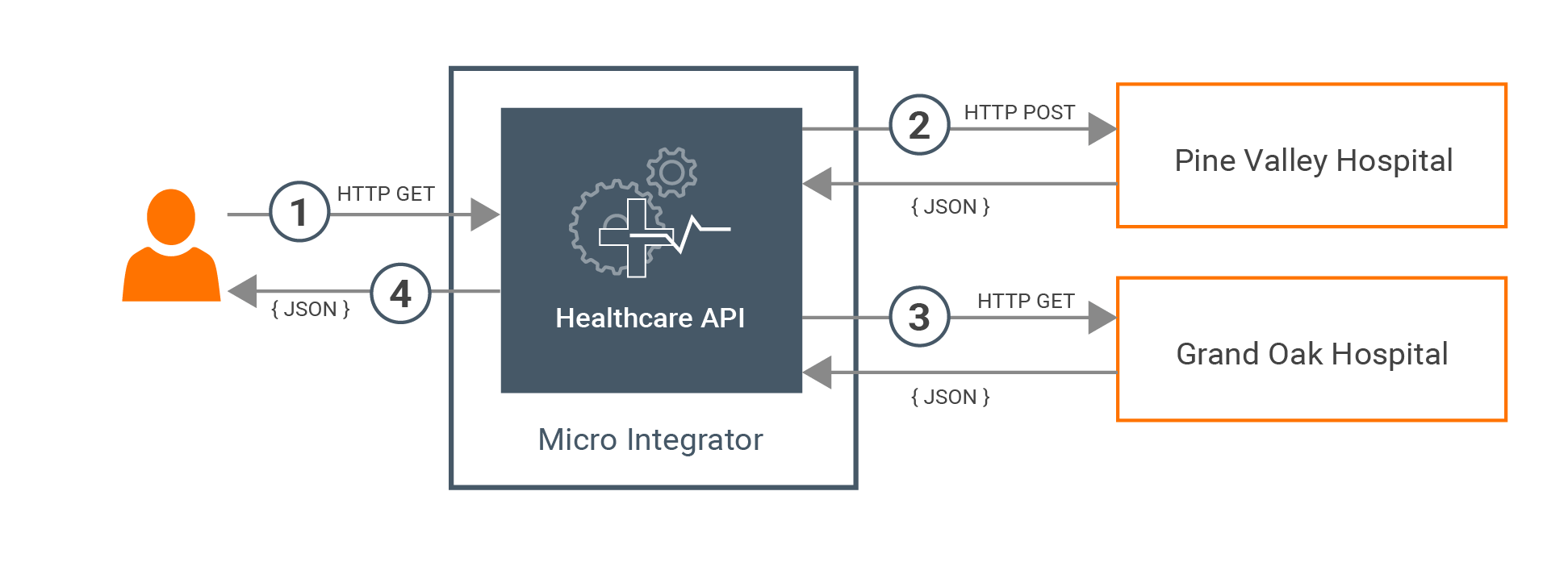
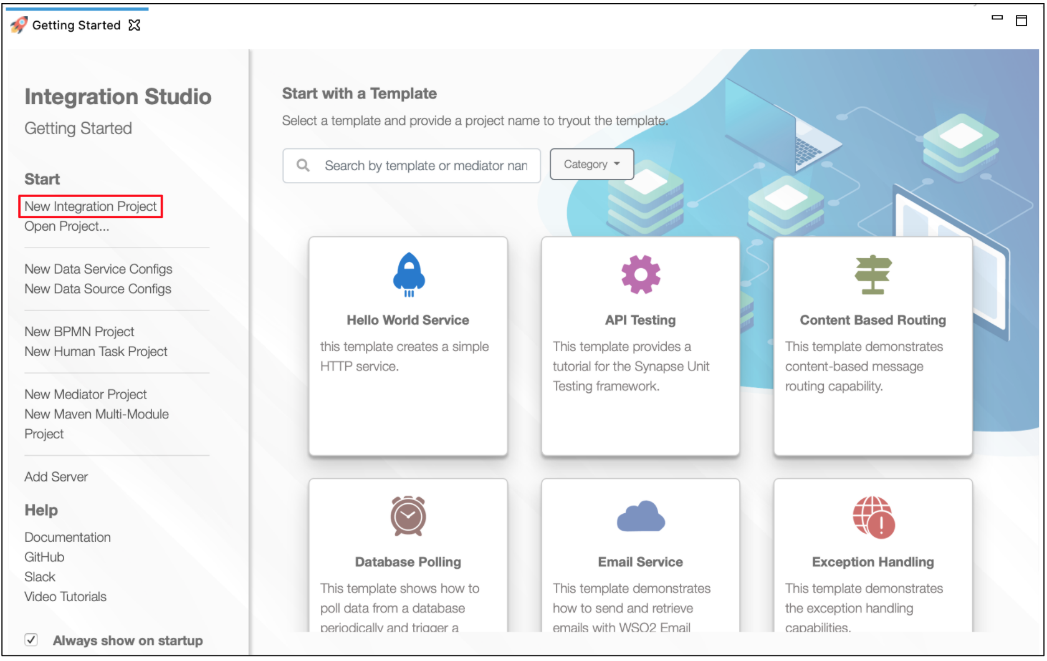
ایجاد یک پروژه ساده

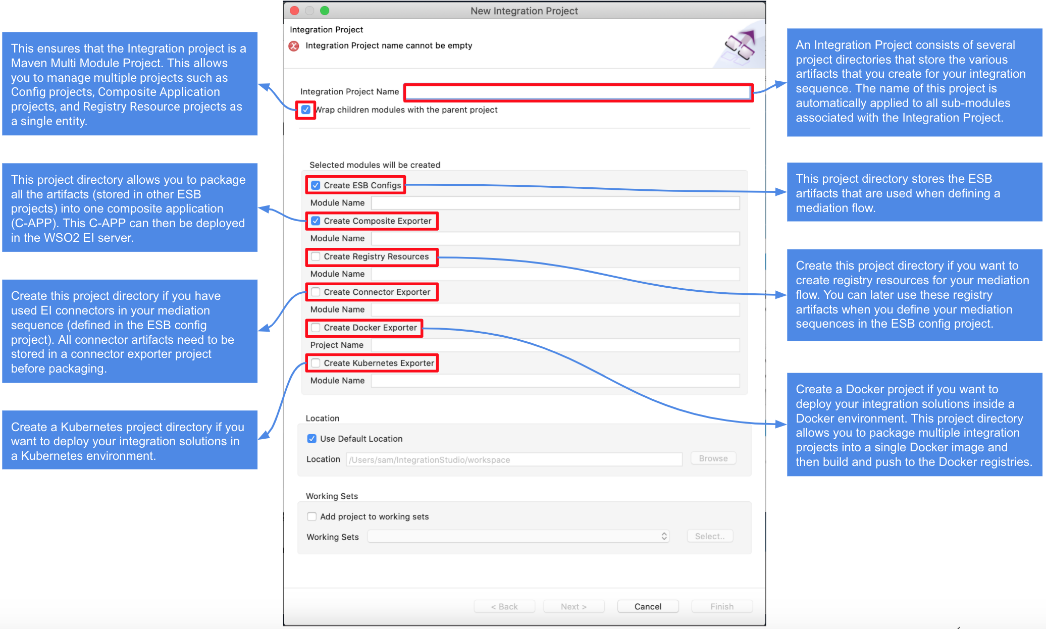
1. ابتدا WSO2 را دانلود و سپس نصب می‌کنیم.
2. سناریوی کلی پروژه (فرایند رزرو قرار ملاقات با پزشک/ بررسی مریض با پزشک‌های دو بیمارستان ...)



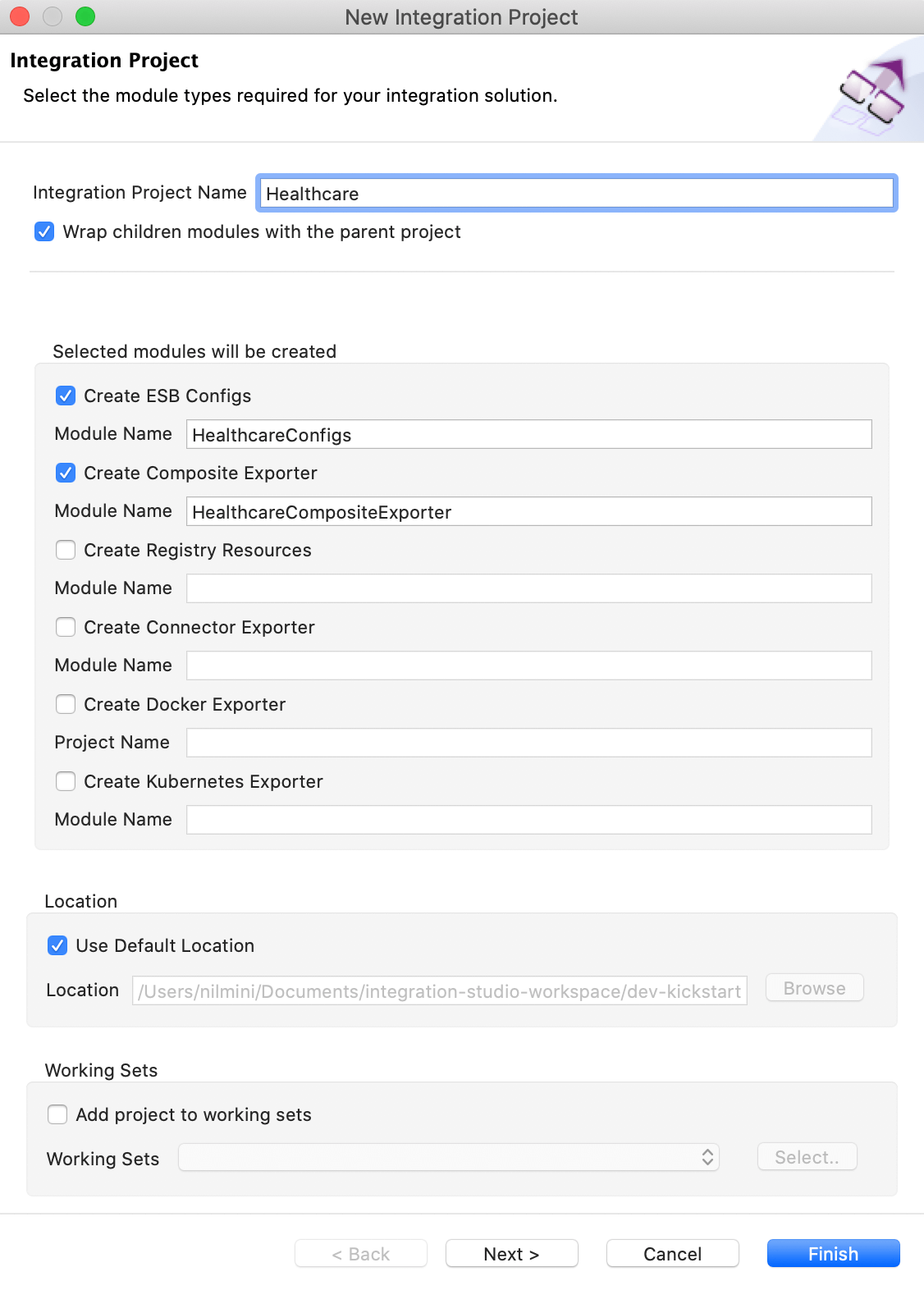
1. یک محل را برای workspace در نظر می‌گیریم.
2. بعد از باز کردن برنامه در محیط ابتدایی بر روی New integration Project در سمت چپ پنجره کلیک کنید.



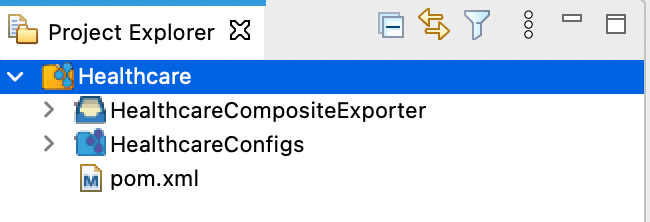
1. در کادر New Integration Project که باز می‌شود، نامی برای پروژه ادغام خود وارد کنید. اگر می‌خواهید علاوه بر تنظیمات ESB و صادرکننده ترکیبی، منابع رجیستری و ... کادرهای انتخاب مربوطه را انتخاب کنید.



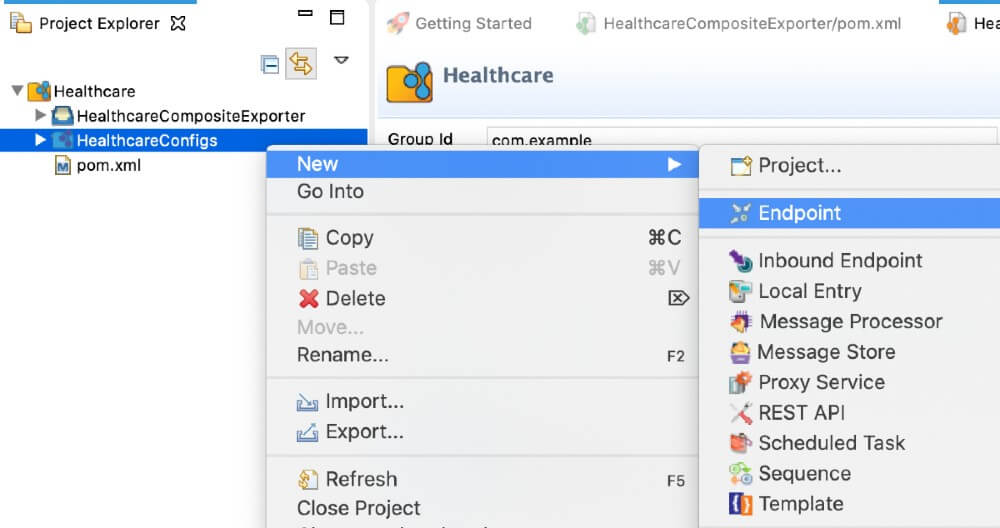
با توجه پروژه کادر را به‌صورت زیر تکمیل و بر روی Finish می‌زنیم

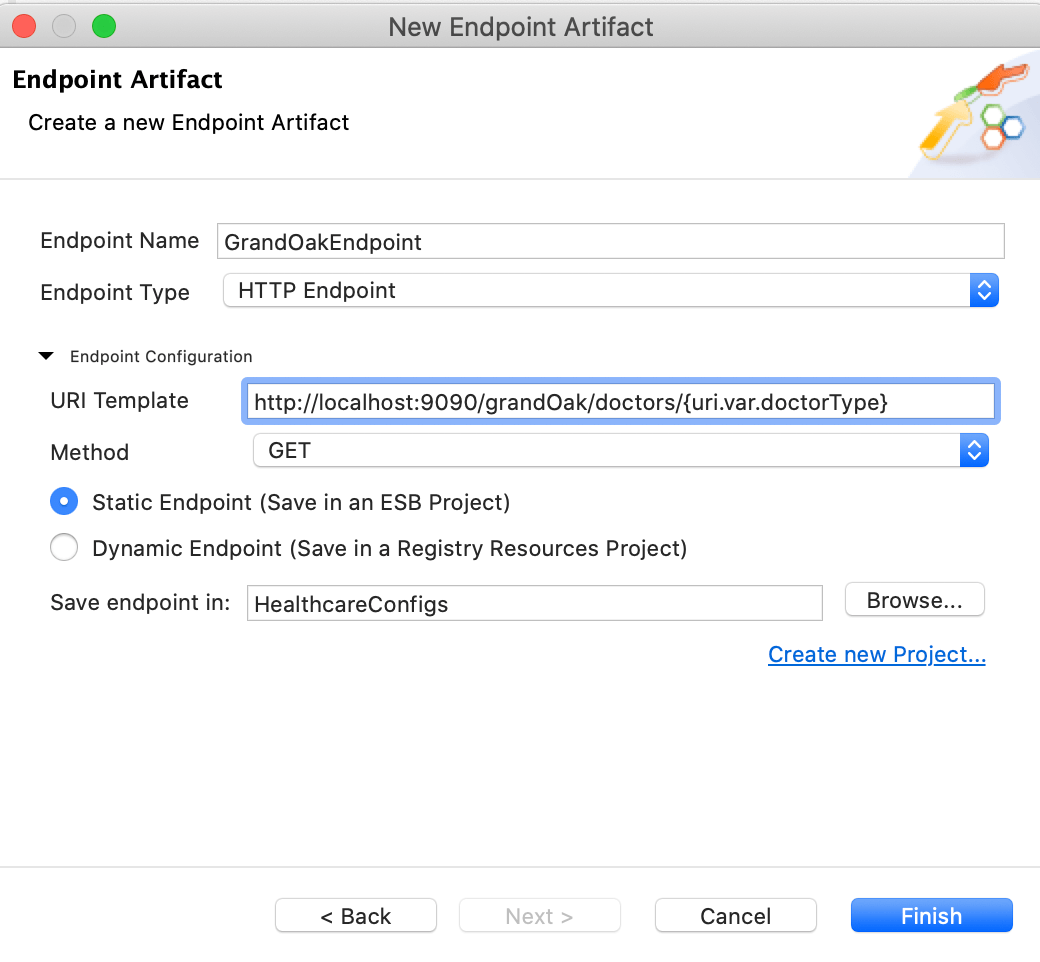


پس از بالا آمدن پروژه، پوشه بندی پروژه به شکل زیر قابل مشاهده است



1. با کلیک روی HealthcareConfigs و انتخاب Endpoint از New برای دو بیمارستان اندپوینت را به‌صورت زیر می‌سازیم

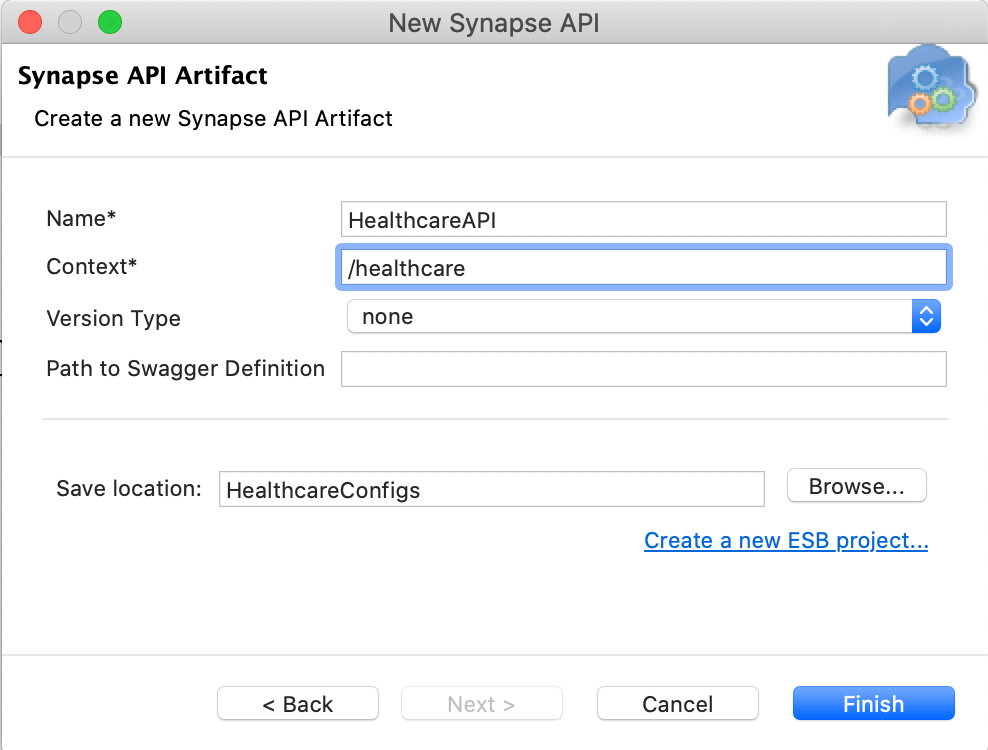




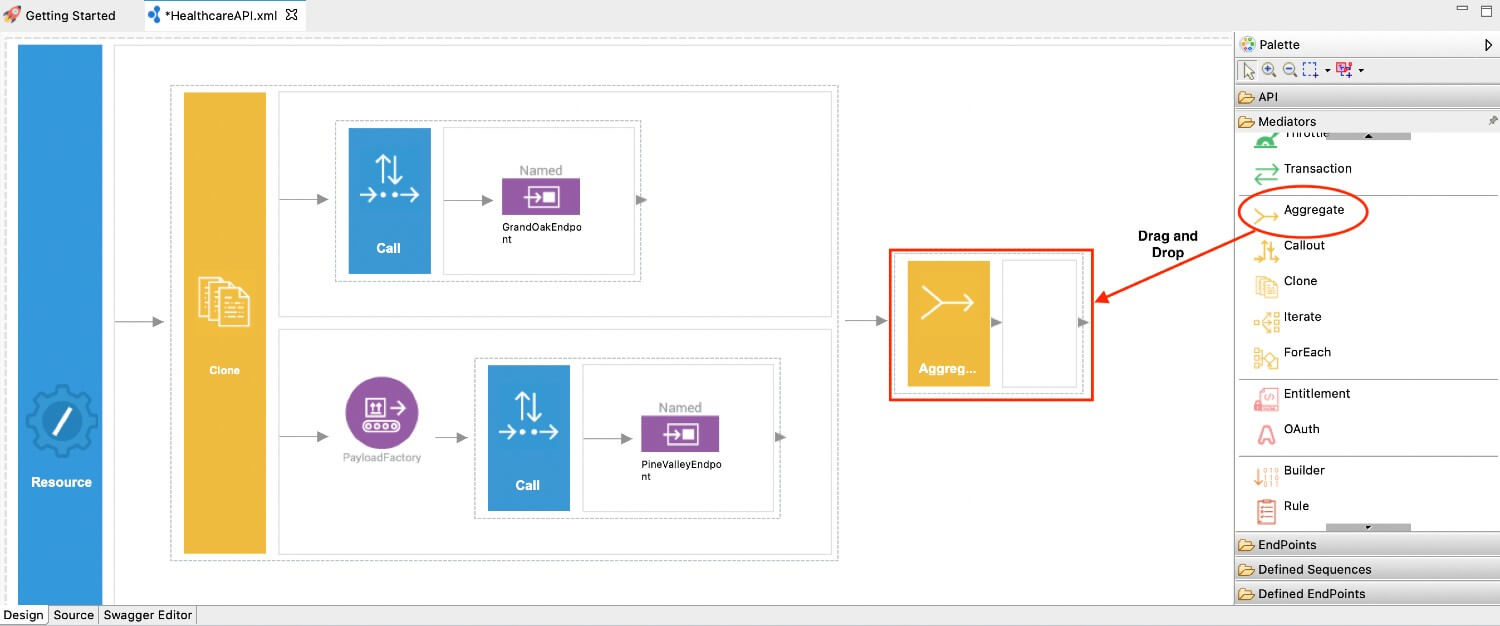
و برای بیمارستان دوم، PineValleyEndpoint برای Endpoint Name، در قسمت URI Template مقدار http://localhost:9091/pineValley/doctors و POST برای Method وارد می‌کنیم.

1. چند سرویس را تنظیم و برای برقراری ارتباط یک واحد API را در اختیار کاربران به ایجاد یک REST API می‌پردازیم که برای این منظور از تب New بر روی REST API می‌زنیم و از بین گزینه‌های پیشرو Create A New API Artifact را انتخاب و بر روی Next می‌زنیم.

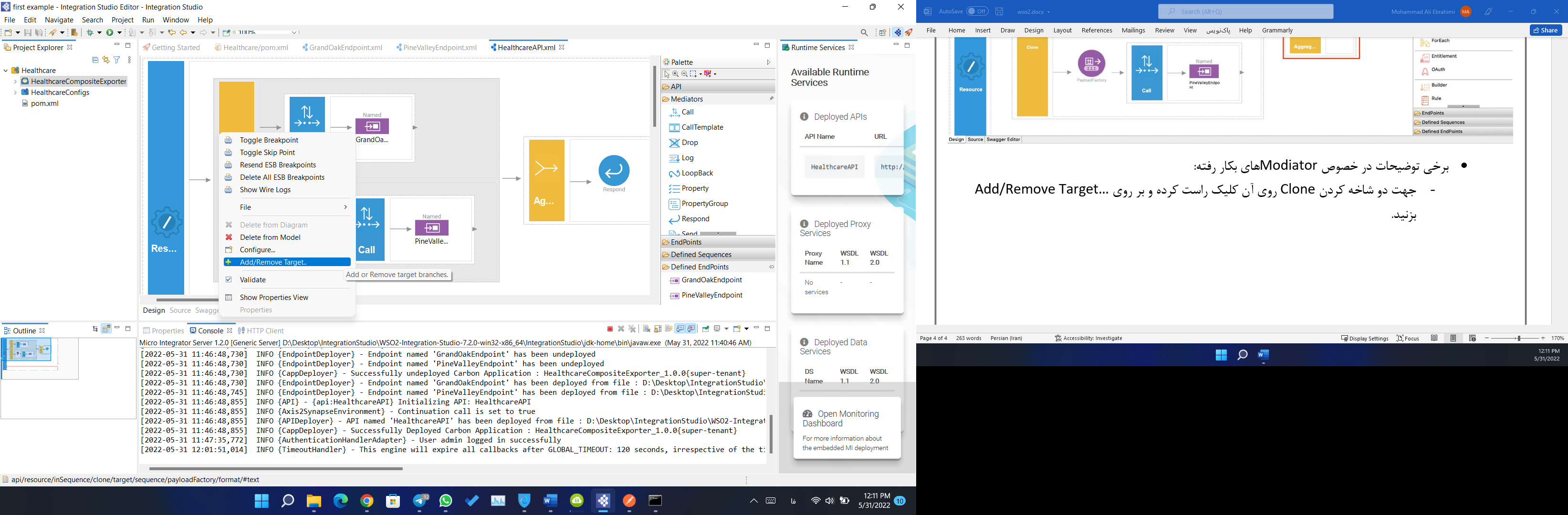
سپس در پنجره برای خصوصیات خواسته شده مطابق شکل عمل می‌کنیم و Finish را می‌زنیم.



1. بر روی HealthcareAPI.xml و از Mediators داخل Palette شکل زیر را ایجاد می‌کنیم.



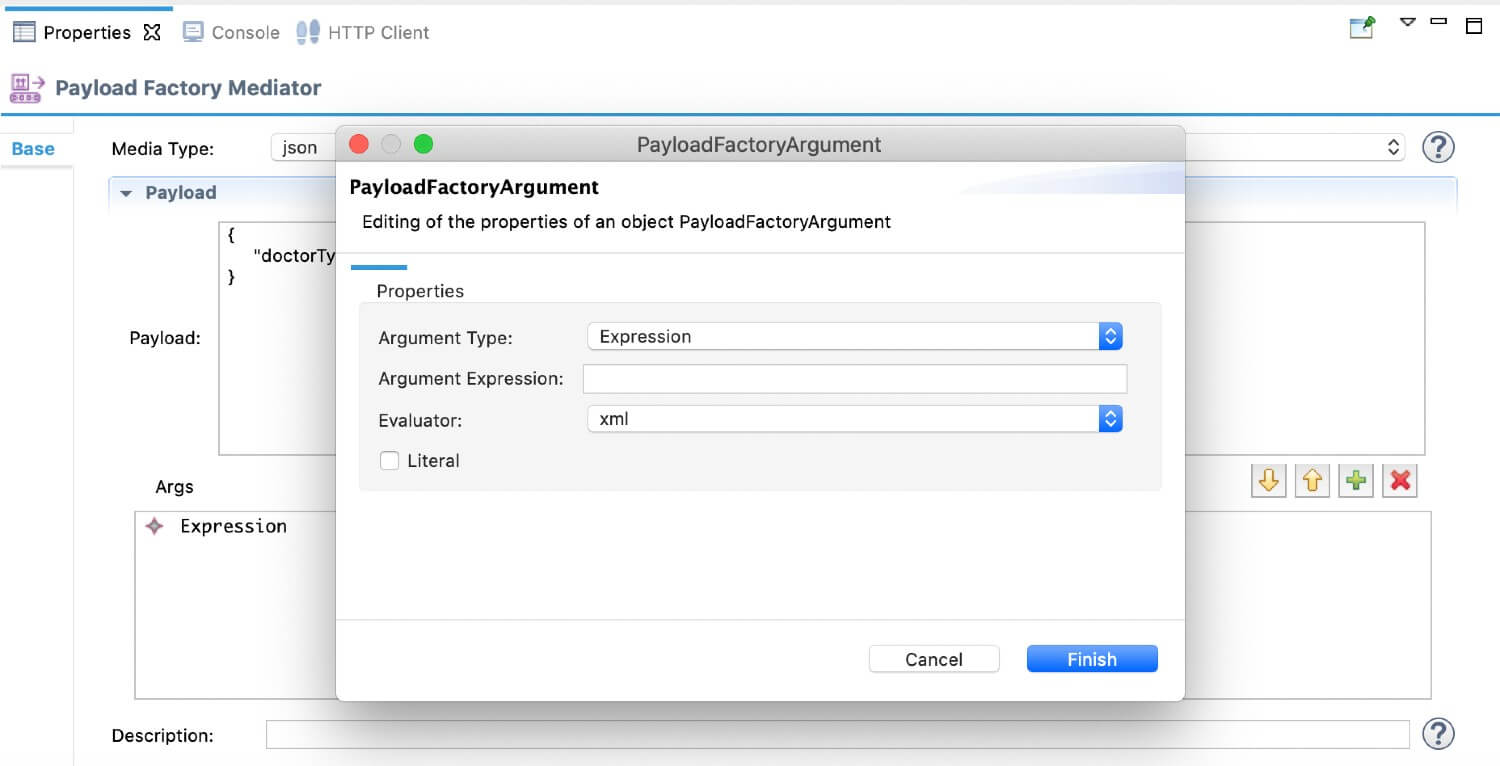
* برخی توضیحات در خصوص Modiatorهای بکار رفته:
* جهت دو شاخه کردن Clone روی آن کلیک راست کرده و بر روی Add/Remove Target… بزنید.



* Endpointها را از بخش Defind Sequences انتخاب می‌کنیم.
* برای مشخصات PayloadFactory پس از ایجاد موارد زیر را اعمال می‌کنیم.

|  |  |
| --- | --- |
| Parameter | Value |
| Payload Format | Inline |
| Media Type | json |
| Payload | { "doctorType": "$1" } |
| Args | $ctx:uri.var.doctorType |

برای واردکردن Args در بالای کادر Args روی ➕ کلیک کرده و در پنجرة باز شده مقدار Argument Type را Expression قرار داده و با کلیک روی فضای خالی مقابل Argument Expression مقدار Args موجود در جدول بالا را در کادر خالی مقابل XML می‌نویسیم.



* برای مشخصات **Aggregate** پس از ایجاد از json-eval($.doctors.doctor) برای مقدار Aggregation Expression استفاده می‌کنیم.

نهایتاً با رفتن به تب **Source View به کد زیر مواجه می‌شویم:**

<?xml version="1.0" encoding="UTF-8"?>

<api context="/healthcare" name="HealthcareAPI" xmlns="http://ws.apache.org/ns/synapse">

<resource methods="GET" uri-template="/doctor/{doctorType}">

<inSequence>

<clone>

<target>

<sequence>

<call>

<endpoint key="GrandOakEndpoint"/>

</call>

</sequence>

</target>

<target>

<sequence>

<payloadFactory media-type="json">

<format>{

"doctorType": "$1"

}

</format>

<args>

<arg evaluator="xml" expression="$ctx:uri.var.doctorType"/>

</args>

</payloadFactory>

<call>

<endpoint key="PineValleyEndpoint"/>

</call>

</sequence>

</target>

</clone>

<aggregate>

<completeCondition>

<messageCount max="-1" min="-1"/>

</completeCondition>

<onComplete expression="json-eval($.doctors.doctor)">

<respond/>

</onComplete>

</aggregate>

</inSequence>

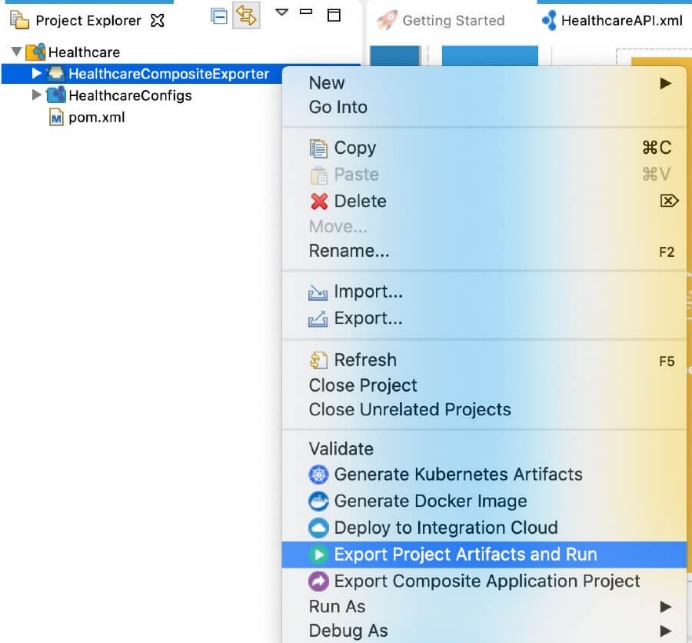
<outSequence/>

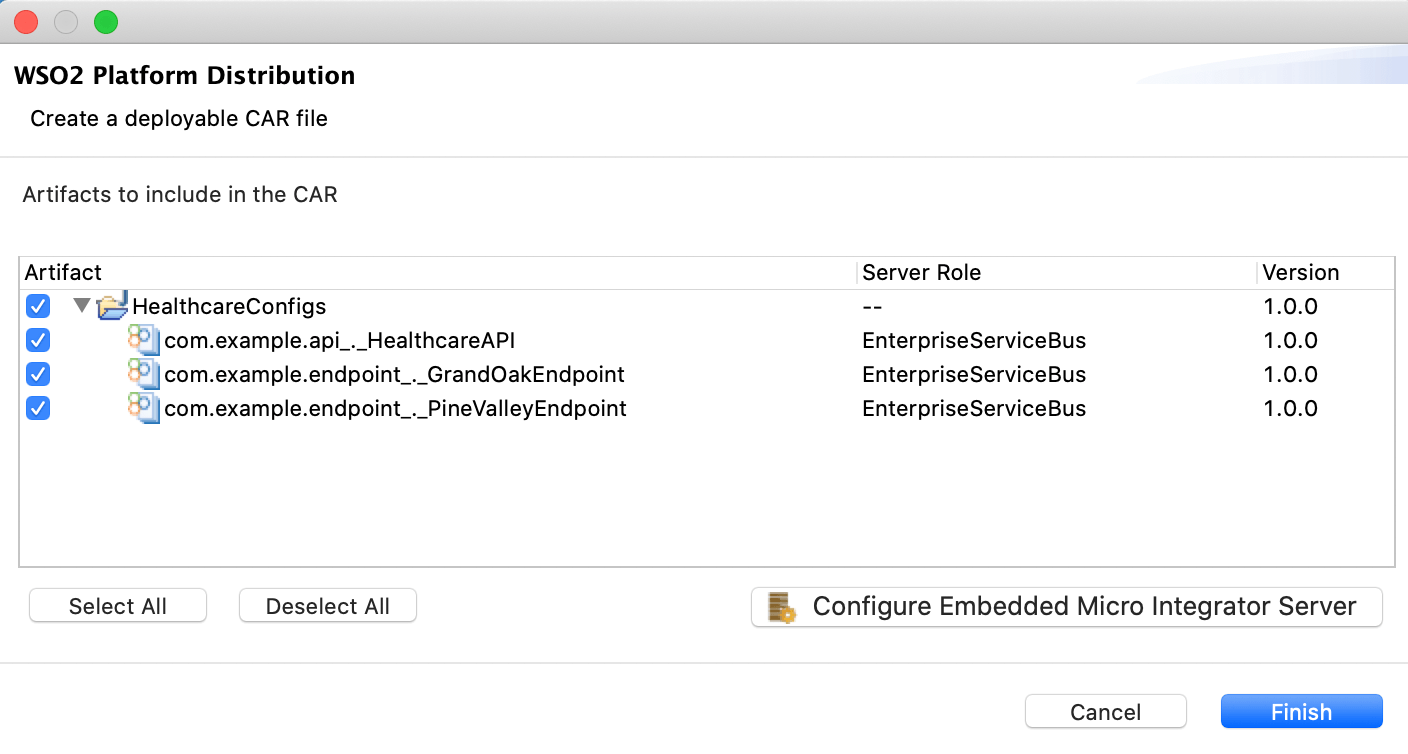
<faultSequence/>

</resource>

</api>

1. پس از مراحل فوق برای ساختن و خروجی گرفتن روی HealthcareCompositeExporter کلیک راست کرده و روی **Export Project Artifacts and Run کلیک می‌کنیم و در پنجرة باز شده** HealthcareConfigs را **انتخاب و بر روی Finish می‌زنیم.**





1. WSO2 Micro Integrator را از https://wso2.com/integration/micro-integrator دانلود می‌کنیم در همان صفحه Ml Dashboard را نیز دانلود می‌کنیم. پس از اتمام دانلود در هر دو فایل اکسترکت شده cmd را در داخل پوشه bin آنها اجرا و فایل‌های micro-integrator.bat و dashboard.bat را اجرا می‌کنیم.
2. برای دیتاهای برنامه از لینک https://apim.docs.wso2.com/en/4.1.0/assets/attachments/developing-first-integration/doctorinfo-jdk11.jar فایل را دانلود و برای فعال‌کردن از خط java -jar DoctorInfo-JDK11.jar در cmd بهره می‌بریم.
3. اینک برای نمایش نهایی پس از نصب Postman(https://www.postman.com/downloads) و اجراکردن آن، روی Create a collection زده سپس برای ریکوئست Add a new request را انتخاب می‌کنیم. سپس با انتخاب متود GET در بخش URL مقدار http://localhost:8290/healthcare/doctor/Ophthalmologist می‌نویسیم. مشاهد می‌کنیم پس از زدن دکمة send در خروجی برنامه سه دکتر مشاهده می‌شود.

[

[

{

"name": "John Mathew",

"time": "03:30 PM",

"hospital": "Grand Oak"

},

{

"name": "Allan Silvester",

"time": "04:30 PM",

"hospital": "Grand Oak"

}

],

[

{

"name": "John Mathew",

"time": "07:30 AM",

"hospital": "pineValley"

},

{

"name": "Roma Katherine",

"time": "04:30 PM",

"hospital": "pineValley"

}

]

]

* البته می‌توان همین خروجی را در cmd نیز مشاهده کرد. برای این کار خط کد زیر را در cmd بنویسید.

curl -v http://localhost:8290/healthcare/doctor/Ophthalmologist

|  |
| --- |
| \* در این پروژه همان‌طور که دیدیم یکی از بیمارستان‌ها از متود GET و دیگری از POST پیروی می‌کرد که برای اینکه بتوان از داده‌های هر دو بیمارستان بهره ببریم از PayloadFactoryArgument استفاده کردیم که {"doctorType": "$1"} برای تبدیل POST به GET قبل از صداکردن بیمارستان PineValley اجرا می‌شود. |