Николаев-Аксенов И.С. ВЕБ-приложение для управления паролями // Международный журнал информационных технологий и энергоэффективности. – 2023. – Т. 8 № 5(31) ч.2. с. 103–108

Международный журнал информационных технологий и энергоэффективности

Сайт журнала:
http://www.openaccessscience.ru/index.php/ijcse/

УДК 004.9

ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ УПРАВЛЕНИЯ ПАРОЛЯМИ

Николаев-Аксенов И.С.
ФГБУО ВО «МИРЭА - Российский технологический университет», Москва, Россия (119454, г. Москва, пр. Вернадского, 78), e-mail: 9frischmann@gmail.com

Данная работа направлена на создание веб-приложения для управления паролями. В ходе выполнения данной работы был произведен обзор существующих конкурентных решений, выбор инструментов и методов для создания веб-приложения, спроектирована и разработана архитектура, клиентская и серверная часть, модель жизненного цикла и схема базы данных веб-приложения для управления паролями. Результатом работы является разработанное веб-приложения для управления паролями.

Ключевые слова: Веб-приложение, управление паролями, проектирование приложения, хранение паролей, интернет.

WEB APPLICATION FOR MANAGING PASSWORDS

Nikolaev-Aksenov I.S.
MIREA - Russian Technological University, Moscow, Russia (119454, Moscow, Vernadskogo Ave., 78), e-mail: 9frischmann@gmail.com

This work is aimed at creating a web application for managing passwords. In the course of this work, a review of existing competitive solutions was made, the choice of tools and methods for creating a web application, the architecture, client and server parts, the life cycle model and the database schema of a web application for password management were designed and developed. The result of the work is the developed web application for password management.

Keywords: Web application, password management, application design, password storage, interne.

Введение

В настоящее время появляется все больше сайтов, на которых пользователям необходимо регистрироваться, для предотвращения угрозы взлома учетных записей рекомендуется не повторять пароли при регистрации на разных сайтах, часто обновлять пароли, использовать пароли длиной более 8 символов [1].

Также рекомендуется не использовать пароли, которые легко угадать, по статистике пользователи используют чаще всего следующие пароли в 2023 году [2]: «123456», «123456789», «qwerty», «password», «12345», «qwerty123», «1q2w3e», «12345678», «111111», «1234567890».

В 33% пользователи используют клички своих питомцев, в 22% свое имя, в 15% имя своего партнера и в 14% случаев имя своего ребенка [3]. Чаще всего люди используют пароли размером 8 и 6 символов [2].

Но на практике этими правилами зачастую пренебрегают, так как запомнить множество комбинаций паролей представляется невозможным. Так как сложный для взлома пароль должен иметь как минимум 10 символов, из них как один символ верхнего регистра, цифру и специальный символ [4].

Для решения этой проблемы были созданы менеджеры паролей – это программное обеспечение, которое позволяет хранить пароли в одном месте, для доступа к ним необходимо лишь придумать мастер-пароль.

Программное обеспечение для управления паролями делится на три основных категории:

- установленные на ПК или мобильное устройство – представляют собой программное обеспечение с локальной базой данных, зачастую не имеют выхода в Интернет;
- переносные устройства – это некое устройство, в основном USB-флеш-накопитель, на котором имеется ПО для управления базой данных паролей;
- облачные сервисы – эта категория, в которой база данных паролей находится на внешнем сервере, доступ к ней предоставляется пользователю посредством сети Интернет. Эта категория является наиболее удобной для конечного пользователя, так как не требует дополнительных настроек среды выполнения, а также является более надежной, так как база данных паролей находится не у конечного пользователя, а значит в случае неисправностей у него останется доступ ко всем его паролям.

В рамках данной работы сфокусируемся на создании менеджера паролей последней категории, так как она является наиболее распространенной, это и будет целью работы.

Обзор существующих конкурентных решений

Выделим для обзора 7 существующих конкурентных решений:

- KeePassXC – бесплатная программа менеджер паролей с открытым исходном кодом, является ответвлением программы KeePass с добавлением библиотек Qt5 для достижения кроссплатформенности и предания более современного вида. Использует в качестве базы данных зашифрованный файл в расширении kdbx;
- LastPass – условно-бесплатная программа для хранения паролей, разработанная компанией LastPass, пароли хранятся в «облаке» и могут быть синхронизованы между устройствами;
- Bitwarden – менеджер паролей с открытым исходном кодом, использует для сохранения данных облачный сервис, также есть возможность развертывания решения локально;
- 1Password – программа для хранения паролей разработанная AgileBits Inc. Предоставляет возможность хранить различные пароли, данные банковских карт и т.д;

- Kaspersky Password Manager – инструмент управления учетными записями в интернете и приложениях от Лаборатории Касперского. Вся информация хранится в специальной базе данных на компьютере в зашифрованном виде;
- Zoho Vault – веб-приложение для управления паролями с закрытым исходном кодом;
- Dashlane Password Manager – сервис представляющий доступ к своим услугам хранения паролей по подписке.

Выбор инструментов и методов создания веб-приложения

Для реализации серверной части веб-приложение выберем строго типизированный объектно-ориентированный язык программирования общего назначения Java, универсальный фреймворк с открытым исходным кодом Spring.

Разрабатывать клиентскую часть будем на языке программирования TypeScript, данный язык расширяет возможности JavaScript и предоставляет возможность явного статического назначения типов, что должно повысить скорость разработки, облегчить читаемость кода, рефакторинг и т.д. Также будем использовать фреймворк Next.js, данный инструмент позволяет разрабатывать приложения на основе React с Server Side Rendering, а также генерировать статические вебсайты.

В качестве базы данных выберем свободную объектно-реляционную систему управления базами данных PostgreSQL.

Для контейнеризации приложения будем использовать Docker. Это программное обеспечение для автоматизации развертывания и управления приложениями в средах с поддержкой контейнеризации.

Проектирование и разработка веб-приложения

В качестве архитектуры веб-приложения был выбран паттерн MVC (Model-View-Controller).

В качестве модели жизненного цикла веб-приложения была выбрана каскадная модель. В рамках данной модели процесса разработки программного обеспечения, жизненный цикл выглядит как поток, последовательно проходящий фазы анализа требований, проектирования, реализации, тестирования, интеграции и поддержки. Данная модель является наиболее предпочтительной в виду наличия стабильности требований в течение всего жизненного цикла разработки, определенности и понятности шагов модели и простоты ее применения, этапы работ выполняются в логической последовательности и позволяют планировать сроки завершения всех работ.

В клиентской части веб-приложения реализуем следующие страницы: главная страница, страница регистрации, страница аутентификации, страница профиля пользователя. В качестве примера приведем реализацию страницы профиля пользователя на Рисунке 1.

Рисунок 1 – Демонстрация страницы профиля пользователя

На стороне сервера следует создать конечные точки интерфейса RESTful API. Данные точки должны позволять добавлять, изменять, удалять, обмениваться сущностями. Также нужно предусмотреть точки получения как всех сущностей, так и по идентификационному номеру в базе данных.

В базе данных создадим 9 таблиц: таблица пользователей, таблица сохраненных платежных карт, таблица сохраненных паролей, таблица сохраненных адресов, таблица сохраненных заметок, также создадим 4 вспомогательные таблицы, в которых будет идентификационный номер сущности и список электронных адресов пользователей, которые имеют доступ к этой сущности. Приведем схему базы данных в качестве демонстрации на Рисунке 2.

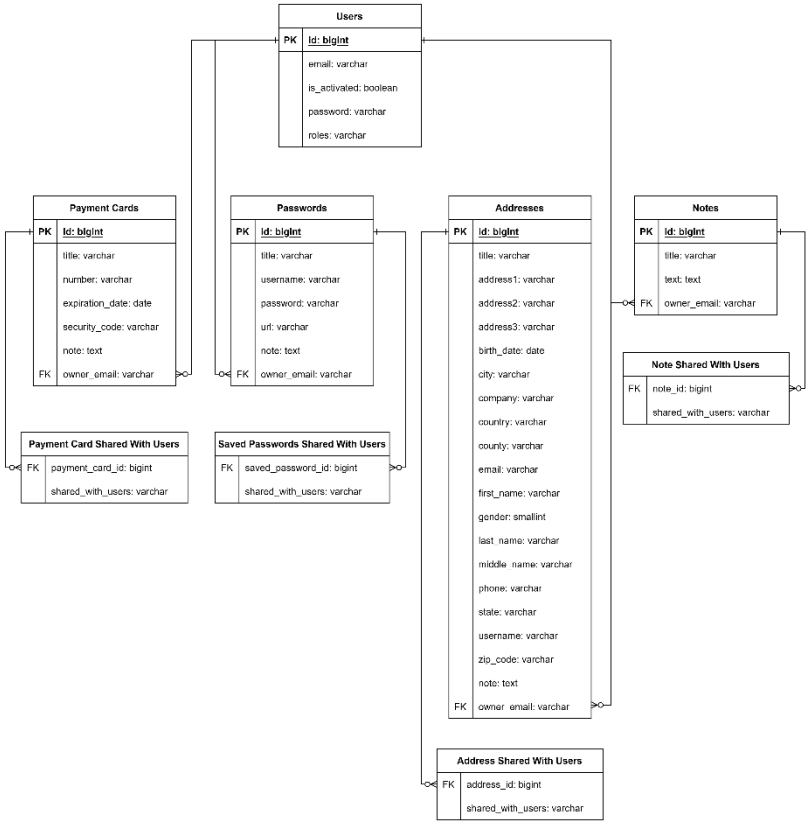

Рисунок 2 – Демонстрация схемы базы данных

Заключение

В ходе выполнения данной работы был проведен обзор существующих конкурентных решений, выбор инструментов и методов создания веб-приложения, выполнено проектирование веб-приложения. В результате произведенных действий удалось разработать веб-приложение для управления паролями.

Список литературы

1. 7 Bad Password Habits to Break Now : сайт. — URL: https://blog.lastpass.com/2021/01/7-bad-password-habits-to-break-now-2/ (дата обращения: 19.05.2023).
2. Most common passwords: latest 2023 statistics : сайт. — URL: https://cybernews.com/best-password-managers/most-common-passwords/ (дата обращения: 19.05.2023).
3. The United States of P@ssw0rd : сайт. — URL: https://storage.googleapis.com/gweb-uniblog-publish-prod/documents/PasswordCheckup-HarrisPoll-InfographicFINAL.pdf (дата обращения: 19.05.2023).
4. How Safe Is Your Password? : сайт. — URL: https://www.statista.com/chart/26298/time-it-would-take-a-computer-to-crack-a-password/ (дата обращения: 19.05.2023).

References

1. 7 Bad Password Habits to Break Now : сайт. — URL: https://blog.lastpass.com/2021/01/7-bad-password-habits-to-break-now-2/ (Accessed on 19.05.2023).
2. Most common passwords: latest 2023 statistics : сайт. — URL: https://cybernews.com/best-password-managers/most-common-passwords/ (Accessed on 19.05.2023).

3. The United States of P@ssw0rd : сайт. — URL: https://storage.googleapis.com/gweb-uniblog-publish-prod/documents/PasswordCheckup-HarrisPoll-InfographicFINAL.pdf (Accessed on 19.05.2023).
4. How Safe Is Your Password? : сайт. — URL: https://www.statista.com/chart/26298/time-it-would-take-a-computer-to-crack-a-password/ (Accessed on 19.05.2023).