

Имена: Иван Живков Арабаджийски	фн: 81631
Начална година: 2020	Програма: бакалавър, (КН) Курс: 4
Тема: 28.1	
Дата: 2020-02-02	Предмет: w15prj_KN_final имейл: ajabaja16@gmail.com

преподавател: доц. д-р Милен Петров

ТЕМА: Система за заемане на книги/реферати за 'четене'

1. Условие

Системата трябва да позволява от колекции от книги/списания/реферати (по уеб да речем) - онлайн и печатни, да се 'импортират' групово, да могат да се запазват и връщат, като да може да се задават параметри на заема - дни - да речем по подразбиране 20, брой на ресурси (да речем до 5 или 7), вид на ресурса - да може да се задава един ресурс от колко човека едновременно може да бъде зает - ако е зает му се дава достъп до линка - може да се кодира или нещо такова, ако му е изтекъл достъпа - трябва да отиде и да 'върне' ресурса и му се дава възможност да заема нов ресурс. да има статистика на най-четени/поръчвани ресурси, по потребители, по групи и т.н.

2. Въведение

Системата е направена да изглежда красиво и да има нужната функционалност. Направена е като уебсайт в който потребителите могат динамично да използват ресурсите. Навигацията е достатъчна за да ги улесни, а за самите функционалности са използвани база от данни за пазене на информацията. Нещата са поднесени на потребителя по красив начин. Има възможности да купиш, да върнеш, да качиш книга, както и история с твоите книги. Може да се регистрираш като нов потребител, като всеки започва със 10 точки за купуване на книги. Има достатъчна валидация на полетата, така че да не можеш да се регистрираш с вече съществуващо име или да влезеш без правилната парола. Уебсайта използва модерна архитектура и е под формата на "Single Page Application".

3. Теория

За име на нашия сайт сме избрали IT-Library. Името идва тематично от имената на неговите създатели – Иван и Теодор, като също така е и игра с думи с IT сферата. В нашата платформа за четене/заемане на книги искаме да дадем възможност на хората да се регистрират/аутентикират, да разглеждат, търсят и взимат за четене ресурси. Всеки потребител има даден брой кредити като взимането на **един** ресурс струва **един** кредит. При липса на кредити трябва да бъдат върнати ресурси. Пази се история на взетите ресурси, тези, които още не са върнати, могат да бъдат четени. Избрали сме да вградим четенето на pdf в canvas в браузъра, за да се предотврати вариант, в който потребителя успява да изтегли ресурса. Всеки потребител може и да качва книги за четене, като сме предпочели batch update, по подаден .csv файл да се качат всички книги написани в него, а не да трябва да се добавят един по един ръчно. Имаме и

функционалност за цитиране на книга – директно се копира на clipboard-а текст, който е научния труд готов за цитиране в друг научен труд. Използвали сме трислойна архитектура – html/css/js за фронтенда, php за бекенда и mysql за база данни.

<https://github.com/18ivan18/IT-Library> – гитхъб на проекта.

4. Използвани технологии

Фронтеда използва архитектурата MVC – model-view-controller. Използва се js за създаване на SPA – single page application, използвайки уеб компоненти. Има изграден рутер за рутиране аз страниците като в зависимост от различните пъща се визуализират различни компоненти. За споделяне на състояние из компонентите имаме имплементиран store – опростена версия на известния дизайн патърн редукс. За development използваме webpack dev server, а за production ще използваме компилираната от webpack версия, която ще бъде отворена в apache сървъра. Използва се npm за управление на различни development зависимости, като единствената production зависимост е пакетът „lit-html“, който се използва за улеснение при рендерирането на уеб компонентата. NPM версия 6.14.10, node версия 14.15.4.

За сървърната част използваме последната версия на XAMPP – v3.2.4 използвайки основно модулите Apache и MySQL и PHP – 7.3.26 – последна промяна 19.01.2021г.

Базата данни под името “project” съдържа в себе си три таблици – “users”, “transactions”, “library”.

Първата пази информация за всеки потребител като име, парола, имейл, име и точки за купуване на книга. Конкретен пример за потребител има в точка 7.

Втората пази информация за всяка направена заявка за купуване или връщане на книга. Там се пази коя е книгата, кой я е купил, дата на купуване и кога е била върната. Ако още не я е върнал датата си стои празна стойност. Конкретен пример – точка 7.

Последната таблица отговаря за самите книги и характеризира дадена книга с име, автор, бройки в библиотеката, описание и други.

За вграждане на pdf в браузъра се използва помощна библиотека pdfjs.

5. Инсталация и настройки

След клониране на проекта от гит репото се изпълнява еднократно команда `npm i` и за стартиране на development сървъра се пуска команда `npm run serve`. Localhost порта може да бъде конфигуриран в config.js файла. За подготвяне за production се пуска команда `npm run serve`. След като webpack е генерирал папката dist, тя се поставя в htdocks (C:\xampp\htdocs) и след стартиране на апачи сървъра и mysql базата от хамп сайтът се достъпва през браузъра чрез <http://localhost/dist/index.html>.

За да използваме настройките на нужната база от данни, трябва папката “project” да се копира в папката на XAMPP и по-конкретно `xampp/mysql/data`. На същото място се поставят и 3-те файла `ib_logfile0`, `ib_logfile1` и `ibdata1`.

6. Кратко ръководство на потребителя

В началната страница сайта е с традиционно меню за навигация, поле с информация и новини и два бутона горе в дясно за влизане и регистрация. Всички гости могат да търсят наличните книги и да видят контакти със създателите. Това е за гостите. Ако потребителя избере регистрация, може да се регистрира с уникално име и бива препратен в началната страница. Следва да се логне като напише име и парола и от там получава допълнителните функционалности. В първата страница, той вижда история с неговите книги, както и информация за профила му. От там той може да прочете книга или да върне, както и да следи за минали срокове. Също така има опция за качване на нови ресурси. Ако отвори библиотеката може да си закупи нови, ако нужните условия са налице – да има точки и книгата да е в наличност. При четене на дадена книга той може да я цитира като автоматично му се запазва в клипборда специфичен начин за цитиране на книга. Важно е да се отбележи, че самата книга не може да бъде изтеглена и плагиатствана, а само четена като ембедната в сайта.

7. Примерни данни

Примерни данни за базата “library”

1. ID – 9
2. Title – Book8
3. Author – Mark Twain
4. Type – book
5. Description – Once upon a time..
6. Count – 10
7. URL - <http://www.africau.edu/images/default/sample.pdf>
8. #tags - #kingdom
9. coverURL - <https://smartmobilestudio.com/wp-content/uploads/2012/06/leather-book-preview.png>
10. Quote – London.2002

Примерни данни за базата “transactions”

1. ID – 1
2. bookID – 9
3. buydate - 2021-02-02 06:58:26
4. user – teo
5. returndate – NULL

Примерни данни за базата “users”

1. ID – 5
2. username – teo
3. password – teo
4. name – teo
5. email – teo@teo.bg
6. points – 9
7. picture - <https://cdn3.iconfinder.com/data/icons/fillies-small/64/id-card-512.png>

8. Описание на програмния код

Фронтеда използва архитектурата MVC – model-view-controller. Използва се js за създаване на SPA – single page application, използвайки уеб компоненти. Функцията bootstrap слага компонента на дадено място в html-а. Тя се използва за начално стартиране на целия фронтенд.

```
1  import { App } from "./App";
2  import { bootstrap } from "./utils";
3
4  bootstrap(document.body, App);
5
```

В папка utils са изнесени доста от полезните функции, например router.js и подпапка store.

```
4  export class Router {
5    constructor(outlet, routes) {
6      this.routes = routes;
7      this.outlet = outlet;
8    }
9
10   route = () => {
11     const path = this.parseLocation();
12     const component = this.findComponentByPath(path) || { ctor: NotFound };
13     bootstrap(this.outlet, component.ctor, component.params);
14   };
15
```

Йерархически в папка components се намират всичките уеб компонентим, които използва рутера.

```

1  import { initialState, reducer } from "./reducers";
2
3  const createStore = (reducer, initialState) => {
4    const store = {};
5
6    store.state = initialState;
7    store.listeners = [];
8    store.subscribe = (listener) => store.listeners.push(listener);
9    store.dispatch = (action) => {
10
11      store.state = reducer(store.state, action);
12      store.listeners.forEach((listener) => listener(action));
13    };
14    store.getState = () => store.state;
15
16    return store;
17  };
18
19  export const Store = createStore(reducer, initialState);

```

Основния клас, който използваме е database. Той се свързва с базата данни и има член функции нужни за промени в базата данни или достъп до нея. Всеки отделен php файл създава обект от този клас и вика нужната му член функция.

Следват примери за самия клас, някои от член функциите и как другите php файлове го използват. Резултата към фронт енда се връща с echo.

```

1  <?php
2  class database{
3      private $connection;
4      public function __construct() {
5          $host = "localhost";
6          $username = "root";
7          $password = "";
8          $dbname = "project";
9          $this->connection = mysqli_connect('127.0.0.1', 'root', '', $dbname);
10         if (!$this->connection){
11             echo "not connected";
12         }
13         if (!mysqli_select_db($this->connection, 'project')){
14             echo "Cannot get db";
15         }
16     }
17     public function getLibraryInfo($data){
18         $title = $data[0];
19         $titleExactMatch = $data[1];
20         $author = $data[2];
21         $authorExactMatch = $data[3];
22         $tag = $data[4];
23         $sortBy = $data[5];
24         $order = $data[6];
25         $type = $data[7];
26
27         if ($titleExactMatch == 'true'){
28             if ($authorExactMatch == 'true'){
29                 if ($order == 'Descending'){
30                     $stmt = $this->connection->prepare("SELECT * FROM library WHERE Title=? and Author = ? and Tags LIKE CONCAT ( '%', ?, '%' ) and Type LIKE CONCAT ( '%', ?, '%' ) O");
31                 }
32                 else{
33                     $stmt = $this->connection->prepare("SELECT * FROM library WHERE Title=? and Author = ? and Tags LIKE CONCAT ( '%', ?, '%' ) and Type LIKE CONCAT ( '%', ?, '%' ) ORDER");
34                 }
35             }
36             else{
37                 if ($order == 'Descending'){
38                     $stmt = $this->connection->prepare("SELECT * FROM library WHERE Title=? and Author LIKE CONCAT ( '%', ?, '%' ) and Tags LIKE CONCAT ( '%', ?, '%' ) and Type LIKE CONC");
39                 }
40                 else{
41                     $stmt = $this->connection->prepare("SELECT * FROM library WHERE Title=? and Author LIKE CONCAT ( '%', ?, '%' ) and Tags LIKE CONCAT ( '%', ?, '%' ) and Type LIKE CONC");
42                 }
43             }
44         }
45     }
46 }

```

```

1 <?php
2 require_once('database.php');
3 header("Access-Control-Allow-Origin: *");
4 header("Content-Type: application/json; charset=UTF-8");
5 header("Access-Control-Allow-Methods: OPTIONS,GET,POST,PUT,DELETE");
6 header("Access-Control-Max-Age: 3600");
7 header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, Authorization, X-Requested-With");
8 $db = new database();
9 $result = $db->buyBook($_POST['id'],$_POST['username']);
10 echo json_encode($result);
11

```

```

170 }
171 public function quoteBook($data){
172     $id = $data;
173     $stmt = $this->connection->prepare("SELECT * FROM library WHERE ID = ?");
174     $stmt->bind_param("s",$id);
175     $stmt->execute();
176     $result = $stmt->get_result();
177     if ($result->num_rows > 0){
178         $row = $result->fetch_assoc();
179         if ($row['type'] == 'book'){
180             $bookQuote = $row['Author'].'. '.$row['Title'].'. '.$row['quote'];
181             return ['quote'=> $bookQuote];
182         }
183         else if ($row['Type'] == 'paper'){
184             $paperQuote = '['.$row['Author'].']. '.$row['Title'].', Available from:'.$row['quote'];
185             return ['quote' => $paperQuote];
186         }
187         else{
188             $magazineQuote = $row['Author'].(' '.$row['quote'].'), '.$row['Title'];
189             return ['quote' => $magazineQuote];
190         }
191     }
192 }
193 public function buyBook($idInput,$usernameInput){
194     $id = $idInput;
195     $user = $usernameInput;
196     $stmt = $this->connection->prepare("SELECT * FROM library WHERE ID = ?");
197     $stmt->bind_param("s",$id);
198     $stmt->execute();
199     $result = $stmt->get_result();
200     $row = $result->fetch_assoc();
201     if ($row['Count'] < 1){
202         return ['success' => false, 'message' => 'This article is sold out'];
203     }
204     $stmt4 = $this->connection->prepare("SELECT points FROM users WHERE username = ?");
205     $stmt4->bind_param("s", $user);
206     $stmt4->execute();
207     $result2 = $stmt4->get_result();
208     $row2 = $result2->fetch_assoc();
209     if ($row2['points'] < 1){
210         return ['success' => false, 'message' => 'Not enough points to buy the article'];
211     }
212     $newPoints = $row2['points'] - 1;

```

9. Приноси на студента, ограничения и възможности за бъдещо разширение

По-голямата част от времето се занимавах с архитектурата на фронтенда, като след това помогнах за разширяване на някои от функционалностите на сървъра, както и работих малко по базата данни. Ограниченията са основно откъм четене на ресурси, сегашната имплементация не е оптимална, но трябва да се отдели много повече време и да се измислят някои хитри решения на проблеми за да се подобри. Бъдето разширение би било възможност всеки човек да води записки по своите взети ресурси и после да може да експорт-ва цитати оттам, но това би означавало направа на собствен pdf редактор в брауъра. Друго подобрение за по краткото бъдеще е изработването на страница със статистика, оценяване на книгите, сортиране по най-взимани/харесвани.

10. Какво научих

Работа в екип. Работа с технологията php. Враждане на pdf четец в брауъра.

11. Използвани източници

<https://lit-html.polymer-project.org/> - The Polymer Project Authors

<https://stackoverflow.com/questions/10471405/how-to-bind-parameters-in-prepared-sql-statements> - Asked 2013.03, Last active – 2013.03

<https://stackoverflow.com/questions/49674187/how-do-i-connect-javascript-and-mysql-via-php> - Asked 2019.02 , Last active – 2019.02

<https://cdn3.iconfinder.com/data/icons/fillies-small/64/id-card-512.png> - Profile picture

<http://www.africau.edu/images/default/sample.pdf> - Sample PDF

<https://smartmobilestudio.com/wp-content/uploads/2012/06/leather-book-preview.png> - Cover url

<https://mozilla.github.io/pdf.js/> - embed pdf in the browser

Предал (подпис):

/фн, имена, спец., група/

Приел (подпис):

/доц. Милен Петров/