

CSCE 2210 – Introduction to Visual Scripting for Games

Instructor: Curtis Chambers

Office: NTDP E260L

E-mail Address: Curtis.Chambers@unt.edu

Student Hours: See Canvas Course Page and By Appointment.

[Instructor Student Hours](#)

[TA and IA Info](#)

Instructional Support Staff Info

Teaching Assistants	Instructional Assistants
	Haeshitha Indukuri HaeshithaIndukuri@my.unt.edu

Contacting Requirements:

Always start with the instructional support staff (TA/IA) and CC me. The individual who graded you needs to be talked to first: usually an IA. If their response merits such, you may then contact your assigned TA. The TAs will escalate it to the next member in the hierarchy, eventually leading to me. As much as I would like to respond to every concern from every student, it is not feasible.

When contacting the instructor or course assistance staff (TA, IA, etc.), the following is required:

- Sent from a **UNT Provided email and/or Canvas Account**
- Sent to my work email (above) OR through Canvas (preferred).
 - TA/IA Preferred Emails will be posted on Canvas.

- Must have the Course Number AND Section (i.e., 2110.212 or 2110.002) in the Subject Line or appropriate field.
- Subject should be short and to the point.
- Body must be detailed, legible, and respectful.
- Please give at least:
 - 2 business days for a response via Canvas.
 - 3-4 business days for email.

Emails, messages, and the like that fail to remain in compliance with the above standards may impact **efficiency of communication** or (in the worst case) result in **no reply**.

Pay close attention to Canvas! Many times, I have received the same or similar message from multiple students. I will post on Canvas to address all participants in the course for what I find is a “collective concern.” Towards this, I also recommend that you check the Canvas Pages for your Lecture prior to sending an email to make sure that it hasn’t been already addressed.

Course Description:

This introductory course is tailored to immerse students in the realm of game development with a specific focus on visual scripting using industry standard game engine tools. Students will gain hands-on experience predominantly with visual scripting, while also exploring other core features of the game engines such as actors, inputs, user interfaces, collisions, graphics, animations, audio, diagnostics, and optimizations. The course will emphasize best practices in visual scripting for game design, fostering creativity within game engine frameworks, and understanding the integration of visual scripting with other game engine aspects. Participants will learn to develop well-structured, extensible projects that leverage the power of visual scripting, and work effectively with complex platforms, frameworks, and toolsets.

This introductory course immerses students in game development, focusing on visual scripting using common industrial game engine systems. It covers core features such as actors, inputs, user interfaces, and more, emphasizing best practices in game design and visual scripting integration.

Prerequisites

- CSCE 1010

Textbooks and Materials

- Relevant readings and resources will be released weekly with each module on Canvas.

Course Learning Outcomes

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

- Apply visual scripting in game engines to create engaging game experiences.
- Utilize game engine core features, emphasizing visual scripting.
- Implement game design best practices within the context of visual scripting.
- Analyze and apply design patterns in game design and development
- Demonstrate creativity and problem-solving in game development using visual scripting.
- Design and develop structured, extensible projects focused on visual scripting.
- Navigate and utilize complex platforms and frameworks.

Topics

- Introduction to Game Engines and Unreal Engine
- Fundamentals of Blueprint Visual Scripting
- Blueprint Integration for Screen Management
- Blueprint Interaction with Actors
- Advanced Actor Manipulation with Blueprints
- Blueprint-based Collision Handling
- Managing Player Inputs through Visual Scripting
- Recasting in Blueprints
- Visual Scripting for Graphics and Animations
- Implementing Audio Features via Visual Scripting
- Diagnostics and Optimizations in Blueprint Projects
- Additional Blueprint Topics (NPC Behaviors, Automation Scripting, etc.)

- Final Presentations of Integration Assignments with a Focus on Visual Scripting Applications

Grading Policy

You must complete all tasks required on time. Late assignments will not be accepted without appropriate excused documentation. You have **1 week to dispute a grade** after it's posted date. After receiving your grade for an assignment, you must email the grader (TA or IA) and myself (CC'd) **within 7 days** should you wish to discuss/dispute it.

The above is to avoid “end of the semester” rush to alter grade penalties. Each student should keep track of their grades **throughout the semester**. Note that as we near the end of the semester, the time to adjust assignments gets shorter.

Each assignment will have varying requirements. **Pay very close attention** to what I am asking you to deliver in every assignment, program, and exam. Uploads may be on SVN, Canvas, GitLab, in person, or (in many cases) more than one.

Your final grade will be a weighted average according to the following:

- Project and Presentations %40
- Quizzes %20
- Assignments %30
- Peer Reviews %5
- Participation %5

Grades will be posted on Canvas throughout the semester to provide an ongoing assessment of student progress, though final assessment will be measured using the weighted average above.

Once a grade is assigned on Canvas, students have one (1) week to dispute the grade.

The proper channel for grade disputes is to first go to the original grader (i.e., TA/IA) to resolve the issue (*don't forget to include me in the email so I may monitor the exchange*). If, however, a resolution cannot be reached between the student and the grader, the student shall then go to the instructor who will have the final say on the grade.

Most assignments will be due at 11:59 PM on the specified due date to Canvas. All assignments must be completed and submitted according to their specific directives. Any

programming assignment submitted after its deadline **will not be accepted and receive a grade of 0.**

Extra Credit

Extra credit assignments are as follows:

- Assignments Category:
 - Drop Lowest Assignments Grade
 - Attendance will add points to your assignments category
- Quiz Category:
 - Drop Lowest Quiz Grade

Submission Policy:

All projects are expected to be submitted on time with all the correct parts through the Canvas system. The project documentation must be created as a wiki page in GitLab, and no photographed or scanned content will be scored. Further, any content that is deemed illegible will not be scored. The program must be coded in C or C++ and contain ample comments and descriptions. All programs will be compiled and executed on the department's CSE servers, and any that fail to compile or execute on that system will receive a zero. Additionally, a README file, in .txt format, with clear instructions on how to compile and execute your program must be included.

Assessment Methods

- Quizzes to evaluate theoretical understanding.
- Projects and assignments focused on practical application and problem-solving skills.
- Final project presentation assessing the integration of course concepts.

Attendance Policy Detailed:

Attendance is **required**. It will part of your Participation Grade.

Lecture Session: Required.

Should you miss class, you are responsible for the covered course material and assignments you may have missed. The instructor will not be responsible for re-teaching material missed by a student who did not attend class.

As always, if there are extenuating circumstances, please notify your instructor and course assistance staff (such as TAs and IAs) **via your UNT Email ASAP** (BEFORE the missed lecture) so that you can work together to ensure your success in learning the material. **Refer to the section “Contact Requirements” and subsection “Accommodations.”**

Note that the TAs and IAs should only be notified. Do not expect a response from them outright.

Excused Absences Defined:

Students are expected to schedule routine appointments and activities so as not to conflict with attending class. However, some absences cannot be prevented. In the event of a medical emergency or family death, etc., students must request an excused absence as quickly as feasible following the event. While it is preferred that I am notified prior to the event, that cannot always be the case. Send to me (**ASAP**) a brief email from your **UNT provided email address**. You need not go into detail as to the emergency, but you should schedule with me a meeting outside of lecture at your earliest convenience. Students must be able to provide documentation that verifies the reasoning for the excused absence. Above all else, this course is compliant with UNT Policy 06.039 “Student Attendance and Authorized Absences.” Please refer to this policy for more details/information.

Refer to the section “Contact Requirements” and subsection “Accommodations.”

Emergencies:

By definition, emergencies cannot be planned for. Your instructor attempts to make accommodations in these instances that allow for making up missed work and completion of the course in a timely manner. Students must provide documentation that verifies the emergency.

Refer to the section “Contact Requirements” and subsection “Accommodations.”

Accommodations:

THIS DOES NOT APPLY TO ODA ACCOMODATIONS. (These are processed through the ODA Offices)

First and foremost: Review UNT's policies:

- Student Attendance and Authorized Absences: <https://policy.unt.edu/policy/06-039>
- Code of Student Conduct: <https://policy.unt.edu/policy/07-012>

When finished, should you still want to request accommodations (such as make up work, re-takes, extensions, etc.) for an unavoidable absence or impact, you must use this process:

1. Email me and your assigned TAs/IAs from your UNT email.

This is to notify us about the absence pending an excused document from the Dean of Students Offices.

2. Contact the Student Affairs Office. (When applicable)

Email: DEANOFSTUDENTS@UNT.EDU

Phone: 940-565-2648

Note that the DoS offices only *request* accommodations and do not *enforce* it. Work with them to verify the validity of the excused absence or impact. Once their offices process the details with you, they should provide you with a document and/or contact me. Should you get a document, send it to me via email and keep the physical copy for your records.

I'll need these document(s) before I can provide accommodation. This process will keep us both in compliance with UNT standards, practices, policies, and procedures while being the most efficient and private method.

Lastly, each non-valid (or otherwise inappropriate/pervasive) attempts to abuse this will be reported to the appropriate parties/representatives of the university. This will also impact future attempts to request accommodations.

If you have questions, please visit me during my office hours or schedule a meeting with me.

Disability Accommodation:

The University of North Texas makes reasonable academic accommodation for students with disabilities. Students seeking reasonable accommodation must first register with the

Office of Disability Accommodation (ODA) to verify their eligibility. If a disability is verified, the ODA will provide you with a reasonable accommodation letter to be delivered to faculty to **begin a private discussion** regarding your specific needs in a course. You may request reasonable accommodations at any time; however, ODA notices of reasonable accommodation should be provided **as early as possible** in the semester to avoid any delay in implementation.

Note that students must obtain a new letter of reasonable accommodation for every semester and must meet with each faculty member prior to implementation in each class. Students are strongly encouraged to deliver letters of reasonable accommodation during faculty office hours or by appointment. Faculty members have the authority to ask students to discuss such letters during their designated office hours to protect the privacy of the student. For additional information, refer to the Office of Disability Accommodation website at <https://studentaffairs.unt.edu/office-disability-access/index.html>. You may also contact ODA by phone at (940) 565-4323.

Academic Misconduct & Integrity

- This course follows UNT's policy for Student Academic Integrity that can be found at <https://policy.unt.edu/policy/06-003> as well as the Cheating Policy for the Department of Computer Science and Engineering.
- The department, college, and university have very strict guidelines regarding academic misconduct. Students are expected to submit **their own work on all individual**
- You are allowed to discuss solutions, but **do NOT work with other students on shared program/assignment solutions. Do NOT use even partial program solutions from the Internet without properly citing them. Do NOT recycle a complete assignment**, this will result in a failing grade. The complexity of these assignments should not merit the use of external resources. Failure to remain in compliance with the guidelines is considered cheating and will be reported.
- **You will be graded on your contribution to the code.** Be honest—attribute your work. Submitting code or work that you did not solely author (without acknowledging it to the instructor) is **cheating** and will be dealt with in accordance with the department cheating policy.
- If it is determined that you have cheated, the first instance of cheating in the class will result in a grade of **ZERO (0)** on the assignment in question and referral to the

department chairman and dean of engineering. The second instance of cheating in the class will result in **a grade of F** in the class, and a dismissal hearing may be initiated by the dean of engineering. (NOTE: that Instructional Support Staff can now report Academic Integrity violations as well...)

- **You need to do your own work.** Here, there should be no ambiguity at all.
- In case the above description, and in-class discussion of my views on appropriate and inappropriate collaboration does not answer all your questions, please look at the university Student Rights and Responsibilities web page.
- You are responsible for the information covered in class, **whether you attend class or not**. Individualized lectures **will not be given**. Please check with other class members for any notes that might have been missed during an absence. Attendance won't always be taken in lecture and your attendance is strongly recommended to improve your opportunity to meet course objectives.
- You should not work with other students on shared program solutions or use program solutions found on the Internet.
- Specifically, you should **never copy someone else's solution or code**, and **never let a classmate examine your code**.
- A sophisticated program will be used to compare your work to the work of all other students (including students in past classes).
- If you are having trouble with an assignment, please consult with your instructor or course assistance staff (TAs, IAs, Graders, etc.).
- Failure to adhere to these strict standards may be cause for disciplinary action even leading to expulsion from the University.
- Each student should adhere to the university's student code of conduct. The Code of Student Conduct can be found at <http://deanofstudents.unt.edu>.

Below I have taken out two paragraphs from UNT Policy 06.003 Student Academic Integrity.

“UNT promotes the integrity of learning processed and embraces the core values of trust and honesty. Academic integrity is based on educational principles and procedures that protect the rights of all participants in the educational process and validate the legitimacy of degrees awarded by the university. In the investigation and resolution of allegations of student academic dishonesty, the university's actions are intended to be corrective, educationally sound, fundamentally fair, and based on reliable evidence.”

“Students are expected to conduct themselves in a manner consistent with the university's status as an institution of higher education. In the class setting, students shall follow their instructors’ directions and observe all academic requirements published in course syllabi and other course materials. A student is responsible for responding to an academic dishonesty report issued by an instructor or other university official. If a student fails to respond after proper attempt at notification, the university may take appropriate academic actions in the absence of the student.”

UNT policy 06.003 defines the following breaches of academic integrity:

1. **Cheating.** The use of unauthorized assistance in an academic exercise, including but not limited to:
 1. use of any unauthorized assistance to take exams, tests, quizzes or other assessments;
 2. usage of sources beyond those authorized by the instructor in writing papers, preparing reports, solving problems, or carrying out other assignments; usage without permission, of tests, notes, or other academic materials belonging to instructors, staff members, or other students of the university;
 3. dual submission of a paper or project, or resubmission of a paper or project to a different class without express permission from the instructor;
 4. any other act designed to give a student an unfair advantage on an academic assignment.
2. **Plagiarism.** Use of another’s thoughts or words without proper attribution in any academic exercise, regardless of the student’s intent, including but not limited to:
 1. the knowing or negligent use by paraphrase or direct quotation of the published or unpublished work of another person without full and clear acknowledgement or citation.
 2. the knowing or negligent unacknowledged use of materials prepared by another person or by an agency engaged in selling term papers or other academic materials.
3. **Forgery.** Altering a score, grade or official academic university record; or forging the signature of an instructor or other student.
4. **Fabrication.** Falsifying or inventing any information, data or research as part of an academic exercise.

5. **Facilitating Academic Dishonesty.** Helping or assisting another in the commission of academic dishonesty.
6. **Sabotage.** Acting to prevent others from completing their work or willfully disrupting the academic work of others.

Cheating of any sort will not be tolerated in this course. All submissions must be your own original work. Taking information or code from the internet or other students is considered a breach of academic integrity. Failure to adhere to these strict standards will be cause for disciplinary action that could be as severe as expulsion from the university. If it is determined a student cheated on any assignment in this course, they will receive an F for their final course grade and an academic integrity report will be filed with the Office of Academic Integrity.

Further, UNT is now maintaining a database recording any acts of academic dishonesty that is available to employers. Additionally, because these are group projects, if one group member is caught cheating the consequences of their actions will extend to the group (as a whole). It is the responsibility of all group members to ensure that when they put their names on their submission as a whole and submit it, the submission does not contain any instances of cheating. Failure to report known instances of cheating within a group will be deemed facilitation of academic dishonesty and reported as such. For more information see the UNT Student Academic Integrity Policy.

Generative AI:

Unless you are given written permission to use Generative AI (i.e., ChatGPT), do not use these tools in your work. To do otherwise will be considered a violation of Academic Integrity and will be handled as such. Please note that it is much easier for your instructors and the support staff to run models that will detect the use of such as well.

Collaboration Policy:

For each project submission, all work is expected to be your own. If you are instructed to work with group members: you are not to collaborate with other groups for projects, provide solutions to other groups, search for solutions on the internet, or purchase solutions. Doing so will be deemed a breach of academic integrity. However, for any non-graded, practice assignments students are encouraged to work together to solve problems.

Student Responsibilities:

Students are responsible for submitting the **correct assignments** for each applicable assignment submission. Submissions should include the **correct files** and submitted **prior to the deadline**. Late material will NOT be accepted or assessed. Programming assignments **MUST** compile to receive any credit.

Academic Freedom and Academic Responsibility:

Refer to UNT Policy 06.035

Academic freedom and academic responsibility give vitality to the UNT and its mission. As such, the academic freedom to be able to freely consider or investigate important, and, perhaps, controversial questions are essential to the education of students and advancement of knowledge. Faculty have the academic responsibility to subject their knowledge and postulates to rigorous review by peers who are experts in the relevant subject material, to have a firm foundation of their postulates in the most relevant and suitable available evidence, and to work with one another to provide the best education possible for our students.

Syllabus Revisions:

This syllabus may be modified as the course progresses should the instructor deem it necessary. Notice of changes to the syllabus shall be made through Canvas and/or class announcements.

Weekly Course Schedule (Subject to Change)

WEEK 1:

Lecture: Course overview, syllabus review, and introduction to the tools and technologies that will be used throughout the semester.

Lab: No lab session this week.

HW: Review course materials and set up necessary software and group formation.

WEEK 2:

Lecture: Overview of game engines with a focus on Unreal Engine.

Lab: Basic Unreal Engine setup and interface exploration.

HW: Install Unreal Engine and familiarize yourself with the interface.

WEEK 3:

Lecture: Introduction to Blueprint visual scripting system.

Lab: First simple Blueprint project.

HW: Create a basic Blueprint script that performs a simple task.

WEEK 4:

Lecture: Integrating user interface elements and screen management using Blueprints.

Lab: Building a simple menu system using Blueprints.

HW: Enhance the menu system with additional features like buttons and sliders.

WEEK 5:

Lecture: Methods for actors to interact within Blueprints.

Lab: Set up different actor interactions using Blueprint.

HW: Design an interactive scenario with multiple actors.

WEEK 6:

Lecture: More complex manipulations of actors using Blueprints.

Lab: Develop a mini-game or scene with complex actor interactions.

HW: Add additional features or complexity to the actor interactions.

WEEK 7:

Lecture: Handling collisions and overlaps in game scenarios using Blueprints.

Lab: Implementing various collision-based interactions.

HW: Develop a small project that requires intricate collision handling.

WEEK 8:

Lecture: Capturing and responding to player inputs with Blueprints.

Lab: Create custom input mappings and associated actions.

HW: Design a control scheme for a hypothetical game.

WEEK 9:

Lecture: Utilizing ray casting techniques in gameplay elements.

Lab: Implement basic ray casting to interact with game objects.

HW: Create a puzzle or mechanic that uses ray casting.

WEEK 10:

Lecture: Implementing graphics and animations through Blueprints.

Lab: Animate characters or objects in a scene using Blueprint tools.

HW: Animate a complex scene or short sequence using learned techniques. Explore advanced aspects like lighting, materials, textures, shaders, and particle effects in Unreal.

WEEK 11:

Lecture: Adding and controlling audio elements in Unreal Engine via Blueprints.

Lab: Integrate sound effects and background music into a Blueprint project.

HW: Design an audio-rich environment or sequence.

WEEK 12:

Lecture: Techniques for diagnosing and optimizing Blueprint scripts.

Lab: Analyze and optimize an existing project for performance.

HW: Optimize a provided Blueprint project with specific performance targets.

WEEK 13:

Lecture: No new lecture content; focus on project development and improvement.

Lab: Open lab for project work and individual consultations.

HW: Continue developing and refining projects.

WEEK 14:

Lecture: Best practices for documenting projects and code.

Lab: Documentation workshop to assist in project documentation.

HW: Complete all documentation for final projects.

WEEK 15:

Lecture: Preparing and delivering effective presentations.

Lab: Final preparations and dry runs of project presentations.

HW: Finalize presentation materials and rehearse for the final showcase.

WEEK 16:

Last Day of Class: Project Showcase

No Lecture: Day dedicated to final project presentations and showcases.

No Lab: Students present their projects.

HW: Submit final versions of projects and presentations for grading.