

week-2-assignment-fullyaaa

Lifecyle Nedir?

Lifecycle, bir uygulamanın activitylerinde başlangıçtan sona kadar kullanımı esnasında oluşan dögüdür. Uygulamamızda activitynin davranışını lifecycle metotlarıyla belirleyebiliriz. Kullanıcı lifecycle ile uygulamanın farklı adımlarında durabilir, uygulamadan çıkıp tekrar açtığında bu noktada dahil olabilir.

Nasıl yönetiriz?

Lifecycle'ı yönetirken kullanılan metotlar; **onCreate()**, **onStart()**, **onResume()**, **onPause()**, **onStop()** ve **onDestroy()**. Kısaca açıklayacak olursak;

onCreate(): Uygulamayı ayağa kaldırdığımız metoddur. Uygulama ilk açıldığında gerçekleşen olayları içinde barındıran ve her uygulamada olması gereken metoddur. *savedInstanceState* ve *setContentView()* olarak iki parametresi vardır. *setContentView()* kullanıcı arayüzünde gösterilen xml tasarımını çağırır. *savedInstanceState* ile kullanıcının oluşturduğu modelleri çağırabilir, veri alışverişini sağlayabiliriz. **onCreate** metodu sonlandığında, uygulama **onStart()** metodunu çağırır.

onStart(): **onCreate** veya **onRestart()** metotlarından sonra çağırılabilir. Bu metot aktiviteyi kullanıcıya görünür hale getirir yani kullanıcının gördüğü UI'yı başlatılıp gösterildiği metoddur.

onResume(): Uygulamanın kullanıcı ile etkileşime girdiği durumdur. Uygulama durdurulduğu süre boyunca **onPause()**da kalır. Uygulama tekrar çağırıldığında **onResume()** metodu çağırılır. Örneğin; bir telefon araması alındığında uygulama ekranının kapatılmasına sebep olur. Uygulama akışı değiştiğinde **onPause()** metodu devreye girer. Uygulamayı tekrar aktif hale getirdiğimizde ise **onResume()** metoduna geri döneriz.

onPause(): Bu metot uygulama arka plana atıldığında tetiklenir. En yaygın durum olarak uygulama kesintiye uğradığında çalışır(**onResume** metodundaki örneğimizde olduğu gibi).

Uygulamamız çalışırken duraklatıldığında gerçekleşmesi gereken işlemleri yapmamız gerektiğinden dolayı bu metodu kullanırız. Uygulama silinir veya kapatılırsa **onStop()** metodu devreye girer.

onStop(): Uygulama silinir veya kapatılırsa onStop() metodu devreye girer. Kullanıcı uygulamadan çıkış yaptığında bu metoddaki kodlar çalışmaya başlar. Bu metodun kullanılması CPU'daki gereksiz işlemlerin kapatılmasına yardımcı olduğunu için önemlidir.

onDestroy(): Bir activity yok edilmeden önce sistem bu metodu çağırır. Böylelikle yerden tasarruf edilerek aktivite yok edilir.

Ne gibi problemler olur ve nasıl kaçırız?

Bileşenlerin doğru kullanılmaması, sistemin önemli işler yaparken uygulamanın sürecini öldürmesine neden olabilir. Örnek verecek olursak; uygulamayı açtığımızda onCreate(), onStart(), onResume() metotları çalışır. Fakat screen rotate işleminden sonra şu metotları tetiklenir; onPause(), onStop(), onDestroyed(). Ardından destroy içerisinde silinen datalar onCreate içerisinde tekrar yüklenir. onCreate(), onStart(), onResume() tetiklenir. Bu istenmeyen bir dönüştür. Bunu engellemek için, onSaveInstanceState metodu ile veriyi kaydedip, onRestoreInstanceState metodu ile kaydettiği veriyi geri getiriyor. Böylece ikinci kez onCreate metodu çalıştığında kaydettiği veriyi getirmiş olur. Diğer yöntem olarak Android manifeste ekleyeceğimiz android:configChanges="orientation|keyboardHidden|screenSize" ile bu durumdan kaçınabiliriz. Bu eklenti aktivitenin yeniden oluşturulmasını engeller ve veri kaybının önüne geçer.