**MVC** in our program.

The **M**odel **V**iew **C**ontroller architectural pattern has close correlations to the ‘Observer’ design pattern we use in our program. MVC is seen as an implementation of *‘Separations of concerns’* which partitions content from the display and data processing (The ‘model’ in other terms) from the content. MVC uses the Observer design pattern to decouple two of these, the model and the display. In the ‘Observer’ context, the display references the Observer object and the model represents the object which is ‘Observable’. In terms of our project, the ‘Difficulty’ class is seen as data processing model while the ‘Ui class’ is an instance of the display. The ‘Ui’ class and all its subclasses are updated visually when the difficulty class is changed (difficulty updated to easy, medium orhard). This separation of system components means our program is compliant with the rules specified by the model view controller architectural pattern.