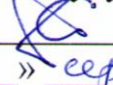


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНБАСЬКА ДЕРЖАВНА МАШИНОБУДІВНА АКАДЕМІЯ

Кафедра «Автоматизація виробничих процесів»


Затверджую:

Декан факультету машинобудування

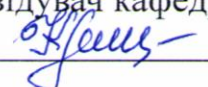

Касов В. Д.
« 31 » серпня 2020р.



Гарант освітньої програми:
канд. тех. наук, доцент


Суботін О.В.
« 22 » червня 2020р.

Розглянуто і схвалено
на засіданні кафедри
автоматизації виробничих
процесів
Протокол № 10 від 22.06 2020р.

Завідувач кафедри

Клименко Г.П.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

„WEB - ПРОГРАМУВАННЯ”

(назва дисципліни)

галузь знань	12 – «Інформаційні технології»
спеціальність	123 – «Комп'ютерна інженерія»
освітній рівень	перший (бакалаврський)
ОПП	«Комп'ютерні системи та мережі»
Факультет	«Машинобудування»

Розробник: Періг О.В., канд. техн. наук, доцент

Краматорськ – 2020 р.

1. Опис навчальної дисципліни «Web-програмування»

Показники		Галузь знань, спеціальність, ОПП (ОНП), професійне (наукове) спрямування, рівень вищої освіти	Характеристика навчальної дисципліни	
			денна	денна (прискорена)
Кількість кредитів		Галузь знань: 12 «Інформаційні технології». Спеціальність: 123 «Комп'ютерна інженерія». ОПП «Комп'ютерні системи та мережі»	Дисципліна вільного вибору	
6	5,5			
Загальна кількість годин				
180	165			
Модулів – 1		РГР 3 [Розробити CRUD-застосунок для роботи з базою даних певної організації (фірми, заводу, бібліотеки, школи, університету, лікарні) із застосуванням можливостей РНР-фреймворку Laravel і JS-фреймворку Vue]	Рік підготовки	
Змістових модулів – 1			4	2
Індивідуальні розрахунково-графічні завдання: РГР 1 [Розробити комп'ютерні коди та письмові звіти за шістьма лекційними темами];			Семестр	
			7	3
			РГР 2 [Розробити комп'ютерні коди та письмові звіти за сімома темами практичних занять];	
Тижневих годин для <u>денної</u> форми навчання: аудиторних – 4; самостійної роботи студента – 8 (денна); 7 (денна прискорена)		Рівень вищої освіти: <u>перший</u> (бакалаврський)	Лекції	
			30	30
			Практичні	
			30	30
			Лабораторні	
			0	0
			Самостійна робота	
			120	105
		Вид контролю		
		залік	залік	

2. Загальні відомості, мета і завдання навчальної дисципліни «Web-програмування»

Освітня компонента «Web-програмування» для ОПП «Комп'ютерні системи та мережі» – це навчальна дисципліна, зосереджена на прикладному ознайомленні студентів із актуальними термінологією, синтаксисом та семантикою серверних та безсерверних мов сучасної Інтернет-інженерії.

Мета дисципліни «Web-програмування» – формування когнітивних, афективних та моторних компетенцій, а також “твердих” та “м'яких” навичок майбутнього Інтернет-інженера, добре обізнаного з термінологією та синтаксисом сучасних мов Мережі та достатньо кваліфікованого для успішного практичного використання широкого спектру сучасних редакторів, середовищ та фреймворків для індивідуальної роботи з **HTML-, CSS-, PHP-** та **JS-**файлами.

Завдання дисципліни «Web-програмування»:

– ознайомлення студентів-системотехніків із сучасними міжнародними тенденціями та комп'ютерно-синтаксичними трендами до поступового переосмислення усталених існуючих патернів класичної системно-мережної та синтаксично-семантичної діяльності фахівця з комп'ютерної інженерії з огляду на швидкий розвиток та безкоштовну доступність потужних обчислювальних можливостей широкого спектру хмарних обчислювальних середовищ та сучасних багатомовних фреймворків та середовищ Веб-програмування для системно-мережного опису складних процесів у клієнтсько-серверних системах;

– розширення професійних уявлень майбутнього фахівця з комп'ютерної інженерії щодо ефективних шляхів прикладного застосування розрахунково-обчислювальних можливостей та синтаксично-семантичних особливостей сучасних редакторів, середовищ та фреймворків Веб-програмування для ефективної індивідуальної роботи з **HTML-, CSS-, PHP-** та **JS-**файлами в рамках створення багаторівневих та багатомовних Веб-проектів широкого призначення, у тому числі в рамках системно-мережного кіберфізичного моделювання нелінійних системно-мережних процесів у складних керованих соціально-технічних системах людино-мережної взаємодії;

– ознайомлення старшокурсників із класичними та сучасними підходами, концепціями, мовами, методами та системно-мережними Веб-алгоритмами для ефективного синтаксично-семантичного Веб-програмування та низькорівневої оптимізації режимів практичного системно-мережевого функціонування наявного та/або мінімально-модернізованого серверно-мережного обладнання;

– розширення професійних уявлень майбутнього фахівця із комп'ютерних систем та мереж щодо шляхів ефективного інформаційного застосування синтаксично-семантичних можливостей сучасних мов та фреймворків до успішного розв'язку Веб-задач прикладного рівня (**OSI-Application Layer**) комп'ютерно-мережевого інжинірингу в рамках низькорівневого системно-мережного моделювання соціально-регламентованих

динамічних особливостей нелінійних інформаційно-потоків процесів у складних багатосерверних техносоціальних інформаційно-мережних системах масового обслуговування;

– докладне дидактичне висвітлення бакалаврам-старшокурсникам 123ї спеціальності широкого спектру Веб-мов та Веб-бібліотек для синтаксично-семантичного розв'язку численних прикладних інформаційно-потоків комп'ютерно-мережних задач для регульованих соціально-технічних процесів та кіберсистем, безпосередньо пов'язаних з нелінійною феноменологічною та/або статистичною інформаційно-мережною динамікою системно-адміністративних та комп'ютерно-інженерних комунікацій у професійній технічній спільноті для всіх мультидисциплінарних областей адміністративно-менеджерської діяльності на усіх рівнях системно-мережного інжинірингу та адміністрування;

– ознайомлення студентів бакалаврату 123ї спеціальності із Веб-синтаксичними та системно-мережевими можливостями Веб-опису нелінійних соціально-технічних задач мережної динаміки із щоденної практичної діяльності комп'ютерного інженера в рамках широкого застосування найбільш поширених Веб-мовних підходів до створення багаторівневих та багатомовних Веб-проектів широкого призначення в рамках індивідуальної та колективної роботи Веб-девелоперів над створенням та удосконаленням коду у авторських **HTML-, CSS-, PHP-** та **JS-**файлах;

– ознайомлення майбутніх інженерів 123ї спеціальності із сучасними міжнародними Веб-синтаксичними підходами, Веб-семантичними парадигмами, багаторівневими кіберфізичними підходами, Інтернет-інжиніринговими та обчислювально-мережевими концепціями, динамічними мережно-синтаксичними критеріями та комп'ютерно-моделювальними підходами до комплексної багаторівневої оцінки якості інформаційно-комунікаційних сервісів та послуг для різних категорій користувачів високошвидкісних Інтернет-каналів та з вітчизняною реалізацією сучасних багатокритеріальних стратегій дотримання якості в роботі фахівця з комп'ютерних систем та мереж;

– ознайомлення майбутніх фахівців з Веб-програмування мережних процесів та систем із актуальними соціально-етичними проблемами ефективного дотримання міжнародних етичних принципів для успішного забезпечення стандартів сталості, добробуту, робочого комфорту та доброчесності в повсякденній практичній діяльності як інженера з комп'ютерних систем та мереж, так і системно-мережного адміністратора на виробництві;

– ознайомлення студентів-комп'ютерників із людино-центричними та соціально-«м'якими» підходами сучасного комп'ютерного системно-мережного Веб-програмування до постійного розвитку емпатії, комунікабельності, людяності та *soft skills* у професійній спільноті ІТ-фахівців шляхом соціальної транспозиції нелінійних моделей феноменологічної мережної динаміки;

– ознайомлення студентів зі шляхами становлення та розвитку «твердих» професійних (*hard skills*) та «м'яких» соціально-комунікаційних (*soft skills*) навичок майбутнього фахівця з комп'ютерних систем та мереж упродовж індивідуального і командного практичного вивчення курсу Веб-програмування;

– додаткове формування у бакалаврів-старшокурсників 123ї спеціальності стійкої та впевненої здатності до Веб-багаторівневого, синтаксично-багатомовного, алгоритмічно-послідовного, акаузально-логічного, інженерно-технічного та системно-мережного творчого мислення; ефективне сприяння успішному становленню належного рівня соціально-менеджерської, інформаційно-професійної та інформаційно-обчислювальної комп'ютерно-мережної культури, а також забезпечення подальшого розширення професійного соціально-технічного кругозору майбутнього інженера-практика в прикладній Веб-синтаксичній міждисциплінарній науково-технічній галузі системно-мережної комп'ютерної інженерії інформаційно-мережних потоків даних у високошвидкісних каналах сучасних інформаційно-комунікаційних мереж.

Передумови для ефективного вивчення дисципліни «Web-програмування»: попереднє вивчення студентами 123ї спеціальності дисциплін «Фізика», «Електричні вимірювання та прилади», «Електроніка та комп'ютерна схемотехніка», «Компоненти сучасних комп'ютерних систем», «Основи комп'ютерної інженерії», «Апаратні та програмні засоби комп'ютерної інженерії», «Технологія проектування комп'ютерних систем», «Вища математика», «Чисельні методи і моделювання на ЕОМ», «Основи системного аналізу», «Системи комп'ютерної алгебри», «Теорія ймовірностей, ймовірнісні процеси і математична статистика», «Теорія інформації та кодування», «Теорія алгоритмів та автоматів», «Комп'ютерні технології та програмування», «Комп'ютерна практика», «Інженерія програмного забезпечення», «Системне програмування», «Інформаційні мережі», «Технологія програмування складних систем», «Системи штучного інтелекту та інтелектуальний аналіз даних», «Ділова риторика», «Психологія», «Технології психічної саморегуляції та взаємодії», «Соціологія», «Філософія», «Етика та естетика», «Професійна етика», «Інтелектуальна власність», «Історія Української культури», «Політологія» та «Інформаційні війни».

Мова викладання: українська.

Обсяг навчальної дисципліни «Web-програмування» та його розподіл за видами навчальних занять:

– загальний обсяг для **денної форми навчання** становить 180 годин / 6,0 кредитів, в тому числі: лекції – 30 годин, лабораторні роботи – не плануються, практичні заняття – 30 годин, самостійна робота студентів – 120 годин; курсовий проект – курсова робота не планується;

– загальний обсяг для **денної прискореної форми навчання** становить 165 годин / 5,5 кредитів, в тому числі: лекції – 30 годин, лабораторні роботи – не плануються, практичні заняття – 30 годин, самостійна робота студентів – 105 годин; курсовий проект – курсова робота не планується.

3. Програмні результати навчання з дисципліни «Web-програмування»

Освітня компонента «**Web-програмування**» (ВК21) повинна сформувати наступні програмні результати навчання, що передбачені освітньо-професійною програмою підготовки бакалаврів спеціальності «Комп'ютерна інженерія», які навчаються за ОПП «Комп'ютерні системи та мережі»:

- ЗК1. Здатність до абстрактного мислення, аналізу і синтезу.
- ЗК2. Здатність вчитися і оволодівати сучасними знаннями.
- ЗК3. Здатність застосовувати знання у практичних ситуаціях.
- ЗК7. Вміння виявляти, ставити та вирішувати проблеми.

ФК1. Здатність застосовувати законодавчу та нормативно-правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі комп'ютерної інженерії.

ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

ФК3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

ФК18. Знання принципів, методів і алгоритмів комп'ютерної графіки, уміння застосовувати їх під час розробки графічних інтерфейсів взаємодії людини з комп'ютером.

ПР3. Знати новітні технології в галузі комп'ютерної інженерії.

ПР21. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.

ПР25. Здатність застосовувати методи і алгоритми комп'ютерної графіки та сучасні технології автоматизації проектування складних систем, продуктів і сервісів інформаційних технологій, сучасні парадигми та мови програмування.

4. Програма та структура навчальної дисципліни «Web-програмування»

Денна прискорена форма навчання – «Web-програмування»

Розподіл обсягу дисципліни «Web-програмування» **за видами навчальних занять** для **денної прискореної форми навчання** наведено нижче:

Вид навчальних занять або контролю	Розподіл між учбовими тижнями								
	1	2	3	4	5	6	7	8	9
Л – Лекції	2	2	2	2	2	2	2	2	2
П. р. – Практичні	2	2	2	2	2	2	2	2	2
Л. р. – Лабораторні роб.	0	0	0	0	0	0	0	0	0
Сам. робота	7	7	7	7	7	7	7	7	7
Контр. роботи						КР1			
Модулі	Модуль 1								
Контроль по модулю	ПР1 РГР1 РГР2 РГР3	ПР1 РГР1 РГР2 РГР3	ПР2 РГР1 РГР2 РГР3	ПР2 РГР1 РГР2 РГР3	ПР3 РГР1 РГР2 РГР3	ПР3 КР1 РГР1 РГР2 РГР3	ПР4 РГР1 РГР2 РГР3	ПР4 РГР1 РГР2 РГР3	ПР5 РГР1 РГР2 РГР3

Вид навчальних занять або контролю	Розподіл між учбовими тижнями					
	10	11	12	13	14	15
Л – Лекції	2	2	2	2	2	2
П. р. – Практичні	2	2	2	2	2	2
Л. р. – Лабораторні роб.	0	0	0	0	0	0
Сам. робота	7	7	7	7	7	7
Контр. роботи			КР2			КР3
Модулі	Модуль 1					
Контроль по модулю	ПР5 РГР1 РГР2 РГР3	ПР6 РГР1 РГР2 РГР3	ПР6 КР2 РГР1 РГР2 РГР3	ПР7 РГР1 РГР2 РГР3	ПР7 РГР1 РГР2 РГР3	КР3 РГР1 РГР2 РГР3

Л – Лекція; ПР – практична робота; КР – контрольна робота; РГР – розрахунково-графічна робота; М – модуль.

Денна форма навчання – «Web-програмування»

Розподіл обсягу дисципліни «Web-програмування» за видами навчальних занять для денної форми навчання наведено нижче:

Вид навчальних занять або контролю	Розподіл між учбовими тижнями								
	1	2	3	4	5	6	7	8	9
Л – Лекції	2	2	2	2	2	2	2	2	2
П. р. – Практичні	2	2	2	2	2	2	2	2	2
Л. р. – Лабораторні роб.	0	0	0	0	0	0	0	0	0
Сам. робота	8	8	8	8	8	8	8	8	8
Контр. роботи						КР1			
Модулі	Модуль 1								
Контроль по модулю	ПР1 РГР1 РГР2 РГР3	ПР1 РГР1 РГР2 РГР3	ПР2 РГР1 РГР2 РГР3	ПР2 РГР1 РГР2 РГР3	ПР3 РГР1 РГР2 РГР3	ПР3 КР1 РГР1 РГР2 РГР3	ПР4 РГР1 РГР2 РГР3	ПР4 РГР1 РГР2 РГР3	ПР5 РГР1 РГР2 РГР3

Вид навчальних занять або контролю	Розподіл між учбовими тижнями					
	10	11	12	13	14	15
Л – Лекції	2	2	2	2	2	2
П. р. – Практичні	2	2	2	2	2	2
Л. р. – Лабораторні роб.	0	0	0	0	0	0
Сам. робота	8	8	8	8	8	8
Контр. роботи			КР2			КР3
Модулі	Модуль 1					
Контроль по модулю	ПР5 РГР1 РГР2 РГР3	ПР6 РГР1 РГР2 РГР3	ПР6 КР2 РГР1 РГР2 РГР3	ПР7 РГР1 РГР2 РГР3	ПР7 РГР1 РГР2 РГР3	КР3 РГР1 РГР2 РГР3

Л – Лекція; ПР – практична робота; КР – контрольна робота; РГР – розрахунково-графічна робота; М – модуль.

Розподіл обсягу дисципліни «Web-програмування» за темами навчальних занять для денної форми навчання наведено нижче, причому цифрами в дужках зазначаються години для денної прискореної форми навчання:

Найменування розділів, тем та семестрових атестацій дисципліни «Web-програмування»	Всього	Розподіл за темами та за видами занять					
		Аудиторні заняття				Самост робота	
		Всього	Лекції	Лаб.	Практ.	Всього	У т. ч. ІСЗ
<p>□ Тема 1. Вступ. Принципи роботи мережі. W3C об'єктна модель документа HTML DOM (<i>Document Object Model</i>). Каскадні таблиці стилів CSS та CSS3 (<i>Cascading Style Sheets</i>). CSS-препроцесори {шаблонізатори}: ▪ скриптова метамова SASS (<i>Syntactically Awesome Stylesheets</i>); ▪ скриптова метамова SCSS (<i>Sassy CSS; Sassy Cascading Style Sheets</i>) та ▪ динамічна мова стилів LESS (<i>Leaner Style Sheets</i>).</p>	Для <u>денної форми навчання</u>						
	30	10	5	0	5	20	20
	Для <u>денної прискореної форми навчання</u>						
	27,5	10	5	0	5	17,5	17,5
<p>□ Тема 2. PHP-Серверна сторона. Гіпертекстовий препроцесор PHP (<i>Hypertext Preprocessor</i>). Розробка сервера на PHP. PHP-черги запитів (<i>PHP queue</i>). Робота з реляційними базами даних {MySQL; SQL} в PHP. Робота з нереляційними базами даних {MongoDB} в PHP. Робота з пошуковим індексом Elasticsearch в PHP. PHP-фреймворк Laravel. Laravel-черги (<i>Laravel Queues</i>). PHP-фреймворк Symfony.</p>	Для <u>денної форми навчання</u>						
	30	10	5	0	5	20	20
	Для <u>денної прискореної форми навчання</u>						
	27,5	10	5	0	5	17,5	17,5
<p>□ Тема 3. Типи, підходи та Веб-застосування прикладного програмного інтерфейсу API (<i>Application Programming Interface</i>). Підхід REST (RESTful; <i>Representational state transfer</i> {<i>передача репрезентативного стану</i>}) API до побудови програмного інтерфейсу Веб-застосунків. Підхід SOAP (<i>Simple Object Access Protocol</i> {<i>простий протокол доступу до об'єктів</i>}) API до побудови програмного інтерфейсу Веб-застосунків. Мова програмування API-запитів GraphQL. Життя і смерть технології API BaaS EOL (<i>API Backend as a Service End of</i></p>	Для <u>денної форми навчання</u>						
	30	10	5	0	5	20	20
	Для <u>денної прискореної форми навчання</u>						
	27,5	10	5	0	5	17,5	17,5

<i>Life</i>) {effective June 30, 2019}. Технології Backendless REST API та Counter API .							
<input type="checkbox"/> Тема 4. JS -Клієнтська сторона. Прототипно-орієнтована скриптова мова JavaScript (JS) . Основи та особливості написання скриптів у клієнтському JavaScript . JavaScript-фреймворк {бібліотека} AngularJS . JavaScript-фреймворк {бібліотека} Vue.js . JavaScript-фреймворк {бібліотека} React .	Для <u>денної форми навчання</u>						
	30	10	5	0	5	20	20
	Для <u>денної прискореної форми навчання</u>						
	27,5	10	5	0	5	17,5	17,5
<input type="checkbox"/> Тема 5. Пакетні менеджери {та системи багатомодульної розробки} { <i>PHP-based</i> } Composer та { <i>JS-based</i> } Node.js . <i>Git</i> -розподілена система контролю версій вихідного коду та керування версіями файлів (Git Version Control). JWT -авторизація за стандартом токена доступу на основі JSON (<i>JSON Web Token</i>). Системи контейнеризації Docker та Docker compose . Docker { <i>Compose</i> }-докеризація бази даних (database), бекенда (backend) та фронтенда (frontend) { Dockerizing Front- & Backend }. Методи автентифікації, куки (cookies) та сесії у React, Express та Docker .	Для <u>денної форми навчання</u>						
	30	10	5	0	5	20	20
	Для <u>денної прискореної форми навчання</u>						
	27,5	10	5	0	5	17,5	17,5
<input type="checkbox"/> Тема 6. Веб-програмування хмарних обчислювальних середовищ. Безсерверні обчислення (Serverless Computing). Мова Веб-програмування Go .	Для <u>денної форми навчання</u>						
	30	10	5	0	5	20	20
	Для <u>денної прискореної форми навчання</u>						
	27,5	10	5	0	5	17,5	17,5
Всього для <u>денної форми навчання</u>	180	60	30	0	30	120	120
Всього для <u>денної прискореної форми навч.</u>	165	60	30	0	30	105	105

Лекції з навчальної дисципліни «Web-програмування»

№ лек	Зміст тем, лекцій, дидактичних засобів з навчальної дисципліни «Web-програмування»	Год. (д/дп)	Літ.
1	2	3	4
1	<p>□ Лекційна тема 1. Вступ. Принципи роботи мережі. W3C об'єктна модель документа HTML DOM (<i>Document Object Model</i>). Каскадні таблиці стилів CSS та CSS3 (<i>Cascading Style Sheets</i>). CSS-препроцесори {шаблонізатори}: ▪ скриптова метамова SASS (<i>Syntactically Awesome Stylesheets</i>); ▪ скриптова метамова SCSS (<i>Sassy CSS; Sassy Cascading Style Sheets</i>) та ▪ динамічна мова стилів LESS (<i>Leaner Style Sheets</i>).</p> <p>Корисні Веб-посилання до Теми 1 HTML DOM та CSS:</p> <ul style="list-style-type: none"> □) https://developer.mozilla.org/en-US/docs/Learn □) https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/How_the_Web_works □) https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/Installing_basic_software □) https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/HTML_basics □) https://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model/Introduction □) https://www.w3schools.com/js/js_htmlDOM.asp □) https://frontender.info/dom/ □) https://learn.javascript.ru/dom-nodes □) https://htmlacademy.ru/courses/299 □) https://htmlacademy.ru/courses/26/run/13 □) http://diveintohtml5.net/ □) https://livebook.manning.com/book/hello!-html5-&-css3/ □) https://livebook.manning.com/book/html5-in-action/ □) https://livebook.manning.com/book/html5-for-net-developers □) https://www.w3schools.com/css/ □) https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/CSS_basics □) https://developer.mozilla.org/en-US/docs/Archive/CSS3 □) https://css.in.ua/css/properties □) https://habr.com/ru/post/140612/ □) https://habr.com/ru/post/144309/ □) https://drops.dagstuhl.de/opus/volltexte/2017/7943/pdf/OASlcs- 	<p>5 (д)</p> <hr/> <p>5 (дп)</p>	<p>[2], [3], [4], [8], [10], [13], [14], [21], [36].</p>

	<p>SLATE-2017-8.pdf</p> <ul style="list-style-type: none"> □ https://codeguida.com/post/553 □ https://htmlacademy.ru/courses/26/run/10 □ https://freebiesbug.com/psd-freebies/website-template/ □ https://www.freepik.com/psd/web-templates □ https://freebiesbug.com/?s=blog □ https://livebook.manning.com/book/css-in-depth □ https://livebook.manning.com/book/web-performance-in-action/ □ https://www.pdfdrive.com/learn-css-in-one-day-and-learn-it-well-includes-html5-css-for-beginners-with-hands-on-project-the-only-book-you-need-to-start-coding-in-css-coding-fast-with-hands-on-project-volume-2-e196802719.html □ https://www.pdfdrive.com/html-css-crash-course-learn-html-and-css-with-easy-to-follow-step-by-step-tutorials-e158240321.html □ https://www.pdfdrive.com/learning-web-design-a-beginners-guide-to-html-css-javascript-and-web-graphics-e188549005.html □ https://www.pdfdrive.com/pro-html5-games-learn-to-build-your-own-games-using-html5-and-javascript-e158220357.html □ https://www.pdfdrive.com/build-your-own-2d-game-engine-and-create-great-web-games-using-html5-javascript-and-webgl-e166780734.html <p><i>Застосовуються авторські слайди оригінальних PowerPoint презентацій та мультимедійний проектор.</i></p>		
2	<p>□ Лекційна тема 2. PHP-Серверна сторона. Гіпертекстовий препроцесор PHP (<i>Hypertext Preprocessor</i>). Розробка сервера на PHP. PHP-черги запитів (<i>PHP queue</i>). Робота з реляційними базами даних {MySQL; SQL} в PHP. Робота з нереляційними базами даних {MongoDB} в PHP. Робота з пошуковим індексом Elasticsearch в PHP. PHP-фреймворк Laravel. Laravel-черги (<i>Laravel Queues</i>). PHP-фреймворк Symfony.</p> <p>Корисні Веб-посилання до Теми 2 PHP, MySQL, MongoDB, Laravel та Symfony:</p> <ul style="list-style-type: none"> □ https://www.pdfdrive.com/php-mysql-javascript-all-in-one-for-dummies-e90592496.html □ https://livebook.manning.com/book/php-in-action/ □ https://www.pdfdrive.com/learning-php-e34618311.html □ https://www.pdfdrive.com/learn-php-7-object-oriented-modular-programming-using-html5-css3-javascript-xml-json-and-mysql-e175321455.html 	<p>5 [2], (д) [8], — [9], 5 [10], (дп) [12], [15], [16], [17], [21], [22], [23], [24], [28], [29].</p>	

-) <https://www.pdfdrive.com/php-7-Наиболее-полное-руководство-e187513614.html>
-) <https://www.pdfdrive.com/php-objects-patterns-and-practice-e188490653.html>
-) <https://www.pdfdrive.com/php-7-real-world-application-development-e187686458.html>
-) <https://www.pdfdrive.com/pro-php-and-jquery-2nd-edition-take-your-dynamic-php-7-apps-to-the-next-level-by-adding-a-javascript-layer-using-jquery-e168116222.html>
-) <https://www.pdfdrive.com/practical-php-7-mysql-8-and-mariadb-website-databases-a-simplified-approach-to-developing-database-driven-websites-e187206125.html>
-) <https://www.w3schools.com/php/default.asp>
-) <https://www.php.net/manual/ru/class.ds-queue.php>
-) <https://www.phpclasses.org/blog/post/332-How-to-Use-Queue-To-Speedup-PHP-Processing-Tasks-Part-1-Queueing-Slow-Tasks.html>
-) <https://dev.to/rafaacioly/working-with-queues-and-stacks-in-php--39ch>
-) <https://www.php.net/manual/ru/function.msg-get-queue.php>
-) <https://ruseller.com/lessons.php?id=1664>
-) <https://htmlacademy.ru/tutorial/php/mysql>
-) <https://www.pdfdrive.com/expert-php-and-mysql-e23855036.html>
-) <https://www.pdfdrive.com/practical-php-and-mysql-website-databases-e19716961.html>
-) <https://www.pdfdrive.com/learning-mysql-e17258694.html>
-) <https://www.pdfdrive.com/learning-mysql-and-mariadb-heading-in-the-right-direction-with-mysql-and-mariadb-e167805247.html>
-) <https://www.pdfdrive.com/a-web-based-introduction-to-programming-essential-algorithms-syntax-and-control-structures-using-php-html-and-mariadbmysql-e193157917.html>
-) <https://habr.com/ru/company/microsoft/blog/151773/>
-) <https://www.php.net/manual/ru/security.database.sql-injection.php>
-) <https://livebook.manning.com/book/sql-server-dmvs-in-action/about-this-book/>
-) <https://livebook.manning.com/book/sql-server-mvp-deep-dives/>
-) <https://habr.com/ru/post/103699/>
-) <https://habr.com/ru/post/310254/>
-) <https://www.php.net/manual/ru/set.mongodb.php>
-) <https://livebook.manning.com/book/mongodb-in-action-second-edition>
-) <https://www.pdfdrive.com/mongodb-in-action-covers-mongodb-version-30-d176027685.html>

-) <https://livebook.manning.com/book/getting-mean-with-mongo-express-angular-and-node>
-) <https://livebook.manning.com/book/getting-mean-with-mongo-express-angular-and-node-second-edition/>
-) <https://www.pdfdrive.com/practical-mongodb-architecting-developing-and-administering-mongodb-e179275727.html>
-) <https://www.pdfdrive.com/data-science-fundamentals-for-python-and-mongodb-d158561929.html>
-) <https://www.elastic.co/guide/en/elasticsearch/client/php-api/current/index.html>
-) <https://www.pdfdrive.com/elasticsearch-the-definitive-guide-e26858592.html>
-) <https://livebook.manning.com/book/elasticsearch-in-action/>
-) <https://www.pdfdrive.com/elasticsearch-in-action-e38385853.html>
-) <https://livebook.manning.com/book/relevant-search>
-) <https://www.pdfdrive.com/relevant-search-with-applications-for-solr-and-elasticsearch-e158065972.html>
-) <https://www.pdfdrive.com/elasticsearch-cookbook-2nd-edition-over-130-advanced-recipes-to-search-analyze-deploy-manage-and-monitor-data-effectively-with-elasticsearch-e176029722.html>
-) <https://www.pdfdrive.com/elasticsearch-5x-cookbook-e184787982.html>
-) <https://www.pdfdrive.com/mastering-elasticsearch-5x-master-the-intricacies-of-elasticsearch-5-and-use-it-to-create-flexible-and-scalable-search-solutions-e200696189.html>
-) <https://www.pdfdrive.com/beginning-laravel-a-beginners-guide-to-application-development-with-laravel-53-e188492564.html>
-) <https://www.pdfdrive.com/learning-laravel-5-e38609488.html>
-) <https://www.pdfdrive.com/laravel-up-running-a-framework-for-building-modern-php-apps-e189833029.html>
-) <https://www.pdfdrive.com/easy-laravel-5-a-hands-on-introduction-using-a-real-world-project-e158463040.html>
-) <https://www.pdfdrive.com/design-patterns-in-php-and-laravel-e55291610.html>
-) <https://laravel.su/docs/5.0/queues>
-) <https://laravel.ru/docs/v5/queues>
-) <https://laravel.com/docs/8.x/queues>
-) <https://divinglaravel.com/queue-workers-how-they-work>
-) <https://www.pdfdrive.com/building-php-applications-with-symfony-cakephp-and-zend-frameworkpdf-e20299947.html>
-) <https://www.pdfdrive.com/symfony-certification-e58450074.html>

	Застосовуються авторські слайди оригінальних PowerPoint презентацій та мультимедійний проектор.		
3	<p>□ Лекційна тема 3. Типи, підходи та Веб-застосування прикладного програмного інтерфейсу API (<i>Application Programming Interface</i>). Підхід REST (RESTful; <i>Representational state transfer</i> {передача репрезентативного стану}) API до побудови програмного інтерфейсу Веб-застосунків. Підхід SOAP (<i>Simple Object Access Protocol</i> {простий протокол доступу до об'єктів}) API до побудови програмного інтерфейсу Веб-застосунків. Мова програмування API-запитів GraphQL. Життя і смерть технології API BaaS EOL (<i>API Backend as a Service End of Life</i>) {effective June 30, 2019}. Технології Backendless REST API та Counter API.</p> <p>Корисні Веб-посилання до Теми 3 API, REST, SOAP та GraphQL:</p> <ul style="list-style-type: none"> □) https://cloud.google.com/apigee □) https://docs.apigee.com/ □) https://www.soapui.org/learn/api/soap-vs-rest-api/ □) https://habr.com/ru/post/326986/ □) https://habr.com/ru/post/464261/ □) https://habr.com/ru/post/483202/ □) https://developer.mozilla.org/en-US/docs/Web/API □) https://quality-lab.ru/blog/rest-api-testing/ □) https://quality-lab.ru/blog/soap-api-testing/ □) https://www.altexsoft.com/blog/engineering/what-is-soap-formats-protocols-message-structure-and-how-soap-is-different-from-rest/ □) https://mockapi.io/ □) https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch □) https://livebook.manning.com/book/the-design-of-web-apis □) https://livebook.manning.com/book/api-design-patterns □) https://livebook.manning.com/book/designing-apis-with-swagger-and-openapi □) https://livebook.manning.com/book/irresistible-apis □) https://livebook.manning.com/book/understanding-api-security □) https://www.pdfdrive.com/designing-web-apis-building-apis-that-developers-love-e176234255.html □) https://www.pdfdrive.com/pro-restful-apis-design-build-and-integrate-with-rest-json-xml-and-jax-rs-e158249393.html 	5 (д) — 5 (дп)	[1],

	<ul style="list-style-type: none"> □) https://www.pdfdrive.com/building-telegram-bots-develop-bots-in-12-programming-languages-using-the-telegram-bot-api-e185770362.html □) https://www.pdfdrive.com/modern-api-design-with-aspnet-core-2-building-cross-platform-back-end-systems-e187622716.html □) https://www.pdfdrive.com/practical-api-architecture-and-development-with-azure-and-aws-e187648539.html □) https://livebook.manning.com/book/graphql-in-action/ □) https://livebook.manning.com/book/fullstack-graphql-applications-with-grandstack □) https://www.pdfdrive.com/learning-graphql-declarative-data-fetching-for-modern-web-apps-e158550106.html □) https://www.pdfdrive.com/graphql-язык-запросов-для-современных-веб-приложений-e188633229.html <p><i>Застосовуються авторські слайди оригінальних PowerPoint презентацій та мультимедійний проектор.</i></p>		
4	<p>□ Лекційна тема 4. JS-Клієнтська сторона. Прототипно-орієнтована скриптова мова JavaScript (JS). Основи та особливості написання скриптів у клієнтському JavaScript. JavaScript-фреймворк {бібліотека} AngularJS. JavaScript-фреймворк {бібліотека} Vue.js. JavaScript-фреймворк {бібліотека} React.</p> <p>Корисні Веб-посилання до Теми 4 JavaScript та AngularJS, Vue.js, React:</p> <ul style="list-style-type: none"> □) https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/JavaScript_basics □) https://developer.mozilla.org/uk/docs/Learn/JavaScript □) https://eloquentjavascript.net/ □) https://livebook.manning.com/book/web-performance-in-action/ □) https://livebook.manning.com/book/javascript-application-design □) https://livebook.manning.com/book/secrets-of-the-javascript-ninja-second-edition □) https://livebook.manning.com/book/spa-design-and-architecture/ □) https://www.w3schools.com/js/ □) https://schoolsw3.com/js/ □) https://javascript.info/ □) https://learn.javascript.ru/ □) http://jsforcats.com/ □) https://www.pdfdrive.com/learning-web-design-a-beginners- 	<p>5 (д)</p> <hr/> <p>5 (дп)</p>	[1],

[guide-to-html-css-javascript-and-web-graphics-e188549005.html](https://www.pdfdrive.com/guide-to-html-css-javascript-and-web-graphics-e188549005.html)

-) <https://developers.google.com/web/tools/chrome-devtools>
-) <https://livebook.manning.com/book/angularjs-in-action/>
-) <http://www.angularjsbook.com/angular-basics/chapters/>
-) <https://livebook.manning.com/book/getting-mean-with-mongo-express-angular-and-node>

<https://livebook.manning.com/book/getting-mean-with-mongo-express-angular-and-node>

-) <https://livebook.manning.com/book/getting-mean-with-mongo-express-angular-and-node-second-edition/>

<https://www.pdfdrive.com/angular-up-and-running-learning-angular-step-by-step-e176364392.html>

-) <https://github.com/shyamseshadri/angularjs-up-and-running>
-) <https://metanit.com/web/angular/>
-) <https://habr.com/ru/post/329452/>
-) <https://ru.vuejs.org/v2/guide/>
-) <https://www.pdfdrive.com/vuejs-up-and-running-building-accessible-and-performant-web-apps-e183876883.html>

<https://www.pdfdrive.com/vuejs-up-and-running-building-accessible-and-performant-web-apps-e183876883.html>

-) <https://livebook.manning.com/book/vue-js-in-action>
-) <https://www.pdfdrive.com/vuejs-в-действии-e187525955.html>
-) <https://reactjs.org/docs/getting-started.html>
-) <https://livebook.manning.com/book/react-in-action/>
-) <https://livebook.manning.com/book/react-quickly/>
-) <https://livebook.manning.com/book/testing-javascript-applications>

<https://livebook.manning.com/book/testing-javascript-applications>

-) <https://livebook.manning.com/book/d3-js-in-action>
-) <https://livebook.manning.com/book/d3js-in-action-second-edition>

<https://livebook.manning.com/book/d3js-in-action-second-edition>

-) <https://www.pdfdrive.com/angularjs-programming-for-beginners-learn-coding-fast-angular-js-language-crash-course-a-quick-start-guide-tutorial-book-with-hands-on-projects-in-easy-steps-an-ultimate-beginners-guide-e196540317.html>

<https://www.pdfdrive.com/getting-to-know-vuejs-learn-to-build-single-page-applications-in-vue-from-scratch-e187204417.html>

-) <https://www.pdfdrive.com/full-stack-angularjs-for-java-developers-build-a-full-featured-web-application-from-scratch-using-angularjs-with-spring-restful-e158323677.html>

<https://www.pdfdrive.com/learn-to-build-single-page-applications-in-vue-from-scratch-brett-nelson-e82444335.html>

-) <https://www.pdfdrive.com/angular-5-projects-learn-to-build-single-page-web-applications-using-70-projects-e187615562.html>

<https://www.pdfdrive.com/learning-react-a-hands-on-guide-to-building-web-applications-using-react-and-redux-e185752183.html>

-) <https://www.pdfdrive.com/react-16-essentials-a-fast-paced-hands-on-guide-to-designing-and-building-scalable-and-maintainable-web-apps-with-react-16-e158538535.html>

<https://www.pdfdrive.com/react-16-essentials-a-fast-paced-hands-on-guide-to-designing-and-building-scalable-and-maintainable-web-apps-with-react-16-e158538535.html>

	<ul style="list-style-type: none"> □) https://www.pdfdrive.com/reactjs-by-example-building-modern-web-applications-with-react-get-up-and-running-with-reactjs-by-developing-five-cutting-edge-and-responsive-projects-e166796743.html □) https://www.pdfdrive.com/learning-react-e48389168.html □) https://www.pdfdrive.com/react-и-redux-Функциональная-веб-разработка-e187536217.html <p><i>Застосовуються авторські слайди оригінальних PowerPoint презентацій та мультимедійний проектор.</i></p>		
5	<p>□ Лекційна тема 5. Пакетні менеджери {та системи багатомодульної розробки} {<i>PHP-based</i>} Composer та {<i>JS-based</i>} Node.js. <i>Git</i>-розподілена система контролю версій вихідного коду та керування версіями файлів (Git Version Control). JWT-авторизація за стандартом токена доступу на основі JSON (<i>JSON Web Token</i>). Системи контейнеризації Docker та Docker compose. Docker {<i>Compose</i>}-докеризація бази даних (database), бекенда (backend) та фронтенда (frontend) {Dockerizing Front- & Backend}. Методи автентифікації, куки (cookies) та сесії у React, Express та Docker.</p> <p>Корисні Веб-посилання до Теми 5 Composer, Node.js, Git, JWT, Docker:</p> <ul style="list-style-type: none"> □) https://getcomposer.org/doc/ □) https://nodejs.org/en/docs/guides/ □) https://habr.com/ru/post/439200/ □) https://habr.com/ru/post/458018/ □) https://itchief.ru/php/composer-for-beginners □) https://metanit.com/web/nodejs/ □) https://ruvds.com/img/other/ee86eb4f-db9f-48d3-8094-c76e14414678.pdf □) https://livebook.manning.com/book/node-js-in-action □) https://livebook.manning.com/book/node-js-in-action-second-edition □) https://livebook.manning.com/book/node-js-in-practice/ □) https://livebook.manning.com/book/get-programming-with-node-js □) https://livebook.manning.com/book/serverless-applications-with-node-js □) https://livebook.manning.com/book/express-in-action □) https://www.pdfdrive.com/typed-php-stronger-types-for-cleaner-code-e183831992.html □) https://www.pdfdrive.com/php-cookbook-solutions-examples- 	5 (д) — 5 (дп)	[1]

[for-php-programmers-e195095748.html](https://www.pdfdrive.com/full-stack-javascript-learn-backbonejs-nodejs-and-mongodb-e185917026.html)

□) <https://www.pdfdrive.com/full-stack-javascript-learn-backbonejs-nodejs-and-mongodb-e185917026.html>

□) <https://www.pdfdrive.com/beginning-nodejs-apress-e43544795.html>

□) <https://www.pdfdrive.com/practical-nodejs-building-real-world-scalable-web-apps-e183995134.html>

□) <https://habr.com/ru/post/340146/>

□) <https://habr.com/ru/company/ruvds/blog/457700/>

□) <https://codeguida.com/post/1567>

□) <https://ua-blog.com/туториал-по-node-version-manager-nvm/>

□) <https://git-scm.com/book/en/v2>

□) <https://git-scm.com/book/uk/v2>

□) <https://git-scm.com/book/ru/v2>

□) <https://www.pdfdrive.com/pro-git-d34114789.html>

□) <https://www.pdfdrive.com/git-for-teams-a-user-centered-approach-to-creating-efficient-workflows-in-git-d175594861.html>

□) <https://docs.docker.com/>

□) <https://docs.docker.com/compose/>

□) <https://livebook.manning.com/book/api-security-in-action/>

□) <https://livebook.manning.com/book/docker-in-action-second-edition>

□) <https://livebook.manning.com/book/docker-in-practice-second-edition>

□) <https://livebook.manning.com/book/learn-docker-in-a-month-of-lunches>

□) <https://www.pdfdrive.com/learning-docker-optimize-the-power-of-docker-to-run-your-applications-quickly-and-easily-e166596830.html>

□) <https://www.pdfdrive.com/docker-in-practice-e185808498.html>

□) <https://www.pdfdrive.com/docker-up-running-shipping-reliable-containers-in-production-d189833561.html>

□) <https://www.pdfdrive.com/using-docker-developing-and-deploying-software-with-containers-d185897923.html>

□) <https://www.pdfdrive.com/pro-docker-learn-how-to-use-containers-as-a-service-for-development-and-deployment-e179237255.html>

□) <https://www.pdfdrive.com/pro-docker-learn-how-to-use-containers-as-a-service-for-development-and-deployment-d196623666.html>

□) <https://habr.com/ru/post/316802/>

□) <https://habr.com/ru/post/519500/>

□) <https://habr.com/ru/post/478932/>

□) <https://habr.com/ru/post/479018/>

□) <https://wkrzywiec.medium.com/how-to-run-database-backend->

	<p>and-frontend-in-a-single-click-with-docker-compose-4bcda66f6de</p> <ul style="list-style-type: none"> □) https://milanwittpohl.com/projects/tutorials/full-stack-web-app/dockerizing-our-front-and-backend □) https://codenamecrud.ru/ruby-on-rails/sessions-cookies-and-authentication □) https://dev.to/macru/cookies-with-react-express-and-docker-4h97 <p><i>Застосовуються авторські слайди оригінальних PowerPoint презентацій та мультимедійний проектор.</i></p>		
6	<p>□ Лекційна тема 6. Веб-програмування хмарних обчислювальних середовищ. Безсерверні обчислення (Serverless Computing). Мова Веб-програмування Go.</p> <p>Корисні Веб-посилання до Теми 6 Serverless Computing, Go:</p> <ul style="list-style-type: none"> □) https://www.pdfdrive.com/beginning-serverless-computing-developing-with-amazon-web-services-microsoft-azure-and-google-cloud-d158318812.html □) https://www.serverless.com/blog/framework-example-golang-lambda-support □) https://livebook.manning.com/book/algorithms-of-the-intelligent-web-second-edition □) https://livebook.manning.com/book/serverless-architectures-on-aws □) https://livebook.manning.com/book/serverless-architectures-on-aws-second-edition □) https://livebook.manning.com/book/serverless-applications-with-node-js □) https://livebook.manning.com/book/amazon-web-services-in-action □) https://livebook.manning.com/book/cloud-native-patterns □) https://livebook.manning.com/book/the-cloud-at-your-service □) https://livebook.manning.com/book/google-cloud-platform-in-action □) https://livebook.manning.com/book/go-web-programming □) https://livebook.manning.com/book/go-in-action □) https://livebook.manning.com/book/go-in-practice □) https://livebook.manning.com/book/get-programming-with-go □) https://livebook.manning.com/book/kubernetes-in-action-second-edition □) https://www.pdfdrive.com/internet-infrastructure-networking- 	<p>5 (д)</p> <hr/> <p>5 (дп)</p>	[1]

<p>web-services-and-cloud-computing-d183943698.html</p> <ul style="list-style-type: none"> □) https://www.pdfdrive.com/java-в-облаке-spring-boot-spring-cloud-cloud-foundry-e176457037.html □) https://www.pdfdrive.com/kubernetes-management-design-patterns-with-docker-coreos-linux-and-other-platforms-e187301150.html □) https://www.pdfdrive.com/docker-management-design-patterns-swarm-mode-on-amazon-web-services-d187857888.html □) https://www.pdfdrive.com/pro-devops-with-google-cloud-platform-with-docker-jenkins-and-kubernetes-e189412469.html □) https://www.pdfdrive.com/google-cloud-platform-in-action-e186006829.html □) https://www.pdfdrive.com/building-your-next-big-thing-with-google-cloud-platform-e157851562.html <p><i>Застосовуються авторські слайди оригінальних PowerPoint презентацій та мультимедійний проектор.</i></p>		
<p>Усього лекційних годин з дисципліни «Web-програмування» для денної форми навчання</p>	<p>30 (Д)</p>	
<p>Усього лекційних годин з дисципліни «Web-програмування» для денної прискореної форми навчання</p>	<p>30 (ДП)</p>	

**Теми практичних занять з навчальної дисципліни «Web-програмування»
для денної та денної прискореної форм навчання**

№ Роб	№ Тем	Год. д/дп	Найменування <u>практичних занять</u> з дисципліни «Web-програмування» для <u>денної</u> та <u>денної прискореної форм навчання</u>	Література
1	1	4(д) — 4(дп)	<p align="center">□ Практична тема 1. Основи Web-дизайну</p> <p>□1.1) Ознайомитися зі структурою HTML-сторінки https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/HTML_basics та зверстати власну HTML-сторінку за зразками https://www.w3schools.com/html/html_intro.asp https://htmlacademy.ru/courses/26/run/13</p> <p>□1.2) Ознайомитися з основами CSS за посиланням https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/CSS_basics та змакетувати власний CSS-файл за зразками https://www.w3schools.com/html/html_css.asp https://htmlacademy.ru/courses/26/run/10</p> <p>□1.3) Ознайомитися з різними типовими варіантами існуючих PSD-макетів Інтернет-блогів за зразками https://freebiesbug.com/?s=blog та зверстати власну Веб-сторінку відповідно до індивідуально-обраного PSD-макету</p> <p>□1.4) Ознайомитися з JavaScript за посиланням https://developer.mozilla.org/uk/docs/Learn/Getting_started_with_the_web/JavaScript_basics та ознайомитися з інструментами Chrome DevTools за Веб-посиланнями https://developers.google.com/web/tools/chrome-devtools https://dou.ua/lenta/articles/chrome-dev-tools-guide/ https://htmlacademy.ru/blog/boost/tools/chrome-devtools-1</p> <p>Практичне завдання з DevTools: В консолі dev-tools chrome на зверстаній сторінці виконати завдання:</p> <ul style="list-style-type: none"> ▪) Пошук елементів за певними критеріями і виведення у консоль їх розмірів і положення за зразками: https://developer.mozilla.org/en-US/docs/Web/API/Element/getBoundingClientRect https://developer.mozilla.org/en-US/docs/Web/API/Window/getComputedStyle ▪) Видалення та додавання певних елементів: https://developer.mozilla.org/en-US/docs/Web/API/ChildNode/remove ▪) Ознайомитися з атрибутами “innerHTML” та “outerHTML” Element DOM інтерфейсу для WebAPI за 	[2], [3], [4], [6], [8], [9], [10], [13], [14], [18], [19], [21], [36].

			<p>https://developer.mozilla.org/en-US/docs/Web/API/Element/innerHTML</p> <p>https://developer.mozilla.org/en-US/docs/Web/API/Element/outerHTML</p> <ul style="list-style-type: none"> ▪) Ознайомитися з методом Node.appendChild() для зміни положення певних елементів за посиланням https://developer.mozilla.org/en-US/docs/Web/API/Node/appendChild □1.5) Ознайомитися з інтерфейсом “Event” подій https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Building_blocks/Events https://developer.mozilla.org/ru/docs/Web/API/EventTarget/addEventListener https://developer.mozilla.org/uk/docs/Web/Events <p>Практичне завдання з Event: За кліком миші на вибраних елементах реалізувати додавання/видалення або зміщення певних інших елементів відповідно до варіанту індивідуального завдання. Також можна реалізувати зміну розміру та редагування тексту елементів за бажанням.</p>	
2	3, 5.	4(д) — 4(дп)	<p>□ Практична тема 2. MockAPI та Node.js</p> <p>□2.1) Ознайомитися з сервісом MockAPI за лінками https://mockapi.io/ https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch https://archakov.im/post/mockapi-servis-dlya-sozdaniya-testovogo-restfulapi.html</p> <p>Практичне завдання з MockAPI: За кліком миші на певний елемент реалізувати відправлення get запиту до MockAPI і у випадку успішної відповіді додати елемент з текстом одержаної серверної відповіді, причому в разі помилки показати відповідне вікно з помилкою.</p> <p>□2.2) Ознайомитися з Node.js за посиланнями: https://nodejs.org/en/ https://nodejs.dev/learn/how-to-install-nodejs/ https://nodejs.dev/learn/run-nodejs-scripts-from-the-command-line https://nodejs.dev/learn/how-to-exit-from-a-nodejs-program https://nodejs.dev/learn/an-introduction-to-the-npm-package-manager https://nodejs.dev/learn/the-package-json-guide https://nodejs.dev/learn/build-an-http-server</p>	[1], [31]

			<p>https://nodejs.dev/learn/making-http-requests-with-nodejs</p> <p>Практичне завдання з Node.js: ▪) Реалізувати сервер, який буде оброблювати GET-запит від клієнта замість MockAPI. Відповідь має бути такою самою як і від MockAPI. ▪) Реалізувати надсилання вибраних статичних файлів, а саме зверстаного html файлу та відповідного css файлу на сервер (при запиті, скажімо, на localhost:3000 відправляти html та/або css файл(и)).</p>	
3	2, 5.	4(д) — 4(дп)	<p>□ Практична тема 3. Docker, mongoDB та JWT:</p> <p>□3.1) Ознайомитися з системою Docker Engine: https://docs.docker.com/engine/install/ https://docs.docker.com/engine/install/ubuntu/ https://docs.docker.com/docker-for-windows/install/ https://docs.docker.com/compose/install/ https://docs.docker.com/get-started/</p> <p>Завдання до Docker Engine: ▪) налаштувати докер для сервера з образом бази даних. ▪) приклад докерфайлу:</p> <pre>FROM node:13.12.0-alpine RUN npm install nodemon -g # Create app directory WORKDIR /app # Install app dependencies ADD package.json /app/package.json RUN npm install # If you are building your code for production # RUN npm ci --only=production # Bundle app source COPY . . EXPOSE 8080</pre> <p>▪) приклад docker-compose.yml файлу (у даному випадку використовується реляційна база даних PostgreSQL {postgres}):</p> <pre>version: "3" services: api: build: . environment: - DATABASE_URL=postgres://admin:admin@postgres:5432/blog - DEBUG_COLORS=true - DEBUG=Server command: npm run dev volumes: - ./app - /app/node_modules/ ports:</pre>	[2], [8], [10], [14]

			<p>- 5000:5000 depends_on: - postgres postgres: image: postgres:12.2-alpine ports: - "35432:5432" environment: POSTGRES_USER: admin POSTGRES_PASSWORD: admin POSTGRES_DB: blog</p> <p>□3.2) Ознайомитися з mongoDB за посиланнями: https://mongoosejs.com/ https://hub.docker.com/_/mongo Завдання до mongoDB:</p> <ul style="list-style-type: none"> ▪) Додати текстове поле та кнопку в html файлі (за відсутності таких елементів); ▪) При кліканні на додану кнопку надіслати POST запит на сервер з текстом із доданого текстового поля; ▪) На серверній стороні записувати надісланий текст у базу даних; ▪) Після завантаження сторінки відправити GET запит на сервер на одержання тексту для цього поля; ▪) Реалізувати обробку запиту на одержання тексту поля на стороні серверу (у тому числі одержання із БД). <p>□3.3) Ознайомитися з системою JWT: https://jwt.io/ https://codeguida.com/post/1567</p> <ul style="list-style-type: none"> ▪) Завдання до JWT: Реалізувати автентифікацію та авторизацію через JWT відповідно до інструкцій. 	
4	4, 5.	4(д) — 4(дп)	<p>□ Практична тема 4. Мови та фреймворки Typescript, Express.js, NestJS, ReactJS та React Redux:</p> <p>□4.1) Ознайомитися з мовою Typescript: https://www.typescriptlang.org/ https://eslint.org/docs/user-guide/getting-started Завдання для мови Typescript:</p> <ul style="list-style-type: none"> ▪) Переписати серверну частину на Typescript; ▪) Застосувати можливості інструменту ESLint для аналізу і контролю стилю та якості написання коду. <p>□4.2) Ознайомитися з Express.js та NestJS: https://expressjs.com/en/starter/installing.html https://expressjs.com/en/guide/routing.html https://docs.nestjs.com/ https://docs.nestjs.com/first-steps Завдання для фреймворків Express.js та NestJS:</p>	[1], [2], [6], [8], [9], [10], [13], [14], [18], [19], [21], [25], [26], [27].

			<ul style="list-style-type: none"> ▪) Переписати серверну частину із застосуванням фреймворків Express.js та/або NestJS. Ознайомитися з концепцією MVC. ▪) Реалізувати роутинг; ▪) Реалізувати моделі і контролери для користувачів, постів, коментарів, ролей і т.д. і т.п. <p>□4.3) Ознайомитися з ReactJS: https://reactjs.org/docs/getting-started.html https://habr.com/ru/post/461541/</p> <p>Завдання для фреймворків ReactJS та Typescript:</p> <ul style="list-style-type: none"> ▪) Переписати клієнтський застосунок із застосуванням фреймворків ReactJS та/або Typescript. <p>□4.4) Ознайомитися з React Redux: https://react-redux.js.org/ https://habr.com/ru/post/498860/ https://react-redux.js.org/introduction/quick-start#:~:text=React%20Redux%20is%20the%20official,the%20store%20to%20update%20data</p> <p>Завдання для фреймворку React Redux:</p> <ul style="list-style-type: none"> ▪) Додати у клієнтський застосунок React Redux 	
5	3, 4.	4(д) — 4(дп)	<p>□ Практична тема 5. Мова GraphQL та фреймворк V8 JavaScript engine:</p> <p>□5.1) Ознайомитися з мовою GraphQL: https://www.apollographql.com/docs/tutorial/introduction/?_ga=2.209885213.925791822.1610627493-114951234.1610627493</p> <p>Завдання для мови GraphQL: ▪) Додати Apollo GraphQL у клієнтський застосунок; ▪) Додати Apollo GraphQL Server у серверний застосунок.</p> <p>□5.2) На прикладі аналізу циклу подій (EventLoop) ознайомитися з роботою рушія V8 JavaScript engine як для Node.js, так і для браузера (є відмінності !) з посилань https://v8.dev/ https://nodejs.dev/learn/the-v8-javascript-engine https://nodejs.dev/learn/the-nodejs-event-loop https://nodejs.org/en/docs/guides/event-loop-timers-and-nexttick/ https://developer.mozilla.org/en-US/docs/Web/JavaScript/EventLoop https://developer.mozilla.org/uk/docs/Web/JavaScript/EventLoop https://medium.com/devschacht/javascript-eventloop-explained-f2dcf84e36ee</p>	[1], [2], [6], [8], [9], [10], [13], [14], [18], [19], [21], [25], [26], [27].

			<p>Завдання для V8 JavaScript engine:</p> <ul style="list-style-type: none"> ▪) Застосування можливостей V8 JavaScript engine до роботи над оптимізацією клієнтського застосунку шляхом вимірювання продуктивності за допомогою дебаггера та пошук проблемних частин коду; ▪) Вивчення способів оптимізації застосунку з V8 JavaScript engine та застосування їх на практиці. 	
6	2	5(д) — 5(дп)	<p>□ Практична тема 6. Мова PHP:</p> <p>□6.1) Завантаження та встановлення опенсервера (<i>openserver</i>) з https://ospanel.io/ ;</p> <p>□6.2) Встановлення мови PHP з https://www.php.net/ та ознайомлення з її синтаксичними особливостями;</p> <p>□6.3) Встановлення та ознайомлення з менеджером потоків Composer з https://getcomposer.org/ ;</p> <p>□6.4) Ознайомлення студентів з PSR-стандартами для межі дії (області видимості) та кореневої директорії: https://www.php-fig.org/psr/ https://habr.com/ru/post/458484/ https://www.php.net/manual/ru/language.variables.scope.php</p> <p>□6.5) Ознайомлення з об'єктною моделлю в мові PHP з https://www.php.net/manual/ru/oop5.intro.php для питань області видимості, наслідування та пізнього статичного зв'язування відповідно до посилання https://www.php.net/manual/ru/language.oop5.late-static-bindings.php</p> <p>□6.6) Ознайомлення з PHP-інтерфейсами об'єктів https://www.php.net/manual/ru/language.oop5.interfaces.php , з синтаксисом для абстрактних класів та методів https://www.php.net/manual/ru/language.oop5.abstract.php , з механізмами перевикористання спільного коду в класах https://www.php.net/manual/ru/language.oop5.traits.php , з динамічними public-«сутностями» первантажень методів https://www.php.net/manual/ru/language.oop5.overloading.php та з патернами https://refactoring.guru/uk/design-patterns</p> <p>Завдання для 6.6 з Singleton-патерн:</p> <ul style="list-style-type: none"> ▪) PHP-реалізувати Singleton-патерн «одинака» за https://refactoring.guru/uk/design-patterns/singleton/php/example https://refactoring.guru/design-patterns/singleton/php/example https://designpatternsphp.readthedocs.io/ru/latest/Creati onal/Singleton/README.html 	[9], [10], [12], [16], [17], [22], [23], [24].

			<ul style="list-style-type: none"> ▪) Реалізувати Singleton- та/або PDO-доступ до БД https://stackoverflow.com/questions/37733391/singleton-alternative-for-php-pdo https://phpdelusions.net/pdo/pdo_wrapper https://ruhighload.com/singleton+в+php+на+примере+подключення+к+mysql http://phpfaq.ru/pdo/pdo_wrapper 	
7	2	5(д) — 5(дп)	<p>□ Практична тема 7. PHP-фреймворк Laravel:</p> <p>□7.1) Встановлення PHP-фреймворка Laravel відповідно до https://laravel.com/docs/8.x/installation ;</p> <p>□7.2) Ознайомлення студентів з ▪) маршрутизацією https://www.w3schools.in/laravel-tutorial/routing/ https://laravel.com/docs/8.x/routing https://laravel.su/docs/5.4/routing https://laravel.ru/docs/v5/routing https://habr.com/ru/post/474788/ ознайомлення з ▪) класом контролерів (<i>controllers</i>) https://www.w3schools.in/laravel-tutorial/controllers/ https://laravel.com/docs/8.x/controllers https://laravel.su/docs/6.x/controllers https://laravel.ru/docs/v5/controllers ознайомлення з ▪) Blade-шаблонами https://laravel.com/docs/8.x/blade https://laravel.su/docs/5.4/blade https://laravel.ru/docs/v5/blade</p> <p>□7.3) Ознайомлення студентів з ▪) моделями (<i>Eloquent model</i>) https://laravel.com/docs/8.x/eloquent ; з ▪) основами роботи та зв'язками з базами даних відповідно до https://laravel.com/docs/8.x/database ; з ▪) конструктором запитів до бази даних відповідно до https://laravel.com/docs/8.x/queries ; з ▪) заповненням бази даних початковими даними https://laravel.com/docs/8.x/seeding https://laravel.ru/docs/v5/seeding https://laravel.su/docs/5.2/seeding https://unetway.com/tutorial/laravel-seeding та з ▪) міграцією між базами даних https://laravel.com/docs/8.x/migrations https://laravel.ru/docs/v5/migrations https://laravel.su/docs/5.2/migrations https://www.digitalocean.com/community/tutorials/how-to-use-database-migrations-and-seeders-to-abstract-database-setup-in-laravel-ru</p> <p>□7.4) Ознайомлення студентів зі ▪) зв'язками між</p>	[11], [15], [28], [29].

		<p>таблицями за допомогою моделей https://laravel.com/docs/8.x/eloquent-relationships https://laravel.ru/docs/v5/eloquent-relationships</p> <p>□7.5) Ознайомлення студентів із постачальниками послуг (або сервіс провайдерами) в Laravel: https://laravel.com/docs/8.x/providers https://laravel.com/docs/8.x/container https://saribzhanov.ru/tehnо/servis-provayderyi-v-laravel-dlya-samyih-malenkih/ https://code.tutsplus.com/ru/tutorials/how-to-register-use-laravel-service-providers--cms-28966 https://laravel.ru/docs/v5/providers https://laravel.su/docs/5.4/providers</p> <p>□7.6) Ознайомлення студентів з роботою із файлами, а також з авторизацією та принципами роботи сесій https://laravel.com/docs/8.x/filesystem#introduction https://laravel.com/docs/8.x/requests#introduction https://laravel.com/docs/8.x/authentication#introduction</p> <p>□7.7) Ознайомлення студентів з посередниками (<i>middleware</i>), доступом до сторінок, ролі користувачів та роботою з адмініструванням: https://laravel.com/docs/8.x/middleware#introduction https://www.tutorialspoint.com/laravel/laravel_middle_ware.htm https://laravel.ru/docs/v5/middleware https://code.tutsplus.com/ru/tutorials/understand-the-basics-of-laravel-middleware--cms-29147 https://habr.com/ru/post/429214/</p>	
Усього годин	30 (д)	Для тем практичних занять з дисципліни « Web-програмування » для денної форми навчання	
Усього годин	30 (дп)	Для тем практичних занять з дисципліни « Web-програмування » для <u>денної прискореної форми навчання</u>	

Теми індивідуальних занять з навчальної дисципліни «Web-програмування» для денної та денної прискореної форм навчання

Індивідуальна робота з дисципліни «Web-програмування» містить наступні етапи:

- проробка лекційного матеріалу в рамках широкого використання вітчизняної та міжнародної англomовної навчальної літератури;
- підготовка до теоретичного опитування та контрольних робіт;

– самостійне вивчення частини теоретичного матеріалу згідно з рекомендованою літературою та складення додаткових конспектів;

– самостійне опанування комп'ютерно-обчислювальними можливостями такого вільного та відкритого сучасного програмного забезпечення як редактори, середовища та фреймворки для роботи з **HTML-, CSS-, PHP-** та **JS-**файлами;

– виконання *розрахунково-графічних завдань* індивідуального характеру **РГР 1 [Розробити комп'ютерні коди та письмові звіти за шістьма лекційними темами]; РГР 2 [Розробити комп'ютерні коди та письмові звіти за сімома темами практичних занять]; РГР 3 [Розробити CRUD-застосунок для роботи з базою даних певної організації (фірми, заводу, бібліотеки, школи, університету, лікарні) із застосуванням можливостей PHP-фреймворку Laravel і JS-фреймворку Vue].**

Тематика та перелік індивідуальних розрахунково-графічних завдань з навчальної дисципліни «Web-програмування» для денної та денної прискореної форм навчання

№ з/п	Назва теми, з якої виконується індивідуальне завдання	Назва і вид індивідуального завдання
1	Лекційні теми 1 – 6.	Розрахунково-графічна робота №1 «Розробити комп'ютерні коди та письмові звіти за шістьма лекційними темами»
2	Практичні теми 1 – 7.	Розрахунково-графічна робота №2 «Розробити комп'ютерні коди та письмові звіти за сімома темами практичних занять»
3	Із застосуванням можливостей PHP-фреймворку Laravel і JS-фреймворку Vue розробити CRUD-застосунок (Create-Read-Update-Delete) для роботи з базою даних (БД) певної організації (фірми, заводу, бібліотеки, школи, університету, лікарні), розробити міграції (<i>migrations</i>), сідери та фабрики для автоматичного наповнення БД, розробити адміністрування, визначити ролі користувачів, створити посередники (наприклад перевірки ролі користувача), забезпечити роботу над даними (скажімо через	Розрахунково-графічна робота №3 «Розробити CRUD-застосунок для роботи з базою даних певної організації (фірми, заводу, бібліотеки, школи, університету, лікарні) із застосуванням можливостей PHP-фреймворку Laravel і JS-фреймворку Vue »

ресурсний контролер), підготувати сторінки для виконання цих CRUD -операцій над моделями БД, а також валідацію і виведення помилок.	
--	--

Контрольні роботи з навчальної дисципліни «Web-програмування»

Контрольні роботи з теоретичної частини курсу «Web-програмування» є розподіленими наступним чином:

№ роботи	№ лекц. теми	Тема контрольної роботи	Кількість варіантів
1	1	Розробка html та css файлів власного проекту	20
2	4	Розробка js файлу власного проекту	20
3	2	Розробка php файлу власного проекту	20

5. Контрольні заходи та критерії оцінювання знань студентів ДДМА в рамках навчального процесу упродовж вивчення залікової навчальної дисципліни «Web-програмування»

Підсумкові оцінки за триместр з залікової дисципліни «Web-програмування» в цілому переводяться за національною шкалою та шкалою ECTS відповідно до нижченаведеної міжнародної таблиці переведення оцінок, яка визначається діючим в Донбаській державній машинобудівній академії (ДДМА) положенням про організацію навчального процесу в кредитно-модульній системі підготовки фахівців:

Рейтингова оцінка	У національній шкалі	У шкалі ECTS
90-100	Відмінно (зараховано)	A
81-89	Добре (зараховано)	B
75-80	Добре(зараховано)	C
65-74	Задовільно (зараховано)	D
55-64	Задовільно (зараховано)	E
30-54	Незадовільно (не зараховано)	FX
0-29	Незадовільно (не зараховано)	F

В контрольні заходи з дисципліни «Web-програмування» для студентів денної та **денної прискореної** форми навчання входять:

- письмові та усні контрольні опитування з дисципліни «Web-програмування» в семестрі;
- виконання та захист *трьох* індивідуальних *розрахунково-графічних завдань РГР 1 – РГР 3* з «Web-програмування» в семестрі;
- виконання та захист додаткового індивідуального розрахунково-обчислювального завдання із Веб-застосуваннями мови **Go**;

- захист комп'ютерних кодів та звіту з шести тем лекційних занять з «Web-програмування» як РГР 1;
- захист комп'ютерних кодів та звіту з семи тем практичних занять з «Web-програмування» як РГР 2;
- залік з «Web-програмування».

Поточний контроль знань та умінь студентів денної форми навчання з «Web-програмування» проводиться шляхом оцінювання практичних робіт та проведення письмових контрольних робіт із застосуванням індивідуальних тестових завдань {у тому числі із використанням системи <http://moodle.dgma.donetsk.ua> }.

Засоби оцінювання знань студентів денної та денної прискореної форми навчання з курсу «Web-програмування»

№	Назва і короткий зміст контрольного заходу	Характеристика змісту засобів оцінювання знань студентів з курсу «Web-програмування»
1	▪ Письмові та усні контрольні опитування	▪ опитування за попередньо вивченим та поточним термінологічним матеріалом
2	▪ Захист звітів з трьох індивідуальних розрахунково-графічних завдань (РГР 1 – РГР 3)	<ul style="list-style-type: none"> ▪ коротке опитування за термінологічним, концептуальним, комп'ютерно-синтаксичним, семантичним комп'ютерно-алгоритмічним матеріалом, що відповідає темі індивідуально-виконаної розрахунково-графічної роботи; ▪ оцінювання коректності та якості програмно-реалізованих алгоритмів в рамках виконання індивідуальних розрахунково-графічних робіт ▪ оцінювання аргументованості звітів щодо розбору та розуміння ситуаційних завдань; ▪ оцінювання активності участі у дискусіях ▪ коротке опитування за термінологічним, концептуальним, комп'ютерно-синтаксичним, семантичним комп'ютерно-алгоритмічним матеріалом, що відповідає темі індивідуально-виконаної обчислювальної роботи за розділами лекційного та практичного матеріалу; ▪ оцінювання коректності та якості програмно-реалізованих алгоритмів в рамках виконання практичних та лекційних робіт; ▪ оцінювання аргументованості звіту щодо розбору та розуміння ситуаційних завдань; ▪ оцінювання активності участі у дискусіях.

3	<ul style="list-style-type: none"> ▪ Захист індивідуального додаткового завдання 	<ul style="list-style-type: none"> ▪ письмовий звіт та супровідні електронні файли усіх кодів індивідуально-розроблених комп'ютерних програм, підготовлених в рамках виконання індивідуального додаткового завдання; ▪ оцінювання самостійності та якості виконання завдання та розробки комп'ютерних програм упродовж звіту-захисту та співбесіди
4	<ul style="list-style-type: none"> ▪ Поточні та підсумкові контрольні роботи 	<ul style="list-style-type: none"> ▪ стандартизовані тести; ▪ аналітично-розрахункові завдання;
	<ul style="list-style-type: none"> ▪ Підсумковий контроль 	<ul style="list-style-type: none"> ▪ стандартизовані тести; ▪ аналітично-розрахункові завдання;

Оцінювання з «**Web-програмування**» виконується за стобальною системою з подальшим урахуванням вагового коефіцієнту в межах кожного із трьох модулів та з урахуванням вагового коефіцієнту для кожного із трьох модулів задля підсумкової атестації з «**Web-програмування**».

Захист звіту з практичних з «**Web-програмування**» проводиться в межах кожного модулю за розкладом. Кінцевий звіт приймається на останньому занятті при наявності письмового (або електронного) тексту звітів з «**Web-програмування**» зі стандартним титульним аркушем, обов'язковою наявністю усіх електронних файлів програмних кодів для всіх робіт та спроможності студента відповісти на всі поставлені викладачем запитання щодо вивченого матеріалу з навчальної дисципліни «**Web-програмування**».

Залік з навчальної дисципліни «**Web-програмування**» студент складає після захисту усіх звітів та модулів. Склад матеріалу та об'єму заліку залежить від модулів, які студент бажає перездати для підвищення рейтингу, або у разі потреби у додатковій перевірці теоретичних знань. Для отримання позитивної оцінки з «**Web-програмування**» студент повинен скласти всі обов'язкові контрольні точки курсу та одержати не менше ніж 55 балів сумарної оцінки як на протязі семестру, так і безпосередньо на заліку. Студент, який на протязі семестру склав всі контрольні точки курсу «**Web-програмування**» і набрав не менше 55 балів сумарної оцінки, має право отримати підсумкову «автоматичну» оцінку без необхідності складання заліку.

Результати прийому заліку з дисципліни «**Web-програмування**» оцінюються за 100 – бальною рейтинговою шкалою. При оцінюванні результатів заліку може також використовуватися національна 5–бальна шкала та вищенаведена таблиця переведення балів в рамках діючого в ДДМА положення про організацію навчального процесу в кредитно-модульній системі підготовки фахівців.

**Перелік обов'язкових контрольних точок для оцінювання знань
студентів денної та денної прискореної форми навчання
з курсу «Web-програмування»**

№	Назва і короткий зміст контрольного заходу з дисципліни « Web-програмування »	Max балів	Характеристика критеріїв досягнення результатів навчання для отримання максимальної кількості балів
1	Виконання та захист <i>трьох</i> індивідуальних <i>розрахунково-графічних завдань РГР 1 – РГР 3</i> з « Web-програмування »	60	Студент практично здатний продемонструвати критичне осмислення лекційного та позалекційного матеріалу, брати кваліфіковану участь у послідовній професійній дискусії з наведенням аргументації. Студент успішно виконав розрахунково-графічні завдання, обчислювальні практичні завдання з « Web-програмування », а також навів ґрунтовні і аргументовані відповіді на уточнювальні та додаткові запитання викладача та колег.
2	Контрольна робота №1 до теми №1 «Розробка html та css файлів власного проекту»	10	Студент виконав письмові, тестові, обчислювальні практичні та індивідуальні розрахунково-обчислювальні завдання, що відповідають програмним результатам навчання за відповідними темами модулю №1
3	Контрольна робота №2 до теми №4 «Розробка js файлів власного проекту»	10	Студент виконав письмові, тестові, обчислювальні практичні та індивідуальні розрахунково-обчислювальні завдання, що відповідають програмним результатам навчання за відповідними темами модулю №2
4	Контрольна робота №3 до теми №2 «Розробка php файлів власного проекту»	10	Студент виконав письмові, тестові, обчислювальні практичні та індивідуальні розрахунково-обчислювальні завдання, що відповідають програмним результатам навчання за відповідними темами модулю №3
5	Виконання додаткового індивідуального розрахунково-обчислювального завдання щодо « Web-	10	Студент здатний засвоїти та практично застосувати необхідні сучасні методики « Web-програмування » шляхом розробки та програмної реалізації власних розроблених алгоритмів в рамках

	програмування»		прикладного використання наступних методів та концепцій мови Go для Веб-програмування.
Поточний контроль з дисципліни «Web-програмування»	3	100 (*1/2)	Студент сформував загальне професійне враження від вивченого курсу «Web-програмування»
Підсумковий контроль		100 (*1/2)	Студент виконав основні та додаткові письмові, тестові, обчислювальні практичні, індивідуальні розрахунково-обчислювальні, рутинні технічні, творчі технічні та соціально-інженерні завдання, що відповідають досягненню програмних результатів успішного інженерного навчання з курсу «Web-програмування»
Всього		100	-

Перелік обов'язкових контрольних точок для оцінювання знань студентів заочної форми навчання з курсу «Web-програмування»

№	Назва і короткий зміст контрольного заходу з дисципліни «Web-програмування»	Max балів	Характеристика критеріїв досягнення результатів навчання для отримання максимальної кількості балів
1	Тестова контрольна робота з дисципліни «Web-програмування», яка індивідуально виконується студентом в системі Moodle	40	Студент виконав тестові завдання, що відповідають програмним результатам навчання з дисципліни «Web-програмування»
2	Письмовий екзамен (іспит) з дисципліни «Web-програмування»	60	Студент виконав основні та додаткові письмові, тестові, обчислювальні практичні, індивідуальні розрахунково-обчислювальні, рутинні технічні, творчі технічні та соціально-інженерні завдання, що відповідають досягненню програмних результатів успішного інженерного навчання з курсу «Web-програмування»
Всього		100	-

6. Навчально-методичні матеріали та рекомендовані джерела інформації

6.1. Основна література до курсу «Web-програмування»

- 1) Casciaro, M., & Mammino, L. (2020). *Node.js Design Patterns: Design and implement production-grade Node.js applications using proven patterns and techniques*. (3rd Ed.). Birmingham, UK: Packt Publishing Ltd. Retrieved from <http://library.lol/main/F94E37AA0C43246FD1F08DEF9FFD802A> and from <http://library.lol/main/24578591239D51A2748D8FF4677E4146> and from <https://github.com/PacktPublishing/Node.js-Design-Patterns-Third-Edition/> { Каскиаро, М., & Маммино, Л. (2017). *Шаблоны проектирования Node.js: Воспользуйтесь самыми мощными компонентами и шаблонами платформы Node.js для создания масштабируемых модульных приложений*. М.: ДМК Пресс. Retrieved from <http://library.lol/main/59B8904AFC323A68863216D062F9B52F>. Доступ до повного тексту підручника є за адресою https://download.studytube-staging.nl/documents/22/085559dc8e932a55cc3be2de9eff5f619b44fdf1d4c2887d2f51b89fe4436c9e/- - - - - node-js - 2017.PDF?academy_id=1 }
- 2) Connolly, R., & Hoar, R. (2018). *Fundamentals of Web Development*. (2nd ed.). NY: Pearson Education. Retrieved from <http://library.lol/main/7B8FD13E6DFD93D475D0AE5792B51FF5>
- 3) Dean, J. (2019). *Web Programming with HTML5, CSS, and JavaScript*. Burlington, MA: Jones & Bartlett Learning. Retrieved from <http://library.lol/main/E271D73F02CCC45CB3972199CE531A9A> and from [http://193.236.39.101/recursos/ledes/outros_livros/%5BJohn Dean%5D Web Programming with HTML5, CSS, and JavaScript.pdf](http://193.236.39.101/recursos/ledes/outros_livros/%5BJohn%20Dean%5D%20Web%20Programming%20with%20HTML5,%20CSS,%20and%20JavaScript.pdf) and from <https://www.pdfdrive.com/web-programming-with-html5-css-and-javascript-e187657772.html>
- 4) Felke-Morris, T. (2019). *Web Development and Design Foundations with HTML5*. (9th Ed.). NY: Pearson Higher Education. Retrieved from <http://library.lol/main/ABBE6BB1F72365359788F83A643876E7>
- 5) Griffiths, I. (2019). *Programming C# 8.0: Build Cloud, Web, and Desktop Applications*. (1st ed.). Sebastopol, CA: O'Reilly Media. Retrieved from <http://library.lol/main/CA394FB3D465CD39F5CA84E76BB83115> and from <https://github.com/idg10/prog-cs-8-examples>
- 6) Haverbeke, M. (2019). *Eloquent JavaScript: a modern introduction to programming*. (3rd ed.). San Francisco, CA: No Starch Press. Retrieved from <http://library.lol/main/39D4CA25C917300F0968E256386743F5> and from <https://eloquentjavascript.net/> { Хавербеке, М. (2019). *Выразительный JavaScript. Современное веб-программирование: Серия «Для профессионалов»*. (3-е изд.). СПб.: Питер. Retrieved from <https://eloquent-javascript.karmazzin.ru/> and from <http://library.lol/main/868B33F6CC9342A7D0D527C9592648C5> and from https://gotoadm.ru/files/eloquentjavascript_ru.pdf }

7) McGavren, J. (2019). *Head First Go*. Sebastopol, CA: O'Reilly Media. Retrieved from <http://library.lol/main/1FFBF91A3662DFC40A1CEECC5F2C78A7> { Макгаврен, Дж. (2020). *Head First. Изучаем Go*. СПб.: Питер. Retrieved from <http://library.lol/main/94D6151CE51297924CE6DA7C89665665> }

8) Mendez, M. (2014). *The Missing Link: An Introduction to Web Development and Programming*. Geneseo, NY: Open SUNY Textbooks. Retrieved from <https://knightscholar.geneseo.edu/cgi/viewcontent.cgi?article=1016&context=oer-ost> and from <https://milneopentextbooks.org/the-missing-link-an-introduction-to-web-development-and-programming/> and from <http://themissinglink.site/> and from <https://milnepublishing.geneseo.edu/themissinglink/>

9) Nixon, R. (2018). *Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5*. (5th ed.). Sebastopol, CA: O'Reilly Media. Retrieved from <http://library.lol/main/7D5B67E3287C146DC546D4248DCCCBDF> { Никсон, Р. (2016). *Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5*. (4-е изд.). СПб.: Питер. Retrieved from <https://booster.by/files/oeu.pdf> }

10) O'Kane, M. (2017). *A Web-Based Introduction to Programming: Essential Algorithms, Syntax, and Control Structures Using PHP, HTML, and MariaDB/MySQL*. (4th ed.). Durham, North Carolina: Carolina Academic Press. Retrieved from <http://library.lol/main/FFCEE03531E5BBE5464A33252E84315> and from <https://www.mikeokane.com/textbooks/WebTech/support.php>

11) Potencier, F. (2020). *Symfony 5: The Fast Track*. (v1.0.6 - Generated on February 10, 2020). Clichy, France: Symfony SAS. Retrieved from <http://library.lol/main/2C748BC089F525776BA82734028A43CE> and from <https://symfony.com/doc/current/the-fast-track/en/index.html> { Potencier, F. (2020). *Symfony 5. Быстрый старт*. Clichy, France: Symfony SAS. Retrieved from <http://library.lol/main/0216A785861E532FB6931086F3DD3BE8> and from <https://symfony.com/doc/current/the-fast-track/ru/index.html> }

12) Reiersol, D., Baker, M., & Shiflett, C. (2007). *PHP in Action: Objects, Design, Agility*. Shelter Island, NY: Manning Publications. Retrieved from <http://library.lol/main/3CF86E4CD3941E19E2CD0CAA3C8F680> and from <https://livebook.manning.com/book/php-in-action> and from <https://oiipdf.com/php-in-action> and from <https://www.manning.com/downloads/960>

13) Robbins, J. N. (2018). *Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics*. (5th ed.). Sebastopol, CA: O'Reilly Media. Retrieved from <http://library.lol/main/2CAD8EB632E7AAE4D523D17731FA58D2> and from <http://wtf.tw/ref/robbins.pdf> { Роббинс, Дж. (2014). *HTML5, CSS3 и JavaScript. Исчерпывающее руководство*. (4-е изд.). М.: Эксмо. Retrieved from <http://library.lol/main/6C61BA1E6252F9F9A733753E8125A8F6> and from <http://library.lol/main/363662E9C0A03FF0BD31A74D4BBFC1BC> }

14) Scobey, P., & Lingras, P. (2018). *Web Programming and Internet Technologies: An E-Commerce Approach*. (2nd Ed.). Burlington, MA: Jones & Bartlett Learning. <http://library.lol/main/74677BF043DC2707017C0936BD0633F1>

15) Stauffer, M. (2019). *Laravel: Up & Running: A Framework for Building Modern PHP Apps*. (2nd ed.). Sebastopol, CA: O'Reilly Media. Retrieved from <http://library.lol/main/8A1D1CB90D1797DDD6DCE06F1D29EBB4> and from <https://www.pdfdrive.com/laravel-up-running-a-framework-for-building-modern-php-apps-e189833029.html> { Стаффер, М. (2020). *Laravel. Полное руководство: Бестселлеры O'Reilly*. (2-е изд.). СПб.: Питер, 2020. Retrieved from <http://library.lol/main/52CBC9F953172A047034C8D9B2958D40> }

16) Tatroe, K., & MacIntyre, P. (2020). *Programming PHP: Creating Dynamic Web Pages*. (4th ed.). Sebastopol, CA: O'Reilly Media. Retrieved from <http://library.lol/main/4EEE20597C81294D21B10B338994DB31>

17) Ullman, L. (2018). *PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide*. (5th ed.). San Francisco, CA: Peachpit Press, Pearson Education. Retrieved from <http://library.lol/main/7B9E94EE742556CC4DB70FEDC703B2AB> and from <https://github.com/LarryUllman/phpmysqlvqp-5ed>

18) Wagner, G., & Diaconescu, M. (2017). *Web Applications with JavaScript or Java. Volume 1: Constraint Validation, Enumerations, Special Datatypes*. Berlin, Boston: De Gruyter Oldenbourg. DOI: <https://doi.org/10.1515/9783110499957> . Retrieved from <http://library.lol/main/6237B6E14F4CD33CD40CDD2CB4D5F6F3> and from <https://web-engineering.info/book/WebApp1/> and from <https://web-engineering.info/WebAppBook> and from

19) Wagner, G., & Diaconescu, M. (2021). *Web Applications with JavaScript or Java. Volume 2: Associations and Class Hierarchies*. Berlin, Boston: De Gruyter Oldenbourg. Retrieved from <https://web-engineering.info/book/WebApp2/> and from <https://web-engineering.info/WebAppBook>

6.2. Додаткова література до курсу «Web-програмування»

20) Sinha, S. (2020). *Quick Start Guide to Dart Programming: Create High-Performance Applications for the Web and Mobile*. Berkeley, CA: Apress. Retrieved from <http://library.lol/main/C872326F7C222A817A51580F7C273521> and from <https://doi.org/10.1007/978-1-4842-5562-9>

21) Мельник, А. О., Стіренко, С. Г., Поліновський, В. В., Ситніков, В. С., Купін, А. І., Погорілий, С. Д., Харченко В. С. (2018). *Стандарт вищої освіти України: перший (бакалаврський) рівень, галузь знань 12 – Інформаційні технології, спеціальність 123 – Комп'ютерна інженерія*. Київ: Міністерство освіти і науки України. Retrieved from

<https://mon.gov.ua/storage/app/media/vyshcha/standarty/123.pdf> ;
<https://mon.gov.ua/storage/app/media/vishcha-osvita/zatverdzeni%20standarty/12/21/123-kompyuterna-inzheneriya-bakalavr.pdf>

6.3. Корисні Веб-ресурси з інсталяторами до курсу «Web-програмування»

22) Open Server Panel. Retrieved from <https://ospanel.io/>

23) Ampps stack of Apache, PHP, MySQL, Perl & Python. Retrieved from <http://ampps.com/>

24) Bitnami for XAMPP - PHP development environment: Retrieved from <https://www.apachefriends.org/index.html>

25) AngularJS. Retrieved from <https://angular.io/>

26) Vue.js - The Progressive JavaScript Framework. Retrieved from <https://vuejs.org/>

27) React. Retrieved from <https://uk.reactjs.org/>

28) Laravel - The PHP Framework For Web Artisans. Retrieved from <https://github.com/laravel/> and from <https://laravel.com/> and from <https://laravel-news.com/> and from <https://laracasts.com/> and from <https://laravel.demiart.ru/> and from <https://laravel.ru/> and from <https://laravel.su/>

29) Symfony - High Performance PHP Framework for Web. Retrieved from <https://symfony.com/>

30) Composer - A Dependency Manager for PHP. Retrieved from <https://getcomposer.org/>

31) Node.js – JavaScript-оточення, побудоване на JavaScript-рушієві Chrome V8: <https://nodejs.org/en/>

32) ASP.NET: Retrieved from <https://dotnet.microsoft.com/apps/aspnet>

33) Go - an open source programming language. Retrieved from <https://golang.org/>

6.4. Корисні хмарні Веб-редактори до курсу «Web-програмування»

34) DartPad (online editor): Retrieved from <https://dartpad.dev/?>

35) The Go Playground (online editor): Retrieved from: <https://play.golang.org/>

6.5. Корисні Веб-ресурси з відеолекціями до курсу «Web-програмування»

36) Web Demystified {by Mozilla Hacks}. Retrieved from <https://www.youtube.com/playlist?list=PLo3w8EB99pqLEopnuz-dOOBJ8t-Wgt2g>