CS74.42: Introduction to Game Coding Sections 5363 and 5824, Spring 2020 Course Syllabus

Instructor: Ethan Wilde, ewilde@santarosa.edu

Course Description

This course introduces students to the design, development, and coding of simple graphical computer-based games. During the course, students will be introduced to various game engines and development environments. Students will gain experience working individually and in a team environment. Emphasis is placed on engaging players through compelling application of game mechanics, dynamics, and aesthetics, as well as on playtesting and iterative development to ensure user-centered design goals are met.

Recommended Preparation: Course completion of CS 110A and/or programming experience

Prerequisites: None

Whether you want to become a member of a professional AAA game development team, or just want to try your hand as an independent game developer, mastery of the game development process, including coding, is essential to those goals. We will work with the ECMA-compliant JavaScript language and real-world tools to develop proficiency in the creation of browser-based, mobile and console games. This course will follow the Transformational Process in development of final game projects.

Student Learning Outcomes

Students will be able to:

1. Understand and apply systems-level thinking and game development methodology best practices to the design and development of simple graphical computer-based games.
2. Develop team-based game creation skills that cover game logic sequencing; storyboarding; artistic and technical coordination; and understanding of technical requirements and limitations.

Upon completion of the course, students will be able to:

1. Explain factors that motivate game players and apply those to the design of games that satisfy the needs of players. This includes the design of a logical sequence of game rules, play flow, and interaction opportunities.

2. Develop game coding skills to implement simple versions of games they design. Desired functions include: graphical, interactive user interface; sprite creation, collision detection, applied simulation of physical forces; classes, object instantiation; and modular implementation.
3. Determine that games are of measurable high quality and error-free through playtesting, iterative development, and an adherence to best practices in quality assurance, including a proper and documented testing process.
4. Work independently and in teams using an iterative process to prepare and create Game Design Documents, storyboards, game assets and scripting to create game simulations.
5. Utilize the concepts of game mechanics, dynamics, and aesthetics to discuss game play of games of different genres on varied platforms.

Topics and Scope:

I.   Game Development Fundamentals
    A.   Introduction to systems thinking
        1.   Flow control and diagramming
        2.   Visual scripting systems
    B.   Introduction to scripting
        1.   Variables and data types
        2.   Expressions and operators
        3.   Control structures
        4.   Functions
        5.   Objects and classes
        6.   Events and triggers
    C.   Mathematics and physics fundamentals
        1.   Cartesian coordinate systems
        2.   World space, object space, camera space
        3.   Vectors, forces and physics simulations
    D.   Development methodologies
        1.   Unified and agile processes
        2.   Iterative design
        3.   Human-centered design
        4.   Design patterns
        5.   Project management and quality assurance
II. Considerations of Game Creation
    A.   Types of games
    B.   Game mechanics, dynamics, and aesthetics
        1.   Mechanics of gameplay
        2.   Dynamics of gameplay
        3.   Aesthetics of gameplay
    C.   Motivational and emotional aspects of playing games
    D.   Generic game design rules
    E.   Technologies, platforms, tools
    F.   Game development frameworks and engines
III. Game Development Process

       A.   Setting goals for the game
       B.   Developing a storyboard and designing gameplay
       C.   Drafting a Game Design Document (GDD)
       D.   Iterative prototyping
       E.   Stakeholder feedback loop and playtesting
       F.   Features, functions, and program components
              1.   Game loop
              2.   Assets
              3.   User input
              4.   Sprites and collisions
              5.   Audio representation
       G.   Game rules
       H.   Navigation and wayfinding
       I.   Graphical user interfaces
IV.   Supporting Disciplines
       A.   Game playtesting and validation
              1.   Quality assurance testing plan
              2.   Bug tracking and resolution
       B.   Working as a team
       C.   Productization and packaging
       D.   Art work

Assignments:

1. Read approximately 25-30 pages a week
2. Prepare 1-2 written Game Design Document(s) (GDD) that closely mirror the documentation process used in the field. (3-7 pages each)
3. Regular group discussion contributions (0 - 12) such as:
   A. Play logic and navigation design
   B. Storyboarding, player motivation
   C. Theories of game development
   D. Design and implementation of game logic (rules and programmatic navigation)
   E. Use of design patterns
   F. Graphical user interface design
   G. Implementation technologies and validation
4. Game creation and scripting assignments that solve particular technical challenges (6 - 12)
5. Midterm and final examinations that evaluate critical thinking skills such as:
   A. Game requirements analysis and validation
   B. Game storyboard design and implementation
   C. Software games, techniques, and technologies
   D. Tools and techniques used in the software industry
6. A final class project that includes topics such as:
   A. Selection of a viable, realistic project
   B. Interviews with players and project stakeholders to collect requirements and risks

C. Creation of a comprehensive software-based game
D. Summary of playtesting findings and suggested improvements for future versions

Class Meetings

Spring 2020 Schedule

Class Delivery	Day and Time	Platform
Online or Face to Face Meetings	Weeks start on Mondays	Canvas shell or Kunde 152
Live weekly Web conference (optional) https://cccconfer.zoom.us/j/961817861	Mondays, 12:00pm - 3:50pm	Zoom

All class materials for each module will be released online in Canvas on Mondays throughout the entire semester. A weekly live meeting will be held on Mondays, 12:00pm - 3:50pm in Kunde 152.

Instructor Contact

Ethan Wilde

Email: ewilde@santarosa.edu

Phone: 707-527-4855

Spring 2020 Office Hours
January 14 – May 14, 2020

Day	Time	Location
Wednesdays (in office)	10:40am - 11:40am	Maggini 2937 (inside Computer Studies Dept office on 3rd floor)
Thursdays (online)	10:40am - 4:40pm	Online: Email ewilde@santarosa.edu or Skype *ethanwilde*

» Reserve a future office hour appointment

I typically respond to emails within 48 hours, weekends excepted. I never respond on Sundays.

Course Web Site

Students will use the Canvas course web site for assignment instructions, submitting assignments, viewing classmates' work, sharing resources, and viewing grades. *The Google Chrome browser is recommended for viewing the Canvas-powered course site. Internet Explorer is not recommended.*

Textbooks

Eloquent JavaScript (3rd)
Marijn Haverbeke
978-1593279509 (ISBN 13)
[Free PDF eBook available](#)

[***The Transformational Framework*** (1st)](#)
Sabrina Culyba
978-1387895274 (ISBN 13)
[Free PDF available](#)

All required textbooks are available online without cost. If you would like a printed copy, you can locate and order textbooks online via the [SRJC Bookstore](#).

Equipment

- **A personal computer**, either at home, work or on the Santa Rosa or Petaluma campuses.

Required Software + Services

- **Internet access**
- **Web browser**
    - [Google Chrome](#) strongly recommended
    - Support for the Adobe Flash browser plugin
- **Coding education site**
    - [CodeCombat](#) free account required for all students.
- **Cloud hosting + development services**
    - [Repl.it](#) IDE (Integrated Development Environment) required for all students, starting Week 2, for hosting class assignments. Repl.it will provide a text editor and file transfer support without any additional software needed. *Complete the hosting survey to get your free account.*
    - [PlayCanvas](#) WebGL Game Engine and IDE required for all students, starting Week 10, for hosting 3D-based class assignments. PlayCanvas will provide a text editor and file transfer support without any additional software needed. *You may*

create a free account at PlayCanvas or use Repl.it to host your PlayCanvas projects.
- o [MIT Scratch](#) simple coding IDE that is optional for all students. Scratch will provide a code editor without any additional software needed. *You may create a free account at MIT Scratch.*
- **2D Graphics software** such as:
  - o Adobe Photoshop, part of a [Creative Cloud](#) subscription
  - o [Gimp](#) open source application
  - o [Pixlr](#) browser-based image editor
- **Spritesheet and tilemap editing software** such as:
  - o Tiled map editor found online at http://www.mapeditor.org/
  - o TexturePacker texture/sprite sheet editor found online at https://www.codeandweb.com/
- **Phaser.js JavaScript library, version 3.x:**
  - o Found online at [phaser.io](#)
- **JavaScript code validator**:
  - o http://esprima.org/demo/validate.html
- **PDF display software** such as:
  - o **[Adobe Reader](#)**

Optional Software

The additional software listed below is often used for game development. Our IDEs – the Repl.it cloud-based IDE for our work with Phaser.js, and the browser-based Scratch software – both provide code editors and file transfer support without any additional software needed.

- **3D modeling software** including:
  - o [Blender](#)
  - o [Trimble Sketchup Pro](#) ($49 education license)
- **Enterprise-class game development engine software** including:
  - o [Unity3D, Personal Edition](#)
  - o [Unreal Engine](#)
- **Text editor** such as:
  - o [TextWrangler](#) (Mac OS only)
  - o [Brackets](#) (Windows, Mac OS)
  - o [Sublime Text](#) (Windows, Mac OS, Linux)
- **Additional Web browsers** including:
  - o [Mozilla Firefox](#)
  - o Apple Safari (Mac OS only)
  - o Microsoft Edge (Windows 10 only)
- **File Transfer Protocol (FTP) software** such as:
  - o [CyberDuck](#) (Mac OS and Windows, free)
  - o [Fetch](#) (Mac OS only)
  - o [WinSCP](#) (Windows only)

Important Dates

Day Class Begins: Monday, January 13, 2020

Day Class Ends: Friday, May 22, 2020

Last Day to Add without instructor's approval: Sunday, January 19, 2020

Last Day to Drop with refund: Sunday, January 26, 2020

Last Day to Add with instructor's approval: Sunday, February 2, 2020

Last Day to Drop without a 'W' symbol: Sunday, February 2, 2020

Last Day to Opt for Pass/No Pass: Sunday, February 23, 2020

Last Day to Drop with a 'W' symbol: Sunday, April 19, 2020

Dropping the Class

If you decide to discontinue this course, it is your responsibility to officially drop it. A student may be dropped from any class when that student's absences exceed ten percent (10%) of the total hours of class time. It is strongly advised that if you need to miss more than one class/homework deadline in a row that you contact the instructor to avoid being dropped from the class.

Attendance

Students who fail to complete the requirements of the first and second class modules will be dropped by the instructor. **Students enrolled in the online section 5363 must view and participate in online materials released each week in the Modules section of the course Canvas website. Students enrolled in the face-to-face section 5824 must attend class in person each week, as well as view and participate in online materials released each week.**

Pass-NoPass (P/NP)

You may take this class P/NP. You must decide before the deadline, and add the option online with TLC or file the P/NP form with Admissions and Records. With a grade of C or better, you will get P.

You must file for the P/NP option by February 23, 2020. Once you decide to go for P/NP, you cannot change back to a letter grade. If you are taking this course as part of a certificate program, you can probably still take the class P/NP. Check with a counselor to be sure.

Instructor Announcements

The instructor will post announcements on the "Announcements" page in Canvas throughout the semester. Canvas notifies students according to their preferred Notification Preferences.

Late Policy

All assignments are due at 11:59pm Pacific time on the Sunday corresponding to the due date. A late submission will receive a 10% penalty for each week it is late. Submissions more than two weeks late are not accepted without prior written arrangement.

Exams

There will be online midterm and final exams. The material comes from the textbook, class lectures and supplemental materials. If any exam is missed, a zero will be recorded as the score, unless you have made prior written arrangements with me. It is your responsibility to take the exams by the due date.

Grading Policy

Click the "Grades" link in Canvas to keep track of your grades. I grade once a week and post grades and comments in the Canvas gradebook.

Grades will be assigned as follows:

Letter Grade	Percentage	Points Total
A	90% - 100%	900 points or more
B	80% - 89%	800 to 899 points
C	70% - 79%	700 to 799 points
D	60% - 69%	600 to 699 points
F	59% or lower	599 points or less

Grading Breakdown

Percent	Points	Grading Category
70%	700 points	**Projects + Assignments**
10%	100 points	**Discussions + Attendance (Participation)**
6%	60 points	**Quizzes**
7%	70 points	**Midterm**
7%	70 points	**Final Exam**
100%	1000 points	**1000 points possible**

Standards of Conduct

Students who register in SRJC classes are required to abide by the SRJC Student Conduct Standards. Violation of the Standards is basis for referral to the Vice President of Student Services or dismissal from class or from the College. See the Student Code of Conduct page.

Collaborating on or copying of tests or homework in whole or in part will be considered an act of academic dishonesty and result in a grade of 0 for that test or assignment. Students are encouraged to share information and ideas, but not their work. See these links on Plagiarism:
SRJC Writing Center Lessons on avoiding plagiarism
SRJC's statement on Academic Integrity

Special Needs

Every effort is made to conform to accessibility standards for all instructor-created materials. Students should contact their instructor as soon as possible if they find that they cannot access any course materials. Students with disabilities who believe they need accommodations in this class are encouraged to contact Disability Resources by calling (707) 527-4278 or visit online at drd.santarosa.edu.

Student Health Services

Santa Rosa Junior College offers extensive health services to students. Visit Student Health Services online at shs.santarosa.edu or call them at (707) 527-4445.

Course Outline

Start Date	Canvas Module	Topics	Assignments
1/13	Week 1	The World of Game Development / Play a Game, Learn to Code	Hosting Signup Survey Assignment 1: Syllabus Quiz Assignment 2: CodeCombat (due Week 3) Discussion 1: Check-in Discussion
1/20	No Class	MLK Jr. Day Holiday	
1/27	Week 2	Introduction to the Transformational Process of Game Development	Assignment 3: Transformational Process
2/3	Week 3	Introduction to JavaScript + Systems Thinking	Assignment 4: First Game Discussion 2: History + Origins of Games
2/10	Week 4	Get Started with Browser-Based Games / The VR Experience	Assignment 5: First Phaser Game
2/17	No Class	Presidents Day Holiday	
2/24	Week 5	Sprites, Controls + Basic Physics	Assignment 6: Working with Sprites + Controls Discussion 3: Game Typologies

Start Date	Canvas Module	Topics	Assignments
3/2	Week 6	Spritesheets, Texture Atlases, + Animation	Assignment 7: Using Animation + Spritesheets Quiz 1
3/9	Week 7	Tilesets, Tilemaps, + Cameras	Assignment 8: Using Level Maps + Tiles Discussion 4: Approaches to Game Design
3/16	No Class	Spring Break	
3/23	Week 8	User Interfaces + Sound / Defining a Game	Assignment 9: UI + Sound Assignment 10: Draft Game Design Document
3/30	Week 9	Building a Game: Team Formation	Discussion 5: GDD Presentations + Team Formation Discussion 6: Game Team Roles
4/6	Week 10	Advanced Physics + Special Effects / Midterm Review	Assignment 11: Advanced Physics Midterm Exam
4/13	Week 11	The World of Three Dimensions	Assignment 12: First PlayCanvas Game Discussion 7: Working in 3D
4/20	Week 12	Final Project: Design Your Game	Assignment 13: Final Game Design Document
4/27	Week 13	Build Sprint 1: Final Project	Assignment 14: Rough Prototype Discussion 8: History of Console Games

Start Date	Canvas Module	Topics	Assignments
5/4	Week 14	Build Sprint 2 + Playtest: Final Project	Quiz 2
5/11	Week 15	Build Sprint 3 + Playtest: Final Project	Final Project Discussion 9: Final Project Sharing
5/18 Mon	Week 16	No Regular Class	Final Exam Discussion 9: Final Project Sharing

Note to students: the assignments listed above will become available as modules are released in sequence each week. To view course content, go to **Modules**.