**Имена:** Венцислав Терзиев **фн:** 81670

**Имена:** Мартин Великов **фн:** 81696**Начална година:** *2021*  **Програма:** бакалавър, (КН) **Курс: 4**   
**Тема: Анимиране на SVG  
Дата: 2022-02-14 Предмет: w17prj\_KN \_final**

**имейл:** [**martin\_evg@abv.bg**](mailto:martin_evg@abv.bg)**;** [**venciterziev97@gmail.com**](venciterziev97@gmail.com)

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

# ТЕМА: Анимиране на SVG

## 1. Условие

Да се направи система, чрез която да може да се добавят **.svg** файлове. Да може да се настройват параметрите на анимацията и да се съхраняват отделно в базата от данни с цел редакция и запазване на нова версия. Системата да дава възможност да се свалят анимираните .svg файлове.

## 2. Въведение

Този проект предоставя лесна възможност за анимиране на **.svg** изображения. Добавени са примерни изображения в **.svg** формат, с помощта на които може да се покаже функционалността на проекта.

## 3. Теория

Системата има трислойна архитектура:

1. **Презентационен слой** – ***HTML****,* ***CSS****,* ***JavaScript***

**2. База от данни** – ***MySql***

**3. Бизнес логика** – ***Php***

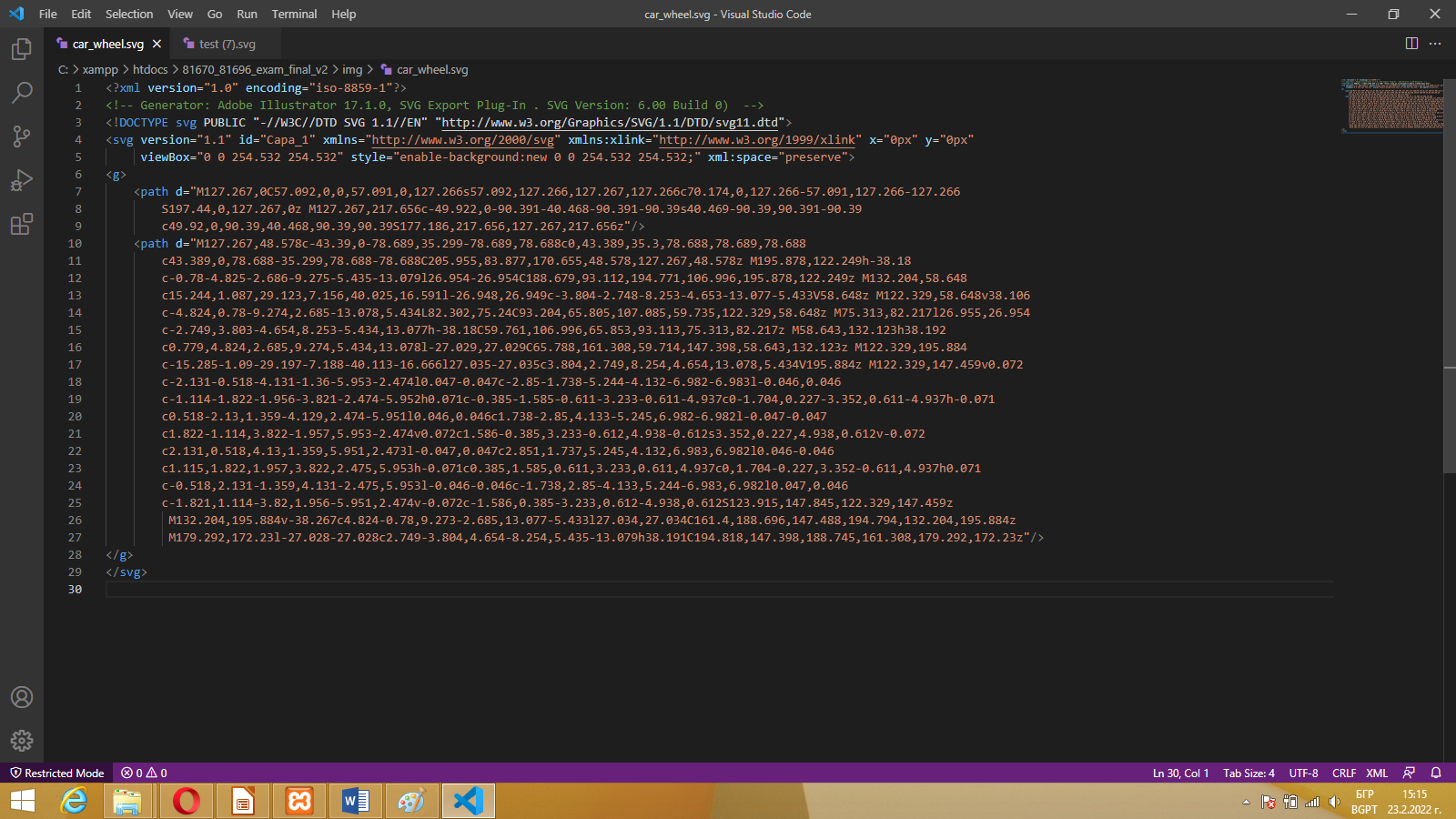
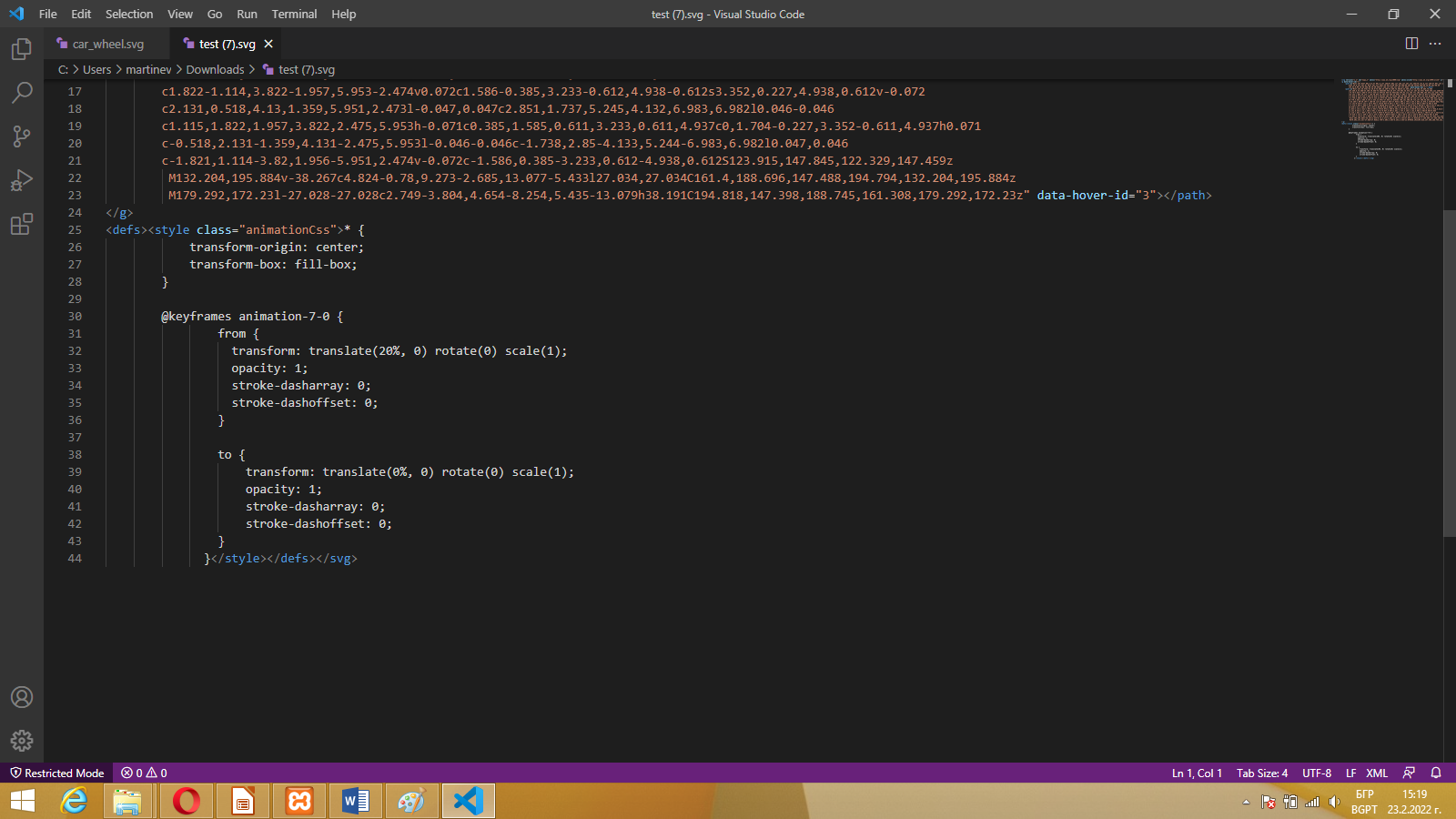
Системата има **една** роля – **потребител**. Всеки потребител има свои .svg файлове, които може да редактира. Направените промени по файловете се записват в базата от данни и могат да бъдат прегледани в по-късен етап. Всички атрибути, които се попълват за анимиране, са всъщност ***CSS стойности***.

Нека вземем файла “**car\_wheel.svg**” от папката “img”.

Записваме стойности например на атрибута **X** съответно **from:** **20%** ; **to:** **0%.**

Запазваме вече анимираната картинка и да видим кода:

Виждаме, че системата добавя един **style** таг и в него упоменава чрез CSS анимацията, която сме задали като стойности на атрибути в системата (в този случай имаме *translate from 20% to 0%*).

Част от кода на анимираната картинка

Код на оригиналната картинка

***Функционалностите и мерните единици за полетата на атрибутите:***

**X** – *транслация по абсцисната ос X* – в проценти (**%**) или в пиксели (**px**);

**Y** – *транслация по ординатната ос Y* – в проценти (**%**) или в пиксели (**px**);

**Opacity** – *степен на непрозрачност* – реално число в граници **[0;1]**;

**Rotate** – *ротация (завъртане)* – в градуси (**deg**) или в радиани (**rad**);

**Scale** – *мащабиране* – неотрицателно число ( **>=0** );

**Stroke** – *щриховане (за симулиране изписването на буква)*

**Stroke-dasharray** – *създава тирета (щрихи) и* *указва разстоянието между тях (колкото е по-голяма стойността, толкова повече разстояние ще има между отделните тирета )*  – може да приема всякакви единици за дължина, включително и безразмерни стойности (2em, 15%, 150…), които се превръщат в относителни единици за дължина в координатната система, зададена от полето за изглед;

**Stroke-dashoffset** – *дефинира местоположението по SVG path-a, откъдето ще започне изчертаването (колкото е по-голяма стойността, толкова по-далеч от началото на path-a ще се защрихова той)* – може да приема стойности в **проценти** (**%**), или **числова стойност** (100, 25%...). Единици (**em**, **px**) **НЕ СА** задължителни. **Безразмерните числа се превръщат в пикселни единици! Процентните единици са относителни към процента от текущия изглед!**;**Ease** – *специфицира скоростната крива на анимацията*:

* ***ease-in*** – *определя ефект с бавно начало;*
* ***ease-out*** – *определя ефект с бавен край;*
* ***ease-in-out*** – *определя ефект с бавно начало и бавен край;*
* ***linear*** – *определя ефект с константна скорост от начало до край.*

*Основни скоростни криви*

**

**Duration** – *времетраене на анимацията* – в секунди (**s**) или в милисекунди (**ms**);

**Delay** – *забавяне изпълнението на анимацията* – в секунди (**s**) или в милисекунди (**ms**);

**Repeat** – *брой повторения на анимацията* – цяло число **> 0** или **infinite (безкрайно повторение)**;

## 4. Използвани технологии

## Използваните технологии в този проект са:

## Front-end: *HTML, CSS, JavaScript*

**Back-end:*****PHP***

**Database**:***MySQL***

***Версии на технологиите, на които е тествана системата:***

**ОС***: Windows 8.1, Edition: Professional 64-bit*

**XAMPP**: *7.4.27*

**MariaDB**: *10.4.22*

**PHP**:*7.4.27 (VC15 X86 64bit thread safe) + PEAR*

**Server**: *Apache/2.4.52*

**ОС**: *Windows 11, Edition: Professional 64-bit*

**XAMPP**: *Version: 7.4.5*

**MariaDB**: *10.4.11*

**PHP**: *7.4.5 (VC15 X86 64bit thread safe) + PEAR*

**Server**: *Apache/2.4.43*

**ОС**: *Windows 8.1, Edition: Professional 64-bit, Windows 11, Edition: Professional 64-bit*

**XAMPP**: *8.1.2*

**MariaDB**: *10.4.22*

**PHP**: *8.1.2 (VC15 X86 64bit thread safe) + PEAR*

**Server**: *Apache/2.4.52*

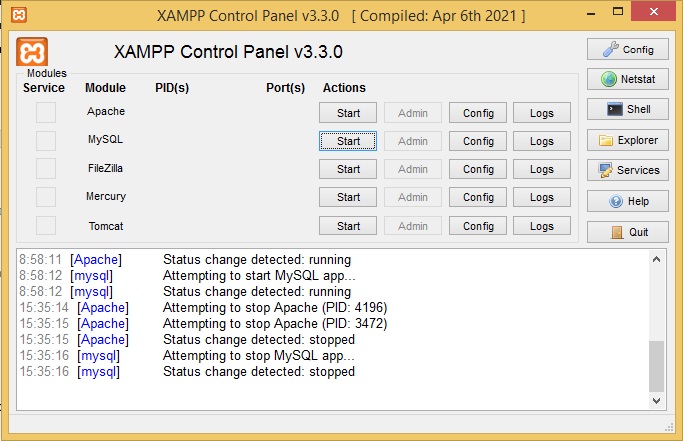
***Системни изисквания:***

* ***64 MB*** RAM
* ***750 MB*** свободно дисково пространство
* ***Windows 7, Windows 8, Windows 10*** или по-нова ОС

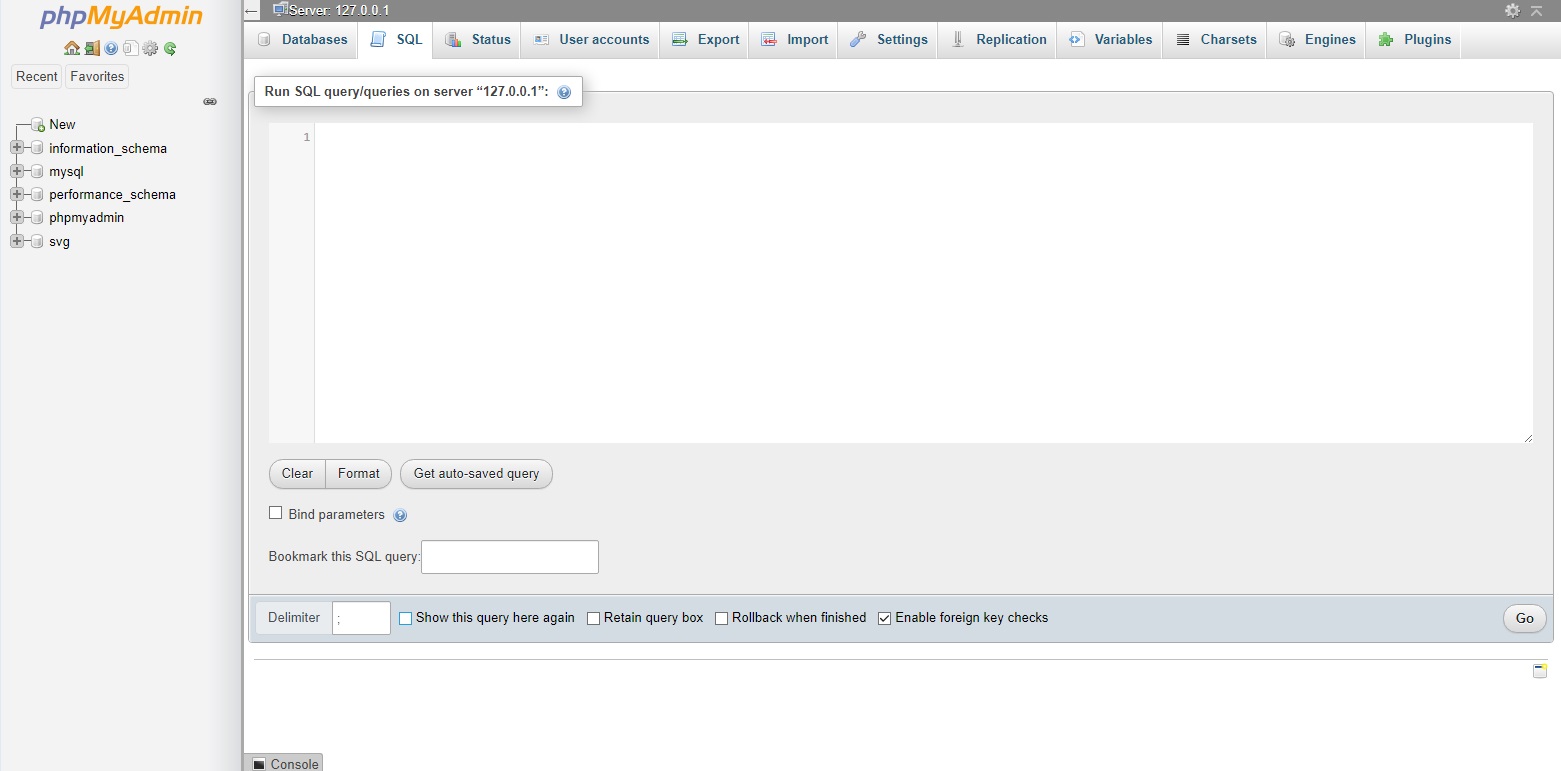
## 5. Инсталация и настройки

## *За да се инсталира проекта е необходимо да имате XAMPP.*

1. Стартирате XAMPP - Кликвате върху **Start** бутона за **Apache** и за **MySQL**.

****

2. В **phpMyAdmin** качвате базата от данни (скрипта “**createDb.sql**” в папката “config”).

****

3. В браузър, през **localhost**, отваряте проекта, регистрирате се и влизате в системата, като се логнете.

## 6. Кратко ръководство на потребителя

Инструкциите за употреба (“**userguide\_svg\_animation\_tool.pdf**”) можете да намерите в папката на проекта.



## 7. Примерни данни

За първите пет примерни анимации използваме файла “**car\_wheel.svg**” от папката “img“.

***1. Транслация по абсцисната ос***

Ако искаме да транслираме цялата картинка (а не отделни нейни слоеве), избираме **svg** от секцията SVG Tree.

Ако искаме да анимираме отделни слоеве на дадена .svg картинка, от секцията SVG Tree избираме желания path (Важи и за долните примери!).

За **svg**, в секцията Animate, записваме стойности на следните атрибути:

***X***: **from**: **-100%**, **to**:**100%**

С горепосочените настройки ще видим картинката как се движи **хоризонтално**, от крайната лява до крайната дясна част на прозореца за визуализация.

***2. Транслация по ординатната ос Y***

Избираме **svg** от секцията SVG Tree.

За **svg**, в секцията Animate, записваме стойности на следните атрибути:

**Y**: **from**: **-1000px**, **to**:**1000px**

С горепосочените настройки ще видим картинката как се движи **вертикално**, от крайната горна до крайната долна част на прозореца за визуализация.

***3. Ротация***

Избираме **svg** от секцията SVG Tree.

За **svg**, в секцията Animate, записваме стойности на следните атрибути:

**Rotate**: **from**: **0deg**, **to**: **360deg**

С горепосочените настройки ще видим картинката как се завърта около центъра си един пълен оборот.

Ако искаме например картинката да се върти N пъти или безкрайно дълго, на атрибута **Repeat** записваме съответно **N** или **infinite** (приложимо е за всички анимации!).

***4. Непрозрачност (Opacity)***

Избираме **svg** от секцията SVG Tree.

За **svg**, в секцията Animate, записваме стойности на следните атрибути:

**Opacity**: **from**: **0**, **to**: **1**

С горепосочените настройки ще видим картинката как плавно преминава от напълно прозрачна към напълно непрозрачна.

***5. Мащабиране (Scale)***

Избираме **svg** от секцията SVG Tree.

За **svg**, в секцията Animate, записваме стойности на следните атрибути:

**Scale**: **from**: **0**, **to**: **1**

С горепосочените настройки ще видим картинката да се появява като безкрайно малка (в далечината) и постепенно придобиваща нормалните си размери (приближаваща се).

За следващите две примерни анимации добавяме файла “**svg\_tool.svg**” от папката “img”.

***6. Щрих (Stroke) – симулация на изписване на буквите***

Избираме **svg** от секцията SVG Tree.

За **svg**, в секцията Animate, записваме стойности на следните атрибути:

***stroke-dasharray***: **from**: **200**, **to**: **200** (т.е. имаме сравнително голямо разстояние между тиретата (толкова е голямо, че дори ще имаме само едно тире – самата буква), което не се променя по време на анимацията)

***stroke-dashoffset***: **from**: **200**, **to**: **0** (т.е. започваме от края на path-а и стигаме до началото)

С горепосочените настройки ще видим симулирано ръчно изписване на буквите от картинката.

***7. Щрих (Stroke) – щриховане границите на един слой – комбинация от атрибути***

Избираме например **4-ти path** (4-та буква) от секцията SVG Tree.

За **4-ти path**, в секцията Animate, записваме стойности на следните атрибути:

***stroke-dasharray***: **from**: **0**, **to**: **30**

***duration:*** **4s**

***ease:*** **linear**

***repeat:*** **3**

***delay:*** **1500ms**

*Така анимацията ще трае 4 секунди, с константна скорост, 3 повторения и забавяне след натискане на бутона “****Run Animation****” с 1500 милисекунди*

За по-нагледно представяне на примерните анимации вижте инструкциите за употреба (файлът, упоменат по-горе)!

## 8. Описание на програмния код

## Структурата на програмния код се състои от 3 части – протребителска интеракция със сайта – HTML, CSS, JS, бизнес логика – PHP и база данни – MySql.

## В index.html се съдържа основната страница на платформата, която визуализира различни части от сайта, в зависимост от това дали потребителят е логнат или не. В страницата се зареждат два .css файла – index.css и main.css като единият отговаря за стилизацията на header-a, а другият – за стилизацията на частта от сайта, който е за анимиране на .svg файлове.

## Файловете login.css и register.css стилизират съответно страниците login.html и register.html. JS файлa index.js се грижи за проверката дали потребителят е логнат или не, а файлът main.js - за интеракцията със страницата, в която се анимират .svg-тата.

## Файлът login.js се грижи за влизането на потребителите в системата, а файлът register.js – за регистрирането на потребителите в системата. PHP файловете се грижат за комуникацията между JavaScript-a и базата от данни.

Логин в системата



## C:\Users\martinev\AppData\Local\Microsoft\Windows\INetCache\Content.Word\erm.jpg

Модел същност-връзка на базата от данни

## C:\Users\martinev\AppData\Local\Microsoft\Windows\INetCache\Content.Word\database.jpg

Създаване на двете таблици

Връзка с базата от данни



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

## Чрез създаването на този проект усъвършенствахме уменията си за работа в екип.

Всеки модул от проекта разработихме заедно. По този начин много бързо намирахме решения за възникналите проблеми и умело отстранявахме грешките, които се появяваха в процеса на разработка на отделните компоненти. В бъдеще могат например да се добавят още възможности за промяна графиката на скоростта на възпроизвеждане на анимацията. Може да се добави например и кубична крива на Безие, чрез която потребителите биха могли ръчно да настройват параметрите на скоростната крива и да придобият още по-голяма свобода при анимирането на векторни картинки. Поради факта, че анимирането е всъщност добавяне на CSS код, системата предоставя лесна разширяемост на компонентите си.

Системата дава възможност да се разширява откъм атрибути на анимацията дотолкова, доколкото позволява технологията CSS, т.е. огромна възможност за разширяемост!

## 10. Какво научих

## Посещаването на курса по WEB технологии във ФМИ осезаемо разшири познанията ни, свързани с WEB програмирането, и множеството от разнообразни технологии, които се използват в разработването на системи и приложения в тази сфера. Реализирането на този проект е ключов момент за задълбочаване на знанията ни за базовите технологии, които се използват за разработка в WEB, а именно HTML, CSS, JavaScript, PHP и MySQL. В края на курса вече имаме представа и за ауторизацията, автентикацията, използването на сесии и комуникацията между отделните компоненти, и за това колко са важни в съвременното програмиране в сферата на WEB.

## 11. Използвани източници

[1] [Chris Coyier](https://css-tricks.com/author/chriscoyier/), “How SVG Line Animation Works”, <https://css-tricks.com/svg-line-animation-works/>, обновен на Aug 31, 2021  
[2] [Sara Soueidan](https://css-tricks.com/author/sarasoueidan/),“A Guide to SVG Animation (SMIL)”, <https://css-tricks.com/guide-svg-animations-smil/>, обновен на Aug 29, 2018

[3] [https://developer.mozilla.org/en-US/docs/Web/API/CSSStyleSheet/insertRule](https://developer.mozilla.org/en-US/docs/Web/API/CSSStyleSheet/insertRule?fbclid=IwAR32T_LBDV1MMGZGwdXIIcQIzA8TRqwL83xrtX0UQHYyqVwMoKJoJC75oEs)

[4] ”PHP Documentation”, <https://www.php.net/docs.php>

Предал (подпис): ………………………….

/*81670, Венцислав Терзиев, КН, 1*/

Предал (подпис): ………………………….

/*81696, Мартин Великов, КН, 1*/

Приел (подпис): ………………………….

/доц. *Милен Петров*/