המכללה האקדמית להנדסה בראודה A black background with purple letters

Description automatically generated

מבוא למחשוב ענן פרויקט מסכם

**מגישים:**

1. **שי יוסף 209247600**
2. **יוסי שם טוב 315092445**
3. **איתי מוחבטי 206525008**
4. **עמנואל דוידוב 209122639**
5. **אופיר עוזיאל 313594699**
6. **אדם כיאל 211584271**

**קישור לתיקיית גיט:**

<https://github.com/yossishemtov/Introduction-to-Cloud-Computing-61773.git>

**קישור לקולאב:**

[HW2\_Wolf.ipynb - Colab (google.com)](https://colab.research.google.com/drive/1RGWPWedyuJkfwJ2nNIPCYudtdNPLUzVt?usp=sharing#scrollTo=KUslhM7Aw9ib)

הוראות הרצה:

על מנת להפעיל את הקוד יש להעלות את הקבצים הבאים לקולאב:

1. app.py

לאחר העלאת הקבצים יש ללחוץ על:

Runtime-> run all

1. **מהות המוצר ומרכיביו – מודולים ופיצ'רים מעניינים. יש להתיחס במיוחד ליעדים שקיבלתם בתרגיל בית 3, לפי מספר ה -JSON שהצוות שלכם עסק בו (נספח א בתרגיל בית 3).**

המוצר נועד לעזור למנהלי פרויקטים המשתמשים בתוכנת Onshape לזקק נתונים וסטטיסטיקות לגבי צוותי פרויקטים בפרויקטים שונים.

נתונים אלו עוזרים למנהל לראות מה היו הפעילויות העיקריות של העובד? האם הם יצירתיות, צפייה, או ניהוליות.

תמונה שמכילה טקסט, תרשים, צילום מסך, פתקייה

התיאור נוצר באופן אוטומטי2. **ארכיטקטורת המערכת – תרשים ומאפיינים מרכזיים.**

**3. Use case מעודכן של פונקציונליות המערכת.**

תמונה שמכילה טקסט, תרשים, צילום מסך, קו

התיאור נוצר באופן אוטומטי

4. **דרישות לא פונקציונליות - אתגרים איתם הפרויקט מתמודד (מסווגים לפי קישור WIKIPEDIA).**

דרישות לא פונקציונליות:

* תדירות: המערכת תבטיח שסנכרון הנתונים יתרחש כל 5 דקות.
* שלמות הנתונים: המערכת תשמור על שלמות הנתונים ועקביות במהלך הסנכרון.
* מדרגיות: המערכת תהיה ניתנת להרחבה לטיפול במספר אינטגרציות בו זמנית ללא ירידה בביצועים.
* זמינות: למערכת תהיה זמן פעולה של 99.9% כדי להבטיח סנכרון רציף.
* שימושיות: המערכת תספק ממשק אינטואיטיבי לניהול הגדרות סנכרון.
* ממשק משתמש: המערכת תספק ממשק ידידותי למשתמש להתאמה אישית של לוח המחוונים.
* היענות: המערכת תתמוך בהיענות גבוהה ובעדכונים בזמן אמת עבור ווידג'טים של לוח המחוונים.
* אבטחה: המערכת תבטיח אבטחה ופרטיות עבור תצורות ספציפיות למשתמש והדמיות נתונים.
* ביצועים: המערכת תפעל ביעילות עם זמן תגובה של פחות מ-2 שניות לאינטראקציות המשתמש.
* תאימות: המערכת תהיה תואמת לדפדפני אינטרנט ולצגים גדולים.
* דיוק: המערכת תעבד נתונים ותייצר תחזיות עם דיוק של 90% לפחות.
* זמן תגובה: המערכת תספק תחזיות והצעות תוך 10 שניות.
* פרטיות נתונים: המערכת תבטיח פרטיות נתונים ותעמוד בתקנות הגנת מידע רלוונטיות.
* חוסן: המערכת תהיה חזקה מספיק כדי להתמודד עם מערכי נתונים גדולים ללא ירידה בביצועים.
* שקיפות: המערכת תספק יומנים ודוחות מפורטים על תהליך הניתוח החזוי לשקיפות.

5. **תיק תחזוקה – תיאור של כל הקבצים והאובייקטים המרכזיים, ותיעוד קצר של כל פונקציה בקוד.**

**תיק תחזוקה - Wolf**

**סקירה כללית על המערכת:**

תיק זה מציג את הקבצים העיקריים, הפונקציות ודפוסי העיצוב המעניינים שהשתמשנו במערכת לניהול וניתוח פרמטרים וסטטיסטיקות. המערכת מורכבת ממספר מסכים, כל אחד משרת מטרה מסוימת, תוך התמקדות בממשק ידידותי למשתמש.

**שמות הקבצים המרכזיים:**

**App.py – קובץ מרכזי של הפרויקט אחראי על הלוגיקה של המערכת. מטרת הקובץ היא ליצור יישום אינטרנט באמצעות ספריית Streamlit המאפשרת למתכנת באופן נוח ליצור פיצרים עבור משתמשים כגון להעלות קבצים, להציג מידע אודותם, לסנן מידע ולנתח נתוני JSON.**

**עושה שימוש בספריות כגון: pandas, matplotlib, nltk ו-seaborn לצורך ניתוח נתונים, יצירת ויזואליזציות ועיבוד שפה טבעית.**

**פונקציות מרכזיות:**

* **categorize\_activity()** - פונקציה זו מסווגת את הפעילות על פי תיאור הפעולה (description). הפונקציה משתמשת במילות מפתח שונות כדי להחליט לאיזו קטגוריה לשייך את הפעולה.
* **load\_json()** - פונקציה זו מנסה לקרוא קובץ בפורמט json ולהחזיר את התוכן שלו. אם מתרחשת שגיאה במהלך התהליך (כלומר הפירמוט של הjson אינו כראוי), הפונקציה תופסת את החריגה ומציגה הודעת שגיאה למשתמש באמצעות פונקציית st.error של Streamlit.
* **display\_json()** - פונקציה זו משתמשת בשיטת st.json של Streamlit כדי להציג את נתוני ה-JSON בצורה שנוחה לקריאה ע"י משתמש.
* **filter\_json\_data()** - פונקציה זו עוברת על נתוני ה-json ומעדכנת את המידע לפי הסינונים שסופקו על ידי מנהל המערכת.
* **is\_date()** - פונקציית עזר זו בודקת האם מחרוזת נמצאת בתבנית תאריך ספציפית ('%Y-%m-%d'). אם כן, היא מחזירה True, מה שמציין שהמחרוזת היא תאריך. אחרת, היא מחזירה False.
* **admin\_screen()** - פונקציה זו מציגה כותרת ומאפשרת למנהל מערכת להעלות קבצי JSON ולצפות בהיסטוריית ההעלאות. היא שולפת את היסטוריית ההעלאות מ-Firebase ומציגה אותה. משתמשים יכולים להעלות קבצים חדשים, כאשר הקבצים נבדקים מפני כפילויות לפני שנוספים להיסטוריה ומעודכנים ב-Firebase. יש drop down המאפשר למשתמש לבחור ולטעון נתוני JSON מקבצים שהועלו בעבר. אם אין קבצים זמינים, מוצגת הודעה מתאימה.

כאשר קובץ מסוג json מועלה, היא קוראת לפונקציה load\_json כדי לטעון את הנתונים, אם הטעינה הצליחה, המידע מאוחסן בsession במשתנה בשם 'json\_data'.

* **display\_chat\_history()** - פונקציה זו מציגה את היסטוריית החיפושים של הצאט בוט על גבי המסך.
* **chatbot\_secreen()** - הפונקציה מאתחלת את הצאטבוט על גבי המידע הנטען, מציגה שאלות אשר ניתן לשאול את הצאטבוט וקולטת לחיצות ושאילתות לצאטבוט אשר עונה בהתאם להם, בשל בעיה שעלתה שמשתמש יכול לטעות בפרומט של השאלה הוספנו list שבלחיצה עליו נפתחות כל השאלות ובצורה זאת הבחירה בלחיצה ונמנענו מהקושי להזין שאלה ידנית.
* **parameter\_selection\_screen()** - פונקציה זו מאפשרת למשתמש לבחור פרמטרים מתוך נתוני ה-JSON הטעונים בsession ושומרת אותם במשתנה בשם 'filters, לאחר מכן היא מיישמת את הסינון באמצעות הפונקציה filter\_json\_data ומאחסנת את הנתונים המסוננים בsession במשתנה בשם 'filtered\_data'.

המשתמש יכול לבחור פרמטרים מתוך הרשימה 'filtered\_data' ולהזין ערכי סינון עבורם. בנוסף, הוא יכול לבחור טווח תאריכים עבור הסינון. אם קיימים נתונים התואמים למסננים שהמשתמש הזין, הוא יקבל הודעה על הצלחה. אחרת, תוצג הודעת שגיאה שאין נתונים התואמים למסננים שהוזנו. אם אין נתוני JSON זמינים, תוצג הודעה שיש להעלות קובץ JSON בדף ה- admin.

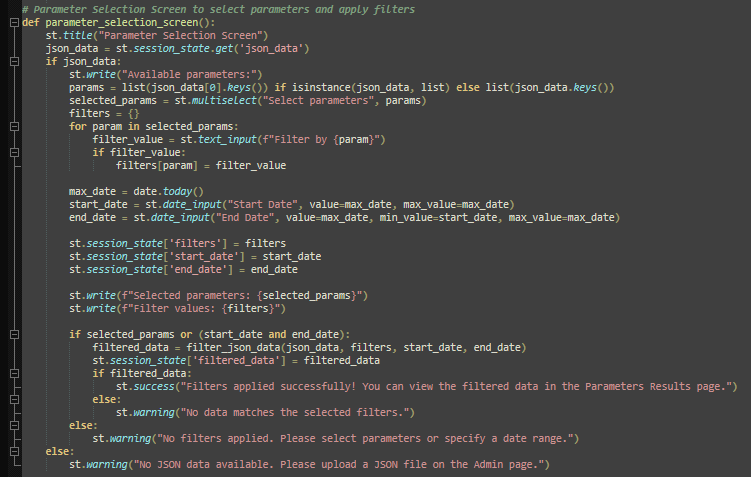
* **parameters\_results\_screen()**- פונקציה זו נועדה להציג את הנתונים המסוננים במסך תוצאות הפרמטרים באמצעות פונקציית st.write של Streamlit .אם קיימים נתונים מסוננים, הם מוצגים בטבלה וניתן להוריד אותם כקובץ Excel. אם אין נתונים זמינים, מוצגת הודעה למשתמש שעליו לבחור סינונים בעמוד בחירת הפרמטרים.
* **interesting\_statistics\_screen() -** פונקציה זו מציגה סטטיסטיקות וגרפים מעניינים על בסיס הנתונים המסוננים מהקובץ שהועלה. המשתמש יכול לבחור מתוך מגוון גרפים להצגה, כולל התפלגות פעילויות לפי קטגוריות, Heatmap של פעילויות משתמשים לפי יום ושעה, משתמשים עם מספר הפעולות הרב ביותר, והתפלגות סוגי פעילויות לאורך זמן. אם אין נתונים מסוננים, מוצגת הודעה למשתמש שעליו לבחור סינונים בעמוד בחירת הפרמטרים.
* **Main()** – פונקציה זו מגדירה את הניווט בסרגל הניווט הצדדי וקוראת לפונקציה המתאימה בהתבסס על בחירת המסך על ידי המשתמש.

**חלקי קוד מרכזיים (של המסכים):**

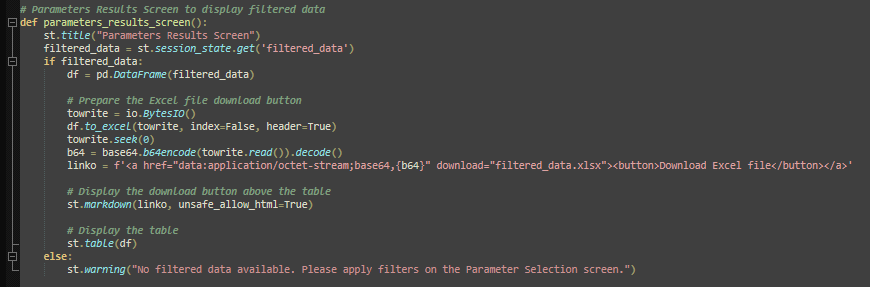
admin screen -

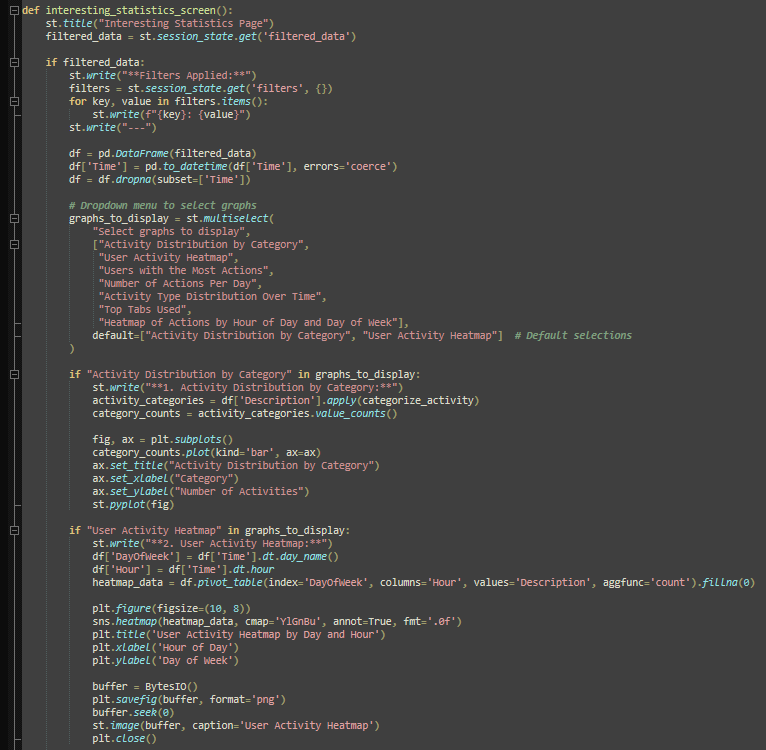


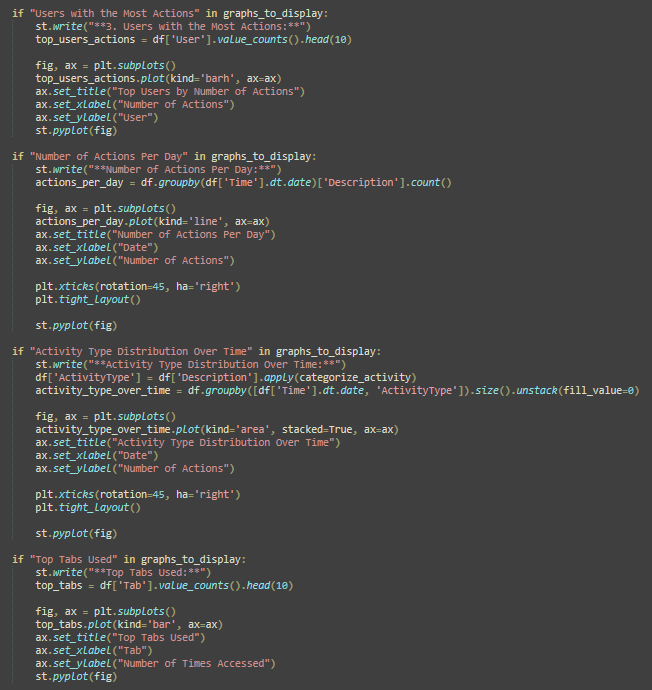
parameter selection screen -

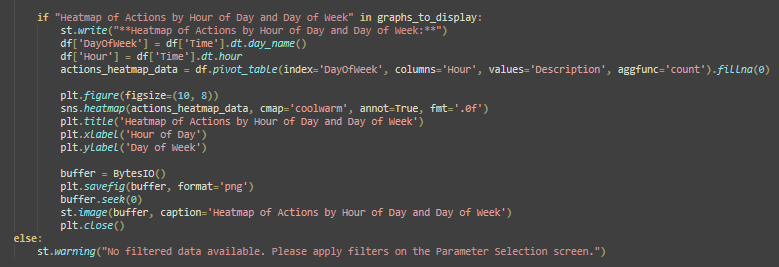


parameters results screen -

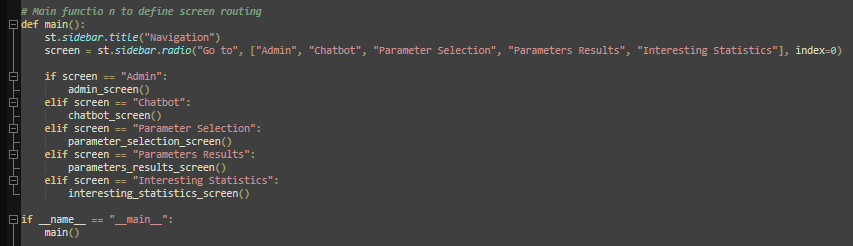


interesting statistics screen -





פונקציית main לטעינת המסכים בהתאמה -



6. **ביקורת עמיתים אשר ניתנה במהלך הצגת הפרויקט (שבוע 7), וכיצד התמודדתם איתה. נא לחשב גם את ציון SUS ולהתייחס אליו.**

תמונה שמכילה טקסט, צילום מסך, גופן, מספר

התיאור נוצר באופן אוטומטי

ציון ה-SUS הממוצע עבור המערכת הוא 67.6, שזה ציון יחסית נמוך. אך זה היה בשלביו הראשונים של הפרויקט לפני שהיה ממומש בפועל. לאחר מכן בתהליך המימוש התייחסנו להערות אלו ועשינו את המערכת פרקטית, אינטואיטיבית וידידותית למשתמש ככול הניתן.

**7. אתגרים אשר עלו במהלך העבודה, וכיצד התמודדתם איתם.**יש להתייחס ל:

א. מטלת design thinking אשר ביצעתם בתרגיל בית יבש 1.

חשיבה עיצובית: אתגרים ופתרונות

1. הבנת צרכי המשתמש:

אתגר: אחד האתגרים הראשונים היה זיהוי מדויק של צרכיו של המשתמש במערכת.

פתרון: חשבנו מה אנחנו בתור מנהל מערכת היינו רוצים לראות, ערכנו ראיונות משתמשים ויצרנו פרסונות. דבר זה עזר לנו להבין את צורכי המשתמשים.

2. הרעיון:

אתגר: יצירת רעיונות חדשניים הפותרים את הבעיות שזיהינו..

פתרון: במהלך שלב הרעיונות, קיימנו מפגש סיעור מוחות תוך שימוש בטכניקות חשיבה עיצובית. זה אפשר לנו לחשוב על מגוון רחב של פתרונות לפני שבחרנו בפתרונות שבחרנו.

3. יצירת אב טיפוס ובדיקה:

אתגר: תרגום הרעיונות לאב טיפוס שמשתמשים יוכלו לראות ולקבל הסבר (סטודיו שבוע 7) כדי לאסוף משוב.

פתרון: בשלב אב הטיפוס, יצרנו מסכים שידגימו את הרעיון של הפרוייקט באמצעות Figma. אב טיפוס זה נבדקו ע"י משתמשים כדי לאסוף משוב בשלב מוקדם של התהליך, מה שאיפשר לנו לבצע פעולות חוזרות במהירות ולשפר את העיצוב לפני פיתוח בקנה מידה מלא.

4. יישום ואיטרציה:

אתגר: שילוב משוב משתמשים במוצר הסופי תוך ניהול אילוצי זמן ומשאבים.

פתרון: אימצנו גישה זריזה בשלב היישום, המאפשרת בדיקות ואיטרציה מתמשכות. נתנו עדיפות למשוב של משתמשים שהתאים בצורה הכי קרובה למטרות הפרויקט, והבטחנו שהתוצר הסופי יהיה גם ידידותי למשתמש וגם נמסר בזמן.

|  |  |  |
| --- | --- | --- |
| **הערה** | **תגובה** | **האם בכוונת הצוות לבצע שינוי בעקבות ההערה? יש לפרט. אם כן, מה השינוי. אם לא, מדוע** |
| אין אפשרות לפלטר לפי טווח של תאריכים | נכון | כן, יתווסף אפשרות לבחור טווח תאריכים ולא רק תאריך ספציפי. |
| הזנת תאריך לפי בחירה (קליק) מאשר להזין טקסט ידני | צודקים | נשנה את הזנת התאריכים מהזנה ידנית לבחירת תאריך באמצעות העכבר, מטעמי נוחות. |
| חזרה מהresult לselection עושה reset לפילטור | פחות מסכימים | לא, נראה לנו הגיוני שאם מישהו חוזר לSELECTION הנתונים יתאפסו כי ברצונו (יוסר) להזין נתונים חדשים. |

ב.התייחסות למשוב שקיבלתם בשבוע 10 – code review – יש להגיש את המשוב שקיבלתם מהצוות השני בחלק הקבוצתי , ולכתוב התייחסות מתאימה:לכל הערה, יש לרשום אותה ולציין האם נעשה שינוי בקוד בעקבות זאת.

8**. שקיפות אלגוריתמית – כיצד הבהרתם למשתמשים את האלגוריתמים והנתונים הנאספים בקוד שלכם?**

עבור השקיפות האלגוריתמית, בעת ביצוע סינון הצגנו את המסננים שנבחרו על ידי המשתמש, דבר שמסביר את סדר הצגת התוצאות בטבלה.

בנוסף, אנו מציגים גרפים וסטטיסטיקות שקשורים למידע השמור במסד הנתונים (קבצי הJSON), דבר זה מספק תובנות למשתמש אודות המידע שנאסף ומוצג לו.

9. **בשבוע האחרון בקורס תציגו את המערכת שלכם. עליכם להתייחס למצגות בסשן שלכם. ההצגה ומילוי המשוב הם חובה.**

**לאחר ההצגה:**

**תקבלו באופן אנונימי את המשובים של חבריכם, וכן את המשוב שלנו.**

**יש להגיש את הטבלה הבאה , תוך התיחסות למשובים שקיבלתם:**

|  |  |  |
| --- | --- | --- |
| הערת משוב | האם לדעתכם יש צורך בשינוי במערכת בעקבות ההערה? | נימוק |
| נראה די טוב ועונה על הדרישות בצורה טובה. | לא |  |
| התפריט ניווט יכול להיות בצורה אחרת יותר נוחה וברורה למשתמש | לא | המערכת נוחה ואינטואיטיבית לשימוש. |
| הקבוצה הזאת היא ששעתה לנו CODE REVIEW ואת הדברים שהיו לדעתי צריך לשפר אמרתי להם ב CODE REVIEW והם שינו את זה בהתאם. אז לדעתי אין מה לשפר | לא |  |
| להזיז את הכפתור ASK מתחת ל INPUT של המשתמש ב CHATBOT | לא | אין צורך בשינוי.הכפתור נמצא במקום בולט לעין. |
| side menu radio buttons are not appropriate in my honest opinion | לא | המערכת נוחה ואינטואיטיבית לשימוש. |
| את הביצועים של באתר הוא קצת איטי | לא | זה משום הNGROK חינמי אין לנו דרך לטפל בזה. |
| המערכת קצת איטית,ניתן לשפר את המהירות | לא | זה משום הNGROK חינמי אין לנו דרך לטפל בזה. |
| הצ'אט בוט נראה לא כל כך מסודר, החלוקה לבחירת שאלה ספציפית הופכת את זה יותר לדף שאלות תשובות ומאשר לדף צ'אט בוט. אפשרות של בחירה מרובה בפילטוריזציה. | לא | להפך, עצם זה שגם אפשר להזין טקסט וגם אפשר לבחוק אלות מקנה למשתמש יותר אפשרויות לשימוש נוח. |
| אפשר להוסיף הסברים בגרפים | לא | הגרפים דיי אינטואיטיבים. ולכל גרף יש כותרת שמציגה את תפקיד הגרף. |
| שימוש באפשרויות סינון: לאחר כל בחירת סינון האתר נכנס למצב טעינה (מפעיל את הפילטור) הצעה:  לעשות את הפילטור לאחר סיום כל הבחירות של הסינון בעזרת כפתור של חפש/סנן | לא | זה קורה במצב שבחירת הפילטור עובדת על עשרות אלפי שורות מידע. כמובן ששימוש בגסאות חינמיות לא מקל על התופעה. |
| גרפים שיהיו יותר ברורים ומדוייקים מבחינת נתונים, פילטרציה ברורה יותר | לא | הגרפים דיי אינטואיטיבים. הפלטור מתבצע בעמוד של הפלטור, ניתן לסנן לפי ימים, לפי טאבים לפי משתמשים ולפי פרוייקט |

10. **מקורות. יש לצטט את המקומות מהם אתם לוקחים את הנתונים השונים. יש לכלול פרומפטים לכלי AI, במידה והשתמשתם בהם.**

. ChatGPT

פרומפטים ששאלנו את הצאט:

* How does this code interact with Firebase, and what kind of data is it retrieving?
* How is the data being visualized using Matplotlib and Seaborn in this app?
* How does the code handle potential errors or exceptions, especially when dealing with external data from Firebase?
* How is the NLTK library being used in this code for chat functionality?
* What are the 'Chat' and 'reflections' from NLTK, and how are they utilized in this project?
* How does this code use Streamlit to create an interactive web app?
* What are the key Streamlit components used in this app, and how do they contribute to the user interface?
* How is the app handling file inputs and outputs, especially with respect to `io` and `BytesIO` modules?
* What is the purpose of the `base64` encoding and decoding in this code?
* How is the `datetime` module being utilized in this project?

2. GeeksforGeeks - שימש למידע בתכנות ב-Python.

3. Stack Overflow - שימש לפתרון בעיות בנושא Python, תוך שימוש בהצעות שנכתבו על ידי מפתחים.

4. W3Schools - שימש להבנה ופיתוח אתרים.