17COP505 – Internet Systems Assignment

Semester 1 2017

1 Introduction

Mobile phones are a convenient way of communication. There were 54,403 unique Base Stations in the UK* providing coverage for millions of mobile phone users. Whenever your phone is switched on, it is connected to a base station at a certain signal strength. As you move around, it is connected to a different base station at a different signal strength. These data are collected by mobile phone companies and processed in order to identify areas for improvement: e.g. areas that have poor coverage, or extremely busy areas that might need more base stations to balance the load. In this assignment we will simulate such a mobile phone network monitoring system.

By doing this assignment, you will gain practical experience in:
* Python programming
* RESTful API design and implementation
* HTTP communication between client and server
* Storing data in mongoDB
* Problem-solving and system design skills
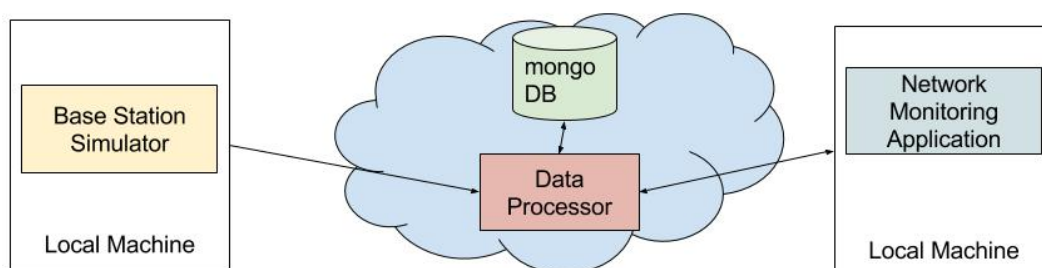
You will work in groups and groups will be allocated during the first week of the module. In order to arrive at an individual mark for each group member we will use a peer assessment system that will ask you to reflect on the performance of yourself in relationship to the others in your group. This will be shown during the lectures.

2 Base Station Simulator (25%)

Your first task is to create a program in Python that **simulates** a Base Station. Each Base Station should have the following properties:
- Unique ID
- Location (Street Name, Town/City, County)
- Capacity (how many users can connect to it at the same time)

Your Base Station should have the following functionalities:
- At startup, it should inform the rest of the system its properties and the time (timestamp) it started.
- Your program should simulate mobile phones connecting to the Base Station and leaving it, and generate random (but realistic) information. Each time a mobile phone is connected to a Base Station, the phone number, time of connection, and which Base Station it connects to is recorded. Upon leaving the Base Station, its record is deleted.
- Your program should send these data to the Data Processor on the server-side to be stored.
- The flow rate of traffic – when new mobile phones join the Base Station– should be random, not hard coded.
- Offline functionalities: your program should be able to save messages in memory (in a data structure of your choice) when an Internet connection is not available, and send them when the connection is established again.

Multiple instances of Base Station are expected to run on a local computer, to simulate the network of Base Stations in an area.

3 Data Processor (25%)

Another task of the system is a data processor in Python deployed on the co-project server, which receives the messages sent from the Base Stations and stores them for future analysis. In this coursework we will use a mongoDB database to store the data. Instructions on using sci-project will be given in the lecture or lab.

The data processor should provide the following functionalities:
- Run continuously
- Provide series of well-designed RESTful APIs, and communicate with the client-side (Base Station Simulator and Network Monitoring Application) using HTTP request response.
- Store the Base Station information and the mobile phone connection information in two separate collections.

4 Network Monitoring Application (20%)

The network monitoring application is a separate Python program that queries the store data to provide interesting insights. It should be able to:
- Produce a list of Base Stations and their details in a given area (town, city, county)
- Produce a list of busiest Base Stations (those that have reached 90% or more of its capacity)

5 Report (30%)

You should submit a report summarising the work carried out and evaluating what you achieved. The report should include but is not limited to:
- Clear instructions on how to use your system, including any pre-requisite conditions
- Details of system design and implementation of each task
- Justification of any assumptions made and any decisions made
- Test plan and result analysis
- Reflection on any difficulties met, problems solved, lessons learned, and any possible future improvement

6 Submission

One person per group should submit all the source code and the group report in a single .zip file on Learn by 4pm on Thursday in week 3.

Peer Assessment Questions and Answers

Peer Assessment exists to provide some discrimination in marks between different members of the same group. We will use the webpa system to determine this. The webpa questions will be based on the learning outcomes for the activity. Questions are asked of one student to describe the abilities of their peers.

————————————————————————————————-
How competent is this person in python programming?
Scoring range: 1-4
Score 1 : They cannot code at all
Score 2 : They can design code but are unsure about the more advanced details of the language.
Score 3 : Their code seems to work, but is presented untidyly with little reference to style guidelines.
Score 4 : You are fully confident in their abilities.
————————————————————————————————-
How competent is this person at debugging programs, given your experience of the group work.
Scoring range: 1-4
Score 1 : They cannot code.
Score 2 : They are able to debug, but are slow at spotting the issues.
Score 3 : They can think about the problem and analyse it to a solution.
Score 4 : They can lead the team in the debugging exercise.
————————————————————————————————-
How competent is this person when it comes to producing working software.
Scoring range: 1-4
Score 1 : They don't really understand what working means.
Score 2 : They have an attitude of when something works, it all works.
Score 3 : They can fully analyse the requirements and understand that their solution fulfils all that was asked.
Score 4 : They can think beyond the requirements into the future and how their code may be reused.
————————————————————————————————-
How competent is this person at RESTful API design and implementation?
Scoring range: 1-4
Score 1 : They don't really understand much.
Score 2 : They understand in theory but their implementation was of little success.
Score 3 : They can implement some functions.
Score 4 : You are fully confident in their abilities.
————————————————————————————————-
How competent is this person at making the data in mongoDB persistent?
Scoring range: 1-4
Score 1 : They don't really understand much.
Score 2 : They understand in theory but their implementation was of little success.
Score 3 : They can write some simple functions that save data and query them.
Score 4 : The group depended on their abilities to finish the most complicated tasks.
————————————————————————————————-
How much of the group report is understood by this person?
Scoring range: 1-4
Score 1 : Not much at all
Score 2 : Less than half
Score 3 : A large amount
Score 4 : Everything - the project depends greatly on their contribution.