Advanced Programming Practices (SOEN 6441)

Summer 2019

Project - Build2

Battleship Game

Coding Standards

Submitted By: Team1

Aruni Patel - 40077021

Dhvani Agola - 40081593

Sanjana Udar - 40094029

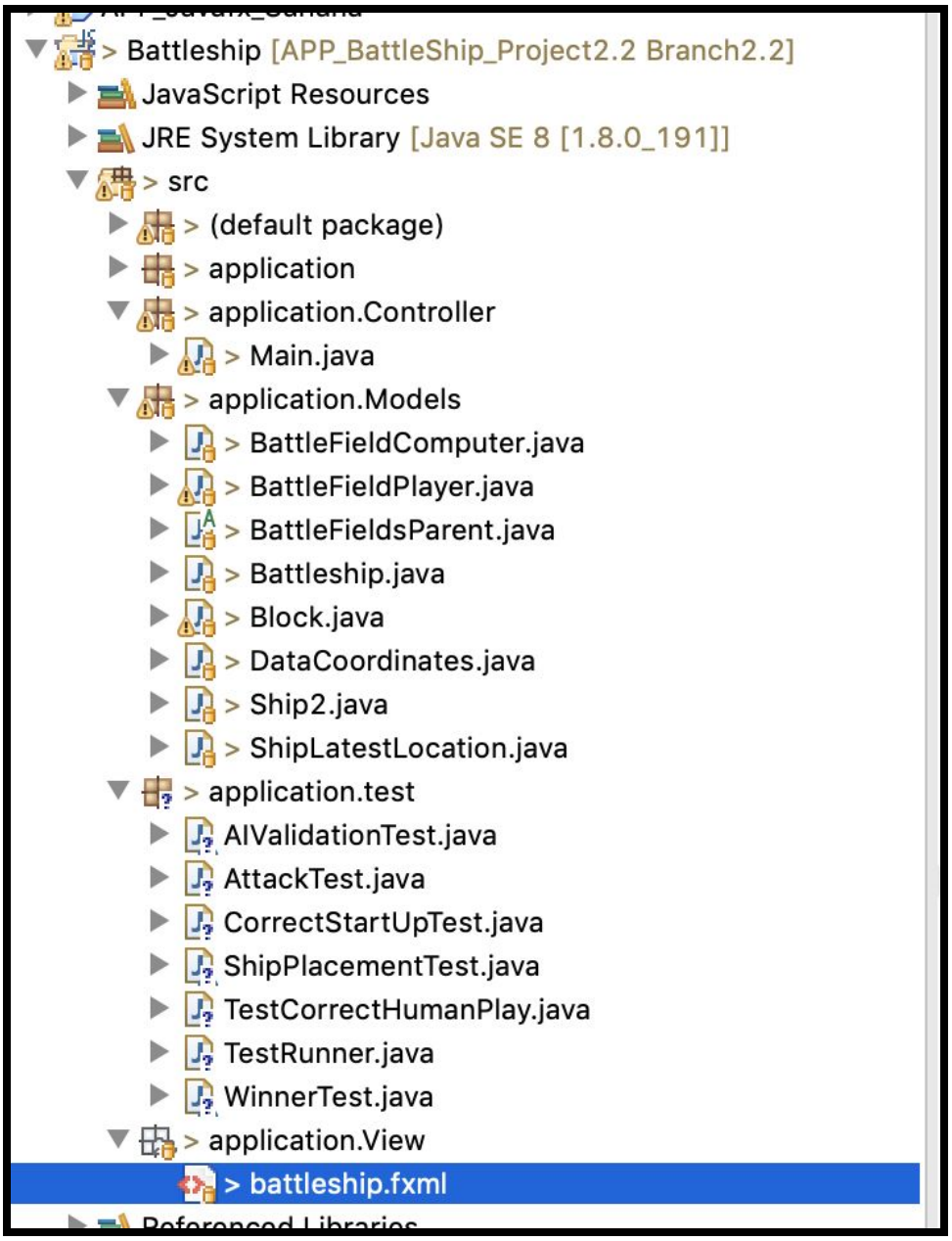
Sahil Savaliya - 40080380

Sahana Shankar - 40092026

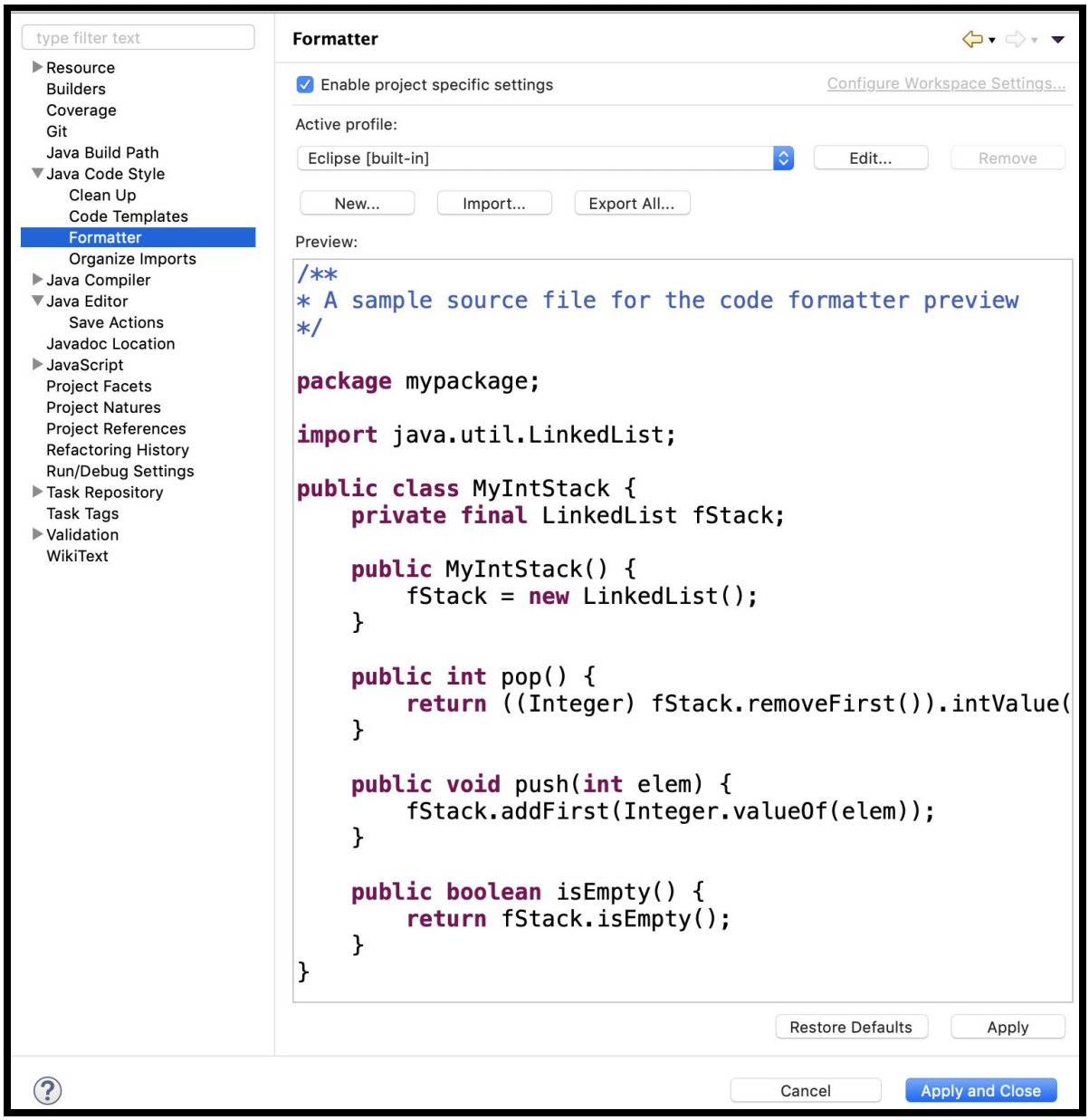
Submitted To:

Nagi Basha

- **File Structure**
    - The packages ‘Controller’, ‘Models’, ‘View’ and ‘test’ are included inside the ‘application’ package.
    - The ‘Controller’ package has controller file ‘Main.java’.
    - The ‘Models’ package has several java files which is shown in figure.
    - The ‘View’ package contains battleship.fxml file.
    - The ‘test’ package contains all unit test files.



- **Code Style :**
  - Open curly braces written in the same line of the respective code block to reduce the number of lines of code.
  - Close curly braces written in the line to overcome the issue of readability.
  - Indentation : After every open curly brace, the new code block will start by left one tab space.



- **Naming Conventions :**
  - Package Name : All package names starts with uppercase letter and the first letter of each word is in capital letter
  - Class name : All class names starts with uppercase letter and first letter each word is in capital letter.
  - Variable and parameter name : All variable names are in camelcase and starts with its datatype.
  - Method name : All method names are in camelcase.
  - Constant name : All characters are in capital letters and each word separated by underscore('_')

```
39 public class Main implements Initializable {
40     public static BattleFieldPlayer bfPlayer;
41     public static BattleFieldComputer bfComputer;
42     private double CELL_SIZE = 30.0;
43     private boolean boolIsClicked = false;
44     private boolean boolIsRequiredToRotate = true;
45     private boolean isRotated = false;
46     private Rectangle currentShip;
47     public boolean boolIsComputerTurn = false;
48     public boolean boolIsSalvaVariation = false;
49     public int intNoOfTurns = 1;
50     private long longTime = System.currentTimeMillis();
51     public static List<Rectangle> lstNodeToSelect = new ArrayList<>();
52     private HashMap<Node, ShipLatestLocation> mapShipLocation = new HashMap<>();
```

- **Comments :**
  - To fulfill the purpose of readability and creating a javadoc, we have put detailed comments over each and every class, constructor and methods.
  - Written comments to each class, constructor and methods to describe which functionality(ies) is included in the class, constructor and methods.
  - These comments includes
    - Brief description of the class, constructor and method.
    - The list of parameters and also brief description of those parameters.
    - What the method is returning.
    - Author(s) of the class.

```
178    /**
179     * This method checks that the hit is successful or missed and update the result
180     * in data structure and also update the UI.
181     *
182     * @param x       x-coordinate of hit
183     * @param y       y-coordinate of hit
184     * @param isSalva the game is normal or salva variation
185     * @param r       Rectangle on which player hit
186     * @param i       Index of the list
187     * @return Returns true is player hit on ship or on already hit place else
188     *         returns false
189     */
190    public boolean isHit(int x, int y, boolean isSalva, Rectangle r, int i) {
191        if (this.gameBoard.get(x).get(y).getCharOccupiedFor() == 'M'
192                || this.gameBoard.get(x).get(y).getCharOccupiedFor() == 'H') {
193            if (this.gameBoard.get(x).get(y).getCharOccupiedFor() == 'M') {
```

- **Absence of Commented Out Code**
  - To increase the readability we remove commented out code from all the files of the application.

BEFORE

```
347                                         - currentShip.getWidth() / 2 - CELL_SIZE / 2
348                                 endX = x;// **
349                                 endY = y + size - 1;// **
350 //                              endX = x + size - 1;// **
351 //                              endY = y;// **
352                             } else {
353                                 currentShip.setLayoutX(r.getLayoutX() + r.getParent(
354                                 currentShip.setLayoutY(r.getLayoutY() + r.getParent(
355                                 endX = x + size - 1;// **
356                                 endY = y;// **
357 //                              endX = x;// **
358 //                              endY = y + size - 1;// **
359                             }
360
361 //                          Ship2 ship = new Ship2(size, x, y, endX, endY, false);
362 //                          bfPlayer.addShip(ship, isRotated);
363 //
364 //                          if (bfPlayer.isValidToPlace(x, y, size, isRotated)) {
365 //                              bfPlayer.lstShip.add(ship);
366 ////                                 ship.setIsSet(true);
367 //                              if (bfPlayer.lstShip.size() == 5)
368 //                                  startButton.disableProperty().set(false);
369 //
370 //                          }
371
```

AFTER

```
341                         int endX, endY;
342                         if (isRotated) {
343                             System.out.println("is rotated");
344                             currentShip.setLayoutX(r.getLayoutX() + r.getParent().get
345                                     - currentShip.getWidth() / 2 + CELL_SIZE / 2);
346                             currentShip.setLayoutY(r.getLayoutY() + r.getParent().get
347                                     - currentShip.getWidth() / 2 - CELL_SIZE / 2 + 5)
348                             endX = x;// **
349                             endY = y + size - 1;// **
350                         } else {
351                             currentShip.setLayoutX(r.getLayoutX() + r.getParent().get
352                             currentShip.setLayoutY(r.getLayoutY() + r.getParent().get
353                             endX = x + size - 1;// **
354                             endY = y;// **
355                         }
356
357                     }
358
359                     anchorPane.setOnMouseReleased(null);
360
```

- **References :**
  - https://www.geeksforgeeks.org/java-naming-conventions/
  - https://battleship-game.org/en/
  - https://github.com/pip-/Battleship
  - https://stackoverflow.com/questions/10872444/mvc-pattern-in-javafx-with-scene-builder
  - https://www.thesprucecrafts.com/the-basic-rules-of-battleship-411069