

Deliverme

Samyam Pandey

Isabel Fernandez

Cesar Garcia

Soohwan Kim

Samyam Pandey: I worked on files order.php, delete_order_name.php, customer_delete.php, select_order_driver.php, driver_order.php

-order.php: in this file when the customer logs in and clicks on the order tabs it redirects to an orders page. It uses SELECT and INSERT in this page. In this file it asks the customer to add an order and when the user clicks submit it creates an order.

- delete_order_name.php: This is the page that comes up after clicking submit in the order.php file. In this page it just asks the user if they want to delete an order that was made and if they do. It asks them to type in their username. In this file it uses select to select the user id corresponding to the username the user typed.

-customer_delete.php: This page is the page that comes up after the user types in their username and clicks submit in the delete_order_name.php. In this page it shows the orders the user had made and it asks if the user wants to delete an order and if they do it asks them to type in the order id. This file uses DELETE to delete an order the user doesn't want any more.

- select_order_driver.php: in this file when the driver logs in and clicks on the order tabs it redirects to an orders page. In this page it shows all the orders that have not been selected by a driver. It asks the driver to enter the order id for the order he wants and the driver username. This file uses SELECT.

- driver_order.php: This is the page that comes up after clicking submit in the select_order_driver.php. In this page it shows the orders the driver selected from the orders that have not been completed. This file uses SELECT, DELETE, UPDATE. This file also asks the driver to type in the order id for the orders the driver has completed. After the driver has clicked on the completed button it deletes the order from the orders table because the order is completed.

Isabel Fernandez: I worked on the files index.php, customer_header.php, driver_header.php, customer_profile.php, driver_profile.php, edit_profile.php, edit_driver.php

- index.php: This file serves as the base customer login for our site, a user enters their username and password, and a SELECT occurs on the database to ensure that the entered username and password has an associated account. This page then leads into the customer_profile.php.
- customer_profile.php: This file serves as the profile page for a user, and is the first thing they see when they login. A SELECT occurs on the main landing page which takes the user_id found after login to retrieve the user's associated first and last name, and their associated address and then displays them on the page. There are two buttons on this page, one that says 'Edit Account', and another that says 'Delete Account'. Here if a user chooses 'Delete Account', a form is silently submitted that drops the user from all tables. This happens by first finding all the order entries associated with the user, and then looping through all of the list items associated with each order associated with the user and using DELETE to drop them all. Then all associated order entries are dropped using DELETE. After this is done the user can be safely dropped from the customer table using DELETE, and finally the user is completely deleted from the database by using another DELETE on the users table.

- `edit_profile.php`: When 'Edit Account' is pressed on the `customer_profile.php` page, the site routes to this file. In this file a user sees a form with all of the information associated with their account that they are free to edit. A user can fill out the form to update their information, and when the 'Save Profile' button is clicked, an UPDATE occurs on the database, first selecting the user based on their `user_id`, and then selecting their address, finally an UPDATE updates the user's address to whatever was filled out in the form, and another UPDATE updates the user's names and username / password in the users table.
- `driver_profile.php`: This file serves as the profile page for a user, and is the first thing they see when they login. A SELECT occurs on the main landing page which takes the `user_id` found after login to retrieve the user's associated first and last name, and how many orders are assigned to them and then displays them on the page. There are two buttons on this page, one that says 'Edit Account', and another that says 'Delete Account'. Here if a user chooses 'Delete Account', a form is silently submitted that drops the user from all tables. This happens by first finding all the order entries associated with the user, and dropping them using DELETE. After this is done the user can be safely dropped from the driver table using DELETE, and finally the user is completely deleted from the database by using another DELETE on the users table.
- `edit_driver.php`: When 'Edit Account' is pressed on the `driver_profile.php` page, the site routes to this file. In this file a user sees a form with all of the information associated with their account that they are free to edit. A user can fill out the form to update their information, and when the 'Save Profile' button is clicked, an UPDATE occurs on the database, using UPDATE along with the filled out information from the form, the user's account information is updated in the database.
- `customer_header.php` & `driver_header.php`: Both of these files serve simply as the navigation bars at the top of the pages that pass along the `user_id` to every page.

Soohwan Kim: I worked on the files `register.php`, `driver_register.php`, `review.php`, and `driver_review.php`

- `register.php`: This file displays a form for registering new customers. The form contains fields for personal details, account information, and address. When the user submits the form, it parses the data given by the user, and performs an INSERT query to the users table, customer table, and address table. Additionally it connects the user entity with the customer entity.
- `driver_register.php`: This file displays a form to register new drivers. The form contains fields for the driver's personal details and account information. When the user submits the form, it parses the data given by the user, and performs an INSERT query to the users table and drivers table. Then it connects the user entity with the driver entity.
- `review.php`: The main purpose of this file is to allow the customers to leave reviews for their driver. This file uses SELECT queries to display a list of orders associated with a customer's account, including the order date, the driver assigned to the order, each review associated with the driver, and the store where the order was placed. Users can submit a text for each driver, which performs an INSERT query to the review table in the

database. Users can also edit their reviews by clicking the edit button, which performs an UPDATE query to the review table in the database. Users can delete their reviews by clicking the 'x' button, which performs a DELETE query to the review table in the database.

- Driver_review.php: The main purpose of this file is to allow the drivers to leave a reply to each review they got. This file performs a SELECT query to retrieve driver's information, and reviews associated with the driver. Drivers can submit a reply text, edit a reply, and delete a reply, which performs UPDATE query to the reviews table and updates the reply attribute in the table.

Cesar Garcia: I worked on the following files: admin_login.php, admin_profile.php, admin_user_orders.php, admin_user.php

- Admin_login.php: This is the basic login for an admin. An admin is not a user and anyone can log in as an admin as they just require a username and password. It verifies an admin with that username and password exists using SELECT. This directs the page to admin_profile.php
- Admin_profile.php: This page shows all the users in the database whether they are a driver or customer. An admin can add a user, alter a user, delete a user, or see a customer's orders if they have any. The add user functionality uses an INSERT, deleting a user uses a DELETE, and altering a user uses an UPDATE. Clicking on adding a user brings up a few text boxes that lets the admin input the username, password, first name, and last name for the new user. Altering a user brings the user to the admin_user.php page. Clicking on the user's username brings the user to the admin_user_orders.php page.
- Admin_user_orders.php: This page shows all the orders for a particular user. It shows the entire order and allows an admin to delete the order if they click on it.
- Admin_user.php: This page allows the admin to see a user's first and last name and lets him alter them if he chooses to, using an UPDATE. Clicking on the change first name allows admin the change the first name and he can do the same thing with the last name button.