

דוח התקדמות פרויקט גמר תשפ"ה

שם הפרויקט : מערכת זיהוי פייק ניוז בטיקטוק

שם הארגון המבקש:

IMT School for Advanced Studies Lucca, Italy

שם מנחה :

ד"ר נדב וולך

מגישים :

יונתן כהן - 207906777

גל סבן - 313133084



תוכן עניינים :

3.....	הקדמה
4.....	פיתוח מענה טכנולוגי לבעיה
5.....	אתגרים בפרויקט ופתרונות
6-7.....	מרכיבי הפרויקט ותכנונם
8-13.....	נספחים ותוצרי הפרויקט עד כה
14.....	רשימת מקורות

הקדמה :

רקע וסקירת ספרות:

בשנים האחרונות, תופעת הפייק ניוז (Fake News) הפכה לאתגר עולמי המשפיע על תחומים כמו פוליטיקה, בריאות הציבור וכלכלה. הפצת מידע כוזב ברשתות החברתיות – Facebook, Twitter, TikTok ו-WhatsApp – מתבצעת במהירות רבה ומערערת את אמון הציבור במוסדות, תוך השפעה על קבלת החלטות. בעידן של תפוצת מידע מיידית, קשה לאתר ולהפריך מקורות שקריים בזמן אמת, ולכן עולה הצורך בכלים טכנולוגיים לזיהוי והפרכת פייק ניוז במהירות ובאמינות.

בין הפתרונות הקיימים בתחום:

- **ClaimBuster:** מערכת אוטומטית לאימות עובדות, המבוססת על תמלול וידאו, עיבוד שפה טבעית (NLP) ולמידת מכונה (ML). היא מזהה טענות עובדתיות, בודקת אותן מול מקורות אמינים ומציגה את תוצאות האימות למשתמש. [\[1\]](#)
- **Azure Video Indexer ו-Azure Text Analytics:** כלים לניתוח וידאו וטקסט, המבצעים תמלול דיבור, זיהוי ישויות (NER), ניתוח רגשות וזיהוי ביטויים מרכזיים באמצעות מודלים מתקדמים כמו BERT ו-Transformers. [\[2\].b](#)
- **Speak AI:** מערכת לניתוח מתקדם של אודיו ווידאו, תמלול אוטומטי, זיהוי מילות מפתח, ניתוח סנטימנט וחיבור לאינטגרציות API לצורך עיבוד מידע נרחב. [\[3\]](#)
- **כלי וידאו עובדות נוספים:**
 - Google Fact Check Tools: איתור טענות שנבדקו והצגת מקורות אמינים.
 - TinEye: מנוע חיפוש הפוך לזיהוי מקורות תמונה ולחשיפת עריכות והקשרים שקריים.
 - Whois, FactCheck.org, Snopes: מסדי נתונים וכלים לאימות מקורות אתרים ומידע ברשת.
- תחום זה ממשיך להתפתח עם שילוב בין אלגוריתמים חכמים, בינה מלאכותית ומאגרי מידע עולמיים, במטרה להיאבק בתפוצת מידע מטעה ולחזק את אמון הציבור במידע המופץ ברשתות החברתיות. [\[4\]](#) [\[5\]](#)

פיתוח מענה טכנולוגי לבעיה:

- בשלב זה של הפרויקט, פיתחנו מענה טכנולוגי ראשוני המיועד לאפשר עיבוד סרטוני טיקטוק והכנת התשתית לניתוח תוכן.
המערכת כוללת מספר מרכיבים עיקריים:
- הורדת סרטוני וידאו: שימוש בספריית yt_dlp להורדה ישירה של סרטונים מהאינטרנט.
- חילוץ אודיו: המרת קובץ הווידאו לפורמט שמע באמצעות ספריית ffmpeg.
- תמלול האודיו לטקסט: ביצוע תמלול קולי מדויק באמצעות מודל Whisper.
- עקיפת מגבלות אבטחה ברשת: שימוש ב-ssl להגדרת תעבורת HTTPS מאובטחת באופן ידני, על מנת לאפשר חיבור לשירותים גם כאשר קיימות מגבלות אבטחה ברשת המקומית.
- הקמת תשתית שרת (Backend): פיתוח אפליקציית Flask בסיסית לניהול קבלת קלט מהמשתמשים ולהפעלת תהליך עיבוד הנתונים.
- המערכת בשלב זה מאפשרת קבלת סרטון מטיקטוק, חילוץ האודיו ותמלול באופן אוטומטי.
בהמשך, נשלב שלבים נוספים הכוללים ניתוח שפה טבעית (NLP), בניית מסווג מבוסס Machine Learning, וניתוח סטטיסטי של התוצאות. [\[6\]](#)

אתגרים ופתרונות בפרויקט:

במהלך שלב הפיתוח הראשוני של המערכת, נתקלנו במספר אתגרים טכנולוגיים ותשתיתיים. להלן האתגרים המרכזיים והפתרונות שננקטו:

1. התאמת סביבת עבודה לפיתוח:

בתחילת הפרויקט נדרשנו לבחור סביבת עבודה שתאפשר עבודה נוחה עם קבצי וידאו, ספריות עיבוד מתקדמות ושפת Python. לאחר בחינת מספר אפשרויות, בחרנו ב-Visual Studio Code, שהציעה גמישות, קלות עבודה ותמיכה נרחבת בספריות הנדרשות לפרויקט.

2. טיפול במגבלות רשת ואבטחה בגישה לתוכן:

במהלך הורדת סרטוני טיקטוק נתקלנו בחסימות אבטחה שהקשו על שליפת התוכן. כדי לעקוף בעיה זו, נעשה שימוש בפתרון SSL מותאם שהגדיר גישה מאובטחת אך גמישה, תוך דילוג על אימותים מחמירים שאינם נדרשים עבור תהליך הפיתוח. כך התאפשר חיבור ישיר והורדת סרטונים גם מרשתות מוגנות.

3. בחירת ספרייה מתאימה לחילוץ אודיו:

ניסינו בתחילה להשתמש בספריית MoviePy לצורך חילוץ אודיו מהווידאו, אך היא לא סיפקה תוצאות אמינות במספר פורמטים. בעקבות כך עברנו לעבוד עם ספריית ffmpeg, שהבטיחה חילוץ מדויק, יציב ויעיל של רצועות שמע מקבצי וידאו.

4. בניית תשתית חיבור בין ממשק לקוח לשרת:

נדרשנו לייצר תהליך עבודה אוטומטי שבו המשתמש מעלה סרטון ומקבל טיפול מלא מצד השרת. לצורך כך פיתחנו שרת Flask בסיסי, שבו הוגדר קובץ HTML לקליטת כתובת URL של הסרטון, שליחה לשרת באמצעות בקשת POST, והמשך עיבוד התוכן ברקע בצורה אוטומטית.

5. שיפור חוויית המשתמש בפתיחת המערכת:

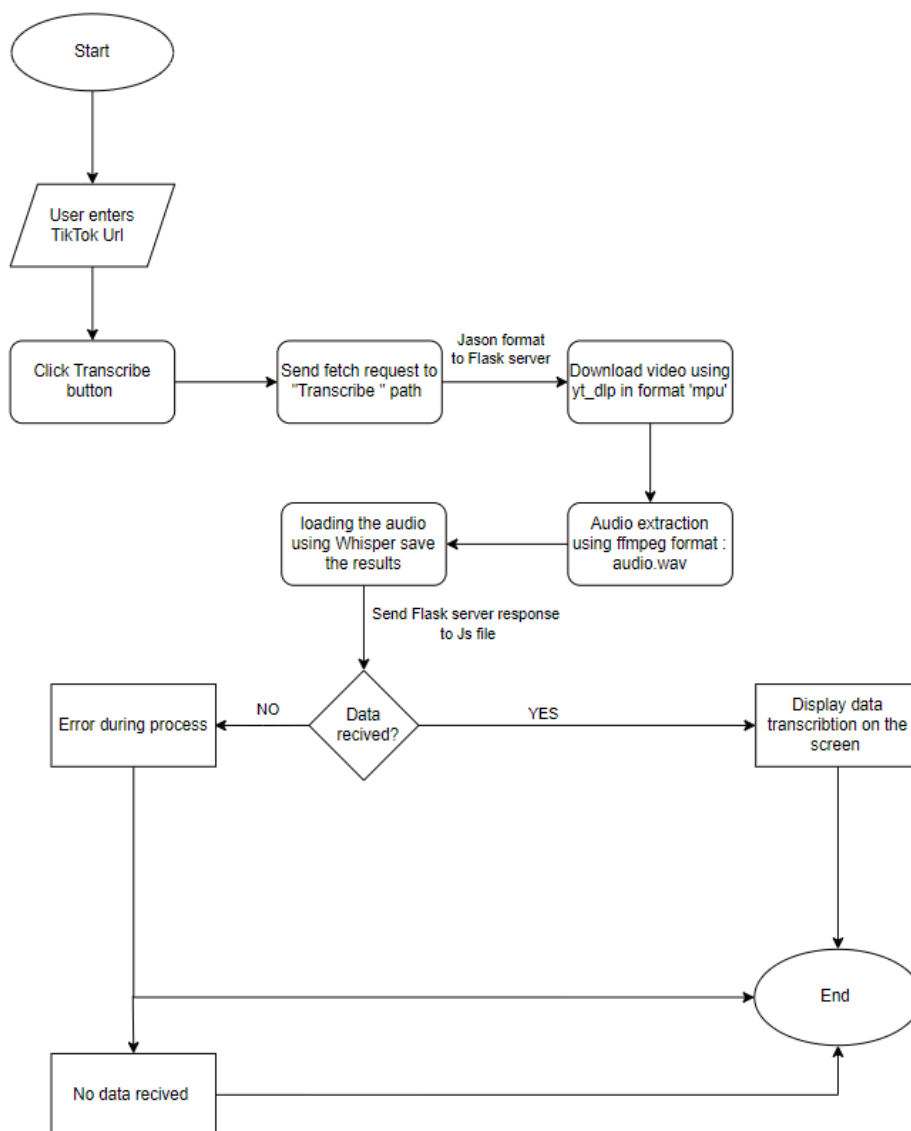
כדי לייעל את תהליך ההפעלה ולחסוך זמן למשתמש, שילבנו שימוש בספריית webbrowser, אשר פותחת אוטומטית את דף האינטרנט עם הרצת השרת, ללא צורך בהקלדת כתובת ידנית.

הפתרונות שנבחרו אפשרו להתגבר על האתגרים בצורה יעילה ולבסס תשתית ראשונית יציבה להמשך בניית מערכת זיהוי פייק ניזז מבוססת וידאו.

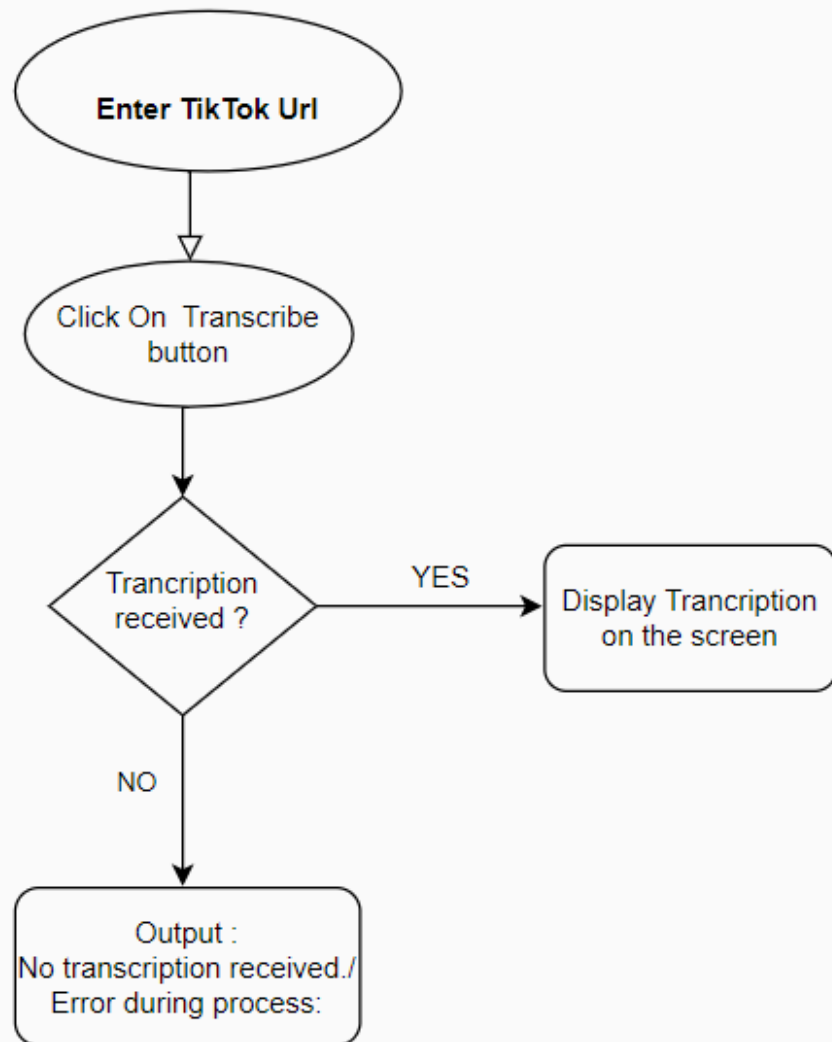
מרכיבי הפרויקט ותכנונם:

במהלך עבודה על הפרויקט יצרנו תרשים זרימה המתאר את הפרטים אשר קורים במערכת מאחורי הקלעים, כמו כן השתמשנו בספריית NLP מתאימות עד כדי קבלת תשובה מהמודל שלנו והצגת הפרטים ללקוח באופן מפורט. כמו כן, נציג את התרשים הקורה מאחורי הקלעים ברגע שהלקוח ילחץ על כפתור תמלל.

Back-end Architecture



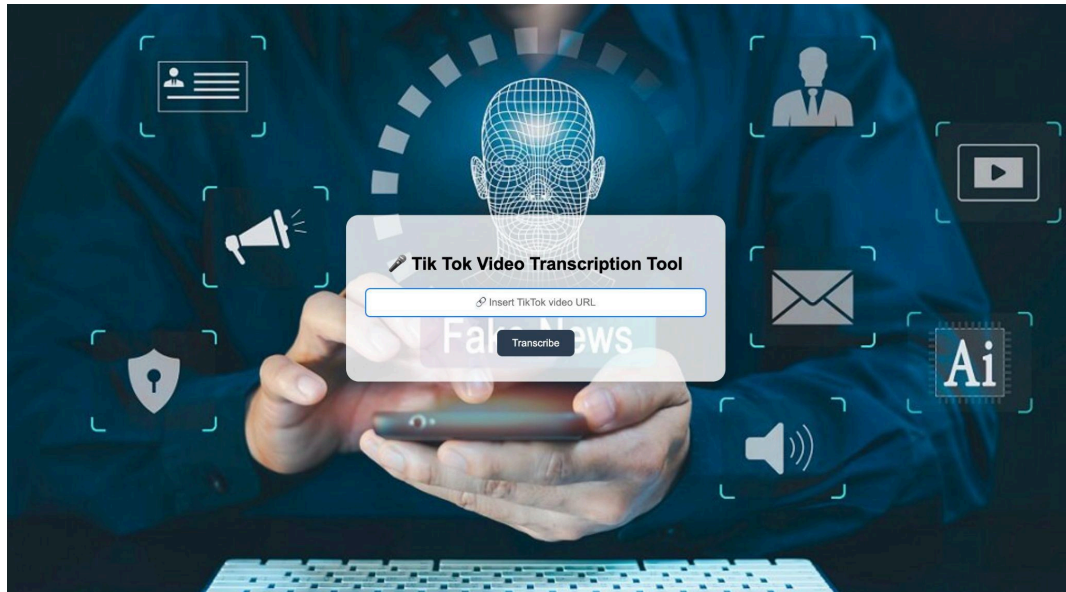
User Architecture



נספחים ותוצרי הפרויקט:

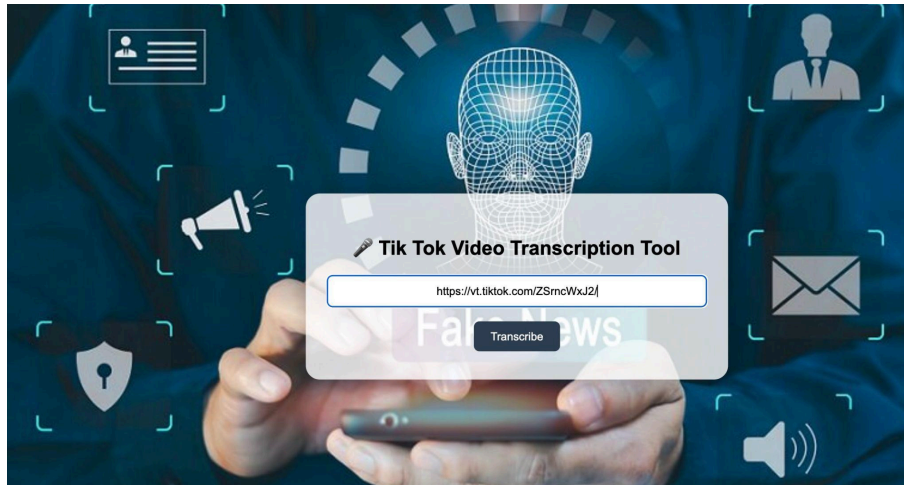
במהלך בניית הממשק של הפרויקט, ניסינו לחשוב על ממשק שיהיה נוח למשתמש וההליך יתבצע בצורה הכי פשוטת שאפשר.

להלן תיאור המסך הראשי של הפרויקט:

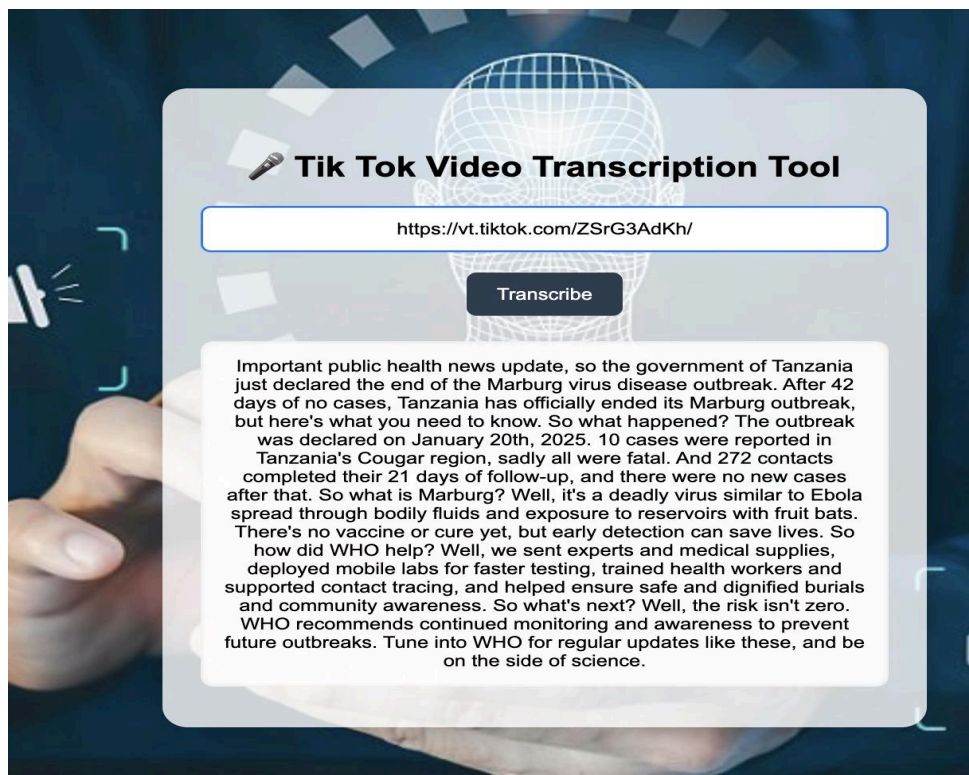


המערכת עובדת בצורה פשוטת ביותר, הלקוח יכניס קישור מהרשת החברתית Tik-Tok, ומיד יוכל לראות את המידע המתומלל בלי שום בעיה.

הכנסת כתובת ה url:

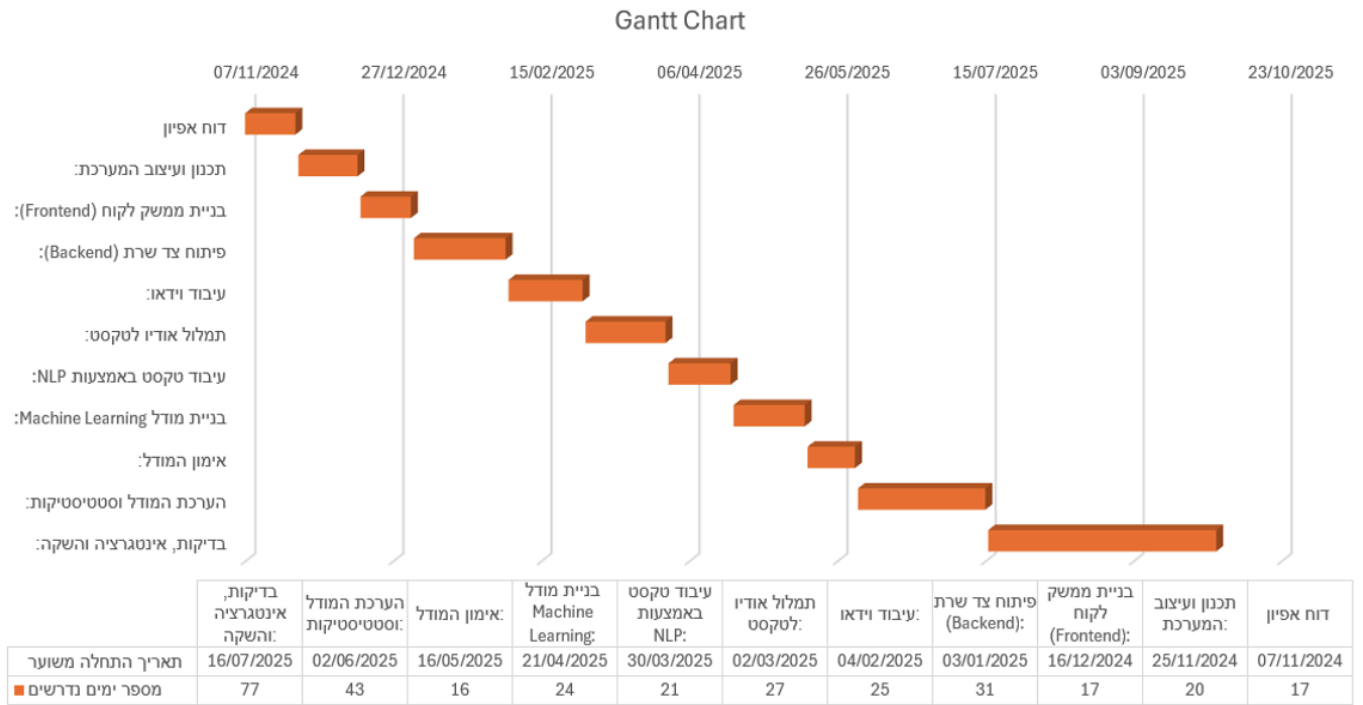


הצגת הטקסט על המסך:



בהמשך הפרויקט נמשיך לעצב את הממשק, הלקוח יראה נתונים על סמך data sets שנקבל ויוצא על המסך אמינות הסרטונים באמצעות למידת מכונה, כמובן שנבחר מסווג מתאים ונאמן אותו על סדרת אימון וסדרת מבחן כמובן שנבחר ספריות קלח מתאימות בהמשך. במהלך הפרויקט נתייעץ עם המנחה לגבי ממשק להעלאת הפרויקט לאתר וכמו כן איזה נתונים לשמור במסד הנתונים ובחירת צד שרת מתאים על מנת שהממשק שלנו לא יהיה לוקאלי ויציג את התוצר הסופי שלנו ברמה הגבוהה ביותר

גאנט :



לוח זמנים ואבני דרך:

תאריך סיום משוער	מספר ימים נדרשים	תאריך התחלה משוער	פירוט המשימה	משימות
24/11/2024	17	07/11/2024	כתיבת סקריה מפורטת על דרישות הפרויקט, רקע וסקירת ספרות, קביעת פגישה עם המרצה ותיאום ציפיות הפרויקט	דוח אפיון
15/12/2024	20	25/11/2024	הבנת דרישות המערכת.	תכנון ועיצוב המערכת:
02/01/2025	17	16/12/2024	יצירת קבצי HTML עם מבנה דפי האתר. הוספת עיצוב ותמיכה באמצעות CSS. יצירת ממשק ידיוטי ואינטראקטיבי בסביבת העבודה VSCode בנוסף פגישה עם המנחה	בניית ממשק לקוח (Frontend):
03/02/2025	31	03/01/2025	תכנון מבנה מאגרי המידע. בניית API לניהול התקשורת בין הלקוח לשרת באמצעות ספריית flask.	פיתוח צד שרת (Backend):
01/03/2025	25	04/02/2025	הכנסת סרטונים למערכת- הורדה באמצעות ספריית yt_dlp, חילוץ אודיו מהווידאו באמצעות ספריית ffmpeg.	עיבוד וידאו:
29/03/2025	27	02/03/2025	שימוש בספריית whisper לתמלול האודיו לטקסט.	תמלול אודיו לטקסט:
20/04/2025	21	30/03/2025	עיבוד טקסט בעזרת ספריית spaCy ביצוע ניתוח תחבירי,	עיבוד טקסט באמצעות NLP:

	זיהוי ישויות, וסיווג תוכן.			
בניית מודל Machine Learning:	תכנון המודל. בחירת אלגוריתם מתאים לסיווג טקסט (Fake/True).	21/04/2025	24	15/05/2025
אימון המודל:	שימוש בנתונים שנקבל על ידי המנחה על מנת לאמן את המסווג, שיפור ביצועי המודל.	16/05/2025	16	01/06/2025
הערכת המודל וסטטיסטיקות:	ביצוע הערכה סטטיסטית לתוצאות (מדדים כמו דיוק, רגישות). כתיבת דוח מסכם על ביצועי המודל (אופציונלי).	02/06/2025	43	15/07/2025
בדיקות, אינטגרציה והשקה:	בדיקת המערכת כולה (Frontend ו-Backend). בדיקות קצה לקצה. שיפור חוויית המשתמש והשקת הפרויקט. העלאת הפרויקט על ידי ממשק שיועלה לאתר ונגיש למשתמש.	16/07/2025	77	01/10/2025

תכנון מול ביצוע:

נעשו כמה שינויים באבני הדרך ובספריות ששומשו על מנת לדייק את הדברים בהם נעשה שימוש עד כה בפרויקט.

השינויים שנעשו:

1. **פיתוח צד שרת** - עדכון השימוש בספריית flask בפייתון על מנת ליצור קשר בין קובץ הפייתון לממשק האתר.

2. **עיבוד וידאו** - שימוש ספריית yt_dlp על מנת להוריד את הסרטון, עדכון השימוש בספריית ffmpeg על מנת לחץ את האודיו של הסרטון שהורד.

3. **אימון המודל** - נאמר לנו על ידי המנחה שנקבל data sets על מנת להשתמש בלימוד מכונה ובדיקת אמינות של הסרטונים הנמצאים במאגר הנתונים.

4. **בדיקות, אינטגרציה, והשקה** - בשלב הזה עודכן שנעלה את הפרויקט לממשק, נבדוק מה כדאי לעדכן במסד נתונים וכמובן כמה שיותר נגישות למשתמש.

רשימת מקורות :

[ClaimBuster](#) [1] •

[Azure Video Indexer](#) [2] •

[Speak AI](#) [3] •

[Google Fact Check](#) [4] •

[TinEye](#)[5] •

[Nlp library](#) [6] •

[Diagram App](#)[7] •