*Florida International University*

*School of Computing and Information Sciences*

INSTALLATION GUIDE

Note this is also part of the “Installation/Maintenance Document” appendix in the Final Document

Note that a video of this process is also available on youtube in the project playlist

**Team Member:**

Jose Ponce

Dafna Steinberg

**Product Owner(s)**:

Masoud Sadjadi

Fransisco Ortega

**Mentor(s)**:

Leila Zahedi

**Instructor**:

Masoud Sadjadi

# 

**Installation Guide:**

1. **Development Environment Installation:**
   1. Pre-requisites:
      1. NodeJS(and npm) installed: Any version after the one mentioned in the “Hardware and Software Resources” section would be fine. <https://nodejs.org/en/>
      2. MongoDB: You will need a MongoDB server to run this application, if you don’t have one, you can install and run it on your own development laptop/desktop, no special configuration is required. Any version after the one mentioned in the “Hardware and Software Resources” section would be fine. <https://www.mongodb.com/download-center#community>
      3. Git: <https://git-scm.com/downloads>
      4. An IDE is recommended: any IDE that can deal with NodeJS/HTML/CSS should do.
   2. Execute:
      1. Checkout the git repository for the project, if you don’t know where it’s located just ask your Product Manager. From here on i will refer to the folder where the code was checked out from git as {{gitCheckoutDir}}.
      2. “Install” the application by running “{{gitCheckoutDir}}/Code/install.sh” for linux/MAC or “{{gitCheckoutDir}}/Code/install.bat” for windows. Note this only does an npm install in the “{{gitCheckoutDir}}/Code/” and “{{gitCheckoutDir}}/Code/Deployment/” folders. If you don’t understand what that means, you may want to research it. If this script fails, then there is a problem with your npm install.
      3. Configure the application to run locally:
         1. Make sure the host and protocol are correct in the “{{gitCheckoutDir}}/Code/server.js” file. You normally want that file to have two lines like this near the beginning of the file:

app.set("host", "localhost");

app.set("protocol", "http");

* + - 1. Make sure the “{{gitCheckoutDir}}/Code/config/config file looks like this:

module.exports = {

'port':3000,

'externalPort':3000,

'database':'mongodb://localhost:27017/admin',

'secret':'whateveryouwant',

'secure':false

};

Adjust the mongo url if your mongoDB is not on your development machine.

* + 1. Run the application by executing “node {{gitCheckoutDir}}/Code/server.js” into your terminal. If this fails, then there is a problem with your nodejs installation.
    2. You can now access the website at “<http://localhost:3000>”
    3. You can create a PI account with sadjadi’s email for testing purposes on your local environment since that account requires no verification.

\*Note that google+ login(student login) will not work in your local environment.

1. **Production Installation(this also includes the vip-dev server):**

\*\* Note that it is very important that the deployment folder stays the same(/var/www/VIP-FINAL/) so that the certificate renewal process works automatically.

* 1. Stop the web server:
     1. “sudo forever stopall” should work, if it doesn’t then execute “sudo kill $(pidof nodejs)” to force close/kill the process.
  2. Copy the Code folder to the “/var/www/VIP-FINAL/” folder into the server(vip-dev or production):
     1. Note it has to be this folder and not elsewhere, nor can this folder be renamed because there is a cron job in the server that will make sure the certificates are put there whenever they are renewed and this process would be broken/useless if you deploy elsewhere/rename the folder.
     2. Note that if any changes to the certificateRenewal.bash file are to be really used, this file needs to be copied to the “/var/www/” folder and you have to execute “sudo chmod 711 /var/www/certificateRenewal.bash” afterwards to make the script executable.
  3. Https/ssl certificate setup (copy certificates to the deployment folder):
     1. Run

sudo cp /etc/letsencrypt/live/vip-dev.cis.fiu.edu/cert.pem /var/www/VIP-FINAL/Code/

* + 1. Run

sudo cp /etc/letsencrypt/live/vip-dev.cis.fiu.edu/chain.pem /var/www/VIP-FINAL/Code/

* + 1. Run

sudo cp /etc/letsencrypt/live/vip-dev.cis.fiu.edu/privkey.pem /var/ww w/VIP-FINAL/Code/

Note that in all of the above commands you need to change “vip-dev.cis.fiu.edu” by “vip.fiu.edu” when in the production server.

* 1. Configure /var/www/VIP-FINAL/Code/api/config/config:
     1. Make sure /var/www/VIP-FINAL/Code/api/config/config looks like this(use your editor of choice, example, nano or emacs):

module.exports = {

'port':3000,

'externalPort':443,

'database':'mongodb://localhost:27017/admin',

'secret':'whateveryouwant',

'secure':true

};

Note this is different from the configuration on your local environment.

* 1. Configure /var/www/VIP-FINAL/Code/api/config/auth.js:
     1. Make sure /var/www/VIP-FINAL/Code/api/config/auth.js looks like this:

module.exports ={

‘googleAuth': {

'clientID' : '1056710173783-00k7c8he5utpi75h8jtn3183cns3suq1.apps.googleusercontent.com',

'clientSecret': 'gqwU3jy3K-anElJ6Vf3j7Py6',

'callbackURL': 'http://vip-dev.cis.fiu.edu/auth/google/callback'

}

};

Note that in production you want vip-dev.cis.fiu.edu to be replaced by

vip.fiu.edu in the callbackURL.

* 1. Configure /var/www/VIP-FINAL/Code/server.js:
     1. Make sure that the host and protocol are correctly set in the /var/www/VIP-FINAL/Code/server.js file. There should be two lines near the top of the file that start with app.set("host" and app.set("protocol" , make sure they say app.set("host", "vip-dev.cis.fiu.edu"); and app.set("protocol", "https"); respectively.

Note that when deploying to production(and not vip-dev) you want the host to be “vip.fiu.edu” instead.

* 1. Start the web server with forever:
     1. Run sudo forever start /var/www/VIP-FINAL/Code/server.js
  2. Verify server is running:
     1. Perform at least a student login through google+ and a regular login just to ensure the server is correctly configured and running.