## **Basic Prerequisites for Project:**

1. [**Document DB**](https://aka.ms/documentdb-emulator)
2. [Azure SDK](https://www.microsoft.com/web/handlers/webpi.ashx/getinstaller/VWDOrVs2015AzurePack.appids)

## **How to see it running:**

1. Code can be downloaded from Git Repo: <https://github.com/shaikhwajeed/GloboMart>
2. It has two solutions:
   1. GloboMart.sln (Service Part)
   2. GloboMartClient.sln (Demo client part)
3. Make sure that document DB is running. (You could see it in system tray)
4. Run Service Project first. It’s configured to run both the Service Projects. (Each will be eventually deployed separately to cloud)
5. Run the Client Demo Solution.
6. You could Try Create New, Delete functionality.
7. Unit Test for service part can be found in Respective Test project.

## **Implementation Basics:**

1. Product.Catalogue
   1. Web API project which handles product catalogues.
   2. Provides, Add, Delete, GetAll functionality
   3. It interacts with ***pricing service*** to delete price information whenever product is getting deleted from system.
2. Product.Pricing
   1. Provides API to Add,Delete,Get pricing details about particular product.
3. GloboMart.WebClient
   1. Web MVC Client which is used to show interaction with catalogue and pricing services.

## **Assumptions:**

1. Product.Catalogue and Product.Pricing will be deployed as separate web services in Azure.
2. There will be no sync between databases of Product Catalogue and Pricing.
3. Even if pricing service is down, only product catalog information can be served to user.

## **Risks:**

1. In this example we have just considered two services, it’s manageable right now. However if we have large number of microservices, then it might be difficult to manage. We need to have strong failover mechanism and monitoring system.
2. It’s little difficult to understand the flow of system as multiple services are involved.