

# Rajshahi University of Engineering & Technology



Course No. **CSE 2188**

Course Name: **Software Engineering Sessional**

**Group 01**

Lab Report on:

**Gravity Simulator Application**

## Submitted by-

S.M. Khalid Bin Zahid  
(1908001)

Najeeb Ahmed Bhuiyan  
(1908002)

Puja Mazumdar  
(1908003)

Hameem Zulfiqar  
(1908004)

Tamzid-Ul-Islam  
(1908005)

## Submitted to-

Md. Faisal Rahman Badal  
Lecturer

Dept. of Mechatronics Engineering  
Rajshahi University of Engineering &  
Technology (RUET)

Md. Mehedi Hasan  
Lecturer  
Dept. of Mechatronics Engineering  
Rajshahi University of Engineering &  
Technology (RUET)

## **Introduction:**

Gravity Simulator Application is a project developed under the course CSE 2188, Software Engineering Sessional. This app is developed by a group of students from the department of Mechatronics Engineering, Rajshahi University of Engineering & Technology (RUET). The main inspiration of the Application was from a Reddit post. This project is from Group-01 and the members are S. M. Khalid Bin Zahid (1908001), Najeeb Ahmed Bhuiyan (1908002), Puja Mazumdar (1908003), Hameem Julfiqar (1908004) & Tamzid-UI-Islam (1908005).

## **Gravity Simulator:**

A gravity simulator application is an application which basically simulates the effect of gravity on falling objects at certain heights, much like our real world. It uses Newton's laws of gravity in order to make the simulation. Not only the simulator can simulate gravitational force like our earth, but also it can simulate the gravitational force same as that in other celestial bodies also.

## **Features of Gravity Simulator:**

The following are key features of the Application are:

- ❖ Keep track of the position of the falling object at certain intervals of time.
- ❖ Shows the path of the falling object.
- ❖ Users can set the parameters such as, height, initial velocity, acceleration and time interval as per their wish.
- ❖ It can simulate the falling of multiple objects at the same time.
- ❖ This application can also generate the displacement vs time graph of the falling objects after the simulation.

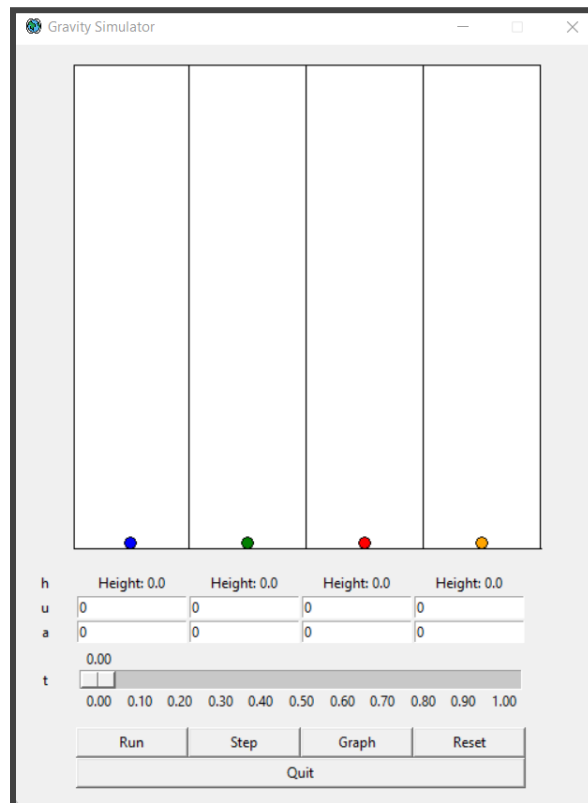
## **Tools in Building the Application:**

We have used:

- ❖ **Python:** Python is an interpreted high-level general-purpose programming language. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small-and-large-scale projects.
- ❖ **Tkinter:** The tkinter package ("Tk interface") is the standard Python interface to the Tcl/Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, including macOS, as well as on Windows systems.
- ❖ **Matplotlib:** Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an

object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

### Graphical User Interface (GUI) of the Application:



### Parameters & Buttons:

- ❖  $h$  denotes the *height* and it sets when the user drags the ball using the mouse in the canvas.
- ❖  $u$  denotes the *initial velocity* and for each ball the user has to set the value manually. Note: Velocity in the upwards direction is (+) and downwards direction is (-).
- ❖  $a$  denotes the *acceleration* and for each ball the user has to set the value manually. Note: Acceleration in the upwards direction is (+) and downwards direction is (-).
- ❖  $t$  denotes the *time interval* and using the slider the user can give the required input from 0 to 1 (But do not input 0 otherwise the application might crash).

- ❖ **`Step`** Button makes step by step simulation.
- ❖ **`Run`** Button makes the whole simulation at once.
- ❖ **`Graph`** Button generates a **\*\*Displacement Vs. Time\*\*** curve of each of the balls.
- ❖ **`Reset`** Button resets all the parameters to its initial value.
- ❖ **`Quit`** Button makes the user get an exit from the Application.

### Code & Installer File:

Using Tkinter the GUI was developed. It included the following things:

- ❖ Main Window
- ❖ Canvas Frame
- ❖ Function Frame
- ❖ Lines
- ❖ Balls
- ❖ Buttons
- ❖ Scale

And then the following functionalities were attached with GUI using Python Function:

- ❖ **move(event)** function moves the ball on the canvas.
- ❖ **update\_height()** function updates the height of the ball according to the position of the ball.
- ❖ **step\_button\_func(t)** function is the main function of the **step** button. Does the step by step simulation.
- ❖ **run\_button\_func()** function is the main function of the **run** button. Does the full simulation at once.
- ❖ **graph\_button\_func(t)** function is the main function of the **graph** button. Generates the graph of the simulation.
- ❖ **reset\_button\_func(t)** function is the main function of the **reset** button.

Some related information of the code:

**Line of Code (LOC):** 208

**Total imported libraries & packages:** 8

**Mostly used libraries & packages:** 3

Using **NSIS or Nullsoft Scriptable Install System**, the installer file was created through which the executable file of the application could be installed in the computer. NSIS is a script-driven installer authoring tool for Microsoft Windows backed by Nullsoft, the creators of Winamp. NSIS is released under a combination of free software licenses, primarily the zlib license. The size of the installer is 31,388 KB.

### **Errors:**

- ❖ Sometimes when the time slider is set to 0 while doing the simulation.
- ❖ The installer file sometimes cannot properly install the executable file.

### **Usage of Gravity Simulator Application**

There are dynamic usage of the application but it is most commonly used in:

- ❖ Astrophysics.
- ❖ Understanding the Motion Equations.
- ❖ Free Falling Body Experiment Simulation & Calculation.
- ❖ Making Physics Interesting.

etc.

### **Discussion:**

Gravity Simulator Application was a team work for all the group members. Although we completed the project successfully, we had faced some difficulties while working on this project, for example, the graph was not properly generated and the aspect ratios of the GUI were not on point. Also we had common coding bugs while doing the project. But in the end, with minimum errors we completed the project and it works perfectly.

### **Conclusion:**

As the Gravity Simulator was developed by the students of Mechatronics Engineering, it was kept in mind whether proper synergistic integration between different fields were made. In our 2nd Year, Odd Semester, we have studied *Kinematics of Particles* under the course *ME 2155*. And the laws and equations studied there are implemented in this Application as we use the Motion equations to make the simulations. Firstly while doing the work, all the group members did not know about Python and the related libraries. All of the members learned the basics and completed the Gravity Simulator Application.