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

* TODO – חלון שמטרתו היא הצגה למשתמש את כל האירועים שהוא צריך להגיב אליהם בזמן הקרוב או שמחכים לו. לאחר החיבור יופיעו למשתמש כלל האירועים שהוא צריך להגיד אם הוא מגיע/לא אליהם.

בנוסף, יופיעו לו כלל הפוסטים בהם תוייג ועדיין לא הגיב, כדי להימנע מתקריות חברתיות.

* Feed מועדפים – מאפשר בחירת החברים המועדפים עליך ולראות את ה-posts שלהם בלבד. כך המשתמש מייצר לו feed המתאים לו בדיוק. המשתמש בוחר את חבריו אותם רוצה לראות ב-feed האלטרנטיבי והוא מקבל את כלל הפוסטים שלהם.

### תבנית מס' 1 – Observer

* סיבת הבחירה / שימוש בתבנית:

לאחר המשתמש מבצע Login למערכת מתבצעת טעינה של נתונים אל המסך וכך גם לאחר שהמשתמש מתנתק אז מתבצע ניקוי של הנתונים. אם בעתיד נראה להוסיף/להוריד אנחנו צריכים בכל פעם להוסיף ולהוריד את המהלכים ממחלקת ה-Form המרכזית באופן פרטני וספציפי לכל נתון שנטען. ועל כן אנחנו משתמשים ב-observers שמחכים לאירוע ההתחברות/ההתנתקות והם יעדכנו את המסך בהתאם לאירוע שעולה.

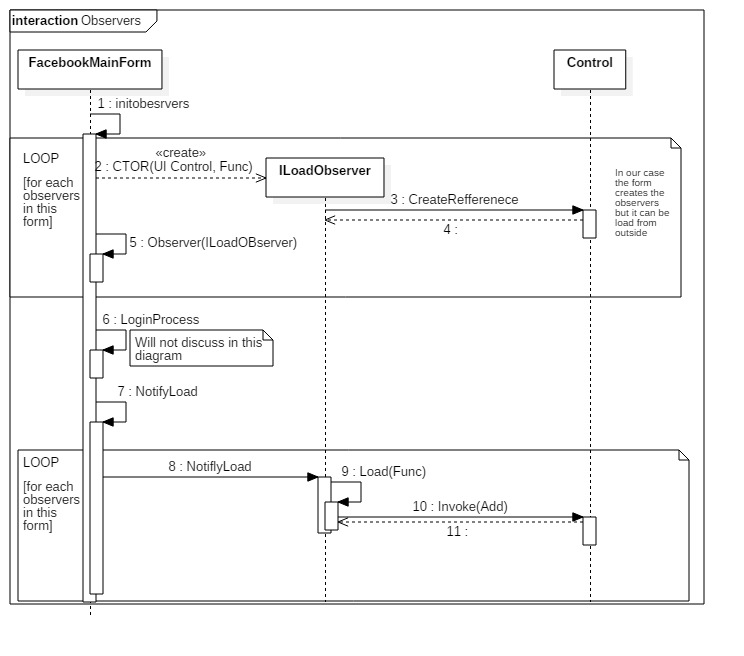
* אופן המימוש

ILoadObserver – interface שמי שממש אותו הוא Observer והוא מבצע את פעולות הnotify אשר במקרה שלנו הם התחברות והתנתקות.

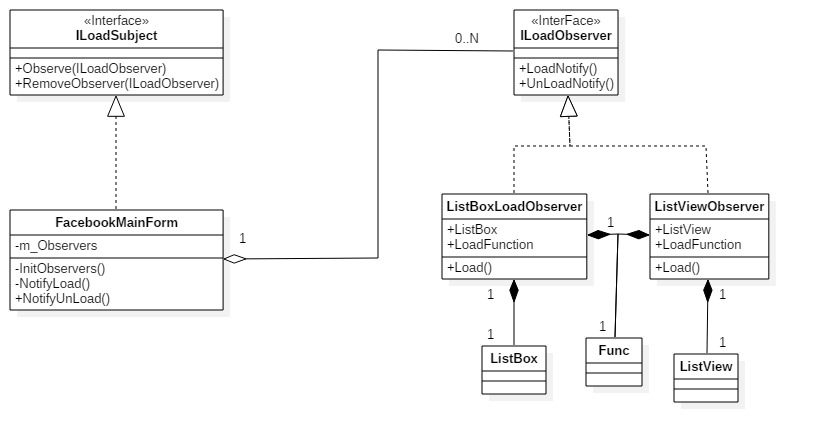
המחלקות ListBoxLoadObserver<T> ו-ListViewLoadObserver<T> הינם המחלקות הממשות את הממשק, הם מחזיקות את אובייקט ה-UI המתאים אליו צריך לטעון את הנתונים ואת פעולה (FUNC<>) הטעינה. T מייצג את המחלקה הנטענת אל הפקד, במקהר שלנו כולם Adapters.

ב-Form יש רשימה של ILoadObservers וכאשר יש אירוע מתאים אז מודיעים לכלל ה-observer לבצעו.

* Sequence Diagram



* Class Diagram



### תבנית מס' 2 – [שם התבנית]

* סיבת הבחירה / שימוש בתבנית:

[תיאור הסיבה / הצורך בשימוש בתבנית במערכת שלכם]

* אופן המימוש:

[תיאור המימוש והיכן ניתן למצוא אותו בקוד]

* Sequence Diagram

[sequence diagram שמציגה את התהליך שקשור לתבנית והאינטראקציה בין המחלקות]

* Class Diagram

[class diagram שמתארת את המחלקות שלכם שמעורבות בתבנית (תיאור מלא שכולל Properties ו- Methods) והיחסים ביניהם (תאור מלא עבור כל יחס כפי שלמדנו בכיתה). עבור כל מחלקה שלכם, כיתבו מי המקבילה שלה (אם יש כזו) בתבנית (Pattern) שבחרתם

### תבנית מס' 3 – [שם התבנית]

* סיבת הבחירה / שימוש בתבנית:

[תיאור הסיבה / הצורך בשימוש בתבנית במערכת שלכם]

* אופן המימוש:

[תיאור המימוש והיכן ניתן למצוא אותו בקוד]

* Sequence Diagram

[sequence diagram שמציגה את התהליך שקשור לתבנית והאינטראקציה בין המחלקות]

* Class Diagram

[class diagram שמתארת את המחלקות שלכם שמעורבות בתבנית (תיאור מלא שכולל Properties ו- Methods) והיחסים ביניהם (תאור מלא עבור כל יחס כפי שלמדנו בכיתה). עבור כל מחלקה שלכם, כיתבו מי המקבילה שלה (אם יש כזו) בתבנית (Pattern) שבחרתם