**המחלקה להנדסת תוכנה**

**פרויקט גמר – תש"פ**

**אפליקציה לניהול משק בית**

**House-Keeping-App**

**מאת : גיא עודי 203169180**

**שרי זרביב 312755614**

GitHub **:** <https://github.com/guuy1/house-keeping>

יומן : <https://trello.com/b/9VY5rvWg/house-keeping-app>

סרטון גרסת אלפא: <https://drive.google.com/file/d/1So2gOLwr8crRwkBpnZ0ji1I0bMAe-ovy/view>

**מנחה אקדמי: גברת באש תמר אישור תאריך:**

**רכז הפרויקטים: ד"ר שפיינר אסף אישור: תאריך:**

תתארו לכם מציאות שבה אף מוצר במקרר לא נזרק בגלל שפג תוקפו, מציאות בה מוצר לא נשכח במעמקי המקרר ומחכה שמישהו ישים לב אליו לפני שגם הוא ייזרק, מציאות בה נדע בכל רגע מה חסר בבית באמצעות המשפחה או באמצעות מערכת הלומדת את צרכיי המשתמש, קשה ?

מסמך זה מפרט בפניכם את הפתרון שלנו לבעיות אלה ,

פתרון אשר יהיה נגיש לכל אדם בעל מכשיר חכם .

# תוכן עניינים

[תוכן עניינים 3](#_Toc30850369)

[מבוא 4](#_Toc30850370)

[תיאור הבעיה 5](#_Toc30850371)

[דרישות ואפיון הבעיה 5](#_Toc30850372)

[הבעיה מבחינת הנדסת תוכנה 9](#_Toc30850373)

[תיאור הפתרון 10](#_Toc30850374)

[Use Case 12](#_Toc30850375)

[טבלת סיכונים 13](#_Toc30850376)

[טבלת דרישות 14](#_Toc30850377)

[דרישות נוספות ולא פונקציונאליות 14](#_Toc30850378)

[טכנולוגיות בשימוש 15](#_Toc30850379)

[תכנית בדיקות 16](#_Toc30850380)

[סקירת עבודות/בספרות והשוואה/סקר שוק 17](#_Toc30850381)

[סיכום ומסקנות 18](#_Toc30850382)

# מבוא

בשנים האחרונות אנו שמים לב לתופעה שהתחילה בעקבות תרבות המערב, תופעה המתפתחת בקצב מתמיד, והיא תופעת השפע . בכל מקום , בכל שעה , מוצרים שאנו צורכים בין אם אנו זקוקים להם ובין אם לא - זמינים לרכישה כמעט מידית .

דבר זה יצר תרבות של קנייה מאסיבית , ולא בהכרח קנייה של מוצרים אשר הכרחיים למשק הבית .

משפחות רבות מוצאות את עצמן שבוע אחר שבוע במצב של בזבוז וזריקת מוצרים, אשר לא ניתן להשתמש בהם מכיוון שאינם בעלי תוקף מתאים.

הפרויקט שלנו הוא פרויקט יזמי , אשר מתרכז בשיפור ניהול משק הבית בדגש על ניהול המקרר ,ובשיפור התקשורת המשפחתית הנוגעת לניהול משק הבית כמו רשימת קניות משותפת. בנוסף נתמקד בלמידת פעולת המשתמש וכך יוכלו להינתן המלצות אשר ישפרו את חוויית השימוש באפליקציה.

הפרויקט מתחלק לשני חלקים עיקריים :

1. קריאת ברקוד של מוצרים ומעקב אחרי תוקפם , בניית רשימת קניות משותפת.
2. למידה של פעולות המשתמש על מנת לשפר את חווית המשתמשים .

# תיאור הבעיה

כפי שהזכרנו במבוא אחת הבעיות העיקריות שנתקלנו בהן שהביאו אותנו לחשוב על האפליקציה היא בעיית הבזבוז בכל בית .

משפחות רבות שבוע אחר שבוע מוצאות את עצמן זורקות עשרות מוצרים שלא הספיקו להשתמש בהם לפני התאריך המומלץ לשימוש אחרון ,הדבר יוצר תסכול מבחינה כלכלית ומבחינה אנושית על זריקת האוכל שלא נעשה בו שימוש.  
בעיה נוספת היא שבשל קניות מרובות לעיתים תכופות קורה שמוצר נשכח במעמקי המקרר, ולכן עד שהמוצר מתגלה עובר תוקפו והוא לא ניתן לשימוש יותר – וזה מחזיר אותנו לבעיית הבזבוז המיותר.   
חוסר תקשורת בין בני הבית עלול ליצור מצב של אי סנכרון בקניות ורכישה כפולה של מוצרים, או להיפך מוצר חסר לא נרכש מכיוון שאין תקשורת בין בני הבית ולא ידוע שהמוצר הנ"ל חסר מראש .

בעיה נוספת היא שכיום לכל אדם קיימות אינספור התחייבויות במהלך היום ,דבר שעלול לגזול זמן הוא הכנת רשימת קניות כל שבוע מחדש ובימנו כאשר המנטרה המובילה היא זמן שווה כסף , אנשים נמצאים בחיפוש מתמיד לפתרונות אשר חוסכים זמן .

## דרישות ואפיון הבעיה

כיום ישנה התקדמות במערכות לניהול נוח וקל של משק הבית , למשל מקררים חכמים אשר סורקים את תכולת המקרר ובהתאם לזאת נותנים פתרון למשתמש . הבעיה העיקרית שטכנולוגיות אלו נמצאות בשימוש בעיקר אצל בעלי יכולת כלכלית .

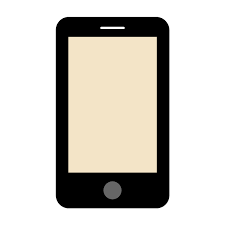
מטרת הפרויקט היא להנגיש טכנולוגיית ניהול משק הבית לכלל האוכלוסייה , כלומר כל אדם שברשותו מכשיר חכם יוכל להשתמש באפליקציה זו.

**הדרישות העיקריות שנפתור עבור המשתמש הן :**

* הרשמה מהירה לאפליקציה ואפשרות להוסיף משתמשים לחשבון המשפחתי.
* אפליקציה נוחה וברורה לשימוש .
* אפשרות לעדכן מוצרים שבמקרר ע"י קריאת ברקוד או חיפוש ברשימה ועדכון תוקפם.
* זיהוי הברקוד במהירות (על מנת לא להטריח את המשתמש) , והתאמה מהירה בין הברקוד למאגר המוצרים הקיים .
* אפשרות ליצירת רשימת קניות אשר מספר משתמשים יכולים להתחבר אליה בו-זמנית , וכל שינוי ברשימה יתעדכן מידית לכל משתמשי הבית(בחלק זה אין צורך בהתראה למשתמשים ) .
* אפשרות לעדכן את רשימת הקניות בכתב יד על מסך מגע או בהקלדה .
* קבלת המלצות מהמערכת לרכישות עתידיות בהתאם לפעולות עבר של המשתמש .

## המסכים העיקריים באפליקציה

מסך כניסה / הרשמה -

****

Forget password

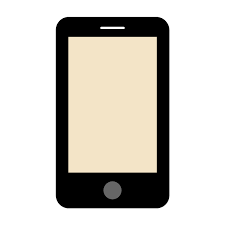
enter

Sign in

password

Welcome



****מסך ראשי -

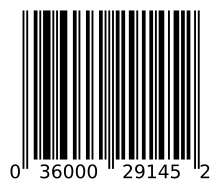
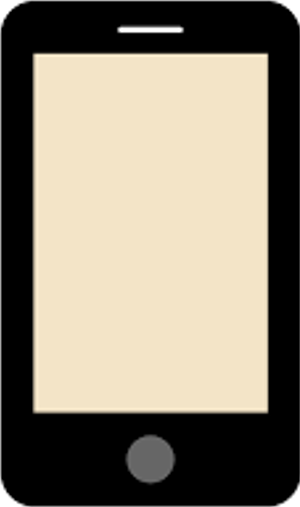
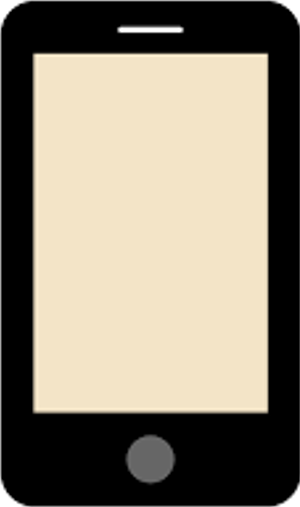
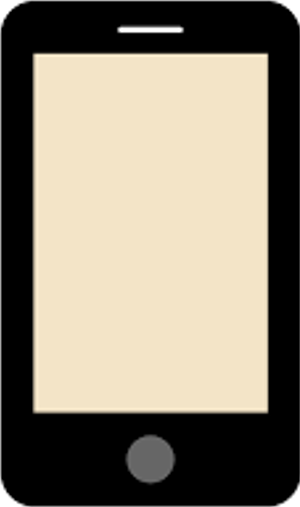
**Recommendations System**

**You should add milk to your list , its look like you forget .**

**Read Barcode**

**My Products**

**Shopping List**

****

All lists

* chesse
* chicken
* Milk 22.02.2020(expiration date)
* Koteg 15.02.2020
* Egges 23.03.2020
* Hummos 10.02.2020

**My Products**

Barcode

New list

+ Add product

* Milk
* Eggs
* Cola

Shopping List

21/01/2020

## הבעיה מבחינת הנדסת תוכנה

* שימוש ולמידת פלטפורמת React בפעם הראשונה .
* אחסון נתונים ב firebase בזמן אמת – משתמשים חדשים, מוצרים ועוד ..
* הוספת משתמשים לחשבון הראשי בעלי הרשאות גישה .
* סריקת ברקוד בזמן אמת דרך המצלמה של המכשיר החכם.
* התאמה מהירה בין הברקוד למאגר המוצרים והצגתו .
* אפשרות להוספת מוצר חדש בכתב יד על מסך מגע או הקלדה .
* עדכון בזמן אמת לכל משתמשי הבית של רשימת הקניות .
* עדכון תמידי של מאגר המוצרים .
* למידת פעולות המשתמשים – סטטיסטיקות ,רכישות דומות , התאמה בין משתמשים דומים על מנת לתת המלצות לרכישות עתידיות המתאימות למשתמש .
* למידת רשימת קניות של הבית ובדיקת מוצרים שחוזרים על עצמם על מנת לעזור למשתמש להכין רשימה מדי שבוע .
* ניהול המוצרים שנסרקו ושליחת התראות למשתמשי הבית על מוצרים שעומדים להתקלקל.

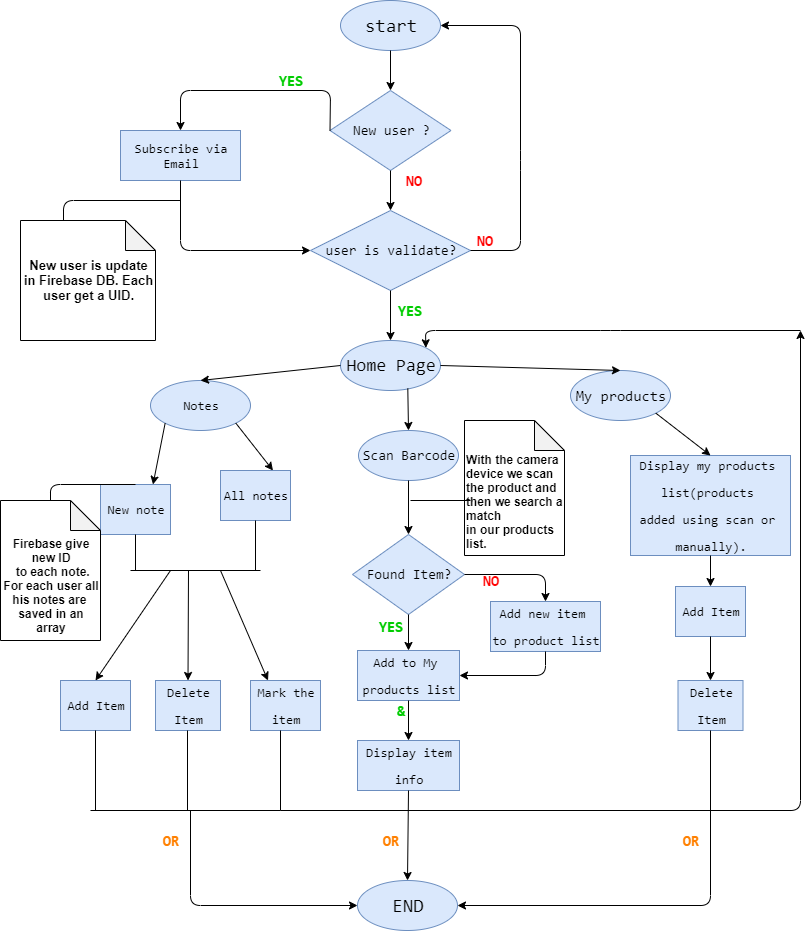
# תיאור הפתרון

הפתרון אשר אנו מציעים הוא יצירת אפליקציה באמצעות React - ספריית קוד פתוח של JS המאפשרת בניית אפליקציות ברמה גבוהה . על מנת לאחסן את כל הנתונים נשתמש בשירות של Firebase , אנו נשתמש ב DB מסוג REAL TIME , כלומר בזמן אמת המידע מתעדכן אצל כל המשתמשים .

בשלב הראשון נאפשר למשתמש להירשם לאפליקציה באמצעות אימייל וסיסמא .

משתמש זה יהיה ה owner והוא יוכל להוסיף משתמשים נוספים לחשבון המשפחתי באמצעות X .

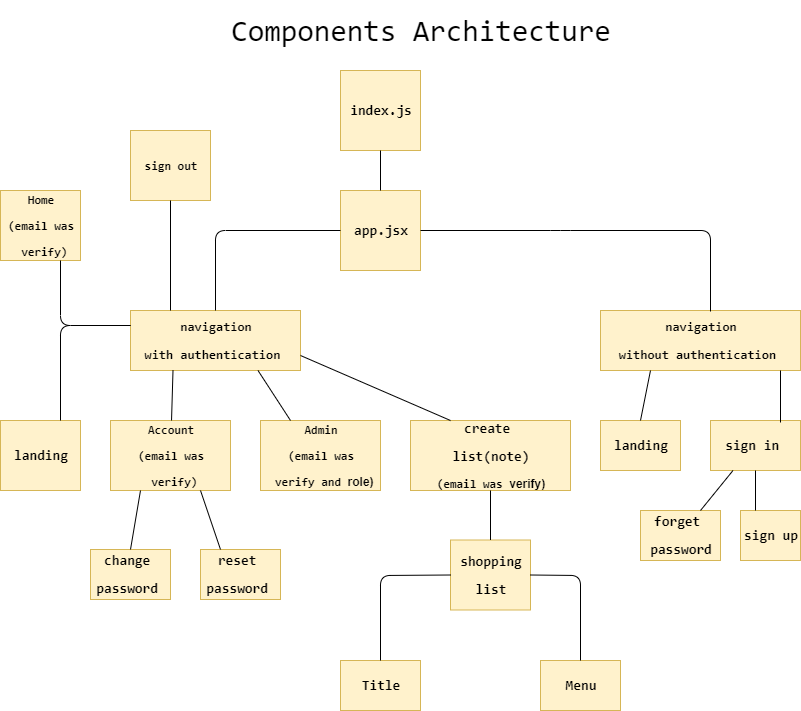
לאחר ההרשמה Firebase דואגת לנו לאימות של המשתמש ורק לאחר אישור הפרטים המשתמש מקבל גישה לדף הבית של האפליקציה .

בפני הלקוח עומדות כעת כמה אפשרויות , יצירת רשימה(למשל רשימת קניות) , סריקת ברקוד או בדיקת המוצרים שלו.

להלן תרשים זרימה המציג את אפשרויות

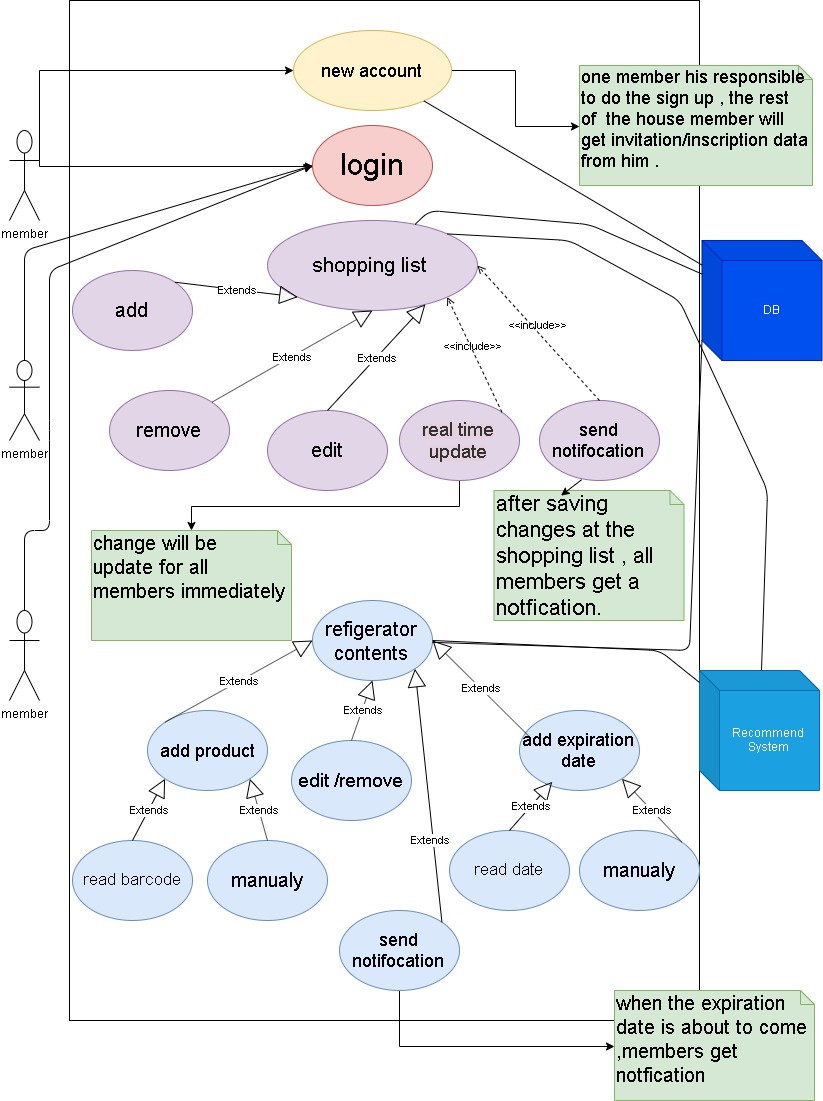
המשתמש באפליקציה :

כעת נציג את ארכיטקטורת ה components \* :



\*תרשים זה מתאר את הקומפוננטים הקיימים בשלב זה.

## **Use Case**



## **טבלת סיכונים**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | | | **PRIORITIZE** | | | **PLAN TO REDUCE IMPACT** |
| **Sort by priority** | **Description of risk** | **Owner** | **Possible Impact** | **Prob (%)** | **Impact (L, M, H, VH)** | **Risk Code (color per table above)** | **Mitigation Plan / Contingency Plan** |
| 1 | אי עמידה בזמני ההגשות של פרויקט הגמר | גיא  ושרי | אי הגשה של המשימות,  הורדה בציון. | 15% | VH |  | בניית לוח זמנים עבור משימות הפרויקט ומעקב אחר סיום המשימות בזמן. העשרת הידע האישי ע"י חקירה של שלל הטכנולוגיות הקיימות. |
| 2 | אי מוכנות למכשולים טכנולוגיים אשר לא ניתן לזהות מראש. | גיא  ושרי | עיכוב בהתקדמות הפרויקט  וייאוש בצוות. | 15% | H |  | חקירה ברשת והתייעצות עם אנשי מקצוע רלוונטיים . |
| 3 | כישלון ביצירת אפליקציה שימושית ונוחה למשתמש . | גיא  ושרי | אי שימוש באפליקציה מצד המשתמש. | 20% | VH |  | טרם תחילת העבודה הטכנית על הפרויקט, נבדוק חוות דעת של משתמשים פוטנציאלים.  במהלך הפרויקט נשתמש בנסיינים ונעדכן את האפליקציה בהתאם להערותיהם. |
| 4 | מחלה או היעדרות של אחד מחברי הפרויקט | גיא  ושרי | עיכוב בהתקדמות הפרויקט והפלת עומס על אחד מחברי הצוות. | 60% | H |  | עבודה משותפת מראש כדי שכולם יהיו מחוברים ולא ייווצר פער בין חברי הצוות . עמידה בזמנים שנקבעו בשביל לכפר על זמן היעדרות. |
| 5 | תקופת מבחנים + פרויקטים אחרים | גיא  ושרי | פחות זמן עבודה על הפרויקט והתמקדות במבחנים/פרויקטים | 60% | H |  | חלוקת משימות ועבודה יותר חזקה בתקופה שלפני. עמידה בלוח הזמנים שנקבע . |
| 6 | לקראת סיום זמן העבודה על הפרויקט מתגלות תקלות רבות | גיא  ושרי | כישלון של הפרויקט | 15% | VH |  | במהלך כל הפרויקט יבצעו בדיקות קטנות כך שבמקרה של באג כלשהו נוכל לחזור לשלב האחרון ולבדוק היכן טעינו . |
| 7 | הטכנולוגיה אשר בשימוש משתנה | גיא  ושרי | עיכוב בהתקדמות הפרויקט  וייאוש בצוות. | 20% | H |  | במהלך כל הפרויקט להישאר מעודכנים בשינוים העלולים להגיע, ולהכין תוכנית לשילוב השינויים . בנוסף יש להשתדל להישאר בלו"ז ובכך להקטין את הסיכוי לשינויים גדולים . |
| 8 | תכנון זמנים לא נכון | גיא  ושרי | עיכוב בהתקדמות הפרויקט | 40% | VH |  | יש בלבצע עבודת מחקר ותכנון ארכיטקטורת הפרויקט ברמה גבוהה. דבר זה יצמצם את בעיות העלולות להופיע בהמשך. |

## טבלת דרישות

|  |  |
| --- | --- |
|  | **תיאור הדרישה** |
| **1** | תמיכה בכתב יד על המסך |
| **2** | ממשק ברור וידידותי למשתמש |
| **3** | סריקה מהירה של הברקוד |
| **4** | עדכון רשימת קניות משותפת המתעדכנת בזמן אמת |
| **5** | חיבור/הסרת משתמשים נוספים לחשבון הראשי של הבית |
| **6** | קבלת התראות והמלצות מהמערכת על מנת לשפר את חוויות המשתמש |

## דרישות נוספות ולא פונקציונאליות

* מהירות (זמן):
* זמן מעבר בין מסכים לא יעלה על 0.5 שניות
* זמן כניסה למערכת למשתמש רשום לא יעלה על 1 שניות
* זמן הרשמה למערכת למשתמש חדש לא יעלה על 3 שניות
* זמן הוספת/הורדת/עריכת פריט ברשימה לא יעלה על 0.2 שניות
* עומס:
* האפליקציה תאפשר למספר משתמשים להתחבר לאותו חשבון

בו זמנית, מבלי שדבר זה ישפיע על איכות הביצועים.

* זמינות:
* האפליקציה זמינה למשתמש בכל רגע ביום, ומתעדכנת בזמן אמת.
* קיבול (צריכת זיכרון / CPU):
* האפליקציה לא תדרוש מהמשתמש הרבה זיכרון מכיוון שרוב המידע נשמר בענן.
* ניידות:
* האפליקציה תפעל מכל מקום ומכל מדינה .
* תחזוקה:
* האפליקציה לא תדרוש תחזוקה רבה מצד המתכנתים, מלבד בדיקות שגרתיות שהכול מתנהל כשורה. עדכון מאגר המוצרים הקיימים בסופר ייעשה באמצעות תוכנה מדי פעם.
* אבטחה:
* כל המידע הקשור למשתמש יהיה חסוי ולא ניתן יהיה לגשת אליו ללא הרשאה.

## טכנולוגיות בשימוש

על מנת לבנות אפליקציה זאת אנו משתמשים בטכנולוגיות הבאות –

* React - ספריית קוד פתוח בשפה JS המשמשת לפיתוח אפליקציית web.

לאחר חקירה של מספר פלטפורמות ,נראה כי React הכי מתאים למטרותינו, מכיוון שהוא מהיר ומאפשר הבנה יותר קלה של המערכת ושל הקוד שכתבנו.

* Firebase - מסד נתונים מסוג NoSQL מסוג Key/Value , מאפשר שמירת נתונים ועדכונם בזמן אמת .
* Visual Studio Code Editor – לכתיבת הקוד .
* לקריאת הברקוד נשתמש בטכנולוגיית X ( עפ"י חקירה שנעשתה נראה כי התוכנה הכי מתאימה לקריאת ברקודים בזמן אמת היא QuaggaJS, אך טרם התנסינו בה).

# תכנית בדיקות

* משתמש שאינו רשום יכול להיכנס לשני דפים מלבד landing וכניסה למערבת.
* הרשמה תתאפשר רק עם סיסמא של לפחות 6 תווים.
* ההרשמה תתאפשר רק עם כתובת אימייל חוקית(אימות הכתובת נעשה ע"י Firebase באמצעות אימייל שנשלח לכתובת שהמשתמש הכניס).
* לאחר אימות משתמש בדיקה שפרטיו נשמרו ב Firebase,כל משתמש מקבל UID .
* משתמש רשום יכול לגשת לדפים נוספים שהם Home Page ,רשימת קניות ,דף משתמש שם יש אפשרות לשנות סיסמא , ודף יציאה מהמערכת.
* כניסה לדף הרשימות יופיעו רשימות ישנות של המשתמש ואפשרות ליצור רשימה חדשה.
* כל רשימה חדשה שנפתחת מקבלת UID ב – Firebase, כל הפריטים של הרשימה נשמרים ב-DB תחת הרשימה.
* בשלב זה בעת כניסה לרשימת קניות הכותרת שתופיע כברירת מחדל היא "רשימת קניות".
* ניתן לשנות כותרת באמצעות לחיצה על ה Input .
* הוספת פריט לרשימה תתאפשר ע"י לחיצת על מקש enter או ע"י לחיצה על אייקון + .
* מחיקת פריט תתאפשר ע"י לחיצת אייקון X המופיע מימין לפריט ברשימה .
* כל עדכון ברשימה יתעדכן ב Firebase בהתאם – עדכון פריט ,מחיקת פריט, הוספת פריט, הוספת רשימה, מחיקת רשימה ושינוי כותרת רשימה.
* בעת כניסה לדף משתמש הכותרת בשלב זה תהיה כתובת האימייל של המשתמש .
* איפוס סיסמא יתאפשר רק עם הכנסת כתובת האימייל של המשתמש בצורה תקנית ולאחר הכנסת סיסמא חדשה ואימותה דרך שירות ה-Firebase.

# סקירת עבודות/בספרות והשוואה/סקר שוק

|  |  |  |
| --- | --- | --- |
| שם יצרן | תיאור יכולות | עלות |
| אוניברסיטת "לנקשייר" | המקרר סורק עצמאית את מגירות המקרר ומזמין מוצרים חסרים,נותן המצלות על מתכונים מתאימים בהתאם לתכולת המקרר , מערכת שינוע מוצרים בעלי תוקף ישן לקדמת המקרר . | מחיר אינו מצויין במפורש, אך מודגש כי העלות יקרה . |
| Siemens fridge with home connect | מצלמות פנימיות המאפשרות לבחון בכל עת את תכולת המקרר , אפשרות שליטה על הגדרות המקרר מרחוק. | 14,000 ש"ח |
| Samsung Family-Hub series | בעקבות הכנסת מתכון המקרר מזהה איזה מוצרים צריך עבור המתכון, עדכון רשימת קניות משפחתית באמצעות עוזרת קולית וגישה משותפת לרשימת הקניות . | 11,400 ש"ח |
| Google Keep | ביצירת רשימות משותפות למספר משתמשים , כאשר כל עדכון של רשימה נעשה אצל כל המשתמשים בזמן אמת . | ללא עלות |

ניתן לראות מסקר השוק כי טכנולוגיות מאוד מתקדמות קיימות במקררים מודרניים על מנת לשפר את ניהול המקרר, אך המחיר עבור מקרר חכם גבוה ביחס למקררים אחרים ולכן טכנולוגיה זו עדיין אינה מונגשת לכלל האוכלוסייה . רשימת הקניות המשותפת קיימת אך אנו נרצה להנגיש את כל האופציות לשיפור ניהול משק הבית במקום אחד.

# סיכום ומסקנות

בשלב זה האפליקציה מאפשרת למשתמש את הפעולות הבאות –

* הרשמה באמצעות אימייל וסיסמא.
* כניסה למערכת שינוי/איפוס סיסמא(במקרה שנשכחה).
* כניסה לחלון רשימות בו יוצגו כל הרשימות שנפתחו ע"י המשתמש, ניתן לעבור בין הרשימות הקיימות.
* בכל רשימה יש אפשרות להוסיף פריט ,לסמן פריט או למחוק אותו .
* ניתן למחוק רשימה שנבחרה ע"י המשתמש.

הפרויקט עד עתה העלה בפנינו כמה מסקנות –

* React מאפשר לנו כתיבת קוד יותר בטוחה, מכיוון ש Data מועבר מאב לבן, שינוי של קומפוננט ייעשה רק אם באמת נרצה לשנות אותו. כתיבה בצורת ירושה נותנת לנו יותר הבנה ושליטה במה שהקוד שלנו עושה .
* React משתמש ב Virtual Dom, אשר מאפשר לנו לערוך html בתוך ה JS עצמו, כלומר אין יותר הפרדה בין html ו-css , דבר זה נותן לנו מהירות .
* השפה JSX היא הנוחה ביותר כדי לכתוב html בתוך Javescript .
* Firebase לא מבטיח לנו עדכון מידי בזמן אמת ולכן יש להתחשב בזה בעת כתיבת הקוד.