**מדריך קבצי JavaScript - מעודכן על בסיס הקוד הקיים**

**קבצים עיקריים (שורש)**

**calendar-integration.js**

**מטרה בפועל:** קובץ מרכזי המשמש כמתאם בין כל המודולים ואוסף נתונים לשליחה סופית

**פונקציות שזוהו בקוד:**

* collectFormData() - פונקציה מרכזית ומורכבת שאוספת את כל הנתונים מהטופס
  + מטפלת בלוגיקת רכב שני (hasSecondCar) ושיתוף כתובות
  + אוספת נתוני vehicle data מ-dataset attributes (צבע, גיר, סוג מנוע וכו')
  + **אוספת נתוני תקלות וסוגי גרר** מהמערכת החדשה
  + יוצרת מבנה מורכב של formData עם כל הנתונים הדרושים
  + מחברת נתונים מ-PricingManager ו-VehicleManager
* collectPricingData() - אוסף נתוני תמחור מ-PricingManager
  + קורא את המצב הנוכחי של התמחור (טיר נבחר, מחירים מחושבים)
  + מטפל במחירים ידניים מול אוטומטיים
* **collectDefectDetails()** - **חדש** - אוסף תקלות נבחרות מהמודל האינטראקטיבי
  + קורא תגיות תקלות מהקונטיינר
  + מחזיר מחרוזת מפורמטת של התקלות
* **collectTowSelection()** - **חדש** - אוסף סוגי גרר נבחרים מהמודל
  + קורא תגיות גרר מהקונטיינר
  + מחזיר מחרוזת מפורמטת של סוגי הגרר
* processAddress() - מטפל בכתובות מורכבות עם טקסט מקורי וכתובת פיזית
* resetFormKeepUserData() - איפוס מתקדם של הטופס
  + מנקה לחלוטין את כל נתוני הכתובות והרכבים
  + **מאפס בחירות תקלות וגרר**
  + משמר רק פרטי משתמש (אימייל, חברה, מחלקה)
  + מאפס את מערכת התמחור למצב התחלתי
* **resetDefectSelections()** - **חדש** - מאפס בחירות תקלות למצב ברירת מחדל
* **resetTowSelection()** - **חדש** - מאפס בחירות גרר למצב ברירת מחדל
* sanitizeText() - מנקה טקסט מגרשיים עבריים ואנגליים
* getSelectedPaymentType() - מזהה את סוג התשלום הנבחר

**חשיפה גלובלית של פונקציות:**

* window.collectDefectDetails - גישה לפונקציית איסוף התקלות
* window.collectTowSelection - גישה לפונקציית איסוף הגרר

**PricingManager.js**

**מטרה בפועל:** מחלקה מלאה וחכמה לניהול תמחור דינמי

**פונקציות מרכזיות שזוהו בקוד:**

* init() - מאתחל את כל המערכת כולל Google Maps
* calculateTotalPrice() - פונקציה מרכזית המחשבת מחיר כולל
* calculateDistance() - משתמשת ב-Google Maps Distance Matrix API
* setupAutomaticCalculation() - מערכת חכמה למעקב אחר שינויים
* getRecommendedTier() - לוגיקה מתקדמת להמלצת מחיר לפי זמן
* updatePriceDisplay() - מעדכן את כל התצוגות של המחירים
* recalculatePrices() - מחשב מחדש בעת שינוי הגדרות

**תיקיית js/modules/**

**FormManager.js**

**מטרה בפועל:** מנהל את האינטראקציות והלוגיקה של הטופס

**פונקציות חדשות שנוספו:**

* **setupDefectSelector()** - **חדש** - מאתחל מערכת בחירת התקלות
* **showDefectModal()** - **חדש** - מציג מודל בחירת התקלות
* **createDefectModal()** - **חדש** - יוצר את המודל האינטראקטיבי לתקלות
* **setupDefectOptions()** - **חדש** - מגדיר אירועי בחירה לתקלות (בחירה מרובה)
* **confirmDefectSelection()** - **חדש** - מאשר ושומר בחירת תקלות
* **updateSelectedDefects()** - **חדש** - מעדכן תצוגת התקלות הנבחרות
* **setupTowSelector()** - **חדש** - מאתחל מערכת בחירת הגרר
* **showTowModal()** - **חדש** - מציג מודל בחירת הגרר
* **createTowModal()** - **חדש** - יוצר את המודל האינטראקטיבי לגרר
* **setupTowOptions()** - **חדש** - מגדיר אירועי בחירה לגרר (בחירה מרובה)
* **confirmTowSelection()** - **חדש** - מאשר ושומר בחירת הגרר
* **updateSelectedTow()** - **חדש** - מעדכן תצוגת הגרר הנבחר

**פונקציות קיימות:**

* ניהול מעברים בין סוגי גרירה
* הצגה/הסתרה של חלקי טופס
* ניהול רכב שני (הוספה והסרה)
* ניהול שיתוף כתובות בין רכבים
* אימות טופס לפני שליחה
* ניהול מצבי UI (loading, success, error)

**תיקיית js/utils/**

**formatters.js, validators.js, helpers.js**

פונקציות עזר שנשארו ללא שינוי מהמערכת המקורית.

**תיקיית js/config/**

**constants.js**

נשאר ללא שינוי מהמערכת המקורית.

**רשימת תקלות וסוגי גרר נתמכים**

**תקלות זמינות:**

1. אין חשמל
2. פנצ'ר
3. תאונה
4. גלגל עקום או שבור
5. נזילת מים/שמן
6. לא נדלק/לא מניע
7. חניון תת קרקעי
8. מניע/נדלק ונוסע
9. אחר - מילוי ידני

**סוגי גרר זמינים:**

1. מובילית
2. רמ-סע
3. משקפיים
4. דולי

**סדר הטעינה המומלץ:**

1. constants.js - קבועים בסיסיים
2. formatters.js, validators.js, helpers.js - כלי עזר
3. כל המודולים (ApiManager.js, AddressManager.js, VehicleManager.js, PricingManager.js, FormManager.js)
4. calendar-integration.js - פונקציות איסוף נתונים (תלוי בכל האחרים)
5. main.js - חיבור הכל יחד

**עדכונים במערכת Google Apps Script**

**שינויים בתיאור האירוע:**

* **ב-createDefectiveDescription():** הוספת חלק "🚛 גרר מתאים" אחרי פירוט התקלה
* **ב-createExchangeDescription():** הוספת חלק "🚛 גרר מתאים" אחרי פירוט התקלה של הרכב התקול

**מבנה המידע החדש ביומן:**

🔧 פירוט התקלה:

────────────────────

[רשימת התקלות שנבחרו]

🚛 גרר מתאים:

────────────────────

[רשימת סוגי הגרר שנבחרו]

**הערות חשובות מעודכנות:**

* המערכת מורכבת יותר ממה שהמדריך המקורי תיאר
* **נוספו מודלים אינטראקטיביים** לבחירת תקלות וסוגי גרר
* יש שימוש מתקדם ב-data attributes לשמירת מידע זמני
* המערכת תומכת בכתובות מורכבות עם שמירת טקסט מקורי ופיזי
* ה-PricingManager הוא מחלקה מלאה עם state management
* יש אינטגרציה מתקדמת עם Google Maps לחישוב מרחקים
* המערכת מנהלת רכבים מרובים עם שיתוף נתונים ביניהם
* **המערכת כוללת מודלים רספונסיביים** עם עיצוב מותאם למובייל וטאבלט
* כל מודול משתמש ב-window object לחשיפת API ציבורי
* **המערכת משתלבת בצורה חלקה** עם מערכת איסוף הנתונים והיומן הקיימת