



אפקה המכללה האקדמית להנדסה בתל-אביב  
TEL-AVIV ACADEMIC COLLEGE OF ENGINEERING

המחלקה להנדסת תוכנה

שם הפרויקט: מערכת המלצות קורסים ONLINE

Personalization and Recommendation for  
courses

ספר הפרויקט

שמות הסטודנטים: אור שולרופר, דן קוזק

מספר תעודת זהות: 312497175, 310994710

שם המנחה: ד"ר דינה גורן-בר

חתימת המנחה: עמ' 2

תאריך ההגשה: 23.07.2020

# אישור המנחה

אישור הגשה ספר פרויקט וכל המסמכים

Inbox x



**Diana Goren-bar**

to me, Dan

1:33 PM (0 minutes ago)



Hebrew



English

[Translate message](#)

[Turn off for: Hebrew](#)

לאור ודן שלום,

אני מאשרת הגשה של ספר הפרויקט וכל הנספחים.

מאחלת הצלחה בהמשך הדרך!

דינה



Reply



Reply all



Forward

# תודות

ברצוננו להודות למנחה שלנו, ד"ר דינה גורן בר, שישבה וכיוונה אותנו בסבלות בכל היבט של הפרויקט, מתכנן האלגוריתמים ועד לנראות הפרויקט. בהחלט למדנו הרבה ממך בתקופה זו והתפתחנו בכל צעד. אנחנו שמחים שנבחרת להנחות אותנו. אז תודה לך דינה! לא יכולנו לעשות זאת בלעדייך.

10	2 תקציר
11	Executive Summary 3
13	4 מילון מונחים
13	K-MEAN 4.1
13	4.2 דמיון קוסינוסי
13	Naive Bayes 4.3
13	4.4 משפט בייס
14	Decision Tree 4.5 - עץ החלטה
14	Random Forest 4.6 - יער אקראי
14	Logistic Regression 4.7
15	within cluster sum of errors - wcss 4.8
15	4.9 "מטריצת משתמש קורס"
15	איור 1.1 - מטריצת משתמש קורס
15	4.10 למידה מפקחת ((Supervised learning
16	4.11 למידה לא מפקחת ((Unsupervised learning
16	4.12 מסלול למידה
16	4.13 מרחב ההיפותזות
16	4.14 תוחלת
16	5 מבוא
17	6 מטרות ויעדים
17	6.1 מטרות הפרויקט:
18	6.2 יעדי הפרויקט:
18	6.3 מדדים:
18	6.3.1 אופן השגתם, דרך המדידה ותוצאותיהם של המדדים
19	7 סקירת ספרות ביקורתית מורחבת / סקר שוק.
19	7.1 סקירת ספרות
19	7.1.1 דמיון קוסינוסי לחישוב דימיון בין שני מסמכים
20	איור 1.2 - נוסחאות לחישוב דמיון קוסינוסי
20	7.1.2 דמיון רך ומדידת קוסינוס רכה למציאת דימיון בין שני מסמכים
21	איור 1.3 - השוואה בין דמיון קוסינוסי לדמיון קוסינוס רך
21	7.1.3 מערכת ממליצה
22	7.1.3.1 מה היא מערכת ממליצה?

22	7.1.3.2 הינה כמה מסוגי מערכות ממליצות הרלוונטיות לפרוייקט שלנו:
22	Collaborative Recommender System 7.1.3.3
23	Content based Recommender System 7.1.3.4
23	Demographic based Recommender System 7.1.3.5
23	7.1.4 למידת מכונה
25	k-means 7.1.5
25	Within Cluster Sum of Squares -WCSS 7.1.5.1
25	7.1.5.2 שיטת המרפק
26	איור 1.4 - גר של שיטת מרפק
26	7.2 סקר שוק
27	איור 1.5 - טבלת השווה בין מתחרים
<b>28</b>	<b>8 ניתוח חלופות מערכת</b>
28	8.1 מימוש מאגר נתונים על קורסים:
28	8.1.1 סריקה מילולית אוטומטית של אתרי האינטרנט.
28	8.1.2 הכנסת קורסים ע"י מציעי הקורסים (חלופה שנבחרה).
28	8.1.3 החלופה שנבחרה והושמה במערכת שלנו
29	8.2 איסוף פרטים אישיים ממשתמשים מחפשי ההמלצות
29	8.2.1 התחברות והרשמה למערכת דרך פרופיל של רשתות חברתיות.
29	8.2.2 הרשמה למערכת דרך מילוי טופס הרשמה
29	8.2.3 החלופה שנבחרה והושמה במערכת שלנו
30	8.3 סוג פלטפורמה
30	8.3.1 הקמת פלטפורמת web
30	8.3.2 הקמת פלטפורמת Android
30	8.3.3 החלופה שנבחרה והושמה במערכת שלנו
<b>31</b>	<b>9 דרישות המערכת</b>
31	9.1 דרישות פונקציונליות:
31	9.1.1 דרישות עבור לקוח אשר מעוניין לקבל המלצה לקורס:
31	9.1.1.1 הצלבת נתונים בין משתמשים שונים
31	9.1.1.2 PERSONALISE אלגוריתם
31	9.1.1.3 ממשק הרשמת משתמש חדש למערכת
31	9.1.1.4 ממשק חידוש פרטי המשתמש
31	9.1.1.5 ממשק צפיה בקורסים המוצעים למשתמש

31	9.1.1.6 ממשק קישור לאתר של ספק הקורס
32	9.1.2 דרישות עבור ספק הקורס
32	9.1.2.1 ממשק הזנת קורס
32	9.1.2.2 ממשק שינוי פרטים של ספק הקורסים
32	9.1.2.3 ממשק המראה את רשימת הקורסים של ספק הקורסים
32	9.2 דרישות לא פונקציונליות
32	9.2.1 דרישות עיצוב, דרישות שימוש וממשק משתמש
32	9.2.1.1 ממשק ידידותי למשתמש
32	9.2.1.2 עיצוב אחיד
32	9.2.1.3 פעולות ברורות למשתמש
33	9.2.2 דרישות ביצועים
33	9.2.2.1 מינימום של 80% הצלחה בהצגת קורסים המתאימים למשתמש
33	9.2.2.2 התאמות לקורסים יתעדכנו בזמן אמת
33	9.2.3 דרישות הפעלה
33	9.2.3.1 ממשק קל ומהיר
33	9.2.3.2 השרת יחד עם ה data ישבו על ענן.
33	9.2.4 דרישות תחזוקה ותמיכה
33	9.2.4.1 עדכון מערכת Web לא דורש פגיעה בתפקוד עמדות משתמשי הקצה.
33	9.2.4.2 המערכת תתפקד בצורה זהה בדפדפנים הסטנדרטיים.
33	9.2.5 דרישות אבטחה
33	9.2.5.1 ניהול משתמשים במערכת יתאפשר ע"י מנהל מערכת בלבד.
34	9.2.5.2 רישום מחפש הקורס, ומספק הקורס לאתר יתאפשר עם מייל תקין בלבד.
34	9.2.6 זמינות
34	9.2.6.1 המערכת תהיה זמינה ב 95% מהזמן כלומר שירות רציף.
34	<b>10 אפיון המערכת</b>
35	10.1 תרשים בלוקים
35	איור 1.6 - תרשים בלוקים של כל המערכת
36	<b>11 ניתוח חלופות טכנולוגיות.</b>
36	11.1 בחירת מאגר נתונים חיצוני
36	<b>12 תכן המערכת</b>
36	12.1 ארכיטקטורת המערכת
36	12.1.1 בצד המשתמש יוכל המשתמש לעשות את הדברים הבאים:

37	12.1.2 כל הלוגיקה של המערכת תתבצע בצד השרת:
38	12.1.3 האלגוריתם "ההיברידי" של כל המערכת
38	איור 1.7 - תרשים זרימה בין חלקי האלגוריתם
38	12.1.3.1 יחסים בין חלקי האלגוריתמים:
39	12.1.3.2 סינון דמוגרפי
39	12.1.3.2.1 הכנת נתונים
40	12.1.3.2.2 יישום האלגוריתם
40	12.1.3.3 סינון שיתופי
40	12.1.3.3.1 הכנת נתונים
40	3.1.3.2 יישום האלגוריתם
40	12.1.3.4 סינון מבוסס תוכן
40	12.1.3.4.1 הכנת נתונים
41	12.1.3.4.2 יישום האלגוריתם
41	12.1.3.5 איחוד תוצאות אחרי ריצת סינון שיתופי וסינון מבוסס תוכן
42	12.2 חלופות לתכן המערכת
42	12.2.1 כמות קורסים בהוצאת המלצות
42	12.4 הרציונל מאחורי שילוב של אלגוריתמים לאלגוריתם היברידי אחד של למידת מכונה מקורי של "מתלבט בשבילך"
<b>43</b>	<b>13 התוצר</b>
44	13.1 תוצרים שפותחו
44	13.1.1 אלגוריתמים שפותחו
44	13.1.2 שירותים שפותחו
44	13.1 אלגוריתם להמלצות למידת קורסים און ליין, היברידי בהתאמה אישית
45	13.1.1 אופן בדיקת האלגוריתם
45	13.1.2 תסריט הפעלה של קבלת המלצות לקורסים בהתאמה אישית
46	איור 1.9 - תרשים זרימה לקבלת המלצות של קורסים בהתאמה אישית
46	13.2 הוצאת קורסים לפי נושא
46	13.2.1 תסריט הפעלה של קבלת קורסים לפי נושא
47	איור 1.10 - תרשים זרימה לקבלת קורסים לפי נושא
47	13.3 רישום משתמש חדש למערכת באמצעות מייל
47	13.3.1 תסריט הפעלה
48	איור 1.11 - תרשים זרימה לרישום משתמש חדש באמצעות מייל
48	13.4 רישום משתמש חדש למערכת באמצעות חשבון גוגל

48	13.4.1 תסריט הפעלה
49	איור 1.12 - תרשים זרימה לרישום משתמש חדש באמצעות חשבון גוגל
49	13.5 עדכון פרטים דמוגרפיים של משתמש
49	13.5.1 תסריט הפעלה
50	איור 1.13 - תרשים זרימה של עדכון פרטים דמוגרפיים של משתמש
50	13.6 עדכון קורסים שנלקחו ע"י משתמש
50	13.6.1 תסריט הפעלה
51	איור 1.14 - תרשים זרימה של עדכון קורסים שנלקחו ע"י משתמש
51	13.7 הכנסת קורס חדש ע"י משתמש מורשה
51	13.7.1 תסריט הפעלה
52	איור 1.15 - תרשים זרימה של הכנסת קורס חדש ע"י משתמש מורשה
52	13.8 כניסה למערכת באמצעות מייל וסיסמה
53	איור 1.16 - תרשים זרימה של כניסה למערכת באמצעות מייל וסיסמה
53	13.9 כניסה למערכת באמצעות חשבון גוגל
54	איור 1.17 - תרשים זרימה של כניסה למערכת באמצעות חשבון גוגל
<b>54</b>	<b>14 תכנון הפרויקט</b>
54	14.1 ריכוז שינויים
55	איור 1.18 - טבלת ריכוז שינויים
55	14.2 ניהול סיכונים
55	14.2.1 מציאת סט נתונים מתאים
<b>55</b>	<b>15 בדיקות והערכה</b>
55	15.1 בדיקות מדדים ואלגוריתמים
55	15.1.1 אופן בדיקת אלגוריתם היברידי ראשי
56	15.1.2 בדיקת השפעת משקולות של נתונים דמוגרפיים על אלגוריתם היברידי
56	15.1.2.1 תוצאות
57	15.1.3 בדיקת שימוש דמיון קוסינוסי או דמיון קוסינוס רך
57	15.1.3.1 תוצאות
57	15.1.4 בדיקת תרומה של סינון דמוגרפי
57	15.1.5 בדיקת תרומה של סינון שיתופי וסינון מבוסס תוכן בנפרד
58	15.1.5.1 תוצאות
58	15.1.6 בדיקת הגבלת כמות הקורסים, לשימוש בסינון שיתופי
58	15.1.6.1 תוצאות



59	15.1.7 מסקנות של בדיקות מדדים ואלגוריתמים
59	15.2 בדיקות פונקציונליות
59	15.1.6 בדיקת הזנת נתונים בעת רישום משתמש במערכת
59	15.1.7 פונקציונליות של כפתור בקשת המלצות לקורסים
59	15.1.8 בדיקת מהירות ריצת האלגוריתם
59	15.1.9 בדיקת נכונות פלט אחרי לחיצה על כפתור בקשת המלצות לקורסים
59	15.3 בדיקות שמישות
60	15.3.1 סיבוכיות של הגעה להמלצות
60	15.3.2 עדכון פרטים אישיים
60	15.3.3 קבלת קורסים לפי נושא
60	15.3.4 עורך התהליך של הגעה להמלצות
60	15.4 דוגמאות הפעלה
60	15.2.1 בקשת המלצות בהתאמה אישית
60	15.2.2 הרשמה למערכת והזנת פרטים דמוגרפיים
64	15.2 בדיקות שמישות
64	15.3.1 סיבוכיות של הגעה להמלצות
64	15.3.2 עדכון פרטים אישיים
64	15.3.3 קבלת קורסים לפי נושא
65	15.3.4 עורך התהליך של הגעה להמלצות
65	<b>16 הצעה לעבודת המשך</b>
65	<b>17 רשימת מקורות</b>
66	<b>18 נספחים</b>
67	הפוסטר
67	18.1.נספח SRD
67	1. הקדמה
68	2. תיאור הארכיטקטורה שנבחרה
69	PYSPARK 2.1
69	mysqlDB 2.2
69	2.3 user interface - ממשק משתמש
69	3 הגדרת דרישות
69	3.1 ניתוח פונקציונאלי ראשוני
69	3.1.1 רישום משתמש חדש למערכת

70	3.1.2 בקשה להמלצות על קורסים - משתמש שלקח קורסים בעבר
71	3.1.2 בקשה להמלצות על קורסים - משתמש שלא לקח קורסים בעבר
72	3.2 דרישות ביצועים
72	3.3 דרישות משאבים
72	3.4 דרישות אבטחה
72	1 הקדמה
73	2 תכן ארכיטקטורת המערכת
73	2.1 ארכיטקטורה נבחרת
73	2.2 צד הלקוח
73	2.3 צד השרת
74	2.4 ארכיטקטורת ממשקים
74	3 תיאור רכיבים
74	3.1 שרת ענן
75	PYSPARK 3.2
75	MYSQL 3.3
76	4 עיצוב ממשק משתמש
78	<b>נספח std</b>
78	1. הקדמה
78	2. פריטים נבדקים
78	2.1 תכונות לבדיקה
79	2.2 תכונות שלא יבדקו
79	2.3 דרישות סביבה
80	2.4 אירעי בדיקה
80	2.4.1 בדיקת הזנת נתונים בעת רישום משתמש במערכת:
80	2.4.2 פונקציונליות של כפתור בקשת המלצות לקורסים
81	2.4.3 בדיקת נכונות פלט אחרי לחיצה על כפתור בקשת המלצות לקורסים
81	2.4.4 בדיקת נכונות פלט אחרי לחיצה על כפתור בקשת המלצות לקורסים לפי נושא
81	2.4.5 בדיקת מהירות ריצת האלגוריתם
82	2.4.6 בדיקת אחוזי קליעת הפלט של האלגוריתם לנתוני בדיקה
82	3. בדיקות פונקציונליות ושימושיות.
83	4. לוח זמנים

## 2 תקציר

היום יותר ויותר אנשים מעוניינים בלמידת קורסים און ליין בין אם זה לקידום קריירה או העשרה. ברשת קיימים המון מוסדות המספקים קורסים און ליין בכל נושא. כלומר מי שיחפש להעשיר את הידע שלו, יצטרך להתלבט בין המון קורסים שיתאימו לצרכים שלו. בשביל לפתור בעיה זו, יצרנו מערכת המלצות לקורסים ONLINE: מתלבטים בשבילך (להלן תקרא "מתלבטים בשבילך") אשר תצמצם אפשרויות עבור מחפש הקורסים בהתאמה אישית עבור כל משתמש, באמצעות הוצאת המלצות לקורסים.

ההתאמה האישית היא בשתי היבטים: הראשון זה קבלת אפשרויות להמשך מסלול הלמידה הנוכחי של המשתמש, ע"פ מסלולי למידה של משתמשים אחרים המתחילים כמו המסלול השלם של המשתמש הנ"ל. ההיבט השני הוא קבלת המלצות לקורסים שהכי דומים לקורסים שנלקחו בעבר ע"י המשתמש הנ"ל. התאמות אלו מושגות ע"י הרצת אלגוריתם למידה היברידית, המשלב סינון מבוסס תוכן, סינון דמוגרפי וסינון שיתופי. כאשר סינון דמוגרפי מהווה סינון קדם לסינון שיתופי, ולצד זה ירוץ סינון מבוסס תוכן. ובסוף ריצתם יהיה איחוד בין התוצאות. תוצאות מכל אלגוריתם יהי ומוגבלות במספרם. מסינון מבוסס תוכן ילקחו 2 המלצות עבור כל קורס שנלמד ע"י המשתמש. מתוך ההמלצות שקיבלנו מאלגוריתם סינון שיתופי, ניקח את 5 הקורסים הכי פופולריים, אשר לא נמצאים בהמלצות שקיבלנו מסינון מבוסס תוכן. בסוף ריצת האלגוריתם מציגים את כל הקורסים שאספנו כהמלצות לממשק משתמש.

המערכת פועלת בפלטפורמת WEB בשיטת client-server, ומשתמשת במסדר נתונים חיצוני. המערכת מכילה ממשק משתמש אשר כולל הרשמה עבור משתמשים, מסירת הנתונים

הדמוגרפיים והאישיים שלהם על ידם, קבלת המלצות לקורסים בהתאמה אישית, קבלת קורסים לפי נושא והכנסת קורסים חדשים למערכת (ע"י משתמש מיוחד).

המדד שקבענו להצלחת המערכת הוא שמשמש יהיה מעוניין לקחת לפחות קורס אחד בין כל הקורסים המומלצים, במינימום של 80 אחוז מהמקרים של בקשת המלצות.

בשביל להגיע למדד המערכת, השתמשנו בנתונים אמיתיים הכוללים 2 סטים של נתונים:

(1) סט נתונים של MIT ו HARVARD אשר הורדנו ברשת, והוא כלל יותר 400,000 משתמשים, פרטים דמוגרפיים שלהם וקורסים שלמדו בעבר שהורדנו.

(2) סט נתונים על קורסים אשר מסופקים ע"י MIT ו HARVARD הכולל 120 קורסים. את סט זה הרכבנו לבד ע"י לקיחת פרטים על כל קורס בנפרד מאתרים של מוסדות אלו.

ע"י לימוד ואימון המערכת על בסיס נתוני אימון ואז הרצת בדיקות על נתוני בדיקה, הגענו למדד 85.63 אחוז.

לסיכום עמדנו במדד המרכזי של המערכת (80% דיוק), ואף עברנו אותו והגענו לרמת דיוק של 85.63% בקבלת המלצות.

בגרסה הבאה הבאה של המערכת נרצה לשפר את זמן הגישה למאגר נתונים חיצוני, שכרגע מוגבל עקב שיקולים כלכליים. ונרצה לשפר את זמני ריצת האלגוריתם. נרצה גם להוסיף עוד פיצ'רים אשר יקלו ירחיבו את חווית המשתמש במערכת.

### 3 Executive Summary

Today more and more people are interested in learning online courses, whether it is for career advancement or enrichment. There are plenty of institutions on the network that provide online courses on every subject. That means, those who seek to enrich their knowledge will have to decide among lots of courses to suit their needs.

To resolve this issue, we have created an online course recommendation system: Decide for You (hereinafter referred to as "Decide for You") which will reduce course options for users by issuing personalized course recommendations.

The personalization is in two aspects, the first being the possibility of continuing the learning pathways of other users whose beginning of the learning path matches the learning path of the searcher. These personalization are

achieved by running a hybrid learning algorithm that combines content-based filtering, demographic filtering, and collaborative filtering.

From collaborative filtering, which gives learning pathways, we take 5 most popular courses from all learning pathways which he issued. And from content-based filtering we take 2 courses most similar for each course that user already took. At the end we combine all of these courses to recommendations. The system runs on the WEB platform using the client-server method, and uses an external DataBase. The system contains a user interface which includes registering for users, submitting their demographic and personal data, receiving recommendations for courses, receiving courses by subject and introducing new courses into the system.

The measure that we have determined for the success of the system is that a user will want to take at least one course among all the recommendations, in a minimum of 80 percent.

To reach the systems' measure of success, we used real data from 2 datasets:

1) MIT and HARVARD data set that we downloaded online, and included more than 400,000 users, their demographics, and courses we learned in the past that we downloaded.

2) A data set on courses provided by MIT and HARVARD which includes 120 courses. We have compiled this set by taking details of each course separately from the websites of these institutions.

By studying and training the system based on training data and then running tests on test data, we reached 85.63 percent.

In the next next version of the system, we would like to improve the access time to an external database, which is currently limited due to economic considerations. And we want to improve the running times of the algorithm. We would also like to add more features that make it easier to extend the user experience in the system.

## 4 מילון מונחים

### K-MEAN 4.1

זהו אלגוריתם למידת מכונה שמטרתו לחלק את התצפיות ל- $k$  אשכולות לפי מרכזי כובד. כל תצפית משויכת לאחד מ"מרכזי הכובד". זהו מודל סטטיסטי שאינו מתבסס על ידע מוקדם על הנתונים אלא רק על תצפיות בפועל.

### 4.2 דמיון קוסינוסי

דמיון קוסינוסי הוא מדד המשמש למדידת עד כמה המסמכים דומים ללא קשר לגודלם. מבחינה מתמטית, הוא מודד את הקוסינוס של הזווית בין שני וקטורים המוקרנים בחלל רב-ממדי. הדמיון הקוסינוסי הוא יתרון מכיוון שגם אם שני המסמכים הדומים נמצאים זה מזה במרחקים האוקלידיים (בגלל גודל המסמך), רוב הסיכויים שהם עשויים להיות מכוונים קרוב יותר זה לזה. ככל שהזווית קטנה יותר, כך הדמיון הקוסינוסי גבוה יותר.

### Naive Bayes 4.3

מסווג Naive Bayes הוא מודל למידת הסתברות מכונה המשמש למשימת סיווג. עיקר המסווג מבוסס על משפט בייס.

### 4.4 משפט בייס

הוא תוצאה בתורת ההסתברות המאפשרת לחשב הסתברות מותנית של מאורע כאשר יודעים את ההסתברויות המותנות ההפוכות.

## Decision Tree 4.5 - עץ החלטה

עץ החלטה הוא מודל חיזוי בתחום הסטטיסטיקה, כריית נתונים והלמידה החישובית המספק מיפוי בין תצפיות לערכים המתאימים עבורן. עץ החלטה יכול לשמש כמודל חיזוי, הממפה תצפיות על פריט ויוצר מסקנות על ערך היעד של הפריט. הם בנוסף נקראים עצי סיווג או עצי רגרסיה. במבנה של עצים אלה, עלים מייצגים סיווגים אפשריים וענפים מייצגים צירופים של תכונות אשר יובילו למחלקות הסיווג.

## Random Forest 4.6 - יער אקראי

יער אקראי מורכב מעצי החלטה. העצים נוצרים לרוב על ידי דגימה מתוך המאפיינים או מתוך התצפיות (כלומר, באמצעות גישת ה-bagging). כל אחד מהעצים נותן תוצאה לא-אופטימאלית (Suboptimal), על פי רוב, החיזוי בדרך זו משתפר. גישה נוספת ליצירת יער אקראי היא להגריל באופן מקרי את רמות הסף (Thresholds) המשמשות עבור המאפיינים, במקום לחפש את רמת הסף האפשרית הטובה ביותר. זה יכול להיות יעיל מבחינת חישובית שכן מציאת רמת הסף האופטימלית של המאפיין בכל צומת עלולה לקחת זמן רב. ניתן לחשב את חשיבותו של כל אחד מהמאפיינים השונים ביער אקראי באמצעות הרווח הממוצע המשוקלל מהמידע (אשר נמדד על בסיס מדד אנטרופי או על פי מדד ג'יני) עם משקולות שפרופורציונאליות למספר התצפיות שנלקח בצומת מסוים.

## Logistic Regression 4.7

רגרסיה לוגיסטית היא רגרסיה בה המשתנה התלוי הוא בינארי – כלומר, יש בו רק שתי קטגוריות (כמו "כן" ו-"לא", או "גבר" ו-"אישה"), המוצגות באמצעות הערכים 0 ו-1. בשימוש במודל זה נקבל פונקציה של ההסתברות לערך 1 כפונקציה של המשתנים הבלתי-תלויים, כלומר: מה הסיכוי שהערך של המשתנה התלוי יהיה 1 כפונקציה של המשתנים הבלתי-תלויים? נשתמש ברגרסיה לוגיסטית כאשר המשתנה המוסבר הוא בסולם קטגוריאלי בעל שני ערכים בלבד.

## within cluster sum of errors - wcss 4.8

משתמשים במדד wcss כדי למצוא את מספר האשכולות האופטימלי.

## 4.9 "מטריצת משתמש קורס"

	משתמש 1	משתמש 2	.....	משתמש n
קורס 1				
.....				
קורס n				

### איור 1.1 - מטריצת משתמש קורס

כאשר משתמש x לקח קורס y אז במשבצת (x,y) יהיה 1 ואם לא לקח אז יהיה 0.

## 4.10 למידה מפוקחת (Supervised learning)

למידה מפוקחת היא טכניקה המאפשרת לבנות "מכונות" שלומדות להכליל פתרונות על בסיס מאגר גדול של דוגמאות "פתורות". לדוגמה - אם רוצים לבנות "מכונה" שמקבלת תמונות ויודעת להגיד האם יש בתמונה "כלב", "חתול" או אף אחד משניהם - ניתן לבנות "מכונה" כזו, כך שהיא תוכל לסווג אוסף גדול של תמונות בתנאי תאורה משתנים, כאשר סוגי החתולים והכלבים משתנים וכדומה - ובלבד שאיפשרו למכונה "להתאמן" על כמות גדולה של תמונות "מתויגות" - כלומר, "נתנו לה לראות" תמונה אחרי תמונה, ביחד עם הסווג - האם יש בה כלב, חתול או אף אחד משניהם. בלשון פורמלית יותר: למידה מונחית היא טכניקה בלמידה חישובית על מנת ללמוד פונקציה ("היפותזה") על בסיס סט אימון. סט האימון מכיל דגימות מתוך התפלגות משותפת של מרחב האלמנטים ומרחב התיוגים. מטרת הלמידה המונחית היא ללמוד על ההתפלגות המשותפת, ומתוך כך לנבא עבור אלמנט את התיוג שלו. בעיות שבהן עוסקת למידת מונחית כוללות בין היתר רגרסיה ובעיות סיווג.



## 4.11 למידה לא מפוקחת (Unsupervised learning)

טכניקה בלמידה חישובית שבה מנסים ללמוד את התכונות והמבנה של אוסף דוגמאות נתונים כאשר הנתונים זמינים כפי שהם ללא תוספת תיוגים. למשל, נתונים הכוללים מידע רפואי על נבדק, כמו: חום, דופק, לחץ דם; ללא תיוג המציין אם הנתונים שייכים לאדם חולה או בריא. התוכנות לגבי התכונות של הנתונים הבלתי מתויגים יכולות לשמש למשל כדי לזהות אנומליות או כדי לחלק את הנתונים לקטגוריות.

## 4.12 מסלול למידה

מסלול למידה של משתמש זה כל הקורסים שהוא לקח עד היום ומתועדים במערכת שלנו. קורסים אלו מבטאים כיוון למידה ומחשבה של משתמש.

## 4.13 מרחב ההיפותזות

מרחב ההיפותזות הוא מרחב המכיל פונקציות ממרחב האלמנטים למרחב התיוגים. עבור סט האימון לכל היפותזה יש שגיאה אמפירית - אחוז השגיאות של ההיפותזה על סט האימון. לכל היפותזה יש שגיאת הכללה - התוחלת של טעות בתיוג על פני ההתפלגות המשותפת.

## 4.14 תוחלת

בתורת ההסתברות ובסטטיסטיקה, התוחלת של משתנה מקרי היא ממוצע הערכים אותם צפוי המשתנה לקבל, משוקלל על-פי ההסתברויות לקבלת הערכים השונים. לפי החוק החזק של המספרים הגדולים, התוחלת היא הערך אליו שואפת התוצאה הממוצעת של ניסוי כשמספר החזרות שואף לאינסוף.

## 5 מבוא

### אין רקע מספרי לגבי קורסים ומחפשי קורסים!!!!!!

אנשים רבים כיום מעוניינים לרכוש ידע בתחומים שונים, אם זה ידע טכנולוגי או ידע מעשי כגון לדעת לבשל וכו'. לצורך כך, הם נרשמים לקורסים העונים על צרכים אלה. לעיתים אנשים אלו לא יודעים איזה קורסים מתאימים להם ואף לא יודעים על קיומם של קורסים מסוימים.

### הפתרונות שהשוק מספק לצרכן הם:

מערכת ההמלצות כיום מפותחות ע"י אתרים למטרות רווח למיניהם, בשביל לתת המלצות אישיות על פי דמוגרפיה, התנהגות צרכנית, פופולריות המוצר. המלצות אלו הם רק עבור קורסים און ליין הקיימים באתר עצמו, ובדרך כלל רק לקידום קריירה (מקצועיים) ולא העשרה. בנוסף לכך אין אתרים שמביאים המלצות מסוג זה לגבי קורסים שהם לא און ליין. מה שקיים זה רק כל ארגון עסקי מביא קורסים משלו והמלצות על עצמו.

### הפתרון שאנו מספקים לבעיה זו הוא:

פלטפורמה מבוססת WEB המציעה התאמות/המלצות אישיות בעזרת אלגוריתם מכונת למידה היברידית, הכולל בתוכו שלושה אלגוריתמים של למידת מכונת. בשביל קבלת המלצות ידרש המשתמש לספק נתונים דמוגרפיים בסיסיים ולבחור קורסים שלמד בעבר מתוך קורסים הקיימים במערכת.

הפלטפורמה שלנו תספק המלצות לקורסים הנמצאים בתוך מאגר קורסים של המערכת שלנו. עם הזמן מאגר זה יגדל ע"י הכנסת קורסים חדשים ע"י מנהלי המערכת. חשוב גם לציין שאלגוריתם שלנו יהיה נקי משיקולים אסטרטגיים עסקיים וקידום אינטרסים.

## 6 מטרות ויעדים

### 6.1 מטרות הפרויקט:

- לספק למשתמשים המלצות לקורסים לפי התאמה אישית אוטומטית.

## 6.2 יעדי הפרויקט:

- יצירת מאגר נתונים של קורסים ולקוחות
- בניית ממשק משתמש WEB .
- בניית מודל צפי לקורסים המתאימים למשתמש.

## 6.3 מדדים:

- מודל בעל מינימום 80 אחוז של צפי נכון לקורסים המתאימים למשתמש.

### 6.3.1 אופן השגתם, דרך המדידה ותוצאותיהם של המדדים

לקחנו 2 מערכי נתונים של MIT ו Harvard University ביחד. מערך ראשון הוא בעל יותר מ 600 אלף שורות של משתמשים, קורסים שלקחו ונתונים דמוגרפיים שלהם. ועוד מערך נתונים שאספנו לבד על 120 קורסים ממוסדות אלו, הכולל מספר קורס, מרצה, נושאים בקורס ותיאור הקורס.

חילקנו את מערך נתוני משתמשים שלקחו קורסים ל 80% נתוני אימון, 10% נתוני ולידציה ו 10% נתוני בדיקה.

את הוולידציה לקחנו כל פעם 10% אחרים (סה"כ 9 חלקים שונים שווים). בכל קטע הרצנו על כל המשתמשים שבולידציה את האלגוריתם המלצות ההיברידי של המערכת שלנו, ובדקנו האם בקורסים שמוציא האלגוריתם מתוך נתוני אימון יש את כל הקורסים שלקחו המשתמשים בנתוני הוולידציה. ממוצע של כל חלקי הוולידציה היה 86.3% של פגיע בקורסים אלו בכלל משתמשים.

# 7 סקירת ספרות ביקורתית מורחבת / סקר שוק.

## 7.1 סקירת ספרות

### 7.1.1 דמיון קוסינוסי לחישוב דימיון בין שני מסמכים

מכל טקסט בונים וקטור בעזרת מילים של הטקסט ובודקים את גודל זווית הקוסינוס בין הטקסטים. **בשלב ראשון** - מנקים את הטקסט מתווים של כללי פיסוק ומספרים. וגם הופכים אותיות גדולות לקטנות.

**שלב שני** - מפרידים את המילים בטקסט וכל מילה הופכת לאסימון.

**שלב שלישי** - יוצרים "שק מילים" שבו בעצם ליד כל מילה נבדלת רשום כמה פעמים היא מופיע באותו טקסט.

**שלב רביעי** - יוצרים TF-IDF הוא גם מודל שקית-מילים אך בשונה ה corpus הרגיל, נותן משקל למטה למילים שמופיעים לעתים קרובות על פני כל תיאורי הקורסים. בעצם מחשב משקולות של מילים: (מספר מופעי מילה בתיאור אחד חלקי מספר מילים באותו התיאור) כפול (לוג של מספר התיאורים שבה מופיע מילה חלקי מספר כל התיאורים)).

**שלב של בדיקת דמיון** - הרלוונטיות לטקסט מחושבת על ידי שימוש בפונקציית הדמיון קוסינוסי:

$$relevance = \frac{wkre \cdot wkse}{\sqrt{wkr^2} \cdot \sqrt{wks^2}} \quad (2)$$

$$wkre = \frac{\sum kre}{wkr_{max}} \quad (3)$$

$$wkse = \frac{\sum kse}{wks_{max}} \quad (4)$$

$$wkr = \frac{\sum kr}{wkr_{max}} \quad (5)$$

$$wks = \frac{\sum ks}{wks_{max}} \quad (6)$$

## איור 1.2 - נוסחאות לחישוב דמיון קוסינוסי

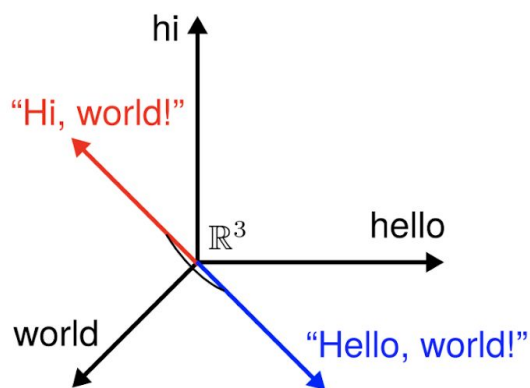
בנוסחאות 5 ו 6 התוצאה מושגת ע"י סכימת משקלים של הטקסט המדובר (TF-IDF) וחילוק במשקל אחד מקסימלי בטקסט השני.

בנוסחאות 3 ו 4 התוצאה מושגת ע"י סכימת משקולות של מילות מפתח משותפות בין שני הטקסטים וחילוק במשקל מקסימלי אחד מהטקסט השני.

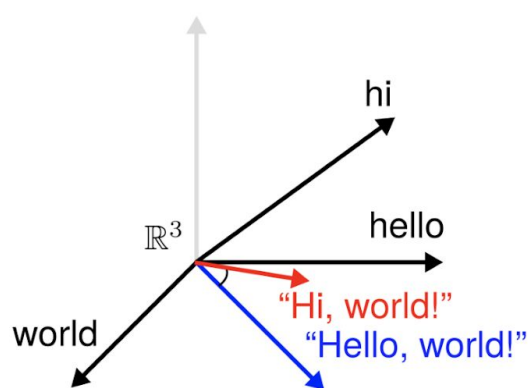
בנוסחה 2 מחשבים את דימיון קוסינוסי בין שני טקסטים, ע"י מה שמצאנו בשאר הנוסחאות. [1]

## 7.1.2 דמיון רך ומדידת קוסינוס רכה למציאת דימיון בין שני מסמכים

Soft similarity יוצאת מנקודת הנחה שמילים יכולות להיות שונות אך עדיין קרובות במשמעותם. למשל מילים כמו play ו game אומנם שונות אך עדיין יש להם הרבה במשותף. במבט זה מציע נוסחאות שונות לחישוב מדויק או משוער של הקוסינוס הרך מידה. לדוגמא באחד מהם אנו שוקלים עבור VSM (מיפוי שרשרת הערך) מרחב תכונות חדש המורכב מזוגות של המאפיינים המקוריים משוקללים על ידי הדמיון שלהם, ועבור תכונות שאינן דומות זו לזו, שלנו הנוסחאות מצמצמות את מידת הקוסינוס הסטנדרטית.



Cosine Similarity



Soft Cosine Measure

### איור 1.3 - השוואה בין דמיון קוסינוסי לדמיון קוסינוס רך

[2]

#### 7.1.3 מערכת ממליצה

מערכות ממליצות הן כלים לאינטראקציה עם מרחבי מידע גדולים ומורכבים. הם מספקים תצוגה אישית של מרחבים אלו (פריטים העשויים לעניין את המשתמש). התחום, שהוקם בשנת 1995, צמח מאוד במגוון הבעיות שטופלו ובטכניקות בהן נעשה שימוש, כמו גם ביישומים המעשיים שלו. מחקר מערכות ממליצים שילב מגוון רחב של טכניקות בינה מלאכותית, כולל למידת מכונות, כריית נתונים, מודלי משתמשים, נימוקים מבוססי מקרה וסיפוק אילוצים, בין היתר. המלצות בהתאמה אישית הן חלק חשוב ביישומי מסחר אלקטרוני מקוונים רבים כמו Amazon.com, Netflix ו-Pandora. שפע זה של ניסיון יישומי פרקטי העניק השראה לחוקרים להרחיב את טווח ההגעה של מערכות ממליצים לתחומים חדשים ומאתגרים.

[3 page 1]

### 7.1.3.1 מה היא מערכת ממליצה?

במערכת ממליצה טיפוסית אנשים מספקים המלצות כקלט, שהמערכת אז צוברת ומכוונת לנמענים מתאימים. בחלק מהמקרים השינוי העיקרי הוא בצבירה; אצל אחרים הערך של המערכת טמון ביכולתה ליצור התאמות טובות בין הממליצים למבקשים המלצות.

[3 page 1]

### 7.1.3.2 הינה כמה מסוגי מערכות ממליצות הרלוונטיות לפרויקט שלנו:

1. Collaborative Recommender System

2. Content based Recommender System

3. Demographic based Recommender System

### Collaborative Recommender System 7.1.3.3

הטכניקה הבולטת ביותר בהמלצה היא המלצה שיתופית. התובנה הבסיסית לטכניקה זו היא סוג של המשכיות בתחום הטעם - אם למשתמשים לאליס ולבוב יש את אותו עניין בפריטים 1 עד  $k$ , הסיכוי הוא שיהיה להם את אותו עניין עבור פריט  $k + 1$ . בדרך כלל כלי עזר אלה מתבססים על דירוגים שהמשתמשים סיפקו עבור פריטים שהם כבר מכירים.

[3 page 3]

היתרון העיקרי בשיטה השיתופית הוא הפשטות שלה. בעיית כלי החישוב הופכת לבעיית חילון ערכים חסרים במטריקס הדירוג. המטריצה הדלילה בה כל משתמש הוא שורה, כל פריט בעמודה, והערכים הם הדירוגים הידועים.

ניתן לסווג סינון שיתופי לשלושה סוגים רחבים:

1. מערכת המלצות שיתופיות פעילה - המשתמש משתתף באופן פעיל בבניית מערכת

ההמלצות על ידי מתן דירוגים או חוות דעת ישירה על פריט. לצד נתוני פרופיל

הפריט, נתוני פרופיל המשתמש נדרשים לשמור במאגר מידע.

2. מערכת המלצות שיתופיות לא פעילה - לוכד ומנתח פעילויות של משתמשים כמו

קנייה רצף או גישה לאינטרנט וכו'.

3. מערכת המלצות שיתופיות מבוססת פריטים - צוות מומחים מנתח את הנתונים של

פריטים דומים ונותנים המלצה ישירה.

[7 page 4]

### Content based Recommender System 7.1.3.4

מערכות המלצות מבוססות תוכן מנתחות תיאורי פריטים שהשתמש בהם מרוצה מהם בעבר כדי לזהות פריטים אחרים המעניינים במיוחד את המשתמש.

[4 page 1]

פריטים שניתן להמליץ למשתמש מאוחסנים לרוב בטבלת מסד נתונים

[4 page 2]

פרופיל האינטרסים של המשתמש משמש את רוב מערכות ממליצות. פרופיל זה עשוי לכלול מספר סוגים שונים של מידע:

1. תיאור של סוגי הפריטים שמעניינים את המשתמש. ע"י הפעלת פונקציה אשר עבור כל פריט מנבא את הסבירות שהמשתמש מעוניין בפריט זה.
2. היסטוריה של אינטראקציות המשתמש עם מערכת ההמלצות. זה עשוי לכלול אחסון של הפריטים שמשתמש צפה ביחד עם מידע אחר על האינטראקציה של המשתמש.

[4 page 4]

### Demographic based Recommender System 7.1.3.5

ניתן להבחין בין סוגי מערכות ההמלצה לשתי גישות המלצה הנפוצות ביותר, מערכת המלצות שיתופיות ומערכת ממליצים על בסיס תוכן. עם זאת, גישות אלו סובלות מבעיית משתמש חדש, המכונה בעיה בהתחלה קרה, אשר יש חוסר דירוגים ראשוני כאשר משתמש חדש מצטרף למערכת. מערכת המלצות דמוגרפית מתמשת בנתונים של משתמש עצמו כמו גיל, מגורים, השכלה, מגדר וכו'.

[6 page 1]

### 7.1.4 למידת מכונה

למידת מכונה היא תת קטגוריה של מדעי המחשב המאפשר יכולת למידה של תוכנה מבלי שנדרשנו ליצור את הכל בקוד בצורה קשיחה ומראש. בעזרת אלגוריתמים שונים יכולה המערכת



לבצע תהליך למידה בעזרת נתוני "אימון" (training data) ולבצע חיזוי של תוצאה בהינתן קלט כל שהוא. ישנם מספר בעיות למידת מכונה שאיתן צריך להתמודד במהלך יצירת אלגוריתם מתאים למשימה ספציפית והן מסווגת בצורה הבאה:

1. סיווג: הקצאת קטגוריה לכל פריט. לדוגמא סיווג קורסים לנושאים כמו: מדעים ומתמטיקה, היסטוריה, ספרות ועוד. ישנם הרבה סיווגים שניתן לעשות לקבוצת פריטים. שיטות להתמודדות עם הבעיה:

• Naive Bayes

• Decision Tree

• Random Forest

• Logistic Regression

2. רגרסיה: חיזוי ערך אמיתי לכל פריט. דוגמאות לרגרסיה כוללות חיזוי ערכי מניות או וריאציות של משנים כלכליים. בבעיה זו, הקנס (penalty) על חיזוי שגוי תלוי בגודל ההבדל בין הערכים האמיתיים לחזויים, בניגוד לבעיית הסיווג, שבה בדרך כלל אין יחסי קרבה בין קטגוריות שונות.

3. דירוג (Ranking): מיון פריטים ע"פ קריטריון ספציפי. הדוגמה הפופולרית ביותר היא

תוצאות חיפוש של מנוע חיפוש אינטרנטי, המתאימות לשאילתת החיפוש של המשתמש.

4. קיבוץ (Clustering): חלוקת פריטים לתחומים אחידים. הקיבוץ מתבצע לרוב לניתוח מערכי נתונים גדולים מאוד.

לדוגמה, בהקשר של ניתוח רשתות חברתיות, אלגוריתמים מקבצים מנסים לזהות "קהילות" בתוך קבוצות גדולות של אנשים.

5. הפחתת ממדיות (Dimensionality reduction): הפיכת הייצוג ראשוני של פריטים לייצוג

של ממדים קטנים יותר של פריטים אלה תוך שמירה על כמה מאפיינים של הייצוג הראשוני.

עם הזמן, הכמות של הנתונים השונים נהיית גדולה מדי לפענוח. לכן, ע"י מציאת קורלציה בין

כמה משתנים, ניתן להפחית אותם למשתנה יחיד.

[5 page 2]

למידת מכונה מתחלקת למספר סוגים, עיקריים שבהם:

Supervised learning – שיטה בה יש לך מיפוי של מוצר X לדירוג/ציון Y שלו.

Unsupervised learning – שיטה בה אין מיפוי ישיר בין מוצר X לדירוג/ציון Y שלו.

[5 page 7]

### k-means 7.1.5

זהו אלגוריתם למידה ללא פיקוח (Unsupervised learning), מטרת האלגוריתם למצוא מקבצים על נתונים.  
מספר המימדים של מערכות צירים יהיה כמספר הפיצ'רים שאליהם רוצים להריץ את האלגוריתם.  
צעד 1:  
לקחת מרכזי אשכולים רנדומליים.  
צעד 2:  
לכל משתמש לחשב מרחק בין הוקטור שלו למרכזי אשכולים.  
צעד 3:  
לשייך את המשתמשים למרכז אשכול הכי קרוב.  
צעד 4:  
חישוב מחדש של מרכזי האשכולים.  
צעד 5:  
אם מרכזי האשכולים לא השתנו סיימנו את האלגוריתם, אם לא חזור לצעד 2.

[8]

### Within Cluster Sum of Squares -WCSS 7.1.5.1

מדידה אחת היא בתוך סכום ריבועי אשכול (WCSS), המודד את המרחק הממוצע מהממוצע בריבוע של כל הנקודות בתוך האשכול עד צנטרואידי האשכול. כדי לחשב WCSS, תחילה תמצא את המרחק האוקלידי בין נקודה נתונה לבין סנטרואיד אליו הוא מוקצה. לאחר מכן אתה חוזר בתהליך זה עבור כל הנקודות באשכול ואז מסכם את הערכים עבור האשכול מחולק במספר הנקודות. לבסוף, אתה מחשב את הממוצע בכל האשכולות. זה ייתן לך את ממוצע WCSS.

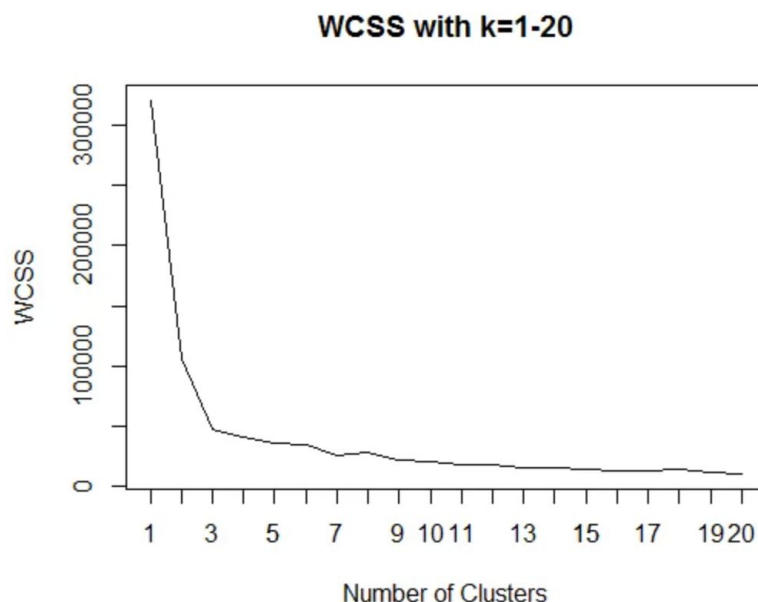
[8]

### 7.1.5.2 שיטת המרפק

כדי לדעת מה הוא מספר אופטימלי של מקבצים שאיתו צריך להריץ את אלגוריתם k-means, צריך להשתמש בשיטת המרפק.  
בשביל זה אנו צריכים 2 נתונים:  
1.  $k$  - מספר המקבצים  
2.  $wcss$  - בתוך אשכול סכום של ריבועים (7.1.6)

מריצים את האלגוריתם k-means התחל ממקבץ 1, ואז 2 וכך הלאה ובכל פעם בודקים את wcss. כאשר נגדיל את k ו-wcss לא יקטן משמעותית אז נאמר שמצאנו k אופטימלי.

[8]



#### איור 1.4 - גר של שיטת מרפק

## 7.2 סקר שוק

משקולת		Udemy	Alison	Coursera	Deci For You
10	הצגת קורסים און ליין לפי התאמה אישית			V	V
5	הצגת קורסים על פי תחומים...	V	V	V	V
2	דרישה של הכנסת קורסים שנלקחו בעבר ע"י משתמש				V
2	לקיחת פרטים אישיים (גיל, רמת השכלה, מגדרת, ארץ מגורים)				V

2	שאלות באיזה נושאי קורס ספציפיים מעוניין המשתמש	V	V	V
8	הגבלת מגוון הקורסים לקורסים שהאתר עצמו מספק בלבד	V	V	V

## איור 1.5 - טבלת השווה בין מתחרים

Udemy - פלטפורמת למידה מקוונת. בתור פיצ'ר צדדי -מציג רשימות קורסים לפי קטגוריות חיפוש או סקלות אחרות למיניהם. יש התאמה אישית לא ברורה ובטוח לא טובה.

Alison - פלטפורמת למידה חינם. בתור פיצ'ר צדדי -מציג רשימות קורסים לפי קטגוריות, חיפוש או סקלות אחרות למיניהם. אין שום התאמה אישית

Coursera -חברה למטרת רווח המציעה שירותי למידה מקוונת. בתור פיצ'ר צדדי -מציג רשימות קורסים לפי קטגוריות, חיפוש או סקלות אחרות למיניהם. יש התאמה אישית טובה למדי. בנוסף, התאמה זו מתבססת על קורסים שהמשתמש נרשם/ניסה להירשם אליהם.

פלטפורמת WEB שלנו - ממליצה על קורסים אונליין לפי התאמה אישית ולפי נושא.

### היתרונות של המתחרים עלינו:

- 1) אפשר לראות בטבלה שאנו היחידים שמחייבים את משתמש למסור את פרטיו האישיים (גיל, רמת השכלה, מגדרת, ארץ מגורים) באת הרשמה.
- 2) אנו היחידים שמבקשים מהמשתמש להכניס קורסים שלמד בעבר (מתוך רשימת בחירה).

### היתרונות שלנו על המתחרים:

- 1) אנחנו מביאים קורסים און ליין בלי הקשר לאתר או מוסד מסויים.
- 2) האלגוריתם שלנו נקי משיקולים מסחריים.
- 3) אלגוריתם שבנינו הוא אלגוריתם "היברידי" מיוחד שמותאם לאופי הלקוחות שלוקחים קורסים און-ליין.
- 4) אנו לא מבקשים שמשתמש יענה על שאלות המצמצמות נושאים לקבלת קורסים והמלצות לקורסים.
- 5) המערכת שלנו לא מגבילה את מגוון הקורסים שלה לקורסים השייכים לאתר עצמו בלבד. המלצות יכולים להיות על קורסים מתוך מגוון אתרים ומוסדות.

## 8 ניתוח חלופות מערכת

חלופות תשתית מרכזיות באות לידי ביטוי דרך 3 נושאים עיקריים - מימוש מאגר הקורסים, איסוף מידע דמוגרפי מהמשתמש ובחירת פלטפורמה

### 8.1 מימוש מאגר נתונים על קורסים:

#### 8.1.1 סריקה מילולית אוטומטית של אתרי האינטרנט.

##### תיאור:

לעשות מערכת סריקה טקסטואלית של אתרי אינטרנט ורשתות חברתיות לפי מילות מפתח, כך שנוכל ליצור אצלנו מאגר קורסים המתעדכן אוטומטית. ועיבוד נתונים אלו להשמטת מידע לא רלוונטי או לא נכון טקסטואלית שיהיה אפשר להכניס אותו למאגר נתונים אשר ממנו שואב מידע האלגוריתם.

##### חסרונות:

- אלגוריתם מסובך, גם אם מקבלים את שם הקורס, לא תמיד המידע מדויק.

#### 8.1.2 הכנסת קורסים ע"י מציעי הקורסים (חלופה שנבחרה).

##### תיאור:

מערכת web שלנו תממש ממשק אשר בו מציעי הקורסים (ארגונים שמעבירים את הקורסים) יוכלו להזין את הקורסים עם פרטיהם למערכת. וקורסים אלו יכנסו ל database למען הפעלת אלגוריתם personalise שלנו.

##### יתרונות:

- נתונים על קורסים הם אמינים ככל האפשר.

##### חסרונות:

- נדרש צד שלישי למערכת

#### 8.1.3 החלופה שנבחרה והושמה במערכת שלנו

החלופה שנבחרה והושמה היא דווקא לא אחת החלופות שהוצעו בדוחות הראשוניים, אלא נבחר שהכנסת קורסים חדשים למערכת תיושם ע"י אנשים בעלי הרשאה מתאימה. וזה מתוך שיקולים של מניעת הכנסת מידע לא אמין למערכת.

## 8.2 איסוף פרטים אישיים ממשתמשים מחפשי ההמלצות

### 8.2.1 התחברות והרשמה למערכת דרך פרופיל של רשתות חברתיות.

#### תיאור:

באת הרשמה למערכת יוצע למשתמש להתחבר דרך פרופיל שלו ב facebook או google. וברגע שיאפשר את השימוש בפרטים של פרופיל הנבחר המערכת שלנו תעביר את הנתונים הרלוונטים ל database שלנו.

#### יתרונות:

- מאפשר הרשמה מהירה ונוחה יותר למערכת שלנו

#### חסרונות:

- דורש סינכרון ועבודה עם תוכנה חיצונית

### 8.2.2 הרשמה למערכת דרך מילוי טופס הרשמה

#### תיאור:

כאשר מחפש הקורס רוצה להרשם למערכת שלנו הוא יתבקש למלא טופס הרשמה עם כל הפרטים הרלוונטיים להזדהות ולהפעלת האלגוריתם שלנו, התופס יבדק ע"י מערכת בדיקות אוטומטית לפני אישור הרשמה והכנסת נתונים ל database.

#### יתרונות:

- לא נדרש סינכרון ועבודה עם תוכנה חיצונית.
- מאפשר לקחת פרטים אישיים בלי הגבלות משחקן שאין לנו שליטה עליו.

#### חסרונות:

- הרשמה פחות ידידותית למשתמש.

### 8.2.3 החלופה שנבחרה והושמה במערכת שלנו

החלופה שנבחרה והושמה היא מין שילוב של שני החלופות. ההרשמה למערכת יכולה להתבצע דרך מיייל וסיסמה או דרך חשבון google. וישר לאחר מכן, בהמשך להרשמה מופיע טופס בו המשתמש מחוייב למלא את כל הפרטים.

## 8.3 סוג פלטפורמה

### 8.3.1 הקמת פלטפורמת web

#### תיאור:

שימוש בטכנולוגיית web מסוימת עבור הUI של המערכת כגון: AngularJs או Asp.net או React

#### יתרונות:

- נוח וקל לשימוש, מעוצב באופן ידידותי למשתמש.
- מיועד בעיקר למערכות הפעלה של מחשבים (Linux, Microsoft, ...).

#### חסרונות:

- לא תמיד יש תאימות עם מערכות הפעלה של מכשירי mobile (לדוגמא, Android, IOS, ...)

### 8.3.2 הקמת פלטפורמת Android

#### תיאור:

שימוש בטכנולוגיית Android ליצירת אפליקציית mobile.

#### יתרונות:

- ניידות (זמין מכל מכשיר mobile בעל מערכת הפעלה Android)

#### חסרונות:

- לא זמין במערכות הפעלה אחרות של mobile או מחשב אישי.

### 8.3.3 החלופה שנבחרה והושמה במערכת שלנו

השמנו ממשק משתמש וובי בגלל שתוכן מוצרינו מתאים יותר לממשק וובי. שאלנו אנשים וכולם אמרו שאם יחפשו המלצות לקורסים אז יעשו את זה על מחשב נייד או נייד שלהם.

## 9 דרישות המערכת

### 9.1 דרישות פונקציונליות:

9.1.1 דרישות עבור לקוח אשר מעוניין לקבל המלצה לקורס:

#### 9.1.1.1 הצלבת נתונים בין משתמשים שונים

האלגוריתם יתחשב בקורסים שנלקחו בעבר ע"י משתמשים (collaborative filtering). יהיה תמיד אפשרות להוסיף קורסים לרשימה זו.

#### 9.1.1.2 PERSONALISE אלגוריתם

אלגוריתם יעשה שימוש בפרטי המשתמש (נתונים דמוגרפיים), אשר ימסרו על ידו.

#### 9.1.1.3 ממשק הרשמת משתמש חדש למערכת

ממשק רישום משתמש חדש למערכת יהיה בנוי משני שלבים:  
(א) הרשמה דרך מייל  
(ב) מסירת פרטים דמוגרפיים וקורסים שלמדו בעבר ע"י המשתמש

#### 9.1.1.4 ממשק חידוש פרטי המשתמש

יהיה אפשר להוסיף קורס לקורסים שנלקחו בעבר ולשנות את פרטי המשתמש.

#### 9.1.1.5 ממשק צפיה בקורסים המוצעים למשתמש

אפשר יהיה לראות את המלצות לקורסים דרך ממשק וובי של המערכת.

#### 9.1.1.6 ממשק קישור לאתר של ספק הקורס

במידה ויהיה מוזן האתר של ספק הקורס בתוך פרטי הקורס, יהיה קישור לאתר במסך פרטי הקורס.



## 9.1.2 דרישות עבור ספק הקורס

### 9.1.2.1 ממשק הזנת קורס

מימוש ממשק הזנת קורס הכולל בחירת קריטריונים מסוימים מתוך רשימה, ופירוט הקורס (אופציונלי).

### 9.1.2.2 ממשק שינוי פרטים של ספק הקורסים

ספק יוכל בכל עת לעדכן את פרטיו במערכת.

### 9.1.2.3 ממשק המראה את רשימת הקורסים של ספק הקורסים

ספק יוכל לראות את הקורסים שהכניס בעבר למערכת

## 9.2 דרישות לא פונקציונליות

### 9.2.1 דרישות עיצוב, דרישות שימוש וממשק משתמש

#### 9.2.1.1 ממשק ידידותי למשתמש

הדיאלוג של המערכת עם המשתמש יהיה בשפה מובנת וחד משמעית (ל 90% של משתמשים).

#### 9.2.1.2 עיצוב אחיד

שני הממשקים הן של מחפש הקורס והן של ספק הקורס ישתמשו בבסיס עיצוב זהה. לדוגמה: צבעים במערכת, נראות של פקדים. מתן אינדיקציה בזמן טעינה או הכנה של נתונים על פי כללי העיצוב המקובלים בתעשייה. לדוגמה: בזמן טעינת נתונים במידה והמערכת לא זמינה לביצוע פעולות יוצג loader waiting.

#### 9.2.1.3 פעולות ברורות למשתמש

כל הפעולות יהיו נוחות וקלות להבנה למשתמש. ז"א שמשתמשים יוכלו להשלים כ 90% מהפעולות האפשריות במערכת ללא עזרה.

## 9.2.2 דרישות ביצועים

9.2.2.1 מינימום של 80% הצלחה בהצגת קורסים המתאימים למשתמש

9.2.2.2 התאמות לקורסים יתעדכנו בזמן אמת

## 9.2.3 דרישות הפעלה

9.2.3.1 ממשק קל ומהיר

ממשק WEB לחיפוש קורס קל ומהיר (עד 5 לחיצות אחרי הרשמה).

9.2.3.2 השרת יחזק עם ה data ישבו על ענן.

## 9.2.4 דרישות תחזוקה ותמיכה

9.2.4.1 עדכון מערכת Web לא דורש פגיעה בתפקוד עמדות משתמשי הקצה.

9.2.4.2 המערכת תתפקד בצורה זהה בדפדפנים הסטנדרטיים.

## 9.2.5 דרישות אבטחה

9.2.5.1 ניהול משתמשים במערכת יתאפשר ע"י מנהל מערכת בלבד.

9.2.5.2 רישום מחפש הקורס, ומספק הקורס לאתר יתאפשר עם מייל תקין בלבד.

## 9.2.6 זמינות

9.2.6.1 המערכת תהיה זמינה ב 95% מהזמן כלומר שירות רציף.

# 10 אפיון המערכת

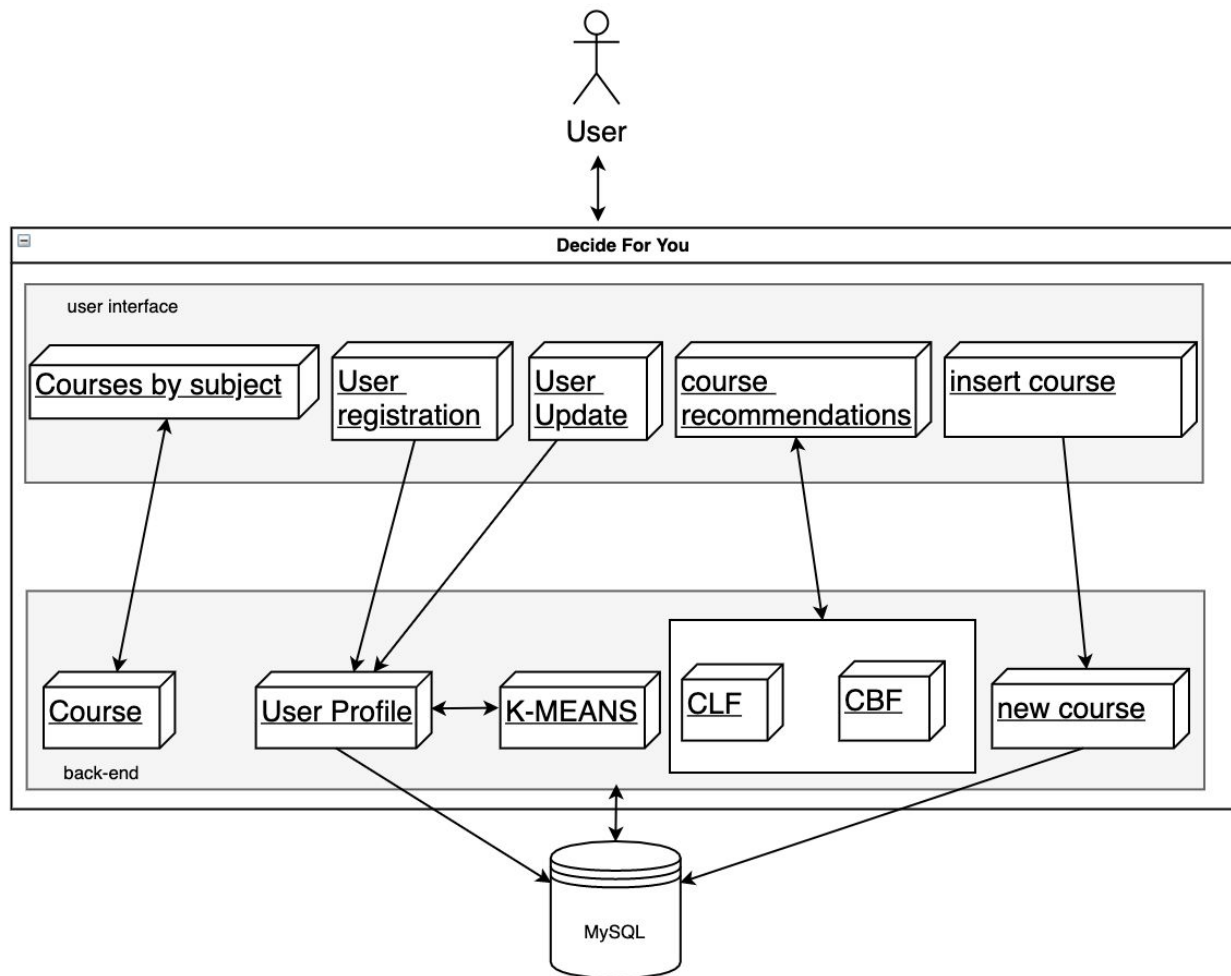
"מתלבט בשבילך" נבנה ארכיטקטורת של client - server. כאשר בצד המשתמש יהיה הדפדפן וובי שעליו יוצג הממשק (SRD- 2.3), ודרכו המשתמש יוכל לבקש המלצות לקורסים מהשרת או לראות קורסים לפי נושא:

קורסים לפי נושא יוכל משתמש לראות בכל אתר מאז רישום למערכת.

אופציה של המלצות לקורסים ע"י שילוב של 3 אלגוריתמי למידת מכונה (סינון משותף, סינון מבוסס תוכן ויצירת אשכולים) יתאפשר רק לאחר מילוי פרטים דמוגרפיים ולפחות קורס אחד שנלקח בעבר ע"י המשתמש וקיים במערכת.

השרת יפעיל את האלגוריתמים המתאימים, לפי בקשות ממשק משתמש. במהלך ריצת האלגוריתם ישלפו נתונים מבסיס נתונים MySQL.

## 10.1 תרשים בלוקים



### איור 1.6 - תרשים בלוקים של כל המערכת

**Courses by subject** - ממשק משתמש המאפשר למשתמש לקבל רשימת קורסים לפי נושא שהוא בוחר.

**Course** - מחלקה בשרת שמטפלת בבקשות של הוצאת קורסים לפי נושא. שולפת את הנתונים הנדרשים מ MySQL.

**User registration** - ממשק משתמש לרישום משתמש חדש

**User Update** - ממשק משתמש לעדכון משתמש קיים

**User profile** - מחלקה של השרת שמטפלת בפרופילים של משתמשים.

**K-MEANS** - מחלקה אשר אחראית על קיטלוג של המשתמש לאחד הקבוצות הדמוגרפיות הקיימות במערכת, ומעדכנת את הקיטלוג ב MySQL. בשביל קיטלוג המשתמש מחלקה זו משתמשת בנתונים שב MySQL.

**Course Recommendation** - ממשק לבקשת המלצות לקורסים און ליין.

**CLF** - מחלקה האחראית על הוצאת המלצות לקורסים דרך סינון שיתופי. מחלקה משתמשת בנתונים מ MySQL.

**CLB** - מחלקה האחראית על הוצאת המלצות לקורסים דרך סינון מבוסס תוכן. מחלקה משתמשת בנתונים מ MySQL.

**Insert course** - ממשק משתמש שדרכו אפשר להזין קורסים חדשים לתוך המערכת.

**New course** - מחלקה בשרת שמטפלת בהכנסת קורסים חדשים ל MySQL.

## 11 ניתוח חלופות טכנולוגיות.

### 11.1 בחירת מאגר נתונים חיצוני

היה צריך לבחור באחד ממאגרי נתונים חיצוניים בין אם הוא רלציוני או לא. בחרו במאגר נתונים MySQL שהוא מסד נתונים רלציוני, רב נימי ורב משתמשים מבוסס שפת SQL (Structured Query Language).

## 12 תכן המערכת

### 12.1 ארכיטקטורת המערכת

הארכיטקטורה שנבחרה היא client-server צד המשתמש (דפדפן) וצד השרת.

#### 12.1.1 בצד המשתמש יוכל המשתמש לעשות את הדברים הבאים:

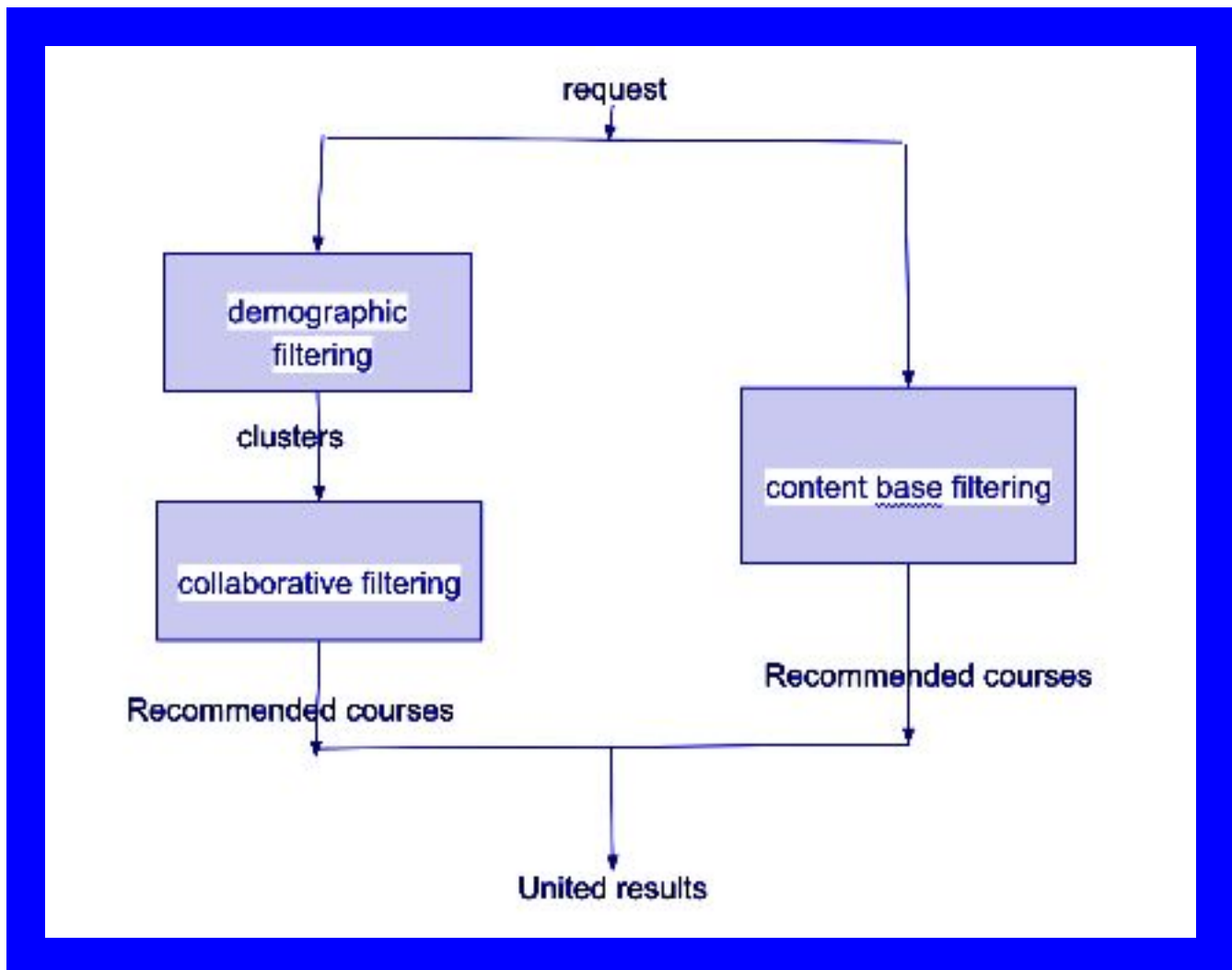
- רישום משתמש חדש למערכת ע"י מייל וסיסמה או חשבון גוגל.
- הכנסת פרטים דמוגרפיים וקורסים שנלקחו בעבר ע"י המשתמש.
- שינוי פרטים דמוגרפיים וקורסים שנלקחו בעבר ע"י המשתמש.
- אפשרות לקבלת קורסים לפי נושא.

- אפשרות קבלת המלצות לקורסים.
- משתמש עם הרשאות מתאימות יוכל להכניס קורסים חדשים למערכת.

## 12.1.2 כל הלוגיקה של המערכת תתבצע בצד השרת:

- הוצאת קורסים לפי נושא ממסד נתונים MySQL.
- הכנסת נתונים דמוגרפיים ומידע על קורסים שנלמדו בעבר ע"י משתמש, באת רישום משתמש למערכת או שינוי פרטים של משתמש, לתוך מסד נתונים MySQL. וזה לאחר עיבוד הנתונים כך שיתאימו לריצת האלגוריתמים.
- הרצת אלגוריתם למידה היברידי מקורי למען הוצאת המלצות קורסים למשתמש.

### 12.1.3 האלגוריתם "ההיברידי" של כל המערכת



איור 1.7 - תרשים זרימה בין חלקי האלגוריתם

#### 12.1.3.1 יחסים בין חלקי האלגוריתמים:

סינון שיתופי וסינון מבוסס תוכן מתבצעים בהפרדה מוחלטת, כל אחד יש לו היבט משלו, והם משלימים את החולשות אחד של השני. סינון דמוגרפי מהווה שלב מקדים וסינון נתונים, עבור סינון שיתופי.

האלגוריתם ההיברידי מתחיל בשיוך משתמשים למקבצים על פי נתונים דמוגרפיים, עוד בעת הרשמתם למערכת. תהליך זה בעצם משמש כפילטר לחלק של סינון שיתופי. כאשר סינון שיתופי יקח רק נתונים מהמקבץ שאליו שייך המשתמש.

כאשר נכנסת הבקשה להמלצות קורסים יודעת מאיזה משתמש היא באה, לפי זיהוי פרופיל משתמש בשכבת ה UI. כמו שאמרנו קודם סינון שיתופי אקח את המידע הרלוונטי למקבץ של המשתמש (מידע על משתמשים אחרים במקבץ של מבקש ההמלצות). ויוצאי המלצות משלו. כל מה שתואר עד עכשיו זה צד אחד של הוצאת המלצות של האלגוריתם ההיברידי. צד השני זהו צד שבא להוציא המלצות לקורסים בהיבט שונה לגמרי מהצד הראשון, ובאצם מהווה השלמה לצד הראשון. צד זה משתמש בנתונים של המשתמש עצמו וגם נתונים על כל הקורסים במערכת, כמובן אשר נלקחים ממאגר נתונים נפרד. ומוציא את המלצותיו לקורסים. בסוף האלגוריתם כאשר שני הצדדים הוציאו כל אחד את ההמלצות שלו, התוצאות מתאחדות ומועברות לממשק משתמש.

### 12.1.3.2 סינון דמוגרפי

#### 12.1.3.2.1 הכנת נתונים

מצאנו ברשת dataset "משתמש-קורס" שבו יש מעל 600,000 שורות. כאשר כל שורה כוללת את מספר משתמש, מספר קורס אחד שלקח, שנת לידה, רמת השכלה, ארץ מגורים, מגדר וציון בקורס. כשביל להכניס נתונים דמוגרפיים כגון שנת לידה, רמת השכלה, ארץ מגורים ומגדר היינו צריכים להפוך נתונים אלו לנתונים מספריים.

שנת לידה הפכנו לגיל, נרמלו את הגיל ע"י הנוסחה הבאה:

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

איור 1.8

בעמודת רמת השכלה היו 5 דרגות השכלה. אשר אותם הפכנו למספרים בדרגות של 0.25. כאשר הם רצים בין 0 ל 1. כאשר 0 זו רמה הכי נמוכה של השכלה ו 1 הכי גבוהה.



בשביל שנוכל להשתמש בנתון "ארץ מגורים", לקחנו מהרשת dataset של מדינות וקווי אורך ורוחב שלהם. ובמקום עמודת ארץ מגורים הכנסנו שני עמודות אחת של קו רוחב והשנייה של קו אורך. ונרמלנו אותם באותה שיטה שהראנו קודם. בעמודת מגדר הפכנו גבר ל 0 ואישה ל 1 ובמקומות שלא צויין המגדר סמנו 0.5.

### 12.1.3.2.2 יישום האלגוריתם

מ dataset שהכנו לקחנו נתונים מעמודות של גיל, מגדר, רמת השכלה, קו אורך וקו רוחב. כל עמודה מקבלת משקולת שונה לפני ריצת האלגוריתם. ומכל שורה בעמודות אלו יצרנו וקטור. ואז על הוקטורים שיצרנו הפעלנו אלגוריתם K-MEANS מבוזר של SPARK ליצירת מקבצים של משתמשים. הוספנו ל dataset עמודה של מספר קבוצה ושמרנו אותו. בעזרת עמודה זו נדע איזה משתמש שייך לאיזו קבוצה, נשתמש בנתון זה ב collaborative\_filtering.

### 12.1.3.3 סינון שיתופי

#### 12.1.3.3.1 הכנת נתונים

אלגוריתם זה משתמש ב dataset אשר נוצר אחרי demographic filtering, ונתונים מ dataset של משתמש-קורס.

#### 3.1.3.2 יישום האלגוריתם

מוצאים ממאגר נתונים את כל המשתמשים עם הקורסים שעשו בעבר, השייכים לאותה קבוצה דמוגרפית של משתמש מבקש ההמלצות. בונים "מטריצת משתמש קורס" ואז בעזרת מטריצה זו אלגוריתם מסנן משתמשים שלקחו את כל הקורסים שמשתמש מבקש ההמלצות לקח. ולאחר זאת אם משתמשים אלו לקחו עוד קורסים מעבר, אז אלגוריתם מוציא את הקורסים שמעבר כהמלצות.

### 12.1.3.4 סינון מבוסס תוכן

#### 12.1.3.4.1 הכנת נתונים

ברשת מצאנו dataset של קורסים הכולל 123 קורסים התואם את ה dataset של "משתמש - קורס" אשר השתמשו בו ב collaborative\_filtering ו demographic\_filtering. ובו עמודות כגון:

מספר קורס, שם הקורס, מכון שבו מועבר הקורס, נושא הקורס ועוד מספר עמודות עם נתונים סטטיסטיים על הקורס.

יצרנו dataset חדש כאשר העברנו אליו את העמודות של מספר קורס ושם הקורס. וגם הוספנו עמודה של תיאור קורס, שבה יהיו תיאורים של כל הקורסים הנמצאים במערכת. את התיאורים של הקורסים מצאנו אחד אחד ברשת דרך מספר קורס ייחודי, והוספנו אותם ל dataset החדש.

## 12.1.3.4.2 יישום האלגוריתם

1. מה dataset שהכנו לקחנו את העמודה שבא תיאורים של כל הקורסים, והפכנו אותה לרשימה של תיאורי קורס.
2. מרשימה של תיאורי כל הקורסים יצרנו רשימה של tokens. ההפרדה בין תיאורי הקורס ברשימה נשמרה גם לגבי ה tokens, כל תיאור וה tokens שלו.
3. מרשימת tokens של כל התיאורים יצרנו dictionary אשר ממפה מילה לאינדקס ייחודי שלה, ובכך יצרנו מאגר מילים שאלגוריתם ישתמש בו. כדי לעבוד על מסמכי טקסט, Gensim דורש שהמילים (אסימונים) יומרו למזהים ייחודיים.
4. מפה המשכנו ויצרנו corpus שהוא בעצם אובייקט שמכיל את אינדקס המילה בתוך מאגר מילים שיצרנו בשלב 3 ואת התדירות שלה בתיאור אחד. זאת אומרת יש הפרדה בין תיאורים בתוך ה corpus.
5. בעזרת ה corpus יצרנו את TF-IDF. TF-IDF הוא גם מודל שקית-מילים אך בשונה ה corpus הרגיל, נותן משקל למטה למילים שמופיעים לעיתים קרובות על פני כל תיאורי הקורסים. בעצם מחשב משקולות של מילים: (מספר מופעי מילה בתיאור אחד חלקי מספר מילים באותו התיאור) כפול (לוג של מספר התיאורים שבה מופיע מילה חלקי מספר כל התיאורים)
6. בהמשך השתמשנו ב TF-IDF כדי ליצור אובייקט similarity, שישמש אותנו בהמשך כדי לראות כיצד שני אובייקטים קשורים זה לזה. כאשר יצרנו את אובייקט ה similarity בחרנו להשתמש בשיטת מדידה cosine.

## 12.1.3.5 איחוד תוצאות אחרי ריצת סינון שיתופי וסינון מבוסס תוכן

מאלגוריתם מבוסס תוכן לקחנו עד 2 קורסים הכי דומים בתוכן הקורס לכל קורס שלקח מבקש ההמלצות בעבר (אם התוצאות חופפות אז יש פחות משתי המלצות על כל קורס). מההמלצות של אלגוריתם סינון שיתופי לקחנו עד 5 קורסים (יכול להיות שיהיה פחות מ 5 המלצות) הכי פופולריים בקבוצה הדמוגרפית של מבקש ההמלצות. אם קורס הופיע בהמלצות של שני האלגוריתמים הוא יקודם לראש הרשימה.

## 12.2 חלופות לתכן המערכת

### 12.2.1 כמות קורסים בהוצאת המלצות

הייתה ההתלבטות של כמה קורסים להוציא למשתמש. בסוף הוחלט כמו ב 12.1.3.5. מצד אחד כמה שנביא יותר קורסים אז תוצאות בדיקה יהיו יותר טובים. למשל אם היינו מביאים 9 קורסים מסינון שיתופי הינו מקבלים 100 אחוז פגיע בולידציה. אבל בכל זאת החלטנו להגביל את כמות המלצות של הקורסים מסינון שיתופי ל 5 קורסים בלבד שהכי פופולריים בקבוצה הדמוגרפית של מבקש ההמלצות מ2 סיבות:

1. בגלל שבסט נתונים שהצלחנו להשיג על משתמשים וקורסים שהם לקחו הופיעו רק 13 קורסים. אז כמה שנעלה את כמות ההמלצות מסינון שיתופי כך ערך תוצאות הבדיקה היה מאבד את הנכונות שלו וליגליות שלו בעולם האמיתי כשיש מבחר גדול בהרבה יותר של קורסים. האיזון שמצאנו להכי נכון הוא 5 קורסים ולא יותר.
2. מטרת הוצאת ההמלצות היא לצמצם משמעותית את כמות הקורסים שמשתמש צריך לשקול לקחת. ולכן אסור בשום אופן להעמיס על משתמש בכמות ההמלצות אם ברצונו שאפליקציה זו תהיה רלוונטית למשתמשיה.

מסינון מבוסס תוכן החלטנו גם להגביל בהמלצות, ועל כל קורס שנלמד בעבר ע"י מבקש ההמלצות הבאנו 2 קורסים. מצד אחד לא נרצה להעמיס בהמלצות את המשתמש כמו שהוסבר בפסקה הקודמת (סיבה 2). אבל מצד שני אם מעבר של מבקש ההמלצות משתמע שהוא מעוניין במבחר נושאים רחב נרצה להמליץ לו מכל נושא שלמד בעבר לפחות 2 קורסים באותו נושא.

## 12.4 הרציונל מאחורי שילוב של אלגוריתמים לאלגוריתם היברידי אחד של למידת מכונה מקורי של "מתלבט בשבילך"

- מערכת של "מתלבט בשבילך" משתמשת ומשלבת 3 אלגוריתמים של למידת מכונה:
1. יצירת קבוצות דמוגרפיות ע"י הפעלת אלגוריתם K-MEANS ליצירת קבוצות דמוגרפיות של משתמשי המערכת (למידה לא מפוקחת). חילוק לקבוצות בא ולחדד את התוצאות של אלגוריתם סינון שיתופי. מפני שסינון שיתופי משתמש במידע על אנשים אחרים במערכת אם נתחשב רק באנשים שדומים למחפש ההמלצות מבחינת נתונים דמוגרפיים בסיסיים זה ישפר את נכונות תוצאות ההמלצות שיוציא האלגוריתם. במיוחד גם כאשר משתמשים במשקולות על

- כל סוג נתון ( הסבר ב 12.2.3). נכונות הוספת חלק זה לאלגוריתם היברידי נבדק כאשר הורץ אלגוריתם בלי חלק זה, וכאשר הורץ אלגוריתם עם חלק זה וכיול משקולות. הבדיקות האלו שתוצאות האלגוריתם ההיברידי משתפרות כב 5 אחוז כאשר משתמשים ביצירת מקבצים ומריצים את הסינון השיתופי רק על קבוצה עליו שייך מחפש ההמלצות. בנוסף שימוש בקבוצות דמוגרפיות מעניק בונוס של ביצועים כאשר מריצים אלגוריתם סינון שיתופי הוא רץ מהר יותר כי הוא מתחשב בפחות אנשים.
2. יצירת המלצות דרך אלגוריתם סינון שיתופי ( הסבר ב 7.1.3.3). מכיוון שאם משתמשים אחרים הקיימים במערכת לקחו את כל אותם קורסים שמבקש ההמלצות לקח, זה מצביע על כך שרוב הסיכויים הם הולכים באותו מסלול למידה. מכיוון שיש להם קורסים מעבר לקורסים של מחפש ההמלצות הם התקדמו יותר במסלול למידה של מחפש ההמלצות. אז רק הגיוני שנמליץ על מסלולים של משתמשים אלו.
3. יצירת המלצות דרך אלגוריתם סינון מבוסס תוכן ( הסבר ב 7.1.3.4). חלק זה באלגוריתם היברידי של המערכת בא להשלים את החולשות של סינון שיתופי. חלק זה מאוד חשוב ואף קריטי למערכת המלצות של קורסים. במיוחד למשתמשים שלוקחים קורסים און ליין. תובנה זו באה מתוך סט נתונים של MIT ו HARVARD אמין ואמיתי של אנשים וקורסים שהם למדו בעבר און ליין. מכיוון ששם אפשר לראות שאנשים אלו לקחו קורסים שהם חלק באיזשהו מסלול או נושא מסויים אבל ממש לא רק. חלק ניכר מאנשים אלו לקחו קורסים שאין נראה לעין שום קשר ביניהם לא בנושא ולא במסלול למידה. זאת אומרת שמי שלוקח קורסים און ליין הרבה פעמים ילמד קורסים מכל מיני נושאים שמעניינים אותו באופן כללי, להבדיל אם היו לומדים במסגרת למידה מסודרת כלשהי. ולכן אם הם עשו קורסים לידע כללי או מתוך עניין אישי בנושא רוב הסיכויים שיהיו מעניינים בקורסים עם תוכן דומה.

## 13 התוצר

"מתלבט בשבילך" הוא אפליקציה ברשת, אשר מספק המלצות לקורסים שאפשר לקחת און ליין בהתאמה אישית. וקיימת גם אופציה לראות איזה קורסים קיימים בנושא שמשתמש מתעניין בו. גרסה זו היא גרסת BETA אשר מוכנה לגמרי לשימוש ולריצה על סרבר. גרסה זו בשלב זה תופעל על סרבר ענן של AWS. כאשר המערכת תרוץ על כמה מכונות וירטואליות להגברת זמן ריצת האלגוריתם. המערכת תשתמש במאגר נתונים של MySQL. מערכת זו פותחה בשפת PYTHON וכוללת בתוכה את כל הספריות הנדרשות לפרויקט. מערכת גם משתמשת בערכת כלים ליישומי ממשק שרת אינטרנט בשם FLASK. וחלק מהמערכת נעזרת בספריית PYSPARK אשר מאפשרת הרצה מבוזרת.

## 13.1 תוצרים שפותחו

### 13.1.1 אלגוריתמים שפותחו

1. אלגוריתם להמלצות של למידת קורסים און ליין, היברידי בהתאמה אישית.
2. עיבוד והמרת נתונים למען הרצות של אלגוריתמים
3. יצירת אלגוריתמי בדיקה של כל האלגוריתמים

### 13.1.2 שירותים שפותחו

1. הוצאת קורסים לפי נושא.
2. הכנסת קורסים חדשים למערכת
3. רישום משתמשים חדשים
4. הכנסת פרטים דמוגרפיים
5. עדכון פרטים דמוגרפיים
6. בדיקת הרשאות
7. הוצאות והכנסות של נתונים למאגר מידע חיצוני
8. רישום משתמש חדש באמצעות חשבון גוגל
9. עיבוד והמרת נתונים למען הרצות של אלגוריתמים

## 13.1 אלגוריתם להמלצות למידת קורסים און ליין, היברידי בהתאמה אישית

זהו תוצר מרכזי של המערכת. כל אחד שמעוניין ללמוד קורסים און ליין יכול לבקש המלצות על קורסים המתאימים לו אישית ללמידה. תוצר זה משתמש באלגוריתם K-MEANS של SPARK שכבר קיים, אלגוריתם סינון שיתופי אשר נבנה כולו על ידינו ואלגוריתם של סינון מבוסס תוכן שגם נבנה כולו על ידינו חוץ מחלק של חישוב דמיון קוסינוסי. אלגוריתם זה משתמש במאגר נתונים חיצוני MySQL ובחלקו בספריית PYSPARK לריצה מבוזרת של האלגוריתם.

### 13.1.1 אופן בדיקת האלגוריתם

לקחנו 2 מערכי נתונים של MIT ו Harvard University ביחד. מערך ראשון הוא בעל יותר מ 600 אלף שורות של משתמשים, קורסים שלקחו ונתונים דמוגרפיים שלהם. ועוד מערך נתונים שאספנו לבד על 120 קורסים ממוסדות אלו, הכולל מספר קורס, מרצה, נושאים בקורס ותיאור הקורס.

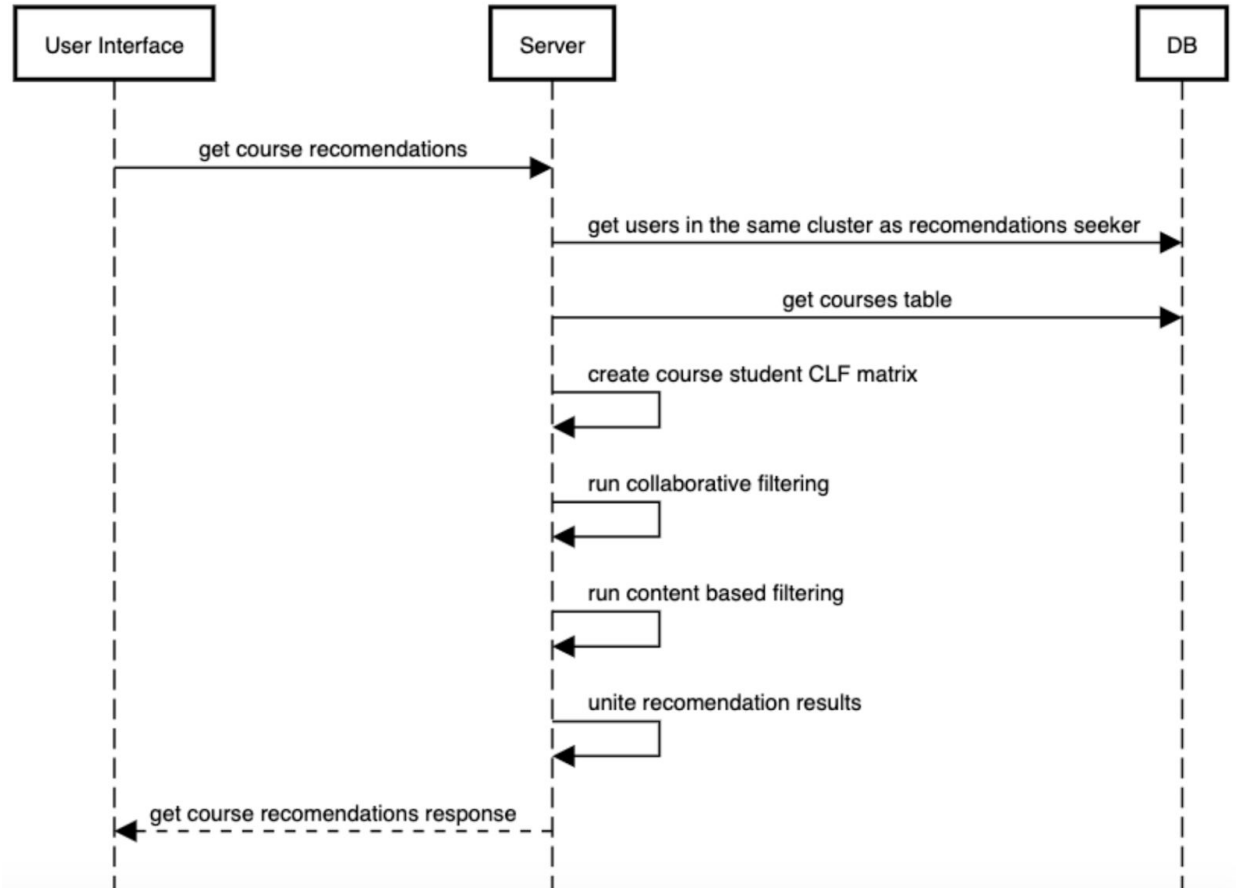
חילקנו את מערך נתוני משתמשים שלקחו קורסים ל 80% נתוני אימון, 10% נתוני ולידציה ו 10% נתוני בדיקה.

את הוולידציה לקחנו כל פעם 10% אחרים (סה"כ 9 חלקים שונים שווים). בכל קטע הרצנו על כל המשתמשים שבולידציה את האלגוריתם המלצות ההיברידי של המערכת שלנו, ובדקנו האם בקורסים שמוציא האלגוריתם מתוך נתוני אימון יש את כל הקורסים שלקחו המשתמשים בנתוני הוולידציה. ממוצע של כל חלקי הוולידציה היה 85.63% של פגיע בקורסים אלו בכלל משתמשים.

### 13.1.2 תסריט הפעלה של קבלת המלצות לקורסים בהתאמה אישית

1. לוחצים על כפתור בקשת המלצות בממשק משתמש
2. בקשה עוברת לסרבר
3. סרבר מזהה את המשתמש
4. סרבר מוציא נתונים דמוגרפיים של משתמשים השייכים לאותה קבוצה דמוגרפית של מחפש ההמלצות
5. סרבר בונה מטריצת משתמש קורס
6. סרבר מוציא את כל טבלת הקורסים
7. סרבר מריץ סינון שיתופי
8. סרבר מריץ סינון מבוסס תוכן
9. סרבר מאחד תוצאות
10. סרבר שולח תוצאות חזרה לממשק משתמש
11. ממשק משתמש מראה רשימה של קורסים מומלצים עם פרטיהם ותיאורם

## קבלת המלצות לקורסים בהתאמה אישית



### איור 1.9 - תרשים זרימה לקבלת המלצות של קורסים בהתאמה אישית

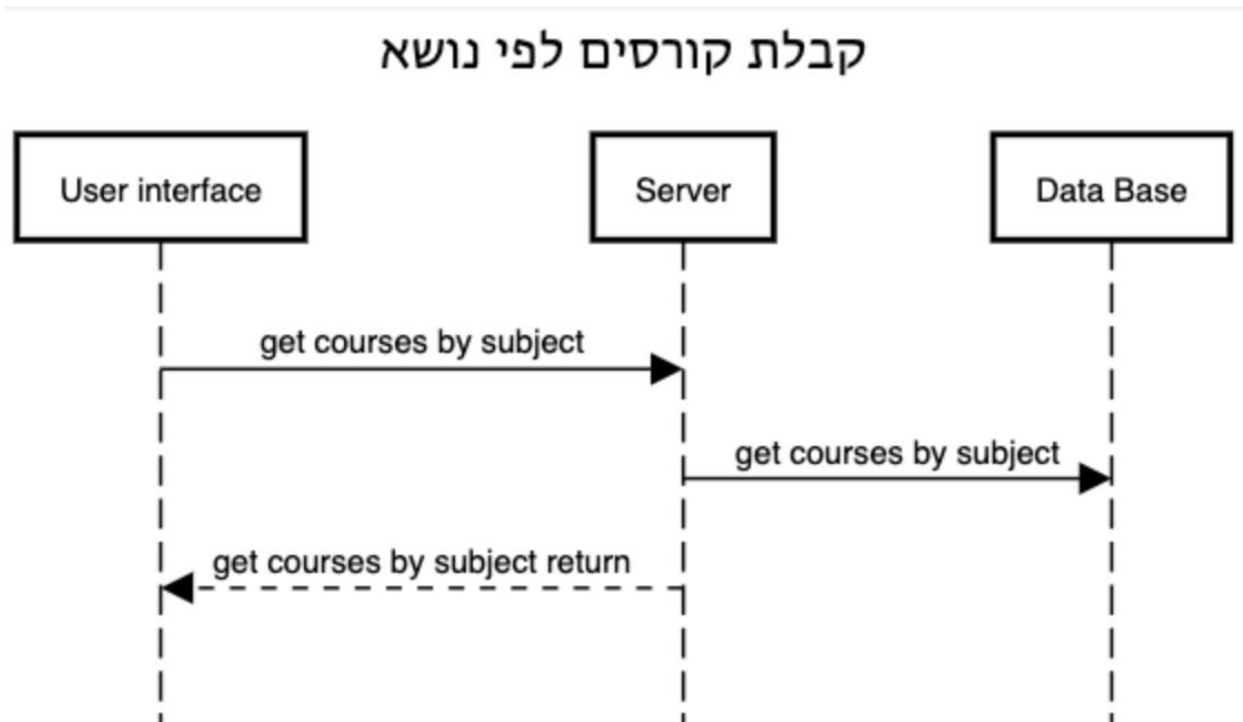
## 13.2 הוצאת קורסים לפי נושא

תוצר צדדי למשתמשים שרק נרשמו למערכת ועוד לא הכניסו פרטים דמוגרפיים וקורסים שלקחו בעבר, או למשתמשים שרוצים לראות קורסים שקיימים בנושא מסוים.

### 13.2.1 תסריט הפעלה של קבלת קורסים לפי נושא

1. בוחרים נושא מתוך רשימת נושאים המופיעים בקומבו בוקס.
2. לוחצים על כפתור קבלת קורסים לפי נושא.

3. בקשה עוברת לסרבר
4. הסרבר מוציא את הקורסים עם הפרטים שלהם בנושא המדובר ממאגר נתונים חיצוני
5. הסרבר מחזיר את הקורסים עם הפרטים שלהם לממשק משתמש
6. ממשק משתמש מראה רשימה של קורסים מומלצים עם פרטיהם ותיאורם



**איור 1.10 - תרשים זרימה לקבלת קורסים לפי נושא**

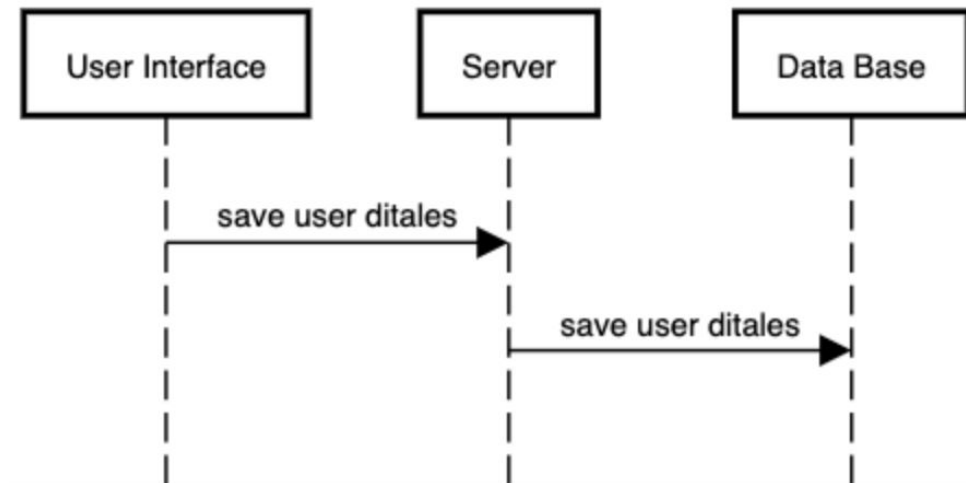
## 13.3 רישום משתמש חדש למערכת באמצעות מייל

### 13.3.1 תסריט הפעלה

1. המשתמש לוחץ על כפתור Login/Sign Up
2. בטופס שנפתח, המשתמש לוחץ על Sign up new account וממלא את הפרטים: אימייל סיסמא ושם משתמש. מייל חייב להיות תקני וסיסמה בת 8 תווים לפחות.
3. משתמש לוחץ על כפתור Sign Up לסיים הרשמה.
4. משתמש מועבר לטופס עדכון פרטים דמוגרפיים.



## רישום משתמש חדש למערכת באמצעות מייל



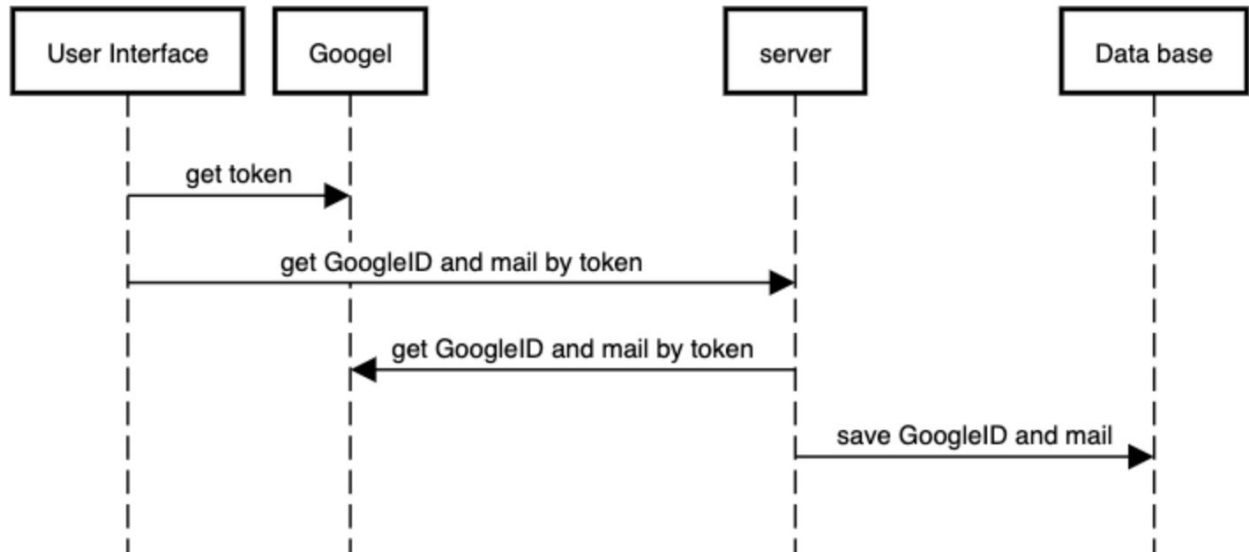
איור 1.11 - תרשים זרימה לרישום משתמש חדש באמצעות מייל

## 13.4 רישום משתמש חדש למערכת באמצעות חשבון גוגל

### 13.4.1 תסריט הפעלה

1. המשתמש לוחץ על כפתור Login/Sign Up
2. בטופס שנפתח, המשתמש לוחץ על כפתור Sign In With Google
3. נפתח חלון נוסף, בו המשתמש מתחבר לחשבון הגוגל שלו (באמצעות אימייל וסיסמא) ומיד לאחר ההתחברות מועבר לדף הראשי של האתר.

## רישום משתמש חדש למערכת באמצעות חשבון גוגל



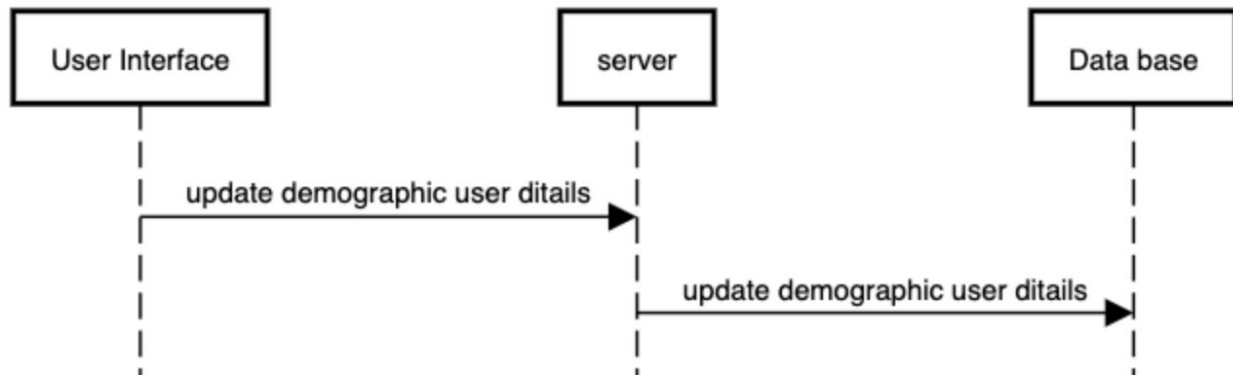
איור 1.12 - תרשים זרימה לרישום משתמש חדש באמצעות חשבון גוגל

## 13.5 עדכון פרטים דמוגרפיים של משתמש

### 13.5.1 תסריט הפעלה

1. במידה ומשתמש רשום טרם עדכן את פרטיו הדמוגרפיים, נפתח חלון עם שדות שאותם הוא צריך למלא בעת הכניסה למערכת או לחילופין אפשר לפתוח חלון זה ע"י לחיצה על שם המשתמש מצד שמאל של המסך, ומהרשימה שנפתחת ללחוץ על Profile. את הפרטים הללו חובה למלא לפני בקשת המלצות מהמערכת.
2. המשתמש ממלא פרטים על ידי בחירת אופציות מרשימות הבחירה להלן:  
מין, ארץ, שנת לידה, השכלה
3. בסוף לוחץ על כפתור עדכון. הפרטים נשלחים לשרת לצורך עדכוןם וגם לצורך חישוב קלסטר חדש עבור המשתמש.

## עדכון פרטים דמוגרפיים של משתמש



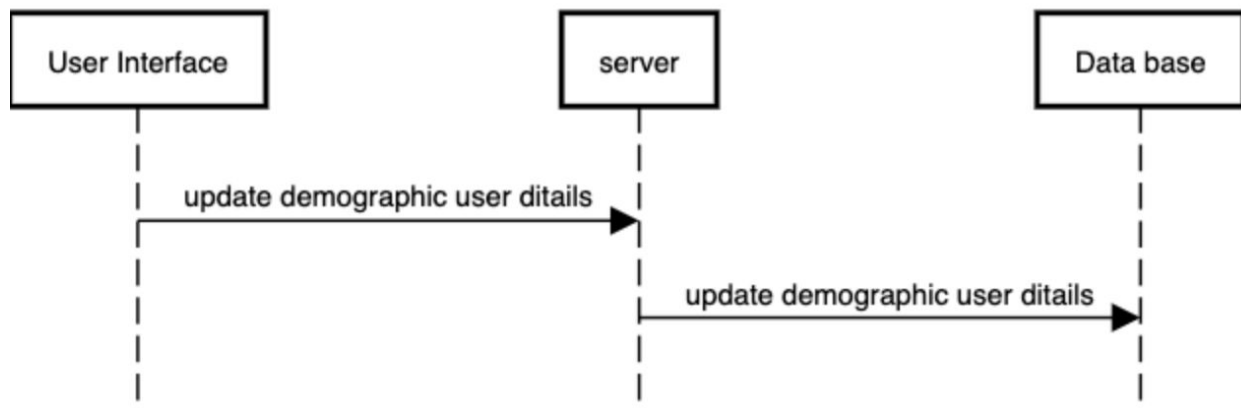
איור 1.13 - תרשים זרימה של עדכון פרטים דמוגרפיים של משתמש

## 13.6 עדכון קורסים שנלקחו ע"י משתמש

### 13.6.1 תסריט הפעלה

1. עדכון הקורסים נעשה מתוך חלון Profile עבור משתמש רשום. חובה למלא לפחות קורס אחד לפני בקשת המלצות מהמערכת.
2. המשתמש בוחר את הקורסים שהוא לקח בעבר מתוך רשימת כל הקורסים שקיימים במערכת.
3. לאחר מכאן לוחץ על כפתור עדכון, והקורסים נשלחים לשרת לצורך עדכוןם ב-DB.

### עדכון קורסים שנלקחו ע"י משתמש

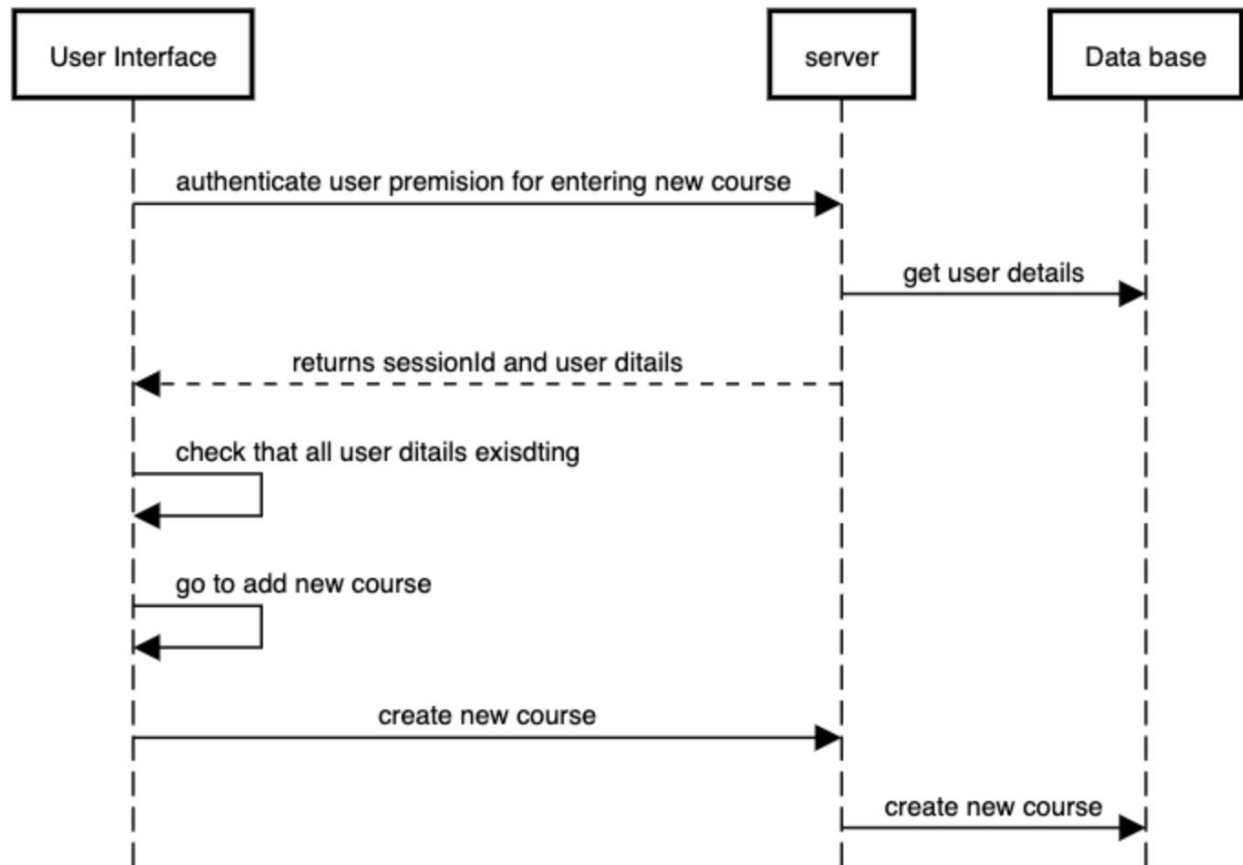


## 13.7 הכנסת קורס חדש ע"י משתמש מורשה

### 13.7.1 תסריט הפעלה

1. משתמש אשר יש לו הרשאות מיוחדות יכול להוסיף קורס למערכת.
2. לאחר התחברות, לחיצה על שם המשתמש למעלה משמאל ואז על כפתור Profile.
3. בחלון שנפתח, יש לוודא שהמשתמש נמצא על לשונית Add Course.
4. המשתמש ממלא את הפרטים הבאים:
  - a. שם הקורס
  - b. תיאור מפורט של הקורס
  - c. נושאי הקורס (עד 3 נושאים)
  - d. המוסד/האתר בו הקורס מתקיים
  - e. לינק לאתר האינטרנט של הקורס
5. לאחר לחיצה על כפתור Add Course, פרטי הקורס החדש נשלחים לשרת ונשמרים שם לצורך בקרה של מנהלי האתר, על מנת שבהמשך הקורס יהווה חלק מהמערכת.

## הכנסת קורס חדש ע"י משתמש מורשה

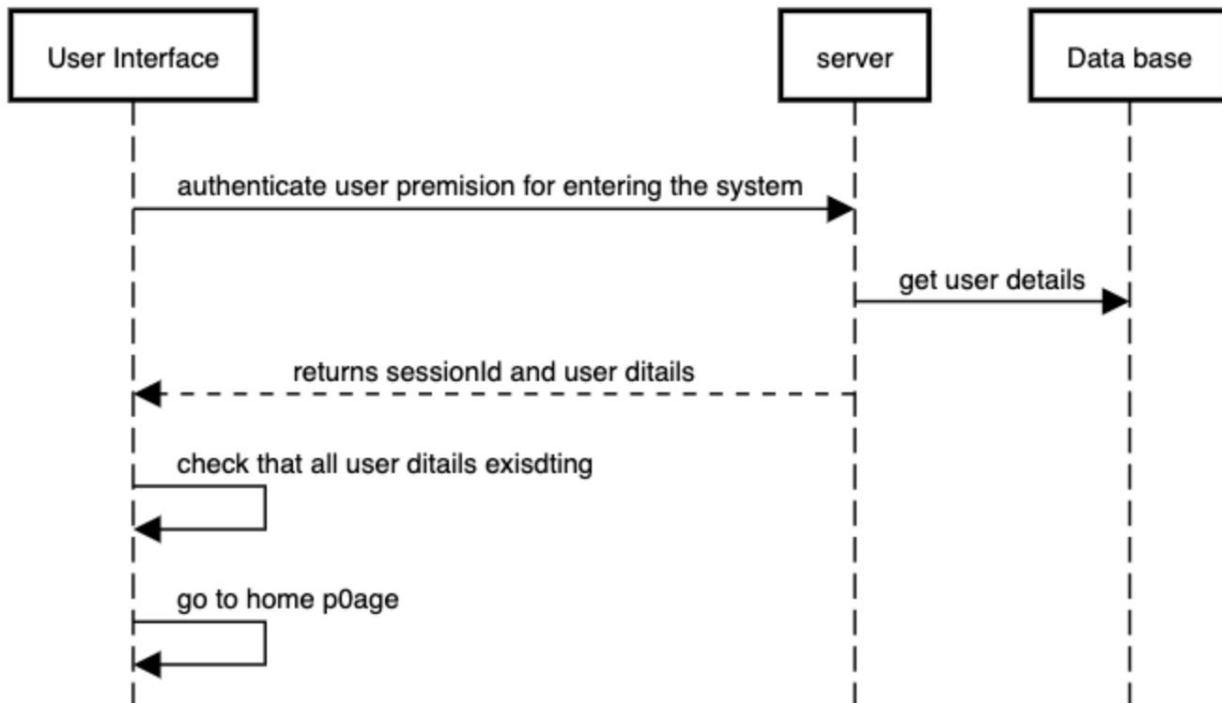


איור 1.15 - תרשים זרימה של הכנסת קורס חדש ע"י משתמש מורשה

## 13.8 כניסה למערכת באמצעות מייל וסיסמה

1. המשתמש לוחץ על כפתור Login
2. בטופס שנפתח, המשתמש לוחץ על כפתור Sign In
3. נפתח חלון נוסף, בו המשתמש מתחבר לחשבון הגוגל שלו (באמצעות אימייל וסיסמא) ומיד לאחר ההתחברות מועבר לדף הראשי של האתר.

## כניסה למערכת באמצעות מייל וסיסמה

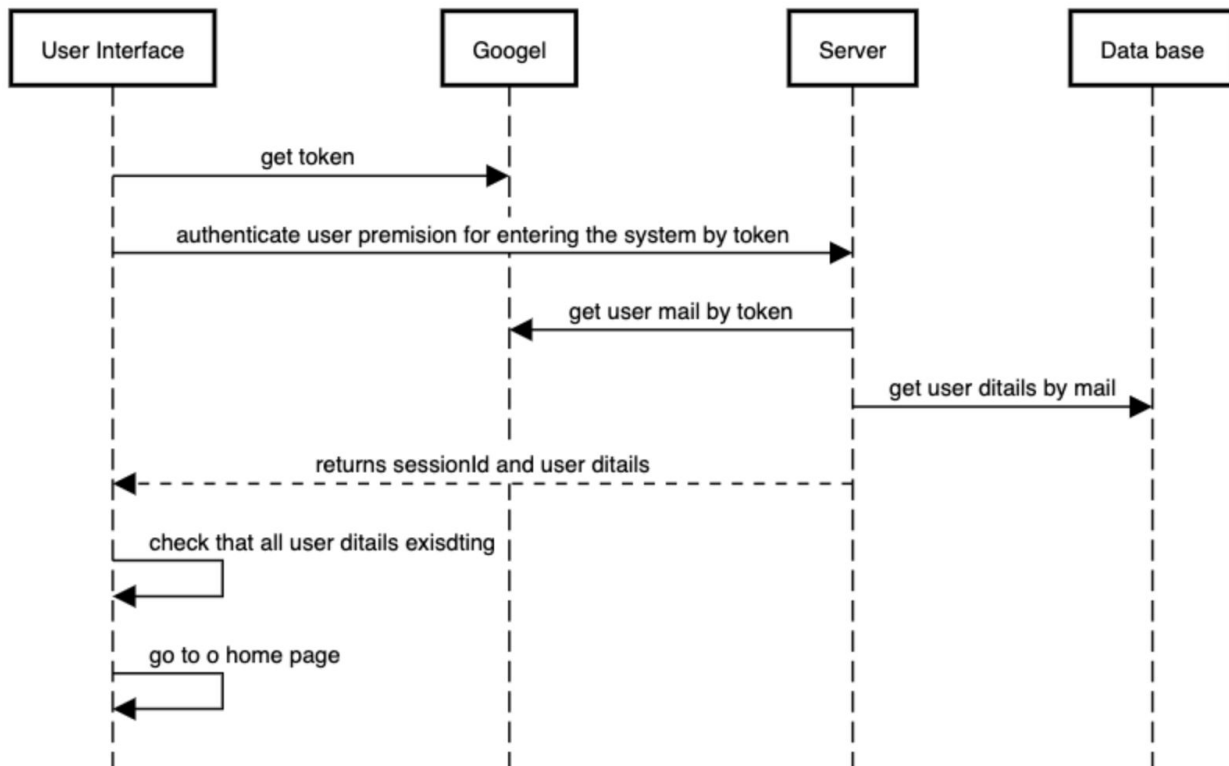


איור 1.16 - תרשים זרימה של כניסה למערכת באמצעות מייל וסיסמה

## 13.9 כניסה למערכת באמצעות חשבון גוגל

1. המשתמש לוחץ על כפתור Login
2. בטופס שנפתח, המשתמש לוחץ על כפתור Sign In
3. נפתח חלון נוסף, בו המשתמש מתחבר לחשבון הגוגל שלו (באמצעות אימייל וסיסמא) ומיד לאחר ההתחברות מועבר לדף הראשי של האתר, במידה ומילא כבר את פרטיו הדמוגרפיים אחרת מועבר למילוי טופס פרטים דמוגרפיים.

## כניסה למערכת באמצעות חשבון גוגל



איור 1.17 - תרשים זרימה של כניסה למערכת באמצעות חשבון גוגל

## 14 תכנון הפרויקט

### 14.1 ריכוז שינויים

שינוי	סיבת שינוי	תאריך שינוי	משמעות לפרויקט
הורדת המלצות לקורסים שהם לא און ליין	היות ולא מצאנו סט נתונים מספיק טוב על קורסים שאנשים לקחו לא און ליין	6.2019	שינוי בתוצר הפרויקט. המלצות יהיו רק על קורסים שנלקחים און ליין.

שינוי בפונקציונליות של הפרויקט. שנוי זה הביא להמלצות מדויקות יותר.	1.2020	בגלל תוצאות לא טובות עשינו תכנון ויישום מחדש של האלגוריתם	שינוי של אלגוריתם content based filtering
שינוי בדרישה המביא שיפור חווית משתמש	4.2020	שינוי זה בא עקב בחינה מחדש של חביית משתמש	שינוי ברישום משתמש חדש למערכת (9.3.1)
שינוי בדרישה אשר מעלה את אמינות הנתונים שבהם משתמש האלגוריתם ובכך לא נותן לדיוק ההמצות לרדת.	5.2020	שינוי זה בא עקב רצון לשמירת אמינות נתונים שנמצאים במערכת.	שינוי אופן הכנסת קורס חדש למערכת (9.3.2)

## איור 1.18 - טבלת ריכוז שינויים

## 14.2 ניהול סיכונים

### 14.2.1 מציאת סט נתונים מתאים

חיפשנו סטים של נתונים של קורסים און ליין ולא און ליין. מצאנו סטים מספקים מידע רלוונטי ואמין לאלגוריתם רק עבור קורסים און ליין. ובעקבות זה שינינו את מהות המערכת והאלגוריתם לקורסים שהם רק און ליין.

## 15 בדיקות והערכה

### 15.1 בדיקות מדדים ואלגוריתמים

#### 15.1.1 אופן בדיקת אלגוריתם היברידי ראשי

לקחנו 2 מערכי נתונים של MIT ו Harvard University ביחד. מערך ראשון הוא בעל יותר מ 600 אלף שורות של משתמשים, קורסים שלקחו ונתונים דמוגרפיים שלהם. ועוד מערך נתונים



שאספנו לבד על 120 קורסים ממוסדות אלו, הכולל מספר קורס, מרצה, נושאים בקורס ותיאור הקורס.

חילקנו את מערך נתוני משתמשים שלקחו קורסים ל 80% נתוני אימון, 10% נתוני ולידציה ו 10% נתוני בדיקה.

את הוולידציה לקחנו כל פעם 10% אחרים (סה"כ 9 חלקים שונים שווים). בכל קטע הרצנו על כל המשתמשים שבולידציה את האלגוריתם המלצות ההיברידי של המערכת שלנו, ובדקנו האם בקורסים שמוציא האלגוריתם מתוך נתוני אימון יש את כל הקורסים שלקחו המשתמשים בנתוני הוולידציה. ממוצע של כל חלקי הוולידציה היה 85.3% של פגיע בקורסים אלו בכלל משתמשים.

## 15.1.2 בדיקת השפעת משקולות של נתונים דמוגרפיים על אלגוריתם

### היברידי

למען שיפור תוצאות אלגוריתם ההיברידי, שיחקנו עם המשקולות שנתנו לכל עמודה של נתונים דמוגרפיים מנורמלים, לפני הרצת אלגוריתם K-MEANS. הרצנו את בדיקת אלגוריתם היברידי (15.1.1) כל פעם עם וריאציות שונות של משקולות כאשר כל משקולת קיבלה פעם ערך 1 ופעם ערך 10. בדקנו את דיוק האלגוריתם, ובחרנו בוריאציה של משקולות אשר מביא את האלגוריתם ההיברידי לדיוק הכי גבוה. מתוצאות הבדיקה המשקולות שנבחרו הם:

1. מגדר - משקולת של 10
2. גיל - משקולת של 1
3. רמת השכלה - משקולת של 10
4. קו רוחב במיקום גיאוגרפי - משקולת של 1
5. קו אורך במיקום גיאוגרפי - משקולת של 10

### 15.1.2.1 תוצאות

מתוצאות הבדיקה עולה שכאשר משתמשים יעדיפו יותר ללכת במסלולי לימוד של משתמשים אחרים, אשר באותו מגדר ורמת השכלה מאשר גיל. ההפתעה הכי גדולה בתוצאות, הייה לראות שדווקא אנשים בעלי קרבת מיקום גיאוגרפי, מבחינת קו אורך יותר קרובים במסלולי לימוד מאשר בקירוב בקו רוחב.

עוד דבר חשוב לציין זה סינון דמוגרפי עם משקולות לא נכונות יכול להזיק לאלגוריתם היברידי יותר מאשר עם לא היו משתמשים בו בכלל. כי כשבדקנו אלגוריתם היברידי בלי שילוב של סינון שיתופי יצא 4.87 אחוז פחות מאשר עם. אבל כאשר בדקנו עם שילוב סינון שיתופי ועם הכי לא נכונות יצאו ב 11 אחוז פחות.

### 15.1.3 בדיקת שימוש דמיון קוסינוסי או דמיון קוסינוס רך

כאשר בנינו את האלגוריתם סינון מבוסס תוכן, שהוא חלק מאלגוריתם היברידי ראשי, התלבטנו בין שתי השיטות הכי רלוונטיות לאלגוריתם זה:

1. דמיון קוסינוסי (7.1.1)

2. דימיון קוסינוס הרך (7.1.2)

בשביל לבדוק מי יותר טוב השתמשנו בכל אחד מהם בנפרד בתוך אלגוריתם סינון מבוסס תוכן, ובדקנו את את הדיוק של האלגוריתם ההיברידי (15.1.1). דמיון קוסינוסי הביא דיוק יותר גבוה ב 9 אחוז. ולכן שיטה זו נבחרה לישום.

### 15.1.3.1 תוצאות

התוצאות סותרות את הטענה שדמיון קוסינוסי הרך, יביא תוצאות לתוצאות יותר טובות בגלל שמתחשב גם במשמעות המילים (7.1.2), כאשר משתמשים באלגוריתם למידה מסוג סינון מבוסס תוכן טקסטואלי.

### 15.1.4 בדיקת תרומה של סינון דמוגרפי

כדי לוודא שחלק של סינון דמוגרפי אכן תורם לאלגוריתם ההיברידי, ולא גורע ממנו. הרצנו בדיקת אלגוריתם היברידי (15.1.1) פעם אחת עם התחשבות של חלוקה למקבצים (סינון שיתופי פועל רק על קבוצה שאליה שייך המשתמש) שנעשתה בסינון דמוגרפי ופעם אחת בלי להתחשב בחלוקה (כולם באותה קבוצה). תוצאות של סינון שיתופי עם סינון דמוגרפי ומשקולות שנבחרו ב 15.1.2 היו תוצאות יותר גבוהות ב 4.87 אחוז מאשר רק סיון שיתופי בלי סינון דמוגרפי. חשוב גם לציין שנבדקו 20 אופציות למספר מקבצים, ונמצא שמספר מקבצים שמביא לדיוק מקסימלי הוא 10.

### 15.1.5 בדיקת תרומה של סינון שיתופי וסינון מבוסס תוכן בנפרד

בדקנו את תרומת שני חלקי האלגוריתם בנפרד:

(1) סינון דמוגרפי עם סינון שיתופי

(2) סינון מבוסס תוכן

כאשר עשינו בדיקות כמו בסעיף 15.1.1 אבל אלגוריתם היברידי כלל רק סינון שיתופי שנעזר בסינון דמוגרפי (בלי סינון מבוסס תוכן) הדיוק של האלגוריתם עמד על 81%. וכאשר את אותה בדיקה עשינו כאשר אלגוריתם היברידי כלל רק סינון מבוסס תוכן אז תרומתו הייתה 43 אחוז. מיזוג של שני החלקים הביא לדיוק של 85.3 אחוז.

### 15.1.5.1 תוצאות

אפשר לראות שרוב הדיוק מתקבל ע"י סינון שיתופי, ושיפרנו אותו ע"י הוספת סינון מבוסס תוכן.

אומנם לפי הנתונים אפשר לחשוב שחלק של סינון מבוסס תוכן הוא הרבה פחות משמעותי. אבל מחקירה מעמיקה של סט נתונים, הגענו להבנה שסינון מבוסס תוכן נותן אחוז דיוק קטן יותר רק משום שיש הרבה משתמשים שלקחו קורס אחד בכל נושא או בחלק של הנושאים שהם למדו. ועקב כך המלצות שהבאנו לא יכולות להתאים לקורסים אחרים שלמדו. לפי דעתינו תופעה זו פחות מייצגת מציאות של אנשים הלומדים קורסי און ליין, אלא יותר מייצגת את ההגבלה במגוון הקורסים שעמדו לרשותינו (מתוך קורסים שלקחו משתמשים) בבדיקות. ואם המגוון היה יותר רחב אז אחוזי תרומה של שני החלקים בנפרד היו יותר מאוזנים, כאשר עדיין היה נשמר אחוז דיוק בשילוב של שניהם.

### 15.1.6 בדיקת הגבלת כמות הקורסים, לשימוש בסינון שיתופי

עשינו בדיקה האם שווה להגביל סינון שיתופי לאנשים שלקחו לפחות 2 קורסים או 3 קורסים בעבר. כאשר הגבלנו למינימום של 2 קורסים הדיוק ירד מ 81 אחוז (במינימום קורס אחד) ל 78 אחוז, וכאשר הגבלנו למינימום 3 קורסים אז הדיוק ירד ל 76 אחוז. בעקבות זה הגענו למסקנה שלא נעלה את הרף של מינימום קורסים מעבר להכרחי שזה אחד. וזה גם יביא לנו מרחב גדול יותר של לקוחות.

#### 15.1.6.1 תוצאות

בתוצאות של הבדיקות קיבלנו דיוק פחות טוב בהמלצות על קורסים לאנשים שעשו מסלולים של 2 קורסים ויותר ועד קצת פחות ל 3 קורסים ויותר. תוצאות אלו משום מה סותרות את הגיון האלגוריתם שכביכול הדיוק אמור לעלות ככל שמסלול משותף עולה (כמה שיותר קורסים משותפים כך דיוק ההמלצות עולה). חוסר הדיוק יכול להיגרם מ-2 סיבות:

1. מחסור בנתונים - מסלול הלמידה עבור לוקחי הקורסים הם לא שלמים (ישנו מידע על סטודנטים שלקחו קורסים רק עבור 13 קורסים, כאשר יש לנו בסה"כ מידע על 120 קורסים, וניתן להניח שלסטודנטים אלו קיימים עוד קורסים, ואנו מקבלים רק תמונה חלקית על מסלול הלימודים שלהם.

2. מסלול הלמידה של לוקחי קורסי האונליין הוא לא עקבי (לפחות לפי הנתונים שלנו), הם לא לומדים קורסים בנושאים שקיימים תחת מחלקה משותפת, לכן אין מכנה משותף בין הקורסים אין דפוס אחיד, וזה יכול להעיד שמגמת התוצאות מקרית בלבד.

### 15.1.7 מסקנות של בדיקות מדדים ואלגוריתמים

מהתוצאות אפשר לראות שכל חלק באלגוריתם ה-"היברידי"(סיון שיתופי, סינון מבוסס תוכן וסינון דמוגרפי) משפר את הדיוק הוצאת המלצות. כלומר אם נוציא אחד משלושת החלקים מתוך האלגוריתם "ההיברידי" אז הדיוק של האלגוריתם יפגע. כאשר חשוב מאוד גם לתת משקולות נכונים לכל נתון דמוגרפי של משתמש, ובחירת מספר מקבצים נכון. גם מצאנו שדמיון קוסינוסי רגיל מביא לתוצאות יותר טובות מאשר דמיון קוסינוס רך.

## 15.2 בדיקות פונקציונליות

### 15.1.6 בדיקת הזנת נתונים בעת רישום משתמש במערכת

בדקנו שאי אפשר להזין או לעדכן נתונים בפורמט לא נכון, או כאשר חסר חלק מהנתונים.

### 15.1.7 פונקציונליות של כפתור בקשת המלצות לקורסים

בדקנו שכל עוד לא עודכנו פרטים ע"י המשתמש לא יהיה לו אופציה לקבל המלצות.

### 15.1.8 בדיקת מהירות ריצת האלגוריתם

מרגע לחיצה על כפתור קבלת המלצות לקורסים ועד קבלת המלצות עברו 2 שניות.

### 15.1.9 בדיקת נכונות פלט אחרי לחיצה על כפתור בקשת המלצות לקורסים

עשינו בדיקה שכל הפלט ברור ומסודר כמו שצריך.

## 15.3 בדיקות שמישות

נתנו לאנשים בעלי עניין בנושא שאנחנו מכירים ודרך גוגל פורם שאלון ובדקנו את תשובתם.

### 15.3.1 סיבוכיות של הגעה להמלצות

ביקשנו שישתמשו במערכת להוצאת המלצות לקורסים, ושאלנו האם הסתדרו בלי בעיה בכל תהליך ההרשמה, מילוי פרטים דמוגרפיים וקבלת המלצות וב 95 אחוז מהמקרים אנשים הבינו בעצמם את הפעלת המערכת עד לקבלת המלצות.

### 15.3.2 עדכון פרטים אישיים

שאלנו האם תהליך עדכון פרטים ברור וקל לדעתם. התשובות היו ב 100 אחוז כן.

### 15.3.3 קבלת קורסים לפי נושא

שאלנו האם תהליך קבלת קורסים לפי נושא הוא ברור וקצר לפי דעתם. התשובות היו ב 100 אחוז כן.

### 15.3.4 עורך התהליך של הגעה להמלצות

שאלנו האם תהליך קבלת המלצות מספיק קצר כדי שלא יתייאשו בדרך לקבלת ההמלצות (מטרת המערכת). ב 90 אחוז אנשים אמרו שהדרך לא ארוכה להם.

## 15.4 דוגמאות הפעלה

### 15.2.1 בקשת המלצות בהתאמה אישית

1. לוחצים על כפתור get recommendations

2. מקבלים את ההמלצות על קורסים

#	Title	Description	Main Topics	Id
1	Design and Development of Educational Technology	This project-based course explores educational technologies and the theories underlying their development through interviews with experts in the field.	Computer Science	11132x
2	Implementation and Evaluation of Educational Technology	This course provides a practical overview for selecting, integrating, implementing, and evaluating educational technology initiatives in formal educational settings, primarily in the US. It will include the perspectives of stakeholders that make such initiatives possible, and consider how to evaluate for efficacy.	Computer Science	11133x

### 15.2.2 הרשמה למערכת והזנת פרטים דמוגרפיים

1. לוחצים על כפתור sign in/up בפינה השמאלית למעלה

Sign In / Up

-- get courses by subject --



**DecidesForYou**

GET COURSE R

2. נפתח חלון הרשמות למערכת/ כניסה למערכת

Welcome!



Sign In

G+ Sign in with Google

OR

Email address

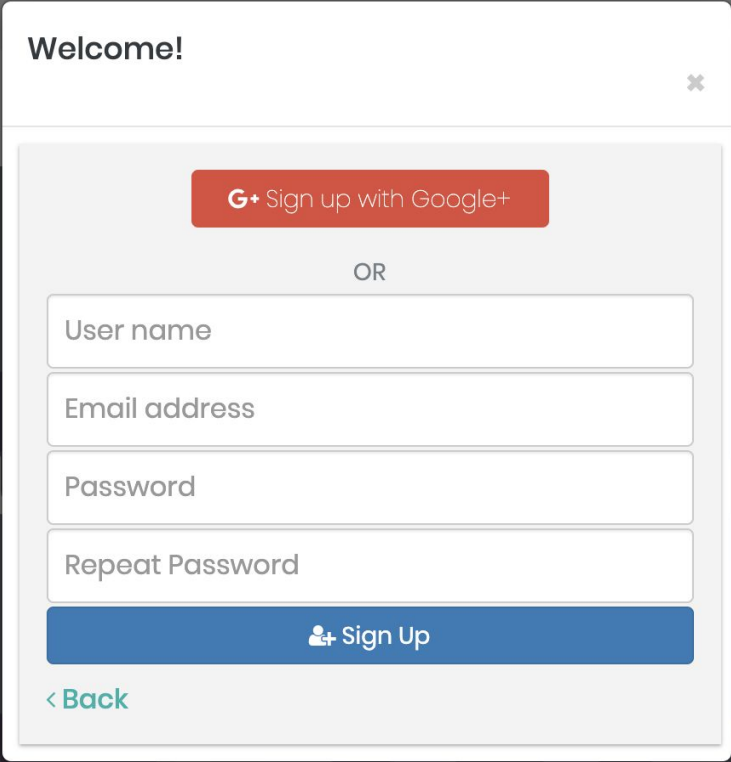
Password

Sign in

+ Sign up New Account

3. לוחצים על כפתור Sign up New Account

4. מופיעה חלון הרשמה



Welcome! x

G+ Sign up with Google+

OR

User name

Email address

Password

Repeat Password

+ Sign Up

< Back

5. מכניסים את הפרטים ולוחצים על כפתור Sign Up

6. מופיע חלון להכנסת פרטים דמוגרפיים

## To Complete Registration You Must Fill Profile Details



Profile

Add Course

Pick a User Name

rrr

Gender



Male



Female

Select your country

--choose--



Select year of birth



Select education

--choose education--



Select courses you took in the past



No Courses



Title

Description

Update profile

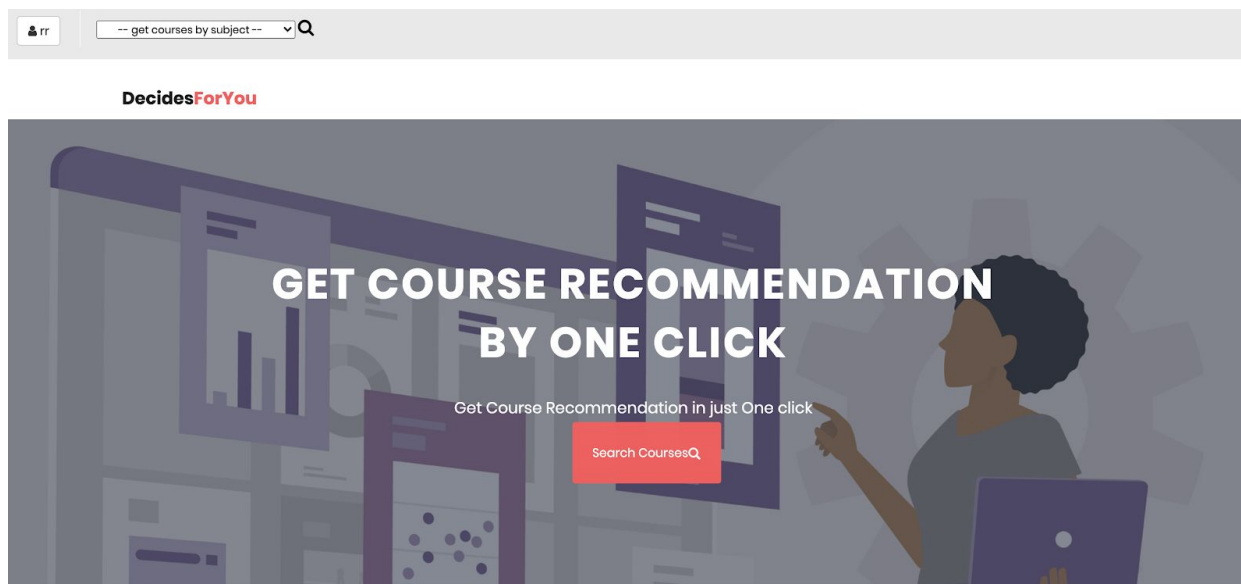
reset

Sign out



7. מכניסים את כל הפרטים וקורסים שנלמדו בעבר ע"י משתמש ולוחצים על כפתור Update profile

8. המשתמש מועבר לדף הבית של הפרופיל שלו



## 15.2 בדיקות שמישות

נתנו לאנשים בעלי עניין בנושא שאנחנו מכירים ודרך גוגל פורם שאלון ובדקנו את תשובתם.

### 15.3.1 סיבוכיות של הגעה להמלצות

ביקשנו שישתמשו במערכת להוצאת המלצות לקורסים, ושאלנו האם הסתדרו בלי בעיה בכל תהליך ההרשמה, מילוי פרטים דמוגרפיים וקבלת המלצות וב 95 אחוז מהמקרים אנשים הבינו בעצמם את הפעלת המערכת עד לקבלת המלצות.

### 15.3.2 עדכון פרטים אישיים

שאלנו האם תהליך עדכון פרטים ברור וקל לדעתם. התשובות היו ב 100 אחוז כן.

### 15.3.3 קבלת קורסים לפי נושא

שאלנו האם תהליך קבלת קורסים לפי נושא הוא ברור וקצר לפי דעתם. התשובות היו ב 100 אחוז כן.

### 15.3.4 עורך התהליך של הגעה להמלצות

שאלנו האם תהליך קבלת המלצות מספיק קצר כדי שלא יתיאשו בדרך לקבלת ההמלצות (מטרת המערכת). ב 90 אחוז אנשים אמרו שהדרך לא ארוכה להם.

## 16 הצעה לעבודת המשך

1. בגרסה הבאה נרצה להכניס פיצ'ר להוצאת המלצות אישיות רק בנושא אחד לפי בחירה.
2. לשפר את זמן הוצאה ועדכון נתונים במאגר נתונים חיצוני. בגלל מגבלות כספיות כרגע המאגר מוגבל בזכרון ומשאבים.
3. כתיבת קובץ הרצה אשר מכוון את המשקולות ומספר קלסטרים אוטומטי. ע"י הרצת בדיקות ובחירת הערכים הטובים ביותר.
4. לנסות לשפר את האלגוריתם ההיברידי של "מתלבט בשבילך" להוצאת המלצות לקורסים אונליין ע"י שילוב של אלגוריתמי למידה נוספים.

## 17 רשימת מקורות

- [1] D Gunawan  
The Implementation of Cosine Similarity to Calculate Text Relevance between Two Documents  
Journal of Physics: Conf. Series 978 (2018)
- [2] Grigori Sidorov, Alexander Gelbukh, Helena Gomez-Adorno, and David Pinto  
Soft Similarity and Soft Cosine Measure: Similarity of Features in Vector Space Model  
1Centro de Investigacion en Computacion, Instituto Politecnico Nacional, Mexico D.F., Mexico.

2 Facultad de Ciencias de la Computacion, Benemerita Universidad Autonoma de Puebla, Puebla, Mexico.

Computacion y Sistemas Vol. 18, No. 3, 2014

[3]. Robin Burke<sup>1</sup>, Alexander Felfernig<sup>2</sup>, Mehmet H. Göker<sup>3</sup>,  
Recommender Systems: An Overview, DePaul University, Graz University  
Article in Ai Magazine · September 2011

[4]. Michael J. Pazzani and Daniel Billsus, Content-Based Recommendation Systems, Rutgers University, FX Palo Alto Laboratory. P. Brusilovsky, A. Kobsa, and W. Nejdl (Eds.): The Adaptive Web, LNCS 4321, pp. 325 – 341, 2007

[5]. Mohr.M, Rostamizadeh.A, Talwalkar.A, Foundations of Machine Learning, The MIT Press Books, 2012.

[6]. Laila Safoury and Akram Salah, Exploiting User Demographic Attributes for Solving Cold-Start Problem in Recommender System. Lecture Notes on Software Engineering, Vol. 1, No. 3, August 2013

[7]. Souvik Debnath, Machine Learning Based Recommendation System, Indian Institute of Technology, Kharagpur. May 2008

[8] Purnima Bholowalia and Arvind Kumar, EBK-Means: A Clustering Technique based on Elbow Method and K-Means in WSN

International Journal of Computer Applications (0975 – 8887) Volume 105 – No. 9, November 2014

<https://www.youtube.com/watch?v=YIGtalP1mv0>

# 18 נספחים

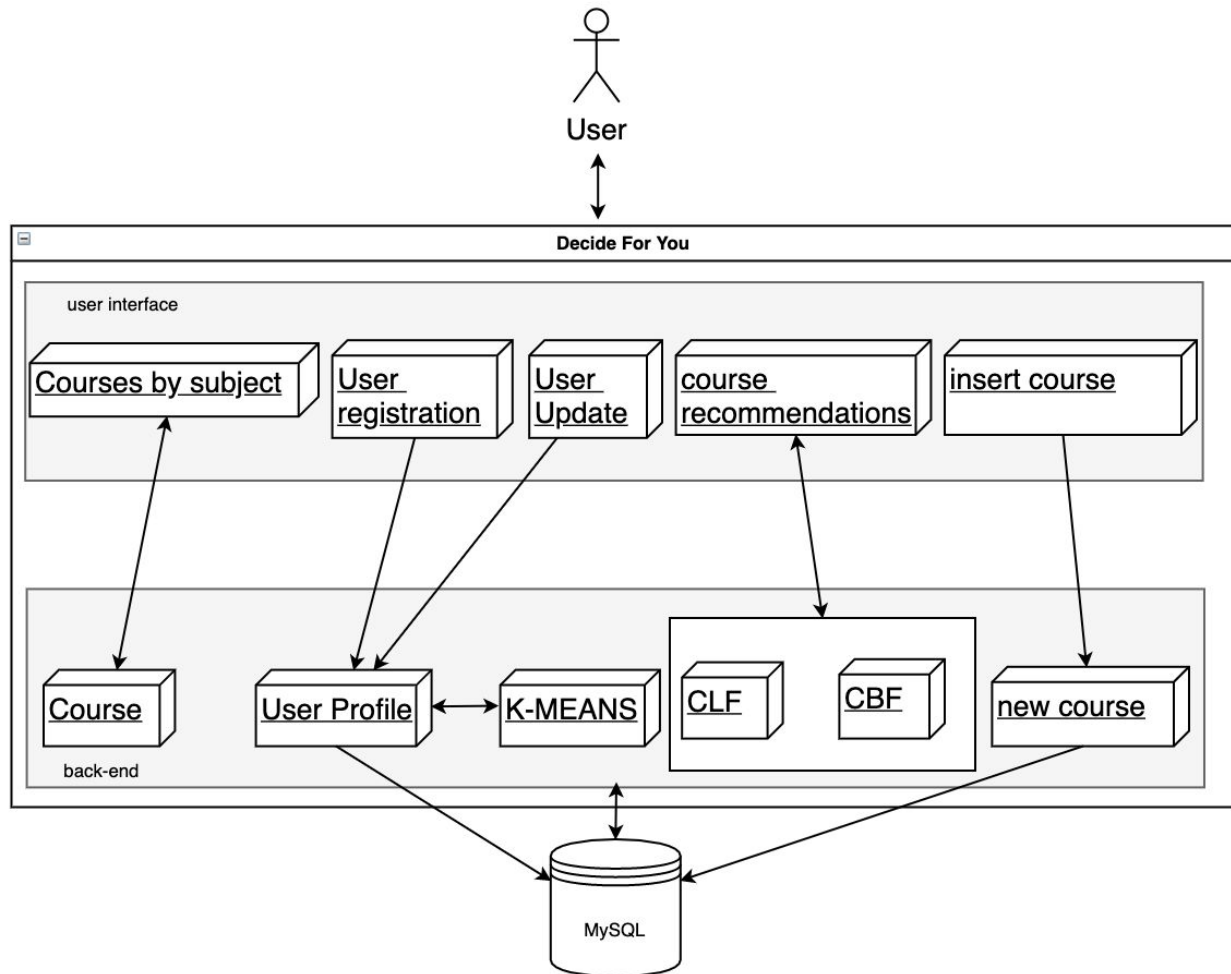
## הפוסטר

### 18.1 נספח SRD

#### 1. הקדמה

מטרת מסמך זה היא להגדיר את הדרישות הנדרשות והפונקציונליות של הפרויקט "מתלבט בשבילך". מערכת זו תביא המלצות על קורסים לפי נושא. או שתוכל להפעיל אלגוריתם למידת מכונה למשתמשים שיכניסו קורסים שלקחו בעבר ונתונים דמוגרפיים שלהם. מוצר זה פותח לרמת אב טיפוס במסגרת פרויקט גמר במכללת "אפקה", שיכלול פיתוח ממשק WEB.

## 2. תיאור הארכיטקטורה שנבחרה



בחרנו להשתמש ארכיטקטורה של client - server. כאשר בצד המשתמש יהיה הדפדפן וובי שעליו יוצג הממשק ממשק, ודרכו המשתמש יוכל לבקש המלצות לקורסים מהשרת או קורסים לפי נושא.

השרת יפעיל את האלגוריתמים המתאימים, לפי בקשות ממשק משתמש. במהלך ריצת האלגוריתם ישלפו נתונים מבסיס נתונים נפרד.

## PYSPARK 2.1

אלגוריתם K-MEANS רץ תוך שימוש בספריית PYSPARK אשר יודעת להתמודד עם כמויות גדולות של Data ע"י הרצה של תהליכים במקביל. וזאת למען קיצור זמן של שיוך משתמש חדש למקבץ. או באת שינוי פרטים דמוגרפיים

## mysqlDB 2.2

ה Data נשלף ע"י קוד צד שרת ב python משרת mysqlDB נפרד. בסיס נתונים שהשתמשנו בו הוא רלציוני. וזאת באשר התאמה טובה לסוג הנתונים אשר אנו משתמשים בהם, ואופי האלגוריתמים.

## 2.3 user interface - ממשק משתמש

צד הלקוח יהיה על פני דפדפן אינטרנט שיקבל את הנתונים מצד השרת. הבחירה ב Web UI היא משיקולי נוחות בעיצוב (שימוש בספריות JavaScript קיימות וקבצי סגנונות), תוך כדי הפרדת השרת ממשק המשתמש.

ממשק משתמש כולל:

- רישום משתמש חדש למערכת ע"י מייל וסיסמה או חשבון גוגל.
- חלונית הכנסת פרטים דמוגרפיים וקורסים שנלקחו בעבר ע"י המשתמש.
- חלונית שינוי פרטים דמוגרפיים וקורסים שנלקחו בעבר ע"י המשתמש.
- אפשרות לקבלת קורסים לפי נושא.
- אפשרות קבלת המלצות לקורסים.
- ממשק הכנסת קורסים חדשים למערכת ע"י משתמש עם הרשאות מתאימות.

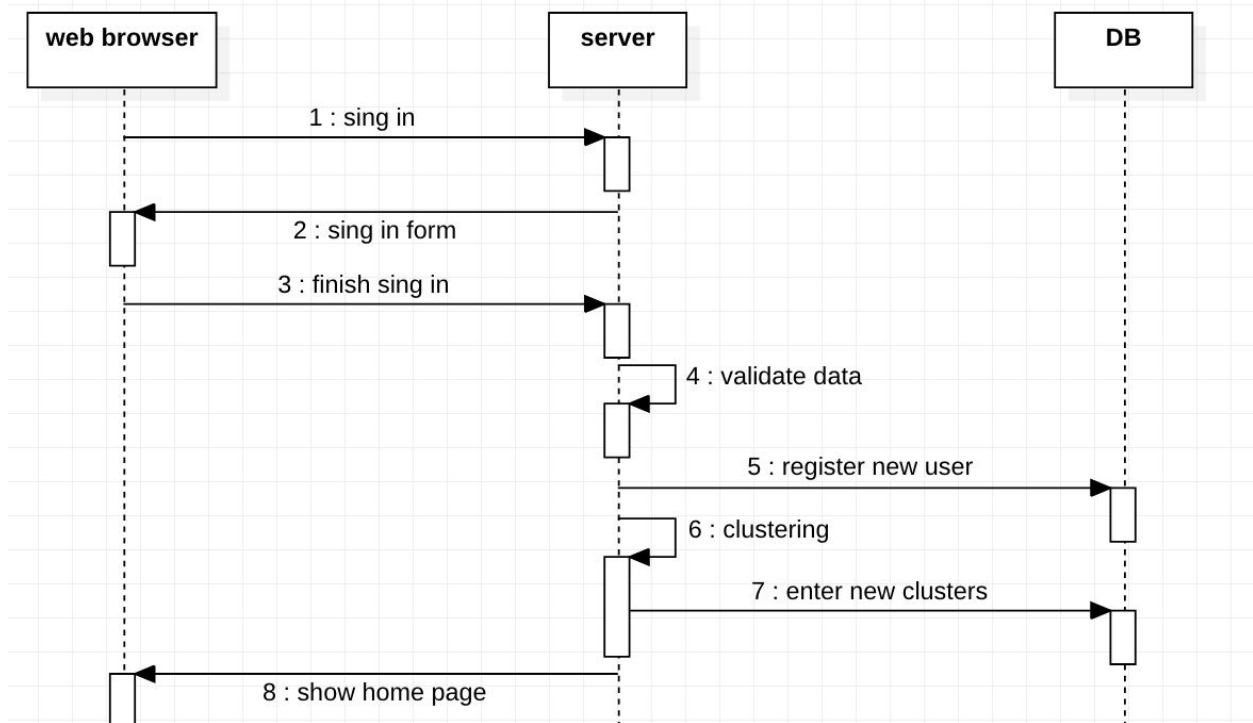
## 3 הגדרת דרישות

### 3.1 ניתוח פונקציונאלי ראשוני

#### 3.1.1 רישום משתמש חדש למערכת

ביצוע של סיווג דמוגרפי יתבצע בעת רישום משתמש למערכת, מיד לאחר מסירת נתונים דמוגרפיים ע"י המשתמש החדש, וגם לאחר בדיקתם (אם הנתונים לא יתאימו לסטנדרט של שדות בשאלון הרשמה יוצג בפני המשתמש הודעת לבקשת תיקון שדה). בעת סיווג דמוגרפי המשתמש יקוטלג לקבוצת משתמשים אשר הכי קרובים אליו בנתונים דמוגרפיים

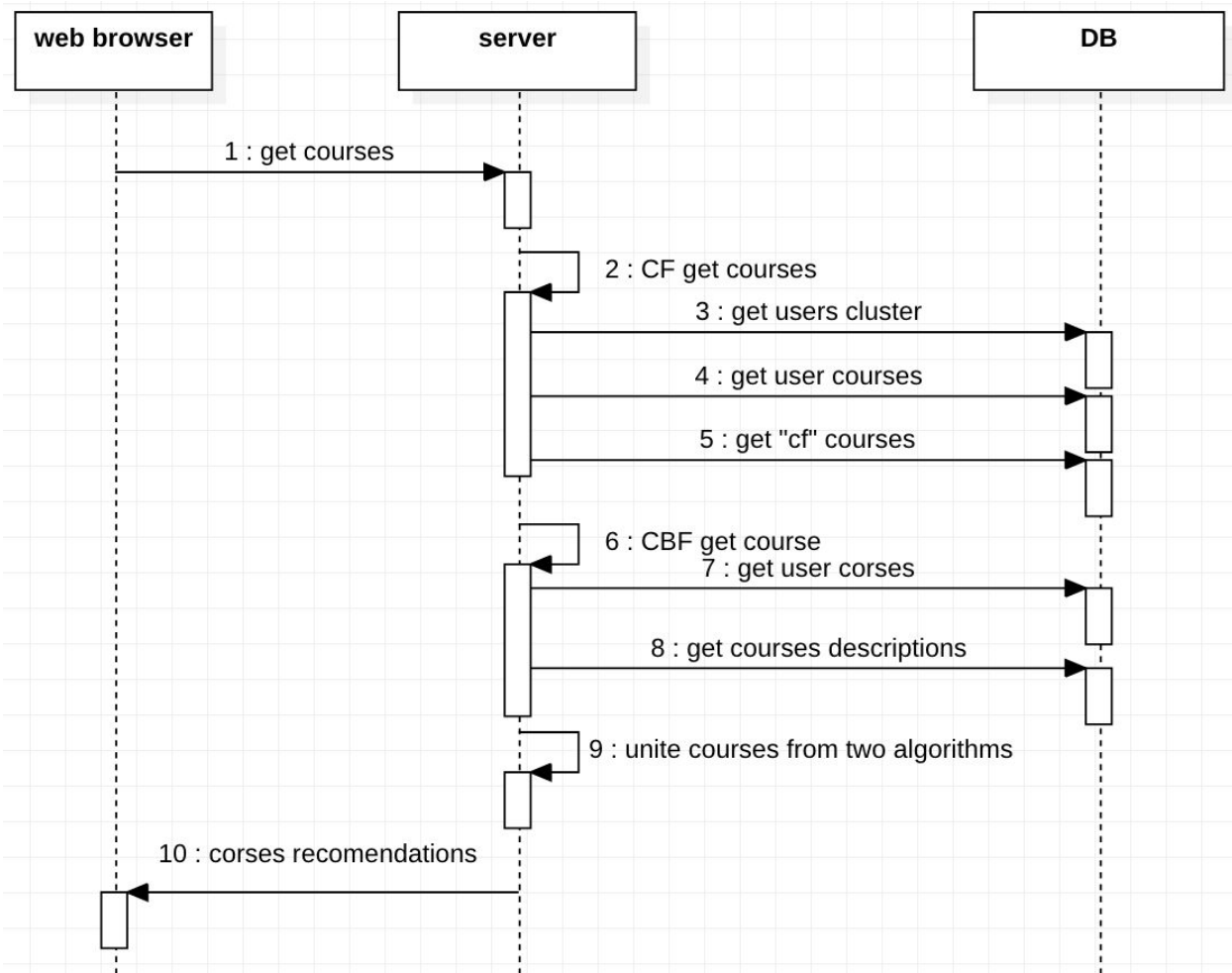
זזה מושג ע"י הרצת אלגוריתם למציאת מקבצים ששמו k-means. ובסוף יוצג בפני המשתמש דף הבית של האתר שלנו.



### 3.1.2 בקשה להמלצות על קורסים - משתמש שלקח קורסים בעבר

- אופציה זו תפתח רק למשתמשים שלקחו קורסים בעבר.
- המשתמש מבקש המלצות לקורסים דרך ממשק ה WEB.
- הבקשה נשלחת לשרת שמתחיל להריץ אלגוריתם של סינון שיתופי ואחריו ירוץ סינון מבוסס תוכן – שניהם ע"פ נתוני הקורסים שהמשתמש לקח בעבר. תוך כדי הרצת האלגוריתמים הסרבר מבצע שאילתות SQL על מאגר נתונים.

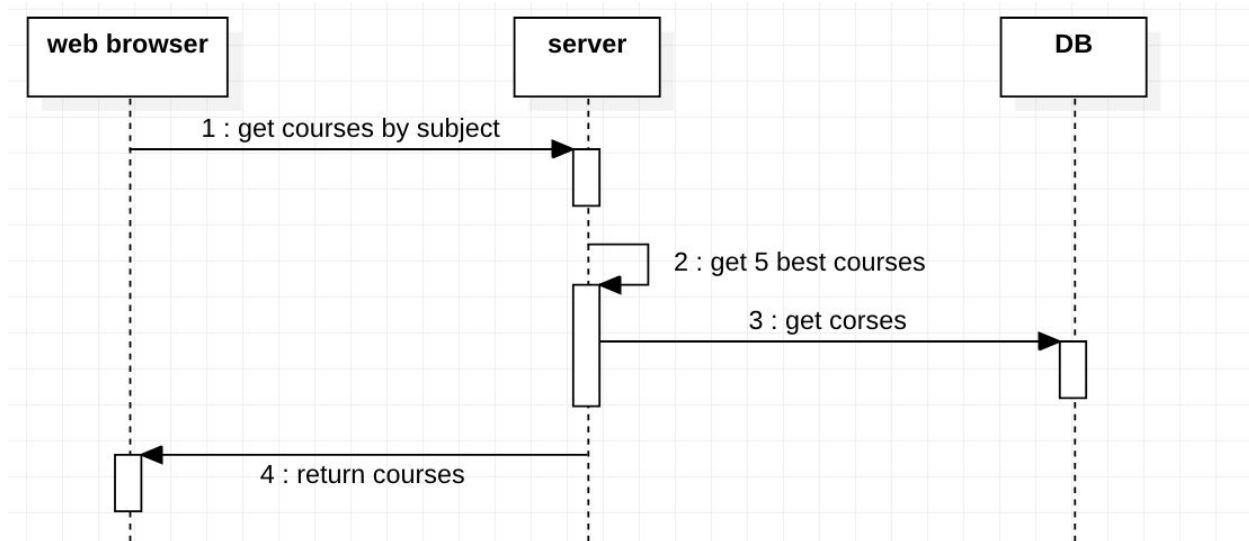
- לאחר סיום האלגוריתמים ומציאת הקורסים המתאימים למשתמש, השרת יחזיר תשובה לצד הלקוח, כלומר לממשק ה WEB שלנו, ביחד עם רשימת קורסים ומידע אודותיהם.
- הממשק ימיר את המידע לתצוגת UI מסודרת של רשימת קורסים.



### 3.1.2 בקשה להמלצות על קורסים - משתמש שלא לקח קורסים בעבר

המשתמש יצטרך לבחור נושא לפני מרשימת נושאים שיוצגו בפניו. ולאחר מכן יפתח הכפתור של בקשת המלצות לפי נושא. לאחר לחיצה על הכפתור השרת יתחיל להריץ אלגוריתם למציאת 5 קורסים בעלי עדיפות ע"פ אותו נושא, בהעזרות של מאגר נתונים.





### 3.2 דרישות ביצועים

3.2.1 המערכת תוציא המלצות לקורסים דרך אלגוריתם למידת מכונה בפחות מעשר שניות.

3.2.2 המערכת תוציא המלצות לקורסים לפי נושא בפחות משלוש שניות.

### 3.3 דרישות משאבים

3.3.1 ממשק משתמש של המערכת יופעל ע"י דפדפן אינטרנט.

3.3.2 נתוני המערכת יואכסנו וישמרו במאגר נתונים רלציוני MySQL.

3.3.3 הסרבר יופעל ע"י רשת מחשבים (האלגוריתם ירוץ על כמה מחשבים במקביל באם על ענן או רשת מקומית).

### 3.4 דרישות אבטחה

אין דרישות אבטחה

## נספח א - SDD

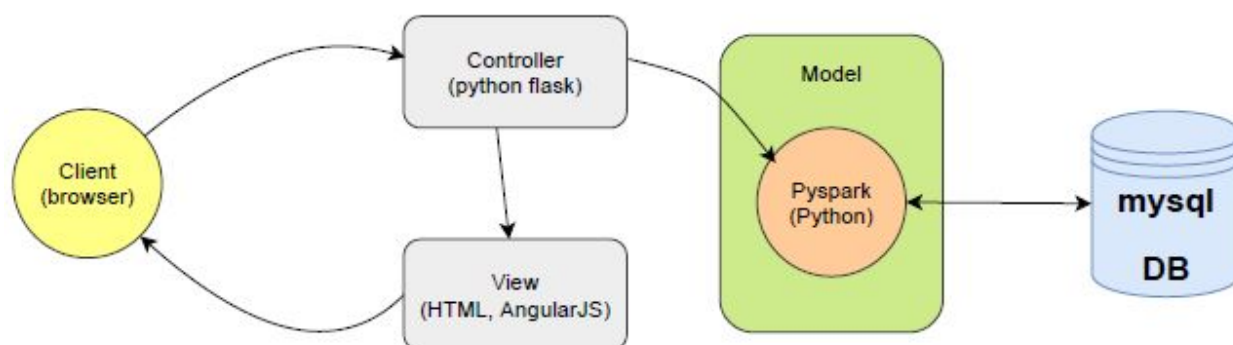
## 1 הקדמה

מסמך עיצוב מתאר את צד התוכנה מבחינת ארכיטקטורה נבחרת, ממשקים בין רכיבים ותוכניות, תיאור רכיבים משמעותיים וממשקי משתמש.

## 2 תכן ארכיטקטורת המערכת

### 2.1 ארכיטקטורה נבחרת

הארכיטקטורה שנבחרה היא client-server צד המשתמש (דפדפן) וצד השרת. כל הלוגיקה של המערכת תתבצע בצד השרת: קבלת בקשות להמלצות מהמשתמש, הפקת רשימת קורסים עבור מבקש ההמלצה, אימות משתמש המערכת. תיאור כללי של הארכיטקטורה:



### 2.2 צד הלקוח

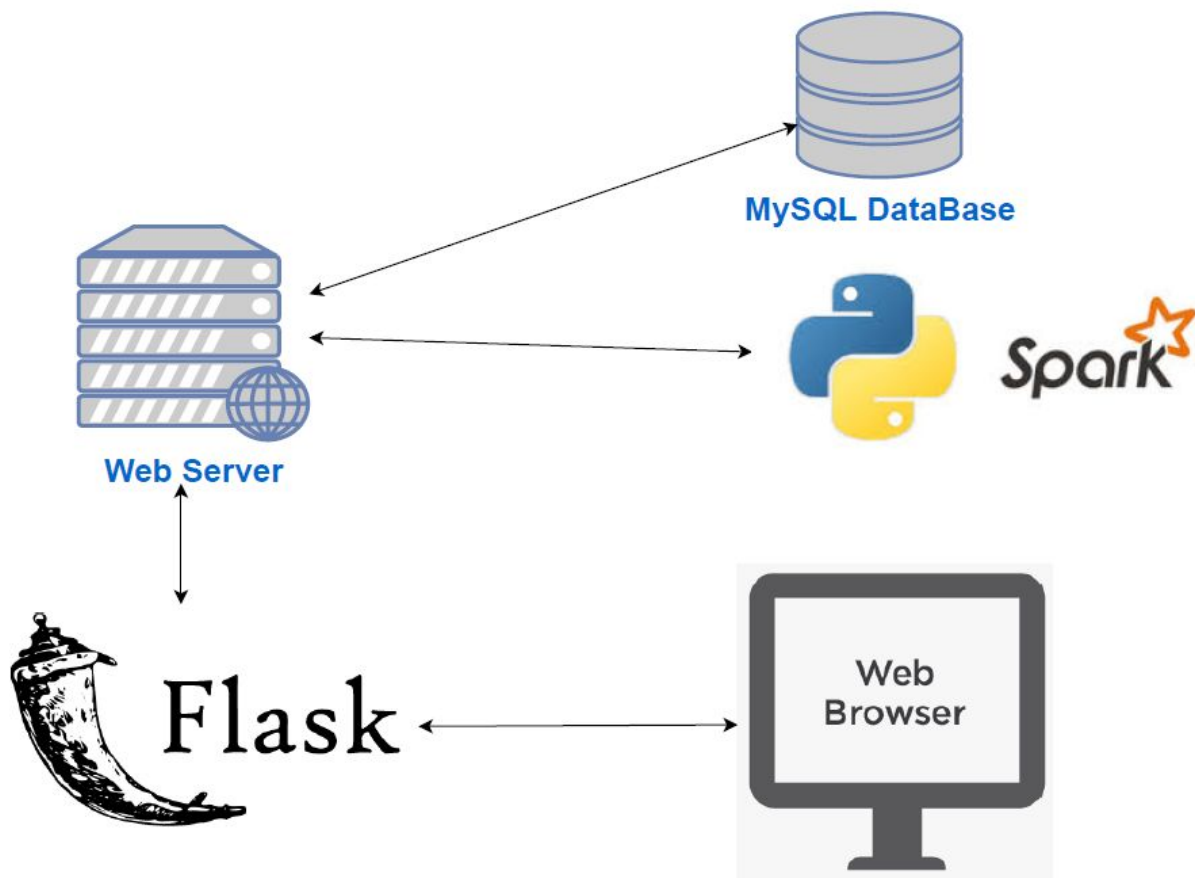
צד הלקוח הוא הדפדפן שהמשתמש מפעיל. דרכו יכול המשתמש להתחבר למערכת, לבקש המלצות לקורסים שבהם הוא מעוניין, ולצפות ברשימת הקורסים שהופקה עבורו. צד הלקוח משתמש בAngularJS ביחד עם HTML.

### 2.3 צד השרת

מדובר בשרת יישום שיושב על ענן (Cloud). השפה הראשית של צד השרת היא PYTHON, תוך שימוש בספריות של השפה שהמרכזיות בהן היא Flask לצורך הגדרת שרת WEB, PySpark שהוא הכלי העיקרי בלוגיקה לצורך חישוב יעיל של אלגוריתמים ליצירת המלצות.

נעשה שימוש במסד נתונים של MySQL שיאחסן את כל הנתונים של המערכת, לדוגמא: משתמשי המערכת, קורסים, טבלת קשר בין משתמשים לקורסים. מסד הנתונים יהיה על שרת זה או לחלופין יופעל בשרת אחר נפרד וייעודי.

## 2.4 ארכיטקטורת ממשקים



## 3 תיאור רכיבים

### 3.1 שרת ענן

3.1.1 סוג

שרת פיזי + תכנית

3.1.2 תכלית

שרת שיקבל את בקשות המשתמש מהדפדפן, יעבד אותן ויחזיר תשובה בהתאם.

### 3.1.3 קלט ופלט פונקציונליים

קלט http get request מדפדפן המשתמש. לדוגמא: התחברות משתמש, מעבר לעמוד אחר באתר, בקשת המלצות. הפלט יכול להיות רשימת קורסים ונתוני משתמשים בפורמט JSON, דפי HTML.

## PYSPARK 3.2

### 3.2.1 סוג

ספריית תוכנה

### 3.2.2 תכלית

חישוב אלגוריתמים מסובכים של למידת מכונה.

### 3.2.3 קלט ופלט פונקציונליים

קלט אפשרי: מחרוזות שמתארות מספר מזהה של משתמש, נושא של קורס.

פלט אפשרי: רשימה של קורסים המיועדים למשתמש במבנה של DataFrame.

### 3.2.4 מידע

הפונקציות שישמשו בספרייה זו יעבירו ביניהם נתונים של קורסים בצורת DataFrame.

## MYSQL 3.3

### 3.3.1 סוג

מסד נתונים רלציוני

### 3.3.2 תכלית

שמירת נתונים שישמשו עבור הבקשות הבאות מהשרת.

### 3.3.3 קלט ופלט פונקציונליים

קלט אפשרי: מחרוזת מזהה של משתמש, מחרוזת מזהה של קורס, מחרוזת שמתארת

נושא קורס

פלט אפשרי: רשומות מטבלה מסוימת במסד נתונים, כמו: רשומות של קורסים, רשומה של משתמש.

3.3.4 מידע

טבלאות נתונים המכילות קורסים, נתוני משתמשים, מידע אודות קורסים שמשתמשים לקחו בעבר.

## 4 עיצוב ממשק משתמש

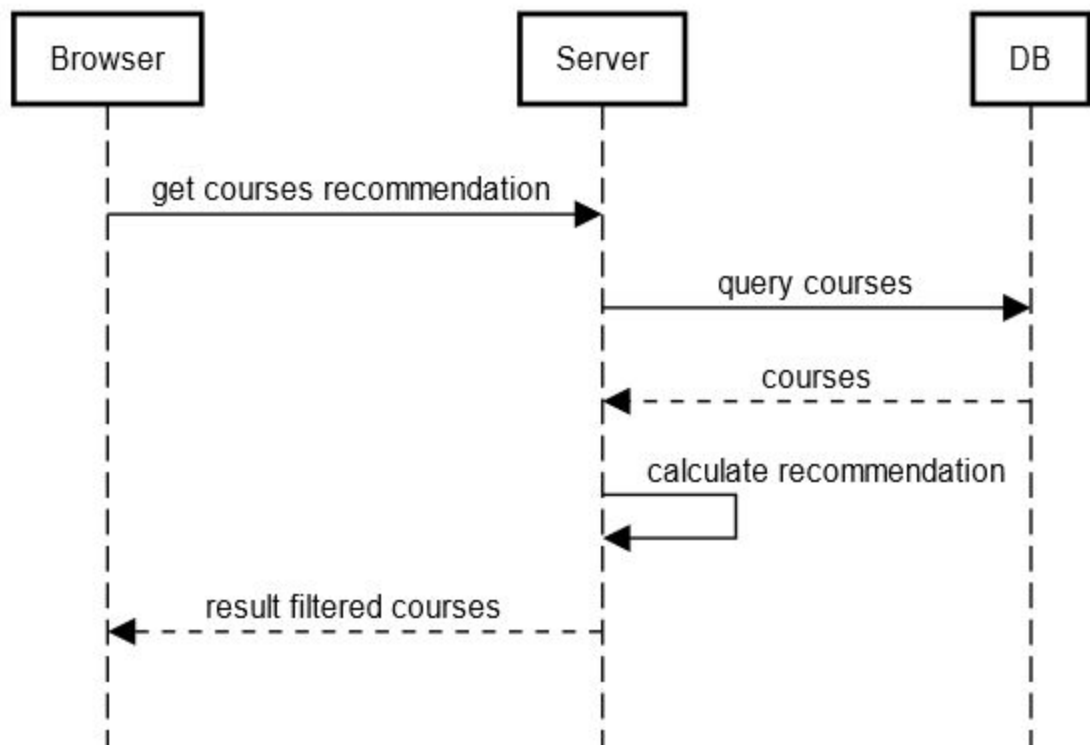
### 4.1 מזהה ממשק משתמש: בקשת המלצות מהמערכת

#### 4.1.1 תיאור ממשק משתמש

ממשק זה מאפשר לבקש המלצה קורסים על פי נושא קורס לבחירתו. בנוסף עבור משתמש שיש תיעוד שלקח קורסים בעבר תהיה אפשרות לבקש המלצה לקורסים ללא בחירת נושא.

#### 4.1.2 אובייקטים ופעולות

### recommendation request flow



#### 4.1.2 תמונת מסך



Title	Description	Main Topics	Id
Challenges of World Poverty	This is a course for those who are interested in the challenge posed by massive and persistent world poverty, and are hopeful that economists might have something useful to say about this challenge. The questions we will take up include: Is extreme poverty a thing of the past? What is economic life like when living under a dollar per day? Are the poor always hungry?	Social Sciences	14.73x
The Challenges of Global Poverty	This is a course for those who are interested in the challenge posed by massive and persistent world poverty, and are hopeful that economists might have something useful to say about this challenge. The questions we will take up include: Is extreme poverty a thing of the past? What is economic life like when living under a dollar per day? Are the poor always hungry?	Social Sciences	14.73x
Elements of Structures	2.01x introduces principles of structural analysis and strength of materials in applications to three essential types of elastic load-bearing elements: bars in axial loading, axisymmetric shafts in torsion, and symmetric beams in bending. The course covers fundamental concepts of continuum mechanics, including internal resultants, displacement field, stress, and strain.	Engineering	2.01x
Elements of Structures	2.01x introduces principles of structural analysis and strength of materials in applications to three essential types of elastic load-bearing elements: bars in axial loading, axisymmetric shafts in torsion, and symmetric beams in bending. The course covers fundamental concepts of continuum mechanics, including internal resultants, displacement field, stress, and strain.	Engineering	2.01x
Introduction to Solid State Chemistry	This first-year University chemistry course explores the basic principles of the chemical bond by studying the properties of solids. Properties such as stiffness, electrical conductivity, thermal expansion, strength, and optical properties are the vehicle by which you can learn a great deal of practical chemistry.	Chemistry	3.091x
Introduction to Computational Thinking and Data Science	6.00.2x will teach you how to use computation to accomplish a variety of goals and provides you with a brief introduction to a variety of topics in computational problem solving. This course is aimed at students with some prior programming experience in Python and a rudimentary knowledge of computational complexity.	Computer Science	6.00.2x
Circuits and Electronics	Want to learn about circuits and electronics, but unsure where to begin? Wondering how to make computers run faster or your mobile phone battery last longer? This free circuits course taught by edX CEO and MIT Professor Anant Agarwal and colleagues is for you.	Electronics	2.15x

## נספח std

### 1. הקדמה

במסמך ה-STD שלהלן נתאר את התכונות שיבדקו בפריטים הנבדקים ואת התכונות שלא יבדקו. בנוסף נגדיר את דרישות הסביבה של הרצת הבדיקות ונתאר את תסריטי הבדיקה ותוצאות הרצתם.

### 2. פריטים נבדקים

- שכבת UI
- אלגוריתם סינון מבוסס תוכן
- אלגוריתם סינון שיתופי
- אלגוריתם יצירת מקבצים (K-MEANS)
- שכבת השרת

#### 2.1 תכונות לבדיקה

בשכבת ה UI יבדקו התכונות הבאות :

- פונקציונליות הכפתורים
- הוצאת פלט תקין המורכב מקורסים ותיאוריהם
- ממשק הרשמות למערכת ע"י משתמש
- ממשק רישום קורס למערכת

באלגוריתם סינון מבוסס תוכן יבדקו תכונות הבאות:

- אחוזי קורסים שפולט אלגוריתם סינון מבוסס תוכן התואמים לנתוני בדיקה

- מהירות האלגוריתם - מודלים משפרי מהירות ריצת האלגוריתם

באלגוריתם סינון שיתופי יבדקו תכונות הבאות:

- אחוזי קורסים שפולט האלגוריתם סינון שיתופי התואמים לנתוני בדיקה, במודלים שונים, באופי מדידת מרחקים.
- אחוזי קורסים שפולט האלגוריתם סינון שיתופי התואמים לנתוני בדיקה, המודלים השונים באופי הטקסטים המשמשים את האלגוריתם (שינוי קורפוסים).
- מהירות ריצת האלגוריתם

באלגוריתם יצירת מקבצים יבדקו תכונות הבאות:

- אחוזי קורסים שפולט האלגוריתם הכללי (ההיברידי) התואמים לנתוני בדיקה, בשינוי משקולות על נתונים דמוגרפיים.
- אחוזי קורסים שפולט האלגוריתם הכללי (ההיברידי) התואמים לנתוני בדיקה, בשינוי מספר מקבצים.

בשכבת השרת יבדקו תכונות הבאות:

- הרצת מערכת על רשת מחשבים במעבדה
- הרצת מערכת על מקבץ בענן

## 2.2 תכונות שלא יבדקו

- המערכת לא תיבדק על שום פלטפורמה אחרת השונה מפלטפורמת WEB.
- באלגוריתם היברידי לא יבדקו תוספות של אלגוריתמים אחרים ממה שצוינו בדוח: סינון מבוסס תוכן, סינון שיתופי ו K-MEANS .

## 2.3 דרישות סביבה

- הבדיקות יבוצעו באותה סביבה ותחת אותן דרישות משאבים כפי שהוגדר ב SRD .



## 2.4 אירעי בדיקה

### 2.4.1 בדיקת הזנת נתונים בעת רישום משתמש במערכת:

תכלית:

בדיקת של אי התאפשרת הכנסת נתונים, הלא תואמים לתבניות של שדות טופס הרשמה. קלטים:

מייל, ארץ מגורים, גיל, רמת השכלה ומגדר.

צפי לפלט וקריטריון מעבר:

אם כל השדות ימלו בתואם לתבניות שהוגדרו לשדות של תפס ההרשמה המשתמש יועבר לדף הבית. אחרת המערכת תוציא הודעה שגיע מודגשת בצבע אדום ליד השדה שלא הוכנס כראוי.

נהלי בדיקה :

הרצת המערכת

הזנת נתונים כמשתמש חדש שנרשם למערכת

לחיצה על כפתור סיום בטופס הרשמה

### 2.4.2 פונקציונליות של כפתור בקשת המלצות לקורסים

תכלית:

בדיקת אי התאפשרות של לחיצה על כפתור, למשתמשים שלא הכניסו קורסים שעשו בעבר.

קלטים:

אין קלטים

צפי לפלט וקריטריון מעבר:

אם משתמש הכניס אפילו קורס אחד שלמד בעבר, יפתח הכפתור.

נהלי בדיקה :

הרצת המערכת

הזנת נתונים כמשתמש חדש שנרשם למערכת

ניסיון לחיצה על הכפתור

הזנת קורס שנלמד בעבר

ניסיון לחיצה על כפתור

### 2.4.3 בדיקת נכונות פלט אחרי לחיצה על כפתור בקשת המלצות לקורסים

תכלית:

בדיקת פלט תקין ומובן למשתמש.

קלטים:

לחיצה על כפתור בקשת המלצות לקורסים

צפי לפלט וקריטריון מעבר:

אחרי לחיצה על כפתור אחרי כ 10 שניות אמור להופיע פלט תקין של מספרי קורסים ותכונותיהם.

נהלי בדיקה :

הרצת המערכת

לחיצה על כפתור בקשת המלצות לקורסים

לחקות לפלט עד 20 שניות

לבדוק תקינות הפלט כמו שהוגדר במערכת

### 2.4.4 בדיקת נכונות פלט אחרי לחיצה על כפתור בקשת המלצות לקורסים לפי נושא

תכלית:

בדיקת פלט תקין ומובן למשתמש.

קלטים:

לחיצה על כפתור בקשת המלצות לקורסים לפי נושא

צפי לפלט וקריטריון מעבר:

אחרי לחיצה על כפתור אחרי כ 5 שניות אמור להופיע פלט תקין של מספרי קורסים ותכונותיהם.

נהלי בדיקה :

הרצת המערכת

לחיצה על כפתור בקשת המלצות לקורסים לפי נושא

לחקות לפלט עד 10 שניות

לבדוק תקינות הפלט כמו שהוגדר במערכת

### 2.4.5 בדיקת מהירות ריצת האלגוריתם

תכלית:

בדיקת מהירות הוצאת פלט של המלצות לקורסים אחרי לחיצה על כפתור.  
קלטים:  
לחיצה על כפתור בקשת המלצות לקורסים  
צפי לפלט וקריטריון מעבר:  
אחרי לחיצה על כפתור אחרי כ 10 שניות אמור להופיע פלט תקין של מספרי קורסים  
ותכונותיהם.  
נהלי בדיקה :  
הרצת המערכת  
התחלת מדידת זמן בסטופר במקביל ללחיצה על כפתור בקשת המלצות לקורסים  
לחקות לפלט  
כפלט מופיע לעצור סטופר

#### 2.4.6 בדיקת אחוזי קליעת הפלט של האלגוריתם לנתוני בדיקה

תכלית:  
לבדוק עד כמה האלגוריתם מוציא המלצות לקורסים שמשתמשים ירצו באמת לקחת על  
סמך נתונים אמיתיים מהעבר.  
קלטים:  
נתוני אימון  
נתוני בדיקה  
צפי לפלט וקריטריון מעבר:  
לפחות 20 אחוז מפלט האלגוריתם צריך להיות תואם לנתוני בדיקה  
נהלי בדיקה :  
בניית מערכת בדיקה, אשר מריצה את האלגוריתם כל פעם על משתמש אחר בתוך נתוני  
אימון ומצליבה את הפלט לנתוני בדיקה.

#### 3. בדיקות פונקציונליות ושימושיות.

\*\*\*בשלב זה לא נעשתה חשיבה מעמיקה על הUI ולכן בשלב זה הUI הוא אינו ברור ולא מותאם  
למשתמש. ולכן בשלב זה לא נבצע בדיקות אלו. \*\*

#### 4. לוח זמנים

