

Final Proje - Python Kullanarak Ürün Fiyat Karşılaştırma Uygulaması



Gerekli Tahmini Süre: 1.5 saat

- Bu final projede, entegre bir uygulama oluşturmak için birden fazla mikro hizmeti dağıtacaksınız. Bu bileşenler, biri Python'da diğeri Node.js'de geliştirilmiş iki arka uç mikro hizmeti ve bir ön uç mikro hizmeti içermektedir.

Hedefler:

Bu laboratuvarı tamamladıktan sonra şunları yapabileceksiniz:

- IBM Cloud Code Engine üzerinde Python ve Node.js arka uç mikro hizmetlerini dağıtmak.
- IBM Cloud Code Engine üzerinde ön uç mikro hizmetlerini dağıtmak.

Bölüm A: Arka Uç Mikrosunucularını Dağıtma

- Code Engine CLI'yi açın.
- Ürün bilgilerini almak için API uç noktaları sağlayan Ürün Detayları mikrosunucusunu dağıtın.

build-source - https://github.com/ibm-developer-skills-network/dealer_evaluation_backend.git

build-context-dir - products_list

port - 5000

```
ibmcloud ce application create --name prodlist --image us.icr.io/${SN_ICR_NAMESPACE}/prodlist --registry-secret icr-secret --port 5000 --build-context-dir products_list --build-source https://github.com/ibm-de
```

Dağıtım URL'sini kopyalayın ve bir not defteri veya başka bir metin düzenleyicisinde kaydedin.

Başarılı dağıtımın ekran görüntüsünü alın ve product_details_deploy.png olarak kaydedin.

```
theia@theiadocker-lavanyas:/home/project$ ibmcloud ce application create --name prodlist
--image us.icr.io/${SN_ICR_NAMESPACE}/prodlist --registry-secret icr-secret --port 5000
--build-context-dir products_list --build-source https://github.com/ibm-developer-skill
s-network/dealer_evaluation_backend.git
Creating application 'prodlist'...
Submitting build run 'prodlist-run-230110-074851157'...
Creating image 'us.icr.io/sn-labs-lavanyas/prodlist:230110-1248-j11hw'...
Waiting for build run to complete...
Build run status: 'Running'
Build run completed successfully.
Run 'ibmcloud ce buildrun get -n prodlist-run-230110-074851157' to check the build run s
tatus.
Waiting for application 'prodlist' to become ready.
Configuration 'prodlist' is waiting for a Revision to become ready.
Ingress has not yet been reconciled.
Waiting for load balancer to be ready.
Run 'ibmcloud ce application get -n prodlist' to check the application status.
OK

https://prodlist.xj562h09ws5.us-south.codeengine.appdomain.cloud
```

Lütfen FAILED Wait failed for application 'prodlist' hatasıyla karşılaşsanız, uygulamanın adını prodlist1 olarak değiştirip komutu yeniden çalıştırabileceğinizi unutmayın.

- Bayi Fiyatlandırması için mikroservisi dağıtın; bu, bayi fiyat bilgilerini almak için API uç noktaları sağlar.

Not: Lütfen dağıtım komutu için aşağıdaki parametreleri kullanın.

build-source - https://github.com/ibm-developer-skills-network/dealer_evaluation_backend.git

build-context-dir - dealer_details

port - 8080

name - dealerdetails

image - us.icr.io/\${SN_ICR_NAMESPACE}/dealerdetails

Dağıtım URL'sini kopyalayın ve bir not defteri veya başka bir metin düzenleyicisinde kaydedin.

Başarılı dağıtımın ekran görüntüsünü alın ve dealer_details_deploy.png olarak kaydedin.

```
theia@theiadocker-lavanyas:/home/project$ ibmcloud ce application create --name dealerde
tails --image us.icr.io/${SN_ICR_NAMESPACE}/dealerdetails --registry-secret icr-secret --p
ort 8080 --build-context-dir dealer_details --build-source https://github.com/ibm-develo
per-skills-network/dealer_evaluation_backend.git
Creating application 'dealerdetails'...
Submitting build run 'dealerdetails-run-230110-074851157'...
Creating image 'us.icr.io/sn-labs-lavanyas/dealerdetails:230110-1248-j11hw'...
Waiting for build run to complete...
Build run status: 'Running'
Build run completed successfully.
Run 'ibmcloud ce buildrun get -n dealerdetails-run-230110-074851157' to check the build r
un status.
Waiting for application 'dealerdetails' to become ready.
Configuration 'dealerdetails' is waiting for a Revision to become ready.
Ingress has not yet been reconciled.
Waiting for load balancer to be ready.
Run 'ibmcloud ce application get -n dealerdetails' to check the application status.
OK

https://dealerdetails.xj562h09ws5.us-south.codeengine.appdomain.cloud
```

Lütfen FAILED Wait failed for application 'dealerdetails' hatasıyla karşılaşsanız, uygulamanın adını dealerdetails1 olarak değiştirip komutu yeniden çalıştırabileceğinizi unutmayın.

Bölüm B: Frontend Mikroservisini Dağıtma

- Yeni bir terminal açın, /home/project dizinine gidin.

```
cd /home/project
```

2. Depoyu https://github.com/ibm-developer-skills-network/dealer_evaluation_frontend.git adresinden /home/project dizinine klonlayın.

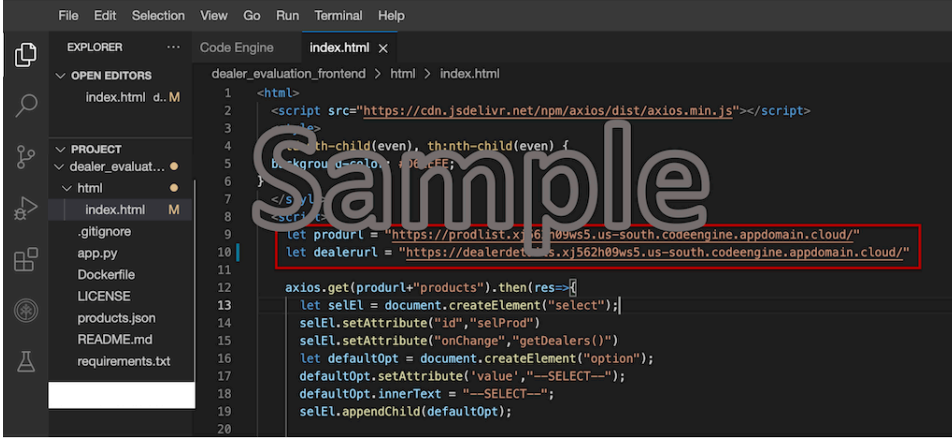
Başarılı bir git klonlama ekran görüntüsü alın ve git_clone.png olarak kaydedin.

```
theia@theiadocker-lavanyas:/home/project$ git clone https://github.com/ibm-developer-skills-network/dealer_evaluation_frontend.git
Cloning into 'dealer_evaluation_frontend'...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (12/12), done.
remote: Total 16 (delta 2), reused 14 (delta 0), pack reused 0
Unpacking objects: 100% (16/16), done.
theia@theiadocker-lavanyas:/home/project$ cd dealer_evaluation_frontend/
```

3. dealer_evaluation_frontend dizinine geçin.

4. index.html dosyasını mikroservis dağıtımlarından elde edilen dağıtım URL'leri ile güncelleyin. (http://localhost:5000/ ve http://localhost:8080/), uygun yere kopyaladığınız dağıtım URL'lerini yerleştirin. URL'lerinizi / ile bitirdiğinizden emin olun.

Değişikliklerin ekran görüntüsünü alın ve index_urlchanges.png olarak kaydedin.



5. Dealer Evaluation ön yüz mikroservisini mevcut dizine işaret ederek dağıtın.

build-source - .

port - 5001

name - frontend

image - us.icr.io/\${SN_ICR_NAMESPACE}/frontend

Başarılı dağıtımın ekran görüntüsünü alın ve frontend_deploy.png olarak adlandırın.

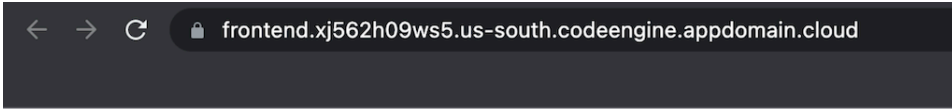
```
theia@theiadocker-lavanyas:/home/project/dealer_evaluation_frontend$ ibmcloud ce application create --name frontend --image us.icr.io/${SN_ICR_NAMESPACE}/frontend --registry-secret icr-secret --port 5001 --build-source .
Creating application 'frontend'...
Packaging files to upload from source path '.'...
Submitting build run 'frontend-run-230110-080653852'...
Creating image 'us.icr.io/${SN_ICR_NAMESPACE}/frontend:230110-1306-lccc'...
Waiting for build run to complete...
Build run status: 'running'
Build run completed successfully.
Run 'ibmcloud ce build-run get -n frontend-run-230110-080653852' to check the build run status.
Waiting for application 'frontend' to become ready.
Configuration 'frontend' is waiting for a Revision to become ready.
Ingress has not yet been recorded.
Waiting for load balancer to be ready.
Run 'ibmcloud ce application get -n frontend' to check the application status.
OK
https://frontend.xj562h09ws5.us-south.codeengine.appdomain.cloud
```

FAILED wait failed for application 'frontend' hatasıyla karşılaşırsanız, uygulamanın adını frontend1 olarak değiştirip komutu yeniden çalıştırabilirsiniz.

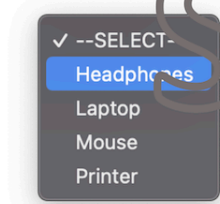
6. Ana sayfayı yüklemek için bağlantıya tıklayın. Lütfen sayfanın ilk kez eriştiğinizde yüklenmesinin zaman alacağını unutmayın.

7. Ürünlerin doldurulup doldurulmadığını görmek için ürünler açılır menüsüne tıklayın.

Ürün listesini gösteren ana sayfanın ekran görüntüsünü alın ve homepage.png olarak adlandırın.



Products price comparison



8. Ürün için belirli bir bayiyi seçin ve fiyatın görüntülendiğini doğrulayın.

Seçilen ürünü ve mikroservis tarafından döndürülen listelenen ürünü sağlayan bayileri gösteren tüm sayfanın ekran görüntüsünü alın ve product_dealer.png olarak adlandırın.

Products price comparison

Headphones ▾

✓ --SELECT--

Binglee

DXC Electronics

Bobay

All Dealers

9. Bayiler açılır menüsü doldurulduktan sonra, ürün için belirli bir bayiyi seçin ve o bayinin talep ettiği fiyatın görüntülendiğini kontrol edin.

Sayfanın yüklenmesi için 10 ila 20 saniye bekleyin.

Seçilen ürünü, seçilen bayiyi ve mikroservis tarafından döndürülen fiyatı gösteren tüm sayfanın ekran görüntüsünü alın ve product_dealer_price.png olarak adlandırın.

Products price comparison

Headphones ▾

Binglee ▾

Headphones costs \$30 at Binglee

10. Bir ürün için Tüm Bayiler seçeneğini seçin (birden fazla bayisi olan bir ürün seçtiğinizden emin olun). Ürünü sunan tüm bayilerin fiyatları ekranda gösterilmelidir.

Seçilen ürünü, Tüm Bayiler seçeneğini ve mikroservis tarafından döndürülen tüm bayilerin talep ettiği fiyatları gösteren tüm sayfanın ekran görüntüsünü alın ve product_all_dealers_prices.png olarak adlandırın.

Products price comparison

Headphones ▾

All Dealers ▾

Binglee	\$30
DXC Electronics	\$20
Bobay	\$20

Tebrikler! Final projesini tamamladınız!

Özet:

Bu laboratuvar çalışmasında, entegre bir uygulama oluşturmak için birden fazla mikroservisi başarıyla dağıttınız. Bu, Python'da geliştirilen iki arka uç mikroservisi ve bir ön uç mikroservisini içerir.

Author(s)

Lavanaya T S

© IBM Corporation. Tüm hakları saklıdır.