

Práctica 1 proyecto POO

Proyecto:

Casino Diamond

Materia:

Programación Orientada a Objetos

Grupo 02, Equipo 04

Integrantes:

Emanuel Palacio Perez

Juan Jose Gomez Grajales

Angie Melissa Montoya Urrego

Juan Diego Cardenas Acevedo

Profesor:

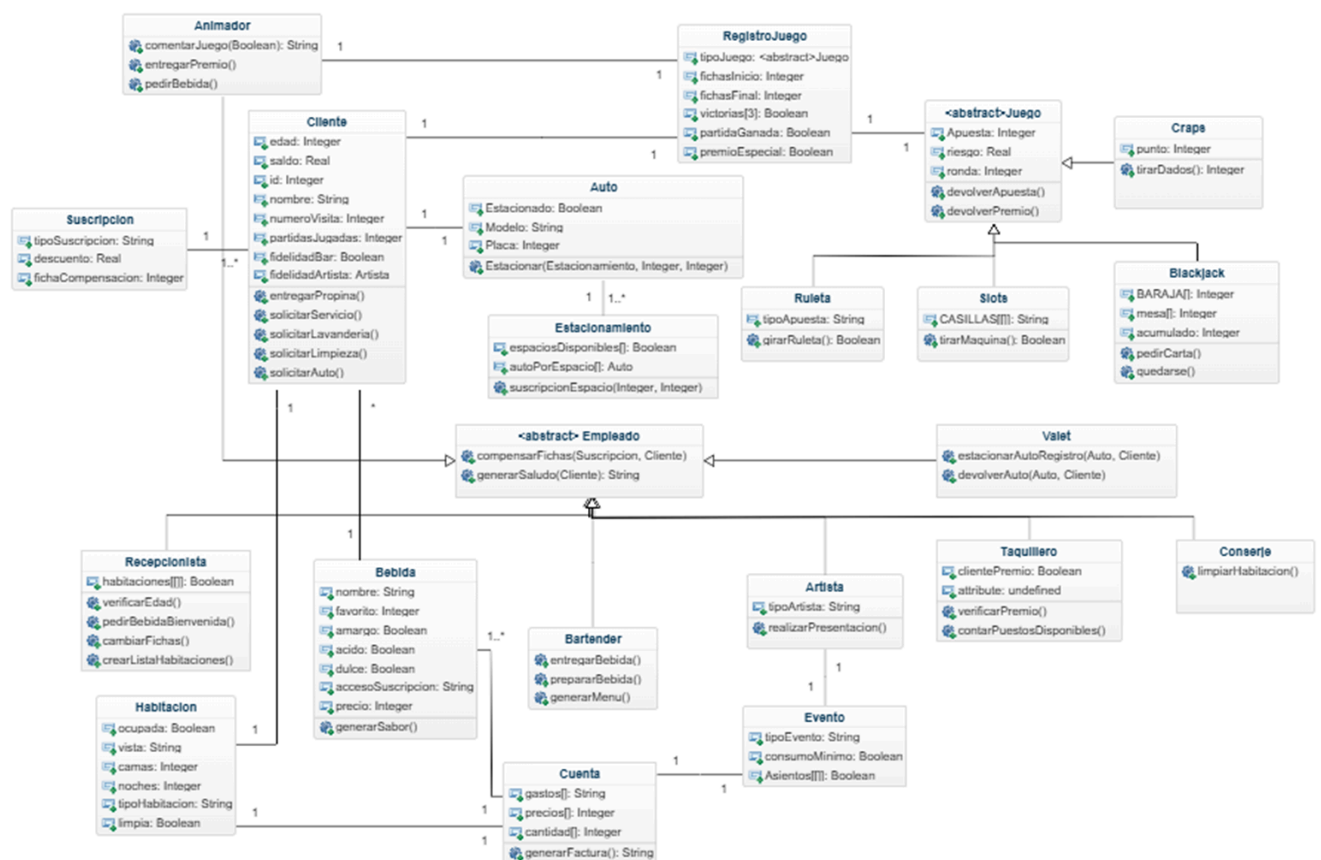
Jaime Alberto Guzmán Luna

Universidad Nacional de Colombia - Sede
Medellín

2024-2

El programa consiste en una abstracción de un casino, y se centra en el paso de un cliente por el mismo, utilizando sus atributos para crear una experiencia personalizada y recompensar ciertos comportamientos dentro del casino, de modo que según el cliente y su perfil es que giran las funcionalidades, perfil que contiene cosas como una suscripción, cuentas y un registro de sus estadísticas de juego. Para el uso de estos perfiles cada cliente tendrá que poner al iniciar una identificación, y se verificará si existe su perfil dentro del código para ser usado en el resto de la ejecución.

Descripción del diseño estático del sistema en la especificación UML.



Personal:

Animador: Entrega recompensas y maneja el registro de los juegos en la funcionalidad de juegos.

Artista: Hace parte de la funcionalidad de Eventos y sus atributos determinan el tipo de evento que se realiza.

Bartender: Maneja la preparación de bebidas de bienvenida en la funcionalidad de recepción, además de la generación de menú y la preparación de bebidas en su propia funcionalidad Bar.

Cliente: Sobre esta giran todas las funcionalidades, determina un perfil para el usuario que es usado en la ejecución del programa de forma activa.

Conserje: Maneja los servicios de lavandería y limpieza de habitación en las habitaciones (no implementado).

Empleado: Clase abstracta que sirve como molde para el resto de los empleados, define un método general para la compensación de fichas.

Recepcionista: Hace parte de la funcionalidad de recepción, se encarga de verificar clientes existentes y actualizarlos, o de crear registros de clientes nuevos, también se encarga del cambio de fichas y la funcionalidad de Habitaciones (no implementada).

Taquillero: Se encarga de la funcionalidad eventos en la reserva de asientos y opciones de eventos.

Valet: Maneja la interacción 1 de la funcionalidad de recepción, se encarga de estacionar y registrar el auto del cliente cuando primero llega al casino.

Servicios:

Juegos:

Blackjack: Genera una partida de blackjack (ver más en funcionalidad Juegos).

Craps: Genera una partida de craps (ver más en funcionalidad Juegos).

Juego: Clase abstracta de todos los juegos que define los atributos generales de riesgo y apuesta y métodos para devolver la apuesta y jugar una partida.

Ruleta: Genera una partida de ruleta (ver más en funcionalidad Juegos).

Slots: Genera una partida de slots (ver más en funcionalidad Juegos).

Asiento: Representa a los asientos que se reservan en la funcionalidad Eventos e implementa atributos para ver si está ocupado, su zona y su precio.

Auto: Representa al auto en el que llega el usuario al inicio del programa, usándose en la funcionalidad de recepción para estacionarlo y guardar su puesto.

Bebida: Usada en la funcionalidad Bar para representar cada bebida, contiene los ingredientes que la componen, que se usan junto con sus otros atributos para generar una descripción de sí misma.

Casino: Se utiliza para guardar espacios importantes del casino, como el estacionamiento, el hotel o el teatro.

Cuenta: Se utiliza cada vez que el cliente genera un gasto, consiste en la descripción y el precio de este.

Evento: Hace parte de la funcionalidad de eventos, maneja los tipos de eventos y el acceso a los asientos.

Habitación: Hace parte de la funcionalidad de habitaciones, maneja los atributos de la habitación escogida por el cliente.

Ingrediente: Compone a las bebidas y tiene calidad propia que depende de la suscripción del cliente.

RegistroJuego: Consiste en un registro específico que guarda cada partida jugada por el cliente en la funcionalidad Juegos.

Suscripción: Es principalmente un atributo de cliente, y es usado en todas las funcionalidades para determinar accesos especiales u otros beneficios.

Descripción de la implementación de características de programación orientada a objetos en el proyecto.

Clase abstracta:

```
gestorAplicacion > Servicios > juegos > Juego.java > Language Support for Java(TM) by Red Hat > Juego > setApuesta(int)
1  package gestorAplicacion.Servicios.juegos;
2
3  import gestorAplicacion.personal.Animador;
4  import gestorAplicacion.personal.Cliente;
5
6  public abstract class Juego {
7      // Atributos
8      int apuesta;
9      private float riesgo;
10
11     // Constructor
12     protected Juego(int apuesta, float riesgo) {
13         this.apuesta = apuesta;
14         this.riesgo = riesgo;
15     }
16
17     // Getters y Setters
18     public int getApuesta() {
19         return apuesta;
20     }
21
22     public void setApuesta(int apuesta) {
23         this.apuesta = apuesta;
24     }
25
26     public float getRiesgo() {
27         return riesgo;
28     }
29
30     public void setRiesgo(float riesgo) {
31         this.riesgo = riesgo;
32     }
33 }
```

Ubicación: gestorAplicacion.servicios.juegos, línea 6.

Uso: Se usa una clase abstracta de tipo juego para que la creación de las clases y objetos de Blackjack, Craps, Slots y Ruleta sean más sencillas.

Metodo abstracto:

```
// Metodo abstracto que desarrolla el juego
public abstract void jugar(Cliente cliente, Animador animador);
```

Ubicación: Clase Juego, gestorAplicacion.Servicios.juegos,Juego , Línea 46.

Uso: Se hizo pensando en que clase que representa cada juego iba a tener el método jugar, pero cada juego es distinto, por lo tanto se hizo el método abstracto para que se especifique en cada clase como jugar.

Herencia:

```

gestorAplicacion > personal > J Bartender.java > Language Support for Java(TM) by Red Hat > Bartender
1  package gestorAplicacion.personal;
2
3  import gestorAplicacion.Servicios.Bebida;
4  import gestorAplicacion.Servicios.Cuenta;
5  import gestorAplicacion.Servicios.Ingrediente;
6  import gestorAplicacion.Servicios.Suscripcion;
7  import java.io.Serializable;
8  import java.util.ArrayList;
9  import java.util.List;
10
11 public class Bartender extends Empleado implements Serializable {
12     private static final long serialVersionUID = 1L;
13     private static List<Bebida> barraDeBebidas;
14     private static List<Ingrediente> barraDeIngredientes;
15     private ArrayList<Bebida> menuActual;
16
17     public Bartender(String rol, String puesto, List<Bebida> barraDeBebidas, List<Ingrediente> barraDeIngredientes) {
18         super(rol, puesto);
19         this.barraDeBebidas = barraDeBebidas;
20         this.barraDeIngredientes = barraDeIngredientes;
21         this.menuActual = new ArrayList<>();
22     }
23
24     public Bartender(String rol, String puesto){
25         super(rol, puesto);
26     }
27
28     @Override
29     public String generarSaludo (String nombre, String rol){
30         return "Hola, " + nombre+ "soy un " + rol;
31     }
32

```

Ubicación: Clase Bartender, gestorAplicacion.personal.bartender. Línea 11.

Uso: La clase bartender se hereda de la clase empleado, para usar los métodos como generarSaludo, y tener acceso a sus atributos como rol y puesto.

Encapsulamiento:

```

public abstract class Empleado {
    private String rol;
    private String puesto;
    private static ArrayList<Empleado> empleados = new ArrayList<>();

    protected Empleado(String rol, String puesto) {
        this.rol = rol;
        this.puesto = puesto;
        empleados.add(this);
    }

    public void compensarFichas(Suscripcion suscripcion, Cliente cliente) {
        int fichasCompensacion = suscripcion.getFichaCompensacion(); // Obtiene la cantidad de fichas de compensacion
        cliente.setFichas(cliente.getFichas() + fichasCompensacion); // Actualiza las fichas del cliente
    }
}

```

Ubicación: Clase empleado, gestorAplicacion.personal.empleado líneas 9, 11, 17.

Uso: Debido a que la clase empleado está pensada para la herencia, el constructor de esta se encuentra encapsulado con el modificador de acceso protected.

Método estático:

```
public class MainJuegos {

    public static void funcionalidadJugar(Cliente usuarioActual) {
        Scanner scanner = new Scanner(System.in);
        Animador animador = new Animador();
    }
}
```

Ubicación: Clase MainJuegos, uiMain.MainJuegos, línea 14.

Uso: Se implementó este método para facilitar que el main quedara lo más limpio posible, haciendo uso de la clase mainjuegos en donde está toda la información necesaria para la funcionalidad jugar.

Atributos de clase:

```
public class Bartender extends Empleado implements Serializable {
    private static final long serialVersionUID = 1L;
    private static List<Bebida> barraDeBebidas;
    private static List<Ingrediente> barraDeIngredientes;
    private ArrayList<Bebida> menuActual;
}
```

Ubicación: Clase Bartender, gestorAplicacion.empleados.Bartender, línea 12-14.

Uso: Todos los objetos de tipo bartender aprovechan la misma barra de bebidas y de ingredientes, por lo tanto se hace uso de atributos estáticos para que sea igual en todos los objetos a menos de que se precise cambiarlo de manera estática

Constante:

```
public class Slots extends Juego {

    // Atributos
    private static final String[] SIMBOLOS = {"pica", "corazon", "trebol", "diamante"};

    // Constructor
    public Slots(int apuesta) {
        super(apuesta, riesgo:1.5f); // Establecemos una apuesta inicial y un riesgo básico
    }
}
```

Ubicación: Clase Slots, gestorAplicacion.Servicios.Juegos.Slots, línea 11.

Uso: Para facilitar el método Jugar de la clase slots, se precisaba de una lista simple que contuviera los símbolos del slot, en base a eso se crea un atributo con los símbolos y se hace constante ya que siempre para el método jugar y para cualquier objeto de tipo slot los símbolos serán los mismos.

Constructores:

```
public Cliente(String nombreCliente, int edadCliente, long id, float saldo) {
    this.nombreCliente = nombreCliente;
    this.edadCliente = edadCliente;
    this.id = id;
    this.saldo = saldo;
    this.suscripcion = new Suscripcion(numeroVisitas); //Se inicializa segun el numero de visitas

    / Constructor con parámetros
    public Cliente(String nombreCliente, int edadCliente, long id, float saldo, Auto auto, Suscripcion suscripcion) {
        this.nombreCliente = nombreCliente;
        this.edadCliente = edadCliente;
        this.id = id;
        this.saldo = saldo;
        this.auto = auto;
        this.suscripcion = suscripcion;
    }
}
```

Ubicación: Clase Cliente, gestorAplicacion.personal.Cliente, línea 35, 38, 46.

Uso: Se definen distintos constructores según la información, por ejemplo, para un cliente que por primera vez usa el casino, se debe hacer uso del cliente con todos los parámetros.

```
public class Bartender extends Empleado implements Serializable {
    private static final long serialVersionUID = 1L;
    private static List<Bebida> barraDeBebidas;
    private static List<Ingrediente> barraDeIngredientes;
    private ArrayList<Bebida> menuActual;

    public Bartender(String rol, String puesto, List<Bebida> barraDeBebidas, List<Ingrediente> barraDeIngredientes) {
        super(rol, puesto);
        this.barraDeBebidas = barraDeBebidas;
        this.barraDeIngredientes = barraDeIngredientes;
        this.menuActual = new ArrayList<>();
    }

    public Bartender(String rol, String puesto){
        super(rol, puesto);
    }
}
```

Ubicación: Clase Bartender, gestorAplicacion.empleados.Bartender, línea 17-24.

Uso: Se usan dos constructores para la clase bartender, dependiendo de si hay que hacer una modificación específica a la barra de bebidas y/o de ingredientes.

Enumeracion:

```
public class Asiento {

    public enum ZonaAsiento {

        Palco,
        Balcon,
        Centro,
        Atras;
    }

    private ZonaAsiento zona;
    private int cantidad;
    private boolean reservado;
    private Double precio;
}
```

Ubicación: clase Asiento, gestionAplicacion.Servicios.Asiento, línea 5.

Uso: Para facilitar el manejo del atributo ZonaAsiento, se hizo la elección de usar enumeradores.

Descripción de cada una de las 5 funcionalidades implementadas.

Descripción de la funcionalidad: Proporcione una descripción detallada y completa del comportamiento de cada funcionalidad. Asegúrese de explicar claramente cada una de las interacciones que componen la funcionalidad, describiendo cómo los diferentes componentes del sistema colaboran para lograr el objetivo de la funcionalidad.

Diagrama de interacción: Incluya una imagen del diagrama de interacción correspondiente a cada funcionalidad. Asegúrese de que el diagrama sea claro y permite observar el comportamiento general de la funcionalidad, destacando las interacciones clave y los flujos de datos entre los componentes del sistema.

Funcionalidad 1: recepción y cambio de fichas (Objetos *Cliente*, *valet*, *auto*, *estacionamiento*, *bebida*, *Bartender*, *casino*, *repcionista*, y *suscripción*):

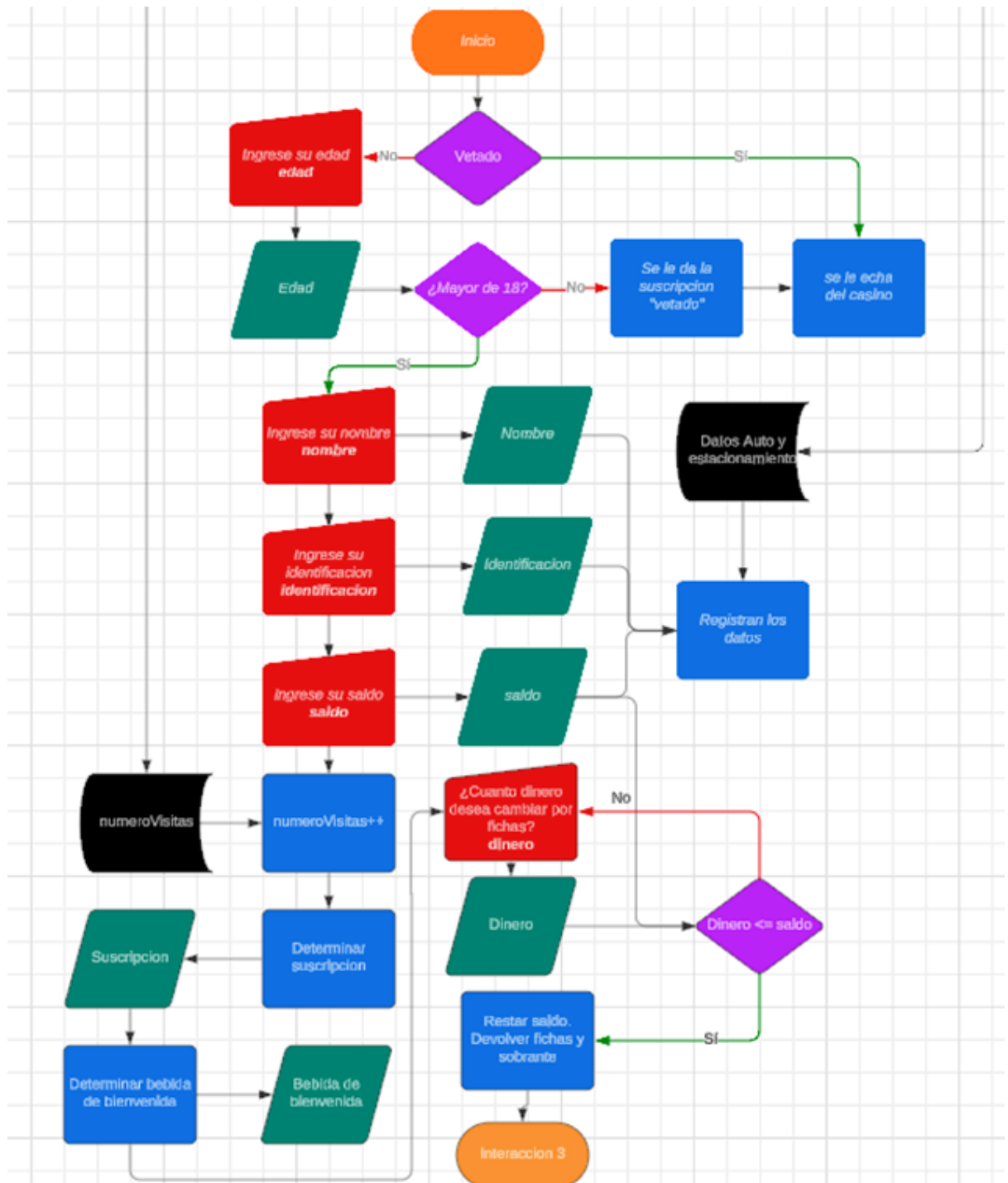
Interacción 1:

El cliente pasa por la entrada en su auto, el valet le pide el modelo y la placa de su auto, luego, le pide su identificación para revisar si el cliente tiene un registro en el casino y revisar su suscripción, y según esta verifica que haya espacios disponibles en el estacionamiento, que tendrá las dos primeras filas exclusivas para los clientes con suscripción platinum, le muestra las opciones al cliente en una matriz con los espacios disponibles señalados, y le pide escoger el espacio que va a usar indicando la columna y la fila correspondiente, si el espacio no está disponible se le pide repetir el proceso una vez más, si el espacio escogido corresponde a uno de suscripción menor a la señalada por el cliente, el valet le entrega fichas al cliente por compensación correspondiente a su suscripción, entonces lo lleva al espacio de estacionamiento escogido, y este es ocupado. Finalmente, entrega la información del auto, que incluye su espacio en el estacionamiento, para ser usado en el registro de la funcionalidad siguiente (valet(empleador), suscripción, cliente, auto, estacionamiento)



Interacción 2:

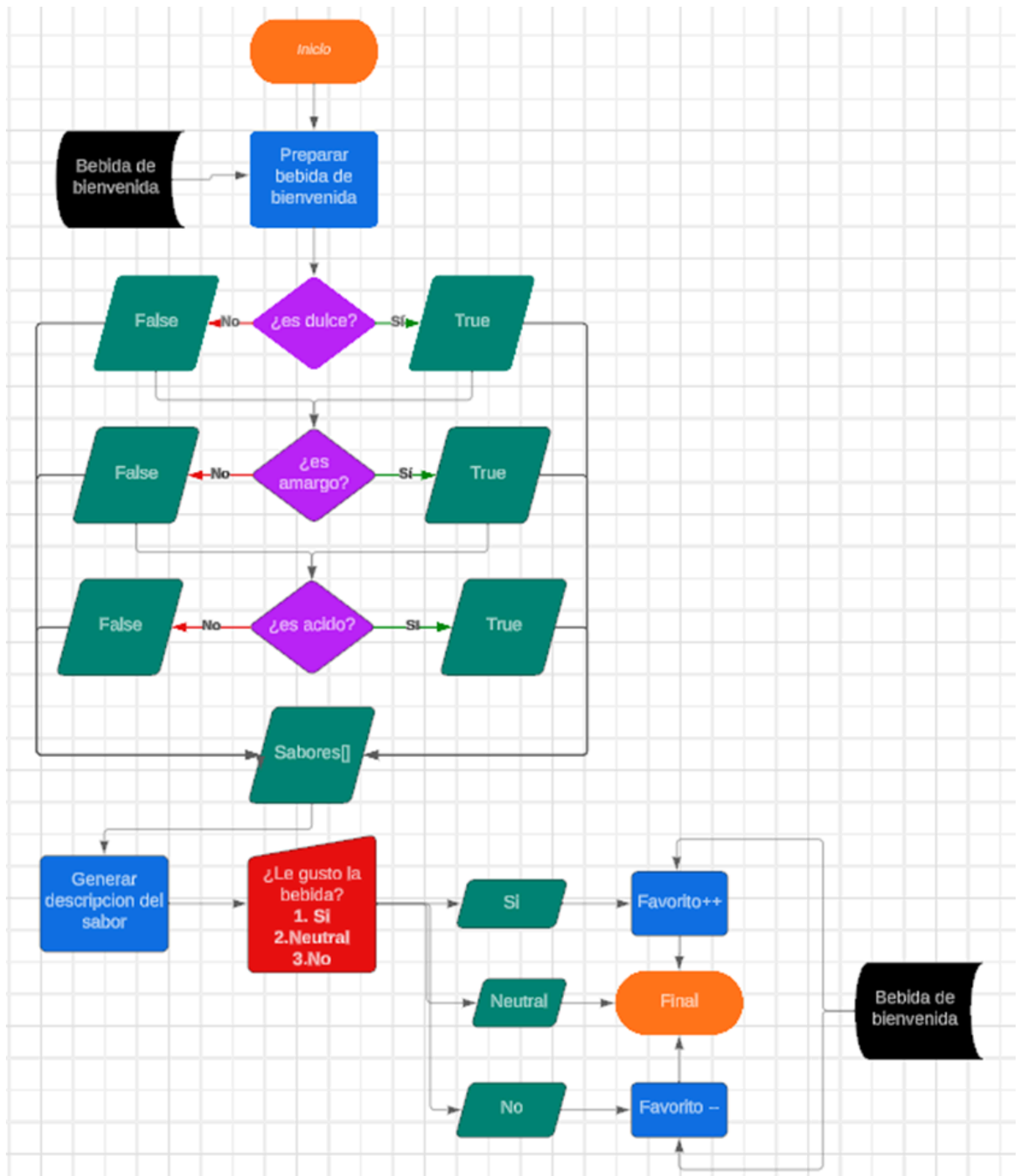
El cliente entra al casino y se le pide la siguiente información: edad(int), saldo(int), identificación(long) y nombre(string). El recepcionista verificará que la edad del cliente sea mayor a 18 para dejarlo entrar, si no lo es, quedará vetado, de lo contrario, la información que ingrese se usará junto con la información de su auto para crear su registro, si la identificación y nombre dados están ya registrados no se vuelven a guardar, solo se actualiza el saldo, luego, se aumenta en uno y se verifica su atributo de “numeroVisitas”, según este número se actualizará su suscripción(los tipos de suscripción son: “vetado”(caso especial), “Por defecto”(2 a 3), “Primera visita”(1), “silver”(3 a 9), “Platinum”(10 o mayor))se verifica si el cliente está vetado, en cuyo caso se echa del casino, de lo contrario continuamos, según esto se creará y llamara al objeto correspondiente de *suscripción* para más adelante determinar opciones de descuentos, regalos y opciones, en particular la bebida de bienvenida que se dará en la fase siguiente. Se preguntará al cliente la cantidad de dinero que desea cambiar por fichas, siendo que cada ficha tenga un precio de 1000 pesos, y se le devolvera el número de fichas correspondiente más el sobrante en pesos, todo esto teniendo en cuenta que el saldo del cliente sea mayor o igual al número de fichas que desee cambiar. Entrega la información del cliente, junto con su suscripción.(Suscripción, Cliente, Recepcionista)



Interacción 3:

Se llama al objeto bartender para que traiga una bebida de bienvenida, que dependerá de su atributo de bebida favorita si lo tiene, de otra forma, de la bebida del bar que tenga el mayor valor de favorito, que será un atributo que determine la popularidad de la bebida; su calidad dependerá de la instancia de suscripción que tenga el cliente. Se describirá el sabor de la bebida en consola, que será un string que se generará según los atributos de la misma, sean sus sabores e ingredientes, finalmente, se le preguntará

al cliente si le gustó, en una encuesta con 3 opciones: una negativa, una neutral y una positiva, según este resultado el atributo favorito de la bebida disminuye en 1, se queda igual o aumenta en uno respectivamente.(Bartender, Recepcionista, bebida, Cliente).



Funcionalidad 2: Jugar (objetos juego, cliente, animador(emplead), taquillero(emplead), bartender y subscripción):

Interacción 1:

Primero, el animador saluda al cliente y determina su suscripción, según esta se le adiciona un monto de fichas adicionales, se genera entonces un registro de juego si no lo tiene aún, para guardar información detallada de cada cliente sobre sus fichas y partidas jugadas. Luego, se le da a escoger al cliente de una serie de juegos (ruleta, craps, slots, blackjack), se aumenta la cantidad de partidas jugadas del cliente en 1 en su registro de juego y se asigna el juego. Se escogen los parámetros de los juegos, sean una apuesta en fichas y un riesgo proporcional de cada juego los cuales son atributos de cada objeto nuevo de los que se heredan de la clase objeto. Se devuelve la apuesta, que será usada para determinar las ganancias en la siguiente interacción. (Cliente, Animador, Juego(abstrata), Suscripción, registroJuego).

❖ **Ruleta:**

Riesgo: x2 color, x35 número

Desarrollo: Se le pide al cliente escoger el tipo de apuesta, escogerá entre color o número, y el monto a apostar. Luego, se girará la ruleta para ver en qué número caerá, si es necesario se determina el color de la casilla según el número sea par o impar. Hay un caso especial si la ruleta escoge el número 0 o la “casilla verde”, en este caso gana la apuesta y se le da el premio especial.

❖ **Craps:**

Riesgo: x2

Desarrollo: Se le pide al cliente tirar dos dados de seis caras y se anota la suma de los resultados, si esta suma es un 7 o un 11, se considera un “natural” y gana la apuesta; si es 2, 3 o 12, se llama un “crapping out” y pierde la apuesta; si no es ninguno de estos números, el número en el que caiga se le llama el “punto”, luego, se aumenta el riesgo en un 0.5 y aplican nuevas reglas, debe tirar los dados hasta sacar el “punto”, en cuyo caso gana la apuesta, o un 7, también llamado “siete fuera”, en cuyo caso pierde la apuesta. si saca un 7 a la primera tirada, gana la apuesta y se le da el premio especial.

❖ **Slots:**

Riesgo: x1.5 (espadas), x2 (corazones), x3 (tréboles), x8 (diamantes)

Desarrollo: El cliente hace una apuesta y tira de la máquina de slots, que escoge aleatoriamente elementos de 3 ruedas dentro de la máquina, si salen los 3 símbolos iguales, gana la apuesta con el riesgo correspondiente. si gana con los diamantes, se le da el premio especial.

Blackjack:

Riesgo: x2 (apuesta base), x2.5 (Blackjack)

Desarrollo: El jugador apuesta un monto inicial y recibe dos cartas. Puede pedir cartas adicionales, quedarse con su mano, duplicar la apuesta o dividir si tiene pares. El objetivo es sumar 21 o acercarse más que el crupier sin pasarse. El crupier sigue reglas fijas para pedir cartas. Si el jugador gana, recibe el

doble de su apuesta; un "blackjack" (21 con las dos primeras cartas) paga 1.5 veces la apuesta. Si llega justo a las 21 se le entrega el premio especial.

Interacción 2:

El desarrollo de estos juegos será comentado por el animador, que mostrará distintos mensajes según el cliente esté ganando o perdiendo, teniendo en cuenta cosas como racha de victorias o racha de pérdidas, ejemplo, si el cliente gana 3 partidas seguidas de cualquier juego, el animador anunciará con el nombre del cliente y la frase “está en racha”, si gana, obtiene las fichas que apostó multiplicadas por el riesgo, mas si pierde no las recupera, hay una interacción especial si el cliente gana además el premio especial, indicado en el desarrollo de cada juego, será felicitado por el animador y recibirá 100 fichas como premio. Se devuelve el número de rondas ganadas y la cantidad actual de fichas para generar el registro de juego. (Animador, Cliente, Juego(abs), Suscripción).

Interacción 3:

Según el resultado del juego, aumenta la cantidad en sus juegos ganados(si llega a 15 se sube su suscripción a “Ganador Usual”, en este caso, el animador anuncia al cliente que su suscripción a subido a ganador usual), si pierde se le da una cantidad pequeña fija de fichas como consolación. Si el cliente tiene 10 o más partidas jugadas y un promedio de victorias del 100%, se le echa del casino y se le da la suscripción “vetado”. Finalmente, se genera y muestra su registro de juego hasta el momento, que contiene las fichas con las que entró y salió, la cantidad de partidas que ganó, además de su tasa de victorias y número de partidas jugadas.(cliente, animador, bartender, suscripción).

Funcionalidad 3: Bar(objetos Bar, cliente, Bartender(Empleado), bebida, recomendación, y subscripción):

Interacción 1:

El bartender usa el objeto de suscripción del cliente, le da una bienvenida personalizada e informa sobre posibles descuentos a los que puede aplicar según su suscripción. Adicionalmente, se realiza una encuesta pequeña para conocer sus preferencias, primero, se le da la opción de bebidas alcohólicas y no alcohólicas, luego se le pregunta uno a uno si le gustan los sabores dulces, amargos o ácidos, bebidas que tengan esos sabores serán restringidas del menú, que será generado justo después, este tendrá en cuenta las especificaciones escogidas por el cliente y cada bebida estará acompañada por una recomendación(poco recomendado(-), neutral(N) y muy recomendado(+)) que será calculada según el atributo favorito de cada bebida; tendrá en cuenta además la bebida favorita del cliente si es que la tiene. Devuelve el menú que será mostrado para la selección del cliente(bartender, suscripción, cliente, bebida).

Interacción 2:

Se le muestra el menú generado acompañado por los descuentos asociados a la suscripción y a la fidelidad del bar si la tiene(ver siguiente interacción), una vez el cliente toma su decisión, el bartender prepara su bebida, buscando cada uno de los ingredientes para prepararla y determinando su calidad según la suscripción del cliente. Adicionalmente, si es la tercera vez que el cliente pide la misma bebida, se guardará esta bebida como su bebida favorita. Al final, el bartender entrega al cliente la bebida, se genera una cadena de texto que describe la bebida y su sabor según sus atributos(Dulce, Amargo y/o Ácido), además de los ingredientes que se usaron para prepararla. Devuelve la bebida preparada(bartender, cliente, suscripción, bar, ingrediente).

Interacción 3:

Después de que el cliente haya consumido, se genera la factura de pago, esta muestra la bebida pedida por el cliente, su precio en bruto, descuentos aplicables por suscripción y fidelidad, y finalmente el valor a pagar, por último, le da la opción de dar una propina al bartender, los clientes que hayan pedido al menos 3 ítems y en más de la mitad de los pedidos hayan dejado propina se les dará un descuento especial mientras mantengan este estado. La cuenta se guarda en el perfil del cliente y es utilizada para determinar su bebida favorita si la tiene, se pagan una vez decide salir del casino(bartender, bar, cliente, bebida, cuenta).

Funcionalidad 4: Reservar habitación(objetos Recepcion, Cliente, Recepcionista(Empleado), habitación, y suscripción): (no implementado en código)

Interacción 1:

El recepcionista llama al objeto de suscripción del cliente para determinar el descuento fijo y opciones extra de habitaciones. Se presenta una lista con las habitaciones disponibles del casino clasificadas por tipo (estándar, presidencial y suite), vista y camas, esta lista de las habitaciones se muestra en consola teniendo en cuenta estos parámetros. Luego, el cliente selecciona la habitación y el número de noches que se queda, esta información se guarda en su cuenta, que será usada para generar el recibo de pago al final, se le entrega la llave de su habitación, que será usada para que solo él o ella misma pueda entrar a esta. Se le pregunta cuántas maletas lleva, que será considerado como su equipaje, y según este número se determinará su peso. En caso de que supere los 20 kilos, se guardará un costo adicional a su cuenta. Además, puede elegir si quiere o no que su equipaje sea llevado directamente a su habitación si no desea cargarlo, lo que conlleva otro gasto que dependerá del número de maletas. (suscripción, habitación, cliente, recepcionista, cuenta, casino) Corregir interacción objetos

Interacción 2:

Se muestra al cliente las habitaciones del piso, y debe escoger en cual entrar, pero solo podrá acceder a aquella la cual tenga su llave correspondiente, y en esta puede hacer uso de los servicios disponibles según el tipo de habitación que escogió, sea servicio a la habitación, limpieza, lavandería o llamada para preparar su automóvil. Cada uno de estos servicios tendrá un costo que se recargará a la cuenta de la habitación. (conserje(Empleado), bartender, valet, cliente, recepción, cuenta).

❖ **Servicio a la habitación:**

El cliente llama directamente al bar y se ejecuta parte de la funcionalidad bar, pero no guarda los gastos en la cuenta del bar sino en la cuenta de la habitación.

❖ **Limpieza:**

Se llama al conserje y se le pide limpiar la habitación, si la habitación está limpia, se le hará saber al cliente y se cancela el servicio, de lo contrario, se ejecuta normalmente, este estado sucio o limpio se maneja en función de que, cada dos noches que pase la habitación ocupada después de ser limpiada, tendrá el estado sucio, y cada que se realice la limpieza vuelve a estar limpia.

❖ **Lavandería:**

El cliente puede llamar y pedir en cualquier momento el servicio de lavandería, en este, puede entregar el número de prendas que indiqué, que serán llevadas para ser lavadas y se recargara un gasto por cada prenda a su cuenta. Desde el día siguiente que se haga la lavandería se devolverán sus prendas a la habitación o a la recepción si es el día del checkout.

❖ **Pedir el carro para salir:**

Una vez el cliente hace uso de este servicio se asume que va a hacer su checkout de la habitación y además va a salir del casino, se llama al valet para sacar al carro del estacionamiento y se libera el espacio que ocupaba en el mismo, se lleva al cliente a la recepción y se hace el checkout de manera normal sin la parte de sacar el carro.

Interacción 3:

Finalmente, una vez el cliente decida hacer el checkout o se acabe su estadía, volverá a la recepción y se realizará un cálculo de sus gastos, el total parcial de estos se almacena en la cuenta, y se generará un recibo de pago con un descuento aplicado según su suscripción, los clientes con suscripción “primera vez” o menor, deben pagar inmediatamente o serán vetados del casino, los clientes de las suscripciones superiores tendrán plazo para realizar el pago hasta que decidan salir del casino, y podrán realizar el resto de funcionalidades normalmente. Si no puede pagar se le vetará del casino.(cuenta, cliente, recepción, habitación)

Funcionalidad 5: Eventos(objetos taquilla, asientos, auditorio, recepcionista(Empleado), cliente, espectáculo, artista y suscripción):

Interacción 1:

El recepcionista da la bienvenida al cliente y se le hace saber sobre los eventos disponibles, escoge un evento, se verifica su suscripción, si es platinum, se le regalará un asiento de palco y una bebida gratis especial en el evento que elija. Según la suscripción del cliente y el evento elegido se aplica un descuento distinto al precio de la taquilla, después de que escoja un evento, se le da la opción de escoger su asiento, las opciones son: primera fila, balcón, centro y atrás. Con disponibilidad descendente en ese orden según suscripción, se muestran únicamente la zona y el número de asientos disponibles. Si un cliente escoge un asiento menor al mayor al que podría acceder con su suscripción, se le regalan fichas de compensación. Devuelve el asiento en el que está el cliente. (asiento, casino, cliente, recepcionista)

Interacción 2:

Al cliente se le asigna el asiento según la localización que escogió, clientes con suscripción platinum tienen el beneficio de que independiente de la localización que pague en un evento, al iniciar este recibirá de parte del bar una bebida especial como obsequio para el disfrute del show, en contraste, clientes que tengan como suscripción “Primera vez” tendrán un consumo mínimo en los eventos, por lo tanto estarán obligados a comprar una bebida del bar antes de empezar el evento. El cliente es llamado al escenario y se le da la opción de decidir el tema del espectáculo (tipo de chiste, truco de magia) que variará según su elección. Empieza el evento, dependiendo del evento y tema escogido por el cliente el desarrollo es diferente, por ejemplo, dado el caso de que el cliente escoja un espectáculo de stand up (comedia) en consola se mostrará un chiste, dado el caso de que escoja un espectáculo de magia en consola se narrará el acto del mago, lo mismo con conciertos y obras de teatro. (Cliente, Bebida, Bartender, Espectáculo//Artista//Asiento)

Interacción 3:

Se genera la cuenta teniendo en cuenta el precio de taquilla y bebidas que se hayan pedido si aplican, posterior a esto el recepcionista sugiere una propina a los artistas implicados en el evento, en caso de aceptar se selecciona el monto y el administrador anuncia tras conocer el monto: “El clienteX ha hecho una donación de “monto de la donación” al artistaY”, cada cliente que haya atendido a al menos a tres eventos del mismo artista y haya dejado propina en más de la mitad, tendrá un estatus especial y su identificación será guardada en la taquilla, con esto, podrá acceder a los eventos de ese artista completamente gratis, siempre que mantenga este estado, finalmente finaliza la interacción con un agradecimiento del artista y se indica si aplica a esta lista de fidelidad. (Cliente, Recepción, Artista, Administrador)

6. Manual de usuario.

CASINO DIAMOND.

Manual de usuario.

Índice:

1. Comienzo y recepción.
 - 1.1. Perfiles inicializados.
2. Juegos; como jugar y demás interacciones.
 - 2.1 Suscripciones.
 - 2.2 Opciones de juego.
 - 2.3 Instrucciones para cada juego.
 - 2.4 Consultar estadísticas.
 - 2.5 Consejos útiles.
3. Bar, bebidas.
4. Habitaciones(N/A)
5. Eventos, artistas y demás.

1. Comienzo y recepción

Se le da la bienvenida al casino, en la abstracción, se simula que el cliente llega en un auto y es recibido por un valet, se le pregunta por su ID para ver si está registrado en el casino, luego, se le pide el modelo y placa de su auto. Se le muestra una matriz que representa el estacionamiento del casino, una vez escoja una columna y fila accesibles es estacionado su auto.

El cliente entra el casino, se le pregunta nuevamente por su ID por las mismas razones, se le pregunta su nombre, edad y saldo, no se le permite entrar al casino si es menor de 18 años, no se le pregunta su nombre si se encuentra su registro. Posteriormente, se le pide ingresar el monto de dinero que desee cambiar por fichas, que son usadas en los juegos más adelante, continúa hasta recibir una cantidad válida.

Finalmente, se le otorga una bebida de bienvenida al casino, que es escogida de entre todas por cuál es la más recomendada o cual es la favorita del cliente si aparece en su registro. Se le hace una encuesta para evaluar la bebida, lo que modifica su atributo de favorito.

1.1. Perfiles inicializados

En función de que se puedan probar las variaciones de las funcionalidades según distintos perfiles de clientes, se inicializan los siguientes:

```
public static void InicializarClientes() {  
    // Inicialización de clientes  
    Cliente clientePlatinum = new Cliente(nombreCliente:"james", id:45, saldo:1000000, new Suscripcion(visitas:6));  
    Cliente clienteSilver = new Cliente(nombreCliente:"samanta", id:37, saldo:1000000, new Suscripcion(visitas:4));  
    Cliente clientePorDefecto = new Cliente(nombreCliente:"jose", id:29, saldo:1000000, new Suscripcion(visitas:2));  
    clienteSilver.setFidelidadBar(fidelidadBar:true);  
}
```

para acceder a cada uno solo basta poner la id correspondiente cuando la pide el valet y el recepcionista. Se ejecuta entonces el resto del programa según el perfil de cliente seleccionado. Las características de cada uno son:

james: Suscripción Platinum

samanta: Suscripción Platinum, fidelidad Bar

jose: Suscripción Platinum

2. Juegos; como jugar y demás interacciones.

Justo después de la recepción, cuando el cliente se encuentre registrado, puede tomar la opción de Jugar (opción 1 en el menú) acto siguiente se dispondrán de los 4 juegos disponibles en el casino, los cuales son: Blackjack, Craps, Slots y Ruleta.

Antes de escoger el juego dependiendo de la suscripción activa del cliente se le regalará una cantidad de fichas pequeña para que juegue.

2.1 Suscripciones.

El nivel de tu suscripción determina la cantidad de fichas que se te otorgan antes de comenzar, en el siguiente orden:

Primera vez: este tipo de suscripción recibe 20 fichas antes de jugar, es una suscripción exclusiva de la primera visita al casino del cliente.

Por defecto: la obtención de esta suscripción se da al visitar el casino de 1 a 3 veces, recibe 10 fichas antes de jugar.

Silver: Si has visitado el casino de 3 a 5 veces tu suscripción es Silver, recibe 50 fichas antes de jugar.

Platinum: Obtenida visitando el casino más de 5 veces, recibe 100 fichas antes de jugar.

2.2 Opciones de juego.

Al ingresar a la sección de juegos, podrás elegir entre las siguientes opciones:

- 1. Blackjack
- 2. Ruleta
- 3. Craps
- 4. Slots

Introduce el número correspondiente al juego que deseas jugar.

2.3 Instrucciones para cada juego

Blackjack

1. Descripción:
En este clásico juego de cartas, tu objetivo es alcanzar una puntuación lo más cercana posible a 21 sin pasarte. Compites contra el crupier.
 2. Cómo jugar:
 - Después de seleccionar Blackjack, ingresa la cantidad de fichas que deseas apostar.
 - Se te repartirán dos cartas iniciales. Decide si deseas pedir más cartas ("Hit" o "Pedir") o quedarte con tu mano actual ("Stand" o "Plantarse").
 - Si tu puntuación supera la del crupier sin exceder 21, ¡ganas!
 3. Resultado:
Las fichas ganadas o perdidas se actualizan automáticamente en tu cuenta, siendo en este caso el doble de lo apostado, en caso tal de que tengas un Blackjack (21 en tus 2 primeras cartas) tus fichas se multiplican por 2.5.
-

Ruleta

1. Descripción:
Apuesta a un número o color, y gira la ruleta. Si la bola cae en tu elección, ¡ganas!
 2. Cómo jugar:
 - Selecciona Ruleta y define tu apuesta.
 - Elige tu tipo de apuesta: número exacto, color (rojo/negro).
 - La ruleta girará y mostrará el resultado.
 3. Resultado:
Si aciertas, ganas fichas según el tipo de apuesta, si ganas escogiendo color, tu apuesta se multiplica por 2, si ganas escogiendo número tu apuesta se ve multiplicada por 35.
-

Craps

1. Descripción:
Un emocionante juego de dados donde se apuesta al resultado de los lanzamientos.
2. Cómo jugar:
 - Selecciona Craps e ingresa la cantidad de fichas que deseas apostar.
 - Lanza los dados. Dependiendo del resultado, ganarás, perderás o continuarás lanzando:
 - 7 u 11 en el primer lanzamiento: ganas.
 - 2, 3 o 12 en el primer lanzamiento: pierdes.
 - Cualquier otro número se convierte en tu "punto", y sigues lanzando hasta igualarlo o sacar un 7.

3. Resultado:
Las fichas se ajustan automáticamente según el resultado, multiplicándose por 2 en caso de ganar.
-

Slots

1. Descripción:
Las clásicas tragamonedas donde buscas alinear símbolos iguales en los carretes.
 2. Cómo jugar:
 - Selecciona Slots y define tu apuesta.
 - Los carretes girarán automáticamente. Si alineas tres símbolos iguales en una línea, ganas según el valor de los símbolos.
 3. Premio especial:
Si consigues alinear tres diamantes, obtendrás el premio mayor el cual multiplica tu apuesta por 8, picas por 1.5, corazones por 2, trebol por 3.
-

2.4 Consultar estadísticas.

Puedes consultar tus estadísticas de juego seleccionando la opción correspondiente (opción 5) en el menú principal:

- Número de partidas jugadas.
 - Partidas ganadas.
 - Porcentaje de victorias.
 - Racha de victorias actual.
 - Fichas iniciales y fichas actuales.
-

2.5 Consejos útiles.

1. Asegúrate de revisar las reglas específicas de cada juego antes de comenzar.
2. Gestiona tus fichas sabiamente para maximizar tus posibilidades de ganar.
3. ¡Disfruta y juega responsablemente!

3. Bar y bebidas

Después de pasar por la recepción, si se escoge la opción 2, será dirigido al bar.

Una vez en el bar, se le hará una encuesta pequeña acerca de sus preferencias, uno a uno, se le preguntará si quiere que su bebida sea: alcohólica, ácida, amarga y/o dulce. Según esto se generará y se le mostrará un menú personalizado, por cada “no” que indique en la encuesta, se excluirá del menú a cada bebida con esa característica. la barra es algo reducida, entonces le pedimos que tome su decisión teniendo en cuenta estas 5 bebidas:

Barra	Mojito	Piña Colada	whisky Sour	Cerveza Artesanal	Martini
Dulce	Si	Si	Si	No	No
Amargo	No	Si	No	No	No
Ácido	No	No	Si	No	Si
Alcohólico	No	Si	Si	Si	Si

Una vez se genera el menú, se le pide al cliente que escoja una bebida de entre las opciones, entonces el bartender prepara su bebida y se le muestra una descripción de esta con su nombre, sabores, ingredientes y su calidad.

Al final, se le pregunta si quiere dejar propina, recordemos de la funcionalidad del bar que una vez el mismo cliente haya dejado 3 veces propina, se le dará un descuento en el bar por fidelidad. Entonces se crea y se muestra una factura al cliente que contiene su pedido, su precio, descuentos a los que aplica por suscripción o fidelidad, y el precio neto finalmente.

5. Área Eventos

Después de pasar por recepción, cuando el cliente se encuentra registrado, puede optar por ingresar al área de eventos [Opcion 3 del menú]

En esta selección, los clientes tienen la oportunidad de elegir entre distintos espectáculos programados por el casino y personalizar su experiencia, eligiendo el asiento según su preferencia y nivel de suscripción, además la suscripción activa determina descuentos especiales en los costos de los eventos y beneficios adicionales

5.1 Suscripciones

El tipo de suscripción del cliente afecta el precio final del evento y los beneficios adicionales que recibe:

Primera vez: Clientes con esta suscripción reciben un descuento del 5% en el precio del evento. Es una suscripción exclusiva para clientes que visitan el casino por primera vez.

Por defecto: Para clientes que han visitado el casino entre 1 y 3 veces. No se aplica descuento en el evento.

Silver: Clientes que han visitado el casino entre 3 y 5 veces reciben un descuento del 15% en el precio del evento.

Platinum: Clientes con más de 5 visitas al casino obtienen un descuento del 25% en el precio del evento y beneficios exclusivos, como una bebida especial gratuita y acceso prioritario a asientos de mayor categoría (Primera Fila o Palco).

5.3 Proceso de selección

Primero: Se muestran los eventos disponibles.

Segundo: Aplicación de descuento.

Después de seleccionar el evento, se calcula el costo final con el descuento según la suscripción del cliente.

El saldo del cliente se actualiza automáticamente al descontar el costo final del evento.

Tercero: Selección de asiento

El cliente elige una zona de asiento. Dependiendo de su suscripción, se mostrarán las opciones disponibles:

- **Zonas de asiento:**
 - Primera Fila
 - Balcón
 - Centro
 - Atrás

Ejemplo de selección:

- Si el cliente tiene una suscripción Platinum, el sistema asigna automáticamente "Primera Fila" y otorga fichas de compensación si opta por una zona de menor categoría.
- Para otras suscripciones, se despliegan las zonas disponibles y el cliente selecciona manualmente.

5.4 Resumen de reserva

Una vez completada la interacción, se imprime un resumen de la reserva:

5.5 Consejos útiles

- Revisa la lista de eventos y elige según tus preferencias.
- Aprovecha los descuentos y beneficios adicionales según tu nivel de suscripción.
- Recuerda elegir tu asiento con cuidado, ya que impacta tu experiencia en el espectáculo.
- ¡Disfruta de los eventos y vive una experiencia única en el casino!