

Спецификација за втор проект

Brainster Library

Проект: Brainster Library | Full Stack Academy

За потребите на проектот **Brainster Library** треба да се изработи веб решение за менаџирање на онлајн библиотека.

ОПИС НА ПРОЕКТОТ

Крајното решение на овој проект треба да понуди онлајн платформа каде што корисниците ќе можат да оставаат приватни забелешки и јавни коментари за книгите што ги читаат и да можат во секој момент да се навратат на истите. Во продолжение следува опис на секоја од функционалностите што треба да ги содржи проектот како и табела за бодување на функционалностите.

Автентикација

На проектот се разликуваат 2 типа на корисници: администратори и клиенти (крајни корисници). Достапните опции на платформата се менуваат во зависност од тоа каков тип на корисник е најавен. Следуваат описи за секој тип корисник.

1. Администратор

Улогата на администраторот е да менаџира со сите достапни книги, категориите во кои припаѓаат, авторите како и јавните коментари што ги оставаат крајните корисници. Кога ќе се најави администратор, истиот треба да има пристап до следните менија:

- CRUD за категории. Секоја книга ќе припаѓа на одредена категорија. При додавањето на книга ќе треба да се избере една од понудените категории. Овој CRUD треба да му овозможи на администраторот да ги креира категориите кои ќе се понудат при креирање на книга. За секоја категорија знаеме само назив (title) кој што е задолжителен. Категориите треба да може да се бришат и едитираат. Бришењето да се изведе со т.н. Soft-delete¹ техника.

¹ Soft-delete: записот од база не се брише туку се маркира како избришан со помош на дополнителна колона (пр. deleted_at, is_deleted, is_archived итн.)

- CRUD за автори. Секоја книга ќе има 1 автор. При додавањето на книга ќе треба да се избере еден од постоечките автори. Овој CRUD треба да му овозможи на администраторот да ги додаде авторите кои ќе се понудат при додавање на книга. За секој автор знаеме име (required), презиме (required) и кратка биографија (required, minimum 20 chars). Авторите може да се бришат и да се едита нивното инфо. Бришењето да се изведе со т.н. Soft-delete техника.
- CRUD за книги. Од тука администраторот ќе може да додава нови книги, да едита и да брише постоечки книги. За додавање на нова книга, потребни полиња се: наслов (required), автор (required, existing_author_id), година на издавање (required, number), број на страници (required, number), слика (required, url), автор (required, foreign key) и категорија (required, foreign key). Сликата не треба да се прикачува туку да биде урл до слика од интернет. Една книга има еден автор и припаѓа на една категорија. Во заграда до секое поле се дадени потребните валидации. Ако администраторот избере да избрише книга треба да му се појави поруп (користете sweetalert библиотека), во кој ќе пишува дека со бришење на книгата се бришат сите коментари и забелешки на корисниците за истата. Доколку потврди, се брише книгата заедно со сите коментари и забелешки. Доколку ја откаже операцијата за бришење, не се случува ништо.

Покрај овие CRUD-ови, администраторот треба да има пристап до страница каде што ќе може да ги уредува сите оставени коментари за книгите. Крајните корисници ќе можат да оставаат коментари на секоја книга. Поради тоа што коментарите се јавни, администраторот ќе мора да ги одобри пред истите да станат видливи во каталогот со книги.

Администраторот треба да има и пристап до сите коментари што ги има одбиени и да може истите да ги одобри. Ова може да се направи со посебно мени - Одбиени коментари или во постоечкото мени со коментари да се филтрираат само одбиените со опција за одобрување.

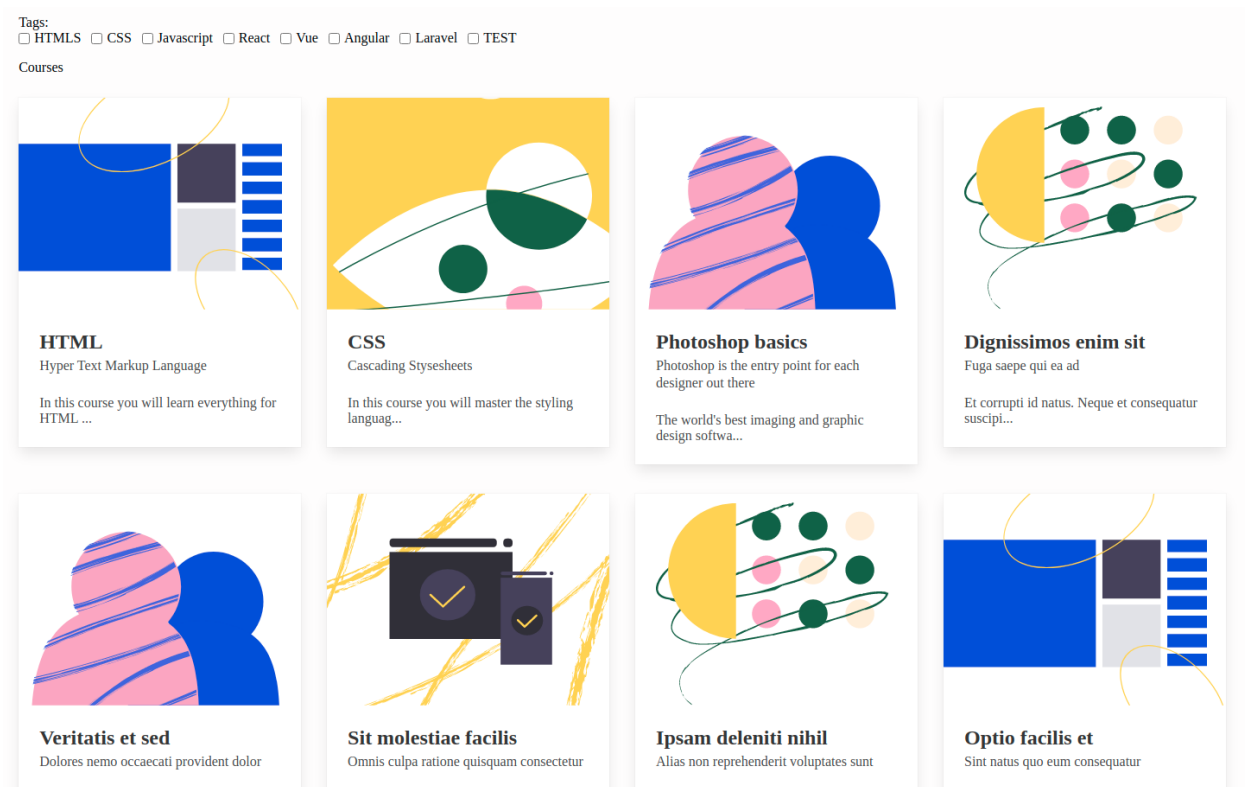
Сумарно, админот треба да има опција да менаџира со сите ресурси кои го сочинуваат ова софтверско решение и тоа: управување со категории, управување со автори, управување со книги и управување со коментари.

Секој посетител на овој проект треба да може да си регистрира свој профил. Сите креирани профил се кориснички и никој не може да се регистрира како администратор. Даден корисник може да стане администратор ако администраторот со бази на податоци директно му додаде таква рола во база на податоци.

2. Краен корисник

Кога корисник за првпат ќе ја посети почетната страница на овој проект треба да му се прикаже каталог со сите книги што постојат во база. Може да ја пристапи било кој, не е потребна претходна автентикација. Книгите треба да бидат прикажани во формат на картички каде што ќе има слика од книгата, наслов, автор и категорија. Покрај каталогот треба да се наоѓа опција за филтрирање на книгите по категорија. Корисникот треба да може да избере повеќе категории при што му се покажуваат книгите во селектираните категории. На пример, ако избере категорија 'Технологија' и категорија 'Менаџмент', му се прикажуваат сите книги што се во категорија технологија и сите книги во категорија менаџмент.

Пример за тоа како **може** да изгледа каталогот заедно со филтрите е прикажан на следната слика.



Слика: Пример за каталогот со книги

Ако кликне на одредена книга, корисникот се носи на нова страница каде што се прикажани сите детали за избраната книга: наслов, автор, година на издавање, број на страници, слика и категорија. Покрај тоа, треба да се прикажани и сите јавни коментари што постојат за таа книга (ако постојат) и кои се одобрени од администраторот.

Внимание: Податоците за книгата, името на авторот и името на категоријата од база да се извадат со едно (JOIN) query наместо со 3 посебни.

Автентизиран корисник

Ако корисникот е најавен тогаш треба да има опција и самиот да остави коментар (ако веќе нема оставено) или да го избрише постоечкиот (ако веќе има оставено). Еден корисник не може да остави 2 коментари и не може да едитуира постоечки коментари. Кога даден корисник ќе остави коментар за одредена книга, истиот не е видлив за другите корисници се дури администратор не го одобри. Сепак, корисникот што го оставил коментарот може да го избрише и пред да му биде одобрен.

Покрај коментарите што се јавни, корисникот треба да може да остави и приватни забелешки за секоја книга. Идејата зад оваа функционалност е како што ја чита книгата, да си прифаќа цитати или идеи и да може во секој момент да им се наврати. Забелешките што за дадена книга ги оставил еден корисник се приватни, не може да ги види никој друг освен тој што е логиран (дури ниту администраторот). Корисникот може да остави повеќе забелешки за една книга и може да брише и едитуира постоечки забелешки.

Екстра поени: Сите горенаведени функционалности носат вкупно 100 поени доколку се правилно имплементирани согласно дадените насоки. Со овој проект можат да се освојат и дополнителни 20 поени со имплементирање на следните функционалности:

1. Приватните забелешки што ги оставаат корисниците за книгите да бидат целосно имплементирани со AJAX технологија. Тоа значи дека со секоа оставање на нова забелешка не се рефрешира целата страница, туку се испраќа AJAX повик за таа да се запише во база и истата се додава во листата на постоечки забелешки. Исто во делот за едитирање и бришење.
2. Да се додаде футер на целата веб платформа, каде што на секое вчитавање ќе се појавува различен цитат. Цитатите се земаат од постоечки апи, достапно на следниот линк:

<http://api.quotable.io/random>

Совет: За поедноставно комплетирање на front-end делот препорачано е користење на Bootstrap, но не е задолжително. Користење на разни CSS работни рамки (frameworks) се охрабрува, иако не се бодува. Наш совет е целиот проект да се изработи во Tailwind.

Систем на бодување

| Функционалност | Поени |
|---|------------------|
| Општи | |
| Role-based автентикација | 10 |
| Backend валидација на инпут полиња | 10 |
| Поставување на проектот на GIT и работи со гранки | 10 |
| Функционалности во Кориснички Панел | |
| Каталог со можност за филтрирање по категорија | 10 |
| Приказ на сите одобрени коментари кога ќе се отвори книга | 5 |
| Приказ на сите забелешки од најавениот корисник за отворената книга | 5 |
| Додавање и бришење коментар на книга за најавени корисници | 5 |
| Додавање, бришење и едитирање забелешки за одредена книга за најавен корисник | 10 |
| Функционалности во Админ Панел | |
| CRUD за категории | 5 |
| CRUD за автори | 5 |
| CRUD за книги | 10 |
| Рорир при бришење на книга | 5 |
| Опција за одобрување и одбивање на коментари | 5 |
| Приказ на сите одбиени коментари и опција за одобрување | 5 |
| Вкупно | 100 поени |
| Дополнителни поени | 20 поени |
| Менаџирање на приватните забелешки комплетно имплементирано со AJAX | 15 |
| Random цитат во футер преку API повик | 5 |
| Максимален број на освоени поени | 120 поени |

