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

* **Memories game:**

**פיצ'ר המאפשר למשתמש לשחק במשחק הזיכרון ברמות קושי שונות, כאשר הקלפים במשחק נמשכים באופן אקראי מאלבומי התמונות של המשתמש. הפיצ'ר נועד לספק למשתמש חווית משחק מהנה המשלבת זיכרונות נוסטלגיים מהעבר.**

* **Grouper:**

**פיצ'ר המאפשר למשתמש ליצור רשימות חברים (friend lists) חדשות על בסיס מכנה משותף שלו איתם, מתוך רשימת קטגוריות אפשריות קבועות מראש:**

1. **עובדים \ עבדו יחד.**
2. **לומדים \ למדו יחד.**
3. **נמצאים באותו סטטוס רומנטי.**
4. **אוהדים את אותה קבוצה \ קבוצות.**
5. **גרים באותה העיר.**
6. **בני אותו שכבת גיל.**

**הפיצ'ר יציג למשתמש את רשימת חבריו המתאימים לקטגוריה הנבחרה, וכל מה שנותר יהיה למשתמש לעשות הוא לבחור מתוך הרשימה מי מהחברים הוא רוצה לצרף ל"friend list" וללחוץ על כפתור יצירת הרשימה, ורשימה בעלת שם מתאים תיווצר עבורו.**

### תבנית מס' 1 – Singleton

* **סיבת הבחירה / שימוש בתבנית:**

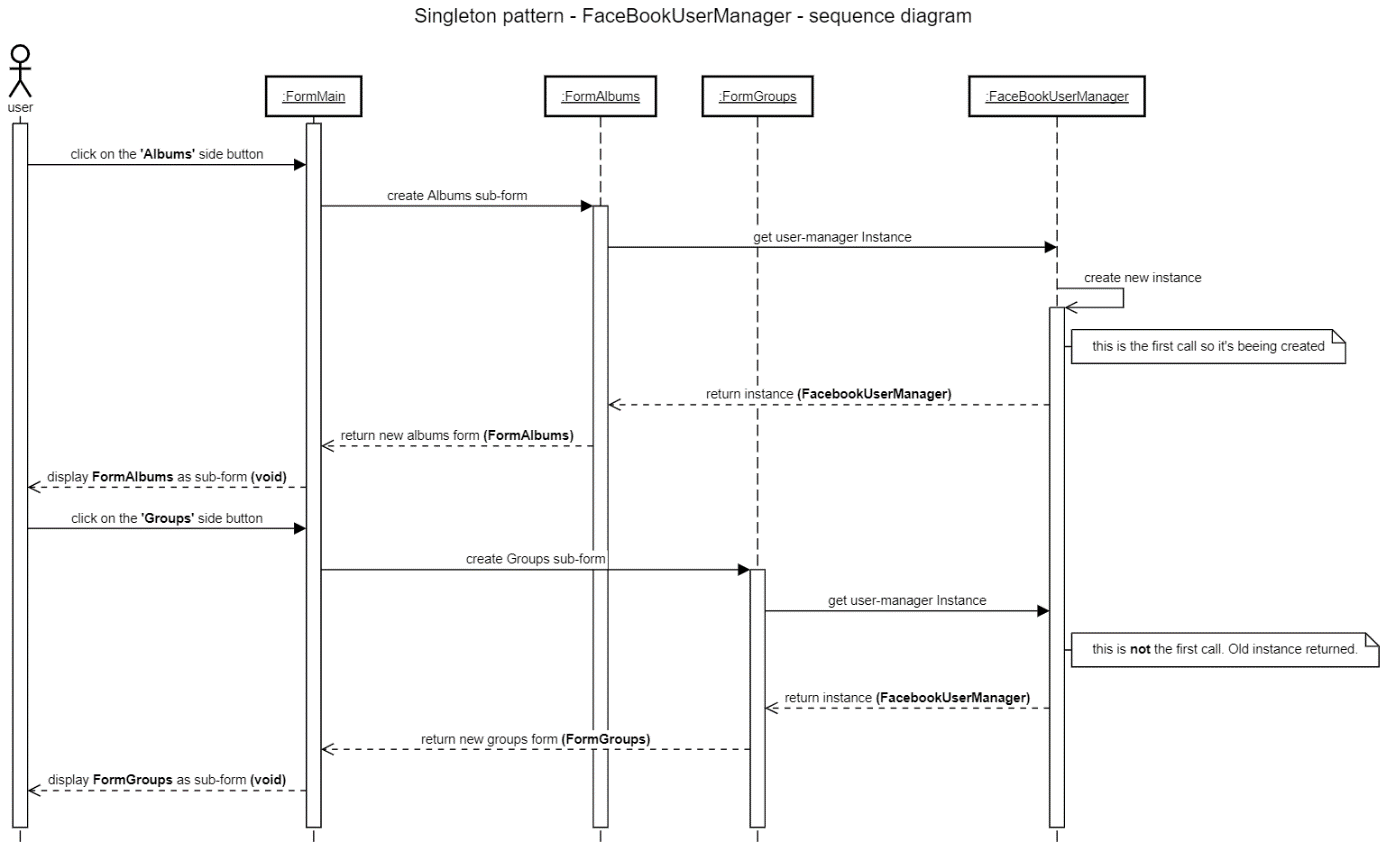
בפרויקט שלנו המחלקה שמממשת **singleton** היא מחלקה הנקראת ‘***FaceBookUserManager’****.* תפקיד המחלקה הוא להחזיק את פרטי המשתמש המחובר, לבצע את לוגיקת ההתחברות וההתנתקות מהמערכת, ולשמור את הגדרות האפליקציה. פרטי המשתמש וההגדרות הם יחידים לאפליקציה, ואין שום היגיון בהחזקת מספר משתמשים פעיליםבמקביל. בנוסף השימוש במחלקה הוא תדיר מפני שכמעט בכל רכיב באפליקציה יש צורך במשיכת מידע אודות המשתמש, ובבדיקה אם מחובר אחד כזה. כל אלו נשמרים ב‘***FaceBookUserManager’****,* ולכן היינו רוצים שמופע המחלקה היחיד שמחזיק את המידע הזה יהיה נגיש למשיכה בכל רכיב במערכת מיוזמתו, מבלי צורך לכפות תלויות בין רכיבים במערכת.

בגלל כל הסיבות הנ"ל בחרנו לממש את ‘***FaceBookUserManager’*** כ-**Singleton** שכן תבנית זו פותרת בדיוק את הבעיות ההנדסיות שציינו.

* **אופן המימוש:**

המחלקה ‘***FaceBookUserManager’*** מממשת את תבנית **singleton** באופן הבא:

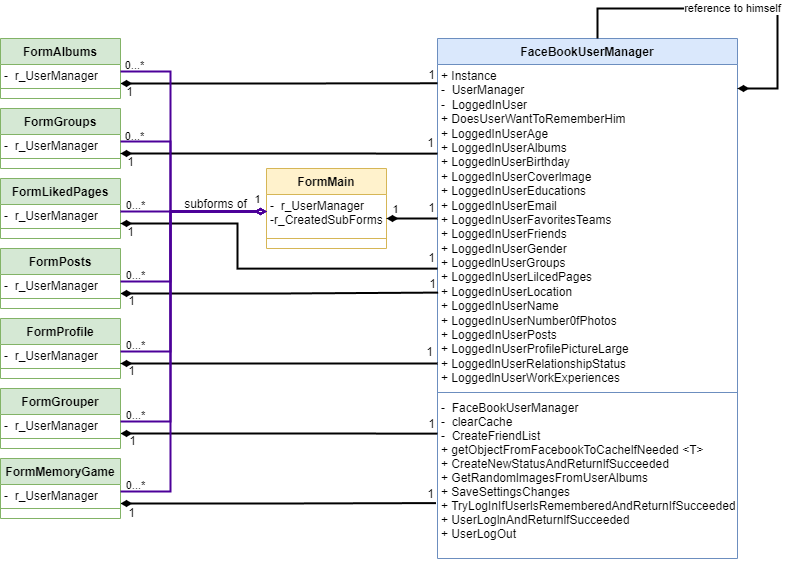
* המחלקה ממומשת כ - **sealed class**.
* בנאי המחלקה הוא .**private**
* המחלקה מחזיקה **public static property** בשם ‘***instance’*** המחזיק את המופע היחיד של המחלקה. ‘***instance’*** מאותחל להיות **null**, ונוצר באופן ‘עצלן’, כלומר נוצר רק בזמן הבקשה הראשונה. תהליך היצירה מוגן משגיאות היכולות להיווצר כתוצאה מפניות מרובות תהליכונים שונים באמצעות מנגנון **double** **check lock**.
* **Sequence Diagram**



* **Class Diagram**

כאמור ‘***FaceBookUserManager***’ היא המחלקה המממשת את התבנית **singleton**.

הטפסים השונים משמשים כ**’client’**, והם אלו שמשתמשים במופע היחיד של ‘***FaceBookUserManager***’.



### תבנית מס' 2 – Factory method

* **סיבת הבחירה / שימוש בתבנית:**

השתמשנו בתבנית ‘**Factory method’** על מנת ליצור את ה-sub forms השונים.

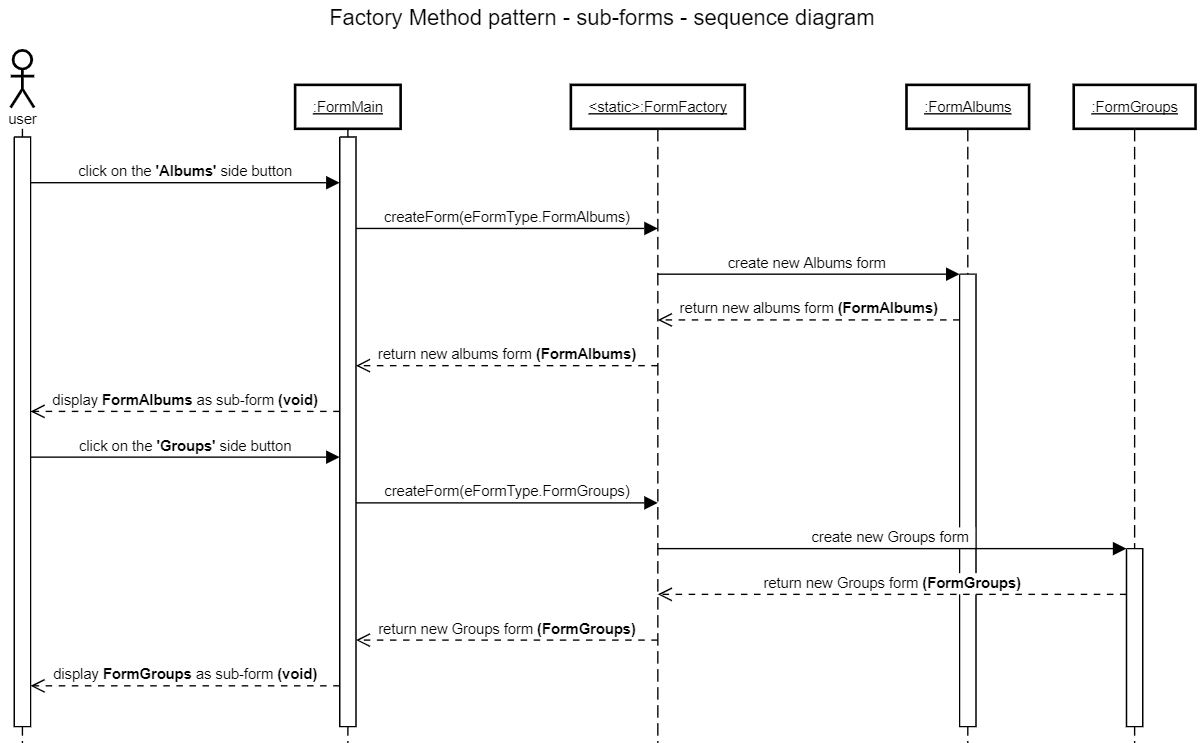
כל תתי הטפסים הם פולימורמים (מעצם ירושתם מform), ונתקלנו בצורך ליצור בכל פעם form אחר, בהתאם לבחירת המשתמש. כמו כן רצינו שתהליך יצירת הטפסים יתבצע במקום מרוכז, בצורה שתתמוך באופן מירבי בהוספה עתידית של טופס נוסף בצורה נוחה.

בגלל כל הסיבות הנ"ל בחרנו לממש את תהליך יצירת הקבצים בעזרת ‘**Factory method’** שכן תבנית זו פותרת בדיוק את הבעיות ההנדסיות שציינו.

* **אופן המימוש:**

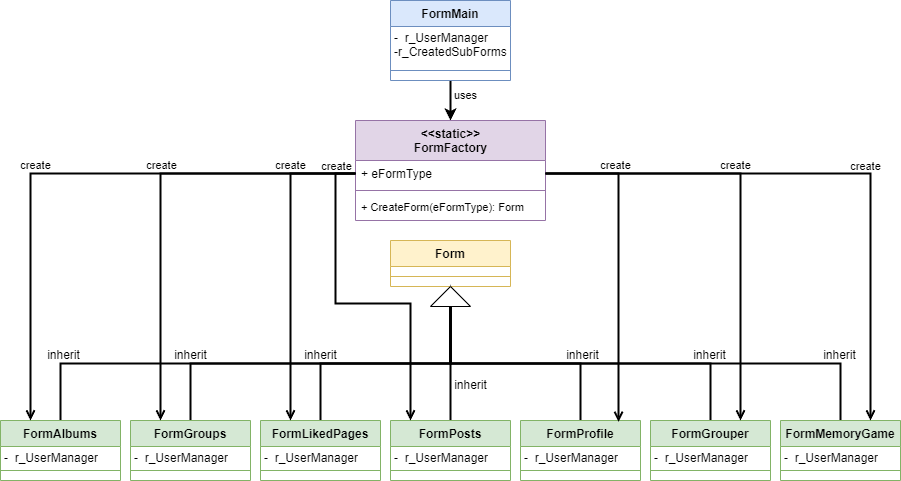
תהליך יצירת תתי הטפסים נעזר בתבנית ‘**Factory method’** באופן הבא:

* יצרנו מחלקה סטטית ‘***FormFactory’*** שנמצאת תחת הפרויקט ‘***FacebookAppWinformsUI***’ שתשמש כרכיב שיוצר את הטפסים השונים ומחזיר אותם (הFactory). המחלקה מחזיקה **Enum** בשם ‘***eFormType’***, שם שמורים סוגי הטפסים אותם המפעל יכול לייצר, ומטודה סטטית בשם ‘***CreateForm***’ שמקבלת ערך מ‘***eFormType’,*** ומחזירה מופע חדש מסוג הטופס המתאים.
* ה’***MainForm’*** משמש כclient, ומבקש מהמפעל ליצור את הטופס המתאים בהתאם לבקשת המשתמש.
* **Sequence Diagram**



* **Class Diagram**

כאמור ‘***FormFactory’*** היא המחלקה שמשמשת כ’מפעל’, הטפסים השונים הם המשפחה הפולימורפית, והטופס הראשי הוא הקליינט.



### 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