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

\*\*\*מידע על הTest User שיש לעשות בו שימוש:

מייל: mzacgqusud\_1533077714@tfbnw.net

סיסמא: design.patternspp

* פיצ'ר משחק הזיכרון : מוצגות למשתמש כמה תמונות פרופיל ועליו לבחור רק את החבר שלו בפייסבוק. התמונות זזות תוך כדי על מנת ליצור "בלבול" אצל המשתמש. התמונות של ה"לא חברים" נלקחות מתוך מאגר ששמור בResources.

ראה מחלקה MemoryGame .

* פיצ'ר ניתוח סנטימנטלי של פוסטים : המשתמש יכול לצפות בחלוקה של הפוסטים שלו לפוסטים עם סנטימנט שלילי ופוסטים עם סנטימנט חיובי. זאת על ידי בחירת מסווג מתחום למידת המכונה מבין SVM, Rocchio. בכך ניתן לראות איך המסווגים השונים לומדים את הטקסט האישי של המשתמש.

ראה מחלקה SentimentAnalyzer .

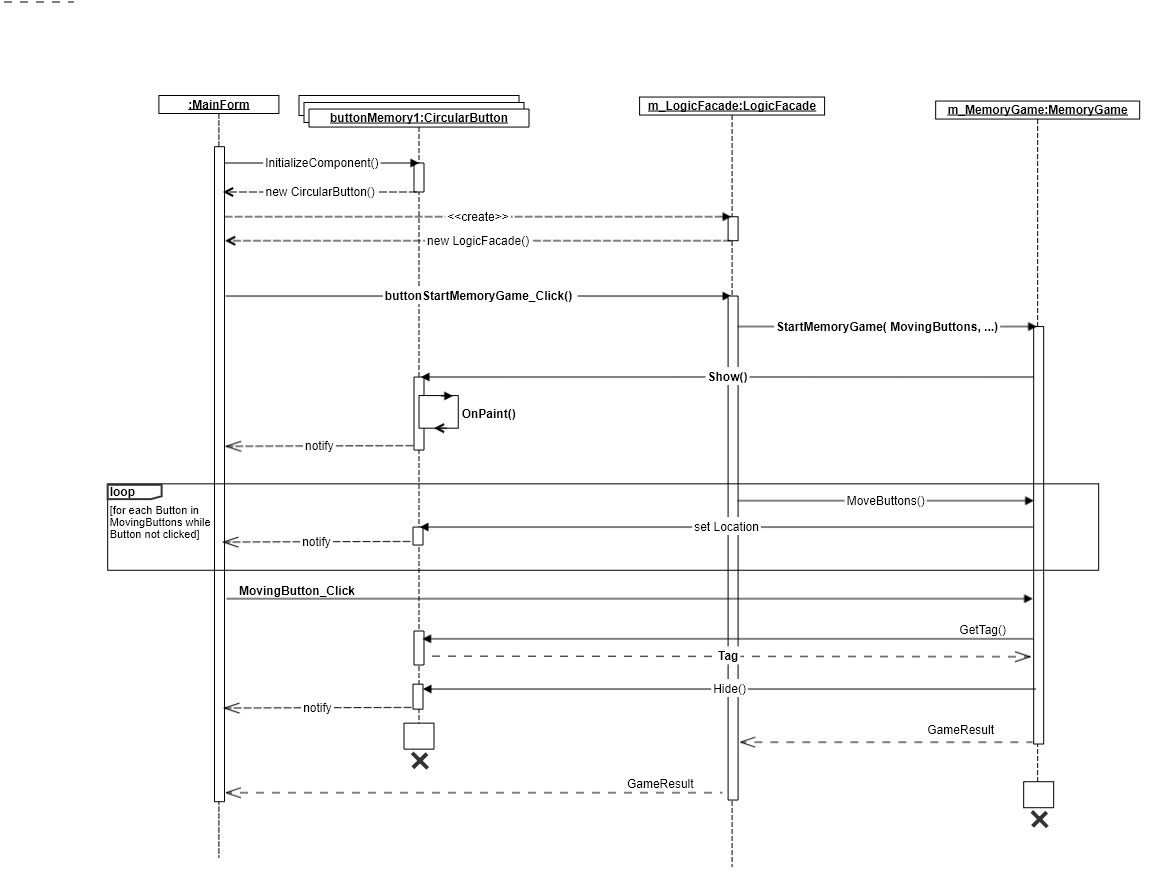
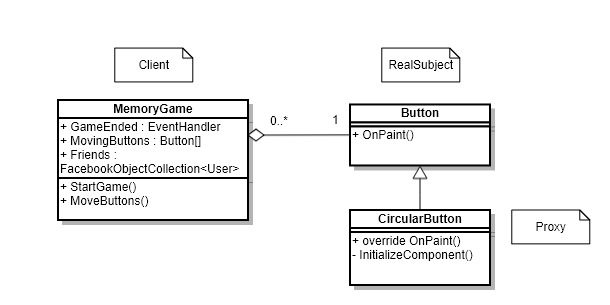
### תבנית מס' 1 – Proxy

* סיבת הבחירה / שימוש בתבנית:

על מנת לשכלל את האופן בו הפקד Button מצייר את עצמו, אך עדיין לתת ל-client אפשרות לעבוד איתו כ-Button ללא שינוי קוד הclient (או Button).

* אופן המימוש:

יצרנו את מחלקת CircularButton שיורשת מ-Button ועשינו Override למתודת OnPaint() כדי שהפקד יצייר את עצמו כעיגול.

* Sequence Diagram
* Class Diagram

.

### תבנית מס' 2 – Transparent Facade

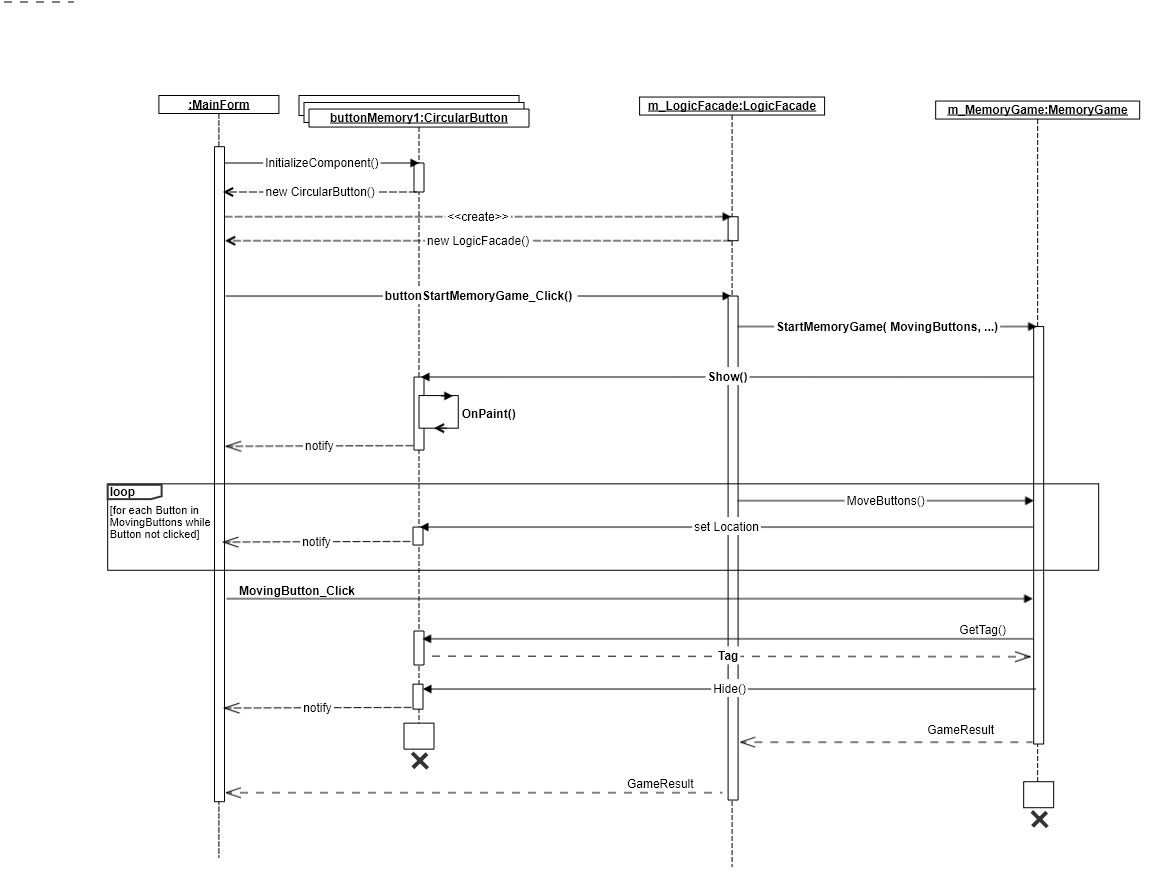
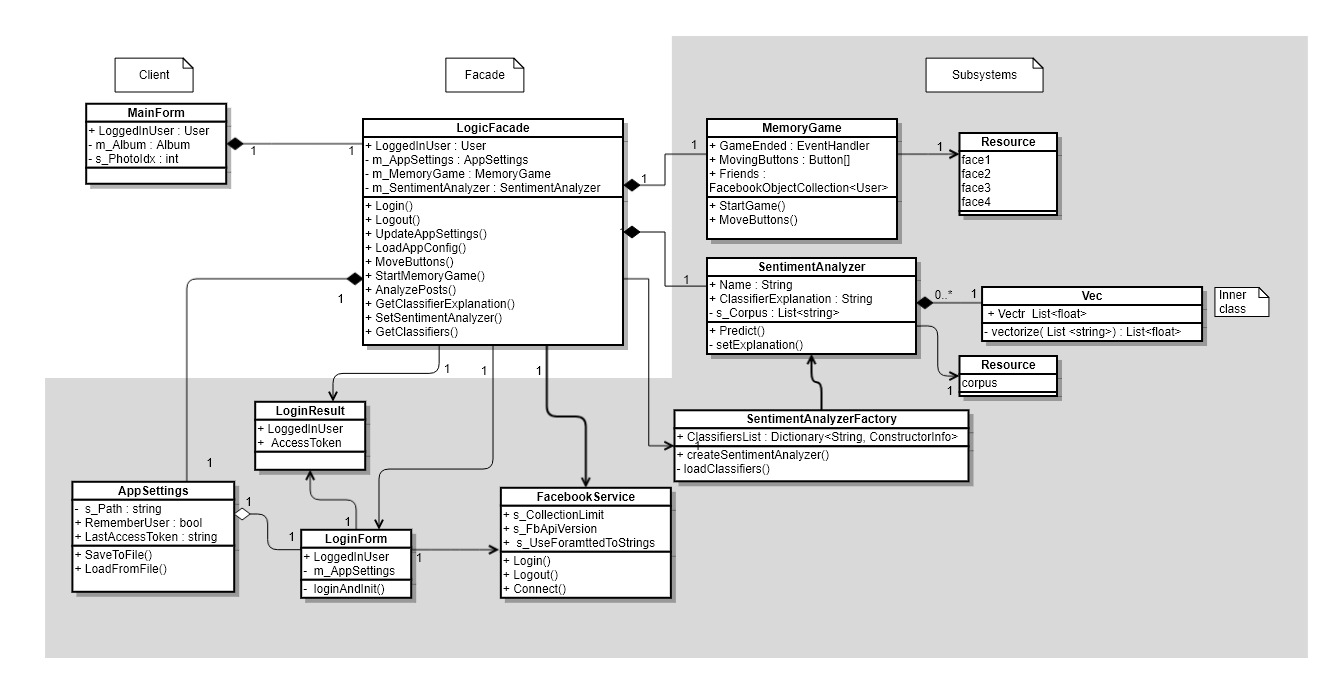
* סיבת הבחירה / שימוש בתבנית:

על מנת להפריד בין ה-MainForm ללוגיקה העומדת מאחוריו (הפיצ'רים החדשים, הבאת נתוני משתמש מפייסבוק) וליצור מצב בו הUI אינו תלוי באפון המימוש של הלוגיקה.

* אופן המימוש:

מחלקת LogicFacade מהווה facade של הלוגיקה. יצרנו את המחלקה כך שכל המתודות הקשורות ללוגיקת התכנית מגיעים מהUI למחלקה וחוזרים ממנה אל הUI.

כולל login ו-logout של המשתמש.

* Sequence Diagram
* Class Diagram

### תבנית מס' 3 – Factory Method

* סיבת הבחירה / שימוש בתבנית:

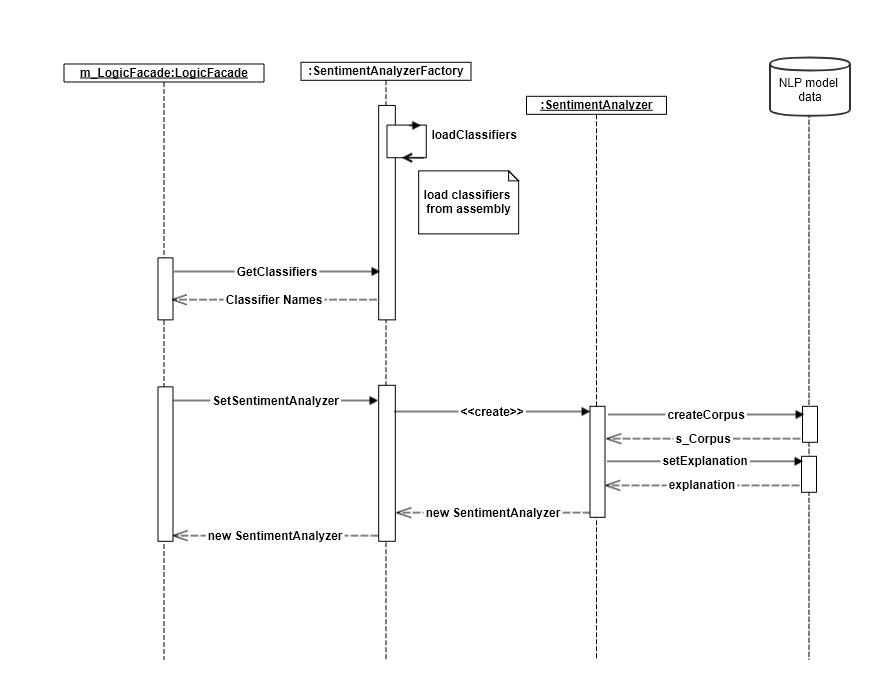
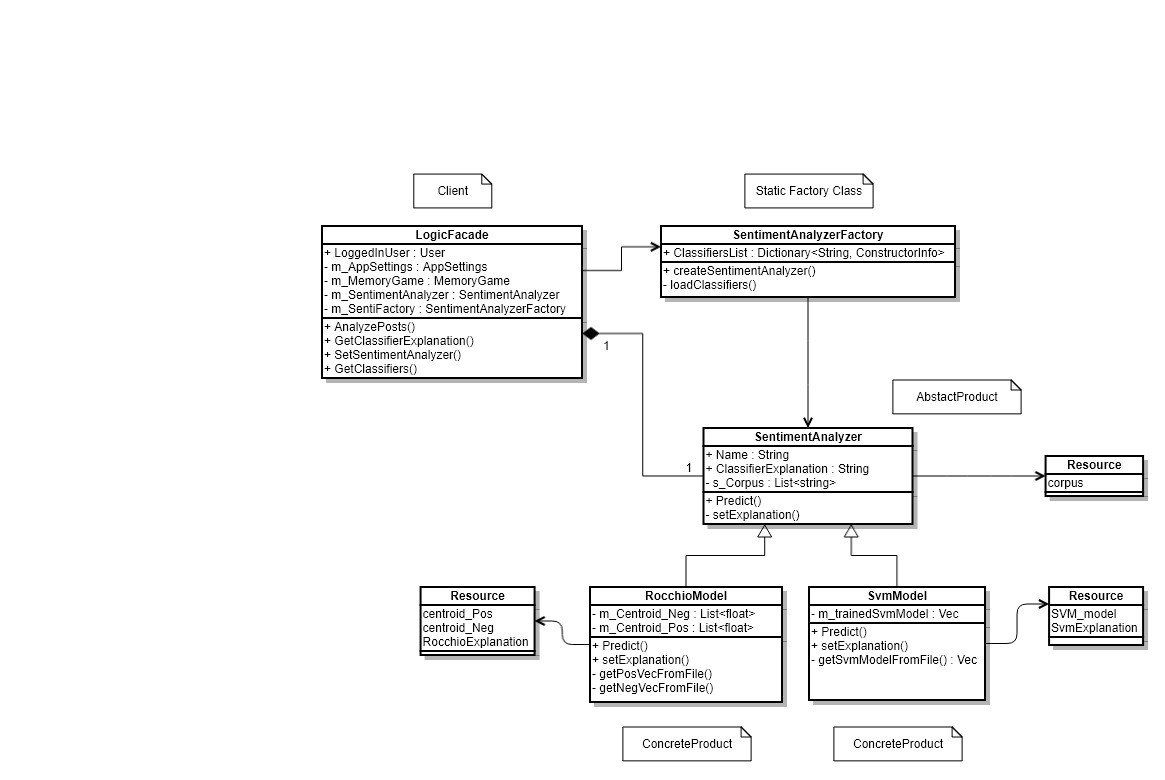
כדי לאפשר מימוש של כל אלגוריתם סיווג טקסט מסוג Sentiment Analysis להתווסף לAssembly - באופן אורגני ולהתשתלב בתכנית ללא ידיעה מראש אף פרט עליו.

* אופן המימוש:

הפכנו את ה- SentimentAnalyzer לאבסטרקטית כאשר מתודות וירטואליות הן Predict(), ו-setExplanation(). ויצרנו שתי מחלקות קונקרטיות – RocchioModel, SvmModel היורשות ממנה.

יצרנו מחלקה סטטית בשם SentimentAnalyzerFactory שמשתמשת ב-Reflection כדי לזהות את כל המחלקות היורשות מהמחלקה האבסטרקטית SentimentAnalyzer הנמצאות במארז הכולל. מכל מחלקה כזו שנמצאת במארז ה- SentimentAnalyzerFactory שומר מיפוי מהשם של המחלקה לבנאי שלה.

מתודת יצירת SentimentAnalyzer מסוג מתאים נעשה על ידי Invoke לבנאי התואם מתוך המיפוי.

* Sequence Diagram
* Class Diagram

### עבודה אסינכרונית ו-Data Binding

* עבודה אסינכרונית:

על מנת לטעון מידע )מתודות fetch למיניהם כגון fetchAlbums() ) משרתי פייסבוק באופן שבו המשתמש יכול לראות את הUI ללא עיכובים, ולקבל שליטה על החלון תצוגה לפי שהכל מספיק להיטען בוצעה עבודה אסינכרונית. בקוד ניתן לראות את אופן העבודה עם הבאת פוסטים בצורה אסינכרונית – backgroundWorkerPost או באופן שונה ב- fetchAlbums במחלקת MainForm.

* עבודה עם Data Binding:

נעשה שימוש בכך במחלקת MainForm – במתודות fetchPosts() ו- fetchEvents()ניתן לראות את החיבור של ה-Binding source לאובייקטים שיש למשתמש. ובUI ניתן לראות גם שהמידע הכללי אודות המשתמש בראש העמוד הראשי נעשה בפאנל אחד שמקבל את המידע שלו מהמשתמש בעזרת Data Binding.