

Имена: Красимира Бадова

фн: 61866

Начална година: 2017

Програма: бакалавър, (СИ)Курс: 3

Тема: Уеб базирана система за създаване на документация

Дата: 2017-07-04

Предмет: wwwTech2017_18_8ed_SI_winter

имейл: badovakrasi@gmail.com

преподавател: доц. д-р Милен Петров

1. Условие

Да се създаде уеб базирана система за създаване и редактиране на документация на проект по уеб технологии. Системата да дава възможност за избираемо добавяне на различни точки по шаблон от примерна документация (виж шаблона за документация за изпитният проект по уеб технологии). Да се добави и функционалност за качване на диаграми от страна на даден автор. Различните документи да са видими от всички потребители на системата.

2. Въведение

Истина е, че в днешно време има много редактори, които позволяват създаването и редактирането на документ, било то споделяно или online, или локално на дадена машина. Повечето студенти, обаче, когато имат задачата да напишат документация за даден проект, имат различни виждания за начина, по който документа да изглежда. Дори да им е наложен шаблон, който трябва да попълнят, винаги има един-двама, които не са разбрали. А защо да не съществува система, която се справя точно с този проблем – генерира документация по даден шаблон, като дава възможност на своите потребители да генерират избираемо кои точки да включат, да качват диаграми и да експортират документа в единен формат в pdf.

3. Теория

Теоретично системата се базира на ООП модела. Всеки създаван от потребителя документ се записва в базата от данни, като избираемите му полета са записани с boolean променливи, а самите полета, които се попълват от потребителя се тип text. Авторите на документите също са модели с различни полета (fn, email, name, ...). Така свободно можем да извлечем всички инстанции на документи и автори, да ги визуализираме и редактираме.

4. Използвани технологии

За написването на функционалността на системата съм използвала главно PHP. Помогна ми за динамичното вмъкване на функционалност в даден HTML. За динамичното зареждане на някои елементи от страницата, без умишлено презареждане. Съм използвам чист Javascript (my_js.js) – например добавянето на начални точки, както и качването и премахването на диаграма. За да стилизирам страниците, съм

използвала CSS и наместа Bootstrap(style.css), а за да са семантични - семантичен HTML.

5. Инсталация и настройки

За стартирането на проекта локално е необходим apache2 сървър. За целта го стартираме чрез „sudo service apache2 start“, като проекта е достъпен на http://localhost/home_page.php.

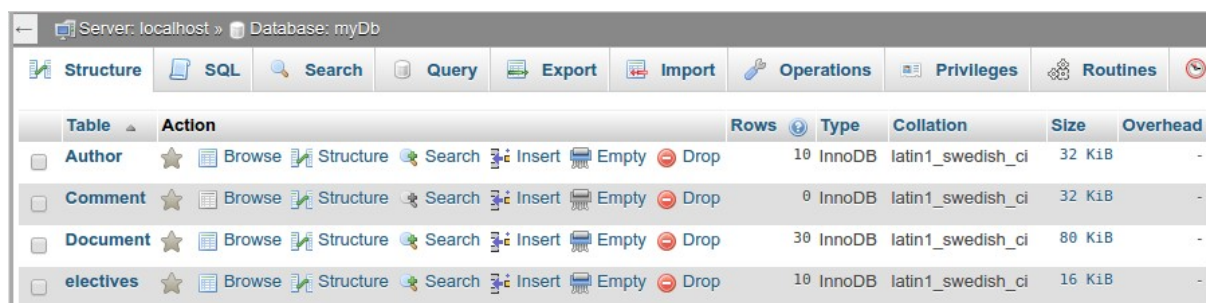


Table	Action	Rows	Type	Collation	Size	Overhead
Author		10	InnoDB	latin1_swedish_ci	32 KiB	-
Comment		0	InnoDB	latin1_swedish_ci	32 KiB	-
Document		30	InnoDB	latin1_swedish_ci	80 KiB	-
electives		10	InnoDB	latin1_swedish_ci	16 KiB	-

За създаването на база данни използвам phpmyadmin, като при самото му инсталиране въвеждам свои credentials, а които да имам достъп до базата данни. В phpmyadmin създавам база данни, като определям съответните таблици:

Във всяка таблица може да се създават колони и записи, като ,например, в таблицата „Document“ имаме следните полета:

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	
2	title	varchar(200)	latin1_swedish_ci		Yes	NULL		
3	author	int(11)			Yes	NULL		
4	created_at	date			Yes	NULL		
5	my_condition	text	latin1_swedish_ci		Yes	NULL		
6	has_my_condition	tinyint(1)			Yes	NULL		
7	introduction	text	latin1_swedish_ci		Yes	NULL		
8	has_introduction	tinyint(1)			Yes	NULL		
9	theory	text	latin1_swedish_ci		Yes	NULL		
10	has_theory	tinyint(1)			Yes	NULL		
11	comment	int(11)			Yes	NULL		
12	technologies	text	latin1_swedish_ci		Yes	NULL		
13	has_technologies	tinyint(1)			Yes	NULL		
14	settings	text	latin1_swedish_ci		Yes	NULL		
15	has_settings	tinyint(1)			Yes	NULL		
16	guide	text	latin1_swedish_ci		Yes	NULL		
17	has_guide	tinyint(1)			Yes	NULL		
18	example_data	text	latin1_swedish_ci		Yes	NULL		
19	has_example_data	tinyint(1)			Yes	NULL		
20	description	text	latin1_swedish_ci		Yes	NULL		
21	has_description	tinyint(1)			Yes	NULL		
22	contribution	text	latin1_swedish_ci		Yes	NULL		
23	has_contribution	tinyint(1)			Yes	NULL		
24	learned	text	latin1_swedish_ci		Yes	NULL		
25	has_learned	tinyint(1)			Yes	NULL		
26	resources	text	latin1_swedish_ci		Yes	NULL		
27	has_resources	tinyint(1)			Yes	NULL		
28	file1	varchar(255)	latin1_swedish_ci		Yes	NULL		

След като успешно сме създали всички полета от базата данни, трябва да окажем съответно името на базата и credential-ите в приложения код. За целта е необходимо да се попълнят само на едно място - „db_connect.php“. При възникнала грешка или неосъществено свързване с базата, сигнализиране за грешки и ги показваме на потребителя.

За зареждане на примерни потребителски данни може да използваме „myDb.sql“;

6. Кратко ръководство на потребителя


На заглавната страница на системата са визуализирани всички генерирани досега документи с информация за автора и неговия email, с възможност за пълен преглед или корекция:



<div> <div>СЪЗДАЙ ДОКУМЕНТ</div> <div>СЪЗДАЙ АВТОР</div> </div>				
Тема	Автор	Дата		
zagl	krasito, kras_i_b3342@abv.bg	2017-05-27		
fsfs	krasi, kras_i_b2@abv.bg	2017-05-27		
fr	krasi, kras_i_b2@abv.bg	2017-05-27		
rsfrd	krasito, kras_i_b3342@abv.bg	2017-05-27		
fds	krasito, kras_i_b3342@abv.bg	2017-05-27		
sdrfcd	krasi, kras_i_b2@abv.bg	2017-05-27		
ed	krasito, kras_i_b3342@abv.bg	2017-05-27		
sd	krasi, kras_i_b2@abv.bg	2017-05-27		
se	krasi, kras_i_b2@abv.bg	2017-05-27		
es	krasi, kras_i_b2@abv.bg	2017-05-27		
efs	krasi, kras_i_b2@abv.bg	2017-05-27		

При клик върху „създай автор“, потребителя бива редиректнат към форма за създаване на автор:





Създаване на Автор

Име:

Емейл:

ФН:

Курс:

Специалност:

Програма:

СЪЗДАЙ

Страницата за създаване на документ започва първо с попълване на checkbox-чета, за да сме сигурни кои полета ще включва документа ни:



Избери точки, които ще включва твоя документ:

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 разширение ☐
10. Какво научих ☒
11. Използвани източници ☒

ГЕНЕРИРАЙ

След като сме генерирали точките, които ще включим в документа ни, се отваря форма, за да попълним съдържанието на документа:



Създаване на Документ

Заглавие

Автор:

krasi.krasi_b2@abv.bg

Условие

Въведение

Използвани технологии

Какво научих

Използвани източници

Качи диаграма

Choose File

No file chosen

ЗАПАЗИ

Попълнили формата, системата ни пренасочва към страницата, на която можем да видим как изглежда досегашния ни документ:



Автор	kranibadova	ФН	456	Имейл	kranib@kranib.com	Курс	3
Специалност	Програма		Дата	Предмет		WEB технологии	
Преподавател		доц. д-р Милена Петров					
Тема							
заглавие							
Условие							
д-ф-з-с-ф							
Въведение							
с-ф-з-с-ф-з-с							
Използвани технологии							
д-ф-з-с-ф-з-с-з-с-ф							
Какво научих							
д-ф-з-с-ф							
Използвани източници							
с-ф-з-с-ф-з-с							
Редактирай				Download			

Бутона „Download“ ни сваля документа в pdf, а бутона „Редактирай“ ни пренасочва към страница, на която можем да поправим/добавим точки от документа:



Редактиране на Документ

Автор	krasibadova	ФН	456	Имейл	krasi@krasi.com	Курс	3
Специалност	Програма	Дата	Предмет	WEB технологии			
Преподавател		доц. д-р Милен Петров					

Тема

заглавие

Въведение

сдфасдфас

Използвани технологии

дсфасдсдсдф

Исползвани източници

сдфасфас

Диаграма



Добави друга точка

Теория

Качи диаграма

[Choose File](#) | No file chosen

ЗАПАЗИ

8. Описание на програмния код

Всеки файл, който съдържа някаква функционалност, свързана с php е с разширение php. В папка „forms“ се съдържат формите, които използваме за генериране на съответните точки от документацията и за самото създаване на документа. В папка „imgs“ съхраняваме качените диаграми.

Всеки php файл има горе-долу еднотипна структура, като обикновено обработваме някакви данни. Ако искаме да извлечем някакви данни и да ги визуализираме по подходящ начин, първо се пита дали request метода е GET и ако е такъв, се прави заявка към базата данни чрез PDO модела на php. Обикновено тази бизнес логика се осъществява в отделни файлове и връщат само извлечените данни от базата („fetch_data.php“), като свободно може да се достъпват в записаните променливи:

```
<?php
if ($_SERVER['REQUEST_METHOD'] === 'GET') {

    include "fetch_data.php";

    echo "<div class='document_holder'><div class='fields_holder'>
    <h2 id='create-document-h2'>Редактиране на Документ</h2>

    <h3>Автор:</h3><input disabled placeholder='".$name."'>
    <h4>ФН:</h4><input disabled placeholder='".$fn."'>
    <h4>Имейл:</h4><input disabled placeholder='".$email."'>
    <h4>Курс:</h4><input disabled placeholder='".$course."'>
    <h4>Специалност:</h4><input disabled placeholder='".$specialty."'>
    <h4>Програма:</h4><input disabled placeholder='".$program."'>

    <h4>Дата:</h4><input disabled placeholder='".$created_at."'>
    <h4>Предмет: </h4><input disabled placeholder='WEB технологии'>
    <h4>Преподавател: </h4><input disabled placeholder='доц. д-р Милен Петров'>

    <p>
    <label for='title'>Заглавие:</label>
    <br>
    <input type='text' name='title' id='title' placeholder='".$title."' value='".$title."'>
    </p>";
```

view_doc.php

home_page.php

edit_doc.php

fetch_data.php

choose_content.php

```
<?php
include "db_connect.php";

$document_id = $_GET['document_id'];

try {
    $conn = new PDO("mysql:host=$servername;port=$port;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $result = $conn->query("SELECT * FROM Document where id=".$document_id);

    while ($row = $result->fetch(PDO::FETCH_ASSOC)){
        $author = $row['author'];
        $title = $row['title'];
        $created_at = $row['created_at'];

        $has_my_scondition = $row['has_my_condition'];
        $my_condition = $row['my_condition'];

        $has_introduction = $row['has_introduction'];
        $introduction = $row['introduction'];

        $has_theory = $row['has_theory'];
        $theory = $row['theory'];

        $has_technologies = $row['has_technologies'];
        $technologies = $row['technologies'];

        $has_settings = $row['has_settings'];
        $settings = $row['settings'];

        $has_guide = $row['has_guide'];
        $guide = $row['guide'];

        $has_example_data = $row['has_example_data'];
        $example_data = $row['example_data'];

        $has_description = $row['has_description'];
        $description = $row['description'];

        $has_contribution = $row['has_contribution'];
        $contribution = $row['contribution'];

        $has_learned = $row['has_learned'];
        $learned = $row['learned'];

        $has_resources = $row['has_resources'];
        $resources = $row['resources'];

        $file1 = $row['file1'];

        $null_fields = array();
        foreach ($row as $key => $value) {
            if (is_null($value) && substr($key, 0, 3 ) === "has")
                { $null_fields[] = $key; }
        }

        $res = $conn->query("SELECT * FROM Author where id=".$author);
        while ($roww = $res->fetch(PDO::FETCH_ASSOC)){
            $name = $roww['name'];
            $fn = $roww['fn'];
            $email = $roww['email'];
            $course = $roww['course'];
            $specialty = $roww['specialty'];
            $program = $roww['program'];
        }
    }
}
catch(PDOException $e){
    echo $sql . "<br>" . $e->getMessage();
}
$conn = null;
?>
```

Аналогично е и за request метода POST, като файлът, който се грижи за обработването на post заявки е под името „post_data.php“. Тук има обаче една особеност – при update-ването на даден документ и при избираемое добавяне на точки от документа, не знаем всеки път полетата ни как се казват, затова всеки път ги извличаме от \$_POST и ги натрупваме в списък с съответните им имена. След което си контролираме сами Update заявката към базата.

Още повече, при наличието на качен файл(if empty(\$_FILES[«file1»])) го добавяме към този списък.

```
<?php
include "db_connect.php";

try {
    $conn = new PDO("mysql:host=$servername;port=$port;dbname=$dbname", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $id = $_POST['id'];

    $posted_data = array();
    foreach ($_POST as $key => $value) {
        if( $key != "id" ){
            $asd = $key . " = '" . $value . "'";
            $posted_data[$key] = $asd;
        }
    }

    $list_to_string = "";
    foreach ($posted_data as $key => $value) {
        $list_to_string .= " " . $value . ",";
    }
    if (empty($_FILES['file1'])) {
        $list_to_string .= " file1 = '" . $_FILES['file1']['name'] . "',";
    }

    $params = substr_replace($list_to_string, "", -1);
    $query = "UPDATE Document SET" . $params . " where id=".$id;
    $prepared = $conn->prepare($query);
    $prepared->execute();
}

catch(PDOException $e){
    echo $sql . "<br>" . $e->getMessage();
}
$conn = null;

define ('SITE_ROOT', realpath(dirname(__FILE__)));

$upload_dir = SITE_ROOT."/imgs/";
$upload_file = $upload_dir . basename($_FILES['file1']['name']);

move_uploaded_file($_FILES['file1']['tmp_name'], $upload_file);
?>
```

За свалянето на документа в pdf използваме библиотеките „simple_html_dom“ и „tcpdf_include“, като първо вземам целия HTML от дадена страница:

```
$html = file_get_html($_GET['redirect'], NULL, NULL, 400);
```

```
$doc = str_get_html($html);
```

вземам съответния клас от нея:

```
$href = $doc->find('.fields_holder')[0];
```

и добавям съдържанието в pdf:

```
<?php
require_once('tcpdf_include.php');
require 'simple_html_dom.php';
try
{
    $pdf = new TCPDF(PDF_PAGE_ORIENTATION, PDF_UNIT, PDF_PAGE_FORMAT, true, 'UTF-8', false);

    $html = file_get_html($_GET['redirect'], NULL, NULL, 400);
    $doc = str_get_html($html);
    $href = $doc->find('.fields_holder')[0];
    $asd = ' ';
    $href .= $asd;

    $pdf->AddPage();
    $pdf->SetFont('dejavusans', 'B', 12);
    $pdf->writeHTML($href, true, false, true, false, '');
    $pdf->Output('document.pdf', 'I');
}
catch(PdfcrowdException $why)
{
    echo "Pdfcrowd Error: " . $why;
}
?>
```

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

Бъдеща насока за развитие би била възможността да се качват повече от една диаграма за даден документ, както и ввеждането на сесия и session authentication, вместо създаването на Автор както в момента.

10. Какво научих

Като цяло до момента, в който не започна курса по уеб технологии, не се бях занимавала с език като php. Мога да кажа че свикнах със синтаксиса му за времето, което отделих по проекта. Мога да съмнитвам форми, които да записват информация в база данни, мога да визуализирам данни, идващи пак от базата данни. Научих какво е phradmin и как да създавам/променям базата от там по по-лесен начин от заявки в конзолата. Изчетох доста статии за това как като качиш някакъв img, той да се запаzeti не само в базата, но и локално. Също така доста време отделих и по генерирането на pdf от чист HTML.

11. Използвани източници

<https://github.com/pdfcrowd>

https://www.w3schools.com/php/php_forms.asp

<http://php.net/manual/bg/reserved.variables.post.php>

https://www.w3schools.com/php/php_mysql_select.asp

<http://php.net/manual/en/reserved.variables.get.php>

Предал:

/фн, имена, спец., група/

Приел:

/доц. Милен Петров/