# Зщита на дипломен проект – част теория на професията.

1. Представяне на ученика и темата.

Добър ден, моето име е Иван Халачев и с удоволствие ще ви представя моят проект - конфигурационен модул, който беше разработен с използването на мощния инструмент PyQt5. Проектът ми е част от по-голяма система за автоматично архивиране на информация от работни станции към отдалечен сървър за съхранение на данни.

1. Проблемът

Проблемът, който решавам с този проект, е сложността на конфигурирането на клиентски приложения, работещи в автономен режим. Предвидено е клиентският модул на архивиращата система да взема своите конфигурационни данни от ini файл. По същество това е обикновен текстов файл, който би могъл да се редактира „на ръка“. Но ръчната конфигурация изисква компетентност, време и усилия, а също така възникват и рискове от човешки грешки. Неконтролираните грешки и неефективността на този процес са предизвикателства, които трябва да се преодолеят.

1. Интерфейсът

Представям ви графичния интерфейс на нашия конфигурационен модул. Това е интуитивен и лесен за използване изцяло графичен клиентски интерфейс. Потребителят работи с него по привичния в наши дни начин – с мишка и клавиатура, данните се визуализират в отделни стандартни графични контроли. Модулът предоставя възможност както за настройка на връзката с FTP сървъра, така и за задаване на други необходими локални настройки.

1. Графични интерфейси в Python

Тъй като екипът е избрал като платформа за разработка езика Python, то лoгичната ми първа стъпка в разработката беше да разгледам възможностите му за играждане на графични приложения. А те не са малки.

На първо място, това е вградената библиотека tkinter. Тя е лесна и удобна за работа, но предлага доста ограничен набор визуални компоненти.

Друга често използвана графична библиотека е Kivy. Могат да се споменат още wxPython, PyGTK, Pygame, PySimpleGUI и др.

1. PyQT

Но, безспорно, най-добър за създаване на графични приложения е инструментът QT и неговата версия за Python PyQT. Това е мощен мултиплатформен графичен framework. Възможностите му са буквално неограничени. Освен поддръжката класове и инструменти за изграждане на графични потребителски интерфеси (GUI) PyQT предлага и поддръжка на XML, мрежова комуникация, регуларни изрази, нишки, SQL бази данни, уеб-броузинг и много други възможности.

Недостатък е необходимостта от лиценз за комерсиално ползване. Но винаги може да се използва и неговия „близнак“ с отворен код PySide.

1. QtDesigner

Допълнително удобство при използването на PyQT e пълната му интеграция с QtDesigner. Пълноценна графична среда с отворен код за визуално проектиране на графичен интерфейс.

1. Функционалност - конфигурация

Доколкото става дума за конфигурационен модул, то неговата първа задача е да борави с конфигурационните данни – настройките, на архивиращия модул. Тези данни се съхраняват в локалната папка като ini файл. Използвам библиотеката configparser за да чета и записвам стойностите на използваните в приложението параметри – например, работна папка, тип на наблюдаваните файлове и др.

1. Функционалност - графика

А доколкото става дума за графичен интерфейс, то другата задача беше да се обвържа четенето, редактирането и записа на конфигурационните данни с графичния интерфейс. Към тази част могат да се отнесат и някои допълнителни функци, които съм добавил. Например, прочитане и визуализация на съхранените на сървъра файлове за даден клиент.

1. Бъдещето

Вярвам че проектът ни има бъдеще. А за бъдещото му развитие ние вече имаме някои идеи. Например, да добавим поддръжка за FTPS и SFTP протоколи. Или пък - да разширим възможностите за автоматично синхронизиране на данни между работните станции и сървъра.

Ние в една или друга степен сме зависими от данните с които работим. А мрежата освен предимствата, които дава, крие и много заплахи. Нуждата от надеждно и лесно архивиране ще нараства. И тук идва на помощ проекта на нашия екип, в това число и моята разработка.

Благодаря за вниманието!

Ако имате въпроси – готова съм да отговоря.