*In this exercise you will be required to demonstrate the ability to setup a webserver, design a webpage/app and write BASH scripts. Please email your solution to faisal-recruitment@sfu.ca with the subject **[COOP interview] Programming Exercise**. If you are selected for an on-site interview, you would be requested to demo your solution during the on-site interview.*

A powerful computer server is available at our disposal. We want to setup a processing pipeline that would accept a list of jobs (computing tasks) from multiple users and run these jobs on the compute server and email the results back to the users. The two main desired features of the processing pipeline are:

1. Webpage for getting the job lists from the users along with their personal information.
2. A **scheduler** script written in BASH, that submits the jobs to p processing threads, where only ***k*** serial jobs per thread are allowed. The scheduler script should be run periodically using the **cron** daemon and check if there is vacancy on the processing threads and submit jobs to accordingly. The scheduler should behave "optimally" in the sense that all the ***p*** processing threads must be utilized always, unless of course the total number of jobs is less than ***p***.

You are required to implement the above processing pipeline and successfully demonstrate its working under the following conditions:

- Multiple users simultaneously submitting job lists via the web interface.
- Job lists of different lengths.
- Jobs with varying run-times.
- A reasonable choice for ***p*** and ***k***.

NOTE: You are free to choose the "test" jobs on the job lists as you like. The only restriction is that the "test" jobs cannot expect any further user input and they should be able to run readily on the compute server without any dependencies.