

## תכנון ותכנות מערכות טלפונים חכמים

### תחת מערכת ההפעלה אנדרואיד

תאריך: 1.9.2012

#### תוכן עניינים

##### הקדמה

##### מטרות ביצועיות

##### דרישות קדם

##### חלוקה השעות ושעות הפרויקט

##### כיתה י"א, פרקים

פרק 1 : סקירה, מה זה ה"אנדרואיד" Android?.

פרק 2 : תכנות מונחה עצמים בשפת JAVA חלק 1: הורשה, מחלקות מופשטת, ממשקים,

פולימורפיזם.9

פרק 3 : אוספים

פרק 4: חריגות

פרק 5 : מבוא ליישום/אפליקציה: פעילות/ Activity

פרק 6 : ממשק משתמש UI-User Interface: רכיבים ותבניות

פרק 7 : תכנות מונחה אירועים ושימוש במחלקה פנימית

פרק 8 : מסרים Intents.

פרק 9: רכיבים מתקדמים: תפריט, תיבת דו-שיח, הודעות Status bar וטיפול באירועים..

פרק 10 : משאבי האפליקציה Application Resources.

פרק 11 : אחסון נתונים Storage Data.

פרק 12 : תכנון וניתוח אפליקציות, ניתוח בשיטת UML, תאור מסכים, תרשים זרימה.

כיתה י"ב, פרקים

פרק 13 ספקי תוכן Content Providers.

פרק 14 : תהליכים ותהליכונים (Threads & Processes)

פרק 15 שירות Service.

פרק 16 : מקלט שידורים BroadCast Reciever ו App Widget.

פרק 17 : טלפוניה.

פרק 18 : רכיבים מתקדמים: מולטימדיה ומצלמה, חיישנים, GPS, מפות.

פרק 19 : גרפיקה ואנימציה.

פרק 20 : החיה והגשת עבודת גמר.

ההבדלים בין רמת העבודה 1,3,5 יח"ל:

פרויקט גמר ברמה של 1 יח"ל,

פרויקט גמר ברמה של 3 יח"ל,

עבודת גמר של 5 יח"ל,

משימות ולוח זמנים..

משימות ולוח זמנים לביצוע הפרויקט (י"ב)

גיליונות ההערכה בתכנון ותכנות מערכות טלפונים חכמים תחת מערכת ההפעלה אנדרואיד.

דברי הסבר.

גיליון הערכה לעבודת גמר 5 יח"ל, בתכנון ותכנות מערכות טלפונים חכמים תחת מערכת

ההפעלה אנדרואיד

גיליון הערכה לפרויקט/עבודת גמר 3 יח"ל, בתכנון ותכנות מערכות טלפונים חכמים תחת

מערכת ההפעלה אנדרואיד

גיליון הערכה לפרויקט/עבודת גמר 1 יח"ל, בתכנון ותכנות מערכות טלפונים חכמים תחת

מערכת ההפעלה אנדרואיד

מפרט הגשת פרויקט/עבודת גמר, תכנון ותכנות מערכות טלפונים חכמים תחת מערכת

ההפעלה אנדרואיד

### הקדמה:

החלופה תכנות תחת מערכת ההפעלה אנדרואיד באה לתת מענה במערכת החינוך הבית

ספרית לחידושים ולהתקדמות בעולם בתחום הטלפונים החכמים, ולחשוף תחום זה בפני

התלמידים מגיל צעיר, במדינה שכלכלתה נשענת על תעשיית ההי-טק.

הטלפון החכם היום נחשב לתחום הגדול ביותר בעולם, טלפון חכם שנחשב למחשב לכל

דבר, עם יכולות ומאפיינים העולים על המחשב הרגיל כמו התוספות הקשורות לטכנולוגית

התקשורת (GSM, WIFI, GPS,BT), חיישנים, מסך מגע ועוד.

מערכת ההפעלה אנדרואיד נבחרה להיות מערכת ההפעלה של מכשירים חכמים ע"י עשרות

חברות הי-טק, ומהן ענקיות הטכנולוגיה והתקשורת כמו Google, HTC, Intel, LG, Motorola,

Nvidia, Samsung, Sony EricssonToshiba, Vodafone, T-Mobile, שמאוגדים תחת ברית

Open Handset Alliance -OHA. בהובלת חברת Google.

אנדרואיד הינה מערכת הפעלה עם קוד פתוח שהליבה שלה הינה ליבת לינוקס. המכונה

הוירטואלית הייחודית שלה שנקראת ה-DALVIK הינה האחראית על ריצת יישומים שלה,

אותם יישומים נכתבים בשפת JAVA עם החבילות הייחודיות למערכת האנדרואיד. יישומים

אלו מאופיינים כ-"חבילת אנדרואיד" (Android Package) בקיצור APK. חברת גוגל בחרה

בסביבת הפיתוח ECLIPSE לסביבת פיתוח היישומים בשפת JAVA, כך שמפתחת כלים

ותוספים המיוחדים לסביבת פיתוח זו כגון ה ADT Plugin. גם כן ניתן לתכנת גם בסביבת  
NETBEANS עם התוסף המתאים לאנדרואיד.

### מטרות ביצועיות:

מטרת ההתמחות: להקנות לתלמידים כלים לפתח יישום בממשק משתמש גרפי בשפת  
Java, על פי עקרונות הנדסת תוכנה, הרצות על מכשירים חכמים וטאבליטים, תחת מערכת  
ההפעלה "אנדרואיד" של "גוגל".

- התלמיד יסביר את העקרונות המתקדמים בפיתוח תוכנה בשפת תכנות מונחת  
עצמים.
- התלמיד יפתח תכניות הכוללות תכנות מונחת אירועים.
- התלמיד יסביר ולהשתמש בתהליכים בתכנות מקבילי.
- התלמיד יסביר את מאפייני התוכנה שרצה על מכשירים חכמים.
- התלמיד יסביר ויפרש את הייחודיות של מערכת ההפעלה "אנדרואיד" של "גוגל"  
והמבנה שלה.
- התלמיד יסביר את עקרונות המכונה הוירטואלית ותפקידה וה- DAVLIK כדוגמא.
- התלמיד יכתוב ממשק משתמש הגרפי מתקדם בהיקשר של ה"אנדרואיד".
- התלמיד יפתח תכניות הכוללות שימוש בתכנות מונחה אירועים.
- התלמיד יזהה ויאפיין את הרכיבים המתקדמים כגון: מצלמה, מיקרופון, GPS, BT  
וחיישנים.
- התלמיד יכתוב אפליקציה בשפת ג'אווה למכשירים המופעלים ע"י מערכת ההפעלה  
"אנדרואיד" של "גוגל".

### דרישות קדם:

התלמיד סיים את שלוש יח"ל הלימוד במדעי המחשב "יסודות מדעי המחשב" בהצלחה  
(שאלון 899300). כלומר את הבחינה בכתב + בחינת המעבדה. 899222+899127.

## חלוקה השעות ושעות הפרויקט:

מעצם טבעה של תוכנית לימודים זו, נדרשת התנסות רבה של התלמיד בעבודה מעשית. חלקה המעשי של התוכנית הינה 67% מסך השעות הנלמדות בכיתות יא-יב. חלקה העיוני של התוכנית הינו 33% מסך השעות הנלמדות בכיתות יא-יב. שעות הלימוד העיוניות ושעות ההתנסות צריכות להיות משולבות זו בזו. שימו לב, כי הנושא "הנחיית פרויקט" נכלל בשעות ההתנסות. הערכה לביצוע פרויקט צריכה להיעשות ע"י מורה תוך כדי למידה של נושאי הלימוד השונים במסגרת שעות התנסות חלוקה מומלצת:

כיתה י"א – 180 שעות ( 6 ש"ש )

כיתה י"ב – 270 שעות ( 9 ש"ש )

אנו מאפשרים חלוקה פנימית של ביה"ס

### כיתה י"א, פרקים:

פרק מס'	נושא לימוד	שעות עיוני	שעות מעשי	סה"כ
1	סקירה: מה זה ה"אנדרואיד" Android?	1	2	3
2	תכנות מונחה עצמים בשפת JAVA חלק 1: הורשה, מחלקות מופשטת, ממשקים, פולימורפיזם.	12	24	36
3	אוספים	2	4	6
4	חריגות	1	2	3
5	מבוא ליישום/אפליקציה: פעילות / Activity	3	6	9
6	ממשק משתמש UI-User Interface: רכיבים ותבניות	10	20	30
7	טיפול אירועים ושימוש במחלקה פנימית	3	6	9
8	מסרים Intents	4	8	12
9	תפריט, תיבת דו-שיח, הודעות Status bar וטיפול באירועים	5	10	15
10	משאבי האפליקציה Application Resources	3	6	9
11	אחסון נתונים Data Storage	12	24	36
12	תכנון וניתוח אפליקציות, ניתוח בשיטת UML, תאור מסכים, תרשים זרימה.	4	8	12
	<b>סך הכול שעות לימוד</b>	<b>60</b>	<b>120</b>	<b>180</b>

## פרק 1 : סקירה, מה זה ה"אנדרואיד" Android?

**יעדים:** היכרות וסקירה כללית למערכת ההפעלה אנדרואיד.

### תכנים:

- **משמעות המילה אנדרואיד:** הוא יצור מלאכותי (בדרך כלל רובוט) הדומה בצורתו ובהתנהגותו לבן אדם. מקור המונח ביוניתי: "אנדרו" = גבר, "איד" = דמוי)
- היסטוריה:  
אנדרואיד הינה מערכת הפעלה מבוססת לינוקס המיועדת לטלפונים ניידים שפותחה ע"י חברה סטארט-אפ קטנה (Android inc), שמיקומה בפלורידה. נרכשה בשנת 2005 ע"י חברת גוגל, ב 5/11/2007 הוקמה ברית Open Handset Alliance -OHA שכללה חברות גדולות בעולם בתחום האלקטרוניקה, התקשורת והסילולר בהובלת חברת גוגל, אפשר למנות חלק מהן ( Google, HTC, Intel, LG, Motorola, Nvidia, ) ועוד עשרות אחרות. (Samsung, Sony Ericsson, Toshiba, Vodafone, T-Mobile)
- ארכיטקטורת אנדרואיד:  
האפליקציות Applications, מסגרת העבודה Framework, הספריית Libraries, מכונה וירטואלית Davlik VM, Runtime-Core Library, ליבת לינוקס Linux Kernel.
- ג'אווה והאנדרואיד.
- היישום באנדרואיד: קובץ ה APK.
- גרסאות וההבדלים המהותיים.
- אנדרואיד מרקט (Android Market).
- היכרות בהתקנה ושימוש ב Android SDK.
- התנסות במערכת ע"י הסימולאטור של אנדרואיד ה AVD-Android Virtual Device, ומערכת הניהול שלהם AVD Manager.

### מטרות ביצועיות:

- התלמיד יתאר מידע כללי על מערכת האנדרואיד.
- התלמיד יסביר את מרכיבי מערכת האנדרואיד.

- התלמיד יתקין את SDK Manager להתקנת מערכת האנדרואיד הגרסאות שונות.
- התלמיד יתנסה ביצירת סימולטור אנדרואיד ע"י AVD Manager.
- יתנסה במערכת האנדרואיד ע"י ה AVD.
- התלמיד יזהה את שוק היישומים של מערכת האנדרואיד .

#### דרכי הוראה:

- מטרת הפרק לתת סקירה כללית ואינפורמציה על מערכת ההפעלה אנדרואיד, אפשר להעבירה במצגת ובאתר האינטרנט הרשמי של אנדרואיד.

#### דרכי הערכה:

- בחינת בכתב - ידע כללי.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
סקירה כללית היסטוריה, ארכיטקטורת אנדרואיד וכו`	1		1
התנסות ב SDK Manager ו AVD Manager		2	2
סה"כ שעות:	1	2	3

## פרק 2 : תכנות מונחה עצמים בשפת JAVA חלק 1: הורשה,

### מחלקות מופשטת, ממשקים, פולימורפיזם.

**יעדים:** הכרת מושגי יסוד של תכנות מונחה עצמים מתקדם, הורשה, פולימורפיזם, ממשקים, מחלקה מופשטת, המרה כלפי מעלה וכלפי מטה, דריסת פעולות, זימון פולימורפי.

#### תכנים:

- **הורשה:**
- הורשה – מייצגת חלוקה היררכית טבעית-אנושית: תהליך של מיון וסיווג היררכי. ייצוג היחס "סוג של" (is a).
- הורשה בג'אוה (single inheritance) – התוספת extends בכותרת המחלקה, מחלקת-על (super class) ותת-מחלקה (sub class), הרשאת הגישה מוגן – protected.
- הורשה – מה עובר? שיטות בונות, השימוש ב-super לגבי שיטה רגילה ולגבי קונסטרוקטור.
- דריסה (הגדרה מחדש) של שיטות – overriding.
- הורשה מהמחלקה הראשונה – Object.
- הורשה כתומכת ברעיון הסתרת מידע, שימוש חוזר בקוד.
- פולימורפיזם – רב צורתיות: היתרון, העוצמה של הרעיון והשימוש בו.
- זימון פולימורפי של שיטות, המרה למעלה (up casting), המרה למטה (down casting).
- האופרטור instanceof, תוך הדגשת צמצום השימוש בו לטובת רעיון הפולימורפיזם.
- מחלקות עוטפות – הדרך להפוך טיפוסים בסיסיים לאובייקטים.
- **ממשק interface:**
- ממשק כמגדיר התנהגות ומייצג את היחס "מתפקד כ-". (לעומת "סוג של" בהורשה).
- ממשק אינו מחלקה: הוא מגדיר טיפוס אך אינו משמש כתבנית ליצירת עצמים.
- הממשק כמגדיר חוזה. כל מחלקה המעוניינת לממש את הממשק – חייבת לממש את כל המפורט בחוזה. חובות וזכויות המחלקות המממשות.
- הממשק בג'אוה: תוספת ה-implements לכותרת מחלקה המממשת את הממשק.
- הורשה בין ממשקים. מימוש מרובה של ממשקים.
- ממשקים בשירות הפולימורפיזם.
- **מחלקות מופשטות:**
- הגדרת הצורך בהגדרת מחלקות מופשטות: המחלקה המופשטת כמגדירה רעיון שאינו ניתן למימוש וליצירת עצמים בשלב נתון.
- המחלקה המופשטת בג'אוה: כותרת המחלקה. מימושים חלקיים. הגדרת שיטות מופשטות. שיטות בונות.
- המחלקה המופשטת כמגדירה חוזה. כל מחלקה המעוניינת לממש את המחלקה – חייבת לממש את כל המפורט בחוזה. חובות וזכויות המחלקות המממשות. הגדרה מחדש במחלקות היורשות. ש במחלקות היורשות.
- מחלקות מופשטות בשירות הפולימורפיזם.

- תרשים UML לתיאור מחלקות, מחלקות מופשטות וממשקים והקשרים האפשריים ביניהם.

#### מטרות ביצועיות:

- התלמיד יזהה ולהבדיל בין ירושה להרכבה בין נחלקות.
- התלמיד ייתנסה בביצוע ירושה ממחלקה קיימת.
- התלמיד ייתנסה בביצוע ירושה.
- התלמיד יסביר את מהות ותפקיד הממשקים (מאזיונים לדוגמא).
- התלמיד ייתנסה ביצירת ממשקים ושימוש בהם.
- התלמיד ישתמש ויסביר במחלקה המופשטת והשיטות המופשטות.
- התלמיד יתאר ע"י תרשים UML את היחסים האפשריים בין המחלקות ובין המחלקות והממשקים.
- התלמיד ייתנסה בשימוש בפולימורפיזם וזימון פולימורפי: בשימוש בירושה, או במימוש ממשק.

#### דרכי הוראה:

- בהוראת הורשה מומלץ להתחיל את זה ע"י דוגמאות קרובות לתפיסת התלמידים כמו היחס בין בעלי חיים, בעלי חוליות, חסרי חוליות, יונקים, עופות, דגים וכו'. מומלץ להשתמש במצגות היחידה החמישית תכנות מונחה עצמים של האוניברסיטה העברית.
- לתרגל עם התלמידים זיהוי היחס בין מחלקות ע"י שימוש "סוג של" ו "מורכב מ", ולתאר את היחס ב UML.
- לתרגל עם התלמיד את התחביר לכתיבת ההורשה לפחות פעם אחת, ולתת עבור תרשים UML המכיל יחסי הורשה והרכבה. לתת לתלמיד לממש את התרשים במחברת ובמחשב.
- אחרי המימוש במחשב התמידים ייצרו מחלקה עם פעולה ראשית לביצוע בדיקה למה שנכתב, ולהתנסות מעשית בהמרה כלפי מעלה (פולימורפיזם) והמרה כלפי מטה ומשמעותה.
- לחזור על הדברים תוך שימוש במחלקה מופשטת. כגון שימוש במחלקה "צורה" והפעולה המופשטת כדי לחשב שטח. יש לבקש מהתלמיד ליצור מחלקות מלבן, ריבוע ומעגל כהרחבה של המחלקה "צורה" ולבצע דריסה לפעולת חישוב השטח.
- בשלב זה אפשר להתנסות באופן מעשי ומוחשי את הזימון הפולימורפי על מערך של צורות.



- לגבי ממשקים ניתן לייצר ממשק בשם AreaAble שמכיל פעולת ממשק לחישוב שטח לבקש מהתלמיד לצור מחלקות מלבן, ריבוע ומעגל שממשימים ממשק זה ולבצע דריסה לפעולת חישוב השטח. בשלב זה אפשר להתנסות באופן מעשי ומוחשי את הזימון הפולימורפי על מערך מטיפוס AreaAble.

#### דרכי הערכה:

- בחינת מעבדה בה יתבקש התלמיד לממש על המחשב תרשים UML וממשק מחלקות נתון, ומחלקה בדיקה תוך רישום דרישות בחרות. המבחן יכיל שאלות הבנה שיכול לבדוק את הטיעונים מעשית מול המחשב.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
הורשה	6	12	18
ממשקים	4	8	14
מחלקה מופשטת	2	4	6
סה"כ שעות:	12	24	36

## פרק 3 : אוספים

**יעדים:** היכרות עם מבני נתונים ואוספים המובנים במערכת האנדרואיד וההבדלים ביניהם.

#### תכנים:

- אנדרואיד תומכת במגוון אוספים גנריים, שכולל מחלקות, מחלקות מופשטות וממשקים למימוש אוספים לפי הצורך.

- `ArrayList<E>`. אוסף שממש את `List` ומתנהג כמערך. מבנה דומה `Vector<E>`, כמו `ArrayList` שמוגן גישה ע"י `synchronized`.
- `HashMap<K, V>`: מחלקה המממשת את הממשק `Map`, משחזיקה את הנתונים בשיטת מפתח/ערך.

#### מטרות ביצועיות:

- התלמיד יסביר את תפקיד ויזהה את ההבדל בין האוספים.
- התלמיד יתנסה בשימוש באוספים הגנריים.
- התלמיד יכתוב קוד שיבצע: הוספה, חיפוש, עדכון ומחיקה מכל סוג של האוספים.

#### דרכי הוראה:

- לבנות מחלקה שמנהלת נתונים, כגון: ניהול ספר טלפונים.
- ליישם את הדוגמא ב `ArrayList<E>` או `Vector<E>` ו `HashMap<K, V>` ולעמוד על יתרונות וחסרונות כל יישום.

#### דרכי הערכה:

- מטלת ביצוע, לניהול נתונים לפי בחירת התלמיד כגון: ניהול עבודות בית, ציוני תלמיד (תעודה), נוכחות תלמידים, ציוני תלמידים (כיתה) ועוד.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
<code>Vector&lt;E&gt;</code> , <code>ArrayList&lt;E&gt;</code>	1	2	3
<code>HashMap&lt;K, V&gt;</code>	1	2	3
סה"כ שעות:	2	4	6

## פרק 4: חריגות

**יעדים:** שגיאת זמן ריצה או החריגה, סוגי החריגות, הטיפול בחריגות.

#### תכנים:

- חריגות `Exception` מה היא חריגה: שגיאה מול חריגה.
- חריגה היא עצם, המחלקה `Exception`.

- טיפול בחריגות בגאווה. throw, try.. catch.. finally.
- חריגות מוגדרות מראש, כל החריגות הן סוג של Exception.

#### מטרות ביצועיות:

- התלמיד יתנסה בקיום ויסביר את משמעות החריגה.
- התלמיד יטפל בחריגה ע"י תפיסתה באמצעות הבלוקים של try.. catch.. finally ויסביר את משמעות מילות שמורות אלו.
- התלמיד יטפל בחריגה ע"י זריקה של חריגה באמצעות throw.
- התלמיד יבחין בכדאיות או הכרחיות של טיפול בחריגה ע"י try.. catch.. finally או זריקתה ע"י throw.
- התלמיד יזהה את הפעולות הקיימות ב API שזורקות חריגות.

#### דרכי הוראה:

- לבצע חריגות מכוונות כמו: גלישה מתחום מערך, חלוקה באפס, גישה להפניה שהינה null.
- לזהות איך המערכת והיישום מגיבים ללא טיפול בחריגות, ואחר כך להוסיף טיפול ע"י try.. catch.. finally, ולעמוד על ההבדלים והתוצאות.
- לבצע טיפול באחת החריגות ע"י זריקה לפחות 2 רמות עד להגעה למחלקה שיכולה להשתמש בממשק משתמש גרפי כדי להודיע על החריגה ואופן הטיפול בה.

#### דרכי הערכה:

- בחינה עיונית/מעשית הכוללת קוד עם סיכוי של חריגות, ושימוש בפעולות שזורקות חריגות, על התלמיד לכתוב ולטפל בחריגות האפשריות ובחריגות שנזרקות ע"י הפעולות הנתונות. התלמיד יכין עבודה בבית ויגן בכיתה.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
המחלקה, Exception, throw, try.. catch.. finally.	1	2	3

3	2	1	סה"כ שעות:
---	---	---	------------

## פרק 5 : מבוא ליישום/אפליקציה: פעילות/ Activity

האפליקציה באנדרואיד נכתבת בשפת ג'אווה ועוברת הידור ע"י Android SDK. והתוצר מכיל את כל הקוד המהודר כולל קבצי הנתונים והמשאבים בחבילה אחת בקובץ דחוס בסיומת apk שהינה קיצור ל Android package. הפעילות או ה Activity מהרכיבים החשובים של היישום. הפעילות מייצגת מסך ממשק משתמש גרפי יחיד מהאפליקציה. אפליקציה יכולה להכיל מספר של פעילויות. לכתיבת פעילות צריך לבצע הרחבה (ירשה) למחלקה Activity

**יעדים:** מבוא למהות היישום, מבנה, אבני היסוד, משאבים, קובץ המניפיסט AndroidManifest.xml. מעגל חיי הפעילות, המתודות המופעלות המעגל חיי הפעילות.

### תכנים:

- אבני יסוד האפליקציה:  
Activity , Service , Broadcast receiver , Content provider , Intent.  
הרכיבים המרכזים שדרך כותבים את האפליקציה. נחשבים לנושאים המרכזיים ויוסברו בהרחבה בהמשך. האפליקציה שחיים בה רכיבים אלו נקראת הקשר היישום ( Application Context), כך מאפשרת שיתוף נתונים ביניהם.  
· הגדרות האפליקציה.  
מבוא למבנה קובץ המניפיסט AndroidManifest.xml . המכיל הגדרות עבור הפעלת האפליקציה המצינים את היכולות, דרישות והרשאות.  
· מבוא משאבי האפליקציה. Resources.  
הם צרכים חיצוניים שנצטרך להם ביישום שלנו, כגון תמונות, מחרוזות, צבעים, סגנונות ועוד.  
כל אלה אמורים להימצא בתיקה המתאימה תחת תיקיית המשאבים בפרויקט "res".

#### מעגל חיי הפעילות.

לכל פעילות מעגל חיים, ומספר מצבים שעוברת דרכם במהלך חייה, מרגע היצירה עד הסיום. מתחילה ממצב האתחול Starting ומסתיימת במצב מושמד Destroyed, וביניהם יכולה להיות במצב ריצה Running (גם נקרא Resumed), מושהה Paused או מופסק Stopped.

#### מתודות/פעולות מעגל חיי הפעילות.

( onCreate(), onStart(), onResume(), onPause(), onStop(), onDestroy() )

מתודות שתפקידן להעביר את הפעילות ממצב למצב, הפעלת מתודות אלו במהלך ריצת היישום מתבצע כתוצאה מצורכי המערכת ומהאינטראקציה עם המשתמש. כדי לממשן צריך לדרוס אותן בהרחבת המחלקה Activity.

#### מטרות ביצועיות:

- התלמיד יסביר את אבני היסוד של היישום, קיום קובץ הגדרות חיוני ומשאבי היישום.
- התלמיד ייצור פרויקט אנדרואיד בסביבת הפיתוח eclipse.
- התלמיד יבחין ויזהה את מבנה היישום בסביבת הפיתוח eclipse.
- התלמיד יזהה וינמק את תפקיד הפעילות והבנת מעגל חיייה כולל המעבר ממצב למצב, ותפקיד המתודות המופעלות בכל מעבר.
- התלמיד יתנסה בזיהו הפעולות המופעלות המעבר ממצב למצב.

#### דרכי הוראה:

- לדון באבני היסוד ותפקידם תוך מתן דוגמאות. לבקש מהתלמיד ליצור פרויקט ולעקוב אחרי מבנה הפרויקט ב eclipse.
- יעבר לתלמיד תרשים מעגל החיים של הפעילות שכולל את המצבים ושמות המתודות המופעלות בכל מעבר ממצב למצב. חשוב שתתבצע דוגמא מעשית תוך שימוש בפלט של ה Logcat לזיהוי זימון פעולה מסוימת במעגל חיי הפעילות.

#### דרכי הערכה:

- בחינה מעשית שתכלול יכולת של התלמיד ליצור פרויקט ולזהות את מרכיביו.
- בחינה בכתב בנושא מעגל חיייה הפעילות והיתכנות כל מצב.
- בחינת המעבדה בה מתבקש התלמיד לבצע דרישות הדפסה מסוימת במעבר ממצב למצב ולהבחין שהינו יכול לזהות מתי פועלת כל פעולה.

## חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
אבני יסוד האפליקציה, וקובץ ההגדרות	1	2	3
מבנה הפרויקט, קובץ ההגדרות והמשאבים	1	1	2
מעגל חיי הפעילות.	1	3	4
סה"כ שעות:	3	6	9

## פרק 6 : ממשק משתמש UI-User Interface: רכיבים ותבניות

ממשק משתמש גרפי באנדרואיד מורכב מרכיבים גרפיים שנקראים View ו- ViewGroup. כל הרכיבים הגרפיים הם תתי מחלקה של View שנקראים גם וידג'טים (widgets). לשימוש ברכיבי ה UI באנדרואיד ניתן בשתי אפשרויות. הראשונה "תכנותית" ע"י קוד ג'אווה כמו שניעשה ב Java AWT והשנייה הצהרתי ע"י קובץ XML.

**יעדים:** שימוש ברכיבי UI, שמירת מצב הפעילות וטיפול בשינוי תצורה. תבניות UI ב XML לסידור רכיבים, קישור עם הפעילות, רכיבי UI שימושיים והפלטה של Eclipse.

### תכנים:

- ממשק משתמש (UI) ע"י קובץ תבניות XML ה- XML Layouts.
- שימוש במשאב תבנית XML מקוד JAVA.
- מאפיינים ופרמטרים בקובץ תבניות XML.
- Form Widgets: שכוללת רכיבים של הצגת טקסט, מקשים לסוגיהם, ציר התקדמות, סרגל דירוג וכו'.
- שדה טקסט: קיימים כמה סוגים ואופציות של שדות טקסט לפי גודל ויעוד שדה הטקסט.
- לחצנים למיניהם.
- תמונה ומידיה: רכיב תמונה ImageView וכפתור תמונה ImageButton, רכיב וידאו VideoView.
- סידור ממשק המשתמש ע"י תבניות.
- קיימות מספר אופציות לסידור ממשק המשתמש שמספקת אותם אנדרואיד ע"י המחלקות שמגדירות סוגי התבניות: RelativeLayout, TableLayout, LinearLayout, FrameLayout, AbsoluteLayout.

- מיקום הפריסה, גודל, ריפוד ושוליים.
- שמירת מצב הפעילות.
- במעבר מפעילות לפעילות אחרת שגורם למעבר ממצב למצב, לפעמים יש צורך בשמירת מצב הפעילות הנעלמת מהמסך. מהכלים והפעולות שעוזרים בשמירת המצב: `onSaveInstanceState()`, `setSaveEnabled()`, `android:saveEnabled`.
- טיפול בשינוי תצורה.
- חלק מהמכשירים עוברים שינוי תצורה בזמן ריצת פעילות מסוימת, כגון שינוי כוון המסך או הצגת/הסתרת מקלדת, שינוי שפה וכו'. תגובת המערכת בדרך כלל לשינויים כאלה ע"י השמדה ויצירה מחדש של הפעילות (ע"י זימון `onDestroy()` ומיד זימון `onCreate()`). להשפעה על שינויים כאלה צריכים לממש את הפעולות `onRestoreInstanceState()`, `onSaveInstanceState()`.

### מטרות ביצועיות:

- התלמיד יבנה ממשק משתמש גרפי ע"י תבנית XML.
- התלמיד יבצע שימוש בתבנית ה UI מקוד גאווה כמו הצגת התבנית קבלת הפניות על רכיבי הממשק, ועוד.
- התלמיד ישתמש בלפחות שלוש מהרכיבים הגרפיים ל UI.
- התלמיד יאתר רכיבים שלא למד וילמד איך להשתמש בהם ע"י ממשק כל רכיבי.
- התלמיד ישתמש ברכיבי ממשק משתמש גרפי פשוטים שיוצר אותם ע"י קוד JAVA.
- התלמיד ישתמש ויסביר שיטות לשמירת מצב (נתוני) הפעילות אחרי ביצוע שינוי על הצגת כגון סיבוב המכשיר, הצגת והעלמת מקלדת וכו'.

### דרכי הוראה:

- להתנסות עם התלמידים בכתיבת ממשק פשוט בשתי השיטות התכנותית בשפת JAVA והצהרתית ב תבנית XML, ולימוד על ההבדל בין בשתי האופציות בפרט כשאר רוצים לבצע סידור לרכיבים ומתן ערכים לתכונות של כל רכיב והיתרון של כתיבה בתבנית XML שני מקרים אלו, אבל אי אפשר לוותר על הקוד ב JAVA לצורך טעינת והצגת התבנית וטיפול באירועים ובתכני רכיבי ה UI האמורים להשתנות כתוצאה מהאינטראקציה עם המשתמש.
- לצורך התנסות מעמיקה צריך לבצע דוגמא מעשית שכוללת בניית ממשק בעזרת תבנית XML שכוללת לפחות שדה קלט, שדה טקסט לדוגמא, וכפתור. אפשר לקחת דומא לחישוב ה BMI שכוללת שני שדות טקסט לקלט הגובה והמשקל, כפתור לצורך ביצוע החישוב

וטקסט להצגת התוצאה. אפשר לשדרג את הדוגמא ע"י הוספת תמונות המתאימות לתוצאות או צבעים וכו'.

- דוגמאות נוספות כמו מחשבון והמרת מטבעות. התלמיד יתנסה בבניית אותו ממשק בסידורים (תבניות שונות).
- לתרגל עם התלמידים איתור רכיב גרפי באמצעות הפלטה (Palette) של רכיבי ה UI שמספקת אותם eclipse בזמן עריכת תבנית XML. ומשם לאתר את התיעוד והממשק או לחפש את ממשק הרכיבים באתר הרשמי של האנדרואיד <http://developer.android.com>.
- כדי לזהות שמירת/איש שמירת מצב צריך לבצע תרגיל של פעילות שמכילה לפחות שדה טקסט אחד, במהלך הניסוי תתבצע כתיבה לתוך הדשה, ביצוע שינוי תצורה כמו סיבוב מסך.

#### דרכי הערכה:

- בחינה מעשית לבניית ממשק משתמש שהמורה מצייר או מראה אותו (עבודת בית עם הגנה בכיתה).
- להוסיף שורות קוד המאפשרות שמירת הנתונים שכבר הוכנסו לממשק המשתמש בזמן שינוי מצב המסך מאנכי לאופקי.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
תבניות XML שימוש במשאב	5	10	15
תבנית XML מקוד JAVA ורכיבי ממשק	4	8	12
סידור הממשק ע"י תבניות.	1	2	3
שמירת מצב הפעילות בשינוי תצורה.	10	20	30
סה"כ שעות:			



## פרק 7 : תכנות מונחה אירועים ושימוש במחלקה פנימית

האינטראקציה בין המשתמש ליישום מתבצעת דרך מנגנון האירועים, מערכת האנדרואיד מדווחת ליישומים ולרכיבים הגרפיים על אירועים ששמתרחשים והקשורים לאותו רכיב.

**יעדים:** טיפול באירוע לחיצה ע"י XML, אירועים ומאזינים.

### תכנים:

- אירועים Events :  
כאשר מתרחש אירוע בממשק המשתמש, הפריט (View) הקשור לאירוע הוא האחראי ללכוד ולטפל באותו אירוע, ומערכת האנדרואיד והרכיבים בנויים לכך.  
מאזינים:
- הינם ממשקים המהווים חלק מה- (View), כדי לבצע טיפול באירוע צריך לממש את המאזין המתאים ודריסת שיטת התגובה ומהם:  
View.OnClickListener, View.OnLongClickListener, View.OnFocusChangeListener, View.OnTouchListener, View.OnCreateContextMenuListener, View.OnKeyListener
- מחלקות פנימיות (Inner Classes) והרשאות גישה.
- מימוש מאזין כמחלקה פנימית.

### מטרות ביצועיות:

- התלמיד יתכנת תגובה לאירוע לחיצה בשיטת XML.
- התלמיד יסביר את תהליך הטיפול באירוע: אירוע ← מאזין ← תגובה.
- התלמיד יתכנת תגובה לאירועים המטופלים ע"י מאזינים (2 מאזינים לפחות). על פי השלבים: יצירת המאזין ← רישום/הצמדת המאזין לרכיב ← כתיבת התגובה.
- התלמיד יתנסה ביצירת אובייקט מאזין: ה Activity מממשת מאזין, מחלקה פנימית כמאזין, יצירת עצם מאזין בשורה (Inline).

### דרכי הוראה:

- להמשיך את הדוגמאות והתרגילים הקודמים מפרק ממשק משתמש גרפי (BMI), מחשבון, ממיר מטבעות וכו') להוסיף אירועים שונים בשיטות שונות.

## דרכי הערכה:

- מטלה ביצועית במעבדה או בבית לפעילות עם ממשק משתמש שמגיב לאירועים: כמו מחשבון, הגרלות, אלבום תמונות וכו'.

## חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
טיפול באירוע לחיצה בשיטת XML	1	1	2
בניית מאזינים: פעילות ממשק, מחלקה פנימית ויצירת מאזין בשורה.	2	5	7
סה"כ שעות:	3	6	9

## פרק 8 : מסרים Intents

הרכיבים העיקריים של האפליקציה activities, services, ו-broadcast receivers מופעלים ע"י מסרים שנקראים intents. ה intent נחשב לכלי קשירה בזמן ריצה בין רכיבים אלו ששיכים לאותו יישום או ליישומים שונים, ומהווה מנגנון משלוח מסרים/נתונים. מהשימוש הנפוץ שלו מעבר מפעילות (מסך) לפעילות אחרת, הפעלת שירותים services ועוד אחרים הקשורים לרכיבים שנלמד בהמשך ואז יפורט ויודגש השימוש ב-intent. לדוגמא על ידי מסר, פעילות יכולה לבקש להפעיל פעילות אחרת.

**יעדים:** תפקיד העצם Intent, סוגים: Implicit, Explicit, ומסנני מסרים Intent filters, פעילות מחזירה תוצאה Activity for Result.

## תכנים:

- העצם Intent: הינו עצם משייץ מבנה נתונים מופשט, ונראה כחבילה של מידע שמעניינת את מקבל המסר, ובנוסף מידע שמעניין את מערכת האנדרויד. מהם: שם הרכיב Component name שמטפל ב Intent, הפעולה Action ועוד.

- מרכיבי ה Intent: **Component Name**, **Action Name**, **Data**, **Category**, **Extra**, **Flags**.
- Explicit Intent או מסר מפורש, בסוג זה מציינים במפורש את הרכיב שאמור להיקרא על ידי מערכת האנדרואיד, באמצעות המחלקה ב JAVA כמזהה.  
אפשר להוסיף נתונים למסר ע"י הפעולה `putExtra(key,value);`
- זימון פעילות ע"י מסר מפורש, מתבצע ע"י זימון פעולות ההקשר המזמין (פעילות, שירות וכו') `startActivity(Intent)` ו `startActivityForResult(Intent)`.
- Implicit Intent מסר המרומז, לא מציינים את המחלקה ב JAVA שאמורה להיקרא אלא מציינים את הפעולה שאמורה להתבצע. ייתכן עוד URI שישמש את הפעולה.
- העברת וקבלת נתונים ב Implicit Intent מסר המרומז, הרכיב שמקבל את המסר יכול לקבל את הנתונים מהמסר ע"י פעולות אחזור  
(getters) לדוגמא `getData()`, `getExtras()`, `getAction()` ועוד.
- מסנני המסרים Intent filters: תפקידם ליידע את המערכת על איזו מסרים (intents) מרומזים צריך לטפל. כל מסנן מתאר יכולת של הרכיב, איזו קבוצת מסרים (intents) מוכן לקבל.
- מחסנית המשימות .
- פעילות מחזירה תוצאה (Activity for result)

#### מטרות ביצועיות:

- התלמיד יסביר את משמעות המסרים ותפקידיהם.
- התלמיד יסביר משמעות כל רכיב מרכיבי המסר.
- התלמיד יתנסה בשני סוגי המסרים ויבחין ביניהם.
- התלמיד יתנסה לנווט ממפעילות לפעילות אחרת באותו יישום בעזרת (intents) שני הסוגים.
- התלמיד יתנסה בשימוש במסנני המסרים. שימוש במסנני מסרים קיימים או יצירת מסננים מסרים משלו.
- התלמיד יתנסה בזימון פעילות מחוץ ליישום כמו דפדפן, בוחר תמונות, בוחר איש קשר וכו'.
- התלמיד יתנסה בהעברת נתונים בין פעילויות.
- התלמיד יתנסה בקבלת תוצאה מפעילות קיימת (לדוגמא בוחר תמונות).

- התלמיד יתנסה בניית פעילות שמחזירה תוצאה, ובניית פעילות מפעילה את הפעילות ומטפלת בתוצאה המוחזרת. (ActivityForResult).
- התלמיד יתאר את מחסנית המשימות ואת התהליך המתבצע על כל פעילות בזמן הצגה או העלמה מהמסך. (אין צורך בשינוי או השפעה על מצב מחסנית המשימות)

#### דרכי הוראה:

- ליצור שתי פעילויות ולבצע ניווט הדדי תוך שימוש במסרים והעברת נתונים וקריאתם. כדי להמחיש את אחד מתפקידי המסרים.
- ליצור דוגמאות בהן המסרים מזמינים פעילויות מחוץ ליישום.
- לבצע דוגמא המשמשת מסנני מסרים קיימים ו/או יצירת מסנני מסר מלשו ולבצע שימוש בהם.
- לבצע הדגמה להעברת נתונים ע"י מסרים.

#### דרכי הערכה:

- מטלת ביצוע מסכמת שמבצע אותה התלמיד בכיתה או בבית, לדוגמא:  
יישום שמכיל שתי פעילויות הראשונה עם ממשק של מספר שמייצג סכום כסף ושני שדות ומציינים מהיכן היכן לבצע המרה.  
הפעילות השנייה מופעלת למטרת תוצאה שמקבלת את הערכים מהפעילות הראשונה שמבצעת המרה ומחזירה את התוצאה לפעילות המבקשת.  
במעבר מפעילות הראשונה לשניה אפשר להדגים שני סוגי המסרים.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
משמעות המסרים וסוגיהם וביצוע הדגמה לכל סוג.	1	2	3
מסנני מסרים	1	2	3
פעילות למטרת תוצאה	2	4	6
סה"כ שעות:	4	8	12

## פרק 9: רכיבים מתקדמים: תפריט, תיבת דו-שיח, הודעות Status bar וטיפול באירועים

**יעדים:** תפריטים, תיבות דו שיח, הודעות Toasts ושורת מצב.

### תכנים:

- תפריטים Menus: Context menu, Options menu.
- טיפול באירועי התפריטים: דריסת onOptionsItemSelected, ודריסת onContextItemSelected של המחלקה Activity.
- תיבות דו שיח Dialogs: AlertDialog, ProgressDialog, Custom Dialog, DatePickerDialog, TimePickerDialog.
- טיפול באירועי תיבות הדו-שיח: DialogInterface.OnClickListener, DialogInterface.OnMultiChoiceClickListener, DialogInterface.OnTimeSetListener, DialogInterface.OnDateSetListener.
- הודעות Notifications: הודעות טוסט Toast Notifications, הודעות שורת מצב Status Bar Notifications.
- טיפול באירוע בחירת הודעת שורת המצב: setLatestEventInfo.

### מטרות ביצועיות:

- התלמיד יתנסה ביצירת תפריטים לסוגיהם.
- התלמיד יתנסה ביצירת תיבות דו שיח לסוגיהן.
- התלמיד יתנסה ביצירת הודעות טוסט Toast Notifications.
- התלמיד יתנסה ביצירת הודעות שורת מצב ומימוש התגובה הקשורה בבחירת ההודעה.

### דרכי הוראה:

- בתפריטים: לצור פעילות המדגימה את שני סוגי התפריטים.
- בתיבות דו שיח: לצור פעילות המדגימה את כל סוגי תיבות הדו שיח שנלמדו. ומימוש הטיפול באירועים הקשורים בתיבת הדו-שיח:

- בהודעות שורת המצב, לצור ממשק של שני כפתורים אחד מציג את ההודעה והשני מבטל. ולצור תגובה לבחירת ההודעה שמעבירה אותנו למסך אחר.

#### דרכי הערכה:

- מטלת ביצוע במעבדה או בבית המדגימה את כל מה שנלמד, לנסות לשלב את החידושים בדוגמאות ופרויקטים שנוצרו בפרקים קודמים ( פרק ממשק משתמש גרפי ופרק טיפול באירועים).  
לדוגמא תוספת תפריטים ותיבות דו שיחשב BMI, במחשבון או ממיר המטבעות (ערך ממטבע למטבע).

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
תפריטים	1	2	3
תיבות דו שיח	2	4	6
הודעות	2	4	6
סה"כ שעות:	5	10	15

### פרק 10 : משאבי האפליקציה Application Resources

משאבי אפליקציה הם צרכים חיצוניים שנצטרך להם ביישום שלנו, כגון תמונות, מחרוזות, צבעים, סגנונות ועוד. כל אלה אמורים להימצא בתיקה המתאימה תחת תיקיית המשאבים בפרויקט "res/". השיטה הזאת מאפשרת לנו להתאים את היישום שלנו לפי סוג המכשיר, שפה, מצב מסך המכשיר (לפני או אחרי סיבוב).

**יעדים:** היכרות סוגי המשאבים, ארגוןם בפרויקט תחת "res/", שימוש בהם (במחלקה R), יצירת סוגים שונים מהמשאבים.

## תכנים:

- אספקת המשאבים Providing Resources: סוגי המשאבים, איפה לשמור אותם, התאמת המשאבים לתצורות המכשיר.
- גישה למשאבים Accessing Resources: איך לגשת למשאבים ולהשתמש בהם, אם זה מקוד הג'אווה או ממשאב XML אחר.
- טיפול בשינויים בזמן ריצה Handling Runtime Changes: איך לנהל ולהשתמש במשאבים כאשר מתרחש שינוי תצורה בזמן שיישום שלנו בפעולה.
- תבניות שפות Localization: הצורך בהתאמת היישום ליותר משפה.
- סוגי משאבים Resource Types: איך לכתוב ולהשתמש בסוגים השונים של המשאבים ע"י רכיבי XML. מהסוגים הקיימים: מחרוזת String, צבעי מצב Color State List, וכאלו הניתנים לאפשר לציור (תמונות או XML וכו') Drawable, פריסה Layout, תפריט Menu, סגנון Style, הנפשה Animation, וסוגים אחרים ידועים כמו Booleans, integers, dimensions, colors, וarrays.

## מטרות ביצועיות:

- התלמיד יתאר את המבנה של תיקיית המשאבים "res".
- התלמיד יתאר את הקשר בין המשאבים ואת המחלקה R שנוצרת באופן אוטומטי.
- התלמיד ייצור ערכי טקסט כמשאב XML, ואת הזיקה עם תבניות למספר שפות.
- התלמיד ייצור תפריט, צבעים, מערכים, ממדים dimensions.
- התלמיד ייצור הנפשה ב XML.

## דרכי הוראה:

- פרק זה מרכז את סוגי המשאבים שהתלמיד השתמש בהם ולמד אותם בפרקים קודמים או שילמד אותם בהמשך. לכן אפשר ללמד פרק זה באופן נפרד או כתוספים לפרקים המתאימים לסוג המשאב. לדוגמא כשיוצרים פעילות אזי רצוי להרגיל את התלמידים בשימוש בערכים של משאב המחרוזת String לארגן את ערכי הטקסט החזותיים המשושמים במפעילות, כשלומדים על התפריטים אז נלמד את המשאב של התפריטים, כשלומדים יישומונים (App Widgets) אזי רצוי ללמד את המשאב של

ממדים dimensions, בפרק ההנפשות (אנימציות) נלמד את משאב ההנפשה. את שאר  
הנושאים יש ללמד באופן נפרד.

#### דרכי הערכה:

- בחינה עיונית שבה יסביר התלמיד סוגים של המשאבים.
- בחינת מעבדה שבה התלמיד מתבקש ליצור מספר משאבים מסוגים שונים שבהם יצטרך  
לשייך אותם לתיקיה המתאימה בתיקיית המשאבים "res/".

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
מבנה תיקיית המשאבים וסוגיהם	2	2	4
יצירת וארגון סוגים שונים מהמשאבים	1	4	5
סה"כ שעות:	3	6	9

## פרק 11 : אחסון ניתונים Data Storage

מערכת האנדרואיד מספקת מספר אפשרויות לאחסון נתונים, בחירת אופציית האחסון תלוי  
בדרישות שלך ובצרכים של היישום, כגון כמות הנתונים הנשמרים, האם הנתונים הנשמרים  
פרטיים ליישום או משותפים עם יישומים אחרים וכו'.

— יעדים: שמירת העדפות/הגדרות, אחסון פנימי, אחסון חיצוני, SQLite Database.

#### תכנים:

- אחסון העדפות משותפות Shared Preferences. ליישום בזיכרון המכשיר ע"י יצירת  
ממשק משתמש מסך העדפות PreferenceActivity.
- אחסון העדפות משותפות באופן ידני ע"י ממשק כלשהו.
- אחסון פנימי Internal Storage (זכרון המכשיר) קריאה/כתיבה קבצים בזיכרון הפנימי.
- קריאת קובץ שנמצא במשאבי היישום Resource File.



- אחסון חיצוני External Storage, כתיבה/ קריאה בהתקן אחסון חיצוני (SDCARD) לדוגמה).
  - השימוש בהרשאות : android.permission.WRITE\_EXTERNAL\_STORAGE
  - אחסון בבסיסי נתונים SQLite Databases:
- SQLite הוא מסד נתונים קוד פתוח אשר מוטבע באנדרואיד. מאפשר אחסון וניהול נתונים בעזרת שאלות SQL שמספקת SQLite.
- המחלקות השימושיות ב SQLite : SQLiteOpenHelper , SQLiteDatabase , Cursor , ContentValues.
  - הצגת נתונית בעזרת רשימה ListView ; SimpleCursorAdapter , ListView , פעילות רשימה ListView.
  - טיפול באירועי לחיצה על פריט ברשימה onItemClick , onItemClick.

#### מטרות:

- התלמיד יתנסה בבנית ממשק משתמש לשמירת הגדרות/העדפות Preferences.
- התלמיד יתנסה בשמירה ושחזר קבצים בזיכרון המכשיר ויתאר את מאפיינים שלם שמירה זו.
- התלמיד יבדוק קיום מדיה חיצונית במכשיר לשמירה נתונים בקצים ויתנסה בשמירת ושחזר קבצים בזיכרון החיצוני ויתאר את מאפייני שמירה זו.
- התלמיד יבנה טבלה במסד הנתונים SQLite Database ולבצע: שליפה, הוספה, עדכון ומחיקה. שזה מתאפשר בשפת SQL או במחלקות המקלות על השימוש במסד נתונים זה.
- התלמיד יציג תוכן מסד נתונים ע"י ListView , בשימוש במתאם SimpleCursorAdapter.
- התלמיד יטפל באירוע לחיצה רגילה ו/או ארוכה על פריט ברשימה ListView : onItemClick , onItemClick.

#### דרכי הוראה:

- לסקור בפני התלמידים את תכונות כל שמירה שכולל רמת הפרטיות והאבטחה. את הצורך לשימוש בכל שיטה.

- לבצע דוגמא של שמירת שני נתונים "משתמש" ו"סיסמא". שמירת הנתונים נבצע במספר שיטות כך התלמידים יחשפו לדרכי השימוש בכל שיטה.

#### דרכי הערכה:

- מטלת ביצוע שדרכה מתבקש התלמיד לצור ממשק משתמש לטיפול במסד נתונים כולל שמירה, שליפה, מחיקה ועדכון. והצגת הנתונים ע"י רשימה ListView .

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
אחסון העדפות משותפות Shared Preferences	1	2	3
אחסון פנימי	1	2	3
אחסון חיצוני	1	2	3
SQLite Database	7	14	21
SimpleCursorAdapter , ListView	2	4	6
סה"כ שעות:	12	24	36

## פרק 12 : תכנון וניתוח אפליקציות, ניתוח בשיטת UML, תאור

### מסכים, תרשים זרימה

**יעדים:** אפיון וניתוח יישום, ניתוח מחלקות ע"י תרשימי UML, תרשים זרימה ותיאור מסכים, זיהוי מרכיבים ייחודיים, תיאור הנתונים הנשמרים ואופן שצירתם

#### תכנים:

- אפיון האפליקציה.
- תאור הצורך והשימוש באפליקציה.
- זיהוי משימות ראשיות באפליקציה (פעילויות, שירותים, מקלטים, מסרים, ספקי תוכן).

- שרטוט סקיצות למשימות.
- תכנון תרשים זרימה לאפליקציה.
- יצירת תבניות או מסכים.
- קביעת הנתונים שרוצים לשמור ואו לבצע גיבוי, מבנה ושיטת השמירה לצורך זה.
- ניתוח מחלקות ע"י תרשים UML.

#### מטרות ביצועיות:

- התלמיד יגדיר את מטרת היישום.
- התלמיד יאפיין עצמים ומרכיבים חיוניים ביישום.
- התלמיד יתאר את המראה הגרפי של היישום.
- התלמיד יתאר את תרשים הזרימה של המסכים.
- התלמיד יקשר בין המסכים ובין הרכיבים והמשימות הייחודיות של היישום.
- התלמיד יתאר ב UML את המחלקות החיוניות והקשרים ביניהם.
- התלמיד ינהל לוח זמנים וסידור בניית היישום בשלבים.
- התלמיד ימדוד את התקדמותו בפיתוח היישום בהתאם לניתוח שביצע.

#### דרכי הוראה:

- לבחור יישום מייצג לבצע את כל האפיונים, ניתוחים, התרשימים, צרכים, מסכים וכו'.  
אותו יישום צריך לבצע אותו כולו עד סופו כדוגמא לפרויקט מלווה.

#### דרכי הערכה:

- בשלב זה התלמיד ייתכן שהתלמיד יבחר לעצמו יישום שירצה לפתח ולבצע את ניתוח והאפיון המתאים לקראת פרוייקט הגמר.

#### חלוקת שעות:

נושא			שעות עיוני	שעות מעשי	סה"כ
מטרת היישום והצרכים			1	2	3

משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

3	2	1	עצמים ורכיבים חיוניים ליישום
6	4	2	שרטוט הישויות ביישום והקשרים ביניהם
			כולל מסכים, שירותים, מקלטים, ספקי
			תוכן, נתונים הנשמרים.
12	8	4	סה"כ שעות:

כיתה י"ב,

פרקים:

פרק מס'	נושא לימוד	שעות עיוני	שעות מעשי	סה"כ
13	ספקי תוכן Content Providers	8	16	24
14	תהליכים וההליכונים (Threads & Processes)	15	30	45
15	שירות Service	8	16	24
16	מקלט שידורים BroadCast Reciever	7	14	21
17	טלפוניה	8	16	24
18	רכיבים מתקדמים: מולטימדיה ומצלמה, חיישנים, GPS, מפות.	12	24	36
19	גרפיקה ואנימציה.	8	16	24
20	הנחיה והגשת עבודת גמר	24	48	72
	<b>סך הכול שעות לימוד</b>	<b>90</b>	<b>180</b>	<b>270</b>

## פרק 13 : ספקי תוכן Content Providers

מאחר ואין שטח אחסון משותף לכל האפליקציות באנדרואיד, ספקי התוכן היא סוג מיוחד של מאגר הנתונים שחושף דרכים סטנדרטיים כדי לאחזר ולטפל נתונים המאוחסנים. אנדרואיד מגיע עם מספר ספקי תוכן שימושיים.

**יעדים:** מטרת ספקי התוכן, שימוש בספקי התוכן הקיימים של אנדרואיד, וביצוע שאלות.

#### תכנים:

- משמעות ספקי התוכן והסוגים שמספקת אותם המערכת ע"י השימוש ב URI המתאים.
- שימוש בספק נתונים: לאנשי קשר, יומן השיחות, תכני דפדפן, הגדרות ההתקן, קבצי מדיה, יומן.
- ביצוע שאלות על ספק תוכן: אחזור, הוספה, עדכון ומחיקת נתונים. ( ), query(), (insert(),update(),delete(), getType(), onCreate())

#### מטרות ביצועיות:

- התלמיד יתנסה בשימוש לפחות באחד מספקי התוכן הקיימים: לאנשי קשר, יומן השיחות, תכני דפדפן, הגדרות ההתקן, קבצי מדיה, יומן.
- התלמיד יתנסה לפחות בביצוע שאלת אחזור query().

#### דרכי הוראה:

- להבהיר את משמעות ספקי התוכן וחשיבותם, לצור טבלה שמכילת את ה URI המתאים לכל סוג מהספקים. ולבצע שימוש ספק נתונים אחד לפחות עם התלמידים תוך שימוש בכל סוגי השאלות האפשריות.

#### דרכי הערכה:

- מטלת ביצוע בה התלמיד יתבקש לשהתמש בספק תוכן אחר ממה שיצרו בכיתה, ויקבל הנחיות חיוניות לביצוע השאלות שעליו לבצע.

#### חלוקת שעות:

נושא			שעות עיוני	שעות מעשי	סה"כ
משמעות ספקי התוכן וה URI.			2	1	3

9	7	2	שימוש בספקי תוכן.
12	8	4	ביצוע שאלות.
24	16	8	סה"כ שעות:

## פרק 14 : תהליכים ותהליכונים (Threads & Processes)

תהליכון או Thread הוא רצף של פעולות שמתבצעות באופן עצמאי במטרה לבצע משימה מסוימת, וזאת במקביל לתהליכונים אחרים. במערכת האנדרואיד מקצה תהליך יחיד ותהליכון עבור כל יישום שמבקש לרוץ. פרק זה דן איך התהליכים Processes והתהליכונים Threads ממומשים ועובדים ביישומי האנדרואיד.

**יעדים:** Thread, Processes, "main" thread או "UI" thread, AsyncTask, Handler.

### תכנים:

- מה זה תכנות מקבילי? עיקרון מימוש המקביליות בעולם המחשב והצורך שלו וה-Thread כדוגמה למימוש תכונה זו בעולם המחשבים. תכונות ומרכיבי התהליכון בשפת JAVA.
- יצירת תהליכון באמצעות הורשה מהמחלקה Thread.
- תהליכון בתוך אפליקציה, מימוש הממשק Runnable.
- תהליכונים מרובים: תזמון בין רצפי פעולה - threads scheduling, סיום חייו של thread, המתודה join ו isAlive.
- שמירת שלמות הנתונים. Synchronization ובעיית הDeadlock.
- מתודות wait() ו-notify().
- תהליכים Processes: כל רכיבי אותו יישום רצים על אותו תהליך (Process) שמקצה מערכת הלינוקס לכל יישום.
- מעגל חיי התהליך: מערכת האנדרואיד מנסה לתת לתהליך לרוץ זמן ארוך ככל האפשר, בסופו של דבר צריך לסיים חלק מהתהליכים לצורך התייעלות בניצול וניהול הזיכרון. בחירת התהליך המיועד לסיום מסתמכת על רמת עדיפות של 5 רמות:
  - o Foreground process

Visible process	o
processService	o
Background process	o
Empty process	o

- תהליכים Threads והיישומים באנדרואיד.

המערכת מקצה לפחות תהליכון אחד עבור כל יישום המבקש לרוץ. תהליכון זה נקרא ה"תהליכון הראשי" ("main thread") תהליכון זה אחראי על שיגור אירועי ממשק המשתמש הגרפי. גם כן הינו התהליכון שבו מתבצעת האינטראקציה בין היישום לרכיבי ערכת הכלים של ממשק המשתמש הגרפי של אנדרואיד (Android UI toolkit). לכן תהליכון זה נקרא גם "תהליכון ממשק משתמש" או "UI thread".

- התהליכים הפועלים (Worker threads): תהליכון לביצוע משימה מקבילה לתהליכון הראשי בלי לחסום את ריצתו או לשבש את ריצתו. נעזרים בדרך כלל בפעולות הבאות (Activity.runOnUiThread(Runnable), View.post(Runnable), View.postDelayed(Runnable, long))

- שינוי ממשק המשתמש מתוך Thread ע"י Handler בשיטת שליחת ההודעות.
- שינוי ממשק המשתמש מתוך Thread ע"י משימה אסינכרונית AsyncTask: מאפשרת לבצע משימה אסינכרונית על ממשק המשתמש. כך שמבצעת את הפעולות החוסמות ב"תהליכון הפועל" ואז מעדכנת את התוצאות על תהליכון ממשק המשתמש "UI thread", בלי לטפל בתהליכים.

### מטרות ביצועיות:

- התלמיד יתאר את משמעות התהליכים והתהליכים באופן כללי ותיאורטי.
- התלמיד יתאר את הצורך המערכת בסיום תהליכים ואיך עושה זאת מערכת האנדרואיד לפי עדיפויות.
- התלמיד יבנה תהליכון ע"י Thread, Runnable.
- התלמיד יתנסה בעדכון ממשק המשתמש ע"י Thread אחר בעזרת Handler או ע"י המשימה האסינכרונית AsyncTask.

### דרכי הוראה:

- להסביר את משמעות התכנות המקבילי ואת הגדרת התהליכון. לבנות דוגמה פשוטה שמדגימה שני תהליכים. לדגומה לבנות מחלקה שמרחיבה Thread שמכילה תכונה אחת של מחרוזת שערכה מאותחל בשיטה הבונה, לבצע קטע קוד מקבילי שמדפיס את תוכן

התכונה. אחרי שמסיימים בונים שני עצמים עם שני ערכים שונים ולבצע הפעלה של שני ההליכונים ולצפות בתוצאות.

- לחזור על הדוגמא בעזרת מימוש המחלקה Runnable.
- כדי להמחיש את הצורך בHandler ו AsyncTask. לבנות פעילות עם Thread שמשנה משהו בתוך ה UI לדוגמא מציג מספר אקראי בעזרת TextView או Toast. אז נדגיש את הבעיה שזה לא מתאפשר לך משתמש בHandler או AsyncTask כפתרון לבעיה.

### דרכי הערכה:

- בחינה עיונית על משמעות התכונות המקבילי שכולל כתיבה קוד.
- בחינת מעבדה או מטלת ביצוע לכתיבת קטע קוד שירץ המקביל ולתת לתלמידים לבחור את השיטה שנוחה להם.

### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
התכונות המקבילי Thread	2	4	6
Runnable	3	6	9
תהליכים Processes	2		2
Handler או AsyncTask	8	20	28
סה"כ שעות:	15	30	45

## פרק 15 שירות Service

לשירות יש שני סוגים לפי היעוד שלהם הראשון "started" הרגיל ושני "bound" שקשור לרכיב אפליקציה אחר (כגון פעילות אחרת). השירות מתחיל כ "started" אחרי זימון רכיב אחר את הפעולה startService() ומסתיים אחרי שהוא מסיים את עצמו ע"י הפעולה



stopSelf() או רכיב אחר מסיים אותו ע"י הפעולה stopService().  
ומתחיל כ "bound" אחרי זימון רכיב אחר (הלקוח) את הפעולה bindService() ומסתיים בזימון הלקוח את הפעולה unbindService().  
בפרק זה נתמקד רק בסוג הראשון started.  
**יעדים:** משמעות ותפקיד השירות, בניית שירות started, מעגל החיים, ה IntentService, שירותים קיימים המערכת.

### תכנים:

- מעגל חיי השירות: לשירות יש 3 מצבים במהלך חייו Starting, Running, Destroyed(Shut-down).
- מתודות/פעולות מעגל חיי השירות: במעבר ממצב למצב מופעלות הפעולות הבאות:  
onCreate(), onStartCommand(), onBind(), onUnbind(), onUnbind(), onRebind(), onDestory().  
להתערב ולהשפיע במעבר ממצב למצב צריך לדרוס את פעולות אלו (שהן חלק מהמחלקה Service). סדר הקריאה לפעולות שונה משני סוגי השירות "started" ו "bound".
- בניית Started Service. כהרחבה של Service ו IntentService וההבדל ביניהם. (ה IntentService הינו שירות שרץ ב thread נפרד מהפעילות, לדוגמא שירות שמוריד קובץ מהאינטנט).
- שירות חזית Foreground Services: שירות חזית הוא שירות שאמור להיות משהו ידוע למשתמש כפעיל. השירות הופך להיות שירות חזית אחרי זימונו לפעולה startForeground עם הודעת שורת המצב.
- השימוש AlarmManager לצורך תזמון תחילת שירות ותזמון שירות מחזורית.

### מטרות ביצועיות:

- התלמיד יתאר את משמעות השירות ועל קיום שני סוגים "started" הרגיל ושני "bound" שקשור לרכיב אפליקציה אחר.
- התלמיד יתנסה בבניית השירות "started".
- התלמיד יפעיל ולעצור שירות.
- התלמיד יתאר את מעגל חיי השירות והיכולת לדרוס את פעולות שבמעבר ממצב למצב לפי הצורך.
- התלמיד יתכנת שירות חזית.

- התלמיד יבצע תזמון לתחילת שרות בעזרת AlarmManager ולבצע שירות מחזורי כל פרק זמן.

#### דרכי הוראה:

- רצוי להתחיל בהוראה פרונטאלית שמסבירה את משמעות השירות ותפקידה תוך כדי הדגמות של שיעון מעורר ועדכון דוא"ל, עדכון מזג האוויר וכו'.
- לצור שירות מסוג "started" ולדרוס את פעולותיו, בתוך כל פעילות הציג את שם הפעילות בעזרת "התראת טוסט" לבנות פעילות עם שני כפתורים כך שאחד מפעיל והשני עוצר את השירות מסוג "started", ולבצע את תגובת כל כפתור לפי התפקיד שלו.
- אפשר לשדרג את הדוגמא הקודמת לשירות חזית ע"י זימון השירות ל startForeground.
- בשלב המתקדם לבצע אותה משימה לתזמון עירור השירות בעזרת AlarmManager ושימוש ב (PendingIntent) ולשדרג לגרסה נוספת החוזרת על הפעלת השירות כל פרק זמן קצוב.
- נושא השירות יחזור בנושאים מתקדמים כגון תהליכונים (thread) ושירותי מערכת.

#### דרכי הערכה:

- מטלת ביצוע שצריכה לצור שירות והפעלתו. (עבודת בית והגנה בכיתה)

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
סוגי השירות, ומעגל החיים .	2	2	4
בניית שירות כהרחבה של Service ו IntentService	3	7	10
תזמון שירות ושירות מחזורי	2	5	7
שירות חזית	1	2	3
סה"כ שעות:	8	16	24

### פרק 16 : מקלט שידורים Broadcast Reciever ו App Widget.

Broadcast Receiver **מקלט שידורים** הוא מחלקה של היישום, שתפקידו להאזין למסרים (intents) שמשודרים. הינו מחלקה שמרחיבה את המחלקה BroadcastReceiver ורשומה ביישום AndroidManifest.xml. כל מקלט כזה נרשם לקליטת

מסר או קבוצת מסרים, והינו מיועד ע"י מערכת האנדרואיד והיישום ששייך לו במסרים שמשודרים, כלומר מתפקד כמאזין למסרי היישום והמערכת לפי הרשאות ויעוד. לדוגמא המערכת משדרת ליישומים שסיימה העלאה לכן יופעל המקלט עם מסן `android.intent.action.BOOT_COMPLETED`, כמו כן לזיהוי שיחה נכנסת או SMS שהתקבל עם מסן `android.provider.Telephony.SMS_RECEIVED` וכדומה. ביצוע תגובה למסר זה ע"י המתודה `onReceive()`.

**יעדים:** בניית `BroadcastReceiver`, שימוש והפעלתו- שידור מסרים למקלט, קליטת שידורי מסרים משירותי המערכת.

### תכנים:

- בניית מקלט `BroadcastReceiver`.
- שידור מסרים `Broadcast Intents` ע"י `void sendBroadcast(Intent intent)`.
- קליטת מסרי מערכת האנדרואיד: המסרים המסופקים ע"י שירותי המערכת ו `BroadcastReceiver`.
- הרשאות ואבטחה: שימוש בהרשאות קיימות והגדרת הרשאות אבטחה.
- שימוש בהרשאות:
- יישום בסיסי אינו מכיל הרשאות לכל דבר במערכת, כדי לתת הרשאה מסוימת צריך לציין אותה בקובץ המניפסט של היישום ע"י המרכיב `uses-permission`. לדוגמא רוצה היישום יקבל גם את מסרונים (SMS) הנכנסים, או לתת לא גישה למצלמה או GPS וכדומה.
- הגדרת ודרישת הרשאות:
- אפשר להגדיר הרשאות לפי דרישות המתכנת בקובץ המניפסט ע"י הרכיב `<permission>`
- `App Widgets`: יישומון הינו יישום קטן, שמהווה תצוגה מצומצמת או זעירה של יישום מסוים שניתן להטמיע אותה באפליקציות אחרות כמו מסך הבית של מערכת האנדרואיד. יישומון זה יכול להתעדכן באופן תקופתי. (לדוגמא: יישומון המציג מזג האוויר, יישומון המציג את סדר היום של לוח שנה וכו').
- ליצירת יישום יש צורך ב `AppWidgetProviderInfo` שמספק תיאור ומאפיינים ליישומון, ו `AppWidgetProvider` שהינו סוג של `broadcast receivers` או מקבלי מסרים, עצם כזה מקנה לאפליקציה יכולת קבלת מסרים (Intents) ששודרו ע"י המערכת או אפליקציה אחרת. אפילו כאשר רכיבים אחרים של היישום אינם פעילים.

### מטרות ביצועיות:

- התלמיד יתאר את מנגנון מקלט המסרים.
- התלמיד יתנסה בתהליך הבנייה: בניית מקלט, רישום, שליחת מסרת קבלת המסר והתגובה.
- התלמיד יסביר את הצורך בהרשאות אבטחה והשימוש בהן.
- התלמיד יצור הרשאה משלו למקלט שהוא בנה.
- התלמיד ישתמש לפחות במסר של שירות מערכת אחד כמו: סיום עליית המערכת, שיחה נכנסת, SMS התקבל וכו'.

#### דרכי הוראה:

- להבליט את חשיבות מנגנון זה תוך מתן דוגמאות ממסרי המערכת. לצור דוגמא של הפעלת שירות או הצגת פעילות כתגובה לקליטת מסר שנשלח ע"י יישום התלמיד או מסר של המערכת.

#### דרכי הערכה:

- מטלת ביצוע לצור דוגמא של הפעלת שירות או הצגת פעילות כתגובה לקליטת מסר שנשלח ע"י יישום התלמיד או מסר של המערכת.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
המחלקה BroadcastReceiver, בניית מקלט, רישום, שליחה וקליטת מסר.	3	6	9
שירותי מערכת, הרשאות ואבטחה.	2	4	6
App Widgets	3	6	9
סה"כ שעות:	8	16	24

*telephone ביוונית פירושה "speech over a distance."  
tele, "distant", phone "speech"*

**יעדים:** בירור מצב הטלפון, שיחות חיוג קבלת שיחה נכנסת, הודעות טקסט SMS שליחה וקבלת הודעה. אנדרואיד מספקת חבילה עם מחלקות שמטפלות בכל נושא הטלפוניה.

### תכנים:

- מנהל הטלפוניה TelephonyManager.
- מאזין מצב הטלפון PhoneStateListener.
- קבלת תכונות ומצב הטלפוניה של הטלפון.
- שימוש במסרים (Intent) לבקשת חיוג עם Intent.ACTION\_CALL. וביצוע חיוג ע"י Intent.ACTION\_DIAL.
- זיהוי שיחה נכנסת.
- שליחת SMS ע"י Intent או ע"י מנהל המסרונים SmsManager.
- הרשאות הנחוצות לחיוג, קבלת שיחה, שליחת SMS וקבלת SMS.

### מטרות ביצועיות:

- התלמיד יסביר כיצד לקבל את תכונות הטלפון.
- התלמיד יתנסה בביצוע בקשת חיוג או ביצוע חיוג.
- התלמיד ישתמש במאזין מצב הטלפון לזיהוי מצב הטלפון וזיהוי שיחה נכנסת.
- התלמיד יזהה שיחה יוצאת.
- התלמיד יבצע שליחת וקבלת מסרון SMS.
- התלמיד יתנסה בשימוש מקלטי שידור BroadcastReceiver לצורך זיהוי שיחה יוצאת.
- התלמיד יתנסה בשימוש מקלטי שידור BroadcastReceiver לצורך זיהוי קבלת SMS.

### דרכי הוראה:

- הדגמת השימוש במאזין מצב הטלפון PhoneStateListener ושימוש BroadcastReceiver.
- ההדגמה תכלול ביצוע שיחות זיהוי שיחה יוצאת זיהו שיחה נכנסת, שליחת SMS וקבלתו.
- תרגיל שימוש בשירות קיים במערכת שבה מתנהל הטלפון. או תרגיל ל BroadcastReceiver ושימוש בהרשאות.

### דרכי הערכה:

- בחינה מעשית או מטלת ביצוע על פי המטרות הביצועיות.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
TelephonyManager ו PhoneStateListener	3	6	9
זיהוי שיחה יוצאת.	2	4	6
קבלת ושליחת SMS	3	6	9
סה"כ שעות:	8	16	24

## פרק 18 : רכיבים מתקדמים: מולטימדיה ומצלמה, חיישנים, GPS,

### מפות.

רוב המכשירים המופעלים ע"י אנדרואיד מכילים רכיבי חישה ומקלט מערכת מיקום גלובלית. נלמד השימוש ברכיבי GPS- (Global Positioning System) באופן שטחי. אנדרואיד מספקת מרחב אפשרויות לנגינה קבצי קול, והפעלת וידאו הצגת תמונות, הקלטה קול, לכידת תמונה ווידאו.

**יעדים:** שימוש במנהל החיישנים, SensorManager, SensorEventListener, חיישני תנועה, חיישני מיקום, חיישני סביבה, מיקום ומפות, נגינה אודיו וידאו, הקלטת קול, לכידת תמונה, צילום וידאו.

### תכנים:

- מנהל החיישנים SensorManager.
- המאזין לחיישנים SensorEventListener.
- חיישני תנועה:
- משתמשים בהם לזיהוי תנועת המכשיר, כגון, להטות לנער, להניף סיבוב.

חיישנים אלו מורכבים מ-מד התאוצה "accelerometer", ומד כוון "gyroscope" שניהם צריכים להיות חלק מהחומרה. ועוד שלוש שיכולים ממושים בחומרה או בתוכנה (חיישן משיכה, תאוצה לינארית, וחיישן ויקטור הסיבוב).

• חיישני מיקום:

מערכת האנדרויד מספקת שני חיישנים כדי לאבחן מיקום המכשיר "חישן השדה המגנטי של כדור הארץ" וחישן כיוון שבנויים בחומרה.

• חיישני סביבה:

מערכת האנדרויד מספקת 4 סוגים לחישת סביבה: לחות, אור, לחץ אוויר וטמפרטורה.

• מיקום ומפות:

אנטנת GPS מאפשרת מערכת האנדרויד לקבל את הקואורדינטות העולמיות של מיקום והמכשיר. אנדרויד מאפשרת גישה למפות שמאפשר מראה ויזואלי אמיתי למיקום.

LocationManager הינה המחלקה האחראית לשימוש וניהול כלי זה.

• MediaPlayer: המחלקה האחראית להפעלת קבצי מוזיקה ווידאו.

• AudioManager: מנהל מקורות וכלי השמעת והקול.

• הקלטת קול ע"י MediaRecorder .

• המצלמה: מערכת האנדרויד מספקת כלים לשליטה במצלמה כך שניתן לצלם תמונות

או וידאו ע"י MediaRecorder.

### מטרות ביצועיות:

• התלמיד יתאר את סוגי החיישנים האפשריים.

• התלמיד יתנסה בשימוש בחיישן אחד לפחות (מקבוצת החיישנים וה-GPS), והשימוש

המאזין החיישנים.

• התלמיד יציג מפה ע"י Mapview.

• התלמיד יפעיל קבצי מידיה (קול, תמונה, ווידאו).

• התלמיד יסביר אפשרות ללכידת: קול, תמונה ווידאו, ואת המחלקות האחראיות על זה.

### דרכי הוראה:

• לסקור את סוגי החיישנים האפשריים.

• לתרגל עם התלמידים שימוש בחיישן אחד לפחות (מקבוצת החיישנים וה-GPS).

מפני שהסימולאטור לא מספק אופציות לכל החיישנים אבל מספק אופציה לשימוש ב

GPS (ע"י ה-DDMS), לכן אפשר לתרגל רק את השימוש ב-GPS.

- אם יש בכיתה מכשיר אז עדיף לבצע עוד שימוש בחיישן, ההדגמה תהיה על המכשיר עצמו.

- לתרגל הפעלת קבצי קול ווידאו והפסקתם וגם על אופציות החזרה בהפעלת כל סוג.

#### דרכי הערכה:

- מטלת ביצוע במעבדה או בבית לשימוש במקלט ה GPS כמו חישובי מרחק וחישובי מהירות וכו'. ניתן להוסיף הפעלת קול כתגובה לאירועים דוגמת לחיצה על כפתור.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
סקירת סוגי החיישנים.	2	4	6
אופן השימוש בחיישנים ע"י מנהל החיישנים וההאזנה להם.	3	6	9
ה GPS כדוגמא ומפות.	4	8	12
מדיה	3	6	9
סה"כ שעות:	12	24	36

## פרק 19 : גרפיקה ואנימציה

בפרק זה נלמד לעצב רכיבי תצוגה (View) על ידי ציור דו-ממדי (2D), שנבצע באמצעות כלים העומדים במחלקה Canvas או יצירת רכיבים חדשים הניתנים לציור Drawable. בנוסף נבצע אנימציה ואינטראקציה עם המשתמש.

**יעדים:** ציור וגרפיקה בעזרת XML ו Canvas, ביצוע אנימציות ב XML ועל גבי SurfaceView.

#### תכנים:

- ציור בעזרת XML.
- ציור על View: מתבצע ע"י דריסת הפעולה View.onDraw(). משתמשים בשיטה זו כאשר הרכיבי לא צריך לבצע אנימציה מהירה או מורכבת, יותר מתאים לגרפיקה יותר סטטית וקבועה.



- Canvas : זאת המחלקה שמהווה את הממשק היוזאלי של הרכיבי הגרפי. מחלקה זו מכילה את הפעולות האחראיות על ציור צורות גרפיות שונות.
- Paint: מחלקה המחזיקה במידע על צבע וסגנון כדי לתאר איך מצירים צורות גיאומטריות, טקסט ומפת סיביות Bitmap.
- BitmapFactory לטעינת תמונות לתוך עצמם מטיפוס מפת סיביות Bitmap.
- ציור על SurfaceView: מחלקה המתאימה לביצוע אנימציות. הייחודיות שלה בזה שמציעה תהליכון (Thread) עצמאי לביצוע הצביעה כך שהרכיב לא ממתין בתורו עד שהמערכת מציירת אותו.
- SurfaceHolder לפתרון בעיית העדכון (השנוי) הגרפי על SurfaceView.
- Drawables: אנדרואיד מציעה חבילת מחלקות הניתנים לציור בדו-מימד android.graphics.drawable. בחבילה זו ניתן למצוא BitmapDrawable, PictureDrawable, ShapeDrawable, LayerDrawable ועוד אחרות. רכיבים אלו אפשר להשתמש במשאבים קיימים כמו תמונות, ולצור אותם ע"י XML.
- חבילה זו מאפשרת אנימציה frame-by-frame שניתן לכתוב אותה בשיטה XML, בעזרת המחלקה AnimationDrawable.
- אנימציה ע"י AnimationDrawable.
- אנימציה בשיטת View Animation.
- אנימציה בעזרת תהליכון רגיל (Thread, Timer) על SurfaceView שבאחריות SurfaceHolder.

### מטרות ביצועיות:

- התלמיד ישתמש בכלים הגרפיים העומדים לרשותו.
- התלמיד יבצע ציור וצביעה ויצירת רכיבים גרפיים: אם זה בשיטת XML או שימוש ב Canvas.
- התלמיד יבצע סוג אחד של אנימציה: AnimationDrawable, ViewAnimation או ע"י ציור על View ע"י הרחבה של View או SurfaceView.

### דרכי הוראה:

- התלמידים יתרגלו את שיטות הצביעה והציור ב XML, באופן מעשי.
- לבצע דוגמת אנימציה ע"י AnimationDrawable. ואנימציה בשיטת View Animation.

- לבצע הרחבה למחלקה View ולבצע דריסה לפעולה onDraw(Canvas canvas). ולהשתמש לב Canvas לבצע צביעה וציור, לדוגמא צביעת רקע, ציור צורות (קו, מלבן, מעגל וכו') תוך שימוש מחלקה Paint, ציור תמונת Bitmap.
- בשלב שני לשנות את הירחשה בדוגמא הקודמת מ View ל SurfaceView, ולזהות שלא התבצע ציור. וכאן בא הצורך והתפקיד של ה SurfaceHolder.
- לבצע אנימציה כתגובה לאירוע מגע במסך. לדוגמא הזזת תמונה או עיגול על המסך בהתאם לאירוע המגע.
- לבצע אנימציה שמשתמשת ביותר מרכיב אנימציה על המסך. לדוגמא שני עיגולים או תמונות שזזות באופן שרירותי על המסך.

#### דרכי הערכה:

- מטלת ביצוע מודרכת ליצירת רכיב מעוצב. או ממשק עם רכיבים שמבצעים אנימציות בעזרת AnimationDrawable ו View Animation.
- מטלת ביצוע לצור אנימציה ותגובה לאירוע המגע כהרחבה SurfaceView.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
ציור בעזרת XML	1	2	3
ציור על View, Canvas, Paint, BitmapFactory	2	4	6
ציור על SurfaceView, SurfaceHolder	2	4	6
אנימציה ע"י AnimationDrawable	1	2	3
אנימציה בשיטת View Animation	2	4	6
אנימציה בSurfaceView, SurfaceHolder	2	4	6

24	16	8	סה"כ שעות:
----	----	---	------------

## פרק 20 : הנחיה והגשת עבודת גמר

**יעדים:** ביצוע פרויקט הגמר, קביעת את הדרישות ורמה של עבודת הגמר.

בשלב זה מבצעים ניתוח פרויקט ע"י תרשימים (תרשים מסכים ו מחלקות ע"י UML) , ובצוע מעשי של הפרויקט. התלמידים מתחילים בביצוע של הפרויקט שלהם.

### תכנים:

- ההבדל בין הרמות של עבודות הגמר.
- בחירת נושא לפרויקט מלווה.
- ביצוע ניתוח לפרויקט שכולל את הרמות הבאות:
  - רמת הממשק: הגרפי מסכים ותרשים זרימה בין המסכים.
  - רמת התכנות: זיהוי ישויות, עצמים ורכיבים "יחודיים של אנדרואיד. קביעת המחלקות שצריכים (כולל המחלקות של ממשק המשתמש הגרפי), ולזהות את אלה שקיימות וניתן להשתמש בהן ואלה שצריך לצור. ולצור תרשים UML לתיאור את המחלקות והקשרים ביניהם.
  - רמת הלוגיקה: זרימת פעילות היישום, חוקים וכדומא בפרט כשמדובר במשחקים.
  - רמת הנתונים: תיאור הנתונים שצריכים להישמר על המכשיר, דרך האחסון שנבחרה (הגדרות, קבצים, SQLITE).
  - רמת השירותים והתכנות המקבילי: תיאור את הצורך ברכיבים או קטעי בקוד שצריכים לרוץ ברקע (שירותים לדוגמא) והצורך בתכנות מקבילי.
- קביעת לוח זמנים שמתאר את סדר ביצוע הפרויקט החל מההצעה עד תיק הפרויקט.
- ביצוע הפרויקט שנבחר עד תומו.
- כתיבת תיעוד של הקוד בשיטת ה Javadoc בזמנית עם כתיבת הקוד.
- ביצוע כתיבת תיק פרויקט.

## מטרות ביצועיות:

- התלמיד יתאר את ההבדל בין הרמות של עבודות הגמר.
- התלמיד יתאר את דרישות וצרכי הרמה של עבודת הגמר שבחר לבצע.
- התלמיד יתנסה ביחד עם המורה בביצוע ניתוח לפרויקט המלווה לרמות במפורטות בתכנים.
- התלמיד יבצע את הניתוח לפרויקט שלו בעבודת הגמר על פי הרמות שמפורטות בתכנים.
- התלמיד יתנסה בקביעת לו"ז וסדר לביצוע הרכיבים של הפרויקט.
- התלמיד יתנסה בביצוע הפרויקט המלווה ביחד עם המורה.
- התלמיד יתרגל לתעד את כל המחלקות והשיטות והתכונות בשיטת ה Javadoc.
- התלמיד יצור תיק פרויקט לפרויקט המלווה.

## דרכי הוראה:

- על המורה לבחור הצעת פרויקט כדי לבצע אותו עם התלמידים משלב הניתוח עד הביצוע. הפרויקט יכול לעמוד בדרישות המינימאליות של רמת עבודת הגמר.
- על המורה להכין באופן כתוב את הצעת הפרויקט המלווה שתהיי דוגמא מייצגת לתלמידים לאופן הגשת ההצעה.
- המורה יבצע ניתוח לפרויקט ביחד עם התלמידים תוך כדי רישום וציור. המטרה לתוצר של מסמך שכולל תרשים זרימת מסכים, תרשים UML, רכיבים "יחודיים" של אנדרואיד ותיאור הנתונים הנשמרים.
- ליצור תרשים שמקשר את כל הרכיבים והתרשימים שבוצעו.
- לדרוש מהתלמידים להגיש הצעה לעבודות גמר שלהם בשלב ראשון ומאחר האישור של המורה לדרוש ביצוע ניתוח התרגול.
- אפשר לבצע את שלב הניתוח של הצעות התלמידים בקבוצות, כך שכל קבוצה תבצע ניתוח של כל חברי הקבוצה.
- להתחיל בביצוע של הפרויקט המלווה ביחד עם התלמידים ומומלץ לאמץ את "מודל הקליפות" על פי השלבים:
  - o שלב 1: בכל שלב מגדירים יעד שניתן להשיג בזמן קצר באופן סביר ומתחילים ביישומ.
  - o שלב 2: כשמסיימים בודקים שהיעד הושג על ידי ביצוע סדרת בדיקות.

o שלב 3: יוצרים גרסה חדשה (עוזבים את הישנה) ומתחילים משלב 1.

בשיטה זו הפרויקט יגדל שלב אחרי שלב "קלפיה אחרי קלפיה", ותמיד תהיה גרסה אחרונה יציבה ותקינה כך שאם מאבדים או מקלקלים את הגרסה האחרונה תמיד יהיה פרויקט בגרסה של שלב אחד לפני.

מומלץ להחזיק מחברת ביצוע פרויקט שכוללת את הניתוח והתרשימים, מחברת זו עוזרת בתכנון הזמן ואת השלב הבא ומה נשאר לבצע. במחברת זו מציינים בכל פעם שניגשים להמשך ביצוע הפרויקט את מספר הגרסה והיעד, ובסוף מציינים אם הושג כן או לא, ומתכננים ומציינים את הצעד הבא בגרסה הבאה.

- מומלץ שהמורה יקבל את כל המידע לפני ביצוע הפרוייקט על ידי התלמידים.

#### דרכי הערכה:

- להעריך את עבודת התלמידים לפי השלבים שצריך להגיש: הצעת פרויקט, ניתוח וכו'.
- לתת לתלמידים משימה בה יצטרכו לבצע שלב בפרויקט המלווה.
- לתת לכל תלמיד בנפרד ביצוע שלב מינימאלי בפרויקט העצמי שלו ולהעריך.
- לתת לו"ז ביצוע הפרויקט האישי לכל תלמיד ולהעריך אותו על מידת העמידה בלו"ז.

#### חלוקת שעות:

נושא	שעות עיוני	שעות מעשי	סה"כ
הבדל בין הרמות	1	0	1
ניתוח פרויקט מלווה.	10	18	28
קביעת לוז	2	0	2
ביצוע הפרויקט.	8	25	33
תיק פרויקט	3	5	8
סה"כ שעות:	24	48	72

### ההבדלים בין רמת העבודה 5,3,1 יח"ל:

היקף הפרויקט יקבע על פי גודלו ומורכבותו כלומר:

- שימוש בעקרונות הנדסת תוכנה ותכנות מונחה עצמים: הורשה, פולימורפיזם, ממשקים, מחלקות מופשטות, יחס הכלה והורשה ומספר המחלקות.
- שימוש באבני היסוד של יישום באנדראיד: Activity, Service, Intent, BroadcastReceiver, ContentProvider.
- אחסון נתונים הקשורים ליישום.
- נושאים מתקדמים: טלפוניה, גרפיקה, אנימציה, חיישנים, GPS, מפות, מצלמה, מיקרופון.

### פרויקט גמר ברמה של 1 יח"ל,

מערכת תהווה יישום משתמש ותכלול את המרכיבים הבאים:

- לפחות שתי מחלקות ושימוש בהורשה פשוטה (מחלקת הבסיס והמחלקה הנגזרת) ושימוש במימוש ממשק (interface) מהקיימים כמו מאזינים וכו'.
- ממשק משתמש גרפי לפחות, שני ממשקים/מסכים.
- טיפול באירועים.
- אבני היסוד: Activity, Intent.
- אחסון נתונים (שמירה ושליפה), לפחות בדרך אחת.

### פרויקט גמר ברמה של 3 יח"ל,

מערכת תהווה יישום משתמש ותכלול את המרכיבים הבאים:

- לפחות 3 מחלקות ושימוש בהורשה פשוטה (מחלקת הבסיס והמחלקה הנגזרת) ושימוש במימוש ממשק (interface) מהקיימים כמו מאזינים וכו'.
- ממשק משתמש גרפי לפחות, שני ממשקים/מסכים.
- שימוש בתפריטים או תיבות דו-שיח.
- טיפול באירועים.
- אבני היסוד: Activity, Intent, Service (ניתן לצור שירות ולהפעילו, או להשתמש בשירות קיים).
- אחסון נתונים (שמירה ושליפה), לפחות בדרך אחת.

### עבודת גמר של 5 יח"ל,

מערכת תהווה יישום משתמש ותכלול את המרכיבים הבאים:

- לפחות 5 מחלקות ושימוש בהורשה פשוטה (מחלקת הבסיס והמחלקה הנגזרת) ושימוש במימוש ממשק (interface) מהקיימים כמו מאזינים או יצירת חדש וכו'.
- ממשק משתמש גרפי לפחות, שני ממשקים/מסכים.
- שימוש בתפריטים וב תיבות דו-שיח.
- טיפול באירועים.
- אבני היסוד: Activity, Intent, Service (ניתן לצור שירות ולהפעילו, או להשתמש בשירות קיים) ואחד מהשניים: ContentProvider (שימוש בקיים או יצירת חדש) או BroadcastReceiver (שימוש בקיים או יצירת חדש).

- אחד מהנושאים המתקדמים: טלפוניה, גרפיקה, אנימציה, חיישנים, GPS, מפות, מצלמה, מיקרופון.
- אחסון נתונים (שמירה ושליפה), לפחות בדרך אחת.

מומלץ להתחיל את עבודת הגמר בתקופת חופשת הפסח בכיתה י"א. בתקופה זו ייבחר התלמיד את נושא עבודת/פרוייקט הגמר, בהנחיית המורים ובעזרתם.

העבודה תתבצע במסגרת השעות המוקצבות, בהנחיית מורי בית הספר או בהנחיית מומחים שמחוץ לבית הספר.

מרכז המגמה לטכנולוגיות מידע ומורים בחלופה, ינהלו את העבודות, יקבעו את לוחות הזמנים לביצוע העבודות, יתאמו את ההנחיה ויעריכו את העבודות והכנתן כחלק מקביעת הציון השנתי.

הציון השנתי יקבע ע"י המורה המנחה את עבודות הגמר וישקף את רמת עבודת הגמר, צורת ביצועה ע"י המורה ועמידה בלוח זמנים שנקבע להשלמת כל המשימות הכרוכות בעבודת הגמר.

### משימות ולוח זמנים

#### משימות ולוח זמנים לביצוע הפרויקט (י"ב)

בתחילת כיתה י"ב יקבל כל תלמיד דף הדרכה המכיל משימות ולוח זמנים.

מועד הביצוע	משימה
ספטמבר - אוקטובר	כתיבת הצעת פרויקט: בחירת נושא, כתיבת תסריט
אוקטובר - נובמבר	ניתוח הבעיה, זיהוי עצמים, מחלקות וקשרים, תיאור מסכי ממשק המשתמש הגרפי ותיאור הנתונים שצריך לשמור. התוצר של שלב זה:  1. תרשים UML המתאר את כל המחלקות והקשרים ביניהם. (ירושה, הכלה וכו')  2. תאור מבנה כל מסך ותרשים זרימה של המסכים למעגל חיים מושלם.  3. תאור מבנה הנתונים שצריך לאחסן ואת שיטת האחסון.  4. זיהוי את אבני היסוד שצריכים בביצוע היישום (Activity , Service , Intent , ContentProvider , BroadcastReceiver).
נובמבר – פברואר	תחילת פיתוח: הפרדה בין עצמים ומחלקות הממשק הגרפי ושאר המחלקות. בחירת סדר מימוש המחלקות, מימוש הדרגתי של מחלקות,

משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

	תוך כדי ביצוע בדיקות ביניים. מומלץ לבצע את שלב בניית הפרויקט כמודל הקליפות: גרסה חדשה, ביצוע השינוי ובסוף בדיקה.
פברואר-מרץ	השלמת פיתוח הפרויקט: השלמת מעגל חיי האפליקציה, ובדיקת תקינות.
פברואר - מרץ	כתיבת ספר הפרויקט

**גיליונות ההערכה בתכנון ותכנות מערכות טלפונים חכמים תחת מערכת ההפעלה  
אנדרואיד**

**דברי הסבר**

בגיליון שני חלקים עיקריים: בדיקת רמת הפרויקט והערכת הפרויקט. מטרת החלק הראשון היא לבדוק האם הפרויקט עונה לדרישות פרויקט ברמה המוגדרת. אם הפרויקט לא עונה לדרישות החובה (1-3) הוא אינו מתקבל כפרויקט גמר וציונו נקבע 0. אם הפרויקט עונה לדרישות 1-3 ולא עונה לדרישות האחרות, אז הוא זכאי לבדיקת התאמתו לרמת פרויקט נמוכה יותר (עבור הפרויקטים 5 ו-3 יחידות).

החלק השני- הערכת הפרויקט. ניתן להעריך את הפרויקט בתנאי שהוא עבר דרישות בדיקת רמת הפרויקט.

סעיף ד' של גיליון הערכה מאפשר להוסיף עד 10 נקודות בonus עבור תוספות בפרויקט מעבר לתוכנית הלימודים המותאמת לרמה העבודה: שימוש בידע מחוץ למדעי המחשב, שימוש באלגוריתמים מתקדמים או יישום מבנה נתונים מורכבים, בינה מלאכותית, מעבר בין מספר רמות בפרויקט וכדומה. הציון הכולל כמובן אינו יכול לעבור 100.

**גיליון הערכה לעבודת גמר 5 יח"ל, בתכנון ותכנות מערכות טלפונים חכמים תחת  
מערכת ההפעלה אנדרואיד**

**פרטי התלמיד**

שם בית הספר	
שם התלמיד	
נושא העבודה	

**בדיקת רמת הפרויקט**

- הדרישות 1 – 3 חייבים להתבצע. אם לא, ציון הפרויקט יהיה 0



משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

- הפרויקט נחשב ברמה של 5 יח"ל אם כל הדרישות 4-8 מתקיימות. אחרת הפרויקט אמור להיבדק להתאמתו לרמה 3 יח"ל.

דרישות	נוכחות בפרויקט
1 <b>בזמן ההצגה על התכניות להיות במצב עבודה</b>	
2 <b>התוכנה מהווה אפליקציה לטלפונים חכמים שרצים במערכת הפעלה אנדרואיד.</b>	
3 <b>הפרויקט מהווה תכנית אינטראקטיבית, המנוהלת ע"י ממשק גרפי למשתמש.</b>	
4 <b>שימוש מתקדם בתכנות מונחה עצמים (ירושה, פולימורפיזם וכו')</b>	
5 <b>שימוש בתפריטים ובתיבות דו-שיח</b>	
6 <b>שימוש באבני היסוד: Intent, Activity, Service ואחד מהשניים: BroadcastReceiver או ContentProvider.</b>	
7 <b>אחד מהנושאים המתקדמים: טלפוניה, גרפיקה, אנימציה, חיישנים, GPS, מפות, מצלמה, מיקרופון.</b>	
8 <b>אחסון נתונים (שמירה ושליפה), לפחות בדרך אחת</b>	

הערכת פרויקט

נימוקים והערות	ניקוד בפועל	ניקוד מרבי מתוך 100	
		(25)	<b>חלק א' – תיק עבודת הגמר</b>
		2	מטרת התוכנה.
		3	תיאור מטרות והאופציות שנותנת תוכנה זו למשתמש.
		4	תיאור מסכים כל מסך, תכולת ותפקיד. תרשים זרימה של המסכים.
		3	דרישות/מוגבלויות להפעלת התוכנה. הרשאות שצריכה התוכנה. חומרה מיוחדת: חיישנים, GPS וכו'.
		4	תיאור המחלקות תוך שימוש ב-uml ותיעוד JAVADOC.

משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

		4	תיאור ארגון הנתונים המאוחסנים ואת שיטת השמירה הנבחרת לצורך זה.
		5	מדריך משתמש ליישום
		(55)	<b>חלק ב' - תכנות</b>
		3	קריאות התכנית, תיעוד וארגון הקבצים
		15	תכנות מונחה עצמים: חלוקה למחלקות, ירושה הכלה וכו', סגנון התכנות: בחירת משתנים, תיעוד, חלוקה לפעולות וכו'.
		8	שימוש בתפריטים ובתיבות דו-שיח.
		6	שימוש באירועים (שימוש בפקדים ומאזינים, אירועי מקשים, אירועי מצביע)
		8	שימוש באבני היסוד: Intent, Activity, Service ואחד מהשניים: ContentProvider או BroadcastReceiver.
		5	שימוש באחסון נתונים .
		4	מקוריות, יצירתיות, תחכום התוכניות
		6	ממשקים והתאמתם לדרישות היישום. מוצגות (הנדסת אנוש, אסתטיקה, נוחות שימוש, שימושיות, ממשק ברור)
		(20)	<b>חלק ג' - הצגה והגנה</b>
		5	הצגת עבודת הגמר
		15	שליטה בחומר עיוני: תשובות לשאלות הבוחן
		(10)	<b>חלק ד' – בונוס (הנושאים מעבר לתוכנית הלימודים המותאמת לרמה העבודה, שימוש בידע מחוץ למדעי המחשב, שימוש באלגוריתמים מתקדמים או יישום מבנה נתונים מורכבים וכו')</b>
		מעבר ל-100	
		ציון סופי*	

תאריך	שם הבוחן	חתימה

גיליון הערכה לפרויקט/עבודת גמר 3 יח"ל, בתכנון ותכנות מערכות טלפונים חכמים  
תחת מערכת ההפעלה אנדרואיד

### פרטי התלמיד

שם בית הספר	
שם התלמיד	
נושא העבודה	

### בדיקת רמת הפרויקט

- הדרישות 1 – 3 חייבים להתבצע. אם לא, ציון הפרויקט יהיה 0
- הפרויקט נחשב ברמה של 3 יח"ל אם כל הדרישות 4-7 מתקיימות. אחרת הפרויקט אמור להיבדק להתאמתו לרמה 1 יח"ל.

נוכחות בפרויקט	דרישות	
	בזמן ההצגה על התכניות להיות במצב עבודה	1.
	התוכנה מהווה אפליקציה לטלפונים חכמים שרצים במערכת הפעלה אנדרואיד.	2.
	הפרויקט מהווה תכנית אינטראקטיבית, המנוהלת ע"י ממשק גרפי למשתמש.	3.
	שימוש מתקדם בתכנות מונחה עצמים (ירושה, פולימורפיזם וכו')	4.
	שימוש בתפריטים או בתיבות דו-שיח	5.
	אבני הייסוד: Intent, Activity ו Service (ניתן לצור שירות ולהפעילו, או להשתמש בשירות קיים).	6.
	אחסון נתונים (שמירה ושליפה).	7.

### הערכת פרויקט

משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

נימוקים והערות	ניקוד בפועל	ניקוד מרבי מתוך 100	
		(25)	<b>חלק א' – תיק עבודת הגמר</b>
		2	מטרת התוכנה.
		3	תיאור מטרות והאופציות שנותנת תוכנה זו למשתמש.
		4	תיאור מסכים כל מסך, תכולת ותפקיד. תרשים זרימה של המסכים.
		3	דרישות/מוגבלויות להפעלת התוכנה. הרשאות שצריכה התוכנה. חומרה מיוחדת: חיישנים, GPS וכו'.
		4	תיאור המחלקות תוך שימוש ב-uml ותיעוד JAVADOC.
		4	תיאור ארגון הנתונים המאוחסנים ואת שיטת השמירה הנבחרת לצורך זה.
		5	מדריך משתמש ליישום
		(55)	<b>חלק ב' - תכנות</b>
		3	קריאות התכנית, תיעוד וארגון הקבצים
		15	תכנות מונחה עצמים: חלוקה למחלקות, ירושה הכלה וכו', סגנון התכנות: בחירת משתנים, תיעוד, חלוקה לפעולות וכו'.
		8	שימוש בתפריטים ובתיבות דו-שיח.
		6	שימוש באירועים (שימוש בפקדים ומאזינים, אירועי מקשים, אירועי מצביע)
		8	שימוש באבני היסוד: Intent, Activity, Service.
		5	שימוש באחסון נתונים.
		4	מקוריות, יצירתיות, תחכום התוכניות
		6	ממשקים והתאמתם לדרישות היישום. מוצגות (הנדסת אנוש, אסתטיקה, נוחות שימוש, שימושיות, ממשק ברור)

משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

		(20)	חלק ג' - הצגה והגנה
		5	הצגת עבודת הגמר
		15	שליטה בחומר עיוני: תשובות לשאלות הבוחן
		(10)	חלק ד' – בונוס (הנושאים מעבר לתוכנית
		מעבר ל- 100	הלימודים המותאמת לרמה העבודה, שימוש בידע מחוץ למדעי המחשב, שימוש באלגוריתמים מתקדמים או יישום מבנה נתונים מורכבים וכו')
		ציון סופי*	

תאריך	שם הבוחן	חתימה

גיליון הערכה לפרויקט/עבודת גמר 1 יח"ל, בתכנון ותכנות מערכות טלפונים חכמים  
תחת מערכת ההפעלה אנדרואיד

**פרטי התלמיד**

שם בית הספר	
שם התלמיד	
נושא העבודה	

**בדיקת רמת הפרויקט**

- הדרישות 1 – 3 חייבים להתבצע. אם לא, ציון הפרויקט יהיה 0
- הפרויקט נחשב ברמה של 1 יח"ל אם כל הדרישות 4-6 מתקיימות. אחרת הפרויקט אמור להיפסל ציון הפרויקט יהיה 0.

דרישות	נוכחות בפרויקט
1. בזמן ההצגה על התכניות להיות במצב עבודה	
2. התוכנה מהווה אפליקציה לטלפונים חכמים שרצים במערכת הפעלה אנדרואיד.	
3. הפרויקט מהווה תכנית אינטראקטיבית, המנוהלת ע"י ממשק גרפי למשתמש.	

משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

4.	שימוש מתקדם בתכנות מונחה עצמים (ירושה, פולימורפיזם וכו')	
5.	אבני היסוד: Intent, Activity.	
6.	אחסון נתונים (שמירה ושליפה).	

הערכת פרויקט

נימוקים והערות	ניקוד בפועל	ניקוד מרבי מתוך 100	
		(25)	<b>חלק א' – תיק עבודת הגמר</b>
		2	מטרת התוכנה.
		3	תיאור מטרות והאופציות שנותנת תוכנה זו למשתמש.
		4	תיאור מסכים כל מסך, תכולת ותפקיד. תרשים זרימה של המסכים.
		3	דרישות/מוגבלויות להפעלת התוכנה. הרשאות שצריכה התוכנה. חומרה מיוחדת: חיישנים, GPS וכו'.
		4	תיאור המחלקות תוך שימוש ב-uml ותיעוד JAVADOC.
		4	תיאור ארגון הנתונים המאוחסנים ואת שיטת השמירה הנבחרת לצורך זה.
		5	מדריך משתמש ליישום
		(55)	<b>חלק ב' - תכנות</b>
		3	קריאות התכנית, תיעוד וארגון הקבצים

משרד החינוך  
מנהל מדע וטכנולוגיה  
מגמת הנדסת תוכנה

		15	תכנות מונחה עצמים: חלוקה למחלקות, ירושה הכלה וכו', סגנון התכנות: בחירת משתנים, תיעוד, חלוקה לפעולות וכו'.
		8	שימוש ברכיבי ממשק משתמש גרפי.
		6	שימוש באירועים (שימוש בפקדים ומאזינים, אירועי מקשים, אירועי מצביע)
		8	שימוש באבני הייסוד: Intent, Activity.
		5	שימוש באחסון נתונים .
		4	מקוריות, יצירתיות, תחכום התוכניות
		6	ממשקים והתאמתם לדרישות היישום. מוצגות (הנדסת אנוש, אסתטיקה, נוחות שימוש, שימושיות, ממשק ברור)
		(20)	<b>חלק ג' - הצגה והגנה</b>
		5	הצגת עבודת הגמר
		15	שליטה בחומר עיוני: תשובות לשאלות הבוחן
		(10) מעבר ל- 100	<b>חלק ד' – בונוס (הנושאים מעבר לתוכנית הלימודים המותאמת לרמה העבודה, שימוש בידע מחוץ למדעי המחשב, שימוש באלגוריתמים מתקדמים או יישום מבנה נתונים מורכבים וכו')</b>
		ציון סופי*	

תאריך	שם הבוחן	חתימה

מפרט הגשת פרויקט/עבודת גמר, תכנון ותכנות מערכות טלפונים חכמים תחת מערכת  
ההפעלה אנדרואיד.

<b>א. ספר העבודה:</b>
<p>1. <b>דף שער:</b></p> <ul style="list-style-type: none"><li>• שם הפרויקט</li><li>• שם המבצע</li><li>• שם המנחה</li><li>• שם בית הספר</li><li>• תאריך ההגשה</li></ul>
<p>2. <b>תוכן עניינים</b></p>
<p>3. <b>הקדמה (מבוא)</b></p> <ul style="list-style-type: none"><li>• תאור מילולי קצר של הפרויקט המסביר לקורא את מהות העבודה. במקום כתיבת תיאור קצר לפרויקט ניתן להציב את דף הצעת הפרויקט שנכתב בתחילת הפיתוח.</li><li>• קהל היעד לשימוש בתוכנה (אם פרויקט מיועד לקהל ספציפי)</li><li>• הרקע למשימה (אם זה משמעותי למהות הפרויקט)</li></ul>
<p>4. <b>תיאור התוכנה ומטרותיה :</b></p> <p>מטרת התוכנה.</p> <p>תיאור האופציות שנותנת תוכנה זו למשתמש.</p> <p>תיאור מסכים כל מסך, תכולה ותפקיד.</p> <p>תרשים זרימה של המסכים.</p> <p>דרישות/מוגבלויות להפעלת התוכנה: הרשאות, חומרה (חיישנים, רשת, GPS) וכו'.</p>
<p>5. <b>נתונים:</b></p> <p>הסבר/תיעוד חלוקת הקבצים השונים לספריות.</p> <ul style="list-style-type: none"><li>• מבניות, מדולריות וחלוקה למחלקות:</li><li>○ תאור מבני נתונים.</li></ul> <p>תיאור קשרי המחלקות ע"י תרשים UML.</p> <p>תיעוד המחלקות (ממשק המחלקה), ע"י שימוש בתוכנת JAVADOC שמייצרת קבצי תיעוד ב- HTML.</p> <p>תיאור מבנה/מחלקות הנתונים המיועדים לאחסון. (לדוגמא: מחלקת נתוני שחקן, מחלקת הגדרות התוכנה וכו') (אפשר להשתמש בתוצר של JAVADOC למחלקה זו)</p> <p>תיאור ארגון הנתונים הנשמרים ושיטת השמירה (מיקום, שמות של קבצים, תיאור טבלאות והשדות אם יש שימוש ב- SQLITE).</p>



<p>6. <b>שימוש באבני היסוד:</b></p> <ul style="list-style-type: none"> <li>• Activity, בפרט אם השתמש ב Activity for Result.</li> <li>• Intent, סוגי ומטרות המסרים שהשתמש בהן. לפרט איפה השתמש במסרים ולאילו מטרות, בפרט אם יש שימוש בהעברת נתונים ע"י מסרים, או הפעלת פעילות או שירות.</li> <li>• Service, הסבר מפורט על המטרה/הצורך בשירות, האם השירות קיים או שנוצר ע"י התלמיד, מתי מופעל ומתי מופסק.</li> <li>• ContentProvider, המטרה/הצורך מהשימוש בספק תוכן, האם השימוש בספק תוכן קיים או יצירת ספק תוכן חדש ע"י התלמיד והשימוש בו.</li> <li>• BroadcastReceiver, המטרה/הצורך מהשימוש במקלט שידורים. תיאור השימוש במקלט קיים במערכת או יצירת חדש.</li> <li>• הרשאות, תיאור ההרשאות שצריך היישום כדי שיפעל ומשמעות כל הרשאה.</li> </ul>
<p>7. <b>מדריך משתמשים:</b> מדריך זה מיועד למשתמש. תפקידו להנחות משתמש כיצד להפעיל את התוכנה. הוא אמור להיות מנוסח בשפה הרגילה שאינה מכילה מושגים במדעי המחשב.</p> <p>הסבר קצר על מהות התוכנה. הסבר כיצד להתקין ולהריץ את התוכנה. תיאור כיצד מפעילים ממשק המשתמש: פירוט משמעות ותוכן כל מסך כולל התפריטים(עץ תפריטי, מפת הפעולות וכדומה). (כדאי לעשות זאת בעזרת צילומי המסכים הרלוונטיים ומתן הסברים על פניהם). תיאור האופציות לשינוי הגדרות בתוכנה (רקע, צבע, כוון כתיבה, הגדרות משתמש/שחקן וכו').</p>
<p>8. <b>רקע תיאורטי (מדעי)</b></p> <p>אם בפרויקט נעשה שימוש בידע מדעי או אלגוריתם מיוחד במדעי המחשב, יש לתאר אותו כאן: נוסחאות, חוקים, תיאוריות, אלגוריתמים וכו'.</p>
<p>9. <b>סיכום אישי</b></p> <p>בפרק הזה יופיע סיכום אישי של מבצע הפרויקט. נקודות ההתייחסות האפשריות:</p> <ul style="list-style-type: none"> <li>• מהם הצדדים החזקים ומהם הצדדים החלשים של הפרויקט.</li> <li>• מה היה ניתן לשנות בפרויקט ומה היה ניתן להוסיף לפרויקט במידה והיה זמן נוסף לעבוד עליו.</li> <li>• מה למדת מהפרויקט: מבחינת ידע בתכנות של טלפונים ניידים, מבחינת תכנות, מבחינת התמודדות עם פרויקט גדול, מבחינת ארגון העבודה.</li> </ul>
<p>10. <b>נספחים</b></p> <ul style="list-style-type: none"> <li>• תדפיסים (listings) של כל הקודים בפרויקט.</li> <li>• מילון מושגים (אם יש מושגים מיוחדים).</li> <li>• רשימת מקורות (רשימת כל המקורות שהתלמיד השתמש מעבר לתכנית הלימודים).</li> </ul>

**ב. תכנות**

**התוצר של פרק זה הוא יישום שעובד התואם לאפיונים שנקבעו לפי רמת הפרויקט.**

קריאות התכנית: שמות משמעותיים לשגרות ושדות, תיעוד פנימי (המתאים לכללי JAVADOC), הערות והסברים בגוף התכנית.

ממשקים והתאמתם לדרישות היישום ובהתאם למכשיר המיועד (לחצנים, מסך מגע): ממשקים נוחים, עיצוב מסכים והנדסת אנוש, עיצוב היישום לנוחות המשתמש, עיצוב הפלטים. הנדסת תוכנה ופיתוח מונחה עצמים בשפת-JAVA.

**ג. הצגה**

התלמיד יריץ את עבודתו תוך מתן הסברים נלווים.  
ההרצה תתבצע על האימיוליטור ועדיף גם על טלפון נייד שייבחר התלמיד.  
הוכחת ידע והבנת החומר הינם תנאי הכרחי לקבלת ציון עובר.