

Trabajo Integrador CAC JS NODE 2024

Grupo 2 - Comisión 24122

- Ana Sosa
- Mariela Montenegro
- Maria del Carmen Rohr





Al servicio de tus Mascotas



Ingreso

Inicio

Servicios

Especialidades

Plan de Salud

Contacto

Profesionales

Atención Veterinaria en la Ciudad de Buenos Aires

Quiénes somos

Somos un consultorio veterinario de gran reconocimiento en la zona, con más de **20 años de antigüedad**.

Contamos con personal y equipo calificado para ofrecer a los clientes y a sus mascotas un servicio de primera calidad.

No dude en visitarnos, estamos en **Almagro, Ciudad de Buenos Aires**.

Veterinaria

En Consultorio Veterinario Patitas encontrará **Médicos Veterinarios de gran profesionalismo** que se encargarán de brindar la mejor atención a su animal de compañía.

Consúltenos y reciba un excelente asesoramiento veterinario. Lo esperamos.

Pet shop

Ofrecemos medicamentos veterinarios, alimentos balanceados, juguetes, entre otros productos para su mascota.

Representamos su mejor opción en el rubro gracias a nuestro **amplio catálogo de servicios y productos**.

Servicios

Nos destacamos por ser un referente importante en el rubro ya que somos especialistas en

Atención Clínica y Cirugía.

Además, brindamos servicio de **Análisis Clínicos de Laboratorio, Ecografías, Cardiología y Rayos X.**



Repositorio GitHub: <https://github.com/mmontenegr/Integrador.git>

The screenshot shows the GitHub repository page for 'Integrador' by user 'mmontenegr'. The repository is public and has 37 commits, 2 branches, and 0 tags. The main branch is 'master'. The repository contains a file tree with the following files and their last commit times:

File	Commit	Time
public	Prueba FavIcon	2 days ago
src	PutPersonasOK	2 days ago
.gitignore	Integracion	2 weeks ago
LICENSE	Version 0.0.1	3 weeks ago
README.md	PutPersona	2 days ago
index.js	JWT	last week
package-lock.json	PutPersonasOK	2 days ago
package.json	JWT	last week

The README file is selected, showing the title 'tp-node-comision24122' and the subtitle 'Codo a Codo - Comisión 24122 - Grupo 2'. The main heading is 'Veterinaria Patitas App'. The text describes the project and provides instructions for installation:

```
npm i express
```

```
npm i mysql2
```

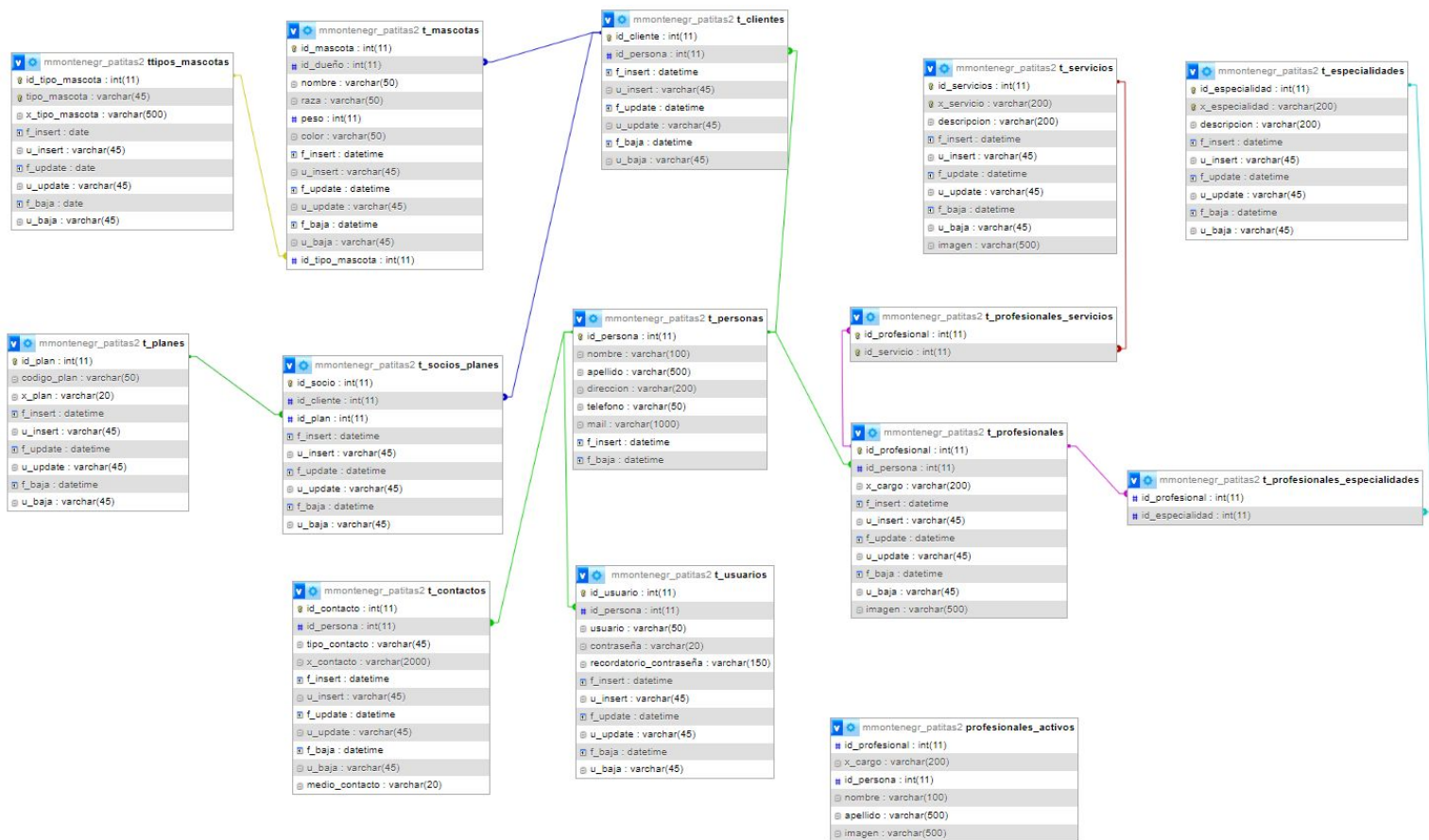
```
npm install dotenv
```

```
npm install jsonwebtoken bcryptjs
```

The instructions conclude with 'y poner a andar el proyecto'.

The right sidebar contains information about the repository, including the 'About' section (Codo a codo - Comision 24122 - Full Stack Javascript Node - Trabajo Integrador), 'Releases' (No releases published), 'Packages' (No packages published), 'Deployments' (33 deployments, including Production and Preview), and 'Languages' (HTML 35.1%, JavaScript 33.7%, CSS 31.2%). The 'Suggested workflows' section shows a workflow for 'Jekyll using Docker image'.

Modelo de Datos:



CRUD: GET - <https://integrador-node-js.onrender.com/especialidades>

My Workspace | New | Import

Collections

- Alwaysdata
- Apex
- https://integrador-node-js.onrender.c...
- GET RecuperarEspecialidades**
- GET RecuperarEspecialidad
- POST NuevaEspecialidad
- PUT Actualiza Especialidad
- DEL Elimina Especialidad
- GET RecuperarServicios
- GET RecuperarProfesionales
- GET RecuperarPersonas
- PUT PUT Personas
- GET RecuperarUsuarios
- GET RecuperarContactos
- POST Post Register
- POST Post Login

Environments

- Patitas - LocalHost:4000
- Replit
- GET Get-Servicios

Params | Auth | Headers (7) | Body | Scripts | Tests | Settings

Query Params

Key	Value	Description
Key	Value	Description

Body | Cookies | Headers (14) | Test Results

200 OK 962 ms 737 B | Save as example

Send

```
[
  {
    "id_especialidad": 1,
    "x_especialidad": "Gastroenterología",
    "descripcion": null,
    "f_insert": "2024-06-01T20:48:38.000Z",
    "u_insert": "user()",
    "f_update": null,
    "u_update": null,
    "f_baja": null,
    "u_baja": null
  },
  {
    "id_especialidad": 2,
    "x_especialidad": "Nutrición",
    "descripcion": null,
    "f_insert": "2024-06-01T20:48:38.000Z",
    "u_insert": "user()",
    "f_update": null,
    "u_update": null,
    "f_baja": null,
    "u_baja": null
  },
  {
    "id_especialidad": 3,
    "x_especialidad": "Clínica médica",
    "descripcion": null,
    "f_insert": "2024-06-01T20:48:38.000Z",
    "u_insert": "user()",
    "f_update": null,
    "u_update": null,
    "f_baja": null,
    "u_baja": null
  },
  {
    "id_especialidad": 4,
```

CRUD: POST - <https://integrador-node-js.onrender.com/especialidades>

```
{  
  "x_especialidad": "Cirugia"  
}
```

The screenshot displays a REST client interface with a sidebar on the left showing a collection of API endpoints. The main panel is titled 'NuevaEspecialidad' and shows a POST request to 'https://integrador-node-js.onrender.com/especialidades'. The request body is a JSON object: `{ "x_especialidad": "Cirugia" }`. The response is a 200 OK status with a 418 ms response time and 505 B of data. The response body is a JSON object: `{ "message": "Especialidad creada con id 21", "id_especialidad": 21 }`.

My Workspace | New | Import

Collections

- Alwaysdata
- Apex
- https://integrador-node-js.onrender.c...
- GET RecuperarEspecialidades
- GET RecuperarEspecialidad
- POST NuevaEspecialidad**
- PUT Actualiza Especialidad
- DEL Elimina Especialidad
- GET RecuperarServicios
- GET RecuperarProfesionales
- GET RecuperarPersonas
- PUT PUT Personas
- GET RecuperarUsuarios
- GET RecuperarContactos
- POST Post Register
- POST Post Login
- Patitas - LocalHost:4000
- Replit
- GET Get-Servicios

https://integrador-node-js.onrender.com / NuevaEspecialidad

POST | https://integrador-node-js.onrender.com/especialidades

Params | Auth | Headers (9) | **Body** | Scripts | Tests | Settings

raw | JSON | Beautify

```
1 {  
2   "x_especialidad": "Cirugia"  
3 }
```

Body | Cookies | Headers (14) | Test Results

200 OK | 418 ms | 505 B | Save as example

Pretty | Raw | Preview | Visualize | JSON | Beautify

```
1 {  
2   "message": "Especialidad creada con id 21",  
3   "id_especialidad": 21  
4 }
```

CRUD: PUT - <https://integrador-node-js.onrender.com/especialidades/11>

```
{
  "x_especialidad": "Cirugía General",
  "descripción": "Cambio desde la web"
}
```

The screenshot displays a REST client interface with a sidebar on the left containing 'My Workspace', 'Collections', 'APIs', 'Environments', 'Monitors', 'Flows', 'History', and 'Replit'. The main area shows a PUT request to `https://integrador-node-js.onrender.com/especialidades/11`. The request body is in JSON format:

```
{
  "x_especialidad": "Cirugía General",
  "descripción": "Cambio desde la web"
}
```

The response status is 200 OK, with a response time of 395 ms and a body size of 488 B. The response body is in JSON format:

```
{
  "message": "Especialidad actualizada"
}
```

CRUD: DELETE -

<https://integrador-node-js.onrender.com/especialidades/11>

The screenshot displays a REST client interface with a sidebar on the left containing a file explorer and a list of API endpoints. The main panel shows a DELETE request to the URL `https://integrador-node-js.onrender.com/especialidades/11`. The response is a JSON object with a message: `"message": "Especialidad eliminada"`.

Left Sidebar:

- My Workspace
 - Collections
 - Alwaysdata
 - Apex
 - https://integrador-node-js.onrender.c...
 - GET RecuperarEspecialidades
 - GET RecuperarEspecialidad
 - POST NuevaEspecialidad
 - PUT Actualiza Especialidad
 - DEL Elimina Especialidad
 - GET RecuperarServicios
 - GET RecuperarProfesionales
 - GET RecuperarPersonas
 - PUT PUT Personas
 - GET RecuperarUsuarios
 - GET RecuperarContactos
 - POST Post Register
 - POST Post Login
 - Patitas - LocalHost:4000
 - Replit
 - GET Get-Servicios

Main Panel:

URL: `https://integrador-node-js.onrender.com / Elimina Especialidad`

Method: **DELETE** | URL: `https://integrador-node-js.onrender.com/especialidades/11` | **Send**

Params: none | Headers: 7 | Body: Scripts | Tests | Settings

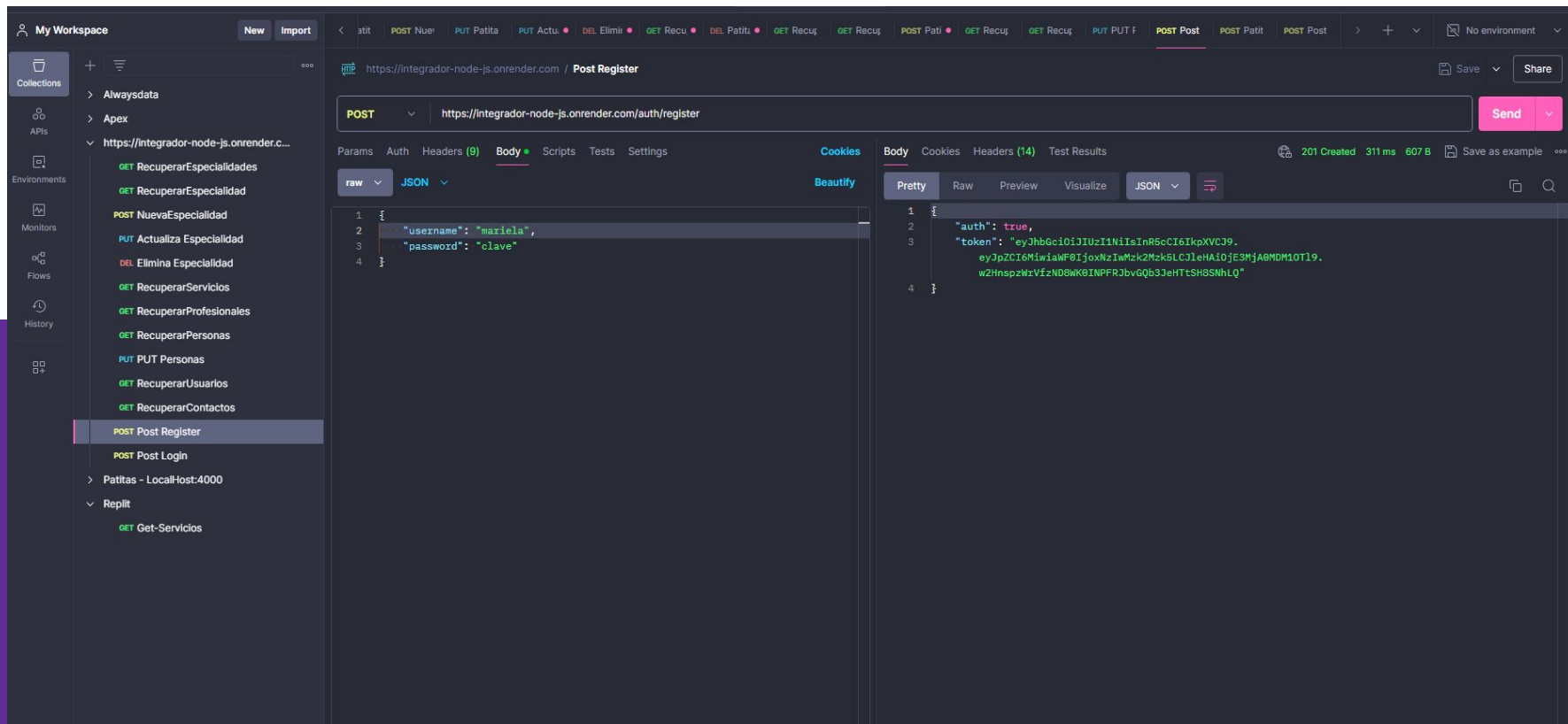
Body: `This request does not have a body`

Response: `200 OK 400 ms 486 B` | Save as example

Body (JSON):

```
1 {
2   "message": "Especialidad eliminada"
3 }
```


AUTENTICACIÓN: REGISTER - <https://integrador-node-js.onrender.com/auth/register>



AUTENTICACIÓN: LOGIN - <https://integrador-node-js.onrender.com/auth/login>

The screenshot displays a REST client interface with a sidebar on the left showing a collection of API endpoints. The main area is divided into two panels: the left panel shows the request details, and the right panel shows the response details.

Request Details:

- Method:** POST
- URL:** <https://integrador-node-js.onrender.com/auth/login>
- Body (JSON):**

```
1 {
2   "username": "mariela",
3   "password": "clave"
4 }
```

Response Details:

- Status:** 200 OK
- Time:** 364 ms
- Size:** 603 B
- Body (JSON):**

```
1 {
2   "auth": true,
3   "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6Im1iwiaWF0IjoxNzIwMzZlNDE5LCJleHAiOjE3MjA8MDM2MT19.Sns_MpHnPd0_zVPrsU0pj3hBiaqsZaukIR8QKC4NwJA"
4 }
```

Integración Frontend - Backend - Servicios, Especialidades, Profesionales y Contactos



Profesionales

[Ingreso](#)[Inicio](#)[Servicios](#)[Especialidades](#)[Plan de Salud](#)[Contacto](#)[Profesionales](#)

Nuestros especialistas



Juan López

Veterinario coordinador de clínica
médica y responsable de clínica felina.



Atendemos emergencias veterinarias las 24hs.



Rúbrica:

- La base de datos debe desarrollarse con MySql, tener al menos 4 tablas, con al menos 4 diferentes tipos de datos (10) **Listo**
 - Entre las tablas al menos deberá haber una relación "uno a muchos" (10) **Listo**
 - A través del servidor levantado con Node y Express se debe poder realizar al menos un tipo de alta (POST) (15) **Listo**
 - De la misma forma se debe poder realizar modificaciones de los registros (PUT) (15) **Listo**
 - Se debe poder acceder a los registros de la tabla (GET) (15) **Listo**
 - Se debe poder realizar borrado físico de los datos (DELETE) (15) **Listo**
-
- El trabajo práctico deberá subirse a un servidor online y compartirse mediante un repositorio de git (Mandatorio) (5) **Listo**
 - La página deberá subirse a un servidor on-line para poder ser navegada por el docente (Mandat.) (5) **Listo**
 - Se valorará la aplicación de un token o método de autenticación (5) **Listo**
 - El backend debe estar integrado con un frontend (5) **Listo**