**FIUBA - 75.07**

**Algoritmos y programación III**

*Trabajo práctico 2: AlgoFormers*

1er cuatrimestre, 2016

Alumnos:

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Padrón** | **Mail** |
| Carbón, Joaquín | 97.537 | joakoracing@hotmail.com |
| García Guarrochena, Carlos Ignacio | 87.299 | cignaciogarcia@gmail.com |
| La Penna, Mariano | 98.432 | mlapenna79@hotmail.com |

***Fecha de entrega final***: Jueves 23/06/2016 - Lunes 27/06/2016

***Tutor***: Gabriel Devoto

***Comentarios***:

4ta y última Entrega:

Trabajo Práctico completo funcionando, con interfaz gráfica final, sonidos e informe completo.

**Avance respecto a la entrega previa**

Se agrego la interfaz gráfica completa, con un tablero, los Algoformers de cada jugador y sus respectivas barras (de ataque, vida, distancia de ataque y velocidad), y los botones con las acciones que puede realizar el Algoformer.

En cuanto al modelo, a grandes rasgos, se agrego la clase modo, en la cual se delego el movimiento del Algoformer (Algoformer dejo de conocer a Tablero y Movimiento). También se agrego la clase turno, encargada de organizar el juego, a la cual conoce el Algoformer. Por último, se implemento la lógica de la victoria por obtener la chispa o matar a todos los Algoformers rivales.

Se agregaron nuevos tests para probar las modalidades y clases nuevas, y se modificaron los de las entregas pasadas para que acepten estos cambios en el modelo (por ejemplo constructores a los que se les pasan parámetros de nuevas clases).

Algunas modificaciones relevantes:

* Turno: encargada de la organización del juego.
* Excepciones nuevas para las modalidades agregadas.
* Modo: dividida en modo alterno, combinado y humanoide. Se encarga del movimiento y como es afectado por la superficie.
* Vista: (las más importantes)

**-TableroVista:** La vista tanto del Tablero como de las barras de los Algoformers.

**-CasilleroVista**: Setea las superficies y es la Vista de cada Casillero.

**-AlgoformerVista:** Da las imágenes a cada Algoformer y es la vista de los mismos.

**-JuegoVista:** Arma la ventana con los botones, contiene la lógica de crear el juego y llama a las otras vistas.

* Controladores: (los más importantes)

**-MouseClickHandler**: Contiene la lógica de la interacción del jugador y el juego mediante el “click”.

-**Juego:** Contiene el main.

**Supuestos del Trabajo Practico**

* Cuando se crea un Algoformer se crea en forma Humanoide
* Se carga el mapa ya creado en JSON. Con esto nos ahorramos la problemática de generar automáticamente un mapa.
* Se considera que hay dos jugadores y ninguno más.
* La velocidad de los algoformers indica la cantidad máxima de casilleros que se puede desplazar en un turno. En el enunciado solo muestra el caso en que se mueve la misma cantidad de casilleros pero no dice si se puede mover menos.
* Los movimientos que se pueden realizar son horizontales, verticales y diagonales, análogo a la “dama” o “reina” en ajedrez. No se permiten mezclas de ambos.
* Cuando las espinas reducen la vida de una unidad que lo atraviesa, definimos que el 5 por ciento se multiplique por cantidad de casilleros atravesados. O sea si por ejemplo una unidad con 100 de vida avanza desde (0, 0) hasta (3, 0) siendo los casilleros (1,0), (2,0) y (3,0) de superficie espinosa terrestre, la unidad termina el recorrido con 85 de vida.
* Al pasar por un pantano, se pierde el doble de tiempo, incluso si se termina el mismo, por ejemplo, si se realiza un movimiento de 4 casilleros en el que el primero es pantano y los otros no, igualmente se moverá dos casilleros.
* Al pasar por un pantano con una cantidad de casilleros es impar (por ejemplo 5), la cantidad de veces que se moverá (por la condición de que cueste el doble atravesarlo) se redondea para abajo (se mueve 2 casilleros).
* Para combinarse, los algoformers deben estar alineados verticalmente y sin espacio entre ellos, o sea, uno al lado de otro de forma consecutiva.
* Los algoformers combinados no podrán descombinarse.
* Cuando un algoformer en cualquiera de sus modos, es afectado por la tormenta psionica, el daño por el que es afectado en el modo actual, será el mismo que en el sigueinte.

**Informe**

Para realizar el juego utilizando el modelo orientado a objetos, creamos las clases necesarias, que interactúan tanto como es necesario para el buen funcionamiento. La clase principal es Algoformer, esta es abstacta y de ella heredan dos clases que son Autobot y Decepticon, de las cuales heredaran las clases de cada algoformer en particular, donde se redefinirá algún método necesario para diferenciarlos y sus características.

Para que el algoformer se mueva y la superficie que haya lo afecte, se utiliza la clase Movimiento, pero el algoformer no la conoce, solo conoce a la clase Modo, la cual tiene un atributo de la clase movimiento. De la clase Modo heredan ModoAlterno, ModoHumanoide y ModoCombinado, y la clase Movimiento a demás de dividirse por el modo en el que este el algoformer, también diferencia si es aéreo o terrestre.

Para la base del juego se utiliza la clase Tablero, la cual maneja un ArrayList de ArrayList<Casillero>, siendo Casillero la clase elemental que contiene las superficies aérea y terrestre(cada superficie es una clase que sabe como afectar al algoformer que pasa por ella) y en el Casillero se albergara un “Contenido”, este contenido es una clase de la que heredan Algoformer, Vacio y Chispa, y se utiliza como método de abstracción para que el tablero y el casillero no sepan que es lo que tienen. Como comentario a este párrafo, el tablero se guarda en un archivo JSON y luego levantara este archivo listo para usar.

Por último, las clases Juego y Jugador son las que crean el juego (“armar” el tablero, crear los algoformers, etc) y lo organizan.

Otras clases no principales son:

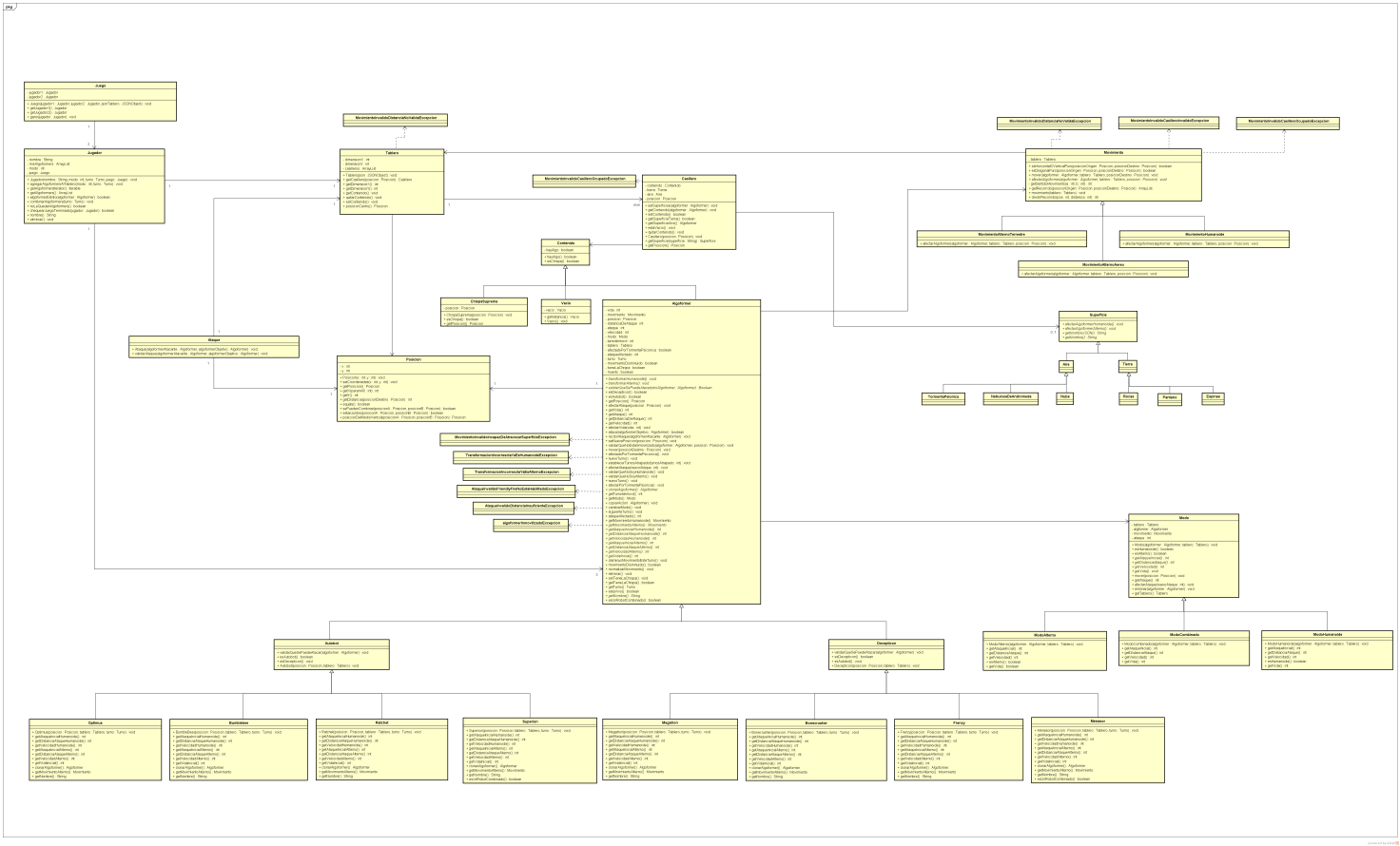
* Ataque: clase mediante la cual un algoformer ataca a otro.
* Posición: Ordena las coordenadas en un “vector”.
* Turno: Organiza el juego, también avisa cuando termino el mismo.
* Excepciones.

Por otra parte, la vista y los controladores son los que se encargan de la parte visual de juego y la interaccion con el usuario. Para esto utilizamos las clases:

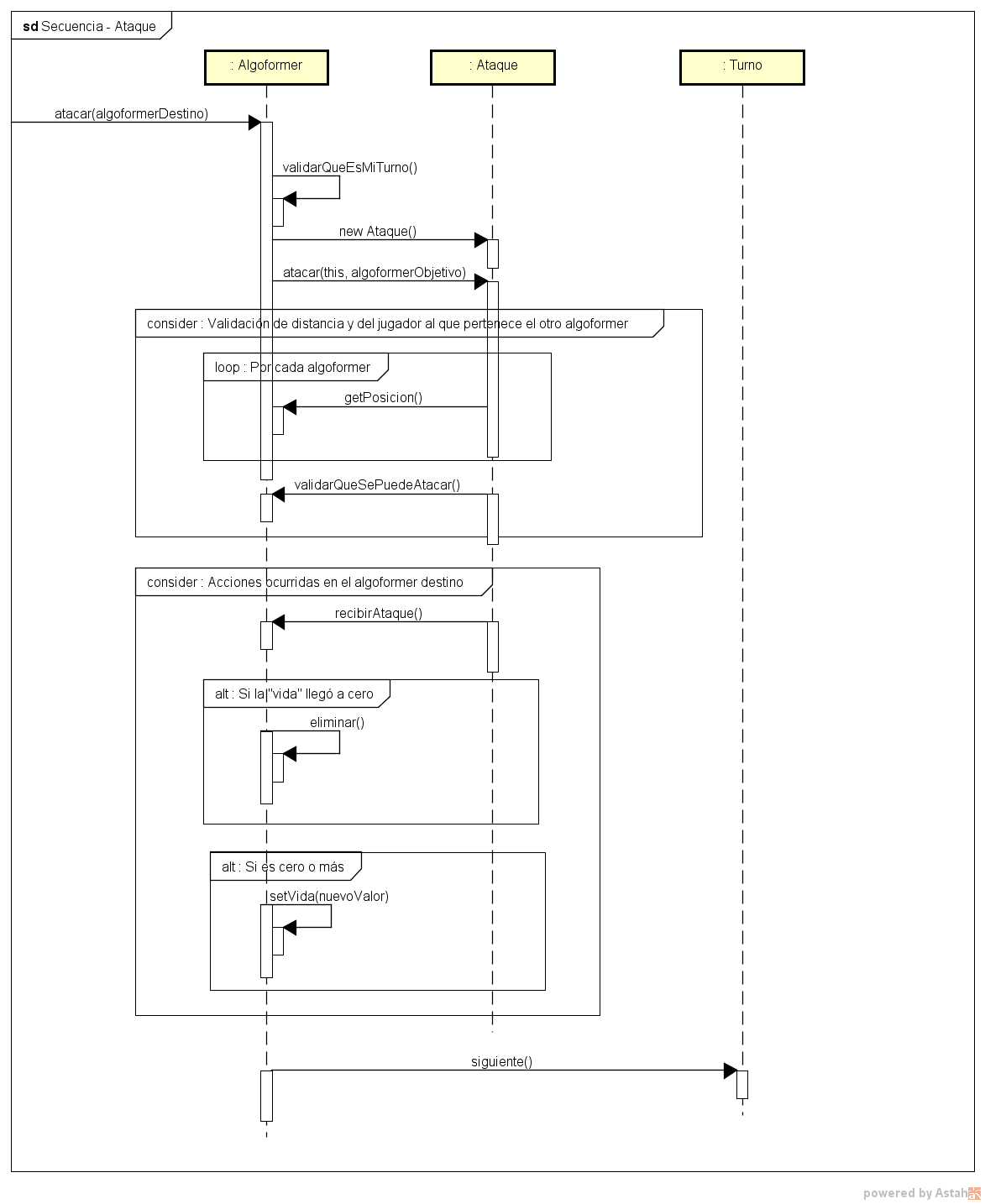
* JuegoVista: que arma la ventana (con los botones, el tablero, el nombre, etc).
* TableroVista: La vista tanto del Tablero como de las barras de los Algoformers.
* CasilleroVista: Setea las superficies y es la Vista de cada Casillero.
* AlgoformerVista: Da las imágenes a cada Algoformer y es la vista de los mismos.
* MouseClickHandler: Contiene la lógica de la interacción del jugador y el juego mediante el “click”.
* Juego: Contiene el main.

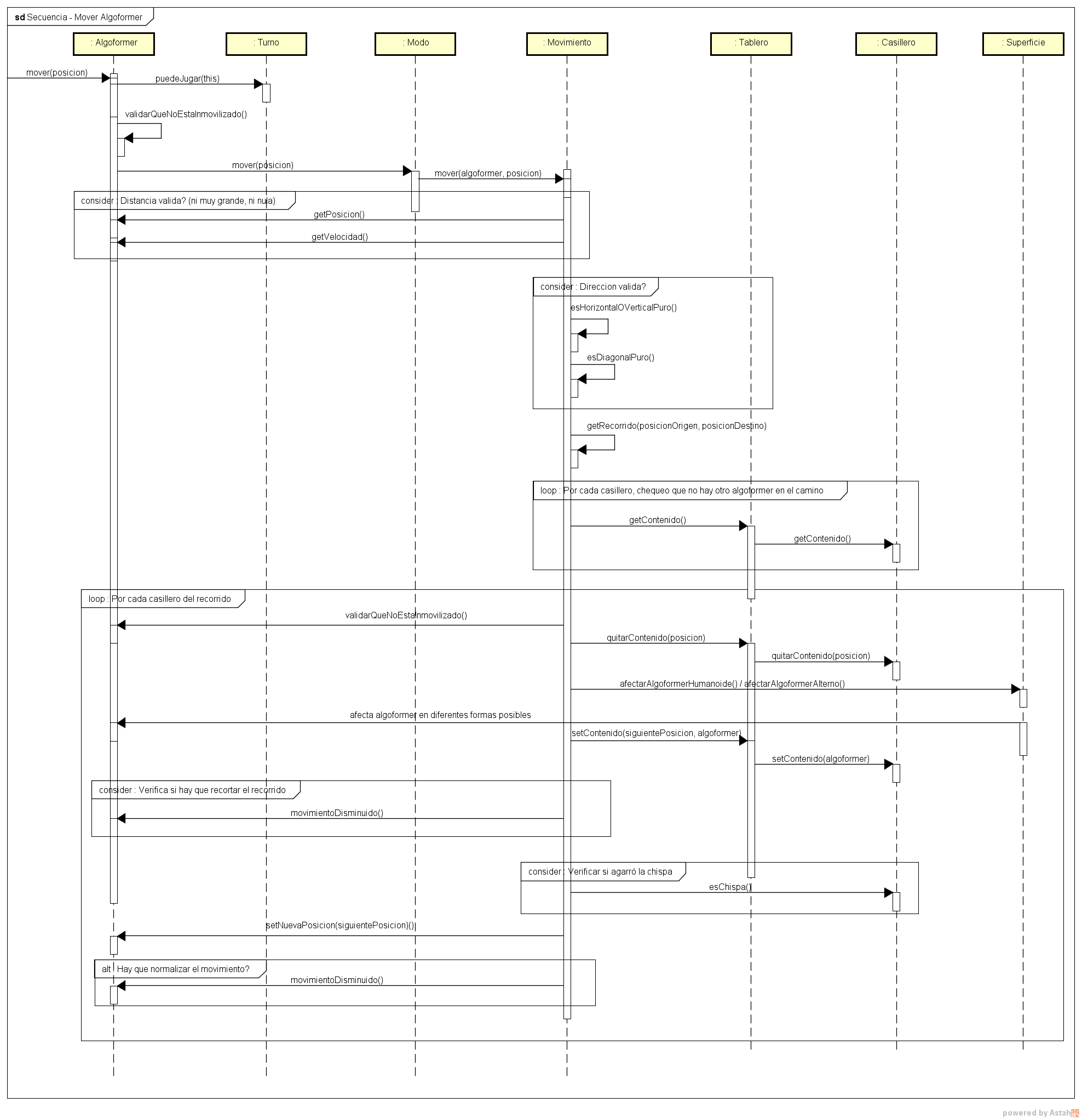
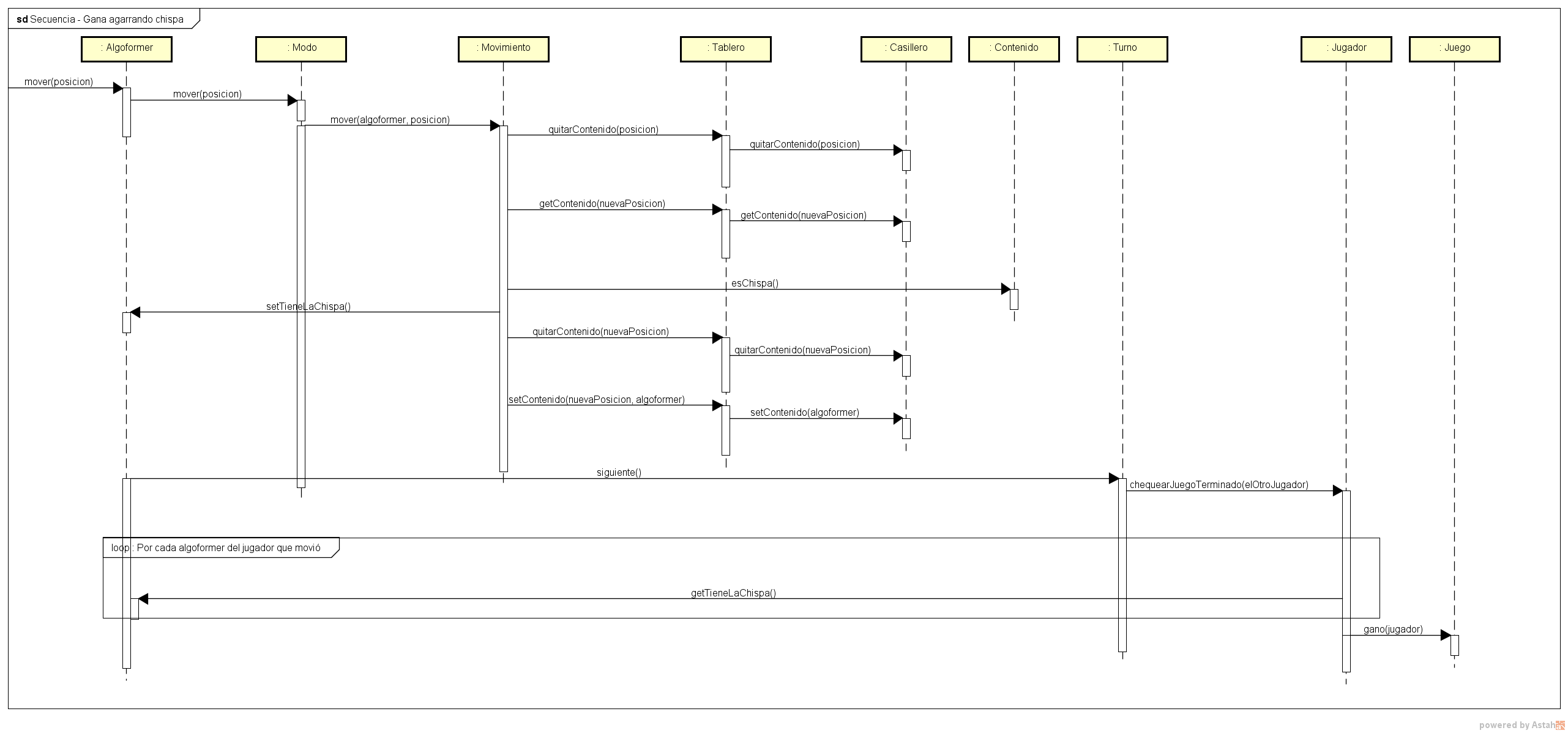
Otras clases de la vista y los controladores son BarraJugadorVista, ChispaSupremaVista, MostrarAlUsuarioError, AlertHandler, y las clases para cada botón (estas últimas, controladores).

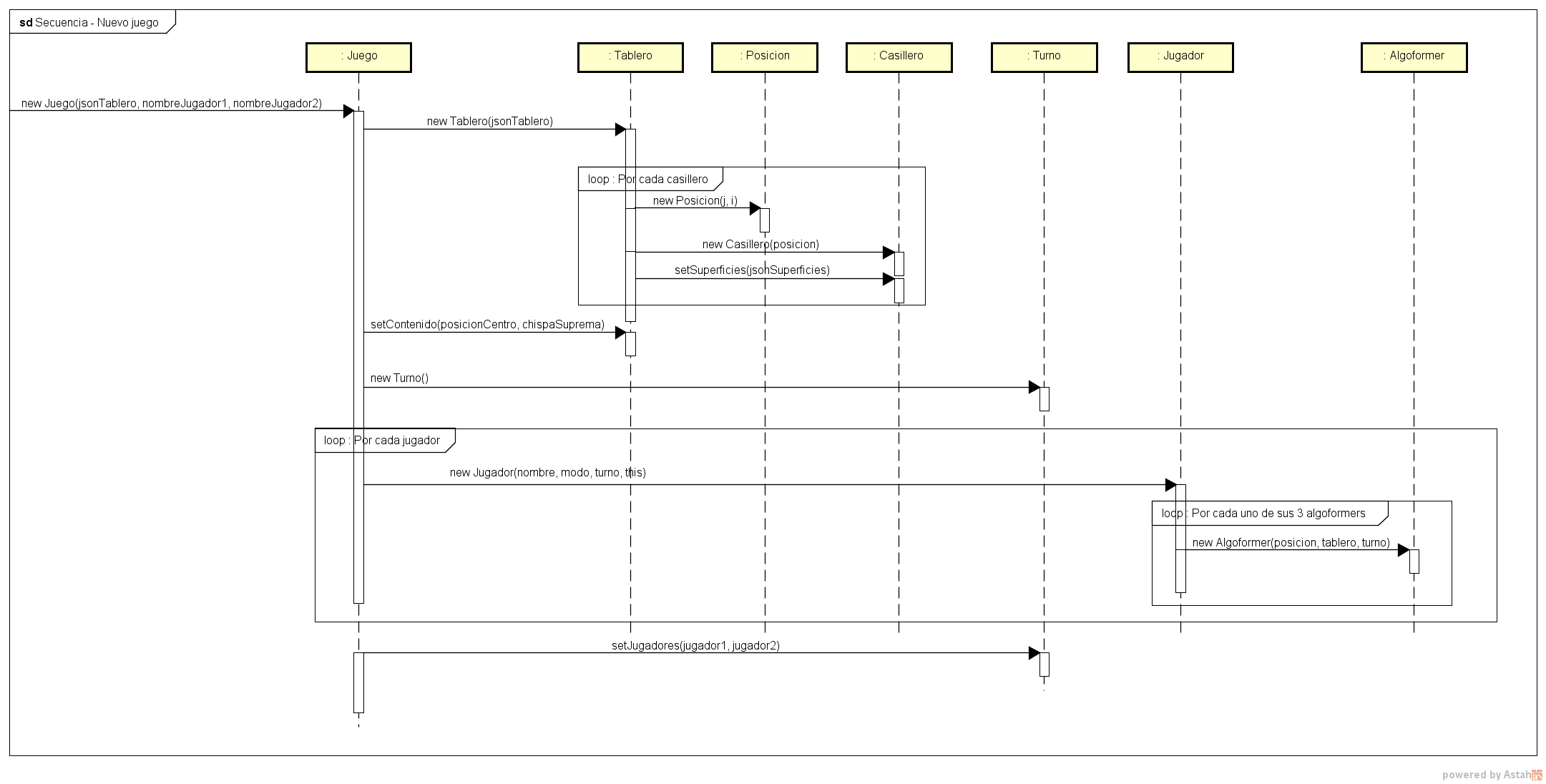
Diagrama de Clases

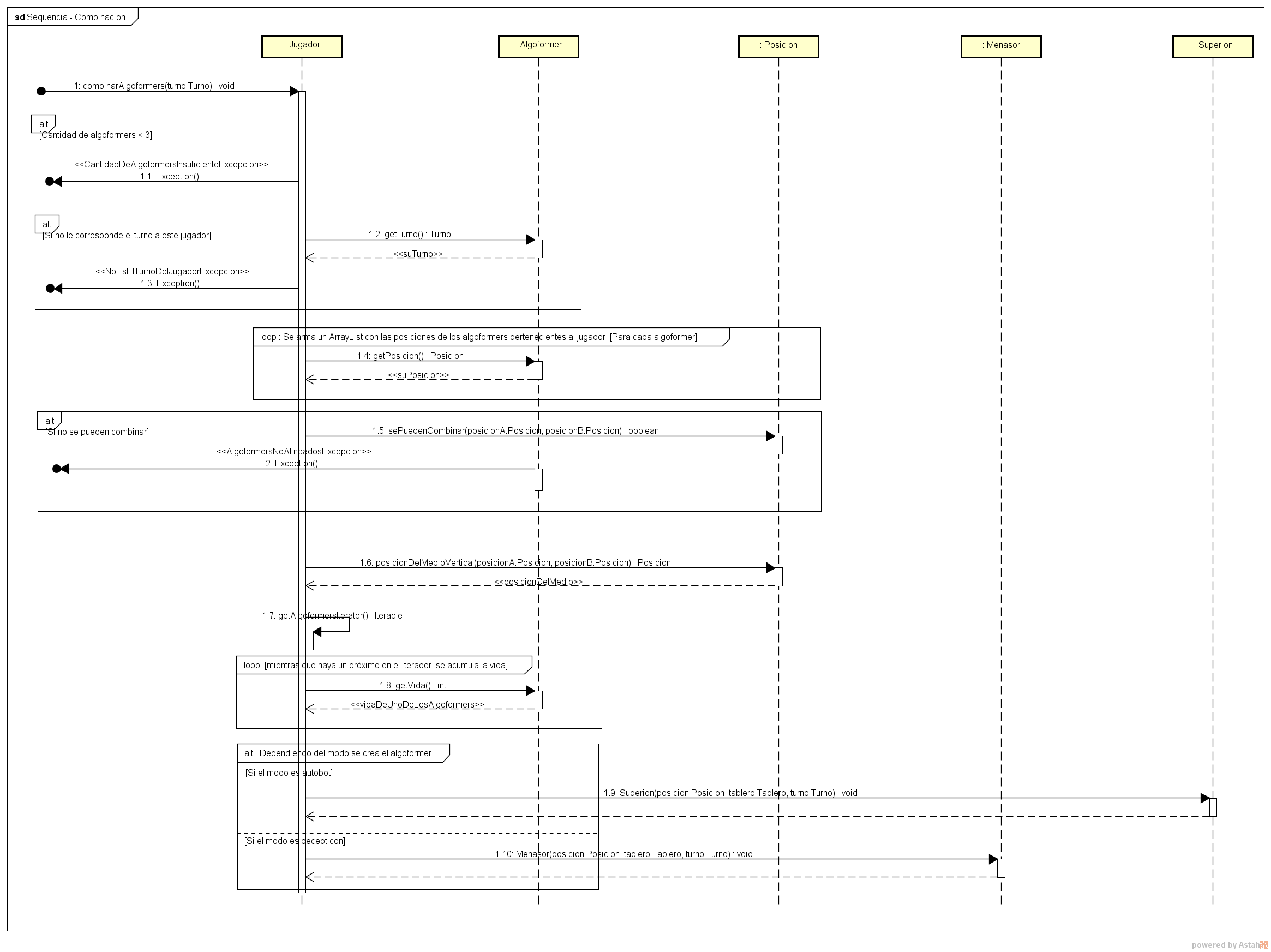


Diagramas de Secuencia









**Salida de ejecución de ANT**