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

* אינפורמציה סטטיסטית : פיצ'טר שאוסף ומציג למשתמש מידע סטיטסי על הפעילות שלו כגון : כמה לייקים קיבל המשתמש על פוסטים, מה הפוסט שקיבל הכי הרבה לייקים, מה האלבום שקיבל הכי הרבה ליקיים, מה התמונה שקיבלה הכי הרבה לייקים
* מוד מסיבה : פיטצ'ר שנותן למשתמש אפשרות לעשות מסיבה בעמוד הפרופיל שלו. הפיטצ'ר משמיע למשתמש מוזקית מסיבות עדכנית וגורם לעמוד הפרופיל להתחיל לרקוד (משנה צבעים מזיז פקדים)

### תבנית מס' 1 – Strategy Iterator

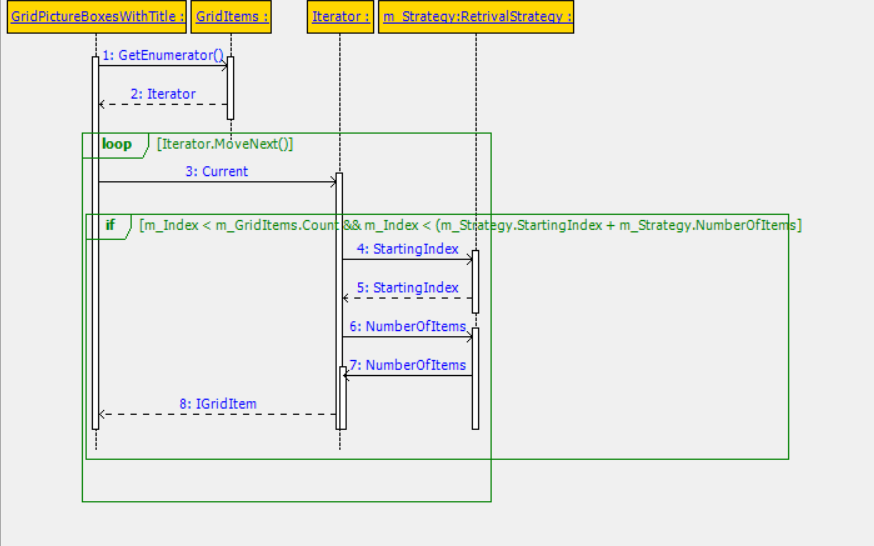
* סיבת הבחירה / שימוש בתבנית:

במערכת שלנו יש רכיב שמכיל קולקציה של איברים. רכיב זה יכול להציג מספר משתנה של של פריטים מהקולקציה ולכן ממישנו איטרטור אם אסטרטגיה שמחזירה לרכיב בדיוק את הפריטים שהוא רוצה להציג

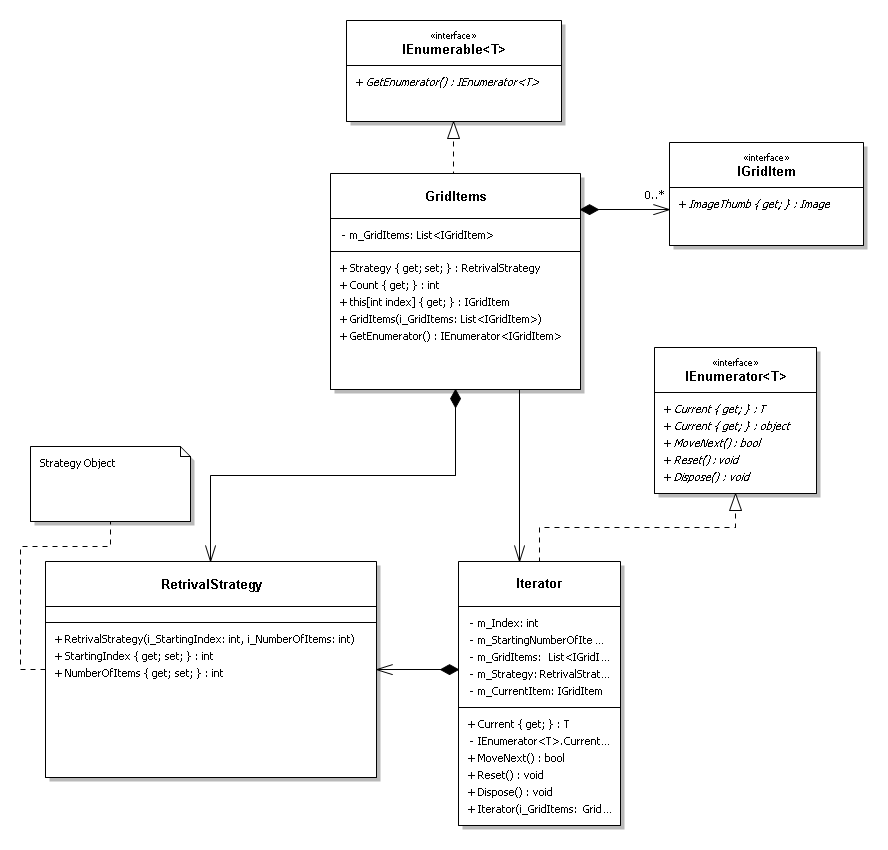
* אופן המימוש:

מומשה מחלקה GridItems אשר ממשת ממשק IEnumerable ומספק Iterator שמומש פנימית בתוך המחלקה

* Sequence Diagram



* Class Diagram



### תבנית מס' 2 – Visitor

* 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 – Observer

* 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