

**קריית החינוך – אורט קריית באיליק**

**פרוייקט בתכנון ותכנות מערכות**

**שנה"ל תש"פ 2020**



**שם: טל בן עמי**

**ת"ז: 212525257**

**שם המנחים: איתמר פלדמן, עמליה אלמוג**

**תאריך: 30/04/2020**

# תוכן עניינים

## **פרק ראשון - הקדמה**

עמ' 3.....

## **פרק שני - תיאור התוכנית ומטרותיה**

\*מטרת הפרויקט ותיאור האופציות שנותנת פרויקט זה למשתמש. עמ' 4

\*תיאור מסכים כל מסך, תכולה ותפקיד. עמ' 5

\*תרשים זרימה של המסכים. עמ' 13

\*דרישות/מוגבלויות להפעלת התוכנה. עמ' 13

## **פרק שלישי – מדריך למפתח**

\*תיעוד חלוקת הקבצים השונים לספריות. עמ' 14

\*מבניות, מדולריות וחלוקה למחלקות. עמ' 15

\*תיאור קשרי המחלקות ע"י תרשים uml. עמ' 18

\*תיעוד המחלקות ע"י שימוש בתוכנת JAVADOC. עמ' 19

\*תיאור ארגון הנתונים הנשמרים ושיטת השמירה. עמ' 20

## **פרק רביעי - שימוש באבני יסוד**

## **פרק חמישי - מדריך משתמשים**

\*הסבר קצר על מהות התוכנה. עמ' 25

\*הסבר כיצד להתקין ולהריץ את התוכנה. עמ' 25

\*תיאור כיצד מפעילים ממשק המשתמש. עמ' 26

## **פרק שישי - פיצ'רים מיוחדים ואלגוריתמיקה**

## **פרק שביעי - רפלקציה**

## **פרק שמיני - נספחים**

\*מקורות מידע וקוד. עמ' 33

## פרק ראשון - הקדמה

הפרויקט "הכתבה" הוא אפליקציה של ממשק למשתמש התומך במספר אתרים שבן דודי בנה. הממשק תומך באתרים ללמידה ותרגול מילים בעברית ואנגלית וגם משמש כמפנה לאתרי תרגול מתמטיקה לילדים שבן דודי בנה. הממשק נותן אפשרות למשתמש לפתוח לעצמו חשבון, באותו חשבון המשתמש יכול לעצב רשימות מילים באנגלית ובעברית אותם יכול לשמור ולטעון. הם יבואו לשימוש האישי מתי שירצה ללמוד ולתרגל אותם. הממשק מקדד את המילים ברשימות המילים ושולח את המשתמש לאתר על מנת להתחיל את תרגולו שם. האפליקציה מכילה מסך הרשמה והתחברות חשבון, מסך אפשרויות לתרגול מתמטיקה ותרגול מילים באנגלית ועברית, מסך אופציות לטעינה, שמירה ועיצוב רשימות מילים באנגלית ועברית, מסך בחירת משחקים לתרגול מתמטיקה ומסך הפעלת התרגול של מילים באנגלית ועברית. האפליקציה מבצעת שימוש רציף ב firebase אשר שם נשמרים ונשלפים פרטי המשתמש ורשימות המילים שלו באנגלית ועברית. הכוונה באפליקציית ממשק למשתמש היא נתינת אפשרות למשתמש להשתמש בפונקציות האתר דרך האפליקציה חוץ מהמשחקים עצמם המובנים באתר.

לדוגמה:

The screenshot shows the app's main screen titled "הכתבה - אתר לתרגול עצמי של הכתבות". It features a list of words for a matching game, with red arrows pointing to specific elements and text boxes explaining their functions.

**Annotations and their targets:**

- מילים שנבחרו באפליקציה אשר שלחה את המשתמש לאתר זה** (Words selected in the app that sent the user to this site): Points to the word list.
- פונקציות שהמשתמש יכול לבחור באתר על מנת לתרגל את המילים שבחר באפליקציה** (Functions the user can choose on the site to practice the words they chose in the app): Points to the bottom navigation bar.
- האתר שאליו נשלח המשתמש** (The site to which the user was sent): Points to the "אתר" (Site) button.
- קשר הדדי בין האפליקציה לאתר** (Mutual connection between the app and the site): Points to the "אתר" and "האפליקציה" buttons.

**App Interface Elements:**

- Buttons: "אתר" (Site), "האפליקציה" (App), "מילים" (Words), "משחקים" (Games).
- Word List: 1. שלום, 2. היי, 3. מה, 4. אבא, 5. אמא, 6. אהבה, 7. , 8. , 9. , 10. .
- Bottom Bar: "משחקים", "תרגול מילים", "הכתבה", "קריאה עם קולות", "קריאה עם פורחים", "משחק עם כרטיסים", "משחק עם תפירות", "משחק עם תליות".

## פרק שני - תיאור התוכנית ומטרותיה

### מטרת הפרויקט:

מטרת הפרויקט היא לספק תמיכה לאתר שדוד שלי בנה בכך שהאפליקציה מהווה כממשק משתמש לתרגול מילים בעברית, אנגלית ותרגול מתמטיקה. האפליקציה תומכת באתרו של דוד שלי ומאפשרת למשתמש ליצור משתמש באפליקציה, להתחבר למשתמש שלו, לשמור ולטעון רשימות מילים אישיות שלו לשימוש האישי ללמידה ותרגול.

### תיאור האופציות שנותנת פרויקט זה למשתמש:

כאשר נכנסים לאפליקציה מופיע מסך המאפשר להתחבר למשתמש קיים באמצעות שם משתמש וסיסמה ומופיע כפתור המוביל למסך דיאלוג של יצירת משתמש. המשתמשים נשמרים ב firebase עם שמם, סיסמתם ורשימות המילים שלהם בעברית ואנגלית. נדרש חיבור אינטרנטי על מנת לבצע את הרשמת המשתמש או ההתחברות. כאשר נכנסים למשתמש מגיעים לתפריט הראשי המברך את המשתמש ומאפשר מספר בחירות שהם השמעת מוסיקת רקע, תרגול אנגלית, עברית ומתמטיקה. במסך תרגול המתמטיקה מוצעים משחקי חשבון של חיבור, חיסור, כפל וחילוק וישנם 7 משחקים בכל סוג של משחק חשבון. במסך תרגול האנגלית והעברית ישנה אפשרות למעבר למסך פתיחת ושמירת רשימות מילים של אותו משתמש, להוסיף ולמחוק מילים לרשימת המילים הנוכחית שעליה עובד המשתמש ולהתחיל את התרגול של רשימת המילים הנוכחית. במסך פתיחת ושמירת רשימות המילים ישנה אפשרות לפתיחת, מחיקת ושמירת רשימות מילים שונות ואפשרות למחוק מילים בתוך רשימת נוכחית שהמשתמש בחר לפתוח וישנה אפשרות חזרה למסך תרגול המילים. כל הפעולות שציינתי לעיל דורשות חיבור אינטרנטי של המשתמש שאותו האפליקציה דורשת ולא נותנת למשתמש לבצע את הפעולות ושולחת למשתמש הודעות מתאימות הדורשות חיבור אינטרנטי.

## תיאור המסכים ותכולתם:

### מסך כניסה למשתמש (lo\_login\_screen)

שימוש ברכיבי: Text View, Edit Text, Check Box, Button

Intent, Menu

בסיס נתונים firebase



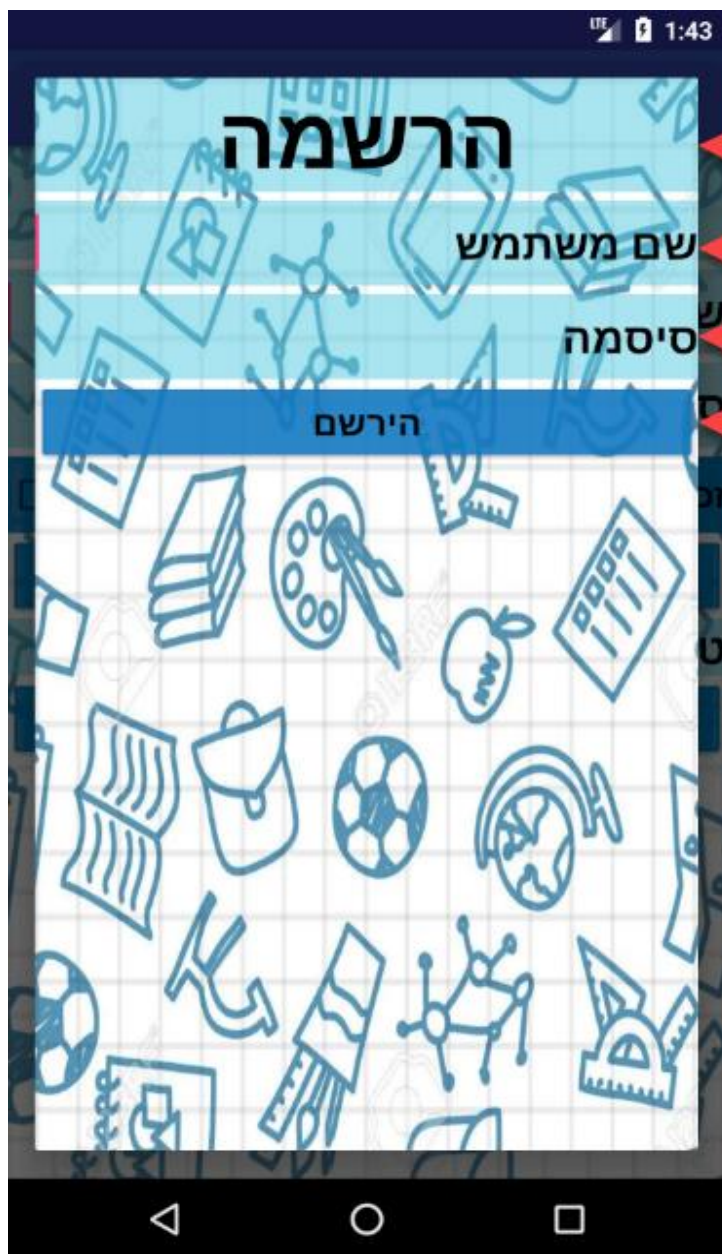


## דיאלוג יצירת/הרשמת משתמש (d\_register)

שימוש ברכיבי: Text View, Edit Text, Button

Intent

בסיס נתונים firebase



**tvWelcome**

**etUserName**

**etPassword**

**btnRegister**

## מסך אפשרויות (lo\_options)

שימוש ברכיבי: Text View, Button

Intent, Menu



## מסך תרגול מילים באנגלית/עברית (lo\_hebrew\_or\_english)

שימוש ברכיבי: Text View, Edit Text, Button, List View

Intent, Menu





מסך שמירת וטעינת רשימת מילים באנגלית/עברית (lo\_save\_or\_load\_word\_list)

שימוש ברכיבי: Text View, Edit Text, Button, List View

Intent, Menu

בסיס נתונים firebase



## מסך סוגי משחקי מתמטיקה (lo\_math)

שימוש ברכיבי: Text View, Button

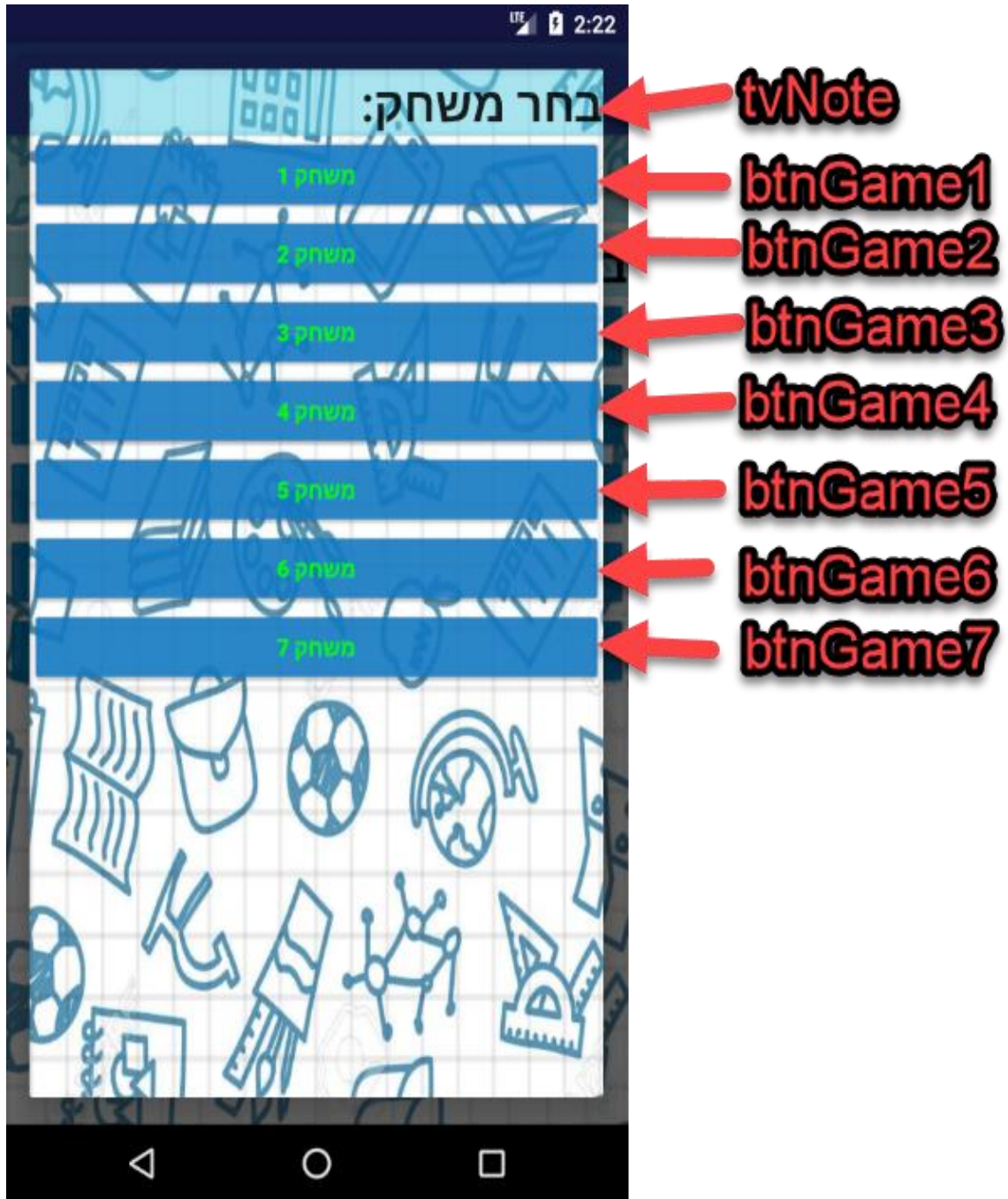
Intent, Menu



## דיאלוג משחקי מתמטיקה (d\_math\_games)

שימוש ברכיבי: Text View, Button

Intent



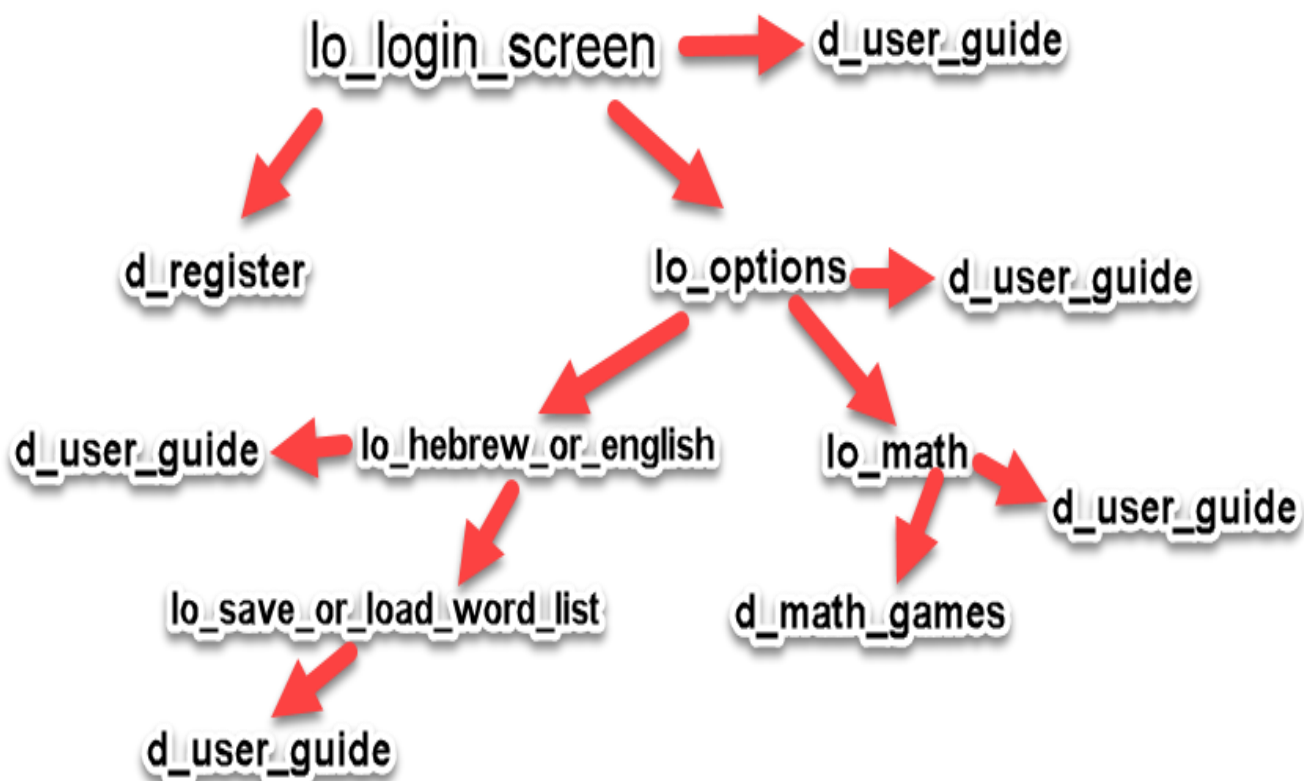


## דיאלוג מדריך למשתמש (d\_user\_guide)

שימוש ברכיבי: Text View, Scroll View, Image View



## תרשים זרימה של המסכים:



## דרישות להפעלת האפליקציה:

**API 15+\***

**\*אינטרנט**



# פרק שלישי - נתונים

## תיעוד חלוקת הקבצים השונים לספריות:

### תיקיית: `java.com.example.comps.myapplication`

BackgroundSoundService – הקוד של הסרויס של מוזיקת הרקע  
HebrewOrEnglish – הקוד עבור מסך תרגול המילים באנגלית ועברית  
ListOfWordList – מחלקה עבור רשימה של רשימות מילים  
ListOfWordListAdapter – custom adapter עבור המחלקה רשימה של רשימות מילים

LoginScreen – הקוד עבור מסך ההתחברות

Math – הקוד עבור מסך לימוד מתמטיקה

Options – הקוד עבור המסך המכיל את אפשרויות האפליקציה

SaveOrLoadWordList – הקוד עבור המסך השומר וטוען רשימות מילים

User – מחלקה עבור המשתמש

WordList – מחלקה עבור רשימת מילים

WordListAdapter - custom adapter עבור המחלקה רשימה של מילים

### תיקיית: `res`

### תיקיית: `drawable` - מכיל את קבצי התמונות שיש בהן שימוש באפליקציה

login – רקע ראשי לאפליקציה

Pic1 – מדריך למשתמש

### תיקיית: `layout` - מכיל את קבצי ה `xml` של כל מסכי האפליקציה

custom\_row – שורה עבור listviews בפרויקט

d\_math\_add\_games – דיאלוג למשחקי מתמטיקה

d\_register – דיאלוג ליצירת משתמש

d\_user\_guide – דיאלוג מדריך למשתמש

lo\_hebrew\_or\_english - מסך תרגול המילים באנגלית ועברית

lo\_login\_screen - מסך ההתחברות

lo\_math - מסך לימוד מתמטיקה

lo\_save\_or\_load\_word\_list - המסך השומר וטוען רשימות מילים

lo\_options - המסך המכיל את אפשרויות האפליקציה

**תיקיית: menu - מכיל את קובץ ה xml של התפריט**

menu – קובץ xml של התפריט

**תיקיית: raw - מכיל את קובץ מוזיקת הרקע אשר רץ ברקע בעת הרצת האפליקציה**

relaxing\_music - קובץ מוזיקה

**תיקיית התחלה: app.src.main**

AndroidManifest - קובץ XML של הרשאות, רישום מסכים,  
BroadcastReceiver ו- Service

## **מבניות, מדולריות וחלוקה למחלקות:**

**User:** מחלקה המחזיקה את נתוני המשתמש ורשימות המילים שלו

userName – שם המשתמש

password – סיסמת המשתמש

hebrewListOfWordList – רשימה של רשימת מילים בעברית של המשתמש

englishListOfWordList - רשימה של רשימת מילים באנגלית של המשתמש

```
package com.example.comps.myapplication;

import java.io.Serializable;
import java.util.List;

/**
 * This class holds the user who has his own username, password, hebrew and english word lists.
 * This class have getters, setters and constructors.
 */

public class User implements Serializable {
    private String userName;
    private String password;
    private ListOfWordList hebrewListOfWordList = new ListOfWordList();
    private ListOfWordList englishListOfWordList = new ListOfWordList();

    public User() {
    }

    public User(String userName, String password) {
        this.userName = userName;
    }
}
```

```

        this.password = password;
    }

    public ListOfWordList getEnglishListOfWordList() {
        return englishListOfWordList;
    }

    public void setEnglishListOfWordList(ListOfWordList englishListOfWordList) {
        this.englishListOfWordList = englishListOfWordList;
    }

    public ListOfWordList getHebrewListOfWordList() {
        return hebrewListOfWordList;
    }

    public void setHebrewListOfWordList(ListOfWordList hebrewListOfWordList) {
        this.hebrewListOfWordList = hebrewListOfWordList;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getUsername() {
        return userName;
    }

    public void setUsername(String userName) {
        this.userName = userName;
    }
}

```

## **WordList**: מחלקה המחזיקה רשימת מילים

words – רשימת מילים

listName – שם הרשימה

```

package com.example.comps.myapplication;

import java.io.Serializable;
import java.util.ArrayList;

/**
 * This class is for holding a list of words.
 * This class have getters, setters and constructors.
 */

public class WordList implements Serializable {
    private ArrayList<String> words = new ArrayList<>();
    private String listName;

    WordList() {
    }

    WordList(ArrayList<String> words, String listName) {
        this.words = words;
        this.listName = listName;
    }
}

```

```

}

@Override
public String toString() {
    return listName;
}

public ArrayList<String> getWords() {
    return words;
}

public void setWords(ArrayList<String> words) {
    this.words = words;
}

public String getListName() {
    return listName;
}

public void setListName(String listName) {
    this.listName = listName;
}
}

```

## ListOfWordList: מחלקה המחזיקה רשימה של רשימת מילים

listOfWordList – רשימה של רשימת מילים

```

package com.example.comps.myapplication;

import java.io.Serializable;
import java.util.ArrayList;

/**
 * This class is for holding a list that holds word lists.
 * This class have getters, setters and constructors.
 */

public class ListOfWordList implements Serializable {
    private ArrayList<WordList> listOfWordList = new ArrayList<>();

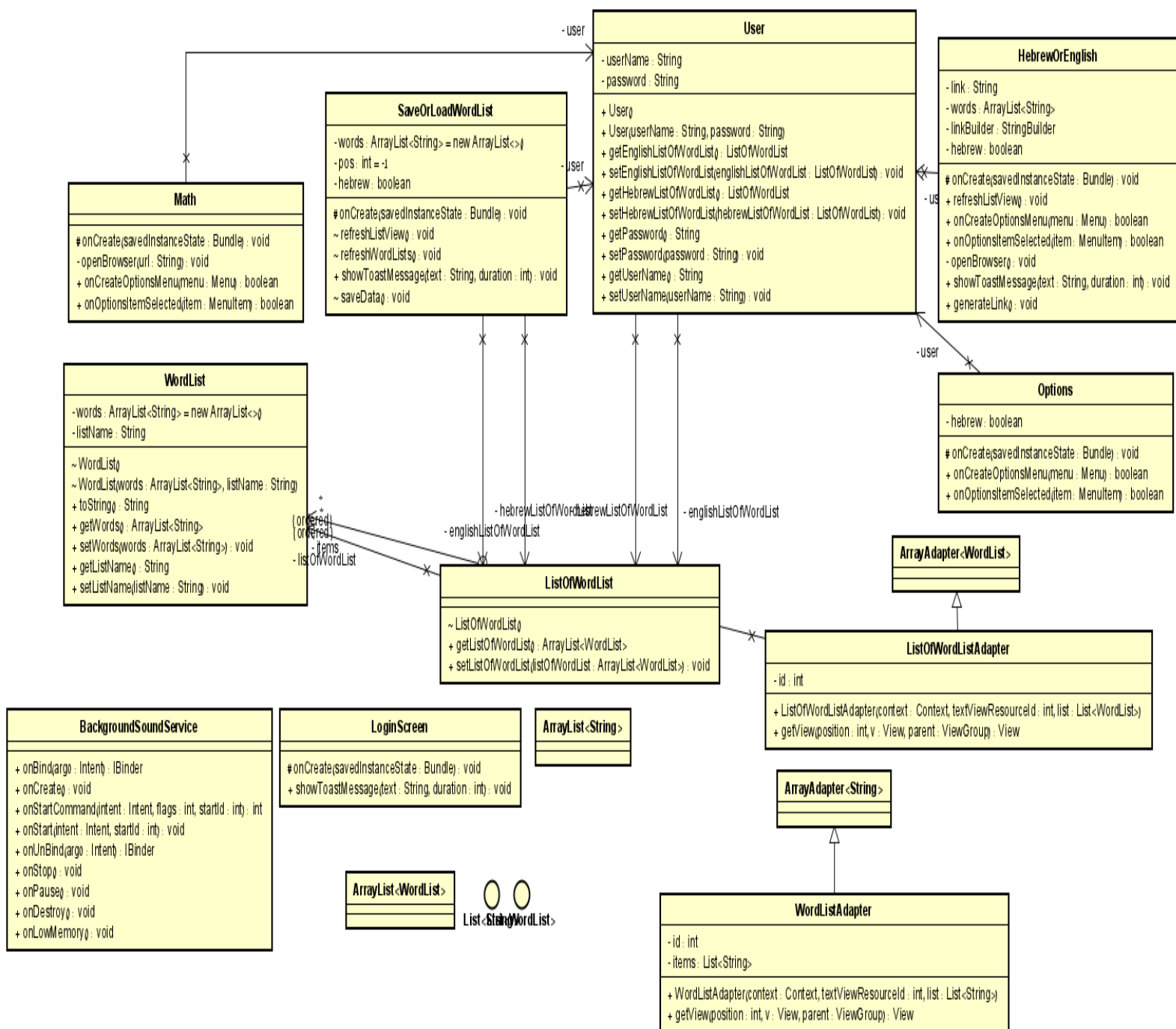
    ListOfWordList() {
    }

    public ArrayList<WordList> getListOfWordList() {
        return listOfWordList;
    }

    public void setListOfWordList(ArrayList<WordList> listOfWordList) {
        this.listOfWordList = listOfWordList;
    }
}

```

# תיאור קשרי המחלקות ע"י תרשים UML:





# תיעוד המחלקות ע"י שימוש בתוכנת :JAVADOC

[https://drive.google.com/drive/folders/1LQ4aqt1OAWKyVtSoZQTqmIkN\\_I5rB7qm?usp=sharing](https://drive.google.com/drive/folders/1LQ4aqt1OAWKyVtSoZQTqmIkN_I5rB7qm?usp=sharing)

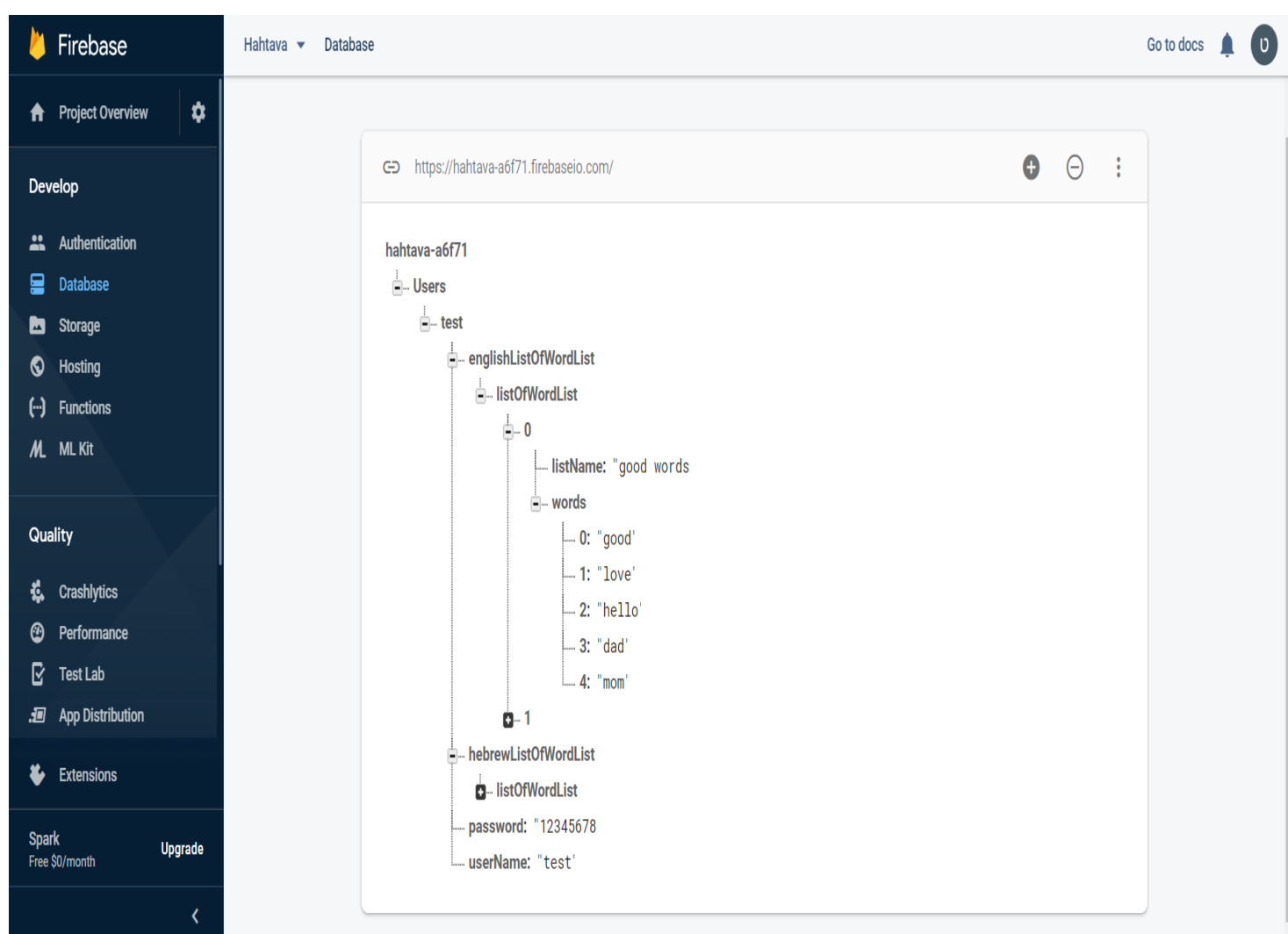
All Classes	PACKAGE CLASS TREE DEPRECATED INDEX HELP																								
BackgroundSoundService HebrewOrEnglish ListOfWordList ListOfWordListAdapter LoginScreen Math Options SaveOrLoadWordList User WordList WordListAdapter	PREV PACKAGE NEXT PACKAGE FRAMES NO FRAMES																								
	Package com.example.comps.myapplication																								
	<b>Class Summary</b>																								
	<table><tr><th>Class</th><th>Description</th></tr><tr><td>BackgroundSoundService</td><td>This service allows to start background music to the user.</td></tr><tr><td>HebrewOrEnglish</td><td>This activity is where the words' practice happens.</td></tr><tr><td>ListOfWordList</td><td>This class is for holding a list that holds word lists.</td></tr><tr><td>ListOfWordListAdapter</td><td>This is a custom adapter that changes the strings in a listview to blue.</td></tr><tr><td>LoginScreen</td><td>This activity is the login screen for the user where he can log-in with his user and password or register.</td></tr><tr><td>Math</td><td>This activity allows the user to play math games that he chooses and go back to the menu.</td></tr><tr><td>Options</td><td>This activity gives the user the option to choose if he would like to practice hebrew or english words or to play math games.</td></tr><tr><td>SaveOrLoadWordList</td><td>This activity allows the user to save or load word lists of his own in or from his data base.</td></tr><tr><td>User</td><td>This class holds the user who has his own username, password, hebrew and english word lists.</td></tr><tr><td>WordList</td><td>This class is for holding a list of words.</td></tr><tr><td>WordListAdapter</td><td>This is a custom adapter that changes the strings in a listview to black.</td></tr></table>	Class	Description	BackgroundSoundService	This service allows to start background music to the user.	HebrewOrEnglish	This activity is where the words' practice happens.	ListOfWordList	This class is for holding a list that holds word lists.	ListOfWordListAdapter	This is a custom adapter that changes the strings in a listview to blue.	LoginScreen	This activity is the login screen for the user where he can log-in with his user and password or register.	Math	This activity allows the user to play math games that he chooses and go back to the menu.	Options	This activity gives the user the option to choose if he would like to practice hebrew or english words or to play math games.	SaveOrLoadWordList	This activity allows the user to save or load word lists of his own in or from his data base.	User	This class holds the user who has his own username, password, hebrew and english word lists.	WordList	This class is for holding a list of words.	WordListAdapter	This is a custom adapter that changes the strings in a listview to black.
Class	Description																								
BackgroundSoundService	This service allows to start background music to the user.																								
HebrewOrEnglish	This activity is where the words' practice happens.																								
ListOfWordList	This class is for holding a list that holds word lists.																								
ListOfWordListAdapter	This is a custom adapter that changes the strings in a listview to blue.																								
LoginScreen	This activity is the login screen for the user where he can log-in with his user and password or register.																								
Math	This activity allows the user to play math games that he chooses and go back to the menu.																								
Options	This activity gives the user the option to choose if he would like to practice hebrew or english words or to play math games.																								
SaveOrLoadWordList	This activity allows the user to save or load word lists of his own in or from his data base.																								
User	This class holds the user who has his own username, password, hebrew and english word lists.																								
WordList	This class is for holding a list of words.																								
WordListAdapter	This is a custom adapter that changes the strings in a listview to black.																								
	PACKAGE CLASS TREE DEPRECATED INDEX HELP																								
	PREV PACKAGE NEXT PACKAGE FRAMES NO FRAMES																								

## תיאור ארגון הנתונים הנשמרים ושיטת השמירה:

הנתונים הנשמרים נשמרים באמצעות מחלקת User המכילה את התכונות שם משתמש, סיסמה, רשימות המילים לתרגול של המשתמש בעברית ואנגלית ניתן לשמור את רשימות המילים של המשתמש המעוניין לתרגל אותם יותר מאוחר

המשתמש טוען את רשימת המילים אשר מעוניין לתרגל באמצעות שליפה ממבנה הנתונים של אותו המשתמש

הנתונים נשמרים באמצעות Firebase



## פרק רביעי - שימוש באבני יסוד:

### :Activity

באפליקציה יש 5 Activities. לכל Activity יש מסך - Layout משלו שעוצב בהתאם.

באפליקציה קיימים: מסך התחברות, מסך בחירת אפשרויות, מסך תרגול עברית/אנגלית, מסך שמירה וטעינת רשימות מילים ומסך תרגול מתמטיקה.

דוגמה לפתיחת Activity:

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.io_login_screen);  
}
```

### :Intent

באפליקציה נעשה שימוש בIntent עבור שלוש מטרות:

מעבר בין מסכים

העברת נתונים בין מסכים

התחברות ל Service

דוגמה לשימוש בIntent למעבר ממסך ההתחברות למסך האפשרויות:

```
Intent myIntent = new Intent(LoginScreen.this, Options.class);  
myIntent.putExtra("user", u);  
startActivity(myIntent);
```

### :Dialog

באפליקציה אנו עושים שימוש בדיאלוג אשר פירושו חלון קופץ המשמש לתצוגה של סטטוס ואינטראקציה עם המשתמש. אנדרואיד מספק classes המאפשרים טיפול פשוט בדיאלוגים

דוגמה לפתיחת דיאלוג ליצירת חשבון למשתמש:

```
Dialog d = new Dialog(LoginScreen.this);  
d.setContentView(R.layout.d_register);  
d.show();
```





## :Menu

אנו עושים שימוש ב menu על מנת לאפשר למשתמש להגדיר את ההגדרות וההעדפות שלו. כפתור פתיחת ה menu נמצא למעלה בצד ימין באפליקציה. באפליקציה נעשה שימוש ב menu על מנת לתת אפשרות למשתמש להתנתק מחשבונו ולהגיע לדיאלוג מדריך למשתמש.

דוגמה לבניית ה menu:

```
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu, menu);
    return super.onCreateOptionsMenu(menu);
}
```

דוגמה להפעלת ה menu:

```
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId())
    {
        case R.id.itLogout:
            Intent myIntent = new Intent(Math.this, LoginScreen.class);
            SharedPreferences preferences =
            getSharedPreferences("checkbox", MODE_PRIVATE);
            SharedPreferences.Editor editor = preferences.edit();
            editor.putString("remember", "no");
            editor.apply();
            startActivity(myIntent);
            break;
        case R.id.itUserGuide:
            Dialog d = new Dialog(Math.this);
            d.setContentview(R.layout.d_user_guide);
            d.show();
            break;
    }
    return super.onOptionsItemSelected(item);
}
```

## :Service

אנו עושים שימוש ב Service על מנת להפעיל תהליך מסוים ברקע אשר ממשיך לפעול גם במידה אם המסך של ה-Activity שהפעיל את ה-Service יורד מהתצוגה. באפליקציה אנו עושים שימוש ב Service למטרת הפעלת נגן מוסיקה ברקע.

דוגמה להפעלת תהליך ה Service (באמצעות Intent):

```
Intent svc = new Intent(Options.this, BackgroundSoundService.class);
startService(svc);
```

מחלקת ה Service המפעילה את המוסיקה:

```
public class BackgroundSoundService extends Service {

    private MediaPlayer player;

    public IBinder onBind(Intent arg0) {

        return null;
    }

    @Override
```

```

public void onCreate() {
    super.onCreate();
    player = MediaPlayer.create(this, R.raw.relaxing_music);
    player.setLooping(true);
    player.setVolume(100, 100);
}

public int onStartCommand(Intent intent, int flags, int startId) {
    player.start();
    return Service.START_STICKY;
}

```

## :SharedPreferences

באפליקציה יש שימוש ב SharedPreferences על מנת לתת למשתמש אפשרות לשמור את פרטי התחברותו באפליקציה ולהתחבר עם פרטים אלו אוטומטית בכניסות הבאות לאפליקציה (אפשרות "זכור אותי" במסך הכניסה).

שמירת פרטי המשתמש במידה ולחצן ה-"זכור אותי" מופעל.

```

if (cbRemember.isChecked())
{
    SharedPreferences preferences = getSharedPreferences("checkbox",MODE_PRIVATE);
    SharedPreferences.Editor editor = preferences.edit();
    editor.putString("remember","yes");
    editor.putString("userName",etUserName.getText().toString());
    editor.putString("password",etPassword.getText().toString());
    editor.apply();
}
else
{
    SharedPreferences preferences = getSharedPreferences("checkbox",MODE_PRIVATE);
    SharedPreferences.Editor editor = preferences.edit();
    editor.putString("remember","no");editor.apply();
}

```

שליפת הפרטים בעזרת SharedPreferences. במידה והלחצן "זכור אותי" היה מופעל, פרטי המשתמש נשלפים ומוכנסים למערכת והיא מבצעת התחברות אוטומטית (יש טיימר לדילי בהפעלת הקוד מכיוון שה- BroadcastReceiver צריך זמן על מנת לפעול).

```

preferences = getSharedPreferences("checkbox",MODE_PRIVATE);
String checkbox = preferences.getString("remember","");
if(checkbox.equals("yes"))
{
    etUserName.setText(preferences.getString("userName",""));
    etPassword.setText(preferences.getString("password",""));
    cbRemember.setChecked(true);
    final Handler handler = new Handler();
    handler.postDelayed(new Runnable() {
        @Override
        public void run() {
            btnLogin.performClick();
        }
    }, 100);
}

```

ביטול הלחצן "זכור אותי" במידה והמשתמש מבצע התנתקות מהמשתמש בעזרת ה Menu.



```
public boolean onOptionsItemSelected(MenuItem item) {
    Intent myIntent = new Intent(HebrewOrEnglish.this, LoginScreen.class);
    SharedPreferences preferences = getSharedPreferences("checkbox",MODE_PRIVATE);
    SharedPreferences.Editor editor = preferences.edit();
    editor.putString("remember","no");
    editor.apply();
    startActivity(myIntent);
    return super.onOptionsItemSelected(item);
}
```

## :BroadcastReceiver

באפליקציה יש שימוש ב BroadcastReceiver על מנת לבדוק את חיבור המשתמש לאינטרנט. רוב האפליקציה מסתמכת על חיבור אינטרנטי של המשתמש ולכן כאשר אין חיבור זה, האפליקציה מונעת מהמשתמש לתרגל מילים באנגלית ועברית, לתרגל מתמטיקה, לשמור ולפתוח את רשימות המילים של המשתמש.

```
BroadcastReceiver receiver = new BroadcastReceiver() {
    @Override public void onReceive(Context context, Intent _ )
    {
        ConnectivityManager connMgr = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
        NetworkInfo wifi = connMgr.getNetworkInfo(ConnectivityManager.TYPE_WIFI);
        NetworkInfo mobile = connMgr.getNetworkInfo(ConnectivityManager.TYPE_MOBILE);

        boolean isConnected = wifi != null && wifi.isConnectedOrConnecting() ||
            mobile != null && mobile.isConnectedOrConnecting();
        if (isConnected) {
            showMessage("לאינטרנט מחובר",1500);
            btnPractice.setTextColor(Color.GREEN);
            btnOpenOrLoadWordList.setTextColor(Color.GREEN);
        } else {
            showMessage("לאינטרנט חיבור אין",1500);
            btnPractice.setTextColor(Color.RED);
            btnOpenOrLoadWordList.setTextColor(Color.RED);
        }
    }
};
this.registerReceiver( receiver, new IntentFilter("android.net.conn.CONNECTIVITY_CHANGE") );
```

## :Handler

באפליקציה עשיתי שימוש ב Handler בתור טיימר שאחריו ביצעתי פעולות מסוימות כגון: כיבוי הודעת Toast ולחיצה על כפתור. ה Handler עושה שימוש ב Thread אשר תפקידו לרוץ עם הפעלתה של התוכנית ולבצע פעולות מסוימות בזמן ריצת התוכנית. באפליקציה שלי ה Handler מבקש מה Runnable לרוץ על ה Thread המחובר ל Handler ולבצע פעולת טיימר בזמן ריצת התוכנית וביצוע פעולה בסיום זמן הטיימר. (פעולת run של ה Runnable) דוגמה 1:

```
Handler handler = new Handler();
handler.postDelayed(new Runnable() { @Override
    public void run() {
        toast.cancel();}}, duration);
```

דוגמה 2:

```
Handler handler = new Handler();
handler.postDelayed(new Runnable() { @Override
    public void run() {
        btnLogin.performClick();}}, 100);
```

# פרק חמישי – מדריך למשתמש:

## מהות התוכנה:

התוכנה היא אפליקציית ממשק משתמש התומכת במספר אתרים שבנה בן דודי. אתרים אלו הם אתרי תרגול של מילים באנגלית, עברית ואתרי תרגול מתמטיקה המכילים גם משחקים אטרקטיביים לתרגול מקצועות אלו. האפליקציה מוסיפה פונקציית יצירה של חשבון והתחברות לחשבון אשר באפשרותה לספק למשתמש אפשרות ליצור, לשמור ולפתוח רשימות מילים משלו שברצונו לתרגל אותם. שמירת פרטי המשתמש ורשימות המילים שאותם מעוניין לשמור ולשלוף אותם לתרגול נשמרים בעזרת מבנה נתונים ב firebase.

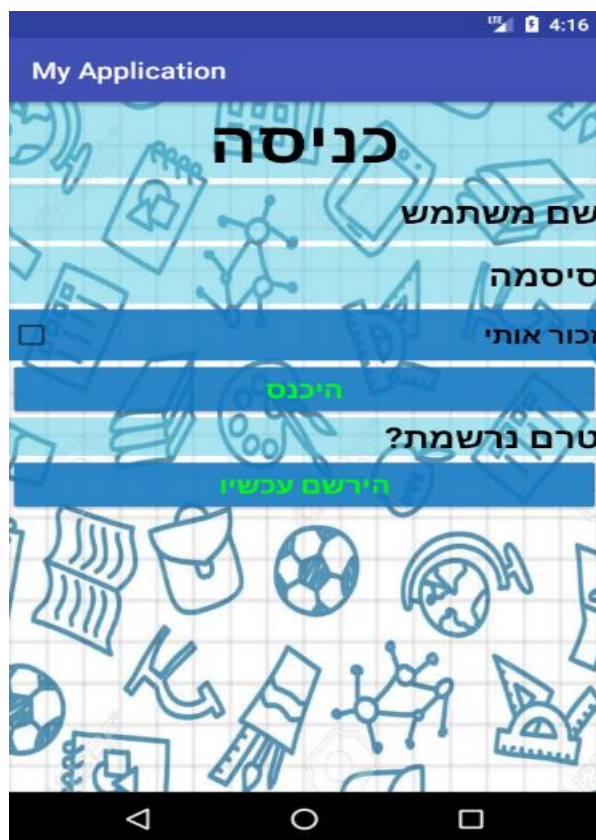
התוכנה נותנת אפשרות למשתמש לפתוח לעצמו חשבון, באותו חשבון המשתמש יכול לעצב רשימות מילים באנגלית ובעברית אותם יכול לשמור ולפתוח. הם יבואו לשימוש האישי מתי שירצה ללמוד ולתרגל אותם. הממשק מקדד את המילים ברשימות המילים ושולח את המשתמש לאתר על מנת להתחיל את תרגולו שם. האפליקציה מכילה מסך הרשמה והתחברות חשבון, מסך אפשרויות לתרגול מתמטיקה ותרגול מילים באנגלית ועברית, מסך אופציות לטעינה, שמירה ועיצוב רשימות מילים באנגלית ועברית, מסך בחירת משחקים לתרגול מתמטיקה ומסך הפעלת התרגול של מילים באנגלית ועברית. האפליקציה מבצעת שימוש רציף ב firebase אשר שם נשמרים ונשלפים פרטי המשתמש ורשימות המילים שלו באנגלית ועברית.

## כיצד להתקין ולהריץ את התוכנה:

<https://drive.google.com/file/d/1TKzA3wYD0buDTGdl20VsoGxXBIPAHXW8/view?usp=sharing>

קישור זה הוא קישור דרכו ניתן להוריד apk של האפליקציה. כאשר מורידים את ה apk תינתן אפשרות להתקין אותו בפלאפון ולאחר ההתקנה יהיה ניתן להשתמש באפליקציה באופן חופשי בתנאי שהפלאפון עומד בדרישות ההפעלה.

## תיאור כיצד מפעילים את ממשק המשתמש:

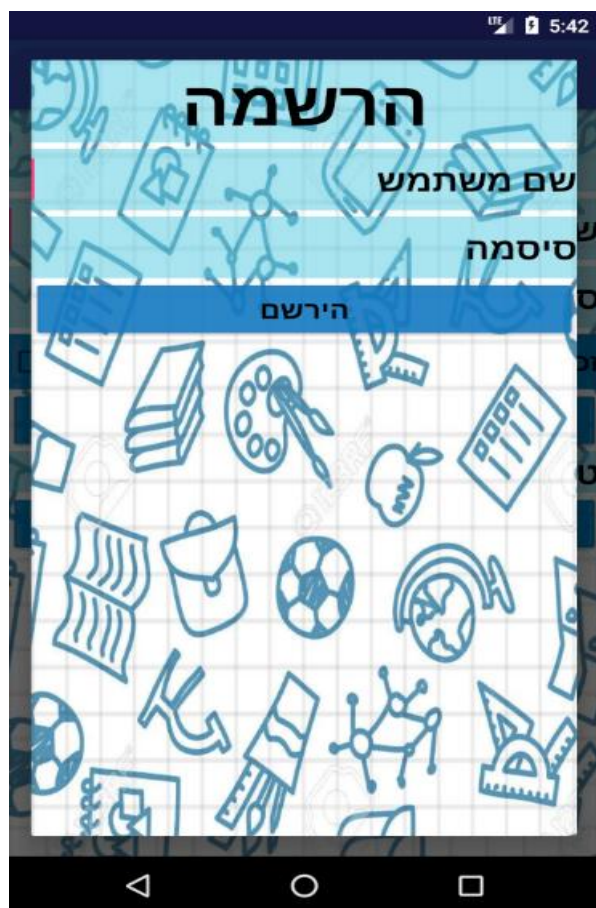


### מסך הכניסה למשתמש

במסך זה ניתן להיכנס למשתמש קיים באמצעות משתמש קיים בעל שם משתמש וסיסמה. הכפתור הראשון מוביל לתפריט הראשי בתנאי שהשם משתמש והסיסמה נכונים והכפתור השני מוביל למסך יצירת משתמש. ניתן ללחוץ על הכפתור זכור אותי המאפשר התחברות אוטומטית למשתמש בכניסות הבאות לאפליקציה עד שיבחר להתנתק מחשבנו דרך ה Menu.

### דיאלוג יצירת משתמש

בדיאלוג זה ניתן ליצור משתמש שנתוניו ישמרו ב Firebase וכך פרטיו ורשימות מילותיו לא ימחקו. לאחר יצירת המשתמש ניתן להשתמש בו כדי להתחבר במסך הכניסה.





## מסך האפשרויות

המסך הראשון לאחר כניסת המשתמש המברך אותו ומציב בפניו אפשרויות בחירה:

- (1) לימוד עברית.
- (2) לימוד אנגלית.
- (3) לימוד מתמטיקה.
- (4) השמעת מוסיקה.
- (5) יש Menu המאפשר למשתמש להתנתק מחשבנו.



## מסך תרגול עברית/אנגלית

במסך זה ניתן לתרגל מילים הנמצאים ברשימת המילים. יש למשתמש את האפשרויות הבאות:

- (1) לערוך את הרשימה ע"י הוספת ומחיקת מילים מהרשימה.
- (2) לתרגל את המילים מהרשימה.
- (3) לעבור למסך שמירת הרשימה ופתיחת רשימה שמורה.
- (4) לחזור למסך התפריט.
- (5) יש Menu המאפשר למשתמש להתנתק מחשבנו.





## מסך שמירת וטעינת רשימות מילים

- במסך זה ניתן לפתוח ולשמור רשימות מילים של אותו חשבון המחובר למערכת. יש למשתמש את האפשרויות הבאות:
- (1) למחוק מילים מרשימת המילים הנוכחית.
  - (2) לשמור את רשימת המילים הנוכחית ("הוסף רשימה").
  - (3) לפתוח רשימה שמורה.
  - (4) למחוק רשימת מילים מסומנת.
  - (5) לחזור למסך תרגול המילים.



## מסך תרגול מתמטיקה

- במסך זה ניתן לבחור סוג משחקי מתמטיקה אשר המשתמש מעוניין לשחק. יש למשתמש את האפשרויות הבאות:
- (1) משחקי חיבור
  - (2) משחקי חיסור
  - (3) משחקי כפל
  - (4) משחקי חילוק
  - (5) לחזור למסך התפריט.
  - (6) יש Menu המאפשר למשתמש להתנתק מחשבון.





## דיאלוג בחירת משחק מתמטיקה

בדיאלוג זה ניתן לבחור משחקי מתמטיקה אשר המשתמש מעוניין לשחק. יש למשתמש את האפשרויות הבאות:

- 1 משחק 1
- 2 משחק 2
- 3 משחק 3
- 4 משחק 4
- 5 משחק 5
- 6 משחק 6
- 7 משחק 7

## דיאלוג מדריך למשתמש

בדיאלוג זה ניתן לגלגל במדריך המשתמש של האפליקציה אשר מופיע לעיל.

## פרק שישי – פיצ'רים מיוחדים ואלגוריתמיקה:

### פיצ'רים מיוחדים:



#### \*פיצ'ר Speech To Text (STT)

באפליקציה יש פיצ'ר (המיקרופון בתמונה) אשר מאזין למילה וממיר אותה לטקסט בעזרת Google STT.

### אלגוריתמיקה:

#### \*אלגוריתם השמירה

באפליקציה הנתונים נשמרים ב firebase. כל פעם שמשתמש פותח חשבון ומוסיף רשימת מילים חדשה הנתונים הללו נשמרים. העדכון נעשה באמצעות פעולה שמעדכנת את הנתונים של המשתמש עם השינוי שנעשה.

```
void saveData() {  
    user.setHebrewListOfWordList(hebrewListOfWordList);  
    ref.child("Users").child(user.getUserName()).setValue(user);  
    user.setEnglishListOfWordList(englishListOfWordList);  
    ref.child("Users").child(user.getUserName()).setValue(user);  
}
```

#### \*אלגוריתם השליפה

באפליקציה הנתונים נשלפים מה firebase. כל פעם שמשתמש מנסה להתחבר לחשבון, פרטי כל החשבונות נשלפים וישנה בדיקה האם שם המשתמש והסיסמה של המשתמש קיימים במבנה הנתונים. בכל פעם שהמשתמש נכנס למסך שמירת וטעינת רשימות מילים, שמותיהן של רשימות המילים של החשבון נשלפות מהמערכת ומופיעות ב ListView. כאשר המשתמש לוחץ על רשימה, רשימת המילים עם המילים עצמם נפתחת גם היא ב ListView אחר.

```
Query q = ref.child("Users").orderByValue();  
q.addListenerForSingleValueEvent(new ValueEventListener() {  
    @Override  
    public void onDataChange(DataSnapshot dataSnapshot) {  
        if (etUserName.getText().toString().length() != 0 &&  
            etPassword.getText().toString().length() != 0) {  
            for (DataSnapshot dst : dataSnapshot.getChildren()) {  
                User u = dst.getValue(User.class);  
            }  
        }  
    }  
});
```

## \*אלגוריתם ריבוי המסכים

באפליקציה מתבצע ריבוי מסכים, כלומר יצרתי מסכים שונים מ xml אחד, בעזרת שימוש בתנאים מתאימים בקוד של ה Activity. לדוגמה ישנו מסך תרגול של עברית ומסך תרגול של אנגלית. שני מסכים אלו בנויים מאותו xml והם אותו Activity. השימוש באלגוריתם זה עזר לי לחסוך המון שורות קוד משוכפלות ומיותרות. דוגמת קוד קטנה:

```
if (hebrew) {  
    tvTitle.setText("עברית");  
} else {  
    tvTitle.setText("אנגלית");  
}
```

## \*אלגוריתם הקידוד

באפליקציה תרגול המילים בעברית ובאנגלית מתבצעים באתרים שכן דודי בנה ע"י שליחתם לאתרים. שליחת המילים מתבצעת ע"י קידוד סודי שכן דודי החליט עליו ושיתף אותי בו. קידוד זה חשוב מאוד לשם חיבור האפליקציה עם האתר. הוא מאפשר למשתמש לתרגל את מילותיו שרשם באפליקציה אשר מקודדות ועוברות באופן מדויק מהאפליקציה ישירות לאתר.

דוגמה חלקית לקידוד: (הקידוד מצונזר)

```
public void generateLink() {  
    if (hebrew) {  
        linkBuilder = new  
StringBuilder("https://www.hachtava.co.il/index.html?");  
    } else {  
        linkBuilder = new  
StringBuilder("https://www.hachtava.co.il/english.html?");  
    }  
    for (String word : words) {  
        linkBuilder.append(*censored*);  
        for (int i = 0; i < word.length(); i++) {  
            String character = Character.toString(word.charAt(i));  
            switch (character) {  
                case "א":  
                    linkBuilder.append(*censored*);  
                    break;  
                case "ב":  
                    linkBuilder.append(*censored*);  
                    break;  
                case "ג":  
                    linkBuilder.append(*censored*);  
                    break;  
                case "ד":  
                    linkBuilder.append(*censored*);  
                    break;  
                case "ה":  
                    linkBuilder.append(*censored*);  
                    break;  
                case "ו":  
                    linkBuilder.append(*censored*);  
                    break;  
            }  
        }  
    }  
}
```

## פרק שביעי – רפלקציה:

העבודה על הפרויקט "הכתבה" הייתה מהנה ומאתגרת עבורי.

נהייתי מלראות במהלך היצירה של המשחק את ההתקדמות אשר נתנה לי מוטיבציה להמשיך לכתוב קוד ולהתקדם עוד יותר בפרויקט עד לקבלת התוצר הסופי. נדהמתי לראות כיצד ניתן לבנות ולפתח אפליקציה אטרקטיבית ושימושית עבור אנשים מאפס.

האפליקציה אשר מבוססת על אתרים שבן דודי בנה, הפכו את העבודה על הפרויקט ליותר כיפית ומהנה מכיוון שאני התחברתי לקונספט השימושי ביותר של האתרים המאפשר לאנשים לשפר את אוצר המילים שלהם בעברית ובאנגלית ולשפר את היכולות החישוביות שלהם במתמטיקה.

העבודה הייתה מאתגרת מכיוון שזאת פעם ראשונה שלי שיצרתי אפליקציה גדולה באמצעות התוכנה Android Studio. על מנת לעמוד באתגר קראתי רבות באינטרנט מידע, עקרונות, וכלים בסיסיים של התוכנה ובזכות כך ידעתי כיצד להתקדם ומה מטרתי הבאה בהתקדמות. בנוסף לכך למדתי רבות על התוכנה בשנים י"א י"ב שלי ממורי החרוצים והרציניים איתמר ועמליה אשר ליוו אותי לאורכו ועזרו לי להתגבר על קשיים.

אחד מהמכשולים הגדולים שלי הפרויקט הייתה שמירה הנתונים ב firebase. קושי זה נבע מזה שנדרשתי לשמור נתון שהוא רשימה של רשימות מילים אשר כל רשימה מכילה את שמה ומילים בתוכה. שמירת נתון זה המורכב ממחלקה בתוך מחלקה היוותה עבורי קושי. קיבלתי עזרה לפתרון קושי זה מהמורה שלי איתמר, אשר הביא סרטוט מובן שעזר לי לפתור את הבעיה של שמירת הנתון המורכב.

דברים שהייתי שמח מאוד לשפר ולהוסיף לאפליקציה היא תמיכה מלאה במשחקי התרגול של העברית, אנגלית ומתמטיקה, כלומר לא להעביר את המשתמש לאתריו של בן דודי אלא לתת למשתמש לשחק בתוך האפליקציה עצמה במשחקי תרגול המובנים בתוך האפליקציה. הייתי בנוסף שמח להוסיף השמעת מילים בעברית ובאנגלית באפליקציה.

# פרק שמיני - נספחים

## מקורות מידע:

<https://www.w3schools.com/>

<https://stackoverflow.com/>

<https://developer.android.com/>

## קוד:

### Activity

## BackgroundSoundService

```
package com.example.comps.myapplication;

import android.app.Service;
import android.content.Intent;
import android.media.MediaPlayer;
import android.os.IBinder;

/**
 * This service allows to start background music to the user.
 * The music continuous in the background in loops and when the application
 * is closed too.
 */
public class BackgroundSoundService extends Service {

    private MediaPlayer player;

    public IBinder onBind(Intent arg0) {

        return null;
    }

    /**
     * This function creates the media player that plays the music.
     * Works when the user click the button that starts the music.
     */
    @Override
    public void onCreate() {
        super.onCreate();
        player = MediaPlayer.create(this, R.raw.relaxing_music);
        player.setLooping(true);
        player.setVolume(100, 100);
    }

    /**
     * This function starts the music.
     * This function is called when the user clicks the start background

```

```

music button.
    *
    * @param intent
    * @param flags
    * @param startId
    * @return 1, signals that the service started in sticky mode.
    */
    public int onStartCommand(Intent intent, int flags, int startId) {
        player.start();
        return Service.START_STICKY;
    }

    public void onStart(Intent intent, int startId) {

    }

    public IBinder onBind(Intent arg0) {

        return null;
    }

    public void onStop() {

    }

    public void onPause() {

    }

    @Override
    public void onDestroy() {
        player.stop();
    }

    @Override
    public void onLowMemory() {

    }
}

```

## HebrewOrEnglish

```

package com.example.comps.myapplication;

import android.app.Dialog;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.SharedPreferences;
import android.content.res.ColorStateList;
import android.graphics.Color;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.net.Uri;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.TextView;
import android.widget.Toast;

```

```

import java.util.ArrayList;

/**
 * This activity is where the words' practice happens.
 * From here you can reach back to the menu activity or to save or load word
 * list activity.
 * You can add, remove and practice any words you wish in english or hebrew
 * as chosen in the menu.
 */
public class HebrewOrEnglish extends AppCompatActivity {
    private EditText etWord;
    private Button btnMenu, btnAddWord, btnOpenOrLoadWordList, btnPractice;
    private ListView lvWords;
    private String link;
    private ArrayList<String> words;
    private StringBuilder linkBuilder;
    private User user;
    private TextView tvTitle;
    private boolean hebrew;

    /**
     * With this function you can do the following things: get back to menu
     * activity,
     * get to menu activity, add words, delete words and practice words.
     * This function starts when the activity is opened.
     * @param savedInstanceState
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.lo_hebrew_or_english);

        user = (User) getIntent().getSerializableExtra("user");
        words = getIntent().getStringArrayListExtra("words");
        hebrew = getIntent().getBooleanExtra("hebrew?", false);

        if (words == null) {
            words = new ArrayList<>();
        }

        btnMenu = findViewById(R.id.btnMenu);
        btnOpenOrLoadWordList = findViewById(R.id.btnOpenOrLoadWordList);
        btnAddWord = findViewById(R.id.btnAddWord);
        lvWords = findViewById(R.id.lvWords);
        etWord = findViewById(R.id.etWord);
        btnPractice = findViewById(R.id.btnPractice);
        tvTitle = findViewById(R.id.tvTitle);

        if (hebrew) {
            tvTitle.setText("עברית");
        } else {
            tvTitle.setText("אנגלית");
        }

        btnMenu.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent myIntent = new Intent(HebrewOrEnglish.this,
Options.class);
                myIntent.putExtra("user", user);
                startActivity(myIntent);
            }
        });

        btnOpenOrLoadWordList.setOnClickListener(new View.OnClickListener()
{
    @Override

```



```

        public void onClick(View v) {
            if(btnPractice.getCurrentTextColor() == Color.GREEN) {
                Intent myIntent = new Intent(HebrewOrEnglish.this,
SaveOrLoadWordList.class);
                myIntent.putExtra("user", user);
                myIntent.putExtra("words", words);
                myIntent.putExtra("hebrew?", hebrew);
                startActivity(myIntent);
            }
            else
            {
                showMessage("לשמור על לאינטרנט מחובר להיות צריך אתה",1500);
            }
        }
    });

    btnAddWord.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if (etWord.getText().toString().trim().length() > 0 &&
!words.contains(etWord.getText().toString())) {
                String word = etWord.getText().toString();
                words.add(word);
                etWord.getText().clear();
                refreshListView();
            }
        }
    });

    lvWords.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
            showMessage(words.get(position) + " נמחק", 750);
            words.remove(position);
            refreshListView();
        }
    });

    btnPractice.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            if(btnPractice.getCurrentTextColor() == Color.GREEN) {
                generateLink();
                link = linkBuilder.toString();
                showMessage("לנכנס!",1500);
                openBrowser();
            }
            else
            {
                showMessage("מנת על לאינטרנט מחובר להיות צריך אתה",1500);
            }
        }
    });
    checkWifi();
    refreshListView();
}

/**
 * This function refreshes the listview.
 * This function starts when the user adds and removes words from the
list of words.
 */
public void refreshListView() {

```

```

        WordListAdapter arrayAdapter = new WordListAdapter(this,
R.layout.custom_row, words);
        lvWords.setAdapter(arrayAdapter);
    }

    /**
     * This function builds the menu
     * This function opens when this activity is reached
     */
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu1, menu);
        return super.onCreateOptionsMenu(menu);
    }

    /**
     * This function allows to select items in the menu.
     * This function opens when the user hits the item(button).
     */
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch(item.getItemId())
        {
            case R.id.itLogout:
                Intent myIntent = new Intent(HebrewOrEnglish.this,
LoginScreen.class);
                SharedPreferences preferences =
getSharedPreferences("checkbox",MODE_PRIVATE);
                SharedPreferences.Editor editor = preferences.edit();
                editor.putString("remember","no");
                editor.apply();
                startActivity(myIntent);
                break;
            case R.id.itUserGuide:
                Dialog d = new Dialog(HebrewOrEnglish.this);
                d.setContentview(R.layout.d_user_guide);
                d.show();
                break;
        }
        return super.onOptionsItemSelected(item);
    }

    /**
     * This function helps to open a link in a browser.
     * It called every time the user clicks to practice.
     */
    private void openBrowser() {
        Intent intent = new Intent();
        intent.setAction(Intent.ACTION_VIEW);
        intent.addCategory(Intent.CATEGORY_BROWSABLE);
        intent.setData(Uri.parse(link));
        startActivity(intent);
    }

    /**
     * This function shows a toast message.
     * It is called every time the programmer wants to put a toast in the
screen.
     *
     * @param text is the text of the toast.
     * @param duration is the time duration of the toast.
     */
    public void showToastMessage(String text, int duration) {
        final Toast toast = Toast.makeText(HebrewOrEnglish.this, text,
Toast.LENGTH_SHORT);
        toast.show();
        Handler handler = new Handler();
        handler.postDelayed(new Runnable() {

```

```

        @Override
        public void run() {
            toast.cancel();
        }
    }, duration);
}

/**
 * This function constantly listens to a wifi connection.
 * It is called every time this activity opens (HebrewOrEnglish).
 * The user needs to be connected to the internet to practice words.
 */
public void checkWifi() {
    BroadcastReceiver receiver = new BroadcastReceiver() {
        @Override public void onReceive(Context context, Intent _ )
        {
            ConnectivityManager connMgr = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
            NetworkInfo wifi =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_WIFI);
            NetworkInfo mobile =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_MOBILE);

            boolean isConnected = wifi != null &&
wifi.isConnectedOrConnecting() ||
                mobile != null && mobile.isConnectedOrConnecting();
            if (isConnected) {
                showMessage("לאינטרנט מחובר!",1500);
                btnPractice.setTextColor(Color.GREEN);
                btnOpenOrLoadWordList.setTextColor(Color.GREEN);
            } else {
                showMessage("לאינטרנט חסר",1500);
                btnPractice.setTextColor(Color.RED);
                btnOpenOrLoadWordList.setTextColor(Color.RED);
            }
        }
    };
    this.registerReceiver( receiver, new
IntentFilter("android.net.conn.CONNECTIVITY_CHANGE") );
}

/**
 * This function generate the link in the site with the words the user
chose to practice.
 * It called every time the user clicks to practice.
 */
public void generateLink() {
    if (hebrew) {
        linkBuilder = new
StringBuilder("https://www.hachtava.co.il/index.html?");
    } else {
        linkBuilder = new
StringBuilder("https://www.hachtava.co.il/english.html?");
    }
    for (String word : words) {
        linkBuilder.append("&");
        for (int i = 0; i < word.length(); i++) {
            String character = Character.toString(word.charAt(i));
            switch (character) {
                case "א":
                    The rest is censored
            }
        }
    }
}
}

```

# ListOfWordList

```
package com.example.comps.myapplication;

import java.io.Serializable;
import java.util.ArrayList;

/**
 * This class is for holding a list that holds word lists.
 * This class have getters, setters and constructors.
 */

public class ListOfWordList implements Serializable {
    private ArrayList<WordList> listOfWordList = new ArrayList<>();

    ListOfWordList() {
    }

    public ArrayList<WordList> getListOfWordList() {
        return listOfWordList;
    }

    public void setListOfWordList(ArrayList<WordList> listOfWordList) {
        this.listOfWordList = listOfWordList;
    }
}
```

# ListOfWordListAdapter

```
package com.example.comps.myapplication;

import android.content.Context;
import android.graphics.Color;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.TextView;

import java.util.List;

/**
 * This is a custom adapter that changes the strings in a listview to blue.
 */

public class ListOfWordListAdapter extends ArrayAdapter<WordList> {

    private Context mContext;
    private int id;
    private List<WordList> items;

    /**
     * Sets the custom adapter.
     * Works when the adapter is defined.
     *
     * @param context
     * @param textViewResourceId
     * @param list
     */
    public ListOfWordListAdapter(Context context, int textViewResourceId,
```

```

List<WordList> list) {
    super(context, textViewResourceId, list);
    mContext = context;
    id = textViewResourceId;
    items = list;
}

/**
 * Sets the blue color.
 * Works when the adapter is defined.
 *
 * @param position
 * @param v
 * @param parent
 * @return
 */
@Override
public View getView(int position, View v, ViewGroup parent) {
    View mView = v;
    if (mView == null) {
        LayoutInflater vi = (LayoutInflater)
mContext.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        mView = vi.inflate(id, null);
    }

    TextView text = (TextView) mView.findViewById(R.id.textView);

    if (items.get(position) != null) {
        text.setTextColor(Color.BLUE);
        text.setText(items.get(position).getListName());
    }

    return mView;
}
}

```

## LoginScreen

```

package com.example.comps.myapplication;

import android.app.Dialog;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.CountDownTimer;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.CompoundButton;
import android.widget.EditText;
import android.widget.Toast;

import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;

```

```

import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.Query;
import com.google.firebase.database.ValueEventListener;

import java.util.Timer;
import java.util.TimerTask;

/**
 * This activity is the login screen for the user where he can log-in with
 * his user and password or register.
 * When you log-in to the account the activity reads from the database if
 * the data that was typed is correct.
 * When you register the activity writes the new user's data in the
 * firebase.
 */
public class LoginScreen extends AppCompatActivity {
    private EditText etUserName, etPassword, etDUserName, etDPassword;
    private Button btnRegister, btnLogin, btnDRegister;
    private CheckBox cbRemember;
    private FirebaseDatabase database = FirebaseDatabase.getInstance();
    private DatabaseReference ref = database.getReference();
    private SharedPreferences preferences;

    /**
     * With this function you can do the following things: login and
     * register an account.
     * This function starts when the activity is opened.
     * @param savedInstanceState
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.lo_login_screen);
        btnLogin = findViewById(R.id.btn_login);
        btnRegister = findViewById(R.id.btn_register);
        etUserName = findViewById(R.id.et_username);
        etPassword = findViewById(R.id.et_password);
        cbRemember = findViewById(R.id.cb_remember);

        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (btnLogin.getCurrentTextColor() == Color.GREEN) {
                    Query q = ref.child("Users").orderByValue();
                    q.addListenerForSingleValueEvent(new
ValueEventListener() {
                        @Override
                        public void onDataChange(DataSnapshot dataSnapshot) {
                            {
                                if (etUserName.getText().toString().length() !=
0 && etPassword.getText().toString().length() != 0) {
                                    for (DataSnapshot dst :
dataSnapshot.getChildren()) {
                                        User u = dst.getValue(User.class);
                                        if
(u.getUserName().equals(etUserName.getText().toString()) &&
u.getPassword().equals(etPassword.getText().toString())) {
                                            if (cbRemember.isChecked())
                                            {
                                                SharedPreferences preferences =
getSharedPreferences("checkbox", MODE_PRIVATE);
                                                SharedPreferences.Editor editor
= preferences.edit();
                                                editor.putString("remember", "yes");
                                                editor.putString("userName", etUserName.getText().toString());

```

```

editor.putString("password", etPassword.getText().toString());
        editor.apply();
    }
    else
    {
        SharedPreferences preferences =
getSharedPreferences("checkbox", MODE_PRIVATE);
        SharedPreferences.Editor editor
= preferences.edit();
        editor.putString("remember", "no");
        editor.apply();
    }
    Intent myIntent = new
Intent(LoginScreen.this, Options.class);
    myIntent.putExtra("user", u);
    showToastMessage("נכנס!", 3000);
    startActivity(myIntent);
    return;
}
}
showToastMessage("לא סיסמה או משתמש שם",
"נכנסים", 1000);
        etUserName.getText().clear();
        etPassword.getText().clear();
    }
}

@Override
public void onCancelled(DatabaseError databaseError)
{
}
});
}
else
{
    showToastMessage("מנת על לאינטרנט מחובר להיות צריך אתה",
להתחבר", 1500);
}
}
});

btnRegister.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
    if (btnLogin.getCurrentTextColor() == Color.GREEN) {
        Dialog d = new Dialog(LoginScreen.this);
        d setContentView(R.layout.d_register);
        d.show();
        etDUserName = d.findViewById(R.id.etUserName);
        etDPassword = d.findViewById(R.id.etPassword);
        btnDRegister = d.findViewById(R.id.btnRegister);
        btnDRegister.setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View v) {
            if (etDUserName.getText().toString().length() !=
0 && etDPassword.getText().toString().length() != 0) {
                if
(etDPassword.getText().toString().length() < 8) {
                    showToastMessage("להכיל חייבת הסיסמה",
"תווים 8 לפחות", 1000);
                    return;
                }
                Query q = ref.child("Users").orderByValue();
                q.addListenerForSingleValueEvent(new

```





```

    * This function shows a toast message.
    * It is called every time the programmer wants to put a toast in the
screen.
    *
    * @param text      is the text of the toast.
    * @param duration is the time duration of the toast.
    */
    public void showToastMessage(String text, int duration) {
        final Toast toast = Toast.makeText(LoginScreen.this, text,
Toast.LENGTH_SHORT);
        toast.show();
        Handler handler = new Handler();
        handler.postDelayed(new Runnable() {
            @Override
            public void run() {
                toast.cancel();
            }
        }, duration);
    }
}
/**
 * This function constantly listens to a wifi connection.
 * It is called every time this activity opens (HebrewOrEnglish).
 * The user needs to be connected to the internet to practice words.
 */
public void checkWifi() {
    BroadcastReceiver receiver = new BroadcastReceiver() {
        @Override public void onReceive(Context context, Intent _ )
        {
            ConnectivityManager connMgr = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
            NetworkInfo wifi =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_WIFI);
            NetworkInfo mobile =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_MOBILE);

            boolean isConnected = wifi != null &&
wifi.isConnectedOrConnecting() ||
mobile != null && mobile.isConnectedOrConnecting();
            if (isConnected) {
                showToastMessage("לאינטרנט מחובר",1500);
                btnLogin.setTextColor(Color.GREEN);
                btnRegister.setTextColor(Color.GREEN);
            } else {
                showToastMessage("לאינטרנט חסר",1500);
                btnLogin.setTextColor(Color.RED);
                btnRegister.setTextColor(Color.RED);
            }
        }
    };
    this.registerReceiver( receiver, new
IntentFilter("android.net.conn.CONNECTIVITY_CHANGE") );
}
/**
 * This function builds the menu
 * This function opens when this activity is reached
 */
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu2, menu);
    return super.onCreateOptionsMenu(menu);
}

/**
 * This function allows to select items in the menu.
 * This function opens when the user hits the item(button).
 */
@Override

```

```

        public boolean onOptionsItemSelected(MenuItem item) {
            switch(item.getItemId())
            {
                case R.id.itLogout:
                    Intent myIntent = new Intent(LoginScreen.this,
LoginScreen.class);
                    SharedPreferences preferences =
getSharedPreferences("checkbox",MODE_PRIVATE);
                    SharedPreferences.Editor editor = preferences.edit();
                    editor.putString("remember","no");
                    editor.apply();
                    startActivity(myIntent);
                    break;
                case R.id.itUserGuide:
                    Dialog d = new Dialog(LoginScreen.this);
                    d setContentView(R.layout.d_user_guide);
                    d.show();
                    break;
            }
            return super.onOptionsItemSelected(item);
        }
    }
}

```

## Math

```

package com.example.comps.myapplication;

import android.app.Dialog;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.net.Uri;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

/**
 * This activity allows the user to play math games that he chooses and go
back to the menu.
 */
public class Math extends AppCompatActivity {
    private User user;
    private Button btnMenu, btnAddGames, btnSubGames, btnMulGames,
btnDivGames, btnGame1, btnGame2, btnGame3, btnGame4, btnGame5, btnGame6,
btnGame7;

    /**
     * With this function you can do the following things: go back to the
menu and choose a game to play.
     * This function starts when the activity is opened.
     *
     * @param savedInstanceState
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}

```

```

        setContentView(R.layout.lo_math);

        user = (User) getIntent().getSerializableExtra("user");

        btnMenu = findViewById(R.id.btnMenu);
        btnAddGames = findViewById(R.id.btnAddGames);
        btnSubGames = findViewById(R.id.btnSubGames);
        btnMulGames = findViewById(R.id.btnMulGames);
        btnDivGames = findViewById(R.id.btnDivGames);

        btnMenu.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent myIntent = new Intent(Math.this, Options.class);
                myIntent.putExtra("user", user);
                startActivity(myIntent);
            }
        });
        btnAddGames.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openGamesDialog(1);
            }
        });
        btnSubGames.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openGamesDialog(2);
            }
        });
        btnMulGames.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openGamesDialog(3);
            }
        });
        btnDivGames.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openGamesDialog(4);
            }
        });
    }

    /**
     * This function opens the game list dialog.
     * Its called every time the user clicks on certain math game type.
     *
     * @param gameType is the math game type(1 - addGames, 2 - subGames, 3 -
     * mulGames, 4 - divGames)
     */
    private void openGamesDialog(final int gameType)
    {
        Dialog d = new Dialog(Math.this);
        d.setContentView(R.layout.d_math_games);
        d.show();
        btnGame1 = d.findViewById(R.id.btnGame1);
        btnGame1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                switch (gameType)
                {
                    case 1:
                        openGame(btnGame1, "https://www.free-training-tutorial.com/addition/towncreator/tc-addition.html");
                        break;
                    case 2:
                        openGame(btnGame1, "https://www.free-training-tutorial.com/subtraction/towncreator/tc-subtraction.html");

```

```

        break;
    case 3:
        openGame(btnGame1, "https://www.free-training-
tutorial.com/times-tables/towncreator/tc-multiplication.html");
        break;
    case 4:
        openGame(btnGame1, "https://www.free-training-
tutorial.com/division/towncreator/tc-division.html");
        break;
    }
}
});
btnGame2 = d.findViewById(R.id.btnGame2);
btnGame2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        switch (gameType)
        {
            case 1:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/addition/sealife/sl-addition.html");
                break;
            case 2:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/subtraction/sealife/sl-subtraction.html");
                break;
            case 3:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/times-tables/sealife/sl-multiplication.html");
                break;
            case 4:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/division/sealife/sl-division.html");
                break;
        }
    }
});
btnGame3 = d.findViewById(R.id.btnGame3);
btnGame3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        switch (gameType)
        {
            case 1:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/addition/addition-sharks.html");
                break;
            case 2:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/subtraction-math-basketball.html");
                break;
            case 3:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/times-tables/multiplication-airplanes.html");
                break;
            case 4:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/division-demolition.html");
                break;
        }
    }
});
btnGame4 = d.findViewById(R.id.btnGame4);
btnGame4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        switch (gameType)
        {

```

```

        case 1:
            openGame(btnGame1, "https://www.free-training-
tutorial.com/addition/central-park.html");
            break;
        case 2:
            openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/subtraction-mission.html");
            break;
        case 3:
            openGame(btnGame1, "https://www.free-training-
tutorial.com/times-tables/multiplication-hot-air-balloons.html");
            break;
        case 4:
            openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/division-croc-doc.html");
            break;
    }
}

});
btnGame5 = d.findViewById(R.id.btnGame5);
btnGame5.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        switch (gameType)
        {
            case 1:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/addition/addition-dragons.html");
                break;
            case 2:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/subtraction-pearl-search.html");
                break;
            case 3:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/times-tables/apollo-moon-landers.html");
                break;
            case 4:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/division-fruit-shoot.html");
                break;
        }
    }
});
btnGame6 = d.findViewById(R.id.btnGame6);
btnGame6.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        switch (gameType)
        {
            case 1:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/addition/addition-washington-monument.html");
                break;
            case 2:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/subtraction-blast.html");
                break;
            case 3:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/multiplication-meteor.html");
                break;
            case 4:
                openGame(btnGame1, "https://www.free-training-
tutorial.com/math-games/division-number-invaders.html");
                break;
        }
    }
});
}

```



```

    });
    btnGame7 = d.findViewById(R.id.btnGame7);
    btnGame7.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            switch (gameType)
            {
                case 1:
                    openGame(btnGame1,"https://www.free-training-
tutorial.com/addition/addition-empire-state.html");
                    break;
                case 2:
                    openGame(btnGame1,"https://www.free-training-
tutorial.com/math-games/subtraction-matching.html");
                    break;
                case 3:
                    openGame(btnGame1,"https://www.free-training-
tutorial.com/math-games/times-tables-shooting.html");
                    break;
                case 4:
                    openGame(btnGame1,"https://www.free-training-
tutorial.com/math-games/division-matching.html");
                    break;
            }
        }
    });
    checkWifi();
}
/**
 * This function opens a math game.
 * Its called every time the user clicks on certain math game.
 *
 * @param btnGame is the button of the game.
 */
private void openGame(Button btnGame, String url)
{
    if (btnGame.getCurrentTextColor() == Color.GREEN) {
        showToastMessage("ונכנס!", 1500);
        openBrowser(url);
    } else {
        showToastMessage("לתרגל מנת על לאינטרנט מחובר להיות צריך אתה", 1500);
    }
}
/**
 * This function helps to open a link in a browser.
 * Its called every time the user clicks to practice.
 */
private void openBrowser(String url) {
    Intent intent = new Intent();
    intent.setAction(Intent.ACTION_VIEW);
    intent.addCategory(Intent.CATEGORY_BROWSABLE);
    intent.setData(Uri.parse(url));
    startActivity(intent);
}
/**
 * This function builds the menu
 * This function opens when this activity is reached
 */
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu1, menu);
    return super.onCreateOptionsMenu(menu);
}
/**
 * This function allows to select items in the menu.
 * This function opens when the user hits the item(button).

```

```

    */
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch(item.getItemId())
        {
            case R.id.itLogout:
                Intent myIntent = new Intent(Math.this, LoginScreen.class);
                SharedPreferences preferences =
getSharedPreferences("checkbox",MODE_PRIVATE);
                SharedPreferences.Editor editor = preferences.edit();
                editor.putString("remember","no");
                editor.apply();
                startActivity(myIntent);
                break;
            case R.id.itUserGuide:
                Dialog d = new Dialog(Math.this);
                d setContentView(R.layout.d_user_guide);
                d.show();
                break;
        }
        return super.onOptionsItemSelected(item);
    }

    /**
     * This function shows a toast message.
     * It is called every time the programmer wants to put a toast in the
screen.
     *
     * @param text is the text of the toast.
     * @param duration is the time duration of the toast.
     */
    public void showToastMessage(String text, int duration) {
        final Toast toast = Toast.makeText(Math.this, text,
Toast.LENGTH_SHORT);
        toast.show();
        Handler handler = new Handler();
        handler.postDelayed(new Runnable() {
            @Override
            public void run() {
                toast.cancel();
            }
        }, duration);
    }

    /**
     * This function constantly listens to a wifi connection.
     * It is called every time this activity opens (HebrewOrEnglish).
     * The user needs to be connected to the internet to practice words.
     */
    public void checkWifi() {
        BroadcastReceiver receiver = new BroadcastReceiver() {
            @Override
            public void onReceive(Context context, Intent _) {
                ConnectivityManager connMgr = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
                NetworkInfo wifi =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_WIFI);
                NetworkInfo mobile =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_MOBILE);

                boolean isConnected = wifi != null &&
wifi.isConnectedOrConnecting() ||
                mobile != null && mobile.isConnectedOrConnecting();
                if (isConnected) {
                    showToastMessage("לאינטרנט מחובר!", 1500);
                    btnGame1.setTextColor(Color.GREEN);
                    btnGame2.setTextColor(Color.GREEN);
                    btnGame3.setTextColor(Color.GREEN);
                }
            }
        };
    }

```

```

        btnGame4.setTextColor(Color.GREEN);
        btnGame5.setTextColor(Color.GREEN);
        btnGame6.setTextColor(Color.GREEN);
        btnGame7.setTextColor(Color.GREEN);
    } else {
        showMessage("לאינטרנט חיבור אין", 1500);
        btnGame1.setTextColor(Color.RED);
        btnGame2.setTextColor(Color.RED);
        btnGame3.setTextColor(Color.RED);
        btnGame4.setTextColor(Color.RED);
        btnGame5.setTextColor(Color.RED);
        btnGame6.setTextColor(Color.RED);
        btnGame7.setTextColor(Color.RED);
    }
}

};
this.registerReceiver(receiver, new
IntentFilter("android.net.conn.CONNECTIVITY_CHANGE"));
}
}

```

## Options

```

package com.example.comps.myapplication;

import android.app.Dialog;
import android.content.Intent;
import android.content.SharedPreferences;
import android.media.MediaPlayer;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

/**
 * This activity gives the user the option to choose if he would like to
 * practice hebrew or english words or to play math games.
 */
public class Options extends AppCompatActivity {
    private TextView tvHelloUser;
    private Button btnMath, btnEnglish, btnHebrew, btnMusic, btnStopMusic;
    private User user;
    private boolean hebrew;
    private MediaPlayer player;
    /**
     * With this function you can do the following things: You can reach to
     * math, english, hebrew activities and play music.
     * This function starts when the activity is opened.
     * @param savedInstanceState
     */
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.lo_options);

        user = (User) getIntent().getSerializableExtra("user");

        tvHelloUser = findViewById(R.id.tvHelloUser);
        btnMath = findViewById(R.id.btnMath);
        btnEnglish = findViewById(R.id.btnEnglish);
        btnHebrew = findViewById(R.id.btnHebrew);
        btnMusic = findViewById(R.id.btnMusic);
        btnStopMusic = findViewById(R.id.btnStopMusic);
    }
}

```

```

        btnMath.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent myIntent = new Intent(Options.this, Math.class);
                myIntent.putExtra("user", user);
                startActivity(myIntent);
            }
        });

        btnEnglish.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent myIntent = new Intent(Options.this,
HebrewOrEnglish.class);
                myIntent.putExtra("user", user);
                hebrew = false;
                myIntent.putExtra("hebrew?", hebrew);
                startActivity(myIntent);
            }
        });

        btnHebrew.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent myIntent = new Intent(Options.this,
HebrewOrEnglish.class);
                myIntent.putExtra("user", user);
                hebrew = true;
                myIntent.putExtra("hebrew?", hebrew);
                startActivity(myIntent);
            }
        });

        btnMusic.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent svc = new Intent(Options.this,
BackgroundSoundService.class);
                startService(svc);
            }
        });

        btnStopMusic.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent svc = new Intent(Options.this,
BackgroundSoundService.class);
                stopService(svc);
            }
        });

        tvHelloUser.setText("היי " + user.getUserName());
    }

    /**
     * This function builds the menu
     * This function opens when this activity is reached
     */
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu1, menu);
        return super.onCreateOptionsMenu(menu);
    }

    /**
     * This function allows to select items in the menu.
     * This function opens when the user hits the item(button).
     */
    @Override

```

```

        public boolean onOptionsItemSelected(MenuItem item) {
            switch(item.getItemId())
            {
                case R.id.itLogout:
                    Intent myIntent = new Intent(Options.this,
LoginScreen.class);
                    SharedPreferences preferences =
getSharedPreferences("checkbox",MODE_PRIVATE);
                    SharedPreferences.Editor editor = preferences.edit();
                    editor.putString("remember","no");
                    editor.apply();
                    startActivity(myIntent);
                    break;
                case R.id.itUserGuide:
                    Dialog d = new Dialog(Options.this);
                    d setContentView(R.layout.d_user_guide);
                    d.show();
                    break;
            }
            return super.onOptionsItemSelected(item);
        }
    }
}

```

## SaveOrLoadWordList

```

package com.example.comps.myapplication;

import android.app.Dialog;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.SharedPreferences;
import android.graphics.Color;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Toast;

import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

import java.util.ArrayList;
/**
 * This activity allows the user to save or load word lists of his own in or
from his data base.
 * The user can delete words from the list, remove word lists and go back to
HebrewOrEnglish activity
 */
public class SaveOrLoadWordList extends AppCompatActivity {
    private FirebaseDatabase database = FirebaseDatabase.getInstance();
    private DatabaseReference ref = database.getReference();
    private ArrayList<String> words = new ArrayList<>();
    private ListOfWordList hebrewListofWordList;
    private ListOfWordList englishListofWordList;
    private ListView lvWords, lvWordLists;
    private Button btnAddWordList, btnBack, btnRemove;
    private EditText etWordListName;

```



```

private User user;
private int pos = -1;
private boolean hebrew;
/**
 * With this function you can do the following things: you can go back
to the menu,
 * save/add word lists, remove word lists, remove words from word list
and load word lists.
 * This function starts when the activity is opened.
 * @param savedInstanceState
 */
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.lo_save_or_load_word_list);

    lvWords = findViewById(R.id.lvWords);
    lvWordLists = findViewById(R.id.lvWordLists);
    etWordListName = findViewById(R.id.etWordListName);
    btnAddWordList = findViewById(R.id.btnAddWordList);
    btnBack = findViewById(R.id.btnBack);
    btnRemove = findViewById(R.id.btnRemove);
    user = (User) getIntent().getSerializableExtra("user");
    words = getIntent().getStringArrayListExtra("words");
    hebrew = getIntent().getBooleanExtra("hebrew?", false);

    hebrewListOfWordList = user.getHebrewListOfWordList();
    englishListOfWordList = user.getEnglishListOfWordList();

    btnAddWordList.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String name = etWordListName.getText().toString();
            WordList wordList = new WordList(words, name);
            if (etWordListName.getText().toString().trim().length() > 0)
            {
                if (hebrew) {
                    for (WordList list :
hebrewListOfWordList.getListOfWordList()) //bad code
                    {
                        if (list.getListName().equals(name)) {
hebrewListOfWordList.getListOfWordList().remove(list);
                            break;
                        }
                    }
                }
                hebrewListOfWordList.getListOfWordList().add(wordList);
                pos =
hebrewListOfWordList.getListOfWordList().size() - 1;
            } else {
                for (WordList list :
englishListOfWordList.getListOfWordList()) //bad code
                {
                    if (list.getListName().equals(name)) {
englishListOfWordList.getListOfWordList().remove(list);
                        break;
                    }
                }
                englishListOfWordList.getListOfWordList().add(wordList);
                pos =
englishListOfWordList.getListOfWordList().size() - 1;
            }
            showToastMessage("אמת יצרת: " + name, 1250);
        }
    });
}

```

```

        }
        etWordListName.getText().clear();
        saveData();
        refreshWordLists();
        refreshListView();
    }
});

btnBack.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent myIntent = new Intent(SaveOrLoadWordList.this,
HebrewOrEnglish.class);
        myIntent.putExtra("user", user);
        myIntent.putExtra("words", words);
        myIntent.putExtra("hebrew?", hebrew);
        startActivity(myIntent);
    }
});

btnRemove.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (pos != -1) {
            words = new ArrayList<>();
            if (hebrew) {
                showToastMessage("את מחקת: " +
hebrewListOfWordList.getListOfWordList().remove(pos).getListName(), 1250);
            } else {
                showToastMessage("את מחקת: " +
englishListOfWordList.getListOfWordList().remove(pos).getListName(), 1250);
            }
            pos = -1;
            saveData();
            refreshListView();
            refreshWordLists();
        }
    }
});

lvWords.setOnItemClickListener(new AdapterView.OnItemClickListener()
{
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
        showToastMessage(words.get(position) + " נחקק", 750);
        ArrayList<String> wordsCopy = new ArrayList<>(words);
        words = wordsCopy;
        words.remove(position);
        refreshListView();
    }
});

lvWordLists.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
        if (pos != position)
        {
            pos = position;
            if (hebrew) {
                words =
hebrewListOfWordList.getListOfWordList().get(position).getWords();
                showToastMessage("ב בחרת: " +
hebrewListOfWordList.getListOfWordList().get(position).getListName(), 1250);
            }
        }
    }
});

```

```

        } else {
            words =
englishListOfWordList.getListOfWordList().get(position).getWords();
            showToastMessage("ב בחירת: " +
englishListOfWordList.getListOfWordList().get(position).getListName(),
1250);
        }
        refreshListView();
    }
}

});
refreshListView();
refreshWordLists();
checkWifi();
}

/**
 * This function refreshes the listview of words.
 * This function starts when the user adds,removes and selects words
from the list of words.
 */
void refreshListView() {
    WordListAdapter arrayAdapter = new WordListAdapter(this,
R.layout.custom_row, words);
    lvWords.setAdapter(arrayAdapter);
}

/**
 * This function refreshes the listview of word lists.
 * This function starts when the user adds and removes word lists from
the list of words.
 */
void refreshWordLists() {
    ListOfWordListAdapter arrayAdapter;
    if (hebrew) {
        arrayAdapter = new ListOfWordListAdapter(this,
R.layout.custom_row, hebrewListOfWordList.getListOfWordList());
    } else {
        arrayAdapter = new ListOfWordListAdapter(this,
R.layout.custom_row, englishListOfWordList.getListOfWordList());
    }
    lvWordLists.setAdapter(arrayAdapter);
}

/**
 * This function shows a toast message.
 * It is called every time the programmer wants to put a toast in the
screen.
 *
 * @param text is the text of the toast.
 * @param duration is the time duration of the toast.
 */
public void showToastMessage(String text, int duration) {
    final Toast toast = Toast.makeText(SaveOrLoadWordList.this, text,
Toast.LENGTH_SHORT);
    toast.show();
    Handler handler = new Handler();
    handler.postDelayed(new Runnable() {
        @Override
        public void run() {
            toast.cancel();
        }
    }, duration);
}

/**
 * This function saves the hebrew and english word lists in the
firebase.

```

```

    * This function starts when the user adds and removes word lists from
    the list of words.
    */
    void saveData() {
        user.setHebrewListOfWordList(hebrewListOfWordList);
        ref.child("Users").child(user.getUserName()).setValue(user);
        user.setEnglishListOfWordList(englishListOfWordList);
        ref.child("Users").child(user.getUserName()).setValue(user);
    }
    public void checkWifi() {
        BroadcastReceiver receiver = new BroadcastReceiver() {
            @Override public void onReceive(Context context, Intent _ )
            {
                ConnectivityManager connMgr = (ConnectivityManager)
context.getSystemService(Context.CONNECTIVITY_SERVICE);
                NetworkInfo wifi =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_WIFI);
                NetworkInfo mobile =
connMgr.getNetworkInfo(ConnectivityManager.TYPE_MOBILE);

                boolean isConnected = wifi != null &&
wifi.isConnectedOrConnecting() ||
                mobile != null && mobile.isConnectedOrConnecting();
                if (!isConnected) {
                    showToastMessage("לשמור על לאינטרנט מחובר להיות צריך אתה"
לשמור על לאינטרנט מחובר להיות צריך אתה", 3000);
                    Intent myIntent = new Intent(SaveOrLoadWordList.this,
HebrewOrEnglish.class);
                    myIntent.putExtra("user", user);
                    myIntent.putExtra("words", words);
                    myIntent.putExtra("hebrew?", hebrew);
                    startActivity(myIntent);
                }
            }
        };
        this.registerReceiver( receiver, new
IntentFilter("android.net.conn.CONNECTIVITY_CHANGE" ) );
    }
    /**
     * This function builds the menu
     * This function opens when this activity is reached
     */
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu1, menu);
        return super.onCreateOptionsMenu(menu);
    }

    /**
     * This function allows to select items in the menu.
     * This function opens when the user hits the item(button).
     */
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch(item.getItemId())
        {
            case R.id.itLogout:
                Intent myIntent = new Intent(SaveOrLoadWordList.this,
LoginScreen.class);
                SharedPreferences preferences =
getSharedPreferences("checkbox", MODE_PRIVATE);
                SharedPreferences.Editor editor = preferences.edit();
                editor.putString("remember", "no");
                editor.apply();
                startActivity(myIntent);
                break;
            case R.id.itUserGuide:
                Dialog d = new Dialog(SaveOrLoadWordList.this);

```

```

        d.setContentView(R.layout.d_user_guide);
        d.show();
        break;
    }
    return super.onOptionsItemSelected(item);
}
}

```

## User

```

package com.example.comps.myapplication;

import java.io.Serializable;
import java.util.List;

/**
 * This class holds the user who has his own username, password, hebrew and
 * english word lists.
 * This class have getters, setters and constructors.
 */

public class User implements Serializable {
    private String userName;
    private String password;
    private ListOfWordList hebrewListOfWordList = new ListOfWordList();
    private ListOfWordList englishListOfWordList = new ListOfWordList();

    public User() {
    }

    public User(String userName, String password) {
        this.userName = userName;
        this.password = password;
    }

    public ListOfWordList getEnglishListOfWordList() {
        return englishListOfWordList;
    }

    public void setEnglishListOfWordList(ListOfWordList
englishListOfWordList) {
        this.englishListOfWordList = englishListOfWordList;
    }

    public ListOfWordList getHebrewListOfWordList() {
        return hebrewListOfWordList;
    }

    public void setHebrewListOfWordList(ListOfWordList hebrewListOfWordList)
{
        this.hebrewListOfWordList = hebrewListOfWordList;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getUsername() {
        return userName;
    }

    public void setUsername(String userName) {

```

```
        this.userName = userName;
    }
}
```

## WordList

```
package com.example.comps.myapplication;

import java.io.Serializable;
import java.util.ArrayList;

/**
 * This class is for holding a list of words.
 * This class have getters, setters and constructors.
 */

public class WordList implements Serializable {
    private ArrayList<String> words = new ArrayList<>();
    private String listName;

    WordList() {
    }

    WordList(ArrayList<String> words, String listName) {
        this.words = words;
        this.listName = listName;
    }

    @Override
    public String toString() {
        return listName;
    }

    public ArrayList<String> getWords() {
        return words;
    }

    public void setWords(ArrayList<String> words) {
        this.words = words;
    }

    public String getListName() {
        return listName;
    }

    public void setListName(String listName) {
        this.listName = listName;
    }
}
```

## WordListAdapter

```
package com.example.comps.myapplication;

import android.content.Context;
import android.graphics.Color;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.TextView;

import java.util.ArrayList;
import java.util.List;

/**
```



```

* This is a custom adapter that changes the strings in a listview to black.
*/

public class WordListAdapter extends ArrayAdapter<String> {

    private Context mContext;
    private int id;
    private List<String> items;

    /**
     * Sets the custom adapter.
     * Works when the adapter is defined.
     *
     * @param context
     * @param textViewResourceId
     * @param list
     */
    public WordListAdapter(Context context, int textViewResourceId,
List<String> list) {
        super(context, textViewResourceId, list);
        mContext = context;
        id = textViewResourceId;
        items = list;
    }

    /**
     * Sets the black color.
     * Works when the adapter is defined.
     *
     * @param position
     * @param v
     * @param parent
     * @return
     */
    @Override
    public View getView(int position, View v, ViewGroup parent) {
        View mView = v;
        if (mView == null) {
            LayoutInflater vi = (LayoutInflater)
mContext.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
            mView = vi.inflate(id, null);
        }

        TextView text = (TextView) mView.findViewById(R.id.textView);

        if (items.get(position) != null) {
            text.setTextColor(Color.BLACK);
            text.setText(items.get(position));
        }

        return mView;
    }
}

```

# Manifest

## AndroidManifest.xml

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

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".LoginScreen">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".Options" />
        <activity android:name=".Math" />
        <service android:enabled="true"
android:name=".BackgroundSoundService" />
        <activity android:name=".HebrewOrEnglish" />
        <activity android:name=".SaveOrLoadWordList"></activity>
    </application>

</manifest>
```

# XML

## custom\_row.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:id="@+id/textView"
        android:textStyle="bold"
        android:textSize="25sp" android:paddingTop="10dip"
android:paddingBottom="10dip"/>
</LinearLayout>
```

## d\_math\_games.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="@drawable/login"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/tvNote"
        android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:background="#8054cce4"
        android:text="חשוק בחור"
        android:textSize="30sp"
        android:textStyle="bold" />

<Button
    android:id="@+id/btnGame1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="קטן 1"
    android:textSize="18sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnGame2"
    android:textStyle="bold"
    android:backgroundTint="#E61379C2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:text="קטן 2" />

<Button
    android:id="@+id/btnGame3"
    android:layout_width="match_parent"
    android:backgroundTint="#E61379C2"
    android:textStyle="bold"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:text="קטן 3" />

<Button
    android:id="@+id/btnGame4"
    android:textStyle="bold"
    android:backgroundTint="#E61379C2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:text="קטן 4" />

<Button
    android:id="@+id/btnGame5"
    android:textStyle="bold"
    android:layout_width="match_parent"
    android:backgroundTint="#E61379C2"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:text="קטן 5" />

<Button
    android:id="@+id/btnGame6"
    android:layout_width="match_parent"
    android:textStyle="bold"
    android:backgroundTint="#E61379C2"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:text="קטן 6" />

<Button
    android:id="@+id/btnGame7"
    android:textStyle="bold"
    android:backgroundTint="#E61379C2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="18sp"

```

```
        android:text="קטן 7" />
</LinearLayout>
```

## d\_register.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/login"
    android:orientation="vertical"
    tools:context="com.example.comps.myapplication.LoginScreen">

    <TextView
        android:id="@+id/tvNote1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#8054c4"
        android:cursorVisible="false"
        android:gravity="center"
        android:hapticFeedbackEnabled="true"
        android:text="הערות"
        android:textAllCaps="false"
        android:textColor="@android:color/black"
        android:textSize="50sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:background="#8054c4"
        android:ems="10"
        android:layout_marginTop="5dp"
        android:hint="שם משתמש"
        android:inputType="textPersonName"
        android:textColorHint="@android:color/black"
        android:textSize="25sp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:background="#8054c4"
        android:layout_marginTop="5dp"
        android:ems="10"
        android:hint="סיסמה"
        android:inputType="textPersonName"
        android:textColorHint="@android:color/black"
        android:textSize="25sp"

        android:textStyle="bold" />

    <Button
        android:id="@+id/btnRegister"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:backgroundTint="#E61379C2"
        android:text="הירשם" />
```

```

        android:textColor="@android:color/black"
        android:textSize="20sp"
        android:textStyle="bold" />

```

```
</LinearLayout>
```

## d\_user\_guide.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/login"
    android:orientation="vertical"
    android:scaleType="fitCenter"
    android:adjustViewBounds="true"
    tools:context="com.example.comps.myapplication.LoginScreen">

    <TextView
        android:id="@+id/tvNote1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#8054cce4"
        android:cursorVisible="false"
        android:gravity="center"
        android:hapticFeedbackEnabled="true"
        android:text="לחצתהש מדרוך"
        android:textAllCaps="false"
        android:textColor="@android:color/black"
        android:textSize="50sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <ScrollView
        android:id="@+id/scrollView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scrollbarAlwaysDrawVerticalTrack="true">

        <ImageView
            android:id="@+id/ivUserGuide"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:adjustViewBounds="true"
            android:contentDescription="Specs"
            android:scrollbars="vertical"
            android:src="@drawable/pic1" />

    </ScrollView>

</LinearLayout>

```

## lo\_hebrew\_or\_english.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:background="@drawable/login"
    android:weightSum="8"

```

```

android:layout_height="match_parent">

<TextView
    android:id="@+id/tvTitle"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#8054cce4"
    android:gravity="center"
    android:textSize="40sp"
    android:layout_weight="1"
    android:textStyle="bold" />

<EditText
    android:id="@+id/etWord"
    android:layout_width="match_parent"
    android:layout_marginTop="5dp"
    android:layout_height="wrap_content"
    android:background="#8054cce4"
    android:ems="10"
    android:hint="מילה חכמה"
    android:inputType="textPersonName"
    android:textColorHint="@android:color/black"
    android:textSize="25sp"
    android:layout_weight="1"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnAddWord"
    android:textStyle="bold"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:textSize="20sp"
    android:layout_weight="1"
    android:text="מילה הוסף" />

<Button
    android:id="@+id/btnOpenOrLoadWordList"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="שמורה רשימה הרשימה/פתיחת שמירת"
    android:textSize="20sp"
    android:layout_weight="1"
    android:textStyle="bold" />

<TextView
    android:id="@+id/tvNote"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:background="#8054cce4"
    android:text="למחוק שברצונך מילה על תלחץ"
    android:textSize="20sp"
    android:layout_weight="1"
    android:textStyle="bold" />

<ListView
    android:id="@+id/lvWords"
    android:layout_marginTop="5dp"
    android:layout_width="match_parent"
    android:layout_height="285dp"
    android:background="#8054cce4"
    android:layout_weight="1"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnPractice"

```



```

        android:layout_width="match_parent"
        android:textStyle="bold"
        android:layout_height="wrap_content"
        android:backgroundTint="#E61379C2"
        android:textSize="20sp"
        android:layout_weight="1"
        android:text="לתרגל התחל" />
    <Button
        android:id="@+id/btnMenu"
        android:layout_width="match_parent"
        android:textStyle="bold"
        android:backgroundTint="#E61379C2"
        android:textSize="20sp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="לתפריט חזרה" />
</LinearLayout>

```

## lo login screen.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/login"
    android:orientation="vertical"
    tools:context="com.example.comps.myapplication.LoginScreen">

    <TextView
        android:id="@+id/tvNote1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#8054cce4"
        android:cursorVisible="false"
        android:gravity="center"
        android:hapticFeedbackEnabled="true"
        android:text="כניסה"
        android:textAllCaps="false"
        android:textColor="@android:color/black"
        android:textSize="50sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/etUserName"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:background="#8054cce4"
        android:ems="10"
        android:layout_marginTop="5dp"
        android:hint="שם משתמש"
        android:inputType="textPersonName"
        android:textColorHint="@android:color/black"
        android:textSize="25sp"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:background="#8054cce4"
        android:ems="10"

```

```

        android:layout_marginTop="5dp"
        android:hint="ס'יסמח"
        android:inputType="textPersonName"
        android:textColorHint="@android:color/black"
        android:textSize="25sp"

        android:textStyle="bold" />

<CheckBox
    android:id="@+id/cbRemember"
    android:layout_width="match_parent"
    android:layout_height="45dp"
    android:background="#E61379C2"
    android:layout_marginTop="5dp"
    android:text="אות' זכור"
    android:textSize="20sp"
    android:textStyle="bold"
    android:textColor="@android:color/black" />

<Button
    android:id="@+id/btnLogin"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:backgroundTint="#E61379C2"
    android:text="ה'כנס"
    android:textColor="@android:color/black"
    android:textSize="20sp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/tvNoAccount"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="נרשמת טרם?"
    android:background="#8054cce4"
    android:textColor="@android:color/black"
    android:textStyle="bold"
    android:textSize="25sp" />

<Button
    android:id="@+id/btnRegister"
    android:layout_width="match_parent"
    android:layout_height="50dp"
    android:backgroundTint="#E61379C2"
    android:text="עכשיו הרשם"
    android:textColor="@android:color/black"
    android:textSize="20sp"
    android:textStyle="bold" />

</LinearLayout>

```

## lo\_math.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="@drawable/login"

    android:layout_height="match_parent">

    <TextView
        android:id="@+id/tvNotel"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#8054cce4"

```

```

        android:gravity="center"
        android:text="תחנת יקנה"
        android:textSize="40sp"
        android:textStyle="bold" />

<TextView
    android:id="@+id/tvNote2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:background="#8054cce4"
    android:text="חשוקים סוג בחר"
    android:textSize="30sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnAddGames"
    android:layout_width="match_parent"
    android:textStyle="bold"
    android:backgroundTint="#E61379C2"
    android:textSize="20sp"
    android:layout_height="wrap_content"
    android:text="חידור חשוקי" />

<Button
    android:id="@+id/btnSubGames"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="חיסור חשוקי"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnMulGames"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="כפל חשוקי"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnDivGames"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="חילוק חשוקי"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnMenu"
    android:textStyle="bold"
    android:textSize="20sp"
    android:backgroundTint="#E61379C2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="לתפריט חזרה" /></LinearLayout>

```

## lo options.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="@drawable/login"

```

```

android:layout_height="match_parent">

<TextView
    android:id="@+id/tvHelloUser"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#8054cce4"
    android:gravity="center"
    android:textSize="40sp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/tvWhatToDo"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="5dp"
    android:text="לעשות תרצה מה?"
    android:background="#8054cce4"
    android:gravity="center"
    android:textStyle="bold"
    android:textSize="30sp" />

<Button
    android:id="@+id/btnHebrew"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="עברית לימוד"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnEnglish"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="אנגלית לימוד"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnMath"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="מתמטיקה לימוד"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnMusic"
    android:layout_width="match_parent"

    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="מוסיקה השמעת"
    android:textSize="20sp"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnStopMusic"
    android:layout_width="match_parent"

    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="מוסיקה הפסקת"
    android:textSize="20sp"

```

```
        android:textStyle="bold" />
    </LinearLayout>
```

## lo save or load word list.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:background="@drawable/login"
    android:weightSum="9"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/tvNote1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#8054cce4"
        android:gravity="center"
        android:text="מילים רשימות ושחירות פתיחת"
        android:textSize="28sp"
        android:layout_weight="1"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/tvNote2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:background="#8054cce4"
        android:text="למחוק שברצונך מילה על תלחץ"
        android:textSize="20sp"
        android:layout_weight="1"
        android:textStyle="bold" />

    <ListView
        android:id="@+id/lvWords"
        android:layout_width="match_parent"
        android:layout_height="155dp"
        android:layout_marginTop="5dp"
        android:background="#8054cce4"
        android:layout_weight="1"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/etWordListName"
        android:layout_width="match_parent"
        android:layout_marginTop="5dp"
        android:layout_height="wrap_content"
        android:background="#8054cce4"
        android:ems="10"
        android:hint="לרשימה שם הכנס"
        android:inputType="textPersonName"
        android:textColorHint="@android:color/black"
        android:textSize="25sp"
        android:layout_weight="1"
        android:textStyle="bold" />

    <Button
        android:id="@+id/btnAddWordList"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:backgroundTint="#E61379C2"
        android:text="רשימה הוסף"
        android:textSize="20sp"
        android:layout_weight="1"
        android:textStyle="bold" />
```

```

<TextView
    android:id="@+id/tvNote3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:background="#8054cce4"
    android:text="לפתוח שברצונך מילים רשימת על תלחץ"
    android:textSize="20sp"
    android:layout_weight="1"
    android:textStyle="bold" />

<ListView
    android:id="@+id/lvWordLists"
    android:layout_width="match_parent"
    android:layout_height="155dp"
    android:choiceMode="singleChoice"
    android:listSelector="#80FF1414"
    android:layout_marginTop="5dp"
    android:background="#8054cce4"
    android:layout_weight="1"
    android:textStyle="bold">

</ListView>

<Button
    android:id="@+id/btnRemove"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="רשימה מחק"
    android:textSize="20sp"
    android:layout_weight="1"
    android:textStyle="bold" />

<Button
    android:id="@+id/btnBack"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:backgroundTint="#E61379C2"
    android:text="חזרה"
    android:textSize="20sp"
    android:layout_weight="1"
    android:textStyle="bold" />

</LinearLayout>

```

## menu1.xml

```

<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/itUserGuide"
        android:textStyle="bold"
        android:title="לחשתחש הדריך" />
    <item
        android:id="@+id/itLogout"
        android:textStyle="bold"
        android:title="התנתק" /></menu>

```

## menu2.xml

```

<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/itUserGuide"

```



```

        android:textStyle="bold"
        android:title="לחשתמש מדר'ך" />
</menu>

```

## colors.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#3F51B5</color>
    <color name="colorPrimaryDark">#303F9F</color>
    <color name="colorAccent">#FF4081</color>
</resources>

```

## strings.xml

```

<resources>
    <string name="app_name">My Application</string>
</resources>

```

## styles.xml

```

<resources>

    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
        <item
name="android:textViewStyle">@style/TextAppearance.AppCompat.Headline</item>
    </style>

</resources>

```

## values.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="default_web_client_id" translatable="false">355005516140-
grmgtekdldhooaef4nociql4op8fb3h6k.apps.googleusercontent.com</string>
    <string name="firebase_database_url"
translatable="false">https://hahtava-a6f71.firebaseio.com</string>
    <string name="gcm_defaultSenderId"
translatable="false">355005516140</string>
    <string name="google_api_key" translatable="false">AIzaSyCBJFuM-
mRQ8i3vjp287Nhd-5yepMRbaZo</string>
    <string name="google_app_id"
translatable="false">1:355005516140:android:e485db748d1f34a43e8b49</string>
    <string name="google_crash_reporting_api_key"
translatable="false">AIzaSyCBJFuM-mRQ8i3vjp287Nhd-5yepMRbaZo</string>
    <string name="google_storage_bucket" translatable="false">hahtava-
a6f71.appspot.com</string>
    <string name="project_id" translatable="false">hahtava-a6f71</string>
</resources>

```