

Bienvenidos al Genius Guess Project.

¿Qué es?

Es una plataforma de aprendizaje diseñada en el semestre cursado 2021-I por los estudiantes Jorge Alarcón, Daniel Castellanos, David Holguín, Duván Izaquita y Fernanda Páez para la clase de Programación e Introducción a los métodos numéricos de la Universidad Nacional de Colombia, fue desarrollado por medio de lenguaje Python junto a librerías como Tkinter, Pygame y Sys con el que se busca reforzar y ampliar la conexión histórica en torno a temáticas de física y sus personajes ligados a éstas.

¿Cuál es su experiencia de usuario?

*Genius Guess* se ejecuta por medio de su acceso directo, ejecutando el menú principal y una consola de código, en el menú principal el usuario tendrá acceso a un botón denominado “Senda del Genio” en donde al dar *click* encontrará 4 botones que redirigen a nuevas ventanas con 4 cuestionarios en total de las temáticas de Mecánica, Relatividad, Astronomía y Electricidad (cabe mencionar que la ventana emergente se deberá ampliar manualmente ya que viene minimizada por defecto).

En el menú principal, también podrá acceder a 4 botones que redirigen abriendo una ventana, a los 4 minijuegos que están anclados a cada una de las temáticas anteriormente expuestas.

En el menú principal, el usuario podrá seleccionar un botón en la barra inferior donde tendrá un nuevo menú en el que podrá detallar los logros de cada curso.

En el menú principal, en la barra inferior, el usuario podrá seleccionar dos botones, uno que se refiere a una nueva ventana que muestra la sección de datos curiosos y el otro a una ventana de frases.

Temáticas y metodología.

La metodología de la plataforma se basa en preguntas de selección múltiple para manejar aspectos básicos de tópicos como el Electromagnetismo, Mecánica Newtoniana, Relatividad, Astronomía y derivados, todos acogidos de recursos bibliográficos que en su principio están pensados para formar destrezas académicas para pruebas institucionales como el ICFES, pruebas de admisión, entablando un sistema donde el usuario sólo conoce después de hacer su sesión, cuáles fueron las preguntas correctas, con la intención de que se deba repetir la prueba hasta que se resuelva satisfactoriamente de manera completa.

¿Cómo son sus mecánicas?

Juegos.

*Genius Guess* tiene múltiples juegos arcade relacionado a cada uno de los cursos, con el que se permite que el usuario interactúe desde otra perspectiva, intente alcanzar los

mayores puntos posibles y corrobore la flexibilidad del aprendizaje, en el momento existen 4 juegos disponibles, cada uno con conceptos y mecánicas independientes, además la plataforma ofrece dos apartados en forma de cartas de lectura donde se resaltan datos curiosos, referencias y frases de distintos personajes importantes.

*En primera instancia, “Scientific Flappy” es un juego de desplazamiento lateral en el que el jugador controla un pájaro extraterrestre, creado por los integrantes del grupo, intentando volar entre columnas de tubos verdes sin golpearlos. Cada vez que el pájaro vuela exitosamente entre las columnas se obtiene un punto, pero si se toca el suelo o alguna columna, se termina la partida. Adicionalmente, este juego se basó en el famoso juego de plataforma móvil “Flappy Bird”, imitando las mecánicas del pájaro y los característicos tubos verdes y la música de fondo es Paris del reconocido artista Else.*

*En segundo lugar, “Alien and Soldier” es un juego 1 vs IA el cual consiste en derrotar a un alien que ha llegado al planeta tierra para invadirlo. Con el fin de vencerlo, es necesario reducir su vida usando una pistola láser. Además, hay que evitar sus esporádicos ataques, ya sea saltando o moviéndose alrededor del campo de batalla.*

*Por otro lado, “Maxwell vs Tesla” es un multijugador 1 vs 1 2D que consiste en una batalla entre dos grandes científicos, Maxwell y Tesla. Inicialmente, ambos jugadores tienen 10 vidas; si son golpeados por un rayo gamma de la pistola del rival, pierden una vida. Quien logre atinarle 10 rayos gamma al otro rival, gana.*

*Por último, “Juego de Memoria” es un juego de agilidad mental y memoria muy divertido, apto para que niños o adultos conozcan a los científicos más reconocidos en la historia de la ciencia.*

¿Cómo se instala?

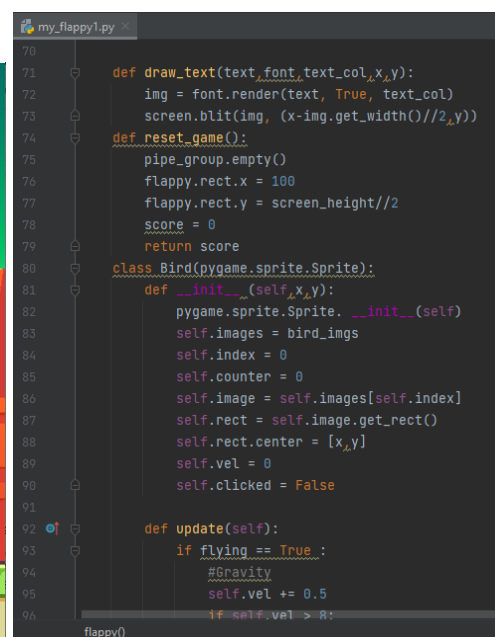
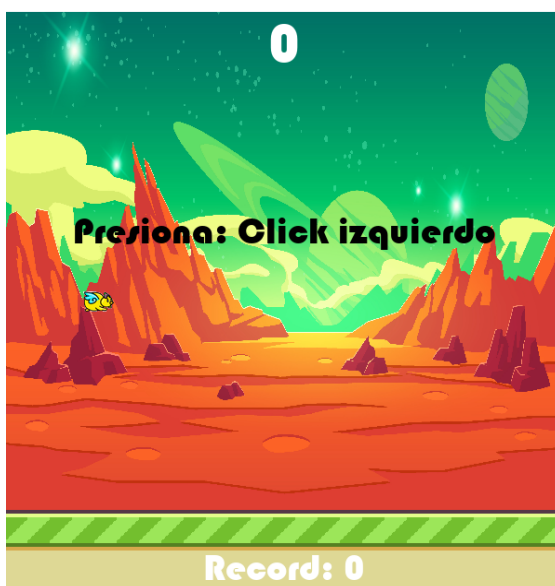
El proceso de instalación es sencillo, debes ejecutar el “mysetup.exe” y no cambiar la dirección de las configuraciones predeterminadas de donde quedará instalado. Sin embargo, tal instalable implica un complejo apartado específico de distintos procesos, que de ser reorganizados, se dan las condiciones para que se notifiquen errores, ello se debe a los métodos utilizados para que por medio de la carpeta original (donde se codificó el juego) los cuales fueron el módulo de “cx\_Freeze” y el programa “innosetup builder” se pudiera implementar un archivo de tipo msi, instalable en casi cualquier PC de SO windows, cabe destacar que el instalable que se denota dentro de este *git*, es totalmente funcional, pero tiene dos apartados jugables (el juego alusión a flappy y el juego alusión al ogro designados con los archivos “my\_flappy1.py” y “3Game\_1.py” respectivamente) que están mal indexados en lo que respecta a la dirección en donde están guardados los archivos de imágenes, por ello al intentar correrlos con el módulo “subprocess” por medio del botón de la interfaz señalado para ejecutarlo, lanza el de que no encuentra los archivos, en torno a ello, se sube una carpeta más actualizada llamada “Genius” que básicamente contiene lo que instala el “mysetup.exe” sólo que con las direcciones corregidas de estos programas anteriormente postulados. Cabe destacar que *Genius Guess* tiene por defecto instalado todas las librerías y ajustes necesarios de *Python* para su óptimo funcionamiento y que reducir tanto archivo que podría llegar a ser obsoleto pero que a la vez funcione sin ningún tipo de error es nuestro mayor reto actual.

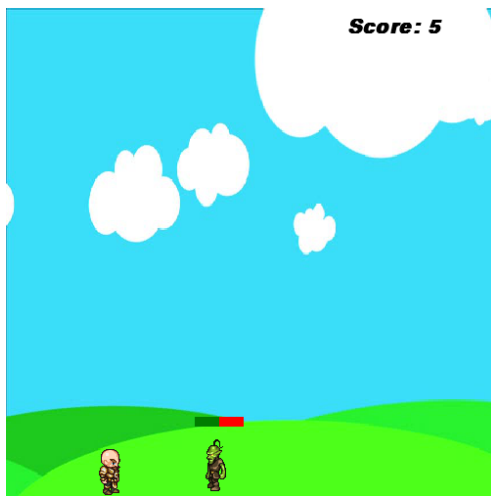
¿Cómo se desarrolló cada parte de *Genius Guess*?

Interfaz1.py: Este archivo es el menú principal, con lo primero que el usuario se encontrará al iniciar el juego (a parte de la consola) donde podrá abrir otros menús, entrar a los minjuegos, cuestionarios, frases y datos curiosos, desde aquí se coordinan todos los demás .py por medio del módulo “subprocess”, el cual le da funcionalidad a un objeto al ser accionado, en este caso, eso se traduce en que al hacer *click* en el botón destinado, por ejemplo, a abrir el juego de *Scientific Flappy*, este será ejecutado en una nueva ventana y abriendo el .py que se identifica con este minijuego.



*my\_flappy1.py*, *Juegomemoria.py*, *3Game\_1.py* y *mini\_juego.py* son programas que bajo su lenguaje Python tienen distintos desarrollos pero que comparten el mismo estilo de código, ya que todos ellos se desarrollaron por medio de librerías de *Pygame* y su técnica radica en plantear “clases” y por medio de estas, definir “funciones” con la que se procesan todos los archivos como se puede comparar en las siguientes imágenes:



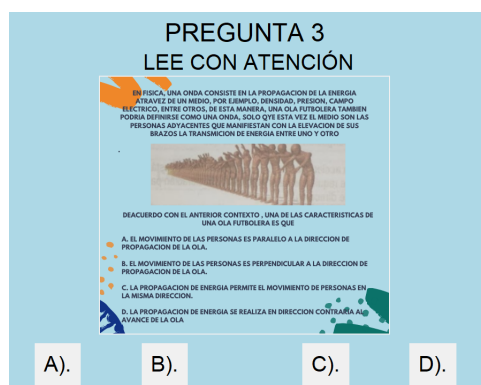


```

3Game_1.py
40 class Player(object):
41     def __init__(self, x, y, width, height):
42         self.x = x
43         self.y = y
44         self.width = width
45         self.height = height
46         self.vel = 5
47         self.isJump = False
48         self.left = False
49         self.right = False
50         self.walkCount = 0
51         self.jumpCount = 10
52         self.standing = True
53         self.hitbox = (self.x + 17, self.y + 11, 29, 52)
54         self.shootLoop = 0
55
56     def draw(self, win):
57         if self.walkCount + 1 >= 27:
58             self.walkCount = 0
59
60         if not(self.standing):
61             if self.left:
62                 win.blit(walkLeft[self.walkCount//3], (self.x, self.y))
63                 self.walkCount += 1
64             elif self.right:
65                 win.blit(walkRight[self.walkCount//3], (self.x, self.y))
66                 self.walkCount += 1

```

*MecánicaP1.py*, *AstronomíaP1.py*, *ElectricidadP1.py* y *RelatividadP1.py* son programas que bajo su lenguaje Python y el uso principal de los módulos de las librerías de *Tkinter* se desarrollan desde el mismo estilo, buscando definir por medio de clases, todas las preguntas, respuestas correctas e incorrectas con sus respectivas imágenes que funcionan como enunciado de cada una de las temáticas que componen a cada uno de los cuestionarios.



```

RelatividadP1.py
22 def comm1():
23     name = entry1.get()
24     labelk = Label(root, text='PREGUNTA 1', font=('Helvetica', 35), bg='light blue')
25     labelk.place(x=540, y=10)
26     labelk1 = Label(root, text='LEE CON ATENCIÓN', font=('Helvetica', 30), bg='light blue')
27     labelk1.place(x=500, y=70)
28     labelk2 = Label(root, image=img1)
29     labelk2.place(x=420, y=120)
30
31 def comm2():
32     global count
33     count += 1
34     labelz = Label(root)
35     labelz.pack()
36     labelz.after(10, lanjut2)
37
38 def lanjut2():
39     labell = Label(root, text='PREGUNTA 2', font=('Helvetica', 35), bg='light blue')
40     labell.place(x=540, y=10)
41     labell1 = Label(root, text='LEE CON ATENCIÓN', font=('Helvetica', 30), bg='light blue')
42     labell1.place(x=500, y=70)
43     labell2 = Label(root, image=img2)
44     labell2.place(x=420, y=120)
45
46 def comm3():
47     global count
48     count += 1

```

*codigo general datos curiosos.py* y *Codigo general frases.py* son programas que bajo su lenguaje *Python* y el uso mayoritario de la librería *Pygame* junto a una definición de “funciones” y un “while True” se desarrolló un apartado dentro del menú principal, donde el usuario puede ejecutarlos para conocer una lista de múltiples datos curiosos y frases en forma de imágenes consecutivas ligadas a los personajes más representativos de cada uno de los cursos.



```

codigo general datos curiosos.py
63
64     while True:
65         ventana.fill(color)
66         ventana.blit(imagen23,(0,0))
67         for event in pygame.event.get():
68             if event.type==pygame.MOUSEBUTTONDOWN:
69                 if cursor1.collidirect(botons.rect):
70                     imagen23=imagen24
71                     if imagen24==imagen24:
72                         imagen24=imagen25
73                     if imagen25==imagen25:
74                         imagen25=imagen26
75                     if imagen26==imagen26:
76                         imagen26=imagen27
77                     if imagen27==imagen27:
78                         imagen27=imagen28
79                     if imagen28==imagen28:
80                         imagen28=imagen29
81                     if imagen29==imagen29:
82                         imagen29=imagen30
83                     if imagen30==imagen30:
84                         imagen30=imagen31
85                     if imagen31==imagen31:
86                         imagen31=imagen32
87                     if imagen32==imagen32:
88                         imagen32=imagen33
89                     if imagen33==imagen33:

```

Agradecemos el interés por nuestro proyecto encabezado por el Profesor Ricardo Amezcuita Orozco, quien fue el guía y principal inspector de cada una de nuestras intenciones en torno a la planeación y ejecución, además de los múltiples canales de información digitales de los cuales nos apoyamos para concretar tal idea de una plataforma educativa. Finalmente, se describen una serie de ideas y detalles que se buscaría implementar en dado caso de de generar versiones futuras y próximas, como lo son crear un registro funcional en el que el usuario tenga un progreso con capacidad de guardarlo y seguir donde lo dejó la última vez, que los datos y frases sean aleatoriamente y no de manera consecutiva, implementar un tipo de cuestionario más interactivo, que se pueda personalizar el avatar presente en el menú principal por medio de un sistema de puntos y monedas.