**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**



**OBJECT ORIENTED PROGRAMMING**

**MINI PROJECT REPORT**

**INTERACTIVE SIMULATION OF THE COMPOSITION OF FORCES**

**Instructor: Nguyen Thi Thu Trang**

|  |  |  |
| --- | --- | --- |
| **Student name** | **Roles** | **Student ID** |
| **Trần Ngọc Dung** | **Member** | **20194742** |
| **Đào Quang Dương** | **Member** | **20194747** |
| **Nguyễn Đinh Duy** | **Member** | **20190096** |

**Hanoi, 7/2022**

1. **Mini-project description**

* Project create a simple interactive simulation for demonstrating Newton’s laws of motion.
* The system includes three components: one main object, the surface, forces which apply to object.
* The user can control all the components as:

+ Select the shape of the object by dragging and dropping it in the center of the screen

+ Click right mouse to open, close menu to change the parameters of object

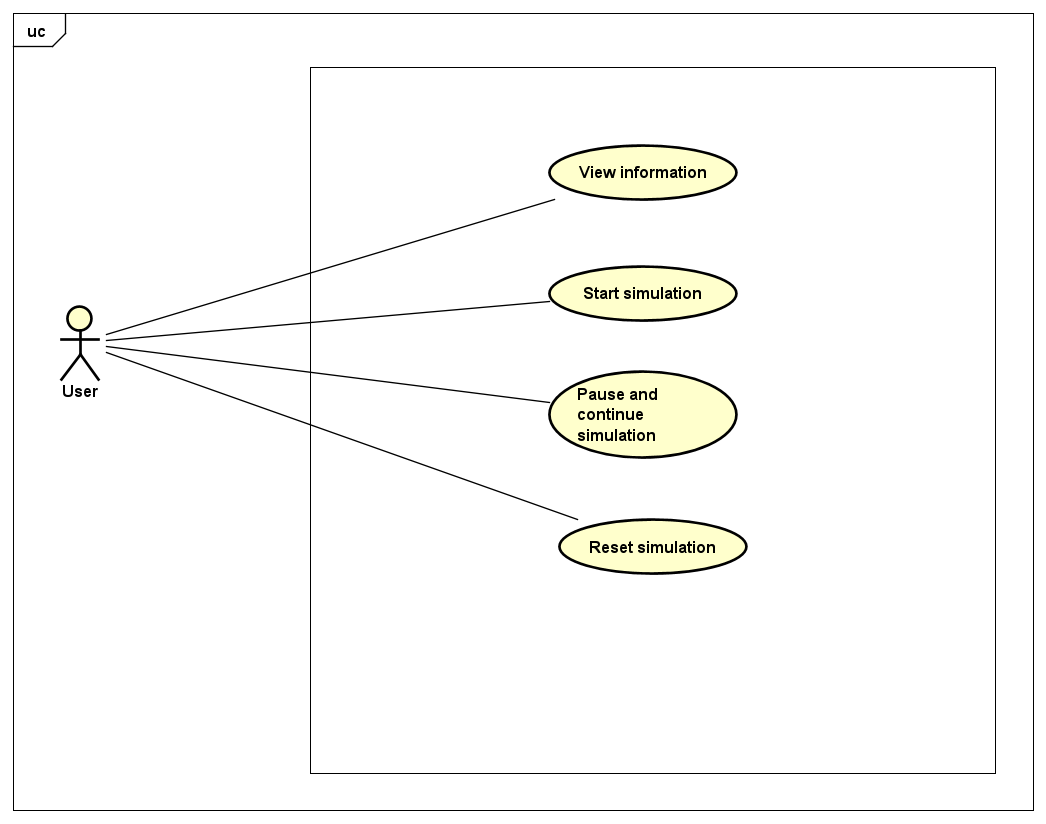
+ Change the external force parameter

+ Change coefficient of friction

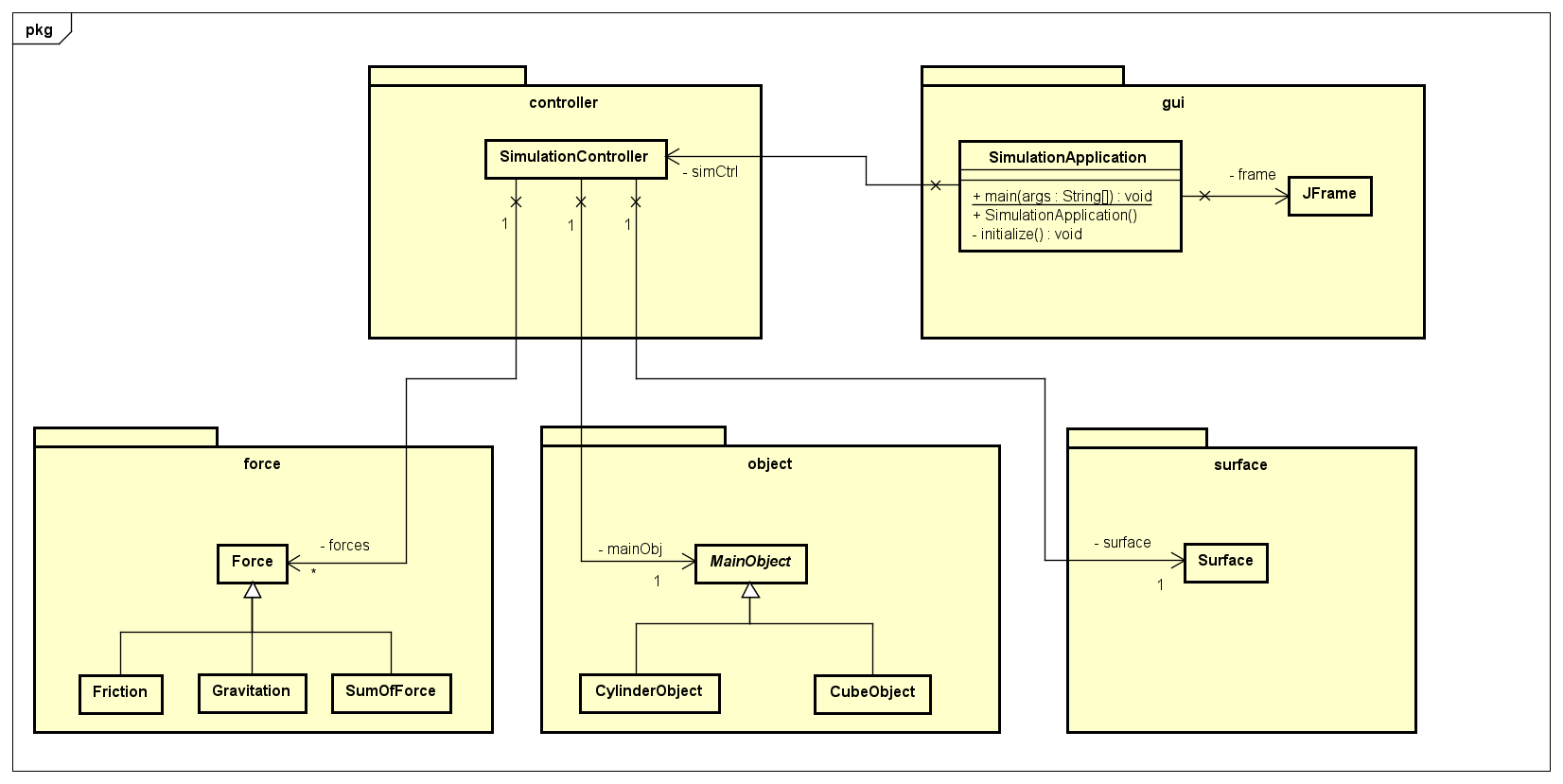
+ Checkbox in the upper corner to select the parameters you want to see details

+ pause: pause the program, continue: continue, reset: reset: run the program again.

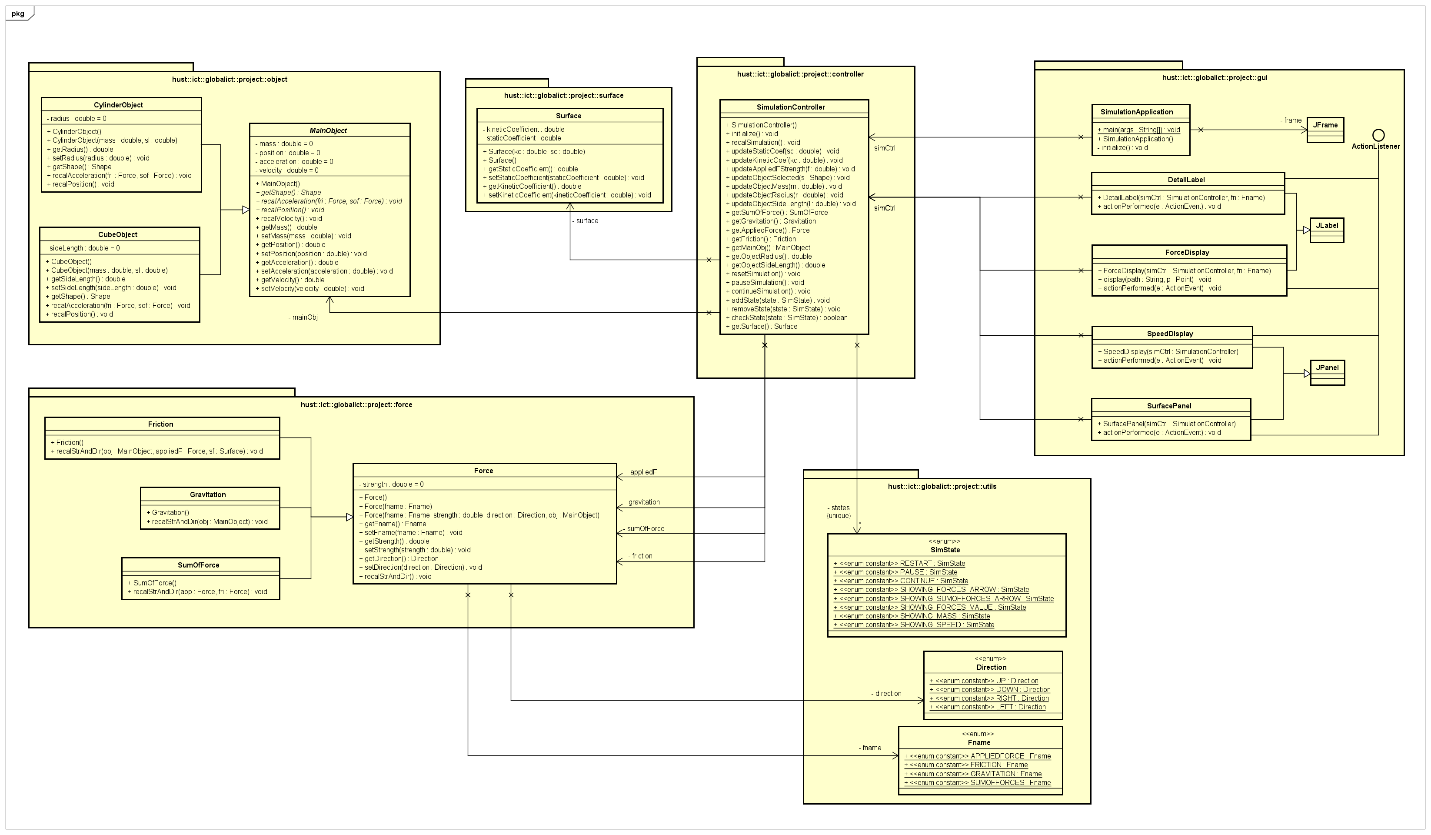
* Program will calculate forces, sum of force, acceleration, current velocity, current position every 1 second.
* Use case diagram :

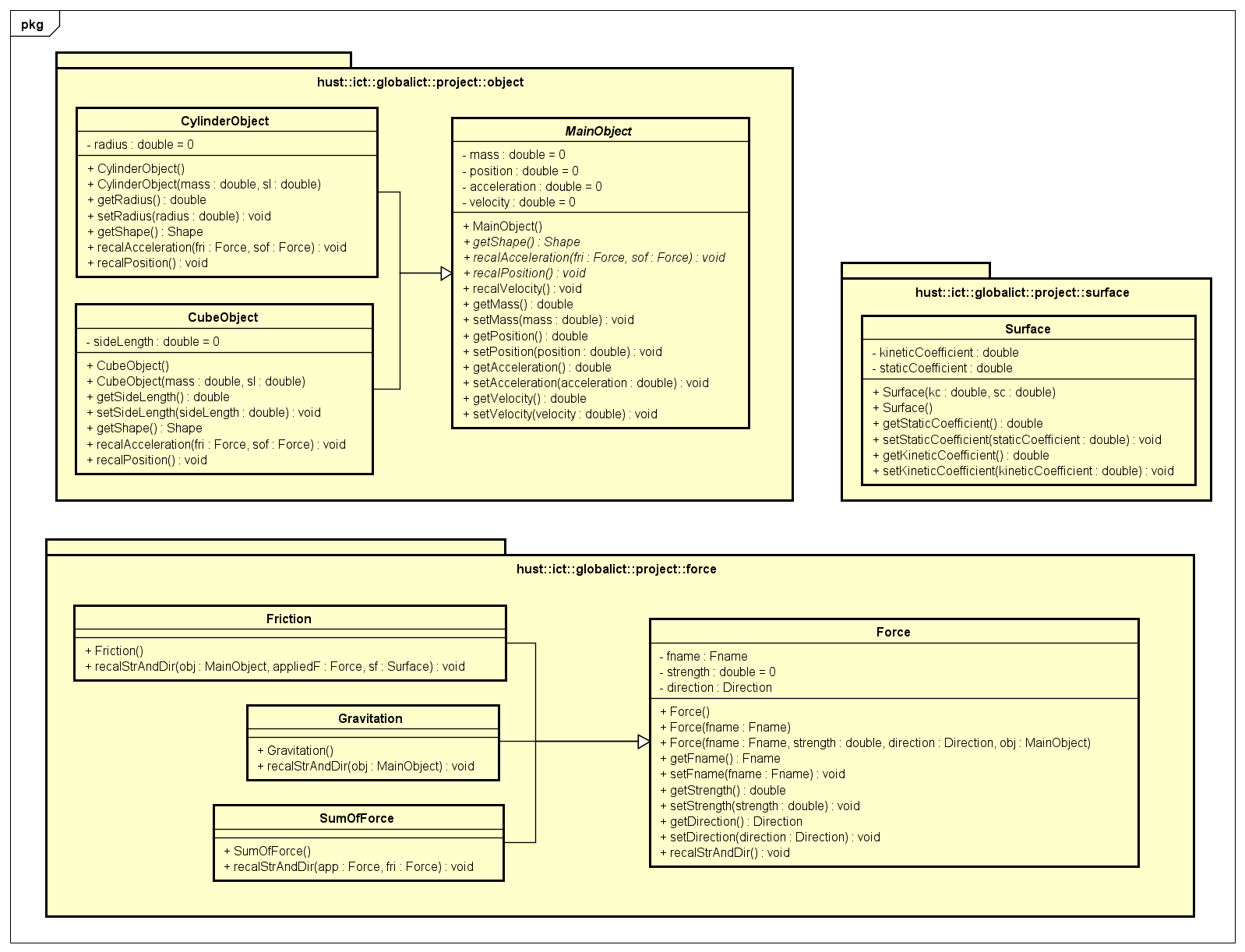


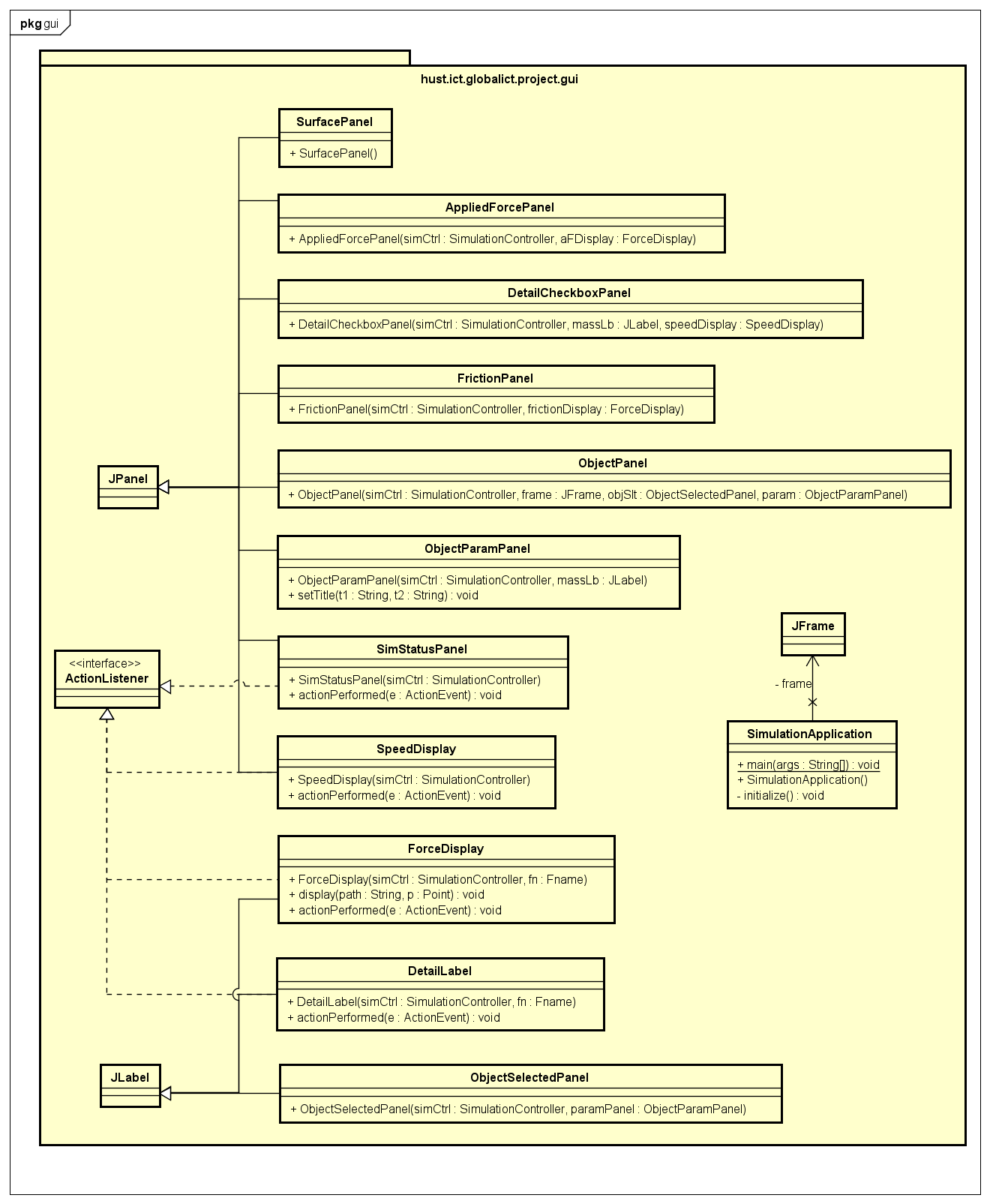
1. **Classes and Packages**
2. **Class Diagram**
   1. **General class diagram**

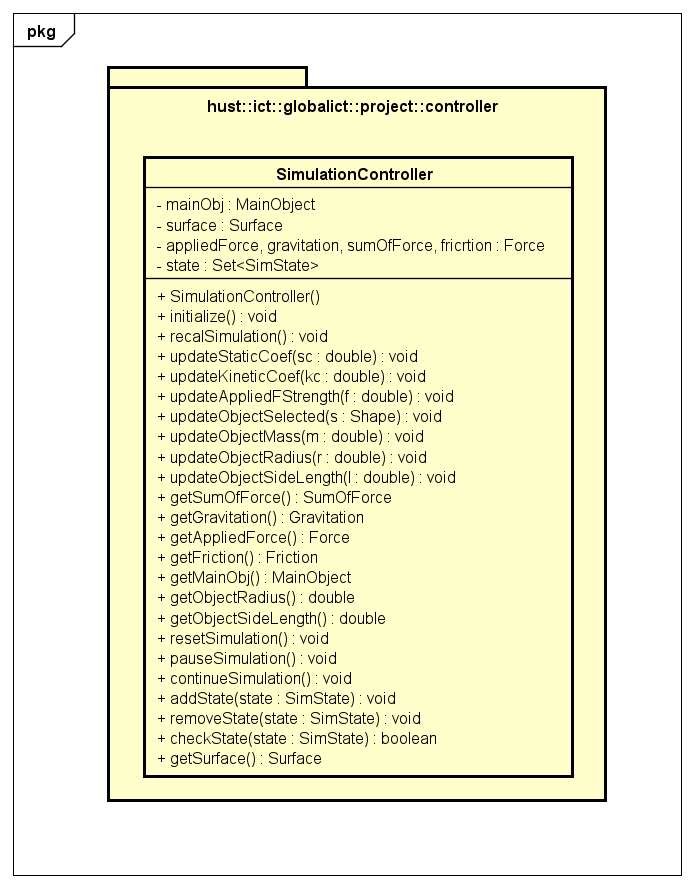
****

**1.2. Detail class diagram**

****

****

****

****

* Explanation of the design: Describe the relationships between classses:

+ Inheritance:

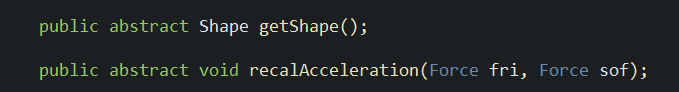
* CylinderObject Class and CubeObject Class inherit MainObject.
* Friction Class, Gravitation Class, SumOfForce Class inherit Force Class.
* DetailLabel Class, ForceDisplay Class inherit JLabel.
* SpeedDisplay Class, SurfaceDisplay Class inherit JPanel.

+ Association:

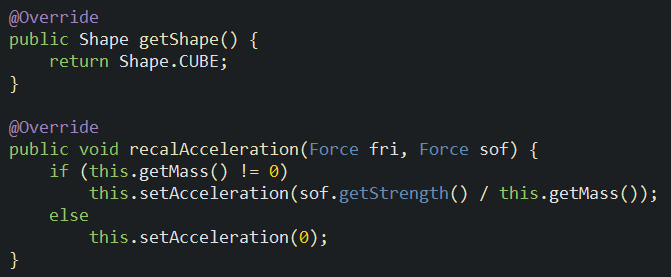
* SimulationController has 1-1 relationship with Surface, MainObject.
* SimulationController handle some forces, 1-many with Force
* SimulationApplication is controlled by 1 SimulationController, 1-1 relationship.

+ Polymorphism:

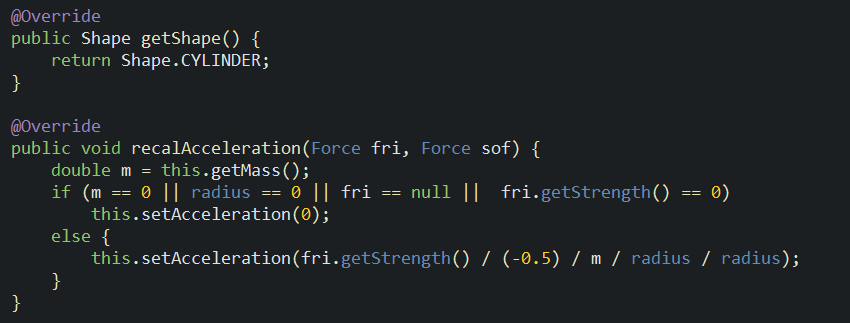
Abstract class: MainObject with abstract method getShape and recalAcceleration



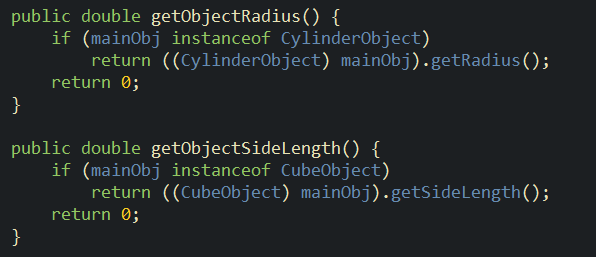
Cube object completely override 2 above method:



Cylinder object completely override 2 above method:

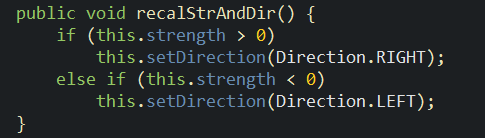


SimulationController using “instanceof” operator:



* Overloading:

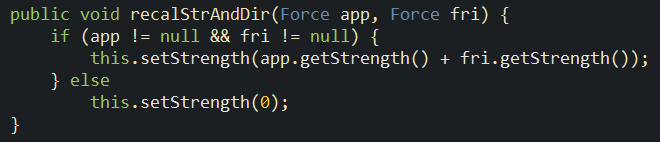
Force:



Friction:



SumOfForce:



+ Upcasting, downcasting:

