**DREAM WORLD CASINO**

**INTEGRANTES:**

**CORDERO FALLAS YERLIN**

**ORTIZ ACUÑA LEONARDO ALBERTO**

**ROJAS MIRANDA KENDALL GERARDO**

**CAMACHO PICADO FERYHERI**

**UNIVERSIDAD FIDELITAS**

**PROGRAMACION BASICA**

**ALVARO CAMACHO MORA**

**2023**

Contenidos

[Introducción 3](#_Toc141809279)

[Objetivos 3](#_Toc141809280)

[Descripción del problema 4](#_Toc141809281)

[Estructuras de software 4](#_Toc141809282)

[Algoritmo 4](#_Toc141809283)

[Requerimientos del sistema 5](#_Toc141809284)

[Manual de usuario 6](#_Toc141809285)

[Bibliografía 6](#_Toc141809286)

# Introducción

El juego ha sido parte inherente de la cultura humana desde tiempos inmemoriales, y los casinos han sido espacios emblemáticos para la recreación y el entretenimiento a lo largo de la historia. La primera evidencia documentada de un casino data del siglo XVII en Venecia, Italia, donde se estableció el famoso Casino di Venezia en el año 1638. Desde entonces, los casinos han experimentado un crecimiento constante, evolucionando y adaptándose a las cambiantes preferencias y tecnologías (Schwartz, 2006).

Con el surgimiento de internet, la industria del juego experimentó una transformación significativa. A mediados de la década de 1990, los primeros casinos en línea comenzaron a operar, permitiendo a los jugadores disfrutar de sus juegos favoritos desde la comodidad de sus hogares. A lo largo de los años, la popularidad de los casinos en línea ha aumentado considerablemente, y hoy en día son una parte integral del panorama de entretenimiento virtual (Crawford, Gosling y Light, 2013).

Este proyecto académico se centra en la programación y diseño de un casino en línea, tomando en cuenta las lecciones aprendidas a lo largo de curso de programación básica (SC-115). Este casino contará con dos juegos en específico; blackjack y tragamonedas. Se considerarán aspectos esenciales para el desarrollo de la plataforma, como la seguridad y protección de los usuarios, la implementación de algoritmos de generación de números aleatorios para garantizar la equidad de los juegos, y la integración de un sistema de pagos que guarde registro de cuánto dinero poseen los jugadores.

En resumen, este proyecto de programación del casino en línea DreamWorld pretende ser una práctica relevante para el desarrollo de las habilidades de programación, trabajo en equipo y organización de potenciales proyectos comerciales a los que nos enfrentaremos en el futuro profesional. Además, se busca fomentar el conocimiento sobre el desarrollo de plataformas seguras y responsables (en este caso, en el ámbito del juego en línea), contribuyendo así al avance y mejora de las diferentes industrias con las que se podrá trabajar en el futuro profesional.

# Objetivos

Objetivo general:

Desarrollar un programa para la empresa Global Casinos que le permita a la empresa el manejo de su casino online llamado DreamWorld Casino.

Objetivos específicos:

Construir un menú principal que conste de las siguientes opciones: Registro de usuario, Casino DreamWorld, Configuración avanzada y Salir.

Escribir el código de los dos juegos con los que contará el casino; Black Jack y Tragamonedas.

Generar una opción de salida que guarde la información relevante del usuario.

# Descripción del problema

La empresa Global Casinos Inc. Busca desarrollar un software para el manejo de su próximo casino online llamado DreamWorld Casino. En ese sentido, el problema consiste en desarrollar la arquitectura y diseño de este software. El Casino debe contar con todas las medidas de seguridad que garanticen un manejo correcto del dinero de los usuarios, una plataforma de registro de usuarios, juegos en línea y acceso avanzado para la configuración del sistema.

El software debe contar con las siguientes características:

1. Menú Principal: Al ejecutar el programa, el usuario visualizará un menú principal que consta de cuatro opciones: a. Registro de usuario nuevo b. DreamWorld Casino c. Configuración Avanzada d. Salir
2. Registro de usuario nuevo: Para hacer uso del DreamWorld Casino, todo usuario debe someterse a un registro previo. Al seleccionar esta opción, el sistema guiará al usuario a través de los pasos para completar el registro : creación de nombre de usuario, escogencia del PIN, depósito obligatorio y, por último, debe guardar información y estructura de archivos: Una vez completado el registro del usuario, el sistema deberá guardar toda la información del nuevo usuario y generar la estructura de carpetas y archivos asociados a dicho usuario.

## Estructuras de software

1. Funciones
2. Estructuras cíclicas (ciclo “while”)
3. Estructuras de decisión (if, elif, else)
4. Comentarios

## Algoritmo

1. Primero, se importa el módulo getpass para permitir que los usuarios ingresen contraseñas de forma segura sin que se muestren en la pantalla.
2. A continuación, se definen cuatro listas vacías: listausuario, listacontrasena, listanombre y listadivisa. Estas listas se utilizarán para almacenar la información de los usuarios registrados.
3. Se define la función registrousuario() que se encargará de solicitar y registrar la información de un nuevo usuario.
4. Dentro de la función registrousuario(), se inicia un ciclo while donde se solicita al usuario que ingrese un nombre de usuario. Si la longitud del nombre de usuario es menor a 5 caracteres, se muestra un mensaje de error y se incrementa un contador. Si el contador llega a 3, se muestra un mensaje indicando que se ha alcanzado el límite de intentos y se finaliza la función.
5. Si el nombre de usuario tiene una longitud mayor o igual a 5 caracteres, se agrega a la lista listausuario.
6. A continuación, se solicita al usuario que ingrese su nombre y se agrega a la lista listanombre.
7. Luego, se inicia otro ciclo while donde se solicita al usuario que ingrese una contraseña utilizando la función getpass.getpass(). Si la longitud de la contraseña es menor a 6 caracteres, se muestra un mensaje de error. Si la longitud es mayor o igual a 6 caracteres, se agrega a la lista listacontrasena.
8. Después de registrar la contraseña, se inicializa la variable monto a 0 y se inicia otro ciclo while.
9. Dentro de este ciclo, se le muestra al usuario un menú para realizar un depósito. El usuario debe seleccionar una opción (1 para dólares, 2 para colones, 3 para bitcoin) y luego ingresar un monto para depositar en su cuenta.
10. Dependiendo de la opción seleccionada, se realiza una conversión de divisa y se actualiza el valor de la variable monto con el monto correspondiente en dólares.
11. Si el usuario selecciona una opción inválida, se muestra un mensaje de error.
12. Si el contador llega a 3, se muestra un mensaje indicando que se ha alcanzado el límite de intentos en el depósito y se eliminan los datos de usuario registrados previamente.
13. Una vez que se ha registrado el monto válido, se agrega a la lista listadivisa.
14. Finalmente, se muestra un mensaje indicando que la transacción se ha realizado satisfactoriamente y se muestra el saldo en dólares.
15. Después de definir la función registrousuario(), se inicia un ciclo while principal que muestra un menú con cuatro opciones al usuario.El usuario puede seleccionar una opción ingresando un número del 1 al 4.
16. Dependiendo de la opción seleccionada, se ejecutará una función correspondiente o se muestra un mensaje de error si la opción no es válida.

# Requerimientos del sistema

1. Sistema operativo: El programa debe ser compatible con sistemas operativos estándar como Windows, macOS y Linux.
2. Espacio de almacenamiento: El programa no requiere una gran cantidad de espacio de almacenamiento, ya que no implica una interfaz gráfica compleja. Sin embargo, se debe disponer de espacio suficiente para almacenar los archivos de datos asociados al casino y los usuarios.
3. Interfaz de línea de comandos: La interacción con el programa se realizará mediante una interfaz de línea de comandos, por lo que no se requerirá una interfaz gráfica. El usuario debe estar familiarizado con la navegación y ejecución de programas en la terminal o el símbolo del sistema.
4. Memoria RAM: El programa no requiere una gran cantidad de memoria RAM, pero se recomienda tener al menos 4 GB de RAM (lo cual es a memoria estándar) para un rendimiento óptimo.
5. Procesador: Se necesita un procesador compatible con las versiones de Python utilizadas.
6. Permiso de escritura y lectura: El usuario debe tener permisos de escritura y lectura en los directorios donde se almacenarán los archivos de datos asociados al casino y los usuarios.
7. Bibliotecas: El programa utiliza bibliotecas estándar de Python, por lo que no se requiere la instalación adicional de bibliotecas externas.

# Manual de usuario

# Bibliografía

Crawford, G., Gosling, V. K., y Light, B. (2013). The social and cultural significance of online gaming. *Gamers: The social and cultural significance of online games*, 1-22.

Schwartz, D. G. (2006). *Roll the bones: The history of gambling* (Vol. 494). New York: Gotham Books.