

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS

**PROGRAMACIÓN AVANZADA**

**PROYECTO FINAL**

**JUEGO: PAC-JUMP**

**Desarrollado por:**

Carolina González

Victor Pacha

**Profesor:**

Ing. Juan Pablo Zaldumbide

**Fecha de entrega:**

Jueves 11 de agosto de 2016

**INTRODUCCIÓN**

En el transcurso del presente semestre hemos estado aprovechando el potencial de un nuevo lenguaje de programación, Python presenta un sin número de ventajas al momento de elaborar varios programas y aplicaciones con distintas funcionalidades, sin lugar a duda es indispensable conocer todo lo que se pueda del potencial de este lenguaje el cual requiere de un esfuerzo no tan considerable en comparación con otros lenguajes al momento de aprender toda su sintaxis y manejo de datos y demás requerimientos que requiere un lenguaje propiamente dicho.

Para poder obtener un dominio el cual sea de un nivel capaz de permitirnos desenvolvernos a nivel profesional se propuso que para el final de este curso se realice un proyecto que incluya todos los aspectos que cubren el aprendizaje de este lenguaje, el cual consiste en elaborar un juego funcional, innovador que sea entretenido.

**OBJETIVOS**

* Crear un juego capaz de incluir todos los temas vistos en el curso

**OBJETIVOS ESPECÍFICOS**

* Resolver todos los problemas que pudieren presentarse
* Analizar si el objetivo se cumplió

**JUEGO: PAC-JUMP**

Debido a que existen varios juegos que conservan las mismas características variando solamente quizás en las imágenes usadas para la animación, música de fondo y sonido se buscó una alternativa que quizás en la cual el jugador pueda sentirse identificado, llamando la atención también a gente que quizás no esté relacionado con los videojuegos, que sea amigable y de fácil comprensión para ser jugado, es por esto que se tomó la decisión de escoger una actividad que requiera de destreza en la vida real y esa se pueda trasladar al videojuego.

El presente proyecto constara de un menú inicial y el juego propiamente dicho, el cual consistirá en hacer que el jugador ingrese la opción en el menú para empezar el juego, una vez ingresado en el juego el jugador tiene que interactuar tratando que la figura central no toque con la animación de la soga, de esta forma tratamos de simular un juego de saltar la soga.

**CRONOGRAMA DE DESARROLLO DE ACTIVIDADES**

Las actividades estimadas a cumplir serán las siguientes:

* Elaborar un bosquejo de lo que se desea hacer en el proyecto
* Planteamiento de los problemas a resolver
* Elaborar el objetivo principal del proyecto
* Investigar las librerías a usarse
* Escoger el dispositivo con el que se va a interactuar e investigar cómo se lograra que se lo reconozca dentro del programa.
* Comentar el código del juego
* Elaborar y diseñar los diferentes niveles
* Mejorar sus características

**HERRAMIENTAS**

* Interprete Python: para la ejecución del código que se implementara al juego
* Editor de texto: Sublime Text, necesario para codificar proyectos de gran tamaño que requieran un control tanto de la sangría como de la estructura
* Módulo de lenguaje de Python Pygame, el cual cargara todas las librerías necesarias para la ejecución de animaciones, sonido y toso lo requerido para el juego
* Adobe Ilustrator para los gráficos de las animaciones, cuando sea necesario renderizar los gráficos, ajustar su tamaño en pixeles para las animaciones
* AIMP Converter para la conversión del formato de sonido, esta fue na herramienta necesaria para trabajar transformando las extensiones de los tipos de archivos de sonido, ya que a veces se necesitó cambiar su formato para que sea compatible con Python y su modulo Pygame

**CODIFICACIÓN**

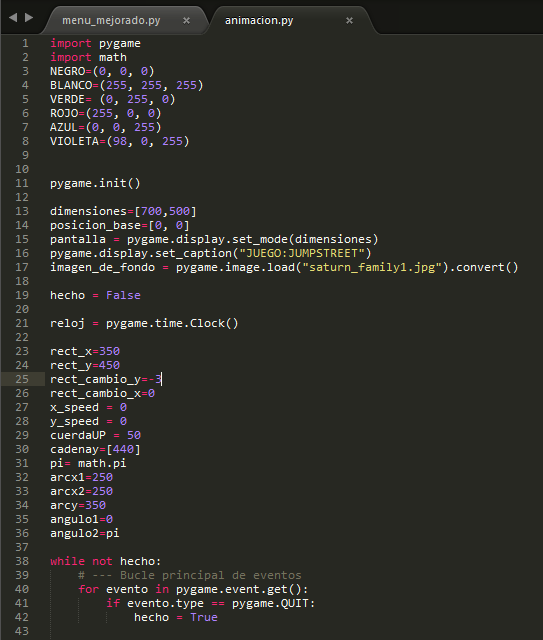
El paradigma a emplear será el de Programación Orientada a Objetos, el cual nos permite usar objetos e interactuar con ellos, para diseñar aplicaciones y programas informáticos, este paradigma está basado en varias técnicas, incluyendo la herencia, cohesión, abstracción, polimorfismo, acoplamiento y encapsulamiento.

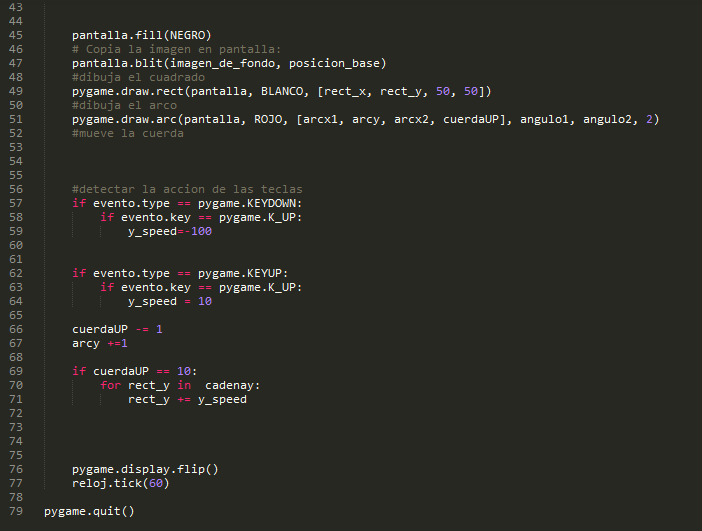
Además nos permite tener una visión más natural de nuestro problema a resolver, existe la ventaja adicional en la reducción de código, muy útil al momento de querer corregir errores luego de haber estado codificando una gran cantidad de líneas de código.

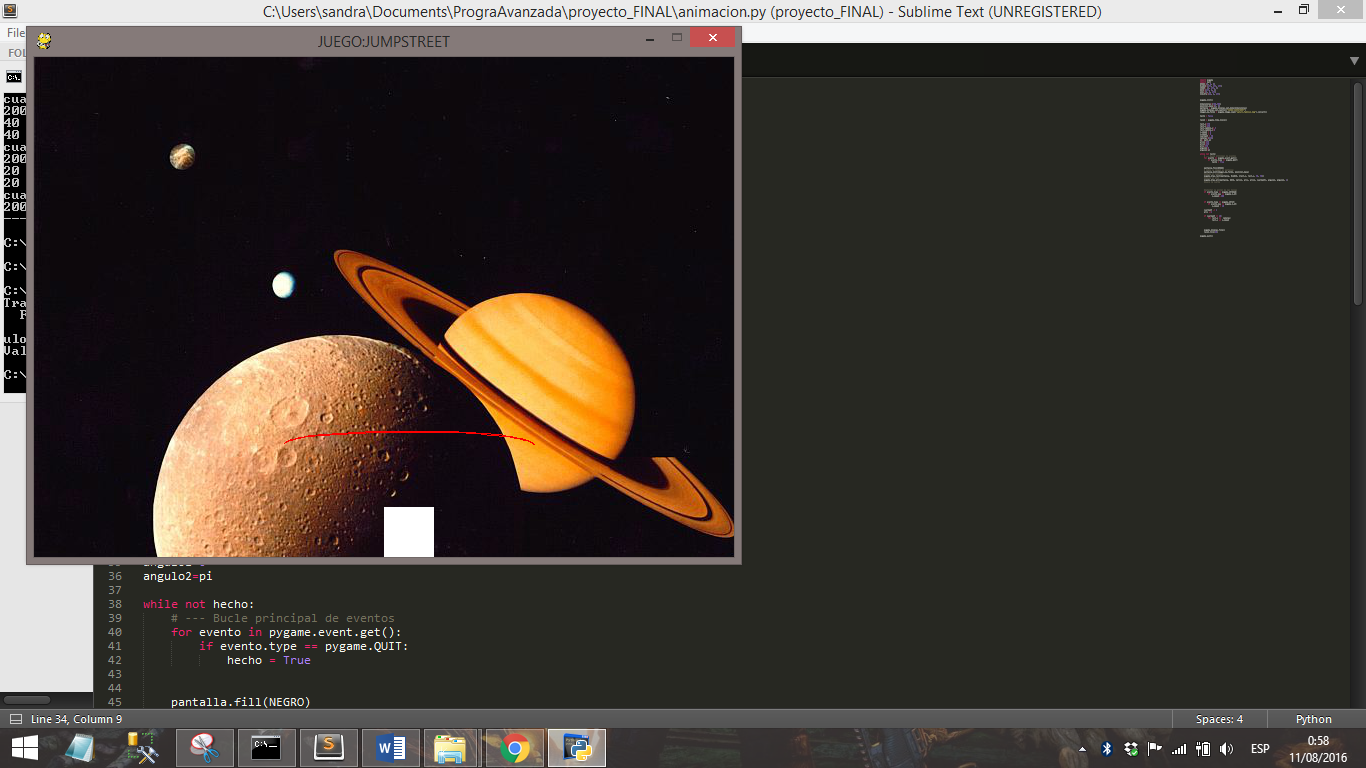
Al utilizar un editor de texto tenemos la desventaja de que no podemos analizar los errores antes de compilar el código, tenemos en cierta manera organizarnos para tener una estructura coherente en como distribuimos nuestro código, y el principal problema que tiene Python es la sangría que sirve para identificar o separar las estructuras de decisión como de las funciones que se usaran en todo el programa.

**RESULTADOS PREVIOS**

En nuestro primer intento se logró incluir el salto del jugador y un movimiento de soga, nos ayudamos del movimiento de un rectángulo para el jugador y de un arco para el movimiento de la soga







**RESULTADO FINAL**

Luego de la corrección de errores y la implementación del menú se obtuvo los siguientes resultados del proyecto

Pantalla inicial que muestra las opciones que el jugador necesitara

Pantalla del juego en donde el usuario interactuara con nuestra aplicación

**Menú de inicio**



**Instrucciones**



**Juego**



**CONCLUSIONES**

* Elaborar este juego resulto muy desafiante, ya que no se contó con la ayuda de ejemplos previos que podían en alguna forma servir de guía, el movimiento de la soga y el salto del jugador fue lo más complicado de realizar.
* El resultado final es muy satisfactorio ya que se pudo realizar un juego, en el cual el jugador interactúa tanto con el movimiento del jugador y sincronizándolo para que no toque la soga.

**REFERENCIAS**

|  |  |
| --- | --- |
| [1] | A. R. Verdugo, «Programas Juegos Arcade con Python y Pygame,» 2015. [En línea]. Available: http://programarcadegames.com/index.php?lang=es. [Último acceso: 28 Junio 2016]. |