**ENGINEERING JOURNAL TEMPLATE**

# Date

27th – 31st Jan

# Tasks

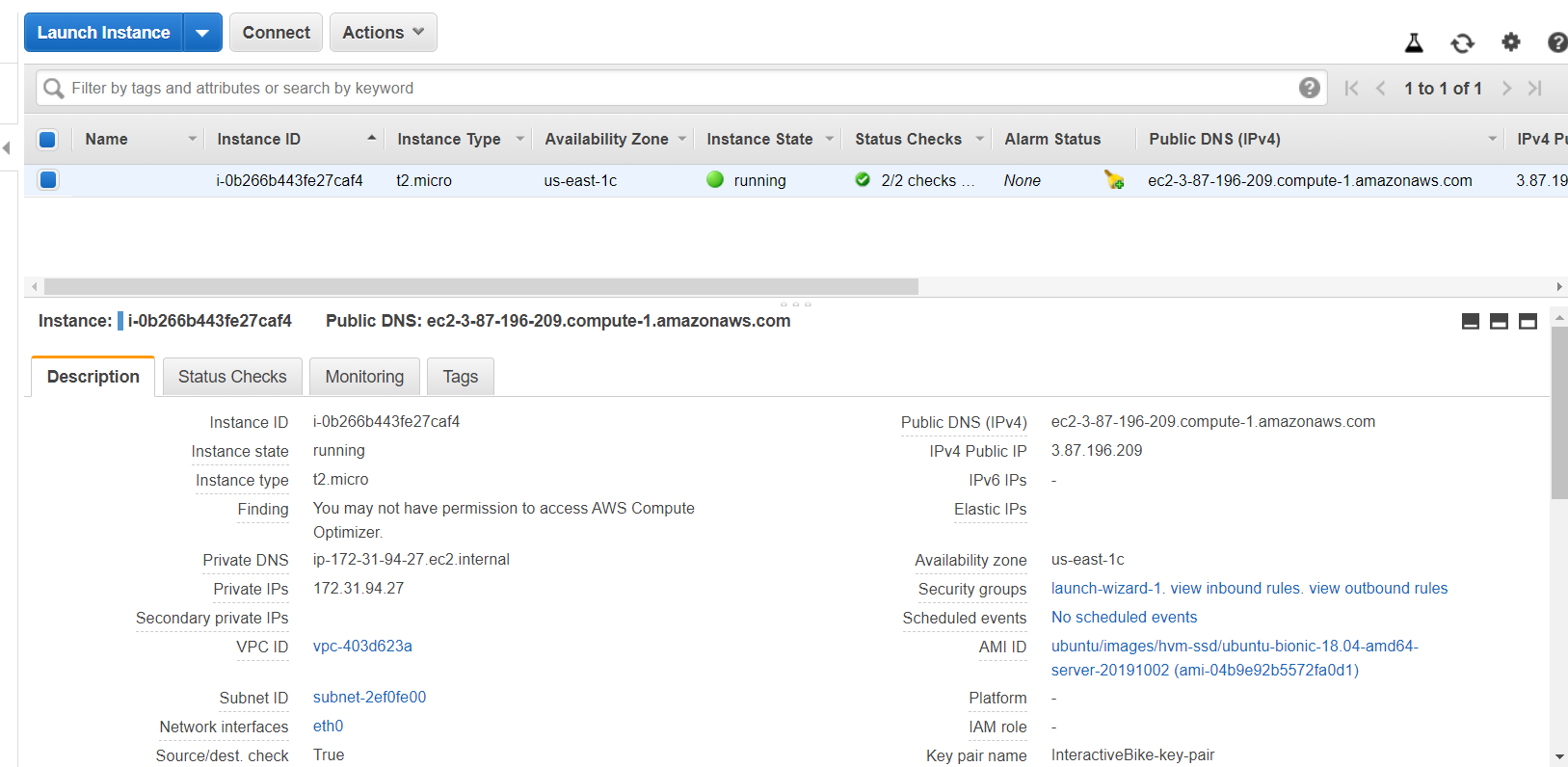
The plan for this semester is to get a fully functioning website working that will store and display the data from the android app. I have done research on this and will create an instance on AWS (Amazon web services), this server will store the data using MongoDB and then I will display it to a website

# Reflection

My refection is that I am happy with the progress from getting the app functional and displaying the data, and feel like if I get this database operational so a cyclist can log on at any time to see there previous rides, then I will be happy that I achieved my goal of completing a functional efficient project.

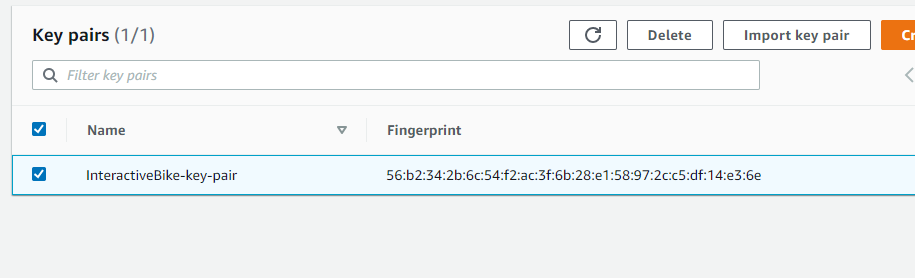
# Solutions

*Software:*



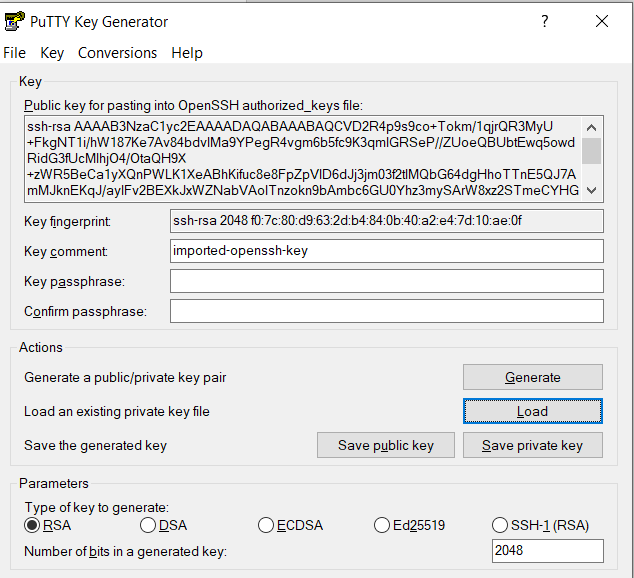
I created a student account with Amazon Web Services. It allows you 100 credits to work with which is more than enough. I then created an instance of type t2 micro, and it is running off ubuntu.

I then created a key pair, this is needed to allow SSH access from my local computer and the pi, to the server. This is a .pem file

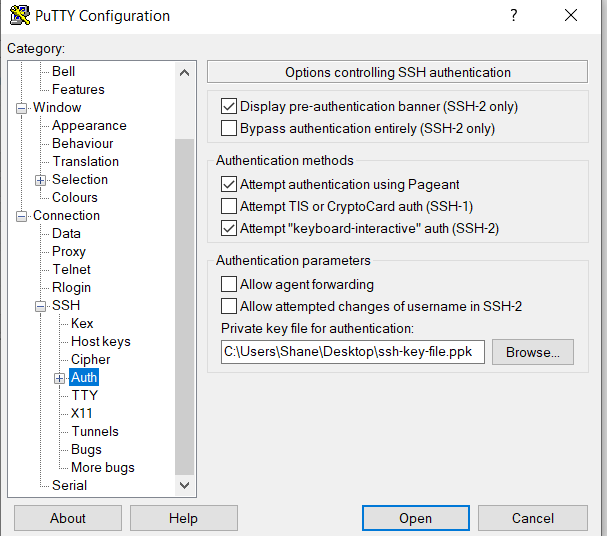


Too build a connection to the Amazon EC2 console I have chosen to use PUTTY, as I have used it for many years in college.

I first used the PUTTY Key Generator to convert my .PEM file that was created above to a .PPK file. PUTTY doesn't natively support the private key format (.pem) generated by Amazon EC2. You must convert your private key into a .ppk file before you can connect to your instance using PuTTY



I then opened PUTTY. My default username will be ubuntu followed by @myipaddress. I also saved this so anymore I can just click ‘Interactive Bike Computer’ which will automatically connect me to the server



I also added the .ppk file in Connections ->SSH->Auth. This successfully let me set up an SSH connection with my EC2 instance.

SSH=Secure Shell

Secure Shell is a cryptographic network protocol for operating network services securely over an unsecured network. Typical applications include remote command-line, login, and remote command execution, but any network service can be secured with SSH.(WIKI)