ספר פרויקט

חלופת תכנות טלפונים חכמים  
אביב סאייר ת.ז. 326552304  
ביה"ס עירוני ד' ע"ש אהרון קציר

תוכן עניינים

[מענה לדרישות הפרויקט - 4 -](#_Toc166359793)

[מבוא - 5 -](#_Toc166359794)

[שירותים - 6 -](#_Toc166359795)

[שירות מוזיקת הרקע – **BackgroundMusicService** - 6 -](#_Toc166359796)

[משתנים - 6 -](#_Toc166359797)

[פעולות - 6 -](#_Toc166359798)

[מסכים - 7 -](#_Toc166359799)

[מסך הפתיחה – **StartActivity** - 7 -](#_Toc166359800)

[משתנים - 7 -](#_Toc166359801)

[פעולות - 7 -](#_Toc166359802)

[מסך הקרדיטים – **CreditsActivity** - 9 -](#_Toc166359803)

[משתנים - 9 -](#_Toc166359804)

[פעולות - 9 -](#_Toc166359805)

[מסך הוראות המשחק – **HowToPlayActivity** - 11 -](#_Toc166359806)

[משתנים - 11 -](#_Toc166359807)

[פעולות - 11 -](#_Toc166359808)

[מסך השלב – **LevelActivity** - 13 -](#_Toc166359809)

[משתנים - 13 -](#_Toc166359810)

[פעולות - 13 -](#_Toc166359811)

[מסך התפריט – **MenuActivity** - 19 -](#_Toc166359812)

[משתנים - 20 -](#_Toc166359813)

[פעולות - 22 -](#_Toc166359814)

[מחלקות - 24 -](#_Toc166359815)

[מחלקת פעולות עזר - **Util** - 24 -](#_Toc166359816)

[פעולת ה-within - 24 -](#_Toc166359817)

[פעולת ה-between - 24 -](#_Toc166359818)

[פעולת ה-withAlpha - 24 -](#_Toc166359819)

[פעולת ה- isCounterClockwise - 24 -](#_Toc166359820)

[פעולת ה-doesIntersect - 24 -](#_Toc166359821)

[פעולת ה-generateVignette - 25 -](#_Toc166359822)

[פעולת ה-randomElement - 25 -](#_Toc166359823)

[פעולת ה-timeToString - 25 -](#_Toc166359824)

[פעולת ה-updateUUID - 25 -](#_Toc166359825)

[פעולת ה-getLevel - 26 -](#_Toc166359826)

[פעולת ה-stringToTime - 26 -](#_Toc166359827)

[מחלקת חוזה ממסד הנתונים – **DatabaseContract** - 27 -](#_Toc166359828)

[מבנה ממסד הנתונים - 27 -](#_Toc166359829)

[טבלת המשתמשים – **UserEntry** - 27 -](#_Toc166359830)

[מחלקת עוזר הממסד – **DatabaseHelper** - 28 -](#_Toc166359831)

[משתנים - 28 -](#_Toc166359832)

[פעולות - 28 -](#_Toc166359833)

[מחלקת ספק/כותב הנתונים – **DataSource** - 28 -](#_Toc166359834)

[משתנים - 28 -](#_Toc166359835)

[פעולות - 28 -](#_Toc166359836)

# מענה לדרישות הפרויקט

|  |  |  |
| --- | --- | --- |
| **פרק במחוון** | **נושאים שמומשו** | **באיזו מחלקה?** |
| **6** | כתיבת מחלקת **Service** | **BackgroundMusicService** |
| יצירת אנימציה (לא כולל שימוש במחלקת אנימציה) | **Player** |
| **7** | אחסון וטיפול בנתונים (חובה שמירה ושליפה) | **DatabaseHelper**  **DataSource**  **LevelDataSource**  **UUIDDataSource** |
| **9** | **SurfaceView** | **Game**  **Level**  **LevelPreview** |
| **Thread**s | **PeriodicThread**  **JobScheduler** |
| **10** | **SharedPreference**s | **UUIDDataSource** |

# מבוא

# שירותים

כל השירותים הם בתיקייה service.

## שירות מוזיקת הרקע – **BackgroundMusicService**

תורשה מטיפוס ה-**Service**.

השירות מפעיל את R.raw.*background1* ברקע בלולאה אין סופית.

השמע מופעל בחצי מהעוצמה המקסימלית.

השירות הוא STICKY – משמע שאם תהליך השירות נקטע, השירות יתחיל מחדש עם **Intent** של null.

### משתנים

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| player | android.media.**MediaPlayer** | private | מוזיקת הרקע הטעונה. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| onStartCommant | android.content.**Intent**, int, int | int | public | טוענת את מוזיקת הרקע ל-player, מפעילה אותו בלולאה אינסופית בחצי מהעוצמה המקסימלית, ומחזירה שהיא במצב START\_STICKY. |
| onDestroy | none | void | public | עוצרת את player ומשחררת אותו. |

A screenshot of a computer program

Description automatically generated

# מסכים

כל המסכים הם בתיקייה activity.

## מסך הפתיחה – **StartActivity**

תורשה מטיפוס ה-**Activity**.

המסך הוא אבן הדרך בין כל המסכים במשחק, למעט **LevelActivity**.

המסך הוא באוריינטציה אופקית ובמסך מלא.

יש אפשרות למעבר למסך הקרדיטים (**CreditsActivity**), למסך הוראות המשחק (**HowToPlayActivity**), ולמסך התפריט (**MenuActivity**), ע"י בחירת האפשרות ב- rg\_options\_menu ואז לחיצה על הכפתור btn\_go.

בהתחלה אפשרות המעבר אל **MenuActivity** היא האפשרות המסומנת ב-rg\_options\_menu.

מפני שזה המסך הראשון, פה נבדק ה-uuid של המשתמש ומעודכן כפי שמפורט ב-**Util**.updateUUID.

### משתנים

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| rg\_options\_menu | android.widget.**RadioGroup** | private | תפריט האפשרויות למעבר בין חלונות. |
| btn\_go | android.widget.**Button** | private | הכפתור שמעביר בין החלונות ע"פ rg\_options\_menu. |
| click | android.media.**MediaPlayer** | private | אפקט השמע של לחיצה על כפתור. |
| load | android.media.**MediaPlayer** | private | אפקט השמע של מסך טוען. |
| uuidDataSource | .database.datasource.**UUIDDataSource** | private | מקור מידע ל-Universal User Identity. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| onCreate | android.os.**Bundle** | void | Protected | טוענת את start\_activity.xml, את כל קלות השמע, מפעילה את השמע load, ומגדירה את מה שקורה שכפתורים נלחצים. |

A screenshot of a computer

Description automatically generatedA screenshot of a computer program

Description automatically generated

A screenshot of a video game

Description automatically generated

A screen shot of a computer

Description automatically generated

## מסך הקרדיטים – **CreditsActivity**

תורשה מטיפוס ה-**Activity**.

מסך בו יש את כל המידע על המוזיקה שבמשחק, האיורים, ומי יצר אותו.

המסך הוא באוריינטציה אופקית ובמסך מלא.

יש אפשרות למעבר למסך הבית (**StartActivity**) בכפתור ה"אחורה" (btn\_back) שבפינה השמאלית העליונה של המסך.

בנוסף, יש קישורים הניתנים ללחיצה המעבירים אותך אל המקורות הנ"ל.

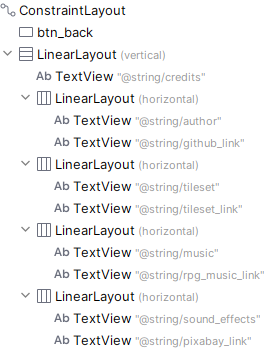
### משתנים

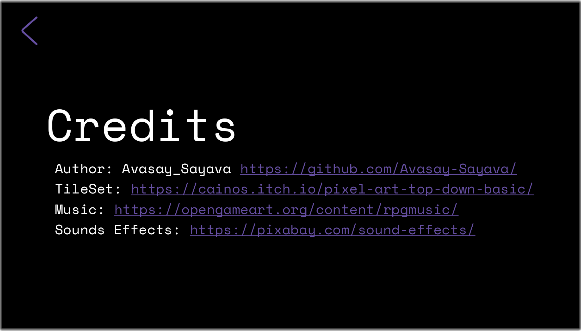
|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| btn\_back | android.widget.**Button** | private | כפתור החזרה לאחור. מוביל אל מסך הבית (**StartActivity**). כשהוא נלחץ מופעל השמע click. |
| click | android.media.**MediaPlayer** | private | אפקט השמע של לחיצה על כפתור. |
| load | android.media.**MediaPlayer** | private | אפקט השמע של מסך טוען. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| onCreate | android.os.**Bundle** | void | protected | טוענת את credits\_activity.xml, את כל קלות השמע, מפעילה את השמע load, ומגדירה את מה שקורה שכפתורים נלחצים. |

A screenshot of a computer

Description automatically generated



A blue screen with white text

Description automatically generated

## מסך הוראות המשחק – **HowToPlayActivity**

תורשה מטיפוס ה-android.app.**Activity**.

במסך יש תמונה המסבירה איך לשחק במשחק.

המסך הוא באוריינטציה אופקית ובמסך מלא.

יש אפשרות למעבר למסך הבית (**StartActivity**) בכפתור ה"אחורה" (btn\_back) שבפינה השמאלית העליונה של המסך.

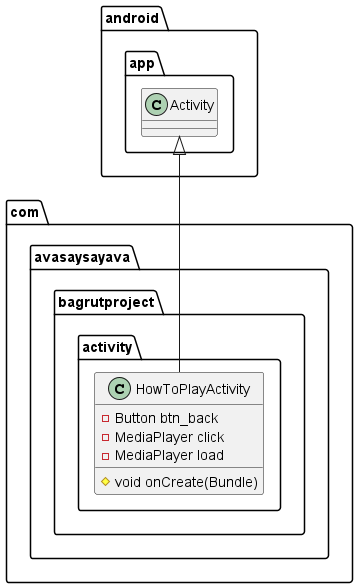
### משתנים

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| btn\_back | android.widget.**Button** | private | כפתור החזרה לאחור. מוביל אל מסך הבית (**StartActivity**). כשהוא נלחץ מופעל השמע click. |
| click | android.media.**MediaPlayer** | private | אפקט השמע של לחיצה על כפתור. |
| load | android.media.**MediaPlayer** | private | אפקט השמע של מסך טוען. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| onCreate | android.os.**Bundle** | void | protected | טוענת את how\_to\_play\_activity.xml, את כל קלות השמע, מפעילה את השמע load, ומגדירה את מה שקורה שכפתורים נלחצים. |

A close-up of text

Description automatically generated



A blue screen with white text

Description automatically generated

## מסך השלב – **LevelActivity**

תורשה מטיפוס ה-android.app.**Activity**.

במסך מופעל השלב (**Level**) הנבחר במסך ה-**MenuActivity**, ומופעלת מוזיקת רקע ע"י **BackgroundMusicService**.

המסך הוא באוריינטציה אופקית ובמסך מלא.

יש אפשרות לעצור את המשחק בכפתור העצירה, ואז מופעל מסך העצירה.

במסך הנ"ל מתגלים כפתורי ה-toggle למצבי ה-debug והגרפים, ומתגלה כפתור היציאה, שמוסתרים בהתחלה.

אין אפשרות להפעיל רק את מצב הגרפים, והוא יבוא תמיד עם מצב ה-debug.

### משתנים

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| sv\_level | .game.**Level** | private | כפתור החזרה לאחור. מוביל אל מסך הבית (**StartActivity**). כשהוא נלחץ מופעל השמע click. |
| backgroundMusicService | android.content.**Intent** | private | שירות הפעלת מוזיקת הרקע טעון ב-Intent |
| img\_btn\_play | android.widget.**ImageButton** | private | כפתור העצירה/המשכה של המשחק. כשהוא נלחץ מופעל השמע click. |
| img\_btn\_debug | android.widget.**ImageButton** | private | כפתור toggle למצב debug. כשהוא נלחץ מופעל השמע click. |
| img\_btn\_graph | android.widget.**ImageButton** | private | כפתור toggle למצב גרפים. כשהוא נלחץ מופעל השמע click. |
| img\_btn\_quit | android.widget.**ImageButton** | private | כפתור לעזיבת המשחק. כשהוא נלחץ מופעל השמע click, והמשתמש מועבר לדיאלוג השואל אותו האם הוא באמת רוצה לעזוב. |
| levelDataSource | .database.datasource.**LevelDataSource** | private | מקור מידע על השלב הנוכחי. |
| uuidDataSource | .database.datasource.**UUIDDataSource** | private | מקור מידע ל-Universal User Identity. |
| click | android.media.**MediaPlayer** | private | אפקט השמע של לחיצה על כפתור. |
| load | android.media.**MediaPlayer** | private | אפקט השמע של מסך טוען. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| onCreate | android.os.**Bundle** | void | protected | טוענת את level\_activity.xml, את כל קלות השמע, מפעילה את השמע load, מגדירה את מה שקורה שכפתורים נלחצים, ומפעילה את מוזיקת הרקע. |
| resumeLevel | none | void | private | ממשיכה את פעילות השלב הנוכחי, מסתירה את כל כפתורי מסך העצירה, ומפעילה את אנימציית הכפתור img\_btn\_play כך שתיראה בו תמונה המתארת עצירה. |
| pauseLevel | none | void | private | עוצרת את פעילות השלב הנוכחי, מגלה את כל כפתורי מסך העצירה, ומפעילה את אנימציית הכפתור img\_btn\_play כך שתיראה בו תמונה המתארת הפעלה. |
| showLeaveDialog | none | void | private | יוצרת מראה את דיאלוג היציאה. |
| onStart | none | void | protected | מפעילה את השמע load, מעדכנת את השלב במצב הנוכחי. |
| onResume | none | void | protected | מפעילה את מוזיקת הרקע, מעדכנת את השלב במצב הנוכחי. |
| onPause | none | void | protected | עוצרת את השלב ואת מוזיקת הרקע, מעדכנת את השלב במצב הנוכחי. |
| onStop | none | void | protected | מעדכנת את השלב במצב הנוכחי. |
| onBackPressed | none | void | public | עוצרת את השלב ומראה את דיאלוג היציאה. |
| onCompleted | long | void | public | מעדכנת את ה-database בזמן הטוב ביותר ומשווה אותו לזמן שהיא מקבלת (הזמן הנוכחי), ומראה את דיאלוג סיום השלב. |
| showTimeDialog | long, long, boolean | void | private | יוצרת ומראה את דיאלוג סיום השלב. |

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

מצב פועל:

A screenshot of a video game

Description automatically generated

A blue background with white text

Description automatically generated

מצב עצור:

A screenshot of a video game

Description automatically generated

A blue rectangular object with white text

Description automatically generated

מצב debug:

A screenshot of a video game

Description automatically generated

מצב גרפים + debug:

A screenshot of a video game

Description automatically generated

דיאלוג יציאה:

A screenshot of a video game

Description automatically generated

דיאלוג סיום שלב:

A screenshot of a video game

Description automatically generated

דיאלוג סיום שלב – שיא חדש:

A screenshot of a video game

Description automatically generated

## מסך התפריט – **MenuActivity**

תורשה מטיפוס ה-android.app.**Activity**.

במסך מופעל **LevelPreview** על השלב הראשון בהתחלה (כששום שלב לא נבחר), ואז על השלב האחרון שנבחר.

המסך הוא באוריינטציה אופקית ובמסך מלא.

יש במסך כפתור לכל תשעת השלבים שאפשר לבחור, כפתור לחזרה אל מסך הפתיחה (**StartActivity**) וכפתור להתחלת השלב שמעביר ל-**LevelActivity**.

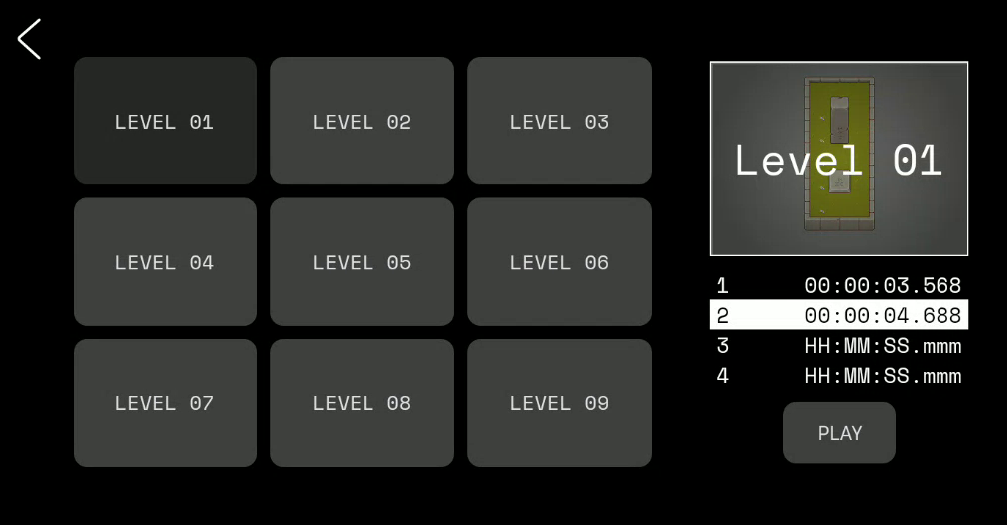
לכל שלב יש לוח תוצאות משלו, הנטען מה-Database, ובו 4 אפשרויות הצגה הנשלטות ע"י המערכת.

1. המשתמש ראשון בדירוג, אז רואים את המקום הראשון (שהוא המשתמש) מסומן בלבן, ואת שלושת המקומות שאחריו.

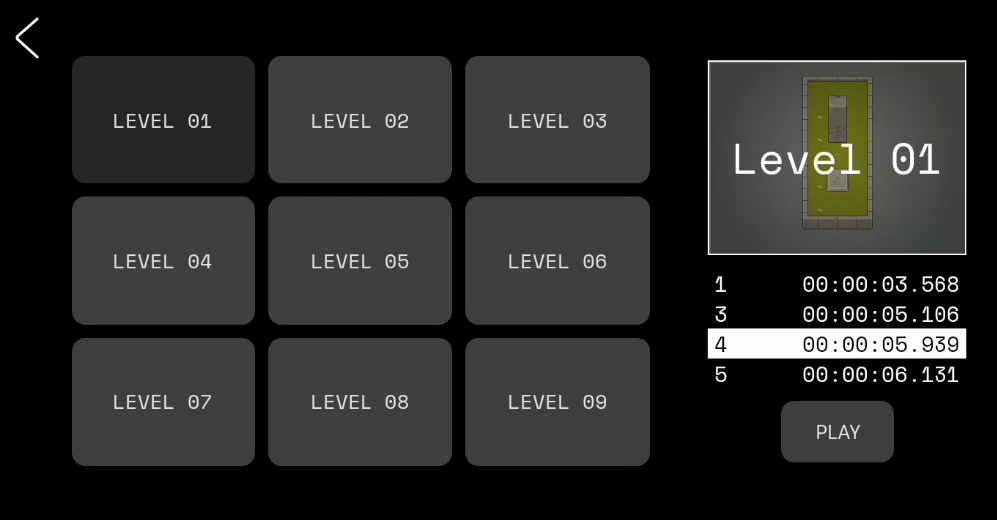
A screenshot of a video game

Description automatically generated

1. המשתמש שני בדרוג, אז רואים את המקום הראשון, את המקום השני מסומן בלבן (שהוא המשתמש) ואת שני המקומות שאחריו.



1. המשתמש שלישי ומעלה בדירוג, אז רואים את הראשון בשורה הראשונה, אחד מעל המשתמש בשורה השנייה, את המשתמש בשורה השלישית מסומן בלבן, ואת המקום אחרי המשתמש מתחתיו.



1. המשתמש עדיין לא שיחק – רואים את ארבעת הראשונים.

A screenshot of a video game

Description automatically generated

אם אין מידע על דירוג (לדוגמה יש רק 2 שחקנים ששיחקו ויש 4 מקומות בלוח התוצאות), אז רואים "HH:MM:SSS.mmm" במקום הזמן. (כמו שאפשר לראות בדוגמאות קודמות)

### משתנים

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| levels | android.widget.**Button** | private | מערך המכיל את תשע כפתורי השלבים. שלב 1 הוא levels[0], שלב 2 הוא levels[1] ... שלב 9 הוא levels[8]. כשכל אחד מהכפתורים הללו נלחץ, מופעל השמע click. |
| currentLevel | int | private | ה-index של השלב הנוכחי. בהתחלה תמיד 0 (לשלב 1). |
| btn\_play | android.widget.**Button** | private | מעביר אל **LevelActivity** כדי להתחיל את השלב הנבחר. כשהוא נלחץ מופעל השמע click. |
| btn\_back | android.widget.**Button** | private | כפתור החזרה לאחור. מוביל אל מסך הבית (**StartActivity**). כשהוא נלחץ מופעל השמע click. |
| levelDataSource | .database.datasource.**LevelDataSource** | private | מקור מידע על השלב הנוכחי. |
| uuidDataSource | .database.datasource.**UUIDDataSource** | private | מקור מידע ל-Universal User Identity. |
| rl\_rank\_first | android.widget.**RelativeLayout** | private | מחזיק את המקום של הראשון ואת התוצאה שלו בתוכו. |
| rl\_rank\_faster | android.widget.**RelativeLayout** | private | מחזיק את המקום של מי שמקום אחד מעל המשתמש ואת התוצאה שלו. |
| rl\_rank\_current | android.widget.**RelativeLayout** | private | מחזיק את המקום של המשתמש ואת התוצאה שלו בתוכו. |
| rl\_rank\_slower | android.widget.**RelativeLayout** | private | מחזיק את המקום של מי שמקום אחד מתחת המשתמש ואת התוצאה שלו. |
| txt\_rank\_first\_time | android.widget.**TextView** | private | מחזיק את התוצאה של הראשון. |
| txt\_rank\_faster\_time | android.widget.**TextView** | private | מחזיק את התוצאה של מי שמקום אחד מעל המשתמש. |
| txt\_rank\_current\_time | android.widget.**TextView** | private | מחזיק את התוצאה של המשתמש. |
| txt\_rank\_slower\_time | android.widget.**TextView** | private | מחזיק את התוצאה של מי שמקום מתחת למשתמש. |
| txt\_rank\_first\_number | android.widget.**TextView** | private | מחזיק את המקום של הראשון. |
| txt\_rank\_faster\_number | android.widget.**TextView** | private | מחזיק את המקום של מי שמקום אחד מעל המשתמש. |
| txt\_rank\_current\_number | android.widget.**TextView** | private | מחזיק את המקום של המשתמש. |
| txt\_rank\_slower\_number | android.widget.**TextView** | private | מחזיק את המקום של מי שמקום מתחת למשתמש. |
| sv\_level\_preview | .game.**LevelPreview** | private | **SurfaceView** המראה תצוגה מקדימה של השלב. |
| click | android.media.**MediaPlayer** | private | אפקט השמע של לחיצה על כפתור. |
| load | android.media.**MediaPlayer** | private | אפקט השמע של מסך טוען. |
| txt\_level\_title | android.widget.**TextView** | private | מחזיק את כותרת השלב. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| onCreate | android.os.**Bundle** | void | protected | טוענת את menu\_activity.xml, את כל קלות השמע, מפעילה את השמע load, מגדירה את מה שקורה שכפתורים נלחצים, ומפעילה את sv\_level\_preview על השלב הראשון. |
| loadLeaderboards | int | void | private | טוענת את לוח התוצאת על השלב באינדקס id. |
| previewMap | Int, GameMap | void | private | מעדכנת את הכותרת של השלב הנבחר, מפעילה את sv\_level\_preview על השלב הנבחר, וקוראת ל-loadLeaderboards עליו. |
| onStart | none | void | protected | מפעילה את השמע load, מעדכנת את sv\_level\_preview במצב הנוכחי. |
| onResume | none | void | protected | מפעילה את מוזיקת הרקע, מעדכנת את sv\_level\_preview במצב הנוכחי. |
| onPause | none | void | protected | עוצרת את השלב ואת מוזיקת הרקע, מעדכנת את sv\_level\_preview במצב הנוכחי. |
| onStop | none | void | protected | מעדכנת את sv\_level\_preview במצב הנוכחי. |
| showTimeDialog | long, long, boolean | void | private | יוצרת ומראה את דיאלוג סיום השלב. |

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

# מחלקות

## מחלקת פעולות עזר - **Util**

### פעולת ה-within

כותרת הפעולה:

public static boolean within(double lower, double num, double upper)

הפעולה מחזירה האם num הוא בין lower ל-upper או שווה לאחד מהם.

### פעולת ה-between

כותרת הפעולה:

public static boolean between(double lower, double num, double upper)

הפעולה מחזירה האם num הוא בין lower ל-upper ולא שווה לאחד מהם.

### פעולת ה-withAlpha

כותרת הפעולה: public static int withAlpha(int color, int alpha)

הפעולה מחזירה את הצבע color עם ערך שקיפות alpha.

### פעולת ה- isCounterClockwise

כותרת הפעולה:

public static boolean isCounterClockwise(PointF A, PointF B, PointF C)

הפעולה משתמשת בשיפועי הישרים AB, AC כדי להחליט אם הנקודות מסודרות נגד כיוון השעון.

### פעולת ה-doesIntersect

כותרת הפעולה:

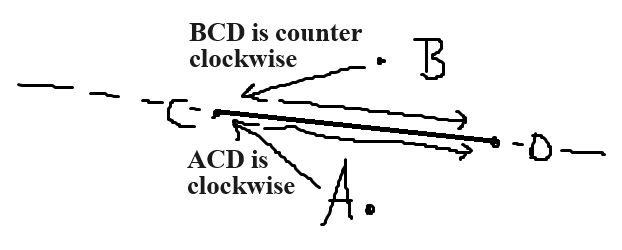
public static boolean doesIntersect(PointF A, PointF B, PointF C, PointF D)

הפעולה מחזירה האם הסגמנט AB חותך את הסגמנט CD. הפעולה תמיד מחזירה false אם הסגמנטים מקבילים.

איך הפעולה עובדת? הפעולה מחזירה את הטענה הבאה:

return isCounterClockwise(A, C, D) != isCounterClockwise(B, C, D) && isCounterClockwise(A, B, C) != isCounterClockwise(A, B, D);

נוכל לתרגם את isCounterClockwise(A, C, D) != isCounterClockwise(B, C, D) כ"האם הנקודות A ו-B בצדדים שונים של הישר CD?", כמו שאפשר להבין מהסרטוט הבא:



פה רצף הנקודות BCD מסודר נגד כיוון השעון, בניגוד לרצף ACD שמסודר עם כיוון השעון – וכך A ו-B בשני צדדים שונים של ה**ישר** (לא סגמנט) CD. בניגוד, אם A ו-B היו באותו צד של הישר CD: A diagram of a person's body

Description automatically generated

מכך, אם isCounterClockwise(A, C, D) != isCounterClockwise(B, C, D) אזי ה**ישר** CD חותך את ה**סגמנט** AB.

כדי לדעת אם ה**סגמנט** CD חותך את ה**סגמנט** AB, נצטרך גם לבדוק את אותו דבר רק על הישר AB:

isCounterClockwise(A, B, C) != isCounterClockwise(A, B, D)

אם שני התנאים מתקיימים, אז הסגמנטים נחתכים.

### פעולת ה-generateVignette

כותרת הפעולה:

public static @NonNull Bitmap generateVignette(int width, int height)

הפעולה מחזירה תמונה (**Bitmap**) שקופה עם אפקט האפלה על פינותיה בגודל width על height.

### פעולת ה-randomElement

כותרת הפעולה: public static <T> T randomElement(T[] arr)

הפעולה מחזירה איבר רנדומלי מ-arr.

### פעולת ה-timeToString

כותרת הפעולה:

public static String timeToString(long millis)

הפעולה מחזירה מחרוזת בפורמט HH:MM:SS.sss הנוצרת מהמילי שניות המתורגמות לשעות, דקות, שניות ומילי שניות.

### פעולת ה-updateUUID

כותרת הפעולה:

public static void updateUUID(UUIDDataSource uuidDataSource)

הפעולה בודקת האם למשתמש UUID (Universally Unique Identifier) שקיים ב-Database. אם לא, הפעולה יוצרת למשתמש UUID חדש ומעדכנת את ה-Database.

### פעולת ה-getLevel

כותרת הפעולה:

public static String getLevel(GameMap map)

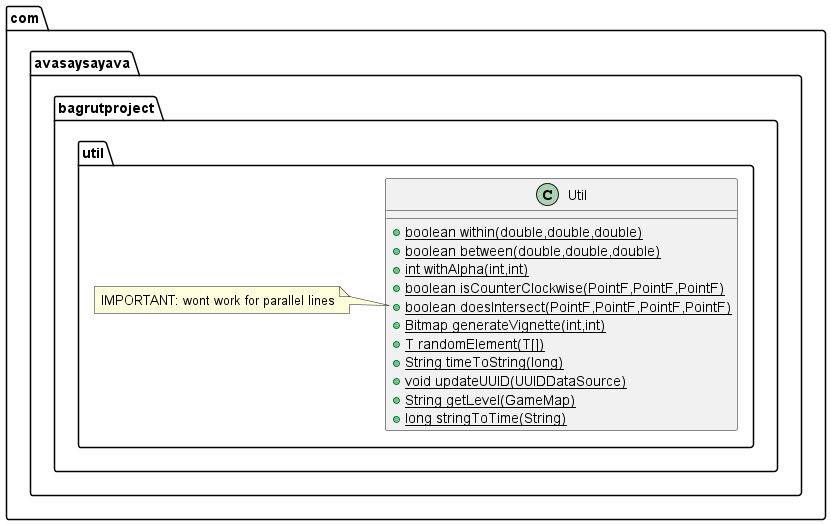
הפעולה מחזירה את שם העמודה בטבלת ה-Users ב-Database המתאים לשלב.

### פעולת ה-stringToTime

כותרת הפעולה:

public static long stringToTime(String timeString)

היפוך תהליך [פעולת ה-timeToString](#_פעולת_ה-timeToString).



## מחלקת חוזה ממסד הנתונים – **DatabaseContract**

A screenshot of a computer

Description automatically generatedהמחלקה מחזיקה נתונים סטטיים על ממסד הנתונים, ובפרט על הטבלאות שבו.

### מבנה ממסד הנתונים

ממסד הנתונים מורכב מטבלה אחת: Users.

בטבלה 11 עמודות:

* עמודה ל-id (המספר הסידורי של המשתמש) מסוג INTEGER.
* עמודה ל-uuid (רצף רנדומלי של 36 תווים. לכל משתמש רצף משלו. למה לא להשתמש ב-id כדי להבין מי זה מי? אם מישהו מצליח להכנס עם id אחר, זה מאוד קל לבחור id, כי אתה יודע שה-id מסודר בסדר עולה עוקב מ-1. מצד שני, אם מישהו מצליח להכנס עם uuid אחר, יהיה לו קשה לנחש מתוך 36 בחזקת 32 אפשרויות של uuid-ים)
* עמודה לכל שלב (מסוג Time).

### טבלת המשתמשים – **UserEntry**

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| TABLE\_NAME | **String** | public static | שם הטבלה. |
| COLUMN\_UUID | שם עמודת ה-uuid. |
| COLUMN\_LEVEL1 | שם עמודת השלב הראשון. |
| COLUMN\_LEVEL2 | שם עמודת השלב השני. |
| COLUMN\_LEVEL3 | שם עמודת השלב השלישי. |
| COLUMN\_LEVEL4 | שם עמודת השלב הרביעי. |
| COLUMN\_LEVEL5 | שם עמודת השלב החמישי. |
| COLUMN\_LEVEL6 | שם עמודת השלב השישי. |
| COLUMN\_LEVEL7 | שם עמודת השלב השביעי. |
| COLUMN\_LEVEL8 | שם עמודת השלב השמיני. |
| COLUMN\_LEVEL9 | שם עמודת השלב התשיעי. |

## מחלקת עוזר הממסד – **DatabaseHelper**

תורשה מטיפוס **SQLiteOpenHelper**.

המחלקה נותנת גישה ל-Database הראשי – Database.db, שהוא ממסד נתונים של SQLite

### משתנים

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| DATABASE\_NAME | **String** | private static | שם ממסד הנתונים. |
| SQL\_CREATE\_ENTRIES | השאילתה ליצירת הטבלאות בתוך הממסד. |
| SQL\_DELETE\_ENTRIES | השאילתה למחיקת הטבלאות בתוך הממסד, למקרה של עדכון גרסה. |
| DATABASE\_VERSION | int | גרסת ממסד הנתונים. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| onCreate | android.database.sqlite.**SQLiteDatabase** | void | public | מופעלת כשממסד הנתונים נוצר. יוצרת את כל הטבלאות הנדרשות בעזרת SQL\_CREATE\_ENTRIES. |
| onUpgrade | android.database.sqlite.**SQLiteDatabase**,  int, int | void | public | מופעלת כשמעדכנים את גרסת ממסד הנתונים. יוצרת מחדש את הטבלאות בעזרת SQL\_DELETE\_ENTRIES, וקריאה למתודה onCreate. |

## מחלקת ספק/כותב הנתונים – **DataSource**

מחלקה אבסטרקטית המהווה מבנה למחלקות ה-**LevelDataSource** ול-**UUIDDataSource**.

למחלקה **DatabaseHelper** משלה, המספק **SQLLiteDatabase** הניתן לקריאה או לכתיבה.

מחלקות מסוג זה אמורות לספק גישה לנושא מסוים בלי להסתבך כל פעם מחדש עם תפעול ה-database.

### משתנים

|  |  |  |  |
| --- | --- | --- | --- |
| שם | סוג | גישה | הערות |
| databaseHelper | .database.**DatabaseHelper** | protected | עוזר הממסד לפי ה-context. |
| context | android.content.**Context** | protected | ה-constext של המצב הנוכחי. |
| database | android.database.sqlite.**SQLiteDatabase** | protected | ה-database המתקבל ע"י databaseHelper בעזרת openReadable, openWriteable. |

### פעולות

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם | קלט | פלט | גישה | הערות |
| openWriteable | none | void | public | מבקש ב-databaseHelper writeable-database ומיישם אותו אל המשתנה database. |
| openReadable | none | void | public | מבקש ב-databaseHelper readable-database ומיישם אותו אל המשתנה database. |
| close | none | void | public | סוגר את databaseHelper, ומשחרר אותו ואת כל ממסדי הנתונים הקשורים אליו. |