CMPE 343 – Fall 2025-2026 – Course Project # 3

Topic: Local Greengrocer Application

Reviewers: Asst. Prof. Dr. İlktan Ar, Berkay Mustafa Arıkan, Çağatay Batuker, Doruk Ege Aksu, Fatih Gölge, Ozan Kutlar, Selin Budak, Uğurkan Bekit, and Volkan Erdoğan.

Brief of Project and Important Dates

In Project 3 of the CMPE 343 course, you will build a desktop application for a Local Greengrocer based on JavaFX GUI and JDBC. All the related lecture notes can be found in the Learn system. You will generate documentation with JavaDoc tool that allows for the automatic generation of HTML documentation for your code and is the standard for Java developers. The team will collaborate on the code using GitHub, and you will share your source code with the related assistant. Further details about the project are under the following headers: Scenario-Objectives, Important Notes, Deliverables, and Grading Formula.

Important dates for the Project 3 are:

- 9. December.2025 Declaration of the Project-3 via Learn.
- 2. January.2026 23:59 Due date/Cut-off date for upload of deliverables.
- 7-8. January.2026 09:00-21:00 Presentations @ D104.

Note that no late submission or excuse will be accepted. Students who uploaded their files to the Learn system can participate in the presentation.

Project Scenario-Objectives

Your app will welcome the user with a login interface. After entering the username and password pair correctly your application will close the login interface and will open a new interface based on the role of authenticated user such as customer, carrier, and owner. If the user fails authentication, your app will warn the user. Users can register themselves as a customer with this interface. During registration, collect necessary user details, such as username, password, and any other relevant information. Include validation checks for unique usernames and strong password requirements. Store user data in the database. Allow customers to edit their profile information (e.g., address, contact details).

In the customer interface, the user sees his/her username at the appropriate corner of the scene (top left, top right, bottom left, or bottom right) and the stage's title as "GroupXX GreenGrocer". The user can view available vegetables and fruits under different TitledPanes. These products are listed with name, image, and price. The lists must be sorted by product name. Note that there will be 12 vegetables and 12 fruits at least in the beginning of the project. The customer can enter the amount in terms of kilograms (0.5 kg, 1 kg, 2.25 kg, etc) for the requested product and add this item to the shopping cart if available in stock. If stock is not enough, your app will warn the user about stock size. If the user wishes, he may filter the products by a keyword, and the quick search bar can be used to efficiently locate a specific product. Note that the shopping cart will be on a different stage (interface). The shopping cart will list the added items' names, amounts, and prices with the total cost including taxes (VAT). The user can add/remove more items to the shopping cart or conclude shopping by selecting the requested delivery date and time (the user must select a delivery date within 48 hours at most after purchasing). Provide a summary of the order before finalizing the purchase. To complete a purchase, the customer must meet the minimum cart value requirement. If the customer possesses a discount coupon earned from previous purchases, it shall be applied to

the total amount. Additionally, the customer may be eligible for a loyalty discount based on their past completed transactions, and the conditions under which this discount is earned and applied must be explicitly stated. When the purchase operation is completed all related information will be updated/inserted on related tables in the database. The invoice must be stored in the database in PDF format and must also be shared with the customer. The user can also view his/her deliveries, and log out via this interface. Additionally, allow customers to cancel orders within a certain time frame and provide a history of past orders and delivery status. Customers may evaluate the carrier based on a rating system and may communicate with the owner through the application's messaging feature.

In the carrier interface, the carrier can view deliveries/orders in three different areas such as completed, available, current/selected. Available deliveries display the order details such as order ID, product list, customer name, customer address, total including V.A.T, requested delivery date... Carriers can select any (one or many) available orders. Selected orders will be transferred to the current/selected area. After the carrier delivers the products and gets the money, completes the order by entering the delivery date.

In the owner interface, the owner can:

- add/remove/update products with related attributes (price, name, threshold,...),
- employ/fire carrier,
- view all orders (all types),
- view/reply the customer messages,
- set / adjust coupons, loyalty standards,
- view carriers ratings,
- view reports as charts based on product/time/money...

The threshold amount defines when the given prices will be doubled! For example, a threshold of 5kg for potatoes means that the customer will see the doubled price if the current stock of potatoes is equal to or less than 5kg. Yes, the owner is greedy ☺

Images related to the products should be stored in the database as Binary Large Objects (BLOBs). Transaction logs/invoices should be stored in the database as Character Large Objects (CLOBs).

Important Notes

This work is not homework, it is a **project**. Comprehensive study and research will be your priority. You will use SceneBuilder to form different .fxml files for GUIs. You will use MySql as your database. Your application will work on a **local network!**. The initial size of the GUIs will be 960x540 pixels, centered on the screen. When you maximize the app window, the scene and nodes will increase the size respectively. The Back-end and Front-end source codes must be separated like in the examples (controllers are important) we reviewed in the lectures. There will be at least **six different event handlers** in this project.

All groups will connect to the MYSQL db server using the same username and password pair. un: **myuser@localhost** PSW: **1234**

Your project will include these users at least (you can add more):

- un: **cust** PSW: **cust** role: customer**,**
- un: **carr** PSW: **carr** role: carrier,
- un: **own** PSW: **own** role: owner**.**

Your database may contain different tables such as UserInfo, ProductInfo, and OrderInfo...

- UserInfo table holds id, username, password, role, address,...
- ProductInfo table holds id, name, type, price, stock, imagelocation, threshold...
- OrderInfo table holds id, ordertime, deliverytime, products, user, carrier, isdelivered, totalcost...

There is no limit on the count of classes, methods, and attributes in your application. Higher is better. You will use the database-adapter class to operate on the database.

Please make sure to use Object-Oriented Design principles while creating classes and other components. This project encourages the application of encapsulation, inheritance, and polymorphism, promoting a structured and efficient system design.

Please make sure your code runs without errors when you submit your projects.

Some logical errors which are not handled (with try-catch blocks or other techniques) such as:

- Entering zero or negative amounts in customer GUI for product amount,
- Entering non-double value (can be char, string, etc.) in customer GUI for product amount,
- Displaying products with zero stock in customer GUI,
- Not merging the same product in the shopping cart (When the customer added 1.25 kg tomatoes, then added 0.75 kg tomatoes in the same order, the customer will view tomatoes 2 kg as merged in the shopping cart, not separate!)
- Selecting the order selected by different carriers in carrier GUI,
- Threshold values are not working,
- Entering zero or negative threshold value for a product in owner GUI...

can cause early termination of your presentation and demo!

Deliverables

Each student will upload project files to the learn.khas.edu.tr via assignments tab:

First - .zip file (GroupXX.zip) that contains:

- zip file containing java source codes of the project (GroupSourceXX.zip),
- zip file containing Javadoc documents related to your code (GroupDocXX.zip),
- Database Export: The MySQL exported .sql file containing the database schema and related (at least 25 rows for each table) data (GroupXX.sql),
- A video formed using screen capture. This video will demonstrate all operations of the application (including all roles, database display, and testing of various data input types) within a maximum of 8 minutes (GroupXX.avi, GroupXX.mp4, GroupXX.mpeg, or GroupXX.mpg),
- Brief instructions on how to run your code: step by step in intro.txt,
- zip file containing FXML files for JavaFX application user interfaces (GroupFxml.zip),
- Peer scoring GroupXX.txt (you will assign scores to your teammates out of 5),
- zip file containing all the related image files (GroupImagesXX.zip).

XX is the number of your group 01, 02,...,30

If a student fails to upload the GroupXX.txt he/she will not be eligible to receive a group score for the project.

Grading Formula

We will use the below formula to form your score for Project 3:

$$YourScore = [w_1 \times (GroupScore) + w_2 \times IndividualPerformance]$$

while w_1 is 0.3, and w_2 is 0.7 for this project.

IndividualPerformance indicates your performance on the Live Demo & QA which ranges between 0 and 100. GroupScore indicates the Completeness, Correctness, and Consistency of the uploaded files which ranges between 0 and 100.

Presentation Details

Each group member is required to participate in the presentation. There will be no make-up presentations..

Reviewers may interrupt the presentation at any point and ask other members to continue. Additionally, the reviewers will ask questions to each team member regarding the project's implementation, their individual contributions, and related course topics.

The presentation schedule will be available on learn.khas.edu.tr (23.December.2025).