

ПРОГРАММНАЯ СИСТЕМА ДОКУМЕНТИРОВАНИЯ ИНФОРМАЦИОННЫХ ПРОЕКТОВ И ЕЕ ЭРГОНОМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Конохов А.П.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Научный руководитель: Борисик М.М. – магистр техн. наук, ст. преподаватель кафедры ИПиЭ

Аннотация. В данной статье рассматриваются предметная область, существующие аналоги, а также особенности проектирования системы, направленной на структурированное хранение, обработку и управлению проектной документацией. Выполнен анализ ключевых задач, решаемых системой, определены эргономические аспекты интерфейса для повышения удобства работы пользователей.

Ключевые слова: документация, вики-платформы, управление проектами.

Введение. Документация является фундаментальной частью любого программного проекта, обеспечивая не только инструкции по использованию программного обеспечения, но и предоставляя информацию о причинах выбора определенных подходов, компромиссов и ограничений, с которыми сталкиваются разработчики. Организации также извлекают пользу из документации, поскольку она улучшает восприятие пользователями и снижает чрезмерную зависимость от служб поддержки. Она объясняет все особенности проекта, информирует нас о том, как мы можем с ними работать, помогает ознакомиться с функциональными возможностями проекта и позволяет нам сократить время на адаптацию.

Основная часть. Появление средств создания статической документации и онлайн-инструментов документирования связано с большими объемами информации. Когда компания *Facebook* только начинала свою программу открытых исходных кодов, многие команды создавали собственные веб-сайты для каждого из своих проектов с открытым исходным кодом. Такой подход создал проблемы, когда команду программы открытого кода попросили помочь проектным командам улучшить документацию. Поскольку каждый сайт был уникальным, добавление базовой инфраструктуры, такой как блог, последовательная навигация, поиск и т. д., стало сложной задачей [1].

Команда программы открытого кода попыталась решить эту проблему, разработав стандартный шаблон на основе *Jekyll*, который можно было бы использовать в качестве отправной точки для создания сайта проекта. Этот шаблонный подход был принят большинством запущенных проектов с открытым исходным кодом; некоторые существующие проекты даже перевели свои собственные реализации сайтов на новый шаблон. Проблема с подходом «скопировать шаблон в свое репозиторий» заключается в том, что, несмотря на согласованность платформы, распространение обновлений становится невозможным для всего набора проектов, уже использующих шаблон. Это происходит потому, что мы теряем контроль над шаблоном после того, как проект копирует его в свой репозиторий. Проекты могли свободно изменять шаблон по своему усмотрению и применять к нему свои собственные функции, характерные для конкретного проекта. Таким образом, хотя проекты используют одну и ту же платформу для создания сайтов, они уже настолько разошлись, что не могут воспользоваться новыми функциями, которые добавлены в шаблон с течением времени. Теперь, копирование новой версии шаблона нарушало обратную совместимость, поскольку это могло сломать существующий сайт или удалить функции, которые были добавлены самостоятельно. Это решение имело достаточно ограниченный уровень масштабируемости, например добавление поддержки

интернационализации в шаблоне требовало низкоуровневых изменений в структуре и генерации шаблона.

Существуют различные решения, которые позволяют быстро запускать проекты с веб-сайтами документации, особенно для команд, которые не имеют большого опыта в веб-разработке или хотят иметь базовый сайт для демонстрации своего проекта. При этом они могут расширять и настраивать свой сайт, но с выходом обновлений с исправлениями и новыми функциями, проектная документация может обновляться просто и без каких-либо ломающих изменений. Среди примеров таких инструментов можно отметить *GitHub Wiki*, *Confluence*. Кроме того, набирают популярность различные вики-платформы, такие как *Read the Docs* и *Wiki.js*, которые позволяют создавать совместную редактируемую документацию, аналогичную Википедии. Они обеспечивают удобный способ хранения и обновления информации, поддерживают версиюность и предоставляют возможность коллективной работы над документами, что делает их незаменимыми в процессе разработки и управления проектами [2].

В качестве графического отображения пользовательских требований используется диаграмма вариантов использования, представленная на рисунке 1.

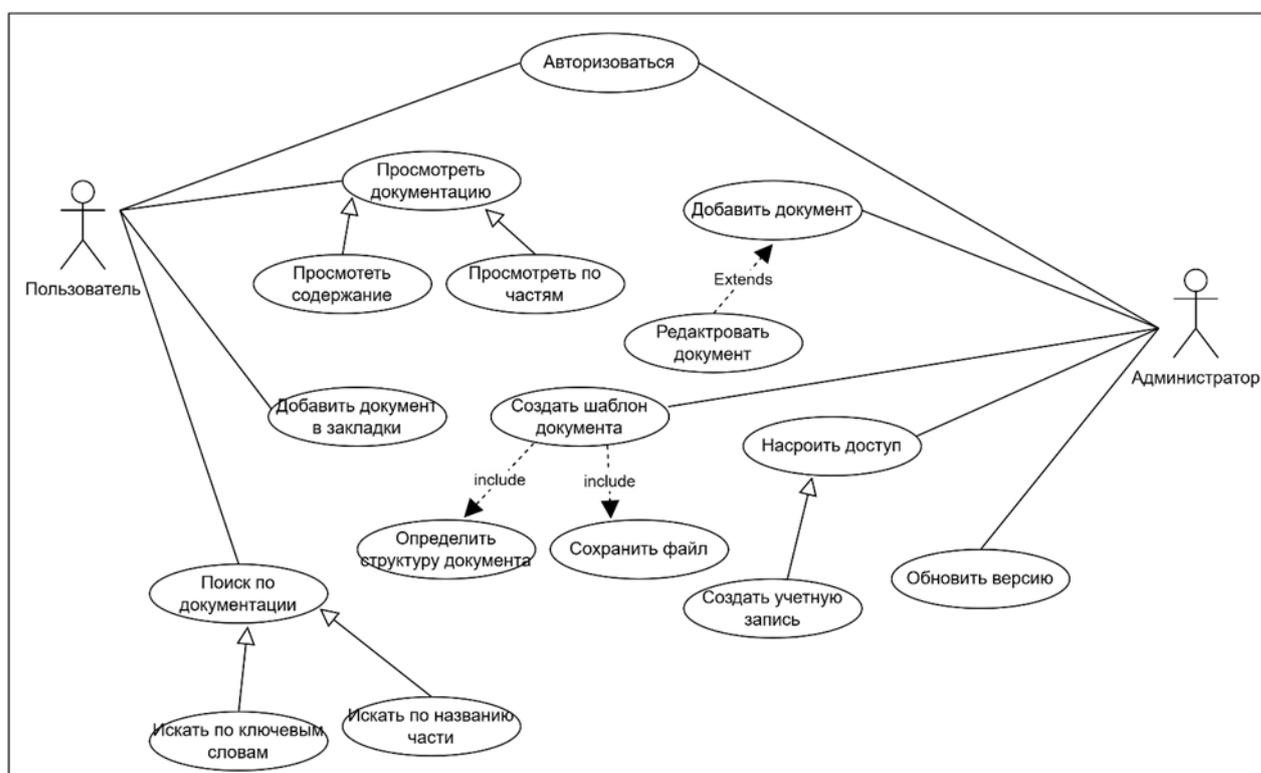


Рисунок 1 – Диаграмма вариантов использования

На диаграмме отображены две ключевые роли пользователей программного средства, одна из которых – администратор, ключевая в плане управления проектной документацией (права доступа на редактирование есть у всех, но применять изменения может только группа администраторов). Также существует роль супер-администратора. Деятельность оператора в системе зависит от выполняемой им роли. Непривилегированный пользователь имеет в системе доступ только на чтение. Он может просматривать файлы, которые представлены в качестве страниц документации, по доступным параметрам, а также выполнять поиск. Пользователь-администратор имеет базовые методы взаимодействия с системой, которые описаны для обычного пользователя, вдобавок для него доступна панель редактирования и администрирования. Администратор вводит свои данные в форму с

авторизацией для получения доступа к панели администратора, а после авторизации может редактировать и добавлять новые страницы документации.

Для поиска необходимой информации пользователь может воспользоваться встроенной строкой поиска, вводя ключевые слова. В ответ система выдает релевантный список результатов, выделяя совпадения для удобства восприятия. Если пользователь предпочитает изучать документацию структурировано, от начала и до конца, он может использовать боковое меню навигации, переходя по разделам и категориям. Каждый раздел сопровождается оглавлением и ссылками на связанные материалы, что упрощает процесс навигации.

Актуальность создания программной системы документирования информационных проектов обусловлена сложностью и избыточностью существующих систем. Не менее важным аспектом является эргономическое обеспечение системы, так как удобство работы с документацией напрямую влияет на скорость и качество взаимодействия пользователей с системой.

Заключение. Представленное программное средство предоставляет комплексный подход к управлению документацией в информационных проектах, обеспечивая централизованное хранение, анализ, мониторинг и совместное использование данных. Это позволяет командам эффективно организовывать процессы документирования, поддерживать актуальность информации, упрощать доступ к данным и улучшать взаимодействие между участниками проекта. Внедрение такой системы дает значительные преимущества: повышение продуктивности работы с документацией, минимизацию ошибок за счет структурированного хранения и автоматизированного обновления данных, а также улучшению координации внутри команды и успешной реализации проектов в условиях современных требований к качеству и скорости разработки.

Список литературы

1. *Awesome Static Web Site Generators*. [Электронный ресурс]. Режим доступа: <https://github.com/myles/awesome-static-generators/>– Дата доступа: 02.02.2025.
2. *Introducing Docusaurus*. [Электронный ресурс]. Режим доступа: <https://docusaurus.io/blog/2017/12/14/introducing-docusaurus/>– Дата доступа: 02.02.2025.

UDC 004.774:005.92

PROGRAM SYSTEM FOR DOCUMENTATION OF INFORMATION PROJECTS AND ITS ERGONOMIC SUPPORT

Konokhov A.P.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Borisik M.M. – Master of Sci., Senior Lecturer at the Department of EPE

Annotation. The article deals with the subject area, existing analogues, and peculiarities of designing a system aimed at structured storage, processing, and management of project documentation. The main tasks solved by the system are analyzed, ergonomic aspects of the interface are defined to improve its usability.

Keywords: documentation, wiki platforms, project management.