*Juris Civilis Development Diary*

| Date of Session | Time Spent | Development Segment | Notes |
| --- | --- | --- | --- |
| 14/10/2021 | 20 Minutes | Meeting | Discussed what documentation was needed and what steps were required for starting the project. |
| 21/10/2021 | 30 Minutes | Meeting | Discussed progress on project plan and any steps forward needed |
| 04/11/2021 | 2 Hours | Analysis | Began writeup of the project literature review, specifically the review of “Dwarf Fortress” and its use of world generation and world simulation |
| 04/11/2021 | 1 Hour | System Design | Began development of some basic tools for the project to work from – specifically some basic camera controls (movement and zooming speed), as well as some work on mouse clicking interactions and the spawning of basic 2D circles as a response to clicking interactions. |
| 06/11/2021 | 3 Hours | Analysis | Added multiple segments to the literature review, including references to Europa Universalis 4, Civilization 5 and a past project of mine with relevance to the topic area. |
| 06/11/2021 | 3 Hours | System Design | Added more to the aforementioned tools, including the basic ability to generate polygons around a selected point, and application of properties to these polygons which can be accessed by clicking on them. This system has some flaws, notably with the hit detection of the polygons, but these should be mitigated as the project progresses. |
| 13/11/2021 | 4 Hours | System Design | Began to use the research collated from the literature review stages to document the design of the world generation procedure – specifically the discussion of the geographical features of the world and values such as rainfall, elevation, and temperature. Began discussion of province mechanics. |
| 14/11/2021 | 7 Hours | World Generation – Objective 1 | Attempted to create a prototype for the diamond-square algorithm to be imported into the main system. This unfortunately was largely a failure but can be built from in the future. |
| 15/11/2021 | 7 Hours | World Generation – Objective 1 | Attempted to implement a Perlin noise algorithm, to limited success. The algorithm still needs work to be used in the project. |
| 18/11/2021 | 20 minutes | Meeting | Discussed progress with literature review and world generation methods, confirming progress on these topics has been acceptable thus far |
| 18/11/2021 | 4 Hours 20 minutes | World Generation – Objective 1 | Added the first working terrain generation algorithm, using a reworked Perlin noise algorithm and fractal Brownian motion. This algorithm is able to differentiate land and sea as well as mountainous regions. The source code for running the other Perlin functions is still present in the code for later use in other functions such as temperature. |
| 24/11/2021 | 6 Hours | World Generation – Objective 1 | Started the unity side of the project and ported over the existing terrain generation algorithm into the unity system – then added the ability to perform additional Perlin generation algorithms with variable settings to represent other features such as temperature. As of yet the algorithm is largely inefficient and poorly implemented, but changes will be done to improve the performance of this system. |
| 25/11/2021 | 5 Hours | World Generation – Objective 1 | Added some code to make a basic equator – this code is not yet perfected as it does not properly blend with its surroundings, and needs some reworking to ensure it blends properly with the Perlin temperature system. |
| 28/11/2021 | 1 Hour 50 minutes | World Generation – Objective 1 | Reworked the equator system to blend better with the surroundings, also removed a significant amount of the randomness of the system to provide a more consistently positive result. |
| 29/11/2021 | 1 Hour | World Generation – Objective 1 | Added some small cosmetic changes – including some basic threading incorporation to stop “not responding” issues when the program takes time to generate stages. As of yet this means that the software simply tells the user what stage they are on – but there may be merit to implementing a system to draw the map after each stage, though this could cause some slowdown issues. |