

**Cairo University**

**Faculty of computer science and information**

**CS352 – Software Engineering II**

**Phase 1**

**2017**

**A3**

**Team Members:**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| **20140192** | **Amr Saeed Hosny** | **amrshosny@gmail.com** | **01125376468** |
| **20140196** | **Amr Magdy Ibrahim** | **a.magdy7077@gmail.com** | **01141148336** |
| **20140064** | **Ashrakat Mokhtar Hossni** | **ashrakatelnow@gmail.com** | **01114178143** |

**TAs:**

**Eng Ragia Mohamed r.mohamed@fci-cu.edu.eg**

**Eng Amr Tarek a.tarek@fci.cu.edu.eg**

# Review Check List

**Design and Code Checklist**

**General Review Issues**

**1. Variables, objects, and functions’ names should begin with small letters and classes’ names with capital letters.**

**2. Indentation style should be the same for all classes and functions**

**3. Accessing classes attributes directly without using setters and getters violating privacy rules.**

**4. Some functions could be divided into many smaller functions. E.g. AddGame, EditGame methods in GameController class.**

**5. MVC must be applied.**

**6. Commented code is not allowed.**

**7. Unproper line code position is confusing.**

**8. SOLID principles must be applied.**

**9. Variables’ names must be representative for their job.**

**10. Non-sense repeated lines are not allowed. It could be put in separated functions.**

**11. Non-sense lot of variables’ initialization in many functions.**

**12. Throwing file exceptions without using files.**

**13. Lack in error-handling.**

**Detailed Review Issues**

**Account Class:**

**Line 2 – 5 : Unused imports.**

**Line 8 : 13 variable naming capitalization & public attributes.**

**Line 37 : parameter name too short.**

**Line 56 : 69 commented code.**

**AccountController Class:**

**Line 7 – 9 : Public attributes.**

**Line 10, 28, 49, 54, and 99 : Function naming capitalization.**

**Line 32, and 34 : Taking input in non-boundary class.**

**Line 99 – 100 : saveInfo does nothing related to its functionality.**

**Student & Teacher Class:**

**Line 2 – 4 : Unused imports.**

**Only stores comments made by them nothing else.**

**GameController Class:**

**AddGame Function:**

**Line 18 : Variable name Categoryy should be changed as it can cause conflict.**

**Line 22 – 23 : No need for two different scanners beside that they should be closed at the end of the function.**

**Line 47 – 48 : Two assignments to the same variable with the same value.**

**Line 58 – 61 : Using mychoice to question can cause a conflict.**

**Line 64 – 76 : Choices aren't saved for the game.**

**SaveGame Function:**

**Line 85 : Function name is contradicted with the return type of the function.**

**Line 87 – 88 : No need for the two variables game and Catee as they already sent as parameters.**

**PlayGame Function:**

**Line 111 : Throwing FileNotFoundException without using files.**

**Line 116 : The user may be already logged in.**

**Line 154 – 158 : Expressions code be minimized.**

**Line 160 : Comment should be dragged to illustrate what is game type 1 & 2.**

**Line 160 – 194 : Repeated code that can be optimized in both if statements.**

**Line 175 : Else if should be used instead of another if.**

**Line 178 – 180 : There are no choices to show up.**

**Line 199 : The system out statement should show to the user what to enter**

**EditGame Function:**

**Issues about this function are in the general section above.**

**RateGame Function:**

**Line 298 : Error handling.**

**SaveScore Function:**

**Function doesn't add the score to account.**

**RemoveGame Function:**

**Line 313 : categ object isn't assigned to a specific category.**

**test class:**

**Line 14 – 22 : It could be separated function.**

**Line 22 – 23 : It doesn't make sense to make the user to register and login every single time.**

**Each if state could be separated function to remove redundancy.**

**Game Class:**

**TeacherName variable could not be declared so GameCat, they’re deduced from relations.**

**Variable HighScore could be deduced from scores Array no need to be declared. UNUSED variable.**

**Category Class:**

**Public attributes.**

**SaveGame function contradicts with SaveGame Function in GameController class as both classes do the same job.**

**myQuestion Class:**

**Issues about this function are in the general section above.**

**Edit In coding:**

**GameController Class:**

**PlayGame and AddGame Function:**

**1 - Moving the Scanner “in” in the 2 functions to be attribute to in the class itself, As there were 2 scanner in function AddGame “in , inInt”, so when we redirected the input form file, the second scanner “inInt” wrote from the beginning of the file so it was confusing to the compiler and we couldn’t make both of them have the same pointer.**

**2 - Line 37 , 62 , 59 , 66 , 70 in AddGame and line 163 , 180 , 199 in PlayGame to read values “in.Next() - in.NextInt()” was used, so there were a problem in reading from file so we changed them to “in.NextLine()”.**

**3 – Line 116 in PlayGame: we commented this line as it’s a call for “log in” function and it has already been tested.**