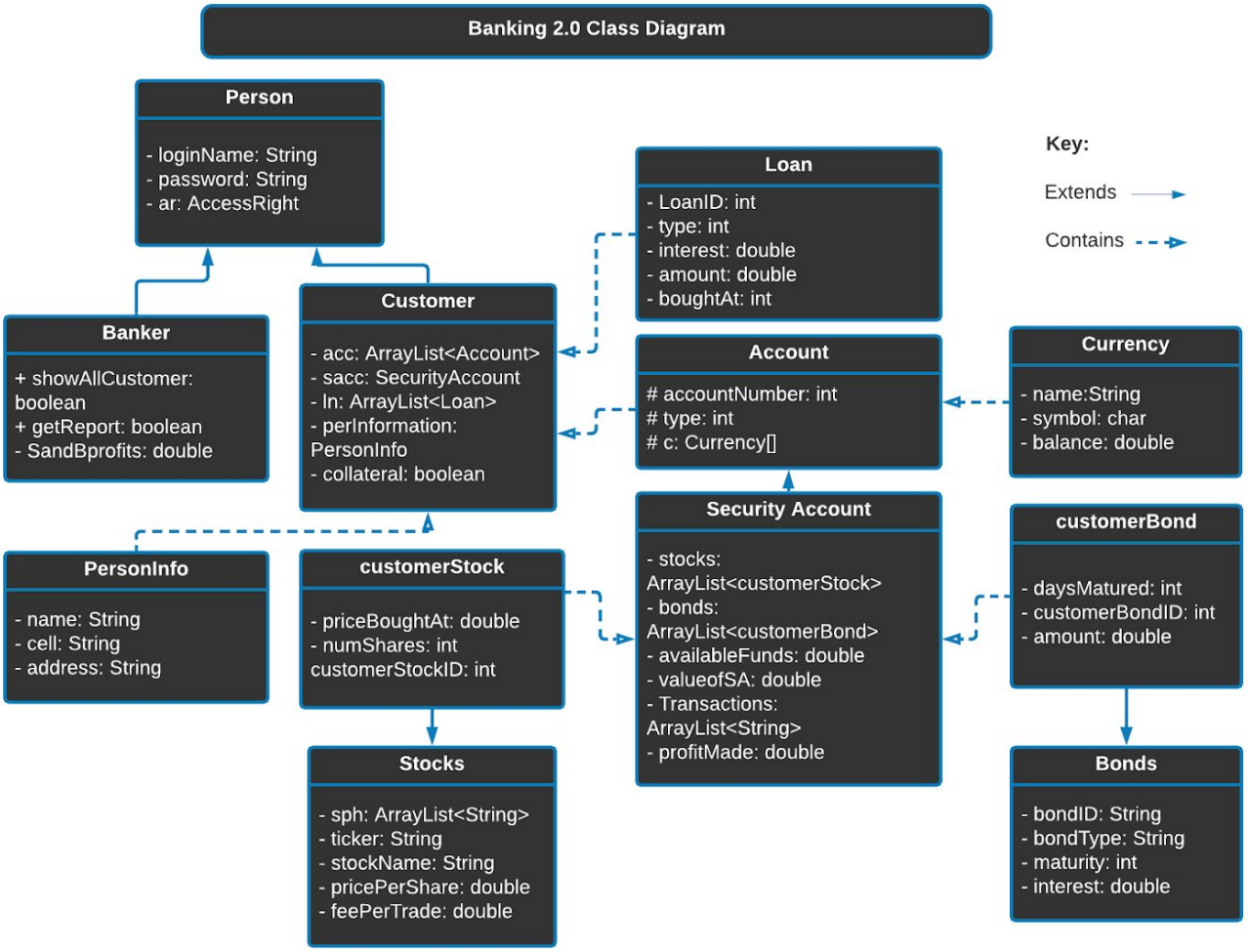
Team 6:

Team Members: Sean Chun, Xuan Hy Nguyen, Yancheng Liu, Haiqi Ma



Documentation:

Structure:

The biggest key to our class structure was understanding what needed to be stored and what each person has access to. Thus, we started with Person class and extending it to the people that would be accessing our interface, Customer and Banker. The banker was easy to implement as there was not much we needed to keep track of. However, within the Customer object, there were multiple variables and objects that each customer should have access to.

Design:

Firstly, every customer has the ability to create a savings or checking account. We decided to store this within an ArrayList as there could be multiple accounts that the Customer has access to. This ArrayList will prove essential in determining whether a Customer can make a Security Account.

Many of Alex's previous implementations had to be reformatted for the database and to fit our new project goals, but the key to our project was implementing a Security Account for each customer such that they have the ability to make money.

Within the Customer's security account exists an ArrayList of stocks, an ArrayList of bonds, an ArrayList of all transactions within the SA and the value of the SA. In the security account, every customer should have the ability to purchase as many stocks and bonds as possible, given that they have the necessary funds to purchase these objects. Each of these purchasable objects made it necessary that we create a class for each.

Problems:

However, during implementation, we noticed that we needed to separate the difference between (Stocks and Bonds) and (Customer Stocks and Customer Bonds). This is because stocks and bonds would be interacting with both the Banker and the various Customer so we needed to differentiate what each object should be able to do.

Firstly, for Bonds, every bond should have a BondID, BondType, Maturity, and interest rate. However, for a Customer, it is necessary for them to know how many days the bond has matured and the total amount of the bond they have purchased. Thus, we added these attributes to a CustomerBond class to separate what's accessible to a Customer and a Banker. Similarly for a Stock, a banker and customer should all know what the ticker, price and fee per trade of a stock should be. However, the customer cares about at what price the stock was bought and the number of shares currently owned. So from this analysis, we decided to split Stocks into a stock class and stock customer class.