**КОРПОРАТИВНАЯ СИСТЕМА ЛОЯЛЬНОСТИ**

**«Syn»**

**РАБОЧАЯ ДОКУМЕНТАЦИЯ**

**Руководство программиста**

Листов – 8

Киров

2022

ОГЛАВЛЕНИЕ

[1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММЫ 4](#_Toc119946798)

[2. ХАРАКТЕРИСТИКИ И РАБОТА ПРОГРАММЫ 5](#_Toc119946799)

[ЛИСТИНГ ПРОГРАММЫ 8](#_Toc119946800)

**АННОТАЦИЯ**

В данном программном документе приведено руководство программиста по работе с программным обеспечением “Syn”

В разделе «Назначение и условия применения программы» указаны назначение и функции, выполняемые программой, условия необходимые для его работы.

В разделе «Характеристика программы» приведено описание основных

характеристик и особенностей.

1. НАЗНАЧЕНИЕ И УСЛОВИЯ ПРИМЕНЕНИЯ ПРОГРАММЫ

Целью продукта является предоставление работодателю платформы для распространения товаров между ее работниками за счет повышения сверхурочного труда работника по его желанию в получении выгоды в виде товара компании.

Поскольку систему необходимо ввести в компанию между работниками, решено было сделать программное обеспечение для ПК, где каждый может в любое удобное время использовать приложение.

1. ХАРАКТЕРИСТИКИ И РАБОТА ПРОГРАММЫ

Требования к программному обеспечению, которые были выполнены в ходе разработки:

1. Авторизация для доступа к приложению, чтобы посторонний не мог получить к нему доступ.

2. Витрина товаров, а также страница с товаром.

3. Личный кабинет.

4. Покупка товаров за имеющиеся баллы.

5. Административные функции: начисления и снятия баллов; добавление и изменения доступа у пользователей; добавление и снятие товара с витрины.

На рисунке 1 представлена файловая архитектура приложения.

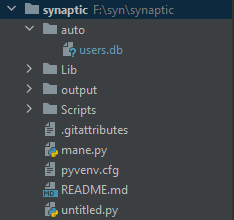


Рисунок 1 – Файловая архитектура приложения

При создании приложения использовался язык программирования python и его библиотека для формирования оконного приложения PyQt5.

Программа состоит из 10 страниц, каждая из которых открывается по своей функции основой которой является метод open и closed от PyQt. К каждой кнопке привязана отдельная функция и они не повторяются.

Для создания приложения необходимо было учитывать два сценария: вошел пользователь или вошел администратор, если пользователь, то в приложении ему будут доступны только главная страница с товарами и личный кабинет, для администратора же будет страница admin для взаимодействия с пользователями, товарами и баллами.

На рисунке 2 представлен вид приложения для пользователя, а на рисунке 3 представлено уже для администратора

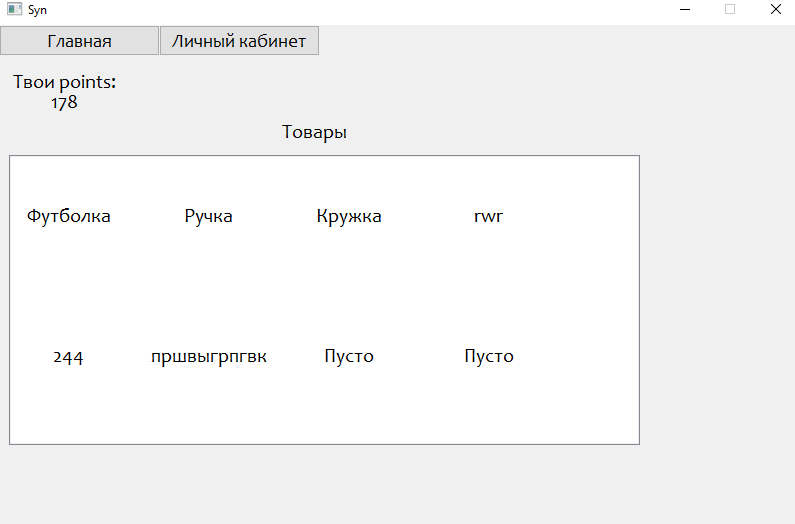


Рисунок 2 – Вид приложения у пользователя без прав администратора

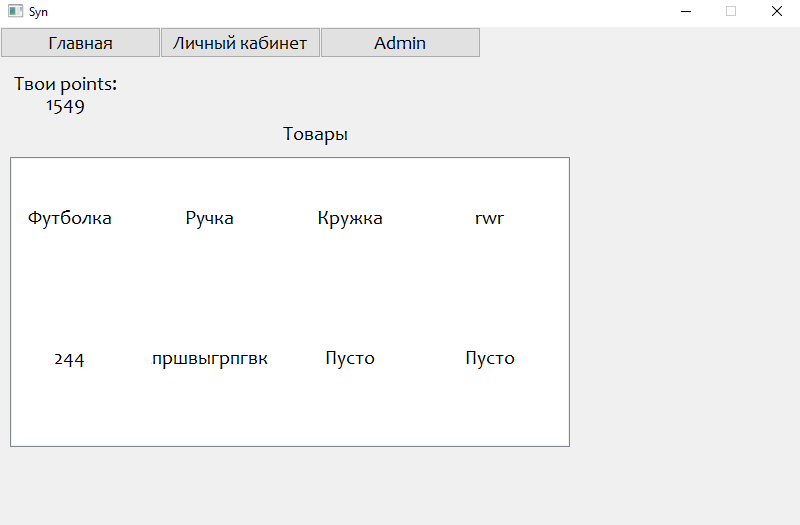


Рисунок 3 – Вид приложения у пользователя с правами администратора

На рисунке 4 представлен код авторизации аккаунта.

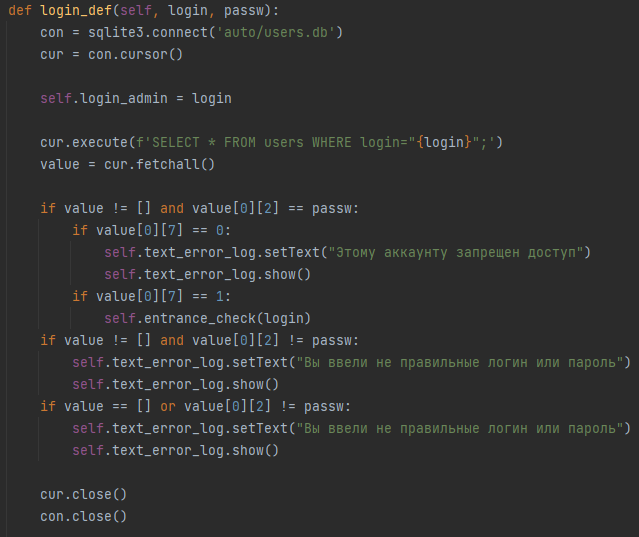


Рисунок 4 – Код авторизации аккаунта

# ЛИСТИНГ ПРОГРАММЫ

«Авторизация пользователя»

def login\_def(self, login, passw):  
 con = sqlite3.connect('auto/users.db')  
 cur = con.cursor()  
  
 self.login\_admin = login  
  
 cur.execute(f'SELECT \* FROM users WHERE login="{login}";')  
 value = cur.fetchall()  
  
 if value != [] and value[0][2] == passw:  
 if value[0][7] == 0:  
 self.text\_error\_log.setText("Этому аккаунту запрещен доступ")  
 self.text\_error\_log.show()  
 if value[0][7] == 1:  
 self.entrance\_check(login)  
 if value != [] and value[0][2] != passw:  
 self.text\_error\_log.setText("Вы ввели не правильные логин или пароль")  
 self.text\_error\_log.show()  
 if value == [] or value[0][2] != passw:  
 self.text\_error\_log.setText("Вы ввели не правильные логин или пароль")  
 self.text\_error\_log.show()  
  
 cur.close()  
 con.close()