National University of Computer & Emerging Sciences
Karachi Campus

Distributed task queue system using socket programming

Project Proposal
Computer Networks [CN]
Section: H

Group Members:
21k-3261 Muhammad Umer Tariq
21k-4933 Ibrahim Jawaid
21k-4878 Sarosh Irfan

Introduction And Project Description

We are developing a distribution task queue, complete with producers and consumers, along with a controller to manage the flow of tasks between them. These components will be interconnected through a network, facilitating seamless communication and task distribution.
To achieve this, we will utilize Socket Programming. In our system, producers will send tasks to the controller using sockets, and similarly, the controller will distribute these tasks to consumers through socket communication.
Our system will incorporate various features to ensure efficiency and reliability. This includes functionalities of a load balancer, which evenly distributes tasks among consumers, as well as mechanisms for concurrency control to manage simultaneous task execution. Additionally, we will implement fault tolerance measures and robust error handling to maintain system stability.

Salient Features

1. **Providing Fault tolerance**
2. **Load balancing for optimal resource utilization**
3. **Concurrency control**
4. **Priority Queue:** Implement a priority-based task queue where tasks with higher priorities are processed before lower-priority task
5. **Security Measures:** Enhance security by implementing encryption mechanisms

Tools & Technologies

- **Language: Python**
- **IDE: VS Code/PyCharm**
- **Modules:**
  1. **Socket**: For creating sockets and exchanging tasks among processes.
  2. **Cryptography**: For performing encryption/decryption of messages.
  3. **Flask**: For implementing authentication.
  4. **Celery**: For task management, distribution, and scheduling.
  5. **Queue**: For implementing the task queue.
  6. **Threading**: To create multiple processes and ensure concurrency control between them