Assignment 3

Overview

In this application, we will once again implement the water delivery system but this time we will be using a MongoDB database to store information and use an HTML presentation layer implemented using ExpressJS.

The code will be implemented using a standard 3-tier closed layered architecture without any need for the "currency" layer from the previous assignment.

Presentation > Business > Persistence

Starting Code

There is no starting code for this assignment, but you are free to utilize your work from the previous assignment.

Coding Restrictions (read carefully)

- The ExpressJS application must be hosted on port 8000.
- Styles must be defined in a separate .css file. You must use the express.static() function to serve the static resources.
- You must use the handlebars-express template engine.
- You are permitted (and encouraged) to use a layout but it is not mandatory.
- The use of JavaScript higher-order array methods such as "forEach", "map", "reduce", "find", etc. are not permitted in this assignment. A 25% penalty will be applied for solutions using these constructs. Use only plain loops (for loops, while loops, etc.) for this submission.
- All code must be properly indented.
- JSDoc style function comments must be provided for all functions.
- You must use the strict equality and inequality operators (===, !===) when comparing data.
- Variable names must be meaningful.
- The use of client-side JavaScript for any purpose is not permitted and a score of zero will be recorded for any submissions utilizing client side JavaScript for any purpose. If in doubt, right click on the browser and pick View-Source. If you see JavaScript then you have client side JavaScript.

MongoDB Database Configuration

- The database must be called 'water'
- The collections must be called 'customer' and 'delivery'.
- You can load the existing JSON files into the MongoDB database as is. Please keep the "id" field for the customer in addition to the _id generated by MongoDB when you load the data.
- You must configure your software to use connection caching so that performance of the solution is acceptable after the first connection to the Atlas cloud database.

INFS3201 – Web Technologies II
Due: March 6, 2025 @ 1159pm

- Make the database connection string easy to change. Create a constant in your persistence file so the database can be changed in one place. Do not make the instructor change the connection string in multiple places.

Add appropriate indexes to your database for your application and include a screenshot from MongoDB showing which collections/fields have had an index added. Failure to provide this screenshot will result in a reduced grade. Keep the screenshot in a folder called 'screenshots'.

HTML Screens

This application requires some basic HTML styling but nothing beyond what has been mentioned. You are free to add additional styling to make the application look better but there are no points associated with this. You must however use the links and buttons as shown in the screens (you cannot switch the links to a button for example).

Landing Page (i.e. /)

Water Management

Customer Name	Phone Number	Coupons		
Aisha Khalil	9123-4567	4	Add Coupon	Deliver 1 Bottle
Omar Farouk	8345-6789	0	Add Coupon	Deliver 1 Bottle
Fatima Al-Mansoor	7456-1234	3	Add Coupon	Deliver 1 Bottle
Ali Hassan	6789-0123	1	Add Coupon	Deliver 1 Bottle
Layla Nasser	5643-2109	10	Add Coupon	Deliver 1 Bottle

Delivery History

Date: dd / mm / yyyy ☐ Get Deliveries

- The row(s) that have zero coupons must be highlighted in yellow.
- If the user clicks on the "Add Coupon" button, the number of coupons is increased by 10. However, the number of coupons is limited to a maximum of 10. If there is an error increasing the number of coupons, you can just use a "res.send()" call to display a simple error message. If the additional coupon is successful then the user needs to be shown the landing page again (through a redirect)
- Clicking on the "Deliver 1 Bottle" will cause the number of coupons to be reduced by 1 and a new record being added to the database collection. After performing the operation, redirect the user to the landing page again.
- Clicking on the customer's name will bring the user to the "Customer Detail" page.
- Picking a date and clicking on "Get Deliveries" will bring the user to the Delivery Details page.

Customer Details

- You must use a URL parameter for this page.
- Sort the delivery dates with the **most recent date at the top** of the table (sorry… the image below shows it the opposite!)

← → C ⓘ 127.0.0.1:8000/customer-details/4831

Customer Details

Name:	Aisha Khalil
Phone:	9123-4567
Coupons:	4

Previous Deliveries

Date	Bottles
2024-12-15	3
2024-12-25	2
2025-01-24	1

Delivery Summary

- You must use a query parameter for this page.

← → C ⓘ 127.0.0.1:8000/deliveries?date=2024-12-10

Deliveries on 2024-12-10

Customer	Bottles
Ali Hassan	4
Fatima Al-Mansoor	4
Omar Farouk	1

Late Submissions

Late submissions will be accepted until March 8 at 11:59pm with a penalty of 50%. No submissions after this time will be accepted.

INFS3201 – Web Technologies II
Due: March 6, 2025 @ 1159pm

Grading

Criteria	Exemplary	Satisfactory	Developing	Unsatisfactory
Persistence Layer	6 points The persistence layer is using MongoDB. All operations are efficient. Indexes created and used as necessary.	5 points MongoDB is being used but indexes are not created or not demonstrated. Operations are acceptable. Additional unnecessary indexes were created	3 points Database operations are overly complex such as using looping to update multiple values rather than a single query.	0 points The persistence layer is using JSON files or the data has been loaded into the collections in a way that is not feasible to be used by MongoDB.
Presentation Layer	8 points All functionality is implemented and works as expected. Template engine is used.	6 points A minor functionality item is missing or flawed. Template engine is used.	4 points A major functionality feature is missing, or multiple minor problems exist. Template engine is used.	0 points Many major functionality features are missing or do not function. Template engine is not used.
Code Structure	1 point All functions and variables are well named. Formatting of the code is excellent. JSDoc style comments provided	0 points Formatting issues or missing/malformed comments.	0 points Many functions and variables have inappropriate names and/or significant parts of the code need additional formatting	0 points Most variables and functions have poorly chosen names. Significant formatting issues.