**בניית מערכות ממוחשבות מבוססות WEB**

**- פרויקט הקורס -**

חלק 3 – צד השרת



מגישים:

מאי דורי 203264874

הילית חזן 316421304

שוהם שבת 308160456

ליאור נוימן 312128325

7.2.2021

תוכן עניינים

[הנחות 2](#_Toc63620195)

[מידע כללי על האתר 2](#_Toc63620196)

[יצירת בסיס נתונים 3](#_Toc63620197)

[יצירת מחלקות 3](#_Toc63620198)

[שאילתות **SQL** 3](#_Toc63620199)

[תוכן דינמי – מימוש עמודי התבנית 5](#_Toc63620200)

[מימוש הטפסים 8](#_Toc63620201)

# הנחות

1. ה-Method בעמודי הטפסים הינו POST מאחר ואנו רוצים לשמור על רגישות המידע של לקוחות האתר.
2. מידע הוא נכס ולכן לא נרצה לאפשר מחיקה של נתונים מתוך ה-DB, נאפשר רק הוספה ועדכון.
3. אנו לא שומרים פרטי אשראי של לקוחות. התשלום הסופי על ההזמנות מתבצע באתר חיצוני של חברת האשראי, בו גם תתבצע הבדיקה האם פרטי האשראי תואמים לתנאים.

# מידע כללי על האתר

העסק עליו בחרנו לפתח אתר נקרא illusion\_Bakery, עסק קטן וביתי המוכר עוגיות, עוגות ומאפים. האתר כתוב בשפה העברית לפי בקשת בעלת העסק ומתאים עצמו לכל גודל מסך מחשב.

# יצירת בסיס נתונים

בסיס הנתונים של האתר מכיל 10 טבלאות המכילות נתונים על המוצרים הקיימים באתר, על המשתמשים ועל ביצועיהם באתר:

|  |  |  |
| --- | --- | --- |
| **מספר** | **שם בסיס הנתונים** | **באילו עמודים משתמשים בו** |
| 1 | Contact\_me | צור קשר |
| 2 | Orders | עגלת קניות, תשלום |
| 3 | Product\_in\_order | עגלת קניות |
| 4 | Products | החנות שלי: עוגות, עוגיות, מארזים |
| 5 | Recipe\_ingredient | אופים איתי |
| 6 | Recipes | אופים איתי |
| 7 | Recommendations | לקוחות מתוקים |
| 8 | Users | הרשמה, תשלום |
| 9 | Events | אירועים |
| 10 | Wishlist | עגלת קניות |

# יצירת מחלקות

כל ההתנהלות של האתר אל מול בסיס הנתונים מתבצעת דרך המחלקות המתווכות. בנוסף למחלקת DBManager שהייתה נתונה, יצרנו עוד 2 מחלקות מתווכות נוספות:

מחלקה ראשונה, DB \_Queries, מחלקה אשר מכילה את כל השאילתות הרלוונטיות להתנהלות מול בסיס הנתונים עבור כל העמודים שמכילים נתונים כלליים על השימוש באתר כגון הרשמה.

מחלקה שניה, DB\_Forms\_Queries, מחלקה אשר מכילה את כל השאילתות הרלוונטיות לצורך כתיבה ועדכון של בסיס הנתונים עבור הטפסים הקיימים באתר.

מחלקה שלישית, DB\_Product\_Queries, מחלקה אשר מכילה את כל השאילתות הרלוונטיות להתנהלות מול בסיס הנתונים עבור כל העמודים אשר מציגים באתר מוצרים קיימים הניתנים לרכישה.

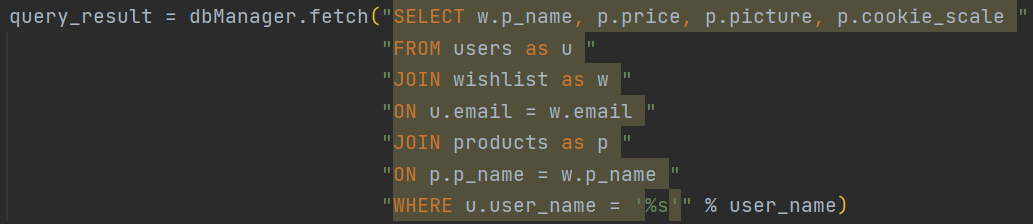
# שאילתות SQL

**כל סינון הדאטה מתבצע מול בסיס הנתונים באמצעות השאילתות השונות. לאחר קבלת המידע מהשאילתה, דאגנו לא לסנן את התוצאות באמצעות פייתון ולהשתמש בכלל המידע שהשאילתה החזירה.**

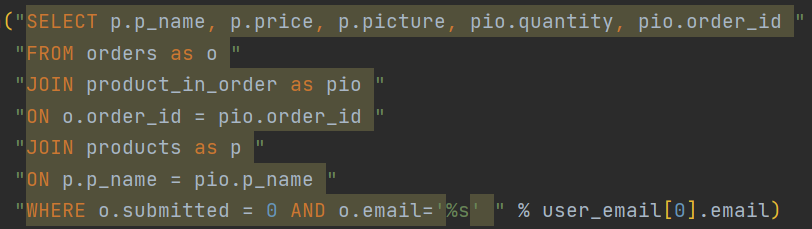
* שאילתות בחירה:  
  שאילתות fetch המבקשות נתונים מבסיס הנתונים:
  + בקשת כל פרטי המוצר לפי שם מוצר בפונקציה search\_products או כל המוצרים מסוג מסוים בפונקציה getAllProducts.
  + חילוץ כל העמודות מטבלת whishlist לפי כתובת המייל של המשתמש (עבור משתמש מסוים) בפונקציה isProductInUserWishlist.
  + חילוץ מייל המשתמש לפי שם המשתמש שלו, בפונקציה getUserEmail.
  + קבלת מספר ההזמנות מטבלת הזמנות עבור משתמש מסוים, אשר submitted=0, על מנת לדעת אם קיימת או לא הזמנה פתוחה של אותו משתמש (עגלת קניות עם לפחות מוצר אחד בפנים).
  + חילוץ שמות המוצרים אשר נמצאים בהזמנה מסוימת, עבור מוצר מסוים (על מנת לבדוק אם מוצר מסוים קיים בעגלת הקניות), בפונקציה is\_product\_in\_order:



* + בקשת פרטי המוצרים עבור משתמש באתר. השאילתה מכילה שני JOIN על שלוש טבלאות שונות:



* + בקשת פרטי המוצרים של עגלת קניות של משתמש מסוים.  
    הסינון מתבצע באמצעות שני JOIN על שלוש טבלאות, כך שעושים אגרגציה בין טבלת ההזמנות והמוצרים בהזמנות על מנת לדרוש submitted=0 (עגלת קניות נוכחית, במקום הזמנה שכבר התבצעה ושולמה). כמו כן, האגרגציה מתבצעת גם מול טבלת מוצרים על מנת לקבל את פרטי המוצרים שבסל (כדי שנוכל להציג אותם בעמוד עגלה):



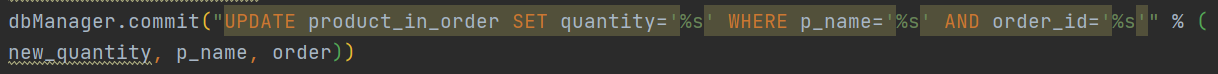
* שאילתות הוספה:
  + הוספת הזמנה חדשה לפי מייל המשתמש והתאריך הנוכחי (פתיחת הזמנה), בפונקציה addProductToOrder.
  + הוספת מוצר לטבלת המוצרים המועדפים (wishlist) עבור משתמש מסוים:



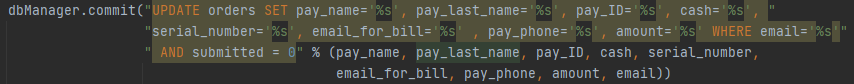
* + הוספת מוצר להזמנה מסוימת, לפי שם ההזמנה, שם המוצר, והכמות מאותו המוצר, בפונקציה addProductToOrder:



* שאילתות עדכון:
  + עדכון הסיסמא של משתמש בטבלת משתמשים, לפי המייל שלו, בפונקציה updatePassword.
  + עדכון כמות להזמנה של מוצר מסוים בעגלת קניות של משתמש:



* עדכון כל פרטי המשתמש בטבלת orders עבור הזמנה בתהליך (submitted=0) של משתמש ספציפי (לפי המייל שלו).



* שאילתות מחיקה:
  + מחיקת מוצר מטבלת ה-wishlist, עבור משתמש מסוים, בפונקציה removeProductFromWishlist:



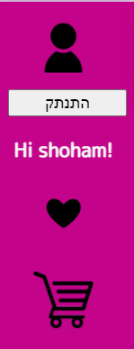
* + מחיקת מוצר מטבלת cart – הורדתו מסל הקניות של המשתמש.

# תוכן דינמי – מימוש עמודי התבנית

ערכו את העמודים הסטטיים שיצרנו בחלק הקודם של הפרויקט כך שהנתונים נלקחים מהטבלאות הרלוונטיות שבבסיס הנתונים. כך, הפכנו עמודים אלו לעמודים דינאמיים עם תוכן משתנה כתלות בבסיס הנתונים.

**1. סרגלי ה-nav (ראש העמוד), bar (בחלקו השמאלי של העמוד) footer (תחתית העמוד) וכותרת header**

ארבעה רכיבים (components) מייצגים ארבעה חלקים אשר מופיעים ברב עמודי האתר – שלושה סרגלים וכותרת העמוד. ההוספה של מרכיבים אלו מתבצעת בקובץ ה-base.html, אשר אליו רב עמודי ה-html של האתר עושים extends.

נוסף על כך, דרך רכיב ה-bar המשתמש מתחבר לאתר, יכול לבצע רישום ואף רואה את שמו מופיע על הסרגל לאחר התחברותו. ההתחברות לאתר נעשית דרך POST על מנת להעביר בצורה מאובטחת את פרטיו האישיים של המשתמש (שם משתמש וסיסמא). ב-bar נבדק ששם המשתמש נמצא בבסיס הנתונים וכי הסיסמא שהזונה נכונה. בהתאם לנתונים אלו, במידה וניסיון ההתחברות לאתר של המשתמש מצליח, ה-session של user\_name מעודכן להיות שם המשתמש וכך גם ה-session של logged\_in. ב-sessions אלו אנו משתמשים רבות באתרים הדינאמיים על מנת לחלץ נתונים מתאימים למשתמש המחובר לאתר.

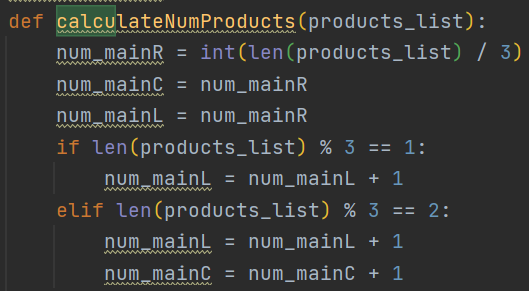
ה-bar דואג לשלוח לעמוד ה-html את ה-query בשם logout\_disp, אשר מייצג האם יש להציג או לא את כפתור ה"התנתק". רק במידה והמשתמש הצליח להתחבר לאתר, יוצג הכפתור. לאחר התחברות לאתר, שמו הפרטי של המשתמש מופיע בסרגל ה-bar (צד שמאל), למעט במקרה בו שם המשתמש הוא 10 תווים ומעלה, ואז מוחלף שמו ל-Friend מאילוצי מקום.

**2. עמוד מוצר**

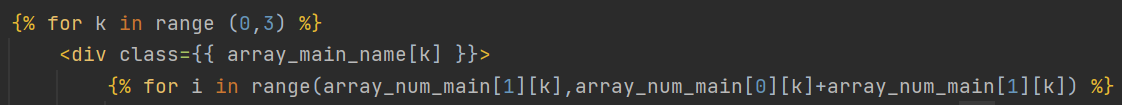
עמוד זה מציג מוצר בודד – עוגייה, עוגה או מארז יחיד. ההגעה לעמוד זה מתבצעת דרך ה-route: /product/<p\_name>, כך ש-p\_name מייצג את שם המוצר. בעזרת הפונקציה getProductDetails אנו מחלצים את פרטי המוצר הרלוונטי מטבלת ה-products שבבסיס הנתונים ושולחים אותם לעמוד html בתור queries.  
במידה והמשתמש מחובר לאתר (ה-session של logged\_in הוא True), הפונקציה קוראת לשתי פונקציות נוספות, isProductInUserWishlist ו-is\_product\_in\_order, אשר מחלצות מבסיס הנתונים את המידע – האם המוצר הנ"ל נמצא ברשימת המועדפים של המשתמש והאם נמצא בעגלת הקניות שלו ומוסיפות מידע זה ל-queries שנשלח ל-html.

בעמוד ה-html מוצג כל המידע אודות המוצר, הכולל את שמו, תמונה, מחיר ומספר עוגיות שחיתות (באמצעות לולאת for) בהתאם למופיע בבסיס הנתונים. במידה והמשתמש מחובר לאתר, יוצגו אפשרויות של להוסיף או להחסיר את המוצר מרשימת המועדפים ומעגלת הקניות, בהתאם למצב הנוכחי. ההורדה או ההוספה תעשה באמצעות route שיוביל לפונקציה מתאימה בקובץ ה-py (add\_product\_to\_wishlist, remove\_product\_from\_wishlist, add\_product\_to\_cart, remove\_product\_from\_cart) אשר יתחבר לבסיס הנתונים ויוסיף/ימחק את הרשומה המתאימה. לדוגמא, עבור משתמש שהמוצר כבר נמצא בסל הקניות – יוצג אייקון של הסרה מעגלה, המאפשר להוריד את המוצר מעגלת הקניות. לעומת זאת, במידה והמוצר איננו נמצא בסל, יוצג במסך אייקון של עגלה ריקה עם אופציה להוסיף את המוצר לסל תוך כדי בחירת הכמות המועדפת.

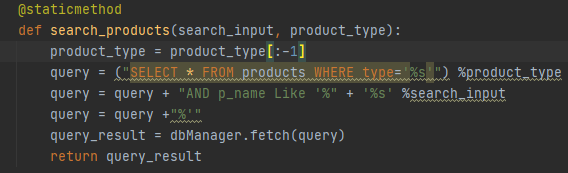
במידה והמשתמש איננו מחובר לאתר, אפשרויות אלו כלל לא יוצגו או יתאפשרו לשינוי. תוצג לו הודעה מתאימה המסבירה כי עליו להרשם או להתחבר לאתר אם ברצונו להוסיף את המוצר לעגלת הקניות או לרשימת המועדפים.

**3. עמוד חנות – חנות עוגיות/עוגות/מארזים**

עמוד זה מציג את החנות הרלוונטית – חנות עוגיות/עוגות/מארזים, בה מוצגים כל המוצרים המבוקשים מבסיס הנתונים. ההגעה לעמוד מתבצעת דרך ה-route: /shop/<product\_type>, זאת על מנת לדעת איזו חנות להציג (עוגיות/עוגות/מארזים). חנות המוצרים שלנו בנויה משלושה חלקים: שמאל, אמצע וימין. על מנת לדעת כמה מוצרים יש להציג בכל חלק, הפונקציה calculateNumProducts מחלצת מבסיס הנתונים את מספר הפריטים הכוללים שיש להציג בחנות, מחשבת ומחזירה (בעזרת פעולות מודולו %) כמה פריטים יש להציג בכל אחד מחלקי העמוד. לדוגמא, אם בבסיס הנתונים ישנם 8 עוגות, במידה ונכנס לחנות עוגות חלק (טור) שמאל יציג 3 עוגות, החלק האמצעי יציג 3 עוגיות והחלק הימני יציג 2 עוגיות בלבד (סך הכל 8 עוגיות).

בעמוד html ישנה לולאת for המתבצעת שלוש פעמים, עבור כל אחד מחלקי העמוד (שמאל, אמצע וימין). בכל אחד מהחלקים ישנה לולאת for נוספת אשר רצה לפי מספר הפריטים שחושב קודם, ומפיקה את מספר המוצרים המבוקש – הכולל שם מוצר, תמונה, מחיר ומד שחיתות, המחולצים מבסיס הנתונים.

כמו כן, בעמוד החנות ניתן לבצע חיפוש מוצר. החיפוש מבוצע ע"י שאילתת SELECT על טבלת PRODUCTS שמחזירה את כל המוצרים (בעמוד החנות הספציפי) שמכילים את התווים שהוכנסו לחיפוש בשם המוצר.



סוג המוצר (עוגייה/עוגה/מארז) מועבר לשאילתה כפרמטר לפי העמוד בו נמצא המשתמש. כל הפריטים שתואמים את החיפוש מוצגים במסך. במידה ולא נמצא כל פריט שתואם את הוראת החיפוש מוצגת למשתמש הודעה מתאימה.

**4. עמוד Whishlist – המועדפים שלי**

עמוד זה מציג את המוצרים המועדפים על המשתמש, המוצרים אשר לחץ על האייקון "לב" כאשר ביקר בעמוד המוצר שלהם. ההגעה לעמוד מתבצעת דרך ה-route: /wishlist. תחילה הפונקציה בודקת אם ה-session של logged\_in הוא True. במידה וכן, היא קוראת לפונקציה נוספת getWishlist אשר מחלצת מבסיס הנתונים את המוצרים המועדפים של אותו המשתמש.

המוצרים המועדפים מוצגים בצורה דומה לתצוגתם בעמוד shop. במידה ואין למשתמש מוצרים שנמצאים במועדפים תוצג לו הודעה מתאימה ("רשימת המועדפים שלך ריקה"). במידה ואין משתמש מחובר תוצג לו הודעה כי עליו להרשם או להתחבר על מנת לצפות ברשימת המועדפים.

כלל המוצרים בעמוד מוצגים עם אייקון של לב שבור, על מנת לתת למשתמש את האפשרות להוריד מוצרים מרשימת המועדפים. בעת לחיצה על אייקון הלב השבור, ה-route ישתנה ויוביל לפונקציה removeProductFromWishlist שתבצע הורדה של המוצר מרשימת המועדפים של אותו משתמש. בסוף הפונקציה ולאחר ההורדה מבסיס הנתונים, תחזיר הפונקציה את עמוד המועדפים שהיה, רק כעת מעודכן (ללא המוצר אשר הוסר).

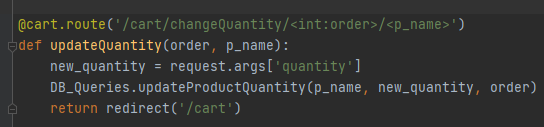


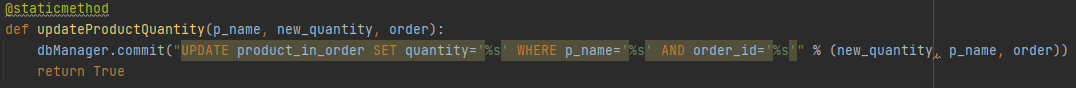
**5. עמוד עגלת קניות**

עמוד זה מציג את המוצרים שהמשתמש הוסיף לעגלת הקניות שלו (המוצרים אשר לחץ על אייקון עגלה תוך בחירת הכמות, כאשר ביקר בעמוד המוצר שלהם). ההגעה לעמוד מתבצעת דרך ה-route: /cart. תחילה הפונקציה בודקת אם ה-session של logged\_in הוא True. במידה וכן, היא קוראת לפונקציה נוספת getMyCart אשר מחלצת מבסיס הנתונים את המוצרים שנמצאים בעגלה של אותו המשתמש. עגלת הקניות של משתמש הינה רשומה בטבלת orders (הזמנה) אשר העמודה submitted שלה = 0. כאשר submitted=1 מדובר בהזמנה שהתבצעה והסתיימה, כאשר היא שווה ל-0, זוהי הזמנה בהתרחשות (עגלת קניות נוכחית). על כן, לכל משתמש ישנה עד רשומה אחת בטבלת orders עבורה submitted=0. במידה ואין למשתמש מוצרים בעגלת הקניות, תוצג לו הודעה מתאימה ("עגלת הקניות שלך ריקה"). במידה ואין משתמש מחובר תוצג לו הודעה כי עליו להירשם או להתחבר על מנת לצפות בעגלת הקניות.

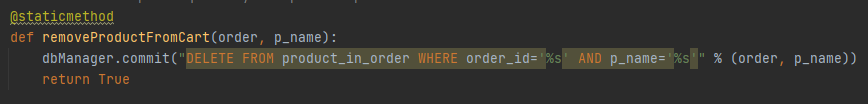
בעמוד ה-html מוצגים כלל המוצרים בעגלת הקניות של המשתמש. המוצרים מוצגים זה אחר זה באמצעות לולאת for, כאשר הנתונים שלהם נשלחו לעמוד html דרך ה-queries. המשתמש יכול בכל עת לשנות את הכמות של כל מוצר, כך שסך המחיר למוצר וסך מחיר העגלה מתעדכנים אוטומטית עם כל שינוי. השינוי מתבצע ישירות בבסיס הנתונים באמצעות שאילתת update. כמו כן, המשתמש רשאי ללחוץ על אייקון האיקס האדום על מנת להוריד מוצר מעגלת הקניות. לחיצה על אייקון זה תגרור הורדה של המוצר מבסיס הנתונים בטבלה המתאימה בעזרת שאילתת DELETE.

שאילתת עדכון הכמות- הפרמטרים מועברים כ-query parameters לפי המוצר והכמות הספציפיים שהזין המשתמש. העדכון מתבצע בטבלת products in order.





שאילתת מחיקת פריט- גם במצב זה הפרמטרים הרלוונטיים לשאילתה (מספר הזמנה ושם המוצר) מועברים כ- query parameters. השאילתה מתבצעת גם כן על טבלת products in order.



# מימוש הטפסים

באתר ישנם 4 עמודים שונים אשר ממומשים באמצעות שיטת POST, מתוך חשיבות השמירה על רגישות המידע של לקוחות האתר. הטפסים בעמודים אלו מבצעים כתיבה או עדכון של ה-DB הרלוונטי, באמצעות שאילתות הכתובות במחלקת "DB\_Forms\_Queries". כמו כן, יש שימוש ב-3 טפסים שונים אשר ממומשים באמצעות שיטת GET, מאחר ואין חשיבות לשמירה על רגישות המידע של לקוחות האתר. בטפסים אלו מתבצעים קריאה, עדכון ומחיקה מתוך ה-DB.

**1. טופס "צור קשר"**

לטופס זה המשתמש יכניס את הפרטים שלו ואת סיבת הפניה, לצורך יצירת קשר עם בעלת העסק:

* שם פרטי
* שם משפחה
* כתובת מייל
* סיבת פניה

לפני שליחת הטופס, ביצענו ולידציה אשר מוודאה שכל השדות מולאו כראוי. אם אחד השדות או יותר לא מולא כראוי – כאשר המשתמש ילחץ על שליחת הטופס תקפוץ לו הודעה בשדה הבעייתי. כאשר הטופס נשלח ללא בעיות – קופצת הודעה למשתמש "השליחה בוצעה בהצלחה".

כאשר המשתמש ילחץ על כפתור "שליחה", הנתונים המתקבלים מהטופס יוכנסו ל-DB בשם "contact\_me", אשר מכיל את הפניות המתקבלות ע"י המשתמשים על מנת שבעלת העסק תחזור אליהם. בעמוד contact.py שמרנו את הקלטים המתקבלים ע"י המשתמש והכנסנו אותם לשאילתה בשם "insertNewComment" אשר מבצעת הכנסה שלהם ל-DB, בתוספת התאריך והשעה שבה נעשתה הפניה, כלומר שבה נשלח הטופס.

**2. טופס "הרשמה"**

לטופס זה המשתמש יכניס את הפרטים שלו וסיסמא, לצורך התחברויות עתידיות לאתר:

* שם פרטי
* שם משפחה
* כתובת מייל
* שם משתמש
* סיסמא (+ סיסמת וידוי)

לפני שליחת הטופס, ביצענו ולידציה אשר מוודאה שכל השדות מולאו כראוי. אם אחד השדות או יותר לא מולא כראוי – כאשר המשתמש ילחץ על שליחת הטופס תקפוץ לו הודעה בשדה שמולא לא כראוי. כאשר הטופס נשלח ללא בעיות – קופצת הודעה למשתמש "השליחה בוצעה בהצלחה".

כאשר המשתמש ילחץ על כפתור "הירשם", הנתונים המתקבלים מהטופס יוכנסו ל-DB בשם "users", אשר מכיל את פרטי המשתמשים הרשומים באתר. בעמוד register.py שמרנו את הקלטים המתקבלים ע"י המשתמש והכנסנו אותם לשאילתה בשם "insertNewUser" אשר מבצעת הכנסה שלהם ל-DB. בנוסף, במידה והמשתמש כבר קיים ב-DB (לפי המייל) המשתמש יקבל הודעה שהיוזר כבר קיים במערכת.

**3. טופס "אירועים"**

לטופס זה המשתמש יכניס את הפרטים שלו ואת סוג האירוע בו הוא מתעניין, לצורך קביעת אירוע עתידי:

* סוג האירוע – בחירה מתוך תיבת אפשרויות
* שם מלא
* זכר / נקבה
* מייל
* טלפון
* הערות

לפני שליחת הטופס, ביצענו ולידציה אשר מוודאה שכל השדות מולאו כראוי. אם אחד השדות או יותר לא מולא כראוי – כאשר המשתמש ילחץ על שליחת הטופס תקפוץ לו הודעה בשדה שמולא לא כראוי. כאשר הטופס נשלח ללא בעיות – קופצת הודעה למשתמש "השליחה בוצעה בהצלחה".

כאשר המשתמש ילחץ על כפתור "שלח", הנתונים המתקבלים מהטופס יוכנסו ל-DB בשם "events", אשר מכיל את האירועים שמשתמשי האתר מתעניינים בהם על מנת שבעלת העסק תחזור אליהם וכך יוכלו לתאם אירוע. בעמוד events.py שמרנו את הקלטים המתקבלים ע"י המשתמש והכנסנו אותם לשאילתה בשם "insertNewEvent" אשר מבצעת הכנסה שלהם ל-DB, בתוספת התאריך והשעה שבה נעשתה הפניה, כלומר שבה נשלח הטופס.

**4. טופס "תשלום"**

לטופס זה המשתמש יכניס את פרטי המשלם ואת פרטי האשראי במידה ובחר לשלם באמצעותו, לצורך ביצוע תשלום וקבלת אישור תשלום:

* שם פרטי
* שם משפחה
* תעודת זהות
* מזומן / ויזה / מאסטרקארד
* כתובת מייל
* טלפון

במידה ובחר לשלם באשראי:

* מספר כרטיס אשראי
* CVV

לפני שליחת הטופס, ביצענו ולידציה אשר מוודאה שכל השדות מולאו כראוי. אם אחד השדות או יותר לא מולא כראוי – כאשר המשתמש ילחץ על שליחת הטופס תקפוץ לו הודעה בשדה שמולא לא כראוי. כאשר הטופס נשלח ללא בעיות – קופצת הודעה למשתמש "העסקה הושלמה בהצלחה".

כאשר משתמש מתחבר לאתר ויוצר עגלת קניות, אוטומטית נפתחת לו רשומה ב-DB בשם "orders", אשר מכילה מספר שדות ביניהם השדה Submitted אשר מקבל ערך 1 כאשר ההזמנה הושלמה וערך 0 כאשר היא טרם הושלמה. כלומר לאותו המשתמש יכולות להיות מספר רשומות ב-DB אשר מסמלות הזמנות שבוצעו (Submitted=1), אך רק רשומה אחת אשר מסמלת הזמנה בתהליך (submitted=0).

כאשר המשתמש ילחץ על כפתור "בצע תשלום" בעמוד התשלום, הנתונים המתקבלים מהטופס יעדכנו ב-DB "orders", את הרשומה של הזמנה שבתהליך של אותו היוזר. בעמוד payment.py שמרנו את הקלטים המתקבלים ע"י המשתמש והכנסנו אותם לשאילתה בשם "updatePayment" אשר מבצעת עדכון שלהם ב-DB, בתוספת התאריך והשעה שבה נעשתה הפניה, כלומר שבה נשלח הטופס. בנוסף, השאילתה משנה את שדה submitted להיות בעל ערך "1" כדי לציין שההזמנה הושלמה.

**5. עדכון כמות בעגלת קניות**

בעמוד זה מוצגים המוצרים אשר המשתמש הוסיף לעגלת הקניות שלו, אותם הוא מעוניין לרכוש בעתיד. ניתן לצפות בעמוד זה לאחר התחברות לאתר. כאשר משתמש מתחבר לאתר ויוצר עגלת קניות, אוטומטית נפתחת לו רשומה ב-DB בשם "orders".

השימוש אשר נעשה בטופס בעמוד זה הוא על מנת לשנות את הכמות הרצויה מכל מוצר שבעגלה. כל שינוי שיתבצע, הכולל עדכון כמות או הסרה של פריט, יעודכן ב-DB "Product\_in\_order" באופן מידי. שינוי הכמות יעדכן את מחיר העגלה לאחר לחיצה על מקש Enter.

**6. עדכון כמות והוספת מוצר (עוגייה/עוגה/מארז)**

בעמודי המוצר של החנות (עוגייה/עוגה/מארז) מוצגים המוצרים אשר המשתמש יכול לרכוש. השימוש אשר נעשה בטופס בעמוד זה הוא על מנת לשנות את הכמות הרצויה מכל מוצר שבעגלה או בעמוד 'המוצרים המועדפים עליי'. כל שינוי שיתבצע על מנת להכניס מוצר חדש לעגלת הקניות וכך להכניסו ל-DB "Product\_in\_order" או להכניס את המוצר לרשימת המוצרים המועדפים עליו וכך להכניסו ל-DB "Wishlist".

**7. חיפוש מוצר (עוגייה/עוגה/מארז)**

בעמודי המוצר של החנות (עוגייה/עוגה/מארז) מוצגים המוצרים אשר המשתמש יכול לרכוש. השימוש אשר נעשה בטופס בעמוד זה הוא על מנת לחפש מוצר מבין סך המוצרים המוצעים למכירה. כל חיפוש שמתבצע, מבצע קריאה מ-DB "Products".