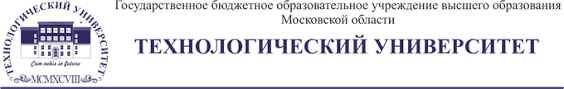
****

**Колледж космического машиностроения и технологий**

**Текстовое сопровождение к презентации**

**Предмет: МДК.01.02 Прикладное программирование**

**Тема: Разработка системы классов для приложения «Склад»**

**Выполнил:**

**Студент 3 курса**

**Группы П2-19**

**Подъяблонский Виталий Андреевич**

**Преподаватель:**

**Гусятинер Леонид Борисович**

Королев

2022 г.

# **Слайд 1**

Здравствуйте, уважаемые коллеги и гости. Меня зовут Подъяблонский Виталий. Я представляю проект на тему «Разработка системы классов для приложения «Склад».

Целью работы является разработка приложения для ОС Windows, которое будет выступать вспомогательным инструментов для работы с базой данных.

# **Слайд 2**

Технологический стек, т. е. набор языков программирования, фреймворков и ПО, необходимых для разработки приложения, представлен на слайде.

Python – универсальный и самый популярный язык программирования в мире общего назначения, который широко применяется в различных областях: от создания банальных веб-страниц до систем управления роверами на других планетах.

PyQt – библиотека Python для создания приложений с графическим интерфейсом с помощью инструментария Qt.

Qt – популярная среда для написания приложений с помощью графического интерфейса для всех основных настольных, мобильных и встраиваемых платформ.

SQLite – быстрая и легкая встраиваемая СУБД, которая не имеет сервера и позволяет хранить всю базу локально на одном устройстве.

# **Слайд 3**

Диаграмма прецедентов – схема, на которой изображаются отношения между актерами (действующими лицами) и вариантами использования (прецедентами).

# **Слайд 4**

В проекте реализован единый тип пользователя, который наделен всеми возможным правами. Он может добавлять, удалять, выводить и изменять данные. Саму диаграмму мы можем наблюдать на слайде.

# **Слайд 5**

Сценарий проекта – определенное сочетание условий, которые приводят к определенным результатам.

# **Слайд 6**

Исходя из диаграммы на слайде можно увидеть, что первичный запуск приложения наделяет пользователя всем функционаом:

# **Слайд 7**

Диаграмма классов – схема, на которой изображается внутренняя структура программы в виде классов и связей между ними.

# **Слайд 8**

Класс «database» в правой части слайда содержит в себе реализацию работы с базой данных: подключение, создание таблиц, запросы;

Группа классов в левой части слайда, объединенных в модуль «models», описывает все модели таблиц, использующихся в программе. Под описыванием понимается название таблицы и ее поля со всеми настройками.

# **Слайд 9**

Раздел «Графический интерфейс», который представлен на слайде, является одним из самых трудоемких и включает в себя модули:

* инициализирующие окна и страницы приложения;
* реализующие логику взаимодействия пользователя с интерфейсом.

# **Слайд 10**

Вспомогательные модули, представленные на слайде, делятся по типу их принадлежности:

* модуль «Excel» отвечает за работу с одноименной программой;
* styles – стили для форматирования текста в «MS Excel»;
* utils – методы, в равной степени используемые во всей программе в целом: вывод в файл с расширением json и т. д.

# **Слайд 11**

Раздел «Константные файлы», представленный на слайде, состоит из одного модуля:

* модуль «config» хранит списки и словари, необходимые в программе.

# **Слайд 12**

В связи с тем, что для большинства пользователей окна являются самым удобным способом взаимодействия с компьютером, мой проект, как и многие современные графические интерфейсы, построен на основе окон.

Последующие слайды будут отражать вид и краткий функционал окон, присутствующих в графическом интерфейсе.

# **Слайд 13**

Главное окно – окно, которое пользователь встречает при запуске приложения. Из этого окна доступен весь функционал интерфейса:

1. Открытие таблиц – представление базы данных в табличном виде;
2. Добавление элементов – занесение элементов в базу в выбранную таблицу;
3. Удаление элементов – стирание элементов из выбранной таблицы без возможности восстановления;
4. Изменение элементов – возможность изменения значения любого поля, присутствующего в выбранной таблице;
5. Вывод в файл – представление данных из базы во второстепенном файле.

# **Слайд 14**

На слайде представлено окно добавления в таблицу.

Это окно может меняться в визуальном смысле, т. е. подстраиваться под поля, присутствующие в таблице.

# **Слайд 15**

Изменение данных в базе происходит в 2 этапа. Первый этап заключается в выборах уникального идентификатора записи в таблице и поля, значение которого необходимо изменить.

# **Слайд 16**

Второй этап предоставляет возможность изменения значения, которое хранится под выбранным идентификатором и полем.

# **Слайд 17**

Вся работа с базой данных в интерфейсе построена на идентификаторах. Удаление не является исключением. Для того, чтобы стереть запись, необходимо указать ее идентификатор, после чего нажать на кнопку «Удалить».

# **Слайд 18**

Графический интерфейс предоставляет возможность вывода в два типа файлов:

1. Excel;
2. Json.

Помимо выбора типа выходного файла, есть возможность выбора типа выводимой информации.

# **Слайд 19**

Вывод в файл Excel происходит со стандартным форматированием, что дает возможность для моментальной печати документов без внесения каких-либо поправок.

# **Слайд 20**

JSON – простой, основанный на использовании текста, способ хранить и передавать структурированные данные.

После создания строки JSON, ее легко отправить другому приложению или в другое место сети, так как она представляет собой простой текст.

JSON имеет следующие преимущества:

1. Он компактен;
2. Его предложения легко читаются и составляются как человеком, так и компьютером;
3. Его легко преобразовать в структуру данных для большинства языков программирования;
4. Многие языки программирования имеют функции и библиотеки для чтения и создания структур JSON.

# **Слайд 21**

Моё выступление окончено. Спасибо за внимание.