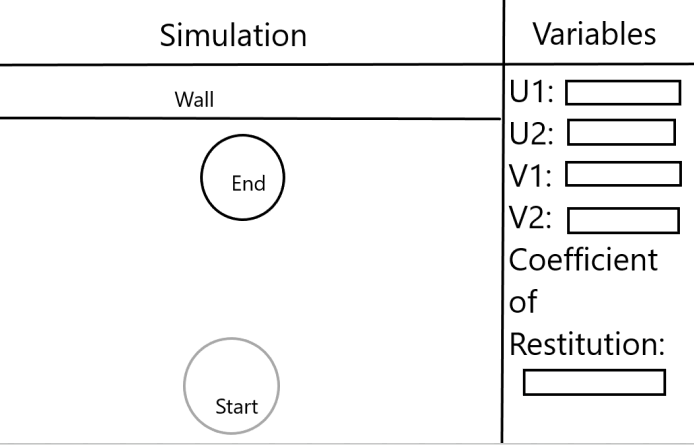
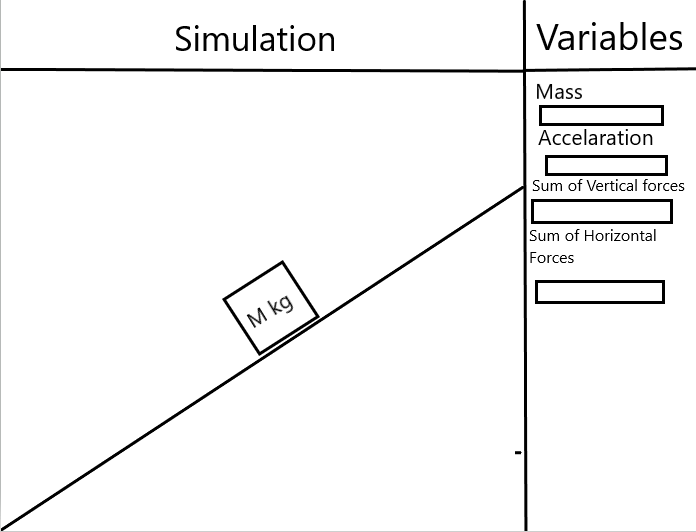
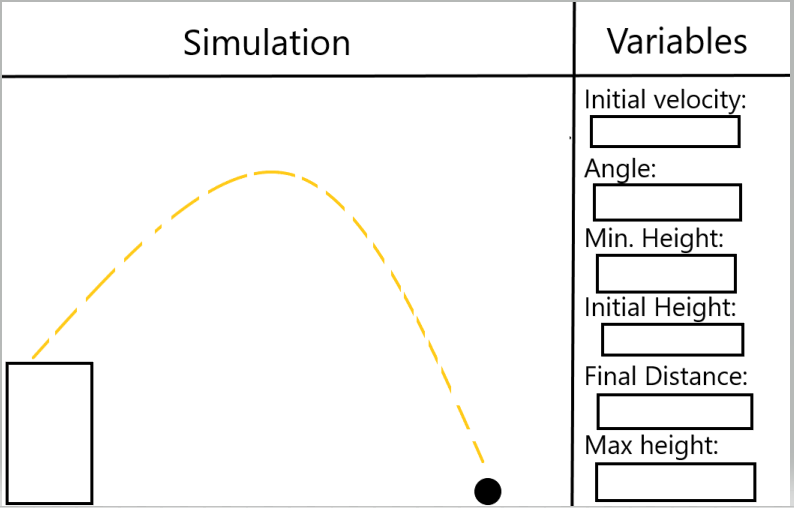
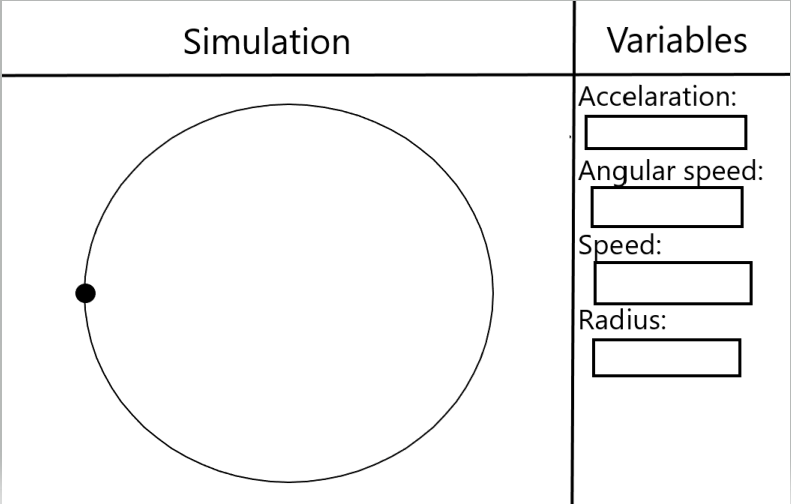
**Project description and objectives**

I **am** going to create a mechanics simulation aimed at A-level mechanics **students**. It will show students what is **happening** in their question in real time. This will help students to visualise problems as well as improving their physical intuition. It will allow students to choose between different question types **such** as Blocks on slopes, Collisions, **Circular** motion, Projectiles



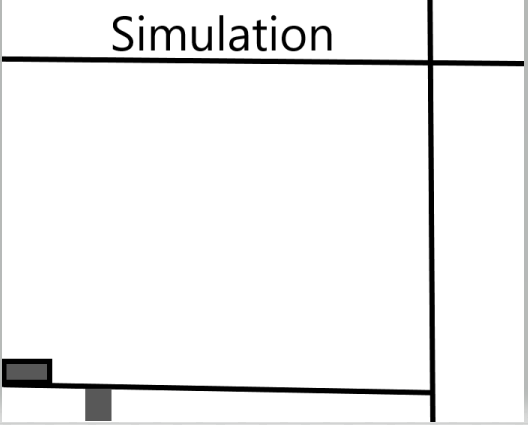
Blocks on slopes Collisions



Circular Motion Projectiles

**Objectives**

1. Users can choose from 4 different question types that are inspired by the A level mechanics syllabus. The program will allow the user to input **all** of the variables classically associated with the question type. It will not run if the program does not have enough information to sufficiently answer the question.
   1. There will be validation for certain inputs for example not having any negative masses
   2. In certain questions where variables should be answered in a strange way such as a wall in collisions having infinite mass and 0 velocity then there will be default ‘wall’ option for that or similar for other questions.
   3. The GUI will be very user friendly and make it easy to use
2. Users will be able to pause the simulation at any point, there will also be a drag bar along the bottom of t**he screen which will allow the user to change the time in the physical situation. This drag bar** will also be accompanied by an input box that allows for exact time finding.
   1. The drag bar should have a clear indicator of time to the millisecond
   2. Should allow for precise time input in an input box as well as having the bar to drag along



1. Throughout the simulation the program will output all the variables associated with the problem to the side of the simulation video. This will pause when the simulation pauses as well as changing when the time input is changed.
2. Once the simulation has finished, the user will be able to choose to either restart the same question with different inputs or decide to select a different question to run or to kill the program entirely.
   1. This should be done through a clear drop-down menu
   2. While to drop **down** menu is up the simulation should continue to do whatever it has been told to do in the background – this **reduces** downtime.
3. The program should produce graphs at the end at the users request.
   1. If the user wants a graph they can choose between different variables plotted against time
   2. The **graphs** will be accurate and come with **accompanying** data

**Extension objectives**

1. Add different types of questions as well as variation on questions such as **projectiles** colliding or questions with non constant acceleration.
2. Give the user the **ability** to change variables whilst the simulation is **running** allowing for **real** time showing of different speeds or masses on the physical situation

**Key Critical areas**

1. Calculations for the different problem types – creating a base template function for each of the question types and adapting differently phrased questions to all fit the template.
2. Handling multiple objects with labels in PyGame
3. Handling collisions using Pymunk and clearly having different masses affecting the process.

**Research**

I sent out a questionnaire to a number of current further and single mathematicians and asked them the following questions in order to get a better understanding of problems within the discipline.

The answers were as follows:

