**המחלקה להנדסת תוכנה - קורס מבוא למחשוב ענן**

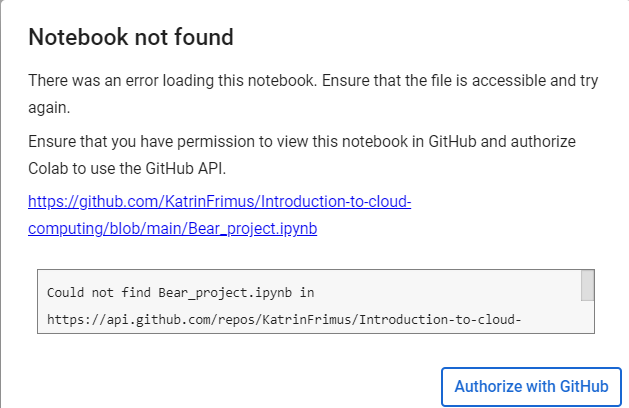
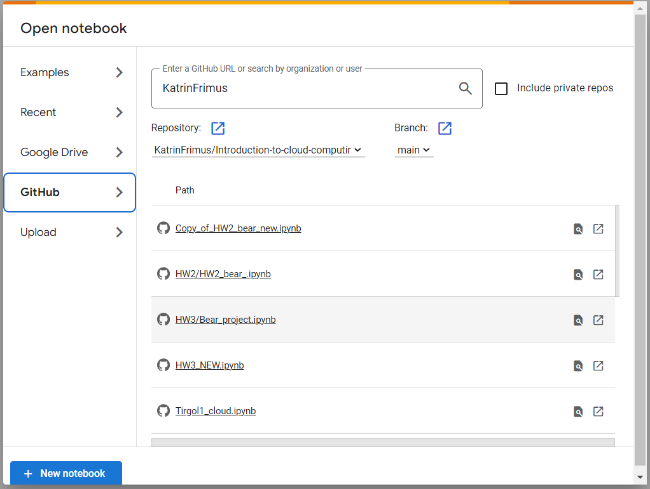
**סמסטר אביב התשפ"ד**

**תרגיל בית 3**- עבודה **בצוותי העבודה**

מועד הגשה: 15.8.24

המשימה בתרגיל זה: סיום בניית המערכת, ביצוע קוד ריוויו לצוות אחר - איטרצית פיתוח מס 2

הערה: הקוד לא תמיד מצליח להיפתח בגיט- לפעמים מגיעים לאפשרות של בחירת מחברת ואז צריך לבחור את המחברת HW3/bear\_project



קישורים:

גיט:

https://github.com/KatrinFrimus/Introduction-to-cloud-computing

Colab:

https://colab.research.google.com/drive/150S2V0nbB1XKU4fQTw7HN1hOUB89Vo4Z?usp=sharing

שימו לב: למטלה זו שלושה חלקים

**חלק ראשון.**(10 נקודות).עליכם להגדיר את בעלי התפקידים לאיטרציה זו (יש להחליף מתרגיל בית 2) יש לרשום מי מהנדס.ת המערכת, האחראי.ת בתרגיל זה.

מהנדס מערכת: תומר בן לולו

|  |  |  |
| --- | --- | --- |
| **שם חבר הצוות** | **משימות שהוקצו** | **משימות שהושלמו** |
| יותם גלעד | השלמת הגרפים החסרים  +מילואים | V |
| תומר בן לולו | השלמת הגרפים החסרים+ מילואים | V |
| אדר בודומסקי | צ'אטבוט+ תיק מתכנת | V |
| שובל בן שושן | צ'אטבוט+ תיק משתמש | V |
| קטרין פרימוס | מסך ראשי+ סרטון | V |
| ליטל חיבטוב | מסך ראשי+ תיקון הערות | V |

בדיקות קבלה:

1. גרפים- בדיקה שכל הגרפים עובדים, מציגים נתונים בצורה ברורה, כל הפילטרים מציגים נתונים נכונים
2. ייצוא PDF- עובד עבור כל גרף ועבור כל טבלה
3. קוד מסודר וברור לקריאה כולל תיעוד
4. צ'אט בוט מציג תשובות הגיוניות ונכונות ללא טעויות
5. מסך ראשי אינטואיטיבי המאפשר ניווט בין הפעולות השונות
6. כפתור Back עובד בכל מסך

***חלק שני: בניית המערכת*** (70 נקודות)

עליכם לסיים את בניית המערכת.

בפרט, יש לוודא קיום של כלל האלמנטים הבאים:

1. כל הפיצ'רים שהתחלתם לממש בתרגיל בית 2 - יש לוודא שכל המסכים תקינים ועובדים כראוי, **בקולאב**,ולעבוד עם firebase לצורך שמירת נתונים .אין להשאיר קישורים או כפתורים למסכים שלא קיימים.
2. צטבוט שישולב במערכת (כפי שלמדתם בתרגול 8), הכולל מידע אשר מגיע מה - DB שלכם.
3. התייחסות לאלמנטים רלוונטיים לקובץ ה -JSON שקיבלתם ( לפי מספר הקובץ, ולפי השאלות בנספח א)
4. 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.

|  |  |  |
| --- | --- | --- |
| הערה | התייחסות שלנו | האם בוצע שינוי? |
| יש באג שניתן להוציא PDF של גרף ריק | חסמנו את האפשרות, המשתמש מקבל הודעת שגיאה | כן |
| אותו באג גם בטבלה | כנ"ל | כן |

***חלק שלישי : סגירת הפרויקט \_( 20 נקודות)***

בחלק זה תכינו מסמכים המתארים את הפרויקט שלכם:

1. יש לבנות תיק למתכנת הכולל את שמות כל הקבצים המרכזיים, פונקציות מרכזיות, קטעי קוד/תבניות עיצוב מעניינים שהשתמשתם בהם.
2. יש לבנות תיק למשתמש , הכולל הסבר כללי על המערכת , פירוט מסכים, מעברים בין מסכים והסבר על טעויות אפשרויות.
3. יש להכין סרטון קצר של 30-60 שניות, המתאר את השימוש במערכת. הסרטון משמש כ – elevator pitch  למערכת שלכם, כלומר יש לכלול בו הסבר מקצועי ועם זאת שיווקי , המדגיש את האלמנטים המיוחדים של המערכת שבניתם. יש להגיש את הסרטון בפורמט mp4.

תיק מתכנת:

תיק מתכנת On-Shape Analayzer

שם הקבוצה: Bear

קבצים מרכזיים:

1. קובץ הקוד של הפרויקט שלנו (סיומת ipnyb), כל הפרויקט נכתב בgoogle colab
2. קובץ הג'ייסון הקבוצתי- קובץ מס' 6.

פונקציות מרכזיות בפרויקט:

1. index\_words(soup): מקבלת אובייקט מסוג BeautifulSoap ויוצרת אינדקס של מילים ותדירותן בטקסט.
2. remove\_stop\_words(index): פונקציה המקבלת את כל אינדקס המילים ומורידה ממנו את כל ה"stop words" שאלו בעצם מילים שהן "חסרות משמעות"
3. apply\_stemming(index): פונקציה המבצעת "פישוט" של המילים למצב הבסיסי שלהם וסופרת את תדירות המילה הבסיסית באינדקס
4. split\_time(record): לוקחת את השדה באובייקט שמכיל את המידע של קובץ הג'ייסון שהוא תאריך ושעה מחוברים ומפרידה אותו לשני שדות.
5. create\_graph\_pdfs(button): פונקציה היוצרת קובץ PDF של הגרף המוצג באתר ומורידה אותו למחשב.
6. create\_pdfs\_table(button): פונקציה היוצרת קובץ PDF של הטבלה המוצגת באתר ומורידה אותו למחשב.
7. build\_graph\_by\_num\_of\_operation(change): יוצרת גרף של מספר פעולות פר פילטר מסוים שהיא מקבלת.
8. display\_filtered\_table(change): פונקציה מציגה את נתוני קובץ הג'ייסון בצורת טבלה נוחה לעין לפי שדה סינון וערך סינון שבוחרים.
9. plot\_student\_hours(selected\_student): הפונקציה מציגה פילוח שעות עבודה פר סטודנט
10. activity\_hours\_per\_student(): הפונקציה מציגה גרף של פירוט הפעילויות של סטודנט מסוים לפי שעות.
11. overall\_activity\_by\_hours(): הפונקציה מציגה גרף של פעילויות לפי שעות עבור כלל הסטודנטים.
12. detailed\_categorize\_activity(description): פונקציה זו מקבלת תיאור של פעילות כפי שמפורט בקובץ הג'ייסון ומסווגת אותה לסוג פעילות ספציפי.
13. show\_widgets(b): מציגה את המידע הרלוונטי עבור כל כפתור שנלחץ בדף הבית.
14. reset\_interface(b): מאתחלת את האפליקציה למצב ההתחלתי שלה (דף הבית)
15. chat\_bot(): הפונקציה יוצרת צ'אט בוט בעל שאלות מובנות אשר המידע עליהן מוסק מהנתונים הנמצאים בקובץ הג'ייסון.

תבניות עיצוב הקיימות במערכת:

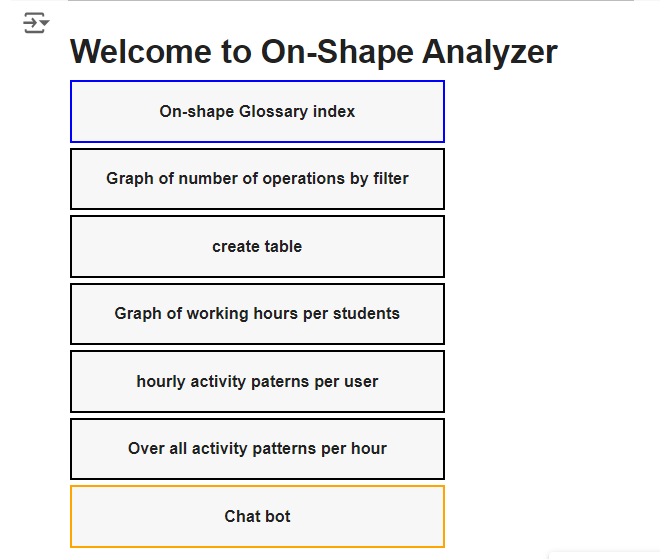
1. שימוש ב- ipywidgets: באמצעות שימוש בספרייה זו יצרנו במערכת שלנו ממשק משתמש אינטראקטיבי קל ונוח לתפעול המאפשר למשתמש לצפות בגרפים, טבלאות וכו'.
2. שימוש ב- Seaborn של ספריית matplotlib: נתן לנו אפשרות ליצור גרפים אסתטיים ואינפורמטיביים.
3. שימוש ב- NLTK: מאפשר יצירת צ'אט בוט היות והוא כלי לעיבוד שפה טבעית לניתוח טקסט.

תיק משתמש :  
תיק משתמש On-Shape Analayzer

שם הקבוצה: Bear

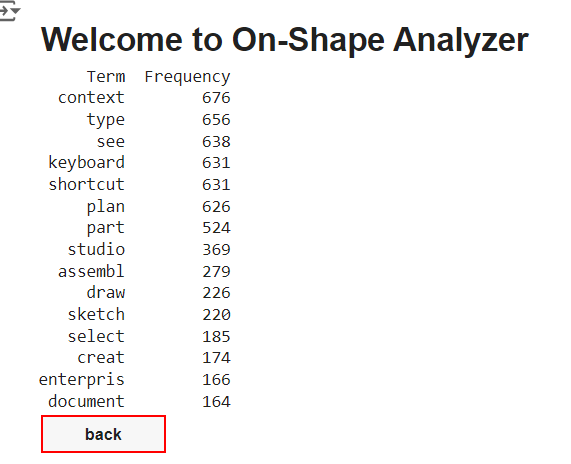
1. מסך הכניסה:

במסך הכניסה ניתן לבחור אחת מבין האפשרויות הנ"ל.



1. On-shape Glossary index:

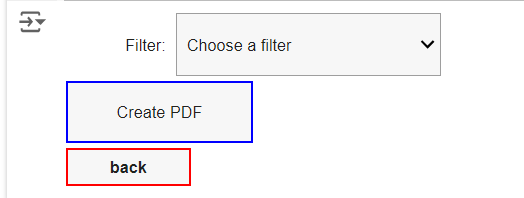
בלחיצה על הכפתור הנ"ל, ניתן לראות את כמות המילים שמופיעות הכי הרבה ואת תדירותן בדף ה Glossaryמהאתר של On-shape.

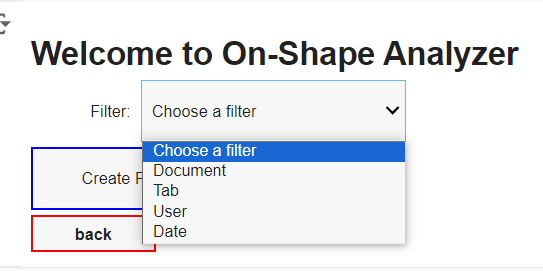


1. Graph of number of operations by filter:

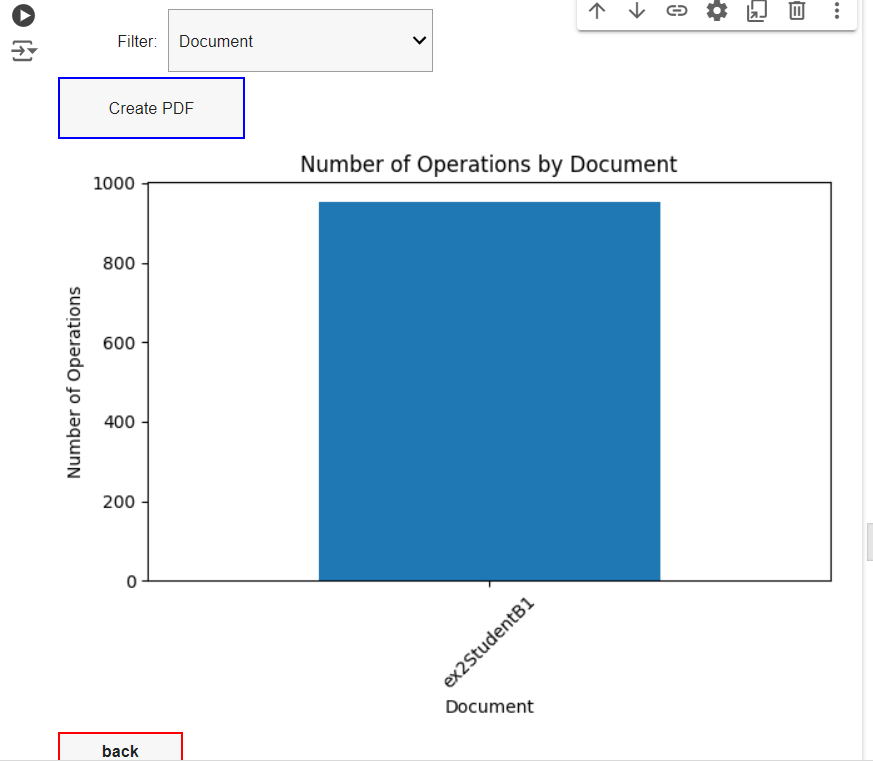
בלחיצה על הכפתור הנ"ל, נבחר את האופציה לקבלת גרף של מספר הפעולות מסונן ע"פ פילטר הניתן לבחירה.

בנוסף, קיים כפתור create pdf היוצר קובץ pdf של הגרף המבוקש ושומר אותו במחשב שלך.

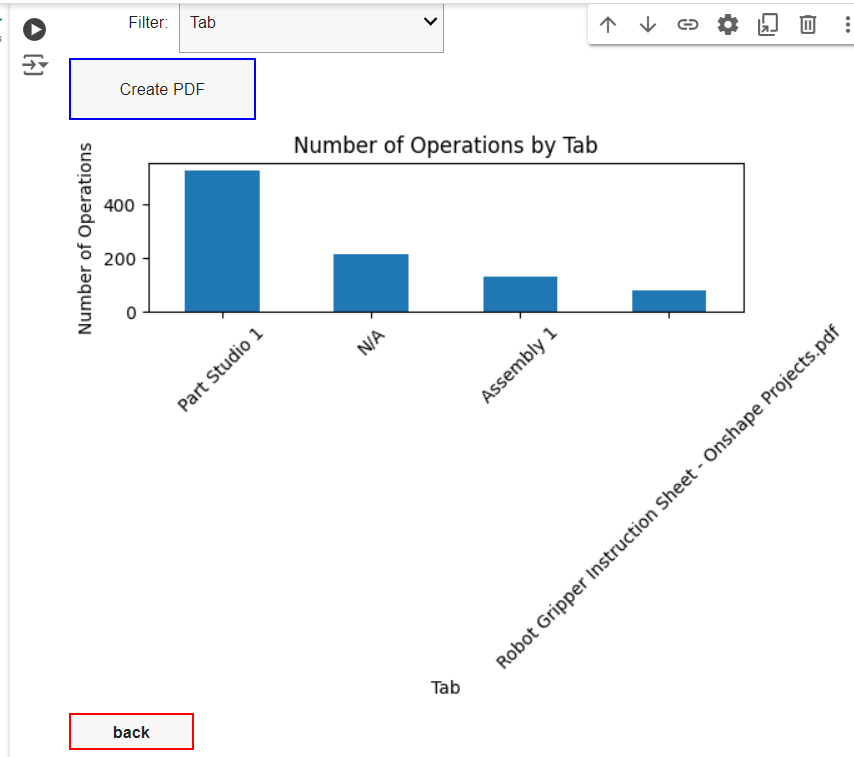




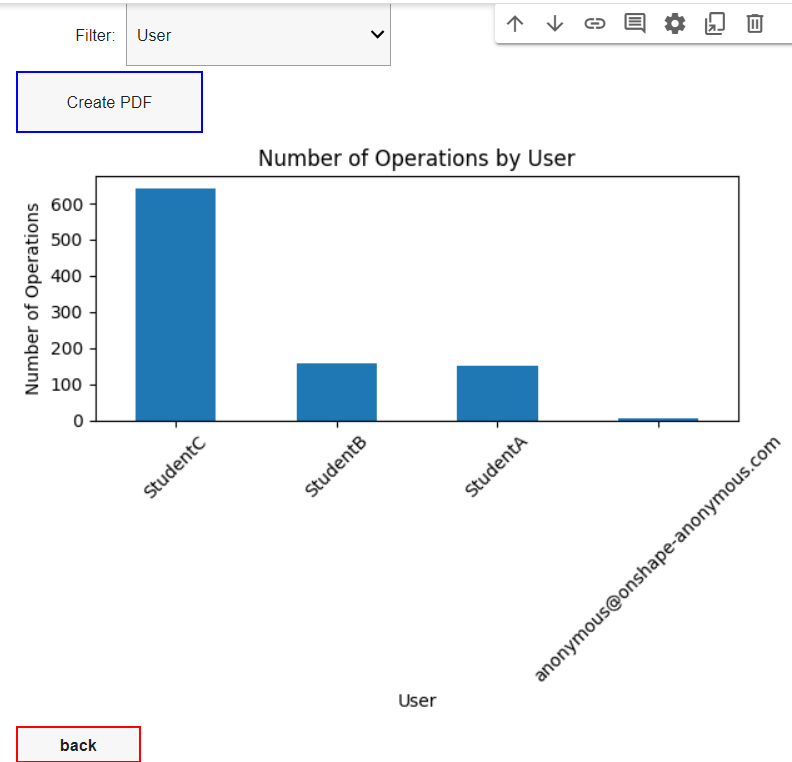
* 1. Document:

ע"י לחיצה על הכפתור הנ"ל, נקבל גרף המציג כמות פעולות שהתבצעו לפי מסמך מסוים.

* 1. Tab:

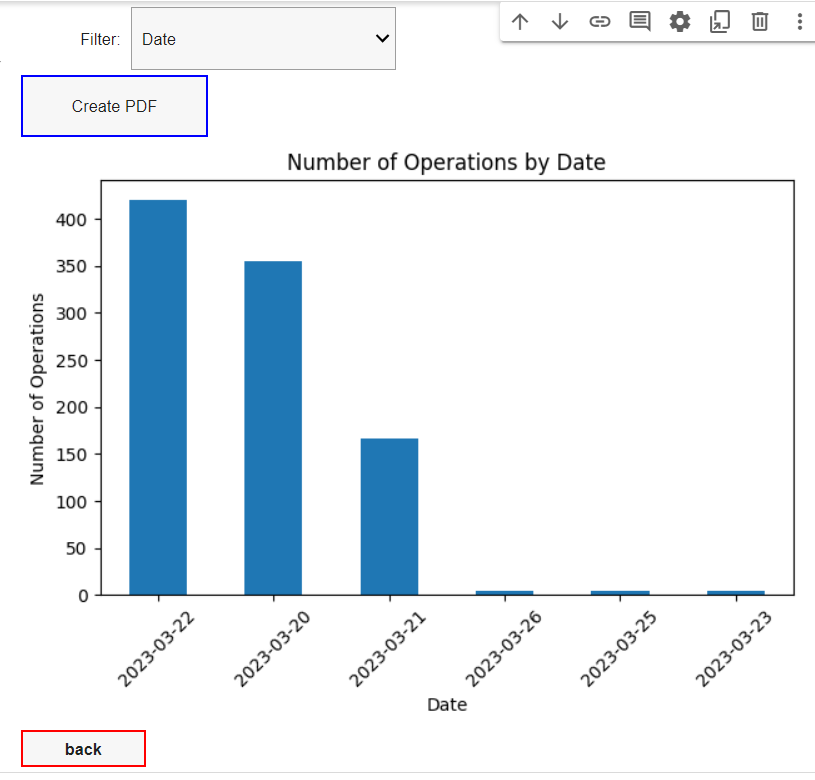
ע"י לחיצה על הכפתור הנ"ל, נקבל גרף המציג כמות פעולות שהתבצעו לפי tab מסוים.

* 1. User:

ע"י לחיצה על הכפתור הנ"ל, נקבל גרף המציג כמות פעולות שהתבצעו לפי משתמש מסוים.

* 1. Date:

ע"י לחיצה על הכפתור הנ"ל, נקבל גרף המציג כמות פעולות שהתבצעו לפי תאריך מסוים.

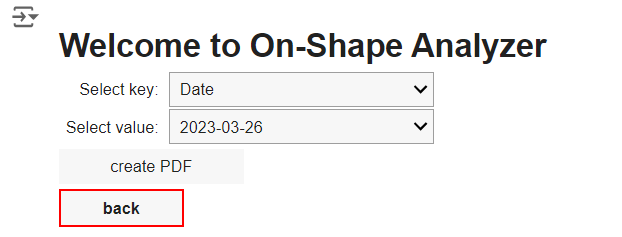


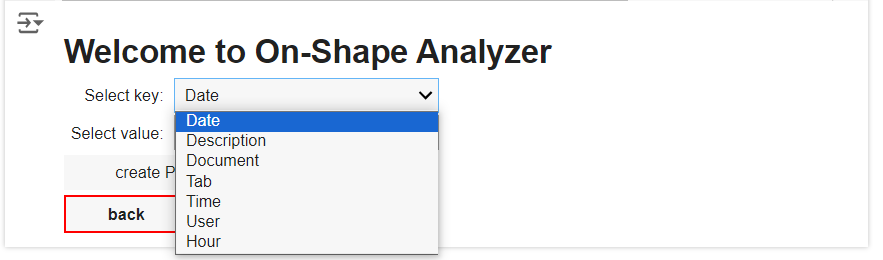
1. Create table:

בלחיצה על הכפתור הנ"ל, נבחר את האופציה לקבלת הערכים מפולחים בטבלה.

עליך לבחור key שע"פ נתוני הטבלה יופיעו, ולבחור value שיסנן את הערכים המבוקשים.

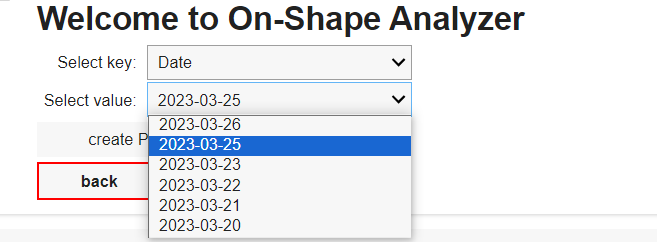
גם כאן, ניתן ליצור קובץ PDF ולשמור אותו במחשב.

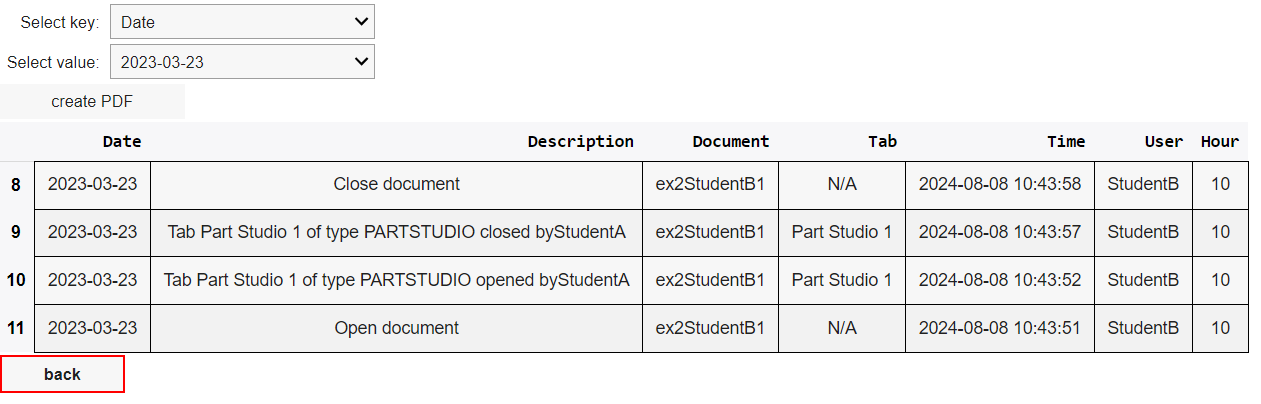




* 1. בחירת סינון ע"פ Date:

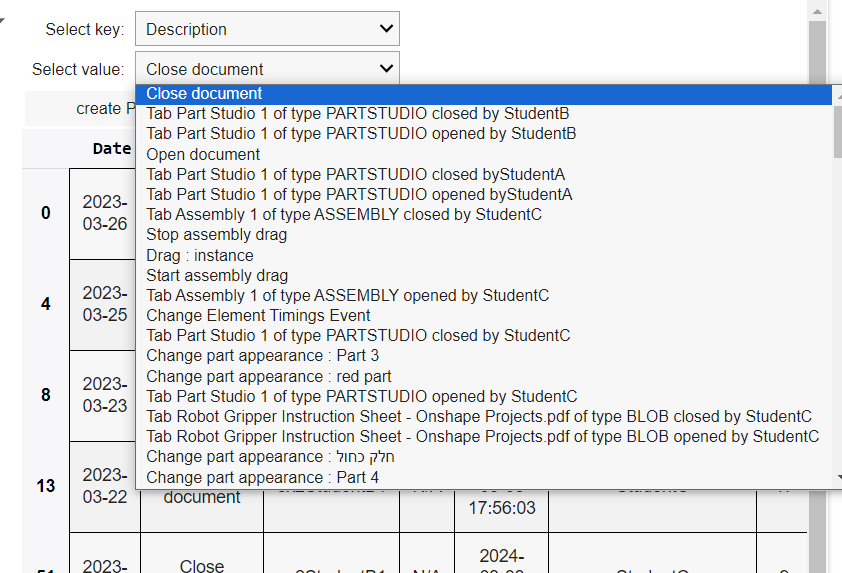
נצטרך לבחור תאריך שעליו נרצה לקבל את הנתונים בטבלה.

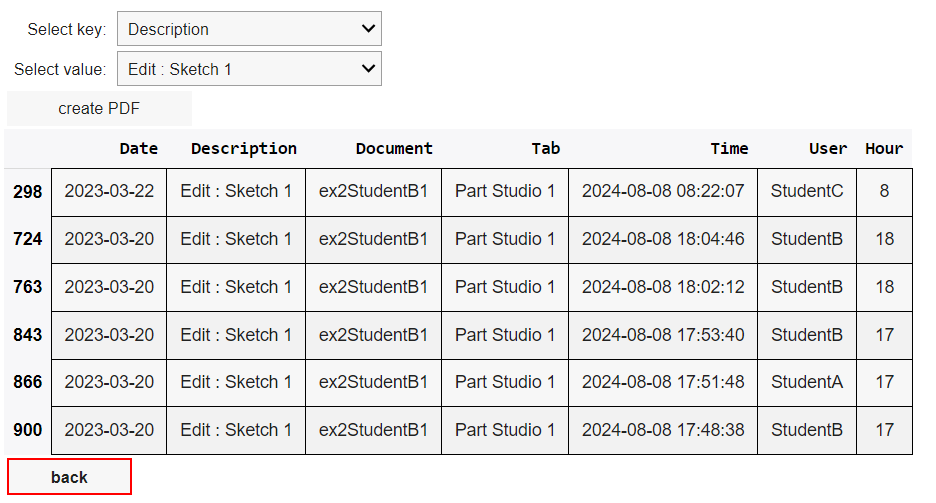




* 1. בחירת סינון ע"פ description:

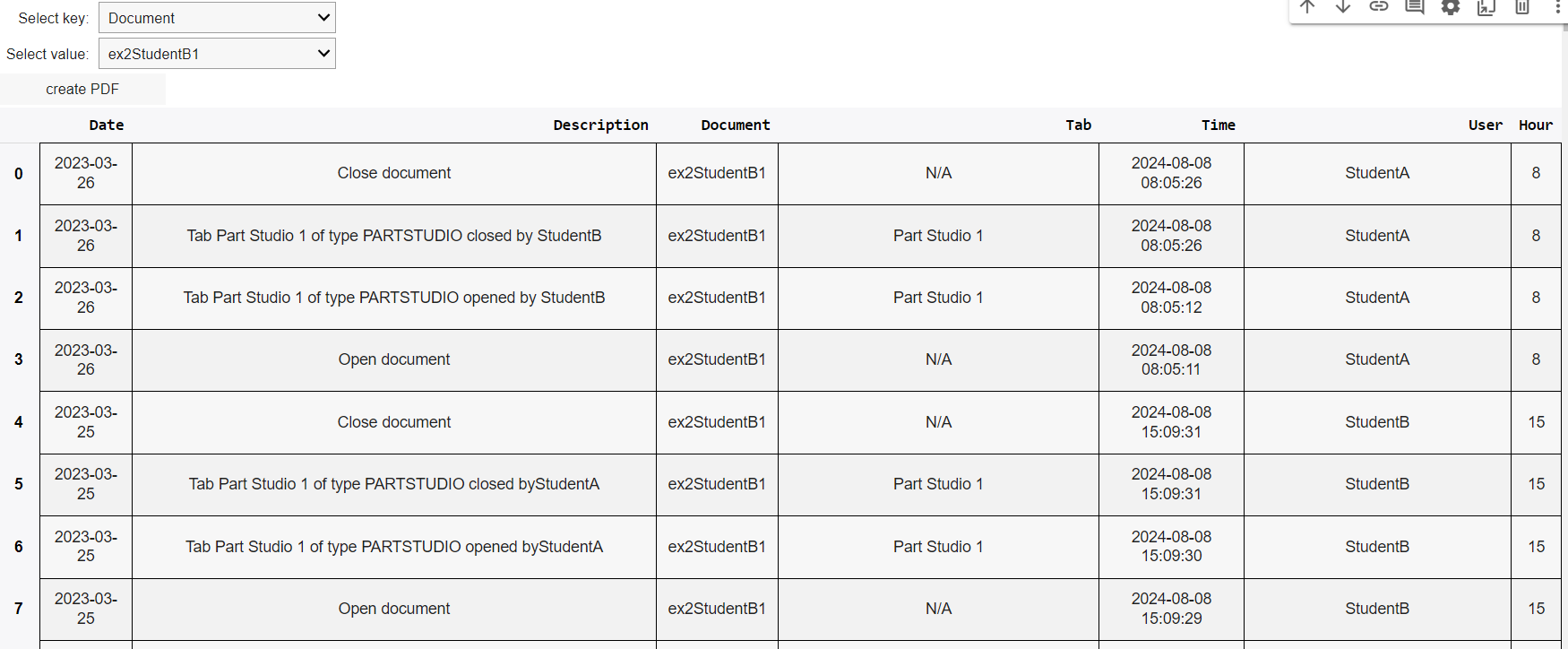
הסינון מתבצע לפי תיאור הפעולה שבה נבחר.





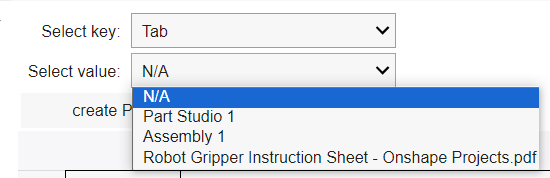
* 1. בחירת סינון ע"פ document:

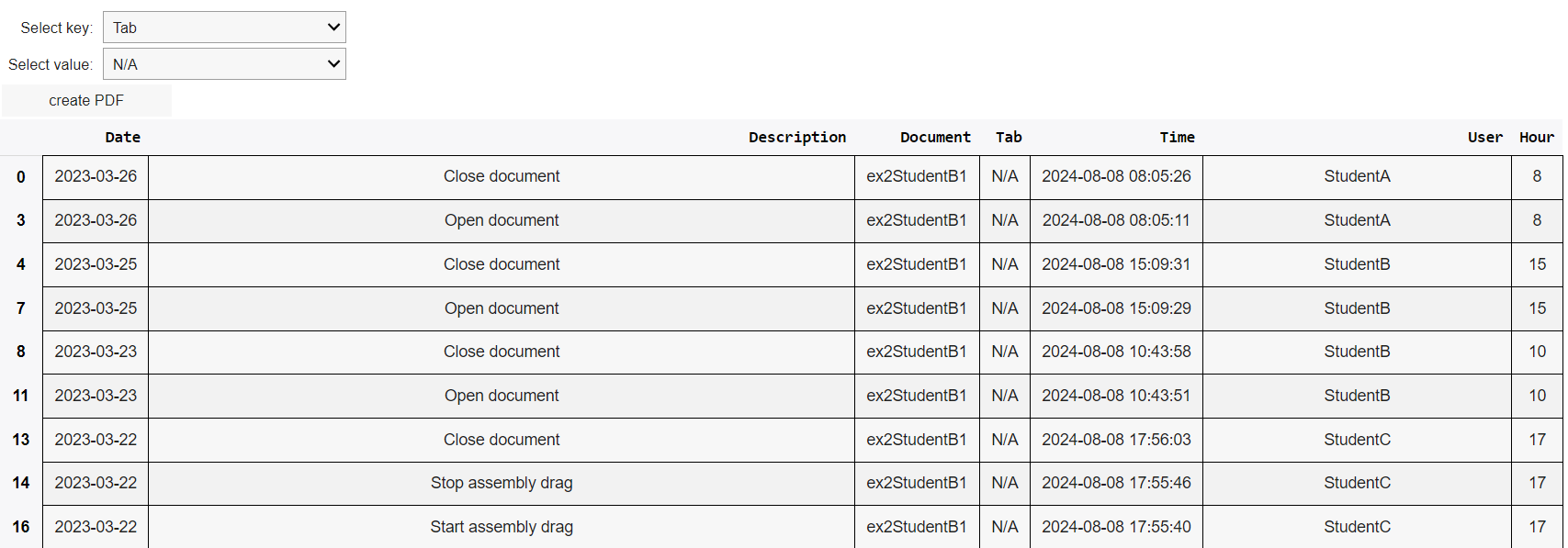
הסינון מתבצע לפי שם המסמך שנבחר.



* 1. בחירת סינון ע"פ tab:

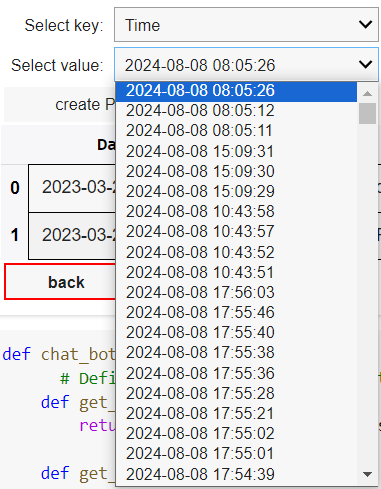
הסינון מתבצע לפי tab שנבחר.

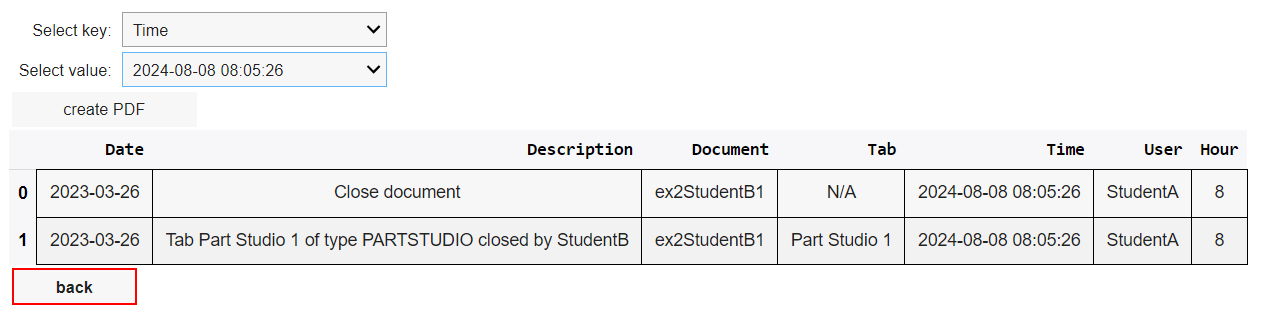




* 1. סינון לפי Time :

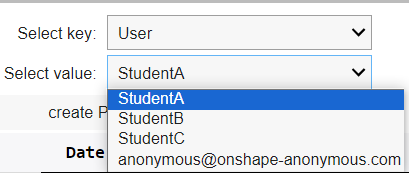
הסינון מתבצע ע"פ שעה ותאריך שנבחר.

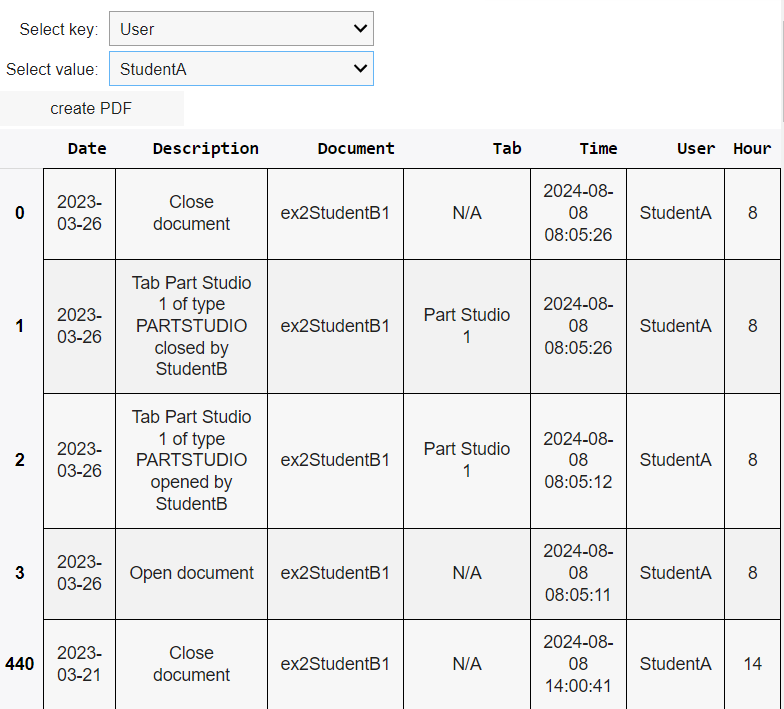




* 1. סינון ע"פ User:

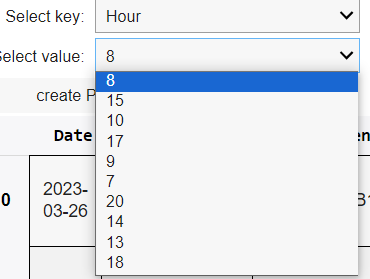
הסינון מתבצע ע"פ שם המשתמש שנבחר.

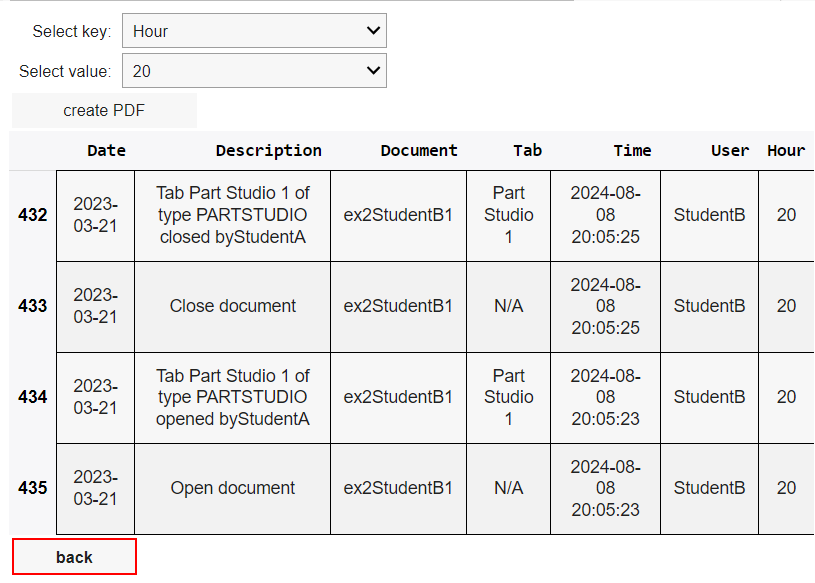




* 1. סינון ע"פ Hour:

הסינון מתבצע ע"פ השעה הנבחרת.

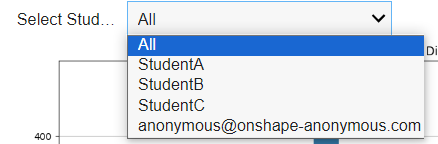


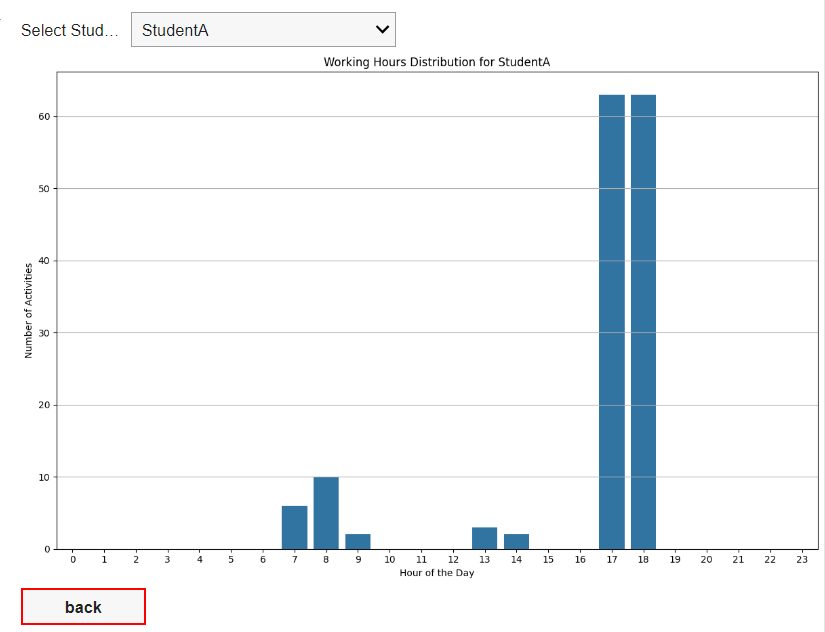


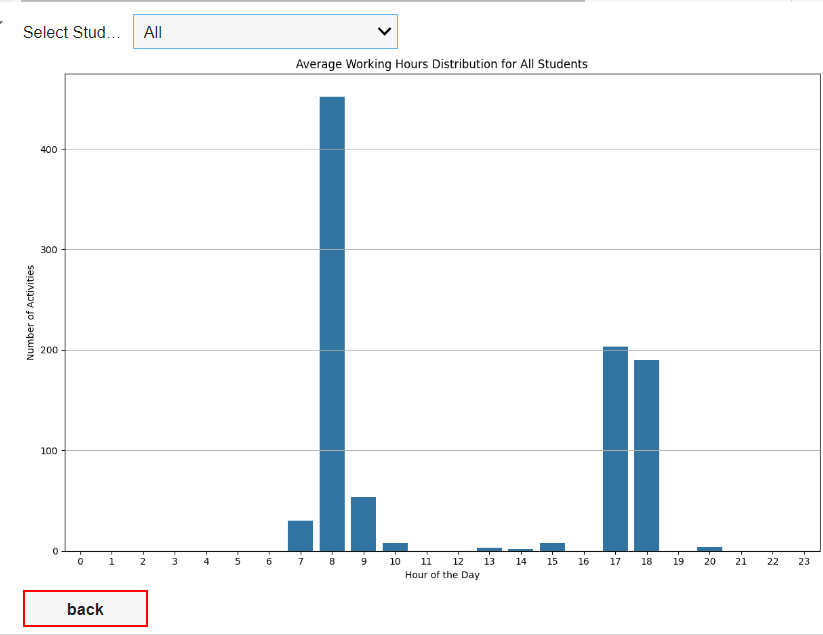
1. Graph of working hours per student:

בלחיצה על הכפתור נקבל גרף ע"פ שעות העבודה של כל סטודנט.

נוכל לבחור סטודנט ספציפי ולראות כמה פעולות ביצע בשעה כלשהי של היום, או לבחור את כל הסטודנטים ולראות ע"פ שעות העבודה שלהם את הממוצע של כמות הפעולות שביצעו.







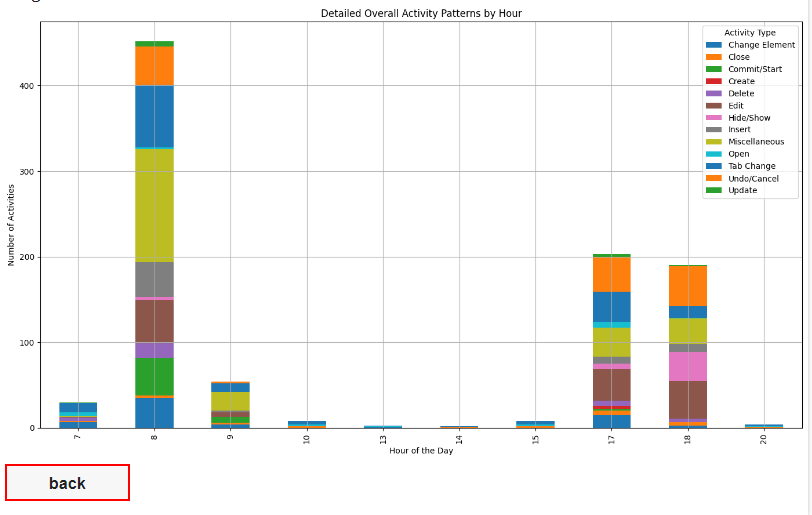
1. Hourly activity patterns per user:

בלחיצה על הכפתור נוכל לצפות במידע המציג כמה פעמים סטודנט מסוים ביצע פעולה מסוימת, בשעה מסוימת.



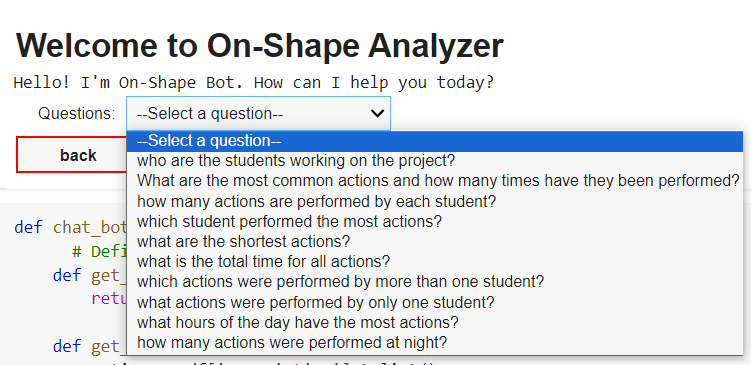
1. Over all activity patterns per hour:

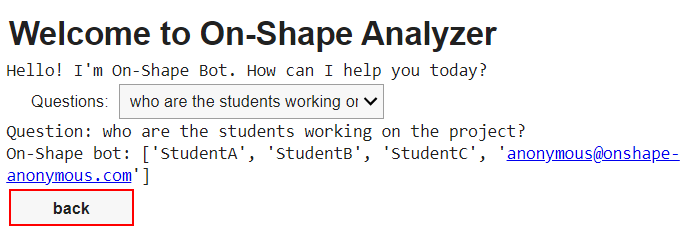
בלחיצה על הכפתור נוכל לצפות במידע המציג עבור שעה מסוימת, את כמות הפעולות שבוצעו, מחולק ע"פ נושאים.



1. Chat bot:

בלחיצה על הכפתור הנ"ל, נגיע לchat bot שאותו אפשר לשאול שאלות מובנות מראש, ולקבל עליהן תשובות, בהסתמך על המידע הקיים.





1. בכל שלב שאליו נתקדם מופיע הכפתור back, אשר מחזיר אותנו למסך הראשי.