

Problem statement:

Write an application for movie rental.

The application will store:

- movies: <id>, <title>, <description>, <type>, etc
- clients: <id>, <name>, <CNP>, etc

Create an application which allows to:

- manage the list of movies and clients.
- add, remove, update and list movies and clients
- search for a movie; search for a client.
- Rent/return movies
- Reporting. The reporting part of the application will allow generating a list of clients or

books (maybe of certain type) ordered based on the user preference. Examples: most rented movies, most active clients, clients with rented movies ordered alphabetically, by number of rents, by year of apparition

Movie rental - Feature list

F1. Add client
F2. Remove client
F3. Update client
F4. List clients
F5. Add movie
F6. Remove movie
F7. Update movie
F8. List movies
F9. Search movie
F10. Search client
F11. Rent/return movie
F12. Reporting/statistics Examples: most rented movies, most active clients, clients with rented movies ordered alphabetically, by number of rents, by year of apparition

Iteration plan

Iteration	Planned features
I1	F1, F2, F3, F4, F5, F6, F7, F8

I2	F9, F10, F11
I3	F12

Iteration 1

Running scenario

User	Program	Description
	The list of option	
1		Add a new client command
	Client id:	Give and id
3		Client id
	Client name:	Give client name
Daniel Newman		The client name
	Client cnp	Give the client cnp
19352056631		The client cnp
	Client added	
4		List all clients command
	ID Name CNP 3 Daniel Newman 19352056631	All clients
3		Update clients command
	Client id:	Give and the client id that you want to update
3		Client id to update
	Client name:	Update client name
Claudiu Newman		The u pdateclient name
	Client cnp:	Update the client cnp
19352056631		
4		List all clients commnd

	ID Name CNP 3 Claudiu Newman 19352056631	All clients
2		Remove client command
	Client id:	Client id to delete
3		The client id from the user
	Client deleted	Message
4		List all clients command
	Empty repository	All Clients
0		Quit program command

Work items/tasks

T1	Filter the input from command line and get the specific command
T2	Implement user interface.
T3	Add a client
T4	Remove client
T5	Update client
T6	List clients
T7	Add a movie
T8	Remove movie
T9	Update movie
T10	List movies

Iteration 2

Running scenario

User	Program	Description
	The list of option	
1		Add a new client command

	Client id:	Give and id
3		Client id
	Client name:	Give client name
Daniel Newman		The client name
	Client cnp	Give the client cnp
19352056631		The client cnp
	Client added	
4		List all clients command
	ID Name CNP 3 Daniel Newman 19352056631	All clients
3		Update clients command
	Client id:	Give and the client id that you want to update
3		Client id to update
	Client name:	Update client name
Claudiu Newman		The u pdateclient name
	Client cnp:	Update the client cnp
19352056631		
4		List all clients commnd
	ID Name CNP 3 Claudiu Newman 19352056631	All clients
2		Remove client command
	Client id:	Client id to delete
3		The client id from the user
	Client deleted	Message
4		List all clients command
	Empty repository	All Clients
0		Quit program command

Work items/tasks

T11	Search client
T12	Search movie
T13	Rent movie
T14	Return movie

Iteration 3

Running scenario

User	Program	Description
	The list of option	
4		List all clients command
	ID Name CNP 3 Daniel Newman 19352056631	All clients
3		Update clients command
	Client id:	Give and the client id that you want to update
3		Client id to update
	Client name:	Update client name
Claudiu Newman		The u pdateclient name
	Client cnp:	Update the client cnp
19352056631		
4		List all clients commnd
	ID Name CNP 3 Claudiu Newman 19352056631	All clients
2		Remove client command
	Client id:	Client id to delete
3		The client id from the user
	Client deleted	Message

4		List all clients command
	Empty repository	All Clients
0		Quit program command

Work items/tasks

T15	Get the most rented movies
T16	Get the most active clients