

# Fruit Store

Due to the COVID-19 epidemic, Mr. An's Century fruit store has had a slight change in the way of doing business during this social distancing period. The number of customers coming to buy directly at the store decreased, but the number of orders by phone, Facebook, Zalo ... increased significantly. Everything looks good at first, but in the long run, when the number of orders is too large, it leads to overload for the order manager. Because he wants to challenge his son (Binh - currently working as a programmer), Mr. An does not use external applications, but wants him to build a system that meets at least **the following requirements**:

- Allows customers to order, view purchase history by order, purchase time
- Allows shop owners to add, delete and update items

Unfortunately, although Binh has completed the user interface of the system, due to the busy work of the company, he cannot complete the back-end part. So he wants to ask you to build this backend in the form of Restful API.

## Note:

- This assignment expects about **24 hours** to complete
- Unit for each type of fruit is kilogram
- The examples we give are for reference only, help you to be clear about the problem, feel free to be creative. Let's take a look at some examples:

Place order
POST
<pre>{   "fruits": {     "mango": 1,     "apple": 1,     "pineapple": 1   },   "user": "mason" }</pre>

Get order history
GET
<pre>{   "1535414400": {     "mango": 5,     "orange": 6   } }</pre>

**Requirements:**

You are required to demonstrate your skills and your deep understanding of the selected technologies. You also need to prove your ability to deliver production-grade outputs.

- Use NodeJS or Python as backend language, feel free to use framework.
- You MUST use a database. A dedicated database server (PostgreSQL, Mongo, Redis, ... ) or an embedded database (SQLite, RocksDB, ...) is welcome.
- As we use docker in everyday work, your product MUST be delivered with materials that facilitate the dockerization.
- Documentation and tests are required.

We will follow the instructions in your documentation, therefore, make sure that your documentation is well tested.

Happy coding!