Cairo University Faculty of Computers and Information



**CS352 – Software Engineering II**

**Software design specification document**

**2018**

**Project Team**

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

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

Title page has no header or footer.

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

Contents

[Instructions [To be removed] 2](#_Toc507236834)

[Class diagram design 2](#_Toc507236835)

[Sequence diagram design 3](#_Toc507236836)

[Github repository link 5](#_Toc507236837)

# Instructions [To be removed]

* **IMPORTANT. Rename this document to CS352-SE2016-LeaderID-TANAME-SprintNumber-SDS Document.docx**

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

**Examples:**

**CS352-20120001-MohamedSamir-Sprint1-SDS Document.docx**

**CS352-20120001-OmarKhaled- Sprint1-SDS Document.docx**

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

# Class diagram design

* **You should provide clean version for your class diagram design.**
* **Class diagram is a static diagram and should not represent any dynamic flow of events.**
* **Put Relationships between classes and the types of the relationships.**
* **Put multiplicity.**
* **Put relationship name (e.g. faculty "offer" course).**
* **Put attributes in the classes.**
* **Put functions & Put parameters.**
* **Put data types of each attributes and the parameters.**
* **Highly perfered: Each class has a corresponding interface**
  + **Let all objects parameters and returns be of interface type.**
* **See Shopping Cart Case Study**



# Sequence diagram design

* **List Sequence diagrams for the most important user story (according to your opinion).**
* **Make sure that each object in the sequence diagram has a corresponding class in the class description table above. If not, it will be REJECTED.**
* **Put actual function calls with proper parameters and return types corresponding to class diagrams.**
* **Following are couple of examples for small / meduim examples. We expect such diagrams, however there is a missing thing in them. Most of calls don’t have parameters. Please always specify the parameters in the call, matching the class diagram.**



* 

# Github repository link

* **URL for your repository, should contains implemented services**,
* ***Your implementation should follow coding style (java style) you should also provide documentation for each class in your project***