

Battleship Use Cases

List of steps, user action, system response (starts with user action, list prerequisites)

1. Start Setup – Two players begin setting up a game of Battleship (includes name input)
2. Place Ships – A player places their ships onto their ship grid
3. Start Battleship Game – Both players begin the game after completing setup
4. Fire Missile at Enemy – A player attempts to attack an enemy ship
5. Hit/Miss Missile – A player either successfully or unsuccessfully hits a missile strike
6. Sink Enemy Ship – A player hits all points of a ship, causing the ship to sink
7. Surrender – A player forfeits the game by surrendering
8. Change Turn – Intermediate screen between the two players to help prevent cheating.
9. End Game – The game ends, either by a player sinking all enemy ships or by surrender

View Rules:

Preconditions – none

User opens the application	Program initiates and then displays the title screen
User clicks the rules button	Program open rules txt files showing the rules to the user

Start Setup:

Preconditions – none

User opens the application	Program initiates and then displays the title screen
User clicks the start game button	Program switches to the player name screen
User enters the name for both players then clicks the done button	Program saves the player names for both players and moves to the ship

	placement screen.
--	-------------------

Postconditions – Place Ships

Place Ships:

Preconditions – Start Setup use case

User clicks the carrier ship button and then a cell on the ship grid	Program puts the carrier on the ship grid in the specified location for that user
User clicks the battleship ship button and then a cell on the ship grid	Program puts the battleship on the ship grid in the specified location for that user
User clicks the destroyer ship button and then a cell on the ship grid	Program puts the submarine on the ship grid in the specified location for that user
User clicks the submarine ship button and then a cell on the ship grid	Program puts the destroyer on the ship grid in the specified location for that user
User clicks the patrol boat ship button and then a cell on the ship grid	Program puts the patrol boat on the ship grid in the specified location for that user
User clicks the done button	Program locks in the ship locations and saves them to that player's ship grid. Then he goes to the second player's ship setup if this was the first player or starts the game if it's the second player.

Postcondition – first player's turn

Fire Missile at Enemy:

Precondition – Start Battleship use case & is acting users turn & hasn't fired yet this turn

User selects a cell on the missile grid.	Program fires missile at opponent's ship grid.
User clicks fire missile button.	Program fires missile at opponent's ship grid. If it is on an empty cell the missile is white. If it is on a ship cell then it hits that section of the ship.

Sink Enemy Ship:

Precondition – Hit/Miss Missile use case & missile was a hit

User fires missile that hits and hits a ship that has now had all sections of it hit.	Program sets the hit ship's sunk flag to true
---------------------------------------------------------------------------------------	-----------------------------------------------

Postcondition – acting user's turn OR End Game use case

Surrender:

Precondition – Start battleship use case & Is acting user's turn

User clicks surrender button	Program causes game to end, with acting user as the loser
------------------------------	-----------------------------------------------------------

Postcondition – End Game use case

Change Turn:

Precondition – Is acting user's turn

User clicks end turn button	Program switches to the intermediate screen
Other user clicks the start turn button	Program switches to other user's turn screen

Postcondition – opponent of initial acting user's turn

End Game:

Precondition – Sink Enemy Ship use case OR Surrender use case

User sinks the last ship of their enemy	Program shows the victory screen and displays who the winner of the game is along with the last move.
-----------------------------------------	-------------------------------------------------------------------------------------------------------