

CSCE 242 - Client-Server Computing

1. Course number and name: **CSCE 242 - Client-Server Computing**

2. Credit: 3-hrs; Contact: 3 lecture periods of 50 minutes or 2 periods of 75 minutes per week

3. Instructor: Jose Vidal

4. Textbook: Selected readings from:

- a. Douglas Crockford, "JavaScript: The Good Parts", O'Reilly press, 2008.
- b. Chuck Musciano and Bill Kennedy, "HTML & XHTML: The Definitive Guide", O'Reilly press, 2006.
- c. Dan Sanderson, "Programming Google App Engine", O'Reilly press, 2009.
- d. David Flanagan, "JavaScript: The Definitive Guide", O'Reilly press, 2006.
- e. Mark Lutz, "Programming Python", O'Reilly press, 2011.
- f. Eric A. Meyer, "CSS: The Definitive Guide", O'Reilly press, 2006.
- g. Leonard Richardson and Sam Ruby, "Restful Web Services", O'Reilly press, 2007.

5. Specific course information

- a. Catalog description: Web technologies to support client-server computing. Implementation of client-server applications.
- b. Prerequisites: CSCE 190, 146
- c. Lower Division CSCE elective course.

6. Specific goals for the course

- a. Specific outcomes of instruction are that students will be able to:
 1. Design and build the datastore for a web application.
 2. Design and build the server-side code for a web application.
 3. Design and build the client-site code for a web application.
 4. Deploy a 3-tier web application.
- b. Relation of course outcomes to Student Outcomes: CE: see page 2; CS & CIS: see page 3

7. Topics covered and approximate weight 56 14

1. Web history: HTTP, firebug (4 hours)
2. Google app engine. Python. HTML (8 hours)
3. App engine datastore: a noSQL datastore. (8 hours)
4. Django templates. CSS (6 hours)
5. JavaScript: the language. Functional programming. Prototype inheritance. (8 hours)
6. JavaScript and the DOM. jQuery. (8 hours)
7. App engine memcache. (2 hours)
8. HTML5: localStorage. jQuery for iPhone. (2 hours)
9. Canvas. SVG (2 hours)
10. App engine XMPP (2 hours)
11. REST (2 hours)
12. Mashups: facebook, delicious, flickr, and others (2 hours)
13. Another way: LAMP on EC2 (2 hours)

Computer Engineering

Relation of Course Outcomes to EAC Student Outcomes*

Course Outcomes (CE)	Student Outcomes											
	(a) apply knowledge of mathematics, science, and engineering	(b) design and conduct experiments, ... interpret data	(c) design a system, component, or process to meet desired needs ...	(d) function on multidisciplinary teams	(e) identify, formulate, and solve engineering problems	(f) an understanding of professional and ethical responsibility	(g) communicate effectively	(h) the broad education to understand and the impact of engineering solutions ...	(i) a recognition of the need for, and an ability to engage in life-long learning	(j) a knowledge of contemporary issues	(k) use the techniques, skills, and modern engineering tools	(CE) demonstrate knowledge of discrete mathematics [CE]
Criteria	a	b	c	d	e	f	g	h	i	j	k	CE
1. Datastore	2		3		3						3	
2. Server-side code.	2		3		3						3	
3. Client-side code.	2		3		3						3	
4. Deploy		1			3	1				1	3	
5.												

* 3 = major contributor, 2 = moderate contributor, 1 = minor contributor; blank if not related

Computer Science & Computer Information Systems

Relation of Course Outcomes to CAC Student Outcomes*

Course Outcomes (CS & CIS)	Student Outcomes											
	All									CS		CIS
	(a) apply knowledge of computing and mathematics appropriate to the discipline	(b) analyze a problem, and identify and define the computing requirements ...	(c) design, implement, and evaluate a computer-based system, ...	(d) function effectively on teams to accomplish a common goal	(e) An understanding of professional, ethical, legal, ... responsibilities	(f) communicate effectively with a range of audiences	(g) analyze the local and global impact of computing on ... society	(h) Recognition of the need for ... continuing professional development	(i) current techniques, skills, and tools necessary for computing practice	(j) apply mathematical foundations, algorithmic principles, and CS theory ...	(k) apply design and development principles	(j) An understanding processes that support the information systems environment.
Criteria	a	b	c	d	e	f	g	h	i	j	k	j
1. Database	2	2	3							1	3	
2. Server-side code	2	2	3							1	3	
3. Client-side code	2	2	3							1	3	
4. Deploy					1							1
5.												

* 3 = major contributor, 2 = moderate contributor, 1 = minor contributor; blank if not related