

22C:196 FUNDAMENTALS OF WEB PROGRAMMING

Syllabus

Instructor: Teodor Rus

Office: 201J MLH, Phone: 335-0742

Class hours: MWF 9:30–10:20am, 210 MLH

Office hours: MWF 1:30–2:30pm, 201J MLH

Note: This course is given by the College of Liberal Arts and Sciences. This means that class policies on matters such as requirements, grading, and sanctions for academic dishonesty are governed by the College of Liberal Arts and Sciences. Students wishing to add or drop this course after the official deadline must receive the approval of the Dean of the College of Liberal Arts and Sciences. Details of the University policy of cross enrollments may be found at: <http://www.uiowa.edu/provost/deos/crossenroll.doc>

Instructor's note: Copying material from any source and using it without citation as one's self work is an academic dishonesty and, depending on its gravity, can be subject of a penalty within a range of punishments from zero credit for the copied work until exclusion from the program.

1 Rationale

Web programming together with the world wide web grew during the last few years at a stage where almost entire computer technology is web programming dominated. The short term goal of this course is to provide fundamental knowledge one needs in order to engage in web programming applications. On the other hand I believe that web programming reached a stage where its accomplishments need a systematic foundation and therefor the long term goal of this course is to initiate the development of an infrastructure that supports web programming.

2 Topics

Fundamental problems addressed: Conventional programming environments (independent of programming paradigm) did evolve themselves on stages, from ad hoc problem solving approaches to sound and well defined systems, through a process of infrastructure development. This is why I think that currently we are in front of a web programming stage where an infrastructure of web programming is on reach. The elements of this infrastructure are already available and they only need to be synthesized into a final product. They are provided by socket programming, communication protocols, and stand-alone software components and software registry. Socket programming facilitate computer interaction on the network; communication protocol provide the framework for semantic manipulation of

web information; stand alone software components and UDDI and software registry provide the basis for software service organization into language independent software libraries that provide support for process programming where processes interpret messages and perform services they encode. Software architectures built on top of these elements are available in all web applications, and are illustrated by business process execution languages (BPEL), cloud computing, service oriented architectures (SOA), event-driven architectures (EDA), and other. Therefore the development of an infrastructure of web programming is on reach, and, since I see it so clearly, I would like my students to be the first to start building it systematically, and thus, first to take advantage of it.

3 Student enrollments

Both graduate and undergraduate students can benefit from the knowledge I intend to debate in this class and can equally contribute to the class teaching end evolution towards the goal of an infrastructure of web programming. The myriad of industrial and academic web application projects that populate today computing research will provide myriad of ideas for new research development in our environment; myriad of new computer-based industrial and academic companies that pop-up on daily basis will provide a solid basis for jobs. And the satisfaction of understanding where computer based problem solving come from and where does it go to in the future will be a joy for all of us.

4 Methodology

I intend to use a hands-on-application basis while teaching this courses. I already collected the URL-s of the main software packages we need. The topics I intend to cover in Fall 2009 offering of this class class will be grouped into three components: (1). Socket programming and network development, (as seen by ICANN (Internet Corporation for Assigned Names and Numbers)), and the activity involved in moving from the socket addressing structure, now referred to as Internet Protocol Version 4 (IPv4) to the Internet Protocol Version 6 (IPv6). (2). Client-Server programming and its Web support and implementation. I intend to make sure that we understand how browsers and servers that populate today Internet are performing and what kind of programming tools are there that help people develop such amazing application as Google and eCommerce. (3). In order to understand the power of computer based problem solving process using web programming the third group of topics will cover the event-driven architectures illustrated with such applications as airline flight control, anti-money laundering, and event-driven productivity infrastructure.

5 Textbook

The textbook support for this offering of 22C:196, Fundamentals of Web Programming will be: (a) My lecture notes based on Unix Network Programming Volume 1, Addison-Wesley 2004, for socket programming; (2) The book “Programming the World Wide Web, (Fifth Edition), by Robert Sebesta, Addison-Wesley 2009s, for Client-Server programming; (3) My lecture notes based on ”Event Driven Architecture”, Addison-Wesley 2009, for web applications.

6 Student activity and assessment

To achieve its goal we need to rely on lot of software packages which may be freely available but which, for one reason or another, are not at our reach. Hence, one component of the class will be the downloading of this software, installing it on our lap-tops and on the machines available in our labs, and the implementation of the most illustrative projects using this software. This activity will provide each student (or group of students if necessary) with a project that will be used as the class exam. In other words, there will be no formal exams or assignments during this class teaching. Student evaluation and grade will be determined by the student’s involvement in the class through presentations, software download and installation, and web program development to solve specific problems formulated by the instructor or by students themselves.

7 Class Policy

To ensure that student expectations are achievable and at the same time to keep class promise I suggest the following class policy to be set as basis for student grading in this class:

1. Class attendance (no absentees) 25% (a D);
2. (1) above and a self-selected project showing ability to handle sockets programming 50% (a good C)
3. (1), (2) above and four client-based applications chosen from the textbook chapters 1 through 8, 75% (a B)
4. (1), (2), (3) above and four server-based applications self-chosen from the textbook chapters 9 through 12, 90% (an A-)
5. (1), (2), (3), (4) above and a self-selected application (Ajax, Tomcats, WS) 100% (an A+)

However, this class is very heterogeneous: attendees expand on a large spectrum of knowledge and interests (from early undergraduate to late PhD). Therefore, this policy need be amended, allowing students to replace the assessment requirements by special projects they can perform during this class, such as:

- Presents one or more lectures on topics of interest but not scheduled for this offering of the class. Example: CSS, XML;
- Chose to demonstrate Web programming by performing special project's. Example implementing and demonstrating Ajax.
- Chose to demonstrate research on Web programming by approaching and solving various hot problems. **Examples:** Airline Flight Control, Anti-Money Laundering, Event-Driven Productivity Infrastructure.
- Using Service Oriented Architectures to perform Computational Emancipation of Application Domains!

Hence, students can get individual class assessment wavers using the following procedure:

1. Prepare a written document where the student proposes a project (among those enumerated above or another project suggested by the student).
2. Provide a thoroughly description of the work to be performed specifying the amount of grading credits claimed for this work;
3. Provide a schedule of the work to be done proposing the validation method at each step of the schedule;
4. Submit this document to the instructor.

Instructor will study this document and will approve/disapprove it as appropriate.

8 General CLAS Policies

- **Administrative Home:** The College of Liberal Arts and Sciences (CLAS) is the administrative home of this course and governs matters such as the add/drop deadlines, the second-grade-only option, and other related issues. Different colleges may have different policies. Questions may be addressed to 120 Schaeffer Hall or see the Academic Handbooks, www.clas.uiowa.edu/students/academic_handbook/index.shtml.
- **Academic Fraud:** Plagiarism and any other activities when students present work that is not his or her own are academic fraud. Academic fraud is reported to the departmental DEO and to the Associate Dean for Academic Programs and Services who enforces the appropriate consequences, www.clas.uiowa.edu/students/academic_handbook/ix.shtml.

- **Making a Suggestion or a Complaint:** Students with a suggestion or complaint should first visit the instructor, then the course supervisor and the departmental DEO. Complaints must be made within six months of the incident, www.clas.uiowa.edu/students/academic_handbook/ix.shtml#5.
- **Accommodations for Disabilities:** A student seeking academic accommodations should register with Student Disability Services and meet privately with the course instructor to make particular arrangements, www.uiowa.edu/sds/.
- **Understanding Sexual Harassment:** Sexual harassment subverts the mission of the University and threatens the well-being of students, faculty, and staff, www.sexualharassment.uiowa.edu.
- **Reacting Safely to Severe Weather:** In severe weather, the class members will seek shelter in the innermost part of the building, if possible at the lowest level, staying clear of windows and free-standing expanses, (Operations Manual 16.14. i.).