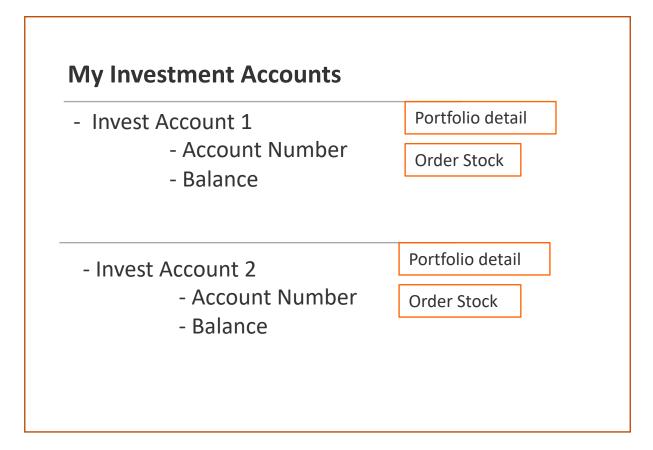# HCL-ING Hackaton

▶ 29th May 2021

Investments

Customer login (screen 1)

Login ING web

- Customer id
- Password

Continue

Source: any customer from ING customers database and its password
Build a backend solution to make login into the application.

ING

Overview ING Investment (screen 2)

My Investment Accounts

- Invest Account 1
  - Account Number
  - Balance

Portfolio detail

Order Stock

- Invest Account 2
  - Account Number
  - Balance

Portfolio detail

Order Stock

Source: ING Investment Accounts from the database
Build a backend solution to make the info available to the front end.

ING

Overview Portfolio Details (screen 3)
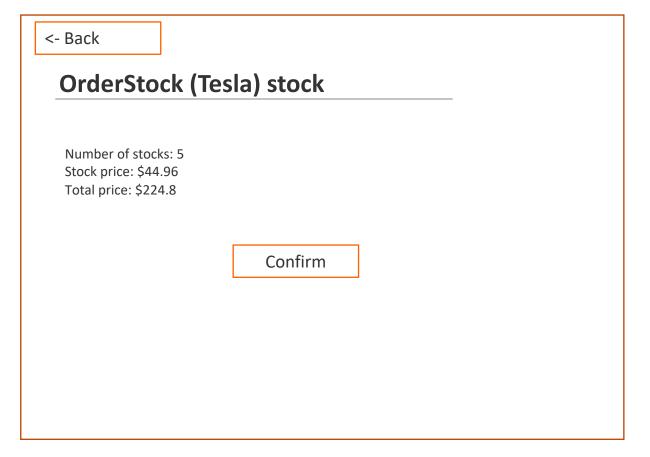
Investment Account 1

- Total Portfolio value (As of today)
- Balance Amount to invest (static)

- **Portfolio Details**
    - Apple
        - quantity
        - price (API/CSV?)
        - value (quantity*price)
    - Tesla
        - quantity
        - price (API/CSV?)
        - value (quantity*price)

Source: : Stock price from any datasource/ Third party API
Build a backend solution to make the info available to the front end.
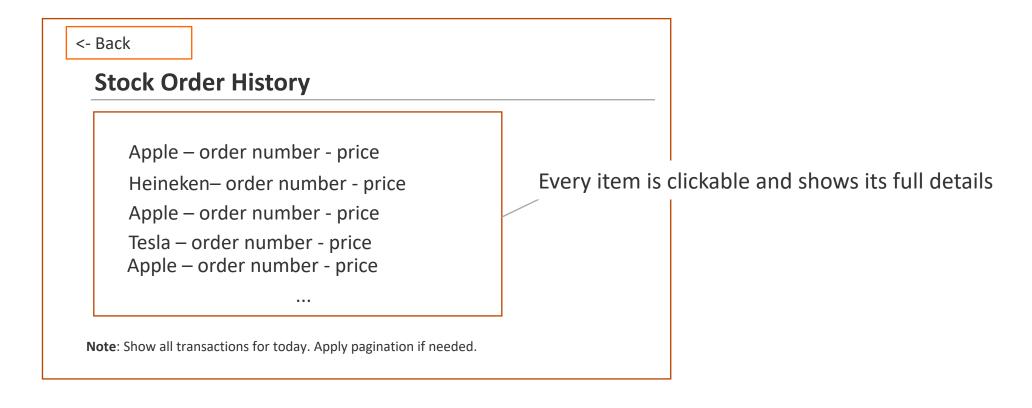
Order Stock (screen 4)

<- Back

OrderStock (Tesla) stock

Number of stocks: 5
Stock price: $44.96
Total price: $224.8

Confirm

Source: Stock's latest price to be obtained from any datasource/ Third party API and money should be be transferred from investment account.

Error Handling: In case of any error the user should be notified with details.

ING

Stock Order History Details (screen 5)

<- Back

Stock Order History

Apple – order number - price

Heineken– order number - price

Apple – order number - price

Tesla – order number - price
Apple – order number - price

...

Every item is clickable and shows its full details

Note: Show all transactions for today. Apply pagination if needed.

Source: Details to be obtained from the database

ING

Criteria to be taken into account

- Clean code (Readibility, Mantainability)
- Incremental delivery

- Functional programming approach with Java >= 8 (must)
- Spring (must)
- Spring Boot
- Swagger

- Proper DB model
- Proper REST guildelines implemented

- Test Driven Development
- Unit Tests (With Junit) (must)
- Create different tests types (UT/IT/CT) (Optional)
- Code coverage by test

ING