

Софийски университет “Св. Климент Охридски”  
Факултет по математика и информатика

*Разработка на клиент-сървър (fullstack)  
приложения с Node.js + Express.js +  
React.js*

Проект на тема:

***Pass-FMI***

Изготвили:

1. Биляна Господинова, 81121
2. Даниел Шушков, 81102

Ръководител:.....

(Траян Илиев)

## Съдържание

1. Мисия на проекта.....	3
2. Типове потребители в системата.....	3
3. Основни потребителски случаи.....	4
4. Основни изгледи.....	5
5. Server Resource API.....	5
6. Използвани технологии.....	7
7. Нефункционални характеристики.....	7
8. Дизайн.....	8
9. Заключение - срещнати/преодолени трудности и перспективи за бъдещо развитие на системата.....	11
10. Източници.....	11

## Мисия на проекта

В процеса на своето обучение във висше учебно заведение студентите се нуждаят от голямо количество учебни материали, с които да се подготвят за курсовете си. Понякога намирането на подходящ източник за подготовка се оказва доста трудно. От друга страна, след приключване на съответния курс доста от студентите имат учебници и учебни помагала, които не са им необходими вече и желаят да ги предоставят на други студенти, на които да им бъдат от полза.

Проектът *Pass-FMI* цели да предостави система на студентите, в която по лесен, ефективен и удобен начин те да имат възможност да търсят и предлагат учебни материали с цел подпомагане на обучителния процес.

## Типове потребители в системата

- Анонимен - това е потребител, който няма регистрация в сайта или не се е вписал в дадения момент. Този потребител има достъп до началната страница на приложението, може да разгледа мисията на проекта и качените учебни материали. Предоставя му се възможност да се регистрира или впише в системата.
- Студент - това е анонимен потребител, който има възможност да пише коментари към добавените учебни материали, да добавя свои и да ги модифицира.
- Администратор - това е студент или университетски служител, който може да изтрива коментар на всеки потребител, да

администрира всички потребители, да добавя, редактира и премахва всички учебни материали в системата.

## Основни потребителски случаи

Потребителски случай	Кратко описание	Участващи потребители
Разглеждане на информация и материали	Всеки потребител може да разгледа началната страница, мисията на проекта и качените учебни материали, може да ги сортира по категории и търси по ключови атрибути.	Всички
Регистрация	Анонимните потребители могат да се регистрират в системата, като предоставят потребителско име, парола, първо име, фамилия, електронен адрес и телефон. По подразбиране след регистрация, потребителят има ролята Студент.	Анонимен потребител
Вписване	Регистриран потребител може да се впише в системата, като предостави потребителско име и парола.	Студент, Администратор
Разглеждане на лична информация	Вписаният потребител може да разгледа личните си данни, които е въвел по време на регистрация, както и собствените качени материали с опция да ги редактира през отделно view.	Студент, Администратор
Администриране на потребители	Администраторът може да прегледа списък с всички потребители и качените от тях учебни материали, може да ги изтрива или модифицира.	Администратор
Администриране на учебни материали	Студентите и Администраторите могат да добавят учебни материали. Студентите могат да редактират и изтриват собствени учебни материали. Администраторите могат да редактират и изтриват учебни материали на всички потребители.	Студент, Администратор

Филтриране на учебни материали	Всички потребители могат да задават критерии, по които да търсят материали	Всички
--------------------------------	--	--------

## Основни изгледи

Име на изгледа	Кратко описание	URI
Начало	Предоставя информация за целта на проекта, предоставя възможност за регистрация и вход в системата.	/home
Мисия	Предоставя информация за мисията на проекта.	/about
Лични данни	Визуализира информация за вписания потребител.	/personal
Учебни материали	Предоставя списък с всички налични учебни материали и филтри по категории.	/items
Потребители	Визуализира списък с всички регистрирани в системата потребители	/users

## Server Resource API

Име на ресурса	Кратко описание	URI
Регистрация	POST: Създава потребител по зададени атрибути	/api/users/register
Вход в системата	POST: Вход на потребител в системата посредством потребителско име и парола	/api/users/login
Изход от системата	GET: Дава възможност на потребител да излезе от системата	/api/users/logout

Списък с потребители	GET: Връща списък с всички регистрирани потребители в системата	/api/users/usersList
Данни за потребител	GET: Връща всички данни за даден потребител	/api/users/{username}
Списък с учебни материали	GET: Връща списък с всички добавени в системата учебни материали	/api/items/itemsList
Списък с учебни материали за даден потребител	GET: Връща списък с всички добавени учебни материали за даден потребител	/api/items/itemsList/{username}
Данни за учебен материал	GET: Връща данни за даден учебен материал	/api/items/{id}
Модифициране на учебен материал	PUT: Модифициране на данни за даден учебен материал	/api/items/{itemId}
Добавяне на учебен материал	POST: Добавяне на учебен материал с дадени атрибути	/api/items
Изтриване на учебен материал	DELETE: Изтриване на учебен материал по негово id	/api/items/deleteItem/{itemId}
Списък с всички добавени катедри	GET: Връща списък с всички катедри към учебните предмети	/api/items/departments
Списък с всички предмети	GET: Връща списък с всички предмети	/api/items/subjects
Добавяне на коментар към учебен материал	POST: Добавя въведен от потребителя коментар към даден учебен материал	/api/items/addComment/{itemId}
Изтриване на коментар към учебен материал	DELETE: Изтриване на коментар към даден учебен материал	/api/items/deleteComment/{itemId}/{commentId}
Списък с всички добавени коментари	GET: Връща списък с всички добавени коментари	/api/comments/commentsList

## Използвани технологии

- Сървърна част
  - Express - библиотека, предоставяща routing и middleware framework за изграждане на NodeJS приложения;
  - Passport - библиотека, даваща възможност за изграждане на аутентикация в NodeJS приложения;
  - Mongoose - библиотека, посредством която се извършва достъп до базата данни;
  - Joi - библиотека за верификация, която се използва за проверка на потребителския вход
- Клиентска част
  - React - библиотека за изграждане на UI;
  - React-router - дава възможност за навигация между отделните изгледи;
  - Redux - дава възможност за управление на данни;
  - Bluebird - библиотека за Promise-и;
  - Lodash - библиотека, предоставяща възможност за работа с обекти, списъци, низове;
  - Axios - HTTP-клиент, базиран на Promise-и;
  - Bootstrap - UI библиотека за респонсив дизайн

## Нефункционални характеристики

- Сигурност - за достъп до модификация на определени ресурси са необходими съответните роли.
- При неподходящо съдържание на качените материали те могат да бъдат редактирани или премахнати от администратор.
- Бързина - при промяна в данните за визуализирания обект, той се рендерира наново от компонентите, които го използват.

- Леснота на използване - работата с приложението е интуитивна и базирана на предишен опит на потребителите. Освен това ако потребителят не желае да добавя коментари или учебни материали, той не е нужно да се регистрира.

## Дизайн

### 1. Начална страница

[Начало](#) | [Мисия](#) | [Учебни материали](#)

Pass FMI

Каква е целта на приложението?

Pass FMI е проект, който цели да подпомогне студентите в намиране на учебни материали, необходими за техните курсове.

За да пишете коментари или да качвате материали, моля първо се регистрирайте и влезте в приложението.

Регистрация

Потребителско име:

Парола:

Име:

Влизане

Потребителско име:

Парола:

Влез



## 2. Лични данни на вписания потребител

[Начало](#) | [Мисия](#) | [Лични данни](#) | [Учебни материали](#) | [Потребители](#) | [Изход](#)

Pass FMI

Име: Admin

Фамилия: Admin

E-mail: admin@abv.bg

Username: Admin

Моите учебни материали

Geometry	Geometry	Admin
----------	----------	-------

## 3. Списък с учебни материали и филтриране

[Начало](#) | [Мисия](#) | [Лични данни](#) | [Учебни материали](#) | [Потребители](#) | [Изход](#)

Pass FMI

Учебни материали

Subject:

-- Избери --

Username:

Department:

-- Избери --

Title:

Изчисти

Заглавие

Предметна област

Добавил

Functional Programming

Programming

Administrator

Geometry

Geometry

Admin

Добави

## 4. Данни за потребител

[Начало](#) | [Мисия](#) | [Лични данни](#) | [Учебни материали](#) | [Потребители](#) | [Изход](#)

### Pass FMI

Потребителско име	Admin
Първо име	Admin
Фамилия	Admin
E-mail	admin@abv.bg
Телефон	123456
Роля	Admin

### Собствени учебни материали:

Учебен материал	Предмет	Катедра	Цена
Geometry	Geometry	A & G	7

## 5. Детайли на учебните материали с коментари

[Начало](#) | [Мисия](#) | [Лични данни](#) | [Учебни материали](#) | [Потребители](#) | [Изход](#)

### Pass FMI

#### Learn OOP

Автор:		Добавил:	dshushkov
Предметна област:	OOP	Телефон:	083654587
Катедра:	CS	e-mail:	dani1@dani.com

Цена: 11

Edit

Delete

#### Коментари

1. 2017-09-01 14:20:51: [dshushkov] : Коментар
2. 2017-09-01 20:33:06: [dani] : Коментар 2



Добавете коментар...

Добави

## Заклучение - срещнати/преодолени трудности и перспективи за бъдещо развитие на системата

Проблемите, с които се сблъскахме, бяха свързани с използването на нови за нас технологии и като цяло нов език за програмиране. За първи път се сблъскахме с моделирането и имплементирането на NoSQL база от данни. MongoDB ни предостави два начина на рефериране и свързване на обекти. Решихме да изпробваме и двата типа. Успяхме да ги съчетаем в една обща структура от документи. Също проблем, основаващ се върху липсата на предиен опит с React и Redux, бяха определянето на структурата на стора и имплементирането на контейнерите и компонентите. Реакт рутера беше най-ужасяващият рутер, с който сме работили. В бъдеще системата може да се развие с качването на картинки, запазване на сесията в Local Storage и възможност за качване на електронни документи в системата, които да са със свободно ползване.

## Източници

1. [REDUX] <https://egghead.io/lessons/javascript-redux-the-single-immutable-state-tree> (август, 2017)
2. [REACT] <https://github.com/ReactTraining/react-router/tree/v3/docs> - React Router Documentation (август, 2017)
3. [PSSPT] <http://passportjs.org/docs> - Passport Documentation (август, 2017)
4. [JOIRF] <https://github.com/hapijs/joi/blob/v10.6.0/API.md> - Joi Reference (август, 2017)
5. [RCTOD] <https://facebook.github.io/react/tutorial/tutorial.html> - React Official Tutorial and Documentation (август, 2017)
6. [RDXOD] <http://redux.js.org/docs/introduction/> - Redux Official Documentation (август, 2017)
7. [MNGOS] <http://mongoosejs.com/docs/api.html> - Mongoose API Documentation (август, 2017)