

**Tema:** Avance del proyecto de Algoritmo II

**Materia:** Algoritmo II

**Catedratico:** Lic. Pedro Noble

**Integrantes:** Edwin Ernesto González Castillo  
Carlos Antonio Soriano García  
Carlos Alberto Molina  
José Antonio Centeno Espinoza

## ***INTRODUCCION***

Flappy bird un juego que se puso muy de moda y en nuestra clase de algoritmo II decidimos mostrar como crear un juego parecido haciendo constar de lo enseñado en las clases ya mencionadas utilizando el entorno gráfico del lenguaje de programación python. Ya que en esta época se han puesto de moda este tipo de juegos como flappy bird o el mismo geometridash.

Pero que es un juego es un fenómeno universal y un derecho de infancia o aun en la adolescencia es por eso que como grupo nos dio curiosidad conocer la estructura de dicho juego por eso el porque de este proyecto el experimentar programarlo en el lenguaje de programación ya mencionado anteriormente y presentarlo como proyecto final de nuestra clase.

Esperamos llegar al resultado final de tener un juego en el que usuario lo pueda disfrutar sin problemas que le agrade y lo apruebe ,esperando adquirido nuevos conocimientos en el lenguaje de programación de python todos los avances de nuestra investigación y mucho mas información en lo que resta del documento.

En el siguiente trabajo se dará a conocer las diversas formas que ocuparemos para llegar al final del proyecto tener terminado el juego por medio de python y el entorno gráfico de pygame

## ***JUSTIFICACION***

Hemos decido realizar el juego llamado flappybird en el lenguaje de programación orientada a objetos Python ya que es un lenguaje multiplataforma y es software libre. Con este nuevo proyecto aplicaremos los conocimientos que ya hemos aprendido y también ahora aplicaremos lo que es la programación con entorno gráfico. (Pygame)

El proyecto en si constara de programar la lineas de código en python haciendo un trabajo investigativo grupal en el que todos aportaremos parte de nuestros conocimientos en el lenguaje de programación python, luego investigaremos las opciones para poder trasladar nuestro código a un entorno grafico para el usuario, con este proyecto esperamos aportar de hacer ver que si se propone , se trabaja en equipo y se investiga acerca de un tema se logran los objetivos planteados mas adelante.

Nuestra linea de investigacion a tomar sera utilizar las herramientas de vanguardia hoy en dia tales como son internet libros en linea basados al lenguaje de programación tambien apoyandonos en nuestro catedratico lic.pedro noble el cual nos ira aconsejando los caminos a tomar con el proyecto.

Obstaculo mas dificil de nuestra investigación sera el del entorno grafico llegar a ese nivel como estudiantes de la carrera de computación sera complicado es por eso que decidimos tomar este reto

## ***Objetivo General***

Crear un juego por medio de python y su entorno demostrando cada proceso de su elaboracion y trabajo en equipo

## ***Objetivos Específicos***

- Aprender cada uno los pasos para poder llegar a comprender el y cada una de las situaciones presentadas en este proyecto
- Obtener una buena calificación en la materia Algoritmos II

## **MARCO TEORICO**

Se trata de un juego de habilidad que apuesta por la sencillez. Eso sí, sólo en cuanto a la mecánica y aspecto visual. Y es que este entretenimiento resulta poco menos que imposible, retando al usuario a superarse en cada partida. Quizá la clave en la que reside su éxito. En este juego se toma el control de un pájaro que ha de superar diferentes barreras en forma de tuberías de forma infinita, tan sólo pasando entre el espacio que dejan estos elementos en pantalla. Un planteamiento realmente sencillo que no lo es tanto a la hora de jugarlo.

basta con pulsar la pantalla para controlar el vuelo del personaje. Un toque le impulsa hacia arriba, mientras que de bajar se encarga él solo. De este modo, con mucho tacto y pericia hay que conseguir hacerlo superar las barreras. Cada tubería dejada atrás es un punto más, siendo el objetivo conseguir el máximo número posible. Sin una meta que de por concluido el juego. Eso sí, tras cada partida, y si se ha conseguido superar un cierto número de tuberías, se logran medallas a modo de

logro para recompensar el esfuerzo y habilidad del usuario.

Un punto fuerte por el que destaca Flappy Bird, además de su sencillez, es el aspecto visual. Y es que los jugadores reconocerán el estilo de los gráficos utilizados. Cuestiones que recuerdan inevitablemente a los juegos de Mario Bros. Elementos con un marcado estilo pixelado o retro, donde el efecto parallax permite dar la sensación de movimiento y profundidad a los diferentes planos que se ven en pantalla mientras el pájaro se desplaza siempre a la derecha en scroll lateral. Todo ello de forma muy colorista y con diferentes escenarios que se alternan aleatoriamente para que las partidas no resulten monótonas.

El jugador controla un pájaro intentando volar entre filas de tuberías verdes sin tocarse con estas. La escena se va desplazando lateralmente.

A medida que se desplazaba va aumentando las dificultades.

El personaje se controlará con el clic derecho del mouse.

## ***Planteamiento del problema***

Se debe de crear una bola que en este caso sera la que suplantara al pajarito, en el juego también debe de contener obstáculos los cuales deben de aparecer de tamaños diferentes y avanzar hacia la bola, la bola deberá de ser controlada con las teclas de subir y bajar de nuestra computadora para poder esquivar los obstáculos, también debe de haber un contador para que cada vez que la bola pase un obstáculo nos sume un punto más, si la bola choca contra uno de los obstáculos el juego nos preguntara si que si queremos continuar o finalizar.

## ***Identificación del problema***

- 1) Crear las líneas de código para crear la bola
- 2) Crear el código que imprima en pantalla los obstáculos, los cuales cada vez que aparezcan deben de tener un tamaño diferente.
- 3) Crear el código para controlar los movimientos de la bola.
- 4) Crear un contador de puntos.
- 5) crear código para que en pantalla aparezca una pregunta la cual nos dará la opción para continuar o finalizar.

## ***Descripción del problema***

- 1) la creación de la bola
- 2) Debemos crear unas barras que aparecerán de forma aleatoria en el juego, las cuales tienen la función de obstaculizar el paso de la bola.
- 3) para esquivar los obstáculos se debe de tener el control de los movimientos hacia arriba y hacia abajo de la bola, los cuales a su vez deben controlarse con dos teclas.
- 4) Cada vez que se esquive un obstáculo se ganara un punto.
- 5) Si la bola choca contra un obstáculo el juego debe finalizar.

## ***Planteamiento de la solución***

Se debe tener un conocimiento fundamental de programación orientada a objetos en la plataforma de python para así resolver los problemas que hemos determinado.

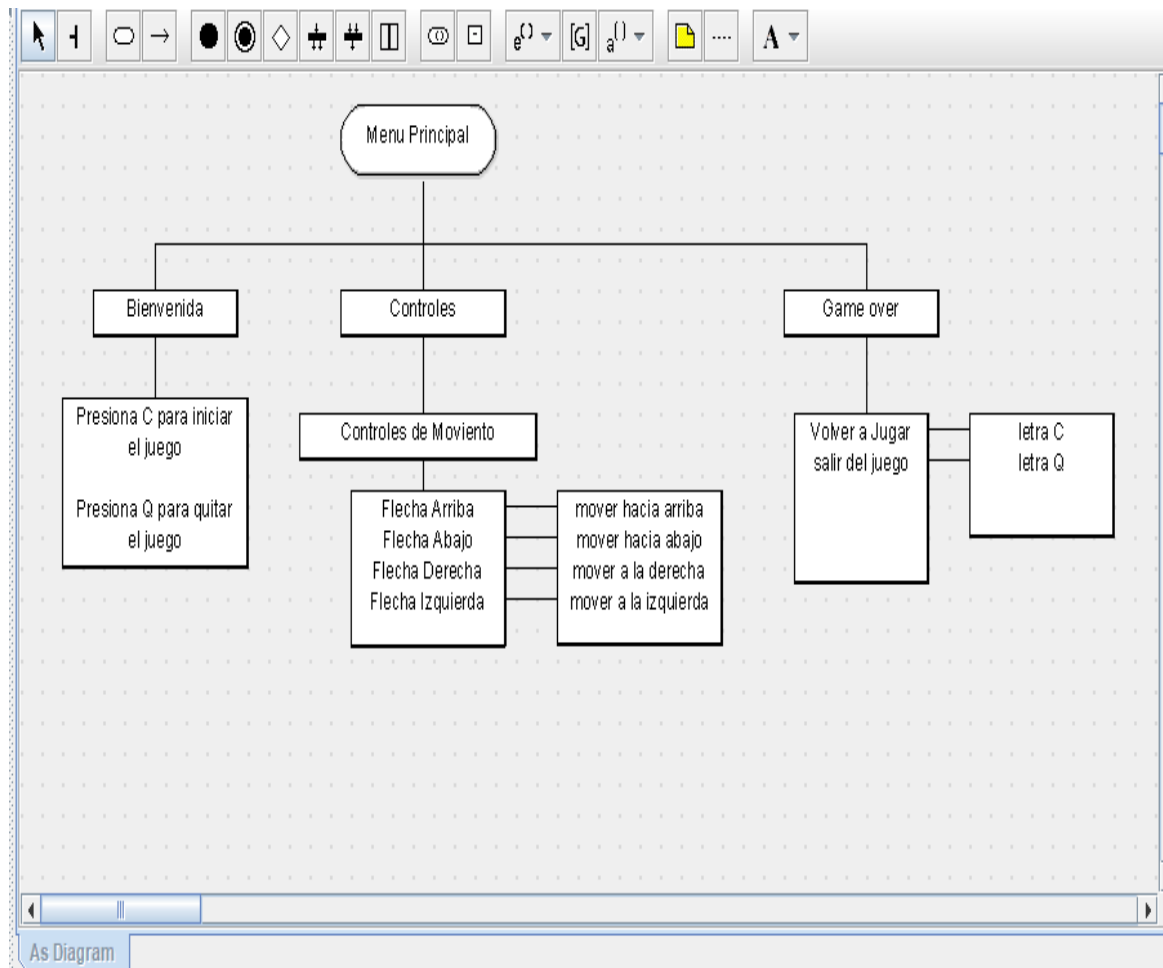
## ***Cómo plantean resolver el problema?***

1. Buscar librerías para juegos en python
2. Buscar una línea de código que nos ayude a dibujar en python y así tener los obstáculos y la bola.
3. Hacer que la bola pueda desplazarse con los cursores de arriba y abajo

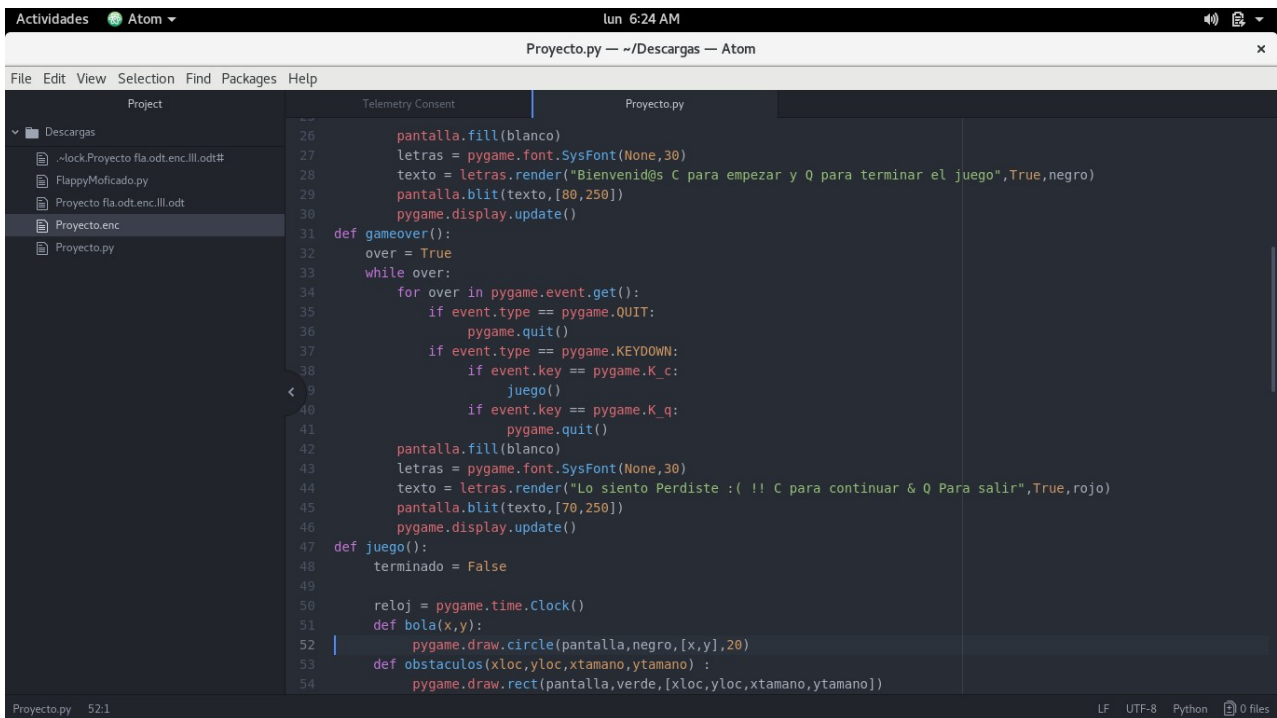
# Marco referencial

**Lenguaje de Programación:** Python

**diagrama de flujo de actividades**



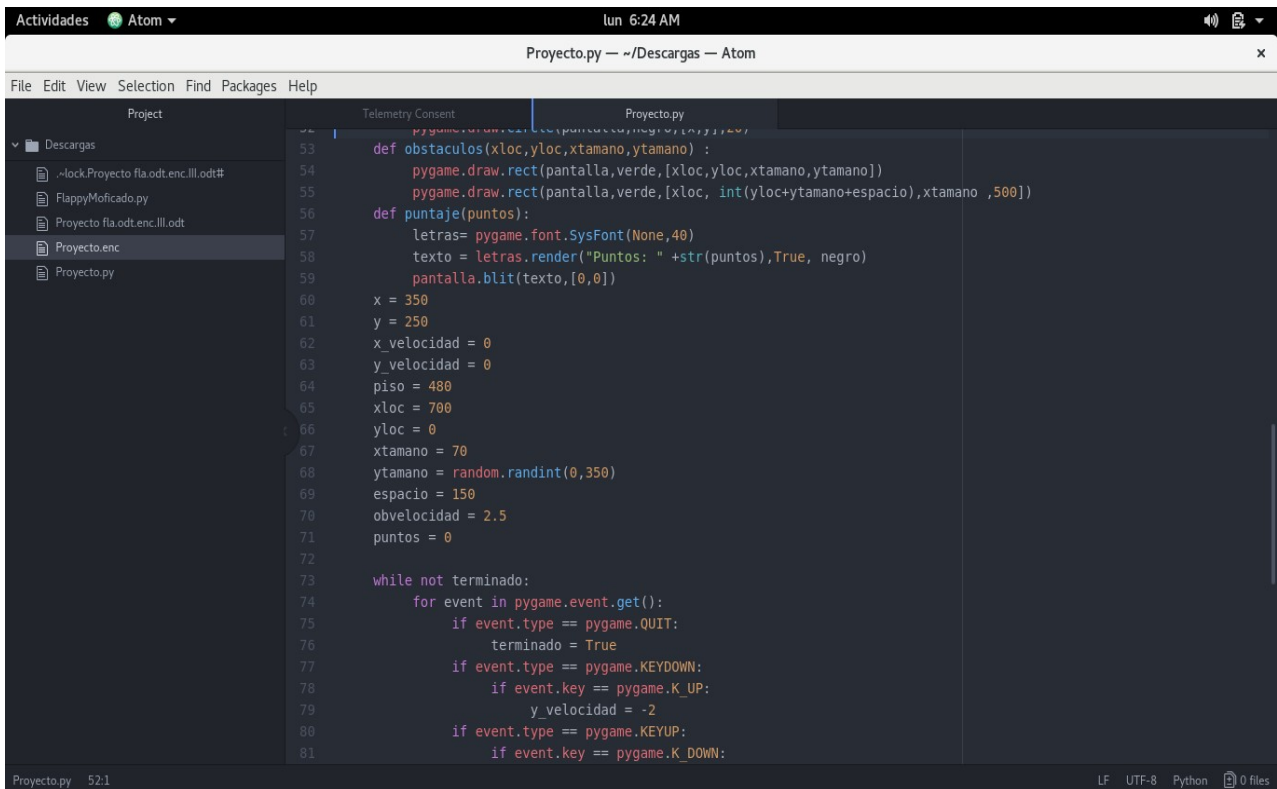
# Implementación de la solución código de la aplicación



```
Project
  Descargas
    .lock.Proyecto fla.odt.enc.Ill.odt#
    FlappyMoficado.py
    Proyecto fla.odt.enc.Ill.odt
    Proyecto.enc
    Proyecto.py

Telemetry Consent
Proyecto.py
26 pantalla.fill( blanco )
27 letras = pygame.font.SysFont( None, 30 )
28 texto = letras.render( "Bienvenid@s C para empezar y Q para terminar el juego", True, negro )
29 pantalla.blit( texto, [ 80, 250 ] )
30 pygame.display.update()
31 def gameover():
32     over = True
33     while over:
34         for over in pygame.event.get():
35             if event.type == pygame.QUIT:
36                 pygame.quit()
37             if event.type == pygame.KEYDOWN:
38                 if event.key == pygame.K_c:
39                     juego()
40                 if event.key == pygame.K_q:
41                     pygame.quit()
42             pantalla.fill( blanco )
43             letras = pygame.font.SysFont( None, 30 )
44             texto = letras.render( "Lo siento Perdiste :( !! C para continuar & Q Para salir", True, rojo )
45             pantalla.blit( texto, [ 70, 250 ] )
46             pygame.display.update()
47 def juego():
48     terminado = False
49
50     reloj = pygame.time.Clock()
51     def bola(x,y):
52         pygame.draw.circle( pantalla, negro, [ x,y ], 20 )
53     def obstaculos(xloc,yloc,xtamano,ytamano) :
54         pygame.draw.rect( pantalla, verde, [ xloc,yloc,xtamano,ytamano ]
```

Proyecto.py 52:1 LF UTF-8 Python 0 files



```
Project
  Descargas
    .lock.Proyecto fla.odt.enc.Ill.odt#
    FlappyMoficado.py
    Proyecto fla.odt.enc.Ill.odt
    Proyecto.enc
    Proyecto.py

Telemetry Consent
Proyecto.py
52 pygame.draw.rect( pantalla, negro, [ x,y,yloc,yloc+ytamano ] )
53 def obstaculos(xloc,yloc,xtamano,ytamano) :
54     pygame.draw.rect( pantalla, verde, [ xloc,yloc,xtamano,ytamano ] )
55     pygame.draw.rect( pantalla, verde, [ xloc, int(yloc+ytamano+espacio),xtamano ,500] )
56 def puntaje(puntos):
57     letras= pygame.font.SysFont( None, 40 )
58     texto = letras.render( "Puntos: " +str(puntos), True, negro )
59     pantalla.blit( texto, [ 0,0 ] )
60 x = 350
61 y = 250
62 x_velocidad = 0
63 y_velocidad = 0
64 piso = 480
65 xloc = 700
66 yloc = 0
67 xtamano = 70
68 ytamano = random.randint( 0,350 )
69 espacio = 150
70 obvelocidad = 2.5
71 puntos = 0
72
73 while not terminado:
74     for event in pygame.event.get():
75         if event.type == pygame.QUIT:
76             terminado = True
77         if event.type == pygame.KEYDOWN:
78             if event.key == pygame.K_UP:
79                 y_velocidad = -2
80             if event.type == pygame.KEYUP:
81                 if event.key == pygame.K_DOWN:
```

Proyecto.py 52:1 LF UTF-8 Python 0 files



Actividades Atom lun 6:22 AM Proyecto.py -- ~/Descargas -- Atom

File Edit View Selection Find Packages Help

```
Project
▼ Descargas
  ~.lock.Proyecto fla.odt.enc.Illl.odt#
  FlappyMoficado.py
  Proyecto fla.odt.enc.Illl.odt
  Proyecto.enc
  Proyecto.py

Telemetry Consent
Project.py
1 import pygame
2 import random
3
4 negro=(0,0,0)
5 blanco=(255,255,255)
6 verde= (0,255,0)
7 pygame.init()
8 rojo = (255,0,0)
9 tamaño= 700,500
10
11 pantalla = pygame.display.set_mode(tamaño)
12 pygame.display.set_caption("FLAPPY BIRD")
13
14 def introjuego():
15     intro = True
16     while intro:
17         for event in pygame.event.get():
18             if event.type == pygame.QUIT:
19                 pygame.quit()
20             if event.type == pygame.KEYDOWN:
21                 if event.key == pygame.K_c:
22                     juego()
23                 if event.key == pygame.K_q:
24                     pygame.quit()
25
26     pantalla.fill(blanco)
27     letras = pygame.font.SysFont(None,30)
28     texto = letras.render("Bienvenid@s C para empezar y Q para terminar el juego",True,negro)
29     pantalla.blit(texto,[80,250])
30     pygame.display.update()
```

Projecto.py 1:1 LF UTF-8 Python 0 files

Actividades Atom lun 6:24 AM Proyecto.py -- ~/Descargas -- Atom

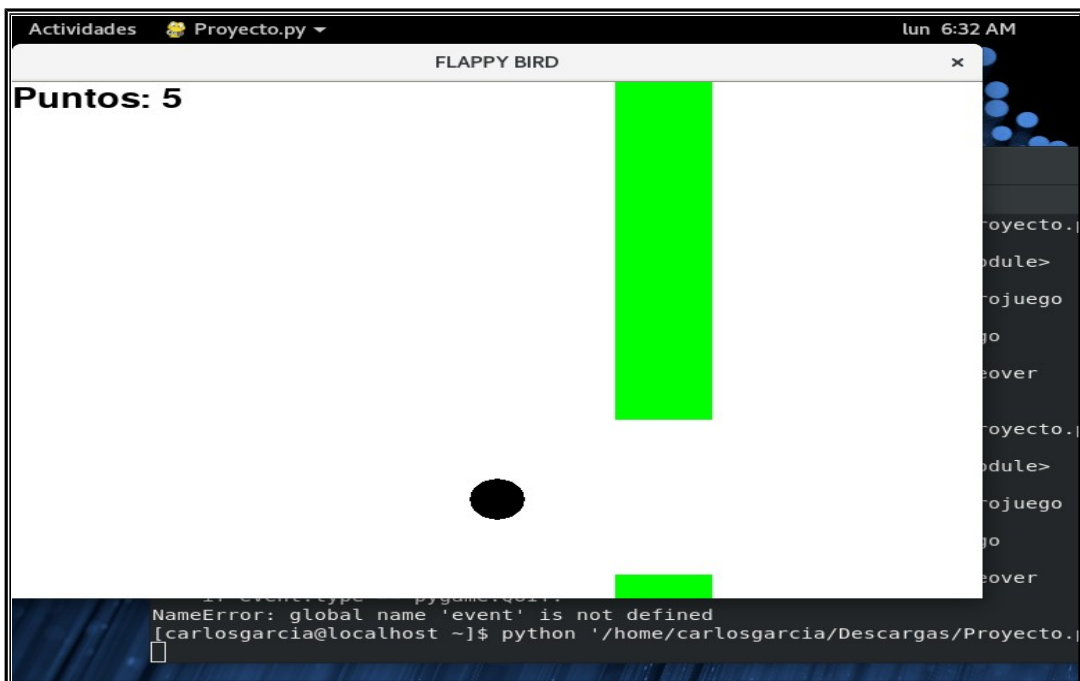
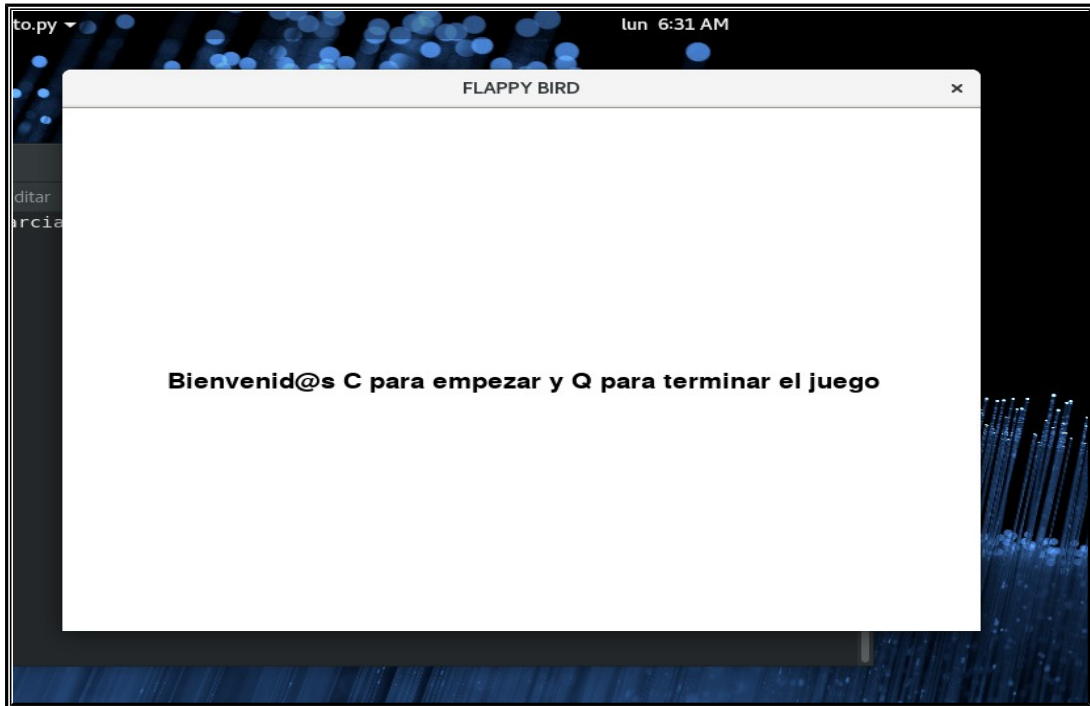
File Edit View Selection Find Packages Help

```
Project
▼ Descargas
  ~.lock.Proyecto fla.odt.enc.Illl.odt#
  FlappyMoficado.py
  Proyecto fla.odt.enc.Illl.odt
  Proyecto.enc
  Proyecto.py

Telemetry Consent
Project.py
80
81     if event.type == pygame.KEYUP:
82         if event.key == pygame.K_DOWN:
83             y_velocidad = 2
84
85     pantalla.fill(blanco)
86     obstaculos(xloc,yloc,xtamano,ytamano)
87     bola(x,y)
88     puntaje(puntos)
89     y += y_velocidad
90     xloc -= obvelocidad
91
92     if y > piso:
93         gameover ()
94
95     if x+20 > xloc and y-20 < ytamano and x-15 < xtamano+xloc:
96         gameover ()
97
98     if x+20 > xloc and y+20 > ytamano+espacio and y-15 < xtamano+xloc:
99         gameover ()
100
101     if xloc < -80:
102         xloc = 700
103         ytamano = random.randint(0,350)
104     if x > xloc and x < xloc+3:
105         puntos = (puntos+1)
106
107     pygame.display.update()
108     reloj.tick(60)
```

Projecto.py 52:1 LF UTF-8 Python 0 files

## Capturas de pantalla





## ***Resultados***

Después de muchas pruebas para la creación del código del juego Flappy Bird podemos decir que el juego ya está hecho y está listo para ser jugado.

### **El juego cuenta con:**

- un entorno virtual
- una bola
- obstáculos.

### **Funcionamiento del juego.**

- Al iniciar el juego aparece una bola
- De pronto aparecen obstáculos de diferentes tamaños, los cuales la bola debe de esquivar
- El jugador controla los movimientos de la bola hacia arriba o hacia abajo con las teclas de su computadora.
- El jugador gana un punto por cada obstáculo superado.
- Si la bola choca contra cualquier obstáculo el juego termina.

## ***Conclusión***

La creación de este juego nos a venido a fortalecer nuestros conocimientos en cuanto a la programación en el lenguaje de programación Python ya que de una u otra forma nos a forjado a investigar y conocer mucho mas acerca de la programación en dicho lenguaje.

A pesar de las dificultades que se nos presentaron a la hora de crear el código consideramos que el proceso nos ha sido de mucho provecho para poner en practica lo aprendido y aprender un poco mas en especifico sobre el entorno grafico Pygame de Python.

## ***Bibliografía.***

1. es.m.wikipedia.org
2. flappy-bird.uptodown.com
3. [www.python.org](http://www.python.org)
4. programacion.net

## ***Anexos.***

1) Flappy Bird es un juego creado para móviles de 2013 desallodo en Hanói por el desarrollador vietnamita Nguyen Ha Dong (Dong Nguyen) y publicado por GEARS Studios, un pequeño desarrollador de juegos independiente, también con sede en Vietnam. El juego, publicado el 24 de mayo de 2013, este juego rápidamente empezó a acumular descargas hasta llegar a ser el más descargado en todes las tiendas de apps, fue eliminado de App Store y Google Play por su creador el 9 de febrero de 2014.

### **2) Glosario**

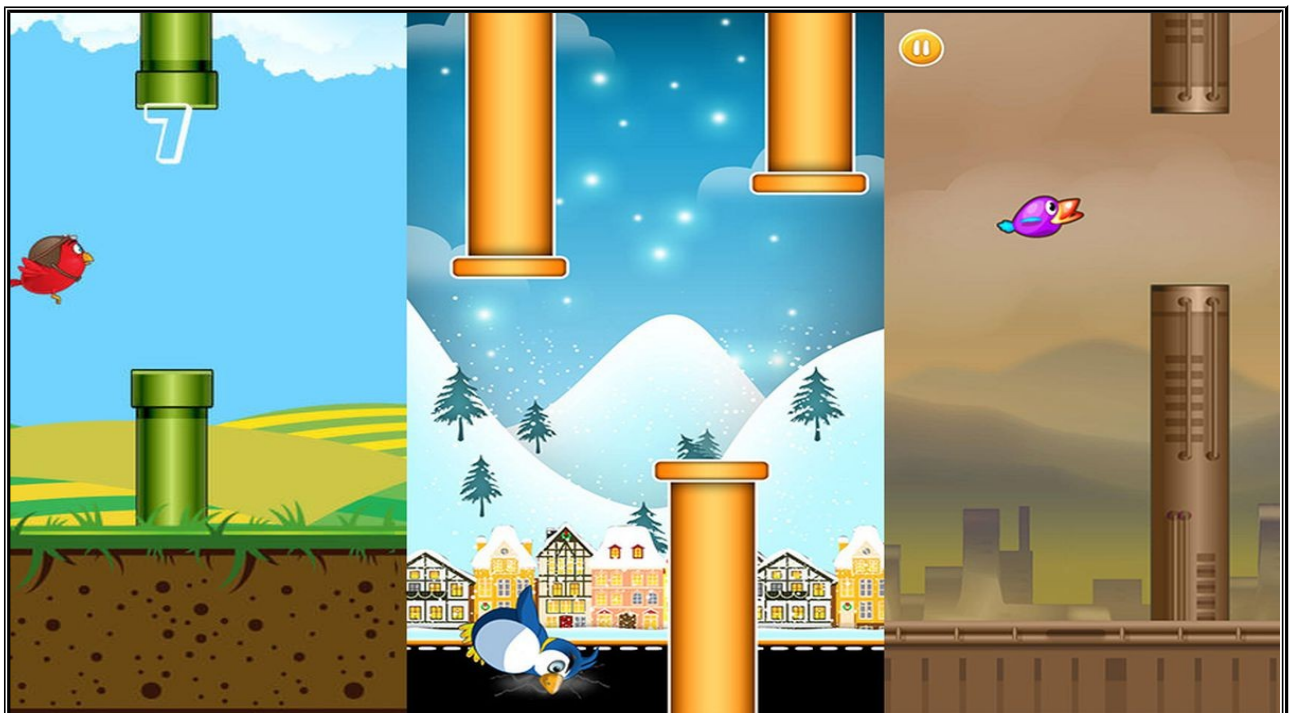
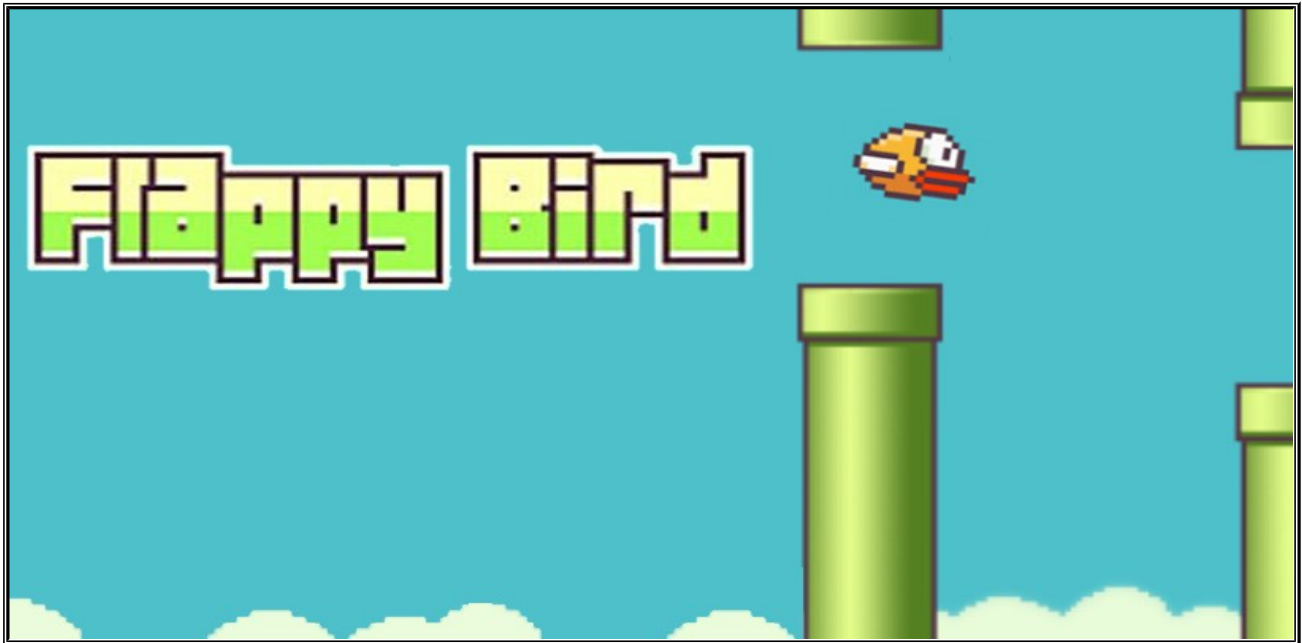
**Pygame:** es un conjunto de módulos del Lenguaje Python que permiten la creación de videojuegos en dos dimensiones.

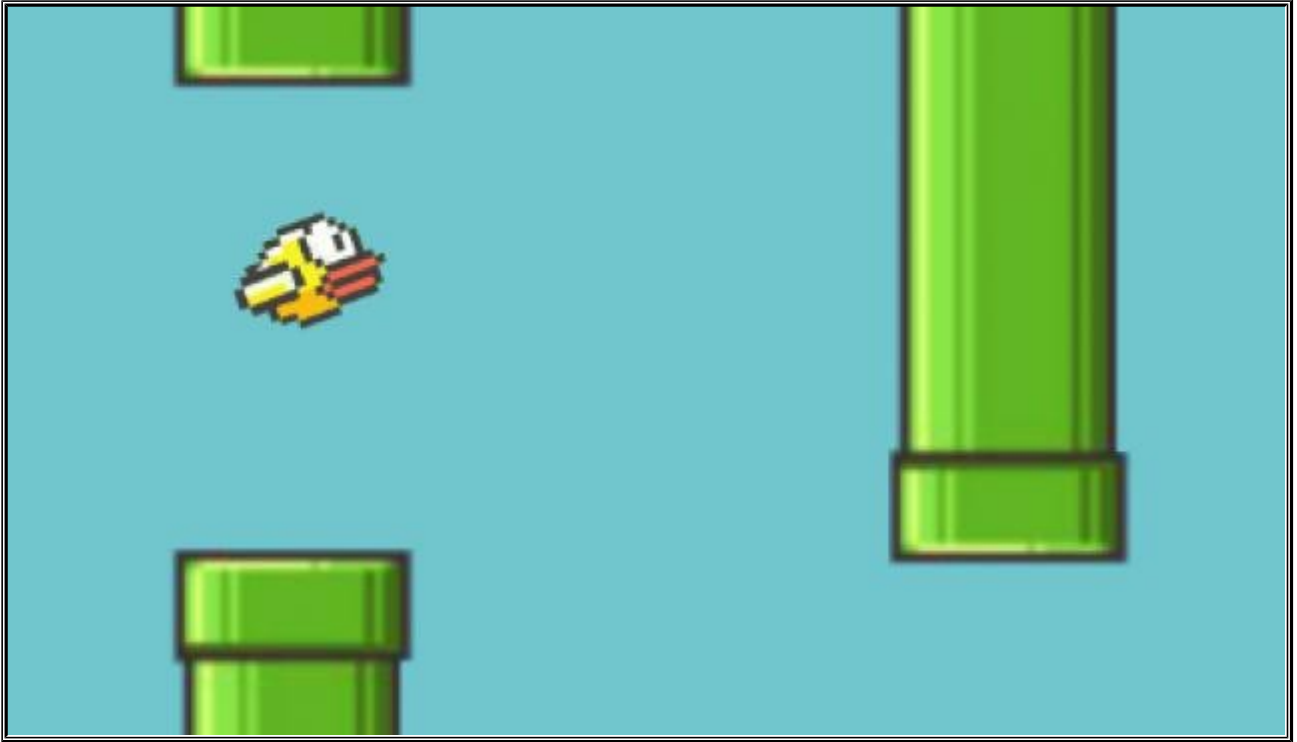
**Pixel:** unidad básica de una imagen digitalizada en pantalla a base de puntos de color.

**Parallax:** es un efecto que al entrar en una web, hace scroll el fondo parece moverse a una velocidad distinta que el contenido, en español **paraleje**.

**Scroll:** se denomina desplazar o deslizar (en inglés **scroll**), rollo o vuelta al movimiento en 2D de los contenidos que conforman el escenario de un videojuego o la ventana que se muestra en una aplicación informática.

### 3) Imágenes con fines ilustrativos.





**Referencias:**

Tomado de WikipediA.

BBC 2018.

Paisdelosjuegos.es