### תיאור קצר של הפיצ'רים:

My Albums

פיצ'ר זה מאפשר למשתמש לדפדף ברשימת התמונות הקיימות באלבום ולבחור בקלות ע"י סימון התמונות הרצויות מהאלבום ולאחר בחירת נתיב ההורדה במחשב, התמונות המסומנות ירדו יחד.

שיפור נוסף למשתמש יש אפשרות סינון של התמונות באלבום על פי מאפיינים : כמות לייקים, כמות תגובות ולפי תאריך העלאה.

Video Streamer

פיצ'ר זה מאפשר למשתמש לצפות בסרטונים הקיימים על קיר המשתמש באמצעות נגן אפליקטיבי ע"י בחירתם מהפלייליסט ומאפשר למשתמש חווית צפייה נוחה

שיפור נוסף הורדת הסרטונים מהפלייליסט למחשב בצורה נוחה.

### תבנית מס' 1 – Adapter Pattern

מטרה:

לייצור אחידות עבור פעולת ההתחברות של המשתמש ל- API.

בעזרת תבנית זו ניתן להפעיל את המתודה Login כפקודה פשוטה ללא פרמטרים במקום להפעילה עם ריבוי פרמטרים ספציפיים ל- Facebook API הנתון.

בנוסף לכך במקרה ונרצה לחברAPI של רשת חברתית אחרת כגון Linkedin, נוסיף לו Adapter נוסף אך לא נצטרך להחליף צורת התחברות בקליינט.

תפקידים:

ה- Client הוא: AppMainForm

ה- Adapter הוא: FacebookService

ה- ClassAdapter הוא: UserLoginAdapter

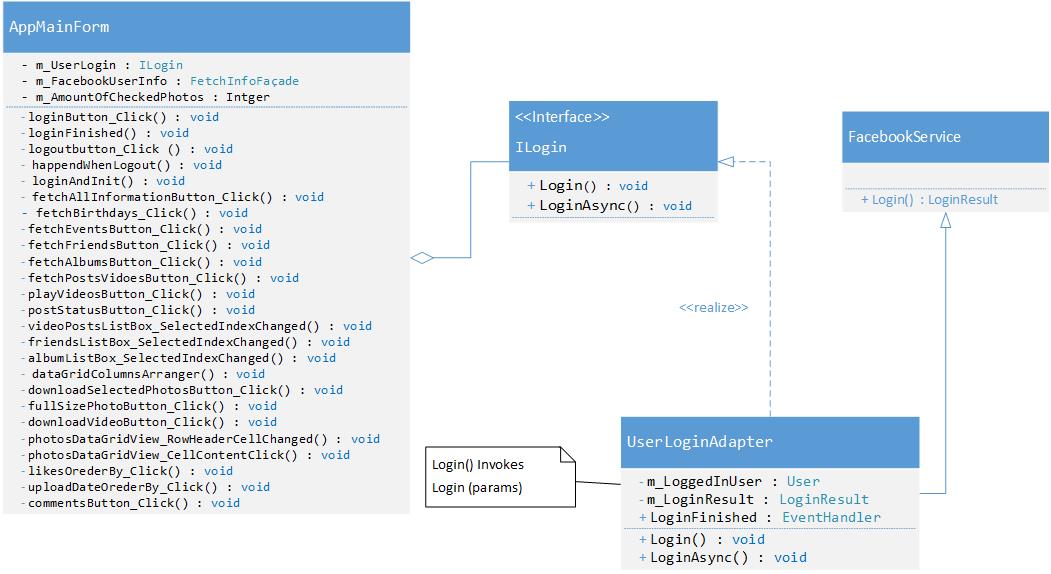
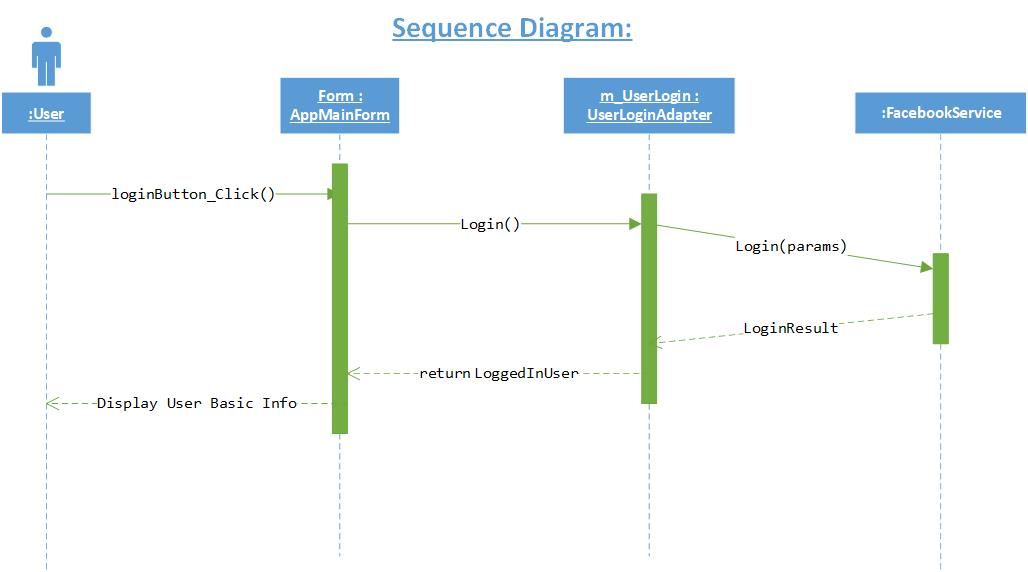
מימוש:

המימוש של Class Adapterניתן למצוא בתיקייה של LoginAdapter בקובץ:

LoginAdapter \UserLoginAdapter.cs

מימוש של UserLoginAdapter באמצעות המתודה Login() שנמצא במתודה LoginAsync()

A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\ AppMainForm.cs



תבנית מס' 2 - Singleton Pattern

סיבת הבחירה / שימוש בתבנית:

באפליקציה שלנו יש אובייקט FetchInfoFaçade שמכיל את הלוגיקה של האפליקציה. החלטנו להפוך אותו לסינגלטון מכיוון שאנו לא רוצים שיהיה יותר מאחד כזה ומכיוון שיש המון גישות אליו לאורך חיי האפליקציה – עבור כל פעולה שהמשתמש באפליקציה מבצע יש קריאה לאובייקט הנ"ל.

אופן המימוש:

יצרתי את המחלקה Singleton<T> כך שבעזרתה נוצר Instance אחד לכל היותר של

.FetchInfoFaçade

הפכנו את המחלקה FetchInfoFaçade למחלקה שנעולה לירושה, והפכנו את ה constructor ל private .

תפקדים:

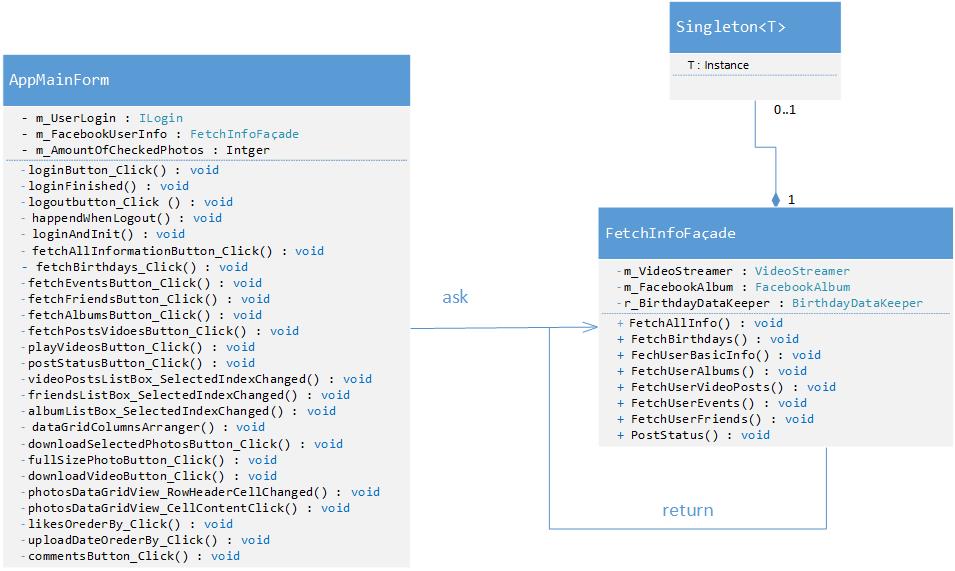
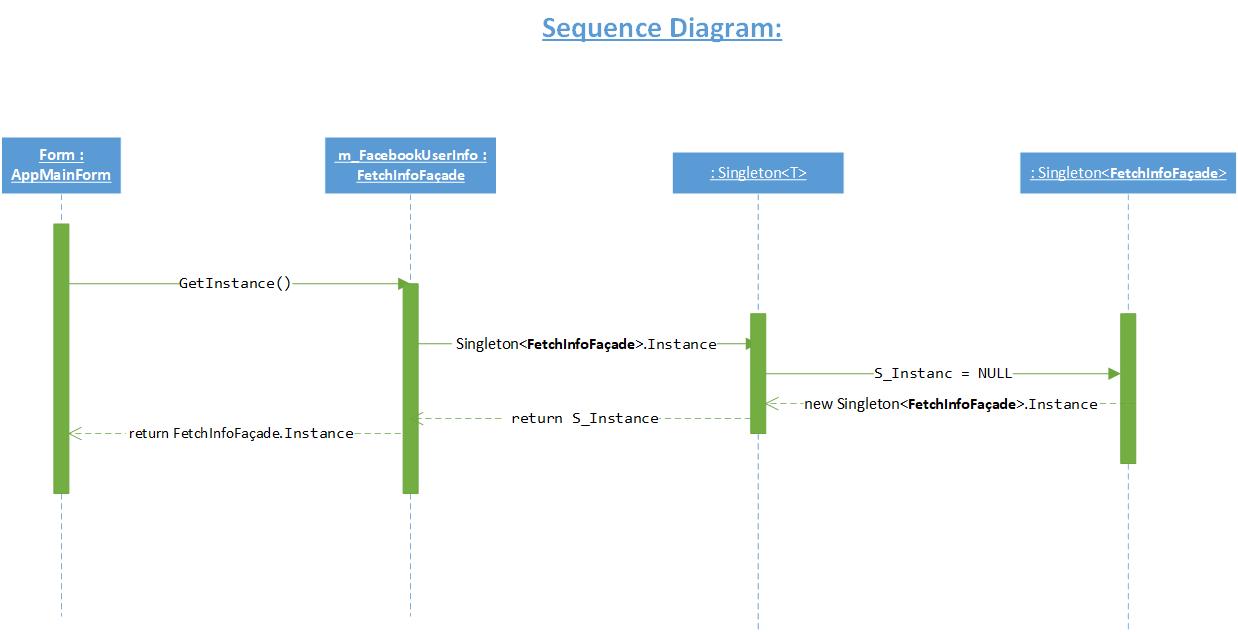
ה- Client הוא: AppMainForm

ה- Singleton הוא: FetchInfoFaçade

המימוש נמצא בקבצים:

A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\ AppMainForm.cs

A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\Logic\FetchInfoFaçade A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\ Singleton.cs



### תבנית מס' 3 – Façade

מטרת התבנית היא להקל את העבודה מול תת מערכות אשר מטפלות באוספים שונים של מידע שהממשק משתמש בהם.

בחרנו בתבנית זו כדי לפשט עבור הטופס את תהליך הבאת המידע מה API ועבודה מול המחלקות השונות בפרויקט, כתוצאה מכך ישנו ממשק פשוט יותר לשימוש אשר ניתן להחליפו לגמרי אם נדרש. בחרנו לממש את התבנית בצורה 'שקופה' יחסית על מנת לאפשר את ביצוע כל תהליך הבאת המידע בפעולה אחת או לחילופין חלקים ממנו.

אופן המימוש:

מימשתי את התבנית ע"י מחלקה אחת, FetchInfoFaçade,שמרכזת את הפונקציונליות של קבלת המידע בחלקו ע"י תצורת מבני הנתונים של *DataTable* בזכות תבנית זו, הצלחנו להכמיס את הלוגיקה של הבאת מידע מה-API ותרגומו.

תפקדים:

ה- Client הוא: AppMainForm

ה- Façade הוא: FetchInfoFaçade

ה -Sub-system Classes:

,VideoStreamer , FacebookAlbum BirthdayDataKeeper

המימוש נמצא בקבצים:

A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\ AppMainForm.cs

A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\Logic\FetchInfoFaçade.cs

A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\Logic\FacebookAlbum.cs A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\Logic\VideoStreamer.cs A18\_Ex02\_Elior\_301797734\_Tzach\_305688269\BirthdayDataKeeper.cs

