**המכללה האקדמית להנדסה בראודה**

המחלקה להנדסת תוכנה - פרויקט במציאות רבודה – אביב התשפה

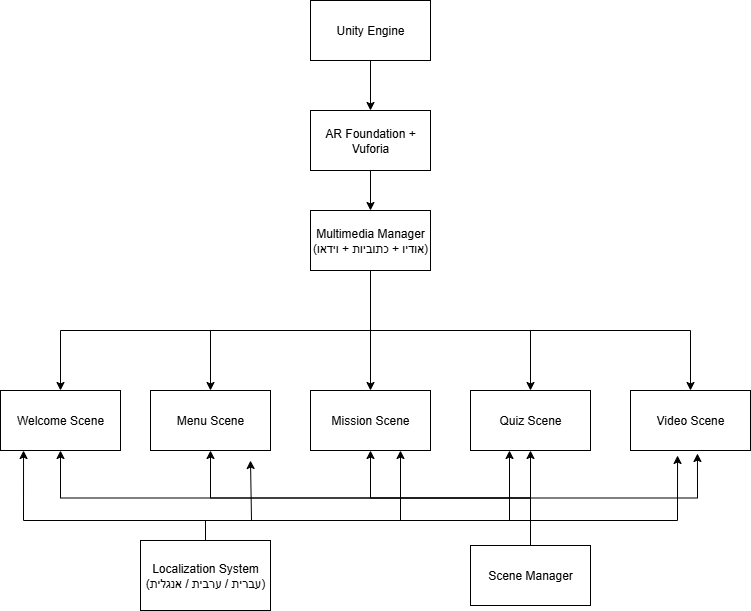
הגשת הפרויקט – להגשה עד 13.7.25

1. פרטו את התפקידים והמשימות השונות שביצע כל חבר צוות במהלך הקורס.

|  |  |  |
| --- | --- | --- |
| **שם חבר הצוות** | **משימות שהוקצו** | **משימות שהושלמו** |
| **מוחמד חטיב** | **WelcomeScene – מסך פתיחה באפליקציה.**  **MissionScene – מסך שמתאר את המשימה של האפליקציה שלנו. אודיו עם טקסט בהתאם לשפה שנבחרה + זיהוי של התמונות ומראה הסבר עליהם עם אודיו.**  **MenuScene – מסך תפריט , שיעביר בין המסכים.**  **תרגילי בית + מצגת.** | **הכל** |
| גאד טאהא | **VideoScene – מסך שמכיל וידו שמסביר על חייו יגאל אלון.**  **QuizScene – מסך חידון , משחוק למשתמש.**  **תרגילי בית + מצגת.** | הכל |

2.הציגו את דרישות המערכת יש להתבסס על המבנה של asana (<https://asana.com/resources/software-requirement-document-template>

* דרישות פונקציונליות (10 דרישות)
  1. **האפליקציה מאפשרת** למשתמש לבחור שפה מתוך שלוש אפשרויות: עברית, ערבית, אנגלית.
  2. האפליקציה מאפשרת ניווט בין סצנות שונות דרך תפריט ראשי הכולל שני כפתורים: האחד לסצנת המשימה, והשני לסצנת הווידאו.
  3. **האפליקציה מאפשרת** השמעת קובץ שמע מותאם לשפה הנבחרת עם שליטה בהפעלה, עצירה וחזרה וקדימה של 10 שניות.
  4. **האפליקציה מציגה** כתוביות המתאימות לתוכן השמע בהתאם לשפה שנבחרה.
  5. האפליקציה מאפשרת זיהוי אוטומטי של 5 תמונות פיזיות באמצעות מצלמת המכשיר, ומשמיעה קובץ שמע ומציגה טקסט תואם לכל תמונה בהתאם לשפה הנבחרת.
  6. **האפליקציה מאפשרת** צפייה בסרטון וידאו המתאר את סיפור חייו של יגאל אלון.
  7. האפליקציה שומרת את השפה שנבחרה בתחילת השימוש ומשתמשת בה בכל הסצנות מבלי לדרוש בחירה חוזרת.
  8. **האפליקציה מאפשרת** מענה על שאלון טריוויה בן 10 שאלות המתבסס על התוכן שהוצג בסצנת המשימה.
  9. **האפליקציה מחשבת** ניקוד לפי מספר תשובות נכונות, כאשר כל תשובה מזכה ב־10 נקודות.
  10. **האפליקציה מתרגמת** הממשק, התוכן והשמע בהתאם לשפה שנבחרה בתחילת השימוש.
* דרישות לא פונקציונליות (10 דרישות, יש לסווג דרישות לא פונקציונליות לפי wikipedia NFR).
  1. Usability (שמישות):-
     + הממשק חייב להיות אינטואיטיבי ופשוט לשימוש עבור ילדים ומבוגרים.
  2. Performance (ביצועים):-
     + זמן טעינה של כל סצנה לא יעלה על 3 שניות.
     + עיבוד תמונות בזיהוי AR יתבצע בפחות מ-1 שנייה מזיהוי.
  3. Reliability (אמינות):-
     + המערכת תפעל ללא קריסות בכל שלבי השימוש.
  4. Portability (ניידות):-
     + האפליקציה תפעל על מכשירי Android התומכים ב- AR.
     + ניתן יהיה להתקין את האפליקציה כ־APK .
  5. Maintainability (תחזוקה):-
     + קבצי התוכן (שמע, טקסטים, תמונות) מופרדים מהקוד וניתנים לעדכון בקלות.
     + קוד מתועד בתגובות מסודרות.
  6. Security (אבטחה):-
     + המערכת לא שומרת מידע אישי של המשתמש.
  7. Scalability (יכולת גידול):-
     + המערכת יכולה להתרחב ולכלול סצנות נוספות בעתיד.
     + ניתן להוסיף שפות נוספות דרך קובצי תרגום.
  8. Localization (התאמה לשפה):-
     + כל הטקסטים והאודיו יתורגמו באופן מלא לשלוש השפות.
     + תשתמש בטבלאות לוקליזציה של Unity .
  9. Accessibility (נגישות):-
     + ניתן לקרוא את הטקסטים המוצגים עם שמע.
     + הממשק כולל כפתורים ברורים ובולטים.
  10. Testability (בדיקות):-
      + כל סצנה תיבדק בנפרד עם בדיקות ליחידת קלט/פלט.
      + יוגדרו תרחישי בדיקה לכל כפתור וסצנה.
* דרישות ממשק חיצוניות.
  1. **מצלמה** – האפליקציה נדרשת לגישה למצלמה לצורך זיהוי תמונות (AR).
  2. **רמקולים** – נדרשת תמיכה בהשמעת שמע.
  3. **מערכת הפעלה של המכשיר (Android)** – כי האפליקציה תלויה ביכולות חומרה ותוכנה של המכשיר.
  4. **- TextMeshPro** להצגת טקסט איכותי בממשק.
  5. **שימוש במנוע AR (AR Foundation / Vuforia)** - כדי לתמוך בטכנולוגיית מציאות רבודה.
* הציגו ארכיטקטורה מעודכנת של האתר (תרשים הכולל את האלמנטים המרכזיים).



שכבת תשתית:-

* Unity Engine :- מנוע המשחק המרכזי – אחראי על הרצת הסצנות והתוכן.
* AR Foundation :- מאפשר זיהוי משטחים ואלמנטים במציאות רבודה.
* Vuforia Engine :- מזהה תמונות בעולם האמיתי ומפעיל תכנים בהתאם.
* Multimedia Manager :- מנהל את השמע, כתוביות ווידאו בהתאם לשפה וסצנה.
* Localization System :- תומך בהצגת תוכן בשפות: עברית, ערבית ואנגלית.
* Scene Manager:- אחראי על מעבר בין הסצנות השונות של האפליקציה.

שכבת ממשק / סצנות:-

* Welcome Scene:- בחירת שפה (עברית, ערבית, אנגלית) והתחלת השימוש.
* Menu Scene:- תפריט ראשי עם שתי אפשרויות: מעבר ל־ Mission Scene , צפייה ב- Video Scene.
* Mission Scene:- ניגון קריינות עם כתוביות תואמות , אפשרות לזיהוי תמונות (AR) עם הסבר קולי נוסף, כפתור מעבר ל Quiz Scene.
* Quiz Scene:- חידון בן 10 שאלות , משוב מיידי לכל שאלה, ניקוד סופי.
* Video Scene:- סרטון היסטורי על יגאל אלון , כפתור לחזרה לתפריט הראשי.

3. הגישו את האפליקציה הכוללת:

1. מסך פתיחה

מסך פתיחה - Welcome Scene

* + מה יש במסך: תמונה של יגאל אלון + הסבר קצר על מטרת האפליקציה.
  + **בחירה:** המשתמש בוחר את שפת הממשק מתוך שלוש שפות: עברית, ערבית, אנגלית.
  + **פעולה:** לאחר הבחירה, המשתמש לוחץ על " Continue" ומועבר לסצנת התפריט.

1. סצנה המתארת את המשימה שלכם (לפי הפרויקט שבחרתם)

סצנה המתארת את המשימה - Mission Scene

* + - **מטרת הסצנה:** להציג מידע היסטורי על יגאל אלון בצורה אינטראקטיבית ומותאמת שפה.
    - מה קורה בה:
      1. השמעת קריינות בשפה הנבחרת עם כתוביות מסונכרנות.
      2. זיהוי תמונות באמצעות מצלמה (Vuforia) - כל תמונה מפעילה אודיו וטקסט רלוונטי.
      3. כפתור מעבר לחידון (Quiz Scene).

1. סצנת סיום (כולל הודעה למשתמש על סיום השימוש).
   * בסיום החידון מוצג ניקוד סופי, עם אפשרות לחזור לתפריט הראשי ((Menu Scene.
   * וגם הודעה למשתמש שסיים את השימוש.
2. אלמנט משחוק כלשהו – פרטו והסבירו .

אלמנט משחוק - Quiz Scene

* + - מדובר בחידון אינטראקטיבי עם 10 שאלות על תוכן הקריינות שהושמעה.
    - לכל שאלה 3 תשובות אפשריות.
    - על כל תשובה נכונה – המשתמש מקבל 10 נקודות.
    - משוב מיידי:
      1. תשובה נכונה נצבעת בירוק.
      2. תשובה שגויה נצבעת באדום והתשובה הנכונה מוצגת בירוק.
    - **תוצאה סופית:** בסיום מוצג ניקוד כולל (עד 100 נקודות).

4.כתבו מדריך למתכנת ומדריך למשתמש עבור האפליקציה.

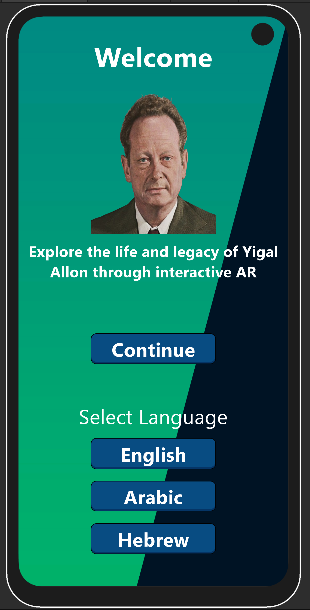
* תיק המשתמש כולל הסבר על המסכים השונים של התוכנה שלכם, ומיועד למשתתפי האפליקציה.
  + Welcome Scene – מסך פתיחה

**הסבר על המסך:**  
זהו המסך הראשון שמופיע בפתיחת האפליקציה. המסך מציג תמונה של יגאל אלון, וכפתורי בחירה שמאפשרים למשתמש להתאים את חוויית השימוש האישית שלו לפי שפתו. העיצוב נקי וברור, עם כפתורים מודגשים לשימוש נוח ומהיר.

**מטרת המסך:**  
לאפשר למשתמש לבחור את שפת השימוש (עברית, ערבית או אנגלית) ולהתחיל את חוויית המציאות הרבודה של האפליקציה בהתאם לשפה שנבחרה. הבחירה בשפה משפיעה על כל התוכן שיוצג בהמשך – טקסטים, כתוביות ושמע.

מה המשתמש עושה במסך הזה:

* בוחר את שפת השימוש באפליקציה על ידי לחיצה על אחד מהכפתורים English, Hebrew, Arabic.
* לוחץ על כפתור  **Continue** כדי להמשיך אל התפריט הראשי (Menu Scene).



* + Menu Scene - מסך תפריט

הסבר על המסך:

זהו המסך המרכזי של האפליקציה, שממנו המשתמש יכול לגשת לשתי פעילויות שונות. כל כפתור מוביל לסצנה שונה עם תוכן אינטראקטיבי, חינוכי וחווייתי.

מטרת המסך:

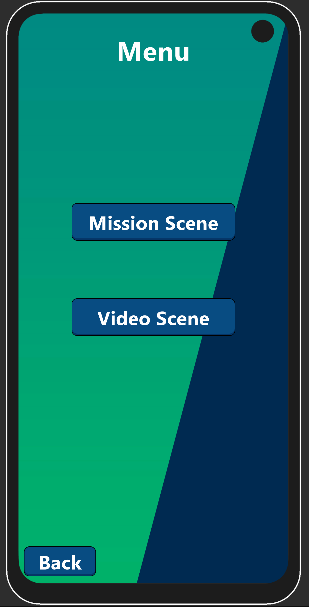
לאפשר למשתמש לבחור את הסצנה שבה הוא רוצה לצפות או להשתתף ,לאחר שכבר בחר שפה במסך הפתיחה. מסך זה משמש כמרכז הניווט של האפליקציה.

מה המשתמש עושה במסך הזה:

לוחץ על אחד מהכפתורים כדי לעבור לסצנה הרצויה:

* Mission Scene - סצנה עם שמע, כתוביות ותמונות AR.
* Video Scene - צפייה בסרטון על חייו של יגאל אלון.

לוחץ על כפתור Back (חזרה) כדי לחזור למסך הפתיחה (Welcome Scene).



* + Mission Scene - מסך המשימה המרכזית

הסבר על המסך:

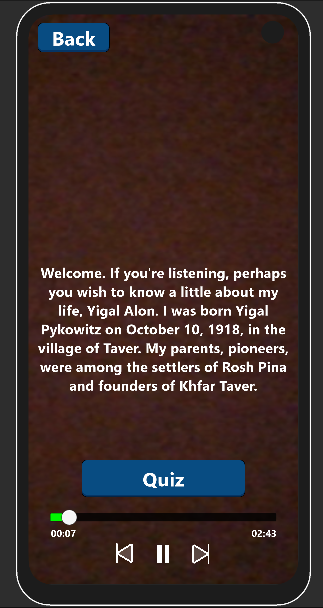
זהו המסך המרכזי של האפליקציה, שבו מוצגת המשימה העיקרית – הקשבה להסבר קולי על חייו ופועלו של יגאל אלון. במסך זה המשתמש נחשף לחוויה מולטימדיה הכוללת אודיו תואם לשפה שנבחרה, כתוביות, ושליטה מלאה על ההשמעה.  
בנוסף, כאשר המצלמה מזהה אחת מ־5 התמונות שתלויות על הקיר, האפליקציה משמיעה הסבר נוסף ומשלימה את החוויה עם טקסט מותאם לתמונה ולשפה שנבחרה.

מטרת המסך:

להעביר למשתמש בצורה חווייתית, אישית ומרתקת מידע היסטורי על יגאל אלון – בשפתו, בליווי תמונות, טקסטים וסאונד תואם.

מה המשתמש עושה במסך הזה:

* לוחץ על **כפתור ההפעלה** כדי להאזין להסבר הקולי.
* יכול לשלוט בהשמעה:
  + Pause – להשהות.
  + Forward – להתקדם 10 שניות.
  + Back - לחזור 10 שניות אחורה.
* רואה **כתוביות תואמות** לשמע (בהתאם לשפה שנבחרה).
* מפנה את המצלמה לעבר אחת מחמש התמונות הפיזיות , וברגע שהתמונה מזוהה, האפליקציה משמיעה הסבר נוסף ומציגה טקסט מתאים.
* לוחץ על כפתור Back בפינה השמאלית העליונה כדי לחזור לתפריט הראשי.



* + Video Scene - מסך הווידאו

הסבר על המסך:

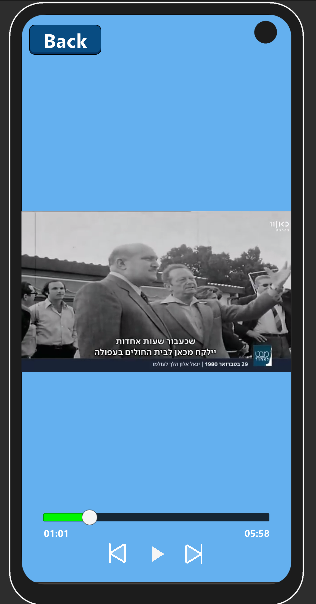
במסך זה מוצג סרטון המתעד רגעים חשובים בחייו של יגאל אלון. הסרטון מלווה בכתוביות, וכולל ממשק שליטה פשוט להפעלה, עצירה, חזרה והתקדמות.

מטרת המסך:

להעשיר את חוויית המשתמש עם מידע אותנטי על יגאל אלון דרך וידאו היסטורי – וכך לחזק את ההקשר בין התוכן הלימודי לאדם האמיתי והאירועים שעמד מאחוריהם.

מה המשתמש עושה במסך הזה:

* צופה בסרטון וידאו באורך כ ־6 דקות.
* שולט בהקרנה באמצעות כפתורים:
  + Play / Pause - התחלה/השהיה.
  + Forward - מעבר קדימה.
  + Back - חזרה אחורה.
* רואה **כתוביות** המופיעות כחלק מהווידאו עצמו.
* יכול ללחוץ על **כפתור** Back בפינה העליונה כדי לחזור לתפריט הראשי.



* + Quiz Scene - מסך החידון

הסבר על המסך:

במסך זה מופיע חידון אינטראקטיבי בן 10 שאלות, אשר נועד לבדוק את הבנת המשתמש בתוכן שהושמע בסצנת המשימה. כל שאלה מוצגת עם שלוש אפשרויות לתשובה. העיצוב נעים ונגיש – טקסט השאלה מופיע במרכז, שלוש תשובות מוצגות מתחת, וכפתור חזרה נמצא בתחתית.

האפליקציה מספקת משוב מיידי על כל תשובה: אם נבחרה תשובה נכונה – היא נצבעת בירוק, ואם נבחרה תשובה שגויה – היא נצבעת באדום והתשובה הנכונה תסומן בירוק. בסיום החידון, מוצג ניקוד סופי.

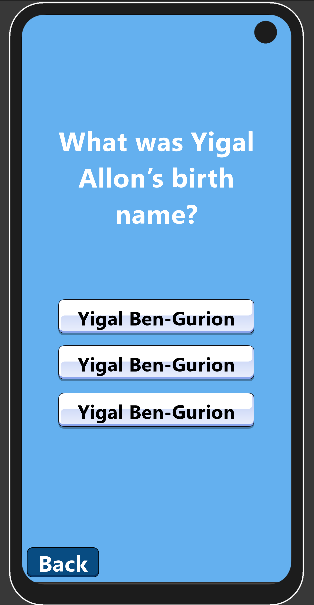
מטרת המסך:

לבחון את המשתמש על המידע שנלמד באפליקציה בדרך חווייתית, לחזק את הזיכרון וההבנה, ולעודד מעורבות פעילה בתוכן ההיסטורי.

הדרך לגעת למסך זה היא על ידי לחיצת כפתור Quiz במסך mission.

מה המשתמש עושה במסך הזה:

* קורא את השאלה המוצגת.
* לוחץ על אחת משלוש התשובות האפשריות.
* מקבל **משוב מיידי:**
  + תשובה נכונה נצבעת בירוק.
  + תשובה שגויה נצבעת באדום והתשובה הנכונה מוצגת בירוק.
* ממשיך אוטומטית לשאלה הבאה (יש 10 שאלות).
* מקבל ניקוד: כל תשובה נכונה מזכה ב־10 נקודות (סה"כ עד 100).
* לוחץ על Back כדי לחזור לתפריט הראשי.



* תיק המתכנת מיועד למתכנתים שיעשו שימוש בעתיד בקוד, וכולל תיעוד מפורט של כל שמות הקבצים באפליקציה, איזה פונקציות בכל קובץ,  ולכל פונקציה - שם הפונקציה, קלט, פלט ותיאור של שורה או שתיים מה היא מבצעת.
* קובץ: WelcomeManager.cs

קובץ זה אחראי לניהול מסך הפתיחה של האפליקציה (Welcome Scene) , כולל בחירת שפה, הצגת טקסטים מותאמים, וטעינת הסצנה הבאה.

פונקציות בקובץ:

* **Start()**

**קלט:** אין

**פלט:** אין

**תיאור:**  
מופעל עם פתיחת הסצנה. מאתחל את השפה כברירת מחדל (אנגלית), מסתיר את כל הטקסטים האחרים, מראה את האנגלית, קובע פונטים ומחבר את הכפתורים.

* SetFonts(Language currentLanguage)

קלט: Language (אנגלית / עברית / ערבית)

**פלט:** אין

**תיאור:**  
עובר על רשימת טקסטים (TextMeshProUGUI) ומשייך להם את הפונט המתאים לפי השפה שנבחרה.

* ProcessButtons()

**קלט:** אין

**פלט:** אין

**תיאור:**  
מגדיר את פעולות שלושת כפתורי השפה (עברית, ערבית, אנגלית) ואת כפתור ההמשך.  
עבור כל כפתור:

קובע את השפה באמצעות LeanLocalization

שומר את השפה ב CommonInfo

מציג את הטקסט בשפה הנבחרת

מעדכן את הפונטים:- עבור כפתור ה (menuButton) Continue , טוען את הסצנה הבאה (Menu).

* קובץ: SceneLoader.cs

קובץ פשוט ושימושי, שמיועד לטעינת סצנות שונות באפליקציה בליווי תצוגת טוען (loading panel). משמש כ utility שמאפשר לכל כפתור לטעון סצנה.

פונקציות בקובץ:

* LoadScene(string sceneName)

קלט: sceneName - מחרוזת עם שם הסצנה שיש לטעון.

פלט: אין.

תיאור:

מציג את חלונית הטעינה (loadingPanel) מפעיל טעינה אסינכרונית של הסצנה לפי השם שהועבר. מאפשר מעבר בין סצנות באופן חלק.

* קובץ: MenuManager.cs

קובץ זה מנהל את סצנת התפריט הראשי (Menu Scene), כולל כפתור חזרה, התאמת גופנים לפי השפה שנבחרה, ואפשרות לטעינת סצנות נוספות.

פונקציות בקובץ:

* Start()

**קלט:** אין

**פלט:** אין

**תיאור:**  
מחבר את כפתור החזרה (backButton) לפונקציית טעינה של הסצנה הקודמת (ברירת מחדל: " Menu").

בנוסף, קורא לפונקציה SetFonts כדי להתאים את הגופנים לשפה שנבחרה.

* SetFonts(Language currentLanguage)

קלט: Language - שפה נוכחית (עברית / ערבית / אנגלית).

**פלט:** אין

תיאור:

עובר על כל הטקסטים שמוגדרים ברשימה (texts) ומשייך להם את הפונט המתאים לשפה. משמש להצגת תפריט נגיש ונכון גרפית בהתאם לשפה.

* LoadScene(string sceneName)

קלט: sceneName - מחרוזת עם שם הסצנה לטעינה.

**פלט:** אין

תיאור:

מציג את loadingPanel ומפעיל טעינה אסינכרונית של הסצנה שנבחרה. מאפשר מעבר ממסך התפריט לסצנות כמו Video , Mission.

* קובץ: FontSetter.cs

קובץ זה אחראי על התאמת הגופנים של רכיבי טקסט בממשק בהתאם לשפה שנבחרה. הוא מטפל גם בטקסטים רגילים (TextMeshProUGUI) וגם בטקסטים מסוג (RTLTextMeshPro) RTL שתומכים בשפות מימין לשמאל.

פונקציות בקובץ:

* **Start()**

**קלט:** אין

**פלט:** אין

תיאור:

מופעל עם טעינת הסצנה. קורא לשפה שנשמרה **(**באמצעות CommonInfo.GetLang()) ומפעיל את הפונקציה SetFonts() כדי להגדיר את הגופנים המתאימים.

* SetFonts(Language currentLanguage)

קלט: Language - שפה נוכחית (אנגלית / עברית / ערבית).

**פלט:** אין

תיאור:

עובר על רשימת הטקסטים texts (TMP רגיל) ורשימת (RTLTextMeshPro) ומגדיר את הפונט המתאים עבור כל אחד מהם, לפי השפה הנבחרת. שימושי במיוחד להצגה תקינה של שפות כמו עברית וערבית.

* קובץ: CommonInfo.cs

מחלקה סטטית שמאפשרת אחסון ושליפה של השפה שנבחרה על ידי המשתמש, באמצעות מערכת PlayerPrefs של Unity.

הקובץ מהווה רכיב מרכזי בהפיכת האפליקציה לרב-לשונית.

פונקציות בקובץ:

* GetLang()

**קלט:** אין

פלט: Language (אנגלית / עברית / ערבית)

תיאור:

מחזיר את השפה האחרונה שנשמרה ב PlayerPrefs תחת המפתח " LAN". אם לא נשמר ערך קודם – מחזיר אנגלית כברירת מחדל.

משתמש בקוד מספרי (0-2) כדי לקבוע את השפה.

* SetLanguage(int i)

קלט: int - מספר בין 0 ל־2:

0 = אנגלית

1 = עברית

2 = ערבית

**פלט:** אין

תיאור:

שומר את השפה שנבחרה (i) במערכת PlayerPrefs תחת המפתח " LAN". השפה תישמר גם בהרצות עתידיות של האפליקציה.

* קובץ: VideoController.cs

קובץ זה שולט על ניגון סרטון הווידאו באפליקציה (Video Scene). כולל ממשק משתמש לניגון, עצירה, דילוג קדימה/אחורה, תצוגת זמן, וטעינת סצנות.

פונקציות בקובץ:

* Awake()

**קלט:** אין

**פלט:** אין

תיאור:

מאתחל את רכיב ה VideoPlayer , מקשר את כפתורי הניגון לפונקציות המתאימות (seek , pause , play).

* Start()

**קלט:** אין

פלט: IEnumerator

**תיאור:**

ממתין להכנת הווידאו (Prepare) , ואז מציג את משך הווידאו, מפעיל אותו, ומתחיל לעדכן את ה UI באופן שוטף דרך UpdateUI().

* TogglePlayPause()

**קלט:** אין

**פלט:** אין

תיאור:

עוצר או ממשיך את ניגון הווידאו, ומעדכן את האייקון בכפתור בהתאם.

* **SeekBy(float delta)**

**קלט:** מספר שניות לדילוג (חיובי או שלילי).

**פלט:** אין

תיאור:

קופץ קדימה או אחורה בזמן הווידאו לפי הערך delta.

* OnSliderDrag(float value)

**קלט:** ערך חדש מהסליידר.

**פלט:** אין.

תיאור:

משנה את זמן הניגון של הווידאו בהתאם לערך שנבחר בסליידר.

* SeekTo(float time)

**קלט:** זמן יעד (בשניות).

**פלט:** אין.

תיאור:

מבצע את הקפיצה בפועל בזמן הווידאו. מעדכן גם את ה UI של הסליידר והזמן.

* UpdateUI()

**קלט:** אין

פלט: IEnumerator

תיאור:

עדכון שוטף של זמן הניגון על המסך והסליידר בהתאם להתקדמות הווידאו.

* FormatTime(float t)

**קלט:** זמן בשניות.

**פלט:** מחרוזת (בפורמט mm:ss)

תיאור:

ממיר את מספר השניות לפורמט זמן קריא להצגה בצד המשתמש.

* LoadScene(string sceneName)

**קלט:** שם הסצנה לטעינה.

**פלט:** אין.

תיאור:

מציג את מסך הטעינה (loadingPanel) וטוען את הסצנה הנבחרת באופן אסינכרוני.

* קובץ: UnitySubtitlePlayer.cs

קובץ זה אחראי על ניגון הקריינות הראשית בסצנת המשימה (Mission Scene), הצגת כתוביות תואמות לפי שפה, שליטה בניגון, וניהול השמעת שמע של תמונות מזוהות בזיהוי AR.

פונקציות בקובץ:

* Awake()

**קלט:** אין

**פלט:** אין

תיאור:

מאתחל את רכיבי השמע, הטקסטים, והכפתורים. קובע את הפונטים לפי השפה שנבחרה, ומכין את הממשק לפעולה.

* **Start()**

**קלט:** אין

פלט: IEnumerator

תיאור:

טוען את קובץ הקריינות (AudioClip) ואת קובץ הכתוביות (TextAsset) לפי השפה שנבחרה, ומתחיל ניגון וקריאת כתוביות בזמן אמת.

* SetFonts()

**קלט:** אין

**פלט:** אין

תיאור:

בוחר את הפונט המתאים עבור הטקסטים (כתוביות וטקסט כפתור חזרה) לפי השפה הנוכחית.

* TogglePlayPause()

**קלט:** אין

**פלט:** אין

תיאור:

עוצר או מפעיל את הקריינות הראשית, ומחליף את האייקון של כפתור ההפעלה בהתאם.

* SeekBy(float delta)

קלט: float - מספר שניות לדילוג קדימה/אחורה.

**פלט:** אין

תיאור:

קופץ קדימה או אחורה בזמן ההשמעה הראשית.

* OnSliderSeek(float value)

**קלט:** ערך חדש מהסליידר

**פלט:** אין

תיאור:

משנה את מיקום ההשמעה לפי הערך שבחר המשתמש.

* SeekTo(float time)

**קלט:** זמן יעד (בשניות)

**פלט:** אין

תיאור:

משנה את מיקום ההשמעה בפועל. מאתחל את הצגת הכתוביות בהתאם לנקודת הזמן החדשה.

* RunSubtitles()

**קלט:** אין

פלט: IEnumerator.

תיאור:

מציג את הכתוביות ברגעים המתאימים לפי זמנים שנלקחו מקובץ SRT. תומך באפקט הקלדה אם מופעל.

* TypeText(string text, float cueEnd, float delay)

**קלט:** טקסט להצגה, זמן סיום, זמן עיכוב בין תווים.

פלט: IEnumerator

תיאור:

מציג כתובית עם אפקט הקלדה הדרגתי, עד שההקלט מסתיים או הזמן עובר.

* **ClearAt(float cueEnd)**

**קלט:** זמן סיום הכתובית.

פלט: IEnumerator

תיאור:

ממתין עד שהכתובית מסתיימת ומנקה את השורה מהמסך.

* **UpdateProgressUI()**

**קלט:** אין

פלט: IEnumerator

תיאור:

מעדכן את תצוגת הזמן הנוכחי וסרגל ההתקדמות באופן שוטף תוך כדי השמעה.

* FormatTime(float t)

**קלט:** זמן בשניות

**פלט:** מחרוזת (פורמט mm:ss).

תיאור:

ממיר שניות לפורמט זמן ידידותי להצגה על גבי המסך.

* **LoadScene(string sceneName)**

**קלט:** שם הסצנה לטעינה

**פלט:** אין

תיאור:

מציג את מסך הטעינה (loadingPanel) וטוען את הסצנה החדשה בצורה אסינכרונית.

* PlayPhotoAudio(AudioClip clipForPhoto)

קלט: AudioClip - קובץ שמע של תמונה שזוהתה.

**פלט:** אין

תיאור:

עוצר את הקריינות הראשית, מנקה כתוביות, ומשמיע את השמע של התמונה שזוהתה.

* ResumeMainNarrationAfterPhotoAudio()

**קלט:** אין

פלט: IEnumerator

תיאור:

ממתין לסיום השמע של התמונה, ואז מחזיר את הקריינות הראשית והכתוביות להמשך מנקודת העצירה.

* קובץ: QuizManager.cs

קובץ זה מנהל את מסך החידון באפליקציה. הוא כולל טעינת שאלות לפי שפה, הצגת השאלות והתשובות, טיפול בבחירת המשתמש, חישוב ניקוד, ומתן משוב בזמן אמת.

פונקציות בקובץ:

* **Awake()**

**קלט:** אין

**פלט:** אין

תיאור:

מאתחל את רשימות השאלות לפי שפה (עברית, אנגלית, ערבית), קורא את השפה הנוכחית מCommonInfo , ומעדכן פונטים בהתאמה.

* SetFonts()

**קלט:** אין

**פלט:** אין

תיאור:

מגדיר את הפונטים של רכיבי הטקסט (שאלה, אפשרויות, תוצאה) לפי השפה הנוכחית.

* **Start()**

**קלט:** אין

**פלט:** אין

תיאור:

בוחר את רשימת השאלות לפי השפה, מציג את השאלה הראשונה, ומסתיר את חלון התוצאה.

* ShowQuestion()

**קלט:** אין

**פלט:** אין

תיאור:

מציג את השאלה הנוכחית עם שלוש אפשרויות תשובה. אם נגמרו השאלות – עובר לתצוגת תוצאה.

* GetQuestionPrefix(int questionNumber)

**קלט:** מספר שאלה (אינדקס).

**פלט:** מחרוזת תיאור ("שאלה מס׳..." , " Question No..." , " سؤال رقم...")

תיאור:

מחזיר את הכותרת המתאימה לפי השפה.

* **OnOptionSelected(int selectedIndex)**

**קלט:** אינדקס התשובה שנבחרה

**פלט:** אין

תיאור:

בודק אם התשובה נכונה או שגויה, צובע את הכפתורים בהתאם, ומתחיל את הפידבק באמצעות Coroutine.

* HandleAnswerFeedback(bool isCorrect, int selectedIndex)

**קלט:** האם התשובה נכונה, ואינדקס התשובה.

פלט: IEnumerator

תיאור:

מראה משוב בצבעים, מעדכן ניקוד, ממתין זמן קצר, ואז מציג את השאלה הבאה.

* UpdateLiveScore()

**קלט:** אין

**פלט:** אין

תיאור:

מציג את הניקוד הנוכחי (מתעדכן בלייב) , מתוך 100.

* ShowResult()

**קלט:** אין

**פלט:** אין

תיאור:

מציג את סיכום החידון – כולל הציון הכולל, כוכבים לפי הצלחה, והודעה מתורגמת לפי שפה.

* ReverseText(string input)

**קלט:** מחרוזת

**פלט:** מחרוזת הפוכה

תיאור:

פונקציה פשוטה שמבצעת היפוך טקסט (מיועדת לשפות RTL כמו עברית).

* **LoadScene(string sceneName)**

**קלט:** שם סצנה

**פלט:** אין

תיאור:

טוען סצנה חדשה בליווי חלון טעינה (loadingPanel).

* קובץ: LanguageSetting.cs

קובץ זה אחראי על הצגת טקסט מתאים (בתלת־ממד) לפי השפה שנבחרה. משמש בעיקר כאשר מוצגות תמונות או אובייקטים AR וכל אחת מהן דורשת טקסט הסבר בשפה מתאימה.

פונקציות בקובץ:

* ShowLangText()

**קלט:** אין

**פלט:** אין

תיאור:

תחילה מסתיר את כל רכיבי הטקסט (RTLTextMeshPro3D) הקשורים לתמונה או לאובייקט, ואז מציג רק את הטקסט המתאים לשפה שנבחרה.

לפי הסדר:

languageTexts[0] = אנגלית.

languageTexts[1] = עברית.

languageTexts[2] = ערבית.

נדרש לוודא שהרשימה languageTexts מסודרת בדיוק לפי הסדר הזה.

* קובץ: PhotoTargetHandler.cs

קובץ זה אחראי על התנהגות תמונה מזוהה בטכנולוגיית Vuforia.

כאשר אחת מהתמונות (Image Target) מזוהה על ידי המצלמה, הוא מציג טקסט לפי השפה ומשמיע את קובץ השמע המתאים.

פונקציות בקובץ:

* Start()

**קלט:** אין

**פלט:** אין

תיאור:

מאתר את רכיב ObserverBehaviour שמחובר לאובייקט, ונרשם לאירוע של שינוי סטטוס (האם התמונה זוהתה). בכך מאפשר תגובה בזמן אמת לגילוי התמונה.

* OnTargetStatusChanged(ObserverBehaviour behaviour, TargetStatus status)

קלט:

Behavior : האובייקט שזוהה

Status : מצב הזיהוי (נמצא / נעלם וכו').

**פלט:** אין

תיאור:

כשמזהים את התמונה, מתבצעים שני דברים:

* + - 1. מציגים את הטקסט התואם לשפה (באמצעות LanguageSetting.ShowLangText()).
      2. מנגנים את קובץ השמע המתאים לשפה דרך UnitySubtitlePlayer.PlayPhotoAudio(...).
* GetAudioForLang(Language lang)

**קלט:** Language - השפה הנוכחית

פלט: AudioClip - קובץ השמע המתאים

תיאור:

מחזיר את קובץ השמע המתאים לשפה: עברית, ערבית, או אנגלית (ברירת מחדל).  
שלושת הקבצים מוקצים מראש ב Inspector.

* ציינו את האלמנטים המיוחדים לדעתכם שעשיתם בהם שימוש בקוד: קטעי קוד מעניינים שכתבתם, קטעי קוד חיצוניים ותוכנות עזר שהשתמשתם בהם (כולל קישורים ורפרנסים , בפרט פרומפטים AI).

במהלך פיתוח האפליקציה נעשה שימוש בכמה כלים ותוספים חיצוניים שתרמו לתמיכה בריבוי שפות, נוחות ממשק, וטיפול בטקסטים ייחודיים:

1. ArabicSupportTools

* ספרייה שמטרתה להציג טקסטים בערבית בצורה תקינה במשחקי Unity.
* שפות כמו ערבית נכתבות מימין לשמאל, ואינן נתמכות באופן טבעי על־ידי Unity.
* הכלי מאפשר להפוך טקסטים בערבית כך ש:
  + האותיות יהיו מחוברות כראוי (ולא מנותקות).
  + הכיוון יהיה RTL מלא.

- <https://assetstore.unity.com/packages/tools/gui/arabic-support-for-unity-2674?srsltid=AfmBOopqzouxF7xus5JngD4XW4hx_ge4hb_tunti3EFbN8UdOBeVUPdY>

2. Lean Localization (Lean Language & Lean Phase)

* תוסף קל לשימוש לתרגום ממשק משתמש לפי שפה.
* אפשר לקבוע טקסטים, כפתורים, ואפילו אנימציות לפי שפה נבחרת.
* Lean Language - מאפשר להחליף שפה בריצה..
* Lean Phase - מאפשר להציג טקסט מדורג באפקט נחמד ונגיש.

3. TextMeshPro + RTL TextMeshPro

* נעשה שימוש ב TextMeshPro להצגת טקסט איכותי (גם בעברית וערבית).
* נעשה שימוש ב RTLTextMeshPro כדי לתמוך ב RTL בצורה מותאמת לשפות שמאלה.

פרומפטים AI :- השתמשנו ב- ChatGPT

* "אני צריך ספרייה שתתמוך בערבית ותציג את האותיות מחוברות"
* "אני רוצה פונקציה שתשלוט בווידאו"
* "אני צריך פונקציה שתשלוט באודיו ובכתוביות יחד"
* "כיצד לחזור אוטומטית לאודיו הראשי לאחר שהסתיים אודיו של תמונה?"

קטעי קוד מעוניינים שתכתבנו:-

1. אפקט Typewriter לפי זמן כתובית מחושב ב- UnitySubtitlePlayer.cs

float dur = cap.endTime - cap.startTime;

float charDelay = dur / Math.Max(1, display.Length);

typingRoutine = StartCoroutine(TypeText(display, cap.endTime, charDelay));

זה מאפשר הצגת כתוביות באפקט של הקלדה הדרגתית, כשקצב ההקלדה מותאם אוטומטית לאורך הכתובית ולזמן שמוקצב לה.

2. שילוב בין שמע ראשי לשמעי תמונות (AR)

public void PlayPhotoAudio(AudioClip clipForPhoto)

{

audioSource.Pause();

if (subtitleRoutine != null) StopCoroutine(subtitleRoutine);

if (typingRoutine != null) StopCoroutine(typingRoutine);

subtitleText.text = "";

photoAudioSource.clip = clipForPhoto;

photoAudioSource.Play();

StartCoroutine(ResumeMainNarrationAfterPhotoAudio());

}

מעבר חכם בין אודיו ראשי לאודיו תמונה שזוהתה – כולל עצירה, ניקוי כתוביות וחזרה אוטומטית.

3. תמיכה בשלוש שפות כולל RTL

switch (language)

{

case Language.English:

languageTexts[0].gameObject.SetActive(true);

break;

case Language.Hebrew:

languageTexts[1].gameObject.SetActive(true);

break;

case Language.Arabic:

languageTexts[2].gameObject.SetActive(true);

break;

}

גישה פשוטה אך יעילה לניהול טקסטים מרובי שפות, עם התאמה ספציפית לשפות RTL כמו עברית וערבית.

5.**סעיף לביצוע עד 19.6:**  
הציגו את האפליקציה לאיש הקשר שלכם, ובצעו ראיון כולל walkthrough .

בקשו מאיש הקשר למלא שאלון SUS, ולהביע דעתו על האפליקציה כולל הצעות לשיפור.רשמו את ציון ה SUS.

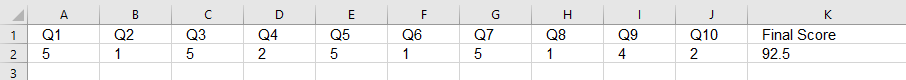
תמללו את הראיון.

מלאו את הטבלה הבאה:

|  |  |  |
| --- | --- | --- |
| הצעה של איש קשר | האם בוצע שינוי? (כן/לא/חלקי) | נימוק |
|  |  |  |
|  |  |  |

אין הצעות של איש קשר כדי לשפר את הפרויקט .

חישוב SUS:-



ציון ה־ SUS של איש הקשר עמד על **92.5 מתוך 100**, ציון גבוה מאוד שמעיד על חוויית שימוש מצוינת. איש הקשר חש שהמערכת קלה מאוד לתפעול, ברורה ונוחה, והממשק עוצב באופן שתומך היטב בהבנת הזרימה ובמעורבות המשתמש. כמו כן, לא זוהו בעיות שמישות או מורכבויות במהלך השימוש, מה שמרמז על התאמה טובה לקהל היעד ועמידה גבוהה בסטנדרטים של חוויית משתמש.

תמליל של הראיון:-

אנחנו: היי נופר, לפני שנתחיל בשאלות, נציג לך בקצרה את האפליקציה שפיתחנו. מדובר באפליקציית מציאות רבודה בנושא יגאל אלון, שמיועדת להצגה בחדר במוזיאון.  
האפליקציה כוללת כמה סצנות – סצנת פתיחה לבחירת שפה, תפריט ניווט, סצנת משימה עם קריינות וכתוביות, זיהוי תמונות AR , סרטון וידאו, וחידון אינטראקטיבי.

(מופעל סרטון הדגמה קצר של האפליקציה)

נופר: האמת, זה מאוד מרשים. הרעיון של השילוב בין מידע היסטורי לבין מציאות רבודה – ייחודי ומעניין.

אנחנו: שמחים לשמוע! חשוב לנו גם שהחוויה תהיה נגישה, לכן כל התכנים באפליקציה תורגמו לשלוש שפות: עברית, ערבית ואנגלית. המשתמש בוחר את שפתו כבר במסך הראשון.

נופר: זה ממש חשוב. ראיתי שרוב הפרויקטים שהוצגו התמקדו בשפה אחת, אז זה לגמרי מייחד אתכם. חשבתם על כל סוגי הקהלים.

אנחנו: תודה רבה! עכשיו נשמח לשמוע מה חשבת על חוויית השימוש בפועל.

נופר: התכנים היו ברורים ונגישים, במיוחד הקריינות עם הכתוביות . ממש אהבתי את הכיוון של הפרויקט. כל הכבוד!

6.הציגו את האתגרים שנתקלתם בהם במהלך העבודה בקורס.

במהלך פיתוח האפליקציה נתקלנו במספר אתגרים טכניים ותכנוניים, שחלקם דרשו פתרונות יצירתיים:

1. תמיכה בשפות RTL (עברית וערבית)

Unity אינו תומך כברירת מחדל בטקסטים בשפות מימין לשמאל.

היינו צריכים לשלב ספרייה חיצונית (ArabicSupportTools) ולבצע התאמות בגופנים וביישור כדי להבטיח הצגה תקינה של טקסטים בערבית ובעברית.

1. ניהול כתוביות מסונכרנות עם אודיו

פיתוח מנגנון שידע לקרוא קבצי .srt, להציג כתוביות בדיוק בזמני הדיבור, ואף לבצע "typewriter effect" - היה מאתגר ודורש סנכרון מדויק עם האודיו.

1. בניית חידון רב־שפתי עם משוב מיידי

יצירת מנגנון חידון תומך שלוש שפות, עם ניקוד, פידבק בצבעים, וסיכום – דרשה תכנון נכון של מבני הנתונים והקוד.

7. הכינו סרטון מלא (לבחירתכם באיזה אורך, מומלץ לכל היותר 2 דקות), המכיל את כל האלמנטים של האפליקציה (ניתן לשלב גם בסרטון חלקים מבניית הקוד -unity, Vuforia , כלים חיצוניים, וכו).

לינק לסרטון:-

<https://drive.google.com/drive/u/1/folders/1mgv_OZNMyy9iltfcDTVJ-yha9qHHWSdp>

הסרטון בשם:- YigalAllonARVideo

8. בתאריך 26/6 תציגו את האפליקציה שלכם בשיעור

כל צוות יציג את עבודתו ויקבל משוב מהסטודנטים האחרים.

לאחר ההצגה תקבלו אנונימית את המשובים מחבריכם.

מלאו את הטבלה הבאה:

|  |  |  |
| --- | --- | --- |
| הצעה לשיפור האפליקציה | האם בוצע שינוי? (כן/לא/חלקי) | נימוק |
| הוספת וידיו | לא | כי יש לנו וידיו |
| לשפר את העיצוב | לא | ההצעה לשיפור העיצוב התקבלה בהתעניינות, אך לאחר בחינה פנימית החלטנו שלא לבצע שינוי בשלב זה, מכיוון שהעיצוב הנוכחי נבחר בקפידה כדי לשמור על פשטות, נגישות. |
| הוספת עוד שאלות | לא | בפרויקט שלנו יש 10 שאלות , וזה מתאים (לא הרבה , ולא קצת). כך ה10 שאלות מכילות כמעט כל הנתונים שמוצגים באפליקציה. |
| לשפר את רצף של המעברים בין החלקים | כן | שלבנו את מסך זיהוי התמונות עם מסך המשימה , והמעבר לחידון מהמסך החדש הזה , אחרי שהמשתמש מבין ורואה כל הנתונים הוא יכול להתחיל את החידון , בלי צורך לחזור אחורה לתפריט. |

יש להכין לקראת ההצגה 2 קבצי APK:  
1. קובץ עם רקע של התמונה אשר המשתמש רואה כשהוא מגיע למיקום הרצת האפליקציה.

2. קובץ זהה ללא רקע, לצורך הרצת האפליקציה במיקום האמיתי.

שתי קבצי ה- apk נמצאים בדרייב :-

<https://drive.google.com/drive/u/1/folders/1mgv_OZNMyy9iltfcDTVJ-yha9qHHWSdp>

קובץ עם רקע של התמונה אשר המשתמש רואה כשהוא מגיע למיקום הרצת האפליקציה בשם :- YigalAllonNoAR.apk

קובץ זהה ללא רקע, לצורך הרצת האפליקציה במיקום האמיתי בשם :- YigalAllonAR.apk

להצגה:

יש להכין מצגת המתארת את המיקום, רקע הסטורי רלוונטי, דרישות , ארכיקטקטורת מערכת ופיצ'רים מעניינים שמימשתם.כמו כן המצגת תכלול הרצת הסרטון/ הדגמה חיה.

חובת הצגה של כלל חברי צוות הפרויקט!

יש להגיש את הפרויקט כולו (בתיקיית גיט עם קישור ל unity cloud, נא לשלוח קישור), וכן הוראות להתקנה והפעלה על מכשיר ייעודי.

לינק ל- Github:- <https://github.com/mhmdkh1905/YigalAllon.git>

באישור ממך העלינו את הפרויקט על google drive , כאן נמצא הפרויקט , מצגת , וידאו , שתי קבצי apk , גם על תרגילי בית.

לינק ל- google drive:-

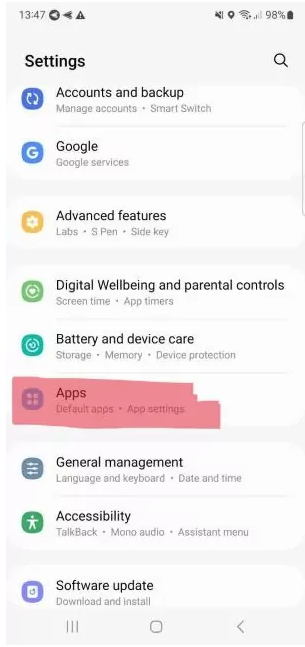
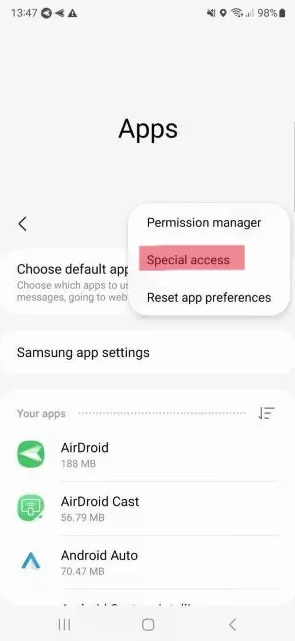
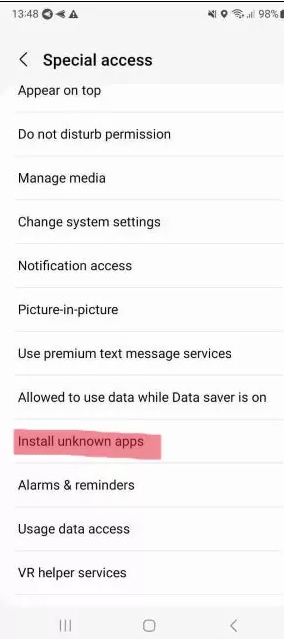
<https://drive.google.com/drive/u/1/folders/1mgv_OZNMyy9iltfcDTVJ-yha9qHHWSdp>

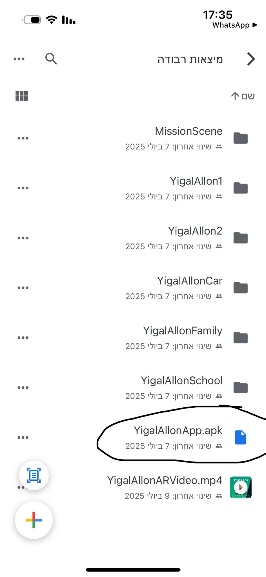
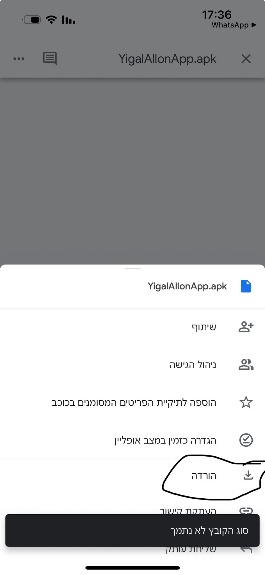
הוראות הפעלה:-

How to download apk on android FIRST TIME

1. **Allow unknown source**
2. You have to enable the installation from unknown sources on your device. To make it possible, perform the following steps:

* 1. Go to Settings > Apps > Special app access > Install unknown apps.



* 1. Locate the APK file on your device. The downloaded APKs can be found in the Downloads app. Or you can use a file manager to navigate to the Downloads folder.
* 2. Tap on the file name to start the installation. If this is the first time you're installing an APK from that file manager, the Downloads app, or the web browser, Android will throw a warning and ask your permission to continue. Give the permission and, depending on the app from where you're installing the APK, you may have to tap back to get to the installer. In some apps, you'll automatically be sent back to the installer.
* 3. Tap on Install, and the application will be installed successfully unless there