

Interior Design App

Research

Case'imizin özetini kullanıcı tarafından verilen oda resmi ve istenilen stil inputlarıyla yapılacak bir işlem sonrasında oda resmini değiştirip, stili oda resmine entegre edeceğiz. Bunun yapılması için en olası method (Stable Diffusion+ControlNet) yapısından oluşan bir pipeline gibi duruyor.

Stable Diffusion modelleri öncesinde bu işlem GAN'lar ile style transfer başlığındaki yöntemlerle denenebilirdi. Fakat GAN'larda style transfer yöntemleri, Stable Diffusion çözümüne göre daha genelleştirilmiş çözümler sunduğu için bizim case'imizde daha kötü bir performans verirdi.

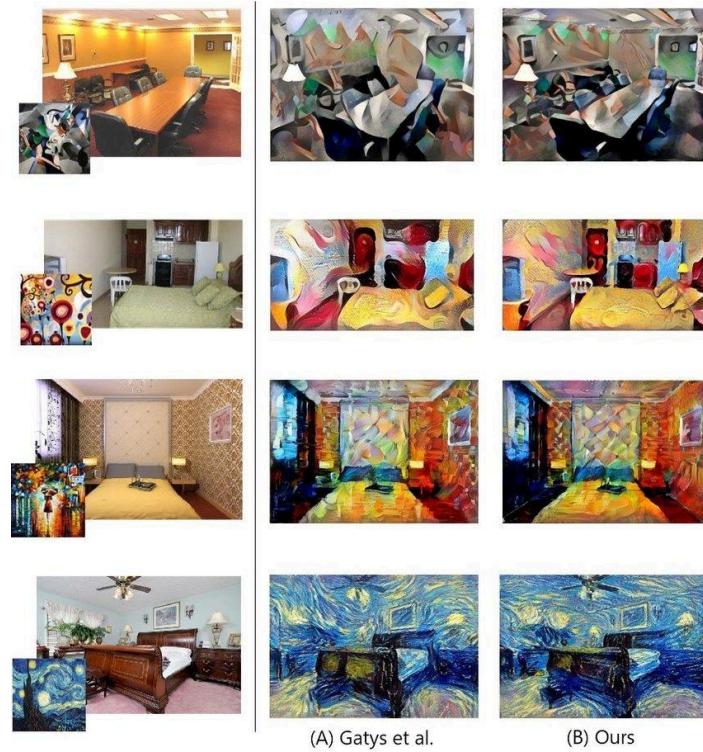


Figure 1. Ev odalarında style transfer

Bu yaklaşım ile Figure 1.'deki veya benzeri sonuçlar elde edilebilir. Daha sonra bu çıktılar Semantic Segmentation yapıları ile desteklenip sadece belirli eşyalarda değişiklikler sağlanabilir.

Gelelim bizim case'imiz için en uygun olduğunu düşündüğüm yapıya. Stable Diffusion'lar geçtiğimiz aylarda kaliteli sonuçlarıyla herkesin dikkatini geçen yeni bir yöntem. Yapısı gereği Stable Diffusion çıktılarını kontrol etmek zordur fakat üzerinde halen birçok çalışma yapılmaktadır. Buna ek olarak son yapılan çalışmalarda kısa süreli de olsa consistent video üretimine doğru bir ilerleme söz konusudur.

Stable Diffusion Nedir?

Stable Diffusion'lar latent tabanlı bir text2image diffusion modeli bütünüdür. Input olarak aldığı text'i CLIP yöntemi ile bir embeddinge çevirir. Ardından bu embedding ile noise'dan UNET ile optimize ederek bir VAE (Variational AutoEncoders) ile son çıktı haline getirir.

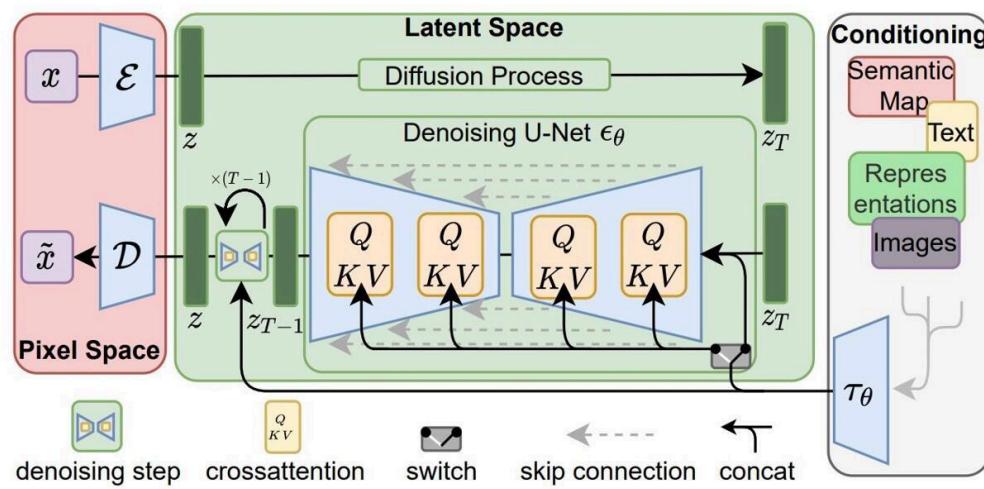


Figure 2. Stable Diffusion Pipeline

Figure 2.'de içeriğinde görüldüğü üzere içeriği Neural Network'ler ve çalışma prensibi bu şekildedir.

ControlNet Nedir?

ControlNet, standart Stable Diffusion modellerinin içerisinde olan layer'lara eklenti olarak kullanılan bir network'tür. Bu eklenti sayesinde hali hazırda weight'leri manipüle ederek çıktıyı ControlNet condition'una doğru manipüle eder ve standart çıktıyı belirli formlara sokabilir.

Daha önce GAN'lar da benzeri veya farklı yöntemlerle manipüle edilebiliyordu. ControlNet'in bu kadar popüler olmasının nedeni Stable Diffusion'larda kontrol edilebilir çıktılarına olanak sağlamasıdır.

ControlNet'in farklı yaklaşımalarla eğitilmiş modelleri mevcuttur. Bunlar input olarak aldığı conditionlar ile Stable Diffusion layer'larına entegre edilip çıktıyı görseli kendi condition formatında üretmesini sağlar.

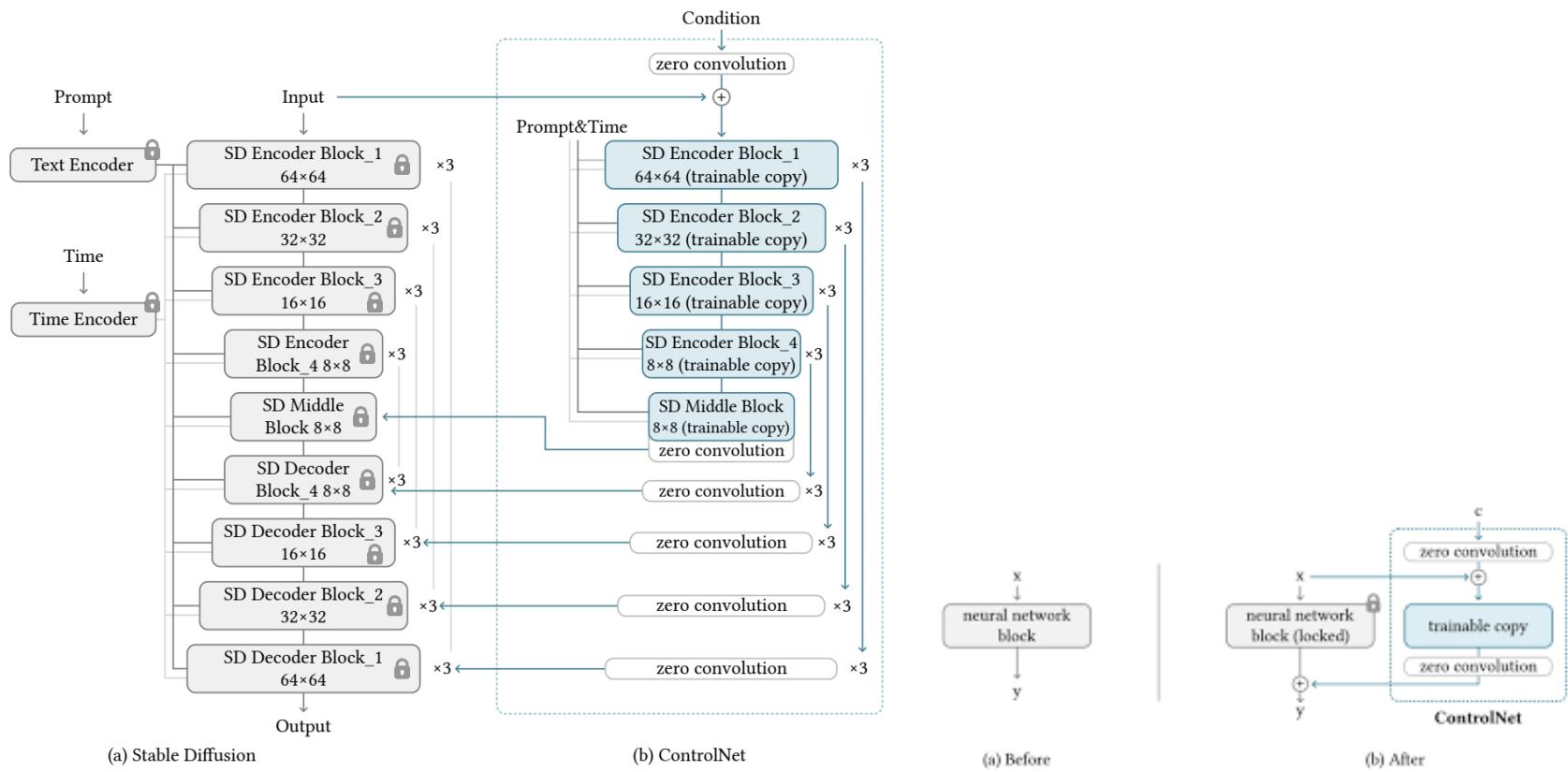


Figure 3. ControlNet Entegrasyon Haritası

Figure 3.'te görüldüğü üzere ControlNet'in kendisi bağımsız bir Neural Network'tür ve hali hazırda kullanılan Stable Diffusion'ın UNet layerlarına entegre ediliyor.

ControlNet'e input olarak verilen condition birçok farklı formatta kullanılıyor. Bunlara en popüler örnekler;

1. Canny edges
2. Depth estimation
3. HED Boundary
4. M-LSD straight line detection
5. Normal Map Estimation
6. Human Pose Estimation
7. Image Segmentation

bu şekildedir. Listedeki yöntemlerin tamamı bizim case'imize uygun değil. Bu nedenle sadece uygun olan yöntemler ile gerekli testler yapıldı.

ControlNet Yöntem Testi

Odalardaki dizaynı değiştirmek için kullanılabilen ControlNet yöntemleri canny edges, depth estimation, HED boundary, M-LSD straight line detection ve image segmentation olarak belirlendi. Kullanıcı tarafından gelebilecek olan oda görselleri çoğunlukla oturma odası, yatak odası ve mutfak olduğunu varsayıarak her oda tipinden birer örnek fotoğraf ile ControlNet yöntemleri test edilip karşılaştırma yapıldı.

Testler sonucunda variolan sonuçlar;

1. Segmentation maskelerinde karışıklık olduğu için objelerin yerleri aşırı değişip odada karışıklık yaşanıyor. Daha başarılı bir semantic segmentation network'ü kullanılarak bu sorun çözülebilir. Segmentation'un avantajı, ürün üzerinde sonrası için modifikasyon yapılması gereklse class'lar bilindiği için obje özelinde değişikliğe olanak sağlayabilir. (Örneğin değişiklik sadece masa, koltuk, tv, vs. yapılacak)
2. Objelerin yapısal olarak değişmesinde **depth estimation** diğer yöntemlere göre daha başarılı.
3. **HED** yöntemi hem yapısal olarak koruyup aynı zamanda güzel değişikliklere olanak tanıyor.

bu şekildedir. Eleme sonucu **HED** ve **Depth** arasında tercih yapılması gerekiğine karar verildi. Eğer ürün mobilyaların ve odadaki diğer objelerin yapısal olarak korunup renk ve desene olarak değişmesi için yapılacak ise **HED** fakat mobilyaların yapısal olarak değişik diğer objelerin duruma göre kaldırılıp farklı objelerin eklenme ihtiyalini göze alınıyor ise **Depth** daha mantıklı olacaktır. Bu linkten [ControlNet Test](#) tüm çıktılar tekrar incelenebilir. Buna ek olarak [HED vs Depth](#) bu linkten farklı stillerde HED ve Depth karşılaştırmasını tekrar incelenebilir. Görsel olarak daha gerçekçi ve eğlenceli değişiklikler olduğunu düşündüğüm için uygulamayı **depth estimation** üzerinden ilerlenecektir. Bu yorumu yapmama neden olan görsellerden birini **Figure 5.** ve **Figure 6.**'da inceleyebiliriz.



Figure 5. Depth Estimation Room Example (Egyptian)



Figure 6. HED Boundary Room Example (Egyptian)

Prompt Testleri

Bu üründe muhakkak en önemli detaylardan biri de prompt olacaktır. Bu nedenle öncelikle dünyadaki en popüler interior design stilleri araştırıldı. Popüler olan stiller;

1. Transitional Interior Design
2. Traditional Interior Design
3. Modern Interior Design
4. Eclectic Interior Design
5. Contemporary Interior Design
6. Minimalist Interior Design
7. Mid Century Modern Interior Design
8. Bohemian Interior Design
9. Modern Farmhouse Interior Design
10. Shabby Chic Interior Design
11. Coastal Interior Design
12. Hollywood Glam Interior Design
13. Southwestern Interior Design
14. Rustic Interior Design
15. Industrial Interior Design
16. French Country Interior Design
17. Scandinavian Interior Design
18. Mediterranean Interior Design
19. Art Deco Interior Design
20. Asian Zen Interior Design

bu şekildedir. Bunların arasında uygun promptları kullanılabilecek olanlar alınmıştır. Buna ek olarak uygulamayı daha eğlenceli hale getirmek için hali hazırda Stable Diffusion'da kullanılan ve popüler olan interior design promptları da incelenmiş ve sonuç olarak;

1. scandinavian
2. hollywood
3. victorian
4. urban
5. coffeshop
6. cyberpunk
7. nouveau
8. egyptian
9. french
10. moroccan
11. deco
12. italian
13. retro
14. cozy

bu stiller ile ürün devam ettirilmiştir. Tüm stillerin çıktılarına [Prompt Test linkinden](#) ulaşılabilir.

Oda Tipi Prompt Testleri

Odanın değişmesine göre içerisinde objeler de değişiklik gösteriyor. Örneğin yatak objesi yatak odasında bulunurken, koltuk genellikle oturma odasında yer alıyor. **YOLO** detection ile input olarak gelen resimde detection işlemi gerçekleştirildi ardından eşyalara göre input görselin hangi odaya ait olduğu bulunup buna göre promptlara (stil prompt + bedroom, stil prompt + kitchen, vs.) eklentiler yapıldı. Arada belirgin bir fark olmadığı için ürüne eklenmeyecek. Sonuçlarına [Room Type Test](#) linkten ulaşabiliriz.

Renk Değişikliği Testi

Kullanıcı istediği stilde çıktıyı aldıktan sonra üzerinde renk olarak değişiklik isteyebilir. Bu nedenle seed sabit kalacak şekilde promptlara renk ile alakalı eklentiler yapıldığı zaman nasıl sonuçlar verdiği test edildi. Çıktı üzerinde yarattığı etki umut verici olduğu için ürüne eklemeye değer bir feature olduğuna karar verildi. Orjinal ve renk promptları ile alınmış çıktılarla [Color Test](#) linkten ulaşılabilir.

Development

Kodun yapısında temel olarak kullanılan ve üzerinde değişiklik yapılması olası komponentler **class(ControlNetFactory)**, **class(StableDiffusionFactory)**, **dict(promptDict)**, **class(Style)** ve **class(Color)**. Bu nedenle bu class'lar yeni implementasyonlar gerektiği zaman biribirleriyle uyumlu çalışacak şekilde develop edilmiştir.

Stable Diffusion + ControlNet yapısını depth estimation ile yapılmasına karar verildi. Github repository'e [bu linkten](#) ulaşılabilir. Kodun genel yapısında Stable Diffusion ve ControlNet modelleri için **factory based** bir yaklaşımyla development'ı gerçekleştirildi. Tüm methodların docstring'lerinde input tipleri, ne işlevleri oldukları vs. gibi bilgiler yer almaktadır. Aşağıda initialize ederken ve inference alırken hangi işlemlerin gerçekleştiği **Figure 7.** ve **Figure 8.**'de yer almaktadır.

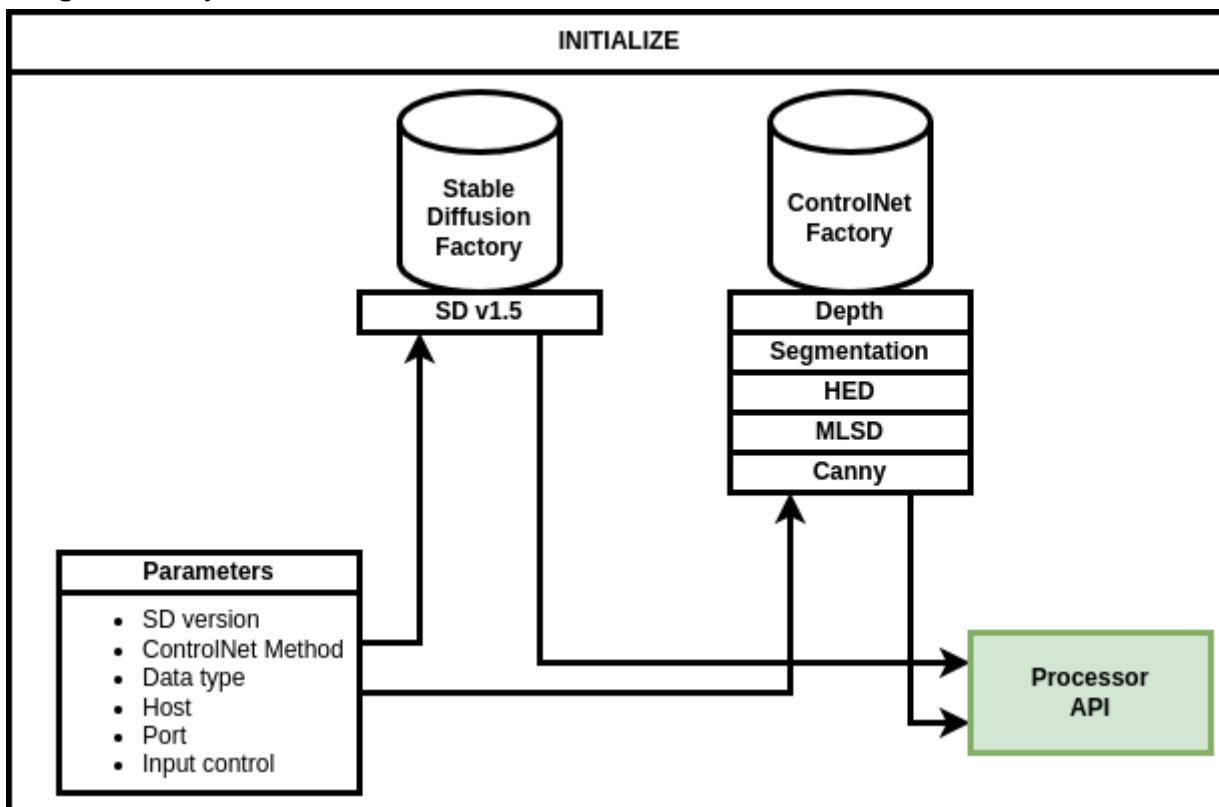


Figure 7. ProcessorAPI Initialize

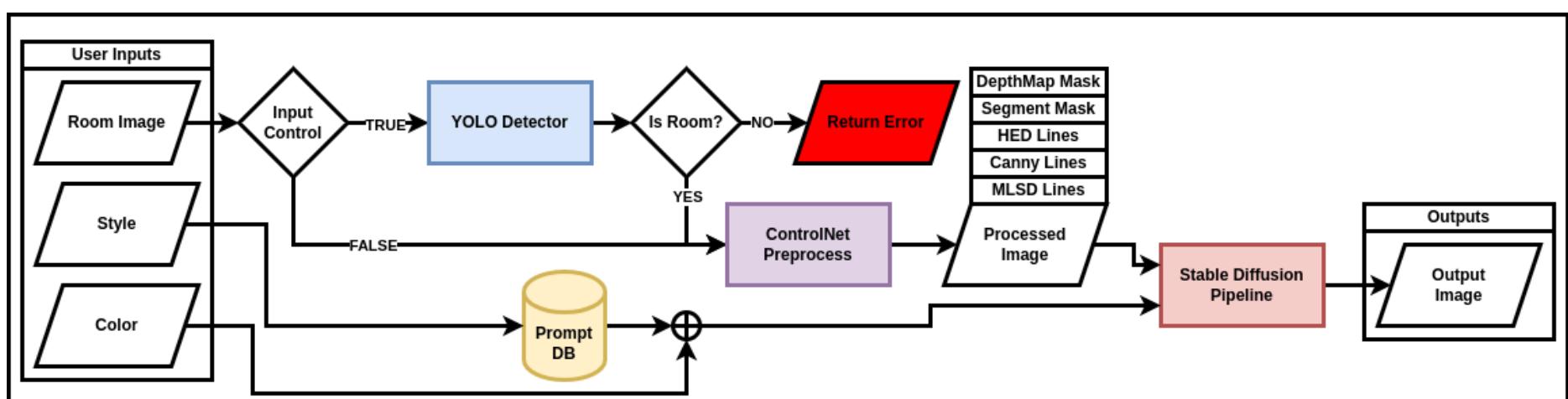


Figure 8. Interior Design App Pipeline

Optimizations

Stable Diffusion'lar yapısı gereği büyük modellerdir. Bu modellerin daha küçük ortamlarda çalışmasını sağlamak için çeşitli çalışmalar yapılmıştır. Uygulanabilecek en kolay yöntem data type'ının fp32'den fp16'ya çevrilmesidir. 4090 ekran kartlı bir bilgisayarda hız testleri yapılmıştır. Bu teste ek olarak çıktılarla [Datatype Test](#) linkten ulaşılabilir. Stable Diffusion modellerinin int8 type ile çalışması üzerine çalışmalar vardır.

- **4090 GPU**
 - **fp32 FPS:** 0.10613391395078976
 - **fp16 FPS:** 0.26057780235377453

Uygulamanın daha doğru çalışması adına YOLO ile input kontrolü methodu geliştirilmiştir. Bu yöntem input gelen görsel üzerinde object detection yapar ve detect edilen class'lar doğrultusunda görselin gerçekten bir odaya ait olup olmadığını tespit eder.

Memory'de daha fazla yük olmaması adına ürüne implementasyonu yapılmamış olsa da RealESRGan gibi super resolution network'leri ile çıktıının çözünürlüğünü artırılabilir. Repository'e direkt olarak eklenmemiş olsa da kendi repository'sinde alınan sonuçlar [Prompt Test HQ](#) linkten ulaşılabilir.

Kullanıcılara daha esnek bir alan sağlamak ve uygulamanın başarısını artıracağımı düşündüğüm için kullanıcının renk kontrolü yapabilmesine olanak sağlanmıştır. Standart çıktıya göre renk tonu vererek odanın o renk tonundaki bir stil ile çıktısını alacaktır.

Deployment

Dockerfile repository'de mevcut. 2 farklı kurulum yöntemi var. Biri manuel olarak 2 adımda docker üzerinde build alıp run ediyor, diğeri ise start.sh dosyası ile verdığınız variable'lar doğrultusunda tek seferde API'yi ayağa kaldırıyor.