

**National College of Ireland**

**BSC (Honours) in Computing - Full-time - Year 4**

# **Repeat Terminal Based Assessment Assignment – Jan 2023 Completed by 06 January 2023**

**Hannah O’Rourke**

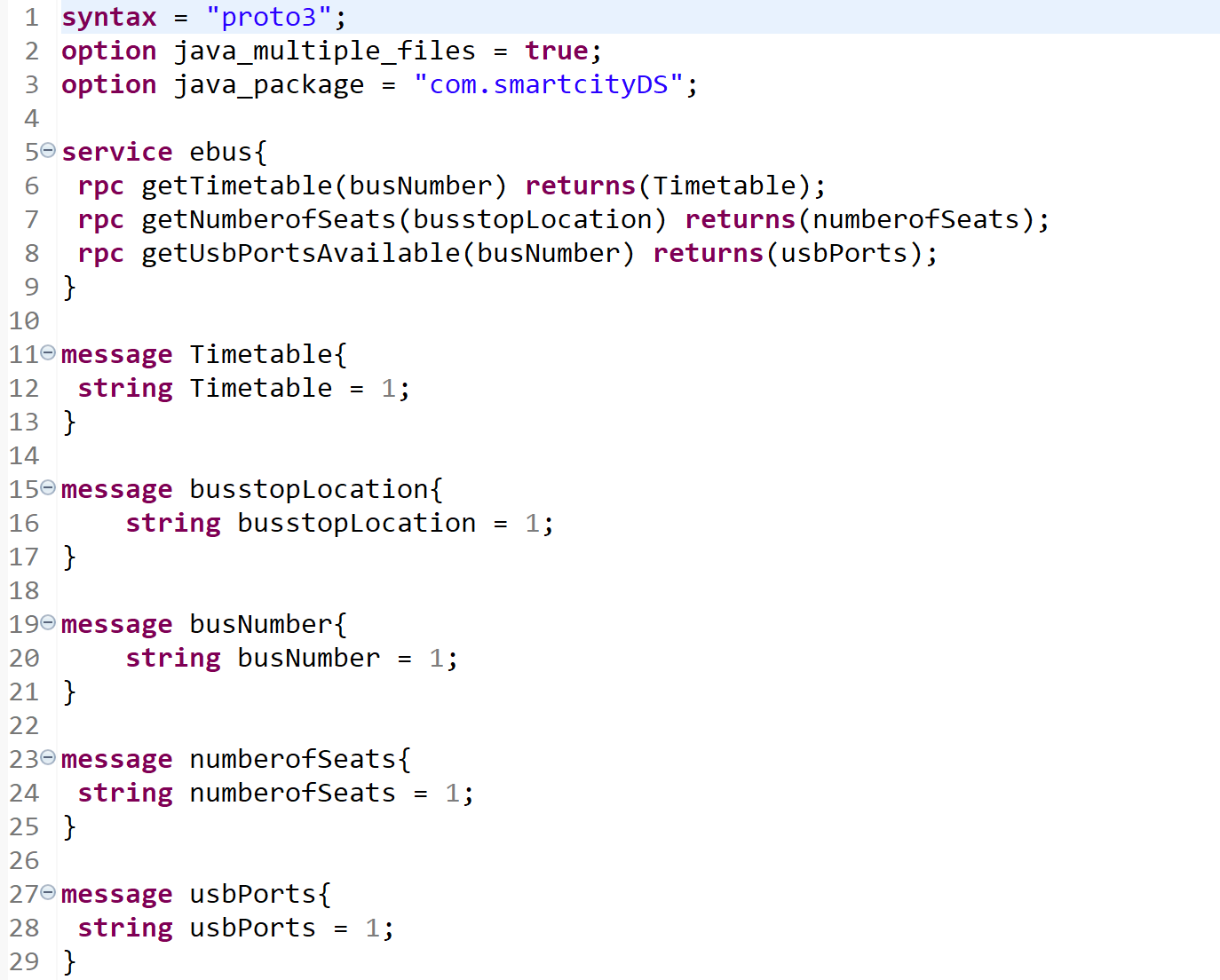
**X18100325**

**Distributed Systems Project Report**

**Dr. Athanasios Staikopoulos**

***Introduction***

This project was developed as part of the Distributed Systems Module at National College of Ireland. The task assigned was to develop a GRPC application in eclipse and implement a GUI to call the services used. The theme of the project is Smart City , which involves eBus , eCar and eScooter. The user can enter their bus stop number and will be returned a selection of electric busses due , the number of complimentary electric car plugin points in the local area and the number of electric scooters available to rent in the local area along with the cost and duration allowed. The smart city application will allow the user to utilize electric transport in their area , resulting in a cleaner , greener environment. Throughout the course of this project report I will outline how I utilized various technologies to achieve success of the project.

***Service Definitions***

The three services that I have utilized in this project are ebus, ecar and escooter .Each of these services have a defined protofile which are utilized throughout the project. Within each of the protofiles I have utilized RPC to retrieve the service and return them via a message I have used the ebus proto file to explain the service definitions utilized; as you can see I have declared three RPC methods , getTimetable , getNumberofSeats and getUsbPorts and returned them using message methods.

***Service Implementation***

The language that I have used to write the services is Java.

Within each of the service classes , I have imported grpc.stub.StreamObserver which receives notifications from an observable stream of messages. It is used by both the client stubs and service implementations for sending or receiving stream messages.

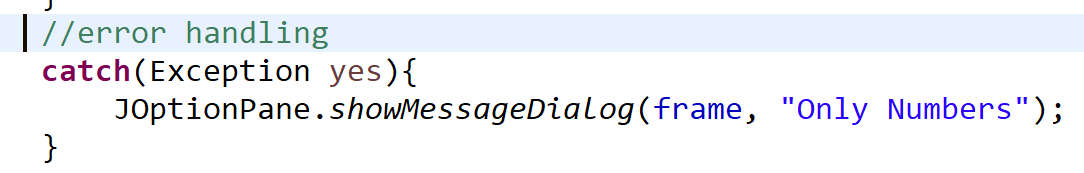
Each service has an implementation base which the service class extends and utilizes.

***Naming Services***

A naming service is a service that allows users and process to add or remove nodes , resulting in the modification of the content of nodes. I have implemented this feature by adding the ability to delete ebus , escooter and ecar nodes.

***Remote Error Handling***

Error handling is crucial in any application , lets say an application does not have error handling , the user enters an invalid input ? the application will crash. I have implemented error handling within my GUI.java code.



***Client - Graphical User Interface***

The GUI in the smart city application was developed using java. Essentially the GUI is the main application that the user will see when running the project. The user will be presented with a content pane with a number of buttons and labels providing access to the three services in the project.

For example; The user will enter a bus stop location in the text field and will be returned the bus timetable for that route.

***Github Repository***

Throughout the development of the project , I ensured that I kept a regular , persistent and concise Github repository. I believe that a Github repository is vital to prevent issues with saving internally , errors in the code and teamwork collaboration. An example of this in my case , I was working on the GUI element of the project and unfortunately I made an error on the code for the GUI.java file , so I simply reverted back to the previous error free commit and was able to carry on with no issues. . It is also crucial to have a Github repository with regular commits to show that the work was consistent , original and well planned. I pushed commits (using Github desktop) to my Github repository after an update was made.

Please see the link below to access my Github repository. [***https://github.com/HannahNCI/DS-TABA***](https://github.com/HannahNCI/DS-TABA)

***Conclusion***

To conclude , the main aim of the project was to design , develop and test a GRPC application using eclipse. I have succeeded by using Java programming , Services files , GUI application , 3 main proto files and sufficient naming services. Each of the services are ebus, ecar and escooter. This project allows the user to run the GUI application , enter their details and therefore avail of the various eservices in their local area.

## Section B – Related Questions (24%)

From this section, please implement or answer **any 2 of the following questions**

**Q1:** Explain the MQTT broker architecture with an example.

**[12 marks]**

**Q2:** Explain and illustrate the sequence of events during a remote procedure call?

**[12 marks]**