Cairo University Faculty of Computers and Information



**CS352 – Software Engineering II**

**Phase 1 Template**

**2017**

**Project Team**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
|  | 1st name is team leader |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Staff:**

**Dr Amr Kamel** [a.kamel@fci-cu.edu.eg](mailto:a.kamel@fci-cu.edu.eg)

**Dr Khadiga Mohamed kelbedweihy@fci-cu.edu.eg**

**[Write here your TA name only in your lab]**

**TAs: Eng Mohamed Samir m.samir@fci-cu.edu.egEng Omar Khaled Ali Ragab o.khaled@fci-cu.edu.egEng Ragia Mohamed r.mohamed@fci-cu.edu.eg**

**Eng Ebtehal yahia ebtehal.yahia@fci-cu.edu.eg**

**Eng Ahmed Emad ahmed.emad@fci-cu.edu.eg**

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

Title page has no header or footer.

Docs must be supplied as pdf –files re-named as requested

Contents

[Instructions [To be removed] 3](#_Toc476413280)

[Review Check List 3](#_Toc476413281)

[Testing 5](#_Toc476413282)

[Git repository link 6](#_Toc476413283)

# Instructions [To be removed]

* **IMPORTANT. Rename this document to CS352-LeaderID-TANAME-Phase2-a.docx**

**[Write TA name in your document name]**

**Examples:**

**CS352-20120001-MohamedSamir-Phase1.docx**

* **Remove the following notes and any red notes**

# Review Check List

* **Use the provided checklist as a starting point for your review. Add more items as issues arise during the review.**

**Design and Code Checklist**

**Design Principles**

1. Does the design follow SOLID principles? ∏ What % …… Related Issues: …….…
2. Does the design follow OOP rules? ∏ What % …… Related Issues: …….…
3. Is the design simple and easy to modify? ∏ What % …… Related Issues: …….…

**Coding Standards**

1. Is the code understandable and readable? ∏0 What % 90 Related Issues: …….…
2. Does the code follow Java Coding Style? ∏ What % …… Related Issues: …….…
3. Is indentation used properly? ∏ What % …… Related Issues: …….…
4. Do variable have good names? ∏ What % …… Related Issues: …….…

**Comments**

1. Is the code commented enough? ∏ What % 50 Related Issues:no enough comments
2. Is every class and method commented? ∏ What % 70 Related Issues: …….…
3. Do comments follow Javadoc style? ∏ What % 0 Related Issues: …….…
4. Is Javadoc generated for all the code? ∏ What % 0 Related Issues: …….…
5. Are there useless / wrong comments? ∏ What % 0 Related Issues: no useless or wrong comments

**Code Structure**

1. Does the code follow the design precisely? ∏ What % …… Related Issues: …….…
2. Are there very long classes or methods? ∏ What % 0 Related Issues: …….…
3. Is there repeated code ?(put put in a function) ∏ What % 0 Related Issues: …….…

**Error Handling**

1. Does the code handle errors and exceptions? ∏ What % 100 Related Issues: …….…
2. Is defensive programming used to avoid errors? ∏ What % 100 Related Issues: …….…

**Logic**

1. Do loops have correct conditions and bounds? ∏ What % 100 Related Issues:
2. Do loops always terminate? Yes

**Overall**

1. **Are the design and code of good quality?** ∏ What % ………

# Testing

* **In the following table, you should describe each testing functions you developed in each testing class and state the result of testing after executing testing class**
* **Number test cases in proper way.**

**Example**

* 1. **UserTesting class**

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Testing function** | **Description** | **Result** |
| **1.** | **getUser(name, pass)** | **Testing function for getUser function in user entity. This test case test the normal login scenario**  **Assumption: name is unique for each user** | **Passed** |
| **2.** | **signup(name, pass)** | **Testing function for sign up function. This test case test the normal login scenario** | **Failed** |

# Git repository link

**You should put here your git repository link**