

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра інженерії програмного забезпечення

АБРАМОВА АННА СЕРГІЇВНА

УДК 004.054

**ЗАСТОСУНОК ДЛЯ АНАЛІЗУ РЕЗУЛЬТАТІВ ТЕСТУВАННЯ
ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

**Автореферат кваліфікаційної роботи на здобуття ступеня
вищої освіти «Бакалавр»**

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня кваліфікація «Бакалавр з інженерії програмного забезпечення»

Миколаїв – 2020

Кваліфікаційною роботою є рукопис.

Робота виконана в Чорноморському національному університеті імені Петра Могили Міністерства освіти і науки України на кафедрі інженерії програмного забезпечення.

Керівник: **Боровльова Світлана Юріївна,**
Чорноморський національний університет імені
Петра Могили, старший викладач кафедри
інженерії програмного забезпечення

Рецензент: кандидат фізико-математичних наук
Кулаковська Інесса Василівна,
Чорноморський національний університет імені
Петра Могили, доцент кафедри інтелектуальних
інформаційних систем

Захист відбудеться «25» червня 2020 р. о 9 год. на засіданні екзаменаційної комісії (ауд. 2.309) у Чорноморському національному університеті імені Петра Могили за адресою: вул. 68 Десантників, 10, Миколаїв, 54003.

З кваліфікаційною роботою можна ознайомитися в бібліотеці Чорноморського національного університету імені Петра Могили за адресою: вул. 68 Десантників, 10, Миколаїв, 54003.

Автореферат представлений «18» червня 2020 р.

Секретар
екзаменаційної комісії

І. О. Кандиба

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність роботи. На сучасному етапі розвитку людства Інтернет виступає в ролі потужного інструменту з пошуку та надання інформації. За статистикою, більше половини жителів планети мають доступ до мережі Інтернет. Як наслідок, розробка web-сайту в мережі Інтернет дозволить використовувати сучасні технології для розвитку інформаційної підтримки та реклами. У зв'язку з цим web-програмування виділяється як самостійна галузь програмування. Спочатку дана галузь не могла бути порівнянна з іншими областями програмування. В результаті розвитку web-технологій комп'ютер став інструментом інформаційної підтримки. Великий пласт соціальних груп, які підключаються до мережі Інтернет і шукають там інформацію, весь час розширюється, оскільки інформація стає доступною людям, що не відносяться до категорії фахівців в області інформаційних технологій.

Метою дипломної роботи є візуалізація результатів тестування програмного забезпечення за рахунок створення вебзастосунку.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- Вивчити теоретичні аспекти створення web-сайтів.
- Провести аналіз існуючого програмного забезпечення для створення web сайтів.
- Розглянути мови програмування, що використовуються в web технологіях.
- Провести аналіз засобів створення баз даних для web-сайтів.
- Визначити структуру і призначений для користувача інтерфейс сайту.
- Провести аналіз працездатності створеного сайту.

Об'єкт дослідження – процес розробки інформаційної системи для аналізу результатів тестування програмного забезпечення.

Предмет дослідження – технології створення інформаційної системи для простої візуалізації результатів тестування, спираючись на отримані результати мануального тестування програмного забезпечення.

Практичне значення отриманих результатів. Отримані теоретичні результати доведено до розробки інформаційної системи для візуалізацій результатів тестування будь-якого програмного продукту. В цій роботі були розглянуті теоретичні основи розробки, сучасні методології та новітні технології створення подібних систем.

Апробація результатів кваліфікаційної роботи. Основні положення кваліфікаційної роботи доповідалися і обговорювалися на ХІХ міжнародній науково-практичній конференції «Ольвійський форум – 2020 : стратегії країн Причорноморського регіону в геополітичному просторі».

Публікації. Результати кваліфікаційної роботи опубліковані в 1 збірнику матеріалів міжнародної науково-практичної конференції.

Структура дипломної роботи. Дипломна робота складається із вступу, чотирьох розділів, висновків, додатків. Загальний обсяг роботи складає 87 сторінок, 27 рисунків, 2 таблиці, 18 посилань на літературні джерела та 3 додатків.

ОСНОВНИЙ ЗМІСТ РОБОТИ

У вступі обґрунтовано актуальність обраної теми, сформульовано мету і задачі дослідження, визначено об'єкт і предмет дослідження.

У першому розділі дипломної роботи приведено основні поняття та аналіз принципів створення сучасних інформаційних систем.

Створення сучасних вебсайтів складається з основних двох етапів:

- Frontend;
- Backend.

Frontend і Backend – два найпопулярніші терміни, що використовуються в веб-розробці. Ці терміни є дуже важливими для веб-розробки, але дуже відрізняються один від одного. Кожна сторона повинна спілкуватися і ефективно працювати з іншою як єдине ціле, щоб поліпшити функціональність веб сайту.

Це дві найбільш фундаментальні частини розробки програмного забезпечення, які відіграють важливу роль в веб-розробці. Зовнішній інтерфейс – це те, що ви бачите і з чим взаємодієте, а внутрішній – як все це працює.

Зовнішній інтерфейс може посилатися на графічний інтерфейс користувача, тоді як внутрішній інтерфейс – це та частина веб сайту, яку ви не можете бачити або з якою не можете взаємодіяти. Ці два терміни дуже важливі для веб-розробки, але вони сильно відрізняються один від одного.

Термін HTML (Hyper Text Markup Language) означає «мова маркування гіпертекстів». HTML – своєрідна протилежність складним мовам програмування, відомим тільки фахівцям.

Перша версія HTML була розроблена в 1989 році Тімом Бенерс-Лі для популярного в минулому браузера Mosaic. Але в той час ні для мови, ні для браузера не знайшлося гідного застосування. У 1993 році з'явився HTML+, і ця версія також залишилася непоміченою. Початок широкого використання гіпертексту дала версія 2.0, яка з'явилася в червні 1994 року. Це був рік зростання популярності WWW по всьому світу. Елементи, включені в версію 2, в більшості своїй використовуються донині.

HTML документ починається з `<!DOCTYPE html>` – кожна веб сторінка повинна починатися саме з нього.

Після йде парний тег `<html> </html>`. Цей тег є контейнером веб сторінки, тобто до нього і після нього нічого не повинно бути.

У парному тезі `<head> </head>` пишеться обов'язковий тег `<title> </title>`. У тезі тайтл пишеться заголовок сторінки, який відображається у видачі пошукової системи і у вкладці браузера. У контейнері `<head> </head>` зазвичай знаходяться різні мета-теги і підключаються стилі і скрипти.

Весь текст, графіка та інший код повинен знаходитися усередині тегів `<body> </body>`.

CSS (англ. Cascading Style Sheets – каскадні таблиці стилів) – формальна мова опису зовнішнього вигляду документа, написаного з використанням мови розмітки. Під терміном CSS ми розуміємо фразу "Каскадні таблиці стилів". Вони впроваджуються в HTML-код і не вимагають ніяких спеціальних редакторів і компіляторів, адже інтерпретуються вони звичайним браузером.

JavaScript – одна з найпопулярніших мов програмування у світі з історією

понад двадцяти років. Це також одна з трьох основних мов програмування для веб-розробників:

- HTML: дозволяє додавати вміст на ваш веб-сайт.
- CSS: визначає стилі та зовнішній вигляд веб-сторінки.
- JavaScript: покращує поведінку веб-сайтів.

JavaScript був створений за 10 днів Брандан Ейхом, який працював у Netscape ще в 1995 році. Спочатку цю мову назвали Mocha, ім'я мови було змінено на Mona, а потім на LiveScript, поки JavaScript нарешті не став доступним. Початкова версія мови була обмежена браузером Netscape і пропонувала вузьку функціональність, але з часом вона продовжувала розвиватися частково завдяки спільноті розробників, які не залишили роботу над нею.

JavaScript був стандартизований в 1996 році і офіційно отримав назву ECMAScript, а ECMAScript 2 випущений у 1998 році, та ECMAScript 3 далі в 1999 році. Це змінилося на сьогоднішній JavaScript, який зараз працює не тільки на різних браузерах, але і на різних пристроях, включаючи мобільні та настільні пристрої.

З тих пір JavaScript продовжував зростати на 92% сайтів, які ним користуються в 2016 році. Лише за двадцять років він перетворився з примітивної мови програмування в один з найпопулярніших інструментів арсеналу веб-розробників.

Що ж стосується інших можливостей – вони залежать від оточення, в якому запущений JavaScript. У браузері JavaScript вміє робити все, що відноситься до маніпуляції зі сторінкою, взаємодії з відвідувачем і, в якійсь мірі, з сервером:

- створювати нові HTML-теги, видаляти існуючі, змінювати стилі елементів, ховати, показувати елементи і т.д.
- реагувати на дії відвідувача, обробляти кліки миші,
- переміщення курсора, натискання на клавіатуру і т.д.

- посилати запити на сервер і завантажувати дані без перезавантаження сторінки (ця технологія називається "AJAX")

- отримувати і встановлювати cookie, запитувати дані, виводити повідомлення

Більшість можливостей JavaScript в браузері обмежена поточним вікном і сторінкою.

- JavaScript не може читати / записувати файли на жорсткий диск, копіювати їх або викликати програми. Він не має прямого доступу до операційної системи.

- JavaScript, що працює в одній вкладці, не може спілкуватися з іншими вкладками і вікнами, за винятком випадку, коли він сам відкрив це вікно або декілька вкладок з одного джерела (однаковий домен, порт, протокол).

Є як мінімум три чудових особливості JavaScript:

- повна інтеграція з HTML / CSS;
- прості речі робляться просто;
- підтримується всіма поширеними браузерами і включений за замовчуванням.

У другому розділі дипломної роботи розглядаються методології проектування, розробки та програмування сучасних інформаційних систем. Також розглядається процес проектування UI вебзастосунку та проводиться огляд використаних технологій.

Web-сайт це інформаційний ресурс, що складається з пов'язаних між собою гіпертекстових документів (web-сторінок). web-сторінка являє собою текстовий файл з розширенням *.html. який містить текстову інформацію і спеціальні команди, що визначають в якому вигляді ця інформація буде відображатися у вікні браузера.

Типи сайтів за метою створення:

- Комерційні.
- Некомерційні

Типи сайтів по доступності:

- Відкриті – доступні для будь-яких відвідувачів
- Напіввідкриті – для доступу необхідна реєстрація
- Закриті – доступні для вузького кола людей

Типи сайтів по функціональності і стилю оформлення:

- Статичні – сайт будується з статичних html-сторінок. Ці сторінки пов'язані між собою за допомогою гіперпосилань. На сервері зберігаються готові зверстані сторінки. Користувач отримує web-сторінки в незмінному вигляді.
- Динамічні – основна особливість полягає в тому, що різні частини сторінки можуть зберігатися в різних файлах в базі даних. При видачі користувачеві запитувана веб-сторінка генерується «на льоту».

Типи сайтів по фізичному розташуванню:

- Загальнодоступні – доступні для користувачів мережі інтернет.
- Локальні – доступні тільки в межах локальної мережі. Це можуть бути як корпоративні сайти організацій, так і сайти приватних осіб в локальній мережі провайдера.

Типи сайтів за особливостями подання інформації та категоріями вирішуваних завдань:

- інтернет-представництва організацій
- інформаційні ресурси
- веб-сервіси (сайти, на яких надаються послуги для виконання певних завдань)
- соціальні мережі (комбіновані веб-сервіси)

Графічні (візуальні) моделі – це інструменти для візуалізації, опису, проектування та документації архітектури системи. Розробка індустріальної моделі програмного забезпечення настільки ж необхідна, як і наявність проекту під час будівництва великої будівлі. Хороші моделі є основою взаємодії учасників проекту та гарантують правильність архітектури. У міру збільшення складності систем важливо мати хороші методи моделювання. Хоча існує багато інших

факторів, від яких залежить успіх проекту, наявність суворого стандарту для мови моделювання дуже важлива.

У структурному аналізі та дизайні використовуються різні моделі, які описують:

1. функціональну структуру системи;
2. послідовність виконуваних заходів;
3. передачу інформації між функціональними процесами;
4. зв'язок між даними.

jQuery – це бібліотека JavaScript, що містить готові функції, всі операції jQuery виконуються з коду JavaScript. Бібліотека jQuery допомагає легко маніпулювати вмістом DOM. При цьому вихідні файли HTML та CSS не змінюються, зміни вносяться лише для відображення сторінки користувачеві. Бібліотека jQuery також пропонує зручний API для роботи з AJAX.

Google Charts – це поєднання двох інтерфейсів програмування (API), API Google Chart і API візуалізації Google. API Google Chart створює візуалізацію статичних даних та вставляє їх у веб-сторінки. Доступні типи візуалізації включають стандартні діаграми розсіювання, лінії, смуги, pie diagram та прямокутні діаграми, а також діаграми Венна, динамічні значки чи мітки, формули тощо. Крім того, можна створювати та вбудовувати карти у веб-сторінки.

У третьому розділі описано моделювання та проектування інформаційної системи.

Розробка вебзастосунків (web-розробка) – це сукупність заходів та дій для планування та створення вебсайтів в Інтернеті залежно від цілей та завдань.

Сучасний вебсайт може вирішити як маркетингові проблеми – просування товарів та послуг компанії на ринку, конкурентні переваги компанії, збільшення попиту на продукцію тощо. Він також може бути джерелом інформації, завдання якого обмежуються інформуванням певної цільової аудиторії.

Безпосередньому створенню сайту передують детальний всебічний аналіз, який визначає критерії, яким повинен відповідати майбутній веб-проект.

Процес створення веб-програми зазвичай включає у себе такі основні етапи:

- визначення цілей та завдань проекту;
- створення, опрацювання технічного завдання (ТЗ) на розробку сайту;
- створення макета дизайну сайту;
- верстка;
- програмування;
- наповнення контентом;
- контроль якості (QA);
- впровадження та обслуговування, SEO оптимізація.

Кожен з цих етапів є достатньо незалежним, що дозволяє вибрати графік роботи та виконавця для кожного з них. Слід враховувати, що під час використання деяких методологій розробки програмного забезпечення, процес підтримки якості розпочинається ще на етапі проектування вимог до інформаційної системи.

Вимоги щодо створення інформаційної системи:

- web-ресурс повинен бути адаптивним, тобто, веб-сторінки повинні забезпечувати правильне відображення web-ресурсу на різних пристроях, підключених до інтернету і динамічно підлаштовуватись під задані розміри вікна браузера;
- web-ресурс повинен бути кросбраузерним (це правильна верстка web-ресурсу за допомогою якої сторінки системи однаково відображаються в різних браузерах);
- web-ресурс повинен бути інтуїтивно зрозумілим та мати привабливий графічний дизайн (це необхідно для збереження відвідуваності веб-застосунк)
- web-ресурс повинен складатися з наступних сторінок: головна, та сторінка створення візуалізацій результатів проведеного тестування будь-якого програмного продукту.

Основним завданням дизайну веб-сайтів є надання користувачам інформації зручним та привабливим способом. Дизайн визначає стиль сайту. Гарний дизайн

дозволяє з першого погляду дати користувачеві зрозуміти, що саме їх чекає на певній сторінці.

Дизайн повинен бути привабливим, але не відволікаючим. Насамперед, користувачі заходять на сторінку заради інформації. Дизайн сайту повинен підкреслювати цю інформацію, допомагати користувачам отримати доступ до неї, та не повинен стати перешкодою.

Розробка вебдизайну відбувається в кілька етапів:

Розробка технічних характеристик. У ТЗ необхідно враховувати структуру сайту, існуючий фірмовий стиль компанії замовника, враховувати побажання замовника та очікування цільової групи.

Структурування сторінки. На цьому етапі створюється приблизна структура сторінок сайту. Сторінки макету містять вміст та всі важливі елементи майбутньої сторінки. Такий підхід економить час – змінити ескіз набагато простіше, ніж готовий дизайн.

Візуальний дизайн. Дизайнер створить дизайн веб сайту на основі структури та технічних умов. Далі макет відправляється замовнику на затвердження. Макет зазвичай затверджується замовником із проханням про незначні зміни. Іноді потрібні значні зміни або розробка нового дизайну.

Макет і програмування. Ці фази більше пов'язані з розробкою сайту. Готовий дизайн зберігається, завантажується в систему управління вмістом та розміщується на хостингу.

Поведінка користувача під час перегляду сторінок сайту описується трьома найбільш типовими моделями: патерн F, діаграма Гутенберга та шаблон Z.

Під час тестування сайтів, 40-60% респондентів (тестерів) вказують на ті самі недоліки.

Висновок простий: люди думають аналогічно. Їм заважають або подобаються однакові речі. Вони переглядають сторінки однаково і бачать однакову інформацію на них.

Валідатор – це комп'ютерна програма, що перевіряє відповідність HTML-коду сторінок web-ресурсу і CSS-коду на відповідність сучасним стандартам.

У спеціальному розділі було здійснено аналіз умов праці та сформовано перелік вимог до робочого місця.

ВИСНОВКИ

Виконання дипломної роботи є важливим елементом учбового процесу з підготовки спеціалістів у області комп'ютерних наук, адже, під час її проходження майбутній програміст застосовує отримані у процесі навчання навички, знання на практиці.

У даній дипломній роботі було розроблено інформаційну систему для візуалізації результатів тестування програмного забезпечення. В цій роботі були розглянуті теоретичні основи розробки, методології та новітні технології створення web-сайтів.

Оцінюючи виконану роботу, можна зробити наступні висновки:

- сайт надає користувачам найбільш потрібний йому функціонал, а саме можливість візуалізації результатів тестування програмного забезпечення;
- дизайн сайту відповідає передбачуваним перевагам цільової групи, часу і цілям перебування потенційних відвідувачів на сайті;
- усі сторінки сайту адаптивні та кросбраузерні;
- швидкість завантаження сайту вища, аніж 80% аналогічних веб-сайтів;

Даний ресурс стабільний в роботі і не вимагає високих знань для роботи з ним.

АНОТАЦІЯ

Абрамова А. С. Застосунок для аналізу результатів тестування програмного забезпечення. – На правах рукопису.

Кваліфікаційна робота на здобуття освітньої кваліфікації «Бакалавр з інженерії програмного забезпечення». – Чорноморський національний університет імені Петра Могили, Миколаїв, 2020.

В даній роботі було розглянуто процес створення інформаційної системи для аналізу результатів тестування програмного забезпечення, яка спрямована на полегшення візуалізації результатів тестування програмного забезпечення.

Метою дипломної роботи є візуалізація результатів тестування програмного забезпечення за рахунок створення вебзастосунку.

Об'єкт дослідження – процес розробки інформаційної системи для аналізу результатів тестування програмного забезпечення.

Предмет дослідження – технології створення інформаційної системи для простої візуалізації результатів тестування, спираючись на отримані результати мануального тестування програмного забезпечення.

Дипломна робота містить наступні розділи: аналіз принципів створення сучасних інформаційних систем, моделювання та проектування інформаційної системи, розробка UI та програмна реалізація інформаційної системи, забезпечення здорових та безпечних умов для праці на місцях.

У першому розділі дипломної роботи приведено основні поняття та аналіз принципів створення сучасних інформаційних систем. У другому розділі дипломної роботи розглядаються принципи моделювання та проектування web-ресурсів. У третьому розділі дипломної роботи описані новітні технології, що використовуються при розробці інформаційної системи.

У результаті роботи було створено інформаційну систему для аналізу та візуалізації результатів тестування програмного забезпечення, що дає змогу отримати зрозумілі навіть звичайним користувачам графіки та схеми.

У цілому робота складається зі 87 сторінок, 2 таблиць, 27 рисунків. Список використаних джерел інформації складає 18 найменувань.

У висновках проводиться аналіз проведеної роботи та отриманих результатів.

У спеціальній частині з охорони праці йдеться про техніку безпеки користування ПК та смартфоном.

***Ключові слова:** інформаційна система, тестування, діаграми, новітні технології.*

ANNOTATION

Abramova A. S. Application for analyzing the results of software testing. – Manuscript.

In this thesis was considered the process of creating an information system for analyzing the results of software testing, which aims to facilitate the visualization of software testing results.

The object of research is the process of developing an information system for analyzing the results of software testing.

The subject of research - technologies for creating an information system for simple visualization of test results, based on the results of manual software testing.

The purpose of this work is to create an information system to visualize the results of software testing.

Thesis contains the following sections: analysis of the principles of modern information systems, modeling and design of information systems, development of UI and software implementation of information systems, providing healthy and safe working conditions in the field.

The first section of the thesis presents the basic concepts and analysis of the principles of modern information systems. The second section of the thesis discusses the principles of modeling and design of web-resources. The third section of the thesis describes the latest technologies used in the development of information systems.

As a result, an information system was created to analyze and visualize the results of software testing, which allows you to get clear even to ordinary users graphics and diagrams.

In total, the work consists of 87 pages, 2 table, 27 figures. The list of used sources of information is 18 items.

The conclusions analyze the work done and the results obtained.

The special part on labor protection deals with the safety of using a PC and a smartphone.

Key words: *information system, testing, diagrams, latest technologies.*