INTEGRATIVE PROJECT IN COMPUTER SCIENCE AND MATHEMATIC
420-204-RE

PROJECT DELIVERABLE 2

Anthony Monaco, M-Amar Kseibi, Youssif Khalifa and Mark Antoun

presented to Dr. Yi Wang

February 24 2023

Table of contents:

Project plan

Tasks Breakdown, Timeline and Tasks Assignment

1. Identify the task, or user stories, to be implemented, assign them to a team member and estimate their durations. (Add more rows if it is necessary)

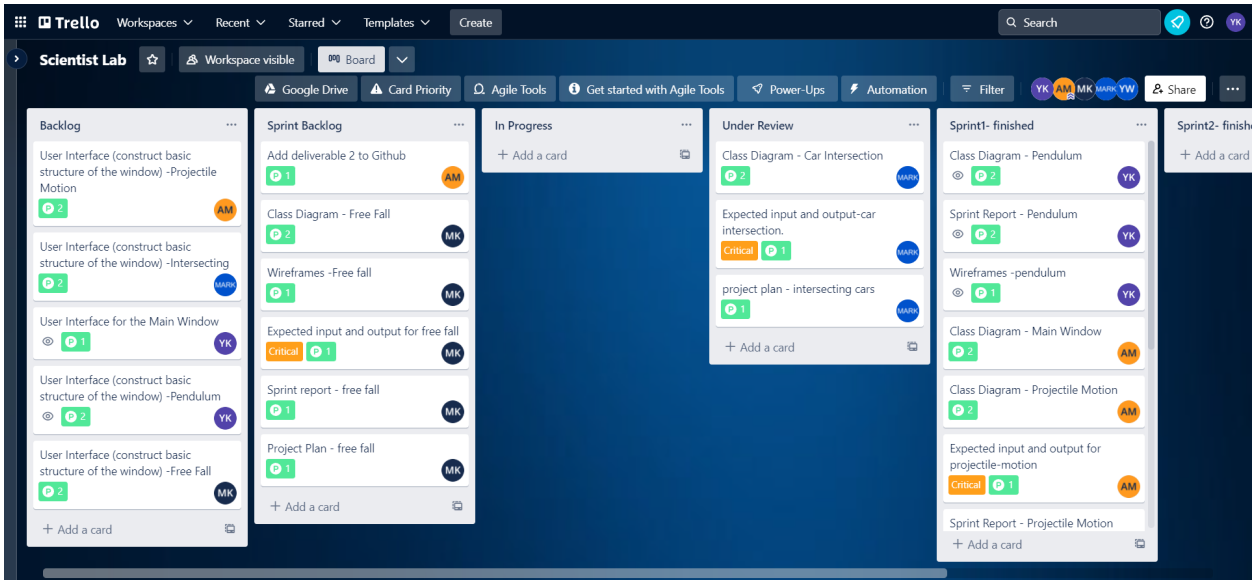
Task/User Story	Start Date - End Date	Assigned to
Create the project file and upload it to GitHub	February 25 - February 26	Anthony
Implement the welcoming window	February 26 - February 28	Youssif
Develop the class that will include all the kinematics formulas.	February 26- February 28	Mark
Create window layout in scene builder for car simulation with needed slider variables	March 1 - March 3	Mark
Implement road and cars to the pane	March 3 - March 8	Mark
Implement the animation to make the car move	March 8 - March 20	Mark
Implement buttons to control the animation	March 20 - March 22	Mark
Make sliders functional to read user input	March 22 - March 25	Mark
Make live statistics feature for both cars	March 25 - March 27	Mark
Implement the graph feature for the car simulation	March 27 - April 10	Mark
Make graph feature functional	April 10 - April 20	Mark
Implement stopwatch for car simulation	April 20 - April 23	Mark

Task/User Story	Start Date - End Date	Assigned to
Create window layout in scene builder for projectile motion with needed slider variables	Feb 26 - Feb 28	Anthony
Connect controls of variable to scene builder	Feb 28 - March 6	Anthony
Create methods which use variables to determine all the specifications of the projectile motion	March 6 - March 15	Anthony
Create the animation of the balls movement	March 15 - March 30	Anthony
Create bin which appears randomly	March 30 - April 15	Anthony
Detection of ball and bin	April 15 - April 25	Anthony
Create window layout in scene builder for pendulum with needed slider variables and buttons	February 26 - February 28	Youssif
Create the controller class and connect it to the class loader	February 26 - February 28	Youssif
Implement the logic for the animation for the bob	March 4 - March 18	Youssif
Implement the logic for the string	March 18 - March 21	Youssif
Create the GraphController class and implement the logic to obtain the current position of the bob to draw the graph	March 21 - April 11	Youssif

Task/User Story	Start Date - End Date	Assigned to
Create a graph class that will display the graph of the current potential energy of the bob according to his position	April 11 - April 25	Youssif
Create a graph class that will display the graph of the current kinetic energy of the bob according to his position	April 25 - April 30	Youssif
Create the basic structure of the simulation in scene builder	February 26 - March 5	Amar
Develop the connection between scene builder controls and java	February 26 - March 5	Amar
Make buttons and sliders functional	March 5 - March 19	Amar
Implement stopwatch feature and manage the output	March 19 - April 9	
Implement the logic for the animation	April 9 - April 23	Amar
Implement the logic for the graphs	April 23 - April 30	Amar

2. Log the tasks or User Stories in GitHub issues.

3. Create a Trello board and add all team members and me to the board.

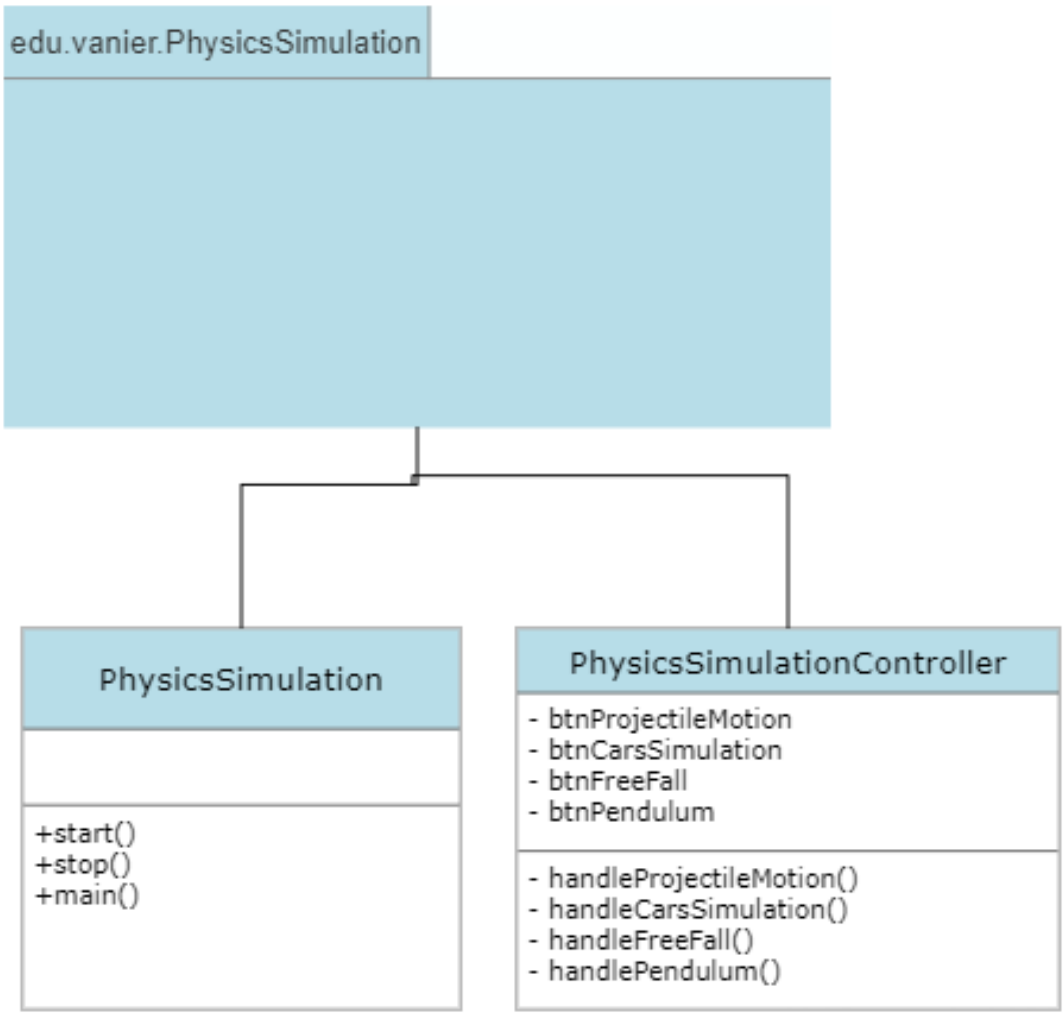

Trello
Workspaces
Recent
Starred
Templates
Create
Search
Scientist Lab
Workspace visible
Board
Google Drive
Card Priority
Agile Tools
Get started with Agile Tools
Power-Ups
Automation
Filter
Share
Backlog
User Interface (construct basic structure of the window) -Projectile Motion
User Interface (construct basic structure of the window) -Intersecting
User Interface for the Main Window
User Interface (construct basic structure of the window) -Pendulum
User Interface (construct basic structure of the window) -Free Fall
Add a card
Sprint Backlog
Add deliverable 2 to Github
Class Diagram - Free Fall
Wireframes -Free fall
Expected input and output for free fall
Critical
Sprint report - free fall
Project Plan - free fall
Add a card
In Progress
Add a card
Under Review
Class Diagram - Car Intersection
Expected input and output-car intersection.
Critical
project plan - intersecting cars
Add a card
Sprint1- finished
Class Diagram - Pendulum
Sprint Report - Pendulum
Wireframes -pendulum
Class Diagram - Main Window
Class Diagram - Projectile Motion
Expected input and output for projectile-motion
Critical
Sprint Report - Projectile Motion
Add a card
Sprint2- finish
Add a card

Class Diagram

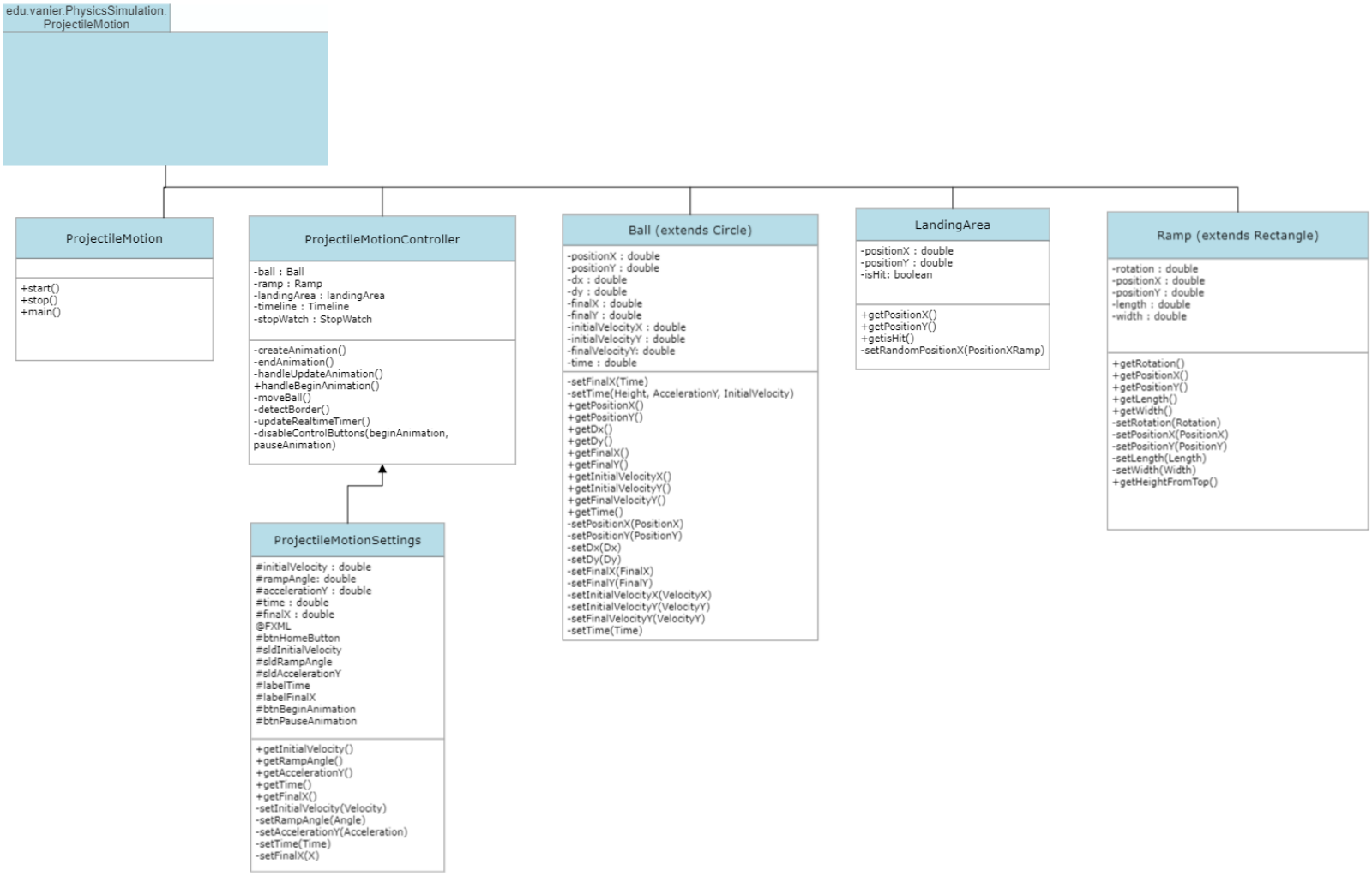
Include a class diagram composed of the key classes that you will be implementing.

https://cloud.smartdraw.com/?nsu=1

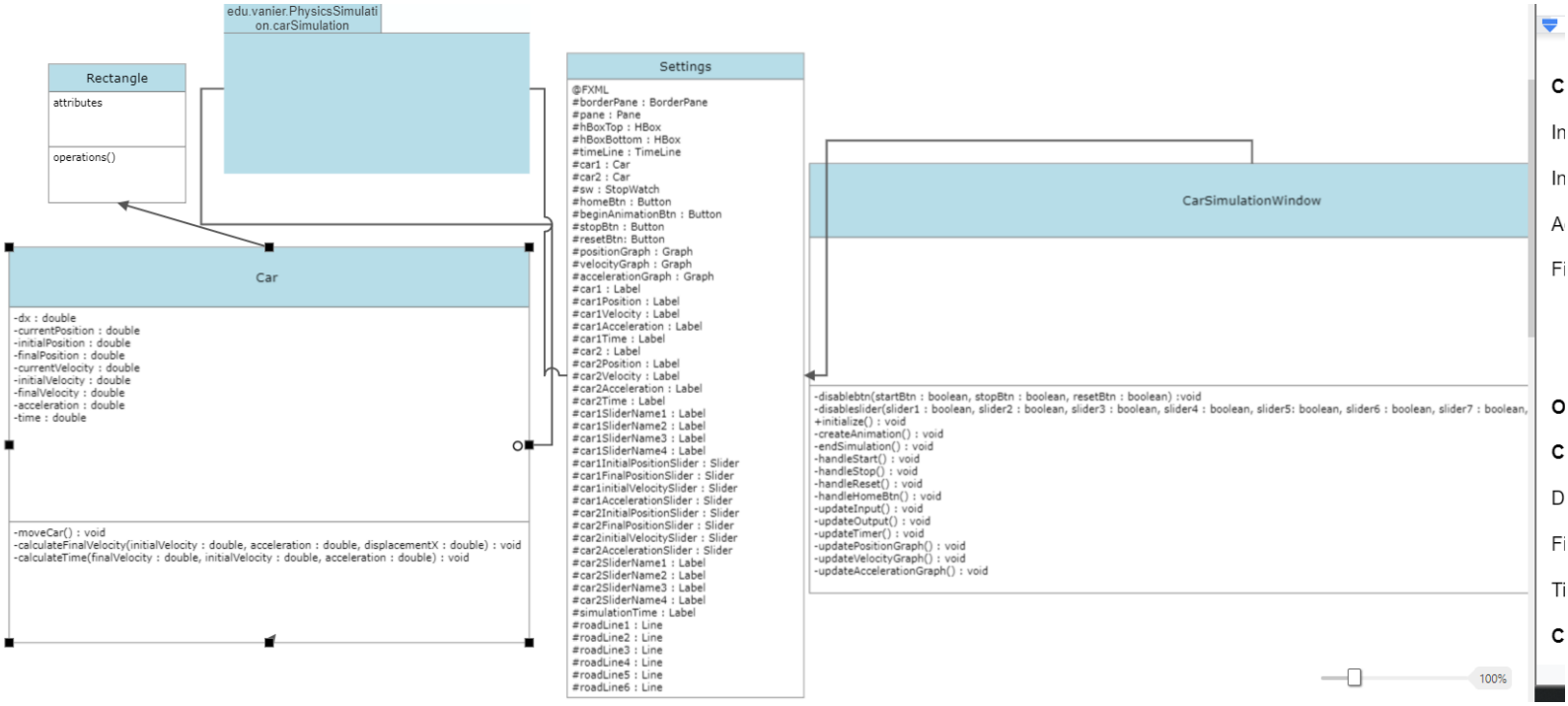
Main Package UML



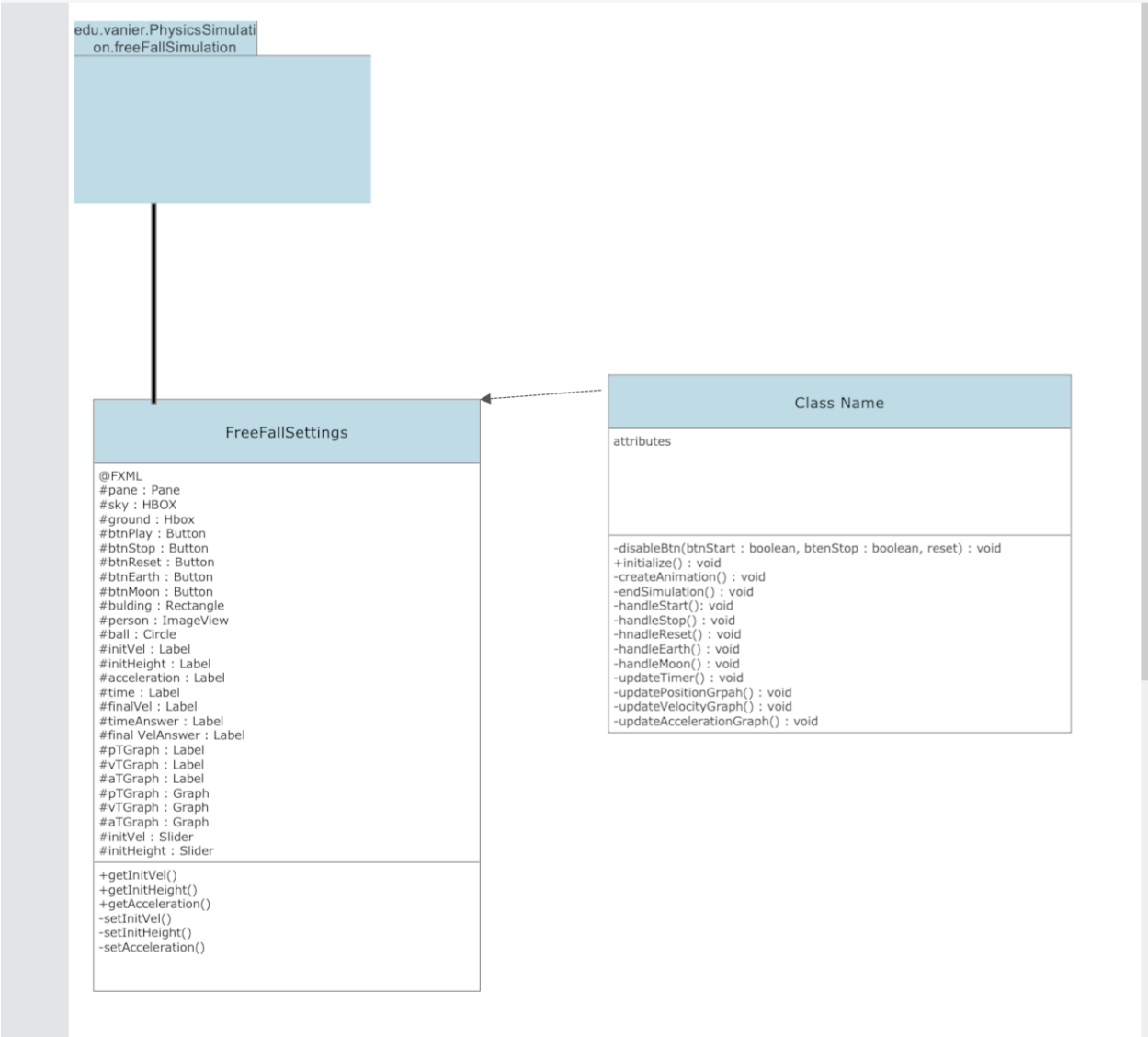
Projectile Motion Class Diagram



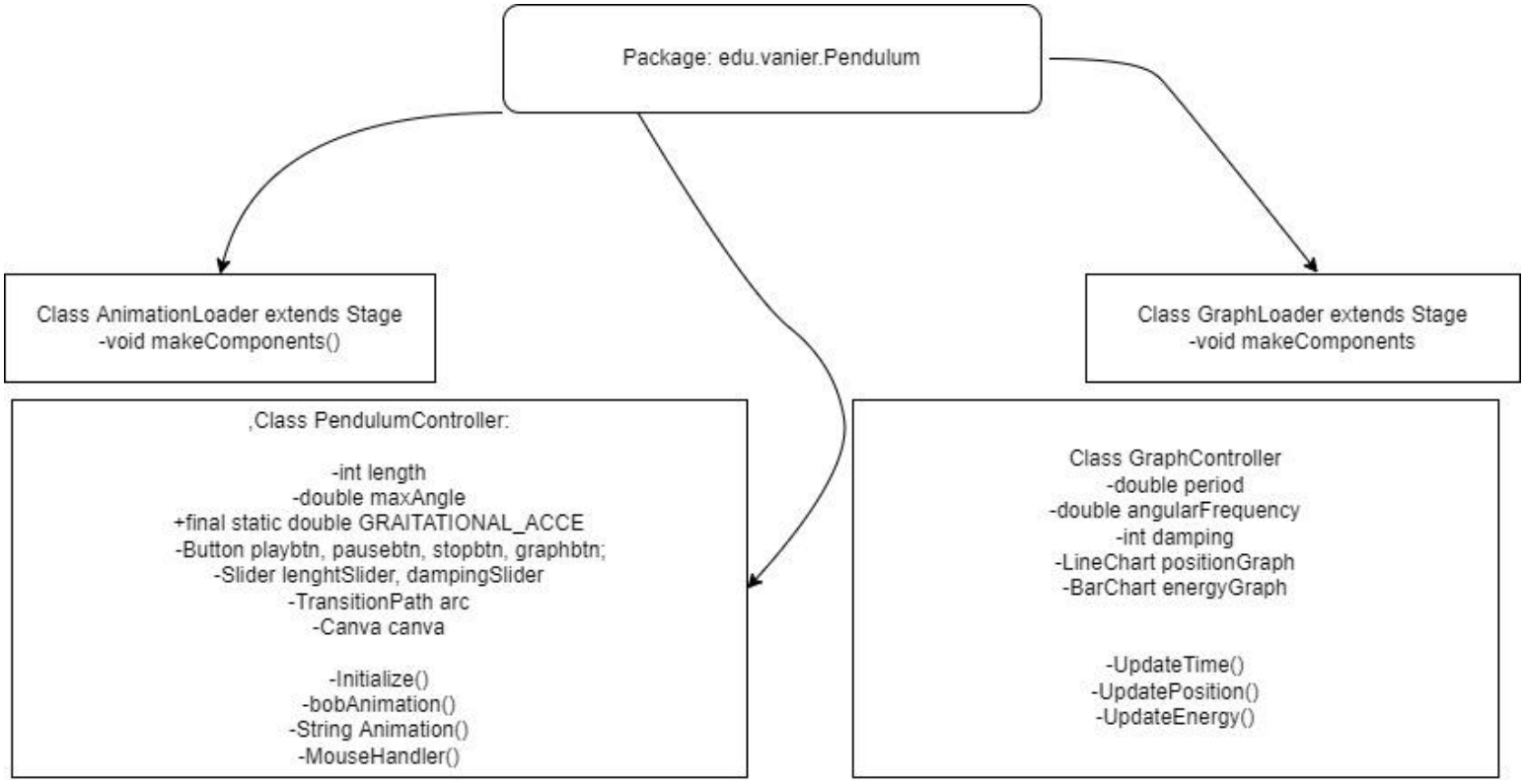
Cars Simulation



Free Fall Diagram



Pendulum Diagram



Sample Input and Output Data Grid

Include a grid showing sample input and corresponding output data values that will be used to showcase the operating program.

Car simulation:

Input:

Car 1:

Initial position : 20m

Initial velocity:10 m/s

Acceleration: 2 m/s^2

Final position : 70

Car 2:

Initial position : 10m

Initial velocity:5 m/s

Acceleration: 2 m/s^2

Final position : 70

Output:

Car 1:

Displacement: 50m

Final velocity: 17.32 m/s

Time:3.66s

Car 2:

Displacement: 60m

Final velocity: 16.27 m/s

Time:5.63s

Please note that the cars will never meet each other.

Projectile Motion:

Default preset values: ($a_x = 0$, Δy = constant, $X_i = 0$, Random landing area placement)

Input:

-Angle of ramp: 45°

-Initial Velocity: 15 m/s

-a_y = 9.8 m/s

Output:

-Visualization of the balls projectile motion

- Time: 5.3s

-Final position x = 5 m

-Win/Lose if ball lands in bin

Free Fall simulation:

Input:

-Initial height = 25 m

-Initial velocity = 10 m/s

-Acceleration (depending on the position of launch: earth or moon): earth (9.8 m/s^2)

Output:

-Time = 1.458 sec

-Final velocity before touching the ground = 24.297

-Position vs time graph

-Velocity vs time graph

-Acceleration vs time graph

Pendulum simulation:

Input:

-Maximum angle in degrees: 15

-Length of the string in cm: 30

-Mass of the bob in grams: 20

-Damping constant (goes from 0 to 10, where 10 is the maximum damping):2

Output:

-The animation of the simple harmonic motion

-Angular frequency in rad per seconds: 5.71

-Period in seconds: 1.1

-Angle function: x = [Ae^ (− bt/ 2m)] cos(ωt + ϕ) = [15e^ (− 50t)]cos(5.71t)

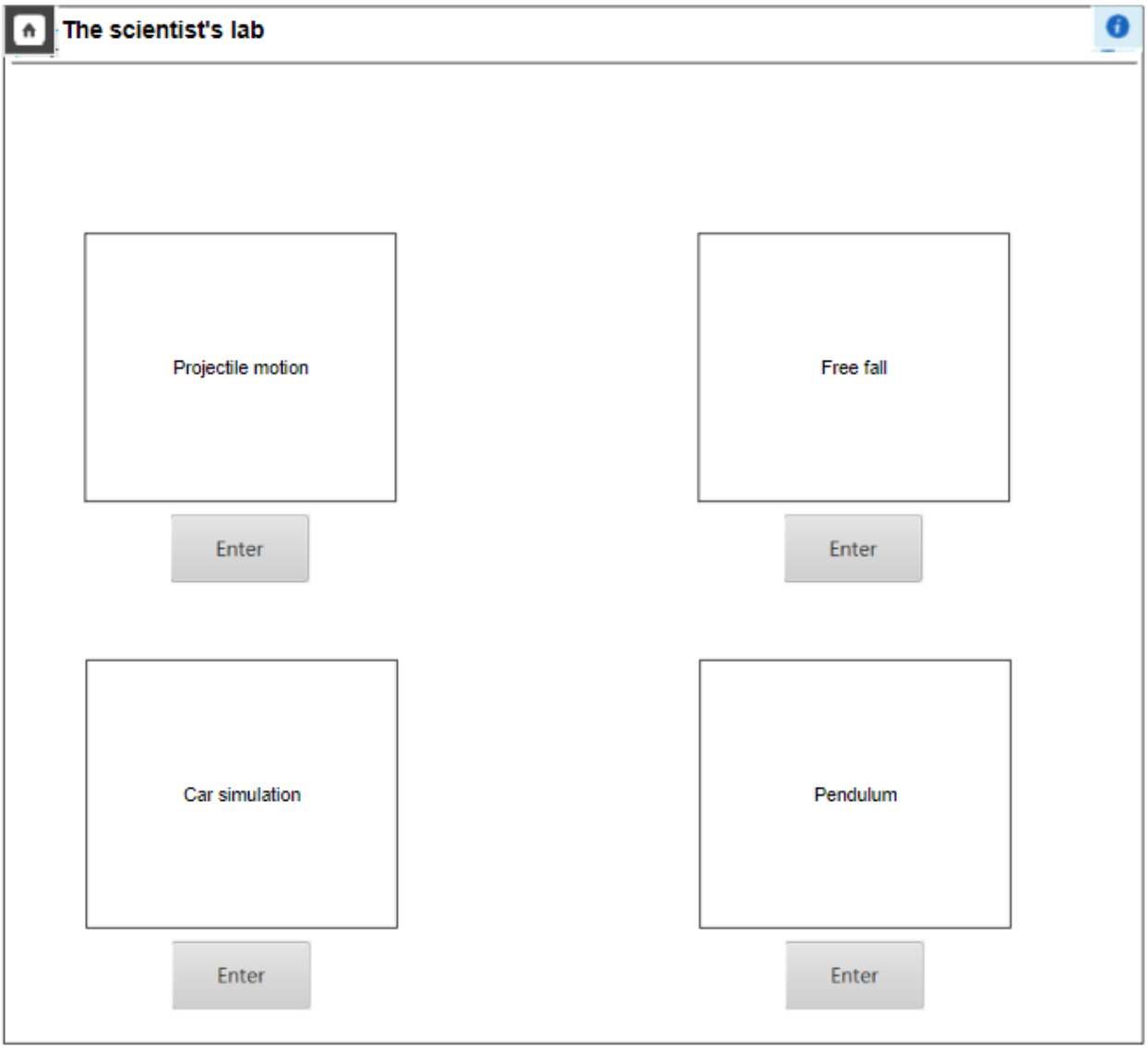
Wireframes

Include wireframes of the user interface. It is important that the user interface has an area where important data is shown to the users so that they understand what they are looking at. That would include a comprehensive title and updated data values that are reflected in the animation or chart area.

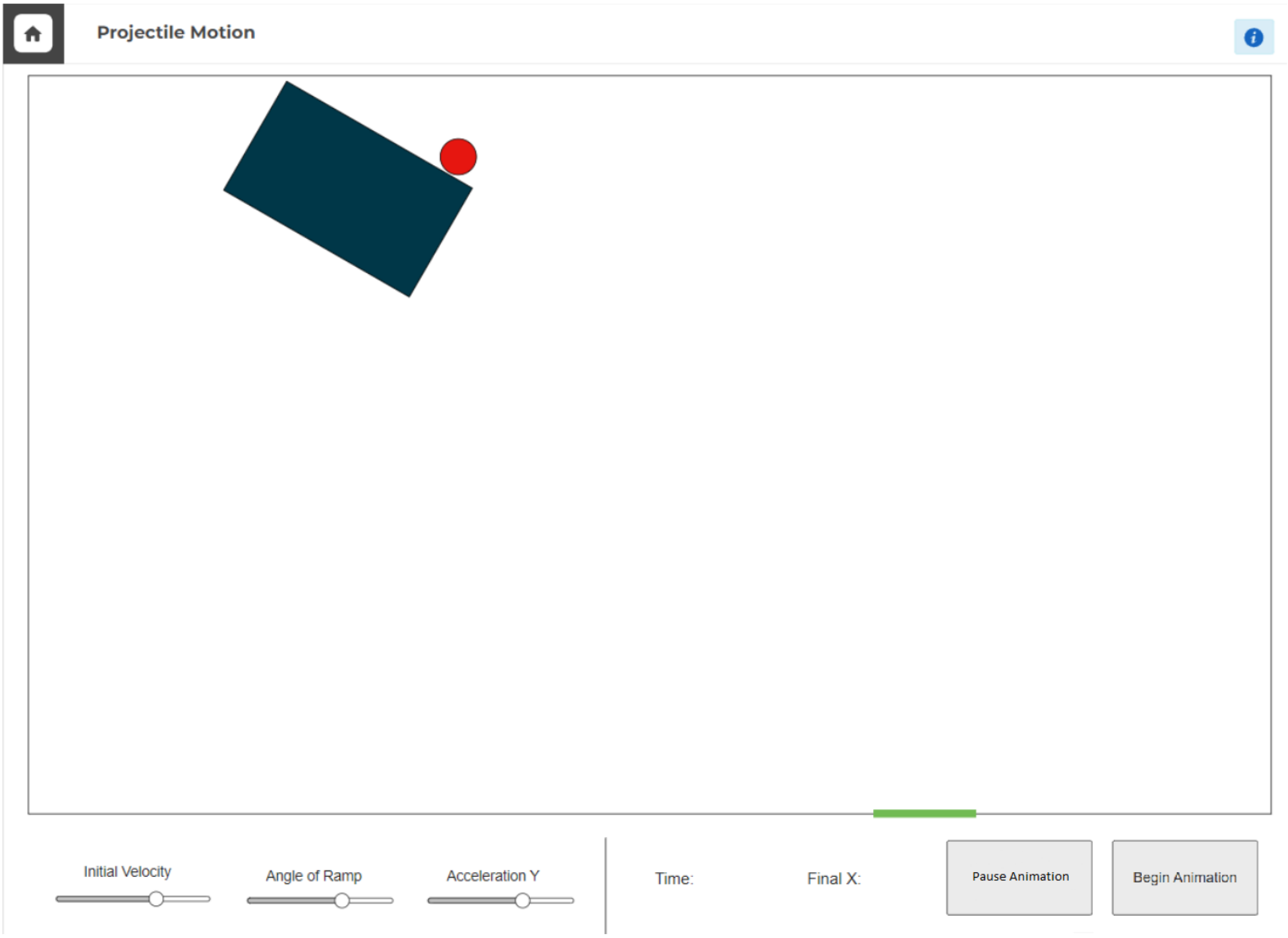
Use this for the wireframe

https://app.moqups.com/ZbKtTildR0twmOPSQcTxnSxCzUtAzs0o/edit/page/a3fbc0074

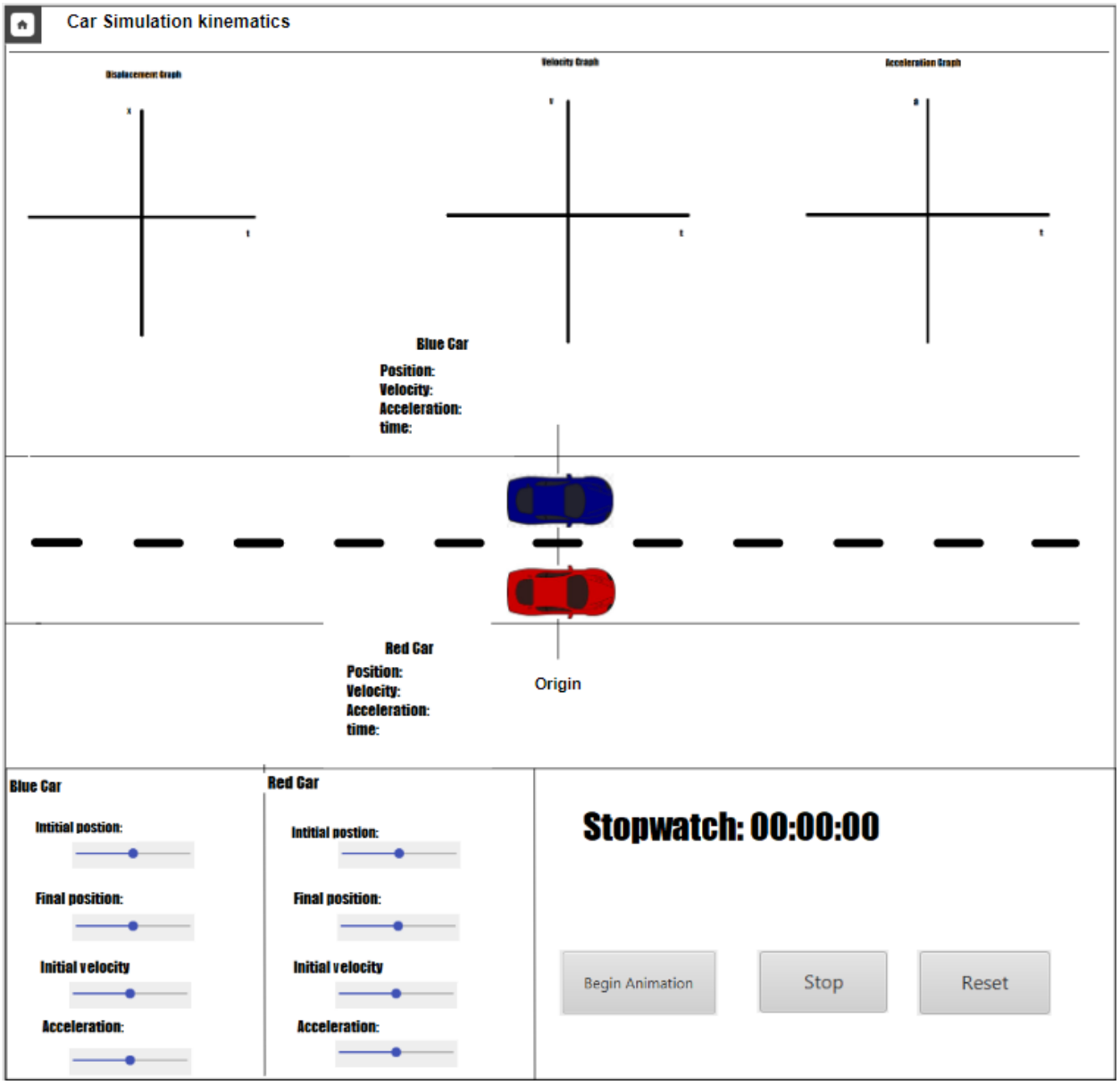
Main Window



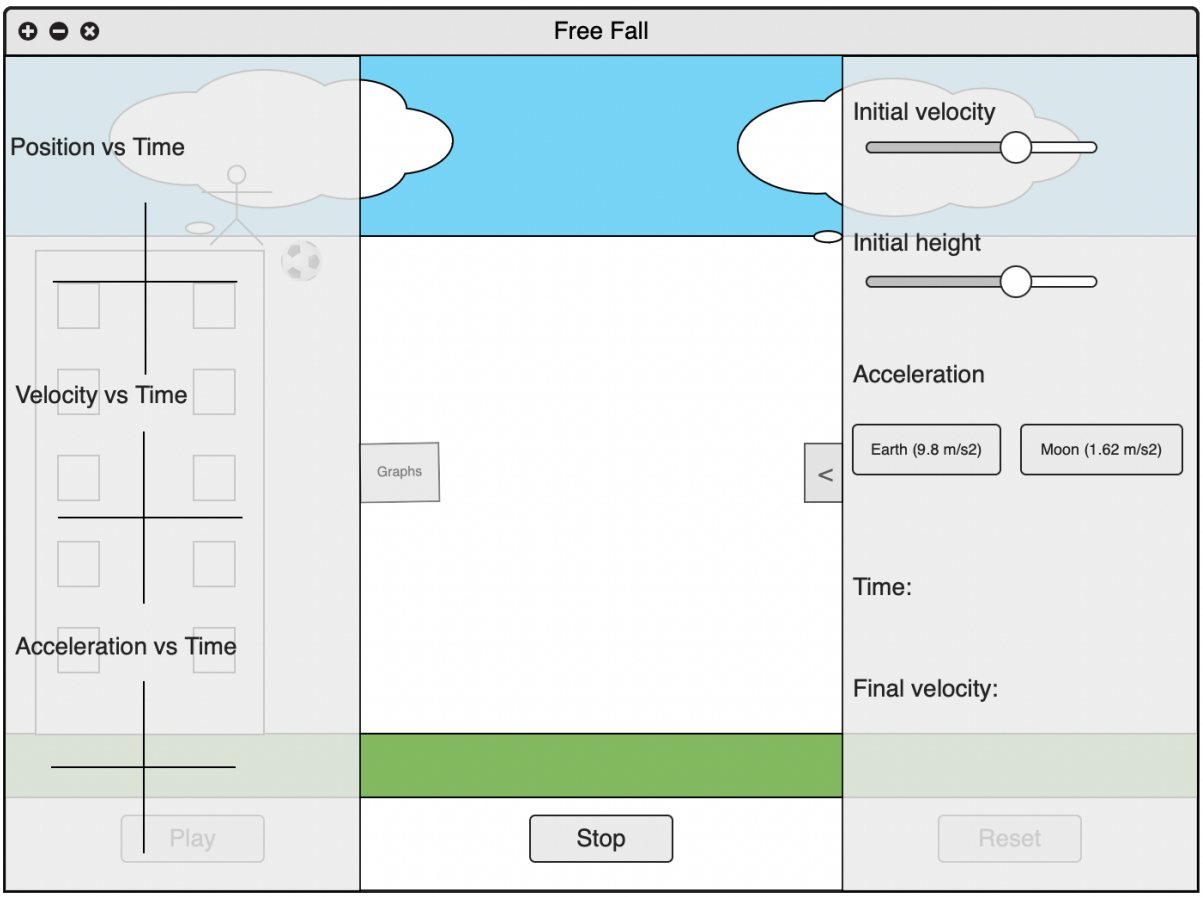
Kinematics Window



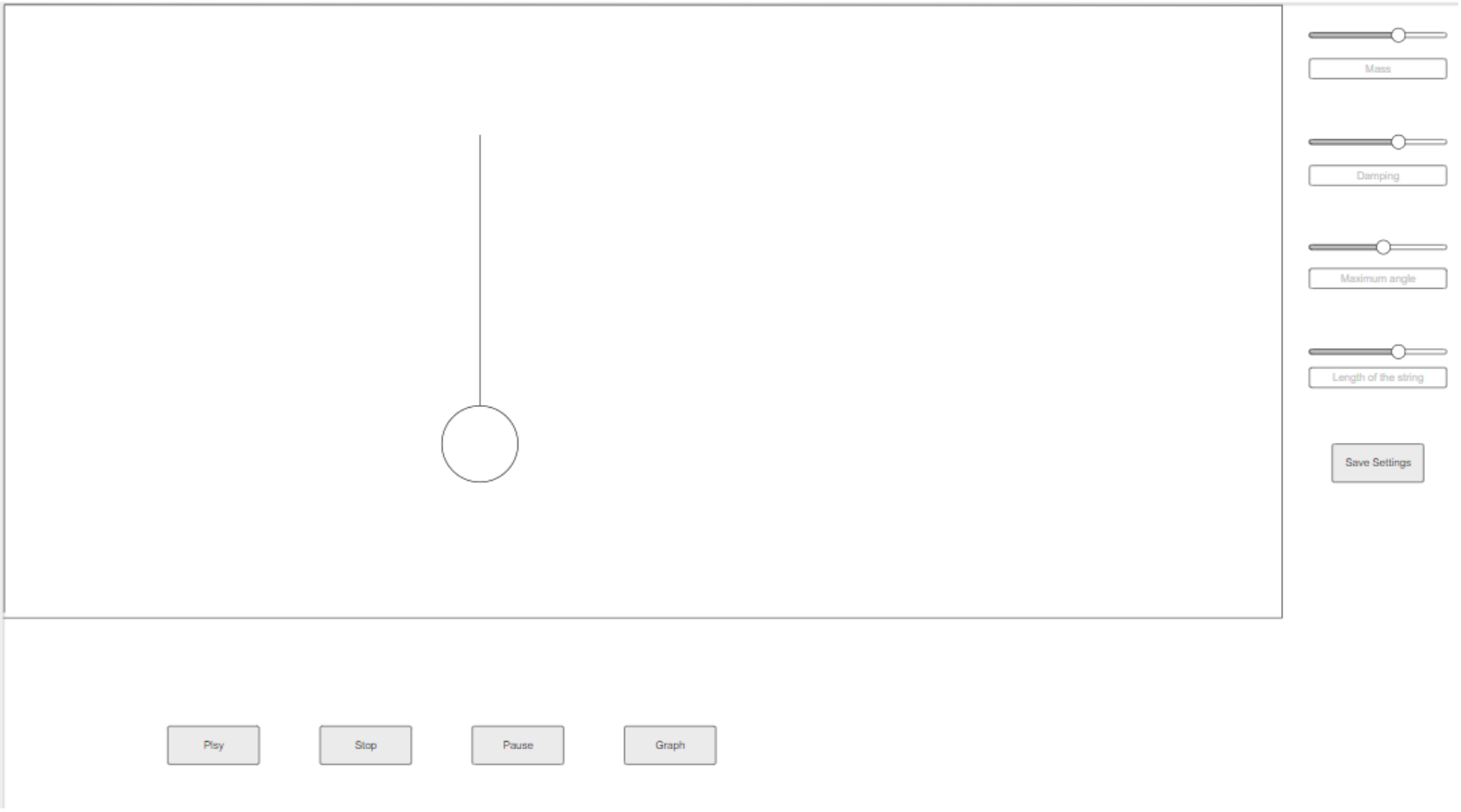
Car Simulation Window:


Car Simulation kinematics
Displacement Graph
Velocity Graph
Acceleration Graph
Blue Car
Position:
Velocity:
Acceleration:
time:
Red Car
Position:
Velocity:
Acceleration:
time:
Origin
Blue Car
Intitial postion:
Final position:
Initial velocity
Acceleration:
Red Car
Intitial postion:
Final position:
Initial velocity
Acceleration:
Stopwatch: 00:00:00
Begin Animation
Stop
Reset

Free Fall:


Free Fall
Position vs Time
Velocity vs Time
Acceleration vs Time
Graphs
<
Initial velocity
Initial height
Acceleration
Earth (9.8 m/s2)
Moon (1.62 m/s2)
Time:
Final velocity:
Play
Stop
Reset

Pendulum:


Mass
Damping
Maximum angle
Length of the string
Save Settings
Play
Stop
Pause
Graph

Sprint Report

Team Sprint

Received stories	Resolved stories	Carry over stories	Blocked stories
Add deliverable to github (1)	Add deliverable to github (1)		
Class Diagram - Main Window (2)	Class Diagram - Main Window (2)		
Wireframe - Main Window (2)	Wireframe - Main Window (2)		
Total Points : 5	**Total Points :5**	**Total Points : 0**	**Total Points : 0**

Mark - Cars Simulation

Received stories	Resolved stories	Carry over stories	Blocked stories
Sprint Report - intersecting cars(2)	Sprint Report - intersecting cars(2)	None	
Class Diagram - Car Intersection(2)	Class Diagram - Car Intersection(2)	None	
Expected input and output-car intersection.(1)	Expected input and output-car intersection.(1)	None	
project plan - intersecting cars(1)	project plan - intersecting cars(1)	None	
General Wireframe (main window)(1)	General Wireframe (main window)(1)	None	
Wireframe -intersecting(2)	Wireframe -intersecting(2)	None	
Total Points :9	**Total Points :9**	**Total Points :0**	**Total Points : 0**

Anthony - Projectile Motion

Received stories	Resolved stories	Carry over stories	Blocked stories

Project Plan (3)	Project Plan (3)		
Class Diagram (2)	Class Diagram (2)		
Expected input and output (1)	Expected input and output (1)		
Wireframe (2)	Wireframe (2)		
Sprint Report (1)	Sprint Report (1)		
Total Points : 9	**Total Points : 9**	**Total Points : 0**	**Total Points : 0**

Youssif - Pendulum

Received stories	**Resolved stories**	**Carry over stories**	**Blocked stories**
Project Plan (3)	Project Plan (3)		
Class Diagram(2)	Class Diagram (2)		
Expected I/O (1)	Expected input and output (1)		
Wireframe (2)	Wireframe (2)		
Sprint Report (1)	Sprint Report (1)		
User Interface for the main window (2)		User Interface for the main window (2)	
Total Points :11	**Total Points :9**	**Total Points :2**	**Total Points :**

Ammar - Free Fall

Received stories	**Resolved stories**	**Carry over stories**	**Blocked stories**
Project Plan (3)	Project Plan (3)		
Class Diagram (2)	Class Diagram (2)		
Expected I/O (1)	Expected I/O (1)		
Wireframe (2)	Wireframe (2)		
Sprint Report (1)	Sprint Report (1)		
Total Points : 9	**Total Points : 9**	**Total Points :**	**Total Points :**