Cairo University Faculty of Computers and Artificial Intelligence



**Software design specification document**

**2022**

**Project Team**

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Email** |
| 20200359 | Omar Mohamed Mostafa | [omar.m.elesawy2002@gmail.com](mailto:omar.m.elesawy2002@gmail.com) |
| 20200514 | Marwa Ahmed Mohamed | [marwamubarak6@gmail.com‏](mailto:marwamubarak6@gmail.com‏) |
| 20200268 | Doha Abd-ElBasset Ahmed | [dohae7222@gmail.com](mailto:dohae7222@gmail.com) |

Contents

[Instructions[To be removed] 2](#_Toc120811426)

[Class diagram design 2](#_Toc120811427)

[Class diagram Explanation 3](#_Toc120811428)

[Sequence diagram design 3](#_Toc120811429)

[Github repository link 4](#_Toc120811430)

# Instructions[To be removed]

* **IMPORTANT. Rename this document to** Phase2\_LabGroupNumber\_Phase1\_StudentID1\_StudentID2\_StudentID3\_StudentID4**\_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**



# Class diagram Explanation

* **Explain here the design pattern(s) that you used and your justification for using them, and the participating classes for each pattern.**

# 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.**



# Requirements Exposure as Web Service API

**Part 1: Exposed Postman Collection**

**Part 2:**

**Explain here the exact mapping between every single requirement and its corresponding web service API operation. A sample example is provided to better explain the concept.**

|  |  |  |
| --- | --- | --- |
| Requirement |  | Exposed API |
| The system should check if the username or the email is registered before. |  | 1- GET /user/check  A service to check if the user exists or not. This service returns all user info if exists  Input: email and password. |
|  |  |  |
|  |  |  |
|  |  |  |

# Github repository link

* <https://github.com/MarwaMubarak/software-project>