**Guide to Deploy to Server**

(Ubuntu 16.4)

**1. Install Packages**

To do this, we can install the build-essential meta-package from the Ubuntu repositories. We will also be downloading the tcl package, which we can use to test our binaries.

We can update our local apt package cache and install the dependencies by typing:

$ sudo apt-get update

$ sudo apt-get install build-essential

* Install NodeJS:

$ sudo apt-get install nodejs

$ sudo apt-get install npm

$ sudo apt-get install nodejs-legacy

* Install GIT

$ sudo apt-get install git

* Install MySQL

$ sudo apt-get install mysql-server

* Install Redis
  + Finally, we need to download tcl:
  + Download the latest stable release tarball from Redis.io.

$ wget http://download.redis.io/releases/redis-stable.tar.gz

* Unpack the tarball by typing:

$ tar xzf redis-stable.tar.gz

* Move into the Redis source directory structure that was just extracted:

$ cd redis-stable

* Now, we can compile the Redis binaries by typing:

$ make

* After the binaries are compiled, run the test suite to make sure everything was built correctly. You can do this by typing:

$ make test

* This will typically take a few minutes to run. Once it is complete, you can install the binaries onto the system by typing:

$ sudo make install

* To access the script move into the utils directory:

$ cd utils

* From there, run the Ubuntu/Debian install script:

$ sudo ./install\_server.sh

$ sudo apt install redis-server

$ redis-server

* Install PM2

sudo npm install pm2 -g

**2. Run Project On Server**

* Create DB (Example: for user "*ubuntu*")

mysql -u root -p -h localhost

mysql> CREATE USER 'ubuntu'@'localhost' IDENTIFIED BY 'vsiics';

mysql> CREATE DATABASE CurrencySwap;

mysql> GRANT ALL PRIVILEGES ON CurrencySwap.\* TO 'ubuntu'@'localhost';

mysql> quit

* Checkout Project (from develope branch)

eval `ssh-agent -s`

chmod 400 ~/.ssh/id\_rsa

ssh-add .ssh/id\_rsa

git clone git@github.com:currencyswap/currencyswap.git

git checkout develop (source code demo tren branch develop)

git pull origin develop

* Create log folder:

EX:

cd ~ (go to ubuntu folder)  
mkdir logs (in path: /home*/ubuntu/*)

* Goto *src* folder (Ex: cd CurrencySwap/src), duplicate *src/appconfig.json.template* and rename to *appconfig.json*

Change config, examle:

|  |
| --- |
| {  **"title"** : **"Currency Swap"**,  **"redis"**: {  **"host"** : **"localhost"**,  **"port"** : 6379,  **"db"** : 10,  **"ttl"** : 3600,  **"disableTTL"**: **false**,  **"prefix"** : **"cs\_app:"** },  **"smtp"** : {  **"host"** : **"smtp.gmail.com"**,  **"port"**: 587,  **"secure"** : **true**,  **"auth"** : {  **"user"**: **"devcurrencyswap@gmail.com"**,  **"pass"**: **"CurrencySwap123"** }  },  **"mailSender"**: {  **"sender"**: **"devcurrencyswap@gmail.com"**,  **"subject"**: **"[Currency Swap] Your password has been reset"** },  **"paging"** : {  **"activityLogs"**: 4  },  **"dateFormat"** : {  **"datetime"**: **"dd-mm-yy HH:MM"**,  **"date"**: **"mmm dd,yyyy"**,  **"time"**: **"HH:MM:ss"** },  **"media"**: {  },  **"logs"** : **"/home/ubuntu/logs"**, #path to log folder   **"tokenExpired"** : **"7 d"** } |

* Duplicate src/server/datasource.json.template and rename to datasource.json

Example:

|  |
| --- |
| {  **"NodeJSApp"**: {  **"host"**: **"localhost"**,  **"port"**: 3306,  **"url"**: **""**,  **"database"**: **"CurrencySwap"**,  **"password"**: **"vsiics"**,  **"name"**: **"NodeJSApp"**,  **"user"**: **"ubuntu"**,  **"connector"**: **"mysql"** } } |

* Run $sudo npm install -g bower
* Run $npm install
* Run $node ./bin/setup.js
* Run $node ./server/server.js

Configuration PM2:

* Create symbolic link (link from "src" folder)

cd /opt/

mkdir app

sudo ln -s "/home/ubuntu/currencyswap/src/" /opt/app/CS

* Startup PM2

sudo pm2 startup

* Run Project from Symbolic link and build-in Clustering

cd /opt/app/CS

sudo pm2 start server/server.js -i 10

* Save config pm2

sudo pm2 save