

מסמך אפיון ראשוני לפרויקט מערכת סופרמרקט מבוסס JAVA

הקדמה

השילוב של נוחות וטכנולוגיה הפך לבעל חשיבות עליונה בשינוי האופן שבו עסקים פועלים ומתקשרים עם הלקוחות שלהם. מתוך הכרה במגמה זו, הפרויקט שלנו שואף לפתח מערכת סופרמרקטים מקיפה מבוססת אינטרנט. פלטפורמה חדשנית זו נועדה לדמות חווית סופרמרקט פיזית, המאפשרת ללקוחות לגלוש ולרכוש מגוון רחב של מוצרים באינטרנט בקלות וביעילות.

מטרת הפרויקט

המטרה העיקרית של פרויקט זה היא ליצור מערכת סופרמרקט מקוונת ידידותית למשתמש, יעילה וחזקה. מערכת זו לא רק תציע חווית קניה חלקה למשתמשים אלא גם תשלב קצה אחורי מתוחכם לניהול מלאי ועיבוד הזמנות.

קהל יעד

המערכת שלנו מותאמת לצרכים של מגוון רחב של משתמשים, כולל:

צרכני קצה: אנשים המחפשים דרך נוחה ויעילה לקנות מצרכים ומוצרי סופרמרקט אחרים באינטרנט.

מנהלי סופרמרקטים: צוות האחראי על ניהול רשימות מוצרים, מלאי ועיבוד הזמנות.

מנהלי מערכת: צוות טכני המפקח על ביצועי המערכת והאבטחה הכוללים.

היקף הפרויקט

מסמך זה מתאר את ההיבטים הבסיסיים של מערכת הסופרמרקטים המבוססת על JAVA

תיאור פונקציונלי: סקירה מפורטת של יכולות המערכת ואינטראקציות המשתמש.

מסגרת טכנולוגית: הטכנולוגיות והכלים הספציפיים המשמשים לבנייה ולתחזוקה של המערכת.

ארכיטקטורת מערכת: מתווה של התכנון המבני של המערכת, כולל רכיבי קצה וחלק אחורי.

עיצוב ממשק משתמש: תיאורי המסכים השונים וזרימת המשתמש של המערכת.

עיצוב מסד נתונים: המבנה והסכמה של מסד הנתונים, המדגישים את הקשרים והתכונות של ישויות נתונים שונות.

תיאור פונקציונלי

ממשק משתמש: דפי HTML עם ReactJS and Bootstrap לתוכן אינטרנט דינמי.

בחירת מוצרים: מאפשרת למשתמשים לעיין במגוון מוצרים, להציג תיאורים מפורטים ולהוסיף פריטים לסל קניות.

תהליך התשלום: משתמשים רשומים יכולים לבצע הזמנות מסל הקניות שלהם.

ניהול מלאי: מעקב בזמן אמת כדי להבטיח שהמוצרים זמינים לרכישה.

רישום ואימות משתמשים: רק משתמשים רשומים יכולים לבצע רכישות.

פאנל ניהול: לניהול מערכת, כולל עדכון פרטי מוצר ומעקב אחר הזמנות.

בקרת גישה: רמות שונות של גישה ללקוחות, משתמשים רשומים ומנהלי מערכת.

טכנולוגיות

Front-End: ReactJS, HTML, and Bootstrap.

Back-End: Jav, Spring Boot, RestAPIs

Database Interaction: Hibernate, JPA (Java Persistence API) for database operations.

Database: MYSQL.

Server: Tomcat server.

ארכיטקטורת מערכת

מודל שרת לקוח: ממשק מבוסס אינטרנט (לקוח) באינטראקציה עם שרת מבוסס Java

ארכיטקטורת MVC: דפוס דגם View-Controller-להפרדת ההיגיון העסקי, ממשק המשתמש ומודל הנתונים.

שכבת מסד נתונים: אחראית על אחסון ואחזור נתונים.

השכבות במערכת:

התקני לקוח: שכבה זו כוללת מגוון של התקני משתמש כגון מחשבים שולחניים, מחשבים ניידים, טאבלטים וסמארטפונים. משתמשים מקיימים אינטראקציה עם מערכת הסופרמרקט באמצעות דפדפן אינטרנט במכשירים אלה. כאן, הטכנולוגיות בצד הלקוח כמו HTML, CSS ו-JavaScript משמשות ליצירת ממשק משתמש דינמי ומגיב.

שרת אינטרנט (Apache Tomcat): כאשר משתמש מבצע פעולה באתר, כגון חיפוש מוצר או הכנסת פריט לעגלת הקניות שלו, הבקשה נשלחת לשרת האינטרנט. Apache Tomcat מקבל את הבקשה וקובע את הפעולה

המתאימה לנקוט. הוא מגיש תוכן סטטי ישירות בחזרה ללקוח ומעביר בקשות דינמיות לשרת היישומים.

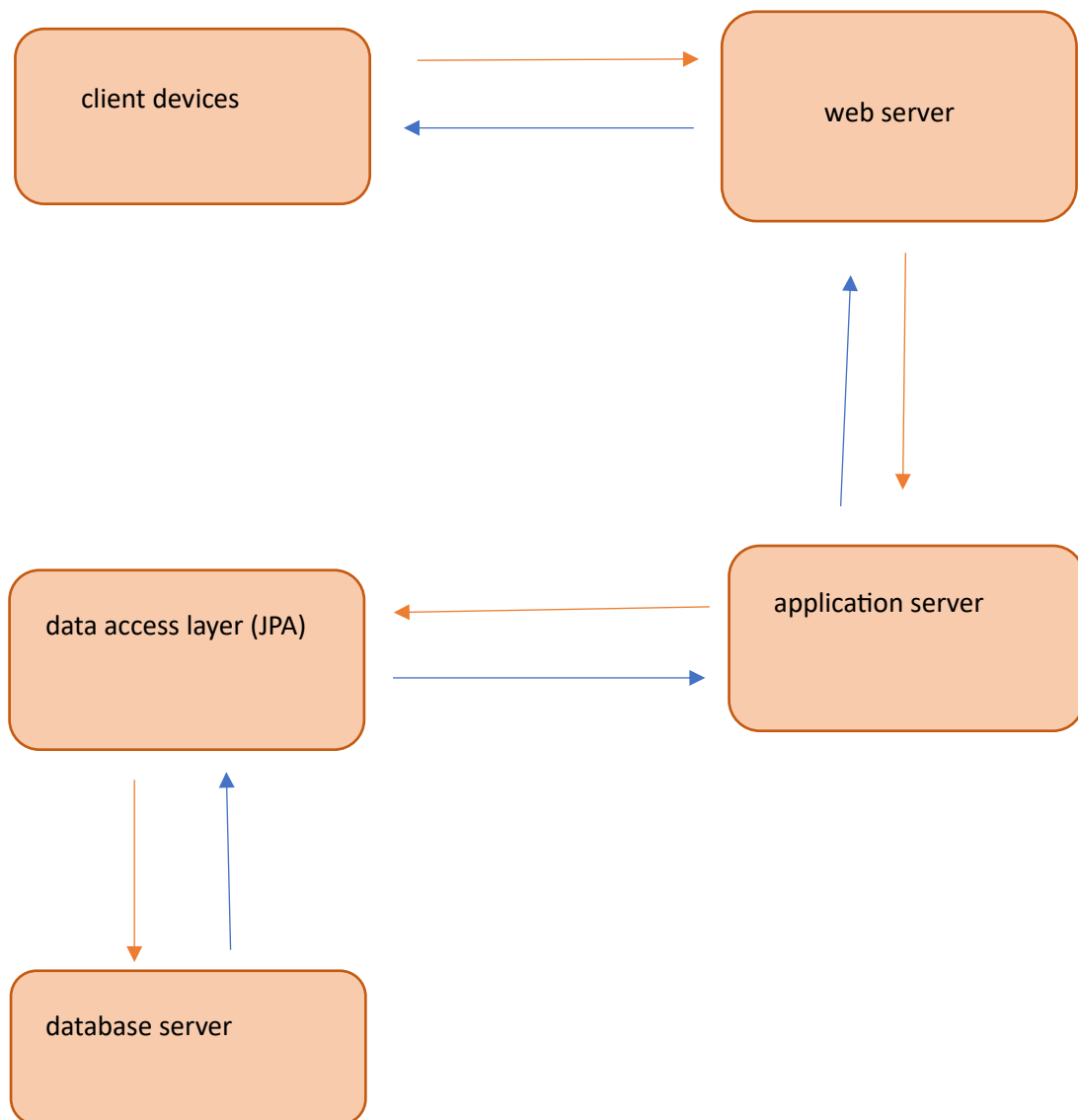
שרת יישומים (Java, SpringBoot): שרת היישומים הוא המקום שבו שוכן ההיגיון העסקי הליבה של מערכת הסופרמרקט. הוא מעבד את הבקשות המתקבלות משרת האינטרנט. שכבה זו אחראית על ביצוע משימות עורפיות כגון ניהול הפעלות משתמש, ביצוע כללים עסקיים, עיבוד תשלומים וטיפול בפעולות עסקאות אחרות.

שכבת גישה לנתונים (JPA): עבור בקשות הדורשות אחזור נתונים או מניפולציה, שרת היישומים מתקשר עם שכבת הגישה לנתונים. שכבה זו מפשטת את פעולות מסד הנתונים הבסיסיות ומספקת API פשוט ומאוחד לשרת היישומים לאינטראקציה עם מסד הנתונים. הוא משתמש ב-JPA (Java Persistence API) כדי לבצע פעולות כמו שאילתות במסד הנתונים, עדכון רשומות מלאי או אחסון מידע משתמש.

שרת מסד נתונים (MySQL): זוהי שכבת האחסון המתמשכת שבה מאוחסנים כל הנתונים הקשורים למוצרים, משתמשים, הזמנות ומידע הכרחי אחר. שרת מסד הנתונים מגיב לשאילתות שיזמה שכבת הגישה לנתונים, מבצע את העסקאות במסד הנתונים ושולח את התוצאות בחזרה דרך המחסנית כדי שיוצגו בסופו של דבר למשתמש.

זרימת נתונים: זרימת הנתונים מתחילה בהתקני הלקוח, עוברת לשרת האינטרנט לעיבוד ראשוני, לאחר מכן לשרת היישומים לביצוע לוגיקה עסקית, דרך שכבת הגישה לנתונים לאינטראקציות בבסיס הנתונים, ולבסוף לשרת מסד הנתונים לאחסון נתונים ושליפה.

אינטגרציה: רכיבים אלו משולבים באופן הדוק, כאשר כל שכבה מתוכננת לתקשר ביעילות עם השכבות הסמוכות. הפרדת הדאגות הזו מבטיחה שהמערכת מודולרית, ניתנת להרחבה וניתנת לתחזוקה.



תיאורי מסך

דף הבית: תפריט – קטגוריות, מוצרים, סל קניות, תוספת למשתמש אדמין של ניהול.

דף מוצרים: תצוגה מפורטת של כל מוצר עם תיאור, מחיר ואפשרות הוספה לסל.

דף סל: סקירה כללית של פריטים נבחרים, מחיר כולל ואפשרות לקופה.

דף תשלום: פרטי תשלום ומשלוח עבור משתמשים רשומים.

דף הרשמה/כניסה: אימות משתמש.

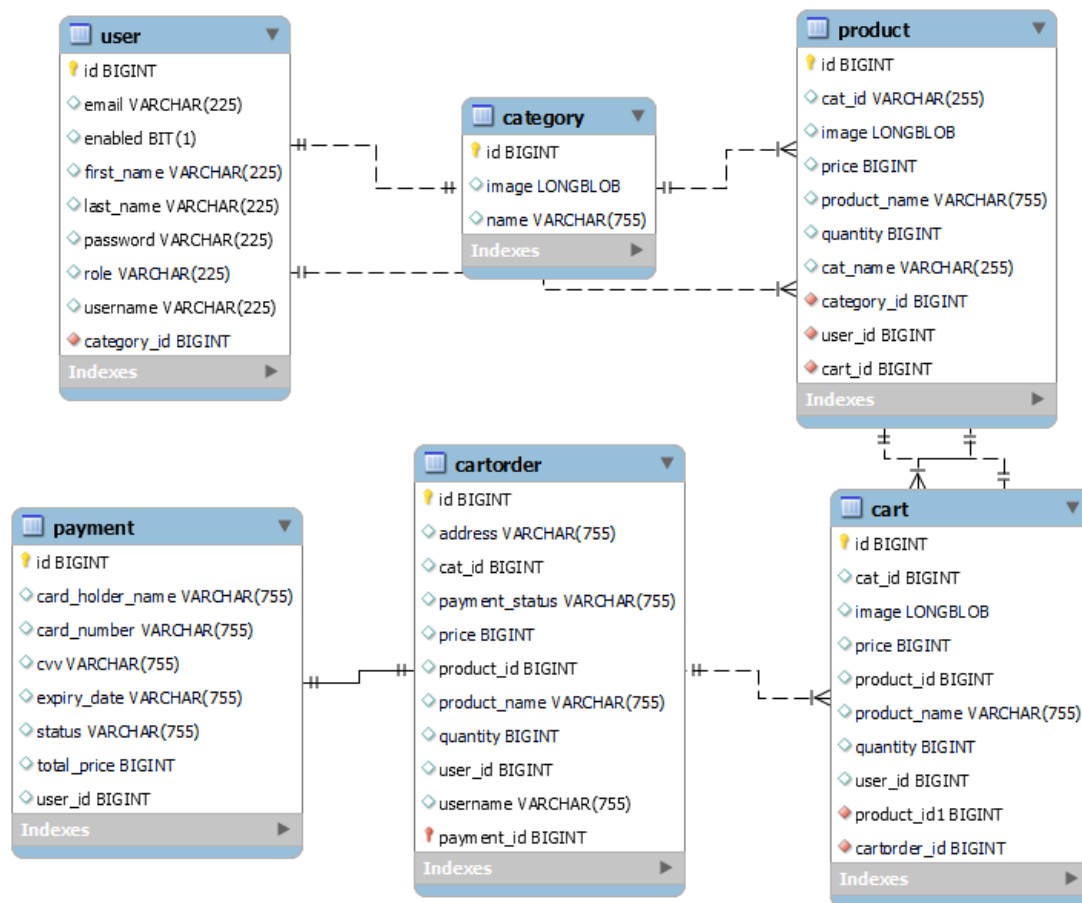
לוח מחוונים לניהול: ממשק לניהול מוצרים והזמנות.

עיצוב מסד נתונים

מסד הנתונים של מערכת הסופרמרקטים יהיה בנוי כדי לאחסן, לאחזר ולנהל ביעילות נתונים הקשורים להיבטים שונים של חווית הקנייה המקוונת. זה כולל מידע על משתמש, פרטי מוצר, היסטוריית הזמנות וניהול מלאי.

ישויות: מוצרים, משתמשים, הזמנות, פרטי הזמנה, קטגוריות וכו'.

טבלאות מפתח ומערכות היחסים ביניהן



סכימת מסד נתונים

טבלת המשתמשים היא מרכזית לניהול פרופילי לקוחות ומנהלים.

טבלת המוצרים מקושרת לקטגוריות, מה שמאפשר ניהול מוצר מסודר.

טבלת ההזמנות מלכדת את הפרטים של עסקאות שבוצעו על ידי משתמשים.

טבלת cartorder מהווה גשר בין הזמנות למוצרים, חיונית לניהול אילו פריטים כלולים בכל הזמנה.

קבצי המערכת:

קבצי צד שרת

שם קובץ	תיאור קובץ
Application module	
EcomApplication.java	הרצת האפליקציה בצד שרת
config module	
JwtAuthenticationEntryPoint.java	רכיב ביישום Spring Boot המטפל בגישה לא מורשית
JwtAuthenticationFilter.java	שימוש בפילטר על מנת לאפשר אותנטיקציה למשתמשים
MySecurityConfig.java	קונפיגורציית אבטחה
Controllers module	הכניסה של הUI לצד שרת מבצעת דרך הקונטרולרים הדפדפן שולח בקשות RestAPI ולפי הurl שנשלח הקונטרולרים יודעים לאן לפנות ומכאן מתחיל הניהול של השרת
CartController.java	קונטרולר של עגלה
CategoryController.java	קונטרולר של קטגוריה
JwtController.java	קונטרולר של JSON Web Token
OrderController.java	קונטרולר של הזמנה
PaymentController.java	קונטרולר של תשלום
ProdutController.java	קונטרולר של מוצר
UserController.java	קונטרולר של משתמש
data transfer object module	אובייקט העברת נתונים (DTO) הוא אובייקט הנושא נתונים בין תהליכים.
CartResponseDTO.java	תגובת עגלה
CategoryResponseDTO.java	תגובת קטגוריה
OrderResponseDTO.java	תגובת הזמנה
PaymentResponseDTO.java	תגובת תשלום
ProductResponseDTO.java	תגובת מוצר
UserResponseDTO.java	תגובת משתמש
Entities module	
CartEntity.java	ישות עגלה
CategoryEntity.java	ישות קטגוריה
OrderEntity.java	ישות הזמנה
PaymentEntity.java	ישות תשלום
ProductEntity.java	ישות מוצר
RequiredResponse.java	משמש כדי להכיל הודעת שגיאה
User.java	ישות משתמש
Handle Exception module	
CustomizedResponseEntityExceptionHandler.java	ניהול זריקת שגיאות
ExceptionResponse.java	תגובת שגיאה
OutOfStockException.java	טיפול בסיום המלאי
RecordNotFoundException.java	טיפול ברקורד שלא נמצא
Model	

CustomUserDetails.java	טיפול בפרופיל של משתמש
JwtRequest.java	משמשת להקיף את הנתונים הנדרשים לבקשת אימות (JSON Web Token (JWT)
JwtResponse.java	משמש כדי להכיל את התגובה שנשלחה בחזרה ללקוח לאחר אימות מוצלח
Role.java	הגדרת סוגי משתמשים
Repository module	מודול שיורש מ JpaRepository כדי לאפשר לישויות במערכת להשתמש בהתנהגויות של הממשק המשמש לביצוע פעולות CRUD (יצירה, קריאה, עדכון, מחק) ושאליות מותאמות אישית
CartRepository.java	שימוש בממשק על אובייקטי עגלה
CategoryRepository.java	שימוש בממשק על אובייקטי קטגוריה
OrderRepository.java	שימוש בממשק על אובייקטי הזמנה
PaymentRepository.java	שימוש בממשק על אובייקטי תשלום
ProductRepository.java	שימוש בממשק על אובייקטי מוצר
UserRepository.java	שימוש בממשק על אובייקטי משתמש
Service module	
CartService.java	ניהול פונקציות של עגלה
CategoryService.java	ניהול פונקציות של קטגוריה
CustomUserDetailsService.java	ניהול פונקציות של פרופילי משתמשים
OrderService.java	ניהול פונקציות של הזמנות
PaymentService.java	ניהול פונקציות של תשלום
ProductService.java	ניהול פונקציות של מוצר
UserService.java	ניהול פונקציות של משתמשים
Utility module	
DataUtility.java	מספק שיטות שונות למניפולציה והמרת נתונים, במיוחד תאריכים ומחרוזות
DataValidator.java	מחלקת שירות המספקת שיטות שונות לאימות נתוני קלט, כגון מחרוזות, מספרי טלפון, כתובות דואר אלקטרוני, תאריכים ועוד
JwtUtil.java	מחלקת שירות המספקת שיטות להפקה, אימות וניתוח של (JSON Web Tokens (JWT. זהו חלק חיוני ביישום אימות מבוסס JWT ביישום Spring Boot
general	
application.properties	קובץ זה מכיל הגדרות תצורה עבור האפליקציה, כולל פרטי חיבור למסד נתונים, הגדרות JPA (Java Persistence API), הגדרות שרת והגדרות העלאת קבצים מרובי חלקים.
pom.xml	קובץ שמכיל מידע על הפרויקט ופרטי קונפיגורציה ונחוץ בכל פרויקט של mvn על מנת לבנות את הפרויקט

קבצי צד לקוח:

שם קובץ	תיאור קובץ
Components module	
Cart.js	מייצג עגלה
Categories.js	מייצג קטגוריה
CategoryList.js	מייצג רשימה של קטגוריות
DeleteCartProduct.js	מחיקת מוצר מעגלה
DeleteCategory.js	מחיקת קטגוריה
DeleteProduct.js	מחיקת מוצר
DeleteUser.js	מחיקת משתמש
Home.js	מייצג עיצוב של דף הבית
Login.js	התחברות למערכת
MainNavigation.js	מייצג את התפריט הראשי של האתר
Order.js	מייצג הזמנה
OrderHistoryList.js	מייצג רשימה של הזמנות קודמות
Payment.js	מייצג תשלום
ProductList.js	מייצג רשימה של מוצרים
Products.js	מייצג מוצר
Signup.js	הרשמה לאתר
Userlist.js	מייצג רשימה של משתמשים
style module	
style.css	דף העיצוב
Pages module	
CartPage.js	דף של עגלה
CategoriesPage.js	דף של קטגוריה
CategoryEditPage.js	דף של עריכה של קטגוריה
CategoryListPage.js	דף של קטגוריות
EditUserPage.js	דף עריכת משתמש
ErrorMessage.js	דף שגיאה
HomePage.js	דף הבית
LoginPage.js	דף התחברות
Logout.js	לחיצה על התנתקות
OrderHistoryPage.js	דף היסטוריית הזמנות
OrderPage.js	דף הזמנה
PaymentPage.js	דף תשלום
ProductEditPage.js	דף עריכת מוצר
ProductListPage.js	דף מוצרים
ProductPage.js	דף מוצר
RootLayout.js	
SignupPage.js	דף הרשמה
UserListPage.js	דף רשימת יוזרים
WelcomePage.js	דף פתיחה
Util module	
auth.js	אותנטיקציה למשתמשים
General module	

עיצוב אפליקציה	App.css
הגדרת אפליקציה	App.js
עיצוב	index.css
נקודת הכניסה לאפליקציית React. זה מגדיר את אפליקציית React על ידי רינדור רכיב השורש	index.js