Vreg113 Vadim Reger

Design Report for Assignment 2

Let’s start by specifying what I have done in the main part of the assignment. I started this assignment by making changes to my domain model, which I have completed in assignment 1. I realized that it would be better if I make a watchlist inside the User class, it will make it more secure and easily accessible when the user is authenticated. Also, I made such that Review class now takes additional parameter timestamp, since when function read reviews from the CSV file, we can easily assign timestamp to the Review object.

After that I started working on the memory repository, I wanted the repository to contain us much information about movies as it has in 1000 movies CSV file. So, I added movie ranks, revenues and made such that every actor has a list of all colleagues he worked with, this will help to further develop the Web Application. I decided to make movie rank my uniquely identified parameter for movie, since all the movies in the CSV file have unique movie rank from 1 to 1000.

When I was working on the services layer. I realized that I need to check (when the user is logged in) whether the user added movie in the watchlist, and change “add to watchlist” button to “remove from watchlist” button. So, I converted every movie into dictionary with all the parameters and added parameter “added_to_watchlist” by default it was false. And on every page, I was checking whether the user’s watchlist contains movies that currently being displayed on the page.

For the search functionality, I decided to complete it using input string. Firstly, on the home page I added a sidebar that offers to choose four different search properties to the user. Search by Title, Actor, Genre and Director, all these buttons redirect user to the prompt page that contain WTForm text field that asks the user to enter the string. When user enters the string, web application generates a webpage that contains 30 or less movies (depending on the search result) that were found by the category and the string user entered. Also, at the bottom of this page user can go to the next and previous page that will contain next 30 movies (if found).

Finally, for the new feature of my web application I decided to make a watchlist for authorized users. So, when the user is not in session watchlist button will be disabled (will not be seen) on the navbar. Since in my design I always have movies displayed with the button add to watchlist I needed to check whether user is logged in or not, and also if the user is logged in, web application will need to check whether every movie displayed on the webpage is in the user's watchlist. So, I added this functionality to every web page that contains movies. Then I decided to make a web page that will show all the movies that user added to the watchlist. On this web page user can click on the title of the movie and see more information about movie he chose, and user can remove this movie from the watchlist, and page will be updated.

I also include my journal in the design report since it states the difficulties I've challenged and also some of my solutions to the problems.

⊛ Genres ✓✓ ⊛ Rating ✓✓

⊛ Description ✓✓ ⊛ Revenue (Millions) ✓✓

⊛ Director ✓✓

⊛ Actors ✓✓

- Movie domain model - movie repo

~~???~~ For now collegue list is incomplete.
(I'm not sure I will need it later)

- user ✓

⊛ user_name ✓✓ ~~* reviews~~ ✓ ?

⊛ password ✓✓ ~~* time spend (min)~~ ✓ ?

~~* watched movies~~ ✓ ?

! Later
memory rep.

- user domain model - user repository.

end day 1 (finish repository, add collegues fiture).

day 2

① finish what haven't done in day 1.

✓ ⊛ colleagues feature.

movie $[\underline{a1}, a2, \ldots, a_n]$

- review

⊛ movie (rank) ✓✓

⊛ review_text ✓✓

⊛ rating ✓✓

⊛ time stamp ✓✓

- review domain model

- review repository

Note: Watch list is not completed in the repository.

② Finished with the repository

③ writing some tests.

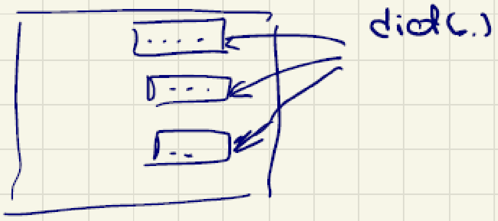
Note: Add more tests later

④ Started creating HTML.

day 3

① Started working on authentication

② Working on HTML and Services.



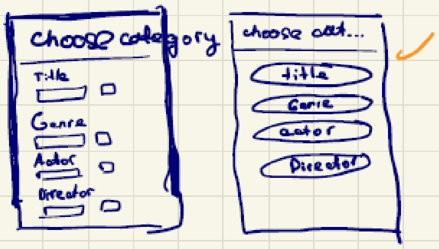
next day:

- after success login in not goes to home page.

Rank	1	2	3	4	5
Index	0	1	2	3	4

day 4

① side bar



! - change timestamp in Review
(doest work propely)

② movie_html and movie.py.
- creating new
blueprints.

<u>next day</u>:
- fix dates on reviews
- make watch list webpage.
- make search

day 5

① Started working on watchlist webpage.

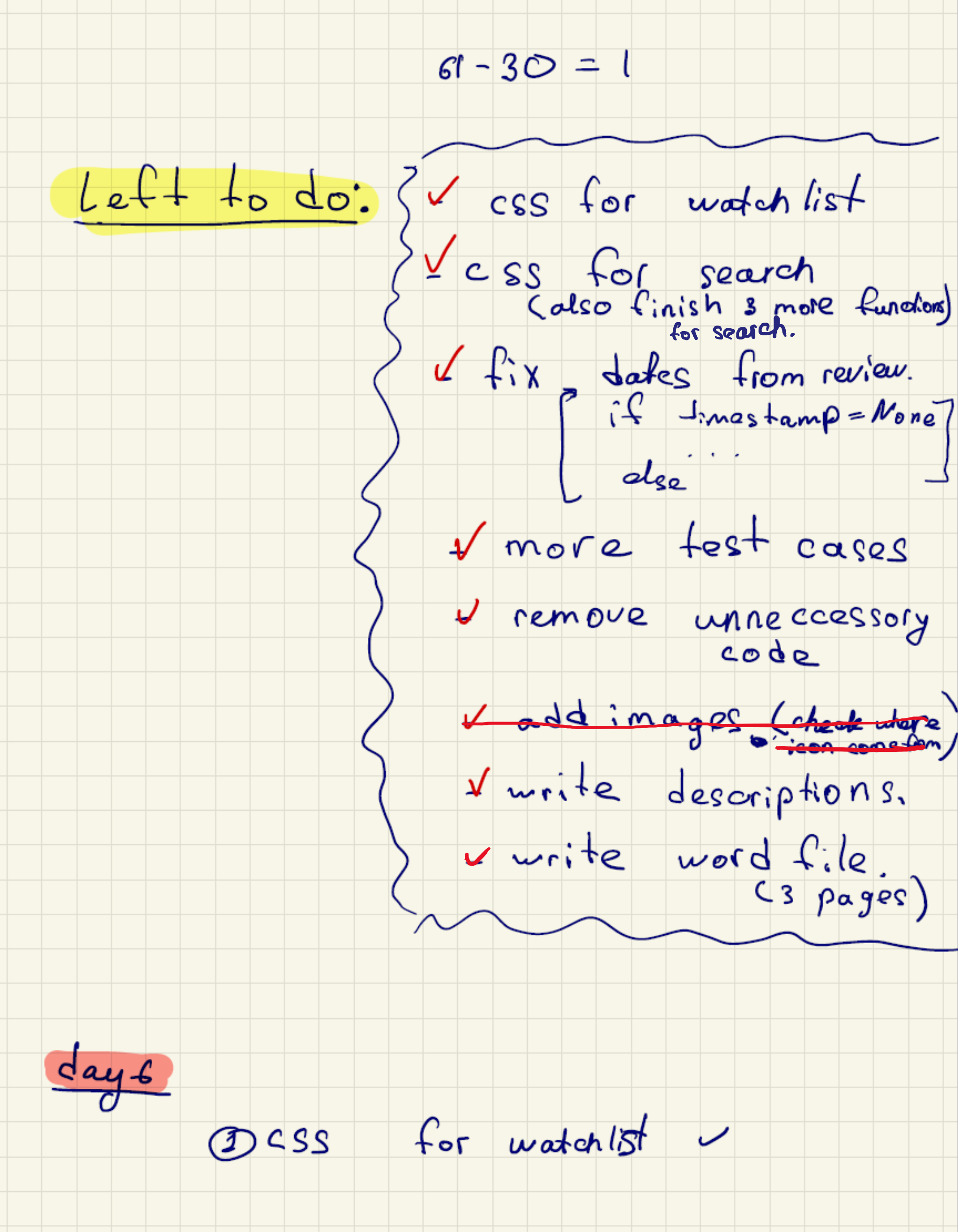
② started working on search webpage

len(list) = 61 - 3 pages

curr_page = 2

(61) - (2*30) = 1

1. 1 - 30
2. 31 - 60
3. 61 - 90



61 - 30 = 1

Left to do:

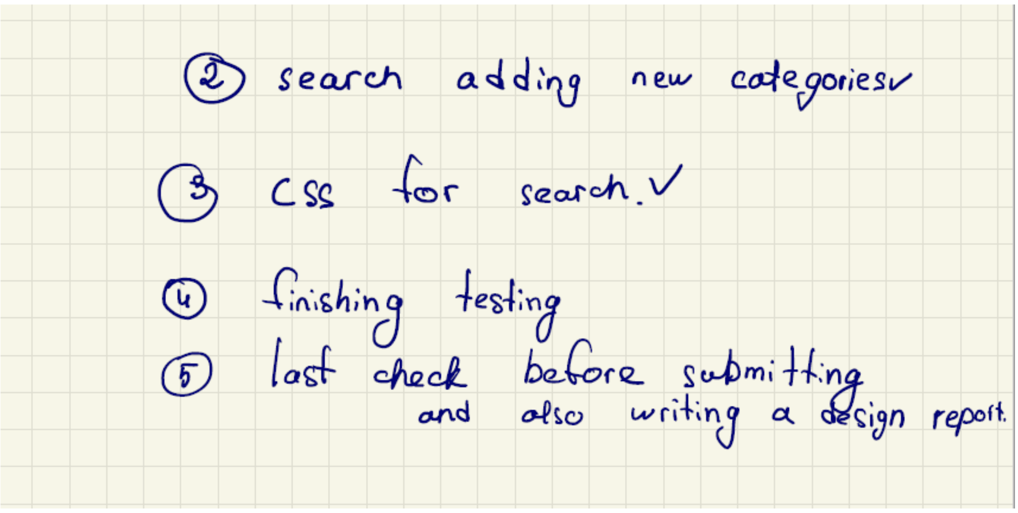
- ✓ css for watchlist
- ✓ css for search (also finish 3 more functions) for search.
- ✓ fix dates from review.

```
if timestamp = None
  . . .
else
```

- ✓ more test cases
- ✓ remove unneccessory code
- ✓ ~~add images (check where~~ • ~~icon come from~~)
- ✓ write descriptions.
- ✓ write word file. (3 pages)

day 6

① css for watchlist ✓



② search adding new categories ✓

③ css for search. ✓

④ finishing testing

⑤ last check before submitting and also writing a design report.