

Team: G#41

FCIS 2026

Watch IT Project

2022170105

2022170094

2022170107

2022170462

2022170453

2022170673

2022170524

• تسنيم محمد أحمد محمد

• بسملة طارق كمال الباجوري

• تقى كرم عبد المجيد مدني

• ندى ايمن محمد إسماعيل

• مي عز الدين سعد

• يوسف احمد سيد ابراهيم

• يوسف محسن رضا جبريل

The Classes:

Admin

Functions:

- ◆ **public String get Username()**

This method doesn't take any parameters and return a string (the username of admin)

- ◆ **public String getPassword()**

This method doesn't take any parameters and return a string (the password of admin)

- ◆ **public void set Username(String username)**

This method takes one parameter from type String (username of admin) and set the username of admin by this

- ◆ **public void set Password(String password)**

This method takes one parameter from type String (password of admin) and set the password of admin by this

- ◆ **public static void read File(File file)**

This method takes one parameter from type File (the file which we write data of users , casts and directors inside it)

- ◆ **public static void read Movies(File file)**

This method takes one parameter from type File (the file which we write data of movies inside it) and read data from it.

◆ **public static void write On File(File file)**

This method takes one parameter from type File(the file which we write data of users , casts and directors inside it) and write data inside it.

◆ **public static void write Movies(File file)**

This method takes one parameter from type File(the file which we write data of movies inside it) and write data inside it.

◆ **public static void setData()**

This method doesn't take any parameters , doesn't return any values , call some methods to get the data of and set this data inside the static arraylists.

◆ **public static String exists In File(String**

This method takes one parameter from type String (username of user),Search about the user who has this user name and return the all data of user in one value from type String.

◆ **static void get Users In Each Plan()**

This method doesn't take any parameters , doesn't return any values,calculate the number of the subscriptions of the each plan.

◆ **static void get User Movie Lists(User user)**

This method takes one parameter (object of User class) , call method exist in file to Search about the user who has this user name , get the movies that he/she watched it and will watch it later and set in static arraylists.

◆ **static void get User Movie Lists_obj(User user)**

This method takes one parameter (object of User class) , get the lists of movies which watched and will watch later , set data inside the arraylist of user and set the rate which user will enter it to the watched movies.

◆ **static ArrayList<Movie> getMoviesObjs()**

This method doesn't take any parameters , call the getonemovie methods to get the data of movies and return arraylist of Movie class.

◆ **static Movie getOneMovie(String movieString)**

This method takes one parameter from type String (the data of movie in one String) , get the data of the movie(get movie as object).

◆ **static Movie getUnkownMovie(String title)**

This method takes one parameter from type String (the title of movie) , search about it by its title and get the data of this movie(get movie as object).

◆ **public static void castAndGenres(String Movie, ArrayList<String> cast_, ArrayList<String> genres)**

This method takes three parameter from type String (the data of movie in one String , arraylist of the data of each cast and the genres of the movie) and set the data of (casts and genres) in this arraylists.

◆ **static ArrayList<Director> getAllDirectors()**

This method doesn't take any parameters , call the getonedirector method to get the data of director , set all directors in arraylist(get director as object) and return this array.

◆ **static Director getDirector(String directorName)**

This method takes one parameter from type String (the name of director) , search about it by its name , set the data of this director(get director as object) and return it(the object of Director).

◆ **static ArrayList<Cast> getAllCast()**

This method doesn't take any parameters , call the getonecast method to get the data of cast , set all casts in arraylist(get cast as object) and return this array.

◆ **static Cast getCast(String name)**

This method takes one parameter from type String (the name of cast) , search about it by its name , set the data of this cast(get cast as object) and return it(the object of cast).

◆ **static ArrayList<User> getAllUsers()**

This method doesn't take any parameters , get the data of user , set all users in arraylist(get user as object) and return this array.

◆ **static ArrayList<Admin> getAllAdmins()**

This method doesn't take any parameters , get the data of admin , set all admins in arraylist(get admin as object) and return this array.

◆ **public String toString()**

This method doesn't take any parameters , get the data of admin , conceit his named with "admin" in one string and return it.

User

Functions:

◆ **public Subscription getSubscription()**

This method doesn't take any parameters and return object of Subscription (the subscription of user).

◆ **public int getUser_ID()**

This method doesn't take any parameters and return int (the ID of user).

◆ **public String getUser_Name()**

This method doesn't take any parameters and return String (the username of user).

◆ **public String getFirst_Name()**

This method doesn't take any parameters and return String (the first name of user).

◆ **public String getLast_Name()**

This method doesn't take any parameters and return String (the last name of user).

◆ **public String getEmail()**

This method doesn't take any parameters and return String (the email of user).

◆ **public String getPassword()**

This method doesn't take any parameters and return String (the password of user).

◆ **public String getProfilePath()**

This method doesn't take any parameters and return String (the profilepath of user).

◆ **public String getPlan()**

This method doesn't take any parameters and return String (the plan of user which he/she choose it).

◆ **public void setFirst_Name(String first_Name)**

This method takes one parameter from type String (first name of user) and set the first name of user by this

◆ **public void setLast_Name(String last_Name)**

This method takes one parameter from type String (last name of user) and set the last name of user by this parameter.

◆ **public void setEmail(String email)**

This method takes one parameter from type String (email of user) and set the email of user by this parameter.

◆ **public void Set_Password(String password)**

This method takes one parameter from type String (password of user) and set the password of user by this parameter.

◆ **public void setPlan(String plan)**

This method takes one parameter from type String (plan which the user choose it) and set the plan of subscription by this parameter.

◆ **public static boolean Userexist(String user_name)**

This method takes one parameter from type String (the username of user), search about it by this username and return (true if it find this user or false if it doesn't find the

◆ **public void Subscribe(int id , Subscription.Plans plan)**

This method takes two parameter from type (int , Subscription.Plans) (the ID of user , the plan which apply on it) and set the data of the subscription .

◆ **public void Updata_Subscription_Plan(Subscription.Plans plan)**

This method takes one parameter from type Subscription.Plans(the plan which apply on it) and set the new plan of the subscription .

◆ **public static User GetUser(String user_name)**

This method takes one parameter from type String (the username of user) and return the user who has this username.

◆ **public UserWatchRecord getWatchedMovie(Movie movie)**

This method takes one parameter from type Movie class (the movie which user watch it) and return it if it found in the list of watched movies.

◆ **public String toString()**

This method doesn't take any parameters and return String (the all data of user in one String).

Artist

Functions:

◆ **public String getFirst_Name()**

This method doesn't take any parameters and return String (the first name).

◆ **public String getSecond_Name()**

This method doesn't take any parameters and return String (the second name).

◆ **public String getNationality()**

This method doesn't take any parameters and return String (the nationality).

◆ **public String getGender()**

This method doesn't take any parameters and return String (the gender).

◆ **public int getAge()**

This method doesn't take any parameters and return int (the age).

◆ **public ArrayList<String> getMovies()**

This method doesn't take any parameters and return arraylist of String (the names of movies).

◆ **public void setFirst_Name(String first_Name)**

This method takes one parameter from type String (first name) and set the first name by this parameter.

◆ **public void setSecond_Name(String second_Name)**

This method takes one parameter from type String (second name) and set the second name by this parameter.

◆ **public void setNationality(String nationality)**

This method takes one parameter from type String (nationality), set the nationality by this parameter and doesn't return any value.

◆ **public void setGender(String gender)**

This method takes one parameter from type String (gender) , set the gender by this parameter and doesn't return any value.

◆ **public void setAge(int age)**

This method takes one parameter from type String (age) , set the age by this parameter and doesn't return any value.

◆ **public void setMovies(ArrayList<String> movies)**

This method takes one parameter from type arraylist of String (the names of movies) , set the names of movies by this parameter and doesn't return any value.

◆ **public abstract String toString()**

◆ **public abstract void update(int index,String movie)**

◆ **public abstract void add(String first_name,String second_name,int age,String gender,String Nationality,ArrayList<String> movie)**

These methods overridden in two classes(Cast , Director).

Cast

This class inherits the Artist Class

Functions:

◆ **public void update(int index, String movie)**

This method takes two parameters from type (int , String) , set the new movie for the cast.

◆ **public void add(String first_name,String)**

This method takes two parameters from type String , search about cast by his first name and second name and return it if it find this cast(return object of cast).

- ◆ `public void add(Stringfirst_name,String second_name,int age,String gender,String ,Nationality,ArrayList<String> movie)`

This method takes six parameters from type (String ,int ,arraylist of String) and set the data of Cast.

- ◆ `public static Cast search(String first_name,String second_name)`

This method takes two parameters from type String , search about director by his first name and second name and return it if it find this director(return object of director).

- ◆ `public String toString()`

This method doesn't take any parameters and return string(the all data of cast in one String).

Director

This class inherits the Artist Class

Functions:

- ◆ `public void add(Stringfirst_name,String second_name,int age,String gender,String Nationality,ArrayList<String> movie)`

This method takes six parameters from type (String ,int ,arraylist of String) and set the data of Director.

- ◆ `public void update(int index,String movie)`

This method takes two parameters from type (int , String) , set the new movie for the director.

- ◆ `public static Director search(String first_name,String second_name)`

This method takes two parameters from type String , search about director by his first name and second name and return it if it find this director(return object of director).

- ◆ `public String toString()`

This method doesn't take any parameters and return string(the all data of director in one String).

Movie

Functions:

◆ `public int getId()`

This method doesn't take any parameters and return int (the ID of movie).

◆ `public int getViews()`

This method doesn't take any parameters and return int (the number of view of the movie).

◆ `public String getTitle()`

This method doesn't take any parameters and return String (the title of the movie).

◆ `public String getRevenue()`

This method doesn't take any parameters and return String (the revenue of the movie).

◆ `public LocalDate getRelease_date()`

This method doesn't take any parameters and return String (the gunning time of the movie).

◆ `public String getRunning_time()`

This method doesn't take any parameters and return LocalDate (the release date of the movie).

◆ `public ArrayList<String> getGenres()`

This method doesn't take any parameters and return arraylist of String (the genres of the movie).

◆ **public String getLanguage()**

This method doesn't take any parameters and return String (the language of the movie).

◆ **public String getCountry()**

This method doesn't take any parameters and return String (the country of the movie).

◆ **public String getPoster_path()**

This method doesn't take any parameters and return String (the poster path of the movie).

◆ **public String getBudget()**

This method doesn't take any parameters and return String (the budget of the movie).

◆ **public float getImdb_score()**

This method doesn't take any parameters and return float (the Imdb_score of the movie).

◆ **public Double getAverage_rating()**

This method doesn't take any parameters and return object of Double (the average rate of the movie).

◆ **public Director getDirector()**

This method doesn't take any parameters and return object of Director (the director of the movie).

◆ **public void setId(int id)**

This method takes one parameter from type int to set the ID of movie by this parameter and doesn't return any value.

◆ **public void setTitle(String title)**

This method takes one parameter from type String to set the title of movie by this parameter and doesn't return any value.

◆ **public void setRelease_date(LocalDate release_date)**

This method takes one parameter from type localdate to set the releasedate of movie by this parameter and doesn't return any value.

◆ **public void setRunning_time(String running_time)**

This method takes one parameter from type String to set the running time of movie by this parameter and doesn't return any value.

◆ **public void setGenre(ArrayList<String> genres)**

This method takes one parameter(arraylist) from type String to set the genres of movie by this parameter and doesn't return any value.

◆ **public void setLanguage(String language)**

This method takes one parameter from type String to set the language of movie by this parameter and doesn't return any value.

◆ **public void setCountry(String country)**

This method takes one parameter from type String to set the country of movie by this parameter and doesn't return any value.

◆ **public void setPoster_path(String poster_path)**

This method takes one parameter from type String to set the poster path of movie by this parameter and doesn't return any value.

◆ **public void setBudget(String budget)**

This method takes one parameter from type String to set the budget of movie by this parameter and doesn't return any value.

◆ **public void setRevenue(String revenue)**

This method takes one parameter from type String to set the revenue of movie by this parameter and doesn't return any value.

◆ **public void setImdb_score(float imdb_score)**

This method takes one parameter from type float to set the imdb-score of movie by this parameter and doesn't return any value.

◆ **public void setViews(int views)**

This method takes one parameter from type int to set the number of views of movie by this parameter and doesn't return any value.

◆ **public void setAvarage_rating(Double avarage_rating)**

This method takes one parameter from type Double to set the average -Rating of movie by this parameter and doesn't return any value.

◆ **public String getDescription()**

This method doesn't take any parameters and return String (the of description the movie).

◆ **public void
setDescription(String
description)**

This method takes one parameter from type String to set the description of movie by this parameter and doesn't return any value.

◆ **public void inc_views()**

This method doesn't take any parameters, increment the number of views of movies and doesn't return any values.

◆ **public List<Cast>
getCast()**

This method doesn't take any parameters and return list of Cast (the object of cast).

◆ **public int
getUsers_rated()**

This method doesn't take any parameters and return int (the number of the users whose rate the movies).

◆ **public int getYear()**

This method doesn't take any parameters and return int (the year of the movie).

◆ **public Double
getTotalRating()**

This method doesn't take any parameters and return object of Double (the of total of the movie).

◆ **public void**
setCast(List<Cast> cast)

This method takes one parameter(arraylist) from type Cast , set the cast of movie by this parameter and doesn't return any values .

◆ **public void**
setDirector(Director
director)

This method takes one parameter from type object of Director, set the director of movie by this parameter and doesn't return any values .

◆ **public void**
setTotalRating(Double
total_rating)

This method takes one parameter from type object of Double , set the total-rating of movie by this parameter and doesn't return any values .

◆ **public void**
setUsers_rated(int
users_rated)

This method takes one parameter from type int, set the user-rated of movie by this parameter and doesn't return any values.

◆ **public void**
UpdateRating(Double
rating)

This method takes one parameter from type object of Double , update rate of movie by the new rate which enter by the user and doesn't return any values

◆ **public static**
ArrayList<Movie>
TopRatedMovies()

This method doesn't take any parameters , sort the movies according to their rate and return array list.

◆ **public static**
ArrayList<Movie>
MostViewedMovies()

This method doesn't take any parameters , sort the movies according to their number of views and return array list.

◆ **public static void**
getDiffGenres()

This method doesn't take any parameters , set the all genres of the movie and doesn't return any value.

◆ **public ArrayList<String>**
getCastNames()

This method doesn't take any parameters , get the cast names of the movie and return arraylist of String(cast names).

◆ **public void**
setCastNames(ArrayList<String> castNames)

This method takes one parameter(arraylist) from type String , set the cast names of movie by this parameter and doesn't return any values

◆ **public String toString ()**

This method doesn't take any parameters , conceit all data of the movie in one String and return it.

◆ **public boolean
equals(Object obj)**

This method takes one parameter from type Object, override the method equals in comparator interface by check if this object is has the same type of the parameter , check its name is similar to the name of this object and return Boolean.

User Watch Record

Functions:

◆ **public Double getRating()**

This method doesn't take any parameters and return object of Double (the rating of the movie which rated by user after watch this movie).

◆ **public Movie getMovie()**

This method doesn't take any parameters and return object of movie (the movie that user already watch it).

◆ **public Double
setRating(Double rating)**

This method takes one parameter from type object of Double , set the rating of movie which already watched by this parameter and return this rating.

Subscription

Functions:

◆ `public void setPlan(Plans plan)`

This method takes one parameter from type object of Plans , set the plan which user apply on it by this parameter and

◆ `public void setPrice(double price)`

This method takes one parameter from type double , set the price of the subscription by this parameter and doesn't return

◆ `public void setStartDate(LocalDate startDate)`

This method takes one parameter from type LocalDate , set the startdate of the subscription by this parameter and doesn't return any value.

◆ `public void setMoviesWatched(int moviesWatched)`

This method takes one parameter from type int , set the number of movies user can watch it according to the type of plan which apply on it the and set it by this parameter and doesn't return

◆ **public Plans getPlan()**

This method doesn't take any parameters and return object of Plans (the plan which user apply on it when he/she

◆ **public LocalDate
getStartDate()**

This method doesn't take any parameters and return object of LocalDate (the start date of the subscription).a

◆ **public boolean
getUpdatePlan()**

This method doesn't take any parameters and return Boolean (if the subscription is updated or not.

◆ **public void addMovie()**

This method doesn't take any parameters , increment the counter of the user can watch it according his/her plan , check the validate case of subscription and doesn't return

◆ **public void
updateSubscription(Plans
newPlan)**

This method takes one parameter from type Plans , set the plan and price of the subscription by the new plan which user choose it and doesn't return any value.

◆ **public Boolean
isSubscriptionActive()**

This method doesn't take any parameters , check if the time the subscription is terminated or not and return Boolean .

Add Movie Controller

Functions:

- ◆ private void addMovie
(ActionEvent event)
throws IOException

This method takes one parameter from type Action event , take data of movie from admin , set the this data for a new movie , add it to the program and doesn't return any value.

Add Users Controller

Functions:

- ◆ private void Adduser()

This method doesn't take any parameters , take data of user from admin , set the this data for a new user , add it to the program and doesn't return any value .

Choose Plan Controller

Functions:

- ◆ private void
Pay(ActionEvent event)
throws IOException

This method takes one parameter from type Action event , take data of Subscription from user to resubscribe to the program after his/her subscription is terminated , set this data for a new subscription to user , and doesn't return any value.

Forgot Password

Functions:

- ◆ public void
ChangePassword(ActionEvent event) throws
IOException

This method for update on password of user if user forgot it , takes one parameter from type Action event , take data of user from user, set the new data for user and doesn't return any value.

Main Page Controller

Functions:

- ◆ private void onSearch()

This method doesn't take any parameters , take the name of Movie which entered by user in search bar, get the data about the movie what the user search about and doesn't return any value .

Move Edit

Functions:

- ◆ public void
Update(ActionEvent event) throws IOException

This method takes one parameter from type Action event , take the new data of movie from admin , set the this data for a movie and doesn't return any value.

Movie Controller

Functions:

◆ `public void watch Movie ()`

This method doesn't take any parameters , get the data about (the movies which user already watched it and the movies which will watch it later) and doesn't return any value .

Profile Page Controller

Functions:

◆ `public void Select (MouseEvent event) throws IOException`

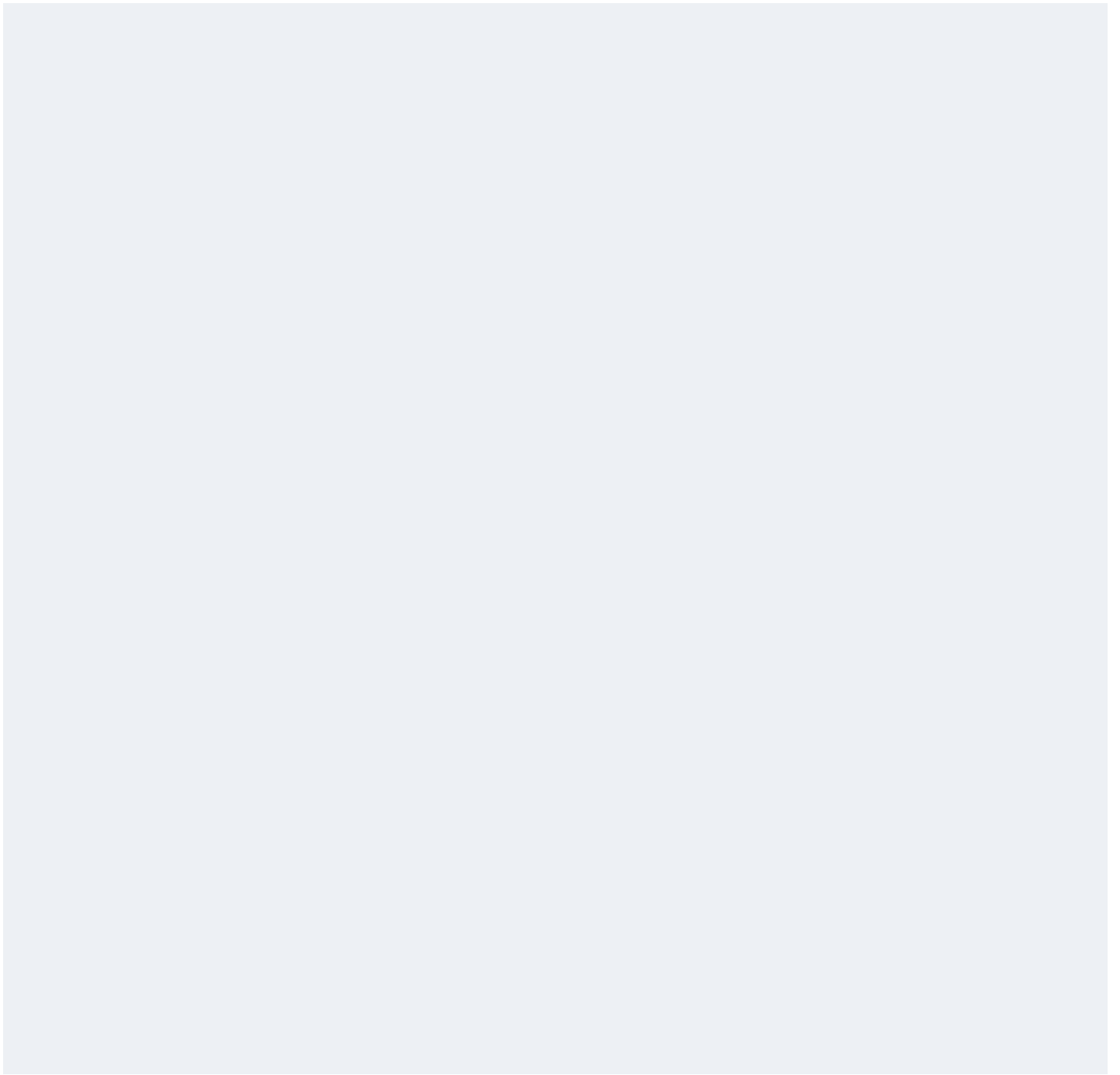
This method takes one parameter from type MouseEvent , select the choice which the user choose it if user choose "Edit profile " then take the new data from user , set the new data for the user , if choose "log out " will exit from his / her account , if choose "delete account" will remove his / her account from program , if choose "delete movies watch to later " will remove all movies the user choose it to watch later and doesn't return any value.

Profile Page Controller

Functions:

◆ `protected void Search (String search name, String type)`

This method takes two parameters from type String , take the first name and last name from user , get the data for a cast/director and doesn't return any value.



Description

The Watch It program is a platform for watching movies that includes various sections of movies, including (horror, comedy, action, and drama).

There are two people who can access the program, the user, and the admin, each of whom has functions to perform.

The admin controls the program from behind the screen, as he is the person who adds new films to the program in terms of his information, including the cast and his director. He can also see all the films that are on the platform and see all the users who use the program. He can also add... A new user can control the types of plans available for subscription by displaying their information, calculating the monthly revenues for subscriptions, and knowing which subscription plan has the largest number of subscriptions.

As for the user, he chooses the movies he wants to watch, as he has a limit on the number of movies, he can watch according to the type of subscription plan to which he applies. He can specify whether he wants to watch the movie immediately or later, and if he watches a movie, he can then He evaluates the film with a rating ranging from 1 to 5 stars. He searches for the film, cast, or even director by his name. He can modify his data, delete his personal account, delete the list of films he has watched, or log out of the program.

UML diagram

