

ספקים + מלאי

207979683 – אחמד מסאלחה

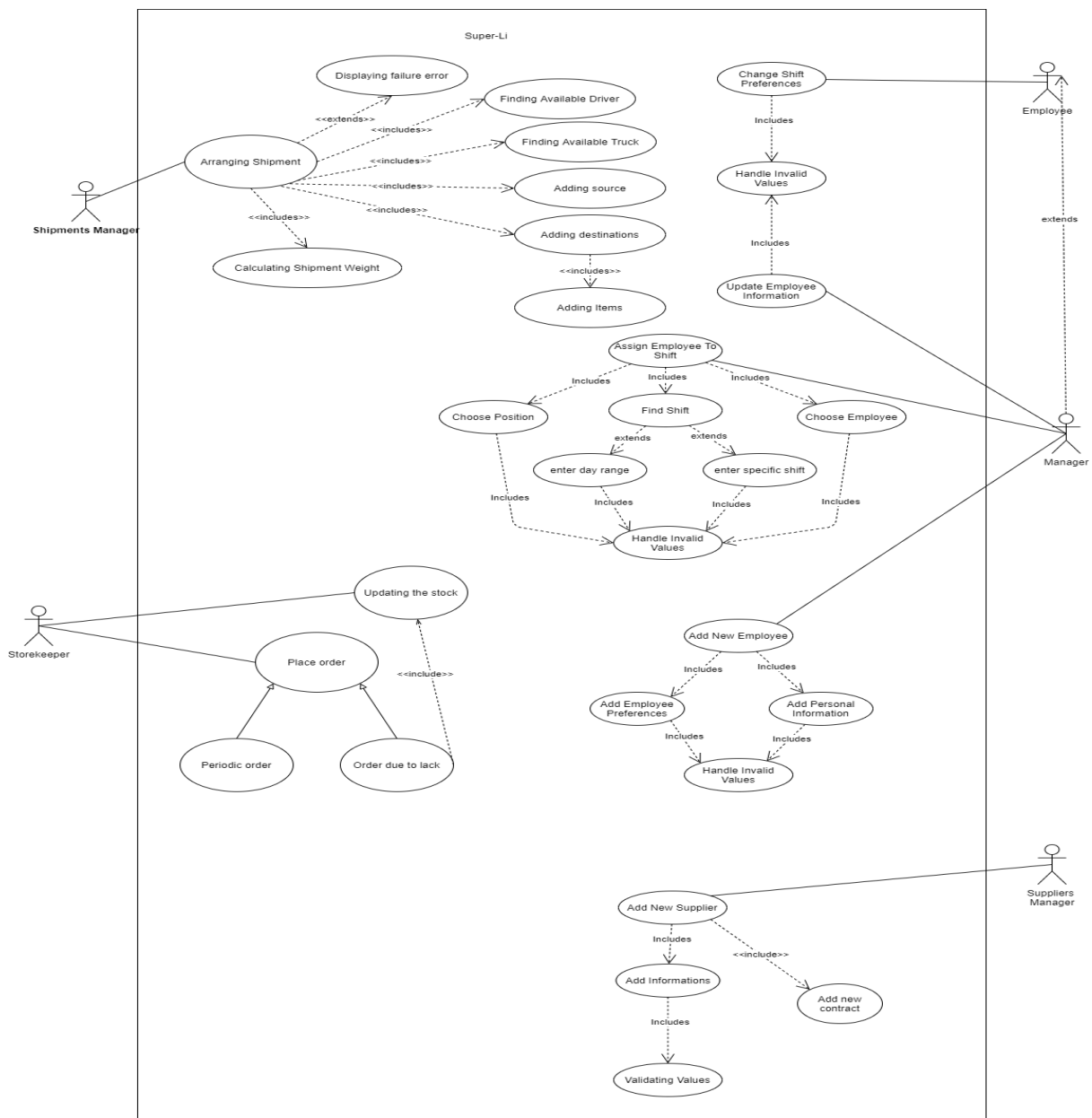
207951773 – לואי עכאשה

208041640 – עאדל ותד

208035824 - מוחמד עראבי

:Use case Diagram(1

(א)



E:

Name: Placing periodic order from supplier

Textual Description: This Use Case representing the action of adding a fixed-order from the supplier, which means, an order sent in periodic manner.

List of Actors: Warehouse worker.

Pre-conditions:

- Warehouse worker is identified and authenticated.
- A contract with this supplier exists.

Post-conditions:

- Order is saved.
- Worker gets an approval message.
- Order created every week according to the input details.

Main success scenario:

1. Warehouse worker logs to Super Li
2. Enters to the orders menu.
3. Enters the supplier's id that we are going place order to him and enters the order's details.
4. The order is sent to external transport department and gets approved.
5. The worker gets an approval message.
6. Order details are saved in the system.

Alternatives/Extensions:*1. At any time, System fails:*

The last action completed by the warehouse is saved in the system, and one can re-enter details from the next point of the chain, after he restarts the system.

2. Worker enters a bad detail:

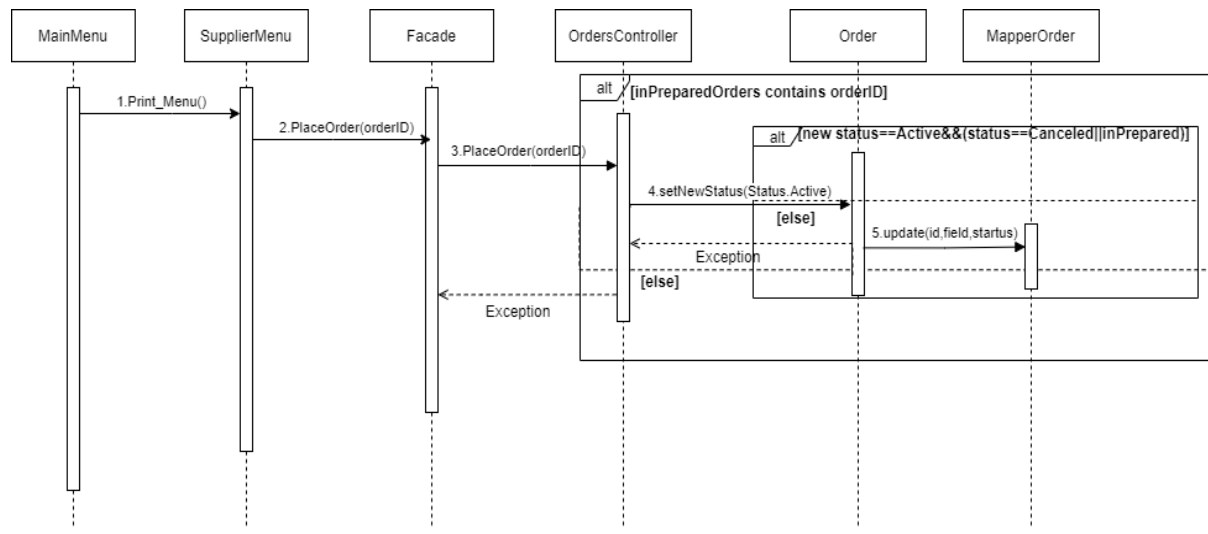
An appropriate message will display to screen, and one can re-enter the order.

3. The order did not get approved by transport department:

A message that clarifies why the order was not approved and the worker can re- enter the new fixed details (after talking with contact person (exists in the system!) about the problem).

4. A Quantity Agreement exists, containing one of the items in the order:

The new price of the item will be calculated and stored in the order, according to the quantity agreement.



F:

Name: Making a new order of minimum amount items.

Textual Description: When the manager gets a warning from the system that an item has reached its minimum amount, he has to make a new order and he has to set the amount to order at least as the minimum amount of the item.

List of actor: Supplier and store manager.

Pre-conditions:

1. Item reached its minimum amount
2. Store has at least one supplier that can prepare the products
3. If the store has more than one supplier that can prepare the order, the order sent to the supplier that offers the lowest price

Post conditions:

1. A new order has been created or a existing order updated
2. New order's amount is at least as the minimum amount of the item

Main success scenario:

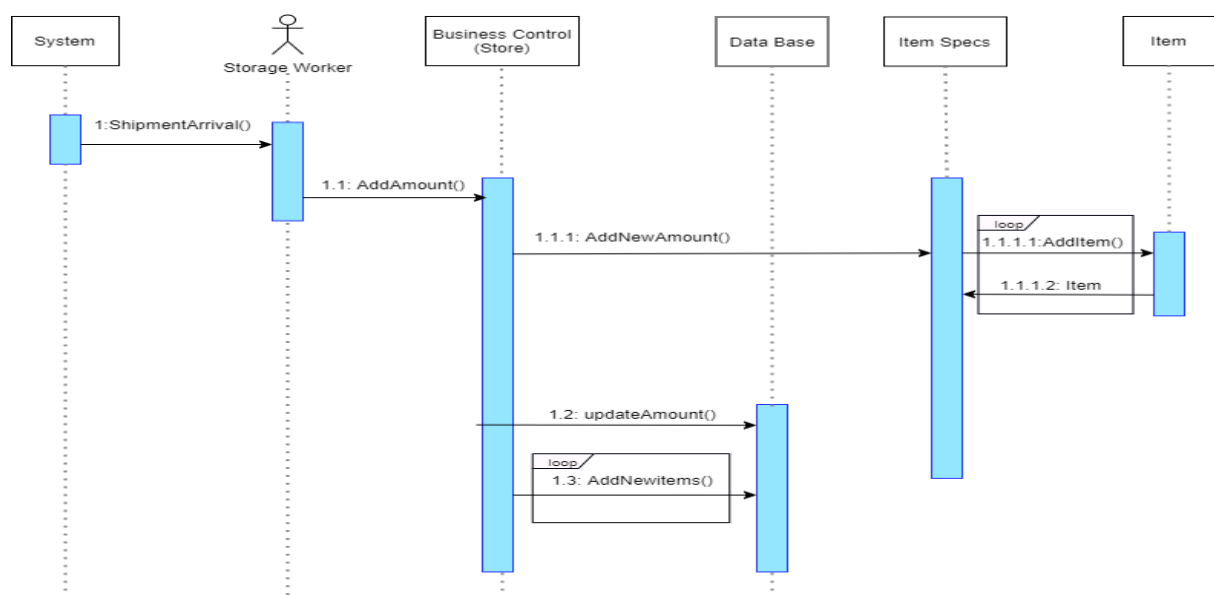
The steps of success scenario are:

1. the store manager selects the log in option from the main menu and he enters email and password.
2. The system checks correctness of email and password and shows warning messages of items with minimum amount

3. The system shows the specific menu of the store manager
4. The store manager selects the *automaticMinOrder*:
 1. The system scans all the items and for every item with minimum amount the system checks the available suppliers/companies that can prepare the order.
 2. If the system does not have a default amount to order, it requests from the store manager to enter amount to order.
 3. The system check legality of the amount entered.
 4. The system chooses the supplier with lowest price and adds the item's information to the supplier's order.
5. The system operates the *order* function, that sends the orders to all the suppliers with prepared orders.

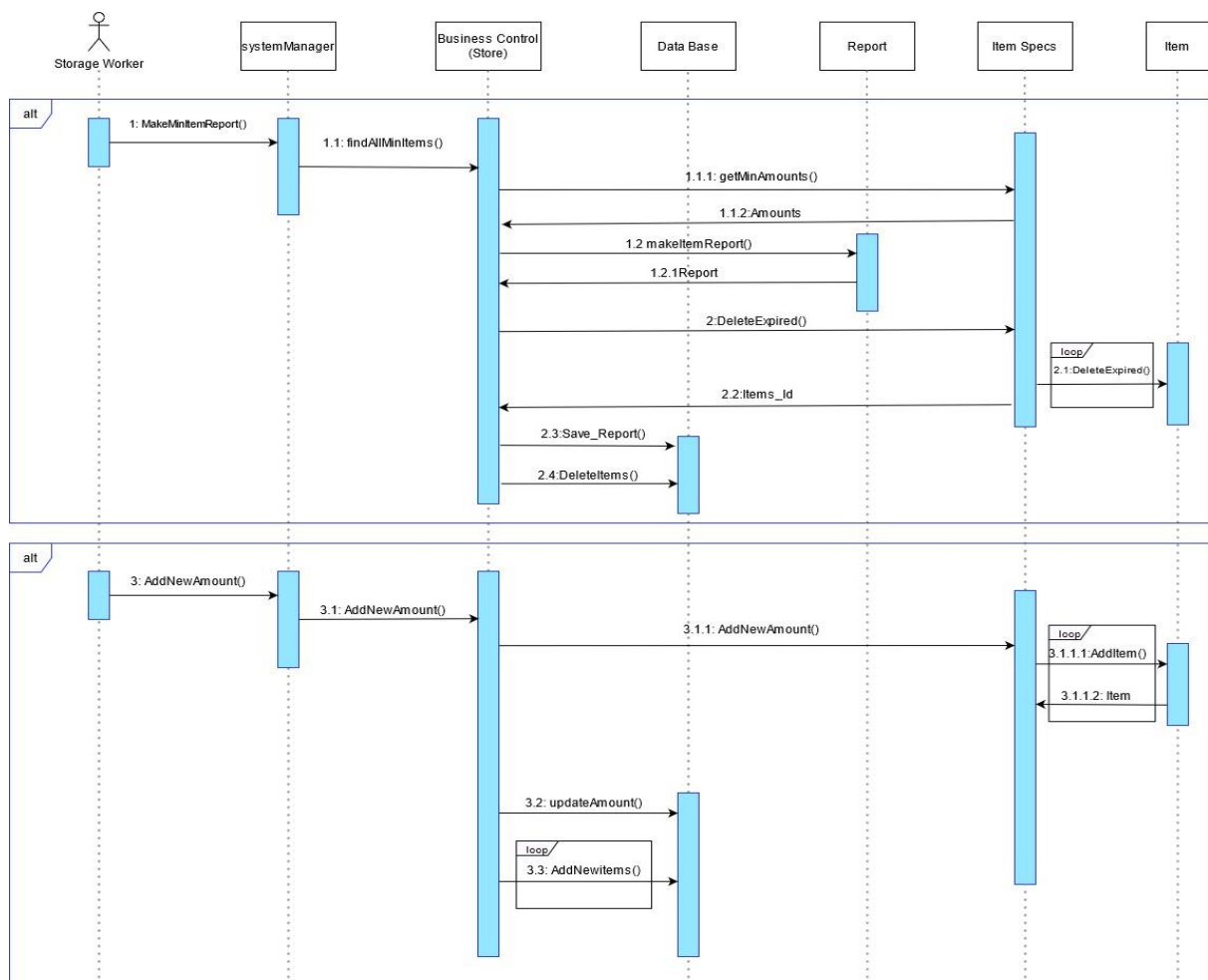
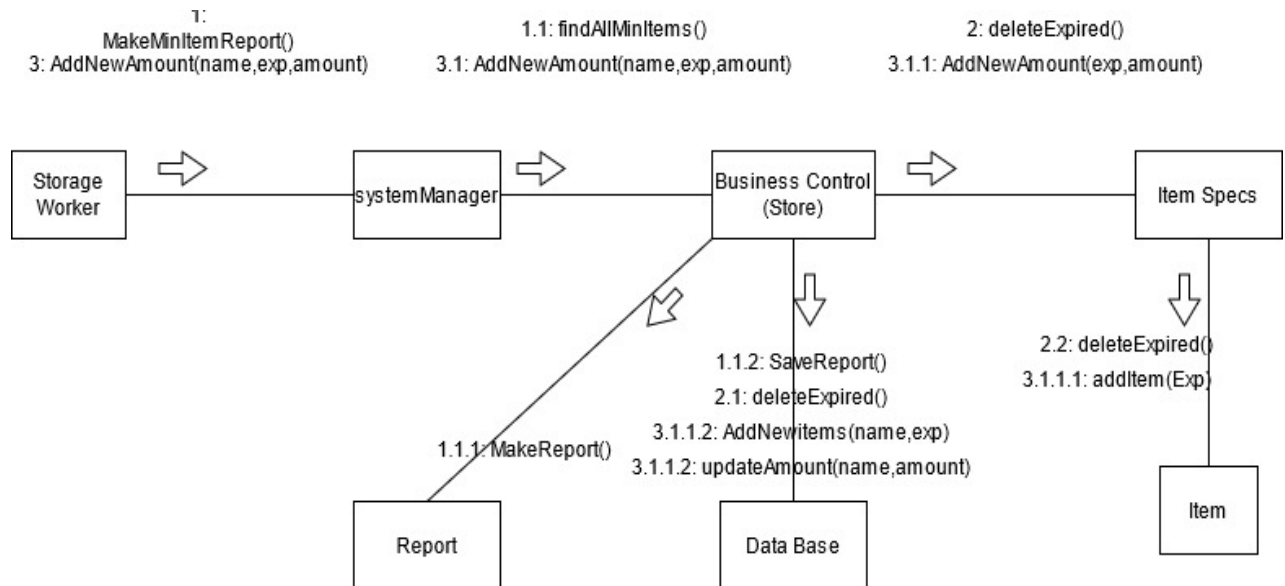
Alternatives/Extensions:

1. Store manager enters wrong email and password, alternative: the system asks again for email and password
2. Store manager selects a irrelevant/wrong option (operation), alternative: the system shows the specific menu again
3. Store manager enters a non positive amount to order, alternative: the system shows a relevant message and shows the specific menu again

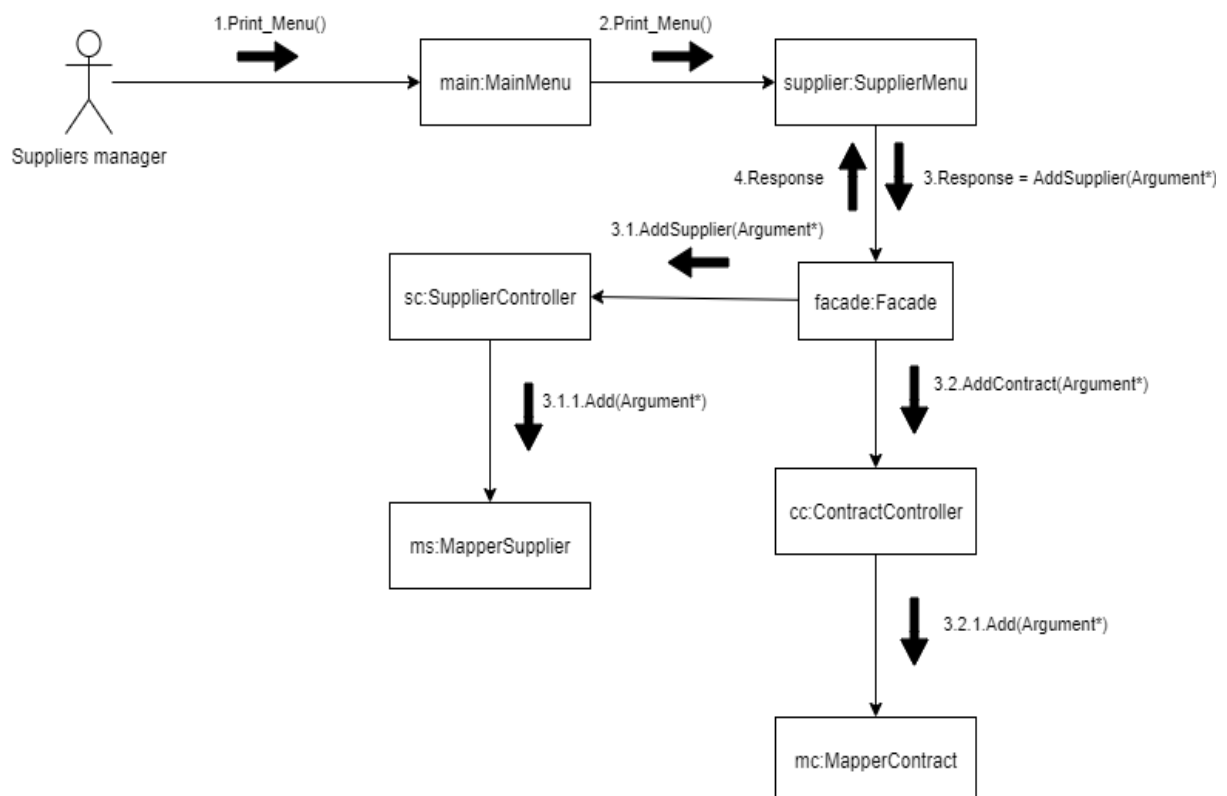
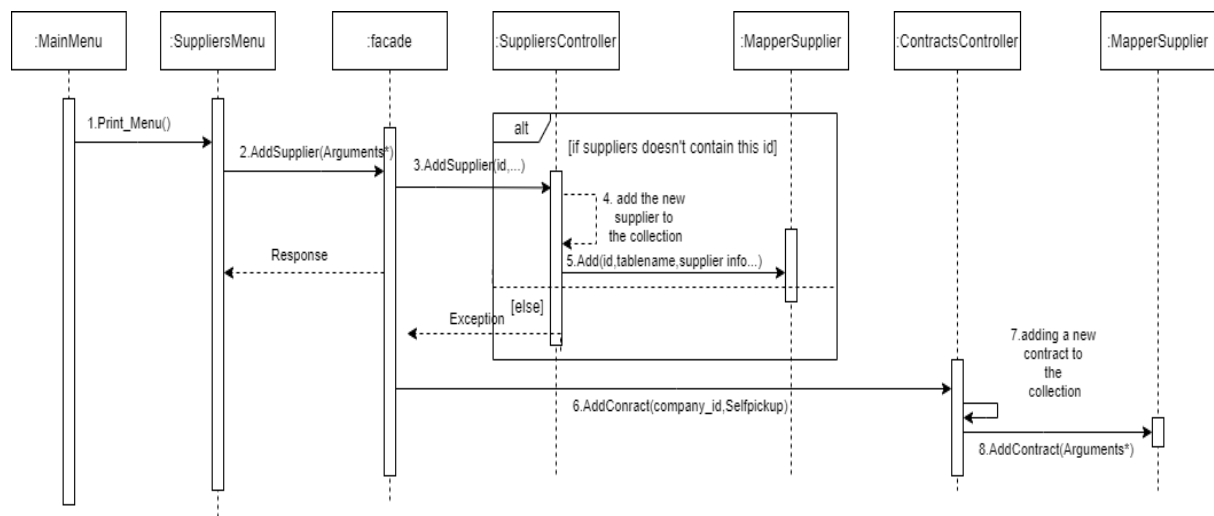


:Sequence & collaborative diagrams(2)

C – Inventory update and missing alert



D – Add a new supplier



(3) תרשים ERD:

התרשים נמצא כקובץ drawio שבאותו קובץ בשם ERDDiagram.drawio .

(4) דרישות:

טבלה 1:

Requirement	Functionality	Prioritization	Risk	Implementation
System allows opening supplier card with name of supplier, manufacturer, number of private company, bank account, payment condition and details of communication with contacts.	Functional	Must have	Low risk	
System manages a contract per every supplier.	Functional	Must have	Low risk	
Contract contains type of supplier, details of items included and price of every single item.	Non-functional	Must have	Low risk	
System manages amount reports per different suppliers who allow it as a part of their contract.	Functional	Must have	Low risk	
Amount report include discount for every single item, details of item and quantities scope of amounts which could get different discounts.	Non-functional	Must have	Low risk	
System labels every supplied item with a category number due to supplier.	Functional	Must have	Low risk	
System documents ordered items per every supplier.	Functional	Must have	High risk	
System can interface suppliers' systems.	Non-functional	Must have	High risk	
System can extract orders from suppliers.	Functional	Must have	Low risk	
המערכת תאפשר למשתמש לעדכן לכל מוצר את כמותו הנוכחית.	פונקציונלית	Must have	גבוה	

המערכת תאפשר למשתמש להעביר מוצרים מהמחסן למדפים וגם ממדפים למחסן.	פונקציונלית	Must have	גבוה	
המערכת תאפשר למשתמש להפיק דוח חוסרים עבור מוצרים שאזלו בעקבות המכירות שהתבצעו במהלך השבוע.	פונקציונלית	Must have	גבוה	
המערכת תאפשר למשתמש להפיק דוח פריטים שיש להזמין על בסיס המלאי הקיים.	פונקציונלית	Must have	גבוה	
המערכת תשלח אזהרה למשתמש עבור מוצרים שהגיעו לכמות מינימלית שהמשתמש קובע לפי זמן אספקה וביקוש המוצר.	פונקציונלית	Must have	נמוך	
המערכת תאפשר להגדיר לכל מוצר תעודת הכרה שתכיל את המספר היסודי של המוצר (ברקוד), מיקום המוצר על המדף, היצרן של המוצר, כמות כללית של המוצר, כמות במדפים וכמות במחסן, אחוזי ההנחה שהספק נותן, תאריך שימוש, מחיר לצרכן, עלות לסופר, אחוז ההנחה מהחנות אם קיימת הנחה וכמות מינימלית.	פונקציונלית	Must have	גבוה	
המערכת תאפשר למשתמש להגדיר/לעדכן קטגוריות שונות.	פונקציונלית	Must have	גבוה	
המערכת תאפשר להגדיר רשימת מוצרים שאסור למכור בגלל שהמוצר פג תוקף או פגום.	פונקציונלית	Must have	גבוה	
רשימת המוצרים שאסור למכור תכיל את המספר היסודי של המוצר, שם המוצר, סיבת מניעת המכירה והכמות.	לא פונקציונלית	Nice to have	נמוך	
המערכת תאפשר להפיק דוחות עבור פריטים פגומים.	פונקציונלית	Nice to have	נמוך	
המערכת תאפשר להגדיר עבור כל קטגוריה ראשית (קטגורית אב) קטגוריה משנית (קטגוריות בן) ומוצרים שונים.	פונקציונלית	Must have	גבוה	
המערכת תשמור עבור כל ספק את ההנחות שהוא נותן ותאריך סיום ההנחה.	פונקציונלית	Must have	גבוה	
המערכת צריכה להציג אפשרויות שונות לפי תפקיד המשתמש (קופאי, מחסנאי, מנהל סניף).	פונקציונלית	Nice to have	גבוה	
המערכת תאפשר למשתמש להזין לכל פריט כמות נוכחית, כמות במדפים, כמות במחסן.	פונקציונלית	Must have	גבוה	
המערכת תשמור על הפריטים הקיימים במלאי היכן ממוקם הפריט, מי היצרן של הפריט, כמות נוכחית, כמות במדפים, כמות במחסן.	פונקציונלית	Must have	גבוה	
When a truck weight exceeds its limits, the trip is rescheduled, one if the destinations replaced or replacing a truck or removing	Non-functional	Must have	High risk	**

some products. all of this should be reflected in the transport document.				
The system must ensure that for each shipment there is driver.	Functional	Must have	High risk	**
When a delivery departs it must be registered in the delivery management database.	Functional	Must have	Low risk	**
The truck driver must receive a numbered (trackable) document with a list of items transported to the given destination, this number must be saved in the system for tracking purposes.	Non-functional	Must have	Low risk	**
When a truck leaves the source location it is weighed first, this weight must be saved also in the delivery document.	Functional	Must have	Low risk	**
The system must issue an alert if the actual weight of a truck exceeds the maximum weight allowed.	Functional	Must have	Low risk	**
When ordering a shipment transportation, it is not possible to place a driver without a proper truck license selected for the transport.	Non-functional	Must have	Low risk	**
Truck can't take part in 2 different shipments simultaneously.	Functional	Must have	Low risk	**
Delivery could be arranged only if there is an available storekeeper	Functional	Must have	High risk	**
The system should define different shipping areas that the shipments are planned to them.	Functional	Nice to have	High risk	**
ושהידע הנ"ל יוטמע במסך השיבוצים למשמרות.	פונקציונלית	Must have	High risk	**
יש צורך לנהל רישום עבור ימים ושעות בהם כל עובד יכול לעבוד.	פונקציונלית	Must have	Medium risk	**
על המערכת לאכוף שלא יהיה ניתן לשבץ עובדים לתפקידים שאינם מוכשרים אליהם.	פונקציונלית	Must have	Medium risk	**
על המערכת לנהל את ההרשאות של כל עובד, לדוגמה עובד רגיל יוכל לראות רק את שיבוצי המשמרות שלו, בעוד שמנהל כ"א יוכל גם לשבץ למשמרות.	לא פונקציונלית	Must have	Medium risk	**
עלה הצורך לעקוב ולנהל את רשימת העובדים בחברה, שמות, ת"ז, פרטי חשבון בנק לצורך העברת משכורת, שכר ותנאי העסקה לכל עובד. כמו-כן יש לשמור את תאריך תחילת העסקה.	פונקציונלית	Must have	Low risk	**

מנהל כ"א מעוניין כי המערכת תדע מי יכול לעבוד באיזה תפקיד	פונקציונלית	Must have	Low risk	**
המנהל מעוניין לשבץ עובדים למשמרות דרך המערכת.	פונקציונלית	Must have	Low risk	**
על המערכת לאכוף שבכל משמרת יהיה מנהל משמרת.	פונקציונלית	Must have	Low risk	**
אסור לאפשר לעובד לעבוד גם בבוקר וגם בערב באותו היום.	לא פונקציונלית	Must have	Low risk	**
עובד יכול לראות את המשמרות בהן הוא עובד.	פונקציונלית	Must have	Low risk	**
תשמר עבורו ההסטוריה של המשמרות.	פונקציונלית	Nice to have	Low risk	**
יש לאפשר לעובדים להכניס ולשנות את אילוצי העבודה שלהם.	פונקציונלית	Nice to have	Low risk	**
על המערכת לשמור ולאפשר לשנות את רשימת התפקידים הדרושים עבור כל 11 המשמרות בשבוע.	פונקציונלית	Nice to have	Low risk	**
עובד יכול לראות את הפרטים האישיים שלו במערכת.	פונקציונלית	Nice to have	Low risk	**
יש לשמור את כל הנתונים של העובדים במערכת גם אם הם כבר עזבו את הרשת.	פונקציונלית	Nice to have	Low risk	**
כדאי שהמערכת תציג למנהל כוח האדם סידור אפשרי למשמרות כדי שיוכל להיעזר בו.	פונקציונלית	Nice to have	High risk	**
If the storekeeper didn't update the order so the order is going to be sent as the old order.	Non-functional	Must have	Low risk	
System should be able to send orders automatically due to the lack in the store.	Functional	Must have	Low risk	
System can send two different types of orders: 1. Order by the lack in the store. 2. A periodic order	Non-functional	Must have	Low risk	
In order to send an order by a lack in the store, the store system must send an item's details.	Functional	Must have	Low risk	
It's must to update the periodic order at least one day before the supplier prepare the order.	Non-functional	Must have	Low risk	
System should confirm that the amount of item in store has to be more than the declared minimum amount of item in store of each item after the arriving of order.	Non-functional	Must have	Low risk	
System should search for the minimum price in the quantity report of every supplier and order the specific item from the chipper one.	Functional	Must have	Low risk	

** דרישה של המודלים האחרים שלא ממשים בשלב הזה אילה בשלב הבא.

טבלה 2: שאלות שאינן משפיעות על המימוש

האם לכל סניף בחברה שלנו יש בסיס נתונים?
האם צריך לשמור לכל פריט את מספר המוצר המיוצג בחנות?
האם צריך להגדיר תנאים מסוימים למכירת מוצרים מסוימים?
"אחד מעובדי החנות" שתואר בתיאור המודל, הכוונה לכל עובד בלי קשר לתפקיד שלו?
האם מוצר מסוים יכול להשתייך ליותר מקטגוריה אחת?
מי אחראי על איתור פריטים פגומים או שפג תוקפם, וברגע שמוצאים את המוצרים אלה? ומה צריכים לעשות בהם (להחזיר לספק\ לזרוק לפח)?
האם אפשר לבטל הנחה לפני תום התאריך המפורסם (למשל חוסר מלאי)?
איך מחשבים הנחה עבור מוצר מסוים, האם סוכמים את ההנחה שנתן הספק וגם ההנחה שהחנות נותן או מחשבים ההנחה המינימלית ביניהם או בשיטה אחרת?
האם אפשר לשנות את מיקום המוצרים הקבוע במדפים?
האם צריך לשמור עותק לדוחות שמפיקים?
האם ייתכן מוצר שלא שייך לאף קטגוריה?
האם נשתמש בבסיס נתונים רלציוני? ג'ייסון?
האם יש לטעון את נתוני משמרות עבר ועובדים בדימוס בכל פעם שהמערכת נטענית? (באיזו תדירות משתמשים במידע הזה?)
Do drivers prefer specific transport area?
Which algorithm should we apply in case of a truck that exceeds its maximum weight? (there are 3) How should we know which one to use and to which one should we give high priority?
האם המערכת חייבת גם להזהיר על המוצרים שמגיעים לכמות מינימלית גם שהיא מזמינה באופן אוטומטי?

טבלה 3: שאלות שמשפיעות על המימוש

שאלה	תשובה
מתי מבצעים ההנחה, וכמה היא?	בעת ביצוע ההזמנה. גובה ההנחה נקבע על פי "כתב כמויות" בהסכם אשר מגדיר לכל פריט מהי ההנחה שתנתן לפריט בעת רכישות בהיקף גדול. כמות ההנחה משתנה על פי כמות הפריטים שהחברה הזמינה מהספק באותה הזמנה.
כמה תחומים הספק יכול לעבוד בהם עם חברה אחת?	כמה שהוא רוצה.
האם הכוונה בחברות השונות הם היצרנים?	כן.
מי הם אנשי הקשר, ומה הכוונה בפרטי התקשורת בניהם לבין הספק בכרטיס הספק?	איש הקשר אצל הספק הוא אותו אדם שאני מתקשר אליו אם צריך לבצע הזמנה דחופה או להזמין פריט שלא נמצא בהסכם או אם יש בעייה באחת ההזמנות שהספק ביצע. לצורך

מהם פרטי הפרטים?	יצירת הקשר אני צריך שם, מספר טלפון, וכתובת אימייל.
מי קובע מאיזה ספק מזמינים ?	המספרים הקטלוגיים שמות ומחירי הפריטים אותם אנחנו קונים אצל ספק זה.
האם הסכם הוא קבוע ולא משתנה?	אני.
האם צריך לאפשר ביטול / החלפה בזמנים המועדפים של העובד?	כמובן שהסכמים יכולים להשתנות. הספק יכול להפסיק את הפעילות שלו או שאנחנו נמצא ספק אחר שמשתלם יותר לקנות ממנו. זה שוק חופשי בסך הכל...
האם צריך לאפשר ביטול / החלפה לאחר שיבוץ?	כן.
האם קביעת מספר התפקידים שצריך לכל משמרת נעשית לכל משמרת באופן מיוחד או קבוע לכל משמרת בשבוע?	כן.
האם הזמנים המועדפים של כל עובד יקבעו כל פעם לזמן ספציפי או פעם אחת באופן קבוע עבור כל משמרת בשבוע ובמידת הצורך העובד יוכל לשנות את העדפותיו?	באופן קבוע לכל משמרת בשבוע, יהיו 11 משמרות בשבוע ולכל אחת מהן העדפה, כאשר מנהל כוח אדם יכול לשנות אותן במקרה וימצא לנכון.
האם יש דרך זיהוי מסוימת?	קבועים עבור המשמרת (עם אפשרות לעדכון).
האם ה"ידע המוטמע" כולל גם את מגבלות מועדי השיבוץ של כל עובד?	מס' זהות ותפקיד ללא סיסמה.
האם המערכת צריכה למנוע שיבוץ של עובד לתפקיד שאיננו מוסמך אליו?	כן.
לגבי מנהל המשמרת - האם המערכת צריכה למנוע קביעת משמרת גם מבלי שהמנהל שיבץ מנהל משמרת?	כן בכל תוקף
מי ישתמש במערכת? עובדים או מנהלים או שניהם? במידה וגם עובדים ישתמשו במערכת האם יש לא לאפשר להם דברים כמו הזנת נתונים?	כן.
מי יכול להכניס את הזמנים המועדפים? עובדים או מנהל?	עובדים ומנהלים. העובדים יוכלו אך ורק לצפות במידע שלהם, במשמרות שלהם, ולהזין זמני עבודה מועדפים, בעוד שמנהלים יוכלו להזין נתונים, לשבץ למשמרות ולעשות שינויים במידע.
האם צריך לאפשר למנהל כוח האדם לבחור אילו תפקידים בכל משמרת ספציפית (למשל משמרת בימי שני בערב) ואז לאכוף את זה כאילו מערכת? או שהכוונה היא לתת למנהל המשמרת פשוט את החופש לבחור אילו תפקידים שהוא רוצה בכל משמרת מבלי לאכוף זאת?	עובדים ומנהל כ"א.
"עלה הצורך לעקוב ולנהל את רשימת העובדים בחברה, שמות, ת"ז, פרטי חשבון בנק לצורך העברת משכורת, שכר ותנאי העסקה לכל עובד." - מהם תנאי העסקה של עובד שעלינו לשמור?	זוהי המלצה בלבד. נאפשר לו לשבץ יותר/פחות עובדים כשיחפוץ בכך.
	גובה שכר, קרן השתלמות, מספר ימי חופש וימי חופשת מחלה בחודש. מספיק ברגע.

לא עושים הסכם חדש עבור כל הזמנה.	האם כל הזמנה צריכה להיבדק על ידי ההסכם של אותו ספק? או כל פעם יש הסכם חדש(לכל הזמנה)?
עד עכשיו היה לנו רק הסכם פעיל אחד עם ספק אחד.	כמה חוזים יכולים להיות עם אותו ספק?
Driver is considered one of the company employees, so he got a unique id.	In case of multiple drivers with the same name, how can we distinguish between them?
Yes, the first area who is in stock shortage got the first treatment.	In case of stock shortage in various areas, do we prioritize shipment to each area?
Yes.	If there is no such driver with proper license for the selected truck, should we cancel the delivery?
No, there will be no changes during the shipment.	Are there any changes from supplier's side that affect shipment transportations while performing them?
When the driver finishes his delivery, he alerts the admin.	How the system knows that a driver finished the delivery for rescheduling a new one? Is there a constant time for each delivery? Is there a single delivery for each driver per day?
Drivers are the company employees, so they work globally and not for a specific branch.	Are drivers belong to a specific branch or they work globally?
Yes, since he got a licensee for X ton, then he can drive a truck of Y ton such that $Y < X$.	Can a truck driver drive more than one truck?
A driver can have one delivery per shift.	How long does it take to transport a delivery? Or the driver can just do one delivery per shift?
בינתיים אין (כשנאחד את המודל עם הובלות נממש את זה)	האם ישנה הגבלה מבחינת מספר פריטים/משקל וכדומה בהזמנה?
לפצל להזמנות שונות	בהנחה כי אי אפשר לבצע הזמנה (עקב משקל/כמות או כל סיבה אחרת) איך המערכת אמורה להגיב?
כרגע לא(כאשר נאחד את המודלים נממש את האפשרות הזאת)	האם עלינו לוודא שאכן יש אפשרות לספק להגיע בימים שהוזנו לו ע"י המשתמש? (בהזמנות שאינן קבועות)
המערכת תבטל את ההזמנה ותדחה אותה בשבוע.	מה קורה אם הזמנה קבועה מתעכבת?
כרגע כן (כשנאחד עם עובדים אולי לא)	האם כל עובד יכול לבצע הזמנה מספק?
כן מוצרים שמוגבלים לגיל מסוים, למשל מכירת סיגריות ואלכוהול מותנת בהצגת תעודת זהות ובדיקת גיל (מעל 18)	האם צריך להגדיר תנאים מסוימים למכירת מוצרים מסוימים?
הדרישה לא הייתה מדויקת, לא הגיוני שכל עובד בסופר יש להם אותו יכולות ביצוע	"אחד מעובדי החנות" שתואר בתיאור המודל, הכוונה לכל עובד בלי קשר לתפקיד שלו?

והחלטה, למשל עובד חדש שעדיין לא מכיר את כל המוצרים, לכן הכוונה בדרישה לעובד ותיק	
לא, כך אנחנו מאבדים את אימון הלקוחות שהגיעו בפרט בגלל מבצע מסוים ולכן צריך לדאוג תמיד להזמין לפני שנגיע למצב של חוסר	האם אפשר לבטל הנחה לפני תום התאריך המפורסם (למשל חוסר מלאי)?
לא מומלץ, כל מוצר ישתייך לקטגוריה אחת	האם מוצר מסוים יכול להשתייך ליותר מקטגוריה אחת?
כל עובד שמאתר פריט פגום (מכל סיבה) צריך לעדכן את זה במערכת, אין חשיבות למוצרים פגומים ולכן זורקים אותם	מי אחראי על איתור פריטים פגומים או שפג תוקפם, וברגע שמוצאים את המוצרים אלה? ומה צריכים לעשות בהם (להחזיר לספק? לזרוק לפח)?
בגלל שאותו מוצר יכול לקבל אחוזי הנחה ממקורות שונים לכן ההנחה הכוללת צריכה להיות סכום של כל ההנחות עם בדיקה שההנחה הכוללת לא תגיע ליותר מ-1%	איך מחשבים הנחה עבור מוצר מסוים, האם סוכמים את ההנחה שנתן הספק וגם ההנחה שהחנות נותן או מחשבים ההנחה המינימלית ביניהם או בשיטה אחרת?
כן ולפעמים מומלץ, כאשר יש הנחה על מוצר מסוים או מוצר עם כמות גדולה במלאי ועם תאריך תפוגה קצר	האם אפשר לשנות את מיקום המוצרים הקבוע במדפים?
כדי לשמור על הפורמט ומבנה המערכת נגדיר קטגוריה כללית שתכיל את כל המוצרים שלא שייכים לקטגוריה באופן ברור וכך כל המוצרים ישתייכו לקטגוריה	האם ייתכן מוצר שלא שייך לאף קטגוריה?
כן מומלץ, המידע הזה יישאר חשוב למשתמש, לדוגמא אם המשתמש שם לב שמוצר מגיע לכמות מינימלית הרבה פעמים אז יעדיף להגדיל את הכמות המינימלית	האם המערכת חייבת גם להזהיר על המוצרים שמגיעים לכמות מינימלית גם שהיא מזמינה באופן אוטומטי?

תרשים מחלקות:

התרשים נמצא כקובץ drawio שבאותו קובץ בשם diagram.drawio .

תרשים אובייקט:

סיפור1:(satisfiable)

לרגל חג הפסח מנהל סניף סופר לי מבקש מהמחסנאי להכין הזמנה לכל המוצרים שהגיעו לכמות מינימלית ולשלוח את ההזמנה לפי הספקים המתאימים לכל מוצר, וגם לאתר את כל המוצרים שתאריך התפוגה שלהם הסתיים ולהוריד את הכמות שלהם מהכמות הכללית שנמצאת בחנות.

המחסנאי נרשם למערכת ומופעלת מערכת סריקה של המוצרים עם כמות מינימלית והתגלה את המוצרים הבאים: במבה, ספגטי וארז, בנוסף לכך התגלה 50 יחידות

במבה של חברת אסם שתאריך התפוגה שלהם הסתיים. שלושת המוצרים שייכים לחברת אסם שני ספקים, אחד מהם מגיע כשישנה הזמנה בלבד שנקרא לו ספק א' והשני איננו מגיע בכלל כלומר איסוף עצמי בלבד ונקרא לו ספק ב', החוזים עם הספקים הללו מוגדרים כך ש: לספק א' יש ימים קבועים והוא מייצג את חברת אסם שמספרה 77 בתחום התזונה שמוכרת במבה, ארז וספגטי שכל מוצר מאלו עולה 10 ש"ח ומספריהם הקטלוגים 1, 2 ו- 3 בהתאמה. לספק ב' אין ימים קבועים והוא מייצג את חברת באג שמספרה 99 בתחום החשמל שמוכרת את המוצרים: מקלדת, עכבר ומסך שכל פריט מאלו עולה 30 ש"ח ומספרם הקטלוגי 4, 5 ו- 6 בהתאמה. לספק א' יש אפשרות תשלום אך ורק במזומנים לעומת ספק ב' שמאפשר תשלומים באמצעות העברה בנקאית או הוראת קבע

*התרשים נמצא כקובץ drawio שבאותו קובץ בשם objectDiagram1.drawio .

סיפור:

לפני הגעת ההזמנות שהוזמנו לרגל החג, משרד הבריאות מזהיר מגל קורונה רביעי וכתוצאה מכך הם ממליצים על סגר כללי ולכן סופרלי מחליטה לבטל את כל ההזמנות אך לסיבות לא ברורות סופרלי מצליחה לבטל אך רק את ההזמנות שלה עם ספק א' והזמנה של ספק ב' נשארת כתוצאה לכך ספק א' מודיע על הפרת חוזה ותום התקשרות עם סופרלי בגלל ההתייחסות הגסה ולכן התקשרות מסתיימת בינו לבין סופרלי.

וספק ב' מפסיק את כתב הכמויות מההסכם שלו עם סופרלי.

*התרשים נמצא כקובץ drawio שבאותו קובץ בשם objectDiagram2.drawio .

שינויים:

1. בעבודה הקודמת המחסנאי היה חייב להפעיל פונקציה מיוחדת שעוברת על המוצרים ועבור כל מוצר להכניס כמות להזמנה, בעבודה הנוכחית עם עליית המערכת, פונקצית *minAmountOrder* מופעלת באופן אוטומטי והיא סורקת מוצרים עם כמות מינימלית ומזמינה לפי כמות מוגדרת מראש (*default value*) ואם לא מוגדרת כמות אז מבקשת מהמחסנאי להכניס כמות.
2. בעבודה הקודמת לא היה מבנה נתונים ועם כל עלייה של המערכת, המערכת מתאפסת מחדש, בעבודה הנוכחית הגדרנו *dataAccessLayer* שבה מאחסנים מידע על קטגוריות השונות, מוצרים שונים, חברות המייצרות, והזמנות.
3. עבור מודל הספקים הוספנו שכבה חדשה שנקראת *data access layer* שהיא מכילה *mappers* ואובייקטים עבור השכבה הזו.
4. העברנו את שמירת האובייקטים ב *ram* מה *controller* ל *mapper* ועכשיו ה *controller* מחזיק את *mapper*.
5. עבור שני המודלים הוספנו *façade* כללי שמשתמש בכל *façade* משני המודלים ולכן הוא מחזיר לשכבת *presentation layer* את ה *response* המתאים.

