|  |  |  |
| --- | --- | --- |
|  |  | ISM 6225 Distributed Information Systems |

SYLLABUS

**Office Hours** : Hour before class and by appointment

**Pre-requisites** : One course in OOP & interest in developing distributed applications

**Resources** : Visual Studio (Community edition recommended) w/ local database, running on a Windows machine. Users with a Mac should install the necessary software on a Windows VM. If needed, please seek an appointment with the TA in the first week of class to complete the required setup.

**Textbooks** : (we will only use a few chapters from each book)

1. (AF) Pro ASP.NET Core MVC 2 [electronic resource] / Adam Freeman, ISBN: 978-1-4842-3150-0 e-book available from the USF library (<http://ezproxy.lib.usf.edu/login?url=http://link.springer.com/10.1007/978-1-4842-3150-0>)
2. (JL) Programming entity framework [electronic resource] : Code First / Julia Lerman & Rowan Miller, ISBN: 9781449323851 (electronic bk.), 1449323855 (electronic bk.), 9781449323844 (electronic bk.), 1449323847 (electronic bk.), e-book available from the USF library

**Optional books (used for the Javascript and OSI model sections, power point slides are available on Canvas)**

1. Business data communications and IT infrastructures (2nd Edition) / Manish Agrawal & Rekha Sharma, ISBN: 978-1-943153-11-4 (AR in outline)
2. A smarter way to learn JavaScript, <http://www.asmarterwaytolearn.com/index.html>

**Course websites**:

Readings : <http://www.ismlab.usf.edu/dis/>

Code repository : <https://github.com/ISM6225>

# Course Objectives

The goal of the course is to teach students full stack development using modern frameworks and understand how the various components of these applications communicate with each other. Students will develop familiarity with using contemporary development tools and deploying full stack applications to the cloud. Given the business analytics focus of the program, projects and exercises will focus on helping students gain familiarity with using data collections and APIs.

Every graduate course in every business school has one primary motivation: why should anyone follow you? The projects and exercises in the class are therefore designed to help students develop confidence in their own technical and leadership capabilities so they can inspire confidence in team members and clients about their ability to develop technology solutions to address business problems.

# Learning Outcomes

1. Reinforce competence with computer programming
2. Reinforce competence with object-oriented programming
3. Learn how to use the MVC framework for full stack web site development
4. Understand and use html and css for web user interfaces
5. Understand and use Java script on web pages for client side interaction
6. Learn how to use data collections
7. Use object-relational mapping to connect web applications to relational data stores
8. Deploy web sites to the cloud
9. Understand the OSI model used application components to access remote resources

# Logistics

Figure : ISM 6225 course module time allocations and objectives

1. The course may be seen as comprised of three modules – Figure 1 shows these modules and their respective time allocations. The first module will reinforce basic programming concepts and introduce algorithmic thinking to solve standard computer programming problems. A benefit of this module is that it introduces students to the approaches adopted by technology intensive organizations in their interviews. Module 2 introduces full stack development, which is useful to help students develop fully operational web applications that use data and analytical models. Module 3 covers the OSI model, the underlying networking technologies, which helps students architect distributed business applications appropriately.
2. All students should have access to an installation of Visual Studio, running on a Windows machine. Users with a Mac should install the necessary software on a Windows VM. If needed, please seek an appointment with the TA in the first week of class to complete the required setup. The Mac version of Visual Studio is not fully feature ready at this time and can therefore not be supported for this class. While MSDNAA provides access to the professional versions of Visual Studio, students may lose access to these versions if they try to update their software upon graduation from the program.
3. Faculty teaching this class (in particular, Prof. Clinton Daniel) are creating video walk-throughs to help students with the hands-on components. These videos are designed to be viewed and followed before class meetings so that class time can be used to help students overcome any technical challenges with implementing the concepts covered. Links to these videos will be available on Canvas.
4. Students may form groups to complete projects. Follow instructions on Canvas for group size. All students should sign up on the groups pre-created for the class, do not create new group sets.
5. Please use the term “ISM 6225” in the subject line of your email (no spaces) to help me filter emails.
6. Readings and assignment deliverables are specified on the course site on Canvas.
7. Deliverables are due by the end of day on the due date (usually this means 11:55pm). The default due days are Sundays, but adaptations are made to accommodate calendar and scheduling issues.
8. Make up opportunities will only be provided for job-related situations and for medical emergencies in the immediate family.
9. I will do my best to try and teach you something useful, not merely certify what you already know. This impacts exam preparation for example, where you should not expect a shortlist of questions to prepare.
10. Subject to logistical challenges, groups whose projects suggest significant effort, will be invited to present their projects to the class in the last week. Please strictly follow the presentation guidelines to make your presentations impressive.
11. On days with low attendance, specific advantage is likely to be provided for exams

# Highly recommended resources on business applications development

If you have anything more than a passing interest in business applications development, you should be very comfortable using the following resources:

* + - 1. Stack Overflow

# Future plans for the course

Angular / React Load balancing and scaling Signal R

Web API Mobile applications Device syncing

# Business Continuity

In the event of an emergency, USF may opt to continue delivery of instruction through methods that include but are not limited to: Canvas, Elluminate, Skype, and email messaging and/or an alternate schedule. It’s the responsibility of the student to monitor Canvas for each class for course specific communication, and the main USF, College, and department websites, emails, and MoBull messages for important general information.

# Presentation

To help build your communication skills, each group will be required to pick a topic from the articles and/ or blogs listed in the presentation section of the course readings website, and present it to the class. Please use the instructions in the presentation assignment for details.

# Grading

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Type** | **Unit weight** | **Total weight** |
| Presentation | Group | 5% | 5% |
| In-class exercises | Individual | 0.5% | 4% |
| Exams | Individual | 15%, 15%, 15% | 45% |
| Assignments | Group | 6%, 12%, 6%, 10%, 6% | 40% |
| Project | Group | 4% | 4% |
| Feed forward | Individual | 1% | 2% |

# Grading Policy

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Total%** | **Grade** | **Total%** | **Grade** | **Total%** | **Grade** | **Total%** | **Grade** |
| >=95  >=90  >=87 | A+(max 10% of class)[[1]](#footnote-1)  A  A- | >=84  >=80  >=77 | B+  B  B- | >=74  >=70  >=67 | C+  C  C- | >=63  >=60  <60 | D+  D  F |

**ISM 6225: Tentative course outline[[2]](#footnote-2),[[3]](#footnote-3)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Week | Module | Lecture topic | Readings/ Class activities | Deadlines |
| 1 – Jan 8 | Module 1 – programming introduction and computational problem solving | Syllabus, IDE introduction, programming introduction, source control | Hello World, Hello World on git | Attendance, calibration |
| 2 – Jan 15 | Programming introduction (contd.) |  |  |
| 3 – Jan 22 | Programming introduction (contd.) |  |  |
| 4 – Jan 29 | Computational problem solving | Searching, sorting, basic list | Assignment 1 – 6225Calculator |
| 5 – Feb 5 | Module 2 – Full stack development | Object-oriented programming concepts introduction, data structures, inheritance |  |  |
| 6 – Feb 12 | MVC components – controllers, action methods, views, models | MyWeb deploy | Assignment 2 – 6225ProblemSolver |
| 7 – Feb 19 | JavaScript, JQuery, charting |  | Exam 1 (programming introduction and computational problem solving) |
| 8 – Feb 26 | Persistence layer - Collections, LINQ, Object-relational mapping |  | Assignment 3 - 6225LookAndFeel |
| 9 – Mar 5 | API consumption (IEXTrading and Azure ML) |  |  |
| Mar 12 | **Spring break** |  |  |
| 10 – Mar 19 | MVC components – routing, model binding |  |  |
| 11 – Mar 26 | MVC misc – Ajax, Form validation, cloud deployment | Azure deploy |  |
| 12 – Apr 2 | Module 3 – OSI model | OSI model |  | Exam 2 (MVC) |
| 13 – Apr 9 | OSI model (contd) |  | Assignment 4 – 6225ActiveCloudSite |
| 14 – Apr 16 | OSI model (contd) |  |  |
| 15 – Apr 23 | OSI model (contd) |  | Assn 5 - OSI 1  Project - Deployed website |
| Finals week |  | Exam |  | Exam 3 (OSI model) |

1. At instructor’s discretion, among the top students in the class [↑](#footnote-ref-1)
2. In the interests of the class, deviations may be made in the coverage of topics as outlined in the tentative course calendar. However, to help plan your calendars for the rest of the semester, assessment and deadline dates will be non-negotiable after the first day of class. Some deadlines may be clubbed together to accommodate schedules. [↑](#footnote-ref-2)
3. Exams 1 and 2 will be 75 minutes each at the start of class [↑](#footnote-ref-3)