Guani: Chatbot for booking a Hotel Room | Development Phase

Course: Project AI Use case
Tutor: Tim Schlippe

Adonai Vera

Msc student IUBH
Mechatronic Engineer
Chief Operations Officer
+57 318 638 4034
Adonai.vera@switchai.co

www.switchai.co
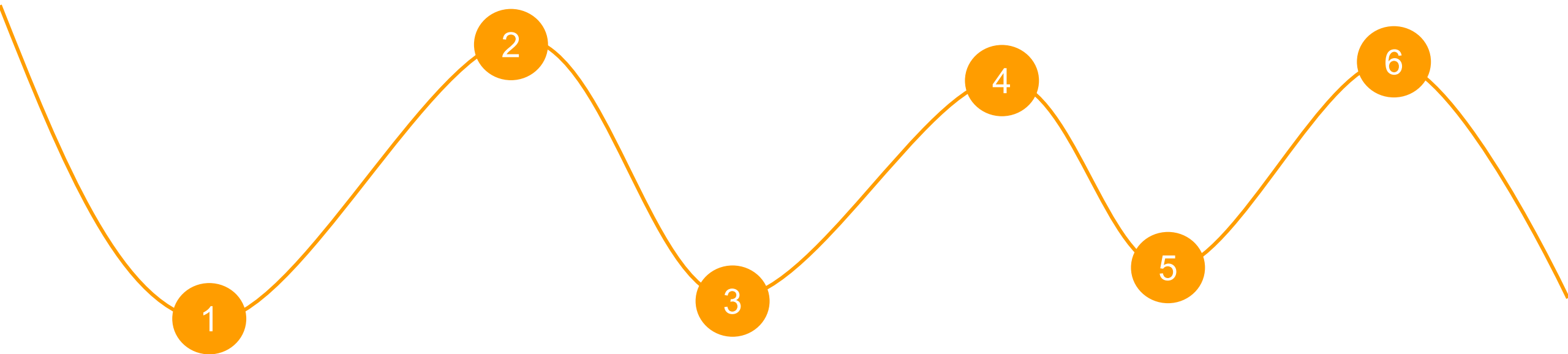
1

Topics

DIAGRAM AND ARCHITECTURE

DATABASE

OPTIMIZATION

2

6

4

1

3

5

FRAMEWORKS AND SETUP

CODE

TRAIN (METRICS)

Frameworks and setup

Configure all the process

Create a virtualenv with Python 3.8

Activate the virtual env

Install Rasa 2.6.2 with pip

init a project to have the architecture ready

Install Rasa X to get more insights

Init Rasa X

Use the requirements file to get everything done.
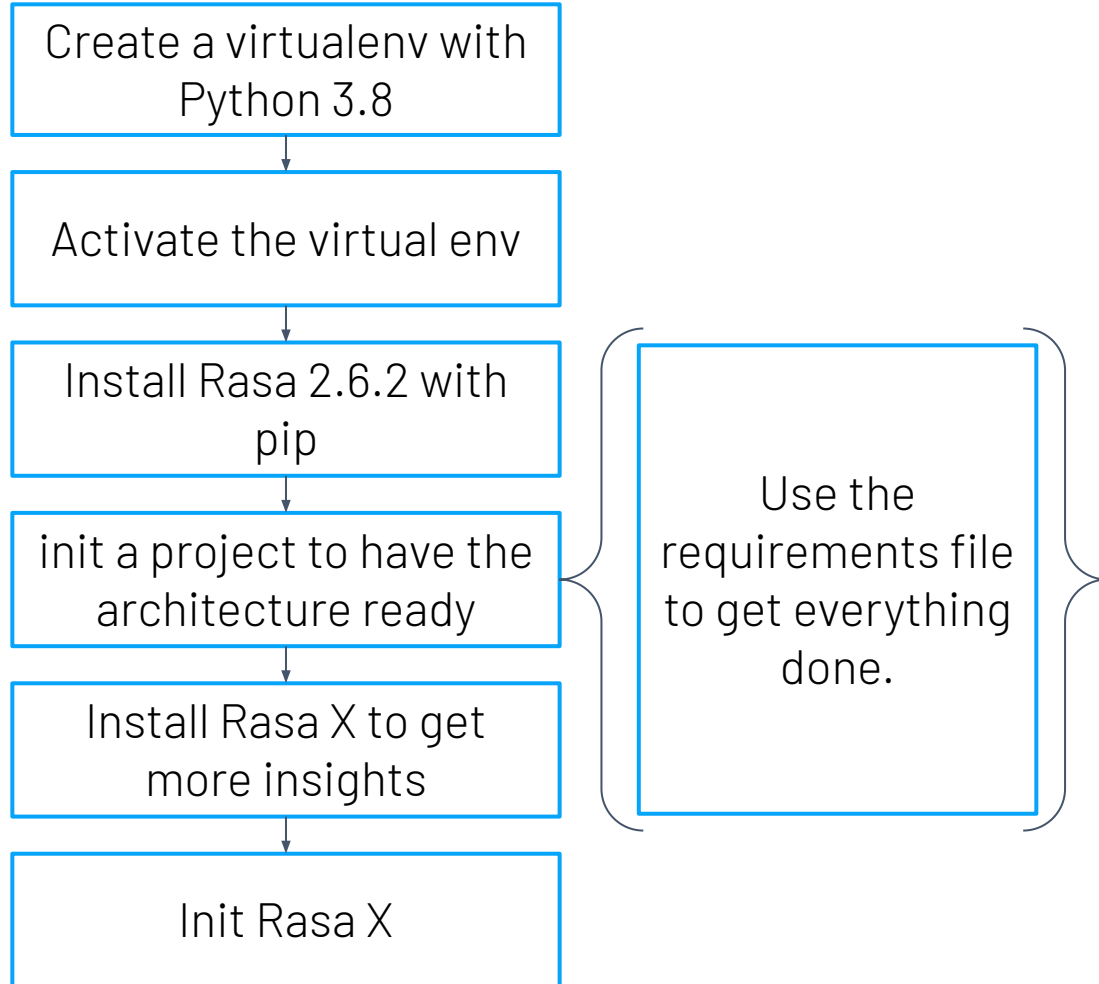
```
virtualenv env -p python3.8
source env/bin/activate
pip install -r requirements.txt
rasa x
```

```
virtualenv env -p python3.8
source env/bin/activate
pip install rasa==2.6.2
pip3 install rasa-x==0.40.0 --extra-index-url https://pypi.rasa.com/simple
pip install sanic-jwt==1.6.0
rasa x
```

Environment
Linux
Ubuntu 20.04
Python 3.8
Rasa

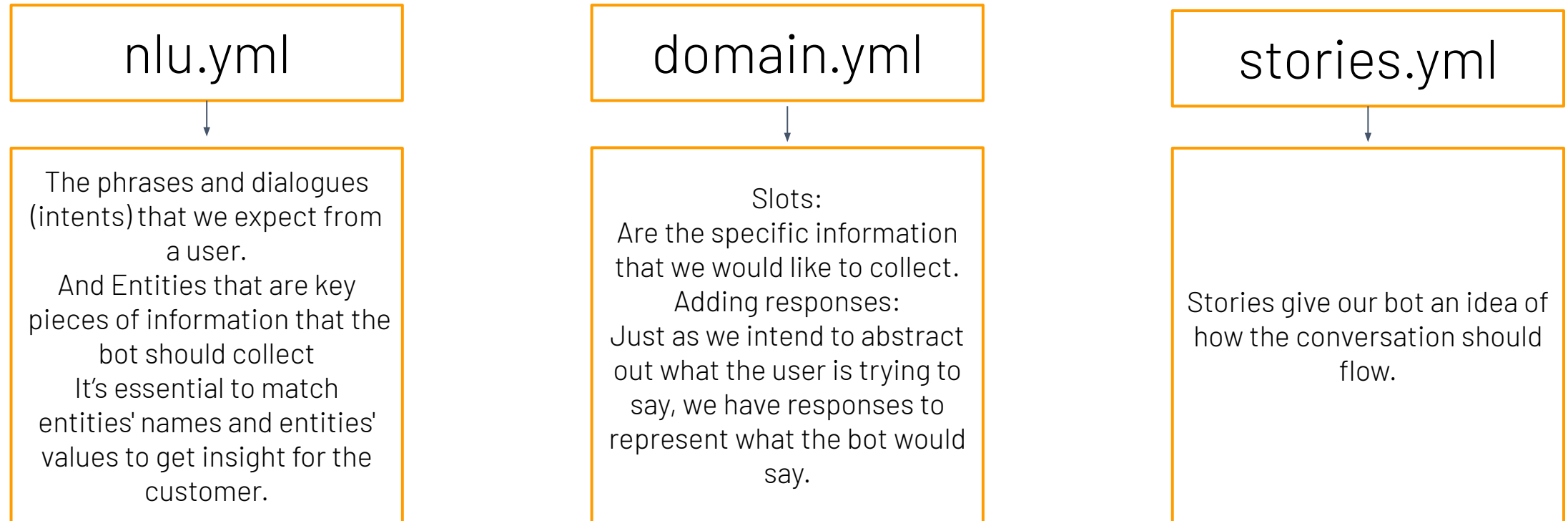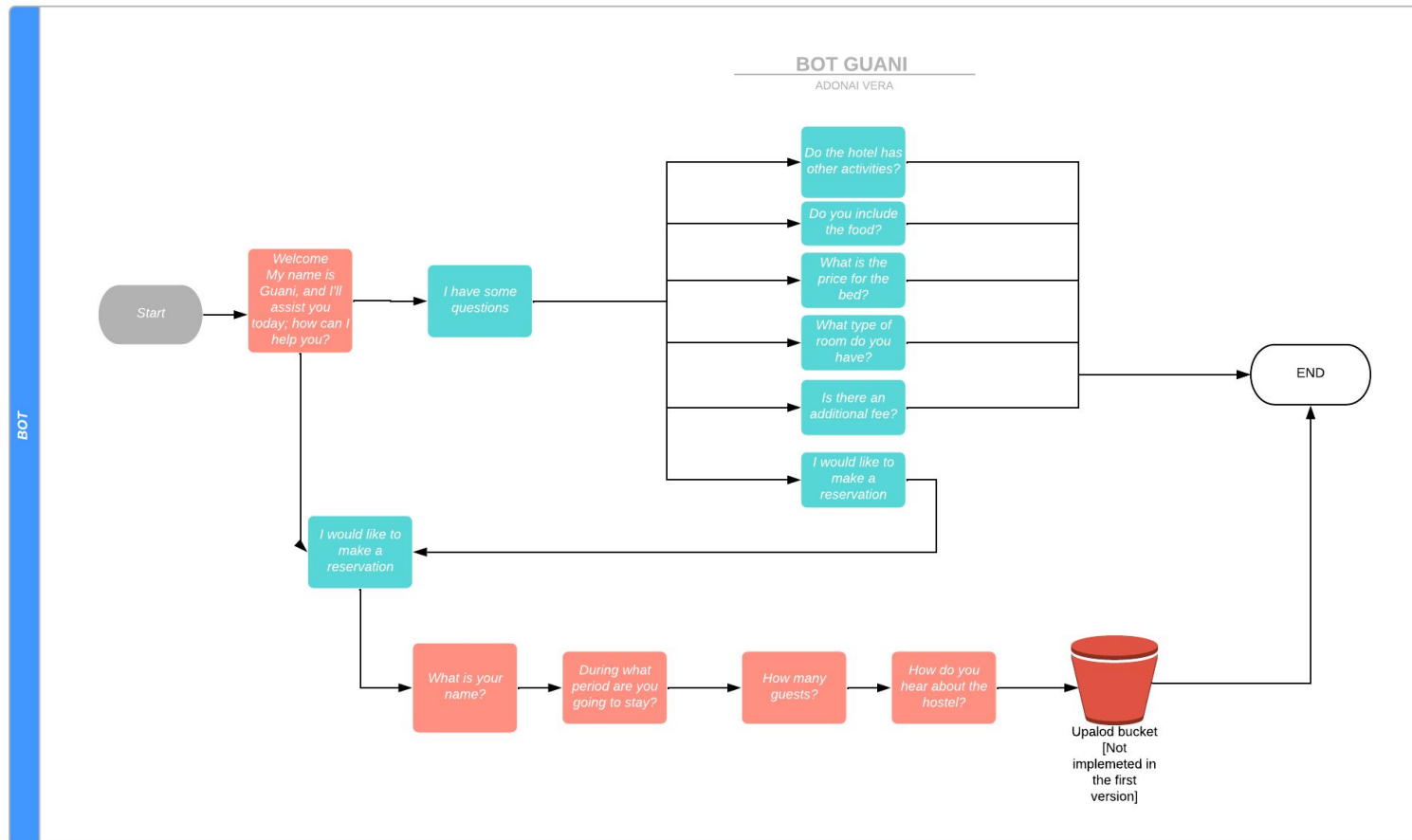# Diagram and Architecture 🌏

"How the chatbot works internally

nlu.yml	domain.yml	stories.yml
The phrases and dialogues (intents) that we expect from a user. And Entities that are key pieces of information that the bot should collect It's essential to match entities' names and entities' values to get insight for the customer.	Slots: Are the specific information that we would like to collect. Adding responses: Just as we intend to abstract out what the user is trying to say, we have responses to represent what the bot would say.	Stories give our bot an idea of how the conversation should flow.

Diagram and Architecture 📷

I applied the same diagram that I proposed in the conception phase but now using the tools that Rasa provided me:
nlu.yml
domain.yml
stories.yml

Database 📷

"How the chatbot works internally

In the database, there is a list with more than 140 sentences and with the respective intents. All the database will be in the finalization phase.
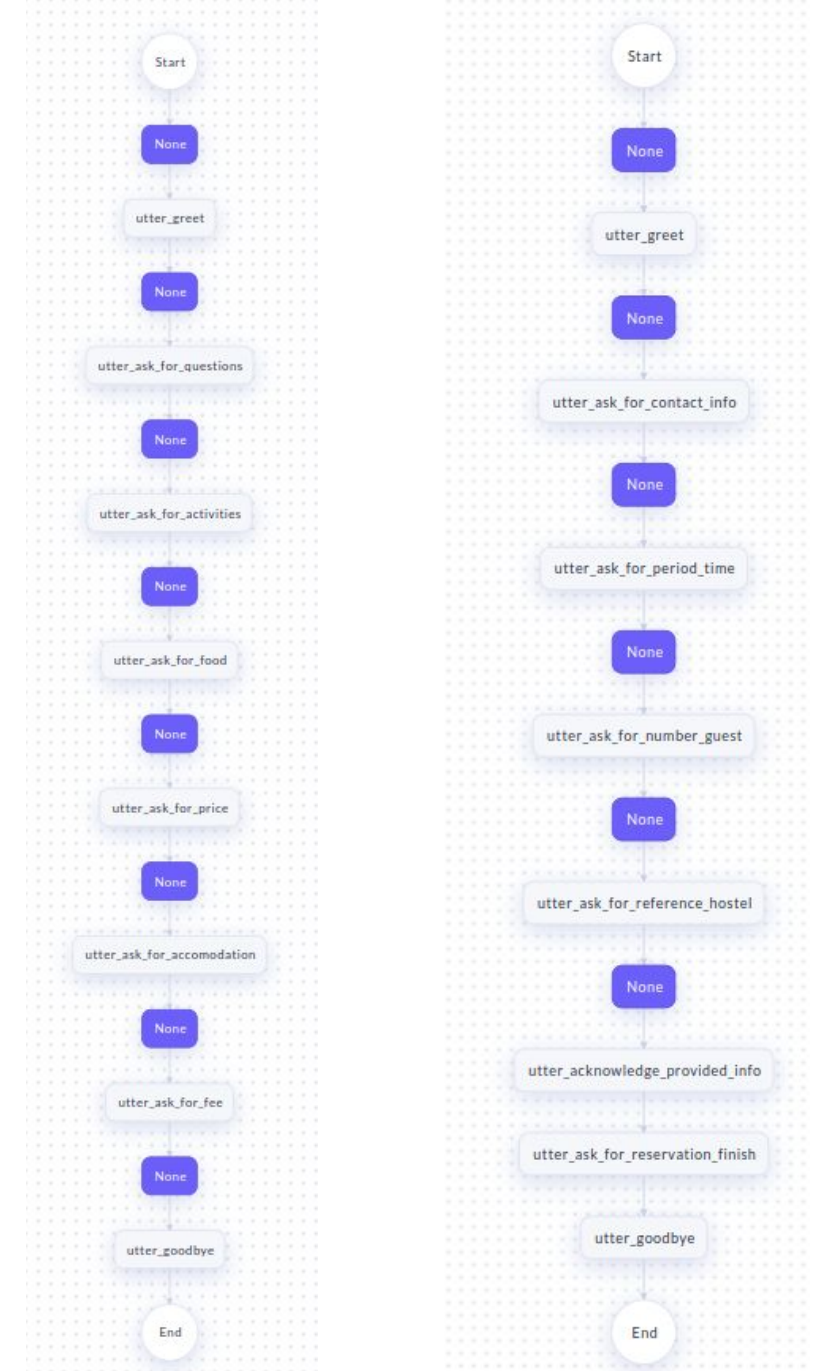
Database 📷

"How the chatbot works internally

Responses		Response variants for utter_ask_for_activities	
	Search responses...		
utter_acknowledge_provided_info	1	Yes {name} we have salsa night on thursday and karaoke on Friday. Besides, you can enjoy the bar open from 11 pm to 3 am	
utter_ask_for_accomodation	1		
utter_ask_for_activities	1		
utter_ask_for_contact_info	1		
utter_ask_for_fee	1		
utter_ask_for_food	1		
utter_ask_for_number_guest	1		

We created the response to each topic that the user could ask.
Besides, we add two stories: Make a reservation | Solve some questions.
To show the flow that the user can take.
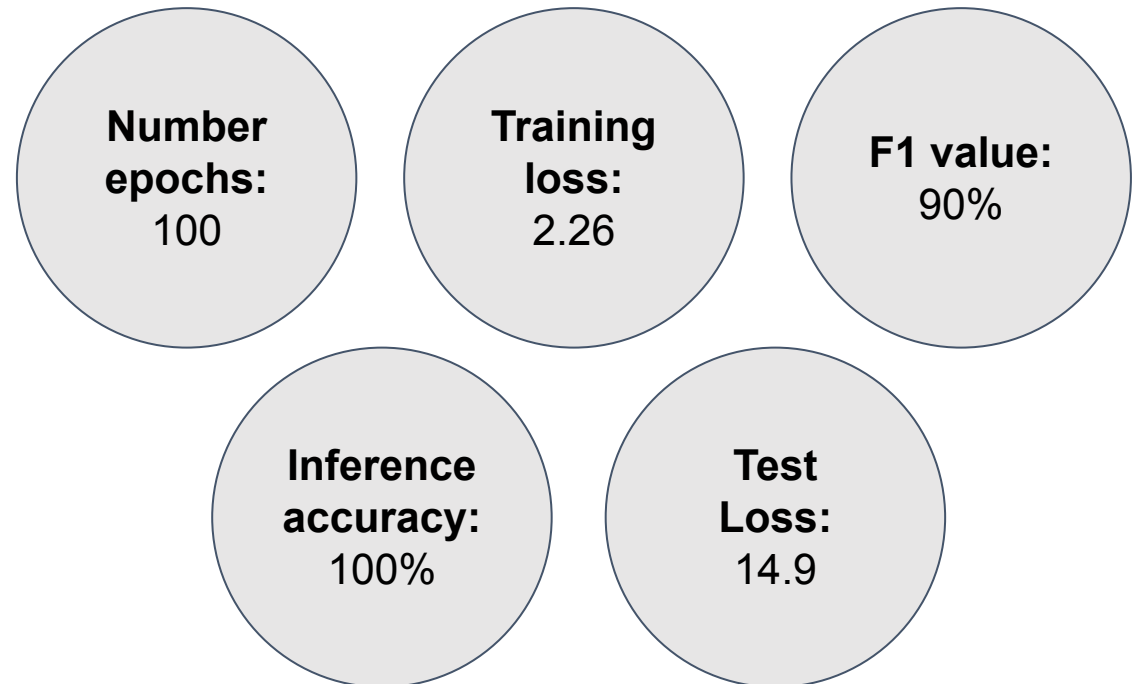We are adding more info about the optimization process.

Flow 1:
Start
None
utter_greet
None
utter_ask_for_questions
None
utter_ask_for_activities
None
utter_ask_for_food
None
utter_ask_for_price
None
utter_ask_for_accomodation
None
utter_ask_for_fee
None
utter_goodbye
End

Flow 2:
Start
None
utter_greet
None
utter_ask_for_contact_info
None
utter_ask_for_period_time
None
utter_ask_for_number_guest
None
utter_ask_for_reference_hostel
None
utter_acknowledge_provided_info
utter_ask_for_reservation_finish
utter_goodbye
End

iu

Train models

Train the model from cmd or from rasa x

Validate if the data to train are okey

↓

Train the model

↓
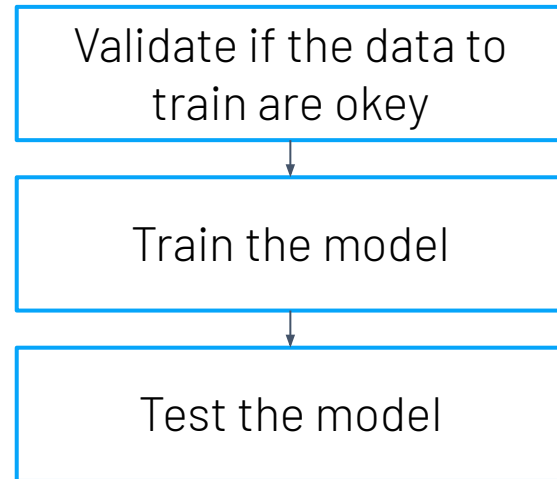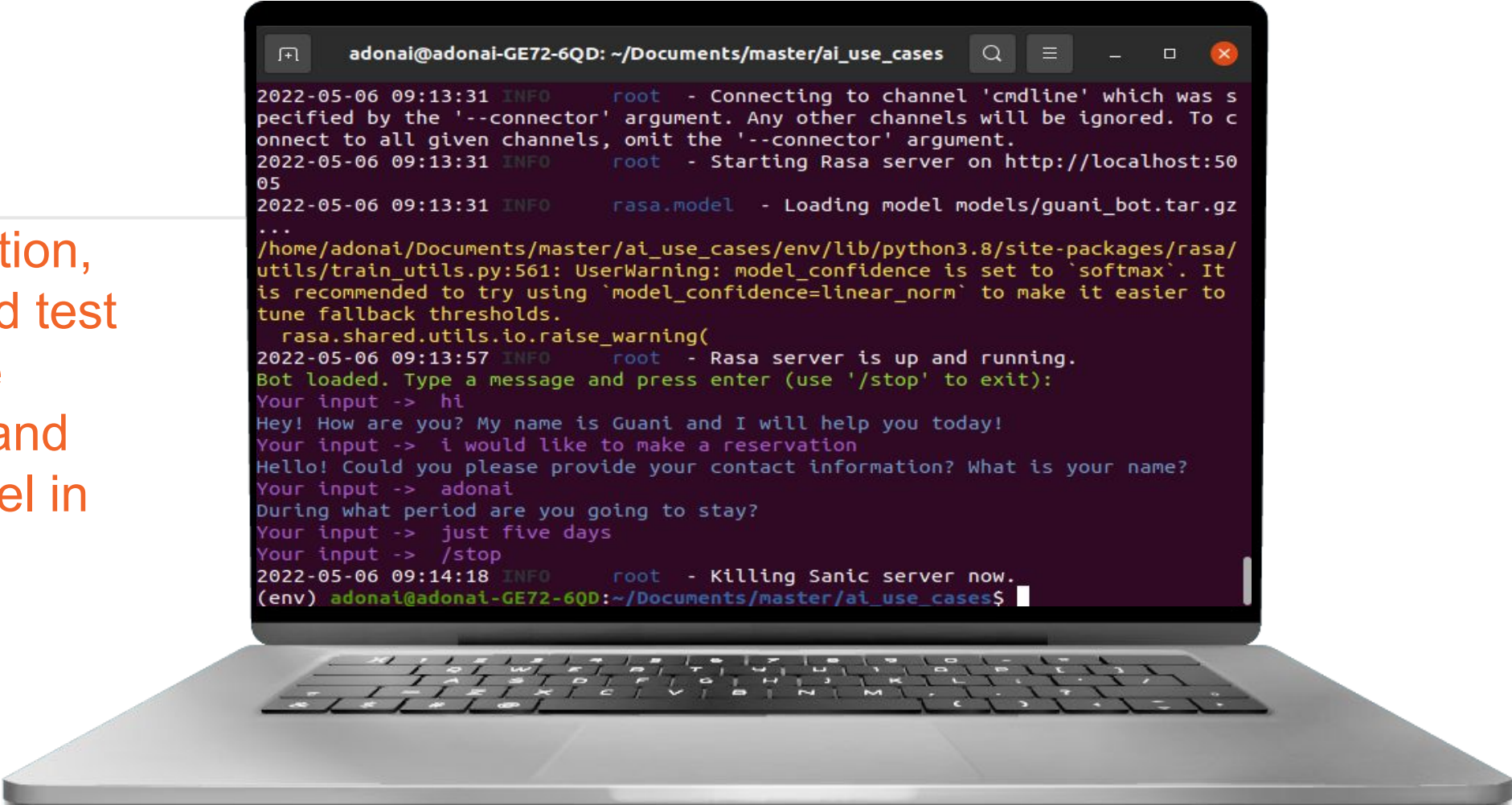
Test the model

```
rasa data validate
rasa train --fixed-model-name guani_bot
rasa shell
```

Metrics

Number epochs:
100

Training loss:
2.26

F1 value:
90%

Inference accuracy:
100%

Test Loss:
14.9

Optimization

We test the model and re train the algorithm with more entities

For the optimization,
we use rasa x and test
with multiple
conversations and
re-train the model in
the service.

Results and next step

We test the model and re train the algorithm with more entities

Next steps
Keep testing the algorithm with new users and improve the database.
Test Guani here:
[Guani test here](#)

We deploy the chatbot locally and generate a URL that the user can test, and we created a channel with NGROK to try the chatbot for three days; the steps will be in the finalization phase.

THANK YOU
Adonai Vera
www.switchai.co