User Guide

Data Used

1. **City of Windsor:**
   a. All the information regarding the hospitals was used from the City of Windsor Open Data website. The file name is: **Hospitals.csv**
2. **Windsor-Essex County Health Unit:**
   a. To get the information about COVID-19 in Windsor, Ontario our group used the WECHU to get all the information about COVID-19 cases.

Classes:

1. Test Class

This is the driving class for our program. It contains the main function that will instantiate an Information object and call upon the getInformation() method to get the open data tool started.

2. Information Class

After the data tool has started, the getInformation() method is called and prompts the user with four different options. First option being the number of confirmed cases in Windsor, Ontario by date. Second is the case counts and the statistics. Third being the hospitals and information about them and lastly the option to exit the tool. The user will type the number corresponding to the option to select it and the program will display the information. For options one and three the user must the file path to load the file and display the information. If the user selects options from one to three and it displayed the information, then the program asks if they would like to see the menu again or to exit.

3. CasesFile Class

After selecting option 1 and entering the file path an object CasesFile is instantiated in the Information class and method getCasesFromFile() is called. Once this method is called in the CasesFile class I used a BufferedReader to read the information from the CSV file and store it into a String array. After this it displays the total cases with the date in which the cases were confirmed.

4. HospitalFile Class

Like the CasesFile class this will do the same exact operation except it will take information form the String array and call up the hospitalRecord method to create a new object of type HospitalInformation and add that to an ArrayList of type HospitalInformation. Once this is done, it calls on the toString from the HospitalInfromation object and prints each hospital with all the necessary information in Windsor, Ontario by calling the printHospitalInformation method.

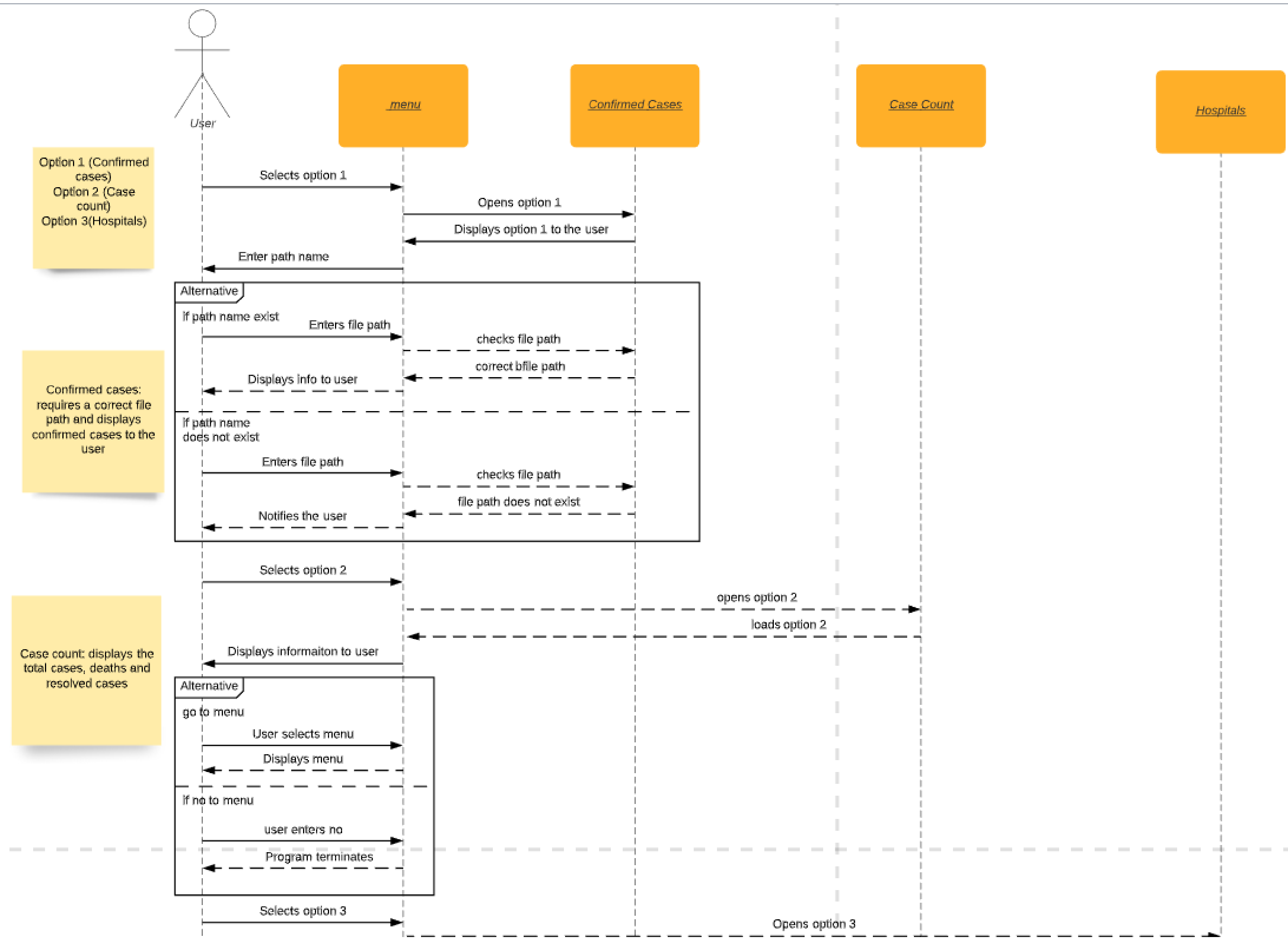
Furthermore, within the printHospitalInformation method the user will be asked if they would like to calculate the distance from their location to any Hospital the select. If they do proceed, the user has to enter their X and Y coordinate. After this, the distance from their location to the hospital they selected is calculated and displayed.

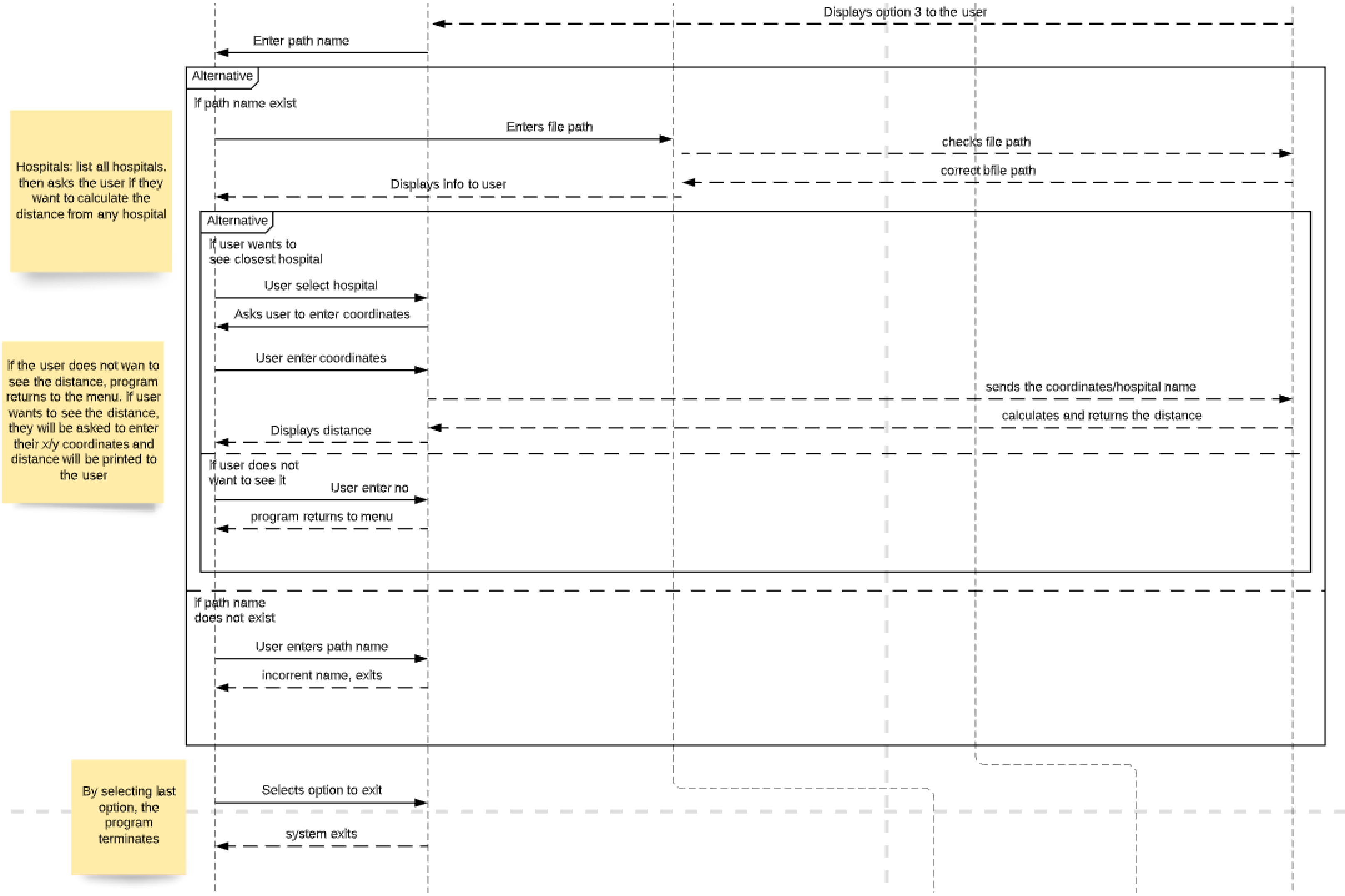
5. HospitalInformation Class

This class contains information about the Hospital that is in the Hospitals.csv file. In the HospitalFile class when the method hospitalRecord instantiates an object of HospitalInformation it sends all the information contained within the Hospitals.csv file to contractor of HospitalInforomation class to initialize all the information.

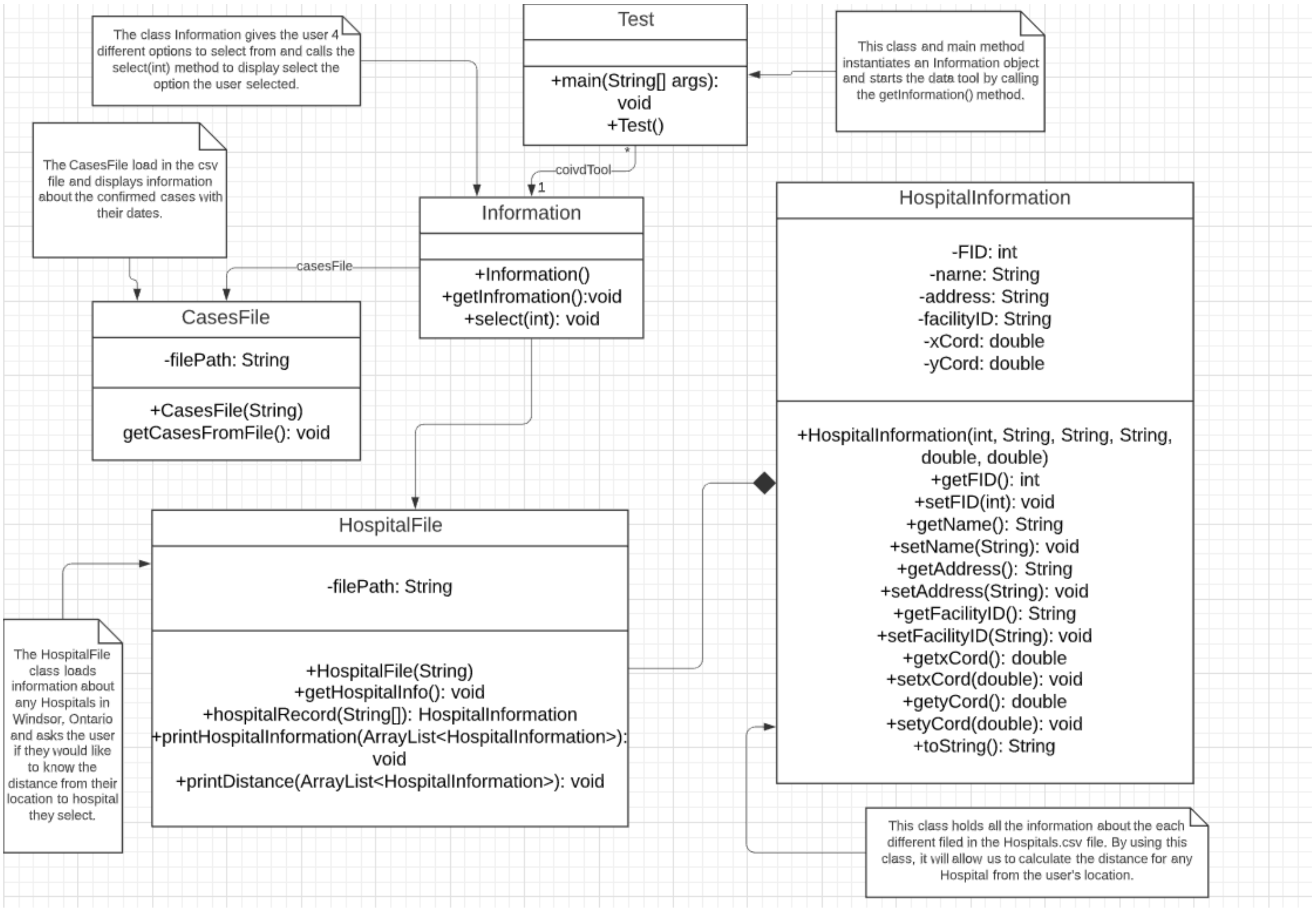
Sequence and Class Diagrams

Sequence Diagram:



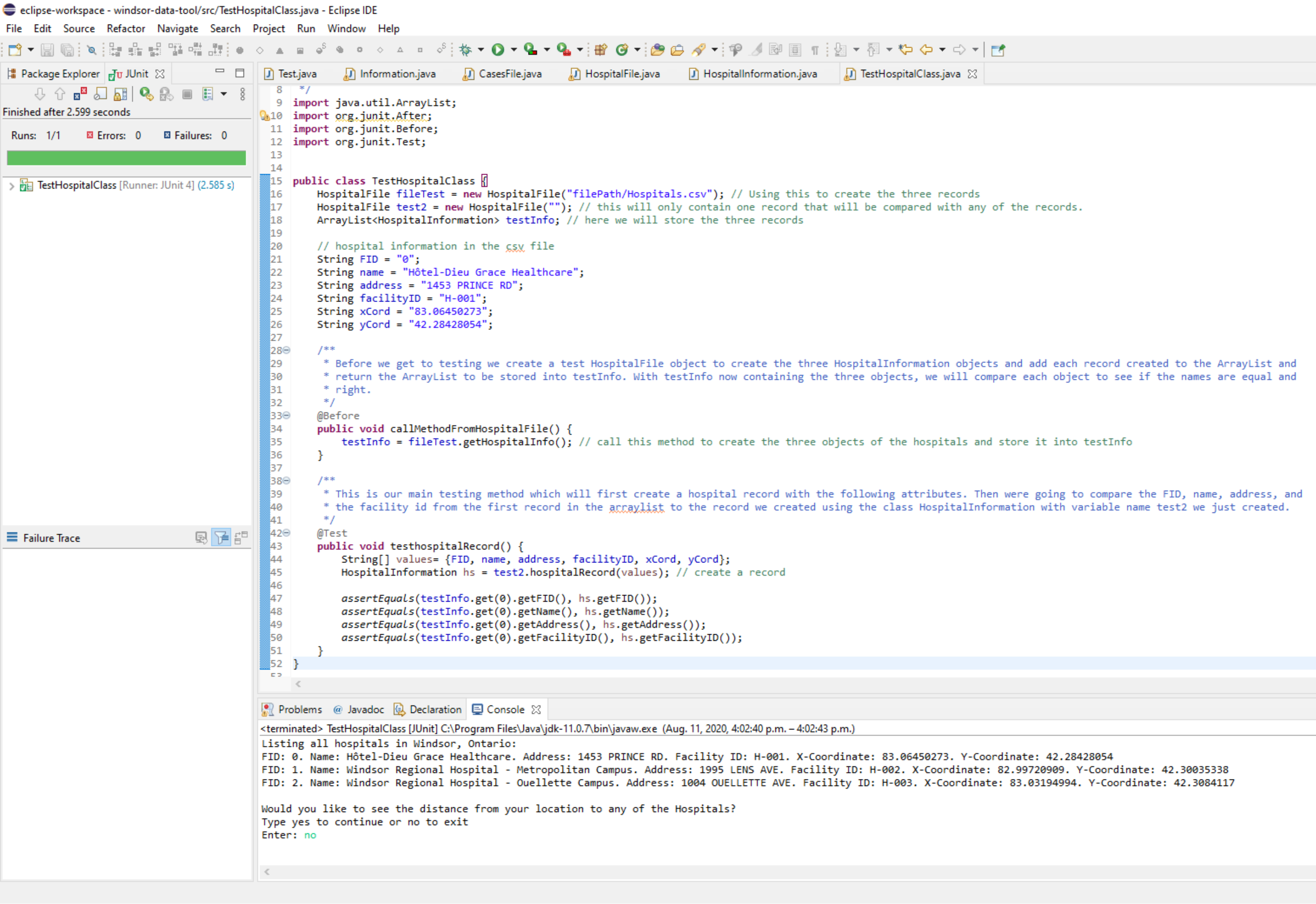

Displays option 3 to the user
Enter path name
Alternative
If path name exist
Enters file path
checks file path
correct bfile path
Displays info to user
Alternative
If user wants to see closest hospital
User select hospital
Asks user to enter coordinates
User enter coordinates
sends the coordinates/hospital name
calculates and returns the distance
Displays distance
If user does not want to see it
User enter no
program returns to menu
If path name does not exist
User enters path name
incorrent name, exits
Selects option to exit
system exits
Hospitals: list all hospitals. then asks the user if they want to calculate the distance from any hospital
If the user does not wan to see the distance, program returns to the menu. If user wants to see the distance, they will be asked to enter their x/y coordinates and distance will be printed to the user
By selecting last option, the program terminates

Class Diagram


The class Information gives the user 4 different options to select from and calls the select(int) method to display select the option the user selected.
Test
+main(String[] args): void
+Test()
This class and main method instantiates an Information object and starts the data tool by calling the getInformation() method.
The CasesFile load in the csv file and displays information about the confirmed cases with their dates.
coivdTool
*
1
Information
+Information()
+getInfromation():void
+select(int): void
casesFile
CasesFile
-filePath: String
+CasesFile(String)
getCasesFromFile(): void
HospitalInformation
-FID: int
-name: String
-address: String
-facilityID: String
-xCord: double
-yCord: double
+HospitalInformation(int, String, String, String, double, double)
+getFID(): int
+setFID(int): void
+getName(): String
+setName(String): void
+getAddress(): String
+setAddress(String): void
+getFacilityID(): String
+setFacilityID(String): void
+getxCord(): double
+setxCord(double): void
+getyCord(): double
+setyCord(double): void
+toString(): String
HospitalFile
-filePath: String
+HospitalFile(String)
+getHospitalInfo(): void
+hospitalRecord(String[]): HospitalInformation
+printHospitalInformation(ArrayList<HospitalInformation>): void
+printDistance(ArrayList<HospitalInformation>): void
The HospitalFile class loads information about any Hospitals in Windsor, Ontario and asks the user if they would like to know the distance from their location to hospital they select.
This class holds all the information about the each different filed in the Hospitals.csv file. By using this class, it will allow us to calculate the distance for any Hospital from the user's location.

Testing Strategy: Unit Testing with Junit

We tested to see whether the information in the Hospitals.csv file was correct. As you can see below our test case did pass.



Finished after 2.599 seconds

Runs: 1/1 Errors: 0 Failures: 0

TestHospitalClass [Runner: JUnit 4] (2.585 s)

```java
 */
import java.util.ArrayList;
import org.junit.After;
import org.junit.Before;
import org.junit.Test;


public class TestHospitalClass {
    HospitalFile fileTest = new HospitalFile("filePath/Hospitals.csv"); // Using this to create the three records
    HospitalFile test2 = new HospitalFile(""); // this will only contain one record that will be compared with any of the records.
    ArrayList<HospitalInformation> testInfo; // here we will store the three records

    // hospital information in the csv file
    String FID = "0";
    String name = "Hôtel-Dieu Grace Healthcare";
    String address = "1453 PRINCE RD";
    String facilityID = "H-001";
    String xCord = "83.06450273";
    String yCord = "42.28428054";

    /**
     * Before we get to testing we create a test HospitalFile object to create the three HospitalInformation objects and add each record created to the ArrayList and
     * return the ArrayList to be stored into testInfo. With testInfo now containing the three objects, we will compare each object to see if the names are equal and
     * right.
     */
    @Before
    public void callMethodFromHospitalFile() {
        testInfo = fileTest.getHospitalInfo(); // call this method to create the three objects of the hospitals and store it into testInfo
    }

    /**
     * This is our main testing method which will first create a hospital record with the following attributes. Then were going to compare the FID, name, address, and
     * the facility id from the first record in the arraylist to the record we created using the class HospitalInformation with variable name test2 we just created.
     */
    @Test
    public void testhospitalRecord() {
        String[] values= {FID, name, address, facilityID, xCord, yCord};
        HospitalInformation hs = test2.hospitalRecord(values); // create a record

        assertEquals(testInfo.get(0).getFID(), hs.getFID());
        assertEquals(testInfo.get(0).getName(), hs.getName());
        assertEquals(testInfo.get(0).getAddress(), hs.getAddress());
        assertEquals(testInfo.get(0).getFacilityID(), hs.getFacilityID());
    }
}
```

<terminated> TestHospitalClass [JUnit] C:\Program Files\Java\jdk-11.0.7\bin\javaw.exe (Aug. 11, 2020, 4:02:40 p.m. – 4:02:43 p.m.)

```
Listing all hospitals in Windsor, Ontario:
FID: 0. Name: Hôtel-Dieu Grace Healthcare. Address: 1453 PRINCE RD. Facility ID: H-001. X-Coordinate: 83.06450273. Y-Coordinate: 42.28428054
FID: 1. Name: Windsor Regional Hospital - Metropolitan Campus. Address: 1995 LENS AVE. Facility ID: H-002. X-Coordinate: 82.99720909. Y-Coordinate: 42.30035338
FID: 2. Name: Windsor Regional Hospital - Ouellette Campus. Address: 1004 OUELLETTE AVE. Facility ID: H-003. X-Coordinate: 83.03194994. Y-Coordinate: 42.3084117

Would you like to see the distance from your location to any of the Hospitals?
Type yes to continue or no to exit
Enter: no
```