



---

# פרויקט קורס בניית מערכות

## ממוחשבות חלק 3 - צד שרת

---

קבוצה 14

git - <https://github.com/Roycko/web-project-g14-Final.git>

**FEBRUARY 22.2022**

מגישים:

314960782

203458617

316173269

311451231

## תוכן עניינים

2	יצירת בסיס נתונים.....
3	יצירת מחלקות.....
3	שאלות SQL.....
3	שאלות בחירה.....
4	שאלות הוספה.....
4	שאלות עדכון.....
5	שאלות מחיקה.....
5	תוכן דינמי – מימוש עמודי התבנית.....
6	מימוש טפסים.....

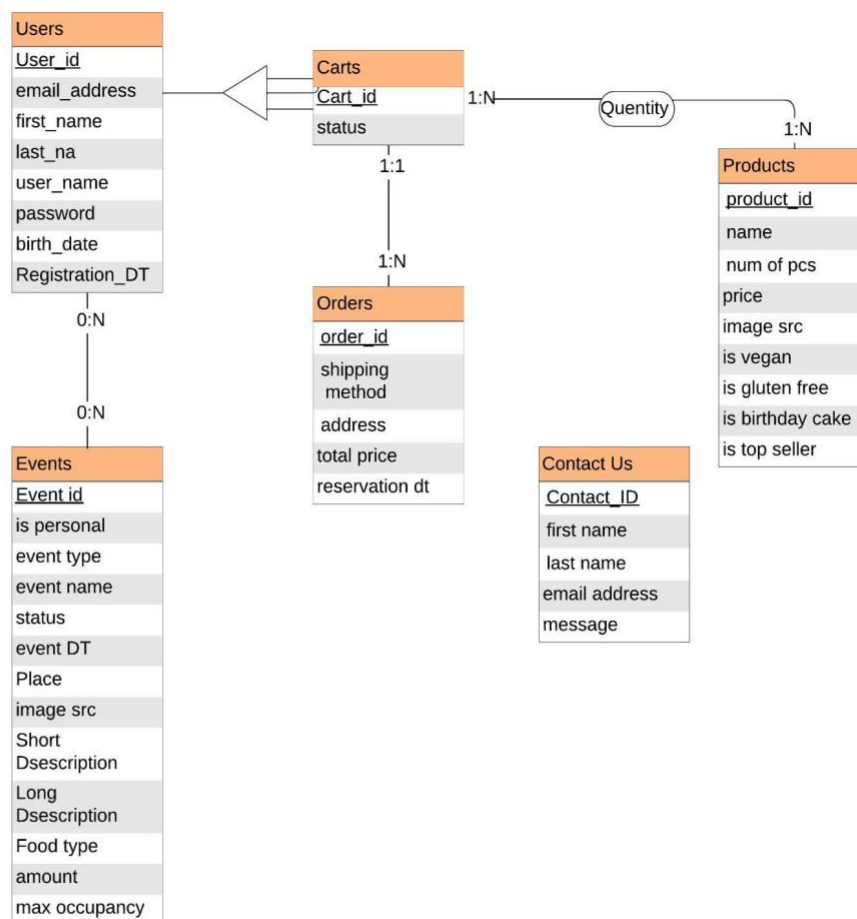
## יצירת בסיס נתונים

על מנת לקשר בין כלל עמודי האתר, וכן לשמור את הנתונים על גבי השרת אפיינו בסיס נתונים המקיף את כלל המהות העסקית של האתר.

על מנת ליצור את בסיס הנתונים ולהכניס את הנתונים אל הסכמה הטבלאית יש להריץ את ה-**script** הנמצא בקובץ בשם **build db** תחת תיקיית '**SQL script**'.

לצורך נוחות יצרנו משתמש דיפולטיבי **user\_name:test\_user ; password:AB123123**

## תרשים ה-ERD



## הנחות ונקודות מהותיות -

נאכפו כחלק מתכנון פונקציונליות האתר:

- לקוח שאינו רשום לאתר אינו יכול לבצע רכישה, להוסיף מוצרים לעגלה, ולהירשם לאירוע.
- לקוח שאינו רשום יכול לבצע פנייה לצוות המאפיה באמצעות טופס בעמוד 'Contact Us'
- ללקוח רשום קיימת בכל זמן נתון עגלה אחת בלבד בסטטוס 'פתוח', ואליה יצטרפו כלל המוצרים אשר הוא מעוניין בהם. בעת תשלום העגלה נסגרת ונשמרת לצורך מידע היסטורי וניתוחים עסקיים, ובמקומה נפתחת עגלה חדשה במספר סידורי עוקב.
- הרשמה לאירוע קיים מותנית במקום פנוי (מספר הרשומים  $> \text{max capacity}$ ).

## יצירת מחלקות

על מנת לשמור על סדר ארגוני נכון בקוד, וכן לאפשר גישה מסודרת ומודולרית אל בסיס הנתונים אפיינו מחלקות מתווכות שונות ורק דרכן ביצענו שאילתות אל מסד הנתונים.

יצרנו 2 מחלקות אובייקטים המתארות את הישויות המרכזיות בבסיס הנתונים שלנו:

- User - מחלקת אובייקט אשר מאפשרת גישה לכלל מאפייני הלקוח באמצעות שיטות `get` ו-`set`.
  - Product - מחלקת אובייקט אשר מאפשרת גישה לכלל שדות המוצר באמצעות שיטות `get` ו-`set`.
- כמו כן יצרנו מחלקה ראשית בשם `General` דרכה נבצע את השאילתות המורכבות יותר, וכן נמדל את מהות התהליכים העסקיים הדורשים התממשקות עם בסיס הנתונים.
- כלל המחלקות נמצאות תחת תקייה בשם `Entities -> Utilities`.

## שאילתות SQL

### שאילתות בחירה

- מנגנון חיפוש - בעמוד התפריט ניתן לבצע חיפוש באמצעות שדות רלוונטים בראש העמוד. החיפוש מתבצע ע"י פונקציית `Menu_Search` אשר באמצעות ארגומנטים רלוונטים הזהים לפילטרים, מחזירה את כלל המוצרים העומדים בפילטר הנדרש.
- שליפת עגלה - עבור כל לקוח רשום בכל זמן נתון קיימת עגלה יחידה בעלת סטאטוס "פעיל". באמצעות הפונקציה `getCart` העגלה האקטיבית מוצגת עבור הלקוח המחובר, וכן כלל המוצרים הקיימים בעגלה זו עם הכמויות.

- שליפת מחיר - בעזרת פונקציה `getCartPrice` שבה לוקחים את כל המוצרים בעגלה וסוכמים עבורם את המחיר\*הכמות של כל מוצר.
- שליפת אירועים - עבור כל לקוח רשום מוצגים האירועים אליהם הוא נרשם וגם האירועים הפרטיים שלו. על מנת שהמשתמש יראה אירוע בדף הפרופיל שלו, האירוע צריך להיות עתידי (תאריך של האירוע גדול מהרגע).

### שאלות הוספה

- הרשמת משתמש - על מנת להירשם לאתר אנו יוצרים אובייקט חדש ע"י שיטה בונה במחלקת `Users`, וכן עבור האובייקט החדש שנוצר (בהתאמה לשדות המתקבלים בטופס הרלוונטי) אנו מבצעים הרשמה באמצעות השיטה `registerUser`. כמו כן, מתבצע סיוע בפונקציות הבודקות כי המשתמש אינו קיים בבסיס הנתונים ( `is_Email_Not_Availale/ is_user_pass_not_available` )
- הוספת מוצר לעגלה - בכדי להוסיף מוצר לעגלה מתבצע בעזרת שליפת העגלה הקיימת בשימוש הפונקציות `getCart` ו-`insertProductToCart`. רק במידה והמוצר אינו קיים כבר בעגלה המוצר מתווסף אל העגלה בצירוף הכמות הרצויה.
- ביצוע תשלום - בכדי לשמור את נתוני התשלום ל-DB הוספנו את הפונקציה `savePaymentToDB` - פונקציה זו תחילה מוסיפה את הנתונים לטבלת `Orders`, ולאחר מכן קוראת לפונקציה `closeCart`, אשר סוגרת את עגלת הקניות הנוכחית והופכת את הסטאטוס שלה ל-`inactive`.
- הוספת עגלה חדשה - כאשר משתמש נכנס לדף הבית, המערכת בודקת האם קיימת עגלה עבורו, אם כן היא עשויה להיות מלאה או ריקה. אם לא, מאחר וביצע הזמנה ותשלום עבור העגלה הקודמת, תיווצר לו עגלה חדשה ריקה.
- הוספת אירוע - כל משתמש יכול להירשם לאירועים קיימים. כאשר לקוח נרשם לאירוע, נוצר שיוך בין האירועים שהמשתמש נרשם אליהם לבין המשתמש עצמו. שיוך זה מוצג בטבלת `users_in_events`.

### שאלות עדכון

- עדכון פרטי המשתמש - כאשר נכנסים לדף הפרופיל ע"י לחיצה על השם של היוזר ב-`header`, ישנו תופס לעדכון חלק מהשדות של יוזר. מתבצע בעזרת הפונקציה `UpdateUser`.
- עדכון מוצר בעגלה - בעמוד העגלה ניתן לעדכן את כמות הרצויה ממוצר מסוים. מתבצע בעזרת שימוש בפונקציה `updateCart`.

- עדכון סטטוס עגלה - בכדי לשמור מידע היסטורי על עגלות הלקוחות, נקפיד על שמירת עגלה יחידה בסטטוס אקטיבי. בסיום רכישת המוצרים בעגלה תסגור, ובמקומה תפתח עגלה חדשה. מתבצע בשימוש הפונקציה closeCart.

## שאלות מחיקה

- מחיקת מוצר מעגלה - ניתן לבצע מחיקה של מוצר מעגלת הקניות. מתבצע בשימוש הפונקציה removeProduct - הפונקציה עוברת על כל הפריטים ובודקת האם הפריט נמצא ברשימה כעת לעומת הרשימה מתחילת התהליך, במידה ולא הכוונה היא שהמשתמש מחק את הפריט ולכן השאלתה תמחק את הפריט זה.
- מחיקת הרשמה לאירוע - דרך עמוד הפרופיל של הלקוח, לצד כל אירוע אליו המשתמש רשום קיים לחצן (אייקון אשפה). בעת הלחיצה נשלח טופס שמממש פונקציה deleteMyEvent.

## תוכן דינמי – מימוש עמודי התבנית

בכדי ליישם את כלל פונקציונליות האתר בהתאם לאפיון שתיארנו בחלק א' ו-ב' בפרויקט, תוך שימוש במידע הקיים בבסיס הנתונים מימשנו את עמודי האתר כדינמיים. מרבית פונקציונליות האתר (הוספת מוצרים לעגלה, תשלום, הרשמה/ יצירת אירוע) מחייבים התחברות למשתמש באתר. עמודי האתר הדינמיים הינם:

1. Header - בכל עמודי האתר מופיע, ומתבצעים בו שינויים בהתאם לאם הלקוח מחובר. בצד שמאל ב-header מופיעים חיוויים למשתמש בנוגע לפעולות שביצע (האם ההתחברות הצליחה/הוספת מוצר לעגלה הצליחה/ הרשמה לאירוע וכו). עבור לקוח מחובר - מופיעים שם המשתמש המאפשר כניסה לאיזור האישי, וכן כפתור יציאה. עבור לקוח שאינו מחובר - מופיע אפשרות להתחברות. מצד שמאל לאייקון הבית מופיעות הודעות המציגות חיוויים לפעולות הלקוח באופן ידידותי וברור (נשלח מצד שרת ע"י שימוש ב – flash)
2. עמוד הבית - עמוד המציג את הסטנדרט הגבוה של האתר והעסק. מציג טעימה מהמאפים המומלצים, וכן את האירועים שניתן לתאם דרך האתר. מבצע שליפה של אירועים ומוצרים מייצגים מבסיס הנתונים.
3. אירועים - מציג את האירועים השונים שניתן לתאם דרך האתר (סדנאות, קייטרינג ועוד). עמוד זה מכיל שני אופציות פונקציונליות מרכזיות:

- הרשמה לאירוע קיים - בחירה והרשמה של אירוע מהקולקציה הקיימת באתר במידה וקיים מקום פנוי באירוע.

- יצירת אירוע פרטי - תיאום אירוע פרטי למשתמש המחובר.

4. תפריט - מאפשר חיפוש בקולקציית המאפים של העסק ע"פ קטגוריות מוגדרות, וכן לבצע בחירת כמות והוספה של פריטים לעגלת הקניות. שליפת הנתונים בהתאם לפלטור הנדרש מתבצעת בעזרת query-parameters.

5. עגלת קניות - ניתן לגשת לעמוד זה ע"י לחיצה על ה cart icon המופיע ב- header. במידה והלקוח לא מחובר לאתר, קופץ חלון ההתחברות. ניתן לערוך את כמויות המוצרים שנרכשו, לצפות במחיר הכולל של העגלה, וכן לעבור אל עמוד התשלום.

6. תשלום -עמוד זה מהווה סיכום להזמנתו של המשתמש וכן רכישה. בעמוד זה המשתמש יראה את סה"כ התשלום על הפריטים שבחר, והמשתמש יוכל לבחור כיצד ברצונו לקבל את הפריטים - איסוף עצמי או הזמנה (אם המשתמש בחר הזמנה הוא אף יתבקש להזין כתובת אליה יגיע המשלוח). בנוסף, המשתמש יזין את פרטי התשלום (כרטיס אשראי, קוד סודי ותוקף הכרטיס). במידה וכל הפרטים שהוזמנו עומדים בפורמט שהוגדר, ההזמנה תתקבל בהצלחה, העגלה הנוכחית תיסגר והמשתמש יקבל חיווי על ביצוע ההזמנה.

7. התחברות למשתמש - ניתן לגשת לעמוד זה ע"י לחיצה על ה person icon המופיע ב- header. מאופיין כחלון צף בו ניתן להכניס שם משתמש וסיסמה. במידה ולא משתמש קיים ניתן לעבור אל עמוד 'הרשמה לאתר'. מופיע רק כאשר הלקוח אינו מחובר.

8. הרשמה לאתר - מאפשר הרשמת משתמש חדש לאתר.

9. איזור אישי - מופיע אך ורק ללקוחות קיימים, ודרכו ניתן לערוך את השדות של המשתמש.

## מימוש טפסים

מימשנו באתר מספר טפסים:

1. התחברות לאתר - טופס קצר המאפשר הכנסת שם משתמש וסיסמה לצורך התחברות לאתר. מוודא כי שם המשתמש והסיסמה לא תפוסים, וכן וולדציות נוספות. הוגדר כ-Post כי מעביר מידע מוצפן (סיסמה)

2. הרשמה לאתר - על מנת לבצע רכישות יש צורך ליצור משתמש חדש לאתר. אכיפת וולדיות השדות מתבצעת בצד הלקוח בהיבט של טיפוס תואם ומאפייני קלט שהוגדרו כרלוונטיים (אורך קלט, מבנה, וכו'). מתבצע בעזרת JS שמוודא את מילוי ה-Mandatory fields. כמו כן התבצעה אכיפה בצד השרת כי שם המשתמש אינו קיים בבסיס הנתונים בצימוד הסיסמה (בכדי לאפשר כניסה באופן חח"ע), וכן כי לא קיים משתמש עם האימייל שהוכנס. הוגדר כ-Post כי מיועד להזנה ל-DB.
3. יצירת אירוע - המשתמש ממלא טופס עם הפרטים של האירוע הפרטי שהוא רוצה לעשות. המשתמש יכניס את שם האירוע, תאריך האירוע ופרטים נוספים. הטופס הוגדר כ-Post מאחר ומיועד להזנה ל-DB.
4. תשלום - לאחר שהמשתמש בחר לסיים את הקנייה, הוא מועבר לטופס התשלום. בטופס זה המשתמש ממלא את אופן קבלת הפריטים (איסוף עצמי או משלוח) וממלא פרטים בהתאם. בנוסף המשתמש מזין את פרטי התשלום שלו: מספר כרטיס אשראי, תוקף וקוד סודי. את פרטים אלו אנו מזינים ל-DB, ולכן פעולה זו הוגדרה כ-Post.
5. טפסי תפריט - כל אחד מלשוניות המוצרים בתפריט הינו טופס. ניתן לשלוט על הכמות הרצויה ולהוסיף את המוצר לעגלה. מאחר והלשוניות נוצרו בלולאות for, ונשלחים כולם אל אותו עמוד - כל אחד מהכפתורים מכיל מספר מזהה שתואם אל מספר המוצר המופיע, וכך ניתן לגשת אל הטופס הרלוונטי. כל טופס הוגדר כ-Post כי מיועד להזנה ל-DB.
6. טופס חיפוש - ניתן לבצע חיפוש בקטלוג המאפייה באמצעות טופס get אשר מאפשר לבחור באמצעות radio button פילטור יחיד רלוונטי. באופן דיפולטיבי מופיעים כלל המוצרים (ALL). הוגדר כ-get כי מבצע שליפה מה-DB.
7. הרשמה לרשימת מיילינג - מוסיף ל-DB של contact\_us בקשה להרשמה לרשימת מיילינג למייל שהוכנס. הוגדר כ- post מאחר ומוסיפים ל-DB רשומה.
8. עדכון העגלה - מיועד לביצוע עדכון פרטי המוצר בעגלה (מחיקת פריט או שינוי כמות). הוגדר כ- post מאחר ומשנה רשומה ב-DB.
9. עדכון פרטי משתמש - מיועד לביצוע עדכון פרטי המשתמש : שם פרטי, שם משפחה, וסיסמה. הוגדר כ- post מאחר ומשנה רשומה ב-DB.
10. מחיקת אירוע - מתבצע בעזרת לחיצה על אייקון האשפה במסך הפרופיל לצד כל אירוע אליו משויך המשתמש. מוגדר כ-POST מכיוון שמבצע שינוי ב-DB.
11. צור קשר - הוגדר כ-Post מאחר ומיועד להזנה ל-DB.