

Короткий виклад

Урок-проект «Сайт БТІ» передбачає створення простого функціонального вебсайту для бюро технічної інвентаризації. Це практичне завдання для стажера, який у процесі роботи навчиться планувати проект по модулям, розробляти структуру сайту, верстати сторінки, реалізовувати контактну форму, налаштовувати хостинг і домен, оптимізувати сайт для пошуку та забезпечувати його доступність. Нижче подано поетапний план робіт із чітким ТЗ, прикладами текстів, діаграмами та чек-листами для тестування і приймання проекту.

План робіт по модулях

Нижче – орієнтовний розклад завдань на два тижні (4 повні робочі дні), з дедлайнами та очікуваними артефактами:

```
gantt
  dateFormat YYYY-MM-DD
  title План реалізації урок-проекту «Сайт БТІ»
  section Підготовчий модуль
  Збір контенту та аналіз БТІ :done, a1, 2026-02-19, 1d
  Проектування структури:done, a2, 2026-02-20, 1d
  section Розробка фронтенду
  Верстка головної та внутрішніх сторінок: active, b1, 2026-02-21, 2d
  Форма звернення (HTML/CSS/JS): b2, 2026-02-23, 1d
  section Бекенд і публікація
  Налаштування хостингу і домену: c1, 2026-02-24, 1d
  Оптимізація (SEO, favicon, 404): c2, 2026-02-25, 1d
  section Перевірка і завершення
  Тестування і мобільна адаптація: d1, 2026-02-26, 1d
  Підготовка адмінінструкції та бекапу: d2, 2026-02-27, 1d
```

- **Модуль 1:** Збір інформації про функції та послуги БТІ, аналіз подібних сайтів. Завершити – контент-план (аркуш із розподілом текстів по сторінках).
- **Модуль 2:** Проектування сайту. Оформити схему сайту (sitemap) і прототипи основних сторінок.
- **Модуль 3:** Верстати головну, «Про БТІ», «Послуги», «FAQ», «Контакти», розмістити приклади текстів. Створити форму звернення (з полями «Ім'я», «Email», «Повідомлення») з базовою валідацією.
- **Модуль 4:** Налаштувати хостинг і домен; наповнити метатеги (title, description) для SEO; додати favicon, сторінку 404, політику конфіденційності.
- **Модуль 5:** Перевірити мобільну адаптивність, доступність (ARIA, перевірка клавіатурою), швидкість (PageSpeed Insight); безпеку форми (CAPTCHA, перевірка даних). Виправити помилки. Здати остаточні артефакти.

Технічне завдання

1. **Структура сайту:** сторінки «Головна», «Про БТІ», «Послуги», «Завантаження документів», «FAQ», «Контакти». На «Головній» – коротке привітання, новини/оголошення БТІ, навігація. На «Про БТІ» – опис організації. На «Послуги» – перелік та опис основних послуг. На «Завантаженнях» – файли з бланками (PDF) для завантаження. На «FAQ» – відповіді на типові запитання. На «Контакти» – адреса, тел/факс, карта, форма звернення.
2. **Поле форми:** «Ім'я» (text, required), «Email» (email, required), «Повідомлення» (textarea, required). Валідація HTML5 (атрибут `required`, тип `email`). Псевдо-відправка: обробити `onsubmit`, наприклад, через JavaScript `alert('Повідомлення надіслано'); return false`.
3. **SEO (метатеги):** В `<head>` кожної сторінки: `<title>Сайт БТІ – [Назва сторінки]</title>` та `<meta name="description" content="Короткий опис сторінки з ключовими словами">`. Як пояснює MDN, вказання опису з релевантними ключовими словами «може зробити сторінку вищою в результатах пошуку» ¹. Структуру URL робити простими (напр., `/pro-bti`, `/poslugi`).
4. **Favicon:** Додати в `<head>` тег `<link rel="icon" href="favicon.ico">`. Зазвичай це невелике 16×16/32×32 зображення. MDN радить, що іконка відображається в закладках браузера і вкладках ².
5. **Сторінка 404:** Зробити лаконічну 404-сторінку («Сторінку не знайдено»), по можливості із посиланням на Головну. Зауважимо, що кількість помилок 404 варто мінімізувати для кращого UX ³. Додайте дружнє повідомлення – це дозволено, але не плутати користувача ⁴.
6. **Політика конфіденційності:** Описати, як обробляються дані з форми звернення. Вказати, що дані з форми використовуються лише для зв'язку з громадянами, не передаються третім особам. Форму скласти на основі типової Згоди (GDPR), де зазначити відповідальність.
7. **Приклад Sitemap (структура):**

```
graph LR
  A[Головна] --> B[Про БТІ]
  A --> C[Послуги]
  A --> D[Завантаження]
  A --> E[FAQ]
  A --> F[Контакти]
  C --> C1[Інвентаризація]
  C --> C2[Техпаспорт]
  F --> F1[Форма звернення]
```

Це наочно ілюструє зв'язки між сторінками.

Інструкції по контенту

- **Текст «Про БТІ»:** Короткий опис ролі підприємства. Наприклад:
«Підприємство «Бюро технічної інвентаризації» здійснює інвентаризацію нерухомості, виготовлення технічних паспортів та надає послуги з легалізації перепланувань. При наданні послуг БТІ враховує всі технічні та юридичні чинники... Вирішуємо будь-які питання у сфері нерухомості у найкоротші терміни за розумними цінами.» ⁵.
- **Послуги:** Перелік основних послуг. Наприклад:
- **Інформаційна довідка з БТІ (вихідні дані про майно)**

- Технічна інвентаризація (виготовлення технічних паспортів на квартиру, будинок тощо)
- Легалізація перепланування (оформлення дозвільної документації)
- Реєстрація права власності (консультації з документів та реєстраційних процедур)
- Фотокопії документів та довідки.
- **FAQ (Питання-Відповіді):** Формулюйте реальні питання клієнтів і коротко відповідайте.
Наприклад:
- Чи можна отримати дублікат технічного паспорта? – Ні. Після виготовлення техпаспорт повертається замовнику і в БТІ оригінал не зберігається. Ви можете замовити нову технічну інвентаризацію та отримати новий паспорт, який відобразить фактичний стан об'єкта на дату інвентаризації ⁶.
- Які документи потрібні для замовлення? – Паспорт, довідка про власність на квартиру (договір, свідоцтво), заява на інвентаризацію.
- **Контакти:** Напишіть повні контактні дані. Наприклад:

Контакти:
 Комунальне підприємство «Бюро технічної інвентаризації»
 Адреса: м. Київ, вул. Антоновича, 8, офіс 1
 Телефон: (044) 278-59-43, (044) 278-61-53
 Email: info@bti.kyiv.ua

Приклад подано на офіційному сайті: «Київ, вул. Антоновича, 8, оф.1; 044 233 16 80; info@bti.ua» ⁷.

- **Файли для завантаження:** Додайте посилання на PDF-форми заяв (заява на інвентаризацію, опитувальний лист тощо). Розмістіть їх на сторінці «Завантаження».



Наочний приклад wireframe дизайну майбутніх сторінок (виконано від руки на планшети).

Варіанти реалізації

1. **WordPress:**
2. Створити сайт на WordPress (локально або на хостингу). Вибрати просту тему (наприклад, «Twenty Twenty-One»).

3. Налаштувати сторінки за допомогою редактора (Gutenberg) або конструктором сторінок (Elementor, Gutenberg блоки).
4. Для форуму звернення можна використати контактний плагін (Contact Form 7) або вставити свій HTML/JS-код у сторінку (гутенберг-блок «HTML»).
5. *Переваги:* легке управління контентом через адмінку, велика кількість готових плагінів і тем ⁸. *Недоліки:* потребує хостинг PHP/MySQL, вища складність налаштувань, потребує оновлень.

6. Google Sites:

7. Заходимо в Google Sites (sites.google.com), створюємо новий сайт.
8. Вибираємо одну з шаблонів, додаємо сторінки відповідно до структури.
9. Вставляємо тексти, зображення, форми (Google Forms) для звернення.
10. *Плюси:* дуже простий інтерфейс, нічого складного не потрібно програмувати, безкоштовний хостинг від Google. *Мінуси:* обмежена кастомізація дизайну, погані SEO-можливості ⁹, немає гнучких плагінів.

11. Статичний сайт (HTML/CSS/JS):

12. Створити HTML-сторінки вручну або використати шаблон. Можна застосувати легкий генератор сайтів (Jekyll, Hugo) та публікувати на Netlify/GitHub Pages.
13. Верстати шаблони за допомогою VSCode. Для форми – JavaScript для відправки/аліт (псевдо-відправка).
14. *Хостинг:* підійде будь-який простий хостинг (до 1 ГБ, 128–256 МБ RAM). Для Netlify – достатньо Git-репозиторію, а домен прямує через DNS.
15. *Домен:* купити в реєстратора (наприклад, nic.ua) і вказати А-запис на хостинг (IP) або CNAME (напр., на Netlify). Google Sites і WordPress.com теж дозволяють прив'язувати власні домени (див. інструкції Google і WordPress).

16. GitHub Pages/Netlify:

17. Завантажити код сайту в репозиторій GitHub. Увімкнути GitHub Pages (в налаштуваннях репозиторію).
18. Або створити аккаунт на Netlify, під'єднати репозиторій – сайт згенерується автоматично.
19. Мінімальні вимоги: стабільне інтернет-з'єднання, DNS доступ, SSL сертифікат (безкоштовно автоматично). Докладні інструкції для Netlify наводяться у офіційній документації (див. [Netlify Docs](#) та Google).



Приклад робочої дошки з нотатками та схемами завдань (на етапі планування). За аналогією слід накреслити графік робіт та розподіл ресурсів.

Чек-лист приймання і тестування

- **Адаптивність (Mobile-Friendly):** Перевірити сайт на різних розмірах екранів (Viewport) або за допомогою інструментів розробника в браузері. Елементи повинні коректно масштабуватися (через CSS `@media`).
- **Доступність:** Кожне поле форми має `<label>` або `aria-label`. Фото/іконки – з атрибутом `alt`. Форма повинна бути зручною для навігації з клавіатури.
- **Перевірка HTML/CSS/JS:** Прогнати сторінки через [W3C Validator](#) для HTML і CSS, усунути критичні помилки. JS-код – перевірити на консольні помилки.
- **SEO:** Окрім title/description, переконаватися, що сторінки мають єдині заголовки `<h1>` ¹⁰, що поля «title» та «h1» різні. Можна використовувати валідатори [rich-results](#) чи Google PageSpeed Insights для загальної оцінки.
- **Швидкість завантаження:** Оптимізувати зображення (WebP/JPEG), кешувати ресурси. Перевірити PageSpeed Insights.
- **Безпека форми:** Додати просту перевірку в JavaScript (на наявність тексту в required полях), пошук вразливих місць (якщо сайт на PHP – уникати SQL-ін'єкцій). Бажано встановити CAPTCHA (якщо використовуємо сторонній сервіс чи плагін).
- **Перевірка 404:** Перейти за неіснуючою URL і переконаватися, що відображається своя 404-сторінка. Картинка/текст зрозумілі, є посилання на головну.

Інструкція адміну (як керувати сайтом)

- **Редагування контенту:** Якщо WordPress – зайти в `example.com/wp-admin`, редагувати пости/сторінки через візуальний редактор або блоки. Для Google Sites – просто відкриваєте сайт в редакторі Google ([sites.google.com](#)) та вносите зміни. Для статичного сайту – змінювати файли HTML/CSS у проекті (локально або через FileManager на хостингу) і повторно публікувати.
- **Завантаження файлів:** Якщо додано розділ «Завантаження», за потреби оновлюйте PDF-шаблони у відповідній папці. У WP можна оновлювати через Медіатеку або FPT. Важливо зберігати структуру посилань.

- **Резервне копіювання:** Регулярно робіть бекап (завантажуйте копію сайту): у WP – через плагін або вручну (експортувати БД та файли `/wp-content/`), для статичного сайту – просто резервна копія всіх файлів і налаштувань хостингу. Раз на місяць рекомендується зберігати копію окремо від основного диску.
- **Створення обліків:** Для безпеки адміністрування зробіть окремі облікові записи (WP) для кожного відповідального. Не працюйте під адміністративним акаунтом.
- **Логування змін:** Фіксуйте (наприклад, у Google Docs) що і коли змінювали. Якщо щось пішло не так, легше відкотити зміни.

Навчальні завдання та контрольні точки

Стажер звітує щодня по модулях. До кожного дедлайну потрібно здати артефакт та відповісти на запитання керівника:

- **День 1 (Збір контенту):** Скласти короткий план сайту та зібрати інформацію. *Артефакт:* документ «План сайту і наповнення» (опис сторінок, ключові пункти). **Питання:** Чому ви обрали таку структуру сторінок?
- **День 2 (Прототип і дизайн):** Намалювати схемний макет (wireframe) головної і внутрішніх сторінок. *Артефакт:* зображення/скрін прототипів (можна на папері, фото). **Питання:** Які блоки будуть на кожній сторінці, навіщо вони потрібні?
- **День 3 (Верстка):** Верстати HTML/CSS “скелет” сайту, включно з меню, заголовками, блоками послуг. *Артефакт:* zip-архів з кодом (index.html, style.css). **Питання:** Чому ви використали саме ці теги/класи?
- **День 4 (Контактна форма):** Додати сторінку «Контакти» з формою. Реалізувати просту валідацію (required). *Артефакт:* оновлений код (zip або Git), плюс скрипт форми. **Питання:** Як здійснюється відправка форми?
- **День 5 (SEO і публікація):** Заповнити метатеги (`<title>`, `<meta>`), favicon, опублікувати сайт на хостингу/Google Sites/Netlify. *Артефакт:* посилання на робочий сайт (або сторінку Google Sites). **Питання:** Як ви налаштували домен і SSL?
- **День 6 (Перевірка):** Перевірити сайт за чек-листом: адаптивність, швидкість, безпека форми. *Артефакт:* звіт у Google Docs («результати тестування»). **Питання:** Які недоліки ви виправили?
- **День 7 (Адмінінструкція):** Описати (в документі) як адміністратор змінює контент і робить бекап. *Артефакт:* документ «Адмінінструкція сайту». **Питання:** Де зберігається резервна копія?
- **День 8 (Презентація):** Підготувати коротку презентацію зробленого: посилання на сайт, опис структури та функцій.

Кожен день стажер прикріплює до звіту лінк/архів результатів та відповіді на запитання. Це полегшить перевірку етапів.

Корисні інструменти і шаблони

- **Документи:** Google Docs/Sheets – для нотаток, плану проєкту, таблиць завдань.
- **Діаграми:** Figma (або Draw.io) – для креслення sitemap, прототипів. Для timeline/таблиць можна використовувати Google Таблиці.
- **Редактор коду:** VSCode (безкоштовний); чи будь-який з підсвіткою синтаксису.
- **Контроль версій:** Git/GitHub – завести репозиторій, щоб зберігати код і бачити історію змін.
- **FTP/файловий менеджер:** якщо хостинг надає FTP, можна оновлювати файли через FileZilla або панель хостингу.

- **Плагіни:** Для WordPress – Contact Form 7, Yoast SEO (для оптимізації title/description), інші за потреби.
- **Інші:** бібліотека іконок (FontAwesome), Google Fonts (для нестандартних шрифтів), онлайн-валідатори (W3C, PageSpeed).

Ризики та заборони

- **Не давайте складні завдання:** Старайтесь не перевантажувати стажера «нетехнічними» завданнями (напр. розглянути сотню договорів) або занадто складними («налаштувати сервер на Debian»). Уникайте роботи з критичною інформацією клієнтів (постороннім доступ до БД).
- **Контроль:** Не покидайте стажера без нагляду під час роботи з сервером або базами. Не просіть одразу вносити правки в робочий сайт органу влади – спочатку нехай тестує на «пісочниці» (локальній копії або тестовому домені).
- **Загальний доступ:** Не давайте логінів від свого акаунту хмарних сервісів – створіть окремий обліковий запис. Уникайте злиття змісту сторінок – нехай кожен робить свій компонент.
- **Технічна безпека:** Забороняється зберігати паролі в кодї, використовувати нешифровані HTTP (лише HTTPS). Не залишати відкритими веб-кабінети з базою (phpMyAdmin) без пароля.

Приклади команд і коду

- **HTML-код контактної форми:**

```
<form id="contact-form" onsubmit="sendForm(); return false;">
  <label>Ім'я:<br><input type="text" name="name" required></label><br>
  <label>Email:<br><input type="email" name="email" required></label><br>
  <label>Повідомлення:<br><textarea name="message" required></textarea></label><br>
  <button type="submit">Відправити</button>
</form>
<script>
  function sendForm() {
    alert('Ваше звернення надіслано!');
  }
</script>
```

Ми додаємо атрибут `required`, як рекомендує MDN – він вимагає заповнення поля перед відправкою ¹¹.

- **Мета-теги у `<head>`:**

```
<title>БТІ – Головна</title>
<meta name="description" content="Офіційний сайт Бюро технічної
інвентаризації. Дізнайтеся про послуги: техпаспорт, інвентаризацію,
легалізацію перепланувань.">
```

Як говорить MDN, `<title>` встановлює заголовок документа (відображається у вкладці браузера), а `<h1>` використовується тільки у тілі сторінки ¹⁰.

- **Example sitemap.xml:**

```
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url><loc>https://bti.example.com/</loc><priority>1.0</priority></url>
  <url><loc>https://bti.example.com/pro-bti</loc><priority>0.8</priority></url>
  <url><loc>https://bti.example.com/poslugi</loc><priority>0.8</priority></url>
  <url><loc>https://bti.example.com/contacts</loc><priority>0.8</priority></url>
</urlset>
```

- **Приклад перевірки сайту:** `curl -Is https://bti.example.com | head -n 1` – ця команда повертає статус HTTP; якщо на початку рядка є `HTTP/1.1 200 OK`, значить сайт доступний. Подібний підхід використовують у скриптах моніторингу, як радять інженери ¹².

- **CSS (адаптивність):**

```
@media (max-width: 600px) {
  nav { flex-direction: column; }
  /* перехід меню в стовпчик на мобільних */
}
```

- **robots.txt (опційно):** Заборонити індексацію службових сторінок, якщо потрібно:

```
User-agent: *
Disallow: /admin/
```

Таблиці та розклад

День/ Модуль	Завдання	Артефакти	Відповідальний
1-2	Збір контенту, план сайту	План сайту (документ)	Стажер
3	Прототипування сторінок	Скетч/малюнок wireframe	Стажер
4-5	Верстка HTML/CSS	Код сайту (index.html, style.css)	Стажер
6	Додати форму звернення	Оновлений код з формою	Стажер
7-8	SEO, фавікон, 404, домен	Налаштування хостингу, прототип 404	Стажер
9	Тестування, виправлення	Звіт про перевірки	Стажер
10	Адмін-інструкція, презентація	Документи (user manual, презентація)	Стажер

Окремою таблицею можна показати відповідальності, але в цьому проєкті – відповідальний завжди стажер під керівництвом наставника.

Зразки текстів (українською)

- **Про БТІ:** «Наше Бюро технічної інвентаризації — комунальне підприємство при міській раді. Ми здійснюємо технічну інвентаризацію об'єктів нерухомості і виготовляємо технічні паспорти на квартири, будинки, гаражі тощо. При наданні послуг БТІ враховує всі технічні та юридичні чинники, стоїть на боці замовника. Вирішуємо будь-які питання у сфері нерухомості у найкоротші терміни за розумними цінами» ⁵.
- **Послуги:** «Перелік наших послуг: **Інформаційна довідка з БТІ** (оформлення довідок про об'єкти нерухомості); **Технічний паспорт** (виготовлення техпаспортів на житлові та нежитлові приміщення); **Легалізація перепланувань** (пакет документів для узаконення перепланування); **Реєстрація прав** (консультації з реєстрації майнових прав); **Консультації та обстеження** (технічні обстеження, висновки технічної можливості поділу, тощо)».
- **FAQ (питання-відповіді):**
- **Питання:** «Чи можна отримати дублікат техпаспорта?»
Відповідь: «Ні. Відповідно до правил, після виготовлення технічний паспорт видається замовнику, і оригінал у БТІ не зберігається. Ви можете замовити інвентаризацію заново та отримати новий техпаспорт, який відобразить фактичний стан об'єкта на даний момент» ⁶.
- **Питання:** «Які документи потрібні для оформлення?»
Відповідь: «Паспорт громадянина України (або інший ID), довідка про приватизацію/договір купівлі-продажу або свідоцтво про право власності на нерухомість, а також заповнена заява (можна завантажити з сайту)».
- **Контакти:** Як показано раніше, адреса вказана повністю. Для форми звернення можна додати пояснення: «Надішліть нам ваше питання через форму, і ми надамо відповідь на електронну пошту або передзвонимо за вказаним номером».

Цей покроковий посібник створено так, щоб стажер міг виконати всі завдання самостійно. Він охоплює планування проєкту, розробку сайту і подальше тестування. Всі приклади коду і текстів – орієнтовні, їх можна доповнювати відповідно до місцевих реалій і потреб БТІ.

Джерела: Рекомендовані офіційні джерела (документація WordPress/Google Sites, MDN Web Docs) використано для технічних рекомендацій (HTML-теги, метатеги, валідація) ¹⁰ ¹ ¹¹ ¹³ ³ ⁴, а приклади контенту запозичені з реальних українських сайтів БТІ ⁵ ⁷ ⁶. Наявність цих посилань дає змогу переконатися в правильності рекомендацій і використовувати їх як довідковий матеріал.

¹ ² ¹⁰ What's in the head? Web page metadata - Learn web development | MDN
https://developer.mozilla.org/en-US/docs/Learn_web_development/Core/Structuring_content/Webpage_metadata

³ ⁴ 404 Not Found - HTTP | MDN
<https://developer.mozilla.org/en-US/docs/Web/HTTP/Reference/Status/404>

⁵ Бюро технічної інвентаризації в Києві - BTIDP - експертно, швидко, мобільно!
<https://btidp.com.ua/>

⁶ Відповіді на запитання користувачів | Київське БТІ. Офіційний сайт
<https://bti.com.ua/questions?page=207>

7 **Контакти - БТИ Киев**

<https://bti.ua/uk/kontakty.html>

8 9 **Google Sites vs WordPress: 2026 Comparison**

<https://www.hostingseekers.com/blog/google-sites-vs-wordpress/>

11 **HTML attribute: required - HTML | MDN**

<https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Attributes/required>

12 **bash - How do i write a shell script for checking website live or not - Stack Overflow**

<https://stackoverflow.com/questions/17522360/how-do-i-write-a-shell-script-for-checking-website-live-or-not>

13 **Your first form - Learn web development | MDN**

https://developer.mozilla.org/en-US/docs/Learn_web_development/Extensions/Forms/Your_first_form