

CAB201 Class Assignment – Card Games

Part B: GUI Games Class

1. Introduction

This part of the assignment you will begin to implement the **forms** which are part of the front-end (presentation layer) of the project which will be part of the **Gui Games** project folder as well as starting to implement the game playing logic for the first of two card games, Crazy Eights.

2. Gui Games Project folder

This project folder currently contains 1 empty form, **Start_Game_Form.cs**. This is the initial form which will be displayed when the application begins. See the document **GUI Notes on Start_Game_Form** which contains screen shots of the form's controls.

The other forms in the project folder will be the **CrazyEightsForm** and the **SolitaireForm**.

3. Start_Game_Form

Once you have created this form to resemble that shown in GUI Notes. Implement the event handler for the ComboBox followed by the event handler for the **Start Button**. If the user selects **Crazy Eights**, the **Crazy Eights** form is displayed; initially the **Crazy Eights** can be an empty form. Likewise the **Solitaire** form initially.

The statement to make the **CrazyEightsForm** to appear is

```
new CrazyEightsForm().Show();
```

4. CrazyEightsForm

The **GUI Notes on CrazyEightsForm** describes the controls that are to appear on this form and the settings of some properties.

Once you have completed designing this form you need to read the **Crazy Eights Rules** document before attempting to implement any event handling for this form. It is recommended that you begin to implement the game logic incrementally in the **Crazy_Eight_Game** class and then add some functionality to the form and test that it works.

5. Game_Class_Library

In the **Game_Class_Library** project there is an empty class, **Crazy_Eight_Game**. This class will contain the game logic for playing **Crazy Eights**. (you can rename the class to **Crazy_Eights_Game** if you want to correct my missed 's' in the class name)

You must first add a **Reference** in the **Game_Class_Library** to the **Shared Game Class Library**. Refer to Activity 3 in Worksheet 9 which showed you how to connect separate classes together by adding a reference to a class in another class.

You can test if you have added the Reference correctly by simply declaring three test static variables in the **Crazy_Eight_Game** class as follows:

```
private static CardPile pile;  
private static Hand myHand;  
static Card newCard;
```

and then compiling successfully. If you have errors relating to the classes of **Shared Game Class Library** then you probably need to build those classes again, remove the reference to the **Shared Game Class Library** and obtain a fresh reference to it.

If you still have problems, talk to a tutor in a workshop with your actual code available. It is not worth implementing any more functionality until you have the Reference correct.

You will also need to add a **Reference** in the **Gui Games** to the **Game_Class_Library** and the **Shared Game Class Library** so that the **CrazyEights GUI** can access the public methods in **Crazy_Eight_Game** and those in the various classes of **Shared Game Class Library** as well as declaring appropriate type variables when required.

It is important to keep the distinction between the variables within this class which will enable you to play the game and the external view of the game which is provided by the GUI.

Some students confuse the two and think by simply updating a variable, eg the player's hand in Crazy_Eight_Game that the corresponding image on the GUI is updated without writing and code to do it.

6. General Game Play

A common pattern for playing various types of games such as card games and dice games is Setting Up the Game, Playing the Game Once with players taking their turns respectively until someone has won or the game is tied.

In a card game this involves setting up the deck of cards, shuffling the deck, dealing each player their hand and the players take in in turns to play a card according to the rules of the game.

Though all care has been taken in the production of this specification, there may be a need to notify by class email any alterations/clarifications to this specification. **Check your QUT email daily, even if you have redirecting this to your own email check QUT directly in case your email provider blocks the QUT group email.**

Part C will be released during week 11 along with detail information on displaying the card images and any other information deemed useful.