**תיק למתכנת- 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 זה 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.

**Admin\_screen.html** – דף זה כולל טופס להעלאת קבצי טקסט בפורמט json. הוא נועד כדי לאפשר למנהלי מערכת להעלות את הקובץ שממנו הם רוצים להפיק מידע לגבי עובדיהם.

**Parameters\_selection.html** - דף זה מספק ממשק משתמש לבחירת פרמטרים מנתוני הjson וציון ערכים לפיהם המידע יסונן. זה עוזר למשתמשים לחדד את הנתונים שהם מעוניינים בהם.

**Parameters\_result.html** - דף זה מציג את התוצאות לאחר שנבחרו פרמטרים לסינון. אפשרות זו עוזרת למנהלי המערכת לראות את הנתונים המסוננים בצורה ברורה ומובנית. למשל עבור עובד ספציפי או עבור תאריך מסוים.

**Interesting\_selection.html** - דף זה נועד להציג תובנות וסטטיסטיקות בצורות שונות שעשויות לעניין את מנהל המערכת על העובדים שלו. הוא משלים את המסכים האחרים על ידי מתן סקירה רחבה יותר של הנתונים מהקובץ שנטען למערכת.

**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.
* **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.

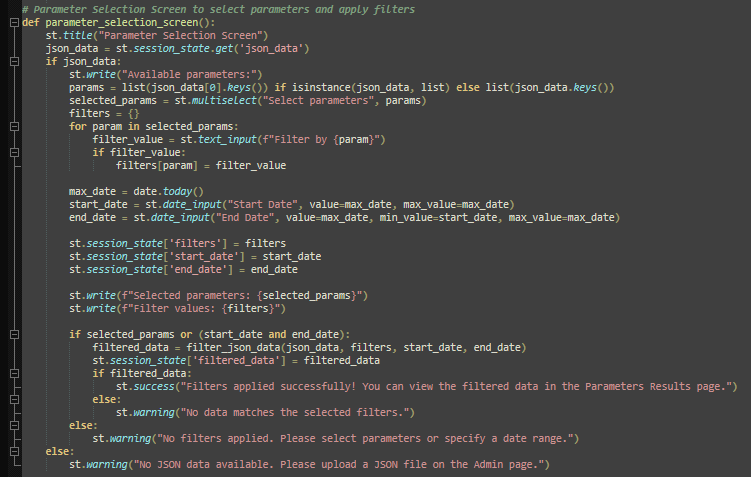
* **parameters\_results\_screen()**- פונקציה זו נועדה להציג את הנתונים המסוננים במסך תוצאות הפרמטרים באמצעות פונקציית st.write של Streamlit .אם קיימים נתונים מסוננים, הם מוצגים בטבלה וניתן להוריד אותם כקובץ Excel. אם אין נתונים זמינים, מוצגת הודעה למשתמש שעליו לבחור סינונים בעמוד בחירת הפרמטרים.
* **interesting\_statistics\_screen() -** 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):**

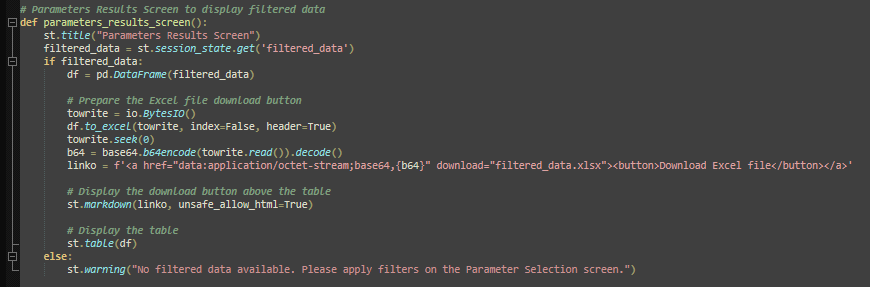
admin screen -

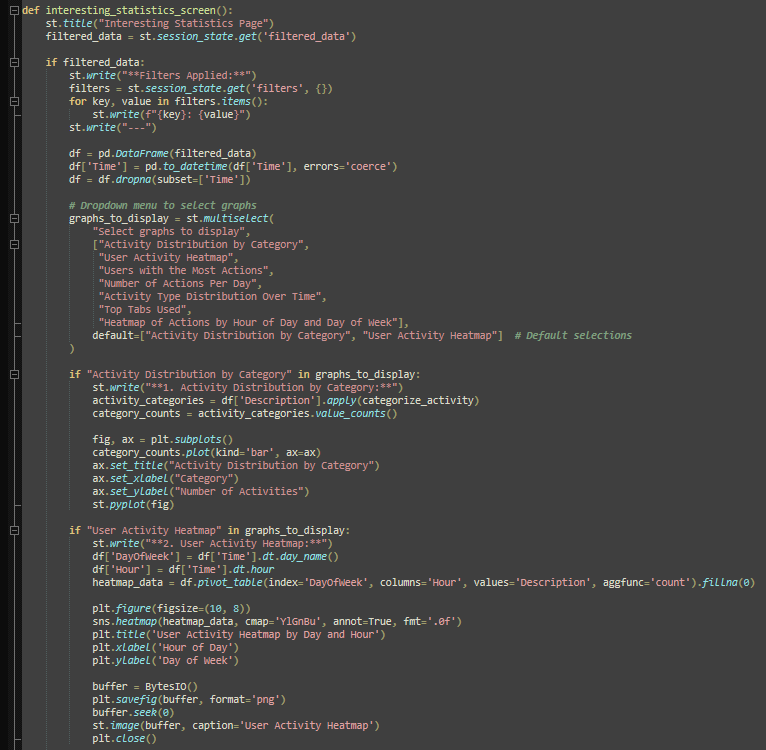


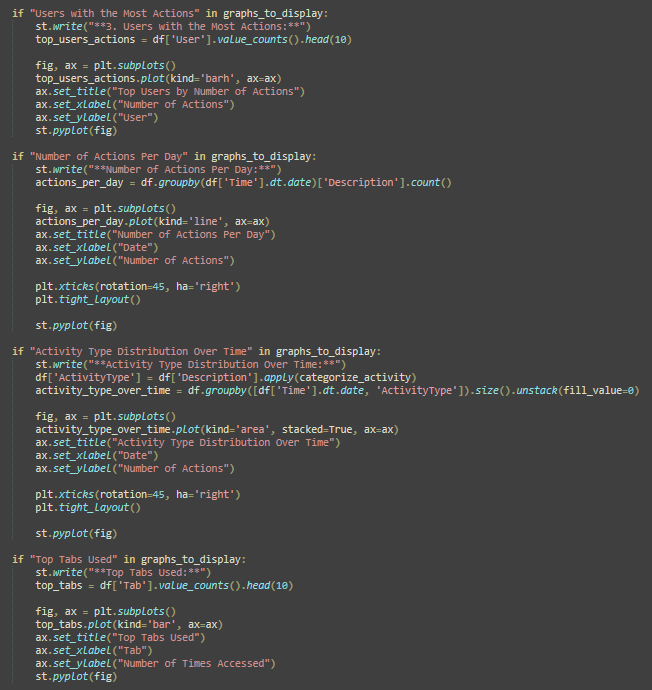
parameter selection screen -

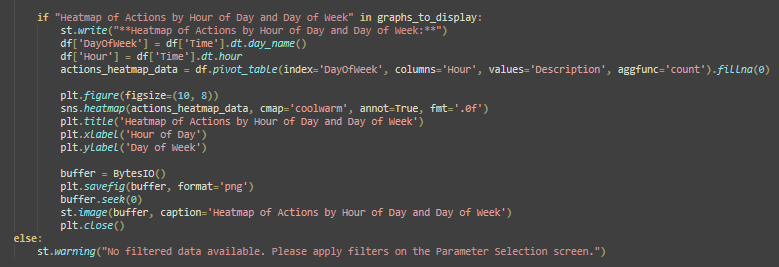


parameters results screen -



interesting statistics screen - 





פונקציית main לטעינת המסכים בהתאמה -

