

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

ЗАТВЕРДЖЕНО

На засіданні кафедри
економічної кібернетики і
системного аналізу
Протокол № 1 від 22.08.2023 р.

ПОГОДЖЕНО

Проректор з навчально-методичної роботи



Каріна НЕМАШКАЛО

WEB-ТЕХНОЛОГІЇ
робоча програма навчальної дисципліни (РПНД)

Галузь знань	12 «Інформаційні технології»
Спеціальність	124 «Системний аналіз»
Освітній рівень	перший (бакалаврський)
Освітня програма	Управління складними системами

Статус дисципліни	обов'язкова
Мова викладання, навчання та оцінювання	українська

Розробники:
к.е.н., доцент

Роман ЯЦЕНКО

викладач

Антон ЯКОВЛІВ

Завідувач кафедри
економічної кібернетики
і системного аналізу

Лідія ГУР'ЯНОВА

Гарант програми

Оксана ПАНАСЕНКО

Харків
2024

ВСТУП

Вивчення навчальної дисципліни «Web-технології» надає здобувачам спеціальності «Системний аналіз» важливу інструментарій для розуміння та розробки сучасних веб-застосунків, які визначають сучасну інформаційну парадигму. У контексті системного аналізу, де важливо розуміти та вдосконалювати інформаційні системи, освоєння Web-технологій є вирішальним, оскільки вони визначають принципи побудови користувацьких інтерфейсів, взаємодії з даними та забезпечують можливість створення масштабованих та динамічних веб-застосунків. Знання цієї дисципліни не лише сприяє розширенню технічних навичок, але й підготовлює здобувачів до вирішення сучасних завдань в галузі розробки програмного забезпечення, а також стимулює їхню конкурентоспроможність на ринку праці. Вивчення Web-технологій впроваджує здобувачів у світ сучасних тенденцій розробки веб-застосунків та розширює їхні можливості в професійному розвитку в сфері системного аналізу та розробки інформаційних систем.

Мета навчальної дисципліни є надання здобувачам вищої освіти системи теоретичних і практичних знань з принципів створення web-сайтів, опису і розмітки web-сторінок, методів і навичок програмування у web-середовищі.

Основним завданням навчальної дисципліни «Web-технології» є вивчення теоретичних і практичних засад web-дизайну, основ будови, методів управління web-сайтами, формування у здобувачів навичок використання отриманих знань на практиці, створення власного web-сайту.

Предметом навчальної дисципліни є методи й інструменти побудови та застосування Web-технологій для підвищення ефективності функціонування економічних систем в сучасних умовах розвитку ринкового середовища.

Об'єктом навчальної дисципліни "Web-технології" є вивчення основних концепцій та технологій, пов'язаних з розробкою та підтримкою веб-застосунків. Здобувачі здобувають розуміння архітектурних принципів веб-застосунків, навички роботи з клієнтською та серверною частинами, а також основи веб-дизайну та взаємодії з базами даних. Дисципліна включає в себе вивчення мов програмування для веб-розробки, протоколів передачі даних, стандартів безпеки та інших аспектів, необхідних для створення функціональних та ефективних веб-застосунків.

Результати навчання та компетентності, які формує навчальна дисципліна визначено в табл. 1.

Результати навчання та компетентності, які формує навчальна дисципліна

Результати навчання	Компетентності, якими повинен оволодіти здобувач освіти
РН8. Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій.	КЗ 2. Здатність застосовувати знання у практичних ситуаціях КЗ 14. Здатність оцінювати та забезпечувати якість виконуваних робіт
РН10. Знати архітектуру сучасних обчислювальних систем і комп'ютерних мереж.	КФ 8. Здатність організовувати роботу з аналізу та проектування складних систем, створення відповідних інформаційних технологій та програмного забезпечення

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Зміст навчальної дисципліни

Змістовий модуль 1.

Основи Web-технологій

Тема 1. Вступ до Web-технологій. Планування Web-сайта

1.1. Мережа Інтернет та сервіси.

Інтернет. Сервіси інтернет. Служба WWW. Web-сервер. Web-браузер.

1.2. Гіпертекст як основа WWW.

Гіпертекст. Uniform Resource Locator – уніфікований покажчик ресурсу.

Види вебсторінок.

1.3. Типи Web-сайтів.

Персональний вебсайт. Сайт організації. Тематичний сайт. Комерційний сайт.

1.4. Планування Web-сайту.

Визначення призначення сайту. Опис аудиторії. Планування контенту.

Планування структури. Планування вебсторінок. Планування навігації.

1.5. Ролі у Web-дизайн.

Команда вебпроекту. Види ролей.

Тема 2. Мова гіпертекстової розмітки HTML.

2.1. Загальні відомості про HTML.

HTML. Історія HTML. Властивості HTML5.

2.2. Тегова модель

Основна ідея. Тег. Загальна структура HTML-документу. Елементи HTML. Атрибути тегів. Основні правила синтаксису HTML.

2.3. Оформлення текстової інформації.

Заголовки. Абзаци. Перенесення рядків. Списки. Марковані списки. Нумеровані списки. Списки визначень. Вкладені списки. Посилання. Елементи для форматування тексту. Вкладеність елементів.

2.4. Створення таблиць.

Засоби використання таблиць. Елементи таблиці. Групування строк таблиці. Вирівнювання елементів таблиці. Фоновий колір або малюнок. Вкладення однієї таблиці в іншу.

Тема 3. Каскадні таблиці стилів CSS.

3.1. Сутність таблиць стилів.

Мова CSS. Історія CSS. Можливості CSS. Використання CSS.

3.2. Визначення вбудованого стилю.

Загальний html атрибут style. Чому style краще не використовувати.

3.3. Формування таблиці стилів.

Правила побудови CSS. Види селекторів. Класи та ідентифікатори елементів. Спадкування. Каскадування. Пріоритети стилів CSS.

3.4. Зовнішні таблиці стилів.

Підтримка CSS браузером. Методи підключення CSS до документа.

3.5. Стилеві властивості.

Перелік стильових властивостей. Ієрархія стильових властивостей. Взаємодія стильових властивостей.

Тема 4. Методи адаптивної верстки.

4.1. Модуль Flexbox.

Основний принцип Flexbox. Основні переваги. Вирівнювання елементів. Перенесення елемента. Усі властивості Flexbox

4.2. Модуль CSS Grid.

CSS Grid Layout. CSS Grid макет. Розташування елементів. Позиціонування елементів. Усі властивості CSS Grid.

4.3. Медіазапити.

Типи пристроїв. Медіазапит. Засоби додавання медіазапитів. Тип носія. Характеристики носія. Metatag viewport.

4.4. CSS-властивості для створення анімації.

Основи анімацій за допомогою CSS. Конфігурації анімацій. Визначення послідовності анімації за допомогою ключових кадрів. Налаштування повторення. Використання подій анімації. Реєстрація подій.

Тема 5. CSS-фреймворки.

5.1. Призначення та функції CSS-фреймворків.

Що таке фреймворк. Переваги та недоліки фреймворків. Використання фреймворків.

5.2. Фреймворк Bulma.

Утиліти фреймворку. Сітка Vulma. Макети Vulma. Елементи та компоненти Vulma.

5.3. Фреймворк Bootstrap.

Twitter Bootstrap. Bootstrap Grid

Змістовий модуль 2. Інтерактивні Web-технології

Тема 6. Система керування версіями Git

6.1. Розподілені системи керування версіями програмного забезпечення.

Система керування версіями. Види систем контролю версії. Виникнення конфліктів та їх вирішення. Поширені системи керування версіями.

6.2. Система Git.

Коротка історія Git. Командна строка Git. Інсталювання та налаштування Git.

6.3. Команди управління репозиторієм та роботи з гілками.

Отримання репозитарію. Запис змін у репозитарій. Перегляд історії змін. Скасування змін у репозиторії.

6.4. Платформа GitHub.

Що таке GitHub. Використання GitHub у власних проектах.

Тема 7. Фреймворк для розробки web-застосунків Django

7.1. Історія та можливості web-фреймворку Django.

Історія фреймворку Django. Актуальність фреймворку Django. Сфера використання фреймворку Django. Переваги та недоліки фреймворку Django.

7.2. Створення проекту та додатків Django.

Підготовка робочого середовища. Створення та налаштування проекту. Створення додатків Django.

7.3. Робота з маршрутами.

Планування сторінок та переходів між. Налаштування глобальної маршрутизації проекту. Налаштування маршрутизації у додатках проекту.

7.4. Підходи до створення контролерів.

Огляд типів контролерів. Переваги та недоліки контролерів-функцій. Переваги та недоліка контролерів-класів. Пов'язання контролерів з маршрутами.

7.5. Проектування БД та моделей.

Огляд принципів створення бази даних. Використання класу models.Model. Створення бази даних sqlite. Методи заповнення та редагування бази даних. Методи отримання інформації з бази даних.

7.6. Розробка шаблонів сторінок.

Створення шаблонів сторінок. Додавання шаблонів у контролери. Використання функцій шаблонів Django.

7.7. Налаштування системи авторизації.

Огляд стандартного інструментарію авторизації. Створення нового інструментарію на основі старого. Оновлення прав доступу до контенту в залежності від авторизації.

Тема 8. Язык програмування JavaScript

8.1. Основи JavaScript.

Що таке JavaScript. Історія JavaScript. Сфери використання JavaScript.

8.2. Стиль та якість програмного коду.

Ключові поняття JavaScript. Синтаксис JS. Посібники зі стилю коду. Автоматизовані засоби перевірки (лінтери).

8.3. Об'єкти як базовий тип даних.

Типи даних. Сутність об'єктів. Визначення об'єктів.

8.4. Методи роботи з типами даних.

Методи типів даних. Правила перетворення типів даних.

8.5. Розширена робота з функціями.

Що таке функції. Як їх використовувати в JavaScript. Які є особливості у використанні функцій.

8.6. Властивості та наслідування об'єктів.

Відображення об'єктів. Властивості об'єктів. Доступ до властивостей об'єктів. Конструктори об'єктів.

8.7. Створення класів.

Що таке класи. Як їх використовувати в JavaScript. Які є особливості у використанні класів.

8.8. Обробка помилок.

Обробка помилок "try..catch". Синтаксис «try...catch». Об'єкт помилки. Використання «try...catch». Генерація власних помилок

8.9. Проміси та асинхронні методи програмування.

Що таке асинхронні методи програмування. Опис об'єкту Promise. Властивості об'єкту Promise. Методи об'єкту Promise.

Тема 9. Управління DOM за допомогою jQuery

9.1. Об'єктна модель документа (DOM).

Що таке DOM. Стандарти DOM. Реалізація DOM у веббраузерах. Модель документа.

9.2. Події браузера.

Що таке подія браузера. Обробники подій. Використання властивостей DOM-об'єкта. Доступ до елемента через this. Об'єкт події.

9.3. Бібліотека jQuery.

Що таке jQuery. Обхід і маніпуляції DOM. Обробка подій.

9.4. Маніпуляції з об'єктною моделлю документа.

Введення в маніпуляції з DOM. Отримання та зміна контенту. Отримання та зміна атрибутів. Отримання та зміна CSS-класів.

9.5. Ефекти jQuery.

Ефекти jQuery. Анімація в jQuery.

Перелік лабораторних занять за навчальною дисципліною наведено в табл. 2.

Таблиця 2

Перелік лабораторних занять

Тема та назва лабораторного заняття	Зміст
Тема 2. ЛР 1. Введення в HTML	Знайомство з мовою гіпертекстової розмітки HTML
Тема 2. ЛР 2. Текстові елементи HTML	Оволодіння навичками використання текстових елементів HTML
Тема 2. ЛР 3. Знайомство з таблицями	Оволодіння навичками використання таблиць HTML
Тема 3. ЛР 4. Селектори, спадкування та каскадування	Знайомство зі спеціальною мовою стиля сторінок CSS
Тема 3. ЛР 5. Оформлення тексту за допомогою CSS	Оволодіння навичками використання CSS для оформлення тексту
Тема 4. ЛР 6. Flexbox та Grid	Оволодіння навичками використання технологій Flexbox та Grid
Тема 4. ЛР 7. Анімація переходів, медіа-запити, адаптивна сітка	Оволодіння навичками використання технологій анімацій у CSS та створення адаптивних сторінок
Тема 5. ЛР 8. Верстка лендінгу за допомогою Bootstrap 4	Знайомство з функціоналом фреймворку Bootstrap 4
Тема 5. ЛР 9. Використання Bootstrap	Поглиблення навичок роботи з функціоналом фреймворку Bootstrap та налаштування робочого місця
Тема 6. ЛР 11. Робота з Git	Знайомство з функціоналом розподіленої систему управління версіями Git
Тема 7. ЛР 12. Створення персонального сайту на Django	Знайомство з функціоналом фреймворку Django, створення проекту
Тема 7. ЛР 13. Форми та додаток користувачів	Продовження знайомства з функціоналом фреймворку Django, створення елементів сайту
Тема 7. ЛР 14. Налаштування доступу для користувачів	Продовження знайомства з функціоналом фреймворку Django, налаштування системи доступу користувачів до контенту сайту
Тема 7. ЛР 15. Оформлення сайту	Використання Bootstrap для оформлення зовнішнього вигляду сайту
Тема 8. ЛР 16. Основи JavaScript та взаємодія с користувачем	Знайомство з мовою JavaScript
Тема 8. ЛР 17. Умовний оператор, цикл с лічильником, цикл з умовою	Використання умовних операторів та циклів у мові JavaScript
Тема 8. ЛР 18. Строки, функції та рекурсії	Робота зі строками та функціями у мові JavaScript
Тема 8. ЛР 19. Масиви	Робота з масивами у мові JavaScript

Перелік самостійної роботи за навчальною дисципліною наведено в табл. 3.

Перелік самостійної роботи

Назва теми	Зміст
Тема 2.	Виконання першого етапу розроблення персонального сайту
Тема 3.	Виконання другого етапу розроблення персонального сайту
Тема 4.	Виконання третього етапу розроблення персонального сайту
Тема 5.	Створення презентації за обраним CSS фреймворком
Тема 7.	Створення власного проекту за допомогою фреймворку Django

Кількість годин лекційних і лабораторних занять та годин самостійної роботи наведено в робочому плані (технологічній карті) з навчальної дисципліни.

МЕТОДИ НАВЧАННЯ

При викладанні дисципліни “Web-технології” для активізації навчального процесу передбачено застосування сучасних навчальних технологій, таких, як: проблемні лекції; міні-лекції; презентації, виконання індивідуальних творчих завдань.

Проблемна лекція «Огляд популярних Javascript-фреймворків» в темі 8 спрямована на розвиток логічного мислення здобувачів. Коло питань теми обмежується двома-трьома ключовими моментами, увага здобувачів концентрується на матеріалі, що не знайшов відображення в підручниках, використовується досвід закордонних навчальних закладів з роздачею здобувачам під час лекцій друкованого матеріалу та виділенням головних висновків з питань, що розглядаються. При викладанні лекційного матеріалу здобувачам пропонуються питання для самостійного розмірковування. При цьому лектор задає запитання які спонукають здобувача шукати розв’язання проблемної ситуації. Така система примушує здобувачів сконцентруватися і почати активно мислити в пошуках правильної відповіді.

На початку проведення проблемної лекції необхідно чітко сформулювати проблему, яку необхідно вирішити здобувачам. При викладанні лекційного матеріалу слід уникати прямої відповіді на поставлені запитання, а висвітлювати матеріал таким чином, щоб отриману інформацію здобувач міг використовувати при розв’язанні проблеми.

Міні-лекція «Система керування завданнями JIRA» в темі 6 передбачає викладання навчального матеріалу за короткий проміжок часу й характеризуються значною ємністю, складністю логічних побудов, доказів та узагальнень. Міні-лекції проводяться, як правило, як частина заняття-дослідження.

На початку проведення міні-лекції за вказаною вище темою лектор акцентує увагу здобувачів на необхідності представити викладений лекційний

матеріал у так званому структурно-логічному вигляді. На розгляд виносяться питання, які зафіксовані у плані лекцій, але викладаються вони стисло. Лекційне заняття, проведене у такий спосіб, побуджує у здобувача активність та увагу при сприйнятті матеріалу, а також спрямовує його на використання системного підходу при відтворенні інформації, яку він одержав від викладача.

Презентації результатів виконання самостійного завдання щодо CSS фреймворків за темою 5 – це виступи перед аудиторією, що використовуються для представлення звіту про виконання індивідуальних завдань. Однією з позитивних рис презентації та її переваг при використанні в навчальному процесі є обмін досвідом, який здобули здобувачі при роботі над індивідуальним завданням або у певній малій групі.

ФОРМИ ТА МЕТОДИ ОЦІНЮВАННЯ

Університет використовує накопичувальну (100-бальну) систему оцінювання.

Система оцінювання сформованих компетентностей у здобувачів враховує види занять, які згідно з програмою навчальної дисципліни передбачають лекційні, лабораторні заняття, а також виконання самостійної роботи. Контрольні заходи включають:

поточний контроль, що здійснюється протягом обох семестрів під час проведення лекційних, лабораторних занять і оцінюється сумою набраних балів (максимальна сума – 60 балів; мінімальна сума, що дозволяє здобувачу скласти іспит, – 35 балів);

модульний контроль, що проводиться у формі колоквиуму як проміжний міні-екзамен з ініціативи викладача з урахуванням поточного контролю за змістовий модуль 1 і має на меті інтегровану оцінку результатів навчання здобувача після вивчення матеріалу з логічно завершеної частини дисципліни – змістового модуля 1;

підсумковий контроль, що проводиться у формі екзамену під час сесії другого семестру, відповідно до графіку навчального процесу.

Поточний та модульний контроль оцінювання знань протягом змістового модуля 1. Основи Web-технологій включає:

поточні контрольні роботи за темами. Загальна кількість балів – 24;

виконання завдань під час лабораторних занять (2 бали за одне завдання).

Загальна кількість балів – 20;

виконання самостійних індивідуальних завдань. Загальна кількість балів – 25;

презентація результатів виконання самостійного завдання щодо CSS фреймворків – 6 балів;

модульний контроль у формі залікового завдання – 25 балів.

Поточний контроль оцінювання знань протягом змістового модуля 2. Інтерактивні Web-технології включає:

поточні завдання за темами. Загальна кількість балів – 40;

презентація результатів виконання самостійного завдання за темою 9 – 8 балів;

виконання контрольної роботи за темою 8. Язык програмування JavaScript. Загальна кількість балів – 12.

Підсумковий контроль знань та компетентностей здобувачів з навчальної дисципліни здійснюється на підставі проведення семестрового екзамену, завданням якого є перевірка розуміння здобувачем програмного матеріалу в цілому, логіки та взаємозв'язків між окремими розділами, здатності творчого використання накопичених знань, вміння формулювати своє ставлення до певної проблеми навчальної дисципліни тощо.

Максимальна сума балів, яку може отримати здобувач вищої освіти під час екзамену (іспиту) – 40 балів. Мінімальна сума, за якою екзамен (іспит) вважається складеним – 25 балів.

Здобувача слід вважати атестованим, якщо сума балів, одержаних за результатами підсумкової/семестрової перевірки успішності, дорівнює або перевищує 60. Мінімально можлива кількість балів за поточний і модульний контроль упродовж семестру – 35 та мінімально можлива кількість балів, набраних на екзамені, – 25.

Більш детальну інформацію щодо системи оцінювання наведено в робочому плані (технологічній карті) з навчальної дисципліни.

Приклад екзаменаційного білету

Стереотипні завдання

Питання 1. Які з наведених тегів не є логічними?

Виберіть одну відповідь:

- a. <STRIKE>
- b. <VAR>
- c. <INS>
- d. <CITE>

e. вірної відповіді немає

Питання 2. Визначає медіа-тип, який використовується для кодування і пересилки певного типу даних, такий атрибут як:

Виберіть одну відповідь:

- a. method
- b. вірної відповіді немає
- c. action
- d. enctype
- e. style

Питання 3. Який тег оголошує маркований список?

Виберіть одну відповідь:

- a.
- b.
- c. вірної відповіді немає
- d.
- e. <DL>

Питання 4. Який тег вставляє розрив рядка в текст?

Виберіть одну відповідь:

- a. вірної відповіді немає
- b.

- c. <HR>
- d. <P>
- e. <TR>

Питання 5. Тег <TITLE> повинен знаходитися між тегами:

Виберіть одну відповідь:

- a. <BODY> </BODY>
- b. <TABLE> </TABLE>
- c. вірної відповіді немає
- d. <CENTER> </CENTER>
- e. <HEAD> </HEAD>

Питання 6. Атрибут start належить тегу:

Виберіть одну відповідь:

- a.
- b. <TYPE>
- c. вірної відповіді немає
- d.
- e.

Питання 7. Унікальним ідентифікатором елемента є:

Виберіть одну відповідь:

- a. вірної відповіді немає
- b. запис, що складається з селектора і визначення стилю
- c. селектор, записаний прописом
- d. селектор, якому передує #
- e. елемент, до якого відноситься певний стиль

Питання 8. Елементи COL і COLGROUP відповідають за групування:

Виберіть одну відповідь:

- a. стовпців таблиці
- b. декількох таблиць
- c. рядків таблиці
- d. вірної відповіді немає
- e. довільного блоку комірок таблиці

Питання 9. Класи стилів задаються:

Виберіть одну відповідь:

- a. записами, що складаються з селектора і визначення стилю
- b. селектором, записаним з крапкою на початку
- c. елементами, до яких відноситься певний стиль
- d. селектором, якому передує
- e. вірної відповіді немає

Питання 10. Задає файл із зовнішнім листом стилів атрибут:

Виберіть одну відповідь:

- a. src
- b. вірної відповіді немає
- c. rel
- d. type
- e. href

Діагностичне завдання

Дан одновимірний масив числових значень, що нараховує N елементів. З елементів вихідного масиву побудувати два нових. У першій повинні входити тільки елементи з позитивними значеннями, а в другій – тільки елементи з негативними значеннями.

Евристичне завдання

1. Створити HTML-файл `index.html`, який має заголовок, що відображається у заголовку вкладки браузера та містить автора і дату виконання завдання. На сторінці навести основні відомості про студента-виконавця екзаменаційного завдання (ПІБ, група, навчальна дисципліна, назва завдання тощо) та фото.

2. На сторінці `index.html` також навести опановані протягом навчального року Web-технології. Сторінка повинна містити такі HTML-елементи, як список та таблиця.

3. Створити HTML-файл `task.html`. Сторінка повинна містити текст програми розв'язку діагностичного завдання (програмування мовою JavaScript) та посилання чи кнопку, що запускає цю програму на виконання. Зв'язати взаємними посиланнями `index.html` та `task.html`.

4. Усі стилі HTML-файлів винести в окремий CSS-файл `style.css`, який підключається до `index.html` та `task.html`. Додатково можна реалізувати адаптивність для HTML-файлів за допомогою будь-якої технології чи фреймворку.

Критерії оцінювання

Екзаменаційний білет охоплює програму дисципліни і передбачає визначення рівня знань та ступеня опанування здобувачами компетентностей. Кожен екзаменаційний білет складається із 10 стереотипних тестових завдань та 2 практичних завдань (діагностичне та евристичне завдання), які передбачають вирішення типових професійних завдань фахівця на робочому місці та дозволяють діагностувати рівень теоретичної підготовки здобувача і рівень його компетентності з навчальної дисципліни.

Загальна оцінка формується за наступною формулою:

кількість балів, отриманих за виконання кожного стереотипного завдання +
кількість балів, отриманих за виконання кожного діагностичного завдання +
кількість балів, отриманих за виконання евристичного завдання

Екзаменаційний білет включає:

Стереотипні тестові завдання: максимальна кількість балів – 10.

Діагностичне завдання: максимальна кількість балів – 10.

Евристичне завдання: максимальна кількість балів – 20.

1. Виконання стереотипного завдання полягає у розв'язанні 10 тестових запитань закритого типу, кожне з яких оцінюється в один бал.

2. Виконання діагностичного завдання полягає у розв'язанні завдання на програмування з використанням базових алгоритмічних конструкцій, яке оцінюється в 10 балів. Потрібно надати розв'язок завдань у вигляді тексту програми мовою JavaScript. Програмний код повинен супроводжуватися коментарями, що пояснюють роботу програми. За неповного виконання завдання чи помилок у програмному коді оцінка за виконання буде пропорційно зменшена.

3. Перевірка виконання евристичного завдання відбувається наступним чином.

3.1. Створено HTML-файл `index.html`, який має заголовок, що відображається у заголовку вкладки браузера та містить автора і дату виконання завдання. На сторінці наведено основні відомості про студента-виконавця екзаменаційного завдання (ПІБ, група, навчальна дисципліна, назва завдання тощо) та фото (максимум – 5 балів).

3.2. На сторінці `index.html` також наведено опановані протягом навчального року Web-технології. Сторінка повинна містити такі HTML-елементи, як список та таблиця (максимум – 5 балів).

3.3. Створено HTML-файл `task.html`. Сторінка повинна містити текст програми розв'язку діагностичного завдання (програмування мовою JavaScript) та посилання чи кнопку, що запускає цю програму на виконання. Зв'язано взаємними посиланнями `index.html` та `task.html` (максимум – 5 балів).

3.4. Усі стилі HTML-файлів винесено в окремий CSS-файл `style.css`, який підключається до `index.html` та `task.html`. Додатково може бути реалізована адаптивність для HTML-файлів за допомогою будь-якої технології чи фреймворку (максимум – 5 балів).

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна

1. Мельник Р.А. Програмування веб-застосувань (фронт-енд та бек-енд). – Л.: Львівська політехніка, 2018 – 248 с.
2. Пасічник В. В. Веб-технології та веб-дизайн. Книга 1. Веб-технології. –К.: Магнолія, 2021. – 336 с.
3. Робсон Е., Фрімен Е. Head First. Програмування на JavaScript. – Х.: ФАБУЛА, 2022. – 672 с.
4. Яценко Р.М. Опорний конспект лекцій «Web-технології» [Електронний ресурс]. – Режим доступу : <http://pns.hneu.edu.ua/course/view.php?id=7113>.
5. Web-технології та Web-дизайн: застосування мови HTML для створення електронних ресурсів : навч. посіб. / І. Л. Бородкіна, Г. О. Бородкін.– Київ: Видавництво Ліра-К, 2020. – 212 с.

Додаткова

6. Молчанов В. П. Технології розробки WEB-ресурсів [Електронний ресурс] : навч. посіб. / В. П. Молчанов, О. К. Пандорін ; Харківський національний економічний університет ім. С. Кузнеця. - Електрон. текстові дан. (7,94 МБ). - Харків : ХНЕУ ім. С. Кузнеця, 2019. - 129 с. : іл. - Загол. з титул. екрану. - Бібліогр.: с. 126.
7. Плеханова Г. О. Аналіз сучасних засобів та методів веб-розробки / Г. О. Плеханова, Т. С. Мисько // Інформаційні технології та системи : матеріали міжнар. наук.-практ. конф., 9-10 квіт. 2020 р. – Х. : ХНЕУ ім. С. Кузнеця, 2020. – С. 20.– Режим доступу : <http://www.repository.hneu.edu.ua/handle/123456789/23350>
8. Плоха О. Б. Удосконалення якості веб-додатків шляхом впровадження тестування навантажень / О. Б. Плоха, Д. Є. Федоренко // Інформаційні технології та системи: матер. міжнар. наук.-практ. конф., м. Харків, 14-15 квіт. 2022 р. : тези допов. – Харків : ХНЕУ ім. С. Кузнеця, 2022. – Секція 1. Інформаційні технології в організаційно-технічних системах. – Ч. 1. – С. 17.– Режим доступу : <http://www.repository.hneu.edu.ua/handle/123456789/27150>
9. Babenko V. O. MarkHub Cloud Online Editor as a modern web-based book creation tool / V. O. Babenko, R. M. Yatsenko, P. D. Migunov at el // CEUR Workshop Proceedings, 2020. – С. 174–184.
10. Parfonov Y. E. Use of the Django framework in the web application for computer-based testing of higher education institution students / Y. E. Pafonov, Y. V. Zmiievskaya // Матеріали X-ої ювілейної міжнар. наук.-практ. конф. «Free and Open Source Software», 20 – 22 листоп. 2018 р. – Х.: ХНУБА, 2018. - С. 47

Інформаційні ресурси

11. Яценко Р.М. Персональна навчальна система з навчальної дисципліни «Web-технології» [Електронний ресурс]. – Режим доступу : <https://pns.hneu.edu.ua/course/view.php?id=7113>.
12. Mozilla Developer Network [Електронний ресурс]. – Режим доступу : <https://developer.mozilla.org/en-US/>.
13. W3Schools Online Web Tutorials Distribution [Електронний ресурс]. – Режим доступу : <https://www.w3schools.com/>.
14. Bootstrap – The most popular HTML, CSS, and JS library in the world [Електронний ресурс]. – Режим доступу : <https://getbootstrap.com/>.
15. Bulma: Free, open source, and modern CSS framework based on Flexbox [Електронний ресурс]. – Режим доступу : <https://bulma.io/>.
16. Сучасний підручник з JavaScript [Електронний ресурс]. – Режим доступу : <https://uk.javascript.info/>.
17. jQuery [Електронний ресурс]. – Режим доступу : <https://jquery.com/>.
18. Django: The web framework for perfectionists with deadlines [Електронний ресурс]. – Режим доступу : <https://www.djangoproject.com/>