# Özet

Sipariş işlemleri için API geliştirmesi yapılacaktır. Uygulamada kullanılacak alt yapı, araçlar ve iş akışı aşağıdaki gibi olmalıdır.

# Teknolojiler

**Framework:** .Net 6

**Dil:** C#

**Database:** Mysql

**ORM**: Entity Framework – Code First Approach

**Memory Cache**: .Net Memory Cache veya Redis (vakit bulursanız Redis tercih sebebi)

**Message Queue**: RabbitMQ

**Araçlar:** AutoMapper, SeriLog

# İş Akışı

Apiyi kullanacak platform önce API’dan ürünleri listeleyecektir. Listelenen ürünlerden bir veya birden fazlası için sipariş oluşturma API metodunu çağırarak sipariş oluşturabilecektir.

# Entity ve Modeller

1. **Product Entity**

Id, Description, Category, Unit, UnitPrice, Status, CreateDate, UpdateDate alanlarından oluşacaktır. Alan tiplerini ve boyutlarını siz tespit ediniz. Test için 1000 adet random veri ile doldurabilirsiniz.

1. **ProductDto**

Id, Description,Category,Unit,UnitPrice

1. **Order Entity**

Id, CustomerName, CustomerEmail, CustomerGSM, TotalAmount alanlarından oluşacaktır. Alan tiplerini ve boyutlarını siz tespit ediniz

1. **OrderDetail Entity**

Id, OrderId, ProductId, UnitPrice,

1. **CreateOrderRequest**

CustomerName, CustomerEmail, CustomerGSM, List<ProductDetail>. ProductDetail objesi: ProductId,UnitPrice,Amount

1. **ApiResponse**

Status (Success,Failed enum) , ResultMessage,ErrorCode,Data (GenericType)

# Api Metodları

Api metotları ApiResponse tipinde dönüş yapacaktır.

1. **GetProducts**

Http metodu GET olacaktır. Category url parametresi alabilecektir. Category parametresi boş gelirse tüm ürünleri, dolu ise ilgili kategoriye ait ürünler listelenecektir. Methoda istek yapıldığın ilk olarak memory cache’den bakılacak ve doluysa buradan ürünler verilecektir. Cache boşsa databaseden kayıtlar okunacak ve Memory cache doldurulacaktır ve ürünler listelenecektir. Response Data alanında List<ProductDto> dönülecektir.

1. **CreateOrder**

Http metodu POST olacaktır. Parametre olarak CreateOrderRequest modelini alacaktır. CreateOrderRequest objesinden faydalanılarak Order ve OrderDetail tablolarına insert işlemi yapılacaktır. Satış yapılan müşteriye mail gönderimi asenkron olarak gerçekleştirilmesi için RabitMq servisindeki SendMail kuyruğuna eklenecektir. Response Data alanında oluşan order kaydının Id bilgisi dönülecektir.

# Background Servis

Mail gönderimlerini asenkron olarak yapabilmek için CreateOrder metodundan SendMail kuyruğuna istekler yazılmıştır. MailSenderBackgroundService adında HostedService geliştirilmelidir. Servis ayağa kalkarken SendMail kuyruğunu dinlemeye başlayacaktır ve kuyruğa eklenen mail gönderim isteklerini işleyecektir.