# Create a HackerNews Search web applica on. (*12 points*)

○ WebApp should have 2 pages - Home & Se ngs (refer to a ached screens) ○ In the Home page,

➤ List content using HackerNews API ([*h*](https://hn.algolia.com/api)[*ps://hn.algolia.com/api*](https://hn.algolia.com/api))

➤ Show a search bar on top to search & get news by tle, url or author

➤ Show search filters (refer a ached screenshot)

➤ Show the count of total results received from API

➤ Each item/news is clickable & should open in a new tab with proper url

➤ Show the "Se ngs" link/bu on on top and that should take us to the se ngs page

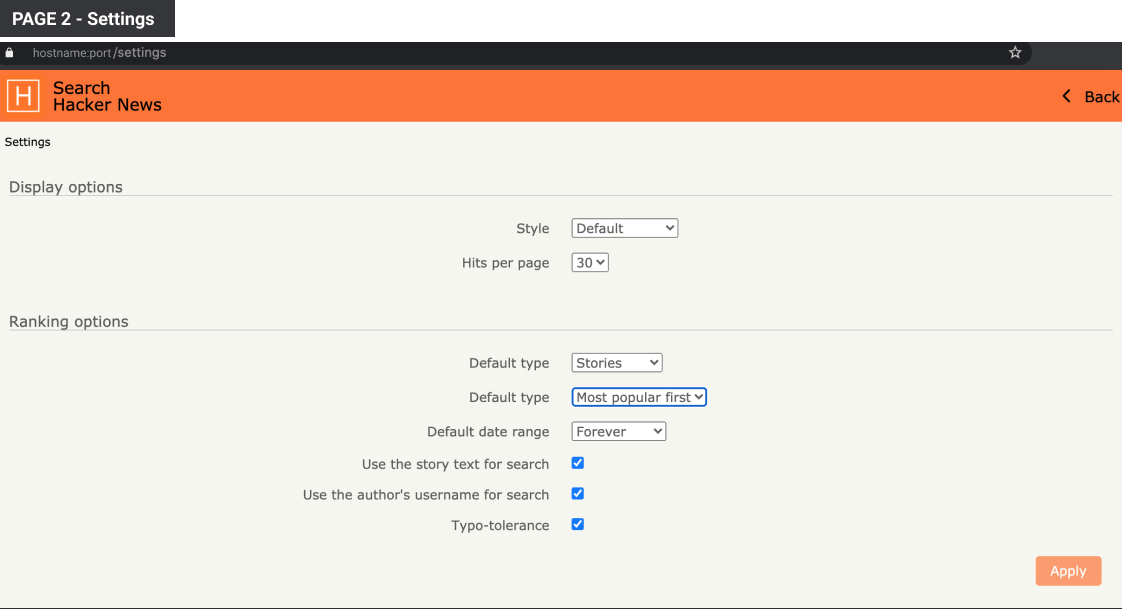
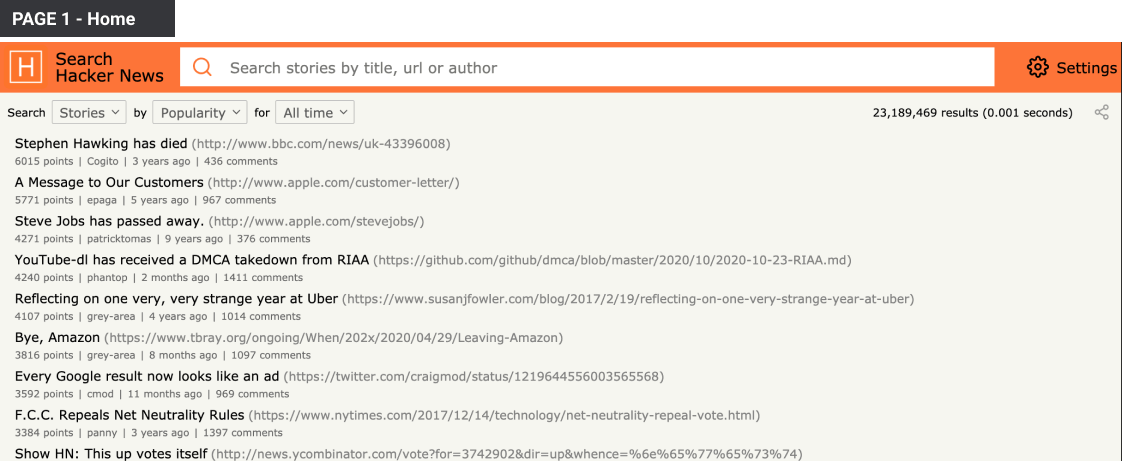
○ In the Se ngs page,

➤ List op ons to modify the default search filters and save the state when clicked on the "Apply Changes" bu on

➤ Show "Go Back" bu on/link to visit the home page

➤ Save the state or new op ons in the localstorage and apply them on the home page

➤ Persist the saved filter op ons even when the page is manually reloaded



# Create a Movies Lis ng applica on. (*20 points*)

○ Pages - Lis ng & Movie Page & Search Page (refer to a ached screens) ○ In the Home/Lis ng/Search page,

➤ List content using TMDB API (*https://developers.themoviedb.org/3/ge ng-started/introduc* [*on*](https://developers.themoviedb.org/3/getting-started/introduction))

➤ UI/design should look similar to the screens a ached below

➤ Refer h ps://www.themoviedb.org/ for the func onali es

➤ Include basic func onali es like lis ng movies from Popular & Trending categories

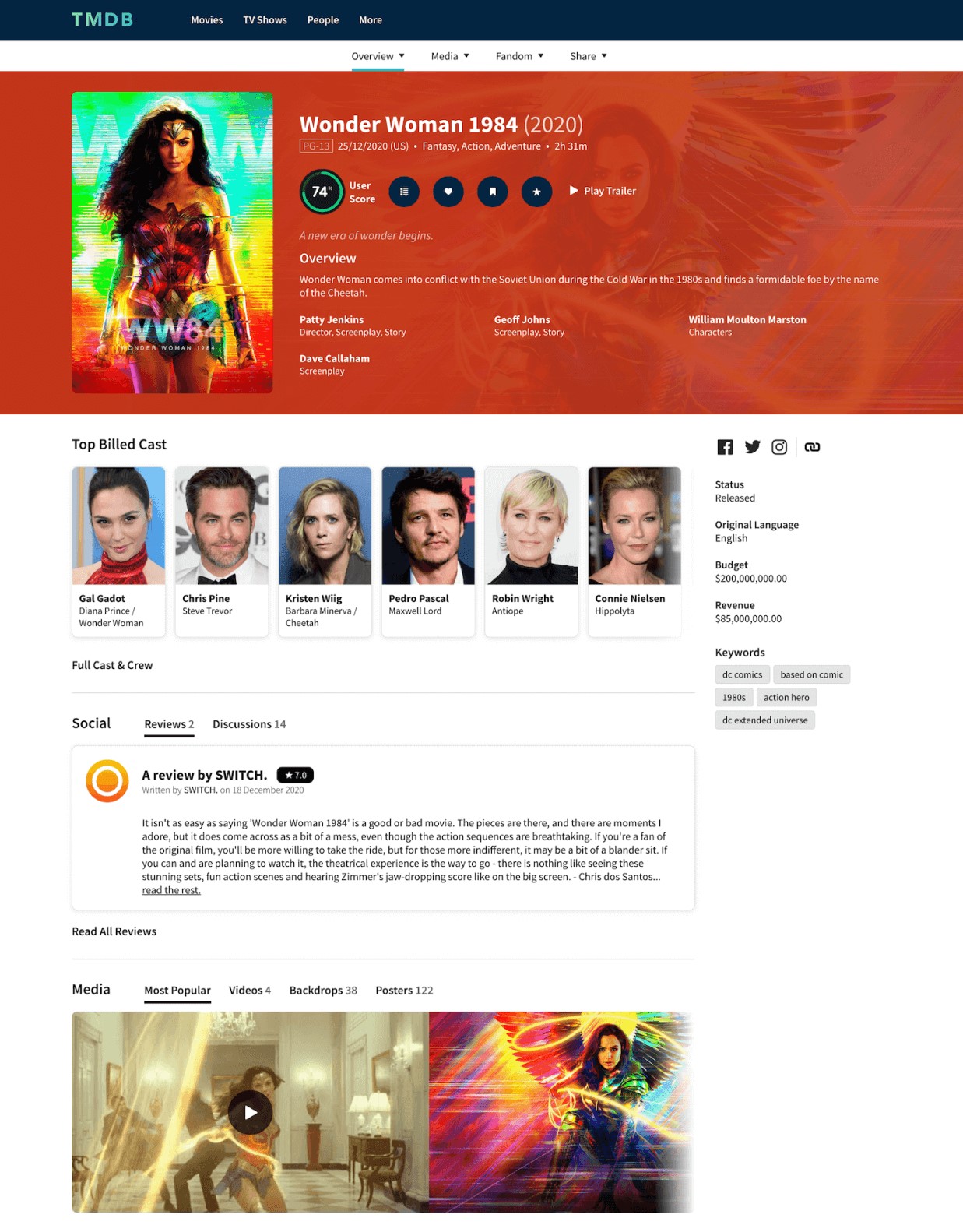
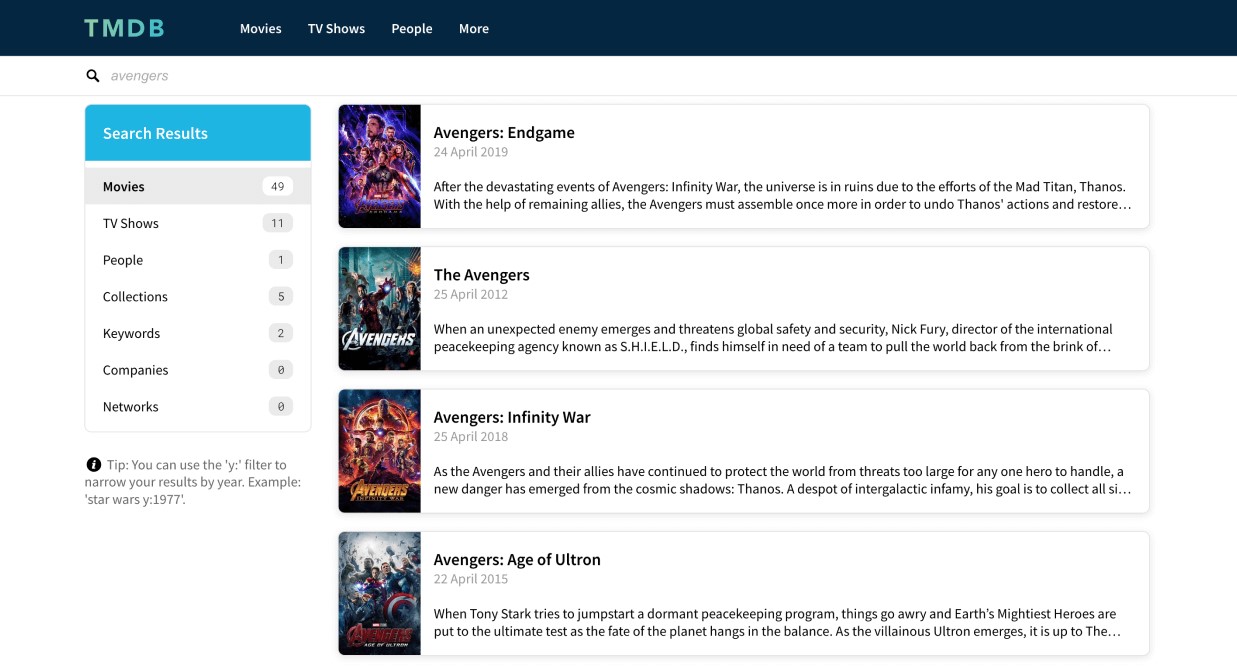
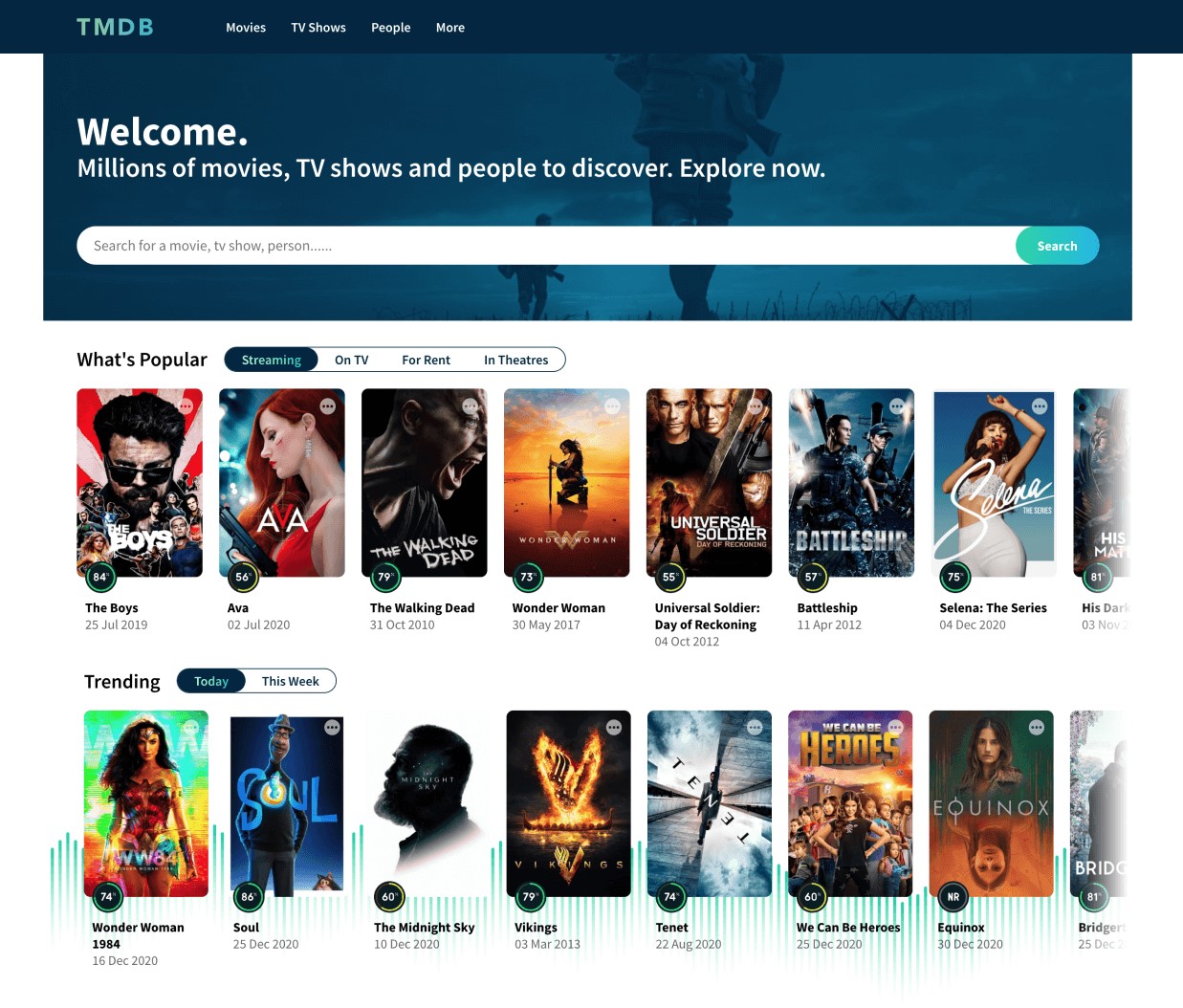
➤ Show movie's detail page on clicking on any movie card/ le

○ In Movie Page,

➤ UI should look similar to the a ached movie details screen

➤ Show basic info about the movie, banners, ra ngs, casts, reviews

➤ Refer h ps://www.themoviedb.org/ for the func onali es and include basics



# Design an Invoice UI using HTML & CSS. (*3 points*)

○ Take any invoice ui design of your choice and develop the design using plain html & css ○ You can use from dribbble.com, behance.net, pinterest.com or similar sites for ref

○ *Bonus*: +2 points if your invoice design inspires us!

# Create a webpage from your Resume. (*5 points*)

○ Use HTML & CSS or React to create a page and place the same content from your resume.

○ You can highlight the content in your own way and showcase your skills ○ Note: Do not use resume templates from online.

○ *Bonus,*

➤ +1 point if your design inspires us!

➤ +2 points if your page looks well in both small & large screens (mobile/laptop)

# Time to solve some problems. (5 ques ons - *1 point each*)

○ Create a method/func on to check whether the given 2 strings are anagram to one another or not. Return true if they are anagram.

**Answer:**

def areAnagram(a,b):

n1 = len(a)

n2 = len(b)

a = input()

b = input()

if n1 != n2:

return 0

a = sorted(a)

b = sorted(b)

for i in range(0, n1):

if a[i] != b[i]:

return False

return True

if areAnagram(a,b):

print("True")

else:\par

print("False")

○ Return the sum of the prime numbers between given range (including given numbers).

Eg., *Input*: 3 & 60. Output should print the sum of all the primes lying between 3 & 60 ○ Return the total number of days from the date given to ll now. Eg., *Input:* 24-01-1995 & Output should return the total days ○ Create a simple encryp on program.

* Eg., *Input:* Hi awesome! *Output:* H#9\_a#23#5#19#15#13#5!
* Replace each character with the correct alphabe cal order number except the first character of each word. Eg., Hi -> H#9 (add # before every number) - Replace spaces with \_

○ Convert array to object based on data type. - Eg., *Input:* ["test", "name", 45, "anything", true, 78] - Eg., *Output:*

{ "strings": ["test", "name", "anything"], "numbers": [45, 78], "booleans": [true] }