[Year]

[Document title]

[DOCUMENT SUBTITLE]

BARNABE MALANDAIN

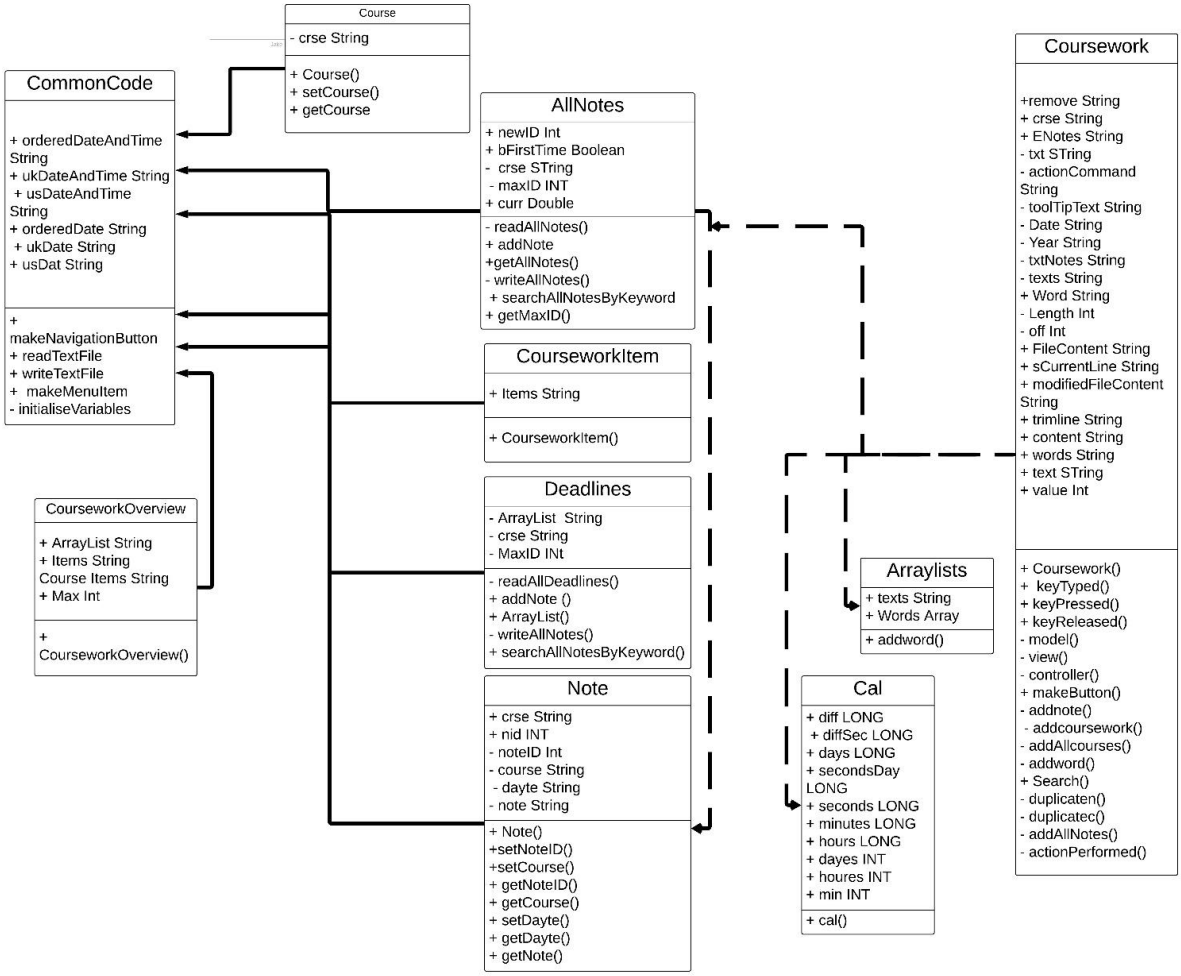
Side note
In hackathon 25 march I did task2 because we forgot to comment which part we did

1 Introduction

At the very core of its design the application is meant to be used for note taking and classification at the same time you create a note choose a course so that your notes are stored in an organized way. With this base some improvements were added the original way of getting the date wasn't properly working so we changed it, we also added a research bar to look for specific string throughout the courses notes and count the number of time that specific string is used and display the searched string with your colour of choice and police of choice. All of that using array lists and display windows, the course system to add delete or modify a course for instance is used via a drop-down menu for; the toolbar as for the notes they are all stored in .txt files. One of our latest improvement is the deadline setting for each course the user can set up a date for a deadline from then a countdown will launch and when only a week will be left a reminder will pop on warning the user of its imminent demise or end of project for the course (similar concept). Another one of the improvement added was the option to see on what date the user wrote the most notes

2 Class Diagram

2.1 The Class Diagram (image created with Daniels Magonis and Jake Nelms)



2.2 A Brief Explanation of the Diagram

We can see that everything is pretty much linked to the main code coursework

3 The code for the Extra Classes Created

3.1 addWord

```
private void addword() throws IOException {

String texts = JOptionPane.showInputDialog("Write text to add!");
words.add(texts);
Object[] result = words.toArray();

System.out.println(words);

}
```

Output a display box where the user can type in a string input which will be the text to add then we put the word in text which is an array list the System.out.println(words) output the word in the shell

3.2 search(in coursework)

```
public void Search() {
 Scanner user_input = new Scanner( System.in );
   String Word;
   Word = JOptionPane.showInputDialog(" Write the word you want to search!");
   Set<String> set = new HashSet<String>(words);

   if (set.contains(Word))
   {
      JOptionPane.showMessageDialog(null, "Word is found! Proceed to editing!");
     String[] Font= {" Georgia ", " Arial "};
 String[] Backgroundcolour= {"green", "yellow", "red"};
 String[] Size={"10" , "12", "14", "40"};

 JComboBox jcd = new JComboBox(Font);
 JComboBox jcm = new JComboBox(Backgroundcolour);
 JComboBox jcy = new JComboBox(Size);
  jcd.setEditable(true);
  jcm.setEditable(true);
  jcy.setEditable(true);
 JOptionPane.showMessageDialog( null, jcd, "Set Font",
JOptionPane.QUESTION_MESSAGE);
 JOptionPane.showMessageDialog( null, jcm, "Set BackGround colour",
JOptionPane.QUESTION_MESSAGE);
 JOptionPane.showMessageDialog( null, jcy, "Size", JOptionPane.QUESTION_MESSAGE);

    String font = jcd.getSelectedItem().toString();
 String colour= jcm.getSelectedItem().toString();
 String sizes = jcy.getSelectedItem().toString();
 JOptionPane.showMessageDialog(this,
      "<html><font face =" + font + "size =" + sizes + " color = " + colour + ">" + Word );
}


   else{
      JOptionPane.showMessageDialog(null,"String is not Found, try to add it First");

   }
```

We use a scanner to see the user input we then create the variable word and create a dialogue box where the user will input the word he desire to search . we then use a loop with 2 option, option 1 being if the set has the word researched in and option 2 being if the word is not here . We then have multiple windows appearing letting the user choose the police of character the colour and the back ground colour to output the word searched as for option 2 it just output that the word have not been found .

3.3

3.4 cal

```
/**
 *
 * @author Daniels Magonis- dm5376y, Barnabe Malandain, Jake Nelms.
 */
import java.util.Date;
import java.util.Calendar;

public class cal {
public static int SECONDS_IN_A_DAY = 24 * 60 * 60;
public static void main(String[] args) {
   Calendar thatDay = Calendar.getInstance();
   thatDay.setTime(new Date(0)); /* reset */
   thatDay.set(Calendar.HOUR_OF_DAY,15);/*here Add ur Time */
   thatDay.set(Calendar.DAY_OF_MONTH,10);
   thatDay.set(Calendar.MONTH,11); // 0-11 so 1 less
   thatDay.set(Calendar.YEAR, 2020);
   Calendar today = Calendar.getInstance();
   long diff =  thatDay.getTimeInMillis() - today.getTimeInMillis();
   long diffSec = diff / 1000;
   long days = diffSec / SECONDS_IN_A_DAY;
   long secondsDay = diffSec % SECONDS_IN_A_DAY;
   long seconds = secondsDay % 60;
   long minutes = (secondsDay / 60) % 60;
   long hours = (secondsDay / 3600); // % 24 not needed
   int dayes=(int)days;
   int houres=(int)hours;
   int min=(int)minutes;

int num = dayes * (24*60*60) + houres * (60*60) + min * 60 ;

   System.out.println(num);
}
}
```

In courseword we adapted it like this

```
String year = jcd.getSelectedItem().toString();
      int y = Integer.parseInt(year);
   String month= jcm.getSelectedItem().toString();
   int m = Integer.parseInt(month) - 1;
   String day = jcy.getSelectedItem().toString();
   int d = Integer.parseInt(day);
   int SECONDS_IN_A_DAY = 24 * 60 * 60;
   Calendar thatDay = Calendar.getInstance();
   thatDay.setTime(new Date(0)); /* reset */
   thatDay.set(Calendar.DAY_OF_MONTH,d);
   thatDay.set(Calendar.MONTH,m); // 0-11 so 1 less
   thatDay.set(Calendar.YEAR, y);
   Calendar today = Calendar.getInstance();
   long diff =  thatDay.getTimeInMillis() - today.getTimeInMillis();
   long diffSec = diff / 1000;
   long days = diffSec / SECONDS_IN_A_DAY;
   long secondsDay = diffSec % SECONDS_IN_A_DAY;
   long seconds = secondsDay % 60;
   long minutes = (secondsDay / 60) % 60;
   long hours = (secondsDay / 3600); // % 24 not needed
   int dayes=(int)days;
   int houres=(int)hours;
   int min=(int)minutes;


String coursework = year + "-" + month + "-" + day;
   int num = dayes * (24*60*60) + houres * (60*60) + min * 60 ;
   if (num <= 604800) {
      JOptionPane.showMessageDialog(rootPane," Your nearest CourseWork
deadline is on  " + coursework);
   }
   System.out.println(num);
```

We set up using calendar we set up hours day month and years we have them in longs so we had to change them in integer and to managed to do a week deadline we putted everything in second and used an if loop saying that if the number outputted was less or equal to the number of second in a week to output a reminder and to improve it we could have added this bit of code to do a countdown during the week

```
int num = dayes * (24*60*60) + houres * (60*60) + min * 60 ;

System.out.println(num);
if (num <= 604800) {

    //long num2 = thatDay.getTimeInMillis() - today.getTimeInMillis();
    System.out.printf("%d days, %d hours, %d minutes and %d seconds\n", days, hours, minutes, seconds);

    //System.out.println(num2);
}
```

4 White Box Testing

4.1 cal

Case#	Action	Inputs	Expected output	Actual output	Pass/fail
1	Add hour	integer	hours	hours	pass
2	Add month	integer	month	month	pass
3	And year	integer	year	year	pass

4.2 deadline

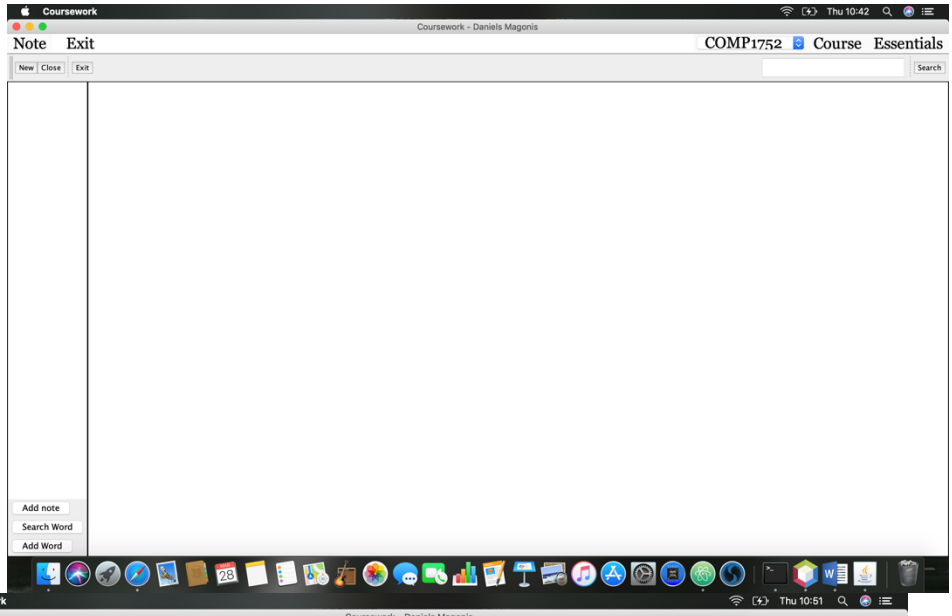
Case#	Action	Inputs	Expected output	Actual output	Pass/fail
1	Add course	string	Course name	Course name	pass
2	Add time ect	integer	Hour month year	Hour month year	passs

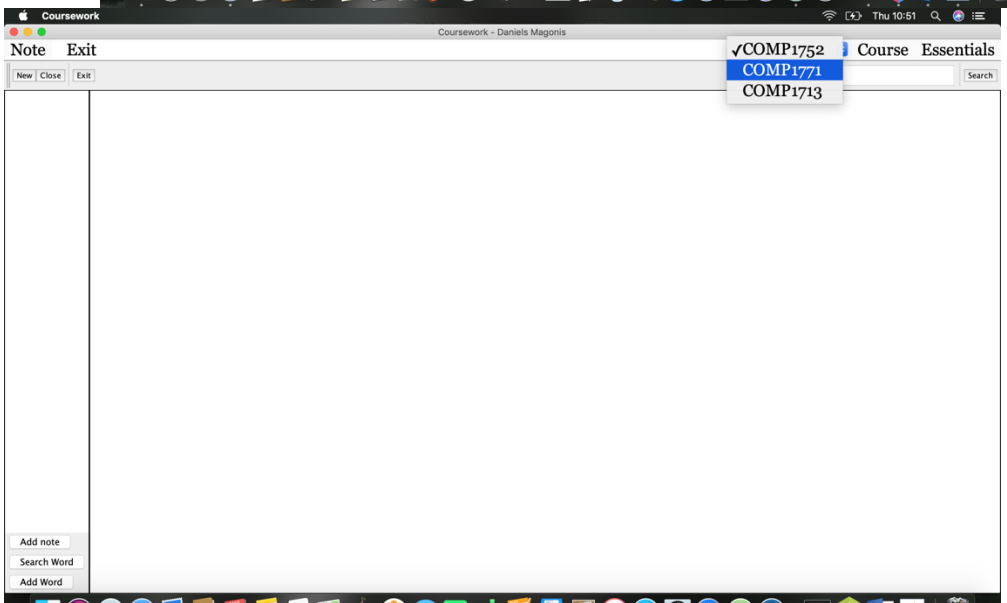
4.3 search

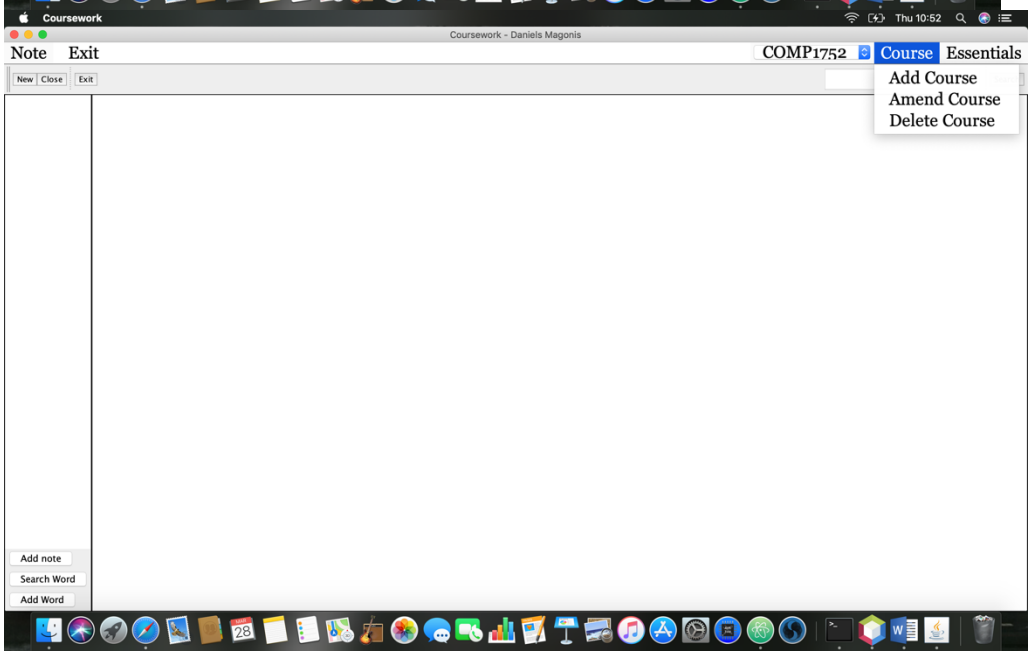
Case#	Action	Inputs	Expected output	Actual output	Pass/fail
1	Add word searched	string	String word found	String word found	pass

5 Screen Shots of the Program Working

5.1


Coursework
Thu 10:42
Coursework - Daniels Magonis
Note Exit
COMP1752 Course Essentials
New Close Exit
Search
Add note
Search Word
Add Word

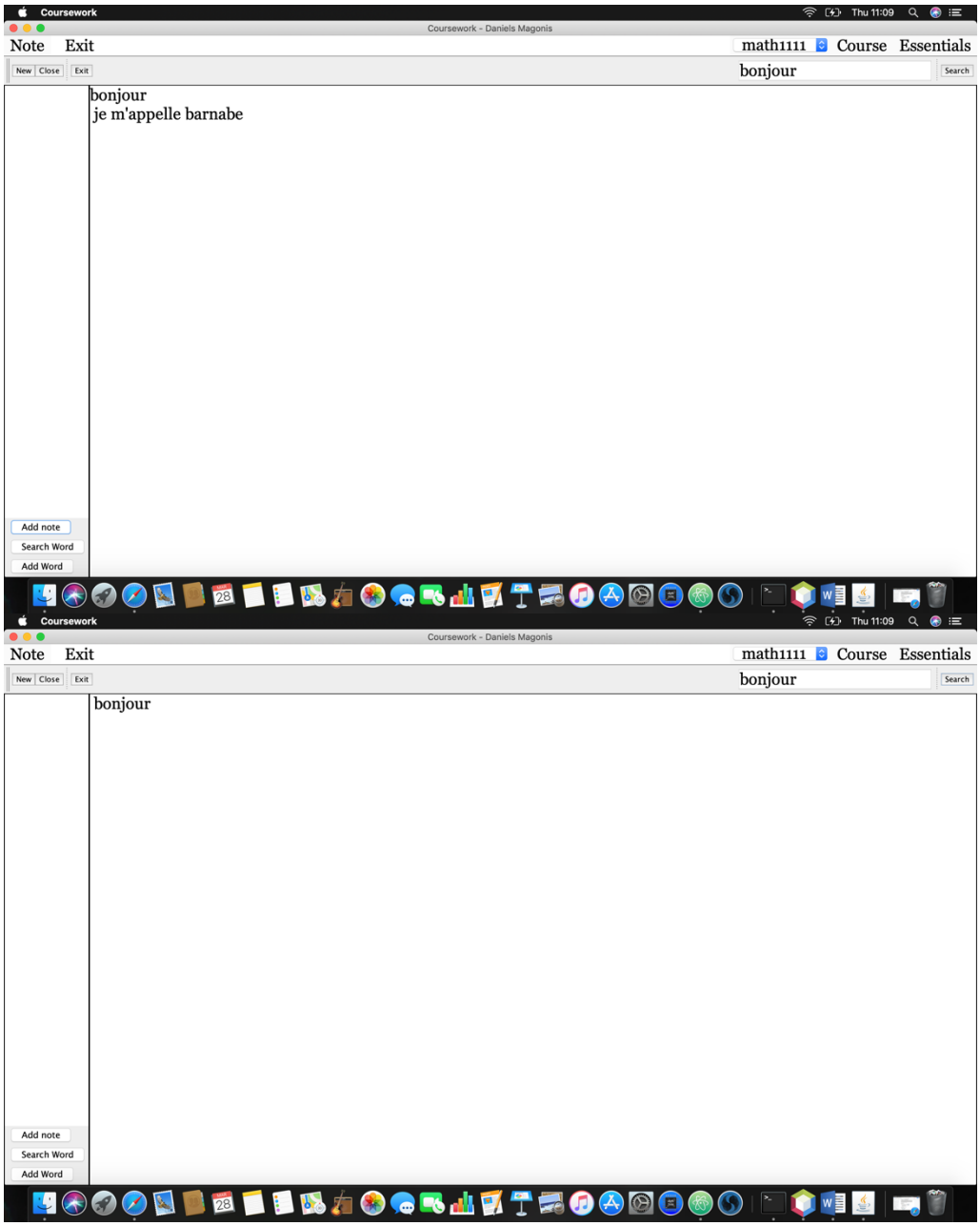

Coursework
Thu 10:51
Coursework - Daniels Magonis
Note Exit
✓COMP1752
COMP1771
COMP1713
Course Essentials
New Close Exit
Search
Add note
Search Word
Add Word


Coursework
Thu 10:52
Coursework - Daniels Magonis
Note Exit
COMP1752 Course Essentials
Add Course
Amend Course
Delete Course
New Close Exit
Add note
Search Word
Add Word

5.2

As you can see the display work perfectly apart from the icon but it s because I am on mac and the path is different the macOS uses / as for windows it s \ but apart from that the drop down menu work and the buttons works

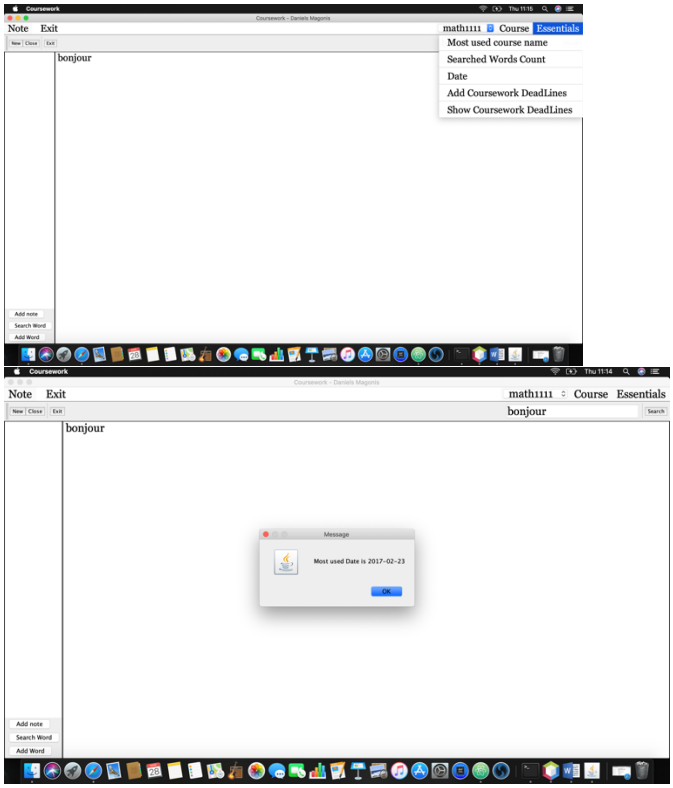
5.3



5.4

The search functionality is outputting the searched string so it s working as intented

5.5



5.6

Date is giving the date so it ‘s working as well

6 The Evaluation

6.1 What went well?

The relationship with the group is solid and the work done together is great we really get along well the work distribution when tasks are asked of us went well everyone was able to participate and give a part of himself to the project which is good morally as well as for improving skills. I consider myself as the weakest member of the group, but I was never cast out and I was able to help participate and do my part in the work that was asked of us.

6.2 What went less well?

The Hackathon even though it went well the pressure putted on us during a short amount of time like that may have cause a few problem but nothing unsolvable even though I think some tasks were more fair than others and I am not speaking within the group I am speaking within the class for instance the task asked of our group from 10 to 12 seemed pretty doable to us and we did it but speaking with other classmate some of the other subject in other groups seemed quite hard. Linking our work to the same code can be sometimes messy as well. Working alone on my code was hard not during the hackathon but during the regular labs because, I don’t feel like I learned that much on java we copy pasted a lot (not speaking about the hackathon) from what was given to us in the lab tasks, I realized then that I wasn’t understanding most of the code even though I read it watched

andy's video etc so I choose to restart from the beginning to try and understand what I didn't get it helped but I think the best way to learn a language is to code by yourself which I also did on code academy but it's wasn't enough I hope in the future exercice to learn a new language will be more similar to what we did in python in the first semester than copy and pasting bits of codes.

6.3 How would you rate your application

I would rate the application as solid which mean it's not pretty but it work if we had more time we could have added a few improvement make it more flexible and removing all this white which is awful for the eyes, this would have been a nice improvement most editor are in grey and the character in lighter grey which is really better for the eyes it's a detail but it could make life easier for the user. We could as well improve the interface and make it more user friendly but like I said the important bits work

6.4 What was learned ?

Sadly, I don't feel like I learned a lot even though I tried after it is my opinion it might be false and I might have learned more than I realized. I definitely learned more about Object Oriented Programming since it was my first time doing any, I learned a bit about java and its grammar but the concepts in programming are often the same. I really hope we will work again on java in a different way so that I can test myself more and see if I know how to really code in java or not but I will finish the java course on code academy to at least recheck is I have the bases.

6.5 How would a similar task be completed differently?

There are as many way of completing a task as there are of people, but it's not completely true cause even if everyone code differently code obey to math and logic reasoning there are indeed multiply way of solving the problem but in every code solving the problem the concepts used will be the same. If I were to use a metaphor a similar task would be completed differently because everyone is different and some might start by doing the display menu some might start by doing the array lists etc in very different manner but the tools used to do the display menu and the array lists will be the same.

7 Self-Grading

Please assess yourself objectively for each section shown below and then enter the total mark you expect to get.

Viva 1 (5)

Not attended or no work demonstrated – 0
Work demonstrated was not up to the standard expected – 1 to 2
Work demonstrated was up to the standard expected – 3 to 4
Work demonstrated exceeded the standard expected – 5

Viva 2 (5)

Not attended or no work demonstrated – 0
Work demonstrated was not up to the standard expected – 1 to 2
Work demonstrated was up to the standard expected – 3 to 4
Work demonstrated exceeded the standard expected – 5

For this section I think I got: 7 out of 10

Features implemented (15)

Required features (up to 10)

None of the required features implemented – 0
Partial implementation but with some bugs – 1 to 3
Partial implementation that is fully functioning – 4 to 6
All features implemented but with some bugs – 7 to 9
All features implemented and fully functioning – 10

Additional feature (implemented at hackathon) (up to 5)

None of the required features implemented – 0
Partial implementation but with some bugs – 1 to 2
Partial implementation that is fully functioning – 2 to 3
All features implemented but with some bugs – 3 to 4
All features implemented and fully functioning – 5

For this section I think I got: 9 out of 15

Professional programming style (5)

Coding (up to 2.5)

Indenting has not been used – 0
Indenting has been used occasionally – 0.5
Indenting has been used, but not regularly – 1
Indenting has been used regularly – 1.5 to 2
All code has been indented correctly – 2.5

Naming of variables, procedure and classes (up to 2.5)

Professional naming has not been used – 0
Professional naming has been used occasionally – 0.5
Professional naming has been used, but not regularly – 1
Professional naming has been used regularly – 1.5 to 2
All items have been named professionally correctly – 2.5

For this section I think I got: 4 out of 5

Use of OOP techniques (40)

Abstraction (15)

No extra classes or objects have been created – 0
Classes and objects have been created superficially – 1 to 3
Classes and objects have been created and used correctly – 4 to 7
New and useful classes and objects have been created – 8 to 11
The use of classes and objects exceeds the specification – 12 to 15

Encapsulation (10)

No encapsulation has been used – 0
Class variables have been encapsulated superficially – 1 to 3
Class variables have been encapsulated correctly – 4 to 6
The use of encapsulation exceeds the specification – 7 to 10

Inheritance (10)

No inheritance has been used – 0
Classes have been inherited superficially – 1 to 3
Classes have been inherited correctly – 4 to 6
The use of inheritance exceeds the specification – 7 to 10

Polymorphism (5)

No polymorphism has been used – 0
A procedure has been polymorphised – 1
A procedure has been polymorphised and used appropriately – 2 to 3
The use of polymorphism exceeds the specification – 4 to 5

For this section I think I got: 24 out of 40

Testing (10)

Testing has not been demonstrated in the documentation – 0
Little white box testing has been documented – 1 to 3
White box testing has been documented for all the coursework – 4 to 6
White box testing has been documented for the whole program – 7 to 10

For this section I think I got: 6 out of 10

Evaluation (10)

No evaluation was shown in the documentation – 0
The evaluation shows a lack of thought – 1 to 3
The evaluation shows thought – 4 or 5
The evaluation shows clearly demonstrates increased awareness – 6 to 10

For this section I think I got: 7 out of 10

Documentation (10)

The documentation cannot be understood on first reading – 0
The documentation is readable, but a section(s) are missing – 1 to 3
The documentation is complete – 4 to 6
The documentation is complete and of a high standard – 7 to 10

For this section I think I got: 7 out of 10

I think my overall mark would be: 50 out of 100