AGH – UNIVERSITY OF SCIENCE AND TECHNOLOGY

Project documentation for

DVD Rental system

Object-oriented programming languages

Electronics and Telecommunication EN, III year
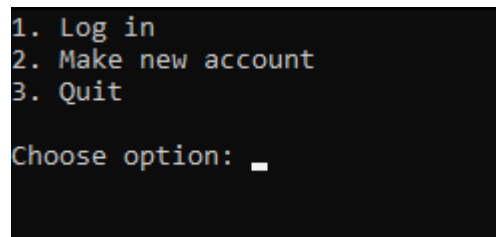
Kacper Godula

lecturer: Rafał Frączek

05.01.2021

1. Project description

The topic of my project is a DVD Rental System. Though DVD rentals are not so popular anymore I tried to make some kind of a platform for both users and admins. With user account you can make an order and manage it, check what DVDs are offered and weather they are available whereas admin account can see more detailed information about all the movies, display all active orders, add new DVDs and change their quantity.

2. User's manual

After running the program we can see the Base Menu where we can either log in to the system, create a new account or quit. We chose certain option by writing an appropriate number and clicking enter.
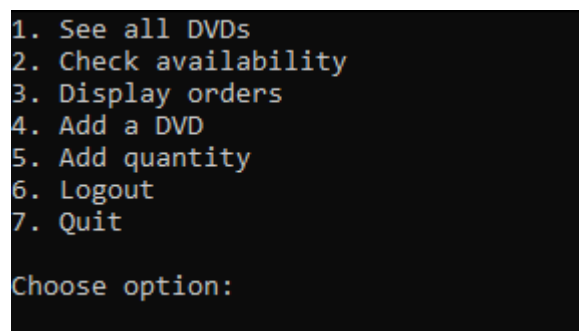

Fig 1. Base Menu

When we log in successfully, we will see the main menu (depending on the type of account we are using, the menus may differ).
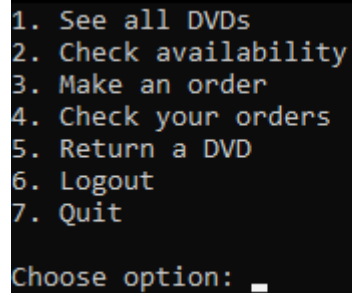

Fig 2. Admin Main Menu

Admin Main menu:
1. See all DVDs – Displays detailed information about all the DVDs
2. Check availability – Lets admin check availability of a certain DVD
3. Display Orders – Displays all active orders
4. Add a DVD – Lets admin add a new DVD to the database
5. Add quantity – Lets admin change the quantity of a certain DVD
6. Logout – Log out from the current account and go back to Base menu

7. Quit – Shuts down the program

Fig 3. User Main Menu

User Main menu:
1. See all DVDs – Displays information about the DVDs
2. Check availability – Lets user check availability of a certain DVD
3. Make an order – Lets user make an order
4. Check your orders – Lets user see their active orders
5. Return a DVD – Lets user return previously ordered DVD
6. Logout – Log out from the current account and go back to Base menu
7. Quit – Shuts down the program

After entering each of the options above admin/user is presented simple instructions explining how to navigate through each of the functionalities.

3. Compilation

In order to run the project it's recommended to either use CodeBlocks with GNU GCC Compiler or to open the Project.exe file directly. The project works probably only on the Windows operating system.

4. Source files

The project consists of the following source files:
- main.cpp – main file with invoking of the Interface class
- headers.h – file with declarations of classes
- dvd_rental.cpp – implementation of DVD_Rental class
- interface.cpp – implementation of Interface class
- sqlite3.h – header file needed for sqlite to operate properly

5. Dependencies

The following external library is used in the project:
> - SQLite – database SQL engine. Website: https://sqlite.org/.

6. Class Description

In the project following classes were created:

- `DVD_Rental` – class containing mechanisms of the DVD Rental system
  - `DVD_Rental()` – Constructor opening the database for further use
  - `void showTitles(bool isAdmin)` - method showing all DVDS (Slightly different for Admin and User)
  - `void orderTitlesShow()` - method for displaying all available DVDs
  - `int determineType(string login_)` – method returning type of currently logged in person
  - `void selectDVD(string title)` - method for displaying info about certain DVD
  - `void checkAvailability(string title)` - method displaying availability info of some certain DVD
  - `int checkLogin(string login_, string psswd_)` - method for validating login
  - `void addDVD(string title_, string description_, string price_, string availability_)` - method adding a new DVD to the database (available only for Admin)
  - `void displayOrders()` - method displaying orders (available only for Admin)
  - `void makeOrder(string title_, string login_)` - method for making orders (available only for User)
  - `void updateQuantity(string title_, string quantity_)` - method for "increasing availability" of a DVD
  - `void checkOrders(string buyer_)` - method displaying orders of a certain user
  - `int anyOrders(string login_)` - method checking if any orders were made
  - `void returnDVD(string title_)` - method for returning a DVD

- o `void makeAccount(string login_, string password_, string type_)` – method for creating a new account
- o `int checkIfLoginExists(string login_)` – method for checking weather account with given login name exists
- o `string getCurrentLog()` - getter for currently_logged

- Interface – class responsible for the communication with user
  - o `Interface()` - constructor of class Interface responsible for overall program behavior
  - o `void loginMenu()` - Base Menu with availability to log in or quit
  - o `void login()` - Logging in menu
  - o `void userMainMenu()` - Main menu for User type of account
  - o `void adminMainMenu()` - Main menu for Admin type of account
  - o `void moviesMenu()` - Menu with list of all DVDS
  - o `void checkAvailability()` - Menu with possibility to check availability of a certain DVD
  - o `void wrongInput()` - Method determining weather input is correct or not
  - o `void addDVDMenu()` - Menu allowing to add a DVD
  - o `void makeOrder()` - Menu allowing to order a DVD
  - o `void displayOrders()` - Menu displaying orders
  - o `void checkOrders()` - Menu displaying orders for a certain user
  - o `void updateQuantity()` - Menu for updating the quantity of a DVD
  - o `void returnDVD()` - Menu allowing to return a DVD
  - o `void signIn()` – Menu allowing to make an account

7. Resources

In the project the following resources are used:
- `database.db` – SQL database with following tables:
  - o `DVDS` – table containing information about each DVD
    - `ID` – unique ID of a DVD
    - `TITLE` – title of a DVD
    - `DESCRIPTION` – descrption of a DVD
    - `PRICE` – price of a DVD
    - `AVAILABILITY` – availability of a DVD

- o `ORDERS` – table containing orders made
  - `ORDER_ID` – unique ID of an order
  - `TITLE` – title of ordered DVD
  - `BUYER` – user that mad an order
- o `USERS` – table containing information about users
  - `USER_ID` – unique ID of account
  - `NAME` – name of account
  - `TYPE` – type of account (1 – User, 2 – Admin)
  - `PASSWORD` – password to a certain account

8. Future development

Probably first, the most important feature that could be adding more verification steps to the signing in procedure (for example to prevent SQL injection).

As the next upgrades, some new features for Admin control coulb be added. Since the structure of the program is rather simple, it's very easy to add new functionality.

9. Other

Some of the predefined accounts:

- Login: `test_admin` Password: `admin_psswd`
- Login: `test_user` Password: `user_psswd`