



שם התלמיד: עמית רון

שם הפרויקט: staem

שנה: 2020-2021

ת"ז: 213211972

שם המורה: יוסי ברון

תיכון: בליך

## תוכן עניינים

3.....	Staem
4.....	תרשים זרימה
5.....	מפת אפליקציה
6.....	ניתוח מערכות
7.....	Wigets
7.....	Layouts
7.....	טכנולוגיות מתקדמות
8-9.....	ספריות חיצוניות
10.....	פירוט טכנולוגיות
10.....	תכנות מונחה אירועים ומחלקות פנימיות
10.....	רכיבים מתקדמים
11.....	חריגות
12.....	שימוש בענן firebase
13.....	Firestore database
13.....	Storage
13.....	Authentication
14.....	רישום מחלקות
15.....	אובייקטים
15.....	Services
16.....	דפי java הקשורים ל layout
17.....	שימוש בהורשה
17.....	שימוש בפולימורפיזם
18-22.....	מסכי דוגמא
23-25.....	ביבליוגרפיה

מותג המשחקים STAEM היא חנות משחקים דינאמית וחדשנית בעל מגוון רחב של משחקים מאתגרים לכל סוגי השחקנים.

בימנו עותק קשיח של המשחקים אנו קונים עולה הרבה יותר מעותק דיגיטלי. בין אם מדובר על משחקי אקשן ויריות, משחקי חשיבה ואסטרטגיה, או אפילו MMORPG. ומכיוון שעותק קשיח של משחקים עולה יותר וגם בזמן האחרון מתחילה לרדת התמיכה לדיסקים במחשבים. שחקנים רבים אוהבים לקנות את המשחקים שלהם און ליין, כך שהפתרון עבורם הינו ביצוע הרכישות באמצעות המחשב או אפילו הטלפון החכם מכל מקום שבו הם נמצאים.

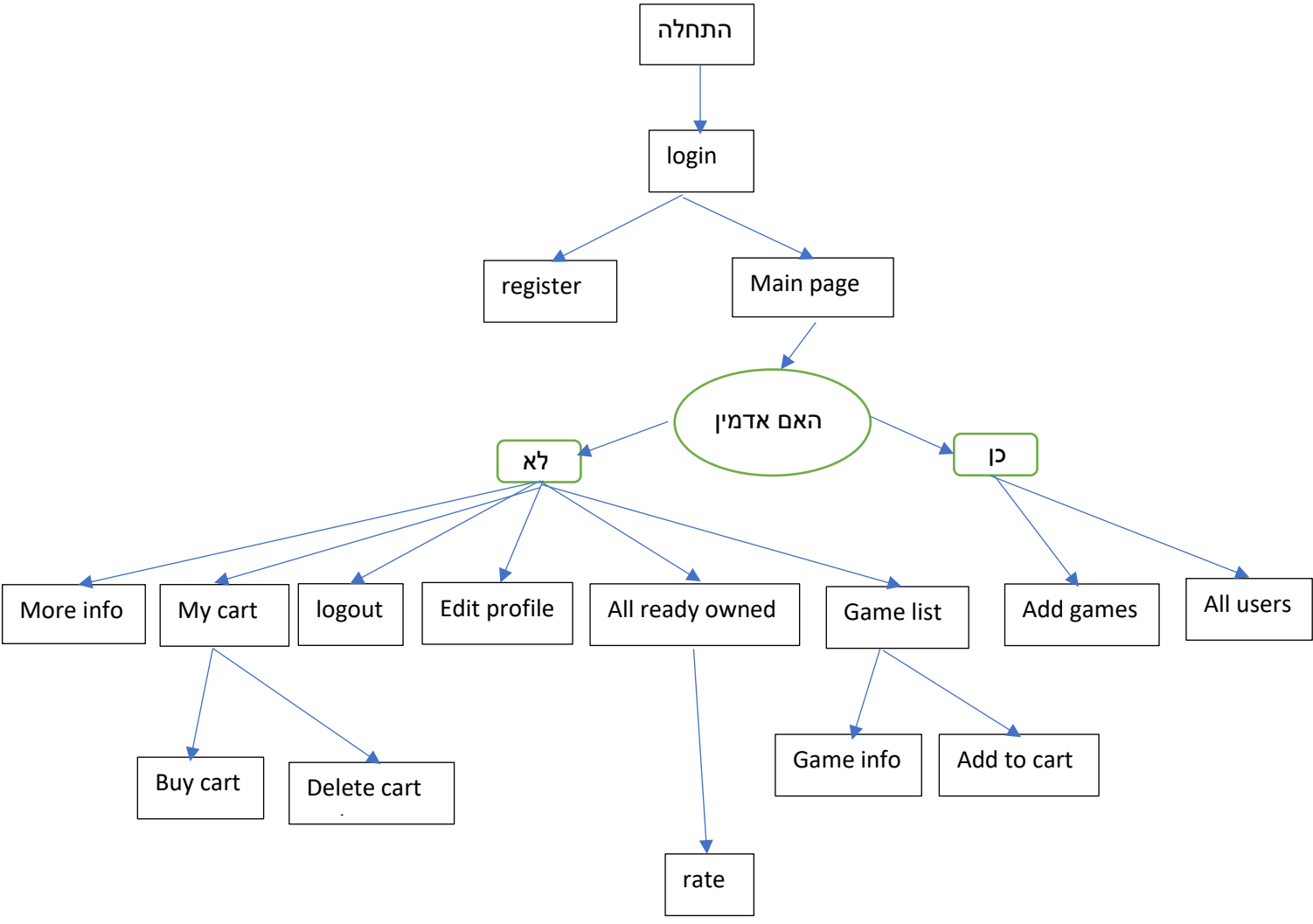
מי שכבר התנסה משחקים באינטרנט יודע, שהרי זהו דבר מופלא שמאוד חבל שלא חשבו עליו לפני שנים רבות מאוד. מי שטרם התנסה בכך, יתכן שעוד לא מבין עד כמה פשוט וקל זה לבצע קניית משחקים באינטרנט. למעשה, מהרגע שמנסים זאת, כבר ניתן להבחין ביתרונות הרבים של רכישות מקוונות.

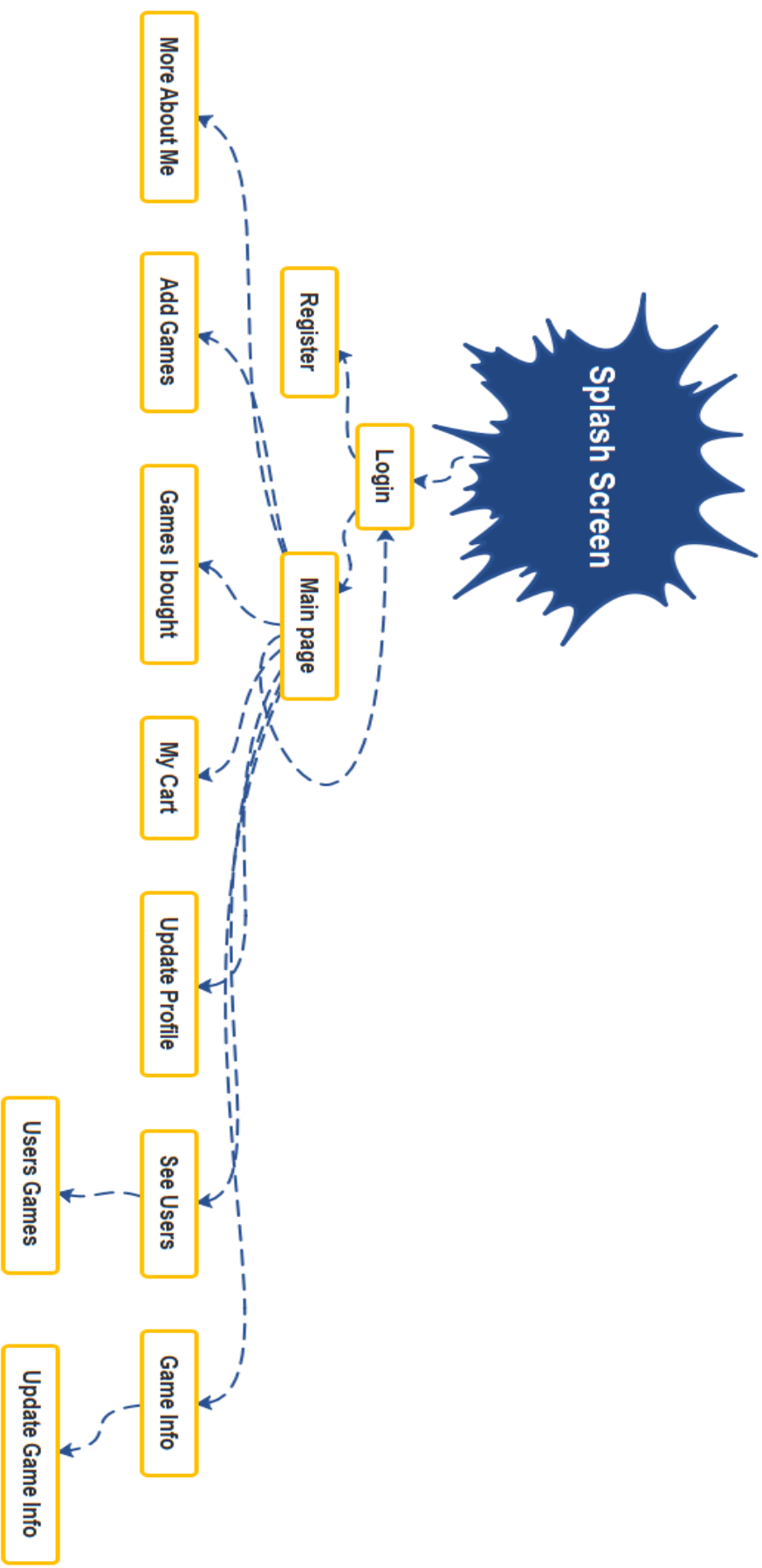
STAEM הינה אפליקציה אשר משמשת לקניית משחקים דרך הטלפון החכם בקלות ויעילות, דבר שמקל מאוד על חווית הקנייה.

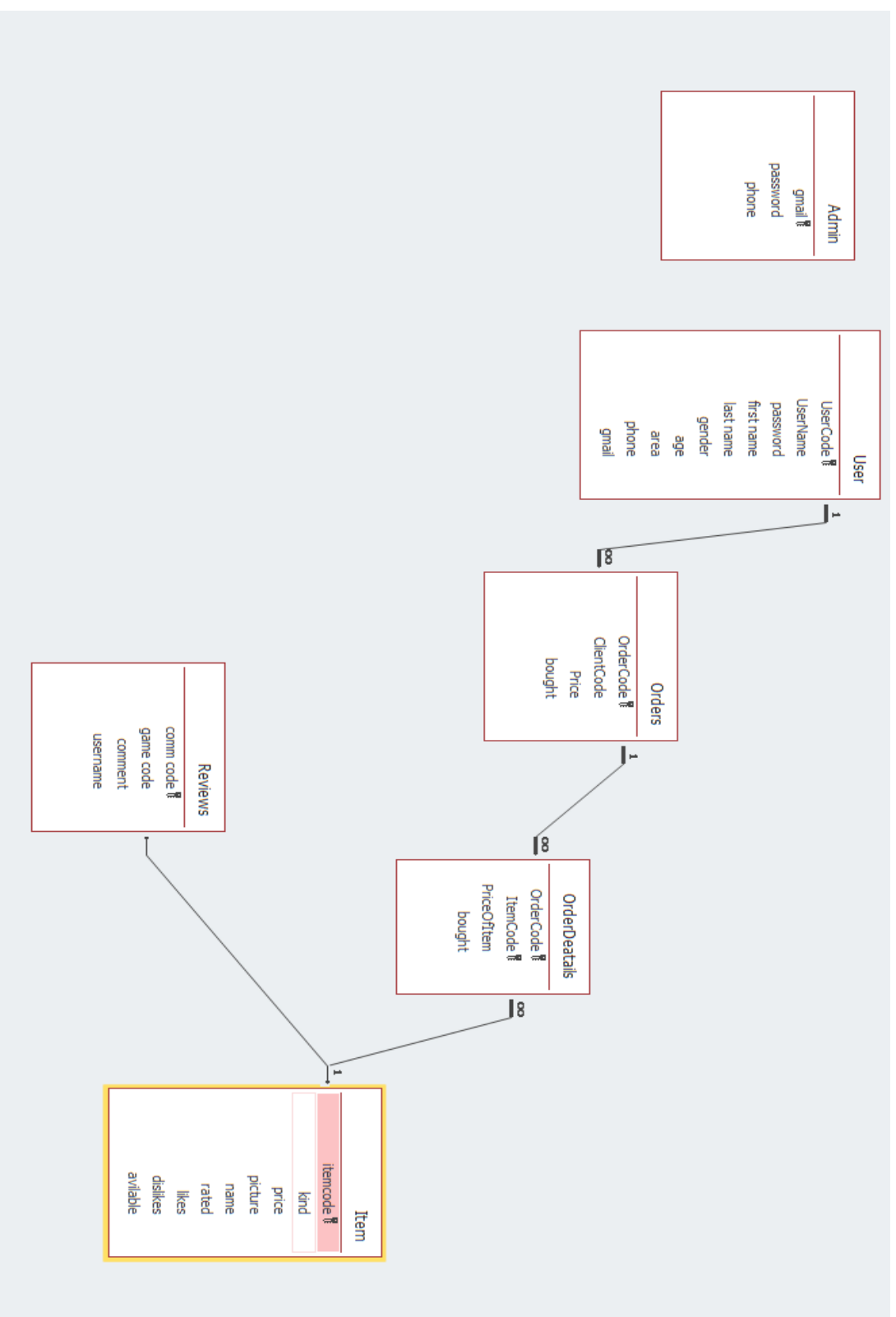
האפליקציה מיועדת לבני נוער ולהורים אשר רוצים לרכוש לילדיהם וכוללת מגוון רחב של סוגי משחקים.

בשונה מאפליקציות אחרות העיצוב של אפליקציית STAEM הוא חדשני ועדכני. עיצוב האפליקציה החדשני מושך את עיניהן של קהל בני הנוער וגורם לאחוזי קניות גבוהים יותר. דבר נוסף המאפיין את STAEM הוא שהחנות מכילה פריטים ממגוון חברות שונות, וכך מכסה מגוון של קהל רחב יותר, קהל האוהב משחקים שונים, וכך מתאפשר לו לקנות את כולם באותו מקום.

האפליקציה מציגה שירות של לקוח ומנהל הכולל מגוון רחב של משחקים, טופס הרשמה והתחברות, אפשרות לעדכן פרטי לקוח, חיפוש משחקים, בשונה מחנות רגילה ל STAEM אין מלאי של פריטים מכיוון שהפריטים שנמכרים הם קודים להפעלת המשחק, קיימת עגלת קניות.







## **Widgets:**

TextView

ImageView

GridView

RadioButton

DatePicker

CheckBox

CardView

ListView

SearchView

EditText

## **Layouts:**

ConstrainLayout

LinearLayout

RelativeLayout

## **טכנולוגיות מתקדמות:**

Intent

CountDownTimer

Menu

Dialog

Toast

MediaPlayer

## ספריות חיצוניות

**Firestore** – הענן בוא השתמשתי במהלך הפרויקט

```
db = FirebaseDatabase.getInstance();
auth = FirebaseAuth.getInstance();
ref = db.getReference("Games");
ref2 = db.getReference("Cart");
storageReference = FirebaseStorage.getInstance().getReference();
```

**Picasso** - מציג את התמונות המופיעות באפליקציה

```
String imageUrl="https://drive.google.com/uc?export=download&id="+p[5];
Picasso.get().load(imageLink).into(img);
```

## BroadcastReceiver

מערכת האנדרואיד פולטת כל מיני הודעות לדוגמא מתנתק מה- wifi, על מנת שהאפליקציות תוכלו לדעת דבר כזה קיים ה Receiver Broadcast - אשר מאזין להודעות מסוג זה הנשלחות ממערכת האנדרואיד לכלל האפליקציות הפעילות.

דוגמא:

```
private BroadcastReceiver wifiStateReceiver = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        int wifiStaeExtra = intent.getIntExtra(WifiManager.EXTRA_WIFI_STATE,
        WifiManager.WIFI_STATE_UNKNOWN);

        switch (wifiStaeExtra){
            case WifiManager.WIFI_STATE_ENABLED:
                break;

            case WifiManager.WIFI_STATE_DISABLED:
                AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(
                    context);
                dialog alert = new dialog();
                alert.showDialog(MainActivity.this, "Error wifi is turned off pleas
                turn it back on or you will not be able to buy from the app");

                // create alert dialog
                AlertDialog alertDialog = alertDialogBuilder.create();

                // show it
                alertDialog.show();
                break;
        }
    }
};
```



## אנימציה בדף layout – AnimationDrawable

המחלקה AnimationUtils מאפשרת ליצור אנימציה המדמה תנועה של ווידג'טים שונים במסך, בעזרת תכנון מסלול התנועה מראש וקריאה למחלקה דרך thread נוסף ניתן ליצור תזוזות שונות של חלקים שונים במסך כתגובה לפעולות שונות, למשל פתיחת האפליקציה, בחירת אופציה מסוימת או סיום ביצוע פעולה כלשהיא ברקע (ניתן להגדיר מראש מה יהיו תזוזות החלקים השונים).

דוגמא לשימוש באפליקציה:

```
Animation animation = AnimationUtils.loadAnimation(this, R.anim.blink_anim);
view.startAnimation(animation);
```

## אוספים:

### List<> , ArrayList<>

דוגמא:

```
ListView listViewProduct;
List<Cart> cartList;
List<Games> gamesList;
```

### Dialog

דוגמא:

```
DatePickerDialog dialog = new DatePickerDialog(Update.this,
    android.R.style.Theme_Holo_Light_Dialog_MinWidth, mDataSetListener, year, month,
    day);
dialog.getWindow().setBackgroundDrawable(new ColorDrawable(Color.TRANSPARENT));
dialog.show();
```

### MediaPlayer

שימוש ב mediaplayer כאשר המשתמש פותח את האפליקציה נשמע רינגטון.

```
MediaPlayer mediaPlayer = MediaPlayer.create(SplS.this, R.raw.bass);
mediaPlayer.start();
```

## Intents

לדוגמא:

```
IntentFilter intentFilter = new IntentFilter(wifiManager.WIFI_STATE_CHANGED_ACTION);  
registerReceiver(wifiStateReceiver, intentFilter);
```

הפעלה של ברודקאסטרסיבר

```
startActivity(new Intent(AddGame.this, MainActivity.class));
```

מעבר בין עמודים

## שימוש במשאבי האפליקציה :

משאבי האפליקציה הם צרכים חיצוניים שנצטרך ביישום, כגון תמונות, מחרוזות, צבעים, סגנונות ועוד. כאמור משאבים אלו אמורים להימצא בתיקה המתאימה תחת תיקיית המשאבים בפרויקט "res". שיטה זו מאפשרת להתאים את היישום המפותח לפי סוג המכשיר, שפה ומצב מסך המכשיר.

דוגמא לשימוש באפליקציה:

הגדרת אובייקט לפי קוד זיהוי מהמחלקה id.R

```
edpass = findViewById(R.id.login_input_password);  
edemail = findViewById(R.id.login_input_email);
```

הגדרת תצוגה לפי תמונה מהמחלקה drawable

```
android:src="@drawable/profpic"
```

## תכנות מונחה אירועים ושימוש במחלקה הפנימית:

כל אינטראקציה המתבצעת עם המסך והאפליקציה נקלטת דרך "מאזינים" (המהווים חלק מה View) מערכת האנדרואיד מדווחת ליישומים ולרכיבים הגרפיים על אירועים שמתרחשים והקשורים לאותו רכיב ומגיבה בהתאם.

דוגמא לשימוש באפליקציה:

```
btnupdate.setOnClickListener(new View.OnClickListener() {  
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
```

## רכיבים מתקדמים תפריט, תיבת דו שיח, הודעות:

ישנם תפריטים, תיבות דו שיח והודעות טואסט ( Toasts ) שמוצגות על ידי האפליקציה בסיום, התרחשות ולפני ביצוע פעולות בכדי לדווח למשתמש מידע חשוב על התהליך.

הצגת הודעה קצרה מסוג Toast

```
public void onCancelled(@NonNull DatabaseError databaseError)
{
    Toast.makeText(Login.this, "oops", Toast.LENGTH_SHORT).show();
}
```

## חריגות:

חריגות הן חלק בלתי נפרד מעולם התכנה המתחלקות לשלושה סוגי שגיאות – קומפילציה, לוגיות, וזמן ריצה. הטיפול בחריגות הוא מניעת שגיאות שגורמות לקריסת היישום.

דוגמא:

```
public void onFailure(@NonNull Exception e)
{
    Toast.makeText(Login.this, e.getMessage(), Toast.LENGTH_LONG).show();
}
```

```
public void onCancelled(@NonNull DatabaseError error) {
```

## שימוש בענן (Firebase)

### רישום ענן:

לאורך כל האפליקציה ישנו שימוש בענן חיצוני – Firebase , בין אם זה שימוש של לקוח, מנהל ועוד.

### User

טבלה זו מאפשרת למשתמש להירשם לאפליקציה להתחבר לשנות את הנתונים שלו להוסיף מוצרים לעגלה לרכוש אותם ולדרג אותם .

### Games

טבלה זו מכילה את כל השמחקים המופיעים באפליקציה ומאפשרת את הצגתם, בעזרת התכונה של אדמין אשר מאפשרת למשתמש ליצור עוד משחקים.

### Cart

טבלה זו מכילה את כל רשימות הקניות של המשתמשים, כאשר משתמש רוכש את העגלה שלו נוצרת לו אחת חדשה והמוצרים שרכש נכנסים לטבלה חדשה .

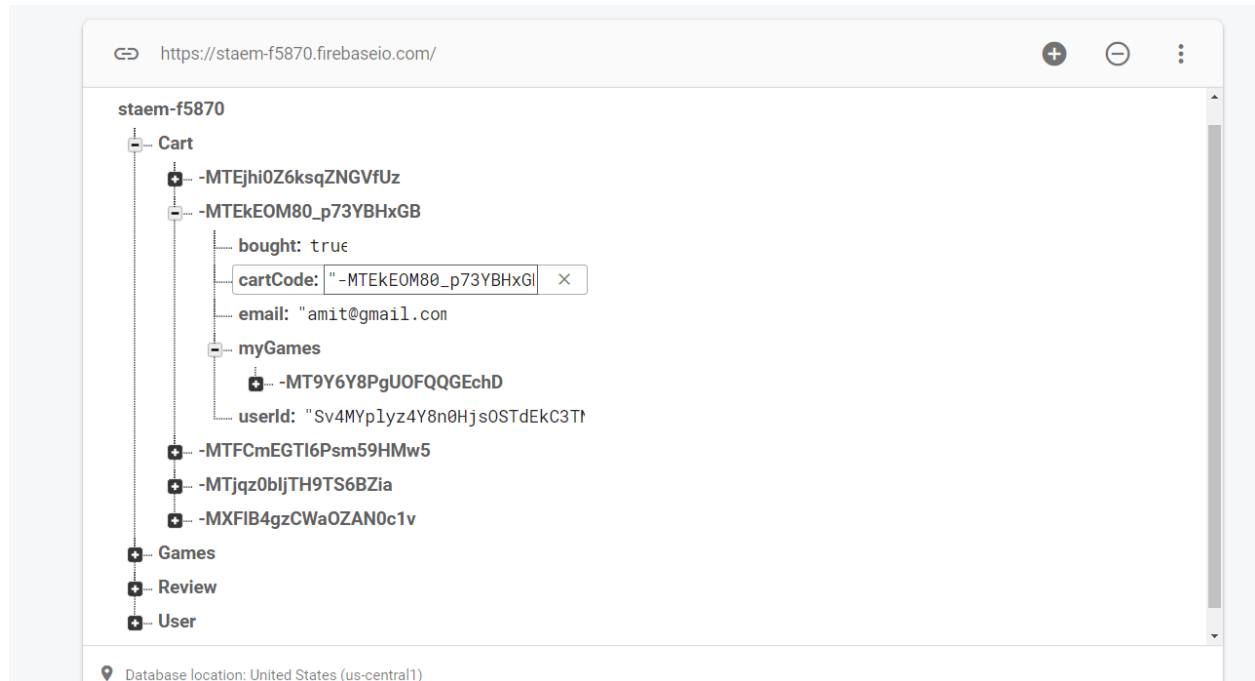
### Owned games / my games

הטבלה הזו הינה טבלה בתוך טבלה היא נוצרת תחת השם "mygames" בתוך "Cart" ובכך נשמרת וזזה כאשר המשתמש קונה את העגלה הנוכחית שלו.

שם הענן – Staem

## תמונות הטבלאות:

טבלאות המכילות נתונים בסיסיים ומידע ב real time



תיקיות המכילות קבצים (תמונות)

The screenshot shows the Google Cloud Storage console for the bucket 'gs://staem-f5870.appspot.com'. It displays a list of files with the following columns: Name, Size, Type, and Last modified.

Name	Size	Type	Last modified
Lol.jpeg	7.7 KB	image/jpeg	Feb 7, 2021
overwatch.jpeg	5.12 KB	image/jpeg	Feb 7, 2021
pubg.jpeg	5.25 KB	image/jpeg	Feb 7, 2021
warzone.jpeg	8.82 KB	image/jpeg	Feb 7, 2021

Authentication מכיל את המידע הדרוש להתחברות בתור משתמש

The screenshot shows the Firebase Authentication console. It includes a search bar with the text 'Search by email address, phone number, or user UID', an 'Add user' button, and a list of users.

Identifier	Providers	Created	Signed In	User UID ↑
amit@gmail.com		Jan 5, 2021	Feb 11, 2021	Sv4MYp1yz4Y8n0Hjs0STdEkC3TN2
amitrongg@gmail.com		Jan 5, 2021	Apr 27, 2021	Z7nNdyj47PSaS3cm9XVqQsJUSK...

Rows per page: 50 1 - 2 of 2

## רישום מחלקות

- © AddGame
- © AdminPage
- © BoughtCarts
- © BoughtGamesList
- © Cart
- © CartList
- © dialog
- © GameInfo
- © Games
- © GamesList
- © Global
- © Login
- © MainActivity
- © MyCart
- © MyData
- © Permission
- © Reg
- © Review
- © SplS
- © Update
- © UpdateGame
- © User
- © UserList
- © UsersCarts

## אובייקטים – (תיקיית models)

### **Public class User**

מחלקה המייצגת טבלה בענן של משתמש, הכוללת שדות כמו סיסמא, ופרטים אישיים.

### **Public class Games**

מחלקה המייצגת טבלה בענן של אובייקטים מסוג משחק מכילה פרטים על המשחק.

### **Public class GamesList**

מחלקה המכילה רשימה של האובייקטים מסוג משחק.

### **Public class Cart**

מחלקה המייצגת טבלה בענן של אובייקטים מסוג עגלות קניות, הכוללות פרטים המקשרים בין המשתמש למשחקים אותם הוא מעוניין לרכוש.

### **Public class CartList**

מחלקה המכילה רשימה של האובייקטים מסוג עגלות.

### **Public class UserList**

מחלקה המכילה רשימה של האובייקטים מסוג משתמשים.

## סרוויסים (תיקיית services)

### **public class AuthService**

מחלקה המשמשת כמאגר פעולות הקשורות לאוטנטיקציה של משתמשים עם המידע בענן.

### **public class FirebaseStorageService**

מחלקה המשמשת כמאגר פעולות הקשורות למאגר המידע "Storage" שנמצא בענן.

### **public class FireBaseDatabaseService**

מחלקה המשמשת כמאגר פעולות הקשורות למאגר המידע ב real time שנמצא בענן.

## דפי Java הקשורים לדפי Layout

AddGame → activity\_add\_game

AdminPage → activity\_admin\_page

BoughtCarts → activity\_bought\_carts

Dialog → activity\_dialog

GameInfo → activity\_game\_info

Login → activity\_login

MainActivity → activity\_main

MyCart → activity\_my\_cart

Reg → activity\_reg

Splq → activity\_splq

Update → activity\_update

UpdateGame → activity\_update\_game



## דוגמאות לשימוש בהורשה באפליקציה

```
public class MyCart extends AppCompatActivity
```

```
public class GameInfo extends AppCompatActivity
```

```
public class UserList extends ArrayAdapter<User>
```

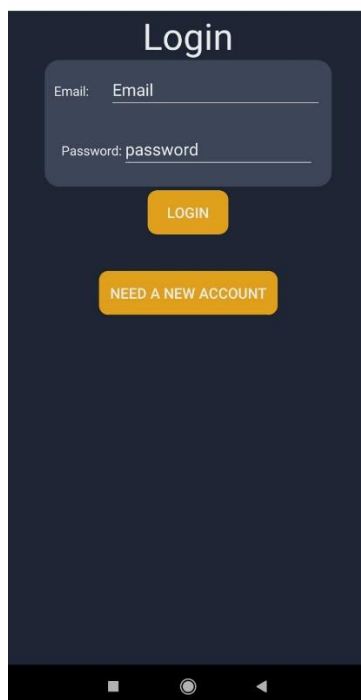
## דוגמאות לשימוש בפולימורפיזם באפליקציה

```
txtname = findViewById(R.id.txtname);  
likes = findViewById(R.id.likes);  
dislikes = findViewById(R.id.dislikes);  
txtprice = findViewById(R.id.txtprice);  
btnupdate= findViewById(R.id.btnupdate);  
btnadd= findViewById(R.id.btnadd)
```

מסכי דוגמא מהאפליקציה:

(עיצוב האפליקציה מותאם לגרסה האנגלית של מכשירי אנדרואיד ועל כן עלול להשתנות בין הגרסאות השונות או שפות שונות)

מסך פתיחה



כניסה

הרשמה

## Registration

Email:

Password:


UserName:

phone:

First Name:

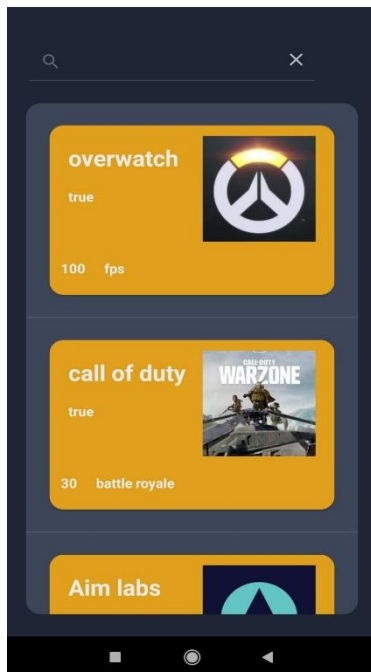
Last Name:

Date  
☐ Male ☐ Female



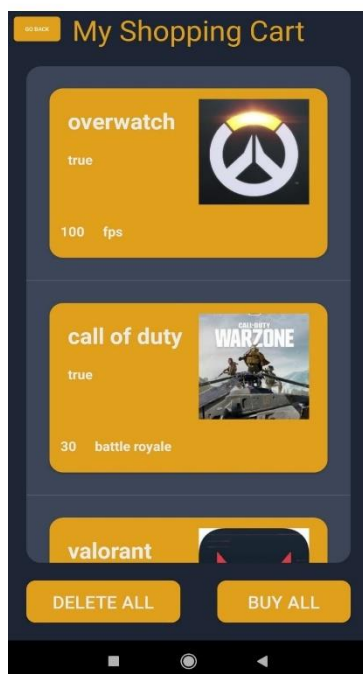
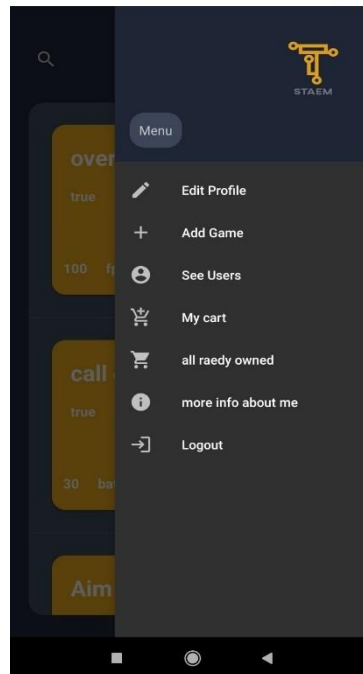
[REGISTER ME](#)

[ALL READY HAVE AN ACCOUNT](#)



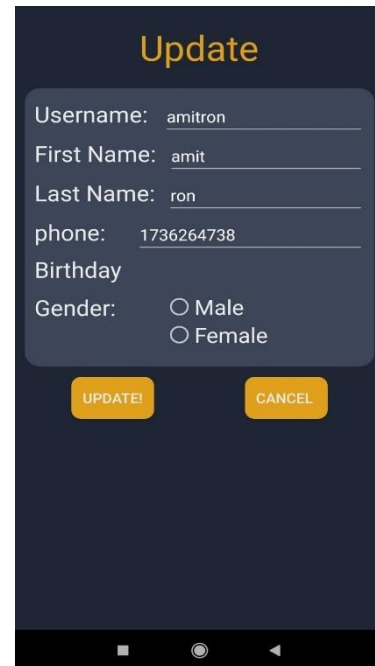
מסך ראשי

תפריט



עגלת קניות

עדכון נתונים



Update

Username:

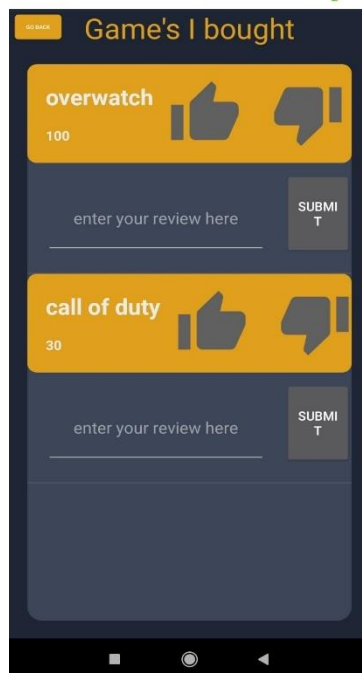
First Name:

Last Name:

phone:

Birthday:

Gender: ☐ Male ☐ Female



Game's I bought

overwatch 100

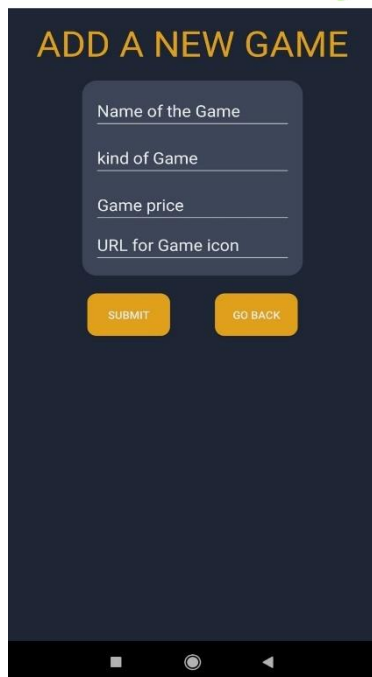
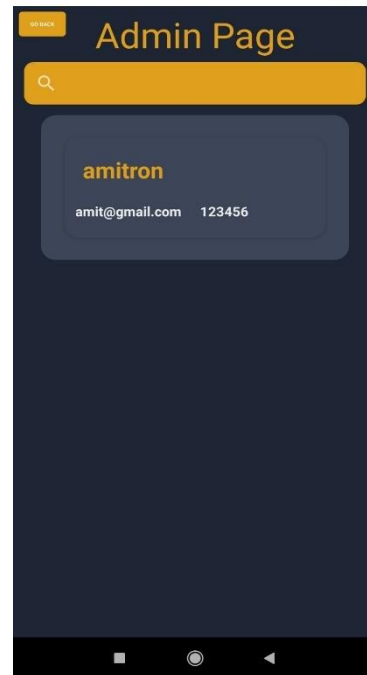
enter your review here

call of duty 30

enter your review here

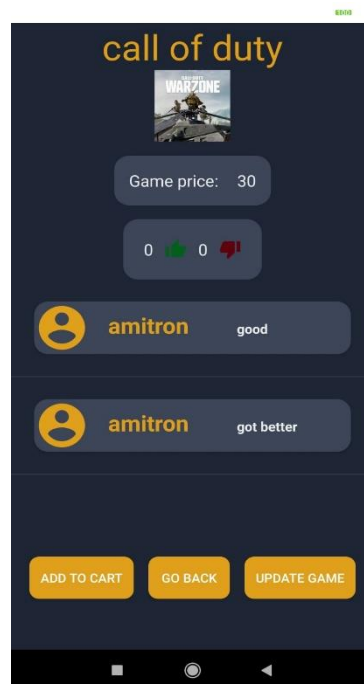
משחקים שנקנו + דירוג שלהם

עמוד אדמין



הוספת משחק

דף מידע על משחק



## מדריך משתמש

כניסה כ- User

prep4testuser@gmail.com-Email

123456-Password

כניסה כ- Admin

prep4testadmin@gmail.com-Email

123456-Password



## ביבליוגרפיה

- <https://gist.github.com/srix55/ec64d2f6a371c80bbbc4>
- <https://medium.com/@zackcosborn/step-by-step-recyclerview-swipe-to-delete-and-undo-7bbae1fce27e>
- <https://androidwave.com/double-tap-listener-android-example/>
- <https://stackoverflow.com/questions/33985719/android-swipe-to-delete-recyclerview>
- <https://stackoverflow.com/questions/25148291/how-to-integrate-google-drive-with-picasso-on-android>
- <https://developer.android.com/reference/android/content/Intent>
- <https://www.youtube.com/channel/UCgASUpf4Hk4gC2Fy3Rv1z-A>