City University of Hong Kong
Course Syllabus

offered by Department of Computer Science
with effect from Semester B 2016/17

Part I   Course Overview

Course Title:	Internet and Distributed Systems Programming
Course Code:	CS6282
Course Duration:	One semester
Credit Units:	3 credits
Level:	P6
Medium of Instruction:	English
Medium of Assessment:	English
Prerequisites: *(Course Code and Title)*	Nil
Precursors: *(Course Code and Title)*	CS5222 Computer Networks and Internets or CS5281 Internet Application Developments or CS6223 Distributed Systems
Equivalent Courses: *(Course Code and Title)*	Nil
Exclusive Courses: *(Course Code and Title)*	Nil

Part II   Course Details

1.   Abstract

This course aims to develop technical competence in programming applications and services over TCP/IP networks and in designing reliable and high performance distributed systems.

2.   Course Intended Learning Outcomes (CILOs)
(CILOs state what the student is expected to be able to do at the end of the course according to a given standard of performance.)

No.	CILOs	Weighting (if applicable)	Discovery-enriched curriculum related learning outcomes (please tick where appropriate)		
			A1	*A2*	*A3*
1.	Explore and identify the challenges and requirements in distributed applications, including cloud computing.	10%		✓	
2.	Program at TCP/IP and socket levels.	20%		✓	
3.	Program multi-thread concurrent applications.	30%		✓	
4.	Apply distributed application development methods, including use of cloud administration and application development tools and API.	20%		✓	
5.	Develop a distributed system and related algorithms for reliability and performance.	20%			✓
		100%			

A1:   Attitude
Develop an attitude of discovery/innovation/creativity, as demonstrated by students possessing a strong sense of curiosity, asking questions actively, challenging assumptions or engaging in inquiry together with teachers.
A2:   Ability
Develop the ability/skill needed to discover/innovate/create, as demonstrated by students possessing critical thinking skills to assess ideas, acquiring research skills, synthesizing knowledge across disciplines or applying academic knowledge to self-life problems.
A3:   *Accomplishments*
Demonstrate accomplishment of discovery/innovation/creativity through producing /constructing creative works/new artefacts, effective solutions to real-life problems or new processes.

3. Teaching and Learning Activities (TLAs)
(TLAs designed to facilitate students' achievement of the CILOs.)

Teaching pattern:
Suggested lecture/tutorial/laboratory mix: 2 hours lecture; 1 hour tutorial.

TLA	Brief Description	CILO No. 1	2	3	4	5	Hours/week (if applicable)
Lecture	Lectures will cover the essential concept, common platforms and core technologies in the Internet and distributed system programming.	✓	✓	✓	✓	✓	
Tutorial	Tutorials will be in form of case studies, analytical questions and answers, and programming exercises. Case studies and analytical questions are designed to review the material covered in the lectures and widen students' exposure on the related topics. Programming exercises provide hand-on practices on the Internet and distributed system programming.	✓	✓	✓	✓	✓	
Programming Assignment	Student will develop a distributed application/system that • demonstrate a good understanding on the characteristics and challenges of distributed system. • apply various knowledge on the Internet and distributed system in solving real-life problem. • explore new distributed application/system	✓	✓	✓	✓	✓	
Written Assignment	Students will conduct a survey on common distributed system/application and provide an evaluation and potential improvement of their findings.	✓	✓	✓			

4. Assessment Tasks/Activities (ATs)
(ATs are designed to assess how well the students achieve the CILOs.)

Assessment Tasks/Activities	CILO No. 1	2	3	4	5	Weighting	Remarks
Continuous Assessment: <u>50%</u>							
Written Assignment	✓	✓	✓			15%	
Quiz	✓	✓	✓	✓		20%	
Programming Assignment	✓	✓	✓	✓	✓	15%	
Examinationˆ: <u>50%</u> (duration: 2 hours)							
						100%	

ˆ For a student to pass the course, at least 30% of the maximum mark for the examination must be obtained.

5. Assessment Rubrics

(Grading of student achievements is based on student performance in assessment tasks/activities with the following rubrics.)

Assessment Task	Criterion	Excellent (A+, A, A-)	Good (B+, B, B-)	Adequate (C+, C, C-)	Marginal (D)	Failure (F)
1. Written Assignment	ABILITY to compare different technologies in the Internet and distributed system programming	High	Significant	Moderate	Basic	Not even reaching marginal levels
	ABILITY to provide concise and thorough evaluation on common distributed systems	High	Significant	Moderate	Basic	Not even reaching marginal levels
2. Programming Assignment	ABILITY to identify the characteristics and challenges of the selected distributed systems and consider these factors in developing their application	High	Significant	Moderate	Basic	Not even reaching marginal levels
	ABILITY to justify their system design and implementation based on a thorough understanding on the strength and weakness of various distributed system technologies	High	Significant	Moderate	Basic	Not even reaching marginal levels
	DEVELOP robust and sophisticated distributed application	High	Significant	Moderate	Basic	Not even reaching marginal levels
3. Exam	ABILITY to describe and identify the characteristics and constraints of programming distributed application	High	Significant	Moderate	Basic	Not even reaching marginal levels
	ABILITY to compare different technologies and algorithms found in the Internet and distributed system, and provide justification on their usage based on different scenarios	High	Significant	Moderate	Basic	Not even reaching marginal levels
	ABILITY to demonstrate working knowledge on the technologies and skills required to develop distributed systems	High	Significant	Moderate	Basic	Not even reaching marginal levels

Part III Other Information (more details can be provided separately in the teaching plan)

1. **Keyword Syllabus**
    (An indication of the key topics of the course.)

    Socket level API's, Java Networking, RMI Programming, Session Tracking, Multithreaded Programming, Programming for Distributed Services, JavaBeans and EJB Technologies, XML Parsers, SAX and Cloud API's, SOAP, Web Services, Programming for Reliability, Load Sharing and Balancing, Cloud Computing, Virtualization, SOA, Utility Computing, SaaS, PaaS, IaaS.

2. **Reading List**
2.1 **Compulsory Readings**
    (Compulsory readings can include books, book chapters, or journal/magazine articles. There are also collections of e-books, e-journals available from the CityU Library.)

1.	*M. L. Liu, Distributed Computing: Principles and Applications, Addison-Wesley, 2004.*
2.	*Thomas Erl, SOA: Principles of Service Design, Prentice Hall, 2007.*
3.	*Andrew S. Tanenbaum, Maarten van Steen, Distributed Systems: Principles and Paradigms, 2nd Edition, Prentice Hall, 2007.*

2.2 **Additional Readings**
    (Additional references for students to learn to expand their knowledge about the subject.)