

גרסה 0 - מידול המערכת – 2023

כל הגרסאות של הפרויקט מתייחסות ותלויות ב-[מסמך הדרישות הכללי](#) וב-[מתודולוגית פיתוח הפרויקט](#). כל גרסה כוללת חלק מהדרישות במסמך הכללי, עם או בלי שינויים, ואולי דרישות נוספות. משימת המידול הראשוני (גרסה 0) מניחה את התשתית לגרסה 1 ובעצם לכל המערכת ומהווה את הבסיס לפרויקט. על כן יש לה חשיבות מכרעת על המשך הפרויקט. הגרסה כוללת:

1. בניית מודלים עבור הדרישות בגרסה.
 2. תכנון בדיקות לקוח (acceptance tests).
- מנהל גרסה:** יש למנות מנהל עבור הגרסה (התפקיד מתחלף בין חברי הצוות בכל גרסה). כל חברי הצוות שותפים במשימת המידול.

תהליך העבודה על גרסה 0:

1. קריאת מסמך הדרישות הכללי והדרישות עבור גרסה 0.
2. בניה (אולי במקביל) של מודל מחלקות לבן וארכיטקטורת מערכת. תוך כדי בניית המודלים, ניתן לפרט את הדרישות הפונקציונליות ולתכנן הטלת האחריות לביצוע הדרישות.
 - a. דיאגרמת מחלקות לבנה עשירה, המציגה מחלקות (ללא היררכיה), קשרים ביניהן ואילוצים המוטלים עליהם.
 - i. לכל מחלקה צריך להיות יעוד מוגדר וברור.
 - ii. כל קשר צריך לנבוע מהדרישות. יש להראות כיווניות, אם ישנה.
 - iii. כדאי להוסיף פירוט טקסטואלי לגבי יכולות ואחריות של כל מחלקה.
 - b. כאשר בונים את ארכיטקטורת המערכת יש לשים לב מהם הרכיבים במערכת, לשים לב למנשק שכל מרכיב חושף, ולשים לב לצורך בשכבת שירות כמנשק כולל עבור הדרישות הפונקציונליות.
3. הטלת אחריות לטיפול בדרישות רמת שירות, פנייה למערכות חיצוניות ואכיפת אילוצי נכונות. עבור כל אילוץ נכונות, צריכים להראות כיצד יובטח קיומו: על ידי הבנייה, או על ידי תרחישי קבלה.
4. פירוט תרחישי השימוש (use cases) ובניית מילון המונחים. תרחישים מורכבים, כגון דרישת משתמשים 2.5 (רכישת עגלת הקניות) יש להפריד לתתי-תרחישים. עבור קונה מנוי, אין צורך לחזור על התרחישים שכבר תוארו עבורו כקונה אורח אלא רק לתאר את ההבדלים.
5. תכנון בדיקות לקוח: יש לכתוב תרחישי קבלה חיוביים ושיליים עבור כל תרחיש שימוש. אלו יהוו את הבסיס למבחני קבלה עבור המימוש

הדרישות בגרסה 0 (מתוך מסמך הדרישות הכללי):

1. דרישות פונקציונליות: מידול תרחישי השימוש הבאים, מתוך מסמך הדרישות הכללי:
 - i. מערכת: כל הפעולות.
 - ii. משתמשים:
 - מבקר אורח: כל הפעולות

- **מבקר מנוי:** כל הפעולות שלו כמבקר-אורח, וכן דרישות 3.1 ו-3.2.
- **בעל חנות:** 4.1, 4.2, 4.4, 4.6, 4.7, 4.9, 4.11, 4.13
- **מנהל חנות:** כל הפעולות, בהתאם להרשאות
- **מנהל מערכת:** 6.4

בשלב זה אין דרישה מוגדרת עבור סוגי רכישה, סוגי הנחה, מדיניות רכישה ומדיניות הנחה ספציפיים בהם נדרש לתמוך, אך תוצרי הגרסה (המידול) צריכים להיות כלליים מספיק על מנת לתמוך בסוגים שונים ובכללים משתנים.

2. **דרישות רמת שירות:** דרישת העקביות (1), ודרישת הפרטיות (2), מתקיימות. בפרט, כל אילוצי הנכונות המפורטים במסמך הדרישות הכללי נשמרים לאחר כל פעולה.

תוצרי גרסה 0:

מסמך המתודולוגיה מתאר את התוצרים השונים ואופן תיאורם. התוצרים בגרסת מידול זו הם:

- א. מודל מחלקות לבן.
- ב. דיאגרמת ארכיטקטורה.
- ג. הטלת אחריות לקיום אילוצי נכונות.
- ד. מילון מונחים.
- ה. תרחישי שימוש.

מצגת כיתה – מערכת לניהול פרויקטים

בכל גרסה יהיה על סטודנט אחד מהצוות להציג מצגת אישית.

אנו ממליצים של מערכת GitHub ככלי לניהול הפרויקט והקוד.

נתח/י כיצד האלמנטים הבאים ניתנים לניהול באמצעות הכלי לניהול פרויקטים (GitHub):

- הגדרת משימות
- תיעדוף ותזמון
- ניהול הצוות וחלוקת משימות
- מעקב אחרי התקדמות הגרסה
- מיפוי (מעקב) בין הדרישות השונות לקוד המתאים ולבדיקות (traceability).
- ניהול הקוד (code review, pull requests, etc.).
- ניהול תיעוד הפרויקט (כולל המודלים).

הדגם/י את אפשרויות הניהול הנ"ל עבור הפרויקט.

הערה: ניתן לבחור פלטפורמה חלופית לניהול הפרויקט בהתייעצות עם ובאישור המנחה.