

ПРИЛОЖЕНИЕ С ГРАФИЧЕСКИМ ИНТЕРФЕЙСОМ PyQt5

«Smart Food Calendar (SF Calendar)»

Smart Food Calendar – приложение, помогающее следить за своим рационом питания. Производит автоматизированный поиск по базе данных питательных веществ различных продуктов, хранит данные дневника питания, приводит статистику рациона питания, дает необходимые советы по его корректировке.

Идея создания такого приложения появилась у меня из-за потребности следить за своим рационом питания. Как известно, секрет здоровья нашего организма заключается в продуктах, которые мы употребляем в пищу. Но как уследить за тем, что мы съели? Записывать в блокнот? – Неудобно, да и к тому же, как обработать на бумаге большое количество информации?! Тут и приходят на помощь использование информационных технологий.

Цель:

Написание приложения, способного выполнять следующие функции:

- поиск энергетической ценности продуктов,
- хранение дневника питания в удобном для пользователя формате,
- редактирование данных,
- анализ дневника питания, вывод советов по корректировке рациона питания.

Задачи, которые в течение проекта мне пришлось решить:

- Выбор языка программирования и библиотеки для создания графического интерфейса,
- Проектирование структуры приложения,
- Выведение формул расчетов норм содержания белков, жиров и углеводов,
- Поиск подходящих средств-виджетов для вывода или получение различной информации,
- Выбор типа базы данных, содержащей информацию об энергетической ценности конкретных продуктов, и ее проектирование.

Результат работы:

В ходе работы над проектом мною была выбрана GUI библиотека PyQt5 языка программирования python3.

Программа была реализована через классы окон, отвечающих за различные функции. Для каждого окна был спроектирован свой интерфейс с сочетающейся между собой цветовой палитрой.

Спроектированные классы:

- MainWindow
- PersonalDialog
- Cabinet
- Statistic
- HelloScreen
- Info

Интересные идеи:

- Загрузочная анимация
- Непрерывный поиск в формате реального времени
- Приветственные, персонализированные надписи
- Работа с датами
- Построение круговой диаграммы с анимацией
- Добавление нескольких элементов в таблицу одновременно

Изученные технологии:

- Несколько форм, их взаимодействие между собой
- Изученные виджеты: QPushButton, QComboBox, QCalendarWidget, QTableWidget, QLayout, QLabel
- Другие виджеты: QChart
- Стандартный диалог получения информации, вручную написанный класс диалогового окна
- Использование Resource Root Qt designer
- Чтение и запись в txt файл
- Связь разделов по id между двумя таблицами в БД

- Работа с БД: Чтение, добавление, обновление, удаление данных
- Ехе-пакет

В результате работы было создано приложение, выполняющее все функции, описанные в цели данного проекта. Приложение работоспособно и готово к использованию.

Рисунок 1 Рабочий стол приложения

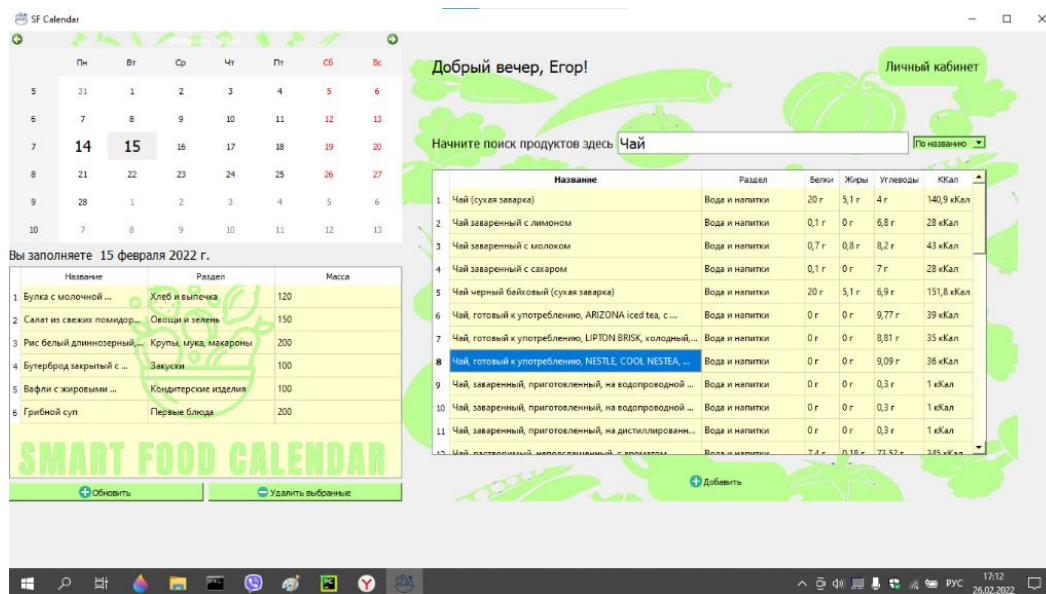


Рисунок 2 Вывод статистики

