Course Syllabus

Course Title
Web Development Using Application Frameworks

Course Number
IS439 – AOG/AOU

Semester
Spring 2021

Instructor
Kevin Trainor

Teaching Assistant
Colleen Heinemann

Class Times
Online meetings will be held on Wednesday from 6:00 PM till 8:00 PM using Zoom. I look forward to joining you there. Please use a headset.

Online Lab Session
On Sunday mornings from 11:00 AM till 12:00 PM, I will be holding an optional Online Lab Session using Zoom. Please drop by to ask a question, to discuss solutions to previous assignments, to get help with the current assignment, to discuss the final project, or just to say hello. I will work with students on a first come, first served basis. Please use a headset.

If you need to arrange a private meeting with me, or if you need to arrange a lab session on a different day/time, please send me an *Individual Meeting Request* via the Service Desk (see *Contacting Instructor or TA* below).

Contacting Instructor or TA
The preferred method for contacting me or Colleen is by entering a request using the Request Center portal for this course. PLEASE, DO NOT send requests to our regular email addresses.

The Service Desk for this course has been implemented using the JIRA Service Management product. If you are new to using the service desk in one of my courses, please visit the Service Desk Introduction for instructions and tips.

On an emergency basis, you may contact me using my mobile phone number: 847-650-9706.

Catalog Course Description
A course in the use and evaluation of Web application frameworks for system architects, designers, and developers.

Detailed Course Description

Today, many substantial Web applications are designed and built using a Web application framework. These frameworks provide standard ways to build and deploy Web applications that can increase developer productivity, increase software quality, and reduce application maintenance. Web frameworks typically use architectural patterns like Model-View-Controller (MVC) to separate code that implements the data model, business rules, and user interface. Frequently, they also employ architectural patterns like a database persistence layer to greatly simplify database coding within the Web application. Other common features of Web application frameworks include built-in support for Web page templating, user authentication/authorization, content caching, URL mapping, session management, Web services, and various content management features.

This course will explore the common features of Web application frameworks in general while providing hands-on experience using Django, a popular open-source Web application framework based on Python. Students will complete a series of coding assignments that build Django design/development/deployment skills while providing insights into standard features of Web application frameworks. In the final project, each student will design and build a full-featured Django Web application to address a problem of her/his own choosing.

The audience for this course includes system architects, designers, and developers who wish to consider a Web application framework as a platform for creating substantial Web applications. These include business applications, digital libraries, institutional repositories, and research data repositories, as well as workflow applications for data collection, data extraction, data cleaning, and data analysis.

Prerequisites

- Experience in creating static Web sites using HTML and CSS
- Experience in Python programming (IS430 or equivalent)
- Experience in creating dynamic Web sites using tools like PHP is helpful but not required.
- Experience in using relational databases is helpful but not required.

Course Outcomes

After completing this course, you should be able to:

1. Identify the advantages of using a Web application framework when designing, developing, and deploying a Web application.
2. Identify common features of Web application frameworks in general.
3. Explain how the specific features of Django correlate with common Web application framework features.
4. Design, code, test, and deploy Web applications that use Django features for:
    a. Models
    b. Templates
    c. URL Mapping
    d. Views
    e. Forms
    f. User authentication/authorization
    g. Deployment

5. Design, code, test, and deploy a full-featured Django application to solve a problem of your own choosing.
6. Evaluate the suitability of a particular Web application framework based upon an anticipated set of functional and non-functional requirements.

Required Texts

Pinkham, A. (2016). Django Unleashed. Pearson Education.
Print ISBN: 9780321985071
eBook ISBN: 9780133812398

Pro Git (2nd Edition)
By Scott Chacon and Ben Straub
Creative Commons Attribution Non Commercial Share Alike 3.0 License
https://git-scm.com/book/en/v2
Please Note: This electronic version is free. It is the version that I recommend.

Technology Requirements

You will be completing coding assignments and conducting your final project using your own computer. I recommend that you use a computer that runs a recent version of Windows 10 or a recent release of macOS. While the software that we will be using for this course does run on Linux computers, there will be substantially less technical support available for Linux. If you want to use a Linux computer for your coursework, please contact me first.

You will be installing the following software on your computers for this class:

- The Anaconda open data science platform for Python
- The PyCharm Professional integrated development environment (IDE) for Python
- The Git version control system
- The SourceTree client for Git

You will be deploying some of your coding assignment solutions to a cloud-based Django hosting facility provided by Python Anywhere. Free accounts are available from Python Anywhere that have all of the capabilities that you will need during the course.

All of the software and services that you will be using are free for your use during this class. I have recorded detailed tutorial videos to help you download, install, and begin using the required software on computers running a recent version of Windows 10 and recent releases of macOS. Links to these resources will be provided in our Weekly Schedule.

Course Schedule

The schedule for this course will be available via our Weekly Schedule at:

https://courseinfo.ligent.net/2021sp/_illinois/is439_aog_aou/index.html

The course schedule is always subject to reasonable change to account for changes in circumstances and to correct errors. When I make changes to the schedule, I will announce them via our Moodle Announcements Forum. Postings to this forum should result in you being sent an email copy of the announcement as well.

Course Elements

1. *Readings*

   Required readings will be assigned from the resources listed above and from other resources that will be identified in the Weekly Schedule. Generally, readings are chosen to accompany our online lecture/discussion session for the week. So, you should expect to complete the readings before class.

   Optional readings will be assigned from time to time. These will typically represent alternate expressions of the same material, or interesting supplementary topics.

2. *Tutorial Videos*

   I have created tutorial videos for the following use cases:

   - There are a number of tutorials that explain how to do activities necessary for the course. These include installing software, reading grading rubrics, submitting assignments, and related activities.
   - Many tutorials are coordinated with the exercises in your coding assignments. They represent a demonstration of the skill you will be expected to use when doing the exercise.

   Because tutorial videos do not duplicate material covered by other resources, I recommend that you always play these recordings. Before playing my tutorial videos, make sure that you have previously played *Tips on Playing My YouTube Videos* to assure that you get the most from your viewing experience.

3. *Coding Assignments*

   There will be weekly Coding Assignments. As mentioned above, coding assignments will often be paired with tutorial videos. These should allow you to complete your Coding Assignment using the same general approach that has been demonstrated in the video.

   A Moodle submission activity will be provided for submitting each assignment. Instructions for each Coding Assignment and a grading rubric will be published in the Weekly Schedule.

   Solutions to Coding Assignments will be posted to our Moodle site before our next online class session. You can expect us to review your solutions and mine at the beginning of our next class.

   A major goal for this course is to build your proficiency in self-evaluation of your work. To build this skill, I will expect you to be able to estimate your grade on each Coding Assignment. The solutions to coding assignments posted to our Moodle site and our review of those solutions during the next class will serve as your primary feedback for the Coding Assignment.

As secondary feedback, your Coding Assignment submissions will be graded and commented upon. This feedback will be published to the Moodle assignment submission activity within 2 weeks.

The grading rubric for Coding Assignments has been designed to promote two important behaviors:

- Submitting your work in a properly named and formatted file. This helps substantially with grading workflow.

- Submitting your work by the week-ending deadline. This assures that you will get the benefit of having tried to solve the problem on your own before seeing the solutions of others.

While separate grading rubric and assignment submission instructions documents will be published, the following is a summary of the coding assignment grading rubric features:

- 10 points will be awarded for submitting a single, properly named and properly formatted file to the proper Moodle assignment submission activity.

- A minimum of 75 points will be awarded for submissions that are submitted on time, and that demonstrate a good faith effort on all parts of the assignment. Late submissions will be awarded 74 points or fewer in this category.

The implication of this grading scheme is that you can expect a score of 85 or higher on all Coding Assignments that meet both of these criteria.

4. *Participation*

Your participation grade will be earned based upon participation points that you will accumulate throughout the semester. The table below lists activities for which you may earn participation points and the points earned for each instance.

Activity	**Points Earned**
A *greetings* post made to the Service Desk for this course by the end of Week 2	10
1 post or reply made in the *Open Discussion* forum.	1
1 speaking contribution during class	2
1 chat contribution during class	1
1 presentation of your Coding Assignment solution during class	5
1 presentation as spokesperson for your a breakout group during class	5

Your participation grade for the course will be calculated at the end of the semester based upon the number of participation points earned. Grading will be done on a curve. A student with the highest number of participation points can expect to earn a grade of 100. A student with the median number of participation points can expect to earn a grade of 85. Students with fewer than 10 participation points can expect to earn a grade of 0.

5. *Framework Evaluation Paper*

Your paper will present the conclusions of your evaluation of the suitability of a candidate **server-side Web application framework** (other than Django) for a set of functional and non-functional requirements which you could reasonably expect to encounter in the workplace. The length of your paper should be 1500 to 2000 words. In addition, you will need to cite a minimum of 10 sources.

Detailed instructions and a grading rubric for the Framework Evaluation Paper will be published separately.

6. Final Project

You will be expected to plan, gather requirements for, design, code, and test a Web application using Django as your Final Project. Detailed instructions and grading rubric for the Final Project will be published separately. The following are highlights from those requirements:

- The Web application should fully demonstrate the Web application framework features covered in the class.
- The Web application should include significant add/change/delete database functionality.
- The Web application should be sufficiently interesting to you that you are likely to continue to develop and maintain after the course is complete.

The Final Project is an individual assignment, NOT a group assignment.

Grading

Basis for Determining Grade
The various components of student work will contribute to the final grade based upon the following percentages:

Participation	10%
Coding Assignments	35%
Framework Evaluation Paper	20%
Final Project	35%

Letter grades will be determined as follows:

A+	97 - 100%;
A	93 - 96%;
A-	90 - 92%;
B+	87 - 89%;
B	83 - 86%;
B-	80 - 82%;
C+	77 - 79%;
C	73 - 76%;
C-	70 - 72%;
D+	67 - 69%;
D	63 - 66%;
D-	60 - 62%;
F	0 - 59%;

ISCHOOL AND UNIVERSITY ACADEMIC POLICIES

Academic Integrity
The iSchool has the responsibility for maintaining academic integrity so as to protect the quality of education and research in our school and to protect those who depend on our integrity. Consequences of academic integrity infractions may be serious, ranging from a written warning to a failing grade for the course or dismissal from the University.
See the student code for academic integrity requirements:
http://studentcode.illinois.edu/article1/part4/1-401/

Statement of Inclusion
http://www.inclusiveillinois.illinois.edu/mission.html
As the state's premier public university, the University of Illinois at Urbana-Champaign's core mission is to serve the interests of the diverse people of the state of Illinois and beyond. The institution thus values inclusion and a pluralistic learning and research environment, one which we respect the varied perspectives and lived experiences of a diverse community and global workforce. We support diversity of worldviews, histories, and cultural knowledge across a range of social groups including race, ethnicity, gender identity, sexual orientation, abilities, economic class, religion, and their intersections.

Religious Observances

In keeping with our Statement of Inclusion and Illinois law, the University is required to reasonably accommodate its students' religious beliefs, observances, and practices in regard to admissions, class attendance, and the scheduling of examinations and work requirements.

If you anticipate the need for an accommodation, please communicate with your instructor in the first two weeks of class. If you are an undergraduate student and your instructor requires an absence letter, you must fill out the Religious Observance Accommodation Request form: https://cm.maxient.com/reportingform.php?UnivofIllinois&layout_id=19 . Other accommodations may be available.

Accessibility Statement

To obtain accessibility-related academic adjustments and/or auxiliary aids, students with disabilities must contact the course instructor and the Disability Resources and Educational Services (DRES) as soon as possible. To contact DRES you may visit 1207 S. Oak St., Champaign, call 333-4603 (V/TTY), or e-mail a message to disability@uiuc.edu .

COVID-19 Statement

In keeping with University and iSchool policy, all students are required to engage in appropriate behavior to protect the health and safety of our community. If you are on campus, this includes wearing a facial covering properly, maintaining social distance (at least 6 feet from others at all times), disinfecting the immediate seating area, and using hand sanitizer.

If you feel ill or are unable to come to class or complete class assignments due to issues related to COVID-19, including but not limited to: testing positive yourself, feeling ill, caring for a family member with COVID-19, or having unexpected child-care obligations, you should contact your instructor immediately and cc your advisor.

Contact Hours

This course will require approximately 54 contact hours.

Last Revised

2020-01-14