

# Final programming project for CPE 476 – Spring 2017

Your primary project for this course consists of a quarter long project to be completed in ~9 weeks. This is your chance to make a portfolio quality game/virtual environment. This project is worth 45% of your final grade in this class (plus 20% for individual technologies + 2% final game play) = 67% of your grade in this class).

We will hold a “game market place” to establish teams and final projects. This marketplace is a chance for potential managers or team members to “pitch” their game idea(s) to the class and try to form teams. Ideally, teams of size 4 are formed. All games must meet the required technical requirements.

All games must be or include (whole team equally responsible for):

- At least one single level complete game or interactive 3D world with nice graphics that contains enough variation from other team’s projects, (see below for details for specific graphics technologies),
- fun and enjoyable to interact with (i.e. winnable game or compelling interactive world) – you will be assessed on this metric via play-testing by the class and feedback from the instructor,
- some kind of audio
- good software development (this is vague, but consider the value of careful design, unit testing, writing clear code and general performance issues)<sup>1</sup> – Mostly, poor software design and or version control is not an excuse to not meet a deadline or de-bug a feature.
- the project must include the following graphics technologies (or instructor approved substitution) – One person on the team needs to choose to be responsible for one of the following **major** technologies – and ALL **major** technologies must be present in all games:
  - A camera/view that changes: (*high performance*) the camera must be mostly free to move with a player in the 3D world, (*arcade*) 2.5D arcade style camera allowed.
  - a nice complex environment with characters that navigate and interact with that environment (*high performance*) articulated character motion, (*arcade*) rigid body character motion.
  - some form of collision detection and response using a spatial data-structure, (*high performance*) 3D spatial data structures, (i.e. bounding sphere hierarchy), (*arcade*) 2D spatial data structure (i.e. uniform spatial subdivision) – note however some hierarchy required see vfc

---

<sup>1</sup>For example: <http://cowboyprogramming.com/2007/01/05/evolve-your-heirachy/>  
(I don’t necessarily endorse component systems but do think you need to think about your software design for your game – but this class is focused on graphics!)

- shadows (*high performance*) shadow maps or shadow volumes, (*arcade*) baked shadows or projection shadows
- hierarchical view frustum culling *or appropriate occlusion culling (pending instructor approval)*
- two specialized rendering techniques from: physically based rendering (pbr)<sup>2</sup> in general or cook Torrance shader<sup>3</sup>, deferred shading, normal mapping, environment mapping, NPR or ambient occlusion *or instructor approved substitute*

Now, depending on your team size, choose 2-6 of the below, (choose as many as needed such that each team member is responsible for at least 1 major technology and one from the below list), all technologies must have a demonstrable visual side-effect:

- level creation tool
- mesh simplification tool
- sophisticated HUD,
- management of level-of-detail modeling, either terrain or character models,
- complex pathing (splines) for characters
- grab bag of shading: ambient occlusion and 2 different shading models (BRDFs) –or - bloom and motion blur (or distance) <exact shaders must be approved by instructor>
- grab bag of game effect (e.g. particle system, lens flare, fog, billboard, etc.) <exact effects must be approved by instructor>
- complex collision response, (scripted) fracturing, complex alpha mapping, etc.
- some bonus feature (graphics or otherwise)\*

\*Note that the above list of technologies does not include a lot of features common in many games (i.e. AI or networking for example) you are welcome to build in these features but they do not supersede/replace any of the graphics requirements. They can, however, count for your bonus feature. In addition, nice assets (i.e. characters, textures, environments, etc.) are important to the look and feel of many games but creating or importing them do not count towards any of the graphics requirements but can be applied as your bonus feature.

Note that this year, in order to help build the most productive teams possible, we will have two categories of games with slightly different criteria: *high performance* or *arcade*. These two categories are primarily to help team's form that includes like-minded teammates and the two categories have two different tiers of possible points (however, any 'arcade' category game can switch to high-performance at any

---

<sup>2</sup> [http://renderwonk.com/publications/s2010-shading-course/hoffman/s2010\\_physically\\_based\\_shading\\_hoffman\\_a\\_notes.pdf](http://renderwonk.com/publications/s2010-shading-course/hoffman/s2010_physically_based_shading_hoffman_a_notes.pdf)

<sup>3</sup> <http://renderman.pixar.com/view/cook-torrance-shader>

time) – to be categorized as ‘arcade’ you do not need to build an arcade game, this distinction is for teams that want to focus on minimum requirements. During game pitches, the pitch should specify which type of team the manager would like to build (be honest). *High performance* games can earn up to 110%, while *arcade* games can earn up to 90% total for this portion of your grade. *(Note, regardless of how you pitch your game if your game has 2 or more criteria from the arcades style, it will be evaluated as such).*

*Teams:* Each team will pick a **manager** and it is assumed that team participants will specialize. Every member of the team must take responsibility for one major graphics technology and either a second major graphic technology or one of the optional technologies. Managers and teammates will help decide grade distribution within the team (see below) and will be held responsible for getting their team to meet deadlines (i.e. the manager will be my point person of harassment). Part of each of your grades depends on your mutual assessment of one another.

*A note about using outside references and technology/code:* You are welcome to use any outside resource/code/technology that will help your team meet its goals (including assets, modeling programs, physics engines (not full collision), tutorials, audio libraries, etc.). However, **all** of the required graphics technology must be implemented by one of your team members in a way that does not take advantage of other classmates (i.e. you copying code from a tutorial for view frustum culling **is** taking unfair advantage of your classmates). You or some member of your team must understand/master all of the required graphics technology required for your game. In addition, you are required to list all references you plan to use in your final project proposal and in your project report. Be very careful when considering the use of outside technology, sometimes it takes as long to learn someone else’s code as it does to write it yourself. ***You are required to individually take responsibility for two graphics technologies (at least one of them being from the ‘major’ technologies list) – to get credit for these technologies; you must review your individual code with the instructor in person. 20% of your final grade will be evaluated based on your individual creation of these graphics technologies and the associated code review.***

*One cool thing:* Your project will need to be a fairly complete single level game, however, I encourage your team should focus energy on ‘one cool thing’ that will make your game stand out. This might be a very rich visual environment or particularly cool character movement/interaction, or a visually exciting effect, or a very particular look/feel to the game, etc. Again, you need to include all the features specified, but focusing your effort on one very nice aspect of the game is a good way to make a simple game stand out.

*Progress demos:* ***You and all your team members are required to be present*** for 4 intermediary progress checks and the final demo of the project. **These demos are mandatory and will be graded.** At each demo, a portion of your grade will be awarded (i.e. the same portion of the project that should be done. In other words

25% of your grade for your final project will be assessed at the 25% check-in ) If you miss a check off, you will automatically lose that portion of your final project grade. These check-offs will happen in lab on approximately the following dates:

- **25% done by 1/26** (First 25% of grade assessed)
- **50% done by 2/16** (Additional 25% of grade assessed)
- **75% done by 3/7** (Additional 25% of grade assessed)
- **90% done by 3/16** (Additional 15% of grade assessed + *game play assessment conducted this session*)
- **100% done and demoed to entire class on 3/23** (final 10% of final project grade assessed– it is expected that you will take into account the feedback given at the playtest session)

**Note that for grading, I will assign each team a grade at the check-in. That grade can then be scaled based on feedback from both the manager and the team members. Your team grade counts toward 45% of your course grade, your individual technology grade is 20% of your final course grade**

---

#### **Formalized Game Proposal:**

Formalized Game Proposal: Each team must submit a 1-2 page write-up describing your final project and game design in more detail <in general I prefer short and concise - bullet lists are great, the proposal can be longer than 2 pages if it has lots of images>.... Be sure to provide the team members names along with a designation of who the project manager is, on all submissions!

You will not be strictly held to this document but it serves as an initial contract between your team and I about what will be included in your project. This is an opportunity to start working on where you will get your assets (character models, world) what the look and feel will be, and what type of interactions you will support. You are required to include in the write up:

1. A storyboard overview drawing of your game
2. 1.5) a general description of the game
3. a short description of the environment for the game (what it will look like and where you will get the models for the environment, e.g. terrain, etc.)
4. a short description of the characters in the game (what they will look like and where you will get the models)
5. a short description of the goal of the game (what is the character trying to do? what tools does the character have to achieve this goal?)
6. a short description of the animations that will be used for moving the character (and any other moving aspects of the environment)
7. a short description of the general rules of the game
8. a short description of who will tackle which major and general graphics technology and which will be included in the game <look at the technology list - don't hog the fun ones>
9. a description of the "one cool" thing that your team will emphasize in the

game (i.e. the one thing you will make look really cool or interact in a sophisticated way, etc.)

10. Some concept art to convey the look and feel of your game <you may use the same concept art used in your presentation!>

**Due one week after team formation!**

## **Final demo/report requirements held on 3/23 (final period):**

- Project Demonstration. Each team will have 10-15 minutes to present their final project to the class. In addition, we will have some in class playability testing, where each team should have a playable version of their game set-up for class mates to try and assess (this will happen at the 90% demo and it is expected that for 100% your team will have addressed major issues that arise in the game play assessment).
- Project Report -- this is in the form of a web page and should contain the following information. Also note that links should be relative (not absolute paths), and that you should submit the html and all necessary image and animation files in the same directory.
- brief description of your project,
- **All team members names**
- a mini user's guide,
- sample output i.e. images
- ***a short animation clip of your game is required***
- a list of all references used (e.g. tutorials, research papers, etc.)
- A list of what the team implemented for each required technology
  - a project executableProject submission (likely thru polylearn) information will be provided later.

Project Grading:

- No **late programs will** be accepted.
- Your final grade will be the culmination of your demos throughout the quarter as specified earlier in this document.

**Playability Testing – 90% assessment:** Note that in addition to the 45% of your final grade for your team’s game, which will be evaluated throughout the quarter for your final project, we will have some in class playability testing, during the 90% assessment time period. This will be a chance for 5 members of the class not on your team to play your game. Your team will need to have your game installed and ready to play by others in the class. A strict rubric for grading one another’s games will be provided and you will grade one another’s game. This playability assessment, (assuming all issues that can be addressed will be), will account for 2% of your final grade.