

APPCENT NASA CASE STUDY

KOCAELİ ÜNİVERSİTESİ BİLGİSAYAR MÜHENDİSLİĞİ

YASİN ÖMER KARA

ÖZET

Bu proje ile pagination ve recyclerview yapıları kullanılarak bir uygulamayı tekrar derlerseniz API'den alınan dataların infinite scroll olarak gösterilmesi ve click eventiyle içeriklerine ulaşılabilmesi hedeflenmiştir.

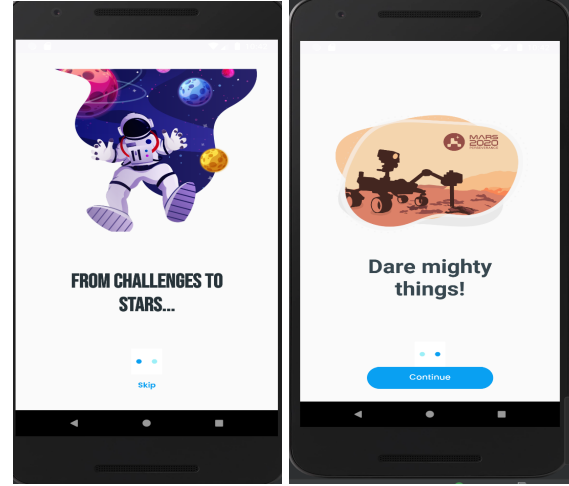
1.GİRİŞ

Uygulama için tavsiye edilen emulator Nexus 5X API 27'dir. Generic olarak Constraint Layoutlar dikkate alınmış olsada test edilmemiştir.

Uygulama ilk açıldığında lottie animation kullanarak bir basic splash screen hazırlanmıştır.



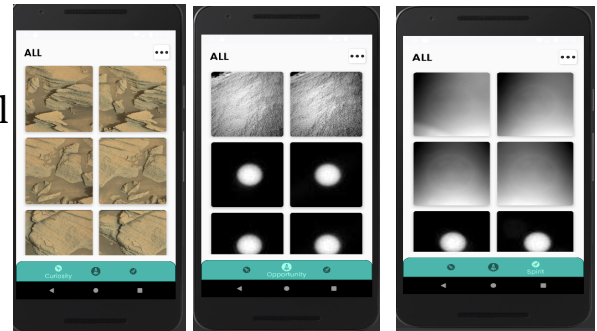
geçerseniz ve



2.Ana Sayfalar

First Launch'a özel olarak 2 sayfalı onboarding screenleri hazırlanmıştır.

Eğer OnBoarding ekranlarını



Tablaların altında yazan isimlere göre API'ye servis istekleri atılmıştır.

Ayrı ayrı Curiosity, Opportunity ve Spirit gemilerinden çekilen datalar, pagination kullanılarak ekrana yansıtılmıştır.

PagingAdapter:

Paging yapısı ve RecyclerView yapısı bind ve create işlemleri için Adapter.kt sınıfını;

```
override suspend fun load(params: LoadParams<Int>): LoadResult<Int, Photos> {
    val page = params.key ?: DEFAULT_PAGE_INDEX
    return try {
        val token = "PicEduciy8bCJj2ak1SEutQPILoalu2gmKVJKUqy"
        when (type) {
            "Curiosity" -> {
                if (cam.isNotEmpty()) {
                    val response = curiosityApiService.getCuriosityImagesCam(
                        cam,
                        token,
                        page,
                        params.loadSize,
                    )
                    LoadResult.Page(
                        response.photos,
                        prevKey = if (page == DEFAULT_PAGE_INDEX) null else page - 1,
                        nextKey = if (response.photos.isEmpty()) null else page + 1
                    )
                } else {
                    val response = curiosityApiService.getCuriosityImages(
                        token,
                        page,
                        params.loadSize
                    )
                    LoadResult.Page(
                        response.photos,
                        prevKey = if (page == DEFAULT_PAGE_INDEX) null else page - 1,
                        nextKey = if (response.photos.isEmpty()) 1 else page + 1
                    )
                }
            }
            "Opportunity" -> {
                if (cam.isNotEmpty()) {

```

Fetching işlemi için;

ImagePagingSource.kt sınıfını inceleyebilirsiniz.

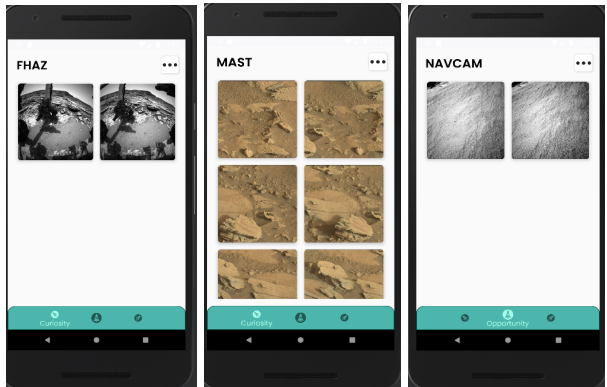
** Burada sayfalarda eğer herhangi bir kamera filtrelemesi yapılmadıysa infinite scroll olacak şekilde ayarlanmıştır. Eğer user sadece belli bir kameranın fotoğraflarını görmek istiyorsa varolan total fotoğrafı pagination yapısıyla görücek şekilde düzenlenmiştir.

**Bu yapı isteğe bağlı ufak bir kod değişimiyle değiştirilebilir.

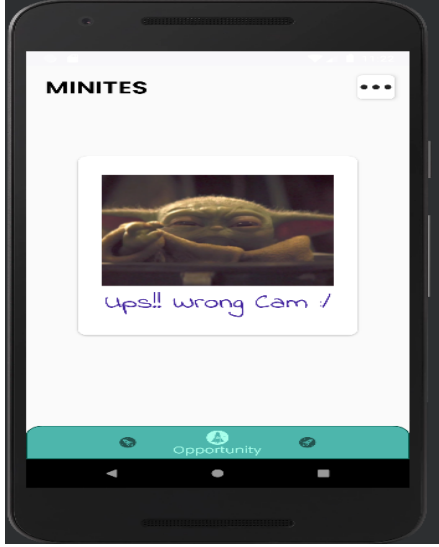
Buradaka cam.isNotEmpty() caseinin içerisinde ;

nextKey =
if(response.photos.isEmpty() null yerine 1 yazılırsa spesifik kameralara ait olan durumlar da infinite scroll olacaktır.

Spesifik kamera seçildikten sonraki görüntüler;



Eğer geminin o kameraya ait fotoğrafı yoksa;



Opportunity ve Spiritin gelen datalarda sadece 1 page datası olduğundan gelen datalarda bazı kameralara ait fotoğraf bulunmamaktadır. Gelen dataları loglardan kontrol edebilirsiniz.

Abbreviation	Camera	Curiosity	Opportunity	Spirit
FHAZ	Front Hazard Avoidance Camera	✓	✓	✓
RHAZ	Rear Hazard Avoidance Camera	✓	✓	✓
MAST	Mast Camera	✓		
CHEMCAM	Chemistry and Camera Complex	✓		
MAHLI	Mars Hand Lens Imager	✓		
MARDI	Mars Descent Imager	✓		
NAVCAM	Navigation Camera	✓	✓	✓
PAN/CAM	Panoramic Camera		✓	✓
MINITES	Miniature Thermal Emission Spectrometer (Mini-TES)		✓	✓

Bu tabloya göre kontrol edilmiştir.

3.KAYNAKÇALAR

- <https://api.nasa.gov/index.html#main-content>
- <https://medium.com/swlh/paging3-recyclerview-pagination-made-easy-333c7dfa8797>
- <https://developer.android.com/topic/libraries/architecture/paging/v3-paged-data>
- <https://www.section.io/engineering-education/handling-recyclerview-clicks-the-right-way/>
- <https://developer.android.com/reference/kotlin/androidx/paging/PagingConfig>
- <https://stackoverflow.com/questions/54954417/android-paging-how-to-refresh-data>
- <https://medium.com/swlh/paging3-recyclerview-pagination-made-easy-333c7dfa8797>
- <https://github.com/android/architecture-components-samples/tree/main/PagingSample>
- <https://stackoverflow.com/questions/8617403/how-can-you-check-if-status-bar-is-visible-in-android>

Paging3 kütüphanesi ile ilk deneyimimdi, bu keyifli case study'e aracı olduğunuz için teşekkürler.

Yasin Ömer Kara