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

* הפיצ'ר הראשון מאפשר למצוא את כל הסרטים, ספרים או מוזיקה המשותפים בין המשתמש האפליקציה לבין כל אחד מהחברים שלו. בעצם, לאחר שהמשתמש בוחר באחד משלושת האופציות ולוחץ על הכפתור תוצג לו רשימה של דברים שהיא איחוד של כל החיתוחים בין המשתמש ובין כל אחד מהחברים שלו.כאשר המשתמש בוחר משהו ברשימה תוזג לו תמונה במידה והיא קיימת.
* הפיצ'ר השני מחשב יחס בין בנים לבנות עבור חברים של המשתמש.בהינתן לתוצאת החישוב תוצג למשתמש הוגעה מתאימה.למשל, אם המשתמש הוא ממין זכר ויש לו לפחות 2/3 חברות מתוך ס"כ חברים אז תוצג לו הודעה:

"Well done!!!Girls love you!!!"

### תבנית מס' 1 – [Strategy Pattern]

* סיבת הבחירה / שימוש בתבנית:

הפיצ'ר השני שלנו מבצע פעולת חישוב מסויימת. כגון, מחשב יחס בין בנים לבנות עבור חברים של המשתמש. אנו חשבנו והחלטנו שניתן לעשות את החישוב הנ"ל בצורה מדוייקת יותר תחת התחשבות ב.RelationshipStatus כלומר, אם חבר של המשתמש הוא ”In a relationships ” or "married" אז לא מתחשבים בו בחישוב. במילים אחרות, יצרנו שתי דרכים לחשב את היחס הנ"ל. כך הוצר צורך בשימוש בתבנית "Strategy". האפליקציה משתמשת באלגוריתם חישוב בהתאם לבחירה של המשתמש.

* אופן המימוש:

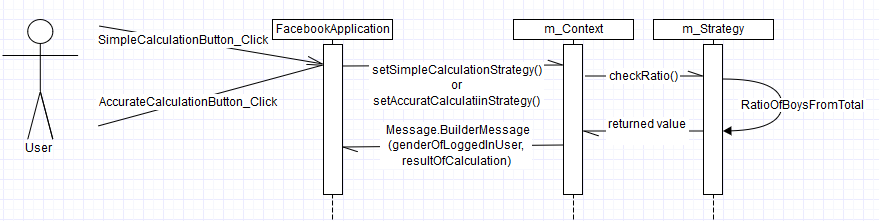
על מנת לממש את הצורך בשתי אלגוריתמי חישוב(כך שהמשתמש יוכל לבחור בינהם) יצרנו

abstract class Strategy שמכילה מתודה

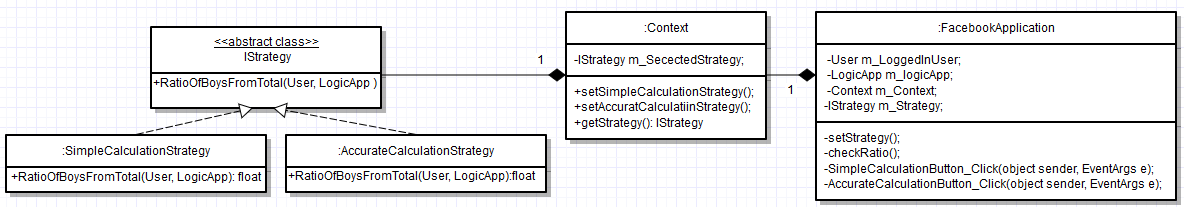
RatioOfBoysFromTotal(User i\_LoggedInUser, LogicApp i\_logicApp) כך ששתי הסטרטגיות (SimpleCalculationStrategy, AccurateCalculationStrategy) מממשות אותה. מחלקה Contextזאת מחלקה שבה בהתאם לבחירה של המשתמש (משתמש לוחץ על אחת משתי כפתורים שהם SimpleCalculation ,AccurateCalculation) מיוצר אוביקט מסוג

SimpleCalculationStrategy או AccurateCalculationStrategy וכתוצאה מכך מופעלת מתודה RatioOfBoysFromTotal מתאימה.

* Sequence Diagram



* Class Diagram



### תבנית מס' 2 – [Template method]

* סיבת הבחירה / שימוש בתבנית:

כדי לממש את ההפיצ'ר השני שמאפשר למשתמש לבחור בין "חישוב רגיל" לבין "חישוב מדויק" היינו חייבים להשתמש ב abstract class IStrategy שכולל בתוכו abstract method שאותו ביצעו באופן שונה שתי מחלקות שמימשו את abstract class הנ"ל.כלומר, המתודה האובסטרקטית

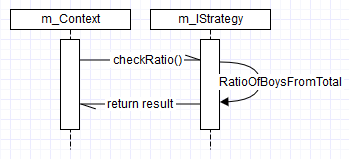
float RatioOfBoysFromTotal(User i\_LoggedInUser, LogicApp i\_logicApp)

שהיא בעצם Template Method.

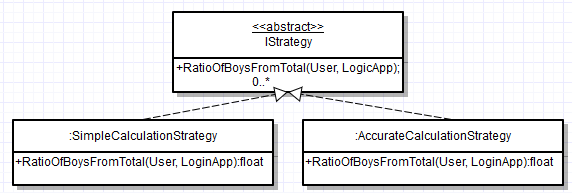
* אופן המימוש:

תיאור השימוש ומיקומו בקוד ניתן לראות בסעיף הקודם.

* Sequence Diagram



* Class Diagram



### תבנית מס' 3 – [Visitor]

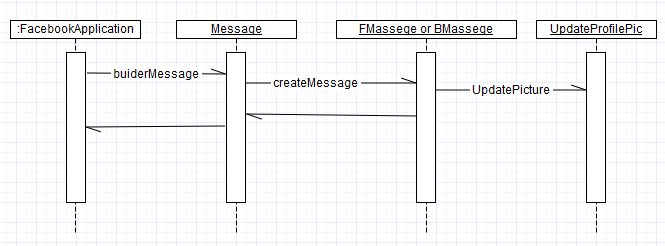
* סיבת הבחירה / שימוש בתבנית:

אנו הוספנו פונקציונליות נוספת לפיצ'ר השני באפליקציה שלנו שהיא שינוי התמונה של הפרופיל של האפליקציה בהתאם לתוצאה החישוב הנ"ל. כך נוצר צורך ליצור מחלקה שמממשת את הפונקציונליות הזאת עבור מחלקות שזקוקות לכך.

* אופן המימוש:

במחלקה FacebookApplication מופעלת המתודה checkRatio שקוראת למתודה BuilderMessege שנמצאת במחלקה Message,שהיא בעצם מפעילה מתודה CreateMessege עבור FMessege וBMessege שמממשות IMessege אינטרפיס. הפונקציונליות שהוספנו נמצאת במחלקה UpdateProfilePic שמוחזקת על ידי כל אחת מהן(FMessege,BMessege). כך, UpdateProfilePic הוא הVisitor .

* Sequence Diagram



* Class Diagram

