**Assignment 7**

Josue Ponce

Montgomery College

4/29/2018

Author’s Note

This project report was prepared for CMSC 203 CRN #30672, taught by professor Ahmed Tarek

**Table of Contents**

GUI Test for Ship Registry Application………………………………………………………..…2

Junit Test with Provided Code from Instructor…………………………………………………....6

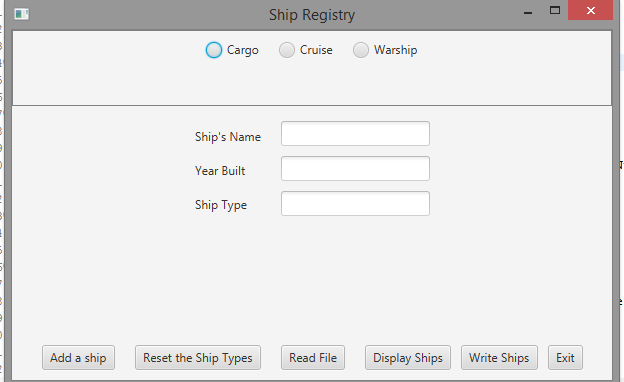
Copy of Junit Test Code Provided by Instructor……………………………………………….....7

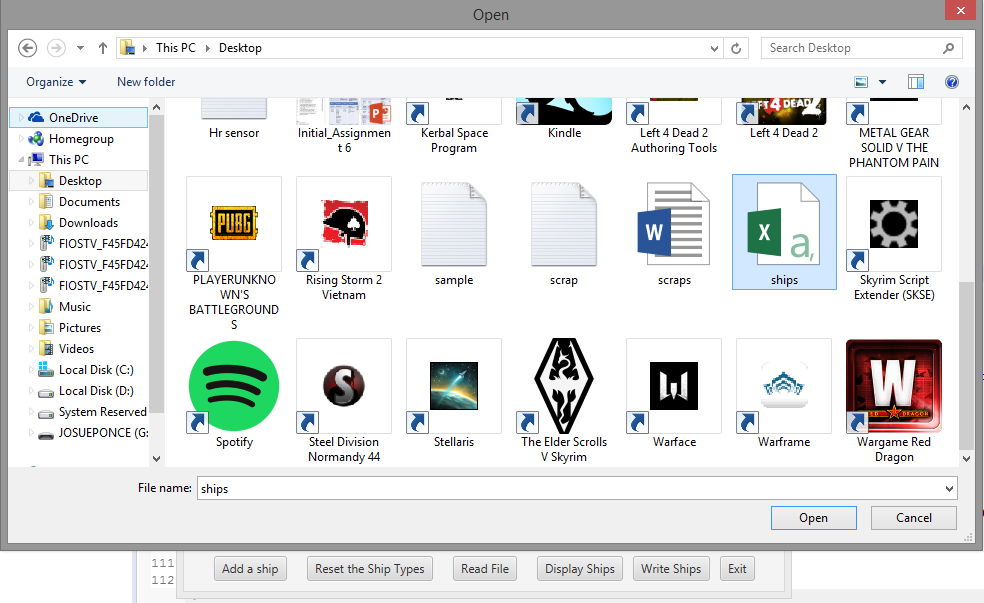
Junit Test with Custom Made Code……………………………………………………………….9

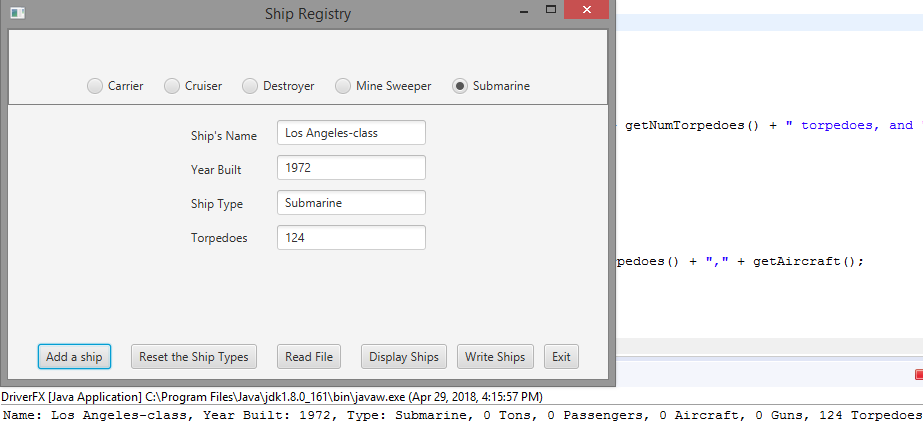
Copy of Custom Made Junit Test Code……………………………………………………...…..10

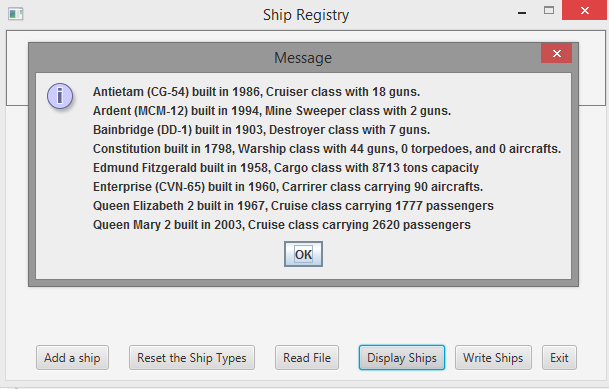
**GUI Test for Ship Registry Application**

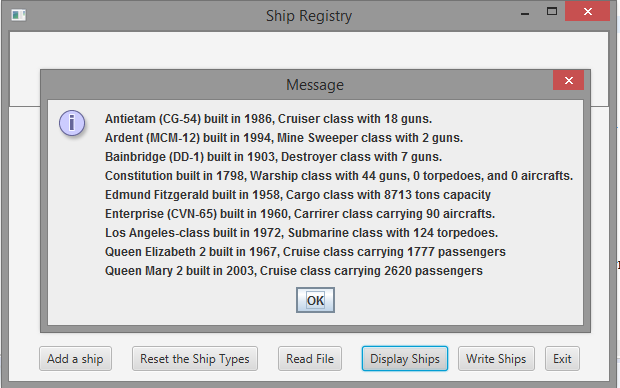
The application’s GUI was tested to ensure that the GUI for the application worked as intended and free of glitches or screen tearing. The tests conducted also ensured that the application displayed the name of ships in alphabetical order and displayed the correct results. Additionally, the buttons were also tested to ensure that the application worked flawlessly. After running the application, the application returned positive results. The application’s buttons and results all worked as intended. A visual representation of the steps taken to conduct the tests can be found down below.

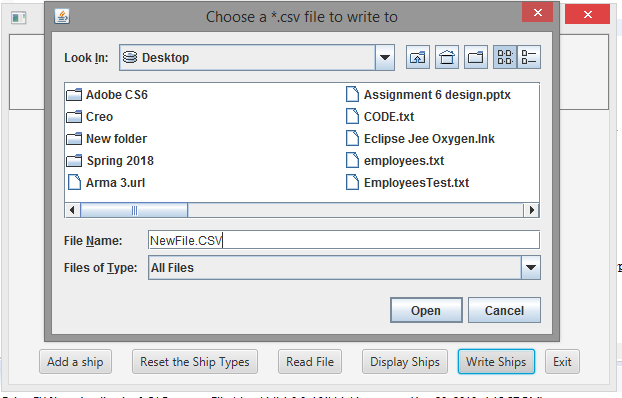


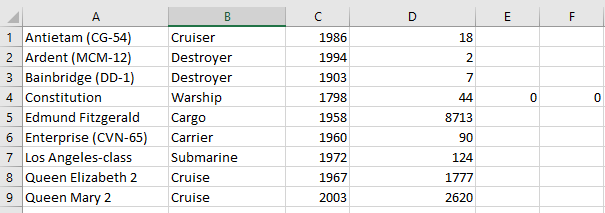






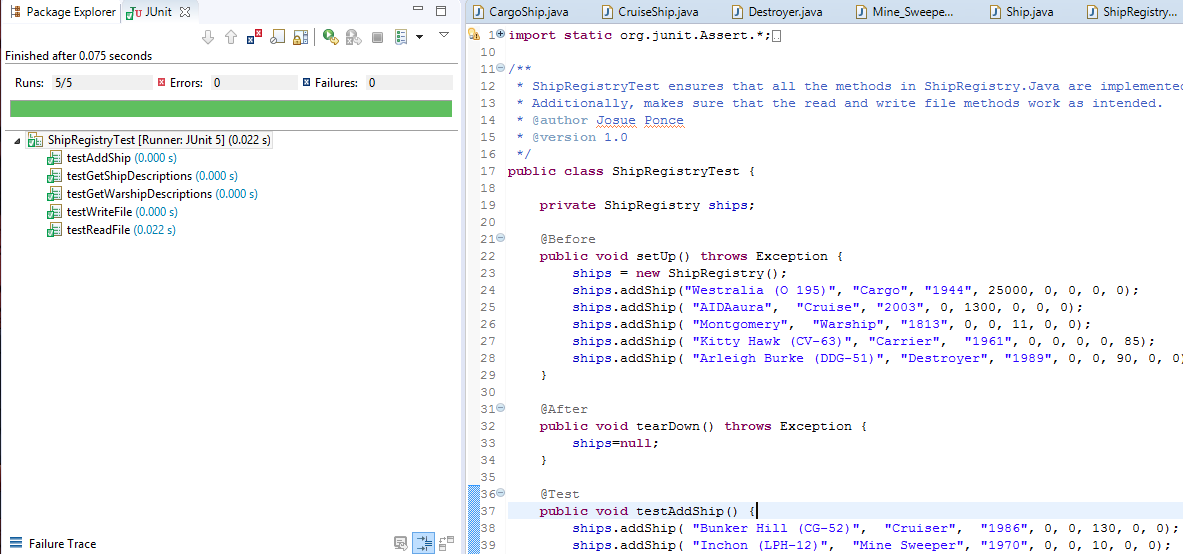






**Junit Test with Provided Code from Instructor**

A Junit test code provided by the instructor was used to test the methods implemented in ShipRegistry.Java. The tests conducted also ensured that the application can read files, write files, get ship descriptions, get warship descriptions, and add ships correctly. After running the Junit unit test on the application all the results came back positive which means that implementing the methods in the application were done correctly and worked as intended. The results from the test can be found down below (Figure 1).



*Figure 1.* Test conducted on the application and passing using the code provided by the instructor.

**Copy of Junit Test Code Provided by Instructor**

**import** **static** org.junit.Assert.\*;

**import** **java.io.File**;

**import** **java.io.FileNotFoundException**;

**import** **java.io.PrintWriter**;

**import** **java.util.ArrayList**;

**import** **javax.swing.JFileChooser**;

**import** **org.junit.After**;

**import** **org.junit.Before**;

**import** **org.junit.Test**;

/\*\*

\* ShipRegistryTest ensures that all the methods in ShipRegistry.Java are implemented correctly.

\* Additionally, makes sure that the read and write file methods work as intended.

\* @author Josue Ponce

\* @version 1.0

\*/

**public** **class** **ShipRegistryTest** {

**private** ShipRegistry ships;

**@Before**

**public** **void** **setUp**() **throws** Exception {

ships = **new** ShipRegistry();

ships.addShip("Westralia (O 195)", "Cargo", "1944", **25000**, **0**, **0**, **0**, **0**);

ships.addShip( "AIDAaura", "Cruise", "2003", **0**, **1300**, **0**, **0**, **0**);

ships.addShip( "Montgomery", "Warship", "1813", **0**, **0**, **11**, **0**, **0**);

ships.addShip( "Kitty Hawk (CV-63)", "Carrier", "1961", **0**, **0**, **0**, **0**, **85**);

ships.addShip( "Arleigh Burke (DDG-51)", "Destroyer", "1989", **0**, **0**, **90**, **0**, **0**);

}

**@After**

**public** **void** **tearDown**() **throws** Exception {

ships=**null**;

}

**@Test**

**public** **void** **testAddShip**() {

ships.addShip( "Bunker Hill (CG-52)", "Cruiser", "1986", **0**, **0**, **130**, **0**, **0**);

ships.addShip( "Inchon (LPH-12)", "Mine Sweeper", "1970", **0**, **0**, **10**, **0**, **0**);

ships.addShip( "John Adams (SSBN-620)", "Submarine", "1964", **0**, **0**, **0**, **20**, **0**);

ArrayList<Ship> shipList = ships.getShips();

assertEquals("Bunker Hill",shipList.get(**5**).toString().substring(**0**,**11**));

assertEquals(**130**,((WarShip) shipList.get(**5**)).getNumGuns());

assertEquals("Inchon",shipList.get(**6**).toString().substring(**0**,**6**));

assertEquals(**20**,((WarShip) shipList.get(**7**)).getNumTorpedoes());

}

**@Test**

**public** **void** **testGetShipDescriptions**() {

String[] shipDescr = ships.getShipDescriptions();

assertEquals("Cargo",shipDescr[**0**]);

assertEquals("Cruise",shipDescr[**1**]);

assertEquals("Warship",shipDescr[**2**]);

}

**@Test**

**public** **void** **testGetWarshipDescriptions**() {

String[] shipDescr = ships.getWarshipDescriptions();

assertEquals("Carrier",shipDescr[**0**]);

assertEquals("Destroyer",shipDescr[**2**]);

assertEquals("Submarine",shipDescr[**4**]);

}

**@Test**

**public** **void** **testReadFile**() {

**try** {

//this statement just rewrites an empty file over any existing ones

**new** **PrintWriter**("testShips.csv");

} **catch** (FileNotFoundException e) {

e.printStackTrace();

}

ships.writeFile(**new** File("testShips.csv"));

ships = **new** ShipRegistry();

ships.readFile(**new** File("testShips.csv"));

ArrayList<Ship> shipList = ships.getShips();

Ship s = shipList.get(**1**);

assertEquals("AIDAaura",s.writeData().substring(**0**,**8**));

s = shipList.get(**3**);

assertEquals("Kitty Hawk",s.writeData().substring(**0**,**10**));

}

**@Test**

**public** **void** **testWriteFile**() {

ArrayList<Ship> shipList = ships.getShips();

Ship s = shipList.get(**2**);

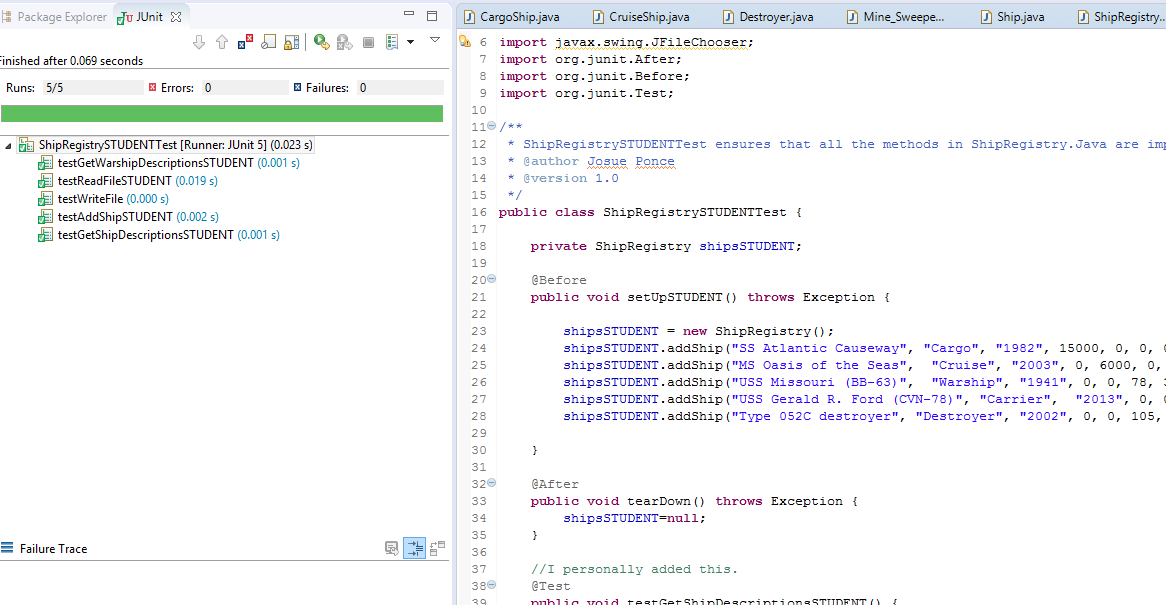
assertEquals("Montgomery",s.writeData().substring(**0**,**10**));

}

}

**Junit Test with Custom Made Code**

The methods were tested even further using a custom-made Junit test application. The second test ensured that the ship registry application can further write more ships to file, read files properly, and most importantly add more ships when needed. After the second test was conducted using the new Junit test cases, the application was still able to return positive results and all the tests passed once again. The results of the second tests can be found down below (Figure 2).



*Figure 2.* Test conducted on the application using a custom-made test code and returning positive results. All the methods have been implemented correctly.

**Copy of Custom Made Junit Test Code**

**import** **static** org.junit.Assert.\*;

**import** **java.io.File**;

**import** **java.io.FileNotFoundException**;

**import** **java.io.PrintWriter**;

**import** **java.util.ArrayList**;

**import** **javax.swing.JFileChooser**;

**import** **org.junit.After**;

**import** **org.junit.Before**;

**import** **org.junit.Test**;

/\*\*

\* ShipRegistrySTUDENTTest ensures that all the methods in ShipRegistry.Java are implemented correctly

\* @author Josue Ponce

\* @version 1.0

\*/

**public** **class** **ShipRegistrySTUDENTTest** {

**private** ShipRegistry shipsSTUDENT;

**@Before**

**public** **void** **setUpSTUDENT**() **throws** Exception {

shipsSTUDENT = **new** ShipRegistry();

shipsSTUDENT.addShip("SS Atlantic Causeway", "Cargo", "1982", **15000**, **0**, **0**, **0**, **0**);

shipsSTUDENT.addShip("MS Oasis of the Seas", "Cruise", "2003", **0**, **6000**, **0**, **0**, **0**);

shipsSTUDENT.addShip("USS Missouri (BB-63)", "Warship", "1941", **0**, **0**, **78**, **34**, **0**);

shipsSTUDENT.addShip("USS Gerald R. Ford (CVN-78)", "Carrier", "2013", **0**, **0**, **0**, **0**, **100**);

shipsSTUDENT.addShip("Type 052C destroyer", "Destroyer", "2002", **0**, **0**, **105**, **0**, **0**);

}

**@After**

**public** **void** **tearDown**() **throws** Exception {

shipsSTUDENT=**null**;

}

//I personally added this.

**@Test**

**public** **void** **testGetShipDescriptionsSTUDENT**() {

String[] shipDescr = shipsSTUDENT.getShipDescriptions();

assertEquals("Cargo",shipDescr[**0**]);

assertEquals("Cruise",shipDescr[**1**]);

assertEquals("Warship",shipDescr[**2**]);

}

//I added this.

**@Test**

**public** **void** **testGetWarshipDescriptionsSTUDENT**() {

String[] shipDescr = shipsSTUDENT.getWarshipDescriptions();

assertEquals("Carrier",shipDescr[**0**]);

assertEquals("Destroyer",shipDescr[**2**]);

assertEquals("Submarine",shipDescr[**4**]);

}

**@Test**

**public** **void** **testAddShipSTUDENT**() {

shipsSTUDENT.addShip("USS Port Royal (CG-73)", "Cruiser", "1992", **0**, **0**, **70**, **0**, **0**);

shipsSTUDENT.addShip("USS Pivot (AM-276)","Mine Sweeper", "1943", **0**, **0**, **95**, **0**, **0**);

shipsSTUDENT.addShip("Akula", "Submarine","1976",**0**,**0**,**0**,**88**,**0**);

shipsSTUDENT.addShip("USS George H.W. Bush (CVN 77)","Carrier","2009",**0**,**0**,**0**,**0**,**123**);

ArrayList<Ship> shipList = shipsSTUDENT.getShips();

assertEquals("USS Port Royal (CG-73)",shipList.get(**5**).toString().substring(**0**,**22**));

assertEquals("USS Pivot (AM-276)",shipList.get(**6**).toString().substring(**0**,**18**));

assertEquals("Akula",shipList.get(**7**).toString().substring(**0**,**5**));

assertEquals(**123**,((WarShip)shipList.get(**8**)).getAircraft());

}

**@Test**

**public** **void** **testReadFileSTUDENT**() {

**try** {

//this statement just rewrites an empty file over any existing ones

**new** **PrintWriter**("testShips.csv");

} **catch** (FileNotFoundException e) {

e.printStackTrace();

}

shipsSTUDENT.writeFile(**new** File("testShips.csv"));

shipsSTUDENT = **new** ShipRegistry();

shipsSTUDENT.readFile(**new** File("testShips.csv"));

ArrayList<Ship> shipList = shipsSTUDENT.getShips();

Ship s = shipList.get(**0**);

assertEquals("SS Atlantic Causeway",s.writeData().substring(**0**,**20**));

s = shipList.get(**3**);

assertEquals("USS Gerald R. Ford (CVN-78)",s.writeData().substring(**0**,**27**));

}

**@Test**

**public** **void** **testWriteFile**() {

ArrayList<Ship> shipList = shipsSTUDENT.getShips();

Ship s = shipList.get(**2**);

assertEquals("USS Missouri (BB-63)",s.writeData().substring(**0**,**20**));

}

}