

# E-Ticaret Demo Api

## **Prisma ORM Kullanarak CRUD İşlemleri**

- 1.Proje Amacı
- 2.Kullanılan Teknolojiler
- 3.Çalışma Ortamının Kurulumu
- 4.Postgresql ve Docker Kurulumu
- 5.Veritabanı Yapılandırması ve Tablo Oluşturma
- 6.Veri ekleme ve Test
- 7.NestJS ve Prisma ORM Kurulumu
- 8.Enpointler

## 1.Proje Amacı

Bir E-ticaret sitesinin backend tarafının yer aldığı örnek bir projenin tablolarının oluşturulması ve Nestjs'de backend projesinin yapılarak CRUD endpoint'lerinin yazılması.

**2.Kullanılan Teknolojiler** Database: Postgresql, Docker üzerinde Veritabanını yönetmek için: JetBrains Datagrip ,Backend: NestJs Framework, ORM: Prisma, Swagger: Endpoint'lerin testi için arayüz

## 3.Çalışma Ortamının Kurulumu

Docker Kurulumu

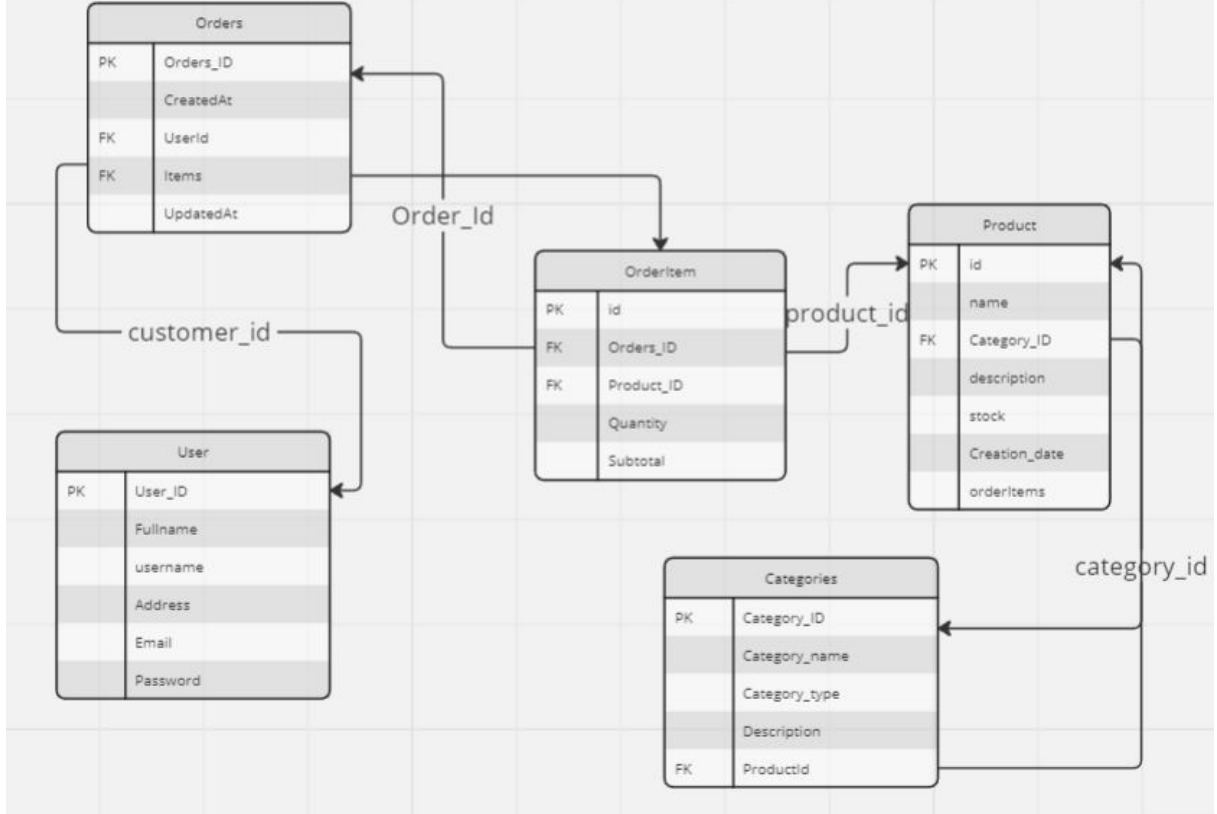
```
docker run --name proje -e POSTGRES_USER=batuhan-e POSTGRES_PASSWORD=*****  
-p 5432:5432 -d postgres
```

**NestJs CLI Kurulumu** \$ npm install -g @nestjs/cli

**Prisma ORM'i Projeye Ekleme** npm install prisma --save-dev

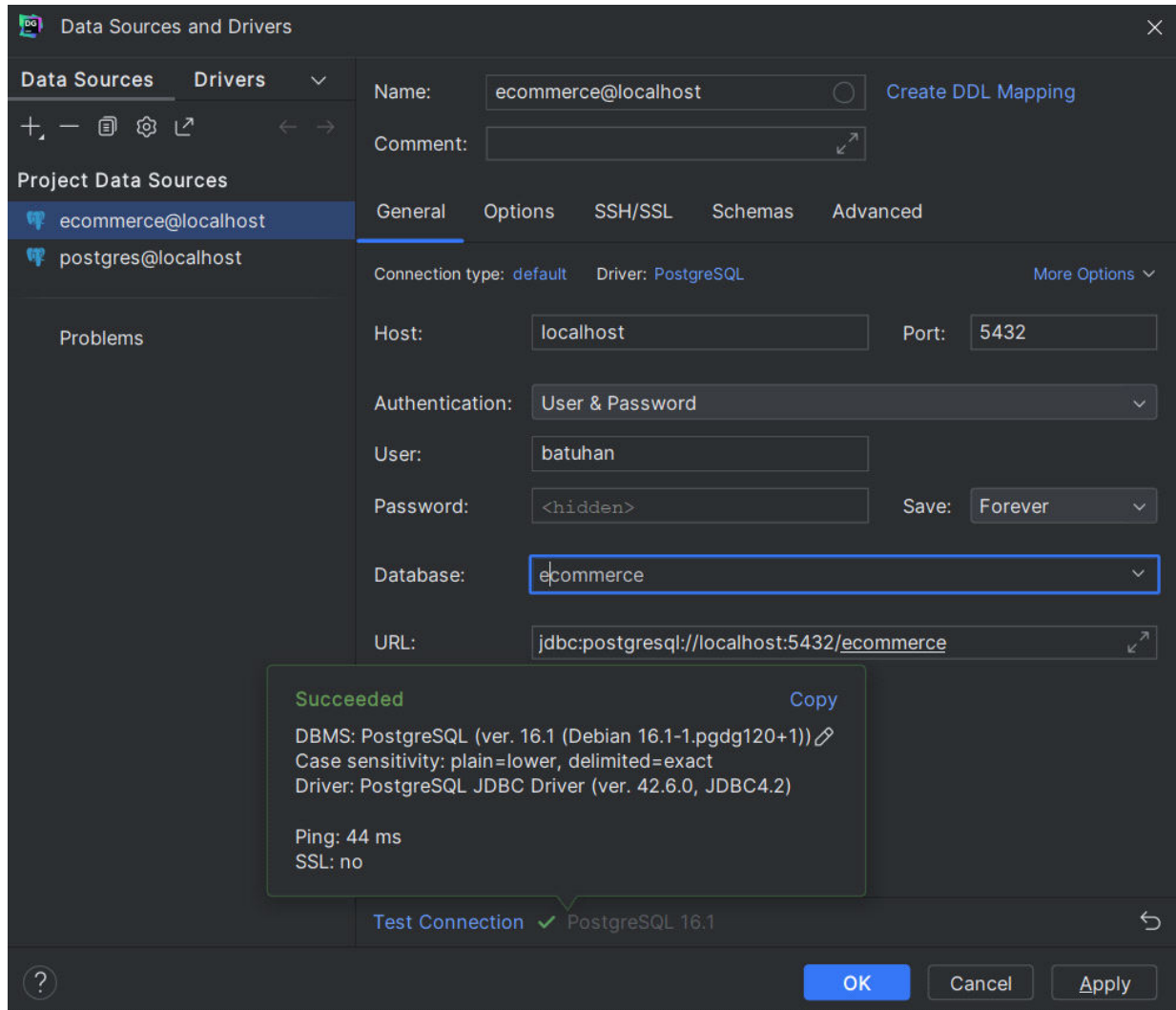
## 4. Veritabanı Tasarımı

# E-TİCARET VERİ TABANI ERD



## 5. Veritabanı Yapılandırması ve Tablo Oluşturma

Datagrip ile veritabanına bağlantı



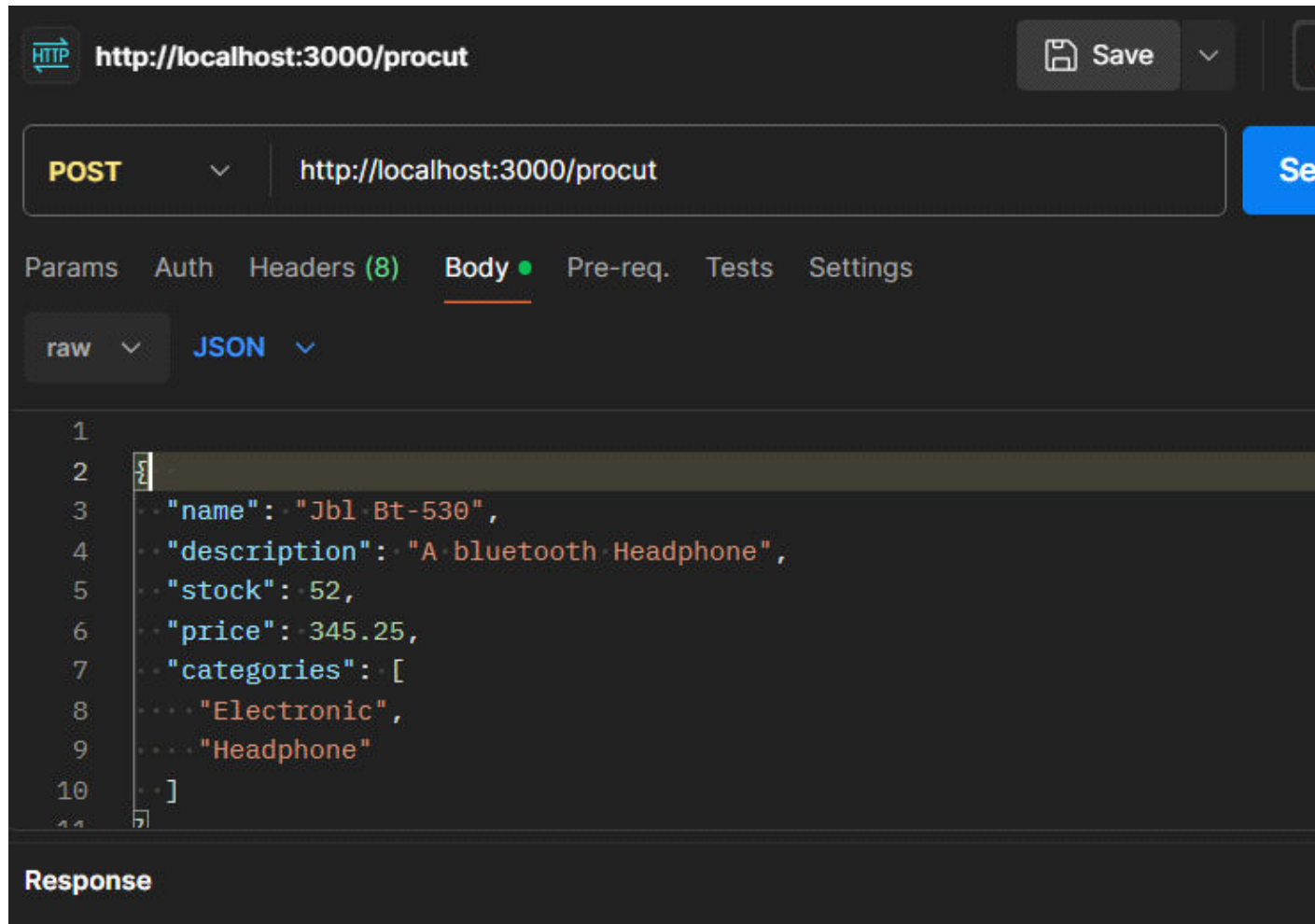
## Prisma üzerinde modellerin oluşturulması

```
model User {
  id          Int      @id @default(autoincrement())
  username    String    @unique
  email       String    @unique
  hashedPassword String
  fullName    String
  address     String?
  orders      Order[]
  created_at  DateTime @default(now())
  updated_at  DateTime @updatedAt
}

model Product {
  id          Int      @id @default(autoincrement())
  name        String
  description  String
  price       Float
  stock       Int
  categories  Category[]
  orderItems  OrderItem[]
  created_at  DateTime @default(now())
}
```

## 7. Veri ekleme

Postman üzerinden örnek veri girişi.



## 8. NestJS ve Prisma ORM Kurulumu

Lokal ortamda NestJS CLI kurulumu. (kurulum komutları)

<code>npm i -g @nestjs/cli</code>	Bir nestjs uygulaması oluşturur
<code>npm install @prisma/client</code>	Prisma ormsini projeye ekler

Prisma ORM entegrasyonu. (komutlar ve tanımları)

<code>npx prisma init</code>	Prisma paketinin yapılandırılmasını sağlar
<code>npx prisma migrate dev</code>	Prisma schemasında tanımlamayı migrate eder
<code>npx prisma db push</code>	şemaları uygular ve veritabanını günceller
<code>npx prisma generate</code>	Prisma modellerini sql kodlarına çevirir

## 9. Enpointler

Path	Method	Description
<code>/auth/signup</code>	post	Yeni bir kullanıcı oluşturur
<code>/product</code>	patch	Var olan bir ürünü günceller

/category	get	Kategorileri gösterir
/order	delete	Bir siparişı siler

**Bağlantı adresi :**

<https://e-commerce-api.azurewebsites.net/swagger>

**Github deposu :**

<https://github.com/K0twHan/E-commerce-api-demo>