



פרויקט בקורס WEB

חלק ג' – צד שרת



קבוצה 4

שלומי כהן - 312502966

רועי בניסטי - 203699137

עמית ינוביץ - 206026924

נופר דביר - 208205807

סער ויס - 311182695



רקע

בחלק זה הקמנו בסיס נתונים בהתאם לצרכי העסק והאתר, בהתבסס על חלק ב' של הפרויקט. יצירת בסיס הנתונים התבססה על מידע קודם שרכשנו בקורסים אחרים וכן לימוד עצמי באינטרנט. בנוסף, יצרנו שאילתות שישירתו את השימוש באתר ויסייעו לממשק בין האתר לבסיס הנתונים. את בסיס הנתונים חיברנו לאתר תוך הכנסה, שליפה, עדכון ומחיקה של נתונים. כמו כן, בחלק זה ישנו תהליך עיבוד מידע ע"י "לקיחת" מיקום המשתמש והחזרת מידע רלוונטי וייחודי, שמעודד רכישה ומקדם את המטרה העסקית של האתר והעסק.

בנוסף, החלטנו לממש את התוכן הדינמי אף על פי שמדובר במשימת רשות (בונוס), במטרה להרחיב את הידע שלנו ולצבור ניסיון והתנסות. את התוכן הדינמי עשינו באמצעות ejs באישורה של נעמה.

הוראות הפעלת האתר נמצאות בנספח בסוף העבודה.

מבנה התיקיות

את הקבצים השונים מהם מורכב האתר שלנו ריכזנו בתיקיות ע"פ סדר הגיוני וכן על פי הנלמד בהרצאה, ותיקיות אלו מרכזות בתיקייה אחת. להלן התיקיות:

DB – בתיקייה זו נמצא החיבור לבסיס הנתונים : db.js, db.config.js.

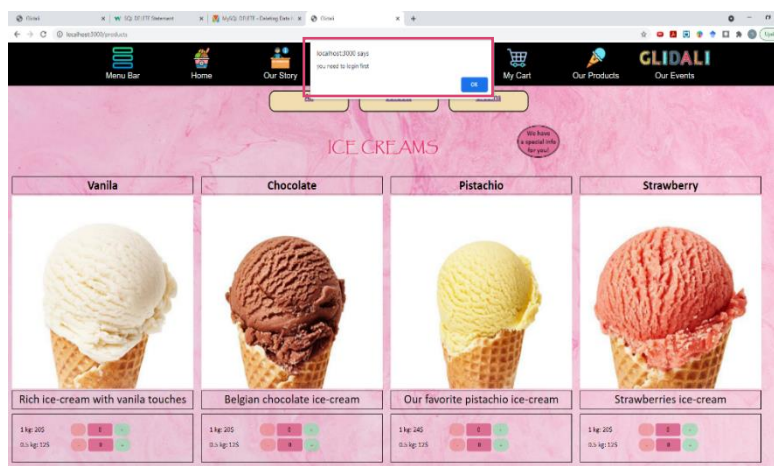
public – בתוך תיקייה זו נמצאות התיקיות מחלק ב': background, images, js, html, css.

views – מסמך התוכן הדינמי של עמוד התפריט Catalog.ejs.

קובץ server.js נמצא בתיקייה הראשית.

הנחות

1. פונקציית המיקום דורשת את אישור המשתמש לשימוש במיקום שלו. הפונקציה להחזרת המידע המוחזר למשתמש תופעל רק אם המשתמש אכן אישר.
2. לכל משתמש יש רשימת קניות אחת.
3. לקוח שירצה לשנות את שם user יצטרך להחליף את כתובת המייל, כמו בכניסה.
4. במידה ולקוח מנסה להעלות את כמות הגלידה שהוא רוצה להזמין, הוא צריך להיות member, כלומר בתחילת התהליך אין User מחובר, רק בעבור פעילות מסוימת על הלקוח להתחבר. במידה והלקוח לא מחובר ומנסה להעלות את כמות ההזמנה, הוא מקבל הודעה מתאימה.





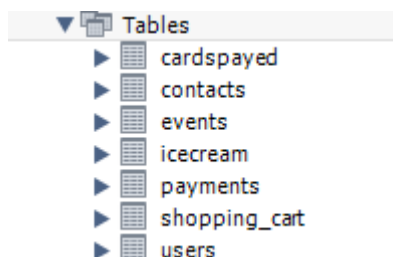
בניית וחיבור בסיס נתונים + מימוש טפסים

(חיבור לבסיס נתונים ושאלות SQL, מימוש טפסים)

חלק זה מתייחס לבניית בסיס הנתונים, בניית שאלות SQL ומימוש טפסים.

בחלק זה של חיבור לשרת התבססנו על החלק הקודם בו שמרנו את המידע שמשמש ספציפי מכניס בתוך ה-LocalStorage. בחלק זה שילבנו בין ה-LocalStorage לבין בסיס הנתונים בהתאם לצורך אולם השתמשנו ב-SessionStorage. שמרנו את הנתונים של כל משתמש בבסיס הנתונים ובעת חיבור לאתר אנו טוענים את הנתונים של משתמש ספציפי לתוך ה-SessionStorage ומבצעים את הפעולות הנדרשות. כמו כן ישנם נתונים שהעבודה איתם נעשית ישירות מול הבסיס נתונים כמו טפסים שונים עליהם נפרט בהמשך.

בסיס הנתונים:



1. טבלאות-

1. users – טבלה בה נשמרים פרטי כל משתמש בעת הרשמה. בטבלה נשמרים השדות הבאים:

- id – מספר מזהה משתמש. (מפתח ראשי).
- email – אימייל המשתמש.
- password – סיסמה.
- firstName – שם פרטי.
- lastName – שם משפחה.
- phone – טלפון.

2. shopping_cart – טבלה בה נשמרים פרטי עגלה של כל משתמש לפי האימייל שלו אחרי התחברות. בטבלה נשמרים השדות הבאים:

- id – מספר מזהה עגלה (מפתח ראשי).
- userID – מספר מזהה משתמש. (מפתח זר).
- selectedProducts – רשימת הקניות שמורה כ-String.
- creda – זמן יצירה.

3. payment – טבלת תשלומים בה נשמרים פרטי התשלום על עגלה. בטבלה נשמרים השדות הבאים:

- id – מזהה תשלום (מפתח ראשי).
- cartID – מזהה העגלה (מפתח זר).
- address – כתובת.
- city – עיר.



- idNumber – תעודת זהות משלם.
- creditCard – מספר אשראי.
- expDate – תאריך תפוגת כרטיס אשראי.
- paymentNumber – מספר תשלומים.
- cvv.

4. icecream – טבלה בה נשמרים פרטי גלידות לצורך שימוש דינאמי. בטבלה נשמרים השדות הבאים:

- id – מספר מזהה מוצר(מפתח ראשי).
- description – תיאור מוצר.
- Price1Kg – מחיר ל-1 ק"ג.
- PriceHalfKg – מחיר לחצי ק"ג.
- category – קטגוריה.
- image – תמונה של המוצר.

5. events – טבלה בה נשמרים פרטי אירועים שמשתמשים דורשים. בטבלה נשמרים השדות הבאים:

- id – מספר מזהה אירוע(מפתח ראשי).
- userFullName – שם מלא של משתמש.
- userEmail – מייל של משתמש.
- eventType – סוג אירוע.
- numOfParticipants – מספר משתתפים.
- eventDate – תאריך האירוע.
- details – פרטים נוספים על האירוע.

5. contacts – טבלה בה נשמרים פרטי ההודעה שהמשתמש מזין בדף צור קשר. בטבלה נשמרים השדות הבאים:

- id – מספר מזהה הודעה(מפתח ראשי).
- firstName – שם מלא של משתמש.
- email – מייל של משתמש.
- message – פרטי ההודעה.

6. cardspayed – טבלה בה נשמרים פרטי העגלות ששולמו. בטבלה נשמרים השדות הבאים:

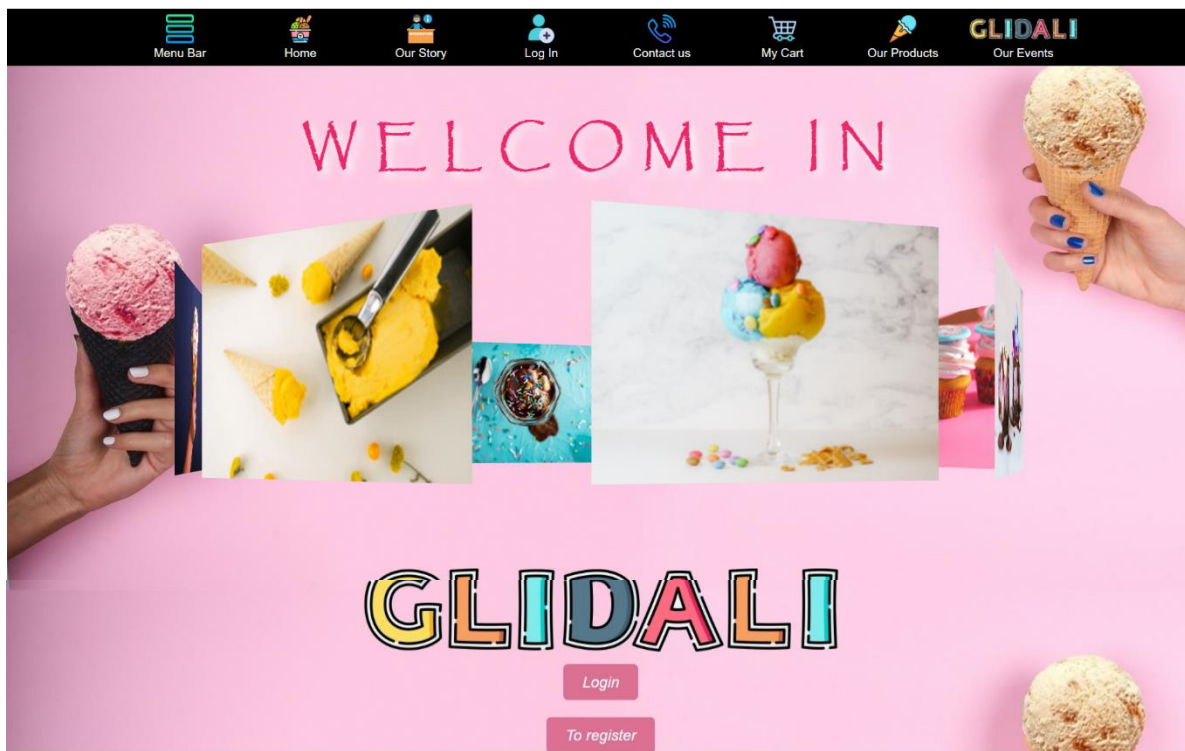
- id – מספר מזהה עגלה ששולמה(מפתח ראשי).
- userID – מספר מזהה של משתמש.
- selectedProducts – רשימת הקניות שמורה כ-String.
- creda – זמן יצירה.



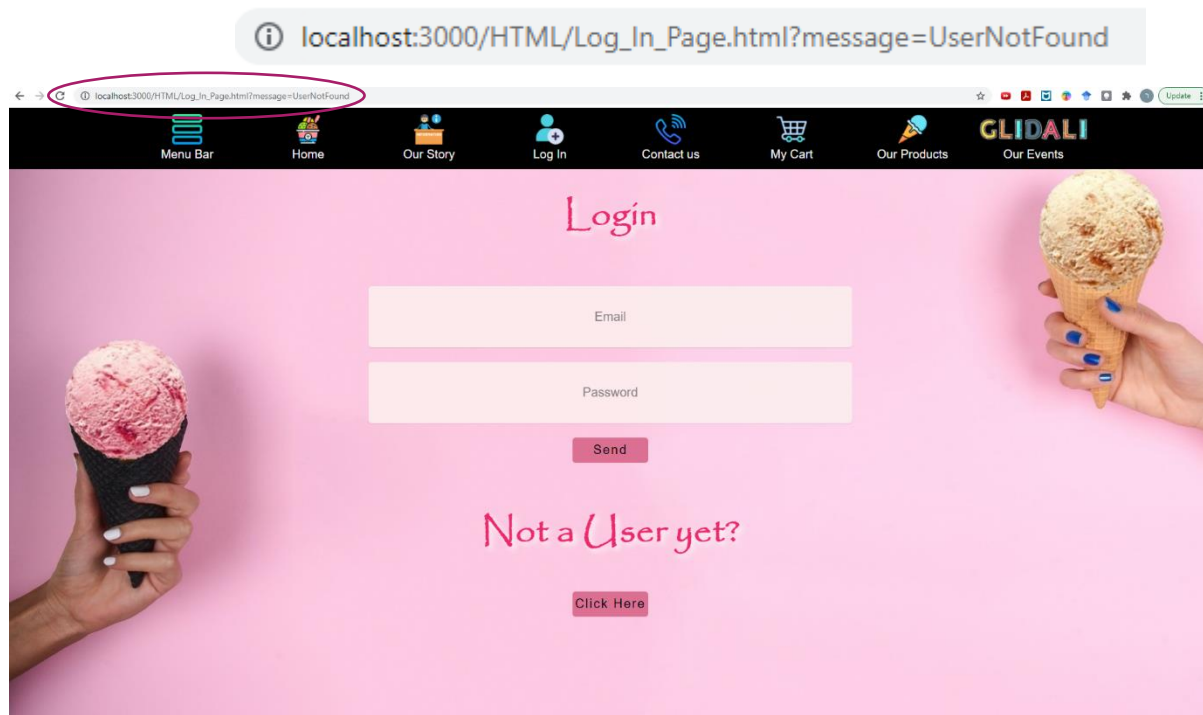
2. שאלות-

רצף פעולות סטנדרטי של משתמש באתר הוא כדלקמן:

1. משתמש נכנס לאתר (בכתובת `/http://localhost:3000`) ומגיע לעמוד הבית:



2. כבר בעמוד הבית יכול המשתמש להחליט להתחבר לאתר או להירשם אליו. בהנחה שהמשתמש אינו רשום ומנסה להתחבר השרת מנתב אותו לכתובת של משתמש לא נמצא:





3. המשתמש נכנס להרשמה:

Menu Bar Home Our Story Log In Contact us My Cart Our Products Our Events

Registration Page

First Name

Last Name

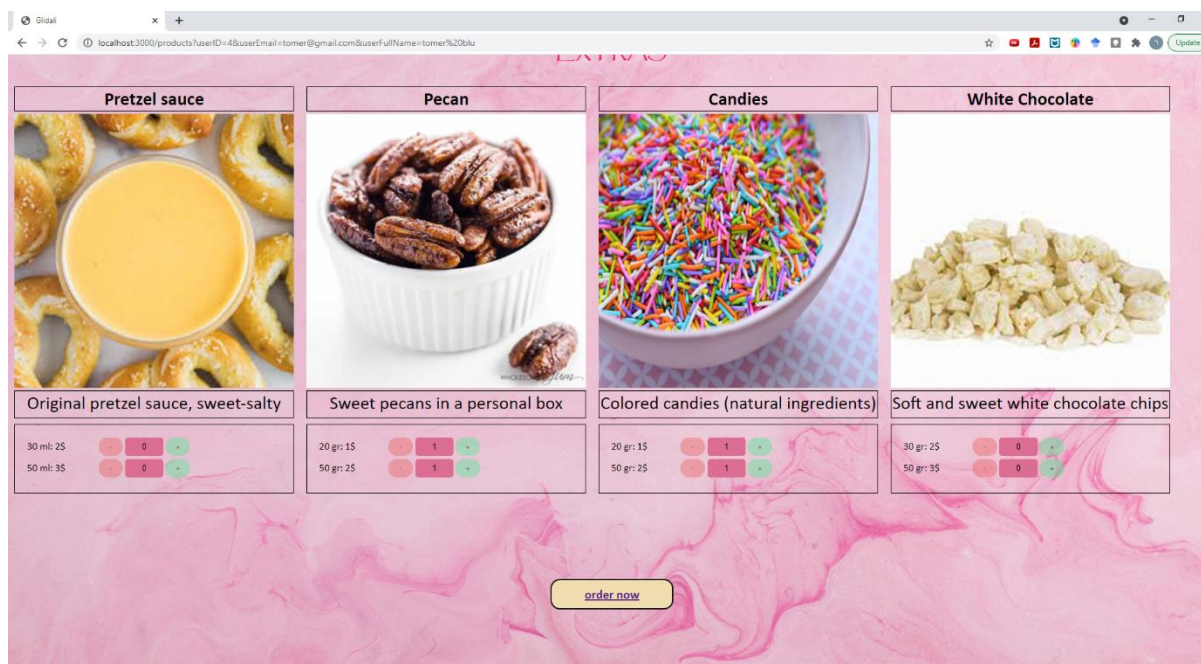
Phone Number

Your Email

Password

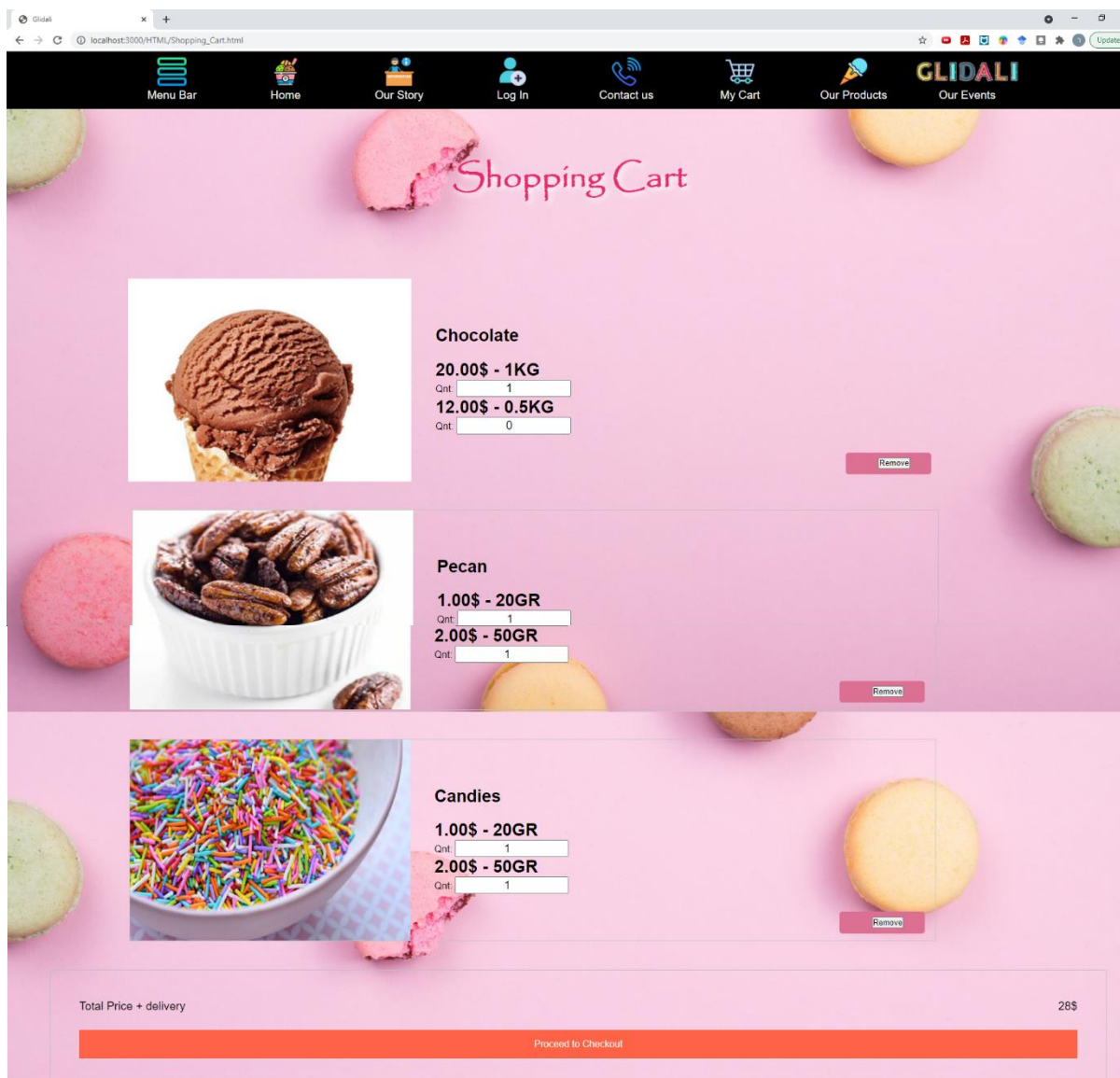
Send

4. המשתמש מכניס את הפרטים שלו ולוחץ על send והנתונים נשמרים בבסיס הנתונים (שאלתת הוספה). לאחר ההרשמה המשתמש צריך להתחבר לאתר תחת עמוד Login (שאלתת בחירה). בשלב זה נפתח התפריט למשתמש, בו הוא בוחר את המוצרים הרצויים. לאחר הבחירה רשאי ללחוץ על order now. בשלב זה נשמרת הבחירה של עגלת הקניות של המשתמש בבסיס הנתונים (שאלתת הוספה) תוך כדי שימוש בsessionStorage והעברת הנתונים לבסיס הנתונים. ניתן לראות כי בשורת הכתובת נמצאים פרטיו של המשתמש.





5. לאחר לחיצה על order now עובר המשתמש לעגלת הקניות:





6. לאחר לחיצה על Proceed to Checkout ממשיך המשתמש לעמוד התשלום:
במסך זה המשתמש מזין את פרטיו והנתונים נשמרים בSQL (שאילתת הוספה).

לאחר תשלום המשתמש על העגלה שלו נמחקת העגלה מטבלת shopping_cart ונתוניה מועברים לטבלת cardsPayed יחד עם פרטי המשתמש. זוהי שאילתה מורכבת המורכבת ממחיקה מטבלה אחת והכנסה לטבלה אחרת תוך כדי קיון שאילתות ולקיחת הנתונים מטבלת shopping_cart:

```
const sql = `
INSERT INTO cardsPayed SELECT * FROM shopping_cart where userID=?;
SET @cartID = (select id from shopping_cart where userID=?);
DELETE FROM shopping_cart where userID=?;
INSERT INTO payments (cartID,address,city,postalCode,country,idNumber,creditCard,expDate,paymentNumber, cvv)
VALUES (@cartID,?,?,?,?,?,?,?,?,?,?)`;
```




נוסף על אלה ישנם טפסים נוספים. הראשון בניהם הוא טופס יצירת קשר. בטופס זה הנתונים נטענים באופן אוטומטי למשתמש קיים ולאחר מילוי פרטי ההודעה הנתונים נשמרים בבסיס הנתונים (שאלתת הוספה).

Menu Bar Home Our Story Log In Contact us My Cart Our Products Our Events

Contact Us

Ya'akov Cohen St 5, Beersheba, Israel

Glidali@email.com

+972-52-1234-567

Sun - Fri 8:00 AM to 8:00 PM

Roy Benisty roy@asf.com

Your Message

SEND

בנוסף ישנו טופס יצירת אירוע שבו משתמש יכול להזין את הנתונים של האירוע שאותו רוצה לארגן ולהוסיףם לבסיס הנתונים גם כן. גם בטופס זה נתוני המשתמש המחובר נטענים באופן אוטומטי בטופס. לאחר מילוי הנתונים, הנתונים נשמרים בבסיס הנתונים (שאלתת הוספה):

Menu Bar Home Our Story Log In Contact us My Cart Our Products Our Events

Our Events

Business events- Company events, conferences, Happy hours, product launch events, customer events and more.

Private events- Weddings, bachelor parties, birthdays, bar/bat mitzvahs, anniversaries, family reunions or any other event.

For more details - 0544556677

Roy Benisty roy@asf.com

Event type

Event Location

Number Of Participants

dd/mm/yyyy

More details

Send



טעינת הנתונים באופן מהיר התאפשרה לנו אודות לעבודה עם LocalStorage בחלק הקודם של העבודה. החלטנו, בהתייעצות עם נעמה, להמשיך לעבוד גם בחלק הזה עם sessionStorage, עובדה שהיוותה אתגר בחיבור לבסיס הנתונים, אך אפשרה לנו גמישות גבוהה הרבה יותר בהמשך בכל שלב טעינת נתונים של משתמש ספציפי: אנו טוענים את עגלת הקניות והפרטים של המשתמש המחובר מבסיס הנתונים ואז העבודה עם הנתונים שלו בטפסים האחרים נעשית באמצעות ה-sessionStorage ולאחר מכן ברגע שאנו רוצים לשמור חזרה נתונים או לעדכןם אנו שומרים חזרה בבסיס הנתונים. עבודה זו נעשתה בתיאום עם נעמה ואפשרה לנו גמישות מאוד גבוהה וטעינה מהירה של הנתונים.

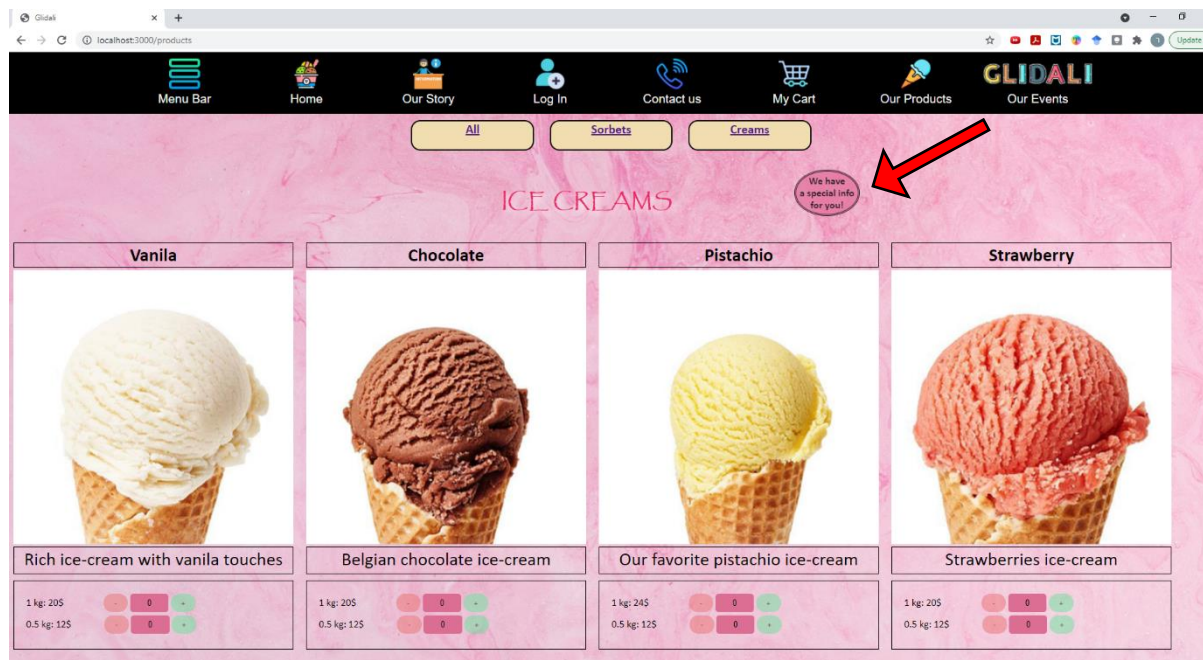
עיבוד מידע

בחלק זה ישנו שימוש במיקום המשתמש והחזרת מידע על מרחק מיקום המשתמש מהעסק, ומתוך כך הערכת הזמן עד להגעת המשלוח.

הפונקציה מתייחסת למספר מקרים: הלקוח קרוב מאוד, הלקוח קרוב, הלקוח רחוק. נציין כי המרחקים המוגדרים לכל קטגוריה נבחרו לאחר התייעצות עם בעל מסעדה המספקת שירות משלוחים, על מנת שזה יהיה מציאותי ויעיל ככל הניתן. עבור כל אחד מהמקרים האלו מתבצע חישוב ומופיעה הודעת חייו מתאימה ללקוח.

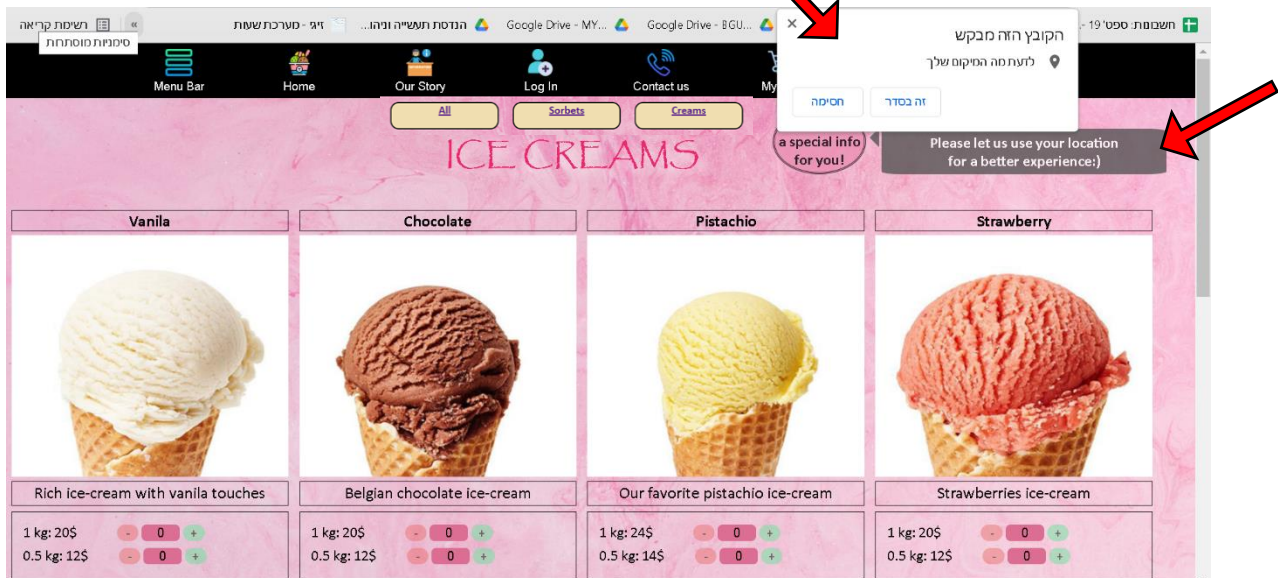
כמו כן, בכפתור, שמומש בhtml ובCSS, תוך שימוש כמובן בפונקציה שכתבנו בJS, ישנם שימושים באנימציה ובפונקציות שונות על מנת שיהיה מזמין, נעים לעין וישתלב עם העיצוב והסגנון של האתר. בנוסף, שמנו דגש על שפה "UXית" שמעצימה את חווית המשתמש ומעודדת רכישה.

להלן תיאור הפונקציה (מצד המשתמש. את הקוד ניתן למצוא בקובץ JS שבבנינו infoProcessingJS).
1. בעת כניסה לעמוד התפריט, מצד ימין לכותרת מופיע הכפתור בהבהוב צבעים המזמין לחיצה:

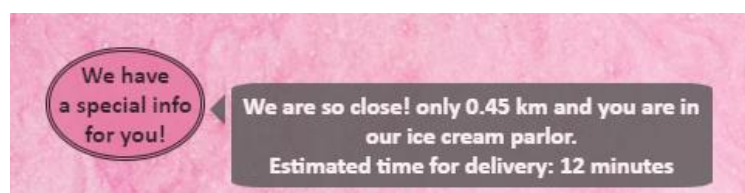
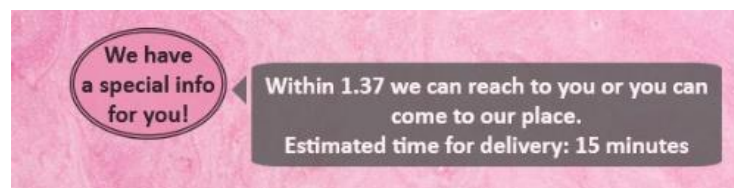




לאחר לחיצה על הכפתור נפתח חלון "פופ אפ" עם ההודעה הבאה (באפור), וכן אישור על שימוש במיקום:



לאחר לחיצה על אישור השימוש במיקום, מתבצע חישוב ותוכן ההודעה מוחלף בהתאם למיקום המשתמש. להלן דוגמאות של מיקומים שונים (נוסח במספר מחשבים):





תוכן דינמי

תכנות דינמי:

בתכנות הדינמי עבדנו על קטלוג מוצרים our products, סוגי הגלידות. עבדנו על התבנית הסטטית והעברנו אותה לדינמית.

השתמשנו בREQUEST PARAMS. כאשר מוסיפים את שם הקטגוריה למעלה בURL מתבצעת שאילתה לפי הקטגוריה שבחרנו. אם יש קטגוריה אז מפלטים לפיה, אחרת - מציגים את כל סוגי הגלידות.

`http://localhost:3000/products` (all ice creams)
`http://localhost:3000/products/sorbet` (only sorbets)
`http://localhost:3000/products/cream` (only creams)



נספח 1 – הפעלת הפרוייקט

1. יש להוריד את התיקייה Final מגיטהב.
2. יש להכנס לmysql וליצור Schema בשם web-project-g4.
3. לאחר הקמת הסכמה יש ליצור query חדש להעתיק אליו את תוכן קובץ הsql 'web-project-g4.sql' ולהריץ. לאחר ההרצה יש לוודא שהוקמו הטבלאות עם הנתונים.
4. לאחר מכן יש להתקין npm בתיקייה באמצעות cmd (npm i).
5. יש לשנות את הפרטים בקובץ db.config.js בהתאם.
6. יש להריץ את ה server באמצעות פקודת node server.js.
בריצת הקובץ תתקבל הודעה כזאת:

```
C:\Users\roych\Desktop\Studies\Summer\Final>node server.js
Server is running on port 3000.
Successfully connected to the database.
```