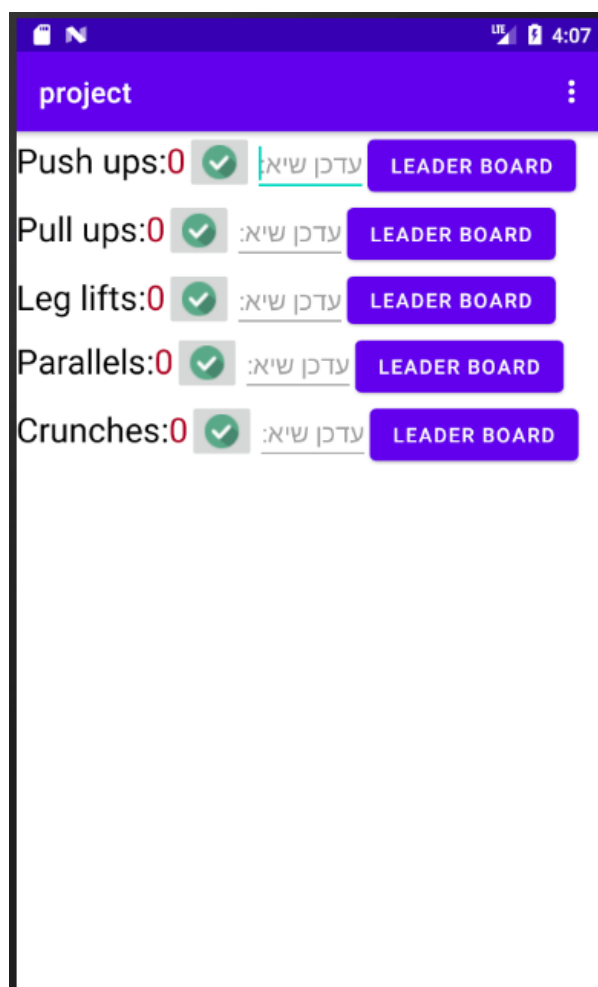


ספר פרויקט גמר י"ב - תכנון ותכנות טלפונים חכמים – הנדסת תוכנה
אפליקציית שיאי אימונים אישיים



בית יצירה קציר
קמפוס חינوبي - משגב



שם המגיש: אבישי עוז
תהודת זהות: 326489069
שנת לימודים: 2021-2022
בית ספר: קציר משגב
מספר סמל המוסד: 640565
שם המנחה: הדס יגאל BSc הנדסת תוכנה

תוכן עניינים:

3	מבוא:
4	מדריך למשתמש
5	תיאור מסכי האפליקציה:
5	מסך ההרשמה/ההתחברות:
6	מסך השיאים האישי:
7	מסך טבלת בשיאים:
8	דיאגרמת מחלקות UML:
9	תרשים זרימה:
10	עמידת בדרישות הפרויקט:
11	הסבר על קטעי קוד:
11	Service:
13	מחלקות ואובייקטים:
14	broadcast receiver :
15	Adapter:
18	Firestore:
19	דיאלוג:
20	תפריט:
21	קובץ המניפסט:
22	סיכום:

מבוא:

מטרתה של האפליקציה שהכנתי היא לעודד את המשתמשים להתאמן במגוון תרגילי כושר שונים ובכל אחד מהם לנסות להיות כמה שיותר טובים

האפליקציה עושה זאת באמצעות טבלה שמדרגת את כל משתמשי האפליקציה כל משתמש לפי השיא שלו בכל אחד מהתרגילים ובכך מעודדת אותם לעלות לראש הטבלה ולשפר את שיאם

כל אדם חדש שמעוניין להיכנס לאפליקציה ירשם וכל השיאים שלו יתחילו ב-0 בנוסף הוא יוכל למצוא הסבר לביצוע כל אחד מהתרגילים, בכל פעם שהמשתמש ישבור שיא הוא יוכל לחזור למשתמש שלו שבו שמורים השיאים הקודמים ולעדכן לשיא החדש

פעולות נוספות של האפליקציה :

- התראה למקום הראשון כאשר מישו עקף אותו
- התראה כאשר אין אינטרנט

בספר זה אציג את המסמכים, המחלקות, מבנה נתונים של האפליקציה וגם אסביר כיצד להשתמש בה

סביבת עבודה: studio android

גרסת האנדרואיד סטודיו: 4.1

שפת תכנות: java

מדריך למשתמש

- התקן את האפליקציה באמצעות סריקת הבר קוד הבא:

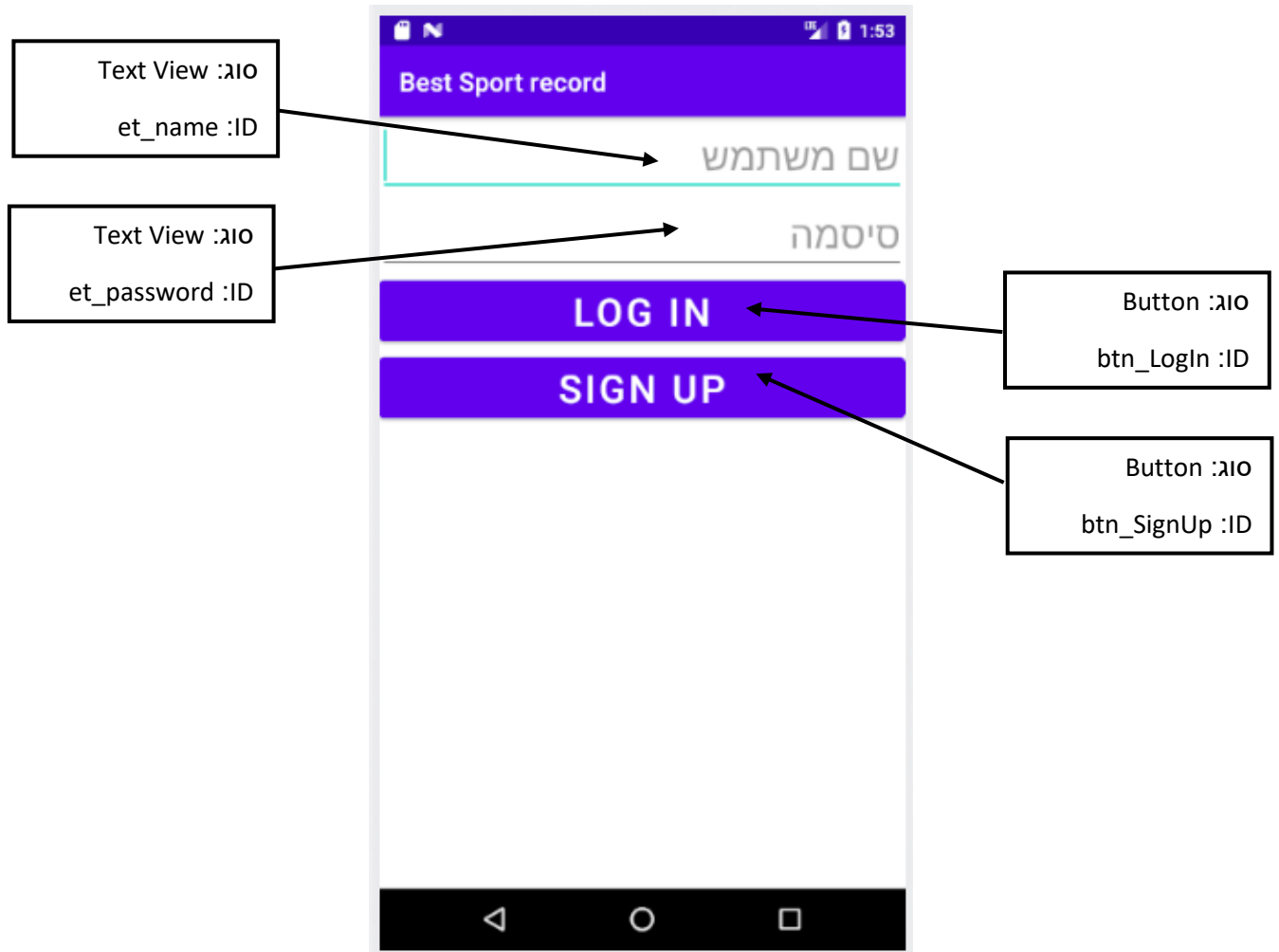


- התחבר \ הירשם באמצעות שם משתמש וסיסמה
- עדכן את השיאים האישיים שלך על ידי הכנסת השיא לתיבת הטקסט ולחיצה את ה "וי" שלידה.
- במידה ואינך יודע לבצע אחד מהתרגילים הבאים תוכל להיעזר בתפריט ההסבר בפינת המסך.
- אחרי שעדכנת את השיאים שלך אתה יכול להיכנס לטבלת השיאים ולראות איכן אתה ממוקם ביחס לחברייך.
- אם אתה מקום ראשון בתרגיל מסוים תקבל התראה שמישהו יעקוף אותך.

תיאור מסכי האפליקציה:

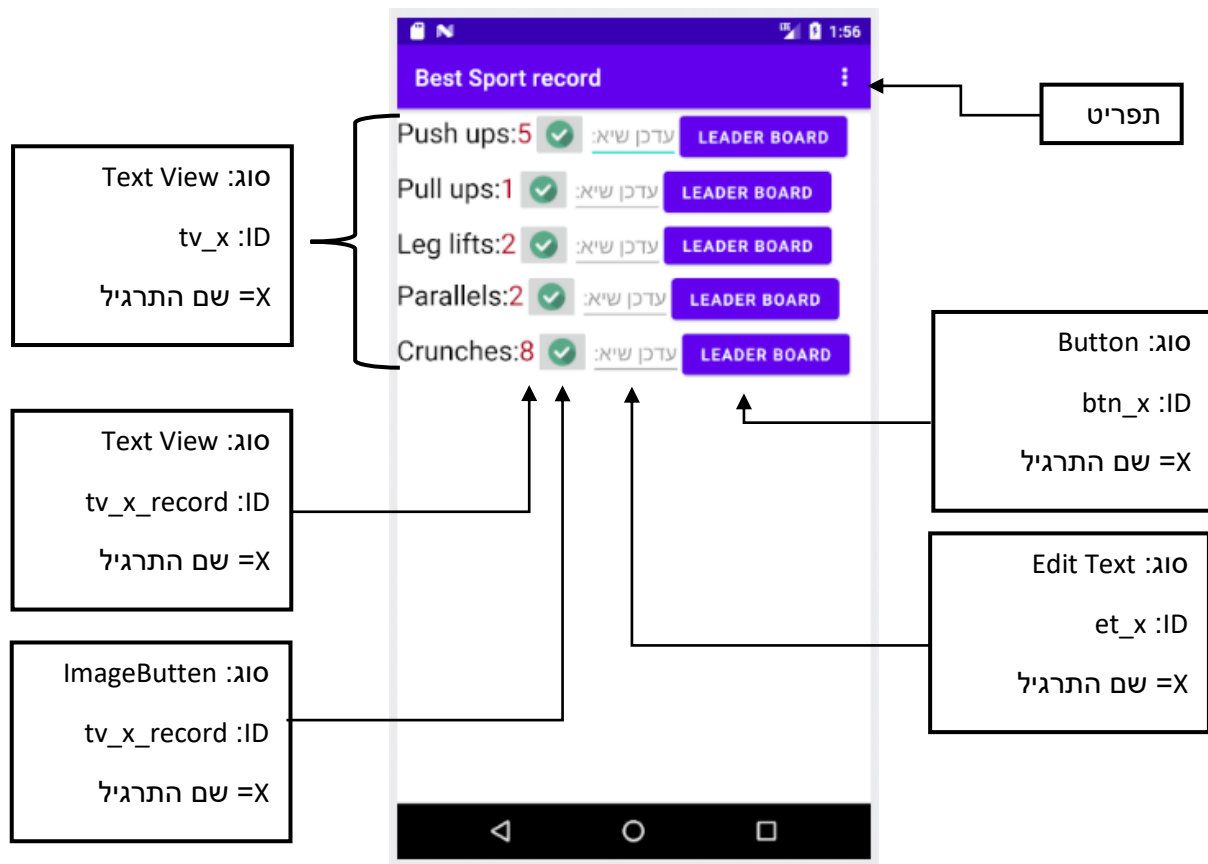
מסך ההרשמה/ההתחברות:

מסך זה הוא המסך הראשון שהמשתמש רואה ובוא הוא יכול להירשם או להתחבר אם כבר יצר משתמש בעבר.



מסך השיאים האישי:

במסך זה כל משתמש יכול לעדכן את השיאים שלו בכל אחד מהתרגילים, בכל פעם שהמשתמש יחזור למסך יופיע השיא שלו



מסך טבלת בשיאים:

במסך זה כל משתמש יכול לראות איפה הוא ממוקם ביחס לשאר משתמשי האפליקציה עבור כל תרגיל בנפרד.

Best Sport record		
Pull Ups records table:		
#1	b	2
#2	a	1
#3	c	0

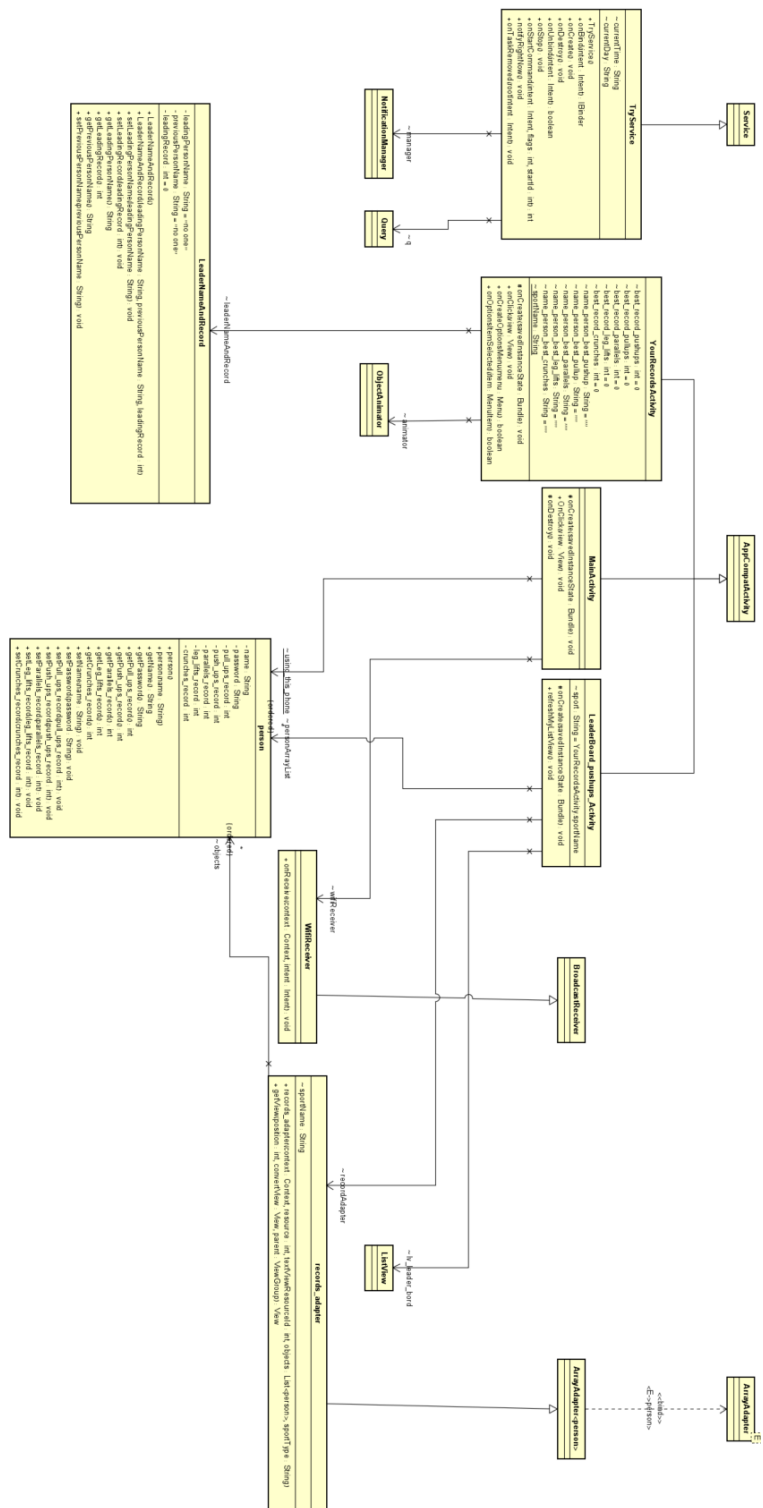
Best Sport record		
Push Ups records table:		
#1	a	5
#2	b	2
#3	c	0

Best Sport record		
Leg Lifts records table:		
#1	a	2
#2	b	2
#3	c	2

Best Sport record		
Crunches records table:		
#1	a	8
#2	b	5
#3	c	3

Best Sport record		
Parallels records table:		
#1	a	2
#2	b	2
#3	c	0

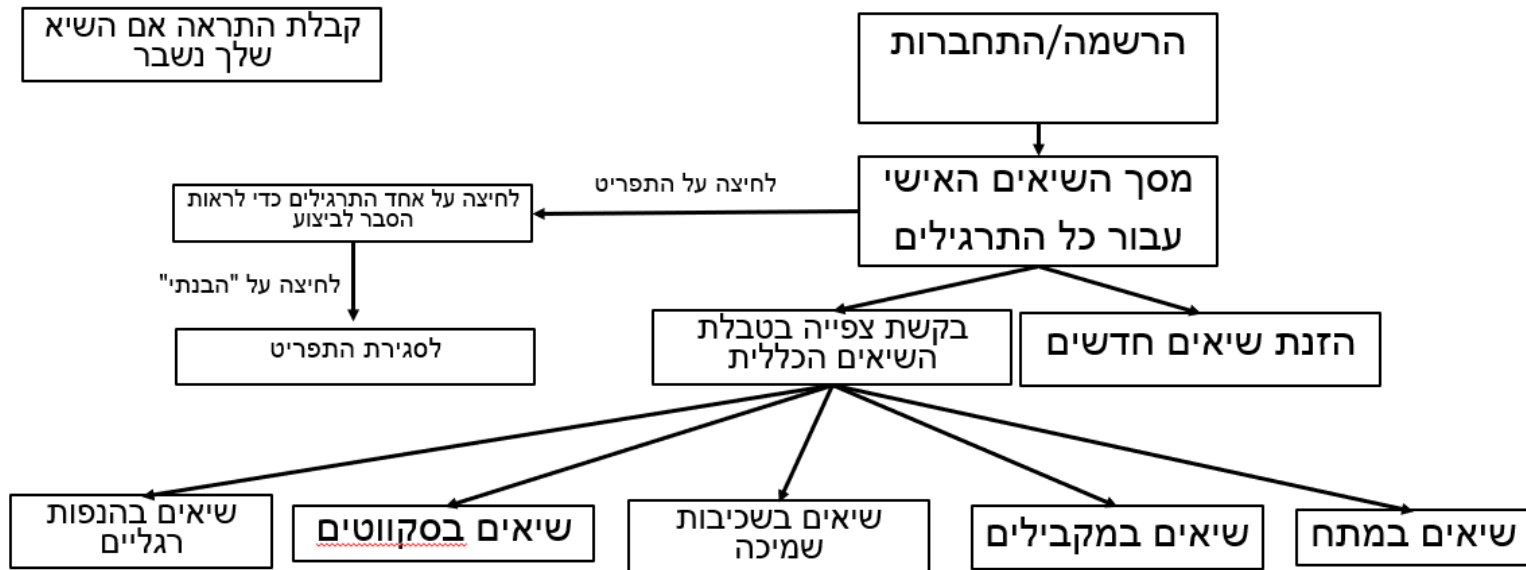
דיאגרמת מחלקות UML:



קישור לתמונה המלאה :

<https://drive.google.com/file/d/1EyghurgvgmdGwUln6h7GGUcM0d4lxl6z/view?usp=sharing>

תרשים זרימה:



עמידת בדרישות הפרויקט:

הדרישה	פירוט מענה על הדרישה בפרויקט
לפחות 5 מסכים שונים	המסכים שלי הם: 1. Login/sign_up 2. Your record 3. Leader board x5 for each exercise
לפחות 5 מחלקות	המחלקות שלי הן: 1. Person 2. MainActivity 3. records_adapter 4. YourRecordsActivity 5. LeaderBoard_pushups_Activity
שימוש בתפריטים	התפריט שלי נמצא במסך השיאים האישי ובתוכו ניתן למצא הסבר לביצוע כל אחד מהתרגילים
שימוש בתיבות דו שיח	בתוך התפריט כאשר תפתח הסבר על אחד התרגילים תאלץ ללחוץ "הבנתי" כדי לסגור את ההסבר
Service	הסרביס אצלי הוא: TryService אם היית ראשון בתרגיל מסויים ועקפו אותך תקבל על כך הודעה
intent	במעבר בין חלונות ושימוש בדיאלוגים יש שימוש ב-intent
רכיבים מתקדמים באפליקציה - אנימציה	במסך השיאים האישי כאשר תעדכן שיא תופיעה אנימציה
Broadcast reciver	בדיקה האם קיים חיבור לאינטרנט או לא

הסבר על קטעי קוד:

:Service

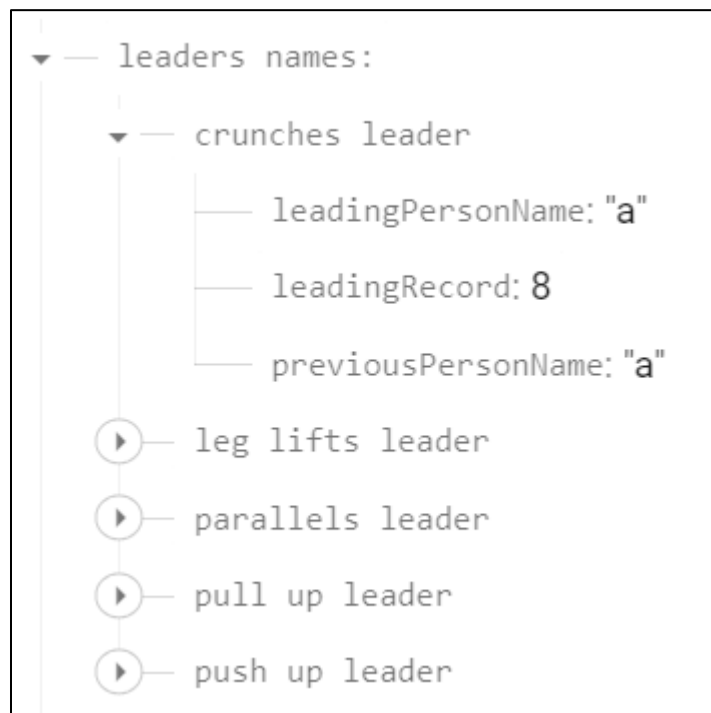
תפקיד ה-Service באנדרואיד הוא לבצע פעולות ברגע שהמפתח החליט שפעולה צריכה להתבצע ואין לו ממשק משתמש כלומר אין לחצנים או אפשרות אחרת למשתמש לגרום ל-service להתחיל לפעול

בפרויקט שלי השתמשתי ב-service לשלוח התראה למשתמש אשר היה במקום הראשון באחד מענפי הספורט באפליקציה ונעקף.

במסך הבית של האפליקציה אני מפעיל את ה-service:

```
my_notification_intent = new Intent( packageContext: MainActivity.this, TryService.class);
startService(my_notification_intent);
```

בפיירבייס אני שומר עבור על ענף ספורט את שם השיאן ואת השיא שלו ובנוסף את מי שהיה השיאן לפניו.



כאשר מופעלת פעולת onStartcommand שעוברת על כל שמות של ה-"leaders names" בפיירבייס ובודקת האם המשתמש שמחזיק כרגע את הטלפון הוא היה השיאן הקודם

```
public int onStartCommand(Intent intent, int flags, int startId) {
    q = myRef.child("leaders names:").orderByKey();
    q.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
            for (DataSnapshot dst : dataSnapshot.getChildren()) {
                String sportNameKey = dst.getKey();
                LeaderNameAndRecord leaderNameAndRecord = dst.getValue(LeaderNameAndRecord.class);
                if(MainActivity.using_this_phone!=null && MainActivity.using_this_phone.getName().equals(leaderNameAndRecord.
```

אם כן: אני מפעיל את הפעולה – notifyRightNow()

```
public void notifyRightNow(){
    builder = new NotificationCompat.Builder( context: this)
        .setSmallIcon(R.drawable.common_google_signin_btn_icon_dark)
        .setContentTitle("Best Sport Record")
        .setContentText(currentTime)
        .setDefaults(NotificationCompat.DEFAULT_ALL)
        .setPriority(NotificationCompat.PRIORITY_MAX);

    int NOTIFICATION_ID = 12345;
    Intent notification_intent = new Intent( packageContext: this, MainActivity.class);
    PendingIntent contentIntent = PendingIntent
        .getActivity( context: this, requestCode: 0, notification_intent, PendingIntent.FLAG_UPDATE_CURRENT);
    manager = (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);
    builder.setContentIntent(contentIntent);
    builder.setContentText("עקפו אותך באחת הקטגוריות");
    manager.notify(NOTIFICATION_ID, builder.build());
}
```

ניתן לראות בשורה אחת לפני האחרונה שהטקסט שהמשתמש קיבל בהתראה הוא "עקפו אותך באחת הקטגוריות"

בנוסף אני רוצה למנוע מצב שאותו אדם יקבל שוב ושוב התראות על אותו אירוע לכן שיניתי לו את השם -

```
leaderNameAndRecord.setPreviousPersonName("Person already got notification");
```

אז אם אני רואה שהשם של הבן אדם הקודם הוא "כבר קיבל התראה" אז לא נשלח לו התראה נוספת.

מחלקות ואובייקטים:

- **Person – מחלקה המייצגת משתמש שהתחבר לאפליקציה ושומרת את סיסמתו ושיאיו בכל אחד מהתרגילים הקיימים.**

```
public class person {  
    private String name;  
    private String password;  
    private int pull_ups_record;  
    private int push_ups_record;  
    private int parallels_record;  
    private int leg_lifts_record;  
    private int crunches_record;  
}
```

:broadcast receiver

- המילה ברודקאסט פירושה לשדר לכל "העולם"
- מערכת ההפעלה אנדרואיד משדרת הודעות על אירועי מערכת כמו התחברות למטען, סוללה חדשה וכו..
- באפליקציה שלי ממשתי ברודקאסט שמאזין רק למצב החיבור לאינטרנט ומתריע כאשר לא קיים חיבור או כשהוא חוזר.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.project">
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="project"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Project">
        <receiver
            android:name=".WifiReceiver"
            android:enabled="false"
            android:exported="false"></receiver>

        <activity android:name=".YourRecordsActivity" />
        <activity android:name=".LeaderBoard_pushups_Activity" />
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

:Adapter

משמעות המילה adapter היא מתאם. מטרת המתאם היא להציג ב listView את המידע שנמצא במערך, כלומר לתאם בין מערך שהוא מבנה נתונים רעיוניים לבין ה listView שנמצא על המסך שהמשתמש יכול לצפות בו.

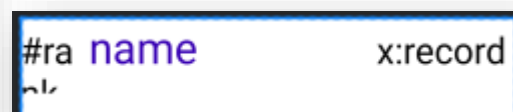
בפרויקט שלי אני מציג את טבלת השיאים בעזרת האדפטר שמציגה לכל משתמש את מיקומו ביחס לאחרים אשר ממויין לפי השיא שלו, את שמו ושיאו.

Best Sport record		
Pull Ups records table:		
#1	b	2
#2	a	1
#3	c	0

האדפטר פועל ב3 שלבים שונים:

שלב 1: ניצור שורה כללית כדי שהאדפטר ידע איך למלא את הטבלה (template)

נגדיר שורה כללית בקובץ xml (t_person_list) המכילה שלושה TextView – מקום יחסי, שם המשתמש, השיא בתרגיל מסויים.



שלב 2:

השימוש באדפטר במחלקה אחרת, נגדיר שני מרכיבים:

1. ListView – הטבלה שהמשתמש יראה בסופו של דבר

2. ArrayList – מערך "אין סופי" שמשמש אותי (המתכנת) לשמור את הנתונים

```
ListView lv_leader_bord ;
ArrayList<person> personArrayList;
```

כל פעם שניצור משתמש חדש נוסיף אותו למערך שלנו:

```
public void onDataChange(@NonNull DataSnapshot snapshot) {
    for (DataSnapshot dst : snapshot.getChildren()){
        person user = dst.getValue(person.class);
        personArrayList.add(user);
    }
}
```

ובנוסף כל פעם שנשנה שיא של מישהו נבדוק האם אכן אוזן שיא שונה ונמייין אותו – בעקבות השינוי צריך לעדכן את הרשימה שלנו ולכן אנו קוראים לפונקציה – refreshMyListView

```
Collections.sort(personArrayList, new Comparator<person>() {
    @Override
    public int compare(person lhs, person rhs) {
        int answer = 0;
        if (sport.equals("Push Ups"))
            answer = (-1) * Integer.valueOf(lhs.getPush_ups_record()).compareTo(rhs.getPush_ups_record());

        else if (sport.equals("Pull Ups"))
            answer = (-1) * Integer.valueOf(lhs.getPull_ups_record()).compareTo(rhs.getPull_ups_record());

        else if (sport.equals("Leg Lifts"))
            answer = (-1) * Integer.valueOf(lhs.getLeg_lifts_record()).compareTo(rhs.getLeg_lifts_record());

        else if (sport.equals("Parallels"))
            answer = (-1) * Integer.valueOf(lhs.getParallels_record()).compareTo(rhs.getParallels_record());

        else if (sport.equals("Crunches"))
            answer = (-1) * Integer.valueOf(lhs.getCrunches_record()).compareTo(rhs.getCrunches_record());

        return answer;
    }
});
refreshMyListView();
```


הפונקציה בודקת באיזה סוג ספורט מדובר כיוון שלכל אחד יש רשימה משלו ולאחר מכן קוראת לאדפטר אשר עובר לשלב 3

```
public void refreshMyListView(){
    if (sport.equals("Push Ups")) recordAdapter = new records_adapter( context: LeaderBoard_pushups_Activity.this, resource
    else if (sport.equals("Pull Ups")) recordAdapter = new records_adapter( context: LeaderBoard_pushups_Activity.this, r
    else if (sport.equals("Leg Lifts")) recordAdapter = new records_adapter( context: LeaderBoard_pushups_Activity.this, r
    else if (sport.equals("Crunches")) recordAdapter = new records_adapter( context: LeaderBoard_pushups_Activity.this, r
    else if (sport.equals("Parallels")) recordAdapter = new records_adapter( context: LeaderBoard_pushups_Activity.this, r
    lv_leader_bord.setAdapter(recordAdapter);
}
```



```
0, textViewResourceid: 0, personArrayList, sportType: "push_ups");
source: 0, textViewResourceid: 0, personArrayList, sportType: "pull_ups");
resource: 0, textViewResourceid: 0, personArrayList, sportType: "leg_lifts");
source: 0, textViewResourceid: 0, personArrayList, sportType: "crunches");
resource: 0, textViewResourceid: 0, personArrayList, sportType: "parallels");
```

שלב 3: במחלקה של האדפטר – יצרנו קישור של כל TextView לפי ה-id, ונסביר לו כיצד למלא כל אחד מהם.

```
public class records_adapter extends ArrayAdapter<person> {
    Context context;
    List<person> objects;
    String sportName;

    public records_adapter(@NonNull Context context, int resource, int textViewResourceId, @NonNull List<person> objects, String sportType) {
        super(context, resource, textViewResourceId, objects);
        this.context = context;
        this.objects = objects;
        this.sportName = sportType;
    }

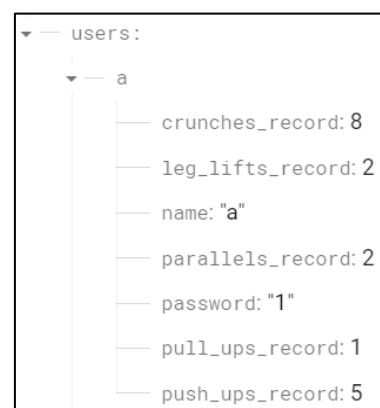
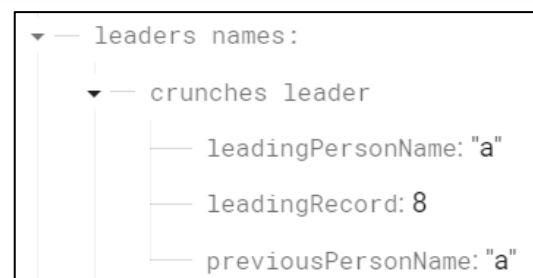
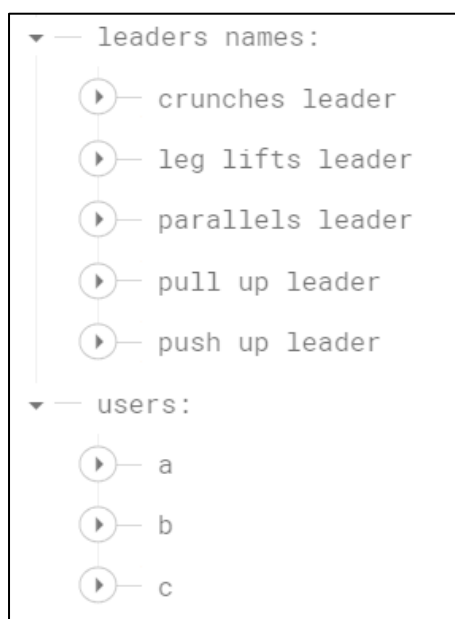
    @NonNull
    @Override
    public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {
        LayoutInflater inflater = ((Activity)context).getLayoutInflater();
        View view = inflater.inflate(R.layout.t_person_line, parent, attachToRoot: false);

        TextView tvPersonName = (TextView)view.findViewById(R.id.name);
        TextView tvPersonRecord = (TextView)view.findViewById(R.id.record);
        TextView tvPersonRank = (TextView)view.findViewById(R.id.rank);
        person temp = objects.get(position);
        tvPersonRank.setText("#"+String.valueOf(position+1));
        tvPersonName.setText(String.valueOf(temp.getName()));
        if (sportName.equals("push_ups")) tvPersonRecord.setText(""+temp.getPush_ups_record());
        if (sportName.equals("pull_ups")) tvPersonRecord.setText(""+temp.getPull_ups_record());
        if (sportName.equals("leg_lifts")) tvPersonRecord.setText(""+temp.getLeg_lifts_record());
        if (sportName.equals("crunches")) tvPersonRecord.setText(""+temp.getCrunches_record());
        if (sportName.equals("parallels")) tvPersonRecord.setText(""+temp.getParallels_record());
        return view;
    }
}
```

:Firebase

- הפיירבייס הוא שרת הנתון למפתחים על ידי גוגל המציע להחליף את כתיבת קוד צד השרת במערכת שלנו.
- אני משתמש בפיירבייס לשמור את נתוני המשתמשים כלומר שם המשתמש והסיסמה וכל שיאיו בתרגילים השונים ובנוסף בעזרתו אני יודע מי השיאן (מקום 1) בכל אחד מהתרגילים.
- Query הינו שאילתה המאפשרת לנו לבדוק את המידע המאוחסן ללא יכולת לשנותו.

```
if(view==btn_Signup){
    Query q = myRef.child("users:").orderByValue();
    q.addListenerForSingleValueEvent(new ValueEventListener() {
        @Override
        public void onDataChange(@NonNull DataSnapshot snapshot) {
            boolean userExist = false;
            for (DataSnapshot dst : snapshot.getChildren()){
                person user = dst.getValue(person.class);
                if (user.getName().equals(et_name.getText().toString())){
                    userExist = true;
                }
            }
            if (userExist==false){
                person newuser = new person(et_name.getText().toString());
                newuser.setPassword(et_password.getText().toString());
                myRef.child("users:").child(newuser.getName()).setValue(newuser);
                Intent i = new Intent( packageContext: MainActivity.this, YourRecordsActivity.class);
                startActivity(i);
            }
            if (userExist==true){
                Toast.makeText( context: MainActivity.this, text: "שם המשתמש תפוס", Toast.LENGTH_LONG).show();
            }
        }
    }
}
```



דיאלוג:

הדיאלוג מתבצע כתגובה לפעולה שהמשתמש עשה לדוגמה אצלי השתמש נכנס להסבר על אחד מתרגילי הכושר והדיאלוג שמתבצע הוא זה שכאשר המשתמש סיים לעיין בהסבר הוא צריך ללחוץ על "הבנתי"

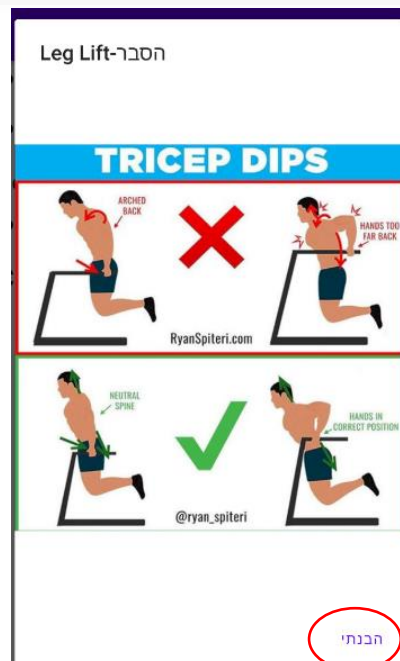
LayoutInflater משמש "לנפח" (=להציג משהו שעדיין לא קיים)

```
public boolean onOptionsItemSelected(MenuItem item){
    int id = item.getItemId();
    if(id==R.id.item_push_up) {
        AlertDialog.Builder builder = new AlertDialog.Builder( context: this);
        builder.setTitle("Push Up-הסבר");
        LinearLayout linearLayout = new LinearLayout( context: YourRecordsActivity.this);
        linearLayout.setOrientation(LinearLayout.HORIZONTAL);

        LayoutInflater factory = LayoutInflater.from(YourRecordsActivity.this);
        final View view = factory.inflate(R.layout.sample_dialog_pushup_xml, root: null);
        builder.setView(view);

        builder.setCancelable(true);
        builder.setPositiveButton( text: "הבנתי", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {

            }
        });
        AlertDialog dialog=builder.create();
        dialog.show();
    }
}
```



תפריט:

שאר ענפי הספורט לפי אותו עיקרון

מתחלק לשתי פונקציות:

1. יוצרת את התפריט
2. בוחרת איבר בתפריט

```
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.main_menu, menu);
    if (menu instanceof MenuBuilder) {
        MenuBuilder m = (MenuBuilder) menu;
    }
    return true;
}

public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();
    if (id == R.id.item_push_up) {
        AlertDialog.Builder builder = new AlertDialog.Builder(context: this);
        builder.setTitle("הסבר-Push Up");
    }
}
```

מטרתו של קובץ זה לתאר את כל ה activities, Receivers Broadcast, services וההרשאות שקיימות בפרויקט.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.project">

    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="project"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Project">
        <service
            android:name=".TryService"
            android:enabled="true"
            android:exported="true"></service>

        <receiver
            android:name=".WifiReceiver"
            android:enabled="false"
            android:exported="false" />

        <activity android:name=".YourRecordsActivity" />
        <activity android:name=".LeaderBoard_pushups_Activity" />
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

סיכום:

העבודה על הפרויקט הייתה לי מאוד חווייתית ומלמדת, במהלך העבודה נתקלתי בקשיים רבים ולמדתי איך להתמודד איתם ולפתור אותם.

במהלך העבודה נתקלתי בבעיה כאשר רציתי לשלוח הודעה לבחור שהיה במקום ראשון בשיא מסוים והוא נעקף הבעיה נוצרה בשל כך שבתכנון שלי ההודעה היא עבור שיא כללי ולא שיא מסויים ואם מישהו היה ראשון בשני שיאים ונעקף בשניהם אני ארצה שהוא יקבל הודעה אחת בלבד

פתרתי בעיה זאת בעזרת הוספה למאגר הנתונים (FireBase) מי מחזיק בשיא עכשיו ומי החזיק בוא לפני עבור כל תרגיל ובנוסף כאשר בדקתי אם הייתה עקיפה אני בודק גם אם אותו בחור קיבל כבר הודעה או לא.

אם היה לי עוד זמן הייתי מוסיף אפשרות להעלות סרטון שלך מבצע את השיא ולשתף אותו אם האנשים או אפשרות לפתוח "ליגות" של חברים ולא של כל המשתתפים

קישור git לקוד הפרויקט:

<https://github.com/avishay-oz/android-final-project>