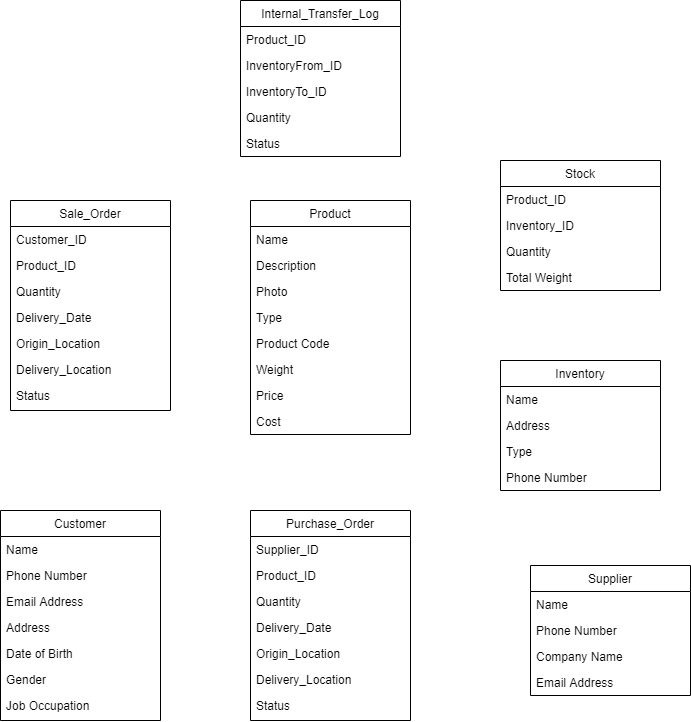
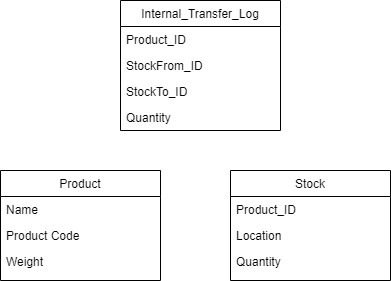
## Short Daily (Day1)

First of all, I read the requirement documents.

Then, started to think of different database design to fit the specification. I have thought of the below design.

However the above design is too complicated for a 3 day project. Therefore, I had decided to simplify the design with 3 tables, which just enough to fulfill the base requirements. See the below class diagram.



%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22Stock%22%20style%3D%22swimlane%3BfontStyle%3D0%3Balign%3Dcenter%3BverticalAlign%3Dtop%3BchildLayout%3DstackLayout%3Bhorizontal%3D1%3BstartSize%3D26%3BhorizontalStack%3D0%3BresizeParent%3D1%3BresizeLast%3D0%3Bcollapsible%3D1%3BmarginBottom%3D0%3Brounded%3D0%3Bshadow%3D0%3BstrokeWidth%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22530%22%20y%3D%22180%22%20width%3D%22160%22%20height%3D%22130%22%20as%3D%22geometry%22%3E%3CmxRectangle%20x%3D%22550%22%20y%3D%22140%22%20width%3D%22160%22%20height%3D%2226%22%20as%3D%22alternateBounds%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%223%22%20value%3D%22Product\_ID%22%20style%3D%22text%3Balign%3Dleft%3BverticalAlign%3Dtop%3BspacingLeft%3D4%3BspacingRight%3D4%3Boverflow%3Dhidden%3Brotatable%3D0%3Bpoints%3D%5B%5B0%2C0.5%5D%2C%5B1%2C0.5%5D%5D%3BportConstraint%3Deastwest%3B%22%20vertex%3D%221%22%20parent%3D%222%22%3E%3CmxGeometry%20y%3D%2226%22%20width%3D%22160%22%20height%3D%2226%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%224%22%20value%3D%22Inventory\_ID%22%20style%3D%22text%3Balign%3Dleft%3BverticalAlign%3Dtop%3BspacingLeft%3D4%3BspacingRight%3D4%3Boverflow%3Dhidden%3Brotatable%3D0%3Bpoints%3D%5B%5B0%2C0.5%5D%2C%5B1%2C0.5%5D%5D%3BportConstraint%3Deastwest%3Brounded%3D0%3Bshadow%3D0%3Bhtml%3D0%3B%22%20vertex%3D%221%22%20parent%3D%222%22%3E%3CmxGeometry%20y%3D%2252%22%20width%3D%22160%22%20height%3D%2226%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%225%22%20value%3D%22Quantity%22%20style%3D%22text%3Balign%3Dleft%3BverticalAlign%3Dtop%3BspacingLeft%3D4%3BspacingRight%3D4%3Boverflow%3Dhidden%3Brotatable%3D0%3Bpoints%3D%5B%5B0%2C0.5%5D%2C%5B1%2C0.5%5D%5D%3BportConstraint%3Deastwest%3Brounded%3D0%3Bshadow%3D0%3Bhtml%3D0%3B%22%20vertex%3D%221%22%20parent%3D%222%22%3E%3CmxGeometry%20y%3D%2278%22%20width%3D%22160%22%20height%3D%2226%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%226%22%20value%3D%22Total%20Weight%22%20style%3D%22text%3Balign%3Dleft%3BverticalAlign%3Dtop%3BspacingLeft%3D4%3BspacingRight%3D4%3Boverflow%3Dhidden%3Brotatable%3D0%3Bpoints%3D%5B%5B0%2C0.5%5D%2C%5B1%2C0.5%5D%5D%3BportConstraint%3Deastwest%3Brounded%3D0%3Bshadow%3D0%3Bhtml%3D0%3B%22%20vertex%3D%221%22%20parent%3D%222%22%3E%3CmxGeometry%20y%3D%22104%22%20width%3D%22160%22%20height%3D%2226%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E

After that , I setup the environment for some time, also reading the update notes of the frameworks. After the project base (source code) is generated, I looked around to observe any updates from the last version that I use before. Lastly, I have tried to generate some angular component and add the routing .

That is all for the first day. Good Night!

## Short Daily (Day2)

At the beginning of this day, I start to design the use flow of the system. My main idea is to develop a inventory management with a product based design, which means the use flow start from listing all the products within the inventory system. Start selecting a product in the list, user can manage its inventory level at different location.

I have also think of a location based design, but the system designed yesterday cannot handle this design efficiently. It would be better to separate location and stock quantity in different tables. I have then realize that system design can greatly alter the system flexibility. Also, system design are greatly depend on requirements and business logic, i.e., no system design is perfect with all situation.

Then I start to implement the system by components, following the use flow, from the data import, product list page, product management page to the operations available on product management page. Within the same component, I developed from the front-end to the back-end.

## Short Daily (Day3)

Basically I do the similar job as the second days and do more testing on the functions. Therefore, I had more debugging works to do in this day.

I also have added data validation on the both client and server side application to ensure the data consistency and integrity.

I wanted to implement the unit test but the time remaining is not quite enough. I had experience of implementing unit test in a Kotlin based Spring. It is basically about implementing the given input to a function and asserting the return of the function.

I have also considered some behaviors of the system. For example, whether a product record is “deletable” or removable in the database, how to handle the repeated data import and whether or not to keep log of stock in and stock out.

For the first question, the answer is no, because normal business is better to keep record adding a product status would be a better option.

I prefer adding a edit button for it to keep it simple.

For the second question, I decided to skip the repeated and invalid product data. Also, I decided to skip the invalid stock data and aggregate the new data. Therefore, stock csv file is actually containing the change in quantity instead of sum of quantity.

For the third questions, the log of stock in and stock out should be saved in sale order and purchase order, so it would be the later enhancement of this project.

And of course, I wrote the README.md file and this short daily on the last day. Hope you would pleased in reviewing of my application, have a good day :D!