# Criterion A: Planning

## Description of the problem

Khoi Nguyen, my client, is a high school student who also sells shoes and basketball equipment. He has his own shop called "Shoes Genix", and because he is still in school, he has difficulty managing his clients' orders on his own. He finds it struggling to manage school and this part-time job; occasionally, he forgets the client's order and receives unfavorable feedback, undermining his credibility. He would like an application to assist him in managing all of the orders and products that he has in stock, so that he may avoid missing orders and focus more on his schoolwork.

We discussed and came up with the most important thing which defines how helpful will the application be convenience. After chatting with him, which a summary can be seen in **Appendix I**, part 1 and part 2 we came up with the solution of an application that will show all of the current orders that haven't been delivered to the clients and also basic functions to help him with the business, such as calculating total sales, and editing products in the storage,…

## Rationale for solution

I will implement an application with a connection to a database, hence allowing users to monitor and manage orders. The application will be used to handle around 30 orders and 100 types of shoes a month. The application will have the function to do tasks such as editing the storage as well as the orders. Search compatibility will also be implemented to allow users to find and check specific orders or products. This program will considerably help my client's problem because it will handle all of the necessary activities for a small retailer, allowing him to devote more time to his studies.

I chose JAVA as the main programming language since it is platform-independent and uses a virtual machine, so it would most likely operate on any computer. I also want to employ JAVA's object-oriented programming (OOP) functionality to create this application. The object-oriented programming functionality allows me to divide the large program into numerous classes, making it easier to build and maintain. Security is another critical component that we have already discussed, as these data are really significant, and OOP gives us encapsulation that can improve data security.

In order to facilitate user interaction with the program, a Java Graphical User Interface (GUI) will also be employed. Additionally, I'll utilize a local SQLite database to hold all of the application's data, including details on users, orders, and products.

## Success criteria

1. The users are able to log in using a personal username and password to access the application
2. The program is able to auto-update the quantity of the corresponding products in the Storage table after an order is created with that products
3. The program can access a local database with multiple tables in order to update and monitor the application
4. The application will have a feature to keep track of the actions that happened in the application, including adding, removing orders or items, creating new accounts, and deleting orders,...
5. The users will be able to use queries to manipulate the database such as adding or removing orders and items, and changing the status of the orders, …
6. The program is capable of doing simple revenue calculations
7. The program can sort through the database to find a specific or group of orders/items information and display those orders/items
8. The program will be able to notify users if there are errors, such as blank text fields, wrong passwords,...
9. The users can be able to delete all the data at the beginning of the month or whenever they want by clicking a button
10. The users can be able to send feedback to the developers by sending emails to the developers
11. The password inserted into the program's database will be hashed to ensure the security of the application

**Word count:** 411