

Конспект за Интернет програмиране

I. Общо представяне на учебната програма

Теория: ...

Практика: ...

II. Цели на обучението по предмета

...

III. Разпределение на учебното време

29 седмици x 6 часа = 174 учебни часа

IV. Учебно съдържание

№	Наименование на разделите	Минимален брой часове
1	Запознаване с TCP/IP и сокети <ul style="list-style-type: none">• Протоколи от TCP/IP protocol suite• Комуникация по сокет• Протокол WebSocket• Упражнения: работа с инструменти за анализ на мрежов трафик и мрежови протоколи	6
2	HTTP протокол <ul style="list-style-type: none">• HTTP заявки и отговори• HTTP хедъри• GET и POST методи• Упражнения с curl, Postman и други инструменти	6
3	Запознаване с HTML <ul style="list-style-type: none">• Език HTML: основни тагове• Създаване на уеб страници	12

	<ul style="list-style-type: none"> Форми, полета, изпращане на форми Упражнения: създаване на уеб страници и уеб форми 	
4	HTTP сървъри и сервиране на статични HTML файлове <ul style="list-style-type: none"> Инсталиране и конфигуриране на HTTP сървър Сервиране на статични файлове Упражнение: инсталация и настройка на HTTP сървър, сервиране на статично съдържание 	6
5	Запознаване с MVC технологична рамка <ul style="list-style-type: none"> Запознаване с някой MVC framework за уеб разработка Упражнение: изграждане на уеб приложение с използване на MVC framework 	12
6	Свързване на HTTP сървър с backend език за програмиране (PHP / Python / Java / C# / JS / други) <ul style="list-style-type: none"> Обработка на HTTP заявка и връщане на HTTP отговор Обработване на данни от HTML форма Упражнения: обработка на данни, изпратени от HTML форма 	6
7	Комуникация с база от данни <ul style="list-style-type: none"> Комуникация с база от данни посредством SQL или ORM заявки Реализиране на CRUD заявки (създаване, четене, промяна и изтриване на данни) в уеб приложенията Упражнения: изграждане на просто приложение с CRUD заявки (например TODO List) 	12
8	Шаблонни езици от страна на сървъра <ul style="list-style-type: none"> Генериране на HTML документи посредством backend темплейтен език (server-side templating) Преизползване на фрагменти Упражнения: изграждане на уеб приложение с използване на сървърен език за шаблони 	12
9	MVC архитектура <ul style="list-style-type: none"> Изграждане на мини MVC framework за уеб приложения: front controller, routing, views, controllers, models Разделяне на изградените до момента уеб приложения по MVC модела Упражнения: изграждане на мини MVC framework 	18

10	Управление на състоянието в уеб приложенията <ul style="list-style-type: none"> • HTTP сесии • Бисквитки (cookies) • Storage - къде пазим сесии и бисквитки? • Упражнения: имплементация на home / login / user home / logout 	6
11	Автентикация и авторизация <ul style="list-style-type: none"> • Login / logout в уеб приложение • Роли (администратор, потребител, анонимен) • Съхранение на пароли в back-end системи • Упражнения: имплементация на home / register / login / user home / admin home / logout 	6
12	Сигурност на уеб приложенията <ul style="list-style-type: none"> • Сигурност на уеб приложенията и добре известни атаки • XSS (cross site scripting) и HTML escaping • SQL injection и параметрични SQL заявки • CSRF (cross-site request forgery) и anti-forgery token • Най-важните 10 уязвимости според OWASP • Упражнения: <ul style="list-style-type: none"> ○ Инжектиране на скрипт чрез XSS атака ○ Промяна на базата данни чрез SQL injection атака ○ Промяна на базата данни чрез CSRF атака ○ Речникова атака на хеширани пароли 	6
13	Създаване REST API <ul style="list-style-type: none"> • Имплементация на CRUD API върху данни • HTTP статус кодове • Упражнения: имплементация на REST услуги с CRUD операции 	6
14	Консумиране на REST API <ul style="list-style-type: none"> • Извикване на REST заявки с JavaScript и AJAX • Упражнения: добавяне / изтриване на бележки в TODO list чрез AJAX заявка към Firebase или друг back-end service 	6
15	Внедряване на проект (deployment) <ul style="list-style-type: none"> • Качване на уеб приложение на хостинг, VPS или PaaS платформа 	6

	<ul style="list-style-type: none"> • Пренасяне на базата данни, кода на приложението, настройки • Упражнение: deployment на проект в Heroku или друга cloud платформа за приложения 	
16	Разработване на практически проект (индивидуално) <ul style="list-style-type: none"> • Разработка на практически уеб проект • Уеб приложение със сървърна част (back-end) и клиентска част (front-end) 	30
17	Защита на практически проект <ul style="list-style-type: none"> • Разработените проекти се защитават пред учителя 	12
	Резерв	6

V. Тематичен план

Като таблицата горе?

VI. Очаквани резултати от обучението

Покрива:

- РУ 10.6 - Използва релационни бази от данни
- РУ 10.7 - Разбира същността на клиент-сървърни архитектури
- РУ 10.8 - Познава различни начини за комуникация между системи
- РУ 12.1 - Разработва уеб front-end
- РУ 12.2 - Разработва сървърен уеб софтуер

VII. Литература

1. Fundamentals of Web Development, 2nd Edition, Ricardo Hoar, Pearson, 2018, ISBN 978-0133407150
2. Practical Web Development, Paul Wellens, Packt Publishing, 2015, ISBN 978-1782175919
3. Understanding Model-View-Controller, Stefano Borini, 2017, <https://stefanoborini.gitbooks.io/modelviewcontroller>
4. ASP.NET Core 2.0 MVC & Razor Pages for Beginners: How to Build a Website, Jonas Fagerberg, CreateSpace, 2017, ISBN 978-1979759953