**פרויקט מסכם – מחשוב ענן**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **אלעד קראוז** | **לי פוטשניק** | **אורי זיו** | **דניאל ברקוביץ** | **אהרון ינוטייב** |
| 207281874 | 318158565 | 206131849 | 324304625 | 203899786 |

**קישור ל-Git של הפרויקט שלנו:** [כאן](https://github.com/Eladkrauz/cloud_computing_course)  
**קישור לקוד שלנו ב-Google Colab:** [כאן](https://colab.research.google.com/drive/1lGk9SHlHmSzi_eA6zQD8erYf3Eml1L6R?usp=sharing)

**1. מהות המוצר ומרכיביו:**

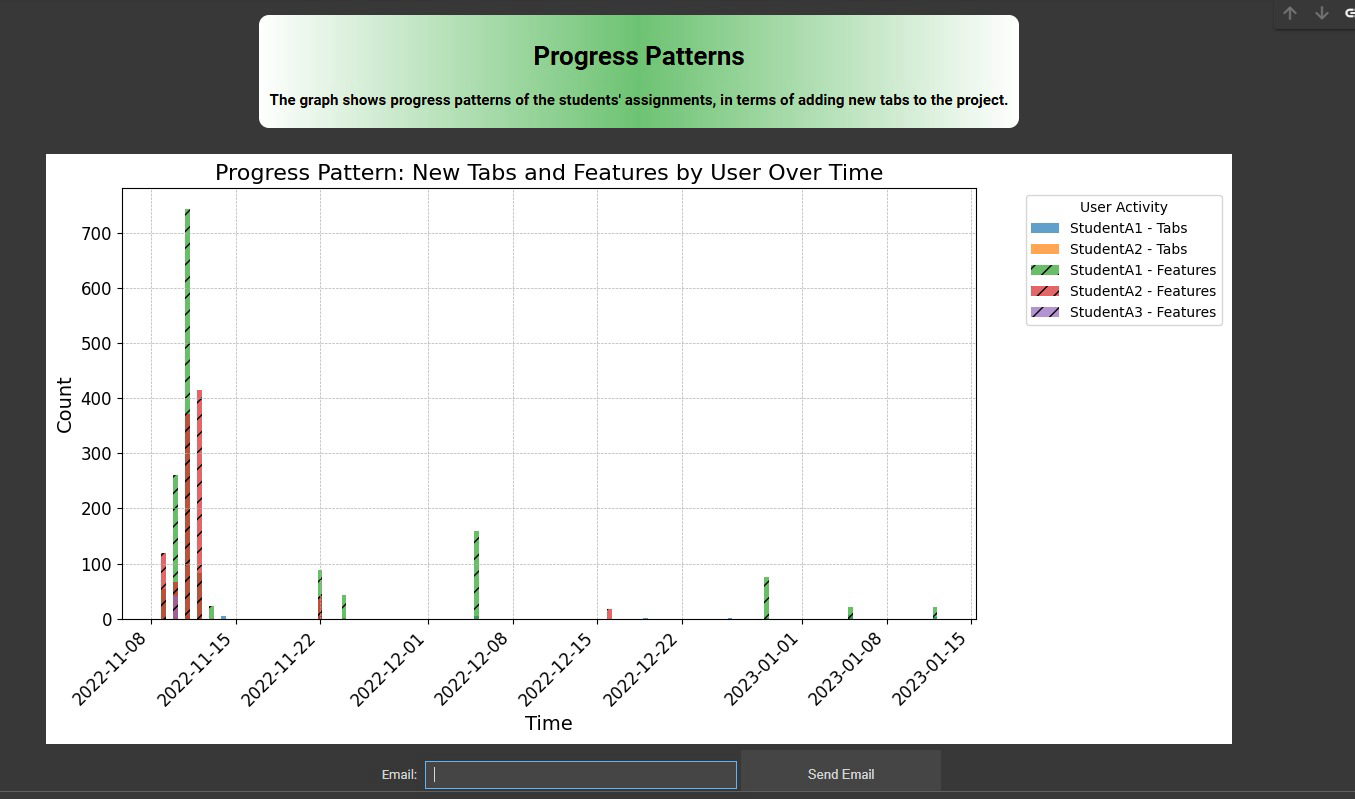
המוצר המדובר הוא פלטפורמה לניהול פרויקטים והמחשה של נתונים, שנועד לסייע למשתמשים במעקב וניתוח של היבטים שונים בפרויקטים שלהם. הכלי משלב תכונות כגון מעקב אחר פעילות משתמשים, ניטור גישה למסמכים, והמחשה של דפוסי התקדמות, כדי לספק למשתמשים תובנות ברורות ומפורטות על זרימות העבודה בפרויקט.

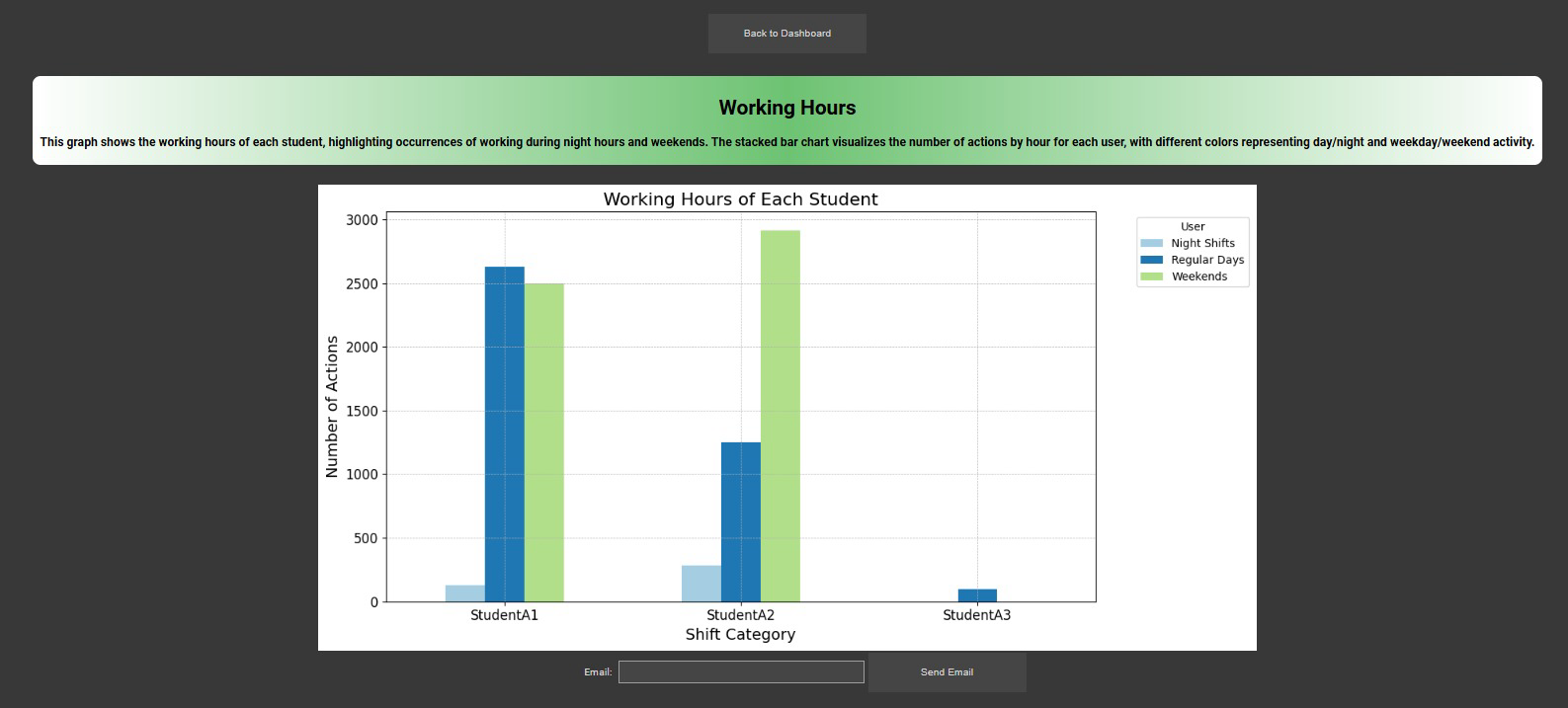
המרכיבים שלו כוללים ספריות ומודולים מתקדמים כגון Matplotlib ליצירת גרפים, Firebase לאחסון נתונים, Seaborn עבור פלטות צבעים ו-Pandas למניפולציה של נתונים. כל המרכיבים הללו פועלים יחד כדי לספק חוויה אינטראקטיבית, המאפשרת למשתמשים לחקור את נתוני הפרויקט שלהם ביעילות דרך גרפים ודיאגרמות המיוצרים באופן דינמי. שילוב של שירותי דוא"ל ו- Chat Bot משדרגים את הכלי ומאפשרים תקשורת חלקה וסיוע מותאם אישית.

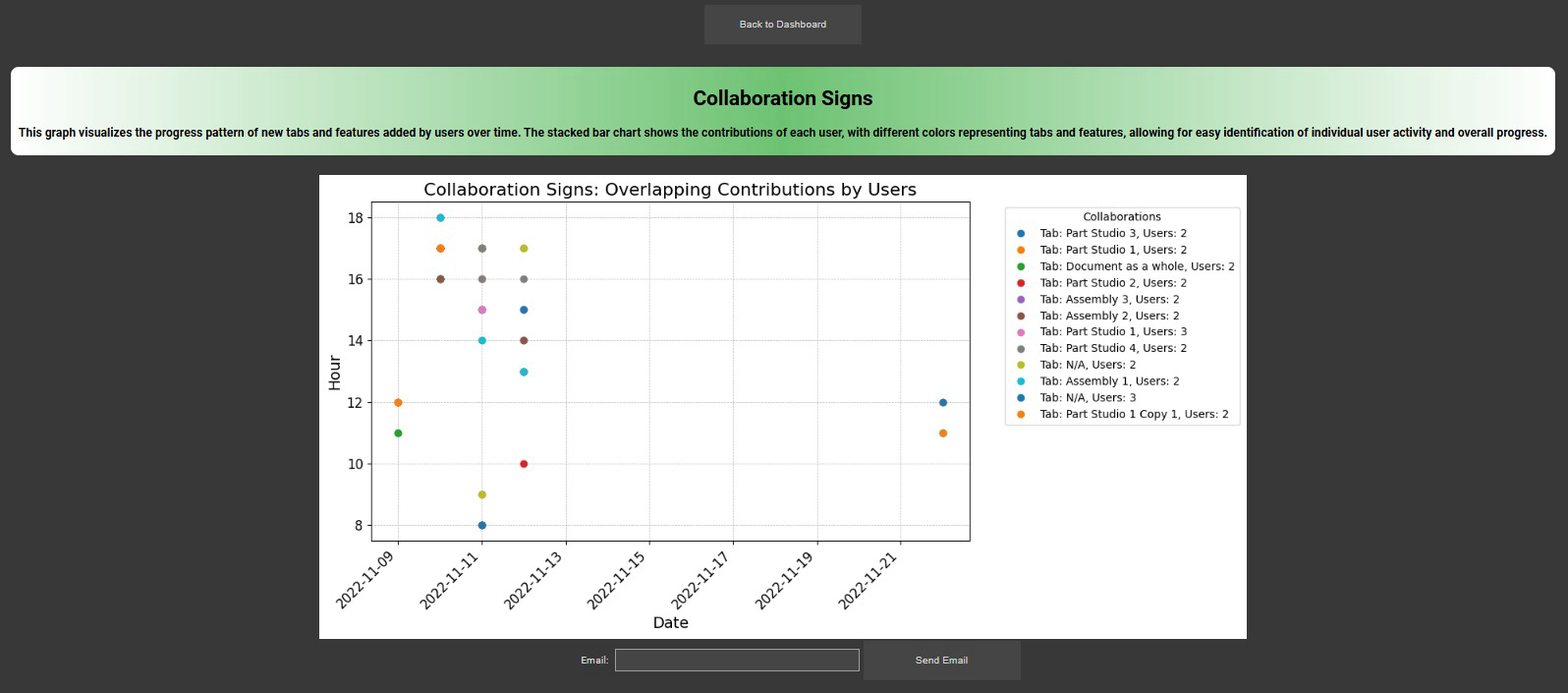
אלמנט ה-JSON שקיבלנו:

find a progress pattern of the students’ assignment: how they create new tabs and add features, who is the main contributor to the task? is there a pattern in the days/hours they worked? can you find collaboration signs (e.g. working during the same hours, contributing to the same tab)?

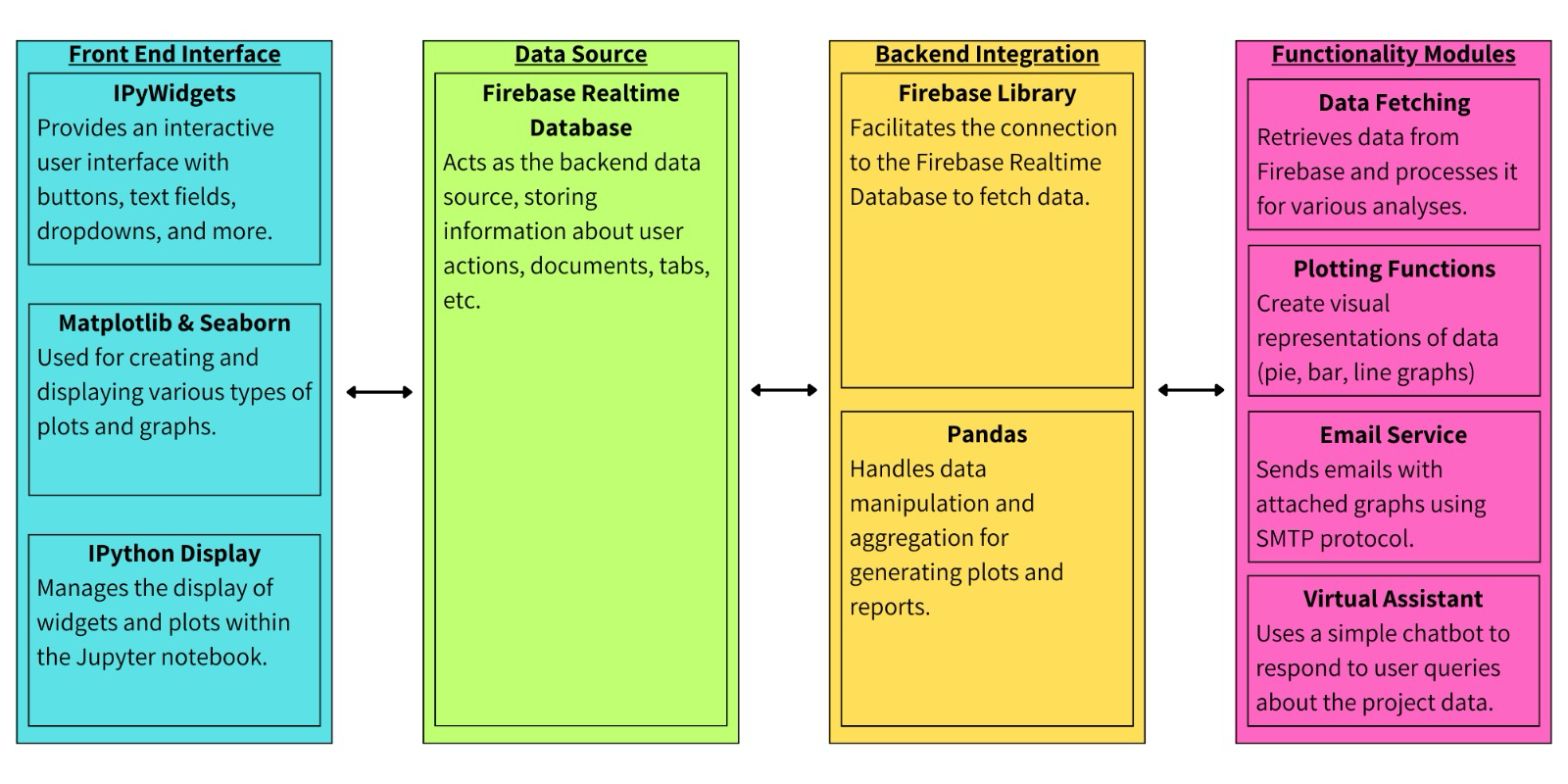
נבחן כעת כיצד התייחסנו לכל אלמנט שנשאל בעת מימוש המערכת, כך שיהיה ברור למשתמש כיצד כל שאלה שהועלתה מוצאת ביטוי בקוד:



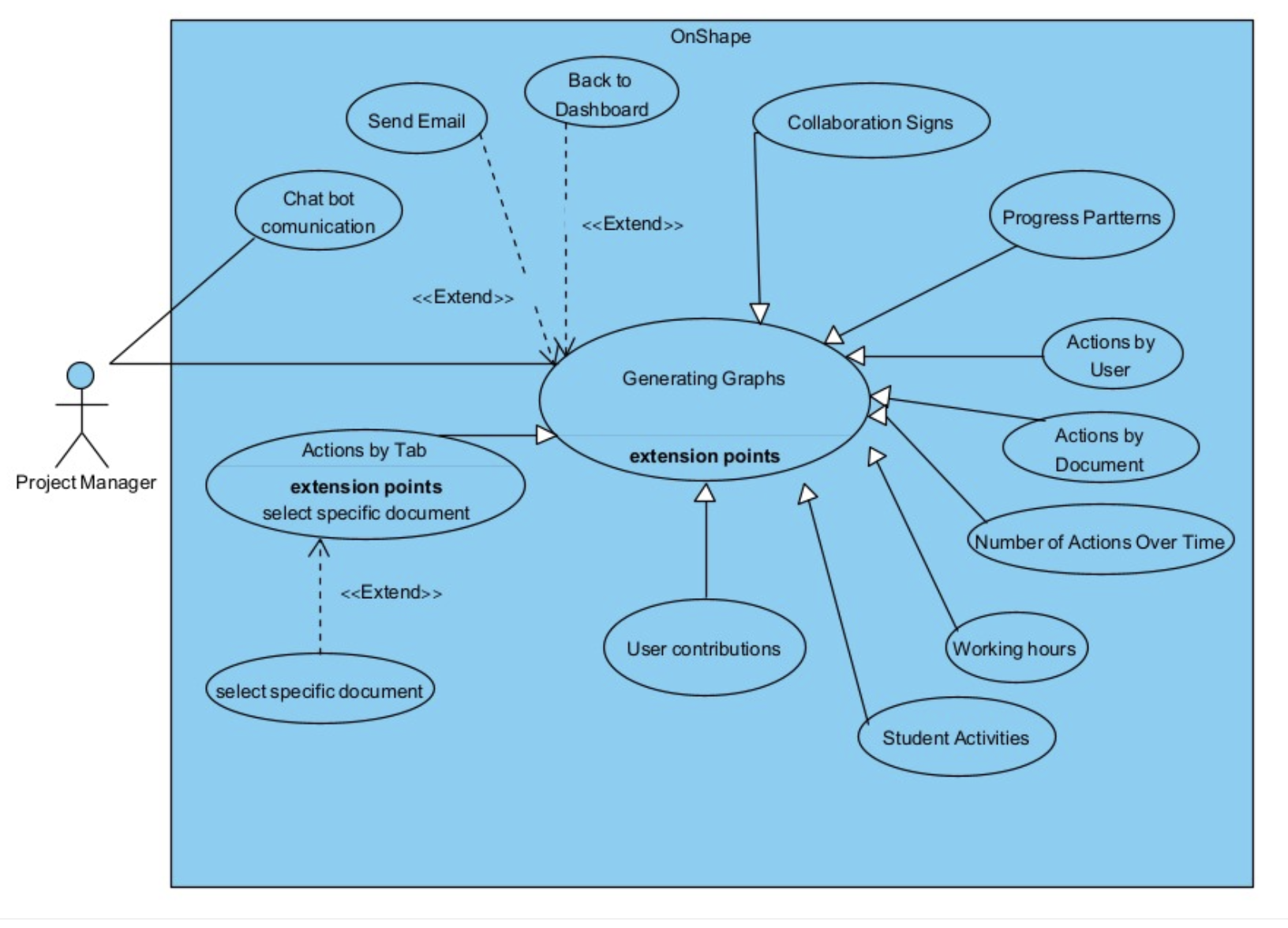




**2. ארכיטקטורת המערכת:**

****

**3. Use Case:**

****

**4. דרישות לא פונקציונליות:**

|  |  |
| --- | --- |
| **אבטחה** | על המערכת לאכוף אבטחת מידע ופרטיות באמצעות הצפנה ובקרת גישה מבוססת תפקידים על מנת להגן על נתוני העובדים ונתוני פרויקטים. |
| **ויזואליזציה** | על המערכת לספק גרפים והדמיות סטטיסטיות של מדדי ביצועים, תוך הבטחת הצגת נתונים ברורה ונגישה לכל המשתמשים. |
| **תאימות** | המערכת חייבת להיות תואמת להתקנים ומערכות הפעלה שונות. |
| **לוקליזציה** | תמיכה בשפה האנגלית, עם אפשרות הוספת שפות נוספות. |
| **תחזוקה** | המערכת תהיה קלה ונוחה לתחזוקה, עדכוני גרסה ותיקוני באגים ותקלות תחזוקה. |
| **עיצוב** | העיצוב צריך להתמקד בפשטות, שה-UI יהיה אינטואיטיבי וקל לשימוש תוך הימנעות מאלמנטים מיותרים. |

**5. תיק תחזוקה:**

**פירוט הפונקציות:**

|  |  |
| --- | --- |
| **fetch\_data\_from\_firebase** | קוראת ומחזירה נתונים ממסד הנתונים. |
| **send\_email\_with\_attachment** | שולחת מייל עם קובץ מצורף באמצעות SMTP. |
| **create\_send\_email\_button** | יוצרת כפתור ושדה קלט לשליחת אימייל עם קובץ מצורף. |
| **show\_popup** | מציגה הודעה קופצת עם כותרת והודעה. |
| **create\_main\_dashboard** | יוצרת את ה-Dashboard הראשי עם כפתורים לפעולות שונות. |
| **display\_main\_dashboard** | מציגה את ה-Dashboard הראשי ומקצה Events לכפתורים. |
| **create\_page** | יוצרת עמוד בודד עם כותרת ווידג'טים נוספים אופציונליים. |
| **truncate\_labels** | מקצרת תוויות לגרפים על מנת להציגן בצורה ברורה על המסך. |
| **plot\_actions\_by\_user** | מציגה תרשים עוגה המציג את מספר הפעולות שביצע כל משתמש ושומרת את הגרף כתמונה. |
| **plot\_actions\_by\_document** | מציגה תרשים עמודות המציג את מספר הפעולות עבור כל מסמך ושומרת את הגרף כתמונה. |
| **plot\_number\_of\_actions\_over\_time** | מציגה את מספר הפעולות לאורך זמן באמצעות תרשים קווים ושומרת את הגרף כתמונה. |
| **plot\_actions\_by\_tab** | מציגה את מספר הפעולות לפי Tab עבור מסמך ספציפי ושומרת את הגרף כתמונה. |
| **display\_document\_selection** | מציגה תפריט נפתח לבחירת מסמך כדי להציג פעולות לפי Tab. |
| **plot\_progress\_pattern** | מציגה את דפוס ההתקדמות של Tabים ותכונות חדשות שנוספו על ידי משתמשים לאורך זמן ושומרת את הגרף כתמונה. |
| **plot\_collaboration\_signs** | מציגה סימנים של שיתוף פעולה על ידי הצגת תרומות חופפות של משתמשים לאותם Tabים במהלך אותן שעות ושומרת את הגרף כתמונה. |
| **plot\_working\_hours** | מציגה את שעות העבודה של כל משתמש, מדגישה מקרים של עבודה במהלך לילה, סופשים וחגים, ושומרת את הגרף כתמונה. |
| **plot\_user\_contributions** | מציגה את התרומה של כל משתמש להתקדמות המשימה, ממוין לפי סוג הפעולה באמצעות תרשים עמודות מוערם ושומרת את הגרף כתמונה. |
| **plot\_student\_activities** | מציגה את הפעילויות העיקריות של כל משתמש תוך סיווג לפעולות יצירתיות, ניהוליות וצפייה, באמצעות תרשים עמודות מוערם ושומרת את הגרף כתמונה. |
| **display\_plot\_with\_send\_email** | מציגה גרף מוכן וכפתור לשליחת הגרף בדוא"ל. |
| **display\_actions\_by\_user** | מציגה את הדף עבור הגרף של פעולות לפי משתמש. |
| **display\_actions\_by\_document** | מציגה את הדף עברו הגרף של פעולות לפי מסמך. |
| **display\_number\_of\_actions\_over\_time** | מציגה את הדף עבור הגרף של מספר הפעולות לאורך זמן. |
| **display\_actions\_by\_tab** | מציגה את הדף לבחירת מסמך עבור גרף פעולות לפי Tab. |
| **display\_progress\_patterns** | מציגה את הדף עבור גרף ההתקדמות של Tabs ותכונות חדשות. |
| **display\_collaboration\_signs** | מציגה את הדף עבור גרף סימני שיתוף פעולה בין משתמשים. |
| **display\_working\_hours** | מציגה את הדף עבור גרף שעות העבודה של המשתמשים. |
| **display\_user\_contributions** | מציגה את הדף עבור גרף תרומות המשתמשים להתקדמות. |
| **display\_student\_activities** | מציגה את הדף עבור גרף הפעילויות העיקריות של המשתמשים. |
| **get\_top\_actions** | מחזירה את הפעולות המובילות שבוצעו על ידי משתמש ספציפי. |
| **set\_patterns** | מגדירה דפוסי מענה עבור הצ'אט בוט לתגובה על לשאילתות שונות. |
| **handle\_specific\_user\_query** | מטפלת בשאילתות משתמש ספציפיות הקשורות לפעולות, מסמכים ו-Tabs. |
| **display\_virtual\_assistant** | מציגה את העמוד של הצ'אט בוט ומגדירה את הממשק שלו. |

**קבצים מרכזיים:**

|  |  |
| --- | --- |
| **קובץ JSON** (עם נתוני הפעולות שבוצעו בפרויקט) | **CloudProject-Tiger-Final.ipynb** |

**אלמנטים מרכזיים בקוד:  
טיפול בשגיאות בקריאות מול מסד הנתונים:** דפוס זה מבטיח שאם אחזור הנתונים של Firebase נכשל או חוזר ריק, המשתמש מקבל הודעה ברורה, ומונע שגיאות בלתי צפויות.

**יצירת גרפים דינמיים עם פונקציונליות לשמירה:** הקוד יוצר באופן דינמי גרפים ומיד שומר אותם כתמונות, מה שמאפשר למשתמשים לייצא ולשתף נתונים חזותיים ישירות מהממשק.

**גרפים גמישים עם אינטראקטיביות למשתמש:** דפוס עיצוב זה מאפשר למשתמשים לקיים אינטראקציה עם הממשק כדי ליצור גרפים ספציפיים, עם הפונקציונליות הנוספת של שליחת גרפים אלו באמצעות דואר אלקטרוני, מה שמשפר את האינטראקטיביות והשימושיות של האפליקציה.

**עוזר וירטואלי מותאם אישית עם עיבוד שפה טבעית:** השילוב של צ'אט בוט מבוסס NLP מאפשר למשתמשים לבצע שאילתות בנתוני הפרויקט באמצעות שיחה, מה שהופך את המערכת ליותר ידידותית ואינטואיטיבית למשתמשים שאינם טכניים.

**עיצוב רספונסיבי עם ווידג'טים וגרידים:** השימוש ב-GridBox ובווידג'טים מבטיח שה-Dashboard מגיב, עם כפתורים ואלמנטים מיושרים על פני גדלי מסך שונים, ומשפרים את חווית המשתמש הכוללת.

**יצירת דפים מודולריים עבור תצוגות שונות:** הגישה המודולרית ליצירת עמודים מאפשרת הוספה או שינוי קלה של תוכן, מה שהופך את המערכת לגמישה וניתנת לתחזוקה.

**6. ביקורת עמיתים הניתנה בשבוע 7:**

|  |  |  |
| --- | --- | --- |
| **הערת משוב** | **האם התבצע שינוי במערכת בעקבות ההערה?** | **נימוק** |
| להראות אלמנטים בנפרד ולא לחוד | כן | פיצלנו את האלמנטים למסכים שונים |
| לסדר את המידע בצורה שתהיה מסודרת לעין | כן | שינינו את סדר הצגת המידע, פיצלנו מידע למסכים שונים ושינינו את הגדלים לפי החשיבות |
| להיות יותר קלה מבחינת שימוש | כן | הוספנו כפתורים ומסכים, כך שהמערכת תהיה יותר נגישה למשתמש ויותר מובנת |
| יותר להתאים לפורמט של הגוגל קולאב | כן | הקוד של המערכת נכתב בגוגל קולאב |
| עיצוב | כן | שינינו את העיצוב של האתר כך שיש יותר מגוון של צבעים ועיצוב נקי |
| הרבה טקסט | כן | פיצלנו את הטקסט למסכים שונים כך שיש פחות טקסט בכל מסך |

ציון ה-SUS שהתקבל משאלון ה-SUS על המערכת שלנו הינו: **96.36904761904762**.  
ציון זה ממקם את המערכת שלנו ברמת שימושיות מצוינת, הרבה מעל הממוצע, ומעיד על התאמתה לצרכי המשתמשים בצורה טובה. מצביע על רמת שביעות רצון גבוהה במיוחד מהממשק והחווית משתמש של המערכת שלנו, כלומר המשתמשים מוצאים את המערכת יעילה וקלה לשימוש.

**7. אתגרים:  
חלק א' – אתגרים והתמודדות עמם:**

|  |  |
| --- | --- |
| **אתגר** | **דרך התמודדות** |
| **אינטגרציה עם Firebase** | כדי להתמודד עם האתגר של אינטגרציה עם Firebase, התמקדנו ביצירת חיבור מאובטח ויעיל על ידי הבנה מעמיקה של ה-API של Firebase, הטמעת שיטות עבודה מומלצות לאחזור נתונים והבטחת טיפול בשגיאות לאינטראקציה חלקה בין קוד ה-Python למסד הנתונים של Firebase. |
| **שליחת דוא"ל עם קבצים מצורפים** | כדי להתמודד עם האתגר של שליחת מיילים עם קבצים מצורפים, הטמענו שיטה באמצעות SMTP תוך הבטחת קידוד וטיפול נכון בקבצים. שילבנו גם הודעות שגיאה ידידותיות למשתמש כדי לשפר את חווית המשתמש הכוללת. |
| **תצוגת דשבורד אינטראקטיבי** | כדי להתגבר על האתגר של יצירת תצוגת לוח מחוונים אינטראקטיבית, השתמשנו ב-ipywidgets ו-matplotlib ב-Python, תוך עיצוב ממשק ידידותי למשתמש שמתעדכן באופן דינמי על סמך קלט המשתמש תוך שמירה על פריסה נקייה ואינטואיטיבית ליצירת גרפים יעילים של נתונים. |
| **בינה מלאכותית לשיפור תהליכים** | כדי לשפר תהליכים באמצעות בינה מלאכותית, שילבנו אלגוריתמי למידת מכונה במערכת שלנו, מה שאפשר קבלת החלטות אוטומטיות וניתוח חזוי. דבר זה בעצם ייעל את הפעולות וסיפק תוצאות מדויקות יותר. עצם העבודה עם כל מה שקשור ל-Chatbot העמיק את יכולתנו להבין בצורה טובה וברורה יותר את הנתונים במסד הנתונים ואת פילוחם וניתוחם. |
| **שמירת ביצועים גבוהים בזמן עיבוד הנתונים** | כדי לשמור על ביצועים גבוהים בזמן עיבוד נתונים, קריאתם ממסד הנתונים, ניתוחם והצגתם על המסך בצורה ויזואלית (גרפים מסוגים וצורות שונות), ביצענו אופטימיזציה של הקוד שלנו באמצעות טכניקות טיפול יעילות בנתונים בכדי להפחית את העומס החישובי ולהבטיח זמני תגובה מהירים גם עם מערכי נתונים גדולים. |

**חלק ב' – Code Review:**  
המשוב שקיבלנו ב-Code Review: בקישור [כאן](https://docs.google.com/document/d/12m60FKgf96Q8yOgMywmpJ5AuQmmRNyNi/edit). **נרשום כאן את התגובות שלנו על ההערות:**

**בדיקות קבלה:**  
בדיקות הקבלה שהם ביצעו כוללות אלמנטים שלא בהכרח היו חלק מהדרישות של הפרויקט. על כן, לא ביצענו אותם בפרויקט שלנו. אולם, מאחר והרעיונות הללו נראים רלוונטיים וענייניים, שקלנו להוסיף אותם כחלק מהפרויקט שלנו אך במסגרת הזמן שהוקצב לנו לא היה אפשרי להוסיף אלמנטים מעבר למה שהיה נדרש.

מדובר כאן על חשיבה עיצובית, ולכל פרויקט כמובן שיהיו רעיונות/אלמנטים שונים. בפרט דברים שמתאימים לפרויקט אחד לא בהכרח יהיו מתאימים לפרויקט אחר.

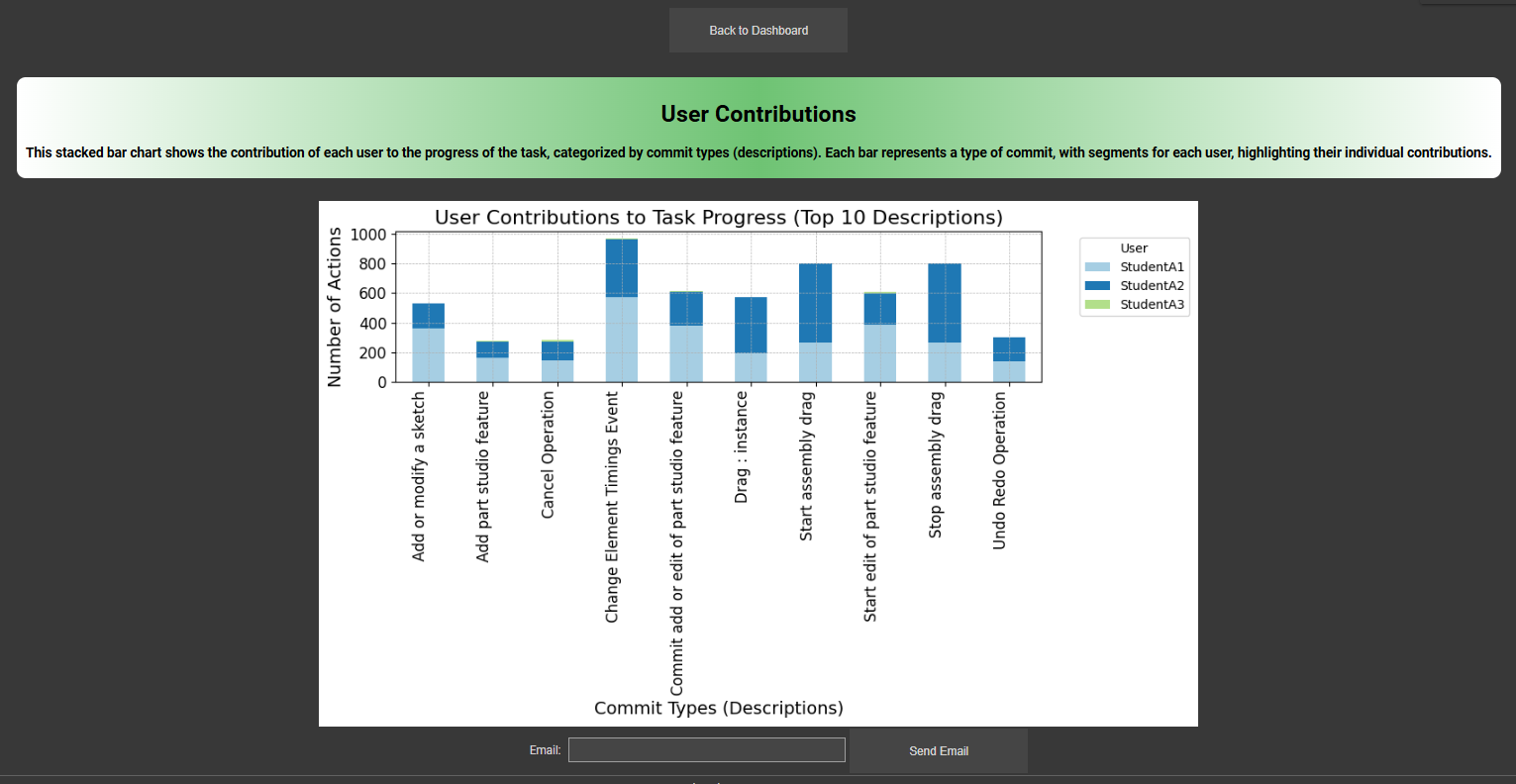
**קריטריונים:**  
יעילות: הוצע שניתן לשפר את יעילות הקוד על ידי שימוש במטמון, שאילתות יעילות יותר ועיבוד מקביל. זה מצביע על כך שאולי יש מקום לאופטימיזציה באופן שבו הקוד מטפל בעיבוד ואחזור נתונים. **אכן ביצענו בדיקה מקיפה לגבי איכות השאילתות שאנו שולחים למסד הנתונים.**

טיפול בשגיאות: הם ציינו שאין הודעה על שגיאות משתמש או שימוש במנגנון לטיפול בשגיאות. זה יכול להיות בעייתי מבחינת מתן חווית משתמש חלקה והבטחת שהמערכת מתנהגת באופן צפוי במקרה של קלט או שגיאות בלתי צפויות. **ביצענו בדיקה לכך שבמקרה של שגיאה אכן יש חיווי על המסך.** הוספנו עבור כל פעולה של קריאה ממסד הנתונים – חיווי על המסך עבור המשתמש במקרה של שגיאה. זה בא לידי ביטוי במצב בו יש צורך להציג גרף תוך לקיחת נתונים מהמסד והפעולה נתקלת בשגיאה. אנו מציגים הודעת פופ אפ למשתמש על כך שלא ניתן להציג את הגרף כעת.

**התייחסות להערות האישיות:**

|  |  |  |
| --- | --- | --- |
| **ת"ז** | **הערה שלהם** | **תגובה שלנו** |
| **314813825** | **סינון נתונים:** אין פילטרים לסינון נתונים אצלם, כך שאין אפשרות לבחור מספר סטודנטים מצומצם  **חלוקת הקוד לחלקים:** עדיף לחלק את הקוד ל-sections, הקוד יהיה מובן יותר ומחולק היטב. | **סינון נתונים:** כאמור, פיצ'ר זה אינו כלול כחלק מדרישות הפרויקט.  **חלוקה:** אנו חילקנו את הקוד במהלך העבודה ובסיומה איחדנו הכל למען שתהיה תוכנית שלמה. זאת התוצאה שהם ראו. |
| **322662065** | **דרישות יבוא ויצוא:** חסר במערכת עבודה ישירה של המשתמש עם המסד. **דרישות התראה:** קיימים הסברים לכל גרף מה המשמעות שלהם, ואפשר לחשוב על זה כהתראה למשתמש | **יבוא ויצוא:** לא הייתה דרישה לעבודה ישירה של המנהל עם מסד הנתונים. היה צורך לקרוא את קובץ ה-JSON אל תוך מסד הנתונים ולעבוד עם נתונים אלו בלבד. |
| **212970156** | **בדיקת פונקציונליות:** כל הפונקציות העיקריות של התוכנה פועלות כמצופה ומשיגות את היעדים והמטרות של התרגיל. **בדיקת אינטגרציה:** השילוב בין הפונקציות עובדות יחד כראוי. | - |
| **206468795** | **האם יש גישה למשתמשים בעלי מוגבלויות?** לא **האם המערכת נוחה לשימוש?** כן. מובנת, לא מסובכת. | **גישה למשתמשים בעלי מוגבלויות** הוא אכן פיצ'ר איכותי ורלוונטי אך במסגרת הזמן, כיוון שלא היה חלק מהדרישות, לא נכלל בפרויקט שלנו. |
| **209001981** | **האם המערכת ידידותית למשתמש?** המערכת מאוד קלה להבנה ולשימוש, והניווט בה אינטואיטיבי כך שאפשר להתחיל לעבוד במהירות ובקלות. **האם המידע מוצג בצורה נכונה ואמינה?** המידע מוצג בצורה ברורה ונכונה בהתאם לקובץ ה-JSON, והפונקציות מנתחות אותו . | - |

**8. שקיפות אלגוריתמית:**אנו מסבירים את האלגוריתם והנתונים שנאספו באמצעות הודעות קופצות (Pop-Up) ברורות ומפורטות. בנוסף, עבור כל אחד מהגרפים, אנו מספקים תיאור מקיף וברור על מה הגרף מציג ומה מטרתו.



בנוסף, הצ'אט בוט (העוזר הווירטואלי) מאפשר למשתמשים לבצע שאילתות במערכת באופן אינטראקטיבי, תוך מתן תשובות מותאמות אישית המבהירות את השימוש בנתונים ותובנות, מה שהופך את המידע לנגיש ומובן למשתמש.

תמונה שמכילה טקסט, צילום מסך, גופן

התיאור נוצר באופן אוטומטי

**9. משוב על הצגת הפרויקט:**

|  |  |  |
| --- | --- | --- |
| **הערת משוב** | **האם יש צורך בשינוי?** | **נימוק** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**10. מקורות שהשתמשנו בהם:**

- [Reading JSON File Using Python](https://www.geeksforgeeks.org/read-json-file-using-python/)

- [How to Use Google Colab](https://www.geeksforgeeks.org/how-to-use-google-colab/)

- [Firebase Fundamentals](https://firebase.google.com/docs/guides)

- [Python Plotting with Matplotlib](https://realpython.com/python-matplotlib-guide/)

- [Pandas Tutorial](https://www.w3schools.com/python/pandas/default.asp)

- Cloud Computing Course – Exercises about using Firebase, Pandas, Colab and Chatbot

- Chat GPT 4o Prompts – Some of the prompts that we have asked:

|  |
| --- |
| I'm analyzing an audit file for a project and would like to understand how to extract patterns related to progress and collaboration. Could you walk me through the process? |
| I’m reviewing acceptance tests for a project. Could you explain the criteria for creating effective tests and suggest some improvements for my current list? |
| I'm working with JSON data in Python and want to ensure I'm handling it efficiently. Could you explain how to parse and manipulate JSON data correctly? |
| I'm trying to fetch data from Firebase in Python and handle cases where the data might be missing. Can you explain the best practices for error handling in this context? |
| I’m working on integrating Firebase with my Python application. Can you walk me through how to establish a secure connection and query data efficiently? |
| I'm exploring how to display real-time data from Firebase in a Python dashboard. Could you explain the process of fetching and updating this data dynamically? |
| I'm working on a Python project that involves sending emails with plot attachments. Can you explain how to set this up securely using SMTP? |