

Frontend Tasks

Rules

1. Please take the time to fully understand the problem statement described below. The website you created is expected to fulfill all the statements. Do not hesitate to contact us if you have any questions regarding the test cases.
2. Your code will be evaluated for both functional (whether the website behaves as expected) and non-functional (code quality, readability, code smells, vulnerability, etc.).
3. We are really interested in a clean and maintainable codebase, so please solve the problem by keeping this in mind. It is highly recommended that you use a well-established design pattern.
4. You are free to choose your own storage system to store the data, be it Local Storage, Cookies, IndexedDB, etc.
5. Unit tests have to be included in the codes. Please aim for complete coverage of the business logic.
6. Core Web Vitals assessment by [PageSpeed Insights](#) should be **Passed**
7. Write the answer for this problem on your own Github repository and make it public. Send the link to us at the latest seven days after you receive this.

Problem Statement

DelosNews is asking you to help them create a website for a premium article shopping system that would allow users to buy the available articles from [NYTimes Most Popular | Dev Portal](#). The system by minimum should consist of 4 pages:

1. Article list page that shows all articles that users can buy and users can filter it by these criteria:
 - a. Most Emailed,
 - b. Most Shared,
 - c. Most Viewed
2. A detail page that consists of information about the Article, should minimum consist of Cover Image, Title, Author, Abstract, and Publish Date.
3. User bought articles page, consists of all articles that the user has bought.
4. Get lucky coin page, for every minimum purchase of 50.000 coins, the user gets 3 tickets to redeem to a lucky draw, with a random reward, minimum consist of:
 - a. 50.000 coins, maximum only once
 - b. 20.000 coins
 - c. Try Again 🙄
 - d. Please use your creativity to add more rewards 😊

The following use cases should be considered when creating the website:

- Users have starting coin of 100.000
- The price of each article is:
 - For articles published more than 7 days ago, is free. But the maximum bought articles should be no more than 5 articles
 - For articles published in period less than or equal to 7 days ago, is 20.000 coins
 - For articles published in a period less than or equal 1 day ago, is 50.000 coins
- Users cannot buy the same book
- The list pages should be able to handle a search function.
- The list pages should have pagination, or you can also use infinite scroll for the pagination.
- The detail page should have an action for users to buy the book using their balance
- Users can open the link to the article after buying it (even though the user will be asked by NYTimes to subscribe)

Instructions

Please create a website written in javascript or typescript using React and NextJS that is able to answer the problem statement. Please also include the instruction to run the website in the `README.md` in your repository root.

This instruction should cover:

1. How to install the website on local machines, e.g. `npm run install`
2. How to run the website, e.g. `npm run dev`
3. Addresses of the pages, e.g. `localhost:8080/articles`
4. Any additional information that you think we should know before running the website
5. Please provide hosted project repository, e.g. GitHub, GitLab, or Bitbucket, and deployed web using free platforms, e.g. vercel, GitHub Pages, etc

As previously mentioned, we will not only test the functional behavior of the app but also things like the code cleanliness, code structure, SoC Paradigm, how easy it is to run the website, etc. Keep in mind that there are multiple approaches to solve this problem and that you are free to create your own solution.

Good luck with the tests, and hope we can see you again soon! 🙌