SOFTWARE ENGINEERING FOUNDATION

CourseWork 2

ARUNESH, SHREYAS
sa2009@hw.ac.uk

1. Technical section

- Jframe- An extended version of java.awt.Frame that adds support for the JFC/Swing component architecture is used to implement the concept of GUI where the listener interface for receiving action events. The class that is interested in processing an action event implements this interface, and the object created with that class is registered with a component, using the component's addActionListener method.
- A JTextArea is a multi-line area that displays plain text.It is intended to be a lightweight component that provides sourcecompatibility with the java.awt.TextArea class where it canreasonably do so. Scrol panel provides a scrollable view of a lightweight component.
- A JScrollPane manages a viewport, optionalvertical and horizontal scroll bars, and optional row andcolumn heading viewports.
- JButtons can be configured, and to some degree controlled, by Actions. Using an Action with a button has many benefits beyond directlyconfiguring a button.
- A border layout lays out a container, arranging and resizingits components to fit in five regions:north, south, east, west, and center.Each region may contain no more than one component, andis identified by a corresponding constant: NORTH, SOUTH, EAST, WEST, and CENTER.
- The abstract method for calculating the 0verall score is different for these classes. The logic used are:

  Average of top n scores where n is the level number- For chess Player

  Weighted average of all scores (choose different weights for different scores and levels)- for Karate Player

  Average of scores disregarding the highest and lowest score – for CounterStrike player.

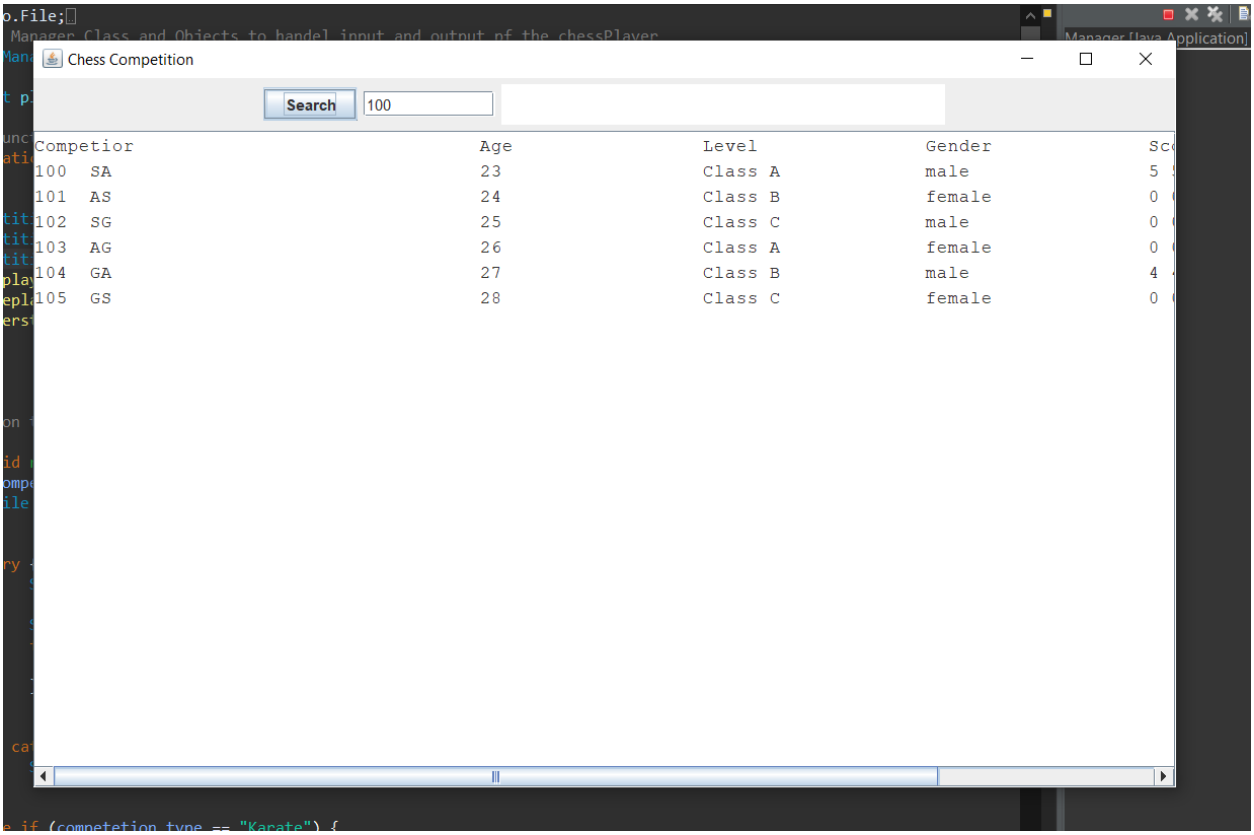

Figure: GUI Implementation of Chess competitor

- The Figure shown above depicts us the GUI implementation of the Chess competitor. When the manager opens the application and passes the parameter as "Chess", the input data is read from the file chessPlayerinput.txt and displays the short details of all the Chess competitors like Chess competitor number, Initials of the competitor Name, age, level, Gender, overall score of that competitor in the center panel of the window. These details are inherited from the competitor class.
- The Search window on the north pannel inputs the file from the user and passes that to the the competitor list class where the method called findChessPlayernumber is called in the application. The player found will be displayed in the outputArea where the message Player found will be displayed else a pop up warning message will be produced to display the Player not found.
- Similarly the same implementation is done to GUI Implementation of Karate competitor and GUI Implementation of CounterStrike competitor

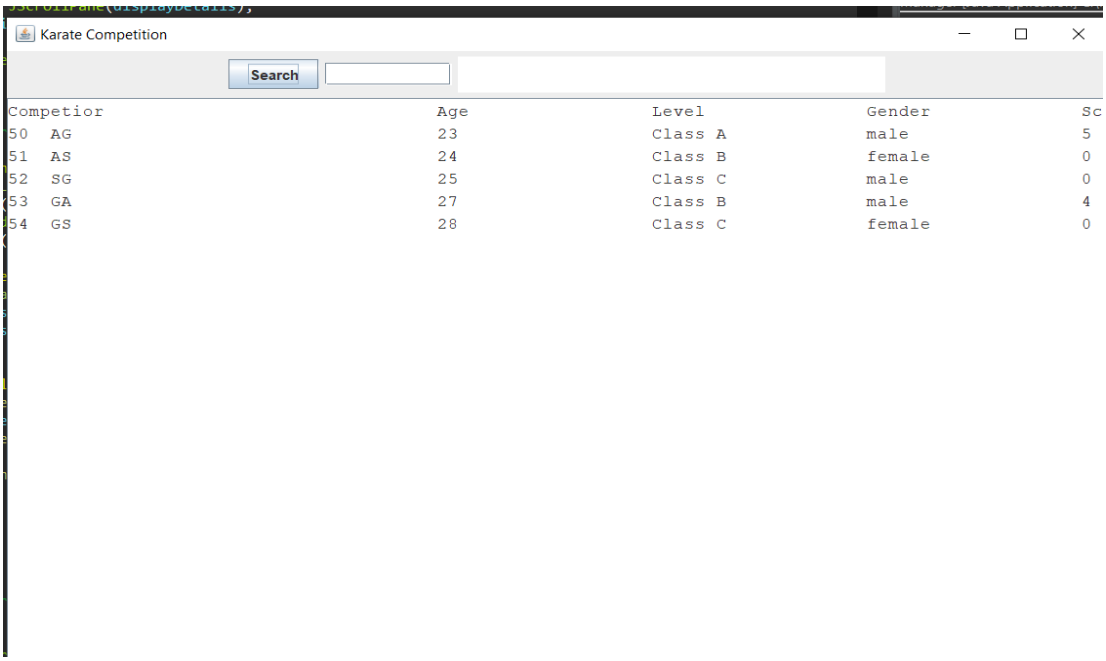

Figure: GUI Implementation of Karate competitor

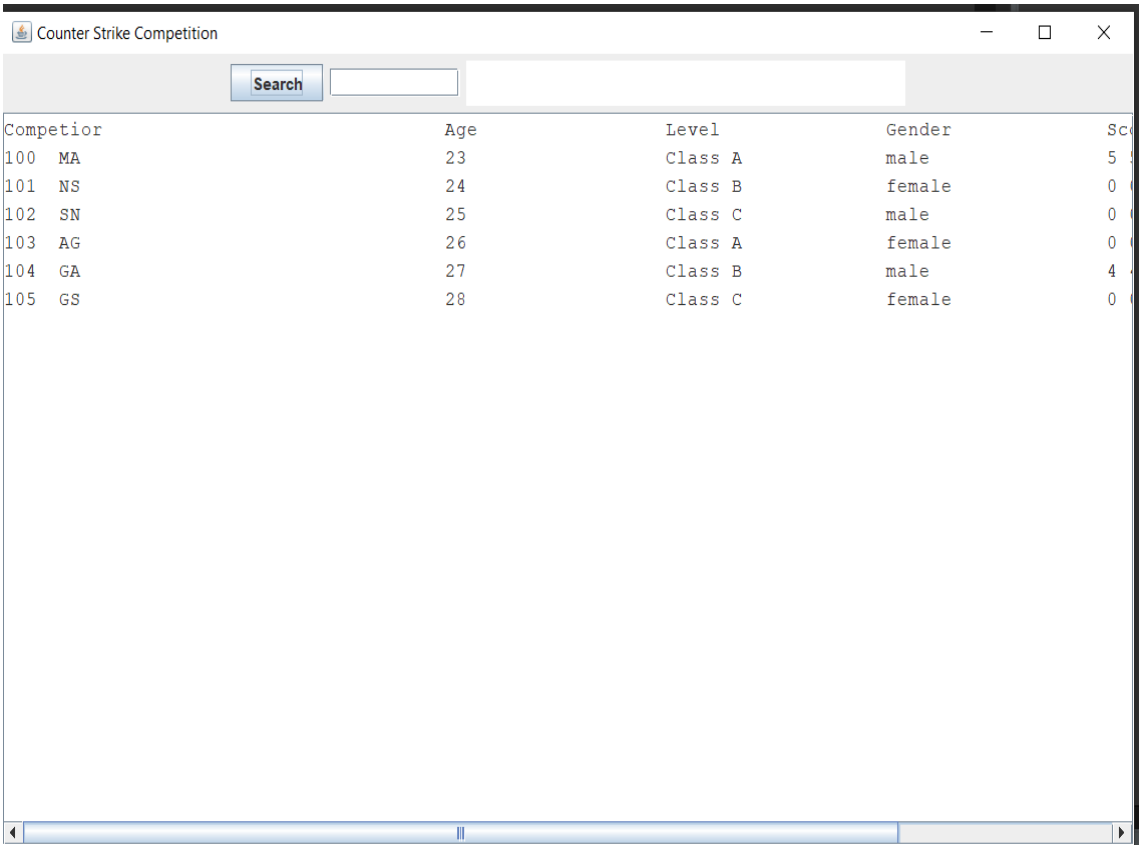

Figure: GUI Implementation of CounterStrike competitor

2. A status section:

Operations	Observations
Search button for chess player	FindChessplayer function not implemented
Search button for Karate Player	FindkaratePlayer function not implemented
Search button for CounterStrike Player	FindCounterStrikeplayer function not implemented

The application partially satisfies the given specifications for GUI. The sections that are not implemented are shown in the above table.

3. Class Diagram:

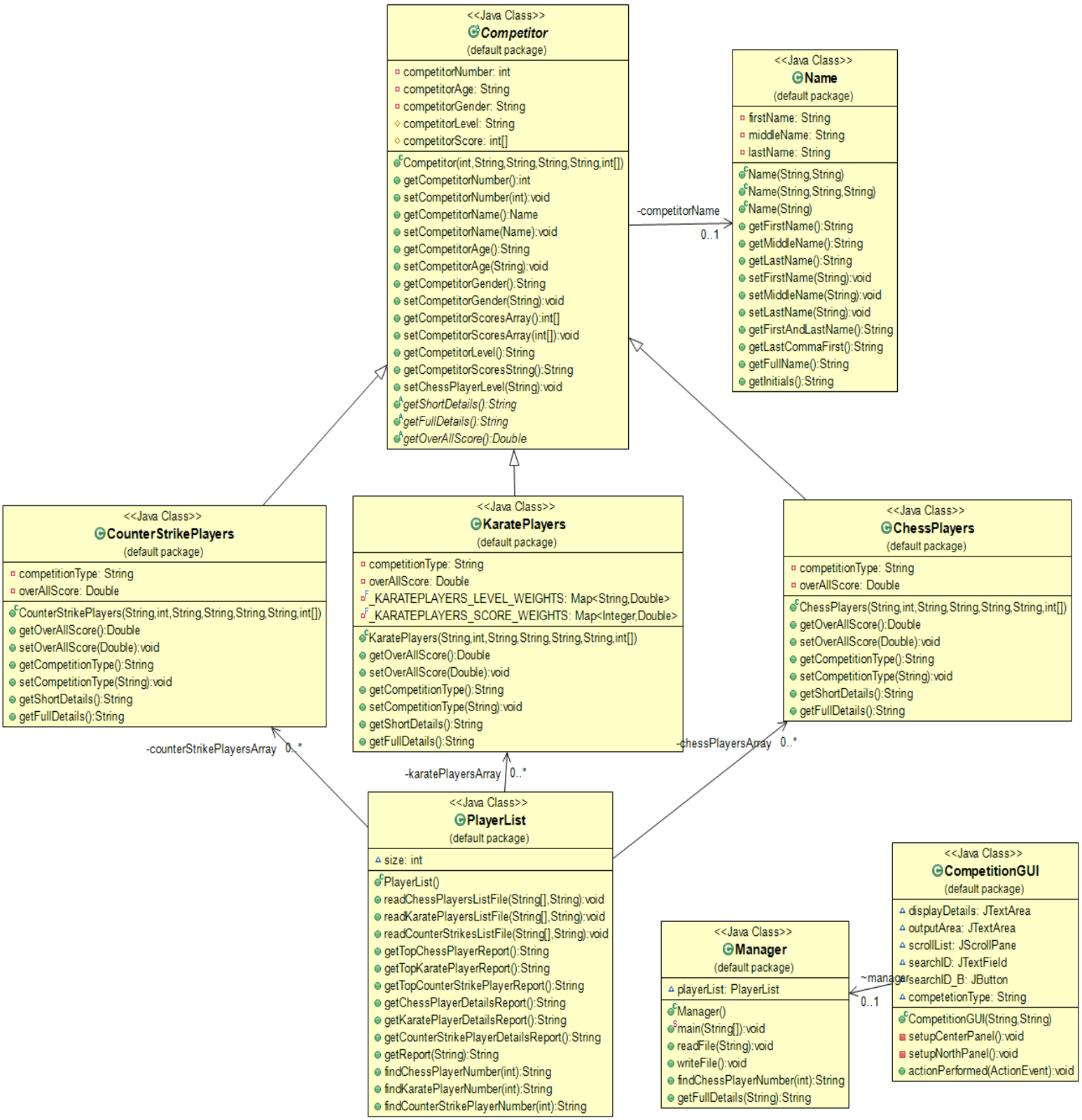

Figure: Class Diagram of the National competition application

4. Activity Diagram

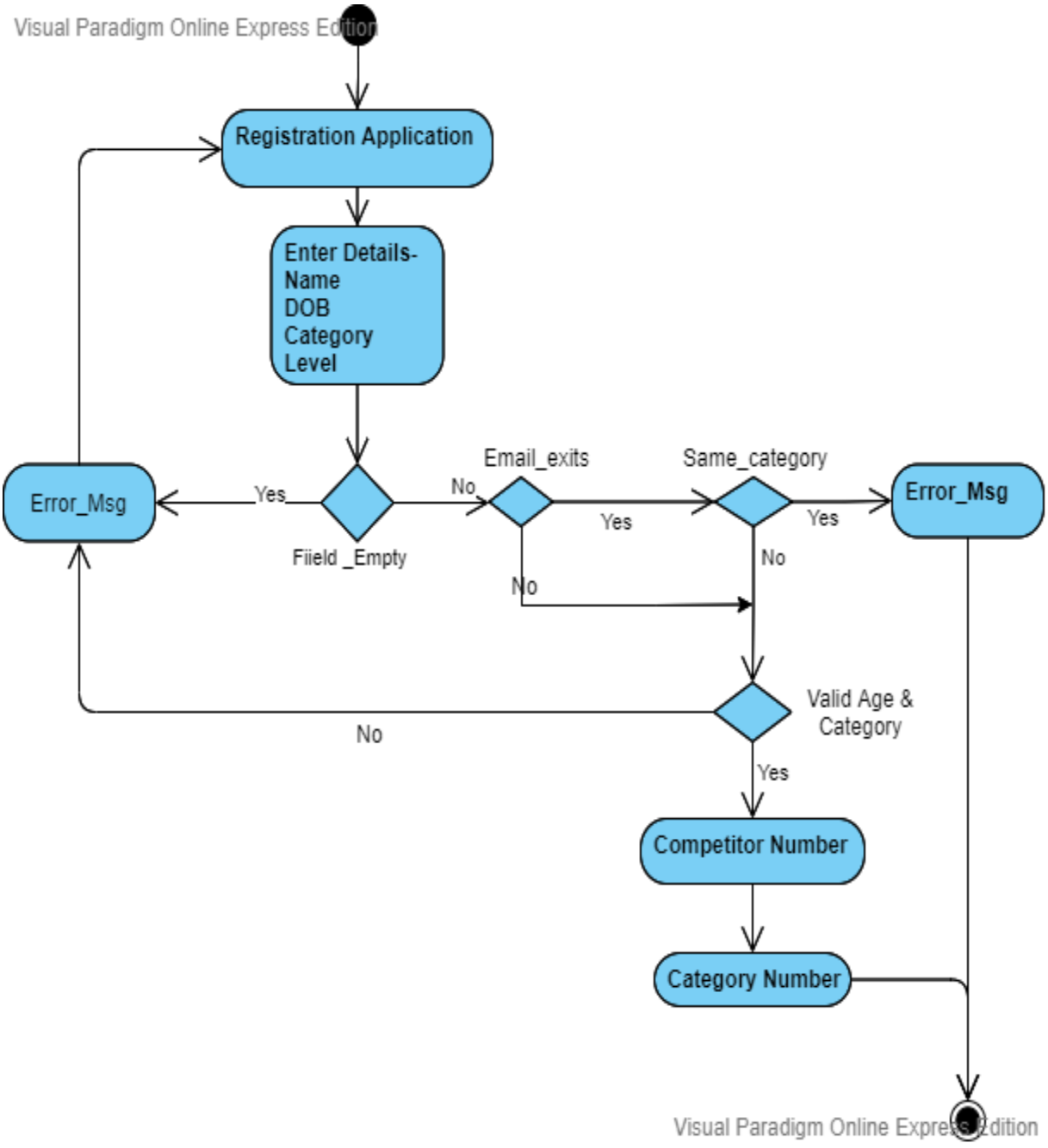

Figure: Activity diagram for Registration form

The figure above describes the Activity diagram of the registration process of different categories of the competition. This was implemented in Visual Paradigm Online Express Editor.

The registration process is represented by the initial node where the process of begins and the next activity is to fill the application form where details of the competitor like Name, DOB, Category, Level is entered. The decision node decides if any field is omitted. If true, is throws an error message activity to resubmit the application form and directs the competitor to start the application again. If Yes, it checks for valid email_ID if it exits. It then checks for the same category even if the email already exits. Then it checks for the valid Age and category is valid. If invalid it throws an error message and directs the user to resubmit the application form. If everything is valid and there is no mistakes in the application form. The competitor is given with an competitor number and also the category number (as there are different categories of competition). Next after assigning the competitor number the program terminates.

5. Use Case Diagram

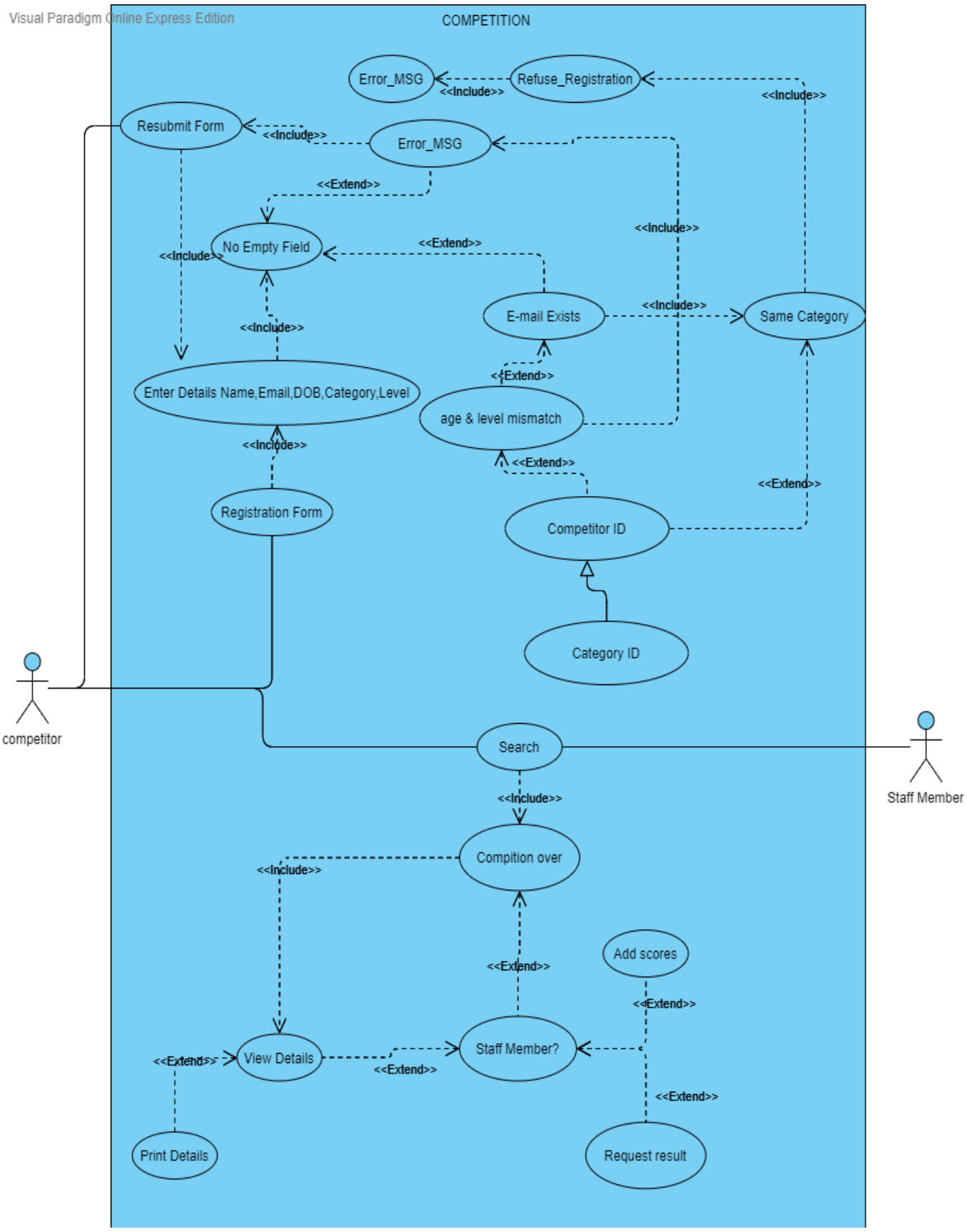
Visual Paradigm Online Express Edition
COMPETITION
Error_MSG
Refuse_Registration
<<Include>>
<<Include>>
Resubmit Form
<<Include>>
Error_MSG
<<Extend>>
<<Include>>
No Empty Field
<<Extend>>
<<Include>>
<<Include>>
E-mail Exists
<<Include>>
Same Category
<<Include>>
<<Extend>>
Enter Details Name,Email,DOB,Category,Level
age & level mismatch
<<Include>>
<<Extend>>
<<Extend>>
Registration Form
Competitor ID
Category ID
competitor
Search
Staff Member
<<Include>>
Compition over
<<Include>>
Add scores
<<Extend>>
<<Extend>>
<<Extend>>
<<Extend>>
View Details
Staff Member?
<<Extend>>
<<Extend>>
Print Details
Request result

Figure: Use-Case diagram of the registration form

The figure above shows us the Use case diagram of the registration form. First the competitor interacts with registration form and to resubmit error. Registration form is included to enter all the details of the competitor, further the Omitted field is checked where it is the included in the details of the competitor. An error message pops up if Email exits and is extended from no field found as they both do not exits every time. The user is asked to resubmit the application form, and this resubmit form always interact with the user Next the check for same category is extended to email exits and age and category mismatch. If the email is same but category is different, it gives the competitor with a different competitor ID and category ID else it throws an error message and terminates the application process.

The competitor and staff member must be allowed to use the search tool to search the competitor number after the competition is over. The competition over is included from the search button. Next it checks if the searching person is a staff. If staff another option to add scores and request results Is extended for staff. The view details can be accessed by both competitor and staff member so its connected in common and Print those details is optional so it extended block from both staff member and the competitor.