Lab Manual

CPE 322: Engineering Design VI for Electrical and Computer Engineers

Lab 1 — GHDL and GTKWave

- Go to the GitHub repository of Digital System Design (DSD)
  - Study VHDL and GHDL
- Go to the GHDL folder
  - Install GHDL and GTKWave
  - Run the Half Adder example
  - Run another example such as D Flip-Flop or 4-to-1 Multiplexer
  - Document the results on your GitHub repository
- Exploration: Icarus Verilog

Lab 2 — Command Line

- Go to the IoT repository
- Study Lessons 1 and 2
- Open a terminal

```
$ hostname
$ env
$ ps
$ pwd
$ git clone
https://github.com/kevin
wlu/iot.git
$ cd iot
$ ls
$ cd
```

```
$ df
$ mkdir demo
$ cd demo
$ nano file
$ cat file
$ cp file file1
$ mv file file2
$ rm file2
$ clear
$ man uname
$ uname -a
$ ifconfig
$ ping localhost
$ netstat
```

Lab 3 — Python

- Study the GitHub repository Lesson 3 labs
- Install required Python packages such as jdcal, astral, and geopy

```
$ cd ~/iot
$ cd *3
$ python3 julian.py
$ python3 date_example.py
$ python3 datetime_example.py
$ python3 time_example.py
$ python3 sun.py 'New York'
$ python3 moon.py
$ python3 coordinates.py 'SC Williams Library'
$ python3 address.py '40.74480675, -74.02532862031404'
$ python3 cpu.py
$ python3 battery.py
$ python3 documentstats.py document.txt
```

Lab 4 — Django and Flask

- Study the GitHub repository Lesson 4 labs
- Install Django and Django REST framework
- Use the default database, i.e., SQLite
- Start Django project "stevens," run server, and view app
- Start Django REST project "mycpu," run server, and view app
- Install Flask if no module named 'flask'
- Run Flask server via hello_world.py and view app

Lab 5 — Paho-MQTT

- Study the GitHub repository Lesson 5 labs
- Install Paho-MQTT
- Change directory to the iot repository
- Update the repository with git pull
- Change directory to Lesson 5
- Run python3 subcpu.py on one Terminal
- Run python3 pubcpu.py on another

Lab 6 — Node.js and Pystache

- Study the GitHub repository Lesson 6
- Install Node.js and run hello-world.js, hello.js, and http.js
  - Refresh the webpage to see server activities
- Install Pystache and run say_hello.py that uses the template in say_hello.mustache

Lab 7 — ThingSpeak and Google Sheets

- Study the GitHub repository Lesson 7
- Sign up and log in MathWorks ThingSpeak
- Run thingspeak_cpu_loop.py or thinkspeak_feed.py in a demo folder
- Install gspread and oauth2client
- Log in the Google Cloud Platform Identity and Access Management, create a project cpudata, enable both Drive API and Sheets API, create and download service account JSON key file
- Start a new Google sheet cpudata, share it with the client email in the JSON file, delete Rows 2 to 1000, and edit the header cells
- Run cpu_spreadsheet.py with the JSON key file in a demo folder

Lab 8 — Data Analysis

- Study the GitHub repository Lesson 8
- Install Python packages
- Save the Lab 7 Google sheet in CSV format to ~/demo
- Copy ~/iot/lesson8/plt_final.py and plt_cv2.py to ~/demo
- Edit plt_final.py and plt_cv2.py to read the CSV file with customized plot titles
- Run plt_final.py and plt_cv2.py

Lab 9 — YANG

- Study the GitHub repository Lesson 9
- Install pyang and PlantUML
- copy ~/iot/lesson9/intrusiondetection.yang to ~/demo
- Run pyang to generate intrusiondetection.yin and intrusiondetection.uml
- Run PlantUML to generate intrusiondetection.png

Lab 10 — Blockchain

- Study the GitHub repository Lesson 10
- Run hash_value.py twice and compare results
- Run snakecoin.py
- Run snakecoin-server-full-code.py on Terminal 1 and mine a new block on Terminal 2
- Clone Python blockchain app and uncomment the last line of node_server.py
- Run node_server.py on Terminal 1 and run_app.py on Terminal 2

Senior Design Projects

- Bookmark and review the following pages:
  - ECE senior design site, Google sheet and TABER form
  - Senior design roadmap
  - Interdisciplinary project rubric
  - Stevens MakerCenter
  - Environmental Health and Safety (EHS)
  - Stevens Annual Innovation Expo
- Stevens Ducks app > find and open the Innovation Expo Guide
- Download a collaboration app, e.g., Asana, Discord, Flock, Lucidspark, Slack, and Trello