



ניתוח ועיצוב מערכות תוכנה (372-1-3401, 372-1-3801) מטלה 2

מועד הגשת תרשימים ומסמכים: עד 10.10.2024 בשעה 23:59, מועד הגשת הקוד: 03.10.2024 עד השעה 23:59, מועד אחרון לשאלות ללקוח הינו 27.9.2024 בשעה 18:00

הוראות הגשה

1. יש לרשום במפורש את שמות חברי הקבוצה + ת"ז (של הזוג) על גבי העבודה בעמוד הראשון.
2. יש להוסיף את מטלה 1 המעודכנת כנספח בסוף מטלה זו.
3. כל התרשימים יבוצעו באמצעות כלי מידול (כגון **Visual Paradigm**, **Visio**, **Draw.io** וכלי **CASE** אחרים).
4. הנחיות ההגשה לחלק התכנותי זהות להנחיות מתרגיל 1 (תיקיות **dev**, **release**, **docs** וכל היוצא בכך) כאשר שם **jarn** של עבודה זו יהיה **adss2024_v02.jar**.

עליכם להרחיב ולעדכן את מטלה 1 באופן הבא:

1. יש לנהל את הנתונים בבסיס נתונים רלציוני.
2. הרחבת הפונקציונליות של המערכת כך שתתמוך גם בדרישות שלא התאפשרו בעבר בעקבות חוסר בנתונים. (דרישות כגון אלה שהייתם צריכים לכתוב בטבלה מספר 2 בעבודה 1).
3. שפרו את עיצוב המערכת: הקפידו על העקרונות של לכידות גבוהה וצימוד נמוך (**high cohesion and low coupling**). עצבו את המערכת, הסבירו את השינויים שביצעתם ונמקו אותם.

הרחבת המערכת:

מודול ספקים + מודול מלאי:

עד עכשיו לעובדי החברה לא הייתה ברירה אלא לפתוח הזמנות באופן ידני על בסיס דו"ח המלאי שהעביר המחסנאי במקרים של הספקה לפי החוסר, כנ"ל גם עבור הספקה בימים קבועים: אם המחסנאי לא יעדכן את ההזמנה, תגיע סחורה מהספק לפי ההזמנה הישנה. דבר זה גרם לחוסר איזון במלאי המוצרים במחסן. עם התרחבות פעולת החברה, ההנהלה מעוניינת לייעל את הטיפול בחוסרים ולהקל על התקשורת מול הספקים באמצעות מערכת הוצאת הזמנות אוטומטית. המערכת מוציאה הזמנה בשני מקרים: הזמנה מספק עקב חוסר במוצר כלשהו והזמנה תקופתית מספק עם זמני הגעה קבועים. על מנת להוציא הזמנה עקב חוסר כלשהו יש לקבל את פרטי המוצר ממערכת המלאי, וכן את הכמות הנדרשת לכל פריט. את ההזמנות התקופתיות יש לעדכן לפחות יום אחד לפני מועד ההספקה על מנת שהספק יוכל להכין את המוצרים בהזמנה. כמו כן, יש לוודא שרמת המלאי שצפויה עם הגעת ההזמנה תעלה על רמת מלאי המינימום המוגדר לכל פריט.

כאשר נעשית הזמנה למוצר שמסופק על ידי יותר מספק אחד, יש לבדוק על פי כתבי הכמויות של הספקים וכמות המוצר הנדרשת אצל איזה ספק יהיה את המחיר הטוב ביותר לפריט ולהזמין ממנו.

חלק א – ניתוח ועיצוב (60%)

1. ניתוח התנהגותי - Use Case (15%) תרחישי שימוש

1.1. (5%) צרו תרשים תרחיש שימוש (Use Case Diagram) עבור כל תרחישי השימוש הבאים המתייחסים לפעילות הקשורה למודל שבחרתם, ובהתאם להגדרות המערכת כפי שתוארו עד כה (כולל ההרחבות).

a. כניסת עובד חדש לתפקיד

b. עדכון פרטי עובד קיים וזמינות למשמרות

c. עדכון מלאי והתראה על חוסרים

d. הוספת ספק חדש

e. הוצאת הזמנה תקופתית מספק

f. הוצאת הזמנה מספק עקב חוסר

g. שיבוץ עובדים למשמרות

h. הוצאת הובלה

1.2. (10%) פרטו 2 תרחישי השימוש שמתייחסים למודל שלכם, כאשר אחד מהם צריך להיות מבין תרחישי השימוש המוצגים בסעיף א'. הציגו אותם לפי התבנית הבאה:

Use case name
Textual Description
List of Actors
Pre-conditions
Post-conditions
Main success scenario
Alternatives/Extensions

את התסריט העיקרי ואת ההרחבות\אלטרנטיבות שלו יש לתאר בצורה מובנית באמצעות אחת מהדרכים הבאות: pseudo code\activity diagram

יש לתאר את כל האינטראקציות של השחקנים עם המערכת במסגרת התרחיש ורק אותן. אין לתאר אינטראקציות בין רכיבי מערכת בסעיף זה (כלומר, עליכם להציג את תהליך תרחיש השימוש בצורה כללית הכוללת את ה-actors בתרחיש אך ללא רכיבי המערכת). טיפול בשגיאות מהותיות ניתן לכלול כאלטרנטיבה לתסריט ראשי עם extension point בכל שלב בתסריט הראשי.

2. ניתוח התנהגותי - Sequence & Collaboration Diagrams (30%)

2.1. (10%) לכל אחת מהפעולות המופיעות בתרחישי השימוש המפורטים (מסעיף 1.2) כתבו חוזה.

2.2. (20%) צרו תרשימי רצף (Sequence Diagram) או תרשימי שיתוף פעולה (Collaboration Diagram) עבור כל אחד מהחוזים מסעיף 2.1.

יש לכלול את כל הdomain objects המעורבים תהליך שמתואר. השתמשו בעקרונות GRASP עבור חלוקת האחריות על ביצוע פונקציות למחלקות השונות. אין צורך לכלול שחקנים בתרשימים אלו, ניתן להתחיל את התרשים ישירות מהחווזה (פונקצייה) המתואר בו. אין לפרט את אופן פעולתן של פונקציות סטנדרטיות כגון מיון מערכים, חיפוש, השוואות, וכד' (אך כן יש לציין שהן חלק מהתהליך). יש לכלול getters בתרשים במידה והאובייקטים שהם מחזירים נדרשים להמשך התהליך.

3. עדכון תוצרי עבודה 1 (15%)

3.1. (10%) עדכנו את התרשימים מעבודה 1 בהתאם לשינויים שבצעתם בעבודה זו: יש לעדכן את תרשימים המחלקות (בפרט הוספת פונקציות) כך שישקפו את השינויים שהוכנסו במערכת. שימו לב, אין לרשום את כל הפונקציות שיש במחלקות אלא רק את העיקריות (רמז, לרוב אלה הפונקציות שמממשות את הדרישות. המטרה היא שמהסתכלות על תרשימים המחלקות ניתן יהיה להסיק בקלות את ה-responsibilities שיש למחלקה).

3.2. (5%) עדכנו את מסמך הדרישות (מטלה 1) בהתאם לאילוצים העדכניים ובהתאם לניתוח שבצעתם עד כה. שימו לב, רוב הדרישות צריכות להיות מתוארות בצורה רשמית ובמבנה של "המערכת תאפשר.."/"המערכת תעשה..". וכדומה.

באפשרותכם להוסיף דרישות חדשות, להסיר או לעדכן דרישות קיימות. כל שינוי חייב להיות בהתאם לאילוצים שהוצגו הן במטלה זו והן במטלה הקודמת (מטלה 1). עליכם לציין אילו שינויים ביצעתם ולנמק אותם. במקרה של סתירות בין אילוצים ממטלות שונות, אילוצי המטלה המאוחרת הם הקובעים. מסמך הדרישות צריך להיות חד משמעי ולא להכיל דרישות סותרות.

תיעודף הדרישות צריך להיות ברמה של **Must have, Nice to have** בלבד. על מסמך הדרישות להכיל את כל הדרישות הרלוונטיות (לכל המערכת, לא רק למודולים שלכם) לדעתכם עד לסוף הפרויקט מתוך מטרה להכיר את המודולים השכנים. מתוך כלל הדרישות יש לסמן אילו מומשו באיטרציה זו, ואילו ימומשו באיטרציות הבאות. בנוסף, מסמך הדרישות צריך להכיל פרק של הגדרת מושגים (למשל הזמנה, ספק, הובלה וכד') ופרק של ההנחות שלכם לגבי המערכת. ההנחות הינן שאלות שנשאלו בפורום, שאלות שעניתם עליהן בעצמכם ושאלות שנשאלו פתוחות. שימו לב, התשובות בפורומים השונים הינן מחייבות לכולם! התעלמות או סתירת ההנחיות בפורום עלולה לגרום הורדה בציון - תהיו ערניים לפורום ותחשבו היטב על השאלות שלכם! יש לצרף את התרשימים המעודכנים ומסמך הדרישות כנספחים בסוף המטלה.

חלק ב – מימוש (40%)

ממשו את המערכת בהתאם לדרישות שהוצגו במטלה זו והעיצוב שלכם בחלק א'. בעבודה זו תממשו Database בעזרת מנוע קיים (לדוגמא: SQLite). עליכם להוסיף בדיקות - Integration & unit tests. לפחות 10 בדיקות (סה"כ ובנוסף ל-10 הבדיקות מהעבודה הראשונה). מימוש הטסטים הינו חובה וחוסר בהם או כישלונם יגררו ירידה בציון. התייחסו לטסטים ברצינות! תהליך שאין לו טסט כאילו לא מומש כלל וכמובן שתהליך שהטסטים שלו לא עוברים הינו תהליך לא תקין ופגם במערכת שלכם.

דגשים נוספים

- יש לצרף לקוד קובץ README, שיכלול את שמות ות.ז. שלכם, הוראות הפעלה מפורטות לקוד, יכלול כל כלי וספרייה שנעשה בהם שימוש.
- ממשק משתמש פשוט ככל שניתן - command line interface \ prompt
- יש לממש את בסיס הנתונים של המערכת על גבי המנוע שבבחר.
- יש להניח כי כל הנתונים יישמרו מקומית על גבי המכשיר/מחשב.
- על בסיס הנתונים להכיל נתונים לדוגמא שמאפשרים את בדיקת הפונקציונליות של המערכת שלכם באופן מלא: למשל שיהיו נתונים של מספר ספקים, חלקם מספקים את אותו מוצר במחירים שונים; שיהיו נתונים של

מספר עובדים שונים באותו תפקיד עם זמינויות שונות, ומספר עובדים לבחור מבניהם ליצירת משמרת, וכד'. עליכם לכלול בקובץ ההנחיות פירוט הטבלאות של הנתונים הקיימים בDB ההתחלתי. 6. בדומה למטלות קודמות, על הפרויקט להיות מקומפל יחד עם כל תלויותיו לקובץ jar. עליכם/ן לוודא שהקובץ רץ עם התקנה נקייה ע"י הרצת הפקודה: java -jar adss2024_v02.jar – ללא צורך בהתקנה של ספריות נוספות!

שימו לב, עבודה שלא תרוץ עם פקודה זו תאבד את כל הנקודות עבור החלק התכנותי וחבל!

7. על ההגשת מסמך האפיון (דרישות ותרשימים) להכיל:

- דרישות בפורמט שצוין מעלה, בפונט וגודל נוח וברור לקריאה!
- תרשימים קריאים וברורים (כתב בגודל של לפחות 8). אם יש צורך לפצל את התרשימים, פצלו אותם בצורה הגיונית ונוחה לקריאה.
- לכל תרשים יש להוסיף כותרות והסברים רלוונטיים (כגון מה התרחיש או האובייקט המתואר). שימו לב, כל תרשים שלא יהיה מודפס בתוכנה המיועדת לכך, באופן קריא וברור יביא לפסילתה של השאלה המתאימה. מומלץ להראות את התרשים למישהו חיצוני ולבדוק אם הוא מסוגל לקרוא.

זכרו, בודק העבודה הינו כמו לקוח - אתם מעוניינים להקל עליו בהבנת המערכת שלכם.

מדיניות הגשה באיחור:

- עד 3 שעות איחור - 5 נקודות.
- עד 24 שעות איחור - 10 נקודות.
- עד 48 שעות איחור - 30 נקודות.
- לאחר מכן העבודה תפסל.
- במידה והקבוצה לא מספיקה להשלים את המטלה בזמן על התלמידים בקבוצה להחליט האם להגיש את העבודה הלא שלמה בזמן או את העבודה השלמה באיחור.

בהצלחה!