

АНАЛІЗ ЖИТТЄВОГО ЦИКЛУ ТА ПРОЦЕСУ РОЗРОБКИ WEB-ДОДАТКІВ

Ищенко М. Д., Білоус М. Ю.

Харківський національний університет радіоелектроніки

Україна, 61166, Харків, пр. Науки 14

E-mail: mykhailo.ishchenko@nure.ua, maryna.bilous@nure.ua

Анотація: У даній статті було проаналізовано етапи життєвого циклу та їх особливості у розробці web-додатків. Розглянуто основні принципи та ключові поняття, на яких побудований кожен етап. В результаті проаналізовано кожен етап і зроблені відповідні висновки стосовно ефективності та особливостей, що в подальшому буде використано при створенні web-сайту.

Ключові слова: веб, життєвий цикл, сайт, карта сайту, фронт-енд

ANALYSIS OF LIFE CYCLE AND PROCESS OF WEB APPLICATION DEVELOPMENT

M. Ishchenko, M. Bilous

Kharkiv Kharkiv National University of Radio Electronics

Ukraine, 61166, Kharkiv, Nauky av, 14

E-mail: mykhailo.ishchenko@nure.ua, maryna.bilous@nure.ua

Abstract: This article analyzed the stages of life cycle and their features in development of web applications. The basic principles and key concepts on which each stage is built are considered. As result, each stage was analyzed and appropriate conclusions were made regarding effectiveness and features, which will later be used in creation of web-site.

Keywords: web, life cycle, site, sitemap, front-end

Сучасні тенденції ведуть до того, що більшу частину ринку завойовує web-сфера через свою простоту у використанні. Пересічному користувачеві звичне поняття сайту і він з легкістю використовує web-технології в повсякденному житті. Нескінченні зміни, оптимізації та покращення просто потребують додаткового аналізу життєвого циклу web-додатків для покращення якості кінцевого продукту.

Життєвий цикл web-додатків складається з цілісних семи етапів (рис. 1).



Рисунок 1 – Життєвий цикл web-додатка

- розуміння вимог клієнта;
- дослідження та розробка документу бізнес-правил;
- планування;

- проектування;
- розробка;
- тестування та розгортання;
- обслуговування.

Найважливішою частиною будь-якого процесу веб-розробки є розуміння того, чого хоче клієнт. Всі інші наступні кроки залежать від цього. Якщо не зробити цей крок правильно, швидше за все не вийде отримати потрібне програмне забезпечення, а отже і подальше фінансування.

Будь-яка невизначеність на цьому етапі повинна узгоджуватися та документуватися у відповідних ресурсах з замовником для усунення непорозумінь.

Перший крок розробки веб-сайту завжди повинен починатися зі збору вимог. Призначення веб-додатку, цілі, цільової аудиторії додатку – обговорення всього цього з клієнтом для розуміння вимог є надзвичайно важливим.

1. Мета розробки веб-сайту повинна бути дуже чіткою. Незалежно від того, чи мета проєкту просувати свої послуги, продавати товари в інтернеті або просто надавати інформацію про бізнес в інтернеті, це має вирішальне значення для компанії, що займається веб-розробкою.

2. Кінцеві цілі є обов'язковими. Важливо, яке бачення стоїть за веб-сайтом та які цілі, на думку клієнта, він досягне.

3. Цільова аудиторія допомагає зрозуміти потік користувачів. Різні вікові групи будуть реагувати на різні UI/UX. Отже, важливо знати, на яку групу аудиторії ми орієнтуємося, перш ніж розробляти продукт.

На початкових етапах обговорення вимог варто організовувати постійні детальні зустрічі з замовниками, щоб отримати повну картину. Також потрібно мати на увазі, що більш деталізованим буде цей етап, то точнішою буде оцінка вартості та часу проєкту.

Далі розглянемо складання документу стосовно бізнес-правил.

Після того, як вимоги зібрані на високому рівні, їх необхідно задокументувати. Будь-який хороший бізнес-аналітик завжди прагне отримати загальне розуміння бізнес-правила, а потім перевести його в Business Requirements Document (BRD), який буде використовуватися в якості довідника для процесу розробки.

BRD завжди має бути точним, легким для розуміння та однозначним. Це документ, на який розробники будуть посилалися, коли почнуть розробку веб-додатку, тому він повинен містити достатню кількість деталей і розбивку, де це необхідно.

Цей конкретний крок процесу розробки веб-сайту також допомагає уникнути перевитрати коштів і часу. Вся розробка повинна відповідати цьому документу, і будь-які нові модифікації вимагатимуть додаткових зусиль, як з точки зору грошей, так і з точки зору часу.

Потім йде планування.

Після того, як була зібрана уся необхідна інформація та отриманий детальний BRD, наступним етапом є планування проєкту. До того, як буде написаний перший рядок коду, вкрай важливо визначити завдання і управляти розподілом ресурсів, задіяних в розробці веб-додатку.

Цей етап в основному складається з наступних пунктів (рис. 2) [1]:

- структурне моделювання;
- визначення архітектури сайту;
- вибір технічного стеку;
- біфуркація віх;
- прийняття рішення про ресурси та кошторис.

Фундаментальний крок планування повинен починатися з побудови структури та визначення архітектури сайту. Незважаючи на те, що технічний прогрес рухається вперед [2-

7], цей етап вже тривалий час залишається незмінним, а саме створення та затвердження каркасу та карти сайту.

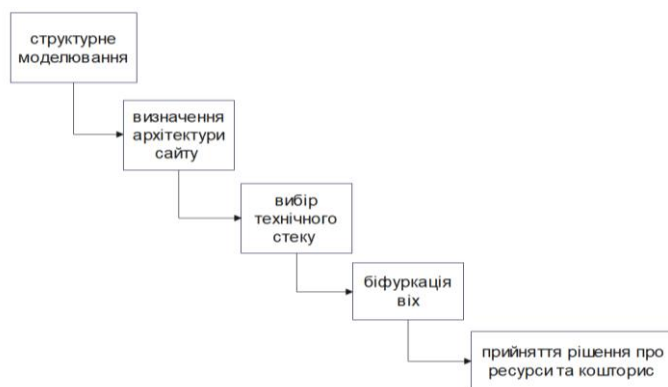


Рисунок 2 – Етапи планування

Карта сайту – це ієрархічний список всіх сторінок сайту, який допомагає команді розробників відштовхуватися від нього. Дизайнери і розробники отримують повне уявлення не тільки про структуру, але і про загальний розподіл сторінок на сайті. Без карти сайту він буде неорганізованим.

Каркас – це спосіб візуалізації дизайну сайту на структурному рівні, який є надзвичайно важливим для виявлення проблем юзабіліті на ранній стадії. Він також допомагає візуалізувати повний шлях користувача і вибрати відповідні компоненти інтерфейсу.

Правильне комплексне планування фокусується на всьому, що необхідно для успішного завершення проекту. Планування проекту допомагає краще використовувати ресурси компанії і гарантує, що час, відведений на кожен етап, буде використаний оптимально.

Що стосовно дизайну, то фінансовий дизайн – це не тільки про естетику, але й про взаємодію з користувачем. Дизайн веб-сайту є дуже важливим для конверсії.

Після того, як каркас і архітектура сайту створені, дизайнери починають працювати над візуальними елементами.

На цьому етапі дизайнери починають працювати над візуальним брендом. Після цього процесу можна фактично уявити, як буде виглядати сайт. Макет, екрани, кнопки, верхні і нижні колонтитули, навігація, фотографії, відео та інші візуальні елементи – все це додається на цьому етапі процесу розробки веб-сайту.

Макет веб-сайту (включаючи колір, зображення логотипів тощо) повинен бути обговорений з клієнтом і адаптований відповідно до його уподобань та смаку цільової аудиторії, що виходить ще з першого пункту, «розуміння вимог клієнта».

Процес розробки. На цьому етапі розробники приступають до кодування і перетворюють статичну веб-сторінку в працююче програмне забезпечення.

Кодування розробки програмного забезпечення складається з двох частин:

Розробка інтерфейсу: це називається розробкою «на стороні клієнта». Це те, що користувачі бачать і з чим взаємодіють у браузері. Взаємодія з користувачем має тут першорядне значення.

Фронтенд розробники фактично не створюють дизайн веб-сайту, але вони відповідають за втілення дизайну в життя. Вони беруть статичні елементи і перетворюють їх на повністю функціонуючі інтерактивні веб-сайти.

Back-end розробка: частина веб-сайту, яка не видима для користувачів. Бекенд взаємодіє з фронтендом і надсилає інформацію в обох напрямках, щоб користувачі могли взаємодіяти з функціями веб-сайту.

Розробники бекенд-програмного забезпечення гарантують, що вся бізнес-логіка та зберігання даних реалізовані належним чином. Створення та інтеграція баз даних, розробка та інтеграція API, перевірка безпеки і т.д. – все це є частиною розробки бекенда.

Тестування та розгортання. Будь-яке програмне забезпечення завжди повинно бути ретельно протестоване перед розгортанням його у виробництві, щоб виробничому середовищу не загрожували помилки. Отже, наявність належного, впорядкованого процесу тестування має вирішальне значення.

Веб-додаток повинен бути ретельно протестований перед тим, як запустити його в роботу. Тестування в основному проводиться для того, щоб переконатися у відсутності помилок або непрацюючих посилань, а також для того, щоб переконатися, що веб-додаток готовий до запуску.

Прийняття користувачами, бета-тестування, функціонального тестування та тестування продуктивності, тестування «білого ящика» або «чорного ящика» – це деякі з багатьох методів тестування, які компанії-розробники програмного забезпечення застосовують для перевірки своєї роботи з метою усунення помилок, які можуть поставити під загрозу якість продукту.

Як тільки тестувальники дають добро, код передається у виробництво.

Завжди потрібно мати копію стадійного сервера, щоб у разі серйозних збоїв у виробництві після розгортання вони могли використовувати резервну копію для відновлення виробництва.

Обслуговування. Процес веб-розробки не закінчується з розгортанням. Обслуговування програмного забезпечення відіграє величезну роль у повному процесі веб-розробки.

Метою супроводу програмного забезпечення є модифікація та оновлення програмних додатків для виправлення помилок, розширення можливостей та додавання функцій для підвищення продуктивності системи.

Будь-які оновлення, модифікації, виправлення помилок, патчі та розробка додаткових функцій з метою підвищення продуктивності існуючого програмного забезпечення включаються в обслуговування програмного забезпечення.

Таким чином, при створенні WEB-додатків необхідно притримуватися того, щоб під час виникнення будь-якої проблеми команда розробників програмного забезпечення негайно мала можливість виправляти. Будь-яка неправильна інформація або застарілий вміст призводить до втрати потенційного клієнта.

ЛІТЕРАТУРА

1. Al-Fedaghi, Sabah. Developing web applications / Sabah Al-Fedaghi // International journal of software engineering and its applications. – 2011. – Vol. 5(2). – P. 57-68.
2. Sotnik, S. Features of Database Types / Z. Deineko, S. Sotnik, O. Vovk, V. Lyashenko // International Journal of Engineering and Information Systems (IJEAIS). – 2021. – Vol. 5 (10). – P. 73-80.
3. Al-Sherrawi, M.H. Information model of plastic products formation process duration by injection molding method / M.H. Al-Sherrawi, A.M. Saadoon, S. Sotnik, V. Lyashenko // International Journal of Mechanical Engineering and Technology. –2018. – Vol. 9 (3). – P. 357–366.
4. Mohammad, A. Informational and Structural-Parametric Models of Inductions Micromotors / A. Mohammad, V. Lyashenko, S. Sotnik et al. // IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE). – Volume 13, Issue 2. – 2018. – P. 66-76.
5. Matarneh, R. Development of an Information Model for Industrial Robots Actuators / R. Matarneh, S. Sotnik, Z. Deineko, V. Lyashenko // IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE). – Volume 16, Issue 1. – 2019. – P. 61-67.
6. Deineko, Zh. Confidentiality of Information when Using QR-Coding / Zh. Deineko, S. Sotnik, V. Lyashenko // International Journal of Academic Information Systems Research (IJAISR). – Vol. 6, Issue 9. – 2022. – P. 10-15.
7. Lyashenko, V., Sotnik S. Semantic Model Workspace Industrial Robot / V. Lyashenko, S. Sotnik // International Journal of Academic Engineering Research (IAER). – 2021. – 2021. – Vol. 5, Issue 9.– P. 40-48.