

## CS - 203 (Principle of Computer Science III)

### Semester IV

### Course Description

<b>Course Code Number</b>	<b>CS-203</b>	<b>Course Title</b>	<b>Advanced Java Programming(J2EE)</b>
<b>Semester Hours</b>	Total 5 hours per week Lecture 2 hours per week Lab 2 hours per week Extra lab 1 hour to practice and submit exercises	<b>No. of Credit Units</b>	3
<b>Prerequisite</b>	<b>ITSM-101, CS-202</b>	<b>Course Coordinator</b>	Dr. Thinn Lai Soe Faculty of Computer Science
<b>Course Length</b>	15 Weeks	<b>Type of Instruction</b>	Lecture + Lab

### Course Objectives

The objectives of the course are as follows:

- To explain how to develop client-server programs in Java.
- To learn and develop distributed Java applications using Remote Method Invocation (RMI)
- To know the fundamentals of Java Web application development features, such as the Java Servlet, Java Server Pages, and Java Standard Tag Library (JSTL).
- To develop a dynamic web application by the use of servlet, JSTL, JSP and database technology.
- To know about Model View Controller theory.
- To be able to develop a small web project independently.

### Course Outlines

The outline of the course covers the following topics:

- Networking Basics & Java Networking using Sockets
- Understanding RMI
- Web-based architecture
- Java Server Page (JSP) technology
- Java Servlets technology
- Java Standard Tag Library (JSTL) technology
- JDBC and Database Programming
- MVC Theory

## Learning Outcomes

On completion of the course the student will be able to:

- Recognize and understand both the elements and characterizing of client-server architecture in java programming.
- Know the architecture of RMI, understand how to serialize an object, create and implement the class that can be used to create remote server objects.
- Know how to design and build a web application using Servlet, JSP, JSTL
- Know how to interact with databases and how to plan web development.
- Gain the skills and project-based experience needed for entry into web application.
- Learn about Model View Controller theory.
- Develop a small project independently.

## Reference Books

1. Java Server Programming (J2EE 1.4 Edition) Black Book (2006 Print) by Steven Holzner et al
2. Online Tutorials

## Course Organization

Student participation in this course will involve the following activities:

- Attending the lectures
- Lab
- Test (Moodle)
- Quiz
- Assignments
- Project

## Assessment Plan for the Course

Project	40 %
Assignments	20 %
Quizzes/ Moodle	20 %
Class Participation	10 %
Lab Test	10 %

## Grading System

UCSY follows a letter grade system comprising of grades A, A-, B+, B, B-, C+, C, C-, D and F. All marks obtained by students during the semester will be used in the grading process. A grade of “C” or better is required in this course because it is a prerequisite for other courses in the program. **The student who gets the grade point less than 2 must sit Re-Exam.**

**The grading scale for this course is:**

Marks obtained	Letter Grade	Grade Point
----------------	--------------	-------------

University of Computer Studies  
B.C.Sc. / B.C.Tech.

>=90	A	4
85 – 89	A-	3.75
80 – 84	B+	3.25
75 – 79	B	3
70 – 74	B-	2.75
65 – 69	C+	2.25
60 – 64	C	2
55 – 59	C-	1.75
50 – 54	D	1
0 – 49	F	0

**Fail Grade and Re-Exam: C-, D, and F (Grade point <2)**

**Class Attendance and Participation Policy:**

- **Attendance:** Class attendance is **mandatory**. Most of the material you will learn will be covered in the lectures, so it is important that you not miss any of them. You are expected to show up **on time** for class, and **stay for the whole lecture**. Students are expected to attend each class, to complete any required preparatory work (including assigned reading) and to participate actively in lectures, discussions and exercises.
- **Mobile phones must be silenced and put away** for the entire lecture unless use is specified by the instructor. You may **not make or receive** calls on your cell phone, or send or receive text messages during lectures.
- You are responsible for all material sent as email. Ignorance of such material is no excuse. You are responsible for all materials presented in the lectures.
- Your conduct in class should be conducive towards a positive learning environment for your class mates as well as yourself.

**Assignments, Quizzes, Labs and Test with Moodle**

Students take a short 3 to 5 quiz for every lecture and 10 or 20 points quiz / Moodle test after each lecture or chapter. The intent of the quiz/Moodle is to discover early where the areas of misunderstanding may lie. They will account for 20% of the student's grade. The Any assignment or quiz is simply missed, regardless of the reason why (e.g. illness, work, traffic, car trouble, computer problems, death, etc.), and earns a grade of zero. You are strongly encouraged to complete all assignments and attend all quizzes so that you can check that you understand the material and can throw out bad grades, or grades for which you had to miss an assignment or quiz for a valid reason. Late submissions will not be accepted for any graded activity for any reason. Students will have the opportunity to review the quizzes and see the correct answers once they have been graded. Student need to answer test which will announce by lecturer.

- **There are no extra credit opportunities.**

Students may not do additional work nor resubmit any graded activity to raise a final grade.

- **Test**

Test will start after one or two chapters finished and the coordinator will announce the date for the test.

- **Project**

Project will start after finishing the whole course. You will have to make your own project using the whole lectures. After finishing the project, you will have to make your project book for your system, project presentation for about 15 minutes and there will be coding test for each person after presentation. You are responsible for your project about able to explain your system and your code, able to update your code at once when the teachers wish to amend your code.

- **Exam**

The exam will be conducted on-campus, in a lab. The dates/times/locations will be posted on Board as soon as possible.

For this course, the following **additional requirements are specified:**

All work submitted for a grade must have been prepared by the individual student. Students are expressly prohibited from sharing any work that has been or will be submitted for a grade, in progress or completed, for this course in any manner with a person other than the instructor and teaching assistant(s) assigned to this course). Specifically, students may not do the following, including but not limited to:

- Discuss questions, example problems, or example work with another person that leads to a similar solution to work submitted for a grade.
- Give to, show, or receive from another person (intentionally, or accidentally because the work was not protected) a partial, completed, or graded solution.
- Ask another person about the completion or correctness of an assignment.
- Post questions or a partial, completed, or graded solution electronically (e.g. a Web site).
- All work must be newly created by the individual student for this course. Any usage of work developed for another course, or for this course in a prior semester, is strictly prohibited without prior approval from the instructor.
- Posting or sharing course content (e.g. instructor provided lecture notes, assignment directions, assignment questions, or anything not created solely by the student), using any non-electronic or electronic medium (e.g. web site, FTP site, any location where it is accessible to someone other than the individual student, instructor and/or teaching assistant(s)) constitutes copyright infringement and is strictly prohibited without prior approval from the instructor.

- **Programming Style and Documentation**

Each assignment program must start with comments stating your name, the assignment number, and a brief description of the program's purpose. Students are expected to follow the Programming Style recommendations given in class and in the text. For example (but not limited to):

- Indent statement

- Use descriptive identifier naming conventions
- Use single blank lines to separate sections of the program
- Align opening and closing braces
- Include appropriate comments throughout the program.

**Note:** Student who does not follow good programming practices will result in a deduction from the grade for that assignment.

### Tentative Lecture Plan

No.	Topics	Week	Remark
	<b>Networking with Java</b>		<b>Chapter (14)</b>
1.	Basics of Networking Sockets in Java Client-Server in Networking Internet Addressing Domain Name Service The URL class Java Net API's The Networking interfaces and classes Creating and using Sockets Working with Datagrams	Week 1+2	
	<b>Remote Method Invocation (RMI)</b>		<b>Chapter (31)</b>
2.	Introduction to RMI (Remote Method Invocation) Client-Server Architecture Implementing RMI - Writing an RMI Server - Designing a Remote Interface - Implementing a Remote Interface - Passing Objects in RMI Implementing the Server's main method - Creating a Client Program - Compiling and Running Example	Week 2+3	
	<b>Introduction to J2EE and Web Containers</b>		<b>Chapter (1) &amp; (7)</b>
3.	Need for Enterprise Programming Introduction to J2EE Components Introduction to J2EE Container Understanding the HTTP Protocol	Week 4+5	

University of Computer Studies  
B.C.Sc. / B.C.Tech.

	Introducing Web Applications and Web Containers Web Application Life-Cycle Creating a Web Application		
	<b>Understanding Servlet Programming</b>		<b>Chapter (8)</b>
4.	Overview of Servlets Servlet Configuration Client-Server Servlet Programming Servlet Life-Cycle Servlet Request and Response Model Reading Data from Servlets Request Delegation and Request Scope	Week 6	
	<b>Understanding Servlet Session</b>		<b>Chapter (9)</b>
5.	What is a Session? Session Tracking Introducing Session Tracking Session Tracking and Java Servlet API Demonstrating Session Life-Cycle with Cookies	Week 7	
	<b>Understanding of Java Server Pages and JSTL</b>		<b>Chapter (10)</b>
6.	Introduction to JSP First Java Server Page Example Life cycle of JSP page Dynamic contents generation techniques in JSP Scripting elements Directives Beans Expression language Implicit Objects Error Handling Custom tag libraries What is JSTL? About JSTL Core Tags	Week 8+9	
	<b>Java Database Connectivity</b>		<b>Chapter (6)</b>
7.	Overview of ODBC Technology Overview of JDBC Technology	Week	

University of Computer Studies  
B.C.Sc. / B.C.Tech.

	Introduction to JDBC Package Demonstrating example database programs in Servlets and JSP	10+11	
	<b>Understanding Model View Controller</b>		
8.	Overview of MVC Understanding 3-tier(Presentation, Business and DataAccess) architecture using MVC Demonstrating example database programs in Servlets and JSP using MVC	Week 11+12	
9.	Project	Week 12 to 15	The last weeks is intended to develop the small web project using Servlet, JSP, JSTL and Database Technology using MVC.