

CIS 3287 Software Design/Practicum

Course: CIS3287
Course Title: Software Design/Practicum
Time: MWF 1:00 – 1:50 M (Lab) 9:00 – 10:50
Place: MWF TL403B M (Lab) CC 207
Instructor: Paul Wolfgang
Instructor Phone: 215-204-5155
Office Hours: M 4:00 – 5:00 W 10:00 – 10:50 & 4:00 – 5:00 F 10:00 – 10:50
Course Web Page: www.cis.temple.edu/~wolfgang
Prerequisites: C or better in:
CIS 2168 Data Structures
CIS 3207 Introduction to System Programming and Operating Systems

Textbooks: Flexible, Reliable Software Using Patterns and Agile Development
Christensen
CRC Press
ISBN: 978-1-4200-9362-9

Object-Oriented Design & Patterns
Horstmann
John Wiley & Sons.
ISBN: 978-0-471-74487-0

Java Power Tools¹
Smart
O'Reilly
ISBN: 978-0-596-52793-8

¹ Available on-line through the Temple Library

References Beautiful Code¹
Oram & Wilson
O'Reilly
ISBN: 978-0-596-51004-6

Pragmatic Unit Testing in Java with JUnit
Hunt & Thomas
The Pragmatic Book Shelf
ISBN: 978-09745140-1-7

JUnit in Action, 2nd Ed
Tahchiev, Leme, Massol, & Gregory
Manning Publications Co.
ISBN: 9781935182023

Head First Design Patterns¹
Freeman & Freeman
O'Reilly
ISBN: 978-0-596-00712-4

Test-Driven Development by Example
Beck
Addison-Wesley
ISBN 978-321-14653-3

Growing Object-Oriented Software, Guided by Tests
Freeman & Pryce
Addison-Wesley
ISBN: 978-0-321-50362-6

Refactoring Improving the Design of Existing Code
Fowler
Addison-Wesley
ISBN: 978-0-201-48567-7

Version Control with Subversion^{1, 2}
Pilato, Collins-Sussman, & Fitzpatrick
O'Reilly
ISBN: 978-0-596-51033-6

Mecural: The Definitive Guide^{1,3}
O'Sullivan
O'Reilly
ISBN: 978-0-596-801311

¹ Available on-line through the Temple Library

² Available on-line from <http://svnbook.red-bean.com>

³ Available on-line from <http://hgbook.red-bean.com/read/>

**Course
Description**

Provides direct experience in the design, development, documentation, testing and maintenance of medium size software projects, in the use of modern software problem solving abstractions and solution patterns, and in the use of software development environments. This course is the capstone of the programming course sequence.

Course Goals:

To introduce the students to the following topics and to demonstrate their practical application.

- Graphic User Interface
- Access to Database
- Remote Procedure Calls (RPC)
- Extensible Markup Language (XML)
- Threading
- Object Oriented Design
- Unified Modeling Language (UML)
- Design Patterns
- Test Driven Development

To provide practical experience in using modern software development tools to perform the following tasks

- Project Management
- Version Control
- Build
- Test
- Issue Tracking

Provide experience working in small teams working on a project to modify an existing open-source program. Most Projects and code will start from or be inspired by:

- Sourceforge projects
- Apache Software Foundation projects
- Google Summer of Code

Grading:

- 10% Weekly quizzes - Brief in class quizzes to verify that students learn the material
- 20% Homework/Lab - Small assignments intended to make student practice with the concepts and tools presented
- 15% Midterm
- 30% Final Project - Six week long project done in small teams
- 25% Final Exam

Exam Dates:

- Midterm: Monday, October 10, 2011
- Final: Monday, Friday, December 16, 2011 10:30 – 12:30

Attendance Policy:

Attendance is mandatory. Unexcused absence will result in reduction of final grade.

Project

The goal of the project phase of this course is for students to gain experience in contributing to an open-source project. The nature of the contribution may be to contribute an enhancement or to fix an open problem. Students may work alone or in a team of up-to 3 students. While project work will be concentrated to the last half of the course, you need to select the project early and join the developer’s mailing list. A written project proposal is due Monday, September 19. This will contain:

- A one paragraph description of the overall project
- A one paragraph description of your proposed contribution

- A URL reference to the project.

Note: contributing to an open-source project is preferred. You may propose a stand-alone project, but if you do, it must be submitted as an open-source project.

Sources of open-source projects:

- Apache software foundation
 - <http://www.apache.org/>
- Open-Office
 - <http://www.openoffice.org/>
- Source-Forge
 - <http://sourceforge.net/>
- Open Hatch
 - <http://openhatch.org/>

Other Important Information

Disability disclosure:

Any student who has a need for accommodation based on the impact of a disability should contact me privately to discuss the specific situation as soon as possible. Contact Disability Resources and Services at 215-204-1280 in 100 Ritter Annex to coordinate reasonable accommodations for students with documented disabilities. (Temple University Policy and Procedures Manual)

Academic freedom:

Freedom to teach and freedom to learn are inseparable facets of academic freedom. The University has a policy on Student and Faculty and Academic Rights and Responsibilities (Policy #03.70.02) which can be accessed through the following link: http://policies.temple.edu/getdoc.asp?policy_no=03.70.02.

Academic Honesty

Academic cheating (such as plagiarism, copying during an exam, copying homework, stealing files and passwords, etc.) is strictly prohibited in this course. The penalty for the first offense will normally be an F in the course. A subsequent offense (in this or any other course) may also be referred to the University Disciplinary Committee.

No collusion what-so-ever during an exam will be tolerated. In particular not talking or other sharing of information (for example during open book exams) is permitted. Keep your eyes on YOUR paper.

IGNORANCE OF ACCEPTABLE GUIDELINES OF CONDUCT IS NO EXCUSE

http://policies.temple.edu/getdoc.asp?policy_no=03.70.12

Dates to Remember

First Day of Class:	Monday, August 29, 2011
Last Day to Drop:	Monday, September 12, 2011
Project Proposals:	Monday, September 19, 2011
Mid-term Exam	Monday, October 10, 2011
Last Day to Withdraw*:	Monday, October 31, 2011
Thanksgiving Break:	November 24 & 25, 2011
Last Day to makeup labs:	Monday, November 7, 2011
Last Day of Class:	Wednesday, December 7, 2011
Final exam	Friday, December 16 (10:30 – 12:30)

* Students may withdraw from or repeat a course only once.

Lecture and Lab Schedule

The lectures will be based on the Christensen text and on other sources.

Week	Date (Monday)	Lecture Topics	Quiz/Exam	Lab
1	8/29/2011	Introduction Test Driven Development Unit Testing Christensen Chapter 5		Lab 0 Account Setup Practice with IDE, development tools, version control
2	9/5/2011	Build Tools Polymorphism, Interfaces and Subclasses, The Strategy Pattern Christensen Chapters 6 – 8 Horstmann Chapter 4	Quiz 1	No Lab (Labor Day)
3	9/12/2011	Threading and Concurrency Horstmann Chapter 9	Quiz 2	Lab 1
4	9/19/2011	PROJECT PROPOSALS DUE XML RPC	Quiz 3	Lab 2
5	9/26/2011	Design Patterns The State Pattern Christensen Chapters 9 - 11	Quiz 4	Lab 3
6	10/3/2011	Advanced Unit Testing Mock Objects Test Stubs Christensen Chapter 12 Horstmann Chapter 7	Quiz 5	Lab 4
7	10/10/2011	Review of Mid-term	Mid-term Exam	Lab 5
8	10/17/2011	Compositional Design Christensen Chapters 15 - 18		Lab Makeup
9	10/24/2011	Pattern Catalogue Christensen Chapters 19 - 29	Quiz 7	Project
10	10/31/2011	Event-Oriented Programming Horstmann Chapter 5	Quiz 8	Project
11	11/7/2011	Frameworks Christensen Chapters 30 – 32 Horstmann Chapter 8	Quiz 9	Last day to make-up labs. Project
12	11/14/2011	Database Access	Quiz 10	Project
13	11/21/2011	Dynamic Code Generation and Image Processing	Quiz 11	Project
14	11/28/2011	Domain Specific Languages		Project
15	12/5/2011	Last Class (Review)		Project
	12/12/2011		Final Exam	