REPORT FOR ORDER MANAGEMENT SYSTEM

PYTHON PROGRAMMING (INT 213)

Github: https://github.com/Ganda-Bhalu

Name:	Adhiraj Singh
Registration No:	12013561
Program:	B. Tech CSE
Semester:	Third
School:	School of Computer Science and Engineering
Name of the University:	LPU
Date of Submission:	20/11/2021

CONTENTS

ABSTRACT

The modern business world is a complicated one. Processes that used to be straightforward, today include a range of extra challenges. Even something as seemingly simple as making a sale and getting the product to a customer is far more complicated than ever before. Retailers and businesses must consider a multitude of factors when selling their wares.

The channels through which consumers are buying like Shipping options & fulfillment challenges, Third-party logistics, Analytics & business intelligence

It's no surprise that many brands need a little assistance. Fortunately, the help that's needed is close at hand in the shape of order management systems (OMS). An OMS is a vital part of operations for ecommerce and other businesses. If you're not up to speed on what an order management system is, it's time to learn.

You should see by now – if you didn't know already – that modern order management is no straightforward task. The days of buying the correct products, marketing them, and letting everything else take care of itself are long gone.

An OMS is the method by which firms handle this vital business process. It can be any tool, platform, or structure to track and control all the elements of the process detailed earlier.

Sales, inventory and fulfillment are often handled by your OMS. That's why an order management system is so critical.

No matter the size or nature of your business, an OMS must fulfill certain functions. How complicated it is to do so depends on your firm and the level at which you operate. That's why some companies need a more comprehensive system than others. But we'll talk more about that later.

ACKNOWLEDGEMENT

I would like to thank myself for doing such a deep research on the project and implementing the understanding with the utmost hard work and dedication.

TEAM MEMBERS

Team lead: Adhiraj Singh
Team member: Adhiraj Singh

LIBRARIES

1. **Tkinter:** Tkinter is a Python binding to the Tk GUI toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's de facto standard GUI. Tkinter is included with standard GNU/Linux, Microsoft Windows and macOS installs of Python. The name Tkinter comes from the Tk interface.
2. **Tkinter.MessageBox:** The tkinter.messagebox module provides a template base class as well as a variety of convenience methods for commonly used configurations. The message boxes are modal and will return a subset of (True, False, OK, None, Yes, No) based on the user's selection. Common message box styles and layouts include but are not limited
3. **Tkinter.Font:** The Font class represents a named font. Font instances are given unique names and can be specified by their family, size, and style configuration. Named fonts are Tk's method of creating and identifying fonts as a single object, rather than specifying a font by its attributes with each occurrence.
4. **SQLite3:** SQLite is a C library that provides a lightweight disk-based database that doesn't require a separate server process and allows accessing the database using a nonstandard variant of the SQL query language. Some applications can use SQLite for internal data storage. It's also possible to prototype an application using SQLite and then port the code to a larger database such as PostgreSQL or Oracle.

ABOUT

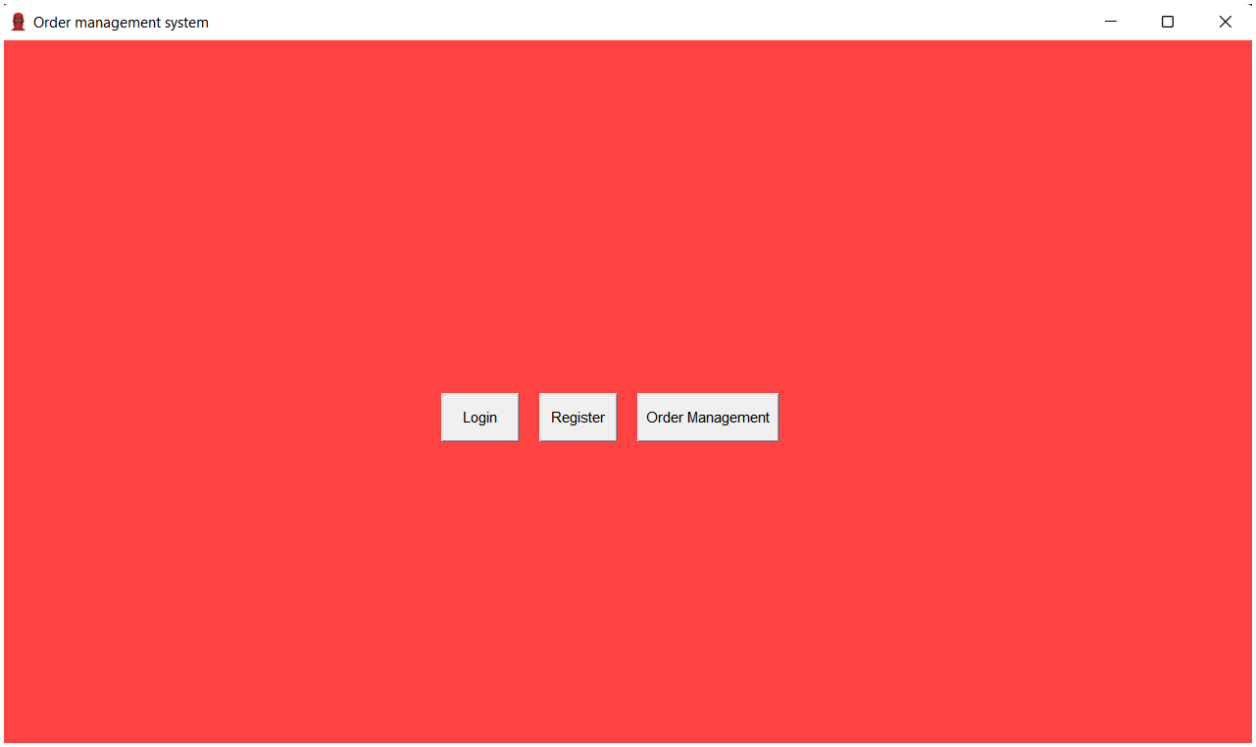
Many order management systems offer real-time trading solutions, which allows the user to monitor market prices and execute orders in multiple exchanges across all markets instantaneously by real-time price streaming. Some of the benefits that firms can achieve from an order management system include managing orders and asset allocation of portfolios.

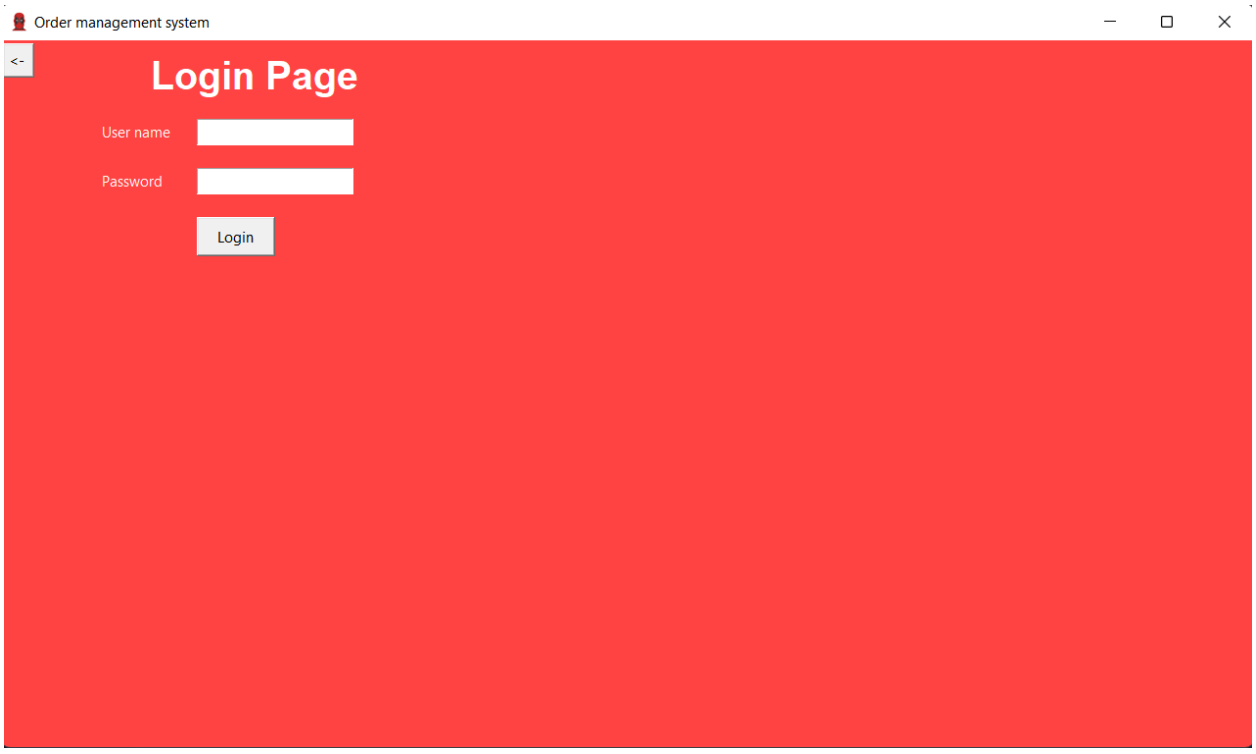
An effective OMS is critical in helping with regulatory compliance including real-time checks of trades both before and after entry. Order management systems help compliance officers with tracking the lifecycle of trades to determine if there's any illicit activity or financial fraud as well as any regulatory breaches by an employee of the firm. An OMS can improve workflow and communication between portfolio managers, traders, and compliance officers.

Order management systems are an important development in the financial services industry because of the real-time monitoring of positions, the ability to prevent regulatory violations, the speed and accuracy of trade execution, and the significant cost savings that result.

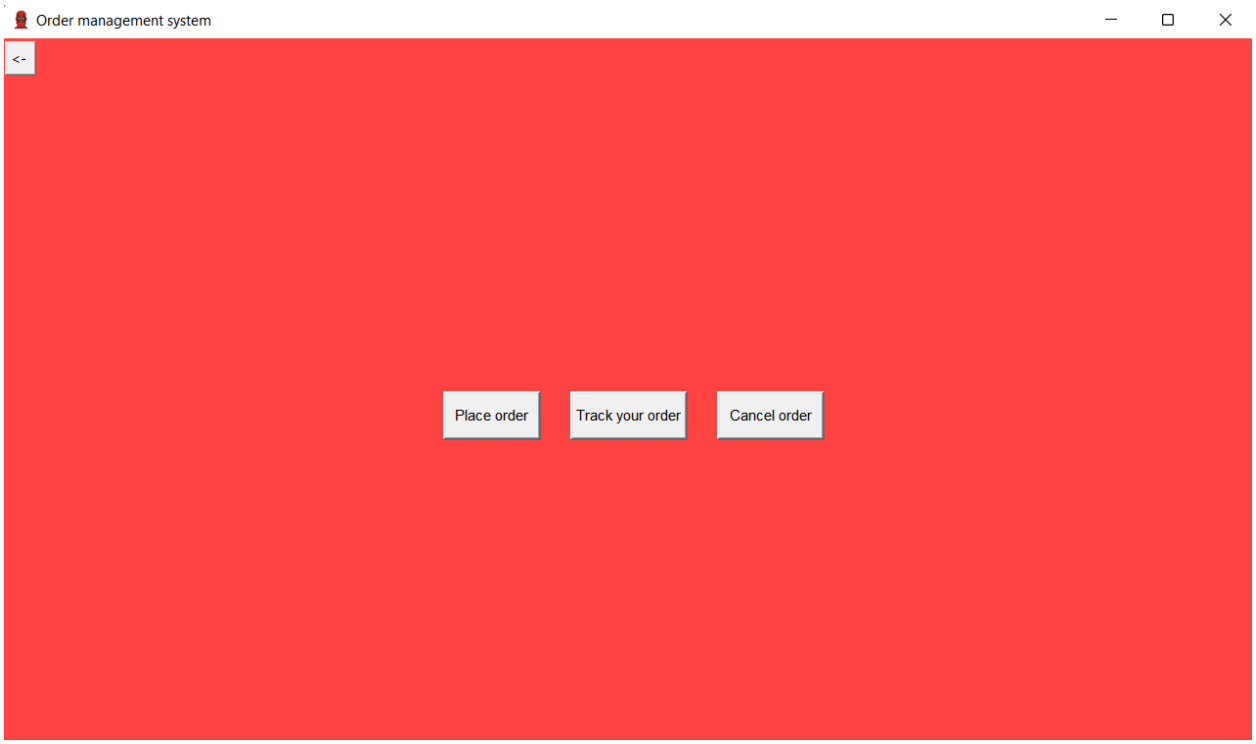
The project takes care of all categories of orders, and effectively manages the data using the database management system.

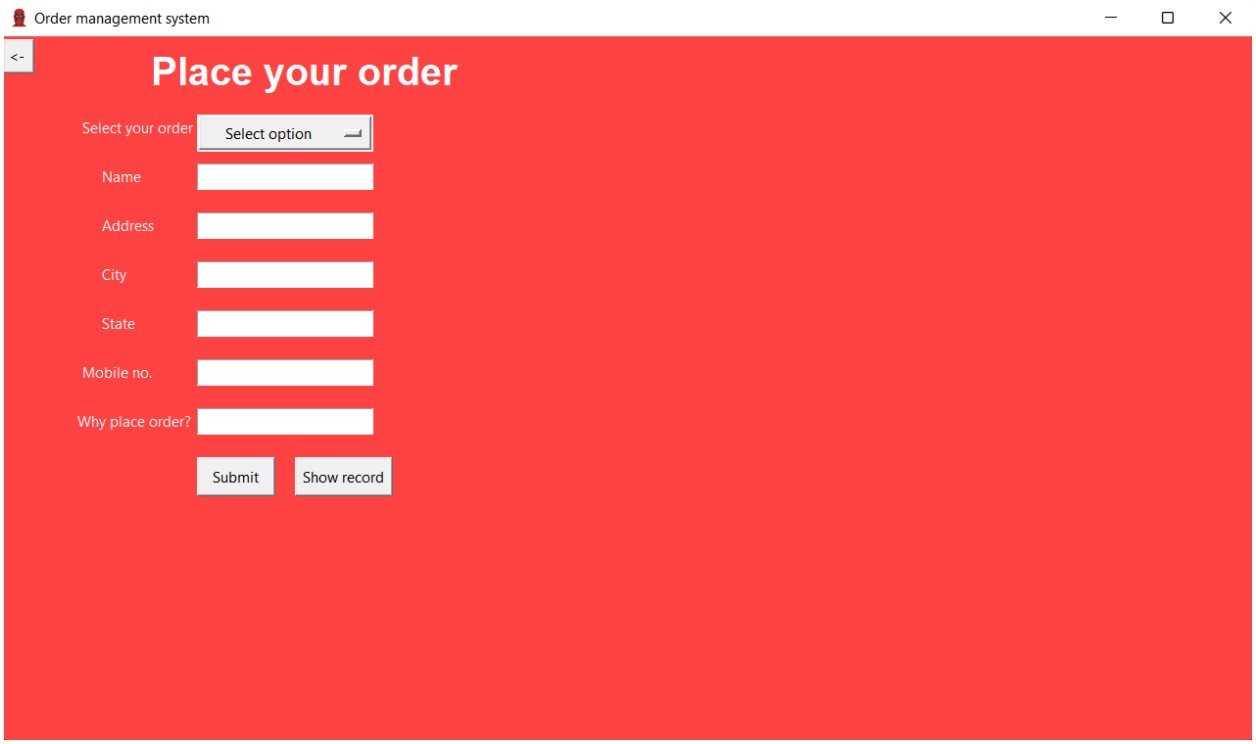
SCREENSHOTS

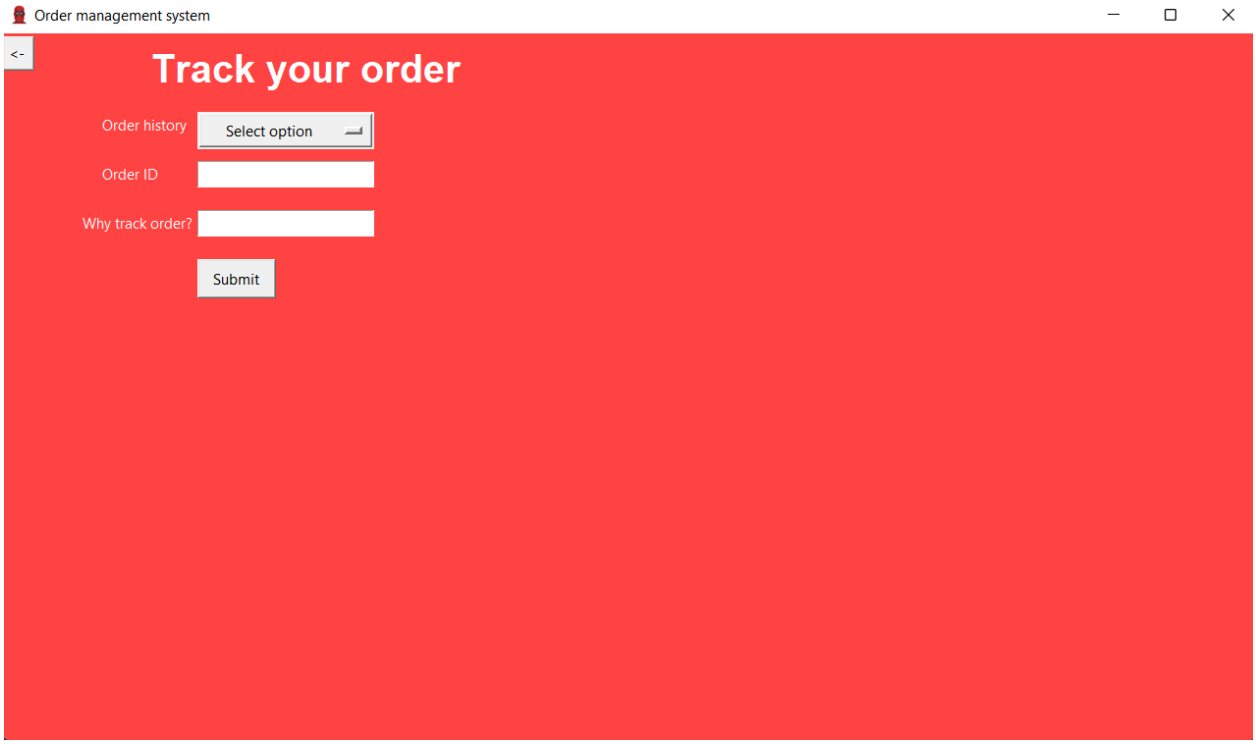


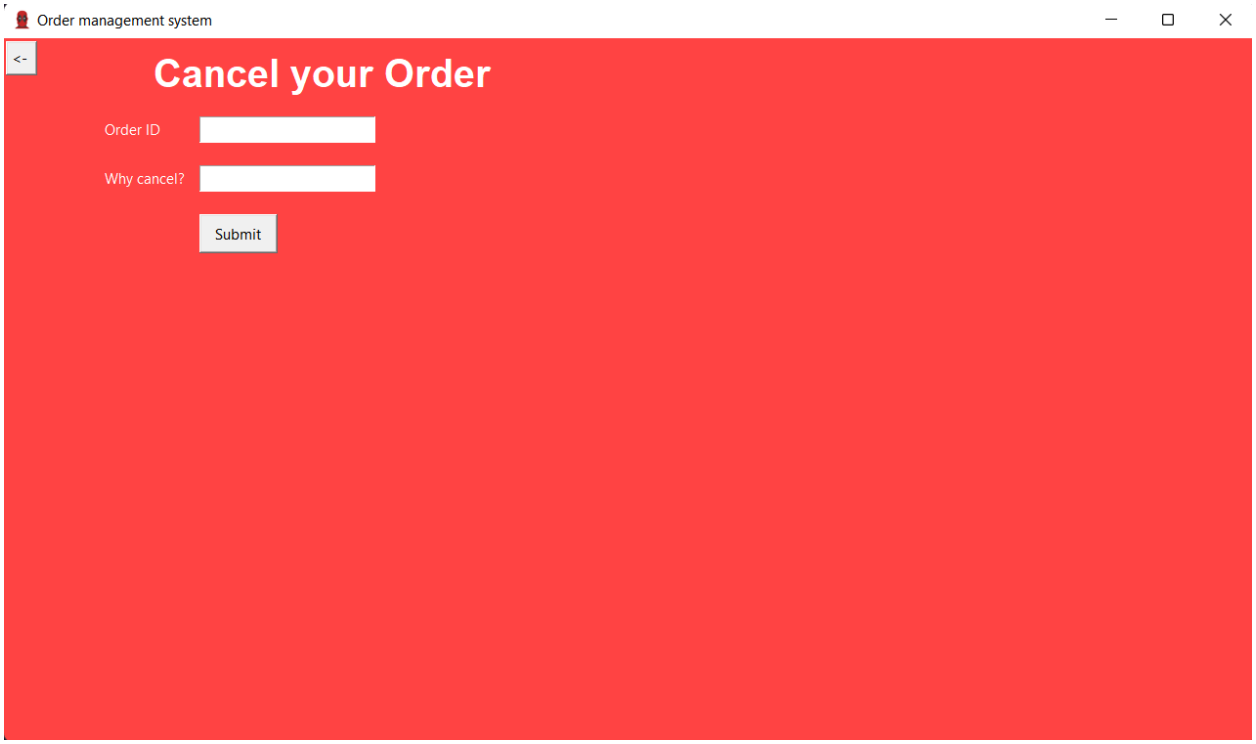


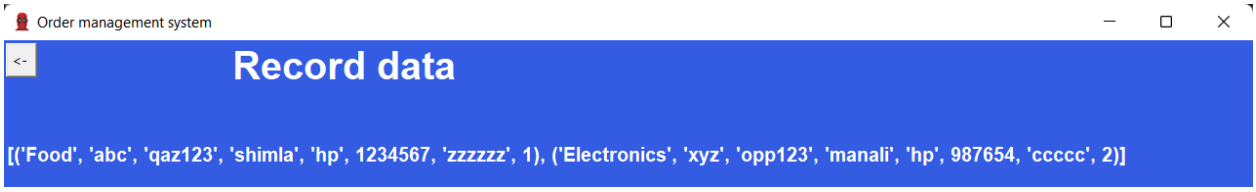

Order management system
<-
Registration page
Name
Address
Gender
Male
Female
Mobile no.
Email
Register


Order management system
<-
Place order
Track your order
Cancel order

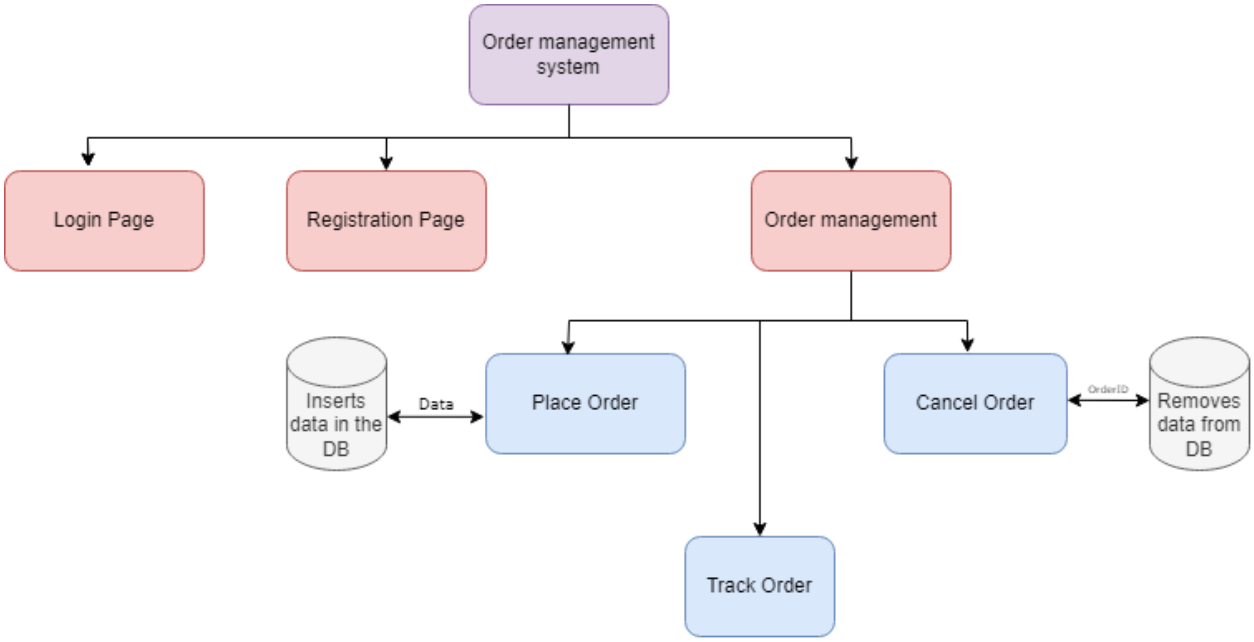

Order management system
<-
Place your order
Select your order
Select option
Name
Address
City
State
Mobile no.
Why place order?
Submit
Show record


Order management system
<-
Track your order
Order history
Select option
Order ID
Why track order?
Submit


Order management system
<-
Cancel your Order
Order ID
Why cancel?
Submit


Order management system
<-
Record data
[('Food', 'abc', 'qaz123', 'shimla', 'hp', 1234567, 'zzzzzz', 1), ('Electronics', 'xyz', 'opp123', 'manali', 'hp', 987654, 'ccccc', 2)]

FLOW CHART


Order management system
Login Page
Registration Page
Order management
Inserts data in the DB
Data
Place Order
Cancel Order
OrderID
Removes data from DB
Track Order

REFERENCES

1. StackOverFlow
2. Python.org