

Міністерство освіти і науки України  
Національний аерокосмічний університет ім. М. С. Жуковського  
«Харківський авіаційний інститут»

Кафедра інженерії програмного забезпечення (№ 603)

**ЗАТВЕРДЖУЮ**

Керівник проектної групи/

 І.Б. Туркін  
(підпис) (ініціали та прізвище)

«30» 08 2019 р.

РОБОЧА ПРОГРАМА ВИБІРКОВОЇ  
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Скриптові мови для Web»  
(назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»  
(шифр і найменування галузі знань)

Спеціальність: 121 «Інженерія програмного забезпечення»  
(код та найменування спеціальності)

Освітня програма: «Інженерія програмного забезпечення»  
(найменування освітньої програми)

**Форма навчання: денна**

**Рівень вищої освіти: : перший (бакалаврський)**

**Харків 2019 рік**

Робоча програма «Скриптові мови для Web» для студентів за спеціальністю: 121 «Інженерія програмного забезпечення» освітньою програмою «Інженерія програмного забезпечення»

«20» 04 2019 р. – 10 с.

Розробник: Сергієнко В.В., к.т.н., доцент каф. 603

  
(підпис)

Робочу програму розглянуто на засіданні кафедри інженерії програмного забезпечення

—  
Протокол № 4 від «20» 04 2019 р. (назва кафедри)

Завідувач кафедри д-р техн. наук., проф. (науковий ступінь і вчене звання)

  
(підпис)

І.Б. Туркін  
(ініціали та прізвище)

### 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни
		денна форма навчання
Кількість кредитів – 4.5	<b>Галузь знань</b> 12 «Інформаційні технології»	Цикл загальної (професійної) підготовки (за вибором)
Кількість модулів – 2	<b>Спеціальність:</b> 121 «Інженерія програмного забезпечення»  <b>Освітня програма:</b> <u>Інженерія програмного забезпечення</u>	<b>Навчальний рік:</b> 2019/2020
Кількість змістових модулів – 2		<b>Семестр</b>
Індивідуальне завдання: немає		4-й
Загальна кількість годин : 48 /135		<b>Лекції</b> <sup>1)</sup>
Кількість тижневих годин для денної форми навчання: аудиторних –4 самостійної роботи студента – 8	<b>Рівень вищої освіти:</b> перший (бакалаврський)	24 год
		<b>Лабораторні</b> <sup>1)</sup>
		24 год
		<b>Самостійна робота</b>
		87 год
		<b>Вид контролю:</b> модульний контроль, залік

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: 48/87.

<sup>\*)</sup> Аудиторне навантаження може бути зменшене або збільшене на одну годину в залежності від розкладу занять.

## 2. Мета та завдання навчальної дисципліни

**Мета вивчення** - надання студентам теоретичних знань і практичних навичок використання мови програмування JavaScript, розробки сучасних веб-додатків за допомогою JavaScript, HTML, CSS, XML, JSON.

**Завдання** - вивчення студентами концепцій побудови динамічних веб-додатків та отримання базових знань для самостійної розробки веб-додатків на абстрактному, логічному та фізичному рівні.

**Результати навчання:** студент має:

**знати**

- основні засоби розробки веб-додатків;
- базові мовні конструкції мов веб-програмування (HTML, CSS, Javascript);
- базові структури даних, абстрактні структури даних.

**вміти:**

- вибирати для конкретної задачі відповідні технології та інструменти, що забезпечують можливість побудови ефективних веб-додатків.

**мати уявлення:**

- про сучасні фреймворки для розробки веб-додатків.

**Міждисциплінарні зв'язки:** дисципліні передують курси «Основи програмування», «Комп'ютерні мережі та HTML», дисципліна потрібна для подальшого вивчення курсу «Мережеве програмування на PHP»

## 3. Програма навчальної дисципліни

### Модуль 1

**Змістовий модуль 1. Розробка веб-додатків з використанням мови програмування JavaScript**

**ТЕМА 1. Предмет вивчення і задачі дисципліни «Скриптові мови для Web».** Огляд сучасних технологій, які використовуються при створенні веб-додатків. Місце дисципліни в навчальному плані. Основні концепції, вимоги, моделі реалізації технологій створення веб-додатків. Основні поняття мережних додатків. Огляд сучасних технологій для проектування веб-додатків.

**ТЕМА 2. Можливості JavaScript, як мови програмування загального призначення.** Базові мовні конструкції і функції. Об'єктна модель. Обробка подій.

**ТЕМА 3. Використанням JavaScript для створення веб-додатків.** Розробка динамічних сторінок на боці клієнта. Бібліотека JQuery. JavaScript-фреймворк AngularJS

**Модульний контроль**

### Модуль 2

**Змістовий модуль 1. Сучасні технології розробки веб-додатків**

**ТЕМА 4. Основні функціональні характеристики фреймворків.**

Фреймворк це базовий web-додаток, який контролює виконання модулів і

закладає загальний стандарт кодування. Він логічно розділений на ядро, завантажувач ядра, бібліотеку загальних функцій і сховище конфігурації. В сучасних фреймворках закладений базовий функціонал і набір архітектурних стандартів. MVC підхід при розробці веб-додатків. MVC - Модель–вигляд–контролер - архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення.

**ТЕМА 5. RESTful інтерфейс програмування застосунків.** REST - підхід до архітектури мережевих протоколів, які забезпечують доступ до інформаційних ресурсів. Базові принципи. Стандартні типи запитів. Взаємодія клієнт-сервер

**ТЕМА 6. Програмування серверної частини веб-додатків** Використання JavaScript для програмування серверної частини (Node.js). Асинхронна модель взаємодії. JavaScript–рушій Chrome V8. Пакетна екосистема. Базова інформація про типовий комплекс для розробки та роботи сучасних веб-додатків Apache+PHP+MySQL. Структура мови програмування PHP. Робота з базами даних на прикладі MySQL

### Модульний контроль

## 4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	денна форма					
	усього	у тому числі				с.р.
л		лаб	інд			
1	2	3	4	5	6	
<b>Модуль 1</b>						
<b>Змістовий модуль 1.</b>						
Тема 1	18	4	4		10	
Тема 2.	23	4	4		15	
Тема 3.	23	4	4		15	
Модульний контроль	2				2	
Разом за змістовим модулем 1	66	12	12		42	
<b>Усього годин</b>	66	12	12		42	
<b>Модуль 2</b>						
<b>Змістовий модуль 1.</b>						
Тема 4.	20	4	4		12	

Тема 5.	22	4	4		12	
Тема 6.	22	4	4		12	
Модульний контроль	2				2	
Разом за змістовим модулем 1.	62	12	12		38	
<b>Усього годин</b>	62	12	12		38	
Індивідуальне завдання						
<b>Контрольний захід</b>	7				7	
<b>Усього годин</b>	135	24	24		87	

### 5. Теми семінарських занять

№ п/п	Назва теми	Кількість годин
1		
2		
	<b>Разом</b>	

### 6. Теми практичних занять

№ п/п	Назва теми	Кількість годин
1		
2		
	<b>Разом</b>	

### 7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
		Денна форма навчання
1	Проектування додатку Загальна постановка задачі. Вибір платформи	4
2	Верстання статичної частини веб-додатку	4
3	Розробка клієнтської частини Web-додатку з використанням Javascript	4
4	Інтеграція в додаток одного з сторонніх сервісів	4
5	Встановлення налаштування та використання	4

	програмного середовища комплексу розробки клієнт-серверного веб-додатку	
6	Реалізація взаємодії клієнтської та серверної частин Web-додатку за допомогою технології AJAX	4
Разом		24

### 8. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Самостійне відпрацювання та повтор матеріалів, отриманих на практичних заняттях, відповідно переліку лабораторних занять	87
	<b>Разом</b>	87

### 9. Індивідуальні завдання

Непередбачено навчальним планом.

### 10. Методи навчання

1. За джерелами придбання знань – словесні: лекція (вступна, традиційна, проблемна, з помилками), бесіда (евристична), диспут, дискусія, робота з друкованими та інтернет-джерелами; наочні: ілюстрація, спостереження; практичні: вправа, лабораторна робота.
2. За характером пізнавальної діяльності тих, хто навчається – інформаційно-репродуктивний, репродуктивний, проблемне викладання, частково-пошуковий.
3. За логікою пізнання – індуктивний, дедуктивний, аналогій, вивідних знань.
4. Методи перевірки й оцінки знань, умінь, навичок: спостереження, усне опитування, контрольні роботи, програмований контроль, тестування (традиційне та машинне).

### 11. Методи контролю

Поточний контроль під час проведення лабораторних занять.

Тестовий контроль під час лекційних занять у формі письмових модульних контрольних робіт.

Форма підсумкового семестрового контролю успішності навчання: залік (письмово)

## 12. Критерії оцінювання та розподіл балів, які отримують студенти

### 12.1. Розподіл балів, які отримують студенти (кількісні критерії оцінювання)

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
<b>Змістовний модуль 1</b>			
Робота на лекціях	0...1	8	0...8
Виконання і захист лабораторних (практичних) робіт	4...8	2	8...16
Модульний контроль	10...20	1	10...20
<b>Змістовний модуль 2</b>			
Робота на лекціях	0...1	4	0...4
Виконання і захист лабораторних (практичних) робіт	4...8	4	16...32
Модульний контроль	10...20	1	10...20
Виконання і захист РГР (РР, РК)			
<b>Усього за семестр</b>			<b>60...100</b>

Семестровий контроль (залік) проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту студент має можливість отримати максимум 100 балів.

Білет для іспиту/заліку складається з одного теоретичного питання (кожне питання 30 балів) та двох практичних питань (кожне питання 35 балів).

### 12.2. Якісні критерії оцінювання

Необхідний обсяг знань для одержання позитивної оцінки: історію створення Javascript; особливості мови Javascript; лексичні основи (зарезервовані ключові слова, ідентифікатори, літери, роздільники); типи даних мови Javascript; роботу зі змінними; цілі типи даних; приведення типів; дійсні типи даних; логічний тип даних; арифметичні операції; логічні операції; умовні оператори if-else, switch; оператори повторення while, do-while, for; оператори переходу continue, break; робота з масивом; основи роботи String.

Необхідний обсяг вмінь для одержання позитивної оцінки: створювати проект для розробки програм мовою Javascript; розробляти алгоритм програми; створювати, редагувати та відлагоджувати застосунки мовою Javascript.



### 12.3 Критерії оцінювання роботи студента протягом семестру

**Задовільно (60-74).** Показати мінімум знань та умінь. Уявляти історію створення Javascript; особливості мови Javascript; лексичні основи (зарезервовані ключові слова, ідентифікатори, літери, роздільники); типи даних мови Javascript; роботу зі змінними; цілі типи даних; приведення типів; дійсні типи даних; логічний тип даних; арифметичні операції; логічні операції; умовні оператори if-else, switch; оператори повторення while, do-while, for; оператори переходу continue, break; робота з масивом; основи роботи String. Розуміти як створювати проект для розробки програм мовою Javascript; розробляти алгоритм програми; створювати, редагувати та відлагоджувати застосунки мовою Javascript. Розуміти про різні середовища розробки застосунків мовою Javascript; про особливості виконання Javascript програм

**Добре (75-89).** Твердо знати мінімум, здати тестування та поза аудиторну самостійну роботу. Досконало знати історію створення Javascript; особливості мови Javascript; лексичні основи (зарезервовані ключові слова, ідентифікатори, літери, роздільники); типи даних мови Javascript; роботу зі змінними; цілі типи даних; приведення типів; дійсні типи даних; логічний тип даних; арифметичні операції; логічні операції; умовні оператори if-else, switch; оператори повторення while, do-while, for; оператори переходу continue, break; робота з масивом; основи роботи String. Досконало вміти створювати проект для розробки програм мовою Javascript; розробляти алгоритм програми; створювати, редагувати та налагоджувати застосунки мовою Javascript. Досконало уявляти про різні середовища розробки застосунків мовою Javascript; про особливості виконання Javascript програм

**Відмінно (90-100).** Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти застосовувати їх.

#### Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

### 13.Методичне забезпечення

1. Розроблений лекційний курс. (//master/студенты/2019-2020/6 факультет/ 2 курс/скрипт\_мови\_Web)
2. Створений комплекс презентацій. (//master/студенты/2019-2020/6 факультет/ 2 курс/скрипт\_мови\_Web)

3. Лабораторні роботи. (//master/студенти/2019-2020/6 факультет/ 2 курс/скрипт\_мови\_Web/Лаб\_роботи)
4. Дібрані матеріали для самостійної роботи студентів. (//master/студенти/2019-2020/6 факультет/ 2 курс/скрипт\_мови\_Web)

## **14.Рекомендована література**

### **Базова**

1. Элизабет Фримен, Эрик Фримен Изучаем HTML, XHTML и CSS. М.: Питер, 2016, -720 с.
2. Дронов В.А. PHP, MySQL, HTML5 и CSS 3. Разработка современных динамических Web-сайтов/ В. А. Дронов. -СПб.: БХВ-Петербург, 2016. -688 с
3. Дунаев В.В. Основы WEB дизайна : самоучитель/ В. В. Дунаев. -2-е изд.. - СПб.: БХВ-Петербург, 2012 (СПб.). -479 с.
4. Kyle Simpson. You Don't Know JS (book series) - 1st Edition/ Электронный ресурс. Режим доступа: <https://github.com/getify/You-Dont-Know-JS/tree/1st-ed>
5. Дакетт Дж. Javascript и jQuery. Интерактивная веб-разработка/ Дакетт Дж.. - Эксмо, 2017. -600 с
6. Клименко Р.А. Веб-мастеринг: изучаем HTML5, CSS3, JavaScript, PHP, CMS, AJAX, SEO/ Р. А. Клименко. -СПб.: Питер, 2014. -508 с.
7. Мак-Дональд М. Создание Web-сайта : пер. с англ./ М. Мак-Дональд. -3-е изд.. -СПб.: БХВ-Петербург, 2013. -612 с.

### **Допоміжна**

1. Робин Никсон. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 .- СПб.: Питер, 2015. - 688 с.
2. Хант К. TCP/IP. Сетевое администрирование – Символ, 2009. –816с.
3. Викрам Васвани. Zend Framework. Разработка веб-приложений на PHP.- Питер, 2012, с.347
4. Пауэрс Д. PHP. Создание динамических страниц/ Д. Пауэрс. -М.: Рид Групп, 2012. -640 с
5. Колисниченко Д.Н. PHP и MySQL. Разработка веб-приложений/ Д. Н. Колисниченко. -5-е изд., [перераб. и доп.]. -СПб.: БХВ-Петербург, 2015. -592 с.
- 6.

## **15.Інформаційні ресурси**

1. Laravel framework: <https://laravel.com/docs/5.5>
2. Learn AngularJS: <https://www.codeschool.com/pages/angularjs-vs-angular>
3. Документація з Node.js: <https://nodejs.org/uk/docs/guides/>