**מבוא למחשוב ענן - סמסטר חורף התשפ"ה**

**תרגיל בית 2** -– **עבודה בצוותי העבודה**

מועד הגשה: 12.1.25

המשימה בתרגיל זה: בניית אינדקס למנוע החיפוש, ובניית מסכים מרכזיים במנוע החיפוש

שימו לב: למטלה זו שלושה חלקים

*חלק ראשון : בניית אינדקס (40 נקודות)*

יש למנות מהנדס.ת מערכת בכל צוות, אשר יהיה אחראי על הגדרת הדרישות ההנדסיות, ועל הממשק מול החומרה.

*בכל צוות על כל אחד לבחור אחד מהתפקידים הבאים (יש להחליף מתרגיל בית 1 ) (10 נקודות)*

*scrum master -מרכז את העבודה*

*,frontend developer – פיתוח החלק האחראי על הצגה ללקוח.*

*,backend developer – פיתוח מסד הנתונים והעבודה מולו.*

*product manager – ייצוג הלקוח בצוות (בהתאם לחשיבה העיצובית שבוצעה).*

*UI – עיצוב הממשק*

*QA – בדיקות התוכנה*

נא לרשום את שם הסטודנט.ית בתרגיל זה. על מהנדס.ת המערכת לכתוב כיצד נעשתה חלוקת העבודה מול הצוות, מה היו המשימות של כל חבר צוות, האם היה ממשק בין חברי הצוות, והאם המשימות מולאו:

תשובות לחלק 1:

מהנדס מערכת: יובל ג'יראד:  
חלקות העבודה מול הצוות נעשתה בפגישה הראשונה שתואמה על ידי הסקראם מסטר, ונוהלה בקובץ דוק אינטרנטי שהיה במעקב רציף של מהנדס המערכת, סקראם מאסטר וה-QA לצורך קידום העבודה.  
הפגישות נעשו בזום פעמיים בשבוע ותואמו על ידי הסקראם מאסטר כדי להתאים בלוז הכולל, בנוסף חברי הצוות שהתעסקו יותר בפרונט קיימו 2 פגישות פרונטליות לצורך קידום עבודה וכנל גם חברי הצוות שעבדו על הבק קיימו 2 פגישות פרונטליות לצורך קידום העבודה. טרם הסטודיו וההגשה בוצעה פגישה פרונטלית של כלל חברי הצוות לצורך הצגת כלל הצדדים של הקוד והסברים מפורטים כך שכל חבר צוות יכיר כל חלק בפרויקט.

|  |  |  |
| --- | --- | --- |
| מטלה 2 | | |
| **שם חבר הצוות ותפקיד בתרגיל זה** | **משימות שהוקצו** | **משימות שהושלמו** |
| רואי בר – Scrum master+Frontend | זימון פגישות, חלוקת משימות, מעקב אחר משימות,עזרה בעיצוב העמודים, עמוד אדמין, בניית דיאגרמה של סטטיסטיקות. | הכל :) |
| דימה - Backend | פיתוח זחלן, קישור צ'אטבוט ל-AI, בניית אינדקס. | הכל :) |
| יובל - Backend+Product | חיבור ל-DB, הפרדת מילים בזחלן, תעדוף אינדקס, יצירת דיאגרמה, וידוא של עמידה בדרישות ורצונות הלקוח, חשיבה על פיצ'ר מעניין עם הלקוח. | הכל :) |
| אלדר - Frontend+QA | בדיקת צ'אטבוט, בדיקת אכלוס יתר של DB, בדיקת הצגת סטטיסטיקות, עיצוב מסכים. | הכל :) |
| שי – Backend | קישור אינדקס והעלאת המידע ל-DB, שיפור בניית אינדקס, הפרדת פונקציות בזחלן כצורך לעתיד. | הכל :) |
| נדיר – UI | עיצוב ממשק נוח וקל לתפעול, חשיבה על זרימת התהליכים, יצירת אבטיפוס של מסכים לצורך בדיקת היתכנות לצוות הפרונט, חשיבה על עיצוב פיצ'ר מעניין. | הכל :) |
| בן - Frontend | בניית מסכי סטטיסטיקה מעניינת, עבודה על ויג'טים, תקשורת מול צ'אטבוט, עמוד חיפוש רגיל. | הכל :) |

בניית אינדקס (30 נקודות)

באיטרציה זו עליכם לבנות את מסד הנתונים שמכיל את האינדקס של המילים המשמעותיות באתר אותו אתם חוקרים.

מבנה האינדקס צריך להיות אחיד לכל הקבוצות , ולכלול **לפחות** את השדות הבאים  **( אין לשנות את שמות השדות!)**:

|  |  |
| --- | --- |
| שם השדה | הסבר |
| term | term |
| DocIDs | רשימת קישורים לדפים המכילים את ה- term, ממוספרים לפי בחירתכם |

ממשו את האינדקס בקולאב.

**קישור לתיקיית קוד של back + front של הפרויקט שלנו.**

**https://drive.google.com/drive/folders/10vClJ-0D3v1d4mIQYF84zlH2qBqXHyfV?usp=sharing**

*חלק שני: בניית מסכים להצגה בכיתה (50 נקודות)*

בחלק זה תכינו 4 מסכים, אותם תציגו לחבריכם בפעילות שתתבצע בכיתה.

המסכים צריכים לכלול (לפחות):

מסך מנהל לעריכת אינדקס, מסך הזנת שאילתא, מסך תוצאות שאילתא, מסך סטטיסטיקות מעניינות.

בשלב זה נדרש לממש במלואם את בניית מסך המנהל ומסך הזנת השאילתא. ניתן את התוצאות להציג כרגע עם data מהעמוד הראשי של האתר שאתם עובדים עימו.  
את מסך המנהל ומסך הסטטיסטיקות עליכם לממש בצורה בסיסית. עם זאת מומלץ לממש ככל הניתן גם את ניהול האינדקס וסריקת האתר, כפי שלמדתם בתרגול 6 😊

כמובן ,שכל תוספת שתחליטו עליה, תוביל להערכה גבוהה יותר של המשימה.

בשבוע ההרצאות של 6-9.1.25 תציגו את המערכות שבניתם. המפגש יתנהל במתכונת סטודיו – כל צוות מגיעה **במלואו** לאחד המועדים עם לפטופ, כל הצותים מציגים במקביל. הסטודנטים מסתובבים בין הצוותים, **מתנסים** במערכת,וממלאים משוב .

ההצגה ומילוי המשוב הם **חובה.**

אנא השתבצו בהקדם לאחת מקבוצות ההרצאה על מנת לוודא שקיים איזון בין הצוותים:

<https://docs.google.com/spreadsheets/d/11Lyni94IKg1O5Py2t3mIL1QhITCnJCMV_-MB_Ey73NE/edit?gid=0#gid=0>

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

תשובות לחלק 2:

1. יש להגיש את הטבלה הבאה , תוך התיחסות למשובים שקיבלתם (10 נקודות):

|  |  |  |
| --- | --- | --- |
| **הערת משוב** | **האם התבצע שינוי באפליקציה בעקבות ההערה?** | **נימוק** |
| ניכר פידבק חיובי מהצ'אטבוט ברמת הנגישות. צוות אחד אמר שישנה את הצ'אט בוט שלו שיתנהג כמו שלנו. | לא | נשמור את המבנה הפשוט של כפתור הסבר לכל מילת חיפוש של המשתמש בלי שינוי משמעותי. |
| קיבלנו פידבק חיובי על העיצוב הפשוט והנגיש של המערכת. | לא | נשמר את הפורמט העיצובי בהמשך פיתוח המערכת. |
| ניכר פידבק חיובי על תוכן הסטטיסטיקות שהצגנו ע"י הדפסות. | כן | עדכנו את פורמט הצגת הסטטיסטיקות אל גרפים ויזואליים. |
| קיבלנו פידבק שלילי על צורת ההצגה של הסטטיסטיקות. | כן | עדכנו את פורמט הצגת הסטטיסטיקות אל גרפים ויזואליים. |
| קיבלנו פידבק שלילי על עיצוב מינימלי של המערכת. | לא | בחרנו להשאר עם עיצוב מינימלי ובנוסף קיבלנו יותר פידבק חיובי מאשר שלילי על הבחירה העיצובית. להגשת הפרויקט הסופי, נוסיף ממשק מסודר, יפה יותר, מורחב יותר, אבל מינימליסטי. |

1. התייחסו ל -8 כללי הזהב של שניידרמן (הוצגו בתרגול). כיצד המערכת שלכם מבטאת אותם? (10 נקודות)

**תשובה:**

1. **שאיפה לעקביות:** המערכת שלנו פשוטה ומינימליסטית, ומציגה עקביות בעיצוב בעזרת שימוש ב-CSS. בנוסף סגנון הכרטיסיות והכפתורים הוא אותו דבר בכולם.
2. **שימושיות אוניברסלית:** במערכת שלנו יש תמיכה ב-2 סוגי משתמשים (מנהל ולקוח) שכוללים הרשאות שונות.
3. **משוב אינפורמטיבי:** המערכת מספקת למשתמשים שלנו תוצאות ברורות ופשוטות לכל מילה שיחפשו באתר, עם הודעת שגיאה ברגע שחיפוש לא הביא לתוצאה.
4. **סגירת דיאלוגים:** המערכת שלנו מאפשרת למשתמש לבצע תהליך חיפוש ברור ומאורגן הכולל התחברות > חיפוש מונח > הצגת תוצאות.
5. **מניעת שגיאות:** המערכת מציגה הודעת שגיאה ברורה ברגע שהתחברות לא התבצעה כשורה, או כאשר מונח כלשהו לא קיים במסד הנתונים.
6. **ביטול פעולות:** קיימת אפשרות לחזור אחורה באמצעות כפתור Back, וגם קיימת אפשרות לבטל עריכות.
7. **שליטת משתמש:** המשתמשים יכולים לנווט בחופשיות בין המסכים במערכת, לבצע מספר לא מוגבל של חיפושים, ומנהלים יכולים לערוך מידע בחופשיות.
8. **הפחתת עומס זכרון:** המערכת מציגה מידע רלוונטי באופן ויזואלי וברור, שכוללת גם סטטיסטיקות וגרפים, ושומרת על מידע חיוני גלוי במסך.
9. יש לרשום את ציון ה SUS של המערכת שלכם.מה מעיד הציון?(10 נקודות)

**תשובה:**

|  |  |
| --- | --- |
| **שאלה** | **ציון SUS משוקלל** |
| הייתי רוצה להשתמש במערכת זו לעיתים תכופות. | 2.86 |
| מצאתי כי המערכת מסובכת ללא סיבה | 2.59 |
| חשבתי שהמערכת קלה לשימוש | 3.25 |
| אזדקק לתמיכת איש טכני כדי שאוכל להשתמש במערכת זו | 3.62 |
| מצאתי כי הפונקציות השונות של המערכת היו מתואמות היטב | 3.21 |
| חשבתי כי היה יותר מידי חוסר עקביות במערכת זו | 3.17 |
| לדעתי רוב האנשים יהיו מסוגלים ללמוד להשתמש במערכת זו בקלות | 3.1 |
| מצאתי כי המערכת מאוד מסורבלת לשימוש | 3.24 |
| חשתי בטחון רב כאשר השתמשתי במערכת | 3.03 |
| עלי ללמוד הרבה דברים לפני שאוכל להשתמש במערכת זו. | 3.28 |
| **סה"כ לאחר כפל ב2.5** | **78.3** |

הציון שקיבלנו הוא מעל הממוצע העולמי שעומד על 68, ולכן אנחנו מסיקים שהמערכת שלנו היא ידידותית למשתמש, קלה ומסודרת, אבל בגלל שהציון לא עולה על 80 שהוא ציון מצוין למערכת, לדעתנו ניתן לשפר אותו באמצעות שיפור והוספת אלמנטים עיצוביים לממשק הסופי שאותו נגיש.

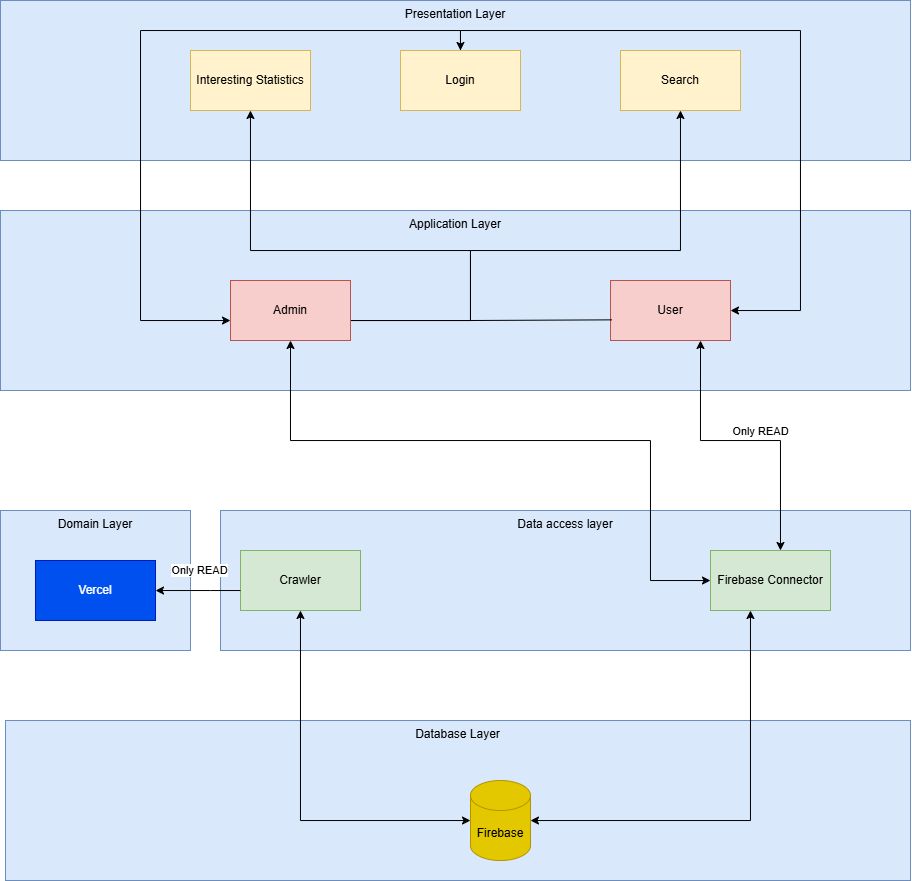
1. הגדירו 3 מדדים להצלחת המערכת (הרצאה 3).(10 נקודות)

**Latency** **זמן תגובה.**הזמן שלוקח מרגע שליחת שאילתת חיפוש ועד קבלת התוצאות  
יעד: זמן תגובה מתחת ל-3 שניות (בהנחה והמערכת רצה ב-google colab)

**Capacity יכולת הרחבה.**היכולת של המערכת לטפל בגידול בכמות הנתונים באינדקס  
יעד: עמידה בעלייה בכמות המידע באתר מול יכולת של המערכת לעבד את הנתונים.

**Security אבטחת מידע.**היכולת לאבטח את המידע במערכת ע"י שימוש במסד נתונים מאובטח.  
יעד: שימוש במסד נתונים מוכר ונאמן.  
מימוש: שימוש ב-firebase.

1. הציגו דיאגרמת ארכיטקטורה של המערכת שלכם. הסבירו באיזה סוג ארכיטקטורה השתמשתם (הרצאה 7), ופרטו את חלקי הקוד ההמתיחסים לכל חלק בארכיטקטורה.(10 נקודות)

****

**המונחים והקוד:**

**Back-end:**

**Data Access Layer: Crawler:** implemented in VercelCrawler\_main\_page.py

Crawling in the page: crawl()

Preprocessing the test and words: preprocess\_text()

Counting the number of appearances and creating a dictionary: create\_count\_dictionary()

**Front-end:**

**Search:** create\_search\_action()

**Login:** login\_action(), on\_login\_click()

**Interesting Statistics:** create\_show\_statistics()

**Application Layer: Admin:** update\_data(), delete\_data()

**Data Access Layer: Firebase Connector:** fetch\_data(), fetch\_statistics(), update\_data(), delete\_data()

**Database Layer:**

Communication established using url:  
FIREBASE\_URL = "<https://vercelcrawler-1c167-default-rtdb.firebaseio.com/>"  
FBconn = firebase.FirebaseApplication(FIREBASE\_URL,None)

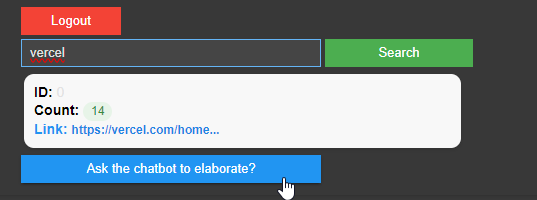
*חלק שלישי : פיצ'ר לבחירתכם* (10 נקודות)

הוסיפו פיצ'ר מעניין למערכת, אשר לא נדרש מכם, לבחירתכם.שימו לב - הכוונה לפיצ'ר פונקציונאלי (ולא עיצובי, או שימוש בשרת כדי להעלות לענן את האתר).

כתבו מספר משפטים להסבר התוספת, וציינו היכן בקוד הוא ממומש. כמו כן הסבירו כיצד הוא מתבטא בחלק המוצג למנהל.

**תשובה:**

הוספנו צ'אטבוט שיודע לקחת את המילה שאותה חיפש המשתמש.



כאשר המשתמש לוחץ על הכפתור, הבוט יודע להציג תקציר מידע רלוונטי בנושא.

