

Имена: Ива Караджова Иван-Асен Чакъров	фн: 81773, 81837	
Начална година: 2021	Програма: бакалавър (КН)	Курс: 4
Тема: 11.2		
Дата: 2022-02-14	Предмет: w17prj_KH_final	имейл: <a href="mailto:karadjovaiva@gmail.com">karadjovaiva@gmail.com</a>

преподавател: доц. д-р Милен Петров

**Предаване:** Задачата се предава в архив с попълнен настоящия документ, проекта/проектите с кодовете, README.txt файл, който описва съдържанието на архива; папка с допълнителни компоненти и използвани ресурси, архива да се казва **9999\_project\_final.zip**. (Успех!). (Редактирайте **маркираните зони** в жълто с коректната информация)

## ТЕМА: Система визуализираща финални надписи

### 1. Условие

Основната задача на системата е да генерира финални надписи и да предлага на потребителя възможност за персонализиране на получените анимации. Характеристиките, които могат да се променят, включват:

- Да се представя текст или вложена уеб страница
- (за уеб страниците) Височината на страницата
- (за текста) Цвят на текста
- Цвят на фона
- Скорост на анимацията
- Източник на музиката – URL или локален файл
- Музика
- Вид анимация – в стил star wars или final credits

При визуализацията на анимацията, проектът предоставя възможност за стопирането ѝ и за премахването на звука ѝ.

Също така системата има функционалност за запазването на вече генерирана анимация, която ще може да бъде достъпвана от друг прозорец.

### 2. Въведение

Уеб сайтът, създаден на проектът, се състои от два основни компонента: форма и анимация. Във формата потребителя може да въведен описаните в условието характеристики и с натискането на един бутон да генерира своите финални надписи в полето за анимация. Там той ще може да слага на пауза анимацията, да я рестартира когато приключи, да пуска и спира избраната от него музика и да я запази за по-късно гледане.

### 3. Теория

В проекта се използват *iframes*. Това представляват вградени рамки, които се използват като *HTML* документ за добавяне на друг *HTML* документ в него. Използва се най-вече в уеб страници или процеси за уеб разработка, за да включва някакво друго съдържание чрез друг източник като реклами на тази уеб страница. В този проект се използват, за да се предостави на потребителя възможност за представяне на други уеб страници в *URL*.

*RESTful API* е уеб приложение, което базирано на принципите на REST и HTTP. Често се използва с медийния тип JSON. Базовите операции, които поддържа, са GET, PUT, POST и DELETE. Системата за генериране за финални надписи предоставя *RESTful API*, чрез който могат да се достъпват, запазват, ъпдейтват и изтриват анимации.

#### 4. Използвани технологии

Използваните технологии включват:

- *HTML5 (HTML, CSS, JavaScript)*
- *PHP*
- *Docker* (при разработката на проекта само)
- *MySQL*
- *Composer*

*HTML5* се употребява за реализация на презентационния слой – това включва генерирането и управлението на единична анимация.

На *PHP* е написана бизнес логиката на проекта, която позволява поддръжката на повече от една анимация. Чрез *RESTful API* презентационния слой комуникира с backend частта и задава функционалността за достъпване, запазване, ъпдейтване и изтриване на анимации.

*MySQL* е необходим за менажирането на база данни, която е необходима за складирането на запазените анимации.

*Docker* се употребява само при разработката на приложението. Благодарение на него бизнес логиката и базата данни да се изпълняват в отделни контейнери, което предоставя по-добра преносимост на приложението и го прави по-лесно за тестване.

#### 5. Инсталация и настройки

##### Настройки

Системата позволява конфигуриране на използвания MySQL server и базата данни. Това може да се случи чрез променяне на файла "*server/api/.env*". В него трябва да се посочат съответно хост, порт, потребител и парола за сървъра, както е демонстрирано отдолу.

*DB\_HOST=DB*

*DB\_PORT=3306*

*DB\_DATABASE=ending\_credits*

*DB\_USERNAME=root*

*DB\_PASSWORD=root*

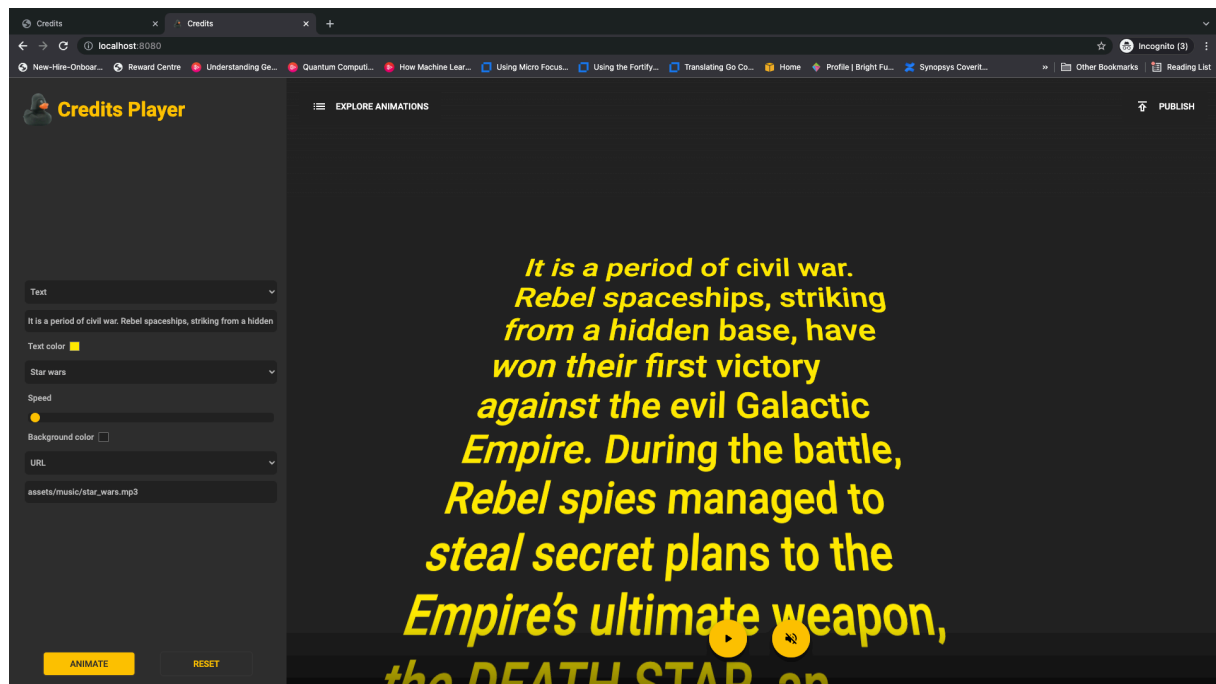
## Инсталация

- 1-ви случай – на машината има инсталирани *Docker* и *Composer*  
Изпълнява се командата `“./run.sh”`. Скриптът стартира два контейнера – един за базата и един за php сървър. Системата е напълно работеща след изпълнението и е достъпна на **localhost:8080**.
- 2-ри случай – няма инсталирани *Docker* и *Composer*
  - а. Създава се база на име **“ending\_credits”** (или каквото име е посочено в **server/api/.env** за **DB\_DATABASE**)
  - б. Изпълнява се командата **“sudo a2enmod rewrite”**

## 6. Кратко ръководство на потребителя

След инсталацията и конфигуриране на системата, потребителя може да достъпи генератора на финални надписи на адрес **localhost:8080**.

Винаги при стартиране на страницата се зарежда анимация по подразбиране, която се вижда на изображението [1].



1. Анимацията по подразбиране, която се зарежда при стартиране на страницата

За да създаде собствена анимация, потребителя трябва да промени информацията във формата от ляво на страницата и да натисне **„Animate”** бутона. Това ще генерира в дясната част на страницата неговите финални надписи. Чрез бутоните **„play”** и **„mute”** може да се контролира и визуализацията на анимацията.

В настройките на анимацията има две възможности за информация, която ще е представена – текст или друга уеб страница. Съответно може да се види как ще изглежда формата на изображения [2] и [3].

Web Page

Animation source

Web Page Height

Star wars

Speed

Background color

URL

assets/music/star\_wars.mp3

2. Задаване на уеб страница за анимация

Text

It is a period of civil war. Rebel spaceships, striking from a hidden

Text color

Star wars

Speed

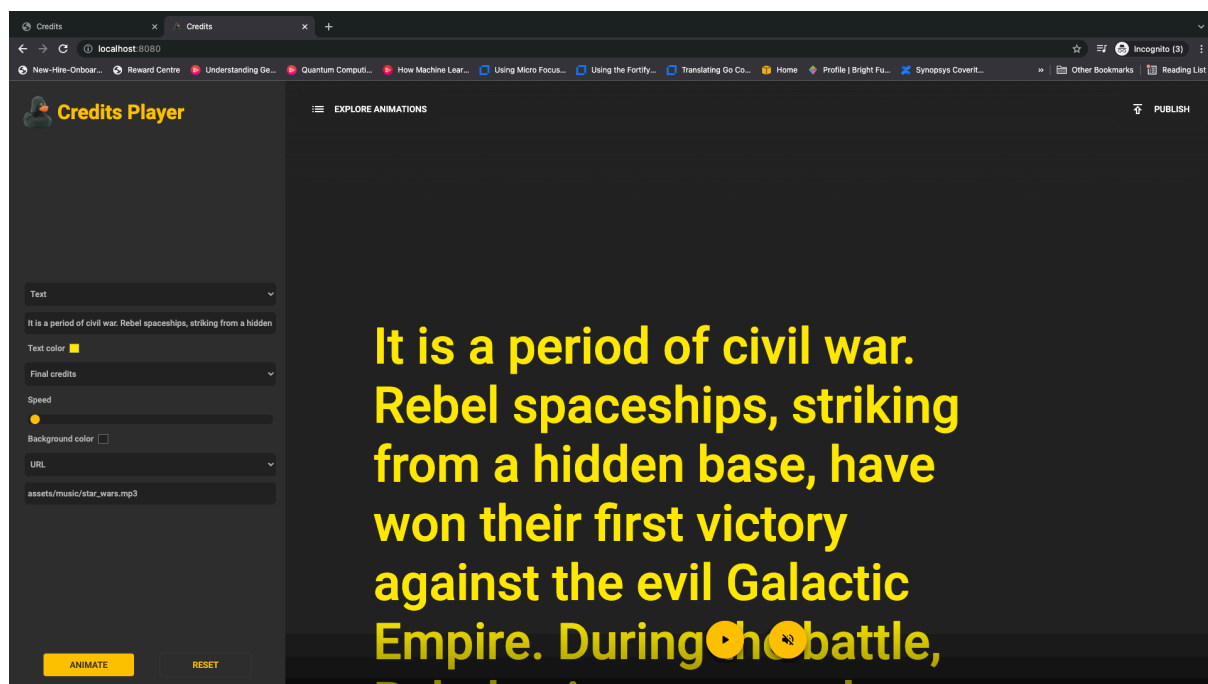
Background color

URL

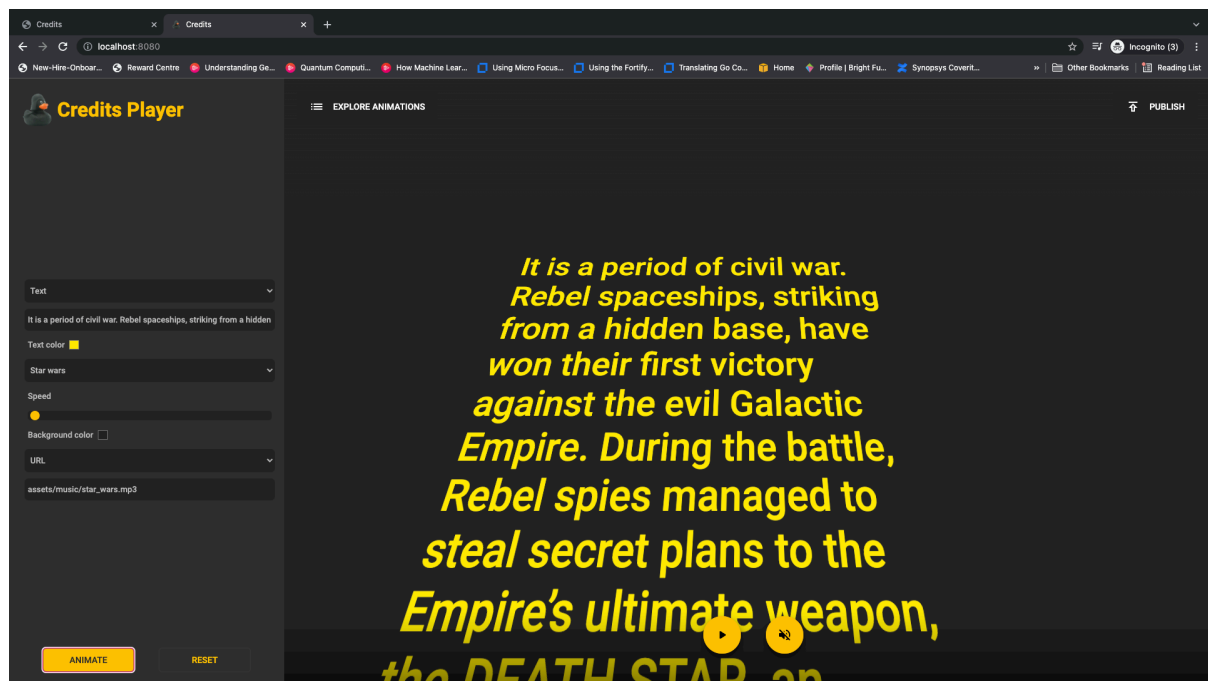
assets/music/star\_wars.mp3

3. Задаване на текст за анимация

Също така има възможност за избор от два вида анимации – Start wars и Final credits. На изображения [4] и [5] могат да се видят разликите между двата стила.

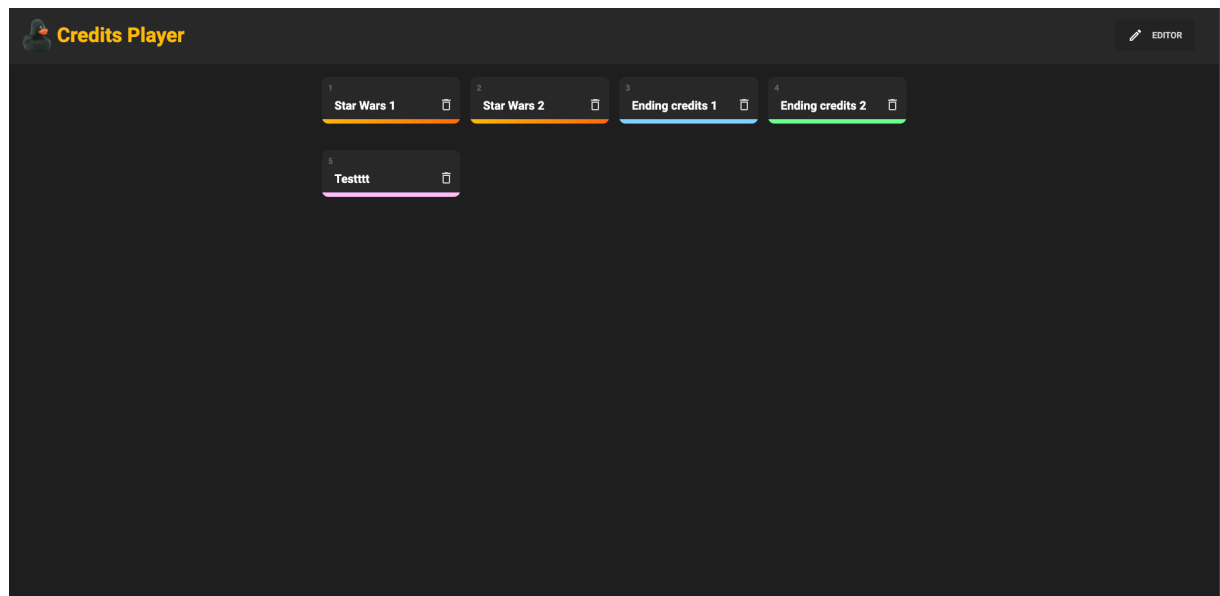


4. Final credits стил анимация



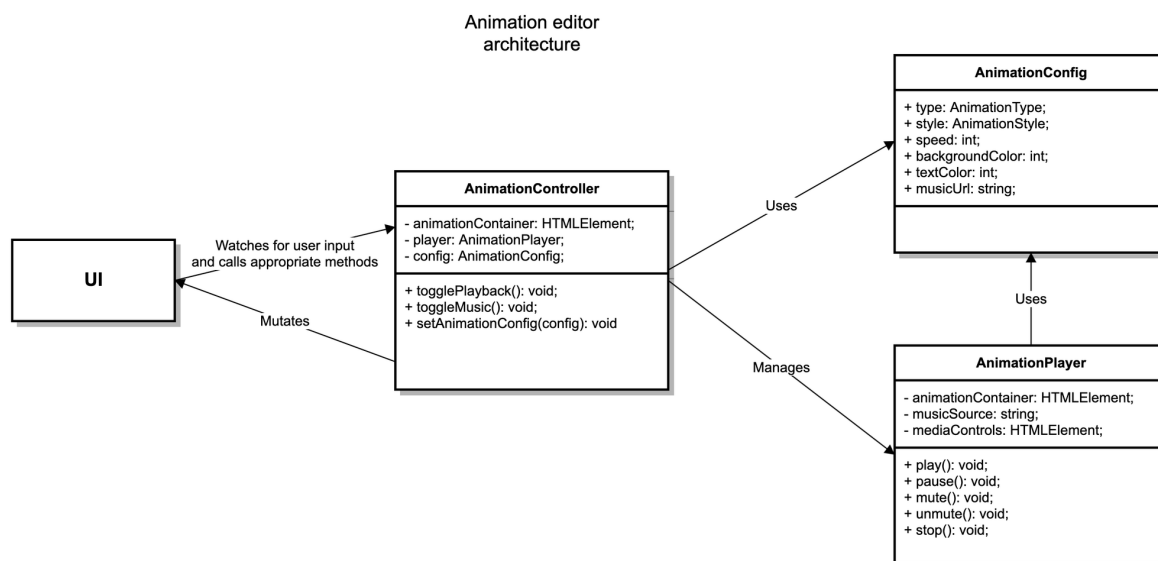
#### 5. Star wars стил анимация

За да се разгледат всички запазени анимации, потребителя трябва да натисне бутона “**Explore animations**”.



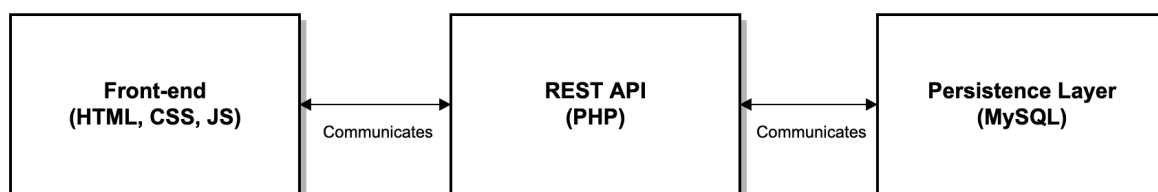
#### 6. Изглед на всички запазени анимации

## 7. Описание на програмния код



### 7. UML диаграм на системата за генериране на анимация

Ъпдейтване на анимацията на страницата се менажира от **AnimationController**. Той следи графичния слой за извиквания нов вход от потребителя. Когато се подаде такъв сигнал към него, той динамично променя html и css на презентационния слой и така се визуализират промените по анимацията.



### 8. Диаграма, изобразяваща комуникацията между бизнес логиката и презентационния слой

Backend-а се състои от *php* сървър и *MySQL* база данни. Основната функционалност, която изпълнява сървъра са CRUD операции върху обекти от тип *AnimationConfig*. Всеки от тези обекти, представлява различна анимация.

Бизнес логиката предоставя един семпъл API интерфейс с 4 метода:

- GET – връща всички анимации от базата
- POST – запазва нова анимация в базата
- PUT – редактира подадената анимация
- DELETE – изтрива подадената анимация

Чрез този REST API се осъществява връзката между потребителския интерфейс и сървърната логика. Когато заявка се подаде на сървъра, той извършва изискваната операция върху базата данни и връща отговор обратно към презентационния слой, който визуализира тези промени.

## 8. Приноси на студента, ограничения и възможности за бъдещо разширение

Бъдещето на проекта е оставено в ръцете на Георги Мечката.



9. Иван-Асен представя проекта на Георги Мечката

## 9. Какво научих

Какво се пише на началните надписи на четвъртия филм от поредицата – ***Star Wars: Episode IV – A New Hope***.

Предал (подпис): .....

/фн, имена, спец., група/

Приел (подпис): .....