**E-Learning System**

## **1. Описание**

Проектът представлява онлайн система за организация на учебния процес за студенти. Целта му е да служи като пространство, в което студентите да виждат и управляват всичко необходимо за своите курсове, а учителите от своя страна да предоставят и преглеждат същите.

## **2. Въведение**

Следният документ представя E-learning системата.Тук са описани начините за инсталациата и използването ѝ, на базата на какви технологии е изградена, какво се случва в кода на този проект, някои тестови данни както и как може да бъде подобрен.

## **3. Теория**

Системата се състои от три части:

● Презентационен слой – Front-end. Главната цел на този слой е да представи лесен и приятен за използване потребителски интерфейс. Тук сме използвали **React**.

● База от данни, където съхраняваме данните на приложението във формата, описан в следващата част. Използвали сме **MariaDB**.

● Back-end- Грижи се за всички заявки и бизнес логиката на приложението. Използвани технологии са **NodeJS** + **NestJS**. Тук се осъществява връзката с базата от данни с помощта на ORM Framework-a **TypeORM**.

## **4. Използвани технологии**

* NodeJS + NestJS
* TypeORM
* MariaDB
* ReactJS (с функционални компоненти)
* Emotion (for React)
* MDBReact (Material Design for React)

## **5. Инсталация, настройки и стартиране**

След изтегляне на кода от репото, трябва да се направят няколко неща:

**Инсталация**

За да инсталираме, необходимите пакети за проекта, трябва да напишем командата **npm install** на root ниво в папката Frontend и отново на root ниво в папката Backend.

**Настройки**

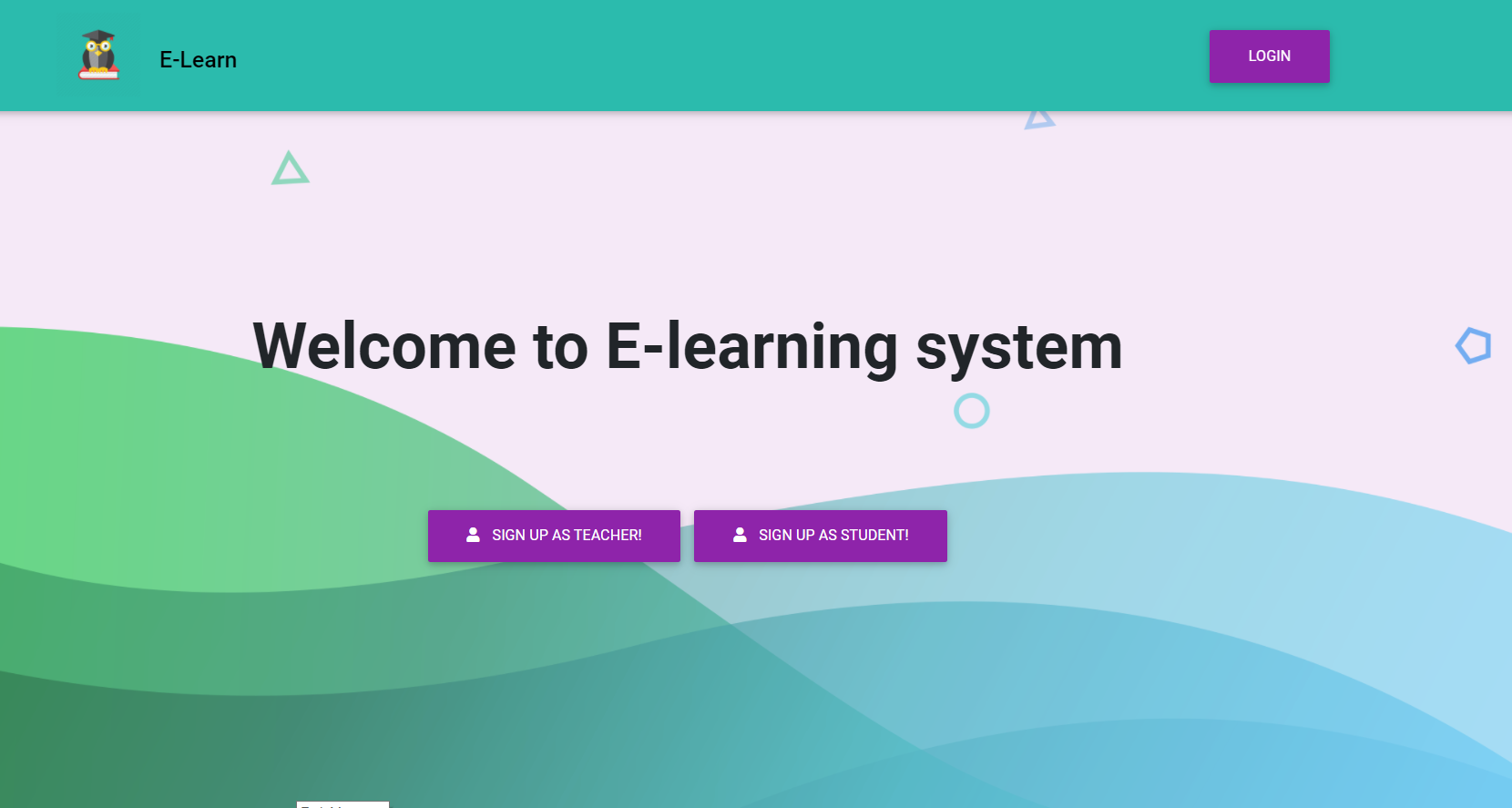
За да стартираме проекта, трябва първо да дадем правилните настройки за нашата база. За целта навигираме до **Backend/src/app.module.ts** и там до password вместо ‘***root***’ слагаме нашата парола (обикновено, ако не сме задали изрично парола, тя е ‘***root***’.След това отиваме в database design tool-a, който използваме (ние сме използвали MySql Workbench и HeidiSQL, но [тук](https://dbmstools.com/categories/database-design-tools/mariadb) можете да намерите и други подобни GUI за целта) и си създаваме схема с име ***‘e\_learning\_system’***.

**Стартиране**

* Стартираме сървъра като отворим конзола в папката Backend на root ниво и напишем **npm run start:dev,** за да подкараме сървъра в developer mode.
* Стартираме клиента като отворим конзола в папката Frontend на root ниво и напишем **npm start**

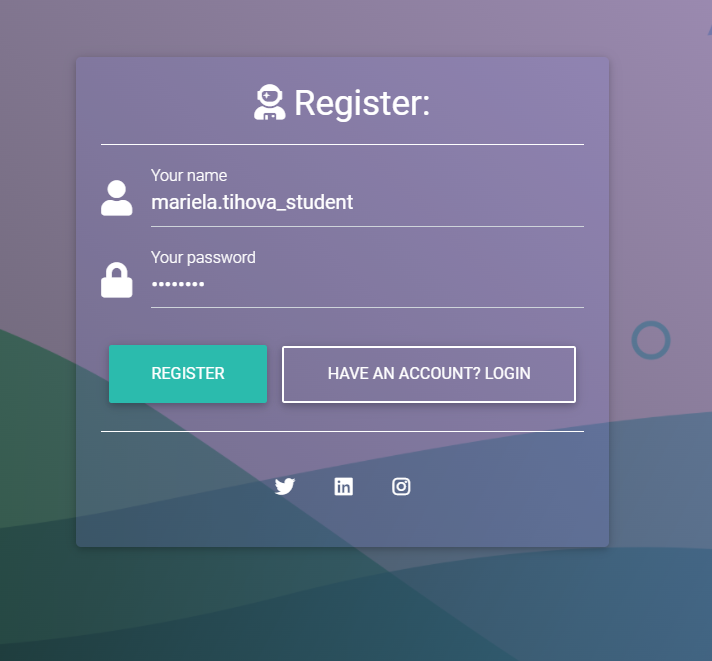
Отваря се прозорец в дефолтния ни браузър и можем да разгледаме приложението там.

## **6. Кратко ръководство на потребителя**



**Фиг.1. Начална страница за невписани потребители**

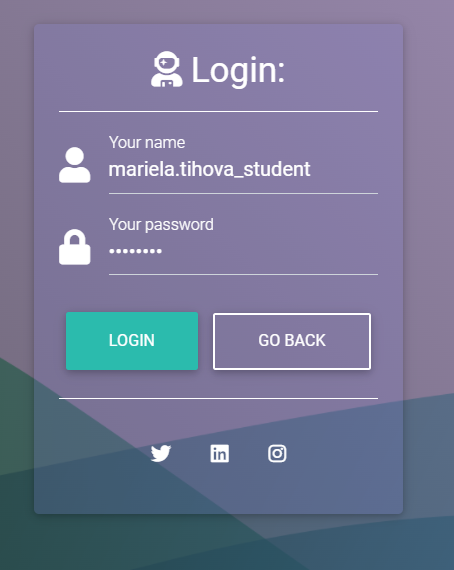
Началният екран за невписани потребители изглежда, както е показано на фигура 1. Там имаме опции за регистрация като учител, регистрация като студент (Sign up бутоните в средата на страницата) и бутон за login.



**Фиг.2. Форма за регистрация**

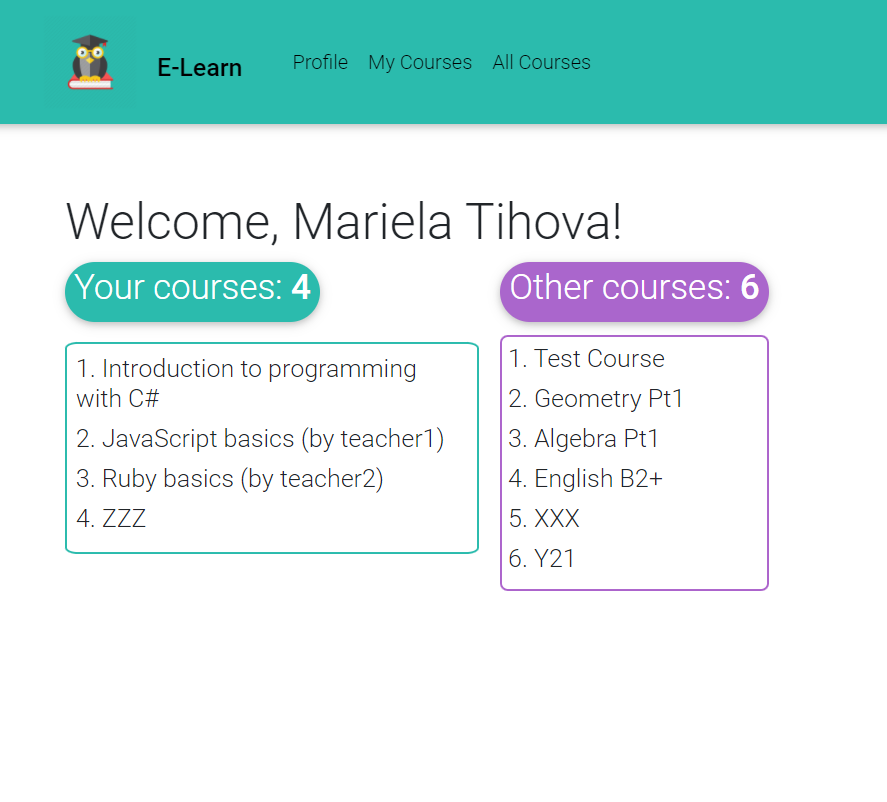
Формата за регистрация изглежда по един и същи начин както за студенти, така и за преподаватели. В полето за име въвеждаме желаният от нас username, а в полето password- паролата ни. Ограничения за полетата:

* *username → между 5 и 30 символа*
* *password → поне 5 символа, да съдържа поне 1 буква и 1 цифра*

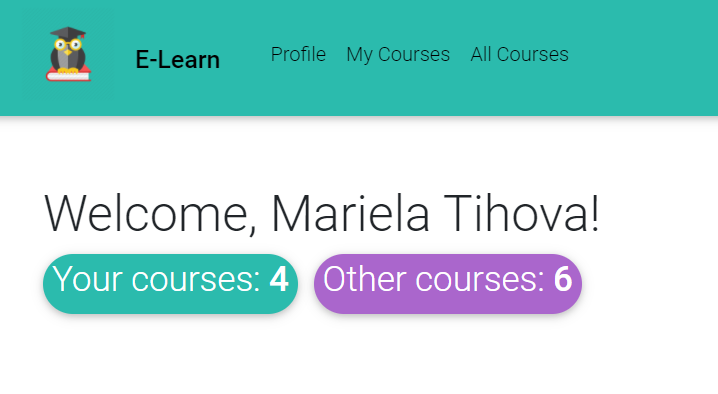
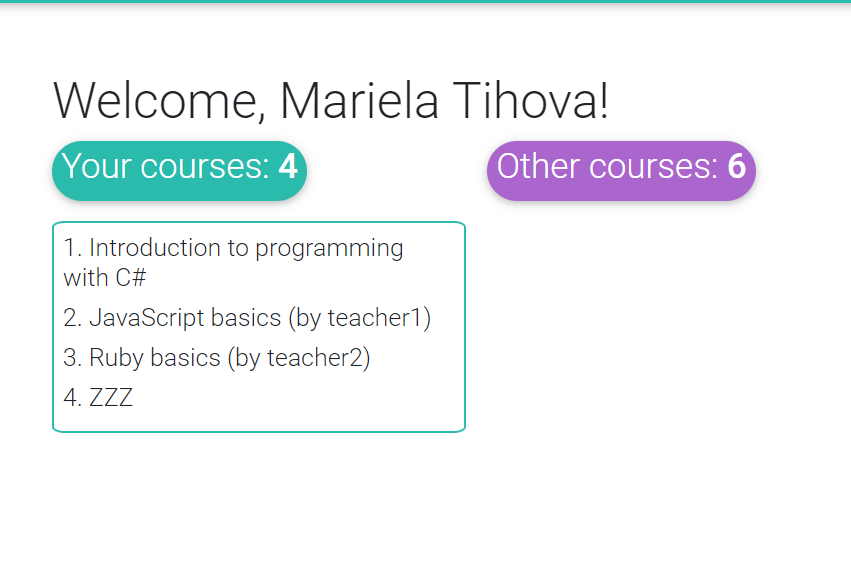
****

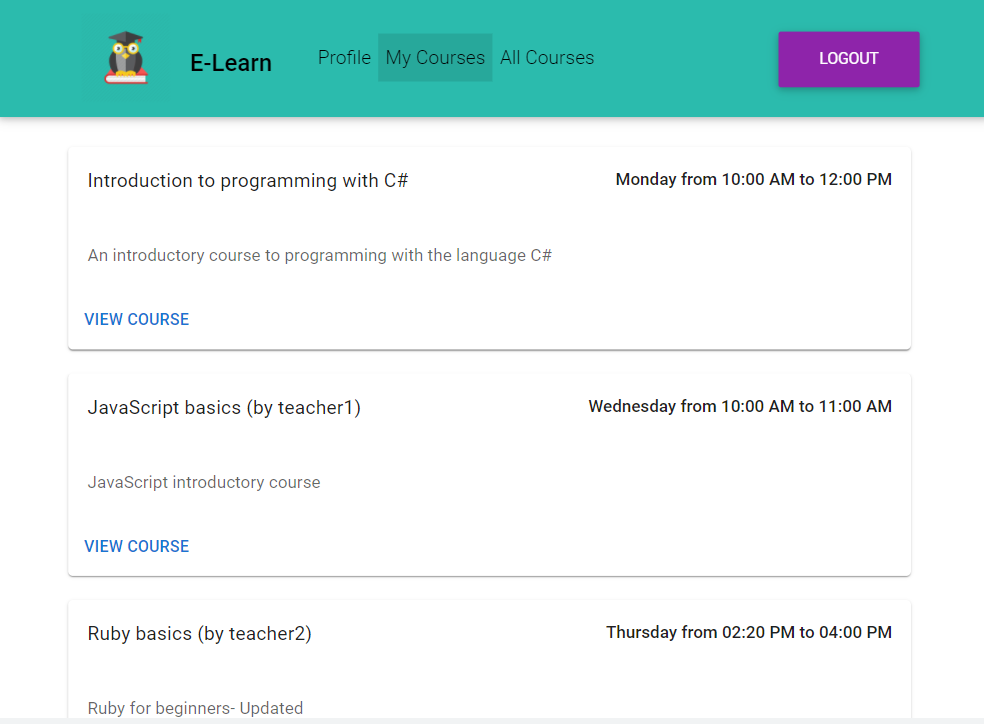
**Фиг.3. Login форма**

След успешна регистрация, бивате препратени към login формата, която изглежда по начина показан на горната фигура. Полетата за име и парола са автоматично попълнени, ако сме достъпили формата непосредствено след регистрация. Ако сме я достъпили чрез login бутона горе вдясно на страницата, тогава те ще бъдат празни.



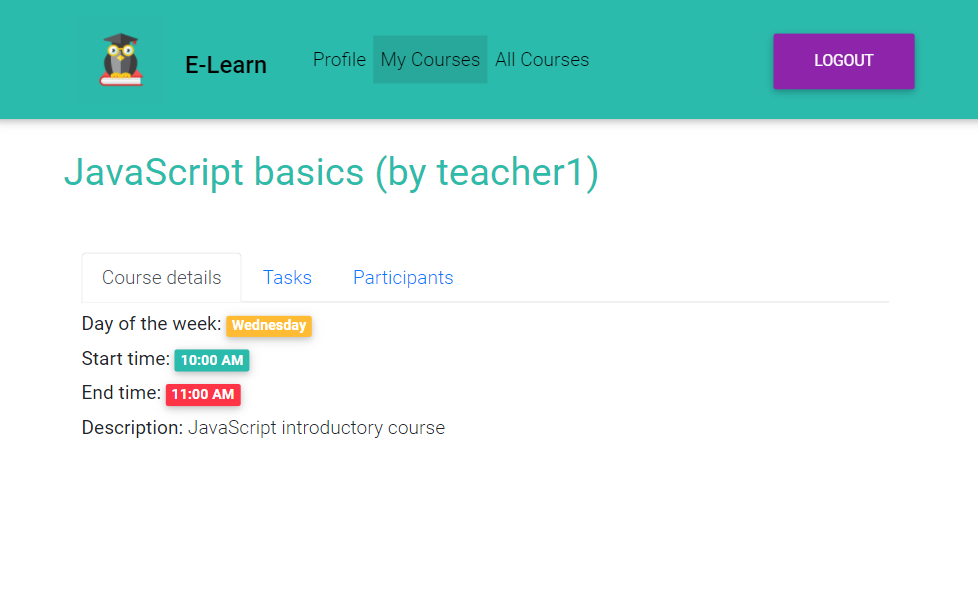
**Фиг.4. Home page**

След вписване в системата, потребителят бива препратен към home page за вписани потребители, който изглежда като на фигура 4. Тук можем да видим всички курсове, в които е записан потребителят (ако той е студент или всички курсове, които води, ако е преподавател) под графата ‘Your courses’, както и всички останали курсове, към които би могъл да се присъедини- под графата ‘Other courses’. Тези 2 графи са collapsible.

****

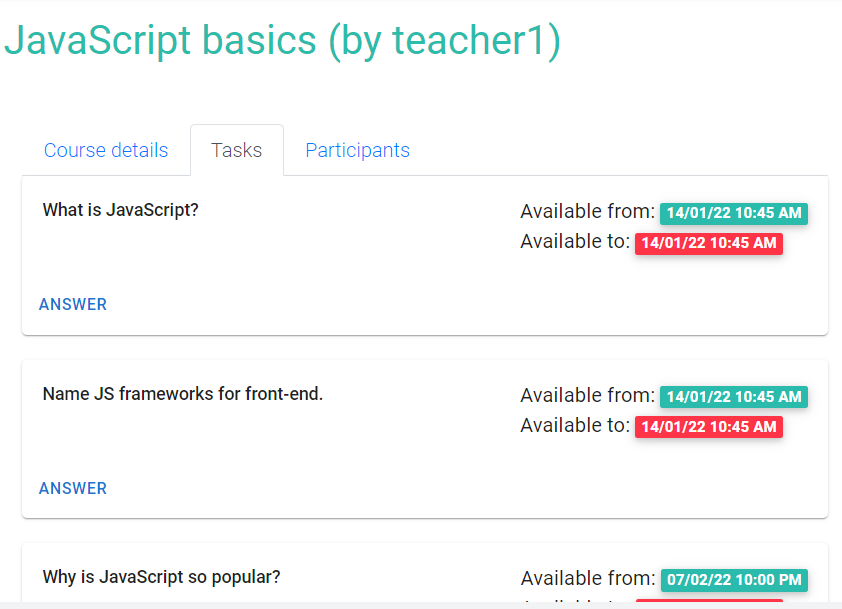
**Фиг.5. My courses**

Всички курсове, в които сме записани, можем да видим като изберем от менюто My Courses. За всеки курс имаме информация за неговото име, кратко описание, ден и часови период, в който се провеждат лекциите. Бутонът ‘View course’ при всеки курс, ни води към по-детайлна информация за съответния курс.



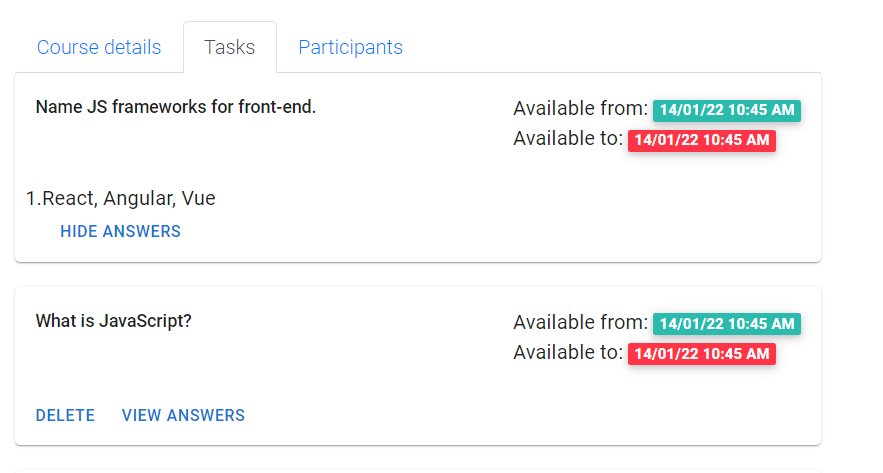
**Фиг.6. Course details**

След кликане на ‘View course’ бутона от фигура 5, потребителят бива препратен към това, което виждаме на фигура 6. В първия таб имаме детайли за курса- ден и часове на провеждане, както и кратко описание.



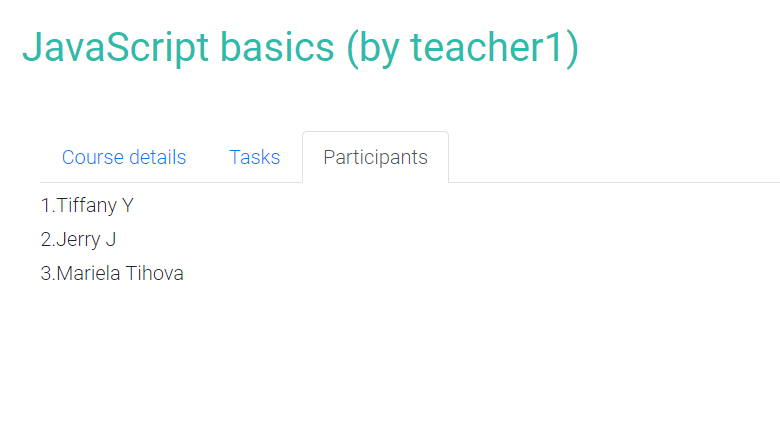
**Фиг.7. Course tasks (Student view)**

На фигура 7 виждаме как изглежда втория таб при view-то за конкретен курс. В него са показани задачите към съответния курс с времевия период, в който са достъпни, както и опция потребителят (студентът) да посочи отговор. Същият таб за преподаватели изглежда така:



**Фиг.8. Course tasks (Teacher view)**

Преподавателите могат да виждат кои са задачите към съответния курс, да ги трият, да създават нови такива, както и да видя отговорите от студенти.

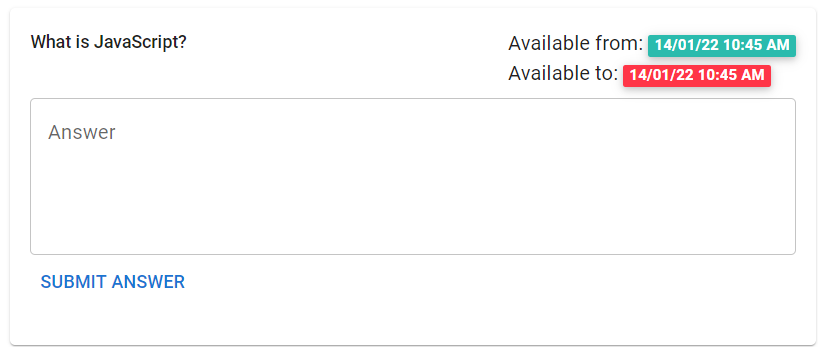


**Фиг.9. Course participants**

## 

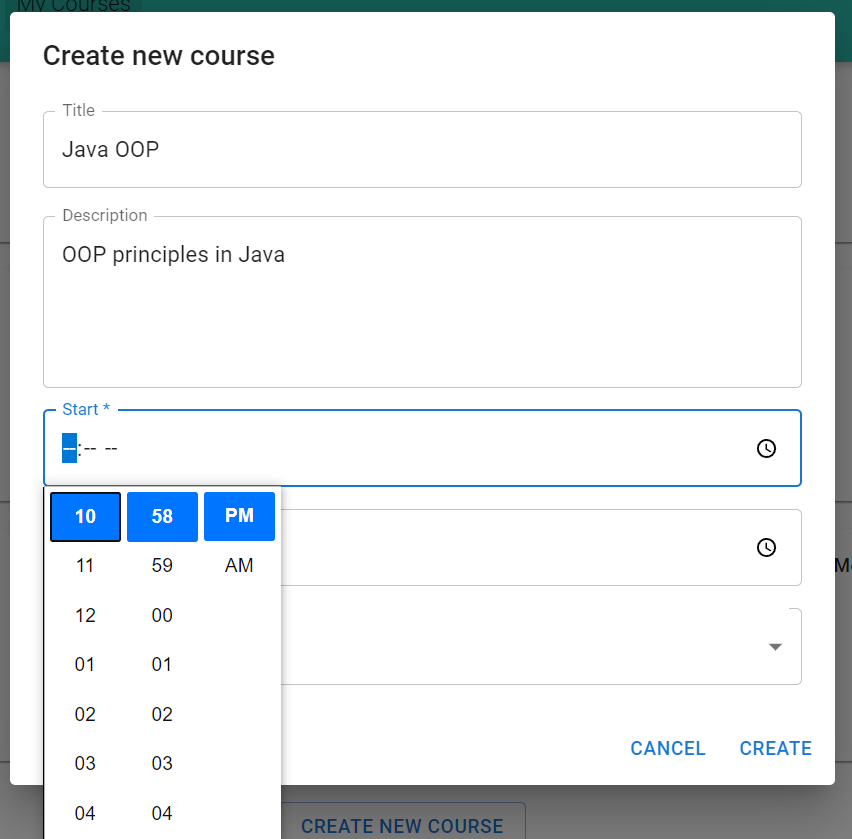
**Фиг.10. All courses**

Всеки студент може да види списък с абсолютно всички налични курсове, като под тези, в които не е записан, има опция ‘Join’. Преподавателите нямат този панел в своето навигационно меню. Единствено могат да видят кои са курсовете, в които не са включени от home page -> Other courses.



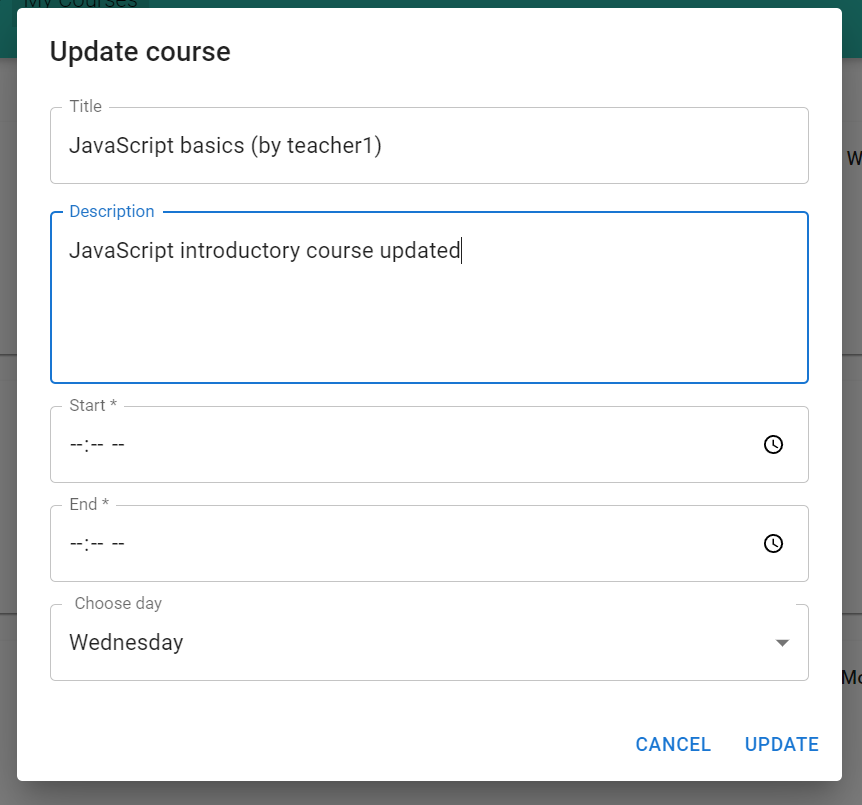
**Фиг.11. Answer task (for students)**

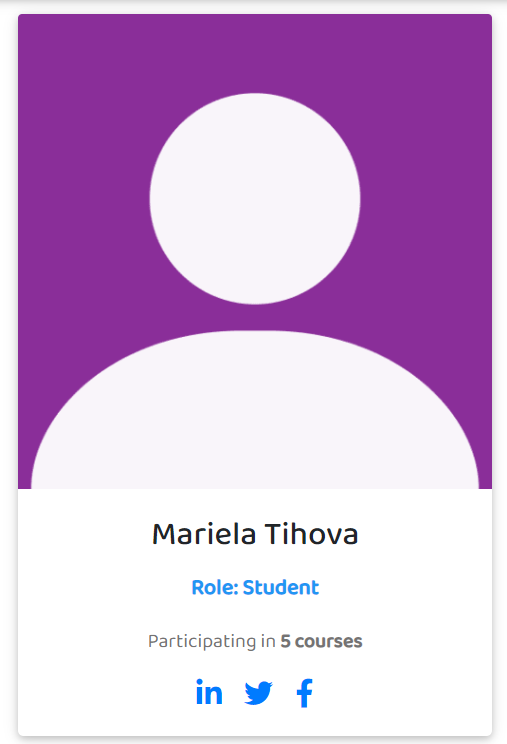
При натискане на бутона ‘Answer’, се отваря едно поле, в което студентът може да въведе своя отговор и да го предаде.



**Фиг.12. Create new course (for teachers)**

Всеки преподавател може да създаде нов курс. Полетата за попълване се виждат на фигура 12. При кликване на бутона ‘Save’ курсът ще бъде създаден и веднага ще бъде видим в списъка от курсове на преподавател. При кликване някъде извън модала за създаване на курс, той ще се затвори.



**Фиг.13. Update course (for teachers)**

**Фиг.14. Profile**

Избирайки Profile от менюто, потребителят може да види своя профил,съдържащ кратка информация за него- име, роля и в колко курса е вписан. За преподаватели се показва броят на курсовете, в които преподават.

## **7. Примерни данни**

Примерни данни за проекта могат да бъдат намерени във файла test-data.sql, който се намира в **Backend/src/test-data**. Може просто да импортнете sql файла в MySql Workbench/HeidiSQL.

**8. Заявки**

Postman колекция със заявки може да бъде намерена в **Backend/src/postman**. Може директно да бъде импортнат файла в postman.

**8. Изпълнени задачи**

При разработването на проекта бяха направени няколко цикъла на обсъждане и изпълнение на задачи. В самото начало екипът, състоящ се от Мариела Тихова и Богомила Пенева, обсъдихме и написахме заедно базовите неща от сорта на изграждане на моделите на backend-a, определяне на връзки между тях и какви основни функционалности ще разработваме. След това си разпределяхме по равно количество задачи, като си правехме срещи след завършване на предходните да обсъдим следващите. Тъй като искахме да движим за един цикъл цели функционалности, предимно разделението беше- за backend дадената функционалност се пише от Мариела, а за frontend от Богомила. Като разбира се и обратното се е случвало, но в по-малка степен. Всяка една от нас обаче е запозната с целия код. Просто установихме, че този подход за разделяне на задачи работи най-добре за нас. Разбира се историята на commit-ите също може да бъде проследена в репото в Github, ако се иска да бъде всичко съвсем конкретно.

**9. Бъдещи подобрения**

За съжаление останаха части от изискванията на проекта, които не успяхме да завършим поради много струпване на проекти, сравнително по-малкото време за работа за нас (тъй като се записахме в курса началото на ноември), както и броя хора- 2 (екипите могат да бъдат до 5 човека). Липсващи функционалности, които може да успеем да довършим до представянето, са:

* Всичко свързано с материали към курс (оказа се, че тази част е не особено тривиална и сложността на проекта се вдига много от нея)
* Тестове към курсове
* Администраторски панел (имаме роля администратор, както и реалзирани функционалности, свързани с него, но към момента те са само в backend-a.

**10. Изработили**

* **Богомила Пенева (f107307)**
* **Мариела Тихова (f106664)**