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

Feature 1: HOW WELL DO KNOW YOUR FRIEND(GAME)

הפיצ'ר הזה מאפשר למשתמש לשחק בשאלון על אחד מחבריו בפייסבוק ולקבל תוצאה על השאלון וככה לדעת עד כמה הוא מכיר את חבריו בפייסבוק .

Feature 2 : POTENTIAL PARTNER

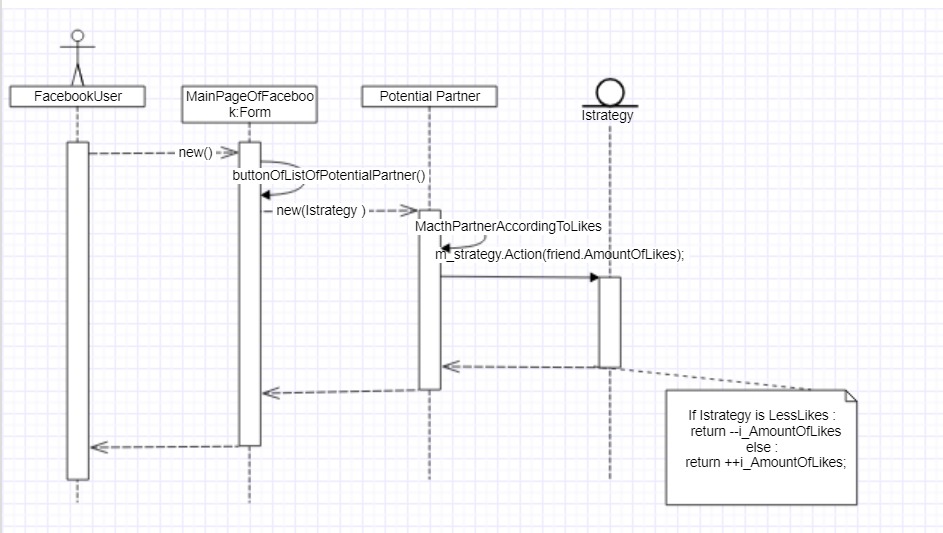
הפיצ'ר הזה מאפשר למשתמש/ת למצוא לו בן זוג או בת זוג פוטנציאלים בכך שמציגים לו לא את החברים בסדר ממוין מה החבר שעשה לו הכי הרבה לייקים לתמונות ולפוסטים, כך שהחבר שבראש הרשימה הוא החבר עם הסיכויים הכי גבוהים להיות בן זוג או בת זוג.

### תבנית מס' 1 – [Strategy]

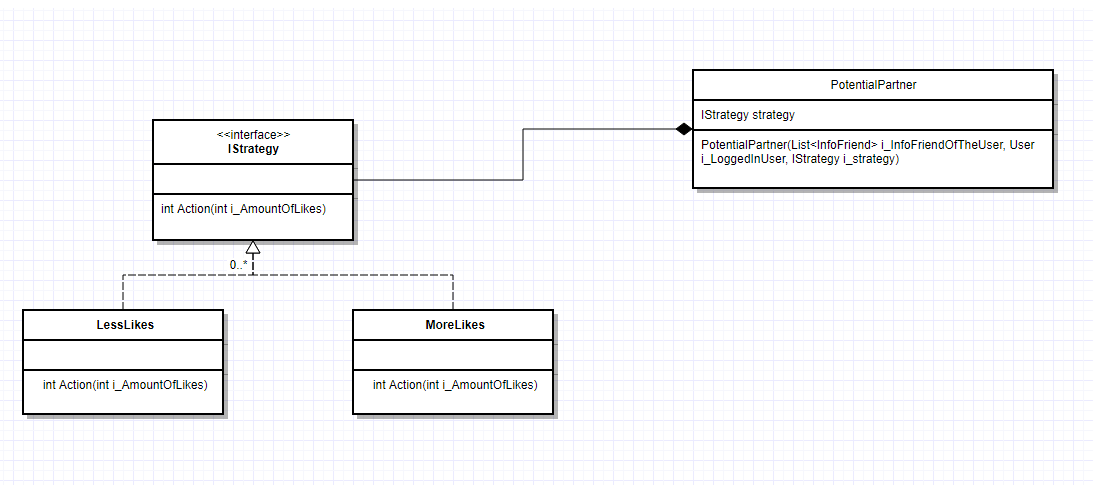
* סיבת הבחירה / שימוש בתבנית:
* **strategy –** באפלקצית הפייסבוקית שלנו ישפיצ'רpotential partner ששם אנו מחפשים את החבר שעשה לנו הכי הרבה לייקים וכך לו יש פוטנציאל להיות חבר\חברה שלנו והוספנו עוד פיצר חדש שכאשר נרצה לראות מי החבר שעשה לנו הכי פחות לייקים נמצא אותו וזה שאלגוריתם של הפיצ'ר החדש מאוד דומה לקודם ונרצה לבחור אלגוריתמים בזמן ריצה ולאפשר לאלגוריתם להשתנות באופן עצמאי בין לקוחות שמתמשים בה ותרחיש זה בדיוק מתאים לתבנית strategy
* אופן המימוש:

מצאנו את Injection Point (המקום שאפשר לשנות את הקוד באלגוריתם שלנו ) שבו היינו סוכמים את כמות הלייקים לאחר מכן הגדרנו interface בשם IStrategy שבה יש פונקציה אחת בשם action שיש לנו לנו שתי מחלקות חדשות שממשות את האינטרפיס הזה כל אחד בדרך שונה.המחלקהmorelikes ממשת את אינטרפיס בצורה שהיא עושה ++ לכמות הלייקים והמחלקה lesslikes פשוט עושה – לכמות הלייקים .והגדרנו במחלקה potinalpartner member מסוג IStrategy שמאותחל בקונסטרקסטור ובצורה הזאת אנו יכולים לשנות את אלגוריתם באופן עצמאי בזמן ריצה .

* Sequence Diagram



* Class Diagram



### תבנית מס' 2 – [Command]

* סיבת הבחירה / שימוש בתבנית:

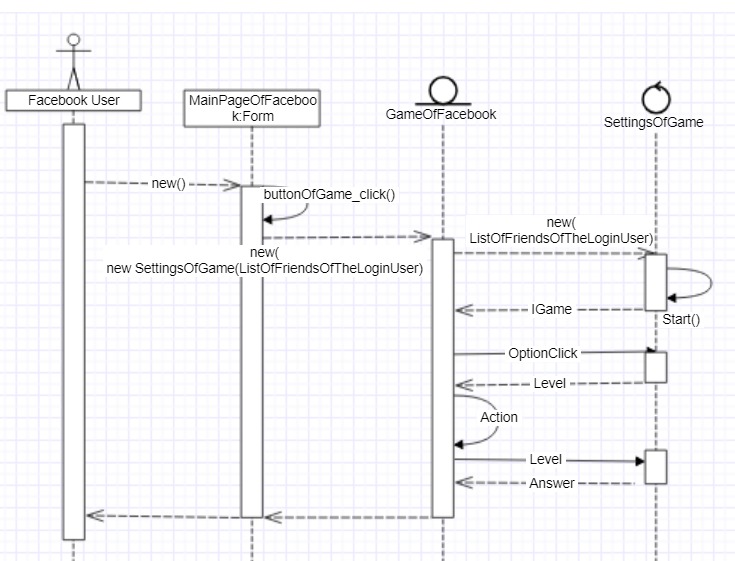
**command** - באפלקצית הפייסבוקית יש את הפיצ'ר של המשחק how well do know your friend ובמחלקת של settingofgame ברגע שהשחקן לחץ על אחד הכפתורים אנו בודקים באיזה מהכפתורים הוא לחץ בswitch case אם זה 1 (תעשה פעולה..) או 2 (תעשה פעולה..) או 3 (תעשה פעולה..) וכו' ובדיקה הזאת לא טובה מכמה סיבות כל פעפ שחקן לוחת על הכפתור אנו עושים בדיקה ב o(n) לכל הפחות ,וזה לא תחזוקתי אם נרצה בעתיד להוסיף עוד כפתור נצטרך לעדכן בswich case ויהיה קל מאוד לשכוח לעדכן הכול ולבעיה כזאת התבנית command פותרת לנו

* אופן המימוש:

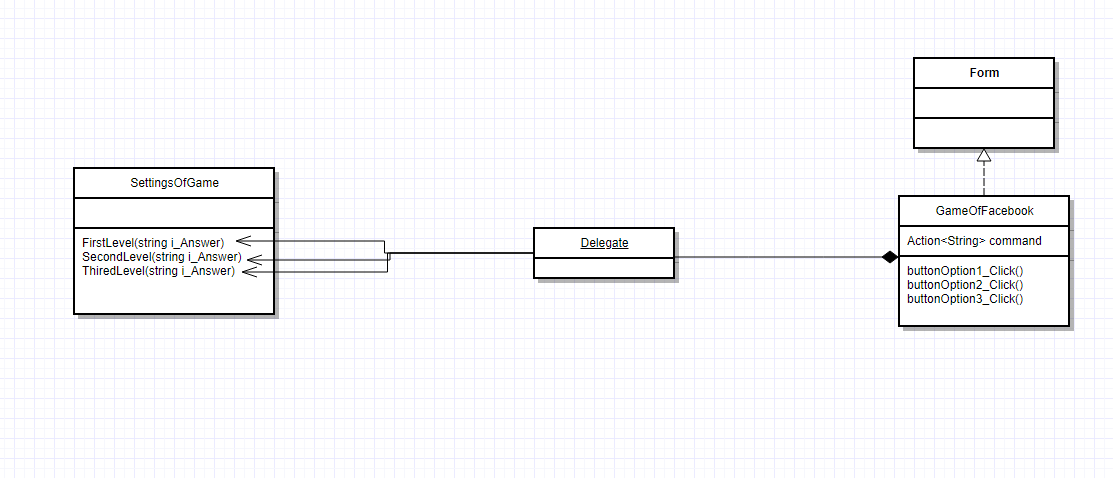
בחרנו לממש את התבנית הזאת עם delegate .

הגדרנ memberמסוג delegate בUI של המשחק במחלקה gameoffacebook כך שברגע של לחיצה על אחד הכפתורים לא נעשה swich case אלה ניגש לפונקציה ישירות בעזרת הdelegate שהגדרנו.

* Sequence Diagram



* Class Diagram

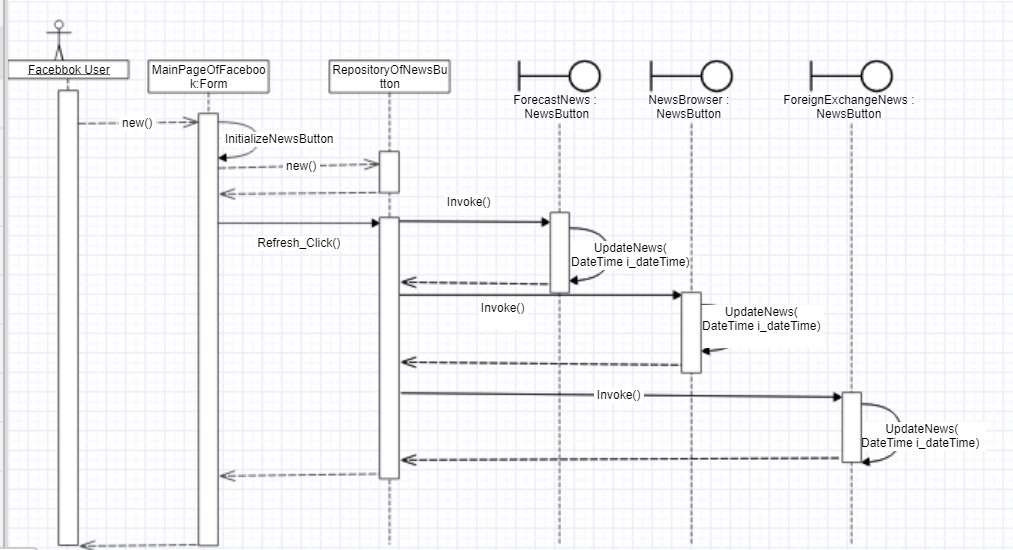


### תבנית מס' 3 – [Observer]

* סיבת הבחירה / שימוש בתבנית:
* **Observer–**  רצינו לממש יכולת לעדכן אוסף של דפי אינטרנט באותו זמן כשמתחרש אירוע חיצוני . או במילים אחרות אוסף של listeners שמקשיבים לNotifier אחד. לכן התרחיש התאים לפטארן observer שבא לפתור את הבעיה הנדסית הזאת ובמיוחד עם מימוש יותר קל על ידי שימוש ב delegate . יצרנו מחלקה RepositoryOfNewsButton שאחראי לעדכן בלחיצת על כפתור לדוגמה את כל ה listerners שלו שהם הדפים של: מזג אוויר , חדשות , מטבעות חוץ. ברגע שהאירוע לחיצת כפתור קורה כל אחד מה listeners של ה Repository עושה Refresh ומקבל את הזמן שקרה ה עדכון.
* אופן המימוש:

יצרנו קודם כל את המחלקה RepositoryOfNewsButton שבעצם תחזיק את אוסף הברווזרים יצרנו לאחר מכן 3 מחלקות ForecastNews, NewsBrowser, ForeignExchangeNewsשיורשים מהמחלקה אבסטרקטית NewsButton שהיא בעצמה יורשת מהמחלקה של winform WebBrowser. ההחלטה של ירושה נבעה מהצורך לחסור מהשכפול הקוד שכל אחד מהמחלקות היורשות יחזיק Datetime וגם בסיס פולימורפי ל Refresh כך כשכל אחד מה ברווזרים יכול לבצוע פעולה רצויה ושונה במידת הצורך .כל אחד מאברווזרים נרשם לאירוע בתןך המחלקה RepositoryOfNewsButton על ידי דיליגייט שמצביע לפונקציה UpdateNews . בלחיצת כפתור ה repository מבצע את הפעולה Invoke שמעדכן כל אחד מהדפים שנרשם להאירוע.

* Sequence Diagram



* Class Diagram

