ERPY

Comprehensive ERP in Python

Requirements:

This project is designed to be able to run in almost any kind of system, as long as it is able to run a Python interpreter, without any kind of necessary changes in it. In other words, it should work out of the box once correctly installed in the system.

Additionally, it could be adjusted to connect to a server through a network to allow multiple systems to use it at the same time.

Minimal requirements:
- x86 or ARM CPU Architecture
- 1GB RAM
- Internet Connection

Prototype:

The prototype provided includes only a local database an basic instructions and GUI design to allow the employees and owners to give necessary feedback and proposals, and get their hands into the basic structure and appearance.

Current working systems are addition of Customers, Sales and Items instances, as well as watching current data in said types within the database. Final GUI and colorscheme is also implemented, although slight variations can be made.

Future development milestones are: More data visualization, search filters, hand-picked data updates or deletes, a connection to a network-based database if specified necessary, and a log-in system for the employees to hide the data under a security layer.

In the last stage, we would implement an API that the online store could use to indicate our program any kind of change in our data and update it accordingly in the database

How to use:

The program is divided in three sections, each with different targets, but it could be extended along with the database to allow more types and extensions of data management.

- Customers: We can add customers to the system providing the necessary information. We can also oversee the current customers in the database. Up to here it is shown in the demo provided. Additionally, we will be able to search by different fields, delete, edit, and display statistics with this data, as well as join them to their related information in other sections of the program
- Items: It imitates the functionality of the previous one in both the demo and the on-develop features, but applied to the items in the database, which will be kept stock updated real time, also being able to update it manually if necessary.
- Sales: Here we could select an existent customer from our database, and add the existent items sold to him. We can create a receipt adding (or deleting) items, that will later be generated in a PDF if we configure to do so, and added to the database, updating real time the stock left in our database or informing us if we're short of stock for the sale.

All errors will be displayed as red warnings on the screen with instructions for the user to solve the problem. All actions will be display a success message in green text in the screen as well.