

Problem Statement

It is known fact that in today's work-from-home world, people prefer ordering food that can be delivered at the comfort of their home. So most of the times people end up ordering food from restaurants that have delivery services. The objective of this problem statement is to come up with a solution for people to order food online and get prompt delivery.

You need to solve the given problem by developing a web application, that should facilitate users to order food online from different restaurants using the web-app to cater their needs.

Scope of work

You have been asked to build the backend system for the following:

- Registration
- Login and
- Display the list of Food items

Backend :

Below are the end points to be built

Method	End point	Description + Payload
POST	/api/register	<p>To register the user with basic details like</p> <p>Input:</p> <pre>{ id: Number, username: String, email: String, password: String address: { houseno: Number, street: String, city: String, state: String, zip: number } }</pre> <p>Return code : 201</p> <p>Return:</p> <pre>{ id: Number, username: String, email: String, password: String address: { houseno: Number, street: String,</pre>

		<pre> city: String, state: String, zip: number } } </pre>
POST	/api/authenticate	<p>To validate the user is registered in the system</p> <p>Input:</p> <pre> { username: String, password: String } </pre> <p>Return:</p> <pre> { message: User logged in successful } </pre> <p>Return Code: 200 When user details are wrong Return Code : 403 (Forbidden)</p>
GET	/api/users	<p>To get all the users who are registered to the system, the end point should return an array,</p> <pre> [{ id: Number, username: String, email: String, password: String address: { houseno: Number, street: String, city: String, state: String, zip: number } }, { id: Number, username: String, email: String, password: String address: { houseno: Number, street: String, city: String, state: String, zip: number } }] </pre> <p>Return code: 200</p>

GET	/api/users/:userID	<p>To return user by specifying id</p> <p>When valid</p> <pre>{ id: Number, username: String, email: String, password: String address: { houseno: Number, street: String, city: String, state: String, zip: number } }</pre> <p>Return Code: 200</p> <p>When Id not found:</p> <pre>{ Message: "Sorry user With <ID> not found" }</pre>
PUT	/api/users	<p>Should update the user specified in the payload which shall match the ID and updated the existing user with the new details.</p> <p>When valid</p> <pre>{ id: Number, username: String, email: String, password: String address: { houseno: Number, street: String, city: String, state: String, zip: number } }</pre> <p>Return Code: 200</p> <p>When Id not found:</p> <pre>{ Message: "Sorry user With <ID> not found" }</pre>
DELETE	/api/users/:userID	<p>Should delete the user which is specified in the :userID</p> <p>When Valid :</p> <pre>{ Message: "User Deleted Successfully" }</pre>

		Return Code: 200 When Id not found: <pre>{ Message:"Sorry user With <ID> not found" }</pre>
POST	/api/food	To add a new food to the system. <pre>{ foodId:number, foodName:String, foodCost:Number, foodType: Indian/Chinese/Mexican }</pre> Return Code : 201, and the id should be generated by the system Return Payload <pre>{ Id:Number foodId:number, foodName:String, foodCost:Number, foodType: Indian/Chinese/Mexican }</pre>
GET	/api/food/:foodID	To return a food item specified with :foodID When Valid <pre>{ Id:Number foodId:number, foodName:String, foodCost:Number, foodType: Indian/Chinese/Mexican }</pre> When :foodID not found <pre>{ Message: "Sorry Food Not Found" }</pre>