CSE 3324
Distributed Client/Server Programming

REQUIRED/ELECTIVE:
Computer Science – Required
Software Engineering – Required
Computer Engineering – Required

CATALOG DATA:
(Prerequisite: CS 2383 with a grade of C or better). Three hours lecture. Three hours laboratory.
Design of software systems for use in distributed environments. Client/Server models,
multithreaded programming, server-side web programming, graphical user interfaces; group
projects involving client/server systems.

PREREQUISITE BY TOPIC:
1. Object-oriented programming methodology.
2. Data structures and algorithms, including stacks, linked lists, binary search trees, AVL trees
and graphs.
3. Memory management, database management models and systems, hash tables, file I/O

TEXTBOOKS AND OTHER REQUIRED MATERIAL:
Handouts and websites www.cse.msstate.edu/~cs3324.

COORDINATOR:
Dr. Donna S. Reese

COURSE OBJECTIVES:
1. To improve programming skills.
2. To introduce new programming paradigms (multi-threaded, event-driven, client/server).
3. To advance understanding of the software process, with particular emphasis on analysis,
design and testing.
4. To introduce students to graphical user interface programming.
5. To develop students’ ability to function in teams.
6. To develop students’ oral communications skills in application to the computing discipline.

TOPICS COVERED:

<table>
<thead>
<tr>
<th>Lecture</th>
<th>(Number of class hrs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Basic web page development</td>
<td>3</td>
</tr>
<tr>
<td>2. Advanced Programming techniques</td>
<td>21</td>
</tr>
<tr>
<td>3. Server Side programming</td>
<td>12</td>
</tr>
<tr>
<td>4. Interfacing server programs with databases</td>
<td>3</td>
</tr>
<tr>
<td>5. Distributed computing</td>
<td>3</td>
</tr>
<tr>
<td>6. Tests</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Laboratory</th>
<th>(Number of lab weeks)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Lab orientation; Introduction to programming environment &amp; basic web page development.</td>
<td>1</td>
</tr>
<tr>
<td>2. Graphical User interfaces and event programming.</td>
<td>3</td>
</tr>
<tr>
<td>3. Multithreaded applications</td>
<td>1</td>
</tr>
<tr>
<td>4. Server side programming</td>
<td>1</td>
</tr>
<tr>
<td>5. Database server accesses programming</td>
<td>1</td>
</tr>
<tr>
<td>6. More Server side programming</td>
<td>1</td>
</tr>
</tbody>
</table>
CONTRIBUTION TO PROFESSIONAL COMPONENT:
Engineering Topics of Engineering Science and Design

ASSESSMENTS:
1. Short quizzes in lecture meetings and tests.
2. Software development group term project.
3. Individual laboratory exercises.
4. Oral presentation.

RELATIONSHIP TO PROGRAM OUTCOMES:
Note: Parenthesized list indicates the ABET criteria, Computer Engineering outcomes, and Software Engineering outcomes addressed by each performance criteria.

Performance Criteria:
1. The student will be able to list and explain the major phases of the software development process. (c, cpe2, se5)
2. The student will be able to conduct a system analysis, write a software requirements specification, design a software product and develop a test plan from requirements specifications. (c, cpe2, cpe7, se5)
3. The student will be able to explain the fundamental concepts, issues, and metrics of distributed computing. (a, c, e, cpe2, cpe9)
4. The student will be able to implement applications software using the Java programming language. (k, cpe9, se3, se4, se7)
5. The student will be able to effectively work on a team to develop a software product. (d, cpe9, se1)
6. The student will be able to effectively present, both orally and in writing, the requirements, design and functionality of a software system. (g, cpe7, se2)
7. The student will be able to explain the event-model programming paradigm and write software using this model. (k, cpe2, cpe9, se7)

PREPARED BY:

ESTIMATE CSAB CATEGORY CONTENT:

<table>
<thead>
<tr>
<th></th>
<th>CORE</th>
<th>ADVANCED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Structures</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Algorithms</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Software Design</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>CORE</th>
<th>ADVANCED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Organization and Architecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Concepts of Programming Languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ORAL AND WRITTEN COMMUNICATIONS:

Every student is required to submit at least 3 written reports (not including exams, tests, quizzes, or commented programs) of typically 10-15 pages and to make 1 oral presentations of typically 10 minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.
SOCIAL AND ETHICAL ISSUES:

None.

THEORETICAL CONTENT:

Concurrent programming concepts including synchronization and communication issues – approximately 3 hours lecture and one laboratory assignment.

PROBLEM ANALYSIS:

For the group project, each student participates in a group of 3-4 students who are required to analyze a problem description and produce a complete requirements document that is based on this analysis. This document goes through two review cycles with the class instructor.

SOLUTION DESIGN:

Each of the 8-9 individual lab assignments requires the students to design a software solution to a specified problem. In addition, students must design and implement a web-based client/server application for their group project. Each group is required to write a formal design document that goes through two review cycles with the class instructor.