

# Fundamentos de Programación Orientada a Objetos

Proyecto de curso

Entrega II

Juan Pablo Lozano Restrepo – 25215052724

Desarrollo de Software

Universidad del Valle

## Tabla de requerimientos

identificación del requerimiento	descripción	Entradas	Resultados (salidas)
R1	El sistema deberá permitir crear y gestionar un subsistema de baraja de cartas.	Solicitud de creación del mazo	Mazo inicializado con las cartas requeridas (52 por defecto, modificable).
R2	El sistema debe permitir repartir cartas al jugador y al crupier.	Solicitud de carta por parte del jugador o del crupier.	Carta entregada al usuario elegido y disminución del mazo
R3	El jugador debe poder solicitar cartas adicionales o plantarse.	Elección del jugador (pedir carta o plantarse).	Mano actualizada del jugador o paso de turno al crupier.
R4	El crupier debe jugar su turno automáticamente siguiendo las reglas del juego.	Fin del turno del jugador.	Mano final del crupier calculada según las reglas (mínimo 17 puntos).
R5	El sistema debe calcular el valor total de las manos considerando el valor de los Ases (1 u 11).	Cartas actuales de una mano.	Valor numérico de la mano.
R6	El sistema debe determinar el ganador de la ronda.	Valores finales de las manos del jugador y del crupier.	Resultado esperado: gana jugador o gana crupier o empate.
R7	El sistema debe detectar un blackjack en las dos primeras cartas.	Dos primeras cartas del jugador.	Resultado: "Blackjack" o valor normal.
R8	El sistema debe iniciar una nueva partida con el mismo mazo o uno nuevo.	Solicitud de reinicio del juego.	Mano reiniciada y mazo actualizado.

## Tabla de responsabilidades

Nombre del método	Requerimiento asociado	Precondición	Postcondición	Modelo verbal
Carta::toString	R1	Se desconoce el valor y tipo de carta	Se conoce el valor y tipo de carta	Retorna un texto con el valor y tipo de carta.
Crupier::jugarTurno	R2, R4	El crupier requiere obtener cartas si su valor es menor a 17	El crupier obtiene las cartas necesarias hasta que su valor sea mayor o igual a 17	El crupier solicita cartas hasta que su valor en baraja sea $\geq 17$
Crupier::calcularValorMano	R1, R5	Se desconoce el valor total de la mano del crupier	Se conoce el valor total de la mano del crupier	Retorna un entero con el valor total de la mano del crupier
Crupier::mostrarMano	R1	Se desconoce la mano que tiene el crupier	Se conoce la mano que tiene el crupier	Si hay cartas en la mano, retorna un texto mostrando cada carta
Crupier::limpiarMano	R1	Colección original con cartas definidas	Se limpia la colección	Permite limpiar la colección de cartas
Crupier::agregarCarta	R1	El crupier requiere agregar una carta a la mano	Se agrega la carta a la mano	Permite agregar cartas a la mano
Jugador::pedirCarta	R1, R3	El jugador solicita una carta del mazo	El jugador obtiene una carta del mazo	El jugador solicita una carta al mazo y este se la otorga
Jugador::plantarse	R1, R3	El jugador requiere plantarse	El jugador se planta	El jugador se planta y muestra el nombre y el valor de su mano
Jugador::calcularValorMano	R1, R5	Se desconoce el valor total de la mano del jugador	Se conoce el valor total de la mano del jugador	Retorna un entero con el valor total de la mano del jugador
Jugador::mostrarMano	R1	Se desconoce la mano que tiene el jugador	Se conoce la mano que tiene el jugador	Si hay cartas en la mano, retorna un texto mostrando cada carta

<u>Jugador::limpiarMano</u>	R1	Colección original con cartas definidas	Se limpia la colección	Permite limpiar la colección de cartas
<u>Jugador::agregarCarta</u>	R1	El Jugador requiere agregar una carta a la mano	Se agrega la carta a la mano	Permite agregar cartas a la mano
<u>Mano::agregarCarta</u>	R1	La mano requiere adicionar una carta a su colección	La mano agrega la carta a su colección	La mano agrega la carta especificada a su colección de cartas
<u>Mano::valorTotal</u>	R1, R5	Se desconoce el valor total de la mano	Se conoce el valor total de la mano	Permite calcular el valor total de la mano
<u>Mano::tieneBlackjack</u>	R1, R7	Se requiere saber si existe Blackjack en las cartas	Determina si existe un Blackjack en las cartas especificadas	Toma las dos cartas y determina si existe Blackjack o no mediante un valor booleano
<u>Mano::mostrarMano</u>	R1	Se desconoce la mano que se tiene	Se conoce la mano que se tiene	Muestra una lista con las cartas que se tienen en la mano
<u>Mano::size</u>	R1	Se desconoce el tamaño de la mano	Se conoce el tamaño de la mano	Retorna un objeto size_t con el tamaño de la mano
<u>Mano::limpiarMano</u>	R1	Colección original con cartas definidas	Se limpia la colección	Permite limpiar la colección de cartas
<u>Mano::obtenerCartas</u>	R1	Se desconoce la mano	Se conoce la mano	Permite retornar una colección copia de la mano
<u>Mazo::barajar</u>	R1	Cartas registradas en orden	Cartas registradas aleatoriamente	Permite organizar aleatoriamente las cartas
<u>Mazo::repartirCarta</u>	R1	Se requiere una carta	Se otorga una carta	Toma una carta de la colección de cartas, la retorna y la elimina de la

				colección original
Mazo::cartasRestantes	R1	Se desconoce la cantidad de cartas restantes	Se conoce la cantidad de cartas restantes	Retorna la cantidad de cartas restantes
Juego::iniciar	R1, R8	No se ha inicializado ninguna mano para el jugador ni para el crupier	El jugador y el crupier tienen una mano inicial	Reparte las primeras cartas al jugador y al crupier y muestra su puntaje
Juego::agregarCartaJugador	R1, R2	El jugador tiene la colección de cartas original	Se agrega una carta a la colección de cartas del jugador	Permite agregar una carta a la colección de cartas del jugador
Juego::mostrarManoJugador	R1	Se desconoce la mano que tiene el jugador	Se conoce la mano que tiene el jugador	Si hay cartas en la mano, retorna un texto mostrando cada carta
Juego::valorJugador	R1, R5	Se desconoce el valor total de la mano del jugador	Se conoce el valor total de la mano del jugador	Retorna un entero con el valor total de la mano del jugador
Juego::turnoCrupier	R1, R2, R4	El crupier tiene su mazo original	El crupier toma una carta mas	Turno del crupier, requiere una carta del mazo
Juego::determinarGanador	R1, R6, R8	Se desconoce el resultado de la ronda	Se conoce el resultado de la ronda	Permite conocer el ganador de la ronda o si existe un empate

# Diagrama de clases

