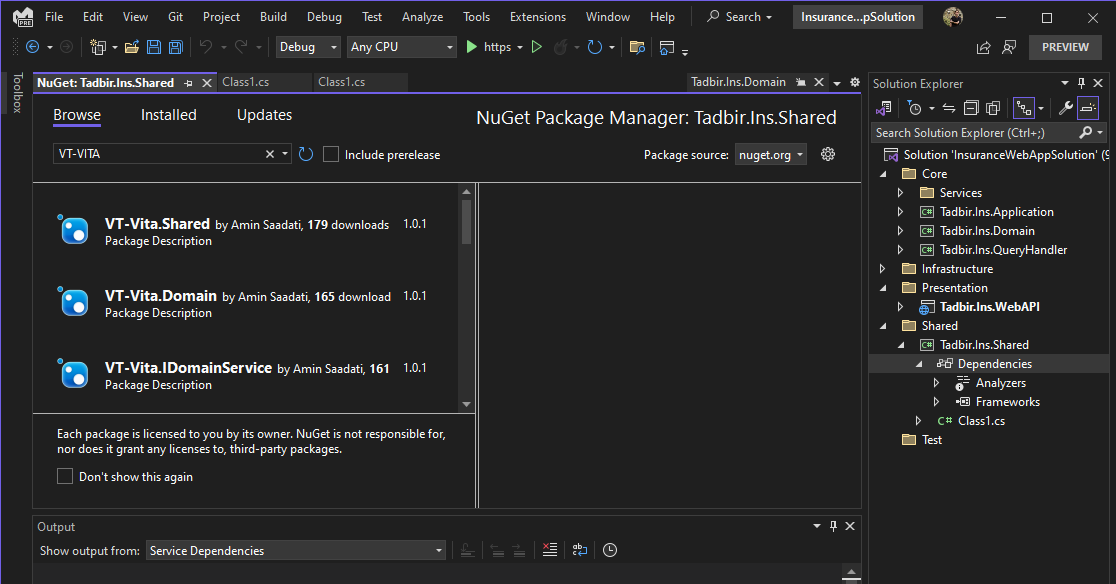
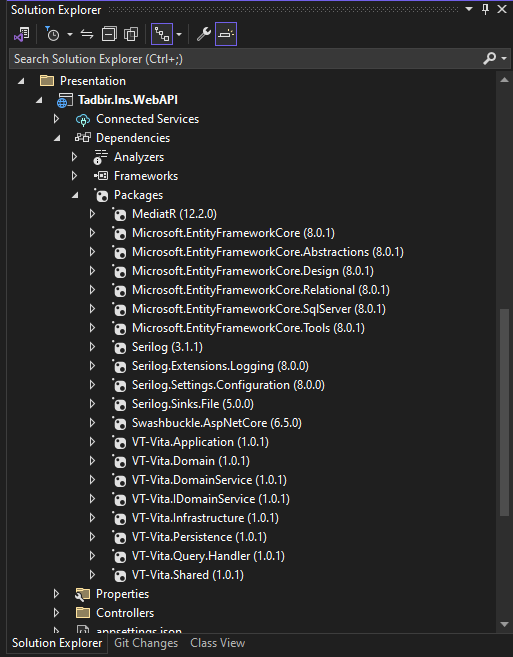
بسمه تعالی

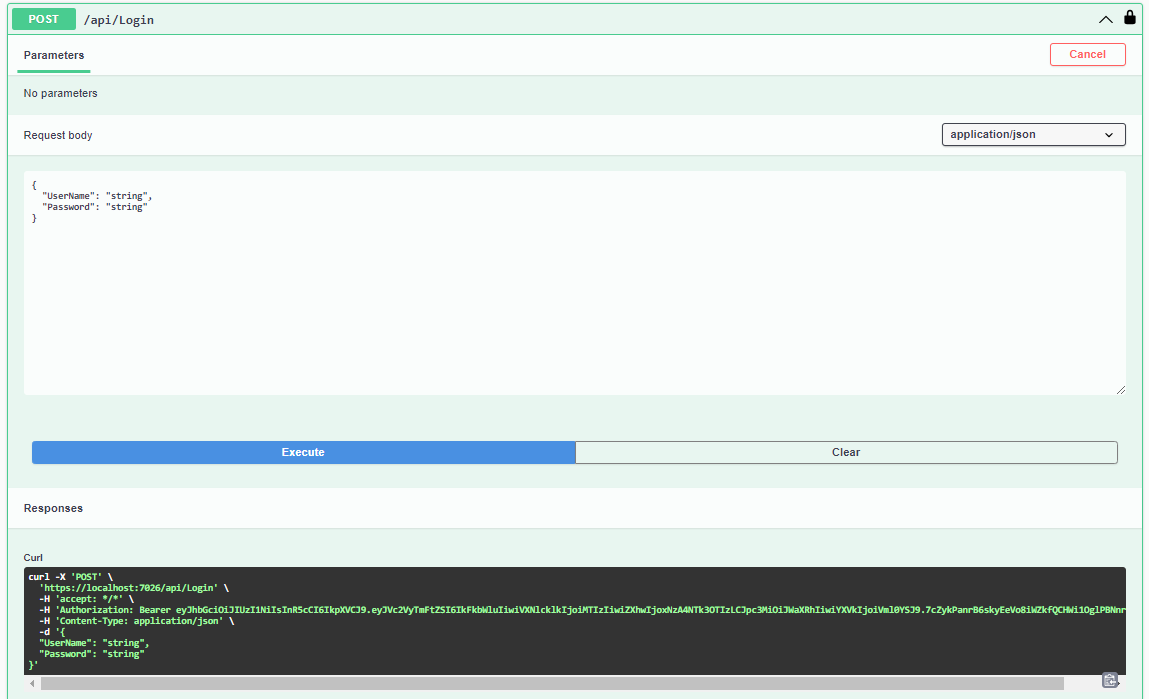
* **راهنما و توضیحات**

1. **در پروژه ها از پکیج هایی به نام VT-Vita به ازای هر لایه استفاده شده . این پکیج ها در Nuget package manager قابل دسترسی هستند و توسط خودم نوشته شده است که خیلی از امکانات رو برای پیاده سازی پروژه شما و چه ... به من میدهد به همین دلیل از آن استفاده کرده ام. این پکیج ها یکسری قابلیت ها به پروژه ها اضافه میکند که یکی از آنها خدمات کامل جهت کار با DbContext است که در پروژه خواسته شده استفاده شده است.**
2. **معماری ساختاری Clean arch است و پترن مورد استفاده CQRS می باشد.**
3. **سیستم قابلیت لاگ دارد و فایل لاگ در پروژه WebAPI ساخته میشود در صورت بروز خطا میتوانید بررسی کنید.**
4. **سیستم مدیریت خطا به صورت Global را دارد**
5. **تصویر از معماری بکار در رفته دریافت شده از NUGET PACKGE**

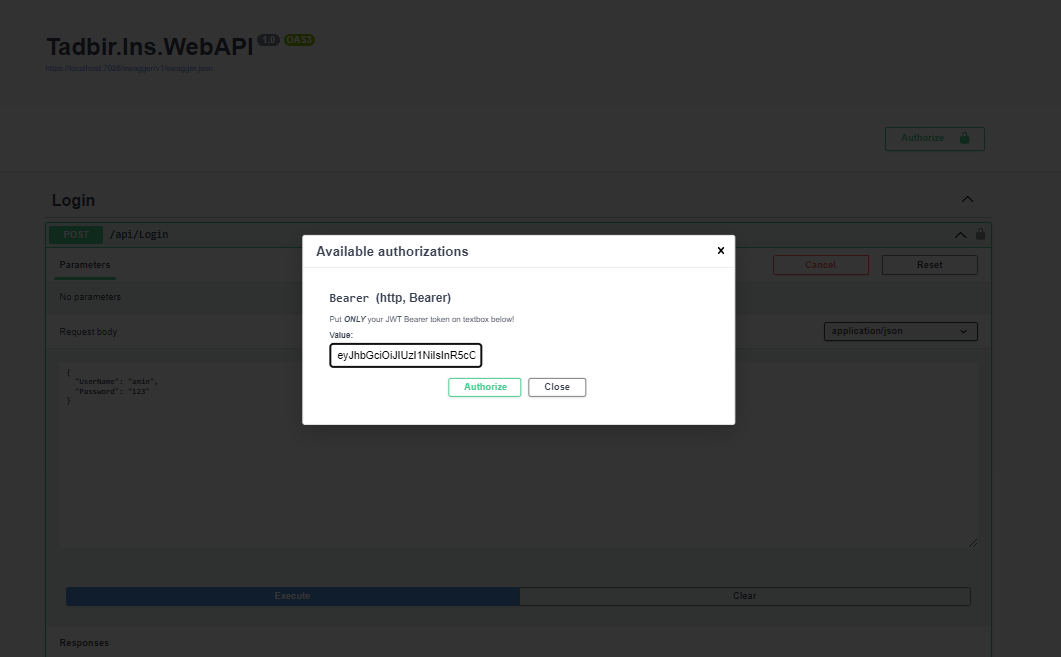




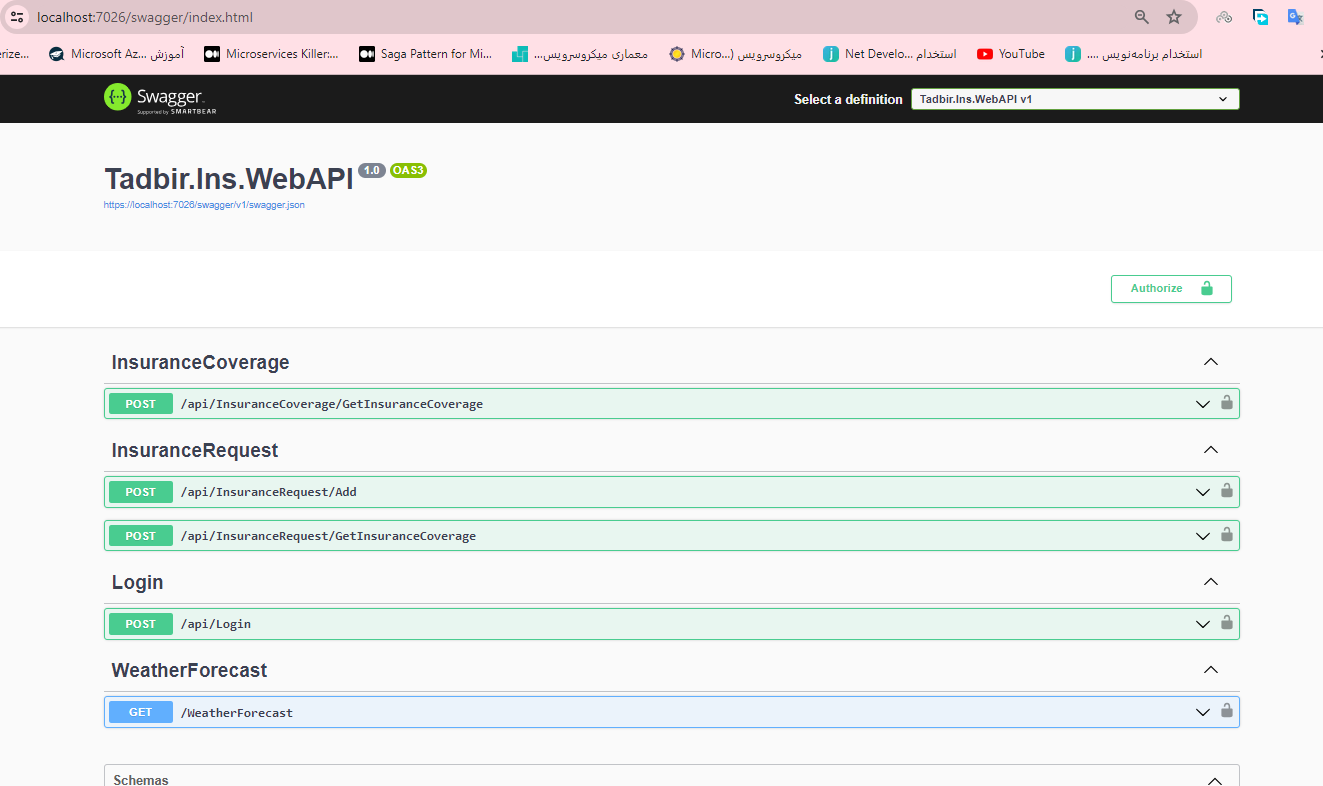
1. **برای پیاده سازی امنیت برنامه از JWT استفاده شده که در مسیر انجام پروژه به ان نیاز دارم و از آن نیز استفاده خواهم کرد.لطفا قبل از فراخوانی متد ها ، متد مربوط به لاگین را بدون وارد کردن نام کاربری و رمز عبور فراخوانی کنید و سپس در قسمت ATHUORIZE مربوط به SWAGGER قرار دهید. زمان توکن یک ساعت است.**



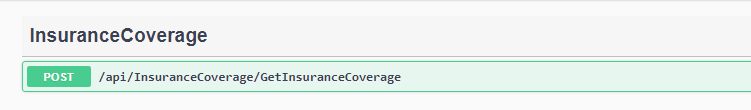
**COPY TOKEN**



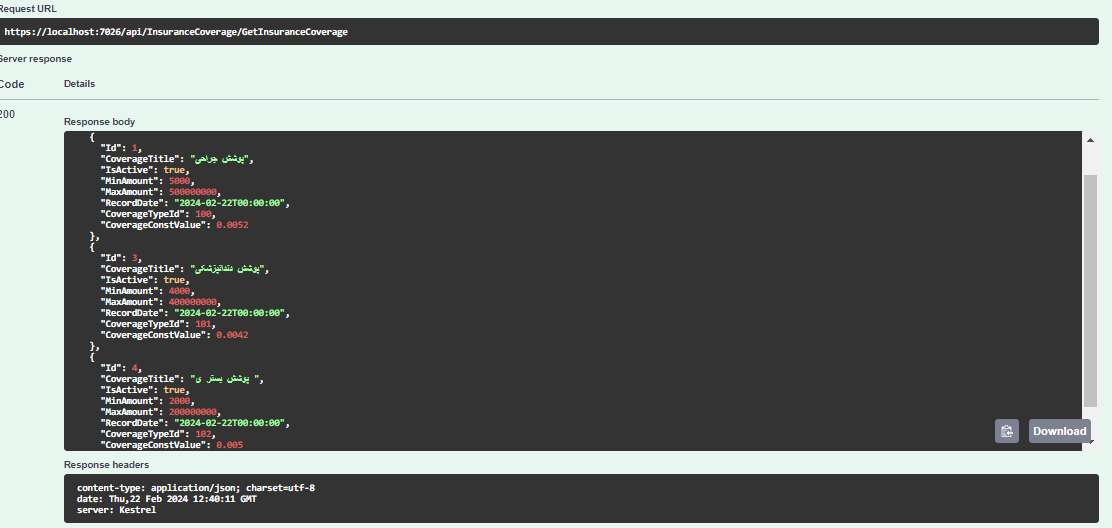
1. **مراحل استفاده : (بدلیل رویکرد پیاده سازی لطفا دقیقا همانند توضیحات زیر فراخوانی ها را انجام دهید)**
2. **در گام اول پس اجرای پروژه از سلامت پکیج ها مطمن شوید**
3. **بایستی مایگریت شما انجام شود پس دیتابیس ایجاد شده را چک کنید (بررسی connectionstring in appsetting). نام دیتابیس** TadbirInsuranceDB
4. **اسکریپت ثبت اطلاعات پایه را اجرا کنید. (در قسمت web api و در پوشه Script) گذاشته شده است. در ایمیل هم پیوست شده است.**
5. **در تصویر زیر پروژه اجرا شده را در اولین زمان خواهید دید.**



1. **لطفا فقط لاگین را فراخوانی اجرا کنید. نیازی به نام کاربری و پسورد نیست. سپس مقدار توکن بدست آمده را در قسمت دکمه Authorize وارد نمایید. تمامی متد ها دسترسی را بدست خواهند آورد.**
2. **برای مشاهده اطلاعات پایه بایستی متد GetInsuranceCoverage را فراخوانی کنید و نیازی به وارد کردن مقدار نیست و وفقط فراخوانی کنید. در این صورت میتواند مقدار مربوط به Id را دریافت کنید. از آنجایی که پروژه خواسته شده فقط بک اند بوده و فرانت نداریم بایستی Id را از این قسمت مشاهده کنید و برای متد بعدی از ان استفاده کنید. (در سمت فرانت کمبو باکس حالت انتخابی می دهد)**



**و نتیجه شما لیست اطلاعات پایه نوع پوشش ها خواهد بود**



1. **ثبت درخواست :**

**برای ثبت درخواست بدلیل محدودیت های رویکرد معماری Vita بایستی مقدار Id هر Entity در زمان ثبت خالی باشد در غیر اینصورت سیستم با خطا مواجه میشود.**

**نمونه json استفاده شده :**

{

  "Title": "درخواست اولی",

  "IsActive": true,

  "InsuranceRequestCoverages": [

    {

      "Id": 0,

      "InsuranceCoverageId": 1,

      "Amount": 5000

    },

 {

      "Id": 0,

      "InsuranceCoverageId": 3,

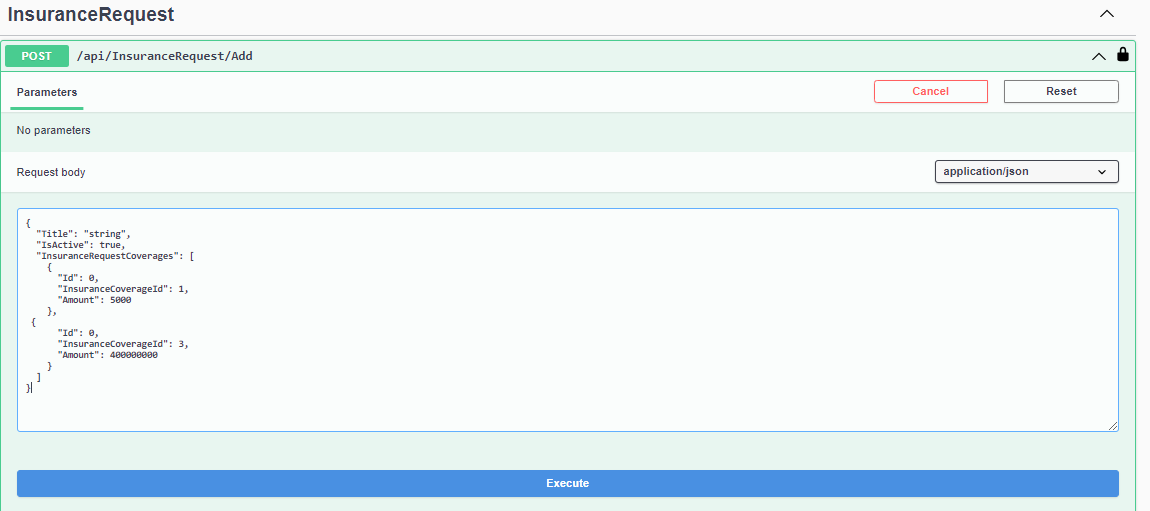
      "Amount": 400000000

    }

  ]

}

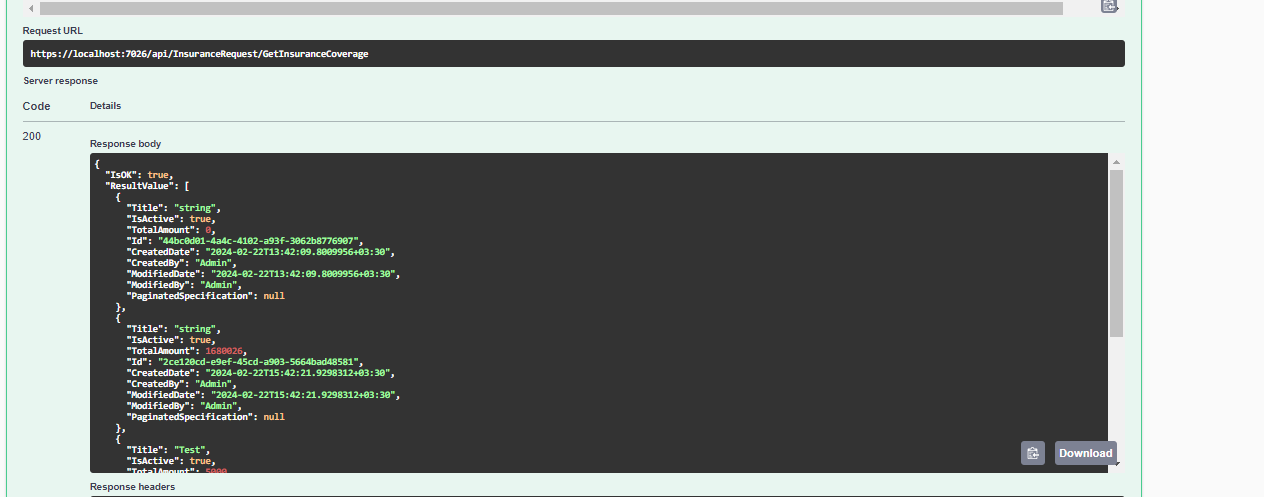
**نمونه فوق را میتواند جایگزی مقدار JSON سوگر پروژه کنید یا لطفا مقادیر Id از json حذف گردد.**



1. **دریافت لیست درخواست های ثبت شده**

**با استفاده از متد** GetInsuranceCoverage

**میتوانید فراخوانی انجام داده و کل درخواست های سیستم را مشاهده کنید.لطفا فقط فراخوانی را انجام دهید و مقدار ورودی نیاز ندارد. قابلیت pagination وجود دارد در سمپل اما چون خواسته نشده بود پیاده سازی نکردم.**



1. **برخی کد ها که در پروژه ساتفده شده است است**

public class InsuranceRequestValidator : VitaFluentValidator<InsuranceRequestDto>

{

    public InsuranceRequestValidator()

    {

        //RuleFor(model => model.Price).GreaterThan(5).WithMessage("Price Must be Greater than 5 : Manual Message");

        RuleFor(model => model.Title).NotEmpty().NotNull().WithMessage(" عنوان را پرکنید");

    }

}

