### תיאור קצר של הפיצ'רים שבחרנו לממש בתרגיל הקודם:

* **CheckIn Reccomandations**

פיצ'ר זה מאפשר למשתמש לקבל המלצות מהאינטרנט על סימוני הCheckIn שלו.

ניתן לחפש המלצות לפי ארבע קטגוריות: תמונות, וידאו, מזג אוויר, הצעות למסעדות.

לשם מימוש הפיצ'ר, יש בדף הבית של האפליקציה כפתור מעבר לפיצ'ר.

לאחר לחיצה על הכפתור – יוצג דף ייעודי ובו תוצג למשתמש רשימת הCheckIn שלו, כמו כן ישנם 4 כפתורי חיפוש לפי הקטגוריות.

המשתמש יבחר CheckIn עליו ירצה לקבל המלצות, כמו כן ייבחר קטגוריה. כעת יוצגו למשתמש תוצאות החיפוש בתוך הדפדפן של האפליקציה, לפי שם הCheckIn והקטגוריה שבחר.

המשתמש יכול להחליף קטגוריה ו/או את הCheckIn המסומן. ישנה אפשרות לחזור לתפריט הראשי.

* **FaceLiker**

פיצ'ר זה מאפשר למשתמש לקבל מידע וסטטיסטיקה על סימוני הלייקים שביצעו חבריו, לתמונות שלו.

לשם מימוש הפיצ'ר, יש בדף הבית של האפליקציה כפתור מעבר לפיצ'ר.

לאחר לחיצה על הכפתור – יוצג דף ייעודי ובו תוצג למשתמש רשימה של חבריו אשר ממויינת לפי כמות הלייקים שביצעו לתמונותיו של המשתמש.

המשתמש יבחר חבר מהרשימה אשר ירצה לקבל מידע וסטטיסטיקה אודות סימוני הלייק שביצע החבר.

המידע וסטטיסטיקה המוצגים למשתמש הם:

* שם החבר + תמונת הפרופיל שלו.
* סה"כ לייקים שבוצעו ע"י החבר.
* התמונה הראשונה עליה ביצע החבר לייק. (לפי תאריך).
* התמונה האחרונה עליה ביצע החבר לייק. לפי תאריך).
* התמונה עם הכי הרבה לייקים עליה ביצע החבר לייק. (סה"כ לייקים כולל משתמשים אחרים).
* התמונה עם הכי מעט לייקים עליה ביצע החבר לייק. (סה"כ לייקים כולל משתמשים אחרים).

המשתמש יכול לבצע פעולת ביטול חברות (Unfriend) עם החבר המסומן.

המשתמש יכול לשנות את החבר המסומן, עליו מקבל מידע. ישנה אפשרות לחזור לתפריט הראשי.

### תבנית מס' 1 – Singelton

* סיבת הבחירה / שימוש בתבנית:

בחרנו לממש את תבנית הSingelton כ- AppManager. אנו מעוניינים שיהיה מנהל אפליקציה יחיד (לכל היותר) בכל זמן.

מנהל האפליקציה מחזיק את מזהה האפליקציה.

מנהל האפליקציה אחראי על ההתחברות וההתנקות מהFACEBOOK. אנו רוצים שבכל עת יהיה לכל היותר משתמש יחיד מחובר. בנוסף אנו רוצים שלכל רכיבי המערכת תהיה גישה למשתמש המחובר (ולמנהל האפליקציה..) ביוזמתם. כמו כן אם בעתיד נוסיף למערכת אובייקט חדש, האובייקט יוכל לקבל מופע של ה- AppManager, ביוזמתו בלי צורך בשינוי קוד המחלקה.

כמו כן מנהל האפליקציה, אחראי על FORMS הפתוחים ומנווט ביניהם.

אם התבנית לא הייתה ממומשת, הדבר היה לא הגיוני לוגית (מס' מנהלים) כמו כן הדבר היה בזבזני (נתונים זהים חוזרים) ומנוהל בצורה לא עקבית.

* אופן המימוש:

דאגנו לאפשרות לשימוש חוזר בתבנית, מימשנו את המחלקה באופן גנרי. Singleton<T>.

דאגנו שהמנגנון יהיה Thread safe בעזרת שימוש ב- Double check lock.

דאגנו שהמנגנון יהיה Fully Lazy.

דאגנו שמי שרוצה להשתמש באובייקט, יוכל להשתמש בו ביוזמתו. (וגם מי שבעתיד יצטרף תהיה לו אפשרות.)

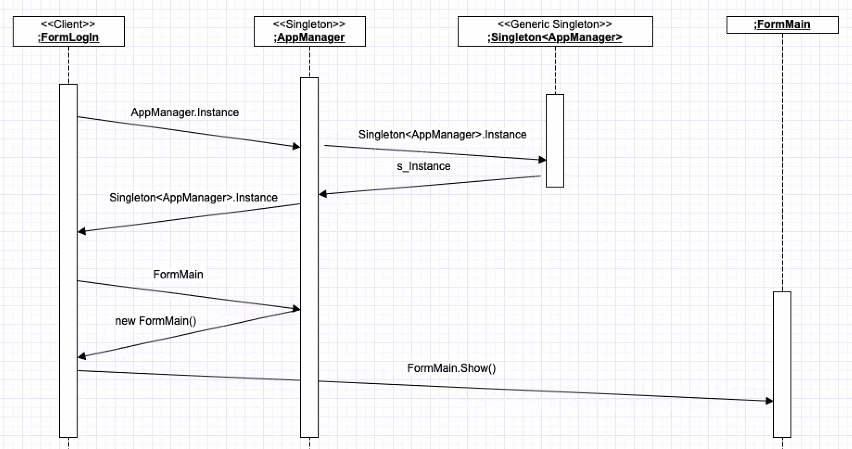
יצרנו סינגלטון מסוג AppManager אשר משתמש במחלקה הגנרית.

* שחקנים בתבנית:

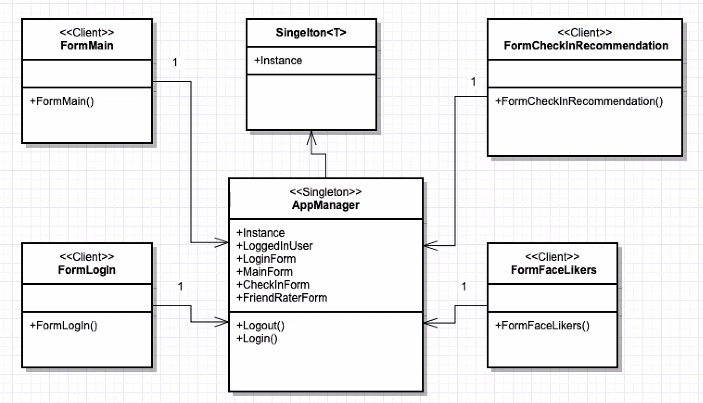
**Singelton: AppManager.**

**Clients: FormCheckInRecommendation,** **FormFaceLikers, FormLogIn, FormMain**

* Sequence Diagram



* Class Diagram



### תבנית מס' 2 – Proxy

* סיבת הבחירה / שימוש בתבנית:

בטופס FormFaceLikersמוצגות מספר רב של תמונות. על מנת שכל התמונות יכנסו בטופס אחד, היה עלינו להקטין את גודלן. דבר אשר פגע בחווית המשתמש, רצינו לאפשר למשתמש להגדיל תמונה שיבחר כדי לראותה באופן ברור ומספק.

זיהינו שיש לנו קשר של צרכן וספק בין שני רכיבים שהם Legacy.

צרכן – FormFaceLikers.

ספק - PictureBox.

רצינו לשפר ולהרחיב את היכולות של הספק, כלומר PictureBox אך מכיוון שהוא הוא רכיב Legacy. בחרנו לממש את תבנית הPROXY.

* אופן המימוש:

יצרנו מחלקה בשם PicturesBoxChangeSizeProxy אשר יורשת מPictureBox. מחלקה זאת ניתנת לשימוש חוזר.

המחלקה החדשה מהווה רכיב חדש אשר מתחזה לרכיב האמיתי, מרחיב ומשפר את יכולתיו של הרכיב אמיתי - אפשרות להגדלת תמונה.

במחלקה PicturesBoxChangeSizeProxy דרסנו את המתודות באופן כזה שישפר את היכולות כפי שרצינו.

- OnMouseEnter – בעת כניסת העכבר לשטח התמונה, התמונה תגדל.

- OnMouseLeave - בעת עזיבת העכבר את שטח התמונה, התמונה תקטן לגודלה המקורי.

כעת מבחינת הצרכן FormFaceLikers אין הבדל, הוא חושב שהוא עובד מול הספק PictureBox אך בפועל הוא עובד מול רכיב הProxy – PicturesBoxChangeSizeProxy.

כעת בחרנו לנצל את הרכיב המשופר ועשות בו שימוש חוזר. השתמשנו בו גם בתמונת הפרופיל אשר נמצאת בטופס הראשי – FormMain.

* שחקנים בתבנית:

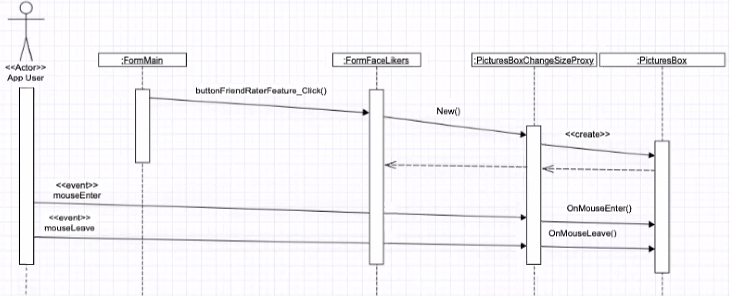
**Proxy:** PicturesBoxChangeSizeProxy

**Isubject –** Icontrol

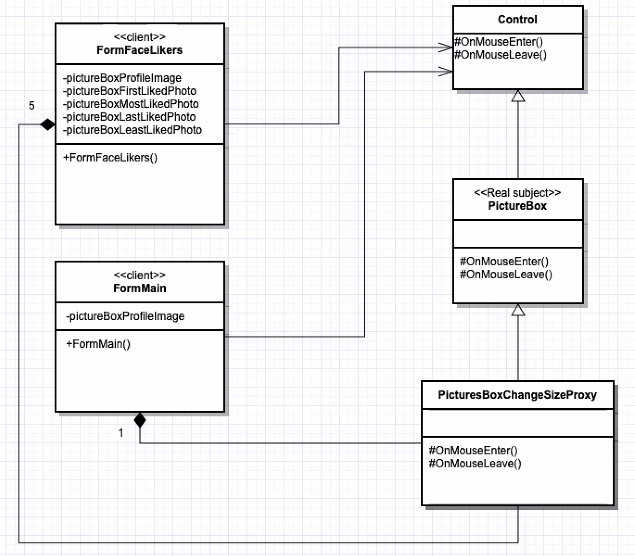
**Real subject –** PictureBox

**Clients:** FormFaceLikers**,** FormMain

* Sequence Diagram



* Class Diagram



### תבנית מס' 3 – Factory Method

* סיבת הבחירה / שימוש בתבנית:

הוספת דף המלצות נוסף בעתיד תדרוש ביצוע פעולות חוזרות רבות (פעולות שכבר ביצענו בעבר) - התאמת גודל הכפתורים ב UI, קישור URL לכפתור, שכפול קוד.

כלומר יש פולימורפיזם בין דפי ההמלצות.

לכן בחרנו להשתמש בPattern Factory Method. עם התבנית, בעת הוספת דף המלצות חדש מספר הכפתורים וגודלם יותאם דינאמית, הURL יקושר דינאמית לכפתור ובנוסף ייחסך שכפול קוד.

* אופן המימוש:

1. ממשק IURL - כל המחלקות שרוצות לספק URL יממשו ממשק זה.

המחלקה URLFactory - אחראית על יצירת אובייקטים מהמשפחה הפולימורפית IURL.

1. הURLFactory יספק ל UI את רשימת השמות עבורם יווצרו כפתורים לחיפוש.

כאשר משתמש ילחץ על כפתור בUI, יימסר ל URLFactory שם הכפתור, URLFactory ייצר ויחזיר לUI אובייקט מסוג IURL. ממנו הUI ייצר URL ויציגו בדפדפן.

1. הEnum eURLType - משמש כשפה משותפת בין ה Factory ל UI.

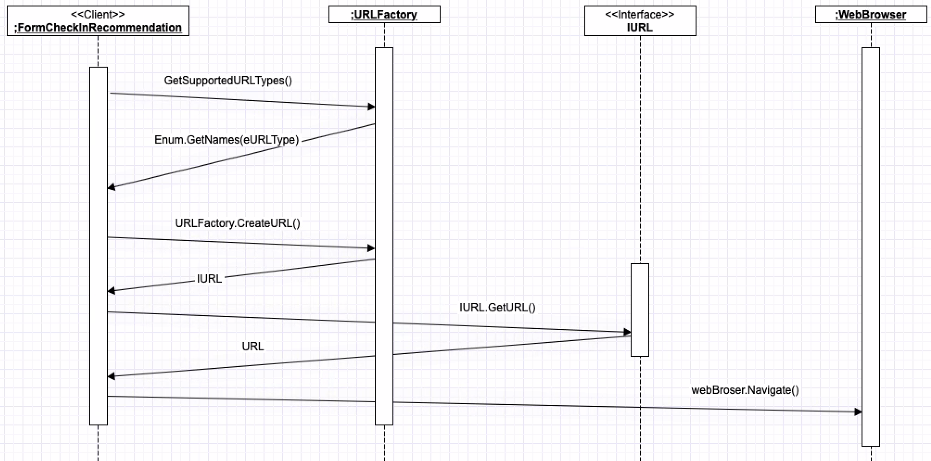
* שחקנים בתבנית:

**Factory:** URLFactory

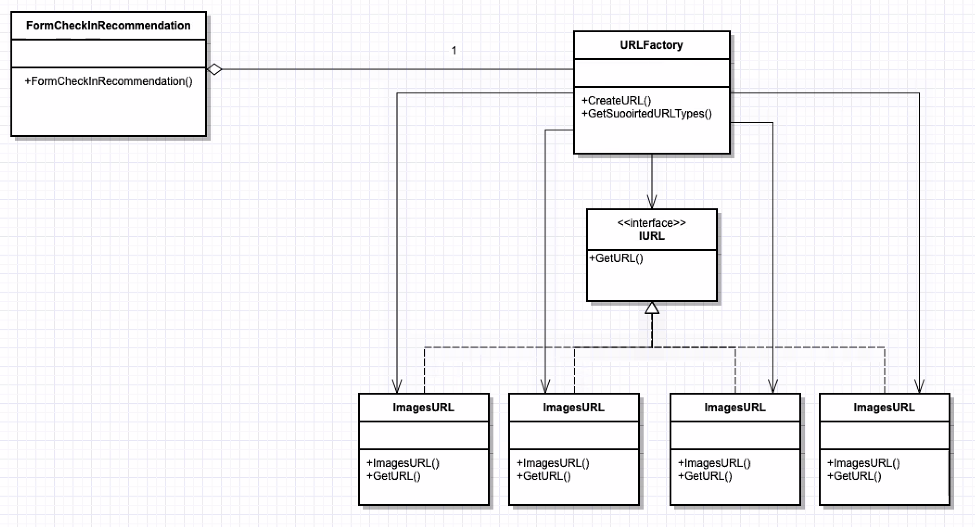
**IURL Interface**: ImagesURL, RestaurantsURL, VideosURL, WeatherURL

**Client** (UI): FormCheckInRecommendation

* Sequence Diagram



* Class Diagram



### עבודה אסינכרונית:

זיהינו כי בFormMain ישנן פעולות אשר עובדות מול השרתים של FB, פעולות אלו לוקחות זמן רב ובנתיים "מקפיאות" את האפליקציה. פעולות אלו אינן משפיעות ו\או קשורות לפעולות אחרות.

הבנו כי ניתן לבצע פעולות אלה באותו הזמן בלי חשש, כמו כן רצינו לשפר את חווית המשתמש ולאפשר לו לנצל את הזמן המת בו הוא ממתין לביצוע הפעולות.

לכן בחרנו לבצע אותן באופן אסינכרוני.

הפעולות המתבצעות באופן אסינכרוני הן:

* fetchFriends()
* fetchPosts ()
* fetchCheckIns ()

שלושתן נמצאות ב בFormMain.

### Data Binding

בחלון הראשי של האפליקציה – FormMain מוצגים נתוניו של המשתמש המחובר. בחרנו לעבוד איתם בשיטת Two way data binding.

המשתמש יכול לעדכן את נתונים אלו (בתנאי שיעמדו בבדיקות תקינות בסיסיות שהגדרנו), הנתונים החדשים יישמרו גם לאחר מעבר בין דפי האפליקציה.

האובייקט ששימש אותנו כ- Data Binding Sourceהוא ה-LoggedInUser והנתונים הם:

* שם מלא.
* מייל.
* תאריך לידה.