ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ФАХОВИЙ БІЗНЕС-КОЛЕДЖ РІК ЗАСНУВАННЯ - 1966

Відділення інформаційних технологій

ЦК програмної інженерії та інтелектуальних систем

СИЛАБУС

Базова інформація про дисципліну	
Назва дисципліни	**SE017 Front-end програмування на React JS / Frontend Programming with React JS**
Рівень вищої освіти / фахової передвищої освіти	Фаховий молодший бакалавр
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення
Семестр	7 семестр (9 кл)
Курс	4 курс (на базі базової середньої освіти)
Анотація курсу	Навчальна дисципліна відноситься до вибіркових дисциплін курса. Дисципліна розрахована на студентів з базовим досвідом програмування на JavaScript і бажанням навчатися новим технологіям для розробки сучасних додатків. При проходженні даної дисципліни, студенти ознайомлюються з поняттям «front-end розробка», сучасними інструментами розробки веб-застосунків, підходами створення багатофункціональних систем з веб-інтерфейсом. На лабораторних заняттях опанують бібліотеки та інструментальні засоби для розробки клієнтської частини веб-застосунків, зокрема Angular, React, Vue. Студенти також ознайомляться з перспективними технологіями, що підтримуються сучасними браузерами та описані останніми версіями стандартів. В курсі передбачений контроль якості отриманих знань у вигляді експрес-контрольних та модульних контрольних робіт.
Сторінка курсу в MOODLE	http://78.137.2.119:2929/course/view.php?id=819
Мова викладання	Українська
Лектор курсу	Дмитрюк Валентин Віталійович Канали комунікації: СНД «Moodle»: повідомлення в чаті E-mail: valaimg2218@gmail.com
Місце дисципліни в освітній програмі	
Освітня програма	http://csbc.edu.ua/documents/otdel/moop_pr23.pdf

Інтегральна компетентність (ІК)	Здатність вирішувати типові спеціалізовані задачі інженерії програмного забезпечення, що вимагає застосування положень і методів відповідних наук (математики, інформатики, інформаційних технологій, тощо) та може характеризуватися певною невизначеністю умов; нести відповідальність за результати своєї діяльності; здійснювати контроль інших осіб у визначених ситуаціях.
Перелік загальних компетентностей (ЗК)	**ЗК:** – Знання та розуміння предметної області та розуміння професійної діяльності. – Здатність до пошуку, оброблення та аналізу інформації з різних джерел. – Здатність застосовувати знання у практичних ситуаціях.
Перелік спеціальних компетентностей (СК)	**СК:** – Здатність алгоритмічно та логічно мислити. – Здатність вдосконалювати знання і навички в галузі інформаційних технологій та усвідомлення важливості навчання протягом усього життя. – Здатність розробляти модулі і компоненти програмного забезпечення за допомогою типових алгоритмів та інструментів. – Здатність вибирати та використовувати ефективні інструментальні засоби розробки програмного продукту.
Перелік програмних результатів навчання	**РН:** – Систематизувати та узагальнювати інформацію про підходи, методи та засоби розробки супроводу програмного забезпечення. – Розробляти та супроводжувати програмне забезпечення. – Обирати інструментальні засоби, ефективні методи та здійснювати тестування програмних систем. – Аналізувати та узагальнювати необхідну інформацію з різних джерел та ресурсів для розв'язання професійних задач з урахуванням сучасних досягнень інформаційних технологій.
Опис дисципліни	
Структура навантаження на студента	Загальна кількість годин – 180 Кількість кредитів – 6 Кількість лекційних годин – 30 Кількість практичних занять – 30 Кількість годин для самостійної роботи студентів – 120 Форма підсумкового контролю – залік
Методи навчання	За подачею навчального матеріалу: методи готових знань, дослідницький метод. З огляду на мету навчання: методи здобуття нових знань, метод формування умінь і навичок, метод застосування знань на практиці, методи закріплення знань, умінь і навичок, методи перевірки і оцінювання знань, умінь і навичок
Зміст дисципліни	

Тема 1. Основи бібліотеки React	Перешкоди та труднощі. React як бібліотека. Новий синтаксис ECMAScript. Популярність функціонального JavaScript. Майбутнє бібліотеки React. Робота з файлами. Файловий репозиторій. Інструменти React-розробника. Встановлення Node.js.
Тема 2. Новий синтаксис JavaScript	Оголошення змінних в ES6 (const, let, шаблонні рядки, Параметри за замовчуванням). Стрілкові функції. Транспіляція ES6. Об'єкти та масиви ES6. Проміси. Класи. Модулі ES6. CommonJS
Тема 3. Функціональне програмування із застосуванням JavaScript	Значення поняття функціональності. Порівняння імперативності з декларативністю. Функціональні концепції. Незмінність. Чисті функції. Перетворення даних. Функції вищого порядка. Рекурсія. Композиція.
Тема 4. Чистий React	Налаштування сторінки. Віртуальний DOM. Елементи React. ReactDOM. Дочірні елементи. Конструювання елементів з даними. Компоненти React. React.createClass. React.Component. Функціональні компоненти, що не мають стану. Відображення DOM.
Тема 5. React з JSX	Елементи React у вигляді коду JSX. Поради щодо застосування JSX. Вкладені компоненти. Відображення масивів на JSX. Введення в Webpack. Завантажувачі Webpack.
Тема 6. Властивості, стан та дерево компонентів	Перевірка властивостей. Перевірка властивостей під час використання createClass. Властивості, які використовуються за замовчуванням. Перевірка властивостей, що настроюється. Класи ES6 та функціональні компоненти, не мають стану. Зворотний потік даних. Посилання у функціональних компонентах, які мають стан. Управління станом React. Впровадження стану компонента. Ініціалізація стану із властивостей. Стан в середині дерева компонента. Передача властивостей вниз по дереву компонентів. Передача даних по дереву компонентів
Тема 7. Удосконалення компонентів	Життєві цикли компонентів. Життєвий цикл установки. Життєвий цикл оновлення, React.Children. Підключення бібліотек JavaScript. Створення запитів за допомогою Fetch. Підключення D3 Timeline. Компоненти вищого порядка. Управління станом за межами React. Реалізації Flux.
Тема 8. Redux	Стан. Дії. Перетворювачі. Перетворювач кольору. Перетворювач кольорів Перетворювач сортування. Збереження в localStorage. Функції проміжної ланки.
Тема 9. Використання технології React Redux	React Redux явна передача сховища. Передача сховища через контекст. Порівняння презентаційних та контейнерних компонентів. Провайдер React Redux. Функція connect бібліотеки React Redux .
Тема 10. React та сервер	Порівняння ізоморфізму з універсалізмом. Код React, який відображається на сервері. Універсальний органайзер кольорів. Універсальний Redux. Універсальна маршрутизація. Обмін даними з сервером. Виконання дій на сервері. Дії з Redux Thunks.

Політика дисципліни	
Політика відвідування	Регулярне відвідування всіх видів занять, своєчасність виконання самостійної роботи. За об'єктивних причин (наприклад, хвороба, міжнародне стажування) навчання зорганізується в он-лайн формі за погодженням із керівником курсу.
Політика щодо дедлайнів та перескладання	Роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку.
Академічна доброчесність	У випадку недотримання політики академічної доброчесності (плагіат, самоплагіат, фабрикація, фальсифікація, списування, обман, хабарництво) передбачено повторне проходження оцінювання.
Використання ШІ	Використання ШІ під час виконання завдань регламентується Політикою «Використання ШІ в освітньому процесі ЧДБК» Завдання мають маркування регламенту використання ШІ.

Система оцінювання

Система оцінювання підсумкової успішності студентів поділяється на **поточний контроль** та **семестровий контроль**.

Поточний контроль здійснюється протягом семестру і охоплює всі види аудиторної роботи (практичні заняття) та виконання індивідуальних завдань. Максимальна кількість балів, яку студент може набрати за цей вид контролю, становить 100.

Підсумковий контроль

Відбувається у формі диференційного заліку (у формі підсумкового модульного контролю).

Розрахунок підсумкової оцінки

Підсумкова оцінка базується виключно на балах, накопичених протягом семестру (S). Ваговий коефіцієнт у цьому випадку становить 1.

Формула: О=S×1

Види навчальної роботи	Загальна кількість балів
Усні відповіді за темами 1по 1 балу	5
Практична робота за темами 1 по 5 балів	55
Модульні контрольні роботи (2 к.р. по 10 балів)	20
Індивідуальна самостійна робота (проєкт)	20
Разом	100

Критерії оцінювання для кожного виду навчальної

Критерії оцінювання усних відповідей:

1 б. – Студент відповів правильно на запитання.

0,5 б. – Студент частково правильно відповів на запитання.

0 б. – Студент не зміг відповісти на запитання.

Критерії практичних робіт

5 б. – Студент виконав всі завдання практичної роботи без помилок, а також правильно оформив звіт.

4 б. – Студент виконав всі завдання практичної роботи, але допустився помилки в одному з них та в оформленні звіту.

3 б. – Студент виконав всі завдання практичної роботи, але допустився помилки в двох з них та є недоліки в оформленні звіту.

2 б. – Студент виконав лише частину завдань практичної роботи, але має помилки при виконанні, а також є недоліки в оформленні звіту.
1 б. – Студент намагався виконати практичну роботу, але в завданнях є помилки та звіт оформлено невірно.
0 б. – студент не виконав практичної роботи та не здав звіт.

Критерії оцінювання модульних робіт

10 б. – виконано всі завдання без помилок, відповіді повні й обґрунтовані.
9 б. – виконано майже всі завдання, допущено декілька незначних помилок.
8 б. – виконано більшу кількість завдань, але є окремі помилки та недоречності у відповідях.
7 б. – виконано три завдання, але з помітними помилками.
6 б. – виконано два завдання повністю та половину третього, але частина з них має помилки.
5 б. – виконано два завдання, але продемонстровано розуміння основного матеріалу.
4 б. – виконано деякі завдання правильно, але більшість з помилками.
3 б. – робота має лише деякі правильні елементи у відповідях.
2 б. – виконано мінімальний обсяг завдань, знання без глибокого розуміння.
1 б. – студент намагався виконати завдання, але відповіді містять помилки й потребують корекції.

Критерії оцінювання індивідуальних робіт (проєктів)

20 б. – завдання виконано повністю, без жодної помилки; звіт правильно й акуратно оформлений, відповіді повні, логічні та аргументовані.
19 б. – усі завдання виконані, але є 1–2 несуттєві неточності у відповідях чи оформленні.
18–15 б. – виконано більшість завдань правильно, звіт загалом оформлений належно; наявні окремі помилки у змісті або дрібні недоліки в структурі/оформленні.
4–13 б. – завдання виконані частково, відповіді містять суттєві неточності чи неповноту; у звіті є помилки в оформленні або бракує аргументації.
12–10 б. – виконано половину чи трохи більше завдань, відповіді часто неправильні або поверхові; звіт має помітні недоліки у змісті та структурі.
9–7 б. – зроблено спробу виконати більшість завдань, проте більшість відповідей неправильні або неповні; звіт оформлено формально, із значними помилками.
6–4 б. – завдання виконані частково, правильних відповідей небагато; звіт майже не відповідає вимогам оформлення.
3–1 б. – зроблено лише символічну спробу виконати завдання; відповіді в основному неправильні; звіт оформлено вкрай слабко.
0 б. – завдання не виконано, звіт відсутній.

ECTS	Бали	Зміст
A	90-100	Бездоганна підготовка в широкому контексті
B	80-89	Повні знання, міцні вміння
C	70-79	Хороші знання та вміння
D	65-69	Задовільні знання, стереотипні вміння
E	60-64	Виконання мінімальних вимог діяльності в стандартних умовах
FX	35-59	Слабкі знання, відсутність умінь
F	1-34	Необхідний повторний курс

Список рекомендованих джерел

1. Marijn Haverbeke. Eloquent JavaScript, 3rd Edition: A Modern Introduction to Programming No Starch Press, 2018. 472 p. Also available online https://eloquentjavascript.net/
2. David Flanagan. JavaScript: The Definitive Guide: Master the World's Most-Used Programming Language. 7th Edition O'Reilly Media, 2020. 706 p.
3. Boris Cherny. Programming TypeScript: Making Your JavaScript Applications Scale. O'Reilly Media, 2019. 324 p.
4. Aristeidis Bampakos, Pablo Deeleman. Learning Angular: A no-nonsense beginner's guide to building web applications with Angular 10 and TypeScript. 3rd Edition Packt Publishing, 2020. 430 p.
5. Alex Banks, Eve Porcello. Learning React: Modern Patterns for Developing React Apps. 2nd Edition O'Reilly Media, 2020. 310 p.
6. Heitor Ramon Ribeiro. Vue.js 3 Cookbook: Discover actionable solutions for building modern web apps with the latest Vue features and TypeScript. Packt Publishing, 2020. 562 p.

Інтернет ресурси

1. Посібник: знайомство з React. URL: https://uk.legacy.reactjs.org/tutorial/tutorial.html
2. Metatin. Сайт вивчення мов програмування. URL: https://metanit.com/web/angular2/1.1.php
3. Vue. Офіційна документація фреймфорку. URL: https://vuejs.org/v2/guide/
4. NodeJS. Офіційна документація мови програмування. URL: https://nodejs.org/uk/docs/