**Cancel reservation:**

* Cancel reservation for a regular user:

This use case starts when a regular user selects the option of looking for a reservation and inputs the reservation number, when all the information is displayed the user selects the cancel a reservation option. The system will check if the movie starts within the next 72 hours. If it does, it displays a message saying cancelation is unavailable for this reservation and use case terminates. The system will display a message informing the user of the cancelation policy and wait for the user to accept it. If the user accepts, the system labels the seats in that reservation as available. Also, the system will send an email, to the email the user used to make a reservation, with the details of the cancelation and a coupon code with the refunded amount.

Candidate objects for regular user:

* Movie-start
* **Display-message**
* Cancelation-unavailable
* **Reservation**
* Cancelation-policy
* **User**
* **Seats-label**
* **Send-email**
* Coupon-code
* Refunded-amount
* Cancel reservation for a registered user:

This use case starts when a registered user logs into their account, checks the reservation that is going to be cancelled and selects the cancel reservation option. The system will check if the movie starts within the next 72 hours. If it does, it displays a message saying cancelation is unavailable for this reservation and use case terminates. The system will display a message asking the user to confirm the cancelation and wait for the user to accept it. If the user accepts, the system labels the seats in that reservation as available. Also, the system adds the amount refunded to the user's account.

Candidate objects for registered user:

* **Login-account**
* Check-reservation
* **Cancel-reservation**
* Movie-start
* **Display-message**
* Cancelation-unavailable
* **Confirm-cancelation**
* User-accept
* **Seats-label**
* Refunded-amount
* **User-account**