

דוט נט תשפ"ו - מיני פרויקט במערכות חלונות - תיאור כללי

1	דוט נט תשפ"ו - מיני פרויקט במערכות חלונות - תיאור כללי
2	תיאור כללי של המערכת
2	תוספות לבחירה
2	שימוש בכלי בינה מלאכותית (AI)
2	ט.ל.ח.
3	פירוט שלבי הפרויקט
3	ציון הפרויקט
3	שלבי הפרויקט
4	משתמשי המערכת
4	משתמש מנהל
4	משתמש שליח
4	אפיון המסכים - תצוגת ממשק המשתמש
4	פירוט המסכים בראשי פרקים
4	תרשים המסכים
6	התנהלות כללית של המערכת
6	סוגי שילוח
6	חישוב מרחק הזמנה בפועל
6	סוגי סיום משלוח
7	זמנים בחיי הזמנה
7	סטטוס הזמנה (OrderStatus)
7	סטטוס עמידה בזמנים (ScheduleStatus)
7	חישוב שכר של שליח
8	תסריטים אפשריים לדוגמא
9	תרשימים להמחשה - תרחישים חיי הזמנה
12	שכבת הנתונים DAL (שלב 1 בפרויקט)
12	תיאור כללי
12	ישות תצורה (קונפיגורציה)
14	ישויות נתונים - DO
17	חוזי השירות (ממשקים) של שכבת הנתונים - DalApi
18	אתחול הנתונים - Initialize
19	איפוס נתוני שכבת הנתונים
19	תוכנית הבדיקה - DalTest
20	שכבת הנתונים DAL - שמירת הנתונים בקבצי XML (שלב 3 בפרויקט)
20	השכבה הלוגית BL (שלב 4 בפרויקט)
20	תיאור כללי
21	ישויות הנתונים הלוגיות (חוזי נתונים - מחלקות) - BO
29	ישויות שירות לוגיות (חוזי שירות - ממשקים) - BIApi
34	מחלקות עזר סטטיות פנימיות ב BL - לניהול לוגי של ישות נתונים
36	שכבת התצוגה (שלב 5 ושלב 6)
36	פירוט המסכים והפונקציונליות שלהם
40	סימולטור (שלב 7)
40	חישוב קואורדינטות של כתובת בצורה אסינכרונית
40	סימולציית (הדמיית) פעילות המערכת - סימולציית פעילות השליחים
41	התנהגות המסכים הפתוחים בזמן הפעלת הסימולטור
43	נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות
43	חישוב ידני מפורש
44	חישוב דרך הקוד - פנייה דרך הקוד לאתרי geocoding

תיאור כללי של המערכת

במסגרת הפרויקט של הקורס, נפתח מערכת לניהול משלוחים בחברה - ניהול וטיפול בהזמנות, שליחים ומשלוחים. למשל: מסעדה שבין השאר גם מקבלת הזמנות של אוכל ומבצעת משלוחים, חנות ספרים שבין השאר גם מספקת משלוחי ספרים וכדומה.

כל זוג שותפים לפרויקט, יבחר את סוג החברה שעבורה תפותח המערכת/הפרויקט.

המערכת תאפשר ל"מנהל" המערכת (המכונה "מנהל") לנהל מאגר של שליחים ולנהל מאגר של הזמנות.

בנוסף, המערכת תאפשר לכל "שליח" לבחור לטפל בהזמנות (לבצע משלוח) ולעדכן את המערכת לאחר הטיפול בהזמנה.

המנהל פותח הזמנות אשר נמצאות בטווח שבין כתובת החברה לבין המרחק האווירי המירבי שהחברה קבעה לעצמה. בנוסף, יוכל להגדיר עבור כל ההזמנות בחברה, טווח זמן אספקה מירבי שאליו החברה מתחייבת (משתדלת) ואשר ממנו והלאה הזמנה נחשבת שסופקה באיחור. המנהל יוכל להגדיר גם טווח זמן סיכון אשר הזמנה שנכנסת לטווח הזה ועדיין לא סופקה נחשבת בסיכון.

שליח יוכל לבחור הזמנה לטיפול מבין ההזמנות הפתוחות. כל שליח יוכל לטפל בהזמנה אחת בו זמנית. כאשר שליח בוחר לטפל בהזמנה, המשמעות היא שהיא הוקצתה עבורו והוא יבצע את המשלוח.

שליחים יוכלו לדווח על סיום המשלוח לפי סוג סיום המשלוח. למשל: ההזמנה סופקה והיא נסגרת. המזמין סירב לקבל את ההזמנה והיא חוזרת לחברה ונסגרת. המזמין לא נמצא וההזמנה חוזרת לחברה ונפתחת להמתנה למשלוח חדש.

מנהל יוכל לצפות בכל היסטוריית המשלוחים לצורך בקרה וניהול. כל שליח יוכל לצפות בהיסטוריית המשלוחים של עצמו בלבד.

מכיוון שהמערכת שלנו עושה הדמיה ועל מנת לתת תחושת של התקדמות בזמן, נזדקק לשעון מערכת שיתוחזק בנפרד מהשעון האמיתי של המחשב. מנהל המערכת יוכל לאתחל ולעדכן (לקדם) את שעון המערכת. בכל פעולה, הזמן הנוכחי של המערכת יהיה הערך הנוכחי של שעון המערכת (ולא הזמן הנוכחי של שעון המחשב DateTime.Now).

במהלך הסמסטר, נבנה את המערכת בשלבים ע"פ ארכיטקטורת מודל השכבות. נושא זה נלמד בהרחבה במהלך הסמסטר.

תוספות לבחירה

תוספות לבחירה יסומנו כך: **בצבע כתום**.

את פירוט התוספות והניקוד עליהן ניתן למצוא במסמך המתעדכן הבא: [דוט נט תשפו - פרויקט - תוספות](#).

סטודנטים יוכלו להציע תוספות לבחירה באישור המרצה, מעבר לרשימת התוספות המקורית. המרצה יעריך את הניקוד של התוספת שהוצעה וככל שניתן יעדכן את מסמך התוספות.

התוספות הן רכיב משמעותי בציון הפרויקט ולכן המלצתנו היא לממש כמה שיותר מהן. וכן, להתחיל לצבור תוספות כבר בשלבים הראשונים.

שימוש בכלי בינה מלאכותית (AI)

ראו בסילבוס.

ט.ל.ח.

המסמכים המתארים את הפרויקט כולו (המסמך הכללי ומסמכי השלבים) הם מלאי פרטים ותוכן ונכתבו תוך השקעה מרובה. נעשה מאמץ מאוד גדול כדי לדייק ולעזור, אך כולנו בשר ודם. במידה ויתגלו טעויות או סתירות, אנא שלחו מייל למרצה שלכם עם שאלתכם והמרצה יעביר את המייל לרכזים. מקומות שיתוקנו יסומנו כך: **בצבע ורוד**.

בכל מקרה, גם אם לדעתכם יש טעות תוכלו להפעיל הגיון בריא ולדעת להצדיק את החלטתכם בצורה מתקבלת על הדעת.

פירוט שלבי הפרויקט

ציון הפרויקט

ציון הפרויקט יחושב על פי 3 מרכיבים:

1. הפרויקט יפותח בשמונה שלבים. הציון המקסימלי המשוקלל על כל השלבים שיבוצעו לפי דרישות חובה בלבד הוא 100 נקודות ויהווה 30% מציון הפרויקט.
 2. בשיעור האחרון תתבצע הרצה והגנה בעל פה מול המרצה, ציון עד 100 נקודות שיהווה 60% מציון הפרויקט.
 3. בנוסף, ניתן לצבור עד 15 נק' ע"י ביצוע תוספות לבחירה.
- שימו לב: כפי שכבר ציינו לעיל התוספות הן רכיב משמעותי בציון הפרויקט ולכן המלצתנו היא לממש כמה שיותר מהן. וכן, להתחיל לצבור תוספות כבר בשלבים הראשונים. בזכות התוספות, סכום 3 המרכיבים יכול להגיע ל 105 נקודות. אך הציון המקסימלי על הפרויקט כולו יהיה כמובן 100.

שלבי הפרויקט

עבור כל אחד משמונת שלבי הפרויקט קיים מסמך נפרד המכיל פירוט מדויק והסברים מעמיקים. לינק לכל מסמך נמצא במודל.

תיאור קצר עבור כל אחד משמונת שלבי הפרויקט. כולל משקל הציון עבור כל שלב:

שלב 0 - שלב מקדים - שבוע אחד (5% מהציון)

- הכרת סביבת עבודה
- יצירת מאגר גיט מקומי ומשותף בענן github.com
- עבודה בסיסית עם C#

שלב 1 - שכבת נתונים (DAL) - שבוע אחד (10% מהציון)

- יצירת חוזה נתונים של שכבת הנתונים (ישויות נתונים DO)
- יצירת חוזה שירות ע"י ממשקים (DalApi) למימוש הגישה לישויות (לכל ישות ממשק משלה)
- שמירת נתונים ברשימות אובייקטים בזכרון
- מימוש מתודות גישה לרשימות הנתונים (CRUD)
- חריגות
- ממשק משתמש (קונסול) - תוכנית בדיקות של שכבת נתונים
- יצירת בסיס נתונים התחלתי

שלב 2 - שכלול ממשק שכבת הנתונים ועבודה עם LINQ - שבוע אחד (10% מהציון)

- יצירת חוזה שירות אחיד של שכבת הנתונים ע"י ממשק גנרי וירושה ממנו
- עדכון ממשקים לשימוש באוספי נתונים כלליים (IEnumerable<>)
- עדכון כל השכבות לעיבוד נתונים בעזרת LINQ בלבד (וללא לולאות)
- הוספת חריגות

שלב 3 - שמירת נתונים ב-XML - שבוע (10% מהציון)

- מימוש נוסף לשכבת הנתונים בעזרת שמירת הנתונים בקבצי XML

שלב 4 - מודל השכבות, תבניות עיצוב והשכבה הלוגית (BL) - שבועיים (20% מהציון)

- השלמת מודל השכבות
- תבניות עיצוב Singleton ו-Simple Factory Method, כולל בשכבות הנתונים
- יצירת חוזה נתונים של שכבה לוגית (ישויות לוגיות BO)
- יצירת חוזה שירות ע"י ממשק גנרי (BIApi) של השכבה הלוגית
- ממשק משתמש (קונסול) - תוכנית בדיקות של השכבה לוגית

שלב 5 - שכבת התצוגה - ממשק גרפי בסיסי - שבוע אחד (10% מהציון)

- שכבת תצוגה בסיסית (ממשק גרפי)

שלב 6 - שכבת התצוגה - ממשק גרפי מלא - 3 שבועות (25% מהציון)

- השלמת ממשק גרפי מלא
- עדכוני תצוגה יעשו דרך binding מתוך XAML בלבד

שלב 7 - תהליכונים וסימולטור - שבועיים (10% מהציון)

- עבודה עם תהליכונים
- הוספת סימולטור (בפרויקט נפרד של VS)

● שימוש בסימולטור בממשק המשתמש הגרפי

משתמש המערכת

ישנם 2 סוגי משתמשים במערכת: משתמש "מנהל" ומשתמש "שליח"

משתמש מנהל

- יוכל לנהל את מאגר השליחים ולצפות ברשימת השליחים (לסנן, למיין ולקבץ לפי קטגוריות)
 - לראות עבור כל שליח סטטיסטיקות של הזמנות שטיפול בהן וכן את ההזמנה שבטיפולו כרגע (אם קיימת)
- יוכל לנהל את מאגר ההזמנות ולצפות ברשימת ההזמנות (לסנן, למיין, לקבץ לפי קטגוריות)
 - עבור כל הזמנה יוכל לראות גם את היסטוריית המשלוחים של (משלוח ראשון המזמין לא היה נמצא והיא נפתחה מחדש להמתנה למשלוח)
- יוכל לנהל את שעות המערכת - לאתחל את השעון, לקדם את השעון בדקות, בשעות, בימים. על מנת לאפשר הדמיה של התקדמות בזמן,
- מנהל יוכל להגדיר משתני תצורה כלליים של המערכת

משתמש שליח

- יוכל לצפות/לעדכן את פרטיו האישיים כולל המרחק האווירי המירבי האישי שעד אליו הוא מעוניין לבצע משלוחים
- יוכל לצפות (בלבד) בכל ההזמנות הפתוחות (בלבד), אשר נמצאות במרחק אווירי קטן/שווה למרחק האווירי האישי שהגדיר לעצמו
- מבין ההזמנות הפתוחות, יוכל לבחור הזמנה שבה הוא רוצה לטפל והיא תהיה מוקצת עבורו בלבד ותעבור למצב "בטיפול" על ידו.
- יכול לטפל בו זמנית בהזמנה אחת בלבד
- יוכל לצפות בפרטי ההזמנה הנוכחית שמשייכת אליו, לעדכן שסיים לטפל בה (להעבירה למצב "סופקה" או כל סוג סיום משלוח אחר)
- יוכל לצפות בהיסטוריית המשלוחים שלו בלבד - לסנן למיין, לקבץ לפי קטגוריות

אפיון המסכים - תצוגת ממשק המשתמש

פירוט המסכים בראשי פרקים

פירוט מעמיק יותר על כל אחד מהמסכים ניתן למצוא בהמשך מסמך זה: [פירוט המסכים והפונקציונאליות שלהם](#)

כעת, נפרט את רשימת המסכים ונסביר בקצרה מה מטרת כל אחד מהם:

- מסך כניסה למערכת - מאפשר כניסה למערכת עבור "מנהל" ועבור "שליח"
- מסכים עבור משתמש "מנהל":
 - מסך ניהול ראשי - משמש את ה"מנהל" לצורך ניהול המערכת כולה
 - מסך ניהול שליחים - משמש לצורך ניהול מאגר השליחים
 - מסך הוספת שליח - משמש לצורך הוספה של שליח חדש
 - מסך ניהול שליח בודד - משמש לצורך צפיה/עדכון/מחיקה בפרטים של שליח קיים
 - מסך ניהול הזמנות - משמש לצורך ניהול מאגר של כל ההזמנות מכל הסטטוסים
 - מסך הוספת הזמנה - משמש לצורך הוספה של הזמנה חדשה
 - מסך ניהול הזמנה בודדת - משמש לצורך צפיה/עדכון פרטים של הזמנה קיימת
- מסכים עבור משתמש "שליח":
 - מסך שליח - משמש לצורך צפיה/עדכון של פרטיו של השליח עצמו כולל ההזמנה שבטיפולו (אם קיימת)
 - מסך בחירת הזמנה לטיפול - משמש לצורך תצוגת הזמנות פתוחות על מנת לבחור מהן הזמנה לטיפול
 - מסך היסטוריית הזמנות - משמש לצורך תצוגת היסטוריית ההזמנות הסגורות שטופלו על ידי השליח

תרשימים המסכים

לפניכם דוגמה של המסכים/החלונות שיופיעו בתצוגה. ממשק המשתמש במערכת. הסטודנטים יהיו רשאים לעשות שינויים בהרכב ותסריטי זרימה בין המסכים/החלונות ע"פ הבנתם של תפקוד האפליקציה וע"פ הגיון בריא. הסטודנטים יצטרכו להסביר את החלטותיהם ואת ההיגיון שלהם בשלב ההגנה של הקורס.

התנהלות כללית של המערכת

סוגי שילוח

סוג השילוח = הדרך בה יבוצעו המשלוחים של אותו שליח.

כל עוד השליח לא מבצע כרגע משלוח (מטפל בהזמנה), אזי השליח/המנהל יוכל לבחור ולשנות את "סוג השילוח". אם יש כרגע משלוח בטיפול של אותו שליח אזי לא ניתן לשנות את "סוג השילוח".

סוג השילוח ישמר בבסיס הנתונים ב- ישות נתונים שליח - DO.Courier וב- ישות נתונים משלוח - DO.Delivery ויוצג גם בתצוגה.

סוגי שילוח:

- רכב
- אופנוע
- אופניים
- רגלי

בישות התצורה תהיינה תכונות של מהירות ממוצעת עבור כל סוג שילוח.

חישובי הזמנים והמרחקים יקבעו על פי סוג השילוח. נפרט מיד.

חישוב מרחק הזמנה בפועל

מרחק ההזמנה מהחברה בפועל יחושב בזמן יצירת ישות נתונים משלוח - DO.Delivery וישמר בבסיס הנתונים. יוצג גם בתצוגה עבור משלוח כלשהו.

חישוב המרחק בפועל בין כתובת החברה לכתובת ההזמנה שהשליח בוחר לבצע יושפע מסוג השילוח:

- מרחק (מסלול) נסיעה ברכב ומרחק רכיבה על אופנוע יחושבו כאותו מרחק - ע"פ ההנחה שהרכב והאופנוע נוהגים באותו מסלול נסיעה.
- מרחק (מסלול) רכיבה באופניים ומרחק של הליכה רגלית יחושבו כאותו מרחק - ע"פ ההנחה שהאופניים והולך הרגל עושים את אותו מסלול רכיבה/הליכה.

כתובת מבוססת על קואורדינטות (קו אורך וקו רוחב בכדור הארץ) וחישוב מרחק (נסיעה/הליכה) בין כתובות מבוסס על קואורדינטות גם הוא. ראה פירוט כאן: [נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות](#)

הערה חשובה:

אנחנו מבדילים בין **חישוב מרחק הזמנה בפועל** שהוא חישוב המסלול האמיתי ומושפע מסוג השילוח (הליכה/נסיעה) לבין **חישוב מרחק אווירי** - מרחק בקו אווירי שאינו מושפע מסוג השילוח וניתן לחישוב מתמטי פשוט בהינתן 2 זוגות של קואורדינטות (ניתן למצוא אותו ברשת בקלות).

סוגי סיום משלוח

נסכם את סוגי "סיום המשלוח" בתוך ישות נתונים משלוח - DO.Delivery על ידי השליח הנוכחי:

- **סופק** - ההזמנה סופקה (בלי קשר לזמן שבו היא סופקה) **ונסגרה**.
- **מזמין סירב לקבל** - השליח הגיע ליעד, אך המזמין מסרב לקבל את ההזמנה (לא מרוצה וכדומה). ההזמנה חוזרת לחברה **ונסגרת**.
- **בוטל** - לאחר שנפתחה ההזמנה המזמין התקשר והתחרט וההזמנה **נסגרת ע"י המנהל**, יש שני מקרים:
 - a. ההזמנה כבר נמצאת בטיפול, כלומר באמצע משלוח, (המנהל יצור קשר עם השליח שיחזור) - ההזמנה **תיסגר** וזמן סיום המשלוח יתעדכן לזמן הנוכחי וסוג סיום משלוח יהיה **בוטל**
 - b. ההזמנה עדיין **פתוחה**, כלומר לא נמצאת באמצע משלוח - במקרה כזה ההזמנה **תיסגר** ויתווסף **משלוח "מדומה"** עם זמן תחילה וסיום זהה, סוג סיום משלוח יהיה **בוטל** ות"ז של השליח יהיה **אפס**
- **מזמין לא נמצא** - השליח הגיע ליעד, אך המזמין לא בבית וגם לא ענה לטלפון. ההזמנה חוזרת לחברה. המשלוח הנוכחי נסגר וההזמנה **נפתחת מחדש** לאיסוף על ידי שליח.
- **נכשל** - בזמן ששליח בוחר לטפל בהזמנה (נוצרת ישות משלוח), התרחשה תקלה בעת חישוב המרחק (המסלול) בין כתובת החברה לכתובת ההזמנה ע"פ סוג השילוח. המשלוח ייסגר וההזמנה **נשארת פתוחה**.

זמנים בחיי הזמנה

נסכם זמנים בחיי הזמנה, חלקם שמורים בבסיס הנתונים כתכונות של ישויות נתונים (DO) וחלקם מחושבים בשכבה הלוגית ומוצגים בתצוגה כתכונות של ישויות נתונים לוגיות (BO):

- **זמן פתיחת ההזמנה** - זמן (תאריך ושעה) שבו נפתחה ההזמנה על ידי המנהל. ישמר בבסיס הנתונים ב- **ישות נתונים הזמנה - DO.Order**. יקבע בזמן שישות ההזמנה נוצרת, על פי הזמן הנוכחי של שעון המערכת (תכונה בישות התצורה).
- **זמן תחילת משלוח** - זמן (תאריך ושעה) שבו ההזמנה הנוכחית נאספה על ידי השליח מהחברה. ישמר בבסיס הנתונים בתוך **ישות נתונים משלוח - DO.Delivery**. יקבע בזמן שישות המשלוח נוצרת, על פי הזמן הנוכחי של שעון המערכת (תכונה בישות התצורה).
- **זמן אספקה צפוי** - זמן (תאריך ושעה) הצפוי לקבלת המשלוח על פי **זמן תחילת המשלוח הנוכחי** ו**בהתחשב בסוג השילוח** (מרחק ההזמנה ומהירות ממוצעת של השליח). יחושב בשכבה הלוגית.
- **זמן סיום משלוח** - זמן (תאריך ושעה) שבו הסתיים המשלוח שגרם לסגירת ההזמנה או לפתיחתה מחדש (ראה פירוט סוגי סיום משלוח).
- **זמן אספקה מירבי** - זמן (תאריך ושעה) אספקה מירבי בהתבסס על **טווח האספקה המירבי** שהחברה התחייבה אליו ועל הזמן שבו ההזמנה נפתחה על ידי המנהל (**בלי להתחשב בסוג השילוח**). יחושב בשכבה הלוגית.

סטטוס הזמנה (OrderStatus)

מבטא את סטטוס ההזמנה **בהיבט של מצב המשלוח האחרון**. יחושב בשכבה הלוגית על סמך הנתונים הקיימים בישויות **ישות נתונים הזמנה - DO.Order** ו- **ישות נתונים משלוח - DO.Delivery**.

סטטוס של הזמנה יחושב וישמר כתכונה בחלק מישויות הנתונים הלוגיות:

- **פתוחה** - לא נמצאת בטיפול של אף שליח כרגע ועוד לא נסגרה מסיבה כלשהי
- **בטיפול** - בטיפול כרגע על ידי שליח
- **סופקה** - ההזמנה **סגורה**, נמסרה למזמין, סוג סיום המשלוח **האחרון** "סופק"
- **מזמין סירב לקבל** - ההזמנה **סגורה**, סוג סיום המשלוח **האחרון** "מזמין סירב לקבל"
- **בוטלה** - ההזמנה **סגורה**, סוג סיום המשלוח **האחרון** "בוטל"

סטטוס עמידה בזמנים (ScheduleStatus)

מבטא את סטטוס ההזמנה **בהיבט של עמידה בזמנים של המשלוח האחרון**. יחושב בשכבה הלוגית על סמך הנתונים הקיימים בישויות **ישות נתונים הזמנה - DO.Order** ו- **ישות נתונים משלוח - DO.Delivery**. כמו כן, **טווח זמן סיכון** ו**טווח זמן אספקה מירבי** למשלוח הם **משתני תצורה** המוגדרים על ידי המנהל ומשפיעים על קביעת סטטוס של הזמנה.

סטטוס עמידה בזמנים יחושב וישמר כתכונה בחלק מישויות הנתונים הלוגיות:

- **בזמן (OnTime)**
 - ההזמנה פתוחה או בטיפול ונשאר עוד מספיק זמן לבצע את המשלוח ולספק את ההזמנה תוך פרק זמן שהחברה התחייבה (**זמן אספקה המירבי** בלי סיכון)
 - ההזמנה סגורה ו**זמן סיום משלוח** הינו עד **זמן אספקה מירבי** (ועד בכלל)
- **בסיכון (InRisk)**
 - ההזמנה פתוחה או בטיפול ונשאר לה פחות מ**טווח זמן הסיכון** עד **זמן אספקה מירבי**
- **באחור (Late)**
 - ההזמנה פתוחה או בטיפול ו**זמן אספקה מירבי** שלה עבר ועדיין לא נסגרה
 - ההזמנה סגורה ו**זמן סיום משלוח** הינו אחרי **זמן אספקה מירבי**

חישוב שכר של שליח

תוספת

שיטת חישוב שכר השליח תוצע ע"י זוג הסטודנטים לפי אחת השיטות הקיימות בשוק (מומלץ להיעזר בשאלתא לבינה מלאכותית ולבחור מבין השיטות, יש להתמקד בשיטות המקובלות בארץ ובצורת העסקת השליחים בחברה שהסטודנטים בחרו בה).

ניתן להרחיב את הנושא לתלושי משכורת, חישוב הוצאות למנהל כיד הדמיון הטובה עליכם.

הסטודנטים יוסיפו תכונות מתאימות בתצורה ובישויות אחרות כנדרש ע"י השיטה הספציפית. לאחר מכן, השכר יוצג במסכים השונים עבור שליח ועבור מנהל - לפי חודשים, לפי שנים וכו'.

תסריטים אפשריים לדוגמא

נתאר כעת תסריט אפשרי להתנהלות במערכת כדי לאפשר לכם להבין יותר את הפונקציונליות שהיא מאפשרת.

מנהל מגדיר את נתוני התצורה עבור המערכת כולה, בין השאר:

- כתובת של החברה
- מרחק מירבי למשלוח
- מהירות נסיעה ממוצעת לכל סוג שילוח
- טווח זמן אספקה מירבי
- טווח זמן סיכון

בכל רגע נתון, מנהל יכול להוסיף שליח למערכת:

- פרטי השליח ישמרו עם מספר ת"ז ופרטים נוספים וכן סוג השילוח, זמן תחילת עבודה בחברה ומרחק מירבי לקבלת משלוח

בכל פעם שלקוח מתקשר לחברה ומזמין, המנהל פותח הזמנה:

- פרטי ההזמנה ישמרו עם מספר מזהה הזמנה רץ, כתובת של ההזמנה ועם זמן פתיחה של ההזמנה
- ההזמנה נחשבת בסטטוס פתוחה - כל עוד לא נסגרה בדרך כזו או אחרת (למשל: סופקה, בוטלה ועוד)
- עבור כל הזמנה יחושבו גם זמן אספקה צפוי וזמן אספקה מירבי אך הם לא ישמרו
- המנהל יוכל לצפות בכל ההזמנות בכל הסטטוסים
- שליח יוכל לבחור את ההזמנה הפתוחה לביצוע משלוח
 - יוכל לצפות רק בהזמנות עם סטטוס פתוחה ואשר נמצאות במרחק שהוא פחות ממרחק מירבי אישי למשלוח שהוגדר על ידו

ברגע ששליח בוחר לבצע משלוח, כלומר לטפל בהזמנה פתוחה:

- פרטי המשלוח ישמרו עם ת"ז של השליח והמספר המזהה של ההזמנה, כאשר זמן תחילת המשלוח יהיה הזמן הנוכחי, זמן סיום המשלוח שזה עתה נפתח וסוג סיום המשלוח יהיו לעת עתה null
- ההזמנה נחשבת בסטטוס בטיפול - אף שליח אחר לא יכול לבחור בה והשליח המטפל בה לא יכול לבחור הזמנה נוספת לטיפול
- כל עוד השליח מטפל בהזמנה היא תוצג כחלק מהנתונים הכוללים של השליח

שליח יכול לדווח שסיים לבצע את המשלוח וההזמנה והיא סופקה בהצלחה:

- זמן סיום משלוח יתעדכן לפי שעון המערכת וסוג סיום משלוח יתעדכן לערך סופק
- ההזמנה נחשבת בסטטוס סופקה על פי הזמן שבו סופקה ביחס לזמן המירבי שחושב לה.
- שליח יוכל לראות אותה בהיסטוריית המשלוחים שלו ומנהל יוכל לראות אותה במסך ניהול הזמנות

שליח יכול לדווח שסיים לבצע משלוח וההזמנה לא סופקה כי המזמין לא נמצא:

- פרטי המשלוח יתעדכנו עם זמן סיום משלוח לפי שעון המערכת וסוג סיום משלוח יתעדכן לערך מזמין לא נמצא
- ההזמנה חוזרת לחברה, תהיה שוב בסטטוס פתוחה ותוכל להיבחר שוב על ידי שליח כלשהו לביצוע משלוח
- שליח יוכל לראות אותה בהיסטוריית המשלוחים שלו ומנהל יוכל לראות אותה במסך ניהול הזמנות

במקרה שלקוח מתקשר לבטל הזמנה שכבר בטיפול ונמצאת באמצע משלוח:

- המנהל יצור קשר עם השליח שיחזור
- ההזמנה נסגרת ונחשבת בסטטוס בוטלה
- זמן סיום המשלוח יתעדכן לזמן הנוכחי וסוג סיום משלוח יהיה בוטל
- שליח יוכל לראות אותה בהיסטוריית המשלוחים שלו ומנהל יוכל לראות אותה במסך ניהול הזמנות

במקרה שלקוח מתקשר לבטל הזמנה שעדיין פתוחה ולא נמצאת באמצע משלוח:

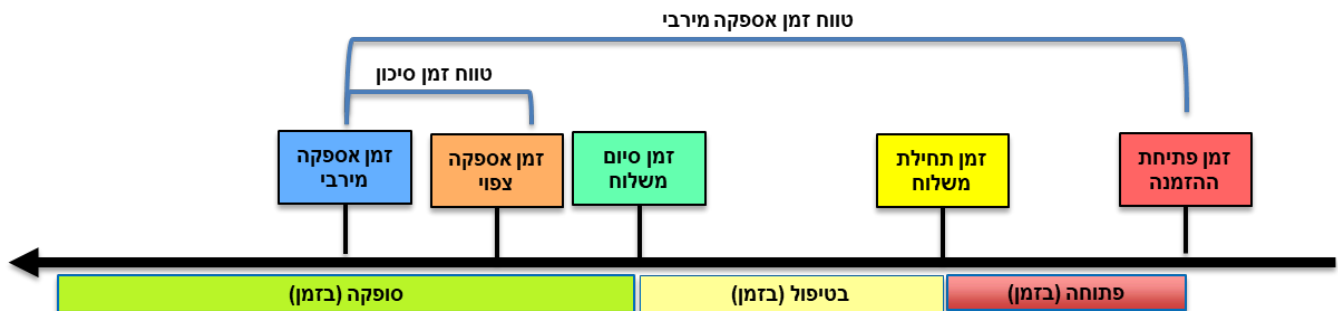
- ההזמנה נסגרת ונחשבת בסטטוס בוטלה
- יתווסף "משלוח מדומה" עם זמן תחילה וסיום זהה, סוג סיום משלוח יהיה בוטל ות"ז של השליח יהיה אפס
- מנהל יוכל לראות את ההזמנה שבוטלה במסך ניהול הזמנות

תרשימים להמחשה - תרחישים חיי הזמנה

נמחיש בעזרת תרשימים חלק מהתרחישים האפשריים עבור הזמנה.

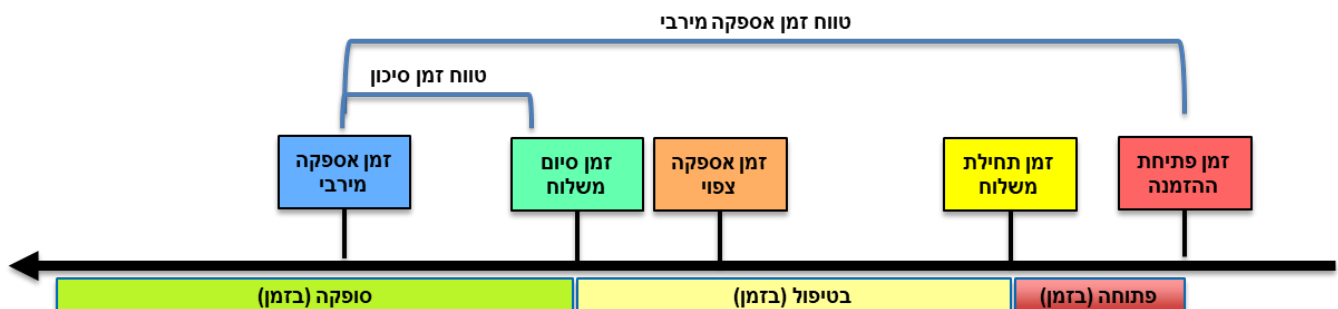
תרחיש 1:

המשלוח הסתיים וההזמנה נסגרה לפני זמן האספקה הצפוי, לפני כניסה לטווח זמן סיכון (בטיפול בזמן) ולפני זמן האספקה המירבי (סופקה בזמן).



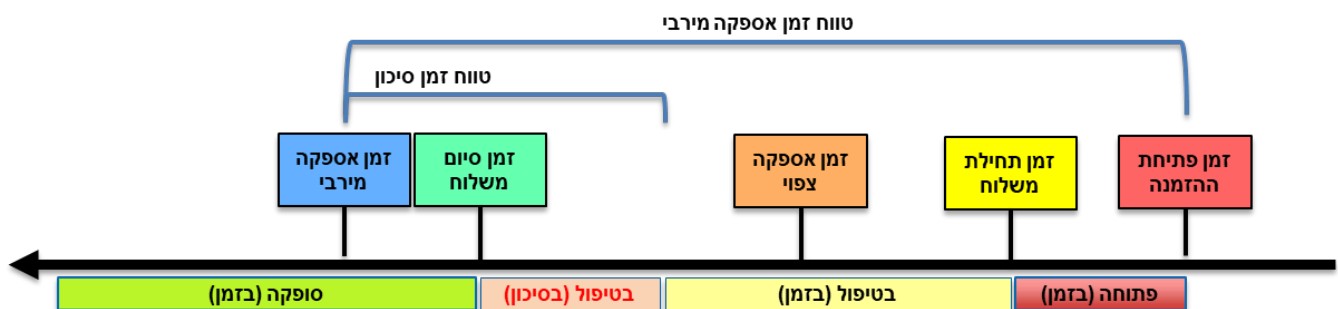
תרחיש 2:

המשלוח הסתיים וההזמנה נסגרה אחרי זמן האספקה הצפוי, לפני כניסה לטווח זמן סיכון (בטיפול בזמן) ולפני זמן האספקה המירבי (סופקה בזמן).



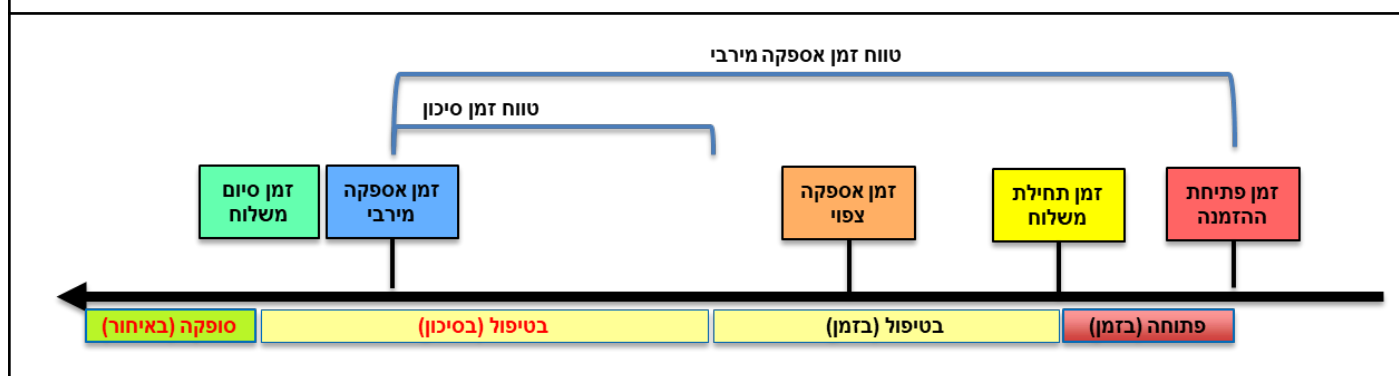
תרחיש 3:

המשלוח הסתיים וההזמנה נסגרה אחרי זמן האספקה הצפוי, אחרי כניסה לטווח זמן סיכון (בטיפול בסיכון) ולפני זמן האספקה המירבי (סופקה בזמן).



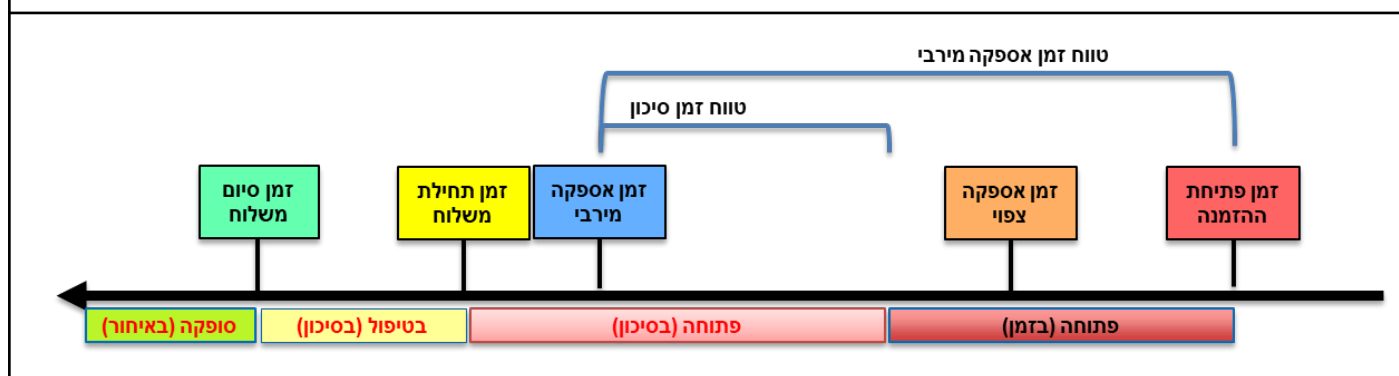
תרחיש 4:

המשלוח הסתיים וההזמנה נסגרה אחרי זמן האספקה הצפוי והמירבי (סופקה באיחור) , אחרי כניסה לטווח זמן סיכון (בטיפול בסיכון).



תרחיש 5:

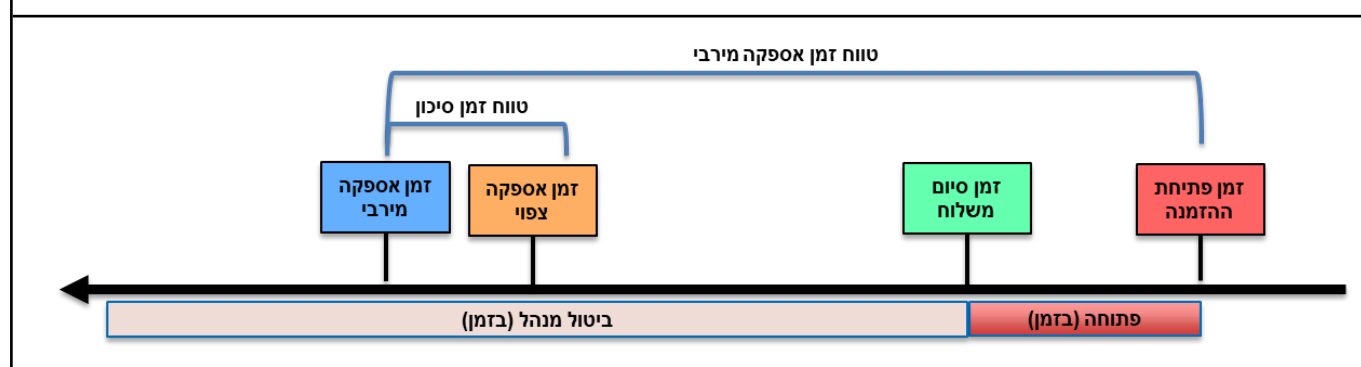
ההזמנה נכנסה לטווח זמן סיכון (פתוחה בסיכון) עוד לפני שנכנסה לטיפול. נכנסה לטיפול אחרי זמן אספקה מירבי (בטיפול בסיכון) וגם נסגרה אחרי זמן האספקה הצפוי והמירבי (סופקה באיחור). המשלוח הסתיים.



תרחיש 6:

המשלוח הסתיים וההזמנה נסגרה לפני זמן האספקה הצפוי, לפני כניסה לטווח זמן סיכון (בטיפול בזמן) ולפני זמן האספקה המירבי (סופקה בזמן).

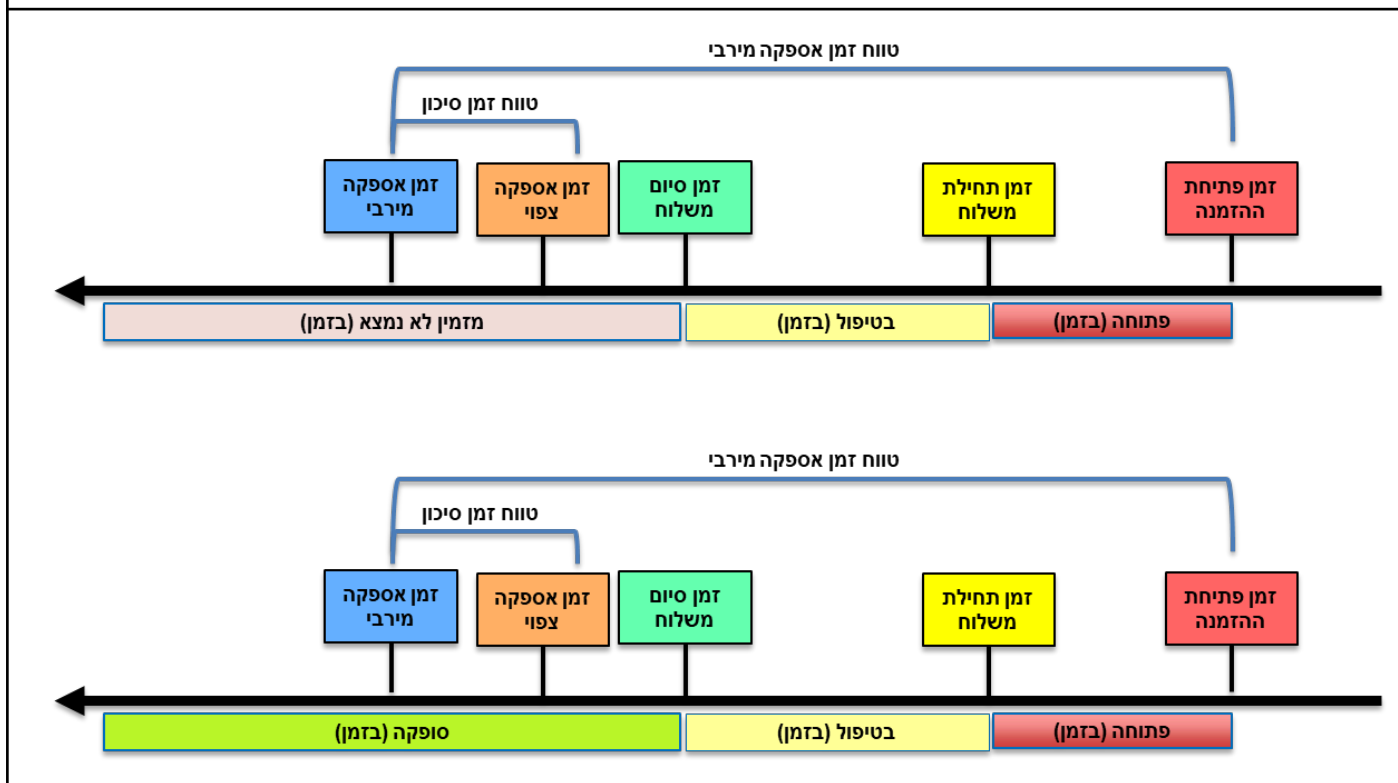
סיבת הסגירה - בוטל (ע"י מנהל) - נסגרה לפני שנכנסה לטיפול - לקוח התקשר לבטל. ההזמנה חוזרת לחברה ונסגרת לתמיד.



תרחיש 7:

ההזמנה נסגרה לפני זמן האספקה הצפוי, לפני כניסה לטווח זמן סיכון (בטיפול בזמן) ולפני זמן האספקה המירבי (סופקה בזמן).

סיבת הסגירה - מזמין לא נמצא. המשלוח הסתיים אך ההזמנה חוזרת לחברה ונפתחת מחדש.



שכבת הנתונים DAL (שלב 1 בפרויקט)

תיאור כללי

מטרת שכבת הנתונים לאפשר שמירה ושליפה של נתוני המערכת בצורה יעילה ושטוחה ללא פעילות לוגית או חישובית. בשכבת הנתונים (DAL) נגדיר:

- **ישות תצורה** - משתני סביבה משותפים
 - [ישות תצורה Config](#)
- **ישויות נתונים** - ישויות DO. הנשמרות בבסיס נתונים.
 - [ישות נתונים שליח - DO.Courier](#)
 - [ישות נתונים הזמנה - DO.Order](#)
 - [ישות נתונים משלוח - DO.Delivery](#)
- **חוזי נתונים** - (ממשק כנגד כל ישות נתונים) שיכילו מתודות לצורך הוספה/צפייה/עדכון/מחיקה של כל ישות מבסיס נתונים.

לאחר מכן, נגדיר את בסיס הנתונים בזיכרון - הגדרת רשימות שיכילו את האובייקטים של כל סוגי ישויות הנתונים. נאפשר לאפס את נתוני התצורה ואת רשימות נתונים - להחזיר את ערכי התצורה לערכי ברירת מחדל ולרוקן את הרשימות מאובייקטים.

נממש מתודה שמייצרת נתונים התחלתיים לבסיס נתונים - מייצרת רשימות עם נתונים תקינים והגיוניים כדי לייצר התחלה של מערכת פעילה בבת אחת.

בנוסף, נבנה תוכנית בדיקה ידנית (manual test program) של שכבת הנתונים (DalTest):

- אתחול ראשוני של בסיס הנתונים
- תוכנית ראשית - ממשק קונסול טקסטואלי (CLI) לבדיקת שכבת הנתונים.

ישות תצורה (קונפיגורציה)

נגדיר ישות תצורה, שתכיל משתני סביבה המשותפים לכל המערכת.

ישות תצורה Config

ישות התצורה (קונפיגורציה, configuration, או בקיצור config) הינה ישות סטטית. כלומר, לא ייווצרו הופעות/אובייקטים עבור יישות זו. אפשר להתייחס לתכונות של ישות זו כמעין "משתני מערכת".

הערה: העמודה "האם ניתן לעדכון" לא רלוונטית עדיין בשלב ה DAL אלא רק בשלבים שלאחר מכן.

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף
מספר מזהה הזמנה רץ NextOrderId	int	לא	לא	מייצג מספר מזהה להזמנה החדשה הבאה. רץ אוטומטית (מתקדם ב-1 בשליפה של המספר, בעת הוספת הזמנה חדשה)
מספר מזהה משלוח NextDeliveryId	int	לא	לא	מייצג מספר מזהה למופע חדש של ישות המשלוח המקשרת בין שליח להזמנה. ממנו יילקח המספר המזהה הבא עבור ישות המשלוח. רץ אוטומטית (מתקדם ב-1 בשליפה של המספר, בעת הוספת ישות משלוח חדשה)
שעון המערכת Clock	DateTime	לא	לא	מכיוון שהמערכת שלנו עושה הדמיה לניהול שליחים, נזדקק לשעון מערכת שיתוחזק בנפרד מהשעון האמיתי של המחשב. מנהל המערכת יוכל לאתחל ולעדכן (לקדם) את שעון המערכת. בכל פעולה, הזמן הנוכחי של המערכת יהיה הערך הנוכחי של שעון המערכת.
ת"ז של מנהל	int	לא	לא	מייצג ת"ז ותקנית המזהה את המנהל. ערך מספרי עם ספרת ביקורת תקינה.
סיסמה של מנהל	string	לא	לא	תוספת - באתחול תינתן סיסמה ראשונית ולאחר מכן הוא יוכל לעדכן אותה. תוספת - בדיקה שהסיסמא חזקה. תוספת - הצפנת סיסמא

תכונה זו לא תהיה קיימת עבור מי שלא עושה את התוספת הזו.				
כתובת מלאה של החברה	string	כן	ע"י מנהל	כתובת מלאה ואמיתית בפורמט תקין, של החברה. מכאן כל השליחים יוצאים כדי לבצע את המשלוחים ולכאן חוזרים כל השליחים בסיום כל משלוח. <u>דוגמא לכתובות:</u> הנשיאים 7, פתח תקוה, ישראל המרכז האקדמי לב, ירושלים, ישראל השכבה הלוגית תבדוק האם הכתובת אכן קיימת במציאות ותחשב את קו הרוחב שלה (latitude) ואת קו האורך שלה (longitude) בכדור הארץ. והמידע הזה יישמר גם הוא בתכונות התצורה. הכתובת תאותחל לערך null. במידה ובזמן עדכון הכתובת תתגלה כלא תקינה, לא ישתנה הערך הקודם שלה בשכבת הנתונים. כל עוד ערך הכתובת הוא null (אין כתובת תקינה) אזי לא ניתן לפתוח הזמנות. בשלב 1 הכתובת תזון דרך הקוד בצורה מפורשת ותקונה. בשלב 4 הבדיקה תתבצע דרך הקוד בצורה סינכרונית ובשלב 7 של הפרויקט החישוב יתבצע בצורה אסינכרונית.
קו רוחב Latitude	double	כן	מתעדכן אוטומטית ע"י השכבה הלוגית כתובת החברה.	קו רוחב - מספר המציין עד כמה נקודה על פני כדור הארץ נמצאת מדרום או מצפון לקו המשווה. קו אורך - מספר המציין עד כמה נקודה על פני כדור הארץ נמצאת ממזרח או ממערב לקו המשווה. כל עוד הכתובת לא תקינה אז קו הרוחב וקו האורך ישארו null. תכונות אלו נועדו לצורך חישוב מרחקים בין כתובות, ולא אמורות להיות מוצגות בסופו של דבר בתצוגה. ערך התכונה יתעדכן על ידי השכבה הלוגית בכל פעם שהכתובת של החברה תעודכן. קיימות ברשת פונקציות שמחשבות אותו. בשלב 1 הערך יזון בצורה מפורשת דרך הקוד. בשלב 4 חישוב הקואורדינטות יעשה בצורה סינכרונית ובשלב 7 בצורה אסינכרונית. ראה פירוט כאן: נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות
קו אורך Longitude	double	כן		
מרחק מירבי כללי למשלוח בק"מ (אווירי)	double	כן	ע"י מנהל	המרחק האווירי המירבי בין כתובת החברה לכתובת ההזמנה. יתקבלו רק הזמנות שנמצאות בטווח הזה. אם ערך התכונה הוא null המשמעות היא שלחברה אין הגבלת מרחק להזמנות. קיימת ברשת פונקציה מתמטית פשוטה לחישוב מרחק אווירי של 2 כתובות בהינתן קואורדינטות של 2 הכתובות.
מהירות נסיעה ממוצעת ברכב בקמ"ש	double	לא	ע"י מנהל	מהירות נסיעה ממוצעת ברכב בקמ"ש. לצורך חישוב זמני משלוח ומרחקים בפועל של הזמנה מהחברה.
מהירות רכיבה ממוצעת באופנוע בקמ"ש	double	לא	ע"י מנהל	מהירות נסיעה ממוצעת באופנוע בקמ"ש. לצורך חישוב זמני משלוח ומרחקים בפועל של הזמנה מהחברה.
מהירות רכיבה באופניים ממוצעת בקמ"ש	double	לא	ע"י מנהל	מהירות נסיעה ממוצעת באופניים בקמ"ש. לצורך חישוב זמני משלוח ומרחקים בפועל של הזמנה מהחברה.
מהירות הליכה ממוצעת בקמ"ש	double	לא	ע"י מנהל	מהירות הליכה ממוצעת בקמ"ש. לצורך חישוב זמני משלוח ומרחקים בפועל של הזמנה מהחברה.
טווח זמן אספקה מירבי	TimeSpan	לא	ע"י מנהל	התחייבות החברה לזמן הגעה של כל ההזמנות ללקוחות. טווח זמן אשר יעזור לנו לחשב אם הזמנה נמצאת בסיכון. יעזור לנו לדעת אילו הזמנות סופקו בזמן ואילו לא. ועוד. יחידות הזמן (שעות/ימים) יקבעו לפי סוג החברה.
טווח זמן סיכון RiskRange	TimeSpan	לא	ע"י מנהל	טווח זמן אשר ממנו והלאה הזמנה נחשבת בסיכון. מתקרבת לזמן האספקה המירבי הנדרש למשלוח ועדיין לא סופקה. יחידות הזמן (שעות/ימים) יקבעו לפי סוג החברה.

טווח זמן חוסר פעילות	TimeSpan	לא	ע"י מנהל	טווח זמן אשר לא נעשתה בו פעילות של שליח וממנו והלאה הוא יוגדר אוטומטית כ'לא פעיל'. לאורך טווח הזמן הזה השליח לא ביצע אף משלוח. יחידות הזמן (שעות/ימים) יקבעו לפי סוג החברה.
----------------------	----------	----	----------	---

ישויות נתונים - DO

לצורכי המערכת נגדיר 3 ישויות נתונים:

- ישויות נתונים שליח - [DO.Courier](#) - תכיל פרטים עבור שליח, כולל ת"ז ייחודית
- ישויות נתונים הזמנה - [DO.Order](#) - תכיל פרטים עבור הזמנה שפתח מנהל ותטופל על ידי שליח, כולל מספר מזהה רץ של ההזמנה
- ישויות נתונים משלוח - [DO.Delivery](#) - מתעדת הקצאה של הזמנה לשליח שבחר לטפל בה. בין השאר, היא תכיל ת"ז של השליח ואת המספר הרץ של ההזמנה. ובה גם יתועד סוג זמן סיום המשלוח.

לכל ישויות נגדיר רשימה של תכונות מטיפוסים שונים:

- חשוב - בשכבת נתונים אנו מניחים שערכי התכונות תקינים. אין צורך לבדוק את תקינות הערכים עבור ישויות DO בשכבת נתונים DAL. את תקינות הערכים נבדוק בהמשך, במימוש השכבה הלוגית BL.
- הסטודנטים יכולים להוסיף תכונות לישויות עבור **תוספות**, לפי צרכי המערכת הספציפית שלהם

הערה: העמודה "האם ניתן לעדכון" לא רלוונטית עדיין בשלב ה DAL אלא רק בשלבים שלאחר מכן.

ישויות נתונים שליח - DO.Courier

ישויות השליח מכילה פרטים אישיים עבור "שליח", כולל ת"ז ייחודית.

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף תקניות הערך תיבדק במימוש השכבה הלוגית. בשכבת הנתונים יש להניח שהערך תקין.
ת"ז שליח id	int	לא	לא	מייצג ת"ז ותקנית המזהה את השליח באופן ייחודי. ערך מספרי עם ספרות ביקורת תקינה.
שם מלא (פרטי ומשפחה)	string	לא	ע"י מנהל או שליח	שם מלא של השליח
טלפון סלולרי	string	לא	ע"י מנהל או שליח	מייצג טלפון סלולרי תקני של השליח. מחרוזת בת 10 ספרות בדיוק המתחילה בספרה 0.
אימייל	string	לא	ע"י מנהל או שליח	מייצג כתובת דואר אלקטרוני תקינה מבחינת פורמט. ראו ברשת כיצד בודקים זאת (בשכבה הלוגית).
סיסמה	string	לא	ע"י מנהל או שליח	תוספת - בהתחלה תינתן סיסמה ראשונית ע"י המנהל ולאחר מכן השליח יוכל לעדכן אותה. תוספת - בדיקה שהסיסמה חזקה. תוספת - הצפנת סיסמה תכונה זו לא תהיה קיימת עבור מי שלא עושה את התוספת הזו.
פעיל	boolean	לא	ע"י מנהל בלבד	האם השליח פעיל או לא פעיל (פרש מהחברה). שליח שפרש נשמרת עבורו גם היסטוריית המשלוחים שלו אך הוא לא יכול לטפל בהזמנות חדשות.
מרחק מירבי אישי למשלוח	double	כן	ע"י מנהל או שליח	המרחק האווירי המירבי האישי בין כתובת החברה לכתובת ההזמנה שעד אליו השליח מעוניין להגיע. צריך להיות קטן שווה למרחק האווירי המירבי הכללי למשלוח של החברה. השכבה הלוגית, תאפשר לשליח לבחור הזמנה לביצוע מבין כל ההזמנות שנמצאות בטווח הזה. אם ערך התכונה הוא null המשמעות היא שלשליח אין הגבלת מרחק להזמנות. קיימת ברשת פונקציה מתמטית פשוטה לחישוב מרחק אווירי של 2 כתובות בהינתן קואורדינטות של 2 הכתובות.
סוג השילוח	ENUM	לא	ע"י מנהל או שליח	שילוח מסוג רכב/אופנוע/אופניים/רגלי. ראה פירוט של סוגי השילוח והשפעתם על חישובי מרחק ומהירות של משלוח, כאן: סוגי שילוח

ניתן לעדכון כל עוד השליח לא מטפל כרגע בהזמנה.				
הזמן (תאריך ושעה) שבו השליח התחיל לעבוד בחברה. זהו בעצם הזמן שבו נוצרה ישות השליח הנוכחי. ייקבע על פי הזמן הנוכחי של שעון המערכת, שנמצא בישות התצורה.	לא	לא	DateTime	זמן תחילת עבודה בחברה

ישות נתונים הזמנה - DO.Order

ישות ההזמנה מכילה פרטים עבור "הזמנה", כולל מספר מזהה רץ ייחודי.

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף תקינות הערך תיבדק במימוש השכבה הלוגית. בשכבת הנתונים יש להניח שהערך תקין.
מספר הזמנה (מספר מזהה רץ של ישות ההזמנה) Id	int	לא	לא	מייצג מספר המזהה את ההזמנה באופן ייחודי. כשהזמנה נפתחת, ניגש לישות התצורה כדי לקבל עבורה את מספר ההזמנה הרץ הבא.
סוג הזמנה	ENUM	לא	ע"י מנהל	סוגים שונים של הזמנות בהתאם לסוג החברה
תיאור מילולי	string	כן	ע"י מנהל	תיאור מילולי קצר של תוכן ההזמנה.
כתובת מלאה של ההזמנה	string	לא	ע"י מנהל	כתובת מלאה ואמיתית בפורמט תקין, של מיקום ההזמנה. דוגמא לכתובת: הנשיאים 7, פתח תקוה, ישראל המרכז האקדמי לב, ירושלים, ישראל השכבה הלוגית תבדוק האם הכתובת אכן קיימת במציאות ותחשב את קו הרוחב שלה (latitude) ואת קו האורך שלה (longitude) בכדור הארץ. והמידע הזה יישמר גם הוא בתכונות ההזמנה. בשכבת הנתונים יכולה להיווצר רק הזמנה עם כתובת תקינה כולל קו רוחב וקו אורך תקינים. בשלב 1 הכתובת תוזן דרך הקוד בצורה מפורשת ותקונה. בשלב 4 הבדיקה תתבצע דרך הקוד בצורה סינכרונית ובשלב 7 של הפרויקט החישוב יתבצע בצורה אסינכרונית.
קו רוחב Latitude	double	לא	מתעדכן אוטומטית ע"י השכבה הלוגית בעקבות עדכון כתובת ההזמנה	קו רוחב - מספר המציין עד כמה נקודה על פני כדור הארץ נמצאת מדרום או מצפון לקו המשווה. קו אורך - מספר המציין עד כמה נקודה על פני כדור הארץ נמצאת ממזרח או ממערב לקו המשווה. תכונות אלו נועדו לצורך חישוב מרחקים בין כתובות, ולא אמורות להיות מוצגת בסופו של דבר בתצוגה. ערך התכונה יתעדכן על ידי השכבה הלוגית בכל פעם שהכתובת של ההזמנה תעודכן. קיימות ברשת פונקציות שמחשבות אותו. בשלב 1 הערך יוזן בצורה מפורשת דרך הקוד. בשלב 4 חישוב הקואורדינטות יעשה בצורה סינכרונית ובשלב 7 בצורה אסינכרונית. ראה פירוט כאן: נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות
קו אורך Longitude	double	לא		
שם מלא של המזמין	string	לא	ע"י מנהל	שם מלא של המזמין
טלפון סלולרי של המזמין	string	לא	ע"י מנהל	טלפון של המזמין. ליצירת קשר במקרה של בעיה. מייצג טלפון סלולרי תקני של השליח. מחרוזת בת 10 ספרות בדיוק המתחילה בספרה 0.
נפח, משקל, שביר וכדומה	תכונות חובה, אך לשיקולכם		ע"י מנהל	תוספת של פרטים על ההזמנה על פי סוג החברה לשיקולכם. הוסיפו תכונות כרצונכם.
זמן פתיחת הזמנה	DateTime	לא	לא	זמן (תאריך ושעה) שבו נפתחה ההזמנה. זהו בעצם הזמן שבו נוצרה ישות ההזמנה הנוכחית. ייקבע על פי הזמן הנוכחי של שעון המערכת, שנמצא בישות התצורה. ראה פירוט כאן: זמנים בחיי הזמנה

ישות נתונים משלוח - DO.Delivery

ישות המקשרת בין "הזמנה" ל"שליח" שבחר לטפל בה. שליח בחר לבצע משלוח עבור ההזמנה. כולל מספר מזהה רץ וייחודי של ישות הקישור. וכן מספר מזהה של ההזמנה ומספר ת"ז של השליח.

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף תקינות הערך תיבדק במימוש השכבה הולגית. בשכבת הנתונים יש להניח שהערך תקין.
מספר מזהה רץ של ישות המשלוח Id	int	לא	לא	מייצג מספר המזהה את ישות המשלוח באופן ייחודי. כששליח בוחר לטפל בהזמנה, נוצרת ישות מקשרת. ניגש לישות התצורה כדי לקבל עבורה את מספר המשלוח הרץ הבא.
מספר מזהה רץ של ישות ההזמנה OrderId	int	לא	לא	מייצג מספר המזהה את ההזמנה שהשליח בחר לטפל בה
ת"ז שליח CourierId	int	לא	לא	מייצג ת"ז של השליח שבחר לטפל בהזמנה. במקרה של "משלוח מדומה", הישות נוצרה כתוצאה מ"ביטול" הזמנה (ע"י מנהל) שעדיין לא עברה לטיפול של אף שליח - אזי ת"ז תהיה בערך 0.
סוג השילוח	ENUM	לא	לא	סוג השילוח בעת פתיחת המשלוח. (סוג השילוח יכול להשתנות אצל בישות נתונים שליח אך לא ישפיע כאן על משלוחים שכבר בוצעו על ידו)
זמן תחילת משלוח	DateTime	לא	לא	זמן (תאריך ושעה) שבו ההזמנה הנוכחית נאספה על ידי השליח מהחברה. זה גם הזמן שבו נוצרה ישות המשלוח הנוכחית. ייקבע על פי הזמן הנוכחי של שעון המערכת, שנמצא בישות התצורה. במקרה של "משלוח מדומה", נוצרה עבורו ישות משלוח כתוצאה מ"ביטול" הזמנה (ע"י מנהל) - אזי זמן סיום המשלוח יהיה זהה לזמן תחילת המשלוח. ראה פירוט כאן: זמנים בחיי הזמנה
מרחק בפועל	double	כן	לא	מרחק מחושב בק"מ של כתובת ההזמנה מכתובת החברה. יחושב על פי סוג השילוח שבחר לטפל בהזמנה. ראה פירוט כיצד לחשב מרחק, כאן: חישוב מרחק הזמנה בפועל המרחק יחושב בשכבה הולגית וישמר בשכבת הנתונים בתכונה הזו. כל עוד חישוב המרחק לא הושלם הערך יהיה null. אם החישוב הסתיים בכישלון (תקלת רשת וכדומה) אזי המשלוח ייסגר עם סוג סיום "נכשל". בשלב הראשונים החישוב יתבצע בצורה סינכרונית ובשלב 7 של הפרויקט החישוב יהיה חייב להתבצע בצורה אסינכרונית.
סוג סיום המשלוח	ENUM	כן	לא	האופן שבו הסתיים הטיפול בהזמנה הנוכחית על ידי השליח הנוכחי. ראה פירוט של סוג סיום כאן: סוגי סיום משלוח במקרה שההזמנה עדיין לא הסתיימה והמשלוח עדיין בדרכו ערך התכונה יהיה null. במקרה של "משלוח מדומה" שנוצרה עבורו ישות משלוח כתוצאה מ"ביטול" (ע"י מנהל) - אזי סוג סיום המשלוח יהיה "בוטל".
זמן סיום משלוח	DateTime	כן	לא	זמן (תאריך ושעה) שבו המשלוח הסתיים בכל אחד מסוגי סיום המשלוח. כאשר מתעדכנת התכונה של "סוג סיום המשלוח" אזי ערך התכונה של "זמן סיום משלוח" מתעדכן בהתאמה על פי הזמן הנוכחי של שעון המערכת (תכונה בישות התצורה). כל עוד המשלוח לא הסתיים 2 תכונות אלו יהיו null. ישות המשלוח ממשיכה להתקיים, ולא נמחקת, לצורך בקרה על היסטוריית המשלוחים. במקרה שההזמנה עדיין לא הסתיימה והמשלוח עדיין בדרכו ערך התכונה יהיה null. במקרה של "משלוח מדומה", נוצרה עבורו ישות משלוח כתוצאה מ"ביטול" (ע"י מנהל) - אזי זמן

סיום המשלוח יהיה זהה לזמן תחילת המשלוח. ראה פירוט כאן: זמנים בחיי הזמנה				
--	--	--	--	--

חוזי השירות (ממשקים) של שכבת הנתונים - DalApi

יש להגדיר ממשק CRUD עבור כל ישות ולממש אותו על פי המפורט במסמך שלב 1 ובמסמך שלב 2 של הפרויקט:

- נגדיר בשכבת הנתונים את כל הממשקים הנדרשים לעבודה מול כל ישויות נתונים.
 - עבור כל ישות נגדיר ממשק שיכיל את כל מתודות ה-CRUD לצורך הוספה/צפייה/עדכון/מחיקה של ישות אחת.
 - עבור כל ישות נממש את ממשק ה-CRUD שלה:
 - חשוב מאוד: על מנת לממש את מתודות הממשק כראוי, להיצמד לתסריט המדויק של כל מתודת CRUD, ראו במסמך של שלב 1 את הנספח שנקרא: "הרחבה בנושא מתודות הגישה CRUD".
 - בנוסף, נגדיר ממשק עבור התצורה IConfig ונממש אותו
- שימו לב: במימוש מתודות ה-CRUD האלו, יש להניח שהקלט למתודה תקין וכל ערכי התכונות כבר מגיעות מלאות בערכים תקינים - אין להפעיל לוגיקה או לבדוק תקינות.

הלוגיקה היחידה שתהיה מותרת בשכבה זו:

1. בדיקת תקינות של מזהה

- בהוספה - יש להניח שכל הערכים הגיעו תקינים. אם זו ישות מגיעה עם מספר מזהה מוכן (כגון ת"ז), אז רק לוודא שאין כבר ישות עם מספר מזהה כזה לעדכון. אחרת, זורקים חריגה. אם זו ישות שלא מגיעה עם מספר מזהה מוכן (כגון מספר רץ), אז שכבת נתונים תייצר אותו בעצמה מתוך מתודת ההוספה.
- בצפייה - רק לוודא שיש אכן ישות עם מספר מזהה כזה לצפייה. אחרת, זורקים חריגה.
- בעדכון - יש להניח שהאובייקט מגיע עם כל הערכים תקינים - רק לוודא שיש אכן ישות עם מספר מזהה כזה לעדכון. אחרת, זורקים חריגה.
- במחיקה - ההחלטה אם מותר למחוק את הישות או לא התקבלה בשכבה הלוגית לכן, בשכבת הנתונים יש להניח שאם בקשו למחוק אז זה כנראה חוקי - רק לוודא שיש אכן ישות עם מספר מזהה כזה למחיקה. אחרת, זורקים חריגה.

2. סינון לפי פילטר המועבר משכבה עליונה (שייך לשלב 2 מתקדם יותר)

את שאר הלוגיקה והבדיקות נבצע בשלבים מתקדמים יותר בשכבה עליונה, שכבת ה-BL.

דגשים נוספים:

- ישות שליח היא ישות שהמספר המזהה שלה הוא ת"ז ואינו מספר רץ, מספר זה מתקבל מראש ביחד עם האובייקט ולא נוצר בשכבת הנתונים (DAL). לכן, בהוספת שליח יש לוודא שאינו קיים כבר שליח עם ת"ז כזו במערכת, אחרת יש לזרוק חריגה.
- לעומת זאת, ישות הזמנה וישות המשלוח הן ישויות שהמספר המזהה שלהן הוא מספר רץ שיוצר בתוך שכבת נתונים כחלק ממימוש המתודה Create, על ידי בקשה לקבלת המספר הרץ הבא מישות התצורה.
- תאריכים של תכונות (של ישות DO) שאמורים להיקבע על פי שעון המערכת (כגון: זמן פתיחה של הזמנה), נקבעים כבר בשכבה הלוגית (או ב DalTest) ומגיעים כבר עם ערך מוכנים לתוך מתודות ה-CRUD.

אתחול הנתונים - Initialize

נרצה לאתחל את בסיס הנתונים בערכים. נרצה למלא את הרשימות בנתונים הגיוניים שיש ביניהם קשר והם יוצרים מערכת מתפקדת שאפשר להתחיל לעבוד איתה ולבדוק אותה.

אתחול נתונים יעשה כחלק מפרויקט DalTest, בתוך מתודת האתחול Initialize.Do, ובמתודות עזר - כפי שמפורט במסמך שלב 1 של הפרויקט.

אתחול משתני התצורה

- מנהל אחד - מספר ת"ז שלו שמור בישות התצורה
- כתובת החברה והקואורדינטות שלה - בשלב האתחול יוזנו ידנית במפורש דרך הקוד בישות התצורה
 - ראה פירוט כאן: [נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות](#)
- כל שאר משתני התצורה ישמרו גם הם באופן ידני דרך הקוד - עם ערכים מתאימים לשיקולכם

אתחול האובייקטים ברשימות נתונים

תזכורת: באתחול הנתונים יש לייצר נתונים תקינים וחוקיים, עם קשרים הגיוניים ביניהם, תוך שמירה על שלמות הנתונים.

הנתונים נוצרים רק ע"י בקשה לשכבת DAL, בקשה להוספת (Create) אובייקט ליישות המתאימה. יש לאתחל תכונות מתאימות של ישות התצורה לפני יצירת אובייקטים ליתר הישיוות. אתחול הרשימות של הישיוות תלוי בנתונים של תצורה (כפי שמשקף מההנחיות הבאות). מותר ליצור מתודות עזר לכל צורך שיעלה על מנת לא ליצור Code Smell של חזרות מיותרות (Needless Repetition).

• שליחים:

- לפחות 20 שליחים
 - רובם פעילים ומעט לא פעילים (שהיו פעילים בעבר, עם פעילות קיימת)
- מכל סוגי השילוח (רכב/אופנוע/אופניים/רגלי)
- יש ליצור אוסף קבוע של שליחים (ת"ז, שמות, כתובת דוא"ל, סוג השילוח, וכו')
- יש להגריל את סוג השילוח בצורה כזו שההסתברות לכל סוג השילוח תהיה שווה
- עבור חלק מהשליחים, יש להגריל בצורה רנדומלית מרחק מירבי לקבלת הזמנה
 - בתוך הטווח המירבי של החברה כולה
 - טווח המספרים חייב להיות הגיוני עבור השליחים, בהתאם לסוג השילוח
- יש להגריל זמני תחילת עבודה שונים שקטנים מהזמן של שעון המערכת ובטווח זמן הגיוני
- לגבי שאר התכונות יש להפעיל הגיון בריא

• הזמנות:

- לפחות 50 הזמנות, מתוכן:
 - לפחות 20 פתוחות
 - לפחות 10 בטיפול כרגע
 - לפחות 20 סגורות (מכל הסוגים)
- מספר ההזמנה נוצר אוטומטית בתוך שכבת DAL, לכן אנחנו ניתן 0 במספר ההזמנה
- בהתאם לסוג החברה, יש ליצור מאגר הגיוני של הזמנות, הזמנות מסוגים שונים עם תיאור מילולי
- יש להגריל זמני פתיחת הזמנות שונים שקטנים מהזמן של שעון המערכת ובטווח זמן הגיוני
- **לגבי כתובות של הזמנות:**

- יש להשתמש בכתובות אמיתיות ובהתאם למרחק המירבי לאספקה שהחברה התחייבה לו, יש למקם את ההזמנות בכתובות שמרוכזות פחות או יותר באותו אזור של כתובת החברה עצמה
- חישוב קווי האורך והרוחב והמרחקים על כל סוגים יעשה ידנית כחלק מעבודת הכנת רשימת הכתובות.

- ראה פירוט כאן: [נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות](#)
- עליכם להכין רשימה (מערך) של כתובות עם קווי האורך והרוחב האמיתיים שלהם והמרחק שלהן מכתובת החברה. כל סוגי המרחקים מהחברה: אווירי, הליכה ונהיגה - נזדקק להם בהמשך האתחולים

- בהזמנות, הכתובת תוגרל מתוך הרשימה הנ"ל
- יש לכתוב מתודת עזר שתחשב את המרחק האווירי בין כתובת נתונה (כתובת של הזמנה) לכתובת החברה ע"פ קואורדינטות (באתחול של רשימת משלוחים ייעשה שימוש במרחק זה)

• משלוחים (הקצאות של שליחים להזמנות):

- משלוחים בכל סוגי סיום המשלוח (כולל כאלו שנסגרו ונפתחו מחדש), בהתאם להזמנות כנ"ל
- אתחול רשימת משלוחים יתבצע **אחר** אתחול רשימות השליחים וההזמנות

- בראשית אתחול משלוחים, נבקש משכבת DAL את רשימות ההזמנות והשליחים ונשתמש בהם עבור יצירת משלוחים כדלקמן
- ניצור את המשלוחים בלולאה:
 - מספר מזהה של המשלוח נוצר אוטומטית בשכבת DAL, לכן ניתן 0 במספר המשלוח
 - נגריל הזמנה (מתוך רשימת ההזמנות) - במשלוח נרשום את מספר ההזמנה, ואת המרחק שלה על פני סוג השילוח
 - נגריל שליח בין השליחים שהמרחק של ההזמנה לא עולה על מרחק מירבי של השליח - במשלוח נרשום את ת"ז של השליח
 - נגריל זמן פתיחת המשלוח בהתאם לזמן פתיחת ההזמנה, בצורה הגיונית, כך שלא יהיה בתוך זמן משלוח שטופל גם על ידי השליח
 - נגריל האם המשלוח בטיפול או הסתיים - יש להבטיח תקינות ושלמות הנתונים על מנת שלא יהיה לשליח משלוחים שהוא טיפל או מטפל בהם בו זמנית
 - אם המשלוח הסתיים:
 - נגריל זמן סיום המשלוח, בצורה הגיונית, בהתאם לדרישת שלמות ותקינות הנתונים עבור השליח, ההזמנה והמשלוח
 - נגריל סוג סיום המשלוח - תיצרו הסתברויות להגרלת כל סוגי סיום לפי הגיון בריא
 - אם המשלוח בטיפול או עם סוג סיום שסוגר את ההזמנה - ההזמנה תימחק מרשימת ההזמנות שאנו משתמשים בה עבור המשלוחים - על מנת שלא נגריל את ההזמנה למשלוח נוסף (בטעות...)

איפוס נתוני שכבת הנתונים

איפוס נתוני התצורה

כחלק מממשק התצורה (IConfig) תצטרכו להגדיר ולממש מתודת איפוס (Reset) - שמאפסת את נתוני התצורה. מחזירה אותם לערכים ההתחלתיים שלהם.

- מספרים מזהים רצים - אתחל בערך התחלתי כרצונך, לא חייב להתחיל מ 1
- שעון המערכת - אפשר לאתחל בערך הזמן הנוכחי של **שעון המערכת**, או בכל ערך הגיוני אחר
- כל שאר נתוני התצורה - אתחל בערך התחלתי, בהתאם לצרכי המערכת שלך

איפוס נתוני הרשימות

ניקוי כל רשימות נתונים עבור כל ישות (רשימה ללא איברים).

יש לעבור על כל אחת מהישויות ולאפס את נתוניה בעזרת מתודת ה CRUD שלה בשם: DeleteAll

תוכנית הבדיקה - DalTest

תוכנית הבדיקה DalTest, צריכה לאפשר בלולאה את האפשרויות הבאות:

- יציאה
- קלט/פלט מהמשתמש לצורך הוספה, צפייה, עדכון, מחיקה, מחיקת הכל - עבור כל אחת מהישויות
 - תציע תפריט לכל אחת מהישויות עם כל אחת מהפעולות CRUD
 - לאחר מכן, תקלוט נתונים, ותזמן את המתודות המתאימות דרך הממשק המתאים לכל ישות.
 - תתפוס חריגות בעת הצורך, ותפלוט למסך את הפלט המתקבל
- אתחול בסיס נתונים
- צפיה בכל בסיס נתונים בבת אחת
- תפריט לקביעת נתוני התצורה
- איפוס כל בסיס נתונים (תצורה+רשימות)

ראו את דרישות תוכנית הבדיקה בהרחבה ובפירוט במסמך שלב 1 של הפרויקט.

תזכורת: בהרצת התוכנית יש להזין נתונים תקינים וחוקיים, עם קשרים הגיוניים ביניהם, תוך שמירה על שלמות הנתונים.

בדיקות תקינות הפורמט של הקלט עבור נתונים שאינם מחרוזת מתבצעות אוטומטית בהפעלת פעולת Parse או TryParse.

ואם הפעולה לא הצליחה - יש לבקש להזין מחדש.

שכבת הנתונים DAL - שמירת הנתונים בקבצי XML (שלב 3 בפרויקט)

- נאפשר שמירה ואחזור של הנתונים ממקור נתונים נוסף - קבצי XML
 - כפי שכתוב בפירוט שלב 3, פרויקט DalXml, יש לשמור קובץ xml נפרד עבור כל ישות נתונים (DO)
 - כלומר 3 קבצי xml, המקבילים לרשימות ב-DataSource, המהווים את מקור הנתונים בפרויקט DalList:
 - קובץ לשמירת רשימת השליחים: couriers.xml
 - קובץ לשמירת רשימת ההזמנות: orders.xml
 - קובץ לשמירת רשימת המשלוחים: deliveries.xml
- נאפשר שמירה ואחזור של נתוני התצורה בקובץ data-config.xml:
 - ערכי מזהים אוטומטים רצים
 - שעון המערכת
 - וכל שאר נתוני התצורה

השכבה הלוגית BL (שלב 4 בפרויקט)

תיאור כללי

ישויות הנתונים הלוגיות כולן נגזרות מהצרכים העולים מהמסכים עצמם.

כדי להבין טוב יותר את צרכי המסכים ניתן להיעזר ב: [תרשימי המסכים](#) וכן ב: [פירוט המסכים והפונקציונאליות שלהם](#) המופיעים במסמך זה.

בשכבה הלוגית נגדיר 2 סוגים של ישויות לוגיות:

- ישויות נתונים לוגיות - ישויות BO. חוזי נתונים (מחלקות). מיועדות לצורכי תצוגה במסכים בלבד ולא נשמרות בבסיס נתונים.
 - [ישויות נתונים לוגיות שליח - BO.Courier](#)
 - כוללת בתוכה ישות מטיפוס "הזמנה בטיפול שליח" - במידה ויש הזמנה שנמצאת כרגע בטיפול
 - [ישויות נתונים לוגיות "הזמנה בטיפול שליח" - BO.OrderInProgress](#) (כלולה בישות "שליח" לעיל)
 - [ישויות נתונים לוגיות "שליח ברשימה" - BO.CourierInList](#)
 - [ישויות נתונים לוגיות הזמנה - BO.Order](#)
 - כוללת בתוכה את רשימת ההקצאות עבור ההזמנה מטיפוס [List of DeliveryInList](#) - עבור כל הקצאות ההזמנה בעבר או בהווה
 - [ישויות נתונים לוגיות "משלוח ברשימה" - BO.DeliveryPerOrderInList](#) (כלולה בישות "הזמנה" לעיל)
 - [ישויות נתונים לוגיות "הזמנה ברשימה" - BO.OrderInList](#)
 - [ישויות נתונים לוגיות "הזמנה פתוחה ברשימה" - BO.OpenOrderInList](#)
 - [ישויות נתונים לוגיות "משלוח סגור ברשימה" - BO.ClosedDeliveryInList](#)
 - ישויות שירות לוגיות - חוזי שירות (ממשקים) המגדירים בתוכם מתודות שעושות שימוש בישויות הנתונים לוגיות כדי לבצע את הפונקציונליות הנדרשת.
 - [ישויות שירות לוגיות "שליח" ICourier](#) - הצעות למתודות
 - [ישויות שירות לוגיות "הזמנה" IOrder](#) - הצעות למתודות
 - [ישויות שירות לוגיות "ניהול" IAdmin](#) - הצעות למתודות
- מיד בהמשך פרק זה נרחיב ונפרט:
- עבור כל ישות נתונים לוגית - הצעה של רשימת התכונות שעליכם להגדיר עבורה ואת משמעותן
 - עבור כל ישות שירות לוגית (ממשק) - הצעה של רשימת המתודות שעליכם להגדיר עבורה ואת משמעותן

חשוב מאוד: ייתכן ותידרשו לישויות נתונים לוגיות נוספות או לתכונות נוספות בישויות הנתונים הלוגיות, או למתודות נוספות בישויות השירות הלוגיות - על כל זוג להפעיל שיקול דעת על פי צרכי המסכים והפונקציונליות של המערכת שלו.

ישויות הנתונים הלוגיות (חוזי נתונים - מחלקות) - BO

בסעיף זה נפרט עבור כל ישות נתונים לוגית - הצעה של רשימת התכונות שעליכם להגדיר עבורה ואת משמעותן.

ישות נתונים לוגית שליח - BO.Courier

- מכילה פרטים אישיים עבור ישות נתונים לוגית "שליח"
 - וכללת בתוכה ישות נתונים לוגית "הזמנה בטיפול שליח" במידה ויש הזמנה שנמצאת כרגע בטיפול.
- ישות שהיא לצפיה/עדכון/הוספה/מחיקה
- הישות תייצג "שליח" במסכים:
 - מסך "ניהול שליח בודד", מסך "הוספת שליח" - עבור מנהל
 - מסך "שליח ראשי" - עבור שליח

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף תקנות הערך נבדקת במימוש השכבה הלוגית
ת"ז שליח id	int	לא	לא	בצפיה - לקוח מ- ישות נתונים שליח - DO.Courier בהוספה - יש לבדוק תקינות בשכבה הלוגית ולשמור בישות הנ"ל
שם מלא (פרטי ומשפחה)	string	לא	ע"י מנהל או שליח	בצפיה - ישות נתונים שליח - DO.Courier בהוספה ועדכון - יש לשמור בישות הנ"ל
טלפון סלולרי	string	לא	ע"י מנהל או שליח	בצפיה - ישות נתונים שליח - DO.Courier בהוספה ועדכון - יש לבדוק תקינות בשכבה הלוגית ולשמור בישות הנ"ל
אימייל	string	לא	ע"י מנהל או שליח	בצפיה - ישות נתונים שליח - DO.Courier בהוספה ועדכון - יש לבדוק תקינות בשכבת הלוגית
סיסמה	string	לא	ע"י מנהל או שליח	תוספת תכונה זו לא תהיה קיימת עבור מי שלא עושה את התוספת הזו. בצפיה - ישות נתונים שליח - DO.Courier בהוספה ועדכון - יש לבדוק תקינות בשכבה הלוגית ולשמור בישות הנ"ל
פעיל	boolean	לא	ע"י מנהל בלבד	בצפיה - ישות נתונים שליח - DO.Courier בהוספה ועדכון - יש לבדוק תקינות בשכבה הלוגית ולשמור בישות הנ"ל
מרחק מירבי אישי למשלוח	double	כן	ע"י מנהל או שליח	בצפיה - ישות נתונים שליח - DO.Courier בהוספה ועדכון - יש לבדוק תקינות בשכבה הלוגית ולשמור בישות הנ"ל
סוג השילוח	ENUM	לא	ע"י מנהל או שליח	בצפיה - ישות נתונים שליח - DO.Courier בהוספה ועדכון - יש לבדוק תקינות בשכבה הלוגית ולשמור בישות הנ"ל
זמן תחילת עבודה בחברה	DateTime	לא	לא	תכונה לצפיה בלבד בצפיה - ישות נתונים שליח - DO.Courier בהוספה - מיוצר מתוך שכבת הנתונים בזמן יצירת הישות הנ"ל
סך ההזמנות שסיפק בזמן	int	לא	לא	תכונה לצפיה בלבד . הערך יחושב בשכבה הלוגית. סך המשלוחים הקיימים עבור אותו שליח עם סוג סיום סופק וזמן סיום משלוח קטן/שונה מהזמן המחושב כזמן אספקה מירבי
סך ההזמנות שסיפק באיחור	int	לא	לא	תכונה לצפיה בלבד . הערך יחושב בשכבה הלוגית. סך המשלוחים הקיימים עבור אותו שליח עם סוג סיום סופק וזמן סיום משלוח גדול מהזמן המחושב כזמן אספקה מירבי
הזמנה בטיפול שליח	BO.OrderInProgresses	כן		תכונה לצפיה בלבד. בשכבה הלוגית, שאילתא שתחזיר את ישות נתונים לוגית "הזמנה בטיפול שליח" - BO.OrderInProgress במידה וקיימת כרגע הזמנה בטיפולו של אותו שליח.

				אחרת - null.
--	--	--	--	--------------

ישות נתונים לוגית "הזמנה בטיפול שליח" - BO.OrderInProgress

- מכילה פרטים עבור ישות נתונים לוגית "הזמנה בטיפול שליח"
- ישות שהיא לצפיה בלבד
- תופיע כחלק מישות הנתונים הלוגית "שליח" במסכים:
 - מסך "ניהול שליח בודד" - עבור מנהל
 - מסך "שליח ראשי" - עבור שליח

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף ישות שהיא לצפיה בלבד ולכן אין בדיקת תקינות
מספר מזהה של ישות המשלוח DeliveryId	int	לא	לא	לא יופיע בתצוגה. לקוח מ- ישות נתונים משלוח - DO.Delivery נמצא ע"י חיפוש ב- DO.Delivery האם קיים משלוח שעדיין לא נסגר עם Id של השליח
מספר מזהה רץ של ישות ההזמנה OrderId	int	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery נמצא ע"י חיפוש ב- DO.Delivery האם קיים משלוח שעדיין לא נסגר עם Id של השליח
סוג הזמנה	ENUM	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
תיאור מילולי	string	כן	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
כתובת מלאה של ההזמנה	string	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
מרחק אווירי	double	לא	לא	הערך יחושב בשכבה הלוגית. מרחק אווירי מהחברה - חישוב מתמטי פשוט שאינו מושפע מסוג השילוח.
מרחק בפועל	double	כן	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
שם מלא של המזמין	string	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
טלפון של המזמין	string	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
זמן פתיחת הזמנה	DateTime	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
זמן תחילת משלוח	DateTime	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
זמן אספקה צפוי	DateTime	לא	לא	הערך יחושב בשכבה הלוגית. זמן (תאריך ושעה) הצפוי לקבלת המשלוח על פי זמן תחילת המשלוח הנוכחי ובהתחשב בסוג השילוח (מרחק ההזמנה ומהירות ממוצעת של השליח). ראה פירוט כאן: זמנים בחיי הזמנה
זמן אספקה מירבי	DateTime	לא	לא	הערך יחושב בשכבה הלוגית. זמן (תאריך ושעה) אספקה מירבי בהתבסס על טווח האספקה המירבי שהחברה התחייבה אליו ועל הזמן שבו ההזמנה נפתחה על ידי המנהל (בלי להתחשב בסוג השילוח). ראה פירוט כאן: זמנים בחיי הזמנה
סטטוס הזמנה OrderStatus	ENUM	לא	לא	הערך יחושב בשכבה הלוגית. סטטוס של הזמנה בהיבט של מצב המשלוח האחרון על סמך הנתונים הקיימים בישויות לקוח מ- ישות נתונים הזמנה - DO.Order ו- ישות נתונים משלוח - DO.Delivery ראה פירוט כאן: סטטוס הזמנה (OrderStatus)
סטטוס עמידה בזמנים ScheduleStatus	ENUM	לא	לא	הערך יחושב בשכבה הלוגית. סטטוס של הזמנה בהיבט של עמידה בזמנים של המשלוח האחרון על סמך הנתונים הקיימים בישויות לקוח מ- ישות נתונים הזמנה - DO.Order ו- ישות נתונים משלוח - DO.Delivery . ובישות התצורה. ראה פירוט כאן: סטטוס עמידה בזמנים (ScheduleStatus)

סך זמן שנותר לסיום ההזמנה	TimeSpan	לא	לא	הפרש הזמנים בין זמן האספקה המירבי של ההזמנה לבין הזמן הנוכחי של שעון המערכת.
---------------------------	----------	----	----	--

ישות נתונים לוגית "שליח ברשימה" - BO.CourierInList

- מכילה פרטים עבור ישות נתונים לוגית "שליח ברשימה"
- ישות שהיא לצפיה בלבד
- תופיע כישות בודדת ברשימה של שליחים, במסכים:
 - מסך "ניהול שליחים" - עבור מנהל

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף ישות שהיא לצפיה בלבד ולכן אין בדיקת תקינות
ת"ז שליח Id	int	לא	לא	לקוח מ- ישות נתונים שליח - DO.Courier
שם מלא (פרטי ומשפחה)	string	לא	לא	לקוח מ- ישות נתונים שליח - DO.Courier
פעיל	boolean	לא	לא	לקוח מ- ישות נתונים שליח - DO.Courier
סוג השילוח	ENUM	לא	לא	לקוח מ- ישות נתונים שליח - DO.Courier
זמן תחילת עבודה בחברה	DateTime	לא	לא	לקוח מ- ישות נתונים שליח - DO.Courier
סך ההזמנות שסיפק בזמן	int	לא	לא	הערך יחושב בשכבה הלוגית. סך המשלוחים הקיימים עבור אותו שליח עם סוג סיום סופק וזמן סיום משלוח קטן/שווה מהזמן המחושב כזמן אספקה מירבי
סך ההזמנות שסיפק באיחור	int	לא	לא	הערך יחושב בשכבה הלוגית. סך המשלוחים הקיימים עבור אותו שליח עם סוג סיום סופק וזמן סיום משלוח גדול מהזמן המחושב כזמן אספקה מירבי
מספר מזהה של ההזמנה בטיפולו (במידה וקיימת)	int	כן	לא	האם קיימת ברשימת ישויות המשלוח ישות עבור השליח שזמן סיום המשלוח הוא עדיין null

ישות נתונים לוגית הזמנה - BO.Order

- מכילה פרטים אישיים עבור ישות נתונים לוגית "הזמנה"
 - הכוללת בתוכה רשימת משלוחים מטיפוס "משלוח ברשימה" (משלוחים קודמים עבור ההזמנה (במידה והיו) ומשלוח נוכחי במידה וקיים)
- ישות שהיא לצפיה/עדכון/הוספה/מחיקה
- הישות תייצג "הזמנה" במסכים:
 - מסך "ניהול הזמנה בודדת", מסך "הוספת הזמנה" - עבור מנהל

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף תקינות הערך נבדקת במימוש השכבה הלוגית
מספר מזהה רץ של ישות ההזמנה Id	int	לא	לא	תכונה לצפיה בלבד בצפיה - לקוח מ- ישות נתונים הזמנה - DO.Order בהוספה - מיוצר מתוך שכבת הנתונים בזמן יצירת הישות הנ"ל
סוג הזמנה	ENUM	לא	ע"י מנהל	בצפיה - לקוח מ- ישות נתונים הזמנה - DO.Order בהוספה ועדכון - יש לשמור בישות הנ"ל
תיאור מילולי	string	כן	ע"י מנהל	בצפיה - לקוח מ- ישות נתונים הזמנה - DO.Order בהוספה ועדכון - יש לשמור בישות הנ"ל
כתובת מלאה של ההזמנה	string	לא	ע"י מנהל	כתובת מלאה ואמיתית בפורמט תקין, של מיקום ההזמנה. דוגמא לכתובות: הנשיאים 7, פתח תקוה, ישראל המרכז האקדמי לב, ירושלים, ישראל השכבה הלוגית תבדוק האם הכתובת אכן קיימת במציאות ותחשב את קו הרוחב שלה (latitude) ואת קו האורך שלה (longitude) בכדור הארץ. והמידע הזה יישמר גם הוא בתכונות ההזמנה. בשכבת הנתונים יכולה להיווצר רק הזמנה עם כתובת תקינה כולל קו רוחב וקו אורך תקינים. בשלבם הראשונים החישוב יתבצע בצורה סינכרונית ובשלב 7 של הפרויקט החישוב יהיה חייב להתבצע בצורה אסינכרונית.
קו רוחב Latitude	double	לא	מתעדכן אוטומטית ע"י השכבה הלוגית בעקבות עדכון כתובת ההזמנה	לא יופיע בתצוגה. קו רוחב - מספר המציין עד כמה נקודה על פני כדור הארץ נמצאת מדרום או מצפון לקו המשווה. ערך התכונה יתעדכן על ידי השכבה הלוגית בכל פעם הכתובת של ההזמנה תעודכן. קיימות פונקציות שמחשבות אותו. תכונה זו נועדה לצורך חישוב מרחקים בין כתובות, ולא אמורה להיות מוצגת בסופו של דבר בתצוגה.
קו אורך Longitude	double	לא	מתעדכן אוטומטית ע"י השכבה הלוגית בעקבות עדכון כתובת ההזמנה	לא יופיע בתצוגה. קו אורך - מספר המציין עד כמה נקודה על פני כדור הארץ נמצאת ממזרח או ממערב לקו המשווה. ערך התכונה יתעדכן על ידי השכבה הלוגית בכל פעם הכתובת של ההזמנה תעודכן. קיימות פונקציות שמחשבות אותו. תכונה זו נועדה לצורך חישוב מרחקים בין כתובות, ולא אמורה להיות מוצגת בסופו של דבר בתצוגה.
מרחק אווירי	double	לא	מתעדכן אוטומטית ע"י השכבה הלוגית בעקבות עדכון כתובת ההזמנה	הערך יחושב בשכבה הלוגית. מרחק אווירי מהחברה - חישוב מתמטי פשוט שאינו מושפע מסוג השילוח.
שם מלא של המזמין	string	לא	ע"י מנהל	בצפיה - לקוח מ- ישות נתונים הזמנה - DO.Order בהוספה ועדכון - יש לשמור בישות הנ"ל
טלפון של המזמין	string	לא	ע"י מנהל	בצפיה - לקוח מ- ישות נתונים הזמנה - DO.Order בהוספה ועדכון - יש לבדוק תקינות בשכבה הלוגית ולשמור בישות הנ"ל
נפח, משקל, שביר וכדומה	תכונות חובה, אך לשיקולכם	ע"י מנהל	בצפיה - לקוח מ- ישות נתונים הזמנה - DO.Order בהוספה ועדכון - יש לשמור בישות הנ"ל	
זמן פתיחת הזמנה	DateTime	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order

זמן אספקה צפוי	DateTime	כן לא	לא	הערך יחושב בשכבה הלוגית. זמן (תאריך ושעה) הצפוי לקבלת המשלוח על פי זמן תחילת המשלוח הנוכחי. אם קיים ובהתחשב בסוג השילוח (מרחק ההזמנה ומהירות ממוצעת של השליח). ראה פירוט כאן: זמנים בחי" הזמנה
זמן אספקה מירבי	DateTime	לא	לא	הערך יחושב בשכבה הלוגית. זמן (תאריך ושעה) אספקה מירבי בהתבסס על טווח האספקה המירבי שהחברה התחייבה אליו ועל הזמן שבו ההזמנה נפתחה על ידי המנהל (בלי להתחשב בסוג השילוח). ראה פירוט כאן: זמנים בחי" הזמנה
סטטוס הזמנה OrderStatus	ENUM	לא	לא	הערך יחושב בשכבה הלוגית. סטטוס של הזמנה בהיבט של מצב המשלוח האחרון על סמך הנתונים הקיימים ב- ישות נתונים הזמנה - DO.Order ו- ישות נתונים משלוח - DO.Delivery . ראה פירוט כאן: סטטוס הזמנה (OrderStatus) נ.ב. יתכן שתמצאו שהתכונה אינה חשובה לרלוונטיות בפועל בישות הזו. אם זאת תהיה מסקנתכם - אין חובה להגדיר את התכונה.
סטטוס עמידה בזמנים ScheduleStatus	ENUM	לא	לא	הערך יחושב בשכבה הלוגית. סטטוס של הזמנה בהיבט של עמידה בזמנים של המשלוח האחרון על סמך הנתונים הקיימים ב- ישות נתונים הזמנה - DO.Order ו- ישות נתונים משלוח - DO.Delivery . ובישות התצורה. ראה פירוט כאן: סטטוס עמידה בזמנים (ScheduleStatus)
סך זמן שנותר לסיום ההזמנה	TimeSpan	לא	לא	אם ההזמנה סגורה אז 00:00:00. אחרת, הפרש הזמנים בין זמן האספקה המירבי של ההזמנה לבין הזמן הנוכחי של שעון המערכת.
רשימת משלוחים עבור הזמנה	List of BO.DeliveryPerOrderInList	כן	לא	רשימה של ישויות מטיפוס: ישות נתונים לוגית "משלוח ברשימה" - BO.DeliveryPerOrderInList . שאינן שמחזירה את כל המשלוחים שנעשו עבור ההזמנה. כולל אלו שלא הסתיימו בהצלחה. בטרם סופקה אם אין אף משלוח עדיין - null

ישות נתונים לוגית "משלוח ברשימה" - BO.DeliveryPerOrderInList

- מכילה פרטים עבור ישות נתונים לוגית "משלוח ברשימה"
- ישות שהיא לצפיה בלבד
- תופיע כישות בודדת ברשימה של משלוחים עבור הזמנה, במסכים:
 - מסך "ניהול הזמנה בודדת" - עבור מנהל

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף ישות שהיא לצפיה בלבד ולכן אין בדיקת תקינות
מספר מזהה רץ של ישות המשלוח DeliveryId	int	לא	לא	לא יופיע בתצוגה. לקוח מ- ישות נתונים משלוח - DO.Delivery
ת"ז שליח CourierId	int	כן	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
שם שליח	string	לא	לא	לקוח מ- ישות נתונים שליח - DO.Courier
סוג השילוח	ENUM	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
זמן תחילת משלוח	DateTime	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
סוג סיום משלוח	ENUM	כן	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
זמן סיום משלוח	DateTime	כן	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery

ישות נתונים לוגית "הזמנה ברשימה" - BO.OrderInList

- מכילה פרטים עבור ישות נתונים לוגית "הזמנה ברשימה"
- ישות שהיא לצפיה בלבד
- תופיע כישות בודדת ברשימה של הזמנות כלליות מכל הסוגים, במסכים:
 - מסך "ניהול הזמנות" - עבור מנהל

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף ישות שהיא לצפיה בלבד ולכן אין בדיקת תקינות
מספר מזהה רץ של ישות המשלוח DeliveryId	int	כן	לא	לא יופיע בתצוגה. לקוח מ- ישות נתונים משלוח - DO.Delivery אך במידה ולא נעשה עדיין אף משלוח על ההזמנה הזו - יהיה null.
מספר מזהה רץ של ישות ההזמנה OrderId	int	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
סוג הזמנה	ENUM	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
מרחק אווירי	double	לא	לא	הערך יחושב בשכבה הלוגית. מרחק אווירי מהחברה - חישוב מתמטי פשוט שאינו מושפע מסוג השילוח.
סטטוס הזמנה OrderStatus	ENUM	לא	לא	הערך יחושב בשכבה הלוגית. סטטוס של הזמנה בהיבט של מצב המשלוח האחרון על סמך הנתונים הקיימים ב- ישות נתונים הזמנה - DO.Order ו- ישות נתונים משלוח - DO.Delivery . ראה פירוט כאן: סטטוס הזמנה (OrderStatus)
סטטוס עמידה בזמנים ScheduleStatus	ENUM	לא	לא	הערך יחושב בשכבה הלוגית. סטטוס של הזמנה בהיבט של עמידה בזמנים של המשלוח האחרון על סמך הנתונים הקיימים ב- ישות נתונים הזמנה - DO.Order ו- ישות נתונים משלוח - DO.Delivery . ובישות התצורה. ראה פירוט כאן: סטטוס עמידה בזמנים (ScheduleStatus)
סך זמן שנותר לסיום ההזמנה	TimeSpan	לא	לא	אם ההזמנה סגורה אז 00:00:00. אחרת, הפרש הזמנים בין זמן האספקה המירבי של ההזמנה לבין הזמן הנוכחי של שעות המערכת.
סך זמן השלמת הטיפול	TimeSpan	לא	לא	אם ההזמנה פתוחה אז 00:00:00. אחרת, חישוב ההפרש שבין זמן פתיחת ההזמנה לבין זמן סיום המשלוח האחרון.
סך משלוחים	int	לא	לא	אם עדיין לא בוצע אף משלוח עבור ההזמנה - אז 0. סך המשלוחים הקיימים עבור ההזמנה הנוכחית ב- ישות נתונים משלוח - DO.Delivery . כמה פעמים נלקחה לטיפול והוחזרה וכו'.. עד לסטטוס הנוכחי.

ישות נתונים לוגית "הזמנה פתוחה ברשימה" - BO.OpenOrderInList

- מכילה פרטים עבור ישות נתונים לוגית "הזמנה פתוחה ברשימה"
- ישות שהיא לצפיה בלבד
- תופיע כישות בודדת ברשימה של הזמנות פתוחות לבחירה ע"י שליח מסוים, במסכים:
 - מסך "בחירת הזמנה לטיפול" - עבור שליח

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף ישות שהיא לצפיה בלבד ולכן אין בדיקת תקינות
ת"ז שליח CourierId	int	כן	לא	לא יופיע בתצוגה. לקוח מ- ישות נתונים הזמנה - DO.Order
מספר מזהה רץ של ישות ההזמנה OrderId	int	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
סוג הזמנה	ENUM	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
נפח, משקל, שביר וכדומה	תכונות חובה, אך לשיקולכם		לא	לקוח מ- ישות נתונים הזמנה - DO.Order
כתובת מלאה של ההזמנה	string	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
מרחק אווירי	double	לא	לא	הערך יחושב בשכבה הלוגית. מרחק אווירי מהחברה - חישוב מתמטי פשוט שאינו מושפע מסוג השילוח.
מרחק בפועל	double	כן	לא	הערך יחושב בשכבה הלוגית. מרחק מחושב בק"מ של כתובת ההזמנה מכתובת החברה. יחושב על פי סוג השילוח שמסתכל כרגע על רשימת ההזמנות הפתוחות ורוצה לבחור אחת מהן לטיפול. ראה פירוט כיצד לחשב מרחק, כאן: חישוב מרחק הזמנה בפועל במידה והשליח יבחר לטפל בהזמנה זו אז אותו ערך של מרחק יישמר בשכבת הנתונים בתכונת המרחק בפועל ב- ישות נתונים משלוח - DO.Delivery . כל עוד חישוב המרחק לא הושלם הערך יהיה null. בשלב 7 של הפרויקט החישוב יתבצע בצורה סינכרונית וגם (מכיוון שהרשימה בשלב זה תוחזר כחלק משאילתא שתופעל בתהליכון נפרד).
הערכת זמן בפועל	TimeSpan	כן	לא	הערך יחושב בשכבה הלוגית. שילוב של המהירות הממוצעת ע"פ סוג השילוח ושל חישוב המרחק בפועל. כל עוד חישוב המרחק לא הושלם הערך יהיה null.
סטטוס עמידה בזמנים ScheduleStatus	ENUM	לא	לא	הערך יחושב בשכבה הלוגית. סטטוס של הזמנה בהיבט של עמידה בזמנים של המשלוח האחרון על סמך הנתונים הקיימים ב- ישות נתונים הזמנה - DO.Order ו- ישות נתונים משלוח - DO.Delivery . ובישות התצורה. ראה פירוט כאן: סטטוס עמידה בזמנים (ScheduleStatus)
סך זמן שנותר לסיום ההזמנה	TimeSpan	לא	לא	הפרש הזמנים בין זמן האספקה המירבי של ההזמנה לבין הזמן הנוכחי של שעות המערכת.
זמן אספקה מירבי	DateTime	לא	לא	הערך יחושב בשכבה הלוגית. זמן (תאריך ושעה) אספקה מירבי בהתבסס על טווח האספקה המירבי שהחברה התחייבה אליו ועל הזמן שבו ההזמנה נפתחה על ידי המנהל (בלי להתחשב בסוג השילוח). ראה פירוט כאן: זמנים בחיי הזמנה

ישות נתונים לוגית "משלוח סגור ברשימה" - BO.ClosedDeliveryInList

- מכילה פרטים עבור ישות נתונים לוגית "משלוח סגור ברשימה"
- ישות שהיא לצפיה בלבד
- תופיע כישות בודדת ברשימה של הזמנות סגורות של שליח, במסכים:
 - מסך "היסטוריית משלוחים של שליח" - עבור שליח

שם התכונה	טיפוס התכונה	האם יכול להיות null	האם ניתן לעדכון	הסבר נוסף ישות שהיא לצפיה בלבד ולכן אין בדיקת תקינות
מספר מזהה רץ של ישות המשלוח DeliveryId	int	לא	לא	לא יופיע בתצוגה. לקוח מ- ישות נתונים משלוח - DO.Delivery
מספר מזהה רץ של ישות ההזמנה OrderId	int	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
סוג הזמנה	ENUM	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
כתובת מלאה של ההזמנה	string	לא	לא	לקוח מ- ישות נתונים הזמנה - DO.Order
סוג השילוח	ENUM	לא	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
מרחק בפועל	double	כן	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery
סך זמן השלמת הטיפול	TimeSpan	לא	לא	חישוב ההפרש שבין זמן תחילת המשלוח לבין זמן סיום המשלוח
סוג סיום המשלוח	ENUM	כן	לא	לקוח מ- ישות נתונים משלוח - DO.Delivery

ישויות שירות לוגיות (חוזי שירות - ממשקים) - BIApi

השכבה הלוגית תכיל ממשק לוגי ראשי בשם IBI: שירכז את הגישה לכל ישויות השירות הלוגיות (ממשקים), שיפורטו מיד. בנוסף, השכבה הלוגית תכיל 3 ישויות שירות לוגיות (ממשקים):

- חוזה שירות שליח ICourier
- חוזה שירות הזמנה IOrder
- חוזה שירות של ניהול IAdmin

בסעיף זה נפרט עבור כל ישויות שירות לוגיות (ממשק) - הצעה של רשימת המתודות שעליכם להגדיר עבורה ואת משמעותן. מתודות אלו הן מתודות ממשק ציבוריות שיופעלו בסופו של דבר דרך התצוגה או דרך BILTest. לא כל המתודות מצויינות ותוכלו להוסיף מתודות על פי הצורך.

הערה חשובה:

המתודות שאמורות להופיע בפרק זה בכל חוזי השירות, הן רק מתודות ציבוריות שאמורות להיות מופעלות דרך התצוגה או דרך BILTest. כל שאר המתודות, יפורטו בפרק הבא והן מתודות סטטיות internal לשימוש פנימי כמתודות עזר (עבור קוד שחוזר על עצמו) או מתודות תקופתיות לעדכון מצב. מתודות אלו יוגדרו במחלקת עזר סטטית פנימית לניהול לוגי של אותה ישות.

הנחיות כלליות לגבי המתודות של ישויות השירות הלוגיות

- כל המתודות של ממשקי כל הישויות יקבלו בפרמטר הראשון את ת"ז של המבקש
 - אם המבקש אינו ראשי לבצע את הבקשה, תיזרק חריגה בהתאם
 - בחלק מהמתודות זה רק המנהל
 - בחלק מהמתודות זה רק השליח הספציפי
 - בחלק מהמתודות זה או המנהל או השליח הספציפי
- בכל מימושי מתודות של ממשקי השירות של כל הישויות:
 - יתבצעו רק בדיקות תקינות פורמט ותווכי ערכים כפי שהם יתבצעו גם בשכבת התצוגה בהמשך
 - חל איסור מוחלט להשתמש בטיפוסים של ישויות נתונים (DO) ולפנות לשכבת הנתונים (DAL) בדרך כלשהי בתוך מימושים של ממשקי כל ישויות השירות הלוגיות
 - יזומנו מתודות העזר המתאימות של מחלקות העזר המתאימות XxxManager על מנת לקבל את התוצאה הנדרשת
 - אם נדרש, ימיינו את התוצאה לפני החזרתה
- מתודות שבמחלקות העזר המתאימות XxxManager:
 - מותר בהן לעבוד עם ישויות DO ולפנות לשכבת הנתונים DAL
 - הן יבצעו כל הבדיקות הלוגיות, לוגיקת הרכבת האובייקטים של הישויות הלוגיות, ועיבוד הנתונים
 - הן יחזירו את התוצאות (כשנדרש) כאובייקטים או אוספים כלליים (<IEnumerable>) של ישויות לוגיות בלבד
- כל ההנחיות הבאות מדברות בלשון מתודות ישויות שירות לוגיות, אך מתייחסות לפונקציונליות הכוללת יחד עם מה שמתבצע במתודות העזר במחלקות XxxManager המתאימות שמזמנות בתוך מימושי מתודות של ממשקי השירות הלוגיות
- בכל מתודה של בקשת אוסף/רשימה של אובייקטים של כל ישות לוגית:
 - פונה בבקשות לשכבת הנתונים (ReadAll) כדי לקבל את פרטי הישויות הרלוונטיות
 - מרכיבה מהנתונים האלה את אוסף האובייקטים של הישות הלוגית הנדרשת
- בכל מתודה של בקשת פרטים או עדכון פרטים של אובייקט או מתודות ייחודיות נוספות מתאימות שזקוקות לנתונים משכבת הנתונים, של כל ישות לוגית:
 - פונה בבקשות מתאימות לשכבת הנתונים (Read) כדי לקבל פרטים עבור אובייקט הישות המבוקש מישויות נתונים המתאימות
 - במידה ולא קיים בשכבת הנתונים אובייקט הישות עם המזהה הדרוש - תיזרק חריגה מתאימה של שכבה לוגית
- בכל מתודה של בקשת פרטים של כל ישות לוגית:
 - מייצרת (מהפרטים שהתקבלו) אובייקט מטיפוס הישות הלוגית המתאים להחזרה
- בכל מתודה של הוספה או של עדכון פרטים של כל ישות לוגית, יש לבדוק (שוב - אין לסמוך על בדיקות שכבת התצוגה) את תקינות כל הערכים מבחינת פורמט. למשל:
 - טלפון סלולרי - מחרוזת בת 10 ספרות בדיוק שמתחילה בספרה 0
 - אימייל במבנה תקין
 - ערכים באורך נדרש
- בכל מתודה של הוספה של כל ישות לוגית, לאחר השלמת כל הבדיקות הלוגיות:
 - מכל הפרטים שהתקבלו, מייצרת את האובייקט להחזרה עם כל המידע הדרוש
 - מתוך פרטי האובייקט הלוגי, מייצרת אובייקט חדש מטיפוס ישות הנתונים המתאימה

- מבצעת ניסיון בקשת **הוספת** אובייקט חדש (Create) עבור הישות המתאימה **בשכבת הנתונים**
- במידה **וכבר קיים** אובייקט ישות הנתונים עם המזהה - תיזרק חריגה מתאימה של שכבה **לוגית**
- בכל מתודה של **עדכון פרטים** של כל **ישות לוגית**, לאחר בדיקת תקינות הפורמטים כלעיל:
 - יש לבקש את פרטי אובייקט **ישות הנתונים** המתאימה משכבת הנתונים
 - יש לבדוק שהערכים שהשתנו מותרים לשינוי (בהתאם למבקש) ואת **תקינות הערכים מבחינה לוגית**. למשל:
 - עבור **שליח**: רק המנהל יוכל לעדכן האם השליח **פעיל** או **לא**
 - עבור **הזמנה**: השתנתה הכתובת והיא תקינה - כתובת אמיתית בעולם שיש לה קווי אורך ורוחב (בעקבותיה יתעדכנו גם תכונת קווי האורך והרוחב באובייקט של הישות הלוגית וכתוצאה - באובייקט של ישות הנתונים המתאימה)
 - מייצרת אובייקט מטיפוס ישות הנתונים מתאימה מאובייקט הנתונים שהתקבל קודם לכן עם כל שינוי הערכים בהתאם
 - מבצעת ניסיון בקשת עדכון (Update) של אובייקט ישות הנתונים המתאימה
- בכל מתודה של **כל ישות לוגית** שבודקת חוקיות לוגית של הבקשה - במידה והבקשה לא חוקית - נזרקת חריגה **לוגית מתאימה**
- **עבור שלבים מתקדמים (שלב 5 ומעלה)** - במתודות רבות העוסקות בעדכון ישויות אשר משפיע על מסכים אחרים שפתוחים כרגע, יהיה צורך "להודיע" על עדכון למסכים הרלוונטיים כדי שיתעדכנו באופן אוטומטי
 - זה קשור לשימוש בתבנית עיצוב **משקיף (Observer)** בשכבת התצוגה והשכבה הלוגית
 - **למשל**: כאשר פתוח **מסך בחירת הזמנה לטיפול** עבור שליח המציג את רשימת ההזמנות הפתוחות המותאמות לאותו שליח. אם במסך **שליח בודד** מתעדכן **סוג השילוח** ו-**לא מרחק מירבי אישי** - אזי מתודת העדכון "תודיע" לשכבת התצוגה (למסך **בחירת הזמנה לטיפול**) על השינוי, כך ששכבת התצוגה תוכל לעדכן את רשימת ההזמנות בהתאם לעדכון

ישות שירות לוגית "שליח" - Courier | הצעות למתודות

מתודת כניסה למערכת: (בשימוש ב: מסך כניסה למערכת)

- מקבלת שם משתמש (ת.ז) **ואסימא (תוספת)**
- אם שם המשתמש (ת.ז) אינו ריק - מזמנת את המתודה המתאימה של CourierManager שמסלימה את העבודה כדלקמן
 - מחזירה את תפקיד המשתמש (מנהל או שליח)
 - אם המשתמש לא קיים **או הסיסמה לא נכונה** זורקת חריגה מתאימה

מתודת בקשת רשימת שליחים: (בשימוש ב: מסך "ניהול שליחים")

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - ערך בוליאני מתאפס (nullable) **לסינון** שליחים פעילים ולא פעילים
 - ערך ENUM מתאפס (nullable) לבחירת תכונה בישות "שליח ברשימה", **למינון**
- מחזירה אוסף של **ישות נתונים לוגית "שליח ברשימה" - BO.CourierInList**
- הרשימה המוחזרת תהיה מסוננת כדלקמן:
 - אם בפרמטר השני הועבר ערך null יוחזר אוסף כללי מלא וללא סינון
 - אחרת, יוחזר אוסף כללי מסונן לפי שליחים פעילים או לא פעילים בהתאם לערך הבולאני
 - אם בפרמטר השלישי הועבר ערך null - הרשימה תמוין לפי ת"ז
 - אחרת, תוחזר רשימה שתמוין לפי התכונה שנבחרה למיון
- **אם אתם מעוניינים** ברשימה שמסוננת גם לפי תכונה נוספת, אפשר לבחור באחת משתי השיטות הבאות:
 - הגדרת מתודה נוספת לבקשת רשימה מסוננת לפי אותה תכונה
 - שכלול המתודה הקיימת כך שתקבל פרמטר נוסף לבחירת אותה תכונה לסינון

מתודת בקשת פרטי שליח: (בשימוש ב: מסך "שליח ראשי", מסך "ניהול שליח בודד")

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - מזהה שליח - מספר עם ערך ת"ז
- מחזירה אובייקט מטיפוס **ישות נתונים לוגית שליח - BO.Courier**
- פונה בבקשה לשכבת הנתונים (Read) **גם** כדי לקבל פרטים של ההזמנה בטיפול שליח (במידה וקיימת)

מתודת עדכון פרטי שליח: (בשימוש ב: מסך "שליח ראשי", מסך "ניהול שליח בודד")

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - אובייקט קיים לעדכון - מטיפוס **ישות נתונים לוגית שליח - BO.Courier**

■ האובייקט מלא כבר בכל הערכים

● לא מחזירה ערך

מתודת בקשת מחיקת שליח: (בשימוש ב: מסך "ניהול שליח בודד")

● מקבלת פרמטרים:

○ ת"ז של המבקש

○ מזהה שליח - מספר עם ערך ת"ז

● לא מחזירה ערך

● פונה בבקשה לשכבת הנתונים כדי לבדוק אם ניתן למחוק את השליח

○ **שליח יוכל להימחק** בתנאי שהוא לא מטפל כרגע או לעולם לא טיפל באף הזמנה

○ במידה ולא ניתן למחוק את השליח זורקת חריגה מתאימה לכיוון שכבת התצוגה

● מבצעת ניסיון בקשת מחיקה (Delete) של השליח משכבת נתונים (שליחה של ת"ז למתודה Delete)

מתודת הוספת שליח: (בשימוש ב: מסך "הוספת שליח")

● מקבלת פרמטרים:

○ ת"ז של המבקש

○ אובייקט חדש להוספה מטיפוס ישות נתונים לוגית שליח - BO.Courier

● לא מחזירה ערך

ישות שירות לוגית "הזמנה" | Order - הצעות למתודות

מתודת בקשת סיכום כמויות הזמנות: (בשימוש ב: מסך "ניהול ראשי")

● מקבלת ת"ז של המבקש

● מחזירה מערך כמויות של הזמנות לפי כל סוגי הסטטוסים

● יצירת המערך ע"י שילוב של סטטוס הזמנה וסטטוס עמידה בזמנים

● בכל תא במערך באינדקס i, תשב כמות ההזמנות שערך הסטטוס שלהן שווה לסוג הסטטוס ה-i במספר

● חובה להשתמש בפונקציית הרחבה **GroupBy** או בשאילתות של **group by** של **linq**

מתודת בקשת רשימת הזמנות : (בשימוש ב: מסך "ניהול הזמנות")

● מקבלת פרמטרים:

○ ת"ז של המבקש

○ ערך ENUM מתאפס (nullable) לבחירת תכונה בישות "הזמנה ברשימה" לסינון

○ ערך מטיפוס **object?** - טיפוס כלשהו מתאפס (nullable)

○ ערך ENUM מתאפס (nullable) לבחירת תכונה בישות "הזמנה ברשימה" למיון

● מחזירה אוסף של ישות נתונים לוגית "הזמנה ברשימה" - BO.OrderInList

● הרשימה המוחזרת תהיה מסוננת וממויינת כדלקמן:

○ כל הזמנה תופיע ברשימה **רק פעם אחת** עם המשלוח האחרון שלה (אם קיים)

○ אם בפרמטר השני הועבר ערך null - תוחזר **רשימת** ההזמנות ללא סינון

■ אחרת, תוחזר רשימה מסוננת לפי שוויון ערך התכונה שנבחרה לסינון לערך הפרמטר השלישי

○ אם בפרמטר **השלישי הרביעי** מועבר ערך null - תוחזר רשימה ממויינת לפי **סטטוס הזמנה**

■ אחרת, תוחזר רשימה ממויינת לפי התכונה שנבחרה למיון

מתודת בקשת פרטי הזמנה: (בשימוש ב: מסך "ניהול הזמנה בודדת")

● מקבלת פרמטרים:

○ ת"ז של המבקש

○ מזהה הזמנה

● מחזירה אובייקט מטיפוס ישות נתונים לוגית הזמנה - BO.Order

מתודת עדכון פרטי הזמנה: (בשימוש ב: מסך "ניהול הזמנה בודדת")

● מקבלת פרמטרים:

○ ת"ז של המבקש

○ אובייקט קיים לעדכון - מטיפוס ישות נתונים לוגית הזמנה - BO.Order

● לא מחזירה ערך

מתודת ביטול הזמנה: (בשימוש ב: מסך "ניהול הזמנות" או: מסך "ניהול הזמנה בודדת")

● מקבלת פרמטרים:

○ ת"ז של המבקש

- מספר מזהה ההזמנה לביטול
- לא מחזירה ערך
- בודקת שההזמנה פתוחה או בטיפול אך לא סופקה עדיין, אחרת הבקשה לא חוקית
- אם ההזמנה פתוחה
- מבצעת ניסיון הוספת (Create) ישות "משלוח מדומה"
- זמן תחילת משלוח וזמן סיום משלוח יהיו זהים ויתעדכנו לפי שעון המערכת
- סוג סיום משלוח יהיה בוטל
- ת"ז של השליח יהיה אפס
- אם ההזמנה בטיפול
- מבצעת ניסיון בקשת עדכון (Update) של ישות נתונים משלוח - DO.Delivery המתאימה משכבת נתונים
- זמן סיום משלוח יתעדכן לפי שעון המערכת
- סוג סיום המשלוח בוטל

מתודת מחיקת הזמנה: (לא בשימוש בתצוגה אך רק בתוכנית הבדיקה BTest)

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - מזהה הזמנה
- לא מחזירה ערך
- זורקת חריגה לכיוון שכבת התצוגה שלא ניתן כלל למחוק הזמנות במערכת

מתודת הוספת הזמנה: (בשימוש ב: מסך "הוספת הזמנה")

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - אובייקט חדש להוספה - מטיפוס ישות נתונים לוגית הזמנה - BO.Order
- לא מחזירה ערך
- נ.ב. המספר המזהה של ההזמנה יוצר אוטומטית בתוך שכבת הנתונים

מתודת סיום טיפול בהזמנה: (בשימוש ב: מסך "שליח ראשי")

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - מזהה שליח - מספר עם ערך ת"ז
 - מספר מזהה המשלוח - מספר המשלוח של ההזמנה שעליה הוא רוצה לדווח על סיום טיפול
- לא מחזירה ערך
- בודקת את חוקיות הבקשה - המבקש הינו השליח המבצע את המשלוח
- מבצעת ניסיון בקשת עדכון (Update) של ישות נתונים משלוח - DO.Delivery המתאימה משכבת נתונים
 - סוג סיום המשלוח "סופק"
 - "זמן סיום המשלוח" יתעדכן לפי שעון המערכת

מתודת בחירת הזמנה לטיפול: (בשימוש ב: מסך "בחירת הזמנה לטיפול")

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - מזהה שליח - מספר עם ערך ת"ז
 - מזהה הזמנה שנבחרה לטיפול
- לא מחזירה ערך
- בודקת שהבקשה חוקית כדלהלן (דרך בקשת רשימת המשלוחים במידה וקיימים עם סינון מתאים וביצוע בדיקות נוספות אם נדרש):
 - ההזמנה פתוחה - עוד לא סופקה ולא נמצאת באמצע משלוח
 - במידה והבקשה לסיום טיפול לא חוקית - מזרקה חריגה מתאימה לכיוון שכבת התצוגה
- מבצעת ניסיון הוספה (Create) של ישות נתונים משלוח - DO.Delivery חדשה משכבת נתונים
 - עם מספר מזהה של ההזמנה ות"ז של השליח
 - "זמן כניסה לטיפול" יתעדכן לפי שעון המערכת
 - "זמן סיום המשלוח" ו"סוג הטיפול" יהיו בינתיים null

מתודת בקשת רשימת הזמנות סגורות שטופלו על ידי שליח: (בשימוש ב: מסך "היסטוריית משלוחים של שליח")

- מקבלת פרמטרים:
 - ת"ז של המבקש

- מזהה שליח - מספר עם ערך ת"ז
- ערך ENUM מתאפס (nullable) לבחירת סוג הזמנה (תכונה בישות "משלוח סגור ברשימה"), לסינון
- ערך ENUM מתאפס (nullable) לבחירת תכונה בישות "משלוח סגור ברשימה", למיין
- מחזירה אוסף ממין של ישות נתונים לוגית "משלוח סגור ברשימה" - BO.ClosedDeliveryInList
 - כל ההזמנות הסגורות של אותו שליח מכל סוג סיום משלוח כלשהו
 - הרשימה המוחזרת תהיה מסוננת וממויינת כדלקמן:
 - אם בפרמטר השלישי הועבר ערך סטטוס null, תוחזר רשימה מלאה
 - אחרת, הרשימה מסוננת לפי סוג הזמנה
 - אם בפרמטר הרביעי הועבר ערך null - תוחזר רשימה ממויינת לפי סוג סיום משלוח סטטוס עמידה בזמנים
 - אחרת, תוחזר רשימה ממויינת לפי ערך התכונה המסוימת

מתודת בקשת רשימת הזמנות פתוחות לבחירה על ידי שליח: (בשימוש ב: מסך "בחירת הזמנה לטיפול")

- מקבלת פרמטרים:
 - ת"ז של המבקש
 - מזהה שליח - מספר עם ערך ת"ז - שעבורו מחזירים את רשימת ההזמנות הפתוחות לבחירה ואת מרחקן ממרחקו הנוכחי
 - ערך ENUM מתאפס (nullable) לבחירת סוג הזמנה (תכונה בישות "הזמנה פתוחה ברשימה"), לסינון
 - ערך ENUM מתאפס (nullable) לבחירת תכונה בישות "הזמנה פתוחה ברשימה", למיין
 - מחזירה אוסף ממין של ישות נתונים לוגית "הזמנה פתוחה ברשימה" - BO.OpenOrderInList
 - כל ההזמנות הפתוחות שמתאימות לאותו שליח מבחינת המרחק האווירי של כל הזמנה מהשליח
 - ביחס למרחק המירבי האישי של אותו שליח
 - הרשימה המוחזרת תכלול את כל ההזמנות בסטטוס "פתוחה" ותהיה מסוננת וממויינת כדלקמן:
 - אם בפרמטר השלישי הועבר ערך סטטוס null, תוחזר רשימה מלאה של הזמנות פתוחות
 - אחרת, תוחזר רשימה מסוננת לפי סוג הזמנה
 - אם בפרמטר הרביעי הועבר ערך null - תוחזר רשימה ממויינת לפי סטטוס עמידה בזמנים
 - אחרת, תוחזר רשימה ממויינת לפי ערך השדה המסוים

ישות שירות לוגית "ניהול" Admin - הצעות למתודות

מתודת איפוס בסיס נתונים בשם ResetDB: (בשימוש ב: מסך "ניהול ראשי")

- לא מקבלת פרמטרים
- לא מחזירה ערך
- תאפס את כל נתוני התצורה (תחזיר לערך התחלתי של כל נתון תצורה)
- תרוקן את נתוני כל הישויות (תרוקן את כל רשימות הנתונים)

מתודת אתחול בסיס נתונים בשם InitializeDB: (בשימוש ב: מסך "ניהול ראשי")

- לא מקבלת פרמטרים
- לא מחזירה ערך
- תאתחל את את בסיס הנתונים:
- תחילה תאפס את בסיס הנתונים

ולאחר מכן תוסיף נתונים עבור כל הישויות ע"פ דרישות אתחול בסיס הנתונים (תמלא את רשימות הנתונים בערכים התחלתיים)

מתודת בקשת שעון בשם GetClock: (בשימוש ב: מסך "ניהול ראשי")

- לא מקבלת פרמטרים
- מחזירה את ערכו של שעון המערכת מטיפוס DateTime

מתודת קידום שעון בשם ForwardClock: (בשימוש ב: מסך "ניהול ראשי")

- מקבלת פרמטר מסוג ENUM של יחידת הזמן לקידום (דקה, שעה, יום, חודש, שנה)
- תקדם את שעון המערכת ביחידת הזמן המתאימה
- תייצר זמן חדש שיקודם בהתאם לזמן הנוכחי של שעון המערכת ויחידת הזמן שהתקבלה
- ותשלח את הזמן החדש למתודת העזר הסטטית במחלקת העזר לניהול AdminManager.UpdateClock
- למשל אם יחידת הזמן שהתקבלה היא 'דקה' (BO.TimeUnit.MINUTE) אז הזמן החדש שיש ליצור הוא AdminManager.Now.AddMinutes(1)
- והמתודה תיקרא למשל כך: AdminManager.UpdateClock(AdminManager.Now.AddMinutes(1))

מתודת בקשת משתני תצורה בשם `GetConfig`: (בשימוש ב: מסך "ניהול ראשי")

- לא מקבלת פרמטרים
- מחזירה את ערכי משתני התצורה בתוך אובייקט `BO.Config`
- רק את משתני התצורה שרלוונטיים כלפי מעלה לשכבת התצוגה (לא את המשתנים שהם לשימוש פנימי כגון מספרים רצים)

מתודת קביעת משתני תצורה בשם `SetConfig`: (בשימוש ב: מסך "ניהול ראשי")

- מקבלת אובייקט `BO.Config` שמכיל את כל ערכי משתני התצורה
- רק את אלו שמגיעים מכיוון התצוגה ולא את אלו שהם לשימוש פנימי כגון מספרים רצים
- לא מחזירה ערך
- מעדכנת את כל משתני התצורה
- רק אלו שהשתנו מאז העדכון הקודם

מחלקות עזר סטטיות פנימיות ב BL - לניהול לוגי של ישות נתונים

בפרק הקודם, פירטנו את חוזי השירות עבור כל אחת מישויות השירות הלוגיות שהכילו מתודות ממשק

- מתודות ציבוריות שאמורות להיות מופעלות דרך התצוגה או דרך BTest.

בפרק זה, נפרט את מחלקות העזר הסטטיות (בהרשאת internal) לניהול לוגי של ישויות נתונים אשר יכילו מתודות עזר לשימוש פנימי בשכבה הלוגית, עבור:

- קוד שחוזר על עצמו
- קוד בעל לוגיקה מורכבת שכדאי להוציא למתודה נפרדת
- קוד לשימוש פנימי שאינו חלק מהממשק הציבורי (למשל, מתודה לצורך עדכון מצב תקופתי)

כפי שמוסבר בהנחיות שלב 4, יש להוסיף לשכבה הלוגית (פרויקט BL) על פי הצורך, תת ספריה בשם Helpers שתכיל:

- מחלקת Tools - מחלקה עזר סטטית בהרשאת internal עבור מתודות עזר כלליות בשכבה הלוגית
- מחלקת AdminManager - מחלקה עזר סטטית בהרשאת internal עבור מתודות לניהול המערכת
- עבור כל ישות נתונים, על פי הצורך (ראו הנחיות מפורטות בהנחיות שלב 4):
 - הגדירו מחלקת עזר סטטית בהרשאת internal - שם המחלקה יהיה שם הישות בתוספת המילה Manager
 - לדוגמא: OrderManager, CourierManager
 - הגדירו וממשו במחלקה זו מתודות עזר סטטיות בהרשאת internal על פי הצורך (מיד נפרט דוגמאות).

דוגמאות למתודות עזר שיוגדרו במחלקות עזר

להלן דוגמאות למתודות עזר סטטיות בהרשאת internal שנרצה להעביר למתודה נפרדת במחלקת העזר הסטטית המתאימה, כתוצאה מקוד חוזר על עצמו או קוד מורכב:

מתודת עזר לחישוב סטטוס הזמנה:

- מקבלת מספר מזהה של הזמנה
- מחזירה את סטטוס ההזמנה בהתאם לכל המידע בבסיס הנתונים ושעון המערכת
- מתודת עזר סטטית פנימית שניתן להגדיר בתוך מחלקת העזר OrderManager
- תזומן מתוך המחלקה המממשת OrderImplementation

מתודת עזר לחישוב סטטוס עמידה בזמנים הזמנה:

- מקבלת מספר מזהה של הזמנה
- מחזירה את סטטוס עמידה בזמנים של ההזמנה בהתאם לכל המידע בבסיס הנתונים ושעון המערכת
- מתודת עזר סטטית פנימית שניתן להגדיר בתוך מחלקת העזר OrderManager
- תזומן מתוך המחלקה המממשת OrderImplementation

מתודת עזר לחישוב קואורדינטות של כתובת:

- מקבלת כתובת (של הזמנה/שליח)
- מחשבת ומחזירה את הקואורדינטות של הכתובת (קווי אורך ורוחב)
- אפשר להיעזר במתודות הקיימות ברשת הפונות לאתר שנותן שירות כזה בעזרת בקשת רשת.
 - ראה פירוט כאן: [נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות](#)
- תזרוק חריגה במידה והכתובת לא תקינה
- מתודת עזר סטטית פנימית שניתן להגדיר בתוך מחלקת העזר OrderManager\ToolsManager
- תזומן מתוך המחלקה המממשת OrderImplementation

מתודת עזר לחישוב מרחק אווירי בין כתובות:

- מקבלת 2 זוגות של קואורדינטות המייצגות 2 כתובות (כתובת החברה וכתובת של הזמנה)
- מחשבת ומחזירה את המרחק האווירי ביניהן - מתודה שהיא חישוב מתמטי לא מורכב במיוחד שניתן למצוא ברשת
- מתודת עזר סטטית פנימית שניתן להגדיר בתוך מחלקת העזר OrderManager\ToolsManager
- תזומן מתוך המחלקה המממשת OrderImplementation

מתודת עזר לחישוב מרחק נסיעה/הליכה בין כתובות:

- מקבלת 2 זוגות של קואורדינטות המייצגות 2 כתובות (כתובת החברה וכתובת של הזמנה)
- מקבלת את סוג המרחק לחישוב נסיעה או הליכה
- מחשבת ומחזירה את המרחק ביניהן

- אפשר להיעזר במתודות הקיימות ברשת הפונות לאתר שנותן שירות כזה בעזרת בקשת רשת.
- ראה פירוט כאן: [נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות](#)
- מתודת עזר סטטית פנימית שניתן להגדיר בתוך מחלקת העזר ToolsManager\OrderManager
- תזומן מתוך המחלקה המממשת OrderImplementation

להלן דוגמאות למתודות עזר סטטיות בהרשאת internal שנרצה להעביר למתודה נפרדת במחלקת העזר הסטטית המתאימה, כתוצאה מצורך פנימי (ביצוע פעולות עדכון תקופתיות לאחר עדכון השעון):

מתודת עדכון תקופתי - עדכון פעילות שליחים:

- מתודת עזר סטטית פנימית שניתן להגדיר בתוך מחלקת העזר CourierManager
- תזומן מתוך המחלקה AdminManager בכל פעם שהשעון מתעדכן
- לא מקבלת פרמטרים
- מבצעת את הלוגיקה הבאה:
- עוברת על כל השליחים שלא ביצעו משלוחים כלל בטווח זמן חוסר פעילות שהוגדר בישות תצורה, ומעדכנת אותם להיות לא פעילים.
- **בנוסף בשלב 5:** עבור כל שליח שעודכן "תישלח הודעה" למשקיפים על עדכון השליחה הספציפי (אם יש) - (ע"י זימון דלגט ה-event המתאים מ"מילון" [Dictionary] של המשקיפים על שליחים ספציפיים
- **בנוסף בשלב 5:** לאחר השלמת המעבר על כל השליחים, "תישלח הודעה" למשקיפים על עדכון רשימת השליחים - ע"י זימון דלגט ה-event המתאים

מתודת "סימולציית פעילות שליחים":

- מתודה שתהיה בשימוש הסימולטור אשר ימומש בשלב 7
- תוגדר ותמומש בשלב 7 ושם יהיה ניתן לראות את פירוט אודותיה

שכבת התצוגה (שלב 5 ושלב 6)

פירוט המסכים והפונקציונאליות שלהם

בסעיף זה נפרט את כל המסכים הנדרשים לשלב 5 ולשלב 6, שהם שלבי מימוש התצוגה.

את התצוגה נפתח בצורה הדרגתית כך שבשלב 5 נממש 3 מסכים ובשלב 6 נשלים את כל שאר המסכים.

שלב 5 של הפרויקט:

בשלב 5 נניח שהמשתמש הוא מסוג "מנהל" שנכנס ישירות למסך הניהול הראשי וכן נממש עבור המנהל רק את המסכים העוסקים בניהול שליחים ולא את המסכים שעוסקים בניהול ההזמנות.

כלומר, בשלב 5 של הפרויקט, נממש את 3 המסכים הבאים:

- מסך ניהול ראשי - יאפשר את כל הפונקציונאליות הנדרשת חוץ מ:
 - לא כולל סיכום כמויות הזמנות לפי סטטוסים
 - לא כולל התייחסות לסימולטור
 - כפתור ניהול הזמנות לא יוביל כרגע לשום פעולה
- עבור ישות השירות הלוגית "שליח":
 - מסך ניהול שליחים - מכיל רשימת שליחים
 - בשלב 5 נבקש שהמסך יאפשר סינון לפי סוג השילוח
 - בשלב 6 תוכלו להוסיף אפשרות למיון וסינון לפי פרמטרים שתבחרו
 - מסך ניהול שליח בודד (מסך אחד שנפתח ב 2 מצבי תצוגה): צפיה ועדכון של שליח והוספה של שליח חדש

פירוט הצעדים ליצירת המסכים תוכלו לראות במסמך הנחיות הפרויקט לשלב 5.

שלב 6 של הפרויקט:

את שאר המסכים כולל מסך הכניסה למערכת תממשו בשלב 6. על מנת להשלים את שכבת התצוגה המלאה, ראו הנחיות מפורטות במסמך שלב 6.

הערה כללית חשובה לגבי אפשרות עדכון/הוספה במסכי התצוגה:

- פרטים שלא ניתנים לעדכון יוצגו אך יחסמו לעדכון בתצוגה
- לאחר לחיצה על כפתור עדכון/הוספה:
 - בשכבת התצוגה יערכו כל בדיקות התקינות של פורמט הערכים ולא בדיקות לוגיקה
 - הבקשה תשלח לשכבה הלוגית לצורך בדיקות תקינות לוגיות וגם בדיקות חוזרות של פורמט הערכים
 - במידה והבקשה תקינה הערכים יתעדכנו:
 - בבסיס הנתונים
 - בכל המסכים הפתוחים בתצוגה הרלוונטים לעדכון/הוספה הנוכחי
- תבנית עיצוב משקיף (Observer)
 - בכל מקרה, המשתמש יקבל אישור על הצלחה או הודעה על תקלה ולאחר מכן יוכל להמשיך לצפות ב- ולעדכן את הערכים כרצונו

מסך כניסה למערכת

- מסך הכניסה למערכת - עבור משתמש "מנהל" ועבור משתמש "שליח"
- יאפשר להזין ת"ז של משתמש (וגם סיסמא במקרה של בחירת **תוספת** זו)
 - ת"ז של המנהל תהיה שמורה בישות התצורה ות"ז של השליחים יהיו שמורים כחלק מישות השליח.
- לאחר הזנת הפרטים המזהים, יאפשר כניסה למסך הראשי של המערכת על פי סוג המשתמש:
 - שליח רגיל ייכנס ישיר למסך ראשי של שליח "מסך שליח"
 - שליח מנהל יוכל לבחור אם להיכנס ל-"מסך שליח" או ל-"מסך ניהול ראשי"
- מסך הכניסה חייב להישאר נגיש כל הזמן על מנת לאפשר בו זמנית כניסת מנהל אחד (לכל היותר) ומספר (לא מוגבל) של שליחים

מסכים עבור משתמש "מנהל"

מסך "ניהול ראשי"

- נפתח לאחר שאושרה הכניסה למנהל במסך "כניסה למערכת"
- המסך יאפשר:
 - לאפס ולאתחול בסיס הנתונים
 - לנהל את שעות המערכת (משתנה התצורה) - לאתחל את השעון, לקדם את השעון בדקות, שעות, ימים
 - לעדכן את ערכם של כל שאר משתני התצורה של המערכת
 - גישה למסך "ניהול שליחים" (שיציג בתחילה את כל השליחים)
 - גישה למסך "ניהול הזמנות" (שיציג בתחילה את כל ההזמנות)
 - לקבוע את קצב התקדמות שעות הסימולטור
 - להפעיל/לעצור את הסימולטור
- המסך יציג גם סיכום כמויות הזמנות לפי כל סוגי הסטטוסים (שילוב של סטטוס הזמנה וסטטוס עמידה בזמנים)
 - לחיצה על תצוגת כמות מתאימה תוביל למסך "רשימת הזמנות" (שיציג רשימה מסוננת מראש בהתאם למה שהמנהל רוצה לראות ע"פ הלחיצה)
 - סיכום הכמויות יתעדכן באופן אוטומטי במידה וממסכים אחרים שפתוחים כרגע נעשו שינויים הרלוונטים לשאילתא
- מסך ראשי של מנהל חייב להישאר נגיש בעת פתיחת מסכי משנה של המנהל
 - בו זמנית יכולים להיות לא יותר ממסך אחד של ניהול הזמנות וממסך אחד של ניהול שליחים (יתכן ששניהם פתוחים בו זמנית, אך לא יותר מאחד מכל אחד מהם)

מסך "ניהול שליחים"

- נפתח לאחר לחיצת מנהל על כפתור "ניהול שליחים" במסך "ניהול ראשי"
- מנהל יכול לראות את רשימת כל השליחים
 - יוכל לסנן ולאו למיין את הרשימה לפי ערכי שדות כאלה או אחרים לפי שיקול דעתכם
- המידע שיוצג ברשימה יהיה חלקי בלבד, יוצגו רק פרטים עיקריים של השליחים
 - כל שורה ברשימה תהיה מטיפוס [ישות נתונים לוגית "שליח ברשימה" - BO.CourierInList](#)
 - הרשימה תתעדכן באופן אוטומטי במידה וממסכים אחרים שפתוחים כרגע נעשו שינויים הרלוונטים לשאילתא
- לחיצה כפולה על שליח שנבחר מהרשימה תפתח מסך "ניהול שליח בודד" עבור אותו שליח
- יש להוסיף בשורה כפתור "מחק" למחיקת השליח הנבחר
 - בתנאי שניתן למחוק את השליח (לא מבצע או לא ביצע אף משלוח לעולם)
 - תוספת - הכפתור יופיע רק בתנאי שאפשר למחוק את השליח
- המסך יכיל גם כפתור "הוספת שליח" אשר לחיצה עליו תוביל לפתיחת מסך "הוספת שליח"
 - לאחר ההוספה השליח החדש יופיע מיד גם הוא ברשימת השליחים
- בזמן שפתוח מסך לצורך צפייה/עדכון/מחיקה/הוספה של שליח, אין למנוע גישה למסך "ניהול שליחים"

מסך "ניהול שליח בודד"

- נפתח לאחר לחיצה כפולה של מנהל על שליח מסוים ברשימת השליחים במסך "ניהול שליחים"
- שליח מיוצג במסך זה על ידי הטיפוס [ישות נתונים לוגית שליח - BO.Courier](#)
 - הכוללת בתוכה את הטיפוס [ישות נתונים לוגית "הזמנה בטיפול שליח" - BO.OrderInProgress](#) במידה וקיימת
- המסך מאפשר:
 - צפייה בפרטי השליח הנבחר בהרחבה
 - צפייה (בלבד) בפרטי ההזמנה שבטיפולו (במידה וקיימת)
 - מחיקת השליח (בתנאי שניתן למחוק אותו) ע"י לחיצה על כפתור "מחק"
 - עדכון פרטי השליח - רק מה שניתן לעדכן (וכן יש להתייחס אם הוא כרגע באמצע משלוח או לא)
- לאחר עדכון הפרטים, המנהל ילחץ על כפתור "עדכן"

מסך "הוספת שליח"

- נפתח לאחר לחיצת מנהל על כפתור "הוספת שליח" במסך "ניהול שליחים"
- שליח מיוצג במסך זה על ידי הטיפוס [ישות נתונים לוגית שליח - BO.Courier](#)
 - מכיוון שישות השליח זה עתה נוצרת אז אין עבורו עדיין הזמנה בטיפול ולכן תכונת הרשימה שבה ריקה

- המסך מבוסס על מסך "ניהול שליח בודד" אך מאפשר עריכה בשדות שהיו חסומים לעדכון וכעת מאפשרים לצורך הוספת פרטים של שליח חדש. כמו כן, המסך כולל כפתור "הוסף" במקום כפתור "עדכן"

מסך "ניהול הזמנות"

- נפתח לאחר לחיצת מנהל על כפתור "ניהול הזמנות" במסך "ניהול ראשי"
- מנהל יכול לצפות ברשימת כל ההזמנות הקיימות בכל סטטוס שהוא:
 - כל הזמנה תופיע ברשימה רק פעם אחת עם המשלוח האחרון שלה (אם קיים)
 - יכול לסנן וגם למיין את הרשימה לפי ערך של שדות כאלה ואחרים לשיקול דעתכם
 - יכול גם לקבץ לפי סוג הזמנה ו-לאו לפי סטטוס או כל תכונה אחרת - תוספת
- המידע שיוצג ברשימה יהיה חלקי בלבד. לא יוצגו כל פרטי ההזמנה, אלא רק מרכיבים עיקריים שלה
 - כל שורה ברשימה תהיה מטיפוס [ישות נתונים לוגית "הזמנה ברשימה" - BO.OrderInList](#)
 - הרשימה תתעדכן באופן אוטומטי במידה וממסכים אחרים שפתוחים כרגע נעשו שינויים הרלוונטיים לשאילתא
- לחיצה כפולה על הזמנה שנבחרה מהרשימה תפתח מסך "ניהול הזמנה בודדת" עבור אותה הזמנה
- יש להוסיף בכל שורה כפתור "בטל" לביטול ההזמנה הנבחרת
 - הביטול יתאפשר רק בתנאי שהיא לא סגורה (ראה פירוט כאן: [סוגי סיום משלוח](#))
 - הכפתור יופיע רק בתנאי שאפשר לבטל את ההזמנה
 - תוספת - במקרה של ביטול הזמנה שכבר באמצע משלוח ישלח מייל לשליח
- המסך יכול גם כפתור "הוספת הזמנה" אשר לחיצה עליו תוביל לפתיחת מסך "הוספת הזמנה"
 - לאחר ההוספה ההזמנה החדשה תופיע מיד גם היא ברשימת ההזמנות
- בזמן שפתוח מסך לצורך צפייה/עדכון/הוספה של הזמנה, אין למנוע גישה למסך "ניהול הזמנות"

מסך "ניהול הזמנה בודדת"

- נפתח לאחר לחיצה כפולה של מנהל על הזמנה מסוימת ברשימת ההזמנות במסך "ניהול הזמנות"
- הזמנה מיוצגת במסך זה על ידי טיפוס [ישות נתונים לוגית הזמנה - BO.Order](#)
 - הכוללת בתוכה רשימה של ישויות מטיפוס [ישות נתונים לוגית "משלוח ברשימה" - BO.DeliveryPerOrderInList](#) במידה וקיימת
- המסך מאפשר:
 - צפייה בפרטי ההזמנה הנוכחית בהרחבה
 - תכונות קו האורך וקו הרוחב של ההזמנה נגזרות מהכתובת של ההזמנה ואינן מוצגות על המסך
 - צפייה (בלבד) בכל רשימת המשלוחים בעבר ובהווה עבור ההזמנה הנוכחית (במידה וקיימים)
 - ביטול ההזמנה - רק במידה וההזמנה לא סגורה (ראה פירוט כאן: [סוגי סיום משלוח](#))
 - כפתור "ביטול" לא יהיה מאפשר במידה וההזמנה פתוחה או בטיפול אך לא סופקה עדיין
 - לאחר הביטול, במידה וההזמנה הופיעה במסכים פתוחים אחרים - היא תופיע בסטטוס המעודכן ברשימת ההזמנות, ותיעלם ממסך של השליח שטיפל בה (במידה והיה כזה)
 - תוספת - במקרה של ביטול הזמנה שכבר באמצע משלוח ישלח מייל לשליח
 - עדכון פרטי ההזמנה - רק במידה וההזמנה פתוחה ורק מה שניתן לעדכון
 - כפתור "עדכון" לא יהיה מאפשר במידה וההזמנה סגורה
 - לאחר עדכון הפרטים, המנהל ילחץ על כפתור עדכון

מסך "הוספת הזמנה"

- נפתח לאחר לחיצת מנהל על כפתור "הוספת הזמנה" במסך "ניהול הזמנות"
 - הזמנה מיוצגת במסך זה על ידי טיפוס [ישות נתונים לוגית הזמנה - BO.Order](#)
 - מכיוון שישות ההזמנה זה עתה נוצרת אז אין עבורה עדיין משלוחים ולכן תכונת הרשימה שבה ריקה
- המסך מבוסס על מסך "ניהול הזמנה בודדת" אך מאפשר עריכה בשדות שהיו חסומים לעדכון וכעת מאפשרים לצורך הוספת פרטים של הזמנה חדשה. כמו כן, המסך כולל כפתור "הוסף" במקום כפתור "עדכן"
- המנהל יערוך את פרטי ההזמנה החדשה וילחץ על כפתור "הוסף". לאחר מכן:
 - במידה והבקשה תקינה - ההזמנה החדשה תתווסף לבסיס הנתונים ותופיע ברשימת ההזמנות הפתוחות לבחירה, במסכים המתאימים
 - בכל מקרה המשתמש יקבל אישור על הצלחה או הודעה על תקלה בהוספת השליח ולאחר מכן החלון ייסגר
- תוספת - כשנוספת הזמנה, המערכת תשלח אימייל עם פרטי ההזמנה לכל השליחים הנמצאים במרחק מתאים על פי דרישת השליח. שליח שקיבל אימייל יוכל להיכנס למערכת ולבחור לטפל בהזמנה.

מסכים עבור משתמש "שליח"

מסך "שליח"

- נפתח לאחר שאושרה הכניסה לשליח במסך "כניסה למערכת"
- שליח מיוצג במסך זה על ידי הטיפוס [ישות נתונים לוגית שליח - BO.Courier](#)
 - הכוללת בתוכה את הטיפוס [ישות נתונים לוגית "הזמנה בטיפול שליח" - BO.OrderInProgress](#) במידה וקיימת
- המסך מאפשר:
 - צפייה בפרטי השליח בהרחבה
 - עדכון פרטי השליח - רק מה שניתן לעדכון ויש להפעיל לוגיקה, למשל:
 - בעדכון המרחק המירבי האישי הנדרש לו לקבלת הזמנה יש לוודא שהוא קטן שווה למרחק המירבי של החברה
 - בעדכון סוג השילוח יש להתייחס אם הוא כרגע באמצע משלוח או לא (יש הזמנה בטיפול)
 - עבור שליח שאינו פעיל - כפתור "בחירת הזמנה" לא יהיה מאופשר
 - לאחר עדכון הפרטים, השליח ילחץ על כפתור עדכן
 - צפייה (בלבד) בפרטי ההזמנה שבטיפולו (במידה וקיימת)
 - במידה ואין כרגע הזמנה בטיפולו של אותו שליח:
 - לא יופיעו על המסך פרטי הזמנה כלשהי ולא תהיה אפשרות לדווח על סיום טיפול בהזמנה
 - כפתור "בחירת הזמנה" יהיה מאופשר ולחיצה עליו תאפשר כניסה למסך "בחירת הזמנה לטיפול"
 - במידה ויש כרגע הזמנה בטיפולו של אותו שליח:
 - שליח יוכל לצפות (צפייה בלבד) בכל פרטי ההזמנה הנוכחית אשר בטיפולו
 - כפתור "בחירת הזמנה" לא יהיה מאופשר
 - שליח ידווח על סוג סיום המשלוח ואז ילחץ על כפתור "סיים טיפול"
 - לאחר מכן ההזמנה כבר לא תופיע במסך הנוכחי.
- המסך יכול גם כפתור "היסטוריית משלוחים" אשר לחיצה עליו תוביל לפתיחת מסך "היסטוריית משלוחים של שליח"
- שליח לא יוכל למחוק את עצמו או לדווח על עצמו כפעיל/לא פעיל - אלא רק דרך המנהל

מסך "היסטוריית משלוחים של שליח"

- נפתח לאחר לחיצת שלח על כפתור "היסטוריית הזמנות" במסך "שליח ראשי"
- שליח יוכל לצפות בכל היסטוריית המשלוחים הסגורים שלו בלבד בכל סוג סיום טיפול שהוא
 - יוכל לסנן ולמייין את הרשימה לפי ערך של שדות כאלה ואחרים לשיקול דעתכם
- המידע שיוצג ברשימה יהיה חלקי בלבד. לא יוצגו כל פרטי ההזמנה, אלא רק מרכיבים עיקריים שלה
 - כל שורה ברשימה תהיה מטיפוס [ישות נתונים לוגית "משלוח סגור ברשימה" - BO.ClosedDeliveryInList](#)
 - הרשימה תתעדכן באופן אוטומטי במידה וממסכים אחרים שפתוחים כרגע נעשו שינויים הרלוונטים לשאילתא

מסך "בחירת הזמנה לטיפול"

- נפתח לאחר לחיצת שליח על כפתור "בחירת הזמנה" במסך "שליח ראשי"
- שליח יוכל לצפות ברשימת ההזמנות הפתוחות אשר נמצאות במרחק שהוא פחות מהמרחק האווירי המירבי האישי שהגדיר עבור עצמו, ולבחור מהן הזמנה לטיפול
 - יוכל לסנן ולמייין את הרשימה לפי ערך של שדות כאלה ואחרים לשיקול דעתכם
- המידע שיוצג ברשימה יהיה חלקי בלבד. יוצגו רק מרכיבים עיקריים שיאפשרו לשליח לקבל החלטה
 - כל שורה ברשימה תהיה מטיפוס [ישות נתונים לוגית "הזמנה פתוחה ברשימה" - BO.OpenOrderInList](#)
 - הרשימה תתעדכן באופן אוטומטי במידה וממסכים אחרים שפתוחים כרגע נעשו שינויים הרלוונטים לשאילתא
- יש להוסיף בכל שורה כפתור "איסוף" שלחיצה עליו תאפשר לשליח לבחור לטפל בהזמנה שבשורה זו
- ברגע ששליח בוחר לטפל בהזמנה מהרשימה:
 - ההזמנה כבר לא תופיע ברשימת ההזמנות הפתוחות, כיוון שהיא עוברת לסטטוס "בטיפול" של אותו שליח
 - אם ירצה לעדכן שהוא סיים לטפל בה, יוכל לחזור למסך "פרטי שליח" ולדווח על כך
 - תוספת - כששליח בוחר לטפל בהזמנה, ישלח אליו אימייל עם פרטי ההזמנה בהרחבה
- בכל בחירה (לחיצה אחת עם העכבר) של אחת ההזמנות הפתוחות ברשימה:

- **תוספת** - תוצג על המסך מפה שמראה את מיקום החברה וגם את מיקום ההזמנה הנבחרת מהרשימה
- **תוספת** - יוצגו על המפה גם קווים אוויריים בין 2 המיקומים
- **תוספת** - יוצגו על המפה גם מסלולי נסיעה/הליכה בין מיקומו למיקום ההזמנות בהתאם לסוג השילוח

סימולטור (שלב 7)

עדיין לא מעודכן לתשפ"ו

חישוב קואורדינטות של כתובת בצורה אסינכרונית

בשלב 7, יש לעדכן את בקשת הרשת לחישוב של הקואורדינטות בהינתן כתובת של החברה/הזמנה בזמן הוספה/עדכון של ישות בודדת.

עד כה חישובתם את הקואורדינטות בצורה **סינכרונית** ובשלב 7 יש לבצע את החישוב בצורה **אסינכרונית** בעזרת Task (async/await). ראו פירוט כיצד לעשות זאת בהנחיות לשלב 7.

לגבי בקשת הרשת לחישוב מרחק בפועל הליכה/נסיעה:

- אם החישוב הוא עבור תכונה של ישות בודדת **שאינה** חלק מרשימה - החישוב יתבצע בצורה **אסינכרונית**
- אם החישוב הוא עבור תכונה של ישות בודדת **שהיא חלק מרשימה** - יש להשאיר את החישוב בצורה **סינכרונית** - כיוון שאנו מחשבים אותם כחלק משאילתא שמחזירה את הישיות - שמבוצעת בשלב 7 על תהליכון נפרד דרך Task.Run().
- לדוגמא החזרת רשימת ההזמנות הפתוחות עבור שליח

סימולציית (הדמיית) פעילות המערכת - סימולציית פעילות השליחים

בשלב 4, הגדרתם ומימשתם במחלקת העזר CourierManager, **מתודה לעדכון תקופתי** שמזמנת מתוך המחלקה AdminManager, לאחר כל עדכון השעון (**מתודת עדכון כל השליחים הלא הפעילים**).

- יש לעבור על ההנחיות בשלב 7 שנוגעות לעדכון המתודה הזו והתאמתה לעבודה עם סימולטור כעת, **בשלב 7**, נתבקשתם להגדיר **מתודה לצורך סימולציה של פעילות המערכת**:
- המתודה תדמה פעילות של שליחים לאורך חיי המערכת ולכן שמה: **מתודת סימולציית פעילות שליחים**
- המתודה תוגדר במחלקת CourierManager כ- internal static void
- המתודה **תזמן בצורה אסינכרונית, פעם אחת בכל שניה, מתוך תהליכון הסימולטור**.
- בהנחיות שלב 7, תועלו למצוא פירוט לגבי **כיצד והיכן מזמנים אותה מתוך תהליכון הסימולטור**.

מתודת "סימולציית פעילות שליחים"

- המתודה מוגדרת כ**סינכרונית** אך תזמן **פעם אחת בכל שניה בצורה אסינכרונית** בעזרת Task.Run()
- יש לממש את המתודה על פי כל ההנחיות המפורטות בשלב 7. נחזור על עיקרי הדברים:
 - יש לעטוף בבלוק של lock כל פניה ל DAL וכמה שיותר צמוד לפניה
 - בין אם זו פניה בודדת ל DAL ובין אם זו טרנזקציה (אוסף רצוף) של פניות ל DAL.
 - בין אם זו פניה ישירה ל DAL ישירה ובין אם זו פניה עקיפה דרך מתודות עזר ב BL
 - ה- Notifications למשקיפים יזומנו רק מחוץ לבלוק של lock
- המתודה תעבור **על כל השליחים הפעילים**
 - חשוב לבצע את השאילתא זו ומיד להפוך אותה לרשימה ToList()
 - כדי לקבל רשימה קונקרטית ולא להיכנס לקטע קריטי עקב הביצוע המושהה של שאילתת linq
 - אם לא נהפוך את זה לאוסף קונקרטי אזי נצטרך לבצע lock שתקיף את כל לולאת ה foreach שעוברת על כל האיברים. וזה ייצור השהיה ארוכה.
 - כאשר אנו הופכים את האוסף לרשימה קונקרטית, זה מאפשר לנו לבצע locks צמודות לכל פעולת DAL בודדת
- עבור כל אחד מהשליחים הפעילים תבצע את הפעולות הבאות:
 - אם לשליח אין כרגע הזמנה בטיפול:

- תתבצע בחירה **רנדומלית** של הזמנה לטיפול מבין כל ההזמנות האפשריות עבורו (על פי כללי הלוגיקה הקיימים, כאילו השליח פתח מסך בחירת ההזמנה לטיפול ובחר באחת מהן)
- ~~הבחירה תתבצע רק מתוך ההזמנות הפתוחות שיש להן קואורדינטות שמחושבות כבר (אין לחשב מחדש את הקואורדינטות בזמן הפעלת השאילתא לקבלת הזמנות פתוחות – אלא להסתמך על הקואורדינטות שכבר חושבו בזמן הוספה/עדכון של שליח ושל הזמנה)~~
- מכיוון שהסימלוציה מופעלת בצורה אסינכרונית (ע"י הרצה ב-Task.Run), ניתן לבצע **בתוכה** את כל חישובי המרחק מכל הסוגים (כולל אלה שדורשים שאילתת רשת) **בצורה סינכרונית** כמו בשלבים הקודמים, ואין צורך בפעולה נוספת.
- הבחירה תתבצע בהסתברות מסוימת (למשל 20%, או אחרת לבחירתכם) - כלומר לפעמים תתבצע בחירה של הזמנה לטיפול ולפעמים לא. כביכול השליח פתח את המסך אך בסוף לא בחר.
- אם לשליח יש כרגע הזמנה בטיפול:
- אם עבר "מספיק זמן" מתחילת הטיפול - ההזמנה תיסגר כמטופלת על ידי אותו שליח (על פי כללי הלוגיקה הקיימים, כאילו השליח לחץ על כפתור "סיום טיפול")
- הפעילו שיקול דעת לגבי ההחלטה אם עבר "מספיק זמן", כתלות במרחק בין השליח להזמנה, ובתוספת פרק זמן רנדומלי אך הגיוני לבחירתכם
- דאגו לגיוון בין סוגי סיום המשלוח
- אם לא עבר "מספיק זמן" מתחילת הטיפול - כלומר ההזמנה לא נסגרה כמטופלת על ידי אותו שליח, אז בהסתברות מסוימת (למשל 10%) המנהל "יבטל" את הטיפול בהזמנה (על פי כללי הלוגיקה הקיימים, כאילו המנהל "לחץ" על כפתור ביטול טיפול במסך ניהול הזמנה בודדת)

התנהגות המסכים הפתוחים בזמן הפעלת הסימולטור

- כמפורט **במסמך ההנחיות של שלב 7**, במהלך הפעלת הסימולטור מסכים יכולים להיות כבר פתוחים או להיפתח/להיסגר במהלך פעולתו.
- במידה ומימשתם נכון את כל ההנחיות של שלב 7**, בכל אחת משלושת השכבות של מערכת השליחים, אזי בזמן הפעלת הסימולטור ההתנהגות של כל מסך (במידה והוא פתוח או נפתח) אמורה להיות כמפורט כאן.
- לחצו על כפתור הפעלת הסימולטור במסך הראשי ובדקו שבזמן פעולת הסימולטור כל אחד מהמסכים מתנהג כמצופה, באופן הבא:**

מסך ניהול ראשי:

- כפתור הפעלת/עצירת הסימולטור יישאר נגיש ללחיצה אך הטקסט שלו ישתנה מ- "הפעל סימולטור" ל- "עצור סימולטור"
- **תיחסם** אפשרות קלט בשדה של קצב התקדמות השעון
- תתעדכן אוטומטית תצוגת שעון המערכת בהתאם לכך שערך שעון המערכת יתקדם אוטומטית ע"י הסימולטור
- **תתעדכן אוטומטית תצוגת סיכום הכמויות של ההזמנות באזור סיכום הזמנות, לפי שינויי הסטטוס ב"זמן אמת"**
- כפתורי ניהול (קידום) השעון, וכפתורי אתחול ואיפוס בסיס הנתונים - **ייחסמו**
- בעת סגירת החלון תתבצע סגירה מסודרת של הסימולטור כאילו נלחץ כפתור עצירת הסימולטור לפני סגירת החלון
- **הכפתורים לפתיחת מסך ניהול הזמנות ומסך ניהול שליחים - לא ייחסמו** ויהיה אפשר ללחוץ עליהם
- בנוסף אפשר יהיה לפתוח כמה מסכים במקביל מכל אחד מהם

מסך ניהול שליחים:

- תצוגת רשימת השליחים, תתעדכן אוטומטית ב"זמן אמת":
- עבור כל שליח ברשימה תתעדכן מספר ההזמנה שבטיפולו - אם כבר קיימת, אם נהייתה קיימת או אם כבר לא קיימת
- וכנ"ל נתונים נוספים המוצגים ברשימה (אם השתנו)
- ניתן יהיה לשנות את תנאי הסינון והמיון והרשימה תתעדכן אוטומטית בהתאם
- כפתור הוספת שליח לא ייחסם - לחיצה עליו תגרום לחריגה שתיזרק משכבת ה BL ותיתפס כאן.
- לחיצה כפולה על שליח ברשימה תתאפשר, ויפתח מסך לניהול שליח בודד

מסך ניהול שליח בודד:

- פרטי השליח הרלוונטיים יתעדכנו אוטומטית ב"זמן אמת"
- כפתורים של עדכון/מחיקה של שליח בודד לא ייחסמו - לחיצה עליהם תגרום לחריגה שתיזרק משכבת ה BL ותיתפס כאן.
- פרטי הזמנה בטיפול המופיעים במסך יתעדכנו אוטומטית ב"זמן אמת"

- אם לא הייתה הזמנה בטיפול והסימולטור "בחר" עבור השליח הזמנה לטיפול בה - היא תופיע אוטומטית
- אם הזמנה שהייתה בטיפול נסגרה בדרך כזו או אחרת - ההזמנה תיעלם מהמסך

מסך הוספת שליח:

- המסך יוכל להיפתח, פקדיו יאופשרו אך הוא אינו רלוונטי
- כפתור 'הוספה' ל ייחסם - לחיצה עליו תגרום לחריגה שתיזרק משכבת ה BL ותיתפס כאן.

מסך שליח:

- פרטי השליח הרלוונטיים יתעדכנו אוטומטית ב"זמן אמת"
- הכפתורים עדכון, סיום טיפול, ביטול טיפול, הסטוריית הזמנות - לא ייחסמו - לחיצה עליהם תגרום לחריגה שתיזרק משכבת ה BL ותיתפס כאן.
- פרטי הזמנה בטיפול יתעדכנו אוטומטית ב"זמן אמת"
- אם לא הייתה הזמנה בטיפול והסימולטור "בחר" עבור השליח הזמנה לטיפול בה - היא תופיע אוטומטית
- אם הזמנה שהייתה בטיפול נסגרה בדרך כזו או אחרת - ההזמנה תיעלם מהמסך
- כפתור בחירת הזמנה יאופשר (לצורך צפיה בהזמנות פתוחות עבור השליח)

מסך ניהול הזמנות:

- תצוגת רשימת ההזמנות, תתעדכן אוטומטית ב"זמן אמת":
- סטטוס הזמנה
- נתונים נוספים המוצגים ברשימה (אם השתנו)
- ניתן יהיה לשנות את תנאי הסינון והמיון והרשימה תתעדכן אוטומטית בהתאם
- כפתור הוספת הזמנה לא ייחסם - לחיצה עליו תגרום לחריגה שתיזרק משכבת ה BL ותיתפס כאן.
- לחיצה כפולה על הזמנה ברשימה תתאפשר, וייפתח מסך לניהול הזמנה בודדת

מסך ניהול הזמנה בודדת:

- פרטי ההזמנה הרלוונטיים (למשל סטטוס, או זמן כלשהו) יתעדכנו אוטומטית ב"זמן אמת"
- רשימת המשלוחים תתעדכן אוטומטית, כולל מיון בהתאם לתנאי המיון

מסך הוספת הזמנה:

- המסך יוכל להיפתח, פקדיו יאופשרו אך הוא אינו רלוונטי
- כפתור 'הוספה' לא ייחסם - לחיצה עליו תגרום לחריגה שתיזרק משכבת ה BL ותיתפס כאן.

מסך היסטוריית משלוחים:

- תצוגת רשימת המשלוחים הסגורים עבור שליח - תתעדכן אוטומטית ב"זמן אמת"

מסך בחירת הזמנה לטיפול

- תצוגת רשימת ההזמנות הפתוחות עבור השליח תתעדכן אוטומטית
 - האפשרות לבחור הזמנה לטיפול לא תיחסם - לחיצה עליו תגרום לחריגה שתיזרק משכבת ה BL ותיתפס כאן.
- ודאו שלחיצה על כפתור עצירת הסימולטור במסך הראשי - גורמת לעצירת הסימולטור ושתצוגת המסכים חוזרת למצבה הרגיל.**

נספח 1 - חישוב קואורדינטות של כתובות ומרחק בין כתובות

בפרויקט שלנו נעסוק בכתובות (כתובת של החברה וכתובת של הזמנות) ובחישוב המרחקים ביניהן (מרחק אווירי, מרחק נסיעה, מרחק הליכה). **דוגמא לכתובות:** הנשיאים 7, פתח תקוה, ישראל. המרכז האקדמי לב, ירושלים, ישראל.

כתובת גיאוגרפית תקנית מיוצגת בעזרת קואורדינטות גיאוגרפיות שהם שני מספרים:

- **Latitude (קו רוחב)** – מציין מיקום צפון-דרום
 - ערכים בין 90° ל- $90^{\circ}+$
 - חיובי = צפון, שלילי = דרום
- **Longitude (קו אורך)** – מציין מיקום מזרח-מערב
 - ערכים בין 180° ל- $180^{\circ}+$
 - חיובי = מזרח, שלילי = מערב

על מנת לחשב את כל סוגי המרחקים בין 2 כתובות יש צורך בקואורדינטות שלה (קו אורך וקו אורך בכדור הארץ). אם הכתובת תקינה (קיימת במציאות) אזי ניתן לחשב לה את הקואורדינטות.

נצטרך עבור הפרויקט את סוגי החישובים הבאים:

- בהינתן כתובת תקינה - חישוב הקואורדינטות (קו רוחב וקו אורך) שלה
- בהינתן 2 כתובות תקינות (2 זוגות של קואורדינטות) - חישוב מרחק (אווירי/נסיעה/הליכה) בין כתובות

חישוב קואורדינטות של כתובת תקינה וכן חישוב מרחק נסיעה והליכה - דורש לבצע דרך הקוד פניית רשת לאתרי מפות חיצוניים (אתרי geocoding).

חישוב מרחק אווירי - הוא חישוב מתמטי פשוט שניתן לעשות בהינתן 2 זוגות של קואורדינטות (ניתן למצוא אותו ברשת בקלות).

בשלב 1 (אתחול), החישובים הללו יעשו בעזרת גישה ידנית לאתרים חיצוניים ויוזנו בצורה מפורשת.

בשלב 4 (השכבה הלוגית) החישובים הנ"ל יעשו דרך הקוד בעזרת מתודה שתפנה לאתר שנותן שירותי מסוג geocoding. המתודה תפעל בצורה **סינכרונית** (ללא שימוש ב `async/await`), כלומר לא עוברים לפקודה הבאה עד קבלת התשובה המבוקשת ברשת.

בשלב 7 (שלב הסימולטור) - נעדכן את הקוד שיעבוד בצורה **אסינכרונית**. הקוד יפנה לאתר שנותן שירותי geocoding בצורה אסינכרונית. ראה פירוט בשלב 7.

חישוב ידני מפורש

נציג אפשרות אחת לקבל קואורדינטות ולחשב מרחקים במורה ידנית - דרך אתר גוגל מפות.

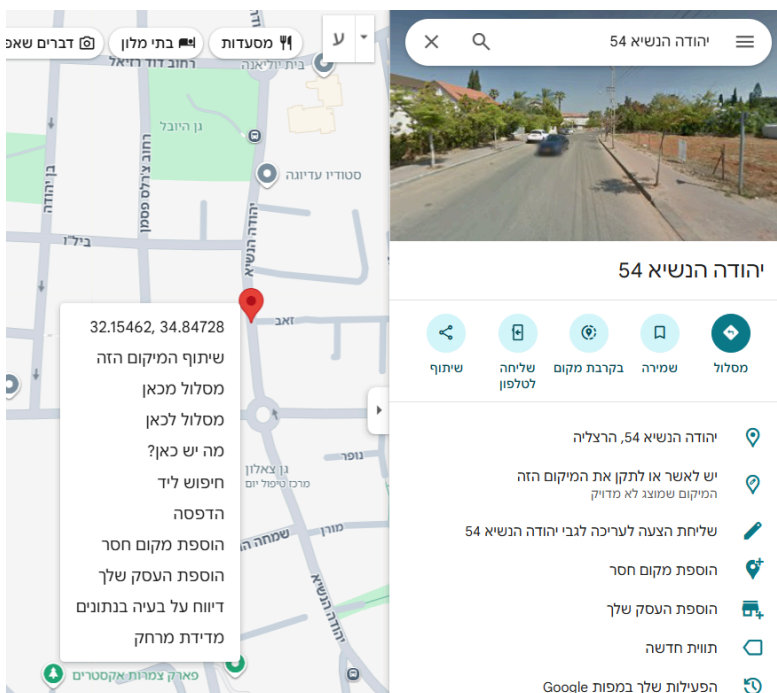
ניתן למצוא גם אתרים אחרים שעושים חישובים אלו. לשעקולכם.

חישוב ידני של קואורדינטות של כתובת

1. דרך אקספלורר כלשהו, הכנסו לאתר **גוגל מפות** (<https://www.google.com/maps>)
2. כתבו את הכתובת הרצויה והקישו על חיפוש
3. תופיע טיפה אדומה בהתאם לכתובת שחיפשתם
4. הקישו עם עכבר ימני על הטיפה האדומה ותקבלו תפריט שיפתח עם ערכי הקואורדינטות כתובים עליו
5. כדי להעתיק את הקואורדינטות, לוחצים על קווי הרוחב והאורך

חישוב ידני של מרחק נסיעה/הליכה בין 2 כתובות

1. דרך אקספלורר כלשהו, הכנסו לאתר **גוגל מפות** (<https://www.google.com/maps>)
2. חשבו מסלול בין 2 הכתובות הרצויות
3. בחר את סוג המסלול נסיעה/הליכה



4. המסלול יופיע על המפה
5. עמוד עם העכבר על המסלול ותופיע בועה עם המרחק של מסלול

חישוב דרך הקוד - פנייה דרך הקוד לאתרי geocoding

יש למצוא אתר שנותן שירותי Geocoding:

- **Forward Geocoding** - המרה של כתובת טקסטואלית לקואורדינטות גיאוגרפיות (Latitude/Longitude)
- **Reverse Geocoding** - המרה של קואורדינטות גיאוגרפיות לכתובת
- **Routing** - חישוב מסלול מנקודת מוצא לנקודת יעד. מסלול הליכה או נסיעה וכו'

בפרויקט שלנו נידרש ל Forward Geocoding ול- Routing.

דוגמא חלקית לאתרים שניתן לפנות אליהם דרך הקוד כדי לקבל שירותי Geocoding:

- **OpenStreetMap (OSM) – Nominatim** - אתר חינמי, לא דורש מפתח API ולא מוגבל במספר בקשות רשת ליום. חסרון: אך הוא אינו מכיר בנוכחות יהודית ביו"ש ובירושלים.
- **Google Maps** - בתשלום אחרי כמות חינמית מוגבלת. דורש הזנת פרטי כרטיס אשראי מראש. בפועל, לא נדרש תשלום בהיקפי השימוש הנדרשים לפרויקט.
- **LocationIQ** - מאפשר 5000 בקשות רשת ליום בחינם. אפשר להתחבר עם חשבון גוגל.
- <https://geocode.maps.co/join> - בחינם אך מצריך יצירת חשבון.

למשל אם ברצונכם לפנות לאחד מהאתרים האלו כדי לחשב את הקואורדינטות של כתובת מסוימת:

1. יש לשלוח דרך קוד C# בקשת רשת HTTPClient ל API של אחד מהאתרים שנותנים שירותי הגיאוקודינג
a. בקשת הרשת היא בעצם מחרוזת אשר כוללת את כתובת האתר ואת הכתובת הגיאוגרפית המבוקשת
2. השרת של השירות מחפש את הכתובת במסד הנתונים שלו
3. התשובה תתקבל בפורמט של JSON או XML עם המיקום (Latitude, Longitude)
4. חילוץ הקואורדינטות מתוך התשובה שהתקבלה