making API calls or performing calculations.
    - These actions are invoked by Rasa Core when certain conditions are met.

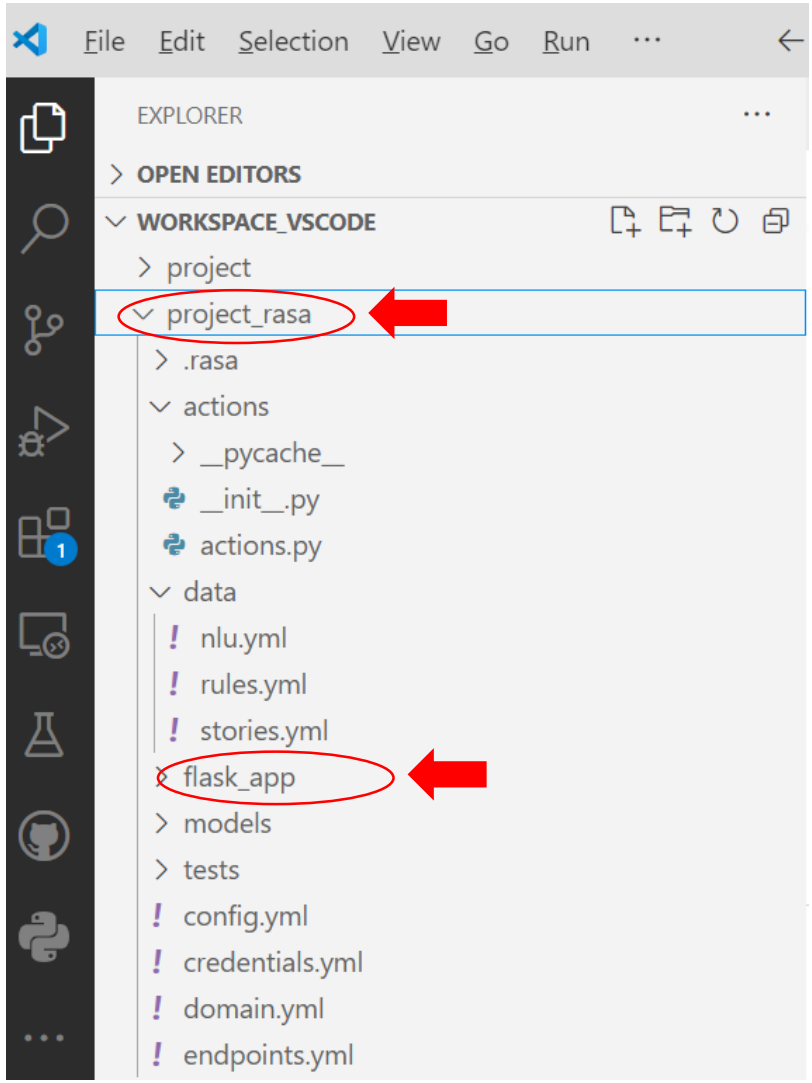
Let's practice!

Install Rasa

- pip install rasa



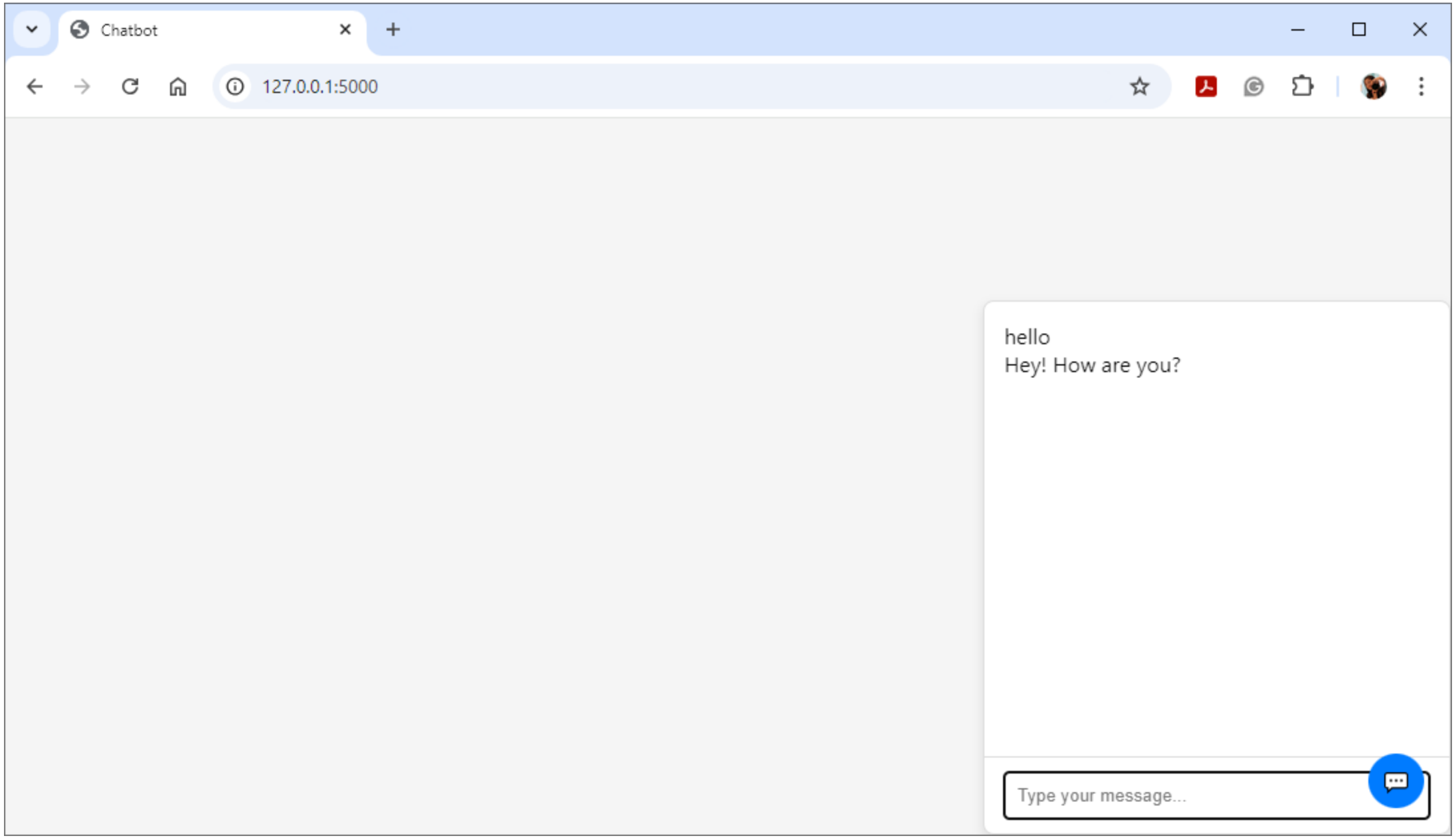
Create folder



- Create a New Rasa Project
- Go to “project_rasa”
  - ../project_rase> **rasa init**
- Interact with your chatbot in the terminal:
  - ../project_rase> **rasa shell**

- Run the Rasa server:
  - ../project_rase> **rasa run**
- Run Flask_app:
  - ../project_rasa/flask_app> python app.py

- Train the model (if not already trained):
  - ../project_rase> **rasa train**


Chatbot
127.0.0.1:5000
hello
Hey! How are you?
Type your message...

Questions