# Skill Assessment Test

(Programmer / Full-stack Developer / Software Engineer / Computer Engineer / Computer Scientist)

## Objective

To assess the applicant's basic programming skills, including:  
- UX Design  
- UI Design  
- Basic System Analysis (SA)  
- Programming in web, desktop, or mobile applications (choose one)  
- Communication and presentation of work

## Time Limit

No fixed time limit. Time starts from when the test is received until the work is presented via Google Meet once finished.

## What to Submit

- A brief presentation of BA, SA, UX/UI  
- A working demo of web app / desktop app / mobile application (choose one), built using any programming language, database or tech stack the applicant is comfortable with  
- Present the project via Google Meet  
  
Note: You do not need to complete everything. Do as much as you can. Present what you have accomplished.

# Order Matching System

## Create SQL Database Tables

users table  
- user\_id (Primary Key, auto-increment)  
- name  
- email  
- phone

order\_types table (two types: buy, sell)  
- order\_type\_id  
- name

orders table  
- order\_id (Primary Key, auto-increment)  
- user\_id  
- product\_id  
- order\_type (buy/sell)  
- price  
- volume  
- timestamp

matchings table  
- matching\_id (Primary Key, auto-increment)  
- seller\_user\_id  
- buyer\_user\_id  
- product\_id  
- price  
- volume

products table  
- product\_id  
- name

## Create a User Interface

A form for users to submit buy/sell orders, saving the data into the orders table.  
  
Form Fields:  
- User  
- Order Type: (Buy / Sell)  
- Product  
- Price ($)  
- Volume (kg)

## Write a Program to Match Orders

Implement a limit order matching algorithm (price-time-volume) to match buy and sell orders and store the results in the matchings table.

## Example Matching Scenario

Assume we're processing orders for product\_id = 1 (forward contract)  
  
Initial Buy Orders:  
1. Buy order #1: 5 kg @ $6.45 (10:01 AM)  
2. Buy order #2: 8 kg @ $6.43 (10:02 AM)  
3. Buy order #3: 3 kg @ $6.42 (10:03 AM)  
  
New Sell Order:  
4. Sell order #4: 10 kg @ $6.40 (10:05 AM)  
  
Matching Logic:  
Since the sell price ($6.40) is lower than the highest buy price ($6.45), a trade will occur.  
  
Priority Rules:  
- Price Priority: Match with highest buy price first  
- Time Priority: Earlier orders at the same price are matched first  
- Volume Priority: Larger volumes may take precedence if price and time are equal (not applicable here)  
  
Matching Steps:  
- Sell order #4 matches 5 kg with Buy order #1  
 - Trade executed: 5 kg @ $6.45  
 - Buy order #1 is fully filled  
 - Sell order #4 has 5 kg remaining  
- Remaining 5 kg of Sell order #4 matches with Buy order #2  
 - Trade executed: 5 kg @ $6.43  
 - Buy order #2 has 3 kg left (partially filled)  
 - Sell order #4 is fully filled  
  
Final State of Orders:  
- Buy order #2: 3 kg remaining @ $6.43  
- Buy order #3: 3 kg @ $6.42 (unchanged)

## Matching Result Display Page

Display the matched orders with the following fields:  
- Matching ID  
- Seller ID  
- Buyer ID  
- Product ID  
- Price  
- Volume