

## SUPUESTOS

- Se asume que sin importar si el jugador atacado esta en modo defensa o ataque, si el atacante tiene menos puntaje de ataque que el atacado el atacante es destruido.
- Se asume que sin importar si el jugador atacado esta en modo defensa o ataque, si el atacante tiene igual puntaje de ataque que el atacado ambos son destruidos.
- Se asume que la única forma para que una carta vaya al cementerio es muerta.
- Se asume que una carta monstruo solo puede atacar si está en modo ataque.
- Se asume que la cantidad de cartas muertas por jugador no superara la cantidad de cartas en el mazo que le corresponde a dicho jugador.
- Se pueden activar efectos y cartas trampa en respuesta al mismo ataque.

## EXCEPCIONES

**CapacidadMaximaException:** Se lanza al querer seguir agregando cartas a un ContenedorDeCartas cuya capacidad está colmada.

**CartaNoEstaEnCampoException:** Se lanza al querer sacar o manipular una carta que no está en el campo de juego.

**CartaNoEstaEnContenedorDeCartasException:** Se lanza al querer manipular una carta que no está en el contenedor de cartas.

**ContenedorDeCartasVacioException:** Se lanza al querer remover o extraer cartas de un contenedor de cartas vacío.

**MonstruoAtacanteNoEstabaEnModoAtaqueException:** Se lanza al querer llevar a cabo una batalla donde la carta atacante no estaba en modo ataque.

**ParaAtacarDirectamenteAlJugadorNoTieneQueHaberMonstruosInvocadosException:** Se lanza cuando se quiso atacar directamente al jugador y había cartas monstruos del oponente invocados.

**AtaqueIntervenidoException:** Se lanza para guiar el flujo en el caso de activarse un efecto o carta trampa que interviene el ataque de una carta monstruo.

## MODELO DE DOMINIO

**Carta**: Modeliza una carta en el contexto del problema. Tiene estados que cambian según el contexto.

**CartaMonstruo**: Deriva de Carta y es la especificación hacia el concepto de monstruo y su comportamiento. Podrían tener efecto.

**CartaEspecial**: Deriva de Carta y es la abstracción de las cartas mágicas, cartas trampa y cartas de campo.

**CartaMagica**: Deriva de CartaEspecial y es la especificación hacia el concepto de las cartas mágicas y su comportamiento. Tienen efecto.

**CartaDeCampo**: Deriva de CartaEspecial y es la especificación hacia el concepto de las cartas de campo y su comportamiento. Tienen efecto.

**CartaTrampa**: Deriva de CartaEspecial y es la especificación hacia el concepto de las cartas trampa y su comportamiento. Tienen efecto.

**MonstruoRequiereUnSacrificio**: Deriva de CartaMonstruo e implementa el comportamiento específico de la invocación que requiere el sacrificio de un monstruo.

**AgujeroOscuro**: Tiene el efecto de matar a todos los monstruos una vez activada. Implementación de la carta del juego.

**Fisura**: Tiene el efecto de matar a la carta monstruo de menor ataque invocada enemiga.

**OllaDeLaCodicia**: Tiene el efecto de hacer que el poseedor tome dos cartas del mazo.

**Sogen**: Tiene el efecto continuado de aumentar 200 puntos de ataque a las cartas monstruo del poseedor y 500 puntos de defensa a las cartas monstruo del oponente.

**Wasteland**: Tiene el efecto continuado de aumentar 200 puntos de ataque a las cartas monstruo del poseedor y 300 puntos de defensa a las cartas monstruo del oponente.

**CilindroMagico**: Tiene el efecto de frenar el ataque enemigo y devolver ese mismo ataque atacando al oponente.

**Reinforcements**: Tiene el efecto aumentar los puntos de ataque de la carta monstruo bajo ataque.

**EspadachinSilencioso**: Requiere un sacrificio. Es una carta monstruo.

**Jinzo**: Es una carta monstruo. Tiene el efecto de poder atacar directamente al oponente.

**InsectoComeHombres**: Es una carta monstruo. Tiene el efecto de volteo de al ser atacada estando boca abajo de destruir al atacante.

**DragonBlancoDeOjosAzules**: Es una carta monstruo. Requiere dos sacrificios.

**DragonArmadoOscuro**: Es una carta monstruo. Requiere 2 sacrificios.

**DragonDefinitivo**: Es una carta monstruo. Requiere 3 sacrificios.

**AncientBrain**: Es una carta monstruo.

**AlphaTheMagnetWarrior**: Es una carta monstruo.

**Exodia**: Abstracción de las partes de Exodia.

**CabezaExodia**: Es una carta monstruo. Parte de Exodia.

**BrazolzquierdoExodia**: Es una carta monstruo. Parte de Exodia.

**BrazoDerechoExodia**: Es una carta monstruo. Parte de Exodia.

**PiernalzquierdaExodia**: Es una carta monstruo. Parte de Exodia.

**PiernaDerechaExodia**: Es una carta monstruo. Parte de Exodia.

**Estado**: Modela los diferentes comportamientos que puede tener las cartas ante los diferentes estados en los que puede estar. Implementa el StrategyPattern.

**EstadoDeCartaEspecial**: Es una clase intermedia en la jerarquía que pretende evitar la repetición de código de los estados posibles de las cartas especiales.

**EstadoDeCartaMonstruo**: Es una clase intermedia en la jerarquía que pretende evitar la repetición de código de los estados posibles de la CartaMonstruo.

**ModoActivo**: Modela el estado en el cual las cartas mágicas y de trampa activan sus efectos.

**ModoInactivo**: Modela el estado en el cual las cartas mágicas y de trampa no manifiestan efectos.

**ModoAtaque**: Modela el estado en el cual las cartas monstruo pueden atacar o activar sus efectos, así como también habilita una forma específica de cálculo de vida a restar.

**ModoDefensa**: Modela el estado en el cual las cartas monstruo pueden no pueden atacar o activar sus efectos, así como también habilita una forma específica de cálculo de vida a restar.

**ModoDefensaBocaAbajo**: Modela el estado en el cual la carta esta además de a la defensiva volteada. Permite la activación de efectos de volteo.

**Campo**: Clase que se ocupa de la administración de las diferentes zonas que componen el “tablero” del jugador. Incluye el cementerio y el mazo. Hay uno por jugador.

**ContenedorDeCartas**: Clase que se ocupa de la administración y almacenamiento de las cartas. Una forma en la que esta clase se manifiesta es en la implementación de las llamadas zonas.

**Mazo**: Modela el comportamiento esperado de un mazo y provee un contrato interactivo con el jugador que lo requiera. Hereda de ContenedorDeCartas.

**Juego**: Clase que modela el rol del juego mismo.

**Jugador**: Modela a la noción de jugador en el contexto del juego. Tiene acceso a su mano, su campo y su oponente.

**Mano**: Hereda de ContenedorDeCartas y posee además un RecolectorDePartesDeExodia. Modela el rol de la mano en el juego.

**RecolectorDePartesDeExodia**: Es la clase encargada de filtrar y recolectar de las cartas que van entrando en la mano cuales son las partes del Exodia.

**FormaDeAfectarAlJugador**: Interfaz que modela los diferentes comportamientos que puede tomar la carta monstruo atacante a la hora de afectar al jugador y a la carta atacada en el contexto de un ataque.

**Restar**: Una forma específica de afectar al jugador en donde se resta puntaje de vida. Se usa en caso de darse una batalla donde ambas cartas están en modo ataque.

**NoRestar**: Una forma específica de afectar al jugador en donde no se resta puntaje de vida. Se usa en caso de darse una batalla donde una carta está en modo ataque y la atacada en modo defensa.

**Efecto**: Es la representación misma de un efecto que puede tener una carta.

**EfectoInvocacionMonstruo**: Deriva de Efecto y modela a los tipos de efectos que se activan al invocarse los monstruos.

**EfectoDeVolteo**: Deriva de Efecto y modela a los tipos de efectos que se activan al darse un volteo, es decir, se pasa de boca abajo a boca arriba o la carta es atacada estando boca abajo.

**EfectoDeCampo**: Deriva de Efecto y modela a los tipos de efectos que se son propios de cartas de campo.

**EfectoEspeciales**: Deriva de Efecto y modela a los tipos de efectos que se son propios de cartas especiales.

**EfectoNulo**: Deriva de Efecto y aplica NullPattern sobre cartas que no poseen efectos.

**EfectoJinzo**: Deriva de EfectoInvocacionMonstruo y modela al efecto de Jinzo.

**EfectoWasteland**: Deriva de EfectoDeCampo y modela al efecto de Wasteland.

**EfectoSogen**: Deriva de EfectoDeCampo y modela al efecto de Sogen.

**EfectoInsectoComeHombres**: Deriva de EfectoDeVolteo y modela al efecto del InsectoComeHombres.

**EfectoCilindroMagico**: Deriva de EfectoEspeciales y modela al efecto del CilindroMagico.

**EfectoReinforcements**: Deriva de EfectoEspeciales y modela al efecto de Reinforcements.

**EfectoOllaDeLaCodicia**: Deriva de EfectoEspeciales y modela al efecto de la OllaDeLaCodicia.

**EfectoFisura**: Deriva de EfectoEspeciales y modela al efecto de la Fisura.

**EfectoAgujeroOscuro**: Deriva de EfectoEspeciales y modela al efecto del AgujeroOscuro.

## DETALLES DE IMPLEMENTACIÓN

**Mecánica de Estados**: La idea de esto fue usar Strategy ya que vimos que las cartas podían cambiar de estado a través del juego y que este influía sobre su comportamiento. Además gracias a la separación del estado de la carta se puede usar DispatchPattern en varias ocasiones.

**Las cartas específicas son más estructuras que clases**: Se buscó usar Template. No se vio otra forma de implementarlo.

**Mecánica de cartas muertas o destruidas**: Las cartas a lo largo de su vida van cambiando de estado y van cambiando de zona en la que “habitan”. Dado que en ocasiones la carta podría estar en dos lugares a la vez, esta posee una Collection donde guarda quien la contiene. Al morir la carta, dado

que esta conoce a su jugador dueño, accede a través de el al cementerio y se mete ahí. Luego, recorre todas las zonas que la contenían y se elimina de ahí.

**Mecánica de efectos:** Cada carta tiene un efecto (aun cuando en la lógica del juego no lo tiene). Este efecto es activado por un estímulo externo y condicionado a que el estado de la carta que lo contiene sea el correcto. De ahí el uso de DispatchPattern. Cada estado “activa” o procede a la activación de distintos efectos de forma distinta. Los efectos son provistos además con referencias a ambos jugadores, y en casos de necesidad a los monstruos involucrados en su ejecución. Estas últimas dos referencias son eliminadas una vez finalizado el efecto para evitar problemas de invariantes en el futuro.

**Interacción Jugador y Cartas:** El jugador tiene acceso a sus cartas a través de su Mano o de su Campo en forma indirecta. Además este tiene acceso a su oponente y por tanto de forma indirecta podría acceder a las cartas enemigas.

**La mecánica de batallas:** Las batallas entre cartas se dan de carta a carta sin clases que medien. Se usa otra vez DispatchPattern para determinar en función del estado de la carta atacada cual es la forma de afectarla (si es modo defensa o ataque). Además, justo antes de realizar el cálculo de daño se disparan los efectos y cartas trampa del jugador o carta afectada. Si el efecto o carta trampa es la adecuada se provee la posibilidad de cortar el ataque enemigo. Esta última parte usa ProxyPattern, donde el proxy vendría a ser el efecto o la carta trampa que se activa antes del cálculo de daño y responde por la carta monstruo atacada.

**Recolecto de Partes de Exodia y condición de victoria:** Hay una clase que cada vez que se agrega una carta a la mano es llamada para que chequee y filtre si esa carta es una parte del Exodia (otra vez usando DispatchPattern). Además chequea si esa parte ya la tiene en la mano, si no, la agrega, si la tenía la deja pasar.