

Setup Report

1. Installed Software, Libraries, and Their Versions:

- **Operating System:** [Specify OS version, e.g., Ubuntu 20.04 / Windows 10]
- **Python:** Python 3.x (Verify using `python --version`)
- **Node.js and npm:** Node.js 16.x, npm 7.x (Verify using `node -v` and `npm -v`)
- **IDE/Text Editor:** Visual Studio Code [or other IDE]
- **Chatbot Framework :** Chosen Framework (e.g., Rasa, Botpress, Dialogflow, or Chatterbot)

Libraries/Dependencies:

- I. **Rasa:** `pip install rasa` (if using Rasa)
- II. **Chatterbot:** `pip install chatterbot chatterbot_corpus` (if using Chatterbot)
- III. **Flask:** `pip install flask` (for web server support)
- IV. **Requests:** `pip install requests` (for HTTP requests)
- V. **Other libraries:** [Add as applicable based on framework]
- VI.

2. Step-by-Step Development Environment Setup:

1. Install Python:

- a. Download and install Python 3.x from the official website.
- b. Verify installation with `python --version`.

2. Set Up Virtual Environment (Recommended):

- a. Create a virtual environment: `python -m venv chatbot_env`.
- b. Activate the environment:
 - i. Windows: `chatbot_env\Scripts\activate`
 - ii. Linux/macOS: `source chatbot_env/bin/activate`

3. Install Required Libraries:

- a. Update pip to the latest version: `pip install --upgrade pip`
- b. Install the necessary libraries for your chosen framework:**
 - i. For Rasa: `pip install rasa`
 - ii. For Chatterbot: `pip install chatterbot chatterbot_corpus`
 - iii. Additional libraries as needed: `pip install flask requests`

4. Install Node.js and npm:

- a. Download and install the latest LTS version of Node.js.
- b. Verify installation with `node -v` and `npm -v`.

5. Install Framework-Specific CLI (e.g., Rasa CLI):

- a. Install Rasa CLI (if using Rasa) to create a new chatbot project:
 - i. `rasa init --no-prompt`

3. Screenshots of Successful Test Runs or Command Outputs:

Include screenshots for each of the following:

- **Python version check: `python --version`**
- **Node and npm versions: `node -v` and `npm -v`**

```
[ ] | pip install python-dateutil # Python version check: python --version

import dateutil # Import dateutil directly

print(dateutil.__version__) # Access the version using __version__

Requirement already satisfied: python-dateutil in /usr/local/lib/python3.10/dist-packages (2.8.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil) (1.16.0)
2.8.2

#Node and npm
| pip install node.js
| pip install npm

ERROR: Could not find a version that satisfies the requirement node.js (from versions: none)
ERROR: No matching distribution found for node.js
Collecting npm
  Downloading npm-0.1.1.tar.gz (2.5 kB)
    Preparing metadata (setup.py) ... done
Collecting optional-django==0.1.0 (from npm)
  Downloading optional-django-0.1.0.tar.gz (9.5 kB)
    Preparing metadata (setup.py) ... done
Building wheels for collected packages: npm, optional-django
  Building wheel for npm (setup.py) ... done
  Building wheel for optional-django (setup.py) ... done
size=3683 sha256=3a790497750ebddf81993628b080b347df3e30ebdd64510c6c777439361d378
074a7b810b0044b9a05fe3e3c04ed088074577b391eb0c
13s completed at 08:43
```

- Virtual environment activation confirmation

```
# prompt: install node and npm and check their version

| npm --version
| node --version

8.19.4
v16.20.2

# prompt: virtual environment activation confirmation

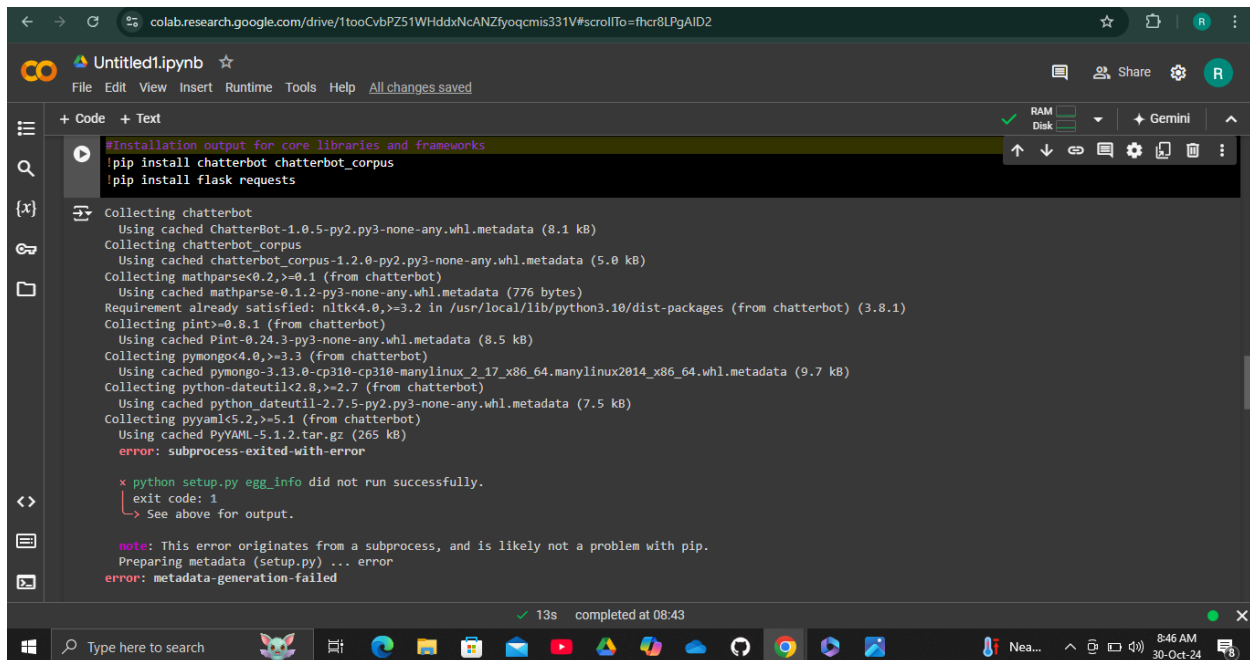
import sys
sys.executable

'/usr/bin/python3'

[1] #Installation output for core libraries and frameworks
| pip install rasa

Requirement already satisfied: rasa in /usr/local/lib/python3.10/dist-packages (3.6.20)
Requirement already satisfied: CacheControl<0.13.0,>=0.12.9 in /usr/local/lib/python3.10/dist-packages (from rasa) (0.12.14)
Requirement already satisfied: PyJWT<3.0.0,>=2.0.0 in /usr/local/lib/python3.10/dist-packages (from PyJWT[crypto]<3.0.0,>=2.0.0->rasa) (2.9.0)
Requirement already satisfied: SQLAlchemy<1.5.0,>=1.4.0 in /usr/local/lib/python3.10/dist-packages (from rasa) (1.4.54)
al/lib/python3.10/dist-packages (from rasa) (1.4.0)
n/local/lib/python3.10/dist-packages (from rasa) (8.2.3)
13s completed at 08:43
```

- Installation output for core libraries and frameworks



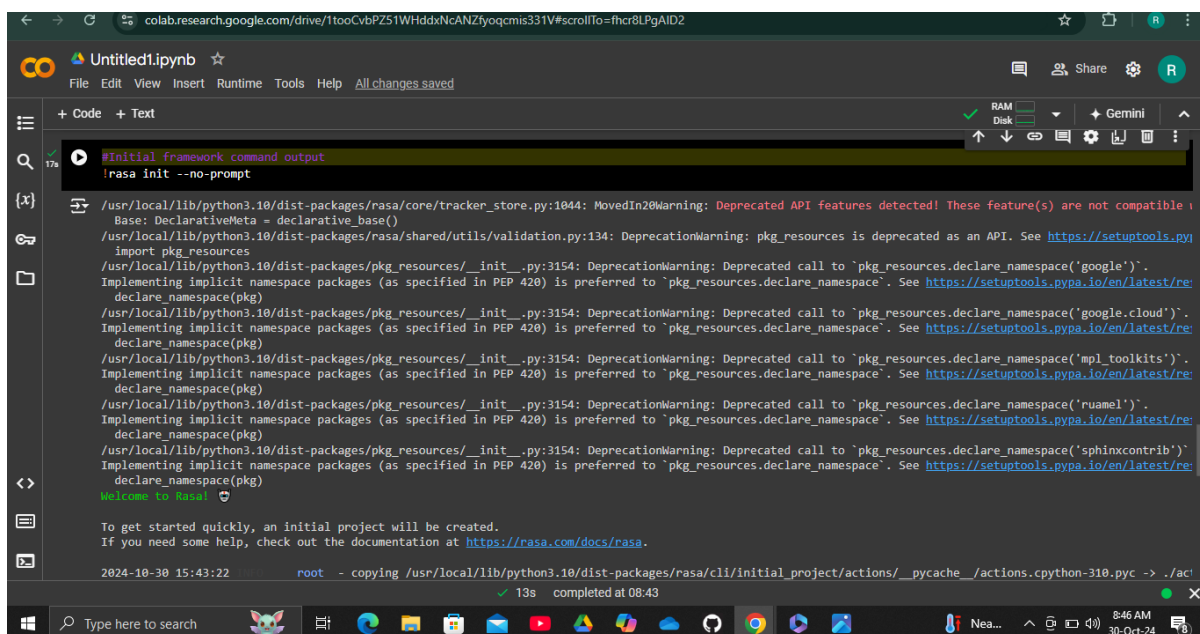
```
#Installation output for core libraries and frameworks
!pip install chatterbot chatterbot_corpus
!pip install flask requests

Collecting chatterbot
  Using cached ChatterBot-1.0.5-py2.py3-none-any.whl.metadata (8.1 kB)
Collecting chatterbot_corpus
  Using cached chatterbot_corpus-1.2.0-py2.py3-none-any.whl.metadata (5.0 kB)
Collecting mathparse<0.2,>=0.1 (from chatterbot)
  Using cached mathparse-0.1.2-py3-none-any.whl.metadata (776 bytes)
Requirement already satisfied: nltk<4.0,>=3.2 in /usr/local/lib/python3.10/dist-packages (from chatterbot) (3.8.1)
Collecting pint>=0.8.1 (from chatterbot)
  Using cached Pint-0.24.3-py3-none-any.whl.metadata (8.5 kB)
Collecting pymongo<4.0,>=3.3 (from chatterbot)
  Using cached pymongo-3.13.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (9.7 kB)
Collecting python-dateutil<2.8,>=2.7 (from chatterbot)
  Using cached python_dateutil-2.7.5-py2.py3-none-any.whl.metadata (7.5 kB)
Collecting pyyaml<5.2,>=5.1 (from chatterbot)
  Using cached PyYAML-5.1.2.tar.gz (265 kB)
error: subprocess-exited-with-error

  x python setup.py egg_info did not run successfully.
  exit code: 1
  See above for output.

note: This error originates from a subprocess, and is likely not a problem with pip.
Preparing metadata (setup.py) ... error
error: metadata-generation-failed
```

Initial framework command output



```
#Initial framework command output
!rasa init --no-prompt

/usr/local/lib/python3.10/dist-packages/rasa/core/tracker_store.py:1044: MovedIn20Warning: Deprecated API features detected! These feature(s) are not compatible with SQLAlchemy 2.0. To prevent disruptions, we have disabled deprecated features. Please see the release notes for details.
Base: DeclarativeMeta = declarative_base()
/usr/local/lib/python3.10/dist-packages/rasa/shared/utils/validation.py:134: DeprecationWarning: pkg_resources is deprecated as an API. See https://setuptools.py
import pkg_resources
/usr/local/lib/python3.10/dist-packages/pkg_resources/_init_.py:3154: DeprecationWarning: Deprecated call to 'pkg_resources.declare_namespace('google')'.
Implementing implicit namespace packages (as specified in PEP 420) is preferred to 'pkg_resources.declare_namespace'. See https://setuptools.py
declare_namespace(pkg)
/usr/local/lib/python3.10/dist-packages/pkg_resources/_init_.py:3154: DeprecationWarning: Deprecated call to 'pkg_resources.declare_namespace('google.cloud')'.
Implementing implicit namespace packages (as specified in PEP 420) is preferred to 'pkg_resources.declare_namespace'. See https://setuptools.py
declare_namespace(pkg)
/usr/local/lib/python3.10/dist-packages/pkg_resources/_init_.py:3154: DeprecationWarning: Deprecated call to 'pkg_resources.declare_namespace('ruamel')'.
Implementing implicit namespace packages (as specified in PEP 420) is preferred to 'pkg_resources.declare_namespace'. See https://setuptools.py
declare_namespace(pkg)
/usr/local/lib/python3.10/dist-packages/pkg_resources/_init_.py:3154: DeprecationWarning: Deprecated call to 'pkg_resources.declare_namespace('sphinxcontrib')'.
Implementing implicit namespace packages (as specified in PEP 420) is preferred to 'pkg_resources.declare_namespace'. See https://setuptools.py
declare_namespace(pkg)

Welcome to Rasa! 🤖

To get started quickly, an initial project will be created.
If you need some help, check out the documentation at https://rasa.com/docs/rasa.

2024-10-30 15:43:22 root - copying /usr/local/lib/python3.10/dist-packages/rasa/cli/initial_project/actions/_pycache_/actions.cpython-310.pyc -> ./ac
```

4. Issues Encountered During Setup and Solutions

- **Issue:** Incompatibility with Python version for certain libraries.
 - ❖ **Solution:** Downgrade or upgrade Python as necessary, ensuring compatibility with the chosen chatbot framework.

- **Issue:** SSL Certificate error when installing packages.
 - ❖ **Solution:** Run pip with --trusted-host or upgrade pip to resolve certificate errors.
- **Issue:** Node and npm permission issues on Linux.
 - ❖ **Solution:** Use sudo npm install -g [package] or adjust user permissions.

5.A simple "Hello World" chatbot script using your chosen framework.

Here's the entire code setup for a "Hello World" chatbot using **RASA**. This includes creating a basic greeting functionality that recognizes when a user greets the bot and responds appropriately.

```
# Simple "Hello World" bot script using Chatterbot

from chatterbot import ChatBot
from chatterbot.trainers import ChatterBotCorpusTrainer

# Initialize chatbot
chatbot = ChatBot("HelloBot")

# Trainer for our chatbot
trainer = ChatterBotCorpusTrainer(chatbot)
```

```
# Train the chatbot on English corpus
```

```
trainer.train("chatterbot.corpus.english")
```

```
# Simple conversation
```

```
response = chatbot.get_response("Hello, how are you?")
```

```
print("Bot:", response)
```