RANCANGAN BOARDGAME

BATTLESHIP

https://cs2103-ay1819s2-w14-3.github.io/main/DeveloperGuide.html#Design-Logic

Class / atribut yang ada di game:

* Arena
* Kapal (beberapa jenis kapal, jumlah titik kapal, koordinat kapal)
* Pemain
* Pin penanda penyerangan

================================================================

kapal(ship)(kemungkinan interface):

* Carrier(5 pin hole) / tranpor udara
* Battleship (4 pin hole) / perang
* Cruiser (3 pin hole) / kapal layar
* Submarine (3 pin hole) / selam
* Destroyer(2 pin hole) / penghancur

pemain(player 1 dan 2):

* Player 1
* Player 2
* punya nama player(½)
* punya koordinat kapal masing-masing

arena:

* ukuran

permainan/game/gameplay

* memiliki pemain( player1 dan player2)
* memiliki giliran( player1 atau player2)
* cek shotCoordinat dari player1 apakah ada di shipCoordinat player2
* cek shotCoordinat dari player2 apakah ada di shipCoordinat player1

================================================================

Player memiliki arena, kapal

**RULE:**

Player:

* Memiliki nama player
* Memiliki kapal (Carrier, Battleship, Cruiser , Submarine, Destroyer)
* Memiliki arena
* dapat *mengakses* arena (class arena)
* dapat *mengakses* kapal (class kapal)
* dapat *mengakses* / masuk dalam game (class game)
* dapat mengatur lokasi kapal (mengatur di class kapal)
* dapat menembak (dengan koordinat)

Kapal:

* memiliki nama kapal
* memiliki lubang pin (ukuran kapal)
* memiliki koordinat kapal (diatur oleh player)
* dapat tenggelam / tidak (tenggelam jika semua koordinat di list kapal terkena serangan / semua koordinat kapal disebut oleh player lain)

Arena:

* memiliki ukuran
* memiliki pemain

GamePlay:

* memiliki pemain( player1 dan player2)
* memiliki giliran( player1 atau player2)
* cek shotCoordinat dari player1 apakah ada di shipCoordinat player2
* cek shotCoordinat dari player2 apakah ada di shipCoordinat player1

**JALAN PERMAINAN**

1. player1 dan player2 menginput koordinat(shipCoordinate) dari 5 kapal(ship)
2. player1 menginput koordinat serangan(shotCoordinate)
3. jika shotCoordinate player1 ada di shipCoordinate player2, player2 mengatakan “Hit”, jika tidak maka player2 mengatakan “Miss”
4. jika shotCoordinate player2 ada di shipCoordinate player1, player1 mengatakan “Hit”, jika tidak maka player1 mengatakan “Miss”
5. langkah 3&4 terus diulang
6. Jika ada kumpulan shipCoordinate dari player1 yang berada di kumpulan shotCoordinate player2, maka nilai isSink dari kapal(ship) player1 dengan shipCoordinate tersebut berubah menjadi true, shipCoordinate dari player1 dihapus sesuai kumpulan shotCoordinate player2.
7. Jika ada kumpulan shipCoordinate dari player2 yang berada di kumpulan shotCoordinate player1, maka nilai isSink dari kapal(ship) player2 dengan shipCoordinate tersebut berubah menjadi true, shipCoordinate dari player2 dihapus sesuai kumpulan shotCoordinate player1.
8. langkah 6&7 terus diulang
9. Jika seluruh shipCoordinate dari salah satu player habis, maka player dengan shipCoordinate tersisa akan menjadi pemenang.

**ATRIBUT DARI CLASS**

==========================================================================

class Player:

-playerName: nama player (player1, dan player2)

- arean: Arena -> untuk mengambil coordinate dari arena yang sudah diatur oleh class PlayGame

- listCoorCarrier: berisi koordinat Carrier

- listCoorBattleShip: berisi koordinat BattleShip

- listCoorCruiser: berisi koordinat Cruiser

- listCoorSubmarine: berisi koordinat Submarine

- listCoorDestroyer: berisi koordinat Destroyer

- listAllShip: List yang berisi semua koordinat kelima kapal, gabungan semua list kapal

- listSafeShip: Salinan nilai dari listAllShip, berfungsi untuk cek apakah coordinate yang dipilih lawan ada dalam listSafeShip, jika ada maka coordinate tersebut dalam listSafeShip akan dihapus, jika isi dalam listSafeShip sudah habis artinya tidak ada kapal tersisa, permain tersebut kalah

- listCoorHit: berisi kumpulan koordinat saat player menyerang

-----------------------------------Method class Player------------------------------------

- SetShipCoor(int, int) -> mengatur koordinat dari kapal

- SetHitCoor(player) -> mengatur koordinat penyerangan

- CheckHitCoor() -> cek apakah koordinat SetHitCoor(player) sudah ada di listCoorHit()

- CheckShipCoor() ->cek apakah koordinat dalam SetShipCoor(int,int) ada di listAllShip

- CheckAreaCoor() -> cek apakah SetShipCoor(int,int) tidak ada dalam class Arena.arenaCoordinate.

===========================================================================

===========================================================================

class IShip (interface)

- shipName(prop): nama dari kapal (string)

- shipSize(prop): ukuran dari kapal (int)

- shipCoordinate(prop): berisi list coordinat kapal

- isSink(prop): status apakah kapal tenggelam

----------------------------------------------

- tidak punya method

===========================================================================

===========================================================================

class Battleship, Carrier, Cruiser, Submarine, Destroyer : IShip

- shipName: nama kapal (string)

- shipSize: ukuran kapal (int)

- shipCoordinate: list koordinat kapal

- isSink: status apakah kapal tenggelam (default value = false)

===========================================================================

class Arena:

- sizeOfArena: berisi ukuran dari arena yang dibuat

- arenaCoordinate: berisi koordinat berdasarkan sizeOfArena, misal nilai sizeOfArena = 10, maka arena yang tercipta adalah 10x10

===========================================================================

class GamePlay

- listPlayer: berisi list player yang nanti akan digunakan pada method PlayerTurn()

-

----------------------------------

- MakeArena(): untuk membuat arena dengan ukuran tertentu, return value int untuk diisi ke class Arena()

- MakePlayer(): untuk membuat instance player1 dan player2

- PlayerTurn(): untuk mengatur giliran antara player1 dan player2

- CheckWinner(): melakukan cek variabel listSafeShip tiap pemain, jika listSafeShip dari salah satu pemain habis, maka pemain dengan listSafeShip yang masih ada akan menjadi pemenang

**COBA MEMBUAT ALUR PERMAINAN**

Class GamePlay membuat arena dengan method MakeArena() dan akan me-return value int yang akan dimasukkan ke class Arena.

Class GamePlay membuat player dengan method MakePlayer() yang berisi class Player, class Player dijalankan,

Class Player membuat 2 instance (player1 dan player2), mengatur koordinat dari seluruh kapal dengan method SetShipCoor(), koordinat tersebut dicek di method CheckShipCoor() apakah sudah ada di variabel listAllShip dan tidak melebihi variabel arena, jika tidak ada maka koordinat dapat ditambahkan di listAllShip, variabel listSafeShip berisi Salinan dari listAllShip,

player1 mengatur koordinat serang dengan method SetHitCoor(player), jika koordinat tidak ada di variabel listCoorHit maka koordinat tersebut ditambahkan ke listCoorHit, jika sudah ada maka koordinat tidak disimpan dan tidak dapat menyerang, SetHitCoor(player) menghasilkan return value(List) yang akan dilempar ke method CheckWinner(), method CheckWinner() akan mencocokan koordinat yang diterima dengan variabel listSafeShip pada player2, jika koordinat tersebut terdapat pada listSafeShip player 2, maka koordinat tersebut akan dihapus dari listSafeShip. Proses tersebut diulang bergantian hingga salah satu isi dari listSafeShip kosong, player yang masih memiliki isi dari listSafeShip menjadi pemenangnya.