



# University of Palestine

Faculty of Applied Engineering and Urban Planning  
Software Engineering Department



## Students

Alaa Sami Abu Zarifa  
Mohammed Fawzi Abu Hassna  
Rawan Nabeah Mushtaha

## Supervisor

Dr. Abed El-Hamid Zougbor

## Abstract

Storing and sharing files is a task that all internet users do every day, in recent years the way people store their files is changed due the emergence of cloud storages services such as Google Drive, OneDrive and Dropbox, etc. They introduced a great solution for most users when it comes to storing files and sharing it with others. There are a ton of cloud storages options out there. Inevitably, everyone finds a favorite, but more than likely, a lot of users end up with a bunch of unused free storage from other services or the circumstances dictates them to use different cloud drives in case the user prefers Google Drive for example, but his work's infrastructure uses Dropbox and his friends keep sharing files and photos using OneDrive. As a result the user begin to suffer from the same problem that was hoping to avoid from the first place, it became so frustrating to manage, store and share his files because it's scattered in several storages and each one requires its own login. Duo to this, MCloud was born. MCloud is a web-based application essentially as cloud service that can put multiple cloud drives into one. It mainly serves for the users who have multiple cloud drives and in the need for managing all the files in their cloud drives through just one single account. MCloud users can connect their MCloud profile to other cloud storage services such as Google Drive, OneDrive and Dropbox. This connection allows the MCloud user to browse all files of these different drives in the same place without opening too many tabs or keeping longing in and out. Furthermore the team made sure that most functionalities that were available in the original cloud drives are also exists in MCloud system as well for instances the user can share files on social media and copy a sharable link to send it via email or a chat, also the major file management tools like creating, deleting and renaming files and folders are available, in addition to that the user also able to download and upload files easily. The team was dedicated to develop the system in the most fitted development practices with the available skills by using the ASP.NET MVC as the development framework with C# as programming language for our system following the incremental model with all the shear amount of advantages it offers. MCloud system is designed with a easy to use and user-friendly interface with responsive design technologies in order to give the user the ultimate solution for using cloud storages services.

# Acknowledgment

*In the name of Allah, Most Gracious, Most Merciful*

All praise and glory to Almighty Allah who gave us courage and patience to carry out this work. Peace and blessing of Allah be upon last Prophet Muhammad (Peace Be upon Him).

We would like express our unrestrained appreciation to our thesis supervisor Dr. Abed El-Hamid Zougbor for his constant help, guidance and tremendous academic support. He has been helping us out with full dedication and truly supported us throughout the course of this work and on several other occasions.

We also acknowledge our colleagues and friends, especially from the Software Engineering Department with whom we had a pleasant, enjoyable and fruitful company, not forgetting the warmest thanks to our Lecturers including, but not limited to Dr. Hassan Qauno, Eng. Ibrahim Al-Shreef, Eng. Eman Al-Ajrami, Dr. Mohammed Awad, Ahmed Al-Astal, for their assistance and support who gave us the knowledge and the guidance throughout our journey in this university, we can't thank them enough for all their support.

We'd like to convey our heartfelt thanks to our home university, University of Palestine for supporting this research and providing us with necessary infrastructure and excellent research environment.

Finally, but by no means least, we extend our acknowledgement and heartfelt love to our families, our loved ones, parents, brothers and sisters for almost unbelievable support who have been with us all the time to spur our spirits. They are the most important people in our world and we dedicate this thesis to them.

# Table of Contents

## CONTENTS

<b>CHAPTER 1: INTRODUCTION .....</b>	<b>2</b>
1.1 Introduction .....	2
1.2 Overview .....	2
1.3 Project Motivation .....	3
1.4 Problem Statement.....	4
1.5 Project Scope .....	4
1.6 Goals & Objectives.....	5
1.7 Contributions .....	5
1.8 Conclusion.....	6
<b>CHAPTER 2: LITERATURE REVIEW .....</b>	<b>8</b>
2.1 Introduction .....	8
2.2 Related Works .....	8
2.2.1 ZeroPC .....	8
2.2.2 Multcloud .....	10
2.2.3 Odrive .....	13
2.3 Comparison .....	17
2.4 Conclusion.....	18
<b>CHAPTER 3: PROJECT METHODOLOGY .....</b>	<b>20</b>
3.1 Introduction .....	20
3.2 Incremental Model.....	20
3.3 Reasons & Advantages for choosing Incremental Model .....	22
3.3.1 Why this this Model?.....	23
3.4 Disadvantages & Limitations of the Incremental Model .....	23
3.5 Increment #1 Phases: .....	24

3.6	Gant Chart .....	27
3.7	Conclusion.....	30
<b>CHAPTER 4: SOFTWARE REQs. SPECIFICATION AND ANALYSIS .....</b>		<b>32</b>
4.1	Introduction .....	32
4.2	Functional Requirements .....	32
4.2.1	Account Requirements.....	33
4.2.2	Files Requirements .....	33
4.2.3	Sharing Requirements.....	34
4.2.4	Management and Access Requirements.....	34
4.3	Non-Functional Requirements .....	35
4.3.1	MCloud Quality Attributes:.....	35
4.4	System Requirements .....	36
4.5	Requirement Analysis.....	37
4.5.1	Application Perceptive.....	38
4.5.2	UML Diagrams.....	38
4.6	Conclusion.....	51
<b>CHAPTER 5: SOFTWARE SYSTEM ARCHITECTURE .....</b>		<b>53</b>
5.1	Introduction .....	53
5.2	System Architecture Overview .....	53
5.3	Model View Controller (MVC) .....	54
5.4	Three-Tier Architecture .....	55
5.5	The MVC implemented in a three-tier architecture.....	56
5.6	Design Patterns.....	62
5.7	Conclusion.....	63
<b>CHAPTER 6 SYSTEM DESIGN AND SPECIFICATIONS .....</b>		<b>65</b>
6.1	Introduction .....	65
6.2	Technology Choices: .....	65
6.3	Choice of Web Design Technologies:.....	69
6.4	Development Tools.....	70

6.5	Class Diagrams .....	71
6.6	MCloud Class Diagram Categories .....	71
6.6.1	Main Class Diagrams.....	71
6.6.2	Models.....	73
6.6.3	Controller .....	75
6.7	Database Model .....	78
6.7.1	Physical ERD Diagram.....	78
6.7.2	SQL Diagram .....	80
6.8	Conclusion.....	82
<b>CHAPTER 7: IMPLEMENTATION .....</b>		<b>84</b>
7.1	Introduction .....	84
7.2	Implementation Plan.....	85
7.2.1	MCloud Website.....	85
7.2.2	MCloud Application .....	85
7.3	Website Implementation .....	86
7.3.1	MCloud Website Graphical User Interface (GUI).....	86
7.4	Application Implementation .....	88
7.4.1	MCloud Application Graphical User Interface (GUI) .....	88
7.4.2	MCloud Application Coding.....	98
7.5	Conclusion.....	108
<b>CHAPTER 8: TESTING AND EVALUATION .....</b>		<b>110</b>
8.1	Introduction .....	110
8.2	Purpose of Testing .....	110
8.3	Testing Methodologies .....	112
8.4	Test Implementation .....	114
8.4.1	DropboxTests Class .....	117
8.5	Test Automation Tools .....	118
8.5.1	Glimpse .....	118
8.5.2	Diagnostic Tools in Visual Studio 2015.....	121

8.6	Testing Checklist .....	123
8.6.1	Functional Testing .....	123
8.6.2	Usability Testing .....	124
8.6.3	Server Side Interface .....	125
8.6.4	Client Side Compatibility .....	126
8.6.5	Performance Testing.....	126
8.6.6	Security Testing.....	127
8.7	Evaluation.....	128
8.8	Conclusion.....	130
<b>CHAPTER 9: FUTURE WORKS AND CONCLUSION.....</b>		<b>132</b>
9.1	Introduction .....	132
9.2	Future Works.....	132
9.2.1	Usability Future Perspective: .....	132
9.2.2	Technical Future Perspective:.....	133
10	Final Conclusion.....	134
<b>REFERENCES .....</b>		<b>135</b>
<b>APPENDICES.....</b>		<b>139</b>

# List of Figures

FIGURE 2 - 1: ZEROPC MAIN USER INTERFACE.....	8
FIGURE 2 - 2: ZEROPC FILE & FOLDER BROWSING WINDOW.....	9
FIGURE 2 - 3: ZEROPC AVAILABLE CLOUD DRIVES .....	9
FIGURE 2 - 4: MULTCLOUD MAIN USER INTERFACE.....	11
FIGURE 2 - 5: MULTCLOUD TRANSFERING FILES WINDOW .....	11
FIGURE 2 - 6: MULTCLOUD AVAILABLE CLOUD DRIVES .....	11
FIGURE 2 - 7: ODRIVE MAIN USER INTERFACE .....	13
FIGURE 2 - 8: ODRIVE MAIN BROWSE WINDOW .....	14
FIGURE 2 - 9 ODRIVE AVAILABLE CLOUD DRIVES.....	15
FIGURE 2 - 10: ODRIVE PROGRESSIVE SYNC FEATURE .....	15
FIGURE 3 - 1: INCREMENTAL MODEL.....	21
FIGURE 3 - 2: INCREMENTAL MODEL AND WATERFALL .....	22
FIGURE 3 - 3: INCREMENTAL MODEL PHASES.....	24
FIGURE 3 - 4: COMMUNICATION PHASE .....	25
FIGURE 3 - 5: MCLLOUD PROCESS SCHEDULING .....	28
FIGURE 3 - 6: GANTT CHART .....	29
FIGURE 4 - 1 USE CASE DIAGRAM .....	45
FIGURE 4 - 2: SET PASSWORD ACTIVITY DIAGRAM .....	48
FIGURE 4 - 3: OAUTH 2.0 AND API CALLS SEQUENCE DIAGRAM .....	50
FIGURE 4 - 4: MVC SEQUENCE DIAGRAM .....	51
FIGURE 5 - 4: MVC IN THREE-TIER ARCHITECTURE.....	56
FIGURE 5 - 5: RESTFUL API.....	60
FIGURE 5 - 7: MVC AS A COMPOUND DESIGN PATTERN.....	62
FIGURE 6 - 1: ASP.NET MVC LOGO.....	65
FIGURE 6 - 2: ADO.NET LOGO .....	67
FIGURE 6 - 3: SQL SERVER LOGO.....	68
FIGURE 6 - 4: AJAX TECHNOLOGY .....	69
FIGURE 6 - 5: FRONT END TECHNOLOGIES.....	69
FIGURE 6 - 6: MAIN CLASS DIAGRAM.....	72
FIGURE 6 - 7: MODEL CLASS DIAGRAM.....	74
FIGURE 6 - 8: CONTROLLERS CLASSES PACKAGE .....	75
FIGURE 6 - 9: ACCOUNT AND APP CONTROLLERS .....	76
FIGURE 6 - 10: THIRD PARTY CONTROLLERS .....	77
FIGURE 6 - 11: PHYSICAL ERD DIAGRAM .....	79
FIGURE 6 - 12: SQL DIAGRAM.....	80
FIGURE 6 - 13: MYUSERS TABLE DETAILED DESCRIPTIONS .....	81
FIGURE 7 - 1: UNIT TEST CLASSES.....	115
FIGURE 7 - 2: THIRD PARTY UNIT TEST CLASSES .....	116
FIGURE 8 - 1: THE TIMELINE TAB .....	120
FIGURE 8 - 2: EVENTS, MEMORY AND CPU UASGE .....	122
A - ACTIVITY DIAGRAM 1: ADD FOLDER.....	175
A - ACTIVITY DIAGRAM 2: DOWNLOAD FILES.....	176
A - ACTIVITY DIAGRAM 3: PREVIEW IMAGES.....	177
A - ACTIVITY DIAGRAM 4: REMOVE CLOUD DRIVE .....	178
A - ACTIVITY DIAGRAM 5: SET PASSWORD .....	179
A - ACTIVITY DIAGRAM 6: SHARE ON SOCIAL MEDIA.....	180
A - ACTIVITY DIAGRAM 7: SHOW STATISTICS .....	181



A - ACTIVITY DIAGRAM 8: UPLOAD FILES .....	182
A- CLASS DIAGRAM 1: DROPBOX REST API MODELS .....	184
A- CLASS DIAGRAM 2: ONEDRIVE REST API MODEL .....	185
A- CLASS DIAGRAM 3: FILE MODEL CLASS DIAGRAM.....	186
A- CLASS DIAGRAM 4: GOOGLE DRIVE API v3.0.....	187
A- CLASS DIAGRAM 5: APPSTART.....	188
A- CLASS DIAGRAM 6: ASPSMSX2SPAP .....	189

# List of Tables

TABLE 2 - 1: SIMILAR SYSTEMS COMPARISON.....	17
TABLE 4 - 1: USE CASE LIST.....	40
TABLE 4 - 2: ACTORS AND THEIR ROLES.....	41
TABLE 4 - 3: TOP LEVEL PRIORITY USE CASES.....	42
TABLE 4 - 4: INTERMEDIATE LEVEL PRIORITY USE CASE.....	43
TABLE 4 - 5: LOW LEVEL PRIORITY USE CASE.....	44
TABLE 8 - 1: FUNCTIONAL TESTING CHECKLIST.....	124
TABLE 8 - 2: USABILITY TESTING CHECKLIST.....	125
TABLE 8 - 3: SERVER SIDE INTERFACE CHECKLIST.....	125
TABLE 8 - 4: CLIENT SIDE COMPATIBILITY CHECKLIST.....	126
TABLE 8 - 5: PERFORMANCE TESTING CHECKLIST.....	127
TABLE 8 - 6: SECURITY TESTING CHECKLIST.....	127
TABLE 8 - 7: EVALUATION CHECKLIST.....	129
A - UC TABLE 1: LOGIN.....	139
A - UC TABLE 2: LOGOUT.....	140
A - UC TABLE 3: FAVORITE FILES.....	141
A - UC TABLE 4: VIEW FAVORITE FILES.....	142
A - UC TABLE 5: MANAGE FILES & FOLDERS.....	143
A - UC TABLE 6: RENAME FILES & FOLDERS.....	144
A - UC TABLE 7: DOWNLOAD FILES.....	145
A - UC TABLE 8: ADD FOLDERS.....	146
A - UC TABLE 9: DELETE FILES & FOLDERS.....	147
A - UC TABLE 10: PREVIEW IMAGES.....	148
A - UC TABLE 11: SHOW STATISTICS.....	149
A - UC TABLE 12: SHOW DRIVES PROFILE.....	150
A - UC TABLE 13: SHARE FILES.....	151
A - UC TABLE 14: SINGLE DRIVE SHARING.....	152
A - UC TABLE 15: MULTIPLE DRIVE SHARING.....	153
A - UC TABLE 16: SET PASSWORD.....	154
A - UC TABLE 17: GET LINK.....	155
A - UC TABLE 18: RESET PASSWORD.....	156
A - UC TABLE 19: SHARE ON SOCIAL MEDIA.....	157
A - UC TABLE 20: AUTHENTICATE BY PHONE NUMBER.....	158
A - UC TABLE 21: UPLOAD FILES.....	159
A - UC TABLE 22: SEARCH FOR FILES OR FOLDERS.....	160
A - UC TABLE 23: VIEW RESULT.....	161
A - UC TABLE 24: ADD CLOUD DRIVE.....	162
A - UC TABLE 25: CONNECT CLOUD DRIVE.....	163
A - UC TABLE 26: SYNC DATA.....	164
A - UC TABLE 27: REMOVE CONNECTED DRIVE.....	165
A - UC TABLE 28: SEND DIRECT MESSAGE.....	166
A - UC TABLE 29: CREATE AN ACCOUNT.....	167
A - UC TABLE 30: VIEW THE WEBSITE.....	168
A - UC TABLE 31: VIEW DIRECT MESSAGES.....	169
A - UC TABLE 32: VIEW SETTINGS.....	170
A - UC TABLE 33: VIEW TASKS.....	171
A - UC TABLE 34: RECEIVE NOTIFICATIONS.....	172
A - UC TABLE 35: VIEW NOTIFICATIONS.....	173

A - UC TABLE 36: COPY TO CLIPBOARD .....	174
A - DB TABLE DESCRIPTION 1: MYUSERS .....	190
A - DB TABLE DESCRIPTION 2: MYUSERCLAIMS .....	191
A - DB TABLE DESCRIPTION 3: MYROLES .....	191
A - DB TABLE DESCRIPTION 4: MYUSERLOGINS .....	192
A - DB TABLE DESCRIPTION 5: MIGRATIONHISTORY .....	192
A - DB TABLE 1: MYUSERLOGINS .....	193
A - DB TABLE 2: MYUSERCLAIMS .....	193

## List of Abbreviations and Acronyms

Abbreviations or Term	Description
Cloud Application	A software application that is never installed on a local machine — it's always accessed over the Internet
Cloud	A metaphor for a global network and commonly used to represent the internet.
Cloud Servers	Virtualized servers that are instantiated via a web interface or API
CaaS	Cloud as a service
Cloud Backup	Backing up data to a remote, cloud-based server
SDLC	System Development Life Cycle
MSDN	Microsoft Developer Network
NuGet	A free and open-source package manager designed for the Microsoft development platform
RESTful	Representational State Transfer Architecture
API	Application Program Interfaces
TDD	Test-Driven Development
DBMS	Database Management System
RDBMS	Relational Database Management System
OOP	Object Oriented Programming
CSS	Cascading Style Sheets
CASE	Computer-Aided Software Engineering
Cookies	A piece of text stored by a user's web browser.
AES	Advanced Encryption Standard
UML	Unified Modeling Language
UI	User Interfaces
UIP	User Interface Process
OS	Operating System
HTML	Hypertext Markup Language
IDE	Integrated Development Environments
AWS	Amazon Web Services
SEO	Search Engine Optimization
ASP	Active Server Pages.
.Net	A software framework for Microsoft Windows operating systems.
MVC	Model-View-Controller Design Pattern
DAL	Data Access Layer
OAuth2	Provides to clients a "secure delegated access" to server resources on behalf of a resource owner



# CHAPTER 1: INTRODUCTION

## Content

- 1.1 Introduction
- 1.2 Overview
- 1.3 Project Motivation
- 1.4 Problem Statement
- 1.5 Project Scope
- 1.6 Goals & Objectives
- 1.7 Contributions
- 1.8 Conclusion

# 1 CHAPTER 1: INTRODUCTION

## 1.1 Introduction

This chapter introduces the reader, stakeholders and developers to the relevant information of this project by giving an overview about the project idea and explaining the objectives and motivations. Then it concentrate on clarifying the problem and how our project will solve this problem and sets the scope and the limitation of the solution.

## 1.2 Overview

Day by day, we are moving towards a trend to develop tools that are less complicated and handy, featuring improved speed and better computing ability. Most trendy users know a wide range of services today enabling greater flexibility on the go. When we're looking for the cloud storage services provided by several software companies nowadays, having only one is almost rare, since most of these cloud storage services provides you with huge storage spaces for no cost and if you want more you can go premium and get extra space. The idea of backing up your data in a secure place and knowing that it will always be there is compelling enough to seek as many free cloud storage as you think you need.

Since most of us seem unable to keep all of their files in one cloud storage, and most online users currently have their data stored in different accounts such as Dropbox, Google Drive and Microsoft OneDrive, also there's different photo services such as Flickr and Instagram, all of which will result in a lot of complexity and makes the accessibility of the desired files much harder.

While the sheer number of different services most people use might be somewhat diverse, the fact that they don't want to waste time managing their stored data on the cloud is a reason to rises the problem. Even within large organizations that would usually be expected to use a single solution, different business units, collaboration with outside contractors and many other reasons results in shaking the core productivity the cloud services expect to provide.

Most of us don't actually need that much space, almost all of the cloud storage services offers free space for just setting up an account on them and that's it, for example Google Drive offers you about 15 GB for free once you setup an account and offers more for

premium deals, but if you did need an extra space, a lot of the users won't choose the premium deals and probably seek a free space from different services such Dropbox.

In most cases the users don't really wait for their Google Drive space to fill up in order to seek another services, they just sign up for a bunch of services without filling half of them, and after some time, uploading files on these services becomes uncontrollable and leads to the underlying issue, and the reasons for that is many, but mostly it's the good marketing that the companies provide to acquire as much clients as they can.

So given that, this unrecognized dilemma is constantly arising and introduces a complexity that user nowadays really don't need, it is important to think of ways to reduce the complexity of this dilemma.

The best way to solve this problem is by finding a way that allows user to access all of their cloud accounts in one place, and all the of files on those different accounts become like they are in one virtual account, that way they don't need to open all of their cloud accounts at once to download some data which exist on more than one account.

Another major feature that those cloud accounts provide is sharing certain data with people regarding to whom, to share some files from different accounts you have to open each one and make "share link" and send it to the targeted people and do the same process all over again which is very frustrating if you have quite data to share, but if all the data from these accounts in one place, you only need to make a single "share link" and that's it.

That's what our project is offering and it's very promising and can actually solve this dilemma dramatically.

## **1.3 Project Motivation**

Google Drive and Dropbox has facilitated the process of storing and managing user's files in one place on the internet and made it accessible by mostly all devices we use nowadays.

But what came obvious is the number of the cloud storage services is increasing and each one is offering exclusive features to gain more users which result in leaning most of the users to sign up for as many services as possible, and with several different cloud storages for users, the management part is taken away from the equation and users are left to face the struggle increasing more with time, also a lot of them will eventually stop using those many services in order to stop the chaos from escalating, which will limit their space

storages and prevent them from experiencing other features that might not be available in the service they settled on.

## **1.4 Problem Statement**

Files are everywhere and people nowadays are just collecting more and more, and one of the best places to store it is by using online cloud storages such as Dropbox and Google Drive; It's a great solution and it's widely accepted. Recent study revealed that in 2016, 65 % of the EU population used internet storage space for saving or sharing files.

Unfortunately it recently came with downsides that truly affected such services, the online storages services that are now available are too many and most people already use several storages which lead scattered files across multiple websites with varying user interfaces, privacy and security protocols, and multiple logins to remember.

This will make people feel overwhelmed with the process of finding, managing and sharing their files online.

In order to prevent this dilemma, we came with a promising solution that will effectively solve the problem, a web application which will unify the most popular online cloud storage into one place, the user only need to make an account and connect the cloud storages that he want.

The application will then sync all of their files and allow users to browse all the synced files from the different cloud storages in the same application without switching tabs or keep logging in and out. Also the application will provide users with a lot of options to manage and organize their files in addition to other handy features.

This solution will give the user a way which will help them save a lot of time and ease the process of handling files online with the optimal file management program.

## **1.5 Project Scope**

MCloud is a responsive web-based application that makes data accessible, independent of storage platforms such as Google Drive, One Drive and Dropbox; and unify these data into a single place with a single login. MCloud will be the first release following the software methodology we choose and adopted.



Currently MCloud is developed as web-based application, but in future releases it will be available for mobile platform such as Android & IOS which also be bringing along new features and functionalities to become more optimized and give better experience for the users.

MCloud will be completely free for this release and it will stay that way until settling for the optimal release which will introduce pricing plans with more exclusive features along the main free plan which will be more than enough for most users.

The system has no limit regarding region or country and it will be available world-wide, but will focus in giving the best experience possible for the Arab states because it will be the first system of this service to support Arabic language along with English as well.

## **1.6 Goals & Objectives**

- The Ability to easily connect online cloud storages together in one place.
- Introducing a unified, easy and secure way to access all files from the cloud storages with a single login.
- Facilitate managing and organizing all the cloud files with a user-friendly and easy to use interface.
- Enabling users to search & discover everything in their cloud files across different cloud storages.
- Securely share any files or folders instantly from a single or multiple cloud storages on various social media websites.

## **1.7 Contributions**

- ❖ Cloud storages is better when it's unified. MCloud utilizes all the benefits from the accessibility of cloud storages by unifying all the storages into one system. One

password, one application.

- ❖ It will save a lot of time for user since all files in one place which is easier to do all the functionalities necessary quickly and more conveniently rather than switching between different applications and keep logging in and out.
- ❖ Sharing files online by cloud storages is essential to most internet user nowadays, MCloud focuses in making the process of sharing files across multiple cloud storages as easy and practical as possible.
- ❖ While there's a few systems that provide similar functionalities to our project but no one support Arabic language. Thus, MCloud will be supporting both Arabic & English and will also introduce features in futures releases that will help the Arab communities to get the best benefits of cloud storages and its capabilities.

## **1.8 Conclusion**

MultiCloud is a cloud service that can put multiple cloud drives into one. It mainly serves for the users who have multiple cloud drives and need to access all their files in a single place across clouds. They are allowed to manage, organize and share all the files of their cloud drives through just one single account in MCloud.



# CHAPTER 2: LITERATURE REVIEW

## Content

- 2.1 Introduction
- 2.2 Related Works
  - 2.2.1 ZeroPC
  - 2.2.2 Multcloud
  - 2.2.3 Odrive
- 2.3 Comparison
- 2.4 Conclusion

## 2 CHAPTER 2: LITERATURE REVIEW

### 2.1 Introduction

This chapter provides an overview of previous research and give a demonstration of the relevant literature describing prior work and similar systems in the field which followed with a discussions on what others have done, what they have achieved, and limitations of their work. If they exist, and finally shows a summarize comparison between the business model and the similar system that currently exist in the business market.

### 2.2 Related Works

#### 2.2.1 ZeroPC

##### 2.2.1.1 Overview

ZeroPC [1] is a cloud content management user interface that turns a browser into a virtual desktop and allow users to unify all of their digital content from popular Internet sites especially cloud storage services, and also from their own computer, into one secure repository requiring a single login.

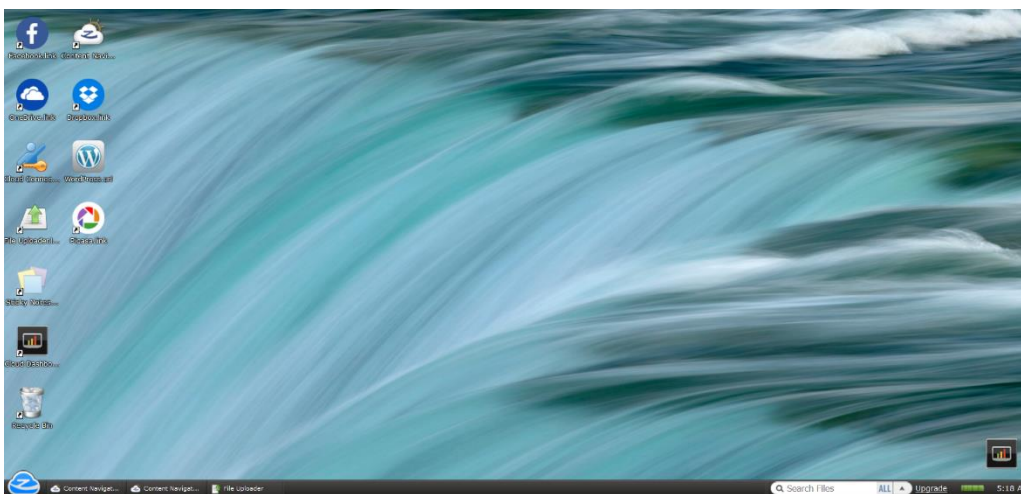


Figure 2 - 1: ZeroPc Main User Interface

A user logging in to ZeroPC is presented with an interface much like Microsoft Windows: icons on a desktop that provide access to files stored in folders and to applications for e-mailing, document editing, and more. But the desktop is delivered using the same technologies used to build interactive Web applications.

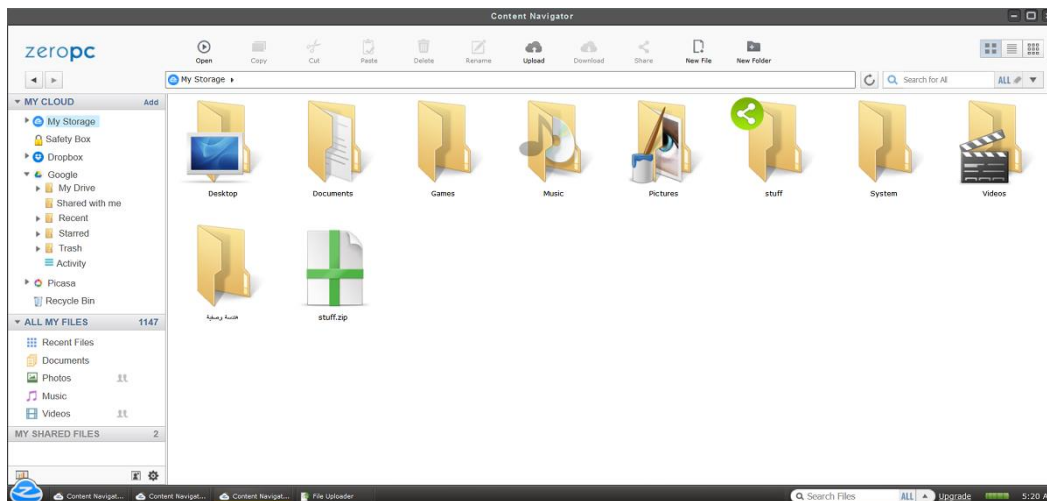


Figure 2 - 2: ZeroPc File & Folder Browsing Window

Through-out the past years ZeroPC can now provide integration with popular content sharing services Google Drive, Dropbox, Box.net, Microsoft OneDrive, SugarSync and a few others, also some photo sharing services such as Flickr, Instagram.

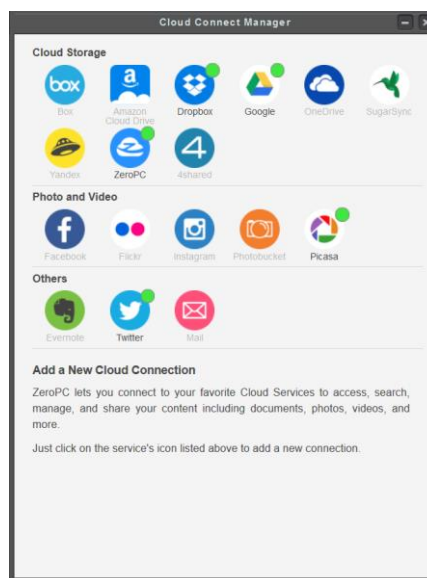


Figure 2 - 3: ZeroPc Available Cloud Drives

With ZeroPC, users begin by moving their content online, so they can access it from any location. Once all documents, videos, photos, emails and social networking tools are all in one place, users can connect their information and files using their laptop, tablet, smartphone or even desktop computer- as long as it has a browser. After they are connected, users can share their information privately or publicly using a secure link or on social media websites.

### **2.2.1.2 Advantages**

These are the major advantages that ZeroPC provides:

- Support integration with many popular cloud drives websites and content sharing services.
- The Windows-like Interface improved the user experience significantly.
- Offers reliable security & capabilities since it's hosted on the Amazon Web Services (AWS) infrastructure.
- Google Docs & Microsoft Office Online integration.
- It's available on other platforms such as Android and IOS.

### **2.2.1.3 Disadvantages**

These are some of the disadvantages that affects ZeroPC:

- The process of managing and organizing the user's content from many sharing services can be overwhelming.
- Does not provide statistics about the user's storage space and files.
- The uploading process to the cloud accounts is not that practical.
- A lot of the important features are exclusive for the premium accounts.

## **2.2.2 Multcloud**

### **2.2.2.1 Overview**

MultCloud [2] is a cloud service that can put multiple cloud drives to one. It mainly serves for the users who have multiple cloud drives and need to transfer files across clouds. They

are allowed to manage all the files of their cloud drives through just one single account in Multcloud website.

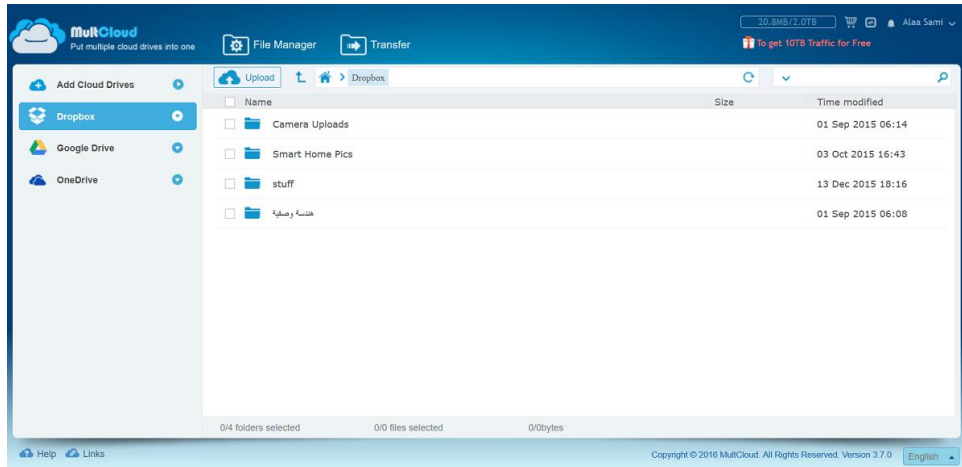


Figure 2 - 4: MultCloud Main User Interface

Multcloud strongest feature is transferring files between different cloud drives, which uses the new background of transmission technology, making the files transfer doesn't need to start the computer, that is to say, in the case of power off, MultCloud can also realize the transmission from a cloud drive to another, the user only need to Copy and Paste, then MultCloud server will handle the transferring process automatically with configuration previously set by the user.

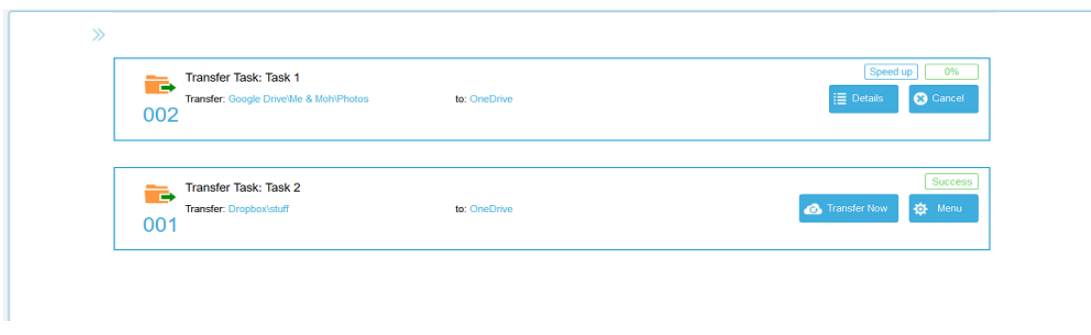
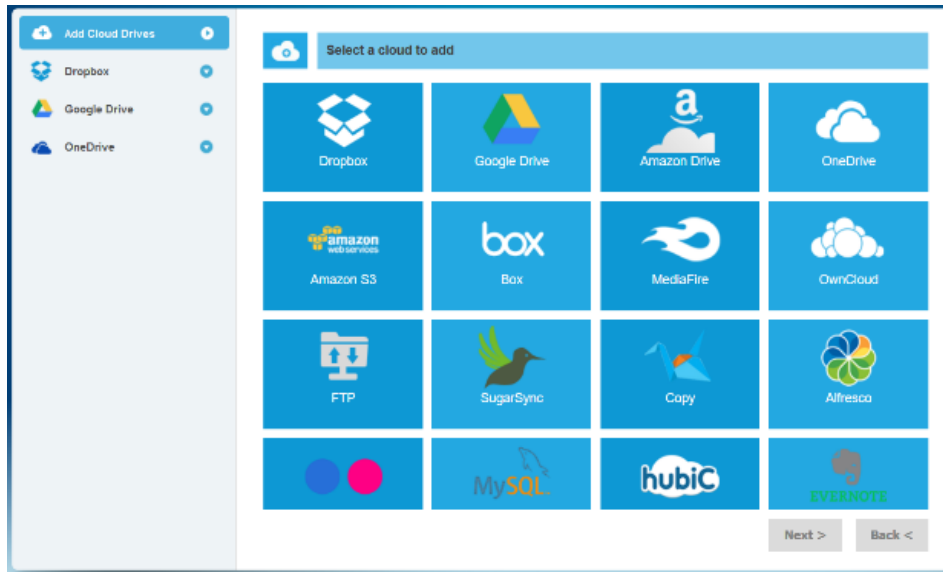


Figure 2 - 5: MultCloud Transferring Files Window

Another bounce for the transferring feature is scheduling, for instance some files may need to backup regularly, for example, make a data transferring on the last day of every month. MultCloud supports scheduled transfer function. It allows the user to set the daily, weekly

Figure 2 - 6: MultCloud Available Cloud Drives

or monthly and also with a specific time for data transmission. It also supports to send an email to you after completing the transmission and inform the result.



While MultCloud supports most of the popular cloud drives and content sharing serveries, but it can actually support multiple accounts from the same cloud drive which provides more storage space to use for free. Generally speaking, each cloud drive will just provide a small free storage space. Thus, a big one will come as a result after adding all free space of this cloud drives together. For instance, if the user have registered five Dropbox accounts and each one has offered 5GB free space, you can use the free 25GB space conveniently with the help of MultCloud into your account at the same time.

### 2.2.2.2 Advantages

These are the major advantages that MultCloud provides:

- Support integration with many popular cloud drives websites and content sharing services.
- File transferring doesn't require the user to be online throughout the transferring process which still transfer even after power off.
- Scheduled transfer allow the user to set daily, weekly or monthly data transmission between cloud drives.



- Multi-server parallel transmission which improves the data transmission speed and stability.

### 2.2.2.3 Disadvantages

These are some of the disadvantages that affects MultCloud:

- Doesn't provide a detailed report when finishing an offline transmission of data.
- Doesn't provide dashboard or task manager for handling data transferring tasks.
- Not available on other platforms such as Android and IOS.
- Does not provide statistics about the user's storage space and files.

## 2.2.3 Odrive

### 2.2.3.1 Overview

The idea of Odrive [3] is to give users a traditional folder interface and very similar to a Dropbox-style interface for browsing all of their online files from many cloud and content sharing services in a single place.

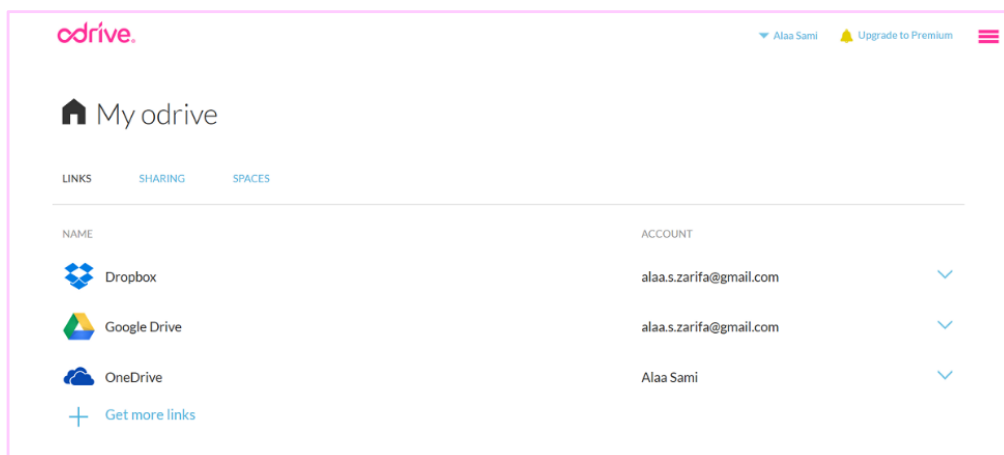


Figure 2 - 7: Odrive Main User Interface

While odrive has a mediocre web application, but is essentially focusing on the desktop platform and without a doubt it's impressively capable, and generally works just like you'd expect from the top cloud storage providers, but its web interface is minimalistic, sometimes to a fault. You can download files, delete files, upload files with a simplistic drag and drop interface, add new folders, and get sharing links.

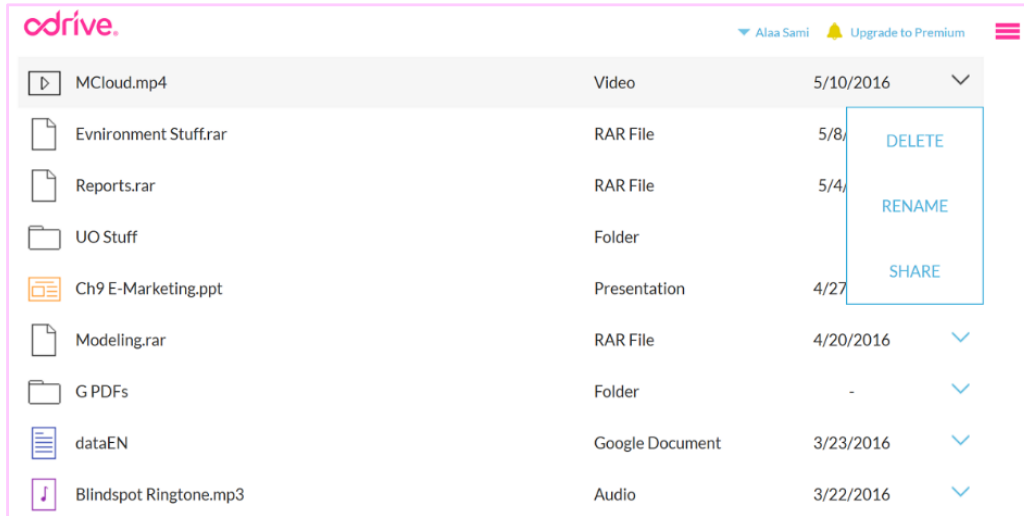


Figure 2 - 8: Odrive Main Browse Window

But that's literally it. There's no moving files, no web-based editing, and other than a simplistic photo viewer, there's no way to open another type of files or anything else in the browser.

Currently, odrive support many cloud services such as Amazon Cloud Drive, Google Drive, Box.com, OneDrive, Dropbox, Copy, Oxygen Cloud, social media images Facebook, Picasa, Instagram, and Flickr, and even Gmail backup.

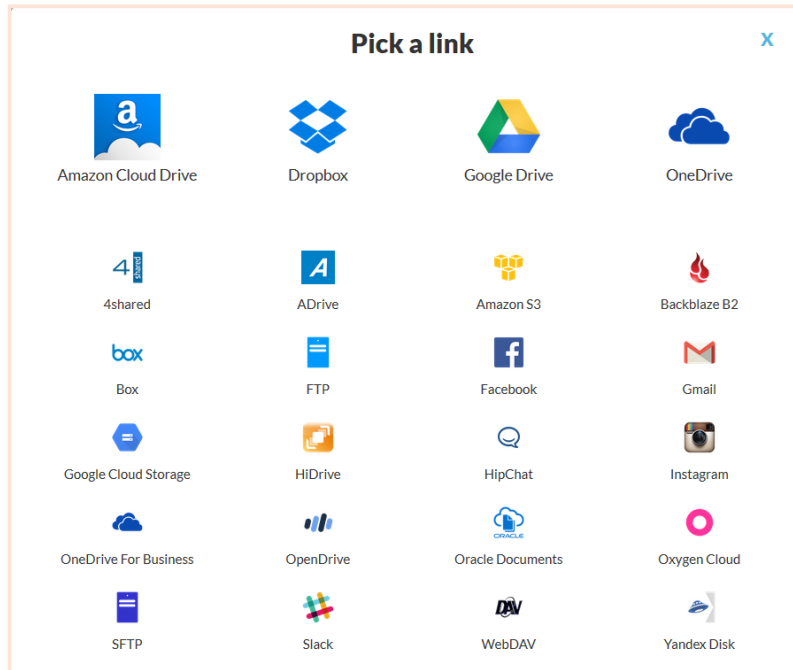


Figure 2 - 9 Odrive Available Cloud Drives

Odrive major feature that makes the desktop application stands is “Progressive Sync” where files that are larger than 10 megabytes only sync when the user select them. That means giant files aren’t downloaded to your hard drive unless you want them to be.

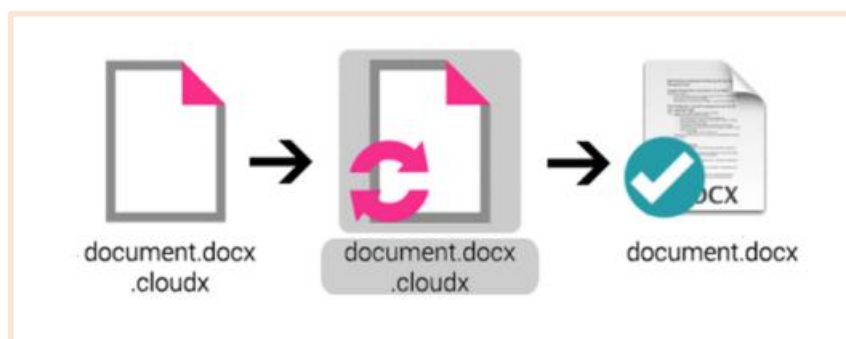


Figure 2 - 10: Odrive Progressive Sync Feature

At first, the software inserts zero-byte placeholder files called CLOUDFX files to represent your linked cloud services and once you select which files the user wants, they automatically syncs and downloaded automatically to the user hard disk, and to save more space on the disk, the users can un-sync these files when the user no longer need them which will return to the placeholder state.

### **2.2.3.2 Advantages**

These are the major advantages that odrive provides:

- Support integration with many popular cloud drives websites and content sharing services.
- Provide access for synced files whether the users are online or offline, on the same network or different networks.
- While “progressive sync” provides seamless sync functionalities, it’s also a logical hard-drive space and bandwidth saving feature.
- Provides excellent search functions by using hashtag searches that will be updated in real time.

### **2.2.3.3 Disadvantages**

These are some of the disadvantages that affects odrive:

- The web application isn’t providing much features and it’s very limited.
- Doesn’t allow transferring files and data between drives.
- It takes a little while to get everything ready for new users.
- Not available on other platforms such as Android and IOS.
- Does not provide statistics about the user’s storage space and files.

## 2.3 Comparison

This table shows the most important differences and advantages between MCloud and the similar systems that were already been presented, It also summarizes the most important features of MCloud.

Features	ZeroPC	MultCloud	Odrive	MCloud
Web Application	<ul style="list-style-type: none"> <li>• Web Application</li> <li>• Desktop Application (Soon)</li> </ul>	<ul style="list-style-type: none"> <li>• Web Application</li> </ul>	<ul style="list-style-type: none"> <li>• Web Application (Limited)</li> <li>• Desktop Application</li> </ul>	<ul style="list-style-type: none"> <li>• Web Application</li> </ul>
Personal Cloud Account And File Manager	Available	Available	Available	Available
File Uploader	Available	Available	Available	Available
Single Sharing Link For Data From Multiple Cloud Accounts	Not Available	Not Available	Not Available	Available
Fast Access to Favorite Files From Any Cloud Accounts	Not Available	Not Available	Not Available	Available
Integration With Cloud Drives And Sharing Services	Many	Many	Many	<ul style="list-style-type: none"> <li>▪ Google Drive</li> <li>▪ OneDrive</li> <li>▪ Dropbox</li> </ul>
Storage Spaces And Files Statistics	Limited	Not Available	Not Available	Available
Pricing	<ul style="list-style-type: none"> <li>▪ Free</li> <li>▪ Premium (extra features)</li> </ul>	<ul style="list-style-type: none"> <li>▪ Free</li> <li>▪ Premium (extra features)</li> </ul>	<ul style="list-style-type: none"> <li>▪ Free</li> <li>▪ Premium (extra features)</li> </ul>	Free
Arabic Language Support	Not Available	Not Available	Not Available	Available
Mobile Application	Available	Not Available	Not Available	Not Available
Responsive Design	Not Available	Not Available	Available	Available

Table 2 - 1: Similar Systems Comparison

## **2.4 Conclusion**

MCloud is a Web-based Application built to help people consolidate all of their content from the major cloud drives without the hassle of migration and providing all these contents in a single place with single login that can be accessed via any browser, and provide only the necessary features that actually will be used without the not-needed features that will only make the experience overwhelming for the user.



# CHAPTER 3: PROJECT METHODOLOGY

## Content

- 3.1 Introduction
- 3.2 Incremental Model
- 3.3 Reasons & Advantages
- 3.4 Disadvantages & Limitations
- 3.5 Increment #1 Phases:
- 3.6 Gant Chart
- 3.7 Conclusion

## **3 CHAPTER 3: PROJECT METHODOLOGY**

### **3.1 Introduction**

The success of a software development project greatly depends upon which process model is used. This chapter emphasizes on the need of using appropriate model as the application to be developed. In this chapter we choose the Incremental model as our software model in the development lifecycle which is essential to provide a method for developing a software product. A proper software life cycle model can help us not only in building a software product but it also serves as a basis for planning, organizing, staffing, coordinating and directing various other software development activities.

Our aim is to create reliable and cost effective software and this models provide just that to start developing the system, but before going through the stages of the development, we first addressed in the chapter the advantage and disadvantages of this particular model and what exactly drive us to choose it regardless of its limitations.

The Gantt chart is also added to manage our project since it's almost essential in most IT project nowadays, which primary used to provide a graphical overview and schedule of all tasks of our project. [4]

### **3.2 Incremental Model**

Is a method of software development where the product is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements.



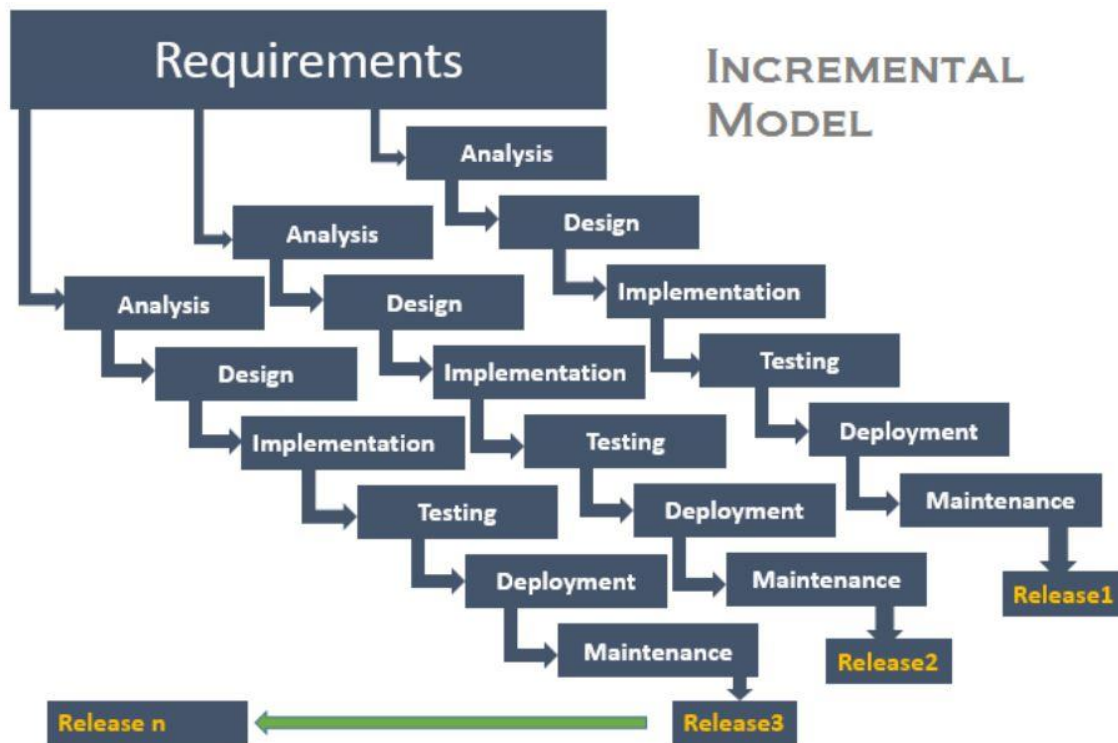
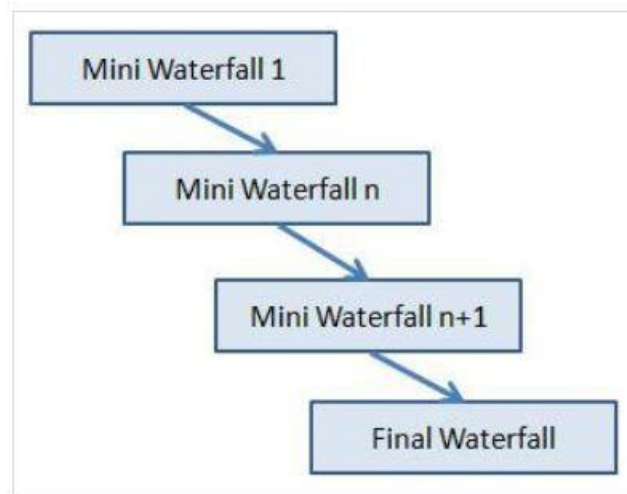


Figure 3 - 1: Incremental Model

The basic idea behind this method is to develop a system through repeated cycles that produces series of releases that referred to as “increments”, with each increment providing more functionality to the customers. After the first increment, a core product is delivered, which can already be used by the customer. Based on customer feedback, a plan is developed for the next increments, and modifications are made accordingly. This process continues, with increments being delivered until the complete product is delivered. [5]

This model combines the elements of the waterfall model with the iterative philosophy of prototyping. It is developed to overcome the weaknesses of the waterfall model. It starts with an initial planning and ends with deployment with the cyclic interactions in between. Increment process model are effective in situations where requirements are defined precisely and there is no confusion about the functionality of the final product. [6]



*Figure 3 - 2: Incremental Model and Waterfall*

In this model no need to wait for all the modules to be developed and integrated. As each module is standalone application and there is no dependencies on other modules so we can deliver the project with initial developed feature and other features could be added on incremental basis with new releases.

### **3.3 Reasons & Advantages for choosing Incremental Model**

Incremental model is able to generates working software quickly and early during the software life cycle. It also helps ease the traumatic effect of introducing completely new system all at once. Along with that, here's a list of additional advantages from using this model: [7]

- Lowers initial delivery cost.
- Easier to test and debug during a smaller iteration.
- More flexible – less costly to change scope and requirements.
- Initial product delivery is faster, since the first release is usable.

- Easier to manage risk because risky pieces are identified and handled during its iteration.
- Guaranteed client satisfaction, because after using the first release, user can give their suggestion and feedback then changes will apply in the next release.

### **3.3.1 Why this this Model?**

All of the previous advantages will suffice and guarantee a smooth and seamless environment for developing our project, but since our team consist only of three members and the time we have is very limited considering our project is quite large, besides that, our project will provide a lot of functionality which will take a lot of time and serious effort to finish a complete and a final version of the software.

This situation drive us to look carefully and do a great deal of research to find the best model that can effectively fix this critical situation, and after serious research and consulting we found that the incremental model is the solution for this particular problem since as mentioned earlier, the first increment is basically a core product where the basic requirements are addressed and the supplementary features are added in the next increments.

Based on that, we will be working only on the first version/increment and after we deliver it as the first official release of the project keeping in mind that it will be a completely tested and usable version with all the basic and the necessary features. In future works a development plan will be set for the next versions after reviewing all the feedback from the users who try the first versions. Thus in every increment, more features and functions are added and the core product is updated. This allows partial utilization of product and avoids a long development time.

## **3.4 Disadvantages & Limitations of the Incremental Model**

This model has very fewer drawbacks because this was designed under consideration to overcome the drawback of waterfall model of SDLC. Regardless, here's a list of disadvantages and limitations that canaries from using this model: [7]

- Needs a clear and complete definition of the whole system before it can be broken down and built incrementally.
- Integration between iteration can be an issue if this is not considered during the development.
- The total cost of this system is higher than waterfall model of software development life cycle.
- Needs good planning and design.

### 3.5 Increment #1 Phases:

In this system modeling approach the developer dividing the functionality into different categories. By applying multiple linear sequential phases similar to waterfall model enhance the functional categories of the system with each successive release of the products until all functional categories are covered. The ultimate goal of this model is to provide a system with overall functionality.

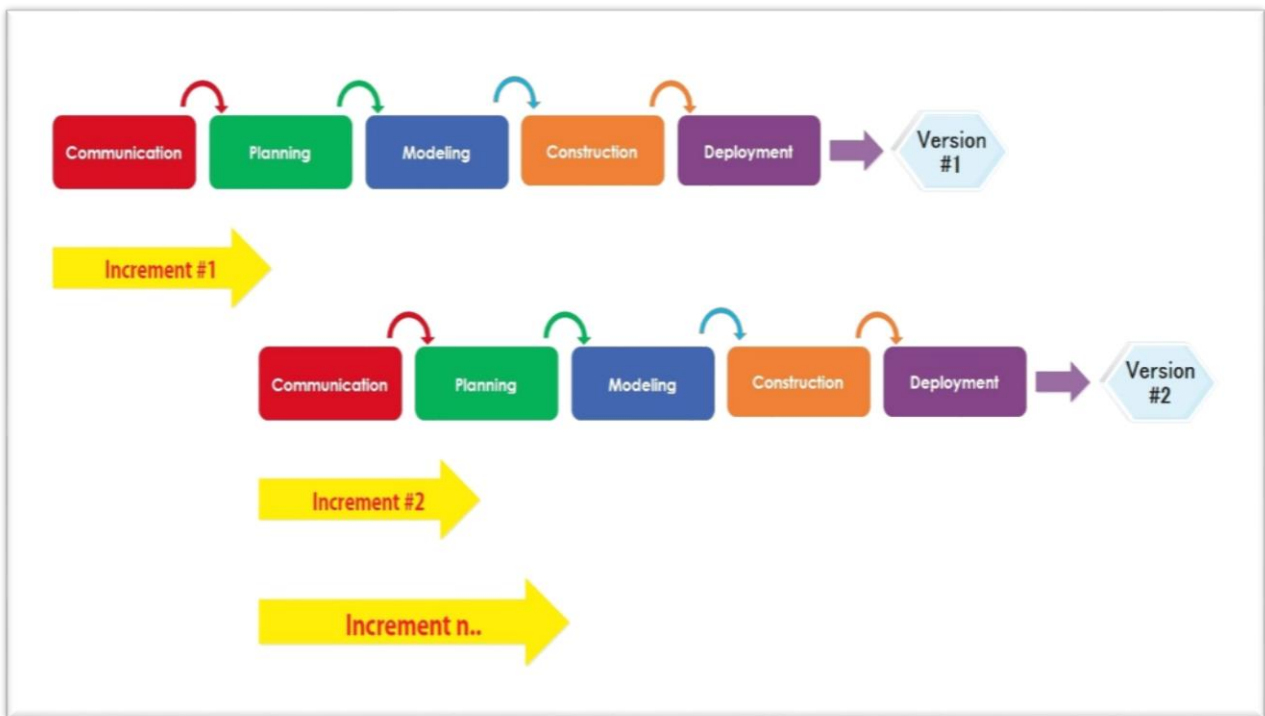


Figure 3 - 3: Incremental Model Phases

### 3.5.1 Communication

Knowing what the problem is, helps to investigate and know whether it's necessary to dive into the development cycle correctly, but in order to that, a good communications and collaboration between our team members is required and essential to provide the best environment for developing our project, and before MCloud team go through the development process all of the necessary aspect of communications were presented and clarified to each member of our team.



*Figure 3 - 4: Communication Phase*

This step is important in this model since pre-setting the right communication methods between the team members saves time and guarantee a comfortable environment for the development cycle and prevent chaos that might happen otherwise, also the lack of communication will cause our MCloud team to doubt if the project is on time, if it will be delivered and if the system is going as agreed. This is why one must make sure to have the framework and procedures in place to keep everyone updated with the project status.

### 3.5.2 Planning

The need of proposed software needs to be well understood and know the market that the software focuses on, and that's what really define this critical stages. The outcome from the planning stages is essentially focuses in gathering and understanding all the requirement for our project which involves the system functional and nonfunctional requirements.

Since this phase plays an important role in the development of product under incremental model of software life cycle, the MCloud team did a huge amount of effort to clearly

identify all the requirements and the system specifications by all the necessary tools available which were documented in details later in chapter 4.

### **3.5.3 Modeling**

This phase actually describes how the solution will work, once the structure of our project is confirmed by all of our team members, the architecture of the system will be designed and created, and for our system to be successful we spend enough time to carefully analyze the project requirements and specifications in order to understand the features, functions, behavior and performance of intended the system.

Design of the incremental model saves time as the design of the first phase can be modified or part of the design can be reused. The other thing that saves time is the use of the output of the first functional unit to the next unit. This is being done with the help of feedbacks from the customers. [8]

Different models for our project are created depending on the requirements gathered in the first phase and the planning done in the second phase, these models involves Business Modeling (i.e. UML), Data Modeling (i.e. Database Models), and Process Modeling (i.e. Gantt chart) and another required diagrams or UI sketch necessary for the project development.

### **3.5.4 Construction**

The actual coding of our project is done in this phase, but we didn't starts coding the moment we finished the software design, but rather we carefully reviewed every component of the project and then we started coding on the basis of the models designed in the modeling phase. Hence in this phase the project is actually developed and tested.

Testing in this phase should not be done at the end of the software as it can be dangerous. Rather we have done it in a timely manner throughout the whole stage and especially by the use of unit testing which where the testing of each individual components are done to check if there are working without bugs or other problems. All of the remaining components and the software features are developed and integrated into each other.

The implementation process of MCloud was diverse in the sense of number of tools and coding techniques in which applied to develop the project and the details of implementation tools and how we use them, is documented in details in chapter 7.

### **3.5.5 Deployment**

The purpose of this deployment phase is to put the product into production. After our team tests the system successfully, the product is ready to go live and ready to be used in a real environment by end users. During this phase, the security subject is important since the user will be using the system so we reviewed all the security risks that were identified in previous phases and verified to ensure they have been fixed.

Once the deployment process finishes, users will begin using the software, any issues encountered with the system functionality, they can easily give their feedback and suggestions in order to repair the issues in the next version of the software to maintain the quality of the product.

## **3.6 Gant Chart**

A Gantt chart provides the illustration of a schedule that helps to plan, coordinate, and track specific tasks in a project. It usually consists of two parts: the project tasks on the left and the time allocation for each task on the right. Gantt charts are helpful in many aspects but essentially they can deal with scheduling tasks, understanding critical paths of a project, and planning of resources. [9]

MCloud Process Scheduling					
TASK #	INCREMENT #1 PHASES	TASKS	DUARNATION (days)	START DATE	END DATE
1	Initiation & Communication		3	10/20/2015	10/23/2015
2		Status Meetings	2	10/20/2015	10/22/2015
3		Understanding the Methods to Communicate	1	10/24/2015	10/25/2015
4	Planning		54	10/26/2015	12/20/2015
5		Brainstorming	3	10/26/2015	10/29/2015
6		Researching for Cloud Related Issues	14	11/1/2015	11/15/2015
7		Consulting and Gathering Informations	15	11/15/2015	11/30/2015
8		Setting Requirements	12	12/2/2015	12/14/2015
9		Phase Documentation	5	12/15/2015	12/20/2015
10	Modeling		63	12/22/2015	2/25/2016
11		Software Methodology Analysis	15	12/22/2015	1/7/2016
12		Requirements Specification & UML Diagrams Development	8	1/9/2016	1/17/2016
13		Architectural Requirements Specification	6	1/18/2016	1/24/2016
14		Technical Design Development	3	1/26/2016	1/29/2016
15		Logical Database Design	4	2/9/2016	2/13/2016
16		User Interface Design	3	2/13/2016	2/16/2016
17		Design Acceptance	2	2/17/2016	2/19/2016
18		Phase Documentation	5	2/20/2016	2/25/2016
19	Construction		65	2/29/2016	5/5/2016
20		Coding	32	2/29/2016	4/2/2016
21		Unit Testing	23	4/5/2016	4/28/2016
22		Test Plan Updates	11	4/5/2016	4/16/2016
23		Test Data Validation	12	4/16/2016	4/28/2016
24		Test Plan Finalization	2	4/29/2016	5/1/2016
25		Final Evaluating	9	3/22/2016	3/31/2016
26		Phase Documentation	3	5/2/2016	5/5/2016
27	Deployment		22	5/25/2016	6/17/2016
28		Reviewing Hosting Options	4	5/25/2016	5/29/2016
29		Security Risks Review	3	6/2/2016	6/5/2016
30		Begin Software Installation	2	6/7/2016	6/9/2016
31		Setting Support & Feedback Methods	5	6/10/2016	6/15/2016
32		Phase Documentation	1	6/16/2016	6/17/2016
33	Final Phase		11	6/20/2016	7/1/2016
34		Check System Functions	4	6/20/2016	6/24/2016
35		Setting Furtre Works	1	6/25/2016	6/26/2016
36		Finalizing Documentation File	4	6/27/2016	7/1/2016

Figure 3 - 5: MCloud Process Scheduling



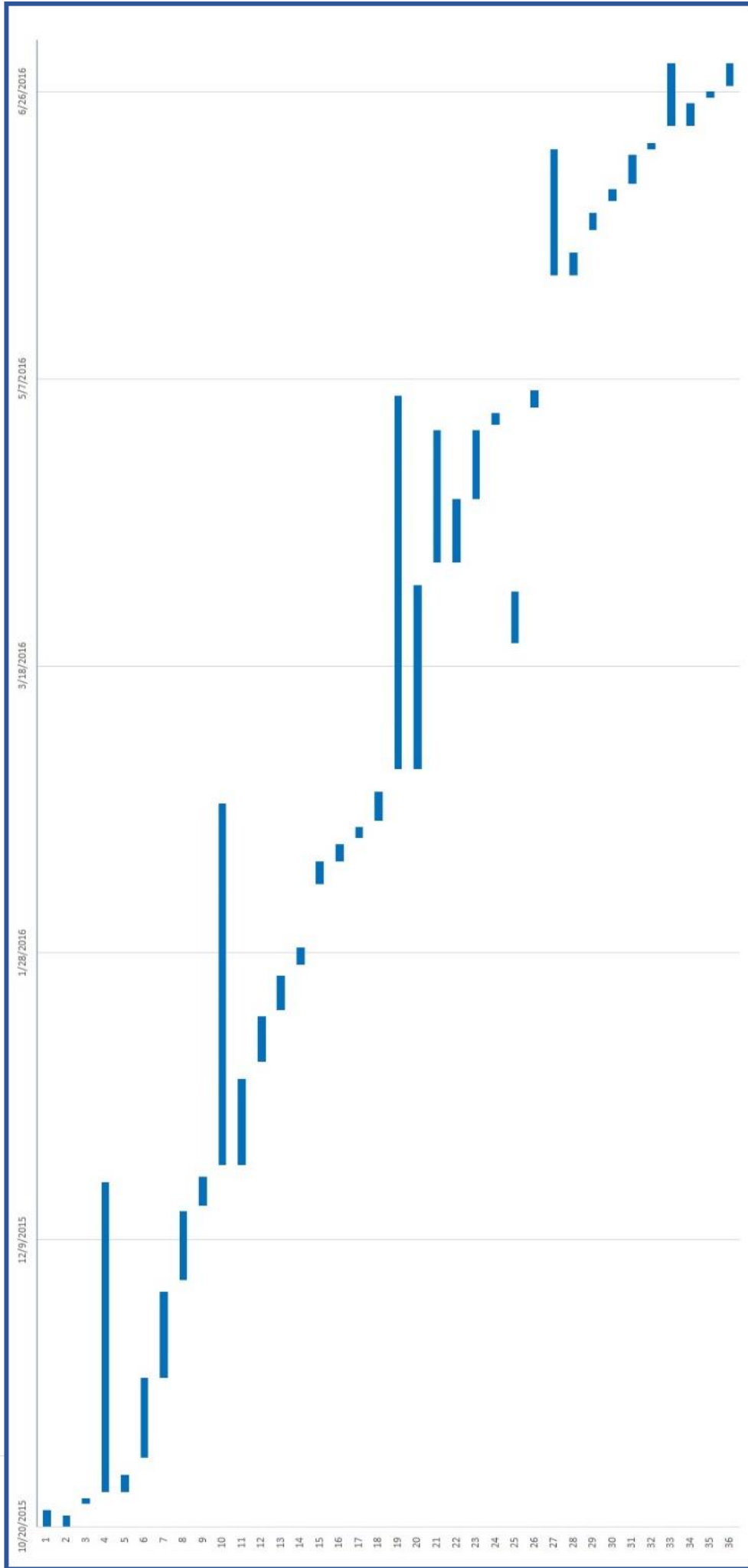


Figure 3 - 6: Gantt Chart

## **3.7 Conclusion**

The Incremental model was our choice for our software development methodology, where model is analyzed, designed, tested, and implemented incrementally. We also address the advantage of using this model and clarify the main reasons for choosing the model regardless of the few disadvantages that might occur which were mentioned too in the chapter, in addition to that, the chapter provides overview of the stages we gone through to develop MCloud system, also as a way of driving the processing models further, we created a Gantt chart which was an effective method for managing and task in a time basis.



## Chapter 4: Software requirements specification and analysis

### Content

- 4.1 Introduction
- 4.2 Functional Requirements
- 4.3 Non-Functional Requirements
- 4.4 System Requirements
- 4.5 Requirement Analysis
- 4.6 Conclusion

# **4 CHAPTER 4: SOFTWARE REQUIREMENTS SPECIFICATION AND ANALYSIS**

## **4.1 Introduction**

A thorough discovery of business requirements is almost never readily available at an analyst's fingertips—rarely can requirements be quickly looked up as one would gather information for a term paper or study for a test. Much of business or technical requirements is not documented anywhere—it resides in the minds of stakeholders, in feedback that has yet to be obtained from end users, and from a study of flowcharts and surveys that have yet to be created. And so requirements must be elicited, or drawn out, and the methodology in doing so must be logical and meticulous. This chapter begins with listing the functional requirements of MCloud system which's aims to capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions system is required to perform then we introduced the non-functional requirements of the system which describe aspects of the system that don't relate to it's execution, but rather to its evolution over time. Then the system requirements mentioned as well. Furthermore the chapter continues with the analysis made for the requirements which is critical to the success of a development project. Requirements must be actionable, measurable, testable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design. The major tool and technique applied for the design and analyzing of requirements is UML modeling which makes understanding the logic of the system much better and help knowing how, when the system will interacts with other components.

## **4.2 Functional Requirements**

Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. [10]

## 4.2.1 Account Requirements

- ❖ MCloud system must allow the user to login externally through multiple options using third-party websites (i.e. Facebook, Google+, Twitter, etc.) or using the standard website registration.
- ❖ MCloud system must validate the user's login and the info from third-party websites by several options such as username and password, phone number or email verification.
- ❖ MCloud system must allow the user to connect to all or any of the three cloud drives available which are (Google Drive, Dropbox and MS OneDrive).
- ❖ MCloud system must allow the user to remove and disconnect from any connected cloud derive at any time.

## 4.2.2 Files Requirements

- ❖ MCloud system must have a tree-view to show all the folders and the files from all the connected cloud drive.
- ❖ MCloud system must allow the user to upload files from his PC into any connected cloud drive if there's space available.
- ❖ MCloud system must allow the user to do most of the major functionalities regarding files and folders such as Add folders, Rename or Delete Files & Folders and Download Files.
- ❖ MCloud system must allow the user to see details of the files inside the file manager from any connected cloud drive such as Name, Type, Size and the Uploaded Date.
- ❖ MCloud system must allow the user to preview Images of any format inside the software.

### **4.2.3 Sharing Requirements**

- ❖ MCloud system must allow the user to quickly share certain files via links on social media websites, these links will allow to download the shared file online.
- ❖ MCloud system should allow the user to share several files from multiple cloud drive in one single sharing link.

### **4.2.4 Management and Access Requirements**

- ❖ MCloud system must enable the user to search for any file or folder inside the connected cloud drives.
- ❖ MCloud system should provide the user with all statistical charts and Information related to the storage space of the connected cloud drives.
- ❖ MCloud system must allow the user to set any files as favorites and provide the user with quick access to these favorite files at any time.
- ❖ MCloud system must allow the user to check the received notifications.
- ❖ MCloud system must allow the Admin to send Direct Messages to users and allow users to check these direct messages.
- ❖ MCloud system must allow the user to quickly check setting which shows the general setting of the system.
- ❖ MCloud system must allow the user to quickly check the current task regarding the system.

## 4.3 Non-Functional Requirements

Requirements that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors, in other words, how well some behavioral or structural aspect of the system should be accomplished. [11]

### 4.3.1 MCloud Quality Attributes:

#### ❖ Availability

A robust software should not lose its availability even in most failure states, and that's why MCloud system took as many precautions as possible to prevent such issue, even if some components are broken down due upcoming updates or releases, it will continue running and stay available 24/7.

#### ❖ Usability

User Interfaces are the only visible parts of MCloud system according to the viewpoint of user, so it was essential for our team to make the UI as simple and user-friendly as possible, and the first time anyone uses MCloud system will surely have no problem figuring out how to handle the system.

#### ❖ Flexibility

Change is inevitable in software development and so, as mentioned before that MCloud system will be delivered on increments basis and by that every increment will have to introduce new functionalities and modifying existing ones if necessary without damaging current system, so it's important for MCloud system to be flexible as needed.

#### ❖ Performance

To ensure the performance of MCloud system, we apply the necessary tools to achieve a high-level system responsiveness, also we reduced the latency as much as possible to ensure smooth and fast experience to the user while using the system.

## ❖ **Manageability**

Managing MCloud system should not be taken for granted, because it's an essential process that guarantees quality regarding system development management, it focuses mainly on bringing an efficient way to deploy, configure, upgrade, and monitor all of the components and services to keep the system performing, secure, and running smoothly.

## ❖ **Security**

To establish a secure connection between MCloud and the other storage services we abide by the OAuth2 implementation provided by the target storage service. Upon completion of the OAuth2 process, MCloud stores the relevant access token in the user's profile. This token, along with all other sensitive password related data is encrypted using (AES 256-bit key size.). Furthermore, we ensured that users are identified and that their identities are properly verified and made sure that they can only access data and services for which they have been properly authorized.

## **4.4 System Requirements**

The hardware and software requirements of MCloud system that are required to use software efficiently.

### **4.4.1 Hardware Requirements**

- Processor: 2.66 GHz Intel Core2 Duo E6700 or 3.00 GHz AMD Athlon 64 X2 6000+ or better recommended
- RAM: 2 GB or more (4 GB recommended)
- Video Card: 512 MB DirectX 9.0c-compliant with Shader Model 4.0 or higher.
- Peripherals Supported: Windows-compatible keyboard, mouse.
- Displaying screen that display OS and its applications clearly.
- Cache memory for caching temporarily stored data (Minimum 512 KB).



- Full internet access via a network card.
- Internet connection via operating Wi-Fi, Wi-Max, Fiber or any other internet provider devices that provide a minimum internet connection with 512 Kb/s of speed.

### **4.4.2 Software Requirements**

Any OS working over device with these features:

- Any OS operating over PC, Laptop, smartphone or tablet that is able to access the internet (Windows, Linux, Mac OS, Windows phone, Android, iOS or others).
- Any browser can surf internet and browse websites and web applications with these features:
  - 1) Ability to handle web surfer's requests and present requests' results.
  - 2) Support Hypertext Transfer Protocol (HTTP) Verbs (GET, POST, PUT and DELETE)
  - 3) Support Hypertext Markup Language version 5 (HTML5), Cascading Style Sheet version 3 (CSS3) and other responsive design technologies.
  - 4) Enabling cookies and save temporarily data on the browser cache.

## **4.5 Requirement Analysis**

Requirements analysis is critical to the success or failure of a systems or software project and it is important for determining whether the stated requirements are clear, complete, consistent and unambiguous, and without any apparent conflicts. [12]

As for us, after finishing eliciting and analyzing the requirements of MCloud system, we have to choose a method to record the analyzed requirements and the most effective method is structuring UML Diagrams to help us understand the behavior of the system which will be developed and what actors are interacting with it.

## 4.5.1 Application Perceptive

This system will consist of two parts: web-based application and third-party cloud servers. The web-based application will be used to let MCloud users to browse files and folders of their connected and synced online cloud drives while the third-party cloud servers will manage the connection and allow the syncing process with our MCloud system.

The web-based application will need to communicate through the API of each connected cloud drives in order to provide the mobile application with all the files and folders and keep them synced all the time to let MCloud users have a single access to multiple cloud drives and do the necessary functionalities like sharing, editing and managing their files across the internet.

In addition to that, a database will be used. Both the web-based application and third-party cloud servers APIs will communicate with the database, however in slightly different ways. The web-based application will use the database to store users' accounts and manage linked cloud drives and logins and the access token management while the third-party cloud servers will keep every account synced all time.

## 4.5.2 UML Diagrams

The **Unified Modeling Language (UML)** is a general-purpose, developmental, modeling in the field of software engineering which is intended to provide a standard way to visualize the design of a system. [13]

We prepare UML diagrams to understand MCloud system in better and simple way. Also a single diagram is not enough to cover all aspects of the system. So structures various kinds of diagrams including structural and behavioral diagrams to cover most of the aspects of a system.

### 4.5.2.1 Use Case Diagram

Use Cases are an effective requirements capture technique that makes requirements available for review by avoiding any implementation bias in the requirements. [14]

Use Case itself is an interaction that a User or other System has with the system that is being designed, in order to achieve a goal. The term Actor is used to describe the person or system that has the goal, this term is used to emphasize the fact that any person or system could have the goal. In addition to that they are used to describe the relationships among the functionalities and their internal/external Actors.

#### 4.5.2.2 Use Case List

U1	Login	U13	Share Files	U25	Connect Cloud Drive
U2	Logout	U14	Single Drive Sharing	U26	Sync Data
U3	Favorite Files	U15	Multiple Drives Sharing	U27	Remove Connected Drive
U4	View Favorite Files	U16	Set Password	U28	Send Direct Message
U5	Manage Files & Folders	U17	Get Link	U29	Create an Account
U6	Rename Files & Folders	U18	Reset Password	U30	Browse Website
U7	Download Files	U19	Share on Social Media	U31	View Direct Messages
U8	Add Folders	U20	Authenticate by Phone Number	U32	View Setting
U9	Delete Files & Folders	U21	Upload Files	U33	View Tasks
U10	Preview Images	U22	Search for Files & Folders	U34	Receive Notifications
U11	Show Statistics	U23	View Result	U35	View notifications
U12	Show Drives Profiles	U24	Add Cloud Drive	U36	Copy to Clipboard

Table 4 - 1: Use Case List

### 4.5.2.3 Actors and their Roles

Actor	Description	Goals
Admin	An actor who is a member and responsible for monitoring, managing the users and maintaining processes of the system.	<ul style="list-style-type: none"> <li>• Manage website contents by adding, editing and removing contents.</li> <li>• Send direct messages to users for help and support.</li> <li>• Manage user accounts with any access or knowledge of the users' files.</li> </ul>
User	An actor who is a member and have the ability to link his personal cloud drives with his MCloud account to be able to browse all of the files and do all functionalities needed.	<ul style="list-style-type: none"> <li>• Add and connect cloud drives or discount them from the MCloud account.</li> <li>• Browse files and folders of the connected cloud drives.</li> <li>• Upload files and folders to the connected cloud drives.</li> <li>• Share any file or folder in social media.</li> <li>• View space storage statistics of the connected cloud drives.</li> </ul>
Visitor	A person who can browse the main website landing page and can create new MCloud account or login if he already have an account.	<ul style="list-style-type: none"> <li>• Browse the website and all its content</li> <li>• Create a new account</li> <li>• Login to the Applicant</li> </ul>
«Web API»	External Entity which responsible for creating a connection between MCloud user accounts and the cloud drives servers such as Google Drive, Drobox and OneDrive in order to link and sync files across the web.	<ul style="list-style-type: none"> <li>• Connects the third-party APIs of the cloud servers with MCloud account.</li> <li>• Maintain syncing of the files and folders.</li> </ul>

Table 4 - 2: Actors and their Roles

#### 4.5.2.4 Use Case Priority Levels

Top Level Priority	
<b>U1</b>	Login
<b>U16</b>	Set Password
<b>U17</b>	Get Link
<b>U18</b>	Reset Password
<b>U20</b>	Authenticate by Phone Number
<b>U21</b>	Upload Files
<b>U24</b>	Add Cloud Drive
<b>U25</b>	Connect Cloud Drive
<b>U26</b>	Sync Data
<b>U27</b>	Remove Connected Drive
<b>U29</b>	Create an Account

*Table 4 - 3: Top Level Priority Use Cases*

Intermediate Level Priority	
<b>U3</b>	Favorite Files
<b>U4</b>	View Favorite Files
<b>U5</b>	Manage Files & Folders
<b>U7</b>	Download Files
<b>U8</b>	Add Folders
<b>U9</b>	Delete Files & Folders
<b>U10</b>	Preview Images
<b>U11</b>	Show Statistics
<b>U12</b>	Share Files
<b>U13</b>	Single Drive Sharing
<b>U14</b>	Multiple Drives Sharing
<b>U15</b>	Share on Social Media
<b>U22</b>	Search for Files & Folders
<b>U23</b>	View Result

*Table 4 - 4: Intermediate Level Priority Use Case*

Low Level Priority	
<b>U2</b>	Logout
<b>U6</b>	Send Direct Message
<b>U12</b>	Rename Files & Folders
<b>U28</b>	Browse Website
<b>U30</b>	View Direct Messages
<b>U31</b>	View Setting
<b>U32</b>	View Tasks
<b>U33</b>	Receive Notifications
<b>U34</b>	View notifications
<b>U35</b>	Copy to Clipboard
<b>U36</b>	Show Drives Profiles

*Table 4 - 5: Low Level Priority Use Case*

#### 4.5.2.5 Use Case Diagram



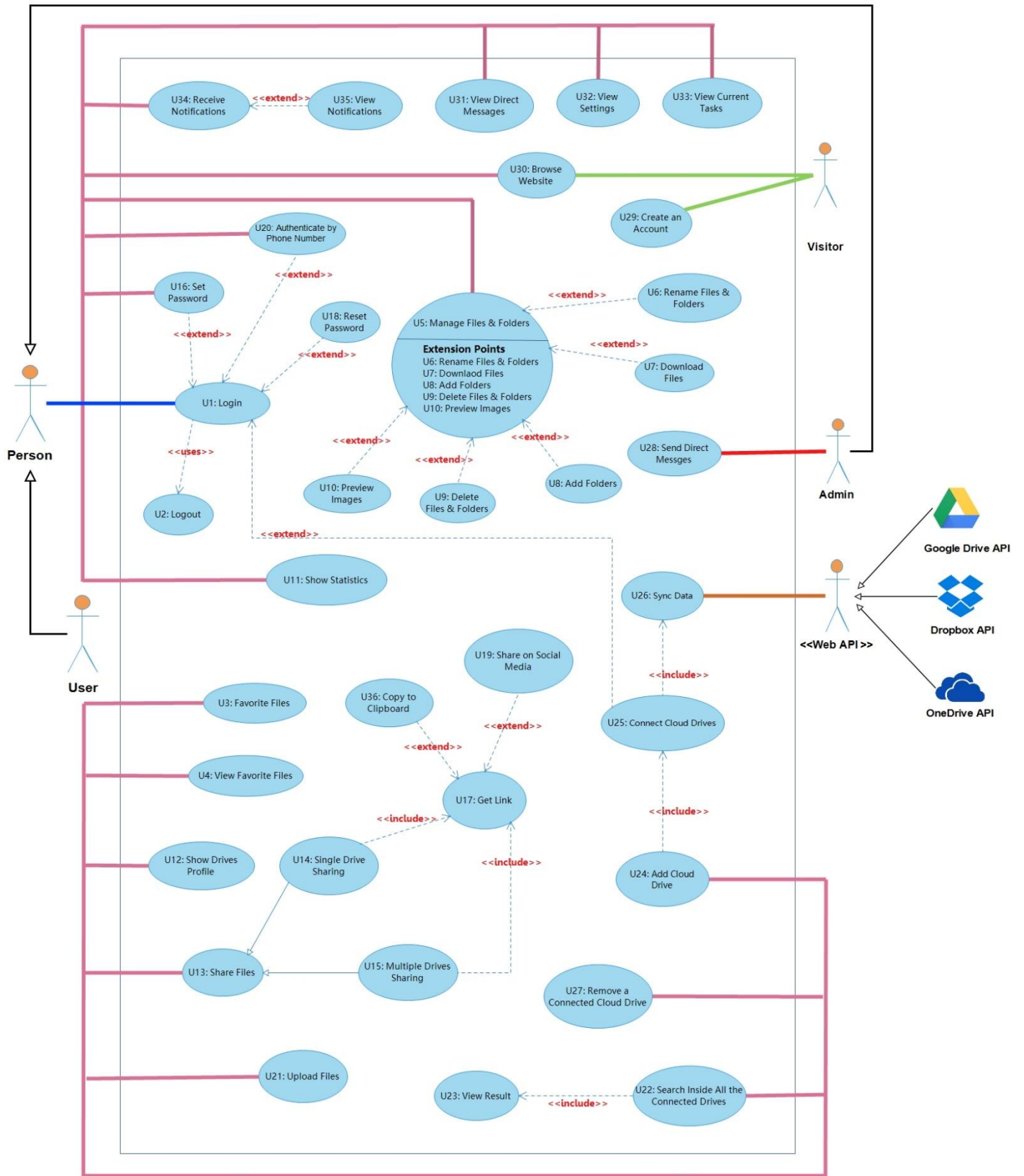


Figure 4 - 1 Use Case Diagram

#### 4.5.2.6 Use Case Details

UC 1 : Login	
Actor	Admin, User
Relation	Uses U2, Extended by U16, U18, U20 and U25
Description	Actor logs in to authenticate his role in the system and be able to use the functionalities of the system.
Preconditions	A valid account has been created for the actor with specific permissions and privileges according to his role.
Post-conditions	The actor can successfully access the system and perform actions appropriate for his role.
Normal Path	<ol style="list-style-type: none"> <li>1) The actor connects to the system.</li> <li>2) The actor chooses one of the social media presented and available to login to the system. Or by entering his username and password.</li> <li>3) The system determines the role of the actor.</li> <li>4) The system displays the main user interface of the application to the actor.</li> </ol>
Abnormal Path	<ul style="list-style-type: none"> <li>• Login failed due to internet connection failure</li> <li>• Actor <b>doesn't</b> have an account.</li> </ul>
Comments	Admin cannot have access to the user's personal files to guarantee the users privacy.

Table 4 - 6: Login Use Case Detailed Description

\*The rest of use cases details tables are addressed in the appendices.

### **4.5.2.7 Activity Diagram**

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram. Activities modeled can be sequential and concurrent. In both cases an activity diagram will have a beginning and an end. [15]

Drawing an activity diagram can help you improve a process. If the diagram of an existing process proves to be very complex, you can consider how the process could be simplified

#### **4.5.2.7.1 Activity Diagram: Set Password**

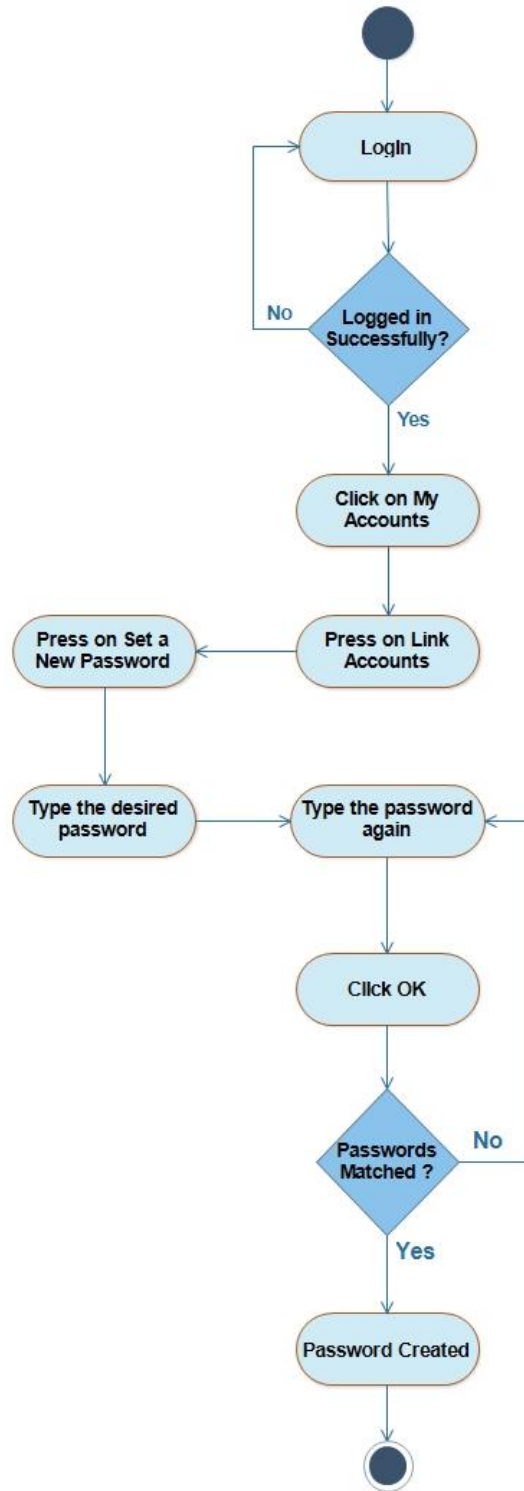


Figure 4 - 2: Set Password Activity Diagram

\*The rest of activity diagrams are addressed in the appendices.

#### **4.5.2.8 Sequence Diagram**

Sequence diagrams are a kind of interaction diagram, because they describe how—and in what order—a group of objects works together. These diagrams are used by software developers and businesspeople alike to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

The sequence diagram is used primarily to show the interactions between objects in the sequential order that those interactions occur. Typically they are associated with use case realizations in the Logical View of the system under development. [16]

### 4.5.2.8.1 Sequence Diagram: OAuth 2.0 and API Calls

This diagram shows how we used OAuth2-based APIs to provide authorization flows for MCloud system by authorizing third-party applications to access the user accounts on an HTTP service while obtaining limited access.

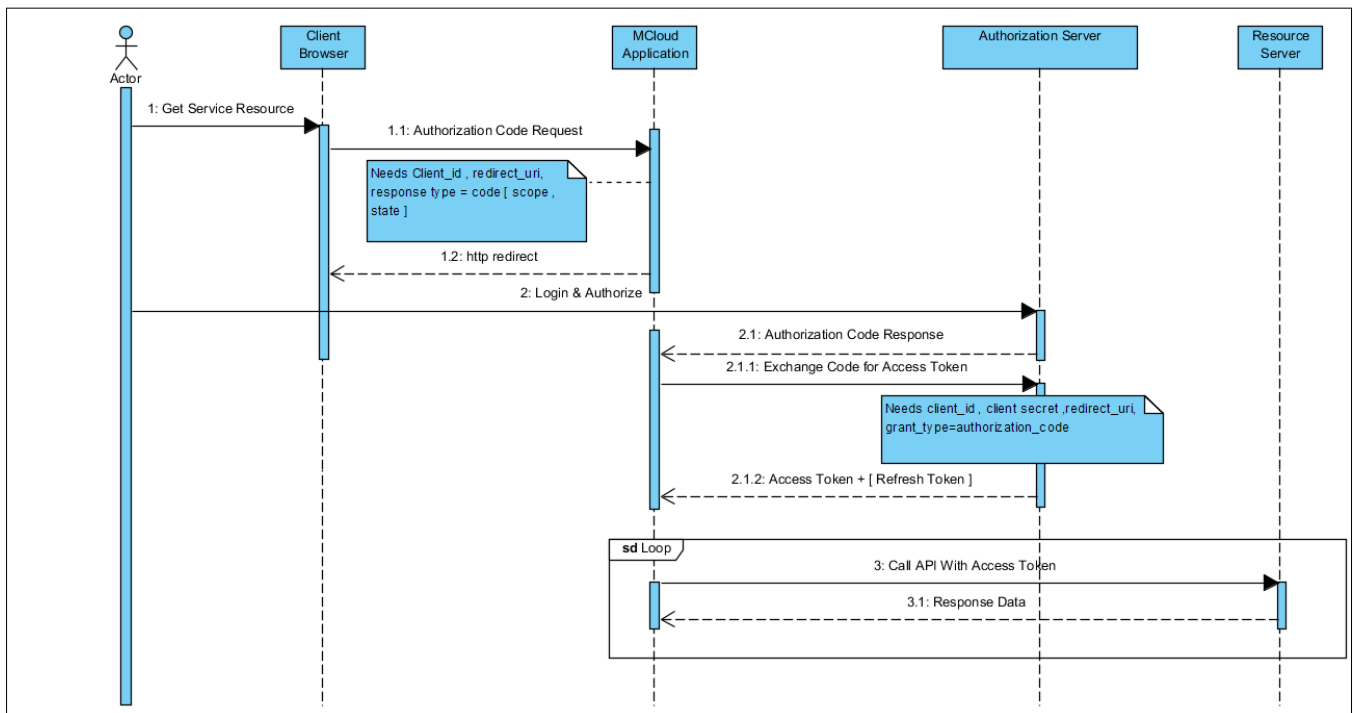


Figure 4 - 3: OAuth 2.0 and API Calls Sequence Diagram

#### 4.5.2.8.2 Sequence Diagram: MVC

The diagram shows how we adopted MVC as a web application framework and also the way it implements the model-view-controller (MVC) design pattern in MCloud system.

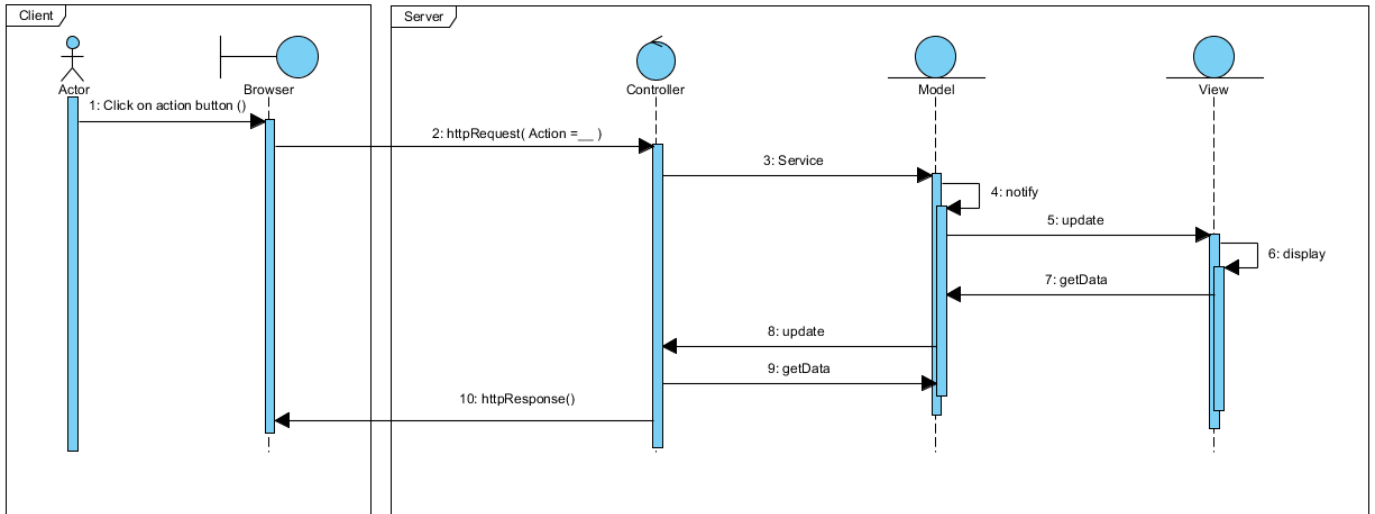


Figure 4 - 4: MVC Sequence Diagram

## 4.6 Conclusion

In this chapter we cover all the requirements process with all of its stage which begin with requirements gathering and elicitation which was accomplished after many brainstorming sessions and research. Then we started with the next stage which is analyzing and breaking down the requirements into something that can make the system easy to understand by visualization tools and mainly UML modeling and its diagram. In addition to that it should also clear any conflicts of any requirements. After finishing the analysis stage we finalized the process with recording the requirements and the analysis that resulted in a clear and consistent way by listing the functional requirements then the nonfunctional requirements then the system requirements as whole. The recording of the analysis was provided by the UML diagrams which constructed for MCloud system such as use case diagram, activity diagram and finally the sequence diagram.



## CHAPTER 5: Software System Architecture

### Content

- 5.1 Introduction
- 5.2 System Architecture Overview
- 5.3 Model View Controller (MVC)
- 5.4 Three-Tier Architecture
- 5.5 The MVC implemented in a three-tier architecture
- 5.6 Design Patterns
- 5.7 Conclusion



# 5 CHAPTER 5: SOFTWARE SYSTEM ARCHITECTURE

## 5.1 Introduction

This chapter introduces the MCloud system’s architecture that describes a system and its underlying structures, which contains software components, its external properties and the relationships among them. Also this chapter includes the design pattern used to solve implementation, integration and deployment of the system.

## 5.2 System Architecture Overview

Client/server architecture is a computing model in which the server hosts, delivers and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or Internet connection. This system shares computing resources.

Client/server architecture may also be referred to as a networking computing model because all the requests and services are delivered over a network. [17]

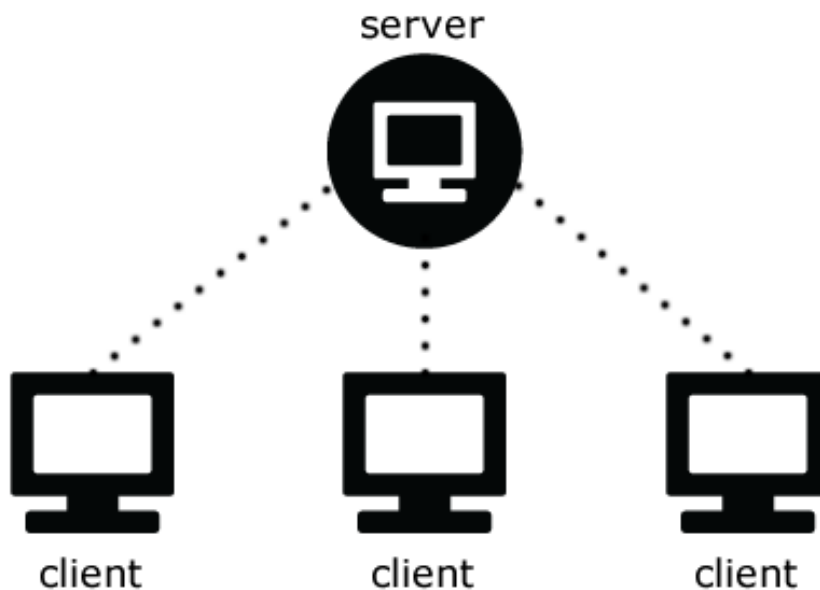


Figure 6- 1: Client/Server Architecture

## 5.3 Model View Controller (MVC)

The increase of web and mobile software development can get one thinking about how to best design and architect an application for these platforms. The two proven approaches that find themselves into this discussion are the Model View Controller (MVC) pattern and the three-tier architecture.

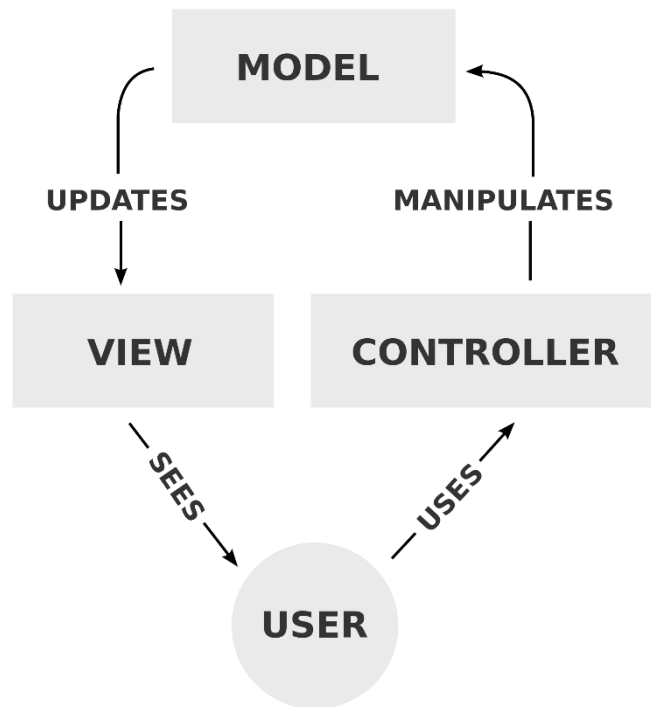


Figure 6- 3: MVC

The MVC pattern provides very well when building applications that are to be consumed by many different platforms (it also provides well when you are building for a single platform). When targeting platforms that are mobile devices, tablets and different browsers on different operating systems the MVC pattern is a proven approach. The MVC pattern separates different parts of rendering the User Interface (UI) into modules, so the code for rendering the actual interface is separated from the code that manages the data which is also separated from the code that handles the user events. [18]

## 5.4 Three-Tier Architecture

The three-tier architecture separates the deployment of software components into three logical layers. These layers include;

- Presentation Tier - responsible for rendering the User Interface
- Business Tier (or Logic Tier) - responsible for processing the business rules or logic
- Data Tier - responsible for interacting with the data storage system

This separation is for a number of reasons which include maintainability, reuse and deployment. When software is modularized and separated into these three-tiers, the modules can be deployed to different server infrastructures for security, scalability and performance. And the deployment approach can change as the need requires. The modules are also easier to maintain and reuse when they are each built for a discrete purpose.

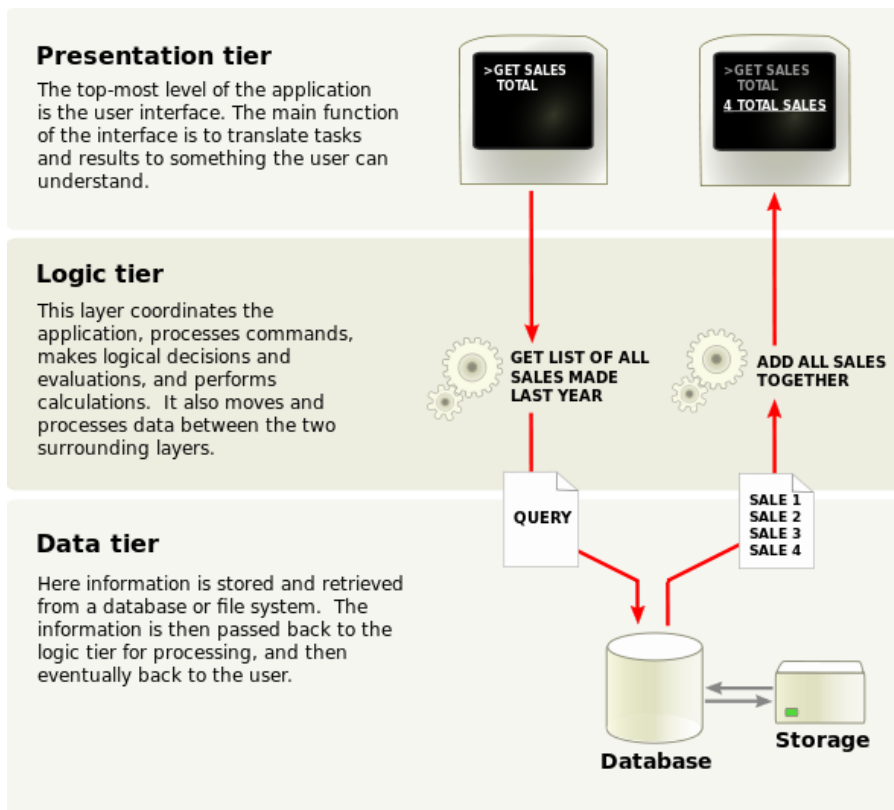


Figure 6- 4: Three-Tier Architecture

## 5.5 The MVC implemented in a three-tier architecture

When the MVC and three-tier approaches are brought together the View and Controller are considered the presentation tier, the Model exists in the business tier (and has access to many business and data tier modules). To a certain extent the Model could span both the business and data tiers. It is this author's view that the Model would not exist in the layer of the data tier that is specifically Data Access Layer (DAL) code including SQL or code optimized for DAL within a cached environment.

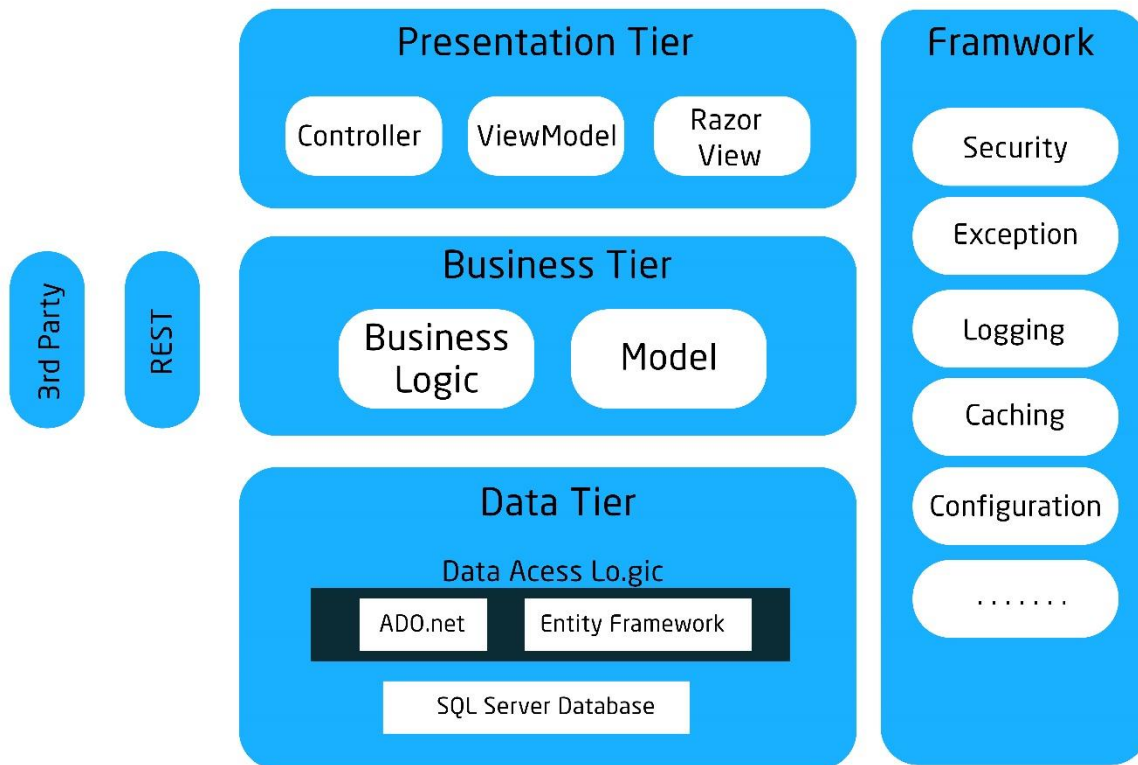


Figure 5 - 1: MVC in Three-Tier Architecture

### 5.5.1 Why the hybrid?

The MVC pattern does not describe how to best design the data access and how to manage the complexity that can occur within the business and data tiers. As the internet has become more open and further entrenched in business processes, the demand to access the business tier without the presentation tier has increased. The creation of Business Tier modules allows for other approaches (such as REST) to access business logic from other sources, enabling business-to-business interactions and innovations.

Building websites can be complicated, particularly when the site engages users and have a lot of content that can be targeted toward and created by users. Why the hybrid of MVC and three-tiers? Two main reasons; First, because the MVC pattern does a great job of simplifying and managing the development of user interfaces over multiple devices and browsers, but it doesn't do a good job of defining how to build scalable server infrastructures. Second, The three-tier architecture does a great job of simplifying and managing the development of high performing, scalable, extensible and maintainable server infrastructures, but doesn't do a good job of defining how to build user interfaces across multiple devices and browsers. With a MVC three-tier hybrid you can utilize the best of both approaches without compromising either. [19]

## 5.5.2 Presentation Layer

For most business applications, a form metaphor is used to structure the presentation layer. The application consists of a series of forms (pages) with which the user interacts. Each form contains a number of fields that display output from lower layers and collect user input.

Two types of components that implement forms-based user interfaces are:

### 5.5.2.1 User Interface Components

For rich-client applications, this pattern uses UI components from the *System.Windows.Forms* namespace of the .NET Framework. For Web applications, this pattern uses ASP.NET components. If the standard .NET components do not meet your needs, .NET supports subclassing of the standard UI components, as well as plugging your own custom components into the framework.

### 5.5.2.2 User Interface Process Components

Complex user interfaces often require many highly complex forms. To increase reusability, maintainability, and extensibility, you can create a separate user interface process (UIP) component to encapsulate dependencies between forms and the logic associated with navigating between them. You can apply the same concept to the dependencies, validation, and navigation between components of a single form. These UIP components are typically custom components that are based on design patterns such as *Front Controller*, *Application Controller* and *Mediator*.

The interaction between UI and UIP components often follows the *Model-View-Controller* or *Presentation-Abstraction-Controller* pattern.

### **5.5.3 Business Layer**

Large enterprise applications are often structured around the concepts of business processes and business components. These concepts are addressed through a number of components, entities, agents, and interfaces in the business layer.

#### **5.5.3.1 Business Components**

Business component defined as follows: [20]

The software implementation of an autonomous business concept or business process. It consists of all the software artifacts necessary to represent, implement, and deploy a given business concept as an autonomous, reusable element of a larger distributed information system.

Business components are the software realization of business concepts. They are the primary units of design, implementation, deployment, maintenance, and management for the life cycle of the business application. Business components encapsulate the business logic, also called business rules. These rules constrain the behavior of a business concept to match the needs of a particular company. For example, the business rule that determines whether a given customer is approved for a line of credit may be encapsulated in the customer business component for small solutions. For larger solutions, it is likely that all credit-related business logic is encapsulated in a separate credit component.

### 5.5.3.2 RESTful API in a Business logic tier

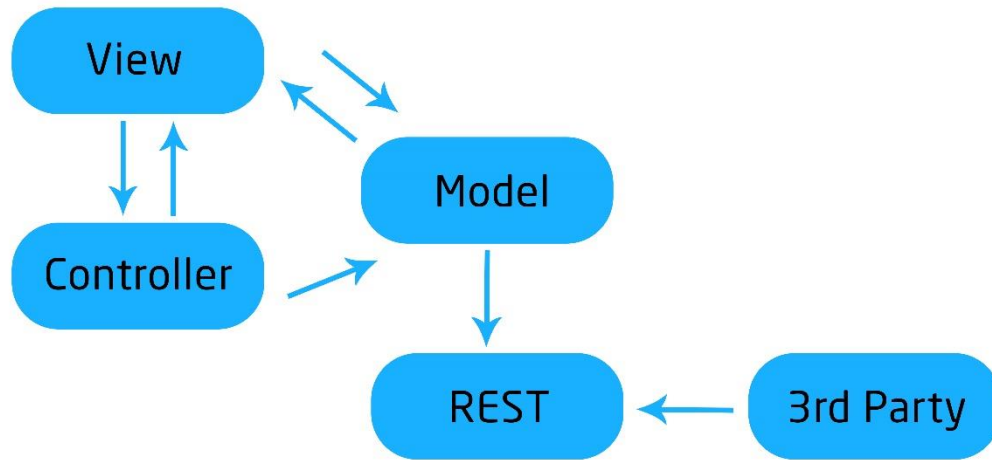


Figure 5 - 2: RESTful API

REST stands for Representational State Transfer. (It is sometimes spelled "REST".) It relies on a stateless, client-server, cacheable communications protocol -- and in virtually all cases, the HTTP protocol is used.

REST is an architecture style for designing networked applications. The idea is that, rather than using complex mechanisms such as CORBA, RPC or SOAP to connect between machines, simple HTTP is used to make calls between machines.

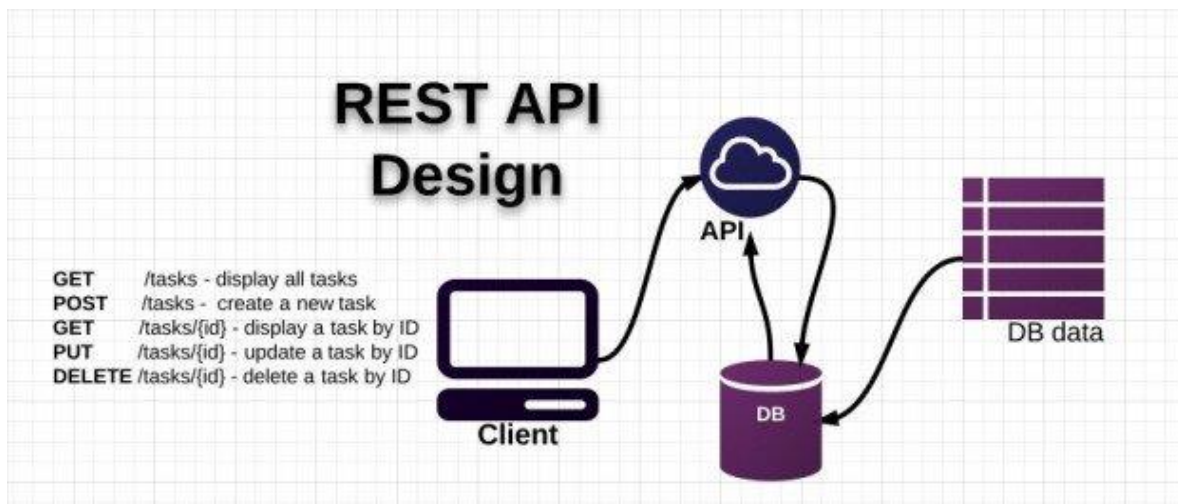


Figure 6- 5: REST API Design



In many ways, the World Wide Web itself, based on HTTP, can be viewed as a REST-based architecture. RESTful applications use HTTP requests to post data (create and/or update), read data (e.g., make queries), and delete data. Thus, REST uses HTTP for all four CRUD (Create/Read/Update/Delete) operations.

REST is a lightweight alternative to mechanisms like RPC (Remote Procedure Calls) and Web Services (SOAP, WSDL, et al.). Later, we will see how much more simple REST is.

Despite being simple, REST is fully-featured; there's basically nothing you can do in Web Services that can't be done with a RESTful architecture. REST is not a "standard". There will never be a W3C recommendation for REST, for example. And while there are REST programming frameworks, working with REST is so simple that you can often "roll your own" with standard library features in languages like Perl, Java, or C#. [21]

## **5.5.4 Data Layer**

Data access components isolates the business layer from the details of the specific data storage solution.

### **5.5.4.1 Data Access Components**

- Minimizes the impact of a change in database provider.
- Minimizes the impact of a change in data representation (for example, a change in database schema).
- Encapsulates all code that manipulates a particular data item in one place. This greatly simplifies testing and maintenance.

ADO.NET can be used directly as the data access components for simple applications. More complex applications may benefit from developing a set of classes over ADO.NET that help you to manage the complexities of object-relational mapping. [22]

## 5.6 Design Patterns

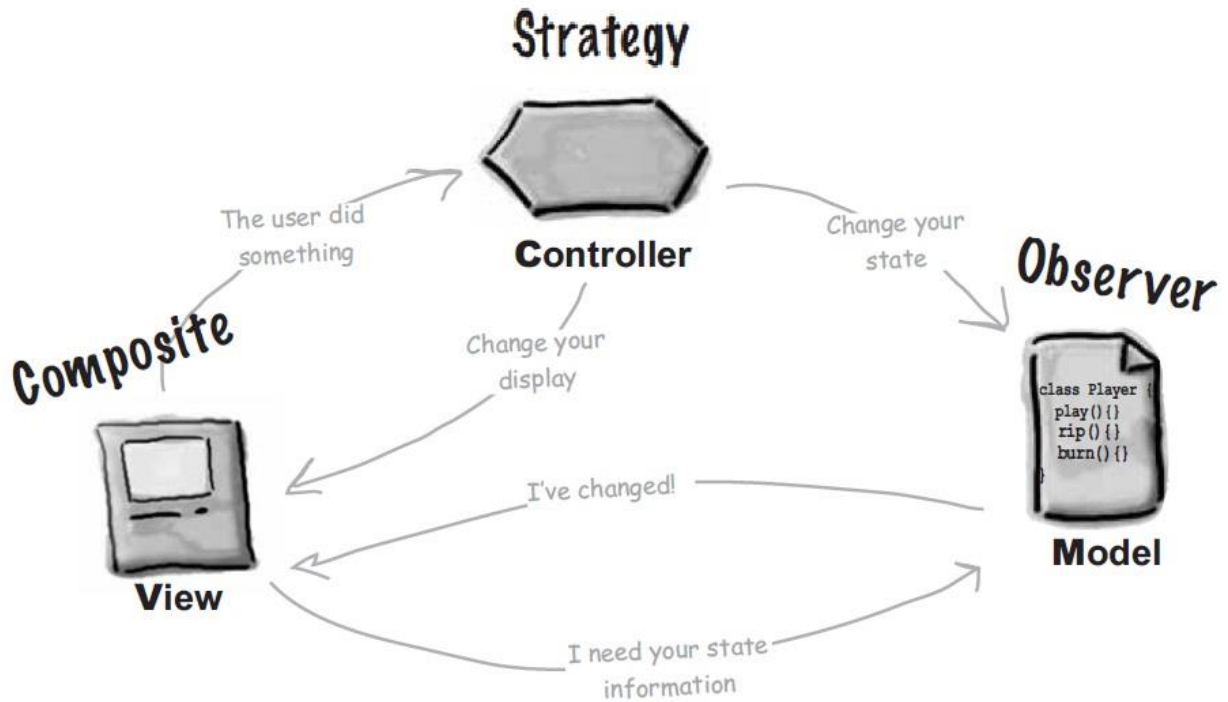


Figure 5 - 3: MVC as a Compound Design Pattern

### 5.6.1 Strategy Pattern:

The View and Controller implement the basic classic Strategy Pattern: the View is an object that is configured with a strategy. The controller provides the strategy. The view is concerned only with the visual aspects of the application, and delegates to the controller for any decisions about the interface behavior. Using the strategy pattern also keeps the view decoupled from the model because it is the controller that is responsible for interacting with model to carry out user requests. The view knows nothing about how this gets done. [23]

### **5.6.2 Composite Pattern:**

The display consists of a nested set of windows, panels, buttons, and text labels and so on. Each display component is a composite (like a window) or a leaf (like a button), it only has to tell the top view component, and composite takes care of the rest.

### **5.6.3 Observer Pattern:**

The model implements the observer pattern to keep interested object updated when state changes occur. Using the observer pattern keeps the model completely independent of the views and controllers. It allows us to use different views with the same model, or even use multiple views at once.

## **5.7 Conclusion**

This chapter summarized the importance of choosing the appropriate components for building the structures of MCloud system in order to develop a system with the proper architectural choice and coherent set of structures that is needed to build a well-structured and effective software system.



## Chapter 6 System Design and Specifications

### Content

- 6.1 Introduction
- 6.2 Technology Choices
- 6.3 Choice of Web Design Technologie
- 6.4 Development Tools
- 6.5 Class Diagrams
- 6.6 MCloud Class Diagram Categories
- 6.7 Database Model
- 6.8 Conclusion

# 6 CHAPTER 6 SYSTEM DESIGN AND SPECIFICATIONS

## 6.1 Introduction

This section show what are the technology choices that are used to build the Plan B system and the reasons for those choices.

## 6.2 Technology Choices:

### 6.2.1 ASP.NET MVC

The Development Language and Framework used in MCloud system.



*Figure 6 - 1: ASP.NET MVC Logo*

ASP.NET is a free web framework for building great web sites and web applications using HTML, CSS, and JavaScript. You can also create Web APIs and mobile sites, and you can use real-time technologies like Web Sockets and more.

ASP.NET Web Pages and the Razor syntax provide a fast, approachable, and lightweight way to combine server code with HTML to create dynamic web content. Connect to databases, add video, link to social networking sites, and include many more features that help you create beautiful sites that conform to the latest web standards.

ASP.NET MVC gives you a powerful, patterns-based way to build dynamic websites that enables a clean separation of concerns and that gives you full control over markup for enjoyable, agile development. ASP.NET MVC includes many features that enable fast, TDD-friendly development for creating sophisticated applications that use the latest web standards. [24]

### **6.2.1.1 Advantages of an MVC-Based Web Application**

The ASP.NET MVC framework offers the following advantages:

- It makes it easier to manage complexity by dividing an application into the model, the view, and the controller.
- It does not use view state or server-based forms. This makes the MVC framework ideal for developers who want full control over the behavior of an application.
- It uses a Front Controller pattern that processes Web application requests through a single controller. This enables you to design an application that supports a rich routing infrastructure. For more information, see Front Controller.
- It provides better support for test-driven development (TDD).
- It works well for Web applications that are supported by large teams of developers and for Web designers who need a high degree of control over the application behavior.

### **6.2.1.2 Features of the ASP.NET MVC Framework [25]**

- Separation of application tasks (input logic, business logic, and UI logic), testability, and test-driven development (TDD). All core contracts in the MVC framework are interface-based and can be tested by using mock objects, which are simulated objects that imitate the behavior of actual objects in the application. You can unit-test the application without having to run the controllers in an ASP.NET process, which makes unit testing fast and flexible. You can use any unit-testing framework that is compatible with the .NET Framework.
- An extensible and pluggable framework. The components of the ASP.NET MVC framework are designed so that they can be easily replaced or customized. You can plug in your own view engine, URL routing policy, action-method parameter serialization, and other components. The ASP.NET MVC framework also supports

the use of Dependency Injection (DI) and Inversion of Control (IOC) container models. DI enables you to inject objects into a class, instead of relying on the class to create the object itself. IOC specifies that if an object requires another object, the first objects should get the second object from an outside source such as a configuration file. This makes testing easier.

- Extensive support for ASP.NET routing, which is a powerful URL-mapping component that lets you build applications that have comprehensible and searchable URLs. URLs do not have to include file-name extensions, and are designed to support URL naming patterns that work well for search engine optimization (SEO) and representational state transfer (REST) addressing.
- Support for using the markup in existing ASP.NET page (.aspx files), user control (.ascx files), and master page (.master files) markup files as view templates. You can use existing ASP.NET features with the ASP.NET MVC framework, such as nested master pages, in-line expressions (<%= %>), declarative server controls, templates, data-binding, localization, and so on.
- Support for existing ASP.NET features. ASP.NET MVC lets you use features such as forms authentication and Windows authentication, URL authorization, membership and roles, output and data caching, session and profile state management, health monitoring, the configuration system, and the provider architecture.

## 6.2.2 ADO.NET

The Choice of Database Management System (DBMS) used for MCloud System.



*Figure 6 - 2: ADO.NET Logo*

The technology which was used to connect to a relational database in ASP.NET code that runs on a server is ADO.NET. ADO.NET communicates with a database management system (DBMS) such as SQL Server.

ADO.NET is a set of classes that expose data access services for .NET Framework programmers. ADO.NET provides a rich set of components for creating distributed, data-sharing applications. It is an integral part of the .NET Framework, providing access to relational, XML, and application data. ADO.NET supports a variety of development needs, including the creation of front-end database clients and middle-tier business objects used by applications, tools, languages, or Internet browsers.

The ADO.NET Entity Framework is the main ORM that Microsoft provides for the .NET Framework

To read or update data, you can use ADO.NET directly by using classes such as SqlCommand, SqlDataReader, SqlDataAdapter, and DataSet. The recommended alternative is to let an object-relational mapper (ORM) framework such as the Entity Framework handle the low-level code that interfaces with an ADO.NET data provider. If you use ADO.NET directly, you have to manually write and execute SQL queries. You also have to write code that converts data from the database's format into objects, properties, and collections that you can work with in code. [26]

### 6.2.3 SQL Server

Some reasons for choosing SQL Server as a relational database for an ASP.NET web application:



*Figure 6 - 3: SQL Server Logo*



- SQL Server is supported by Microsoft.
- SQL Server is integrated with other Microsoft data access technologies such as the Entity Framework. See Object-Relational Mappers later in this topic.
- Visual Studio provides built-in tools for working with SQL Server. SQL Server Data Tools (SSDT), enables you to create databases, manipulate schema and data, generate and run scripts, debug, and deploy databases and database updates.
- Visual Studio includes web deployment features that are designed to facilitate deploying SQL Server databases along with web projects
- The Visual Studio web project templates use SQL Server for the ASP.NET membership database by default.
- Nearly all ASP.NET documentation on the MSDN web site, ASP.NET web sites, and in blog posts uses SQL Server. Relatively little documentation that would help you get started and resolve difficulties is available for other databases

### 6.3 Choice of Web Design Technologies:



Figure 6 - 5: Front End Technologies



Figure 6 - 4: Ajax Technology


- HTML5: is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and current version of the HTML standard. [27]
- JavaScript is the programming language of HTML and the Web.
- CSS is a stylesheet language that describes the presentation of an HTML (or XML) document also describes how elements must be rendered on screen, on paper, or in


other media. [28]


- jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript. [29]
- AJAX stands for Asynchronous JavaScript and XML. In a nutshell, it is the use of the XMLHttpRequest object to communicate with server-side scripts. It can send as well as receive information in a variety of formats, including JSON, XML, HTML, and even text files. [30]


## 6.4 Development Tools

The Choice of Software Development Tools and Computer-Aided Software Engineering (CASE) that are used to build the MCloud system:


**Edraw Max:**  is an all-in-one diagram software that makes it simple to create professional-looking flowcharts, organizational charts, network diagrams, business presentations, building plans, mind maps, science illustration, fashion designs, UML diagrams, workflows, program structures, web design diagrams, electrical engineering diagrams, directional maps, database diagrams and more. [31]

**Visual Studio:**  A free, fully-featured, and extensible IDE for creating modern applications for Windows, Android, and iOS, as well as web applications and cloud services. [32]

**SQL Server 2014:**  As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network (including the Internet). [33]

**IIS:**  Internet Information Server is an extensible web server created by Microsoft for use with Windows NT family. IIS supports HTTP, HTTPS, FTP, FTPS, SMTP and NNTP. [34]

**Notepad++:** is a free source code editor and Notepad replacement that supports several languages. Running in the MS Windows environment, its use is governed by GPL License. [35]

**Adobe Photoshop CS6:**  Software redefines digital imaging with powerful new photography tools, breakthrough capabilities for complex image selections, realistic painting, and intelligent retouching and design website. [36]

## 6.5 Class Diagrams

A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity. Class diagrams are useful in all forms of object-oriented programming (OOP). The concept is several years old but has been refined as OOP modeling paradigms have evolved.

## 6.6 MCloud Class Diagram Categories

MCloud system is implementing the MVC design pattern as the core development approach, with that the process of modeling UML class diagrams for a web-based application is much detailed so it had to be categorized and classified into groups or categories which will present all the class diagrams in MCloud system.

### 6.6.1 Main Class Diagrams

The main class diagram of MCloud system as a whole with all of its components.

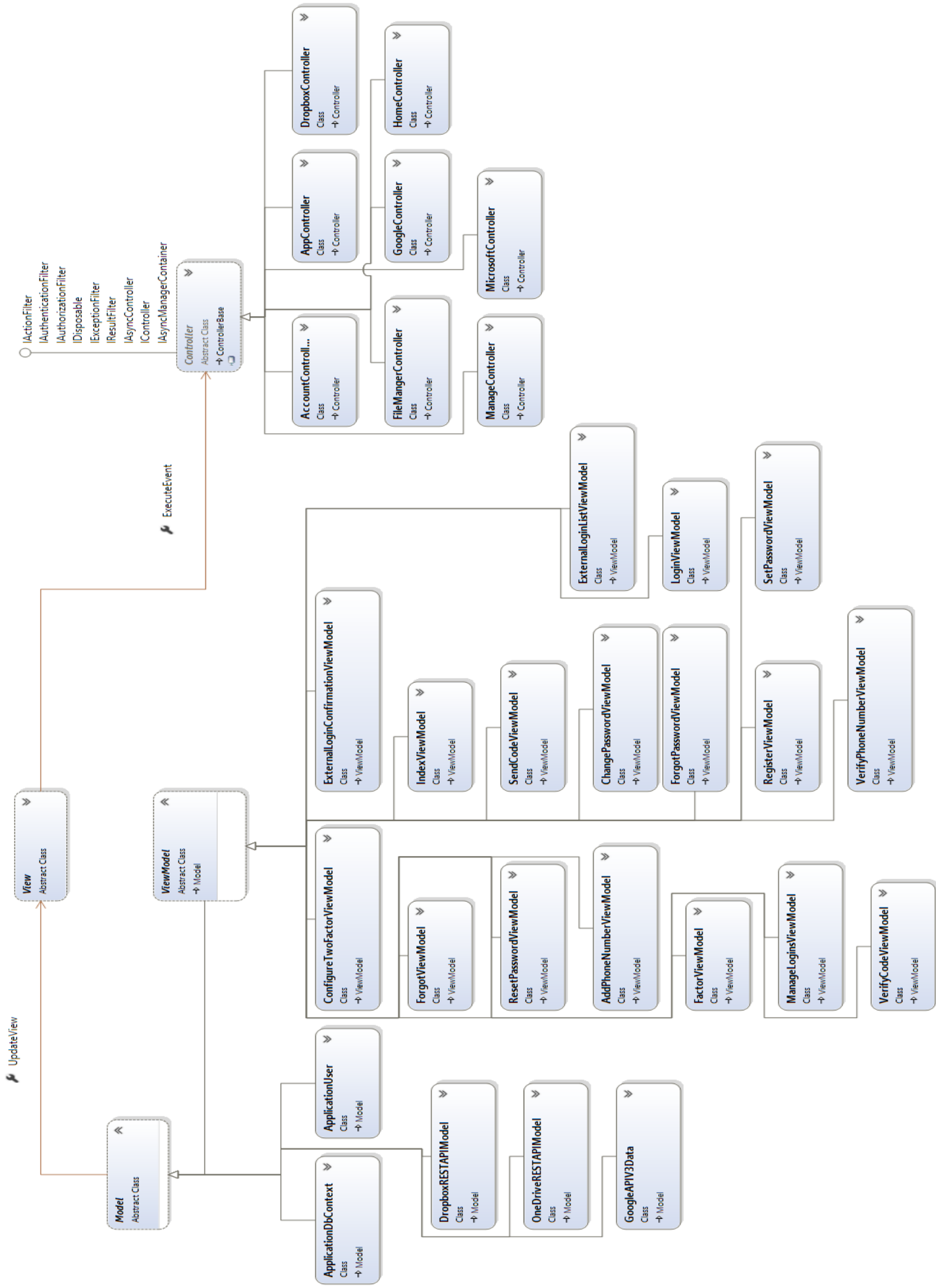


Figure 6 - 6: Main Class Diagram

## 6.6.2 Models

The class diagram of the system model which manages the data, logic and rules of the application.

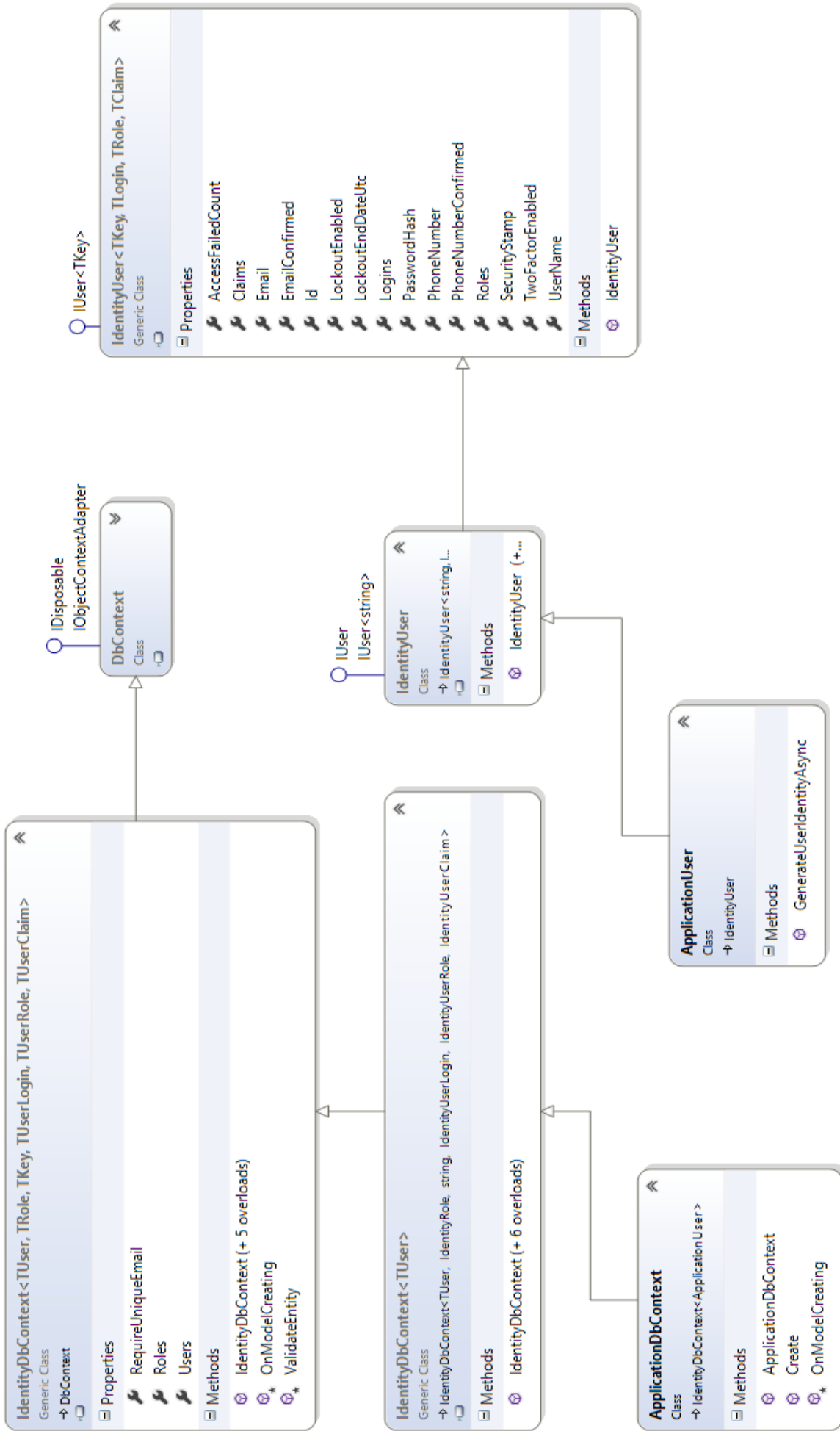


Figure 6 - 7: Models Class Diagram

### 6.6.3 Controller

The class diagrams of the system controllers which handles user interaction with the application.

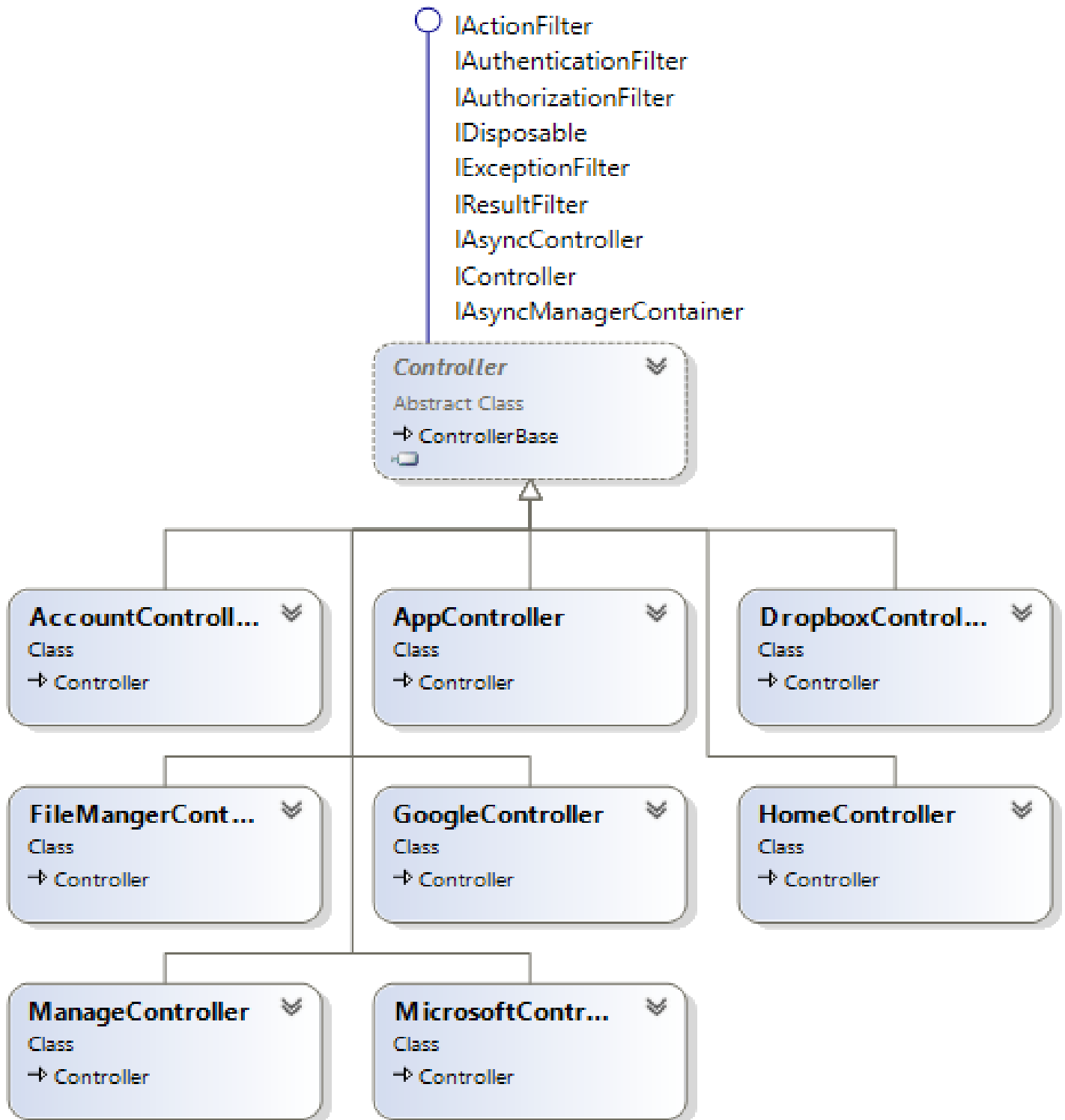


Figure 6 - 8: Controllers Classes Package

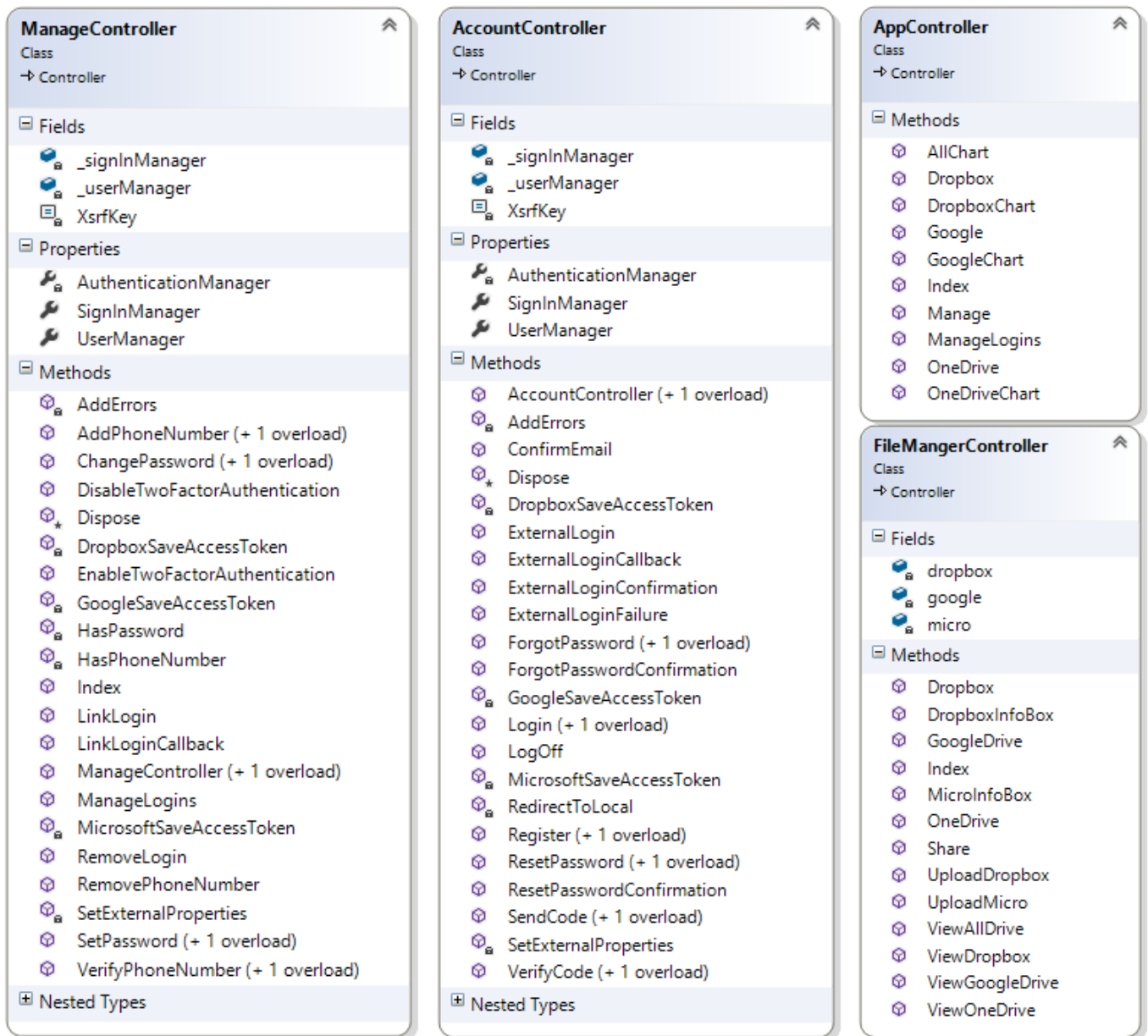


Figure 6 - 9: Account and App Controllers



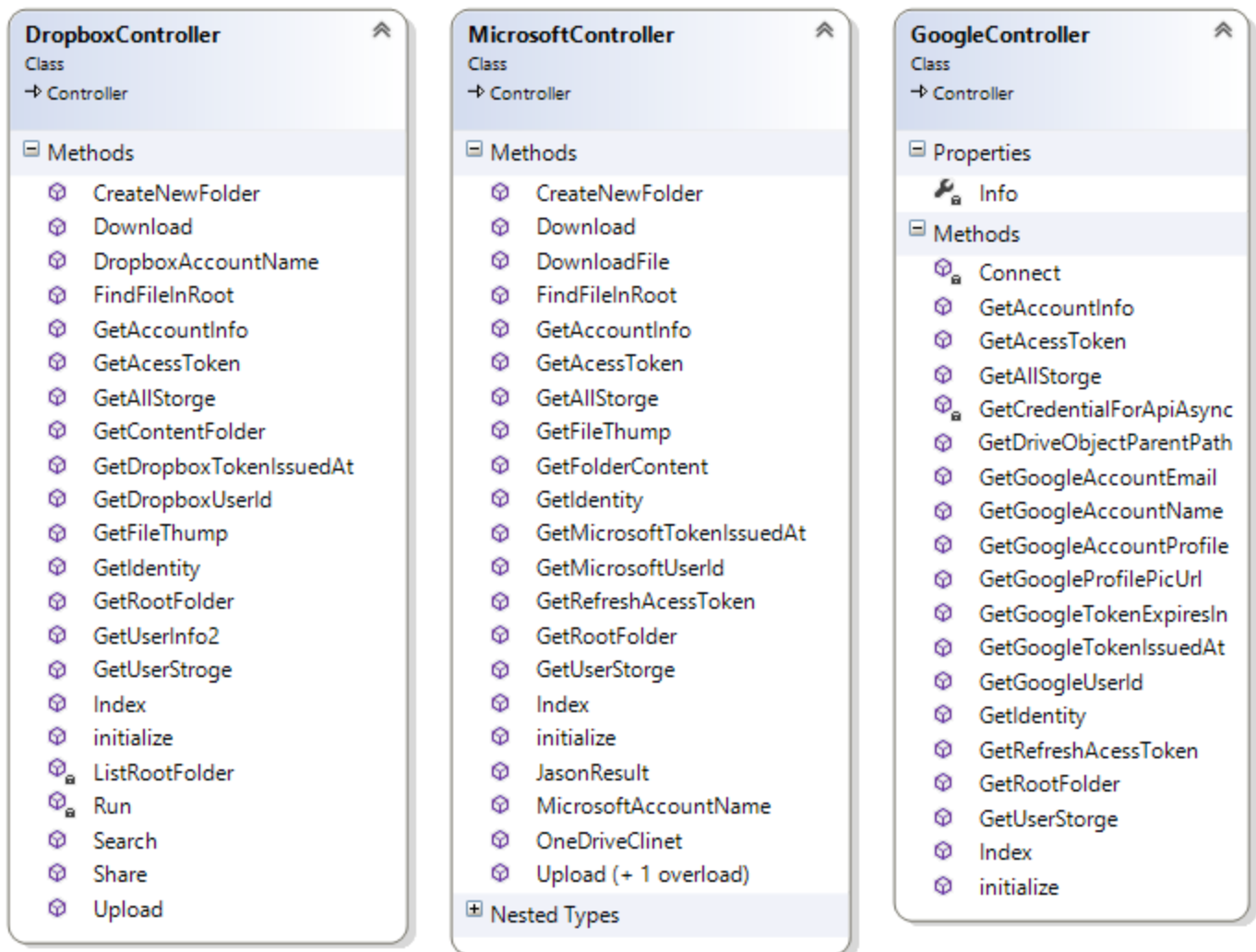


Figure 6 - 10: Third Party Controllers

\*The rest of class diagrams including the model of the third-party application, in addition to the AppStart and ASPSMSX2Spap class diagrams are all addressed in the appendices.

## 6.7 Database Model

A database model is a type of data model that determines the logical structure of a database and fundamentally determines in which manner data can be stored, organized, and manipulated.

From the technological perspective, the choice is usually between pure Object-Oriented, Object-Relational hybrids, pure Relational and custom solutions based on open or proprietary file formats (eg. XML, OLE structured storage). [37]

Database Model in the MCloud system Model is Object-Relational hybrids which is mapped to the class diagram, each class in class diagram is mapped to a table in database model and each attribute is mapped to a column and objects in classes in the code are the rows that contains records in tables.

### 6.7.1 Physical ERD Diagram

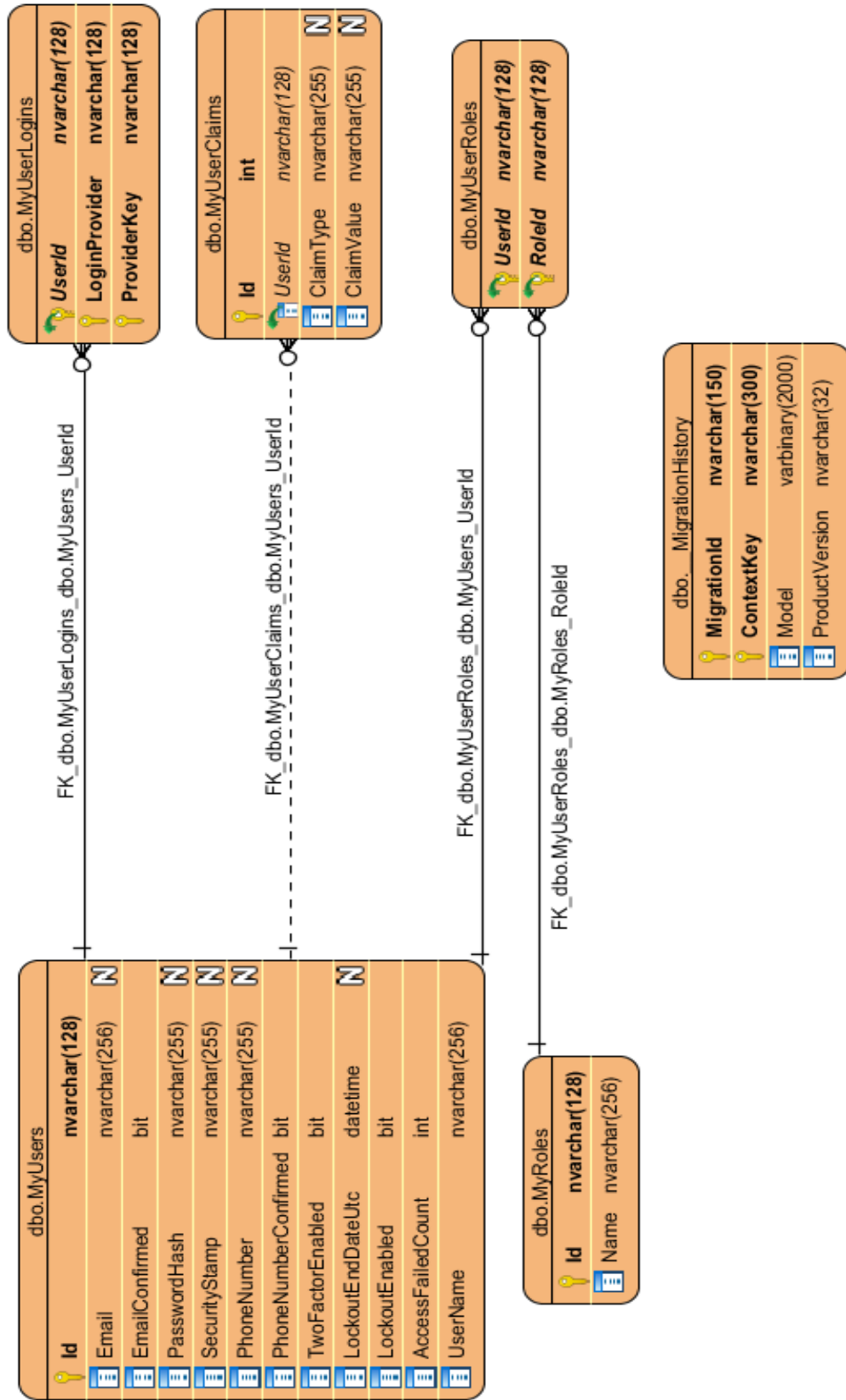


Figure 6 - 11: Physical ERD Diagram

## 6.7.2 SQL Diagram

The Database Designer in SQL Server 2014 is a visual tool that allows to design and visualize a databases. Using Database Designer allows to create, edit, or delete tables, columns, keys, indexes, relationships, and constraints. To visualize a database, it enables to create one or more diagrams illustrating some or all of the tables, columns, keys, and relationships in it.

We used this tool to create MCloud database with the essential tables and all of its relationships.

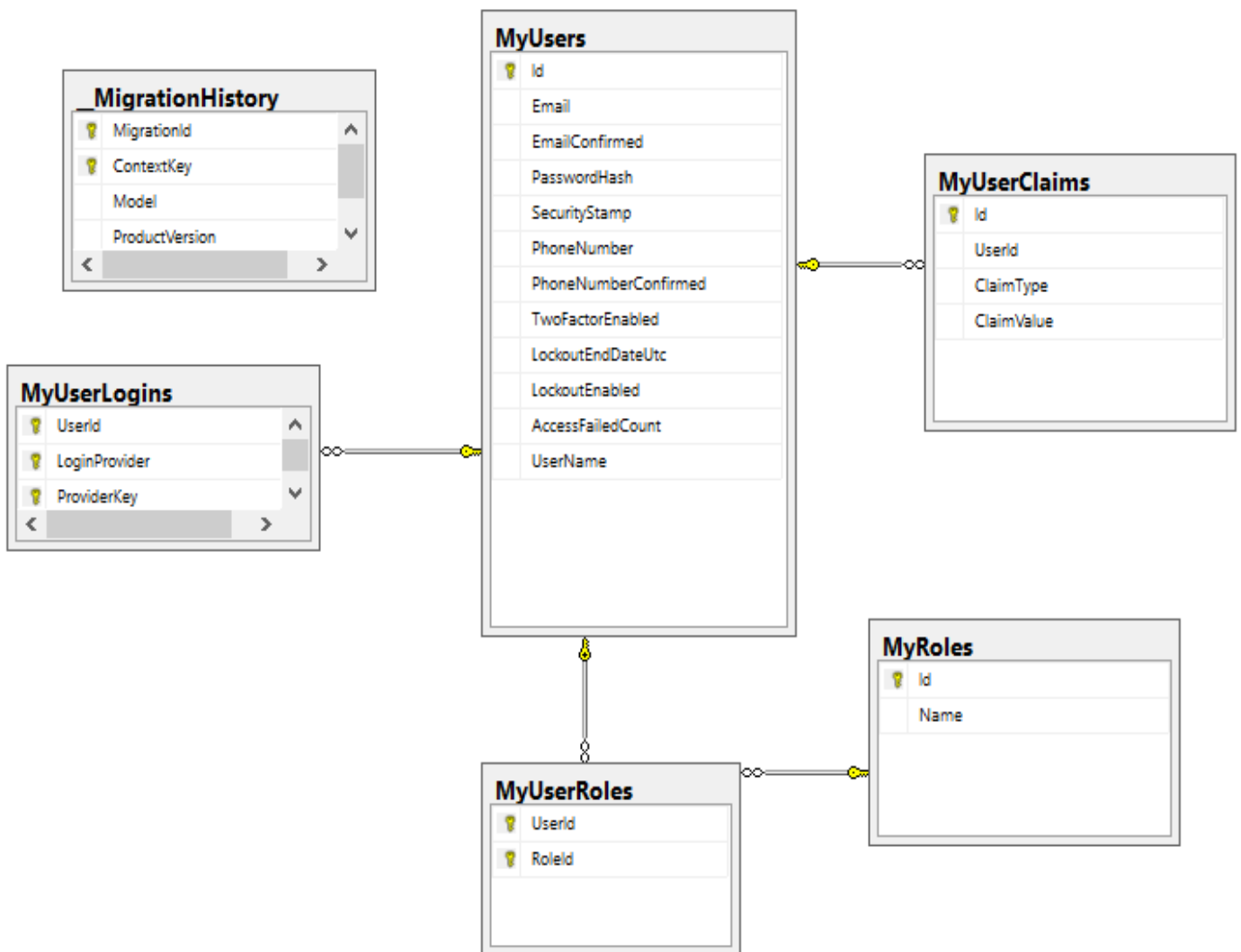


Figure 6 - 12: SQL Diagram

### 6.7.2.1 Database Tables Detailed Description:


MyUsers			
	Column Name	Data Type	Allow Nulls
	Id	nvarchar(128)	<input type="checkbox"/>
	Email	nvarchar(256)	<input checked="" type="checkbox"/>
	EmailConfirmed	bit	<input type="checkbox"/>
	PasswordHash	nvarchar(MAX)	<input checked="" type="checkbox"/>
	SecurityStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>
	PhoneNumber	nvarchar(MAX)	<input checked="" type="checkbox"/>
	PhoneNumberConfirmed	bit	<input type="checkbox"/>
	TwoFactorEnabled	bit	<input type="checkbox"/>
	LockoutEndDateUtc	datetime	<input checked="" type="checkbox"/>
	LockoutEnabled	bit	<input type="checkbox"/>
	AccessFailedCount	int	<input type="checkbox"/>
	UserName	nvarchar(256)	<input type="checkbox"/>
			<input type="checkbox"/>

Figure 6 - 13: MyUsers Table Detailed Descriptions

\*The rest of database description tables are addressed in the appendices.

## **6.8 Conclusion**

This chapter begin with mentioning all the technology and the necessary development programs and tools used to assist building and design MCloud system. Furthermore it addressed the UML class diagrams and the categories which were classified to present the diagrams with a well-organized and detailed approach. Finishing by introducing the database model and the approach which was taken in order to design the system logical and relational databases by starting with the physical ERD diagram for MCloud system and then using SQL Server to design the SQL diagram for the system.



# Chapter 7: Implementation

## Content

- 7.1 Introduction
- 7.2 Implementation Plan
- 7.3 Website Implementation
- 7.4 Application Implementation
- 7.5 Conclusion

# 7 CHAPTER 7: IMPLEMENTATION

## 7.1 Introduction

This chapter focuses on the construction and implementation of MCloud system and it begins upon approval of the designs. This process includes configuring the software and implementing any customized functionality per design documentation. After refining the business processes, development of the first release begins. This is where the business requirements are considered by the team and start the work of programming and configuring the meet the business needs.

There are series of steps considered while the development stage started especially after modeling and designing the database, these steps respectively ordered as the following:

- Wireframes of initial and main user interfaces (UI) of the system for both the website and the application.
- Constructing and implementing the wire-frames into the final design of user interfaces (UI), and also for the structure of the web pages and the cascading styles.
- Coding with the goal to translate the design of the system into code in a given programming language to implement the design in the best possible manner in order to produce a working software application.
- Testing the application with different types and approaches will be performed in the system. User acceptance testing is the last part of testing and is performed by the end users to ensure the system meets their expectations.  
\*More details about this phase will be discussed in the next chapter.



## **7.2 Implementation Plan**

MCloud system goes through a specific and organized Implementation Plan which concise into two essential points, the implementation of the website (landing page) and the web application itself, also considering the integration needed from third-party application to form a complete and finished system.

### **7.2.1 MCloud Website**

The purpose of implementing MCloud website is mainly to be considered as a primarily a means of communication for audiences worldwide. This implementation included the home page as landing page for the website in order to display information, latest features and news with the contact information form.

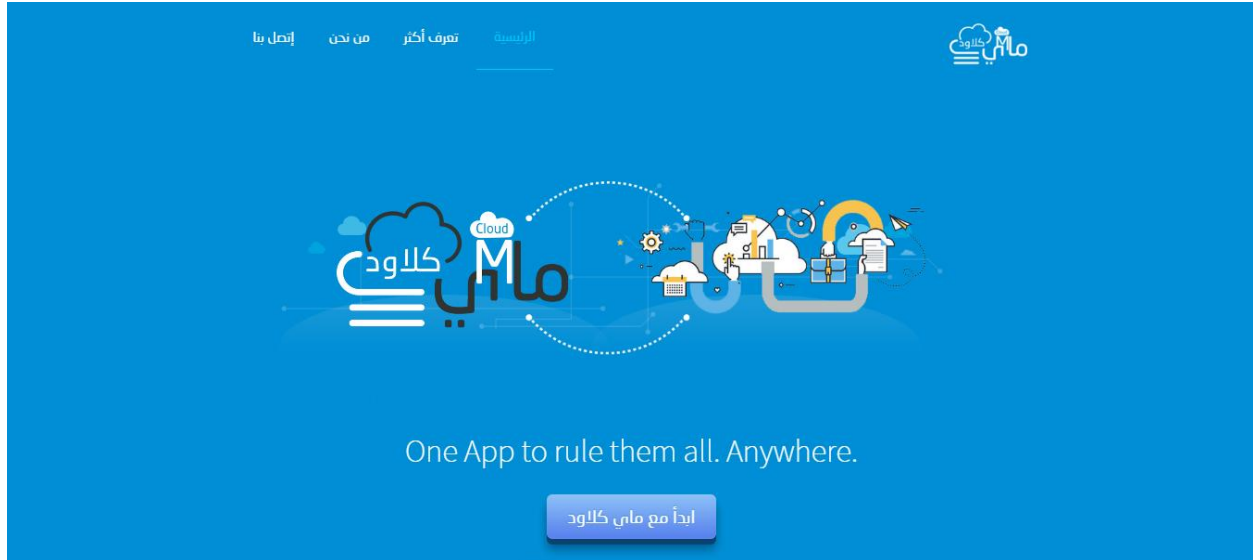
### **7.2.2 MCloud Application**

The implementation of MCloud system involves the web-based application itself which includes coding the actual program by following the pre-defined requirements which need to be considered. Furthermore all features presented in functional requirements will be implanted such as adding cloud drive and browsing files and also sharing it on social media etc., also the integration of third-party application and their APIs is essential to our system and they were implemented properly according to their formal documentation from the source website of APIs.

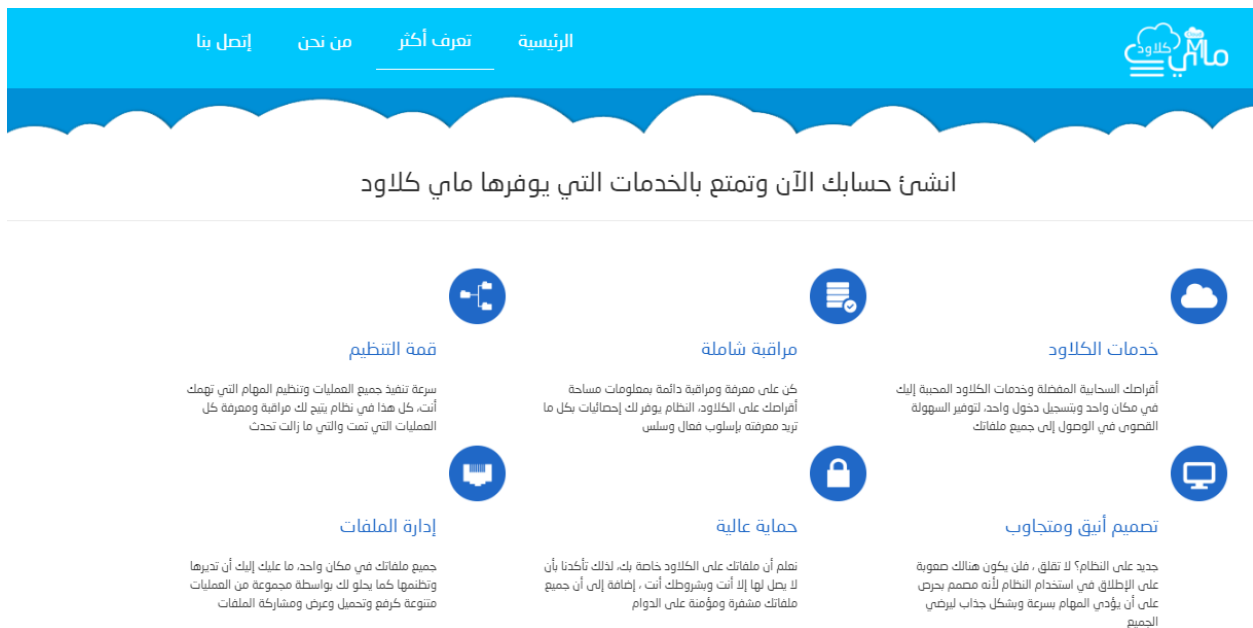
## 7.3 Website Implementation

### 7.3.1 MCloud Website Graphical User Interface (GUI)

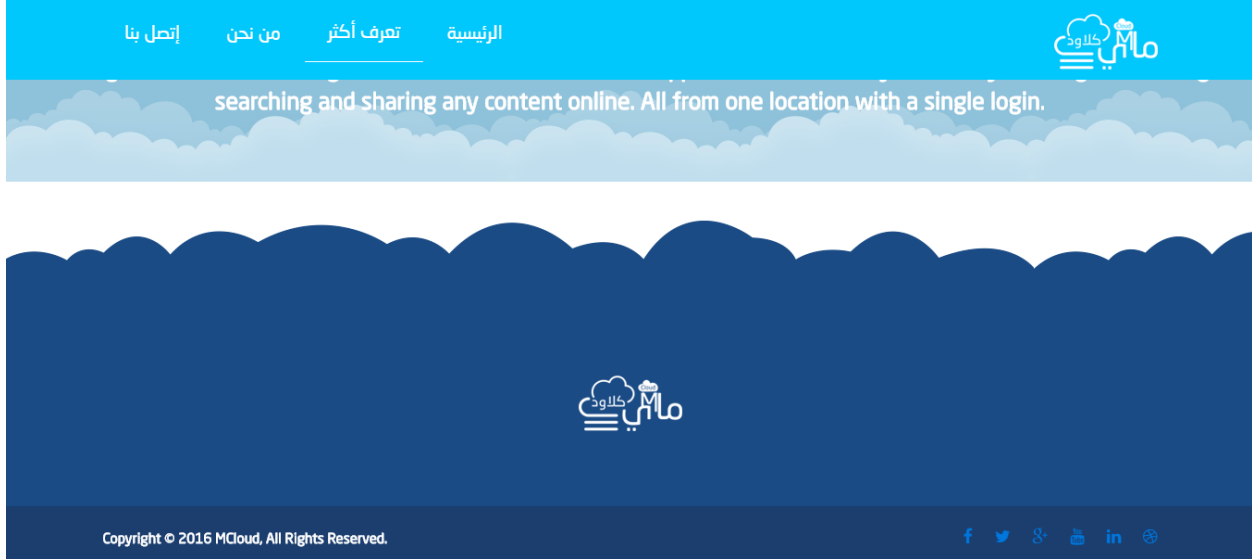
#### 7.3.1.1 Website: Home Section



#### 7.3.1.2 Website: Services Section



### 7.3.1.3 Website: Footer section



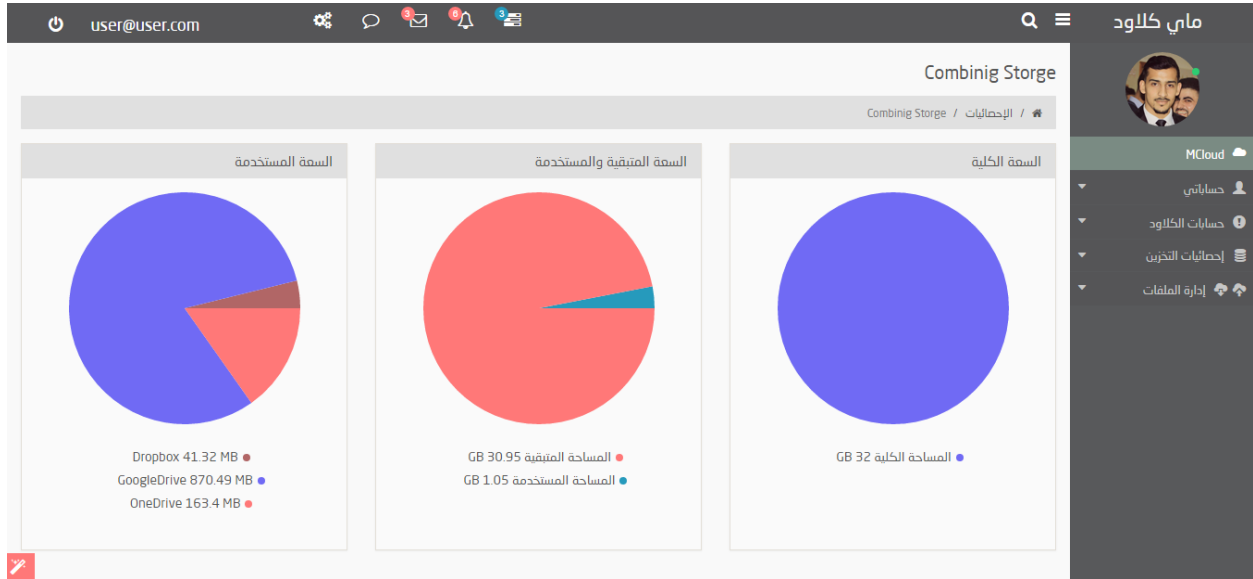
### 7.3.1.4 Website: Login & Register



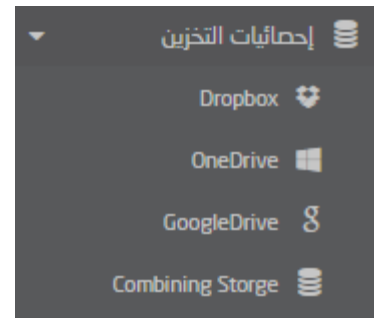
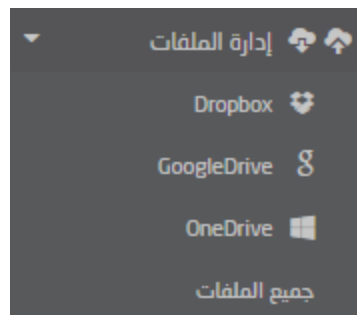
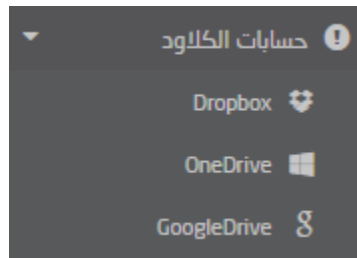
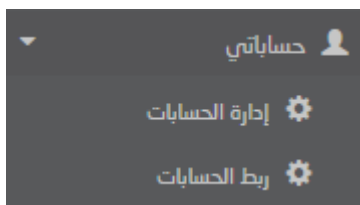
## 7.4 Application Implementation

### 7.4.1 MCloud Application Graphical User Interface (GUI)

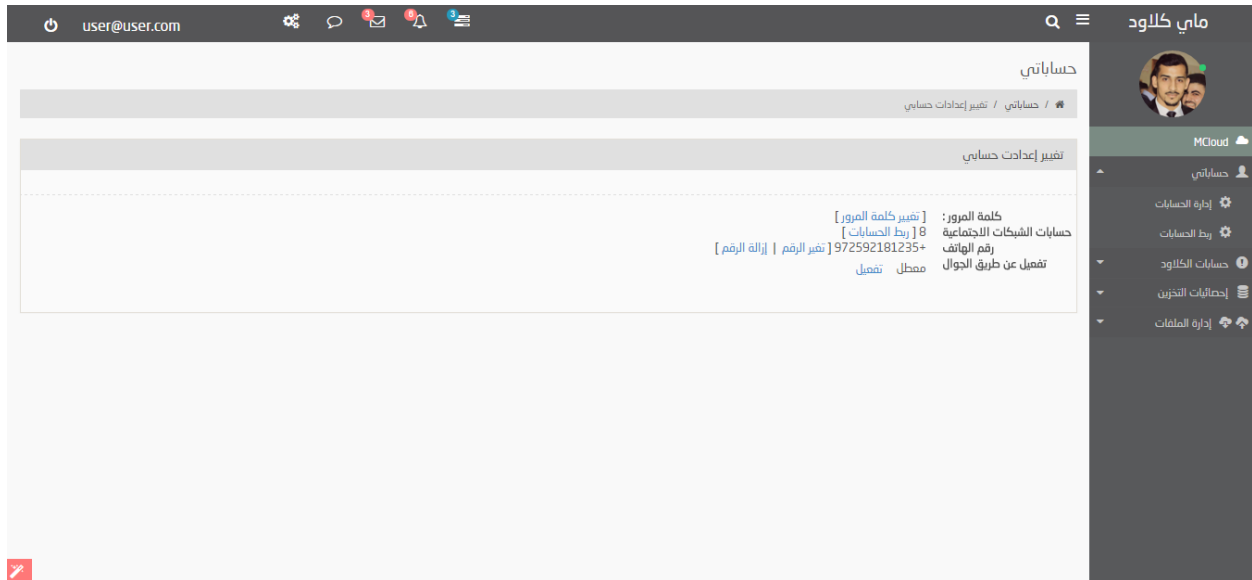
#### 7.4.1.1 Application: Home



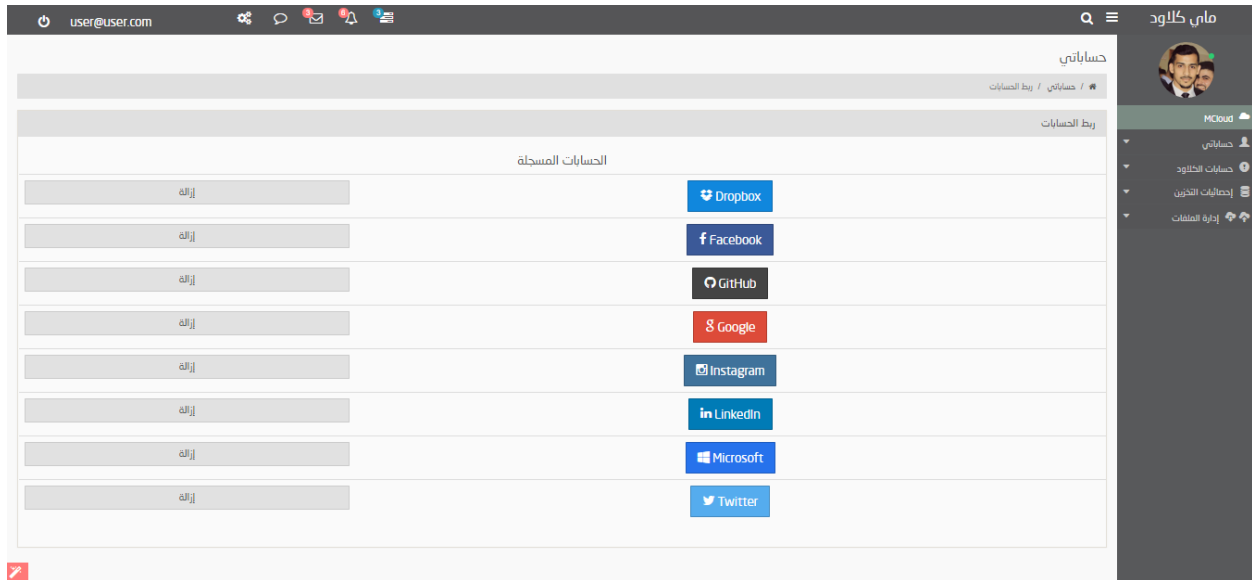
#### 7.4.1.2 Website: Side bar menu



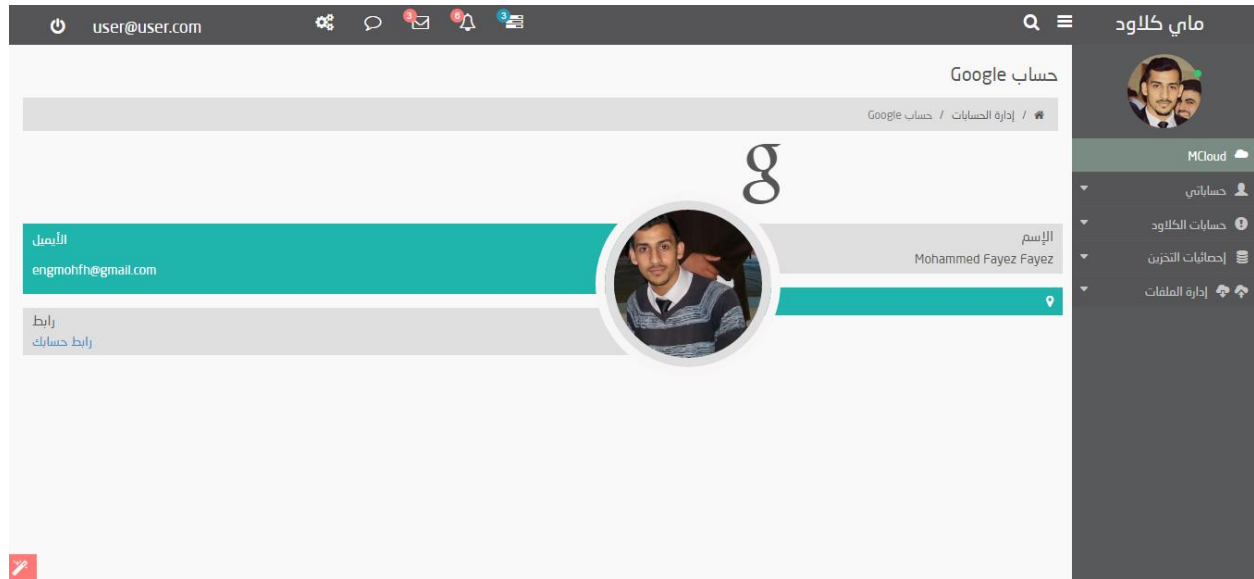
### 7.4.1.3 Application: Account Management



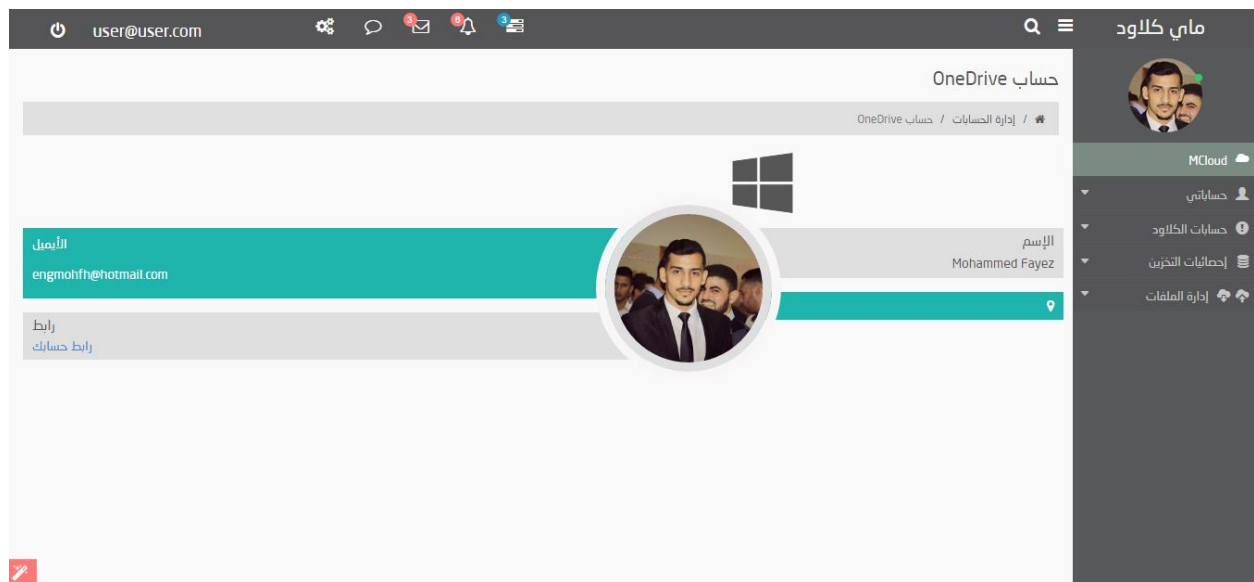
### 7.4.1.4 Application: Third Party Accounts Management



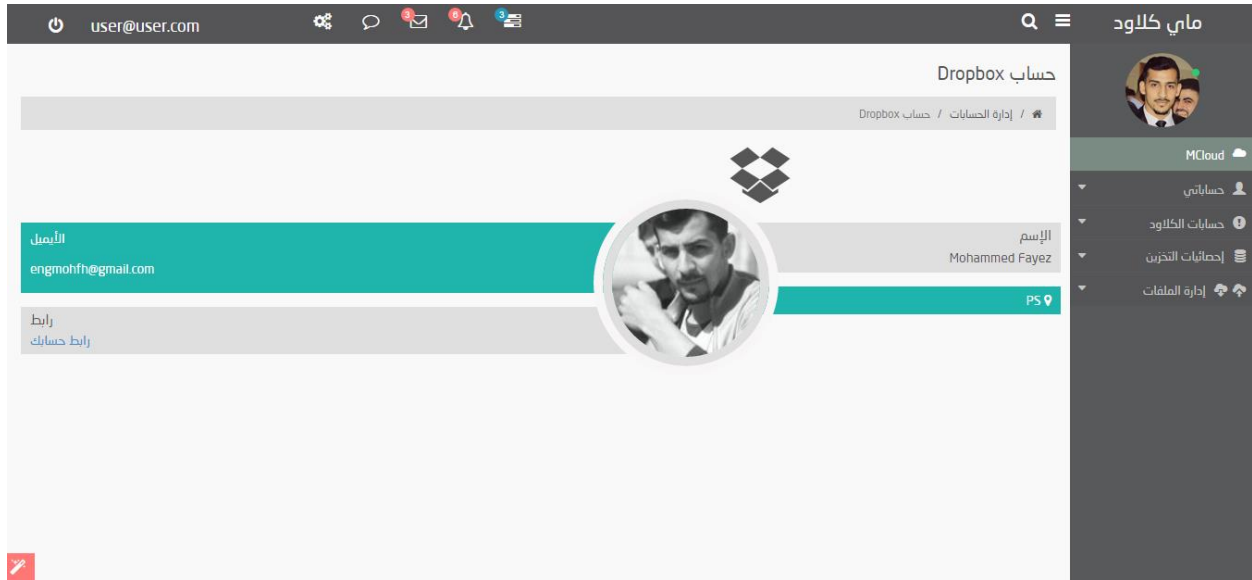
### 7.4.1.5 Application: Google Drive Account



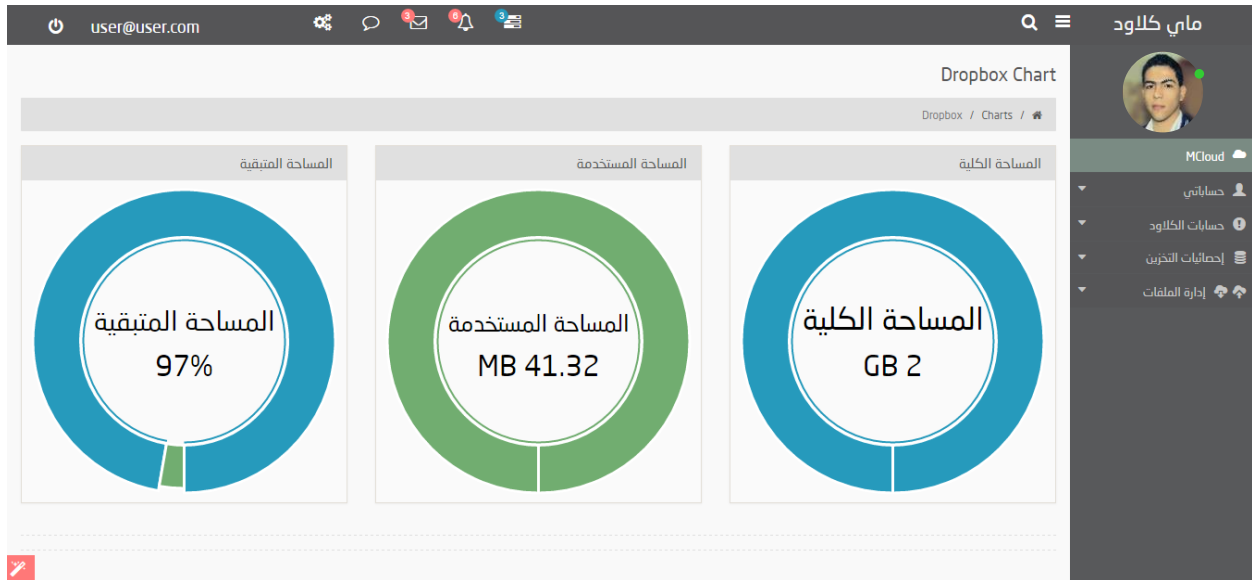
### 7.4.1.6 Application: OneDrive Account



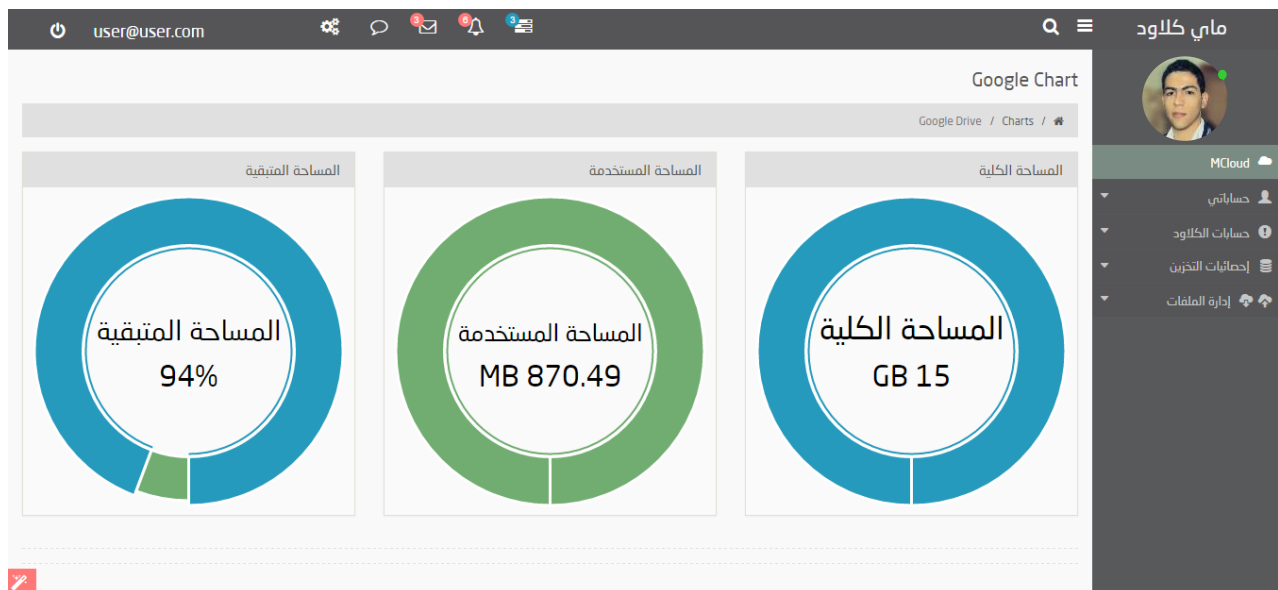
### 7.4.1.7 Application: Dropbox Account



### 7.4.1.8 Application: Dropbox Statistic

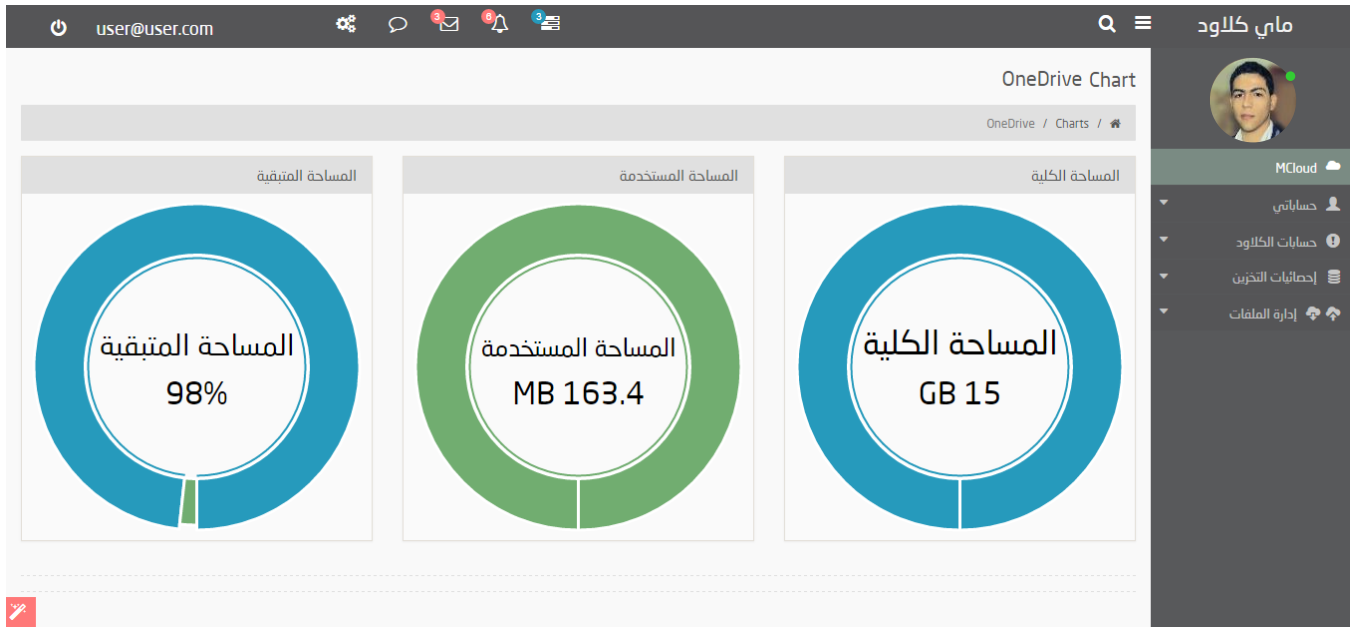


### 7.4.1.9 Application: Google Statistic





## 7.4.1.10 Application: OneDrive Statistic



## 7.4.1.11 Application: Dropbox File Manger

The screenshot displays the Dropbox File Manager application interface in Arabic. The top navigation bar includes the user's email 'user@user.com', search, and menu icons. The main content area is divided into two panes:

- File Preview Pane (Left):** Shows a preview of a file named 'Alaa.jpg'. Below the image, it displays the file name 'Alaa.jpg', the format 'image/jpeg', the size 'KB 76.1', and the upload date 'Thu, 23 Jun 2016 17:29:46 +0000'. There are 'Preview' and 'Share' buttons, and a share link 'https://db.tt/JHSpofTQ' with social media sharing options for LinkedIn, Pinterest, Google+, Twitter, and Facebook.
- File List Pane (Right):** Shows a list of files and folders in the Dropbox account. The files include: '15k.png', 'Ahmad\_Ahmad\_cv.pdf', 'Ahmad\_Ahmad\_cv.rar', 'Ahmad\_Ahmad\_P11.pdf.doc', 'Alaa.jpg', 'Doc\_Final.pdf', 'FaceApp.png', 'Get Started with Dropbox.pdf', 'Graduation Project', 'logo.png', 'MomentCam--0.jpg', 's.jpg', 'Screenshots', 'TechSmith', 'MP4.03-34-00\_2016-04-22', 'Test', and 'برنامج إدارة معارض البصريات.docx'.

The right sidebar shows the user's profile picture and the name 'ماهي كلاود' (Mahey Klawod). Below it, there are links for 'حساباتي' (My Accounts), 'حسابات الكلاود' (Cloud Accounts), 'إحصائيات التخزين' (Storage Statistics), and 'إدارة الملفات' (File Management).

user@user.com

إدارة الملفات

Dropbox / إدارة الملفات /

معلومات الملف

Dropbox

Graduation Project

Upload UI

Advance Upload

.. Browse Upload

Share

Copy | <https://db.tt/YHXQgsGH>

15k.png

Ahmad\_Ahmad\_cv.pdf

Ahmad\_Ahmad\_cv.rar

Ahmad\_Ahmad\_P11.pdf.doc

Doc. Final.pdf

FaceApp.png

Get Started with Dropbox.pdf

Graduation Project

logo\_png.png

MomentCam--0.jpg

s.jpg

Screenshots

TechSmith

Test

برنامج إدارة معارض المصريات.docx

ماي كلاود

MCloud

حساباتي

حسابات الكلود

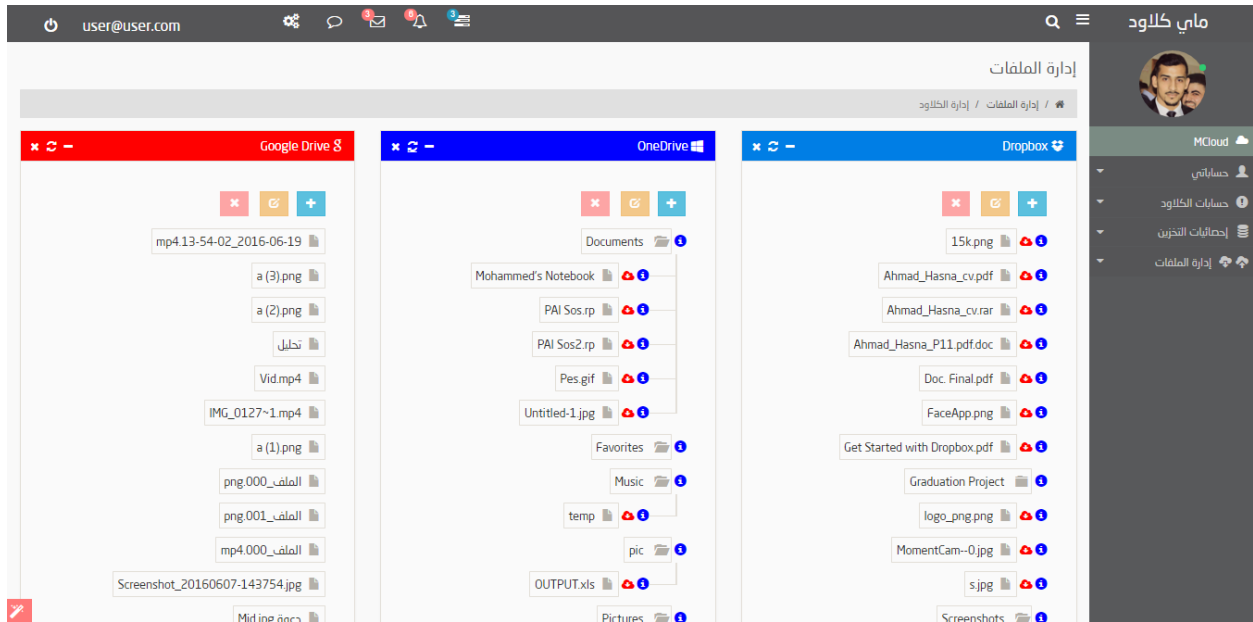
إحصائيات التخزين

إدارة الملفات

## 7.4.1.12 Application: OneDrive File Manger

The screenshot displays a web application interface for file management. The top navigation bar includes a user profile icon, a search icon, and a menu icon. The main content area is divided into two panels. The left panel, titled "معلومات الملف" (File Information), shows a preview of a photograph of a man in a suit. Below the image, the file details are listed: "s.jpg", "image/jpeg", "KB 130.3", and "Sat, 30 Apr 2016 01:13:14 +0000". There are buttons for "Preview" and "Share", along with a "Copy" button and a URL "https://db.tt/cTxMVgYo". Social media sharing icons for LinkedIn, Pinterest, Google+, Twitter, and Facebook are also present. The right panel, titled "Dropbox", shows a list of files and folders, including "15k.png", "Ahmad\_Hasna\_cv.pdf", "Ahmad\_Hasna\_cv.rar", "Ahmad\_Hasna\_P11.pdf.doc", "Doc. Final.pdf", "FaceApp.png", "Get Started with Dropbox.pdf", "Graduation Project", "logo\_png.png", "MomentCam--0.jpg", "s.jpg", "Screenshots", "TechSmith", "Test", and "برنامج إدارة ممرض البصريات.docx". The interface is in Arabic, and the user's name "ماي كلاود" (My Cloud) is visible in the top right corner.

### 7.4.1.13 Application: All Drivers Files Manger



## 7.4.2 MCloud Application Coding

### 7.4.2.1 Class: BundleConfig , Method: RegisterBundles

Bundling and magnification are two techniques that can be used in ASP.NET 4.5 to improve request load time. It also improves the load time by reducing the number of requests to the server and reducing the size of requested assets (such as CSS and JavaScript.)

```
using System.Web;
using System.Web.Optimization;

namespace Mcloud
{
    1 reference
    public class BundleConfig
    {
        1 reference
        public static void RegisterBundles(BundleCollection bundles)
        {
            bundles.Add(new ScriptBundle("~/bundles/jquery").Include(
                "~/Scripts/jquery-{version}.js"));

            bundles.Add(new ScriptBundle("~/bundles/jqueryval").Include(
                "~/Scripts/jquery.validate*"));

            bundles.Add(new ScriptBundle("~/bundles/Custom").Include(
                "~/Scripts/Custom/bootstrap.min.js",
                "~/Scripts/Custom/jquery-countTo.js",
                "~/Scripts/Custom/jquery.appear.js",
                "~/Scripts/Custom/jquery.fancybox.pack.js",
                "~/Scripts/Custom/jquery.mixitup.min.js",
                "~/Scripts/Custom/jquery.nav.js",
                "~/Scripts/Custom/jquery.parallax-1.1.3.js",
                "~/Scripts/Custom/main.js",
                "~/Scripts/Custom/owl.carousel.min.js",
                "~/Scripts/Custom/wow.min.js",
                "~/Scripts/Custom/vendor",
                "~/Scripts/Custom/underscore.js"
            ));

            bundles.Add(new ScriptBundle("~/bundles/App").Include(
                "~/AppAssets/js/pages/pace.min.js",
                "~/AppAssets/js/color.js",
                "~/AppAssets/js/lib/jquery-1.11.min.js",
                "~/AppAssets/js/bootstrap.min.js",
                "~/AppAssets/js/multipleAccordion.js",
                "~/AppAssets/js/lib/jquery.easing.js",
                "~/AppAssets/js/jquery.nanoscroller.min.js",
                "~/AppAssets/js/switchery.min.js",
                "~/AppAssets/js/bootstrap-switch.js",
                "~/AppAssets/js/jquery.easypiechart.min.js",
                "~/AppAssets/js/bootstrap-progressbar.min.js",
                "~/AppAssets/js/chart/flot/jquery.flot.js",
                "~/AppAssets/js/chart/flot/jquery.flot.pie.js",
                "~/AppAssets/js/chart/flot/jquery.flot.resize.js",
                "~/AppAssets/js/pages/layout.js",
                "~/AppAssets/js/countUp.min.js",
                "~/AppAssets/js/skycons.js",
                "~/AppAssets/js/jvectormap/jquery-jvectormap-1.2.2.min.js",
                "~/AppAssets/js/jvectormap/jquery-jvectormap-world-mill-en.js",
                "~/AppAssets/js/jquery.amaran.js",
                "~/AppAssets/js/pages/dashboard.js"
            ));
        }
    }
}
```

### 7.4.2.2 OAuth2, Methods: Configure...()

The three OAuth2 methods below (`ConfigureGooglePlus()`, `ConfigureMicrosoft()` and `ConfigureDropbox()`) enables users to log in with credentials from the specified external provider which are Microsoft, or Google, Dropbox and then integrate some of the functionality from those providers into your web application. MCloud allows logins from other external providers such as Facebook, Twitter, Instagram, GitHub and LinkedIn.

```
1 reference
private GoogleOAuth2AuthenticationOptions ConfigureGooglePlus()
{
    var googlePlusOptions = new GoogleOAuth2AuthenticationOptions()
    {
        AccessType = "offline",
        ClientId = "777400590-i4kjqtnvubt0s3jp31cf68522bb21q4q.apps.googleusercontent.com",
        ClientSecret = "vIB4r7V8jAcVy3a9s6hTON03",

        Provider = new GoogleOAuth2AuthenticationProvider()
        {
            OnAuthenticated = context =>
            {
                context.Identity.AddClaim(new System.Security.Claims.Claim("Google_AccessToken", context.AccessToken));

                if (context.RefreshToken != null)
                {
                    context.Identity.AddClaim(new Claim("GoogleRefreshToken", context.RefreshToken));
                }
                context.Identity.AddClaim(new Claim("GoogleUserId", context.Id));
                context.Identity.AddClaim(new Claim("GoogleTokenIssuedAt", DateTime.Now.ToBinary().ToString()));
                var expiresInSec = (long)(context.ExpiresIn.Value.TotalSeconds);
                context.Identity.AddClaim(new Claim("GoogleTokenExpiresIn", expiresInSec.ToString()));
                context.Identity.AddClaim(new Claim("GoogleAccountName", context.Name + " " + context.FamilyName));
                context.Identity.AddClaim(new Claim("GoogleAccountProfile", context.Profile));
                context.Identity.AddClaim(new Claim("GoogleAccountEmail", context.Email));
                context.Identity.AddClaim(new Claim("GoogleAccountPicture", context.User["image"].Value<string>("url")));
            }
        }
    };
}
```

```

1 reference
private MicrosoftAccountAuthenticationOptions ConfigureMicrosoft()
{
    var options = new MicrosoftAccountAuthenticationOptions
    {
        ClientId = "000000004C18A919",
        ClientSecret = "ky7-670bBCpHGbB1us7QsKkbUKzQbuEg",
        Provider = new MicrosoftAccountAuthenticationProvider
        {
            OnAuthenticated = context =>
            {
                context.Identity.AddClaim(new System.Security.Claims.Claim("Microsoft_AccessToken", context.AccessToken));

                if (context.RefreshToken != null)
                {
                    context.Identity.AddClaim(new Claim("MicrosoftRefreshToken", context.RefreshToken));
                }
                context.Identity.AddClaim(new Claim("MicrosoftUserId", context.Id));
                context.Identity.AddClaim(new Claim("MicrosoftTokenIssuedAt", DateTime.Now.ToBinary().ToString()));
                var expiresInSec = (long)(context.ExpiresIn.Value.TotalSeconds);
                context.Identity.AddClaim(new Claim("MicrosoftTokenExpiresIn", expiresInSec.ToString()));
                context.Identity.AddClaim(new Claim("MicrosoftAccountName", context.FirstName + " " + context.LastName));

                return Task.FromResult(true);
            }
        }
    }
}

```

```

1 reference
private DropboxAuthenticationOptions ConfigureDropbox()
{
    var options = new DropboxAuthenticationOptions
    {
        AppKey = "9s3fls0ov4hpag4",
        AppSecret = "x4c130fh4vipy24",
        Provider = new DropboxAuthenticationProvider
        {
            OnAuthenticated = context =>
            {
                context.Identity.AddClaim(new System.Security.Claims.Claim("Dropbox_AccessToken", context.AccessToken));

                context.Identity.AddClaim(new Claim("DropboxUserId", context.Id));
                context.Identity.AddClaim(new Claim("DropboxTokenIssuedAt", DateTime.Now.ToBinary().ToString()));
                context.Identity.AddClaim(new Claim("DropboxAccountName", context.Name ));

                return Task.FromResult(true);
            }
        },
        SignInAsAuthenticationType = DefaultAuthenticationTypes.ExternalCookie
    };

    return options;
}

```



### 7.4.2.3 Access Token, Method: ..SaveAccessToken()

An access token is an opaque string that identifies a user, app, or Page and can be used by the app to make graph API calls. Access tokens are obtained via a number of methods, below is a few of them from Google, Microsoft and Drobox.

```
2 references
private async Task GoogleSaveAccessToken(ApplicationUser user)
{
    // Get the claims identity
    ClaimsIdentity claimsIdentity =
        await AuthenticationManager.GetExternalIdentityAsync(DefaultAuthenticationTypes.ExternalCookie);

    if (claimsIdentity != null)
    {
        // Retrieve the existing claims
        var currentClaims = await UserManager.GetClaimsAsync(user.Id);

        // Get the list of access token related claims from the identity
        var tokenClaims = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("Google_AccessToken"));
        var RefreshTokenClaims = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleRefreshToken"));
        var GoogleUserId = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleUserId"));
        var GoogleTokenIssuedAt = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleTokenIssuedAt"));
        var GoogleTokenExpiresIn = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleTokenExpiresIn"));
        var GoogleAccountNames = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleAccountName"));
        var GoogleAccountProfiles = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleAccountProfile"));
        var GoogleProfilePic = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleAccountPicture"));
        var GoogleEmail = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("GoogleAccountEmail"));
    }
}
```

```
2 references
private async Task MicrosoftSaveAccessToken(ApplicationUser user)
{
    // Get the claims identity
    ClaimsIdentity claimsIdentity =
        await AuthenticationManager.GetExternalIdentityAsync(DefaultAuthenticationTypes.ExternalCookie);

    if (claimsIdentity != null)
    {
        // Retrieve the existing claims
        var currentClaims = await UserManager.GetClaimsAsync(user.Id);

        // Get the list of access token related claims from the identity
        var tokenClaims = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("Microsoft_AccessToken"));
        var RefreshTokenClaims = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("MicrosoftRefreshToken"));
        var MicrosoftUserId = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("MicrosoftUserId"));
        var MicrosoftTokenIssuedAt = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("MicrosoftTokenIssuedAt"));
        var MicrosoftTokenExpiresIn = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("MicrosoftTokenExpiresIn"));
        var MicrosoftAccountNames = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("MicrosoftAccountName"));
    }
}
```

```

2 references
private async Task DropboxSaveAccessToken(ApplicationUser user)
{
    // Get the claims identity
    ClaimsIdentity claimsIdentity =
        await AuthenticationManager.GetExternalIdentityAsync(DefaultAuthenticationTypes.ExternalCookie);

    if (claimsIdentity != null)
    {
        // Retrieve the existing claims
        var currentClaims = await UserManager.GetClaimsAsync(user.Id);

        // Get the list of access token related claims from the identity
        var tokenClaims = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("Dropbox_AccessToken"));
        var DropboxUserId = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("DropboxUserId"));
        var DropboxTokenIssuedAt = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("DropboxTokenIssuedAt"));
        var DropboxAccountNames = claimsIdentity.Claims
            .Where(c => c.Type.StartsWith("DropboxAccountName"));
    }
}

```

#### 7.4.2.4 Method: GetIdentity()

Part of method which responsible for returning the entity identity (ClaimsIdentity) which's suitable for hashing.

```

4 references
public ClaimsIdentity GetIdentity()
{
    var claimsIdentity = new ClaimsIdentity();
    try
    {
        claimsIdentity = HttpContext.User.Identity as ClaimsIdentity;
    } catch (Exception e)
    {
        ViewBag.error = e;
    }

    return claimsIdentity;
}

```

### 7.4.2.5 Method: GetAccessToken()

This call will return an app access token which can be used in place of a user access token to make API calls

```
2 references
public string GetAccessToken()
{
    var claimsIdentity = GetIdentity();
    if (claimsIdentity != null)
    {
        var DropAccessToken = claimsIdentity.Claims.FirstOrDefault(x => x.Type == "Dropbox_AccessToken");
        if (DropAccessToken != null)
        {
            return DropAccessToken.Value;
        }
        else
        {
            return "No Access Token Please ensure your login";
        }
    }
    else
    {
        return "Please ensure your login";
    }
}
```

### 7.4.2.6 Method: Initialize()

Initializes data that might not be available when the constructor is called.

```
12 references
public Client initialize()
{
    var options = new Options
    {
        ClientId = "9s3fls0ov4hpag4",
        ClientSecret = "x4c130fh4vipy24",
        AccessToken = GetAccessToken(),
    };

    // Initialize a new Client (with an AccessToken)
    var client = new Client(options);

    return client;
}
```

### 7.4.2.7 Method: GetAccountInfo()

This method is specified to each cloud drive we included in our system which is responsible for retrieving all data and files from the specified drive, in this method it's Dropbox.

```
0 references
public async Task<ActionResult> GetAccountInfo()
{
    Client client = initialize();
    // Get account info
    var accountInfo = await client.Core.Accounts.AccountInfoAsync();
    AccountInfo info = new AccountInfo();
    DropboxProfile drop = await GetUserInfo2();
    info.country = accountInfo.country;
    info.display_name = accountInfo.display_name;
    info.email = accountInfo.email;
    info.locale = drop.ImageUrl;
    info.name_details = accountInfo.name_details;
    DropboxRestAPI.Models.Core.QuotaInfo quota = new DropboxRestAPI.Models.Core.QuotaInfo();
    quota.normal = accountInfo.quota_info.normal;
    quota.quota = accountInfo.quota_info.quota;
    quota.shared = accountInfo.quota_info.shared;
    info.quota_info = quota;
    info.referral_link = accountInfo.referral_link;
    info.uid = accountInfo.uid;

    return RedirectToAction("Dropbox", "App", info);
}
```

### 7.4.2.8 Method: GetUserStorage()

The method responsible for retrieving the amount of storage space of a Dropbox cloud drive account of the user.

```
0 references
public async Task<ActionResult> GetUserStorage()
{
    Client client = initialize();
    // Get account info
    var accountInfo = await client.Core.Accounts.AccountInfoAsync();
    DropboxRestAPI.Models.Core.QuotaInfo quota = new DropboxRestAPI.Models.Core.QuotaInfo();
    quota.normal = accountInfo.quota_info.normal;
    quota.quota = accountInfo.quota_info.quota;
    quota.shared = accountInfo.quota_info.shared;

    TempData["DropboxQuota"] = quota;
    return RedirectToAction("DropboxChart", "App");
}
```

### 7.4.2.9 Method: CreateNewFolder()

This method is responsible for allowing the user to create a new folder in a specified cloud drive of the user.

```
0 references
public async Task<bool> CreateNewFolder(string foldername, string path)
{
    Client client = initialize();
    var newFolder = await client.Core.FileOperations.CreateFolderAsync("/" + path + foldername);
    if (newFolder != null)
    {
        return true;
    }
    return false;
}
```

### 7.4.2.10 Method: Download()

This method is responsible for allowing the user to download files from cloud drive into their PC.

```
0 references
public async Task<ActionResult> Download(string path)
{
    Metadata file;
    SharedLink link;
    var tempFile = Path.GetFileName(path);
    using (var fileStream = System.IO.File.OpenWrite(@"f:\\" + tempFile))
    {
        Client client = initialize();
        file = await client.Core.Metadata.FilesAsync(path, fileStream);
        link = await client.Core.Metadata.SharesAsync(path);
    }

    string filename = tempFile;
    string filepath = @"f:\\" + tempFile;
    byte[] filedata = System.IO.File.ReadAllBytes(filepath);
    string contentType = MimeMapping.GetMimeMapping(filepath);

    var cd = new System.Net.Mime.ContentDisposition
    {
        FileName = filename,
        Inline = false,
    };

    Response.AppendHeader("Content-Disposition", cd.ToString());
    System.IO.File.Delete(filepath);
    return File(filedata, contentType);
}
```

### 7.4.2.11 Method: Share()

This method is responsible for allowing the user to share files on social media website.

```
0 references
public async Task<ActionResult> Share(string path)
{
    SharedLink link;
    var tempFile = Path.GetFileName(path);
    Client client = initialize();
    link = await client.Core.Metadata.SharesAsync(path);
    ViewBag.ShareLink = link;
    return PartialView("ShareLink");
}
```

### 7.4.2.12 Method: GetRootFolder()

This method is responsible for retrieving all the folders from a particular cloud drive to be shown in the user MCloud account of that cloud drive.

```
0 references
public async Task<ActionResult> GetRootFolder()
{
    Client client = initialize();
    var rootFolder = await client.Core.Metadata.MetadataAsync("/", list: true);

    foreach (var folder in rootFolder.contents)
    {
        if (folder.is_dir)
        {
            folder.contents = await GetContentFolder("/") + folder.Name);
        }
    }

    TempData["DropboxRoot"] = rootFolder.contents;
    return RedirectToAction("Dropbox", "FileManager");
}
2 references
public async Task<List<MetaData>> GetContentFolder(string path)
{
    Client client = initialize();

    var Folder = await client.Core.Metadata.MetadataAsync(path, list: true);
    foreach (var folder in Folder.contents)
    {
        if (folder.is_dir)
        {
            folder.contents = await GetContentFolder(folder.path);
        }
    }

    return Folder.contents;
}
}
```

### 7.4.2.13 Method: GetFileThump()

This method is responsible for getting the thumbnail of an image.

```
0 references
public async Task<ActionResult> GetFileThump(string path)
{
    Client client = initialize();
    var info = await client.Core.Metadata.ThumbnailsAsync(path, "jpeg", "1");
    System.Drawing.Image image = System.Drawing.Image.FromStream(info);
    Random rnd = new Random();
    string thumpname = "ThumpImage"+ rnd.Next(1000);
    var stream = System.IO.File.OpenWrite(Server.MapPath("~/ " + thumpname + ".jpg"));

    image.Save(stream, ImageFormat.Jpeg);
    stream.Close();
    ViewBag.thumpname = thumpname;
    return PartialView("_thumpImage");
}
}
```

#### 7.4.2.14 Method: FindFileInRoot()

This method is responsible for listing all the files inside a specific folder.

```
0 references
public async Task<bool> FindFileInRoot(string name)
{
    Client client = initialize();
    var rootFolder = await client.Core.Metadata.MetadataAsync("/", list: true);
    var file = rootFolder.contents.FirstOrDefault(x => x.Name == name);
    if (file != null)
    {
        return true;
    }
    else
    {
        return false;
    }
}
```

#### 7.4.2.15 Method: Upload()

This method is responsible for uploading files from the user PC into a cloud drive.

```
1 reference
public async Task<MetaData> Upload (string path, string uploadPath, Stream fileStream, string DropAccessToken)
{
    MetaData uploadedFile;
    var tempFile = Path.GetFileName(path);
    Client client = initialize();
    client.UserAccessToken = DropAccessToken;
    uploadedFile = await client.Core.Metadata.FilesPutAsync(fileStream, uploadPath + "/" + tempFile);
    return uploadedFile;
}
```

#### 7.4.2.16 Method: Search()

This method is responsible for integrated searching for any files inside the connected cloud drives.

```
0 references
public async Task<String> Search(string name)
{
    // Search file based on name
    Client client = initialize();
    var searchResults = await client.Core.Metadata.SearchAsync("/", name);
    foreach (var searchResult in searchResults)
    {
        return "Found: " + searchResult.path;
    }
    return "Not Found";
}
```

## 7.5 Conclusion

This chapter starts with a brief about the construction and implementation plan for MCloud system which was divided into website and web-based application, then follows with screen-shots of graphical user interface (GUI) for both the website and the web application. Furthermore some of the methods and classes of the actual coding of the system were mentioned briefly.

The development team attempt as much as possible to commit with design rules, standards of user Interfaces (UI) as well as code style, convention and readable code comments.





# Chapter 8: Testing and Evaluation

## Content

- 8.1 Introduction
- 8.2 Purpose of Testing
- 8.3 Testing Methodologies
- 8.4 Test Implementation
- 8.5 Test Automation Tools
- 8.6 Testing Checklist
- 8.7 Evaluation
- 8.8 Conclusion

# 8 CHAPTER 8: TESTING AND EVALUATION

## 8.1 Introduction

This chapter describes how the testing and evaluation process will be implemented in MCloud system, and as generic is the idea of testing a software, still it's a phase as crucial as it's important for the sustainability and the quality of the system.

Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. [38]

Software is not unlike other physical processes where inputs are received and outputs are produced. Where software differs is in the manner in which it fails. Most physical systems fail in a fixed (and reasonably small) set of ways. By contrast, software can fail in many bizarre ways. Detecting all of the different failure modes for software is generally infeasible.

Software bugs will almost always exist in any software module with moderate size: not because programmers are careless or irresponsible, but because the complexity of software is generally intractable -- and humans have only limited ability to manage complexity. It is also true that for any complex systems, design defects can never be completely ruled out. [39]

Regardless of the limitations, testing is an integral part in software development. It is broadly deployed in every phase in the software development cycle. Typically, more than 50% percent of the development time is spent in testing. [40]

## 8.2 Purpose of Testing

Testing is usually performed for the following purposes:

- ❖ **To Improve Quality.**

As computers and software are used in critical applications, the outcome of a bug can be severe. Bugs can cause huge losses. Bugs in critical systems have caused airplane crashes, allowed space shuttle missions to go awry, halted trading on the stock market,

and worse. Bugs can kill. Bugs can cause disasters. The so-called year 2000 (Y2K) bug has given birth to a cottage industry of consultants and programming tools dedicated to making sure the modern world doesn't come to a screeching halt on the first day of the next century.

Quality means the conformance to the specified design requirement. Being correct, the minimum requirement of quality, means performing as required under specified circumstances. Debugging, a narrow view of software testing, is performed heavily to find out design defects by the programmer. The imperfection of human nature makes it almost impossible to make a moderately complex program correct the first time. Finding the problems and get them fixed is the purpose of debugging in programming phase. [41]

#### ❖ **For Verification & Validation (V&V) [42]**

Just as topic Verification and Validation indicated, another important purpose of testing is verification and validation (V&V). Testing can serve as metrics. It is heavily used as a tool in the V&V process. Testers can make claims based on interpretations of the testing results, which either the product works under certain situations, or it does not work. We can also compare the quality among different products under the same specification, based on results from the same test.

We cannot test quality directly, but we can test related factors to make quality visible. Quality has three sets of factors -- functionality, engineering, and adaptability. These three sets of factors can be thought of as dimensions in the software quality space. Each dimension may be broken down into its component factors and considerations at successively lower levels of detail. Table 1 illustrates some of the most frequently cited quality considerations.

Good testing provides measures for all relevant factors. The importance of any particular factor varies from application to application. Any system where human lives are at stake must place extreme emphasis on reliability and integrity.

Tests with the purpose of validating the product works are named clean tests, or positive tests. The drawbacks are that it can only validate that the software works for the specified test cases. A finite number of tests cannot validate that the software works for all situations. On the contrary, only one failed test is sufficient enough to show that the software does not work. Dirty tests, or negative tests, refers to the tests aiming at breaking the software, or showing that it does not work. A piece of software must have sufficient exception handling capabilities to survive a significant level of dirty tests.

### ❖ For Reliability Estimation

Software reliability has important relations with many aspects of software, including the structure, and the amount of testing it has been subjected to. Based on an operational profile (an estimate of the relative frequency of use of various inputs to the program, testing can serve as a statistical sampling method to gain failure data for reliability estimation.

Software testing is not mature. It still remains an art, because we still cannot make it a science. We are still using the same testing techniques invented 20-30 years ago, some of which are crafted methods or heuristics rather than good engineering methods. Software testing can be costly, but not testing software is even more expensive, especially in places that human lives are at stake. Solving the software-testing problem is no easier than solving the Turing halting problem. We can never be sure that a piece of software is correct. We can never be sure that the specifications are correct. No verification system can verify every correct program. We can never be certain that a verification system is correct either. [43]

## 8.3 Testing Methodologies

There are different methodologies that can be used when implementing testing tools. The testing methodologies can be used together or individually. It all depends on what the testing is about by providing better resources for different tasks. Below are a few of the main different types that are used but there are other types of testing that can be used:

### 8.3.1 Unit Testing

This is the software used for verification and validation. It's a method in which the programmer can test all the separate code and see if it is viable to use. This type of testing is based on a small scale and uses small units of the program. When looking at procedural programming the unit can be any individual function or a procedure which is written in the same language as the production code.

### 8.3.2 White Box Testing

This is also known as clear box testing, glass box testing, translucent box testing or

structural testing. It uses the internal perspective of the system and then designs test cases based on this internal structure. Basically the code itself and all the conditions, statements and paths along with it are tested. Programming skills are required for noticing all the paths through the software. White box testing does many things such as analyzing the data flow, control flow, information flow and coding practices.

There are many techniques available in white-box testing, because the problem of intractability is eased by specific knowledge and attention on the structure of the software under test. The intention of exhausting some aspect of the software is still strong in white-box testing, and some degree of exhaustion can be achieved, such as executing each line of code at least once (statement coverage), traverse every branch statements (branch coverage), or cover all the possible combinations of true and false condition predicates (Multiple condition coverage). [44]

### **8.3.3 Black Box Testing**

The black-box approach is a testing method in which test data are derived from the specified functional requirements without regard to the final program structure. It is also termed data-driven, input/output driven, or requirements-based testing. In testing, various inputs are exercised and the outputs are compared against specification to validate the correctness. All test cases are derived from the specification. No implementation details of the code are considered. [45]

It is obvious that the more we have covered in the input space, the more problems we will find and therefore we will be more confident about the quality of the software. Ideally we would be tempted to exhaustively test the input space. But as stated above, exhaustively testing the combinations of valid inputs will be impossible for most of the programs, let alone considering invalid inputs, timing, sequence, and resource variables.

### **8.3.4 Regression Testing**

This is a type of software testing which basically looks for software regressions. The regressions can happen whenever the software stops working as it is meant to. The testing makes sure that any bugs or changes have been fixed which may have caused previous working functions to have failed as a result of the newly added features. It is better to do this type of testing under the automated tool environment. The regression testing may take place once the programmer has tried to fix a problem or has purposely added in code to give out errors.

It's important for developers and testers to always bear in mind that even small, seemingly insignificant alterations to an application's source code can ripple outward in surprising ways, breaking functions that seem completely unrelated to the new modification. When you run regression tests, you're checking to make sure that your modification not only behaves as you want it to, but that it also hasn't inadvertently caused problems in functions that had otherwise worked correctly when previously tested. [46]

### **8.3.5 Integration Testing**

This is where the separate software modules are put together and are then tested as a whole. This normally takes place after unit testing has been done. The point of doing integration testing is because it verifies the functional and performance requirements of the design items. It is component of Extreme Programming (XP). Two different ways of doing integration testing is bottom-up and top-down. [47]

## **8.4 Test Implementation**

Test Explorer in visual studio can run third-party and open source unit test frameworks that have implemented add-on interfaces.

After creating Unit Test Project for MCloud System, the test framework we choose generated unit test classes for all the existing classes in the system project, so unit test methods can be implemented for any desired method inside any class.

Test Implementation

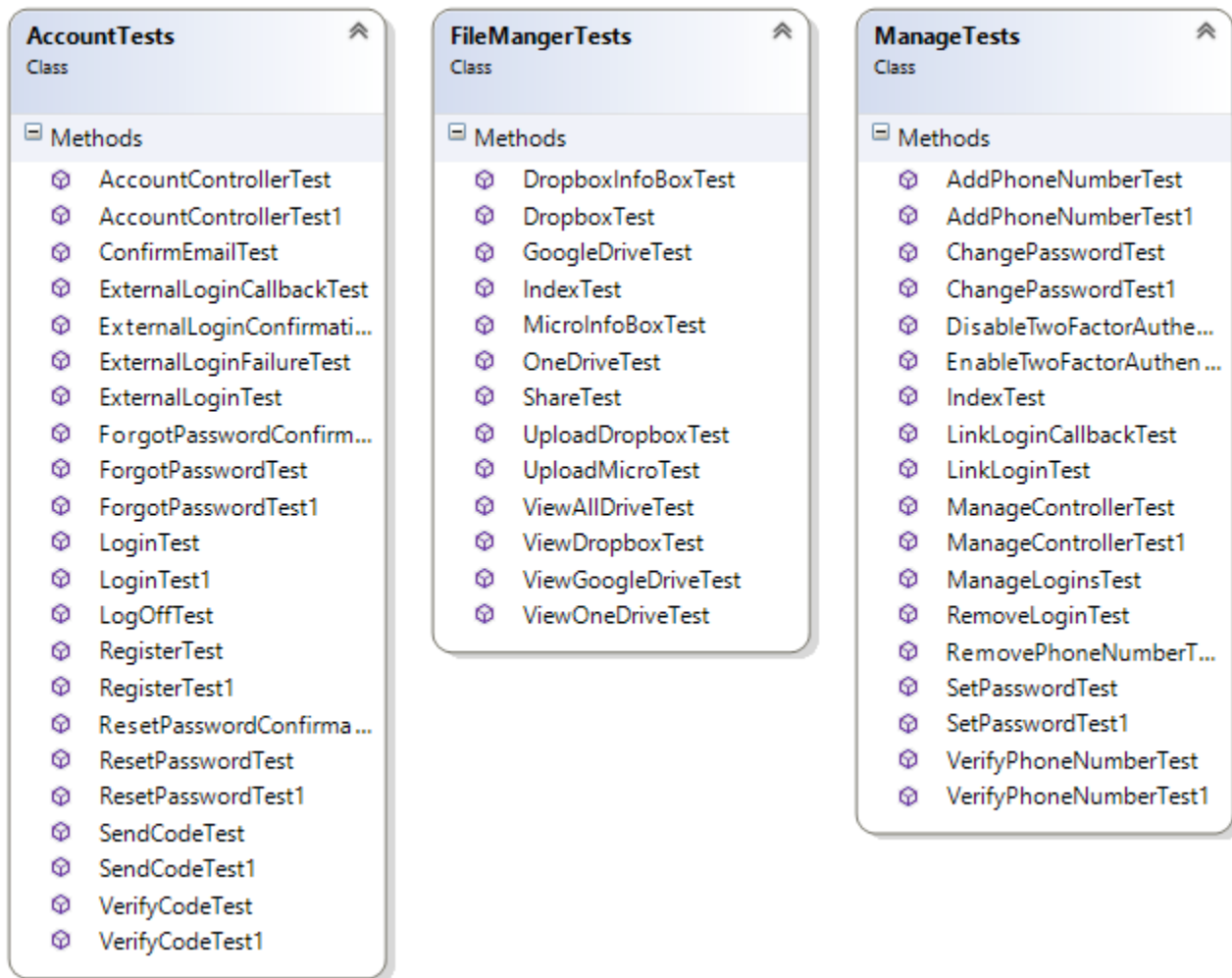


Figure 7 - 1: Unit Test Classes

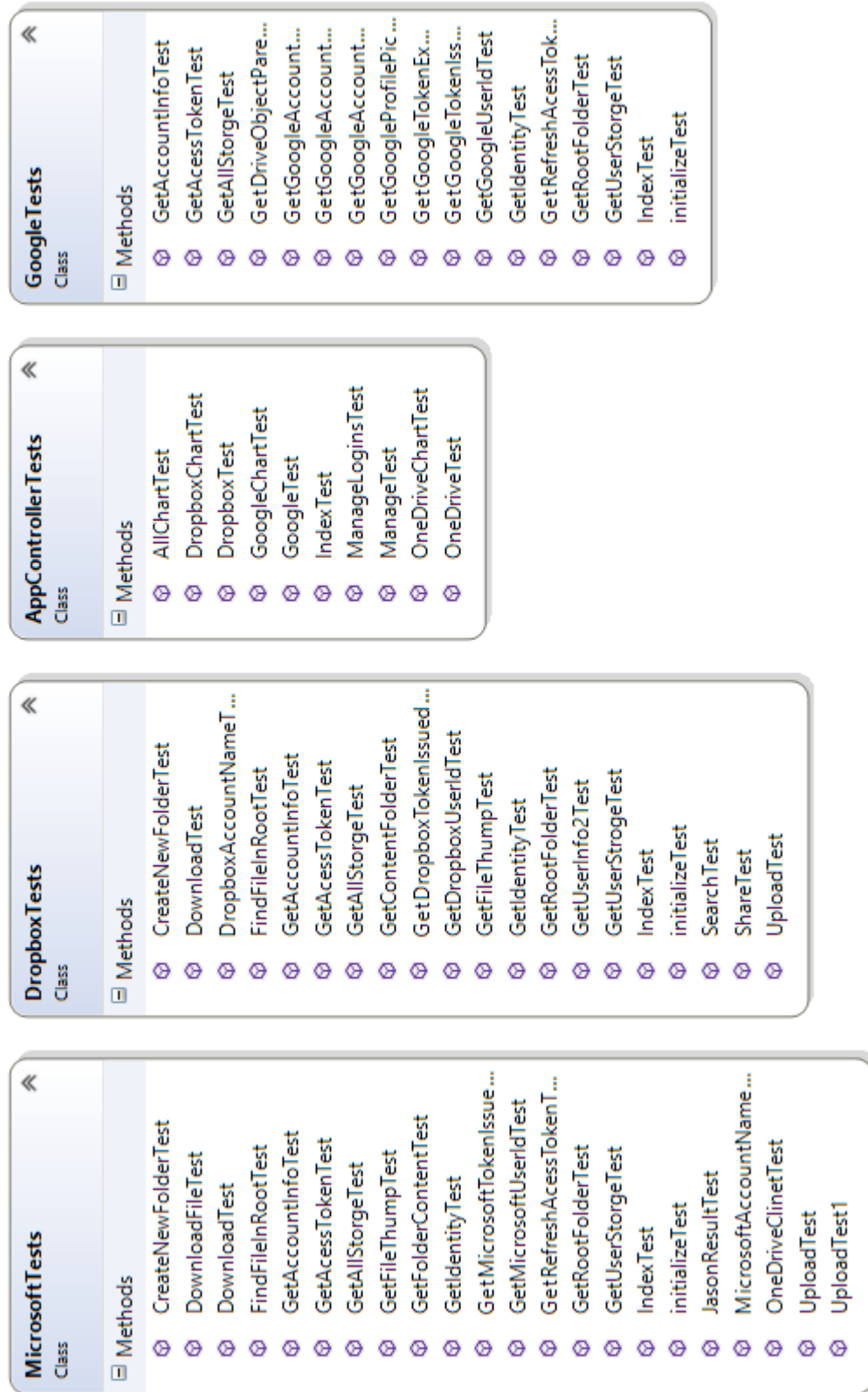


Figure 7 - 2: Third Party Unit Test Classes



## 8.4.1 DropboxTests Class

Implementing a Test Method for all the methods in DropboxTests class.

```
[TestMethod()]
0 references
public void GetIdentityTest()
{
    Assert.Fail();
}

[TestMethod()]
0 references
public void GetAccessTokenTest()
{
    Assert.Fail();
}

[TestMethod()]
0 references
public void GetDropboxUserIdTest()
{
    Assert.Fail();
}

[TestMethod()]
0 references
public void GetDropboxTokenIssuedAtTest()
{
    Assert.Fail();
}

[TestMethod()]
0 references
public void DropboxAccountNameTest()
{
    Assert.Fail();
}
```

## 8.5 Test Automation Tools

Automated software testing is becoming more and more important for many software projects in order to automatically verify key functionality, test for regressions and help teams run a large number of tests in a short period of time. Many teams (especially larger projects) still require a significant amount of manual functional testing in addition to automated testing, either because of the lack of sufficient resources or skills to automate all tests

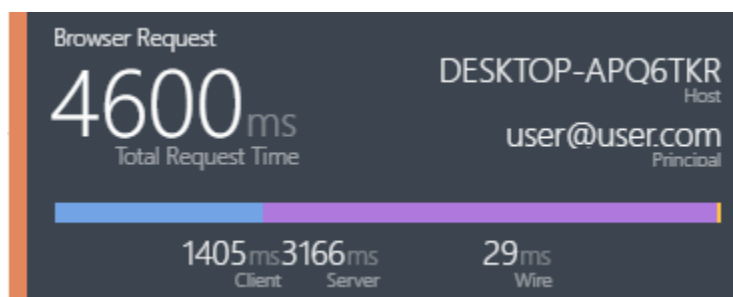
### 8.5.1 Glimpse

Glimpse is a thriving and growing family of open source NuGet packages that provides detailed performance, debugging and diagnostic information for ASP.NET apps. It's trivial to install, lightweight, ultra-fast, and displays key performance metrics at the bottom of every page. It allows you to drill down into your app when you need to find out what's going on at the server. [48]

It is a tool that bridges the gap between knowing what's happening on the client-side with knowing what's happening on the back-end. It runs inside the context of the browsers. There are no tools to install, only a minor installation and configuration change to your server.

Here's some of the result of implementing Glimpse in MCloud system.

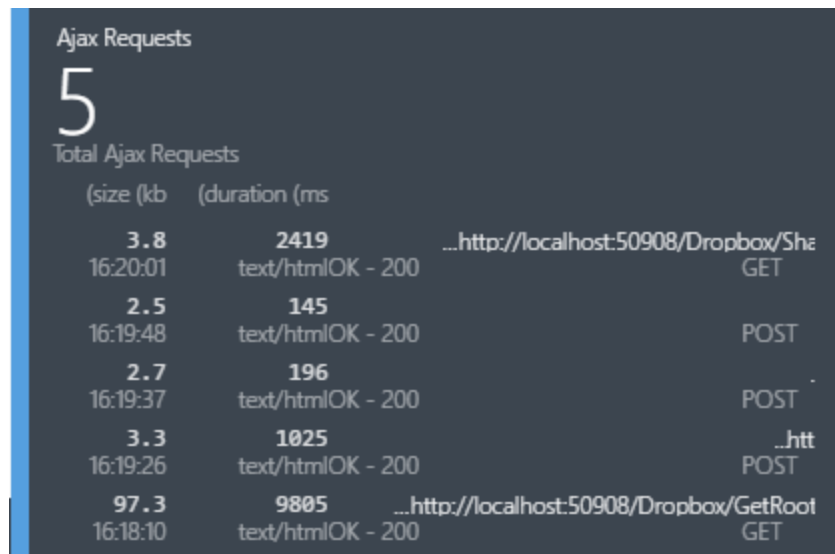
#### 8.5.1.1 Browser Request



### 8.5.1.2 Server Side



### 8.5.1.3 Ajax Requests



### 8.5.1.4 The Timeline tab

It consolidates the event timing information from all tabs into a single visualization.

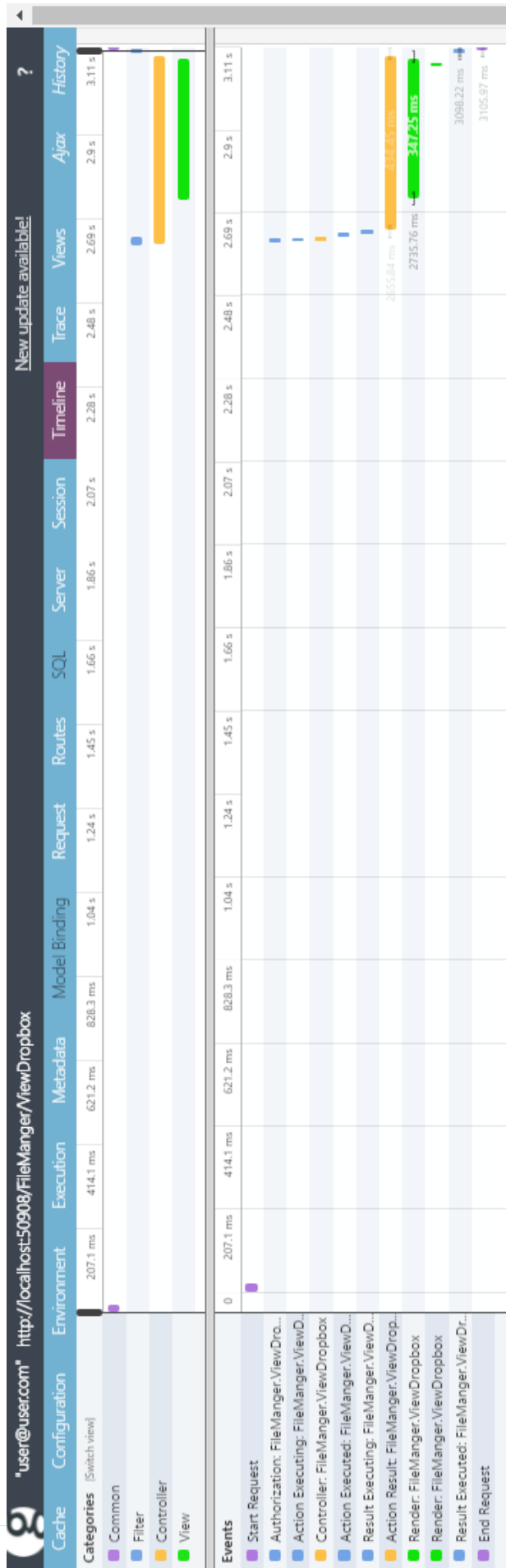


Figure 8 - 1: The Timeline Tab

## 8.5.2 Diagnostic Tools in Visual Studio 2015

The diagnostic tools in visual studio 2015 dramatically enhances the debugging experience and saves valuable debugging time by adding visual graphs and detailed information of the process occurs in the system. Diagnostic tools use graphs to surface at-a-glance information, and when you see something of interest we can select a range of time to filter the detailed information in the specific tabs.

In this release, there are three major tools in the Diagnostics Tools window:

- **Events:** provides a visual history of events that stopped or resumed the execution of the application.
- **Memory Usage tool:** allows monitoring memory usage and investigate memory issues while debugging.
- **CPU Usage tool:** allows to see how much CPU resources are used by the code.

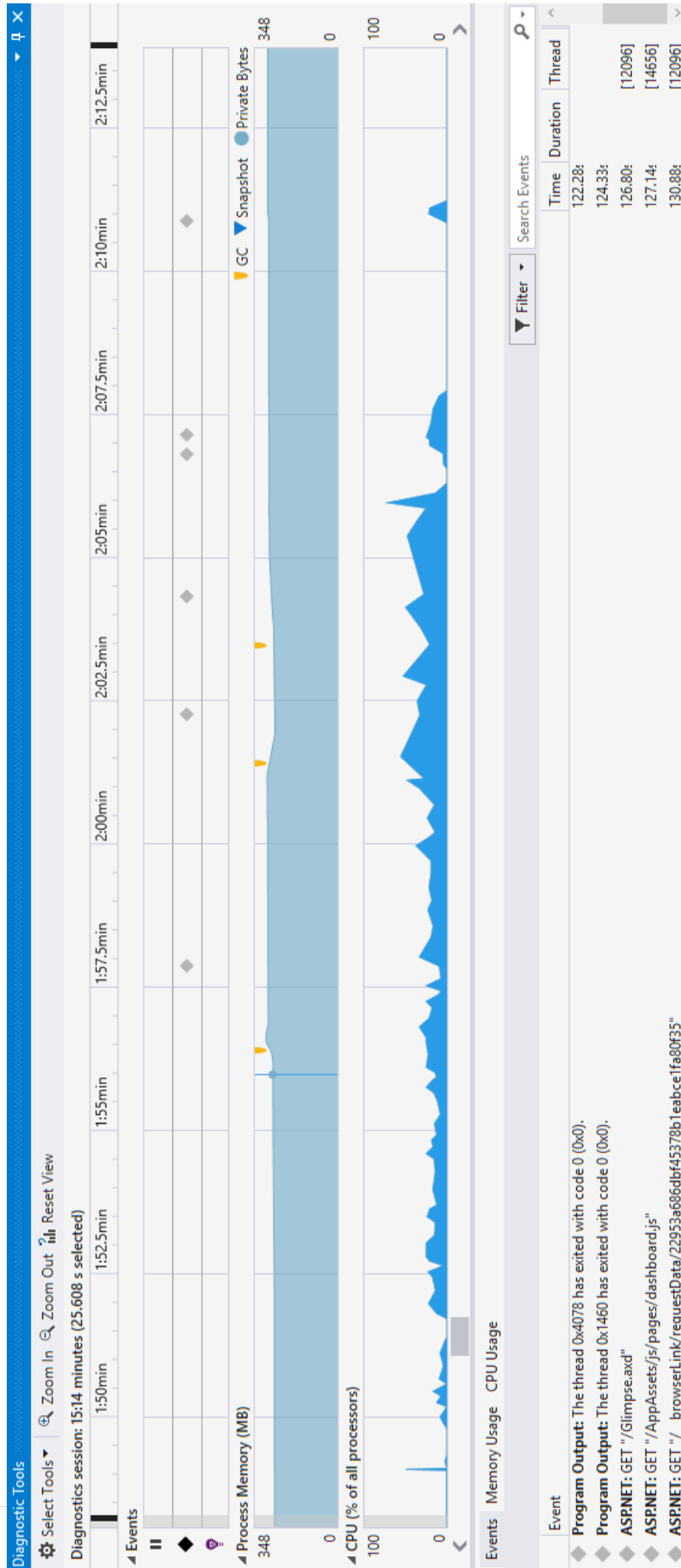


Figure 8 - 2: Events, Memory and CPU Usage

## 8.6 Testing Checklist

### 8.6.1 Functional Testing

Test Carried Out	Expected	Actual
<b>1. Functionality</b>		
<b>1.1 Links</b>		
<b>Internal Links</b>	Should be present	√ <b>Passed</b>
<b>External Links</b>	Should be present	√ <b>Passed</b>
<b>Mail to links</b>	Should open mailbox	√ <b>Passed</b>
<b>Orphan pages</b>	Should not be present	√ <b>Passed</b>
<b>Broken links</b>	Should not be present	√ <b>Passed</b>
		√ <b>Passed</b>
<b>1.2 Forms</b>		
<b>Field Level checks</b>	Checked for length, special characters, numerical characters etc.,	√ <b>Passed</b>
<b>Field Level Validation</b>	Checked Unique records, Date validation	√ <b>Passed</b>
<b>Functional checks</b>	Create, modify, view and delete are working.	√ <b>Passed</b>
<b>Error Handling for wrong inputs or actions.</b>	Appropriate error messages to be displayed.	√ <b>Passed</b>
<b>Optional and mandatory fields.</b>	Mandatory field should not be left blank. Optional should allow the user to skip the field.	√ <b>Passed</b>
		√ <b>Passed</b>
<b>1.3 Cookies</b>		
<b>Check whether cookies are enabled.</b>	It should ask	X <b>Failed</b>
		√ <b>Passed</b>

<b>1.4 Web Indexing</b>		√ <b>Passed</b>
<b>Meta tags</b>	Should be present.	√ <b>Passed</b>
<b>Html Syntax</b>	Should be valid	√ <b>Passed</b>
<b>Frames</b>	To be found ok	√ <b>Passed</b>
		√ <b>Passed</b>
<b>1.5 Database</b>		
<b>Data Integrity</b>	Should not be any missing or wrong data in the database	√ <b>Passed</b>
<b>Output Errors</b>	Errors in writing, reading or editing operations should not be present	√ <b>Passed</b>

Table 8 - 1: Functional Testing Checklist

## 8.6.2 Usability Testing

<b>2. Usability</b>		
<b>2.1 Navigation</b>		
<b>Navigation through Mouse</b>	Should be proper	√ <b>Passed</b>
<b>Navigation through Tab</b>	Should be proper	√ <b>Passed</b>
<b>Main features access</b>	Should be accessed from home/Main page	√ <b>Passed</b>
<b>Hot Keys, Control Keys for menu or action</b>	Should be present	√ <b>Passed</b>
		√ <b>Passed</b>
<b>2.2 Content</b>		
<b>Spelling and Grammar</b>	To be proper	√ <b>Passed</b>
<b>Updated information's.</b>	Past events/ information's to be removed.	√ <b>Passed</b>
		√ <b>Passed</b>
<b>2.3 General Appearance</b>		



<b>Page Appearance</b>	Should not be any overlapping, missing etc.,	√ <b>Passed</b>
<b>Colour, font and size</b>	Should be as per standard	√ <b>Passed</b>
<b>Frames</b>	All frames to be appeared	√ <b>Passed</b>
<b>Consistent Design</b>	Everywhere in the website consistent layout and design should be carried out.	√ <b>Passed</b>

Table 8 - 2: Usability Testing Checklist

### 8.6.3 Server Side Interface

3. Server Side Interface		
<b>3.1 Server Interface</b>	Communication should be correct with respect to Web server, App server and DB server	√ <b>Passed</b>
<b>Compatibility with server hardware, SW, network connections</b>	Should be proper.	√ <b>Passed</b>
<b>Database compatibility</b>	Should be easily portable to other database.	√ <b>Passed</b>

Table 8 - 3: Server Side Interface Checklist

## 8.6.4 Client Side Compatibility

4. Client side Compatibility		
<b>4.1 Platform</b>		
<b>Windows</b>	Should be working	√ <b>Passed</b>
<b>Unix</b>	Should be working	N/A
<b>Linux</b>	Should be working	N/A
<b>Solaris</b>	Should be working	N/A
<b>Mac</b>	Should be working	N/A
<b>4.2 Browsers</b>		
<b>Chrome</b>	Should be working	√ <b>Passed</b>
<b>Firefox</b>	Should be working	√ <b>Passed</b>
<b>Opera</b>	Should be working	√ <b>Passed</b>
<b>Edge</b>	Should be working	√ <b>Passed</b>
<b>Graphics</b>	Load of images, graphics should be proper	√ <b>Passed</b>

Table 8 - 4: Client Side Compatibility Checklist

## 8.6.5 Performance Testing

5. Performance		
<b>5.1 Connection speed</b>		
<b>Timeout</b>	Should give appropriate time to search. Incorrect message, data loss should not be present.	√ <b>Passed</b>
<b>5.2 Load</b>		
<b>Estimated users.</b>	Per requirements	√ <b>Passed</b>

<b>Peak load</b>	Should withstand	√ <b>Passed</b>
<b>Large amount of data from users</b>	Should accept	√ <b>Passed</b>
<b>5.3 Stress</b>		
<b>System Crash</b>	Should not be present	√ <b>Passed</b>
<b>Error in SW, HW, Memory</b>	Leakage, overwrite should not happen.	√ <b>Passed</b>
<b>5.4 Continuous use</b>		
<b>Estimate whether available for 24 Hrs, 7 days a week</b>	Try with various timings.	√ <b>Passed</b>
<b>Downtime</b>	Measure the downtime	√ <b>Passed</b>
<b>Memory or disk space</b>	Should not run out of memory or disk space.	√ <b>Passed</b>

Table 8 - 5: Performance Testing Checklist

## 8.6.6 Security Testing

<b>6. Security</b>		
<b>6.1 Valid and Invalid</b>	Should not enter with Invalid login	√ <b>Passed</b>
<b>Number of tries</b>	Should not be more than 3 times for invalid try.	√ <b>Passed</b>
<b>Enter URL directly without logging in.</b>	Should not display information.	√ <b>Passed</b>
<b>Log files</b>	Should be maintained	√ <b>Passed</b>
<b>Access to server scripts</b>	Authenticated.	√ <b>Passed</b>

Table 8 - 6: Security Testing Checklist

## 8.7 Evaluation

We presents a software evaluation checklist which shows whether or not MCloud system meets the predefined requirements. With questions from how well the software operates, to the overall care accuracy of the content itself, this checklist gives a good look at the big picture.

	YES	NO
<p>1. Does the software support the needs and objectives of the curriculum?</p> <ul style="list-style-type: none"> <li>• Are the objectives of the software documented?</li> <li>• Do the objectives match curriculum goals and mandated standards?</li> <li>• Are activities and evaluations matched to the objectives?</li> <li>• Does the software fill a need not addressed by software you currently use?</li> </ul>	√	
<p>2. Does the target audience match the intended learner audience?</p> <ul style="list-style-type: none"> <li>• Does the software match the learner audience in age, reading, and skill level?</li> <li>• Is the software accessible to students with disabilities?</li> <li>• Is the rate of display of information appropriate to the learner audience?</li> </ul>	√	
<p>3. Is the return on investment justifiable to the school and district?</p> <ul style="list-style-type: none"> <li>• Is the software cost efficient?</li> <li>• Is the necessary hardware available?</li> <li>• Can additional hardware be obtained easily?</li> </ul>	√	
<p>4. Is the software easy to use?</p>	√	

<ul style="list-style-type: none"> <li>• Are directions easily understood?</li> <li>• Can the user navigate through the software easily?</li> <li>• Are student records stored and easily retrieved?</li> </ul>		
<p>5. Is the content accurate?</p> <ul style="list-style-type: none"> <li>• Is the program free of gender and racial stereotypes?</li> <li>• Is the language free of errors in spelling, grammar, and pronunciation?</li> <li>• Does the program provide immediate feedback?</li> <li>• Is the feedback credible and context-based?</li> </ul>	√	
<p>6. Is the software an appropriate medium for learning the targeted skills?</p> <ul style="list-style-type: none"> <li>• Is text displayed consistently and easy to read?</li> <li>• Is the quality of visual and auditory elements acceptable?</li> <li>• Are graphics and animations appropriate to the target audience?</li> <li>• Do the graphics and animations support the program's intentions?</li> <li>• Does the program promote successively higher levels of cognitive activity?</li> <li>• Does the program incorporate active participation?</li> </ul>	√	

Table 8 - 7: Evaluation Checklist

## 8.8 Conclusion

This chapter summarizes the testing and evolution process for MCloud system along with shading some light upon the purpose of using software testing in general also it presented what are the testing methodologies that are commonly used when testing a software. Furthermore it shows how we implemented the test units in our code followed with using some of the automated testing tools available with the results it produced. Finlay it introduces a formal and standard testing and evaluation checklist which shows major testing and evaluation criteria that essential to our system and how it will operate in different levels and which will explain the main points that have been relied upon to evaluate the application, to include all aspects of the system in terms of performance, system interface, safety and compliance etc.



## CHAPTER 9: FUTURE WORKS AND CONCLUSION

### Content

- 9.1 Introduction
- 9.2 Future Works

# 9 CHAPTER 9: FUTURE WORKS AND CONCLUSION

## 9.1 Introduction

This chapter describes what features the next increments or releases should include which is separated in two categories, usability features and technical features, also the feedback which will be received after the publishing of the first release is important and it will be reviewed thoroughly to add the requested features and also to fix any problem that we failed to spot in the testing phase in order to develop and introduce a better experience for user in the future releases.

## 9.2 Future Works

### 9.2.1 Usability Future Perspective:

- MCloud system already supports both Arabic and English languages, but since the system is very extended due its reliance on the cloud storages that are available, it cloud supports other languages to expand the audience and allow more people to benefit from the system.
- Adding more control for Admin and support operators with user-friendly and integrated control panel.
- Providing a clear well-defined usage policy to inform users of what is allowed and what is not.
- Introducing a world class user guidelines and user manual for new comers and have difficulty understanding and dealing with the application, also it should include tips and tricks for user to better methods of managing their space storages and visitors who need help to register in the web application.



## 9.2.2 Technical Future Perspective:

- Supporting **file transferring** between different cloud drives with the best and easy way possible in order to increase the management of the users' files.
  
- Add **offline transferring** is planned which let user select what files they want to transfer and it will keep transferring even if they logged from their account since the process rely fully in the system server.
  
- Developing mobile applications on Android and IOS platforms to expand the accessibility for users in order to use the system easily.
- Developing a **desktop version** is as important as the mobiles versions, it will ease the accessibility of files much better especially if the syncing was in real time and accurate.
- Integrating more cloud storage services and content sharing application.
  
- Link **multiples accounts** of the same cloud drive service, for example, user who has three Dropbox accounts can integrates them together as one huge Dropbox account in MCloud system with access to all files from the three accounts at the same time.
  
- Password protected sharing.
  
- Integrating the major email services such as Gmail, Outlook and Yahoo together as a universal inbox.
  
- Allow more functionalities on the user files online such previewing video and PDF files, also integrating Google Docs and Microsoft Office Online.

## **10 Final Conclusion**

MCloud was built mainly to introduce a better way of using a great service and essential to most internet user nowadays, the online cloud storages changed the way people use and store their data online, and MCloud only exists to fix the main downside of using these cloud services which truly affected the time management, the downside is the scattering of files in many different cloud drives with different logins for each one, our system lets all of these drive come together and allow users to browse all the files in one place with single login.

It's a promising solutions and we are sure that our implementation of the solution will satisfy the users which our main goal period.

# 11 REFERENCES

- [1] [Online]. Available: <http://www.prweb.com/releases/2011/5/prweb8456395.htm>.
- [2] [Online]. Available: <https://www.multicloud.com/product>.
- [3] [Online]. Available: <http://www.goodcloudstorage.net/faq-odrive/>.
- [4] [Online]. Available: <http://www.taskey.com/resources/Related%20articles/importance-of-gantt-charts.aspx>.
- [5] "Evolving a new Software Development Life Cycle Model (SDLC) incorporated with release management, International Journal of Engineering and Advanced Technology (IJEAT), volume-I, Aril 2012, Vishwas Massey, Prof. K. J Satao".
- [6] "International Journal of Computer Science & Information Technology, A Comparison Between Evolutionary".
- [7] [Online]. Available: <http://er.yuvayana.org/sdlc-incremental-model-design-phase-applications-advantages-and-disadvantages/>.
- [8] "System Analysis and Design. With UML 2.0 an Object-Oriented Approach, Second Edition Dennies, Wixon, and Tegarden, Wiley 2005".
- [9] "Charting Your Course: Charts and Graphs for IT Projects - Dawn Li, Ph.D. and Gary McQuown - Data and Analytic Solutions, Inc".
- [10] "Ruth Malan and Dana Bredemeyer, ARCHITECTURE RESOURCES".
- [11] [Online]. Available: [https://www.wikiwand.com/en/Non-functional\\_requirement](https://www.wikiwand.com/en/Non-functional_requirement).
- [12] "Alain Abran, James W. Moore; Pierre Bourque, Robert Dupuis, eds. (March 2005). "Chapter 2: Software Requirements". Guide to the software engineering body of knowledge (2004 ed.)".
- [13] "Unified Modeling Language User Guide, The (2 ed.). Addison-Wesley. 2005. p. 496".
- [14] "Structuring Use Cases with Goals. Alistair Cockburn Humans and Technology."
- [15] [Online]. Available: <https://www.smartdraw.com/activity-diagram/>.
- [16] [Online]. Available: <http://www.ibm.com/developerworks/rational/library/3101.html>.
- [17] [Online]. Available: <https://www.techopedia.com/definition/438/clientserver-architecture>.
- [18] [Online]. Available: [https://msdn.microsoft.com/en-us/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/dd381412(v=vs.108).aspx).
- [19] [Online]. Available: <http://criticaltechnology.blogspot.com/2011/09/mvc-in-three-tier-architecture.html>.

- [20] "Business Component Factory : A Comprehensive Overview of Component-Based Development for the Enterprise 1st Edition , Page 88".
- [21] [Online]. Available: <http://rest.elkstein.org/>.
- [22] [Online]. Available: <https://msdn.microsoft.com/en-us/library/ff648105.aspx?f=255&MSPPErr=-2147217396>.
- [23] "Buschmann, Frank, et al. Pattern-Oriented Software Architecture. John Wiley & Sons Ltd, 1996.".
- [24] [Online]. Available: [https://msdn.microsoft.com/en-us/library/dd381412\(v=vs.108\).aspx](https://msdn.microsoft.com/en-us/library/dd381412(v=vs.108).aspx).
- [25] [Online]. Available: <http://www.codeguru.com/csharp/article.php/c18167/Top-7-Features-of-the-ASPNET-MVC-Framework.htm>.
- [26] [Online]. Available: [https://msdn.microsoft.com/en-us/library/h43ks021\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/h43ks021(v=vs.110).aspx).
- [27] [Online]. Available: <https://en.wikipedia.org/wiki/HTML5>.
- [28] [Online]. Available: <http://www.w3schools.com/js/>.
- [29] [Online]. Available: <https://jquery.com/>.
- [30] [Online]. Available: [https://developer.mozilla.org/en-US/docs/AJAX/Getting\\_Started](https://developer.mozilla.org/en-US/docs/AJAX/Getting_Started).
- [31] [Online]. Available: <https://www.edrawsoft.com/EDrawMax.php>.
- [32] [Online]. Available: <https://www.visualstudio.com/products/visual-studio-community-vs>.
- [33] [Online]. Available: [http://www.wikiwand.com/en/Microsoft\\_SQL\\_Server](http://www.wikiwand.com/en/Microsoft_SQL_Server).
- [34] [Online]. Available: [https://en.wikipedia.org/wiki/Internet\\_Information\\_Services](https://en.wikipedia.org/wiki/Internet_Information_Services).
- [35] [Online]. Available: <https://notepad-plus-plus.org>.
- [36] [Online]. Available: <http://www.adobe.com/products/photoshop.html>.
- [37] [Online]. Available: [http://www.sparxsystems.com/resources/uml\\_datamodel.html](http://www.sparxsystems.com/resources/uml_datamodel.html).
- [38] "Hetzel, William C., The Complete Guide to Software Testing, 2nd ed. Publication info: Wellesley, Mass.".
- [39] [Online]. Available: [http://www.rstcorp.com/definitions/software\\_testing.html](http://www.rstcorp.com/definitions/software_testing.html).
- [40] "Introduction of the tools BoundsChecker by NuMega".
- [41] " Cem Kaner, Testing Computer Software. 5th edition 2014".
- [42] "Hetzel, William C., The Complete Guide to Software Testing, 2nd ed. Publication info: Wellesley, Mass.".

- [43] "Michael R. Lyu , Handbook of Software Reliability Engineering".
- [44] "Norman Parrington and Marc Roper, Understanding Software Testing,".
- [45] "William E. Howden. Functional program Testing and Analysis.".
- [46] [Online]. Available: <https://smartbear.com/learn/automated-testing/what-is-regression-testing/>.
- [47] [Online]. Available:  
[http://searchsoftwarequality.techtarget.com/sDefinition/0,,sid92\\_gci1243430,00.html#](http://searchsoftwarequality.techtarget.com/sDefinition/0,,sid92_gci1243430,00.html#).
- [48] [Online]. Available:  
<http://www.hanselman.com/blog/IfYoureNotUsingGlimpseWithASPNETForDebuggingAndProfilingYoureMissingOut.aspx>.



Appendences

# 12 APPENDICES

## Use Cases Detailed Descriptions

UC 1 : Login	
Actor	Admin, User
Relation	Uses U2, Extended by U16, U18, U20 and U25
Description	Actor logs in to authenticate his role in the system and be able to use the functionalities of the system.
Preconditions	A valid account has been created for the actor with specific permissions and privileges according to his role.
Post-conditions	The actor can successfully access the system and perform actions appropriate for his role.
Normal Path	5) The actor connects to the system. 6) The actor chooses one of the social media presented and available to login to the system. Or by entering his username and password. 7) The system determines the role of the actor. 8) The system displays the main user interface of the application to the actor.
Abnormal Path	<ul style="list-style-type: none"> <li>• Login failed due to internet connection failure</li> <li>• Actor <b>doesn't</b> have an account.</li> </ul>
Comments	Admin cannot have access to the <b>USER'S</b> personal files to guarantee the users privacy.

*A - UC Table 1: Login*

UC 2 : Logout	
Actor	Admin, User
Relation	Used by U1
Description	Logging out of the system.
Preconditions	Logged in to the system with valid account.
Post-conditions	Existing the application and redirected back to the home website.
Normal Path	<ol style="list-style-type: none"> <li>1) The actor presses the logout icon in the top bar.</li> <li>2) The system redirects the actor to the home website.</li> </ol>
Abnormal Path	Error due to network failure.
Comments	None

*A - UC Table 2: Logout*



UC 3 : Favorite Files	
Actor	User
Relation	None
Description	Selecting certain files as favorites to be saved in a dedicated list which can be viewed at any time for ease of access.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully save files as favorites to be view later at any time.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management</li> <li>2) View files of the required cloud drive</li> <li>3) Select any file</li> <li>4) Click on the star icon in the tools bar on top</li> <li>5) The file has been save as favorite successfully.</li> </ol>
Abnormal Path	<ul style="list-style-type: none"> <li>• No files in the connected cloud drive to be saved.</li> <li>• The file already been save as favorite.</li> </ul>
Comments	Favorite files can be removed only from the dedicated favorite files page.

*A - UC Table 3: Favorite Files*

UC 4 : View Favorite Files	
Actor	User
Relation	None
Description	Showing a list of files in dedicated page that already been saved as favorites for ease of access.
Preconditions	Logged in to the system with valid account.
Post-conditions	Successfully show a page listing all the favorite files.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on the quick settings icon on the application top bar.</li> <li>2) Click on the favorite files icon that showed on the pop-up side page.</li> <li>3) List all of the favorite files successfully.</li> </ol>
Abnormal Path	No files has been saved as favorites yet.
Comments	None

*A - UC Table 4: View Favorite Files*

UC 5 : Manage Files & Folders	
Actor	User
Relation	Extended by U6-U10
Description	Ability to perform several management functionalities regarding their files and folders inside the cloud drives.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to perform file & folder management inside the connected cloud drives.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management on the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Select a specific folder or file from the tree view.</li> <li>4) Choose any file management functions from the side panel.</li> <li>5) The functionality performed successfully.</li> </ol>
Abnormal Path	No files have been found in the connected cloud drive.
Comments	None

*A - UC Table 5: Manage Files & Folders*

UC 6 : Rename Files & Folders	
Actor	User
Relation	Extends U5
Description	Ability to rename any folder or files.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive</li> <li>• At least one file or folder already exists in the connected file drive.</li> </ul>
Post-conditions	Successfully allow the user to rename the selected file or folder.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management on the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Select a specific folder or file from the tree view.</li> <li>4) Click on the rename icon from the top tool bar.</li> <li>5) The file/folder renamed successfully.</li> </ol>
Abnormal Path	The new name of the folder or file is already existed.
Comments	None

*A - UC Table 6: Rename Files & Folders*

UC 7 : Download Files	
Actor	User
Relation	Extends U5
Description	Ability to download any file to the user PC.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive</li> <li>• At least one file already exists in the connected file drive.</li> </ul>
Post-conditions	Successfully allow the user to download the desired file on their PC.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management on the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Choose a specific file from the tree view.</li> <li>4) Click on the download icon next to the file.</li> <li>5) Choose a destination on the user PC for the file to be saved.</li> <li>6) The file will be downloaded successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 7: Download Files*

UC 8 : Add Folders	
Actor	User
Relation	Extends U5
Description	Ability to create a new folder inside any connected cloud drive.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive</li> </ul>
Post-conditions	Successfully allow the user to create a new folder.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management on the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Click on the Add Folder icon from top tool bar.</li> <li>4) Enter a folder name and click Confirm.</li> <li>5) The folder created successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 8: Add Folders*

UC 9 : Delete Files & Folders	
Actor	User
Relation	Extends U5
Description	Ability to delete any folder or files.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive</li> <li>• At least one file or folder already exists in the connected file drive.</li> </ul>
Post-conditions	Successfully allow the user to delete the selected file or folder.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management on the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Select a specific folder or file from the tree view.</li> <li>4) Click on the delete icon from top tool bar.</li> <li>5) A confirmation window will appear.</li> <li>6) The user confirms the action.</li> <li>7) The file/folder deleted successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 9: Delete Files & Folders*

UC 10 : Preview Images	
Actor	User
Relation	Extends U5
Description	Show a preview inside the application of Pictures regardless of what format it is PNG, JPG.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive</li> <li>• At least one Image already exists in the connected cloud drive.</li> </ul>
Post-conditions	Successfully show to the user a preview of the desired picture.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management on the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Select a specific picture from the tree view.</li> <li>4) Click on the preview icon from the side panel.</li> <li>5) A preview of the selected picture will show successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 10: Preview images*



UC 11 : Show Statistics	
Actor	User
Relation	None
Description	Shows a details statistical charts and Information about the storage space of the connected cloud drives.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive</li> </ul>
Post-conditions	Successfully show the user all charts and statistics of the connected cloud drives.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on Storage Statistics from the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu to show its statistics or choose the combining storage to show statistics of the sum storage of the all the connected cloud drives.</li> <li>3) The storage statistics will show successfully.</li> </ol>
Abnormal Path	The retrieval of the charts delayed due to a network failure.
Comments	None

*A - UC Table 11: Show Statistics*

UC 12 : Show Drives Profile	
Actor	User
Relation	None
Description	A profile page of any connected cloud drive that shows general information about the cloud drives such as the name of the user, email, profile picture, location and an URL link of main the cloud drive.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully show the user general information about any connected cloud drive.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on Cloud Accounts from the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu</li> <li>3) The information of the chosen cloud drive will show successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 12: Show Drives Profile*

UC 13 : Share Files	
Actor	User
Relation	(Abstract, specialized by U14 and U15)
Description	The user can share files from any connected cloud drives on various social media websites.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• At least one file already exists in the connected cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to share the desired files on the selected social media websites.
Normal Path	None
Abnormal Path	None
Comments	Sharing essentially generate a unique URL to download the shared file, the user has the option to copy the link and save it on a note to access it directly later.

*A - UC Table 13: Share Files*

UC 14 : Single Drive Sharing	
Actor	User
Relation	Includes U17, Specializes U13
Description	The user can share files from a single specific connected cloud drive on various social media websites.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• At least one file already exists in the connected cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to share the desired files on various social media websites.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management from the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Select a specific folder or file from the tree view.</li> <li>4) Click on the share icon from the side panel and a shareable URL link will be generated.</li> <li>5) The user can now click the icon of any social media websites presented.</li> <li>6) The user confirms the share action in the pop-up window.</li> <li>7) The file shared successfully</li> </ol>
Abnormal Path	Unable to share due a cloud service failure which lead to prevention of generating a URL.
Comments	The user can share multiple files only by gathering them into one folder and then sharing the folder.

A - UC Table 14: Single Drive Sharing

UC 15 : Multiple Drive Sharing	
Actor	User
Relation	Includes U17, Specializes U13
Description	The user can share files from multiple connected cloud drive on various social media websites.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• At least one file already exists in the connected cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to share the desired files from multiple cloud drive on various social media websites.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management from the side menu.</li> <li>2) Choose All Drives from the sub menu.</li> <li>3) Select the desired files from the tree views of the connected cloud drives.</li> <li>4) Click on the share icon from the panel and a shareable URL link will be generated.</li> <li>5) The user can now click the icon of any social media websites presented.</li> <li>6) The user confirms the share action in the pop-up window.</li> <li>7) The file shared successfully</li> </ol>
Abnormal Path	Unable to share due a cloud service failure which lead to prevention of generating a URL.
Comments	None

*A - UC Table 15: Multiple Drive Sharing*

UC 16 : Set Password	
Actor	User
Relation	Extends U1
Description	Allows The user who logged in by external logins to set a password which will provide more security to the user account.
Preconditions	<ul style="list-style-type: none"> <li>Logged in to the system with valid account by one of the external logins.</li> </ul>
Post-conditions	Successfully allow the user set a new password.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on My Accounts from the side menu.</li> <li>2) Choose Link Accounts from the sub menu.</li> <li>3) Press on Set a New Password button next to Password.</li> <li>4) Write down the desired password</li> <li>5) Rewrite the password again to confirm it and click on OK button.</li> <li>6) The password will be set successfully</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 16: Set Password*

UC 17 : Get Link	
Actor	User
Relation	Included by U14 & U15 , Extended by U19 & U36
Description	It will automatically generates a shareable link of a certain file or folder to copied or share on social media.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• At least one file already exists in the connected cloud drive.</li> </ul>
Post-conditions	Successfully generates a sharable link.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management from the side menu.</li> <li>2) Choose All Drives from the sub menu.</li> <li>3) Select the desired files from the tree views of the connected cloud drives.</li> <li>4) Click on the share icon from the panel</li> <li>5) The shareable URL link will be generated successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 17: Get Link*

UC 18 : Reset Password	
Actor	User
Relation	Extends U1
Description	Allows The user to reset the existing password and change it to a new one in case the user forgets the old one or wanted to renew it.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> </ul>
Post-conditions	Successfully allow the user reset his password with a new one.
Normal Path	<ol style="list-style-type: none"> <li>7) Click on My Accounts from the side menu.</li> <li>8) Choose Link Accounts from the sub menu.</li> <li>9) Press on Reset Password button next to Password.</li> <li>10) Enter the old password</li> <li>11) Write down the new password</li> <li>12) Rewrite the password again to confirm it and click on OK button.</li> <li>13) The password will be reset successfully</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 18: Reset Password*



UC 19 : Share on Social Media	
Actor	User
Relation	Extends U17
Description	The user can share the shareable URL link of the selected files or folder on various social media websites.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• At least one file already exists in the connected cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to share the shareable URL link of the desired files and folders on various social media websites.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management from the side menu.</li> <li>2) Choose All Drives from the sub menu.</li> <li>3) Select the desired files or folders from the tree view of the connected cloud drive.</li> <li>4) Click on the share icon from the panel and a shareable URL link will be generated.</li> <li>5) The user can now selects the icon of any social media websites presented.</li> <li>6) The user confirms the share action in the pop-up window.</li> <li>7) The link shared successfully</li> </ol>
Abnormal Path	The user doesn't have an account in social media selected.
Comments	None

*A - UC Table 19: Share on Social Media*

UC 20 : Authenticate by Phone Number	
Actor	User
Relation	Extends U1
Description	Allows The user to authenticate his account by his personal phone number which will provide more security to the user account.
Preconditions	Logged in to the system with valid account.
Post-conditions	Successfully allow the user to authenticate his account by his personal phone number.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on My Accounts from the side menu.</li> <li>2) Choose Link Accounts from the sub menu.</li> <li>3) Press on Activate button next to authenticate by phone number.</li> <li>4) Enter a valid phone number and clicks OK.</li> <li>5) The user will receive an SMS message to the entered phone number contains a security digit code</li> <li>6) The user will enter the digit code into the required filed in the application and click OK.</li> <li>7) The Account is now verified with the user phone number successfully.</li> </ol>
Abnormal Path	<ul style="list-style-type: none"> <li>• The phone number entered is not valid.</li> <li>• The code entered is not correct.</li> </ul>
Comments	After successfully activate phone authentication, each time the user tries to login an SMS message will be sent to him with the code which will be asked to enter in order to access the application.

*A - UC Table 20: Authenticate by Phone Number*

UC 21 : Upload Files	
Actor	User
Relation	None
Description	The user can upload any type of files into any connected cloud drive.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• Enough free space inside the connected drive.</li> </ul>
Post-conditions	Successfully allow the user to upload the desired files inside any connected cloud drive.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on file management from the side menu.</li> <li>2) Choose the desired connected cloud drive from the sub menu.</li> <li>3) Select a specific folder from the tree view.</li> <li>4) Click on Browse button from the side panel.</li> <li>5) Selected the desired files from your PC and press OK button from the browse pop-up window.</li> <li>6) Click on Upload button.</li> <li>7) The file will be uploaded successfully.</li> </ol>
Abnormal Path	The uploaded file already exist with the same type and name.
Comments	If the user connects the original cloud drive to his PC as desktop version, and then uploads files inside that cloud drive using our application, MS Windows will show a real-time notification of the new uploaded files.

*A - UC Table 21: Upload Files*

UC 22 : Search for Files or Folders	
Actor	User
Relation	Includes U23
Description	Allowing the user to search for any file or folder by its name in any of the connected cloud drives.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to search for file or folder in the connected cloud drives.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on Search icon on the application top bar.</li> <li>2) The user will enter the name of the file or folder in the search bar and click Enter.</li> <li>3) The application will redirect the user to search result page.</li> <li>4) The user successfully searched for the file or folder desired.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 22: Search for Files or Folders*

UC 23 : View Result	
Actor	User
Relation	Included by U22
Description	View files or folders that matches what the user searched for in a page.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• Already search for file or folder in search bar.</li> </ul>
Post-conditions	Successfully shows to the user the result of the searched file or folder in a page.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on Search icon on the application top bar.</li> <li>2) The user will enter the name of the file or folder in the search bar and click Enter.</li> <li>3) The application will redirect the user to search result page.</li> <li>4) The page will contains a list of files or folder that matches the word the user searched for.</li> <li>5) The user will successfully see the list of files or folders of what he searched for.</li> </ol>
Abnormal Path	Shows nothing is search result due to incorrect typing of the name of the file.
Comments	None

*A - UC Table 23: View Result*

UC 24 : Add Cloud Drive	
Actor	User
Relation	Includes U25
Description	The user can add connected cloud drive that are available which are Google Drive, Dropbox and OneDrive
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already have a valid account in the mentioned cloud drives.</li> </ul>
Post-conditions	Successfully allow the user to add connected cloud drive to his MCloud account.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on My Accounts from the side menu.</li> <li>2) Choose Link Accounts from the sub menu.</li> <li>3) Press on Link Accounts button next to Social Media Accounts.</li> <li>4) Click add button next to the cloud drive the user wish to add.</li> <li>5) The selected cloud drive will be added after completing the linking process successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 24: Add Cloud Drive*

UC 25 : Connect Cloud Drive	
Actor	User
Relation	Includes U26, Included by U24 , Extends U1
Description	The user will connect his account with any of the available and previously selected cloud drive in order to authenticate and have a direct connection between MCloud account and the cloud drive.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already have a valid account in the selected cloud drives.</li> </ul>
Post-conditions	Successfully complete the authentication and connection process between MCloud account and the selected cloud drive.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on My Accounts from the side menu.</li> <li>2) Choose Link Accounts from the sub menu.</li> <li>3) Press on Link Accounts button next to Social Media Accounts.</li> <li>4) Click add button next to the cloud drive the user wish to add.</li> <li>5) Login to the cloud drive you selected in the pop-up window showed</li> <li>6) Click Allow</li> <li>7) The connection now is completed successfully.</li> </ol>
Abnormal Path	The user didn't allow the authentication.
Comments	None.

A - UC Table 25: Connect Cloud Drive

UC 26 : Sync Data	
Actor	User
Relation	Includes U26, Included by U24
Description	Once the user selects one of the available cloud drives to add and then allow the authentication, all the files and folders in that cloud drive will be synced with the MCloud account of the user.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already have a valid account in the selected cloud drives.</li> </ul>
Post-conditions	Successfully complete the syncing process between MCloud account and the selected cloud drive and retrieve all the files and folders to be accessed by the MCloud application.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on My Accounts from the side menu.</li> <li>2) Choose Link Accounts from the sub menu.</li> <li>3) Press on Link Accounts button next to Social Media Accounts.</li> <li>4) Click add button next to the cloud drive the user wish to add.</li> <li>5) Login to the cloud drive you selected in the pop-up window showed</li> <li>6) Click Allow</li> <li>7) The connection now is completed</li> <li>8) The files and folder are now synced successfully.</li> </ol>
Abnormal Path	The user didn't allow the authentication.
Comments	Once the syncing is completed, any change in the original account of the cloud drive will be affect MCloud account as well and vice versa, for example if the user created a folder in the original account of Dropbox and then he open his Dropbox in MCloud he will see the folder there also.

A - UC Table 26: Sync Data



UC 27 : Remove Connected Drive	
Actor	User
Relation	None
Description	The user can remove any connected cloud drive easily and the cloud drive will be disconnected and thus won't be accessible.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to discontent and remove any connected cloud drive.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on My Accounts from the side menu.</li> <li>2) Choose Link Accounts from the sub menu.</li> <li>3) Press on Link Accounts button next to Social Media Accounts</li> <li>4) Click Remove button next to the cloud drive the user wish to remove and disconnect.</li> <li>5) The user will confirm the action.</li> <li>6) The connected cloud drive will be removed successfully from the system.</li> </ol>
Abnormal Path	Unable to discontent because the cloud drive needed to be removed is the only one which used to login.
Comments	None

*A - UC Table 27: Remove Connected Drive*

UC 28 : Send Direct Message	
Actor	Admin
Relation	None
Description	The admin can send direct messages to any of the registered users for help and support purposes.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> </ul>
Post-conditions	Successfully allow the Admin to send direct messages to any registered user.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on User Management from the side menu.</li> <li>2) Search for a particular user or choose one from the list of the users available.</li> <li>3) Click on the message icon from the user profile.</li> <li>4) The admin will type a message.</li> <li>5) The admin will click on Send Message.</li> <li>6) The message will be sent to the user successfully.</li> </ol>
Abnormal Path	The message box was empty and contains nothing when sent
Comments	None

*A - UC Table 28: Send Direct Message*

UC 29 : Create an Account	
Actor	Visitor
Relation	None
Description	The Visitor will be able to sign up by filling the registration form to become a registered user in order to have access to the application.
Preconditions	<ul style="list-style-type: none"> <li>• An valid email address</li> <li>• Validated password</li> </ul>
Post-conditions	Successfully allow the visitor to become a registered user with a valid account.
Normal Path	<ol style="list-style-type: none"> <li>1) Open the website</li> <li>2) Click on Start with MCloud button</li> <li>3) Click on Create new Account</li> <li>4) Fill the form with a valid Email address and valid strong password</li> <li>5) Click on Create Account</li> <li>6) The visitor will be asked to verify his account</li> <li>7) The visitor goes to his email account</li> <li>8) An email form MCloud will ask his to click on the verification link</li> <li>9) The visitor clicks on the link</li> <li>10) The visitor successfully completed the registration process and he is now a verified user.</li> </ol>
Abnormal Path	<p>The email entered is already used</p> <p>The password is not strong enough</p> <p>The visitor did not verify his email address</p>
Comments	None

*A - UC Table 29: Create an Account*

UC 30 : View the Website	
Actor	Admin, User and Visitor
Relation	None
Description	The main website landing page which will show the main features MCloud are providing and also view the latest news regarding the application.
Preconditions	<ul style="list-style-type: none"> <li>• Web Browser</li> <li>• Stable Internet connection</li> </ul>
Post-conditions	Successfully allow anyone to open the website and view what it contains
Normal Path	<ol style="list-style-type: none"> <li>1) Enter the Website URL</li> <li>2) Successfully open the website.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 30: View the Website*

UC 31 : View Direct Messages	
Actor	User
Relation	None
Description	Quickly views the latest messages received from the admin.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> </ul>
Post-conditions	Successfully allow the user to receive direct messages from the admin.
Normal Path	5) Click on Direct Messages icon on the application top bar. 6) View the latest messages successfully.
Abnormal Path	None
Comments	Currently the user cannot send direct messages to the admin, but he will be able to in future versions, he can only now view the messages sent by the admin which are mostly tips and news about the application.

*A - UC Table 31: View Direct Messages*

UC 32 : View Settings	
Actor	User
Relation	None
Description	Showing a summarized Information about the storage of drives for quick access.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully view a page showing summarized charts and information of the connected drives.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on the quick settings icon on the application top bar.</li> <li>2) View setting with the information successfully.</li> </ol>
Abnormal Path	None
Comments	More detailed statistics and charts can be shown in a different feature, this is only a summery.

A - UC Table 32: View Settings

UC 33 : View Tasks	
Actor	User
Relation	None
Description	Quickly views the latest tasks that user was planning in doing later regarding the cloud drives.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully allow the user to view the latest tasks.
Normal Path	<ol style="list-style-type: none"> <li>1) Click on Tasks icon on the application top bar.</li> <li>2) View list of latest tasks successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 33: View Tasks*

UC 34 : Receive Notifications	
Actor	User
Relation	None
Description	The user can receive notification about any action that needs the attention or the latest activities occurred regarding his cloud drives.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully allow receiving notifications for the user
Normal Path	None
Abnormal Path	None
Comments	None

*A - UC Table 34: Receive Notifications*



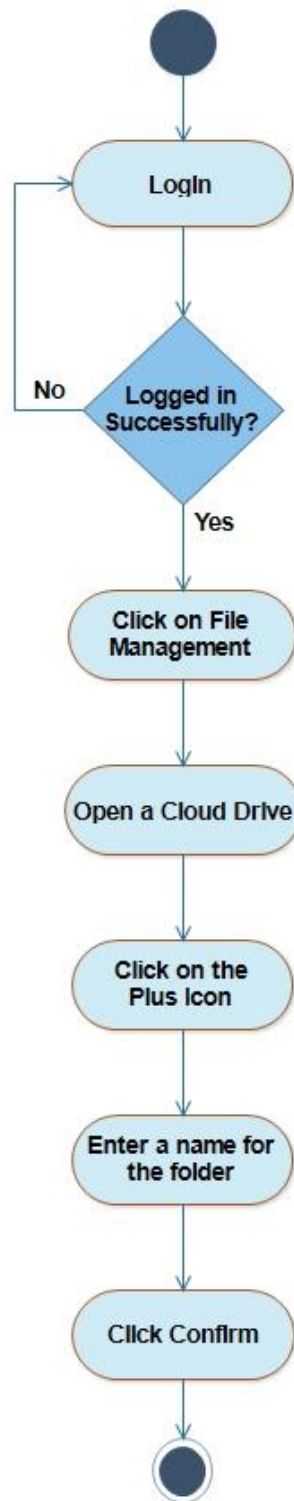
UC 35 : View Notifications	
Actor	User
Relation	None
Description	Quickly views the latest notification about any action that needs the attention or the latest activities occurred regarding his cloud drives.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> </ul>
Post-conditions	Successfully view the notifications for the user
Normal Path	<ol style="list-style-type: none"> <li>1) Click on Notifications icon on the application top bar.</li> <li>2) View list of latest notifications successfully.</li> </ol>
Abnormal Path	None
Comments	None

*A - UC Table 35: View Notifications*

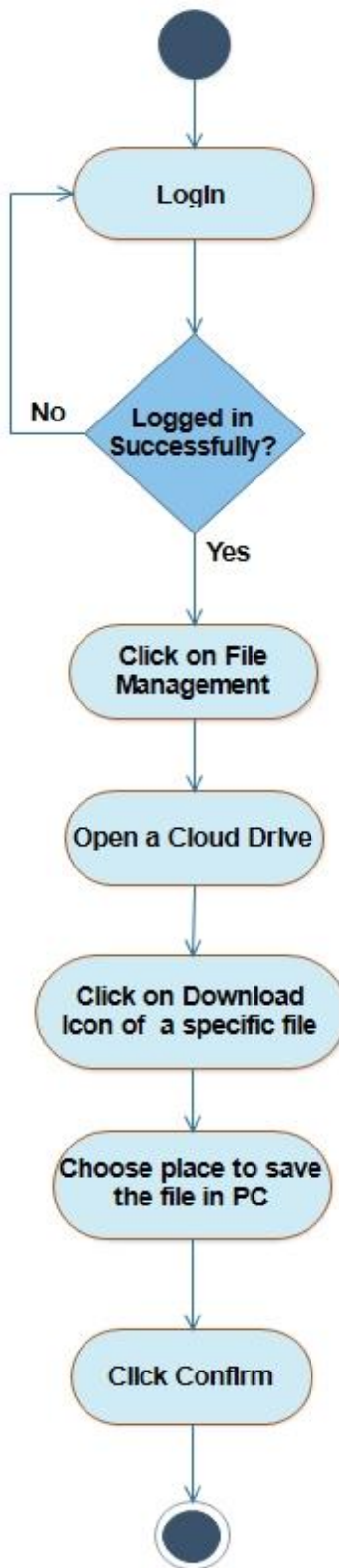
UC 36 : Copy to Clipboard	
Actor	User
Relation	Extends U17
Description	It will automatically copy the shareable URL link to the clipboard to be paste anywhere.
Preconditions	<ul style="list-style-type: none"> <li>• Logged in to the system with valid account.</li> <li>• Already connected to at least one cloud drive.</li> <li>• At least one file already exists in the connected cloud drive.</li> </ul>
Post-conditions	Successfully allow the user set a new password.
Normal Path	<ol style="list-style-type: none"> <li>6) Click on My Accounts from the side menu.</li> <li>7) Choose Link Accounts from the sub menu.</li> <li>8) Press on Set a New Password button next to Password.</li> <li>9) Write down the desired password</li> <li>10) Rewrite the password again to confirm it and click on OK button.</li> <li>11) The password will be set successfully</li> </ol>
Abnormal Path	Unable to share due a cloud service failure which lead to prevention of generating a URL.
Comments	The user can share multiple files only by gathering them into one folder and then sharing the folder.

*A - UC Table 36: Copy to Clipboard*

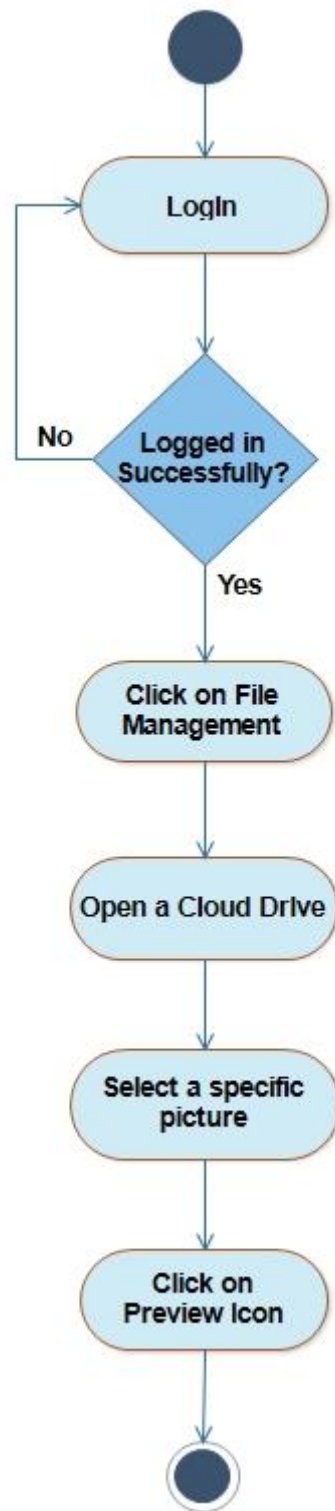
# Activity Diagrams



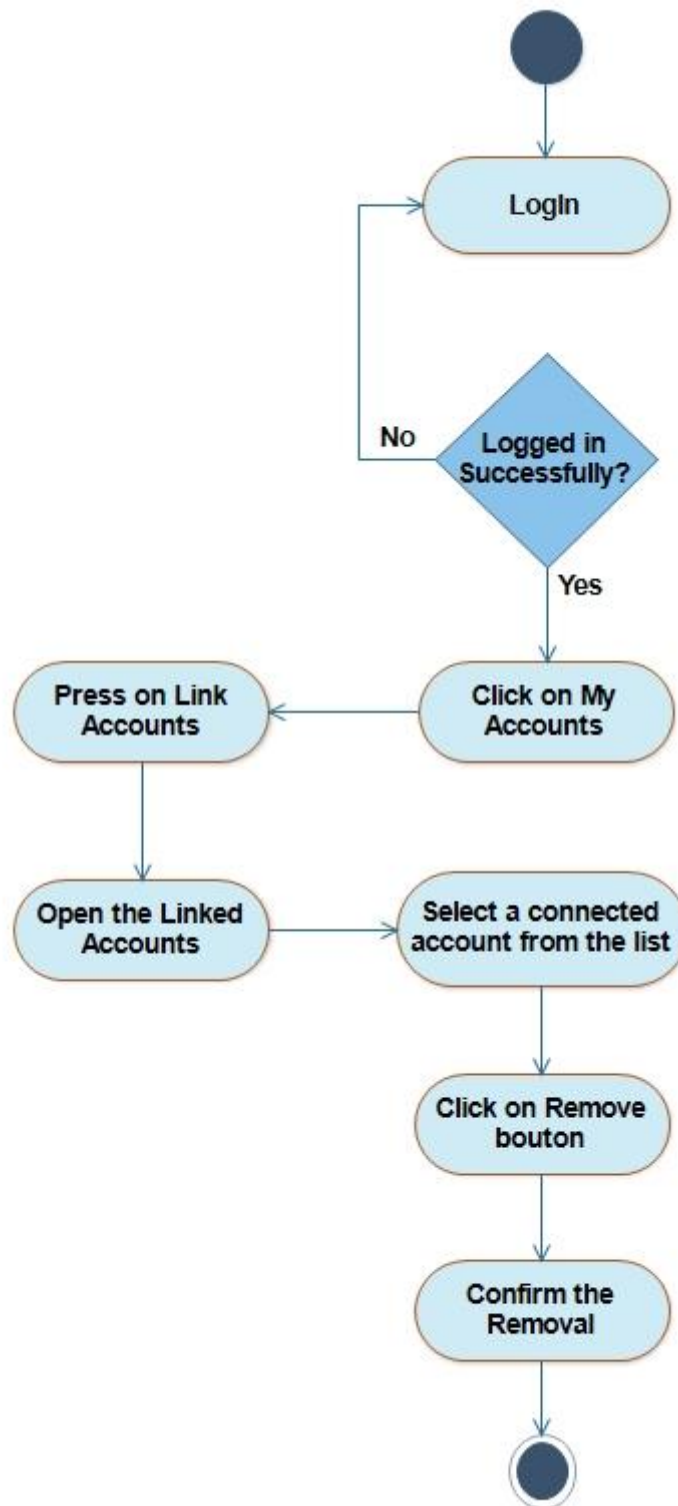
A - Activity Diagram 1: Add Folder



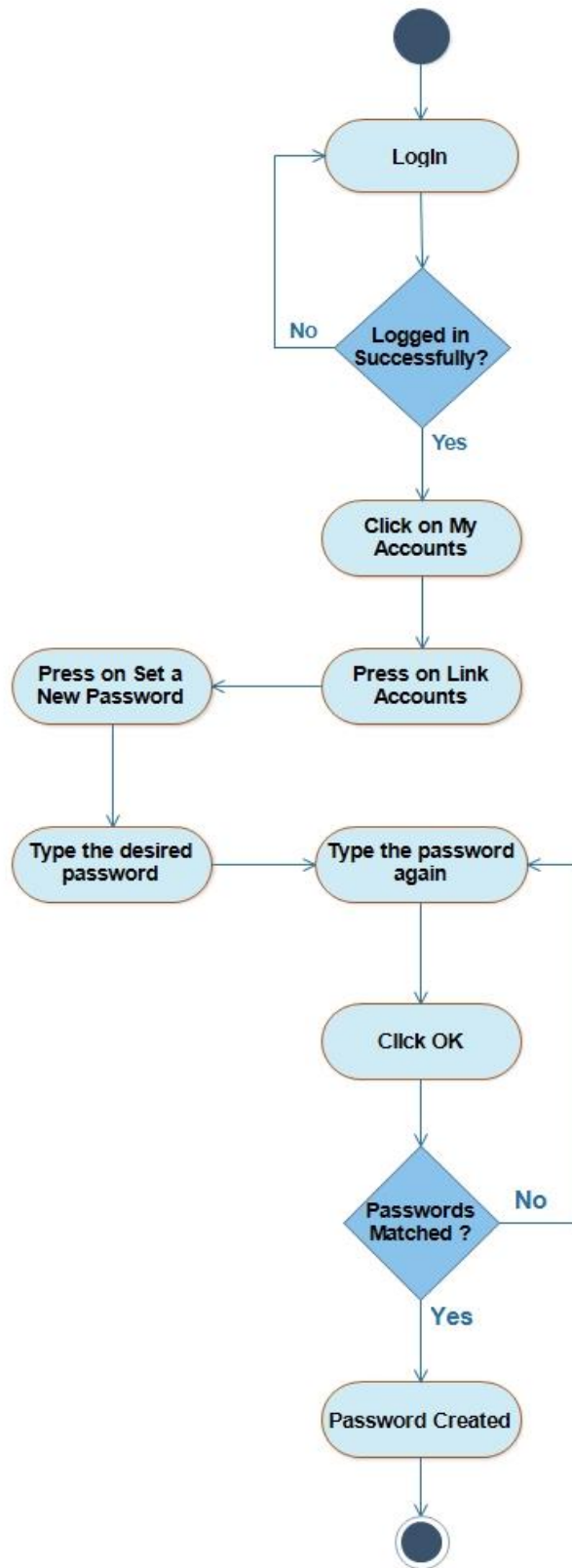
A - Activity Diagram 2: Download Files



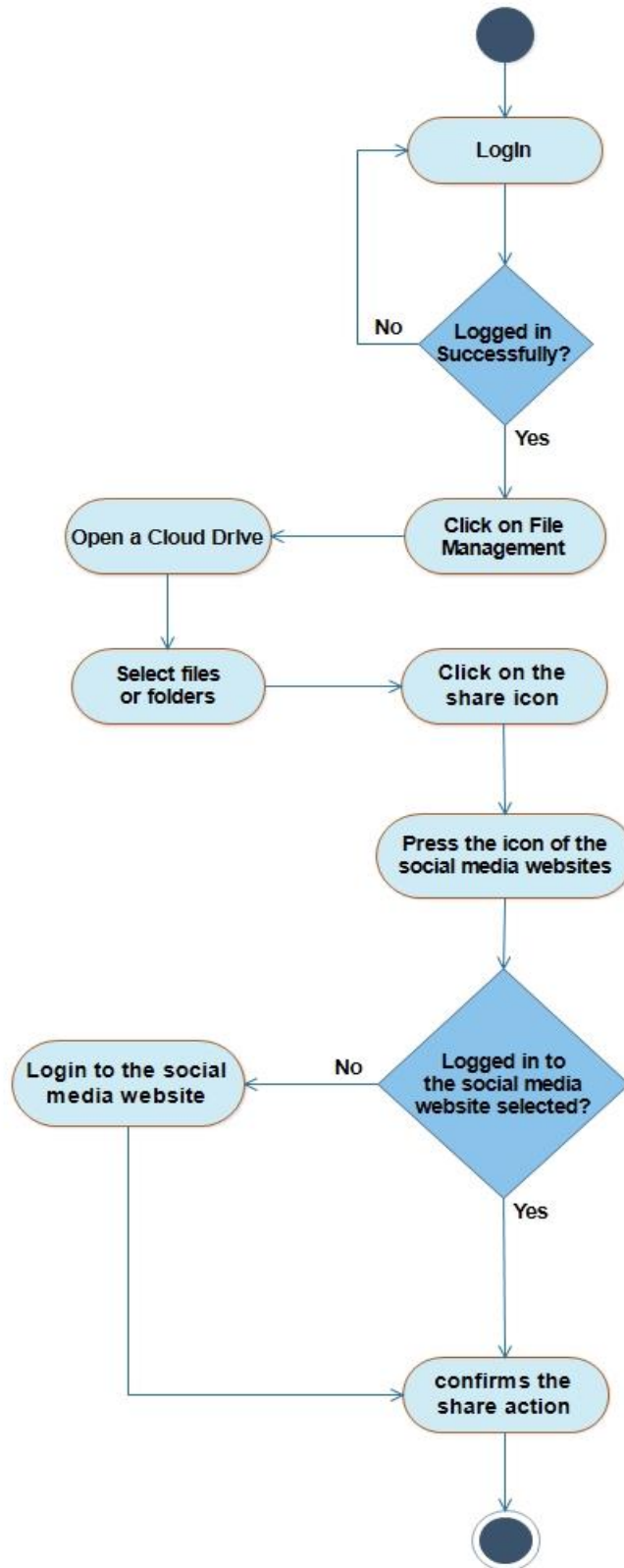
A - Activity Diagram 3: Preview Images



A - Activity Diagram 4: Remove Cloud Drive

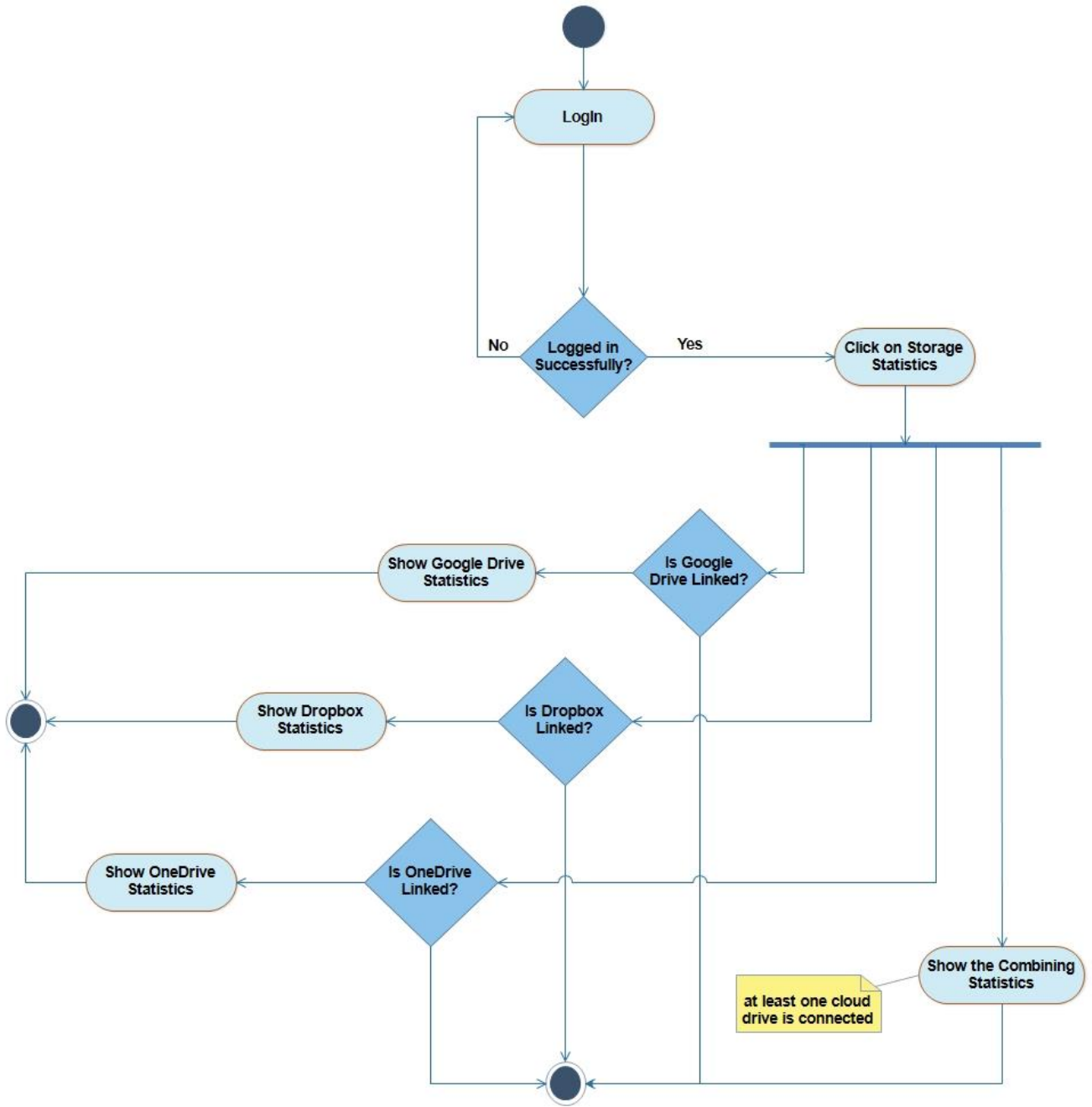


A - Activity Diagram 5: Set Password

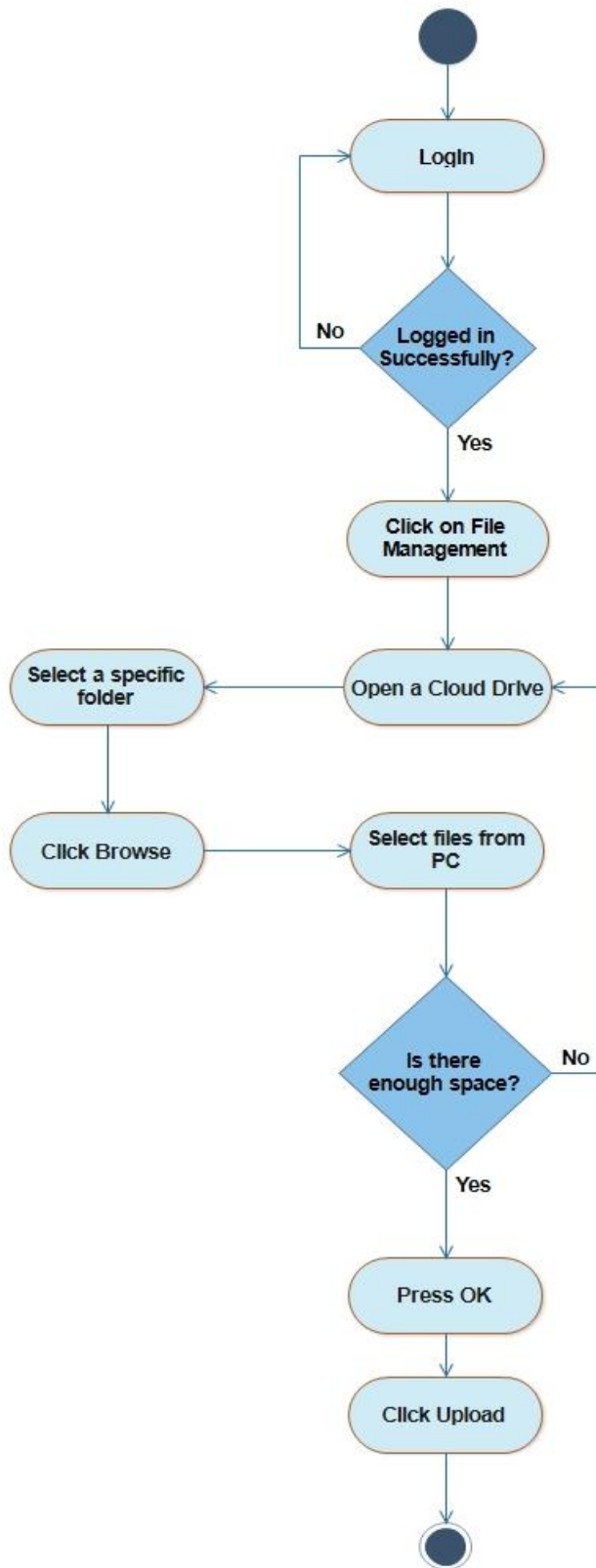


A - Activity Diagram 6: Share on Social Media



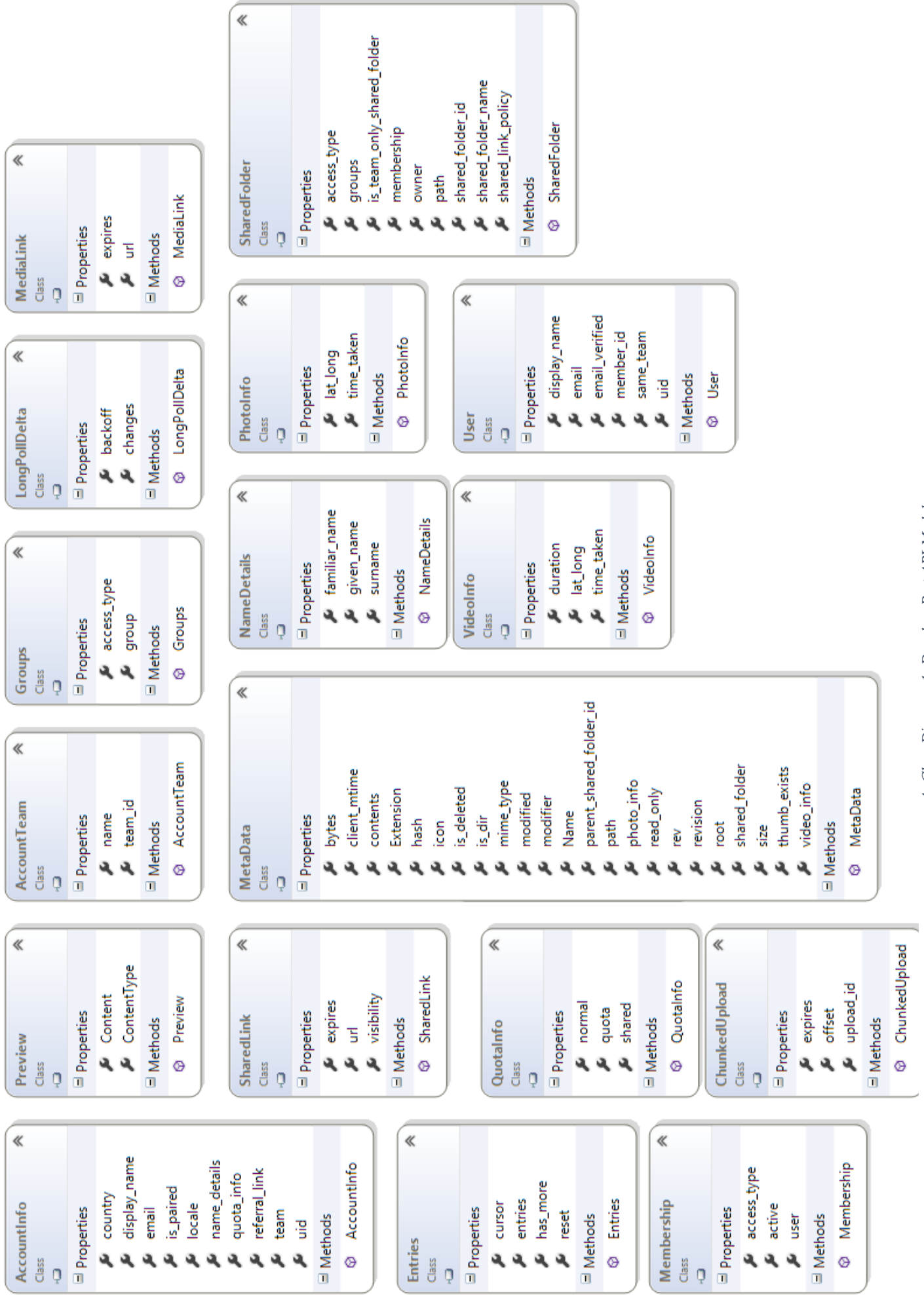


A - Activity Diagram 7: Show Statistics

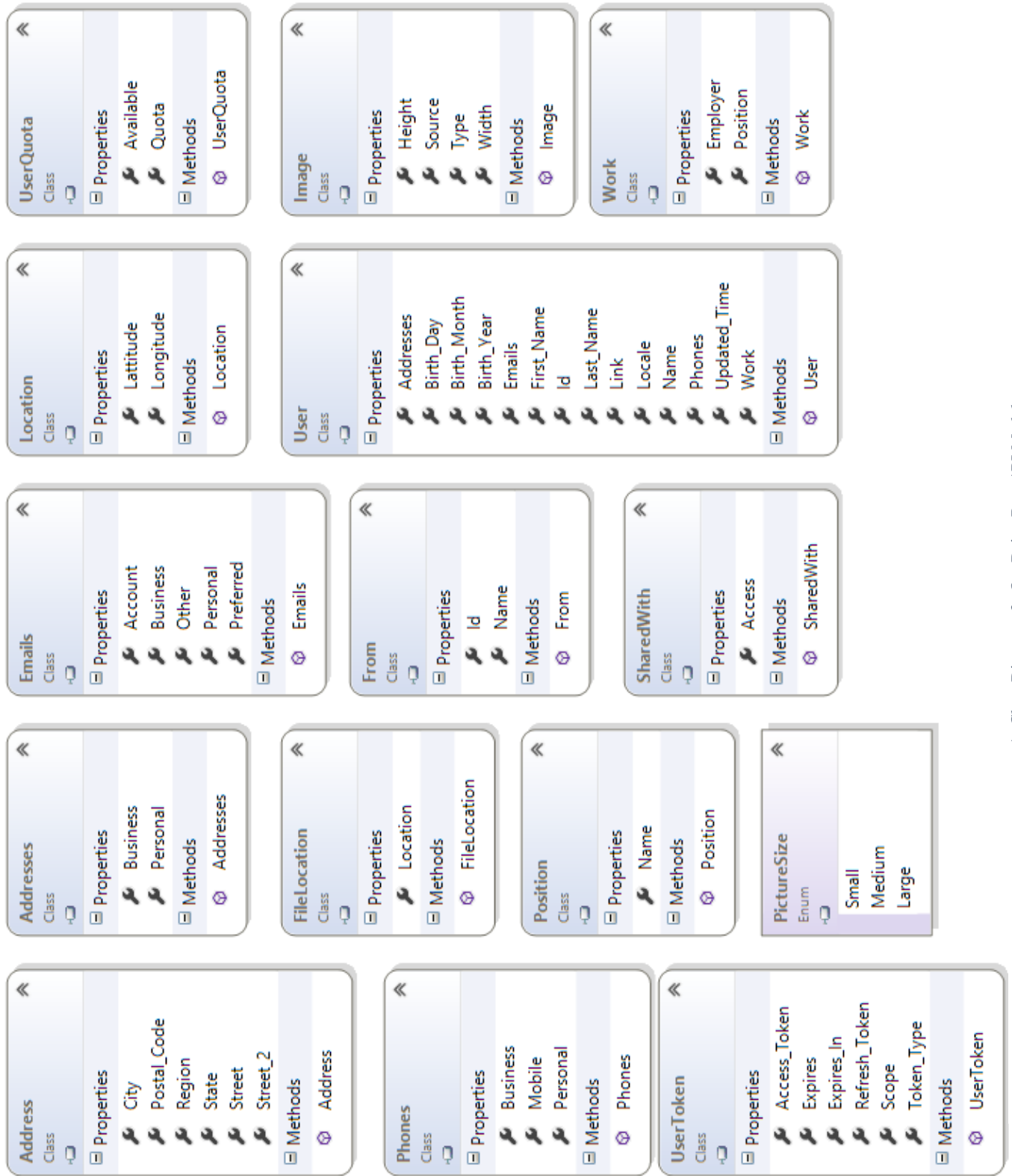


A - Activity Diagram 8: Upload Files

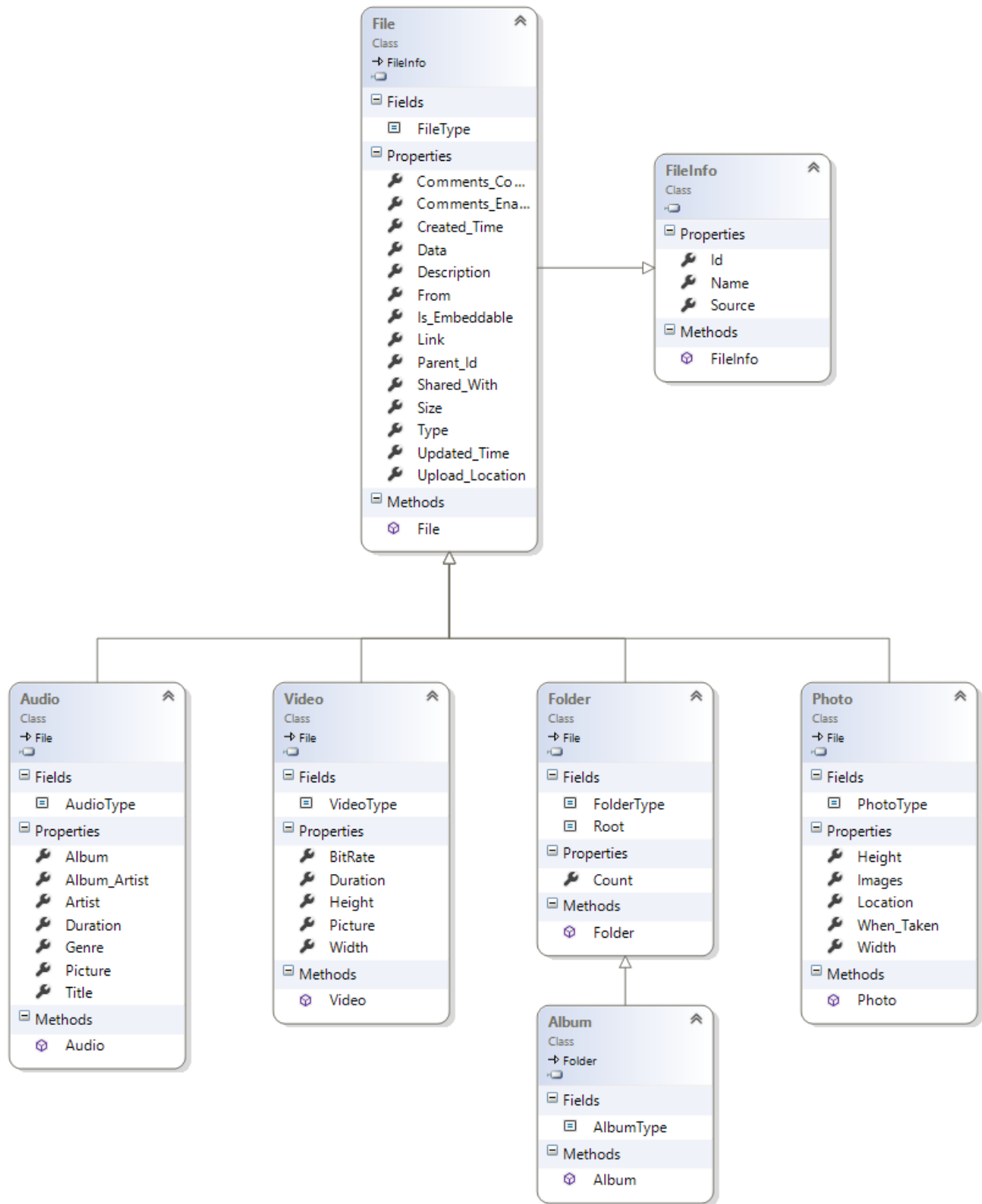
# Class Diagrams



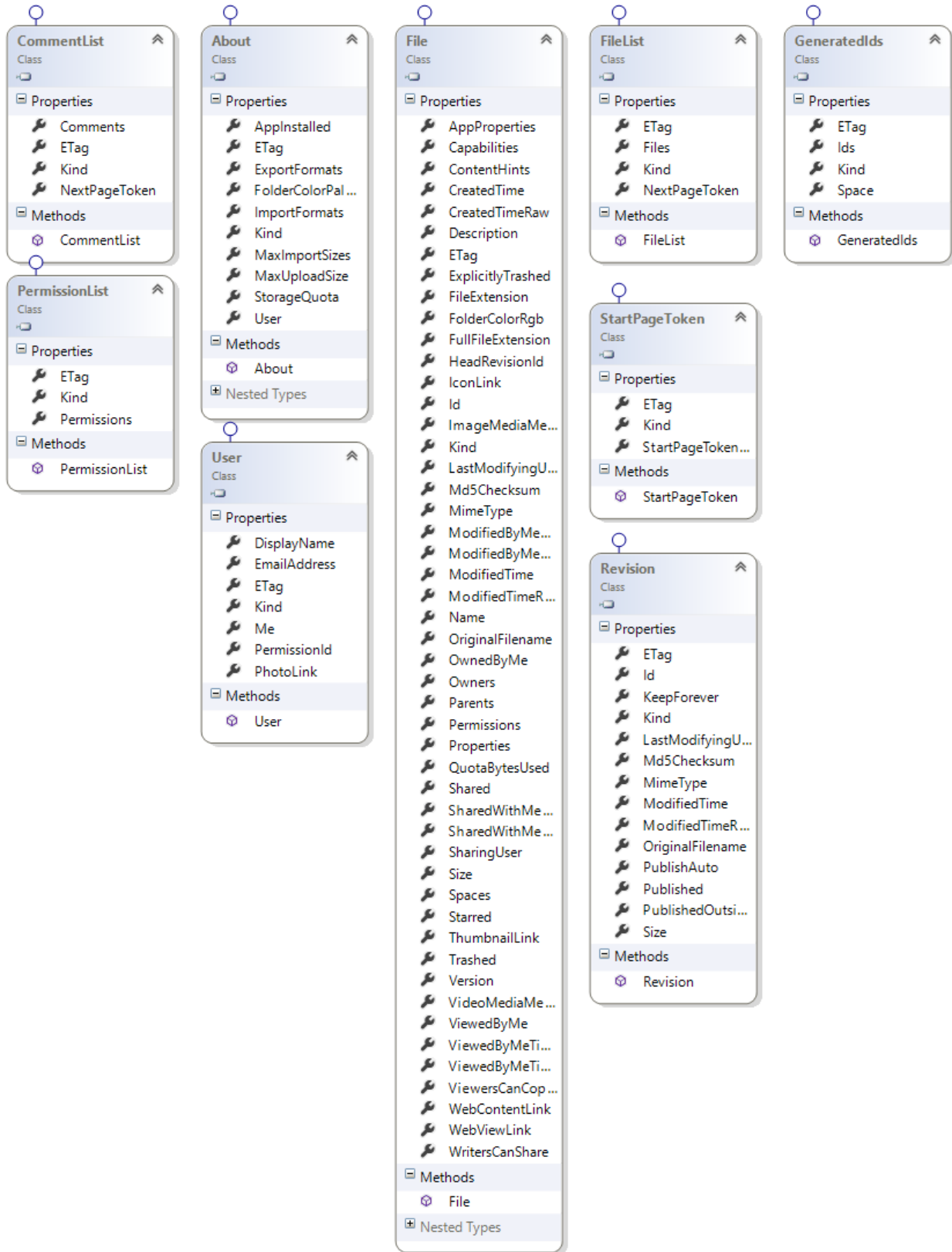
A-Class Diagram 1: Dropbox Rest API Models



A-Class Diagram 2: OneDrive Rest API Model



A- Class Diagram 3: File Model Class Diagram



A- Class Diagram 4: Google Drive API v3.0




A- Class Diagram 5: AppStart






A- Class Diagram 6: ASPSMSX2Spap


# Database Tables Detailed Description

MyUsers			
	Column Name	Data Type	Allow Nulls
	Id	nvarchar(128)	<input type="checkbox"/>
	Email	nvarchar(256)	<input checked="" type="checkbox"/>
	EmailConfirmed	bit	<input type="checkbox"/>
	PasswordHash	nvarchar(MAX)	<input checked="" type="checkbox"/>
	SecurityStamp	nvarchar(MAX)	<input checked="" type="checkbox"/>
	PhoneNumber	nvarchar(MAX)	<input checked="" type="checkbox"/>
	PhoneNumberConfirmed	bit	<input type="checkbox"/>
	TwoFactorEnabled	bit	<input type="checkbox"/>
	LockoutEndDateUtc	datetime	<input checked="" type="checkbox"/>
	LockoutEnabled	bit	<input type="checkbox"/>
	AccessFailedCount	int	<input type="checkbox"/>
	UserName	nvarchar(256)	<input type="checkbox"/>
			<input type="checkbox"/>

A - DB Table Description 1: MyUsers

MyUserClaims			
	Column Name	Data Type	Allow Nulls
	Id	int	<input type="checkbox"/>
	UserId	nvarchar(128)	<input type="checkbox"/>
	ClaimType	nvarchar(MAX)	<input checked="" type="checkbox"/>
	ClaimValue	nvarchar(MAX)	<input type="checkbox"/>
			<input type="checkbox"/>

A - DB Table Description 2: MyUserClaims

MyRoles			
	Column Name	Data Type	Allow Nulls
	Id	nvarchar(128)	<input type="checkbox"/>
	Name	nvarchar(256)	<input type="checkbox"/>
			<input type="checkbox"/>

A - DB Table Description 3: MyRoles

MyUserLogins			
	Column Name	Data Type	Allow Nulls
🔑	UserId	nvarchar(128)	☐
🔑	LoginProvider	nvarchar(128)	☐
🔑	ProviderKey	nvarchar(128)	☐
			☐

A - DB Table Description 4: MyUserLogins

MigrationHistory			
	Column Name	Data Type	Allow Nulls
🔑	MigrationId	nvarchar(150)	☐ <u>__MigrationHistory</u>
🔑	ContextKey	nvarchar(300)	☐
	Model	varbinary(MAX)	☐
	ProductVersion	nvarchar(32)	☐
			☐

A - DB Table Description 5: MigrationHistory

UserId	LoginProvider	ProviderKey
04cb066f-6b8e-4305-9877-d347ec575ae1	Dropbox	524507955
04cb066f-6b8e-4305-9877-d347ec575ae1	Facebook	10204351141537955
04cb066f-6b8e-4305-9877-d347ec575ae1	GitHub	15429511
04cb066f-6b8e-4305-9877-d347ec575ae1	Google	118011818444087776057
04cb066f-6b8e-4305-9877-d347ec575ae1	Instagram	2087934230
04cb066f-6b8e-4305-9877-d347ec575ae1	LinkedIn	XX_NECx3Bs
04cb066f-6b8e-4305-9877-d347ec575ae1	Microsoft	81c8f4b13871e5b5
04cb066f-6b8e-4305-9877-d347ec575ae1	Twitter	790821392

A - DB Table 1: MyUserLogins

Id	UserId	ClaimType	ClaimValue
113	04cb066f-6b8e-4305-9877-d347ec575ae1	Dropbox_AccessToken	ulf6EEOVH2AAAAAAAAAAmW...
114	04cb066f-6b8e-4305-9877-d347ec575ae1	DropboxUserId	524507955
115	04cb066f-6b8e-4305-9877-d347ec575ae1	DropboxTokenIssuedAt	-8587407607412473511
116	04cb066f-6b8e-4305-9877-d347ec575ae1	DropboxAccountName	Mohammed Fayez
189	04cb066f-6b8e-4305-9877-d347ec575ae1	Microsoft_AccessToken	EwCIAq1DBA AUGCCXc8wU/zF...
190	04cb066f-6b8e-4305-9877-d347ec575ae1	MicrosoftRefreshToken	DWH*9hRRRKbXW4m*QVFZKvP05
191	04cb066f-6b8e-4305-9877-d347ec575ae1	MicrosoftUserId	81c8f4b13871e5b5
192	04cb066f-6b8e-4305-9877-d347ec575ae1	MicrosoftTokenIssuedAt	-8587403366125688360
193	04cb066f-6b8e-4305-9877-d347ec575ae1	MicrosoftTokenExpiresIn	3600
194	04cb066f-6b8e-4305-9877-d347ec575ae1	MicrosoftAccountName	Mohammed Fayez
209	04cb066f-6b8e-4305-9877-d347ec575ae1	Google_AccessToken	ya29..zAJC4NpA34SO7zSwjoD...
210	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleRefreshToken	1/vvevQO_mirmYeW_ud42g8-...
211	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleUserId	118011818444087776057
212	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleTokenIssuedAt	-8587402463823374903
213	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleTokenExpiresIn	3600
214	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleAccountName	Mohammed Fayez Fayez
215	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleAccountProfile	https://plus.google.com/11801...
216	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleAccountPicture	https://lh4.googleusercontent.c...
217	04cb066f-6b8e-4305-9877-d347ec575ae1	GoogleAccountEmail	engmohfh@gmail.com

A - DB Table 2: MyUserClaims