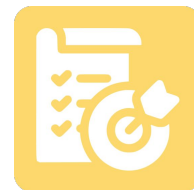


Introduction to the eCommerce Application

Relevel
by Unacademy



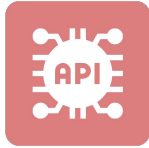
Class Agenda



- We will be introduced to the eCommerce Application backend and various scopes we have here to do.
- We will discuss the resource and their attributes for category table in database.
- We will Identify the RESTful URIs which needs to be created in this project.
- We will initializing the project structure and setup our database.
- We will get to the usage of .env file for setting the environment related parameters.

Educator Introduction

Let's first discuss on some basics

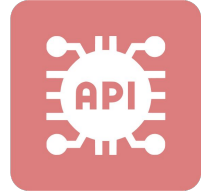


Rest API



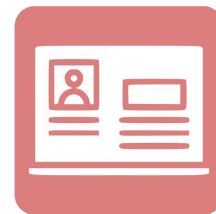
Web Application

Introduction to REST APIs



- What is REST?
- What is a resource in the world of REST and how it's identified?
- What are REST URI, how are they defined?

Introduction to Web Application



- What are the different layers of a web application : FE | BE | Data Layer.
- Get to know about BE in detail - What! Why
- How the FE can talk with the BE using REST APIs
- How creating REST APIs help us to leverage same backend for multiple clients, be it browser app or the mobile app or another backend application itself

Now. let's discuss our eCommerce Application

Topics discussed

- Example of amazon or flipkart and talk about the different use cases.
- The different categories and large number of products that these applications server.
- How a user is authenticated.
- How a user is able to add items in the cart and then places the order.



Scope of the eCommerce Application

Topics discussed

Scopes for category :

- ADMIN should be able to create the category
- ADMIN should be able to update the category
- ADMIN should be able to delete the category
- Anyone should be able to search the category



Scopes for Product :

- ADMIN should be able to create the product
- ADMIN should be able to update the product
- ADMIN should be able to delete the product
- Anyone should be able to search the product
- Anyone should be able to search the products unders a category
- Anyone should be able to filter the products based on price



Scope and access of a user :

- User should be able to register as customer/admin
- User should be able to login
- Authenticated user should be able to place the order
- Authenticated user should be able to update the order
- Authenticated user should be able to checkout the order



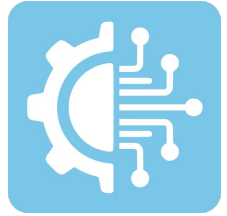
Discussion on tech-stack

- Node.js
- Express as a framework
- Sequelize as the ORM framework
- MySQL as the RDBMS database
- JEST as the test framework



Bootstrapping the eCommerce Application

Steps on how to get started with the project



- Choose a location on your machine and create a folder called eCommerce
- Initialize the node project - npm init
- Add following basic dependencies in the project :
- Express - framework for creating REST APIs
- Sequelize - ORM used for communication with RDBMS
- Mysql2 - dependency used to connect with MySQL
- Body-parser - Middleware used to parse the JSON body

Discuss about the importance of setting the environment variables and reading it through the .env files :

- Application has to run in different environments like DEV | QA | Staging | PROD. Different environments might have different credentials for running the application. For example, on our DEV box, we might have PORT 8080 available for use. But in the Staging machine we would like to run over 9999. Since this PORT value can change based on the environment, we should not hard code it inside the code
- Secondly while running these applications over cloud, almost every cloud provider has set their environment variables. Node.js application can read these env variables/parameters to customize its execution. Node.js retrieves these variable with the help of `process.env.<parameter name>` , ex : `process.env.NODE_ENVIRONMENT`

How to set the .env file

- Create a file named .env inside the root folder
- Install the dependency dotenv. This enables us to read the values set in the .env files as the process.env parameter

Project Setup :

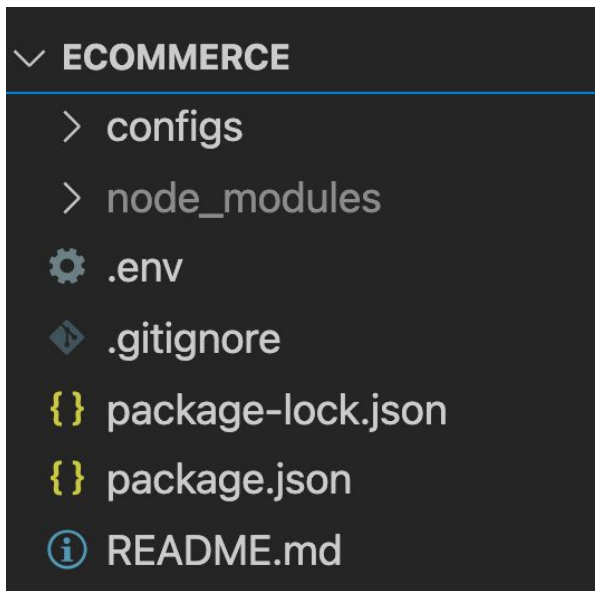
1. Create the .env file
2. Set the value inside the .env file as below :

PORT=8080

3. Create a folder named configs - for storing all the application level configs



Project structure should look like below :



4. Now create a file server.config.js inside the configs folder

- server.config.js :

<https://github.com/Vishwa07dev/eCommerce/blob/session1/configs/server.config.js>

```
if(process.env.NODE_ENV !== 'production'){
  require('dotenv').config();
}
module.exports = {
  PORT : process.env.PORT
}
```

5. Create the starting file server.js, to create the server and check if the application is able to read the port from the .env file

- server.js : <https://github.com/Vishwa07dev/eCommerce/blob/session1/server.js>

```
const express = require('express');
const serverConfig = require('./configs/server.config');

const app = express();

app.listen(serverConfig.PORT, () => {
  console.log(`Application started on the port no : ${serverConfig.PORT}`);
})
```

6. Start the application to see if all works well

7. Setup the configurations for connecting to the database

- Create a file db.config.js inside the configs folder

db.config.js : <https://github.com/Vishwa07dev/eCommerce/blob/session1/configs/db.config.js>

```
module.exports = {
  HOST : "localhost",
  USER : "root",
  PASSWORD : "Welcome1",
  DB : "ecom_db",
  dialect : "mysql",
  pool :{
    max :5,
    min :0,
    acquire: 30000, //max time in ms that a pool will try to get
connection before throwing error
    idle :10000 // maximum time in ms that a connection can be idle
before being released
  }
}
```

Yayy, with this we are done with the setup of the project for the session 1.

MCQs

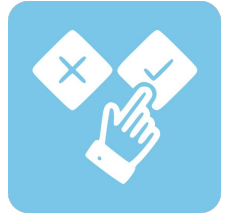
1. Node Js is _____?

- a) Asynchronous
- b) Synchronous



2. Which function is used to include modules in Node Js.

- a) `include();`
- b) `require();`
- c) `attach();`



3. To install Node.js express module

- a) \$ npm install express
- b) \$ node install express
- c) \$ install express
- d) None of the above



4. Node.js runs on _____

- a) Client
- b) Server
- c) Both, server and client
- d) None of the above



5. Which of the following types of application can be built using Node.js?

- a) Web Application
- b) Chat Application
- c) RESTful Service
- d) All of the above



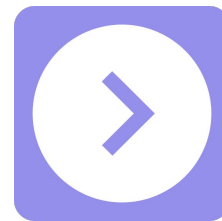
Practice Problems



1. Try creating a new application by initializing a npm/node project directory and install express in it.
2. Now, write using express, make the application listen on port 8080 and send a message as “This API is working” from it.

Next session

- We will start with Category resource
- We will create the Model for Category resource
- And then define and implement the RESTful endpoints for the Category resource



THANK YOU