**Имена:**  Стефан Николаев Ламбов **фн:**  *61990***Начална година:** *2019* **Програма:** бакалавър, (СИ) **Курс: 3**   
**Тема: Цветомузика/ColorsOfTheMusic  
Дата: 2019-07-01 Предмет: wwwTech2018\_19\_12ed\_SI\_summer**

**имейл: chechko97@abv.bg**

**преподавател:** доц. д-р Милен Петров

# ТЕМА: Цветомузика

Съдържание:

## 1. Условие

## 2. Въведение

## 3. Теория

## 4. Използвани технологии

## 5. Инсталация и настройки

## 6. Кратко ръководство на потребителя

## 7. Примерни данни

## 8. Описание на програмния код

## 9. Приноси на студента, ограничения и възможности за бъдещо развитие

## 10. Какво научих

## 11. Използвани източници

Предал (подпис): ………………………….

/*61990, Стефан Ламбов,СИ, 5-та група*/

Приел (подпис): ………………………….

/доц. *Милен Петров*/

**Цветомузика/ColorsOfTheMusic**

1. **Условие**

Color music effects. Дефинират се 'карта' на места в зала (например 101 в ФМИ) и разположение на лаптопи (и/или) телефони. При стартиране на предварително записани ефекти - всеки от свързаните клиенти 'излъчва' (сменя си цветовете) - по предварително дефинирани команди. Да се направи редактор/визуализатор на ефекти. ; (роля администратор и клиент): (евентуално на повече от един монитор/компютър/телефон. – стартира екран и ‚мига‘ съответствие с командите от централната; евентуално различни ограничения/характеристики); - комуникация по websocket-и.

1. **Въведение**

Първото нещо, за което може да се сети човек, след като чуе думата „Цветомузика“, е излъчване на светлина с непрекъсната смяна на цветовете. Идеята на моето приложение да внеса малка доза забавление по време на обучението на студентите във ФМИ, като например се намираме във зала 101 и лекторът в ролята си на админ е създал виртуална стая, в която всеки студент може да се логне като клиент на определено място. Преподавателят може да задава и сменя цвят на фон на устройството на всеки логнат студент, било то от смартфон или лаптоп. По този начин, ще се получи една многоцветна стая, изпълнена с емоции и усмивки на лицата на студентите.

1. **Теория**

Функциоанлностите на системата зависят от типа потребители. Поддържат се два типа потребители – администратор и клиент. При първоначално влизане в системата всички потребители са клиенти. Клиентът става администратор, когато създаде стая и тази роля се отнася само за създадената от него стая. Може да има само един администратор за определена стая, а броят на клиентите зависи от броя на местата на стаята, която е създадена от самия администратор. Може да се създават множество стаи, които се различават по roomID и за всяка стая ще има по един администратор.

Клиентите могат да се логнат в някоя стая като въведат roomID и номер на място и очакват самата „Цветомузика“ да се активира от администратора.

След създаването на определена стая се генерира roomID за съответната стая, чрез което клиентите могат да влезнат в стаята на определено място. След като някой клиент влезне в стаята, на администратора се активира бутонче за мястото в стаята, на което е седнал този клиент и администраторът може да задава и променя цвета на фона на устройството на съответния клиент.

1. **Използвани технологии**
2. Версия 10.1.38 на MariaDB за база данни на приложението.
3. HTML, CSS и Javascript за клиентската част на приложението.
4. Версия 10.16.0 Node.js за имплементиране на backend-a.

Забележка: Версиите се отнасят за ползваната инсталация на XAMPP на Windows 10.

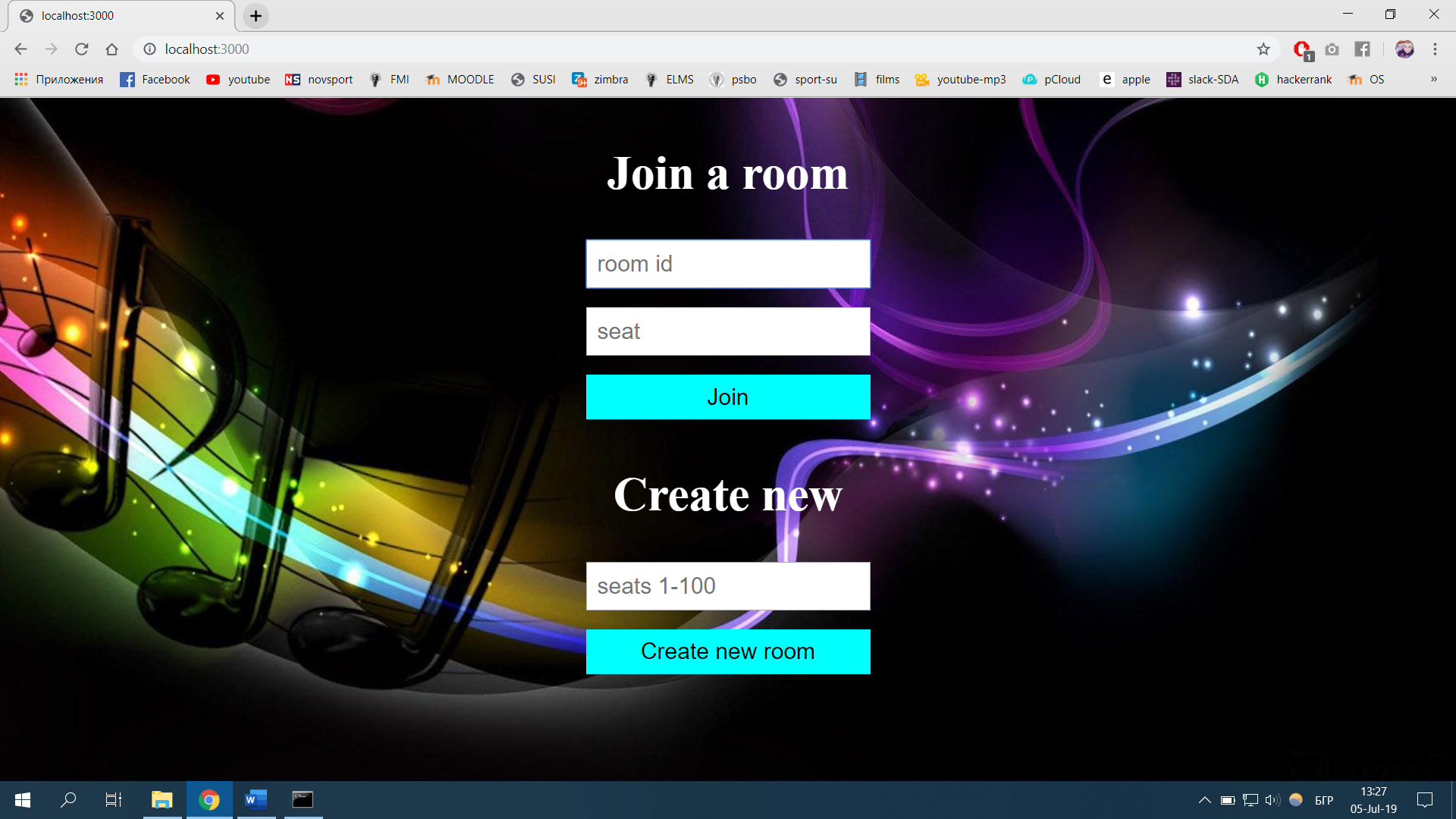
Забележка: Приложението е тествано успешно на Google Chrome.

1. **Инсталация и настройки**

Трябва да се изпълнят следните стъпки:

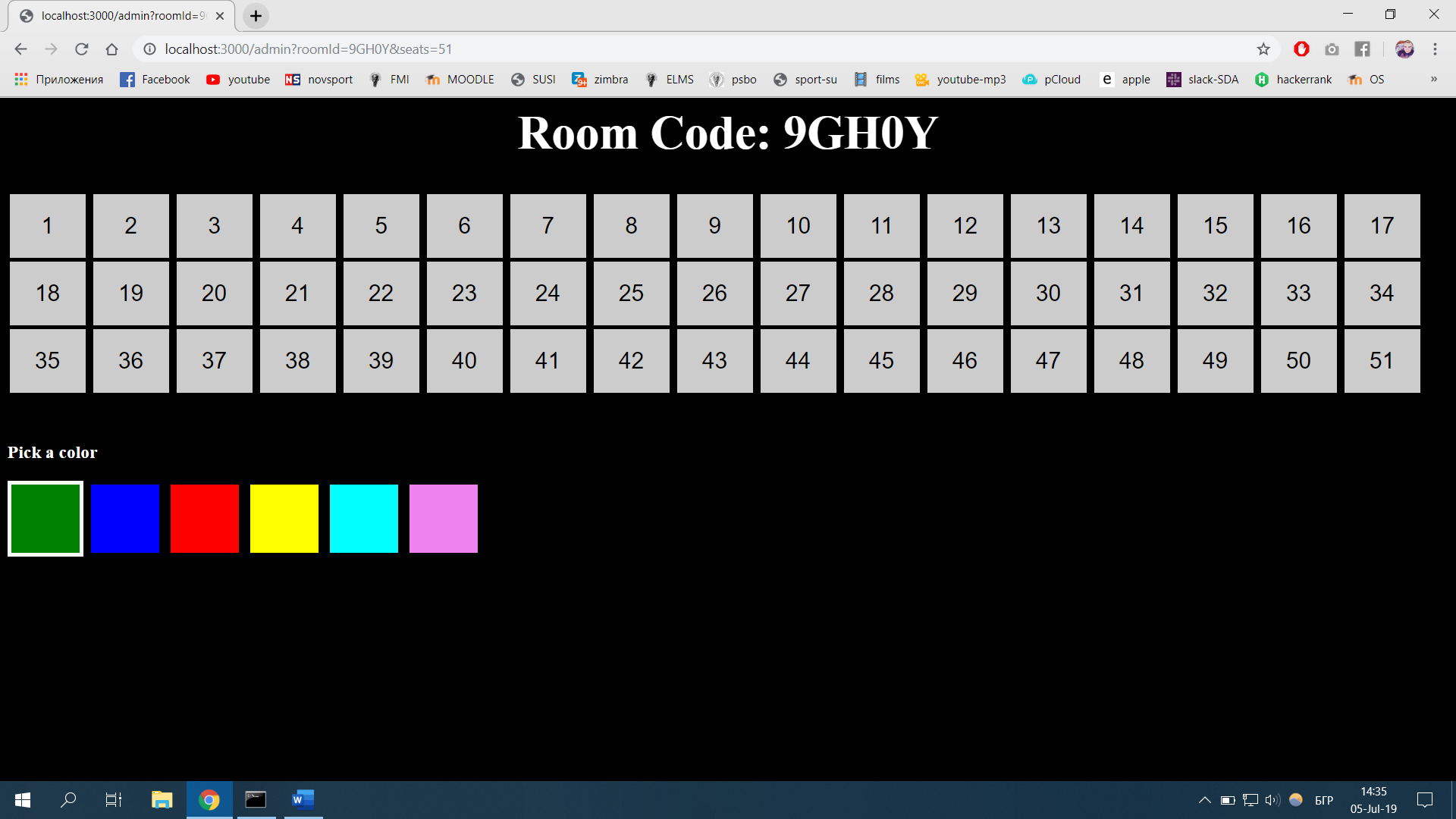
1. Изтегляне и инсталиране на XAMPP.
2. Отваряне на XAMPP Control Panel и стартиране на модулите „MySQL“ и „Apache“.
3. Изтегляне и инсталиране на Node.js.
4. Изтегляне и разархивиране на архива с приложението.
5. Създаване на база данни на адрес <http://localhost/phpmyadmin/> в браузъра с име „colormusic“ и колация „utf8\_general\_ci“. Вмъкване на файл colormusic от папка sql от архива на приложението.
6. При използване на Windows 10 да се отвори Command Prompt и да се отвори папката с файловете на приложението. След това да се въведе командата „node server.js“, за да се стартира сървъра и в браузъра да се отвори адресът: <http://localhost:3000> за стартиране на приложението.
7. **Кратко ръководство за потребителя**

При стартиране на приложението се показва заглавна страница, на която всеки потребител може да влезне в някоя стая чрез въвеждане на коретно roomID и номер на място съответно в полетата „room id“ и „seat“ и накрая да кликне върху бутона „Join“. Също така всеки потребител може и да създава стая чрез въвеждане на брой места от 1 до 100, според това колко седалки желае да има в стаята и след това да кликне върху бутона „Create new room“. По този начин потребителят става администратор за конкретната стая.

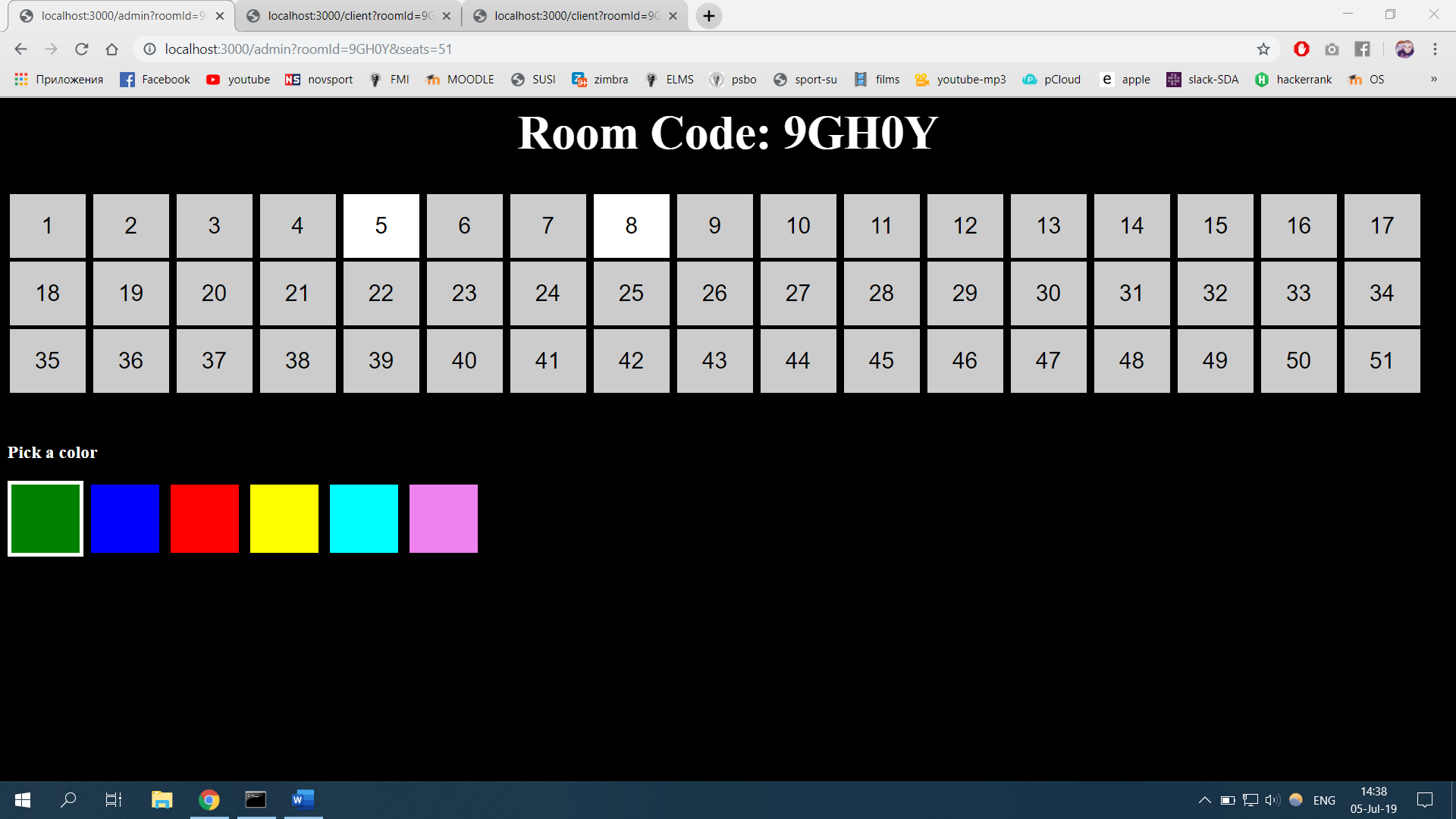


Фиг.1 – Начална страница.

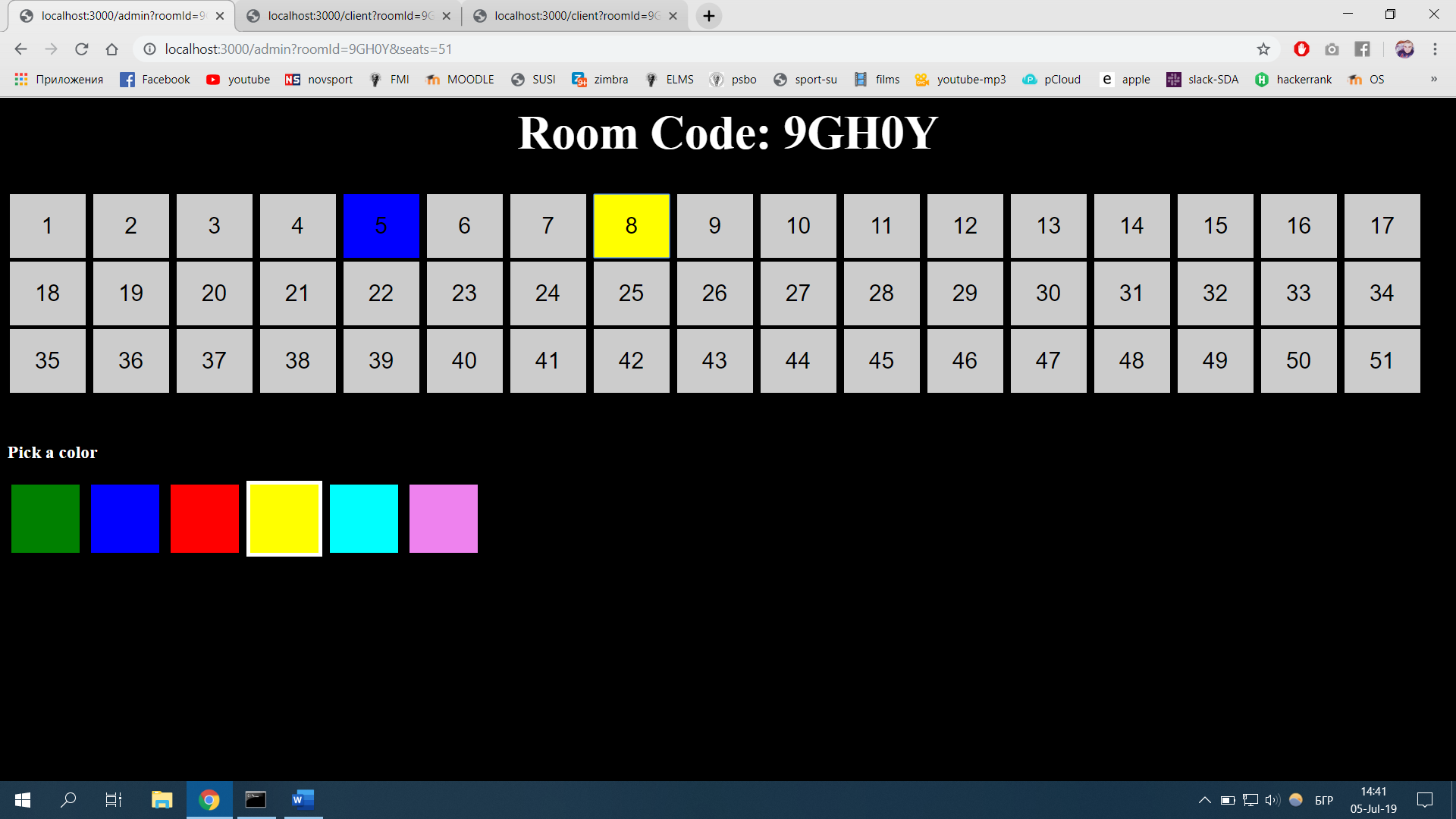
След като даден потребител създаде стая, той се препраща към страница, в която най-отгоре се изписва кодът на стаята, чрез който клиентите могат да се логват в тази стая. Под него са визуализирани номерата на всички седалки(бутоните първоначално са неактивни) в стаята и когато някой клиент влезне в стаята, се активира бутончето за мястото му в стаята(оцветява се в бял цвят). Под номерата на седалките се намират бутони в шест цвята, като при кликане върху определен цвят и после върху бутон за номер на мястото на клиент, фонът на екрана на клиента се оцветява в съответния цвят, както и бутончето при админа за мястото на клиента в стаята се оцветява в същия цвят. След като клиент напусне стаята, бутончето за мястото му в стаята става отново неактивно и се оцветява в сив цвят като останалите неактивни бутони.



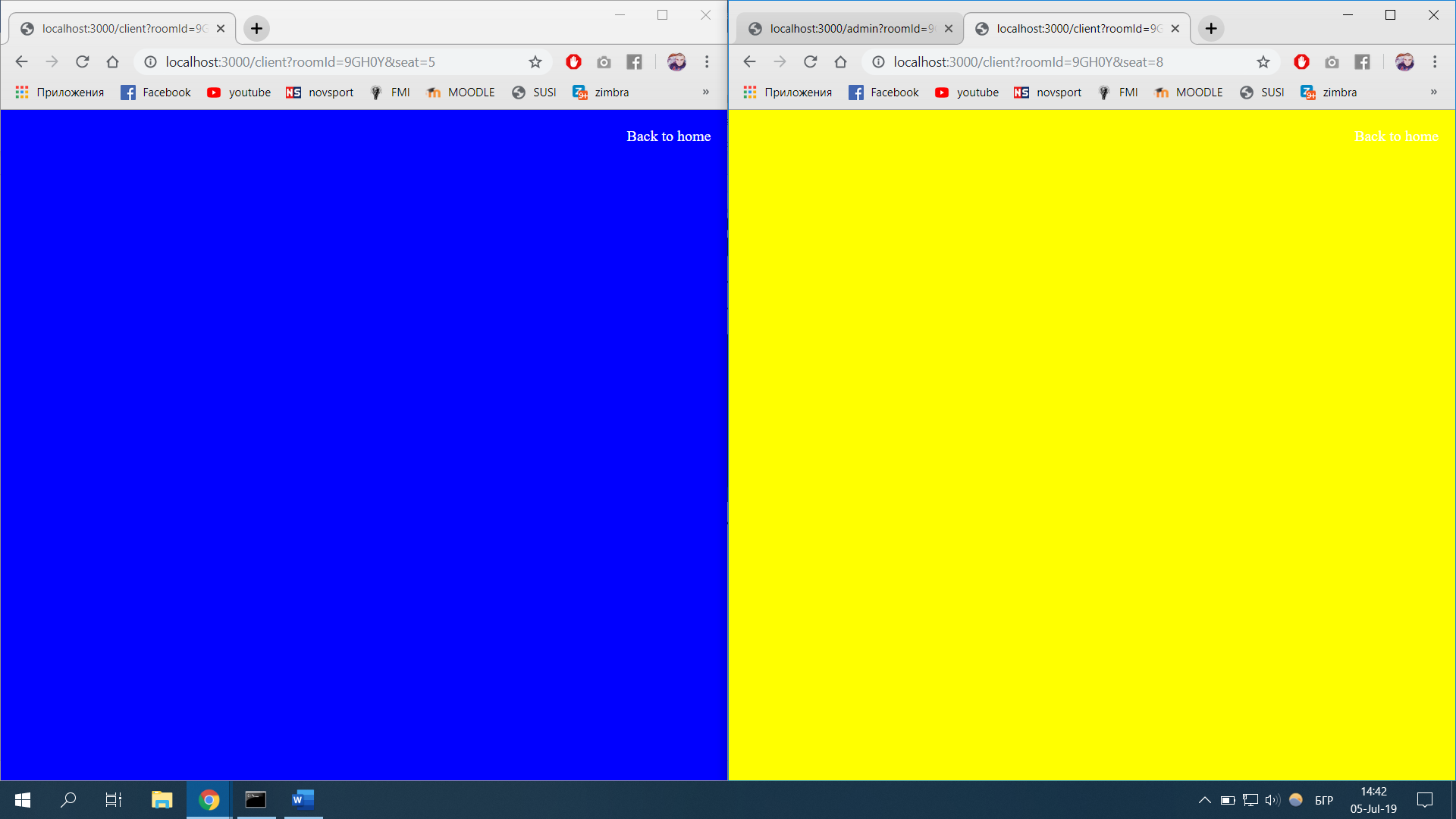
Фиг.2 Създадената стая без наличие на клиенти.



Фиг.3 Създадената стая с наличие на клиенти на места №5 и №8 без зададени цветове за фон на клиентите.



Фиг.4 Създадената стая с наличие на клиенти на места №5 и №8 със зададени цветове за фон на клиентите.



Фиг.5 Страниците съответно на клиент на място №5 и клиент на място №8.

1. **Примерни данни**

Примерни данни за създаване на стая от началната страница в полето „seats 1-100“ могат да бъдат числа от 1 до 100. При грешен вход се извежда съобщение за грешка.

Примерни данни за логване във вече съществуваща стая:

* За полето „room id“ трябва да бъде въведен 5-символен код на съществуваща стая като напр. „9GH0Y“. При грешен вход за несъществуваща стая с такова room id се извежда съобщение за грешка.
* За полето „seat“ трябва да бъде въведен номер на мястото, на което иска да седне клиента, като въведеният номер може да бъде от 1 до максималния брой на местата в стаята. При грешен вход се препраща към страница, цветът на фона на която никога няма да може да бъде променен от администратора.

1. **Описание на програмния код**

**Структурата на кода е следната:**

**config/** - съдържа файл db.json за конфигурация на базата данни - логика за свързване към нея и съответните данни, нужни за това – host, user, парола и име на базата.

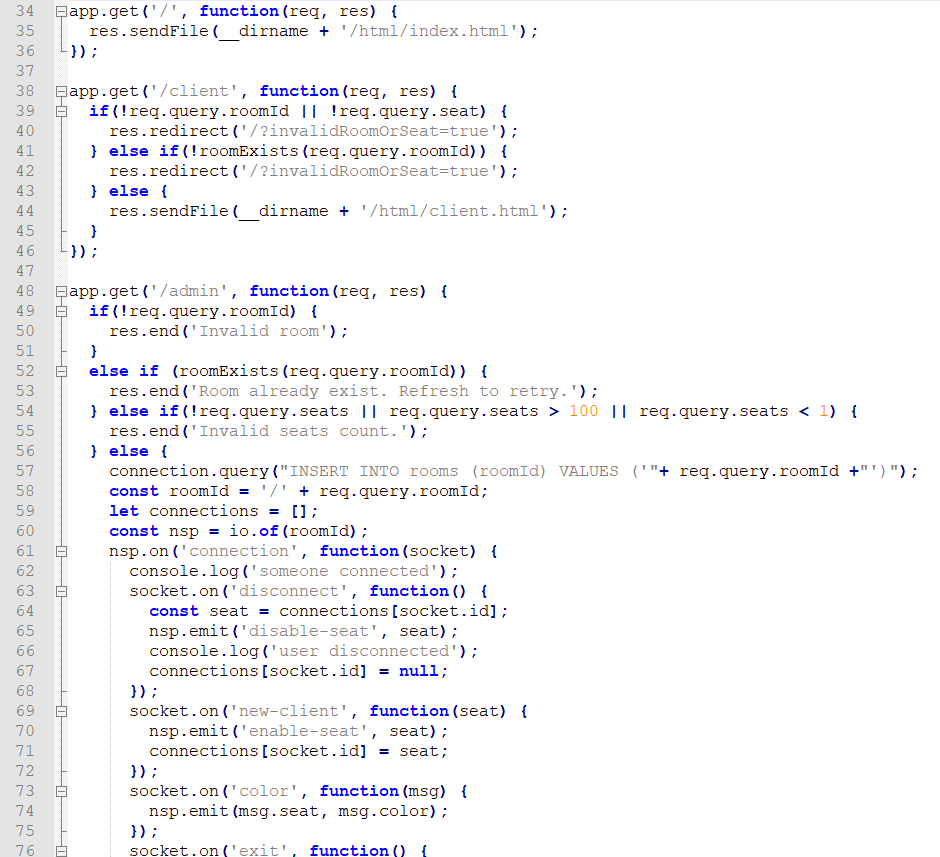
**html/** - съдържа файловете „index.html“(начална страница, която се зарежда първоначално при достъп от ново устройство в системата), „admin.html“(след като даден клиент създаде стая с „n“ на брой места се препраща към тази стая, от която може да променя цветовете на фоновете на всеки един от свързаните към тази стая клиенти) и „client.html“(всеки, който се е присъединил към дадена стая след въвеждане на roomId и място в самата стая – очаква осъществяването на промяна на цвят от администратора).

**public/** - съдържа .css файловете, чрез които се изгражда интерфейса за .html файловете, както и изображението на началната страница на приложението.

**sql/** - съдържа файл colormusic.sql, чрез който се създава таблицa в базата данни, където се пази roomID-то за всяка стая.

- **package.json** и **package-lock.json** са файлове, който са нужни на node package manager да изтегли dependancies на проекта.

- файл **server.js** преставлява файл, който реализира сървърната логика. На долната снимка е показано извършването на http комуникация. Сървърът приема get request-и като за всеки от поддържаните requesti има handle функции, които обработват съответната заявка.



Фиг.6 – Част от кода от файла server.js

1. **Приноси на студента, ограничения и възможности за бъдещо развитие**

Чрез създаването на този проект се внася малка доза забавление по време на обучението на студентите във ФМИ по начин, по който никой не е очаквал. Цветомузиката е една доста интересна идея за излизане за момент от дългото и понякога скучно обучение и принасяне в света на емоциите и забавлението. Ограниченията са основно в липсата на музика, за да не се нарушава спокойствието и тишината в самия факултет. Възможности за бъдещо развитие бих предложил добавяне и на музика, но няма да е предназначено за ФМИ, ами за някое място за забавление, като всички клиенти да се свързват със смартфони и за всеки различен цвят да има различна музика.

**10. Какво научих**

Като за първоначално сблъскващ се с уеб технологиите в този курс без предишен опит научих, че щом човек има желание да научи нещо ново и полезно за бъдещата си кариера, ФМИ е едно доста добро място, където може да се намерят знания и помощ при необходимст в обучението.

От този проект научих как си комуникират клиент и сървър, как се прави връзка с базата данни, как се правят лесни за използване от потребителя интерфейси на приложението и изобщо как се почва от нулата и се изгражда един добър и полезен прототип на система, която при добро бъдещо развитие може да бъде използвана и в по-широка аудитория.

**11. Използвани източници**

За изграждане на проекта е използван голям набор от различни източници на информация, свързани с езиците ***JavaScript*** и ***Node.js***, както и с ***HTML*** и ***CSS***. Сред най-често достъпваните са:

[1] <http://www.w3schools.com/> - за цялостна информация за използваните езици.

[2] <https://hackernoon.com/implementing-a-websocket-server-with-node-js-d9b78ec5ffa8> - за имплементиране на уебсокет сървър с node.js

[3] <http://stackoverflow.com/> - за справяне с най-различни проблеми по кода.

[4] <https://socket.io/docs/rooms-and-namespaces/> - за информация за използване на сокети.