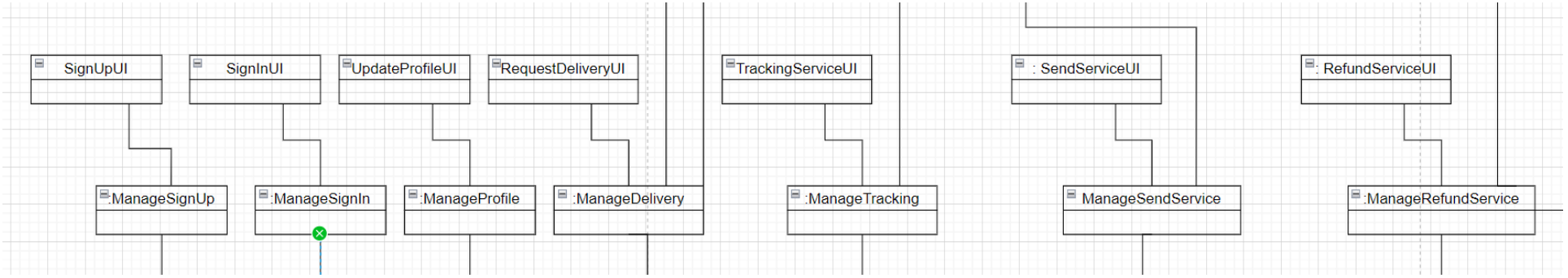
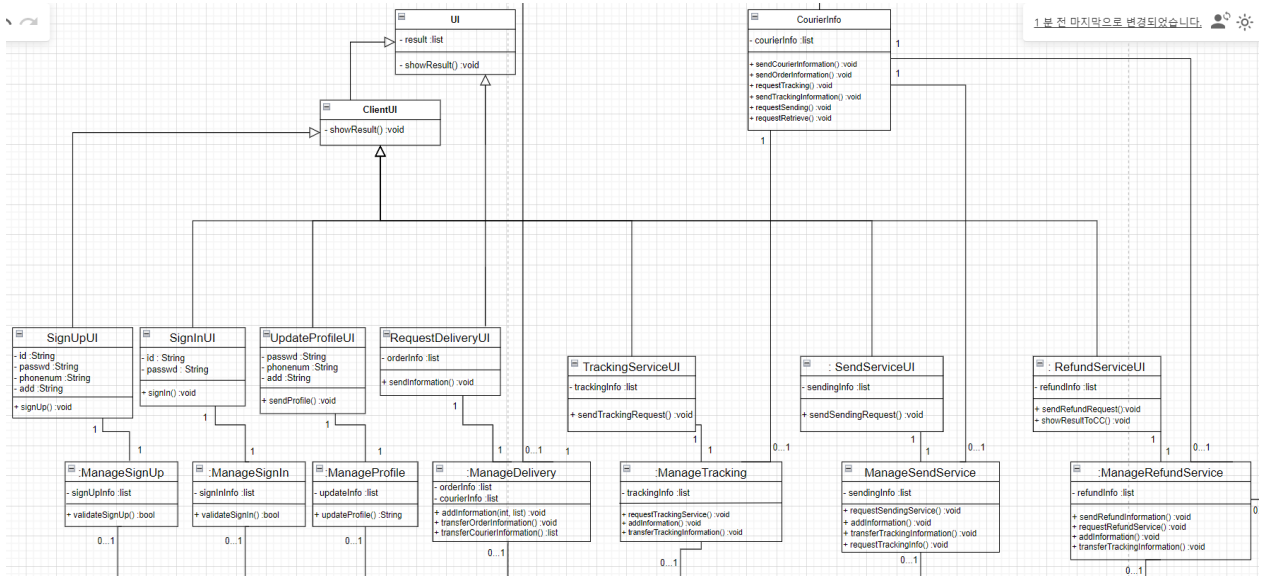
DD1.

- **Analysis**

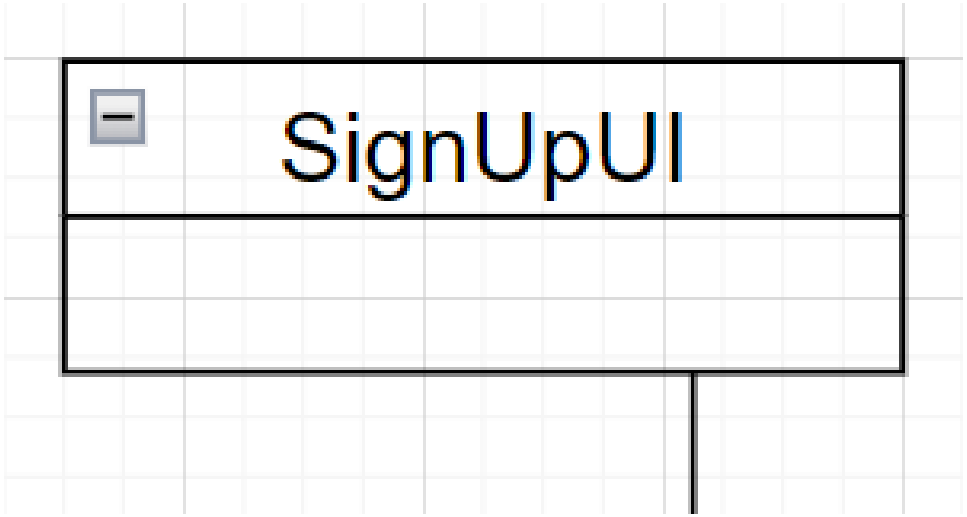
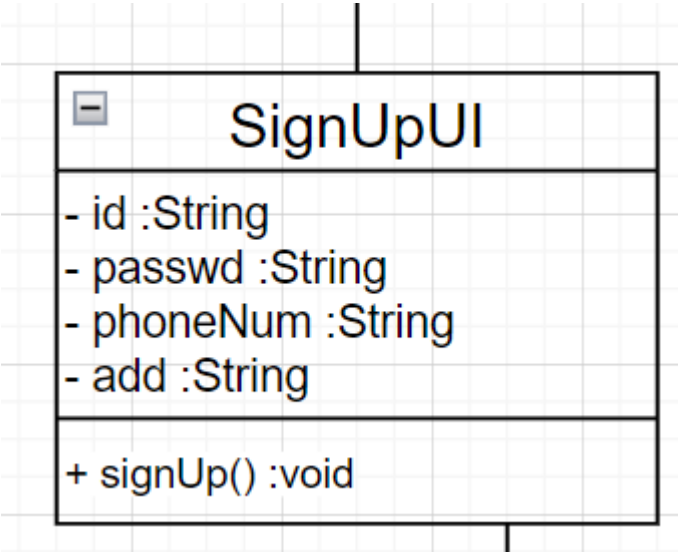


- **Design**



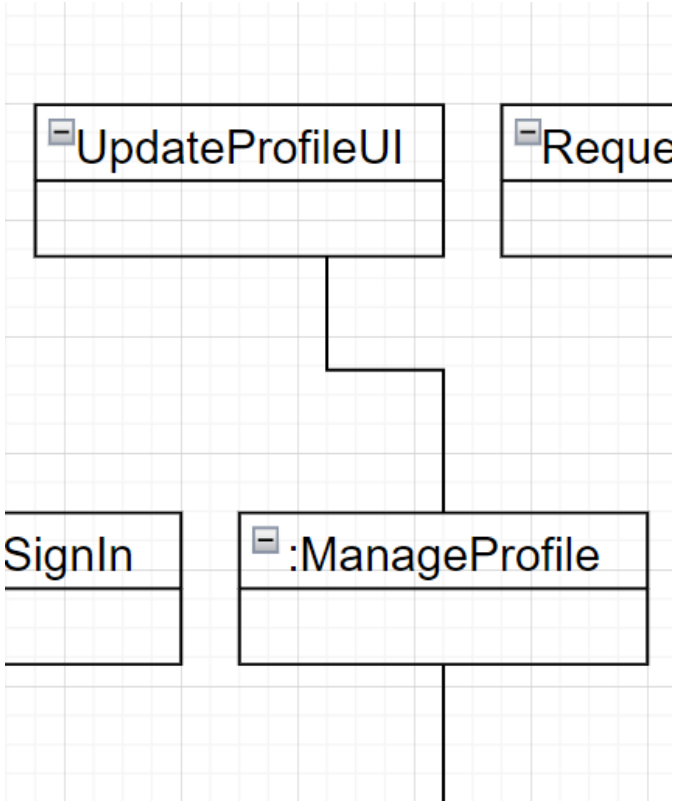
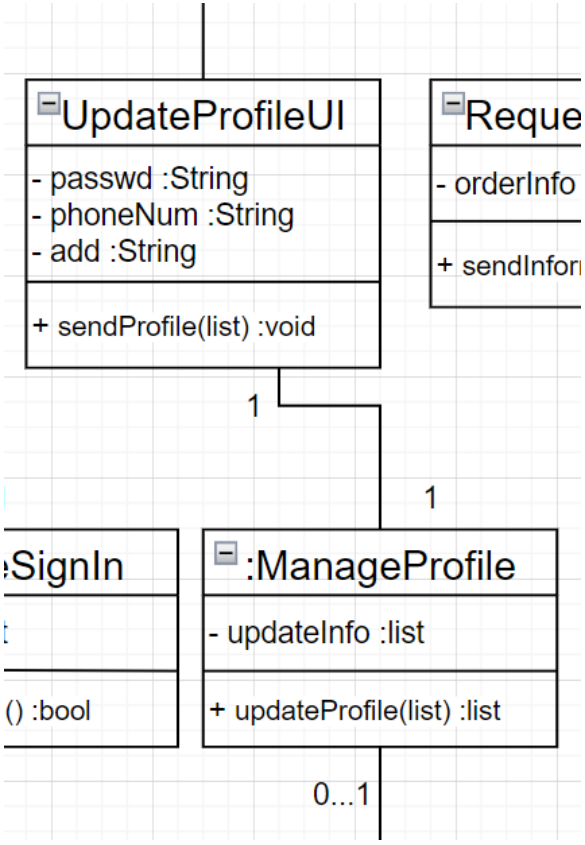
Since all classes corresponding to the UI have the property of result and include the showResult() method, we generalized it for a common part. Subclasses use the showResult() method override. After that, in consideration of scalability, it was classified according to the UI used for each actor and generalized.

DD2.

- Analysis	-Design
	

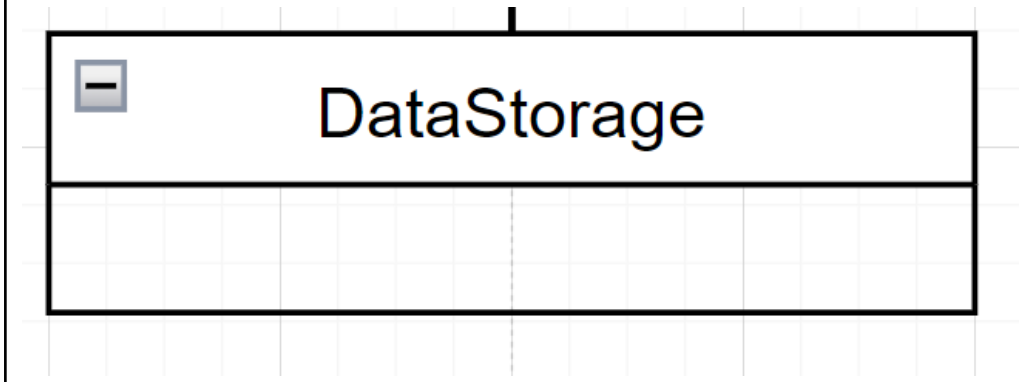
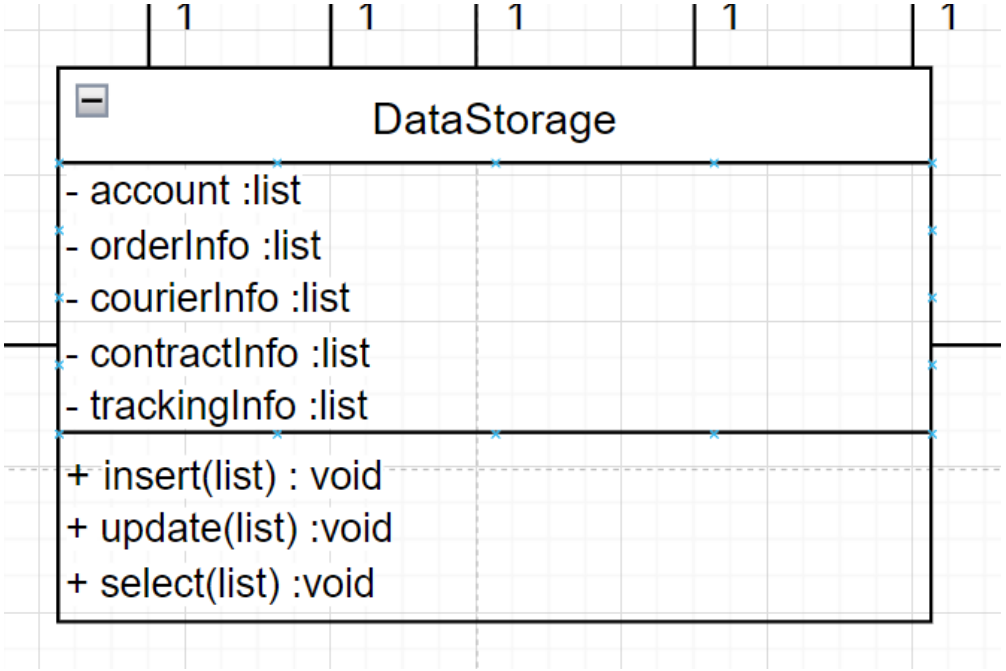
Previously, we thought about receiving it as a parameter, but the reason why we chose it as a property is that it is information related to the class, so we thought it was right to receive related information as a method and have it as an attribute rather than receiving it as a parameter. So most of the parameters of the method became void type.

DD3.

-Analysis	-Design
	

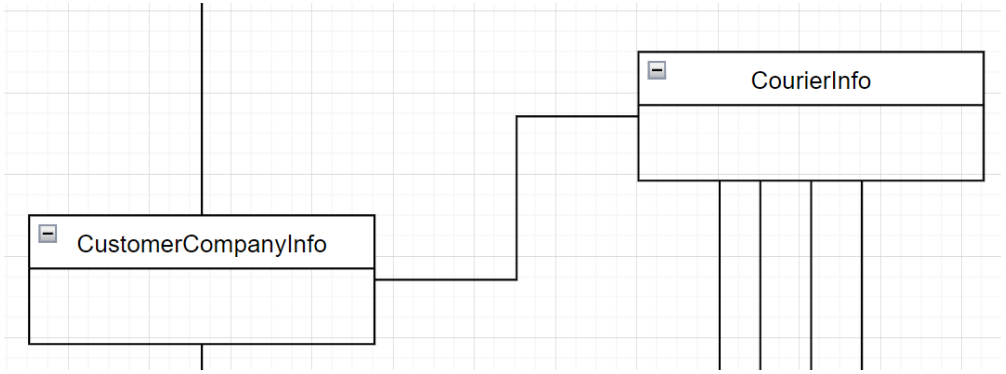
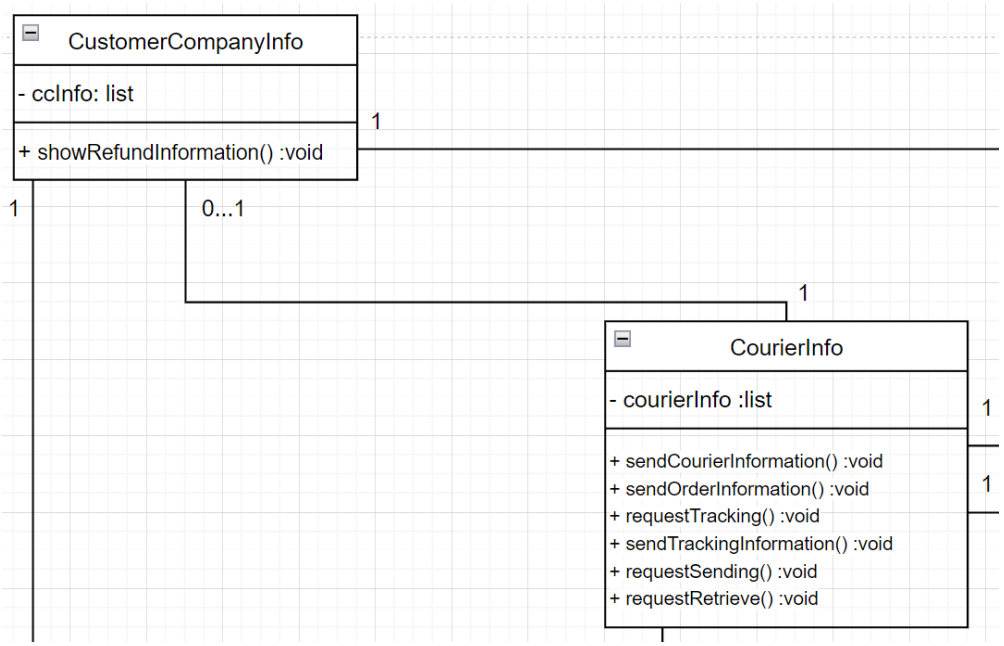
Other functions are designed to have parameters such as void, but profile-related methods have parameters. In order to obtain only information about the user from data storage, the user must be identified by the user's ID, so the user ID was converted into a list type and put in a parameter. The reason why the user id is converted to a list rather than a string is because data storage is designed to store data as a list.

DD4.

-Analysis	-Design
	

The Data Storage class is designed to have the above parameters and methods. Account property has account information, orderInfo property has orderInformation, courierInfo property has courier's information, contractInfo property has contract information, and trackingInfo property has tracking information. In addition, all data storage functions can be used with the above three methods

DD5.

-Analysis	-Design
	

CustomerCompanyInfo and CourierInfo designed as above. The parameters and methods of CustomercompanyInfo are as follows. ccInfo contains information from the customer company and has showRefundInformation() as a method of showing the refund information. The parameters and methods of CourierInfo are as follows. CourierInfo contains the information of the courier and has sendCourierInformation() as a method of sending the information of the courier. It has sendOrderInformation() as a method of sending order information. It is a method for requesting tracking and has request tracking(). It has sendTrackingInformation() as a method that sends tracking information. It has a requestSending() method as a method for requesting a package. It is a method of requesting a retrieve and has a request Retrieve().