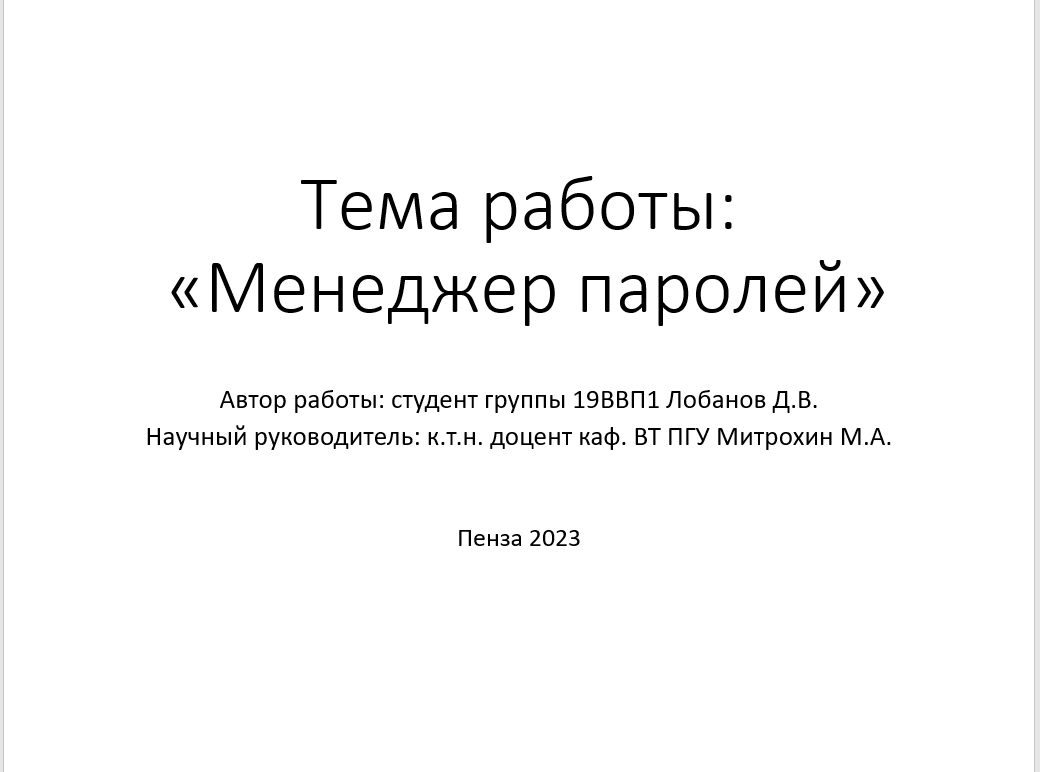
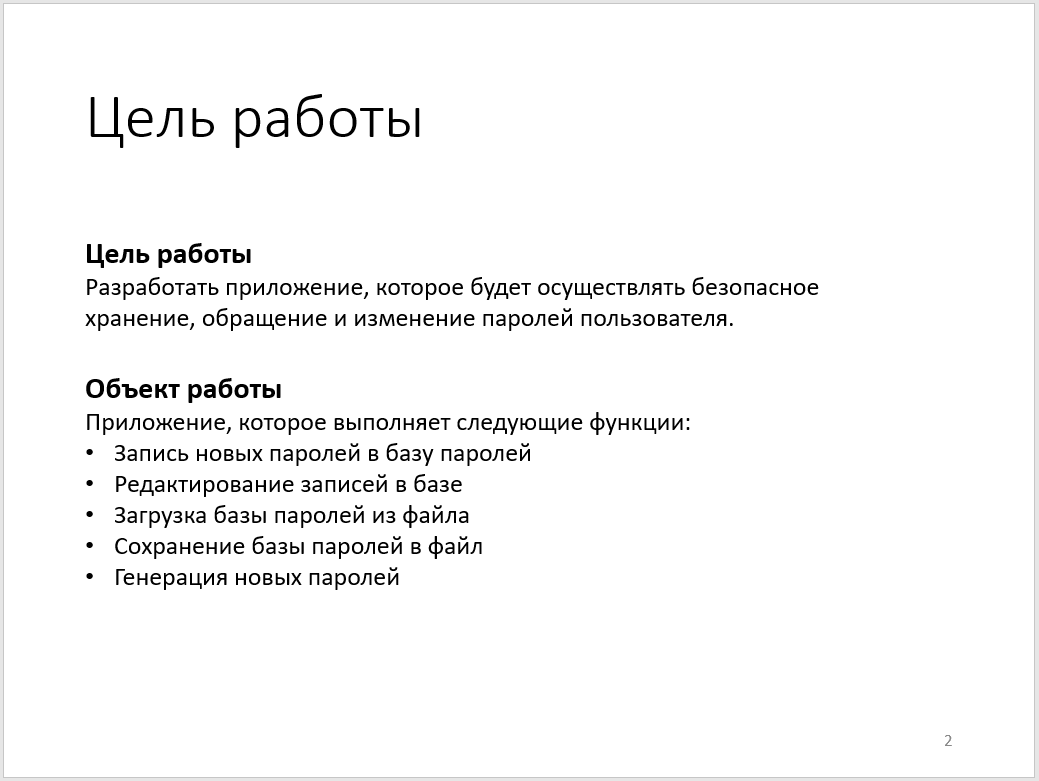
**1 слайд**

****

Темой работы является:

«Менеджер паролей»

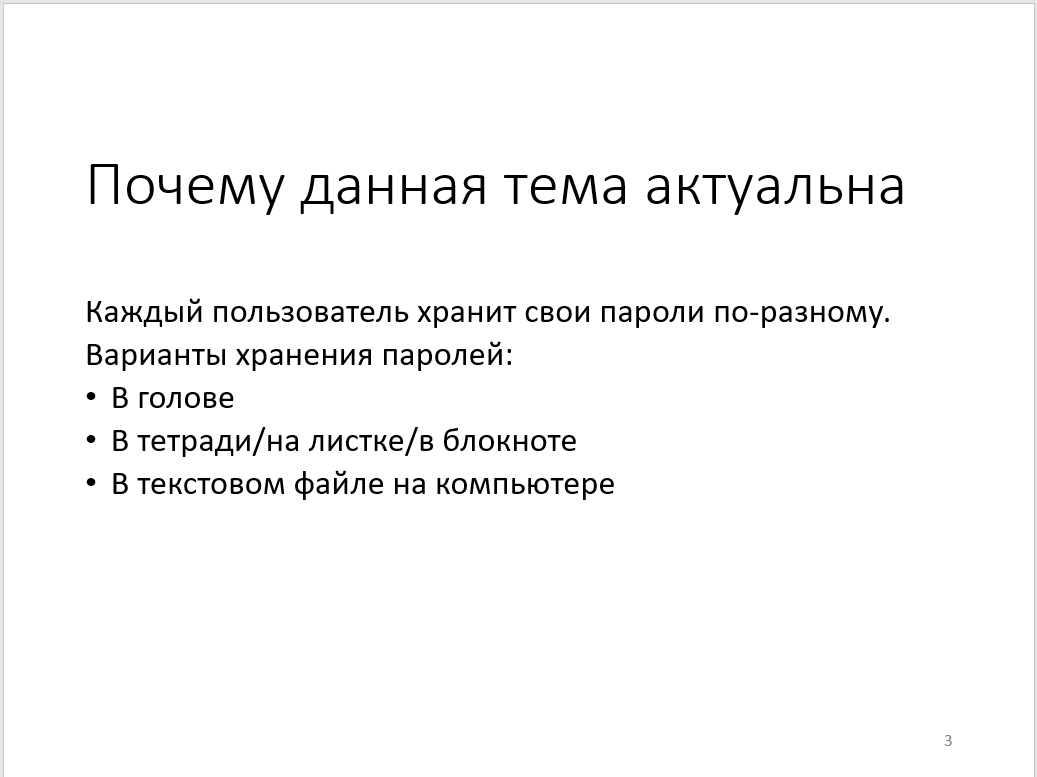
**2 слайд**

****

**(прочитать цель со слайда)**

Одним из наиболее важных аспектов безопасности аккаунта является сохранение конфиденциальности паролей пользователей.

**3 слайд**

****

Каждый пользователь компьютера рано или поздно сталкивается с проблемой хранения паролей, и каждый решает её по-разному.

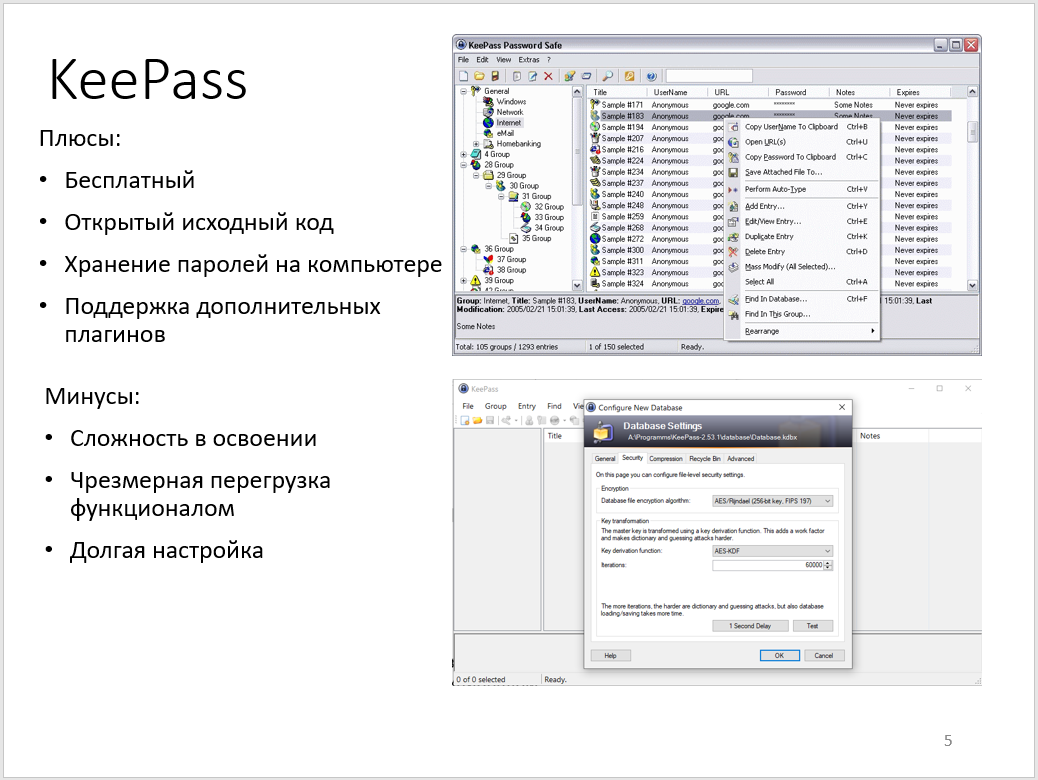
Одни используют простые и запоминаемые пароли, лишая себя бремени самого хранения, другие записывают пароли на бумаге, третьи хранят свои пароли в текстовых файлах.

На помощь приходит менеджер паролей.

**4 слайд**

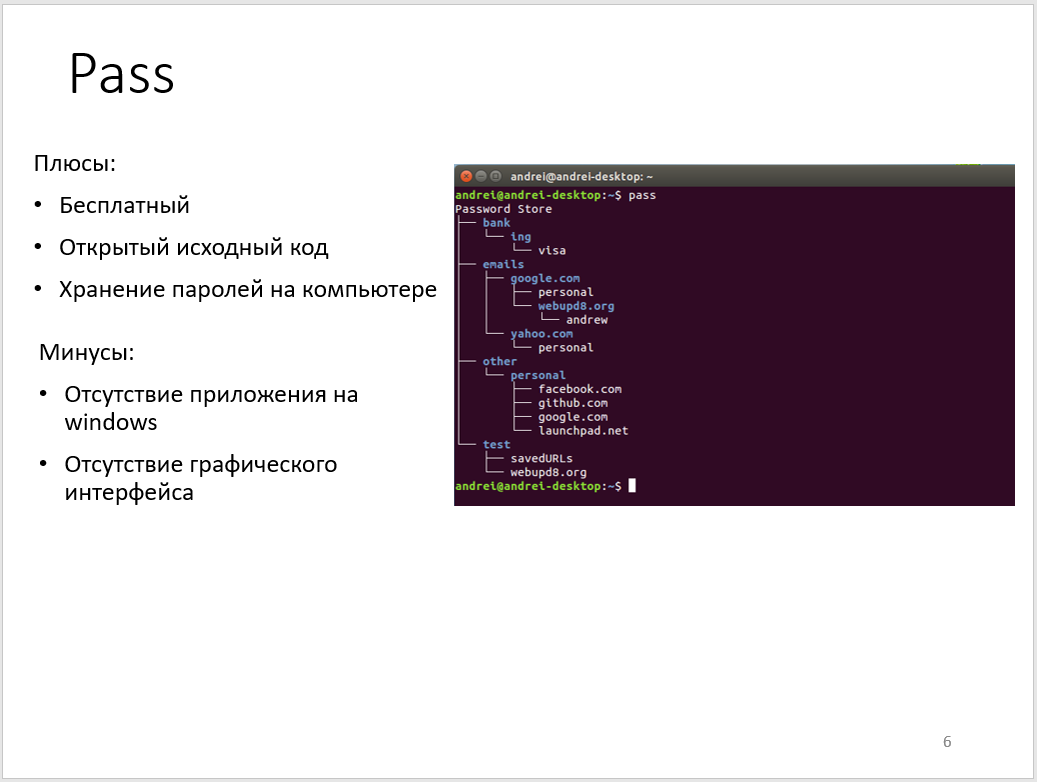
Аналоги

**5 слайд**

****

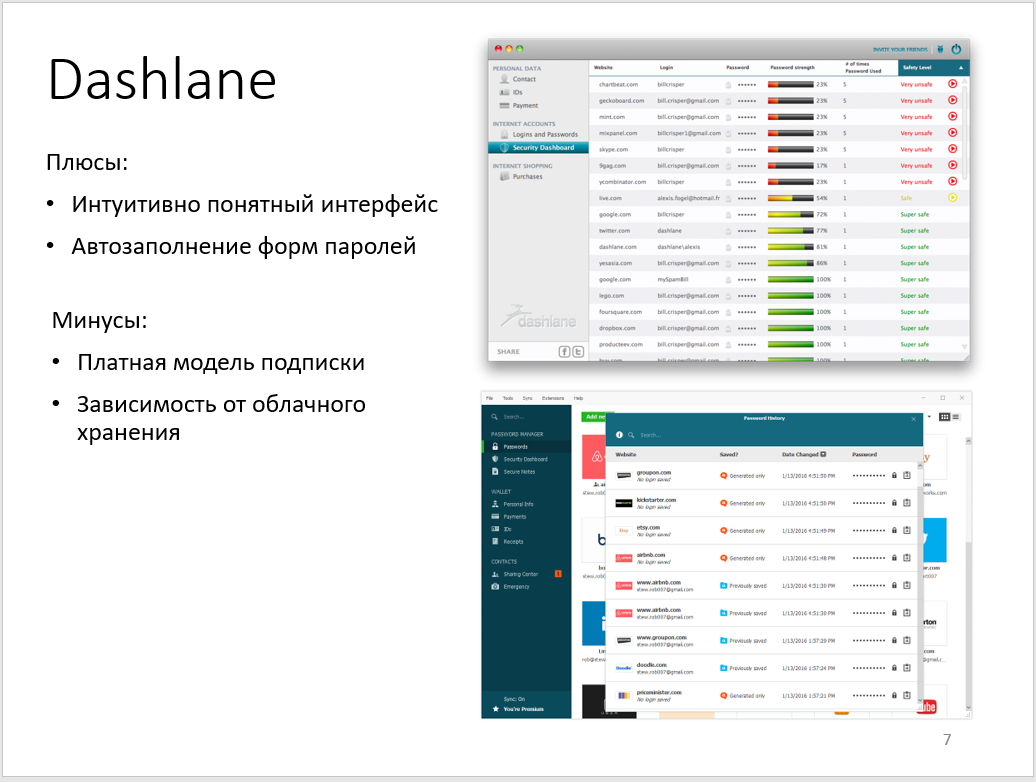
KeePass – популярный менеджер паролей. Из плюсов он бесплатный, у него есть много функций, он локально хранит файлы.

**6 слайд**

****

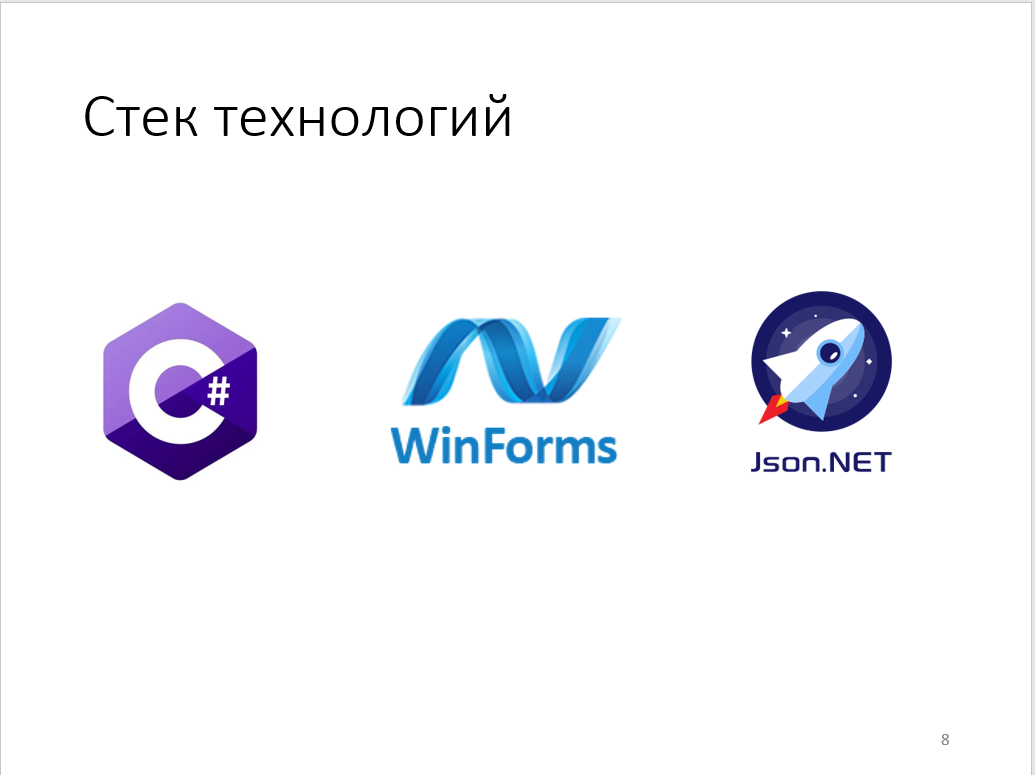
Pass – стандартное хранилище паролей для Unix –подобных систем. Как видно, приложение не предоставляет удобный графический интерфейс.

**7 слайд**

****

Dashlane обладает интуитивно понятным интерфейсом. Если его использовать, доступ к паролям без интернета будет закрыт. Также он является платным, а следовательно, имеет закрытый исходный код.

**8 слайд**.



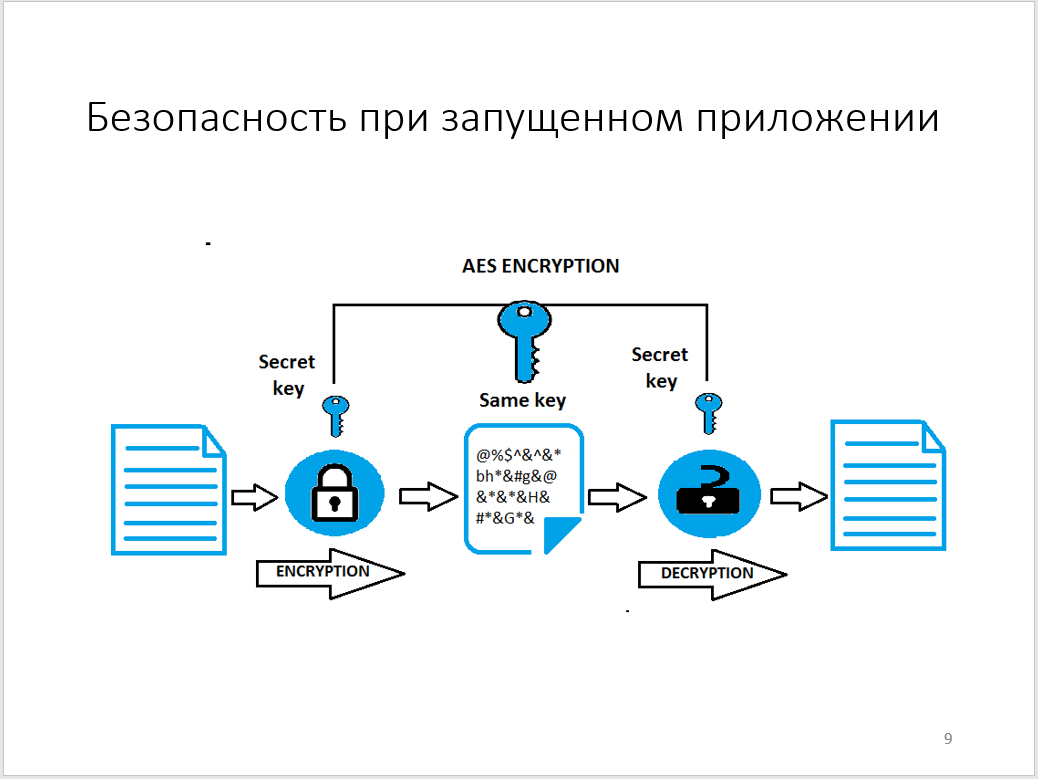
В основном я использовал стандартные инструменты, предоставленные в Visual Studio.

С# и его стандатные библиотеки предоставляют огромный инструментарий, позволяющий шифровать данные.

Windows forms позволяет создавать простые приложения с привычным пользователю интерфейсом.

Newtonsoft.Json использовался для работы с файлами.

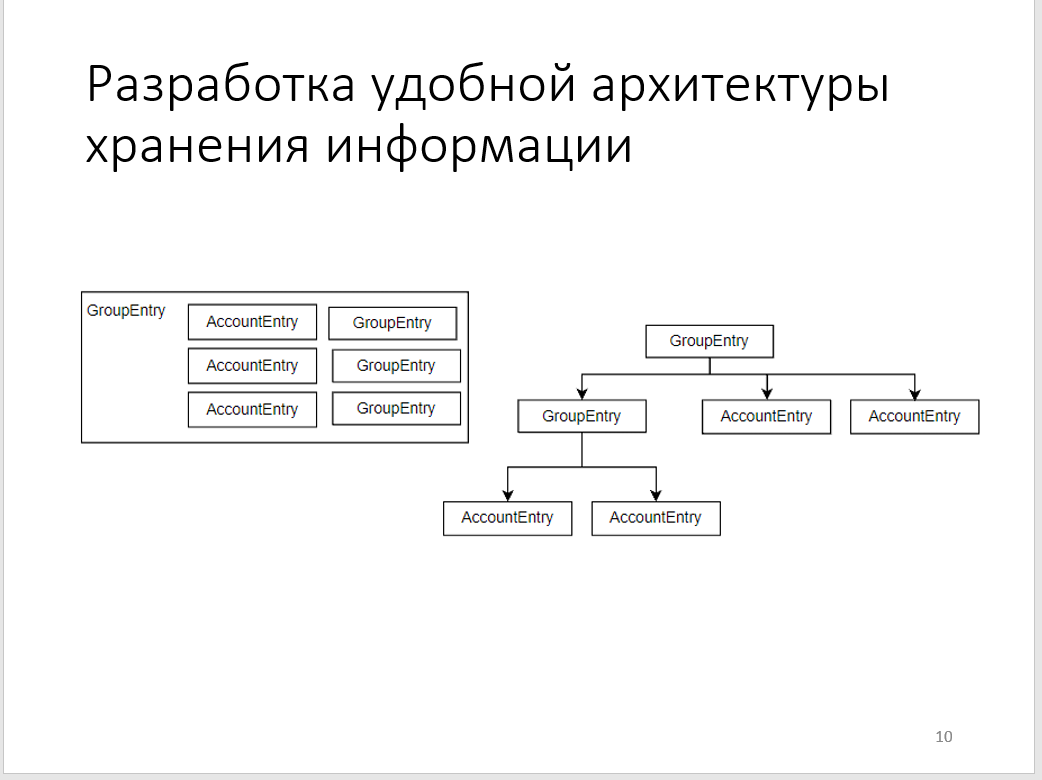
**9 слайд.**

****

Для защиты паролей был реализован класс Password.

при создании экземпляра класса с помощью алгоритма AES и ключа пользователя шифруется пароль. При запущенном приложении, поля обьекта защищены с помощью технологии ProtectedMemory, импортируемая стандартной системной библиотекой System.Security.Cryptography. Реализовано копирование в буфер обмена на ограниченное время (10 секунд).

**10 слайд.**

****

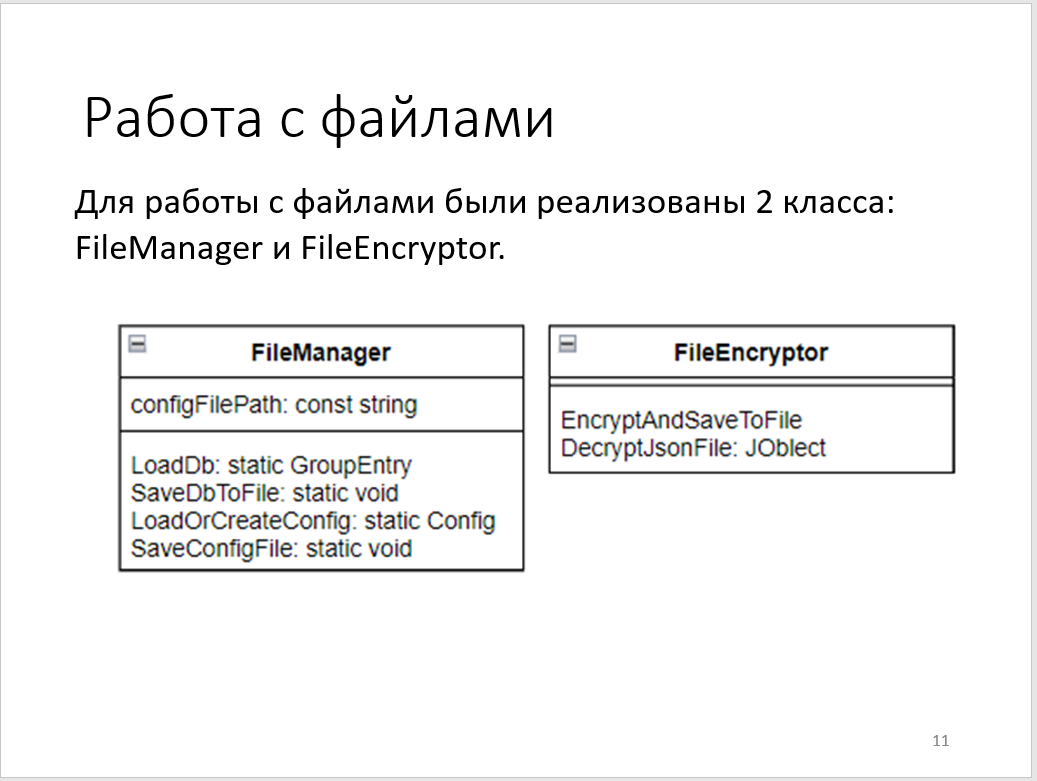
Один пароль для пользователя ничего не значит. К нему необходима дополнительная инфомация, такая как: Название аккаунта, логин, телефон, секретный вопрос и т.д.

Для решения этой задачи мною был написан класс AccountEntry. В нем есть поля Login, Password, AccountName, и список дополнительных полей в Dictonary.

Записи аккаунта тем временем группируются в GroupEntry.

GroupEntry – это класс, который представляет собой древовидную структуру данных, в которой содержится List таких же GroupEntry, и List AccountEntry.

**11 слайд.**

****

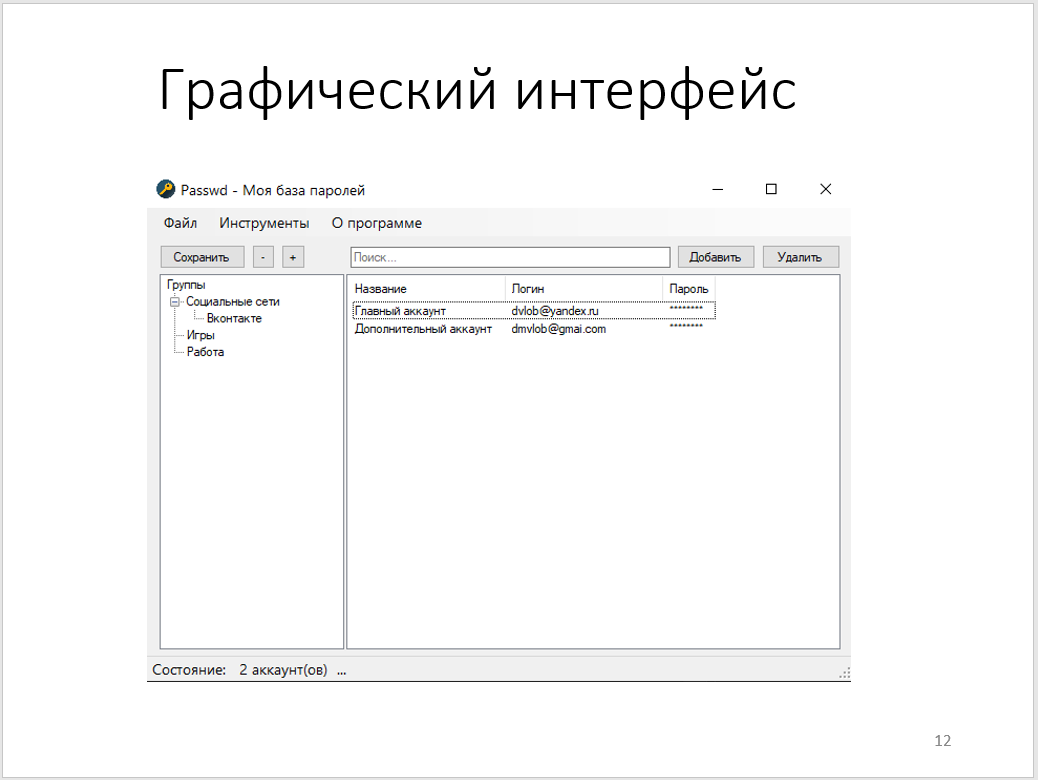
Данное решение позволило удобно экспортировать данные в файл, так как Newtonsoft.Json позоволяет гиибко парсить обьекты в строку.

Для работы с файлами были реализованы 2 класса: FileManager и FileEncryptor.

Эти 2 класса являются статическими, что облегчает процесс создания кода. Нет необходимости создавать экземпляры классов, чтобы использовать только один метод.

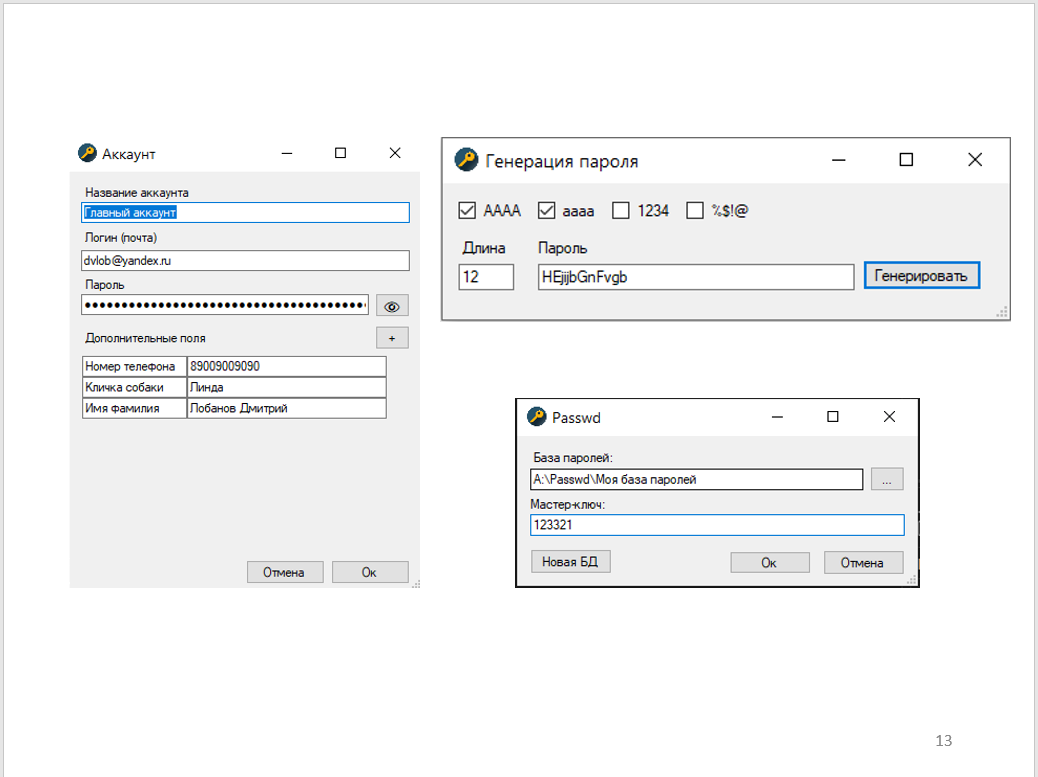
В FileManager передается корневая запись GroupEntry, которая парсится с помощью Newtonsoft.Json в строку. Далее эта строка шифруется алгоритмом AES, и записывается в файл.

**12 слайд.**

****

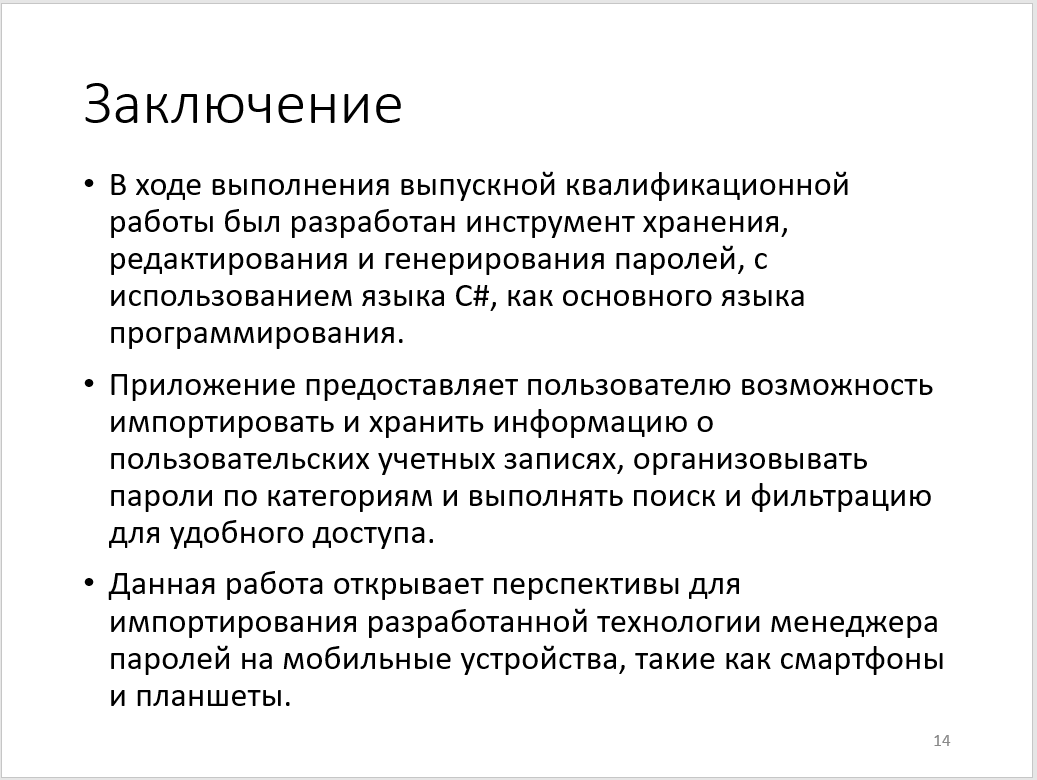
**Был разработан графический интерфейс в соответствии с тем, что пользователь будет делать чаще всего. А именно добавление, удаление, редактирование и обращение к аккаунту.** Для обращения к информации пользователя необходимо кликнуть правой кнопкой мыши на аккаунт, и появится форма, предоставляющая полную информацию об аккаунте. При двойном клике пароль копируется в буфер обмена на 10 секунд.

**13 слайд.**

****

На данном слайде представлена форма, показывающая полная информация об аккаунте. простой инструмент генерации пароля и форма входа. Всё это постигаемо без сложностей, интуитивно понятно и не требует особых знаний в сфере информационных технологий.

**14 слайд**

****

В ходе выполнения выпускной квалификационной работы был разработан инструмент хранения, редактирования и генерирования паролей, с использованием языка C#, как основного языка программирования.

Приложение предоставляет пользователю возможность импортировать и хранить информацию о пользовательских учетных записях, организовывать пароли по категориям и выполнять поиск и фильтрацию для удобного доступа.

Данная работа открывает перспективы для импортирования разработанной технологии менеджера паролей на мобильные устройства, такие как смартфоны и планшеты. Это позволит пользователям удобно и безопасно управлять своими паролями в любое время и в любом месте. Приложение для мобильных устройств может предоставлять аналогичные функции хранения, генерации и управления паролями, а также синхронизировать данные с версией для настольных компьютеров.