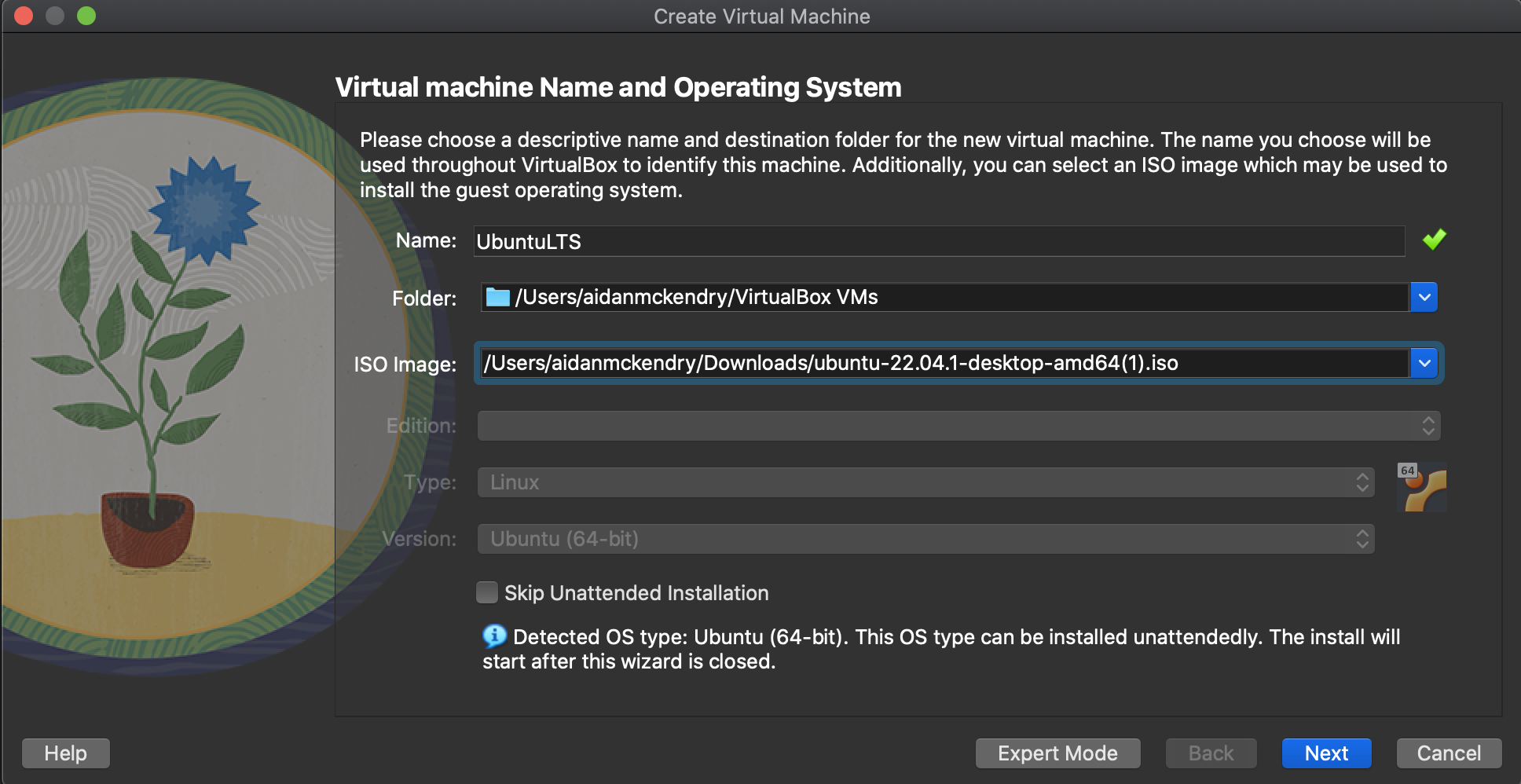
**Project documentation**

Installed VirtualBox for Mac OS (<https://www.virtualbox.org/wiki/Downloads>)

Downloaded ubuntu version 22.04.1 (<https://ubuntu.com/download/desktop>)

* Previously had linux (22.04) and downloaded the ISO file from the official ubuntu website working however this version is not available since.

Created a new virtual machine



Set the username and password

Allocated 4gb RAM and 2 CPUs to the VM.

A screenshot of a computer

Description automatically generated with medium confidence

I then set the memory of the virtual hard drive to 25GB.

Selected Finish to begin the unattended installation process.

* Outcome:
  + Begins running the VM and brings me to a loading screen
  + This froze after ~10 minutes
* Troubleshooting:
  + Powered off the VM and restarted (tried twice)
  + On third try I received errors explaining “Failed to start hostname service”, “Failed to start CUPS scheduler”, “Failed to start live CD Installer”
  + Hold shift during boot and ran in (safe graphics) mode
    - This however brought me to the same loading screen and did not progress from here after ~30 mins.
  + Trying pre allocating the 25GB memory on set up
    - Froze after ~10 minutes.
    - On safe graphics mode – froze again.

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generatedText

Description automatically generated

I downloaded ubuntu 20.04.5 and repeated the creation steps above with this new ISO image

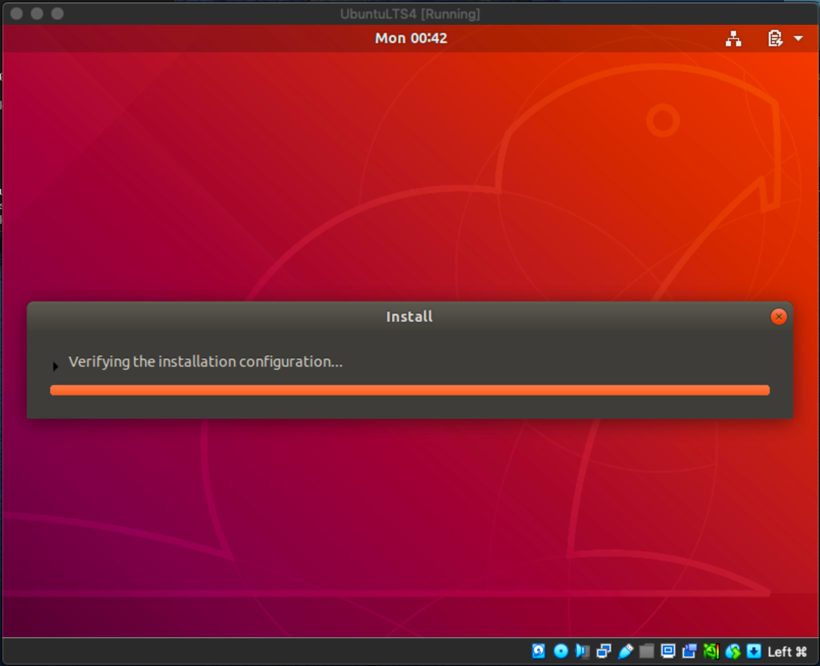
On first boot the install froze at 73% of checking disks - froze while checking the casper/filesystem.squahfs.

Second boot froze at 83% - froze while checking the casper/filesystem.squahfs.

Third boot I used the install without installing (safe graphics) mode however this again froze while checking the casper/filesystem.squahfs

I then downloaded ubuntu 18.04.6 and retried the above creation steps.

1st boot gets past a loading screen however reaches a black screen and does not progress any firther with the installation & the loading cursor icon freezes.

2nd boot progressed to a loading screen however froze while checking installation configuration.

Consequent attempts to start the installation resulted in freezing as well.

I then tried to get the MYSQL example running on this VM instance.

1. Mkdir mysql
2. Mv Install\_mysql.sh /mysql
3. ./installmysql.sh

This returned an error asking for authentication in my VM shown below

Graphical user interface, text

Description automatically generated

1. Sudo install\_mysql.sh - I then tried executing it with the operation sudo however this returned an error that my user was not in the sudoers file.
2. To resolve this I added a new user to the VM with sudo permissions and named it simply newuser with the commands shown below

Graphical user interface, text, chat or text message

Description automatically generated

1. The newuser now has sudo permissions and will be used for the remainder of the installation process
2. Su newuser
3. cd mysql
4. sudo bash install\_mysql.sh (first returned an error)Text

   Description automatically generated
5. sudo ./install\_mysql.sh - I then amended the command (shown above) to which ran okay.
6. Cd .. (home directory)
7. Mkdir mongodb
8. Nano install\_mongodb.sh (creating a shell file to house a list of commands for silent installation of Mongodb)

Graphical user interface, application, Teams

Description automatically generated

1. The above is a screenshot of this first attempt at the shell file. During this attempt I encountered an error that read: “Ignoring file mongodb-org-5.0.list.d as it has an invalid file extension.”
2. A screenshot of a computer

   Description automatically generatedA screenshot of a computer

   Description automatically generatedI tried two more attempts which both encountered errors, shown below with install\_mongodb(\_take2&&\_take3).sh (in take3 I tried to set the versions of the packaged dependancies to 4.4 to avoid the error with version 5.0)

Both of these attempts encountered similar errors including a new error which said I held broken packages therefore I ruled out the idea of pinning mongodb to a previous version.

Take4 ended up being the first fully working script so I have documented its process separately.

1. Sudo apt-get purge mongodb-org\* (removing previous broken packages)
2. Sudo apt remove mongodb
3. Sudo apt autoremove
4. I wrote the below script again using nanoA screenshot of a computer

   Description automatically generated
5. Ran the script with sudo ./install\_mongodb\_take4.sh
6. At first attempt I was missing some dependant packages
7. I resolved this by running the following commands:
   1. sudo -i
   2. wget <http://archive.ubuntu.com.ubuntu/pool/main/o/openssl/libssl1.1._1.1.1f-1ubuntu2_amd64.deb>
   3. sudo dpkg -i libssl1.1\_1.1.1f-1ubuntu2\_amd64.deb
8. Rerunning line 7 of the shell file then installed all the packages fully
9. Sudo systemctl start mongod
10. A screenshot of a computer

    Description automatically generated with medium confidenceSudo systemctl status mongod (output shown in right screenshot)
11. I amended the shell file to execute these lines in the script every time.

Dockerising and silently installing mongodb

Text

Description automatically generatedFirstly, I created a dockerfile to begin with a base ubuntu image and run the silent install script and the import data script.

Dockerfile (Mongo)

I install all dependencies then run the scripts to perform silent installation.

The install\_mongodb\_without\_sudo.sh file has been copied in below.

Text

Description automatically generated

install\_mongodb\_without\_sudo.sh

I have commented out the lines related to installing version 6.0 of mongodb because this retuned an error: “Illegal operation: core dumped(mongod --version)” meaning this was not installed correctly.

I added the new line below wget -q0 … to download the certificate for mongodb version 4.4.

Next all of the packages are installed. With version 4.4 mongodb.

load\_data.sh

Text

Description automatically generated

Line:3 Service is started

Line:4 Verifies the status of the service being started

Line:5 Imports the dataset csv file

Line:4 Stops the service between layers

Graphical user interface, text, application

Description automatically generated

select\_data.sh

Line:3 Service is started

Line:5 should return the logs of the started service, an error in the path prevents this, since we are working inside the app directory of the container…

Troubleshooting:

* Tried cd to bring to root directory, then call tail -F /var/log/mongodb/mongod.log – returns no file or directory.
* Changing command to “../ var/log/mongodb/mongod.log” still returns no file/directory.
* Log file can still be viewed within the container shell despite this error.

Currently uses EXPOSE 27017 > ENTRYPOINT [“/usr/bin/mongod”]

Removing entrypoint and including CMD ./select\_data.sh

Added service mongod start and service mongod stop after importing csv file.

Expected outcome: should be able to docker exec to access shell inside container and view data with mongo…

Result: error in load\_data.sh returned a non-zero code. (1) – mongod: unrecognized service.

Troubleshooting:

* Removing service start and reverting to systemctl start/ stop commands
* This is because we are in the app directory, must cd to root before providing path to log file
* Entering the container with “sudo docker exec -it <container name> sh” we can see the files have been imported and are visible within the running mongo instance…
* Mongo instance is listening for connections on: “mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb”

Running node server.js at this stage gives an error: Connection <monitor> to 127.0.0.1:37017 closed.

Node JS connection error

Text

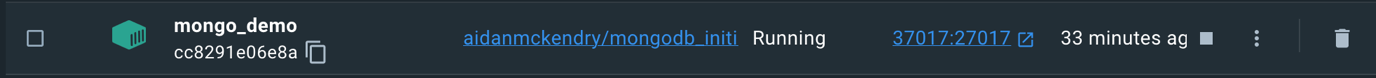
Description automatically generated

**Attempted solutions to connection of mongodb**

Troubleshooting:

* Enter directConnection=true parameter to connection string (StackOverflow) - Error persists.
* Whitelisting any IP connection in mongo configuration file (/etc/mongod.conf) by setting network bindIp to 0.0.0.0 should allow mongodb connections from any port

Can see this is updated correctly in the logs below but the error persists.

Graphical user interface, text, application

Description automatically generatedHere we can see the docker container hosting the database ready for connections on port 0.0.0.0:27107 (from /var/logs/mongodb/mongod.log).

I can also connect to the database via the mongo shell command within the container and view the loaded dataset to verify that mongo is running on the port further. (shown below)

Text

Description automatically generated

Promise-retry

* Ran “npm promise retry”
* I included the mongodb connection line within a try catch block using promise retry which executes the command N number of times. Each retry of this code resulted in the same error response returned from Mongoose.

I tried multiple different addresses for the mongo.connect address, port and table however each other address / port returns ECONNREFUSED error.

I have also tried changing the permissions of the mongo of the mongo library directory with the command “chown -R mongodb:mongodb /var/lib/mongodb”. This was recommended along with deleting the mongod.lock with the command rm however this is not present within my mongo container.



The one time I have received a successful connection to the silently installed database is shown above which was after I received the log message below once the container had been running for ~12 hours.

{"t":{"$date":"2023-01-24T05:31:21.935+00:00"},"s":"I", "c":"COMMAND", "id":23099, "ctx":"PeriodicTaskRunner","msg":"Task finished","attr":{"taskName":"DBConnectionPool-cleaner","durationMillis":1225}} …

does this have anything to do with app now connecting successfully?

Running atom on mac:

* command-space, then type atom

Docker on mac requires MacOS update

* Performing software update for this
* Downloaded and working

Mongoose JS app should connect to database and display all years when a curl request is made to “/allYears”. Will likely need to add environment variables into mongo such as username and password.

Defining this endpoint resulted in no records being returned. This seems to be an issue with the case sensitivity of the mongo collections. Collections should be named as all lower case since mongoose performs the find operation and it is always lower case, if the cases do not match, a new collection will be created in the database with no data stored.

Trouble shooting:

Updated the filename and collection name from SanFransiscoMonthlyTmpRecords to sanfransisco.

* New version of mongodb container needs pushed to dockerhub with tag 1.1.2
* Once pushed to dockerhub, update the version which is pulled in the docker-compose.yml

Mongoose seems to automatically add ‘s’ to the collection name that it tries to connect to. Resulting in no data returned because the collection is newly created. <https://stackoverflow.com/questions/10547118/why-does-mongoose-always-add-an-s-to-the-end-of-my-collection-name>

Pushing to dockerhub

docker login -u "<myusername>” -p "<mypassword>" docker.io

sudo docker push myusername/my\_mogodb:0.0.1

*log into dockerhub to screenshot image that was pushed*

Connecting to MongoDB

https://www.section.io/engineering-education/nodejs-mongoosejs-mongodb/

Installing node.js

Text

Description automatically generated