# המחלקה להנדסת תוכנה

# פרויקט גמר- תשע"ח

# פיתוח אפליקציית אנדרואיד לחיפוש מסעדות על פי מנות וממשק ניהול לבעלי המסעדות

#### חיבור זה מהווה חלק מהדרישות לקבלת תואר ראשון בהנדסה

**מאת**

**דניאל פוטשניקוב ושני ג'ופי.**

**יוני 2018**

**המחלקה להנדסת תוכנה**

# פרויקט גמר- תשע"ח

# פיתוח אפליקציית אנדרואיד לחיפוש מסעדות על פי מנות וממשק ניהול לבעלי המסעדות

#### חיבור זה מהווה חלק מהדרישות לקבלת תואר ראשון בהנדסה

**מאת**

**דניאל פוטשניקוב ושני ג'ופי**

**אישור:**

**אישור:**

**מנחה אקדמי: מר רועי יצחק**

**רכז הפרויקטים: ד"ר שפיינר אסף**

##### תקציר

##### הצהרה

הפרויקט נעשה בהנחיית מר רועי יצחק

במחלקה להנדסת תוכנה,

עזריאלי - המכללה האקדמית להנדסה ירושלים

החיבור מציג את עבודתינו האישית ומהווה חלק מהדרישות לקבלת

תואר ראשון בהנדסה.

##### תודות

ברצונינו להביע תודות לכל האנשים שעזרו לנו במהלך ביצוע פרויקט

הגמר, ובמיוחד -

למנחה הפרויקט, מר רועי יצחק , על עזרתו המקצועית, הליווי

וההדרכה ויחסו האישי לאורך כל הדרך.

לרכז הפרויקטים, מר אסף שפיינר , על המסירות, ההקשבה והנכונות

לסייע בכל בעיה ושאלה.

למשפחה שתמכה לאורך כל הדרך

ולחברים היקרים שהיו שם לשמוע ולתמוך

.

#### 

#### מערכות ניהול הפרויקט ובקרת תצורה

#### 

|  |  |  |
| --- | --- | --- |
| מיקום | מערכת | # |
| **https://github.com/danielpt17/HungryMonkey** | מאגר קוד | 1 |
| **https://paper.dropbox.com/doc/Project-overview-Hungry-Monkey-vvJDrBjUxaNgpJj3rnYDy** | יומן | 2 |
|  | ניהול פרויקט | 3 |
|  | סרטון | 4 |

## מילון מונחים, סימנים וקיצורים

* **Hungry Monkey - שם האפליקציה.**
* **ElasticSearch - מסד נתונים שבו נשתמש כדי לאחסן את פרטי המנות**

**והמסעדות.**

* **guest - שימוש באפליקציה במצב אורח.**
* **exist user - שימוש באפליקציה במצב משתמש קיים.**
* **Redis**  **- מסד נתונים עבור המשתמשים, אשר שימש אותנו**

**למימוש מערכת ההמלצות.**

* **Index - שם המסד נתונים ב ElasticSearch.**
* **Type - סוג האובייקט ב ElasticSearch**
* **Aws - ספקית האחסון אמאזון.**
* **UI/UX - חווית המשתמש הן מבחינת העיצוב והן מבחינת**

**השימושיות.**

* פ'יצר - פונקציונליות באפליקצייה.

#### 1. תיאור מסגרת הפרויקט

לחפור שזה פרוייקט יזמות

#### 2. תיאור הבעיה

# 2.1. הבעיה הכללית

כיום, אחד הבילויים המרכזיים בחיי היום יום שלנו, ומה שמעסיק אותנו במהלך היום הוא: "מה נאכל היום? ואיפה?".

בשוק קיימים מגוון פתרונות לפתירת הבעיה, אשר מציעים לנו שירותי מציאת מסעדות לפי :אווירה, סוג מטבח ומיקום. כמו כן, אם אדם רוצה להזמין אוכל הביתה? הפתרון גם כן קיים.

אך, מה יקרה במצב בו אדם יודע **בדיוק** מה הוא ירצה לאכול/לשתות הערב, וירצה לדעת איזה מסעדות ישנן באזור אשר עונות לצרכיו, ולפי זה לבחור את מקום הבילוי? או לסירוגין הכרת מקומות חדשים אשר גם שם הוא יוכל להנות ממנה דומה.

# 2.2. הבעיה מבחינה הנדסית

* עבודה בצוות- סנכרון ותיאום .
* תכנון המסד נתונים בצורה יעילה ביותר, כך שיוכל להציג תוצאות עבור חיפוש בסיסי, ומתקדם (מנה ספציפית או סוג מאכל).
* התממשקות לשירותים של aws:

1. אחסון צד השרת בec2 וelastic beanstalk.
2. אחסון תמונות בs3.

* מקסום תוצאות חיפוש מתאימות. כלומר: הצגת מנות זהות עם שמות שונים. לדוגמה: שקשוקה ירוקה וצרפתית, הן אותן מנות עם שמות שונים.
* הזדהות ללוקחות שמורים מול צד שרת(sessions).
* עבודה בטכנולגיות/שפות חדשות. במקרה שלנו: שימוש בשני מסדי נתונים מסוג Elasticsearch וRedis. מימוש צד השרת בNodeJS וצד הלקוח בAndroid.
* מימוש מערכת ההמלצות, תוך שימוש ב REDISכדי לממשה ,אשר תלמד את תוצאות החיפוש של המשתמש הקיים ותציג לו הצעות אשר יכולות להתאים לו.
* מימוש צד הלקוח( אפלייקצית המשתמשים, וממשק הניהול) עם UI/UX אולטימטיבי.

#### 3. תיאור הפתרון

# 3.1. סביבת העבודה וכלים

# מסדי נתונים:

# Elasticsearch- זהו מסד נתונים מבוסס json, נטול טבלאות, אשר מספק יכולות multitenant מבוזרת לחיפוש מעל טקסט מלא ביעילות שיא.

# Redis- פלטפורמה לשמירת נתונים בזיכרון, אשר יכולה לשמש כמסד נתונים. מבני הנתונים של Redis הם כמו חלקי "לגו" אשר מפשטים מימוש של פונקציונליות מורכבת(כגון מערכת המלצות), הם יעילים במיוחד מאחר וכל הגישה ופעולות על הנתונים נעשות בזיכרון ממש ליד איפה שהנתונים נשמרים, מה שמקנה לנו שימור מאגרים (מאבד ורשת).

# שירותי איחסון: השתמשנו בשירותי האחסון של AWS:

# Ec2- שירותי אחסון מאובטחים בענן.

# Elastic Beanstalk- שירותי פריסה(deploy) עבור אפליקציה וובית, אשר נמצאת בEc2.

# S3- שירות אחסון עבור קבצים.

# Elasticsearch service- שירותי תמיכה ואיחסון במסד הנתונים עם plugin של Kibana(יפורט בהמשך).

# 

# ממשק ניהול למסדי הנתונים:

# Kibana- מערכת ויזואליזציה ואנליזה המנתחת נתונים ומספקת יכולות בינה עסקית. השתמשנו עבור Elasticsearch.

# Putty- להשילים...

# 

# שפות תכנות+ סביבת עבודה:

# צד לקוח (אפלייקצית מנות)- נכתבה באנדרואיד בandroid studio.

# צד לקוח(ממשק ניהול)- נכתב בJavaScript בWebStorm.

# צד שרת- נכתב בNodeJS בWebStorm.

# 

# 3.2. הפתרון

הפתרון יהיה אפליקציית אנדרואיד, שתיתן למשתמש את האופציה לחפש מסעדות, אשר מגישות את המנה הרצויה, ע"פ שמה או סוגה, או ע"פ פרמטרים נוספים כגון: טווח מחירים ,מרחק, סוג מסעדה. האפליקציה תציג לו את המסעדות העונות על פרמטרים אלו , וכמו כן תדע לנווט אותו למסעדה בה ירצה לאכול.

תהיה למשתמש אפשרות למיין את התוצאות שהתקבלו לפי : מרחק/מחיר, וסינון מנות לפי כשרות .

כמו כן, תהיה מערכת המלצות למשתמש הקיים אשר תמליץ לו על מנות דומות שהוא עשוי לאהוב.

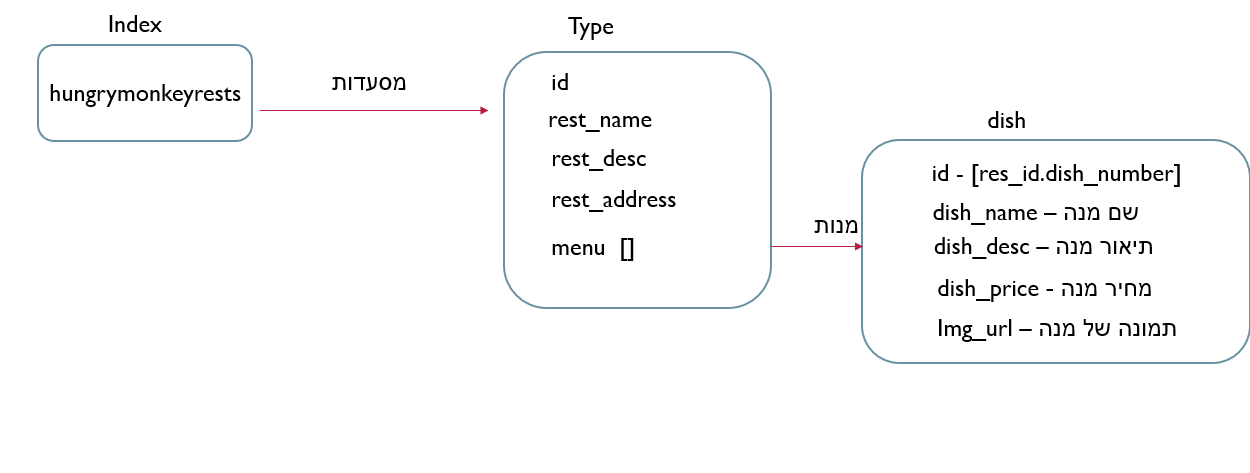
עבור הזנת פרטי המסעדות והתפריטים יהיה ממשק ניהול וובי לבעלי המסעדות, בו הם יוכלו לנהל את דף המסעדה שלהם. בזכות ממשק הניהול ניצור אינטרס פרסום עבורם, וכך בעצם האפליקציה תשרת את שני הצדדים.

# 3.3. ארכטיקטורת המערכת

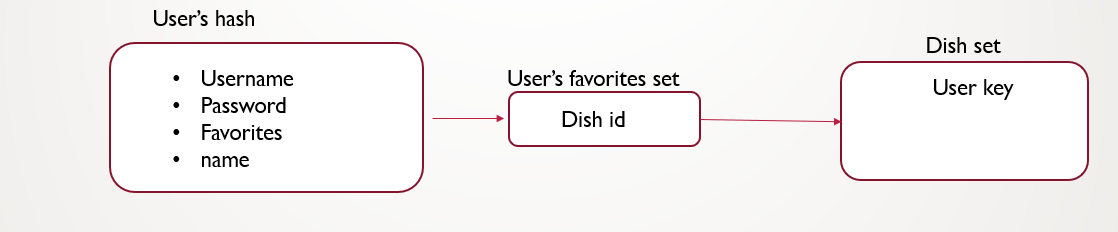
השתמשנו בשני מסדי נתונים, Elasticsearch ו.Redis

* בElasticsearch-, בחרנו להשתמש על מנת לממש חיפושים טקסטואליים מתקדמים ביעילות שיא.

תכננו את המסד נתונים בצורה כזאת: יצרנו אובייקט של מסעדה אשר יכיל את כל פרטי המסעדה בו(שם, כתובת, תיאור, כשרות, מספר מזהה) ואובייקט מקונן של תפריט אשר יורכב מאובייקטים של מנות המסעדה. כל אובייקט של מנה יכיל את פרטי המנה (שם, תיאור, תמונה, מחיר ומספר מזהה אשר יורכב ממספר המזהה של המסעדה כמספר השלם, ומספר העשרוני ייצג את המנה בתפריט של אותה מסעדה).



## **בRedis-, בחרנו להשתמש כדי לממש את מערכת ההמלצות שלנו. לכל משתמש יצרנו טבלת גיבוב עם פרטיו. כמו כן, לכל משתמש יצרנו סט עבור כל המספרים המזהים של המנות שהוא אהב, ועבור כל מנה שלפחות משתמש אחד אהב אותה, יצרנו גם כן סט אשר יכיל את כל המפתחות של כל המשתמשים שאהבו אותה.**



# 3.4. מהי המערכת

**המערכת שלנו מורכבת משני חלקים: אפליקציית אנדרואיד אשר תשמש את האנשים ביום יום לאיתור מסעדות, וממשק ניהול וובי עבור בעלי המסעדות על מנת להזין את מסד הנתונים, ולנהל את פרטי המסעדה שלהם.**

**הפיצ'רים של האפליקציה הינם:**

* הזדהות משתמש קיים מול צד שרת או שימוש במצב "אורח" .
* אופציה לחיפוש בסיסי - כל המסעדות אשר מגישות את המנה עם אותו השם.
* אופציה לחיפוש מתקדם - עם מאפייני חיפוש וסינון כגון : כשרות ,מחיר, מרחק וסוג מסעדה.
* הצגת היסטורית חיפוש מנות, ומקומות מועדפים למשתמש הקיים.
* מיון תוצאות חיפוש לפי מחיר ומרחק.
* מערכת המלצות למשתמש הקיים, אשר תתבסס על מקומות מועדפים של משתמשים אחרים עם העדפות דומות, ותדע להמליץ למשתמש על מנות חדשות שיכולות לעניין אותו.
* הוראות הגעה- קישור לgps.

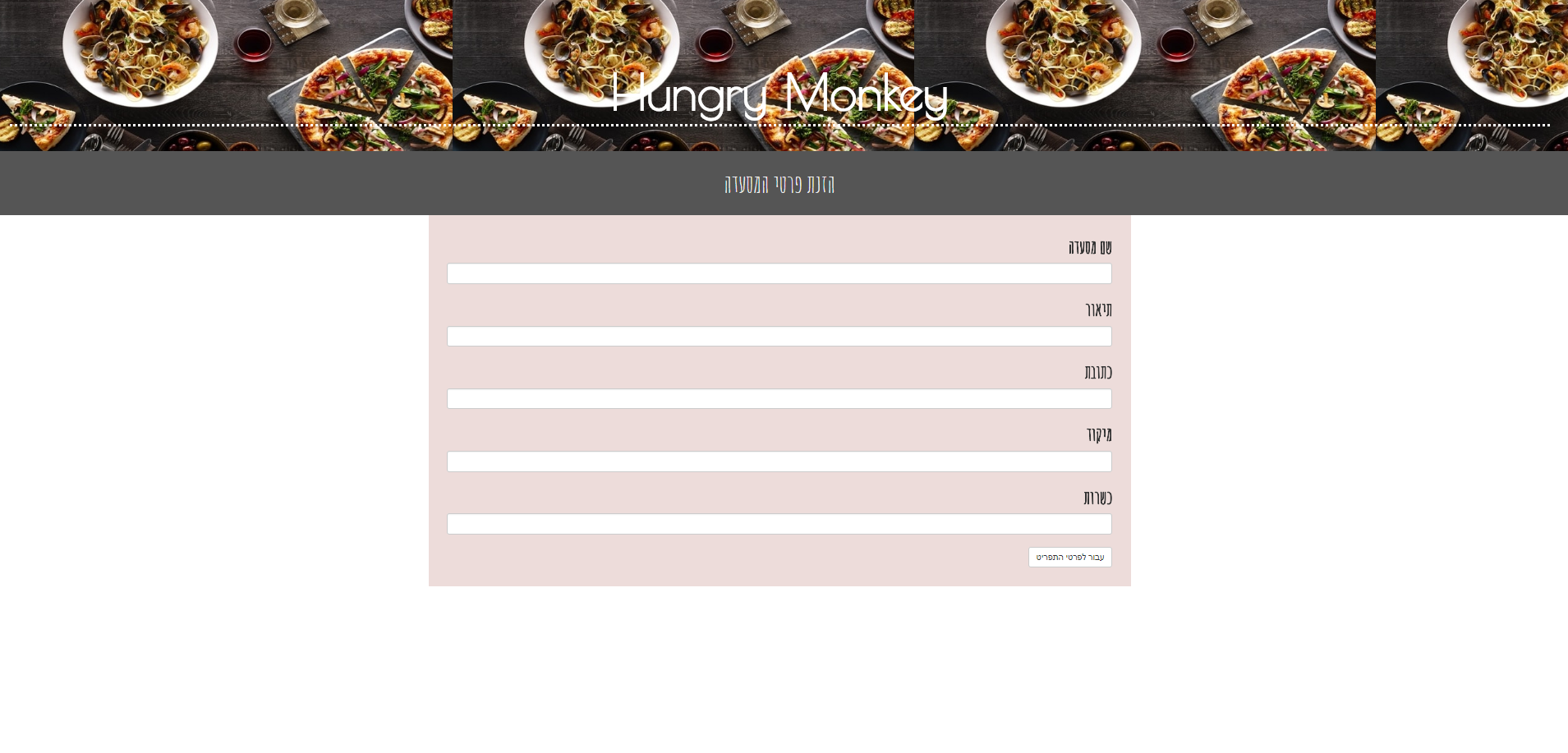
**הפיצ'רים של ממשק הניהול הינם:**

* הזנת ועדכון פרטי המסעדה והתפריט.
* הפקת קובץ תפריט עבור הפרטים שהוזנו.

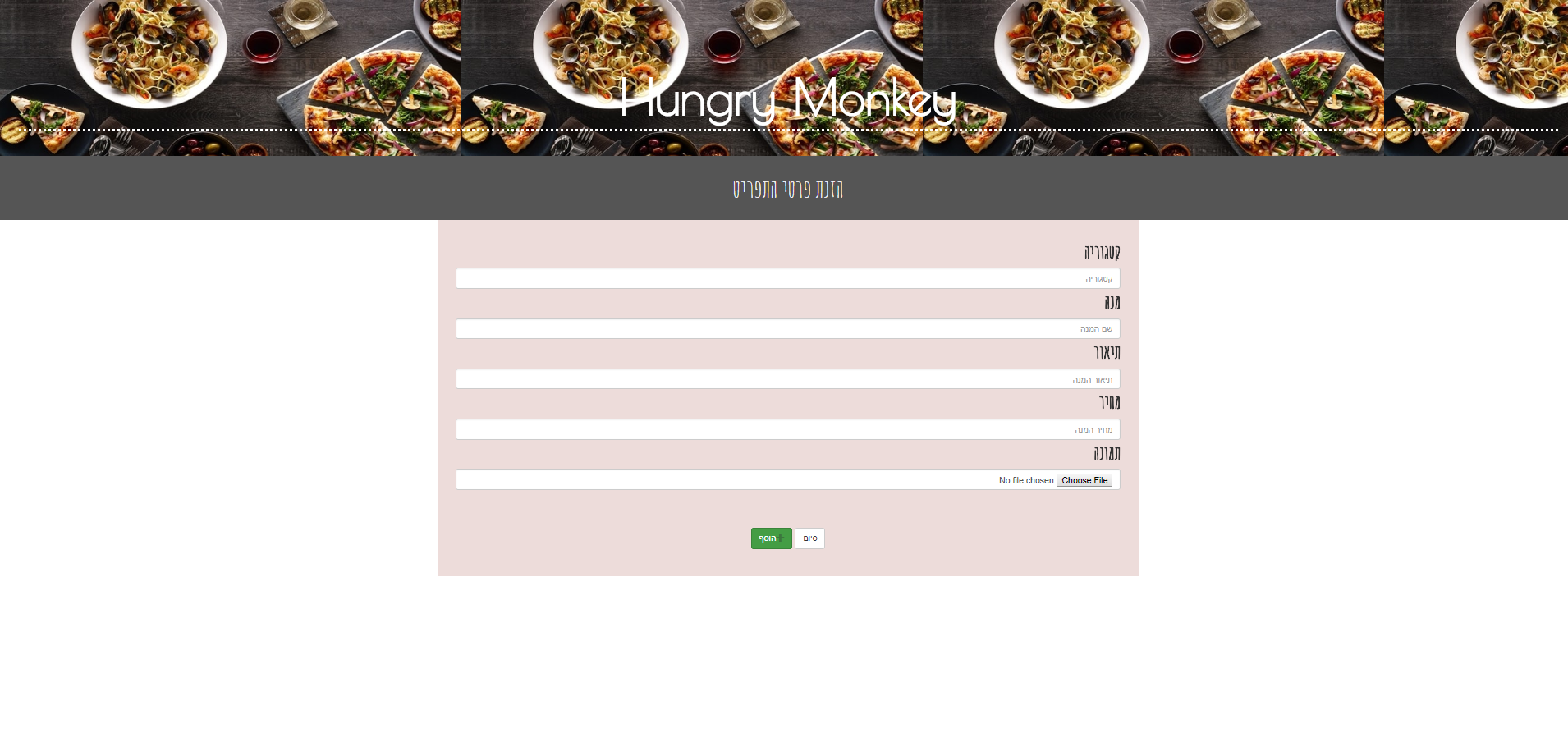
# 5.3. תיאור מערכת שמומשה

# ממשק ניהול

* מסך הזנת פרטי המסעדה. בדף זה ניתן להזין את כל הפרטים של המסעדה(שם, כתובת, תיאור, כשרות)



* מסך הזנת פרטי התפריט: זה הדף הבא ובעצם כאן מזינים את כל פרטי המנות כולל תמונות.



* בסוף ההזנה, מתן אופציה נוספת לזה שמזין את הפרטים לראות מה הוא הזין וכמובן הפקת תפריט לפי כל הקטגוריות שהזין.



אפליקציית אוכל

להשלים...