

# JS Assignment3 – Thanos Apostolou

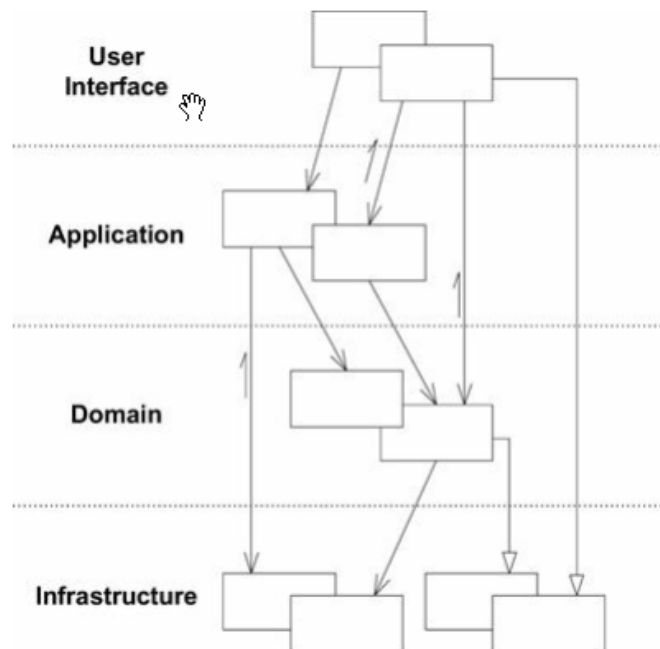
του Αποστόλου Αθανάσιου (mpsp2203)

## Εισαγωγή

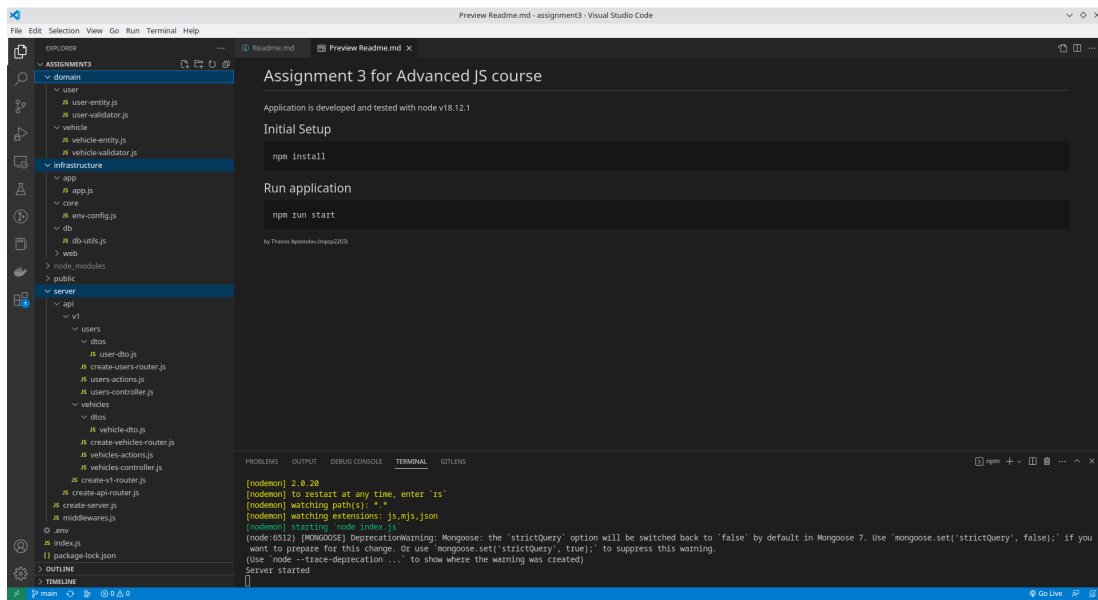
Στην εργασία έχουμε υλοποιήσει ένα API με CRUD operations για 2 collections. Το 1ο είναι collection είναι το **User** και το 2ο το **Vehicle**. Ο κώδικας ακολουθεί το architectural pattern MVC χωρίζοντας τις λειτουργίες των controllers, και models καθώς δεν έχουμε view αφού δεν ζητείται ui από την άσκηση (μπορεί να θεωρηθεί ως view το json που επιστρέφουν οι controllers αλλά είναι πολύ trivial για να σπάσει σε ξεχωριστό μέρος της εφαρμογής). Ωστόσο, ο κώδικας δεν είναι οργανωμένος by types/kinds όπως συνηθίζεται (δηλαδή σε φακέλους **controllers**, **models**, **services**, κλπ) αλλά είναι οργανωμένος σε ένα layered architecture εμπνευσμένο από το Domain Driven Design. Υπάρχουν 3 βασικά layers τα **infrastructure**, **domain** και **application** (φάκελος “server”) χωρίς να υπάρχει user interface layer.

*Domain-Driven Design Tackling Complexity in the Heart of Software [Eric Evans]*

### Layered Architecture



Screenshot από τον κώδικα μας:



Για τον http server χρησιμοποιούμε **express js**.

Για την βάση χρησιμοποιούμε **mongo DB** μέσω της βιβλιοθήκης **mongoose**. Τα στοιχεία που χρειάζονται για την σύνδεση με την βάση ρυθμίζονται από το αρχείο **".env"** που διαβάζεται από την εφαρμογή μας μέσω της βιβλιοθήκης **dotenv**. Η αρχικές ρυθμίσεις είναι για σύνδεση με ένα mongo db cluster μέσω του Atlas προσβάσιμο από όλες τις ip.

Το αρχείο **assignment3.postman\_collection.json** μπορεί να γίνει import από το Postman και περιέχει τα requests που υλοποιήθηκαν για το testing της εφαρμογής.

Ο φάκελος **screenshots** περιέχει τα screenshots που χρησιμοποιήθηκαν σε αυτό το report.

Για το Collection **Vehicle** χρησιμοποιήσαμε και πεδίο εικόνας (**photo**) ως encoded base64 string. Κατά το create/update μπορεί να χρησιμοποιηθεί ένα site όπως το <https://codebeautify.org/image-to-base64-converter> για convert μιας εικόνας σε base64 format και copy paste το base64 string στο αντίστοιχο πεδίο στο Postman. Αντιστρόφως μπορεί να χρησιμοποιηθεί το site <https://codebeautify.org/base64-to-image-converter> για την αντίστροφη μετατροπή του base64 **photo** πεδίου σε εικόνα στα get-all-vehicles ή get-vehicle-by-id flows.

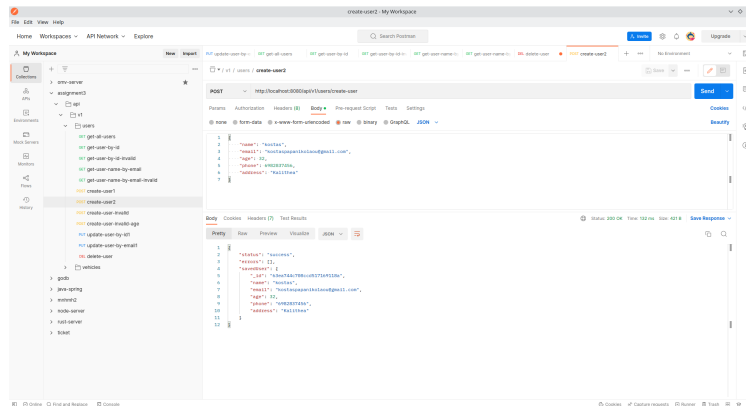
# User Collection

The Controller for all **User** requests is: **server/api/v1/users/users-controller.js**

## 1. Create:

url: <http://localhost:8080/api/v1/users/create-user>

handler: **handleCreateUser()**

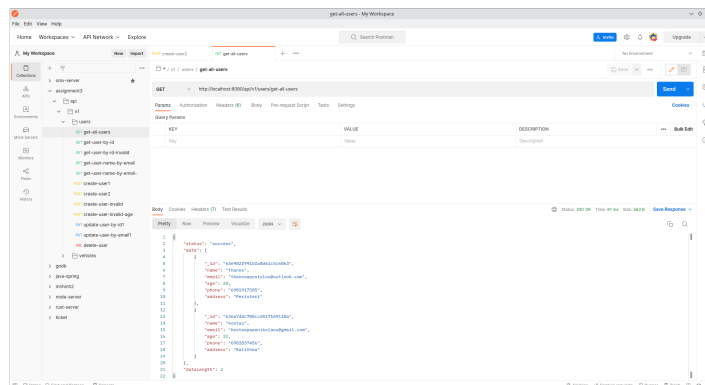


## 2. Read:

a. Όλες οι εγγραφές:

url: <http://localhost:8080/api/v1/users/get-all-users>

handler: **handleGetAllUsers()**



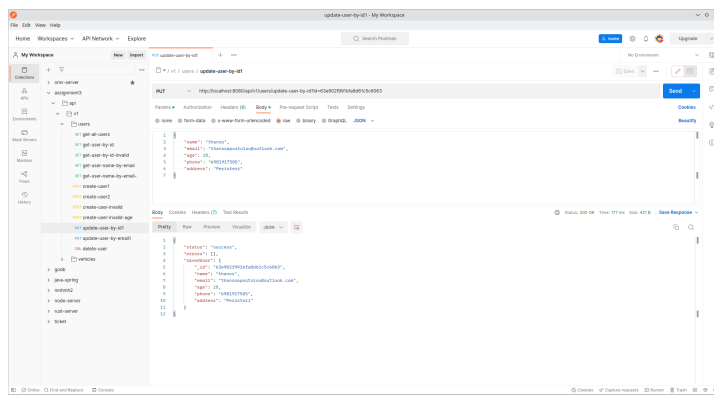
b. Μία εγγραφή

Χρησιμοποιώντας το **id** ως query parameter.

url: <http://localhost:8080/api/v1/users/get-user-by-id>

handler: **handleGetUserById()**

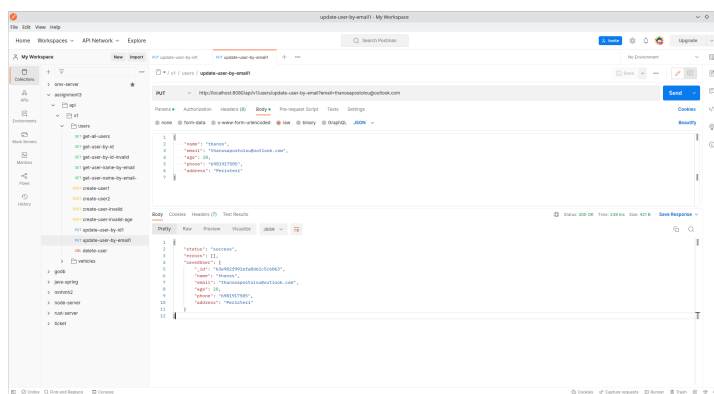
handler: **handleUpdateUserById()**



## a. Update by email

url: `http://localhost:8080/api/v1/users/update-user-by-email`

handler: **handleUpdateUserByEmail()**

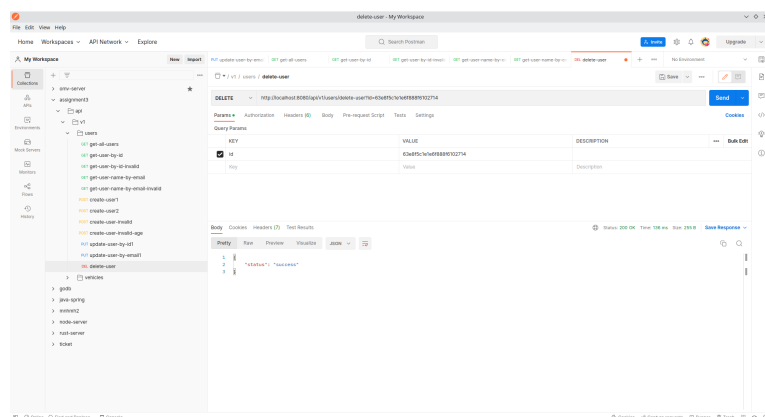


## 4. Delete

Χρησιμοποιώντας το **id** ως query parameter.

url: `http://localhost:8080/api/v1/users/delete-user`

handler: **handleDeleteUser()**



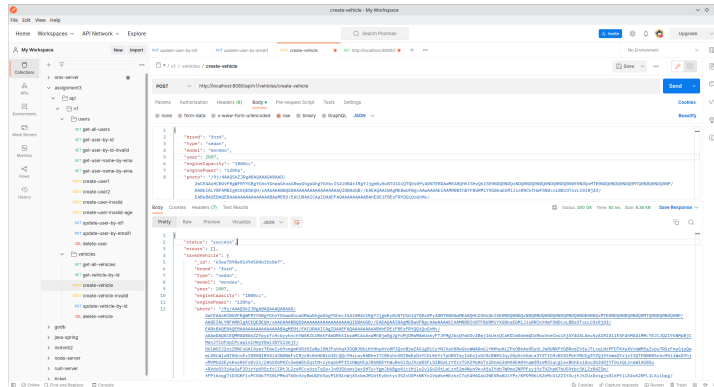
# Vehicle Collection

The Controller for all **Vehicle** requests is: **server/api/v1/vehicles/vehicles-controller.js**

## 1. Create:

url: <http://localhost:8080/api/v1/vehicles/create-vehicle>

handler: **handleCreateVehicle()**

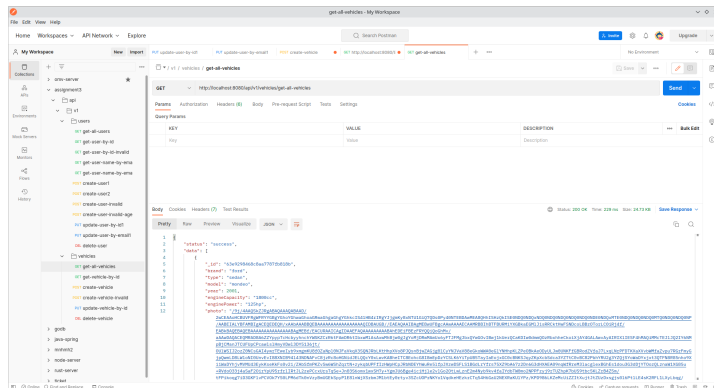


## 2. Read:

### a. Όλες οι εγγραφές:

url: <http://localhost:8080/api/v1/vehicles/get-all-vehicles>

handler: **handleGetAllVehicles()**

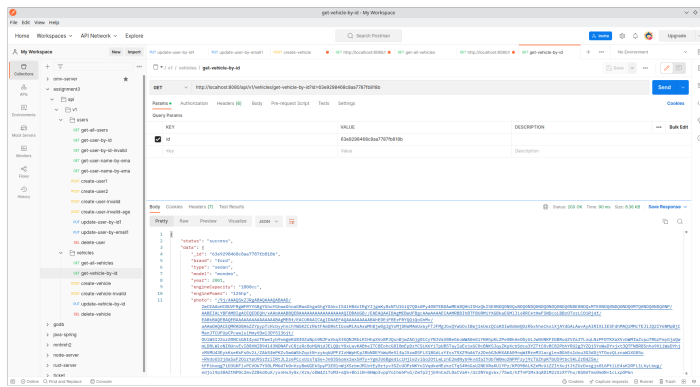


### b. Μία εγγραφή

Χρησιμοποιώντας το **id** ως query parameter.

url: <http://localhost:8080/api/v1/vehicles/get-vehicle-by-id>

handler: **handleGetVehicleById()**

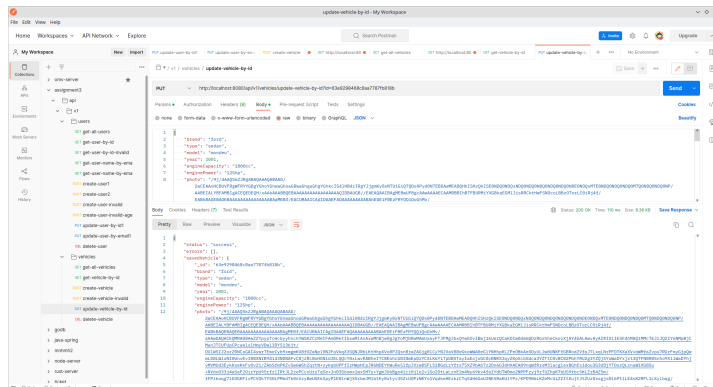


### 3. Update

#### a. Update by id

url: <http://localhost:8080/api/v1/vehicles/update-vehicle-by-id>

handler: **handleUpdateVehicleById()**



### 4. Delete

Χρησιμοποιώντας το **id** ως query parameter.

url: <http://localhost:8080/api/v1/vehicles/delete-vehicle>

handler: **handleDeleteVehicle()**

