

HLD – Coupon Manager מסמך

גרסה: v3 | תאריך: 28.12.2025

מסמך זה מתאר ארכיטקטורה ברמת High Level למערכת לניהול קופונים וקבוצות משתמשים, כולל צד לקוח, צד שרת ו-MongoDB.

1. מטרת ו-Scope

- ניהול קופונים מסוג קופון חנות יחיד (SINGLE) ותו/מולטי קופון (MULTI).
- שיוך קופון לקבוצה אחת בלבד (Group).
- ניהול סכומים: סכום מלא, סכום נוצל, סכום נשאר (חישוב אוטומטי).
- ניהול סטטוסים לקופון + תאריך תוקף.
- שמירת תמונה/ות לקופון לצורך סריקה מהירה בקופה.
- ניהול קבוצות והזמנות לפי מייל עם הרשאות צפייה/עריכה/ניהול.
- Login/Signup: אימייל+סיסמה או Google.
- התראות מייל על קופונים שפגים תוקפם לפי הגדרת המשתמש + תהליך יומי.
- חיפוש לפי חנות: הצגת גם קופונים ישירים של החנות וגם מולטי קופונים שמכילים אותה.
- השלמה אוטומטית (autocomplete) לשם חנות ולשם מולטי קופון.

2. בעלי תפקידים והרשאות

| תפקיד | יכולות מרכזיות |
|---------------------|---|
| viewer | צפייה בקופונים ותמונות בקבוצה. |
| editor | כמו viewer + יצירה/עריכה של קופונים ועדכון שימוש (סכום). |
| admin (בקבוצה) | כמו editor + ניהול חברים והזמנות בקבוצה, מחיקות/ביטולים לפי מדיניות. |
| super_admin (מערכת) | ניהול חנויות, ניהול MultiCouponDefinitions, קבלת התראות על מולטי לא ממופה, ניהול מערכת. |

3. ישויות עסקיות וקשרים

ישויות עיקריות:

- Store: חנות (קטלוג גלובלי).
- MultiCouponDefinition: שם מולטי -> רשימת חנויות (מנוהל ע"י super_admin).
- Coupon: קופון השייך לקבוצה אחת. SINGLE משויך לחנות אחת. MULTI מזהה לפי multiCouponName ומקבל resolvedStoreIds.
- Group + GroupMember: קבוצה וחברים עם role.
- Invitation: הזמנה לקבוצה לפי מייל.
- NotificationPreference: העדפות תזכורות לפי ימים לפני תוקף.
- AdminAlert: אירוע/התראה פנימית (למשל מולטי לא ממופה).

4. ארכיטקטורה - רכיבים

- Client (Netlify): אפליקציית React+TypeScript הכוללת מסכים למשתמשים + מסכי Admin (super_admin).
- Server (Render): Node.js + Express + TypeScript. שכבות: Routes, Controllers, Services, Data (Mongoose).
- MongoDB: אחסון נתונים (Users/Groups/Coupons/Stores/Definitions וכו').
- Image Storage: אחסון אובייקטים (מומלץ Cloudinary/S3). ב-Mongo נשמרים URLs ומטא-דאטה בלבד.
- Email Provider: Resend/SendGrid/SMTP - לשליחת הזמנות ותזכורות.
- Daily Job: תהליך יומי (Render Cron או scheduler) שמבצע בדיקות תוקף ושולח מיילים.

5. זרימות מרכזיות

5.1 הוספת קופון SINGLE

- המשתמש בוחר קבוצה.
- המשתמש בוחר חנות (עם autocomplete).
- מזין: כותרת, תאריך תוקף, סכום מלא, מטבע, הערות.
- מעלה תמונה/ות (upload ישיר ל-storage) ומצרף URL-ים לקופון.
- השרת שומר ומחשב remainingAmount=total-used, וסטטוס ראשוני ACTIVE.

5.2 הוספת קופון MULTI (עם/בלי מיפוי)

- המשתמש מזין multiCouponName (עם autocomplete).
- השרת מחפש MultiCouponDefinition לפי השם:
- אם נמצא: שומר (snapshot) resolvedStoreIds ומסמן mappingStatus=MAPPED.
- אם לא נמצא: מאפשר יצירה, שומר resolvedStoreIds=[], mappingStatus=UNMAPPED.
- במקרה UNMAPPED: השרת מייצר AdminAlert ושולח מייל ל-super_admin על מולטי לא ממופה.

5.3 עדכון שימוש (סכום בלבד)

- המשתמש מעדכן סכום שימוש (ADD או SET).
- השרת מוודא usedAmount לא עובר totalAmount.
- השרת מעדכן סטטוס אוטומטית: ACTIVE / PARTIALLY_USED / USED.

5.4 חיפוש לפי חנות

- המשתמש בוחר חנות (autocomplete).
- המערכת מחזירה את כל הקופונים בקבוצה שהם:
- SINGLE עם storeId=החנות, או
- MULTI עם resolvedStoreIds שמכיל את החנות.
- קופוני MULTI לא ממופים (UNMAPPED) לא יופיעו בחיפוש לפי חנות עד שיוגדר להם מיפוי.

5.5 התראות תפוגה ותהליך יומי

- לכל משתמש מוגדרים ימים לפני תוקף (למשל 3,7).

- תהליך יומי מאתר קופונים שעומדים לפוג בעוד X ימים ושולח מייל לכל משתמש בנפרד לפי הקבוצות שלו והרשאותיו.
- התהליך גם מסמן קופונים שפג תוקפם כ-EXPIRED (אם לא USED/CANCELLED).

6. דרישות לא פונקציונליות

- אבטחה: JWT + Refresh Token, RBAC לפי GroupMember.role, הפרדת super_admin.
- עקביות נתונים: מניעת $usedAmount > totalAmount$ (עדכון אטומי).
- ביצועים: אינדקסים על groupId/status/expiryDate וחיפוש prefix לחנויות ומולטי.
- תפעול: לוגים לאירועי Job ותקלות אימייל/Storage, ומדדים בסיסיים.

7. פריסה

- Client ב-Netlify עם משתנה סביבה VITE_API_BASE_URL.
- Server ב-Render עם משתני סביבה (Mongo, JWT, Google OAuth, Email, Job Secret).
- MongoDB Atlas (מומלץ) + IP allowlist/credentials.
- Cron Job ב-Render שמכה ל-endpoint פנימי מוגן secret.