Routing Table

Name	Endpoint	HTTP Method od	Purpose	Handler	Example (replace localhost with external IP of 'fit2095a2-node' vm if using GCP)
Retrieve all drivers	/34279075/Justin/api/v1/drivers	GET	API to retrieves all drivers in the database	driverController.getAll	localhost:8080//34279075/Justin/api/v1/drivers

Add driver	/34279075/Justin/api/v1/drivers	POST	API to add a new driver to the database	driverController.createOne	localhost:8080//34279075/Justin/api/v1/drivers
					{
					    "driver_name": "Alexa",
					    "driver_department": "Food",
					    "driver_license": "ABC13",
					    "driver_is_active": **true**
					}

Remove driver	/34279075/Justin/api/v1/drivers?id	DELETE	API to remove an existing driver from the database	driverController.deleteOne	localhost:8080/34279075/Justin/api/v1/drivers?_id=66e3dd4a4189a1aa1b3b01ce

Update driver	/34279075/Justin/api/v1/drivers	PUT	API to updates an existing driver's	driverController.updateOne	localhost:8080/34279075/Justin/api/v1/drivers
					{
					    "_id": "66e3dcf14189a1aa1b3b01cb",
					    "driver_department": "Electronic",
					    "driver_license": "SKBD1"

			informat ion		}
Retriev e all packa ges	/34279075/Justin/api/v1/pa ckages	GET	API to retrieve all package s from the databas e	packageController.get All	localhost:8080/34279075/Justin/api/v1/packages

Add packa ge	/34279075/Justin/api/v1/pa ckages	POST	API to add a new package to the databas e	packageController.cre ateOne	localhost:8080/34279075/Justin/api/v1/packages
					{
					    "package_title": "Chairs",
					    "package_weight": 4,
					    "package_destination": "Paris",
					    "is_allocated": **true**,
					    "driver_mongoose_id": "66e3dcf14189a1aa1b3b01cb"
					}

Remov e packa ge	/34279075/Justin/api/v1/pa ckages/:id	DELE TE	API to remove an existing package from the databas e	packageController.del eteOne	localhost:8080/34279075/Justin/api/v1/packages/66e3e14da fc4724cf05fc0e2

Documentation FIT2095 Assignment 2

Update package	/34279075/Justin/api/v1/packages	PUT	API to update an existing package in the database	packageController.updateOne	localhost:8080/34279075/Justin/api/v1/packages
					
					{
					    "_id": "66e3e14dafc4724cf05fc0e2",
					    "package_destination": "Nebraska"
					}

Signup a user	/34279075/Justin/api/v1/signup	POST	API to sign up a user for the application	-	localhost:8080/34279075/Justin/api/v1/signup
					
					{
					    "username": "Justin",
					    "password": "abc123",
					    "confirmPassword": "abc123"
					}

Login a user	/34279075/Justin/api/v1/login	POST	API to login a user for the application	-	localhost:8080/34279075/Justin/api/v1/login
					
					{
					    "username": "Justin",
					    "password": "abc123"
					}

Signup a user	/34279075/Justin/signup	POST	Route to sign up a user for the application	-	localhost:8080/34279075/Justin/signup

Documentation FIT2095 Assignment 2

Login a user	/34279075/Justin/login	POST	Route to login a user for the application	-	localhost:8080/34279075/Justin/login
Statistics of CRUD operations	/34279075/Justin/statistics	GET	Route to view the statistics of CRUD operations	-	localhost:8080/34279075/Justin/statistics
Display signup page	/34279075/Justin/signup	GET	Method to display sign up page.	-	localhost:8080/34279075/Justin/signup
Display login page	/34279075/Justin/login	GET	Method to display login page.	-	localhost:8080/34279075/Justin/login
Logout	/34279075/Justin/login	GET	Method to logout current user.	-	localhost:8080/34279075/Justin/logout

Steps to run the application (locally):

1. Clone the repository at https://git.infotech.monash.edu/fit2095/fit2095-s2-2024/projects/jtan0435/Assignment-2.git
2. Open the project with a code editor / IDE.
3. Make sure node is installed on the system (e.g. by executing node -v)
4. Open the terminal and enter "npm install" or "sudo npm install" (Linux-based systems) if necessary.
5. Open main.js, and edit line 58 to be mongoose.connect('mongodb://localhost:27017/pdma').

6. To run the project, enter "node main.js".
7. Open the browser and navigate to "localhost:8080", and the page will be displayed.

Steps to run the application (with GCP):

1. Start up both VMs (fit2095a2-node, and fit2095a2-mongo) in GCP console.
2. Click SSH on the 'fit2095a2-mongo' VM, and wait until the terminal shows up.
3. Click SSH on the 'fit2095a2-node' VM, when the terminal shows up, enter cd Assignment-2.
4. Start the program by entering "node main.js".
5. Navigate to the page by entering http://<<external-ip>>:8080 in a new window, where external-ip is the external IP of 'fit2095a2-node' which is found in the GCP console. (ex: http://34.129.129.52:8080)