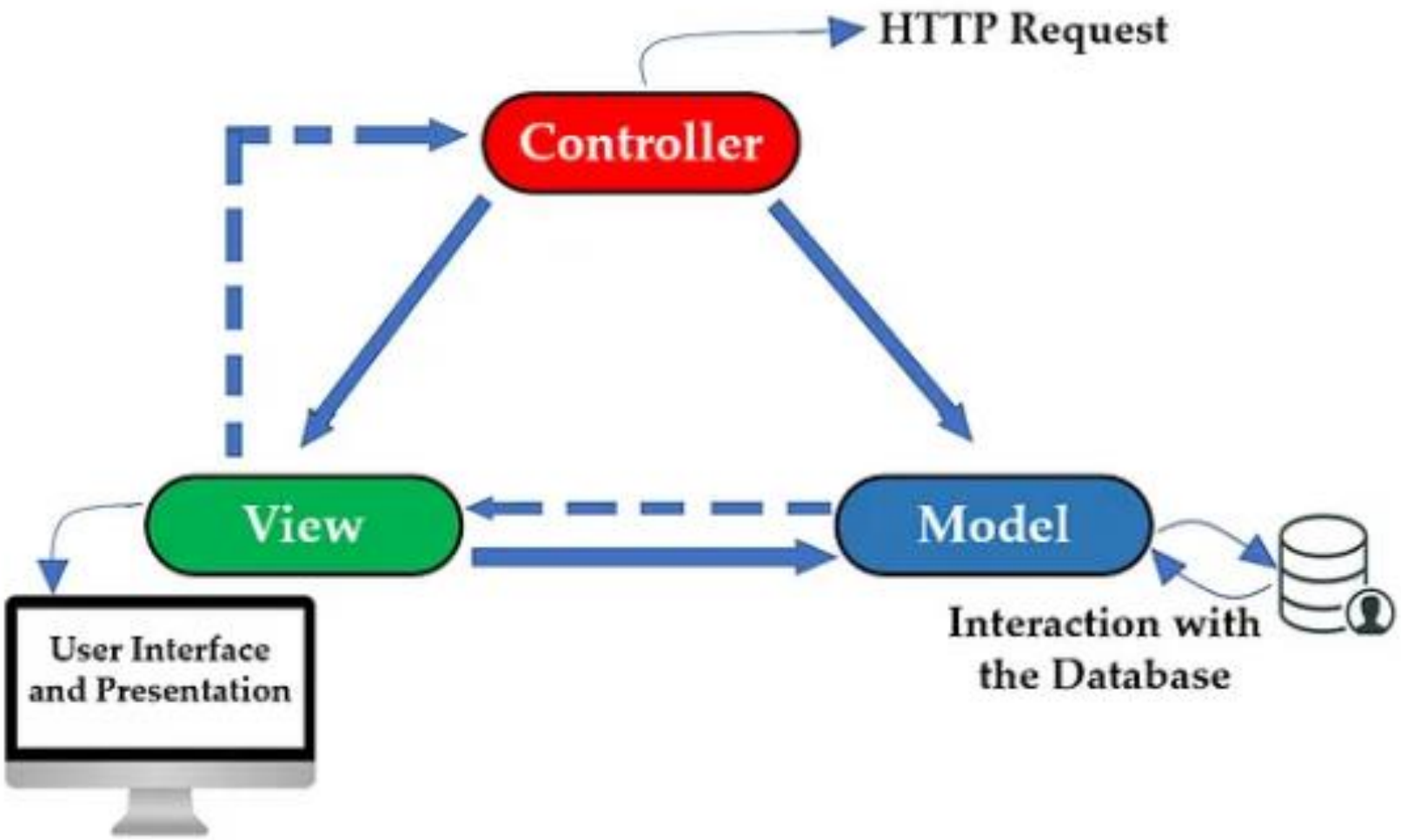
C R U D

Operations

MVC Architecture



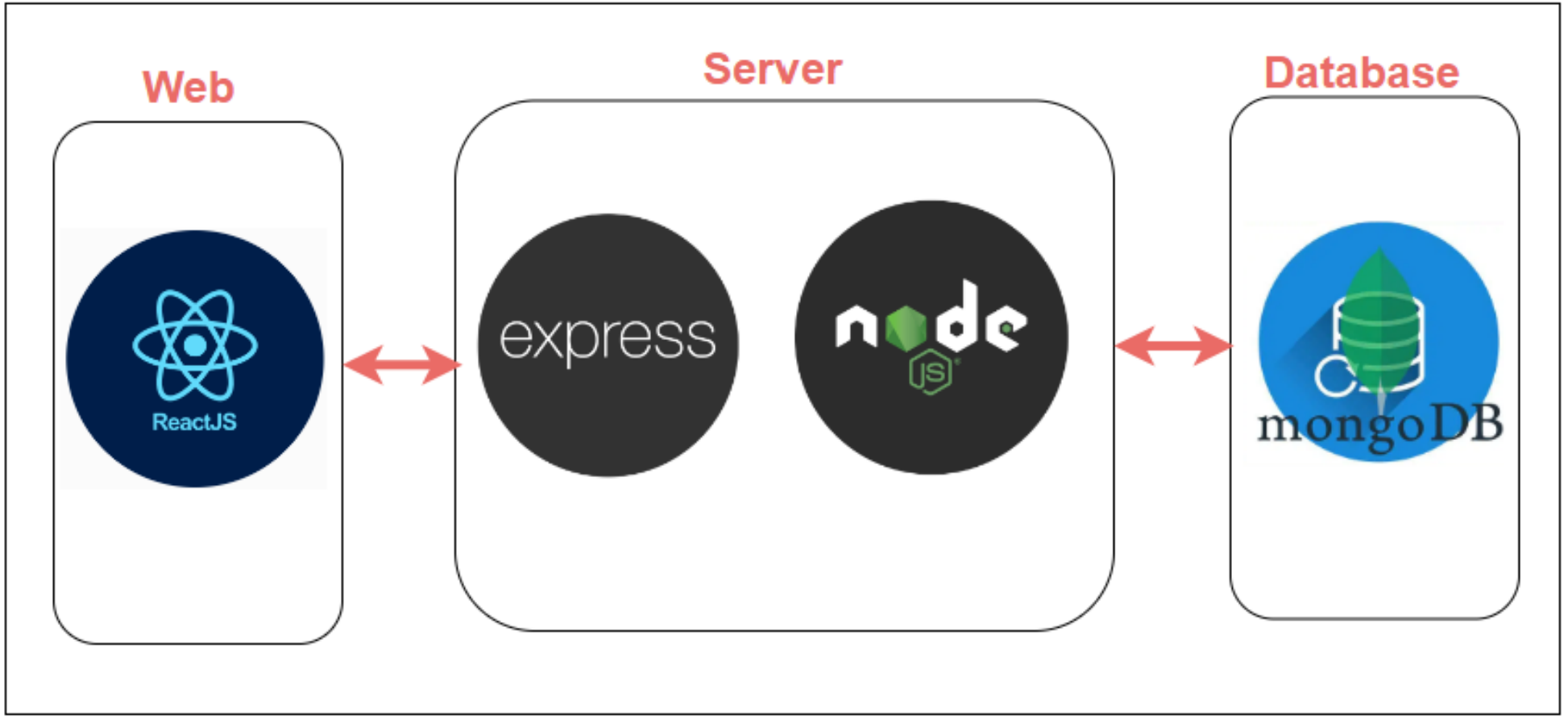
M MongoDB - Document Database

E ExpressJS – Server Side Web FW, Runs in NodeJS, has URL Routing & Handles HttpRequests and Responses

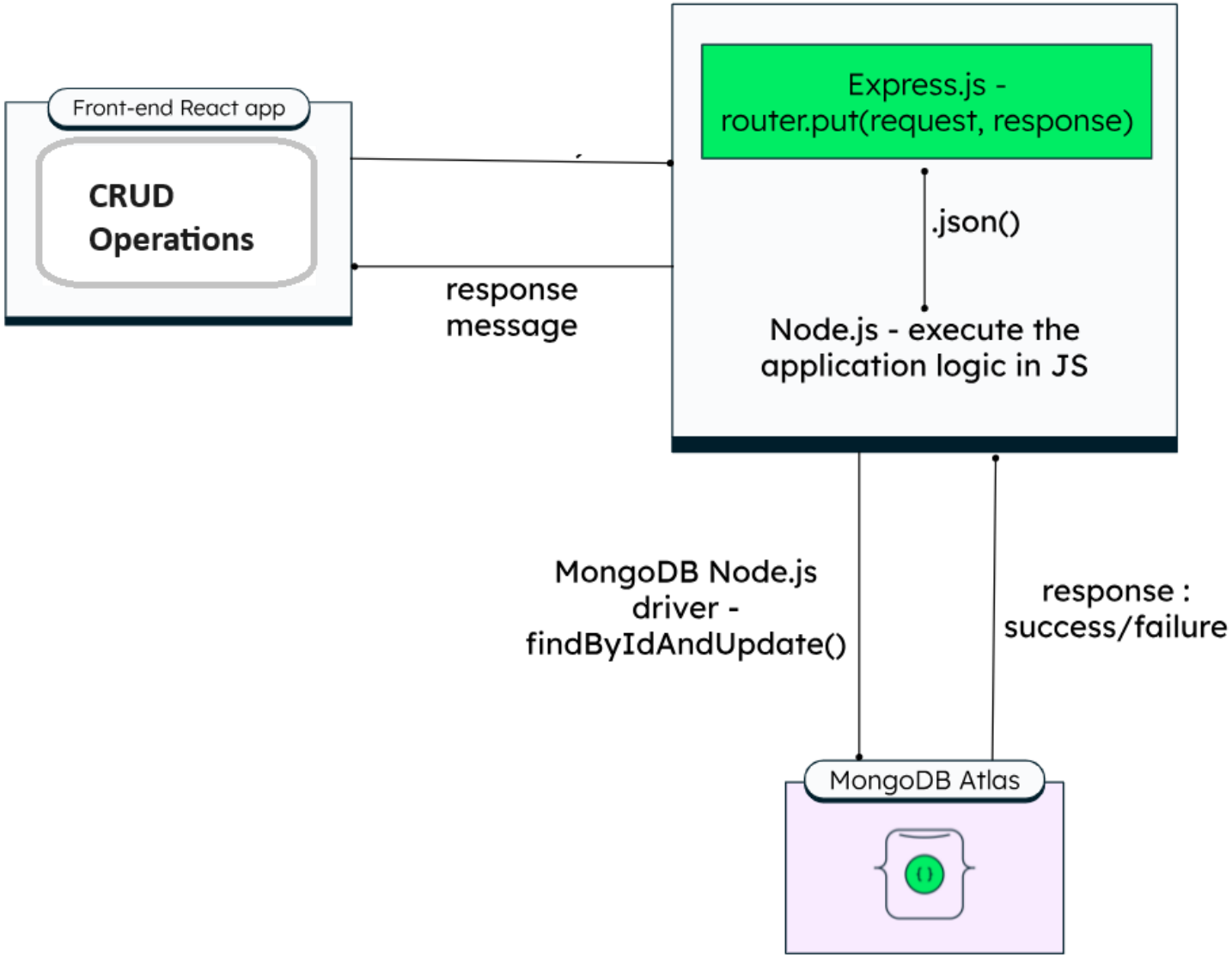
R ReactJS – Client side JavaScript Framework

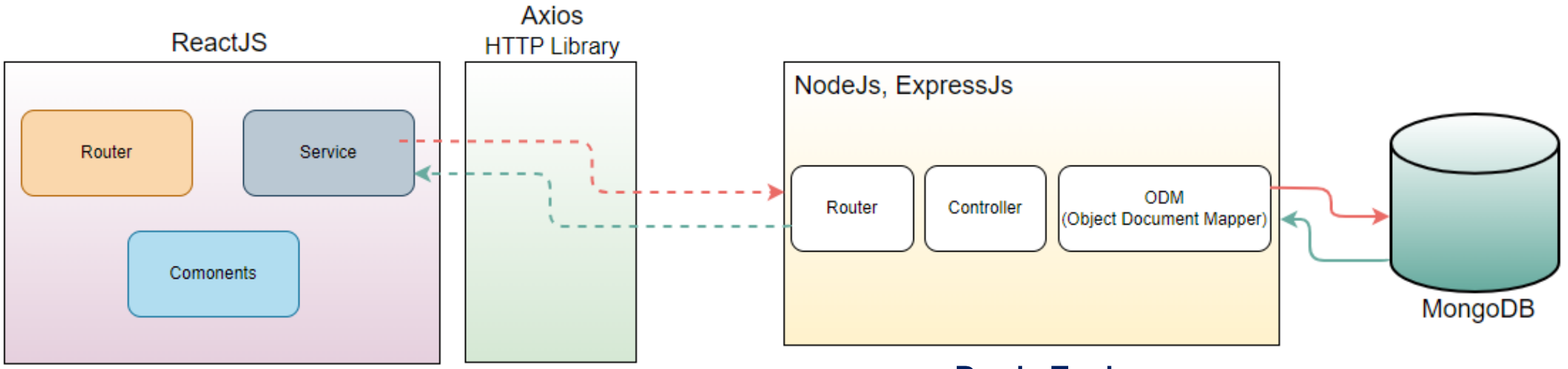
N NodeJS – JavaScript Web Server

How MERN Stack Works



How MERN Stack Works…





Front End

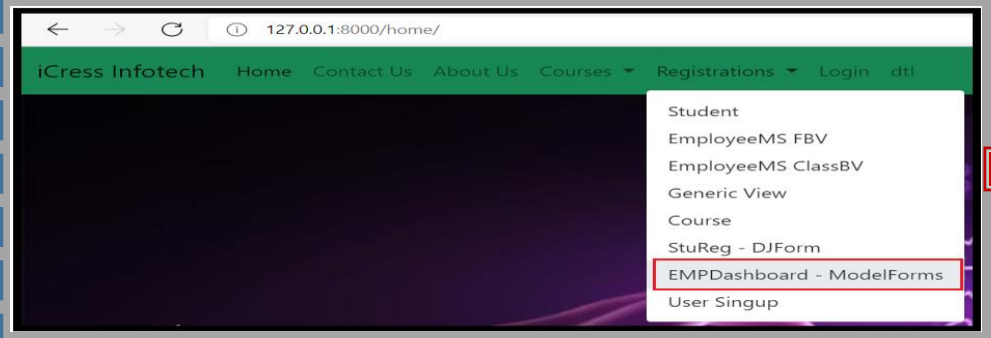
Back End
RESTFul Services

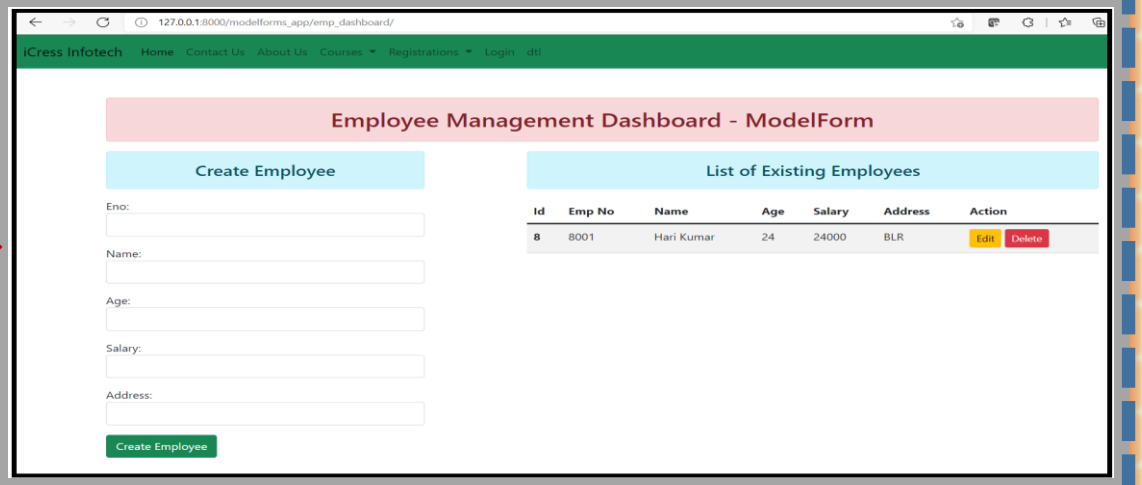
WEBSERVICES

RESTFUL SERVICES

- API
- Web API
- REST
- REST API/RESTFUL API/RESTFul Services

USER





User Interacting (OR) Accessing Application through browser

GUI

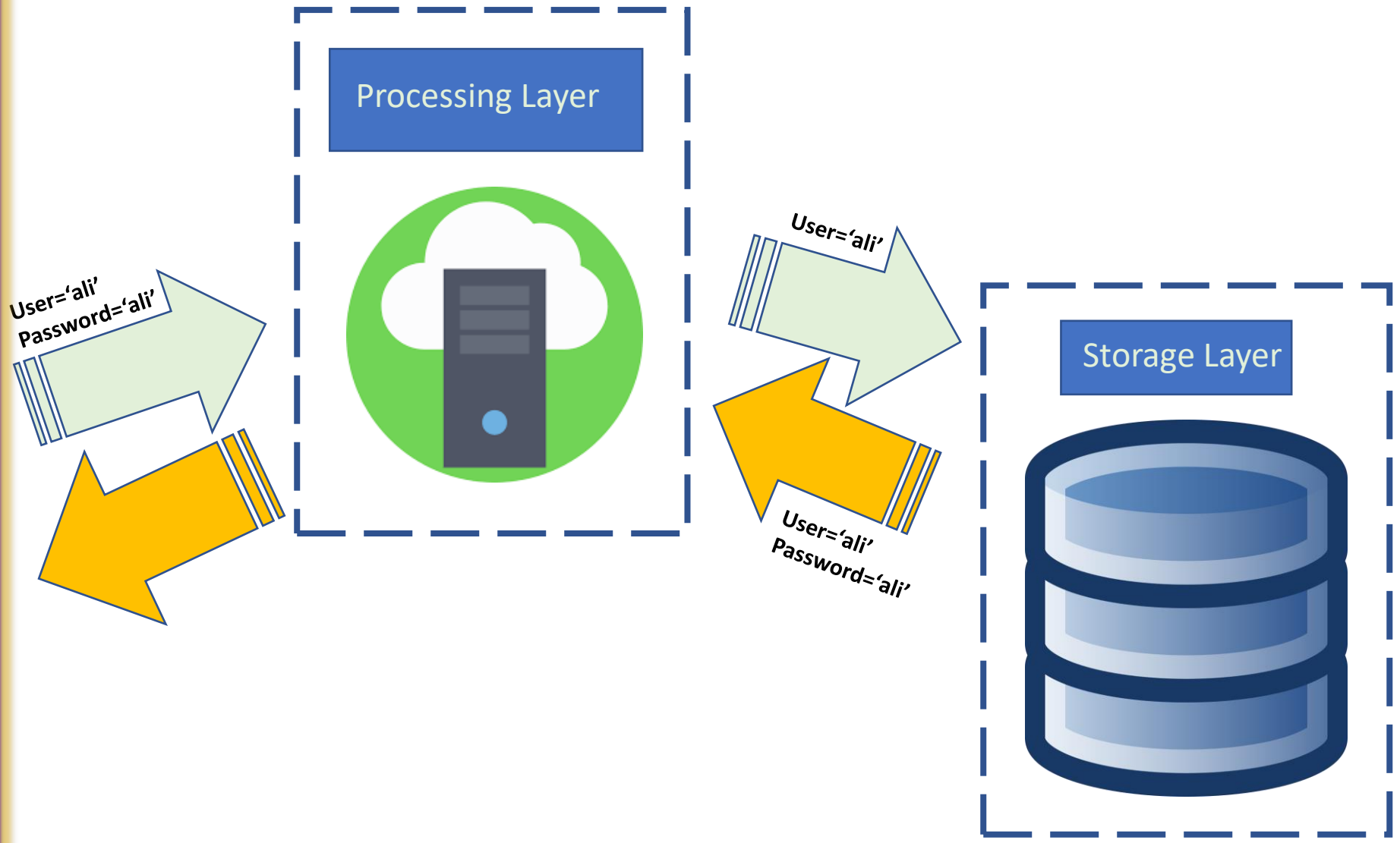
API

API

iCress App

Storage

API


Processing Layer
User='ali'
Password='ali'
User='ali'
User='ali'
Password='ali'
Storage Layer

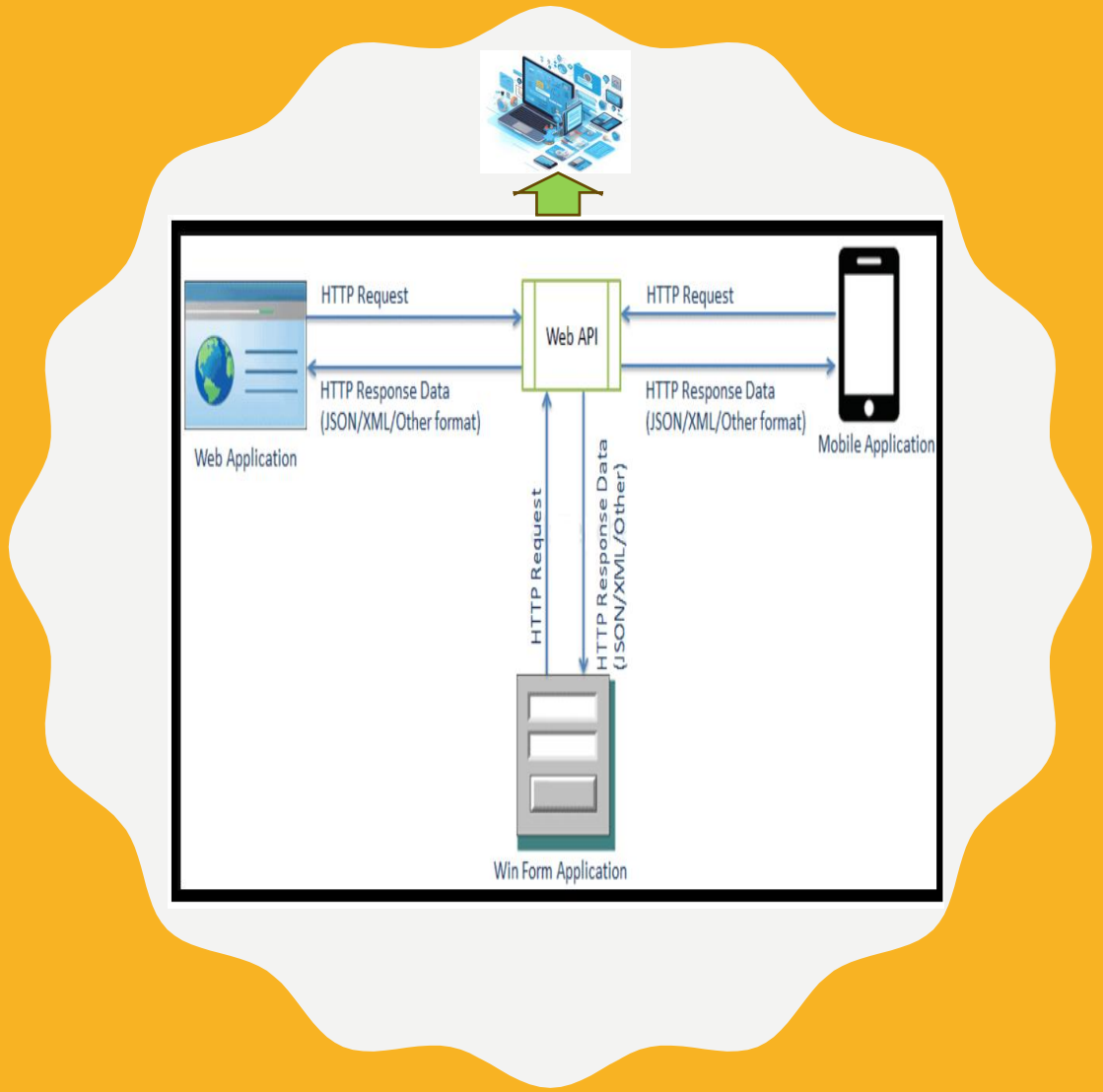
API - APPLICATION PROGRAM INTERFACE

- API is a s/w that allows two applications talk to each other.
- API, is a way to send and receive data between computers/two applications.
- For example, if you want to display Google Maps on your site, but the maps application is running on Google's servers so, you need a way to ask Google to provide you with the maps. The way to ask Google to send you the requested maps is through an API provided by Google that tells you to which web addresses should you send the requests to get the data.

WEB API

- API for web is called Web API
- Service available over the WEB.
- Web API contains one or more end points to define Request & Response

- Client sends a HTTP Request to API
- API Will communicate to Web Application/Database
- Web Application provides Response/Data to API
- API Returns Response/Data back to Client
- Data which we are getting may be XML/JSON Data



TYPES OF API

- SOAP API(Simple Object Access Protocol)
- RESTFul API

REST

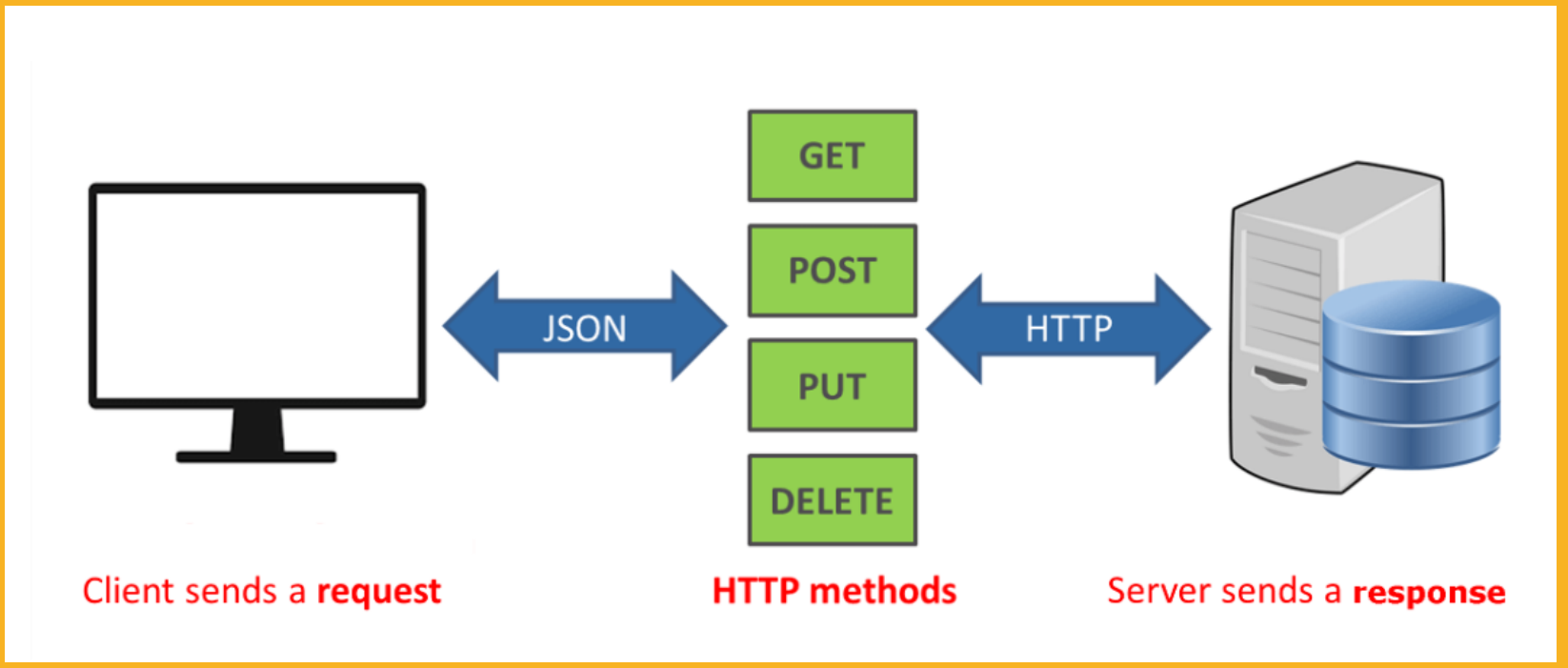
- REST is an architectural guideline for creating Web API that uses HTTP Protocol for communication.

REST API

- The API which is developed using REST is know as REST API/ RESTFul API/ RESTFul Web Service/ RESTFul Services

REST ARCHITECTURAL GUIDELINES (ROY FIELDING)

1. End Point - *Resource on web must be accessible using URL Patterns*
2. HTTP Verbs/HTTP Methods - *HTTP Verbs/HTTP Methods should be used to perform actions on recourse.*
3. Status Codes - *Status codes should be included in response header after each call to the server.*
4. Data/Body - *JSON/XML should be used for exchanging the data between client & server (Request, Response)*
5. Stateless - *Communication between client & server should be STATELESS. Request can not be depends on any other request & All requests made from client to server are fresh requests. No data from the client is stored on the server side.*


GET
POST
PUT
DELETE
JSON
HTTP
Client sends a **request**
HTTP methods
Server sends a **response**

REST API

END POINT

- To view LIST of Employees, a GET Request should be sent to server

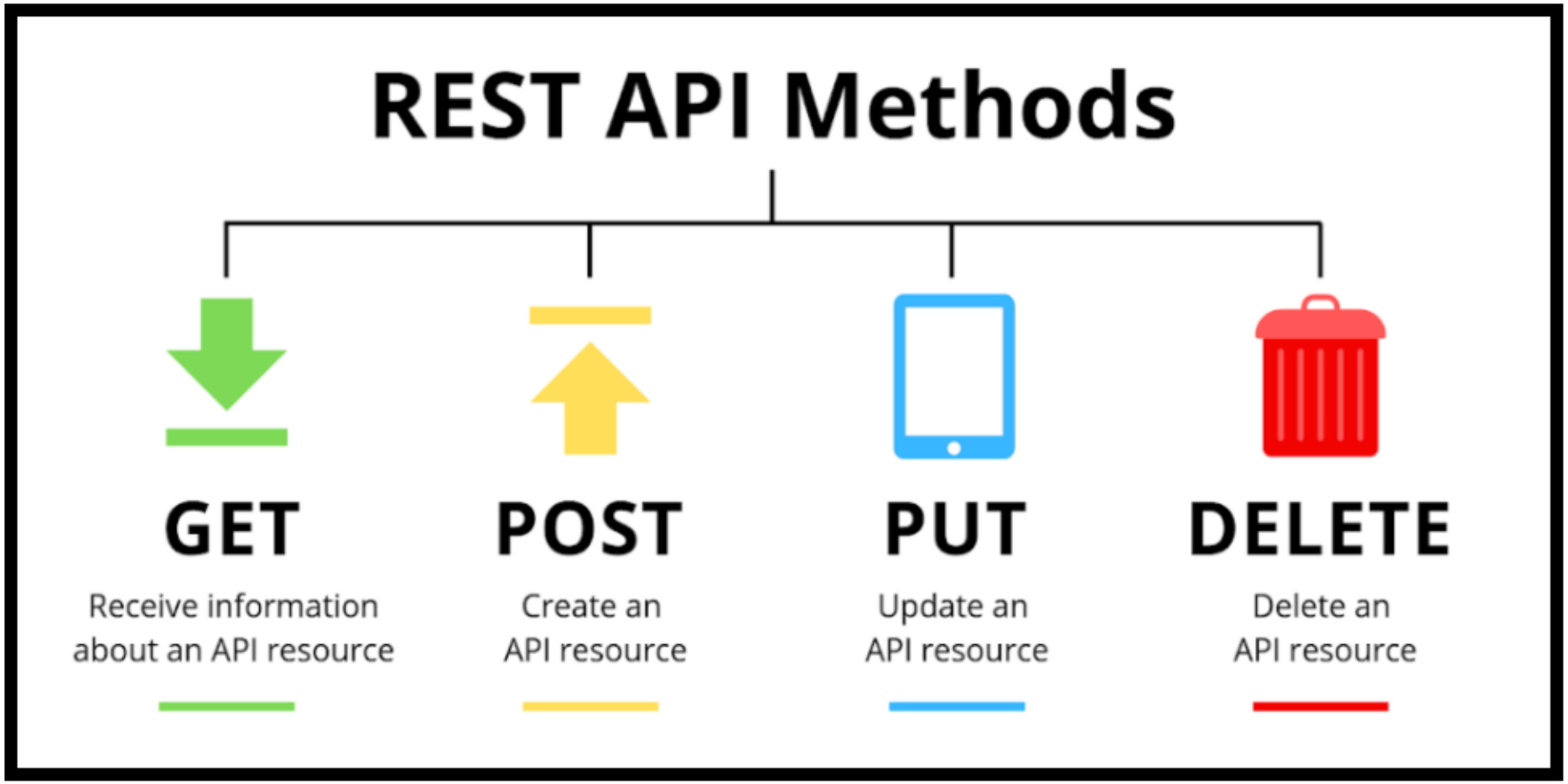
  URI: http://www.icress.in/api/employee
- To GET an employee details with the id of 1001, we send a GET Request.

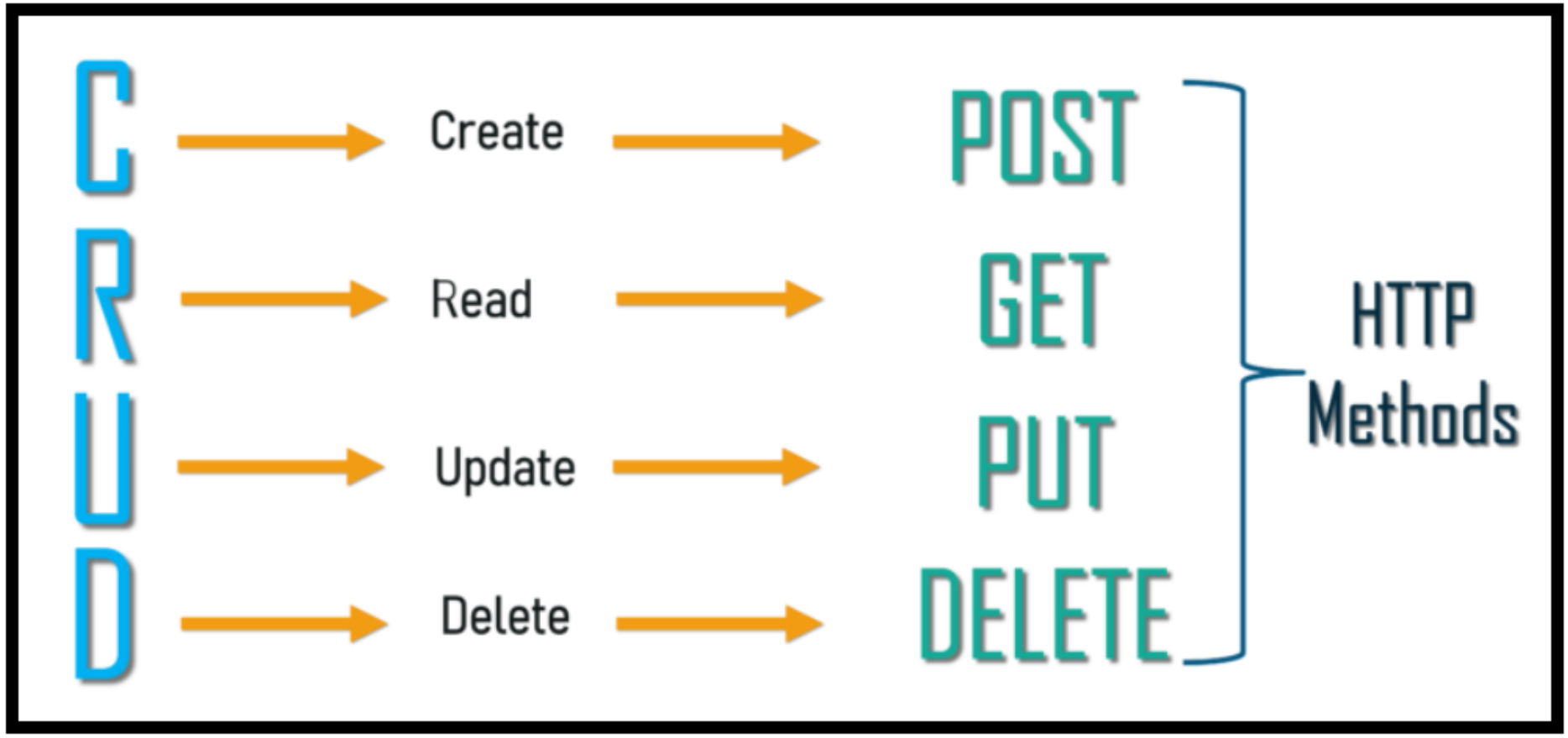
  URI: http://www.icress.in/api/employee/1001
- To CREATE a new Employee, we send the data with the POST Request.

  URI: http://www.icress.in/api/employee
- To UPDATE an Employee with the id of 1001 send PUT/PATCH Request.

  URI: http://www.icress.in/api/employee/1001
- To DELETE an Employee with id 1001 send DELETE Request

  URI: http://www.icress.in/api/employee/1001


REST API Methods
GET
Receive information about an API resource
POST
Create an API resource
PUT
Update an API resource
DELETE
Delete an API resource


C
Create
POST
R
Read
GET
U
Update
PUT
D
Delete
DELETE
HTTP
Methods

HTTP Status Codes

1XX
INFORMATIONAL

2XX
SUCCESS

3XX
REDIRECTION

4XX
CLIENT ERROR

5XX
SERVER ERROR

200 - OK. The request was successful.

204 - No Content. The server successfully processed the request and did not return any content.

301 - Moved Permanently. The server responds that the requested page (endpoint) has been moved to another address and redirects to this address.

302 - Moved temporarily. The resource moved temporarily to another URL.

400 - Bad Request. The server cannot process the request because the client-side errors (incorrect request format).

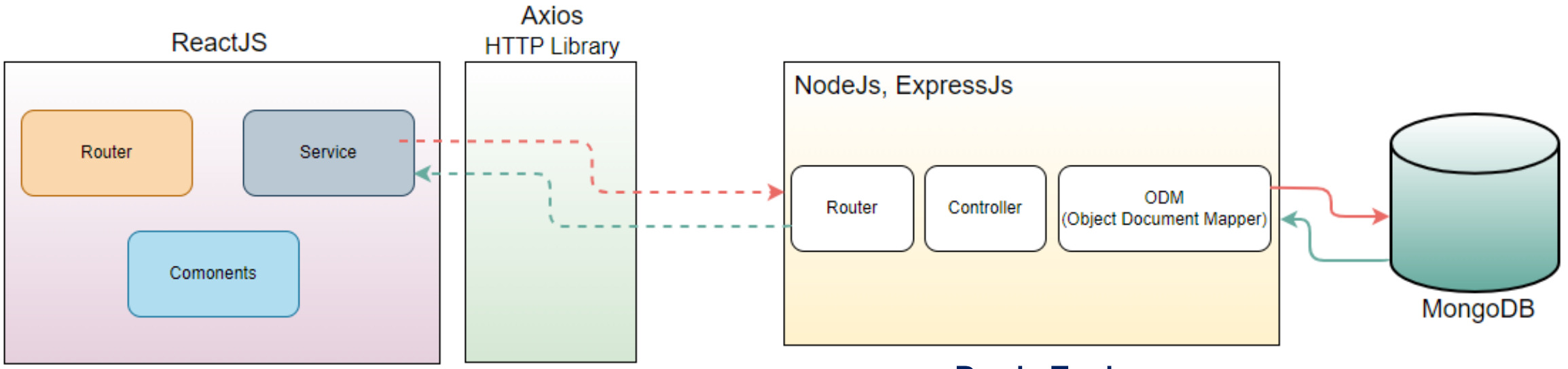
401 - Unauthorized. Occurs when authentication was failed, due to incorrect credentials or even their absence.

403 - Forbidden. Access to the specified resource is denied.

404 - Not Found. The requested resource was not found on the server.

405 - Method not allowed. For example, when sending a DELETE request to a server that doesn't support the method.

500 - Internal Server Error. Occurs when an unknown error has occurred on the server.



Front End

Back End
RESTFul Services