

תיאור עבודת ממשק הצגת חשבוניות למערכת Priority

1 צד השרת –

- 1.1 ממשק עם Priority – בקשות Api למערכת Priority לטובת משיכת חשבוניות.
- 1.2 שמירת החשבוניות ב-DB לטובת שליפה מהירה ו-Caching, חיפוש לפי סטטוס חשבונית או לפי שם/תאריך.
- 1.3 חיפוש – חיפוש יתבצע ע"ב Caching או שליפות מה-DB, ויתאפשר חיפוש לפי שם ותאריך ולפי סטטוס חשבונית.
- 1.4 בדיקה לגבי שינויים שנעשו בחשבוניות (שליפה מול מידע קיים), והפצת הודעות לשירות התראות במידת הצורך לטובת עדכון המשתמש.
- 1.5 DB – אינדוקס לפי נתונים חוזרים (כגון – טווחי תאריכים, סטטוס וכד').

2 צד לקוח –

- 2.1 צד הלקוח יבצע שליפות לפי תאריך אחרון, ב-pagination (בכל פעם 10 לצורך העניין, ושליפה נוספת לפי גלילה).
- 2.2 פילטור צד לקוח – בחרתי בגישה היברידית לטובת ביצוע החיפוש של חשבוניות על בסיס האפשרויות שמוצגות למשתמש (סטטוס, שם ותאריך) – לפי גישה זו, יתבצע קודם כל פילטור בצד הלקוח (פילטור על בסיס החשבוניות הקיימות, בדומה למימוש בקוד ב-client), וכתלות במספר התוצאות העונות לחיפוש, ייתכן שתישלח קריאה לשרת לטובת שליפה עם יותר תוצאות.
- לדוגמה – הפילטור של החשבוניות הקיימות מחזיר חשבונית אחת העונה לקריטריונים, זו תוצאה שלא מספקת שכן המשתמש לא יוכל לגלול ולהציג עוד, כך שרק אם יהיו 10 חשבוניות שמתאימות לפילטור (בתצורה שתאפשר גלילה למטה), אסתפק בפילטור ב-client ולא אוציא קריאה לשרת.
- 2.3 חיבור לשירות התראות – ע"ב SignalR – הצגת חיווי למשתמש אם בוצע שינוי בחשבונית.

3 שירות התראות –

- 3.1 חיבור צד לקוח – שירות התראות - לטובת שליחת עדכונים על שינויים שבוצעו בחשבוניות.
- 3.2 האזנה לתור הודעות (אליו השרת שולח עדכונים) והפצה למשתמשים הרלוונטיים.