CS 276: Game Development Syllabus

Fall 2025

Lucas P. Cordova, Ph.D.

Table of contents

0.1 Syllabus Versioning

The syllabus is a living document that may be updated throughout the semester to reflect changes in course policies, schedules, or other important information. The most current version is **1.0 dated 8/25/2025**.

Version	Date	Rationale
1.0	8/25/2025	Initial version created to outline course policies and expectations.

0.2 Lecture Information

Class Time: MWF 12:00 - 1:00 PM
Location: Ford 301

0.3 Instructor Information

Professor: Lucas P. Cordova, Ph.D.
Email: LPCordova@willamette.edu
Office: Ford 210

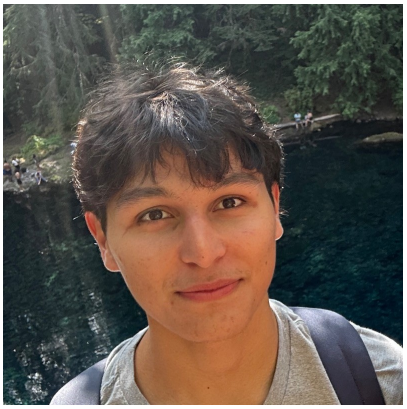
0.3.1 Office Hours

Office hours are available by 15-minute appointments or on a drop-in basis if I am not currently with another student. If the scheduled times do not align with your availability, please don't hesitate to contact me. The scheduling page offers different meeting modalities (in-person, phone, Zoom).

0.3.1.1 Office Hours Schedule

Day	Time
Monday	1:15 PM - 2:15 PM
Tuesday	10:00 AM - 11:00 AM
Wednesday	1:15 PM - 2:15 PM
Thursday	10:00 AM - 11:00 AM

0.4 TA Information



TA: Teo Mendoza
Email: tjmendoza@willamette.edu
Location: QUAD

0.4.1 TA Office Hours

TBD

0.4.1.1 Office Hours Schedule

Day	Time
TBD	TBD

0.5 Course Information

0.5.1 Catalog Description

This course explores the design and implementation of interactive games as a vehicle for learning advanced programming concepts and software engineering practices. Students will work in collaborative teams to design, build, and refine an original game, with an emphasis on object-oriented design, modular architecture, and software design patterns. Projects will integrate narrative design, user experience, and game mechanics to create engaging and cohesive gameplay. The course also emphasizes quality through iterative development, version control, software testing, and structured playtesting. Students will gain experience evaluating their designs through user feedback and adapting their software to meet evolving requirements.

0.5.2 Course Learning Outcomes

1 Course Learning Outcomes

By the end of CS 276 Game Development, students will be able to:

- **Identify and describe** core game design principles, including genres, mechanics, player motivation, and balance, using appropriate game design vocabulary. *(Remember, Understand)*

- **Analyze and critique** contemporary games for functionality, usability, narrative, and artistic design, constructing evidence-based evaluations. *(Analyze, Evaluate)*

- **Apply programming concepts** in Unity by developing scripts in C# to control game objects, implement system dynamics, and manage interactivity. *(Apply)*

- **Design and implement** interactive systems within Unity, including environments, physics, cameras, lighting, sound, user interfaces, and animation. *(Apply, Create)*

- **Integrate narrative techniques** such as branching structures, environmental storytelling, and character development into game prototypes. *(Apply, Create)*

- **Develop and apply** advanced programming and software engineering practices, including object-oriented design, modular architecture, and design patterns to support scalable game development. *(Apply, Create, Evaluate)*

- **Construct and manage** Game Design Documents and asset pipelines while employing project management strategies such as task breakdowns, iteration cycles, and version control. *(Apply, Analyze)*

- **Collaborate effectively in teams** by assuming the roles of Game Developer and Game Reviewer, providing and incorporating constructive feedback in iterative cycles. *(Apply, Evaluate)*

- **Plan, test, and refine** games by applying quality assurance practices, building test plans, and evaluating user experiences through structured playtesting. *(Apply, Evaluate)*

- **Produce and release** a polished, playable game by preparing release candidates, applying versioning strategies, and addressing intellectual property considerations. *(Create, Evaluate)*

1.0.1 Textbook & Materials

- **Guides**:
  - A series of step-by-step guides and video tutorials will be provided to help students grasp key concepts and techniques.
- **Materials**:
  - Class materials will be provided via Canvas and the instructor's website
- **Software Requirements**:
  - Unity (Latest LTS version) - Free Personal Edition
  - Visual Studio or VS Code with C# extensions
  - Git/GitHub for version control
  - Discord for team communication and support
- **Hardware**: A laptop capable of running development environments, required for in-class activities and exams. If you do not have access to a suitable laptop, please contact me as soon as possible to discuss accommodations.

1.1 Assessments

1.1.1 Grade Weighting

Grade Distribution

Component	Weight	Description
Engagement	25%	Attendance, Participation, In-class Activities, Quizzes
Assignments	25%	Design Exercises, Mini-games
Course Project	50%	Design Documents, Journal, Iterative Milestones, Final Demo, Peer Reviews

1.1.2 Letter Grade Distribution

Grade	Range	Grade	Range
A	92.00	C	72.00 - 77.99
A-	90.00 - 91.99	C-	70.00 - 71.99
B+	88.00 - 89.99	D+	68.00 - 69.99
B	82.00 - 87.99	D	62.00 - 67.99
B-	80.00 - 81.99	D-	60.00 - 61.99
C+	78.00 - 79.99	F	59.99

1.1.3 Assessment Descriptions

1.1.3.1 Engagement

Active participation in class discussions, completion of assigned tutorials, and engagement with in-class problem-solving activities. These activities cannot be made up if missed.

1.1.3.2 Homework Assignments

Regular problem sets covering algorithm analysis, complexity theory, and proof techniques. Assignments will include both theoretical problems and algorithm implementation exercises.

1.1.3.3 Course Project

Students will assume two major roles throughout the semester:

Game Developer Design, build, and refine an original game through iterative milestones:

- Concept Pitch (Week 3): Present your game idea and form development teams
- Game Design Document (Week 5): Complete design specification
- Milestone 1: Playable Prototype (Week 7): Core gameplay loop implemented
- Milestone 2: Alpha Build (Week 10): Feature complete, rough implementation
- Milestone 3: Beta Build (Week 13): Polished, ready for testing
- Final Release (Finals Week): Complete game with documentation

Game Reviewer Provide structured, iterative feedback to another development team through:

- Peer review sessions at each milestone

- Written critiques focusing on gameplay, usability, and technical implementation
- Playtesting participation and feedback documentation

1.2 Course Schedule

1.2.1 Important Dates

- **First Day of Classes**: Monday, August 25, 2025
- **Mid-Semester Break**: Friday, October 10, 2025 (no class)
- **Thanksgiving Break**: Wednesday, November 26 - Friday, November 28, 2025 (no class)
- **Last Day of Classes**: Wednesday, December 3, 2025

1.2.2 Weekly Schedule

The following schedule is **tentative** and may be adjusted based on class progress.

Week	Dates	Workshop	Topics	Deliverables
1	Aug 25, 27, 29	Workshop 1	Introduction to Game Design, Unity basics, Course overview	Project teams formed
2	Sep 1, 3, 5	Workshop 2	**No class Sep 1 (Labor Day)** Critical Thinking in Game Design	Quiz 1, Assignment 1 assigned
3	Sep 8, 10, 12	Workshop 3	Game Design Theory, Core design tools	Assignment 1 due, **Game Concept Pitch**
4	Sep 15, 17, 19	Workshop 4	Story and Game Creation, 3D environments	Assignment 2 assigned
5	Sep 22, 24, 26	Workshop 5	System Dynamics, C# Scripting Fundamentals	**Game Design Document due**, Assignment 2 due
6	Sep 29, Oct 1, 3	Workshop 6	Development Tools, Asset Management, IP considerations	Quiz 2, Assignment 3 assigned
7	Oct 6, 8	Workshop 7	Physics and Animation in Unity	**Milestone 1: Playable Prototype No class Oct 10 (Mid-Semester Break)**

Week	Dates	Workshop	Topics	Deliverables
8	Oct 13, 15, 17	Workshop 8	Environments and Level Design principles	Assignment 3 due, Assignment 4 assigned
9	Oct 20, 22, 24	Workshop 9	Cameras and Lighting in Game Environments	**Milestone 2: Alpha Build**, Peer Review 1
10	Oct 27, 29, 31	Workshop 10	Sound Design and Audio Integration	Assignment 4 due, Quiz 3
11	Nov 3, 5, 7	Workshop 11	UI Design, Menus, Build Process	Assignment 5 assigned
12	Nov 10, 12, 14	Workshop 12	Intermediate Animation, Complex UI, Rigging	**Milestone 3: Beta Build**, Peer Review 2
13	Nov 17, 19, 21	Workshop 13	Quality Assurance, Testing, Playtesting sessions	Assignment 5 due, Playtesting feedback
14	Nov 24	Workshop 14	Version Control, Release Management, Polish	**Final Project Prep No class Nov 26–28 (Thanksgiving Break)**
15	Dec 1, 3	Final Wrap-Up	Final Project Presentations, Reflection	**Final Presentations Dec 3**
Finals	Dec 4–10	-	Final Deliverables	**Final Game Release & Documentation**

1.3 Course Policies

1.3.1 Attendance Policy

Consistent attendance is essential for success in this advanced course. I expect you to attend all classes and actively participate in discussions and problem-solving activities. Attendance will be tracked through participation activities that cannot be made up if missed. If you must miss class due to illness or emergency, please notify me as soon as possible.

1.3.2 Late Work Policy

Timely completion of work is essential in this course, as much of the learning is collaborative and iterative. Late work policies vary by assessment type:

1.3.2.1 Engagement

- Engagement activities (attendance, in-class exercises, tutorials, quizzes) are designed for active participation during class sessions.
- **These activities cannot be made up if missed**, as they rely on live interaction with peers and instructor feedback.

1.3.2.2 Homework Assignments

- Homework assignments are due at the stated time on the course schedule or as announced.
- Late homework will be accepted **up to 48 hours after the deadline with a 25% penalty**.
- After 48 hours, homework will not be accepted unless prior arrangements have been made due to documented emergencies.

1.3.2.3 Course Project

- The Course Project is built around **team-based milestones and peer review cycles**. Because milestones affect both your team and your review partners, deadlines are firm.
- **Milestones (Concept Pitch, GDD, Prototype, Alpha, Beta, Final Release):** Late submissions disrupt the entire class schedule and will not be accepted without prior instructor approval.
- **Peer Reviews:** Reviews are due at the scheduled time. Late reviews will not be accepted, as timely feedback is critical to your peers' progress.
- If extraordinary circumstances arise, you must communicate with the instructor **before the deadline** to discuss potential accommodations.

1.3.2.4 Summary

- **Engagement:** Cannot be made up
- **Homework:** Accepted up to 48 hours late with penalty
- **Course Project & Peer Reviews:** Must be submitted on time; late work not accepted without prior approval

1.3.3 Academic Honesty Policy

Academic integrity is essential to your growth as a computer scientist and game developer. Because this course combines individual assignments, collaborative projects, and peer review, you must carefully distinguish between **appropriate collaboration** and **academic dishonesty**.

1.3.4 Allowed Collaboration

- Discussing game design concepts, strategies, and Unity techniques with classmates.
- Working together within your assigned **project team** to design, build, and refine your game.
- Sharing debugging strategies, problem-solving approaches, and high-level algorithmic ideas.
- Using online tutorials, Unity documentation, and official resources to learn techniques (must be cited in your documentation).
- Using generative AI tools (e.g., ChatGPT, Copilot) for brainstorming, prototyping, or boilerplate code **if disclosed and meaningfully modified**.

1.3.5 Prohibited Actions

- Copying or submitting code, assets, or documentation that is not your own (except licensed assets used with explicit credit).
- Sharing or showing your code outside of your project team.
- Using code or project solutions from previous semesters or other online sources.
- Allowing others to copy or use your work.
- Submitting AI-generated code, text, or assets **without disclosure or significant modification**.

1.3.6 Homework Assignments

- Assignments must be completed individually.
- You may discuss strategies conceptually, but **all written work and code must be your own**.
- Cite any resources (textbooks, websites, AI tools) that informed your solution.

1.3.7 Course Project

- Project work is collaborative **within your team only**. All team members share responsibility for the integrity of submitted work.
- Game assets (art, sound, models) must either be original or used under appropriate licensing with clear attribution.
- Peer Reviews must reflect **your own analysis**; copying feedback from others is prohibited.

1.3.8 Playtesting & Peer Review Integrity

- Playtesting is an essential part of the development cycle. You are expected to provide **authentic, constructive feedback** to your peers.

- Fabricating playtesting data, falsifying user responses, or copying another reviewer's feedback constitutes academic dishonesty.
- Reviews should reflect **your honest perspective as a player and evaluator**, supported with specific examples from the gameplay experience.
- Giving insincere or placeholder reviews (e.g., "It was fine" with no detail) will be treated as a failure to meet course expectations.

1.3.9 Consequences

Violations of academic honesty undermine both your learning and your peers' progress. Any suspected violations will be handled according to Willamette University's academic integrity policies, with potential outcomes ranging from a zero on the assignment to failing the course.

When in doubt: ask first. If you are unsure whether a resource, collaboration, or tool use is permitted, check with the instructor before submitting your work.

Violations will result in penalties ranging from assignment failure to course failure, and will be reported to the Academic Standards Committee.

1.4 Willamette Policies

1.4.1 Inclusive Classroom Space

I am committed to creating an inclusive learning environment where all students feel welcome and supported. I will gladly honor your request to address you by your affirmed name and pronouns. Please let me know if your preferred name or pronouns differ from those listed in the course roster, and I will make every effort to use them consistently.

1.4.2 Accessibility and Accommodations

Willamette University is committed to creating accessible learning environments. If you have a documented disability that may impact your performance in this course, please contact Accessible Education Services in Matthews 103 (503-370-6737 or accessible-info@willamette.edu) to discuss accommodations. I am happy to work with you and AES to ensure you have the support needed to succeed.

1.4.3 Time Commitment Expectations

Following Willamette's Credit Hour Policy, you should expect to spend 6-9 hours per week outside of class on course-related activities. This includes reading, homework, programming projects, and exam preparation. The advanced nature of this course requires consistent effort throughout the semester.

1.4.4 Academic Support Resources

- **SOAR Center**: Free access to food, clothing, textbooks, and school materials (3rd floor, Putnam University Center)
- **Tutoring Services**: Available through the Academic Resource Center

- **Technical Support**: IT services for software and hardware issues

1.4.5 Commitment to Positive Sexual Ethics

As a mandatory reporter, I am required to report any incidents of sexual misconduct disclosed to me to Willamette's Title IX Coordinator. For confidential support, contact: - Confidential advocate: confidential-advocate@willamette.edu - WUTalk crisis line: 503-375-5353 - Campus Safety (emergency): 503-370-6911

1.4.6 Land Acknowledgement

We respectfully acknowledge that Willamette University is located on the ancestral lands of the Kalapuya people, who today are represented by the Confederated Tribes of the Grand Ronde and the Confederated Tribes of the Siletz Indians. We honor their deep connection to this land and recognize the ongoing contributions of Indigenous peoples to our academic community.
