

Call for Proposals for Final Projects

Handed out Wednesday, Feb 21. Due **Thu, March 1 at 11:00pm** (via ELMS).

Only *one member* of each team should submit, but please include all the team members' names on the proposal.

The final projects will be due late in the semester, perhaps the last week of classes. The exact due date will be announced later.

Overview: Give a short (roughly one page) high-level synopsis of your proposed game.

Team Members: List all the members of the team. (Working alone is fine. There is an advantage to small teams, say 2–3 people, since it is easiest to balance the workload and avoid coordination issues. If you have 4 or more people, please check with your instructor first.) You may have team members from different sections of the course.

Game Title: Proposed title, which you may change later. (But try to come up with something. “TBD” is not good.)

General Description: The game's general structure (e.g., FPS, puzzle, RPG, single/multi-player) and the game's general “look and feel” (3-d interactive, 3-d isometric, 2-d scrolling, turn-based, etc.). What (if any) concrete games inspired your game? Please feel free to include illustrations or images.

It is not necessary to provide details at this point. We are more interested in your vision of what the game will look like. You are free to make changes in the future. (If the changes are significant, please keep your instructor informed.)

Platform and Resources: On what system do you plan to implement/execute your game? What software tools (e.g., game engine, graphics, geometric modeling, physics, audio) will you use? Will you need any special hardware (e.g., controllers, head-mounted display, or gesture recognition)?

Coordination: (For group projects.) How will you and your teammates coordinate your work? How often do you plan to meet? Where will the source files be maintained? Do you plan to use some form of shared file storage (e.g., GitHub) and/or a revision control system (SVN or CVS)?

Tips and Suggestions:

Before choosing software/sharing: Check that all team members are able to install and run the software and file-sharing systems, otherwise you may need to start over from scratch.

Is it “Demoable”: You are free to propose any general structure you like. The only pragmatic constraint is that it must be possible for you to present a short (say 5-minute) demonstration of the game to the class at the end of the semester, and you will need to provide progress reports to your instructor and/or the teaching assistants a couple of times during the semester.

This can be an issue for turned-based games that take a long time to play. Also, if there is any special hardware needed, you will need to haul it onto campus and get it set up.

Nothing is binding (yet): You can make changes, even radical ones, throughout the semester. If you do, please keep me informed.

Previous projects: To get an idea of what is “doable,” check out the videos of last semester’s projects

<http://www.cs.umd.edu/class/spring2017/cmsc425/final-projects.shtml>

(Of course, don’t be influenced too strongly. A part of your grade is based on how innovative your project is.)

Build in Layers: It is easy to dream up a project that you will not have sufficient time to finish. Start with a basic implementation so that you are confident you can successfully develop. Then, grow the project by adding enhancements and extensions. (Ambitious plans may fall apart near the end of the semester, leaving you with nothing to show.)

Do the hard things first: It is important to identify as early as possible in the development process any implementation issue that might hold up your progress. Try to determine such issues early and design a prototype to be sure that you achieve your minimum goals. Later, you can add the “bells and whistles.”

Do at least one thing well: Face the fact that it is not possible to produce a AAA game in one semester. Rather than doing a so-so job on many different elements, focus instead a single concept that will make your game stand out. (This might be something internal, such as a novel technical feature. If so, part of your final demo will involve an explanation of how you implemented this feature.)