### הערה: ישנן תמונות של הדיאגרמות בנתיב : A24 Ex02 AmitCohen 316108224 DavidCohen 206265183\static\diagrams

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

* **Find Match** – מטרתו לעזור למשתמש למצוא match פוטנציאלי מבין חבריו לפי טווח גילאים ומשיכה מינית.

מימוש:  
צד משתמש – לחיצה על כפתור Find Match, פותח חלון של הגדרות פילטור. הזנת טווח גילאים ומשיכה מינית וביצוע תוצאות ולחיצה על כפתור Let's Go – מביא אותנו לForm הרצוי.

כל לחיצה על חבר מתוך הרשימה המפולטרת מביא לך את התמונה שלו/שלה ואת המידע אודותיו/אודותיה.

צד שרת – בדיקת עבור סטטוס מערכת יחסים, גיל ומין בלולאות.

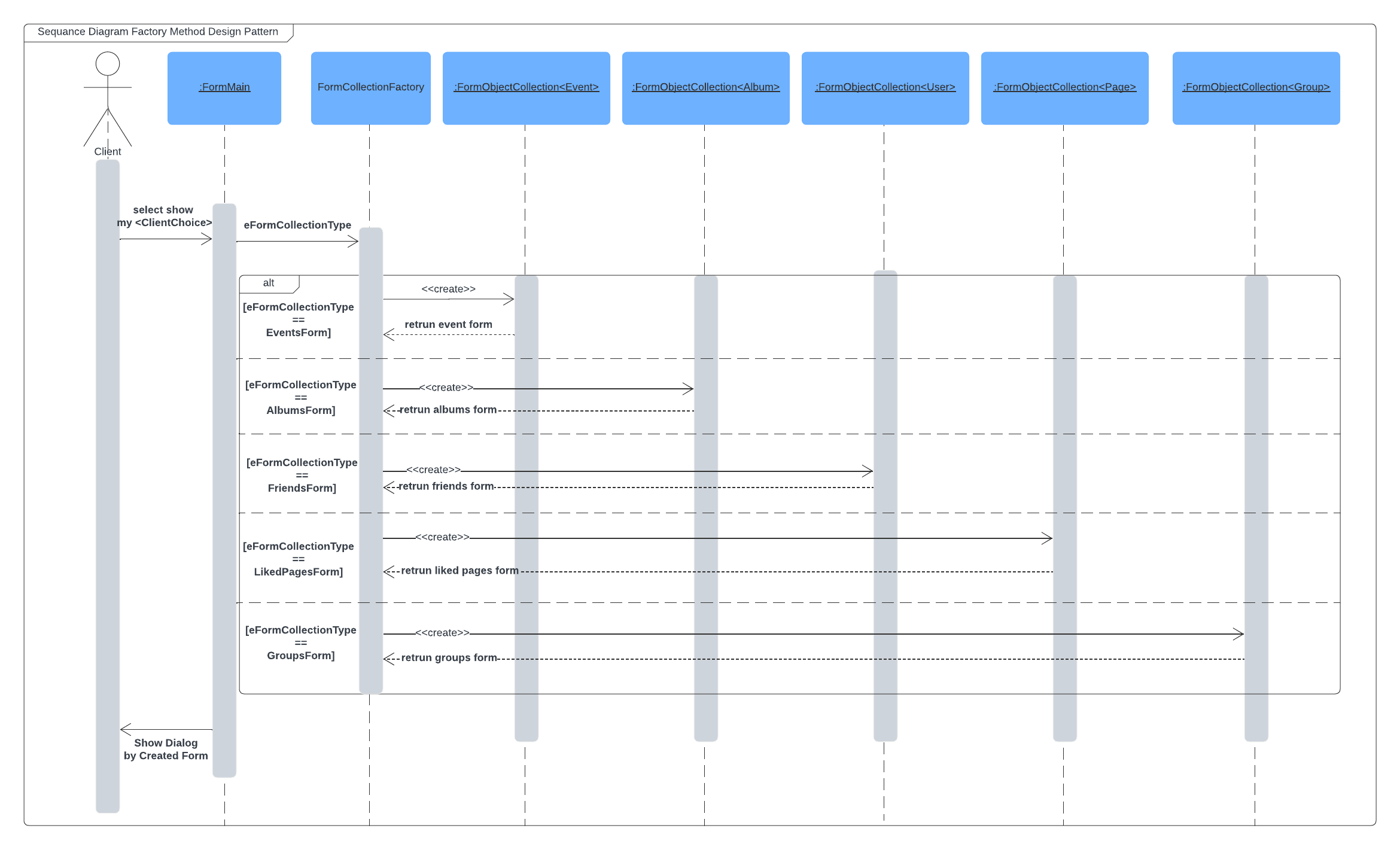
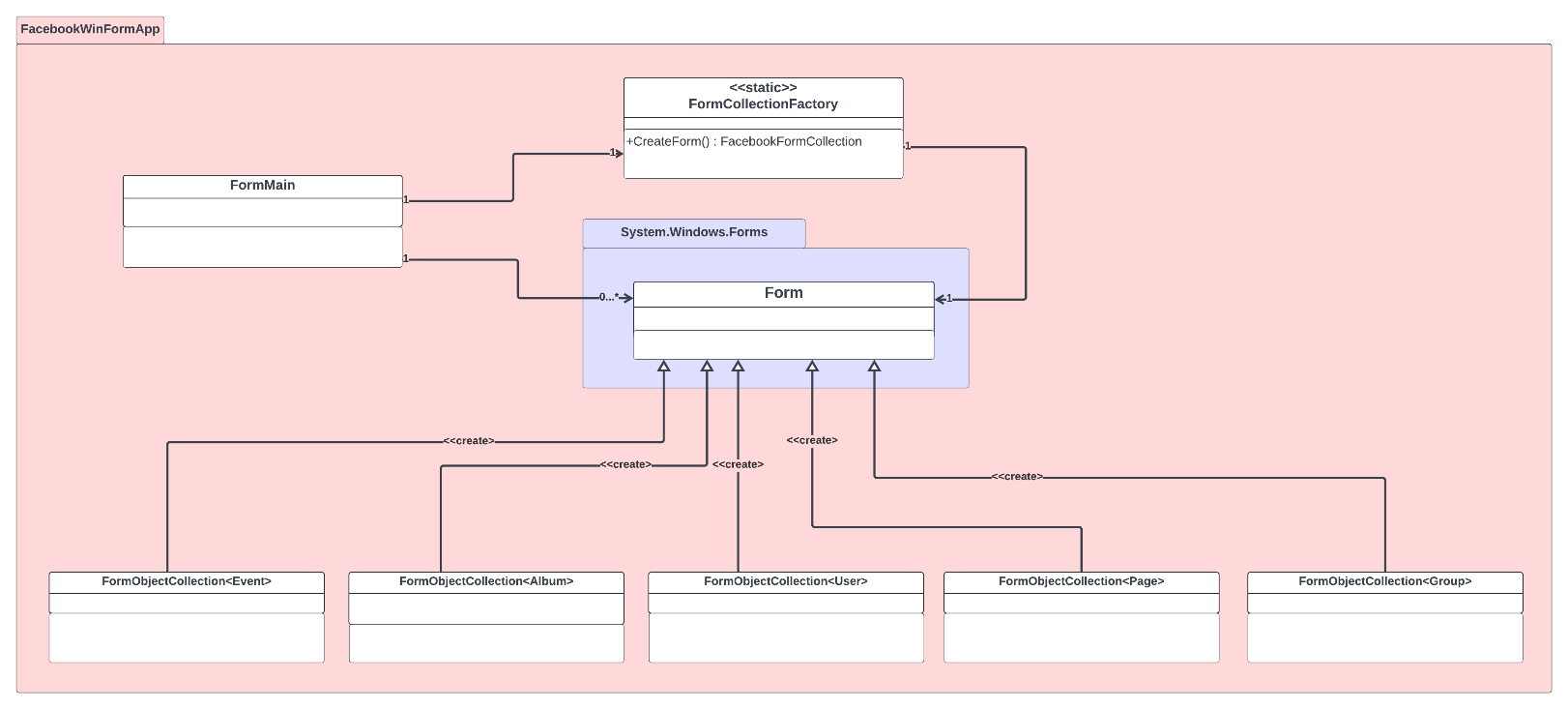
* **Find Things in Common with Your Friends** – מטרתו לעזור למשתמש לראות את כל הדברים המשותפים בין המשתמש לחבריו – דפים, חברים, איוונטים, וקבוצות משותפות.

צד משתמש – לחיצה על כפתור מתאים בForm Main , ובחירת חבר מתוך הרשימה של החברים.

עבור כל בחירת חבר יופיעו הדברים המשותפים בינו לחברו. בנוסף, ועבור כל לחיצה על איבר בlistBox בForm, תופיע תמונה מתאימה למשתמש (עבור חבר, דף, תמונה, אירוע וקבוצה).

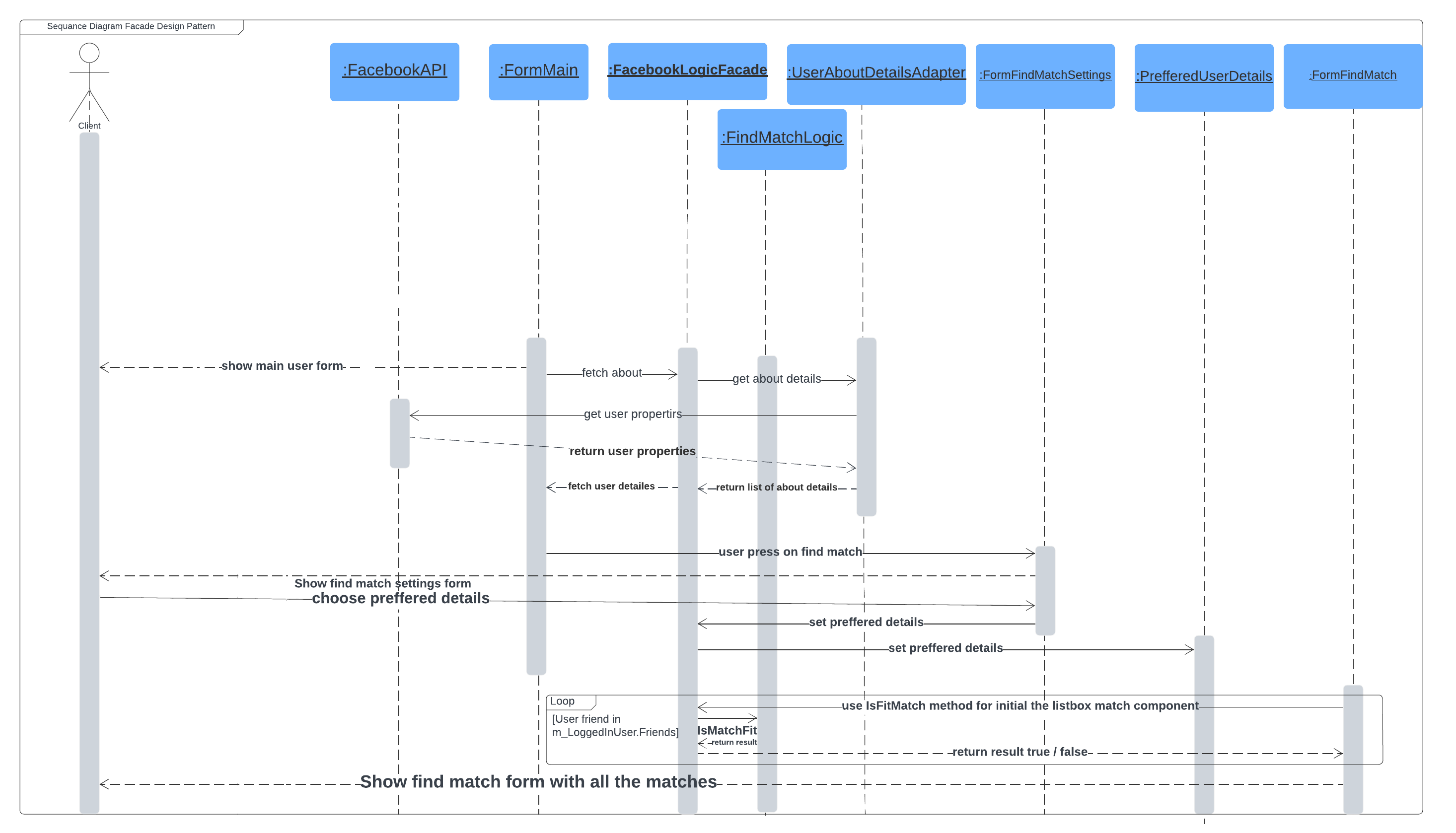
צד שרת – בדיקת דפים, חברים, איוונטים, וקבוצות משותפות ע"י לולאות.

### תבנית מס' 1 – Factory Method

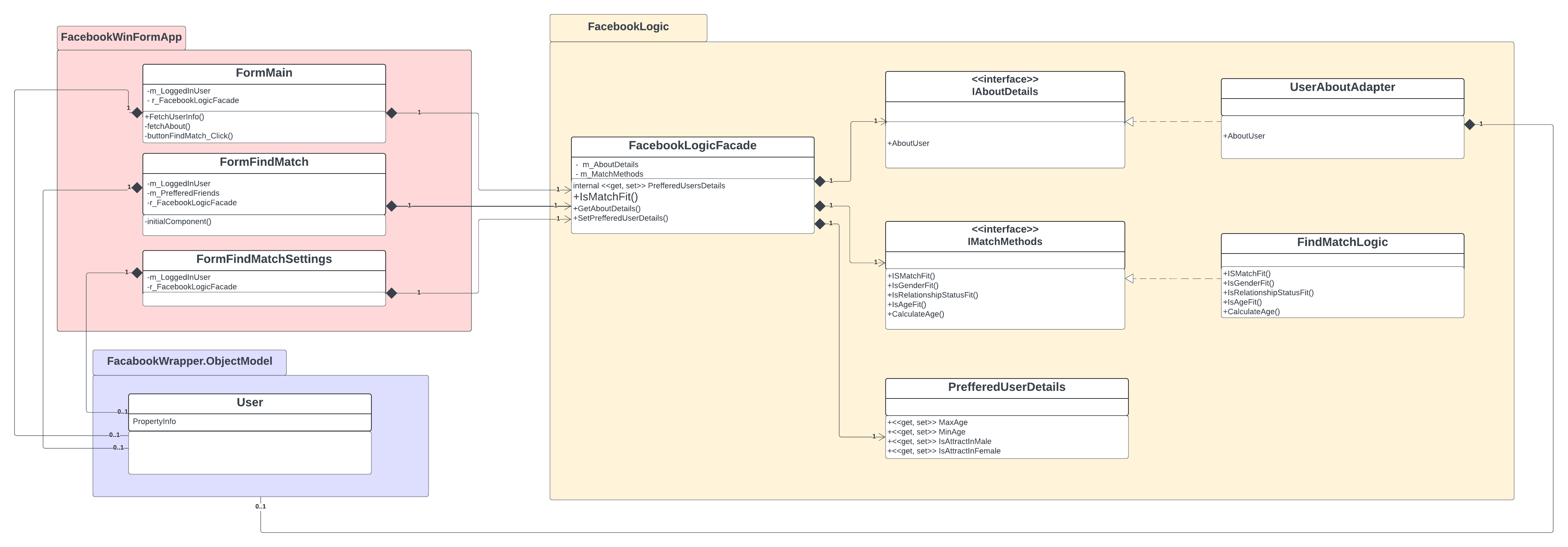
* סיבת הבחירה / שימוש בתבנית:  
  מטרת השימוש בדיזיין הנוכחי היא לייצר טופס כללי בעל מאפיינים זהים **עבור המשפחה הפולימורפית** שיש לנו בקוד – הלוא היא **FormObjectCollection**.  
  בנוסף, אנו רוצים **להפריד ולהסתיר מהClient את יצירת הטפסים** ע"י**Reusable Creational Method** – שתמנע מאיתנו שכפול קוד בעתיד בעל לוגיקה זהה.
* אופן המימוש:   
  יצירת מחלקה סטטית FormCollectionFactory המקבלת 2 פרמטרים:   
  1. m\_LoggedInUser - היוזר המחובר למערכת.  
  2. eFormCollectionType - enum אשר יהווה לנו סיווג עבור סוג הטופס.  
  ברגע שנלחץ על כפתור יצירה בטופס FormMain , נשלח את הפרמטרים הבאים לfactory והוא ייצר לנו טופס כבקשתנו.
* Sequence Diagram
* Class Diagram

### תבנית מס' 2 – Facade

* סיבת הבחירה / שימוש בתבנית:  
  בתכנית **קיים דו שיח משמעותי בין** הForms שהם **הUI, ללוגיקה**.  
  בחרנו **להקל על שימוש המשתמש** ברכיבים אלה **ע"י שימוש facade**.  
  יתר על כן, **הfacade תורם לבטיחות השימוש** במערכת כיוון שכעת לקליינט אין גישה לתתי המערכות בלוגיקה ("המנוע") אלא לאופרציות אותן המנוע יכול לבצע.
* אופן המימוש:  
   - **שם**: **FacebookLogicFacade**.  
   - **מיקום**: בpackage של .FacebookLogic **אנו משתמשים בו כdll**.  
   - **יצירה**: ע"י הFormMain   
   - מחלקות המשתמשות בו: FormMain ו FormFindMatch  
  **המחלקות בלוגיקה שהfacade עושה בהם שימוש הם internal – על מנת שנוכל לגשת לרכיבים אלה מהClient אך ורק דרך הfacade. כך נוצרת בטיחות השימוש והאנקפסולציה בו.**המתודות החשופות למשתמש דרך הfacade, הן רק המחלקות בהן הקליינט יכול להשתמש.  
  בנוסף, הfacade מחזיק אצלו מחלקות בהן הוא משתמש וטוען אותן בצורה lazy.
* Sequence Diagram

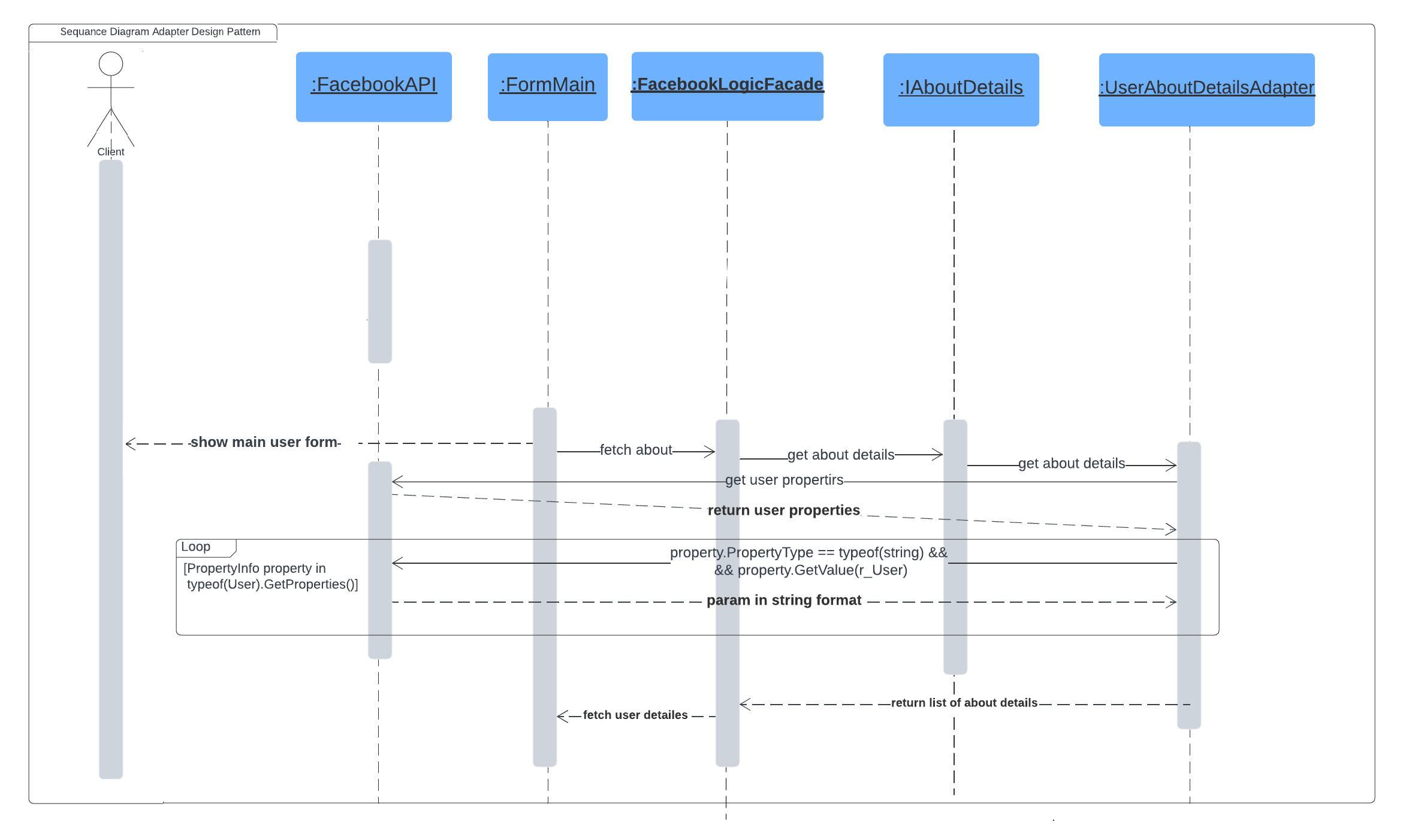


* Class Diagram

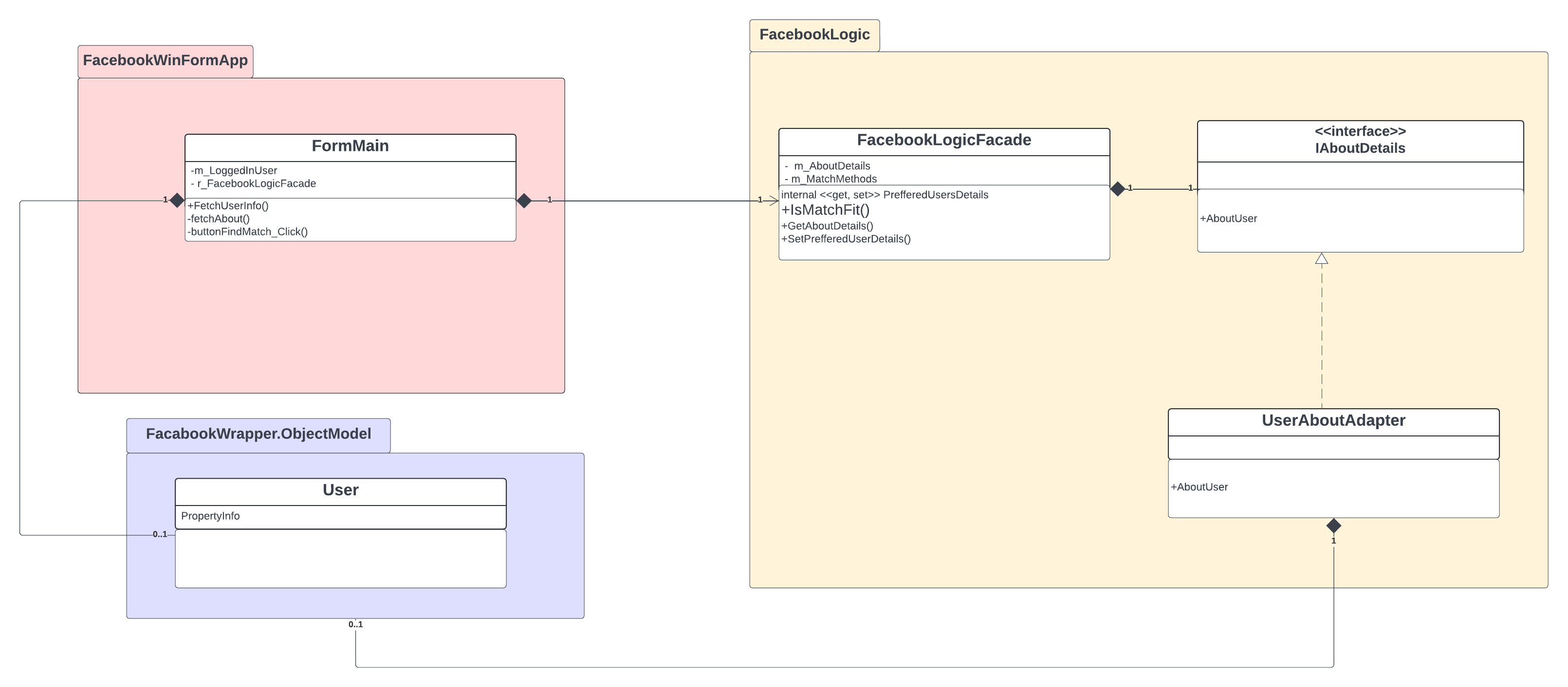


### תבנית מס' 3 – Adapter

* סיבת הבחירה / שימוש בתבנית:  
  כחלק מממשק המשתמש שלנו,   
  בחרנו להציג בFormMain את הAbout details של המשתמש.   
  על מנת להציג מידע זה, עלינו **לבצע פעולות והתאמות בין שני Legacy Components:**  
  1. ListBox  
  2. FacebookWrapper.ObjectModel.User  
  **אשר אינם תואמים ממשקית**.  
  **רצינו שהשימוש יהיה פשוט יותר** ויתבצע ע"י מעבר על רשימת about details של המשתמש, וכן **שתהיה מחלקה שתהווה חציצה בין שני ה Legacy Components, ושתעזור לנו להשיג יכולת פעולה הדדית בין הממשקים השונים.**
* אופן המימוש:  
  1. **יצירת Interface** – IaboutDetails  
  2. **מימוש** - במחלקת Adapter , UserAboutDetailsAdapter  
  3. **גישה** - דרך הfacade .  
  המחלקה adapter , מחזיקה אצלה את User (ה adaptee) ממנו היא לוקחת את המידעים של הabout , ומחזירה את כל המידעים האלה כרשימה של StringBuilders(בהתאמה ל groups ,pages.. שמחזירים רשימות של איברים) על מנת להקל את השימוש עבור המשתמש.
* Sequence Diagram

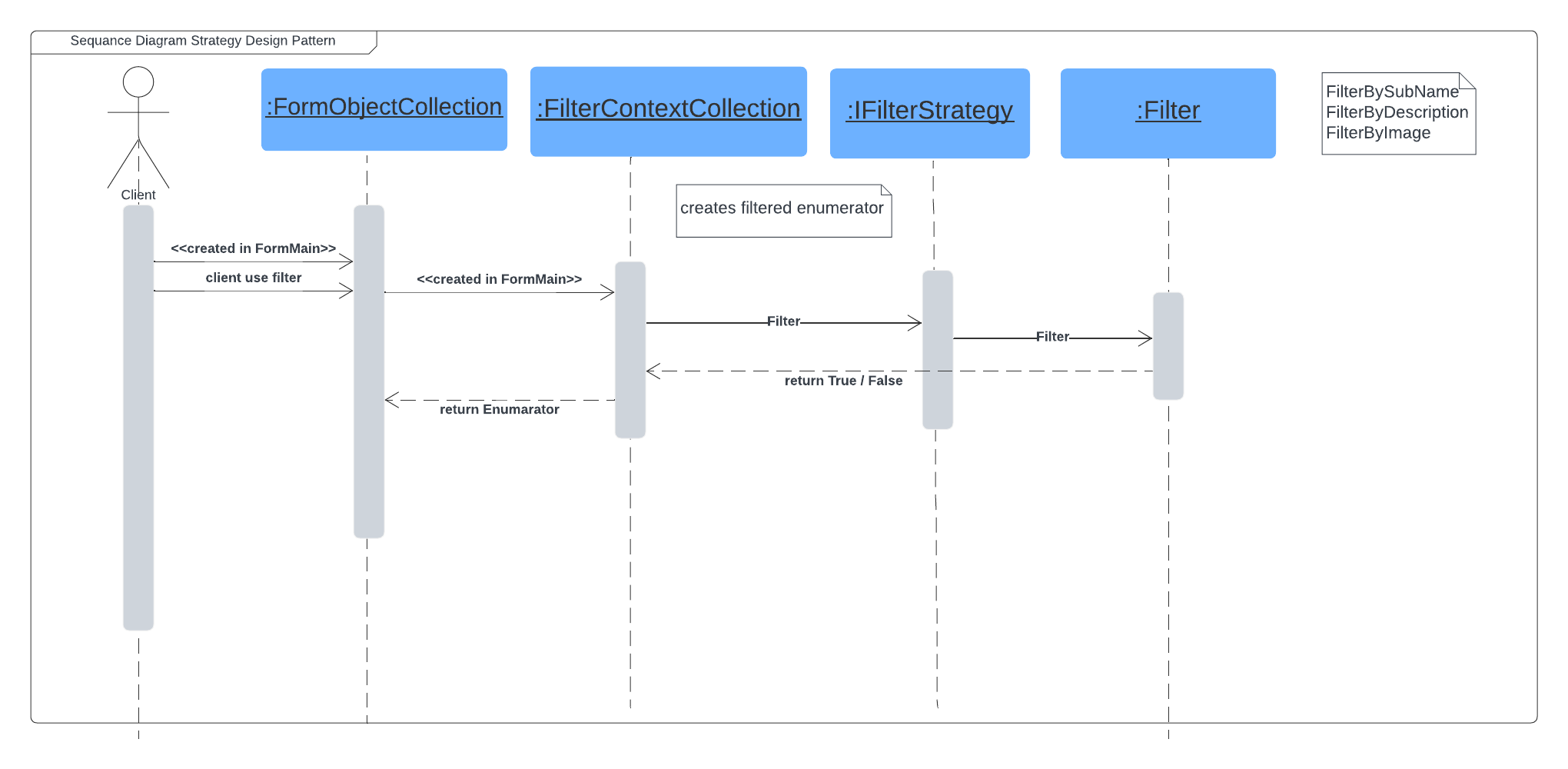


* Class Diagram

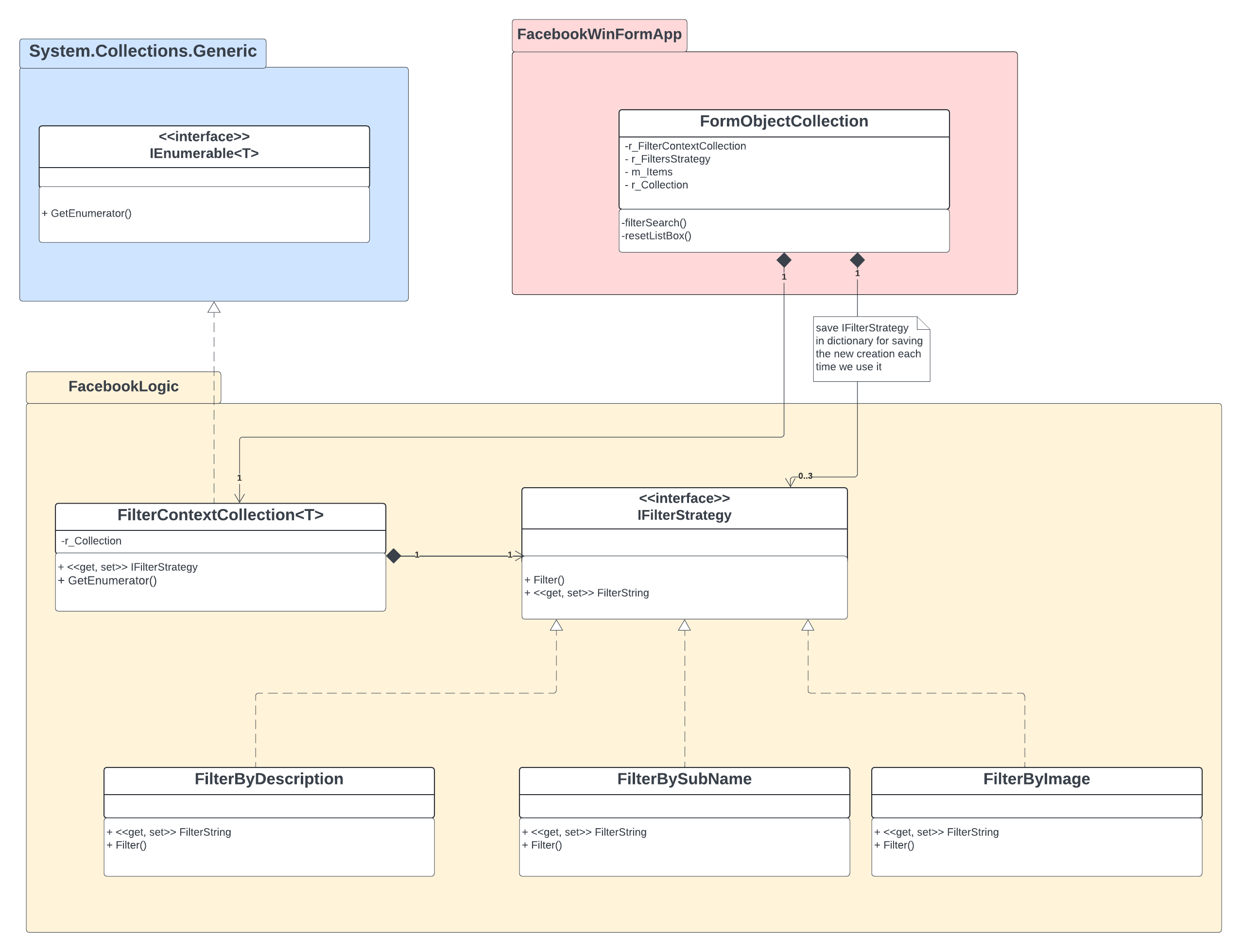


### תבנית מס' 4 – Strategy

* סיבת הבחירה / שימוש בתבנית:  
  **יצרנו פיצ'ר פילטור נוסף עבור אובייקטים** ב FormObjectCollection .  
  בפיצ'ר זה, אנו יכולים **לבצע חיפוש ברשימת בlistBoxCollection בעזרת תת שם, תיאור, ותמונה** (אם קיימת תמונה) .  
  לכן, **בחרנו להשתמש בstrategy במחלקה המממשת איטרטור, על מנת להחזיר איטרטור עם אופציית פילטור לפי בקשת המשתמש.**   
  **אם בעתיד נרצה להרחיב את אפשרויות הפילטור, או שנרצה להשתמש בפלטור במחלקות או קולקציות אחרות, השימוש בStrategy מהווה יסודות עבור הפעולות הללו.**
* אופן המימוש:  
  - **מימושinterface עבור strategy** - בשם **IFilterStrategy<T>**.  
   interface זה מחזיק מתודה בשם Filter , אשר מהווה את הפילטר הרלוונטי לפי בקשת המשתמש.  
  מחלקות המממשות את IFilterStrategy<T> :  
  1. FilterBySubName<T>  
  2. FilterByDescription<T>  
  3. FilterByImage<T>  
    
  בנוסף, אנו מחזיקים מחלקה בשם FilterContextCollection<T>.  
  מחלקה זו:  
  1. מחזיקה בקומפוזיציה IFilterStrategy<T>   
  2. מממשת getEnumarator()  
  3. מחזירה IEnumerator עם שימוש בפיצ'ר yield.  
  לפיכך, המחלקה FilterContextCollection<T> איטרבילית ומפלטרת ע"י שימוש ב .IFilterStrategy
* Sequence Diagram



* Class Diagram



עבודה אסינכרונית עם ממשק משתמש:

* השתמשנו באסינכרוניות בFormMain על **מנת לאפשר לממשק המשתמש להיות זמין למשתמש בזמן שמתבצעות הקריאות להביא את המידע מFacebookAPI.**בנוסף, בשימוש בנוכחי בmulti threading programming , **האפליקציה שלנו רצה מהר יותר כי יש יותר חיבורים עם הFacebookAPI אשר מביאים לנו מידע להצגה באפליקציה בו זמנית**.

אופן המימוש:  
- מיקום: בFormMain במתודה . fetchUserInfo  
יצרנו thread עבור כל משיכת נתונים מהשרת כך שקריאת הנתונים והפעולות שעושים עליהם (במידה ועושים) מתבצעות ב thread אחר ועדכון הנתונים בUI מתבצע בעזרת invoke של הפקד אותו נרצה לעדכן.  
באופן זהה, גם בפיצ'ר של FindThingsInCommon , בבחירת חבר מהרשימת חברים –  
 - כל thread יביא לנו מידע משותף בינינו לבין החבר הנבחר (קבוצות, דפים, חברים, איוונטים).

עבודה עם data binding:

* במחלקה FormFindMatch :  
  **השתמשנו בdataBinding עבור רשימת החברים הפונטציאליים לmatch .**  
  בכל פעם שאנו מסמנים חבר מהרשימה – יופיע לנו :  
  about, roundPictureBox (customize UI component), birthday and email.  
  נשים לב כי בבחירת הroundPictureBox – לא עבד לנו הdataBinding האוטומטי (אולי כי הוא customize..) אז ביצענו את זה ידנית. (רכיב שיצרנו שיורש מpictureBox אשר יוצר תמונה עגולה).
* במחלקות FormFindThingsInCommon , FormObjectCollection :  
  יצרנו את בdataBinding בצורה ידנית מכיוון שאנו מייצרים טופס טמפלייטי בהתאם לבקשת הקליינט, וכן FormFindThingsInCommon יורש מ FormObjectCollectionומבצע הרחבה לטופס .  
  כמו כן, לא שינינו ב FormFindThingsInCommonאת הDataBinding הידני בין בחירת פריט מהlistBox לroundPictureBox, מכיוון שהdataBinding האוטומטי לא עבד לנו עבור פריט customize כבר בשימוש לעיל במחלקה בFormFindMatch.