**Имена:**  Владислава Венциславова Маркова **фн:** *81271***Начална година:** *2015*  **Програма:** бакалавър, (КН) **Курс: 4**   
**Тема: 23.1 Генератор на тест за самопроверка по зададени въпроси към реферат и статистика  
Дата: 2020-01-26 Предмет: Web технологии (w15ed) имейл: vladislavam1996@gmail.com**

**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 използваниресурси**, архива да се казва 9999\_project\_final.zip. (Успех!). (Редактирайте маркираните зони в жълто с коректната информация)**

# ТЕМА: 23.1 Генериране на тест за самопроверка по зададени въпроси към реферат и статистика

## Условие

Да се създаде система, която позволява правенето на тест от даден потребител, генерирайки тест за самопроверка по зададени въпроси към реферат. Системата трябва да поддържа и статистика за направените тестове от даден потребител, за отговорените въпроси – съответно правилно или грешно, както и техния брой. Системата да предоставя възможност за добавяне към базата от данни на нова тема, както и нов въпрос по съответна тема. При започването на тест потребителят трябва да има възможност да избере темата, за която да се генерират въпросите за теста, който ще прави, както и да избере типа на тези въпроси за теста, което ще рече да избере дали иска да прави тест ‘за предварителни знания (pre-question)’, ‘по време на презентацията (poll-question)’ или ‘след презентацията (test-your-knowledge)’. Позволява се вход в системата, като за целта всеки потребител въвежда факултетен номер, чрез който бива вписан в системата и използва съответните необходими и предоставени му функционалности.

## Въведение

Вземайки предвид характера на идеята и разработката на системата, може да кажем, че има немалко системи с аналог на нашата. Това, към което тя се цели като функционалност, която да предостави, обаче, е строго оринетирано към генериране на тестове, свързани с рефератите, създавани от студенти в курса Web технологии, воден от доц д-р Милен Петров във ФМИ на СУ „Св. Климент Охридски”. В основата е предоставянето на възможност на даден потребител да проследява как другите потребители отговарят на въпросите, които той е задал към реферата си. Именно за кой въпрос колко верни отговори има, колко са грешните, колко общо пъти е отговорен, имайки предвид и типа на въпроса, който е задал. Това се прави с цел подобряване на самите въпроси, които студентите в курса пишат по темата си, това е една обратна връзка за тях, която би им помогнала да променят, обновят или добавят различни въпроси. Другият момент опира до самопроверката на потребителите, които използват функционалността да правят тип тест по дадена тема. В някакъв обхват това би ровишило знанията, които студентите придобиват от запознаването с рефератите на останалите, защото, правейки теста, биха имали по-голяма мотивация да запомнят и проучат определени въпроси. Систимата позволява ролята на администратор, за когото има предоставен администраторски панел, чрез който той получава информация и под формата на статистика за това какви тестове са правени и с какви резултати. Има достъп до всички останали функционалности.

## Теория

Използваната архитектура следва до известна степен архитектурния шаблон MVC (Model-view-controller), където се предполага да има обособени съответните компоненти за front-end частта, работата с клиентските заявки и организирането на потребителския интерфейс. В директория api са обособени логиките по отделните елементи – topic, question, test, history и users. Като връзката с базата данни е отделена в utils.php, където чрез определените функции се позволява изпълнението на писане, четене и промяна на записите в базата. Предимството на този модел е по-голямата независимост на отделните части, която се постига, както и по-високата в определени граници, изменяемост на системата.

## Използвани технологии

При разработката на проекта са ползвани **HTML**, **CSS**, **Javascript** и **PHP**. Също така е използвана и библиотека, написана на Javascript, Papa Parse, чрез която да се прочитат CSV файловете, за изпълняване на функционалността import csv за добавяне на въпроси от файл в базата от данни с въпроси. Потребителският интерфейс позволява потребителят да оперира със системата чрез достъпни и разбираеми за обикновен потребител менюта, бутони и визуализации. PHP е използван за сървърната част на системата, основната бизнес логика е реализирана чрез него. Използва се за връзка с базата данни, писане и четене от нея, както и промяна. Валидацията на потребителския вход също се случва чрез PHP. Потребителските заявки се изпращат към сървъра асинхронно чрез Javascript. Визуализирането на обратната връзка с потребителя след изпълнение на изпратена от него заявка, се реализира отново чрез Javascript и HTML. За съответните избрани стилове е използван CSS. Използва се приложението XAMPP, което предоставя и осъщестявва достъп до Apache сървър и позволява правенето на заявки към базата от данни чрез езика MySQL.

* Сървър – **Microsoft Windows 7, x64**
* Сървърен език – **PHP 8.0.0**
* Уеб сървър – **Apache 2.4.46**
* Сървър за База данни – **MariaDB 10.4.17**
* **phpMyAdmin – 5.0.4**

## Инсталация и настройки

Предварително е необходимо да бъде инсталиран XAMPP. Когато решим да пуснем даден проект, какъвто проект е предметът на настоящата документация, трябва всички файлове, с които разполагаме(или цялата папка на проект), да бъдат поставени в  C:\xampp\htdocs\ и, когато искаме да заредим в браузъра някоя от страниците, вместо  C:\xampp\htdocs\ пишем localhost. Трябва чрез контролния панел на XAMPP да се стартират модулите Apache и MySQL като се натиснат бутоните start срещу тях.

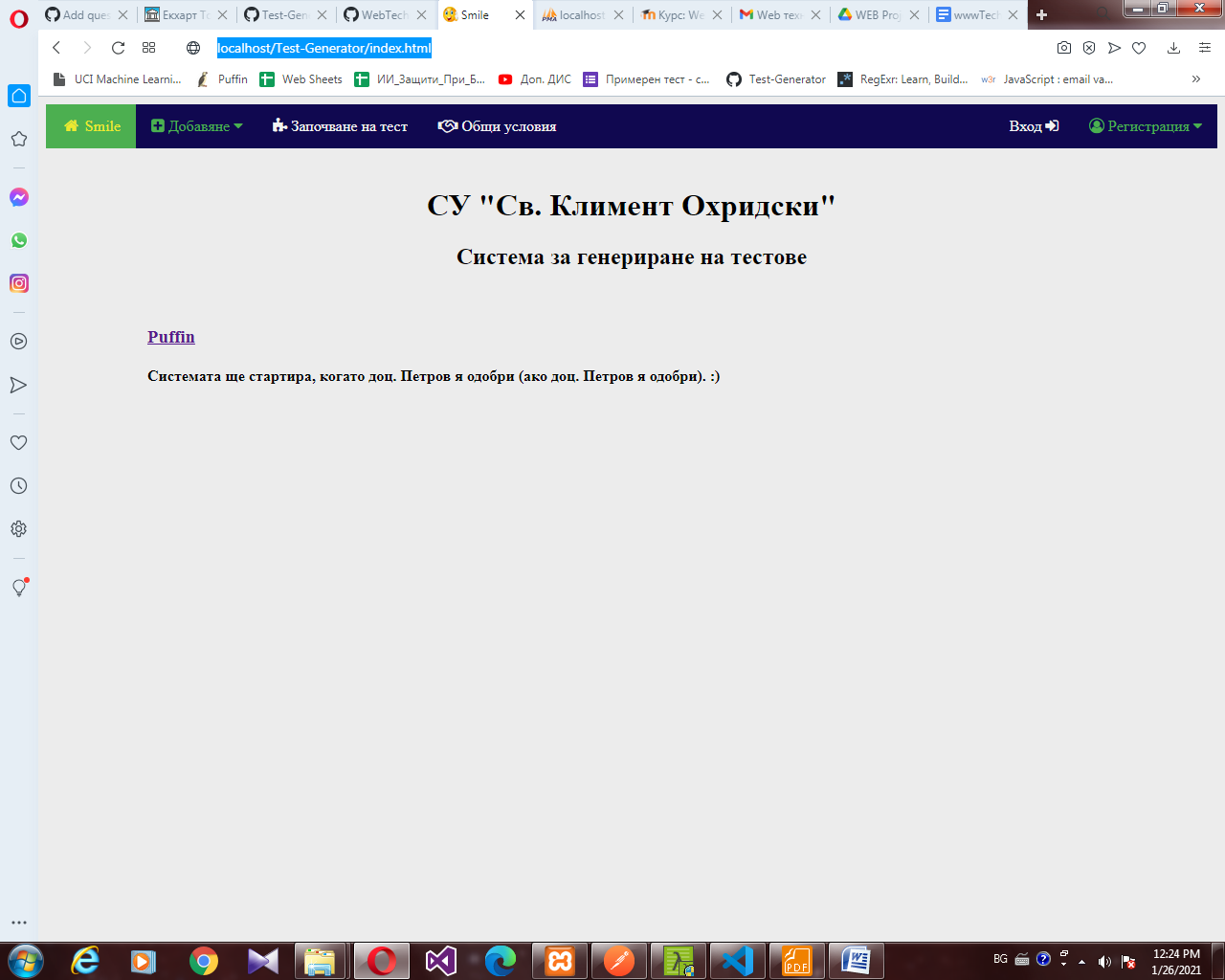
Пример: Ако искаме да достъпим страницата за започване на тест, пишем в браузъра

http://localhost/Test-Generator/pages/Test.html

Другият начин е да отворим началната страница на системата на адрес:

http://localhost/Test-Generator/index.html

и оттам чрез потребителския интерфейс да навигираме до съответните места, които искаме да посетим.



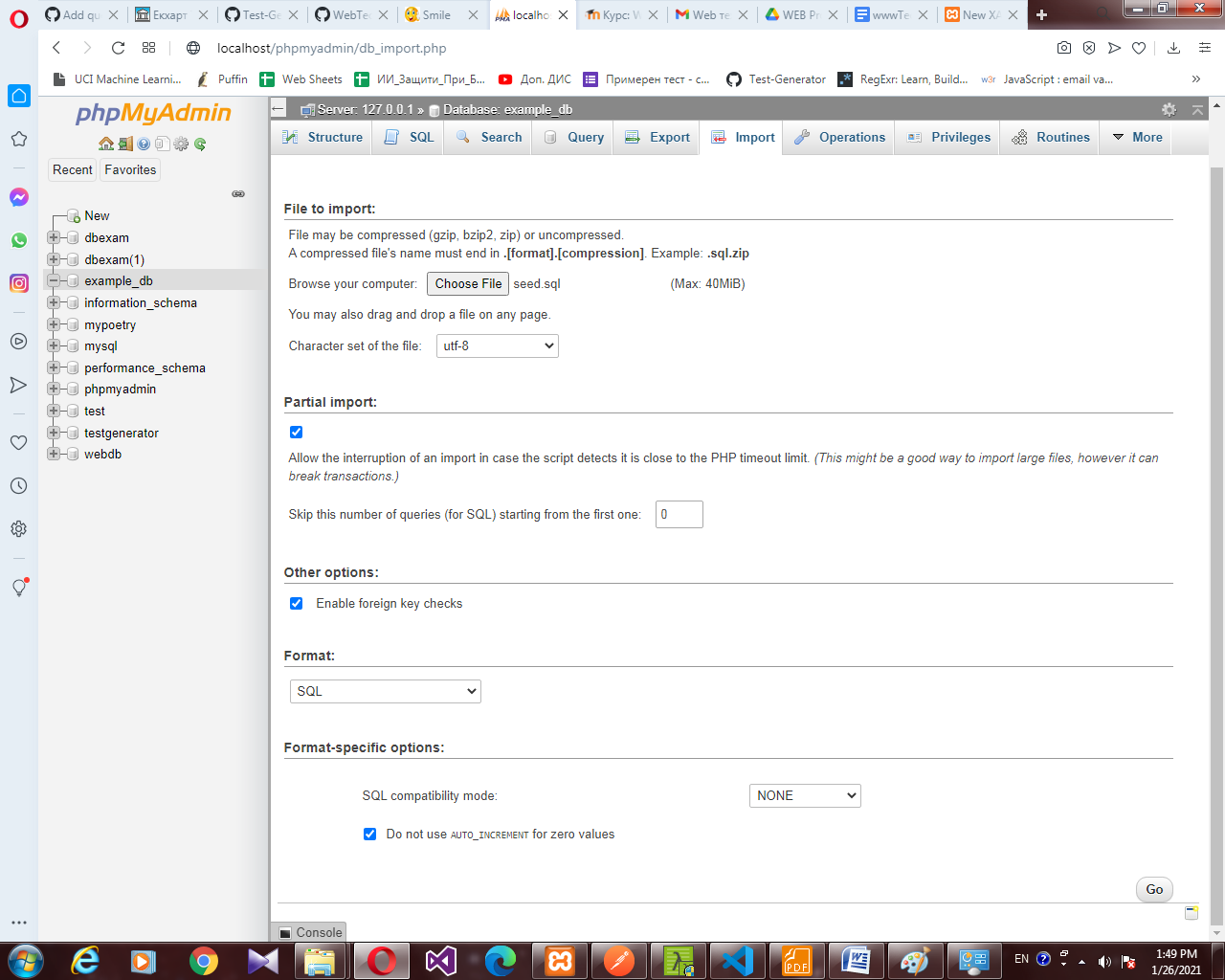
## Кратко ръководство на потребителя

В началото потребителят стартира **XAMPP (C:\xampp\ xampp-control.exe)** и натиска **start** бутоните срещу модулите **Apache** и **MySQL**. След това достъпва през браузъра следния адрес:

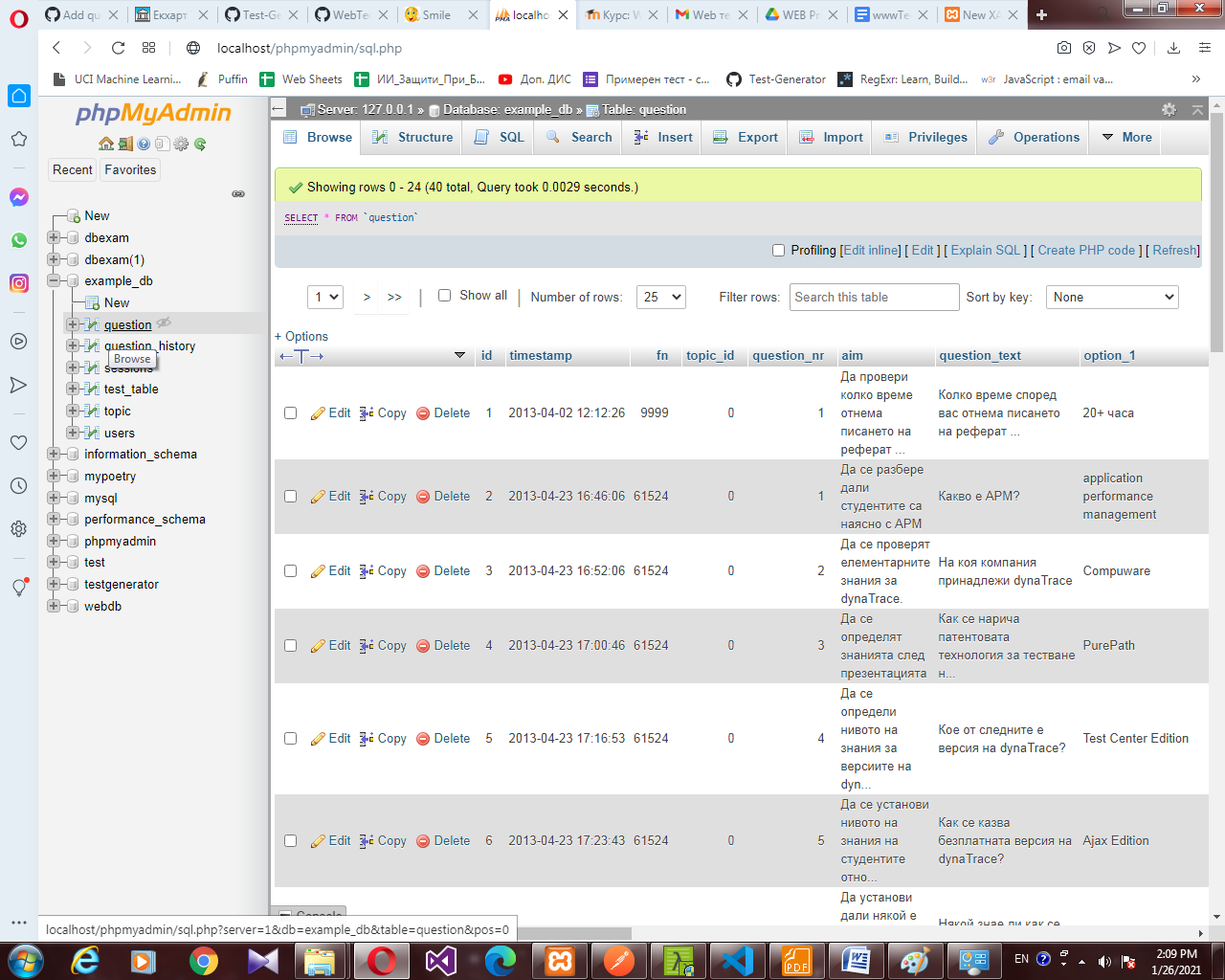
<http://localhost/phpmyadmin/>

Тук с помощта на **phpMyAdmin** потребителят ще създаде собствената база от данни. Какви са стъпките за това?

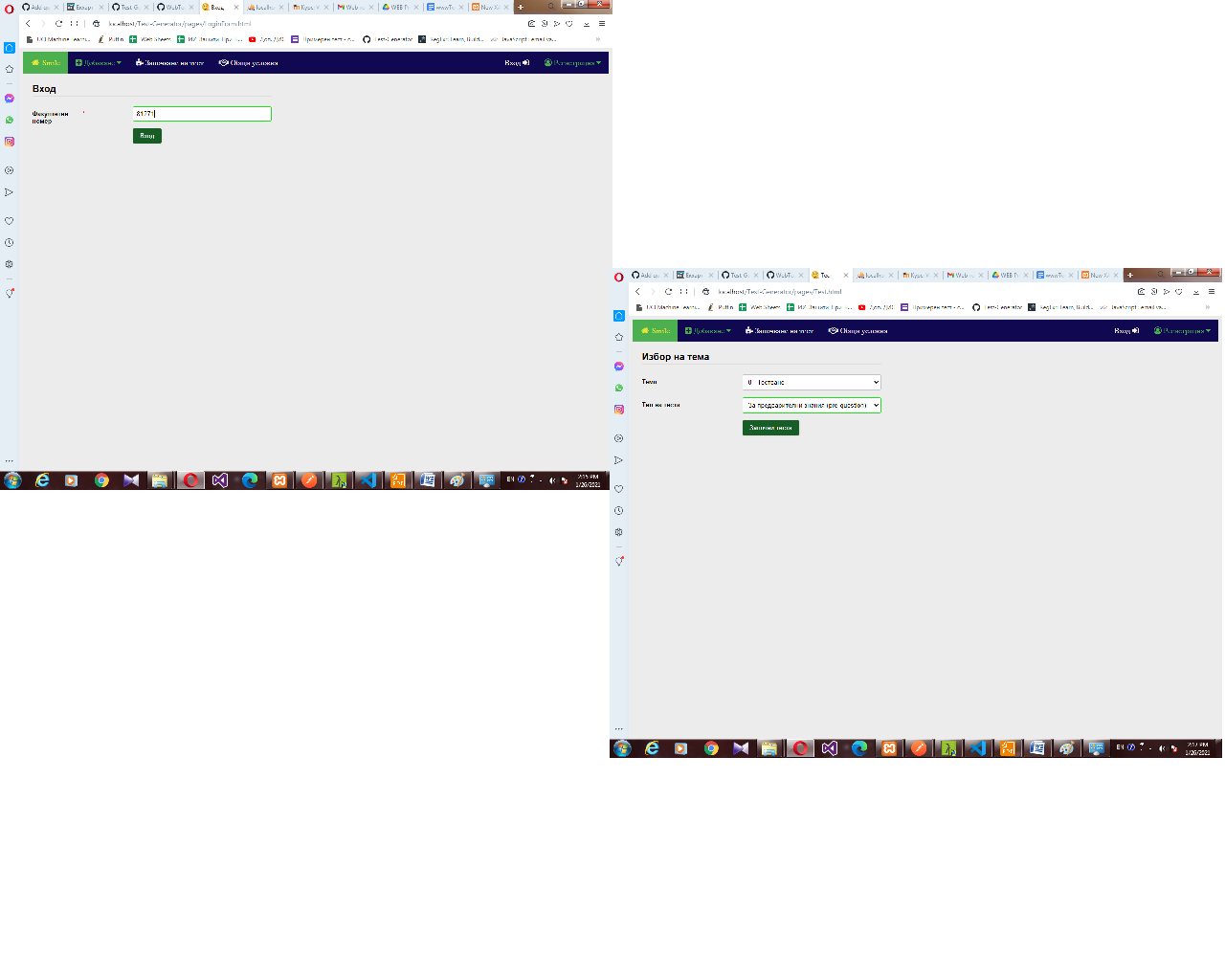
В горния ляв ъгъл натиска опцията **New**, с която ще създаде празна база данни за начало. След което в двете поленца за въвеждане(едно за въвеждане и едно за избиране) пише съответно името на базата, в този случай пише **tesgenerator**  и избира от падащото меню опцията **utf8mb4\_general\_ci**. След това натиска бутона **Create**. Зарежда му се страничката с новосъздадената база данни, която за момента е празна. От менюто с изредените отгоре опции избира **Import**. След което натиска **Choose file** и навигира до желания файл (в нашия случай това е файлът **seed.sql**), пътят е следният: **C:\xampp\htdocs\Test-Generator\database\_seeds\seed.sql**. След това натиска бутона **Go** най-долу вдясно.



След изпълнение на тази заявка отива в менюто отляво и натиска върху името на създадената база – **testgenerator**. Като подмодули може да види отделните таблици (релации), които са се създали при зареждането на файла. Те са **question**, **question**\_history, **sessions**, **test\_table**, **topic**, **users**. При натискането върху някоя от тях може да види цялата информация за записите вътре, може да добавя нови, да редактира и да трие такива, както и други опции, позволени от phpMyAdmin.



Сега вече е ред потребителят да напише в браузъра адреса: **localhost/Test-Generator/index.html**. Отваря се **началната страница**, откъдето да отиде на **Вход** и да влезе със своя факултетен номер. От падащото меню **Добавяне** може да избере съответно да добави **Тема** или **Въпрос**, в **Общи условия** може да прочете и да се запознае с общите условия, касаещи работата му в системата и в **Започване на тест** може да избере тема и тип на въпросите по темата, които да бъдат генерирани, и да започне теста.



## 7. Примерни данни

...

## 8. Описание на програмния код

...

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

...

## 10. Какво научих

...

## 11. Използвани източници

...

Предал (подпис): ………………………….

/*фн, имена, спец., група*/

Приел (подпис): ………………………….

/доц. *Милен Петров*/

## Препоръки за предаване на проекта (ИЗТРИЙ до края-при предаване на проекта!)

Заб. *Спазвайки препоръките по-долу биха спомогнали да направите добри проекти по Уеб технологии.*

## Финален проект (инструкции към 7 издание)

2. Изпитни проекти: (настоящият документ може да съхраните като .docx . За хората, ползващи редактори, различни от MS Office – освен docx/rtf да качат и pdf версия на документацията – за по-сигурно.)

2.1. Темите за изпитните проекти трябва да са съгласувани с мен на място (вече няколко хора го направиха); като тема си запишете ф.н. в гугъл докса - там пише как се записвате (в коя колона и какъв разделител да ползвате); Обем на проектите: 30 човеко-часа на човек на проект. Това е доста относително, но все пак е нещо.

2.2. За документация на проекта ползвайте шаблона (ще гледам да го кача скоро); Задължително потребителска документация (а.к.а. userguide) - няколко скрийншота с кратко разяснение; условие на проекта (т.е. какво сте разтълкували сме се разбрали да правите - то може и да се различава от описанието дадено в гугъл докс-а към момента); както и инструкции за инсталиране; за защитата - ще дам няколко дати през сесията/преди сесията + официалните дати, за който не успее да мине преди това; за защитата - кода, документацията/необходими библиотеки/среди, инсталации - за инсталирането се предават на DVD (ако има още такова нещо като CD-може и на CD). Може по изключение да сложа И форма за качване само на документацията и програмен код и в мудъл, НО идеята е, че ако няма интернет (да речем е паднал мудъла, спрял е тока и т.н. - само по съдържанието на диска, който сте предали да може да се инсталира и подкара проекта); Също така разпечатвате първа страница от проекта (с името ви и заданието), 1 страница от userguide-а и последната страница, където пише предал/приел (там пише вашите и моите имена), т.е. не е необходимо да печатите цялата документация - така или иначе ще я има в електронен формат; За хората, които не ползват MS Office - ще помоля освен изходният документ в docx/rtf, да качат нещата и е pdf формат, т.к. често такъв тип документация се размества и не се чете. В кода сложете и MySql sql скрипт със създаване на на таблицата и скриптове, задаващи примерни данни (т.е. може да изтестваме приложението дали работи без данни, и ако за да се види пълната прелест на проекта е нуждно да се вкарат предварително данни - sql и/или снимки/звуци и т.н. може да ги подготвите на диска). **БЕЗ флашки!**

2.3. Срок за защита на изпитните проекти - до изпита. За съгласуване на теми - ми пишете да се разберем за час за консултации за проектите.

Вижте и инструкциите към 6 издание – които не противоречат с инструкциите за настоящото издание – са валидни и сега.

## Финален проект (инструкции към 6 издание)

Заб. *Спазвайки препоръките по-долу биха спомогнали да направите добри проекти по Уеб технологии.*

**Задължително:** Реферата да е в zip файл с име на зип файла: **fakNo\_final\_project.zip** където вместо **fanNo** пишете факултетният си номер (според инструкциите, зададени в форума на курса).

## Още няколко упътвания:

I. АРХИТЕКТУРА НА УЕБ СИСТЕМАТА: Да има три-слойна архитектура (Препоръки: 1. презентационен слой - css/js/html, 2. БД: MySql и като допълнение по желание може да имате импорт/експорт към XML; 3. Бизнес логика - Php)

II. ФУНКЦИОНАЛНИ ХАРАКТЕРИСТИКИ: Съгласуват се с преподавателя - избира се темата, а каква да е функционалността - питате на лекции или се явявате на предварителна защита, където съгласувате обхвата на изискванията;Да се познават и спазват добрите практики, завършеност - според сложността на приложението – 30 човеко-часа);

III. НЕ-Функционални характеристики:

- Конфигурируемост (лесно да може да се инсталира - например: смяна на едно место ако се смени физически папката на сървъра - да е в под-папка, смяна на адреса - IP/URL, смяна на име/парола/

- Разширяемост - лестно да може да се разширява функционалността на различните слоеве;

- Документация - без да се пуска проекта - да може да се ориентира в неговата функционалност, как се настройва, точки на разширение на отделните слоеве- ако има особенност, примерни данни за тестване - администраторски акаунти, ръководство на потребителя за различните роли - скрийншоти с кратко заглавие).

IV. Други изисквания, зададени на лекция (следете лекциите и форумите).

**Заб.** *Има вероятност да пусна нова инстанция на системата за рефератите, но този път за проекти, където в html да сложите документацията си (т.е. подобен на този шаблон, но в html формат)*.

За изискването конфигурируемост (описвате в документацията), се очаква нещо от вида:

<?php //conf.php

<?php

$vhosts = <<<EOT

#############################

## MP-1.10.1: w11ref.w3c.fmi.uni-sofia.bg

#############################

<VirtualHost \*:80>

ServerAdmin milenp@fmi.uni-sofia.bg

DocumentRoot "C:/xampp/htdocs/w11ref.w3c.fmi.uni-sofia.bg/\_PUB"

DirectoryIndex index.php

ServerName w11ref.w3c.fmi.uni-sofia.bg

# ServerAlias 9999.w3c.fmi.uni-sofia.bg

<Directory "C:/BACKUP\_SYSTEMS/htdocs/w11ref.w3c.fmi.uni-sofia.bg/\_PUB">

Options All

AllowOverride All

Require all granted

</Directory>

</VirtualHost>

EOT;

//echo "<pre>$vhosts . </pre>";echo "<pre>$vhosts . </pre>";

// $configs = include('conf.php');

//if - uncomment/comment -> in calling part can be used as $configs['vhosts'];

**return** (**object**) **array**(

//0. sys\_cfg

'cfg\_ver' => '1',

//1. sys\_cfg

'cfg\_system\_mgmt' => 'w11ref',

'cfg\_system\_name' =>'www\_11ed\_referats',

'cfg\_dns\_prefix' => 'w11ref',

'cfg\_dns\_sufix' => 'w3c.fmi.uni-sofia.bg',

'HTTP\_URL\_PREFIX' => "http://w11ref.w3c.fmi.uni-sofia.bg",,

//2. db\_cfg (from queries.php)

'DB\_SERVERNAME' => '127.0.0.1',

'DB\_USERNAME' => 'w11ed\_fn9999',

'DB\_PASSWORD' => ' w11ed\_fn9999',

'DB\_NAME' => ' w11ed\_fn9999',

//9. vhost

'vhosts' => $vhosts,

'vhosts\_ServerAdmin' => 'milenp@fmi.uni-sofia.bg',

'vhosts\_DocumentRoot' => 'C:/xampp/htdocs/w11ref.w3c.fmi.uni-sofia.bg/\_PUB',

'vhosts\_port' => 80,

'vhosts\_ServerName' => 'w11ref.w3c.fmi.uni-sofia.bg',

);

?>

**За да се използва:**

$configs = include('conf.php');

**header("Location: ". $configs->HTTP\_URL\_PREFIX."/choose.php");**

**Други варианти за конфигурация:**

***<?php***

*//*install.php *- can called once, initiating database, creating configuration file of doesnt exists, etc.*

*//Optional::* ConfigPanel.php *-> Configures/changes settings of app*

*//Required:* config.php, db\_sql.txt

*//required: help.php (can be part of documentation)...*

*//or config.ini -> със същите настройки;*

*class Config {*

*$SITE\_FN = 61999; //can be used bellow*

*$SITE\_CREATOR = "Your Name(s)";*

*$SITE\_ADMIN\_EMAIL = "your@email.com";*

*$SITE\_INFO = "This project was created during ...year, on Web Technologies, Sofia University, FMI, lead by:*

*Name of Instructor, assistant: Name-Of-Assistant";*

$SITE\_URL=“<http://loremipsum.fmi.uni-sofia.bg/WEBTECH/www_6ed_prj/61999_alg_animation>”;

*$ROOT\_FOLDER="c:\xampp\htdocs\www\_6ed\_prj/61999\_alg\_animation"*

*$DB\_USER="61999\_user";*

*$DB\_PASS="61999\_pass";*

*$DB\_NAME="www\_6ed\_61999\_alg\_animation";*

*$SITE\_DESCRIPTION="What is ready, and what can be improved for future";*

*$PROJECT\_REQ="...(from documentation)";*

*}*

***?>***

История на версиите

* Последна модификация 0.3 /2020-01-06/by MP
* Последна модификация 0.2/2018-12-09/by MP

Успех!