Hasnain Mazhar

11479845

Cpts 464 Project 2

April 27, 2018

**Simple Public Transport System**

**Code Location:**

Linux virtual machine

**Positive comment:**

It’s definitely one of the unique project I’ve done during my years as an undergrad at Washington State University. At the beginning, I really struggled to set up the project because I’ve never done a project with multiple publishers and subscribers but thanks to RTI forums I was able to figure out. After finishing the project, I found out that it’s not particularly a hard project as long as you know how to setup and understand the generated code, however, getting the project to work and figuring out bugs took a lot of my time. I believe it was a great learning curve for me and I was surprised that I even got the project to work. Proud of myself!

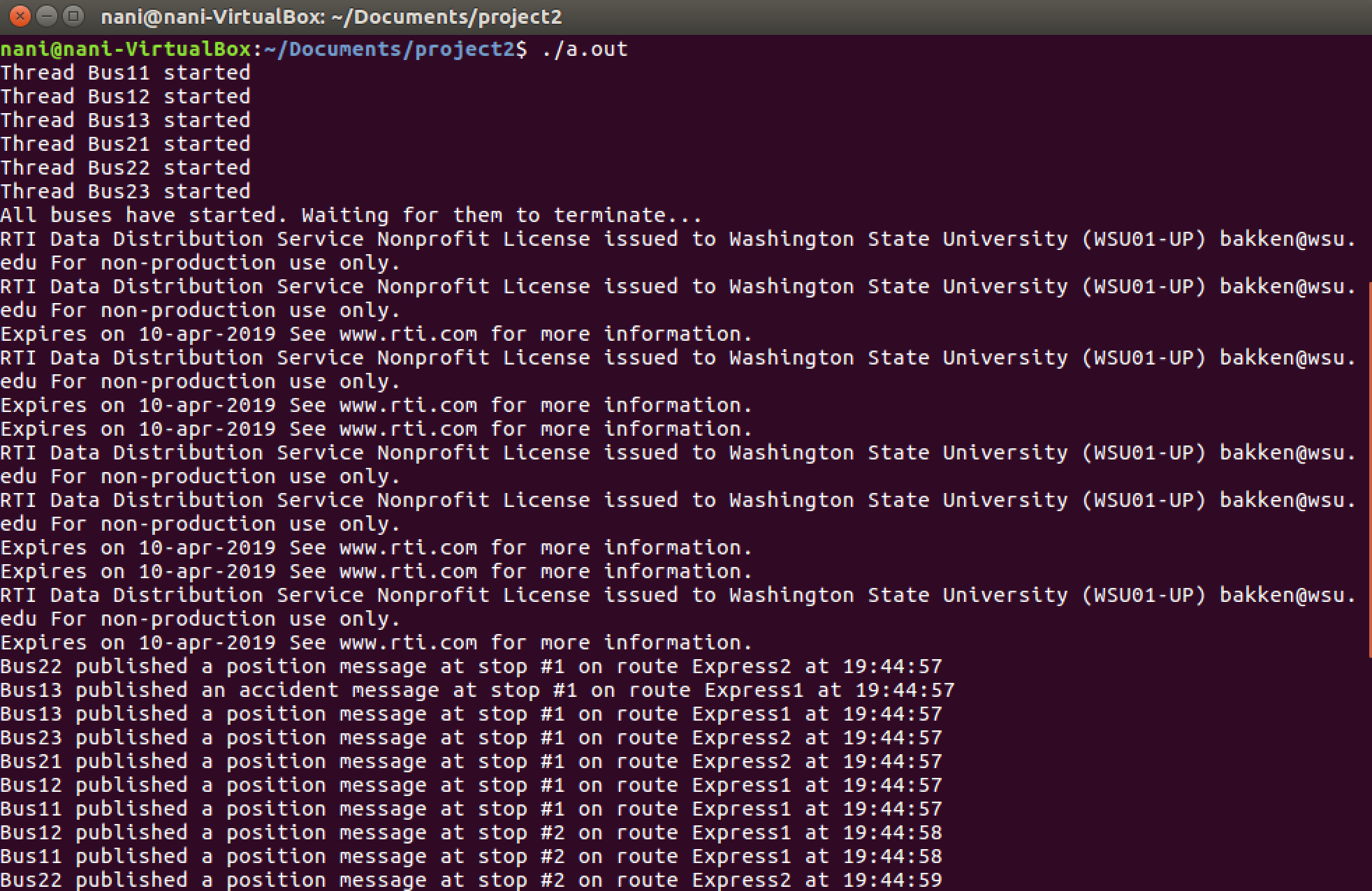
**Suggestions about improving the project:**

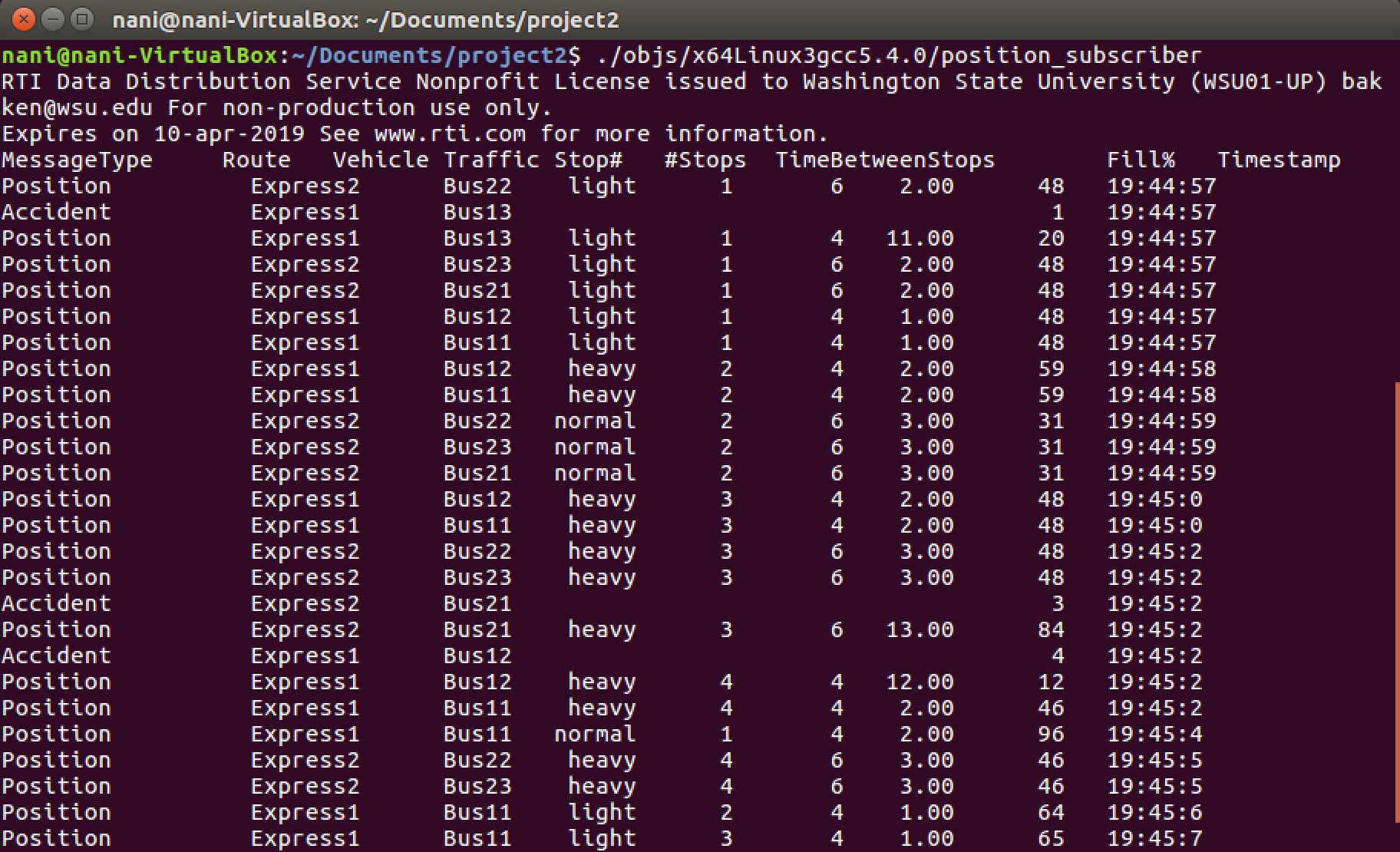
Since I am an undergrad student and had never worked on a distributed application before, it was a little challenging for me to even set up the project. Even though we did project 1 to understand data distributed systems but that project was for single subscriber and publisher. I recommend providing more info on how to set up multi publisher/subscriber application or at least provide a link to understand it. I took me a week to just set up the project.

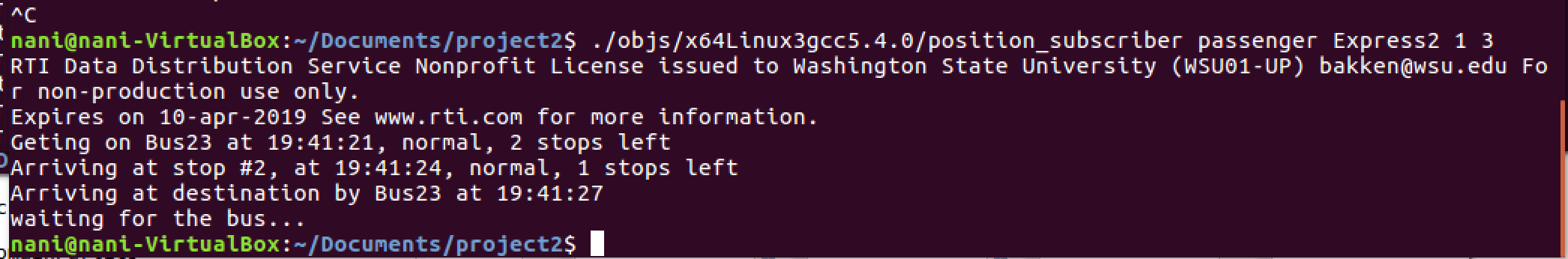
**RTI DDS Pub-Sub Middleware Comments:**

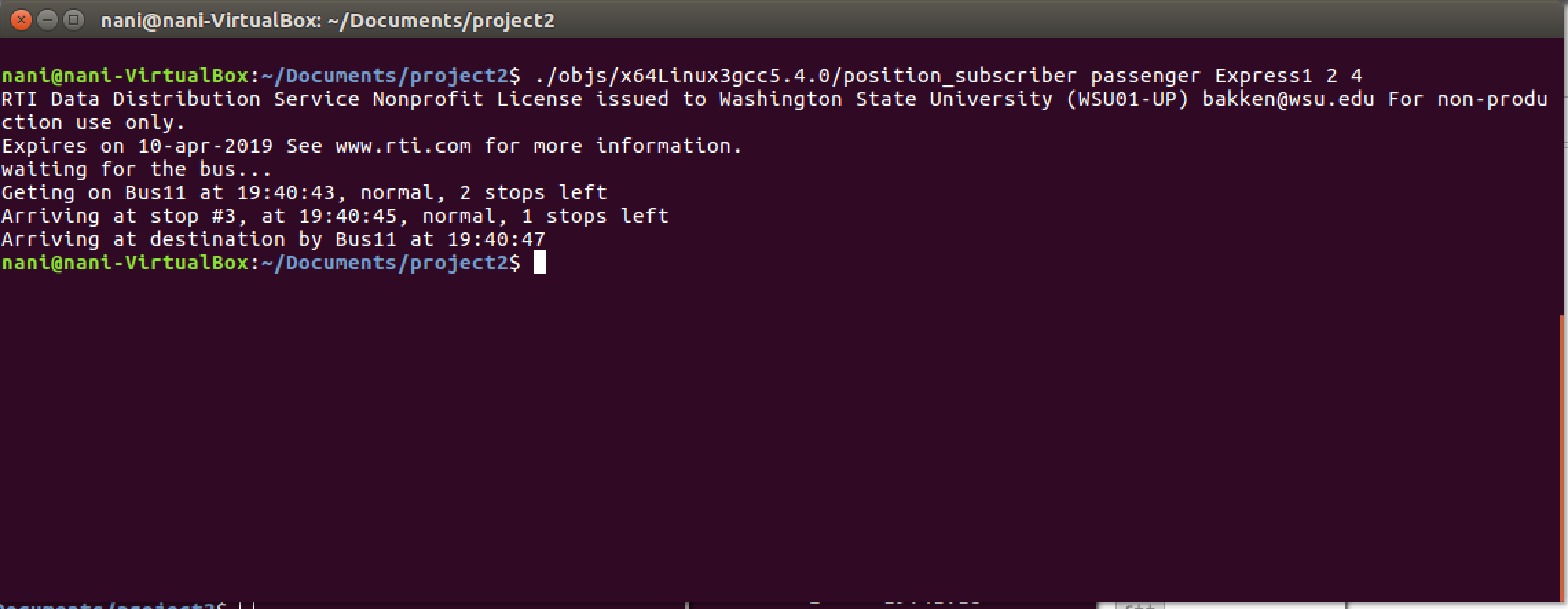
I am very impressed with RTI DDS publisher-subscriber system. It provides a platform for developers to easily setup the environments in no time and move on with the development, however, the code that is being generated is hard to fully understand. For example, in this project I understand position\_publisher/subscriber and or accident\_publisher/subscriber because it was well documented, but I still don’t understand other cxx files like plugin.cxx or support.cxx. Secondly, RTI needs to update their documentation as they take out new versions of RTI because some of the commands that they have suggested on their documentation page doesn’t really work. I started my project on windows platform but RTI didn’t provide sufficient documentation on how to set up multiple publishers and subscribers ton visual studio and I was forced to move to Linux platform. All in all, I think RTI DDS is a great recourse to familiarize with distributed systems. I would definitely recommend anyone who is pursuing the field.

**Run-time screenshots**

****

****

****

****

**NOTE: I ran into an issue towards the end where I wasn’t able to receive data for passengers and I tried everything and spent endless hours to make it work again but it didn’t. I am attaching the screenshots of the working passenger data to show that it was working but then it stopped. I tried going back but I couldn’t undo since I closed the file and reopened it.**

**How to compile:**

1. g++ -w -pthread ./PubLauncher.cxx
2. gmake -f makefile\_position\_x64Linux3gcc5.4.0

**How to run:**

1. for publisher: ./a.out
2. for subscriber operator: ./objs/ x64Linux3gcc5.4.0/position\_subscriber
3. for subscriber passenger: ./objs/ x64Linux3gcc5.4.0/position\_subscriber with **user, route, start, and stop parameters**