Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Звіт

З лабораторної роботи № 3

«Frontend частина системи ConditionsController»

З дисципліни «Архітектура, аналіз та проектування програмного забезпечення»

Підготував:

Ст. гр. ПІ-15-1

Євченко Богдан

2018

Оскільки frontend-частина проекту необхідна використовувати сервер незалежно від себе, то для зв’язку з сервером використовується JS. Тому всі методи запитів тільки повертають відповідну сторінку.

Наприклад:

@bp.route(**'/get'**, methods=[**'GET'**])  
**def** organization():  
 **return** render\_template(**'organization.html'**)

Для зв’язку з сервером використовуються наступні методи:

**var *conditions*** = ***conditions*** || {};  
***conditions***.**server** = (**function** () {

Необхідно вказати інтернет адресу сервера.  
 **var** server\_url = **"https://127.0.0.1:5002/api/"**;  
 *// var server\_url = "https://conditions-controller-olderor.c9users.io/api/";* Функція посилає запит на сервер з відповідними вказаними даними в форматі JSON. Та передає результат в функцію onDone, яка виконує наступні дії з поверненими даними.**function** *sendRequest*(action, data, onDone, url=server\_url) {  
 ***console***.log(action, data);  
 **$**.**ajax**({  
 **url**: url + action,  
 **type**: **'post'**,  
 **cache**: **false**,  
 **dataType**: **'json'**,  
 **contentType**: **'application/json; charset=utf-8'**,  
 **data**: ***JSON***.stringify(data)  
 }).*done*(**function** (response) {  
 ***console***.log(response);  
 **if** (onDone) {  
 onDone(response);  
 }  
 }).*fail*(**function**() {  
 **if** (onDone) {  
 onDone();  
 }  
 }).always(**function** () {  
 ***conditions***.**account**.*updateButtons*();  
 });  
 }  
Функція аналогічна попередній, але використовує авторизацію через токен.  
 **function** *sendAuthorizedRequest*(action, token, data, onDone, url=server\_url) {  
 ***console***.log(action, data);  
 **$**.**ajax**({  
 **url**: url + action,  
 **type**: **'post'**,  
 **cache**: **false**,  
 **dataType**: **'json'**,  
 **contentType**: **'application/json; charset=utf-8'**,  
 **data**: ***JSON***.stringify(data),  
 **headers**: {  
 **"Authorization"**: **"Bearer "** + token  
 }  
 }).*done*(**function** (response) {  
 ***console***.log(response);  
 **if** (onDone) {  
 onDone(response);  
 }  
 }).*fail*(**function**() {  
 **if** (onDone) {  
 onDone();  
 }  
 }).always(**function** () {  
 ***conditions***.**account**.*updateButtons*();  
 });  
 }  
  
 **return** {  
 sendRequest: *sendRequest*,  
 sendAuthorizedRequest: *sendAuthorizedRequest* };  
})();

Приклад функції, яка отримує інформацію про продукт від сервера, вказавши ідентифікаційний номер продукту.  
**function** *getProductInfo*(product\_id, onDone) {  
 **var** data = {**'product\_id'**: product\_id};  
 ***conditions***.**server**.*sendAuthorizedRequest*(**'product/get'**, ***conditions***.**account**.*getToken*(), data, **function** (response) {  
 onDone(response);  
 });  
}

Функція яка оновлює дані веб-сторінки. Використовуючи попередню функцію, вона отримує інформацію з сервера та заповнює веб-сторінку цією інформацією за допомогою інструменту js-template. Також вона генерує графіки стану продукту та QR-код для доступу з мобільного додатка.  
**function** *reloadProductInfo*() {  
 *getProductInfo*(*getCurrentProductId*(), **function** (response) {  
 **var** productError = **$**(**'#product-error'**);  
 ***conditions***.**general**.*processResponse*(response, productError, **function** (response) {  
 **var** dataItems = response.product;  
 **if** (!dataItems[**'tracking\_device\_id'**]) {  
 dataItems[**'tracking\_device\_id'**] = **''**;  
 }  
 **$**(**'#product-description'**).html(**$**(**'#product-template'**).tmpl([dataItems]));  
 ***conditions***.**tracking\_device**.*initAssignDeviceButtons*();  
 **if** (!is\_chart) {  
 is\_chart = **true**;  
 *initChart*(dataItems.**conditions**, dataItems.tracking\_statuses);  
 }  
 **var** qrcode = **new *QRCode***(**"qrcode"**, {  
 **text**: *getCurrentProductId*(),  
 **width**: 128,  
 **height**: 128  
 });  
 });  
 });  
}