

РЕПУБЛИКА БЪЛГАРИЯ

Заместник-министър на образованието и науката

**З А П О В Е Д**

**№ РД 09 – 2360/14.09.2020 г.**

На основание чл. 13д, ал. 2, т. 1 от Закона за професионалното образование и обучение, при спазване на изискванията на чл. 66, ал. 1 и 2 от Административнопроцесуалния кодекс и във връзка с осигуряването на обучението по учебен предмет и Заповед № РД 09-3708/23.08.2017 г. на министъра на образованието и науката

**У Т В Ъ Р Ж Д А В А М**

Учебна програма за специфична професионална подготовка по учебен предмет **учебна практика: интернет програмиране** за специалност код **4810301 „Приложно програмиране“** от професия код **481030 „Приложен програмист”** от професионално направление код **481 „Компютърни науки“.**

Учебната програма влиза в сила от учебната 2020/2021 година.

 *Приложение*

**МИНИСТЕРСТВО НА ОБРАЗОВАНИЕТО И НАУКАТА**

**У Ч Е Б Н А П Р О Г Р А М А**

**ЗА СПЕЦИФИЧНА ПРОФЕСИОНАЛНА ПОДГОТОВКА**

по

**УЧЕБНА ПРАКТИКА: ИНТЕРНЕТ ПРОГРАМИРАНЕ**

УТВЪРДЕНА СЪС ЗАПОВЕД **№ РД 09 – 2360/14.09.2020 г.**

**ПРОФЕСИОНАЛНО НАПРАВЛЕНИЕ: 481 „КОМПЮТЪРНИ НАУКИ“**

**ПРОФЕСИЯ: 481030 „ПРИЛОЖЕН ПРОГРАМИСТ”**

 **СПЕЦИАЛНОСТ: 4810301 „ПРИЛОЖНО ПРОГРАМИРАНЕ“**

**София, 2020 година**

1. **ОБЩО ПРЕДСТАВЯНЕ НА УЧЕБНАТА ПРОГРАМА**

Учебната програма по **учебна практика: интернет програмиране** е предназначена за специалност с код 4810301 „Приложно програмиране”, за вариантите, в които в типовите учебни планове е предвидено изучаването на учебния предмет **учебна практика: интернет програмиране** в 116 часа.

Програмата е разработена в съответствие с Държавния образователен стандарт за придобиване на квалификация по професията.

Учебното съдържание в програмата е структурирано в четири раздела, които дават възможност на учениците да получат знания за основите на Интернет и мрежовите протоколи, както и разработката на интернет приложения и принципите на уеб програмирането.

Обучението по предмета се извършва във взаимна връзка с предмети от отрасловата и специфичната професионална подготовка - „Бази данни”, „Разработка на Софтуер”, „Софтуерно инженерство”.

Обучението по предмета следва да се извърши, чрез използването на актуални средства и технологии за уеб разработка и с прилагането на проектно-базираното обучение.

1. **ЦЕЛИ НА ОБУЧЕНИЕТО ПО ПРЕДМЕТА**

Обучението по предмета има за цел учениците да придобият знания за същността на Интернет, разработката на уеб приложения и софтуерните средства, които се използват. За постигане на основната цел на обучението по **учебна практика: интернет програмиране** е необходимо изпълнението на следните подцели:

* придобиване на знания и умения, свързани с начина на работа на Интернет, сървър-клиент модела, както и начина на работа на уеб браузъра;
* Придобиване на знания и умения, свързани с мрежовите протоколи;
* Разбиране и боравене с различни видове заявки в HTTP;
* Придобиване на знания и умения, свързани с разработката на външен интерфейс за уеб приложения (front-end) чрез HTML, CSS, JavaScript и/или други подходящи технологии за реализиране на (front-end) частта на уеб приложенията;
* Придобиване на знания и умения за разработка на сървърна част на уеб приложения (back-end)
* Придобиване на знания и умения за свързване на БД в уеб приложение чрез директно използване на средствата на програмен език;
* Придобиване на знания и умения за свързване на БД в уеб приложение и реализирането на CRUD операции, използвайки обектно-релационно съпоставяща система (ORM);
* Познаване на основните видове уязвимости в уеб приложенията и на методите за защита от тях;
* Разбиране на принципите на работа на REST комуникацията със собствени и чужди API услуги.
1. **УЧЕБНО СЪДЪРЖАНИЕ**

1. Учебното съдържание е  структурирано в раздели  и теми. За всеки раздел  в програмата е определен минимален брой учебни часове. Учителят разпределя броя учебни часове за нови знания, упражнения и оценяване, при спазване изискванията за минимален брой часове по раздели.

2. Разликата между броя на учебните часове в учебния план и общия минимален брой, предвиден в учебната програма определя резерва часове. Те се разпределят по теми в началото на учебната година от учителя.

3. Раздели и теми

|  |  |  |
| --- | --- | --- |
| **№** | Наименование на разделите | Минимален брой часове |
| 1. | Въведение в Интернет, мрежови протколи и модел „Клиент - сървър” | 6 |
| 2. | Разработка на външен интерфейс (front-end) на уеб приложения | 20 |
| 3. | Разработка на сървърна часст (back-end) на уеб приложения | 20 |
| 4. | Създаване и работа с REST API | 12 |
| 5. | Създаване на цялостно приложение | 50 |
|   | **Общ минимален брой часове** | **108** |
|   | **Резерв часове** | **8** |
|   | **Общ брой часове** | **116** |

###### Раздел 1. Въведение в Интернет, мрежови протоколи и модел „Клиент - Сървър”

* 1. Основи на интернет. Мрежови протоколи. HTTP
	2. Видове HTTP заявки
	3. Клиент - сървър комуникация
	4. Практически проект: Реализиране на чат приложение в модел „Клиент-сървър”

###### Раздел 2. Разработка на външен интерфейс (front-end) на уеб приложения

* 1. Работа с HTML. Основни тагове
	2. Работа със CSS. Селектори и основни правила
	3. Работа с инструментите за разработчици на съвременните уеб браузъри
	4. Създаване на формуляри
	5. Създаване на семантични страници
	6. Създаване на адаптивно (responsive) оформление на страници
	7. Увод в JavaScript. Работа с обекти и събития
	8. Принципи на DOM. Манипулиране на DOM
	9. Практически проект: Реализиране на адаптивен фронт-енд за уеб сайт

**Раздел 3. Разработка на сървърна част (back-end) на уеб приложения**

* 1. Работа с MVC концепцията - модел, изглед, контролер
	2. Комуникация на БД в уеб приложение
	3. Работа с ORM (обектно-релационно съпоставящи) системи
	4. Реализиране на CRUD операции
	5. Създаване на шаблонни изгледи
	6. Управление на сесии и бисквитки
	7. Автентикация и авторизация
	8. Често срещани уязвимости в сигурността на уеб приложенията
	9. Практически проект: Реализиране на сървърна част (back-end) за информационна система

**Раздел 4. Създаване и работа с REST API**

* 1. Принципи на REST API. Създаване на собствено REST API.
	2. Работа с REST API в JSON/XML формат
	3. Консумиране на REST API от външния интерфейс (front-end) с помощта на AJAX
	4. Практически проект: Създаване и използване на REST API

**Раздел 5. Създаване на цялостно приложение**

* 1. Практически проект: Създаване на цялостно уеб приложение с адаптивен (responsive) външен интерфейс (front-end), сървърна част (back-end) и REST API
	2. Практически проект: Качване и конфигуриране на уеб приложение върху реален сървър
1. **ОЧАКВАНИ РЕЗУЛТАТИ ОТ УЧЕНЕТО – ЗНАНИЯ, УМЕНИЯ И КОМПЕТЕНТНОСТИ**

В края на обучението по учебния предмет учениците трябва да:

* + Познават същността на Интернет и мрежовите протоколи.
	+ Различават видовете HTTP заявки
	+ Разбират и прилагат HTTP като протокол за комуникация
	+ Познават семантичното структуриране на HTML страница
	+ Познават правилата за стилизиране на страница със CSS
	+ Разбират и прилагат принципите за създаване на адаптивен (responsive) дизайн на уеб сайтове
	+ Са наясно с механизма за изпращане на информация към сървъра чрез форми и съответните HTTP заявки
	+ Различават клиент и сървър
	+ Познават инструментите за разработчици в уеб браузърите
	+ Разбират модел „Клиент-сървър” и комуникацията между тях
	+ Притежават разбиране за принципите на MVC концепцията в изграждането на уеб приложения
	+ Познават принципите за създаване и работа с REST API и REST комуникация
	+ Умеят да свързват приложение с БД и да извършват основните CRUD операции
	+ Умеят да разработват динамични уеб сайтове със сървърна част (back-end) и външен (front-end) интерфейс (сървърна и клиентска част)
	+ Умеят да създават собствени REST API услуги
	+ Умеят да използват външни REST API услуги
1. **ОБЯСНИТЕЛНИ БЕЛЕЖКИ**

За целите на обучението по предмета следва да се използват актуални версии и стандарти както на уеб технологиите, така и на уеб браузърите.

Авторският колектив препоръчва за избор следните езици/технологии (актуални към 06.2020):

* + - HTML5
		- CSS3
		- JavaScript и Fetch API за реализиране на асинхронни заявки в JavaScript
		- Postman (софтуер за изпращане на HTTP заявки)
		- Google Chrome + Google Chrome Devtools за предпочитан уеб браузър за работа
		- За back-end едно от следните:
			* C# (ASP.NET Core)
			* Java (Spring MVC)
			* PHP (Laravel или Symfony framework)
			* Python (Django)
			* Node.js и подходящи библиотеки

В рамките на последния раздел учениците е препоръчително да работят по екипи върху задания, следвайки Agile (SCRUM) методологията, заложена в предмета „Софтуерно инженерство”. Препоръчително е УП по Интернет програмиране да се кара по такова време, че учениците вече да са запознати с Agile методологията и SCRUM.

1. **АВТОРСКИ КОЛЕКТИВ**

Програмата е разработена, обсъдена и оформена от експертна група към Национална програма „Обучение за ИТ кариера“ към МОН в състав:

1. доц. д-р Димитър Минчев, Бургаски свободен университет, Бургас
2. доц. д-р Ивайло Старибратов, ПУ „Паисий Хилендарски“, Пловдив
3. инж. Хриси Плачкова, МГ „Акад. Кирил Попов“, Пловдив
4. Петър Петров, ПГЕЕ „Константин Фотинов“, Бургас
5. Росен Вълчев, МГ „Акад. Кирил Попов“, Пловдив
6. **ЛИТЕРАТУРА**
7. Колисниченко Д., HTML 5 & CSS 3 - практическо програмиране за начинаещи. Трето издание, Асеневци, София, 2017, ISBN 9786197356113
8. Лонг Д., Java в облака. Spring Boot, Spring Cloud и Cloud Foundry, Асеневци, София, 2019, ISBN 9786197356649
9. Улман Л., PHP7 за Web. Бързо ръководство, АлексСофт, София, 2017, ISBN 9789546563408
10. Haverbeke M., Eloquent JavaScript, No Starch Press, San Francissco, 2018, ISBN 1593279507
11. Pro ASP.NET Core 3: Develop Cloud-Ready Web Applications Using MVC, Blazor, and Razor Pages 8th ed. Edition, Apress, New York, 2020, ISBN 1484254392
12. Learn ASP.NET Core 3: Develop modern web applications with ASP.NET Core 3, Visual Studio 2019, and Azure, 2nd Edition Paperback, Packt, Birmingham, UK, 2019, ISBN 1789610133
13. **Електронни ресурси**
14. https://www.w3schools.com/
15. https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-3.1
16. https://to6esko.github.io/ - Български превод на Eloquent JavaScript