



FLUTTER 101

PART 2

M. Halil Akkaynak - H. Bahadır Arın

Flutter Giriş ve Widget Yapıları



Part 1

Dart Programlama Dilini
Anlama ve Uygulama

Part 2

Part 3

Demo

Widget nedir?

Ekranda gördüğümüz her şeyin temel yapı taşıdır. Bu basit bir buton veya karmaşık bir animasyon hatta bütün uygulama arayüzümüz gibi çeşitli öğeleri içerebilir.

Widget nedir?

widgetlar sayesinde tasarımlarımızı modüler bir hale getirebiliriz

Widget katalogdan hazır widgetları kullanabileceğimiz gibi kendi widgetlerimizi da yazabiliriz

It's All Widgets!



Statefull/Stateless Widgetlar



Stateless Widget



Stateless Widget

Flutter içerisinde ise bu sınıf ekran ilk oluşturulduğunda bir kez çalışan ve orada sabit bir şekilde kalan güncellenmeyen widget'lardır



Özellikleri

- Basit ve performanslı
- Yeniden kullanabilir
- Sadece statik içerik için uygundur.

Statefull Widget



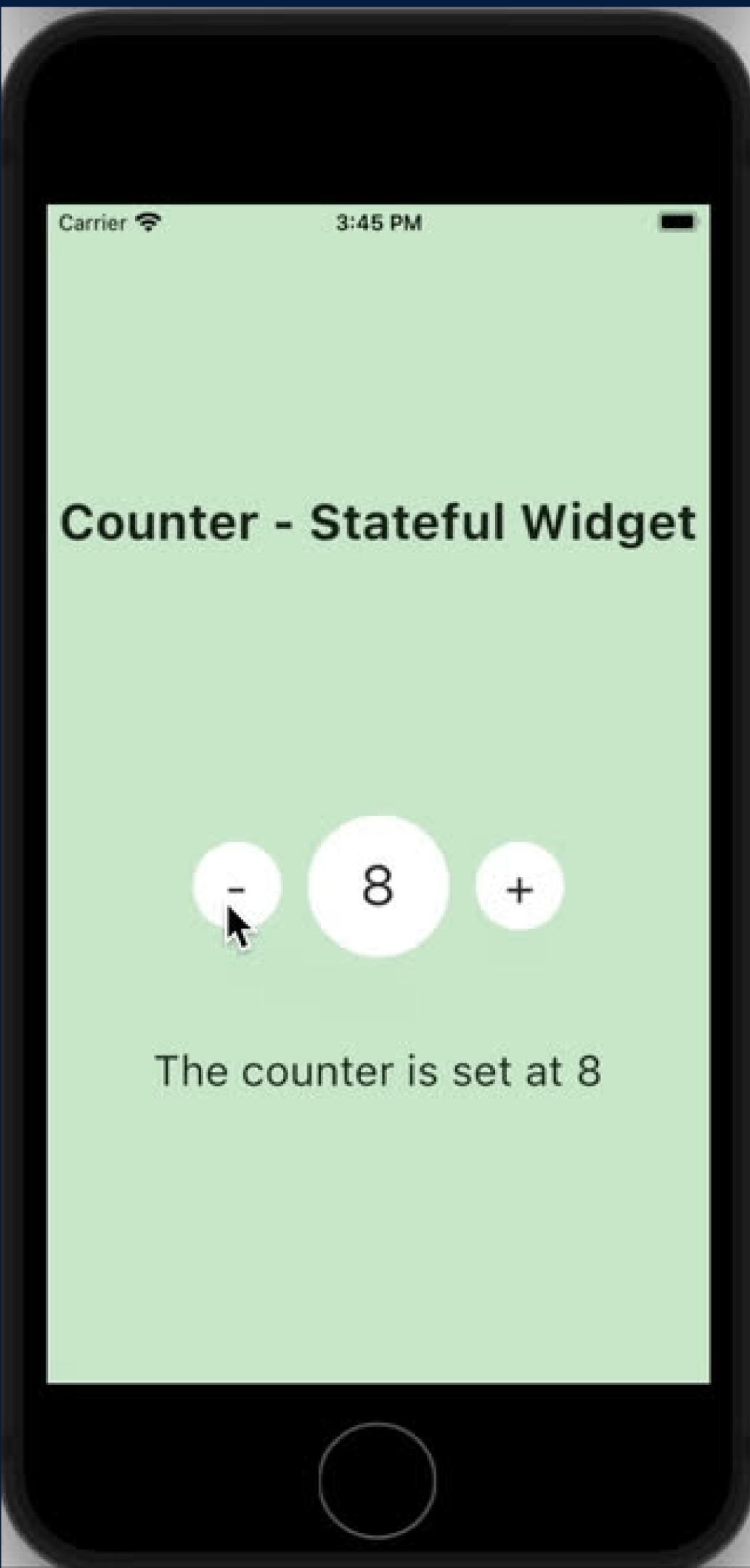
Statefull Widget

Değişebilen duruma sahip olan bir widget türündür. Yani kullanıcı etkileşimi gibi durumlara bağlı olarak içeriği veya görünümü değiştirebilir.

Özellikleri

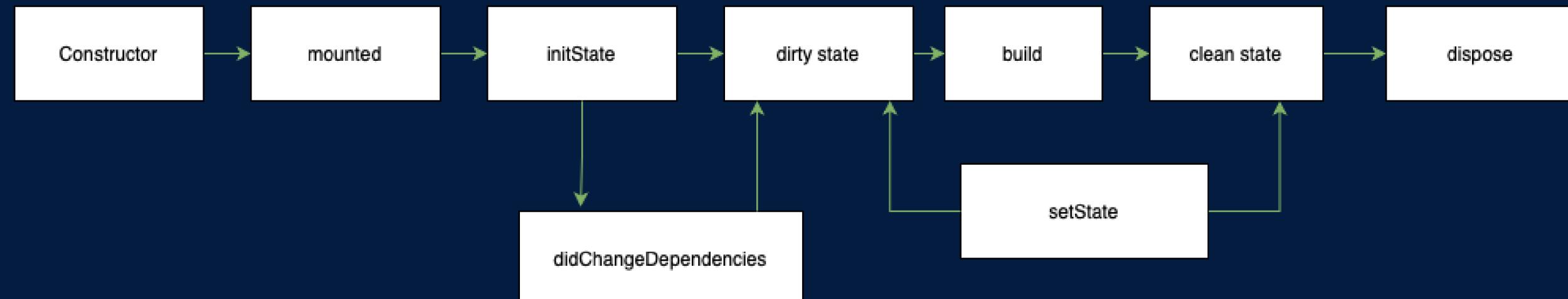
- Dinamik ve interaktiftir.
- Durum yönetimi gerektirir.
- Stateless widget' a göre daha karmaşık ama daha esnektir.

Statefull Stateless



Yaşam Döngüsü





1. **Constructor** çağrırlır.
2. State nesnesi, ağaçta bir `BuildContext` veya konumla ilişkilendirilir. Widget artık **mounted** olarak kabul edilir.
3. **initState** çağrırlır. Bu method sadece bir kere çağrırlır ve state nesnesindeki özellikleri başlatmak için kullanılır.
4. **didChangeDependencies** çağrırlır.
5. Bu noktada state **dirty** kabul edilir ve bu sayede Flutter hangi widget'ların yeniden oluşturulması gerektiğini takip edebilir. İlk kez de dahil olmak üzere, bir state nesnesi oluşturulması gerektiğinde, kendisini **dirty** olarak işaretler.
6. **build** methodu ile state nesnesi tamamen başlatılır.
7. State **clean** olarak işaretlenir. State nesnesi amaçlandığı gibi görüntülenir ve framework'ün kendisine daha fazla talimat vermesini bekler.
8. **setState** state'i her zaman **dirty** olarak işaretler.
9. **dispose** state nesnesinin ağaçtan kaldırılması için çağrırlır.

Widgets



Layout Widgets

- Container: BoxDecoration widget'ları kullanılarak dekore edilmiş dikdörtgen bir kutu.
- Center: Child widget'ı ortalar.
- Row: Child widget'larını yatay doğrultuda yerleştirir.
- Column: Child widget'larını dikey doğrultuda yerleştirir.
- Stack: Widget'ları birbirinin üzerine yerleştirir.



Material Widgets



Öneriler



Öneriler

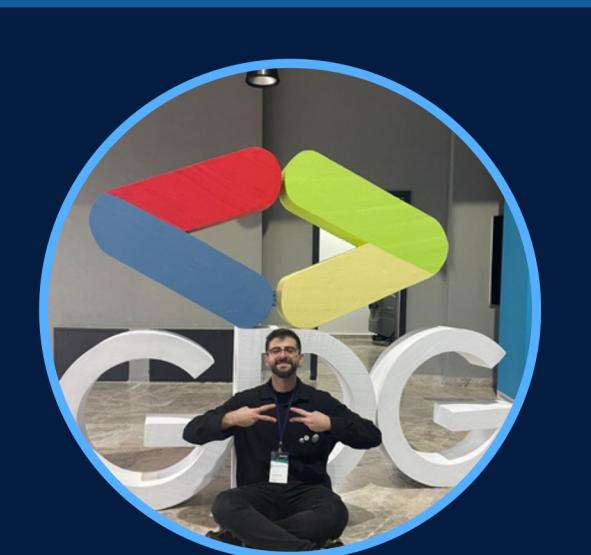
Fonksiyonlar, kodunuzu parçalara ayırmanızı ve tekrar kullanılabilir kod blokları oluşturmanızı olanak tanıyan temel programlama öğeleridir.



Kaynakça

Fonksiyonlar, kodunuzu parçalara ayırmanızı ve tekrar kullanılabilir kod blokları oluşturmanızı olanak tanıyan temel programlama öğeleridir.

SORULAR



M. Halil
Akkaynak

Flutter-Backend
Developer



H. Bahadır
Arın

Flutter
Developer

