

Тема: Система за създаване и провеждане на тестове, включително видео тестове/въпроси

Предмет: Web технологии (w15ed)

Изготвили: Павел Цанов, фн:81 474, phcanov@uni-sofia.bg и Петър Ангелов, фн:81 412, peturpa@uni-sofia.bg

Лектор: Милен Петров, година: 2020/21, зимен семестър

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 студентите, ограничения и възможности за бъдещо развитие**
- 10 Какво научихме**
- 11 Списък с фигури и таблици**
- 12 Използвани източници**

1 Условие

Системата поддържа две основни функционалности, създаване на тестове и проверждане на тестове.

Нека започнем с първата основна роля на приложението и тя е "Създател на тестове". Той може да създава нов тест и за всеки тест ще добавя произволен брой въпроси. Въпросите за сега биват 2 типа: Първи текстови с произволен брой отговори и произволен брой верни отговори, и втори, с видео въпрос, при който потребителят първо трябва да изгледа видео и след това да отговори на добавените въпроси.

Втората роля е обикновен потребител, който ще има възможност да си избира тестове, като най-вероятно тестовете ще са определени по групи, дали предварително зададени или динамично добавящи се от (Админ/Създателя). И съответно ще може да проведе избрания тест, като накрая ще получи оценка(% или точки).

Допълнителни функционални изисквания:

- Експорт на тест към CSV файл
- Импорт на тест от CSV файл
- Сваляне на теста локално
- Страница с отговорите

2 Въведение

Системата е добре известна и срещана във всякакви институции, като учебни заведения, държавни заведения и частни компании. Тестовата система помага за оценяване на хората в изброените заведения. Подобна функционалност има разбира се Мудъл. Тази система се различава от повечето с допълнението си, че предоставя видео въпроси, които провеждащия теста потребител трябва да гледа преди да отговори на последващите въпроси.

Най-важните изисквания на системата са:

1. Създаване на нов тест, към който се добавят произволен брой въпроси, независимо дали видео въпроси или нормални тестови въпроси
2. Провеждане на избран тест
3. Резултати за всеки проведен тест
4. Експорт/ Импорт към/от CSV файл

3 Теория

Архитектурата на системата е трислойна. Презентационния слой се състои от HTML/JS/CSS, по средата имаме бизнес логика PHP, която обработва информацията, от/към клиента. Данните се съхраняват в MySQL бази данни.

4 Използвани технологии

Тествано на конфигурации:

- Сървър: Microsoft Windows 10 Education [Version: 20H2, OS build: 19 042.746]
- Сървърен език: PHP Version 7.4.13 (x64)
- Уеб сървър: Apache/2.4.46 (Win64) OpenSSL/1.1.1h PHP/7.4.13
- БД сървър: MariaDB [Version: 10.4.17-MariaDB - mariadb.org binary distribution]
- HTML5/CSS/JS

5 Инсталация и настройка

Предварителни изисквания са да има работещи уеб сървър и БД сървър.

Съдържание на архива:

81412_81474_project/ - Всичко изброено надолу се съдържа в тази папка

/db/81412_81474_init_seed.sql – Скрипт за инициализиране на нужните таблици и за попълване на тестови данни

video_tests/ - Папката, която съдържа сърцето на проекта

video_tests/index.php – Началният файл

video_tests/db/db_connection.php – Файл за конфигурация и инициализиране на връзката с базата данни

video_tests/create_test.html – ХТМЛ файл за създаване на тест

video_tests/process_tests.php – Файл, който приема информацията при създаване на теста обработва я и я записва в базата

video_tests/questions.js – Помощен файл за динамично създаване на форми при създаване на тест

video_tests/result.php – Файл извличащ данните за резултатите

video_tests/results.php – Страница с резултатите

video_tests/edit_tests.php – Страница за администрация на тестовете – създаване, изтриване, вмъкване на тест

video_tests/delete.php – Помощен файл за изтриване на тест

video_tests/import.php + video_tests/export.php – Помощни файлове за експорт и импорт на тестове към CSV файл

video_tests/test_template.php – Файл за генериране на тест шаблон, при провеждане на тест

img/ – Папка съдържаща изображенията нужни на приложението

css/ – Папката за стилизацията на приложението

Първа стъпка – Настройване на ДБ сървъра:

Използвайте готова или създайте схема с име „**project_video_tests**”. За нея създайте потребител с име **test** и парола **1234**, като му дадете привилегии минимум за: **Insert**, **Delete** и **Update** или използвайте потребителя root.

Втора стъпка – Създаване на нужните таблици + тестови данни:

1. Импорт на `/db/81412_81474_init.sql` файла в схемата която използвате, това ще създаде нужните таблици: **tests**, **questions**, **answers**, **results**
2. (Незадължителна) Импорт на `/db/81412_81474_seed.sql` файла, който съдържа тестови данни. Тази стъпка може да се пропусне, тъй като основна функционалност е да се създаде тест, а след като се създаде тест, вече ще имаме тестови данни, с които може да проведем теста.

Трета стъпка – Конфигуране на връзката с ДБ сървър:

Отворете файла `db_connection.php` Намиращ се в главна папка на проекта и проверете дали следните полета отговарят на вашата конфигурация:

```
DEFINE ('DB_USER', 'test');  
DEFINE ('DB_PASSWORD', '1234');  
DEFINE ('DB_HOST', 'localhost');  
DEFINE ('DB_NAME', 'project_video_tests');
```

Както забелязвате, това са именно тези, които създадохме в първа стъпка, тук може да промените с ваши готови: схема, потребител, а на мястото на localhost, сменете с правилния адрес към вашата база.

Четвърта стъпка – Стартиране:

Поставете папката `video_tests/` на правилното място, например ако използвате XAMPP → `xamp/htdocs/video_tests/`

Отворете браузъра и направете заявка към следният ендпойнт (Ако имате конфигуриран друг адрес, използвайте него, на мястото на localhost) :

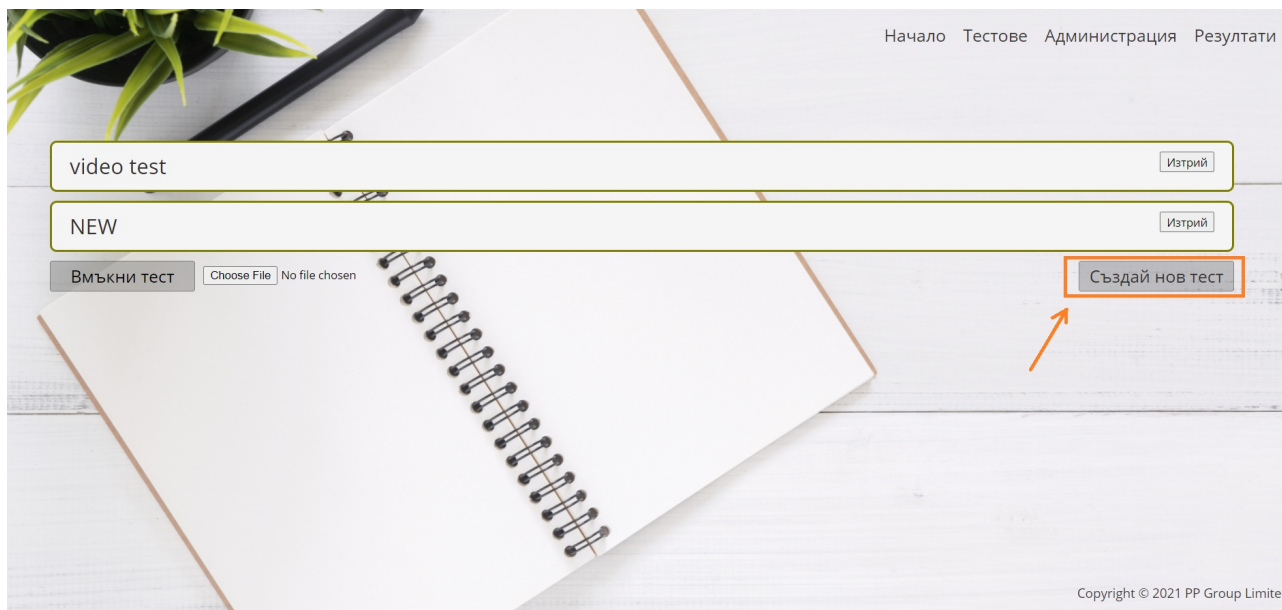
localhost/video_tests/create_test.html

6 Ръководство на потребителя

Потребителският интерфейс е лесен за разбиране и интуитивен, в следващите стъпки ще покажем с илюстрации как може да се създаде един тест и как може да се проведе.

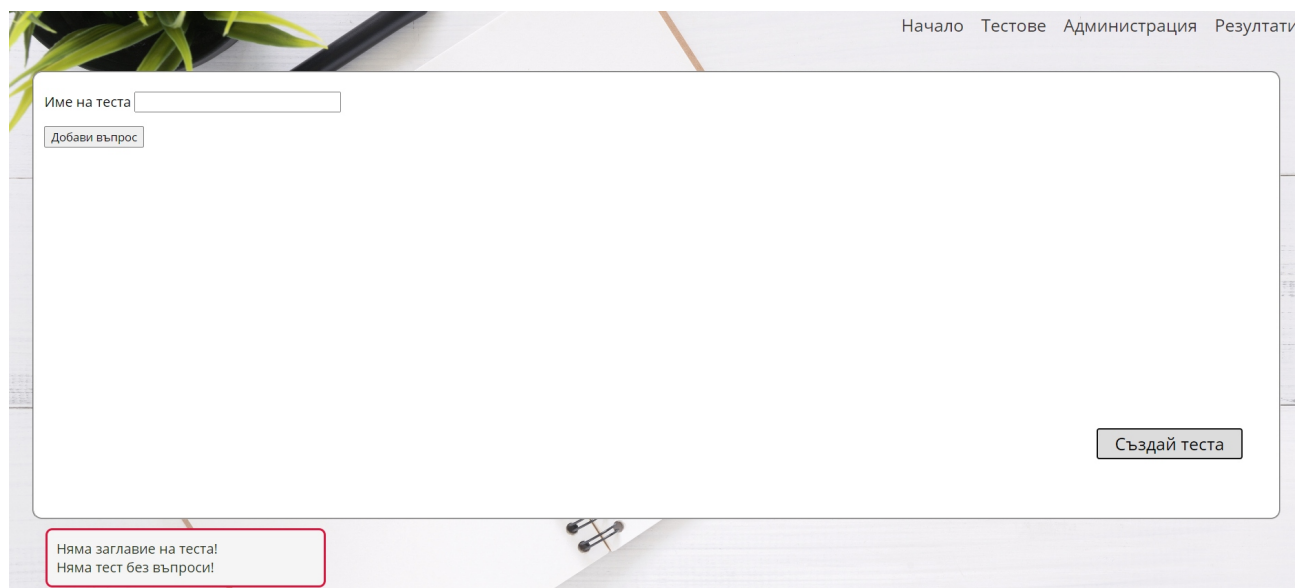
6.1 Създаване

Страницата за създаване е достъпна от две места. Първото място е директно от началният екран, а второто е от раздел `Администрация` (Фиг. 1):



Фиг. 1 Създай нов тест

След кликване, бутонът ще ни заведе до страницата за създаване на тест, на която лесно може да се ориентираме как да добавим нови въпроси и отговори към тях, разбира се има валидации, които ще ни попречат да направим невалиден тест (Фиг.2)



Фиг. 2 Създаване на тест

Създаването, може да стане също чрез вмъкване на тест от CSV формат(Фиг. 1)

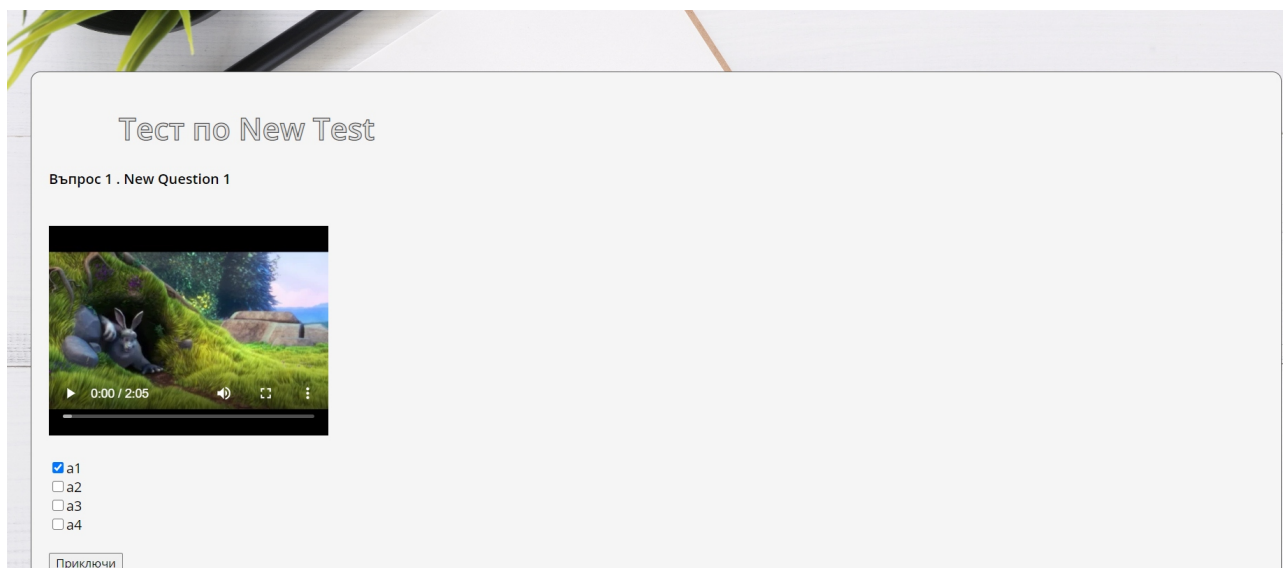
6.2 Провеждане

Страницата за провеждането е също достъпна от две места. Първото място е директно от началният екран, а второто е от раздел `Тестове` (Фиг. 3):



Фиг. 3 Избиране на тест

След избиране на тест, ще бъдем препратени към страницата за решаване на тест (Фиг.4), където можем спокойно да решим теста и да предадем, а след предаване и приключване, получаваме моментен резултат.



Фиг. 4 Провеждане

7 Примерни данни

При изпълнение на вторият скрипт, ще имате зареден примерен тест, с който може да тествате системата на провеждане.

8 Описание на програмния код

Програмният код е възможно най-раздробен на файлове спрямо функционалности, например : *ideo_tests/delete.php*, *video_tests/import.php*

Ясно наименовани файлове, отнасящи се за определена функционалност.

9 Приноси на студентите, ограничения и възможности за бъдещо развитие

При разделяне на работата, е целено маскимально разнородно разпределение на задачите с цел, всеки от изготвилите да научи максимално много от използваните технологии и да се запознае с нови проблеми и техните решения. По-конкретно, разпределение е следното:

1. Приносите на Павел се включват основно във функционалност на проекта на създаване, което е свързано с динамичното съставяне на html форми, провеждане и експорт на тестове, изкарване на резултат от проведен тест, съответно с файлове - *questions.js*, *test_template.php*, *export.php*, *result.php*.

2. Приносите на Петър към проекта са свързани основно с работата с базите данни, дизайна на архитектурата на проекта, импорт на тестове, както и по-голямата част от изготвянето на документацията. Файлове - *db_connection.php*, *process_tests.php*, *edit_tests.php*, *style.css*, *export.php*.

Някои от ограниченията на проекта са свързани с провеждането на теста и неговите резултати. След приключване, потребителят няма как да разбере грешките си и съответно има единствено резултат.

Възможни са редица подобрения по дизайна, като например динамично уразмеряване(респонсив), което ще позволи ползването на приложението от други платформи като таблети и телефони.

10 Какво научихме

През времето, прекарано в писането на проекта, научихме нови неща като динамично създаване на елементи чрез използване на JavaScript и създаване на връзка от интерфейс до база данни. Придобихме също богат и ценен опит в работата с програмния език PHP, както подобриме и текущия си за езиките HTML, CSS, JS и SQL . Също научихме колко е важно планирането на една система и нейната репрезентация в базата данни.

11 Списък с фигури и таблици

1. Фигура 1 – Създай нов тест
2. Фигура 2 – Създаване на тест
3. Фигура 3 – Избиране на тест
4. Фигура 4 – Провеждане на тест

12 Използвани източници

1. W3schools PHP, <https://www.w3schools.com/php>
2. W3schools CSS, <https://www.w3schools.com/css>
3. Phpmyadmin docs, <https://docs.phpmyadmin.net/en/latest/privileges.html>
4. Developer mozilla, <https://developer.mozilla.org/en-US/docs/Web/API/Document>
5. MySQLTutorial, <https://www.mysqltutorial.org>