FULL SAIL UNIVERSITY.

SERVER SIDE LANGUAGES ~ (SSL)

COURSE SYLLABUS – (CAMPUS & ONLINE)

HOME DEPARTMENT

WDDBS Web Design and Development – Bachelor of Science

COURSE NAME

Server-side Languages (SSL-O and SSL-L)

COURSE INFORMATION

Catalog Course Code:	WDD 353-O	**Three-Letter Course Abbreviation:** SSL
Course Director:	Fialishia O'Loughlin (online)	
Telephone:	(786) 505-2904 - Alternate: 407.679.0100 ext. 3276	
Email + AIM:	foloughlin@fullsail.com	
Instructor Hours:	**Online Office Hours**: Instant Message (before 5pm): foloughlin@fullsail.com; Email (after 5pm) **NOTE:** Also use the FSO Portal to ask general questions	
Lab Assistance	**E-mail:** jbmartin@fullsail.com	
Course Description:	The Server-side Languages Course will examine the benefits of a server-side scripting language to heighten human computer interaction with web content. In this class students will learn how to take their existing knowledge of static-based web content and implement a server-side scripting language to develop a more robust web application. By implementing server-side languages within standards compliant XHTML web pages, students will be able to deploy dynamic content to further the level of interaction between client and server communication.	
Course Topics:	• Server-Side Programming Concepts • Installation and Configuration of a Server-Side Development Environment • Object-Oriented Programming Techniques in Server-Side Development • Data Validation and Sanitization • Relational Database Access	

	• Content Targeting • Content Management • Sessions and State Management • Revision Control

Course Learning Objectives:	Through various components of study and application, students will realize these objectives by completing the following milestones:
	• Understand the ADDIO Web Development Process
	  • Critical Thinking & Problem Solving skills to solve logic problems
	  • Analyze PHP Code
	  • Design a flowchart of coding logic
	  • Develop & debug programming issues
	  • Integrate, test, and optimize a Web Application
	  • *NOTE: ADDIO is a quick & easy web development process (developed by Fialishia O'Loughlin) to help students easily remember the process for developing or updating web development projects. ADDIO is based on many of the standard/industry software development processes, including Software Development Life Cycle (SDLC), WaterFall, SCRUM, Agile, etc..*
	• Learn about server-side scripting
	  • Examine what Server-Side scripting is.
	  • Examine how it works, and why we use them.
	  • Examine various languages, stacks, and the pros and cons of each.
	• Set up a server-side development environment
	  • Install and configure a web server
	  • Install and configure a server-side programming language platform
	  • Install and configure server-side programming tools
	  • Learn how to deploy to multiple server environments, including development, test, and production
	• Apply object-oriented programming concepts to server-side development
	  • Research Model-View-Controller development patterns and how they differ from client-side to server-side
	  • Refactor procedural server-side code to a Class-based system with proper encapsulation
	  • Refactor Class-based code to apply object-oriented concepts of polymorphism and inheritance
	  • Understand varying levels of functional complexity, such as value objects, gateways,

	services, and factories, and how they apply to server-side development • Compare and contrast object-oriented patterns between server-side languages • Use a database for persistent data storage via a server-side programming platform • Install and configure database connectivity tools • Query a database to fetch and display data • Implement standard data create, read, update, and delete functionality via a server-side language • Collaborate with a team via revision control software • Research multiple forms of revision control • Integrate revision control with a development environment • Publish and consume files, including source code, to and from a revision control system
General Outcomes:	• Use the ADDIO Web Development Process to analyze & enhance Web Applications • Improve Critical Thinking & Problem Solving skills to resolve logic problems • Understand the concept and purpose of server-side scripting • Install, configure, read, write, and debug at least one server-side language • Write clean object-oriented code to validate and process web form submission data • Properly insert, update, and retrieve data from a relational database via a server-side language • Integrate individual web pages into a common framework to build cohesive web applications • Use a revision control system to track changes to source code and collaborate with a team
General Education Component:	A full grasp of advanced algebra concepts learned in College Mathematics will help the students write efficient, object-oriented server-side code. The students will use what they learned in English Composition in order to discuss and demonstrate knowledge of the subject matter. Art History's assigned philosophical and scientific readings should prepare students for real-life, ethical problem solving.
Degree Connection:	Server-Side Languages gives students their first opportunity to join together many previously learned technologies into a cohesive web application. It begins during the students' 14th month of the WDD program and centers its focus around applying Object-Oriented Programming skills learned in previous courses, such as PWA-1, PWA-2, and DPW. Students will utilize the foundation learned in Networks and Server Structures to understand Server-Side concepts, and will apply concepts from Web Interface and Usability to build simple applications. It also draws upon HTML and CSS skills learned in the Web Standards courses to

	create views. This course is designed to go hand-in-hand with Database Structures, which is taken concurrently, and gives students the concepts and coding skills needed to connect their server-side web applications to relational databases. This course will prepare students for future courses such as Advanced Server-Side Languages and Advanced Database Structures by teaching basic server-side scripting techniques, database access, and revision control systems. This course will be the students' first taste of back-end development, team collaboration, and end-to-end solutions, which will all be centerpieces of Advanced Server-Side Languages as well as Final Project.

SERVER SIDE LANGUAGES ~ (SSL)

COURSE SYLLABUS – (CAMPUS & ONLINE)

Industry & Job Connection:	This course has been designed to simultaneously teach core competencies and current technologies used in today's web development industry. It instructs students in the current relational database standards and server-side programming best practices used by modern web-connected organizations. The labs and assignments are based on real-world examples and popular web sites. As the majority of web development is heavily integrated with relational databases and server-side code, this course will provide the essential knowledge necessary for modern and future web application development.
Research Component	Research plays an important role in the Server-Side Languages course. Lectures and Labs lay the foundation, but often details need to be researched. In the real world, often there is no time to give step-by-step instructions. Students are encouraged to look up unknown syntax or solutions to common problems. Students are expected to search the web, check documentation, browse or post in forums, and collaborate with classmates to get the job done. Students are also taught to keep up on what is new, since part of being a professional in this fast-moving industry is constant growth. • PHP Documentation, http://www.php.net/ • PHP Class Repository, http://www.phpclasses.org/ • ColdFusion Documentation, http://www.adobe.com/support/coldfusion/ • GitHub, https://github.com/ • Lynda Video Tutorials: http://lynda.fullsail.edu; • W3Schools: http://www.w3schools.com/ • Yahoo: http://developer.yahoo.com/performance/ • Mozilla: http://developer.mozilla.org • Stack Overflow: http://Stackoverflow.com • DocHubio: http://www.dochub.io

SERVER SIDE LANGUAGES ~ (SSL)

COURSE SYLLABUS – (CAMPUS & ONLINE)

Safari Books Online Textbook(s):	**Useful Text Book Resources:** **PHP Cookbook: Solutions & Examples for PHP Programmers** **Safari Link**: http://techbus.safaribooksonline.com/book/programming/php/9781449363741 **MySQL Cookbook, 3rd Edition** **Safari Link:** http://techbus.safaribooksonline.com/book/databases/mysql/9781449374112
Additional Required Resources & Supplies:	○ MAC Computer + Internet Access ○ Install AOL Instant Message (aim.com) & setup account by Week 1; ○ Setup & Test MAMP Server (mamp.com) by Week 1; ○ Install WebStorm or Favorite Code Editor by Week 1; ○ Install Flow-Charting Tool
Email Accounts:	Students are responsible for checking their Full Sail email account regularly, as it will be used for purposes of communication between instructor and student. Students are provided with a personal email account upon admission to the program.
Student Support:	All course materials are accessible via the FSO student portal. For FSO Technical Support, please contact via: • Email: fsosupport@fullsail.com; • Phone: (Toll Free 800#) 877-437-6349 | Ext: 2984 | Hours: 8a.m. – 2a.m. • Visit Daily Ticketing Solution: Zendesk • Visit Student Knowledge Base & Chat: studenthelp.fullsail.edu For help with academic advising, financial aid, etc., contact the Student Support Center (SSC) Team, as follows: • Toll Free 800#: 855-FSHELP2 Ext: 2944 • Ticketing Solution: Zendesk • Email: StudentSupport@fullsail.com • Hours: Monday –Thursday (8:30am - 9:00pm); Friday 8:30am - 8:00pm; Saturday 9:00am - 6:00pm

Learning Activities, Homework, and Exams	**FSO Activities (Non-graded)** Throughout the course, students will have several online and in-class exercises relating to various weekly topics or objectives being discussed. These activities are not graded, however,

students are still responsible for the materials covered and are expected to be review each online activity. If not, you may lose GPS participation points. Most of the activities are short and direct and contain small amounts of actual code. However, some of the activities build on each other. As a result, it is imperative that students understand each activity before moving on to the next.

FSO Homework (Graded)
Students will be developing a wide variety of small-scale programming projects to validate the knowledge they gained through out this course. Most assignments will focus on the core learning objectives taught in the previous learning material or two and are usually due before the end of each week (OR in-class lecture or lab—if applicable). This will allow students to have some time at home to complete the assignment if necessary. Although the grade weight of the course assignments is fairly light, understanding and completing them is a very crucial part of preparing the students for the written and practical exams

GRADING RUBRICS: Please see FSO for a complete list of Grading Rubrics or click here: https://docs.google.com/spreadsheets/d/1tig3rWV0JKMB_Fc9UufM3V6eDTOmZtC0K3K5qquVCd8/edit#gid=0

Projects

- **Syntax Primer, Class Objects, and Revision Control**: Extend programming techniques learned in previous courses by applying server side techniques.
- **Models, Views, Controllers**: Understand encapsulation and abstracting web presentation develop server side functionality away from procedural server-side logic.
- **Database Access**: Architect and deploy a simple database for a server-side web application.
- **Server-side Form Validation**: Ensure proper sanitization of data both incoming and outgoing to ensure the security of application users.
- **Users, Password Encryption**: Demonstrate encryption techniques when handling sensitive data.
- **Cookies, Sessions, Login**: Apply session validation logic through out the application.
- **File Uploading, Manipulation**: Demonstrate file/image upload techniques, manipulate and move files inside the web server.
- **Simple Framework**: Create re-usable development framework.

Grading Scale:

A+	**95 – 100 %**	4.0
A	**90 – 94 %**	3.5
B+	**85 – 89 %**	3.0

	B **80 – 84 %** 2.5 C+ **76 – 79 %** 2.0 C **73 – 75 %** 1.5 D **70 – 72 %** 1.0 F **0 - 69 %** 0
Assessment Criteria: *(Percentage Estimations are subject to change)*	• Professionalism: 10% • Projects/Screen Casts: 55% • Activities: 9% • Discussions: 6% • Practical Final Exam: 20% **PROFESSIONALISM GRADE for Lectures** You are required to follow-along in class and test/debug your own code throughout lecture. Not doing so is unprofessional and WILL affect your final professionalism grade for SSL (----deduction of 10% from your professionalism grade). Thanks in advance for your professionalism! =====> You will receive 10% for every Lecture where you demonstrate AND upload your follow-along activities.
Course Specific Grading Rubrics:	**Screencasts & Student Lecture** Professionalism 10% Applicability to the Given Topic 20% Thoroughness 20% Educational Value 25% Synthesis (Building on previous ideas) 25% **Projects** Professionalism 10%

	File Structure	10%
	Web Standards	10%
	Object-Oriented Architecture	20%
	Encapsulation	20%
	Completeness & Applicability to the Assignment	30%

Late Work Policy:

Visit the FSO to read the official **WDDBS Late Policy**. You can also view it by visiting: https://assethub.fso.fullsail.edu/assethub/LateWorkPolicya579b20e-7cf6-481e-9056-37ec2eeae398_e96d84e7-a94b-46b5-af58-6030724b3a28_f44d7d80-8df1-44e0-ac23-0a8296f8eac3.pdf

Additional Notes on Late Work:

- Make-up assignments will be allowed in cases of well-documented student emergencies ONLY. For student emergencies, it is the student's responsibility to contact the instructor and provide documentation AHEAD OF TIME ----or ----within one week of each due date (unless special arrangements have been previously made).
- If you are going to miss a few days of class, you (or a family member) must notify the instructor either by email or phone prior to the absence--- if at all possible. Failure to notify the instructor in advance may result in you not being allowed to make-up any missed assignments or exams.
- All LATE & UNEXCUSED work will automatically result in a 25% reduction of points earned for each day that you are late. You may also lose GPS points;

Academic Honesty:

Each student is required to follow Full Sail University policy regarding academic honesty. All work submitted by students is expected to be the result of the student's individual thoughts, research, and self-expression unless the assignment specifically states "group project." Any act of academic dishonesty will be handled in accordance with Full Sail University policy. See FSO.

Global Professionalism System (GPS):

Easy… be professional & treat this class like its part of your career. Your paycheck is your grade. Review all online activities & be on time to all lectures (if applicable). Complete all non-graded and graded assignments, meet your homework deadlines, be resourceful, stay polite & courteous and this should be an easy 10%. If you have any emergencies or conflicts (work, family, etc.), it is YOUR responsibility to communicate & contact me at least 24 hours ahead of time. Thanks for your professionalism!

Attending the each "LIVE" Session during Weeks 1 & 3 is optional; however, if you cannot attend,

SERVER SIDE LANGUAGES ~ (SSL)
COURSE SYLLABUS – (CAMPUS & ONLINE)

Weekly Screen Casts: (*Online Students Only*)	you MUST post a 50-word summary of what you learned on the Discussion Board for the corresponding GoToTraining session. You will earn 2% towards your general grade if you attend. OR, you can earn your 2% by watching the archived session "AND" posting your summary on the Discussion Board. NOTE: Since all other Screen Casts from Weeks 2 & 4 will be pre-recorded, everyone must watch & summarize the archived sessions to earn your 2%. Otherwise, you may also lose GPS points. Thanks for your professionalism!
Strategies for Successful Learning:	My greatest desire is that all of you succeed in this class, however, one thing to remember about homework is ----ask questions & don't get behind. Again, the importance of doing your homework cannot be stressed enough. Please keep the following strategies in mind over the next four (4) weeks: In addition, please be aware that: • **Get & Stay Organized!** Get started on the right track by completing Steps 1-6 under the "Important Announcements" Tab on FSO to download the required files so that you can get and stay organized. • **Client Satisfaction & Professionalism are Important!** In the real world, being on time & meeting your Client or Manager's deadlines are important to having a success career. Logging into the Full Sail Portal is NOT sufficient enough to obtain a good grade. For each week of class, your grade will be determined by class attendance, completion of your assignments & activities (submitted via FSO and/or GitHub.com when required) and participation in discussions for that week's class by a set due date. • **Schedule Your Study Time.** You cannot understand the topics presented without making time for careful and timely study sessions. If you are having difficulty with a particular topic, by all means please contact me. But FIRST make sure you have taken the time to DO YOUR PART! Dedicate and schedule some time to complete and work through all non-graded activities so that you are prepared for the upcoming graded assignments. • **Learning through repetition AND struggle is normal**. Coding & trouble-shooting errors for some will not come easy, so perseverance is key. You may get frustrated when error messages or bugs occur in your program. This happens to everyone----even the most experienced. Stay focused and code in a modular fashion. Use good debugging techniques, fix one thing at a time, and don't give up! • **Try 3 before ME**: Don't be afraid to ask questions, but learn to be resourceful by researching the problem first. Start by reaching out to the Internet (see "additional resources"), your textbook, a classmate, or the Lab Assistant before contacting me. Training yourself to be resourceful will benefit you greatly in the real world. Once you've tried 3 other resources, feel free to send me an IM or use E-mail and FSO to send me questions. • **Use revision control effectively**: • Commit each small item of progress instead of in a large mass.

	• Merge code often to avoid duplication of work and regressions. • Communicate with teammates to avoid merge conflicts. • **Use effective problem solving**: • Do not try to fix or build everything at once. • Solve or build one small piece at a time. • Don't get hung up on the things you can't do—focus on the things you can do. • **Always get at least three opinions for any research:** • Ensure that research sources are current. • Try to find a dissenting or differing opinion to ensure both sides of each issue are understood. • Writing test code can be an effective learning tool. • Sometimes you just have to jump in and see how it works out. • *Have fun & I look forward to programming with you over the next 4 weeks!*
Portfolio Assignment:	Course portfolios provide recruiters with a demonstrable means to assess student competencies. As a result, SSL students use class work to setup assignments online as part of a professional portfolio.
Instructor Insights:	My greatest desire is that all of you succeed in this class, however, one thing to remember about homework is ----ask questions & don't get behind. Again, the importance of doing your homework cannot be stressed enough. In addition, please be aware that: • Logging into the Full Sail Portal is NOT sufficient enough to obtain a good grade. For each week of class, your grade will be determined by class attendance, completion of your assignments & activities (submitted via FSO when required) and participation in discussions for that week's class. • You cannot understand the topics presented without making time for careful and timely study sessions. If you are having difficulty, by all means please contact me. But FIRST make sure you have taken the time to DO YOUR PART! The best way to contact me is via iChat or E-mail, as listed on this syllabus: foloughlin@fullsail.com. You can also contact me via phone using the contact info listed at the beginning of this document. • Have fun & I look forward to programming with you over the next 4 weeks!
Course Rubrics:	Please check FSO "Course Materials & References" or click the link below for course-specific grading rubrics and weekly calendar of assignments. This syllabus, all grading rubrics and weekly calendar activities may be altered (at the instructor's discretion) during the course of this term. It is the responsibility of the student to make adjustments as announced. Click link below to view: https://docs.google.com/spreadsheets/d/1tig3rWV0JKMB_Fc9UufM3V6eDTOmZtC0K3K5qquVCd8/e

	dit#gid=0