Honours Thesis II
Progress Report 3

Sean Stappas
260639512

November 1$^{\text{st}}$, 2017

1 Project Title & Description

The project is *Prometheus AI: Phase 2*. The Prometheus AI is a software model of the human brain with the goal of controlling and coordinating multiple robots in different environments. As an Honours Thesis, this project is to be done alone. As such, all sections on group work (list of group members and group work report) have been omitted.

2 Project Supervisor

The project supervisor and advisor is Joseph Vybihal, professor in computer science at McGill University. Contact information can be found on his website: http://www.cs.mcgill.ca/~jvybihal/

3 Meetings

A meeting occurred on October 13$^{\text{th}}$ with Prof. Vybihal, where the Maven and Guice structure I have chosen for the project were discussed, as well as the next steps. The usual Friday meetings for the subsequent two weeks were canceled. However, status reports were provided instead to Prof. Vybihal via the Slack channel.

4 Project Readings

The readings for the past three weeks are the same as for the first progress report. Most of the recent work was in reading and writing code, as will be described in the next section.

5 Recent Progress

The bulk of the work over the past three weeks has been in refactoring the Prometheus code base in various ways. First, the project was converted to a Java Maven project to allow easy addition of various dependencies. These dependencies include Google Guice for code structure, Mockito for testing and Apache Commons Lang for standardized `equals` and `hashCode` implementations.

Once the above dependencies were added, a simple Guice structure was implemented in the project. This meant converting the existing ES and KNN layers to use Guice, as well as creating new skeletons for the NN and META layers. Every layer is now associated with a Java package, which contains within it `api`, `guice` and `internal` packages. The `api` packages contain all public interfaces and classes to be accessible by a user of Prometheus. The `guice` packages contain the public Guice modules. Finally, the `internal` packages

contain all internal code and implementation details for each layer. Guice best practices were followed across the code base, including the use of interfaces, encapsulation and factories.

Once this basic structure was set up, work began on creating meaningful unit tests for the ES layer. This meant refactoring the current code if necessary to make it testable. Indeed, a great deal of work was spent making the code more modular and compliant to unit tests. Now, the ES is almost fully covered by tests.

Some more administrative tasks were also completed. Michael Ding, a new student interested in working on the META layer, was added to the Slack channel. A discussion was also had with Mohammad Owais Kerney, another new student who will likely work on the NN layer. A summary of the work I completed, as well as a copy of my ECSE-498 report, were provided to him as background information on the project. Finally, a clone of the private GitHub repository I use for Prometheus was added to the `integrated` directory of the `jvybihal` code repository, so Prof. Vybihal can more easily access the code I have written.

6   Future Plans

The plan for the next two weeks is to refactor the KNN code in the same way the ES was refactored. This includes making the code more modular, using dependency inversion and creating useful unit tests. Relevant test cases will be created and tested on the KNN and ES layers. These will discussed with Prof. Vybihal to ensure the proper behavior of the system.

Beyond that, a draft of the poster for the final presentation will be created and final plans for the project will be discussed with Prof. Vybihal.