

ЛАБОРАТОРНА РОБОТА № 1

Загальна схема наукового дослідження. Пошук, накопичення та обробка наукової інформації

Мета роботи: ознайомитися з основними джерелами науково-технічної інформації процесів, сформувавши загальну схему збору і аналізу наукової інформації. Здійснити пошук наукової інформації, систематизувати її, оформити у вигляді звіту з посиланнями на використані джерела.

Порядок виконання роботи:

1. Виписати оброблену інформацію у вигляді бібліографічних описів інформаційних джерел (30 – 35 джерел). Оформити їх згідно з ДСТУ 8302:2015“Бібліографічні посилання: загальні положення та правила складання” .
2. Відповідно до визначеного напрямку наукового дослідження (магістерської роботи) з використанням вказаних джерел науково-технічної інформації здійснити пошук наукової інформації.
3. Сформувавши висновки по роботі.

Тема дипломної роботи: Система для документування та обміну знаннями в команді розробників.

Опис дипломного проекту:

Проект "Система для документування та обміну знаннями в команді розробників" передбачає створення веб-системи для централізованого управління документацією та знаннями всередині команд розробників. Метою є спрощення процесу створення, редагування та збереження технічної документації, а також статей і нотаток, що сприяють обміну знаннями та досвідом.

Ключовими функціями є підтримка створення та редагування документів у форматі Markdown, управління правами доступу, можливість

пошуку і фільтрації контенту. Проект спрямований на підвищення ефективності роботи команд, покращення комунікації та збереження корпоративної пам'яті.

Актуальність теми:

У сучасній розробці програмного забезпечення ефективна комунікація та збереження знань є ключовими для успіху проєктів. Часто знання залишаються неформальними, що ускладнює їх використання та передачу.

Створення системи для документування та обміну знаннями допомагає командам структурувати, зберігати та ділитися важливою інформацією, сприяючи зростанню ефективності роботи та зменшенню залежності від окремих членів команди.

Наукова новизна:

Запропонована система інтегрує функціонал документування з можливістю створення статей та нотаток, що дозволяє ефективніше зберігати і передавати знання у команді.

Впровадження сучасних методів управління версіями та колаборації дозволяє оптимізувати процеси створення та оновлення документації.

Аналоги програмного засобу:

На ринку існує низка рішень для документування та управління знаннями, таких як Confluence і Notion, які надають потужні інструменти для створення й управління документацією, але можуть бути дорогими для великих команд. Відкриті платформи, такі як Wiki.js і Docusaurus, дозволяють гнучко налаштовувати систему, однак вимагають технічних знань для налаштування. GitBook і Read the Docs ідеально підходять для технічної документації та інтеграції з репозиторіями коду, хоча можуть бути обмежені у функціоналі безкоштовних версій. Кожне рішення має свої переваги та недоліки, що спонукає до розробки власної системи, адаптованої під специфічні потреби команди.

Висновок:

У ході аналізу та дослідження існуючих рішень для документування програмного забезпечення та управління знаннями в командах розробників було визначено, що наявні системи, такі як Confluence, Notion, GitBook, Wiki.js, Docusaurus та Read the Docs, мають свої сильні сторони та обмеження. Ці інструменти забезпечують різні рівні функціональності, інтеграції та гнучкості, що дозволяє користувачам обирати рішення відповідно до своїх потреб і бюджету. Однак, жодна з цих систем не поєднує всі ключові особливості, необхідні для забезпечення максимальної ефективності та зручності використання.

1. Press Release: Confluence Launches Evaluated Fixed Income Pricing Data Feed on Snowflake Marketplace. URL: <https://www.confluence.com/press-release-confluence-launches-evaluated-fixed-income-pricing-data-feed-on-snowflake-marketplace/> (дата звернення: 15.11.2024).
2. Прайс, М. С#10 і .NET 6. Сучасна крос-платформна розробка 2023, 848 с.
3. Троелсен, Э. Мова програмування С# 9 та платформа .NET 5: основні принципи та практики програмування, том 2, 10-е видання, 2021, 770 с.
4. Бен-Ган, И. T-SQL Fundamentals, 4-е издание. Microsoft Press, 2023, 608 с.
5. Що таке markdown .Приклади та застосування. URL: <https://chemerys.site/blog/shcho-take-markdown-rozmitka/> (дата звернення: 15.11.2024).
6. Мартін, Р. Чистий код: створення аналіз рефакторингу, 2019, 448 с.
7. Робертс, Дж., Бібо, Б., Марас, І. Секрети JavaScript ніндзя, 2019, 541с.
8. Резіг, Д., Фергюсон, Р., Пакстон, Д. JavaScript для професіоналів, 2017, 229 с.
9. Томас, М. Т. React в дії, 2018, 368с.

10. Фаулер, М. Шаблони корпоративних додатків, 2003, 506 с.
11. Сталлінгс, В. Cryptography and network security: principles and practice, 2010, 900.
12. Фрімен, А. Pro Entity Framework Core 6, 2023, 650 с.
13. Шелдон, Р. Як вибрати між базами даних SQL та NoSQL. URL: <https://www.red-gate.com/simple-talk/databases/nosql/how-to-choose-between-sql-and-nosql-databases/> (дата звернення: 15.11.2024).
14. Angular vs React vs Vue: Який фреймворк обрати. URL: <https://www.codeinwp.com/blog/angular-vs-vue-vs-react/#gref> (дата звернення: 15.11.2024).
15. Хайетт, А. Domain-Driven Design: Просте пояснення. URL: <https://www.alexhyett.com/domain-driven-design/> (дата звернення: 15.11.2024).
16. Notion. URL: <https://www.notion.so/> (дата звернення: 15.11.2024).
17. Порівняння AWS, Azure, GCP. URL: <https://www.digitalocean.com/resources/article/comparing-aws-azure-gcp/> (дата звернення: 15.11.2024).
18. Варуша, А. AWS проти AZURE проти GCP: порівняння трьох гігантів. URL: <https://www.n-ix.com/comparing-big-3-aws-azure-gcp/> (дата звернення: 15.11.2024).
19. Read the Docs. URL: <https://about.readthedocs.com/> (дата звернення: 15.11.2024).
20. Еланд, М. Один з розділених об'єднань в C#. URL: <https://matteland.medium.com/oneof-discriminated-unions-in-c-132e534bda99> (дата звернення: 13.04.2024).
21. Gitbook: <https://www.gitbook.com/> (дата звернення: 15.11.2024).
22. Wiki.js. URL: <https://js.wiki/> (дата звернення: 15.11.2024).

23. Лафоре, Р. Структури даних і алгоритми в Java, 2019, 704 с.
24. Макконнелл, С. Досконалий код, 2020, 896 с.
25. Кнут, Д. Мистецтво програмування, 2018, 712 с.
26. Фаулер М. UML Distilled: A Bried Guide to the Standard Object Modeling Language, 2016, 208 с.
27. Джефрі, Р. CLR via C#, 2012, 896 с.
28. Андерсон, А, Learning Microsoft Azure: Cloud Computing and Development Fundamentals, 2023, 480.
29. Вітінг,М. Amazon Web Services in Action, 2019, 500.
30. Бенкер, К. MongoDB в дії, 2018, 394.