

San José State University
College of Science / Department of Computer Science
CS-160, Software Engineering, Section 2, Fall 2017

Course and Contact Information

Instructor:	Robert Bruce
Office Location:	Duncan Hall, DH-282
Telephone:	(707) (483-9340)
Email:	Robert.Bruce@sjsu.edu
Office Hours:	Mondays and Wednesdays, 4pm-5pm, or by appointment
Class Days/Time:	Monday and Wednesday, 7:30PM - 8:45PM
Classroom:	SCI-311
Prerequisites:	CS-146 and CS-151 (with a grade of "C-" or better in each); CS-100W (with a grade of "C" or better) or instructor consent.

Course Format

Technology Intensive, Hybrid, and Online Courses

This class is technology intensive. You will need an Intel x86 or compatible laptop or desktop machine. We will be working in a Linux development environment. You will be using command line tools such as Make, gcc, etc. You will be creating and managing SQL tables in Postgres. You will also be creating static and dynamic web pages using the Apache web server. You will need an Internet connection to access materials on the course website and Canvas. I will be providing a 120GB solid state drive configured with Linux for you with a USB3.0 or two USB2.0 interfaces. All software and software development tools we will be using are open source. You may use a variety of languages for your programming project including Java, C, C++, PHP, Perl, Python, etc.

Faculty Web Page and MYSJSU Messaging

Course materials such as syllabus, handouts, notes, assignment instructions, etc. can be found on [Canvas Learning Management System course login website](http://sjsu.instructure.com) at <http://sjsu.instructure.com>. You are responsible for regularly checking with the messaging system through [MySJSU](http://my.sjsu.edu) at <http://my.sjsu.edu> to learn of any updates.

Course Description

Software engineering principles, software process and process models, requirements elicitation and analysis, design, configuration management, quality control, project planning, social and ethical issues. Required team-based software development, including written requirements specification and design documentation, oral presentation, and tool use. Prerequisite: CS 146, CS 151 (with a grade of "C-" or better in each); CS 100W (with a grade of "C" or better) or instructor consent. Computer Science and Software Engineering Majors only.

Course Learning Outcomes (CLO)

Upon successful completion of this course, students will be able to:

1. Design a scalable, relational database schema to support a computer vision face detection pipeline.
2. Apply the Postgres application programming interface in software applications.
3. Use the Active Appearance Modelling (AAM) algorithm from OpenFace to locate facial features.
4. Use Delaunay triangulation from the OpenCV library to create a face mesh.
5. Use the FFMPEG library to convert video into a series of still images.
6. Use the FFMPEG library to extract video metadata information (frames per second, number of frames, etc.).
7. Use named pipes, fork(), and exec() to implement inter-process communication in the face detection pipeline.
8. Create dynamically-driven web page content using Apache.
9. Apply software engineering principles through the entire development lifecycle from design, implementation, testing, release, and maintenance.

Required Texts/Readings

Textbook

Beginning Software Engineering by Rod Stephens. ISBN-13: 978-1118969144.

I've put three personal copies of this book on course reserves for twenty-four hour checkout from the SJSU King Library. Alternately, you may purchase this book from an online bookstore such as Amazon, Barnes and Noble, etc.

Other Readings

PostgreSQL: Up and Running by Regina Obe and Leo Hsu. This book is available electronically from San Jose Public Library at http://discover.sjlibrary.org/iii/encore_sjpl/record/C_Rb5465930_SPostgreSQL%3A%20Up%20and%20Running_Orighresult_U_X7?lang=eng&suite=sjpl

Learning OpenCV: Computer Vision with the OpenCV Library by Gary Bradski and Adrian Kaehler. This book is available electronically from San Jose Public Library at <http://0-proquest.safaribooksonline.com/catalog.sjlibrary.org/?uiCode=califa&xmlId=9781491937983>

Advanced programming in the UNIX environment by W. Richard Stevens and Stephen A. Rago. This book is available in hardcopy from San Jose Public Library at http://discover.sjlibrary.org/iii/encore_sjpl/record/C_Rb4553870_SAdvanced%20programming%20in%20the%20UNIX%20environment_Orighresult_U_X7?lang=eng&suite=sjpl

Other technology requirements / equipment / material

You will be provided with a blank 120GB solid state drive with a USB3.1 interface. On this drive, you will configure and install Linux Mint in a graphical X-windows environment along with a host of GNU

development and programming utilities, a Postres database server, the OpenCV library, OpenFace library, and FFMPEG library. Once configured, you will be able to use this drive specifically for this class with all the tools necessary to create a computer vision pipeline.

Library Liaison

Anamika Megwalu, anamika.megwalu@sjsu.edu

Course Requirements and Assignments

There are six programming assignments in this course. The programming assignments are designed as building blocks which lead to a culminating final project: web-based, computer vision face detection system. Detailed instructions and specifications for each programming assignment will be posted to [Canvas Learning Management System course login website](http://sjsu.instructure.com) at <http://sjsu.instructure.com>. Students are strongly encouraged to ask the instructor for clarification on each programming assignment specification. There will be an in-class oral presentation and demonstration of the final project as well as a written final project report. Lastly, there will be on midterm and one final exam.

Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally 3 hours per unit per week with 1 of the hours used for lecture) for instruction or preparation/studying or course related activities including but not limited to internships, labs, clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus.

Final Examination or Evaluation

There will be a final exam on **Wednesday, December 13 in SCI-311 at 7:45PM**. The final exam will be based on in-class lecture material and material from our textbook. The final exam will be comprised of 15 multiple choice questions. This is a closed note, closed-book exam. No electronic devices (including mobile phone) may be used during the exam. The final exam is worth 15% of your overall grade.

Grading Information

Each programming assignments will include detailed specifications. The programming assignments will be graded collectively as a group. This means everyone in the group gets the same grade for each programming assignment. I expect each of your programs to compile (if applicable) or interpret (where applicable) without syntax errors or logic errors. Points will be deducted for programs that do not meet the specifications and/or programs containing syntax errors.

The programming assignments will be due upon completion of the project on **December 6, 2017**. Only one member of the group needs to submit the programming assignments to the [Canvas Learning Management System course login website](#) at <http://sjsu.instructure.com>. All members of the group will receive the same grade for these assignments.

Each member of the project will individually write and submit a final project report. This is an individual assignment, not a group assignment. The final project report will be due **December 11, 2017**. Everyone must submit their own final project report to the [Canvas Learning Management System course login website](#) at <http://sjsu.instructure.com>. The final project report should include a title page followed by a two-page report. The report should include the following:

- Your contributions to the project.
- An overall assessment of the success and failures (if any) on the project. This is an opportunity to discuss time management issues, communication issues among team members, etc. It's an opportunity for you to reflect upon the project and how well the team performed.

In the final project oral presentation, you will discuss your project as a team and demonstrate it to the class. Each member of the group is expected to speak about their contributions on the project. After each member of the group has spoken, the entire group will present a demonstration of the project itself (i.e. run the project).

Determination of Grades

Grade breakdown

Percent range	Grade
97% to 100% inclusive	A+
93% to 96% inclusive	A
90% to 92% inclusive	A-
87% to 89% inclusive	B+
83% to 86% inclusive	B
80% to 82% inclusive	B-
77% to 79% inclusive	C+
73% to 76% inclusive	C
70% to 72% inclusive	C-
67% to 69% inclusive	D+
63% to 66% inclusive	D
60% to 62% inclusive	D-
Below 60%	F

Grade breakdown

ASSIGNMENT OR EXAM	POINT VALUE
Biographical writeup*	5
Programming Assignment 1: Database schema**	5
Programming Assignment 2: Web-based user login**	5
Programming Assignment 3: Extract still images from video**	10
Programming Assignment 4: Determine facial landmarks**	10
Programming Assignment 5: Track eye pupils**	10
Programming Assignment 6: Web-based display results**	10
Final Project Oral Presentation*	5
Final Project Report*	10
Midterm Exam*	15
Final Exam*	15
TOTAL POINTS	100

* Graded as an individual

** Graded as a group (each member of the group gets same grade)

Incomplete assignments

Points will be deducted for incomplete or partially working solutions. Point penalties vary with each assignment but are listed in the programming project assignment.

Late assignments

Programming assignments submitted after their specified due date will be considered late and subject to minimally 50% loss in points (additional points will be deducted for incomplete projects as noted above).

Classroom Protocol

Regular class attendance is highly recommended. Please plan to arrive early or on-time. Arriving to class late is disruptive to students and the instructor. Students are responsible for knowing all materials covered through in-class lectures and assigned readings. Please be mindful of fellow students and the instructor by not talking on mobile phones during instruction. Students are expected to leave the class quietly in the event they must use their mobile phones.

University Policies

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' [Syllabus Information web page](http://www.sjsu.edu/gup/syllabusinfo/) at <http://www.sjsu.edu/gup/syllabusinfo/>

CS-160 / Software Engineering, Fall 2017, Course Schedule

Course Schedule

Week	Date	Topics, Readings, Assignments, Deadlines
1	Wednesday, August 23	Introduction: Course goals and objectives. Lecture: What is software engineering? Readings: Chapter 1 of <i>Beginning Software Engineering</i>
1	Monday, August 28	In-class assignment: Your programming experience and biographical information
2	Wednesday, August 30	Postgres RDBMS (Relational Database Management System)
2	Monday, September 4	NO LECTURE TODAY. CAMPUS CLOSED.
3	Wednesday, September 6	Document management Readings: Chapter 2 of <i>Beginning Software Engineering</i> <i>Note: Last day to drop course is September 6!</i>
3	Monday, September 11	FFMPEG library and systems programming tools
4	Wednesday, September 13	Project management Readings: Chapter 3 of <i>Beginning Software Engineering</i>
4	Monday, September 18	Processing FORM data from an HTTP POST
5	Wednesday, September 20	Requirement gathering Readings: Chapter 4 of <i>Beginning Software Engineering</i>
5	Monday, September 25	<i>Conceptual design reviews and prototype demos</i>
6	Wednesday, September 27	<i>Conceptual design reviews and prototype demos</i>
6	Monday October 2	High and low level design Readings: Chapters 5 and 6 of <i>Beginning Software Engineering</i>
7	Wednesday, October 4	Secure session management with HTTP
7	Monday, October 9	Programming the Postgres C API (Application Programmer Interface)
8	Wednesday, October 11	Development Readings: Chapter 7 of <i>Beginning Software Engineering</i>
8	Monday, October 16	MIDTERM EXAM
9	Wednesday, October 18	Lecture: Active Shape Modelling (ASM) and Active Appearance Modelling (AAM)
9	Monday, October 23	<i>Application design reviews</i>
10	Wednesday, October 25	<i>Application design reviews</i>
10	Monday, October 30	Testing Readings: Chapter 8 of <i>Beginning Software Engineering</i>
11	Wednesday, November 1	Eye pupil tracking
11	Monday, November 6	Deployment, Metrics, and Maintenance Readings: Chapters 9, 10, and 11 of <i>Beginning Software Engineering</i>
12	Wednesday, November 8	Drawing with OpenCV
12	Monday, November 13	Delaunay triangles

Week	Date	Topics, Readings, Assignments, Deadlines
13	Wednesday, November 15	Predictive models Readings: Chapter 12 of <i>Beginning Software Engineering</i>
13	Monday, November 20	<i>Code reviews</i>
14	Wednesday, November 22	NO LECTURE TODAY. I will be in my office during office hours.
14	Monday, November 27	<i>Code reviews</i>
15	Wednesday, November 29	Iterative models Readings Chapter 13 of <i>Beginning Software Engineering</i>
15	Monday, December 4	RAD Readings Chapter 14 of <i>Beginning Software Engineering</i>
16	Wednesday, December 6	<i>Application demonstration</i> DUE: Programming Assignment 1 DUE: Programming Assignment 2 DUE: Programming Assignment 3 DUE: Programming Assignment 4 DUE: Programming Assignment 5 DUE: Programming Assignment 6
16	Monday, December 11	<i>Application demonstration</i> DUE: Final Project report
Final Exam	Wednesday, December 13	FINAL EXAM at 7:45pm in SCI-311