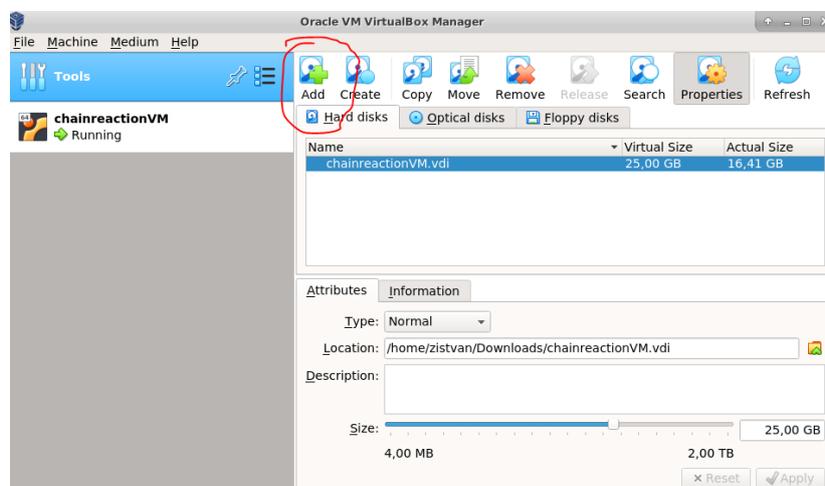


## How to verify that the VirtualBox image has been downloaded and configured correctly?

1) After downloading both the .vdi and .vbox files from the cloud storage, add the .vdi file to the list of drives in VirtualBox:



Then, by using Machine/Add, add the .vbox file so that a new virtual machine is created using the disk above.

2) Start the virtual machine and log in into the VM, then start a terminal.

3) Clone the first tutorial repository into the VM home:

```
cd /home/vmuser/  
git clone https://gitlab.software.imdea.org/zistvan-events/fabric-example-supplychain
```

4) In the tutorial folder, run `./startFabric.sh`. This will start up Fabric and install smart contracts, etc., in the background and exit when everything has been started. **This can take up to 5-10 minutes on the first run!**

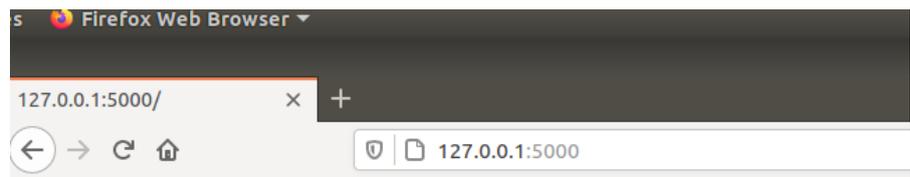
Once finished, you should see something similar to:

```
vmuser@vmachine: ~/fabric-example-supplychain  
File Edit View Search Terminal Help  
CXX(target) Release/obj.target/pkcs11/src/pkcs11/pkcs11.o  
CXX(target) Release/obj.target/pkcs11/src/async.o  
CXX(target) Release/obj.target/pkcs11/src/node.o  
SOLINK_MODULE(target) Release/obj.target/pkcs11.node  
COPY Release/pkcs11.node  
make: Leaving directory '/home/vmuser/fabric-example-supplychain/supplychain/typescript/node_modules/pkcs11js/build'  
> grpc@1.24.2 install /home/vmuser/fabric-example-supplychain/supplychain/typescript/node_modules/grpc  
> node-pre-gyp install --fallback-to-build --library=static_library  
node-pre-gyp WARN Using request for node-pre-gyp https download  
[grpc] Success: "/home/vmuser/fabric-example-supplychain/supplychain/typescript/node_modules/grpc/src/node/extension_binary/node-v72-linux-x64-glibc/grpc_node.node" is  
installed via remote  
npm notice created a lockfile as package-lock.json. You should commit this file.  
npm WARN supplychain@1.0.0 No repository field.  
added 446 packages from 1103 contributors and audited 446 packages in 65.014s  
5 packages are looking for funding  
  run 'npm fund' for details  
found 27 vulnerabilities (20 low, 1 moderate, 5 high, 1 critical)  
  run 'npm audit fix' to fix them, or 'npm audit' for details  
> supplychain@1.0.0 build /home/vmuser/fabric-example-supplychain/supplychain/typescript  
> tsc  
wallet path: /home/vmuser/fabric-example-supplychain/supplychain/wallet  
Successfully enrolled admin user "admin" and imported it into the wallet  
wallet path: /home/vmuser/fabric-example-supplychain/supplychain/wallet  
Successfully registered and enrolled admin user "user1" and imported it into the wallet  
~/fabric-example-supplychain/first-network  
cat <<EOF  
Total setup execution time : 234 secs ...  
vmuser@vmachine:~/fabric-example-supplychain$
```

5) In the tutorial folder, run `./runWebApp.sh`. This will launch a web server that connects to the blockchain. Do not run this command in the background (&) because it will have to be stopped manually before we tear down the blockchain instance. Once started it will output:

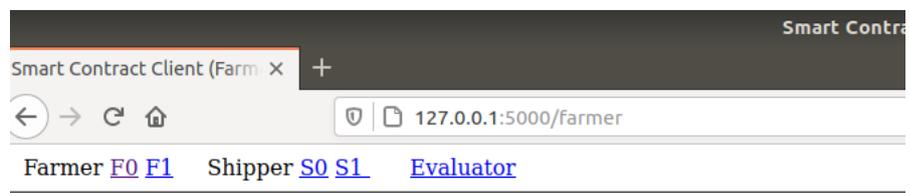
```
vmuser@vmachine:~/fabric-example-supplychain$ ./runWebApp.sh
* Serving Flask app "client.py"
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
CWD: /home/vmuser/fabric-example-supplychain/supplychain
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

6) Open Firefox and navigate to the address and port provided in the script output. You should see:



Please go to [/farmer](#), [/shipper](#) or [/evaluator](#)

Click on Farmer and if you see the following screen, the VM is correct and you shall have no issues following the tutorial!



## Supplychain - Farmer #0



### My Item Infos:

I have no items at the moment

### Add info on good

Item Tag (e.g. Cabbage)  0  To Shipper #0

### IMPORTANT LAST STEP

7) Once you are done, stop the web server (CTRL+C in the terminal window) and execute `./tearDownAll.sh`. This will stop Fabric and clean up docker images, etc. It can take several minutes to finish and it is important to run it *before* starting the blockchain again.