**Міністерство освіти і науки України**

**Національний університет «Запорізька Політехніка»**

Кафедра програмних засобів

**ЗВІТ**

з лабораторної роботи № 3

з дисципліни «Soft skills, групова динаміка та комунікації» на тему:

«СЕРЕДОВИЩЕ INTERNET ЯК ЗАСІБ КОМУНІКАЦІЇ»

**Виконав:**

студент групи КНТ-122 О. А. Онищенко

**Прийняли:**

асистент: Д. А. Каврін

2023

1 СЕРЕДОВИЩЕ INTERNET ЯК ЗАСІБ КОМУНІКАЦІЇ

1.1 Мета роботи

1.1.1 Вивчити основні можливості середовища Internet та здобути навички з пошуку інформації в Internet та використання онлайн-сервісів пошукових систем.

1.1.2 Вивчити основні можливості сучасних веб-браузерів та інструменти, що використовуються для роботи з даними в мережі.

1.1.3 Здобути навички з використання розглянутих інструментів для професійних потреб та роботи в групі.

1.2 Завдання роботи

1.2.1 Ознайомитися з основними теоретичними відомостями за темою роботи, використовуючи дані методичні вказівки, а також рекомендовану літературу.

1.2.2 Вивчити принципи роботи пошукових систем та інструментарій, що надається ними.

1.2.3 Виконати пошук інформації в Internet за допомогою різних пошукових серверів:

a) знайти та навести у звіті визначення п’яти з наступних спеціальних термінів: інформаційна технологія, програмне забезпечення, протокол, інформація, файл, дані, мова програмування, операційна система, база даних, компілятор, ASCII, графічний редактор, комп’ютерна мережа, сервер, комп’ютерний вірус, модуль, система керування вмістом, парадигма програмування, гіпертекст, віджет, комунікація, веб-сайт, URL, текстовий редактор, електронна пошта, інтернет-спільнота, фішинг, RSS, плагін, веб-портал, віртуальна машина, інтерпретатор, інтернет-бот, файрвол, проксі-сервер, веб-сервер, транслятор, емулятор, синтаксис, семантика (за варіантом);

б) знайти за допомогою системи Google Академія та привести у відповідності зі стандартом ДСТУ ГОСТ 7.1:2006 «Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання» бібліографічні дані книг або статей з журналів (за останні 5 років) за спеціальністю: 10 загалом за спеціальністю та 5 за однією з наступних тем: операційні системи Android, Linux, iOS, OS X, FreeBSD, мови програмування Python, Perl, Java, PHP, C#, Ruby, SQL, Objective-C, Erlang, мови розмітки XML та LaTex, системи керування базами даних MySQL та Oracle Database, системи керування вмістом Joomla!, Drupal, WordPress, графічні редактори CorelDRAW та Adobe Photoshop, веб-сервери Apache та IIS (за варіантом);

в) привести у звіті коротку біографію, область наукової діяльності та досягнення 5 з наступних діячів: Алан Кей, Лінус Торвальдс, Метью Грей, Енді Рубін, Тім Бернерс-Лі, Мартін Купер, Денніс Рітчі, Кен Швабер, Конрад Цузе, Чарлз Беббідж, Норберт Вінер, Алан Тюрінг, Френк Розенблат, Марвін Мінський, Алан Тюринг, Стів Джобс, В.М. Глушков, Ґвідо ван Россум, Расмус Лердорф, Б'ярн Страуструп, Кен Томпсон, А́ ндерс Ге́йлсберґ, Брендан Айк, Ні́клаус Вірт, Ада Лавлейс, Білл Гейтс, Пол Аллен, Едгар Кодд, Норберт Вінер, Стів Возняк, Річард Столмен, Ларрі Пейдж, Сергі́й Брін, Джеймс Гослінг, Брем Коен, Ларрі Волл, Міґель де Ікаса, Джордж Буль, Джон фон Нейман, Говард Каннінгем (за варіантом).

1.2.4 Обрати запит для пошуку та порівняти результати пошуку для різних пошукових систем (хоча б трьох), заповнивши таблицю, що містить такі стовпці: – No з/п; – адреса пошукового сервера; – перелік перших 10 сайтів, що найдені за результатами пошуку (URL-адреси); – коротка назва кожного сайту; – релевантність кожного отриманого результату запиту та середній результат для кожного пошукового сервера. Стовпець «релевантність запиту» розділяється на дві колонки, перша з яких заповнюється студентом на власний розсуд як суб’єктивна оцінка (за 100 бальною шкалою) відповідності знайденої інформації введеному запиту, а друга заповнюється як обчислена оцінка Rel:

де Ws – міра, що визначає частоту появи окремого слова в тексті веб-сторінки та дорівнює добутку відношень кількості входжень кожного слова в текст веб-сторінки до кількості слів у документі; k1 – коефіцієнт, що визначає значимість міри Wp по відношенню до Ws (встановлюється студентом суб’єктивно); Wp – міра, що визначає частоту появи цілого тексту запиту в тексті веб-сторінки і дорівнює відношенню кількості входжень повного тексту запиту в текст веб-сторінки до тієї ж кількості, збільшеної на одиницю.

1.2.5 Обрати деяке зображення для пошуку, виконати пошук на різних пошукових серверах, заповнити таблицю порівняння результатів аналогічно п. 1.2.4, навести результати пошуку найбільш релевантної інформації (релевантність оцінити суб’єктивно).

1.2.6 За допомогою бази знань та набору обчислювальних алгоритмів Wolfram Alpha розв’язати диференційне рівняння та систему лінійних алгебраїчних рівнянь, виконати операції над матрицями.

1.2.7 Створити додаткову поштову скриньку та перенаправити листи з неї на основну.

1.2.8 Вивчити принципи роботи в Internet за допомогою браузерів Google Chrome, Mozilla Firefox, Opera, Internet Explorer, Safari (хоча б трьох з перелічених).

1.2.9 Обрати критерії порівняння та виконати порівняння розглянутих браузерів, для чого заповнити відповідну таблицю.

1.2.10 Синхронізувати дані веб-перегляду на різних пристроях з доступом до Internet в одному з браузерів.

1.2.11 Заблокувати рекламу в браузері.

1.2.12 Оформити звіт з роботи.

1.2.13 Відповісти на контрольні запитання.

1.3 Короткі теоретичні відомості

Git та Subversion - дві найпопулярніші системи контролю версій, що використовуються при розробці програмного забезпечення. Ось короткий огляд того, як вони працюють:

Git: Git - це розподілена система контролю версій, яка дозволяє декільком розробникам одночасно працювати над однією і тією ж кодовою базою. Кожен розробник має власну копію сховища, в яку він може вносити зміни незалежно. Зміни потім об'єднуються шляхом порівняння та об'єднання змін, зроблених кожним розробником. Git використовує модель розгалуження, де кожна гілка представляє окрему версію кодової бази. Розробники можуть створювати та об'єднувати гілки за потреби для роботи над різними функціями або виправленнями. Git відомий своєю швидкістю, гнучкістю та потужним інтерфейсом командного рядка.

Subversion: Subversion - це централізована система контролю версій, яка використовує модель клієнт-сервер. Всі зміни до кодової бази вносяться до центрального репозиторію, який розміщується на сервері. Розробники можуть отримати копію сховища на своїй локальній машині, внести зміни, а потім зафіксувати ці зміни назад до центрального сховища. Subversion використовує структуру "стовбур-гілки-теги" (TBT), де основна лінія розробки (стовбур) відокремлена від відгалужень (гілок) і версій (тегів) релізів. Subversion відомий своєю простотою, легкістю у використанні та сумісністю з іншими інструментами.

І Git, і Subversion надають потужні можливості контролю версій, які можуть допомогти командам ефективно співпрацювати над проектами з розробки програмного забезпечення. Вибір між Git та Subversion залежить від ваших конкретних потреб, уподобань та вимог проекту. Git часто обирають за швидкість, гнучкість та модель розгалуження, тоді як Subversion - за простоту, легкість у використанні та централізований підхід.

1.4 Результати пошуку інформації

2.3.3 Використовуючи систему керування версіями Subversion (для пунктів 2.3.3-2.3.15), створити каталог проекту та базову структуру репозиторію.

1.5 Таблиця порівняння обраних до аналізу браузерів

1.6 Порівняльні таблиці, що відображують результати пошуку інформації та зображень у середовищі Internet за допомогою різних пошукових серверів.

1.7 Копії екранних форм з результатами роботи з поштовою скринькою, математичними обчисленнями та веб-браузерами.

Висновки

В цій роботі ми навчилися використовувати можливості систем керування версіями Subversion та Git для групової роботи.

2.5.1 Що таке система контролю версій?

Система контролю версій (СКВ) - це програмний інструмент, який допомагає вам керувати змінами в наборі файлів з плином часу. VCS дозволяє відстежувати історію змін у ваших файлах, співпрацювати з іншими розробниками та керувати кількома версіями вашої кодової бази. За допомогою VCS ви можете легко повернутися до попередньої версії коду, якщо це необхідно, переглянути зміни, внесені різними розробниками, та об'єднати зміни з різних джерел.

2.5.2 Які існують варіанти організації сховищ у Subversion?

Subversion надає декілька варіантів організації сховищ, зокрема

* Структура "стовбур-гілки-теги" (TBT): Це найпоширеніша структура, яка використовується у Subversion, де основна лінія розробки (стовбур) відокремлена від відгалужень (гілок) та версій (тегів).
* Структура одного проекту: Ця структура може бути використана для невеликих проектів, де структура TBT є непотрібною. Усі файли проекту зберігаються у кореневому каталозі сховища.
* Структура з декількома проектами: Ця структура може бути використана для проектів, які складаються з декількох пов'язаних між собою проектів. Кожен проект має власний каталог у сховищі.

Модульна структура: Ця структура може бути використана для великих проектів, які складаються з декількох модулів. Кожен модуль має власний каталог у сховищі.

2.5.3 Які типи файлів можна зберігати у сховищі?

Subversion може зберігати будь-які типи файлів, включаючи файли коду, файли зображень, аудіофайли, відеофайли і текстові файли. Однак, зазвичай рекомендується зберігати у Subversion лише текстові файли, оскільки двійкові файли (наприклад, зображення, аудіо, відео) можуть швидко роздути сховище і ускладнити керування ним. Якщо вам потрібно зберігати бінарні файли у Subversion, рекомендується використовувати розширення SVN Git Large File Storage (LFS) або спеціальну систему керування бінарними артефактами.

2.5.4 Що таке сховище Subversion?

Сховище Subversion - це центральне місце, де зберігаються всі файли проекту Subversion. Сховище містить всю історію версій проекту, включно з усіма змінами, внесеними до файлів з плином часу. Сховище зазвичай розміщується на сервері і до нього можуть мати доступ декілька розробників для спільної роботи над проектом.

2.5.5 Скільки сховищ може використовувати Subversion для зберігання даних?

Subversion може використовувати будь-яку кількість сховищ для зберігання даних, залежно від ваших потреб. Кожне сховище є незалежним і містить власний набір файлів та історію версій. Зазвичай використовується один сховище для одного проекту, але ви також можете використовувати декілька сховищ для одного проекту, якщо це необхідно. Кількість сховищ, які слід використовувати, залежить від вимог вашого проекту та складності вашої кодової бази.