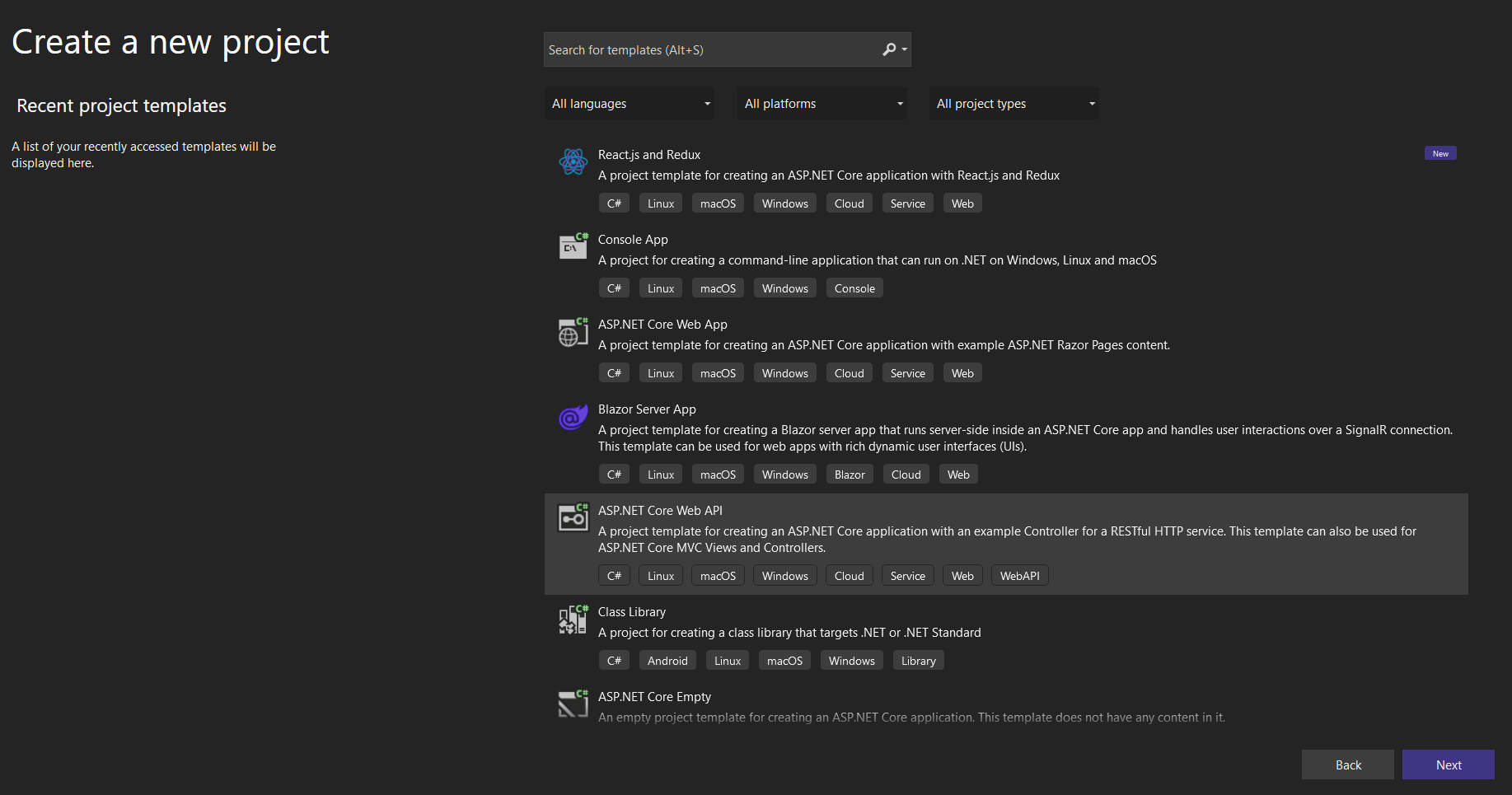
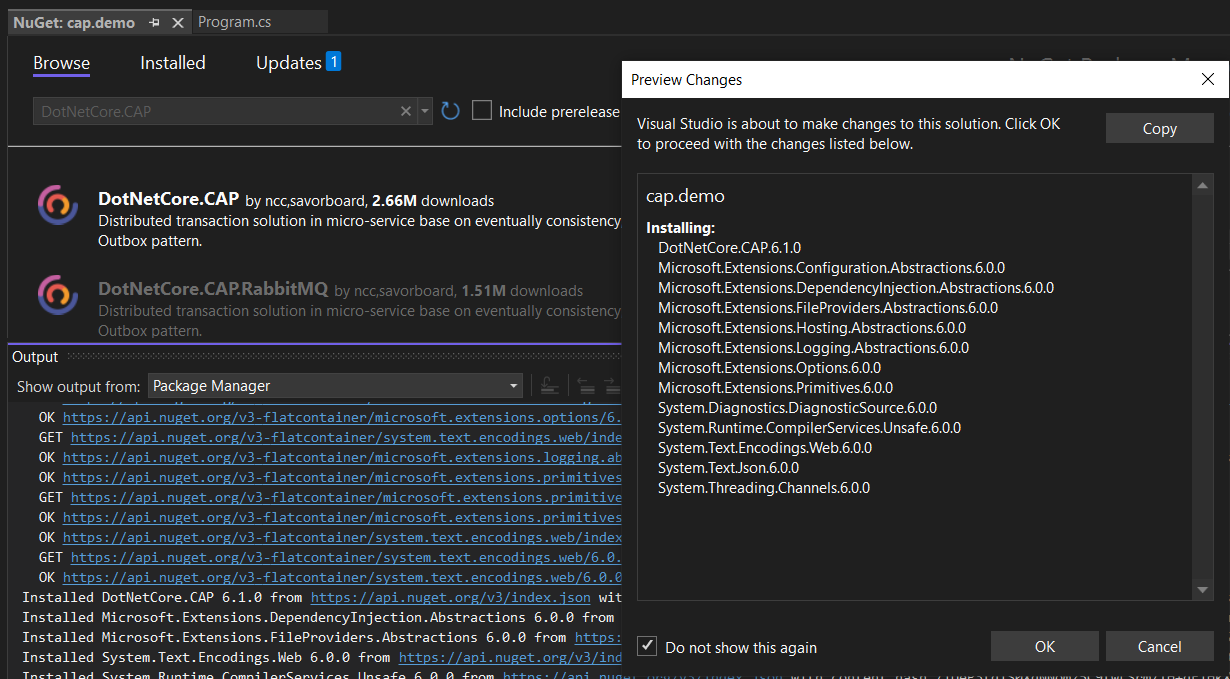
新建Web Api项目



安装依赖DotNetCore.CAP



这里我们先用InMemory的方式模拟数据存储和消息队列：

安装依赖：DotNetCore.CAP.InMemoryStorage

安装依赖：Savorboard.CAP.InMemoryMessageQueue

最新版本的.net core框架，找到program.cs, 加入：

using Savorboard.CAP.InMemoryMessageQueue;

builder.Services.AddCap(x =>

{

x.UseInMemoryStorage();

x.UseInMemoryMessageQueue();

});

添加PublishController

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Mvc;

using DotNetCore.CAP;

namespace cap.demo.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class PublishController : ControllerBase

{

[Route("~/send")]

[HttpGet]

public IActionResult SendMessage([FromServices] ICapPublisher capBus)

{

var header = new Dictionary<string, string>()

{

["my.header.first"] = "first",

["my.header.second"] = "second"

};

capBus.Publish("test.show.time", DateTime.Now, header);

//capBus.Publish("test.show.time", DateTime.Now);

return Ok();

}

}

}

添加ConsumerController

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Mvc;

using DotNetCore.CAP;

namespace cap.demo.Controllers

{

[Route("api/[controller]")]

[ApiController]

public class ConsumerController : ControllerBase

{

[NonAction]

[CapSubscribe("test.show.time")]

public void ReceiveMessage(DateTime time, [FromCap] CapHeader header)

{

Console.WriteLine("message time is:" + time);

Console.WriteLine("message firset header :" + header["my.header.first"]);

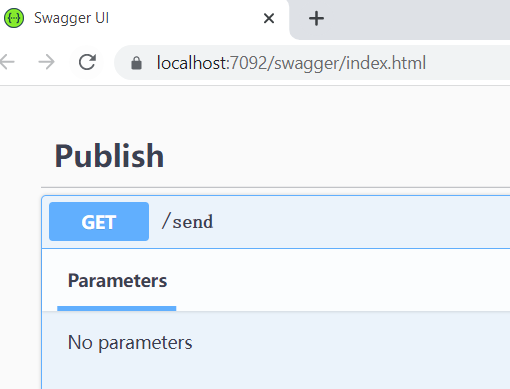
Console.WriteLine("message second header :" + header["my.header.second"]);

}

}

}

最后，运行项目，在Swagger中找到对应方法并执行



参考代码：

<https://github.com/yang-xiaodong/Savorboard.CAP.InMemoryMessageQueue/blob/master/samples/InMemorySample/Startup.cs>

<https://github.com/dotnetcore/CAP/tree/master/samples>

<https://cap.dotnetcore.xyz/user-guide/en/getting-started/quick-start/>

<https://blog.csdn.net/qq_32379229/article/details/124640803>