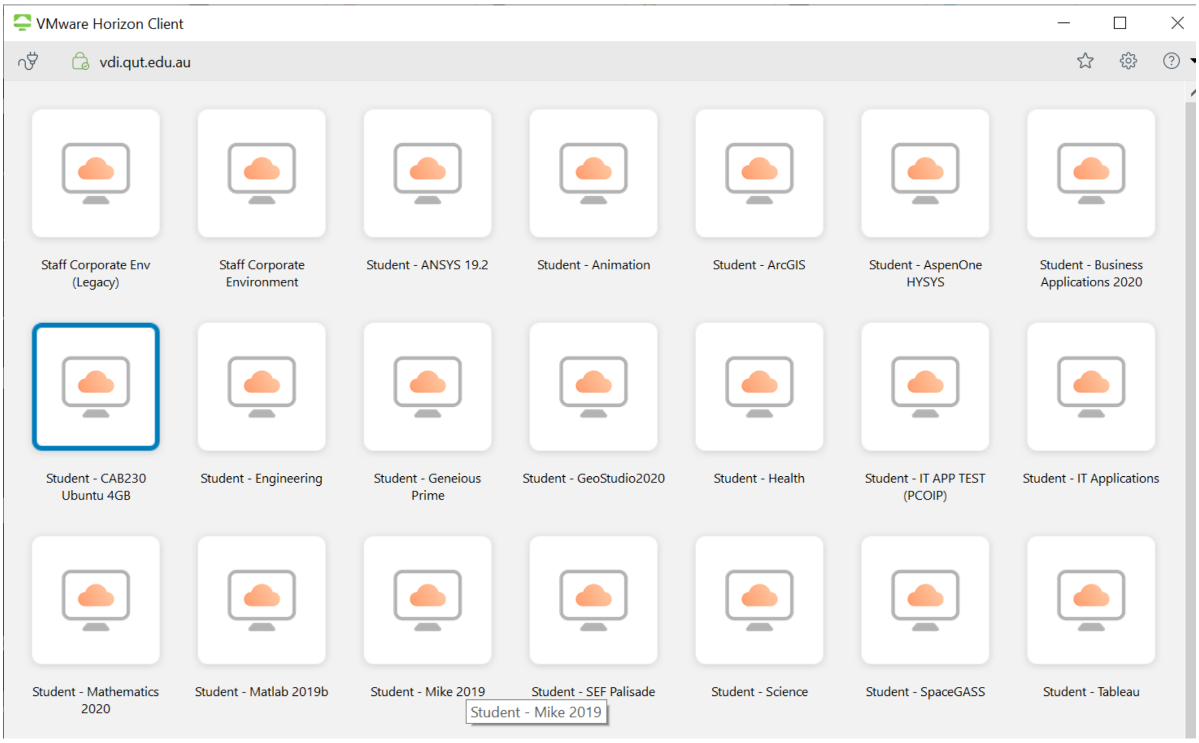
CAB230 VM Deployment Guide

Getting Started

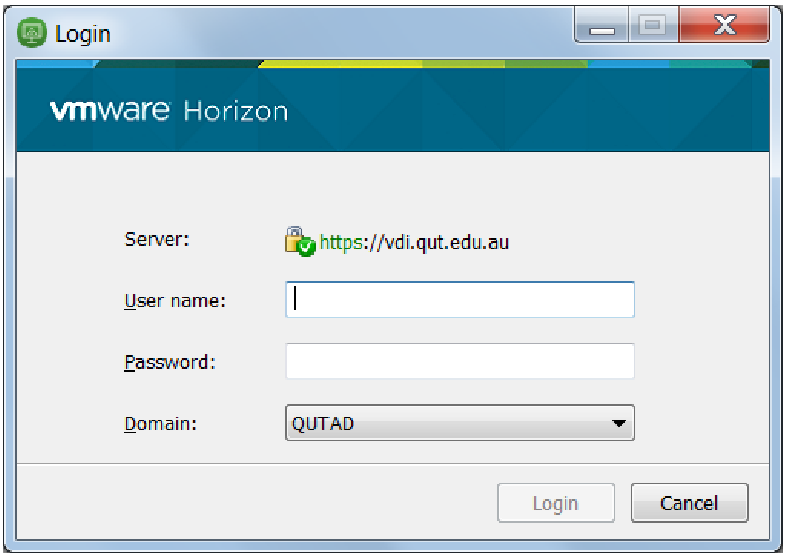
This document tells you how to deploy an application to your QUT Linux VM. As discussed previously, the VM comes pre-installed with node, npm and MySQL. If you are developing on the VM already, the only thing that matters to you is the transition from HTTP to HTTPS, and you may skip a few pages. This document is based on one that we used last year and there is some overlap with the 'Getting started with the Linux VM' guide that we released earlier in the semester. For my convenience and yours I have not tried to optimise this – read the things you need. The examples here are mainly based on deployment of the World Cities Express app, with a few pointers back to the video about MySQL in the Assignment 2 directory, which talks a fair bit about the assignment DB and the tables you need.

Connecting to the VM

This is mainly revision, but using the VMs off campus requires the QUT VPN. You should use the VMWare Horizon client. (Please see the guide in the left hand menu on BB for details on both of these). When the window opens you should see a selection of machine images. The list may vary from the one that I see here, but you should select the one highlighted: `Student – CAB230 Ubuntu 4GB:`



Depending on your authentication status, you may be confronted with a login window like the one below. You should use your usual QUT login details.



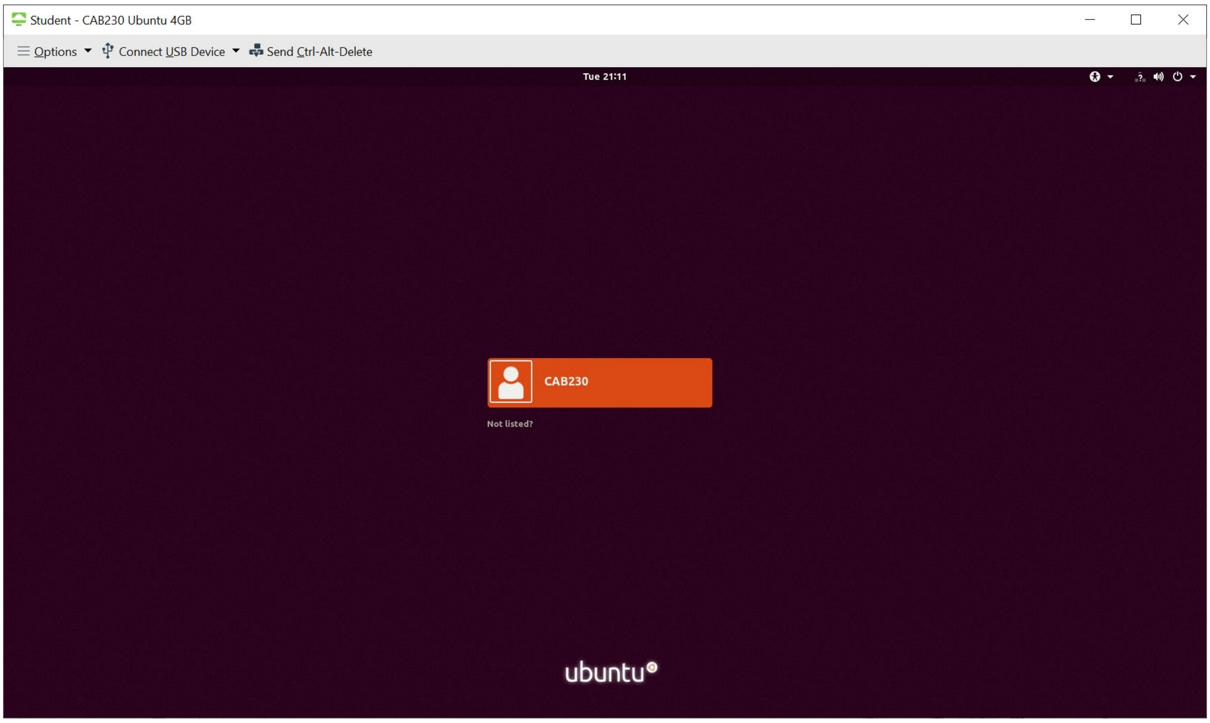
Server: https://vdi.qut.edu.au

User name:

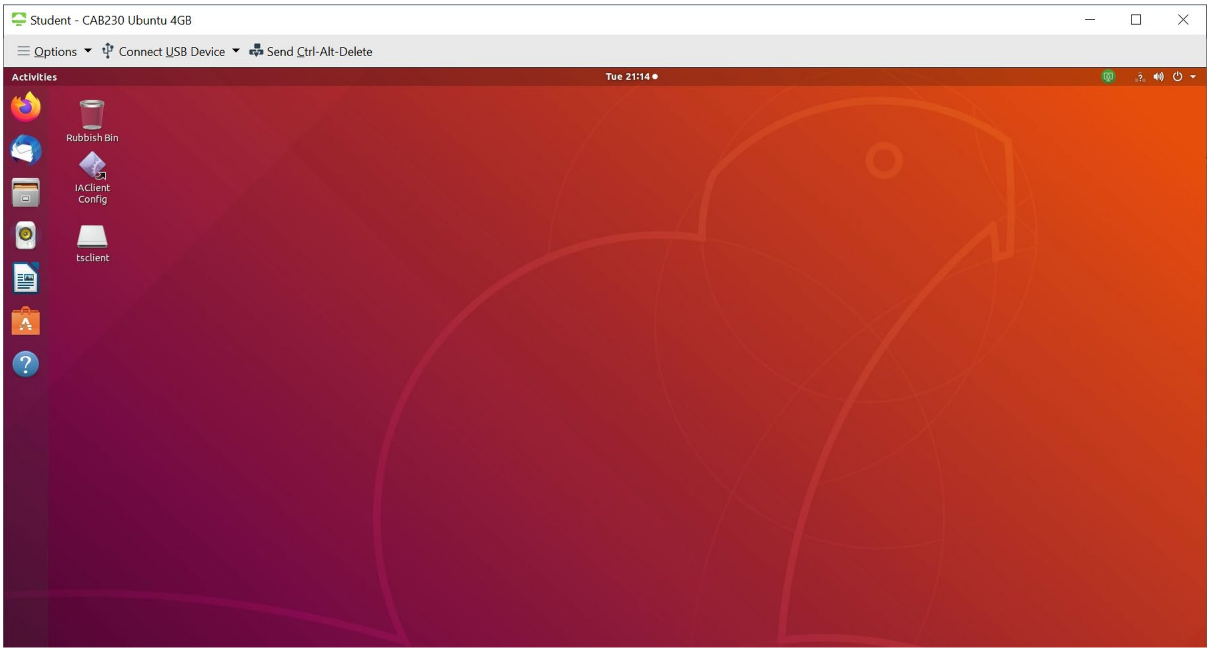
Password:

Domain: QUTAD

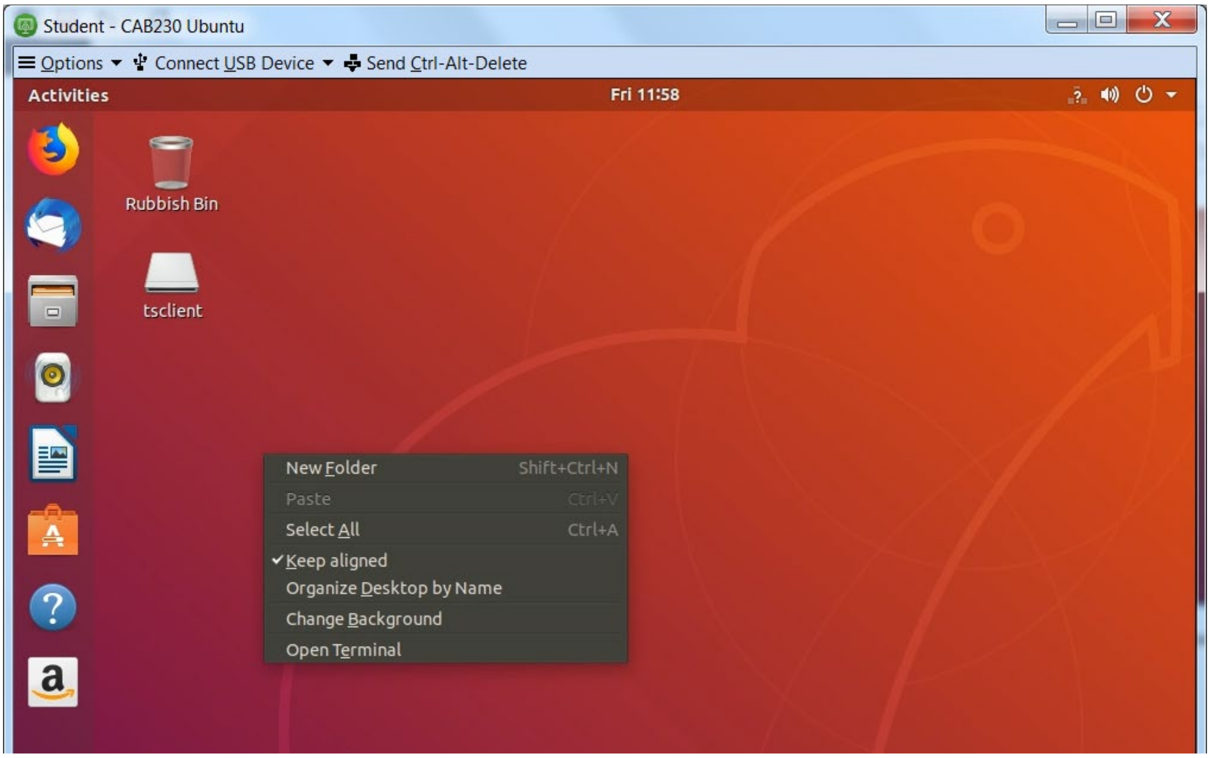
Once authenticated, you will see an Ubuntu desktop VM. To login to the VM, you work with the user account `cab230`, and you need to enter the password: `Cab230!` Please note the upper case ‘C’ in the password. This password is also used for the MySQL root password. We recommend that you do not change it – the QUTAD credentials provide the security anyway.

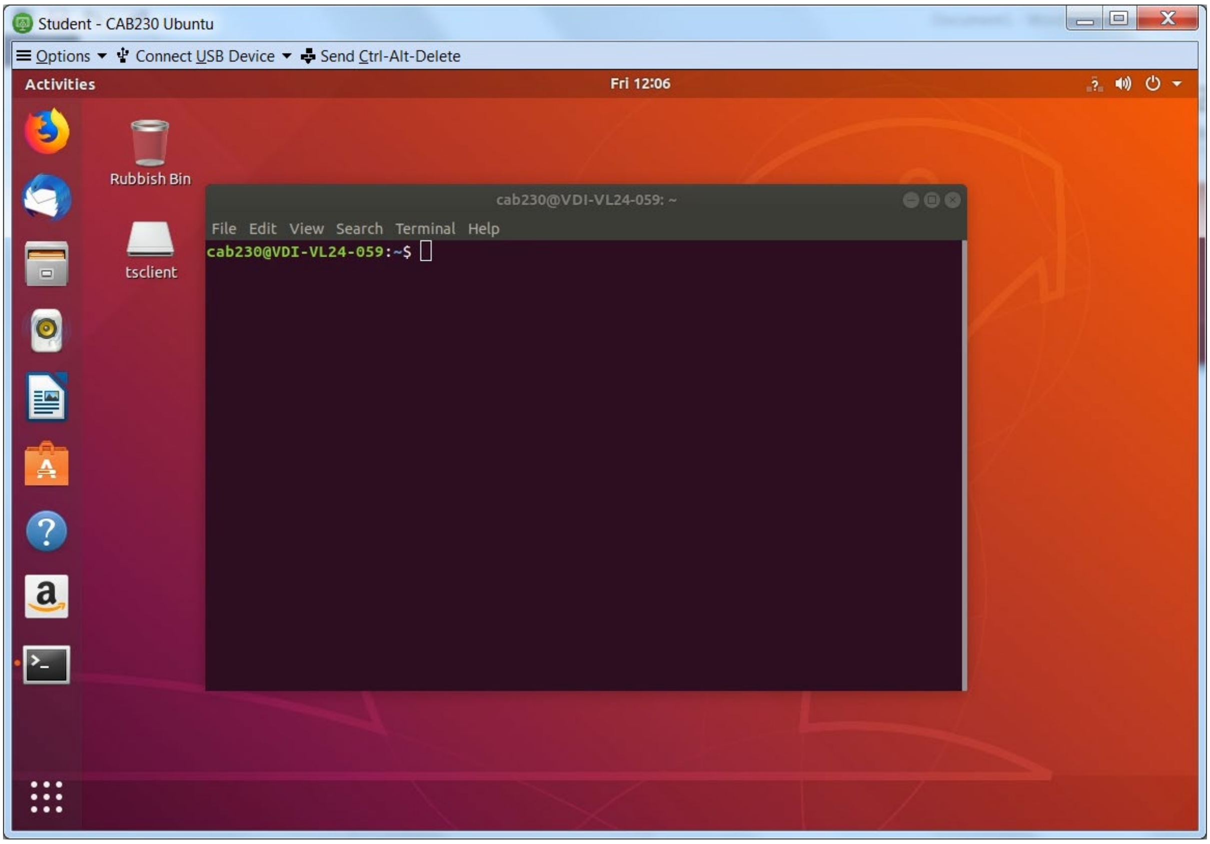


After you login, you will see the standard Ubuntu desktop, as shown cropped below. Right click on the mouse and you will see the black context menu as shown. Select the last option and you will then launch a Linux terminal, in which we will do most of our work. The terminal will open in your home directory – as you will see later, the Linux file manager app has this available as a menu option and this will be the easiest way of transferring files to the VM. We give you guidance on this below.



In what follows I will largely use screenshots from 2019, though the system is largely unchanged: we still use the Ubuntu 18.04 VM.





```
cab230@VDI-VL24-059:~$ 
```

As noted above, the machine image will already have node, npm and MySQL available for you. You should check that they are available by running some simple commands in the terminal.

Installing the DB

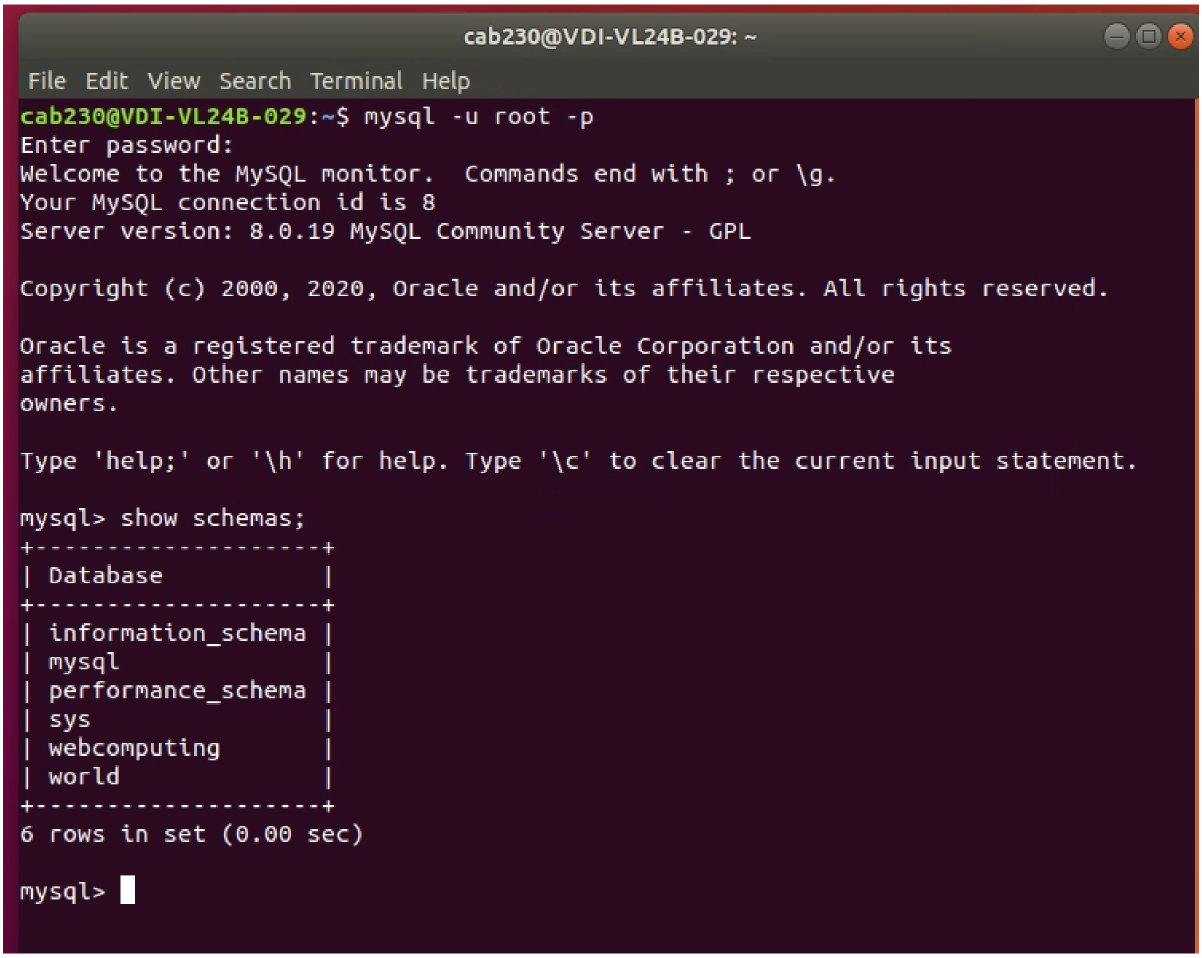
We now turn our attention to MySQL. There is a guide to using MySQL under Linux in the Getting Started with the VM Guide at the left hand side menu on BB. Most of the steps required for deployment of the assignment are covered in the video in the Assignment 2 section on BB, but a this will be a quick reminder. In the screenshot on the next page, we login to the server at the command line using the root account and the password `Cab230!`

The schemas shown in my case include `webcomputing`, the one you will need for the assignment. In the video this is loaded using the command:

```
mysql> source stocks.sql;
```

Follow the approach in the JWT Server Side worksheet and the example in the video to create the table needed for your application user information. If you have already created this locally you should export the user table to a sql script file and follow a similar process to the approach for the stocks table.

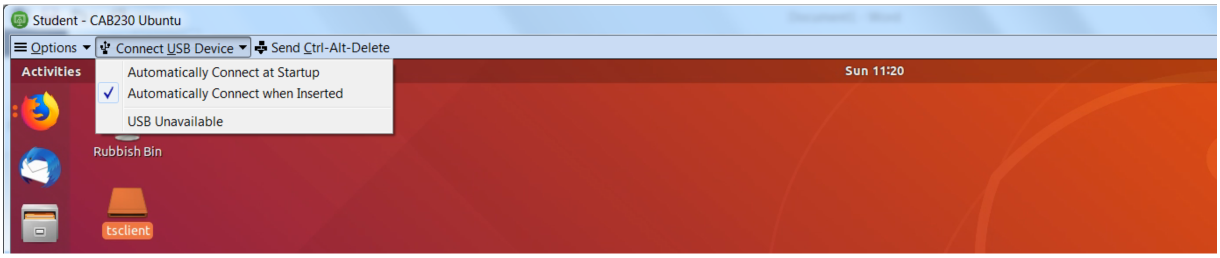
In the case of the example World Cities application, the approach is very similar. We will instead use the `mysqDump.sql` file discussed below in the file transfer section.



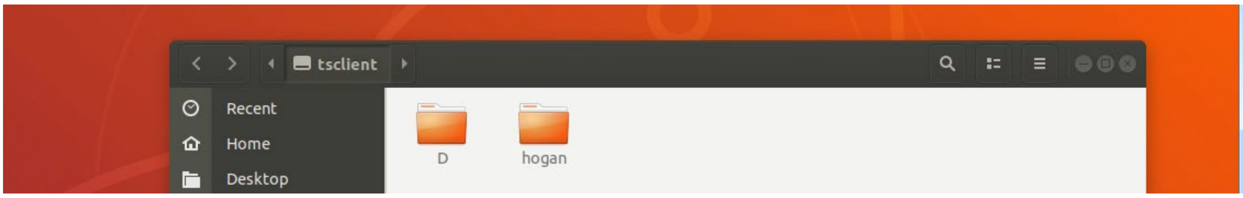
```
cab230@VDI-VL24B-029:~$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.19 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show schemas;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| webcomputing       |
| world              |
+--------------------+
6 rows in set (0.00 sec)

mysql> 
```

Deploying Code to the VM

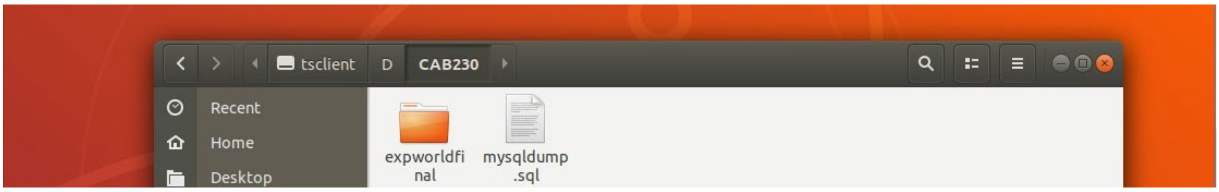
There are a variety of ways to transfer files to the VM, but perhaps the easiest approach is to use an external USB or to copy off your student drive. Both should be available by default. Click on the USB menu as shown so that the drive will automatically connect when inserted.



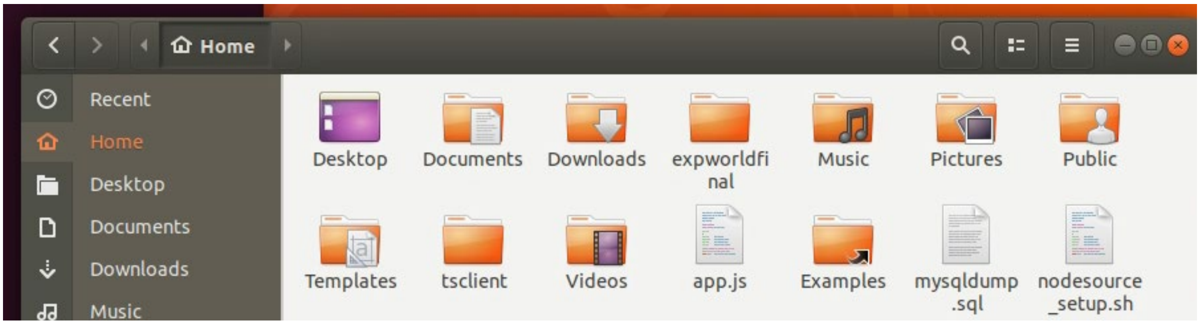
In my case, when I double click on `tsclient`, I see the options below, with drive D corresponding to an external drive plugged into my Windows machine. The drive letter and the view may vary from this one, but the USB drive should be available.



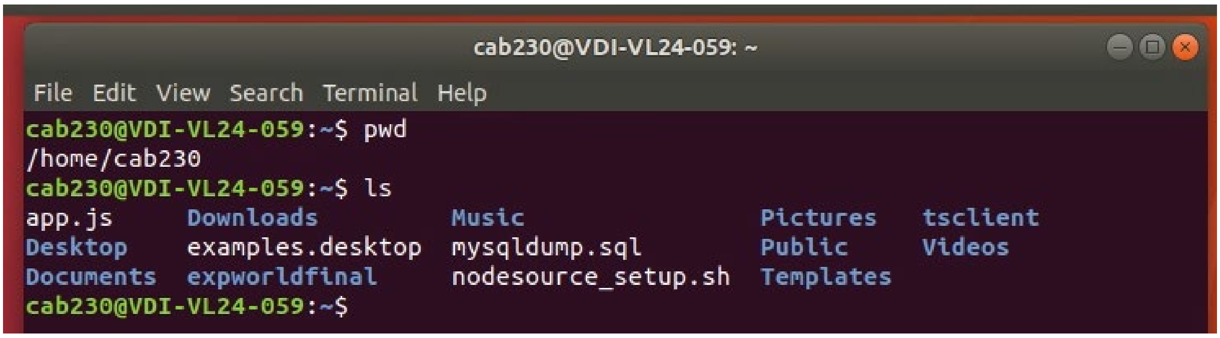
The easiest approach is to find the right directory on your USB, and then copy it to your `Home` directory – this is a menu option as you can see, and the directory your home directory in the terminal: `/home/cab230`. The approach is exactly the same as for Windows explorer, selecting the files and then doing a copy and paste into the home directory. In my list below I have used the example of the `expworld` code and the `mysqldump.sql` used for the world database. A reminder that you don't need this for the assignment.



After the copy is complete, we will see something like the image on the next page.

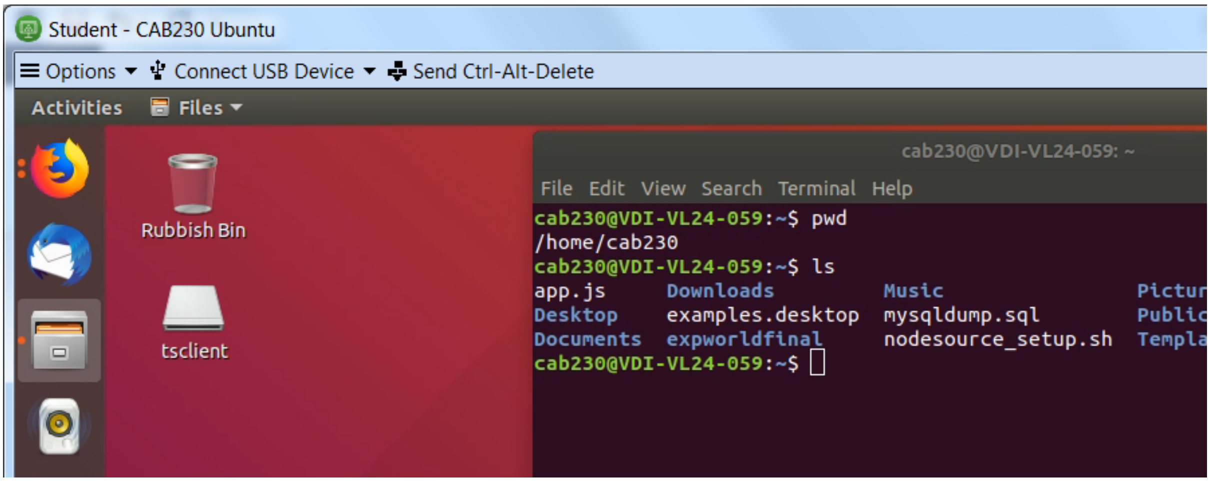


The corresponding terminal view is shown on the next page. Here `pwd` = Present Working Directory, and `ls` is the list files command.



```
cab230@VDI-VL24-059:~$ pwd
/home/cab230
cab230@VDI-VL24-059:~$ ls
app.js     Downloads         Music                Pictures   tsclient
Desktop    examples.desktop  mysqldump.sql        Public     Videos
Documents  expworldfinal     nodesource_setup.sh  Templates
cab230@VDI-VL24-059:~$
```

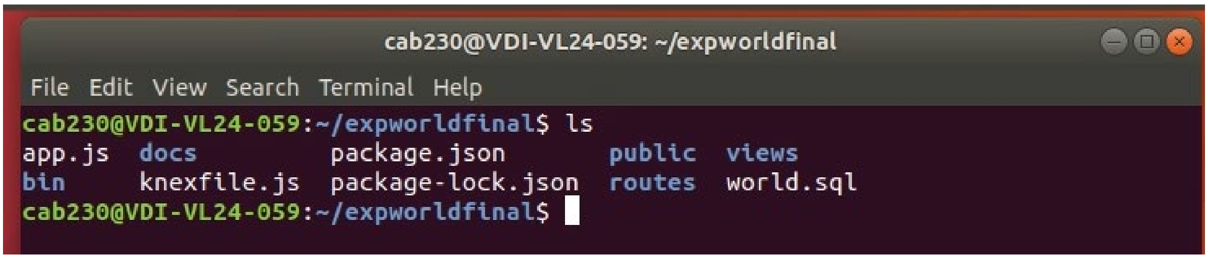
If you wish to look again at the file structure graphically, the files program is available from icon menu on the desktop. It is shown highlighted in the image below, third from the top:



```
cab230@VDI-VL24-059:~$ pwd
/home/cab230
cab230@VDI-VL24-059:~$ ls
app.js     Downloads         Music                Pictur
Desktop    examples.desktop  mysqldump.sql        Public
Documents  expworldfinal     nodesource_setup.sh  Templa
cab230@VDI-VL24-059:~$ 
```

Modules, DB Connection and Security Certificates

We now install node packages, edit the DB connection and install the security certificate for our application. The example here is again based on the World Cities express application.

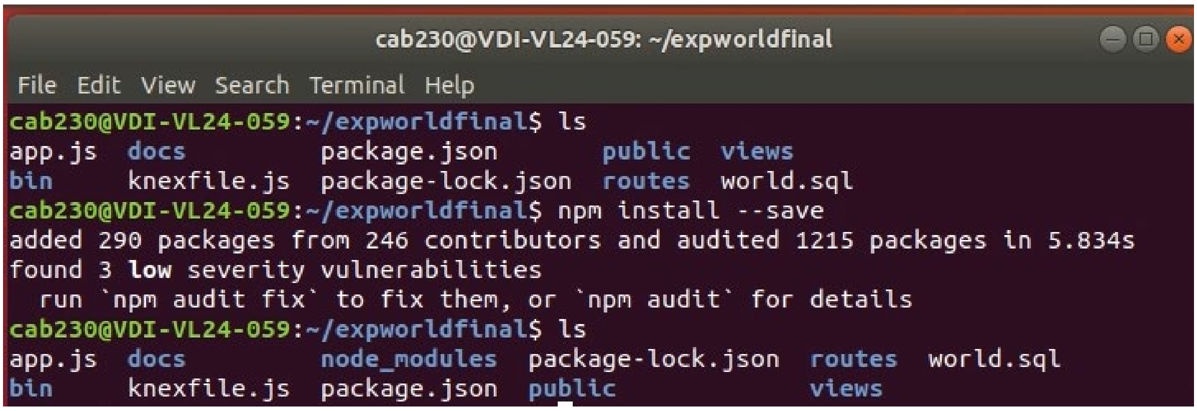

cab230@VDI-VL24-059: ~/expworldfinal
File Edit View Search Terminal Help

```
cab230@VDI-VL24-059:~/expworldfinal$ ls
app.js  docs         package.json       public  views
bin     knexfile.js  package-lock.json  routes  world.sql
cab230@VDI-VL24-059:~/expworldfinal$
```

Before transferring the application, you should make sure you delete the `node_modules` folder completely. Here I have also removed the directory `sslcert` in which I had stored the certificate files – the conventions are different under Linux. The first thing I am going to do is to install the node modules I require. In order to do this, we need to ensure that we have QUT Access authentication. This is handled using a dedicated IAS client, in a process described in an appendix to this document and in the earlier guide. The command to install the node modules locally is, as usual:

```
$ npm install --save
```

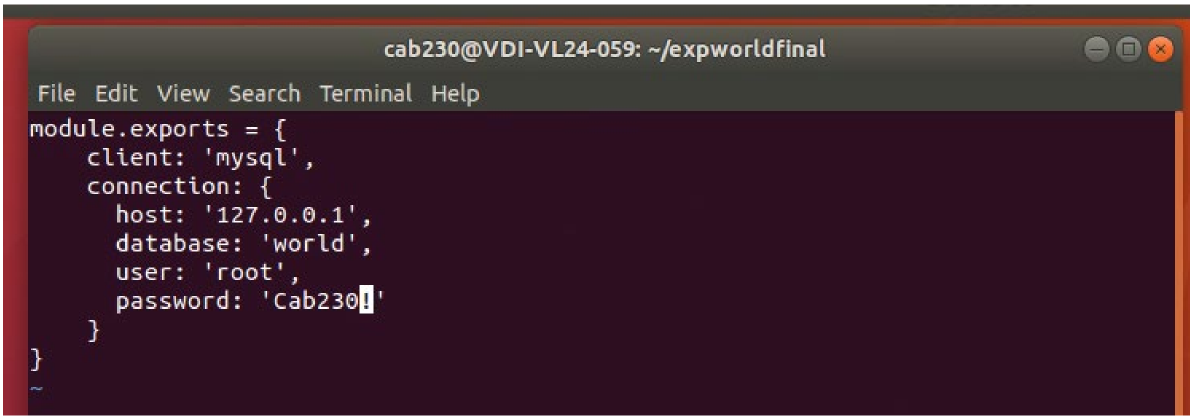
The response is shown below, and the big difference is of course the appearance of a `node_modules` directory.


cab230@VDI-VL24-059: ~/expworldfinal
File Edit View Search Terminal Help

```
cab230@VDI-VL24-059:~/expworldfinal$ ls
app.js  docs         package.json       public  views
bin     knexfile.js  package-lock.json  routes  world.sql
cab230@VDI-VL24-059:~/expworldfinal$ npm install --save
added 290 packages from 246 contributors and audited 1215 packages in 5.834s
found 3 low severity vulnerabilities
  run `npm audit fix` to fix them, or `npm audit` for details
cab230@VDI-VL24-059:~/expworldfinal$ ls
app.js  docs         node_modules  package-lock.json  routes  world.sql
bin     knexfile.js  package.json  public             views
```

Even though we are on a completely different machine, the MySQL installation still operates on `localhost` at the standard port of `3306`. Depending on how you have managed these parameters, you will need to edit `knexfile.js` or the `.env` file in the top directory of your application. Either way, password *must* change to `Cab230!` (or some other selection if you have created another DB user) and you must make sure that you have the appropriate database schema, which should be `webcomputing` rather than `world` for the assignment.

The screenshot on the next page shows an example knexfile.js for the World Cities app, where the port remains the default (3306) and we use the root account.


cab230@VDI-VL24-059: ~/expworldfinal
File Edit View Search Terminal Help

```
module.exports = {
    client: 'mysql',
    connection: {
      host: '127.0.0.1',
      database: 'world',
      user: 'root',
      password: 'Cab230!'
    }
}
~
```

We now move on to setting up the certificate so that we can serve pages via HTTPS. To get a self-signed certificate, we will follow the prac and the guide for Apache here:

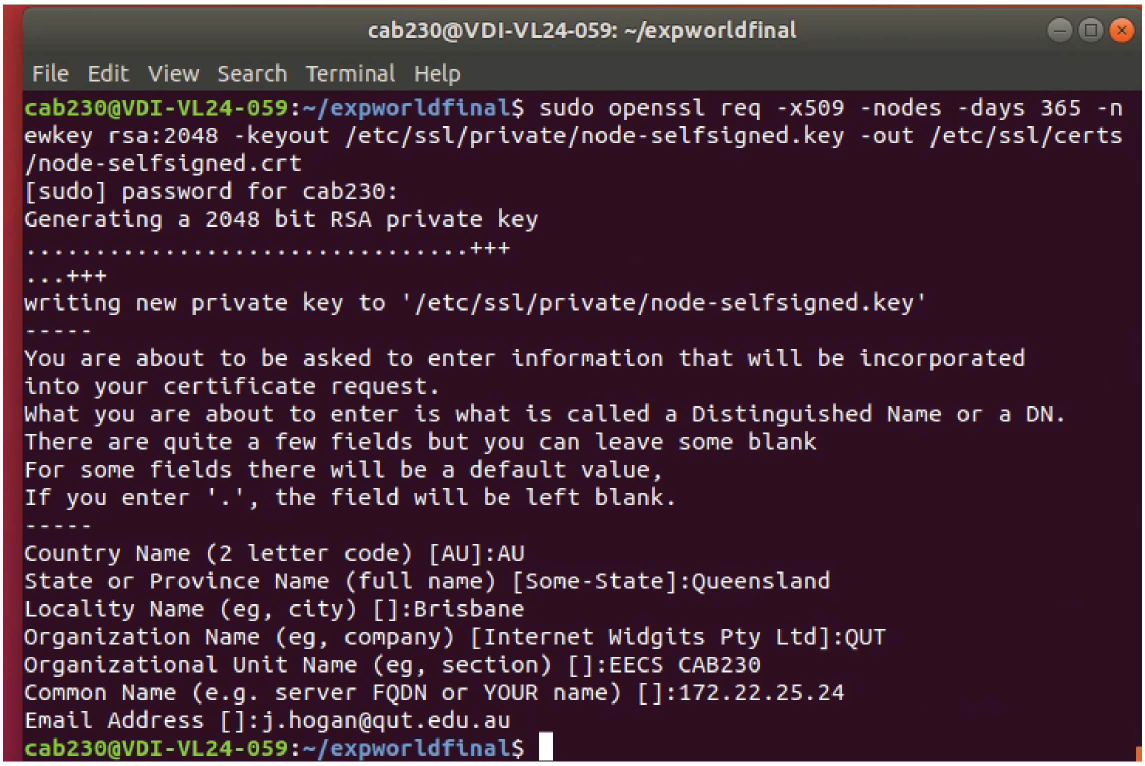
https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-apache-in-ubuntu-18-04.

Keys and certificates are stored in defined locations in the Linux filesystem, so our local folder is not needed. Instead we use:

- `/etc/ssl/private – for keys`
- `/etc/ssl/certs – for certificates`

The `openssl` command is as shown below. The parameters are explained in the Apache guide linked above. The utility openssl is installed by default in Ubuntu 18.04.

```
$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -
keyout /etc/ssl/private/node-selfsigned.key -out
/etc/ssl/certs/node-selfsigned.crt
```

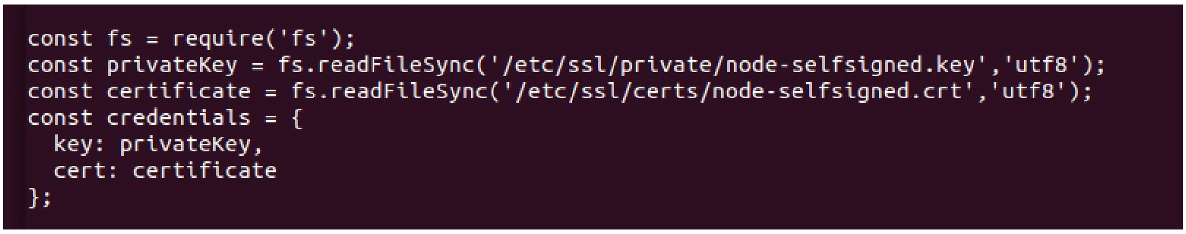

cab230@VDI-VL24-059: ~/expworldfinal
File Edit View Search Terminal Help

```
cab230@VDI-VL24-059:~/expworldfinal$ sudo openssl req -x509 -nodes -days 365 -n
ewkey rsa:2048 -keyout /etc/ssl/private/node-selfsigned.key -out /etc/ssl/certs
/node-selfsigned.crt
[sudo] password for cab230:
Generating a 2048 bit RSA private key
...............................+++
...+++
writing new private key to '/etc/ssl/private/node-selfsigned.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:AU
State or Province Name (full name) [Some-State]:Queensland
Locality Name (eg, city) []:Brisbane
Organization Name (eg, company) [Internet Widgits Pty Ltd]:QUT
Organizational Unit Name (eg, section) []:EECS CAB230
Common Name (e.g. server FQDN or YOUR name) []:172.22.25.24
Email Address []:j.hogan@qut.edu.au
cab230@VDI-VL24-059:~/expworldfinal$ 
```

An example session using 172.22.25.24 is shown above. Please see the appendix to learn how to find your VM IP address. Note that the VM is not internet-facing and this address is from the IP range reserved for local networks.

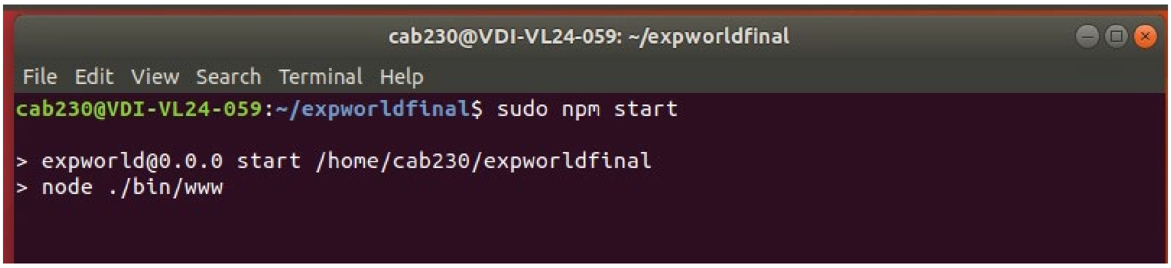
IMPORTANT NOTE: One of the limitations of the environment is that the IP address of your VM in the QUT network may change if you disconnect from the machine. This will mean that your certificate may not be consistent with the server. As you are working with a self-signed certificate, the practical effect may be limited. However, you should check this. A new self-signed certificate will take less than five minutes to organise in any case.

In a more professional deployment you would almost certainly need to deal with firewall settings as we did in the NGINX example. As this VM is behind the QUT firewall, we will not consider using `ufw` here. Our final step is to edit the path to the key and certificate files. The location of this code may vary from mine, but in this example, the code is in the `bin` directory under the main expworld application directory. I am here editing the file `www`:



```
const fs = require('fs');
const privateKey = fs.readFileSync('/etc/ssl/private/node-selfsigned.key','utf8');
const certificate = fs.readFileSync('/etc/ssl/certs/node-selfsigned.crt','utf8');
const credentials = {
  key: privateKey,
  cert: certificate
};
```

My initial test is then to move back to the top directory of `expworld`, and to launch the app using `npm start`. But if we want to serve HTTPS on Linux, we need elevated status:



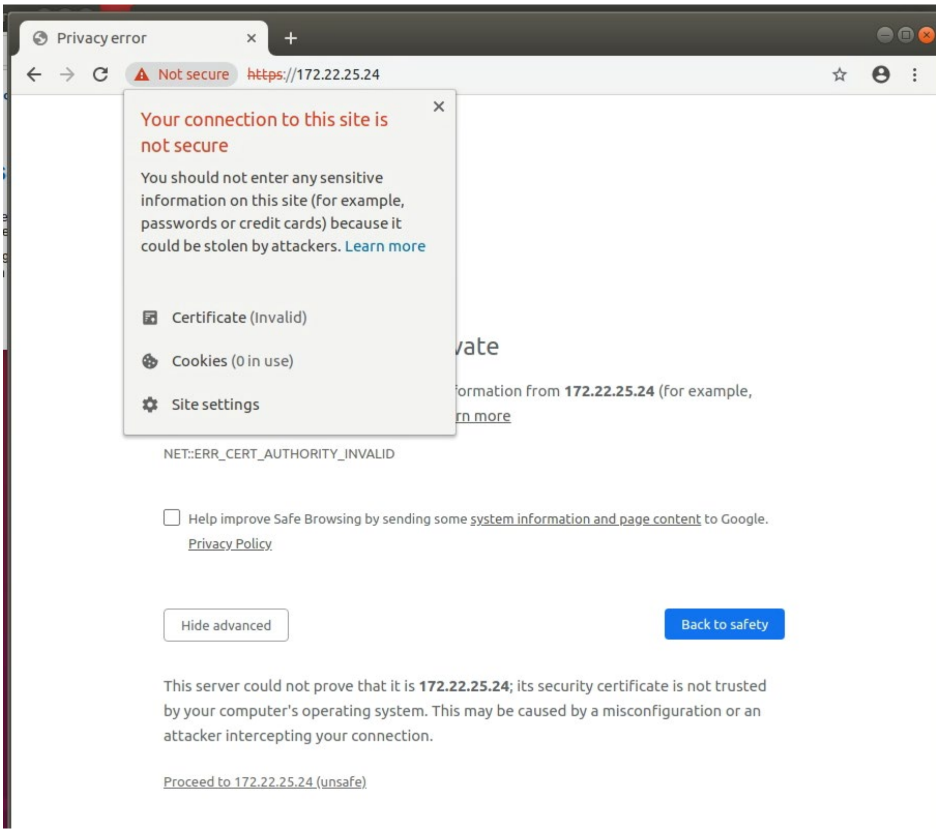
cab230@VDI-VL24-059: ~/expworldfinal
File Edit View Search Terminal Help

```
cab230@VDI-VL24-059:~/expworldfinal$ sudo npm start

> expworld@0.0.0 start /home/cab230/expworldfinal
> node ./bin/www
```

The use of `sudo` means that the application has root level privileges. If you want to continue to use the terminal you can add an ampersand '`&`' at the end of the command to execute it in background. When you submit, we will ask you to make the application available using pm2 – this will keep the application running so that we can hit it with our tests.

When we connect to this IP address on the QUT network via HTTPS, we see a warning, which is perfectly sensible as the certificate has not come from a trusted authority. We continue via the advanced button and accept the invitation to view the certificate.

When we see that the information in the certificate matches the values we expect, then we will be satisfied that HTTPS is working correctly (see the following pages). During marking, we will not go to the extent of entering each of your certificate 'issuing authorities' as trusted, as we did in the prac, as this would slow things up rather too much. So at this point, we are most of the way there, and for a one-off deployment, we can even work with what we have.



Your connection to this site is not secure

You should not enter any sensitive information on this site (for example, passwords or credit cards) because it could be stolen by attackers. Learn more

ormation from **172.22.25.24** (for example,
rn more

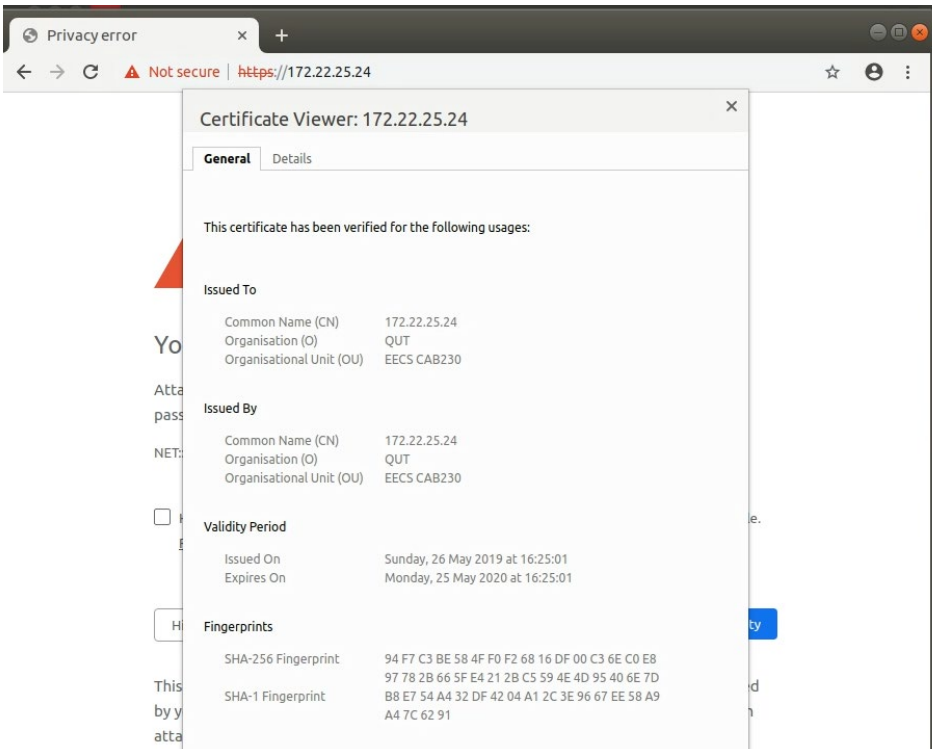
NET::ERR_CERT_AUTHORITY_INVALID

□ Help improve Safe Browsing by sending some system information and page content to Google. Privacy Policy

This server could not prove that it is **172.22.25.24**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

Proceed to 172.22.25.24 (unsafe)

Here we see the self-signed certificate details…



Certificate Viewer: 172.22.25.24

This certificate has been verified for the following usages:

Issued To

Common Name (CN)	172.22.25.24
Organisation (O)	QUT
Organisational Unit (OU)	EECS CAB230

Issued By

Common Name (CN)	172.22.25.24
Organisation (O)	QUT
Organisational Unit (OU)	EECS CAB230

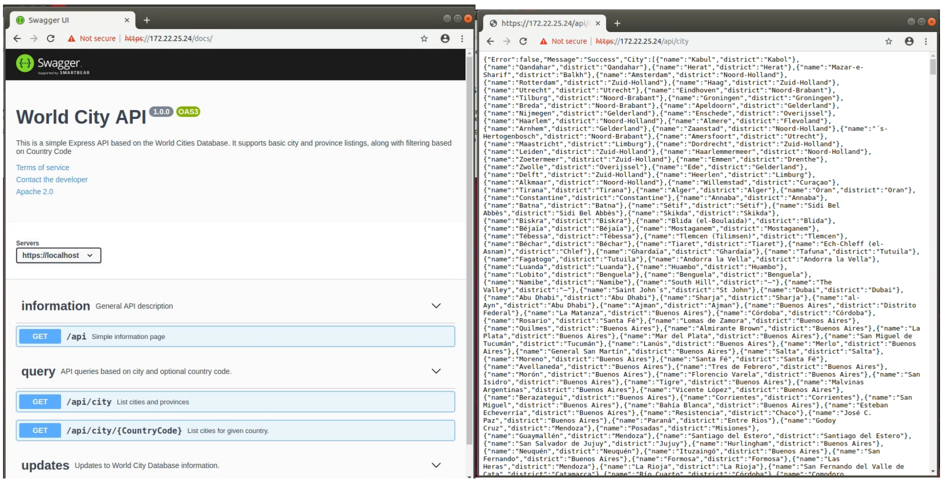
Validity Period

Issued On	Sunday, 26 May 2019 at 16:25:01
Expires On	Monday, 25 May 2020 at 16:25:01

Fingerprints

SHA-256 Fingerprint	94 F7 C3 BE 58 4F F0 F2 68 16 DF 00 C3 6E C0 E8 97 78 2B 66 5F E4 21 2B C5 59 4E 4D 95 40 6E 7D
SHA-1 Fingerprint	B8 E7 54 A4 32 DF 42 04 A1 2C 3E 96 67 EE 58 A9 A4 7C 62 91

The application deploys and runs correctly, though the API tests from within Swagger may fail due to the HTTPS error. When running this app on a Linux system for the first time, I needed to change the `knex` code in the router to refer to the table "`City`" rather than "`city`". So while your assignment will differ, if you have errors, check the logs and look for simple solutions first.



World City API 1.0.0 OAS3

This is a simple Express API based on the World Cities Database. It supports basic city and province listings, along with filtering based on Country Code

information General API description

GET `/api` Simple information page

query API queries based on city and optional country code.

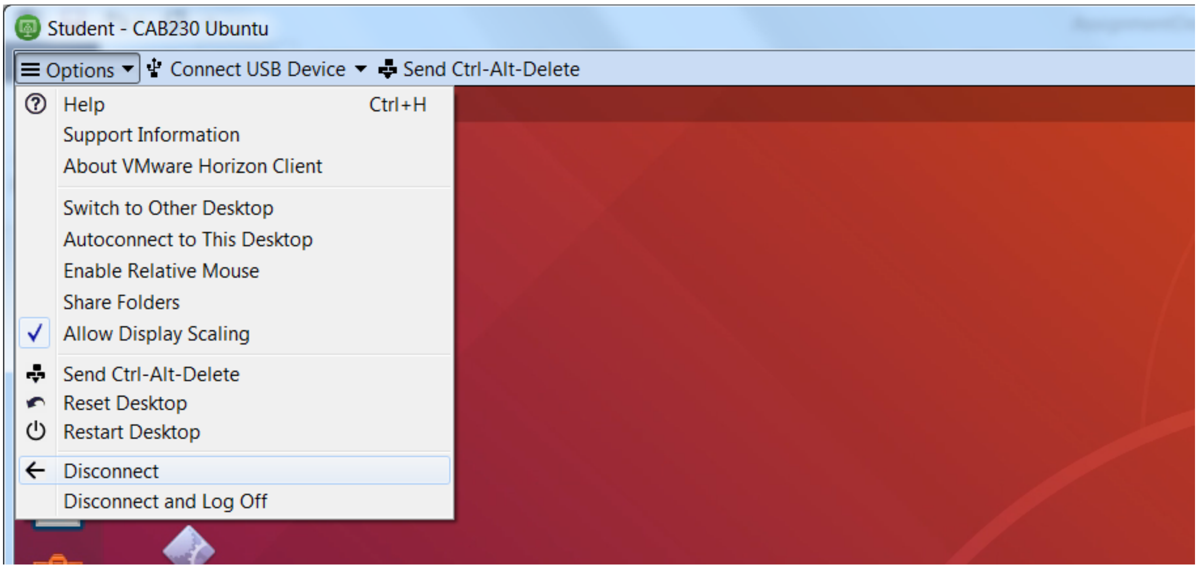
GET `/api/city` List cities and provinces

GET `/api/city/{CountryCode}` List cities for given country.

updates Updates to World City Database information.

```
{"Error":false,"Message":"Success","City":[{"name":"Kabul","district":"Kabol"},
{"name":"Qandahar","district":"Qandahar"},{"name":"Herat","district":"Herat"},{"name":"Mazar-e-
Sharif","district":"Balkh"},{"name":"Amsterdam","district":"Noord-Holland"},
{"name":"Rotterdam","district":"Zuid-Holland"},{"name":"Haag","district":"Zuid-Holland"},
{"name":"Utrecht","district":"Utrecht"},{"name":"Eindhoven","district":"Noord-Brabant"},
{"name":"Tilburg","district":"Noord-Brabant"},{"name":"Groningen","district":"Groningen"},
{"name":"Breda","district":"Noord-Brabant"},{"name":"Apeldoorn","district":"Gelderland"},
{"name":"Nijmegen","district":"Gelderland"},{"name":"Enschede","district":"Overijssel"},
{"name":"Haarlem","district":"Noord-Holland"},{"name":"Almere","district":"Flevoland"},
{"name":"Arnhem","district":"Gelderland"},{"name":"Zaanstad","district":"Noord-Holland"},{"name":"´s-
Hertogenbosch","district":"Noord-Brabant"},{"name":"Amersfoort","district":"Utrecht"},
{"name":"Maastricht","district":"Limburg"},{"name":"Dordrecht","district":"Zuid-Holland"},
{"name":"Leiden","district":"Zuid-Holland"},{"name":"Haarlemmermeer","district":"Noord-Holland"},
{"name":"Zoetermeer","district":"Zuid-Holland"},{"name":"Emmen","district":"Drenthe"},
{"name":"Zwolle","district":"Overijssel"},{"name":"Ede","district":"Gelderland"},
{"name":"Delft","district":"Zuid-Holland"},{"name":"Heerlen","district":"Limburg"},
{"name":"Alkmaar","district":"Noord-Holland"},{"name":"Willemstad","district":"Curaçao"},
{"name":"Tirana","district":"Tirana"},{"name":"Alger","district":"Alger"},{"name":"Oran","district":"Oran"},
{"name":"Constantine","district":"Constantine"},{"name":"Annaba","district":"Annaba"},
{"name":"Batna","district":"Batna"},{"name":"Sétif","district":"Sétif"},{"name":"Sidi Bel
Abbès","district":"Sidi Bel Abbès"},{"name":"Skikda","district":"Skikda"},
{"name":"Biskra","district":"Biskra"},{"name":"Blida (el-Boulaida)","district":"Blida"},
{"name":"Béjaïa","district":"Béjaïa"},{"name":"Mostaganem","district":"Mostaganem"},
{"name":"Tébessa","district":"Tébessa"},{"name":"Tlemcen (Tilimsen)","district":"Tlemcen"},
{"name":"Béchar","district":"Béchar"},{"name":"Tiaret","district":"Tiaret"},{"name":"Ech-Chleff (el-
Asnam)","district":"Chlef"},{"name":"Ghardaïa","district":"Ghardaïa"},{"name":"Tafuna","district":"Tutuila"},
{"name":"Fagatogo","district":"Tutuila"},{"name":"Andorra la Vella","district":"Andorra la Vella"},
{"name":"Luanda","district":"Luanda"},{"name":"Huambo","district":"Huambo"},
{"name":"Lobito","district":"Benguela"},{"name":"Benguela","district":"Benguela"},
{"name":"Namibe","district":"Namibe"},{"name":"South Hill","district":"–"},{"name":"The
Valley","district":"–"},{"name":"Saint John´s","district":"St John"},{"name":"Dubai","district":"Dubai"},
{"name":"Abu Dhabi","district":"Abu Dhabi"},{"name":"Sharja","district":"Sharja"},{"name":"al-
Ayn","district":"Abu Dhabi"},{"name":"Ajman","district":"Ajman"},{"name":"Buenos Aires","district":"Distrito
Federal"},{"name":"La Matanza","district":"Buenos Aires"},{"name":"Córdoba","district":"Córdoba"},
{"name":"Rosario","district":"Santa Fé"},{"name":"Lomas de Zamora","district":"Buenos Aires"},
{"name":"Quilmes","district":"Buenos Aires"},{"name":"Almirante Brown","district":"Buenos Aires"},{"name":"La
Plata","district":"Buenos Aires"},{"name":"Mar del Plata","district":"Buenos Aires"},{"name":"San Miguel de
Tucumán","district":"Tucumán"},{"name":"Lanús","district":"Buenos Aires"},{"name":"Merlo","district":"Buenos
Aires"},{"name":"General San Martín","district":"Buenos Aires"},{"name":"Salta","district":"Salta"},
{"name":"Moreno","district":"Buenos Aires"},{"name":"Santa Fé","district":"Santa Fé"},
{"name":"Avellaneda","district":"Buenos Aires"},{"name":"Tres de Febrero","district":"Buenos Aires"},
{"name":"Morón","district":"Buenos Aires"},{"name":"Florencio Varela","district":"Buenos Aires"},{"name":"San
Isidro","district":"Buenos Aires"},{"name":"Tigre","district":"Buenos Aires"},{"name":"Malvinas
Argentinas","district":"Buenos Aires"},{"name":"Vicente López","district":"Buenos Aires"},
{"name":"Berazategui","district":"Buenos Aires"},{"name":"Corrientes","district":"Corrientes"},{"name":"San
Miguel","district":"Buenos Aires"},{"name":"Bahía Blanca","district":"Buenos Aires"},{"name":"Esteban
Echeverría","district":"Buenos Aires"},{"name":"Resistencia","district":"Chaco"},{"name":"José C.
Paz","district":"Buenos Aires"},{"name":"Paraná","district":"Entre Rios"},{"name":"Godoy
Cruz","district":"Mendoza"},{"name":"Posadas","district":"Misiones"},
{"name":"Guaymallén","district":"Mendoza"},{"name":"Santiago del Estero","district":"Santiago del Estero"},
{"name":"San Salvador de Jujuy","district":"Jujuy"},{"name":"Hurlingham","district":"Buenos Aires"},
{"name":"Neuquén","district":"Neuquén"},{"name":"Ituzaingó","district":"Buenos Aires"},{"name":"San
Fernando","district":"Buenos Aires"},{"name":"Formosa","district":"Formosa"},{"name":"Las
Heras","district":"Mendoza"},{"name":"La Rioja","district":"La Rioja"},{"name":"San Fernando del Valle de
Cata","district":"Catamarca"},{"name":"Río Cuarto","district":"Córdoba"},{"name":"Comodoro
```

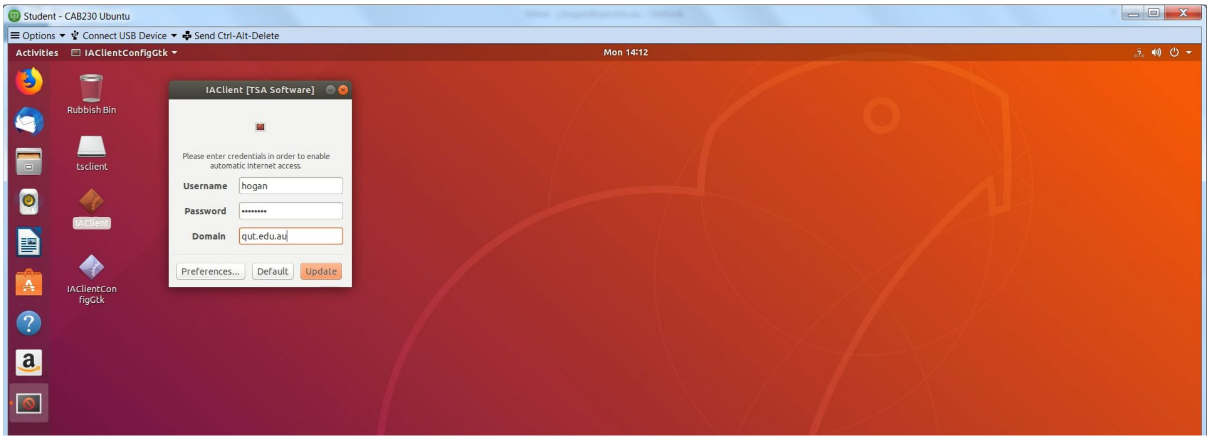
Finally, you might like to consider how to leave your VM behind. The key is to look at the menu in the top left hand corner of the VMware Client:



Generally you should `Disconnect` or `Disconnect and Log Off`. A `Restart Desktop` may be necessary on occasions and may result in a fresh IP address. If you see a message and a cross icon saying that the machine is not available for now, just give it a few minutes to start up again.

Appendix A – IAS Authentication

In order to connect to the outside world, you need to have QUT Access authorisation. On these VMs, this is handled using a dedicated IAS client on the desktop. Use of this client is no different from usual, but we have included here a screenshot to help you get started. The client requires your normal QUT login details, and enter the domain: `qut.edu.au`



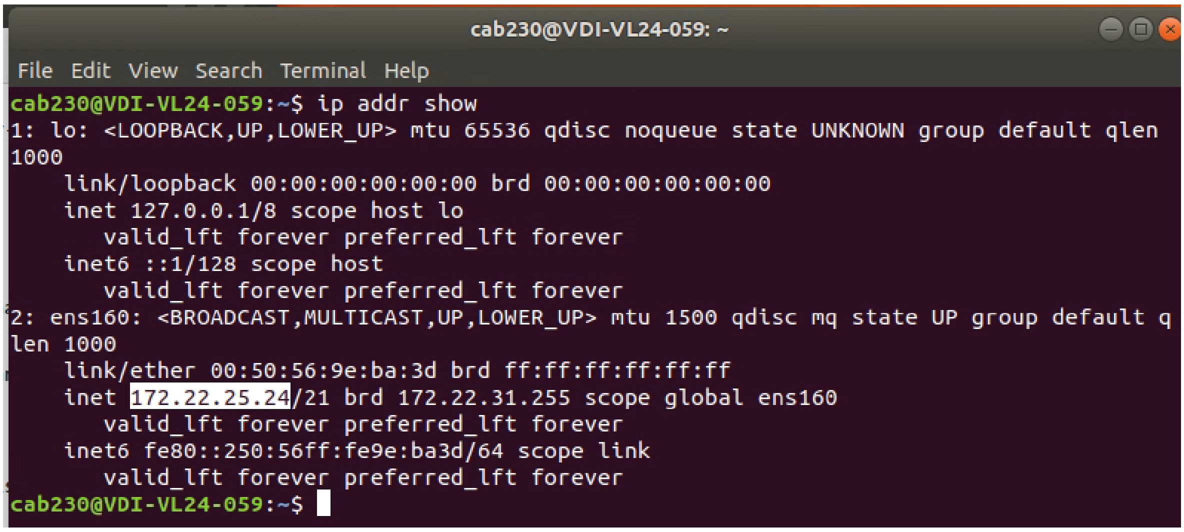
Please enter credentials in order to enable automatic Internet access.

Username: hogan
Password: ••••••••
Domain: qut.edu.au

From there, you will see a status message somewhere on the right hand side of the desktop, and you will then be able to connect to the internet. The internet, however, will not be able to connect to you.

Appendix B – Your IP Address

This short appendix covers finding the IP address of your VM from within Ubuntu. As you will be within the QUT network, your machine is not internet facing and so the IP will be assigned within the ranges expected for a local network. However, your machine will be accessible elsewhere on the QUT network for the appropriate requests. Probably the quickest way to find your IP address is to use terminal:



```
cab230@VDI-VL24-059:~$ ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen
1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: ens160: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default q
len 1000
    link/ether 00:50:56:9e:ba:3d brd ff:ff:ff:ff:ff:ff
    inet 172.22.25.24/21 brd 172.22.31.255 scope global ens160
       valid_lft forever preferred_lft forever
    inet6 fe80::250:56ff:fe9e:ba3d/64 scope link
       valid_lft forever preferred_lft forever
cab230@VDI-VL24-059:~$
```

The command shows the results for a number of network interfaces, with the top one obviously localhost. The highlighted address is the one you want.