

מסמך ניתוח

אמרי שי – 213023500

בר יחזקאל – 314990938

שי גאלי – 315202242

עידו מנגדי – 315310250

חלק ראשון - System Screens Structure

רשימת מסכים:

מסכי אוטנטיקציה:

1. Login Screen (LoginFragment) - אפשר להתחבר דרכו לאפליקציה עבור משתמשים רשומים.
2. Registration Screen (RegisterFragment) – אפשר להירשם דרכו לאפליקציה.

מסכים שקשורים למשתמש:

1. Home Screen (HomeFragment) - המסך המרכזי של האפליקציה המכיל ניווט לכל התכונות העיקריות.
2. Profile Screen (ProfileFragment) - מציג מידע אישי של המשתמש, שינוי הגדרות וניתוק מהחשבון.
3. Admin Dashboard Screen (AdminDashBoardFragment) - ממשק ניהול למנהלי המערכת בו הם יכולים לנהל את האיזור עליו הם אחראים.
4. Chat (ChatFragment) – מסך שבו ניתן לנהל צאט
5. All chats (ChatsFragment) – מסך שבו ניתן לראות את כל הצאטים שהמשתמש ביצע.

מסכים שקשורים למוצר:

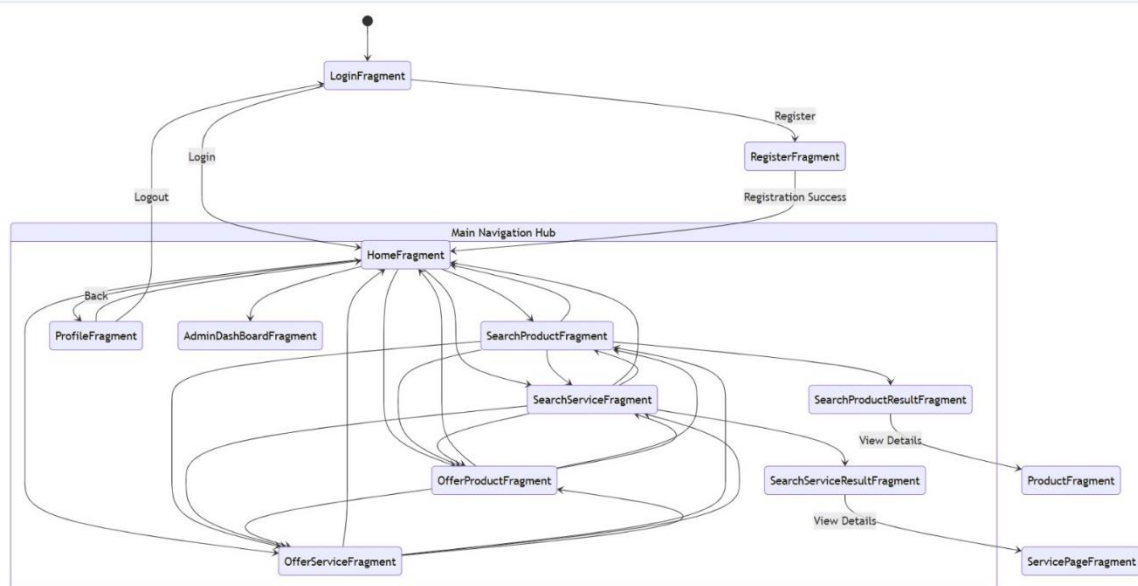
1. Search Product Screen (SearchProductFragment) - ממשק לחיפוש מוצרים זמינים במערכת.
2. Search Product Results Screen (SearchProductResultFragment) - מציג את תוצאות החיפוש של המוצרים.
3. Product Details Page (ProductPageFragment) - מציג מידע מפורט על מוצר ספציפי.
4. Offer Product Screen (OfferProductFragment) - ממשק המאפשר למשתמשים להציע/לרשום מוצרים.

מסכים שקשורים לשירותים:

1. Search Service Screen (SearchServiceFragment) - ממשק לחיפוש שירותים זמינים במערכת (כמו בעלי מקצוע שמעוניינים לעזור)
2. Search Service Results Screen (SearchServiceResultFragment) - מציג את תוצאות החיפוש של השירותים.
3. Service Details Page (ServicePageFragment) - מציג מידע מפורט על שירות ספציפי.
4. Offer Service Screen (OfferServiceFragment) - ממשק המאפשר למשתמשים להציע/לרשום שירותים

קשר בין המסכים:

נציג את המעברים הכללים בין כל המסכים ואז נסביר איך המשתמש עובר.

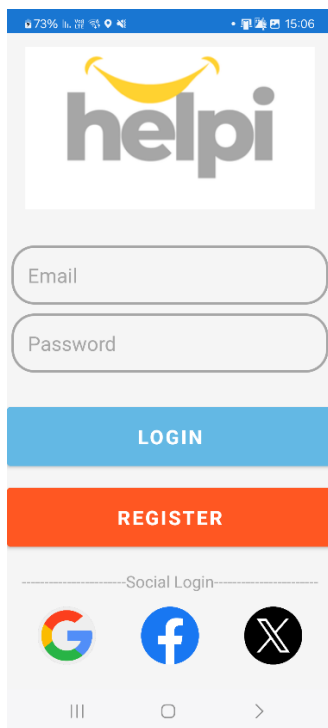


- בין המסך התחברות להרשמות ניתן לעבור בכפתור.
- כאשר המשתמש סיים את התחברות או הרישום הוא לוחץ על כפתור שבודק האם החיבור תקין, אם הוא תקין הוא יעבור למסך הבית
- יש לנו nav bar שמכיל את המסכים הבאים: Home, Search Product, Search Service, Offer
- Product, Offer service. כאשר משתמש לוחץ על האייקון המתאים הוא יעבור למסך המתאים.
- SearchProductFragment --> SearchProductResultFragment: כאשר המשתמש סיים לבחור את הסינונים עבור החיפוש הוא ילחץ על כפתור שיעבור למסך תוצאה
- SearchProductResultFragment --> ProductFragment: ברשימה המסוננת אפשר ללחוץ על אובייקט ברשימה ונעבור למסך כשיציג פירוט מלא עליו
- ProductFragment--> chat: ניתן ללחוץ על כפתור כדי לדבר עם המפרסם
- השלוש הקודמים תקפים גם לשירותים כמו למוצרים בצורה זהה
- מהפרופיל ניתן ללחוץ על כפתור כדי להתנתק
- אם המשתמש הוא מנהל אזור (או אדמין) יהיה לו כפתור שהוא יוכל לעבור אל הדאשבורד הניהולי שלו.

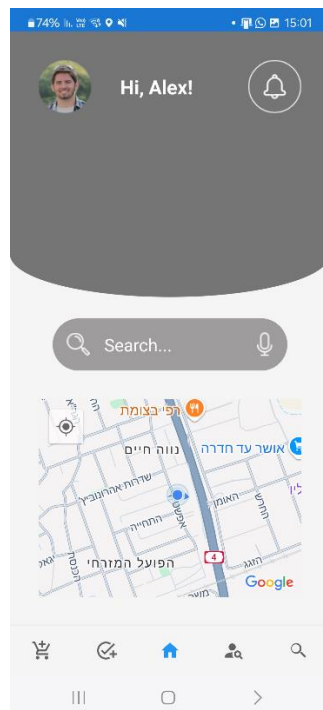
לק"י

הצגה ויזואלית:

1. LoginFragment

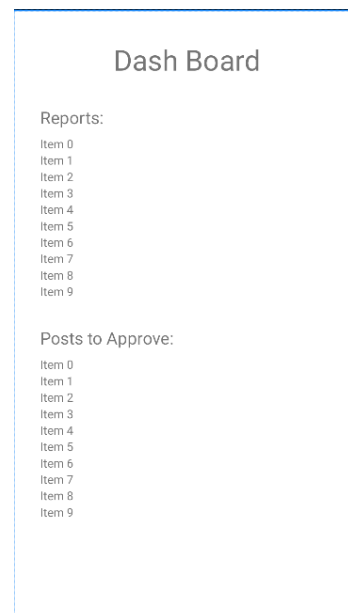


3. HomeFragment

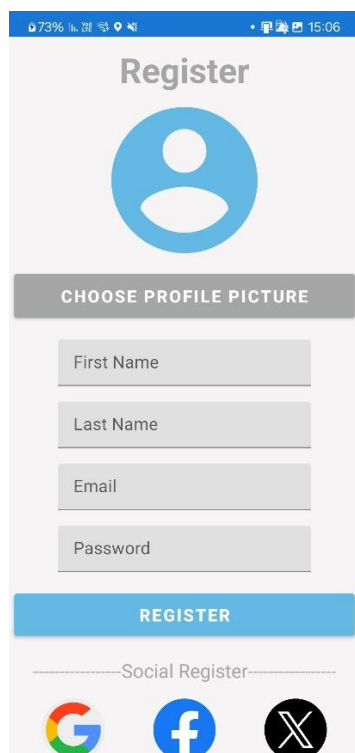


5.

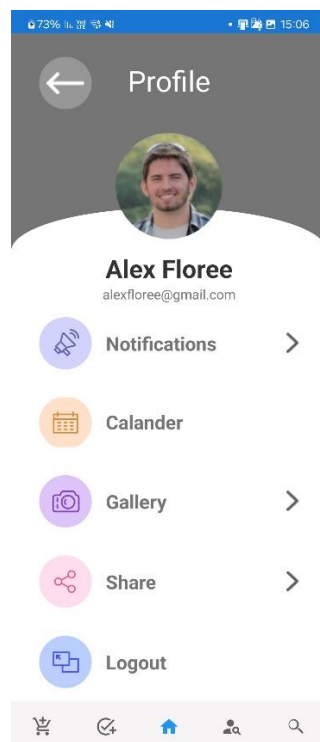
AdminDashBoardFragment



2. RegisterFragment

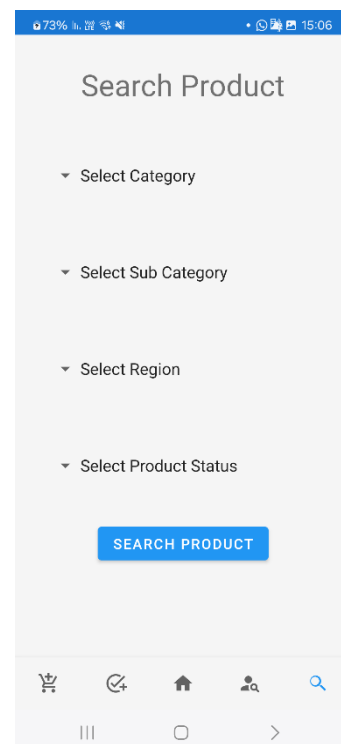


4. ProfileFragment

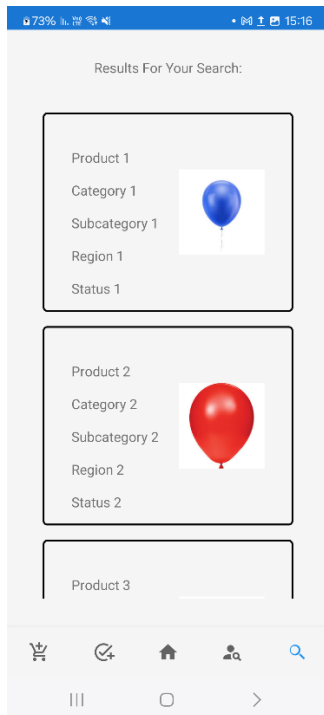


6.

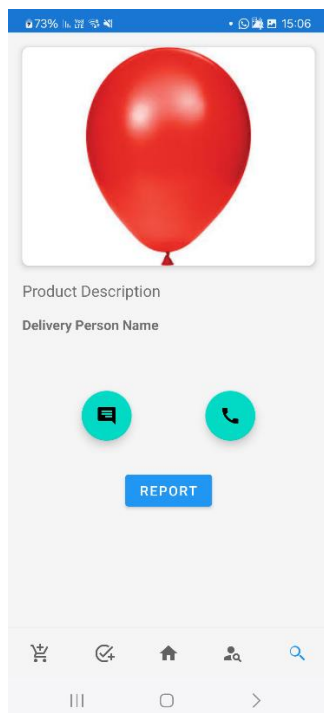
SearchProductFragment



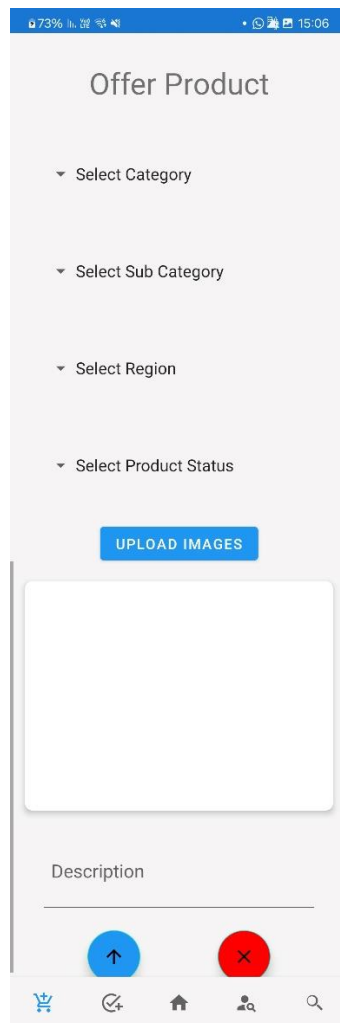
.7
SearchProductResultFragment



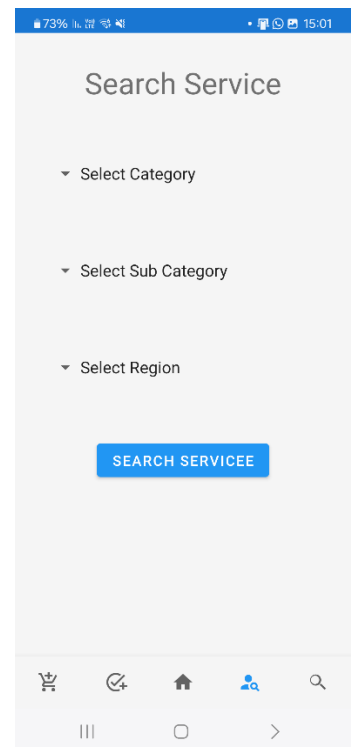
.8
ProductPageFragment



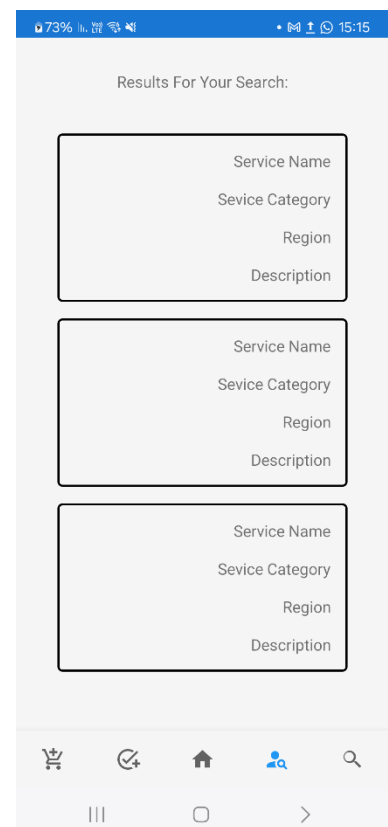
.9
OfferProductFragment



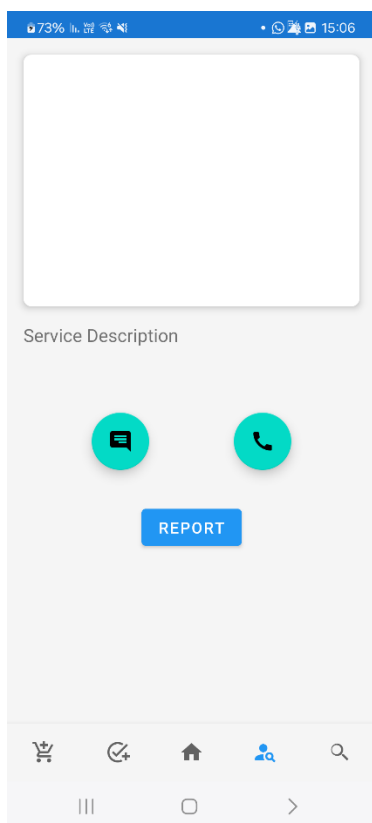
.10
SearchServiceFragment



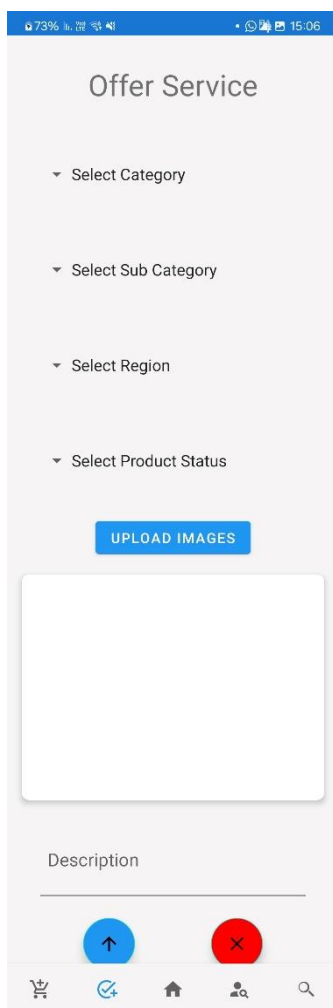
.11
SearchServiceResultFragment



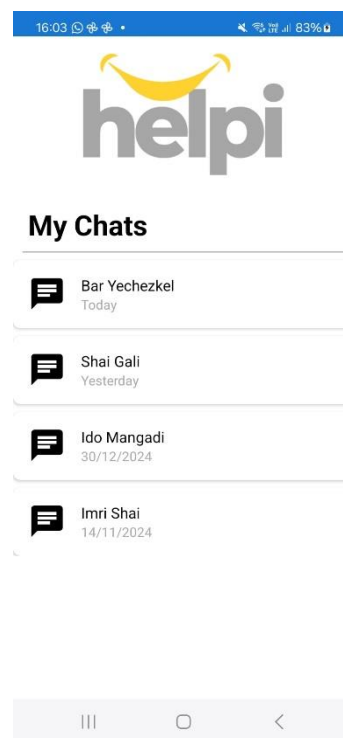
ServicePageFragment .12



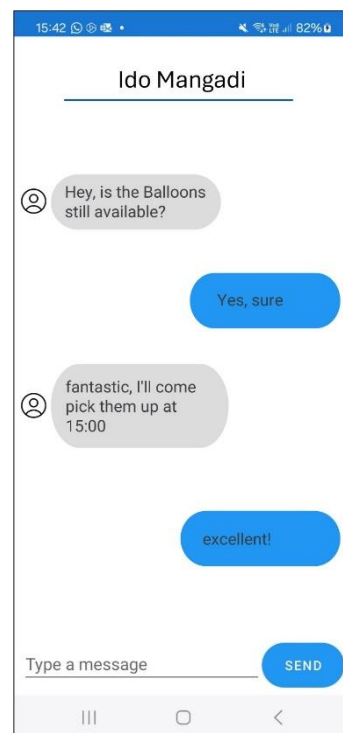
OfferServiceFragment .13



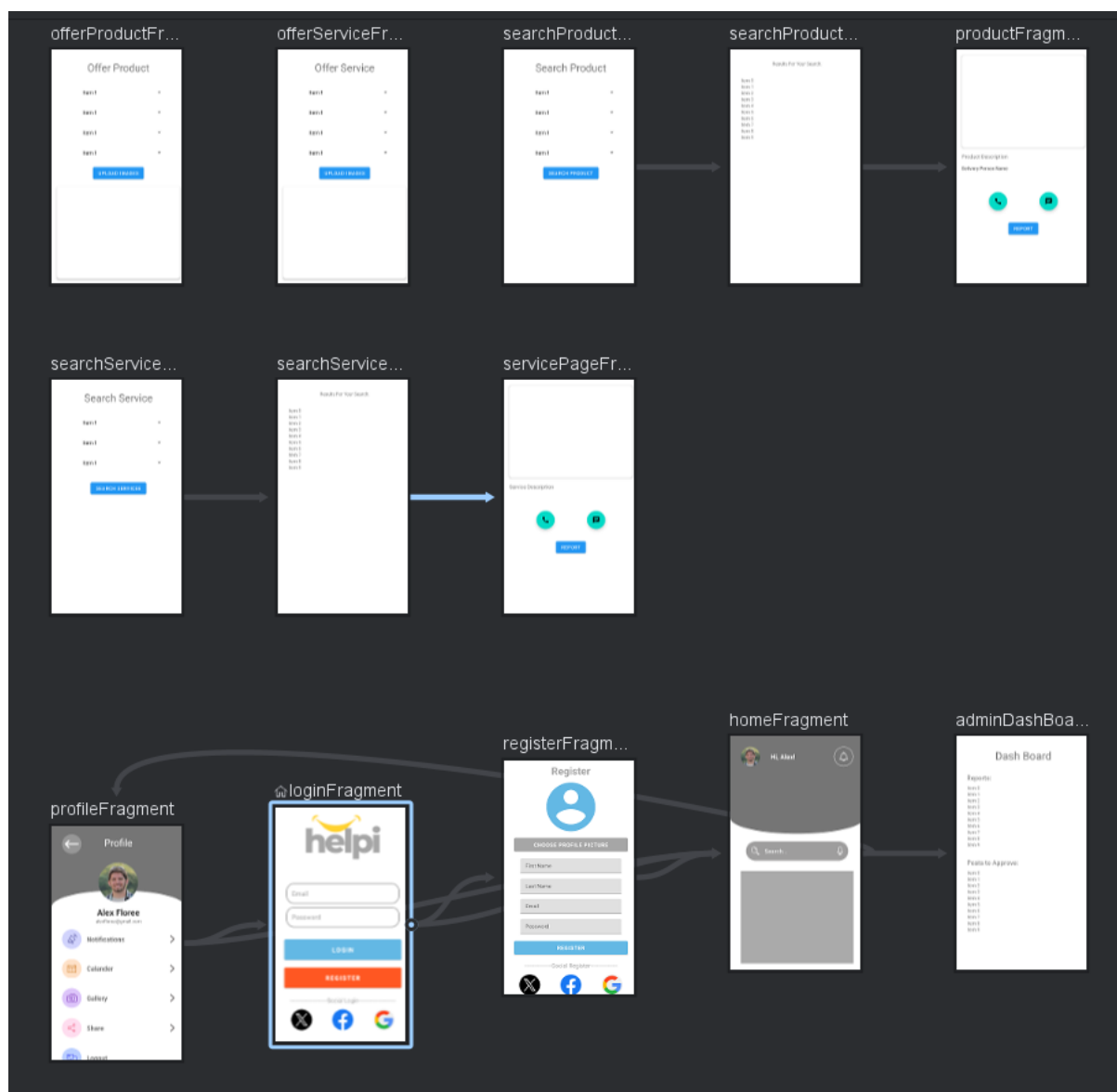
Chats.14



Chat.15



עוד מעברים מהאנדרואיד:



חלק שני - Design Patterns for Android Studio

הסבר על MVVM:

Model - ייצג את הנתונים והלוגיקה של האפליקציה, כולל גישה למסד הנתונים. יכול את המחלקות המייצגות את הישויות העיקריות באפליקציה, כגון: User, Item, Service וכו', וכן את הלוגיקה לאחזור והעלאה של נתונים מ/אל מסד הנתונים.

View - יהיה אחראי על הצגת המסכים השונים באפליקציה, כגון: מסך הבית, מסך חיפוש, מסך פרופיל וכו'. כל מסך יוגדר בקובץ XML בתיקיה res/layout.

ViewModel - יהיה אחראי על הכנת הנתונים לצורך הצגה ב-View. למשל: אחזור רשימת ה-Items או ה-services הרלוונטיים ממסד הנתונים, סינון על פי בקשת משתמש וכו'. בנוסף, ה-ViewModel יחשוף נתונים ל-View באופן ריאקטיבי באמצעות מנגנונים כמו LiveData ו-Flow.

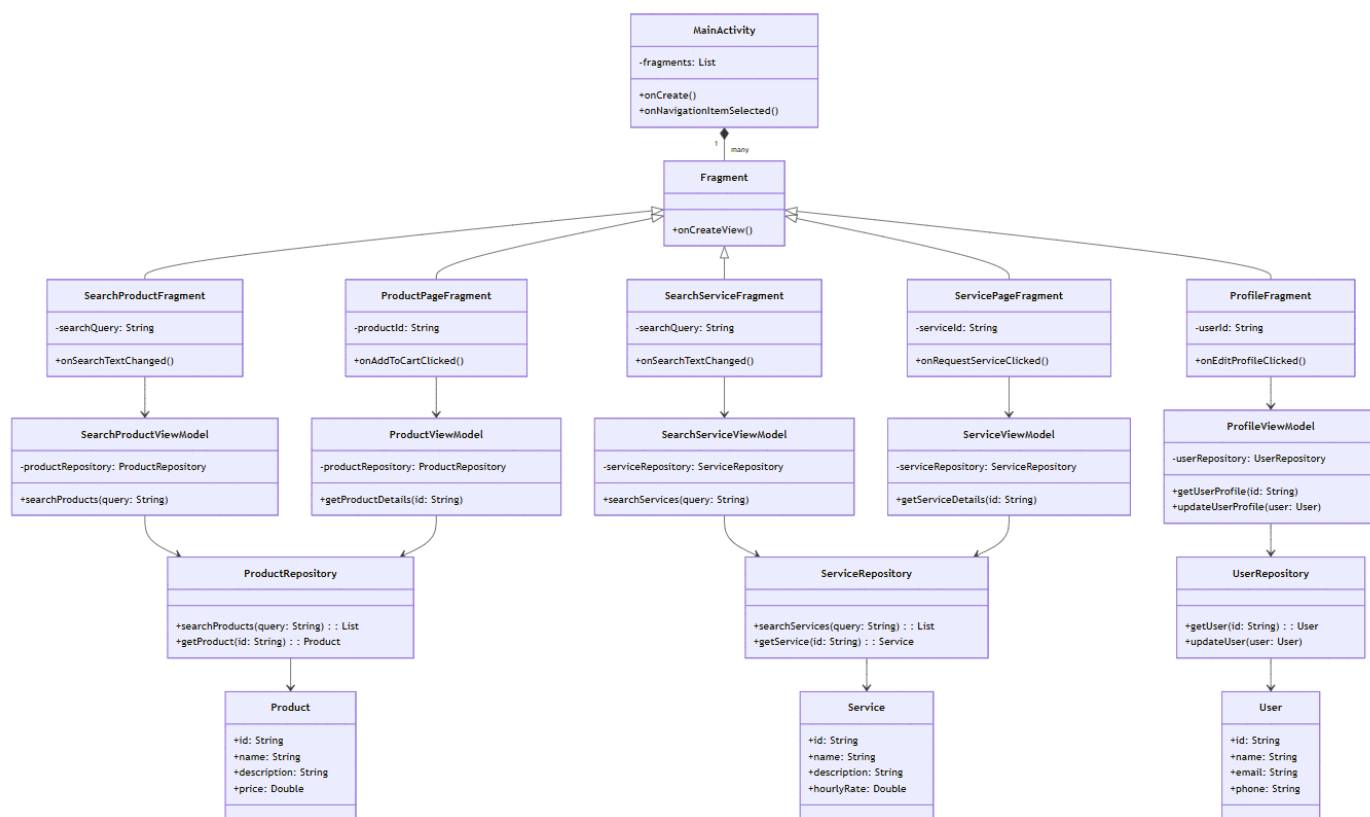
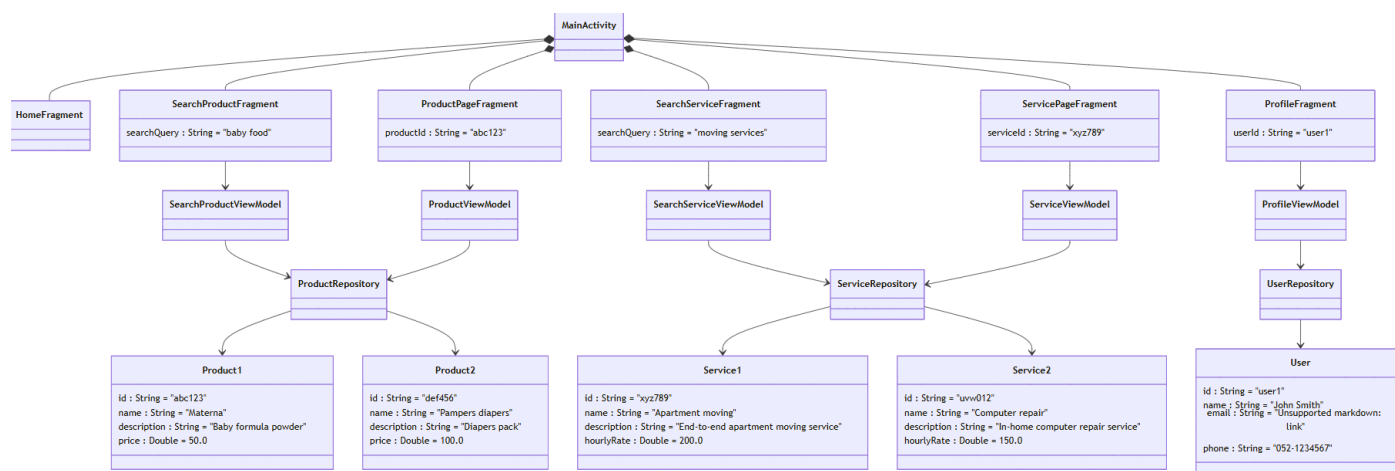
מדוע בחרנו אותו:

מודל MVVM מתאים לאפליקציה שלנו מכיוון שהוא מאפשר הפרדה ברורה בין שכבת הנתונים, שכבת ה-UI ושכבת הלוגיקה. המבנה המודולרי של MVVM יסייע בניהול קוד מאורגן ונקי, דבר חיוני כאשר מדובר בפרויקט מורכב עם ישויות מרובות כמו משתמשים, פריטים ובקשות כמו שלנו. בנוסף, ראינו שיש על המודל המון מדריכים ברשת והרבה מאוד אפליקציות עובדות בשיטה הזו והנחנו שזה יסייע לנו במהלך הפיתוח היות ורוב הצוות חסרי ניסיון בפיתוח אפליקציות אנדרואיד.

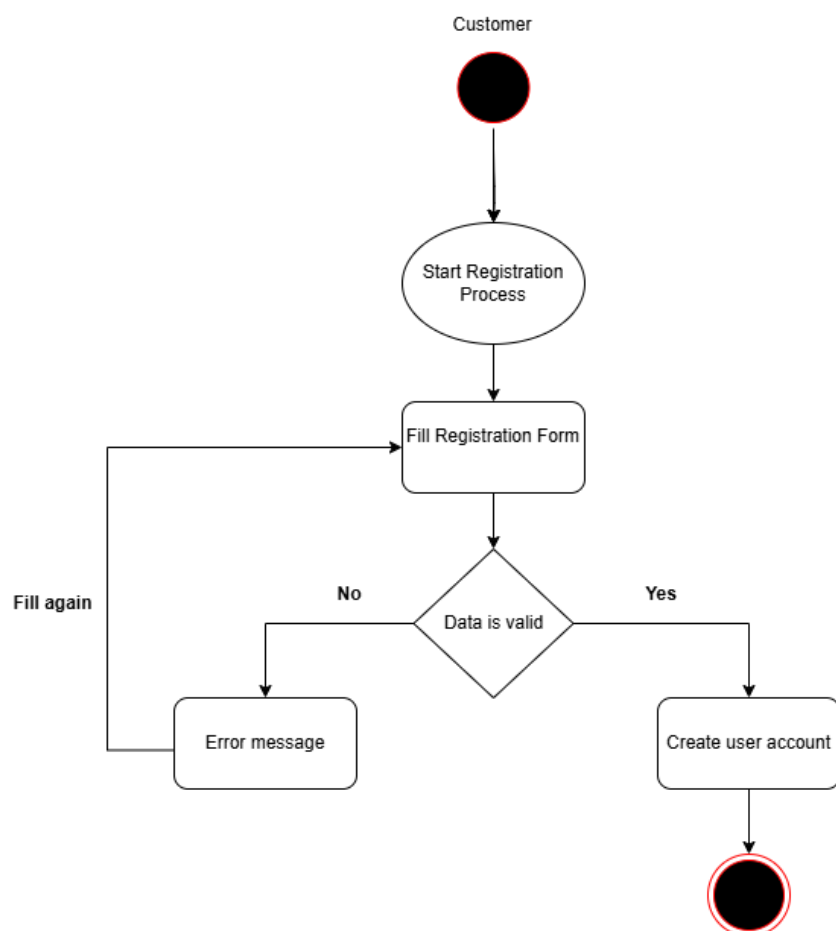
(ב- Class Diagram ניתן לראות את החלוקה ל- Fragment שהיא חלק מה- View, את המחלקות של ה- ViewModel, ואת המחלקות של ה- Model.

בנוסף אפשר לראות ב- Sequence Diagram את השכבות באות לידי ביטוי בתהליך המתואר)

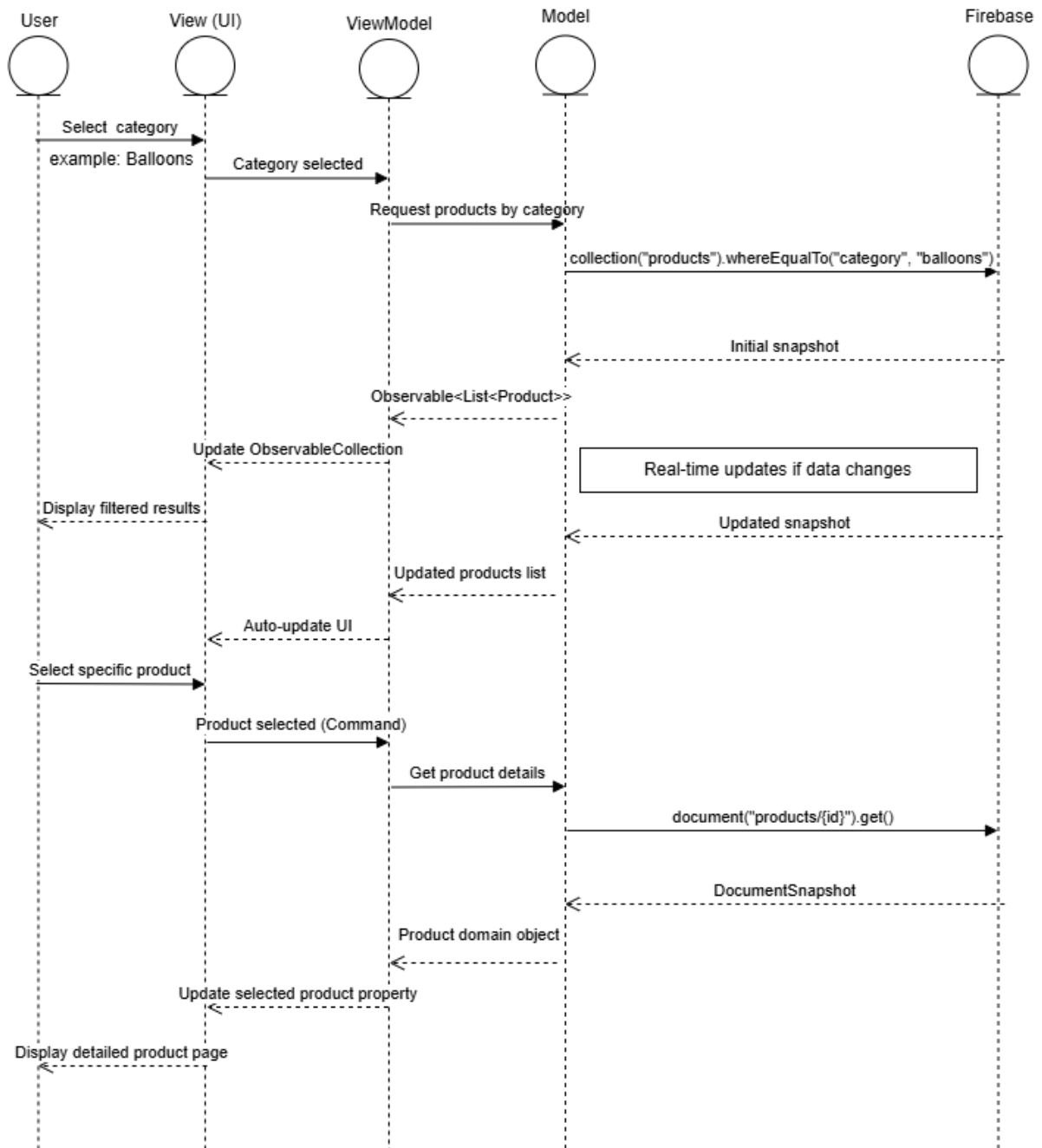
חלק שלישי – Diagrams

דיאגרמת מחלקות – Class Diagrams :דיאגרמת אובייקט – Object Diagram :

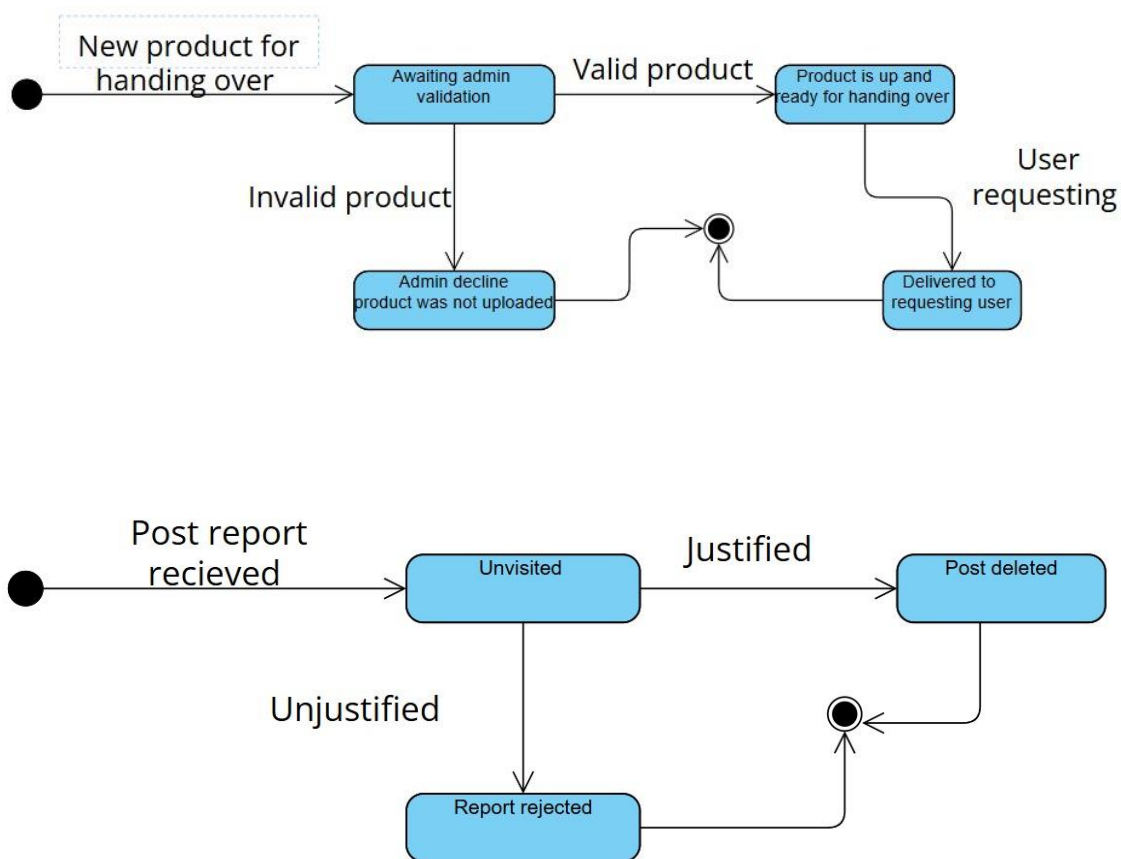
Activity Diagram - Registration Process



Sequence Diagram - Product Search



: State Machine Diagram



: ERD

