

Movies For

You

סדנה בתכנות מונחה עצמים - 20586

מנחה : דני כלפון
סטודנטית: עדי כהן
תז: 315241125

תוכן עניינים

| | |
|----|--------------------------|
| 2 | תוכן עניינים |
| 3 | מסמך אפיון וניתוח |
| 3 | ייעוד המערכת |
| 3 | דרישות המערכת |
| 4 | מרכיבי המערכת |
| 4 | משתמשי המערכת |
| 5 | דיאגרמת תרחישים |
| 5 | ניהול משתמשים |
| 6 | דיאגרמת פעילות |
| 6 | הוספת ביקורת |
| 7 | מסמך תכנון ועיצוב |
| 7 | תיאור כללי של המערכת |
| 8 | שכבת הבסיס נתונים |
| 15 | שכבת המודל Model |
| 17 | שכבת הלוגיקה |
| 17 | מחלקה כללית |
| 17 | מחלקות הView Model |
| 26 | מחלקות הConverters |
| 27 | מחלקות הCommand |
| 29 | שכבת התצוגה |
| 32 | דיאגרמת מסד נתונים |
| 33 | דיאגרמת מחלקות |
| 37 | מדריך למשתמש |
| 37 | תיאור המערכת |
| 37 | אורח |
| 41 | משתמש רשום |
| 46 | מנהל |

מסמך אפיון וניתוח

ייעוד המערכת

MoviesForYou נועדה לשמש כממשק מחבר בין מבקשי תוכניות בידור (משתמש) ומנהל האפליקציה שמספקת שירות זה. ציבור נרחב מבקש בידור כדוגמאת סרטים, מבלי צורך להתלבש ולצאת, אלא מנוחות הבית והמחשב. לעיתים הדרישה לכך נובעת ממצב רפואי או כללי (כדוגמאת הקורונה).

המשתמש מעונין באפליקציה נוחה להפעלה שתתן לו את הסרט בו מעונין בנישה ספציפית ואף להיות שותף בכך שיוסיף ביקורת וחוות דעת. עיקרו של קו זה הינו ליצור הזדהות וקשר אישי למערכת זו. מנהל המערכת אחראי על תפקוד יציב וניהול התנהלות המשתמש ותקשורת בינו למשתמש על ידי מערכת המיילים.

לשם כך יש צורך באפליקציה המשרתת צורך זה בצורה קלה למשתמש כשגם למנהל השירות ניתן בקלות לנהל ולתאם את הביקוש וההיצע, תוך שמירת היסטוריה ופרטי המשתמש הרשום, באופן מרוכז.

דרישות המערכת

דרישות פונקציונליות

- המערכת תספק יכולות לניהול משתמשי המערכת
 - הוספה/ עדכון של משתמש
 - צפייה בפרטי משתמש
 - צפייה בסרטים שהלקוח השכיר / משכיר
- המערכת תספק יכולות לניהול הסרטים
 - הוספה/ הסרה/ עדכון של סרט
 - חיפוש סרטים
 - מיון סרטים לפי זאנרים / גיל
- המערכת תספק יכולות להפקת דוחות
 - הפקת דוח דירוג הסרטים הנשכרים ביותר
 - הפקת דוח דירוג הזאנרים הנשכרים ביותר
 - הפקת דוח דירוג הסרטים לפי דירוג המשתמשים
 - הפקת דוח דירוג הסרטים לפי מספר הביקורות
- המערכת תספק למשתמש יכולות לניהול חשבון
 - הרשמה / התחברות למערכת
 - עדכון הפרטים האישיים של המשתמש
 - השכרה / החזרה של סרט
 - השארת ביקורת על סרט שהמשתמש שכר
 - צפייה בביקורות של סרטים שמשתמשים אחרים כתבו
- המערכת תספק יכולות לניהול ביקורות
 - הוספה/ סירוב / מחיקה של ביקורת
 - שליחת מייל ללקוח עם עדכון על בחירת המנהל

דרישות כלליות

- MoviesForYou תהיה זמינה לשימוש באמצעות מחשב
- MoviesForYou תהיה נוחה לשימוש ואינטראקטיבית
- כל פעולה תתעדכן לשאר משתמשי המערכת.

מרכבי המערכת

בסיס נתונים

בסיס הנתונים ימוקם בשרת אינטרנט שבו ישמר כל המידע של המערכת – משתמשים, סרטים, ביקורות ועוד. מבסיס הנתונים יתבצעו שליפות המידע שיוצג במערכת. רק למנהל המערכת יש גישה ישירה לבסיס הנתונים (שם משתמש וסיסמא) והוא אחרי על מבנה המידע שלו.

אפליקציה

האפליקציה היא אפליקצית מחשב והיא מאפשרת להשתמש במערכת בצורה פשוטה ונוחה. ישנה אפליקציה אחת המשמשת את דורש השירות ומנהל המערכת. המערכת משתמשת בבסיס נתונים משותף עבור המשתמש והמנהל (system administrator) עם זאת, הרשאות הגישה וההפעלה שונות.

משתמשי המערכת:

אורח

אורח הוא משתמש שאינו רשום / מחובר למערכת

אורח יכול

- להתחבר / להירשם
- לחפש סרטים במערכת
- צפייה בפרטי סרטים
- צפייה בביקורות על סרטים

משתמש רגיל

משתמש יכול לבצע את הפעולות הבאות

- חיפוש סרטים במערכת
- שכירת / החזרת סרטים
- הוספת ביקורת על סרט שהוא שכן
- צפייה בפרטי סרטים
- צפייה בביקורות על סרטים
- מעבר על הסטוריית הצפייה שלו

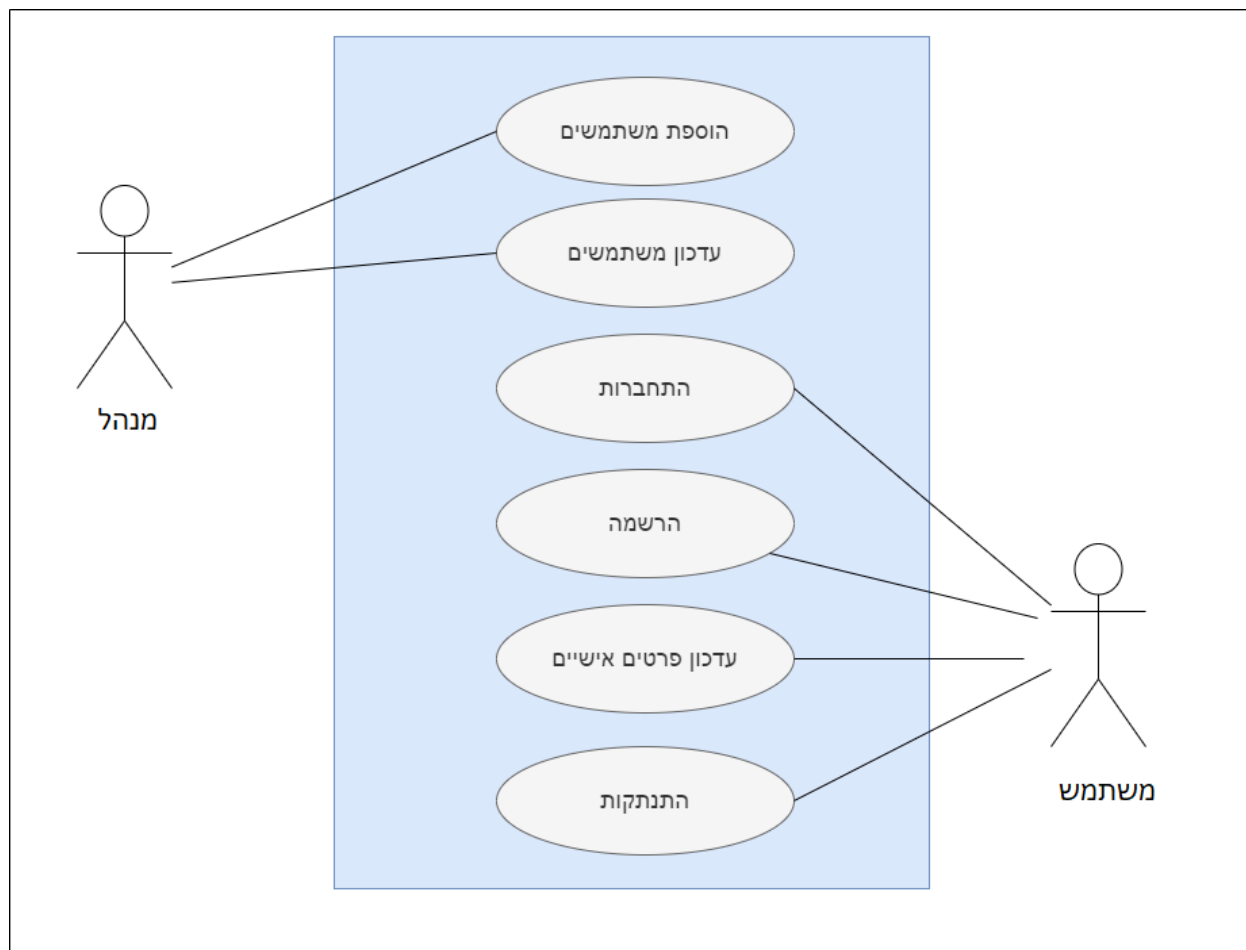
מנהל

מנהל המערכת הוא משתמש המוגדר מראש, הוא היחיד שמסוגל לבצע את הפעולות הבאות:

- הוספת/ עדכון משתמשים במערכת
- הוספת / עדכון / מחיקת סרט מן המערכת
- אישור / מחיקת ביקורות על סרטים
- השכרה / החזרת סרטים עבור משתמשים אחרים
- הפקת דוחות

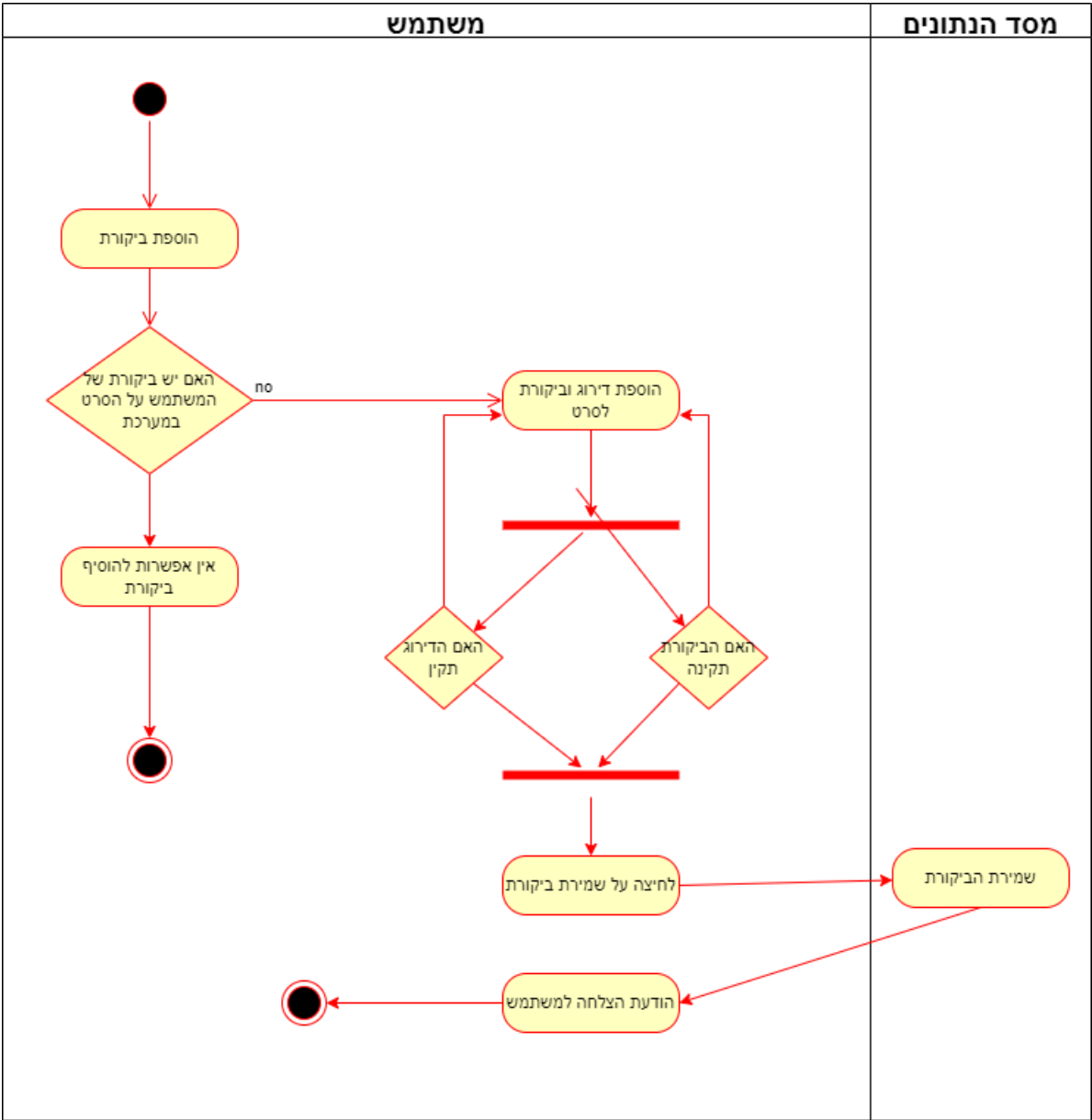
דיאגרמת Use-Case

משתמשים במערכת



דיאגרמת Activity

הוספת ביקורת לסרט



מסמך תכנון ועיצוב

תיאור כללי של המערכת

המערכת נכתבה בשפת c# WPF (Windows Presentation Foundation) שהיא UI פריימוורק שאיתה אפשר ליצור אפליקציות למחשב. הלוגיקה נכתבת ב-C# והתצוגה (GUI) נכתב ב-WPF xaml. תומכת במגוון רחב של תכונות לפיתוח אפליקציות מה שנותן את האפשרות ליצור אפליקציות בצורה דינאמית וגמישה.

המערכת נכתבה לפי דפוס העיצוב (Model, View, ViewModel) MVVM שמתאים עבור WPF, WPF בנוי כך שיהיה קל מאד לבנות אפליקציות עם MVVM. עקרונות התבנית הם המבנה:

- Model - הגדרה של הלוגיקה של המחלקות של האובייקטים
- View – החלק האחראי כל התצוגה (הגרפיקה והUI), היא מקבלת את האיטרקציה של המשתמש עם התצוגה (לחיצה עם עכבר, קלטים וכו') ומעבירה את הטיפול לView Model באמצעות קשירת הנתונים (data binding) שמוגדרים מראש שמקשרים בין הview לview model. View מחזיק מופע של view model על מנת שיוכל לתקשר איתו
- View Model – מקשר בין הview לmodel ובין הview לקוד. הוא מכיל את הלוגיקה של העמודים (view) ומכיל בדרך"כ מופעים של הmodel

הרעיון הוא הפרדה מוחלטת בין ה View-לבין ה Model-כשהתקשורת ביניהם היא באמצעות מנגנוני Commands להעברת פקודות\מסרים, ו Data Binding-בין ה View-ל ViewModel-כדי להציג נתונים.

המערכת מכילה פרוייקט אחד, מסד הנתונים שנבחר הוא Azure Sql Database.

בסיס הנתונים

בסיס הנתונים שנבחר הינו SQL Database

חשוב שבסיס הנתונים יהיה בענן ונגיש מכל מקום, SQL Database נתמך ע"י ספקי ענן גדולים (Azure, Oracle, AWS, Google cloud) ומאפשר מעבר קל בניהם. אני משתמש בבסיס נתונים ב Azure שנותן גישה נוחה לבסיס הנתונים דרך Visual Studio. קל מאד להתחבר לשרת של Azure דרך הקוד ולבצע איתו פעולות. בנוסף Azure נותן מגוון דרכים להגדיר את הבסיס נתונים שלי, כולל סיסמא, כמות חיבורים אפשרית, כמות המקום ועוד.

יתרון נוסף הוא שאפשר לפתוח את הבסיס נתונים שנמצא ב Azure ב Visual Studio Server ובכך לעדכן ולעבוד עם בסיס הנתונים ב Visual Studio עצמו במקום ללכת לאפליקציה חיצונית כל פעם.

המחלקה המתקשרת עם בסיס הנתונים היא DBOperations. מחלקה זו היא מחלקה סטטית שמכילה פונקציות סטטיות. מחלקה זו אחראית ליצור את ההתקשרות מול מסד הנתונים באמצעות אובייקט מסוג SqlConnection כאשר ה connection string מוגדר בקובץ. config.App בנוסף, מחלקה זו אחראית לשלוח נתונים באמצעות שאילתות מסוג Select ולעדכן נתונים באמצעות שאילתות מהסוגים Insert, Delete, Updated. או באמצעות Stored Procedures שמעודכנים בבסיס הנתונים.

Stored Procedures הן שיטות כתיבה של שאילתות שמאוכסנות בשרת של בסיס הנתונים (ב azure הן מאוכסנות בתקיית Store Procedures, באותה תקייה שבה הטבלאות מאוכסנות). הפרוצדורה כוללת קוד של התחלה, פרמטרים, גוף (שבו נכתבות השאילתות של הפרוצדורה) וקוד סיום. משתמשים בפרוצדורה בדרך"כ כאשר רוצים לכתוב שאילתא מסובכת או כמה שאילתות ביחד בצורה זו מפחיתים את תעבורת המידע בין השרת לאפליקציה (אם עושים כמה שאילתות ביחד לא צריך להחזיר לשלוח שאילתא ולקבל מידע על כל אחת מהן) ואפשר לבדוק נכונות נתונים, לטפל בשגיאות בקלות יותר, לבצע מניפולציה על הנתונים ולטעון אותם.

בנוסף בבסיס הנתונים יש טריגרים, טריגרים (Triggers) הם פרוצדורות שנקראות ע"י השרת בכל פעם שמתבצע Update, Insert או Delete על טבלה מסויימת, לפי הגדרת הטריגר. בעזרת הטריגר אפשר לתחזק טבלאות ביותר קלות ולמנוע קוד כפול

טבלאות בבסיס נתונים

טבלת זאנרים (Genres)

טבלה זו מכילה מידע אודות הזאנרים הקיימים במערכת. הטבלה מכילה את השדות הבאים:

| שדה | טיפוס | הסבר |
|----------------|----------|----------------------|
| <u>GenreId</u> | מספר שלם | מזהה זאנר, מפתח ראשי |
| GenreName | מחרוזת | שם הזאנר |

טבלת סרטים (Movies)

טבלה זו מכילה מידע אודות הסרטים הקיימים במערכת. הטבלה מכילה את השדות הבאים:

| שדה | טיפוס | הסבר |
|----------------|-------------------------|---------------------|
| <u>MovieId</u> | מספר שלם | מזהה סרט, מפתח ראשי |
| Title | מחרוזת (50) | כותרת הסרט, יחודי |
| Plot | מחרוזת (max) | עלילת הסרט |
| Actors | מחרוזת (MAX) | שחקנים בסרט |
| ReleaseYear | מספר שלם | שנת היציאה של הסרט |
| Image | מערך בינארי (VARBINARY) | תמונה של הסרט |
| AgeRating | מספר שלם | הגבלת גיל |

טבלת משתמשים (Users)

טבלה זו מכילה מידע אודות המשתמשים הקיימים במערכת. הטבלה מכילה את השדות הבאים:

| שדה | ערך | הסבר |
|-----------------|-------------|---------------------|
| <u>Username</u> | מחרוזת (10) | שם משתמש, מפתח ראשי |
| Firstname | מחרוזת (20) | שם פרטי |
| Lastname | מחרוזת (20) | שם משפחה |
| Password | מחרוזת (20) | סיסמא |
| Email | מחרוזת (50) | אימייל |
| DateOfBirth | תאריך | תאריך לידה |
| IsAdmin | בוליאני | האם המשתמש הוא מנהל |

טבלת זאנרים של סרטים (MoviesGenre)

טבלה זו מכילה מידע אודות זאנר של סרטים הקיימים במערכת. הטבלה מקשרת בין הטבלאות סרטים לזאנרים בקשר מסוג רבים לרבים
הטבלה מכילה את השדות הבאים:

| שדה | ערך | הסבר |
|----------------|----------|-------------------------------|
| <u>MovieId</u> | מספר שלם | מזהה סרט, מפתח ראשי, מפתח זר |
| <u>GenreId</u> | מספר שלם | מזהה זאנר, מפתח ראשי, מפתח זר |

טבלת סרטים מושכרים (RentedMovies)

טבלה זו מכילה מידע אודות הסרטים המושכרים הקיימים במערכת. טבלה זו מקשרת בין הטבלאות "סרטים" ל"משתמשים" בקשר רבים לרבים. הטבלה מכילה את השדות הבאים:

| שדה | ערך | הסבר |
|-------------------|-------------|-------------------------------|
| <u>MovieId</u> | מספר שלם | מזהה סרט , מפתח ראשי, מפתח זר |
| <u>Username</u> | מחרוזת (10) | שם משתמש, מפתח ראשי, מפתח זר |
| <u>DateRented</u> | תאריך | תאריך השכרת הסרט, מפתח ראשי |
| DateReturned | תאריך | תאריך החזרת הסרט |
| CanReview | בוליאני | האם אפשר לדרג את הסרט |

טבלת ביקורות (Reviews)

טבלה זו מכילה מידע אודות הביקורות שקיימות במערכת. טבלה זו מקשרת בין הטבלאות "סרטים" ל"משתמשים" בקשר רבים לרבים. הטבלה מכילה את השדות הבאים:

| שדה | ערך | הסבר |
|-----------------|--------------|-------------------------------|
| <u>MovieId</u> | מספר שלם | מזהה סרט , מפתח ראשי, מפתח זר |
| <u>Username</u> | מחרוזת (10) | שם משתמש, מפתח ראשי, מפתח זר |
| MovieTitle | מחרוזת (50) | כותרת הסרט, מפתח זר |
| Date | תאריך | תאריך הביקורת |
| Rating | מספר שלם | דירוג הסרט |
| Review | מחרוזת (MAX) | ביקורת על הסרט |
| IsApprove | בוליאני | האם המנהל אישר כבר את הביקורת |

Store Procedures

AddOneGenre

פרוצדורה שמקבלת מזהה של זאנר אחד ומזהה סרט, והופכת את הזאנר הזה לזאנר היחיד של הסרט

AddTwoGenres

פרוצדורה שמקבלת מזהה של שני זאנרים ומזהה סרט, והופכת את הזאנרים לזאנר היחיד של הסרט

DeleteMovie

פרוצדורה שמקבלת מזהה סרט ומוחקת אותו מ3 טבלאות – טבלת הסרטים, טבלת הסרטים המושכרים וטבלת הזאנרים של סרטים

GetAllMovies

הפרוצדורה מחזירה את כל הסרטים שקיימים במערכת בנוסף לדירוג הממוצע שלהם ומספר הביקורות שלהם (מטבלת הביקורות)

GetAllRentedMovies

הפרוצדורה מחזירה את כל הסרטים שהושכרו והוחזרו

GetAllUserRentedMovies

הפרוצדורה מקבלת שם משתמש ומחזירה את כל הסרטים שהוא השכיר והחזיר

GetCurrRentedMovies

הפרוצדורה מחזירה את כל הסרטים שהושכרו ועדיין לא הוחזרו

GetCurrUserRentedMovies

הפרוצדורה מקבלת שם משתמש ומחזירה את כל הסרטים שהמשתמש השכיר ועדיין לא החזיר

GetMostRentedGenres

הפרוצדורה מחזירה את רשימת הזאנרים שמדורגת לפי כמה פעמים השכירו אותם

GetMostRentedMovies

הפרוצדורה מחזירה את רשימת הסרטים שמדורגת לפי כמה פעמים השכירו אותם

המחלקה DBOperations

מכילה מתודות בוליאניות ומתודות אסיכרוניות לעבודה מול בסיס נתונים . המחלקה נמצאת בתיקייה service

```
Public static bool UpdateMovie(MovieModel movie, int fGenre, int sGenre)
```

מתודה זו מעדכנת סרט ואת הזאנרים שלו ומחזירה האם הצליחה לעדכן בהצלחה

```
private static SqlCommand ChangeGenres(int fGenre, int sGenre, int movieId, SqlConnection connection)
```

מתודה זו בודקת האם צריך לעדכן את הזאנרים (את שניהם, אחד מהם או אף אחד) ומחזירה SqlCommand עם הפקודה

```
public static async Task<List<MoviesGenreModel>> GetAllMoviesGenresData()
```

מתודה זו מחזירה רשימה של כל הזאנרים של הסרטים

```
public static bool AddMovie(MovieModel movie, int fGenre, int sGenre)
```

מתודה זו מוסיפה סרט ואת הזאנרים שלו לבסיס נתונים ומחזירה האם הצליחה לעדכן בהצלחה

```
public static bool DeleteMovie(int movieId)
```

מתודה זו מוחקת את הסרט עם מזהה הסרט שנשלח לה (ואת הזאנרים שלו) ומחזירה האם הצליחה למחוק בהצלחה

```
public static static async Task<List<GenreModel>> GetAllGenresData()
```

מתודה זו מחזירה רשימה של כל הזאנרים

```
public static async Task<List<MovieModel>> GetAllMoviesData()
```

מתודה זו מחזירה רשימה של כל הסרטים

```
public static bool RentMovie(int movieId, string username)
```

מתודה זו משכירה סרט למשתמש ומחזירה האם הצליחה להשכיר

```
public static async Task<List<RentedMovieModel>> GetAllCurrRentedMoviesData(bool isAdmin, string username)
```

המתודה מחזירה רשימה של כל הסרטים שמושכרים כרגע

```
public static async Task<List<RentedMovieModel>> GetAllRentedMoviesData(bool isAdmin, string username)
```

המתודה מחזירה רשימה של כל הסרטים שהושכרו והוחזרו

```
public static bool ReturnMovie(RentedMovieModel rentedMovieData)
```

המתודה מחזירה את הסרט המושכר ומחזירה האם הצליחה להחזירו

```
public static async Task<List<Tuple<int, string, int>>> GetMostRentedMovies()
```

המתודה מחזירה רשימה של הסרטים לפי כמות ההשכרות שלהם

```
public static async Task<List<Tuple<int, string, int>>> GetMostRentedGenres()
```

המתודה מחזירה רשימה של הזאנרים לפי כמות ההשכרות שלהם

```
Public static async Task<List<UserModel>> GetAllUsersData()
```

המתודה מחזירה רשימה של כל המשתמשים

```
Public static bool AddUser(UserModel newUser)
```

המתודה מוסיפה משתמש חדש למערכת ומחזירה האם הצליחה להוסיפו

```
public static bool UpdateUserInfo(UserModel user, string oldUsername)
```

המתודה מעדכנת מידע של המשתמש ומחזירה האם עדכנה בהצלחה

```
public static async Task<bool> CheckIfUsernameExist(string username)
```

המתודה בודקת האם שם המשתמש קיים במערכת

```
public static async Task<bool> CheckIfEmailExist(string email)
```

המתודה בודקת האם האימייל קיים במערכת

```
public static async Task<UserModel> LoginUser(string username, string password)
```

המתודה בודקת האם המשתמש קיים במערכת ומחזירה אותו במידה וכן

```
public static bool AddReview(ReviewModel review)
```

המתודה מוסיפה ביקורת למערכת ומחזירה האם הוסיפה בהצלחה

```
public static async Task<List<ReviewModel>> GetAllReviewsData(bool isPendingApproval)
```

המתודה מחזירה רשימה של כל הביקורות שעדיין לא אושרו / נדחו

```
public static bool ApproveReview(ReviewModel review)
```

המתודה מאשרת ביקורת ומחזירה האם אישרה בהצלחה

```
public static bool DeleteReview(ReviewModel review)
```

המתודה מוחקת ביקורת (מחיקה / דוחה) ומחזירה האם מחקה בהצלחה

```
public static int GetAllGenresDataCount()
```

המתודה מחזירה את כמות הזאנרים במערכת

```
public static int GetAllCurrRentedMoviesDataCount()
```

המתודה מחזירה את כמות הסרטים שמושכרים כרגע במערכת

```
public static int GetAllRentedMoviesDataCount()
```

המתודה מחזירה את כמות הסרטים שהושכרו והוחזרו במערכת

```
public static int GetAllReviewsDataCount()
```

המתודה מחזירה את כמות הביקורות המאושרות שבמערכת

```
public static int GetAllUsersDataCount()
```

המתודה מחזירה את כמות המשתמשים במערכת

```
private static void CatchSqlConnectionExceptions(SqlException sqlException)
```

המתודה מקבלת שגיאת sqlמציגה אותה למשתמש ושואלת איך להמשיך

שכבת הModel

לכל טבלה יש מודל שמכיל מבנה דומה למבנה הטבלה. בחלק מן המודלים יש תכונות נוספות שכוללות מידע נוסף שאינו חלק מן הטבלה המקורית אך הוא חלק ממידע רלוונטי של רשומות בטבלה שמקושר לטבלה אחרת. כך, אם מראש שולפים את המידע אפשר לצמצם קריאות לבסיס הנתונים, ולהגביר את יעילות ומהירות הקוד.

המחלקות

MovieModel.cs

המחלקה מייצגת סרט במערכת, היא בעלת מבנה דומה לטבלת הסרטים אך נוסף אליהם יש שני תכונות נוספות של דירוג הסרט וכמות הביקורות שלו. שני אלה הם תכונות רלוונטיות של סרט אך הן מגיעות מטבלת הביקורות, ומחושבות בשליפה של הסרטים מבסיס הנתונים. (בפרוצדורה בשרת Databasesen) היא בעלת התכונות הבאות:

```
public int MovieId { get; set; }
public string Title { get; set; }
public string Plot { get; set; }
public string Actors { get; set; }
public int ReleaseYear { get; set; }
public byte[] Image { get; set; }
public int AgeRating { get; set; }
public int Rating { get; set; }
public int ReviewCount { get; set; }
```

GenreModel.cs

המחלקה מייצגת זאנר במערכת. היא בעלת התכונות הבאות:

```
public int GenreId { get; set; }
public string GenreName { get; set; }
```

MoviedGenreModel.cs

המחלקה מייצגת זאנר של סרט במערכת. היא בעלת התכונות הבאות :

```
public int MovieId { get; set; }
public int GenreId { get; set; }
```

RentedMovieModel.cs

המחלקה מייצגת סרט שהושכר במערכת. המחלקה בעלת מבנה דומה לטבלת הסרטים שמושכרים אך נוסף לתכונות של הטבלה יש שלוש שתוכנות נוספות של שם פרטי ומשפחה (של המשתמש ששכר את הסרט) ושם הסרט. התכונות של המשתמש מגיעות מטבלת המשתמשים, ושם הסרט מגיע מטבלת הסרטים. התכונות הנוספות נוספות בשליפה של הסרטים המושכרים מבסיס הנתונים. (בפרוצדורה בשרת Databasesen) המחלקה בעלת התכונות הבאות:

```
public int MovieId { get; set; }
public string Username { get; set; }
public string Firstname { get; set; }
public string Lastname { get; set; }
public string Title { get; set; }
```

```
public DateTime RentDate { get; set; }  
public DateTime ReturnDate { get; set; }  
public bool CanReview { get; set; }
```

ReviewModel.cs

המחלקה מייצגת ביקורת של סרט במערכת. היא בעלת התכונות הבאות

```
public int MovieId { get; set; }  
public string MovieTitle { get; set; }  
public string Username { get; set; }  
public DateTime Date { get; set; }  
public int Rating { get; set; }  
public string Review { get; set; }  
public bool IsApprove { get; set; }
```

UserModel.cs

המחלקה מייצגת משתמש במערכת. היא בעלת התכונות הבאות

```
public string Username { get; set; }  
public string Firstname { get; set; }  
public string LastName { get; set; }  
public string Password { get; set; }  
public string Email { get; set; }  
public DateTime DateOfBirth { get; set; }  
public bool IsAdmin { get; set; }
```


שכבת הלוגיקה –

מחלקה כללית – GlobalValidator.cs

המחלקה מכילה אוסף של פונקציות סטטיות שמבצעות בדיקות שהנתונים תקינים עבור תכונות שונות במערכת, לפני שיש תקשורת עם הבסיס נתונים

```
public static bool IsReviewValid(ReviewModel review)
public static bool IsNewUserValid(UserModel newUser)
public static bool IsUpdatedUserValid(UserModel newUserDetails, string
oldUsername, string oldEmail)
public static bool IsLoginValid(string username, string password)
public static bool IsMovieValid(MovieModel movie)
private static bool IsValidNameOrLastName(string a_name)
private static bool IsUsernameValid(string a_username)
private static bool IsPasswordValid(string a_password)
private static bool IsEmailValid(string email)
private static bool IsDateofbirth(DateTime date)
```

מחלקות הView Model

ViewModelBase.cs

כל ViewModel יורש מן המחלקה הזו. במחלקה הזו נמצאות מתודות שצריך שיהיה לכל ViewModel. כרגע המחלקה מכילה רק מימוש לINotifyPropertyChanged.

```
public event PropertyChangedEventHandler PropertyChanged;
protected void OnPropertyChanged(string propertyName)
```

MainWindowViewModel.cs

מחלקת הVM של MainWindow.xaml שהוא המסך הראשי של המערכת. המחלקה מכילה את התוכנות והמתודות:

```
private ViewModelBase _selectedViewModel;
public ViewModelBase SelectedViewModel
    התכונה מחזיקה את הVM של View שכרגע מוצג על המסך.

public void onCurrentUserChange()
public Users CurrUser
    התכונה מחזיקה את סוג המשתמש, והמתודה מעדכנת את התפריט שהמשתמש משתנה

public ICommand UpdateCurrViewCommand { get; set; }

// ctor
public MainWindowViewModel()

private ICommand _logoutCommand;
public ICommand LogoutCommand

private ICommand viewProfileCommand;
```

```
public ICommand ViewProfileCommand
```

```
private ICommand signupCommand;  
public ICommand SignupCommand
```

```
private ICommand loginCommand;  
public ICommand LoginCommand
```

```
private void Logout()  
public void ViewProfile()  
public void Signup()  
private void Login()
```

הפקודה מפעילה מתודה התנתקות מהמערכת / צפייה בפרופיל / הרשמה / התחברות. צפייה בפרופיל אפשרית רק למשתמש רשום / מנהל. וכל השאר אפשריים רק לאורח

בנוסף למחלקת MainWindowViewModel, יש עוד שתי מחלקות ב MainWindowViewModel.

מחלקה סטטית CurrentUser

המחלקה מחזיקה את התכונות של סוג המשתמש ואת פרטי המשתמש המחובר. המחלקה סטטית ופומבית ולכן אפשר לגשת אליה בכל הפרוייקט. המחלקה הזו מתעדכנת ומעדכנת כאשר המשתמש משתנה / מתנתק שומרת את הפרטים שלו

```
private static Users userType = Users.Guest;  
public static Users UserType  
private static UserModel user;  
public static UserModel User
```

Users Enum

הEnum של Users מכיל את סוגי המשתמשים הקיימים

```
public enum Users  
{  
    Admin,  
    Member,  
    Guest  
}
```

EntanceViewModel.cs

מחלקת VM של EntanceView.xaml שהוא דף הכניסה למערכת. המחלקה מכילה שלוש "פקודות" (Command).

```
public EntranceViewModel()
```

פקודת עדכון View, שמחליפה לView אחר כשמפעילים אותה לפי הפרמטר שהיא מקבלת

```
public ICommand UpdateCurrViewCommand { get; set; }
```

פקודת הרשמה שמפעילה את פונקציית ההרשמה (Signup) שפותחת למשתמש טופס הרשמה למערכת

```
private ICommand signupCommand;  
public ICommand SignupCommand
```

```
public void Signup()
```

ופקודת התחברות, שמפעילה את פונקציית ההתחברות שפותחת למשתמש טופס התחברות למערכת

```
private ICommand loginCommand;  
public ICommand LoginCommand  
private void Login()
```

AddOrUpdateUserViewModel.cs

המחלקת הVM של AddOrUpdateUserViewModel.xaml שהוא חלון להוספת או עדכון משתמשים
המחלקה מכילה את התכונות:

```
Private UserModel currMember;  
public event EventHandler<UserModel> AddUser;  
private bool isAdd = false;
```

```
private string oldUsername;  
private string oldEmail;
```

```
public UserModel CurrMember  
public bool IsAdd  
public string SaveButtonText {get; set}
```

```
public bool CanChangeField  
public bool ChangeIsAdmin
```

```
//ctor
```

```
public AddOrUpdateUserViewModel()  
public AddOrUpdateUserViewModel(UserModel a_currMember)
```

ואת הפקודה שבודקת שכל המשתמש החדש / המעודכן תקין ואז שומרת השינויים ומעדכנת / מוסיפה את
המשתמש למערכת לפי הדרישה

```
private ICommand saveCommand;  
public ICommand SaveCommand  
private void SaveUser(object window)
```

LoginViewModel.cs

מחלקת הVM של LoginView.xaml שהוא דף ההתחברות למערכת.
המחלקה מכילה:

```
public LoginViewModel()
```

תכונות של שם משתמש וסיסמא

```
private string username;  
public string Username {get; set;}
```

```
private string password;  
public string Password
```

פקודת התחברות למערכת, ומתודה התחברות למערכת שבודקת האם שם המשתמש וסיסמא תקינים והאם קיים משתמש עם סיסמא כזו במערכת. במידה וכן היא מחברת אותו למערכת.

```
private ICommand _loginCommand;  
public ICommand LoginCommand {get; set;}
```

```
private void Login(object window)
```

MoviesViewModel.cs

מחלקת הVM של MoviesViewModel.xaml שהוא דף שמציג את כל הסרטים שיש במערכת.

המחלקה כוללת את התכונות הבאות:

```
private List<MovieModel> allMoviesData;  
private List<MoviesGenreModel> allMoviesGenresData;  
private DBOperations database;  
private string filter;  
private int currGenreModel;  
private bool ageFilter;
```

```
// ctor
```

```
public MoviesViewModel()
```

```
public ObservableCollection<MovieModel> MovieList { get; set; }
```

```
public Dictionary<int, string> GenreList { get; set; }
```

רשימות הסרטים והזאנרים שיש במערכת

```
public bool AgeFilter
```

```
public int CurrGenre
```

```
public string Filter
```

```
private void Filters()
```

כאשר אחד מהפילטרים משתנה, מתודה Filters() מופעלת והיא טוענת את רשימת הסרטים מחדש עם הסרטים הרלוונטים לפילטרים שהוכנסו (גיל / זאנר / שם הסרט).

```
public bool IsGuest { get => CurrentUser.UserType == Users.Guest; }
```

```
private List<MovieModel> GetMoviesIDForGenre(int genreId)
```

מתודה שמקבלת מזהה זאנר ומחזירה את כל הסרטים ששייכים לזאנר במערכת

```
private ICommand _selectMovieCommand;
```

```
public ICommand SelectMovieCommand
```

```
private void SelectMovie(object o)
```

פקודת בחירת סרט שמקבלת סרט (שמתקבל כobject) כאשר היא מופעלת עוברים לSelectedMovieView שמציג את הפרטי הסרט.

```
private ICommand addMovieCommand;
```

```
public ICommand AddMovieCommand
```

```
private void AddMovie()
```

פקודת הוספת סרט. כאשר היא מופעלת היא פותחת חלון שבו אפשר להוסיף סרט. המתודה מעדכנת כשהסרט נוסף.

```
private void AddMovieEventHandler(object sender, MovieModel movie)
    כאשר Event חיצוני מופעל, המתודה מוסיפה את הסרט לרשימת הסרטים
```

AddOrUpdateMovieViewModel.cs

מחלקת ה VM ל AddOrUpdateMovieViewModel.xaml שהוא חלון להוספת או עדכון סרטים.
המחלקה מכילה את התכונות:

```
private MovieModel selectedMovie;
private List<GenreModel> allGenresData;
public event EventHandler<MovieModel> ReloadMoviesData;
private DBOperations databsae;
private int currGenre1;
private int currGenre2;
private bool isAdd = false;
```

```
public int CurrGenre1
public int CurrGenre2
public MovieModel SelectedMovie
public bool IsAdd { get => isAdd; }
```

```
public Dictionary<int, string> GenreList { get; set; }
```

רשימה של כל הזאנרים במערכת

```
// ctor
public AddOrUpdateMovieViewModel()
public AddOrUpdateMovieViewModel(MovieModel a_selectedMovie)
```

```
private void LoadGenres()
```

מתודה שטוענת את הזאנרים של הסרט הנבחר במידה ויש.

```
private ICommand addImageCommand;
public ICommand AddImageCommand
private void LoadImage()
```

פקודה שקוראת למתודה שטוענת תמונה מהמחשב של המנהל למערכת

```
private ICommand saveCommand;
public ICommand SaveCommand
private void SaveMovie(object window)
```

פקודה שקוראת למתודה שמוסיפה / מעדכנת את הסרט במערכת

SelectedMovieViewModel.cs

מחלקת ה VM של SelectedMovieViewModel.xaml שהוא מסך שמציג פרטים על סרט שנבחר ונותן אפשרות לבצע פעולות על הסרט.

המחלקה מכילה תכונות ומתודות:

```
public MovieModel selectedMovie;
private List<String> usersList;
private string selectedUsername;
private DBOperations database;
private bool canRentMovie;
private string genres;
```

```
public MovieModel SelectedMovie
public string Genres
```

```
public List<string> UsersList
public string SelectedUser
```

המנהל אינו משכיר בעצמו סרטים. לכן כאשר המנהל נמצא במערכת, המחלקה טוענת את רשימת כל המשתמשים ומקבלת את המשתמש שנבחר שהסרט יושכר אליו.

```
Public bool CanRentMovie
public Visibility MinAgeVis
```

בדיקה האם המשתמש יכול לשכור את הסרט (אם הוא אורח או שהגיל שלו נמוך מידי בשביל הסרט)

```
// ctor
```

```
public SelectedMovieViewModel(MovieModel a_selectedMovie)
```

```
private ICommand rentMovieCommand;
public ICommand RentMovieCommand
private void RentMovie()
```

פקודה שמפעילה את המתודה שבודקת האם אפשר להשכיר את הסרט למשתמש ואם כן אז היא משכירה לו את הסרט.

```
private ICommand updateMovieCommand;
public ICommand UpdateMovieCommand
private void UpdateMovie()
```

פקודה שמפעילה את המתודה שפותחת חלון שבו אפשר לעדכן את פרטי הסרט במערכת. אם הסרט עודכן בהצלחה הפונקציה מודיעה על כך למשתמש

```
private void UpdateMovieEventHandler(Object sender, MovieModel movie)
כאשר Event חיצוני מופעל, המתודה מעדכנת את פרטי הסרט במסך
```

```
private ICommand deleteMovieCommand;
public ICommand DeleteMovieCommand
private void DeleteMovie()
```

הפקודה מפעילה מתודה שמוחקת את הסרט

```
private ICommand reviewsCommand;
public ICommand ReviewsCommand
public bool IsRated
private void MovieReviews()
```

הפקודה מפעילה מתודה שפותחת חלון בו אפשר לראות את הביקורות על הסרט.

MovieReviewsViewModel.cs

מחלקת הVM של MovieReviewsView.xaml שהוא חלון שמכיל את כל הביקורות של סרט.
המחלקה מכילה את התכונות והמתודות:

```
// ctor
public MovieReviewsViewModel(int movieId)
public MovieReviewsViewModel()

public List<ReviewModel> AllMovieReviews
private List<ReviewModel> allMovieReviews;
```

רשימת הביקורות על סרט לפי מזהה הסרט

RentedMoviesViewModel.cs

מחלקת הVM של RentedMoviesView.xaml שהוא מסך שמראה את כל הסרטים שהושכרו
המחלקה מכילה את התכונות והמתודות:

```
// ctor
public RentedMoviesViewModel()

public ObservableCollection<RentedMovieModel> CurrRentedMoviesList { get;
set; }
public ObservableCollection<RentedMovieModel> RentedMoviesList { get; set; }
רשימת סרטים שכרגע מושכרים וסרטים שהושכרו והוחזרו

// return movie command
private ICommand returnCommand;
public ICommand ReturnCommand

// add review to a movie command
private ICommand reviewCommand;
public ICommand ReviewCommand

// Get all the returned rented movies
private void GetAllRentedMovies()

// return a movie, and update the view
private void ReturnMovie(object param)
המתודה מחזירה את הסרט בשביל המשתמש ומעדכנת את רשימות הסרטים המושכרים / שהושכרו
private void ReviewMovie(object param)
הפקודה מפעילה את המתודה, המתודת פותחת חלון שמאפשר למשתמש להוסיף ביקורת
```

AddReviewViewModel.cs

מחלקת הVM של AddReviewView.xaml שהוא חלון שמאפשר הוספת ביקורת לסרט שהושכר
המחלקה מכילה את התכונות והמתודות:

```
private RentedMovieModel currRentedMovie;
private string movieName;
private int movieRating;
```

```
private string reviewDesc;
```

```
// ctor
```

```
public AddReviewViewModel(RentedMovieModel rentedMovie)
```

```
public AddReviewViewModel()
```

```
public string MovieName
```

```
public int MovieRating
```

```
public string ReviewDesc
```

```
private ICommand saveReviewCommand;
```

```
public ICommand SaveReviewCommand
```

```
private void SaveReview(object param )
```

פקודה שקוראת למתודה שבודקת שהמשתמש בחר דירוג לסרט וכתב ביקורת, ואז מוסיף את הביקורת לבסיס נתונים, סוגר את החלון ומודיע שהביקורת מחכה לאישור מנהל

UsersViewModel.cs

מחלקת הVM של UsersView.xaml שהוא מסך שמציג למנהל את כל משתמשי המערכת. המחלקה מכילה את התכונות ומתודות:

```
public ObservableCollection<UserModel> MembersList { get; set; }
```

```
public List<UserModel> MembersList1 { get; set; }
```

```
// ctor
```

```
public UsersViewModel()
```

```
private ICommand addUserCommand;
```

```
public ICommand AddUserCommand
```

```
public void AddUser()
```

הפקודה קוראת למתודה שפותחת חלון שבו אפשר להוסיף משתמש חדש למערכת

```
private void AddUserEventHandler(object sender, UserModel user)
```

כאשר Event חיצוני מופעל, המתודה מוסיפה את המשתמש לרשימת המשתמשים MemberLists

```
private ICommand updateUserCommand;
```

```
public ICommand UpdateUserCommand
```

```
public void UpdateUser(object param)
```

הפקודה קוראת למתודה שפותחת חלון שבו אפשר לעדכן משתמש קיים

```
public void UpdateUsersList()
```

טעינה מחדש של רשימת המשתמשים מהdatabase

UsersReviewsViewModel.cs

מחלקת הVM של UsersReviewsViewModel.xaml שהוא מסך שמציג למנהל את כל הביקורות של המשתמשים

המחלקה כוללת את התכונות והמתודות:


```

// ctor
public UsersReviewsViewModel()

public ObservableCollection<ReviewModel> PendingApprovalReviews { get; set; }
public ObservableCollection<ReviewModel> ApproveReviews { get; set; }
    רשימת הביקורות שמחכות לאישור מנהל / מאושרות

private ICommand approveReviewCommand;
public ICommand ApproveReviewCommand

private ICommand declineReviewCommand;
public ICommand DeclineReviewCommand

private ICommand deleteReviewCommand;
public ICommand DeleteReviewCommand

private void ApproveReview(object param)
private void DeclineReview(object param)
private void DeleteReview(object param)
    פקודות שמפעילות את המתודות לאישור / דחייה / מחיקת ביקורת בהתאמה. במידה ודוחים / מוחקים ביקורת
    המנהל צריך להוסיף סיבה לבחירה שלו. לבסוף שולחים מייל למשתמש על הבחירה

private string GetReason()
    המתודה פותחת חלון (DeleteReviewReason.xaml) שמבקש את הסיבה למחיקת הביקורת. בלי סיבה אי
    אפשר יהיה להמשיך בתהליך

private void SendUpdatedReviewEmail(ReviewModel currReview, ReviewUpdate
reviewUpdate, string reason)
    המתודה שולחת מייל עדכון למשתמש. במייל מפורט אם הביקורת שלו אושרה / נדחתה / נמחקה ומדוע.

```

ReportsViewModel.cs

מחלקת ה VM של ReportsView.xaml שהוא מסך שמראה נתונים סטטיסטיים על המערכת ומאפשר ליצור דוחות. המחלקה מכילה את התוכנות והמתודות:

```

private ReportsEnum seletcedReport;
private List<MovieModel> allMoviesData;

public ReportsEnum SeletcedReport
public IEnumerable<ReportsEnum> ReportsEnumValues
    התכונה טוענת את רשימת הערכים ב ReportsEnum

// ctor
public ReportsViewModel()

public int UsersCount
public int MoviesCount
public int GenresCount

```

```
public int ReviewsCount
public int RentedMvoiesCount
public int CurrRentedMoviesCount
```

התכונות טוענות מבסיס הנתונים את הכמות המשתמשים / סרטים / זאנרים / ביקורות / סרטים שהושכרו והוחזרו / סרטים מושכרים

```
private ICommand createReportCommand;
public ICommand CreateReportCommand
private void CreateReport()
```

הפקודה מפעילה מתודה שבודקת איזה דוח נבחר, מפיקה אותה ושומרת אותו במחשב של המשתמש

ב- ReportsViewModel.cs מוגדר גם Enum, שמגדיר את סוגי הדוחות ואת השמות שלהם לתצוגה.

```
public enum ReportsEnum
{
    [Display(Name = "Select Report")]
    SelectReport,
    [Display(Name = "Most Rented Movies")]
    MostRentedMovies,
    [Display(Name = "Most Reviewed Movies")]
    MostReviewedMovies,
    [Display(Name = "Top Rated Movies")]
    TopRatedMovies,
    [Display(Name = "Most Rented Genres")]
    MostRentedGenres,
    [Display(Name = "Users List")]
    UsersList,
    [Display(Name = "Genres List")]
    GenresList
}
```

מחלקות ה Converters

כל מחלקות ה converters מממשות את IValueConverter ואת 2 הפונקציות Convert, ConvertBack בלבד.

AdminToVisibilityConverter.cs

ה Converter לא מקבל פרמטרים, הוא בודק האם המשתמש הוא מנהל. אם כן הוא מחזיר Visibility.Visible אחרת הוא מחזיר Visibility.Collapsed

BooleanToVisibilityConverter.cs

ה Converter מקבל ערך בוליאני. האם הערך הוא true אז הפונקציה מחזירה Visibility.Visible אחרת היא מחזירה Visibility.Collapsed

BooleanToCollapsedConverter.cs

ה Converter מקבל ערך בוליאני. האם הערך הוא false אז הפונקציה מחזירה Visibility.Collapsed אחרת היא מחזירה Visibility.Visible

EnumToDisplayNameConverter.cs

הקובץ מכיל שני מחלקות. מחלקת ה `con` ומחלקת בשם `EnumHelper` Converter מקבל ערך `enum`. הוא בודק האם הערך הוא `Enum` אם כן אז הוא מחזיר את השם של `Display attribute` של הערך של `Enum`, אחרת הוא מחזיר `null`. בשביל לקבל את שם `display attribute` ה Converter משתמש במחלקת של `EnumHelper` בפונקציה `GetAttributeOfType<T>`

RatingToBooleanConverter.cs

ה Converter מקבל שני מספרים שלמים. והוא מחזיר האם המספרים שווים. ה Converter הזה משמש כדי לסמן איזה דירוג המשתמש בחר בהוספת ביקורת.

StringFormatConverter.cs

ה Converter מקבל שני מחרוזות ומחזיר אותם בפורמט `(string b) string a`. ה Converter משמש בשביל להחזיר את המחרוזות בפורמט של שמות הסרטים + שנת היציאה שלהם.

UserToVisibilityConverter.cs

ה Converter מקבל פרמטר מסוג `Users` (`Enum`) והאם סוג המשתמש הנוכחי הוא אותו סוג כמו הפרמטר הוא מחזיר `Visibility.Visible` אחרת הוא מחזיר `Visibility.Collapsed`. ה Converter משמש לתפריט הראשי, לשינוי התפריט לפי הרשאות המשתמש

מחלקות Command

CommandHandler.cs

המחלקה מקבלת פעולה (עם או בלי פרמטרים) והאם אפשר לבצע את הפעולה. הפעולה שהיא מקבלת בפרוייקט היא פונקציה. משתמשים במחלקה זו בעיקר בפקודות, שכאשר הן מופעלות הן מפעילות מתודה שמקושרת אליהם ב `VM`.

```
private Action < object > _paramAction;  
private bool _canExecute;  
private Action _action;
```

שמירת הפעולה וערך בוליאני של האם אפשר לבצע אותה

```
// ctor
```

```
public CommandHandler(Action action, bool canExecute)  
public CommandHandler(Action < object > action, bool canExecute)
```

בנאים של המחלקה, הראשון הוא של פעולה ללא פרמטרים והשני עם.

```
public event EventHandler CanExecuteChanged;  
public bool CanExecute(object parameter)  
public void Execute(object parameter)
```

בדיקה האם אפשר לבצע את הפעולה, ומתודה לביצוע הפעולה.

UpdateViewCommand.cs

המחלקה מממשת את ICommand - interface , למחלקה יש תכונה של VM של המסך הראשי (MainWindowViewModel) והיא אחראית על להחליף את המסכים ואת VM שלהם במערכת. המחלקה עובדת בדפוס singleton. המחלקה מכילה תכונות ומתודות:

```
private static UpdateViewCommand _instance = null;
private static readonly object padlock = new object();
private MovieModel selectedMovie;

public static UpdateViewCommand GetUpdateViewInstance(MainWindowViewModel
viewModel)
public static UpdateViewCommand GetUpdateViewInstance()
    מימוש ה singleton, המתודות האלו קוראות לבנאי רק אם ה _instance אינו מוגדר. אם הוא כן המתודות
    מחזירות אותו.

private MainWindowViewModel viewModel;

// ctor
private UpdateViewCommand(MainWindowViewModel viewModel)

// Update that the user that use the app has changed
public void UpdateCurrentUser()
    המתודה מעדכנת על שינוי המשתמש המחובר למערכת

public event EventHandler CanExecuteChanged;

public bool CanExecute(object parameter)
public void SetSelectedMovie(MovieModel movieModel)

// change the user control that is being presented in the app and it's view model
public void Execute(object parameter)
    המתודה מקבלת פרמטר ולפיו משנה את ה VM ואת ה View של המסך.
```

שכבת התצוגה

שכבת התצוגה מגדירה ממשק משתמש גרפי (GUI) שמאפשר למשתמשים ליצור אינטרקציה ולהפעיל את המערכת.

שכבת התצוגה מכילה את קבצי View שכתובים בxaml. קבצי הxaml מגדירים איך המסך יראה, את האובייקטים שהמסך יכיל, המיקום שלהם ואת הקישור שלהם לקבצי ViewModel. התקשורת של קבצי התצוגה מתבצעת ע"י data binding. ב-MVVM לעבוד בעיקר עם data binding ובלי code behind כמה שאפשר, כך גם נהגתי בפרוייקט.

לכל קובץ xaml יש קובץ עם שם דומה שמסתיים בxaml.cs. והוא רק מאתחל את קובץ הxaml ומחבר אותו ל-VM ולכן לא אציג קבצים אלה פה.

מחלקות View

App.xaml – עמוד האפליקציה

העמוד שמגדיר את האפליקציה. בxaml זה אין עמודים של המערכת אבל הוא מכיל datatemplates שמגדירים views שמותאים לviewmodel שלהם, כך שכאשר נחליף מסכים בעזרת ה-VM בMainWindowViewModel נראה את המסך המתאים בMainWindow.

MainWindow.xaml

המסך הראשי באפליקציה. מסך זה תמיד מוצג והוא מציג בתוכו את המסכים (views) האחרים. החלון מכיל את התפריט הראשי שמציג למשתמש את האופציות להיכנס לדפים אחרים. התפריט משתנה בהתאם להרשאות המשתמש. הview מקושר לMainWindowViewModel לאורך מוצג בתפריט – Movies, Login, Signup למשתמש רשום מוצג בתפריט – Profile, Movies, RentedMovies, Logout למנהל מוצג בתפריט – Profile, Movies, RentedMovies, Members, Reviews, Reports, Logout

EntranceView.xaml

עמוד זה מוגדר כUserControl והוא לא חלון בפני עצמו. הוא העמוד הראשון שנפתח כשנכנסים לאפליקציה. הוא מציג את השם של המערכת ושלושה כפתורים. הכפתור הראשון הוא כפתור התחברות שהוא פותח חלון התחברות (LoginView) הכפתור השני הוא כפתור הרשמה שפותח חלון הרשמה (AddOrUpdateUserView) והכפתור השלישי נותן להמשיך כאורח ולראות את עמוד הסרטים (MoviesView)

LoginView.xaml

מוגדר כwindow, החלון נפתח מעל המערכת הראשית ובזמן שהחלון פתוח אין אפשרות לבצע אינטרקציה עם החלון הראשי. החלון הוא חלון התחברות שהוא אפשר להזין שם משתמש וסיסמא ולהתחבר למערכת. אם שם המשתמש והסיסמא לא עומדים בדרישות המערכת (שם משתמש בין 2 ל-10 תווים, וסיסמא מעל 8 תווים ועם לפחות אותה קטנה אחת, אות גדולה אחת ומספר) אז קופצת הודעת שגיאה. אם הם כן בודקים אם קיים משתמש כזה במערכת. אם קיים משתמש, המשתמש מתחבר למערכת, מקבלת הודעה על כך שהתחבר, החלון נסגר והוא מועבר לעמוד הסרטים (MoviesView.xaml). אם לא קיים משתמש כזה, קופצת הודעת שגיאה

AddOrUpdateUserView.xaml

מוגדר כwindow, החלון נפתח מעל המערכת הראשית ובזמן שהחלון פתוח אין אפשרות לבצע אינטרקציה עם החלון הראשי. החלון משמש את המערכת בכמה צורות – טופס הרשמה, פרופיל המשתמש, עדכון משתמש והוספת משתמש.

- **טופס הרשמה :** החלון משמש כטופס הרשמה למשתמש חדש. המשתמש יכול להוסיף שם משתמש, שם פרטי, שם משפחה, סיסמא, אימייל ותאריך לידה. כאשר המשתמש לוחץ על כפתור הרישום מתבצעת בדיקה של כל השדות ובמידה והם תקינים המשתמש נוסף למערכת והוא מקבל הודעה שכעת הוא יכול להירשם למערכת.
- **פרופיל:** החלון משמש כפרופיל של המשתמש. החלון מראה את הפרטים של המשתמש ונותן לו אפשרות לעדכן את כל הפרטים חוץ משם משתמש ותאריך לידה (מנהל שנכנס לפרופיל יכול לשנות את כל השדות). כשהמשתמש לוחץ על כפתור השמירה מתבצעת בדיקה של כל השדות ובמידה והם תקינים פרטי המשתמש מתעדכנים במערכת והוא מקבל הודעה על כך
- **עדכון משתמש:** החלון מתנהל כמו החלון של הפרופיל, ההבדל היחיד שהוא נפתח כאשר המנהל רוצה לעדכן פרטים של משתמשים אחרים ואז נוספת לו אפשרות להפוך את המשתמש למנהל.
- **הוספת משתמש:** החלון מתנהל כמו חלון הפרופיל ההבדל היחיד שהוא נפתח כאשר המנהל רוצה להוסיף משתמש ואז נוספת לו אפשרות להוסיף את המשתמש בתור מנהל.

MovieView.xaml

מוגדר כUserControl. המסך מציג את כל הסרטים בצורת דומה לגלריה עם תמונות, כאשר עומדים על תמונה רואים את שם הסרט ותאריך יציאת הסרט, אם לוחצים עליו עוברים למסך עם פרטי הסרט (SelectedMovieView). במסך יש ComboBox שאיתו אפשר לסנן את הסרטים לפי זאנרים ואפשר לחפש סרטים לפי השם שלהם (או חלק ממנו). אם המשתמש הוא לא אורח הוא יראה תחילה רק את הסרטים שמתאימים לגיל שלו ויהיה לו checkbox עם אפשרות לראות את כל הסרטים, גם אלו שהוא לא יכול להשכיר. אם המשתמש הוא מנהל יהיה לו גם כפתור להוסיף סרט (AddOrUpdateMovieView)

SelectedMovieView.xaml

מוגדר כUserControl. המסך יציג את פרטי הסרט ואת דירוג וביקורות הסרט במידה ויש. אפשר יהיה להשכיר סרט רק אם המשתמש רשום ובגיל המתאים ואז יוכל ללחוץ על כפתור השכרת סרט שיעביר אותו למסך הסרטים שמושכרים אם הסרט הושכר (RentedMoviesView). למנהל יראה comboBox עם שמות המשתמשים ויכול להשכיר סרט עבורם או ללחוץ על כפתור עדכון סרט שיפתח חלון לעדכון הסרט (AddOrUpdateMovieView) או ללחוץ על כפתור שימחק את הסרט ויעביר אותו למסך הסרטים (MovieView)

AddOrUpdateMovieView.xaml

מוגדר כwindow, החלון נפתח מעל המערכת הראשית ובזמן שהחלון פתוח אין אפשרות לבצע אינטרקציה עם החלון הראשי. בחלון זה מנהל יכול להוסיף סרט או לעדכן סרט. אם מתבצעת הוספת סרט השדות ריקים והמנהל צריך למלא את כולם ואם מתבצע עדכון השדות מציגים את פרטי הסרט ומנהל יכול לשנות אותם (המנהל לא יכול לעדכן את מזהה הסרט). לבסוף כשהמנהל לוחץ על כפתור השמירה, השדות נבדקים ומתעדכנים לבסיס הנתונים. אחרי הוספת סרט מסך הסרטים מתעדכן (MovieView) ואחרי עדכון הסרט מסך פרטי הסרט מתעדכן (SelectedMovieView)

MovieReviewsList.xaml

מוגדר כwindow, החלון נפתח מעל המערכת הראשית ובזמן שהחלון פתוח אין אפשרות לבצע אינטרקציה עם החלון הראשי. החלון מציג את כל הביקורות של סרט שנבחר. כל ביקורות בנויה מדירוג הסרט, הביקורות ומתחת אליהם המשתמש שכתב את הביקורות והתאריך

RentedMoviesView.xaml

מוגדר כ־userControls. המסך מציג את הסרטים המושכרים והסרטים שהושכרו והוחזרו. המסך זמין למשתמשים רשומים ומנהל. ישנם שתי טבלאות במסך. הטבלה של הסרטים המושכרים מכילה 5 עמודות שהן – שם הסרט המושכר, תאריך השכרה, כפתור צפייה בסרט, כפתור החזרת סרט וכפתור ביקורת. הטבלה של הסרטים שהושכרו מכילה 4 עמודות שהן שם הסרט המושכר, תאריך ההשכרה, תאריך ההחזרה וביקורת. הטבלה של המנהל מכילה עמודות אלו ובנוסף מכילה שם משתמש, שם פרטי ושם משפחה מכיוון שבעוד המשתמש רואה את כל הסרטים שלו, המנהל רואה את כל הסרטים המושכרים / מוחזרים של כל המשתמשים. אם כפתור החזר סרט נלחץ הסרט עובר מטבלת הסרטים המושכרים לטבלת הסרטים המוחזרים. כפתור הביקורת יהיה לחיצה רק אם אין ביקורת של משתמש זה במערכת. במידה ולוחצים של הוספת ביקורת נפתח חלון ביקורת (AddReviewView).

AddReviewView.xaml

מוגדר כ־window, החלון נפתח מעל המערכת הראשית ובזמן שהחלון פתוח אין אפשרות לבצע אינטרקציה עם החלון הראשי. החלון מציג את שם הסרט שאליו מוסיפים ביקורת. בחירה של דירוג הסרט מ־1-5 והוספת ביקורת כתובה. המשתמש צריך למלא את כל השדות. ולאחר מכן לשמור את הביקורת. אם המשתמש מוסיף ביקורת הוא מקבל הודעה שהביקורת שלו מחכה לאישור מנהל, אם המנהל מוסיף ביקורת הביקורת שלו נשמרת ומאושרת אוטומטית.

UsersView.xaml

מוגדר כ־UserControls. המסך זמין רק למנהל והוא מציג את פרטי כל המשתמשים בטבלה. המנהל יכול לעדכן משתמשים ע"י לחיצת הכפתור Update בטבלה או להוסיף משתמש ע"י כפתור Add User בראש העמוד (AddOrUpdateUserView) אחרי ההוספה / עדכון הטבלה מתעדכנת עם הנתונים החדשים.

UsersReviewsView.xaml

מוגדר כ־UserControls. המסך זמין רק למנהל והוא מציג את פרטי כל הביקורות בשתי טבלאות – טבלה אחת לביקורות שעדיין לא אושרו וטבלה לביקורות שאושרו. בטבלה הראשונה המנהל יכול לאשר / לדחות ביקורת ובטבלה השנייה המנהל יכול למחוק ביקורות. אם המנהל מוחק / דוחה ביקורת נפתח לו חלון בו הוא צריך להוסיף את סיבת הדחייה / מחיקה (DeleteReviewView) ורק אז למחוק אותם.

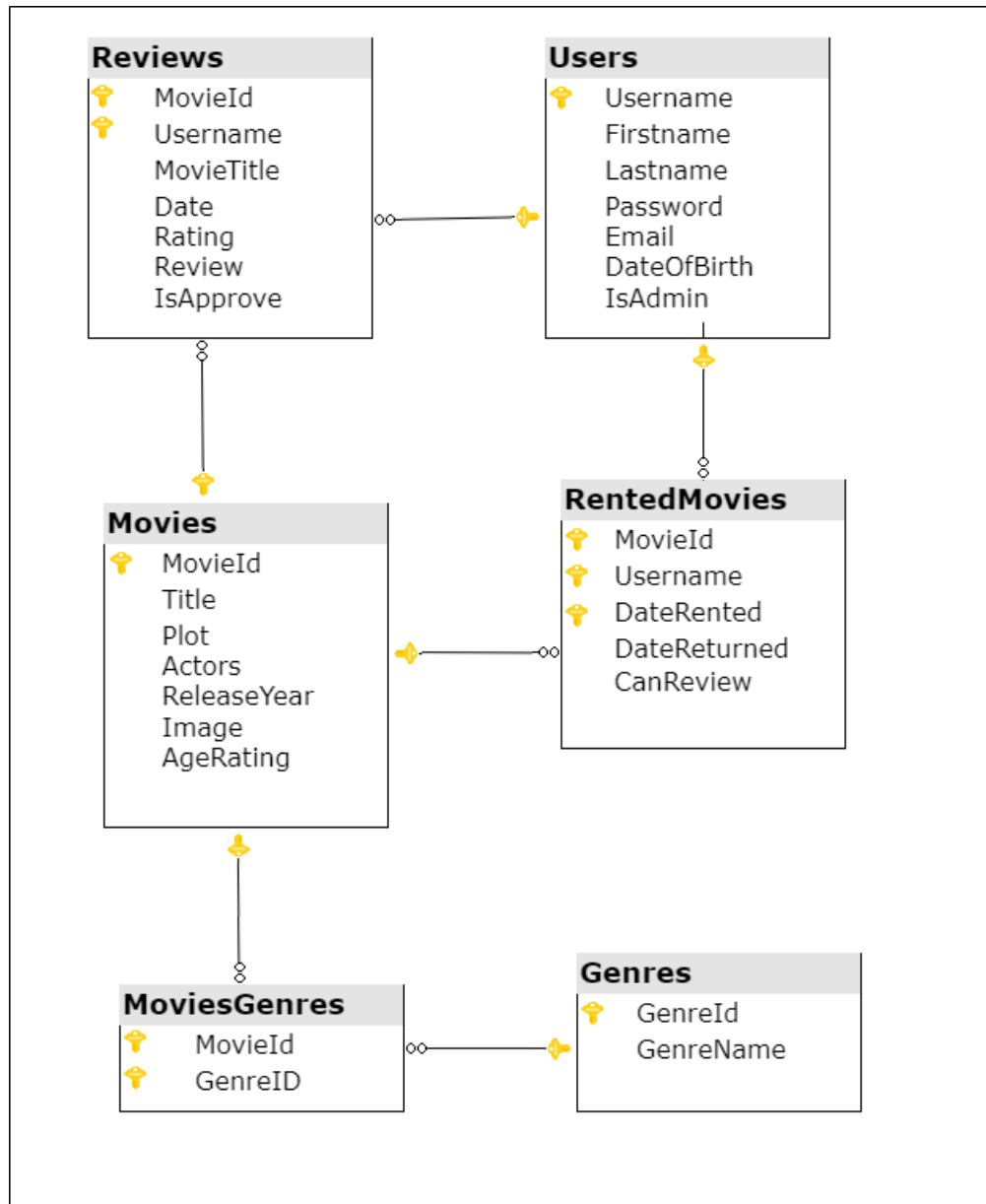
DeleteReviewView.xaml

מוגדר כ־window, החלון נפתח מעל המערכת הראשית ובזמן שהחלון פתוח אין אפשרות לבצע אינטרקציה עם החלון הראשי. בחלון יש שדה בו אפשר להוסיף את הסיבה למחיקה / דחיית הביקורת וכפתור לשמירה. במידה והמנהל שומר את החלון בלי סיבה הוא יקבל הודעה שהוא חייב להכניס סיבה והחלון יסגר.

ReportsView.xaml

מוגדר כ־UserControls. המסך זמין רק למנהל. המסך מציג נתונים סטטיסטיים על המערכת (מספר המשתמשים, מספר הסרטים במערכת, מספר הסרטים המושכרים כרגע, המספר הכולל של סרטים שהושכרו והוחזרו, מספר הזאנרים במערכת ומספר הביקורות המאושרות במערכת) בנוסף יש למנהל אופציה לבחור איזה דוח הוא רוצה להפיק, ללחוץ על כפתור הפקת דוח ושמירתו במחשב המנהל.

דיאגרמת מסד הנתונים (Data Base Diagram)



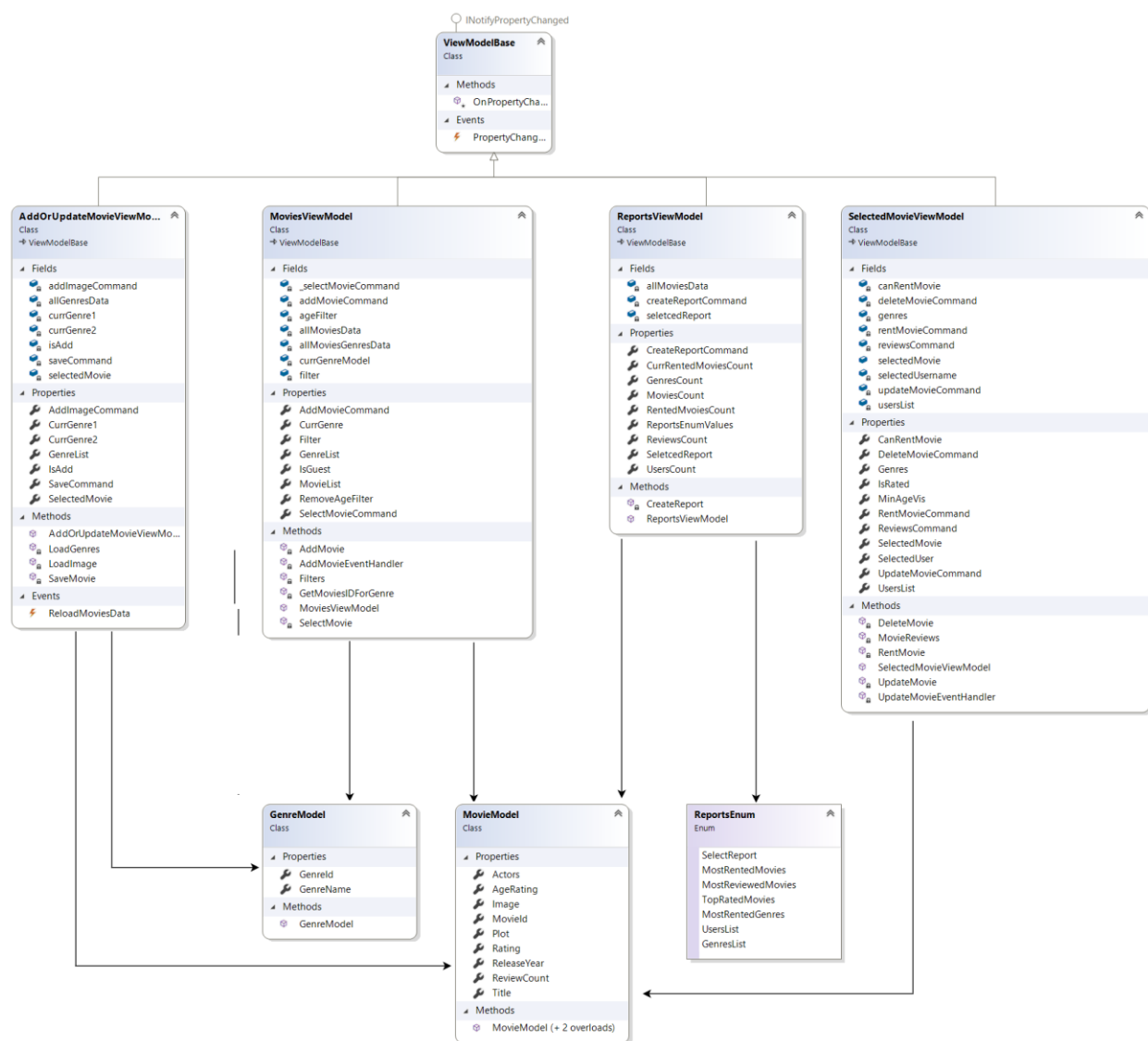
דיאגרמת מחלקות (Class Diagrams)

בעקבות ריבוי מחלקות, חילקתי את הדיאגרמות ל-4 דיאגרמות שונות – דיאגרמת מחלקות שמכילה את המחלקות שמכילות UserModel, דיאגרמת מחלקות שמכילות MovieModel / GenresModel, דיאגרמת מחלקות שמכילות ReviewModel / RentedMovieModel, ודיאגרמת מחלקות של שאר המחלקות. בכל הדיאגרמות המחלקה ViewModelBase נמצאת מכיוון שכל מחלקות הViewModel יורשות ממנה. בדיאגרמות לא הוכנסו המחלקות הסטטיות / Views.

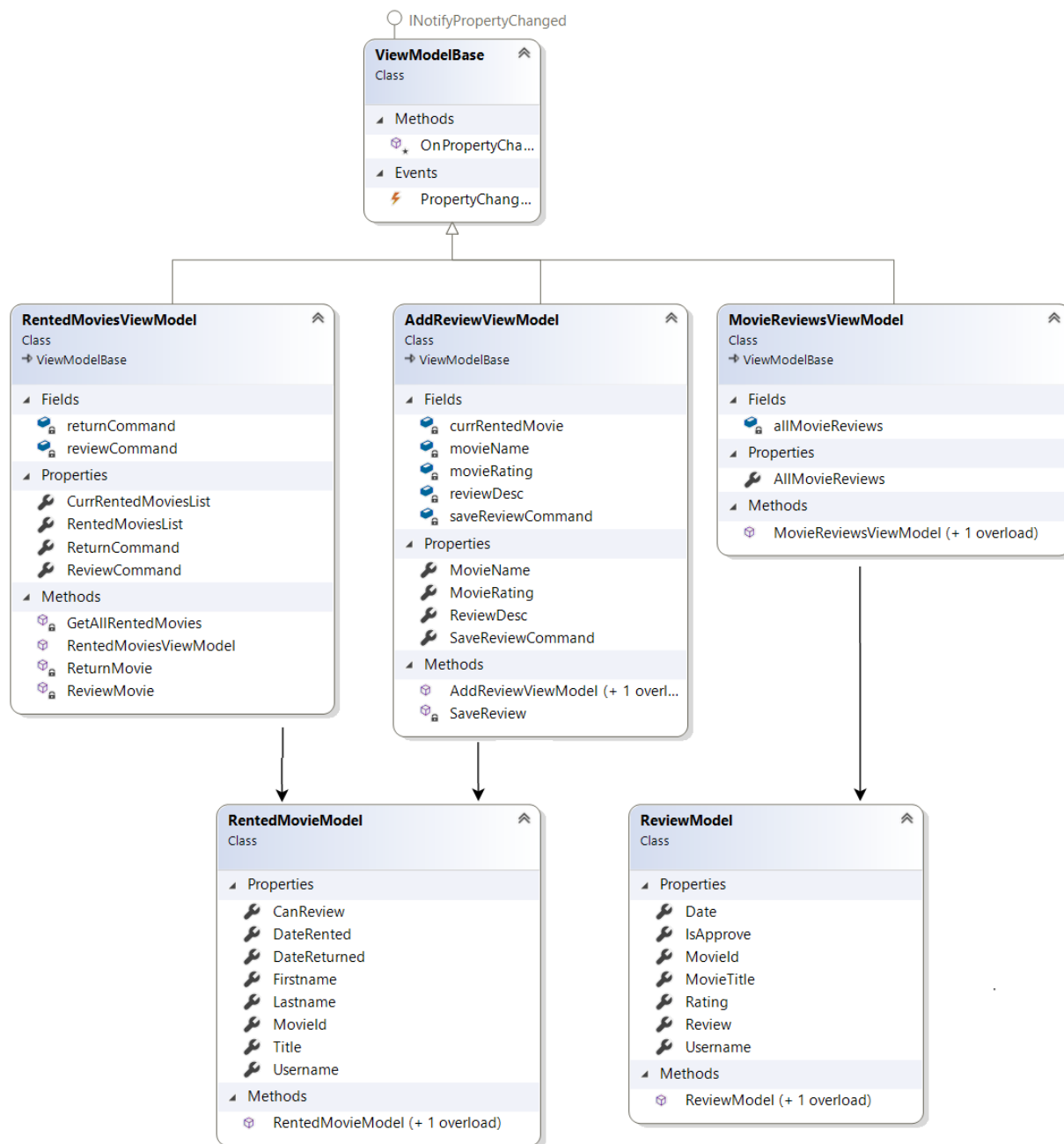
דיאגרמת מחלקות שמכילות UserModel



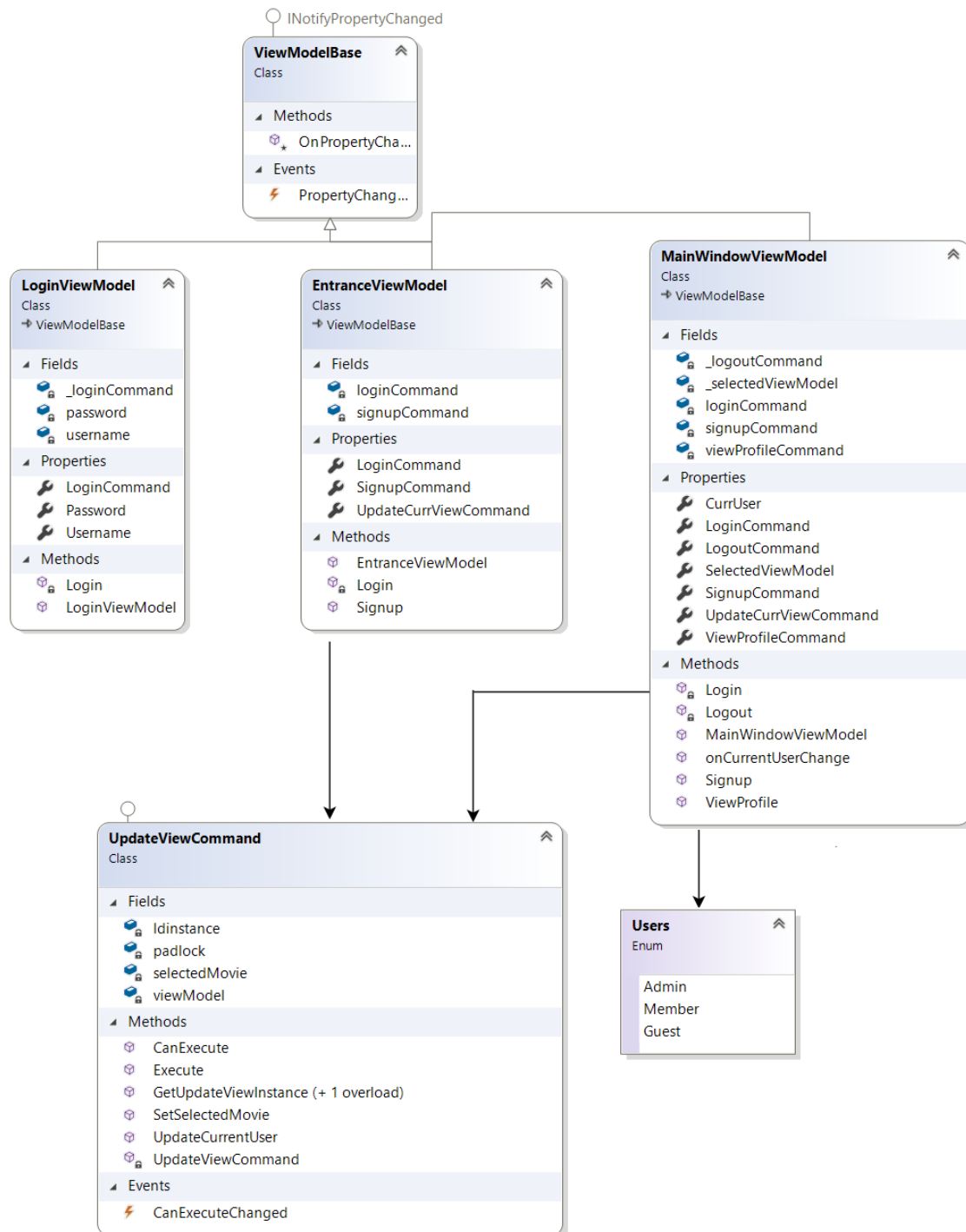
דיאגרמת מחלקות שמכילות MovieModel / GenresModel



דיאגרמת מחלקות שמכילות ReviewModel / RentedMovieModel



דיאגרמת מחלקות של שאר המחלקות



מדריך למערכת

תיאור המערכת

MoviesForYou היא אפליקציה להשכרת סרטים. משתמשים יוכלו להירשם למערכת, לחפש סרטים ולהשכיר אותם.

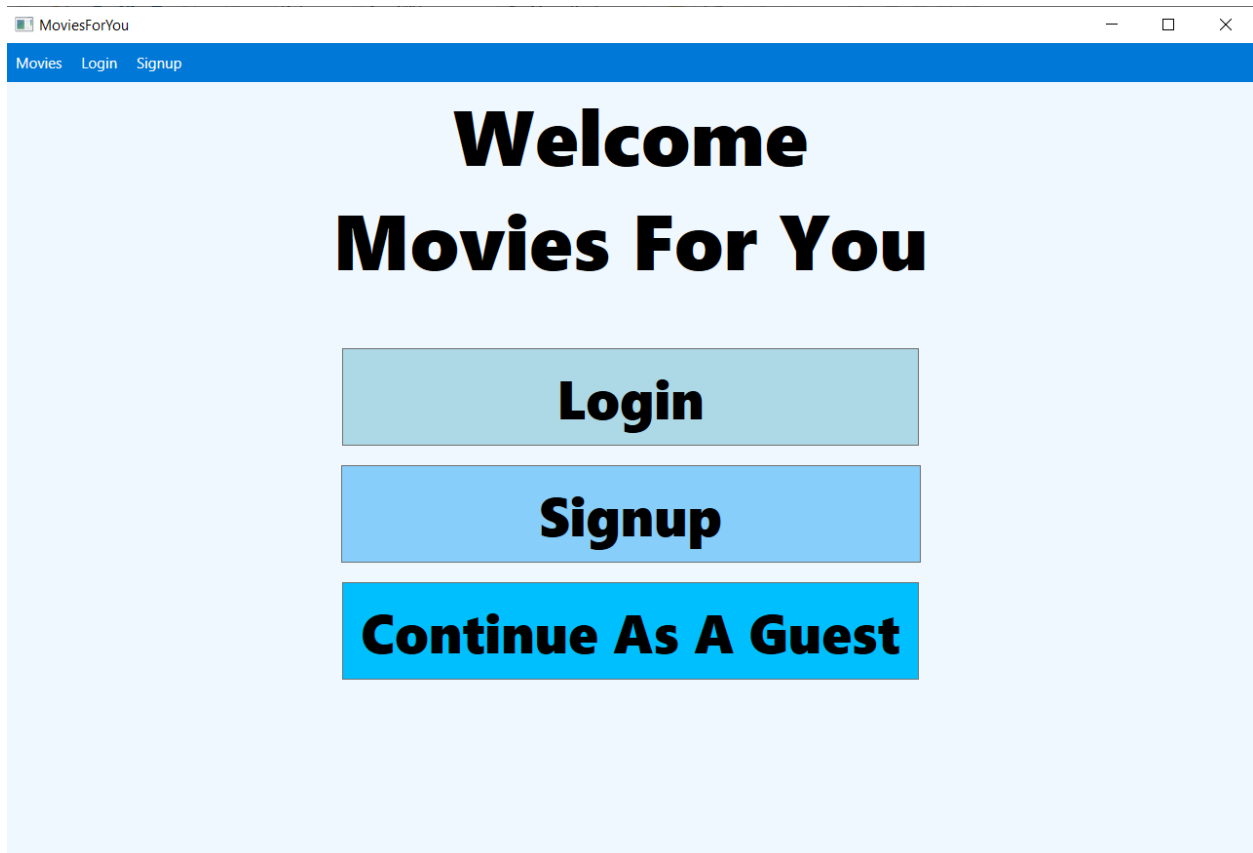
כל משתמש יוכל להשכיר עד 5 סרטים בשבוע. המשתמשים יוכלו לראות ביקורות על כל הסרטים ולהוסיף ביקורת על סרטים שהם השכירו.

MoviesForYou מאפשרת למצוא ולהשכיר סרטים בקלות, לעקוב אחרי הסרטים שהם הזמינו ולראות ולהוסיף ביקורת. במדריך נסביר איך המערכת עובדת ואיך אפשר להשתמש בה.

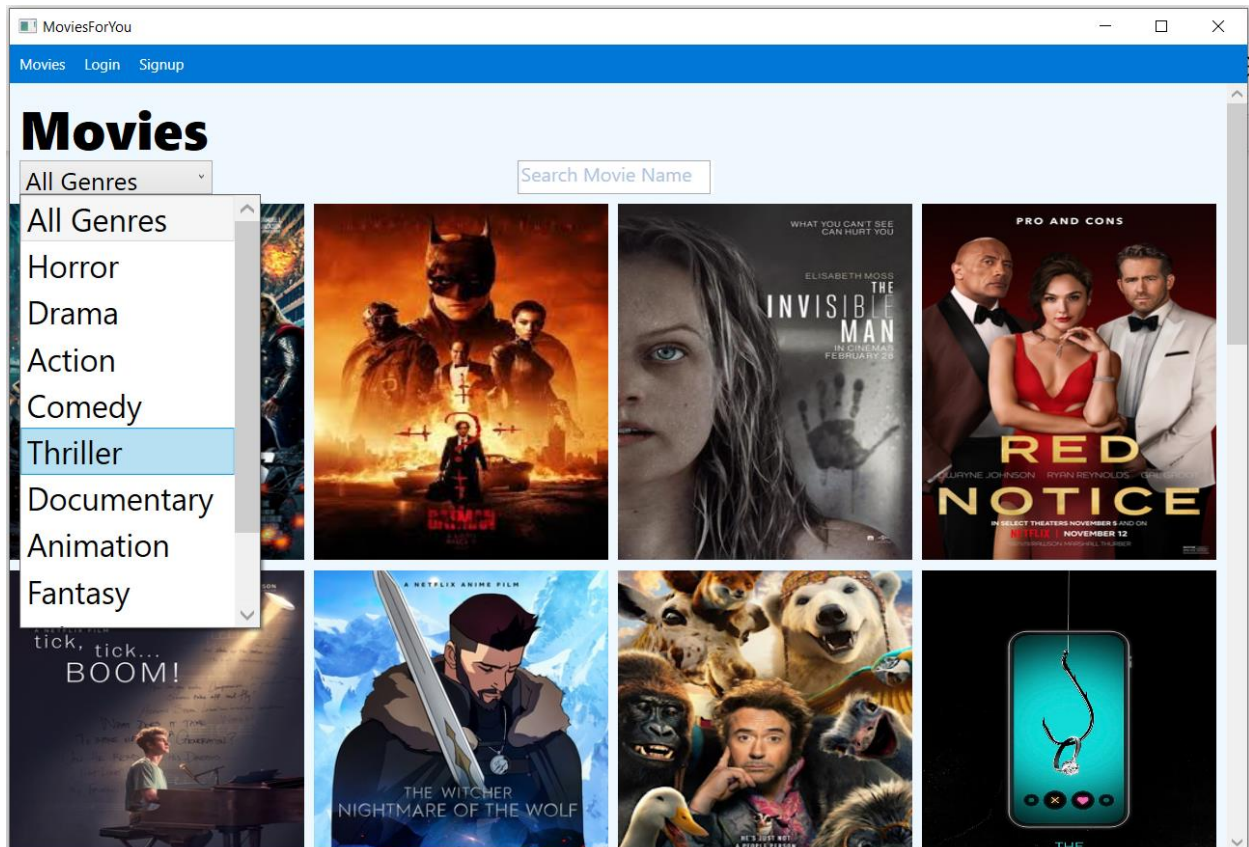
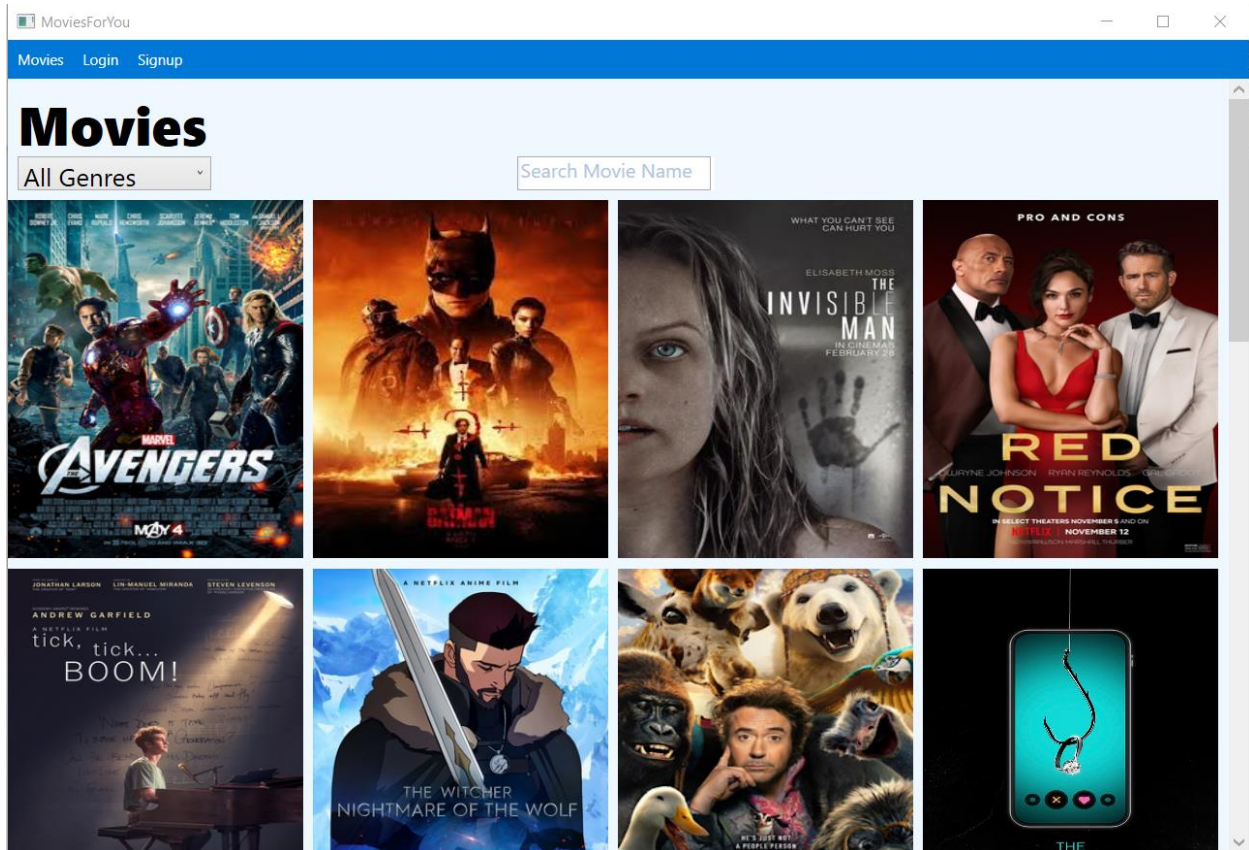
משתמשי המערכת: אורח, משתמש רשום, מנהל.

אורח:

כשמשמש נכנס לאפליקציה הוא יראה את המסך כניסה, בו הוא יוכל להתחבר, להירשם או להמשיך כאורח. אם יבחר להמשיך כאורח הוא יוכל לראות את מבחר הסרטים שיש באפליקציה.

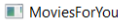


כשמשמשים כאורח או אם נכנסים לאופציה הסרטים באפליקציה נראה את פירוט הסרטים שיש באפליקציה. אם נעמוד עם העכבר על תמונת הסרט באפליקציה, נוכל לראות את השם והתאריך של הסרט. לאורח יש אופציה לסנן את הסרטים לפי מילת חיפוש בשם הסרט או לפי הזאנר שהוא רוצה לראות.




לחיצה על אחד הסרטים תוביל למסך עם פרטים על הסרט – שם, תאריך יציאה, שחקנים, זאנרים, ועלילה, הגבלת הגיל על צפייה בסרט וביקורות. אם אין ביקורות עדיין על הסרט תופיע הודעה על זה. אם יש יהיה אפשר לראות את דירוג הסרט (מ-5) וכמה ביקורות הוא קיבל. כפתור השכרת הסרט הינו בצבע אפור, כלומר, שאינו פעיל עבור אורח מכיוון שאין לו אפשרות להשכיר סרטים.

סרט בלי ביקורות

 MoviesForYou

[Movies](#) [Login](#) [Signup](#)



The Invisible Man

(2020)

Rating This Movie Dont Have Any Reviews

Cecilia's abusive ex-boyfriend fakes his death and becomes invisible to stalk and torment her. She begins experiencing strange events and decides to hunt down the truth on her own.

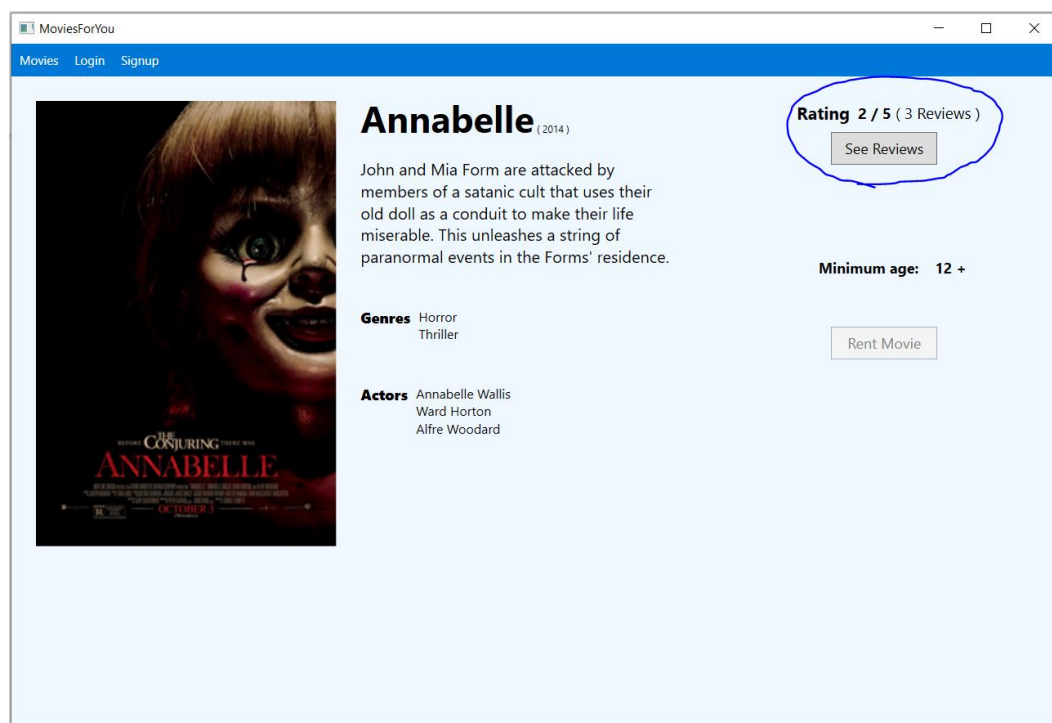
Minimum age: 18 +

Rent Movie

Genres Horror
Thriller

Actors Elisabeth Moss
Aldis Hodge
Storm Reid
Harriet Dyer

סרט עם ביקורות:



Annabelle (2014)

John and Mia Form are attacked by members of a satanic cult that uses their old doll as a conduit to make their life miserable. This unleashes a string of paranormal events in the Forms' residence.

Genres Horror
Thriller

Actors Annabelle Wallis
Ward Horton
Alfre Woodard


Rating 2 / 5 (3 Reviews)

[See Reviews](#)

Minimum age: 12 +

[Rent Movie](#)

אם האורח ילחץ על הכפתור על לראות ביקורות יפתח חלון חדש ובו פירוט של כל ביקורות שכוללת את הביקורת עצמה ואת הדירוג, איזה משתמש הוסיף את הביקורת ומתי.



Annabelle (2014)

Rating 2 / 5 (3 Reviews)

Reviews

2 / 5 The script, by Gary Dauberman, is a cliché mess, lacking humor and surprises. But Leonetti stages some fright-worthy sequences ...
by **aloni** - 3/25/2022

1 / 5 The scares are orchestrated with more flair than usual, even if the best of them are all spoiled in the trailer.
by **avital** - 4/15/2022

3 / 5 Quality is often irrelevant in a horror movie; shock is the key.
by **ronron** - 4/11/2022

12 +

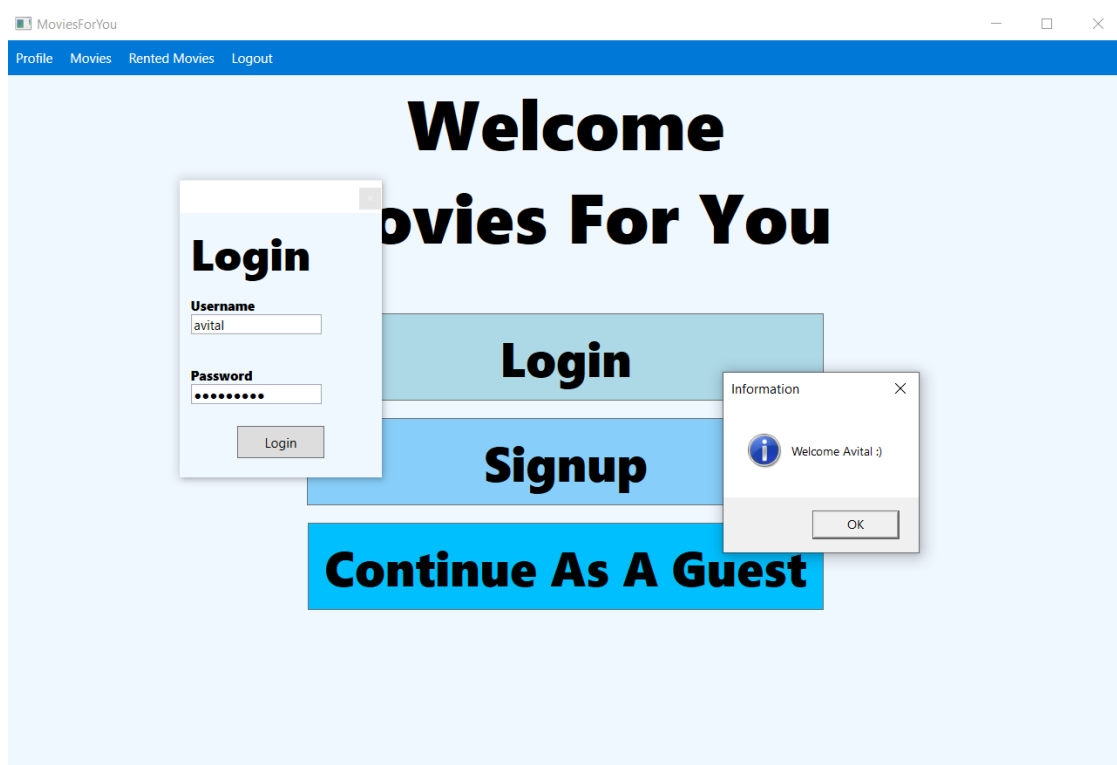
משתמש רשום:

האורח יוכל להירשם למערכת, ולהפוך למשתמש רשום.

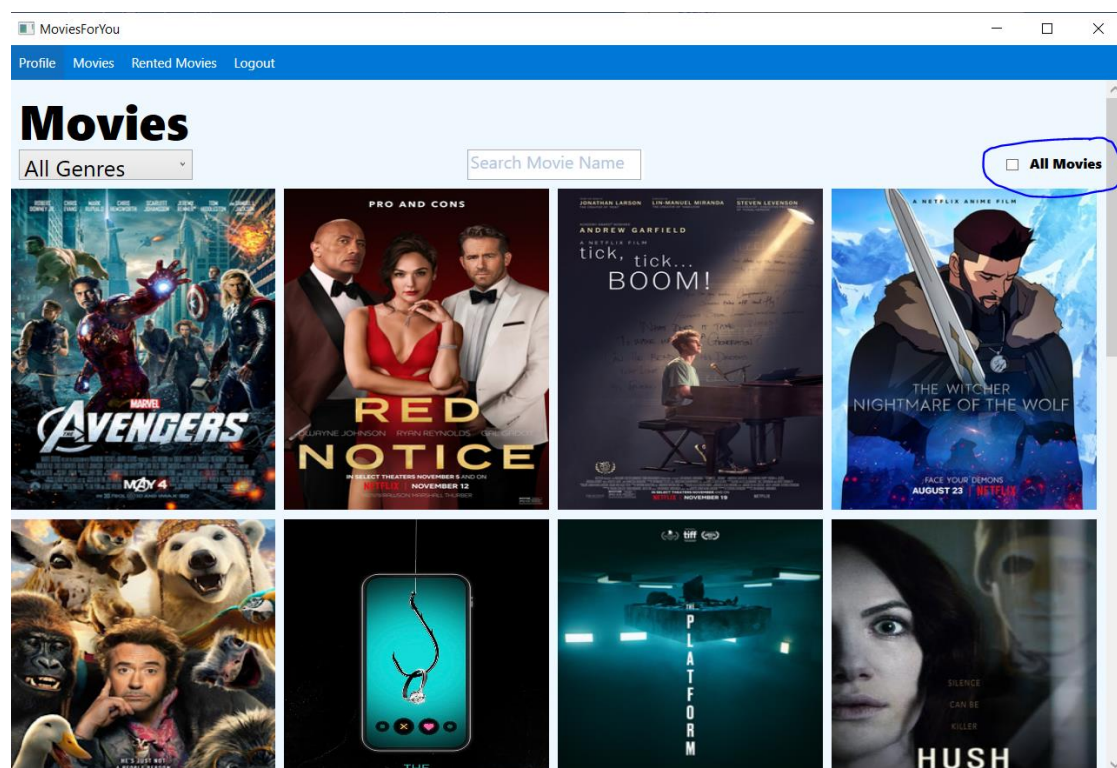
יהיה עליו למלא פרטים (שם משתמש, שם פרטי, שם משפחה, סיסמא, אימייל, ותאריך לידה) המערכת תבדוק שכל הפרטים מלאים ותקינים, שם המשתמש לא תפוס והמייל אינו רשום כבר במערכת. במידה והכל יהיה תקין האורח יקבל הודעה שהוא רשום וכעת יכול להתחבר למערכת.

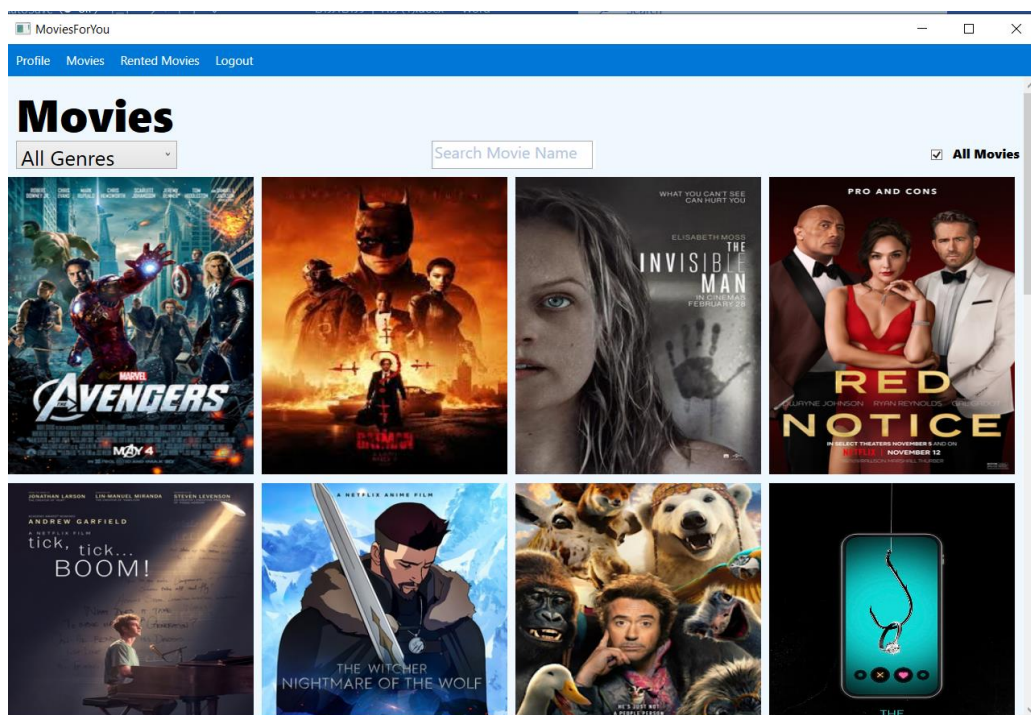
The screenshot shows a web browser window titled "MoviesForYou". The browser's address bar shows "Movies Login Signup". The website has a light blue background. A large, bold, black text "Welcome" is at the top, followed by "ies For You" (partially visible). Below this, there are three large, light blue buttons: "Login", "Signup", and "Continue As A Guest". A sign-up form is overlaid on the left side of the page. The form has a red 'X' close button in the top right corner. It contains the following fields: "Username" (text input), "First name" (text input), "Last Name" (text input), "Password" (text input), "Email" (text input), and "Date of birth" (date picker showing "14/04/2022" and a calendar icon). At the bottom of the form is a "Sign up!" button.

אורח יוכל ללחוץ על התחברות, להזין שם משתמש וסיסמא ובמידה והם נכונים הוא יקבל הודעה על כך שהוא מחובר ויכנס למערכת. המערכת תוביל אותו לעמוד של הסרטים.



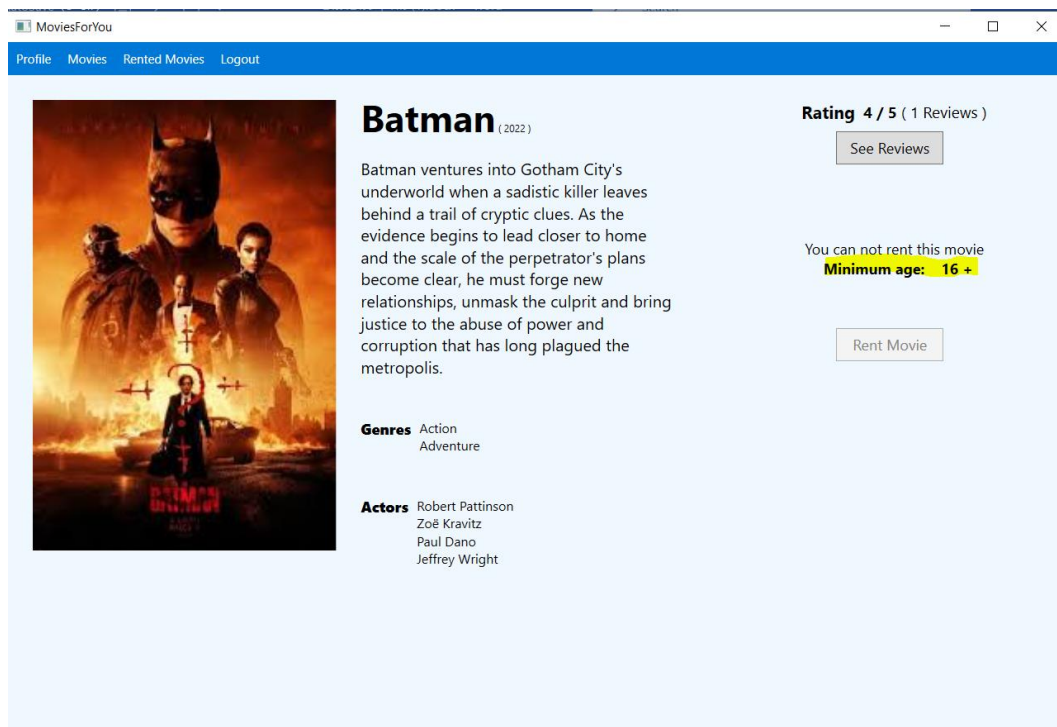
העמוד של הסרטים יראה אותו דבר כמו עמוד הסרטים שאורח רואה אך רשימת הסרטים שהמשתמש יראה תכלול רק סרטים שמתאימים לגיל שלו. אם המשתמש בן 14 לדוגמא הוא יראה תחילה רק את הסרטים שבני 14 ומטה יכולה להשכיר. בנוסף בראש העמוד מצד ימין יהיה כפתור שהמשתמש יוכל ללחוץ עליו על מנת לראות את כל הסרטים הקיימים במערכת. גם אם הוא אינו יכול להשכיר אותם





אם משתמש ילחץ על סרט הוא יעבור למסך עם פרטים על הסרט. המסך הוא אותו מסך כמו שהאורח רואה אך הפעם אם הגיל של המשתמש גדול או שווה להגבלת הגיל של הסרט הוא יוכל להשכיר את הסרט. אם הוא לא יוכל הוא יראה הודעה על כך שהסרט אינו מתאים לו מבחינת הגיל.


סרט עם הגבלת גיל (המשתמש בן 14):



סרט שהמשתמש יכול להשכיר:

MoviesForYou

[Profile](#)
[Movies](#)
[Rented Movies](#)
[Logout](#)



The Avengers (2012)

Earth's Mightiest Heroes stand as the planet's first line of defense against the most powerful threats in the universe. The Avengers began as a group of extraordinary individuals who were assembled to defeat Loki and his Chitauri army in New York City

Rating 4 / 5 (1 Reviews)

[See Reviews](#)

Minimum age: 11 +

[Rent Movie](#)

Genres Action
Adventure

Actors Robert Downey Jr.
Chris Evans
Mark Ruffalo
Chris Hemsworth
Scarlett Johansson
Jeremy Renner
Tom Hiddleston

כשהמשתמש לוחץ על השכרת הסרט, המערכת בודקת שהוא אינו שוכר כרגע את הסרט או שהוא כבר לא שכר 5 סרטים בשבוע האחרון. אם הוא יכול לשכור את הסרט הוא -יועבר למסך של סרטים מושכרים ויראה את כל הסרטים שהוא שכר עד כה.

MoviesForYou

[Profile](#)
[Movies](#)
[Rented Movies](#)
[Logout](#)

Rented Movies List

| Movie Name | Date Rented | | | |
|--------------|-------------|-----------------------------|------------------------------|------------------------|
| The Avengers | 4/14/2022 | Watch Movie | Return Movie | Review |
| Dolittle | 4/12/2022 | Watch Movie | Return Movie | Review |
| Annabelle | 4/14/2022 | Watch Movie | Return Movie | Review |
| The Notebook | 4/12/2022 | Watch Movie | Return Movie | Review |

🕒 All Rental History

| Movie Title | Date Issued | Date Returned | |
|--------------------|-------------|---------------|------------------------|
| Red Notice | 12/28/2021 | 1/13/2022 | Review |
| tick, tick...BOOM! | 10/17/2021 | 10/26/2022 | Review |

במסך של הסרטים המושכרים המשתמש יוכל להחזיר את הסרט בסיום השימוש או להוסיף לו ביקורות שתעבור למנהל לאישור. במידה והיא תאושר הוא יקבל מייל על כך. אם השוכר יחזיר את הסרט הוא יוכל לראות את הסרט בהיסטוריית ההשכרות שלו.

אם המשתמש יבחר בהוספת ביקורת הוא יצטרך לדרג את הסרט מ 1 – 5 ולהוסיף את הביקורת-המשתמש חייב גם לדרג מספרית את הסרט וגם להוסיף מלל. לאחר שהמשתמש מוסיף את הביקורת הוא מקבל הודעה שכתת הביקורת מחכה לאישור מנהל. תשובה תגיע למייל בו נרשם. אם לא יוסיף דירוג ומלל יקבל הודעת שגיאה.

The screenshot displays a web application interface. On the left, a modal window titled 'AddReview' is open for the movie 'The Avengers'. It features a 5-star rating system with the 5th star selected, a text area for the review containing the text 'The story works, the acting is superb, the humor is on key and the action sequence are breathtaking', and a 'Save' button. The main background shows a 'Movies List' section with two tables. The first table, 'Date Rented', has columns for date, 'Watch Movie', 'Return Movie', and 'Review'. The second table, 'Date Issued', has columns for 'Date Issued', 'Date Returned', and 'Review'. An 'Information' modal is also present, displaying the message: 'Your review is pending for admin approval!' with an 'OK' button.

בנוסף, המשתמש יכול לשנות את פרטיו בפרופיל. משתמשים רשומים יכולים לשנות את שמם, שם המשפחה, האימייל (רק במידה והוא לא קיים במערכת) וסיסמא. אין להם אפשרות לשנות את שם המשתמש או את תאריך הלידה שלהם. אם הפרטים עודכנו בהצלחה הם יקבלו הודעת הצלחה על כך.

✕

Username
avital

First name
Avital

Last Name
Yehud

Password
••••••••

Email
avitalyehud@gmail.com

Date of birth
17/08/2007 15

Save

השינוי

✕

Username
avital

First name
Avital

Last Name
Yehudit

Password
••••••••

Email
avitalyehudit@gmail.com

Date of birth
17/08/2007 15

Save

הודעת הצלחה

MoviesForYou

Profile Movies Rented Movies Logout

Rented Movies List

| Movie Name | Date Rented | | | |
|--------------|-------------|-------------|--------------|--------|
| The Avengers | 4/14/2022 | Watch Movie | Return Movie | Review |
| Dolittle | 4/12/2022 | Watch Movie | Return Movie | Review |
| Annabelle | 4/14/2022 | Watch Movie | Return Movie | Review |
| The Notebook | 4/12/2022 | Watch Movie | Return Movie | Review |

⌵ All Rental History

| Movie Title | Date Issued | Date Returned | |
|--------------------|-------------|---------------|--------|
| Red Notice | 12/28/2021 | 1/13/2022 | Review |
| tick, tick...BOOM! | 10/17/2021 | | Review |

Success

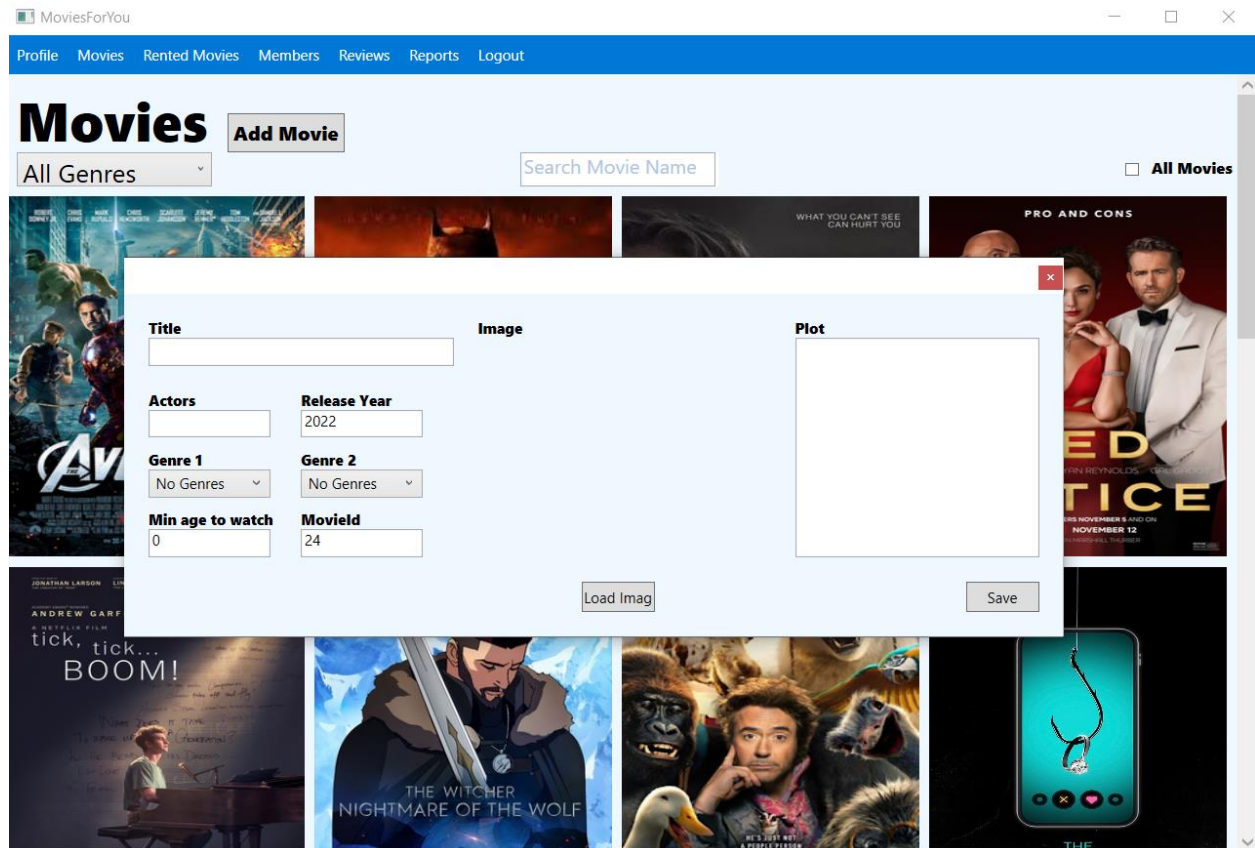
Your profile updated successfully

OK

לבסוף המשתמשים יכולים להתנתק ולחזור למסך הכניסה הראשי.

מנהל מערכת (System Admin)

מנהל המערכת צריך להתחבר גם הוא למערכת והוא יגיע תחילה למסך הסרטים. המסך שלו יראה כמו מסך המשתמש אך יתווסף לו כפתור עם אופציה להוסיף סרט חדש למערכת. בהוספת סרט הוא יצטרך להוסיף את שם הסרט, השחקנים שלו, תאריך יציאת הסרט, זאנרים של הסרט (שהם לא חובה) הגבלת גיל לסרט, ואת הid של הסרט שיוסוף ע"י המערכת, תמונה ואת עלילה הסרט. הסרט יתווסף לעמוד הסרטים רק כאשר כל הפרטים ימולאו.



The screenshot shows the 'Add Movie' form on the MoviesForYou website. The form is a modal window with the following fields and controls:

- Title**: A text input field.
- Image**: A text input field.
- Plot**: A large text area for the movie's plot.
- Actors**: A text input field.
- Release Year**: A text input field with the value '2022'.
- Genre 1**: A dropdown menu with 'No Genres' selected.
- Genre 2**: A dropdown menu with 'No Genres' selected.
- Min age to watch**: A text input field with the value '0'.
- MovieId**: A text input field with the value '24'.
- Load Image**: A button to upload a movie image.
- Save**: A button to save the movie information.

The background of the website shows a grid of movie posters, including 'Avengers: Endgame', 'The Witcher: Nightmare of the Wolf', and 'The Matrix Resurrections'.

אם מנהל המערכת יכנס למסך של סרט מסויים הוא יראה מסך דומה למסך שהמשתמש רואה אך הוא יוכל לעשות מספר פעולות נוספות:

- * הוא יוכל לעדכן או למחוק סרט (הוא לא יכול לעדכן את מספר ה id של סרט)
- * הוא יוכל להשכיר סרט בשביל משתמש אחר. הוא יקבל רשימה של כל המשתמשים באתר וביניהם לחפש מישהו מסוים (עם השלמה אוטומטית של השם), לבחור, ואז להשכיר בשבילו את הסרט, בלי קשר להגבלת הגיל. השכרת הסרט תוביל אותו לעמוד של סרטים מושכרים.

MoviesForYou

Profile

Movies


Rented Movies

Members

Reviews

Reports

Logout



The Avengers (2012)

Rating 4 / 5 (1 Reviews)

See Reviews

Minimum age: 11 +

Choose user

Rent Movie

Update Movie

Delete Movie

Earth's Mightiest Heroes stand as the planet's first line of defense against the most powerful threats in the universe. The Avengers began as a group of extraordinary individuals who were assembled to defeat Loki and his Chitauri army in New York City

Genres

Action

Adventure

Actors

Robert Downey Jr.

Chris Evans

Mark Ruffalo

Chris Hemsworth

Scarlett Johansson

Jeremy Renner

Tom Hiddleston

עדכון סרט :

MoviesForYou

Profile

Movies

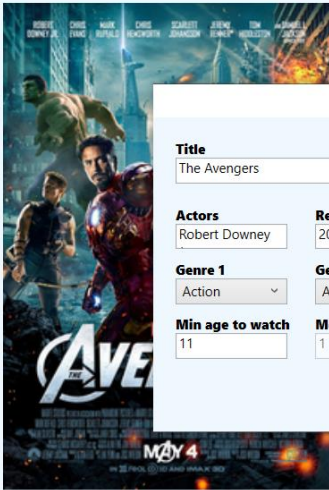
Rented Movies

Members

Reviews

Reports

Logout



The Avengers (2012)

Rating 4 / 5 (1 Reviews)

See Reviews

Earth's Mightiest Heroes stand as the

Title

The Avengers

Actors

Robert Downey

Release Year

2012

Genre 1

Action

Genre 2

Adventure


Min age to watch

11

Moviefid

1

Image



Load Image

Plot

Earth's Mightiest Heroes stand as the planet's first line of defense against the most powerful threats in the universe. The Avengers began as a group of extraordinary individuals who were assembled to defeat Loki and his Chitauri army in New York City

Save

Scarlett Johansson

Jeremy Renner


Tom Hiddleston

48

השכרת הסרט :

MoviesForYou

[Profile](#)
[Movies](#)
[Rented Movies](#)
[Members](#)
[Reviews](#)
[Reports](#)
[Logout](#)



The Avengers (2012)

Earth's Mightiest Heroes stand as the planet's first line of defense against the most powerful threats in the universe. The Avengers began as a group of extraordinary individuals who were assembled to defeat Loki and his Chitauri army in New York City

Genres Action
Adventure

Actors Robert Downey Jr.
Chris Evans
Mark Ruffalo
Chris Hemsworth
Scarlett Johansson
Jeremy Renner
Tom Hiddleston

Rating 4 / 5 (1 Reviews)

[See Reviews](#)

Minimum age: 11 +

- aloni
- avital
- benben
- hazza
- itaytay
- naftaliUz
- noammm
- ronron
- shlomi
- smadi

בעמוד הסרטים המושכרים המנהל יכול לראות את כל הסרטים שמושכרים ע"י כל המשתמשים במערכת. הוא יכול להחזיר בשבילים את הסרטים או להוסיף ביקורת בשבילים (שתאושר במייד) ובנוסף הוא יכול לראות את היסטוריית ההשכרות של כל המשתמשים.

MoviesForYou

[Profile](#)
[Movies](#)
[Rented Movies](#)
[Members](#)
[Reviews](#)
[Reports](#)
[Logout](#)

Rented Movies List

| Movie Name | Username | First Name | Last Name | Date Rented | | | |
|-------------------|-----------|------------|-----------|-------------|-------------|--------------|--------|
| The Avengers | avital | Avital | Yehud | 4/14/2022 | Watch Movie | Return Movie | Review |
| The Avengers | hazza | zohar | Hazoom | 4/12/2022 | Watch Movie | Return Movie | Review |
| The Avengers | ronron | Ronen | Hazoom | 4/12/2022 | Watch Movie | Return Movie | Review |
| Batman | aloni | Alon | Wegner | 4/12/2022 | Watch Movie | Return Movie | Review |
| Batman | smadi | Smader | Shimon | 4/12/2022 | Watch Movie | Return Movie | Review |
| The Invisible Man | aloni | Alon | Wegner | 4/12/2022 | Watch Movie | Return Movie | Review |
| The Invisible Man | hazza | zohar | Hazoom | 4/12/2022 | Watch Movie | Return Movie | Review |
| The Invisible Man | smadi | Smader | Shimon | 4/12/2022 | Watch Movie | Return Movie | Review |
| Dolittle | avital | Avital | Yehud | 4/12/2022 | Watch Movie | Return Movie | Review |
| Dolittle | naftaliUz | Naftali | Uzzi | 4/12/2022 | Watch Movie | Return Movie | Review |
| Dolittle | noammm | Noam | David | 4/12/2022 | Watch Movie | Return Movie | Review |
| Hush | ronron | Ronen | Hazoom | 4/12/2022 | Watch Movie | Return Movie | Review |
| Annabelle | avital | Avital | Yehud | 4/14/2022 | Watch Movie | Return Movie | Review |
| Annabelle | ronron | Ronen | Hazoom | 4/10/2022 | Watch Movie | Return Movie | Review |
| The Notebook | avital | Avital | Yehud | 4/12/2022 | Watch Movie | Return Movie | Review |
| The Blind Side | ronron | Ronen | Hazoom | 4/12/2022 | Watch Movie | Return Movie | Review |

All Rental History

| Movie Title | Username | Name | Surname | Date Issued | Date Returned | |
|--------------------|----------|--------|---------|-------------|---------------|--------|
| Annabelle | aloni | Alon | Wegner | 3/14/2022 | 3/26/2022 | Review |
| Brokeback Mountain | aloni | Alon | Wegner | 1/27/2022 | 2/3/2022 | Review |
| Hush | aloni | Alon | Wegner | 2/9/2022 | 2/13/2022 | Review |
| Red Notice | avital | Avital | Yehud | 12/28/2021 | 1/13/2022 | Review |
| The Avengers | aloni | Alon | Wegner | 4/12/2022 | 4/14/2022 | Review |

מסך נוסף שיש למנהל הוא מסך המשתמשים. המנהל יכול לראות את כל המשתמשים הקיימים ואת הפרטים שלהם. הוא יכול לעדכן את הפרטים של המשתמשים ולהפוך אותם למנהלים. בנוסף הוא יכול להוסיף משתמש חדש למערכת. במידה והמנהל מעדכן / מוסיף משתמש הוא יקבל הודעת אישור על המסך.

MoviesForYou

Profile Movies Rented Movies Users Reviews Reports Logout

Users List

Add Users

| Username | Name | Surname | Password | Email | Birth Of Date | Is Admin | |
|-----------|---------|---------|---------------|------------------------|---------------|-------------------------------------|-------------|
| adico | Adi | Cohen | Adico1998 | adico@gmail.com | 11/17/1998 | <input checked="" type="checkbox"/> | Update user |
| aloni | Alon | Wegner | Aloniw98 | alonwegner@gmail.com | 1/31/1998 | <input type="checkbox"/> | Update user |
| avital | Avital | Yehud | AviYeh123 | avitalyehud@gmail.com | 8/17/2007 | <input type="checkbox"/> | Update user |
| benben | Ben | Cohen | Benben1212 | benben@gmail.com | 7/12/2001 | <input type="checkbox"/> | Update user |
| hazza | zohar | Hazoom | ZoharTheKing1 | asmity22@gmail.com | 9/27/1998 | <input type="checkbox"/> | Update user |
| itaytay | Itay | Tsur | ItayTsur30 | itaytsur30@gmail.com | 6/19/2014 | <input type="checkbox"/> | Update user |
| naftaliUz | Naftali | Uzzi | NafUz12 | naftaliuzzi@gmail.com | 9/12/2010 | <input type="checkbox"/> | Update user |
| noammm | Noam | David | NoamDavid1 | noamDavid1@gmail.com | 5/3/2016 | <input type="checkbox"/> | Update user |
| ronron | Ronen | Hazoom | Hazoom1122 | ronenhaz@gmail.com | 9/27/1990 | <input type="checkbox"/> | Update user |
| shlomi | Shlomi | Ezra | Password1 | shlomiezra@gmail.com | 9/27/2002 | <input type="checkbox"/> | Update user |
| smadi | Smader | Shimon | smadiS64 | smadarshimon@gmail.com | 3/25/1964 | <input type="checkbox"/> | Update user |

MoviesForYou

Reviews Reports Logout

Add Users

Username: avital

First name: Avital

Last Name: Yehud

Password:

Email: avitalyehud@gmail.com

Date of birth: 17/08/2007

Is Admin: ☐

Save

| Username | Name | Surname | Password | Email | Birth Of Date | Is Admin | |
|-----------|---------|---------|---------------|------------------------|---------------|-------------------------------------|-------------|
| adico | Adi | Cohen | Adico1998 | adico@gmail.com | 11/17/1998 | <input checked="" type="checkbox"/> | Update user |
| aloni | Alon | Wegner | Aloniw98 | alonwegner@gmail.com | 1/31/1998 | <input type="checkbox"/> | Update user |
| avital | Avital | Yehud | AviYeh123 | avitalyehud@gmail.com | 8/17/2007 | <input type="checkbox"/> | Update user |
| benben | Ben | Cohen | Benben1212 | benben@gmail.com | 7/12/2001 | <input type="checkbox"/> | Update user |
| hazza | zohar | Hazoom | ZoharTheKing1 | asmity22@gmail.com | 9/27/1998 | <input type="checkbox"/> | Update user |
| itaytay | Itay | Tsur | ItayTsur30 | itaytsur30@gmail.com | 6/19/2014 | <input type="checkbox"/> | Update user |
| naftaliUz | Naftali | Uzzi | NafUz12 | naftaliuzzi@gmail.com | 9/12/2010 | <input type="checkbox"/> | Update user |
| noammm | Noam | David | NoamDavid1 | noamDavid1@gmail.com | 5/3/2016 | <input type="checkbox"/> | Update user |
| ronron | Ronen | Hazoom | Hazoom1122 | ronenhaz@gmail.com | 9/27/1990 | <input type="checkbox"/> | Update user |
| shlomi | Shlomi | Ezra | Password1 | shlomiezra@gmail.com | 9/27/2002 | <input type="checkbox"/> | Update user |
| smadi | Smader | Shimon | smadiS64 | smadarshimon@gmail.com | 3/25/1964 | <input type="checkbox"/> | Update user |

The screenshot shows a web application titled 'MoviesForYou' with a navigation bar containing 'Profile', 'Movies', 'Rented Movies', 'Users', 'Reviews', 'Reports', and 'Logout'. A modal form for adding a new user is open, featuring fields for Username, First name, Last Name, Password, Email, and Date of birth, along with an 'Is Admin' checkbox and a 'Save' button. In the background, a table lists existing users with columns for Password, Email, Birth Of Date, Is Admin, and an 'Update user' button for each row.

| Password | Email | Birth Of Date | Is Admin | |
|---------------|------------------------|---------------|-------------------------------------|-------------|
| Adico1998 | adico@gmail.com | 11/17/1998 | <input checked="" type="checkbox"/> | Update user |
| Aloniw98 | alonwegner@gmail.com | 1/31/1998 | <input type="checkbox"/> | Update user |
| AviYeh123 | avitalyehud@gmail.com | 8/17/2007 | <input type="checkbox"/> | Update user |
| Benben1212 | benben@gmail.com | 7/12/2001 | <input type="checkbox"/> | Update user |
| ZoharTheKing1 | asmity22@gmail.com | 9/27/1998 | <input type="checkbox"/> | Update user |
| ItayTsur30 | itaytsur30@gmail.com | 6/19/2014 | <input type="checkbox"/> | Update user |
| NafUz12 | naftaliuzzi@gmail.com | 9/12/2010 | <input type="checkbox"/> | Update user |
| NoamDavid1 | noamDavid1@gmail.com | 5/3/2016 | <input type="checkbox"/> | Update user |
| Hazoom1122 | ronenhaz@gmail.com | 9/27/1990 | <input type="checkbox"/> | Update user |
| Password1 | shlomiezza@gmail.com | 9/27/2002 | <input type="checkbox"/> | Update user |
| smadiS64 | smadarshimon@gmail.com | 3/25/1964 | <input type="checkbox"/> | Update user |

למנהל יש אופציה לראות את כל הביקורות שניתנו לסרטים.

בשקף התחתון, ניתן לראות את הטבלה העליונה המכילה את הביקורות שעדיין לא אושרו. המנהל יוכל להסתכל על הפרטים של נותן הביקורת ואת הביקורת עצמה ולהחליט האם היא מאושרת או לא. במידה והמנהל מאשר את הביקורת, מייל ישלח למשתמש שהביקורת שלו אושרה והיא זמינה לצפייה בעמוד הסרט. הביקורת תועבר לטבלה התחתונה שהיא הטבלה של הביקורות המאושרות וכעת ניתן יהיה לראותה בביקורות של הסרטים בדף שלהם. במידה והמנהל בוחר לא לאשר ביקורת או למחוק ביקורת קיימת יפתח לו מסך נוסף שבו הוא צריך להוסיף את הסיבה שהוא לא מאשר / מוחק את הביקורת. המשתמש יקבל מייל על כך שהביקורת שלו נמחקה / לא אושרה ואת הסיבה על כך.

MoviesForYou

[Profile](#)
[Movies](#)
[Rented Movies](#)
[Members](#)
[Reviews](#)
[Reports](#)
[Logout](#)

Reviews List

| Movie Title | Username | Review Date | Rating | Review | | |
|-------------------|-----------|-------------|--------|---|---------|---------|
| The Avengers | avital | 4/14/2022 | 5 | The story works, the acting is superb, the humor is on key and the action sequence are breathtaking | Approve | Decline |
| The Avengers | hazza | 4/14/2022 | 5 | The comedic timing, coupled with his ability to maintain the authenticity of the characters, is worthy of praise. The Avengers delivers a popcorn feast thats the perfect summer kickoff. | Approve | Decline |
| The Invisible Man | smadi | 4/12/2022 | 3 | Quality is often irrelevant in a horror movie; shock is the key. | Approve | Decline |
| Dolittle | naftaliUz | 4/13/2022 | 3 | Quality is often irrelevant in a horror movie; shock is the key. | Approve | Decline |

All Approve Reviews

| Movie Title | Username | Review Date | Rating | Review | |
|--------------|----------|-------------|--------|---|--------|
| The Avengers | ronron | 4/13/2022 | 4 | Downeys Stark/Iron Man steals the show because the actor is capable of making the tritest dialogue sound wonderfully witty, but the CGI effects are amazing and, despite its length, the film is very enjoyable. | Delete |
| Batman | aloni | 4/13/2022 | 4 | What this film does achieve, however, is telling a solid new Batman story, one with some pretty compelling twists and a strong point-of-view on who, exactly, the Caped Crusader is. By default, that makes it one of the better Batman movies ever made. | Delete |
| Annabelle | aloni | 3/25/2022 | 2 | The script, by Gary Dauberman, is a clichd mess, lacking humor and surprises. But Leonetti stages some fright-worthy sequences ... | Delete |
| Annabelle | avital | 4/15/2022 | 1 | The scares are orchestrated with more flair than usual, even if the best of them are all spoiled in the trailer. | Delete |
| Annabelle | ronron | 4/11/2022 | 3 | Quality is often irrelevant in a horror movie; shock is the key. | Delete |

MoviesForYou

[Profile](#)
[Movies](#)
[Rented Movies](#)
[Members](#)
[Reviews](#)
[Reports](#)
[Logout](#)

Reviews List

| Movie Title | Username | Review Date | Rating | Review | | |
|-------------------|-----------|-------------|--------|---|---------|---------|
| The Avengers | avital | 4/14/2022 | 5 | The story works, the acting is superb, the humor is on key and the action sequence are breathtaking | Approve | Decline |
| The Avengers | hazza | 4/14/2022 | 5 | The comedic timing, coupled with his ability to maintain the authenticity of the characters, is worthy of praise. The Avengers delivers a popcorn feast thats the perfect summer kickoff. | Approve | Decline |
| The Invisible Man | smadi | 4/12/2022 | 3 | Quality is often irrelevant in a horror movie; shock is the key. | Approve | Decline |
| Dolittle | naftaliUz | 4/13/2022 | 3 | Quality is often irrelevant in a horror movie; shock is the key. | Approve | Decline |

All Approve Reviews

| Movie Title | Username | Review Date | Rating | Review | |
|--------------|----------|-------------|--------|---|--------|
| The Avengers | ronron | 4/13/2022 | 4 | Downeys Stark/Iron Man steals the show because the actor is capable of making the tritest dialogue sound wonderfully witty, but the CGI effects are amazing and, despite its length, the film is very enjoyable. | Delete |
| Batman | aloni | 4/13/2022 | 4 | What this film does achieve, however, is telling a solid new Batman story, one with some pretty compelling twists and a strong point-of-view on who, exactly, the Caped Crusader is. By default, that makes it one of the better Batman movies ever made. | Delete |
| Annabelle | aloni | 3/25/2022 | 2 | The script, by Gary Dauberman, is a clichd mess, lacking humor and surprises. But Leonetti stages some fright-worthy sequences ... | Delete |
| Annabelle | avital | 4/15/2022 | 1 | The scares are orchestrated with more flair than usual, even if the best of them are all spoiled in the trailer. | Delete |
| Annabelle | ronron | 4/11/2022 | 3 | Quality is often irrelevant in a horror movie; shock is the key. | Delete |

Reason:

Save

Review Update Inbox x



Review Update <movierentalreviewupdate@gmail.com>

to me ▼

Hi Smader Shimon,

Your Review of The Invisible Man was approve!
You can see the review in the movie page on the app!

MovieRental



Review Update <movierentalreviewupdate@gmail.com>

to me ▼

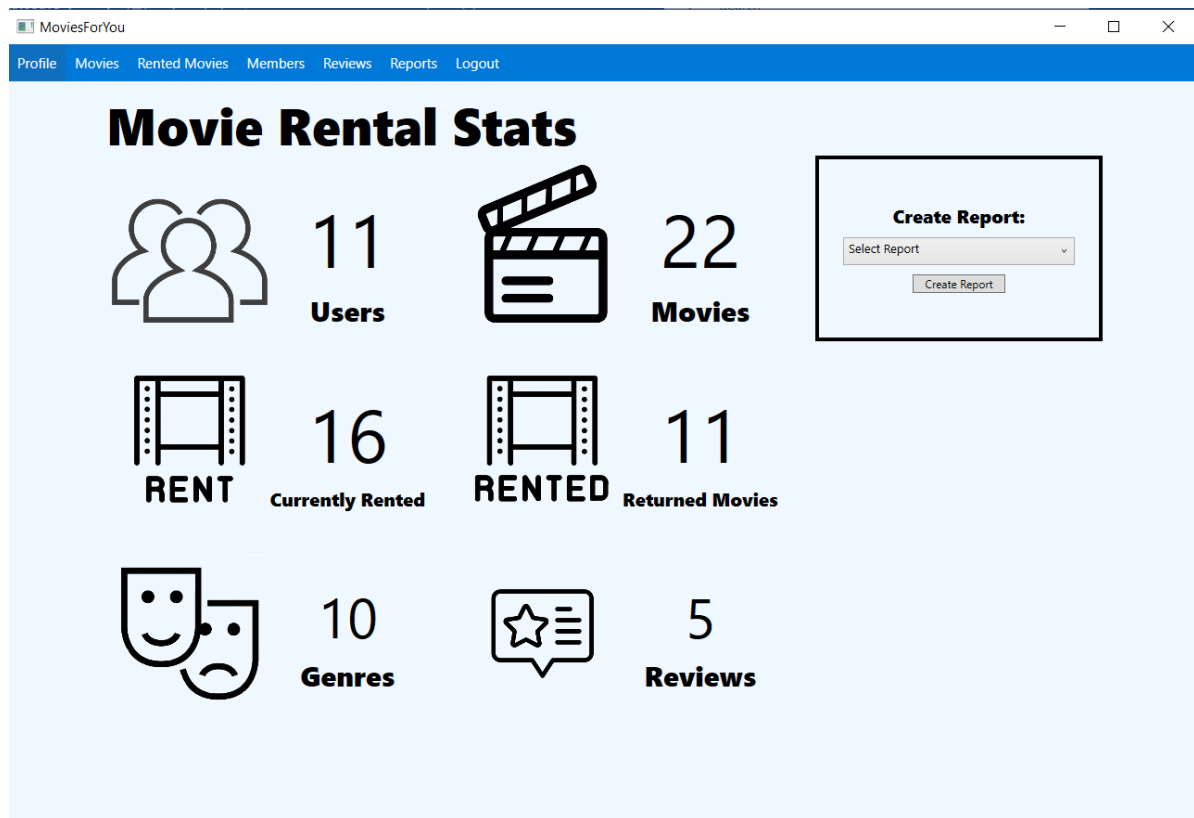
Hi Alon Wegner,

Your Review of Batman was unfortantly delete :(
The admin deleted your review beacause the review was too short .

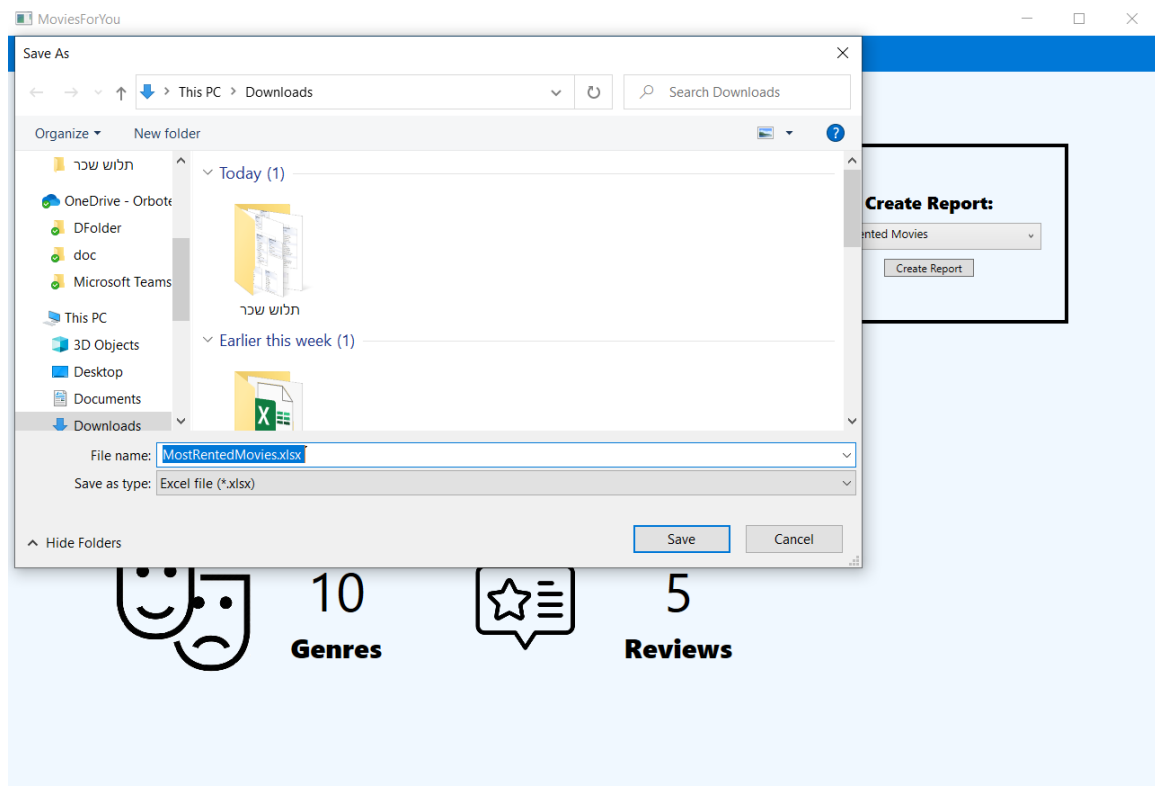
MovieRental

לבסוף המנהל יכול להגיע למסך הדוחות שבו הוא יכול לראות נתונים על המערכת. הוא יכול לבחון את מספר המשתמשים, מספר הסרטים הקיימים במערכת, מספר הסרטים המושכרים כעת, מספר הסרטים שהושכרו בעבר, מספר הזאנרים שיש במערכת ומספר הביקורות שאושרו.

בנוסף הוא יוכל לבחור דוח ולשמור excel של הדוח עם הפירוט. המנהל יכול לבחור אחד מבין 4 הדוחות – הסרטים מדורגים לפי כמות ההשכרות שלהם, הזאנרים מדורגים לפי כמות ההשכרות שלהם, הסרטים מדורגים לפי הדירוג שלהם ע"י המשתמשים והסרטים מדורגים לפי כמות הביקורות עליהם באתר.



שמירת דוח :



AutoSave OFF MostRentedMovies.xlsx Search Adi Cohen AC

File Home Insert Page Layout Formulas Data Review View Help

Clipboard Font Alignment Number Styles Cells Editing Ideas Sensitivity

Movie Id

| Movie Id | Movie Title | Renting Amount |
|----------|---------------------|----------------|
| 1 | The Avengers | 4 |
| 7 | Dolittle | 3 |
| 3 | The Invisible Man | 3 |
| 12 | Annabelle | 3 |
| 11 | Hush | 2 |
| 2 | Batman | 2 |
| 9 | The Tinder Swindler | 1 |
| 10 | The Platform | 1 |
| 4 | Red Notice | 1 |
| 5 | tick, tick...BOOM! | 1 |
| 6 | The Witcher | 1 |
| 13 | The Conjuring | 1 |
| 18 | The Notebook | 1 |
| 19 | The Theory of Every | 1 |
| 20 | Brokeback Mountai | 1 |
| 22 | The Blind Side | 1 |

Most Rented Movies

בנוסף מנהל יכול להיכנס לפרופיל שלו ולשנות את כל הפרטים.

MoviesForYou

Profile Movies Rented Movies Members Reviews Reports Logout

al Stats

22 Movies

11 RENTED Returned Movies

5 Reviews

Create Report:

Most Rented Movies

Create Report

Username

adico

First name

Adi

Last Name

Cohen

Password

.....

Email

adico@gmail.com

Date of birth

17/11/1998

Save