**מערכת להזמנת כרטיסים   
עבודה סמינריונית בתכנות מתקדם בשפת JAVA**

**הגשה על ידי: לירז אהרוני ת.ז 311148092**

**1. מערכת להזמנת כרטיסים – מבט על**

-המערכת תופעל כאפליקציה אינטרנטית ותאפשר לאנשים לצפות בקטלוג הופעות ולהזמין כרטיסים להופעות רצויות.

-ה"הופעות" אשר יהיה ניתן להזמין אליהן כרטיסים יהיו מסוג: סרטי קולנוע, קונצרטים (הופעות מוזיקליות) והצגות

-המערכת תאפשר למנהלי המערכת להזין נתונים (כגון הופעות חדשות שנפתחות) לתוך המערכת, באמצעות האפליקציה האינטרנטית.

-המערכת תנהל רשימת לקוחות והזמנות ותפיק כרטיסים בהתאם.

-המערכת תורכב מ:

* **בסיס נתונים**

-בסיס הנתונים יהיה מאוחסן במערכת בסיסי נתונים DBMS

-יצירת בסיס הנתונים תיעשה באמצעות **טופס SQL**

-התקשורת הלוגית מול בסיס הנתונים (בצד התיכנותי) תיעשה באמצעות **טכנלוגיית JPA**

* **ממשק גרפי**

-הממשק הגרפי יבוא לידי ביטוי בדפי ה html של האפליקציה האינטרנטית

-האפליקציה תיווצר באמצעות **טכנולגיית JSF** אשר תספק שרת ואפשרות לאינטרקציה עם משתמשים שונים

-בניית הממשק הגרפי והעיצוב צפויה להיעשות באמצעות **טכנולוגיות html ו css** , אם כי יש להשתמש ב **xhtml** בעבודה עם JSF, עלי ללמוד עוד על ה xhtml כדי להבין היטב את ההבדל בין דף מסוג זה ל html סטנדרטי וכמובן לבנות את האפליקציה בהתאם

-על מנת לאפשר תצוגה חלקה למשתמש באפליקציה יש להשתמש **בטכנולוגיית Ajax**

* **קוד JAVA**

-באמצעות קוד JAVA תיבנה הלוגיקה שעומדת מאחורי עבודת האפליקציה

-תוכנית שליטה אשר תחבר בין הממשק הגרפי לבסיס הנתונים תיכתב באמצעות JAVA ותשתמש בטכנולוגיות שהוזכרו למעלה

**2. מילון מונחים ופירוט הישויות במערכת**

בצד התיכנוני והתיכנותי של המערכת קיימות מספר ישויות או מחלקות, סעיף זה נועד להסביר היטב את שמות הישויות/המחלקות ואת האינטרקציה ביניהן.

#בבסיס הנתונים של המערכת יש כמה טבלאות המהוות את הבסיס/התשתית של המערכת:

-טבלת הופעות

-טבלת מתחמי הופעות

-טבלת אולמות הופעות הנמצאים בתוך מתחמי הופעות

-טבלת כיסאות השייכים לאולמות הופעות

-טבלת לקוחות

* הופעה היא Show (תקרא גם 'אירוע')

-הכוונה לרשומה סטטית המייצגת הופעה שקיימת, ללא תאריכים קונקרטים או פרטים ספציפים

* מתחם הופעות הוא Venue (ייקרא גם 'מקום')

-הכוונה לרשומה סטטית המייצגת מקום שבו יש אולמות של הופעות, לדוגמה "סינמה סיטי ראשון-לציון" הוא Venue שבתוכו יש אולמות קולנוע.

-ייתכן מצב שבו המקום הוא-הוא המקום בו תתקיים ההופעה, מקרה זה ייפתר על ידי התייחסות למצב זה כמקום עם אולם אחד. לדוגמה "פארק הירקון" הוא Venue שבתוכו אולם אחד "פארק הירקון"

* אולם הופעות הוא Hall

-בהמשך לסעיף הקודם, לכל Venue יש Hall אחד או יותר. ייתכן מצב שבו מקום הופעות הוא אולם ההופעות, עדיין יתקיימו שתי רשומות.   
-המטרה היא שמירה על מבנה בסיס נתונים תקין על ידי אחסון מידע נפרד (בבסיס הנתונים) למתחמי ההופעות ולאולמות ההופעות מתוך הנחה שברוב המקרים המתחמים יחזיקו יותר מאולם אחד להופעות (מספר Hall יהיו שייכים לאותו Venue).

-יש שני סוגים של Hall אחד שמתקיימות בו הופעות בישיבה ואחר שמתקיימות בו הופעות בעמידה והם ניקראים SittingHall ו StandingHall בהתאם

* כיסא הוא Seat (ייקרא גם 'מושב')

-לפי הלוגיקה של התכנון, הכיסא הוא המוצר הסופי אותו אנו מוכרים ללקוחות.

-כיסא לא בהכרח יהיה מסומן – סוג האולם יגדיר אם הכיסא יהיה מסומן או לא.

-בכיסא מסומן הכוונה למספר שורה ומספר כיסא.

-כיסאות לא מסומנים יהיו כמו כיסאות בלתי-ניראים: כאשר המשתמש רוכש כרטיס למופע עמידה (או מופע עם כיסאות לא-מסומנים) הוא למעשה רוכש כיסא (במועד הספציפי של המופע) עם מספר-כיסא (מפתח) ספציפי במערכת, אך לא מקבל כפלט מספר שורה/מושב אלא רק את מספר (כמות) הכרטיסים (המושבים) שהזמין.

* לקוח הוא Customer

-כל משתמש יכול להשתמש במערכת, אך במעמד ההזמנה עליו להפוך להיות לקוח על ידי הזנת פרטים אישיים ובחירת סיסמה

-בסיס הנתונים ינהל טבלת לקוחות והאפליקציה תאפשר ללקוח חוזר להתחבר באמצעות הזנת מייל וסיסמה בלבד

#בנוסף לטבלאות התשתית המוצגות למעלה, בסיס הנתונים יחזיק טבלאות דינמיות יותר, כלומר שנוצרות בהן רשומות בתדירות גבוהה יותר, בעקבות השימוש באפליקציה.

הטבלאות הן:

-טבלת הרצות/הצגות

-טבלת מושבים-הרצות

-טבלת הזמנות

-טבלת תשלומים

-טבלת כרטיסים

* הרצה היא ShowRun (תקרא גם 'הצגה' או 'מופע')

-במערכת מאוחסנות רשומות של Show, בפועל הכרטיסים נמכרים עבור הצגה ספציפית של ההופעה, כלומר ההופעה באולם ספציפי בתאריך ספציפי מייצרת הרצה של ההופעה (הופעה-אולם-תאריך = הרצה).

-מחיר הכרטיס נקבע כתכונה (עמודה) של ShowRun

* מושבים-הרצות יכונו RunSeats

-המטרה לייצר טבלה אשר מציגה כיסאות מכורים לפי הרצות ספציפיות, הם למעשה הכרטיסים שהמשתמש בוחר

-האולם בו מתקיימת ההרצה מספק את המידע על הכיסאות הזמינים עבור הרצה מסוימות

* הזמנה היא Order

-הזמנה נוצרת עבור לקוח ובחירה של RunSeats

-התכנון הוא להחזיק את הכיסאות הנבחרים כרשימה בכל רשומה של Order

* תשלום הוא Payment

-עבור כל הזמנה יידרש תשלום

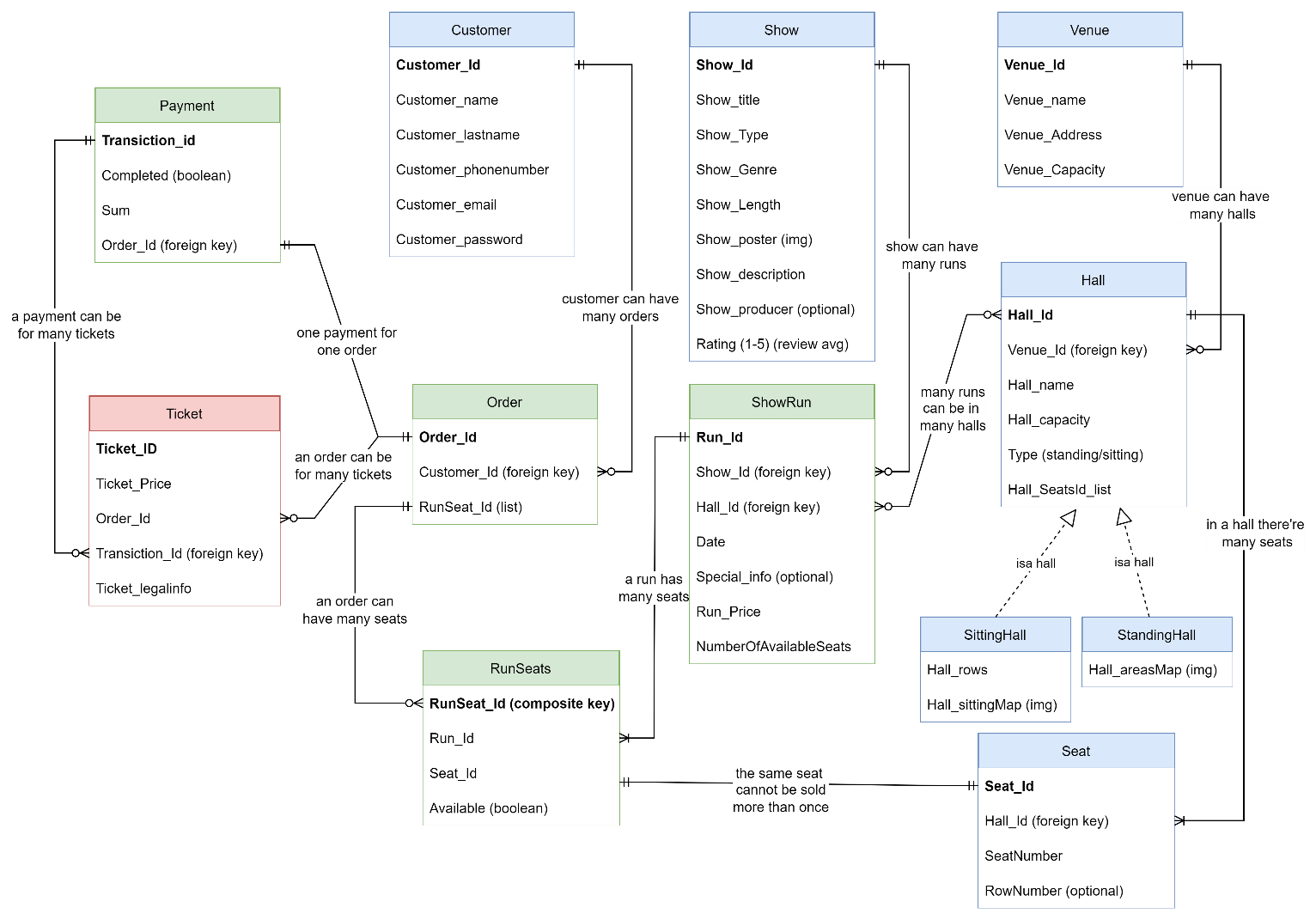
-בשלב זה, המערכת תבקש תשלום פיקטיבי, כלומר לא ייעשה תהליך אימות או סליקה כלשהו, כל נתון שיוזן במסך התשלום יתקבל כתקין

-מדגיש כי ההזמנה מחזיקה את פרטי הלקוח וכמובן התשלום הוא עבור לקוח בהתאם

* כרטיס הוא Ticket

-כרטיס הוא למעשה הזמנה שהתשלום עבורה הושלם

-זה הפלט שמוצג ללקוח בסוף תהליך הרכישה

**3. בסיס הנתונים - דיאגרמת ישויות קשרים ERD**

-בסעיף 2 יש הסבר מפורט על הישויות הקיימות במערכת

-הצבעים נועדו להפריד ויזואלית בין טבלאות שהן יותר סטטיות לעומת טבלאות שהן יותר דינמיות

-ה Hall\_Capacity מגדיר כמה כרטיסים פנויים עבור ShowRun חדש

**4. הממשק הגרפי**

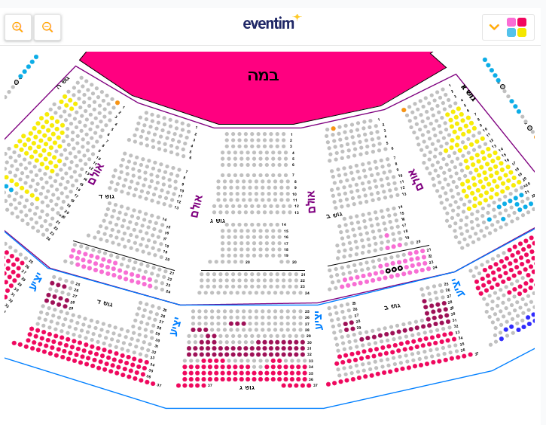
-את תיכנון העיצוב עשיתי בעזרת תוכנת Figma

הנה קישור למצגת של חלונות האפליקציה השונים, ניתן להשתמש בחלק מהכפתורים כדי לעבור מחלון לחלון (לחיצה במקום רנדומלי תדגיש כפתורים פעילים):

<https://www.figma.com/proto/PcVQwFR1UEcjwTOPmdji5B/TicketsAppLiraz?node-id=3%3A7&scaling=scale-down&page-id=0%3A1&starting-point-node-id=3%3A7>

\*במידה והלינק לא תקין, או שיש קושי עם שימוש במצגת, מצורף למייל-התכנון קובץ PDF המציג את חלונות האפליקציה.

-מנקודת המבט שלי, העיצוב עבורי יהיה החלק המאתגר ביותר בפרויקט זה, מצד אחד אני רואה בו חלק חשוב מאוד באפליקציה אינטרנטית (ולכן לא ארצה לעשות עיצוב ברמה נמוכה), מצד שני עלי לרכוש עוד כלים וידע על מנת לכתוב עיצוב ב html ו css ברמה גבוהה, ייתכן כי התכנון העיצובי שלי שאפתני מידי, אך השתדלתי לשמור עליו ריאלי ככל האפשר.

-אתייחס לנקודה אשר מוצגת לראשונה ב ERD . הייתי רוצה לאפשר את בחירת המושבים בעזרת מפת אולם שבה ניתן לבחור מושבים על ידי לחיצה עם העכבר. אם זה אולם ללא כיסאות מסומנים יהיה ניתן לבחור כרטיס לאזור מסוים. תמונות להמחשה מאתרי כרטיסים קיימים:

עם זאת, בשלב הזה אני לא יודע איך לעשות את זה. אני מניח שיש צורך ביצירת תמונה של מפת האולם בתוכנה נפרדת ואז באמצעות html ו-css ליצור דף עבור כל ShowRun שבו יש את התמונה ועל גבי המושבים יהיו כפתורים המייצגים את המושבים (והפונקציונליות בהתאם היא לחיצה על כפתור משריינת מושב על ידי הוספה לרשימה אשר תופיע בהזמנה; לחיצה שניה על אותו מושב תבטל את השיריון על ידי הוצאה מהרשימה).

בעקבות הידע המועט שיש לי בכתיבת עיצוב אינטרנטי, אני מניח שיצירת האפשרות הזו של בחירת מושבים ממפת אולם תהיה מורכבת עבורי ותדרוש זמן רב, אשר בנקודת הזמן הזו – לא בהכרח יהיה לי. ייתכן ואזנח את האפשרות הזו לטובת האלטרנטיבה שאני רואה כקלה יותר: הצגת רשימה של מושבים זמינים ובחירת מושבים מתוך רשימה זו, או מערכת אשר מציעה את המושבים "הטובים ביותר" (קרוב יותר=טוב יותר) ללקוח.

**5. קוד הJAVA**

-מרבית הפונקציליות של האפליקציה תתאפשר באמצעות JPA, ניתן לראות בשלב זה שלמעשה האפליקציה בעיקר משתמשת בשאילתות כדי לאחסן ולאחזר נתונים מבסיס הנתונים

-האתגר המרכזי מבחינתי יהיה סנכרון בין כמה משתמשים ושרת אחד, באמצעות JSF.