*Florida International University*

*School of Computing and Information Sciences*

CIS 4911 - Senior Capstone Project

Software Engineering Focus

Installation Guide

Addigy4

Team # X

**Team Members**

Carlos Ruiz

David Romero

**Product Owner**: Jason Dettbarn

**Instructor:** Masoud Sadjadi

1. **Server** installation:
   1. Make sure server has python environment running. Target version python2.7.
   2. Download project zip file:
      1. Open terminal or from ssh session type: $ *wget https://github.com/FIU-SCIS-Senior-Projects/Addigy4/archive/master.zip*
      2. unzip master.zip file by typing: $ *unzip master.zip*
      3. navigate to Server software directory: $ cd Addigy4-master/Code/tunneler/server/
      4. Change setup.sh permission by: $ chmod +x setup.sh
      5. then run: $ ./setup.sh
      6. Thats it! Server shall be up and running.
2. **Client/Tunneler** installation:
   1. Make sure computer has python installed. Target python version is python2.7.
   2. Download project zip file:
      1. Open terminal or from ssh session type: $ *wget https://github.com/FIU-SCIS-Senior-Projects/Addigy4/archive/master.zip*
      2. unzip master.zip file by typing: $ *unzip master.zip*
      3. navigate to Server software directory: $ cd Addigy4-master/Code/tunneler/
      4. change permission of setup.sh files for **Client/Tunneler:** $ chmod +x client/setup.sh and then $ chmod +x tunneler/setup.sh
      5. run both directories setup.sh files: $ ./client/setup.sh and $ ./tunneler/setup.sh
      6. Now both directories are going to be located on /var/opt. To find them open terminate and type: $ cd /var/opt

Running **Client**/**Tunneler:**

1. After installation is completed, navigate to /var/opt.
   1. Open terminal for each program.
   2. Type: $ cd /var/opt
   3. **Client** and **Tunnel** boths will have to be updated:
      1. open their directory and navigate to "ServerObject.py"
      2. find "**def \_\_init\_\_(self):**" and update "**self.\_\_url**" with the url where the Server is going to be running
      3. save and exit
2. Running Tunneler
   1. Type: $ python tunneler/Main.py <tunnelId>
      * tunnelId: needs to be 36 byte id and unique
      * sample: **(fc86c7ef-f579-4115-8137-289b8a257803)**
3. Running Client
   1. Type: $ python client/Main.py <tunnelId><local port><tunnel port>
      * tunnelId: id for the tunnel you wish to communicate with
        + sample: **(fc86c7ef-f579-4115-8137-289b8a257803 3000** **22)**
      * local port: local port in which Client software is going to be listening
      * tunnel port: port number to which the Tunneler software is going to send the request