

## Gwent - T.P. 2

[TB025] Paradigmas de Programación

Primer cuatrimestre de 2025

**Grupo 07 - Estudiantes:**

Nombre	Padrón	Mail
Matias Domine Folco	112268	mdomine@fi.uba.ar
Enrique Heller	111605	eheller@fi.uba.ar
Andy Mayuri	111492	amayuri@fi.uba.ar
Franco Montanelli	111521	fmontanelli@fi.uba.ar
Andrés Moyano	110017	ammoyano@fi.uba.ar

**Tutor: Joaquín Pandolfi**

## Índice

<b>1. Supuestos.....</b>	<b>3</b>
1.1. Cartas Especiales como modificadores.....	3
<b>2. Diagramas de clase.....</b>	<b>3</b>
2.1. Diagramas del Modelo Tentativo:.....	3
<b>3. Diagramas de secuencia.....</b>	<b>6</b>
3.1. Diagrama de Secuencia de Modelo Tentativo(Entrega-0):.....	6
3.2. Entrega-1:.....	7
3.2.1. Verificar que un jugador posea cartas suficientes para empezar el juego en su mazo.....	7
3.2.2. Verificar que a un jugador se le reparten 10 cartas de su mazo.....	8
3.2.3. Verificar que un jugador pueda colocar una carta en una sección del tablero.....	9
3.2.4. Verificar que un jugador juegue una carta de su mazo y tenga un puntaje parcial.....	10
3.2.5. Verificar que las cartas pasen a la pila de descarte.....	10
3.2.6. Verificar que al modificar una carta con una carta unida se cambien sus puntos y se aplique el valor solo a la ronda.....	10
3.2.7. Verificar que al utilizar una carta especial de clima se reduzca el valor de las cartas de la sección correspondiente.....	10
<b>4. Diagramas de paquete.....</b>	<b>10</b>
<b>5. Detalles de implementación.....</b>	<b>10</b>
<b>6. Excepciones.....</b>	<b>10</b>

## 1. Supuestos

**1.1. Cartas Especiales como modificadores:** Las **Cartas Especiales** se caracterizan por tener la capacidad de modificar aquellas cartas que se encuentren distribuidas por el tablero. Teniendo en cuenta que los **Modificadores** que pueden ser atribuidos a las **Cartas Unidad** también tienen dicha injerencia sobre el tablero en general, atribuimos la característica de modificador a las cartas especiales en general, a partir de que tengan que implementar la interface **Modificador**, siendo cada una responsable de definir de que forma implementará **Modificar**

## 2. Diagramas de clase

A continuación se dará lugar a los diagramas de clase que nos permiten observar las distintas entidades definidas en nuestro modelo y sus relaciones entre sí. Comenzamos presentando aquellas entidades que entendemos como principales dentro del modelo:

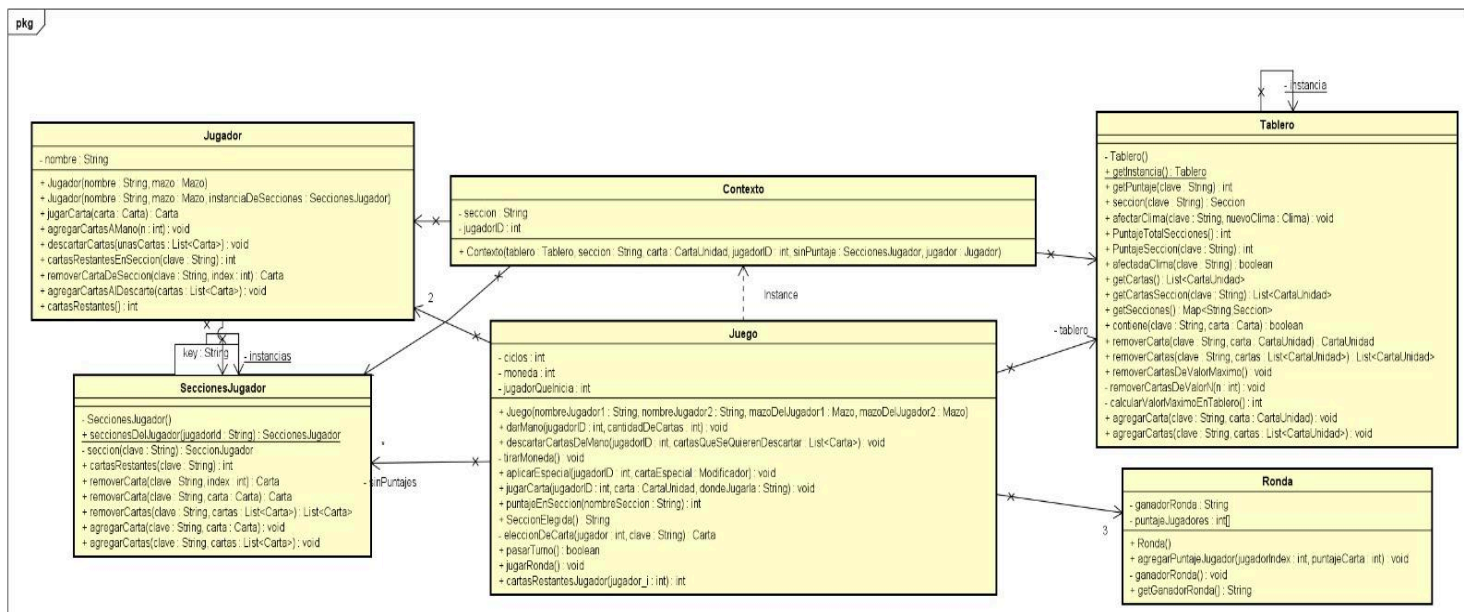


Figura 1: Diagrama de clases principal

Nuestro diagrama presenta la clase **Juego**, la cual se define con el objetivo de cumplir la función de “fachada” de nuestro modelo. Esto quiere decir, que es a partir de ella y solo de

ella que el usuario va a poder interactuar con el sistema. De los métodos que se destacan dentro de esta clase son el constructor **Juego**, el cual permite dar comienzo a la fase inicial, a partir de la creación de los jugadores con sus correspondientes mazos; **darMano**, **descartarCartasDeMano** y **tirarMoneda**, métodos definidos para dar paso a la fase de preparación del juego y finalmente **jugarRonda**, **jugarCarta**, **aplicarEspecial** y **pasarTurno**, que permiten llevar a cabo toda la fase de Juego.

Las **Rondas** son aquellas que almacenarán y administrarán la información que derive de haber finalizado una ronda. Los puntajes que vayan obteniendo los jugadores serán descritos allí.

**Jugador**, definirá la lógica mediante la cual cada jugador interactuará con las secciones propias y compartidas.

Por otro lado se destacan las clase **SeccionesJugador** y **Tablero**, que tienen la particularidad de ser clases con la responsabilidad de alojar su propia instancia, definiendo un **singleton** para cada una. Esta decisión de diseño a priori es tomada, tal que por las características que iremos describiendo notaremos que el acceso constante (principalmente desde los modificadores) a estas entidades deriva en que el acceso a única instancia compartida desde la clase resulte conveniente:

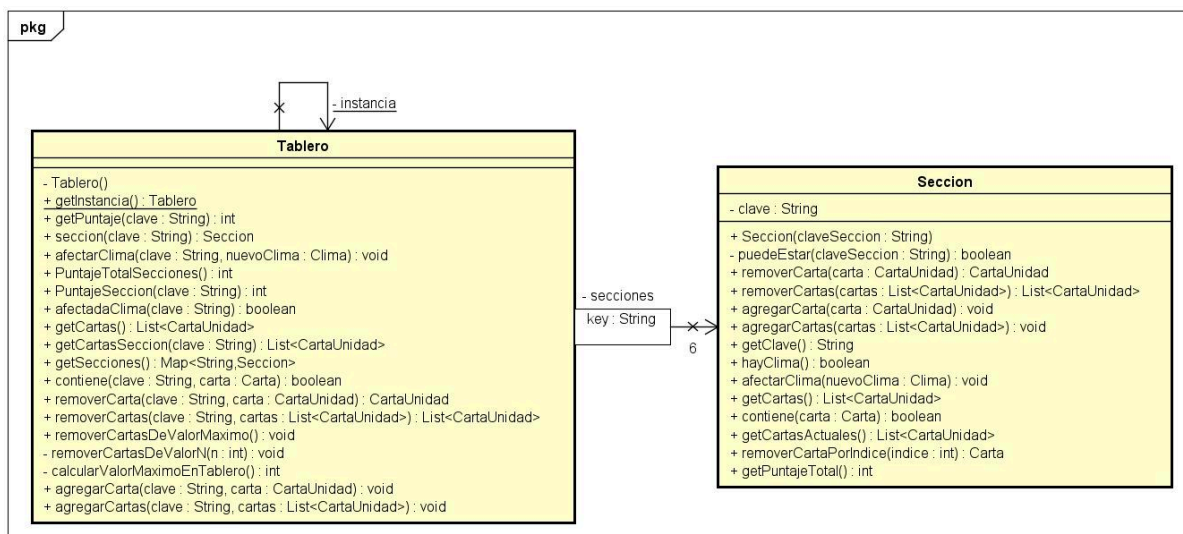


Figura 2: Singleton aplicado

Por otro lado, la clase **Contexto** cumple la función de puente entre el **Jugador** y **Tablero**, almacenando la información relevante para el turno en específico que se esté desarrollando. A

continuación se pasa a mostrar diagrama de clase para comprender en mejores términos su inferencia:

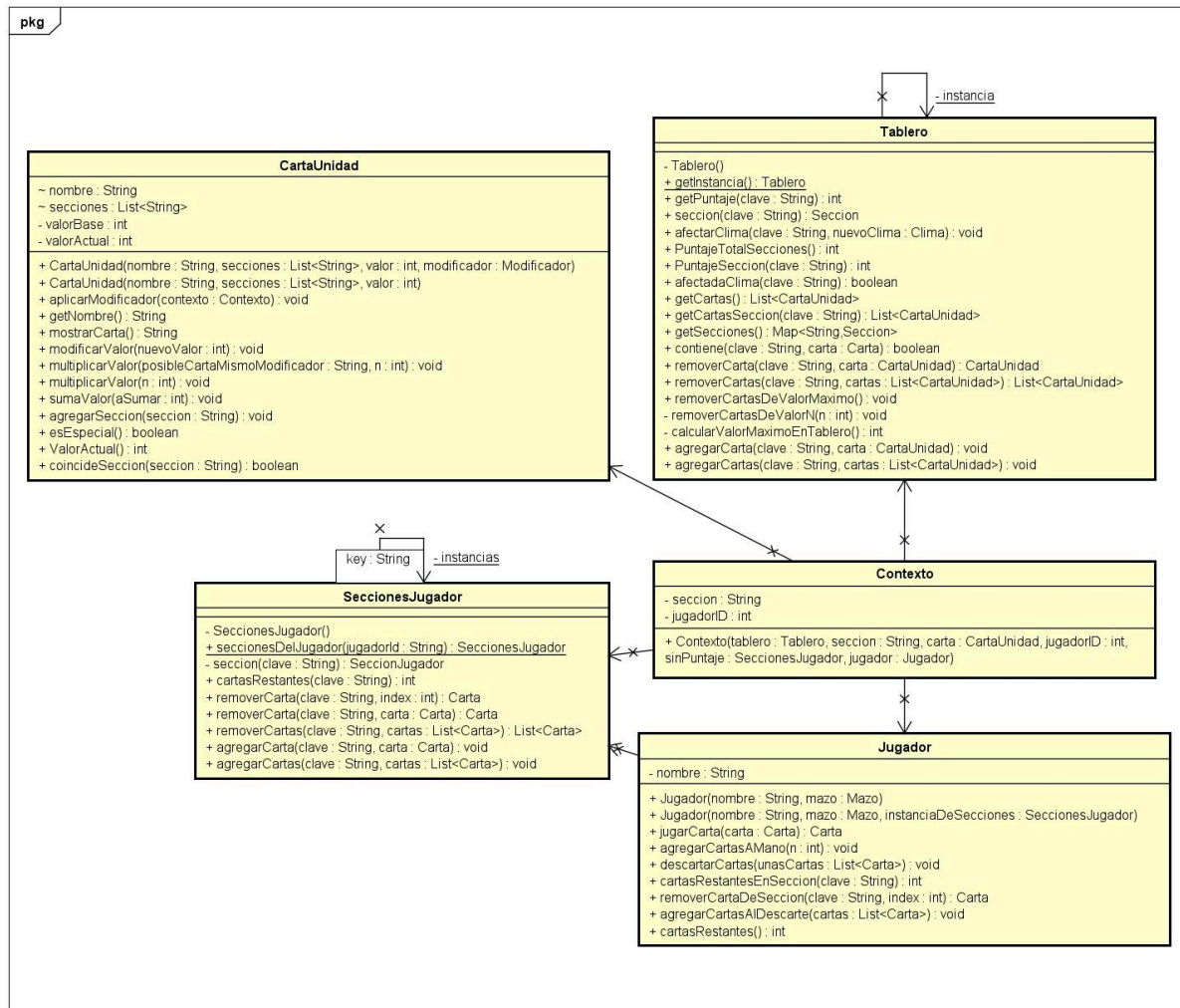


Figura 3: Contexto y sus referencias

A continuación mostramos puntualmente la definición de CartaUnidad:

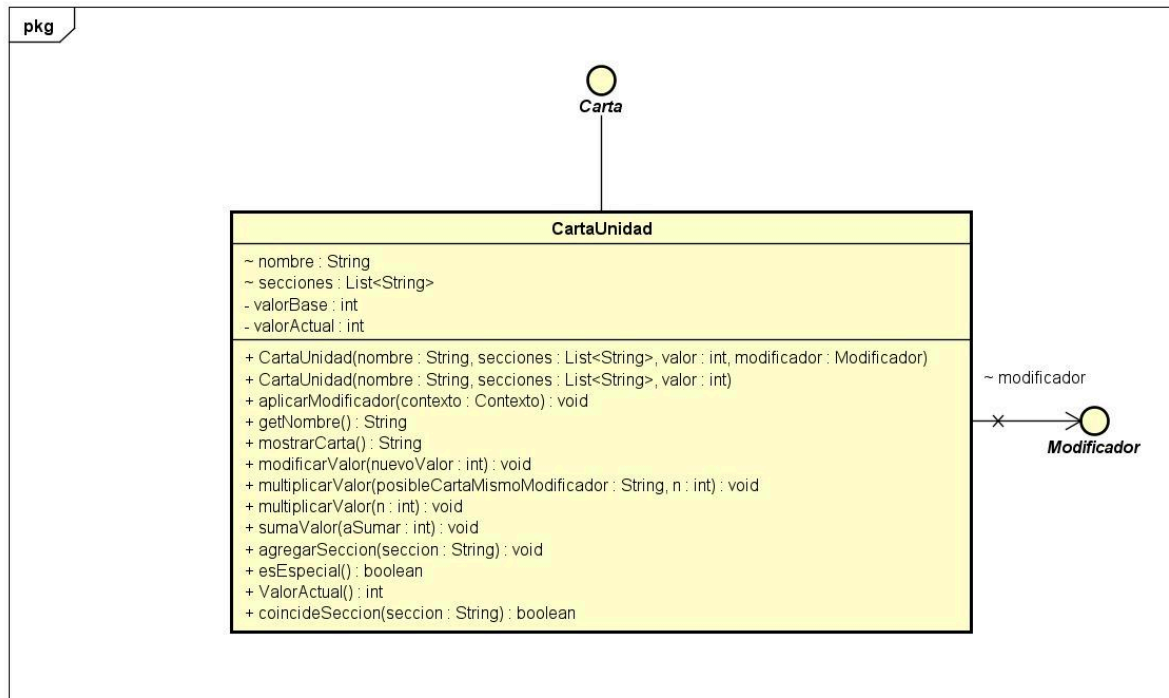
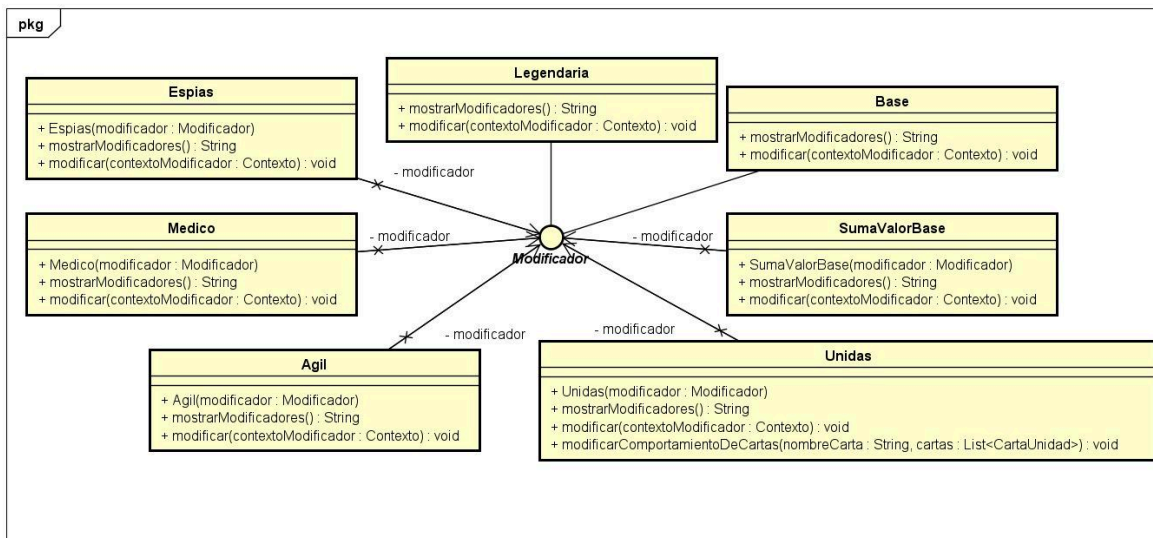


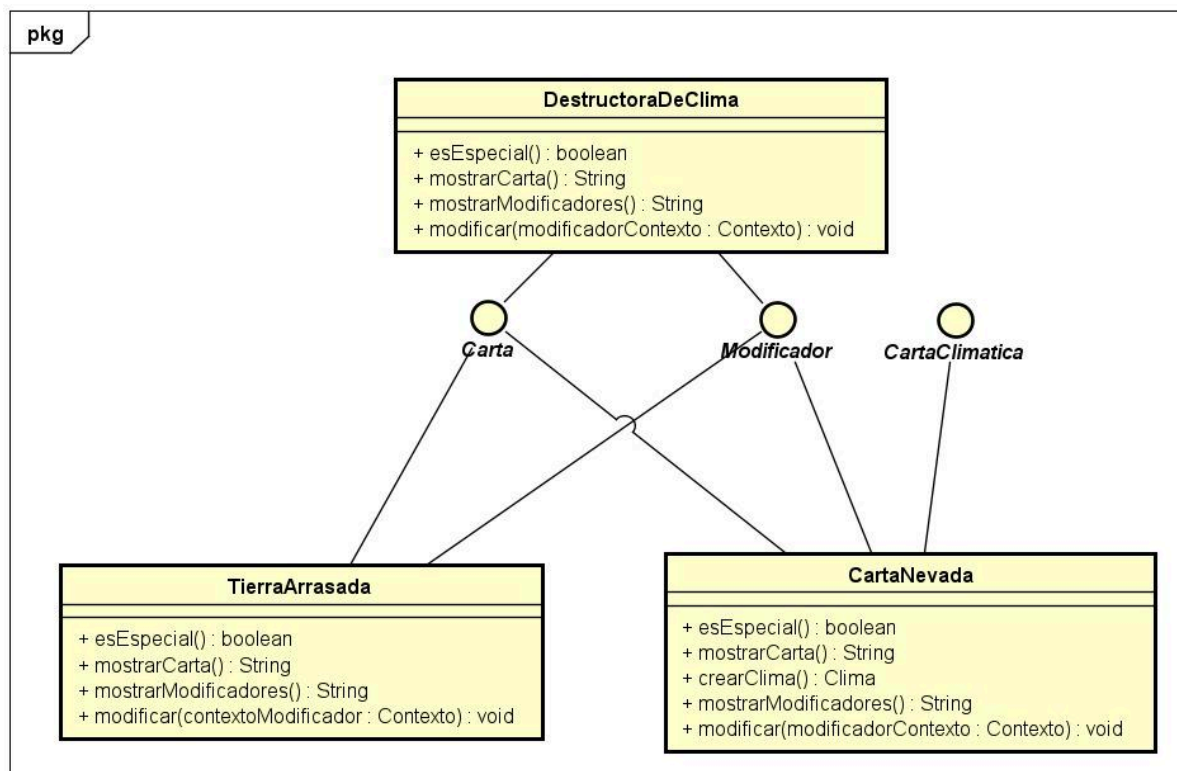
Figura 4: Carta Unidad

Modificadores:



(…)

Cartas Especiales:



(...)

### 3. Diagramas de secuencia (a actualizar)

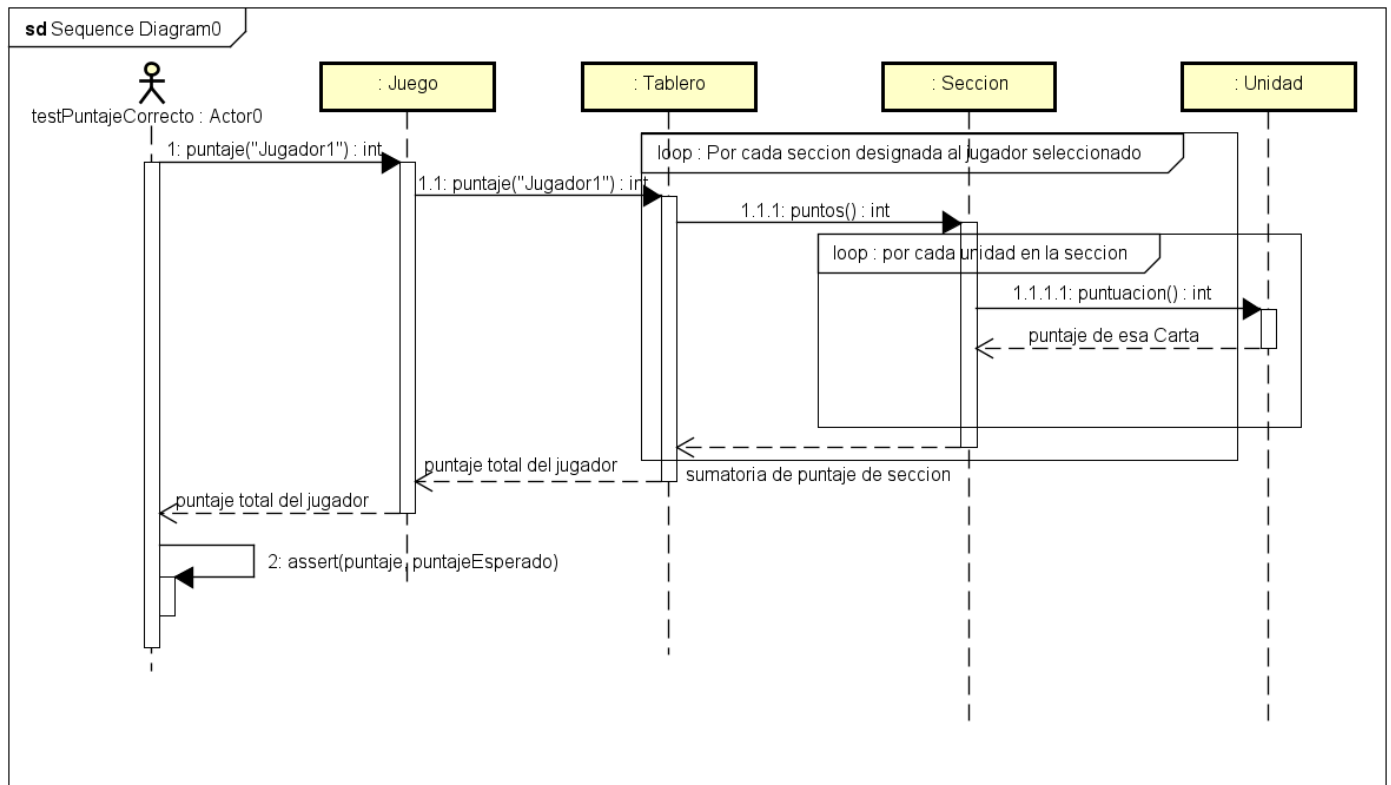


Figura 4: Diagrama de secuencia, cálculo del puntaje de un jugador basado en las cartas que se encuentran en el tablero.

### 3.1. Entrega-1:



3.1.1. Verificar que un jugador posea cartas suficientes para empezar el juego en su mazo.

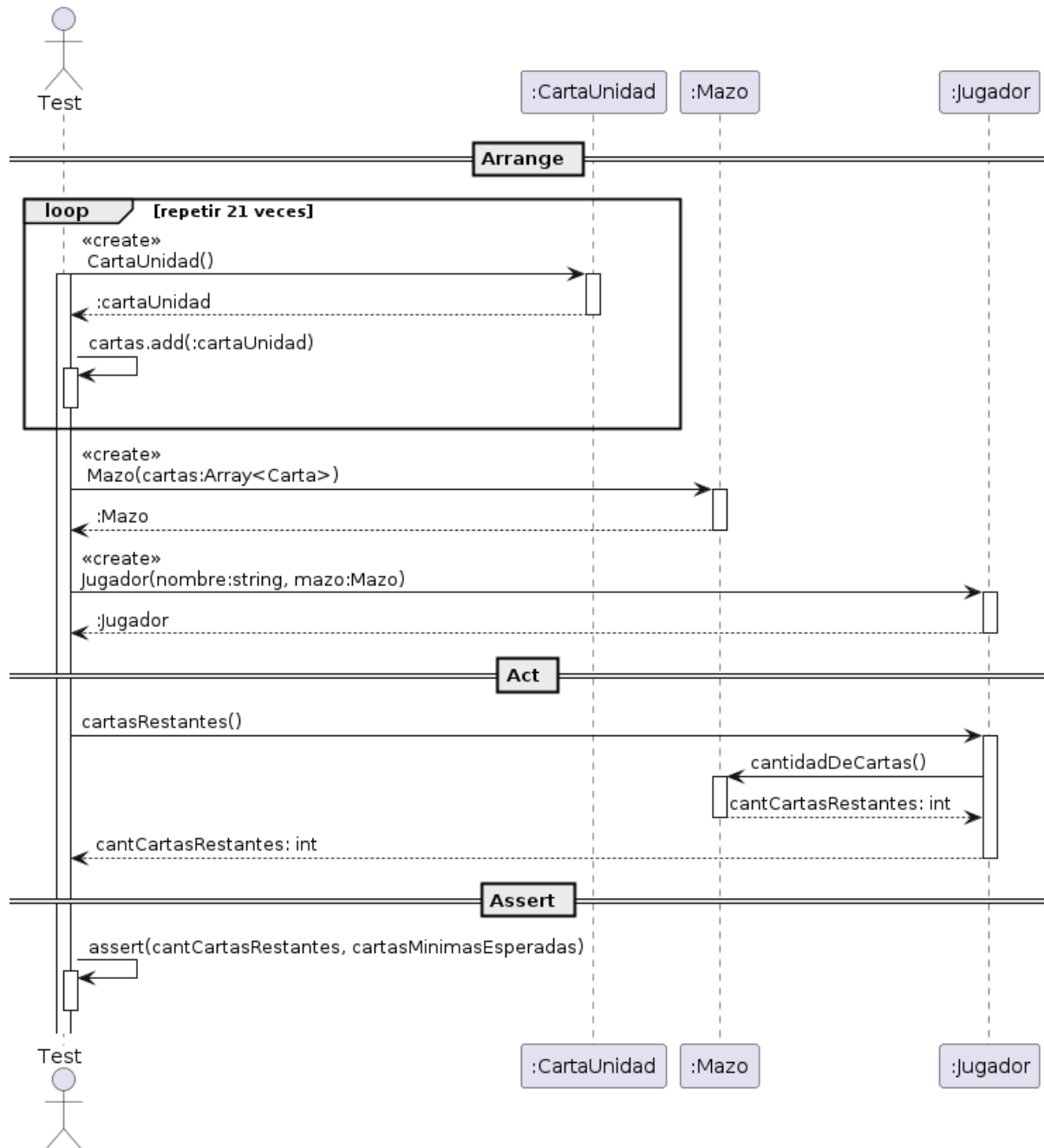


Figura 5: Test-01 Jugador Con Cartas Suficientes Al Empezar

## 3.1.2. Verificar que a un jugador se le reparten 10 cartas de su mazo.

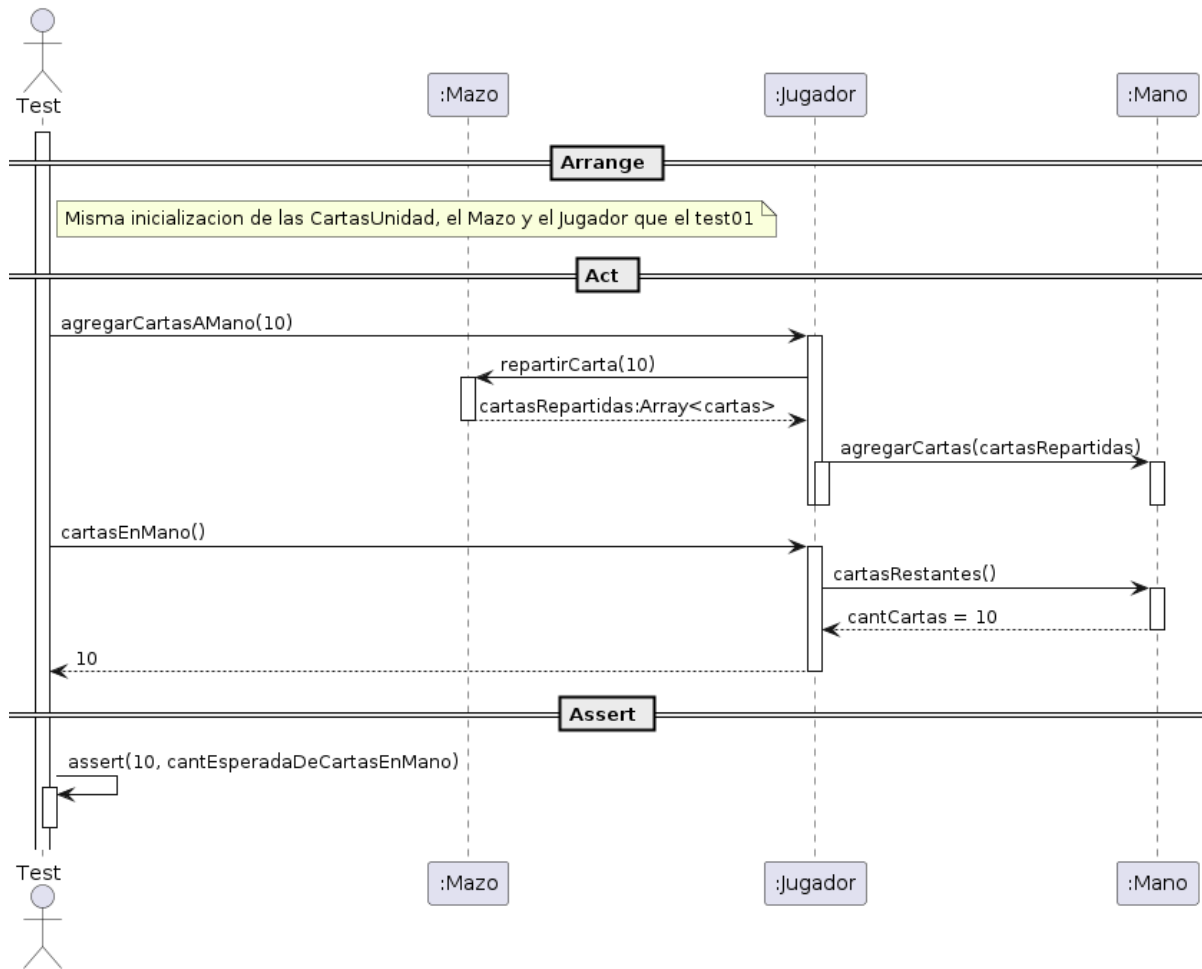


Figura 6: Test-02 Jugador Se Le Reparten 10 Cartas De Su Mazo

## 3.1.3. Verificar que un jugador pueda colocar una carta en una sección del tablero

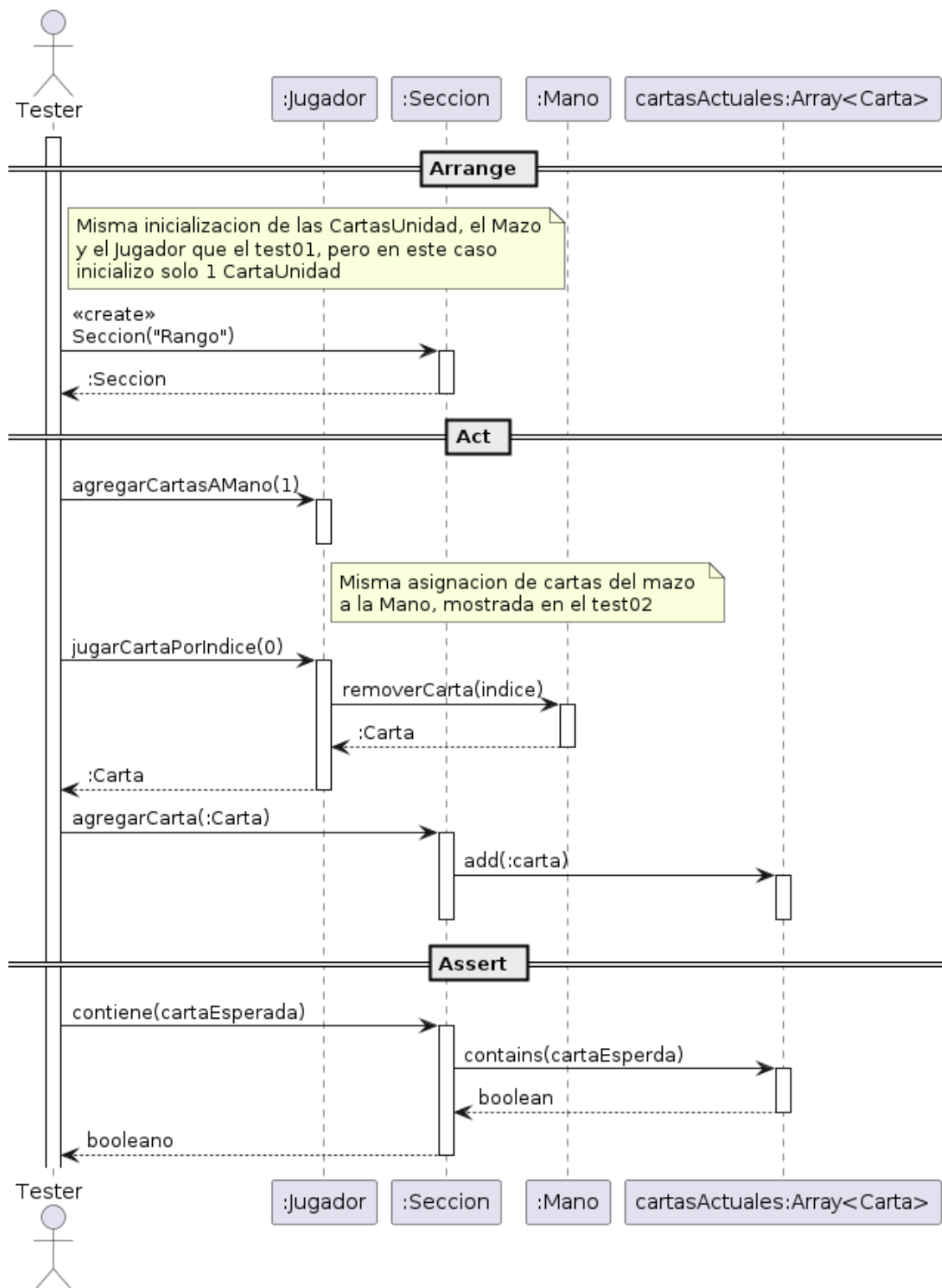


Figura 7: Test-03 Jugador Puede Colocar Carta En Sección Del Tablero

### 3.1.4. Verificar que un jugador juegue una carta de su mazo y tenga un puntaje parcial.

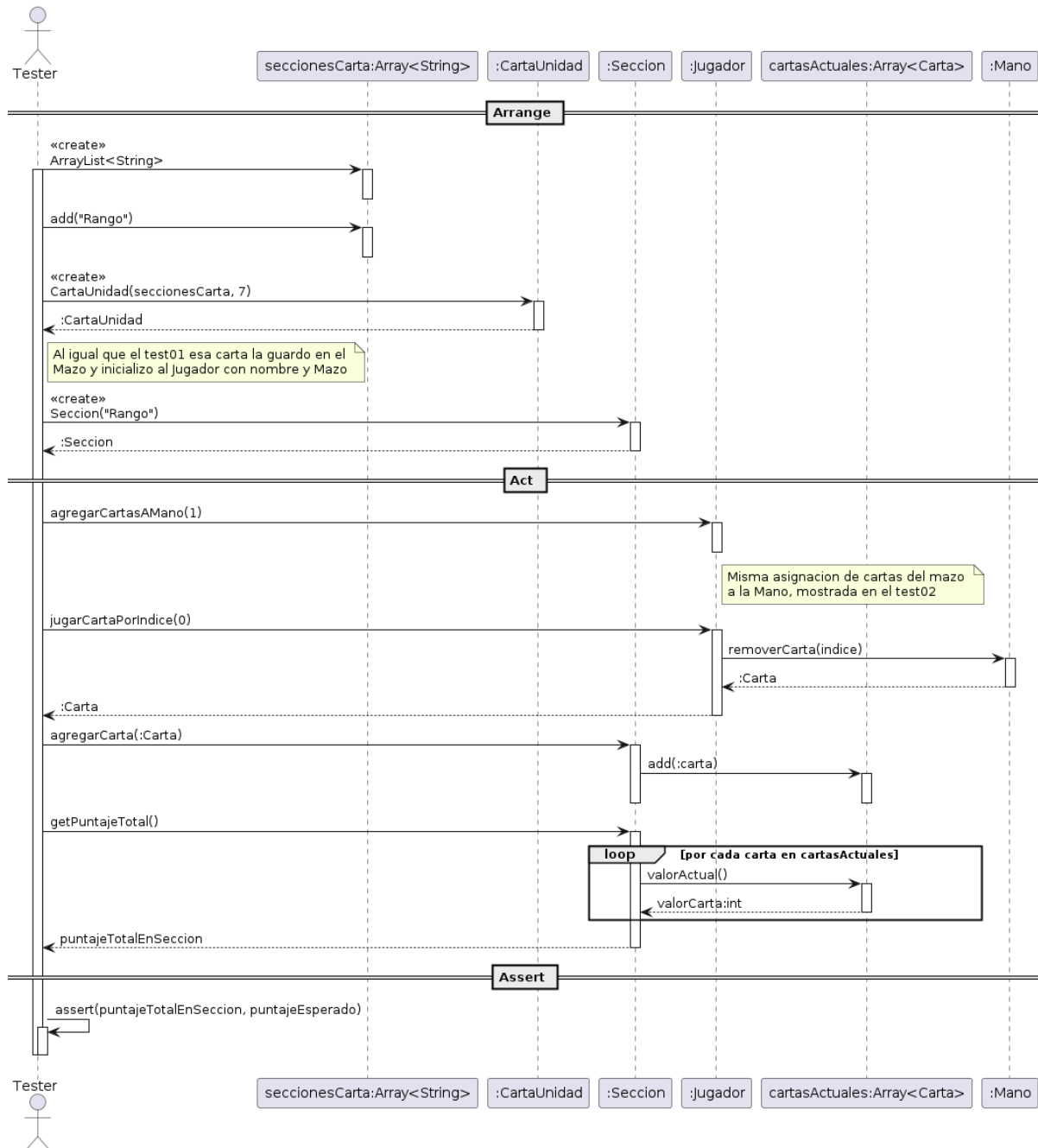


Figura 8: Test-04 Jugador Juega Carta Y Tiene Puntaje Parcial

- 3.1.5. Verificar que las cartas pasen a la pila de descarte
- 3.1.6. Verificar que al modificar una carta con una carta unida se cambien sus puntos y se aplique el valor solo a la ronda.
- 3.1.7. Verificar que al utilizar una carta especial de clima se reduzca el valor de las cartas de la sección correspondiente.

## **4. Diagramas de paquete**

## **5. Detalles de implementación**

## **6. Excepciones**