**Имена:**  Павел Тодоров **фн:** **81089****Начална година:** *2017* **Програма:** бакалавър , КН **Курс: 4**   
**Тема: Drop Out Students  
Дата: 2018-01-24 Предмет: wwwTech2017\_18\_9ed\_KN\_winter имейл: ptodorov\_ed@abv.bg**

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

# ТЕМА: Drop Out Students

## 1. Условие

Условието е да се направи система (приложение), в която от файл да се вкарва информация за студентите, посещаващи даден курс. Тази информация трябва да съдържа факултетния номер на студента, както и данни за това как той се е справил със заданията, поставени в съответния курс. Заданията могат да са контролни, домашни, реферати, проекти, изпити и други подобни. Данните за едно задание включват това колко точки е изкарал студентът, датата на предаване, коментар от преподавател и краен резултат от заданието, който може да е например два пъти по-малък от точките, ако заданието е предадено преди крайния срок. За всяко задание преподавателят трябва да може да дефинира различни прагове и изисквания, като например минимални допустими точки за него, краен срок за предаване и максимален брой точки. След като извлече данните за всички студенти, системата трябва да провери дали студентите са покрили изискванията и да покаже тази информация на преподавателя по подходящ начин. Такъв начин може да е като неудовлетворениете изисквания, като изпуснат срок за предаване, се оцветят в червено. Освен тези данни, преподавателят трябва да може да вижда и сумата на точките от всички задания за курса, както и изискванията за курса.

## 2. Въведение

За всеки курс има студенти, които се спаравят добре и други, които се справят не толкова добре. Целта на проекта е приподавателят да види кои студенти са застрашени от отпадане, като използва определените от него изисквания и данните, които му се предоставят.

## 3. Теория

За изчисляването на точките са използвани прости аритметични опрерации, а за работота с базита данни – операциите за създаване на таблица и вмъкване на данни в нея. За бизнес логиката на проекта са използвани класове и функции.

## 4. Използвани технологии

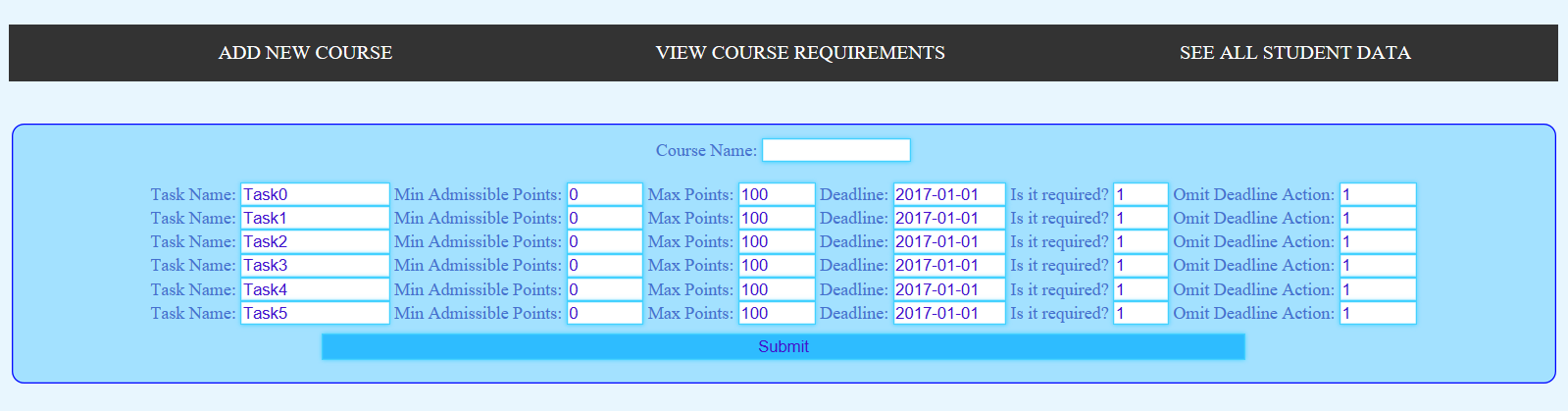
Използван е софтуерът XAMPP, който съдържа  [Apache HTTP Server](https://bg.wikipedia.org/wiki/Apache_HTTP_Server" \o "), база от данни Maria DB и интерпретатор на PHP. За създаването на приложението са използвани и езиците HTML и CSS.

## 5. Инсталация и настройки

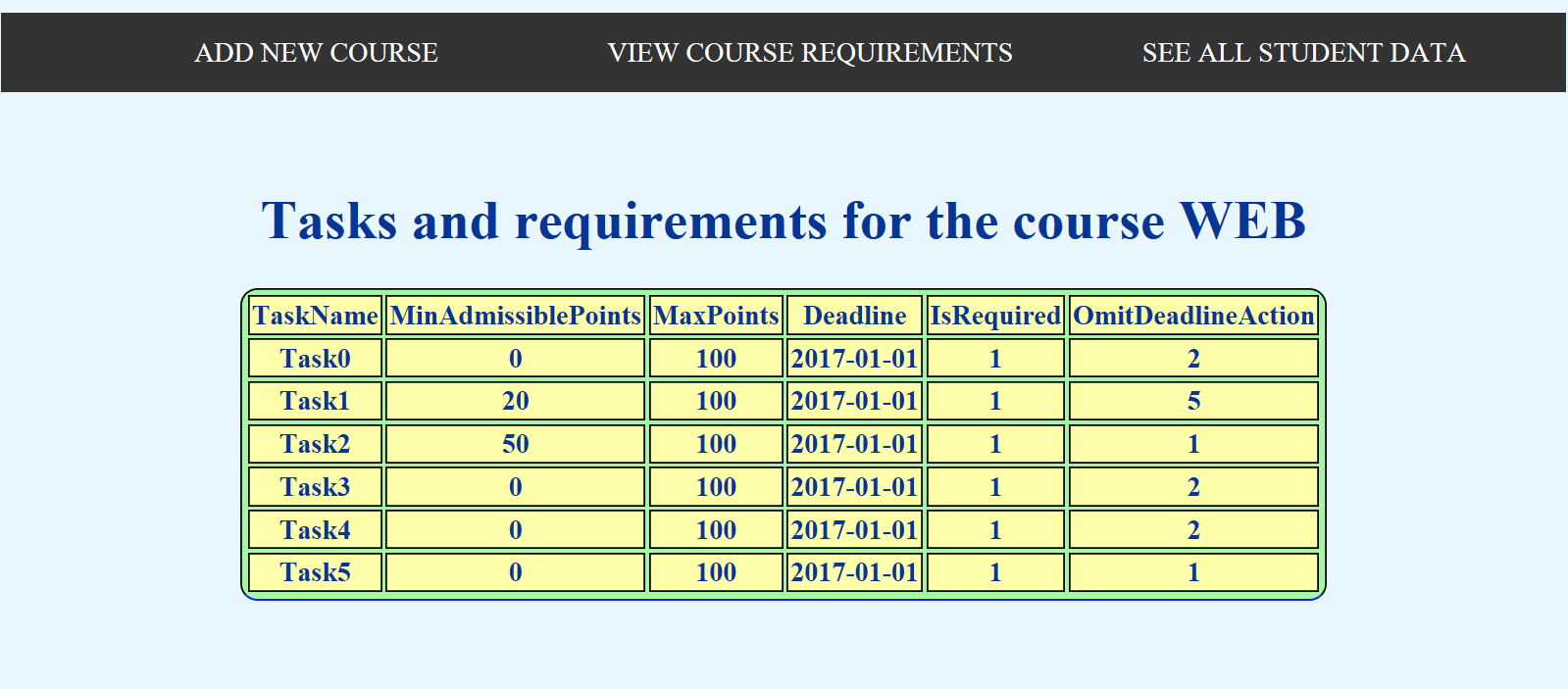
За да се използва приложението, е необходимо то да се разархивира в папката htdocs (C:/xampp/htdocs xampp ако е инсталиран в С ) и да се създаде база от данни с име Drop out students. След това потребителят трябва да сложи файла с резултатите на студентите в папката Files. Този файл трябва да има същото име като това на курса без бели символи. За тестови цели във Files има файл с име WEB.txt, който да се използва за курса по уеб технологии.

## 6. Кратко ръководство на потребителя

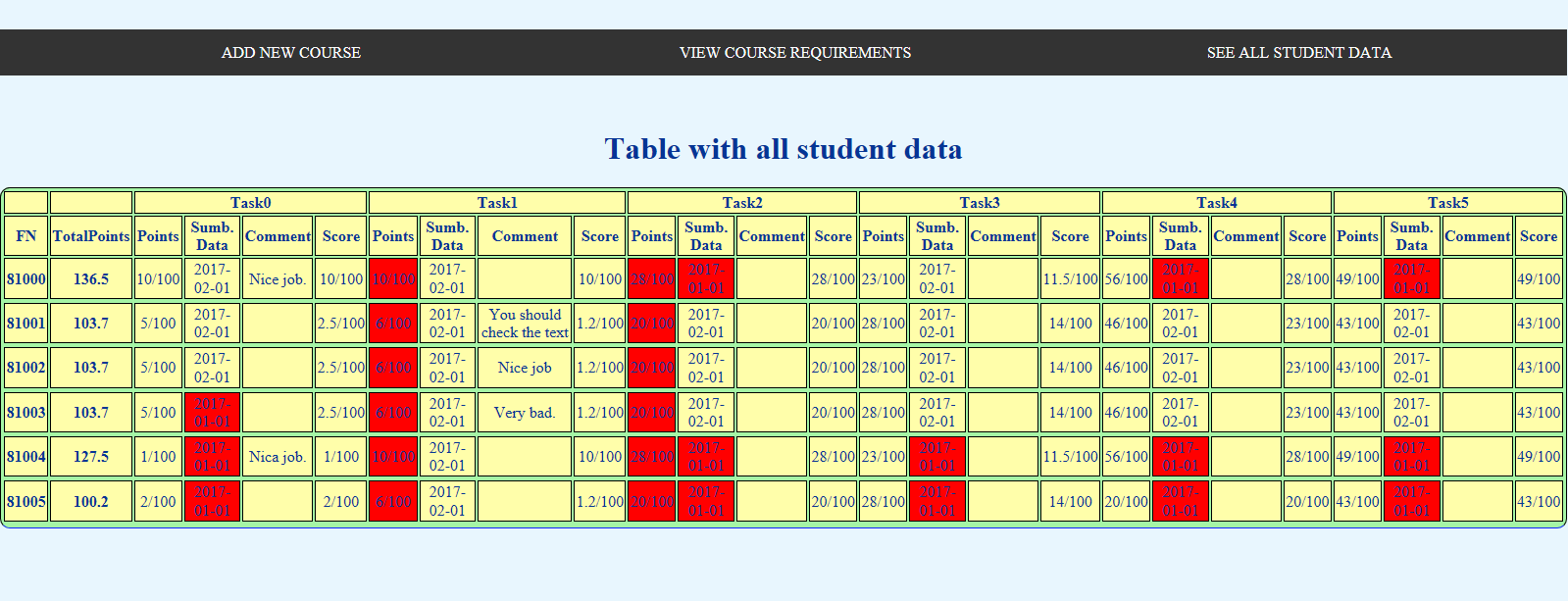
Когато използва приложението, потребителят трябва първо да отиде на основната страница на проекта. След като натисне бутона ADD NEW COURSE, ще се появи форма с изискванията(фиг. 1). Когато я попълни, трябва да натисне бутона Submit. Потребителят може да види изискванията по всяко време, като натисне бутона VIEW COURSE REQUIREMENTS (фиг. 2), а данните за всички студенти - като натисне SEE ALL STUDENT DATA (фиг. 3). При студентските данни в червено са отбелязани заданията, които са оценени с по-малко от минималния праг точки и заданията, предадени след крайния срок.



*фиг. 1 Форма за добавяне на изисквания към курс*



*фиг. 2 Показване на изискванията към даден курс*



*фиг. 3 Показване на резултатите на студентите*

## 7. Примерни данни

В папката Files има файл с име WEB.txt . Достатъчно е потребителят да отиде на страницата ‘’ADD NEW COURSE’’и да попълни изискванията, като за име на курс сложи WEB, за да тества системата.

## 8. Описание на програмния код

Основните функционалности на прилижението се изпълняват от класовете AllStudentDataOnCourse и StudentDataOnCouse, които се намират в папката Students. Първият работи с данните за всички студенти, а вторият – само с един студент. AllStudentDataOnCourse изпълнява функциите, свързани с изчитането на файла със студентските данни, добавянето им в база от данни и извличането им оттам. Аналогично класът има и функции за добавяне и извличане на изисканията от базата от данни. AllStudentDataOnCourse осигурява и показването на данните в приличен вид и изчисляването на крайния резултат за всеки студент. Допълнителни помощни класове са TaskData и TaskRequirements, които улесняват работата със заданията и изискванията.

Класът ConnectToDB осигурява свързването към базата от данни. Това означава, че потребителят трябва да въведе паролата, user-name-a и другата информация за това свързване в самия клас и никъде другаде.

HTML страниците се намират в папката Pages, а в папката CSS има един файл – style.css . Те отговарят съответно за навигацията между страниците и добрия им външен вид.

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

Моят принос е в това да измисля дизайна на проекта и да го имплементирам. Трябваше да реша какви функционалности да поддържа той, как да се организират файловете му и кои да са класовете и функциите, които да осъществяват тези функционалности. Ограничение, което проектът има към този етап на разработването си е, че той работи само за един курс – този по Уеб Технологии. Тъй като всички необхидими таблици за курса се саздават автоматично, чрез PHP скрипт, не би било твърде трудно да се добави и възможността за поддържане на много курсове. Друго нещо, което в бъдеще може да се добави, е броят на задачите към даден курс да може да се променя. В момента този брой е фиксилан на 6. Освен това е добре да може преподавателят да има има възможността да променя изискванията към курса, след като вече ги е добавил и да обновява оценките на студентите. Може да се добави и логин форма, в която всички потребители да се логват, както и всеки от тях да има роля, позволяваща му да вижда само определени данни.

## 10. Какво научих

В процеса на измисляне на дизайна и писането на кода към този проект се научих как да направя приложение, което да се състои от няколко слоя – презентационен (css, html), база от данни(MySql) и бизнес логика(Php).

## 11. Използвани източници

[1] https://www.w3schools.com/

[2] http://php.net/manual/en/language.basic-syntax.php

[3] https://dev.mysql.com/doc/refman/5.7/en/sql-syntax-data-definition.html

Предал: ………………………….

/*81089, Павел Тодоров, КН, група 1*/

Приел: ………………………….

/доц. *Милен Петров*/