

## Trabajo Práctico 2 — Java

[7507/9502] Algoritmos y Programación III  
Curso 1  
Segundo cuatrimestre de 2019

Alumno:	BUCETA , M. Belén
Número de padrón:	102121
Email:	bucetabmb@gmail.com
Alumno:	DI COMO, Juan Pablo
Número de padrón:	123456
Email:	jpgdico@fi.uba.ar
Alumno:	PRATTO, Federico
Número de padrón:	96223
Email:	fpratto@fi.uba.ar

## Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Supuestos</b>	<b>2</b>
<b>3. Modelo de dominio</b>	<b>3</b>
<b>4. Diagramas de clase</b>	<b>4</b>
<b>5. Detalles de implementación</b>	<b>4</b>
5.1. . . . . .	4
5.2. . . . . .	5
<b>6. Excepciones</b>	<b>5</b>
<b>7. Diagramas de secuencia</b>	<b>6</b>
<b>8. Diagramas de estado</b>	<b>8</b>

## 1. Introducción

El presente informe reúne la documentación de la solución del segundo trabajo práctico de la materia Algoritmos y Programación III que consiste en desarrollar una aplicación de un videojuego 'AlgoChess' en Java con un diseño del modelo orientado a objetos y trabajando con las técnicas de TDD e Integración Continua.

## 2. Supuestos

**Batallón:** Al seleccionar un Soldado para moverlo, el Tablero se fija si tiene dos soldados que sean vecinos inmediatos, es decir, si se ubican a un casillero de distancia en alguna dirección (formando una recta entre los tres).

Una importante observación es resaltar que el hecho de que ocurra un batallón no implica que éste vaya a moverse. Para ello debe darse que puedan hacerlo en las direcciones correspondientes a cada uno de sus respectivos desplazamientos.

Otra observación: solo se permite un batallón por turno. Es significa que aunque en el turno se halle la posibilidad de formar más de un batallón en distintos sentidos, el batallón que se concretará será aquel que primero se encuentre y ningún otro más se formará en dicho turno.

Esto es un batallón

S	S	S
---	---	---

Esto no es un batallón.

S	S	S
---	---	---

Si selecciono el Soldado del medio se habrá formado un batallón, en cambio si selecciono a cualquiera de sus vecinos, como estos solo tiene un soldado al lado, no actuaran como batallón.

S	S	S
S	S	S
S	S	S

Seleccionar cualquiera de estos soldados daría como resultado un batallón.

Figura 1: . Formación de un Batallón

Por otro lado, el sentido en el que se realiza la búsqueda de un batallón se determinó en el siguiente orden. Primero, en sentido vertical, luego horizontal, siguiendo con creciente en dirección noreste y finalmente creciente en diagonal noroeste. Una vez hallado un batallón se detiene la búsqueda. Para mayor comprensión se adjunta un gráfico con cada caso.

Para detectar un batallón al elegir un soldado, el tablero trabaja de la siguiente forma:

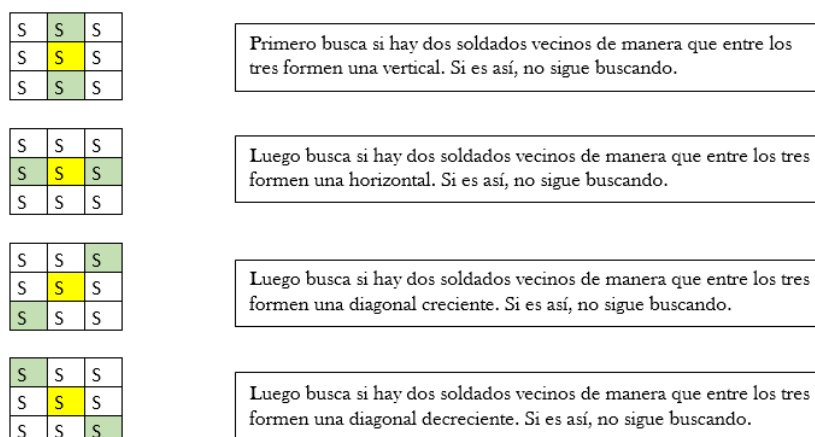


Figura 2: . Detección de un Batallón

■ Supuestos del movimiento de un Batallón:

Si se ha encontrado un batallón, el tablero intentará mover las tres unidades en la dirección solicitada, si no logra mover ninguna pieza debido a que todas están bloqueadas lanzará un error.

### 3. Modelo de dominio

El video juego está diseñado con la siguiente lógica.

Donec fermentum volutpat diam, non sodales dui pulvinar nec. Vestibulum eu nisl lacus. Maecenas vel tortor efficitur, volutpat nulla id, ornare sem. Quisque nisi magna, vulputate sed purus eget, elementum volutpat velit. Suspendisse sit amet feugiat nulla. Vivamus posuere sit amet diam condimentum sagittis. Sed et sapien in purus mattis ullamcorper sit amet non massa. Suspendisse tempus eleifend dapibus.

Ut commodo urna a diam gravida suscipit. In pretium tellus vitae ligula tempus, eu hendrerit eros consequat. Donec scelerisque fermentum sem, vitae pretium augue sollicitudin ac. Cras condimentum eros risus, eu luctus nibh consequat id. Vivamus et vestibulum turpis. Sed luctus ipsum id dui tristique mollis. Cras mattis eget lorem eget posuere.

Aenean imperdiet, tellus non semper dictum, urna ipsum porta mauris, id tincidunt massa mi vitae turpis. Mauris et tortor id ex egestas porta. Aenean laoreet lorem orci, non volutpat nulla volutpat ac. In eget lectus iaculis, tempus lectus id, commodo ligula. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Suspendisse facilisis, ante non interdum rhoncus, felis diam malesuada est, a lacinia dolor nisl at ante. Nulla nisl libero, imperdiet ac libero eget, commodo maximus leo. Suspendisse ut est tortor. Nam tempus sollicitudin elit, et volutpat arcu iaculis ut. Vestibulum lacus tortor, efficitur in purus vitae, luctus tristique risus. Integer pellentesque dictum finibus. Proin at cursus libero.

## 4. Diagramas de clase

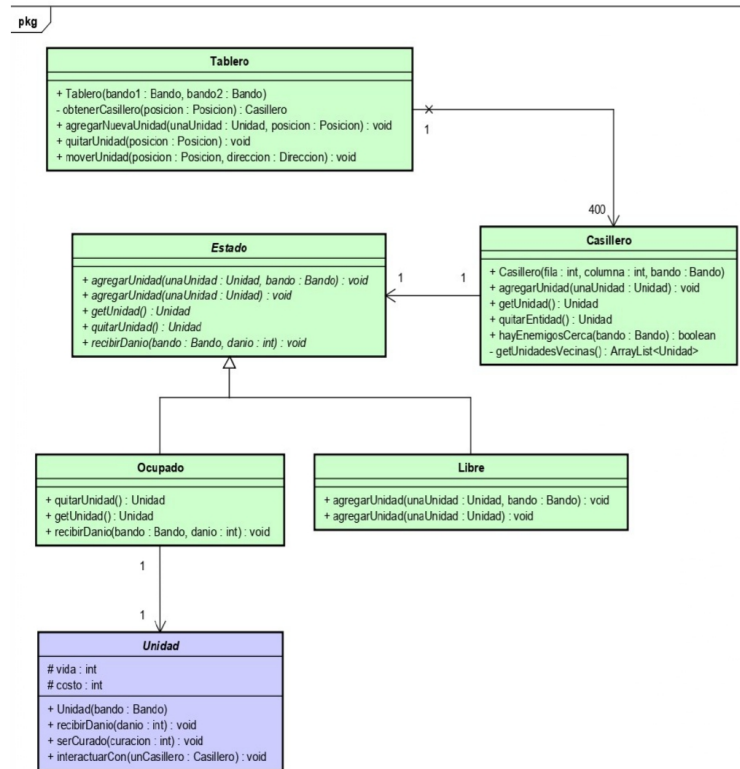


Figura 3: Diagrama del Tablero.

## 5. Detalles de implementación

### 5.1.

```

| rango |
rango := (2 to: 20) asOrderedCollection.
Transcript show: rango ; cr.
rango copy do: [ :unNumero | unNumero isPrime ifFalse: [ rango remove: unNumero ] ].

```

Transcript show: rango.

## 5.2.

Quisque tempus, tortor et convallis interdum, ipsum leo tempus ipsum, in molestie tortor arcu sit amet tellus. Praesent fermentum hendrerit nulla. In maximus ornare maximus. Nullam consectetur placerat enim sit amet lacinia. Etiam pellentesque tellus consectetur hendrerit iaculis. Sed non laoreet felis.

## 6. Excepciones

**CasilleroLibreExcepcion** Sus fines de creación fueron atrapar las siguientes acciones inválidas:

- Se busca quitar una unidad de un casillero libre.
- Se busca obtener unidades vecinas de un casillero vacío, hecho que carece de sentido en la lógica planteada del juego.
- En el caso de la búsqueda de formación de un batallón, la excepción surge como validación. Es decir, si en alguno de los sentidos (horizontal, vertical, etc) en los que se busca un batallón hay un casillero libre, entonces allí no hay un batallón.
- Si se intenta atacar un casillero vacío al realizar un ataque en área.

**CasilleroOcupadoExcepcion** Sus fines de creación fueron atrapar las siguientes acciones inválidas:

- Se busca agregar una unidad a un casillero ocupado.
- Se intenta mover a algún integrante de un batallón a un casillero ocupado.

**CatapultaNoPuedeSerCuradaExcepcion** Se intenta curar una catapulta cuando, según las reglas del juego, eso no es posible.

**DesplazamientoInvalidoExcepcion** Sus fines de creación fueron atrapar las siguientes acciones inválidas:

- Se intenta realizar un desplazamiento fuera del rango posible.

**DistanciaInvalidaExcepcion** Si la distancia entre dos casilleros supera o es inferior a los rangos de distancia máxima y mínima respectivamente.

**DistintoBandoExcepcion** Se utiliza para detectar si la unidad está interactuando con un enemigo.

**MismaUnidadExcepcion** Se utiliza para detectar si la unidad está interactuando con ella misma.

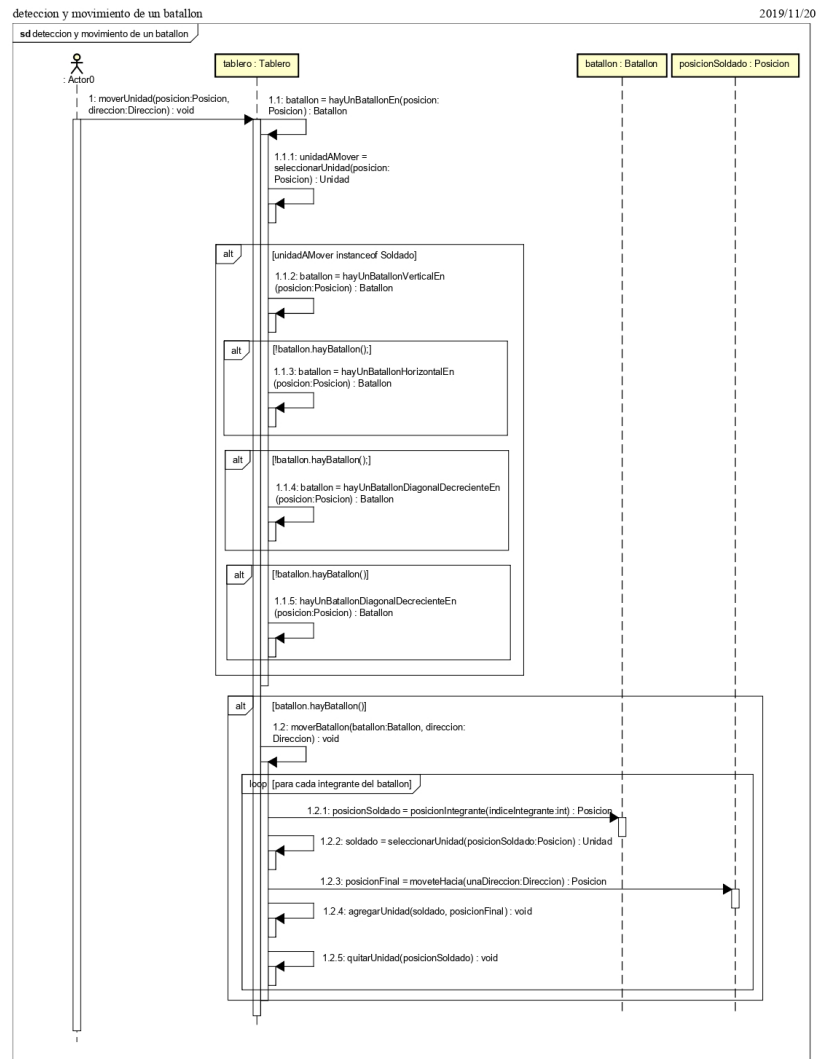
**MismoBandoExcepcion** Se utiliza para detectar si la unidad está interactuando con una de su mismo bando.

**PosicionInvalidaExcepcion** NO ENCONTRÉ MOTIVO ESPECIFICO.

**PuntosInsuficientesExcepcion** El jugador intenta agregar una unidad sin los puntos suficientes para hacerlo.

## 7. Diagramas de secuencia

Deteccion y movimiento de un batallon.jpg

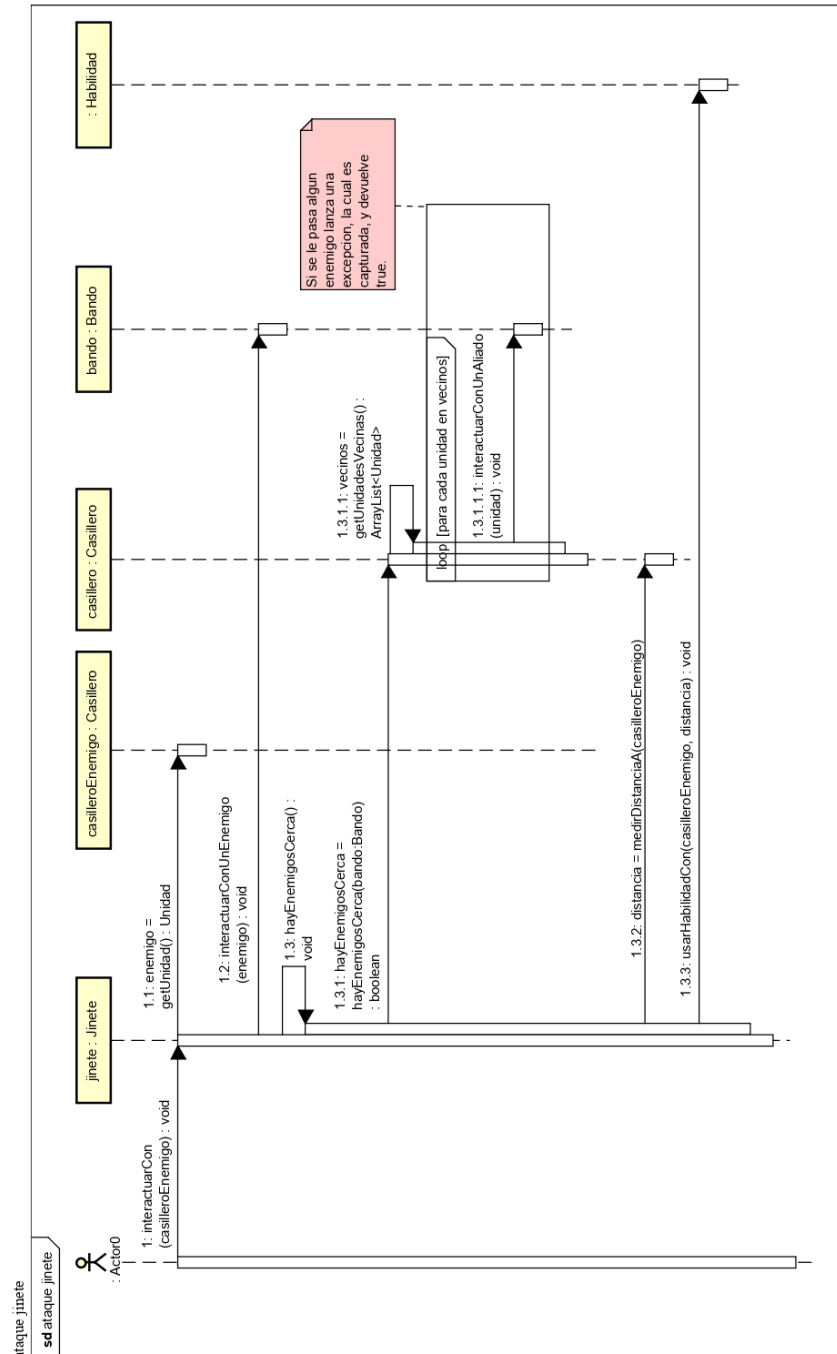


1 / 1

Figura 4: . Detección y movimiento de un Batallón

## Ataque de un jinete.jpg

2019/11/20



1 / 1

Figura 5: Ataque de un Jinete



## 8. Diagramas de estado

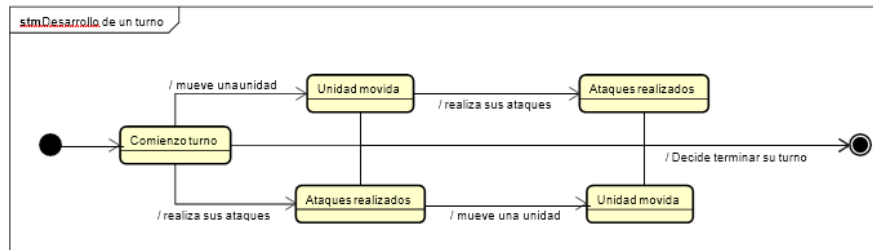


Figura 6: . Turno