

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕПАРТАМЕНТ ОСВІТИ І НАУКИ
ЗАПОРІЗЬКОЇ ОБЛАСНОЇ ДЕРЖАВНОЇ АДМІНІСТРАЦІЇ**

**КОМУНАЛЬНИЙ ЗАКЛАД «ЗАПОРІЗЬКИЙ ОБЛАСНИЙ ЦЕНТР
НАУКОВО-ТЕХНІЧНОЇ ТВОРЧОСТІ УЧНІВСЬКОЇ МОЛОДІ «ГРАНІ»
ЗАПОРІЗЬКОЇ ОБЛАСНОЇ РАДИ**

**Розробник
Бойчук Богдан Едуардович**

«WEB-ПРОГРАМУВАННЯ»

**Навчальна програма з позашкільної освіти
науково-технічного напрямку**

основний рівень – 3 роки навчання

м. Запоріжжя – 2025

Схвалено педагогічною радою комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради, протокол № 3 від 11.09.2025 року

Затверджено наказом комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради» від 11.09.2025 № 93

Укладач:

Бойчук Богдан Едуардович – керівник гуртка «WEB-програмування» комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради

Рецензенти:

Андрєєв Андрій Миколайович – доцент Запорізького національного університету, завідувач кафедри загальної та прикладної фізики, професор кафедри, доктор педагогічних наук

Смирнова Анна Василівна – керівник гуртка «Основи робототехніки та комп'ютерного моделювання» комунального закладу «Запорізький обласний центр науково-технічної творчості учнівської молоді «Грані» Запорізької обласної ради, керівник гуртка-методист

ПОЯСНЮВАЛЬНА ЗАПИСКА

У сучасному цифровому світі володіння навичками вебпрограмування відкриває широкі можливості для творчості, самовираження, навчання та кар'єрного зростання. Гурток «WEB-програмування» створено з метою ознайомлення вихованців 10–18 років з основами створення вебсайтів та вебзастосунків, формування ключових ІТ-компетентностей, розвитку логічного мислення, креативності та вміння працювати в команді.

Актуальність навчальної програми «WEB-програмування» полягає у тому, що WEB-програмування є однією з найбільш затребуваних компетентностей сучасного цифрового суспільства, що стрімко розвивається. Оволодіння навичками створення вебсайтів сприяє розвитку алгоритмічного мислення, креативності та цифрової грамотності, необхідних для успішної самореалізації молоді. Програма забезпечує умови для професійного самовизначення вихованців у сфері ІТ.

Дана навчальна програма реалізується у гуртку відділу STEM-освіти інформаційно-технічного профілю. Програма охоплює три роки навчання з поступовим ускладненням тем і завдань. У центрі уваги – практичне застосування знань, проєктна діяльність, гейміфікація освітнього процесу, підготовка до конкурсів та створення власного портфоліо.

Мета програми - формування у вихованців базових і поглиблених знань та навичок у сфері вебпрограмування, розвиток логічного мислення, алгоритмічної культури, творчого підходу до вирішення задач, а також здатності до самостійного навчання та командної роботи в ІТ-сфері.

Основні завдання полягають у формуванні таких **компетентностей**:

- *пізнавальної*, зміст якої полягає у засвоєнні основ HTML, CSS, JavaScript, розумінні структури та принципів функціонування вебресурсів, оволодінні базовими поняттями веброзробки, розвитку вміння шукати, аналізувати та критично оцінювати інформацію в цифровому середовищі, а також у формуванні стійкого інтересу до ІТ-сфери;

- *практичної*, зміст якої полягає в оволодінні навичками створення вебсторінок, їх стилізації, програмування інтерактивних елементів, використання сучасних інструментів розробки та публікації власних сайтів, а також у набутті досвіду застосування отриманих знань для розв'язання конкретних завдань і створення цифрових продуктів;

- *творчої*, зміст якої полягає у розвитку здатності до самовираження через цифрові проєкти, створенні авторського дизайну вебінтерфейсів, застосуванні креативного підходу до структури та оформлення сайтів, а також у формуванні ініціативності, естетичного смаку й інноваційного мислення;

- *соціальної*, зміст якої полягає у формуванні навичок ефективної комунікації, роботи в команді над спільними проектами, презентації результатів діяльності, дотримання цифрової етики, правил безпечної поведінки в Інтернеті, а також у розвитку відповідальності, самостійності та готовності до активної участі у цифровому суспільстві.

Навчальна програма «WEB-програмування» реалізується у гуртках науково-технічного напрямку інформаційно-технічного профілю. Навчальна програма розрахована на 3 роки основного рівня навчання для вихованців віком 10-18 років:

- основний рівень, перший рік навчання, 144 год. (4 год./тиждень);
- основний рівень, другий рік навчання, 144 год. (4 год./тиждень);
- основний рівень, третій рік навчання, 144 год. (4 год./тиждень).

Гуртківці навчаються основ веброзробки: від розмітки HTML і стилізації за допомогою CSS – до інтерактивної поведінки сторінки на JavaScript. Окрема увага приділяється системі контролю версій Git, платформі GitHub та командній роботі над проектами.

Навчання проводиться у форматі поєднання теорії та практики, із використанням сучасних цифрових інструментів, онлайн-платформ і середовищ для програмування. Програма передбачає індивідуальну, парну та групову роботу, інтерактивні завдання, вікторини, мініпроекти, ігрові механіки та участь у змаганнях.

Програма «WEB-програмування» сприяє формуванню позитивної мотивації до створення та реалізації своєї творчої особистості.

Поряд з груповими, колективними формами роботи проводиться індивідуальна робота з вихованцями відповідно до Положення про порядок організації індивідуальної та групової роботи в закладах позашкільної освіти (із змінами, внесеними згідно з Наказом МОН України № 1123 (з 1322-08) від 10.12.2008 р.). Особливо така робота є актуальною під час підготовки до конкурсів, змагань, фестивалів та інших масових заходів.

Кожен рік завершується створенням власного підсумкового проекту, який включає теоретичні знання та практичні вміння, набуті протягом навчання. Після завершення програми вихованці матимуть у своєму арсеналі кілька власних вебпроектів, досвід роботи в середовищах розробки та впевненість у подальшому навчанні або професійному зростанні у сфері інформаційних технологій.

Формою контролю за результативністю навчання є практичні, підсумкові роботи, участь у конкурсах, творчі звіти.

Навчання в гуртку не передбачає спеціальної підготовки та знань. Навчальний матеріал програми викладено за принципом доступності та відповідності його обсягу віковим особливостям вихованців.

Програма є орієнтовною. За необхідності керівник гуртка може внести до програми зміни, які не повинні впливати на загальний зміст навчальної програми та кількість навчальних годин. Незмінними мають залишатися мета, завдання і прогнозований результат освітньої діяльності.

Основний рівень, перший рік навчання НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН

<i>Тема</i>	<i>Кількість годин</i>		
	<i>теоретичні</i>	<i>практичні</i>	<i>усього</i>
Вступ	2	-	2
Розділ 1. Вступ до веб-програмування	5	5	10
1.1 Що таке веб: вебсайт, сервер, клієнт, браузер	1	1	2
1.2 Інструменти веброзробника	2	2	4
1.3 Структура вебпроєкту	2	2	4
Розділ 2. Основи HTML	16	24	40
2.1 Базові теги HTML	4	6	10
2.2 Зображення та таблиці	4	6	10
2.3 Форми	4	6	10
2.4 Структура та семантика HTML	4	6	10
Розділ 3. Основи CSS	16	24	40
3.1 Підключення CSS. Селектори та властивості	4	6	10
3.2 Оформлення тексту, кольори, фони	4	6	10
3.3 Розмітка блоків і позиціонування	4	6	10
3.4 Flexbox і простий макет	4	6	10

Розділ 4. Адаптивність та оформлення	10	18	28
4.1 CSS Grid	2	6	8
4.2 Медіазапити та адаптивний дизайн	4	6	10
4.3 Повторення та застосування стилів	4	6	10
Розділ 5. Робота з навчальними платформами	4	6	10
Розділ 6. Підсумковий проєкт	6	6	12
Підсумкове заняття	2	-	2
Разом	60	84	144

ЗМІСТ ПРОГРАМИ

Вступ (2 год.)

Теоретична частина: Ознайомлення з поняттям вебпрограмування, цілі курсу, знайомство з планом занять, безпека в Інтернеті. Мотиваційна презентація «Світ вебтехнологій». Вікторина «Що ти знаєш про сайти?». Обговорення очікувань учнів, знайомство у форматі інтерактивної дошки. Налаштування облікових записів на платформах Replit або Thimble.

Розділ 1. Вступ до веб-програмування (10 год.)

1.1 Що таке веб: вебсайт, сервер, клієнт, браузер (2 год.)

Теоретична частина. Принцип роботи Інтернету. Як завантажується сторінка. Пояснення взаємодії сервер-клієнт.

Практична частина. Вивчення структури існуючих сайтів за допомогою інструментів розробника браузера. Пошук тегів і стилів на прикладах.

1.2 Інструменти веброботика (4 год)

Теоретична частина. Огляд редакторів коду, хмарних платформ, браузерних інструментів. Презентація: «5 інструментів сучасного веброботика».

Практична частина. Реєстрація та перше знайомство з середовищем Replit або VS Code. Створення HTML-файлу.

1.3 Структура вебпроекту (4 год)

Теоретична частина. Файлова структура: index.html, css, images.

Важливість правильної організації.

Практична частина. Створення власної структури вебпроєкту. Знайомство з базовою навігацією по папках.

Розділ 2. Основи HTML (40 год)

2.1 Базові теги HTML (10 год)

Теоретична частина. Вступ до HTML, теги: html, head, body, заголовки, абзаци, посилання, списки.

Практична частина. Створення першої сторінки з заголовками, абзацами, списками та посиланнями.

2.2 Зображення та таблиці (10 год)

Теоретична частина. Теги img, table, tr, td. Формати зображень. Підписи до таблиць.

Практична частина. Вставлення зображень, створення простої таблиці з 2–3 стовпцями, таблиця з розкладом.

2.3 Форми (10 год)

Теоретична частина. Форми та елементи введення: input, textarea, select, button. Атрибути name, placeholder, value.

Практична частина. Створення анкети з використанням чекбоксів, радіокнопок, текстових полів.

2.4 Структура та семантика HTML (10 год)

Теоретична частина: Семантичні теги: header, nav, main, section, article, footer. SEO-оптимізація.

Практична частина: Побудова макету сайту з розділами, створення структури інформаційного сайту.

Розділ 3. Основи CSS (40 год)

3.1 Підключення CSS. Селектори та властивості (10 год)

Теоретична частина: Синтаксис CSS, види підключення стилів, селектори тегів, класів, id.

Практична частина: Підключення зовнішнього CSS, стилізація заголовків, абзців, кольорів фону.

3.2 Оформлення тексту, кольори, фони (10 год)

Теоретична частина: Властивості: color, font-size, background-color, text-align.

Практична частина: Створення кольорових блоків, банерів, стилізація шрифтів, заголовків, списків.

3.3 Розмітка блоків і позиціонування (10 год)

Теоретична частина: Поняття блоку. Властивості display, margin, padding, border. Блочна модель CSS.

Практична частина: Побудова блоків контенту, навігаційної панелі, структурування текстових секцій.

3.4 Flexbox і простий макет (10 год)

Теоретична частина: Контейнер, напрямок, вирівнювання, розподіл простору.

Практична частина: Верстка галереї, макету з кількома колонками, футером і шапкою.

Розділ 4. Адаптивність та оформлення (28 год)

4.1 CSS Grid (8 год)

Теоретична частина: Побудова сітки, властивості grid-template, grid-gap, grid-area.

Практична частина: Верстка сторінки у вигляді сітки: новинна стрічка, продуктовий каталог.

4.2 Медіазапити та адаптивний дизайн (10 год)

Теоретична частина: Принцип mobile-first, @media, адаптивність блоків.

Практична частина: Створення мобільної версії сайту. Перевірка через інструменти розробника.

4.3 Повторення та застосування стилів (10 год)

Теоретична частина: Помилки новачків, рекомендації щодо оформлення коду, структури стилів.

Практична частина: Створення власного шаблону сайту з використанням всіх знань.

Розділ 5. Робота з навчальними платформами (10 год)

Теоретична частина: Огляд платформ: Code.org, Sololearn, Grasshopper, особливості навчання.

Практична частина: Реєстрація, проходження вправ та коротких мінікурсів на вибір.

Розділ 6. Підсумковий проєкт (12 год)

Теоретична частина: Планування тематики сайту, етапи розробки, розподіл функцій (для командної роботи).

Практична частина: Створення сайту з нуля. Наповнення сторінок, оформлення, перевірка адаптивності.

Підсумкове заняття (2 год)

Теоретична частина. Презентація робіт, взаємооцінювання, демонстрація найкращих сайтів, рефлексія. Підведення підсумків роботи протягом першого року навчання у гуртку «WEB-програмування». Відзначення кращих вихованців гуртка.

ПРОГНОЗОВАНИЙ РЕЗУЛЬТАТ

Вихованці мають знати:

- що таке вебсайт, як він працює, роль сервера, клієнта, браузера;
- основні поняття HTML і CSS: теги, атрибути, властивості стилів;
- призначення базових HTML-елементів (заголовки, абзаци, посилання, зображення, таблиці, форми);
- основи структури вебсторінки: head, body, блокова та рядкова верстка;
- принципи CSS: селектори, кольори, фони, шрифти, позиціонування;
- концепції адаптивного дизайну, медіазапитів, flexbox та grid-сіток;
- правила створення структури вебпроєкту;
- призначення навчальних онлайн-платформ (Code.org, Sololearn, Grasshopper).

Вихованці мають вміти:

- створювати HTML-документи з використанням основних тегів;
- підключати та використовувати зовнішні й внутрішні CSS-стили;
- оформлювати текст, елементи та блоки за допомогою CSS;
- використовувати макети з Flexbox та CSS Grid для створення адаптивних сторінок;
- розміщувати контент у таблицях, формах та структурованих блоках;

- застосовувати стилі до елементів з урахуванням принципів зручності та читабельності;
- користуватись онлайн-редакторами коду (наприклад, Replit, Thimble);
- створювати прості шаблони сайту та працювати з файловою структурою вебпроєкту.

Вихованці мають набути досвід:

- реалізації особистого мініпроєкту: простого сайту з кількох сторінок;
- колективного та індивідуального планування контенту сайту;
- презентації власного вебпроєкту перед аудиторією;
- виконання інтерактивних вправ та завдань на навчальних платформах;
- застосування творчого підходу при створенні контенту та дизайну сторінки;
- аналізу структури та дизайну реальних сайтів;
- першої самостійної роботи в командному або індивідуальному мініпроєкті.

Основний рівень, другий рік навчання НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН

<i>Тема</i>	<i>Кількість годин</i>		
	<i>теоретичні</i>	<i>практичні</i>	<i>усього</i>
Вступ	2	-	2
Розділ 1. Основи JavaScript	16	24	40
1.1 Змінні, типи даних, оператори	4	6	10
1.2 Умовні конструкції (if, else)	4	6	10
1.3 Цикли та масиви	4	6	10
1.4 Функції та об'єкти	4	6	10
Розділ 2. Взаємодія JavaScript з HTML	16	24	40
2.1 DOM. Пошук елементів, зміна вмісту	4	6	10
2.2 Події користувача	4	6	10

2.3 Динамічне створення елементів	4	6	10
2.4 Робота з формами та localStorage	4	6	10
Розділ 3. Основи Git та GitHub	10	10	20
3.1 Вступ до Git. Команди Git	6	6	12
3.2 GitHub і хостинг	4	4	8
Розділ 4. Вебзастосунки	6	18	24
4.1 Динамічні елементи	2	6	8
4.2 Таймери та анімація	2	6	8
4.3 Застосування JS у реальних проєктах	2	6	8
Розділ 5. Підсумковий проєкт	8	8	16
Підсумкове заняття	2	-	2
Разом	60	84	144

ЗМІСТ ПРОГРАМИ

Вступ (2 год)

Теоретична частина. Повторення HTML та CSS. Вступ до програмування мовою JavaScript. Роль JS у фронтенд-розробці. Порівняння з іншими мовами. Інструменти для написання JS-коду. Ознайомлення з консоллю браузера. Перший скрипт "Hello, JavaScript!". Робота з онлайн-середовищами (Replit, JSFiddle).

Розділ 1. Основи JavaScript (40 год)

1.1 Змінні, типи даних, оператори (10 год)

Теоретична частина. Типи даних: string, number, boolean, undefined, null. Оператори: арифметичні, логічні, порівняння. Константи й змінні (var, let, const).

Практична частина. Створення простих скриптів: калькулятор, виведення інформації у консоль, форматування повідомлень.

1.2 Умовні конструкції (if, else) (10 год)

Теоретична частина. Логіка програм: умовні оператори if, else if, else. Порівняння значень. Булеві значення.

Практична частина. Скрипт "Визначення вікової категорії", створення форми з перевіркою введення.

1.3 Цикли та масиви (10 год)

Теоретична частина. Цикли for, while, do...while. Масиви, методи масивів (push, pop, shift, unshift).

Практична частина. Побудова таблиці множення, обхід масиву і виведення елементів, аналіз масиву оцінок.

1.4 Функції та об'єкти (10 год)

Теоретична частина. Оголошення функцій. Параметри, return. Основи об'єктів: ключ-значення, методи.

Практична частина. Створення калькулятора як функції. Опис об'єкта "учень": ім'я, вік, метод "привітання".

Розділ 2. Взаємодія JavaScript з HTML (40 год)

2.1 DOM. Пошук елементів, зміна вмісту (10 год)

Теоретична частина. Що таке DOM. Методи getElementById, querySelector. Зміна вмісту: innerHTML, textContent.

Практична частина. Створення сайту з динамічною зміною заголовка, вивід інформації в блоки сторінки.

2.2 Події користувача (10 год)

Теоретична частина. Події click, input, mouseover, keypress. Обробники подій.

Практична частина. Кнопка, яка змінює текст або колір фону. Реакція на введення в текстове поле.

2.3 Динамічне створення елементів (10 год)

Теоретична частина. Створення елементів: createElement, appendChild. Робота з класами, стилями.

Практична частина. Генератор карток товарів, кнопка для додавання нових блоків на сторінку.

2.4 Робота з формами та localStorage (10 год)

Теоретична частина. Отримання даних з форм, валідація. Збереження даних у браузері: localStorage.

Практична частина. Створення форми опитування з перевіркою. Збереження і відновлення введених даних.

Розділ 3. Основи Git та GitHub (20 год)

3.1 Вступ до Git. Команди Git (12 год)

Теоретична частина. Що таке система контролю версій. Основні команди: git init, status, add, commit, log.

Практична частина. Ініціалізація репозиторію, створення першого коміту. Тренажер LearnGitBranching.

3.2 GitHub і хостинг (8 год)

Теоретична частина. Що таке GitHub. Створення акаунта, репозиторію. Основи pull request, issue.

Практична частина. Розміщення власного сайту на GitHub Pages. Робота з візуальним інтерфейсом GitHub.

Розділ 4. Вебзастосунки (24 год)

4.1 Динамічні елементи (8 год)

Теоретична частина. Робота з кнопками, чекбоксами, генераторами вмісту. Події та функції зворотного виклику.

Практична частина. Створення "To-Do List" зі зміною стану елементів. Видалення, додавання задач.

4.2 Таймери та анімація (8 год)

Теоретична частина. setTimeout, setInterval, основи CSS-анімації з JS. Керування часом.

Практична частина. Таймер відліку. Ігрова механіка "Встигни натиснути кнопку".

4.3 Застосування JS у реальних проєктах (8 год)

Теоретична частина. Приклади проєктів: вікторини, калькулятори, таймери. Структура HTML/JS для невеликих застосунків.

Практична частина. Реалізація гри "Вгадай число", калькулятора або простого планера.

Розділ 5. Підсумковий проєкт (16 год)

Теоретична частина. Вибір ідеї. Обговорення структури. Планування функціоналу. UX/UI для користувача.

Практична частина. Створення HTML/CSS-шаблону. Додавання JS-функцій. Тестування. Розміщення на GitHub Pages. Створення README.md.

Підсумкове заняття (2 год)

Презентація проєктів. Демонстрація навичок. Збір відгуків, самооцінка. Онлайн-вікторина «Що я знаю про JavaScript?». Підведення підсумків роботи протягом другого року навчання у гуртку «WEB-програмування». Відзначення кращих вихованців гуртка.

ПРОГНОЗОВАНИЙ РЕЗУЛЬТАТ

Вихованці мають знати:

- принципи роботи мови JavaScript, її синтаксис, ключові конструкції та особливості;
- що таке змінні, типи даних, оператори, функції, об'єкти, масиви та як їх застосовують у програмуванні;
- основи DOM (Document Object Model) і способи взаємодії JavaScript із HTML-структурою сторінки;
- роль подій у динаміці сторінки та принцип обробки подій користувача;
- призначення та структуру системи контролю версій Git, основи роботи з GitHub;
- як JavaScript використовується у створенні простих вебзастосунків (todo, таймер, вікторина);
- основи UX/UI планування вебзастосунків.

Вихованці мають вміти:

- оголошувати змінні, писати умовні оператори, цикли, створювати та викликати функції;
- створювати й використовувати об'єкти, масиви та базові методи роботи з ними;
- шукати HTML-елементи на сторінці за допомогою JavaScript і змінювати їхній вміст;
- реагувати на дії користувача через події (клік, введення тексту тощо);
- створювати нові HTML-елементи динамічно та керувати їхніми класами/стилями;
- виконувати основні Git-команди, створювати репозиторії, робити коміти;
- публікувати вебсторінки на GitHub Pages;

- створювати динамічні вебінтерфейси, як-от списки справ, форми збереження, таймери;
- інтегрувати отримані знання для розробки власного міні-проєкту.

Вихованці мають набути досвід:

- самостійного написання коду на JavaScript для вирішення прикладних завдань;
- роботи в парі або команді над спільними вебзастосунками (на рівні проєкту);
- використання системи Git для збереження й версійності коду;
- презентації власного результату (публікація та демонстрація підсумкового проєкту);
- аналізу та налагодження помилок (debugging) у середовищі браузера;
- публікації результатів на публічних платформах (GitHub, GitHub Pages);
- візуального й логічного структурування коду, написання зрозумілого README.md.

Основний рівень, третій рік навчання НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН

Тема	Кількість годин		
	теоретичні	практичні	усього
Вступ	2	-	2
Розділ 1. Проєктна діяльність: реальні задачі	12	60	72
1.1 Персональне портфоліо (сайт-візитка)	2	10	12
1.2 Онлайн-щоденник (з localStorage)	2	10	12
1.3 Сайт-енциклопедія про хобі або гру	2	10	12
1.4 Вебсторінка для виховного заходу або групи	2	10	12
1.5 Мінігра «Вгадай столицю» або таймер-клікер	2	10	12
1.6 Проєкт «Генератор порад або жартів»	2	10	12

Розділ 2. Гейміфіковане навчання та змагання	10	20	30
Розділ 3. Підготовка до конкурсів	10	16	26
Розділ 4. Завершальний підсумковий проєкт	2	10	12
Підсумкове заняття	2	-	2
Разом	38	106	144

ЗМІСТ ПРОГРАМИ

Вступ (2 год)

Теоретична частина. Повторення основ HTML, CSS, JavaScript. Ознайомлення з форматами проєктів третього року. Розбір успішних прикладів сайтів. Підготовка до самостійної роботи та визначення інтересів. Аналіз структури прикладного сайту. Завантаження шаблону та налаштування середовища. Вибір першої ідеї.

Розділ 1. Проєктна діяльність: реальні задачі (72 год)

1.1 Персональне портфоліо (сайт-візитка) (12 год)

Теоретична частина. Структура персонального сайту: сторінка про себе, навички, контакти. Рекомендації щодо дизайну. SEO-базис.

Практична частина. Верстка шаблону, додавання особистої інформації. Адаптація під різні екрани. Публікація на GitHub Pages.

1.2 Онлайн-щоденник (з localStorage) (12 год)

Теоретична частина. Принцип збереження інформації в браузері. UX-орієнтований інтерфейс для записів.

Практична частина. Створення форми запису думок. Збереження та відображення записів з localStorage. Тематичне оформлення.

1.3 Сайт-енциклопедія про хобі або гру (12 год)

Теоретична частина. Каталогізація контенту. Внутрішня навігація, табличні блоки, сортування.

Практична частина. Заповнення розділів сайту про вибране хобі (ігри, спорт, мистецтво). Інтерактивні блоки.

1.4 Вебсторінка для виховного заходу або групи (12 год)

Теоретична частина. Структура презентаційного сайту, афіші, фотогалереї, зворотній зв'язок.

Практична частина. Верстка афіші, слайдер фото, форма реєстрації або коментарів.

1.5 Мінігра «Вгадай столицю» або таймер-клікер (12 год)

Теоретична частина. Основи логіки гри. Стани гри, обробка подій, таймер.

Практична частина. Реалізація JS-гри з балами. Випадкове запитання, відповіді, лічильник.

1.6 Проєкт «Генератор порад або жартів» (12 год)

Теоретична частина. Робота з масивами, випадковими числами, JS-шаблонізація.

Практична частина. Вибір теми (жарти, поради, мотивація). Рандомізоване виведення з масиву на сторінку. Дизайн та UX.

Розділ 2. Гейміфіковане навчання та змагання (30 год)

Теоретична частина. Моделі навчальних ігор, рівні складності, оцінювання. Принцип змагань: чесність, креативність, логіка.

Практична частина. Робота з платформами: CodeCombat, Blockly Games, ScratchJS. Проведення турніру за авторськими мініпроєктами (вікторини, клікери, таймери).

Розділ 3. Підготовка до конкурсів (26 год)

Теоретична частина. Вимоги до конкурсів: юні веброзробники, Technovation, STEM-олімпіади. Презентація, технічне завдання, критерії оцінювання.

Практична частина. Вибір проєкту для участі, оформлення портфоліо, тестування, підготовка презентацій. Репетиції захисту.

Розділ 4. Завершальний підсумковий проєкт (12 год)

Теоретична частина. Повторення тем. Побудова структури, обговорення ідеї, вибір інструментів.

Практична частина. Створення власного проєкту (сайт, гра, інструмент) з HTML/CSS/JS. Використання GitHub. Написання документації.

Підсумкове заняття (2 год)

Теоретична частина. Презентація підсумкових проєктів. Демонстрація. Вікторина на знання тем курсу. Вручення сертифікатів. Онлайн-альбом проєктів групи.

ПРОГНОЗОВАНИЙ РЕЗУЛЬТАТ

Вихованці мають знати:

- принципи побудови повноцінного вебзастосунку (структура HTML, стилізація CSS, логіка на JavaScript);
- поняття UI/UX та основи дизайну для користувача;
- поняття компонентності та модульності у веброзробці;
- значення командної розробки, ролі в команді, базові практики Agile/спринтів;
- важливість і способи валідації даних, зберігання інформації у localStorage;
- основи тестування вебзастосунків і виправлення помилок;
- як оформити проєкт для конкурсу чи портфоліо (структура репозиторію, README, документація);
- роль гейміфікації та інтерактивності у навчальних і розважальних вебпродуктах.

Вихованці мають уміти:

- створювати складніші інтерактивні вебсторінки з HTML/CSS/JS;
- реалізовувати зберігання та відновлення даних за допомогою localStorage;
- додавати інтерактивні елементи: кнопки, форми, таймери, мініігри;
- писати функціональні скрипти для обробки подій, зміни DOM, запуску анімацій;
- працювати з Git та GitHub для командного проєкту (створення гілок, комітів, пул-реквестів);
- планувати, реалізовувати й захищати власний проєкт: від ідеї до хостингу;
- брати участь у конкурсах і хакатонах з веброзробки;
- використовувати онлайн-інструменти для колаборації (Trello, Figma, Google Docs).

Вихованці мають набути досвід:

- реалізації повного циклу розробки вебзастосунка (аналіз задачі, планування, дизайн, кодування, тестування, презентація);
- самостійної і командної роботи над вебпроектами з реальними задачами;
- створення портфоліо (сайт-візитка, блог, тематичні енциклопедії);
- гейміфікації функціоналу для покращення користувацького досвіду;
- виступів перед аудиторією (презентація підсумкового проєкту);
- самооцінки і рефлексії власної роботи;
- участі у змаганнях з веброзробки та презентації результатів зовнішнім експертам.

ЛІТЕРАТУРА

1. Браун, М. JavaScript. Детальний посібник для початківців. – К.: Видавництво «Діалектика», 2021.
2. Брая, Дж. HTML, CSS і JavaScript. Повний курс. – К.: Наш Формат, 2020.
3. Єлізаров, М. Git для новачків. – Х.: Фабула, 2022.
4. Ільїн, Д. Основи фронтенд-розробки. – Львів: Левус, 2021.
5. Коваль, О. Основи вебдизайну. HTML + CSS. – Тернопіль: Навчальна книга, 2020.
6. Крок, Т. Основи розробки вебсайтів. Шкільний курс. – Київ: Педагогічна думка, 2023.
7. Лафоре, Р. Структури даних і алгоритми в JavaScript. – К.: Видавництво «Міленіум», 2020.
8. Литвин, Ю. HTML5 і CSS3. Створення сучасних сайтів. – Х.: Фоліо, 2021.
9. Малюк, І. Вебпрограмування. Практичний посібник для школярів. – Київ: Освіта України, 2022.
10. Матвієнко, С. Проєктна діяльність у веброзробці. – Львів: Світ, 2023.
11. Поліщук, А. Git та GitHub для новачків. Інтерактивний посібник. – Київ: IT-Book, 2021.
12. Резнік, І. Frontend. Шлях від нуля до першого сайту. – Дніпро: Навчальна Лабораторія, 2021.
13. Романюк, Т. Алгоритмізація в JavaScript. – Тернопіль: Навчальні технології, 2022.
14. Фріман, Е., Робсон, Е. HTML і CSS. Розробка сайтів у прикладах. – Харків: Видавництво «Прометей», 2020.

15. Шеферд, Д. GitHub для початківців. Покроковий гід. – Київ: SoftPress, 2023.
16. Щербина, М. Курс JavaScript для учнів. – Одеса: Ранок, 2023.

ІНТЕРНЕТ-РЕСУРСИ

1. Codecademy — курси з веброзробки (англійською) <https://www.codecademy.com>.
2. Code.org — платформа для вивчення основ програмування в ігровій формі <https://code.org>.
3. CodeSandbox — продвинуте середовище для розробки вебпроектів <https://codesandbox.io>.
4. Codewars — платформа для тренування JavaScript через задачі <https://codewars.com>.
5. CSS Grid Garden — гра для вивчення CSS Grid <https://cssgridgarden.com>.
6. DevChallenges — вебзавдання й мініпрактики для реальних ситуацій <https://devchallenges.io>.
7. Flexbox Froggy — гра для вивчення CSS Flexbox <https://flexboxfroggy.com>.
8. freeCodeCamp — повноцінні курси з HTML, CSS, JS, React <https://www.freecodecamp.org>.
9. Frontend Mentor — практичні завдання для верстки та JS <https://frontendmentor.io>.
10. Git — офіційний сайт системи контролю версій Git <https://git-scm.com>.
11. GitHub — платформа для зберігання коду та спільної роботи над проектами <https://github.com>.
12. GitHub Pages — безкоштовний хостинг для сайтів (GitHub Pages) <https://pages.github.com>.
13. Glitch — створення й хостинг невеликих вебзастосунків онлайн <https://glitch.com>.
14. HTML-CSS-JS.com — прості генератори HTML, CSS та JS-коду <https://html-css-js.com>.
15. JSFiddle — середовище для швидкої перевірки HTML/CSS/JS <https://jsfiddle.net>.
16. MDN Web Docs — документація з HTML, CSS, JavaScript українською <https://developer.mozilla.org/uk/>.

17. Replit — онлайн-середовище для написання й запуску коду (HTML, CSS, JS) <https://replit.com>.
18. W3Schools — інтерактивні підручники та онлайн-редактор коду <https://www.w3schools.com>.
19. Web.dev — навчальні матеріали від Google про сучасну веброзробку <https://web.dev>.