



UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

Faculty of Engineering, Built Environment and
Information Technology

COS 301

NEURAL NETWORKS

ASSIGNMENT 3: MINI PROJECT DEMO 1

Rudo Janse van Rensburg: u17296928

Pavlo Adrianatos: u17173389

Yuval Lango: u18174142

Dylan Krajnc: u17029318

Aaron Facoline: u13283881

Brad Zietsman: u15228194

Nigel Mpofu: u15015557

March 13, 2020

Contents

1	Introduction	1
2	Technologies	1
3	Mini Project: Demo 1	1
3.1	Working Service Port	1
3.2	Git and Git Branching	2
3.3	Mocked Functions and Unit Testing	2
3.4	Project Management Tools	2

1 INTRODUCTION

The following document will explain the technologies we've used and plan to use for each section of the neural network as well as some explanation of the implementation.

2 TECHNOLOGIES

- Debian - Linux Server
- Java
- DL4J
- Django - Python framework for the API
- Clickup
- Slack
- Github
- Discord

3 MINI PROJECT: DEMO 1

3.1 WORKING SERVICE PORT

```
POST /api/training/{soundData}
  soundData: { sampling rate: 0,
    Mouth1:{Frequency : [],
      Time: [],
      dB: []
    },
    Mouth2:{Frequency : [],
      Time: [],
      dB: []
    },
    ...},
  checksum : 8d95f67c3a1605410fc123018dfc84a6
```

What the API should do:

- Check for Bots.
- Check if Domain Name is valid.
- Check that the request is a POST request..
- Compare the hash values to see if they match (if any).
- Send JSON through to the Neural Network.

Environment where the Neural Network will be hosted/trained:

- <https://ai.teamgamma.ga/>

3.2 GIT AND GIT BRANCHING

The integration team provided everyone with a Git Repo. Our team was given a 'neuralnetwork' to work on. The 'neuralnetwork' branch has four folders. Member CVs, this document and class diagrams were pushed here. In the server folder we pushed all the code that will be hosted on the server. This includes the API, (currently) the mocked functions and unit testing.

3.3 MOCKED FUNCTIONS AND UNIT TESTING

The neural network will be integrated as a forward feeding neural network (subject to change) using DeepLearning4J, which is an open-source, deep-learning library. The mocked functions are implemented in Java and give an idea of how the neural network will be built and how the data we receive from the conversion team will be converted and fed into the neural network. Educated guesses were made on the type of output the mocked functions would return and JUnit5 was used to create test cases that we can use later on to finalise that the code we have created is correct. A final function will be added to return a model of the trained neural network that will be integrated into the android application.

3.4 PROJECT MANAGEMENT TOOLS

The integration team set up both a Slack and Clickup page for our teams to work on. Our team used the board feature on Clickup to keep track of which tasks to complete and who would be assigned to each task. Slack was used to keep up to date with other assigned teams on Team-Gamma, but our team mainly used Discord and WhatsApp as we could conference with each other using voice chat and share resources with each other easily.