

**Cairo University**  
**Faculty of Computers and Information**



# **CS251**

# **Software Engineering I**

**SDS project 3**

**Software Design**

**December 6, 2019**



# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

### Contents

Instructions [To be removed] .....	<b>Error! Bookmark not defined.</b>
Team .....	3
Document Purpose and Audience .....	3
System Models .....	3
I. System Decomposition .....	4
II. Class diagrams .....	5
Important Algorithm.....	<b>Error! Bookmark not defined.</b>
III. Sequence diagrams.....	8
Class - Sequence Usage Table.....	14
Ownership Report .....	16
Policy Regarding Plagiarism:.....	<b>Error! Bookmark not defined.</b>
References .....	<b>Error! Bookmark not defined.</b>
Authors .....	<b>Error! Bookmark not defined.</b>



# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

### Team

ID	Name	Email	Mobile
20170372	Ahmed Wessam Fathy	ahmed.wessam.1999@gmail.com	+201227647347
20170110	Sara Samer Moustafa	ss.titanic199@stud.fci-cu.edu.eg	01026991646
20170115	Salma Essam Soliman	Ssalma.essamm@gmail.com	01008097149

### Document Purpose and Audience

#### What's this Document?

#### What's this Document?

- This document is to present a detailed description to the related readers of this document about "Social Network back-end API". It will explain the purpose and features of the system, its interfaces, all functionalities of the system, constraints and restrictions that's agreed among stakeholders and developers and how the system actors interact with the system, and it will be proposed to the product owner for its approval.

#### Audience list:

- Project owner
- Project manager
- System architect
- Development team



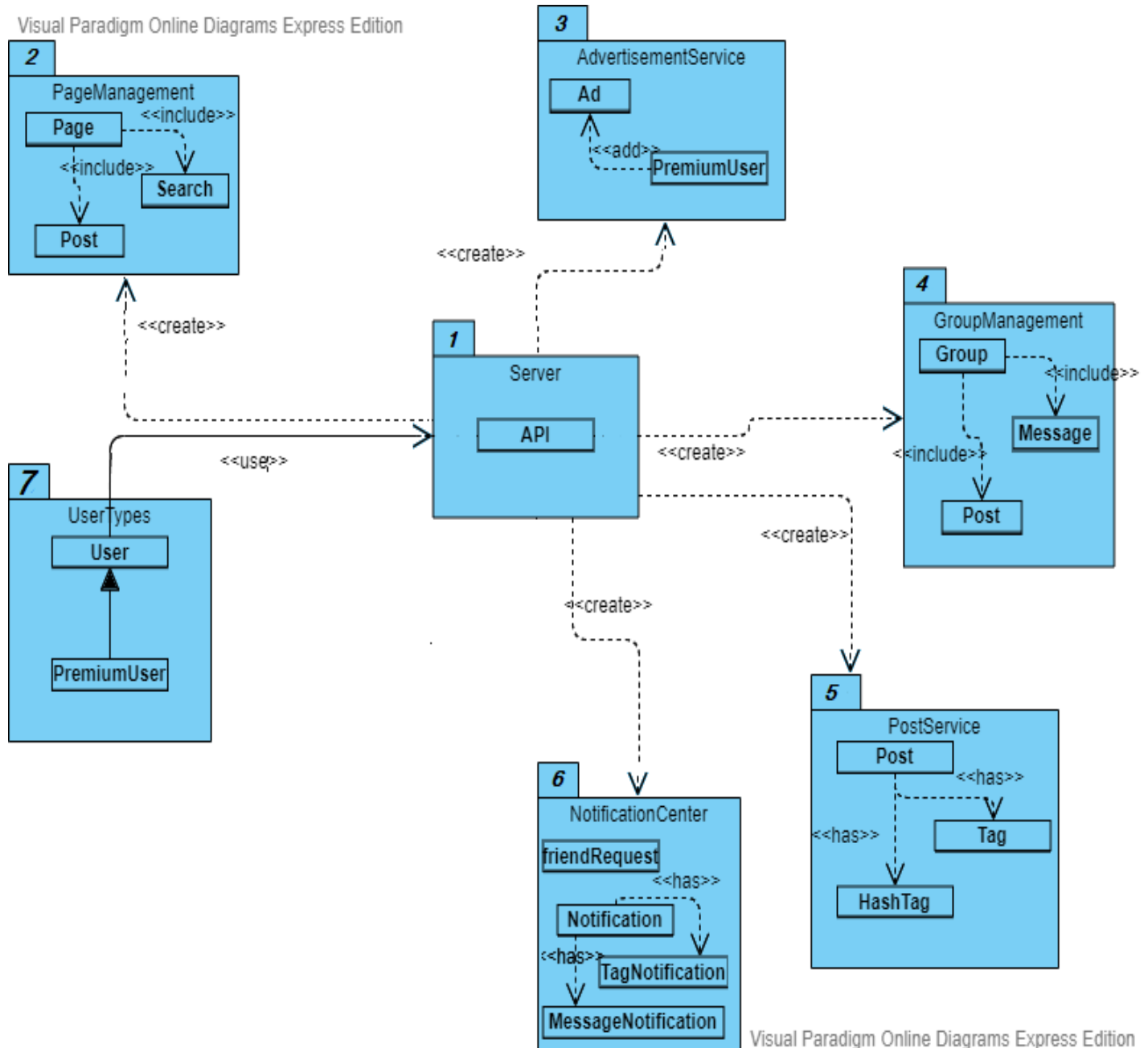
# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

### System Models

#### I. System Decomposition





# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

### II. Class diagrams

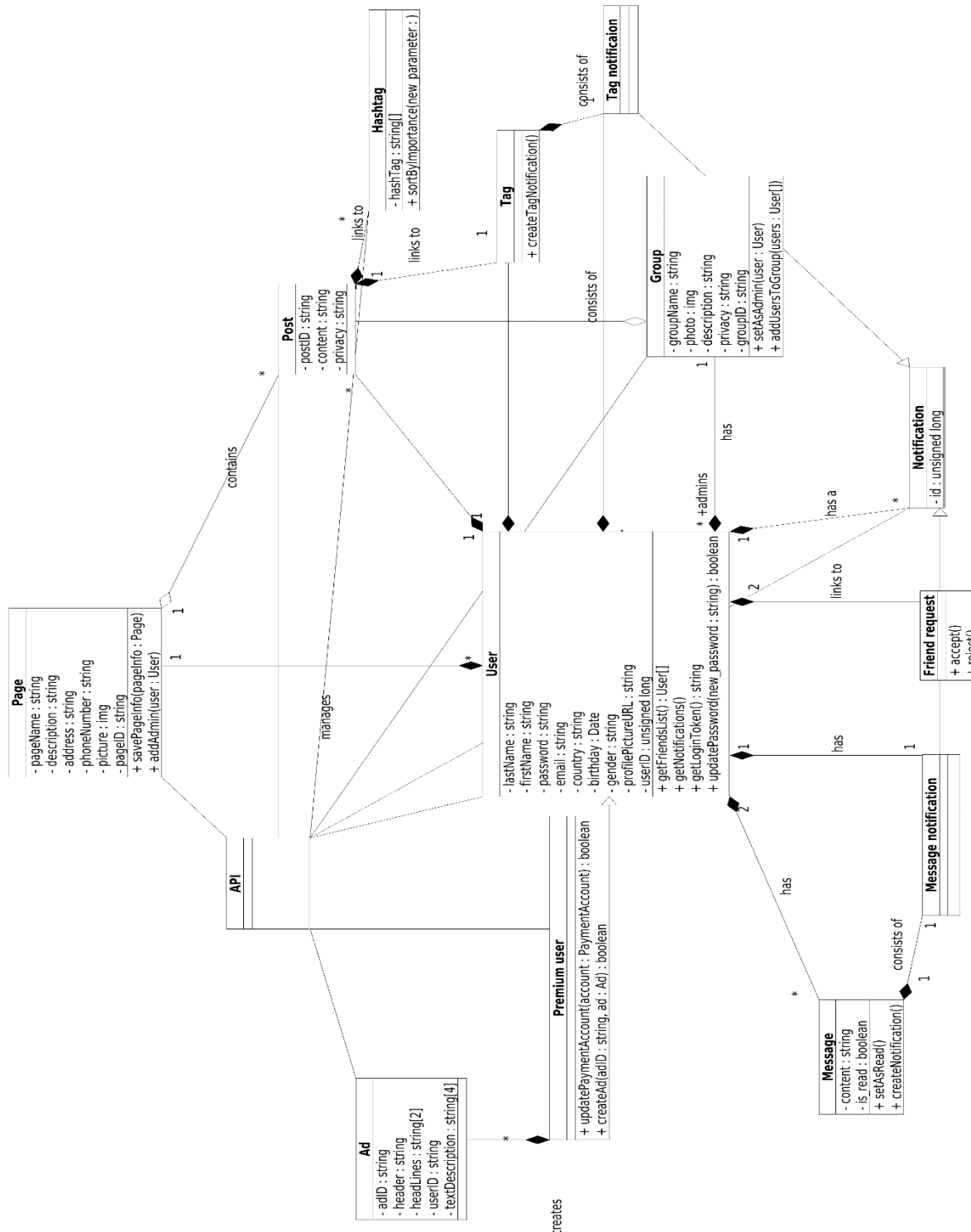
API_1
<pre> # processRequest(header : string, body : string) # signUp(name : string, password : string, email : string, date of birth : Date, is male : boolean, country : string) : User # login(email : string, password : string) : User # getUserByToken(token : string) : User # searchUserByName(name : string) : User[] # searchUserByEmail(email : string) : User[] # sendMessage(sender : User, receiver : User, content : string) : boolean # sendFriendRequest(sender : User, receiver : User) : boolean # getNotifications(user : User) # acceptFriendRequest(request : Friend request) : bool # upgradeUser(user : User, payment : PaymentAccount) : boolean # createAd(user : PremiumUser, ad : Ad, payment : PaymentAccount = null) : boolean # createGroup(group : Group, user : User) : boolean # searchGroupsByName(name : string) # joinGroup(user : User, group : Group) : boolean + createPost(user : User, post : Post) : boolean # pollPosts(user : User) # searchPostsByHashtag(hashtag : HashTag) # validateGroupInfo(groupInfo : Group) : boolean # validatePageInfo(page : Page) # createPage(user : User, page : Page) # getUserById(userId : unsigned long) : User # hasRemainingAds(user : PremiumUser) : boolean # checkPaymentAccount(account : PaymentAccount) : boolean # verifyGroupID(id : unsigned long) # updateUserPassword(user : User, new_password : string) : boolean </pre>



# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification





# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

Class ID	Class Name	Subsystem ID	Description & Responsibility
1	API	1	Acts as server and has the direct interaction with the database User can use it to search for posts, users, groups and pages
2	User	8, 7	Uses API to reach posts and to send messages
2.1	Premium User	8, 3	Acts as a normal user with privilege of posting ads and a high priority hashtags
3	Notification	6	notifies user about an event in his account that may or may not require further action
3.1	Tag Notification	6	Notifies user they were tagged in a post
3.2	Message Notification	6	Notifies the user about a message they received
8	Friend Request	6	The user can send a friend request to another user to view his posts
9	tag	6, 5	A user can post a Post and add another user to it
10	message	6, 4	A chat between one or more users
11	Post	5, 2, 4, 7	Hold the users written thoughts to share them with people in his friend list
12	Hashtag	5	A (#) in a post to categorize it and to make it easy to search
13	Group	4	Groups the posts a number of users that they can view even if they are not in each other's friend list
14	Ad	3	A promotion to a service or a product that a premium user offers
15	Page	7, 2	A fan page for a product or a public figure only the page admins can post on it

- 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.

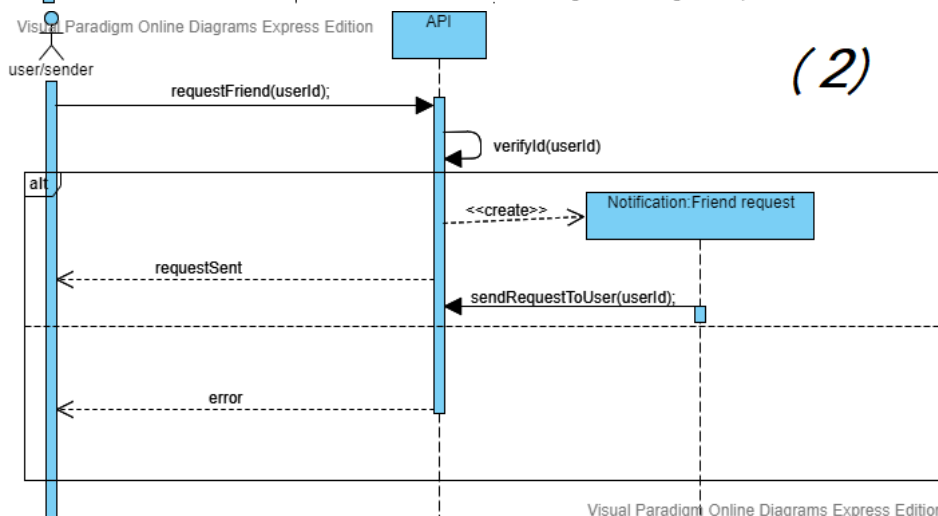
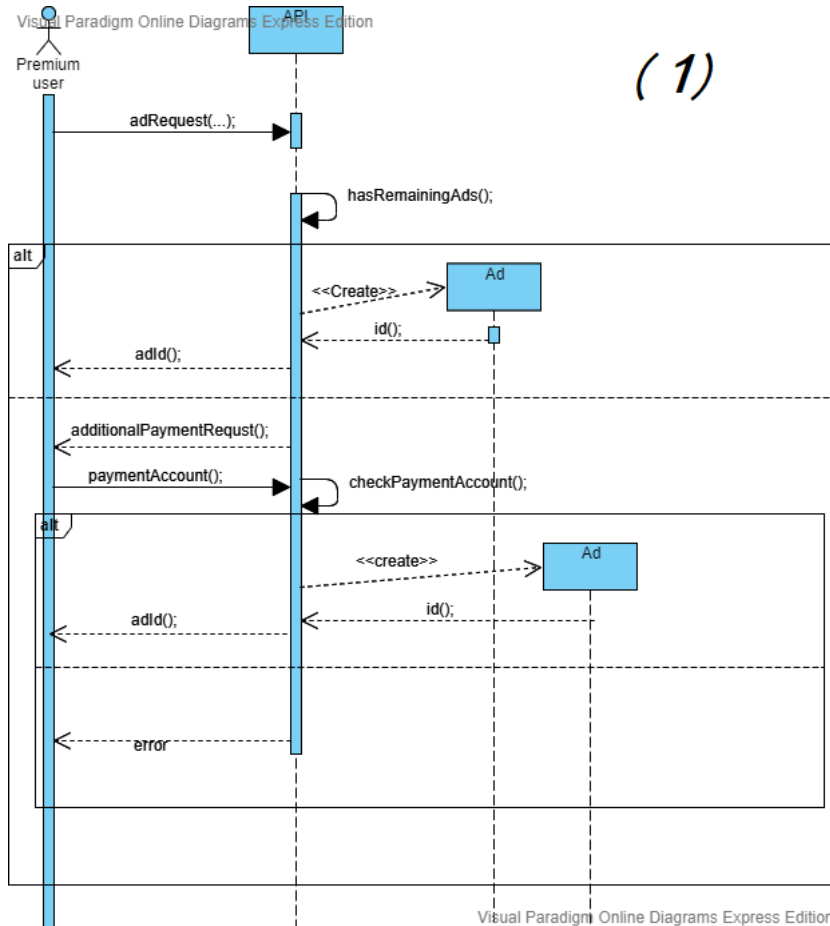


# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

### III. Sequence diagrams



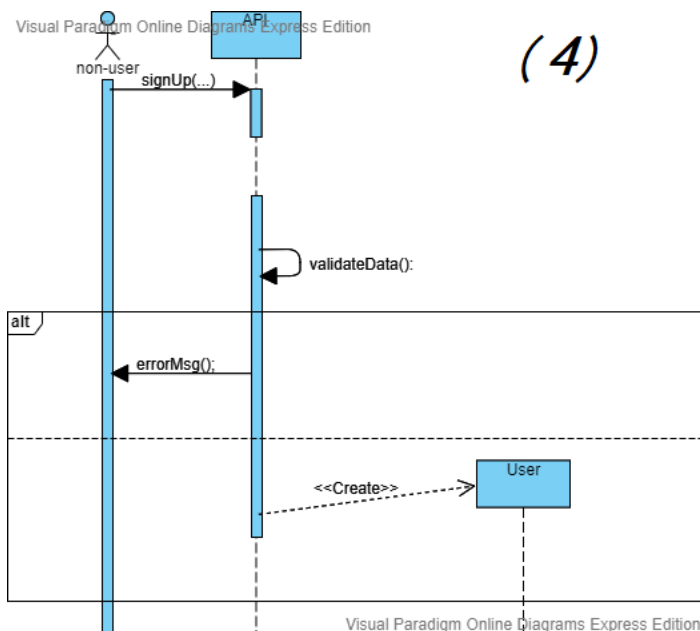
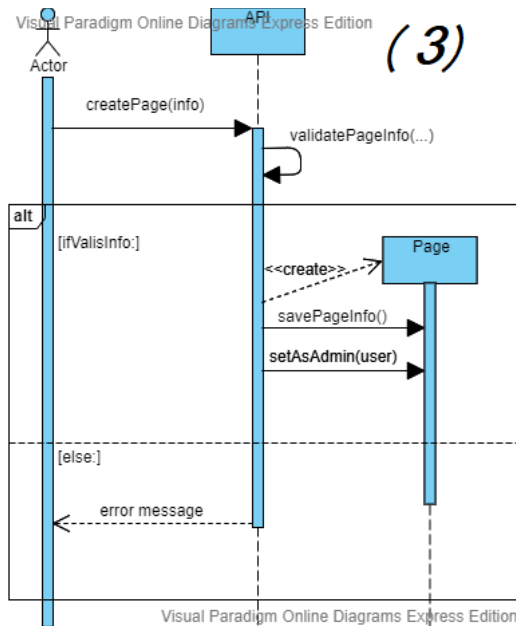




# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

### Software Design Specification

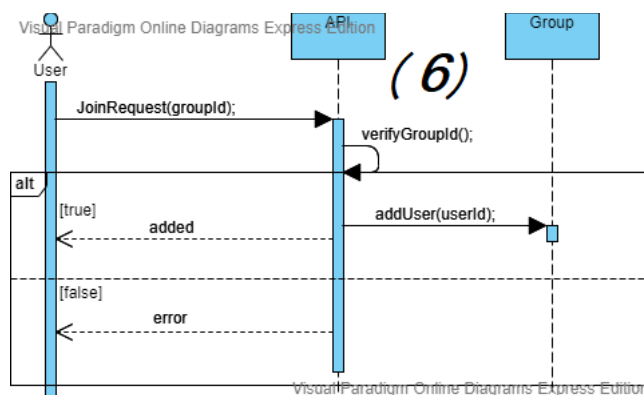
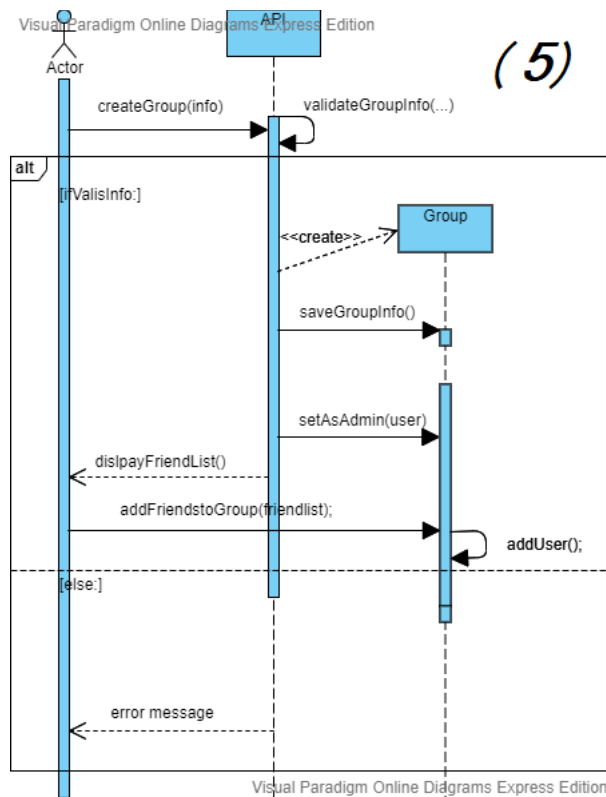




# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

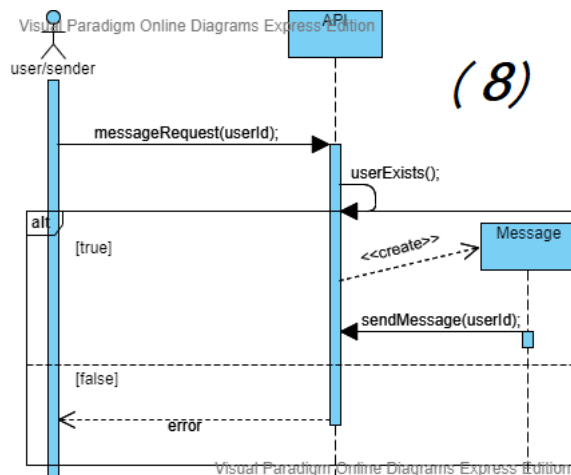
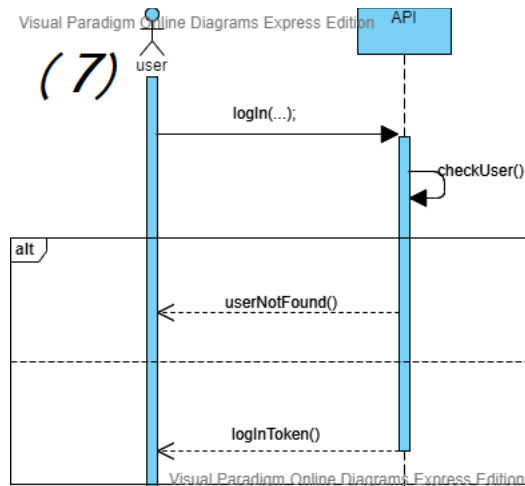




# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

### Software Design Specification

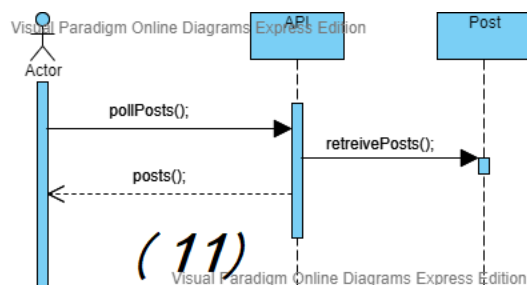
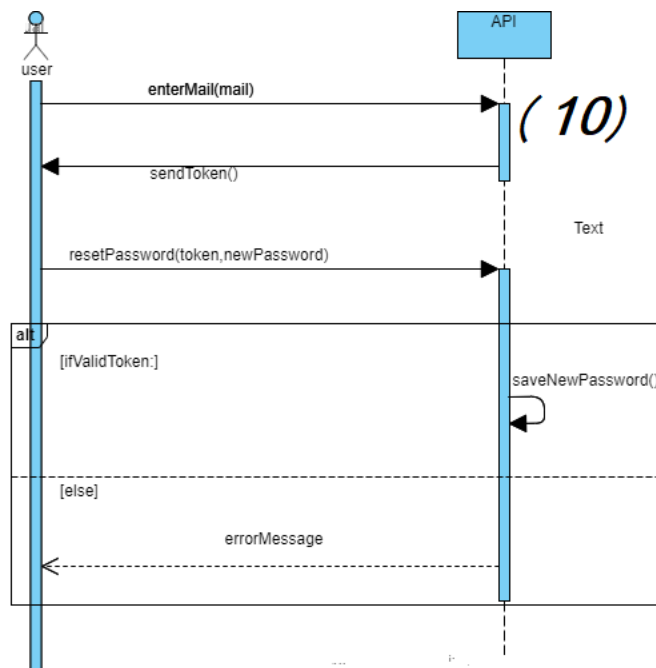
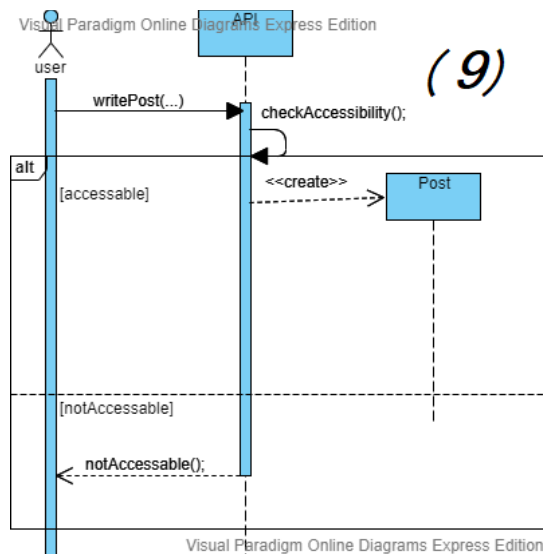




# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

### Software Design Specification

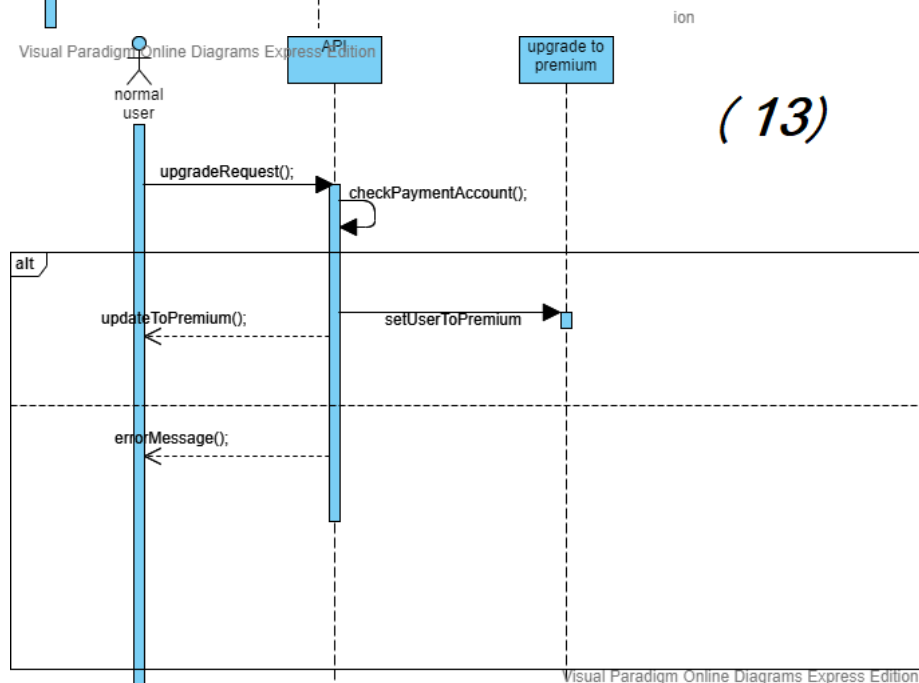
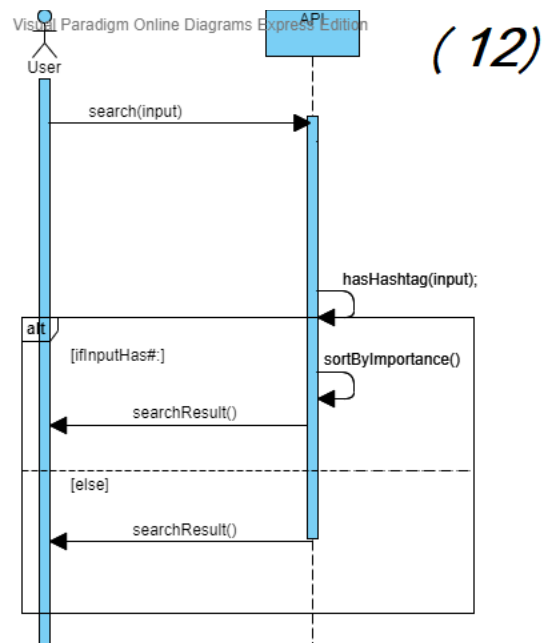




# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

### Software Design Specification





# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

### 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
ad	1	
Friend Request	2	
Page	3	savePageInfo(page info); setAsAdmin(userId);
User	4	
Group	5, 6	saveGroupInfo(page info); setAsAdmin(userId); addUser(userId); addFriendtoGroup(list userId);
Message	8	
Post	9, 11	retrievePosts(userId);
upgradeToPremium	13	setUserToPremium(userId);



# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

Class Name	Sequence Diagrams	Overall used methods
API	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13	adRequest(adInfo); hasRemainingAds(userId); checkPaymentAccount(data); PaymentAccount(data); RequestFriend(userId); verifyId(userId); sendRequestToUser(userId); validatePageInfo(); createPage(pageInfo); signUp(data); validateData(); createGroup(info); validateGroupInfo(); joinRequest(groupId); verifyGroupId(); login(username, password); checkUser(); messageRequest(userId); userExists(userId); sendMessage(userId); writePost(); checkAccessibility(); enterMail(); resetPassword(); savePassword(); pollPosts(userId); search(input); hasHashtag(input); sortByImportance(); upgradeRequest(userId);

BR# ( business/functional requirement number )	Sequence Diagram ID	Package(es) Name(s)
Create Ad	1	Advertisement Service
Friend Request	2	Notification Center
Create Page	3	Page Management
Sign Up	4	User
Create Group	5	Group Management
Join Group Request	6	Group Management , Notification Center
login	7	Server
Send Message	8	User ,Group
Write Post	9	Post Service



# CS251: Phase 2 – The Procrastinators

## Project: SRS-3

## Software Design Specification

<b>Reset Password</b>	<b>10</b>	<b>Server</b>
<b>Retrieve Posts</b>	<b>11</b>	<b>Post Service</b>
<b>Search</b>	<b>12</b>	<b>Server</b>
<b>Upgrade To Premium</b>	<b>13</b>	<b>Server</b>

## Ownership Report

<b>Item</b>	<b>Owners</b>
Component diagram	<i>Salma essam</i>
Sequence diagram	<i>Sara samer</i>
Class diagram	<i>Ahmed wessam</i>