|  |  |
| --- | --- |
| Program: | GDP1 |
| Course: | For “Graphics 2”, “Physics & Simulation 2”, and “Gems” courses |
| Professors: | Michael Feeney |
| Project/Test: | 2024 GDP “Game Jam” |
| Weight: | “one project” and possibly “one exam” worth of weight (see details) |
| Due Date: | April 20th, 2024 |

*Note: This project can be done alone or in groups of up to three   
(but I would strongly suggest that you do it individually or pairs).*

Description and Purpose

Now is your time to shine and create your amazing piece of art/technical demo/game.

Briefly, you will incorporate and demonstrate six (6) “technical things”. Since there are three (3) courses and you need to perform two (2) topics each, you need to incorporate six (6) “technical things” in total. The marks are distributed evenly across all four, but have a different “weight”, depending on the difficulty (more details below).

You may choose to do this Game Jam submission as *just* your final project *and* write a final exam in one (1) or more of the three (3) courses OR you may choose to apply the final exam marks to this Game Jam entry (so it would be worth more in marks, but you would *not* be required to write a separate final exam).

More details about this on the next page.

Critical note: Since it is possible to interpret that the “jam” is a single assignment, worth > 40%, assigned during the last week of classes, you **MUST indicate your choice of the following to me at least two (2) weeks before the “jam”**:

* That you want to do the game jam as a combination of the last project for my three (3) courses (“Graphics 2”, “Physics & Simulation 2”, and “Game Algorithms and Gems”) *in place of the final exam*. They would be marked as one item, and the corresponding mark applied to both courses. In other words, you would be having the jam *instead of* a separate final exam and final project in these courses (aka: no final exam),   
    
  OR
* The “jam” is simply your final project in each of the courses, but y*ou still want to do a final exam in one or more of these courses*. This would mean that you would complete the “jam” (**it’s your final project**) ***and***would complete the final exam for ***each*** of “Graphics 2” and/or “Physics & Simulation 2”, and/or “Game Algorithms and Gems”. These would be scheduled during exam week and worth ½ of your “exam” mark (so 20% of your final mark). The single “jam” mark would be applied to the project mark for each of these three courses.

**Another note:**

* You would still be required to complete six (6) technical topics for the jam
* **No later than the end of “week 12” (two weeks before the last day of class),   
  which is Friday, March 19th, 2024:**
  + You must communicate your choice (“Jam = project + final” OR “Jam = *only* final project, with separate finals”) no later than “week 12” (so two weeks before the last day of class).
  + You also need to communicate if you are doing this as a group and individually. If a group, the entire group received the same mark, and I *completely “wash my hands” of any group dynamics issues* (i.e. you are marked based on your submission to the jam and how your group behaved or didn’t behave, internally, has absolutely no bearing on the evaluation).

**Submission:**

You must submit your solution to the FOL drop box and/or e-mail an equivalent location (Google Drive, dropbox, etc.) to me and submit your submission to the itch.io game jam site:

Here’s the link: <https://itch.io/jam/gdp202324-final-game-jam>

Note that **you have to submit the source code to me**, but you ***do not***have to post the source code to itch.io.

\*\*\*IF YOU DO NOT COMMUNICATE YOUR PREFERENCE TO ME IN WRITING (aka e-mail) BY Friday, March 19th, 2024),   
 **YOU ARE AGREEING TO THE 2nd OPTION**:   
*“Jam = project, three (3) final exams”* as per the original course outline

In other words, please *let me know* that you **DON’T** want to do the exam as well, or I’ll have to assume you ***DO*** want to do the exams as well.

Since this is potentially a group based project, everyone in your group must independently communicate their choice in writing *(via e-mail)*. However, the entire group gets the same mark for their submission.

The four topics:

* 60%: the “four things” you have to do, technically. They can be something that extends what you’ve done in class or something outside of class.
* Your marks are based on how effectively you implemented each of the four topics, with the grade range loosely based on the following. If you are unclear, please clarify with me *prior* to submitting:  
  + - (0%) Something trivial. This is something that we’ve looked at in some depth, is something that has been incorporated for a while, or is something that would take a professional minutes to do – this includes the trivial incorporation of alternative libraries (like assimp instead of ply, or solid instead of texture loading, etc.). This also included anything you’ve done in any other courses (look, I incorporated Bullet Physics! – which I did for Physics 2…)
    - (75%) Easy (something you just coped from class/internet, relatively unchanged, but have extended or enhanced it in some *significant* way. Adding a 2nd spot light *isn’t* something in this category, for example, however implementing a dozen moving, colour changing “disco” spotlights might be…🤔)
    - (100%) Hard (something we covered but is somewhat involved in doing and/or incorporating)
    - (115%) Super Hard (something we barely or didn’t cover, so you had to investigate this all yourself, and *also* is technically changing)
  + You get 1/4 of the mark for each “thing” you do. So if you only do 3 of 4, then you can only get 3/4 of the marks (assuming they aren’t all “Super Hard” that is...)
* 20% : the “ranking” of your projects based on the feedback from the players
* 20% : the “six things” + “ranking” based on a technical lead from a local game company
* Note: I reserve the right to “override” these last 40%, which may include (but not limited to):
  + You’re given some extreme mark, like all zeros (it could happen, I suppose).
  + You’ve violated one of the prior rules.
  + Based on me seeing the code and/or seeing what’s happened in class, etc., I know that you didn’t actually implement this. For example: you used boost; you used a 3rd party library for functionality that you are claiming you did yourself, etc.
  + You are presenting something for the 2nd time and/or are using something from another class. For example, if you’ve already done AABB collision in 1st term (project #2 of Physics 1), but you haven’t changed anything (or trivially changed it), then you can’t submit it again. Same thing for including something from another course, this term, that you have already submitted for a mark in that class.

Details

Possible Topics (this is just some suggestions, **not** a “definitive” list or anything):

* multi-texturing and displacement
  + Decals
  + Normal maps (aka bump maps), displacement mapping, geometry and tessellation (this has to be significant, not just putting a trivial pass-through shader in there)
* Off-screen effects:
  + Blur, Bloom, Depth of field, etc.
  + “Night vision”
  + “XRay” (or something else from Meshlab or Turbo squid)
* Deferred rendering ***FULL*** deferred rendering, not just rendering off-screen, then presenting)
* LOD (Level of Detail)
  + There are many ways to do this:
  + Replacing models (lower resolution versions, like the Stanford Bunny models)
  + Replacing the model with an “imposter” (2D Billboard)
  + Adding an “imposter” that’s not a 2D billboard
* Multiple shaders in scene (or a much more complex “uber” shader)
* GPGPU
  + Tons of options here: rain, particles, depth of field, etc.
  + Trick is: a) getting data into and out of shader and b) using that data in the shader
* Shader stuff:
  + Toon shading
  + Edge detection
  + “fur” or “hair” (geometry shader or adjust model)
  + Deferred rendering (G-buffer)
* Graphical “instancing”
  + There are explicit calls for this (DrawInstanced)
* Particles:
  + Smoke, sparks, flame, etc.
* Something else (transform feedback buffer)
* Threading (this has to show a significant architecture change, usually with a performance increase, either in terms of how “much is done” in a frame and/or a significantly *reduced* CPU load. Just adding something pointless on another thread doesn’t count.
* Another (or “more”) physics things (something Lukas isn’t marking for Physics)
  + Chains, ball-joint constraints
  + Hinge constraints
  + Stacking objects
  + Floating
* Terrain rendering (this is dealing with a lot of data, more than you can usually render)
* Make your game self aware, take over the Internet, then the world, then create robots that go back to the past and attempt to kill Sara Connor. Just an idea.
* Here’s the SimonDev planet video (all in JavaScript! 😱): <https://www.youtube.com/watch?v=hHGshzIXFWY&list=PLRL3Z3lpLmH3PNGZuDNf2WXnLTHpN9hXy&index=1>

Note that this will vary depending on the course – i.e. how many projects you have done. For these courses, the “project” portion of the mark (which could be between 2 and 4 projects, usually 2 or 3) is worth 60%, while the “exams” portion is worth 40%

For example:

* Your course has had 1 project (1/2 of 60%) and 1 mid-term (1/2 of 40%).
* This “jam” is essentially “Project 2” and “the final exam”, so worth 50% of your final mark.

Another example:

* Your course has had 3 projects and 1 mid-term.
* The “jam” would be a combination of “Project 4” and the final:
  + Project 4 = ¼ of 60% = 15%
  + Final = ½ of 40% = 20%
  + For a total “weight” of 35% of your final mark

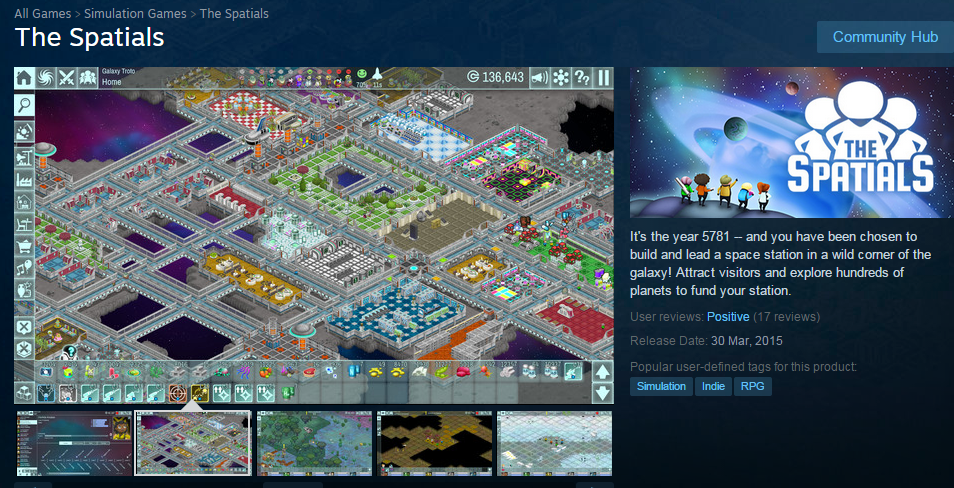
The Game Jam:

**The restrictions:**

* You can use anything you have “accumulated” during the year, as a student, so anything from class, code you’ve written, etc.
* *Can’t* be a game engine, like Unity, Unreal, SDL, Ogre, etc.,
* This *can not* be something you’ve done in another class. While you *can* use code from other classes, you can’t just submit something you’ve already submitted in another class (unless you run this by me first and I can verify that the other instructor is OK with this).

**The process:**

* If you can’t think of an idea, try this to pick a silly name: <http://videogamena.me/>
* You *don’t* have to use the name generator but you *must* have a name for your submission.
* You prepare a brief description of what you’re doing, which includes:
  + A very short “elevator pitch” of what the game is (like you see on Steam, see below)
  + A breakdown of the “technical topics” you are implementing. You need “six things” in there:
    - There are three projects (one per course: Graphics 2, Gems, Animation 2)
    - There are three final exams (one per course)
    - You are convincing me that these things are actually implemented (through a quick video walking me through the code that does it)



This is an example of a “pitch” that you will give to your “audience” of technical and play testing people (the text on the right of the image with “*It’s the year 5781…*”)

Grading Scheme

1. Normally a grade of zero will be assigned to any assignment that is submitted late. However, certain rare exceptions apply according to the Fanshawe College Academic Policies and Procedures, specifically 2-C-02: [*https://old.fanshawec.ca/sites/default/files/assets/policies/pdf/2c02.pdf*](https://old.fanshawec.ca/sites/default/files/assets/policies/pdf/2c02.pdf).
   1. Because this game jam involves getting (very busy) external technical resources/judge (people) here for one day, a “missed evaluation” becomes more complex. In that case, I will try to arrange to have at least one external technical resource to evaluate your submission, but if not, your full evaluation will be done by me, using the same format: i.e. an oral defence of your “technical things”.
   2. Also note that because this is a *project* (even though part of the mark is included as an “exam”), you have weeks to complete it; barring an “exceptional circumstance” that lasts the entire time of development (weeks and weeks), *you will be expected to submit your jam entry on the day of the game jam* – you will simply be evaluated at a later time.
2. You must submit your code so that I can see it and compile it.
3. If you code does not even compile, I will not mark it, and it cannot be submitted as game jam entry. No exceptions. This means that your mark for the game jam would be zero.
4. If you code does not build (i.e. linker error) and run (i.e. no crazy run-time crash that is unexpected), I may investigate this further, but only if there is some simple problem and/or slight configuration error.
   1. **PLEASE** include all the libraries, etc., that are required to build your project. Again, I *might* make an attempt to investigate this, but, ***technically, I don’t have to do this, therefore your project “won’t build”, so you would get zero*** (as per 1. and/or 2. above).

Additional requirements:

* While you may freely “borrow” mine (or anyone other) code ***but*** your code should be “sufficiently” different. This is based on what an average programmer would think, or just some random person.  
    
  To clarify this: while I don’t expect you to recreate everything from scratch (although I encourage this), I don’t want to see something that is essentially my in class demos submitted back to me, essentially unchanged. In other words, I need to see that you have put a lot of work into it.  
    
  As a guide of these terms “essentially unchanged” and “a lot of work”, use what me – or one the highest performing student in the program – could do: if what you did would take one of us < ½ an hour to do (like changing the names of the 3D model files, placing them at another place in the scene, etc. – well, that would take, like, 60 seconds, but you get the idea), then it’s *not* “a lot of work”.
* You also *cannot* simply use an existing game engine (or part of a game engine, Ogre3D for instance) to complete this assignment; it should be either completely new of significantly modified.