**MiniBusiness - אפיון**

1. תיאור המערכת
   1. מטרת הפרויקט

האפליקציה נועדה לאפשר לבעלי עסקים קטנים לנהל את העסק שלהם בצורה נוחה ואינטואיטיבית. לסרוק קבלות ולסווג הוצאות באופן אוטומטי, לייצר דוחות מס שנתיים ולהפיק תובנות עסקיות בעזרת AI. המטרה היא לפשט את ניהול העסק ולהקל על קבלת החלטות.

* 1. קהל היעד

קהל היעד העיקרי של האפליקציה הוא בעלי עסקים קטנים, פרילנסרים ועצמאיים אשר זקוקים לפתרון ניהול קל ונוח. האפליקציה תענה על הצורך בניהול מסודר של הוצאות והכנסות.

* 1. פונקציונליות של המערכת

האפליקציה תכלול את הפונקציות הבאות:

1. העלאת חשבוניות של הוצאות והכנסות ושמירה בענן.
2. זיהוי נתונים מהקבלה (תאריך, סכום, ספק) באמצעות OCR.
3. ניתוח נתונים וסטטיסטיקות על הוצאות והכנסות.
4. המלצות AI לניהול תקציב חכם.
5. שיוך של ההוצאה לקטגוריה רלוונטית.
   1. בעיות שהמערכת פותרת

האפליקציה שומרת בענן את כל החשבוניות של העסק ולוחות הזמנים, פותרת את הבעיה של ניהול מבולגן של הוצאות והכנסות, חוסכת בזמן בקליטת נתונים פיננסיים, נותנת תובנות עסקיות חכמות לניהול פיננסי טוב יותר.

* 1. טכנולוגיות בשימוש

הפרויקט יפותח באמצעות הטכנולוגיות הבאות:

1. צד שרת (Backend): .NET 6 לבניית ה-API.

2. צד לקוח (Frontend): React.js לבניית ממשק משתמש אינטראקטיבי.

3. צד לקוח angular: (Fronted) לבניית מסך התחברות למערכת.

4. מסד נתונים: SQL לאחסון נתונים של משתמשים וספקים ולקוחות, חשבוניות ונתונים נוספים.

5. העלאה לענן – Aws – יעלה את כל החשבוניות לענן.

* 1. מתודולוגיית פיתוח (Agile)

הפרויקט יפותח במתודולוגיית Agile עם חלוקה ל-5 ספרינטים בני שבועיים כל אחד. כל ספרינט יתמקד במשימות ליבה עיקריות.

1. עיצוב ממשק משתמש ויזואלי

התחברות

- מסך כניסה/רישום:

הרשמה

- מסך ניהול חשבוניות:

הצגת קבצים ע"פ קטגוריות

- מסך ניתוח נתונים:

- מסך ניהול לקוחות וספקים:

העלאת קובץ

1. איפיון תמציתי

המלצות המערכת

ניתוח נתונים – גרפים וטבלאות

* 1. רשימת פונקציות המערכת:

ניהול הרשאות משתמשים

דוחות של לקוחות

**- קבלת חשבוניות**

- Route : GET /api/invoices

- פרמטרים:

- Body: { userId: int }

- פלט:

- Data: { file[]: File[], success: bool }

- לוגיקה: קבלת החשבוניות ממסד הנתונים ע"פ משתמש.

**- הוספת חשבונית**

- Route : POST /api/invoices

- פרמטרים:

- Body: { file:File, date: string }

- פלט:

- Data: { invoiceId: string, success: bool }

- לוגיקה: בדיקת תקינות הנתונים והוספת החשבונית למסד הנתונים ע"פ קטגוריה.

**- עדכון חשבונית**

- Route : PUT /api/invoices

- פרמטרים:

- Body: { file: File, invoiceId: string }

- פלט:

- Data: { file: File, success: bool }

- לוגיקה: בדיקת תקינות הנתונים ועדכון החשבונית במסד הנתונים.

**- מחיקת חשבונית**

- Route : DELETE/api/invoices

- פרמטרים:

- Body: { file: File, invoiceId: string }

- פלט:

- Data: { success: bool }

- לוגיקה: מחיקת החשבונית ממסד הנתונים.

**- קבלת המלצות**

- Route : GET /api/ recommendations

- פרמטרים:

- Body: {userId: Int, fileList: File[] }

- פלט:

- Data: { injest: string }

- לוגיקה: קבלת חשבוניות והבאת המלצות על כדאיות של מוצרים מסוימים, הרגלי צריכה וכדו' ע"פ ההוצאות וההכנסות.

**- יצירת דוחות מס**

- Route : GET /api/ tax

- פרמטרים:

- Body: { year: Int, fileList: File[] }

- פלט:

- Data: { reportUrl: string }

- לוגיקה: קבלת חשבוניות והבאת חשבוניות שנתיות ע"פ ההוצאות וההכנסות.

* 1. אימות והרשאות

האפליקציה תבצע אימות באמצעות JWT Authentication. יהיו תפקידים כמו מנהל, משתמש רגיל, עם הרשאות שונות.

* 1. אפליקצית ניהול

יכולות המנהל במערכת:

- צפייה בדוחות וסטטיסטיקות שימוש.

- ניהול משתמשים והרשאות.

- עדכון פרמטרים גלובליים במערכת – ממוצע משתמשים וכו'.

1. תרשים מבנה הטבלאות (Database Schema)

**טבלת משתמשים (Users)**

| שם השדה | סוג נתונים | תיאור |

|---------|------------|--------|

| id | ObjectId | מזהה ייחודי |

| name | String | שם מלא |

| email | String | אימייל (ייחודי) |

| password | String | סיסמה מוצפנת |

| role | String | תפקיד במערכת |

**טבלת קבלות (Receipts)**

| שם השדה | סוג נתונים | תיאור |

|---------|------------|--------|

| id | ObjectId | מזהה ייחודי |

| userId | ObjectId | מזהה המשתמש |

| amount | Number | סכום ההוצאה |

| category | String | קטגוריה |

| date | Date | תאריך ההוצאה |

| supplier | String | שם הספק |

1. הגדרת סבבי פיתוח (ספרינטים)

ספרינט 1: הקמת תשתית בסיסית

**מטרה:** הקמת סביבת עבודה בסיסית ובניית תשתיות.

**משימות:**

פתיחת פרויקטים חדשים ב-.NET 9 ו-React.js.

יצירת טבלאות במסד הנתונים (Users, Receipts).

פיתוח API בסיסי ב-.NET 9 עם endpoints לניהול משתמשים.

הטמעת מערכת אימות משתמשים (Registration/Login) עם JWT.

ספרינט 2: ניהול חשבוניות

**מטרה:** פיתוח פונקציות לניהול חשבוניות.

**משימות:**

פיתוח endpoint להוספת חשבונית (POST /api/invoices).

פיתוח endpoint לקבלת חשבוניות (GET /api/invoices).

פיתוח endpoint לעדכון חשבונית (PUT /api/invoices).

פיתוח endpoint למחיקת חשבונית (DELETE /api/invoices).

ספרינט 3: ניתוח והמלצות

**מטרה:** פיתוח פונקציות לניתוח נתונים והמלצות.

**משימות:**

פיתוח endpoint לקבלת המלצות (GET /api/recommendations).

פיתוח פונקציה לניתוח הוצאות והכנסות.

פיתוח ממשק משתמש להצגת המלצות וסטטיסטיקות.

ספרינט 4: דוחות והגדרות

**מטרה:** פיתוח פונקציות ליצירת דוחות והגדרות מערכת.

**משימות:**

פיתוח endpoint ליצירת דוחות מס (GET /api/tax).

פיתוח ממשק משתמש ליצירת דוחות.

ניהול פרמטרים גלובליים במערכת (ממוצע משתמשים וכו').

ספרינט 5: שיפורים, אבטחה ופריסה

**מטרה:** שיפוט, אבטחה ופריסה של האפליקציה.

**משימות:**

ביצוע בדיקות אבטחה ואופטימיזציות ב-API.

שיפוט ממשק המשתמש לפי משוב.

פריסת האפליקציה בשרת Render.