

UNIVERSIDAD DE MONTEMORELOS



FACULTAD DE INGENIERÍA Y TECNOLOGÍA

Taller de Literatura Científica 2019

Desarrollo de software como apoyo para la diferenciación del  
tiro parabólico con y sin rozamiento del aire

Por: Adriana L. García Garza y Débora J. Cruz Valencia

Asesor: Dr. Pablo Lemus Alonso

Montemorelos, Nuevo León, México

Abril de 2019

## RESUMEN

*El resumen es un extracto del documento que contiene los puntos más importantes del trabajo. El resumen no se divide en secciones ni contiene referencias, tablas, diagramas ó figuras. Debe incluir: objetivo general, delimitaciones, procedimiento empleado, resultados más importantes y conclusiones relevantes.*

*El resumen debe contener entre 200 y 500 palabras, ocupando una sola página.*

*Se deben incluir de 3 a 5 palabras clave (Keywords) al final del resumen.*

Las tecnologías de la información están impactando exponencialmente en cualquier ámbito imaginado, y es necesario que se pueda aprovechar este impacto para promover la enseñanza por medio de ellas. Siendo que son parte esencial en la vida de casi todo estudiante y maestro actual, se propuso un software que pudiera apoyar a estas dos poblaciones a entender y darse a entender mejor. En este documento se presenta el desarrollo de un software con una interfaz gráfica muy sencilla de entender, que permite que el usuario pueda percibir fácilmente las diferencias entre un sistema de tiro parabólico con rozamiento del aire y sin rozamiento del aire. El objetivo de este software es presentar, tanto a alumnos como maestros, un apoyo sencillo y fácil de entender para la resolución de problemas de tiro parabólico. Además de estar diseñado para que el usuario pueda obtener distintos valores y note diferencias entre sistema y sistema, este software está creado con un lenguaje muy sencillo de usar y desarrollado en un entorno especializado en la enseñanza del lenguaje Java. Los resultados de este proyecto, fueron satisfactorios y los objetivos fueron cumplidos de la manera esperada. El software aquí presentado, será de gran interés para quien esté interesado debido a su sencillez y facilidad de uso.

**Palabras clave:** Software, tiro parabólico, rozamiento del aire, Tecnologías de la información (TIC).

# ÍNDICE DE CONTENIDO

*El índice debe ser lo más completo posible e incluir todos los encabezados y sub encabezados del documento. Los encabezados deben copiarse fielmente del texto. Lo mejor es indicar los niveles de los encabezados por medio de numeración y sangrías.*

## Índice

<b>Portada del documento de investigación</b> .....	1
<b>Resumen</b> .....	2
<b>Índice</b> .....	3
<b>1. Introducción</b> .....	4
1.1 Antecedentes .....	4
1.2 Definición del problema .....	4
1.3 Justificación .....	5
1.4 Objetivos .....	5
1.5 Hipótesis .....	6
<b>2. Fundamentos teóricos</b> .....	6
2.1 Marco teórico .....	6
2.2 Estado del arte .....	7
<b>3. Resultados</b> .....	9
3.1 Descripción del análisis y diseño del software .....	9
3.2 La presentación de resultados.....	16
3.3 La discusión o análisis de los resultados obtenidos.....	17
<b>Conclusiones</b> .....	18
<b>Bibliografía</b> .....	19

# 1. SECCIÓN 1 INTRODUCCIÓN

*En esta sección se deben incluir los siguientes elementos que debieron ser presentados en el anteproyecto de investigación: Antecedentes, Definición del problema, Justificación, Objetivo general, Objetivos específicos, Hipótesis.*

## 1.1 Antecedentes

*Se le comunica al lector los motivos que despertaron su interés en la investigación. El objeto de esta parte es motivar al lector. Se debe aclarar el porqué del trabajo, cómo se interesó en él, y qué lo estimuló para que lo llevara a cabo.*

La globalización y las nuevas tecnologías de la información han ayudado en alto grado a la evolución de las sociedades, generando una gran gama de proyectos internacionales a favor del desarrollo de las propias tecnologías, de la educación y de la humanidad en general [1].

En el ámbito educativo, específicamente, han tenido un gran impacto, puesto que estas han llegado a transformar la forma y método de enseñanza-aprendizaje [2]. Aunado a esto, la gran necesidad de los estudiantes de la actualidad por conseguir información útil para sus clases o material de apoyo, son puntos clave en el desarrollo de softwares que faciliten su entendimiento y mejoramiento.

## 1.2 Definición del problema

*Es una pregunta que inicia el proceso de investigación de un proyecto, éste debe de ser claro y viable en términos concretos y explícitos, de manera que sea susceptible de investigarse con procedimientos científicos, se establece de una manera sencilla para concluir con ella los antecedentes puestos anteriormente.*

La física y las matemáticas son parte fundamental de la vida, y sin la comprensión de ellas, no existirían las tecnologías ni avances en estas mismas. Su gran sofisticación permite la aplicación de la tecnología a la vida cotidiana. Tanto alumnos como maestros, pueden apoyarse de las tecnologías para desarrollar un mejor entendimiento y promover el autoaprendizaje. El docente apoyado de las TIC produce recursos colaborativos dentro

del aula, para que el alumno pueda establecer su propio conocimiento apoyado de un aprendizaje reflexivo [3]. Si se implementaran las tecnologías en la enseñanza de tiro parabólico o movimiento de proyectil, ¿podrían facilitar a los estudiantes y maestros las evaluaciones metódicas del desarrollo de una ecuación de tiro parabólico con rozamiento y sin rozamiento del aire?

### *1.3 Justificación*

*Es la parte del proyecto en la que se da a conocer la importancia del estudio de la investigación. Esta sección incluye dos puntos: por qué vale la pena realizar este estudio y quiénes se beneficiarán con los resultados.*

Implementando un software de tiro parabólico, los maestros y estudiantes afines a este tema tendrán la capacidad de evaluar sus procedimientos y resultados, además de promover el uso de las tecnologías de una manera correcta. Alumnos y maestros se verán beneficiados por el software, y encontrarán un apoyo directo de una herramienta muy fácil de usar, facilitando el aprendizaje y diferenciación entre dos sistemas de tiro parabólico.

### *1.4 Objetivos*

*Objetivo general: Enuncia de manera explícita lo que se desea conocer, lo que se desea buscar y lo que se pretende tener como resultado en el proyecto. Debe llegarse a él al finalizar la investigación.*

Realizar un software que ejecute los cálculos necesarios para la obtención de distintas componentes del sistema de un tiro parabólico con rozamiento y sin rozamiento del aire.

*Objetivos específicos: Se enumeran las metas específicas que se pretende alcanzar, se desprenden del objetivo general e identifican las acciones que se van a realizar para lograrlo (cómo se va a resolver el problema planteado o cómo podría ayudar a resolverlo).*

-Conocer y entender las fórmulas y los pasos a seguir para la obtención de resultados.

- Obtener las gráficas de un sistema de tiro parabólico con rozamiento y sin rozamiento del aire.
- Observar con facilidad las diferencias entre los dos sistemas de tiro parabólico.
- Conocer mejor las librerías de Java.
- Uso de las tecnologías de la información para crear un sistema de enseñanza fácil y sencillo de tiro parabólico.

### 1.5 Hipótesis

*Son aquellos supuestos acerca del tema de investigación y son la inquietud que llevan a analizarla. Las hipótesis ayudan a seguir un margen en la investigación, éstas deben aclararse al final de la investigación, ya sea que se rechacen o se acepten.*

Desarrollar un software que simule un tiro parabólico con y sin rozamiento del aire, y facilite la comprensión y diferenciación entre los dos sistemas.

## 2. SECCIÓN 2 FUNDAMENTOS TEÓRICOS

### 2.1 Marco teórico.

*Consiste en desarrollar el conjunto de conceptos, definiciones, y herramientas que se han seleccionado para relacionarlos con los datos del problema a investigar. El propósito principal de esta sección es identificar el conocimiento existente acerca del tema del trabajo que se va a desarrollar. Este trabajo deberá estar insertado en el panorama completo de dicho conocimiento.*

#### 2.1.1 Java

El lenguaje de programación Java es un lenguaje orientado a objetos basado en clase, de tipo general, concurrente y tipificado. Normalmente se compila con el conjunto de instrucciones de bytecode y el formato binario definido en la especificación de la máquina virtual de Java [4]. Es considerado el lenguaje ideal para aprender informática, debido a que incorpora muchos aspectos que en cualquier otro lenguaje solo se encuentran como

extensiones y no como parte del propio lenguaje. Además de implementarlos, la incorporación de estos conceptos es de un modo estándar, sencillo y claro que en cualquier otro lenguaje [5]. Este lenguaje se escogió debido a todas las ventajas antes mencionadas, contando también con lo sencillo que es aprenderlo y la facilidad de librerías que ofrece para desarrollar el proyecto.

### 2.1.2 BlueJ

BlueJ es un entorno de desarrollo Java diseñado específicamente para la enseñanza a un nivel introductorio. Fue diseñado e implementado por el equipo de BlueJ en la Universidad Deakin, Melbourne, Australia, y la Universidad de Kent en Canterbury, Reino Unido [6].

Es un entorno de desarrollo que permite desarrollar programas Java de forma rápida y sencilla. Sus principales características son que es: Simple, diseñado para enseñar, interactivo, portable, maduro e innovador [7]. Este entorno se escogió debido a su factibilidad y compatibilidad con la idea de desarrollo del proyecto, además de las facilidades que proporciona como entorno preparado para la enseñanza y aprendizaje.

### 2.2 Estado del arte

*Constituye la sección documental de la investigación que brinda la base conceptual a la misma. Es una presentación organizada de conceptos, antecedentes históricos, resultados de investigaciones relacionadas y metodologías. Su característica fundamental es que su contenido se deriva de fuentes externas y es la base que da validez conceptual a las acciones emprendidas. Por tanto, no debe incluir opiniones ni comentarios propios de los investigadores y debe cubrir todos los aspectos relacionados con el proceso de investigación que se llevó a cabo. Una vez analizados los fundamentos, es importante se conceptualicen con claridad la relación existente entre este trabajo y la información de otros autores. En este capítulo debe mencionar: Qué es lo que ya se conoce sobre este tema, cuáles son las investigaciones más recientes sobre el tema, en qué teoría se va a fundamentar, cómo se relaciona el trabajo actual con lo anteriormente desarrollado, entre otros aspectos.*

### 2.2.1 Las TIC en la enseñanza-aprendizaje de la física y las matemáticas

En la actualidad las Tecnologías de Información y Comunicación, desempeñan un papel importante, día a día nos marcan un contexto en el ámbito cultural, social, deportivo, de entretenimiento y por supuesto, informativo. Las TIC, se plantean como una necesidad que el mundo globalizado exige, transformando el pensamiento mecánico a la virtualidad, con todos los ámbitos que pueden llegar a confluir, cuando se desarrolla en la mente del estudiante y del maestro un aprendizaje con altas dosis de tecnología, cerrando poco a poco esa brecha que puede llegar a darse cuando los estudiantes no emplean adecuadamente a favor de su educación, los medios tecnológicos [8].

Como parte esencial del mundo actual, las tecnologías son una excelente herramienta, no solo para el trabajo o el comercio, sino para promover nuevas formas de estudio y conocimiento. Con ayuda de las tecnologías, se pueden ofrecer, tanto a alumnos como maestros, herramientas que les sirvan para mejorar la comprensión del estudiante en ciertos temas, como el tiro parabólico. Y a su vez, también sirven para que el maestro pueda encontrar un apoyo al hacer revisiones de trabajo o para mostrar un ejercicio más sencillo y más atractivo por el uso de tecnología que supone.

En la investigación [8], se nos presenta un software que facilita y apoya al docente en el aula al impartir sus clases, y a su vez, ayuda a mejorar el aprendizaje de los estudiantes. Sus objetivos son muy claros, y sus conclusiones y resultados son factibles y apoyan la creencia de que un software sí puede mejorar la enseñanza-aprendizaje de quienes lo usan. A pesar de ser con un enfoque, más que nada matemático, su relación con el impacto que tiene y lo que es, va de la mano con la investigación presente.

### 2.2.2 El tiro parabólico y su simulación por medio de un sistema

El sistema de simulación de tiro parabólico de la investigación [9], demuestra en gráficas simultáneas, dos trayectorias de distintos tipos de tiro parabólico, describiendo trayectorias algo distintas. Esta investigación, presenta detalladamente las gráficas, haciendo muy fácil su diferenciación. El propósito de la investigación, al igual que el de esta investigación, es la de dar a conocer una herramienta elaborada con ayuda de las



TIC como apoyo para la labor docente o como apoyo gráfico para los alumnos. Este sistema permite que el usuario pueda seleccionar e ingresar los valores de las condiciones iniciales del proyectil, y con ayuda de las fórmulas, describirá en forma de gráfica la trayectoria de los tiros parabólicos. Una de las ventajas que este programa ofrece, es la capacidad que el usuario tendrá de modificar los valores cuantas veces desee para facilitar mejor la comprensión del tema. De acuerdo a los parámetros que se le introduzcan, será la manera en que la gráfica se irá modificando, haciendo presente y marcada la diferencia entre un tiro parabólico con ciertos parámetros y otro con distintos parámetros.

Para obtener los resultados numéricos, utilizó las ecuaciones correspondientes al movimiento rectilíneo uniforme con sus respectivos componentes, como la posición horizontal del proyectil y la velocidad del cuerpo en la dirección horizontal. La ecuación de la dirección vertical fue obtenida del movimiento rectilíneo uniformemente variado. En este caso, también fue necesario sacar sus distintos componentes y despejar las fórmulas para obtener ciertos parámetros.

Sus conclusiones fueron exitosas, puesto que el sistema tiene usabilidad y facilidad de manejo, y permite al usuario saber con precisión el alcance y las diferencias entre usar ciertos parámetros y entrar otros. Y como punto importante, concluyeron que el sistema fue un medio efectivo que permitió la interacción del usuario con la computadora de una manera eficaz y sencilla.

### **3. SECCIÓN 3 RESULTADOS**

*Esta sección comprende tres partes fundamentales:*

*3.1 La descripción del análisis teórico y práctico del trabajo (Metodología según se aplique al área de la investigación).*

#### **3.1.1 Ecuaciones del tiro parabólico sin rozamiento del aire**

Para realizar este proyecto, se utilizaron las ecuaciones de movimiento parabólico propuestas en [10]. Como se mencionó en una sección anterior, se buscaron las

componentes del sistema de tiro parabólico: el tiempo de vuelo, la altura máxima y el alcance máximo.

El tiempo de vuelo es el tiempo total que el proyectil permanece en movimiento. Para obtener este tiempo, se tomó en cuenta que la componente  $y = 0$ , cuando el cuerpo llega al suelo. Se iguala la ecuación y queda de la siguiente manera:

$$0 = v_{0y} - \frac{1}{2} g t \quad (1)$$

Se despeja  $t$  de la ecuación anterior, y queda:

$$t = \frac{2v_{0y}}{g} \quad (2)$$

Tomando en cuenta la ecuación del tiempo en relación con el ángulo, queda:

$$t = \frac{2v_0 \operatorname{sen} \alpha}{g} \quad (3)$$

Para el alcance máximo, se sustituye el valor del tiempo obtenido en la ecuación (3), en la ecuación de la distancia horizontal que recorre el proyectil como se ejemplifica a continuación:

$$x = v_{0x} t \quad (4)$$

$$x = v_0 \cos \alpha \frac{2v_0 \operatorname{sen} \alpha}{g} \quad (5)$$

$$x = \frac{2v_0^2 \operatorname{sen} \alpha \cos \alpha}{g} \quad (6)$$

Utilizando la relación trigonométrica  $2\alpha = 2\operatorname{sen} \alpha \cos \alpha$ , se obtiene:

$$x = \frac{v_0^2}{g} \operatorname{sen} 2\alpha \quad (7)$$

Y por último, la altura máxima, que se alcanza cuando  $v_y = 0$ , es decir:  $v_{0y} - gt = 0$ .  
Despejando el tiempo de esta ecuación, obtenemos que:

$$t = \frac{v_{0y}}{g} = \frac{v_0 \sin \alpha}{g} \quad (8)$$

Sustituyendo la ecuación (8) en la ecuación de coordenada  $y$ , se obtiene:

$$y_{\max} = v_{0y}t - \frac{1}{2}gt^2 = \frac{v_0^2 \sin^2 \alpha}{g} = \frac{v_0^2 \sin^2 \alpha}{2g} \quad (9)$$

### 3.1.2 Ecuaciones del tiro parabólico con rozamiento del aire

Para realizar el sistema con rozamiento del aire, se utilizaron las mismas componentes que en la sección anterior, sin embargo, las fórmulas fueron distintas debido a la participación del aire en estas fórmulas. Las fórmulas utilizadas, fueron propuestas por [11], y fueron implementadas en este proyecto de acuerdo a las necesidades y variantes que se deseaban encontrar. Para el tiempo se obtuvo la fórmula:

$$t = \frac{1}{b} \ln \left( 1 + \frac{bv_{0y}}{g} \right) \quad (10)$$

Para la altura máxima se despejó de la fórmula la derivada del tiempo de  $v_y = \frac{dy}{dt} = 0$  para obtener el tiempo, y se obtuvo la siguiente fórmula:

$$y = \frac{v_{0y}}{b} - \frac{g}{b^2} \ln \left( 1 + \frac{bv_{0y}}{g} \right) \quad (11)$$

Para obtener el alcance máximo se hizo una relación entre la fórmula de

$$\frac{2b}{3v_0 \cos \theta} R^2 + R - R_0 = 0 \quad (12) \text{ y la fórmula general } x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (13).$$

De la Ec. (12), se despeja  $R_0$  quedando:

$$R_0 = \frac{2v_{0x}v_{0y}}{g} \quad (14)$$

Para obtener la posición en x, se sustituyó la Ec.(13) y se le aplicó la propiedad de  $ax^2 + bx + c = 0$ , quedando como valores:

$$a = \frac{2b}{3v_0 \cos \theta}, b = 1, c = R_0 \quad (15)$$

Sustituyendo Ec. (15) en la Ec. (13), queda como:

$$x = \frac{-1 \pm \sqrt{1^2 - 4 \left( \frac{2b}{3v_0 \cos \theta} \right) \left( \frac{2v_{0x}v_{0y}}{g} \right)}}{2 \left( \frac{2b}{3v_0 \cos \theta} \right)} \quad (16)$$

### 3.1.3 Implementación de las fórmulas en el software

Se utilizó el entorno BlueJ para Java. Primeramente, se importaron las librerías señaladas en la Fig.1 debido a que, dentro de ellas, se encuentra la clase Math, que es la parte fundamental del programa. La información sobre las librerías fueron encontradas en [12], donde se presenta una clara explicación de sus funciones y algunos ejemplos de código en el que se puede implementar.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
```

**Fig 1.** Librerías utilizadas para realizar el software matemático.

Seguidamente, se declararon las variables a utilizar y se llamó a un ActionListener para desarrollar la interfaz de nuestro software, tal y como se muestra en la Fig.2

```
public class DProyectoIntegrador extends JFrame implements ActionListener
{
    private JTextField txtVelInicial, txtGrados, txtResultado1,txtResultado2,txtRoz;
    private JLabel lblVelInicial, lblGrados, lblResultado1, lblResultado2, lblDatos, lblQue, lblRoz, lblGraph;
    private JButton btnCalcular;
    private JPanel PanGrafica;
    private JRadioButton TiempoAire, Dist, Altura;
    private ButtonGroup operaciones;
```

**Fig.2** Se declararon todas las variables privadas para que solo el programador tuviera acceso a ellas. Se implementó el ActionListener para escuchar los eventos creados por los botones que se mostrarán después.

Con las líneas de código mostradas en la Fig. 3, se crea el área donde se tendrán todos los objetos y componentes del software calculador.

```
15 public static void main(String [] args)
16 {
17     DProyectoIntegrador prog = new DProyectoIntegrador();
18     prog.setSize(1100, 300);
19     prog.setVisible(true);
20 }
```

**Fig.3** La línea de código `prog.setSize(1100, 300);` prepara el campo donde se pondrán los objetos creados.

Se crean los campos de llenado de información con el comando `JTextField`, se implementaron en el código para que el usuario pudiera meter la información que necesitara sin algún impedimento. Tanto para velocidad inicial, ángulo y cantidad de rozamiento, se usó el comando `JLabel` para que aparecieran los nombres junto a sus respectivos `JTextField`, tal y como se muestra en la Fig.4.

```
lblVelInicial = new JLabel("Velocidad inicial");
contenedor.add(lblVelInicial);
lblVelInicial.setBounds(30, 70, 100, 30);

txtVelInicial = new JTextField();
contenedor.add(txtVelInicial);
txtVelInicial.setBounds(140, 70, 100, 30);

lblGrados = new JLabel("Ángulo");
contenedor.add(lblGrados);
lblGrados.setBounds(30, 130, 50, 30);

txtGrados = new JTextField();
contenedor.add(txtGrados);
txtGrados.setBounds(140, 130, 100, 30);

lblRoz = new JLabel("Rozamiento (b<.1)");
contenedor.add(lblRoz);
lblRoz.setBounds(30, 190, 110, 30);

txtRoz = new JTextField();
contenedor.add(txtRoz);
txtRoz.setBounds(140, 190, 100, 30);
```

**Fig.4** Se muestran los comandos respectivos junto a la asignación de nombres a cada Label.

Seguidamente, se crean los botones con un `JRadioButton` para que el usuario escoja lo que desea conocer. Estos botones solo tienen dos variables, o están seleccionados o no. Cuando el botón esté seleccionado, estará puesto en la línea de código como `True`, pero cuando esté en `False`, significa que no está seleccionado.

```

TiempoAire = new JRadioButton("Tiempo en el aire",true);
Dist = new JRadioButton("Distancia en x",false);
Altura = new JRadioButton("Altura en y",false);

```

**Fig.5** La variable TiempoAire está seleccionada, se conoce esto debido al true que se encuentra al final de la primera línea de código.

Después de crear esos botones, se procedió a crear los campos donde se presentarían los resultados de lo especificado por el usuario. Teniendo dos variantes, se crean dos tipos de campos, uno que presente los resultados con rozamiento y otro sin rozamiento. Se les especificó una coordenada para evitar que estos anduviera en cualquier parte de la pantalla que creamos en la Fig. 3. Tal y como se muestra en la Fig.6, se creó un botón que reciba el ActionListener para que suceda un evento posteriormente, en este caso, el evento será hacer los cálculos y plasmarlos en los campos en blanco creados con el JTextField.

```

btnCalcular = new JButton("Calcular");
contenedor.add(btnCalcular);
btnCalcular.setBounds(580,40,100,30);
btnCalcular.addActionListener(this);

lblResultado1 = new JLabel("Sin rozamiento");
contenedor.add(lblResultado1);
lblResultado1.setBounds(520,100,150,30);

txtResultado1 = new JTextField();
contenedor.add(txtResultado1);
txtResultado1.setBounds(640,100,100,30);

lblResultado2 = new JLabel("Con rozamiento");
contenedor.add(lblResultado2);
lblResultado2.setBounds(520,170,150,30);

txtResultado2 = new JTextField();
contenedor.add(txtResultado2);
txtResultado2.setBounds(640,170,100,30);

```

**Fig.6** Creación del botón y los campos de resultado, junto a sus respectivos Labels.

Después de crear los Listeners, se procedió a crear los eventos que el Listener escuchará. Primeramente, en el constructor se declararon nuevas variables y se inicializaron variables antiguas en cero para evitar que la máquina virtual se saturara. Se guardan los valores capturados dentro de los campos a rellenar por el usuario y se implementaron en el nuevo constructor declarándoselos a nuevas variables.

```

public void actionPerformed(ActionEvent event)
{
    double vel1, ang, resultado1, resultado2, roz, angle;
    resultado1=0;
    resultado2=0;
    vel1=Double.parseDouble(txtVelInicial.getText());
    angle=Double.parseDouble(txtGrados.getText());
    roz=Double.parseDouble(txtRoz.getText());
    ang=angle*Math.PI/180.0;
}

```

**Fig.7** Variables con sus respectivos valores declarados.

Por último, se implementaron las fórmulas vistas en la sección pasada, donde se puso en uso la clase Math que venía incluida en la librería que se importó en al principio del código. Para el Tiempo de vuelo se utilizaron los comandos escritos en la Fig. 8. Se muestran dos resultados debido a que son dos distintos tiros que se pueden comparar. El primer resultado corresponde al tiro parabólico sin rozamiento del aire y el segundo a uno con rozamiento del aire.

```

if(TiempoAire.isSelected())
{
    resultado1=((2*vel1)*Math.sin(ang))/9.8;
    resultado2=(1/roz)*(Math.log(1+(roz*vel1*Math.sin(ang)/9.8)));
}

```

**Fig. 8** Se declaran las ecuaciones para obtener los dos posibles resultados si el tiempo en el aire es el botón seleccionado.

Así como se muestra en la Fig. 8, se hizo el mismo procedimiento para la distancia y la altura máxima, y por último, se declararon las líneas de código que imprimirían todos los resultados obtenidos, como se muestra en la Fig. 9

```

txtResultado1.setText(String.valueOf(resultado1));
txtResultado2.setText(String.valueOf(resultado2));
}

```

**Fig.9** Líneas de código que proyectan el resultado en los campos.

### 3.2 La presentación de los resultados.

El software resultante fue el mostrado en la Fig. 10. En el primer apartado de esta, se muestran tres campos vacíos, estos son los campos que el usuario llenará con información que ya tiene.

The screenshot shows a web-based form for calculating projectile motion. It is divided into three main sections. The first section, titled 'Ingrese los datos', contains three input fields: 'Velocidad inicial', 'ángulo', and 'Rozamiento (b<.1)'. The second section, titled '¿Qué quiere encontrar?', contains three radio button options: 'Tiempo en el aire' (which is selected), 'Distancia en x', and 'Altura en y'. The third section contains a 'Calcular' button and two output fields: 'Sin rozamiento' and 'Con rozamiento'.

**Fig. 10** Resultados del ensamblaje de las líneas de código presentadas en las Figs. 1-9.

Una vez hecho eso, podrá escoger entre calcular el tiempo en el aire, la distancia en x o la altura en y. Esta segunda sección, es el resultado de los botones que solo tenían dos opciones, true o false. Por último, se presenta el botón de calcular y dos campos vacíos donde se presentarán los resultados. Se colocaron a la par los campos vacíos para que sea más notoria la diferencia entre un tiro parabólico con rozamiento y otro sin rozamiento, pero ambos con los mismos parámetros.

La primera prueba que se realizó fue donde  $v_0 = 30m/s$ ,  $\theta = 48$  y el rozamiento del aire es de 2, y se decidió sacar el tiempo en el aire que estaría un proyectil. Los resultados fueron los mostrados en la Fig. 11-13. Las pruebas que se hicieron, fueron basadas en los ejercicios propuestos en [15], y se comprobaron con los resultados que presentaban tras realizar los mismos cálculos realizados en esta investigación.



**Tiro Parabolico**

**Ingrese los datos**

Velocidad inicial

Angulo

Rozamiento (b<.1)

**¿Qué quiere encontrar?**

☒ Tiempo en el aire

☐ Distancia en x

☐ Altura en y

**Calcular**

Sin rozamiento

Con rozamiento

**Fig. 11** Se presentan los dos resultados con rozamiento y sin rozamiento para conocer el tiempo en el aire.

**Tiro Parabolico**

**Ingrese los datos**

Velocidad inicial

Angulo

Rozamiento (b<.1)

**¿Qué quiere encontrar?**

☐ Tiempo en el aire

☒ Distancia en x

☐ Altura en y

**Calcular**

Sin rozamiento

Con rozamiento

**Fig. 12** Se presentan los dos resultados con rozamiento y sin rozamiento para la distancia en x.

**Tiro Parabolico**

**Ingrese los datos**

Velocidad inicial

Angulo

Rozamiento (b<.1)

**¿Qué quiere encontrar?**

☐ Tiempo en el aire

☐ Distancia en x

☒ Altura en y

**Calcular**

Sin rozamiento

Con rozamiento

**Fig. 13** Se presentan los dos resultados con rozamiento y sin rozamiento para la distancia en y.

### 3.3 La discusión o análisis los resultados obtenidos.

De la prueba de más de tres ángulos distintos y de todas y cada una de las componentes del software, los resultados fueron satisfactorios y muy parecidos a las respuestas proporcionadas en [15]. Se deseaba obtener una manera sencilla y muy marcada de observar las diferencias entre un tiro parabólico que involucre el rozamiento del aire o no, y se consiguió. Gracias a la interfaz sencilla y el acomodo de las casillas de resultados una bajo la otra, percibir los cambios entre una y otra es muy sencillo, y arroja firmemente los resultados para su estudio. Variando unos decimales entre el resultado del software

con los resultados propuestos en [15], se probó la funcionalidad del programa y la capacidad y sencillez que tiene para que se observe a simple vista que el rozamiento del aire provoca severas disminuciones entre el tiempo de vuelo a diferencia de un sistema que no tiene rozamiento.

*Es importante que en la redacción del texto cada tabla y cada figura se encuentren mencionadas y debidamente numeradas en el documento de investigación.*

## **CONCLUSIONES**

*Las conclusiones se desprenden directamente de los resultados del trabajo. Se debe expresar el cumplimiento de las hipótesis, contrastando lo expresado en el marco teórico con los resultados, se debe demostrar que los resultados alcanzados respaldan o se oponen a publicaciones y trabajos de investigación anteriores y en qué se diferencia o amplía lo desarrollado anteriormente.*

El software desarrollado en este proyecto cumplió los objetivos propuestos al principio. Se lograron dar a conocer de una manera sencilla y fácil las diferencias entre un sistema de tiro parabólico con rozamiento y sin rozamiento del aire. El software presenta una interfaz gráfica muy sencilla y fácil de utilizar, y la usabilidad es muy buena. A pesar de no mostrar gráficas, el proyecto cumplió las mismas medidas que las de la investigación [11]. La capacidad de entendimiento y usabilidad, permitirá a los alumnos y maestros a encontrar respuestas rápidas a sus preguntas. Servirá como apoyo a quienes lo deseen usar, y no representará difícil de entender a diferencia de algunos softwares con cálculos demasiado exagerados para lo que el usuario realmente quiere. La elección de Java como el lenguaje padre para desarrollar este proyecto fue excelente. Java cumplió completamente con los parámetros que le dan renombre, y sus creadores no se equivocaban al decir que es un lenguaje fuertemente orientado a objetos.

Y así como el software desarrollado es sencillo, el entorno escogido para desarrollarlo también lo fue. Siendo que BlueJ es un entorno hecho específicamente para que se enseñe en él, fue de gran utilidad que tenga una interfaz gráfica sencilla, porque así permitía que el trabajo fluyera sin distracciones de no encontrar la manera de usarlo.

## REFERENCIAS BIBLIOGRÁFICAS

*Todas las citas textuales o parafraseadas usadas en el texto deben tener una ficha correspondiente en esta sección. Solamente deben aparecer referencias de los documentos citados.*

[1] C. Amador Padilla and O. Tirado Ochoa, "Retos y demandas educativas del siglo XXI", *La Huella del Coyote*, no. 65, p. 12, 2019.

[2] J. Piñuelas Osuna, "Editorial", *La Huella del Coyote*, no. 65, p. 6, 2019.

[3] N. Soto Guevara, "Competencias en TIC para docentes", *La Huella del Coyote*, no. 65, p. 27, 2019.

[4] "Java Programming Language", *Docs.oracle.com*, 2019. [Online]. Available: <https://docs.oracle.com/javase/8/docs/technotes/guides/language/index.html>. [Accessed: 05- May- 2019].

[5] J. García de Jalón et al., *Aprenda Java como si estuviera en primero*, 1st ed. San Sebastián: Universidad de Navarra, 2000, p. 1.

[6] M. Kölling, *The BlueJ Tutorial*, 2nd ed. Denmark: Maersk Institute, 2006, p. 4.

[7] "About BlueJ", *Bluej.org*, 2019. [Online]. Available: <https://www.bluej.org/about.html>. [Accessed: 05- May- 2019].

[10] M. Meneses O. and L. Artunduaga, "Software educativo para la enseñanza y aprendizaje de las matemáticas en el grado 6°" M.S. thesis, Universidad Católica de Manizales, Pitalito, Colombia, 2014.

[11] G. Becerra Córdova, "Un sistema de simulación para el tiro parabólico" M.S. thesis, Universidad Autónoma Chapingo, Chapingo, México, 2016.

[12] "TIRO PARABÓLICO", *Rekursostic.educacion.es*, 2019. [Online]. Available: [http://recursostic.educacion.es/descartes/web/materiales\\_didacticos/comp\\_movimientos/parabolico.htm](http://recursostic.educacion.es/descartes/web/materiales_didacticos/comp_movimientos/parabolico.htm). [Accessed: 04- May- 2019].

[13] "Alcance", *Rekursostic.educacion.es*, 2019. [Online]. Available: [http://recursostic.educacion.es/descartes/web/materiales\\_didacticos/Tiro\\_parabolico\\_con\\_rozamiento/Alcance.htm](http://recursostic.educacion.es/descartes/web/materiales_didacticos/Tiro_parabolico_con_rozamiento/Alcance.htm). [Accessed: 05- May- 2019].

[14] "Java Platform SE 7", *Docs.oracle.com*, 2019. [Online]. Available: <https://docs.oracle.com/javase/7/docs/api/>. [Accessed: 06- May- 2019].

[15] "Movimiento Parabólico - Ejercicios Resueltos - Fisimat", *Fisimat | Blog de Física y Matemáticas*, 2019. [Online]. Available: <https://www.fisimat.com.mx/tiro-parabolico/>. [Accessed: 04- May- 2019].