KIT318 Assignment

group assignment

2019

**Table of contents**

Yinzi xie(445891)

Jing liu(467849)

Yuyang peng(427324)

Yuhang yang(435737)

**[Architecture](#_Toc27443_WPSOffice_Level1)** **[1](#_Toc27443_WPSOffice_Level1)**

**[Design Features](#_Toc32102_WPSOffice_Level1)** **[2](#_Toc32102_WPSOffice_Level1)**

**[Class Diagram](#_Toc1289_WPSOffice_Level1)** **[3](#_Toc1289_WPSOffice_Level1)**

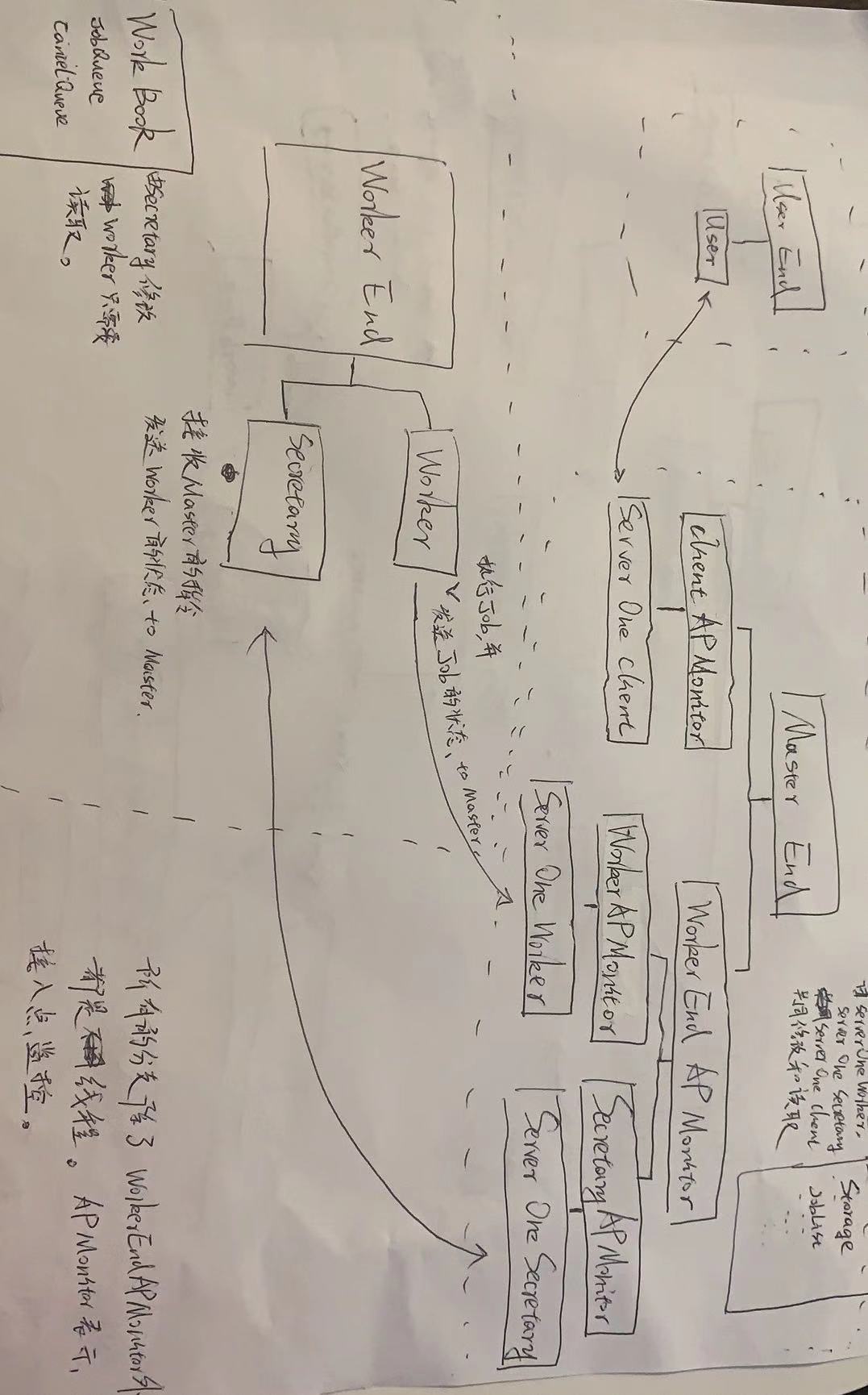
**[Sequence Diagram](#_Toc10039_WPSOffice_Level1)** **[4](#_Toc10039_WPSOffice_Level1)**

**[Evaluation](#_Toc17309_WPSOffice_Level1)** **[5](#_Toc17309_WPSOffice_Level1)**

**[Each member implementation 1](#_Toc6925_WPSOffice_Level1)4**

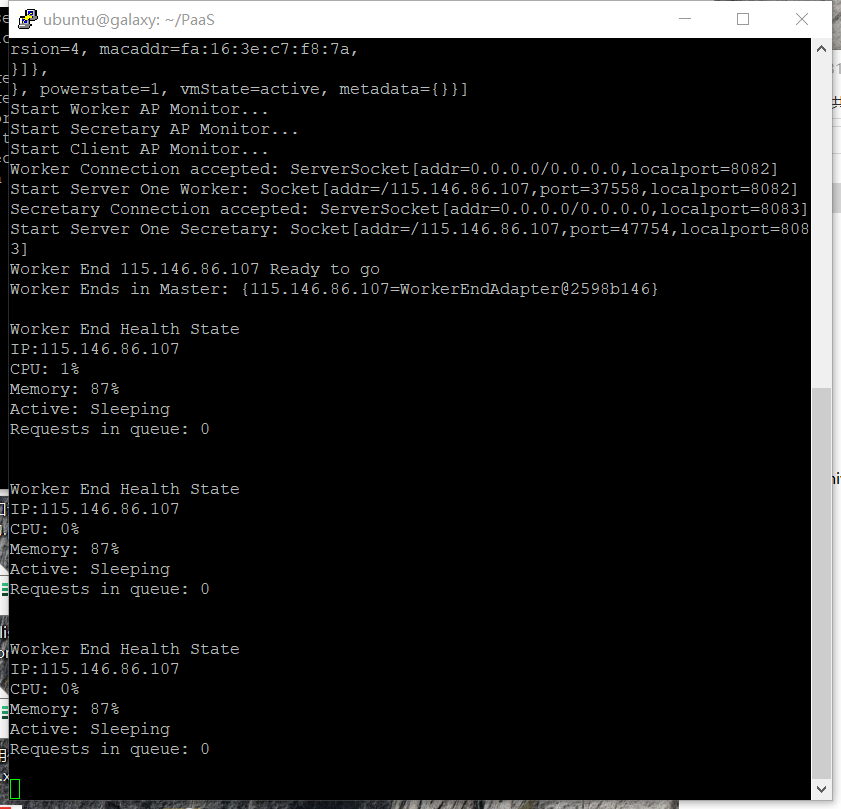
# Architecture

This is our Architecture，This from XIE and Peng’s Designed



z

This is the detection of CPU and Memory when the worker just started.



# Design Features

1 .The program is executed through the process of User-> master-> worker-> master-> user.

Users and the Master communicates through the TCP protocol. When users send a request to the master, the master assigns tasks to available workers. The task allocation is based on a first come first serve mechanism and is randomly assigned to users when there isn’t any available user. When the workload becomes overloaded, the master would start a new worker to help execute the tasks. The creation of a new worker is based on the status of each worker. When workers have more than two jobs in the queue, they are regarded as busy. When all workers are regarded as busy, and there are new tasks coming, a new worker is created for a quicker execution.

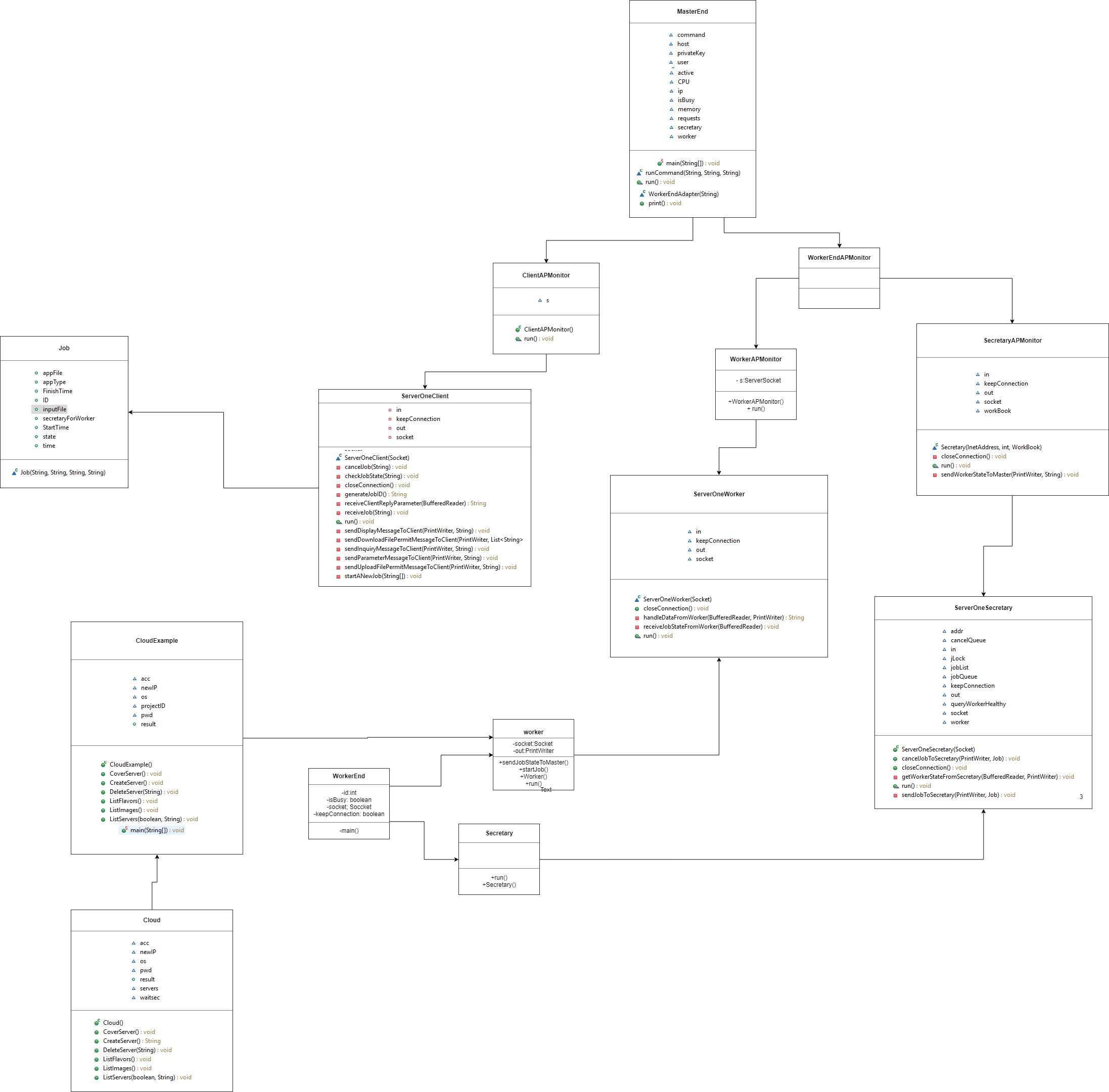
2 . Workers and the master use socket as a TCP protocol to communicate. By using uploading and downloading as a means between workers and the master, they can transfer files.

3.Master can initialize, and stop a worker and check status of a worker.

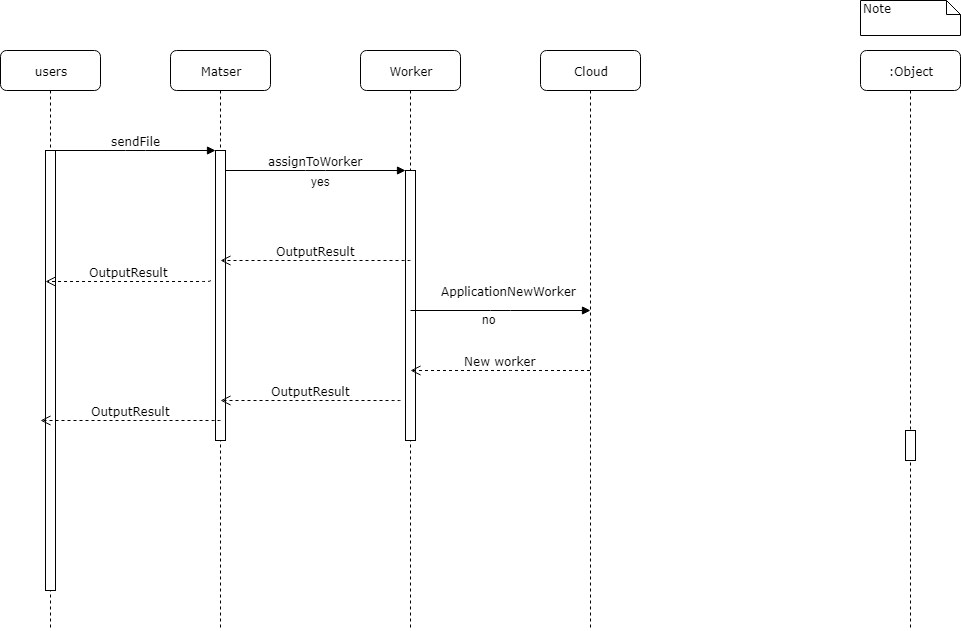
4.When a worker dies, master reallocates the tasks, if other workers are busy, then master randomly assign tasks to a worker. Else the master assigns the task to an available worker.

5.Elasticity FCFS mechanism:The program's system process scheduling algorithm and a network route management mechanism automatically execute queued requests and processes based on the order of arrival. First come, first served, first come first served. The next request will be executed before completion

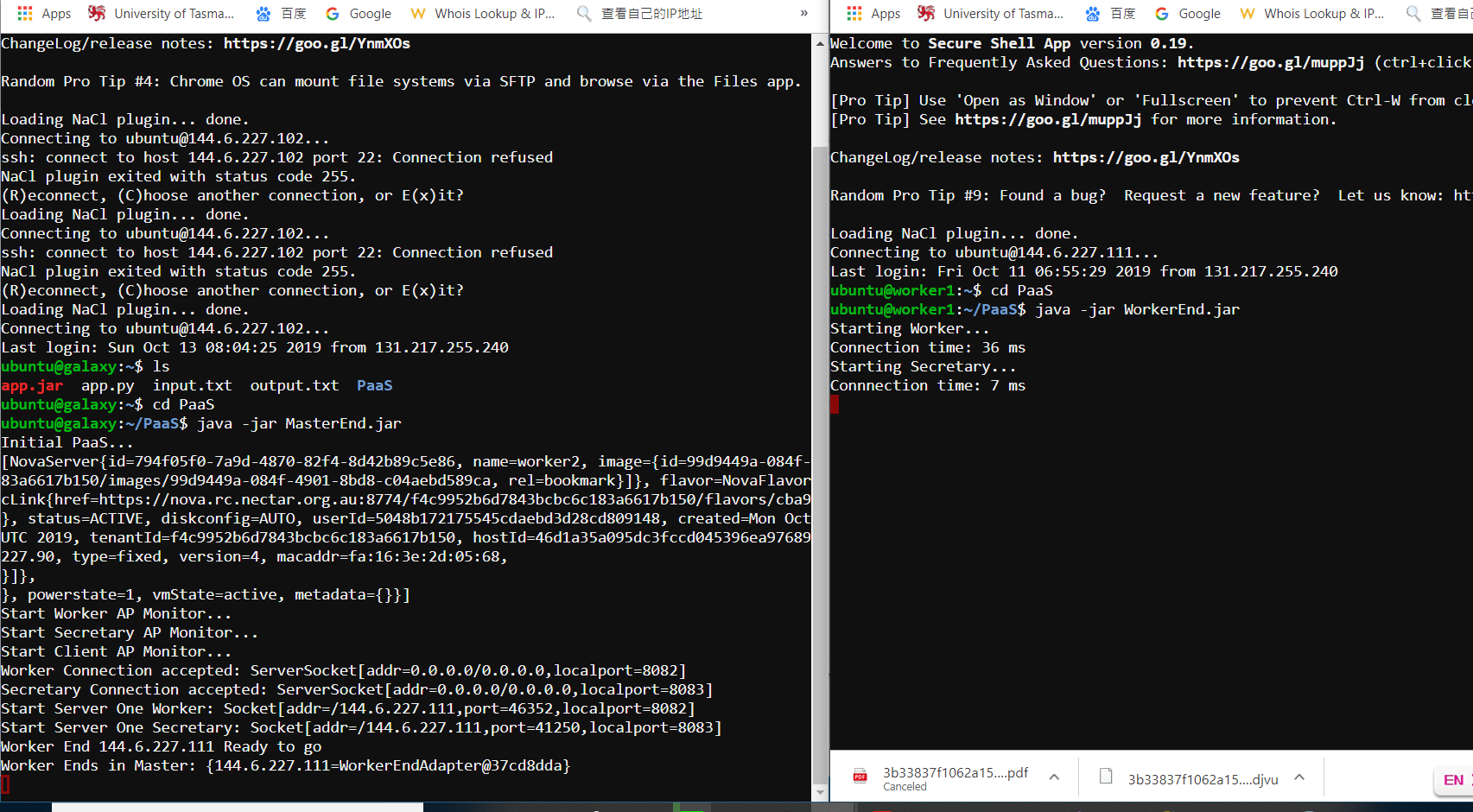
# Class Diagram



# Sequence Diagram



# Evaluation

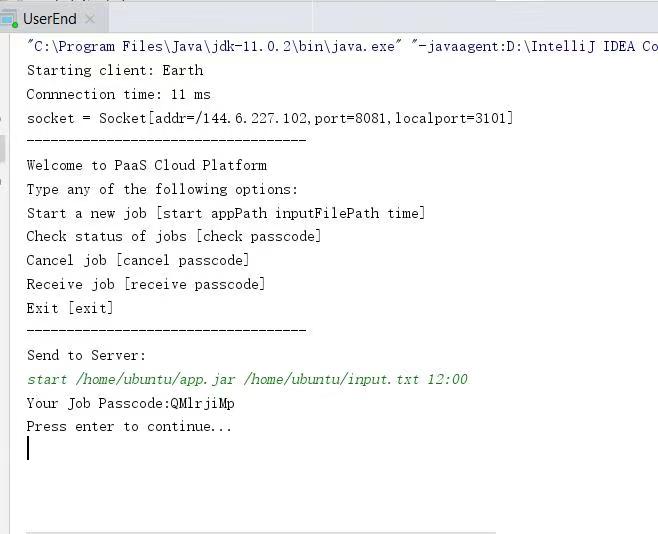
This is the startup cloud, the Master End on the left and the worker end on the right

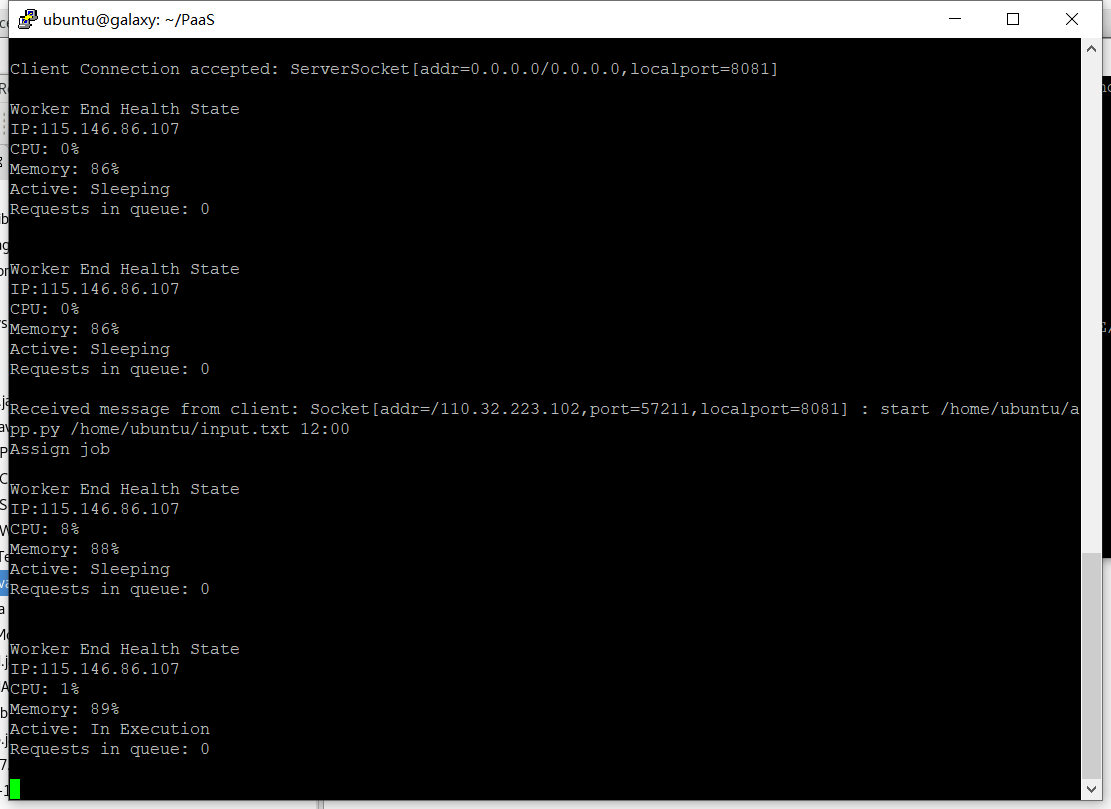
# User end submit a request 068102eafd50b2ce2573e413c0395dd

# This is the output of the user end submission and the generated bill

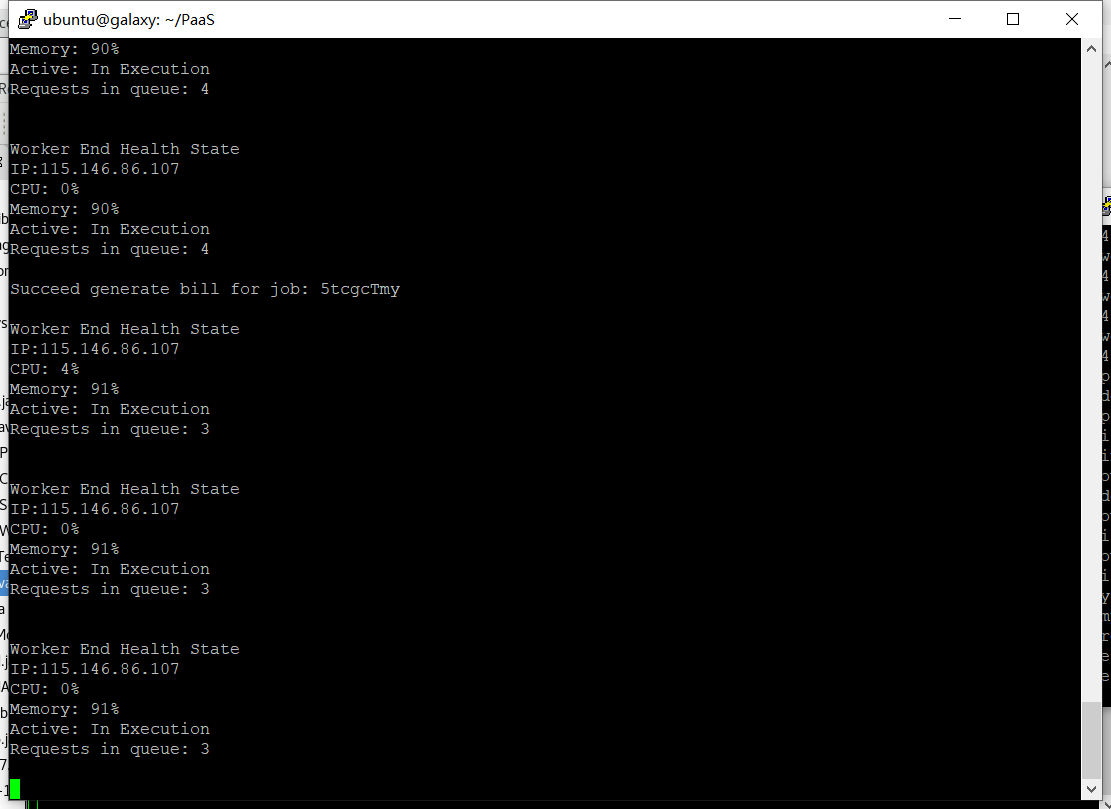
# 1f7a8c964973c51945a99a505f2cf67

This is the menu that the user uses the software, all the executions are shown here.

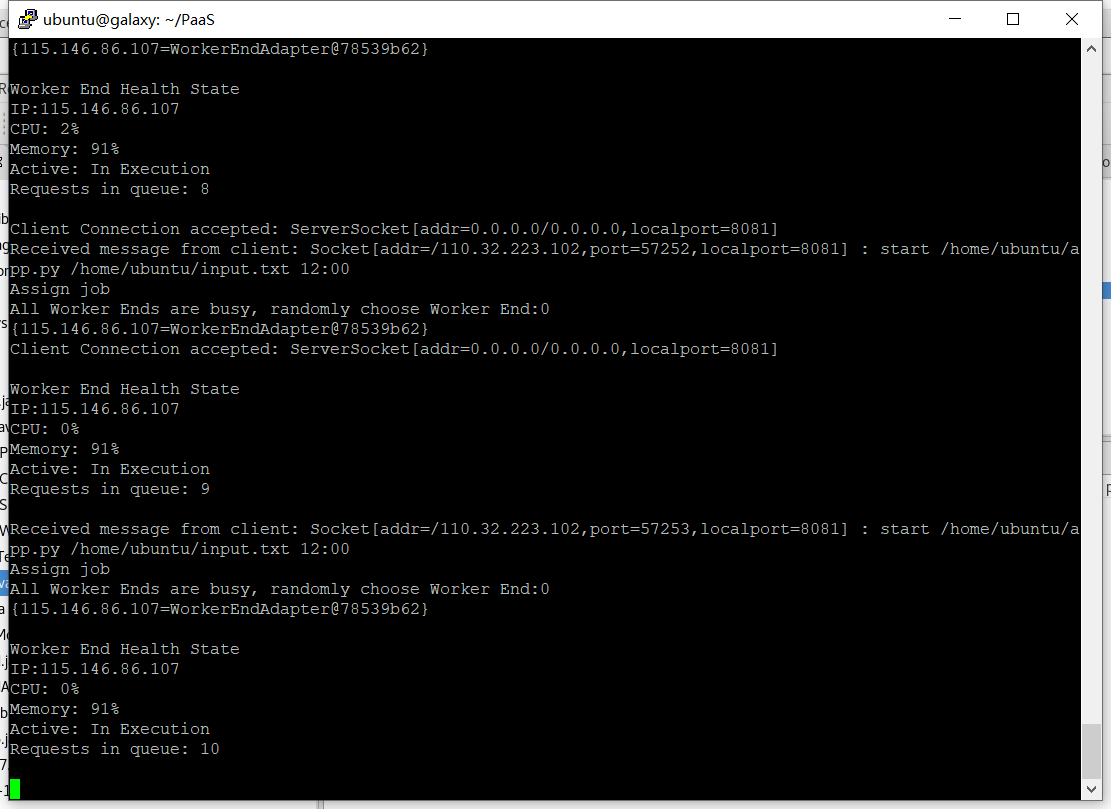




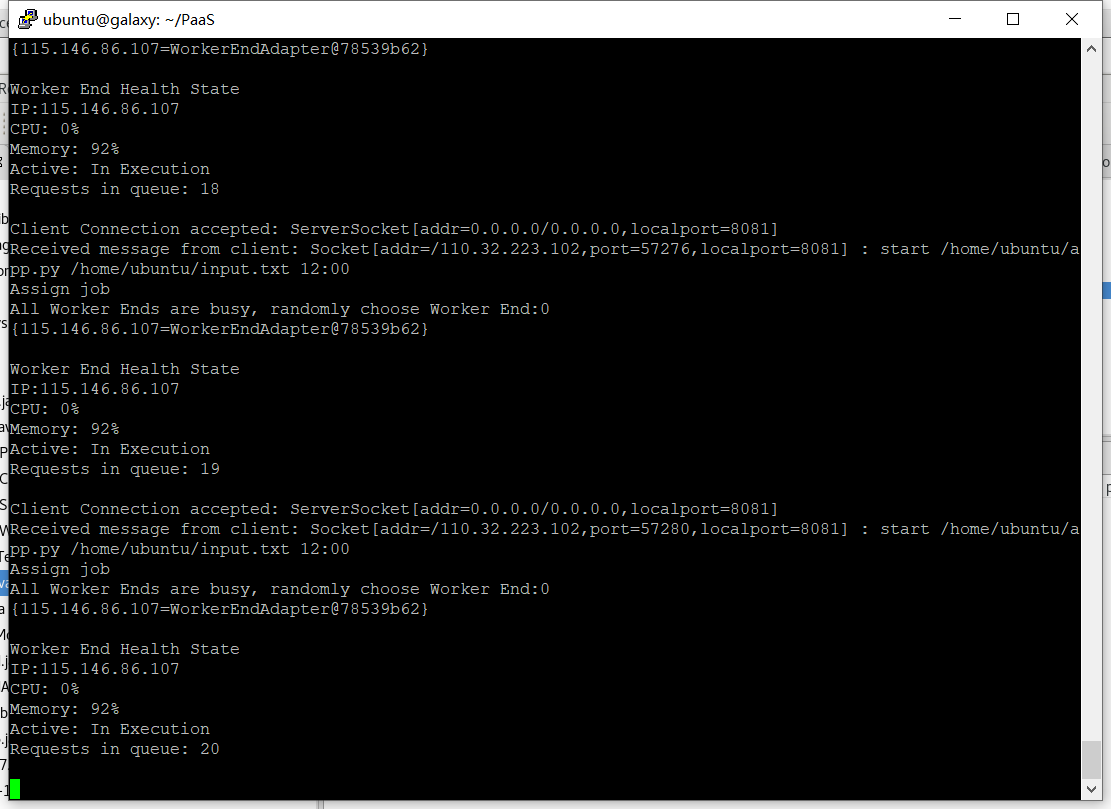
Receieved one request



Receieved 3 requests



Receieved 10 requests



Receieved 20 requests

As the above data, the amount of the requests from 1 to 3 to 10 to 20 (continuous requesting), the status of CPU/Memory are pretty stable, so we can assume the project can support at least small-scale requests elastic.

Besides, it also captured the exceptions during the connection, transferring and communication between server and user end.

# Each member implementation

Yinz xie and yuyang peng designed the framework of the program, the communication process and the client part.

Jing liu designed and completed MasterEnd, secretary and server

Yuhang Yang designed and completed the Worker part, including worker, workerEnd, creating new workers.