Cairo University Faculty of Computers and Artificial Intelligence



**Software design specification document**

**2022**

**Project Team**

|  |  |  |
| --- | --- | --- |
| **ID** | **Name** | **Email** |
| 20206130 | Omar Waleed Zenhom | omarwaleedzenhom2002@gmail.com |
| 20216019 | Omar Wael |  |
| 20206147 | Ayman Hatem |  |
| 20206091 | Youssef Ashraf |  |

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)

# Class diagram design

**Chart, diagram, box and whisker chart

Description automatically generated**

# Class diagram Explanation

* **DE composite**
* **Strategy**
* **MVC**

# 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.**

**Land Service request**

**A picture containing diagram

Description automatically generated**

**Pay on delivery**

**Diagram

Description automatically generated**

**AcceptRefund**

**Diagram

Description automatically generated**

**PaywithDiscount**

**Diagram

Description automatically generated**

**Login\_Error**

**Chart, diagram, box and whisker chart

Description automatically generated**

**translation\_erorr**

**Diagram

Description automatically generated**

**mobile\_recherge**

**A picture containing diagram

Description automatically generated**

**Donation**

**Diagram

Description automatically generated with medium confidence**

# Requirements Exposure as Web Service API

**Part 1: Exposed Postman Collection**

Table

Service Api :

Get DisplayService()

Post SearchService(key)

Post SelectSP(service provider id , servicename)

Get DisplayDiscount()

Post SelectService(id)

Post SelectDiscount(id)

Post DisplayCost(amount , service name , service provider id , choice , disID)

Payment Api:

Post DisplayPayment (cost , payed , choice)

Admin Api:

Post AddDiscount ( rate , name , type , ID , dison)

Post DisplayUserTransaction(username)

User Api:

Post DisplaySignUp (Email, username , password , address)

Post DisplayLogin (email , password)

# Github repository link

* **URL for your repository, should contains implemented services**,

<https://github.com/omarwael36/SDA-phase-2>