Реферат

Бакалаврская работа содержит X листов, X рисунков X таблиц, X источников, X приложений.

Цель работы – создание приложения для хранения паролей. Приложение должно обеспечивать безопасное хранение паролей, шифрование и расшифровку паролей с использованием криптографических алгоритмов, удобный пользовательский интерфейс для добавления, просмотра и удаления паролей.

Объектом разработки является приложение для управления персональными паролями. Приложение должно обладать графическим пользовательским интерфейсом и позволять пользователю добавлять, редактировать и удалять пароли, а также хранить их в зашифрованном виде с использованием алгоритма AES-256.

Разработка проводилась в среде программирования Visual Studio 2022. В качестве языка программирования был выбран C#.

В результате проведенной работы разработано приложение, удовлетворяющее поставленной задаче, а именно: удобный пользовательский интерфейс, возможность взаимодействия с базой данных паролей, автоматическое генерирование паролей, экспорт и импорт паролей в json-формате.

Содержание

Перечень сокращений

Введение

В настоящее время разработка программного обеспечения является важным элементом современного мира. С каждым днем число пользователей, использующих интернет-сервисы, увеличивается, и вместе с ним растут их требования к качеству и безопасности обработки и хранения персональных данных. Одним из наиболее важных аспектов безопасности является сохранение конфиденциальности паролей пользователей.

В данной работе представлено приложение, разработанное для безопасного хранения паролей рядового пользователя. Оно позволяет сохранять пароли в зашифрованном виде и предоставляет возможность удобного редактирования, добавления и удаления сохраненных паролей. Приложение разработано для обеспечения безопасности пользовательских данных и удобства их использования.

Добавить про угрозу безопасности,разные атаки. В общем про ин без. требования

Актуальность темы

Актуальность данной темы обусловлена тем, что в настоящее время важно обеспечить безопасность личной информации и конфиденциальных данных. Многие из пользователей сталкиваются с проблемой запоминания множества сложных паролей для каждого сервиса, и поэтому для ставят один и тот же пароль, что в разы облегчает задачу злоумышленнику, если он захочет получить контроль над каким-либо аккаунтом.

Если пользователь хочет обеспечить безопасность своих аккаунтов в сети, ему необходимо использовать разные пароли к разным аккаунтам. Ставится вопрос о хранении данных аккаунта где-либо и о фантазии придумывания стойких паролей. Есть несколько способов хранения.

Можно попросту писать пароль в блокнотик. В таком случае, около компьютера пользователя будет лежать блокнот, в котором хаотично записаны какие-либо аккаунты, отдельные пароли, номера телефонов вперемешку с записями. Также однажды он может потеряться, и в таком случае необходимо будет заново восстанавливать доступ ко всем записанным туда аккаунтам.

Можно хранить в текстовом файле на компьютере. Этот файлик всегда под рукой, его не нужно искать, и можно просто скопировать с какой-либо страницы твой новый логин-пароль, вставить его в «текстовик» и сохранить. К сожалению, это более опасный вариант, так как если к твоему компьютеру получат доступ, то можно попросту скопировать пароли, или сделать скриншот.

Можно обратиться к существующим сейчас менеджерам паролей. К сожалению, для рядового пользователя компьютера чаще всего даже неизвестно, что такие существуют. Имеющиеся на рынке менеджеры паролей зачастую являются либо платными, либо требуют особые знания и навыки в использовании компьютера, что делает их сложными для рядового пользователя. Поэтому существует потребность в разработке простого и удобного в использовании приложения для хранения паролей.

Благодаря применению современных алгоритмов шифрования, сохраненные пароли остаются надежно защищенными от несанкционированного доступа. Кроме того, приложение обладает простым и интуитивно понятным пользовательским интерфейсом, что делает его использование максимально удобным.

Описать что менеджеры паролей хорошие

Анализ существующих аналогов

Pass - это приложение для хранения паролей и других конфиденциальных данных в зашифрованном виде. Оно основано на командной строке, и позволяет хранить пароли в виде текстовых файлов, зашифрованных с помощью GPG.

Среди преимуществ приложения Pass можно отметить его открытый исходный код, что позволяет пользователям проверять безопасность кода и убедиться в его надежности. Кроме того, приложение не имеет облачного хранения, что уменьшает риски взлома и утечки данных.

Однако, из-за того, что приложение основано на командной строке, оно может показаться сложным для пользователей, не знакомых с этим способом работы. Кроме того, приложение не предоставляет удобный графический интерфейс, что может быть неудобно для пользователей, предпочитающих визуальную навигацию.

В целом, приложение Pass хороший выбор для опытных пользователей, которые предпочитают командную строку и ценят безопасность данных. Однако, для новичков и тех, кто ищет более удобный и простой способ хранения паролей, могут быть лучшие альтернативы с более простым и удобным интерфейсом.

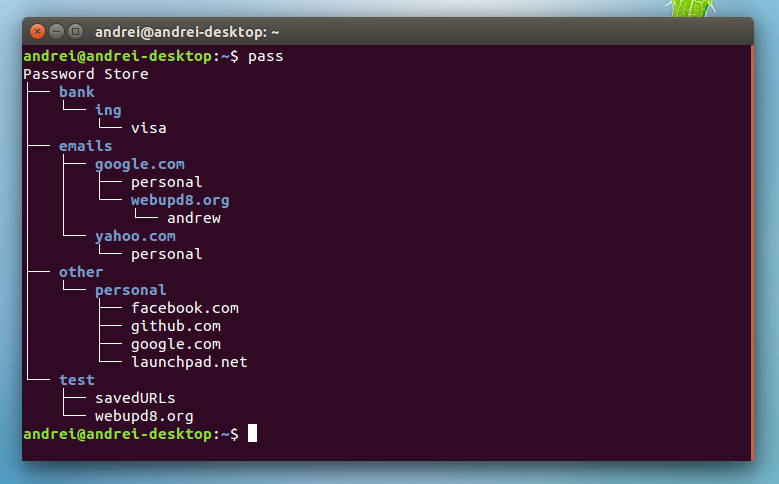


Рисунок 1 – Интерфейс программы Pass

KeePass - это свободное и бесплатное приложение для управления паролями, которое позволяет сохранять и защищать пароли пользователей в зашифрованном виде. KeePass поддерживает множество функций, включая автоматическое заполнение паролей, возможность генерации паролей и привязки к определенным URL-адресам.

KeePass также предоставляет возможность хранения других конфиденциальных данных, таких как номера кредитных карт, PIN-коды и т.д. Все данные хранятся в одном зашифрованном файле, который защищен мастер-паролем.

Среди основных преимуществ KeePass можно отметить его свободное распространение и открытый исходный код, что позволяет пользователям проверять его безопасность и вносить свои улучшения. KeePass также поддерживает множество дополнительных плагинов, которые расширяют его функциональность и позволяют настроить его под конкретные потребности пользователя.

Среди недостатков KeePass можно отметить его сложность в использовании для новичков, особенно в настройке и синхронизации между несколькими устройствами. Кроме того, управление множеством паролей может быть трудным для пользователей без определенных навыков. KeePass – хороший вариант для опытных пользователей, которые не страшатся громоздких интерфейсов, долгой настройки и перегруженности функций.

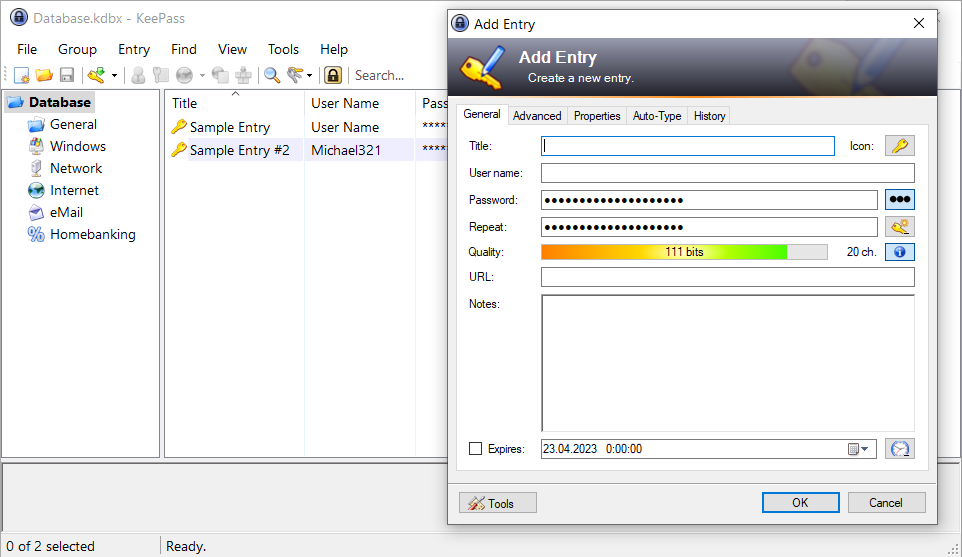


Рисунок 2 – Интерфейс программы KeePass

Доабвить табличку в анализ про преимущества и недостатки добавить еще аналоги

Постановка задачи

Требуется разработать приложение, которое должно соответствовать следующим целям:

– удобный и интуитивно понятный графический интерфейс для работы пользователя с приложением;

– Безопасное хранение паролей с использованием современных криптографических алгоритмов;

– Импорт и экспорт базы данных паролей;

–Возможность генерирования новых паролей;

–Добавление записей в базу данных паролей;

–Возможность редактирования записей в базе данных;

–Возможность добавления дополнительной информации к паролю (логин, название аккаунта, URL сайта, привязанный номер телефона и пр.);

–Блокирование авторизации в программе при многократных неудачных попытках входа;

Прописывается функционал, и требования. Требования к технической части, какие ресурсы (процессор операционка, обьем памяти и тп.)

Тех часть и функционал. Конкретный функционал(почему многооконный), автономное приложение.

Выбор языка программирования

Язык программирования *C++*

*C++* – объектно-ориентированный, компилируемый язык программирования. *C++* гораздо ближе к низкоуровневому ассемблеру, чем другие объектно-ориентированные языки, Но С++ не является достаточно гибким для создания графического интерфейса.

Язык программирования *Java*

*Java* является главным конкурентом языка C#. Этот язык прост в использовании, является кроссплатформенным, а также имеет гибкие настройки для создания пользовательских интерфейсов. К сожалению, бесплатные среды разработки на этом языке менее удобны, чем среда Visual studio.

Язык программирования *Python*

Python - это интерпретируемый язык программирования с динамической типизацией, который широко используется в различных областях, включая науку о данных, машинное обучение, веб-разработку, научные вычисления и другие.

Он предоставляет широкий набор стандартных библиотек и фреймворков, что делает его очень гибким и удобным для разработки быстрых прототипов и решений. Также Python имеет чистый и простой синтаксис, что делает его доступным для новичков.

Несмотря на множество преимуществ, у Python также есть недостатки. В частности, он является интерпретируемым языком, что означает, что он может работать медленнее, чем компилируемые языки. Кроме того, он может иметь проблемы с масштабируемостью, особенно при работе с большими объемами данных.

Язык программирования *C*#

C# разрабатывался Microsoft и является одним из основных языков для разработки приложений под платформу Windows. Он полностью интегрирован с .NET Framework, что позволяет создавать мощные и эффективные приложения.

Этот язык является типизированным языком, что позволяет выявлять ошибки на стадии компиляции. Это позволяет создавать более надежные и безопасные программы.

Одной из главных причин, почему C# является отличным выбором для разработки приложений, является удобство создания графических интерфейсов. С помощью Windows Forms и WPF (Windows Presentation Foundation) можно создавать сложные и красивые пользовательские интерфейсы с минимальным количеством усилий.

Visual Studio - это мощная и широко используемая интегрированная среда разработки, которая поддерживает C# и многие другие языки программирования. Эта среда обладает множеством инструментов и функций, которые делают процесс разработки более удобным и быстрым. Для студентов и начинающих разработчиков Visual Studio доступна бесплатно в версии Community, что делает C# еще более привлекательным выбором.