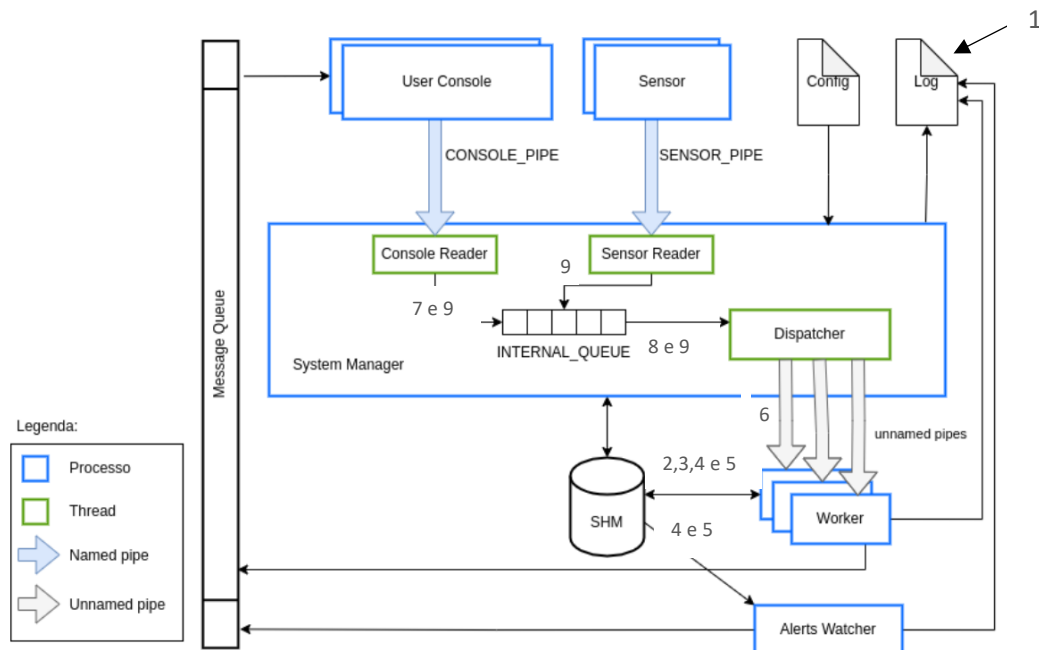


Projeto de SO – Entrega Final



Legenda:

Semáforos:

1. “log_sem” – Controla todos os acessos (“Worker”, “Alert Watcher” e “System Manager”) ao “log”, evitando corrupção na escrita para o ficheiro “Log”;
2. “shm_keys_sem” – Controla o acesso ao segmento da “shared memory” que contem a informação sobre as “keys”;
3. “shm_sensors_sem” – Controla o acesso ao segmento da “shared memory” que contem a informação sobre os sensores que comunicaram com o sistema;
4. “shm_alerts_sem” – Controla o acesso ao segmento da “shared memory” que contem a informação sobre os alertas adicionados ao sistema;
5. “shm_watch_alerts_sem” – Sinaliza ao “Alerts Watcher” para verificar se os valores das “keys” despoletam algum alerta. Abre sempre que ocorre uma mudança nos valores das “keys”;
6. “workers_available_sem” – O valor do semáforo corresponde ao número de “workers” disponíveis. Quando um “worker” começa a trabalhar faz “sem_wait()” e quando termina faz “sem_post()”;

Variáveis de Condição:

7. “queue_full” – Sinaliza quando a “Internal Queue” deixa de estar cheia. Assim a “thread console reader” pode adicionar elementos à fila;
8. “queue_empty” – Sinaliza quando a “Internal Queue” deixa de estar vazia. Assim a “thread dispatcher” pode tirar um elemento da fila;

Mutex:

9. “queue_mutex” – Providencia exclusão mútua nas operações sobre a “Internal Queue”. As variáveis de condição “queue_full” e “queue_empty” só são usadas com este “mutex” fechado.