Modern Web Applications

Devdatta Kulkarni

Fall 2016

Agenda

- Introductions
- Course goals and contents
- Prerequisites and Books
- Assignments
- Examinations
- Expectations from you
- Schedule
- Questions

Introductions

- Teaching staff:
  - Me:
    - Name: Devdatta Kulkarni
    - Email: devdatta@cs.utexas.edu
    - Office: GDC 4.812
  - Teaching Assistant:
    - Eddy Hudson
- You:

Web Applications

- Applications vs. Systems
  - Systems provide basic abstractions that are used for building applications
    - Examples:
      - Operating System and bash shell
      - File system and file browser
- Web Applications
  - Applications that are accessible over the Internet (“Web”)
  - Built using abstractions provided by Web systems
    - Web Servers

Characteristics of Web Applications

- Dynamic content
- Persistent storage
- Scalable
- Work across different platforms
- Secure
- Support concurrent requests

Characteristics cont.

- Robust and reliable
- Responsive
- API access
- Implementing current authentication and authorization standards
- Dependent on other such applications

‘Modern’ Web Applications

- Service-oriented Architecture
  - RESTful APIs (*we will learn)*
- Asynchronous
- Cloud-based
  - Use elastic ‘Cloud’ resources
- Built using modern programming tools, techniques and frameworks

Course Goal

- Learn about fundamental concepts in building modern web applications using some of the ‘modern’ tools and technologies
- Once you understand the concepts, changes in technologies won’t be difficult to handle for building next generations of such applications

Course contents

- Topics that we will learn
  - Hypertext transfer protocol (HTTP), Servlets, Session management, Logging, Dependency Injection, Spring framework, Unit testing, Functional testing, REST, XML and Json parsing, JavaScript, Cloud computing paradigms, OpenStack
- What we won't cover
  - Mobile development (Android/iPhone programming)

Prerequisites

- Principles of Computer Systems (CS 439)
  - Understanding of synchronization, race condition, etc.
- Programming experience in Java/C# (CS 312, CS 314)

Books

- Class Notes
- Reference Books
  - Professional Java for Web Applications Featuring WebSockets, Spring Framework, JPA Hibernate, and Spring Security
    - Author: Nicholas S. Williams
  - RESTful java with JAX-RS 2.0 Designing and Developing Distributed Web Services
    - Author: Bill Burke
  - Online tutorials

Course components

- Assignments
- Home works
- Examinations
  - Midterm
  - Final

Assignments

- 6 programming assignments
  - 5 (Java), 1 (Java + JavaScript)
- Submission
  - We will use Bitbucket for assignment submissions
- Points and late policy
  - Each assignment will be out of 100 points
  - You loose 5 points for each late day
- To be done individually
- Assignment grading
  - May require demonstration to TA

Home work

- Home works will be assigned for self-study purpose
- Important to do them, as some of the questions in midterm and final may be based on home works

Examinations

- Midterm
  - Date: TBD
- Final
  - Examination week
- Exam format
  - Open book vs. closed book
    - Will announce later

Grade distribution

- 6 assignments: 12% each
- Midterm: 13 %
- Final: 15 %

Class communication tools

- Canvas
  - Will be used for
    - Announcements
    - Publishing class notes and assignments
    - Grades
- Github examples
  - https://github.com/devdattakulkarni/ModernWebApps
- Piazza
  - Class discussions
    - Will post sign-up link on Canvas

Grade cutoffs

- > 95%: A
- 90 - 95: A-
- 85 - 90: B+
- 80 - 85: B
- 75 - 80: B-
- 70 - 75: C+
- 65 - 70: C
- 60 - 65: C-
- 55 - 60: D+
- 50 - 55: D
- 45 - 50: D-
- < 45: F

Office hours

- Devdatta Kulkarni:
  - After class on Thursday (starting September 1)
    - 7.15pm – 8.30pm in GDC 6.202
- TA office hours:
  - TBD

Schedule

Dates	Tentative topics
Week 1	Class introduction, HTTP
Week 2	HTTP
Week 3	Servlets, Session management, XML Parsing
Week 4	Unit testing, Logging
Week 5	Dependency injection
Week 6	Spring framework
Week 7	Functional testing
Week 8	REST

Schedule

Dates	Tentative topics
Week 9	Midterm
Week 10	Databases, JDBC
Week 11	JDBC, ORM
Week 12	ORM
Week 13	JavaScript
Week 14	JavaScript, Cloud computing
Week 15	Cloud computing
Week 16	Cloud computing, review
Week 17 (Finals week)	Final

Schedule Implications

- Two to three lectures per topic
  - We won’t be able to cover every aspect of the topic
    - Self-study will be required
    - Most of the learning will happen while implementing assignments
- How to succeed in the course?
  - Start early on assignments
  - Don’t loose patience with the code
  - Embrace debugging
  - Write unit and functional tests (we will learn how to do this)

Questions?