

## Trabajo Práctico 2 — Kahoot

[7507/9502] Algoritmos y Programación III

Curso 1

Primer cuatrimestre de 2020

Grupo 9	
Alumno 1:	BIANCARDI, Julian
Número de padrón:	103945
Email:	jbiancardi@fi.uba.ar
Alumno 2:	HETREA, Joaquín Emanuel
Número de padrón:	103944
Email:	jhetrea@fi.uba.ar
Alumno 3:	TARDITI PELLEGRINO, Leo Damian
Número de padrón:	104521
Email:	ltarditi@fi.uba.ar
Alumno 4:	GALIAN, Tomás Ezequiel
Número de padrón:	104354
Email:	tgalian@fi.uba.ar
Alumno 5:	PACHECO, Federico Jose
Número de padrón:	104541
Email:	fpacheco@fi.uba.ar

# Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Supuestos</b>	<b>2</b>
2.1. Verdadero y Falso . . . . .	2
2.2. Múltiple Choice . . . . .	2
2.3. Ordered Choice . . . . .	2
2.4. Group Choice . . . . .	2
2.5. Jugador . . . . .	2
2.6. Exclusividad . . . . .	2
2.7. Multiplicadores . . . . .	3
<b>3. Modelo de dominio</b>	<b>4</b>
3.1. Diagramas de Paquetes . . . . .	4
3.2. Diagramas de Clases . . . . .	4
3.2.1. Relación general de todo el modelo de dominio . . . . .	4
3.2.2. Jugador y Multiplicador . . . . .	5
3.2.3. Preguntas en general . . . . .	6
3.2.4. Verdadero/Falso (VoF) . . . . .	7
3.2.5. Multiple Choice . . . . .	8
3.2.6. Multiple Choice . . . . .	9
3.2.7. Múltiple Choice . . . . .	10
3.2.8. Modos de Pregunta . . . . .	11
3.2.9. Respuestas . . . . .	12
3.2.10. Opciones . . . . .	13
3.2.11. Juego . . . . .	14
<b>4. Detalles de implementación</b>	<b>15</b>
4.1. Modos de pregunta . . . . .	15
4.2. Multiplicadores y exclusividad . . . . .	15
4.2.1. Patrones de diseño usados . . . . .	16
4.2.2. Patrón de Arquitectura: MVC . . . . .	16
<b>5. Diagramas de secuencia</b>	<b>17</b>
5.1. Creaciones de las diferentes Preguntas . . . . .	17
5.1.1. Creación: Pregunta Verdadero o Falso . . . . .	17
5.1.2. Creación: Pregunta Múltiple Choice . . . . .	18
5.1.3. Creación: Pregunta Ordered Choice . . . . .	19
5.1.4. Creación: Pregunta Group Choice . . . . .	20
5.1.5. Creación: Respuesta Group Choice . . . . .	21
5.2. Evaluación para toda Pregunta . . . . .	22
5.3. Obtención Cálculo de Puntaje para cada Modo . . . . .	23
5.3.1. Para Modo Clásico . . . . .	23
5.3.2. Para Modo Penalizado . . . . .	24
5.3.3. Para Modo Puntaje Parcial . . . . .	25
5.4. Activar multiplicador doble y triple . . . . .	26
5.5. Modo Pregunta con Exclusividades Activadas . . . . .	27
5.5.1. Con 1 Exclusividad activada . . . . .	27
5.5.2. Con 2 Exclusividades activadas . . . . .	28
<b>6. Excepciones</b>	<b>29</b>

## 1. Introducción

Desarrollar una aplicación de manera grupal aplicando todos los conceptos vistos en el curso, utilizando un lenguaje de tipado estático (Java) con un diseño del modelo orientado a objetos y trabajando con las técnicas de TDD e Integración Continua.

La aplicación consiste en un juego por turnos, de dos jugadores conformado de un panel en el cual se mostraran preguntas con múltiples opciones de respuesta.

Cada pregunta será mostrada dos veces, una vez para cada jugador (al estar jugando los dos en la misma computadora, será responsabilidad de cada jugador no mirar la pantalla mientras el otro responde). El jugador dispone de un tiempo limitado para responder cada pregunta.

Existen varios tipos de preguntas, que asignan puntaje en forma diferenciada a cada jugador dependiendo de cómo responde cada uno.

También existen opciones como los multiplicadores y la exclusividad de puntaje que cada jugador puede utilizar para mejorar sus oportunidades de obtener puntos.

El objetivo del juego es lograr más puntos que el otro jugador respondiendo correctamente las preguntas.

## 2. Supuestos

### 2.1. Verdadero y Falso

- No se puede crear una pregunta verdadero y falso en modo puntaje parcial

### 2.2. Múltiple Choice

- Se crea múltiple choice con al menos 1 respuesta correcta y al menos 2 opciones disponibles.
- En el modo penalidad el puntaje será el resultado de la suma de las respuestas correctas menos las respuestas incorrectas.

### 2.3. Ordered Choice

- Se crea Ordered Choice con al menos 2 opciones disponibles.
- No se puede crear una pregunta Ordered Choice con modo puntaje parcial ni con modo penalidad. Solo se crea por default con modo clásico.

### 2.4. Group Choice

- Se crea la pregunta con al menos 2 opciones disponibles.
- No se puede crear una pregunta Group Choice con modo puntaje parcial ni con modo penalidad. Solo se crea por default con modo clásico.

### 2.5. Jugador

- El jugador debe contar con un nombre particular.

### 2.6. Exclusividad

- Al activar el Modo Exclusividad en una Pregunta, toda respuesta será considerada **correcta** cuando el **jugador acierte todas las opciones correctas** que había, sin ninguna incorrecta; independientemente si haya estado en juego o no el Modo de Puntaje Parcial.
- En otras palabras, tener una *respuesta parcialmente correcta* no es considerado correcta al activar el Modo Exclusividad.

- Si al menos una Exclusividad está activa, la pregunta puntuará solo en el caso de que UNA respuesta sea correcta, y ese jugador será el único que recibirá puntos. Pero si hay más de una respuesta correcta: ninguno recibirá puntos.
- Si al menos una Exclusividad está activa, y ningún jugador obtuvo una respuesta *correcta*; o ambos jugadores tuvieron respuestas parcialmente correcta: ningún jugador sumará puntos.
- Otra cosa a tener en cuenta es que si el jugador decide usar la exclusividad no habrá vuelta atrás, es decir el jugador no podrá sacarle la exclusividad a esa pregunta.

## 2.7. Multiplicadores

- Lo mismo ocurre con los multiplicadores. Una vez que se decide usar un multiplicador en alguna pregunta ya no se puede quitar.
- El multiplicador que se use será el ultimo que se seleccione. Es decir si tocamos para activar el multiplicador doble y luego en la misma pregunta tocamos para activar el multiplicador triple, el que contará será este último.

### 3. Modelo de dominio

#### 3.1. Diagramas de Paquetes

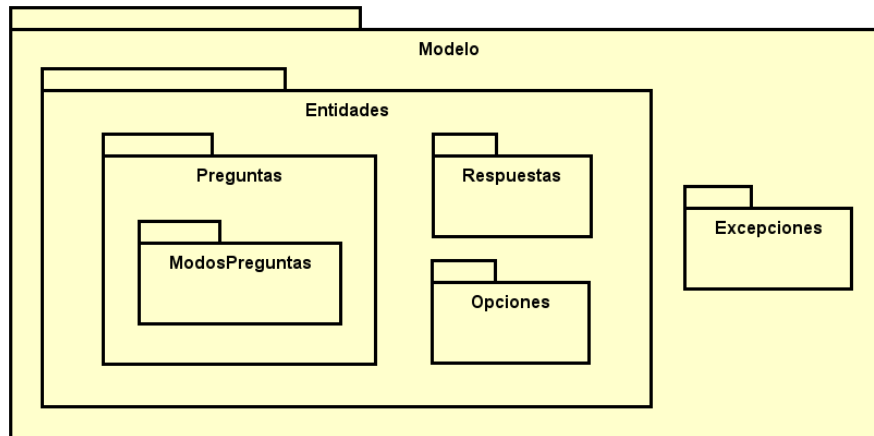


Figura 1: Diagrama de Paquetes en general de todo el modelo de dominio

#### 3.2. Diagramas de Clases

##### 3.2.1. Relación general de todo el modelo de dominio

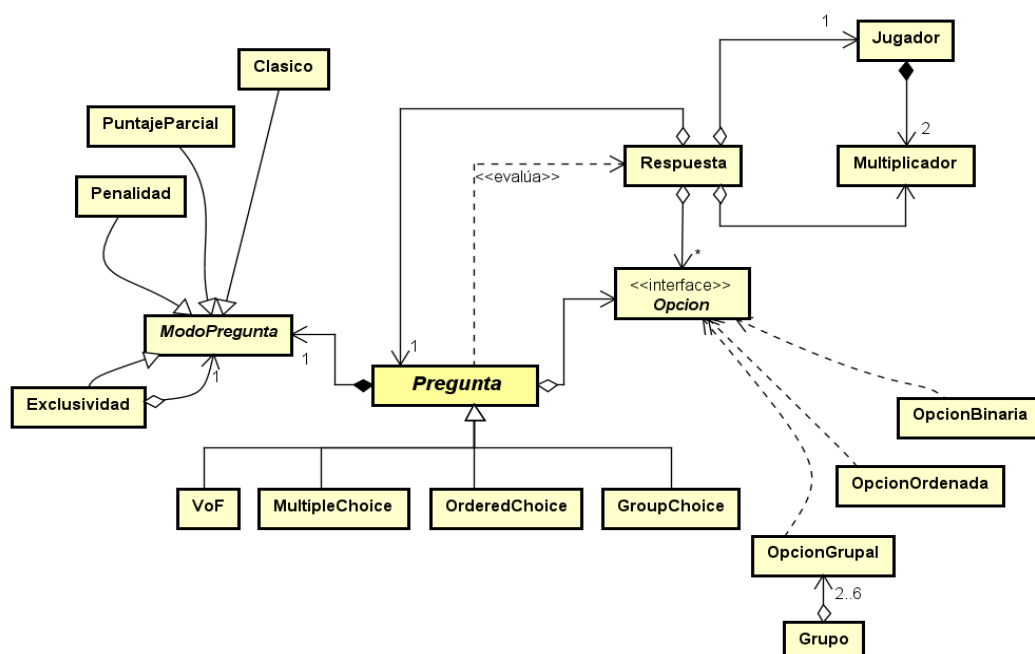


Figura 2: Relación general de las diferentes clases

## 3.2.2. Jugador y Multiplicador

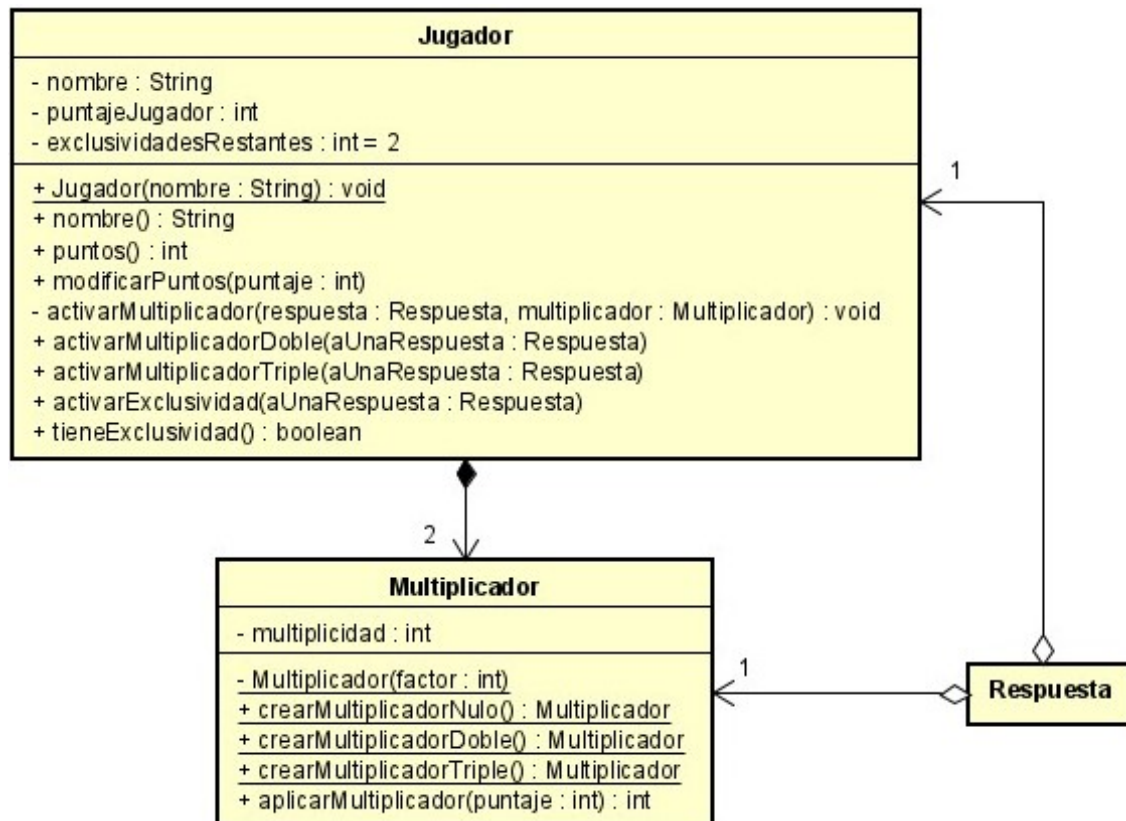


Figura 3: Diagrama de Clase del Jugador y Multiplicador

## 3.2.3. Preguntas en general

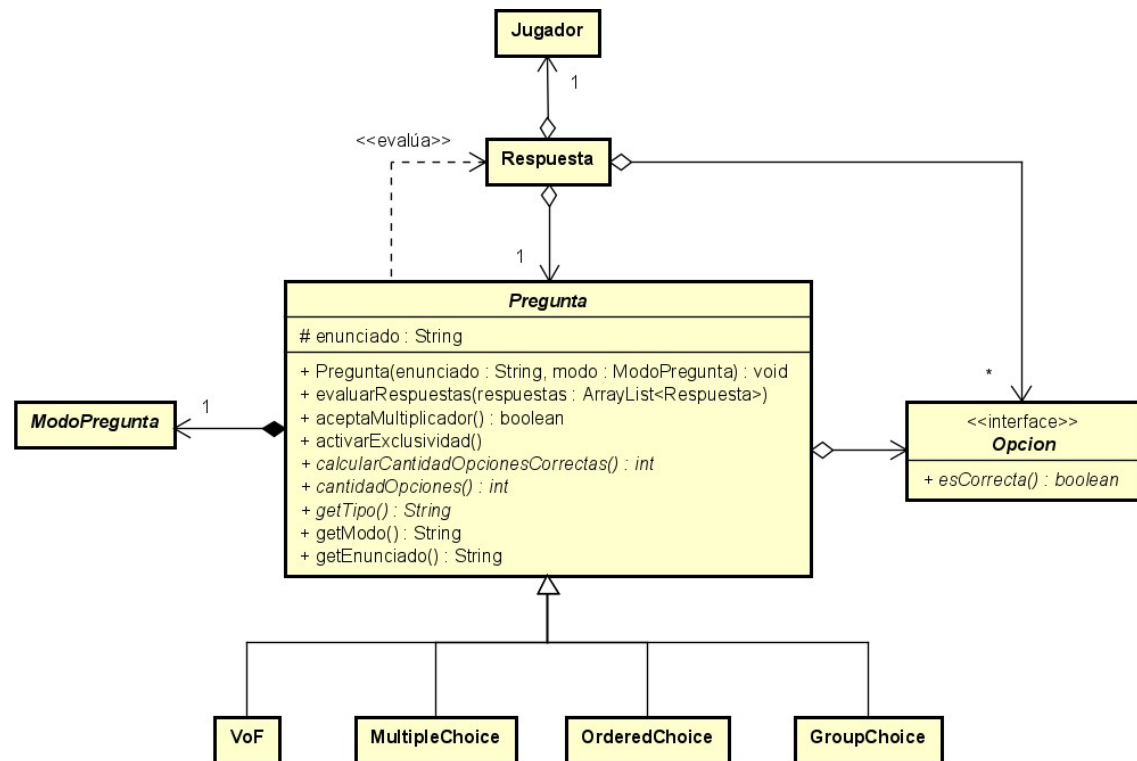


Figura 4: Diagrama general de una Pregunta

## 3.2.4. Verdadero/Falso (VoF)

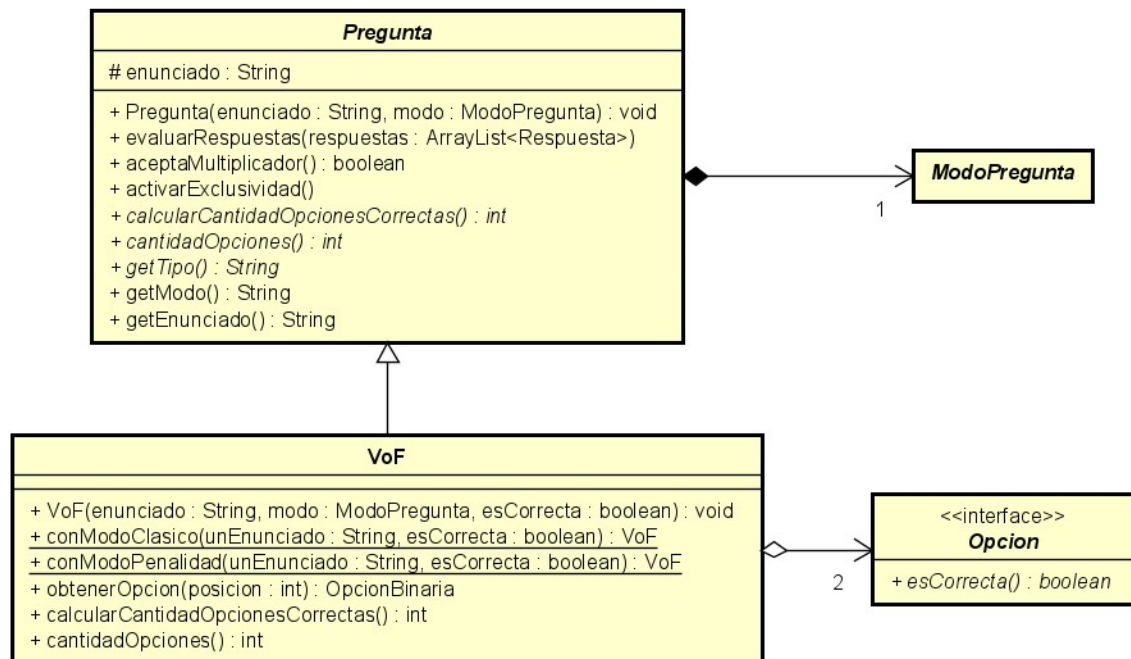


Figura 5: Diagrama específico de la Pregunta Verdadero/Falso (VoF) y Múltiple Choice



## 3.2.5. Multiple Choice

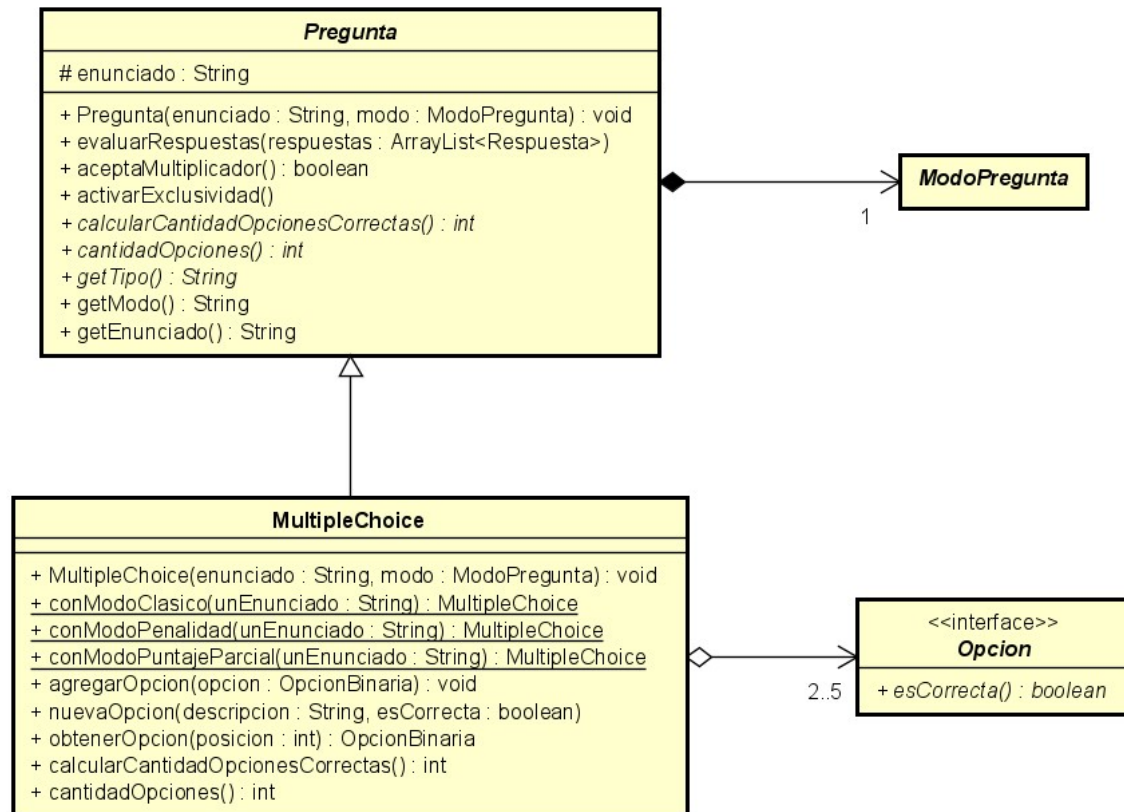


Figura 6: Diagrama específico de la Pregunta Verdadero/Falso (VoF) y Múltiple Choice

## 3.2.6. Multiple Choice

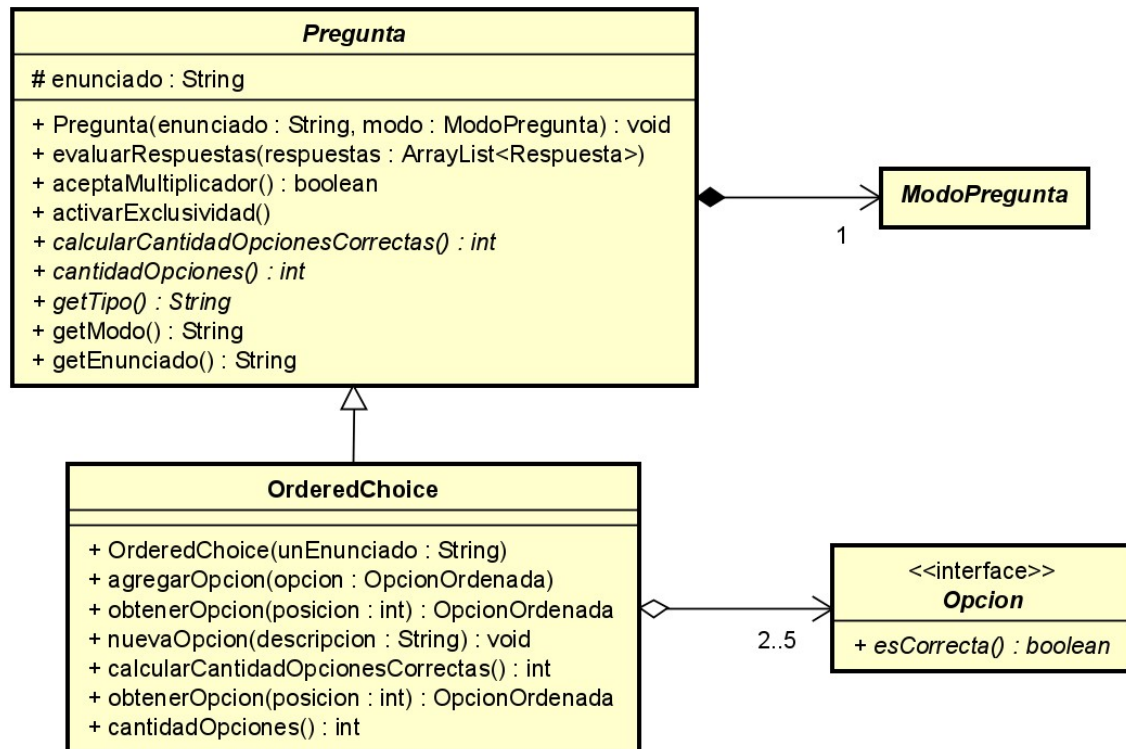


Figura 7: Diagrama específico de la Pregunta Verdadero/Falso (VoF) y Múltiple Choice

## 3.2.7. Múltiple Choice

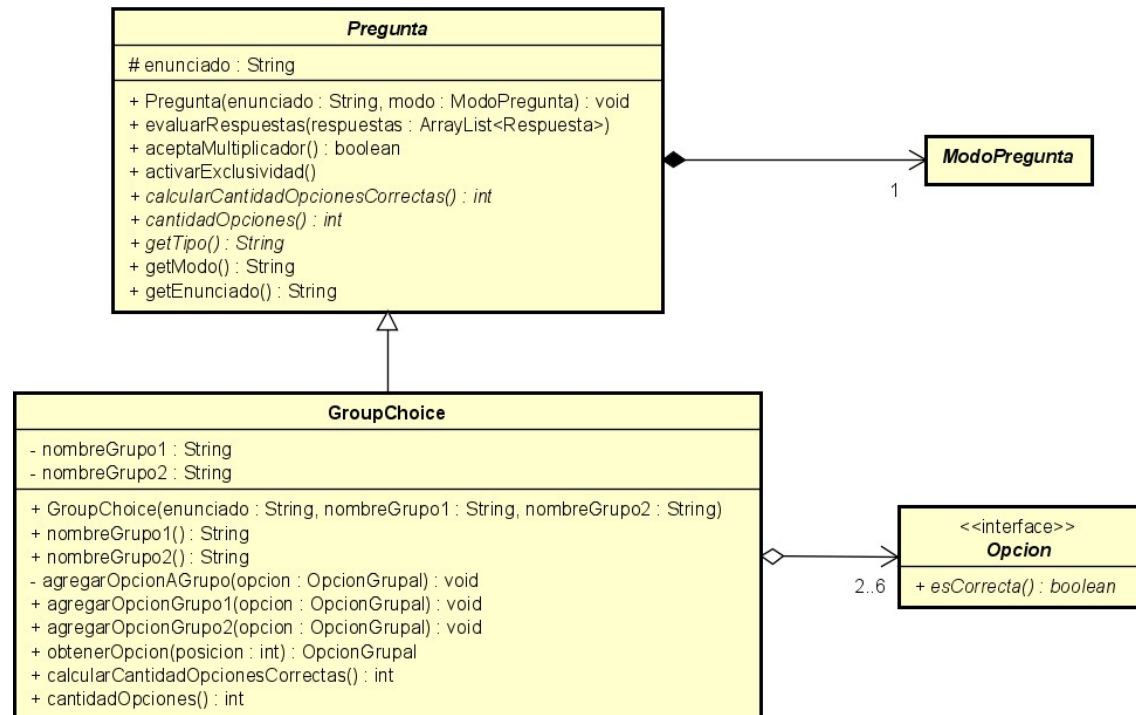


Figura 8: Diagrama específico de la Pregunta Verdadero/Falso (VoF) y Múltiple Choice

### 3.2.8. Modos de Pregunta

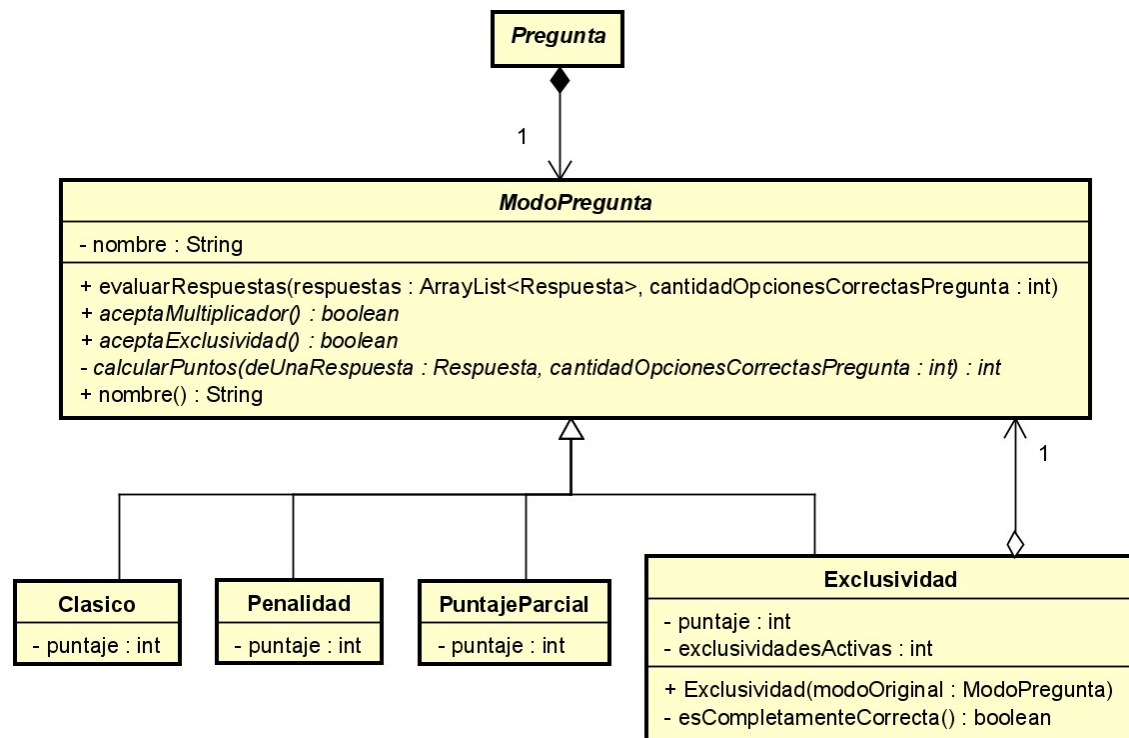


Figura 9: Diagrama específico de los Modos que puede tener la Pregunta

## 3.2.9. Respuestas

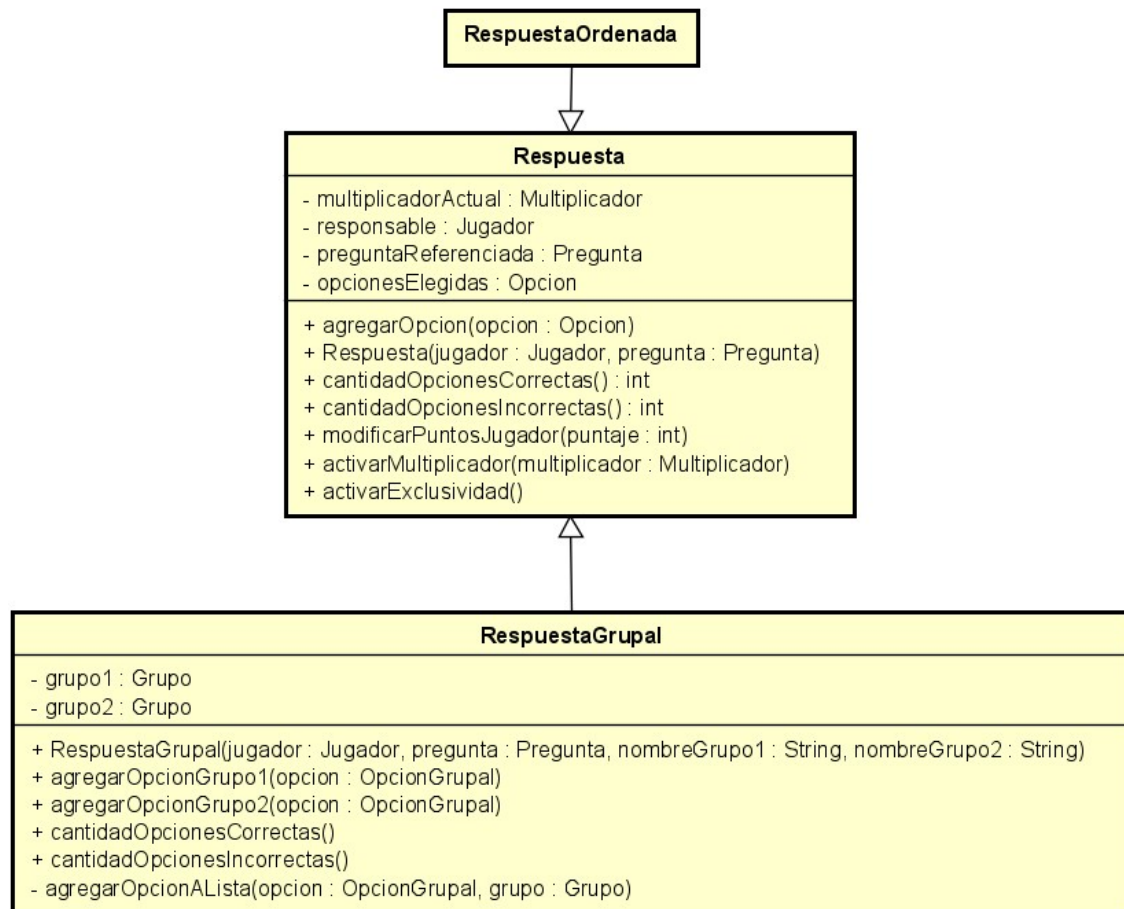


Figura 10: Diagrama de Clase de los diferentes tipos de Respuestas

## 3.2.10. Opciones

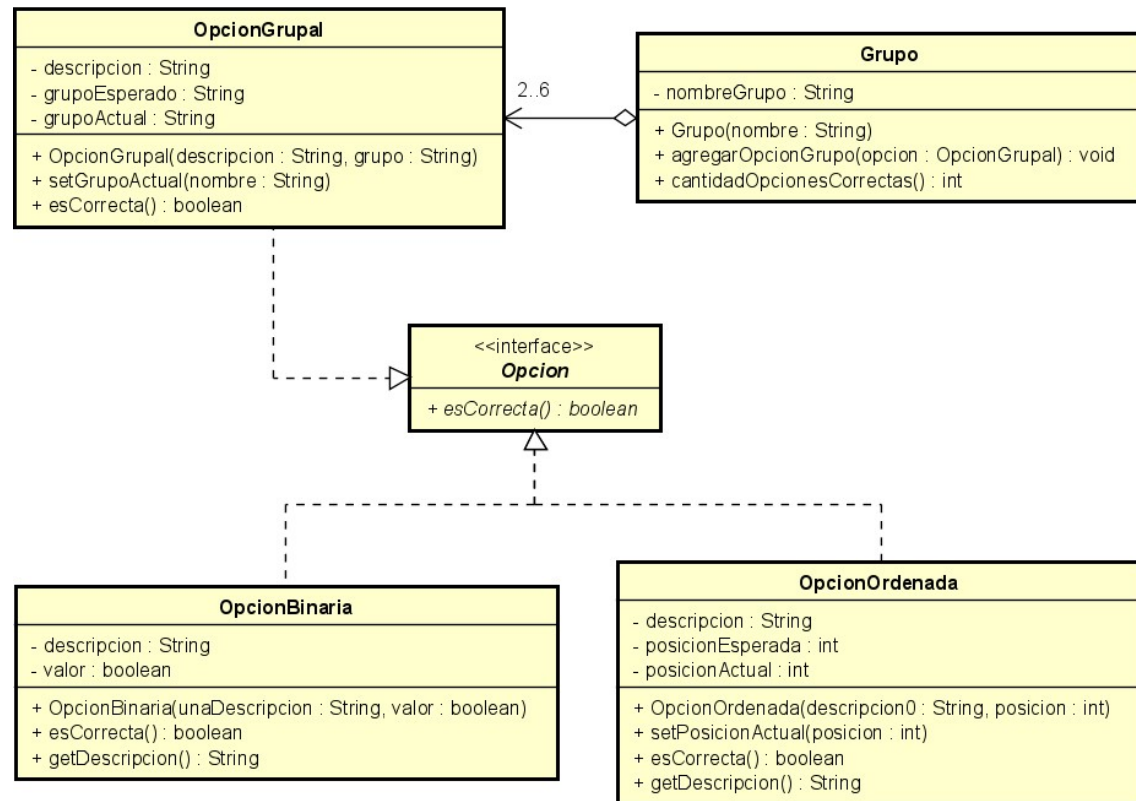


Figura 11: Diagrama de Clase de las diferentes Opciones

## 3.2.11. Juego

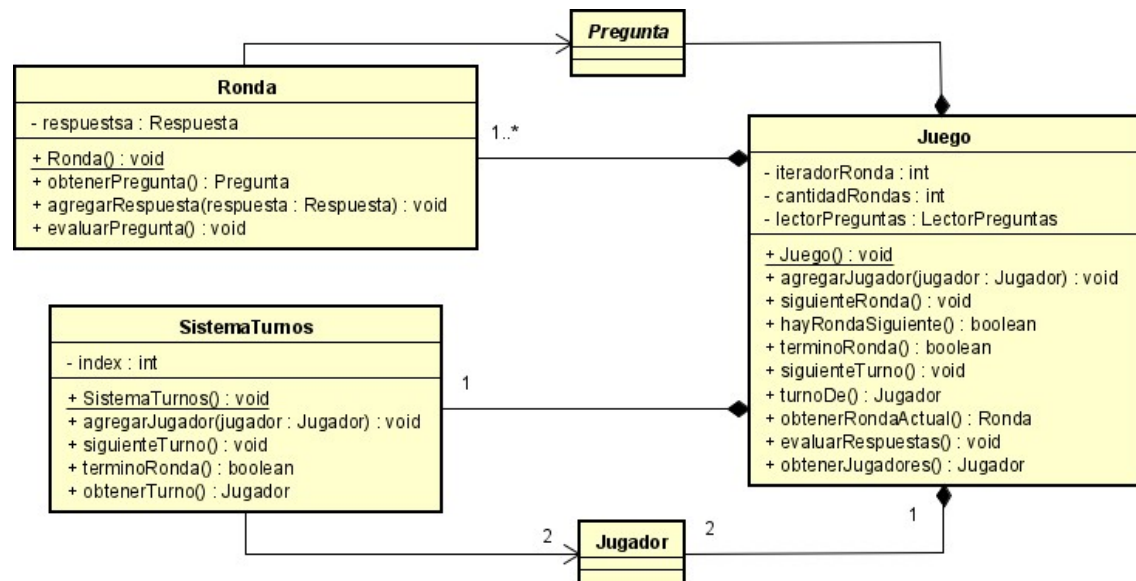


Figura 12: Diagrama de Clase para el juego con su sistema de turnos

## 4. Detalles de implementación

Se mostrarán distintos detalles sobre la implementación en los puntos mas importantes del juego.

### 4.1. Modos de pregunta

Esto está relacionado estrictamente con la pregunta que se decida crear. Como se sabe por consigna, uno puede crear principalmente 4 tipos de preguntas: Verdadero/Falso, Múltiple Choice, Ordered Choice y Group Choice. Pero a su vez, dependiendo del tipo de pregunta que se quiera crear, estas deben contar con un modo, donde dependiendo de este los puntos correspondientes a esa pregunta se calcularán de diferente manera.

Para solucionar este problema se decidió implementar como se ve en la figura 9 distintos tipos de modos, donde cada pregunta tendrá asociado un tipo. Los modos serán los encargados de calcular cuantos puntos hay que asignarle a cada jugador (sin tener en cuenta el multiplicador). Estos calcularan los puntos de distintas formas, por ejemplo:

- Modo clásico: se fijará que la cantidad de opciones correctas de la pregunta sean iguales a la cantidad de opciones correctas de la respuesta, además de que la respuesta no tenga errores. De ser así sumará puntos.
- Modo puntaje parcial: la condición para que una respuesta sume puntos en una pregunta con este modo sera que no contenga errores. Por lo tanto lo que se fijara el modo es que la cantidad de errores de la respuesta sean iguales a 0.
- Modo penalidad: este modo lo que hará será restar la cantidad de opciones correctas con la cantidad de opciones incorrectas que tenga la respuesta, y a partir de ahí multiplicar por el valor del puntaje de la pregunta.
- Modo exclusividad: por último se creó este modo para facilitar el tema de la exclusividad pero se explicará mas adelante.

La idea de esto que es que a la hora de crear la pregunta se cree con un modo ya que toda pregunta tiene un modo. Y una vez que la pregunta tenga cargada la lista de respuestas irá llamando al modo que tenga asignado por cada respuesta para que calcule cuantos puntos sumarle a cada jugador.

### 4.2. Multiplicadores y exclusividad

Los multiplicadores y la exclusividad se centra en modificar los puntos de una respuesta. Primero se explicará como se implementaron los multiplicadores y luego la exclusividad.

Para resolver el tema de los multiplicadores se optó por crear una clase Multiplicador que contenga un factor (o multiplicidad) que será lo que diferenciará de un multiplicador doble de un triple. Una respuesta, además de contener las opciones, contendrá un multiplicador actual, que inicialmente será nulo (factor=1). La idea es que cada jugador contenga una lista de multiplicadores la cual se completa con un multiplicador doble y con un multiplicador triple cuando se crea. Estos multiplicadores se activarán con los mensajes *activarMultiplicadorDoble* y *activarMultiplicadorTriple* y se removerán de lista. Cuando se active el multiplicador se le dirá a la respuesta que cambie su multiplicador actual por un multiplicador doble o triple según corresponda. De esta forma cuando el modo termine de calcular los puntos base, le dirá a la respuesta que modifique esos puntos base según el multiplicador que tenga y la respuesta a su vez se los modificará a los jugadores.

Con respecto a la exclusividad como se mostró anteriormente se creó un nuevo modo de pregunta. Esto es así ya que la exclusividad depende de las opciones correctas e incorrectas de ambas respuestas, algo que no hacen los demás modos.



La idea de este modo consiste en obtener cuantas respuestas son completamente correctas (sin errores y con misma cantidad de opciones correctas que la pregunta). Sabiendo esto la idea es que cuando un jugador quiera usar la exclusividad en una pregunta lo que se hará será reemplazar el modo de la pregunta original que tenía al modo exclusividad, guardando el modo original que tenía la pregunta en el modo exclusividad. En el fondo, es la esencia del patrón decorador. Conociendo esto, el modo exclusividad, solo sumará puntos si hay solo 1 respuesta completamente correcta, y si la hay, el modo original de la pregunta calculará los puntos bases y el modo exclusividad se encargará de modificar estos mismos por la cantidad de exclusividades que haya, modificándolo luego al jugador.

#### 4.2.1. Patrones de diseño usados

Se utilizaron los siguientes patrones:

- **Decorador:** se lo utilizó para asignar dinámicamente responsabilidad adicional al modo de la de pregunta a la hora de evaluar las respuestas de los jugadores cuando hayan activado la exclusividad, pues el activar exclusividad implicaba cambiar la forma de evaluar a todos los jugadores.
- **State:** se utilizó para que las diferentes preguntas existentes tengan un comportamiento diferente a la hora de evaluar respuestas de los jugadores. Esto se ve reflejado a la hora de pedir crear diferentes preguntas con distintos modos. En el fondo se termina delegando según el estado en el que fue inicializada la pregunta.

#### 4.2.2. Patrón de Arquitectura: MVC

El patrón MVC, Modelo-Vista-Controlador (MVC) un patrón de arquitectura de software utilizado para la construcción de interfaces de usuario para así lograr la separación de responsabilidades entre modelo, y las interfaces.

## 5. Diagramas de secuencia

A continuación se muestran algunos diagramas de secuencia para entender la relación dinámica que hay entre las distintas clases.

### 5.1. Creaciones de las diferentes Preguntas

#### 5.1.1. Creación: Pregunta Verdadero o Falso

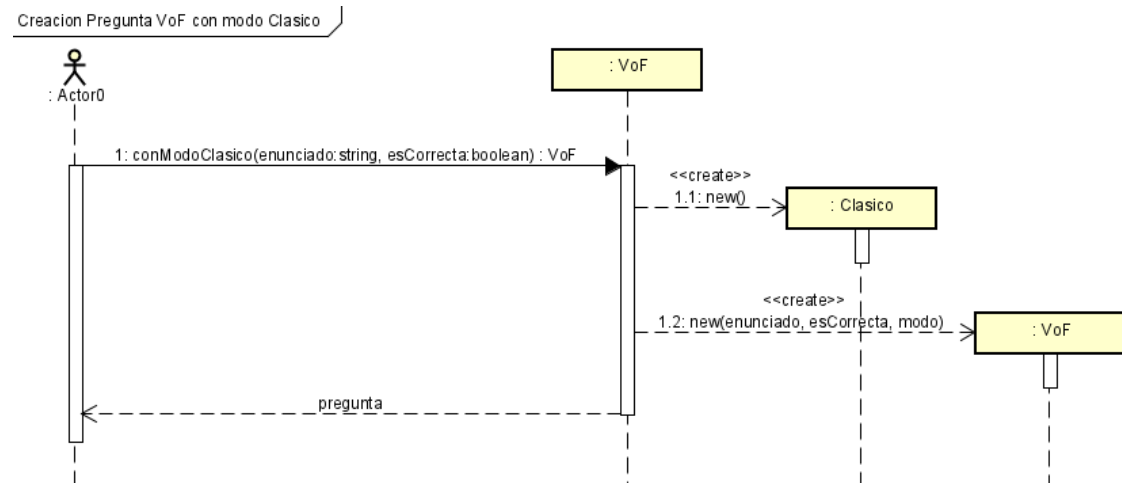


Figura 13: Diagrama de secuencia donde se crea una pregunta Verdadero o Falso en modo clásico.

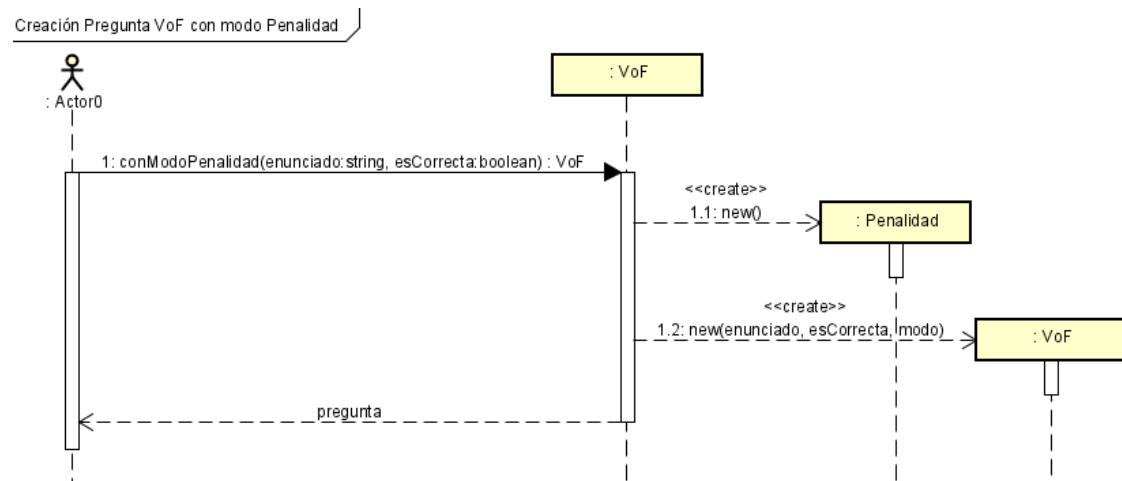


Figura 14: Diagrama de secuencia donde se crea una pregunta Verdadero o Falso en modo penalidad.

### 5.1.2. Creación: Pregunta Múltiple Choice

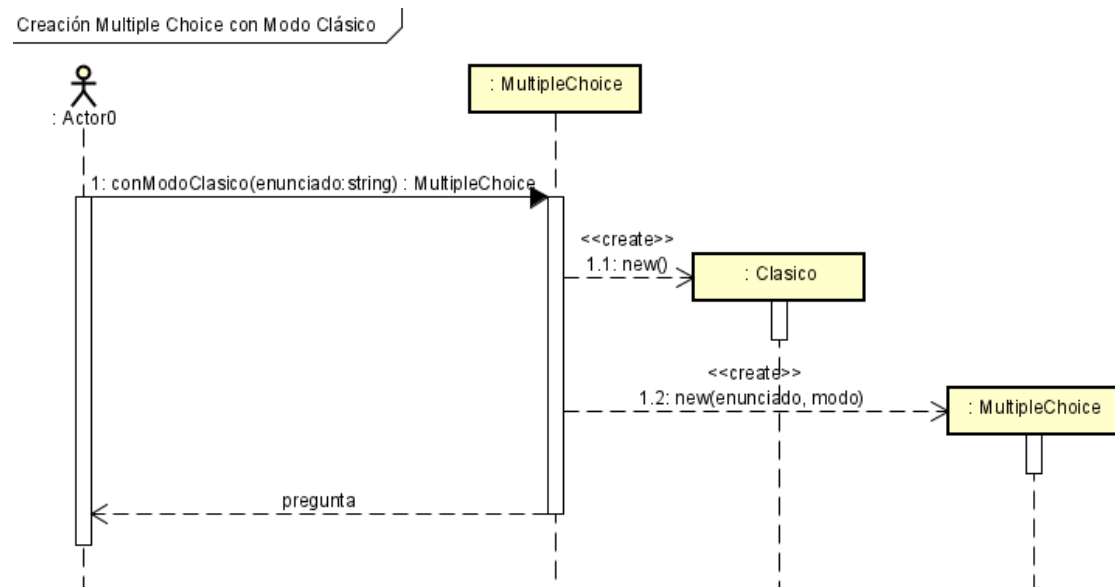


Figura 15: Diagrama de secuencia donde se crea una pregunta Múltiple Choice en modo clásico.

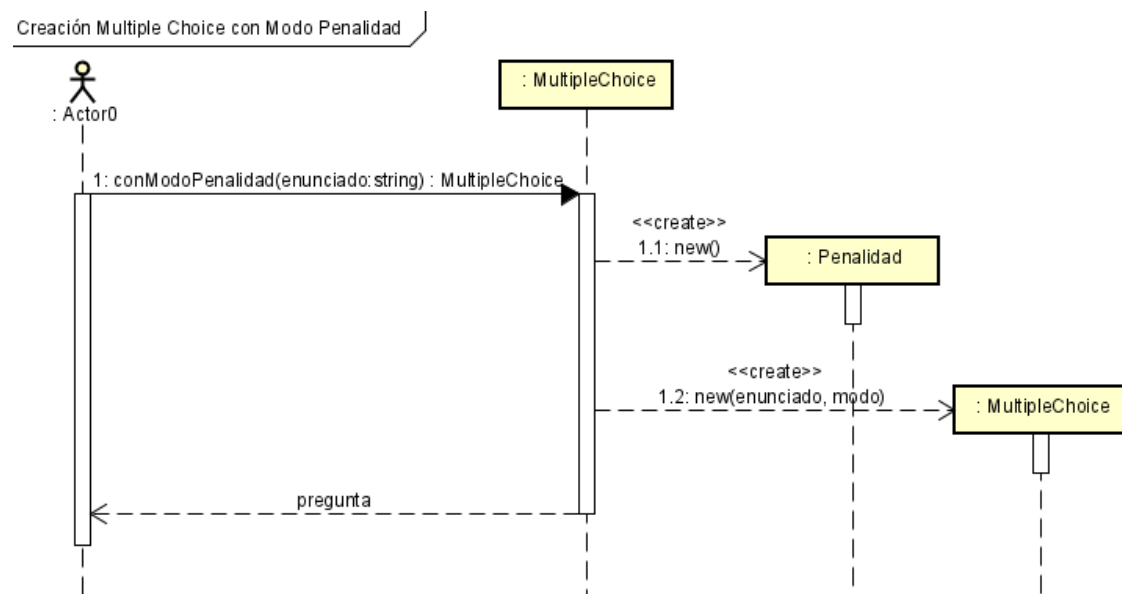


Figura 16: Diagrama de secuencia donde se crea una pregunta Múltiple Choice en modo penalidad.

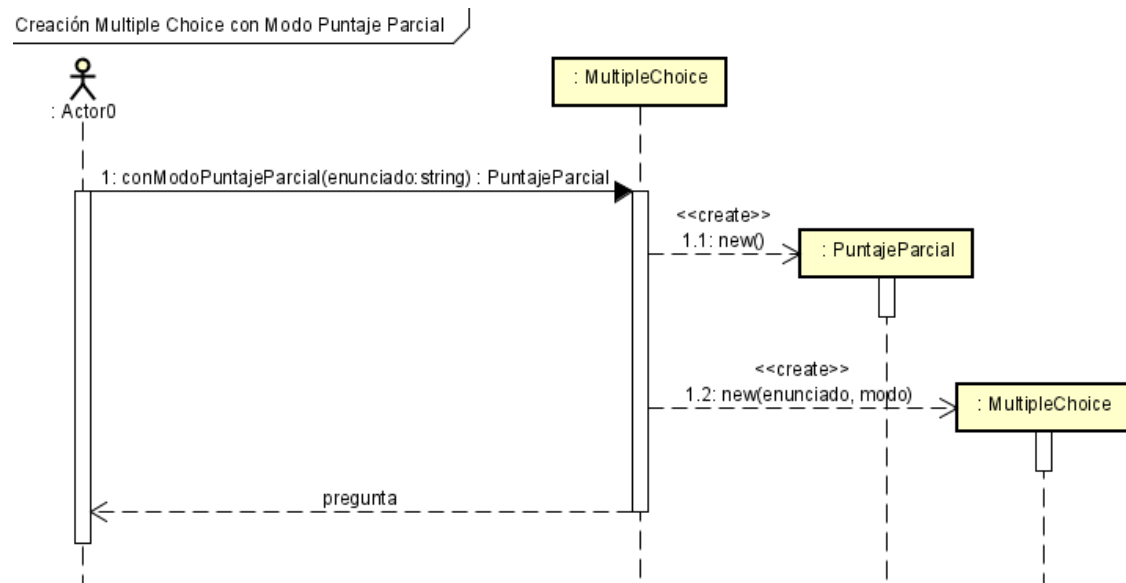


Figura 17: Diagrama de secuencia donde se crea una pregunta Multiple Choice con modo puntaje parcial.

### 5.1.3. Creación: Pregunta Ordered Choice

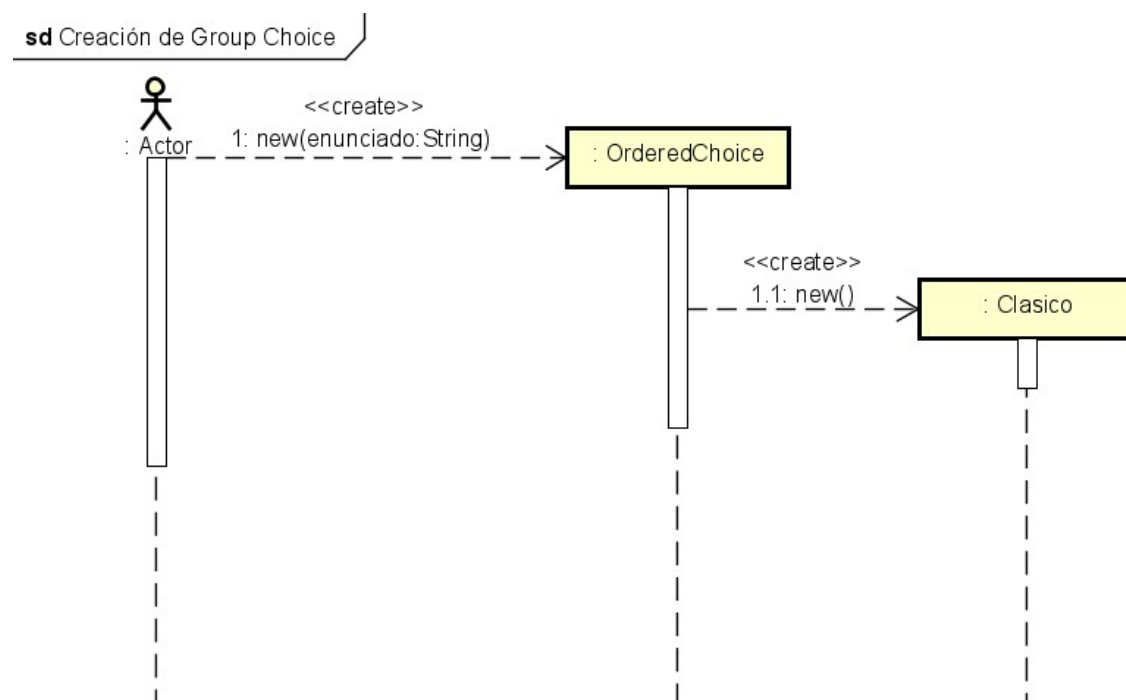


Figura 18: Diagrama de secuencia donde se crea una pregunta Ordered Choice.

#### 5.1.4. Creación: Pregunta Group Choice

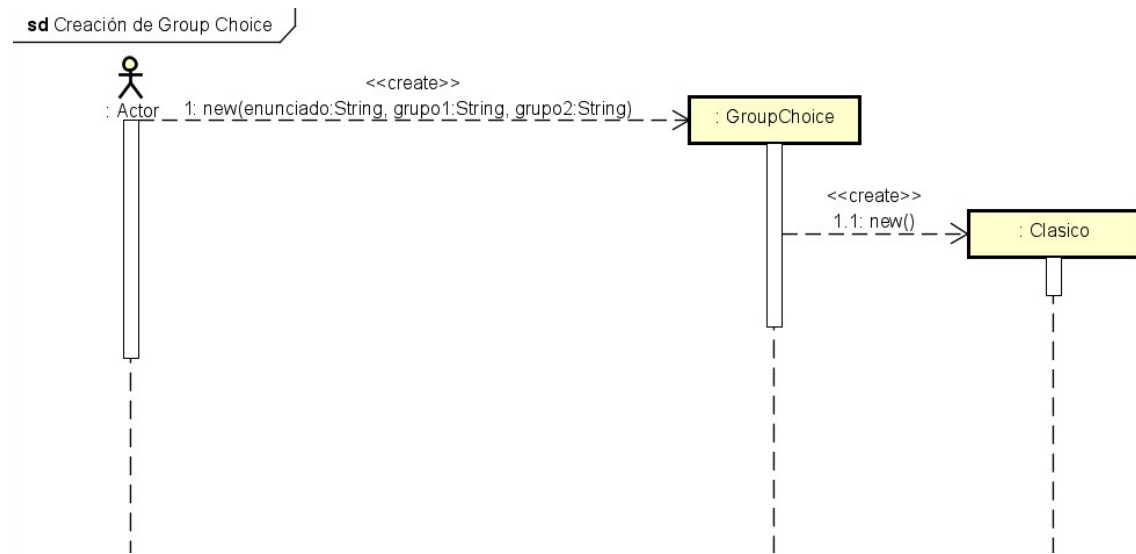


Figura 19: Diagrama de secuencia donde se crea una pregunta Group Choice.

## 5.1.5. Creación: Respuesta Group Choice

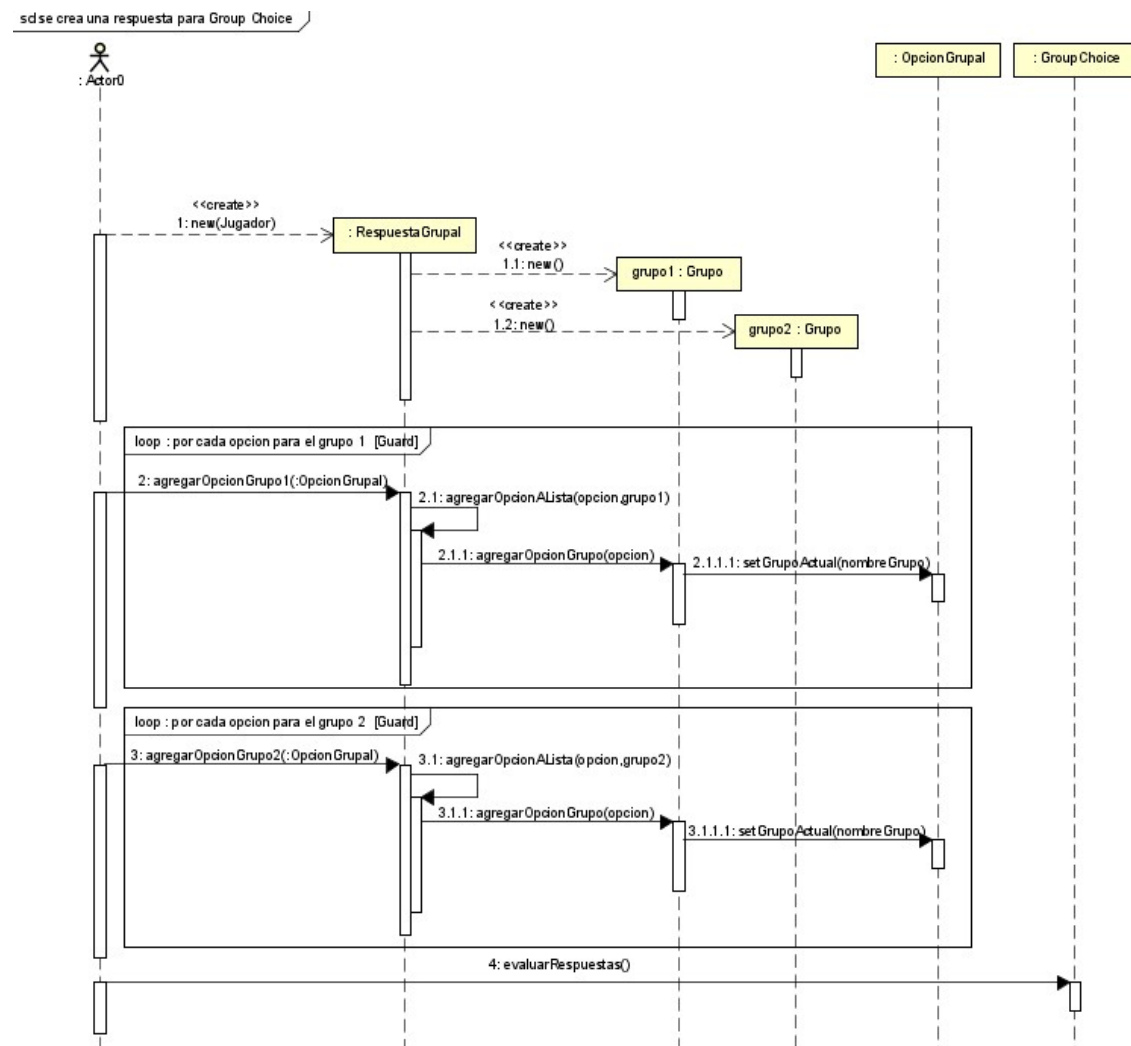


Figura 20: Diagrama de secuencia donde se crea una respuesta para Group Choice.

## 5.2. Evaluación para toda Pregunta

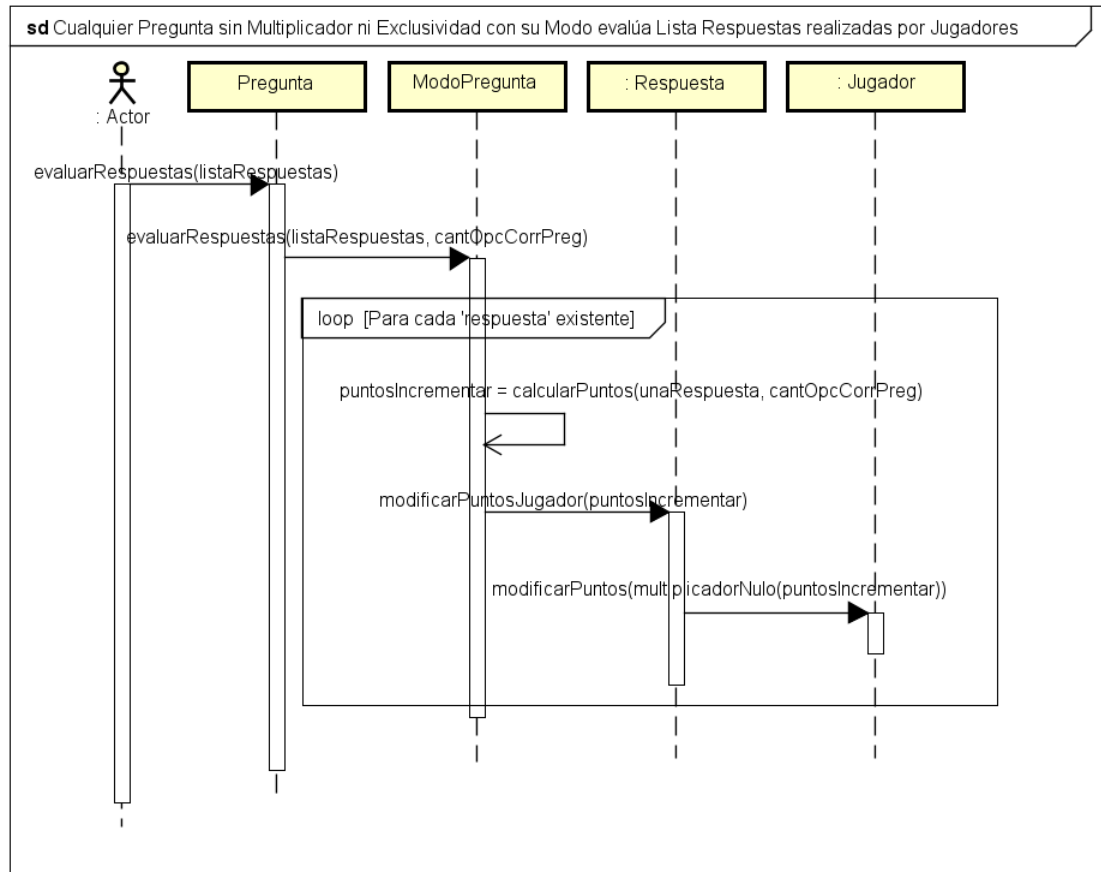


Figura 21: Diagrama de secuencia cualquier Pregunta evalúa una Lista de Respuestas

### 5.3. Obtención Calculo de Puntaje para cada Modo

Cuando una Pregunta evalúa una Lista de Respuestas: delega en su respectivo modo la Evaluación de dicha Lista. Los diferentes Modos internamente calculan la obtención de Puntos a Incrementar en el Jugador de diferente forma, que se la detallará en los ítem siguientes.

#### 5.3.1. Para Modo Clásico

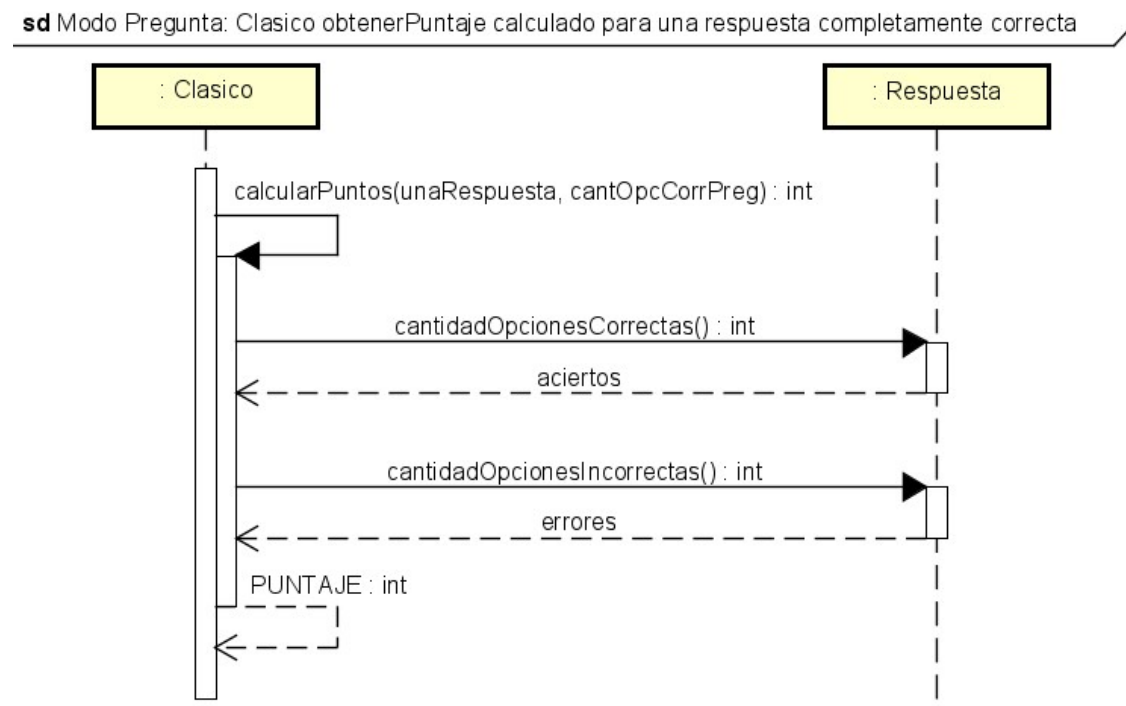


Figura 22: Obtención del Calculo de Puntaje para el Modo Clásico



## 5.3.2. Para Modo Penalizado

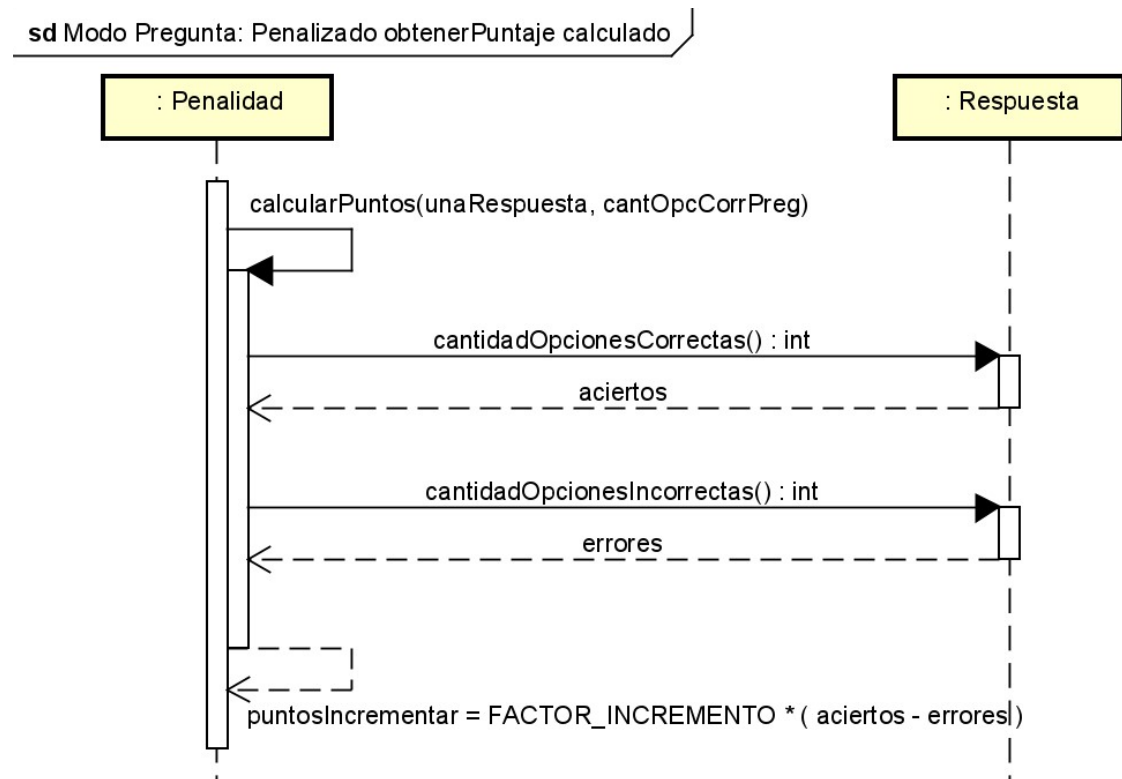


Figura 23: Obtención del Calculo de Puntaje para el Modo Penalizo

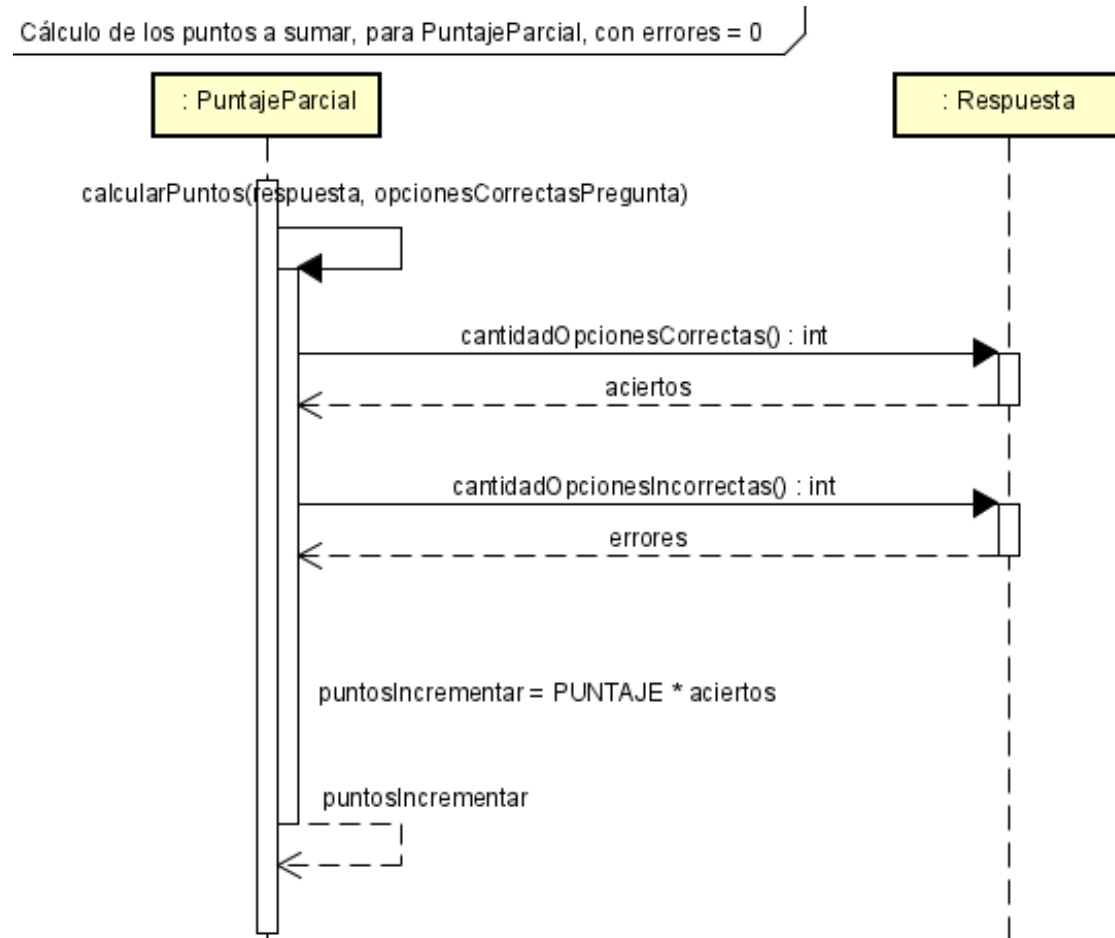
**5.3.3. Para Modo Puntaje Parcial**

Figura 24: Obtención del Calculo de Puntaje para el Modo Puntaje Parcial

## 5.4. Activar multiplicador doble y triple

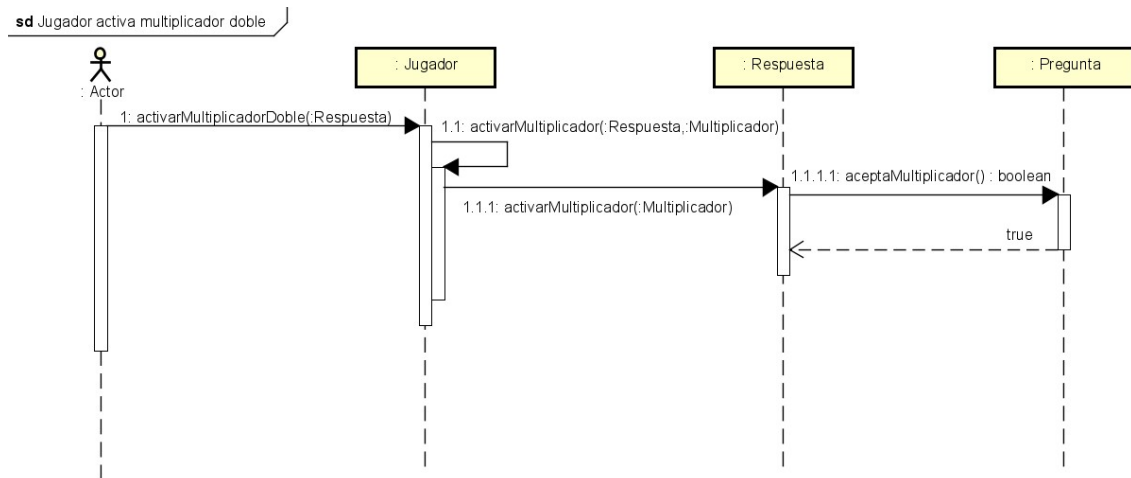


Figura 25: Activación del multiplicador doble para contestar una respuesta

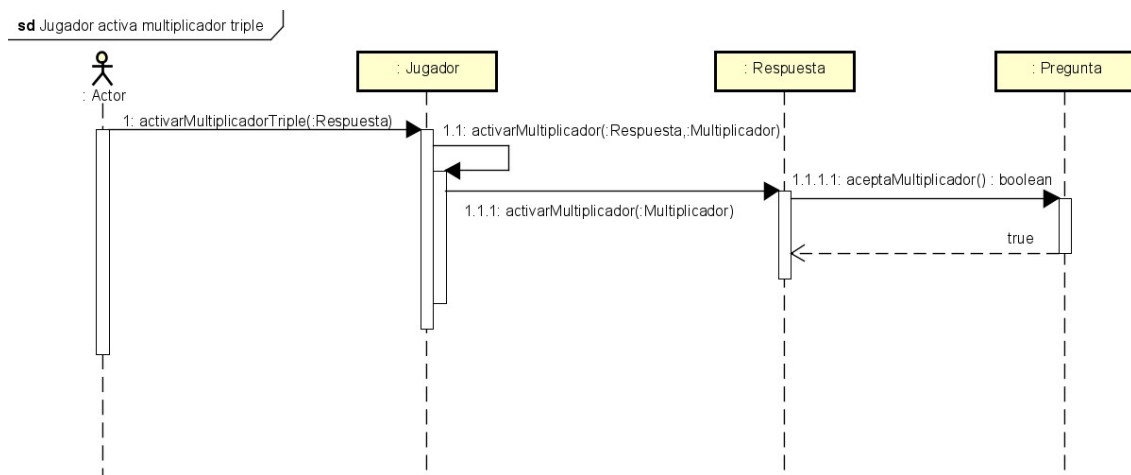


Figura 26: Activación del multiplicador triple para contestar una respuesta

## 5.5. Modo Pregunta con Exclusividades Activadas

Activar la Exclusividad implica agregar una nueva funcionalidad al Evaluar la Lista de Respuestas (el cual una Pregunta delegaba en su Modo). Eso implicó utilizar implícitamente el Patrón de Diseño **Decorador** cambiando ese Comportamiento de Evaluar la Lista Respuesta en los Modos Originales el cuales fueron creadas las Pregunta.

### 5.5.1. Con 1 Exclusividad activada

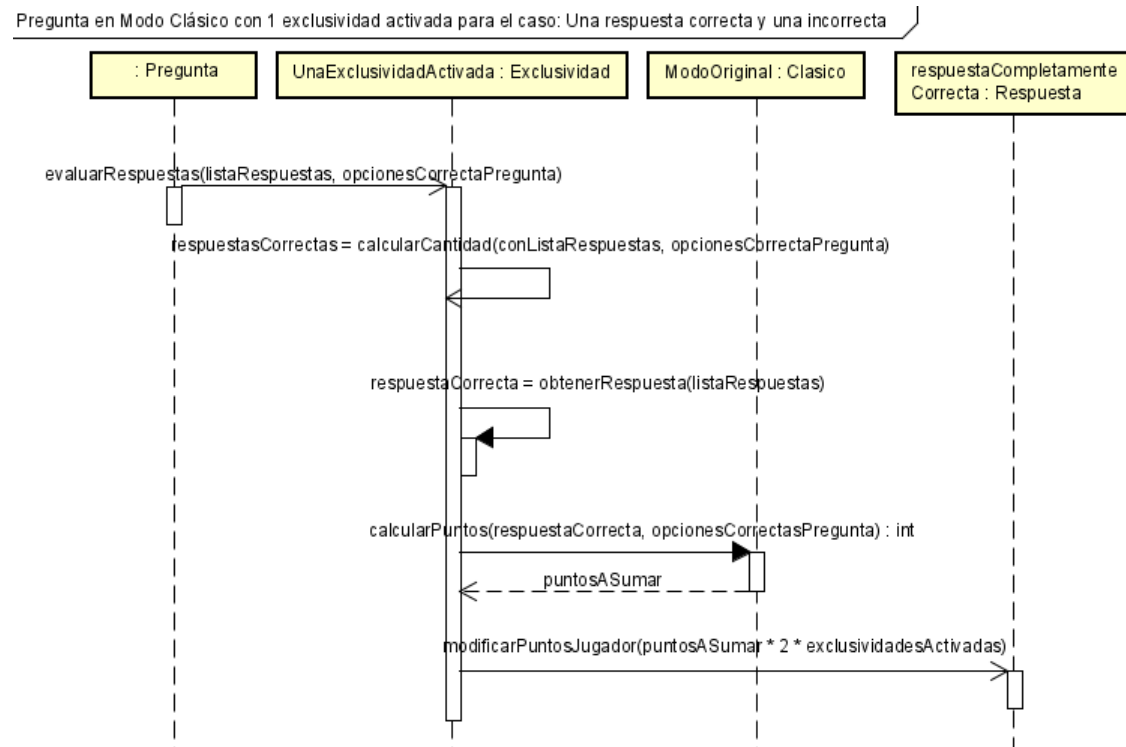


Figura 27: Se evalúa una Pregunta el cual se la ha activado 1 Exclusividad; delegando el Calculo de Puntaje en el Modo Original que almacena la Exclusividad al ser activada.

### 5.5.2. Con 2 Exclusividades activadas

Pregunta en Modo Clásico con 2 exclusividades activadas para el caso: Una respuesta correcta y una incorrecta

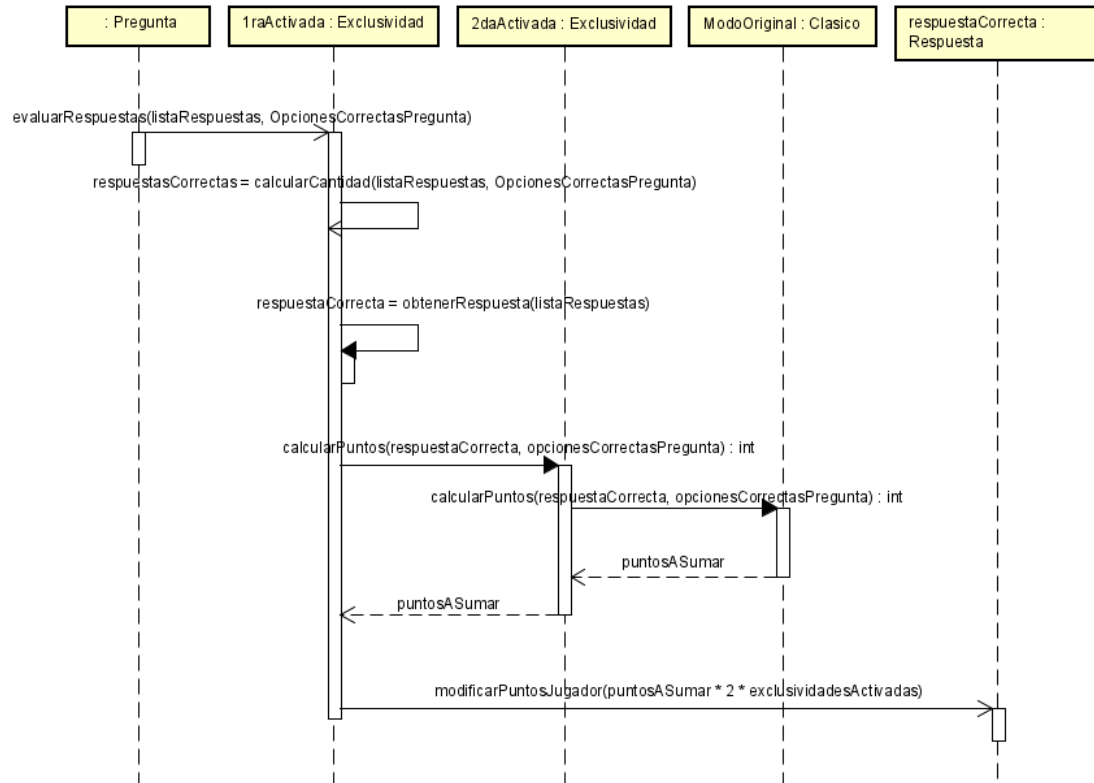


Figura 28: Se evalúa una Pregunta el cual se la ha activado 2 Exclusividades; delegando el Calculo de Puntaje en otra Exclusividad que almacena el Modo Original.

## 6. Excepciones

A continuación se mostraran y se explicaran las excepciones utilizadas:

- **"NombreVacioError"**: esta excepción se utiliza a la hora de realizar la carga de los nombres de los jugadores. Si al comienzo del juego cuando se pide que se ingresen los nombres en alguno de los dos campos no se introduce nada aparecerá un mensaje donde se nos pedirá que completemos los campos para poder comenzar a jugar.
- **"PreguntaSuperaMaxOpcionesError"**: Se hace uso de esta excepción cuando se tiene el máximo de opciones de una pregunta y se intenta agregar una más (la cantidad máxima de opciones varia con el tipo de pregunta)
- **"FormatoDesnocadoError"**: Esta excepción se utiliza en la recuperación de datos del archivo JSON, cuando se lee un campo desconocido.
- **"PreguntaNoAceptaMultiplicadorError"**: Esta excepción la podrá lanzar la Respuesta en el caso de que se le pida agregar multiplicador, y la pregunta que tenga asociada no acepte multiplicadores.
- **"PreguntaNoAceptaExclusividadError"**: Esta excepción la podrá lanzar la Pregunta en el caso de que se le pida que tenga una exclusividad activa, pero que su modo asociada no acepte exclusividad.
- **"MultiplicadorYaUtilizadoError"**: Por último esta excepción será lanzada por el jugador si este quiere utilizar algún multiplicador que ya fue utilizado. Pasará algo similar a estas ultimas dos excepciones con respecto a que si no esta disponible un multiplicador no se le mostrará al usuario.