Sprint 11

Flask Setup

```
from flask import Flask, request
from naoqi import ALProxy

app = Flask(__name__)

# Set up connection to NAO robot
NAO_IP = "ip_address"
NAO_PORT = 9559  # Default port for NAO
tts = ALProxy("ALTextToSpeech", NAO_IP, NAO_PORT)
motion = ALProxy("ALMotion", NAO_IP, NAO_PORT)
robot_on = False  # Flag to track NAO's power status

# Endpoint for speaking
@app.route('/speak', methods=['POST'])
def speak():
    content = request.json
    message = content['message']
    tts.say(message)
    return "Spoken: " + message

# Endpoint for walking commands
@app.route('/walk', methods=['POST'])
def walk():
    content = request.json
    direction = content['direction']
    distance = content['distance']
    if direction == 'forward':
        motion.walkForward(distance)
    elif direction == 'backward':
        motion.walkBackward(distance)
    elif direction == 'left':
        motion.walkTo(distance, 0)
    elif direction == 'right':
        motion.walkTo(-distance, 0)
    return "Walking command executed"

# Endpoint for turning the robot on/off
@app.route('/power', methods=['POST'])
def power():
    global robot_on
    content = request.json
    action = content['action']
    if action == 'on':
        robot_on = True
    elif action == 'off':
        robot_on = False
    return "Robot is now " + action

if __name__ == '__main__':
    app.run(debug=True)
```

Nao Scripting

```
import qi
import sys
import time

class NaoController:
    def __init__(self, nao_ip, nao_port=9559):
        self.nao_ip = nao_ip
        self.nao_port = nao_port
        self.session = None
        self.motion = None
        self.audio = None
        self.memory = None
        self.robot_on = False
    def connect_to_robot(self):
        try:
            self.session = qi.Session()
            self.session.connect("tcp://" + self.nao_ip + ":" +
            self.motion = self.session.service("ALMotion")
            self.audio = self.session.service("ALAudioDevice")
            self.memory = self.session.service("ALMemory")
            self.robot_on = True
            print("Connected to NAO robot.")
        except Exception as e:
            print("Error connecting to NAO robot:", e)

    def speak(self, text):
        if self.robot_on:
            self.audio.setOutputVolume(80)
            self.motion.wakeUp()
            self.motion.setAngles("HeadPitch", 0.0, 0.1)
            self.session.service("ALTextToSpeech").say(text)

    def walk(self, direction, distance):
        if self.robot_on:
            pass

    def power(self, action):
        if action == "on":
            self.robot_on = True
            self.motion.wakeUp()
        elif action == "off":
            self.robot_on = False
            self.motion.rest()

if __name__ == "__main__":
    nao_ip = "nao_robot_ip_address"
    controller = NaoController(nao_ip)
    # Connect to NAO robot
    controller.connect_to_robot()

    # Example usage
    controller.speak("Hello, I am NAO robot!")
    time.sleep(2)
    controller.walk("forward", 0.5)
    time.sleep(2)
    controller.power("off")
```