Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

Social Network

Software Design

|  |  |
| --- | --- |
| Adham Mamdouh | 20170039 |
| Ahmed Nasr ElDardery | 20170034 |
| Belal Hamdy | 20170077 |
| Abdelrahman Mohamed Hassan | 20170150 |

December/2019

Contents

[Team 3](#_Toc26633070)

[Document Purpose and Audience 3](#_Toc26633071)

[System Models 4](#_Toc26633072)

[I. System Decomposition 4](#_Toc26633073)

[II. Class diagrams 4](#_Toc26633074)

[Important Algorithm 5](#_Toc26633075)

[III. Sequence diagrams 6](#_Toc26633076)

[Class - Sequence Usage Table 7](#_Toc26633077)

[Ownership Report 8](#_Toc26633078)

[Policy Regarding Plagiarism: 8](#_Toc26633079)

[References 8](#_Toc26633080)

[Authors 8](#_Toc26633081)

# Team

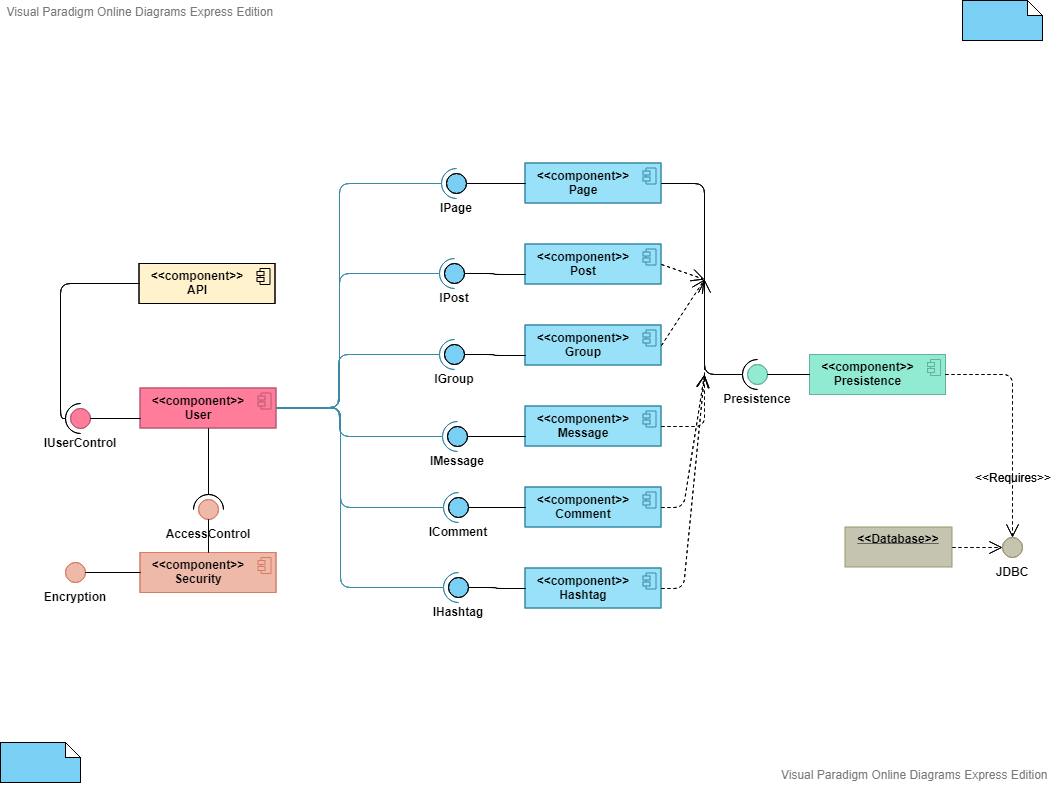
|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20170039 | Adham Mamdouh | Mamdouhadham57@gmail.com | 01111404335 |
| 20170034 | Ahmed Nasr El-dardery | Write your Email | Write your number |
| 20170077 | Belal Hamdy Ezzat | belalhezzat@gmail.com | 01120171784 |
| 20170150 | Abdelrahman Mohamed Hassan | Abdookasha01@gmail.com | 01144788185 |

# Document Purpose and Audience

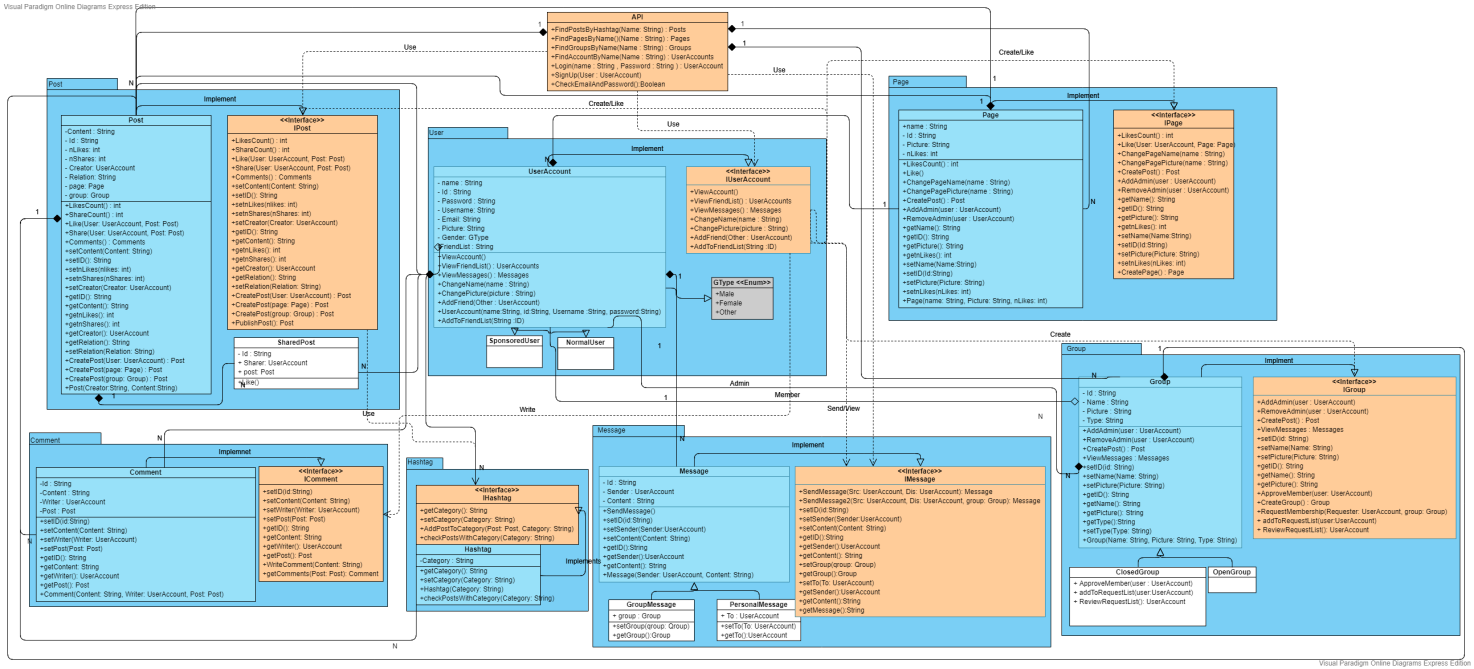
* This document is for illustrating the system design, system composition and sequence diagrams for some use cases.
* Expected audience: System Analyst, Project Manager and lead members.

# System Models

## I. System Decomposition

****

## II. Class diagrams



**List down your classes and describe them**

| **Class ID** | **Class Name** | **Subsystem ID** | **Description & Responsibility** |
| --- | --- | --- | --- |
|  |  |  |  |

* **In the above table make sure that each class belongs to a subsystem.**
* **In the above table ALL classes should belong to subsystems. And each subsystem should at least contain one class.**

### Important Algorithm

* **If any method in a class is implementing an algorithm (complex enough), then you should describe it here.**

## III. Sequence diagrams

* **List Sequence diagrams for all requirements. Provide for each Sequence an ID.**
* **Usually each use case is represented by a sequence diagram or more.**
* **Overall, all the diagrams should represent all requirements and possible flows.**
* **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.**





### Class - Sequence Usage Table

* **In this table, we will list EVERY class in class diagram and which sequences used this class diagram. This helps in avoiding either unused classes or extra classes appears in sequence diagrams. In "Overall used methods" section, put all functions appeared in all sequences. If this table was built in ignorance of actual class / sequence diagrams = REJECTED for whole document.**

| **Class Name** | **Sequence Diagrams** | **Overall used methods** |
| --- | --- | --- |
| User | 1,2 | Create Group, RequestMembership |
| IGroup | 1,2 | CreateGroup, setGroupName, setGroupPicture, setGroupType, RequestMembership |
| Group | 1,2 | CreateGroup, AddToRequestList |
| Post | 3 | CreatePost, setRelation,publishPost |
| IPost | 3 | CreatePost, setRelation,stePostContent, setPostRelation,publishPost |
| IMessage | 4 | setContent,setTo,sendMessage,  getMessage |
| Message | 4 | sendMessage,getSender,getContent |
| IUserAccount | 5 | AddFriend |
| UserAccount | 5 | AddToFriendList |
| API | 6 | CheckEmailAndPassword |
| IHashTag | 7 | checkPostsWithCategory |
| iHashTag | 8 | SetCategory |
| HashTag | 8 | SetCategory,addPostToCategory |

# 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 المقرر.

# References

* http://www.mhhe.com/engcs/compsci/pressman/graphics/Pressman5sepa/common/cs1/design.pdf

# Authors

* Mostafa Saad and Mohammad El-Ramly