Programming project

ModelX

By Joshua Da Silva

# Contents

[Contents 2](#_Toc115858217)

[Analysis 3](#_Toc115858218)

[Problem Identification 3](#_Toc115858219)

[What suits this project to a Computational solution? 3](#_Toc115858220)

[Who is the project suited for? 5](#_Toc115858221)

[Research 6](#_Toc115858222)

[Previous Similar Solutions 7](#_Toc115858223)

[Evaluation 11](#_Toc115858224)

[Features of Project 11](#_Toc115858225)

[Limitations of Project 13](#_Toc115858226)

[System requirements 14](#_Toc115858227)

[Hardware 14](#_Toc115858228)

[Software 14](#_Toc115858229)

[Success Criteria 15](#_Toc115858230)

[Design 17](#_Toc115858231)

[Development stages 17](#_Toc115858232)

# Analysis

## Problem Identification

There is a virtually limitless amount of ways particles can interact with each other. Consequently, the ability to create a model to determine the resulting action of these particles would create a varying amount of benefits allowing for example a physicist collecting a large amount of data and creating conditions that may not be possible with equipment available to them.

To find a solution to the creation of the particle simulation it would require the understanding of all forces that will act upon the objects such as gravity, drag, forces acting on each other etc. To continue, these forces would all be modelled to graphical solutions then having the user be capable to set variables, modelling what they require such as the angle of collision and number of particles. The software would then be capable of taking these measurements and returning them to the user formatted in a table with all results as well as for example the distance travelled at different intervals of time.

## What suits this project to a Computational solution?

This project is suited to a computational due to many facets’ such as for the removal of human error and handling large data sets. Software would allow for the removal of human error as calculations would only be able to be completed by following an algorithm that will not deviate from its instructions. As well as the ability to remove human error, it also allows for any field that may require the use of particle simulation, to advance research without the understanding of calculating collisions, formatting and collecting data allowing for more flexibility amongst many fields.

An important aspect that a computational solution to particle collision offers is accessibility, due to the virtually limitless possibilities, extremes can be reached in variables such as mass which are unavailable to be recorded or conducted by someone with limited resources.

|  |  |  |  |
| --- | --- | --- | --- |
| What features would the project require? | Description of feature | Computational approach | Alternative approach |
| Calculation | This would require for equations of gravity, motion, repulsive forces, centre of mass and many more. This feature would be crucial, and this would allow for the graphical solutions to the collision as well as the collection of data as it allows for example velocity, force and location to be found. | A computational approach would require for a separate function for each calculation as well as, threading would have to be used because many equations will rely on each other. To continue OOP would also be appropriate with it allowing for all calculations assigned to a class | An alternative approach would require for a human to use a calculator alongside a pen and paper, this would still have the ability for human error to be an issue and the ability for a step to be missed this is not present in a computational approach. |
| Graphical User Interface | All particles would have to be modelled so that their paths, collisions and displacement can all be modelled and displayed to a user to visualise and understand what is happening. All particles would be displayed in a plane/ field where all forces can be correctly modelled with this being a 2D plane that the user would be able to have the results be displayed to them through. | To create the GUI, a computational method would include having each particle be modelled and updated in a time frame that allows for feedback for the user on all calculations and positions with dimensions of x and y also taken. This would be able to be replicated by a human but the level of accuracy would not be possible to replicated and there is a likelihood of error. | Another method to create a GUI would require for a hand drawn method with all data being considered, this will not be as accurate as a computational solution and would also be subject to human error. To have a human amend a graphical interface to include multiple sets of results would also be less responsive for the user as it would not be possible for it to be completed in the same time frame. |
| Data collection | Collection of data is required for this project with this data being able to be manipulated by the user and to be used to reach the means desired by the user. This data would need to be formatted and easily available for the user to save and utilize. | Data collection would require for all data to be formatted and put into a table format after all calculations are completed. This would be sorted through an algorithm. | Alternatively, a human could create a results table and manually enter in each set of data at intervals desired by the user. This could cause for some data to be incorrectly entered as well as cause for a much slower process to receive results as it will be a more tedious and inefficient process. |
| Particle variables | This is the determination of variables of each particle with an ability, also varying the number of particles being modelled with variables such as mass and velocity being modelled to change. | Computationally this problem can be approached allowing for a user to input variables through a slider system, stopping high values that may cause for errors such as a mass that may crash a computer to calculate. This approach would allow for easy storing of each variable of the user as well as used in calculation. | This process can be conducted alternatively by having the variables be submitted by the user and then using these variables a person can calculate the result of the collision depending on the submitted variables. However, with this method human error is a factor alongside with inefficiency that causes for a computational approach to be more suitable. |

## Who is the project suited for?

This project has a varying number of potential stakeholders these will include people such as psychists, students, chemists etc. Stakeholders are the most important part of the project, and they are the reason that a project is created whether that be you or others every project will have a stakeholder, a reason for creation. Each stakeholder has unique requirements and uses for the ability to model collisions, a psychist may have different reasoning to use the project than a chemist a chemist may be interested to see gas collisions whereas a physicist may want collisions within a confined plane.

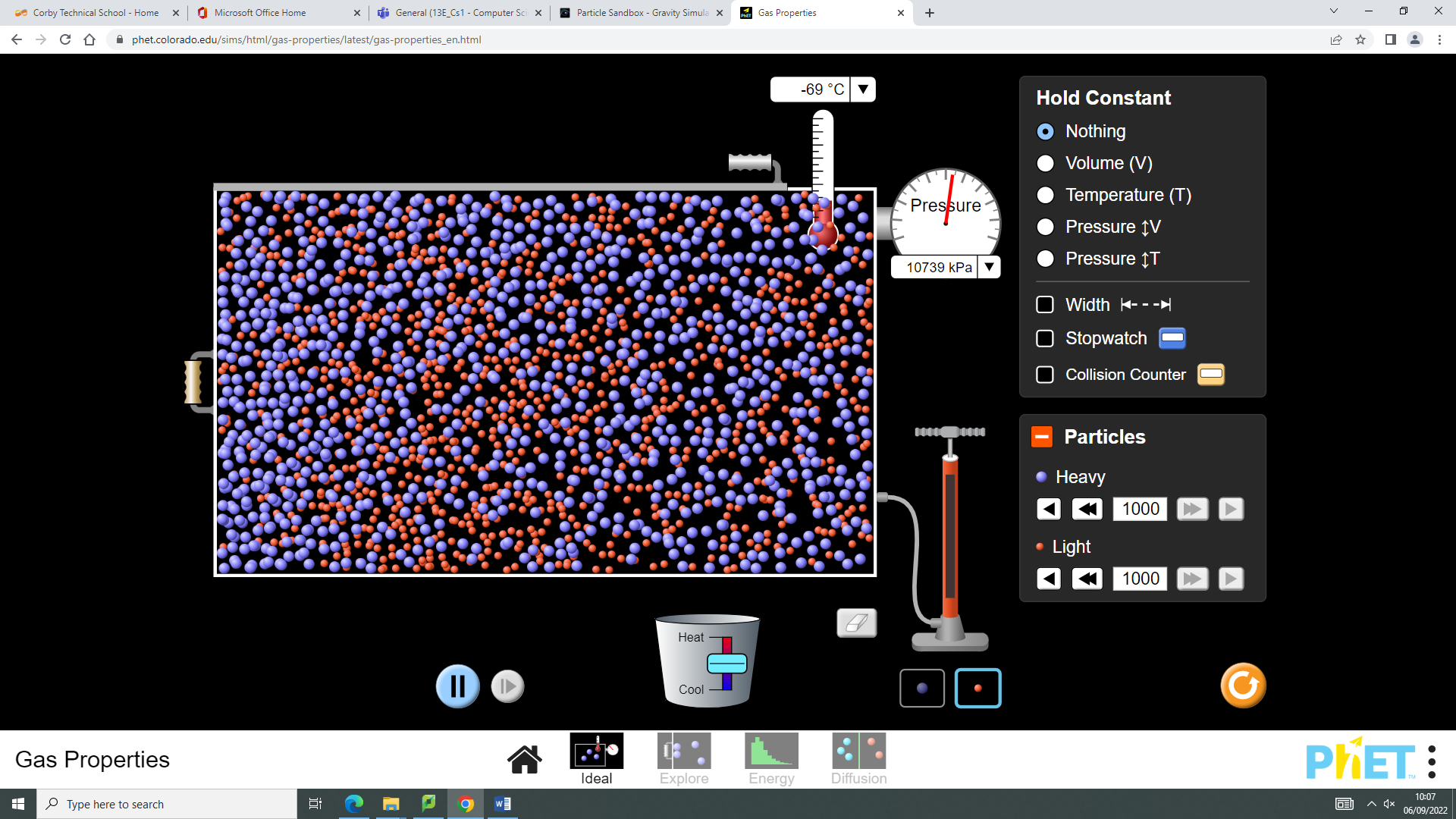
Some stakeholders that may have an interest in my project are

|  |  |  |  |
| --- | --- | --- | --- |
| Stakeholders | Description of interest | Use of solution | What makes it appropriate to their needs |
| Physicists | Create and display models of particles that allow for any potential research, calculations and compilation of data to be continued, started or refined. | The solution would be used to create any situation of variables and particles that are needed by the physicist with this then being formatted and compiled for the physicist without the use of a physical experiment. | It would allow for conditions to create experiments without equipment, meaning that accessibility and location are removed as concerns for particle simulation as it can be completed anywhere. It Is also suited to creating the functionality of formatting for the user allowing for easily digestible results. |
| Teachers | Engage and teach students content through an interesting graphical approach, alongside with removing human error with automated calculations and formatting to save time on their lesson. | Creating experiments with there being the ability to save conditions that can then be relayed to students to observe conditions in particle simulation. | It allows for any lack of conditions to be circumvented, with a teacher not having the capability to display their lesson to students with physical experiments/methods being replaced with the software allowing for it to be displayed and taught digitally. |
| Students | To experiment and explore with conditions and attempt to learn through changing conditions and demonstrating an effect to improve their scope of understanding. | Students would be able to use the resources in the solution to create and explore the possibilities with the scope of variables that can be controlled and have all resources be calculated and displayed to them through the intervals that they want. | This solution enables any student to learn away from a controlled learning environment and initiate means that when they are away from any resources, they may need to conduct a physical experiment they can replicate and increase their understanding. |
| Chemists | Model experiments that are needed for gases, water and particles for research/projects and attempt to remove errors. | The solution can be used to help create conditions that are needed with the number of particles and type of matter such as gas/liquid and having this displayed back to them. | Using the Solution allows for ease of access and accessibility compared to creating an experiment that would require a more rigorous set of circumstances to be adhered to. As well as this the variables that are suited the experiments and data collection are much easier to change to a normal solution. |
| Game designers | To implement particle modelling and simulation to their game to save time and introduce new features. | It would allow for the software to be transferred and implemented to the game that they want with new features such as water collision. | The Solution is suited to Game Developers as it allows for them to use particle simulation without having to understand the physics that are needed in their game. |

## Research

### Previous Similar Solutions

#### Phet



<https://phet.colorado.edu/en/simulations/gas-properties>

Phet is a free educational resource that has many examples and uses of particle simulation with variables able to be edited and the results displayed graphically to the user some features it includes are

|  |  |  |
| --- | --- | --- |
| Features | Description of feature | Would I use it? |
| Constant Variables | The feature that allows the user to change the variables to remain constant for example if the simulation of the particles were in a container and the volume would be made constant. | I would use this feature as it would allow the user to experiment as well as this it may be a requirement for some project/research of a potential stakeholder. |
| Reset Button | Button on the bottom right which allows for all conditions to be reset and the simulation to be stopped and reverted to its original condition/template. | I see this feature as useful so I would look to implement it so the user would not have to delete everything or refresh the page this would also allow for a template they have created to be used instead of the default for every user. |
| Particle Size | Differing sizes of particles that can be inputted into the certain domain with their being heavy and small. | I would use this feature however I would look to add more customisability as the size would have a slider/range that can be used than just set variables of heavy and light. |
| Maximum number of particles | The limiting factor that does not allow the user to create to many particles with 1000 displayed as the maximum possible in this particle simulation. | I think this feature would be necessary as if too many were to be used then it is likely that the program would crash due to the complex calculations that would be required. |
| Graphics for features | Having Graphics be used to change variable such as the bucket to change the temperature of the container or the pump to enter particles. | This feature is likely to be added as this would be making it more useful as an educational tool as well as creating the ability for feature to be translated across languages without translating language itself. |
| Collision Counter | Having the number of collisions that have occurred between the particles be presented to the user. | This feature would be implemented as it would be useful to keep track of if someone were looking to include into a project or research how a variable affects collisions they can visualise and record this effect. |
| Vacuum | The Area that the particles encompass is a vacuum meaning that forces such as drag do not affect the particles. | This would be a useful feature however this would have to be added to be an optional as it would not be useful to simulate conditions that are on earth. |
| Stopwatch | Having the time that passes during the simulation be displayed to the user. | I think this would be implemented however the intervals of time would be recorded so 1s intervals can be taken and results at that point saved to be viewed. Alongside this having a scale of time ranging from small values of time too large for the increased functionality. |
| Stop and Start | Buttons to allow for the user to stop the simulation at a point and continue | I would look to implement this as a feature as the user may need information at a point in time from which this can then be relayed and then the data formatted for any research/project |

#### particlesandbox

<http://particlesandbox.com/>

Particle sandbox

|  |  |  |
| --- | --- | --- |
| Features | Description of feature | Would I use it? |
| Gravity | Particles that have their own separate gravity that allows for them to interact with each other and create gravitational spheres. | I would look at including gravity in my solution with it being useful for the models that may want to be created for the most realistic simulation in the conditions of earth if they are desired by the user or needed for the user’s project. |
| Direction decider | Allows for the user to choose what direction the particle moves in from a position. | I would not look to include this feature as it may not be necessary possible as the particles will all be able to be entered into an area with a clicker to introduce a large amount compared to individual particles with a direction allowed to be inputted. |
| Magnifying glass/zoom | The buttons in the top left that allow the user to zoom in or out to change their perspective/view on the particles. | This feature I would not look to be implemented as I do not think that there will be a use for a 3D model due to the use to display the results of these calculations graphically as well as to display the results in a formatted order. |
| 3D | The model is created in 3D meaning the user can traverse the domain in the x, y and z dimensions. | In my solution I would see a 2D solution to be a more fitting solution as to attempt to display a 3D solution to the user would not be useful to most of my stakeholders. |
| Particle counter | The number of particles that are concurrently on display at a point of time are displayed to the user. | My solution would look to implement this as it allows the user to understand the current conditions that are happening and if more needed to be added to reach a certain condition the user knows this information. |

#### WebGL Particle Physics

<https://nullprogram.com/webgl-particles/>

|  |  |  |
| --- | --- | --- |
| Feature | Description | Would I use it? |
| Objects | The ability to optionally create objects to block particles from moving to an area. | I Would not look to include this as my solution is not concerned in the sandbox approach to particle simulation. |
| Particle customisation | Particles being able to have the properties and conditions be modified with radius, mass speed etc. | My solution would look to implement customisation as it would be useful for particles to be able to have each variable be edited. However, this may not be implemented to the degree of having every particle customisable as it will not be needed for uniform particles which is the use of the program. |
| Colouring | Having the ability to colour the particles and any objects that are in the simulation | I would look to implement this in my solution as it would allow for any potential differentiation for groups/categories of particles that are implemented |

### Evaluation

Following the research on similar solutions, the solution I propose would look along the lines of a 2D domain in which the particles would interact with each other, have the ability to be customised and modified with colours, size, mass etc; with these particles being affected by variables depending on the user’s choice such as gravity and drag. All of this would be able to be calculated in earth conditions or in a vacuum with the results being formatted and available to the user with it being displayed in time intervals which will also be decided by the user, along with the number of particles and particle collisions available to the user at all times. To continue I would also look to implement the ability for the user to control the simulation as it happens with a stop and start functionality, a reset (button) of the conditions and the ability to save a condition set of variables to be used again at a later point in time.

## Features of Project

The initial concept of the project alongside the research of similar projects is along the lines of a program that will initially run displaying a GUI with a menu that will allow the user to choose from the options of templates that are available and the ability to just create their own template or conditions that can be saved/reused. In continuation, once the user has selected from these options they then can visualise the resulting calculations with a graphical interpretation presented with then a sidebar allowing for further customization with a slider and buttons to reset conditions, play and pause. Finally, once the user is finished with the graphical interpretation they can choose to have the results of the experiments be formatted into a table with it being done in time intervals of 1s with all relevant data shown as requested.

|  |  |  |
| --- | --- | --- |
| Features | Description | Why I am using it? |
| 2D | A GUI that is only concerned with the x and y dimensions | I am using a 2D graphical solution as it would allow for more users to be able to use my device with system requirements that would be lowered allowing for this to be more viable of a resource in teaching and for students which are some important stakeholders. There would also not be much use to adding the z dimension as this would not be useful as the visualisation would only need to show a simple collision for the visualisation. |
| Customisation | The ability for the user to customise all aspects of the particles and conditions | Customisation allows the user to create conditions that are necessary for what they want out the simulation, alongside this there is also the use of differentiation where the user may want to have subsections of particles like small and large with this coloured differently to show these effects clearly. In continuation this would also continue to templates as the customisation that had been previously chosen would be able to be saved and I would be able to choose common experiments to be saved |
| Particle/Collison counter | A bar displaying the number of particles on the screen and how many collisions have occurred since the start of the simulation | Having a counter for the number of collisions may be necessary for some experiments with how different conditions effect things like how many collisions occur as if you were to increase temperature it would increase collisions. As for having the number of particles this would help the user know if they are closing in on the max amount and allow them to know in case they only wanted a certain amount for their research. |
| Effects of domain of simulation | The ability to choose what the background conditions that affect the particles are e.g. earth/vacuum | It allows for the user to choose if they want the conditions that affect the particles to simulate earth conditions with gravity and drag etc. However, if the user wants to simulate in conditions that do not have these factors affect it such as vacuum for a certain project/ piece of research they are able to do so. |
| Buttons | Having the ability to pause, reset, play and save | This would be a useful feature as then the user would have more control during the course of the simulation to potentially take screenshots use data at that point or take a second to understand what is currently happening as it may not be easy to understand with students and anyone using it as a learning resource. Then the ability to save and reset means that the other features such as templated can be made use of with it allowing for common experiments to be easily repeated and anything the user specifically may come back to can be done. In implementation this would also look to be implemented with graphics with a save button displayed with a floppy disc and the reset with a loop to help break across language barriers that the program will be constrained to. |

## Limitations of Project

|  |  |
| --- | --- |
| **Limitation** | **Justification** |
| Limited environments are one limitation of my project as for some stakeholders that may need to simulate conditions that are not a vacuum or earth are not possible. This may lead to some stakeholders having a limited use or no use for the project. | I would not be able to justify adding all conditions of planets in our solar system as that would be highly specialised and not necessary for most experiments of my stakeholders as well as this with the time constraint available it may not be possible to have all of them be added. |
| Other limitations include the inability to zoom as it may cause for some loss on detail which may be intriguing to stakeholders like students who may want to see a small section of the program such as a collision | It would not be implemented as it would make it harder to transfer the program to other stakeholders like video game developers and the ability to zoom is not particularly useful as the program is 2D not 3D. |
| One language is a limitation, this would cause for the only language that is used to be English which may cause for all stakeholders that do not speak English to find the program much harder to use and potentially not useable at all. | Although this is a massive limitation, under the time constraints of the project it would not be possible to implement the ability for the user to select their chosen language without it being at the detriment of other more notable features. |
| The project being 2-dimensional limits the project as a real collision between particles would be affected in the z axis, as a result this would then cause for the project to be limited in situations where the user was interested in a completely correct graphical representation as well as limiting features only available in 3D. | A 3-dimensional interpretation would not be necessary for most stakeholders who would want to use the project for its calculations to use in research or other projects, alongside this creating it in 3 dimensions would also cause for a higher set of system requirements that stakeholders like students may not have access to limiting it use as an educational tool. |

## System requirements

To justify the system requirements for the solution a project that is a similar solution will be used as a reference in algodoo which is a 2D physics simulator to create the system requirements. My solution does not have the same functionality as algodoo, which is used to create simulations with the ability for the user to create models by drawing, although both simulations do simulate collisions in 2D which leads to similar system requirements.

### Hardware

|  |  |
| --- | --- |
| **Requirement** | **Explanation** |
| CPU - dual core with a clock speed of at least 1.6 GHz | The CPU should be at least dual core with this allowing for the fetch-decode-cycle to be conducted as efficiently as possible and it would also require a clock speed of at least 1.6GHz to have the user be able to process the calculations/collisions in the required time frame and have it been displayed alongside all the extra graphics. |
| GPU- 512MB of available virtual memory and up to date drivers | To display the program with all collisions, particles and colours to the user graphical processing is required with a minimum of 512 MB’s virtual memory required. |
| RAM - 1GB | The ability to hold the program would need at least 1GB of RAM to hold the program itself and the operating system it runs on. |
| Secondary storage - 100MB | To download the program to use it when offline with the code and libraries required 100Mb’s of storage is required. |
| Peripherals - Mouse, keyboard and monitor | For the user to interact with the program it is required for them to see the results from the computer and results of the input with their keyboard and mouse. |

### Software

### 

|  |  |
| --- | --- |
| **Requirement** | **Explanation** |
| Windows, MacOS or Linux Operating system | To run the program an operating system is required that supports python |
| Python with the pygame and Tkinter libraries | The Program would require pygame for the graphical representation to be displayed to the user and Tkinter for the User Interface |
| Required drivers for the GPU etc | In order for the use of peripherals all updated drivers must be installed to use the program |

## Success Criteria

For my project to be considered a success in my vision I would look to look to implement the specific features listed below

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Justification** | **How to Evidence** |
| A main menu GUI that would appear when the user has loaded the program with this being windowed or full screen depending on what they choose | Having a main menu GUI allows for the stakeholders to easily navigate the program as well as acknowledge that the program has been started. Alongside this having it be displayed in windowed or full screen allows the user to have the ability to multi-task like open a document and record the results or fully focus on the program if it was for an educational purpose in full screen. | - Take a screen shot of the main menu when the project is opened  - Display the code that is associated with the section that creates the GUI |
| The ability for the user to decide when the program has loaded into the main menu what they want to use the program for with many different buttons giving choice to their purpose. | When the user has loaded the program, they have the ability to decide what they want to do. This could be in the form a dropdown of buttons labelled with things such as start, templates etc. | - Take a screen shot of the main menu with the buttons associated with text |
| Having the ability for the user to create, save and load templates of their simulations, alongside the ability for the template to be reopened when using the program with it being stored on local storage. | With this it allows for repeated conditions to be used as a lot of experiments will share the same conditions hence an ability to store this on local storage would be a useful feature that would need to be included. | - Show a screen recording on the main menu loading a template that is saved in files |
| Correctly associated buttons so that if the user wants to press a button it correctly is associated to a page for example when using a template, the template is loaded with the correct variables associated with it. | Without having buttons associated to the correct pages/parts of the program it would be completely dysfunctional as a user may attempt to load a template or create a new one and be loaded onto an incorrect window. | - A screen recording that clearly displays that the buttons work/ are associated with the correct page |
| A GUI that when simulating particle collision correctly displays the position of the particles on the screen as well as the resulting momentum after a collision with the correct velocity in proportion to each other accounting for mass. | Having a successful graphical interpretation means that the many stakeholders who may use the program to teach or learn would be able to understand what is happening. Without an interpretation of the calculations, it would not make sense and making sure that this is accurate would also be necessary to not habilitate any false meaning from an experiment from a student. | - Screen recording of the particles interacting after collisions |
| A counter that displays the number of particles that are currently on the screen alongside with the amount of collisions that has occurred since the start of the simulation | To allow the user to know the effect of the variables on collisions having a counter would be useful to demonstrate this, alongside keeping a track of the number of particles which means the user would find it easier to not reach the maximum to prevent any confusion. | - A screenshot of a running simulation that has the counters on the screen |
| A process that formats all calculations with this being in intervals of 1’s and put into a table that is then presented to the user when the simulation has ended so they can make use of the results of the simulation | The ability for the simulation to be made use of for calculations and research is the main ambition for the project as well as what many of the stakeholders require for experiments to be replicated and useable for research/projects. Without having calculations for the user there would also be no use for the project outside of an educational format. | - A screenshot of the calculations in a tabled format when the simulation is finished |
| The ability to have changeable variables available to the user such as mass and form of particles, ability to create groups such as small and large particles, add particles to the container. With this being done in a slider format or typing in the values with a text box | Customisation would be a necessary feature to allow for the project to be used this would mean that any set of conditions that are needed for the user can be inputted, for this to be missing would cause for the it to be useless as only templates that are set by me would be simulated and anything that a stakeholder may need to experiment with is not possible. | - A screenshot with sliders and textbox to the user on a sidebar |
| Buttons that allow the user to pause the simulation as it is happening as well as continue with all calculations remaining saved and a reset button that allows the user to reset all current conditions to the original conditions it started as. | To have this feature would allow for the user to take time away from the simulation and remain in the current condition as well as adding the ability to reset the conditions if this is required if something different is desired e.g. new conditions. | - A screen recording with evidence of the mouse pressing the button and the correlated action proceeding |
| Having a function that sets a limit to variables such as a certain number of particles/conditions that cannot be exceeded | Making sure that the program does not crash in consideration of the system requirements, a limit on the ability to simulate particles so that the program would not crash | - A screen recording that attempts to go past the particle limit and the user being unable to |
| Graphics that display certain features this could be a floppy disc for the save button or a pause Icon for pause | Making use of this would make it possible to justify the fact that the program would not be translated from English to any other language as this would be able to translate across language through pictures that are universal across language | - A screenshot with the icons displayed with graphics for the associated function |

# Design

## Development stages

|  |  |  |
| --- | --- | --- |
| Problem | Explanation of problem | Justification |
| 1 Python initialisation  Main menu concept | | |
| 1.1 Importing libraries | I will import the library of pygame. This would be used for the initialisation of the program | Using pygame would allow for the creation of the x and y dimensions that the program will be based on referenced in the success criteria as well as allow for the creation of a GUI for the user to navigate. This would be important for hitting the success criteria of a main menu GUI which pygame would be necessary for. |
| 1.2 Setting variables | To set the commonly occurring variables that will occur within the colour scheme as well as that would occur in calculations | To make the program easier to follow and save time the commonly occurring variables will be defined to begin with as well as to keep continuity with the theme of the program through a consistent colour scheme. Having this as a feature would also |
| 1.3 Create opening window and graphics | Upon opening the program, displaying a unique icon in the program to differentiate the program alongside having a windowed program loaded with a GUI shown to the user | Having the program load and display a unique icon to the user helps for the distinctiveness of the project to be branded to the user. Also having an opening window be displayed to the user would be necessary for the main menu to be used as well as help the user acknowledge they can use the program. Within my success criteria this would be hitting within the main menu GUI as this would also add to the appeal of the program and uniqueness |
| 2 Main menu functionalities | | |
| 2.1 Buttons | I will create associated buttons that will be linked with the correct pages such as buttons that will link with the template creation stage | Making sure the user has the ability to use correctly associated buttons so that the program can be used with full functionality, is integral for the user experience as well as the ability to use the program. |
| 2.2 Background | This would be the creation of graphics that would be associated with the program displayed to the user behind the buttons | To present the user with a unique theme that makes the program identifiable and distinguishable. |
| 2.3 Settings | For the ability for the user to choose the settings that the program will preside for will be accessible in the main menu | Granting the user, the ability to use the program as they wish is very important as they may require the windowed screen to be a certain resolution or certain colours not to contrast for colour blind people. This would also be important for reaching my own success criteria in allowing the user to use the program for what they wish on the main menu |
| 3 Class Creation | | |
| 3.1 Creation of particle class | Creating a class with the variables that are necessary for all the calculations implemented to correctly simulate particle collision within a single | Having particles be set under a class of variables would be necessary for the program to work as well as having it be easily modified, to not include this would mean the simulation would be very hard to program as they would have to be set under each function. Implementing this also would help reach me the success criteria in the user being able to change the variables and have calculations be completed |
| 3.2 Domain Class | For the conditions of the domain to be defined in a class to be called upon when needed with all conditions set for each | To allow for the simulation to work the domain in which the particles collide is required as well as this having this be set under a class allows for this to easily be used for calculations |
| 3.3 Creation of particle form class | To separate types of matter for particle simulation to create varying uses such as the classes for gas, solids and liquids | For the ability to have stakeholders with more uses for the project, having different forms of matter for particles that have different results to a regular solid would be beneficial. |
| 4 Particle Collisions | | |
| 4.1 Collison calculation | The template for the calculations that are needed for all the collisions that occur is set out so they can be reused | The simulation requires for the calculation of collisions to be of any use this is also needed in the form of a template so it can be reused for motion as well as for all matter types. |
| 4.2 Collision counter | Having a bar that is displayed that shows how many collisions have occurred starting from when the simulation starts | As some experiments or demonstration may look at the effect of variables on how particles may vibrate more causing for an increase in collisions, this would also link to my success criteria in helping me with the explicit criteria of a collision counter. |
| 4.3 Particle motion | Having the particles be correctly displayed relative to the resulting calculation of the velocity and displacement in the GUI presented to the user | A graphical solution to the calculation of particle collision would be compulsory for the project to be a success with stakeholders needing to visualise the calculations and receiving feedback based on calculations with a visualisation of the motion. |
| 4.4 Limits | To prevent the program crashing there will be limits on conditions, running time as well as the number of particles available at one time | Not having the program crash during the running time is required for the user to receive the results they need, in turn making use of limits allows for the program to not crash so the user is able to run the program and receive what is necessary is needed. Although this will cause for some stakeholders to not be able to simulate larger calculations for big projects, limits are necessary for it to be useable. |
| 5 Customisability | | |
| 5.1 Variable customisation | The variables in which all the simulation would rely on for calculations would be able to be modified this would be through a slider and textbox on the sidebar | Making use of variable customisation is imperative for the project to be usable as without it there would only be pre-set templates that may only be used by a small number of stakeholders this means the project would not reach my ambition and have a limited use. |
| 5.2 Class customisation | Giving the user the ability to create classes as they wish with all variables changeable available as well as being able to colour and name to differentiate the class | Having different classes able to be made means that the user is capable of creating a simulation with alternating interactions with large and small particles or different forms interacting for which class creation is required. |
| 6 Sidebar | | |
| 6.1 Formatting | The location of the sidebar should be allocated to the same position alongside stay proportional depending on the window size so that it takes up an adequate amount of space | For the sidebar to be used effectively it has to be formatted to the same amount and doesn’t take up an excessive amount as well as staying in this percentage despite being changed in window size. |
| 6.2 Variables | To allow the user to change the variables this would be implemented through the sidebar with a slider to vary with ease as well as a textbox to input the desired amounts | So that variables can be changed for them to be easily changed by the user having them placed on the sidebar would be the best option. |
| 6.3 Buttons | During the simulation the ability to pause and play would be necessary with this being displayed on the sidebar with graphics for each button | During the simulation it is needed for the user to be able to pause the simulation to take time away as well as this having them be displayed on the sidebar would be the most suitable location due to there not going to be any space in the domain of simulation for buttons. This would also be necessary of hitting my success criteria to allow the user to pause the simulation and use buttons. |
| 7 Templates | | |
| 7.1 Save | To allow for the user to save the current set of variables in their simulation to be reused as a template whenever they wish a button would be created | For templates as a feature to be used, the ability to save the template will be needed with this being in a form of a floppy disc so that is can be universal across language |
| 7.2 Loading | Alongside the ability for the user to save, I will add the ability to load the template this would be available to the user in the main menu | Loading the template is needed so that a user is able to use a recent project as they may have another purpose for this as well as making use of demonstrations for teachers easily. This also links to the success criteria of allowing the user to create and save templates on local storage. |
| 7.3 Edit template | I will allow the user to change a template during the simulation with this then also being able to be saved. | Editing the template will allow the user to create very specific conditions that may take a while to set meaning they are able to save time alongside setting a linear progression of a project that may require small conditions changed without having to start from scratch for every condition. |
| 8 Calculation formatting | | |
| 8.1 Data gathering | Once the simulation has concluded all the calculations will be gathered within 1’s intervals and entered to a table | Gathering the data to present to the user is vital for the project to be useful with stakeholders requiring the calculations for their potential research. |
| 8.2 Table formatting | It will be necessary to format the results in a table that can be saved/copied to be used for research or projects | Presenting the data that is gathered, into a table will be the easiest way for the user to extract the data as this can then be saved and referenced. Alongside reaching the success criteria of presenting data in a table format in 1’s intervals where this structure would lay out the groundwork for the stakeholders to use the results |
| 9 Ending the simulation | | |
| 9.1 Timer | A limit on the amount of time the program runs for will be used and the program will need to stop when it ends | Allocating a timer to the runtime of the program combats potential crashes due to a large runtime and gives a manageable timeframe with it also being able to be amended by the user as they may have other purposes for their program which require a larger timeframe |
| 9.2 Results | When the program finishes the table with all the results of the calculations are presented | Making sure that table of results displays at the correct time is very important so that the program functions correctly and the user receives what they want when needed. |
| 9.3 End screen | After the user has finished with the program and the results have been displayed | Demonstrating to the user that the simulation has ended is needed for the use to decide what they want to do such as start a new project or load a template. |

## Algorithms

The solution would require for the formatting as follows:

With this being used as a guideline for the basis of the project the individual algorithms used to attempt to form a complete solution would be following in the footsteps of the structure with it beginning with the menu algorithm.

Diagram

Description automatically generated

The next algorithm required would be the algorithm that would