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

### תבנית מס' 1 – Singletone השתמשנו בתבנית ה Singleton כדי לממש את מחלקת FacebookManager במטרה לוודא שיש רק מופע אחד של התחברות לפייסבוק בכל רגע נתון באופן thread-safe. פעולה זו מבטיחה שיהיה רק משתמש אחד מחובר לפייסבוק בכל הזדמנות, ומונעת את האפשרות שמספר משתמשים יתחברו בו זמנית. בנוסף, השימוש בתבנית Singleton מאפשרת השגת התחברות יחידה למערכת ושימוש בה במחלקות אחרות בפרויקט, ומפטיר מהן מהצורך לבצע התחברות מחדש.

.

### תבנית מס' 2 – Builder

בחרנו לממש את התבנית הזו כדי לאפשר למשתמשים להגדיר את האובייקטים שהם רוצים לראות במערכת לאחר הרשמתם לאפליקציה. המשתמשים יהיו יכולים לבחור אילו פריטים מסוימים יהיו חשופים להם באמצעות הComposer וממשק המשתמש (Builder) ייוועץ בהתאם להעדפותיהם. כך יתווספו פחות הזדמנויות לעומס על השרת ונמנע יצירת אובייקטים שאין בהם באמת צורך.

### תבנית מס' 3 – Static Factory

תבנית זו מאפשרת יצירת אוביקטים פולימורפים באמצעות מחלקה סטטית בשם StaticFormFactory. תבנית מונעת שכפול קוד ומאפשרת יצירת אוביקטים עתידיים בצורה נוחה יותר עם הפרדה לבין שכבת הלוגיקה לבין יצירת האוביקטים.

### תבנית מס' 1 – Command

* סיבת הבחירה / שימוש בתבנית:

Command Pattern נבחר על פני שימוש בשיטת switch במתודה "CreateForm" במחלקת static factory ובמקום זאת בנינו מחלקת Dynamic Form Factory מכמה סיבות:

Extensibility: תבנית הפקודה מאפשרת הוספה קלה של פקודות חדשות (במקרה זה, סוגי טפסים חדשים) מבלי לשנות את הקוד הקיים. הוא תומך בעיקרון פתוח-סגור, כלומר ניתן להרחיב את המערכת מבלי לשנות את מבנה הליבה שלה.

Flexibility: אנו יכולים לשנות, להחליף או להרחיב פקודות בודדות באופן עצמאי מבלי להשפיע על חלקים אחרים של בסיס הקוד.

Maintainability: דפוס הפקודה משפר את תחזוקת הקוד על ידי בידוד יצירת הטפסים משאר האפליקציה, מה שמאפשר הוספת טפסים חדשים בעתיד ללא שינוי במחלקת Dynamic Form Factory.

* אופן המימוש:

בקוד, דפוס ה-Command מיושם כדי לנתק את היצירה של צורות שונות (אלבומים, דפים ופוסטים) מה-StaticFormFactory וכדי לקדם עיצוב מודולרי וניתן לתחזוקה יותר. להלן הסבר כיצד מיושמת תבנית הפקודה בקוד ובאילו מחלקות:

יישום דפוס Command:

מחלקות המעורבות בתבנית:

א. AbstractForm (מחלקה אבסטרקטית):

מחלקה אבסטרקטית זו מגדירה את הממשק המשותף לכל מחלקות הטפסים.

בתוכו קיימת המתודה Execute, שאחראית על ביצוע הלוגיקה הספציפית של יצירת הטופס.

ב. AlbumsFormCommand, PagesFormCommand, PostsFormCommand (Concrete

Classes)

מחלקות אלו יורשות מ- AbstractFormCommand.

כל מחלקה מממשת את שיטת ה-Execute כדי ליצור סוג מסוים של טופס ומוכלת בתוך מחלקת הProduct של כל אוביקט בהתאמה (AlbumsFormProduct, PagesFormProduct ו-PostsFormProduct).

ג. DynamicFormFactory (Client Class):

המחלקה DynamicFormFactory משמשת כלקוח בתבנית Command.

הוא שומר על אוסף של פקודות טופס (IFormCommand) במילון, כאשר סוג הטופס (eFormType) הוא המפתח, והפקודה היא הערך.

שיטת RegisterForm משמשת לרישום פקודות טופס קונקרטיות עבור סוגי טפסים שונים.

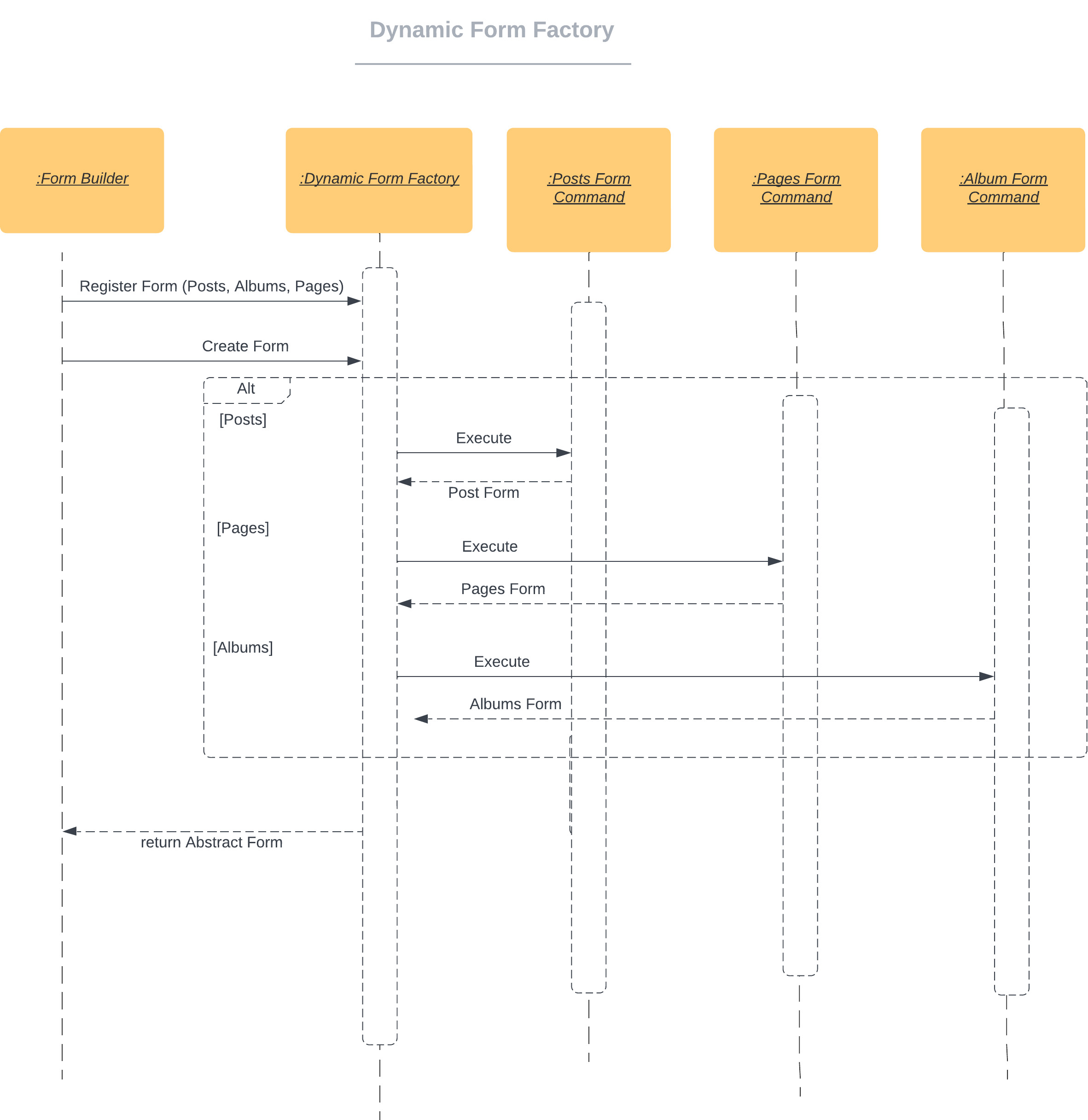
שיטת CreateForm לוקחת סוג טופס כקלט ומשתמשת בפקודה המשויכת ליצירת הטופס המתאים.

ד. FormBuilder (שימוש בתבנית בקוד)

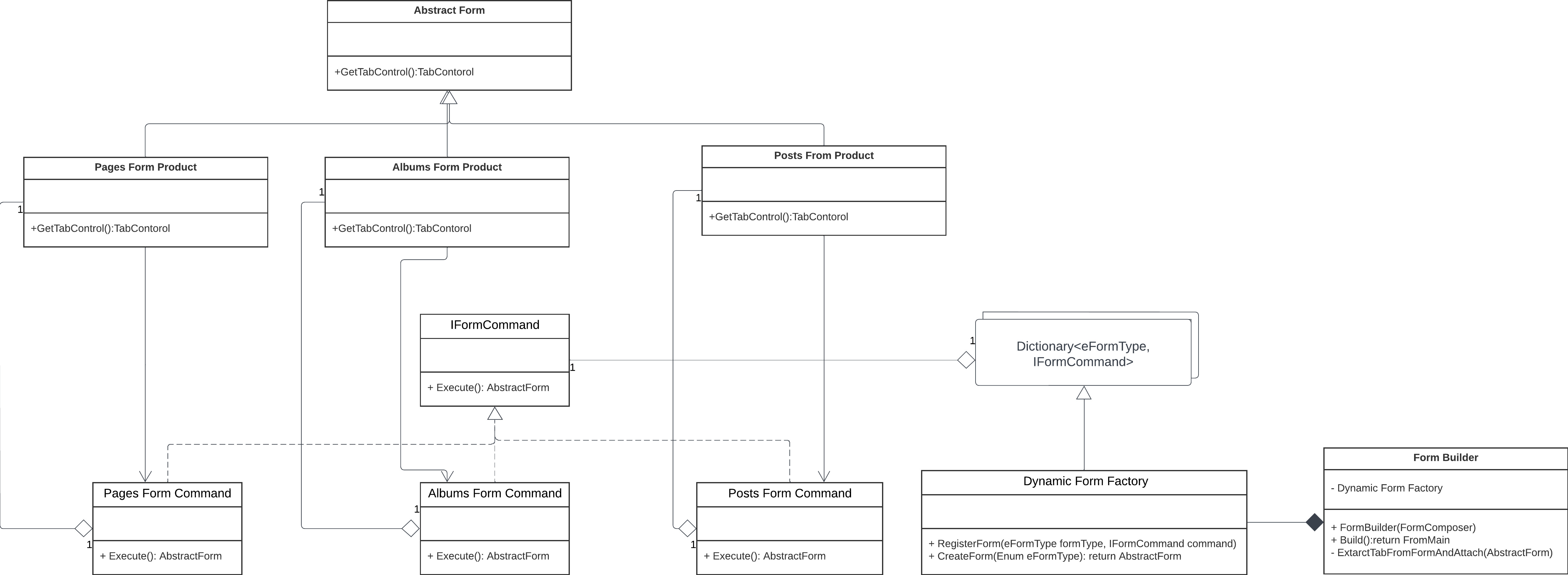
אנו משתמשים ב DynamicFormFactory לצורך בניית הטפסים במחלקה זו.

בconstructor מתבצע רישום הטיפוסים של כל הטפסים הרלוונטים (אלבומים, דפים ופוסטים) ובמתודה Build מתבצע בניית הטפסים לפי הגדרת הComposer.

* Sequence Diagram



* Class Diagram



### תבנית מס' 2 – [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

### 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