DATABASE SYSTEMS
Assignment 3: Application Development

Opened	2018-11-27
Contact	정근성 (aninteger@hanyang.ac.kr, Discord Geunseong Jung#5285)

한양대학교
HANYANG UNIVERSITY

tl;dr

- '**Delivery Hanyang**' App developing scenario
  - This app (or service) links merchants (seller), store, customers, and delivery personnel.
  - You are going to develop application with given database.

- TODO (Point: Basic 150pt, Optional 50pt)
  - <u>User Login feature</u> (10pt)
  - <u>*Seller* feature</u>: *Store*, *Menu*, and *Order* Manager (40pt)
  - <u>Customer feature:</u> User info, placing an *order*, *order* history (40pt)
  - <u>Deliveryman feature:</u> Current Status (20pt)
  - Submit file: Program, <u>Description document(40pt)</u> in <u>ONE .zip file</u> named *NAME_ID_DB01*.zip (e.g. Jones_2018123456_DB01.zip)
  - Optional in Customer feature: Geolocation feature
    - Optional point will be added your past assignments (#1 and #2 as 200pt total). If you have got full point already, it will be discarded.

- Submission deadline
  - Due to Dec. 16th
  - Additional submission: Dec. 18th
  - Additional submission penalty: 50%
  - Overdue penalty: 100%

App scenario

- A *Seller* has one or more *Store*(s).
- A *Store* has one or more *Menu*(s).
- A *Customer* can search stores through your app.
- A Customer can have payment methods.
  - Payment type: account, card
- A Customer can pick *menus* (from single store) to their cart.
- After choosing menus, customer can place an *order* only if they have any payment method.
- The store check and review the order, and confirm or decline the order.
  - If the order is confirmed, the order will be assigned deliveryman nearby. If not, the order will be canceled.
- The customer can notice the delivery status via your app.
  - If the order is confirmed by the store, the customer can accept the delivery and close the order process.

Login feature

- Your should check email and password to confirm the user.
- You should <u>NOT</u> ask the usertype(seller, customer, or deliveryman) before logging in.
  - After logging in, you can ask the user their usertype or show available feature in one page, anything is okay.

Seller feature

- *Seller* feature:
  - Store list (to each store menu)
  - User info manage
    - Changing password
    - Changing name

Store feature

- *Store* Overview:
  - Store info:
    - Store summary (editable)
    - Store tag (editable)
  - Current menus:
    - Adding, editing, and deleting menus
- Current orders:
  - Order list
  - Order confirm/decline
    - Confirm Process:
      - Show the list of the 5 closest delivery man with under 4 stocks.
      - User select delivery man and the order status changed in 'confirmed' with the information of the assigned delivery man.

- Notice
  - The features above can be displayed on seller features, but the stores should distinguished each other.

Customer (1)

- Customer Overview:
  - User info manage: Same as seller's.
  - Shipping list (editable)
    - If the customer has no shipping list, they cannot proceed to payment. But they can add the menu to their cart.
- Order feature:
  - Order history
    - Store, menus, payment method, timestamp
    - Sorted by latest
  - Current Order
    - Order status, store, menu, payment method, timestamp
    - The order in 'delivery in progress,' the delivery man information should be added, and the user (customer) can accept the delivery.
    - Accepted(closed) order will move to order history.
- Payment feature:
  - Payment methods (editable)

Customer (2)

- Search stores:
  - List nearby <u>open</u> stores
    - Use coordinate from customer DB
    - (Optional 15pt) Use current location (via geolocation service, etc.)
    - (Optional 35pt) Show user and stores position on Map API
  - Search stores by address
  - Search stores by tags
  - Search stores by name
- Place an Order
  - Cart: User can put menu(s) of <u>a single store</u> and their quantities in cart
  - User can proceed if they have a payment method
  - Pay
    - Choose own payment method
    - If payment is completed, the order will place and the store can receive the order. (In the scenario, there is no payment failure.).

Deliveryman

- Deliveryman feature:
  - List current delivery
    - Store, customer info (name, phone, location, etc.)
    - Order info
      - Menus
      - Timestamps

Initial DB

- DB \copy command
  - \copy {table_name} from '{.csv file}' with csv header encoding 'utf8';
- Seller
  - Seller_id(PK), name, phone, [local, domain] (email for login), passwd
- Store
  - Sid(PK), address, sname, [lat, lng] (store location), phone_nums (JSON), schedules, seller_id(FK)
  - Schedules (JSON):
    - Day: 0 for Mon, 1 for Tue, ···, 6 for Sun.
    - Holiday: If True, the store closes that day
    - Open, Closed: open, closing time in a format 'HHMM'

Initial DB

- Customer
  - Name, phone, [local, domain] (email for login), passwd, [lat, lng] (current location)
  - Payments (JSON)
    - Type: account, card
    - Data:
      - If type is account;
        bid: bank id from *bank*
        acc_num: account number
      - If type is card;
        card_num: card number
- Bank
  - Bid(PK), code (bank code), name (bank name)

Initial DB

- Delivery
  - Did(PK), {same on sellers'}
  - Stock: current # of delivery stock

- Menu
  - Sid(FK) from *store*
  - menu