**ספר פרויקט**

**ARCADE**

**מאת: רון שגב, ת.ז 216136523**

**יב'6- תיכון "הדרים"**

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**שם המנחה: ניר סליקטר**

**תוכן עניינים**

[**מבוא** 3](#_Toc73814046)

[**תיאור תכולת הספר** 3](#_Toc73814047)

[**הרקע לפרויקט** 3](#_Toc73814048)

[**תהליך המחקר** 4](#_Toc73814049)

[**אתגרים מרכזיים** 4](#_Toc73814050)

[**מוטיבציה לעבודה:** 5](#_Toc73814051)

[**הצורך עליו הפרויקט עונה:** 5](#_Toc73814052)

[**מבנה/ארכיטקטורה של הפרויקט** 6](#_Toc73814053)

[**הצגת הפתרון המוצע והסיבות לבחירתו** 6](#_Toc73814054)

[**תרשים יחידות** 6](#_Toc73814055)

[**באילו טכנולוגיות נעשה שימוש בפרויקט** 11](#_Toc73814056)

[**ספריות לקוד Java באנדרואיד סטודיו:** 11](#_Toc73814057)

[**מדריך למשתמש** 12](#_Toc73814058)

[**הוראות התקנה** 12](#_Toc73814059)

[**היררכיית המסכים והמעברים ביניהם** 12](#_Toc73814060)

[**תיאור מסכי האתר** 13](#_Toc73814061)

[**מדריך למפתח** 18](#_Toc73814062)

[**קישור לקוד המלא בGitHub** 18](#_Toc73814063)

[**הסבר כללי על קוד הפרויקט** 18](#_Toc73814064)

[**הסבר על קבצים, מחלקות ופעולות חשובות** 18](#_Toc73814065)

[**סיכום אישי / רפלקציה** 29](#_Toc73814066)

[**תחושותיי מהפרויקט** 29](#_Toc73814067)

[**מה קיבלתי וכלים שאקח איתי להמשך** 29](#_Toc73814068)

[**קשיים ואתגרים שעמדו בפני** 29](#_Toc73814069)

[**מסקנותיי מהפרויקט** 30](#_Toc73814070)

[**מה הייתי עושה אחרת לו הייתי מתחיל היום** 30](#_Toc73814071)

[**תכונות שהייתי רוצה להוסיף לפרויקט** 30](#_Toc73814072)

[**ביבליוגרפיה** 31](#_Toc73814073)

[**נספחים** 32](#_Toc73814074)

[**נספח 1 – קישור לקוד המלא בGitHub** 32](#_Toc73814075)

[**נספח 2-קוד נוסף שלא הוספתי במדריך למפתח** 32](#_Toc73814076)

# 

# **מבוא**

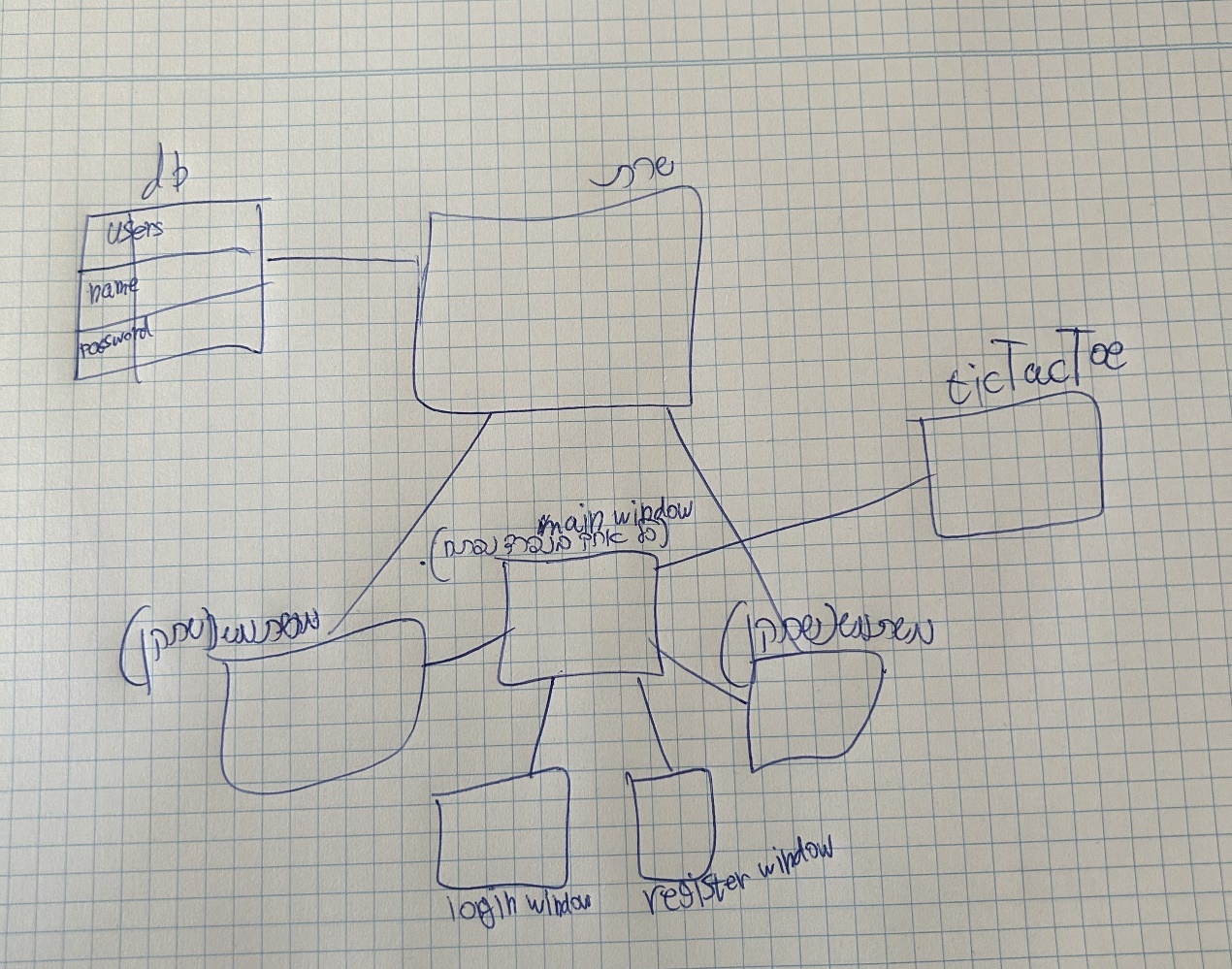
# **תיאור תכולת הספר**

בספר זה אפרט על פרויקט הסיום שלי –תחרות בין שחקנים )משתמשים) שיתחרו בכמה משחקים שונים, פרויקט לו קראתי בשםArcade

הספר כולל את הרקע לפרויקט, מטרתו וקהל היעד שלו. אפרט את החידושים בפרויקט לעומת מוצרים קיימים, ואת התחומים אותם הייתי צריך לחקור וללמוד בשביל להכין את הפרויקט. בספר נמצא תרשים הפרויקט, מדריך למשתמש במערכת וצילומי מסך. בנוסף, מפורט מדריך למפתח ובסוף גם רפלקציה אישית על הרגשתי בזמן העבודה ובסיומה.

# **הרקע לפרויקט**

# **מבנה/ארכיטקטורה של הפרויקט**

**באילו טכנולוגיות נעשה שימוש בפרויקט**

* **Python –** שפת התכנות המרכזית בה נכתב הפרויקט.
* **ספריות שהשתמשתי בהם**
* **Pygame**
* **Sqlite3**
* **Threading**
* **Socket**
* **Tkinter**

**דרישות חובה בפרויקט**

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***נושא בדרישות משרד החינוך*** | ***הרחבה על הנושא*** | ***מימוש בפרויקט שלי*** |
| 1 | **תכנות מונחה עצמים** | מימוש לפחות 4 מחלקות שונות | יש לי מעל 4 מחלקות כמו מחלקת user מחלקת לרישום, מחלקה |
| 2 | **תקשורת** | מימוש שרת ולקוח מבוססי סוקטים.  מימוש שרת מרובה לקוחות.  מימוש פרוטוקול העברת הודעות  מצד לצד ע"י התלמיד. | ביצעתי כמתואר בתיאור פרוטוקול  התקשורת.  יש תקשורת בין המשתמשים בכל המשחקים עצמם. |
| 3 | **מערכת הפעלה** | שימוש בתהליכונים | השתמשתי בThreading |
| 4 | **ממשק משתמש** | מימוש ממשק משתמש אינטראקטיבי | ממשק האפליקציה של הלקוח. |
| 5 | **אבטחה** | הצפנה למידע רגיש העובר בתקשורת | אממש בעתיד הצפנה של סיסמאות המשתמשים בבסיס הנתונים. |
|  |  |  |  |
|  |  |  |  |

**פרוטוקול הפרויקט:**

ההודעות בין המשתמשים לשרת בפרויקט שלי מועברות באותה דרך, על ידי שימוש באותן פעולות שחותכות את ההודעות לחתיכות מידע מתאימות ובאותה מידה יש פעולות שיודעות לפענח את חתיכות המידע ולהבין מה צריך לעשות.

פענוח דוגמה להודעה שהשרת יכול לקבל:

**22-** שם המשתמש ששלח את ההודעה

**11-** שם המשתמש שאליו תועבר ההודעה עם המיקומים

**play –** הפעולה שהמשתמש מבצע

**tictactoe-** איזה משחק משוחק

**pos-** מודיע שהשרת עומד לקבל מיקום לחיצה

**113-** מיקום של הלחיצה

**91-** מיקום של הלחיצה

# **תהליך המחקר**

**סקירת המצב הקיים בשוק**:

כשהתחלתי את עבודתי בדקתי מה קיים בשוק. בשוק כיום ישנן מספר תוכנות דומות, המפורסמת שבהן היא find my iphone, שבהן הורה יכול להיכנס ולראות היכן ילדו נמצא. אולם זה מצריך כניסה אקטיבית לאפליקציה למעקב, בעוד שהמוצר/פרויקט שלי, OMW , שולח הודעה במידה ויש יציאה מהמסלול (בהמשך ניתן יהיה לפתח שליחת הודעות במקרים נוספים כגון: עצירה ליותר מ X זמן וכדומה).

**אילו חידושים יש בפרויקט:**

החידושים הם שההורה לא צריך להיכנס גם הוא לאפליקציה, או לעקוב באופן אקטיבי אחר המסלול, כדי לראות תמונת מצב - אלא ההורה מקבל הודעה אוטומטית ברגע שילדו "יוצא" מהטווח שהוגדר (תישלח הודעה "EXIT" או "EXIT ZONE"). יתרון נוסף הוא שאם קורה משהו לילד הוא לא צריך באופן אקטיבי להתקשר או לשלוח הודעה להורה, וההודעה נשלחת באופן אוטומטי.

# **אתגרים מרכזיים**

**הבעיות איתן התמודדתי:**

במהלך הפרויקט נאצלתי להתמודד עם בעיות רבות ולמצוא להן פתרונות. האתגר הראשון והמשמעותי היה שלא היה לי שום ידע מקדים כיצד עלי להתחיל לבנות אפליקציה והיה עליי ללמוד את הנושא באופן עצמאי, בעיקר מהאינטרנט. בעיות נוספות היו במהלך הפרויקט, כאשר קוד שרשמתי לא עבד בצורה שהתכוונתי שיעבוד, או כשהאפליקציה קרסה והייתי צריך למצוא את הסיבה ולתקן את הבעיה.

**הסיבות לבחירת הנושא:**

כפי שכתבתי ברקע לפרויקט, בחרתי את הנושא משום שחשבתי שהוא יכול לתת מענה לצורך ממשי ובסיסי של דאגת הורים לילדיהם, ולתת פתרון לבעיה של אי שקט של הורה שילדו הולך לבד ליעד מסוים. רציתי ליצור מוצר שנותן ערך מוסף, ויהיה שימושי ומועיל בחיי היום יום לקהל רחב של אנשים (הורים וילדים), בכל העולם.

# **מוטיבציה לעבודה:**

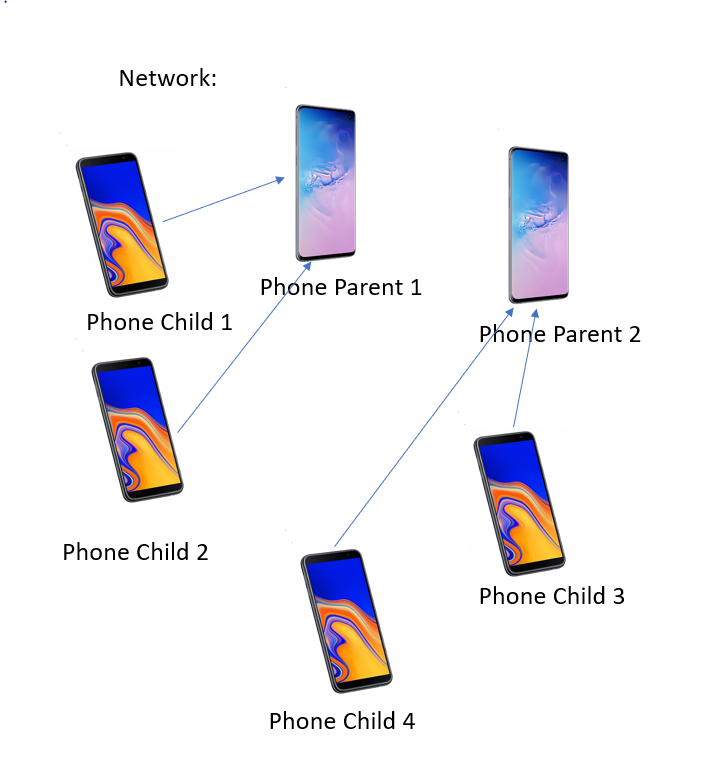
המוטיבציה שלי לעבודה הייתה שתמיד התעניינתי בבניית אפליקציות, ובכללי מעניין אותי ללמוד נושאים חדשים. המוטיבציה שלי גם היתה לראות שאני מסוגל ליצור מוצר חדש, במיוחד שמדובר במוצר שלדעתי יכול לעזור לאנשים.

# **הצורך עליו הפרויקט עונה:**

הפרויקט עונה על הצורך של כל הורה לדעת שילדו הגיע בביטחה ליעדו, כשהוא הולך לבד, בין אם ביום או בלילה. הוא עונה על הצורך החזק הזה על ידי שליחת הודעה כאשר הילד יוצא מהאזור שאמור היה להימצא בו. במקום שהורה יתקשר מספר פעמים לדרוש לשלומו של הילד,

הוא יכול להשתמש באפליקציה ולהיות "רגוע", שאם הילד יוצא מהמסלול תישלח אליו התראה אוטומטית. הפרויקט גם עונה על הצורך של ילד/נער בביטחון, שיכול לדעת שאם מסיבה שלילית (שהוא לא הודיע עליה) הוא יוצא מהאזור בו תכנן לשהות/ללכת, ההורה שלו יידע על כך מידית ויוכל לעזור.

**קהל היעד** הינו הורים לילדים/נוער (מגיל שהילדים הולכים לבד ברגל), וכמו כן הילדים/נוער עצמם.

**תרשים הפרויקט בכלליות** (הודעות כניסה/יציאה/אירועים נשלחות אוטומטית ב SMS מהטלפון של הילד להורה)

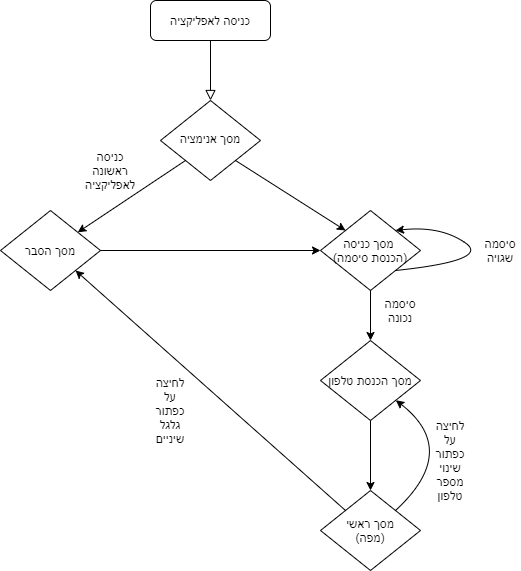
# **מבנה/ארכיטקטורה של הפרויקט**

# **הצגת הפתרון המוצע והסיבות לבחירתו**

כאשר ניגשתי להתחיל לעבוד על הפרויקט עמדה בפניי החלטה האם ליישם את הרעיון שלי באפליקציה או באתר. לאחר מחקר רב הגעתי למסקנה שהדרך הטובה ביותר ליישם את הרעיון שלי לפרויקט היא באפליקציה ולא אתר.

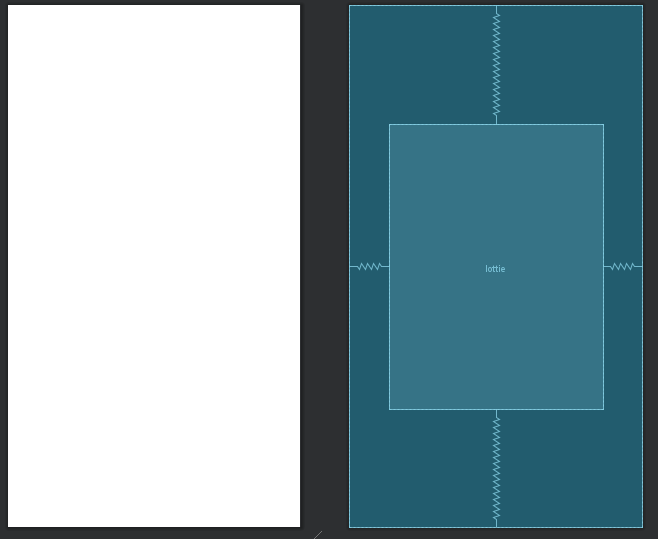
חלק חשוב בתכנון האפליקציה היא פשטות השימוש ומחשבה ותכנון רב הושקעו בהפשטת המערכת ויצירת מסכים קלים להבנה כך שכל אדם עם מעט יכולות טכנולוגיות לדוגמא כל סבתא תוכל להשתמש באפליקציה לשמירה על נכדיה.

# **תרשים יחידות**

****

**מסך האנימציה – splash**

מסך זה כלול את אנימציה קצרה המהווה שער כניסה לאפליקציה, במסך זה מתבצעת הבדיקה האם זו כניסה ראשונה לאפליקציה

****

**מסך הכניסה – login**

בכל כניסה לאחר הכניסה הראשונה מסך זה יופיע מיד אחרי מסך האנימציה, במסך זה האפליקציה בודקת את זהות המשתמש ע"י בדיקת הסיסמא שהכניס המשתמש (אם זו כניסה ראשונה לאפליקציה זו תהיה הגדרת הסיסמא) במקרה בו זו אינה הכניסה הראשונה לאפליקציה תבוצע בדיקה של הסיסמא ובמקרה של הקלדת סיסמא שגויה תתקבל הודעה מתאימה, במידה והסיסמא נכונה האפליקציה תעבור למסך הראשי – מסך המפה

במקרה של כניסה ראשונה תעבור למסך הכנסת מספר טלפון- מסךDATA

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**מסך הכנסת מספר טלפון - מסךDATA**

במסך זה ההורה מכניס את מספר הטלפון שלו (למספר טלפון זה ישלחו ההתראות)

**תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי**

**מסך הסבר – EXPLANATION**

למסך זה ניתן להגיע בשתי דרכים :

במהלך הכניסה הראשונה לאפליקציה זה חלק מהניתוב ההכרחי בכדי לוודא שהמשתמש הבין את האפליקציה.

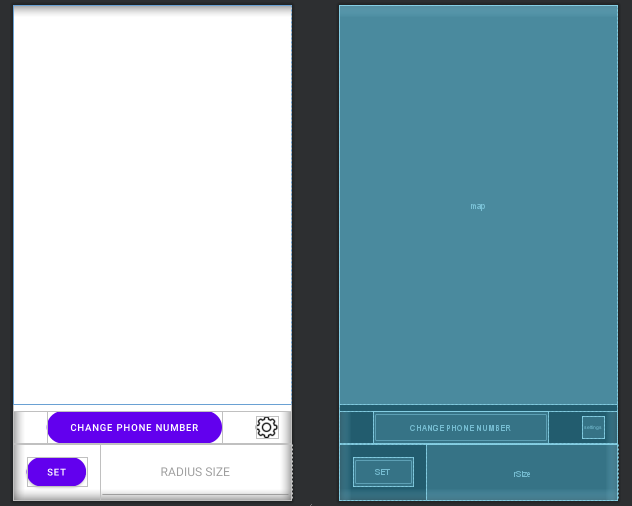
אפשרות נוספת להגיע למסך ההסבר היא סלקטיבית – במסך המפה ניתן ללחוץ על כפתור גלגל שיניים ולהגיע למסך ההסבר

מסך ההסבר כולל הסבר כללי על האפליקציה , השימושים באפליקציה ומטרת המערכת.

**מסך המפה – MAPS**

במסך זה , המסך הראשי באפליקציה, קובע המשתמש את גודל השטח בו יאפשר לילד לשהות, בכל מקרה של כניסה לאזור המוגדר או יציאה מהאזור המוגדר תשלח הודעת SMS למספר הטלפון שהוכנס (ראה בהמשך מסך מידע – DATA) .

אני משתמש במפה של גוגל, היתרון מבחינת המשתמש הוא המוכרות של המפה ומאפשר לשפר את קלות השימוש.



# 

# **באילו טכנולוגיות נעשה שימוש בפרויקט**

* **Android Studio** – סביבת הפיתוח לכתיבה ועריכת קוד
* **XML** – שפת תכנות בה השתמשתי לעיצוב המסכים בפרויקט
* **Java** – שפת התכנות המרכזית בה נכתב הקוד המעשי הפרויקט.
* **Google Play Services** – ספריית API של Google המאפשרת שימוש ב **Google Maps**
* **Geofence -**תחימת אזור על המפה

# **ספריות לקוד Java באנדרואיד סטודיו:**

* Pending intent – מתן אישור לאפליקציה לבצע פעולות במקום המשתמש
* com.google.android.gms.maps -שימוש לטובת עריכת אלמנטים במפה
* com.google.android.gms.tasks- משמש להבנת הצלחה או אי-הצלחה של פעולות המשתמש באפליקציה
* com.google.android.gms.location- שימוש במיקום על המפה וזיהוי מיקום המשתמש
* android.content - להעברת מידע בין מסכים שונים באפליקציה

# **מדריך למשתמש**

# **הוראות התקנה**

**האפליקציה תואמת ניידים עם מערכת הפעלה Android בלבד**

1. בקישור זה נמצא קובץ APK של האפליקציה, קובץ זה ניתן להורדה ישירות בנייד.
2. כמוכן, נמצא בקישור הקוד לפרויקט ובהתאם דרך נוספת להפעלת האפליקציה הינה ע"י הורדת ANDROID STUDIO במחשב והרצת קוד הפרויקט בסימולטור טלפון הקיים בסביבת עבודה זו

תמונה של כפתור הפעלת הסימולטור

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

1. דרך נוספת להוריד את האפליקציה לטלפון היא ע"י העברת הטלפון ל"מצב מפתח" את זאת ניתן לעשות ע"י כניסה להגדרות, לאחר מכן יש ללחוץ על אודות מכשיר ואז ללחוץ על build number מספר פעמים (עד לקבלת הודעת מעבר ל"מצב מפתח"). להוריד למחשב את סביבת העבודה android studio ולפתוח את קובץ הפרויקט. ואז יש לחבר את הטלפון בכבל USB וללחוץ על כפתור הפעלת הסימולטור שיביא להרצת האפליקציה בטלפון.

# **היררכיית המסכים והמעברים ביניהם**

היררכית המסכים שונה בין כניסה ראשונה לכניסות הבאות לאפליקציה, המיון הזה מתבצע במסך הראשון מסך האנימציה, להלן תיאור היררכיית המסכים והמעבר ביניהם בשתי האפשרויות:

1. כניסה ראשונה לאפליקציה :

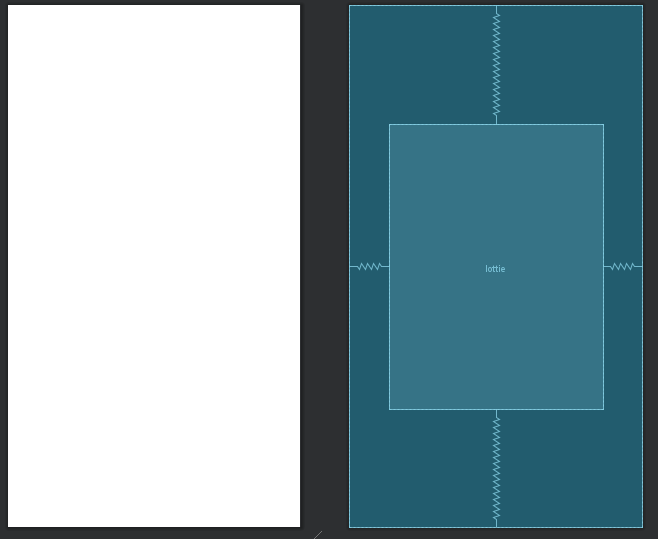
* כאשר משתמש מוריד את האפליקציה ומבצע כניסה ראשונה לאפליקציה, ראשית יופיע מסך הסבר-explanation הכולל הסבר כללי על האפליקציה ומטרתה וכן פירוט של שלבי השימוש ומטרת כל שלב, לאחר שהמשתמש קרא והבין את השלבים הוא מאשר על ידי לחיצה על כפתור "הבנתי" הלחיצה תעביר את המשתמש למסך הבא- מסך הזנת סיסמא.
* במסך הזנת סיסמא – login מזין המשתמש סיסמא ועובר למסך הבא .
* מסך הזנת מספר טלפון- data. במסך זה המשתמש יכניס את מספר הטלפון אליו ישלחו ההתראות, ואז מעבר למסך הבא.
* מסך ראשי – maps , ממסך זה ניתן לעבור למסך הdata בכדי לשנות את מספר הטלפון שהוזן או לעבור למסך ה explanation.

1. כניסה שניה ואילך לאפליקציה:

* המסך הראשון הוא מסך הזנת הסיסמא – login אם הסיסמא נכונה יש מעבר למסך הבא maps אם אינה נכונה תתקבל הודעת שגיאה.
* ממסך ה maps ניתן לעבור למסך הdata או למסך ה explanation

# **תיאור מסכי האתר**

**מסך האנימציה – splash**

****

מסך זה כולל אנימציה קצרה המהווה שער כניסה לאפליקציה, במסך זה מתבצעת הבדיקה האם זו כניסה ראשונה לאפליקציה.

**מסך הסבר – EXPLANATION**

**תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי**

לאחר קריאת ההסבר יש ללחוץ על כפתור "הבנתי " בכדי לעבור למסך הבא.

**מסך הכניסה – login**

**תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי**

יש לבחור סיסמא במידה וזו הכניסה הראשונה אם לא יש להזין את הסיסמה שנבחרה בכניסה הראשונה.

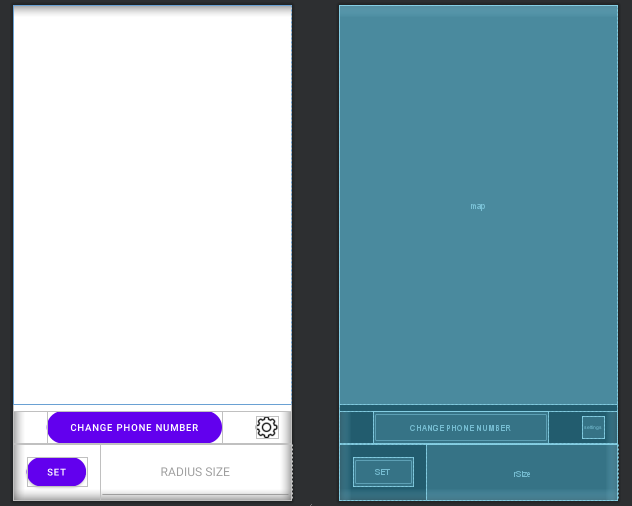
**מסך הכנסת מספר טלפון - מסךDATA**

**תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי**

במסך זה יש להכניס את מספר הטלפון אליו ישלחו התראות, למספר טלפון זה ישלחו ההתראות. לחיצה על הכפתור תקבע את מספר הטלפון ותעביר את המשתמש למסך הבא

**מסך המפה – MAPS**



במסך זה קובע המשתמש את האזור בו יאפשר לילד לשהות ע"י בחירה של מיקום וגודל רדיוס מסביבו. לחיצה על כפתור set תקבע את גודל הרדיוס. לחיצה על כפתור גלגל השיניים תוביל למסך ההסבר, ולחיצה על כפתור change phone number תוביל למסך הכנסת מספר הטלפון ותאפשר למשתמש לשנות את מספר הטלפון שהזין.

# **מדריך למפתח**

# **קישור לקוד המלא בGitHub**

<https://github.com/roys10/ProjectOMW>

# **הסבר כללי על קוד הפרויקט**

הקוד שכתבתי בפרויקט הוא קוד בשפת JAVA בסביבת העבודה אנדרואיד סטודיו. בעמודים הבאים אני אסביר ואצרף את המחלקות והפעולות העיקריות בפרויקט.

# **הסבר על קבצים, מחלקות ופעולות חשובות**

**SplashActivity**

public class SplashActivity extends AppCompatActivity {  
 public void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_splash*);  
 SharedPreferences preferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 String password = preferences.getString("password", "");  
 new Handler().postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 //first entry  
 if (password.equals("")) {  
 Intent intent = new Intent(SplashActivity.this, ExplanationActivity.class);  
 startActivity(intent);  
 finish();  
 }  
 else{  
 Intent intent = new Intent(SplashActivity.this, LoginActivity.class);  
 startActivity(intent);  
 finish();  
  
 }  
 }  
 }, 5000);  
  
 }  
  
}

במסך זה יש אנימציה קצרה אך גם מתבצעת הבדיקה האם זו הכניסה הראשונה לאפליקציה – אם כבר הוכנסה סיסמה ניתן להבין שזו לא הכניסה הראשונה לאפליקציה אך אם לא הוכנסה סיסמה ניתן להבין שזוהי הכניסה הראשונה.

**LoginActivity**

public class LoginActivity extends AppCompatActivity {  
 private Button button;  
 private EditText editText;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_login*);  
 button = findViewById(R.id.*button*);  
 editText = findViewById(R.id.*editText*);  
  
 button.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 checkPassword();  
 }  
 });  
 }  
  
 public void checkPassword() {  
 SharedPreferences preferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 SharedPreferences.Editor editor = preferences.edit();  
 String password = preferences.getString("password", "");  
 if (editText.getText().toString().equals("")) {  
 Toast.*makeText*(this, "Please Enter Password", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 //first entry  
 if (password.equals("")) {  
 editor.putString("password", editText.getText().toString());  
 editor.apply();  
 Toast.*makeText*(this, "PASSWORD SET", Toast.*LENGTH\_SHORT*).show();  
 Intent intent = new Intent(this, DataActivity.class);  
 startActivity(intent);  
 finish();  
 } else {  
 if (editText.getText().toString().equals(password)) {  
 Intent intent = new Intent(this, MapsActivity.class);  
 startActivity(intent);  
 finish();  
 } else {  
 Toast.*makeText*(this, "Wrong password! Please try again", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 }  
 }  
}

פעולת checkpassword – אם אין סיסמה הסיסמה שהוזנה נשמרת ואם יש מתבצעת בדיקה האם הסיסמה נכונה.

**DataActivity**

protected void onCreate(Bundle savedInstanceState) {  
  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_data*);  
 editTextPhone = findViewById(R.id.*editTextPhone*);  
 setButton = findViewById(R.id.*phoneButton*);  
 SharedPreferences sharedPreferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 String Phone = sharedPreferences.getString("phoneNum", "Phone Number");  
 if(Phone.equals("Phone Number")){  
 editTextPhone.setHint(Phone);  
 }  
 else{  
 editTextPhone.setText(Phone);  
 }  
 setButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 if (checkPermission(Manifest.permission.*SEND\_SMS*)) {  
 setPhoneNum();  
 } else {  
 ActivityCompat.*requestPermissions*(DataActivity.this,  
 new String[]{Manifest.permission.*SEND\_SMS*},  
 SEND\_SMS\_REQUEST\_CODE);  
 }  
 }  
 });  
}  
  
private void setPhoneNum() {  
 if(editTextPhone.getText().toString().length()!=10){  
 Toast.*makeText*(this, "Please Enter A Valid Phone Number", Toast.*LENGTH\_SHORT*).show();  
 }  
 else{  
 SharedPreferences sharedPreferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 SharedPreferences.Editor editor = sharedPreferences.edit();  
 editor.putString("phoneNum", editTextPhone.getText().toString());  
 editor.apply();  
 Intent intent = new Intent(this, MapsActivity.class);  
 startActivity(intent);  
 finish();  
 }  
}

פעולת onCreate היא הפעולה הראשית שמתרחשת כאשר המשתמש מגיע למסך ה data. פעולת setPhoneNum היא פעולה המתרחשת כאשר מתבצעת לחיצה על כפתור set . קודם

כל הפעולה בודקת האם הוכנס מספר טלפון תקין, אם כן, תשמור את מספר בטלפון שהוזן ותעבור למסך הMaps .

**ExplanationActivity**

public class ExplanationActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_explanation*);  
 Button exit = findViewById(R.id.*exit*);  
 exit.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(ExplanationActivity.this, LoginActivity.class);  
 startActivity(intent);  
 finish();  
 }  
 });  
 }  
}

כל מה שמתבצע כאן הוא שבלחיצה על הכפתור יש מעבר למסך Login.

**MapsActivity**

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback, GoogleMap.OnMapLongClickListener, LocationListener {  
 public static LatLng *latLng*;  
 private final int FINE\_LOCATION\_ACCESS\_REQUEST\_CODE = 10001;  
 private final int BACKGROUND\_LOCATION\_ACCESS\_REQUEST\_CODE = 10002;  
 private final String GEO\_ID = "GEOFENCE ID";  
 private GoogleMap mMap;  
 private LocationManager locationManager;  
 private static final long *MIN\_TIME* = 400;  
 private static final float *MIN\_DISTANCE* = 1000;  
 private GeofencingClient geofencingClient;  
 private GeofenceHelper geofenceHelper;  
 private float RADIUS = 0;  
 private Button button;  
 private ImageButton settings;  
 private EditText editText;  
 private Button setButton;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_maps*);  
 // Obtain the SupportMapFragment and get notified when the map is ready to be used.  
 SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()  
 .findFragmentById(R.id.*map*);  
 assert mapFragment != null;  
 mapFragment.getMapAsync(this);  
 geofencingClient = LocationServices.*getGeofencingClient*(this);  
 geofenceHelper = new GeofenceHelper(this);  
 locationManager = (LocationManager) getSystemService(Context.*LOCATION\_SERVICE*);  
 if (ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED* && ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_COARSE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 // *TODO: Consider calling* // *ActivityCompat#requestPermissions* // here to request the missing permissions, and then overriding  
 // public void onRequestPermissionsResult(int requestCode, String[] permissions,  
 // int[] grantResults)  
 // to handle the case where the user grants the permission. See the documentation  
 // for ActivityCompat#requestPermissions for more details.  
 return;  
 }  
 locationManager.requestLocationUpdates(LocationManager.*GPS\_PROVIDER*, *MIN\_TIME*, *MIN\_DISTANCE*, this); //You can also use LocationManager.GPS\_PROVIDER and LocationManager.PASSIVE\_PROVIDER  
 SharedPreferences preferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
  
 editText = findViewById(R.id.*rSize*);  
 RADIUS = preferences.getFloat("size", 0);  
 if (RADIUS == 0) {  
 editText.setHint("RADIUS SIZE");  
 } else {  
 editText.setText(String.*valueOf*(RADIUS));  
 }  
  
 settings = findViewById(R.id.*settings*);  
 settings.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(MapsActivity.this, ExplanationActivity.class);  
 startActivity(intent);  
 }  
 });  
 setButton = findViewById(R.id.*setSize*);  
 setButton.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 setRSize();  
 }  
 });  
 button = findViewById(R.id.*button*);  
 button.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 openMenu();  
 }  
 });  
 }  
  
 public void openMenu() {  
 Intent intent = new Intent(this, DataActivity.class);  
 startActivity(intent);  
 }  
  
 */\*\*  
 \* Manipulates the map once available.  
 \* This callback is triggered when the map is ready to be used.  
 \* This is where we can add markers or lines, add listeners or move the camera. In this case,  
 \* we just add a marker near Sydney, Australia.  
 \* If Google Play services is not installed on the device, the user will be prompted to install  
 \* it inside the SupportMapFragment. This method will only be triggered once the user has  
 \* installed Google Play services and returned to the app.  
 \*/* @Override  
 public void onLocationChanged(Location location) {  
 *latLng* = new LatLng(location.getLatitude(), location.getLongitude());  
 CameraUpdate cameraUpdate = CameraUpdateFactory.*newLatLngZoom*(*latLng*, 10);  
 mMap.animateCamera(cameraUpdate);  
 mMap.moveCamera(CameraUpdateFactory.*newLatLngZoom*(*latLng*, 16));  
 locationManager.removeUpdates(this);  
 SharedPreferences preferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 LatLng latLng2 = new LatLng(Double.*parseDouble*(preferences.getString("lat","0")), Double.*parseDouble*(preferences.getString("lng","0")));  
 handleMapLongClick(latLng2);  
 }  
  
  
  
 @Override  
 //חסר לי פתיחת אפליקציה על מיקום של משתמש  
 public void onMapReady(GoogleMap googleMap) {  
 mMap = googleMap;  
 mMap.getUiSettings().setZoomControlsEnabled(true);  
 // מזיז את המצלמה ועושה זום לשרונה  
 //LatLng sarona = new LatLng(32.0718, 34.7848);  
 //mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(sarona, 16));  
 enableUserLocation();  
 mMap.setOnMapLongClickListener(this);  
 }  
  
 //מפה גם מתעסק עם הרדיוס  
 //תקוע בלנסות לשמור על גודל הרדיוס עם יציאה מהאפליקציה  
 public void setRSize() {  
 SharedPreferences preferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 SharedPreferences.Editor editor = preferences.edit();  
 String rSize\_String = editText.getText().toString();  
 float rSize=0;  
 try{  
 rSize = Float.*parseFloat*(rSize\_String);  
 }  
 catch (Exception ignored){}  
 if (rSize <= 0) {  
 return;  
 }  
 editor.putFloat("size", rSize);  
 RADIUS = rSize;  
 editText.setText(rSize\_String);  
 editor.apply();  
 }  
  
 private void enableUserLocation() {  
 if (ContextCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) == PackageManager.*PERMISSION\_GRANTED*) {  
 mMap.setMyLocationEnabled(true);  
 } else {  
 //פה מבקשים רשות  
 ActivityCompat.*requestPermissions*(this, new String[]{Manifest.permission.*ACCESS\_FINE\_LOCATION*}, FINE\_LOCATION\_ACCESS\_REQUEST\_CODE);  
 }  
 }  
  
 @Override  
 public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {  
 super.onRequestPermissionsResult(requestCode, permissions, grantResults);  
 if (requestCode == FINE\_LOCATION\_ACCESS\_REQUEST\_CODE) {  
 if (grantResults.length > 0 && grantResults[0] == PackageManager.*PERMISSION\_GRANTED*) {  
 //יש אישור  
 if (ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED* && ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_COARSE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 // *TODO: Consider calling* // *ActivityCompat#requestPermissions* // here to request the missing permissions, and then overriding  
 // public void onRequestPermissionsResult(int requestCode, String[] permissions,  
 // int[] grantResults)  
 // to handle the case where the user grants the permission. See the documentation  
 // for ActivityCompat#requestPermissions for more details.  
 return;  
 }  
 mMap.setMyLocationEnabled(true);  
 }  
 }  
  
 if (requestCode == BACKGROUND\_LOCATION\_ACCESS\_REQUEST\_CODE) {  
 if (grantResults.length > 0 && grantResults[0] == PackageManager.*PERMISSION\_GRANTED*) {  
 //יש אישור  
 Toast.*makeText*(this, "you can add GeoFence", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 //אין אישור  
 Toast.*makeText*(this, "(background) location access is necessary", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 }  
  
 @Override  
 public void onMapLongClick(LatLng latLng) {  
 if (Build.VERSION.*SDK\_INT* >= 29) {  
 //אם גרסת אנרדואיד 10 או גבוה יותר יש צורך לבקש אישור לעבוד ברקע  
 if (ContextCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_BACKGROUND\_LOCATION*) == PackageManager.*PERMISSION\_GRANTED*) {  
 handleMapLongClick(latLng);  
 } else {  
 //לבדוק לעומק  
 ActivityCompat.*requestPermissions*(this, new String[]{Manifest.permission.*ACCESS\_BACKGROUND\_LOCATION*}, BACKGROUND\_LOCATION\_ACCESS\_REQUEST\_CODE);  
 }  
 } else {  
 handleMapLongClick(latLng);  
 }  
 }  
  
 private void handleMapLongClick(LatLng latLng) {  
 if (RADIUS == 0) {  
 Toast.*makeText*(this, "please set circle radius first", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 mMap.clear();  
 addMarker(latLng);  
 addCircle(latLng, RADIUS);  
 addGeofence(latLng, RADIUS);  
 SharedPreferences preferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 SharedPreferences.Editor editor = preferences.edit();  
 editor.putString("lat", String.*valueOf*(latLng.latitude));  
 editor.putString("lng", String.*valueOf*(latLng.longitude));  
 editor.apply();  
 }  
 }  
  
 private void addGeofence(LatLng latLng, float radius) {  
 Geofence geofence = geofenceHelper.getGeofence(GEO\_ID, latLng, radius, Geofence.*GEOFENCE\_TRANSITION\_ENTER* | Geofence.*GEOFENCE\_TRANSITION\_DWELL* | Geofence.*GEOFENCE\_TRANSITION\_EXIT*);  
 GeofencingRequest geofencingRequest = geofenceHelper.getGeofencingRequest(geofence);  
 PendingIntent pendingIntent = geofenceHelper.getPendingIntent();  
 if (ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 // *TODO: Consider calling* // *ActivityCompat#requestPermissions* // here to request the missing permissions, and then overriding  
 // public void onRequestPermissionsResult(int requestCode, String[] permissions,  
 // int[] grantResults)  
 // to handle the case where the user grants the permission. See the documentation  
 // for ActivityCompat#requestPermissions for more details.  
 return;  
 }  
 geofencingClient.addGeofences(geofencingRequest, pendingIntent).addOnSuccessListener(new OnSuccessListener<Void>() {  
 @Override  
 public void onSuccess(Void aVoid) {  
 Toast.*makeText*(MapsActivity.this, "onSuccess: Geofence added", Toast.*LENGTH\_SHORT*).show();  
 }  
 }).addOnFailureListener(new OnFailureListener() {  
 @Override  
 public void onFailure(@NonNull Exception e) {  
 String errorMessage = geofenceHelper.getErrorString(e);  
 Toast.*makeText*(MapsActivity.this, errorMessage, Toast.*LENGTH\_SHORT*).show();  
 }  
 });  
 }  
  
 //יוצר מרקר  
 public void addMarker(LatLng latLng) {  
 MarkerOptions markerOptions = new MarkerOptions().position(latLng);  
 mMap.addMarker(markerOptions);  
 }  
  
 //יוצר מעגל  
 public void addCircle(LatLng latLng, float radius) {  
 CircleOptions circleOptions = new CircleOptions();  
 circleOptions.center(latLng);  
 circleOptions.radius(radius);  
 circleOptions.strokeColor(Color.*argb*(255, 255, 0, 0));  
 circleOptions.fillColor(Color.*argb*(255, 255, 0, 0));  
 circleOptions.strokeWidth(4);  
 mMap.addCircle(circleOptions);  
 }  
}

מחלקה זו היא המלחקה הראשית בפרויקט בה נקבעים ומיושמים מספר דברים חשובים כמו: המפה, מיקום המשתמש, יצירת אזור השהייה (Geofence) והגדרת פעולות הכפתורים.

**GeofenceBroadcastReceiver**

public class GeofenceBroadcastReceiver extends BroadcastReceiver {  
 private static final String *TAG* = "GeofenceBroadcastReceiv";  
  
 @Override  
 public void onReceive(Context context, Intent intent) {  
 // *TODO: This method is called when the BroadcastReceiver is receiving* // an Intent broadcast.  
 // Toast.makeText(context, "Geofence triggered", Toast.LENGTH\_SHORT).show();  
  
 NotificationHelper notificationHelper = new NotificationHelper(context);  
  
 GeofencingEvent geofencingEvent = GeofencingEvent.*fromIntent*(intent);  
  
 if (geofencingEvent.hasError()) {  
 Log.*d*(*TAG*, "onReceive: Error receiving geofence");  
 return;  
 }  
  
 List<Geofence> geofenceList = geofencingEvent.getTriggeringGeofences();  
 for (Geofence geofence : geofenceList) {  
 Log.*d*(*TAG*, "onReceive: " + geofence.getRequestId());  
 }  
 //המקום שבו קורא הטריגר לגיאופנס (לא המרכז)  
 //Location location = geofencingEvent.getTriggeringLocation();  
 int transitionType = geofencingEvent.getGeofenceTransition();  
  
 switch (transitionType) {  
 case Geofence.*GEOFENCE\_TRANSITION\_ENTER*:  
 Toast.*makeText*(context, "GEOFENCE\_TRANSITION\_ENTER", Toast.*LENGTH\_SHORT*).show();  
 notificationHelper.sendSMS("user entered zone: "+*latLng*.latitude+" : "+*latLng*.longitude);  
 notificationHelper.sendHighPriorityNotification("ENTER", "", MapsActivity.class);  
  
 break;  
  
 case Geofence.*GEOFENCE\_TRANSITION\_EXIT*:  
 Toast.*makeText*(context, "GEOFENCE\_TRANSITION\_EXIT", Toast.*LENGTH\_SHORT*).show();  
 notificationHelper.sendSMS("user exited zone"+*latLng*.latitude+" : "+*latLng*.longitude);  
 notificationHelper.sendHighPriorityNotification("EXIT", "", MapsActivity.class);  
 break;  
 }  
 }  
}

בפעולה onReceive מתבצעת שליחת הSMS עם הודעה בהתאם לאירוע ונ.צ בעת התרחשות האירוע.

# **סיכום אישי / רפלקציה**

# **תחושותיי מהפרויקט**

אהבתי מאוד לעבוד על הפרויקט במהלך השנה, ונהניתי מאוד ללמוד נושא חדש לגמרי שלא היה מוכר לי כלל ואם זאת תמיד התעניינתי בו. נתקלתי בלא מעט קשיים בעשיית הפרויקט ולא הייתי בטוח האם אספיק להגיע ליעד שלי, העבודה חייבה אותי למסגרת זמן נוקשה ועמידה בלוחות הזמנים שקבעתי לעצמי.

כעת כשהאפליקציה עובדת באופן שדמיינתי אני חש סיפוק רב ותחושת הישג גם הפידבק שאני מקבל מהמתנסים באפליקציה חיובי וגורם לתחושת שביעות הרצון שלי, ההשקעה הרבה לא הייתה לשווא.

במבט לאחור אני שמח שהחלטתי להתגבר על הקשיים שהיו לי ולא לוותר למרות שהיו מספר נקודות משבר בהן הייתי קרוב להפסקת הפרויקט, התחושה של התגברות על המשברים מתגמלת ומוסיפה מאד.

# **מה קיבלתי וכלים שאקח איתי להמשך**

את הכלים שקבלתי אני מחלק לשני סוגים : כלים טכניים וכלים אישיים:

במסגרת הכלים הטכניים אני כולל את שיפור יכולות התכנות שלי, ההכרות של תחום טכנולוגי חדש שלא הייתה לי שום היכרות איתו – פיתוח אפליקציה.

לא פחות חשוב מהכלים הטכניים שצברתי אני רואה חשיבות רבה לשיפור היכולות האישיות טכניות : למדתי לנהל פרויקט אישי עצמאי, לפתח ולמקד את יכולת הלמידה העצמית שלי, לבנות לוח זמנים ולהתמודד עם החריגות , להבין שקשיים הם חלק מהדרך והיכולות להתמודד איתם תסייע לי בהמשך אולי אף מעבר ליכולות הטכניות שצברתי.

# **קשיים ואתגרים שעמדו בפני**

האתגר הראשון שעמד בפני היה גיבוש הקונספט לאפליקציה, בשלב הראשון האפליקציה הייתה מורכבת ומאתגרת למשתמש, הבנתי שחשוב ליצר חוויית משתמש פשוטה בכדי שכל הורה ירצה ויוכל להשתמש באפליקציה והשקעתי מחשבה ותכנון רב ביצירת קונספט מינימליסטי.

מאתגר במידה דומה היה האספקט המעשי שהציב בפני את הצורך להתמודד עם לימוד חומר חדש לפתח יכולות שלא היו לי טרם הפרויקט.

בנייה של לוח זמנים ועמידה בו היו אתגר משמעותי שחייב אותי למצוא גם את הזמן וגם את המיקוד והמשמעת הדרושים לעמידה במשימות הרבות .

# **מסקנותיי מהפרויקט**

המסקנה המשמעותית ביותר מבחינתי היא החשיבות של פרויקט כזה לכל תלמיד בתחום, הלימודים במהלך השנה בכיתה מביאים את התלמיד (אותי) לרמה מסוימת ורק כשבונים ומפתחים פרויקט עצמאי הן מחדדים את היכולות מהלימודים והן מפתחים יכולות חדשות.

מסקנה נוספת מבחינתי היא החשיבות של לוח הזמנים ועמידה בו – חשוב להדגיש זאת ביתר עוצמה לכל מי שמתחיל בפרויקט מסוג זה. ללא מחויבות מלאה ללוח הזמנים לא ניתן לבנות פרויקט באופן מסודר.

# **מה הייתי עושה אחרת לו הייתי מתחיל היום**

אני סבור שעבדתי בצורה טובה ויעילה , אם זאת השקעתי זמן רב בפרטים הקטנים במקום להשקיע זמן רב יותר בפיתוח רעיונות נוספים וחידושים לאפליקציה.

בנוסף הייתי מחלק את הפרויקט לתת משימות רבות יותר מעבר למועדי ההגשה של המורה, ויוצר לוח זמנים מפורט יותר ובכך הייתי מבטיח הימנעות מחריגות ומקטין את הלחץ לעמוד במשימות בזמן.

# **תכונות שהייתי רוצה להוסיף לפרויקט**

יש מספר תכונות שהייתי רוצה להוסיף לפרויקט:

* אפשרות לשנות את צורת האזור המוגדר – לא רק עיגול אלה יצירת מתחם גמיש שיאפשר למשתמש יותר דיוק בהגדרת האזור בו הילד יכול לשהות.
* אפשרות להוסיף הגדרת מסלול במקום אזור – אפשרות כזו הייתה מקנה לאפליקציה שימושים נוספים
* אפשרות להגדיר שעה ספציפית בה ישלח מיקום מדויק של הילד
* היית רוצה לחקור יותר את פונקציות העיצוב באנדרואיד סטודיו ובכך לשפר את העיצוב והוויזואליות של האפליקציה

# **ביבליוגרפיה**

**אתרים ומקורות מידע בהם השתמשתי:**

Stack Overflow

<https://stackoverflow.com/>

GeekForGeeks

<https://www.geeksforgeeks.org/>

GitHub

<https://github.com/>

Android for Developers

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

YouTube

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

draw.io

<https://drawio-app.com/>

# **נספחים**

# **נספח 1 – קישור לקוד המלא בGitHub**

<https://github.com/roys10/ProjectOMW>

# **נספח 2-קוד נוסף שלא הוספתי במדריך למפתח**

**NotificationHelper**

public class NotificationHelper extends ContextWrapper {  
  
 private static final String *TAG* = "NotificationHelper";  
 private final String CHANNEL\_NAME = "High priority channel";  
 private final String CHANNEL\_ID = "com.example.notifications" + CHANNEL\_NAME;  
 SharedPreferences sharedPreferences = getSharedPreferences("pref", *MODE\_PRIVATE*);  
 String phoneNum = sharedPreferences.getString("phoneNum", "");  
  
 public NotificationHelper(Context base) {  
 super(base);  
 if (Build.VERSION.*SDK\_INT* >= Build.VERSION\_CODES.*O*) {  
 createChannels();  
 }  
 }  
  
 @RequiresApi(api = Build.VERSION\_CODES.*O*)  
 private void createChannels() {  
 NotificationChannel notificationChannel = new NotificationChannel(CHANNEL\_ID, CHANNEL\_NAME, NotificationManager.*IMPORTANCE\_HIGH*);  
 notificationChannel.enableLights(true);  
 notificationChannel.enableVibration(true);  
 notificationChannel.setDescription("this is the description of the channel.");  
 notificationChannel.setLightColor(Color.*RED*);  
 notificationChannel.setLockscreenVisibility(Notification.*VISIBILITY\_PUBLIC*);  
 NotificationManager manager = (NotificationManager) getSystemService(Context.*NOTIFICATION\_SERVICE*);  
 manager.createNotificationChannel(notificationChannel);  
 }  
  
 public void sendHighPriorityNotification(String title, String body, Class activityName) {  
  
 Intent intent = new Intent(this, activityName);  
 PendingIntent pendingIntent = PendingIntent.*getActivity*(this, 267, intent, PendingIntent.*FLAG\_UPDATE\_CURRENT*);  
  
 Notification notification = new NotificationCompat.Builder(this, CHANNEL\_ID)  
// .setContentTitle(title)  
// .setContentText(body)  
 .setSmallIcon(R.drawable.*ic\_launcher\_background*)  
 .setPriority(NotificationCompat.*PRIORITY\_HIGH*)  
 .setStyle(new NotificationCompat.BigTextStyle().setSummaryText("summary").setBigContentTitle(title).bigText(body))  
 .setContentIntent(pendingIntent)  
 .setAutoCancel(true)  
 .build();  
 NotificationManagerCompat.*from*(this).notify(new Random().nextInt(), notification);  
 }  
  
 public void sendSMS(String message) {  
 if (phoneNum.length() != 10) {  
 Toast.*makeText*(this, "Please enter valid phone number!", Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
  
 if (checkPermission(Manifest.permission.*SEND\_SMS*)) {  
 SmsManager smsManager = SmsManager.*getDefault*();  
 smsManager.sendTextMessage(phoneNum, null, message, null, null);  
 Toast.*makeText*(this, "Message sent", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 Toast.*makeText*(this, "Failed to send", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
  
 public boolean checkPermission(String permission) {  
 int check = ContextCompat.*checkSelfPermission*(this, permission);  
 return (check == PackageManager.*PERMISSION\_GRANTED*);  
 }  
}

**GeofenceHelper**

public class GeofenceHelper extends ContextWrapper {  
 private static final String *TAG* = "GeofenceHelper";  
 PendingIntent pi;  
  
 public GeofenceHelper(Context base) {  
 super(base);  
 }  
  
 public GeofencingRequest getGeofencingRequest(Geofence geofence) {  
 return new GeofencingRequest.Builder().addGeofence(geofence).setInitialTrigger(GeofencingRequest.*INITIAL\_TRIGGER\_ENTER*).build();  
 }  
  
 public Geofence getGeofence(String id, LatLng latlng, float radius, int transitionTypes) {  
  
 return new Geofence.Builder().setCircularRegion(latlng.latitude, latlng.longitude, radius).setRequestId(id).setTransitionTypes(transitionTypes).setLoiteringDelay(5000).setExpirationDuration(Geofence.*NEVER\_EXPIRE*).build();  
 }  
  
 public PendingIntent getPendingIntent() {  
 if (pi != null) {  
 return pi;  
 }  
 Intent intent = new Intent(this, GeofenceBroadcastReceiver.class);  
 pi = PendingIntent.*getBroadcast*(this, 2607, intent, PendingIntent.*FLAG\_UPDATE\_CURRENT*);  
 return pi;  
 }  
  
 public String getErrorString(Exception e) {  
 if (e instanceof ApiException) {  
 ApiException apiException = (ApiException) e;  
 switch (apiException.getStatusCode()) {  
 case GeofenceStatusCodes  
 .*GEOFENCE\_NOT\_AVAILABLE*:  
 return "GEOFENCE\_NOT\_AVAILABLE";  
 case GeofenceStatusCodes  
 .*GEOFENCE\_TOO\_MANY\_GEOFENCES*:  
 return "GEOFENCE\_TOO\_MANY\_GEOFENCES";  
 case GeofenceStatusCodes  
 .*GEOFENCE\_TOO\_MANY\_PENDING\_INTENTS*:  
 return "GEOFENCE\_TOO\_MANY\_PENDING\_INTENTS";  
 }  
 }  
 return e.getLocalizedMessage();  
 }  
}

**GeofenceBroadcastReciver**

public class GeofenceBroadcastReceiver extends BroadcastReceiver {  
 private static final String *TAG* = "GeofenceBroadcastReceiv";  
  
 @Override  
 public void onReceive(Context context, Intent intent) {  
 // *TODO: This method is called when the BroadcastReceiver is receiving* // an Intent broadcast.  
 // Toast.makeText(context, "Geofence triggered", Toast.LENGTH\_SHORT).show();  
  
 NotificationHelper notificationHelper = new NotificationHelper(context);  
  
 GeofencingEvent geofencingEvent = GeofencingEvent.*fromIntent*(intent);  
  
 if (geofencingEvent.hasError()) {  
 Log.*d*(*TAG*, "onReceive: Error receiving geofence");  
 return;  
 }  
  
 List<Geofence> geofenceList = geofencingEvent.getTriggeringGeofences();  
 for (Geofence geofence : geofenceList) {  
 Log.*d*(*TAG*, "onReceive: " + geofence.getRequestId());  
 }  
 //המקום שבו קורא הטריגר לגיאופנס (לא המרכז)  
 //Location location = geofencingEvent.getTriggeringLocation();  
 int transitionType = geofencingEvent.getGeofenceTransition();  
  
 switch (transitionType) {  
 case Geofence.*GEOFENCE\_TRANSITION\_ENTER*:  
 Toast.*makeText*(context, "GEOFENCE\_TRANSITION\_ENTER", Toast.*LENGTH\_SHORT*).show();  
 notificationHelper.sendSMS("user entered zone: "+*latLng*.latitude+" : "+*latLng*.longitude);  
 notificationHelper.sendHighPriorityNotification("ENTER", "", MapsActivity.class);  
  
 break;  
  
 case Geofence.*GEOFENCE\_TRANSITION\_EXIT*:  
 Toast.*makeText*(context, "GEOFENCE\_TRANSITION\_EXIT", Toast.*LENGTH\_SHORT*).show();  
 notificationHelper.sendSMS("user exited zone"+*latLng*.latitude+" : "+*latLng*.longitude);  
 notificationHelper.sendHighPriorityNotification("EXIT", "", MapsActivity.class);  
 break;  
 }  
 }  
}

**קוד עיצובי (XML):**

**activity\_data.xml**

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout 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"  
 tools:context=".DataActivity">  
  
 <TextView  
 android:id="@+id/b"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="100dp"  
 android:gravity="center"  
 android:text="Please enter parent's phone number"  
 android:textColor="@color/black"  
 android:textSize="22sp" />  
  
 <EditText  
 android:id="@+id/editTextPhone"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="50dp"  
 android:layout\_marginEnd="50dp"  
 android:layout\_centerVertical="true"  
 android:gravity="center"  
 android:hint="Phone Number"  
 android:inputType="phone"  
 android:textSize="25sp"/>  
  
 <Button  
 android:id="@+id/phoneButton"  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/editTextPhone"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="40dp"  
 android:background="@drawable/rounded\_button"  
 android:text="SET" />  
</RelativeLayout>

**activity\_maps.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:orientation="vertical"  
 tools:context=".MapsActivity">  
 <androidx.fragment.app.FragmentContainerView  
 android:id="@+id/map"  
 android:name="com.google.android.gms.maps.SupportMapFragment"  
 android:layout\_width="match\_parent"  
 android:layout\_height="590dp" />  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:layout\_marginTop="10dp">  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="Change phone number"  
 android:textColor="@color/white"  
 android:background="@drawable/rounded\_button"  
 android:layout\_marginEnd="50dp"  
 android:layout\_marginStart="50dp"/>  
  
 <ImageButton  
 android:id="@+id/settings"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:src="@drawable/settings"  
 android:background="@color/white"  
 android:layout\_gravity="center"  
 android:layout\_marginEnd="20dp"  
 android:layout\_marginStart="0dp"  
 android:layout\_marginLeft="0dp"  
 android:layout\_marginRight="20dp" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/setSize"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:text="SET"  
 android:textColor="@color/white"  
 android:layout\_margin="20dp"  
 android:background="@drawable/rounded\_button"  
 />  
 <EditText  
 android:id="@+id/rSize"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:gravity="center"  
 android:hint="RADIUS SIZE"  
 android:inputType="number" />  
 </LinearLayout>  
</LinearLayout>

**activity\_explanation.xml**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".ExplanationActivity">  
  
  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fontFamily="sans-serif-medium"  
 android:gravity="center"  
 android:text="שלום וברוכים הבאים לOMW !\n  
האפליקציה שתעזור לשמור על ילדיכם בעזרת קביעת אזור בטוח וקבלת התראות על כניסה או יציאה ממנו. השימוש באפליקציה פשוט מאוד וכולל 3 שלבים בלבד: \n שלב ראשון הוא הכנסת הסיסמה שתשמש לזיהויכם בכניסות הבאות. \n שלב שני הוא הכנסת מספר הטלפון אליו אתם רוצים שישלחו התראות. \n שלב שלישי הוא הגדרת האזור בו תרצו שילדכם ישהה על ידי בחירה של נקודה על המפה ורדיוס (במטרים) מסביבה, אם הילד יצא או יכנס תשלח התראה מתאימה ונ.צ המייצג את מיקום הילד בעת האירוע, את הנ.צ תוכלו להזין בכל מפה לדוגמה google maps ולקבל מיקום."  
 android:textColor="@color/black"  
 android:textSize="18sp"  
 tools:layout\_editor\_absoluteX="0dp"  
 tools:layout\_editor\_absoluteY="-36dp" />  
  
 <Button  
 android:id="@+id/exit"  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:background="@drawable/rounded\_button"  
 android:text="הבנתי"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.497"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.849" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**activity\_login.xml**

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout 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"  
 tools:context=".LoginActivity">  
  
 <TextView  
 android:id="@+id/b"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="100dp"  
 android:gravity="center"  
 android:text="Please enter your password"  
 android:textColor="@color/black"  
 android:textSize="22sp" />  
  
 <EditText  
 android:id="@+id/editText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="50dp"  
 android:layout\_marginEnd="50dp"  
 android:layout\_centerVertical="true"  
 android:gravity="center"  
 android:hint="Password"  
 android:inputType="textPassword"  
 android:textSize="25sp" />  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="200dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/editText"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="40dp"  
 android:background="@drawable/rounded\_button"  
 android:text="Enter"/>  
</RelativeLayout>

**activity\_splash.xml**

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 xmlns:app="http://schemas.android.com/apk/res-auto">  
 <com.airbnb.lottie.LottieAnimationView  
 android:id="@+id/lottie"  
 android:layout\_width="300dp"  
 android:layout\_height="400dp"  
 app:lottie\_autoPlay="true"  
 app:lottie\_loop="true"  
 app:lottie\_rawRes="@raw/anim"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"/>  
</RelativeLayout>

**Manifest**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.omw">  
 <!--  
 The ACCESS\_COARSE/FINE\_LOCATION permissions are not required to use  
 Google Maps Android API v2, but you must specify either coarse or fine  
 location permissions for the "MyLocation" functionality.  
 --><uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION"/>  
 <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION"/>  
 <uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />  
 <uses-permission android:name="android.permission.ACCESS\_BACKGROUND\_LOCATION" />  
 <uses-permission android:name="android.permission.SEND\_SMS" />  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/AppTheme">  
 <activity android:name=".ExplanationActivity"/>  
 <activity android:name=".LoginActivity"/>  
 <activity android:name=".DataActivity" />  
 <activity android:name=".SplashActivity">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 <receiver  
 android:name=".GeofenceBroadcastReceiver"  
 android:enabled="true"  
 android:exported="true" />  
 <!--  
 The API key for Google Maps-based APIs is defined as a string resource.  
 (See the file "res/values/google\_maps\_api.xml").  
 Note that the API key is linked to the encryption key used to sign the APK.  
 You need a different API key for each encryption key, including the release key that is used to  
 sign the APK for publishing.  
 You can define the keys for the debug and release targets in src/debug/ and src/release/.  
 -->  
 <meta-data  
 android:name="com.google.android.geo.API\_KEY"  
 android:value="@string/google\_maps\_key" />  
  
 <activity  
 android:name=".MapsActivity"  
 android:label="@string/title\_activity\_maps"/>  
 </application>  
  
</manifest>

**-------------------------------**