Introducción a la Programación SC-115

DREAMWORLD CASINO

**Prof. Alvaro Camacho Mora**

**Integrantes:**

* Maccoy Artola, Brayton
* Calvo Brenes, Kevin Alberto
* Navarro Román, Jefferson Andrey

II CUATRIMESTRE 2023

22 de agosto, 2023

Contenido

[1 PLANTEAMIENTO DE LA INVESTIGACIÓN 2](#_Toc143628369)

[1.1 Introducción 2](#_Toc143628370)

[1.2 Objetivos 3](#_Toc143628371)

[1.2.1 Objetivo general 3](#_Toc143628372)

[1.2.2 Objetivos específicos 3](#_Toc143628373)

[2 DESCRIPCION DE MODULOS 4](#_Toc143628374)

[2.1 Registro de usuario: 4](#_Toc143628375)

[2.2 DreamWorld Casino: 5](#_Toc143628376)

[2.3 Retirar dinero: 6](#_Toc143628377)

[2.4 Depositar dinero: 7](#_Toc143628378)

[2.5 Menú de juegos: 8](#_Toc143628379)

[2.6 21 blackjack: 9](#_Toc143628380)

[2.7 Tragamonedas 11](#_Toc143628381)

[3 REQUERIMIENTOS DEL SISTEMA 12](#_Toc143628382)

[3.1 Descripción de requerimientos 12](#_Toc143628383)

[4 BIBLIOTECAS / ESTRUCTURAS COMPLEMENTARIAS 13](#_Toc143628384)

[4.1 Biblioteca 13](#_Toc143628385)

[4.2 Biblioteca “os” 13](#_Toc143628386)

[4.3 Biblioteca “random” 13](#_Toc143628387)

[4.4 Biblioteca “time” 13](#_Toc143628388)

[4.5 Biblioteca getpass 14](#_Toc143628389)

[5 MANUAL DE USUARIO 14](#_Toc143628390)

[5.1 Primeros pasos 14](#_Toc143628391)

[5.2 Registro de usuario 15](#_Toc143628392)

[5.3 DreamWorld Casino 15](#_Toc143628393)

[6 CONCLUSIONES Y RECOMENDACIONES 16](#_Toc143628394)

[6.1 Conclusión 1 16](#_Toc143628395)

[6.2 Conclusión 2 16](#_Toc143628396)

[6.3 Conclusión 3 16](#_Toc143628397)

[A lo largo de las 15 semanas del curso, se adquirió un conocimiento profundo de la programación y se aplicó de manera efectiva en el desarrollo de DreamWorld Casino. Los desafíos encontrados durante el proceso de implementación y el análisis nos llevaron a investigar y fortalecer más nuestros conocimientos. 16](#_Toc143628398)

[7 BIBLIOGRAFIA 17](#_Toc143628399)

# PLANTEAMIENTO DE LA INVESTIGACIÓN

## Introducción

En un mundo cada vez más interconectado y digitalizado, la industria del entretenimiento en línea ha experimentado un crecimiento exponencial. En este contexto, la empresa Global Casinos Inc. ha vislumbrado una oportunidad única para llevar la emoción y la diversión de un casino de alta gama a la comodidad de los hogares de usuarios en todo el mundo. Este desafío da lugar al nacimiento de DreamWorld Casino, un innovador casino en línea que busca fusionar la sofisticación de los casinos físicos con la accesibilidad global proporcionada por la tecnología.

La presente propuesta de investigación se centra en el desarrollo de un programa computacional utilizando el lenguaje de programación Python, que permitirá la creación de un casino básico en línea. La finalidad de este proyecto es explorar las habilidades de programación adquiridas para diseñar y construir una plataforma que refleje la experiencia única y emocionante de un casino real, al tiempo que garantice la seguridad financiera de los usuarios y ofrezca un sistema de registro, juegos en línea y opciones avanzadas de configuración.

En este sentido, el proyecto se orienta hacia la creación de un software que cumpla con las más altas medidas de seguridad, asegurando la confidencialidad y protección de la información financiera de los usuarios. Además, se abordará la implementación de una plataforma de registro de usuarios que permita la creación de perfiles individuales, estableciendo así un entorno personalizado y único para cada jugador.

El núcleo de la investigación recae en el desarrollo de juegos en línea que emulen la experiencia de un casino tradicional. Se busca diseñar algoritmos y sistemas que generen resultados aleatorios y justos en juegos como tragamonedas y 21 Blackjack. Estos juegos serán el corazón de DreamWorld Casino, brindando a los usuarios la emoción y la adrenalina característica de los casinos físicos.

## Objetivos

### Objetivo general

Crear un software informático utilizando el lenguaje de programación Python que aplique y ponga en práctica las destrezas de programación adquiridas.

### Objetivos específicos

#### Objetivo 1:

* Evidenciar la comprensión de los requisitos del proyecto mediante un buen análisis que nos guie a concretar el proyecto de una manera exitosa.

#### Objetivo 2:

* Presentar la estructura implementada mediante habilidades comunicativas adecuadas, con el propósito de llevar a cabo la solución al problema.

#### Objetivo 3:

* Demostrar el conocimiento aprendido durante las 15 semanas del curso y llegándolo a aplicar en el proyecto.

# DESCRIPCION DE MODULOS

## Registro de usuario:

Problema: Hacer una función para registrar usuarios en el casino

1. Definir el número máximo de intentos y obtener el valor mínimo de depósito [variable]
2. Inicializar el contador de intentos en 0
3. Iniciar un bucle que se ejecutará mientras el contador de intentos sea menor que el máximo de intentos permitidos [while, variables]
4. Solicitar al usuario ingresar un nombre de usuario o ID [variable, input]
   1. Si el ID es menor a 5 caracteres, mostrar un mensaje de error [if/else, len, print]
   2. Si el ID cumple los requisitos, crear el nombre del archivo de usuario basado en el ID proporcionado
   3. Si el ID ya existe, mostrar un mensaje de usuario ya registrado y volver al menú principal [if/else, print]
   4. Si el archivo no existe, continuar
5. Solicitar el nombre del usuario [input]
6. Solicitar y ocultar la entrada del PIN [input]
   1. Si el PIN tiene menos de 6 dígitos, mostrar un mensaje de error [if/else, print]
7. Solicitar al usuario ingresar el monto de depósito [input, float]
   1. Si el monto es mayor o igual al mínimo requerido: [if/else]
8. Crear un archivo con el nombre del usuario y escribir la información del usuario (nombre, monto depositado y PIN)
9. Crear un archivo separado para el monto depositado
10. Mostrar un mensaje de registro exitoso y volver al menú principal [print, return]
    1. Si el monto es insuficiente, mostrar un mensaje de error [if/else, print]
11. Incrementar el contador de intentos [variable, suma+]
    1. Mostrar cuántos intentos restantes quedan [print]

d. Si el bucle se ejecuta tres veces sin éxito, mostrar un mensaje de intentos agotados y volver al menú principal.

## DreamWorld Casino:

Problema: Crear la aplicación principal del casino donde se albergarán muchas de las funciones

1. Se definen los valores de número máximo de intentos de ingreso de ID y PIN.Deben estar en 0 [variables]
2. Se inicia un bucle while que se ejecutará mientras los contadores no superen los intentos máximos [while, operaciones]
3. Se solicita al usuario ingresar su nombre de usuario [input]
4. Se crea el nombre del archivo del usuario [biblioteca os]
5. Si el archivo del usuario existe: [if/else]
   1. Se inicia un segundo bucle mientras no se haya excedido el límite de intentos [while]
6. Se intenta abrir el archivo del usuario y cargar los datos [try]
7. Se solicita al usuario ingresar su PIN de manera segura [input, getpass]
8. Si el PIN ingresado coincide con el PIN registrado: [if/else]
   1. Se muestra "Bienvenido" [print]
   2. Se llama a la función submenuDreamWorld y se regresa [función, return]
9. Si el PIN ingresado es incorrecto: [if/else]
   1. Se muestra un mensaje de "Pin Incorrecto, intente de nuevo" [print]
   2. Se incrementa el contador de intentos de ingreso de PIN [variable, operación]
   3. Si se alcanza el máximo de intentos de ingreso de PIN, se vuelve al menú principal [break]
10. Usar una excepción si no se encuentra el archivo del usuario, se muestra un mensaje de usuario no encontrado y se sale del segundo bucle [except, break]
11. Si el archivo del usuario no existe: [if/else]
    1. Se muestra "Usuario no encontrado. Vuelva a intentarlo". [print]
    2. Se incrementa el contador de intentos
    3. Si se alcanza el máximo de intentos de ingreso, se vuelve al menú principal [llamara función]

## Retirar dinero:

Problema: Crear función para que el usuario pueda retirar el dinero que haya ganado o con el que cuente en el casino

1. Crear la función de retirar dinero
2. Declarar una variable de saldo actual para poder modificarla.
3. Variable para intentos máximos (intentos = 3).
4. Hacer uso de un try para lo siguiente:
5. Abrir el archivo de depósitos
6. Permitir que se lea el archivo y asignarle valor al saldo actual
7. Cierra el archivo
8. Si el archivo no se encuentra, muestra un mensaje de error y retorna al inicio
9. Usar un bucle while
10. Try:
    1. Solicita al usuario ingresar el monto que desea retirar
    2. Si el valor ingresado no es un número válido, mostrar mensaje de error y continúar con la siguiente iteración del bucle.
11. Si el monto de retiro es menor o igual a 0, muestra un mensaje de error y continuar con el bucle
12. Si el monto de retiro es mayor que el saldo actual:
    1. Decrementa intentos en 1
    2. Muestra un mensaje de saldo insuficiente y muestra la cantidad de intentos restantes
    3. Si intentos llega a 0, muestra un mensaje de límite de intentos alcanzado y se retorna al inicio
13. Si el monto de retiro es válido y menor o igual al saldo actual:
    1. Resta el monto de retiro al saldo actual
    2. Muestra un mensaje de retiro exitoso y muestra el saldo restante.
14. Abrir el archivo de depósito con la información
    1. Escribe el nuevo saldo en el archivo
    2. Cerrar el archivo de depósitos
15. Retornar al inicio del menú

## Depositar dinero:

Problema: Función para que usuario pueda hacer el ingreso de dinero inicial y cada ves que quiera depositar más dinero

1. Inicio de función
2. Abrir el archivo de depósitos
3. Que lea el contenido del archivo
4. Cerramos el archivo
5. Si el archivo no se encuentra, muestra un mensaje de error y devolver al inicio
6. Define la variable con las divisas para el casino
7. Muestra las divisas soportadas y sus tasas de conversión
8. Bucle para depositar el dinero
9. Solicita al usuario ingresar la opción de moneda en la que desea depositar el dinero
10. Si la opción ingresada es válida, sale del bucle. De lo contrario, muestra un mensaje de error
11. Si ocurre un error de conversión, mostrar un mensaje de error y vuelve a solicitar la entrada
12. Crea una lista de monedas a partir de las claves del diccionario
13. Obtiene la moneda seleccionada a partir de la lista de monedas y su tasa de conversión
14. Solicita al usuario el monto que desea depositar
15. Si el monto es menor o igual a 0, muestra mensaje de error y retorna
16. Calcula el monto de depósito en dólares dividiendo el monto ingresado por la tasa de conversión
17. Si el monto en dólares es mayor o igual al mínimo requerido:
18. Incrementa el saldo actual
19. Abrir el archivo de deposito
20. Escribir el nuevo saldo en el archivo
21. Cerramos el archivo
22. Muestra un mensaje de depósito exitoso y el saldo actual
23. Si el monto en dólares es inferior al mínimo requerido, mostrar un mensaje de monto insuficiente
24. Actualiza el valor de conversión de la moneda seleccionada en el diccionario y escribe los nuevos valores en un archivo

## Menú de juegos:

Problema: Crea una interfaz donde se encuentre el menú de los juegos del casino y el usuario seleccione lo que desea jugar

1. Iniciamos función para el menú
2. Mostrar al usuario las opciones del menú
3. Definimos una variable para que el usuario seleccione una opción
4. Estructuras condicionales para determinar la acción a tomar en función de la opción elegida:
5. Si la opción elegida es "1", llama a la función jugarBlackjack()
6. Si la opción elegida es "2", llama a la función jugarTragamonedas()
7. Si la opción elegida es "3", muestra un mensaje de despedida y vuelve al menú principal
8. Si la opción elegida no coincide con ninguna de las anteriores, muestra un mensaje de opción inválida.
9. El bucle se repetirá hasta que el usuario elija la opción "3" para salir del menú de juegos.

## 21 blackjack:

Problema: Realizar un juego de casino (21 blackjack) que sea lo parecido a la vida real y envuelva al usuario.

1. Mostrar instrucciones del juego

[Las reglas o el desarrollo del juego esta al final del documento]

1. Preguntar al usuario por la apuesta
   1. Verificar si el usuario tiene suficiente saldo
   2. Si no tiene suficiente saldo:

2.1. Imprimir mensaje y volver al submenú

1. Repartir 2 cartas al jugador y 2 al crupier
2. Mostrar las dos cartas del jugador y la primera carta del crupier
3. Preguntar al usuario si quiere doblar la apuesta
   1. Si la respuesta es sí, verificar si el usuario tiene suficiente saldo para doblar
   2. Si el usuario quiere doblar y no tiene suficiente saldo:
   3. Imprimir mensaje y continuar
4. Preguntar al jugador si quiere pedir una carta o parar
   1. Si el jugador quiere pedir una carta:
   2. Repartir una carta al jugador
      1. Verificar si la puntuación del jugador es menor o igual a 21
      2. Si es menor o igual a 21:
5. Preguntar si el jugador quiere pedir otra carta o parar
6. Si el jugador quiere pedir otra carta, repetir este paso 6
7. Si el jugador quiere parar, continuar
   * 1. Si el jugador para o su puntuación es mayor a 21:
8. Mostrar la segunda carta del crupier
   * 1. Si el jugador tiene más de 21 puntos:
9. Imprimir mensaje de que el crupier gana
   * 1. Si el jugador tiene 21 puntos:
10. Mostrar mensaje de Blackjack (Gana jugador) si corresponde
    * 1. Si el jugador tiene menos de 21 puntos:
11. Preguntar al crupier si debe pedir carta (puntuación < 17)
12. Si el crupier debe pedir carta:
13. Repartir una carta al crupier
14. Verificar si el crupier supera los 21 puntos
15. Si el crupier supera 21 puntos:
16. Imprimir mensaje de que el jugador gana
17. Si el crupier tiene 17 o más puntos:
18. Comparar puntuaciones del jugador y el crupier
    * 1. Si el jugador tiene más puntos:
    1. Imprimir mensaje de que el jugador gana
       1. Si el crupier tiene más puntos o empate:
    2. Imprimir mensaje de que el crupier gana o empate
       1. Si el jugador gana:
19. Incrementar saldo del jugador
    * 1. Si hay empate:
20. Devolver la apuesta al jugador
    * 1. Mostrar saldo actual del jugador

g. Preguntar si el jugador quiere jugar nuevamente

h. Si el jugador quiere jugar nuevamente:

* + 1. Volver al inicio

1. Si el jugador no quiere jugar nuevamente:
   * 1. Volver al submenú

## Tragamonedas

Problema: Crear el juego de casino tragamonedas apto para que le usuario pueda apostar como en la vida real

1. Mostrar instrucciones del juego
2. Repetir hasta que el jugador decida salir:
3. Preguntar al jugador por la apuesta
4. Verificar si el jugador tiene suficiente saldo
5. Verificar si el jugador tiene suficiente saldo:
   1. Imprimir mensaje y volver al inicio del bucle
6. Mostrar mensaje para jalar la palanca
7. Esperar a que el jugador presione Enter
8. Generar tres figuras aleatorias (@, #, +, 7) y mostrarlas con intervalos de 1.5 segundos
9. Contar el número de juegos jugados
10. Determinar si el jugador ganó
11. Si el jugador ganó:
    1. Calcular premio y actualizar saldo
    2. Si el jugador ganó el acumulado:
       1. Reiniciar el acumulado
12. Mostrar saldo actual
13. Preguntar al jugador si quiere jugar nuevamente
14. Si el jugador quiere jugar nuevamente:
15. Volver al inicio del bucle principal
16. Si el jugador no quiere jugar nuevamente:
17. Imprimir mensaje de despedida

# REQUERIMIENTOS DEL SISTEMA

## Descripción de requerimientos

El sistema para el desarrollo del proyecto de un casino requerirá el uso del lenguaje de programación Python y un entorno de desarrollo para escribir, probar y depurar el código. Será necesario implementar un mecanismo para almacenar y recuperar datos de usuario, como nombres, saldos y configuraciones de juego, utilizando archivos o una base de datos. La interfaz de usuario permitirá a los usuarios interactuar con el casino a través de una interfaz de línea de comandos simple y el uso de bibliotecas como “os”, “random” entre otras. Para simular los resultados de los juegos de casino, se utilizará la biblioteca random de Python para generar números aleatorios. Además, habrá validación de datos para garantizar que los usuarios ingresen información válida, y opciones de configuración avanzada para ajustar parámetros como apuestas y reglas del juego. El manejo de excepciones será implementado para gestionar errores inesperados, y se incluirán comentarios y documentación en el código para facilitar la comprensión de este. Con pruebas esperamos asegurar el correcto funcionamiento en diferentes escenarios y condiciones. Finalmente, poder hacer entrega del proyecto para que pueda ser evaluado por usuarios.

# BIBLIOTECAS / ESTRUCTURAS COMPLEMENTARIAS

## Biblioteca

Una biblioteca es un conjunto de archivos que se utiliza para desarrollar software. Suele estar compuesta de código y datos, y su fin es ser utilizada por otros programas de forma totalmente autónoma. Simple y llanamente, es un archivo importable. (BOTTEGA, 2021)

Las bibliotecas pueden contener una amplia gama de funciones y componentes, como algoritmos, estructuras de datos, utilidades y controladores para interactuar con dispositivos o servicios externos. Al utilizar una biblioteca, los programadores pueden aprovechar el trabajo previo realizado por otros, lo que ahorra tiempo y esfuerzo, además de contribuir a la modularidad y reutilización del código. (AulaDig, 2022)

## Biblioteca “os”

La biblioteca "os" en programación se refiere a la biblioteca "os" (operating system) que proporciona funciones para interactuar con el sistema operativo en el que se está ejecutando el programa. Esta biblioteca es parte de muchas plataformas de programación y se utiliza para realizar diversas tareas relacionadas con la gestión de archivos, directorios y otros aspectos del sistema operativo.

En el caso de nuestro proyecto la biblioteca “os” va a ser utilizada para la manipulación de archivos, eso conlleva la creación, copia, renombre y eliminación de archivos.

## Biblioteca “random”

Esta biblioteca tiene como función generar números aleatorios. La generación de números aleatorios es útil en muchos escenarios de programación, como la simulación de eventos aleatorios, la generación de datos de prueba, la implementación de algoritmos que requieren elementos aleatorios y más. (Marco, 2019)

La biblioteca "random" generalmente incluye funciones que permiten generar números aleatorios de diferentes tipos y rangos. Esta biblioteca va a ser muy importante a la hora de la programación de los juegos del casino.

## Biblioteca “time”

Este módulo proporciona varias funciones relacionadas con el tiempo. Para ver la funcionalidad relacionada, consulte también los módulos de fecha y hora y calendario.

Aunque este módulo siempre está disponible, no todas las funciones están disponibles en todas las plataformas. La mayoría de las funciones definidas en este módulo llaman a funciones de biblioteca de la plataforma C con el mismo nombre. A veces puede ser útil consultar la documentación de la plataforma, porque la semántica de estas funciones varía entre plataformas. (Software, 2023)

Para fines de este proyecto utilizaremos “time.sleep(**segundos**)” para poder pausar o hacer que se ejecute una función cada cierto tiempo en uno de los juegos del casino

## Biblioteca getpass

Es un módulo que proporciona una manera segura de solicitar al usuario contraseñas o información sensible sin mostrarla en la pantalla. Generalmente, se utiliza para capturar contraseñas de manera oculta, evitando que queden expuestas en la pantalla mientras el usuario las escribe. Esto es especialmente útil en entornos en los que la seguridad y la privacidad son importantes, como en aplicaciones de autenticación o interacciones con credenciales. (Sphinx, 2023)

# MANUAL DE USUARIO

## Primeros pasos

Al momento de hacer primer contacto con el DreamWorld Casino aparecerá la siguiente interfaz.

Texto

Descripción generada automáticamente

Donde las opciones “a”, “b” y “d” son para el usuario que desea hacer uso del casino. Como se puede apreciar en la imagen la opción “a” es para el usuario que esta teniendo un primer contacto con el casino de debe hacer registro de su usuario para facilitar el acceso en próximas ocasiones que el usuario desee ingresar.

La opción “b” nos permitirá entrar a las funciones principales del casino como los juegos y opciones de depositar y retirar dinero. Esto lo veremos más adelante.

Por último, como opción “d” tenemos salir que como su nombre lo indica le permitirá al usuario salir del casino en caso de que no desee continuar.

## Registro de usuario

En el momento de seleccionar la opción de registrar usuario, al usuario le aparecerán las siguientes opciones/requisitos para poder registrar de forma exitosa el usuario.

Texto

Descripción generada automáticamente

Como se observa en la imagen es necesario contar con un nombre de usuario, el nombre del usuario que se esta registrando y un pin de seguridad para verificar el ingreso. Por ultimo se le solicita hacer un depósito mínimo de 1000 dólares para poder ingresar al casino. Al completar estos requisitos se hará un registro exitoso del usuario, con este paso hecho ya se podrá ingresar al DreamWorld Casino.

## DreamWorld Casino

Esta es la función principal del casino ya que aquí se podrá encontrar opciones como los juegos del casino, ver el saldo actual del casino, depósitos y retiros de dinero.

Primeramente, se debe verificar el usuario para poder ingresar a esta opción, por eso es muy importante realizar la opción de registro de usuario ya que el sistema solicitara el nombre de usuario y el pin.

Una captura de pantalla de un celular

Descripción generada automáticamente con confianza media

Cuando el usuario ya se encuentre verificado el sistema del casino le dará la bienvenida para que el usuario pueda escoger la opción que desee en el momento.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Para cada una de las opciones el casino está diseñado para guiar al usuario dependiendo la opción que seleccione y pueda seguir los pasos mediante las instrucciones que arroje el sistema, permitiendo al usuario sentirse más cómodo a la hora de usar el casino.

# CONCLUSIONES Y RECOMENDACIONES

## Conclusión 1

Se intenta llevar la emulación, emoción y la adrenalina de los casinos físicos mediante la creación de un casino en línea demostrando cómo la tecnología puede llevar la experiencia de juego a los hogares de usuarios en todo el mundo.

## Conclusión 2

El análisis de los requisitos permitió una comprensión profunda de las necesidades del proyecto, guiando la implementación exitosa de la estructura del casino.

## Conclusión 3

## A lo largo de las 15 semanas del curso, se adquirió un conocimiento profundo de la programación y se aplicó de manera efectiva en el desarrollo de DreamWorld Casino. Los desafíos encontrados durante el proceso de implementación y el análisis nos llevaron a investigar y fortalecer más nuestros conocimientos.

# BIBLIOGRAFIA

AulaDig. (23 de 3 de 2022). *Qué es una librería de programación*. Obtenido de https://auladig.co/que-es-una-libreria-de-programacion/

BOTTEGA, d. b. (24 de 10 de 2021). *Qué es una librería en programación - DevCamp*. Obtenido de https://devcamp.es/que-es-libreria-programacion/

Foundation., P. S. (19 de 8 de 2023). Obtenido de https://docs.python.org/3/library/time.html

Marco, B. S. (4 de 4 de 2019). *Valores aleatorios: la biblioteca random*. Obtenido de https://www.mclibre.org/consultar/python/lecciones/python-biblioteca-random.html

Software, P. (19 de 8 de 2023). *time — Time access and conversions*. Obtenido de https://docs.python.org/3/library/time.html

Sphinx. (4 de 7 de 2023). *getpass — Entrada de contraseña portátil — documentación ...* Obtenido de https://docs.python.org/es/3.9/library/getpass.html