

**Course code**

**Course Name**

Phase 2: Software Design Document Team Name

Full Name : ID:

Yara Zakaria Mohamed 203162

Mayar El-Said Abd El-Phtah 203149

# Dec 2020

Month & Year

**Contents**

[Instructions [To be removed] 3](#_bookmark0)

[Team 3](#_bookmark1)

[Document Purpose and Audience 3](#_bookmark2)

[System Models 4](#_bookmark3)

1. System Decomposition Error! Bookmark not defined.
2. [Class diagrams 4](#_bookmark4)
3. [4](#_bookmark4)[Sequence diagrams 5](#_bookmark5)

[Class - Sequence Usage Table 7](#_bookmark6)

1. [Physical Entity-Relationship Diagram 8](#_bookmark7)
2. [User Interface Design 8](#_bookmark8)
3. [Algorithms and Data Structures 10](#_bookmark9)

[Ownership Report 10](#_bookmark10)

[Policy Regarding Plagiarism 10](#_bookmark11)

References **Error! Bookmark not defined.**

Authors **Error! Bookmark not defined.**

## Instructions [To be removed]

* + **IMPORTANT. Rename this document to TeamName-Topic-SDD.docx**
  + **Remove the following notes and any red notes.**
  + **This document is the template document for your SRS.**
  + **For further guidelines and information, READ project details document (project decription).**
  + **Figures included here are for GUIDANCE purpose. Do not copy them or imitate them. Use the notations taught in class or the best suitable notation for each design item**

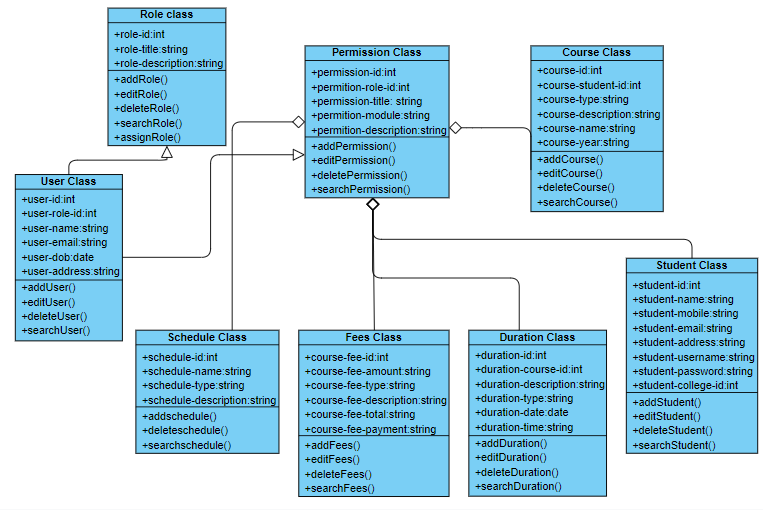
## Team

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

## Document Purpose and Audience

## System Models

### Class diagrams



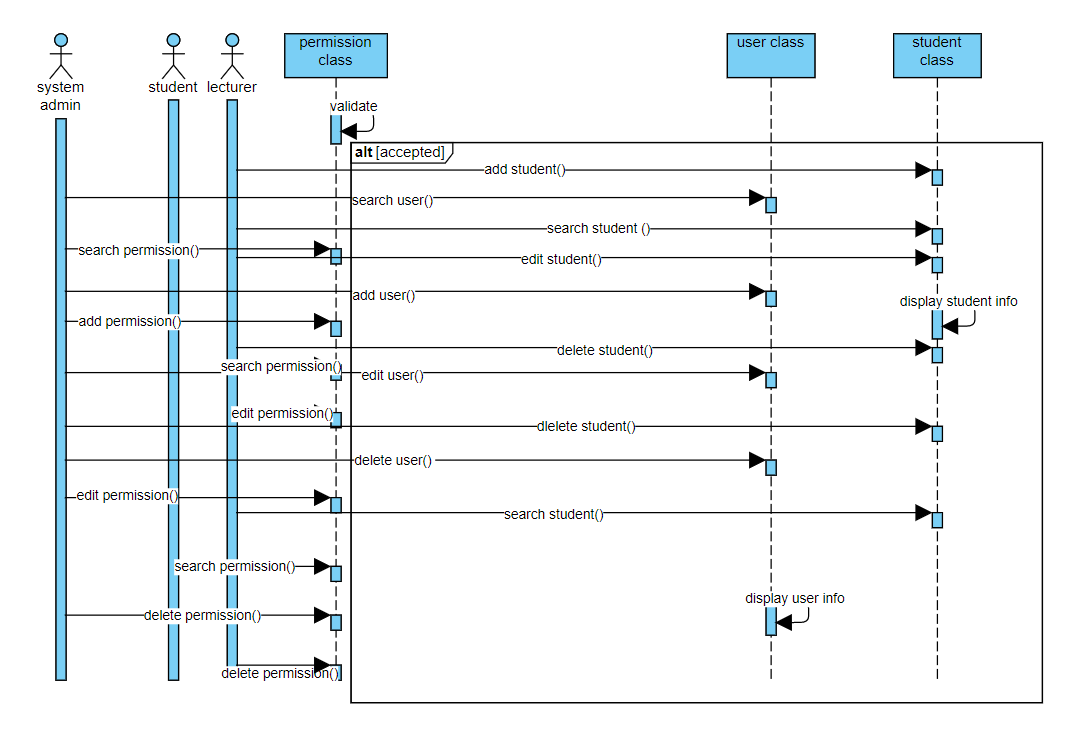
|  |  |  |  |
| --- | --- | --- | --- |
| **Class ID** | **Class Name** | **Subsystem ID** | **Description & Responsibility** |
| 00005 | Role Class | 1 | Each user has some roles.  This class contains all the operations that can be done to roles like: searching for a user role. |
| 00008 | User Class | 1 | To add or delete or search or edit information of user in system. |
| 00011 | Schedule Class | 1 | To allow lecturer to make schedule for his course and edit it. |
| 00077 | Fees Class | 1 | -Lecturer add fees of his course and edit it.  -Allow students to search about fees for each course. |
| 00066 | Duration Class | 1 | To allow lecturer to determine duration for his course and edit it if he wants. |
| 00045 | Student Class | 1 | -To add student in a course  -To allow student to edit his profile information.  -delete student from a course.  -to make search for a student |
| 00111 | Course Class | 1 | To allow each lecturer to add or edit or delete his courses and allow student to search about courses that he wants to take it. |
| 11022 | Permission Class | 1 | -To add or delete or edit permissions of this system.  -Allow users to search about this permissions. |

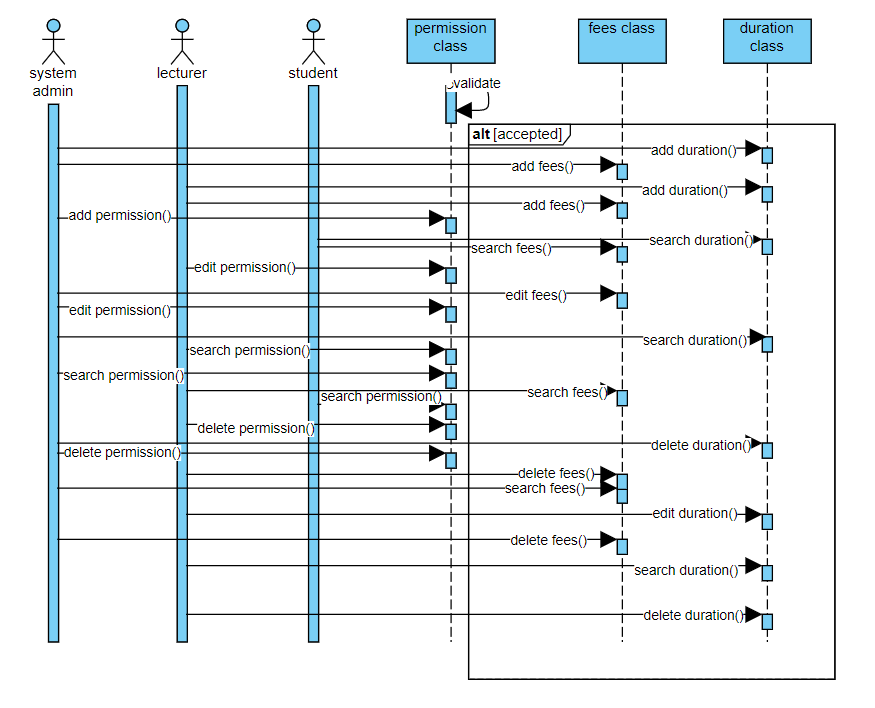
### 

### Sequence diagrams





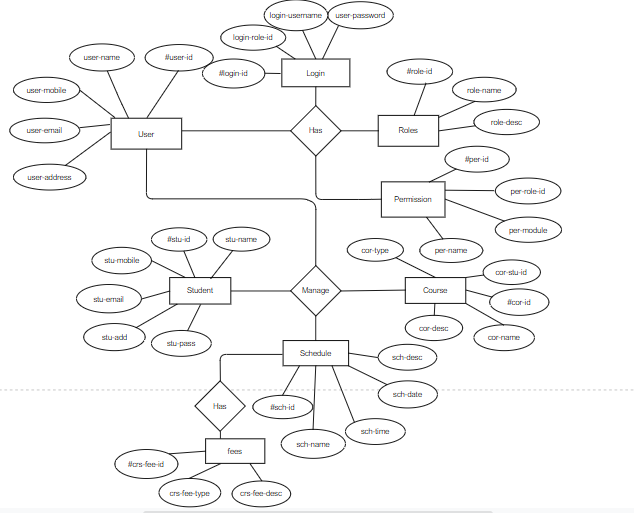




Class - Sequence Usage Table

|  |  |  |
| --- | --- | --- |
| **Class name** | **Sequence diagram** | **Overall used methods** |
| User class | 3 | Add user ()  Edit user ()  Search user ()  Delete user () |
| Student class | 3 | Add student ()  Edit student ()  Search student ()  Delete student () |
| Role class | 1 | Add role ()  Edit role ()  Search role ()  Delete role () |
| Permission class | 1,2,3,4 | Add permission ()  Edit permission ()  Search permission ()  Delete permission () |
| Course class | 1 | Add course ()  Edit course ()  Search course ()  Delete course () |
| Schedule class | 2 | Add schedule ()  Edit schedule ()  Search schedule ()  Delete schedule () |
| Fees class | 4 | Add fees ()  Edit fees ()  Search fees ()  Delete fees () |
| Duration class | 4 | Add duration ()  Edit duration ()  Search duration ()  Delete duration () |

### Physical Entity-Relationship Diagram

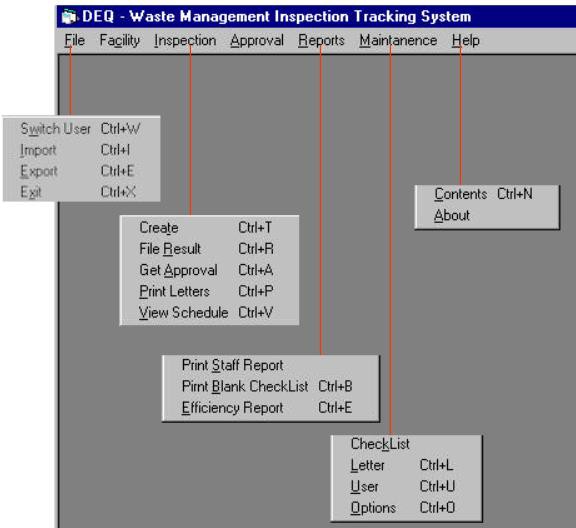


### User Interface Design

* + **Use a prototyping tool like** [**https://app.moqups.com**,](https://app.moqups.com/) [**http://Infragistics.com**](http://infragistics.com/)  **or NinjaMock or using a GUI builder (like the one in NetBeans) to build your interface.**
  + **Develop a prototype for each screen / page that your application will have and relate them to each other showing which one leads to which one.**
  + **For each screen specify the buttons, menus, etc. that will be on it and their functions.**
  + **An example is shown below.**
  + **Screen 1 – Login Screen (example)**



* + **Screen 2 – Main Interface (example)**



* + **Navigation tree:**

Login Screen

|

Main Screen

### Dataflow diagram (DFD)

* + **Provide the DFD**

### Algorithms and Data Structures

* + **Specify what algorithms you need in order to build the application. If it is an existing one, just refer to it. If it is one you will develop, then write in detail in mathematical notation, pseudo code, or as a flowchart. Example of such algorithms:**
    - **The steps for calculating if there is winner in a two-player game.**
    - **The steps for calculating the salary in a payroll program.**
    - **The algorithm for deciding which posts to show first in a social network application.**
  + **Specify which data structures (DS) you will use to store which data in memory, other than regular arrays and array lists. Justify your choice and explain the reasons behind it.**
  + **In the rare occasion that no existing data structures supports your requirements and you need to create a new one or implement a non-implemented one, include the design of this new DS.**

## Ownership Report

* + **Remove the following notes and any red notes**
  + **For every item in this document, write the owners. If someone is owner of something, s/he understands it 100.%**
  + **Team leader must verify the table with the team members.**

|  |  |
| --- | --- |
| **Item** | **Owners** |
|  |  |
|  |  |

Policy Regarding Plagiarism**:**

**Students have collective ownership and responsibility of their project. Any violation of academic honesty will have severe consequences and punishment for ALL team members.**

1. تشجع الكلية على مناقشة الأفكار و تبادل المعلومات و مناقشات الطلاب حيث يعتبر هذا جوهريا لعملية تعليمية سليمة

2. ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم فى الكود و لكن تبادل الحلول غير مقبول و يعتبر غشا.

3. أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.

4. قد توجد على النت برامج مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.

5. إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.

6. فى حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و فى حالة تكرار الغش سيرسب الطالب فى المرر.