General design:

* Update the code base.
  + Refactor and fully annotate all existing code.
    - Learn to use coroutines and IEnumerator.
    - Sort all code into the new folder structure.
* Shader programming
* Kudvenkat C# tutorials
* One or more **font styles**
* **3D models**: (learn Blender using Brackeys’ Blender course)
  + Replace Brackeys’ 3D models with your own. ***Do NOT spend too much time on small details or complex 3D models. The models are to be viewed from a distance. If possible, model and texture them within Unity’s folder structure, so that the changes may be viewed live.***
    - **Towers**
      * **Scattercannon** (shotgun, 1 tile range, massive damage)
        + Model | Texture (UV unwrapped) | Shotgun
        + 1 tile range | Fire rate (2s/1.5s/1s)
      * Look up towers on the Sanctum wiki, guns from the Borderlands wiki, and sci-fi weapons. 3 or 4 towers in total.
      * Use matte colours and look for subtle ways to add more colour to the towers, like changing the trim or barrel colours. Don’t use logos, as these won’t be visible.
        + Branded towers with manufacturers’ names or logos, for future releases?
    - **Nodes –** Get James to do up some sample node textures.
      * Hexagon: Model | Texture
      * Triangle: Model | Texture
      * Square: Model | Texture
    - **Ground**
      * Hexagon: Model | Texture
      * Triangle: Model | Texture
      * Square: Model | Texture
    - **Enemies**: dark purple spheres coupled with particle trails?
      * Non-profit demo – Korosensei heads, melting into liquid on death. To this end, use Anti-Sensei ammunition in your towers.
      * Get the enemies to face the right way. This means splitting apart the Enemy prefab, and adding the Canvas and the Enemy individually to a new Empty.
      * Kongregate/Newgrounds prototype:
        + Basic: Model | Texture | Particles | Light | Sound | Anim
        + Tank: Model | Texture | Particles | Light | Sound | Anim
        + Runner: Model | Texture | Particles | Light | Sound | Anim
    - **Environment**
      * Build your maps around environment set pieces that you will construct in Blender.
    - **Spawning** **portal**: Particles | Textures | Light | Sound | Animation
    - **The** **Core**: Model | Texture | Particles | Light | Sound | Animation
    - **HUD**: Remaining lives | Countdown to next wave | Towers list with keyboard keys for selection |

For the finished prototype:

The most important thing you need to remember about the prototype is that it can under no circumstances contain any references to the Brackeys tutorial series on which it is based.

Remember that while you did follow the tutorials from the beginning and only used the assets offered by Brackeys, you did NOT at any point download any of the Unity Projects he has left up on GitHub. You reconstructed the entire game as per the tutorial, and once his provided assets are gone, along with the way everything is laid out on screen, it becomes your own finished product; something you can be proud to host on your own website and advertise as your own creation.

For full game:

Pathfinding (A\* system using Sebastian Lague tutorials)

* Do NOT spend too much time on the coding!
  + Make sure you research the system thoroughly before you start coding it!
  + If you find yourself stuck, do NOT give up the entire project!
  + Work on it in between more substantial additions to the game. The system is NOT important unless you’re looking to give the player a level of freedom akin to Sanctum. You don’t know how far beyond your skill set this is yet, and that level of freedom isn’t necessary to make a good game!