

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

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	הכל

	<p>השונים של המערכת, שמירה על קשר רציף עם צוות ה-Frontend, ה-QA, וה-UI להבטחת עבודה משולבת ונכונה.</p>	
הכל	<p>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.</p>	<p>חן צפיר - QA, scrum master</p>
הכל	<p>פיצ'ר, לוגו, מסך תוצאות חיפוש, עבודה צמודה עם צוות ה-Frontend וה-Product Manager להבטחת חוויית משתמש עקבית.</p>	<p>יובל לרפלד - UI</p>

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 בפורט).

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

	follows	haven't	if	known	moreover	nothing
else	for	having	ignored	l	most	novel
elsewhere	former	he	immediate	last	mostly	now
enough	formerly	he's	in	lately	much	nowhere
entirely	forth	hello	inasmuch	later	must	o

especially	four	help	inc	latter	my	obviously
et	from	hence	indeed	latterly	myself	of
etc	further	her	indicate	least	n	off
even	furthermore	here	indicated	less	name	often
ever	e	here's	indicates	lest	namely	oh
every	g	hereafter	inner	let	nd	ok
everybody	get	hereby	insofar	let's	near	okay
everyone	gets	herein	instead	like	nearly	old
everything	getting	hereupon	into	liked	necessary	on
everywhere	given	hers	inward	likely	need	once
ex	gives	herself	is	little	needs	one
exactly	go	hi	isn't	look	neither	ones
example	goes	him	it	looking	never	only
except	going	himself	it'd	looks	nevertheless	onto
f	gone	his	it'll	ltd	new	or
far	got	hither	it's	m	next	other
few	gotten	hopefully	its	mainly	nine	others
fifth	greetings	how	itself	many	no	otherwise
first	h	howbeit	j	may	nobody	ought
five	had	however	just	maybe	non	our
followed	hadn't	i	k	me	none	ours
following	happens	i'd	keep	mean	noone	ourselves
	hardly	i'll	keeps	meanwhile	nor	out
	has	i'm	kept	merely	normally	outside
	hasn't	i've	know	might	not	over
	have	ie	knows	more		overall

own	said	some	that	though	us	were	will
p	same	somebody	that's	three	use	weren't	willing
particular	saw	somehow	thats	through	used	what	wish

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

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

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 איפורמטיבי - 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.

2. 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 DB כלומר הוספת אינדקס, הוספת משימות, הסרת משימות, שינוי ניקוד לעובדים, הוספת קישורים לאינדקס קיים, הסרת מילים וכו).		
בחרנו בעיצוב מינימליסטי תוך מתן דגש על הלוגו והעיצוב הכולל.	לא	שיפור העיצוב
הוספנו לכל גרף הסבר כך שיהיה מובן מה הגרף מציג.	כן	להוסיף כותרת ותיאור לתוצאות
מתקשר להוספת כותרת ותיאור לתוצאות. בוצע.	כן	סידור הגרפים

- 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.

3. 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				

X					לדעתי המערכת הינה מסורבלת לשימוש.	8
				X	הרגשתי מאד בטוח באופן השימוש שלי במערכת.	9
X					הייתי צריך ללמוד הרבה לפני שהצלחתי להשתמש במערכת.	10

סכום הניקוד:

$$37 = 4 + 4 + 4 + 3 + 4 + 4 + 3 + 4 + 3 + 4$$

$$92.5 = 2.5 \times 37 = \text{ציון SUS סופי}$$

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.

4. 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

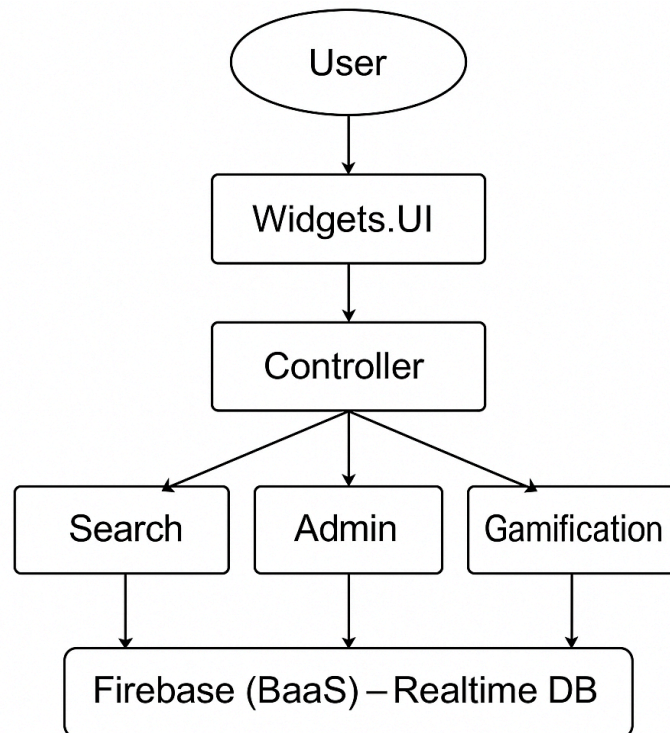
מספר המשתמשים שנכנסו למערכת ביום מסוים וביצעו פעולה במערכת למשל: חיפוש, משחק מעיד על מעורבות ושימוש בפועל – לא רק הרשמה.

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

5. המערכת שלנו בנויה לפי ארכיטקטורת Client-Server עם שכבת **Backend-כ-Firebase** 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 גרפי, 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־`Firestore`. באמצעות שדה חיפוש ייחודי, המנהל יכול להזין מילה כלשהי ולקבל תוצאות גם אם הקליד רק את השורש של המילה (למשל: הקלדת "comput" תציג גם "computer", "compute" 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־`Firestore`, 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.