# Report Practiques

## Game:

|  |  |
| --- | --- |
| **Funcionalitat:** | El main crea un Game que aquest crea una classe Menu(). |
| **Localització:** | **Develop :** **Game** funció main  **Test:** **GameTest** i funció **TestGame**. |
| **Test:** | Caixa blanca: Statement Coverage. Per tal de comprovar que al iniciar el Game es crea un Menu y es mostren les diferents opcions. |

## Menú:

Totes les classes utilitzen **ManagerIOMock()** per simular les entrada de l'usuari per teclat:

|  |  |
| --- | --- |
| **Funcionalitat:** | Mostra el menú con las diferents opcions y executa la Opció 1 Jugar, que comença la partida. |
| **Localització:** | **Develop :** **Menu** class  **Test:** **MenuTest** I function **MenuTestRunning** |
| **Test:** | Caixa blanca: **Condition Coverage/ Statement Coverage/Path Coverage**. Per tal de comprovar que el passem per totes les condicions del Switch, en aquest cas l’opció 1 de començar a Jugar. Comprovem comparant amb el Sting que mostra al començar a jugar. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Mostra el menú con las diferents opcions y executa la Opció 2 Opcions, que mostra les diferents opcions. |
| **Localització:** | **Develop :** **Menu** class  **Test:** **MenuTest** I function **MenuTestOptions**. |
| **Test:** | **Caixa blanca:** **Condition Coverage/ Statement Coverage/Path Coverage**. Per tal de comprovar que el passem per totes les condicions del **Switch**, en aquest cas l’opció 2 de mostrar les opcions. Comprovem comparant amb el String que mostra al seleccionar les opcions. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Mostra el menú con las diferents opcions y executa la Opció 3 Exit, que tanca el Joc. |
| **Localització:** | **Develop :** **Menu** class  **Test:** MenuTest I function MenuTestExit. |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**. Per tal de comprovar que el passem per totes les condicions del Switch, en aquest cas l’opció 2 de mostrar les opcions. Comprovem comparant amb el String que mostra al seleccionar les opcions. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Comprovació de tots el casos que no son correctes, valors límit i frontera |
| **Localització:** | **Develop :** **Menu** class  **Test:** MenuTest I function TestMenuBadOptions. |
| **Test:** | **Caixa negra:** **Particions equivalent i valores límit/frontera.** |

En aquest cas concret al realitzar **el Condition Coverage** per a les diferents condicions del switch també estem realitzant **Statement coverage** , **Decision Coverage i Path Coverage**.

## Jugador:

Totes les classes utilitzen **ManagerIOMock()** per simular les entrada de l'usuari per teclat:

|  |  |
| --- | --- |
| **Funcionalitat:** | Creació de dos jugadors per verificar els diferents tipo (Jugador: 1 i Random: 2) |
| **Localització:** | **Develop :** **Jugaro** class i funcion **Jugador() Contrustor**  **Test:** JugadorTest I function TestCreateJugador. |
| **Test:** | **Caixa blanca:** **Statement Coverage / Path Coverage**. Comprovem la correcta inicialització del contrustor. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Fem que els dos jugadors utilitzen la funció d'atacar |
| **Localització:** | **Develop :** **Jugaro** class i i funcion **atacar()**  **Test:** JugadorTest I function TestJugadorAtacar. |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**. Per tal de comprovar el dos path possibles fem que ataqui el jugador i també el Random. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Funció que indica si ja hi ha un guanyador en la partida |
| **Localització:** | **Develop :** **Jugaro** class i funcion **ganador()**  **Test:** JugadorTest I function TestJugadorGana. |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**. Simulen una partida sencera per tal de comprovar el dos casos, quant encara no ha guanyat ningun i quant finalitza (guanya algu) |

## Partida:

Totes les classes utilitzen **ManagerIOMock()** per simular les entrada de l'usuari per teclat:

|  |  |
| --- | --- |
| **Funcionalitat:** | Simulació d'una partida en què guanya el jugador |
| **Localització:** | **Develop :** **Partida** class i funcion **printGanador()**  **Test:** PartidaTest I function TestGanarJugador. |
| **Test:** | **Caixa blanca:** **Condition Coverage**  Simulen una partida sencera per tal de comprovar el dos casos, quant encara no ha guanyat ningun i quant finalitza (guanya algú). |

|  |  |
| --- | --- |
| **Funcionalitat:** | Simulació d'una partida en què guanya el random |
| **Localització:** | **Develop :** **Partida** class i funcion **printGanador()**  **Test:** PartidaTest I function TestGanarRandom. |
| **Test:** | **Caixa blanca:** **Condition Coverage**  Simulen una partida sencera per tal de comprovar el dos casos, quant encara no ha guanyat ningun i quant finalitza (guanya algú). |

**Simulant el dos casos realitzem també Statement Coverage i Path Coverage**

## Posicion:

|  |  |
| --- | --- |
| **Funcionalitat:** | Creacio de posicions amb valors = X, Y , Tamaño , Orientacion |
| **Localització:** | **Develop :** **Posicion** class  **Test:** PosicionTest I function TestConstructor. |
| **Test:** | **Caixa blanca: Statement Coverage.** Comprovació de los mètodes Constructor ,Setter, Getter i toString. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Retornar la orientació en format String |
| **Localització:** | **Develop :** **Posicion** class  **Test:** PartidaTest I function TestOrientacion. |
| **Test:** | **Caixa blanca: Condition Coverage /Statemenent Coverage** . Realitzem les diferents condicions de la orientació |

**Simulant el dos casos realitzem també Condition Coverage, Statement Coverage i Path Coverage de la classe Posicion.**

## Tablero:

|  |  |
| --- | --- |
| **Funcionalitat:** | Mostrar el Tauler de la Partida |
| **Localització:** | **Develop :** **Tablero** class i funcion **mostrarTablero()**  **Test:** **TableroTest** I function **testMostrarTablero**(). |
| **Test:** | **Caixa blanca:** **Statement Coverage /Path Coverage**. Assegura que es mostra correctament el Tauler(). |

|  |  |
| --- | --- |
| **Funcionalitat:** | Inicialitzar tauler |
| **Localització:** | **Develop :** **Tablero** class i funcion **iniciarTablero()**  **Test:** **TableroTest** I function **testIniciarTablero()** |
| **Test:** | **Caixa blanca:** **Loop Testing / Statement Coverage /Path Coverage**. Simulen una partida sencera per tal de comprovar el dos casos, quant encara no ha guanyat ningun i quant finalitza (guanya algú). |

|  |  |
| --- | --- |
| **Funcionalitat:** | Inserta una posició en el Tablero |
| **Localització:** | **Develop :** **Tablero** class i funcion **inserPosicion()**  **Test:** **TableroTest** I function **testInserPosicion ()** |
| **Test:** | **Caixa blanca:** **Path Coverage**. Insertem una posició correcta. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Inserta una posició incorrecta en el Tablero |
| **Localització:** | **Develop :** **Tablero** class i funcion **inserPosicionKO()**  **Test:** **TableroTest** I function **testInserPosicionKO ()** |
| **Test:** | **Caixa blanca:** **Path Coverage**. Insertem una posició incorrecta. |

**inserPosicion() + inserPosicionKO() = Decision / Condition Coverage i Statement**

|  |  |
| --- | --- |
| **Funcionalitat:** | Inserta una posició genera de forma aleatòria |
| **Localització:** | **Develop :** **Tablero** class i funcion **inserPosicionRandom()**  **Test:** **TableroTest** I function **testInserPosicionRandom ()** |
| **Test:** | **Caixa blanca:** **Statement**/ **Path Coverage**. Generem una posició correcta. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Genera una barrera protectora al voltant d’una posició (1 quadrat) |
| **Localització:** | **Develop :** **Tablero** class i funcion **protecionBarco()**  **Test:** **TableroTest** I function **testProtecionBarco ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  **Caixa Negra: Limites i Frontera. Orientació : (-inf, 0), [0, 3], [4, inf)**  Simulen una partida sencera per tal de comprovar el dos casos, quant encara no ha guanyat ningun i quant finalitza (guanya algú) |

|  |  |
| --- | --- |
| **Funcionalitat:** | Inserta un vaixell dintre del taulell. |
| **Localització:** | **Develop :** **Tablero** class i funcion **insertarBarco()**  **Test:** **TableroTest** I function **testInsertarBarco ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  **Caixa Negra: Limites i Frontera. Orientació : (-inf, 0), [0, 3], [4, inf)**  Inserta Barcos en todas las orientacions possibles. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Comprova si es pot inserta una posició amb una orientacions concreta. |
| **Localització:** | **Develop :** **Tablero** class i funcion **sePuedeInsertarOrientacion()**  **Test:** **TableroTest** I function **testSePuedeInsertarOrientacion ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  **Caixa Negra: Limites i Frontera. Orientació : (-inf, 0), [0, 3], [4, inf)**  Inserta Barcos en todas las orientacions possibles. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Agafa la orientació llegida per teclat |
| **Localització:** | **Develop :** **Tablero** class i funcion **leerOrientacion()**  **Test:** **TableroTest** I function **testLeerOrientacion ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  **Caixa Negra: Limites i Frontera. Orientació : (-inf, 0), [0, 3], [4, inf)**  Insertem Orientacions invalides i valides per cobrir tot el codi. |

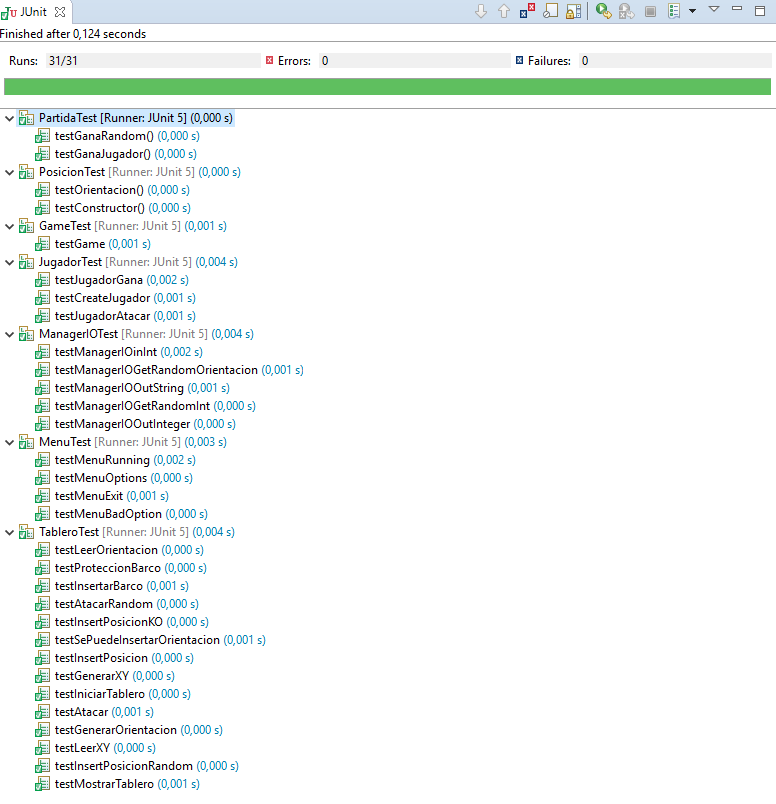
|  |  |
| --- | --- |
| **Funcionalitat:** | Genera una orientació |
| **Localització:** | **Develop :** **Tablero** class i funcion **generarOrientacion()**  **Test:** **TableroTest** I function **testGenerarOrientacion ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  **Caixa Negra: Limites i Frontera.**  Generem Orientacions invalides i valides per cobrir tot el codi. |

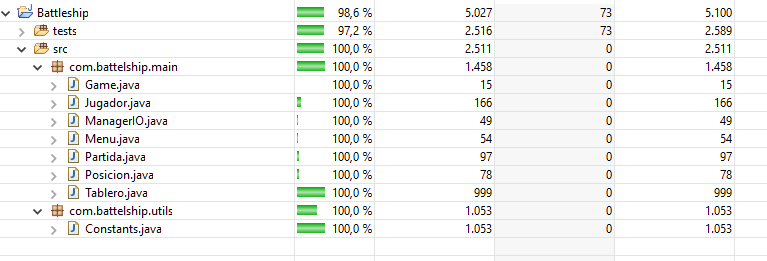
|  |  |
| --- | --- |
| **Funcionalitat:** | Agafa la posició llegida per teclat (x-y) |
| **Localització:** | **Develop :** **Tablero** class i funcion **leerXY()**  **Test:** **TableroTest** I function **testLeerXY ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  **Caixa Negra: Limites i Frontera. Posicion x-y : (-inf, 1), [0, 10], [10, inf)** Lectura de posicions X,Y invalides i valides per cobrir tot el codi. |

|  |  |
| --- | --- |
| **Funcionalitat:** | Genera una posició (x-y) |
| **Localització:** | **Develop :** **Tablero** class i funcion **generarXY()**  **Test:** **TableroTest** I function **testGenerarXY ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  **Caixa Negra: Limites i Frontera. Posicion x-y : (-inf, 1), [0, 10], [10, inf)**  Generació de posicions X,Y invalides i valides per cobrir tot el codi. |

|  |  |
| --- | --- |
| **Funcionalitat:** | El jugador atacar al tauler |
| **Localització:** | **Develop :** **Tablero** class i funcion **atacar()**  **Test:** **TableroTest** I function **testAtacar ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  Atacar a tots el tipus de posicions del tauler per cobrir tot el codi. |

|  |  |
| --- | --- |
| **Funcionalitat:** | El jugador Aleatori atacar al tauler |
| **Localització:** | **Develop :** **Tablero** class i funcion **atacarRandom()**  **Test:** **TableroTest** I function **testAtacarRandom ()** |
| **Test:** | **Caixa blanca:** **Condition Coverage / Statement Coverage /Path Coverage**.  Atacar a tots el tipus de posicions del tauler per cobrir tot el codi. |





## Game:

**Funcionalitat**: El main crea un Game que a su vez ejecuta el menú principal del juego <descripció de la funcionalitat introduida o modificada>

**Localització**: Game function main <Arxiu, classe i mètode desenvolupat>

**Test**: GameTest I function TestGame.

Caixa blanca: Statement Coverage. Per tal de comprobar que al iniciar el Game es crea un Menu y es mostren les diferents opcions.

<Descripcio del tipus de test realitzat (caixa negra, caixa blanca, …) i les tècniques utilitzades (particions equivalents, pairwise testing, mockups, automatització)>

SIEMPRE:

Caixa Negra:

-Particions Equivalents

-Valor límit i frontera

Caixa blanca:

-Statement Coverage

-Decision Coverage0 : minimo de 2 metodos

-Path Coverage

-Loop Testing (loop simple) n y m --

|  |  |
| --- | --- |
| Caixa Negra | |
| Particions Equivalents |  |
| Valor límit i frontera |  |

|  |  |
| --- | --- |
| Caixa Blanca: | |
| Statement Coverage | OK |
| Decision Coverage | OK (minimo 2 metodos) |
| Path Coverage | OK |
| Loop Testing | OK |