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

* הצגת ה- Pages למקומות להם היוזר עשה check-in לפי קטגוריות.

לדוג': מסעדות, בארים, מוזאונים וכו'.

* הצגת רשימת ה- Pages למקומות לפי קטגוריות שאחד החברים עשה check-in, היוזר יכול לראות רשימת חברים באפליקציה ובנוסף תוצג רשימת מקומות (check-ins) לפי קטגוריות, בהם היה היוזר והחבר מרשימת החברים היו ביחד.

### תבנית מס' 1 – [Singelton]

* סיבת הבחירה / שימוש בתבנית:

לא נרצה שיהיה instance נוסף מסוג ApplicationFactory

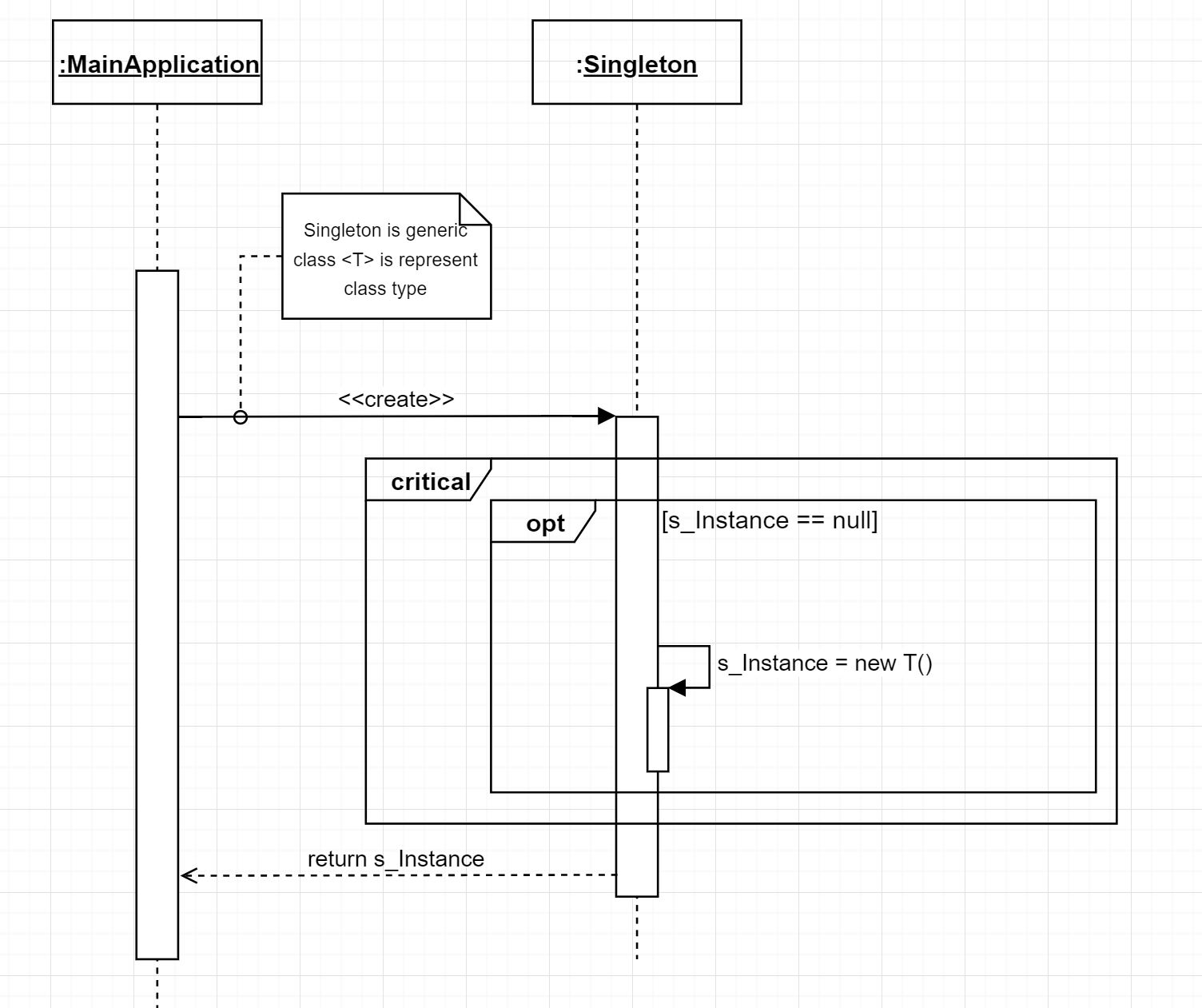
כיוון שנרצה ישות אחת בלבד שמטפלת בבניית אפליקציות. תחזוקת מערכת נוחה במקום יחיד.

* אופן המימוש:

יצרנו generic Singelton class כדי לאפשר בעתיד יצירה נוחה של Singelton classes.

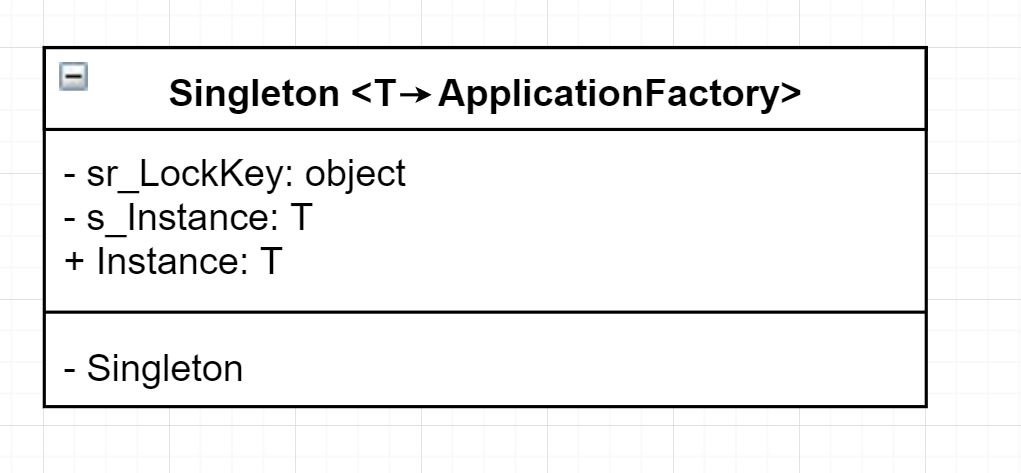
ניתן למצוא בקוד ב- MainApplicationForm ששם יצרנו ApplicationFactory instance: שהוא מסוג Singelton

* Sequence Diagram



* Class Diagram

המקבילה שלה: ApplicationSettings



### תבנית מס' 2 – [Factory Method]

* סיבת הבחירה / שימוש בתבנית:

סיבת בחירה: להטיל אחריות לייצירת אפליקציות רק במקום אחד. אם נרצה בעתיד להוסיף אפליקציות חדשות, ישנו מקום אחד האחראי ליצירת אפליקציה חדשה. (תחזוקה, קריאות, הרחבת מערכת)

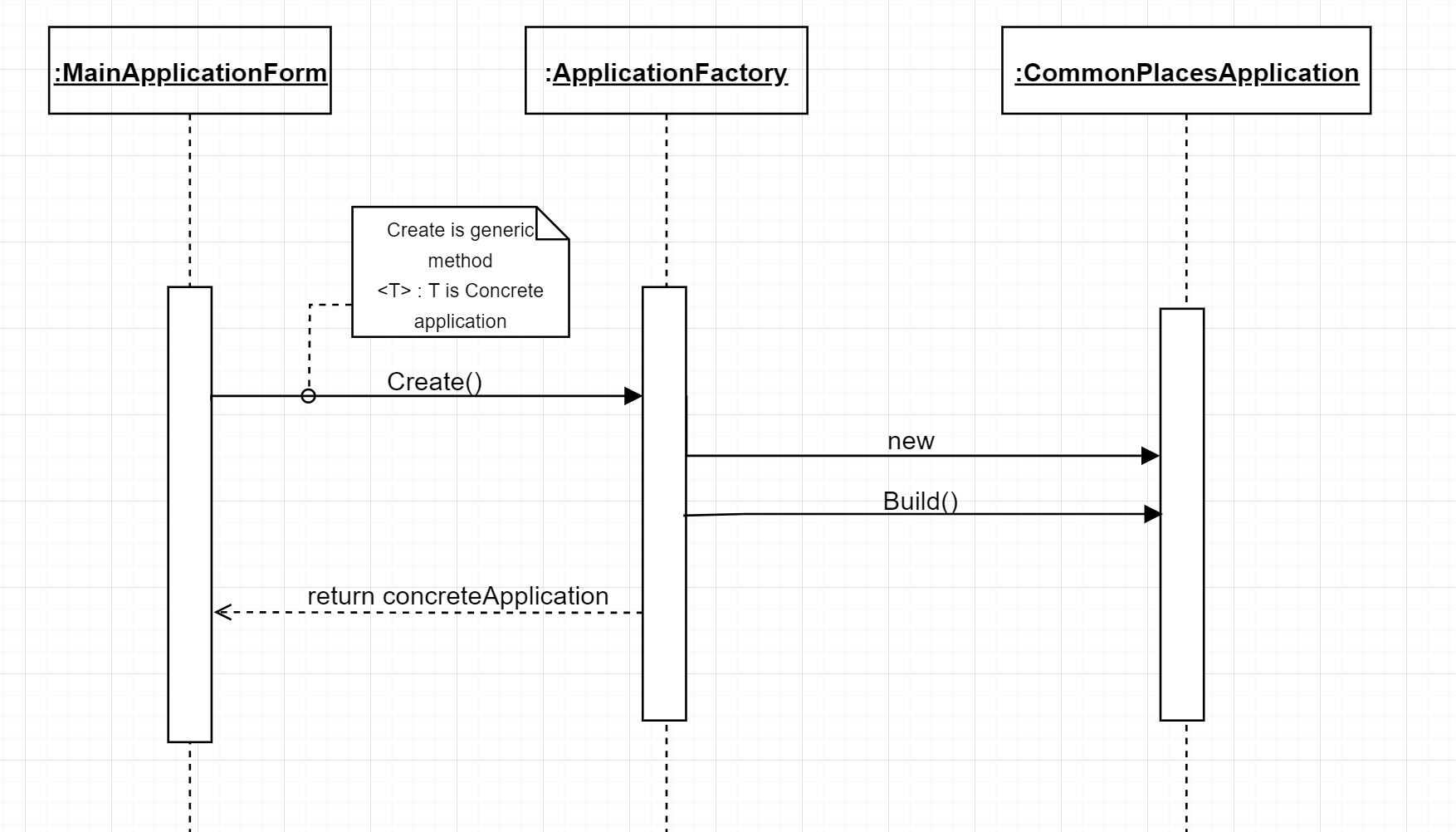
* אופן המימוש:

ApplicationFactory: אחראי על בניית האפליקציה

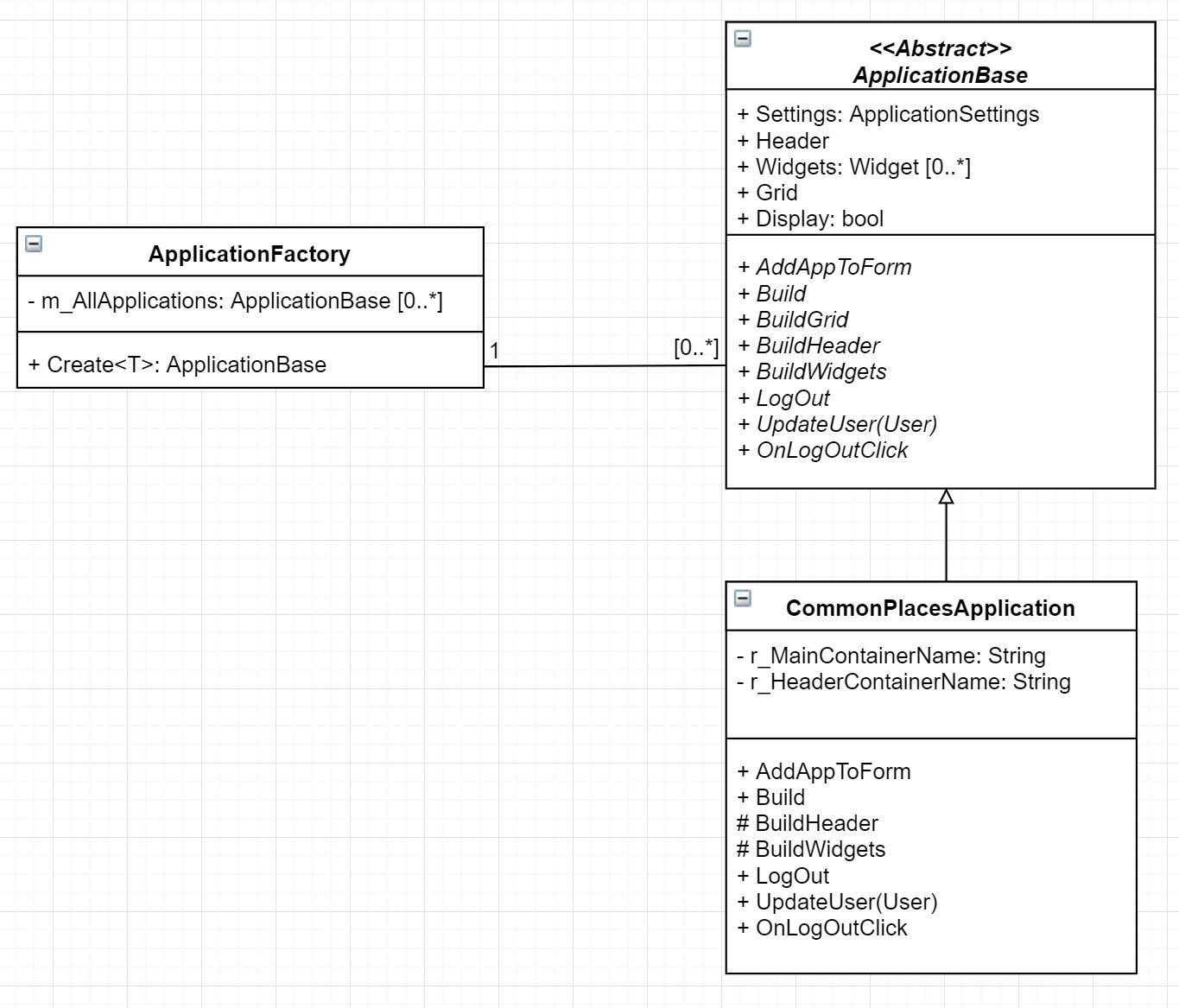
abstract class: ApplicationBase – התבנית של מטודות כלליות (default methods) כולל מימוש מטודות כלליות הניתנות להרחבה בעת הצורך של האפליקציה.

CommonPlacesApplication: אפציקציה קונקרטית אותה יוצר ה- ApplicationFactory.

* Sequence Diagram



* Class Diagram



### תבנית מס' 3 – [Proxy]

* סיבת הבחירה / שימוש בתבנית:

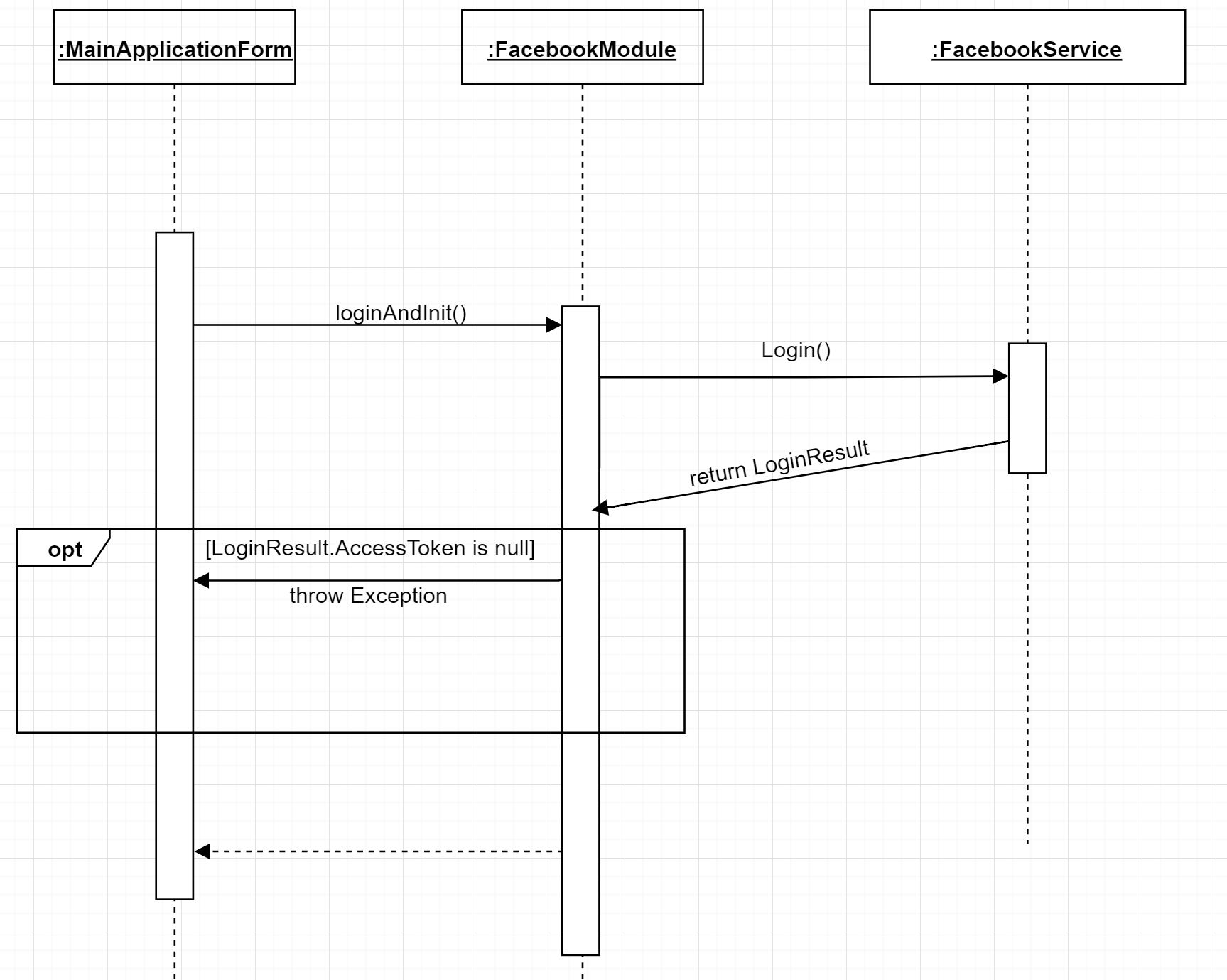
סיבת הבחירה: יכולת שליטה בהתחברות ל- facebook ושליטה ב- facebookService על מנת ליעל performance ולהקנות אפשרות של הוספת/הרחבת פונקציונליות של ה- facebookService

* אופן המימוש:

שימוש ב- facebookService class בעמצאות FacebookModule class חדש מהגדרנו.

ה- MainApplicationForm משתמש בשירותיו של FacebookModule.

* Sequence Diagram



* Class Diagram

