# Fractal Explorer

|  |  |  |  |
| --- | --- | --- | --- |
| **Student** | Louis Durston-Wyatt | | |
| **Teacher(s)** | Steve Wentworth | | |
| **Version** | 1.0 | **Date issued** | ? |

|  |
| --- |
| Background |
| The problem my project will solve is the gap in truly interactive, high-quality fractal visualisers. It will be a useful educational tool for teaching students about fractals in the complex plane in a visual, interactive way.  My program’s aim is to generate and display a fractal in the complex plane (such as the Mandelbrot Set, Julia Sets and Newton’s Fractals, generated using iterative methods) that the user can traverse by panning and zooming. The fractal will regenerate at increasing levels of precision as it is zoomed into, creating the illusion of infinite detail. The user can then “record” a route through the fractal and press “generate”; a high-resolution video following this route through the fractal will be generated using compute shaders, where the GPU can iterate many complex inputs in parallel.  This lends itself to computational methods because generating the sets that represent these fractals requires repeated iteration of functions (such as the Mandelbrot equation, zn+1 = zn2 + c) which would be impossible without computationally performing the calculations to arbitrary detail. Also, displaying these sets on the complex plane cannot be done without a computer as millions of points need to be plotted.  Benefits for this project:   * Helps educate people about a high-level mathematics topic in an interactive way * Parallel processing on the GPU allows for a much quicker generation time than competitors |

|  |
| --- |
| Research |
| **Target Market**   * People interested in fractal mathematics who want to use the software recreationally * Teachers/professors who want to use the software as a tool to teach students * Students who want to use the software to further their understanding of fractal mathematics   **Alternatives**  There is another piece of software under the name “Fractal Explorer” - info can be found at [fractal-explorer.com](http://www.fractal-explorer.com/). The software’s features are similar to that which I intend to develop, however it is outdated as it has not been updated since 2011. Features that I intend to include are: option to generate video that zooms into the fractal; ability to zoom into the fractal; option of generating the Mandelbrot Set; intuitive GUI. An example of a feature that I do not intend to include is generation of geometric fractals, such as the Koch Curve. The software has a limit of zoom depth and does not recursively regenerate the fractal as it is zoomed into; my software will be an improvement upon this as it will create an illusion of infinite detail. I intend upon contacting the developers of the software to enquire about what programming language / development tools they used during the software’s development. |

|  |
| --- |
| Success Criteria |
| Success will be measured by how closely matches the given success criteria. At this stage, the criteria to be met will be detailed as a set of high-level aims below. In further stages, the success criteria will be made more detailed and specific, with additions being made by the stakeholders of the project.   * Menu where the user can choose which kind of fractal they wish to explore, and input the required inputs for the chosen fractal * A graphical fractal environment in which the user can pan around and zoom in/out (Explore Mode) * An intuitive GUI that gives the user access to movement about the fractal * A feature that allows the user to record a path through the fractal * A Cinematic Mode that uses compute shaders to display the recorded path in high precision; a video can also be generated |

|  |
| --- |
| Out of scope |
| * Allowing the user to move around the fractal in cinematic mode; this is out of scope because the HSHL language for compute shaders cannot take inputs from the user while running * Other types of fractals, such as geometric fractals like the Koch Curve; this is out of scope because the fractals that my project generates are plotted in the complex plane, representing properties of mathematical equations |

|  |
| --- |
| Stakeholders / Responsibilities |
| **Stakeholders**   * Elysia Barker: teaching Mathematics at Manchester Metropolitan University… * Samuel So: A-Level Student at ULMaS, interested in fractal mathematics but would like to further understand the topic using Fractal Explorer. Motives are educational/recreational   **Responsibilities**  My stakeholders will be asked to provide ideas for the requirements of the product based on their roles within academia; these different roles (student, teacher/professor, recreational) are representative of the target demographics of my project, allowing the development of the program to be adjusted to the needs of the average consumer.  These ideas for improvements can be given by the stakeholder at the end of each development cycle, of which there will be 3 to 5 before the product is completed. Each cycle will take approximately 1 month to be completed, with the final product being mostly developed by December 2022. This guidance could simply be in the form of an e-mail, in which the stakeholder lays out a few key points on how well the prototype of the product meets their needs as a consumer.  Stakeholders will also take part in user acceptance testing at checkpoints in the program’s development, most likely after each developmental cycle. This will only involve testing a prototype of the product and completing a form, in which they can provide feedback at a level of detail that seems suitable to the stakeholder. |

|  |
| --- |
| Ability |
| I am a proficient programmer, and have been learning Java over the first year of my A-Level computer science course. I will complete online tutorials in HSHL to develop my understanding, giving me the necessary tools to develop compute shaders. I will employ my algorithmic thinking skills, using techniques such as decomposition and abstraction to write efficient and effective code; this will be further discussed in the high level plan.  The project will follow the agile development life cycle, with 3 to 5 development ‘sprints’ before the product is completed. Each cycle will take approximately 1 month to be completed, with the final product being mostly developed by December 2022; this allows constant working prototypes to be developed and improved upon using the guidance given by stakeholders.  The Explore Mode will be programmed in Java, using Apache NetBeans as the IDE; this allows me to easily integrate the ‘Swing’ java package to create a GUI. The Cinematic Mode will be programmed in HSHL (a compute shader language), using Unity as the engine.  I will employ object-oriented programming, creating classes for concepts such as complex numbers, and the complex plane grid. Procedural programming will also be used in order to develop the logical structure of the program, through use of sequence, iteration and selection/branching.  No external hardware requirements are necessary as, if the project requires internet hosting, it can be hosted on my personal web server from home. Necessary software requirements include the Unity engine, which I will use as the engine for the compute shader section of my project. |

|  |  |  |  |
| --- | --- | --- | --- |
| Risks | | | |
| **ID** | **Risk** | **Action to address** | **Owner** |
| 1 |  |  |  |
| 2 |  |  |  |

|  |  |
| --- | --- |
| Sprint Zero Requirements | |
| **Requirement No** | **Description** |
| 1 | High level requirements for entire project (including visual and user experience designs) |
| 2 | Detailed requirements for Sprint One |



|  |  |  |
| --- | --- | --- |
| **Student** | Louis Durston-Wyatt |  |
| **Teacher(s)** | Steve Wentworth |  |
| **Stakeholder [1]** | Elysia Barker | |
| **Stakeholder [2]** | Samuel So | |