# CNSCC311 Distributed Systems
# Coursework Assignment for Assessment

(Autumn Term, 2021)

Scott Piao

## 1. Introduction

This is a software development coursework for assessing your knowledge and skills of developing distributed systems. Your task is, by collaborating with your team members, to develop one of the following types of distributed systems:

1) Online shopping system,
2) Online food delivery system,
3) Online bookshop.

Your assessment marks will consist of the following three parts:

1) 20% from team work code and team report.
2) 40% from your team's software demo and team work presentation.
3) 40% from your individual report on your contribution to the team work.

*Important points:*

1) The software system must be your team's own work, and you cannot share your code with other teams except external public software libraries.
2) If you face technical problems during the coursework, please seek the solution among team members or the teachers, but not from other teams.
3) Each team must submit software source code and a team-work report. In addition, every team member must submit your individual report.
4) You should apply your theoretical knowledge and techniques learned during the course as much as possible to develop a high-quality distributed system.

## 2. Task of coursework

Your team's task is to develop an *online shopping system*, *online food delivery system*, or *online bookshop* in the form of distributed system. Your system will be a small-scale but fully functioning system that provides typical functionalities of real-world similar systems. Typically your system will consist of a distributed database management system, a cluster of web services, and client application/s (web-based application, desktop application, or mobile phone app). Your work should meet the following requirements as closely as possible.

### 2.1 Functionality of your system

Your system should be fully functional, capable of providing basic online operations needed to provide general services for users, such as checking existing items, ordering items, making payment, returning items (e.g. broken items etc.). Each team can decide the types of goods for sale, but your system should provide the following functionalities:

- A user can check what items are available for sale. Your system should be able to list the available items and their prices for the users.
- Your system should be able to help a user to complete a transaction – make payment (of course not with real bank accounts) and deliver the item (again *not real items, but by sending appropriate messages or pictures of goods*).
- Your system should be able to update the database in real-time to reflect the transaction, e.g. decreasing the number of items available for sale.
- Your system should have a basic security mechanism of username-password access control. If you can implement encryption for your system for securely passing users' bank accounts, addresses etc., that would be a plus.

### 2.2 Characteristics of Distributed Systems

Your system should demonstrate as many characteristics of distributed systems as possible. First of all, you should be able to deploy the components of your system at least on three computers, and the more the better. In addition, your system should be able to handle multiple users' requests from different computers (at least two computers).

Furthermore, your system should have a good level of fault-tolerance and transparency. For example, even if a database, a web service, or even a computer is crashed or turned off, your system should still function. Your should implement some algorithms to handle failures of components of your system. For example, if one of your networked databases or services is turned off, your system should still be able to send a reasonable response to the clients.

Finally, your system should be easily expandable, by simply keeping deploying your database and web service on more computers.

### 2.3 Requirements of software implementation

Your software code should be logically structured and clearly written, with proper comments in code. Your program code should allow other software developers to easily understand and reuse your software code and components.

Your system should include at least one RESTful and SOAP service, which should easily be deployable and accessible on multiple computers. While these are various web server software, the GlassFish is suggested as the default one, because it supports convenient and reliable SOAP deployment.

Your client software can be any one or combination of web-based, desktop, and mobile applications. They should provide user friendly interfaces, and should be easily deployable on multiple computers or mobile devices. If you can implement two types of client programs, it would be a plus.

**2.4 Team work report (maximum 3 pages excluding references)**

Each team should jointly produce a report on your work. The team leaders are responsible for organising the writing of the team report. The team report should contain the following key points:

- Group and Team numbers, names of team members (Pin-Yin).
- Explain how team work is organised, e.g. list team members' sub-tasks.
- Brief explanation of system functionality - what is it capable to do?
- Outline of architecture and structure of your system, and justify your choice of the architecture in terms of Distributed System.
- Describe key technical specifications of your system, such as algorithms you used for handling component failures, encryption methods etc.
- A manual that can help users to use your system (mainly about client software).
- If applicable, provide a reference list of papers or other resources you used in your work.

**2.5 Individual report (one page)**

Everyone should write a report on your contribution to your team work. This should include:

a) Which part or aspect of the team work were you responsible for?
b) How did you co-operate with other team members?
c) How did your work help to improve the quality of your team's work?
d) Any suggestions and comments on your team's work for future improvement.

**2.6. Submission of coursework**

A Moodle submission interface will be open from 9th November 2021. You must submit all your work, including software source code (by team leaders), teamwork reports (by team leaders) and individual reports (by everyone) via the submission site by the end of 12th November 2021 (UK Time). Every team needs to plan your work schedule carefully to ensure that everyone can finish the coursework in time with a high quality.

## 3. Teamwork demo and presentation

A team software demo and presentation will be arranged in the lab sessions on 16th November. This is to assess the quality of your team's work as a whole. Each team will have up to 15 minutes. Each team can decide who (at least two team members) or how to carry out the software demo and presentation, but you must demonstrate

your collective capability and skills of effectively presenting your achievement with appropriate materials in clear language.

## 3. Assessment

The three assessment components, “team work code and team report”, “team software demo and presentation” and “individual report” will be marked separately using a 100% scale. Then the marks of the components will be combined with their respective percentages (20%, 40%, 40%) into the final grades.

The marking criteria include:

a) Quality of the submitted code and reports.
b) Merit and contribution of individual team members to the team work.
c) Quality of software demo and presentation.
d) Quality of English writing (of reports) and speaking (in presentation).