

Criterion B - Solution Overview

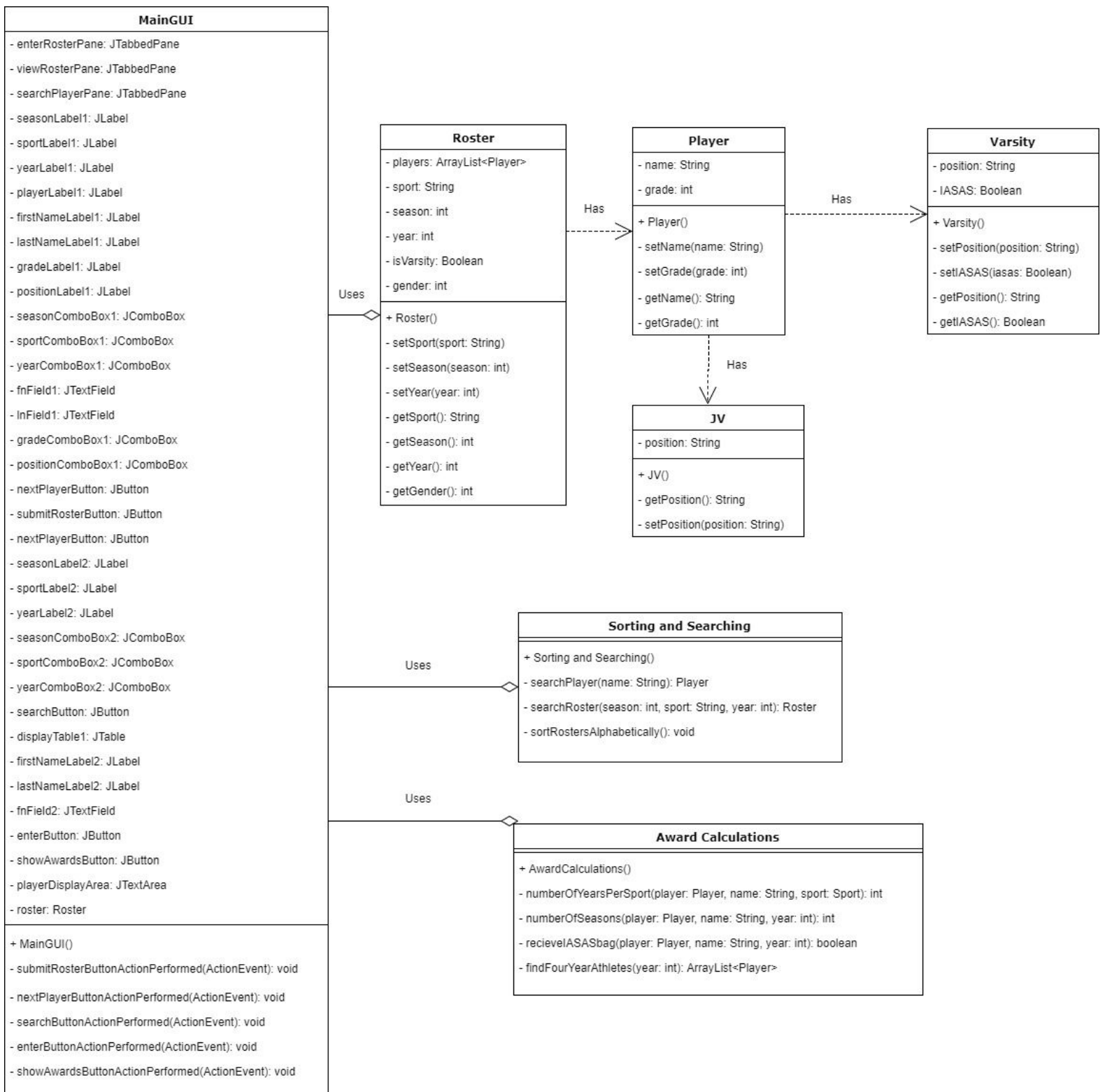
Input & Output

<i>Input</i>	<i>Data Type</i>	<i>Example</i>
Season	Integer	"2"
Sport	String	"Basketball"
Year	Integer	"2017"
First Name	String	"John"
Last Name	String	"Smith"
Grade	Integer	"12"
Position	String	"Captain"

<i>Output</i>	<i>Data Type</i>	<i>Example</i>
Season	String	"1"
Sport	String	"Basketball"
Year	String	"2017"
First Name	String	"John"
Last Name	String	"Smith"
Grade	String	"12"
Position	String	"Captain"
Award	Boolean	"Yes"
Help	String	"You may select one, two, or all three of the parameters above the table for search for each."

Word Count: 77

UML:



Chronological Development (Pseudocode is not included in word count):

1. Code data types and implement inheritance

- Construct Player Datatype
- Implement JV and Varsity as subclasses to the Player superclass

Pseudocode for inheritance

construct Player

initialize Name

initialize grade

abstract JV extends Player

send Name to Player

send grade to Player

initialize Position

abstract Varsity Extends Player

send Name to Player

send grade to Player

initialize position

initialize isLASAS

2. Accepting input to create a roster

- New player data type for each entry
- Player added into new arrayList
- arrayList stored with relevant data

3. Writing to the file

- Create, or modifying an ArrayList of Rosters
- Write ArrayList to file

Pseudocode

construct holder arrayList

name file

construct object output stream to file

try connecting stream to file

try writing to file

populate holder arrayList

catch

output to user that there was an error writing to file

catch

output to user that there was an error writing to file

4. Sort ArrayList of Rosters by JV/Varsity

- File read
- Each roster in the file ArrayList accessed
- Iterate through the file ArrayList
- Sort so that Varsity is placed after JV

Pseudocode

file read

while boolean finished is false

for int i is less than size of ArrayList

for (int a equals i + 1) is less than size of ArrayList - 1

if roster at index i of ArrayList is V

if roster at index a of ArrayList is JV

temp equals index roster at index i of ArrayList

index i of ArrayList equals index a

index a equals temp

finished is false

else

finished = true

end if

end if

break loop

end loop

end loop

end loop

5. Sort ArrayList of Rosters by name

- File read
- Each Roster in the file ArrayList accessed
- Bubble sort used to alphabetically sort each Roster
 - Apply counter from sorting between JV and Varsity

Pseudocode

file read

for each index of the first arrayList

for when i is less than the size of the roster - 1 in the first arrayList

```

        for when a is less than the size of the arrayList - i - 1
            if player name at index a + 1 of roster is before that at index a of roster
                then temp equals player at index a of the roster
                index a of the roster equals index a + 1 of roster
                index a + 1 equals temp
            end if
        end loop
    end loop
end loop
rewrite to file

```

6. Searching for a player

- File read
- ArrayList of Rosters sorted
- Input obtained from user for player parameters
- Parameters used to discard irrelevant data to narrow search
- Player found
- Player displayed to user with player's attributes and additional information

Pseudocode

```

implement sort file method
open file
read arrayList in file
for each roster of each index in the first arrayList of the file
    while low index is less than equal to high index
        mid equals (low + high)/2
        if name at index mid of roster is after search
            then high equals mid - 1
        else if name at index mid of roster is before search
            then low equals mid + 1
        else
            player instance added to separate arrayList
        end if
    end loop
end loop
return arrayList of searched player instances
display information in text area

```

7. Finding number of years per sport for a player

- Player searched for

- Instances of name tallied
- Tally displayed to user

Pseudocode

Implement search player method

```

while roster count is greater than or equal to zero
  while sport of roster is equal to sport inputted by user
    if player found
      then increment counter
    end loop
    if file of rosters is finished being read
      display counter to user
      break
    end loop
  end loop
end loop

```

8. Determining whether an IASAS bag is due

- Search for player
- Add year for each instance of player found to an array
- Check array size
- Determine for which sizes is an IASAS bag due

Pseudocode

implement search player method

```

while roster count is greater than or equal to zero
  if player name found
    if player is IASAS
      then add year of player instance to array1
    end loop
  end loop
  if file of rosters is finished being read
    break
  end loop
end loop
switch for array1 size
  case 1
    IASAS bag given
  case 3
    IASAS bag given
end switch

```

9. Finding fourth year athletes

- Implement the number of years per sport method
- Conditional statement, checking 4 year or no

Pseudocode

implement number of years per sport per player method

if number is 4

then return boolean 4year as true

else

then return boolean 4year as false

end loop .

Word Count: 227

Testing Plan

*Note that N/A is due to the use of Combo Boxes, and therefore input type is given to the user.

<i>Input</i>	<i>Normal</i>	<i>Border</i>	<i>Abnormal</i>	<i>Extreme</i>
Season	int: 2	N/A*	N/A*	N/A*
Sport	String: Basketball	N/A*	N/A*	N/A*
Year	int: 2017	N/A*	N/A*	N/A*
First Name	String: John	No Name - Results in a JOptionPane message asking for re-entry.	Single Letter - Results in a JOptionPane asking for re-entry.	Integer - Results in a JOptionPane asking for re-entry. Symbol - Question mark, exclamation mark, etc, results in a JOptionPane asking for re-entry.
Last Name	String:	No Name	Same as First	Integer

	Smith	- Results in a JOptionPane message asking for re-entry.	Name - Results in JOptionPane confirming entry. If entry confirmed, program continues. Else user is allowed to enter another name.	- Results in a JOptionPane asking for re-entry. Symbol - Question mark, exclamation mark, etc, results in a JOptionPane asking for re-entry.
Grade	int: 12	N/A*	N/A*	N/A*
Position	String: Captain	N/A*	N/A*	N/A*

Word Count: 172