case study

This case study is for integration design, implementation and testing.

Core problem:

We have two separate systems that manage products:

System A) Master data management system. It manages product master data such as product Id and Name. It has endpoint to **GET** products:

https://fa-ji-poc.azurewebsites.net/api/GetAllProducts?code=x3NJjl39-Kqe897vi5lf8rpNVerbBnR_CTq-t6DSRMeHAzFuc85GnQ%3D%3D

System B) Price management system. It manages product prices. IT has endpoint to **GET** product prices:

https://fa-ji-poc.azurewebsites.net/api/GetAllPrices?code=rSdFVmW80_PL9rg0mOFVJ9mk-B285FYctg_a4V3ab_TNAzFuPAw6ZA%3D%3D
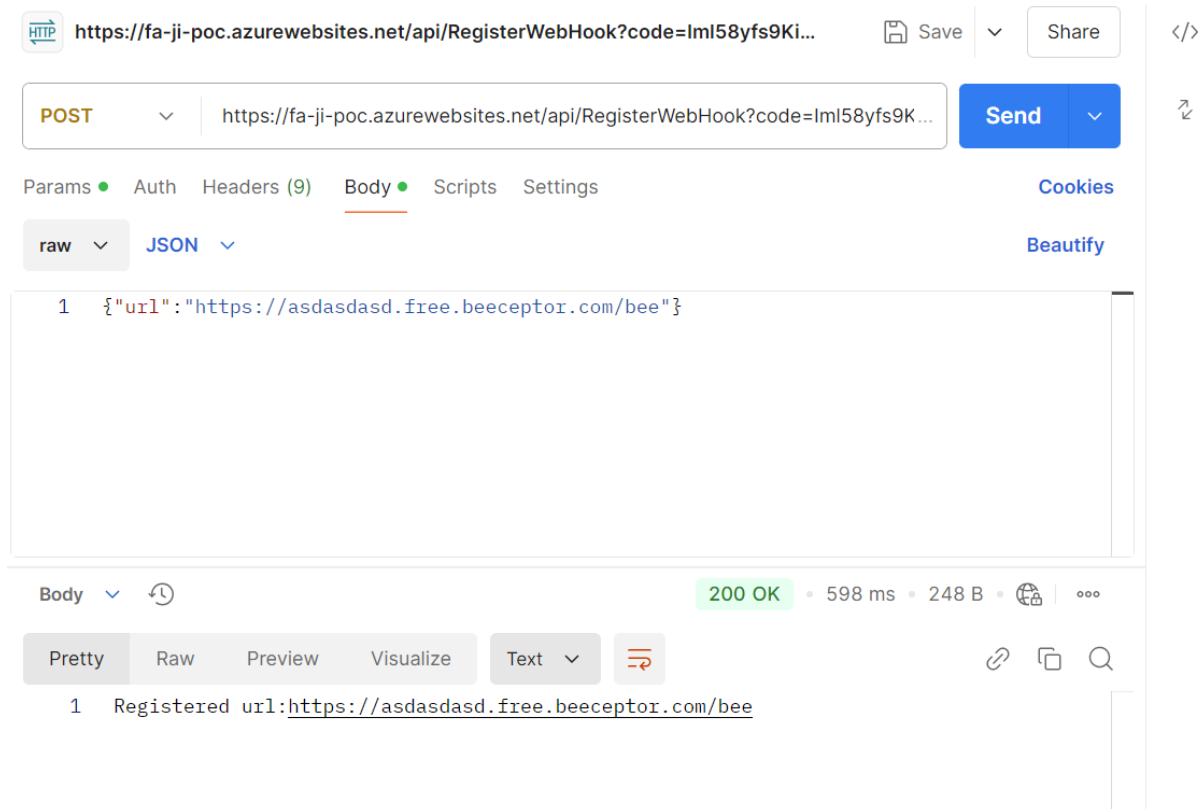
This system can inform other systems if there is change in product price. You can register your **POST** webhook using endpoint:

 https://fa-ji-poc.azurewebsites.net/api/RegisterWebHook?code=Iml58yfs9KihjcKA_J_GBBuOQWLbV1AXLOoiMvb9wx_MAzFuVbk5_A%3D%3D

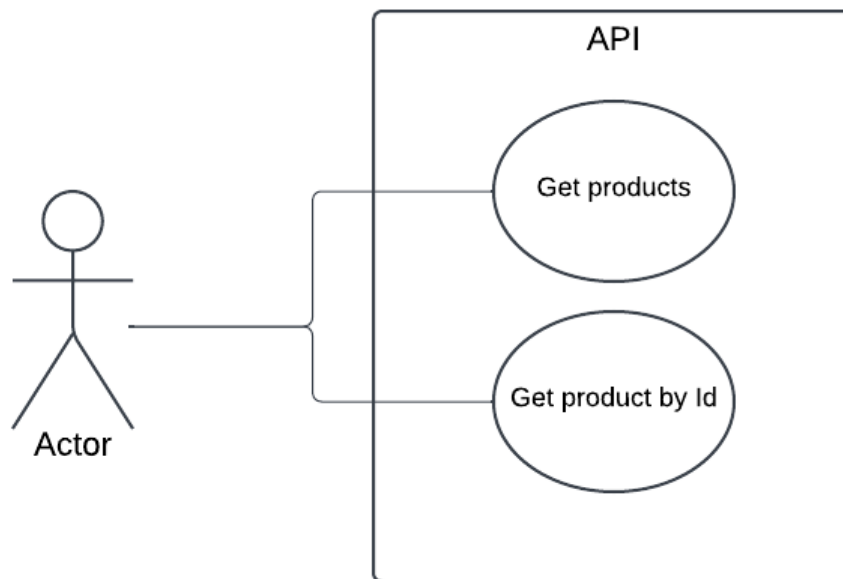Where body is like

Body: {"url":"{your web hook url}"}

F.e.:

System B will send updates to webhook url f.e.

[{"productId":"abj3","productPrice":40},{"productId":"new0","productPrice":0},{"productId":"new1","productPrice":100},{"productId":"new2","productPrice":200},{"productId":"new3","productPrice":300},{"productId":"new4","productPrice":400},{"productId":"new5","productPrice":500},{"productId":"new6","productPrice":600}]

Functional requirements:

We need API that could give data with both product name, id and price.

Non functional requirements:

1) System A and System B should get least amount of request load
2) API Data should not be more than 15min old
3) API should be able to respond in less than 800ms in 10req/s load

Case study deliverables:

1) Minimalistic architecture (System diagram, design choices)
2) Azure function, Web API or .net Aspire or other platform solution source code
3) Unit tests
4) OpenAPI description of APIs

Helpers:

You can use our cosmos DB. Connection string will be sent separately.

No need of deployment – you can run code on your own computer for presentation.

Extra:

1) If task is too easy you can design solution that could adapt to changes of System A, for example if System A could in addition send **weight** without notifying about new field. integration could save that and display in APIs. Design openAPI for such extension points.
2) If product changes – send event to ServiceBus queue. Connection string will be sent separately.