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

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

מועד הגשה: 28.5.25

**Turtels team:**

עינב בן שושן – 315064881

יובל לרפלד – 318186897

מתן עמרן - 316326313

עמיר אזולאי – 206695488

חן צפיר – 319005104

קישור לתיקיית הגיט:  
  
<https://github.com/einavbs1/HW2_Cloud>

קישור למחברת הColab:

<https://colab.research.google.com/drive/19LwUblV3ieF9Z7B7kHDpSVwKlfzCNfio?usp=drive_open#scrollTo=97GiVVIAuS0i>

*חלק ראשון : בניית מערכת (10 נקודות)*

***תיאום עבודה ושיתוף פעולה:*** *הצוות עבד בצורה מסודרת ומסונכרנת על פי הגדרות התפקידים. הייתה עבודה צמודה בין צוותי ה-Frontend וה-Backend, במיוחד בממשק המתקבל בין הממשק הגרפי למסד הנתונים. עינב (Frontend) ועמיר (Backend) היו בקשר רציף על מנת להבטיח שהנתונים המוצגים במסך שאילתה יהיו מעודכנים ומדויקים. כמו כן, יובל (UI) שיתף פעולה עם צוות ה-Frontend וה-Product Manager לוודא שהממשק הגרפי מותאם לצרכים של הלקוח.*

***האתגרים בתהליך:*** *במהלך העבודה, זוהו בעיות קטנות בהתחברות בין ה-Frontend ל-Backend, אך בעיות אלו טופלו תוך שימוש בעבודה צוותית ושיתוף פעולה הדוק, והבעיה נפתרה במהרה.*

| **איטרציה 1** | | |
| --- | --- | --- |
| **שם חבר הצוות ותפקיד בתרגיל זה** | **משימות שהוקצו** | **משימות שהושלמו** |
| מתן עמרן - *product manager* | מעבר על התוצרים,תכנון לו"ז, בניית האינדקס, הוספת שיפור ביצועים בניהול מסד הנתונים, קיום פגישות סדירות עם הצוות לצורך סקירה ובדיקת התקדמות.  יצירת תיעוד מלא של הדרישות למערכת. | הכל |
| עינב בן שושן  *frontend developer* | עיצוב המסכים, מסך שאילתה למנוע חיפוש, חיבור הקוד המלא לתוצר סופי, יצירת ממשק משתמש נגיש לכלל המשתמשים.  עבודה עם צוות ה-UI לוודא התאמה בין העיצוב לבין הקוד. | הכל |
| עמיר אזולאי - *backend developer* | מסך משחק, מסך מנהל המערכת, עיצוב מנגנונים שיבטיחו אינטגרציה חלקה ויעילה של הנתונים בין החלקים השונים של המערכת, שמירה על קשר רציף עם צוות ה-Frontend, ה-QA, וה-UI להבטחת עבודה משולבת ונכונה. | הכל |
| חן צפיר - *scrum master, QA* | מסך סטטיסטיקות, הכנת פייק דאטה, הרצת בדיקות למנוע חיפוש, בדיקות לקוד המלא , ארגון סדנאות ובדיקות עם הצוות לשיפור תהליך העבודה,  ניהול תהליך התקני בדיקה ושחזור תקלות. | הכל |
| יובל לרפלד - *UI* | פיצ'ר,לוגו, מסך תוצאות חיפוש, עבודה צמודה עם צוות ה-Frontend וה-Product Manager להבטחת חוויית משתמש עקבית. | הכל |

בניית אינדקס (20 נקודות)

**רשימת ה stop words -**

1. מילות עצירה כלליות באנגלית, מאוד נפוצות בשפה, נלוות ולא מוסיפות משמעות חשובה לטקסט כך שאין להן ייחודיות.
2. מילים שנפוצות באתרי אינטרנט, ללא ייחודיות מסויימת (לרוב יימצאו בהדר או בפוטר).

| could  couldn't  course  currently  d  definitely  described  despite  did  didn't  different  do  does  doesn't  doing  don't  done  down  downwards | contain  containing  contains  corresponding  could  couldn't  course  currently  d  definitely  described  despite  did  didn't  different  do  does  cannot  contains  corresponding | c'mon  c's  came  can  can't  cant  cause  causes  certain  certainly  changes  clearly  co  com  come  comes  concerning  consequently  consider | becomes  becoming  been  before  beforehand  behind  being  believe  below  beside  besides  best  better  between  beyond  both  brief  but  by  c | appear  appreciate  appropriate  are  aren't  around  as  aside  ask  asking  associated  at  available  away  awfully  b  be  became  because  become | along  already  also  although  always  am  among  amongst  an  and  another  any  anybody  anyhow  anyone  anything  anyway  anyways  anywhere  apart | a  a's  able  about  above  according  accordingly  across  actually  after  afterwards  again  against  ain't  all  allow  allows  almost  alone |
| --- | --- | --- | --- | --- | --- | --- |

| nothing  novel  now  nowhere  o  obviously  of  off  often  oh  ok  okay  old  on  once  one  ones  only  onto  or  other  others  otherwise  ought  our  ours  ourselves  out  outside  over  overall | moreover  most  mostly  much  must  my  myself  n  name  namely  nd  near  nearly  necessary  need  needs  neither  never  nevertheless  new  next  nine  no  nobody  non  none  noone  nor  normally  not | known  l  last  lately  later  latter  latterly  least  less  lest  let  let's  like  liked  likely  little  look  looking  looks  ltd  m  mainly  many  may  maybe  me  mean  meanwhile  merely  might  more | if  ignored  immediate  in  inasmuch  inc  indeed  indicate  indicated  indicates  inner  insofar  instead  into  inward  is  isn't  it  it'd  it'll  it's  its  itself  j  just  k  keep  keeps  kept  know  knows | haven't  having  he  he's  hello  help  hence  her  here  here's  hereafter  hereby  herein  hereupon  hers  herself  hi  him  himself  his  hither  hopefully  how  howbeit  however  i  i'd  i'll  i'm  i've  ie | follows  for  former  formerly  forth  four  from  further  furthermore  g  get  gets  getting  given  gives  go  goes  going  gone  got  gotten  greetings  h  had  hadn't  happens  hardly  has  hasn't  have | else  elsewhere  enough  entirely  especially  et  etc  even  ever  every  everybody  everyone  everything  everywhere  ex  exactly  example  except  f  far  few  fifth  first  five  followed  following |
| --- | --- | --- | --- | --- | --- | --- |

| will  willing  wish  with  within  without  won't  wonder  would  would  wouldn't  x  y  yes  yet  you  you'd  you'll  you're  you've  your  yours  yourself  yourselves  z  zero | were  weren't  what  what's  whatever  when  whence  whenever  where  where's  whereafter  whereas  whereby  wherein  whereupon  wherever  whether  which  while  whither  who  who's  whoever  whole  whom  whose  why | us  use  used  useful  uses  using  usually  uucp  v  value  various  very  via  viz  vs  w  want  wants  was  wasn't  way  we  we'd  we'll  we're  we've  welcome  well  went | though  three  through  throughout  thru  thus  to  together  too  took  toward  towards  tried  tries  truly  try  trying  twice  two  u  un  under  unfortunately  unless  unlikely  until  unto  up  upon | that  that's  thats  the  their  theirs  them  themselves  then  thence  there  there's  thereafter  thereby  therefore  therein  theres  thereupon  these  they  they'd  they'll  they're  they've  think  third  this  thorough  thoroughly  those | some  somebody  somehow  someone  something  sometime  sometimes  somewhat  somewhere  soon  sorry  specified  specify  specifying  still  sub  such  sup  sure  t  t's  take  taken  tell  tends  th  than  thank  thanks  thanx | said  same  saw  say  saying  says  second  secondly  see  seeing  seem  seemed  seeming  seems  seen  self  selves  sensible  sent  serious  seriously  seven  several  shall  she  should  shouldn't  since  six  so | own  p  particular  particularly  per  perhaps  placed  please  plus  possible  presumably  probably  provides  q  que  quite  qv  r  rather  rd  re  really  reasonably  regarding  regardless  regards  relatively  respectively  right  s |
| --- | --- | --- | --- | --- | --- | --- | --- |

3. **לצורך בניית האינדקס בחרנו להשתמש בstem -** כדי לרכז כמה מילים תחת אותו key באינדקס, כאשר כמובן הכוונה שלהם דומה מאוד, במיוחד באתר כמו Mqtt. הריכוזיות של stem היא בעיננו היתרון שלו.

**בנוסף בתחתית העמודים מחוץ למסגרת הטאבים שלנו יש שאילתות על מנת לבצע הרצות ולראות כיצד המנוע עובד ואילו תוצאות הוא מחזיר.**

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

1. כיצד המערכת שלכם מבטאת אותם?

**עקביות -** תשתית העיצוב במערכת שלנו זהה בכל העמודים, רקע לבן וחלק וסגנון פשוט ומודרני. כפתור החיפוש נמצא במקום מוכר למשתמש ממנועי חיפוש מפורסמים. אלמנטים נוספים כמו קישור לחיץ, אייקון רענון מופיעים בצורה הנפוצה על מנת לחתור לעקביות.

**קיצורי דרך -** קיימים קיצורי דרך הנפוצים בדפדפנים - העתק, הדבק, גזירה, undo, redo  
  
**משוב אינפורמטיבי -** בעמוד המשחקיות יש שינוי בניקוד לכל פעולה של מהנדס ועדכון בטבלה. בנוסף בחיפוש מתקבלת הודעה אם המילה לא קיימת.  
  
**עיצוב דיאלוגים ברורים -** כאשר מנסים להתחבר או כאשר מבצעים חיפוש מקבלים תגובה ברורה האם הצליח - מופיעים התוצאות, ואם לא מתקבלת הודעה בצבע אדום.

**מנע שגיאות -** בדף התחברות או החיפוש, אם פעולה לא מתבצעת כשורה - בתגובה לכך תופיע הודעה המסבירה את מהות הבעיה.

**היפוך קל של פעולות -** פעולות (Undo), הן מוגבלות בקולאב מלכתחילה.

**מוקד השליטה אצל המשתמש -** המשתמש שולט במערכת, יכול לעבור בין עמודות, ולבצע פעולות כרצונו.

**הפחתת עומס זיכרון לזמן קצר -** כל האפשרויות ברורות, והמשתמש לא צריך לזכור מידע, הכל מוצג על המסך.

1. טבלת התייחסות למשובים:

| הערת משוב | האם התבצע שינוי באפליקציה בעקבות ההערה? | נימוק |
| --- | --- | --- |
| להוסיף עוד סטטיסטיקות וגרפים | כן | הוספנו גרפים נוספים שיוכלו לתת מידע חשוב על המתרחש במעבדה. |
| להוסיף מסך משחק | כן | הוספנו משחק משחקיות של העובדים על מנת שיוכלו להשלים משימות וזה ידרבן אותם. |
| לחבר את הסנסורים | לא | לא ניתן כרגע לחבר את הסנסורים אך השתמשנו בדאטה ששלפנו. |
| מסך עובדים קצת יותר מעניין | כן | מתקשר למשחקיות שהוספנו. |
| פונקציונליות נוספת | כן | הוספת אפשרות לראות את האינדקס דרך הדף מנהל ( כמובן שבעתיד מתוכנן לשלוט שם על הDB כלומר הוספת אינדקס, הוספת משימות, הסרת משימות, שינוי ניקוד לעובדים, הוספת קישורים לאינדקס קיים, הסרת מילים וכו). |
| שיפור העיצוב | לא | בחרנו בעיצוב מינימליסטי תוך מתן דגש על הלוגו והעיצוב הכולל. |
| להוסיף כותרת ותיאור לתוצאות | כן | הוספנו לכל גרף הסבר כך שיהיה מובן מה הגרף מציג. |
| סידור הגרפים | כן | מתקשר להוספת כותרת ותיאור לתוצאות. בוצע. |

* חשוב לציין כי לא סיימנו את העבודה.
  + **במסך האדמין: אנו מתכננים לבצע הוספה והסרה של קישורים מאינדקס מסויים, וכן הוספה והסרה של אינדקס, אופציה לערוך את ניקוד השחקנים, לראות את העובדים שהיו מחוברים לאחרונה. אנחנו בוחנים אופציה על שמירת המילים מתוך האינדקס שחיפשו הכי הרבה ככה שאולי נדע שנרצה להרחיב את המקורות הידע שלנו בנושאים אלה.**
  + **במסך המשחקיות: נבצע התחברות עם שם משתמש וסיסמא על מנת שמשתמש לא יוכל לבצע משימות על חשבון משתמש אחר, וכן הוספת משחק טריוויה על תקלות נפוצות במפעל.**

**כרגע זה התכנון ואנו פתוחים לרעיונות נוספים על מנת לשפר את המערכת שלנו ולייעל את התנהלות העובדים עם המערכת.**

1. **שאלון שביעות רצון מהמערכת:**

| **מס"ד** | **שאלה** | **מסכים בהחלט** |  | | | **לא מסכים בכלל** |
| --- | --- | --- | --- | --- | --- | --- |
| **5** | **4** | **3** | **2** | **1** |
| 1 | נראה לי שהייתי מעוניין/נת להשתמש במערכת זו לעיתים קרובות. | X |  |  |  |  |
| 2 | לדעתי המערכת מורכבת מידי שלא לצורך. |  |  |  | X |  |
| 3 | מצאתי את המערכת נוחה לשימוש. | X |  |  |  |  |
| 4 | נראה לי שאזדקק לעזרתו של איש טכני, כדי שאוכל להשתמש במערכת. |  |  |  | X |  |
| 5 | נראה לי שהפונקציות השונות של המערכת שולבו היטב ביחד. | X |  |  |  |  |
| 6 | נתקלתי במקרים רבים של חוסר עקביות לאורך השימוש במערכת. |  |  |  |  | X |
| 7 | נראה לי שרוב האנשים יכולים ללמוד להשתמש במערכת במהירות. |  | X |  |  |  |
| 8 | לדעתי המערכת הינה מסורבלת לשימוש. |  |  |  |  | X |
| 9 | הרגשתי מאד בטוח באופן השימוש שלי במערכת. | X |  |  |  |  |
| 10 | הייתי צריך ללמוד הרבה לפני שהצלחתי להשתמש במערכת. |  |  |  |  | X |

סכום הניקוד:  
 4 + 3 + 4 + 3 + 4 + 4 + 3 + 4 + 4 + 4 = **37**

ציון SUS סופי = 37 × 2.5 = **92.5**

**ציון ה-SUS של המערכת עמד על 92.5 – ציון המעיד על רמת שמישות מעולה.**

ציון זה משקף חוויית משתמש חיובית במיוחד, עם ממשק אינטואיטיבי, ברור ונוח לשימוש. המשתמשים הרגישו ביטחון ושליטה בפעולתם, ולרובם לא נדרשה עזרה טכנית כדי להשתמש במערכת. בנוסף, המערכת הציגה שילוב מוצלח של הפונקציות השונות, באופן שתרם לחוויית שימוש זורמת ונעימה. הציון הגבוה מעיד על התאמה מצוינת של המערכת למשתמשים מכל רקע, כולל כאלו ללא ניסיון טכנולוגי קודם.

1. **שלושת המדדים הם -**

### 1. Uptime – זמן זמינות

אחוז הזמן שהמערכת זמינה למשתמשים ללא תקלות.  
 זמינות גבוהה משקפת אמינות. אם המערכת "נופלת" לעיתים קרובות – המשתמשים יאבדו אמון.

### 2. Daily Active Users

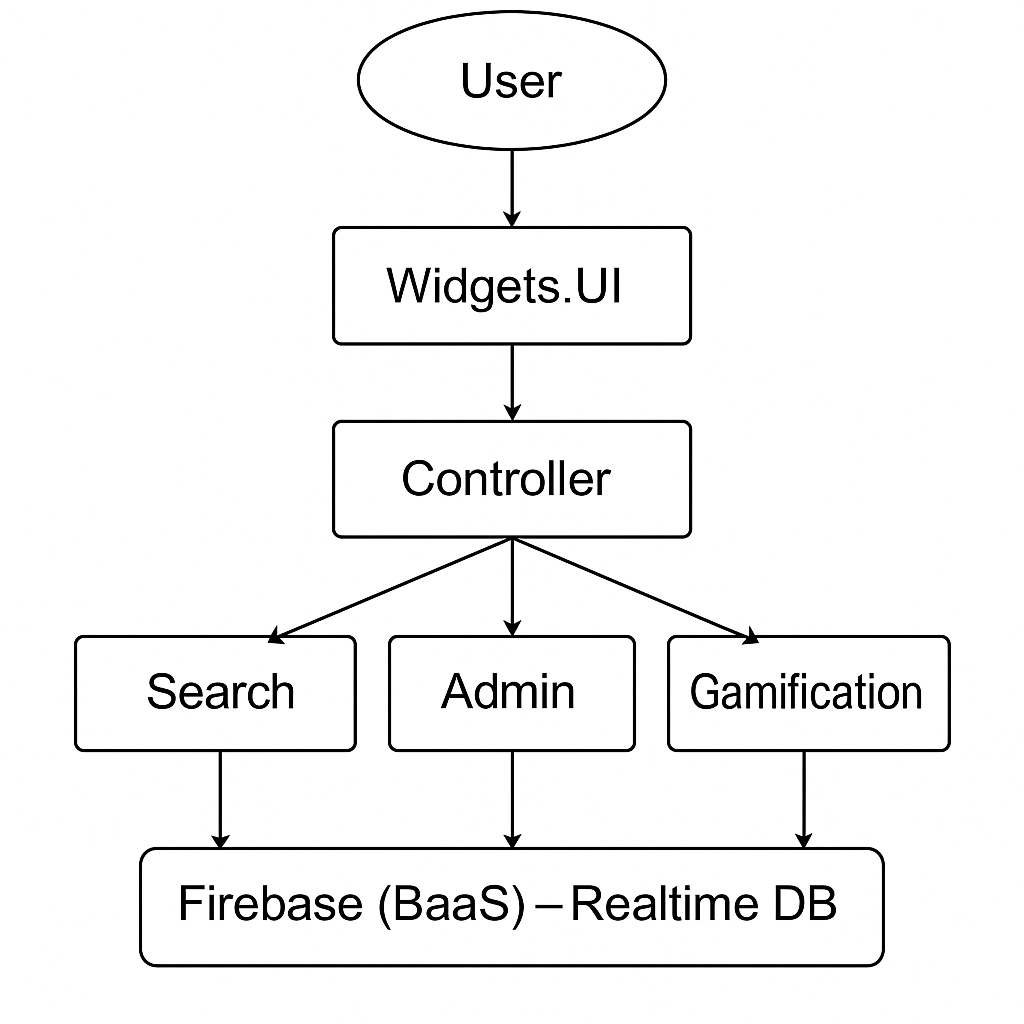
מספר המשתמשים שנכנסו למערכת ביום מסוים וביצעו פעולה במערכת למשל: חיפוש, משחק   
 מעיד על מעורבות ושימוש בפועל – לא רק הרשמה.

### 3. Average Response Time – זמן תגובה ממוצע

כמה זמן לוקח למערכת להחזיר תשובה לבקשות משתמש כמו פתיחת עמוד, מתן תוצאות לחיפוש.  
  במערכת שבה יש חיפוש, שאילתות, הצגת תוצאות – זמן תגובה הוא חלק חשוב מחוויית המשתמש

1. המערכת שלנו בנויה לפי ארכיטקטורת **Client–Server עם שכבת Firebase כ-Backend כשירות (BaaS)**, בדגש על מודל **MVC**, שהיא גישה נפוצה בארכיטקטורות מודולריות ונלמדה בהרצאה 7 בקורס. המערכת פועלת באינטראקציה בין המשתמש (Client), ממשק המשתמש הגרפי (View), לוגיקת העיבוד (Controller), ואחסון הנתונים (Model – Firebase). להלן ההסבר המלא כולל דיאגרמה והקישור לחלקי הקוד:

### הארכיטקטורה שבחרנו: MVC מעל Client–Firebase



### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### פירוט לפי רכיבי הארכיטקטורה:

#### 1. View (ממשק משתמש – UI)

הממשק נבנה באמצעות ipywidgets והוא מציג למשתמש את האלמנטים השונים בכל טאב: טופס התחברות, תוצאות חיפוש, דשבורד גראפי, מערכת גיימיפיקציה.  
 🔹 קובץ קוד: *כל השורות שמשתמשות ב־widgets.Tab, VBox, HBox וכו’* 🔹 דוגמאות:

* main\_container, tab0\_content, tab1\_content, searchResults, logo\_widget
* שדות username\_field, Query, user\_dropdown וכו’.

#### 2. Controller (לוגיקת עיבוד ובקרה)

כאן נמצאים כל הפונקציות שמבצעות לוגיקה על הקלטים של המשתמש ומנהלות את זרימת המידע במערכת.  
 🔹 קובץ קוד: פונקציות כמו:

* on\_click\_search, on\_click\_lucky — שייכות למנוע החיפוש
* assign\_task, update\_tasks\_list — שייכות למנגנון הגיימיפיקציה
* handle\_login\_click, handle\_logout\_click — שייכות לממשק המנהל
* refresh\_dashboard, create\_index\_inspector, update\_index\_word\_box — ניהול האינדקס וטעינה

#### 3. Model (ניהול מידע – אחסון וטעינה מ־Firebase)

האובייקט FBconn מנהל את הגישה ל־Firebase Realtime Database – היכן שמאוחסן כל המידע: אינדקסים, משימות, משתמשים וניקוד.  
 🔹 קובץ קוד:

* IndexService – מכיל את בניית האינדקס, אחזור ושמירה (שורות ~30–230)
* שימושים ב־FBconn.get, FBconn.put, FBconn.patch וכו’ לאורך כל המערכת  
   🔹 דוגמאות:
* load\_index\_from\_firebase, add\_index\_to\_firebase, convert\_index\_for\_firebase
* FBconn.put('/gamification/taskBank/', task\_id, task\_data) – ב־Gamification
* FBconn.get('/myMqttIndex/', None) – טענת אינדקס

### סיכום

המערכת שלנו משתמשת בארכיטקטורת MVC שמופרדת בצורה ברורה:

* **View**: מציג את הנתונים ומקבל קלט מהמשתמש
* **Controller**: מטפל בלוגיקה, עיבוד, שליטה וניווט
* **Model**: שומר את הנתונים ב־Firebase ומחזיר מידע לפי הצורך

שכבות אלו בנויות כך שניתן לתחזק ולשנות כל אחת מהן בצורה מבודדת, בהתאם לעקרונות הארכיטקטורה שהוצגו בהרצאה.

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

**פיצ'ר ראשון: כפתור "I'm Feeling Lucky" ("יותר מזל משכל")**

הוספנו למערכת כפתור פונקציונלי חדש בשם "I'm Feeling Lucky" (בעברית: "יותר מזל משכל"). כפתור זה מאפשר למשתמש לדלג על שלב בחירת התוצאה הרצויה מתוך רשימת תוצאות החיפוש, ולקבל ישירות את התוצאה הרלוונטית ביותר כפי שנקבעה על ידי מנגנון הדירוג של המערכת. הכפתור שימושי במיוחד כאשר מדובר בשאילתה ארוכה או מרובת מילים – במקום לגלול ולעבור בין הקישורים המופיעים בתוצאות, המשתמש מופנה אוטומטית לאתר שהמערכת העריכה כהכי מתאים.

הפיצ’ר ממומש בפונקציה on\_click\_lucky, שממוקמת בקובץ בקירוב לשורה 766. הפונקציה מבצעת חיפוש בהתאם לאינדקס הטעון בזיכרון, ממיינת את תוצאותיו לפי מספר הופעות של מילות השאילתה ומזעיקה את התוצאה הראשונה לפתיחה בלשונית חדשה.

הכפתור שולב בלשונית החיפוש (Tab מספר 3 בממשק) לצד כפתור החיפוש הרגיל, כך שהמשתמש יכול לבחור אם לבצע חיפוש רגיל או להשתמש בגישה המהירה והאוטומטית של פיצ’ר זה.

**פיצ’ר שני: ממשק צפייה באינדקסים עם Stemming עבור המנהל**

בנוסף, הוספנו למערכת פיצ’ר ייעודי למנהלים, המאפשר להם לצפות בכל המילים הקיימות באינדקס של מנוע החיפוש – אינדקס אשר מאוחסן ומתוחזק ב־Firebase. באמצעות שדה חיפוש ייחודי, המנהל יכול להזין מילה כלשהי ולקבל תוצאות גם אם הקליד רק את השורש של המילה (למשל: הקלדת "comput" תציג גם "compute", "computer" וכו'). לצורך כך, תהליך החיפוש משתמש באלגוריתם Porter Stemmer, המזהה מילים על סמך שורש לשוני משותף.

החלק המרכזי של מימוש הפיצ’ר נמצא בפונקציה create\_index\_inspector, בין השורות 1130 עד 1210 בקובץ. ברגע שהמנהל מתחיל להקליד בשדה החיפוש, המערכת מבצעת תהליך stemming ומסננת את תוצאות התצוגה בהתאם לשורש המחושב. בנוסף, ניתן לרענן את הנתונים באמצעות כפתור “ Refresh Index” אשר טוען מחדש את המילים מה־Firebase, ואף לייצא את כל המילים לקובץ טקסט באמצעות כפתור “ Export Words”.

ממשק זה מופיע בטאב הראשון (Admin) ונטען באופן אוטומטי לאחר התחברות מוצלחת של המנהל. הפיצ’ר משולב כחלק מממשק הניהול המרכזי ומספק יכולת בקרה ומעקב מדויקת אחר תוכן האינדקס של מנוע החיפוש.