Simulation of Operating System (Process Management):

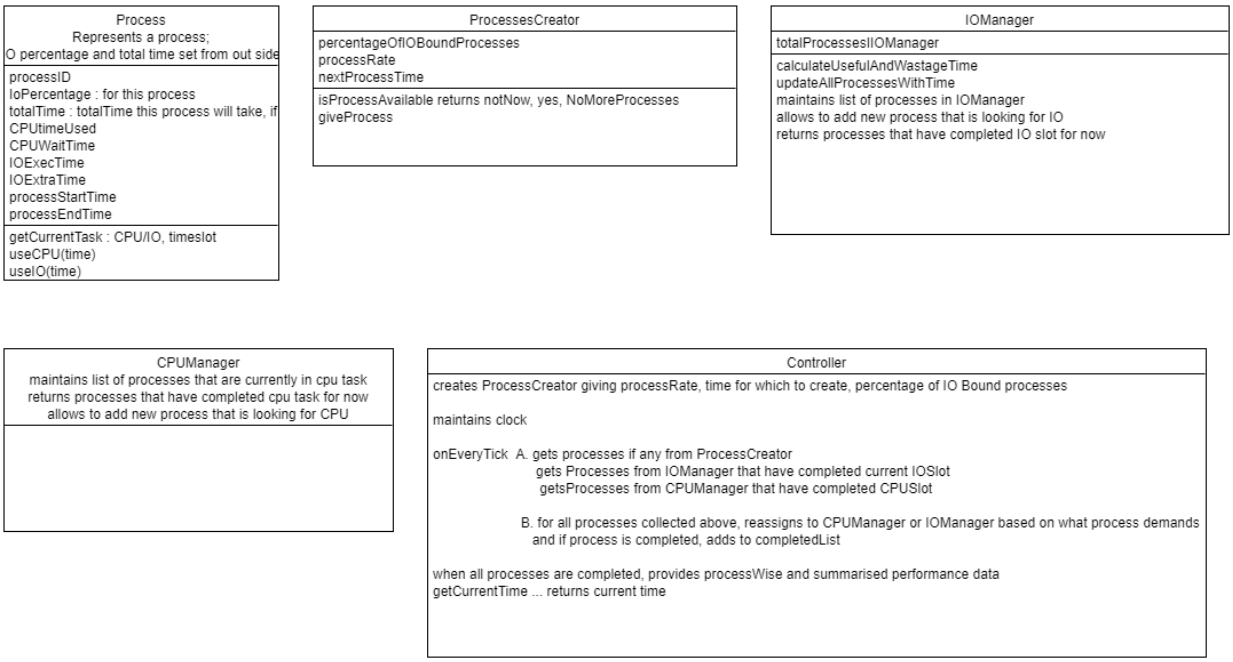
Simulation of processes in shared computer system, to check performance of the system against sequential execution of processes.
Each process is CPU-bound or IO-bound (spending 80% time in -bound activity)
CPU can be used by only one process at a time.
IO can be shared by any number of processes, but getting slower by 5% for every additional user.
Maximum CPU slot of 100 ms.

Simulate above situation so that we can check performance for:
Approximately 20 processes every minute, each needing execution time of approx 3 to 8 seconds if run alone.

and check performance for following percentage of IO-bound processes (0,20,40,60,80,100).
Run simulation for processes getting generated for 5 minutes, and data collection till all processes are completed.

==========

Process Creator
CPU Manager
IO Manager
Controller and Result provider


Process
Represents a process;
O percentage and total time set from out side
processID
IoPercentage : for this process
totalTime : totalTime this process will take, if
CPUtimeUsed
CPUWaitTime
IOExecTime
IOExtraTime
processStartTime
processEndTime
getCurrentTask : CPU/IO, timeslot
useCPU(time)
useIO(time)
ProcessesCreator
percentageOfIOBoundProcesses
processRate
nextProcessTime
isProcessAvailable returns notNow, yes, NoMoreProcesses
giveProcess
IOManager
totalProcessesIIOManager
calculateUsefulAndWastageTime
updateAllProcessesWithTime
maintains list of processes in IOManager
allows to add new process that is looking for IO
returns processes that have completed IO slot for now
CPUManager
maintains list of processes that are currently in cpu task
returns processes that have completed cpu task for now
allows to add new process that is looking for CPU
Controller
creates ProcessCreator giving processRate, time for which to create, percentage of IO Bound processes
maintains clock
onEveryTick A. gets processes if any from ProcessCreator
gets Processes from IOManager that have completed current IOSlot
getsProcesses from CPUManager that have completed CPUSlot
B. for all processes collected above, reassigns to CPUManager or IOManager based on what process demands
and if process is completed, adds to completedList
when all processes are completed, provides processWise and summarised performance data
getCurrentTime ... returns current time