

Конспект за Разработка на софтуер

162 учебни часа (първи срок: **18 седмици по 4 часа** + втори срок: **18 седмици по 5 часа**)

Идея за съдържание:

- Структура на софтуерен проект
 - Трислоен модел (three-tier architecture)
 - Слой за данни
 - Слой за услуги
 - Слой за UI

Изграждане на практически софтуерен проект

- Връзка между приложение и база данни, ORM технологии
- Потребителски интерфейс по избор (Web / desktop / mobile)

Да не се навлиза в технологии, за да се запази фокусът върху структурата на приложението.

I. Общо представяне на учебната програма

Обучението по предмета има за цел да предостави основната рамка за писане на софтуер и да съедини в себе си знанията от всичко ученото до сега в един голям проект.

II. Цели на обучението по предмета

Обучението по предмета има за цел да предостави основната рамка за писане на софтуер и да съедини в себе си знанията от всичко ученото до сега в един голям проект.

III. Разпределение на учебното време

I срок: 18 седмици x (2 часа теория + 2 часа практика) = 72 часа

II срок: 18 седмици x (2 часа теория + 3 часа практика) = 90 часа

IV. Учебно съдържание

№	Наименование на разделите	Минимален брой часове
1	<ul style="list-style-type: none">• Увод в разработката на софтуер – преглед на трислойния модел• Слой за данни (data access layer)• Слой за услуги (service layer)• Слой за потребителски интерфейс (UI layer)	9

	<ul style="list-style-type: none"> Упражнение: създаване на просто трислойно приложение (конзолно, без база данни) 	
2	<ul style="list-style-type: none"> Увод в концепцията за тестване – писане на "Unit" тестове Концепция за тестване на кода. Концепция за “регресия” и регресионни тестове. Покриване на кода с тестове. Упражнение: писане на unit тестове за интеграцията на слоя за услуги със слоя за данни. 	9
3	<ul style="list-style-type: none"> Увод в концепцията за дебъгване – откриване и отстраняване на проблеми Запознаване с различни методики за търсене на дефекти. Използване на дебъгер. Добри практики при дебъгване и отстраняване на дефекти - писане на тестове, които покриват проблема. Упражнение: дебъгване на код с грешки, отстраняване на грешките и писане на unit тестове. 	9
4	<p>Увод в концепцията за рефакториране и правене на "инкрементални промени"</p> <ul style="list-style-type: none"> Основни инструменти и похвати за рефакториране на код Запознаване с идеята за "style guides" и "linters" Запознаване с идеята за намиране на всички употреби на фрагмент код, навигиране към дефиниция на функция Добри практики при рефакториране чрез писане на тестове. Концепция за тестваем код Упражнение: рефакториране на недобре написан код, промяна на кода да стане тестваем, покриване с тестове 	18
5	<ul style="list-style-type: none"> Увод в инструментите за разработка – IDE, текстови редактори, командни инструменти Разлика между IDE и текстов редактор Техники за продуктивно използване на IDE - допълнителни разширения и клавишни комбинации Техники за продуктивно използване на текстов редактор - допълнителни разширения и клавишни комбинации. Използване на базови инструменти за търсене и промяна на кода. Упражнение: добавяне и използване на разширения в текстов редактор и в IDE 	4
6	<p>Използване на външни библиотеки. Управление на пакети</p> <ul style="list-style-type: none"> Търсене, инсталиране и употреба на външни библиотеки, които решават често срещани проблеми. Четене, ориентация и търсене в документация на външна библиотека. Запознаване с концепцията за инструменти за управление на пакети (package managers) Упражнение: инсталиране и използване на външни библиотеки 	9
7	<p>Свързване на приложения с бази от данни</p> <ul style="list-style-type: none"> Използване на "клиентски библиотеки" за свързване с база от данни 	27

	<ul style="list-style-type: none"> • Четене на конфигурационни файлове • Писане на SQL заявки през език за програмиране • Изпълнение на SQL заявки чрез език за програмиране, изпълнение на SQL команди и извличане на резултата от тях • Запознаване с идеята зад "prepared statements" • Създаване на CRUD приложения върху база от данни • Концепция за обектно-реляционно съответствие (ORM frameworks) • Упражнения: създаване на CRUD приложение с база от данни с SQL команди (трислойно приложение, конзолно) • Упражнения: създаване на CRUD приложение с база от данни с ORM и (трислойно приложение, конзолно) 	
8	Създаване на приложения с няколко потребителски интерфейса <ul style="list-style-type: none"> • Разделяне на приложението на три ясно отделени проекта (или библиотеки с класове): за данни, за услуги и за потребителски интерфейс. • Правилно структуриране на проекта: изолиране на бизнес логика, презентационна логика и логика за достъп до данни в съответните проекти (библиотеки с класове) • Създаване на поне два различни потребителски интерфейса за едно и също приложение • Упражнение: изграждане на трислойно приложение с конзолен / GUI / друг презентационен слой 	18
9	Създаване на курсова работа в екип <ul style="list-style-type: none"> • Трислойно приложение със слой за данни, слой за услуги и потребителски интерфейси • Поне два потребителски интерфейса • Използване на база данни • Използване на външни библиотеки • Пълно покритие с unit тестове • Спазване на "style guide" 	36 (4 седмици, ~ 1 месец)
10	Защита на курсова работа <ul style="list-style-type: none"> • Преглед на кода за спазване на добрите практики (code review) • Презентация и защита на проекта 	18
	Резерв часове	5

V. Тематичен план

Като таблицата горе?

VI. Очаквани резултати от обучението (РУ)

Покриват се следните:

- РУ10.5
- РУ10.6
- РУ13.1
- РУ13.2

VII. Авторски колектив

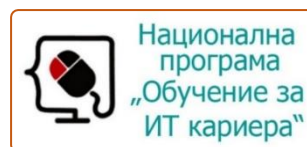
- Радослав Георгиев, Ивайло Бъчваров, <https://hackbulgaria.com>, <https://hacksoft.io>

VIII. Литература

- Software Development, Design and Coding: With Patterns, Debugging, Unit Testing, and Refactoring 2nd edition, John F. Dooley, Apress, 2017, ISBN 978-1484231524
- Extreme Programming Explained: Embrace Change, 2nd Edition, Kent Beck, Addison-Wesley, 2005, ISBN 978-0321278654
- Програмиране 101 към Хак България - <https://github.com/HackBulgaria/Programming101-Python-2016>

Министерство на образованието и науката (МОН)

- Настоящият курс (презентации, примери, задачи, упражнения и др.) е разработен за нуждите на Национална програма "Обучение за ИТ кариера" на МОН за подготовка по професия "Приложен програмист".



- Курсът е базиран на учебно съдържание и методика, предоставени от фондация "Софтуерен университет" и се разпространява под **свободен лиценз CC-BY-NC-SA** (Creative Commons Attribution-Non-Commercial-Share-Alike 4.0 International).

