

Marc Benítez Benavides
Javier de Dios González
TDD-Battleships (Hundir la flota)

CAIXA NEGRA

Funcionalitat: Test creació cell per defecte.

Localització: arxiu: Cell.java, class: Cell, method: Cell

Test: arxiu: CellTests.java, class: CellTests.java, method: SetCellTest

Tipus de test: Caixa negra. Tècniques: La Cell s'inicia de forma correcta. Test bàsic.

Funcionalitat: Test assignar valor a cell i recuperar-lo.

Localització: arxiu: Cell.java, class: Cell, method: setM_Value i getM_Value

Test: arxiu: CellTests.java, class: CellTests.java, method: SetterGetterValueTest

Tipus de test: Caixa negra. Tècniques: Comprova que es fa el set de manera correcta i el get retorna el valor de la Cell. En aquest test no es comproven valors frontera, ja que es comprovarà en un nivell més alt.

Funcionalitat: Test mostrar el valor de cell i recuperar-lo.

Localització: arxiu: Cell.java, class: Cell, method: setM_show i getM_show

Test: arxiu: CellTests.java, class: CellTests.java, method: SetterGetterShowTest

Tipus de test: Caixa negra. Tècniques: Comprova que s'actualitza el valor m_show de la Cell correctament, si es true, la Cell s'ensenyà, si per contra es false no es mostra.

Funcionalitat: Test creació Ship per defecte.

Localització: arxiu: Ship.java, class: Ship, method: Ship

Test: arxiu: ShipTests.java, class: ShipTests.java, method: setShipTest

Tipus de test: Caixa negra. Tècniques: Comprova que es crea el vaixell correctament per defecte.

Funcionalitat: Test creació Ship amb parametres.

Localització: arxiu: Ship.java, class: Ship, method: Ship

Test: arxiu: ShipTests.java, class: ShipTests.java, method: initializeShipTest

Tipus de test: Caixa negra. Tècniques: Comprova que es crea el vaixell correctament amb el constructor amb paràmetres. Al igual que a Cell aquí no es comproven els valors.

Funcionalitat: Test setters i getters X per la classe Ship.

Localització: arxiu: Ship.java, class: Ship, method: setM_x, getM_x, setM_y i getM_y

Test: arxiu: ShipTests.java, class: ShipTests.java, method: SetterGetterXTest i SetterGetterYTest

Tipus de test: Caixa negra. Tècniques: Comprova que es settejen i es recuperen correctament els valors de 'x' i 'y'.

Funcionalitat: Test setter i getter orientació per la classe Ship.

Localització: arxiu: Ship.java, class: Ship, method: setM_orientation i getM_orientation

Test: arxiu: ShipTests.java, class: ShipTests.java, method: SetterGetterOrientationTest

Tipus de test: Caixa negra. Tècniques: Comprova que es setteja i es recupera correctament el valor de l'orientació (char).

Funcionalitat: Test setter i getter longitud per la classe Ship.

Localització: arxiu: Ship.java, class: Ship, method: setM_length i getM_length

Test: arxiu: ShipTests.java, class: ShipTests.java, method: SetterGetterLengthTest

Tipus de test: Caixa negra. Tècniques: Comprova que es setteja i es recupera correctament el valor de length del Ship.

Funcionalitat: Test setter i getter sunk per la classe Ship.

Localització: arxiu: Ship.java, class: Ship, method: setM_sunk i getM_sunk

Test: arxiu: ShipTests.java, class: ShipTests.java, method: SetterGetterSunkTest

Tipus de test: Caixa negra. Tècniques: Comprova que es setteja i es recupera correctament el valor de sunk del Ship.

Funcionalitat: Test si la coordenada dels parametres pertany a aquest Ship.

Localització: arxiu: Ship.java, class: Ship, method: isThisShip

Test: arxiu: ShipTests.java, class: ShipTests.java, method: isThisShipTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent, test de valors frontera. Es testeja tocant al voltant d'un vaixell

Funcionalitat: Test per canviar valors d'una Cell concreta a traves de Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: setCellValue i getCellValue

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: SetterGetterCellValueTest

Tipus de test: Caixa negra. Tècniques: Testeja si el valor de la Cell s'actualitza correctament.

Funcionalitat: Test per ensenyar una Cell concreta a traves de Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: showCell i getCellShow

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: SwitchShowCellTest

Tipus de test: Caixa negra. Tècniques: Testeja si la Cell es mostra i s'actualitza correctament.

Funcionalitat: Test per afegir un vaixell de Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: placeShip

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: PlaceShipTest

Tipus de test: Caixa negra. Tècniques: Pairwise, partició equivalent i test de valors frontera. S'han generat casos pairwise per a cada longitud (2, 3, 4, 5) i s'han ajuntat per fer un test conjunt de la funció placeShip. Rep els valors per col·locar el Ship i el resultat esperat (boolean).

Funcionalitat: Test per afegir vaixells a les fronteres del Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: placeShip

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: PlaceShipBoundariesTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i test de valors frontera. S'han generat casos localitzats a les fronteres del tauler per a cada longitud (2, 3, 4, 5) i s'han ajuntat per fer un test conjunt de la funció placeShip. Rep els valors per col·locar el Ship i el resultat esperat (boolean).

Funcionalitat: Test per comprovar les deteccions al Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: placeShip

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: PlaceShipHorizontalCollisionTest i PlaceShipVerticalCollisionTest.

Tipus de test: Caixa negra. Tècniques: Partició equivalent i test de valors frontera. S'ha col·locat un vaixell que a vegades colisiona amb una casella marcada amb un 1 (simula un vaixell). Rep els valors per col·locar el Ship i el resultat esperat ve donat per una comparació per veure si xoca o no.

Funcionalitat: Test per comprovar que es modifiquen els valors de les Cells de Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: placeShip

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: PlaceShipPlacementHorizontalTest i PlaceShipPlacementVerticalTest.

Tipus de test: Caixa negra. Tècniques: Partició equivalent i test de valors frontera. Es col·loca un vaixell en totes les posicions del tauler i amb totes les possibles longituds i es testeja si s'ha modificat o no el tauler. El tauler s'ha de modificar només si el vaixell hi cap.

Funcionalitat: Test comprovar que no es modifiquen les Cells si no es necessari a Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: placeShip

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: PlaceShipPlacementShipCollisionTest.

Tipus de test: Caixa negra. Tècniques: Partició equivalent i test de valors frontera. Es col·loquen quatre vaixells en diferents posicions al centre del tauler i es mira si es pot afegir el vaixell sense xocs o no. No es comprova si es modifiquen els valors de les caselles perquè ja s'ha fet al test anterior.

Funcionalitat: Test per realitzar un atac sobre Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: attackPoint

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: attackPointTest.

Tipus de test: Caixa negra. Tècniques: Partició equivalent i test de valors frontera. S'escolleix una x i una y i es realitza l'atac al tauler, si ja s'ha atacat la posició es retorna -1, si es aigua 0 i si es tocat 1.

Funcionalitat: Test per veure si un vaixell esta enfosat a Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: isSunk

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: isSunkTest.

Tipus de test: Caixa negra. Tècniques: Partició equivalent i test de valors frontera. S'itera sobre un vaixell i es comprova que no es marca com a enfosat fins que de veritat ho està. Un cop enfosat també es comprova que s'ha modificat el valor de 1 a 2 (valor que representa un vaixell enfosat).

Funcionalitat: Constructor de la classe Player.

Localització: arxiu: Player.java, class: Player, method: Player

Test: arxiu: PlayerTests.java, class: PlayerTests.java, method: setPlayerTest.

Tipus de test: Caixa negra. Tècniques: Comprova que es crea correctament el objecte Player.

Funcionalitat: Classe Player intenta afegir un vaixell al tauler.

Localització: arxiu: Player.java, class: Player, method: addShip

Test: arxiu: PlayerTests.java, class: PlayerTests.java, method: addShipTest.

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Comprova que s'afegeixen vaixells al tauler que té el jugador. Això ho fa a traves de la funció addShip de la classe Gameboard.

Funcionalitat: Classe Player busca un vaixell al vector de vaixells.

Localització: arxiu: Player.java, class: Player, method: findShip

Test: arxiu: PlayerTests.java, class: PlayerTests.java, method: findShipTest.

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Comprova que el vaixell existeix donada una posició (x, y) i fent servir el mètode de la classe vaixell isThisShip. En cas de no trobar el vaixell retorna -1 indicant-ho.

Funcionalitat: Es realitza un atac al tauler del Player.

Localització: arxiu: Player.java, class: Player, method: makeAttack

Test: arxiu: PlayerTests.java, class: PlayerTests.java, method: makeAttackTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Testeja si un atac es bo o no, retorna el valor adient en cada cas. En cas de ser un atac no vàlid es retorna -1.

Funcionalitat: Es comprova si el conteig de vaixells vius de cada jugador ha arribat a 0.

Localització: arxiu: Player.java, class: Player, method: hasLost

Test: arxiu: PlayerTests.java, class: PlayerTests.java, method: hasLostTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Es testea si un jugador ha perdut (0 vaixells vius) i té 1 o més vaixells al array de vaixells, en cas de tenir-lo aquest jugador ha perdut i es retorna true, altrament false.

Funcionalitat: Es comprova si tant el jugador com la maquina han entrat tot tipus de valors a les funcions dels mocks.

Localització: arxiu: BattleshipGame.java, class: BattleshipGame, method: mainMock

Test: arxiu: BattleshipGameTest.java, class: BattleshipGameTest.java, method: mainMockFunctionsTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Testea si les funcions dels mocks interactuen bé amb els inputs que poden fer els usuaris. És un test que assegura que la partida que es simularà pot arribar a funcionar.

Funcionalitat: Es comprova si els valors que s'han testejat prèviament arriben a una conclusió del joc.

Localització: arxiu: BattleshipGame.java, class: BattleshipGame, method: mainMock

Test: arxiu: BattleshipGameTest.java, class: BattleshipGameTest.java, method: mainMockGameJugadorTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Testea que si guanya el jugador es passi per totes les parts que ha de passar en cas que aquest guanyi. Gràcies a la funció anterior sabem si s'arribarà a un resultat satisfactori.

Funcionalitat: Es comprova si els valors que s'han testejat prèviament arriben a una conclusió del joc.

Localització: arxiu: BattleshipGame.java, class: BattleshipGame, method: mainMock

Test: arxiu: BattleshipGameTest.java, class: BattleshipGameTest.java, method: mainMockGameMaquinaTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Testea que si guanya la maquina es passi per totes les parts que ha de passar en cas que aquesta guanyi.

CAIXA BLANCA

Funcionalitat: Test si la coordenada dels parametres pertany a aquest Ship.

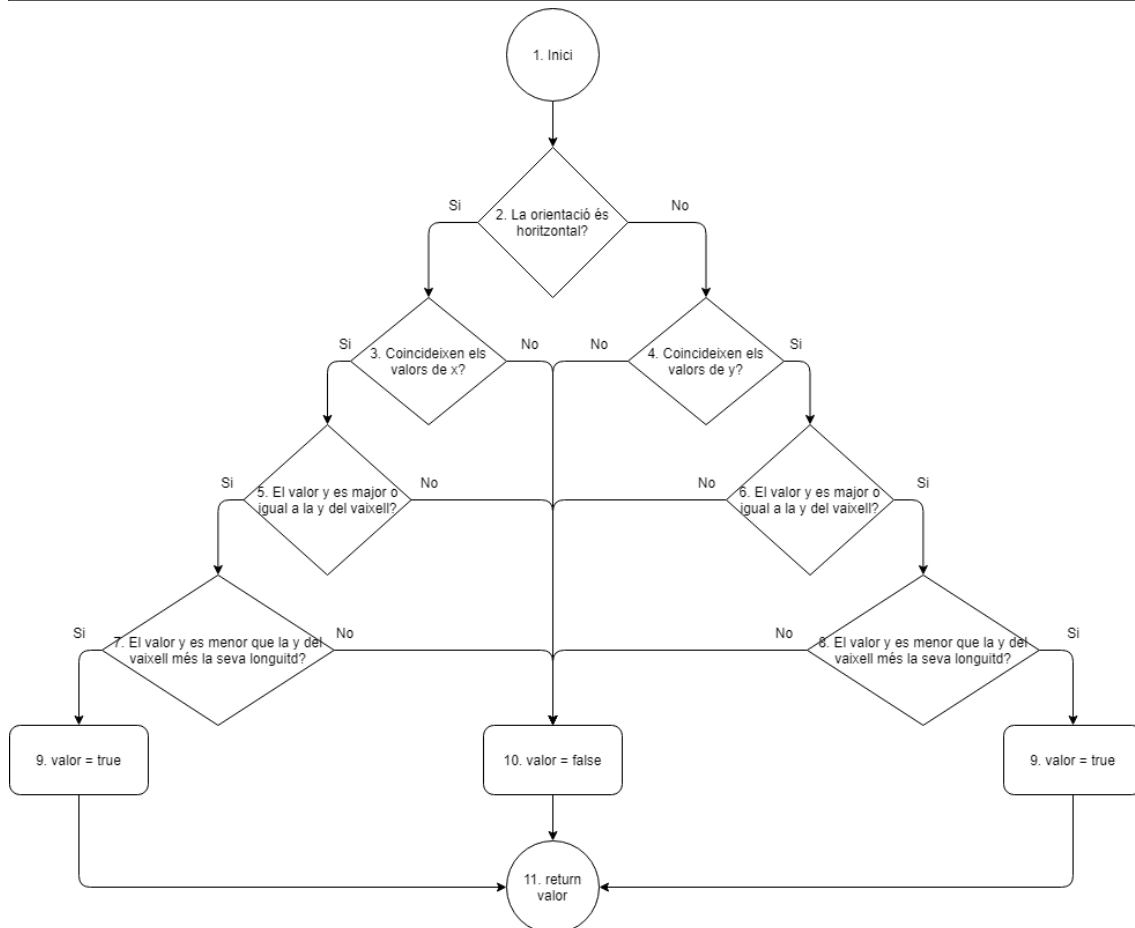
Localització: arxiu: Ship.java, class: Ship, method: isThisShip

Test: arxiu: ShipTests.java, class: ShipTests.java, method: isThisShipTest

Tipus de test: Caixa blanca. Tècniques: Statement coverage, decision coverage, Condition coverage i path coverage. Es testreja tot el voltant d'un vaixell, per tant totes les condicions de les condicions de retorn.

```
38 public boolean isThisShip(int x, int y) {  
39     if (this.m_orientation == 'h') {  
40         return ((this.m_x == x) && (y >= this.m_y) && (y < (this.m_y + this.m_length)));  
41     } else {  
42         return ((this.m_y == y) && (x >= this.m_x) && (x < (this.m_x + this.m_length)));  
43     }  
44 }  
45 }
```

Ship	100% (1/1)	100% (13/13)	100% (25/25)
------	------------	--------------	--------------

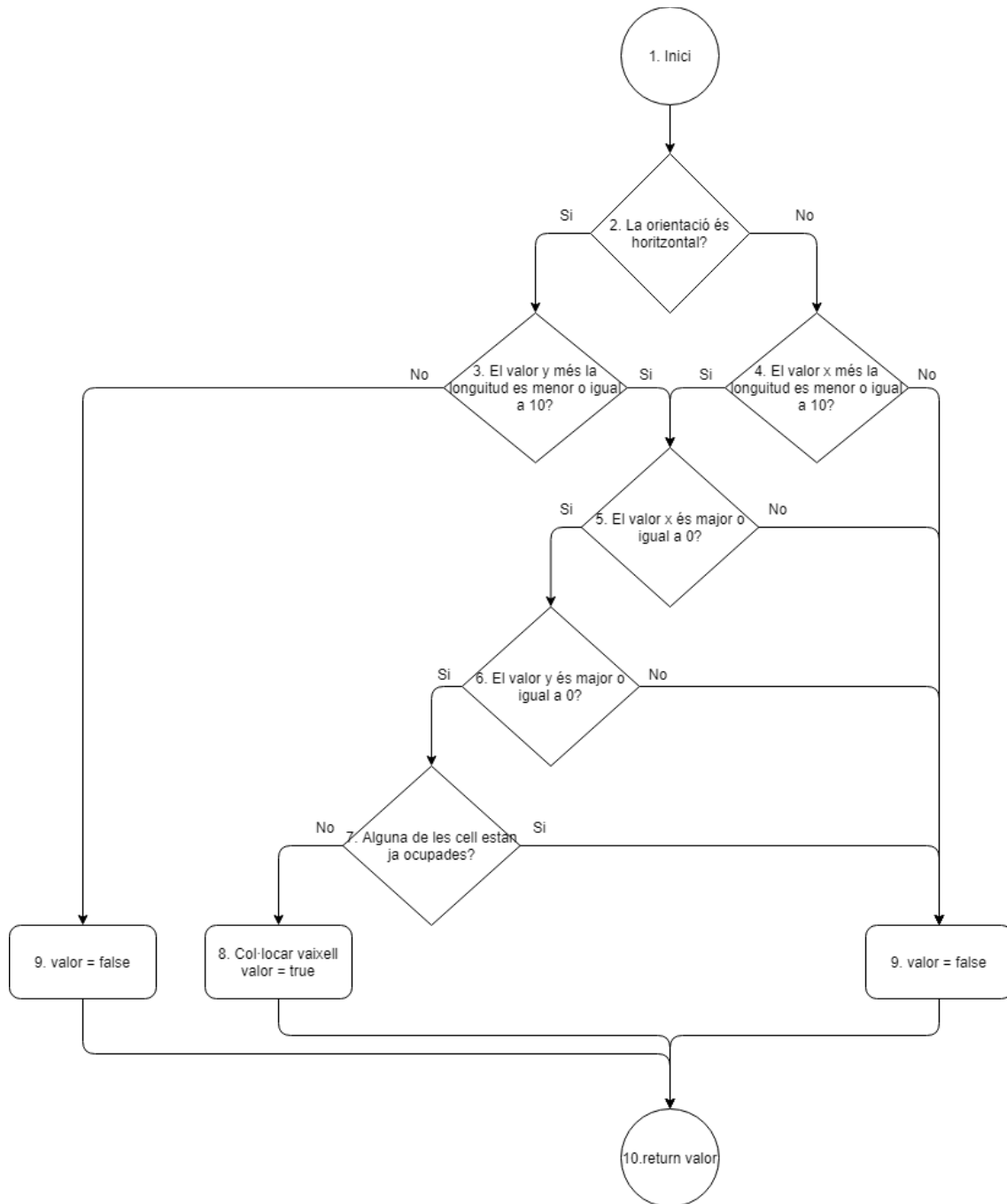


Funcionalitat: Test per canviar valors d'una Cell concreta a través de Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: placeShip

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: PlaceShipTest

Tipus de test: Caixa blanca. Statement coverage, decision coverage, Condition coverage, path coverage. Es testreja si es pot afegir un vaixell en diferents llocs del tauler i en diferents condicions. S'executen els test case amb un fitxer generat pel paramProvider.



Funcionalitat: Test per canviar valors d'una Cell concreta a traves de Gameboard.

Localització: arxiu: Gameboard.java, class: Gameboard, method: placeShip

Test: arxiu: GameboardTests.java, class: GameboardTests.java, method: PlaceShipBoundariesTest

Tipus de test: Caixa blanca. Statement coverage, decision coverage, Condition coverage, path coverage. Es testeja si es pot afegir un vaixell en diferents llocs del tauler (fronteres) i en diferents condicions. S'executen els test case amb un fitxer generat pel paramProvider.

```
21 public boolean placeShip(int x, int y, int orientation, int length) {
22     if (orientation == 'h' && y + length <= 10 && x >= 0 && y >= 0) {
23         for(int i = y; i < y + length; i++) {
24             if (this.getCellValue(x, i) != 0) {
25                 return false;
26             }
27         }
28         for(int i = y; i < y + length; i++) {
29             this.setCellValue(x, i, value: 1);
30         }
31         return true;
32     }
33
34     if (orientation == 'v' && x + length <= 10 && x >= 0 && y >= 0) {
35         for(int i = x; i < x + length; i++) {
36             if (this.getCellValue(i, y) != 0) {
37                 return false;
38             }
39         }
40         for(int i = x; i < x + length; i++) {
41             this.setCellValue(i, y, value: 1);
42         }
43         return true;
44     }
45     return false;
46 }
47 }
```

Funcionalitat: Test que controla que passa quan es simula una partida i guanya el jugador.

Localització: arxiu: BattleshipGame.java, class: BattleshipGame, method: mainMock

Test: arxiu: BattleshipGameTest.java, class: BattleshipGameTest.java, method: mainMockGameJugadorTest

Tipus de test: Caixa blanca. Loop testing aniuat (diversos), Loop testing simple (diversos), statement coverage, decision coverage, condition coverage, path coverage. Es testeja tot tipus de valors per a tots els tipus de inputs tant del jugador com de la maquina. Es passa per tots els bucles i es trenquen aquests. Es recorre tot el codi que té a veure amb el jugador. Aquesta funció és massa gran per ser enganxada al informe.

Funcionalitat: Test que controla que passa quan es simula una partida i guanya la maquina.

Localització: arxiu: BattleshipGame.java, class: BattleshipGame, method: mainMock

Test: arxiu: BattleshipGameTest.java, class: BattleshipGameTest.java, method: mainMockGameMaquinaTest

Tipus de test: Caixa blanca. Loop testing aniuat (diversos), Loop testing simple (diversos), statement coverage, decision coverage, condition coverage, path coverage. Es testeja tot tipus de valors per a tots els tipus de inputs tant del jugador com de la maquina. Es passa per tots els bucles i es trenquen aquests. Es recorre tot el codi que té a veure amb la maquina. Aquesta funció és massa gran per ser enganxada al informe.

Package	Class, %	Method, %	Line, %
BattleshipGame.Controlador	100% (5/5)	100% (38/38)	100% (206/206)

Class ▾	Class, %	Method, %	Line, %
Ship	100% (1/1)	100% (13/13)	100% (25/25)
Player	100% (1/1)	100% (10/10)	100% (32/32)
Gameboard	100% (1/1)	100% (8/8)	100% (40/40)
Cell	100% (1/1)	100% (5/5)	100% (7/7)
BattleshipGame	100% (1/1)	100% (2/2)	100% (102/102)

MOCK OBJECTS

Funcionalitat: Es comprova si es retorna un valor acceptable random.

Localització: arxiu: RandomShip.java, class: RandomShip, method: getRandomX, getRandomY i getRandomOrientation

Test: arxiu: RandomShipTests.java, class: RandomShipTests.java, method: getRandomXYOrientationTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Testeja si retorna valors entre 0 i 9 (ambdós inclosos) i si retorna un valor entre 'h' i 'v'. Són els valors que pot pendre la posició X, Y i orientació d'un vaixell.

Funcionalitat: Es comprova si l'usuari ha entrat un valor vàlid per consola.

Localització: arxiu: UserInputs.java, class: UserInputs, method: getUserShipX, getUserShipY, getUserShipOrientation, getNumberOfShips i getUserShipLength.

Test: arxiu: RandomShipTests.java, class: RandomShipTests.java, method: selectNumberOfShipsTest, getUserShipXTest, getUserShipYTest, getUserShipLengthTest i getUserShipOrientationTest

Tipus de test: Caixa negra. Tècniques: Partició equivalent i valors frontera. Testeja si retorna valors entre 0 i 9 (ambdós inclosos) per X i Y. Valors entre 'h' i 'v' per l'orientació. Valors entre 1 i 9 (ambdós inclosos) per al número de vaixells i per últim valors entre 2 i 5 per a la longitud del vaixell.