Design Inspection, Code Inspection, Unit Testing

Team 20 | Thomas Chen, Kelvin Choi, Scott Merritt, Aaron Althoff, Dan Morton

Design Inspection Defects

Product	Battleship 408 Design Inspection
Date	9/20/2018
Author	Team
Moderator	Scott Merritt
Inspectors	Dan Morton, Thomas Chen, Aaron Althoff, Kelvin Choi

Defect #	Description	Severity	How it was corrected
1	New Game option from the Main Menu module should lead to the New Game Setup page, not directly to the Game module	1	New Game option now redirects to New Game Setup Page
2	In ShipSetting module, user was not notified of which ship they needed to place	1	ShipSetting module now has a textbox appear to notify the user which size ship they need to place
3	RegisterUser module does not have a input-checker on the client-side to ensure valid input	2	Verify Input component has been added to Register User module
4	In PlaceShip operation, backend does not validate ship placement overlap condition	2	Validate input for invalid ship placement on both client-side and backend.
5	Creating NewGame required at least three REST calls to initialize Player, Map, and Ships.	1	The operation should instantiate Map/Player instances together with Ship Placement in a single call to reduce complexity of NewGame operation.

6	Unique state of player not preserved on backend.	1	Add session based UUID to players on initialization of new game.

Code Inspection Defects

Product	Battleship NewGame Module Inspection
Date	9/20/2018
Author	Dan Morton
Moderator	Aaron Althoff
Inspector(s)	Scott Merritt

Defect #	Description	Severity	How it was corrected
1	ShipSetting module did not allow the user to place the 5th ship.	1	Simple issue in the code within a *for* loop has been corrected, should have been “<=” instead of “<”
2	Selecting the very last tile on the Grid caused the system to have unexpected behavior, sometimes resulting in complete system crash	2	Issue with traversing Grid array caused an ArraryIndexOutOfBounds Exception, this has been corrected.
3	Clicking the “Start Game” option led to a 404 error	1	Component for linking pages was corrected
4	User was able to enter invalid input when placing ships (non-adjacent square)	1	Logic was added for input validation
5	User was able to use same space when inputting ship locations	1	Logic was added to disallow using a space twice
6	User was able to hit an already hit coordinate of a ship.	1	Logic was added to make sure that hit cell’s were correctly marked.
7	When a user wanted to move a ship, the user could place the ship in its original location.	1	Logic was added to not let a user use the ship’s old location.

Product	Battleship Authentication Module Inspection
Date	9/20/2018
Author	Dan Morton
Moderator	Aaron Althoff
Inspectors	Scott Merritt

Defect #	Description	Severity	How it was corrected
1	User was able to access views without being authenticated	1	Added activations to all routes in application. Must be authenticated in order to visit views.
2	Unnecessary password column in database - only need to store username and email since google is handling authentication	1	Dropped password column from user model in frontend.
3	System would not redirect user to login view if unauthenticated	3	Added route redirection to login view in login activation code. (Re-directs if user not logged in).

Product	Battleship GameWindow Module Inspection
Date	9/20/2018
Author	Dan Morton
Moderator	Aaron Althoff
Inspectors	Scott Merritt

Defect #	Description	Severity	How it was corrected
1	Row and column locations being reported off by 1	1	Adjusted
2	cellClicked action was only accessible through cell component - not configurable at the grid component level	2	Added @output directives to the grid component allowing parent components to intercept the cell touched (x, y) and respond with an action

Product	Battleship Model Module Inspection
Date	9/20/2018
Author	Thomas Chen
Moderator	Aaron Althoff
Inspector(s)	Scott Merritt, Thomas Chen

Defect #	Description	Severity	How it was corrected
1	Distance function from Line2D library calculates Point distance off by 1 cell when starting with (0,0)	2	Offset output of distance value returned by the function in the Ship Constructor
2	Cell assignment of Map Class was inverted with y values being taken as x values and vice versa	3	Flipped 2D matrix assignment as Java 2D Arrays are [row][column]
3	Ship overlap validation will fail on edge cases where the line segments were co-linear	2	Used the Line2D.linesIntersect function from the Lines2D library which accounts for this edge case
4	Ship constructor allowed Ships of n length to be created even if length of specified Ship type was != n	1	Add validation that the specified coordinates form a line segment of equal length to the length of the specific ship type in the constructor
5	Player constructor should always initialize with a specific grid size and not be able to initialize without one	1	Add constructor which takes in integer value to specify grid size, overload default constructor to constant
6	In order to be able to specify Ship type of cell in return to client, Ship state must be stored somewhere in model	2	Create Ship Class that stands between Map and Player and controls Ship State as map interaction

Product	Battleship API Module Inspection
Date	9/20/2018
Author	Thomas Chen
Moderator	Aaron Althoff
Inspector(s)	Scott Merritt, Kelvin Choi

Defect #	Description	Severity	How it was corrected
1	There existed a redundant win/ POST endpoint that was unnecessary as the backend determines the win condition	1	The endpoint was removed
2	There did not exist a way for the control routes of the API to know which player was sending controls	2	Control endpoints were reworked to identify user based on the OAuth2 Bearer token attached to the request
3	On response to client request, Player was receiving full state of both player grids	2	On the backend resolution of the requestor's identity determines what board state to be returned. Game class stores a copy of each board associated to a Player with only the view of opponent boards based Player scope.
4	There was a redundant variable to determine the current player in the quit game POST endpoint.	1	The redundant variable was removed.
5	There existed a redundant Hit endpoint who’s functionality was already covered by existing Move control endpoint via POST variables	1	Removed the redundant Hit endpoint and configured Move endpoint to fully handle the same control functionality

Unit Testing Defects

Product	Battleship Menu Module Test
Date	09/20/2018
Author	Dan Morton

Defect #	Description	Severity	How it was corrected
1	Application should route to about page when button pressed	1	Added active router functionality and registered the about route in app.module
2	Application should route to settings page when button pressed	1	Added active router functionality and registered the settings route in app.module
3	Application should route to statistics page when button pressed	1	Added active router functionality and registered the statistics route in app.module
4	Application should route to create strategy page when button pressed	1	Added active router functionality and registered the create strategy route in app.module
5	Application should route to new game page when button pressed	1	Added active router functionality and registered the create strategy route in app.module
6	Application should remember where the placement of all the ships were	1	Used a ship interface to define each type of ship, then maintained a dynamic list that contains all the ships being inputted.

Product	Battleship New Game Module Test
Date	09/20/2018
Author	Dan Morton

Defect #	Description	Severity	How it was corrected
1	Application should load page when visiting /newGame	1	Route registered in app.module
2	Application should not allow users to overlap ships while inputting	1	Added logic checking placed ships when a new input is received
3	Application should not route to game window when Username and Victory message are empty	1	Made victory message and username required fields in the new game form.
4	Application should not skip spaces when inputting ships	3	Added logic to ensure next input for ship placement is an adjacent block to the existing ones.
5	Application should not allow you to input any more spaces once all ships are place	3	Boolean flag that restricts user from being able to interact with the ship placement grid once all ships placed.
6	Application should route to Game Window once valid input has been entered	1	On successful form submission, route submit() function to route to GameWindow
7	Application outputs placed ships, username and victory message in a way readable by another component	1	Formatted output as a JSON object that can be parsed by another component.

Product	Battleship Authentication Module Test
Date	09/20/2018
Author	Dan Morton

Defect #	Description	Severity	How it was corrected
1	Application should route user to main menu on successful login	1	Route registered in app.module
2	Application should stay on login page if login not successful	1	Using google sign in api - onSignIn() handler never called if api never authenticates User.
3	Application should log authentication page when visiting /login	1	Route registered in app.module

Product	Battleship Game Window Module Test
Date	09/20/2018
Author	Dan Morton

Defect #	Description	Severity	How it was corrected
1	Application should load view when visiting /gamewindow	1	Route registered in app.module
2	Application should respond with coordinates clicked when user pressed grid	1	Used index of each cell to determine positioning in a 2D array, then displayed to the user as a coordinate pair.

Product	Battleship Grid Module Test
Date	09/20/2018
Author	Dan Morton

Defect #	Description	Severity	How it was corrected
1	Component should take in an input for number of row and cols	1	Used @Input directive to allow component to take input.
2	Component should have an output that can be mapped to a function	1	Used @Output directive to allow parent components to handle on cell pressed events
3	Component should space cells in a grid and should be responsive.	1	Utilized Angular-Material mat-grid-list component to achieve effect.

Product	Battleship Model Module Test
Date	9/20/2018
Author	Thomas Chen

Defect #	Description	Severity	How it was corrected
1	Cell was printing incorrect values within the Map Class with Water cells printing as “1” instead of “0”	1	The ternary operator for the toString() function in Cell was flipped, simply reversed the values.
2	Assignment of type to a single Cell in Map was modifying the entire row of the Map	1	The Map constructor initialized cells in Cell[][] using a single for loop iterating over Cell[], switched initialization to traverse by individual cell
3	getType() function of Ship class returning null even when Ship was initialized	2	The constructor for Ship failed to initialize the type field after validating the type
4	arrayOutOfBounds exception triggering on test case testing adding of Ship to grid because of a lack of validation that Ship coordinates were within grid bounds	1	Added validation for setShip function to verify that Ship object was within grid

Product	Battleship API Module Test
Date	9/20/2018
Author	Thomas Chen

Defect #	Description	Severity	How it was corrected
1	Post statement for login was failing to return 200 Success on valid login credentials.	1	Resolved issue where login credentials for successful login were compared to incorrect authentication state.
2	Player move endpoint was capable of being successfully called even when it was not the Player’s turn.	1	Added request input validation to ensure that successive moves were not accepted and would return a 400 error.
3	Opponent ships that were hit were not being updated properly.	1	Corrected the parameters for the opponent’s ship in the POST statement to match that of the back-end resolver.