

**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY



**SOICT**

**Capstone OOP project report:**

---

**FORCE AND MOTION SIMULATOR**

---

**Guided by:**

Associate Professor Tran The Hung

**Group members:**

Nguy Quang Son - 20225998

Nguyen Long Vu - 20226006

Le Minh Nam - 20226059

# Contents

<b>Contents</b>	<b>2</b>
<b>Abstract</b>	<b>3</b>
<b>1 Project Assignment</b>	<b>3</b>
<b>2 Mini-Project Description</b>	<b>3</b>
2.1 <i>Mini-Project Overview</i> . . . . .	3
2.2 <i>Mini-Project Requirement</i> . . . . .	3
2.3 <i>Use case Diagram</i> . . . . .	4
<b>3 Design</b>	<b>5</b>
3.1 <i>General class Diagram</i> . . . . .	5
3.2 <i>object package</i> . . . . .	6
3.3 <i>force package</i> . . . . .	7
3.4 <i>Application Package</i> . . . . .	7
<b>4 Reference</b>	<b>8</b>

## 1 Project Assignment

We will provide an update on each member's completed tasks by detailing the work they contributed to this project.

Number	Full Name	Student ID
1	Nguy Quang Son	20225998
2	Nguyen Long Vu	20226006
3	Le Minh Nam	20226059

Name	Assignment	Percentage Contribution
Nguy Quang Son	<ul style="list-style-type: none"> <li>● Package: force, object</li> <li>● Controller</li> <li>● UML Diagram</li> <li>● Report and Presentation</li> </ul>	33%
Nguyen Long Vu	<ul style="list-style-type: none"> <li>● Package: force, object</li> <li>● Controller and GUI</li> <li>● Class Diagram</li> <li>● Report</li> </ul>	33%
Le Minh Nam	<ul style="list-style-type: none"> <li>● Package: Application</li> <li>● Controller and GUI</li> <li>● Slide</li> <li>● Report</li> </ul>	33%

## 2 Mini-Project Description

### 2.1 Mini-Project Overview

In this project, we are assigned to design an application that visualizes the Force and Motion Simulator. This game applies the laws of motion in physics to realistically simulate the movement of cylinders and cubes. Players can manipulate forces, friction, and mass to observe how these objects respond in various scenarios, providing an interactive way to understand fundamental physics principles.

### 2.2 Mini-Project Requirement

For a better understanding of the project, here are the requirements that we need to follow during the process:

1. **Bottom left area:** The user can choose and drag an option from the object menu which include Cube-shaped object and Cylinder-shaped object
2. **Bottom area:** The place where the user can change the applied Force, Pause, Continue or Reset the simulation.
3. **Bottom right area:** Which include two sliders to change the Friction coefficients.

- 4. **Right & left area:** To display the value, direction of Forces, Mass, Speed and Acceleration of the Object.
- 5. **Center Area :** Display the movement of the Object.

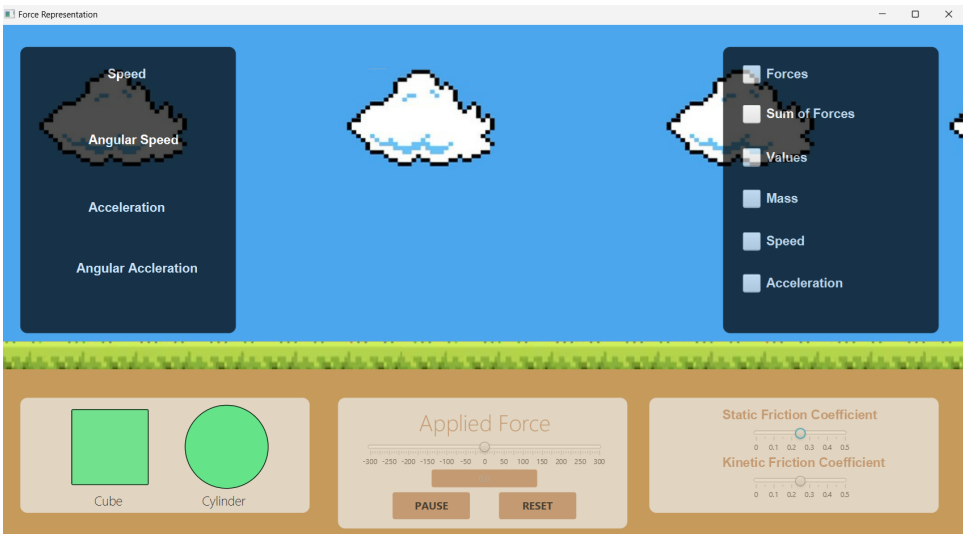


FIGURE 1. *Sample GUI*

2.3 Use case Diagram

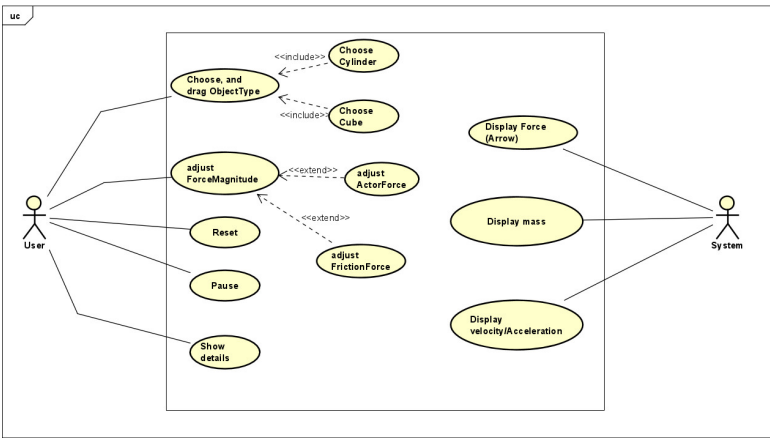


FIGURE 2. *Use Case Diagram*

- Our General Class Diagram includes three main package
  - **object package:** Which includes two type of object (Cube and Cylinder) and its attributes, methods.
  - **force package:** Which includes two type of Force: Unchangeable-Force and Changeable-Force.
  - **Application package:** Which includes main class to run the App and Controller class to modify App's features.
- Relationships between classes:
  - **Inheritance :** The Cube and Cylinder Class inherit the attributes and method from ObjectType Class
  - **Association:** Direct association between the Controller Class and the force elements, ObjectType and Surface.
  - **Implementation:** The Controller class implements the interface Initializable.

## 3.2 object package

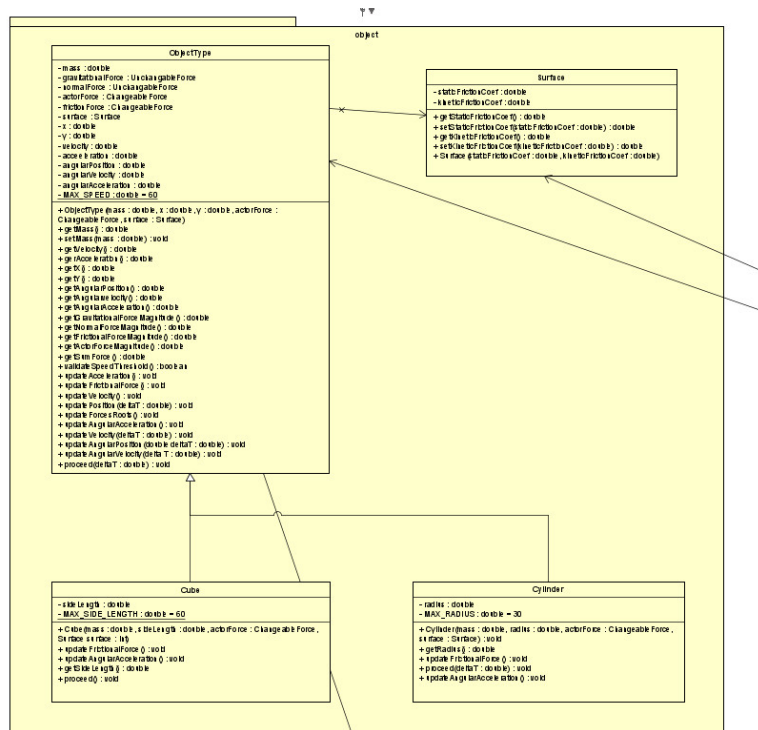


FIGURE 4. Object Package Diagram

### Objecttype Class

- This is the abstract class which includes the attributes: mass, gravitationalForce, normalForce, actorForce, frictionForce,... And also have methods to get those values for the use of other Classes.
- The Objecttype Class also has two inherited classes for specific cases: *Cylinder* and *Cube*.

### Surface Class

- This class has only two attributes: staticFrictionCoef and kineticFrictionCoef to present the friction coefficient of the surface.
- This class also has methods to get and set the value of the surface's friction.

### 3.3 force package

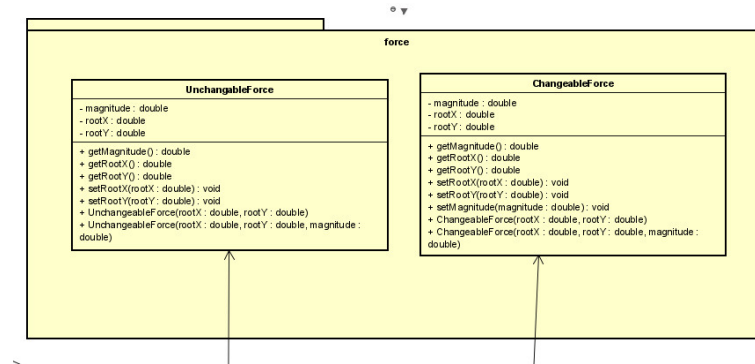


FIGURE 5. Force Package Diagram

- There are two classes in this package: NonchangeableForce and ChangeableClass.
- Those class has the same atributes, and method. However the NonchangeableForce can not modify the magnitude of the Force (Gravitational Force and Normal Force)

### 3.4 Application Package

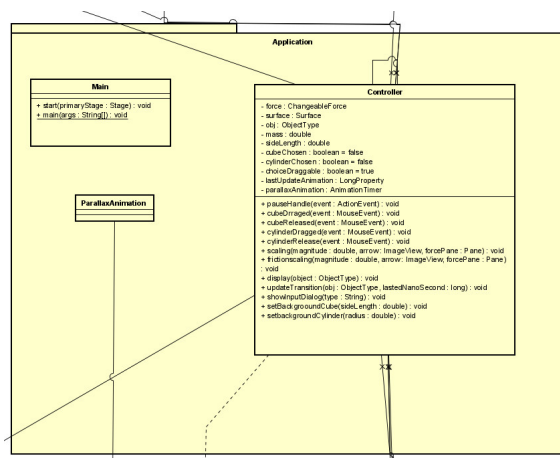


FIGURE 6. Application Package Diagram

### Controller class

- This class is the controller for the application : Force and Motion Simulator, managing the interface, app state transistions and user actions.
- This class has the following atributes:
  - **force**: Represent the ChangeableForce
  - **surface**: Represent the Surface with its friction coefficient
  - **lastUpdateAnimation**:
  - **parallaxAnimation**: To calculate the time of the animation.
  - **mass**: Represent the mass of the object
  - **sideLength**: Represent the sideLength of the object
  - **obj**: Represent the object
- This class has the following methods:
  - **void initialize(URL location, ResourceBundle resources)**: This is used for setting up the initial state of controller when the file FXML is loaded.
  - **void resetBtnPressed()**: This is used to set the application to it's initial condition.
  - **void pauseHandle(ActionEvent event)**: This is used to pause the simulator.
  - **void cubeDragged(MouseEvent event), void cubeReleased(MouseEvent event)**: This is used to choose and cancel the Cube object.
  - **void cylinderDragged(MouseEvent event), void cylinderReleased(MouseEvent event)**: This is used to choose and cancel the Cylinder object.
  - **void scaling(double magnitude, ImageView arrow, Pane forcePane)**: Adjust the arrow according to the magnitude of the general force.
  - **void frictionalscaling(double magnitude, ImageView arrow, Pane forcePane)**: Adjust the arrow according to the magnitude of the friction force.
  - **void display(ObjectType object)**: This is used to display the characteristics of the object.
  - **updateTransition(ObjectType obj, long lastedNanoSecond)**: This is used to update the movement of the Object
  - **showInputDialog(String type)**: This is set for users to enter the values of the Objects.
  - **void setBackgroundCube(double SideLength), void setBackbroundCylinder(double Radius)**: This is used to display the Object.

### Main Class

- This class is used to run the Application: Force and Motion Simulator.

## 4 Reference

- Sample GUI Reference: [https://phet.colorado.edu/sims/html/forces-and-motion-basics/latest/forces-and-motion-basics\\_en.html](https://phet.colorado.edu/sims/html/forces-and-motion-basics/latest/forces-and-motion-basics_en.html)