Operating System Simulator (CMSC 312)

Simulator built in Python using Tkinter for GUI. Also use some bare-bones JavaScript, CSS, SCSS, and HTML to randomly generate program files.

How to run the Operating System Simulator

You need to create a virtual environment to run this operating system simulator
If you have not install python then you need to download latest version of the language.
This is a guideline how to run the operating system on Mac

Step 1: Go to terminal. Check if the latest version of the language is install: `python3 — version`
Step 2: Install pip: `sudo easy_install pip`
Step 3: Install virtualenv for python: `sudo pip install virtualenv`
Step 4: Create a virtual environment:
  - Navigate to Desktop
  - Type: virtualenv <choose a name for your virtualenv> (Ex: Doraemon)
Step 5: Move the project folder (CMSC312-OS-Simulator) to the virtualenv you just created
Step 6: In terminal, navigate to Doraemon
Step 7: Activate the environment : `source bin/activate`
  - To check if you are in the virtual environment, you will see `(Doraemon)` at the beginning
Step 8: Install the latest official version of Django: `sudo pip install django==3.0.1`
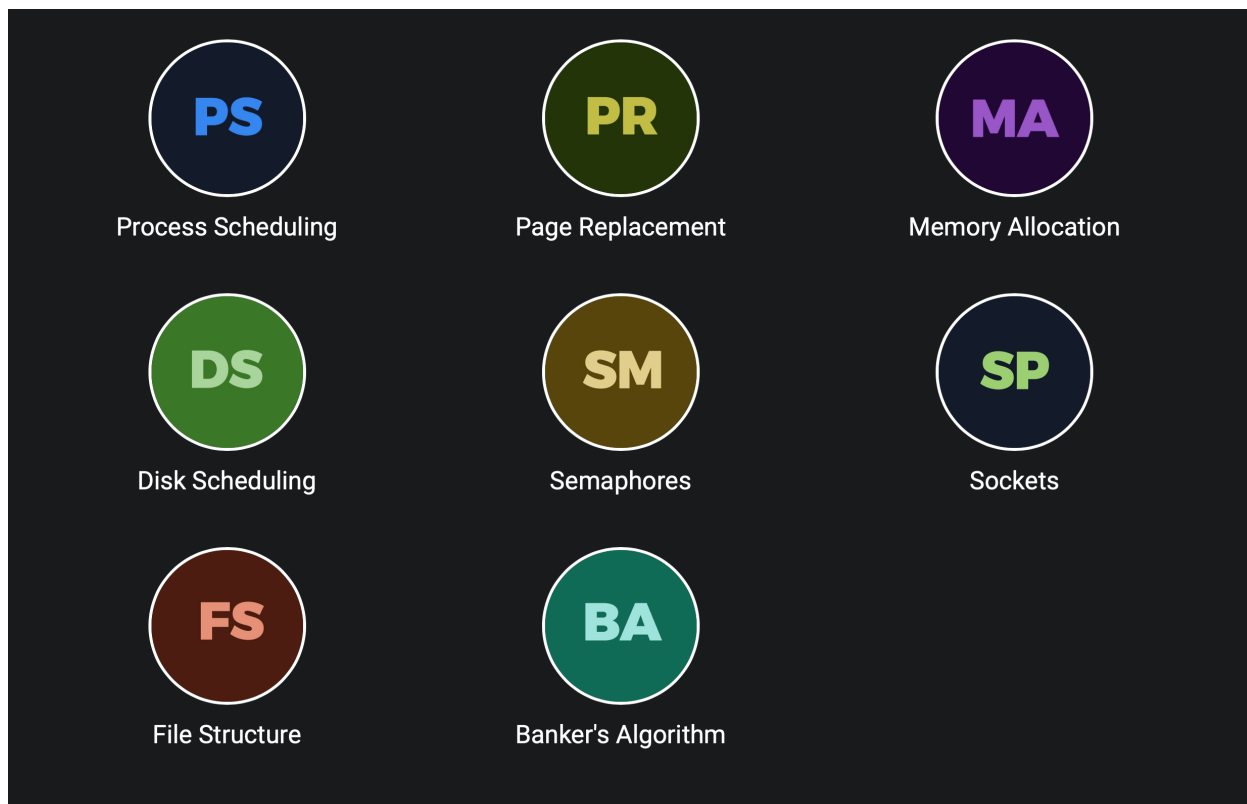**Step 9:** Navigate to OS-Simulator/ossim
Step 10: Start the simulator: `python manage.py runserver`
Step 11: Copy the server address **http://127.0.0.1:8000** and run it in a browser

```
System check identified 10 issues (0 silenced).

You have 5 unapplied migration(s). Your project may not work properly until you apply
the migrations for app(s): admin, auth.
Run 'python manage.py migrate' to apply them.
October 31, 2021 — 13:40:07
Django version 3.2.8, using settings 'ossim.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
[31/Oct/2021 13:40:23] "GET / HTTP/1.1" 200 5560
[31/Oct/2021 13:40:23] "GET /static/ossim/css/pluton.css HTTP/1.1" 200 3741
[31/Oct/2021 13:40:23] "GET /static/ossim/css/jquery.bxslider.css HTTP/1.1" 200 3892
[31/Oct/2021 13:40:23] "GET /static/ossim/css/style.css HTTP/1.1" 200 16433
[31/Oct/2021 13:40:23] "GET /static/ossim/css/bootstrap—responsive.css HTTP/1.1" 200 2
3211
```

PS

Process Scheduling

PR

Page Replacement

MA

Memory Allocation

DS

Disk Scheduling

SM

Semaphores

SP

Sockets

FS

File Structure

BA

Banker's Algorithm

** Only Process Scheduling is functioning correctly at the current moment

Scheduler(s) that are implemented :
1. SJF
2. SRTF
3. Priority scheduling

I have implemented the above scheduler algorithm. User(s) can add process(es) and entering arrival time(s) and burst time(s). Depends on the unique arrival time(s) and burst time(s) of the processes, the process with smaller least arrival time will start execution first. In case the first program still have not completed its execution units yet, it will continue process the next process that arrives will be sent to queue.

GUI
I have done GUI for the simulator; however, I might change something so it is just demo at the current moment. For every time the scheduler(s) being implemented, the simulator will show the states of process(es) in "Operations Taking Place", table of process time(s) as well as a Gantt chart.

Git Repository
https://github.com/Huyenhuynh81/CMSC312-OS-Simulator.git