

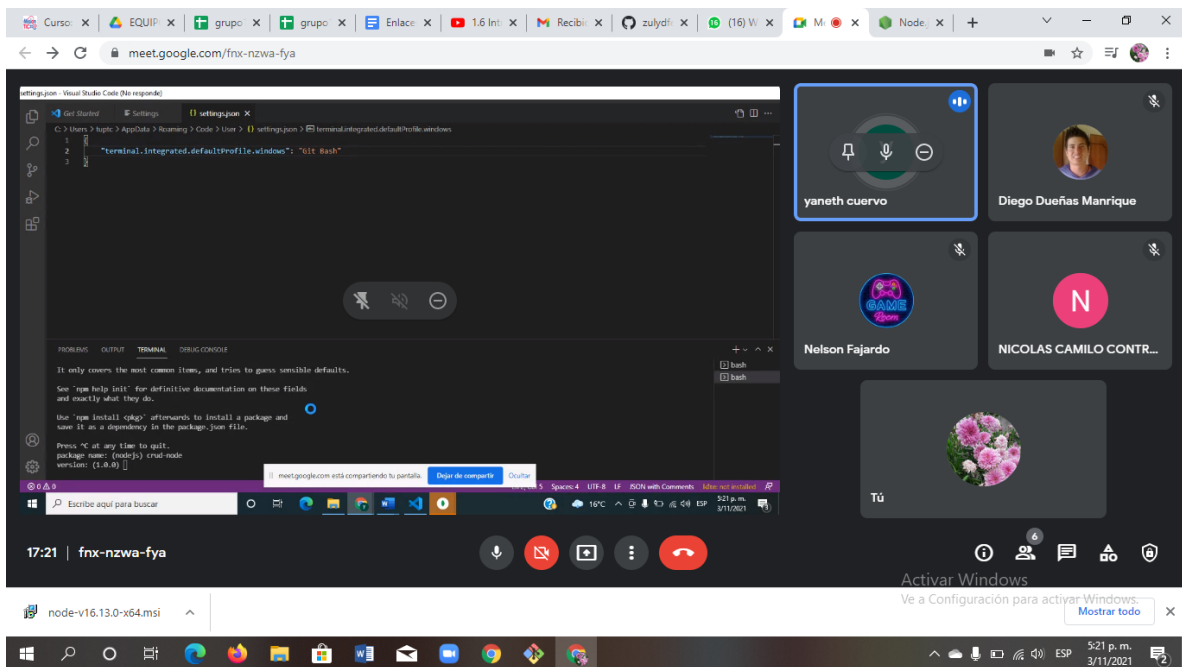
SPRINT 1. PROYECTO TIENDA VIRTUAL MASCOTA MODELO ENTIDAD RELACION - LOOPBACK

Integrantes Grupo 12. Equipo 3 LOS TIC DE LOS NODEJS

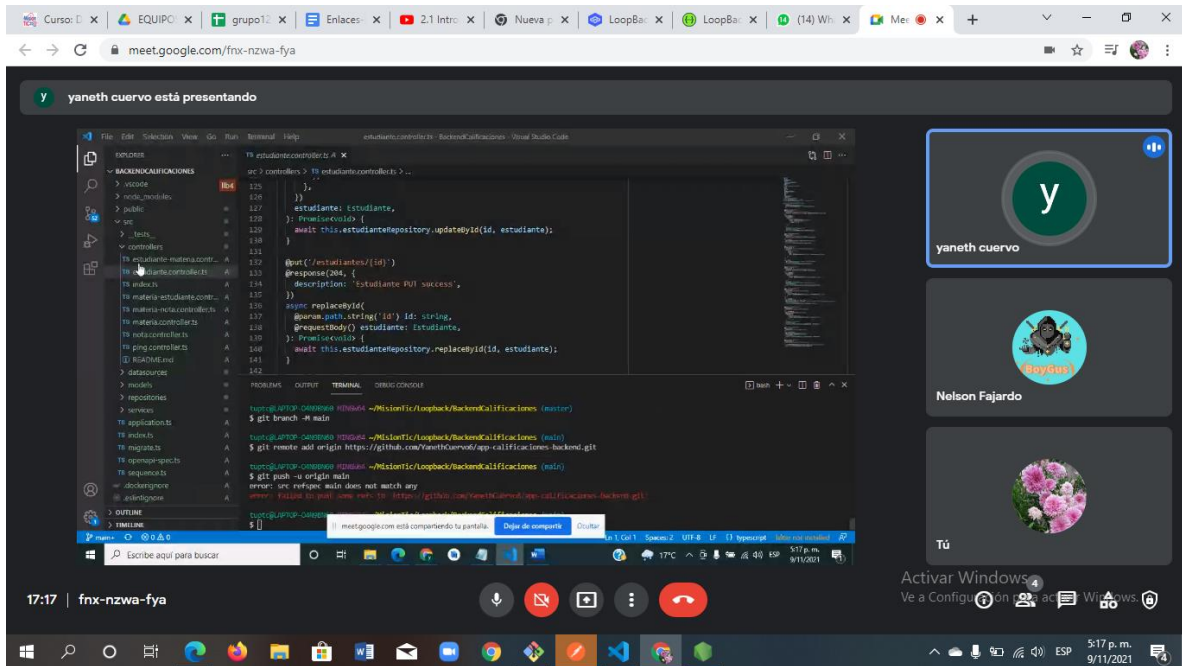
Nombre	Cedula	Rol	Nivel de Participación
CONTRERAS NICOLAS CAMILO		Administrador de Configuración	Alto
DUEÑAS MANRIQUE DIEGO FERNANDO	1026283999	Diseñador UI	Alto
FAJARDO ENRIQUEZ ZULY DANIELA	1086135083	Diseñador de Software	Alto
NELSON GERARDO FAJARDO PATARROYO	80155325	Lider del Equipo	Alto
YANETH MILENA CUERVO BARAHONA	40047928	Tester	Alto

1. Subir las evidencias de las reuniones diarias (pantallazo de las reuniones)

Primera reunión



Segunda reunión



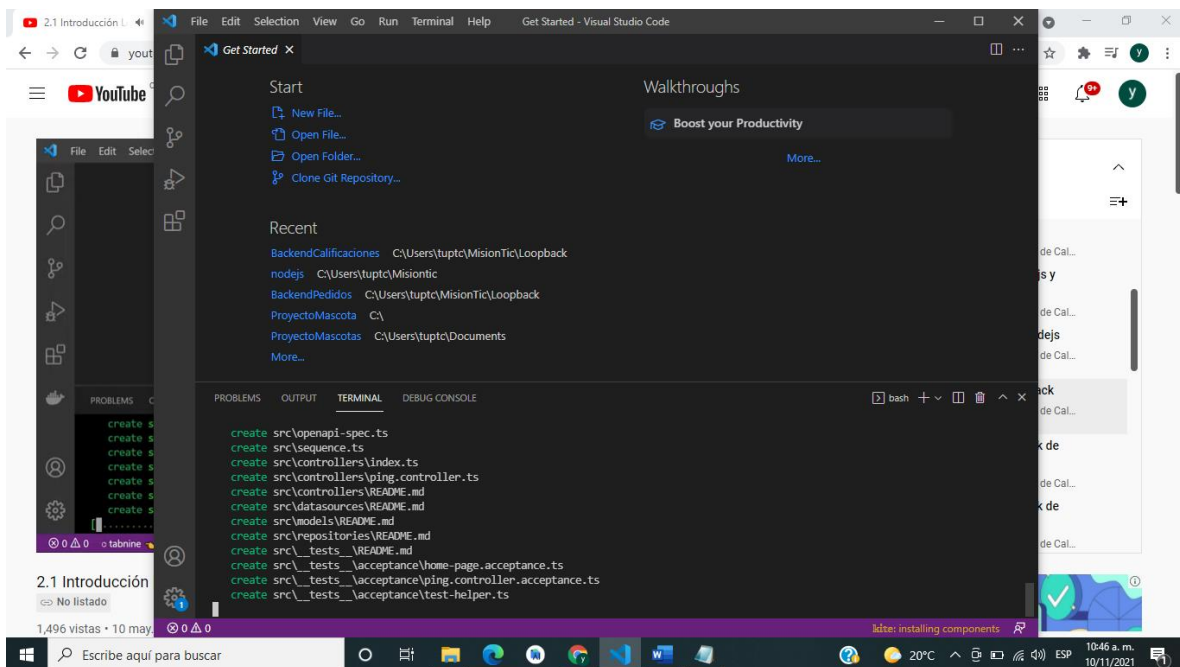
2. Pantallazo del tablero Kanban con las tareas asignadas y realizadas.



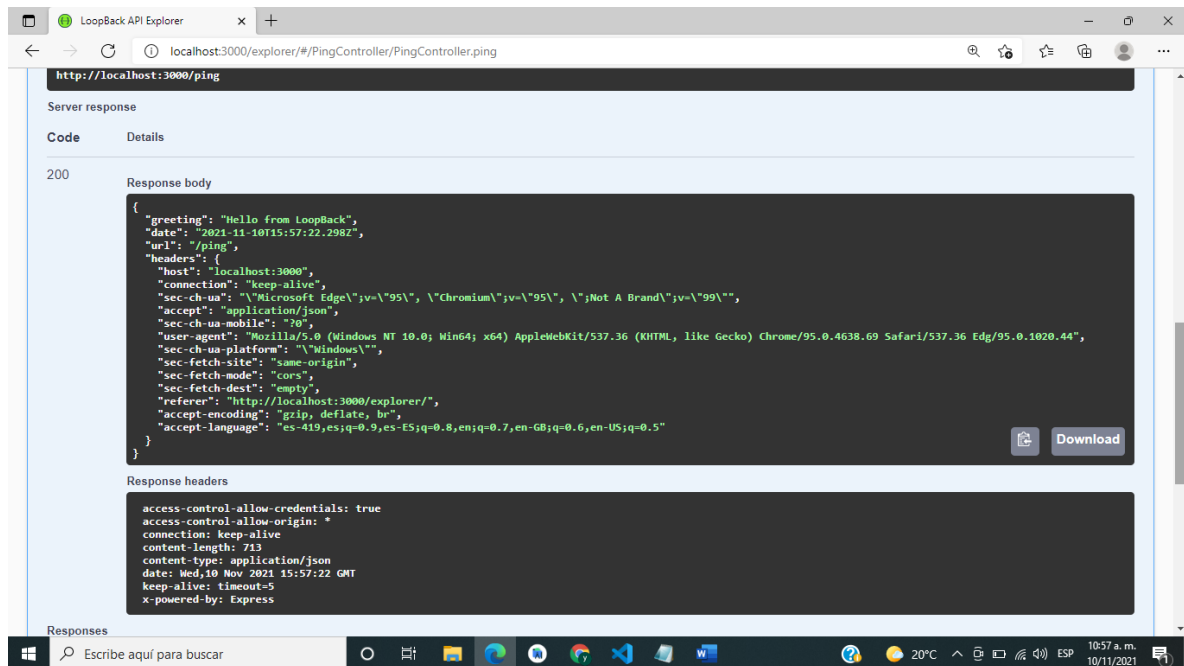
3. Desarrollo del Modelo entidad relación e implementación del proyecto Tienda virtual mascota mediante la plataforma del Loopback.

Paso a paso.

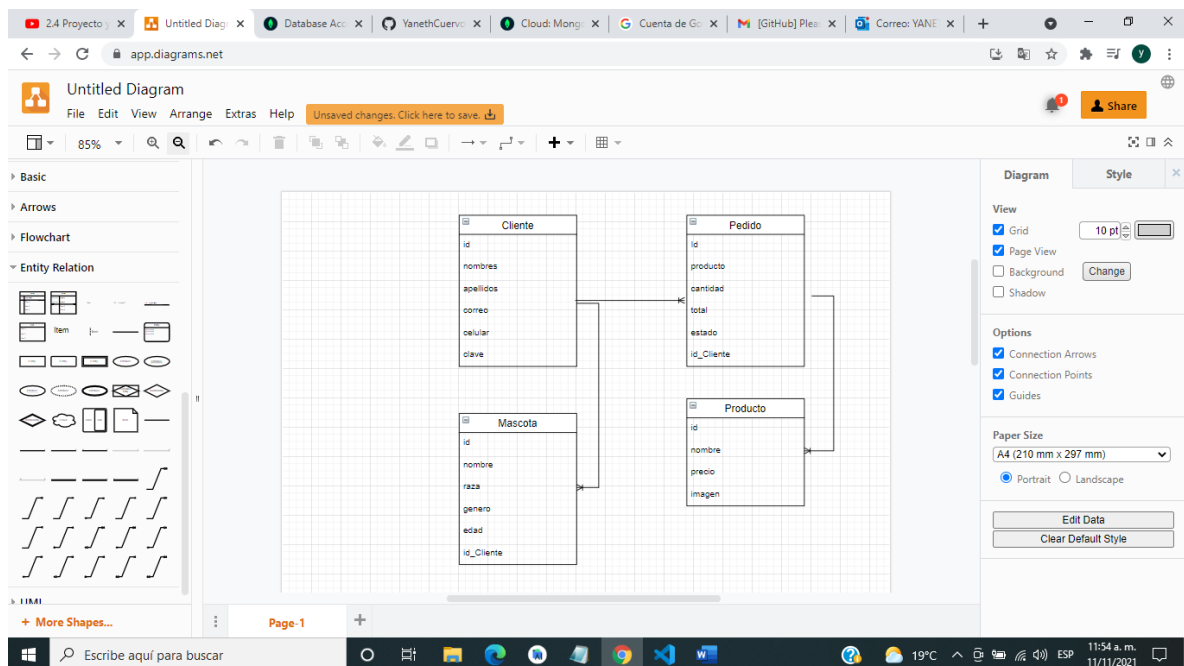
Ingresamos a la terminal luego ingresamos a la carpeta de trabajo para nuestro caso la llamamos **MisionTic**, después ingresamos a la carpeta de trabajo llamada **LoopBack** y dentro de esta creamos nuestra aplicación con la función **lb4 app**, en donde ingresamos los datos solicitados para nuestra aplicación le pondremos al nombre del proyecto **tiendavirtual**



Luego de ingresar la información solicitada validamos el funcionamiento del proyecto en el puerto asignado para nuestro caso el **localhost:3000** mediante la función **npm start**



Luego ingresamos a draw.io en donde vamos a crear las entidades que harán parte de nuestro proyecto



Creamos el datasource

```
src > datasources > TS mongodb.datasource.ts > [0] config
1 import {inject, LifecycleObserver, LifecycleObserver} from '@loopback/core';
2 import {Juggler} from '@loopback/repository';
3
4 const config = {
5   name: 'mongodb',
6   connector: 'mongodb',
7   url: 'mongodb+srv://prog_web:ProgWebMinTic2022@cluster0.ipqur.mongodb.net/TiendaVirtualBD?retryWrites=true&w=maj',
8   host: '',
9   port: 0,
10  user: '',
11  password: '',
12  database: '',
13  useNewUrlParser: true
14 };
15
16 // Observe application's life cycle to disconnect the datasource when
17 // application is stopped. This allows the application to be shut down
18 // gracefully. The 'stop()' method is inherited from 'Juggler.DataSource'.
```

```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
added 10 packages, and audited 587 packages in 16s

68 packages are looking for funding
  run 'npm fund' for details

found 0 vulnerabilities
update src\datasources\index.ts

Datasource MongoDB se ha/han creado en src\datasources

tuptc@LAPTOP-O4N8BN60 MINGW64 ~/MisionTic/Loopback/BackendTiendaVirtual
$ lb4 model
```

Ahora creamos los modelos con la función **lb4 model** para nuestro caso cliente, mascota, pedido, producto

```
src > models > TS mascota.model.ts
1 import {Entity, model, property} from '@loopback/repository';
2
3 @model()
4 export class Mascota extends Entity {
5   @property({
6     type: 'string',
7     id: true,
8     generated: true,
9   })
10  id?: string;
11
12   @property({
13     type: 'string',
14     required: true,
15   })
16  nombre: string;
17
18   @property({
```

```
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
¿Es necesario?: Yes

Vamos a añadir otra propiedad a Pedido
Especifique un nombre de propiedad vacío cuando haya terminado

¿Especifique el nombre de propiedad:
create src\models\pedido.model.ts
update src\models\index.ts

Model Pedido se ha/han creado en src\models

tuptc@LAPTOP-O4N8BN60 MINGW64 ~/MisionTic/Loopback/BackendTiendaVirtual
$ lb4 model
```

Ahora creamos los repositorios con la función **lb4 repository**

The screenshot shows the Visual Studio Code interface with the Explorer view on the left displaying the project structure. The main editor shows the file `src/models/mascota.model.ts` with the following TypeScript code:

```
1 import {Entity, model, property} from '@loopback/repository';
2
3 @model()
4 export class Mascota extends Entity {
5   @property({
6     type: 'string',
7     id: true,
8     generated: true,
9   })
10  id?: string;
11
12   @property({
13     type: 'string',
14     required: true,
15   })
16  nombre: string;
17
18   @property({
```

The bottom panel shows the terminal with the following output:

```
> Descripción de proyecto: mascota
> Directorio raíz de proyecto: mascota
> Nombre de clase de aplicación: }
> Class name is invalid: }
Session contents restored from 10/11/2021 at 1:19:31 p. m.

tuerto@LAPTOP-04N9BN60 MINGW64 ~/MisionTic/Loopback/BackendTiendaVirtual
$ lb4 repository
Selecione el origen de datos MongoDBDatasource
Selecione el modelo o modelos para los que desea generar un repositorio Cliente, Mascota, Pedido
Selecione la clase base del repositorio (Use arrow keys)
> DefaultCrudRepository (Juggler bridge)
----- Repositorios personalizados -----
```

Luego generamos la relación entre nuestras entidades materia y estudiante cuya mediante la opción **lb4 relation**

The screenshot shows the Visual Studio Code interface with the Explorer view on the left. The main editor shows the file `src/models/mascota.model.ts` with the following TypeScript code:

```
28   }
29   genero: string;
30
31   @property({
32     type: 'string',
33     required: true,
34   })
35   edad: string;
36
37   @belongsTo(() => Cliente)
38   clienteId: string;
39
40   constructor(data?: Partial<Mascota>) {
41     super(data);
42   }
43 }
44
45 export interface MascotaRelations {
```

The bottom panel shows the terminal with the following output:

```
$ lb4 relation
Selecione el tipo de relación belongsTo
Selecione el modelo de origen Mascota
Selecione el modelo de destino Cliente
Nombre de clave foránea para definir en el modelo de origen clienteId
Nombre de relación cliente
¿Desea permitir que las consultas de Mascota incluyan datos de las instancias de Cliente relacionadas? Yes
create src/controllers/mascota-cliente.controller.ts

Relation BelongsTo se ha/han creado en src

tuerto@LAPTOP-04N9BN60 MINGW64 ~/MisionTic/Loopback/BackendTiendaVirtual
$
```

The screenshot shows the Visual Studio Code interface with the file explorer on the left, the editor in the center, and the terminal at the bottom. The file explorer shows the project structure with folders like .vscode, node_modules, public, src, and models. The editor displays the file `pedido.model.ts` with the following code:

```
27     required: true,
28   })
29   total: number;
30
31   @property({
32     type: 'number',
33     required: true,
34   })
35   estado: number;
36
37   @belongsTo(() => Cliente)
38   clienteId: string;
39
40   constructor(data?: Partial<Pedido>) {
41     super(data);
42   }
43 }
44
```

The terminal shows the output of the `lb4 relation` command, indicating that the relation `HasMany` has been created successfully between `Pedido` and `Cliente`.

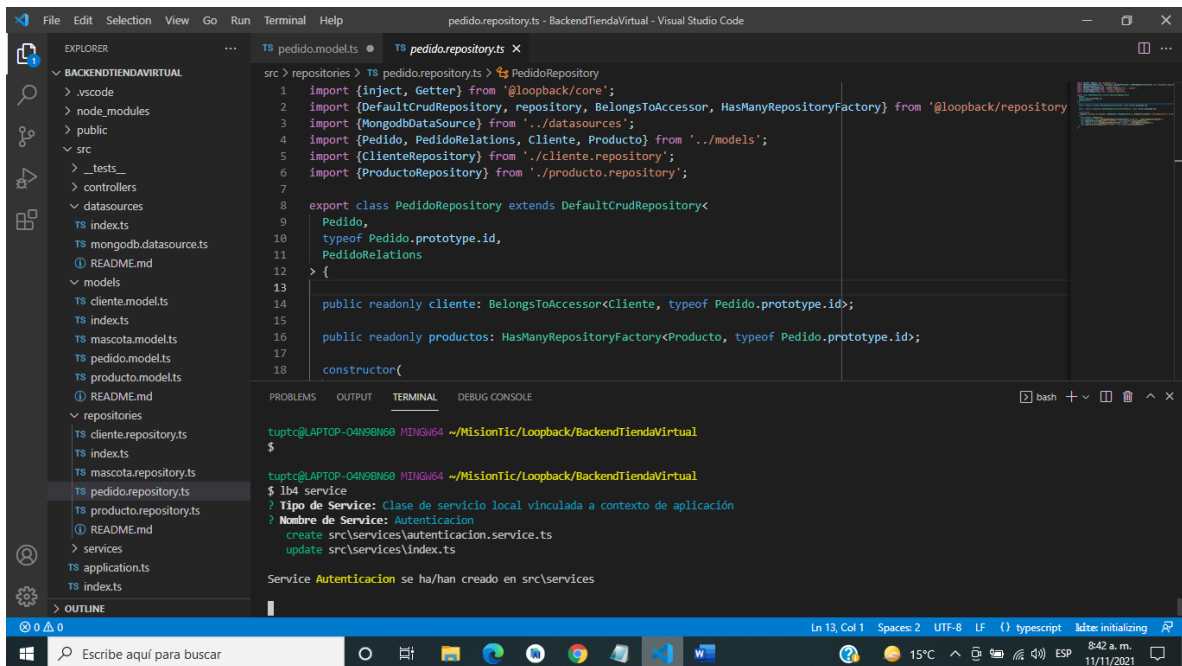
Ahora procedemos a crear los servicios con la función **lb4 service**

Notificacion y Autenticacion

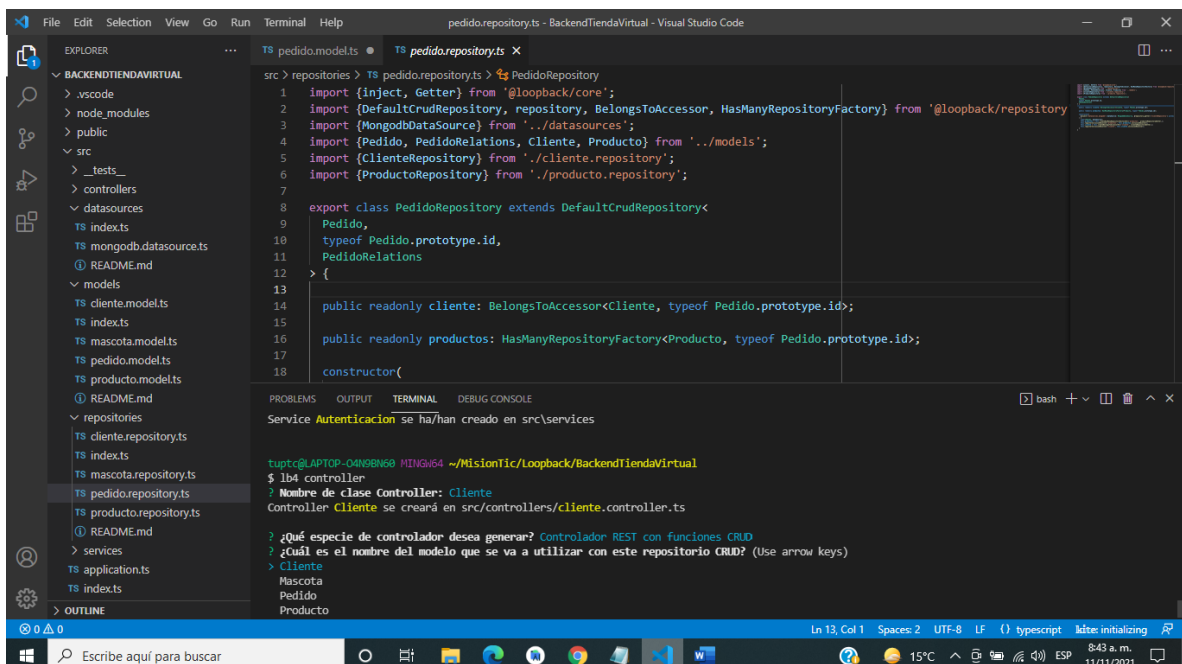
The screenshot shows the Visual Studio Code interface with the file explorer on the left, the editor in the center, and the terminal at the bottom. The file explorer shows the project structure with folders like .vscode, node_modules, public, src, and models. The editor displays the file `pedido.repository.ts` with the following code:

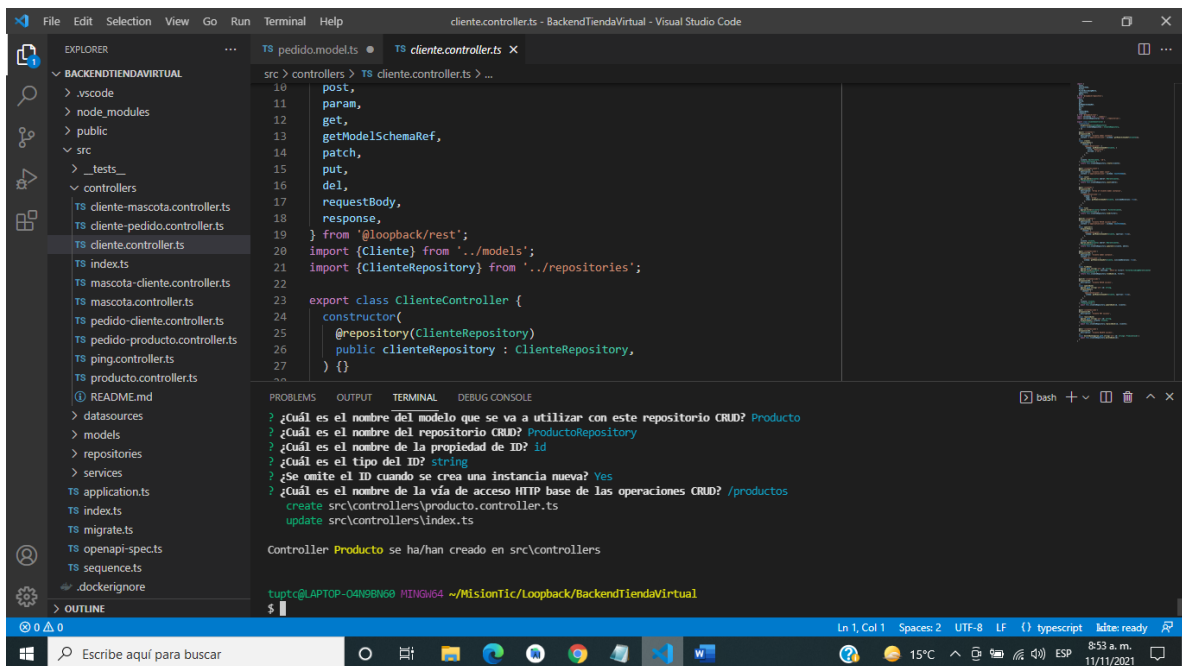
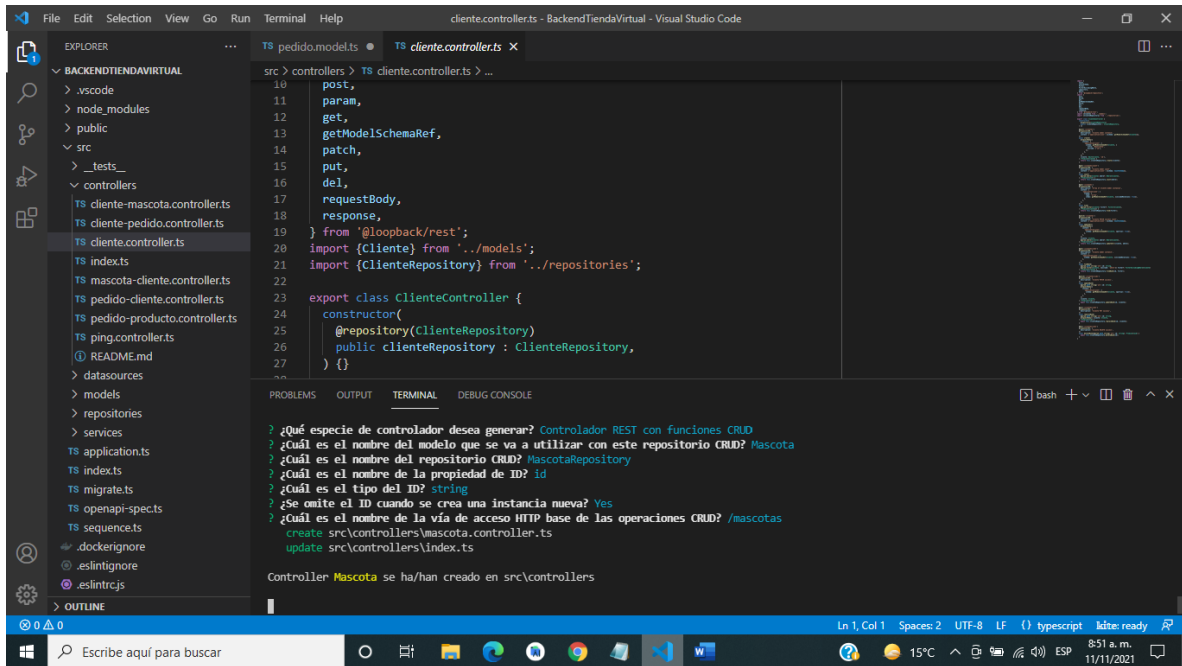
```
1 import { inject, Getter } from '@loopback/core';
2 import { DefaultCrudRepository, repository, BelongsToAccessor, HasManyRepositoryFactory } from '@loopback/repository';
3 import { MongodbDataSource } from '../datasources';
4 import { Pedido, PedidoRelations, Cliente, Producto } from '../models';
5 import { ClienteRepository } from './cliente.repository';
6 import { ProductoRepository } from './producto.repository';
7
8 export class PedidoRepository extends DefaultCrudRepository<
9   Pedido,
10   typeof Pedido.prototype.id,
11   PedidoRelations
12 > {
13
14   public readonly cliente: BelongsToAccessor<Cliente, typeof Pedido.prototype.id>;
15
16   public readonly productos: HasManyRepositoryFactory<Producto, typeof Pedido.prototype.id>;
17
18   constructor(
19
20
```

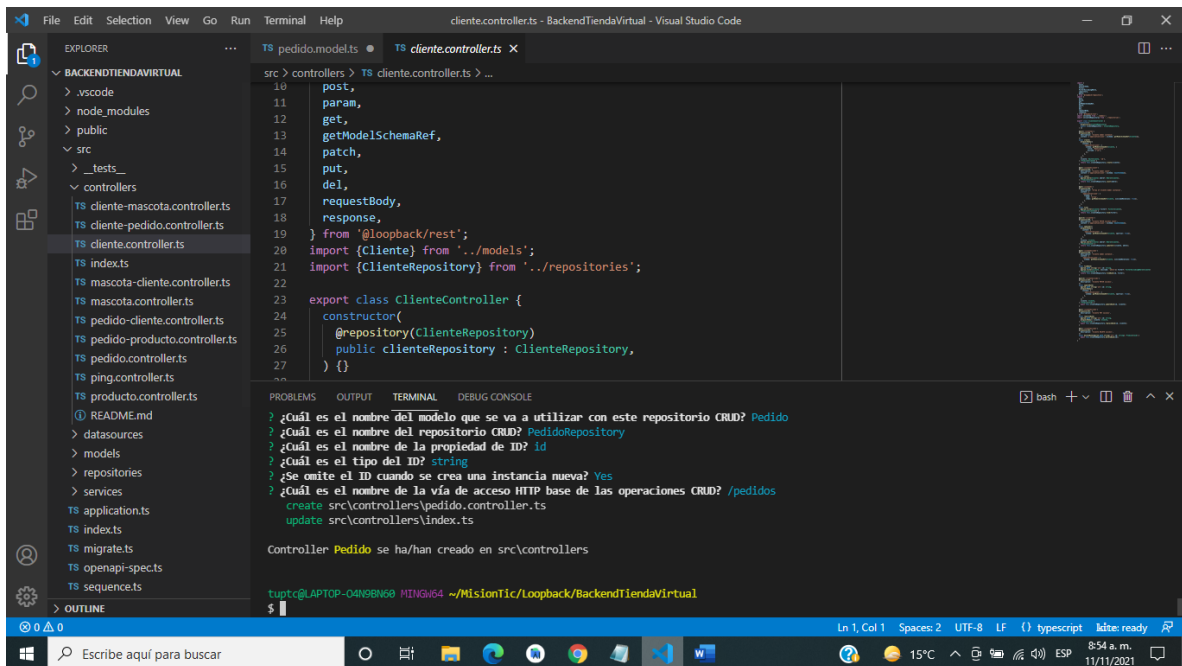
The terminal shows the output of the `lb4 service` command, indicating that the service `PedidoRepository` has been created successfully.



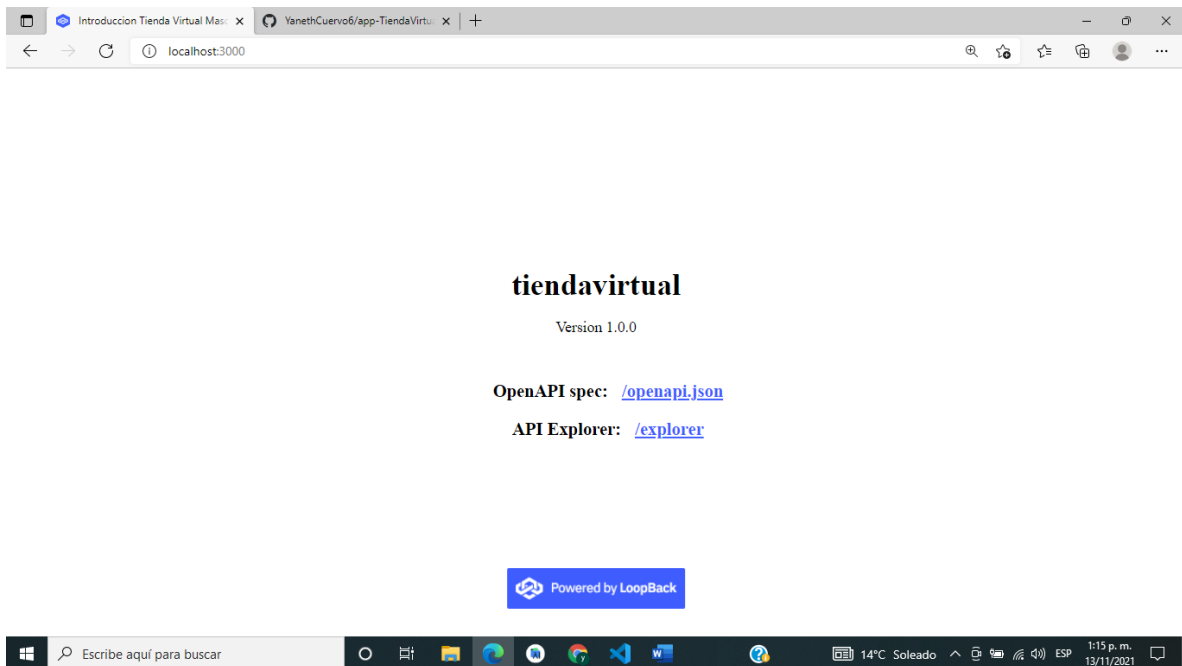
Para finalizar creamos los controladores con la función **lb4 controller** para cada uno de los modelos







Validamos en el localhost:3000 el funcionamiento de nuestra Tienda virtual



LoopBack API Explorer YanethCuervo6/app-TiendaVirtu... | +

localhost:3000/explorer/#/

Filter by tag

- ClienteController
- ClienteMascotaController
- ClientePedidoController
- MascotaController
- MascotaClienteController
- PedidoController
- PedidoClienteController
- PedidoProductoController
- PingController

Windows taskbar: Escribe aquí para buscar, 14°C Soleado, 1:16 p.m. 13/11/2021

LoopBack API Explorer YanethCuervo6/app-TiendaVirtu... | +

localhost:3000/explorer/#/ClienteController/ClienteController.create

Parameters Cancel Reset

No parameters

Request body application/json

```
{  "nombres": "Sofia Isabella",  "apellidos": "Barrera",  "correo": "sofibarrera@hotmail.com",  "celular": "3124567865",  "direccion": "avenida orienta",  "clave": "000000"}
```

Execute

Responses

Windows taskbar: Escribe aquí para buscar, 14°C Soleado, 1:18 p.m. 13/11/2021

LoopBack API Explorer YanethCuervo6/app-TiendaVirtu... | +

localhost:3000/explorer/#/ClienteController/ClienteController.create

```
"direccion": "avenida orienta",
"clave": "00000"
}
```

Request URL

http://localhost:3000/clientes

Server response

Code Details

200

Response body

```
{
  "id": "619002829df9350774aa9094",
  "nombres": "Sofia Isabella",
  "apellidos": "Barrera",
  "correo": "sofibarrera@hotmail.com",
  "celular": "3124567865",
  "direccion": "avenida orienta",
  "clave": "00000"
}
```

Response headers

```
access-control-allow-credentials: true
access-control-allow-origin: *
connection: keep-alive
content-length: 186
content-type: application/json
date: Sat, 13 Nov 2021 18:23:00 GMT
keep-alive: timeout=5
x-powered-by: Express
```

Responses

Code	Description	Links
200		

Windows taskbar: 14°C Soleado, 1:23 p.m., 13/11/2021

LoopBack API Explorer YanethCuervo6/app-TiendaVirtu... | +

localhost:3000/explorer/#/ClienteController/ClienteController.find

PATCH /clientes

GET /clientes

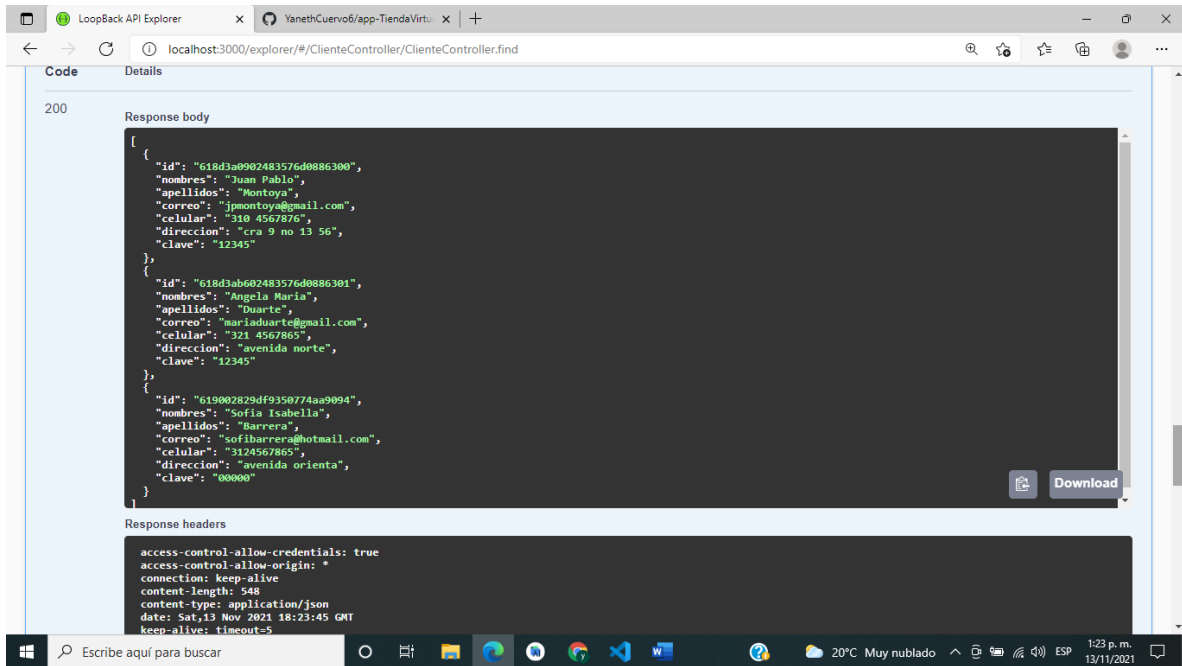
Parameters

Cancel

Name	Description
filter	
object	
(query)	

Execute Clear

Windows taskbar: 20°C Muy nublado, 1:23 p.m., 13/11/2021



Subimos el repositorio al github link del repositorio en el GITHUB

<http://github.com/YanethCuervo6/app-TiendaVirtual-Backend>

[YanethCuervo6/app-TiendaVirtual-Backend: Backend para la aplicacion de la tienda virtual de mascotas \(github.com\)](http://github.com/YanethCuervo6/app-TiendaVirtual-Backend)

