



# Android Tabanlı Akıllı Telefonlar için bir Hava Durumu Uygulaması

Yazılım Mühendisliği Ana Bilim Dalı

Dönem Projesi

Emre Özçelik

ORCID 0009-0007-1187-4976

Proje Danışmanı: Doç. Dr. Sıla Övgü Korkut Uysal

Haziran 2024

# Android Tabanlı Akıllı Telefonlar için bir Hava Durumu Uygulaması

## Öz

Hava durumu bilgilerine erişmek, özellikle iklim koşullarının sert ve değişken olduğu bölgelerde, kullanıcıların iç ve dış mekân aktivitelerini planlamaları açısından önemlidir. Günümüzde akıllı telefonların kullanımı giderek yaygınlaşmaktadır. Akıllı telefonların birçoğunda, Android işletim sistemi kullanılmaktadır. Bu çalışmada, Android tabanlı akıllı telefonlar için bir hava durumu uygulaması geliştirilmiştir. Hava durumu uygulamasının arayüzü, Jetpack Compose araç seti kullanılarak geliştirilmiştir. Programlama dili olarak Kotlin kullanılmıştır.

**Anahtar Kelimeler:** Hava durumu uygulaması, Android, Jetpack Compose, Kotlin

# A Weather Application for Android-Based Smartphones

## Abstract

Access weather information is important for users to plan indoor and outdoor activities, especially in regions where climatic conditions are harsh and variable. Today, the use of smart phones is becoming increasingly widespread. Android operating system is used in most of the smart phones. In this study, a weather application has been developed for Android based smartphones. The interface of the weather application has been developed using Jetpack Compose toolkit. Kotlin has been used as the programming language.

**Keywords:** Weather app, Android, Jetpack Compose, Kotlin

# İçindekiler

Öz .....	i
Abstract .....	ii
Şekiller Listesi.....	iv
<b>1 Giriş .....</b>	<b>1</b>
<b>2 Materyal ve Metotlar .....</b>	<b>3</b>
2.1 Veriler.....	3
2.2 Android Studio .....	3
2.3 Kotlin.....	3
2.4 Jetpack Compose.....	4
2.5 MVVM Mimarisi .....	5
2.6 Kütüphaneler .....	5
2.7 Google Play Hizmetleri .....	6
2.8 Farklı Ekran Boyutlarına Uyumluluk.....	6
2.9 Desteklenen Android İşletim Sistemi Sürümleri.....	7
<b>3 Sonuçlar ve Tartışma.....</b>	<b>8</b>
<b>4 Sonuç .....</b>	<b>20</b>
<b>Kaynaklar .....</b>	<b>21</b>

# Şekiller Listesi

Şekil 1.1: Nisan 2024, Dünya çapında mobil işletim sistemi pazar payı (StatCounter, 2024).....	2
Şekil 2.1: MVVM mimarisi (Bhadoria, 2023).....	5
Şekil 2.2: Android API seviyeleri (API Levels, 2024) .....	7
Şekil 3.1: Konum ekranı .....	8
Şekil 3.2: Ana ekran.....	9
Şekil 3.3: Günlük tahminler ekranı .....	10
Şekil 3.4: Arama ekranı.....	11
Şekil 3.5: Ayarlar ekranı .....	12
Şekil 3.6: Fahrenheit cinsinden sıcaklık değerleri .....	13
Şekil 3.7: Çevrimdışı erişim ekranları .....	14
Şekil 3.8: Uygulamanın İngilizce dil desteği .....	15
Şekil 3.9: Şekil 3.8 (devamı).....	16
Şekil 3.10: İngilizce dilinde çevrimdışı erişim .....	17
Şekil 3.11: 2.7” QVGA ekran görüntüleri .....	18
Şekil 3.12: Şekil 3.11 (devamı).....	19

# Bölüm 1

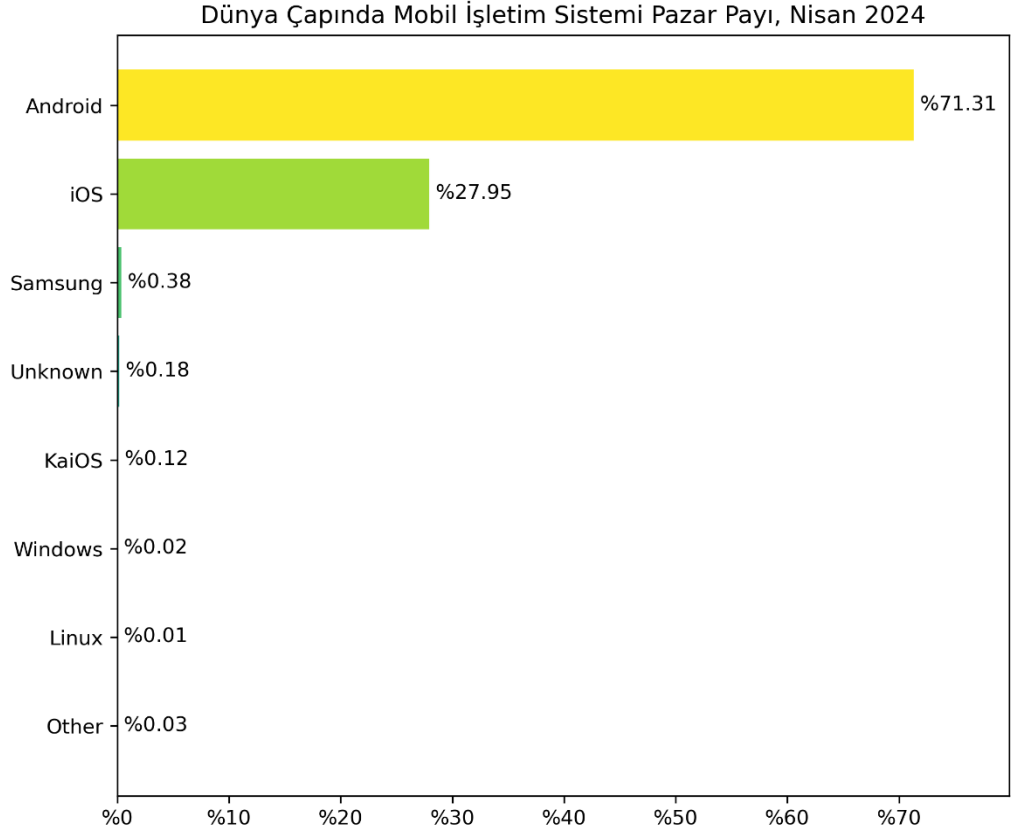
## Giriş

Hava durumu; belirli bir yerde, belli bir anda sıcaklık, rüzgâr, nem, yağış gibi özellikler bakımından atmosferin durumu olarak tanımlanmaktadır (Türk Dil Kurumu, 2024). Hava durumu bilgilerine erişmek, özellikle iklim koşullarının sert ve değişken olduğu bölgelerde, kullanıcıların iç ve dış mekân aktivitelerini planlamaları açısından önemlidir.

2024 yılı için, Dünya'daki akıllı telefon kullanıcıların sayısının 4,88 milyar olduğu ve Dünya nüfusunun %60,42'sinin akıllı telefon kullandığı belirtilmektedir (Turner, 2024). Şekil 1.1'e göre, Nisan 2024 için, Android'in mobil işletim sistemi pazar payı Dünya çapında %71,31 dir. Bu Android'i en çok kullanılan mobil işletim sistemi yapmaktadır. Android'i, %27,95 ile iOS takip etmektedir. İnsanlar günde ortalama 3 saat 15 dakikayı akıllı telefon kullanarak geçirmektedir (Howarth, 2023). Akıllı telefonlar için geliştirilecek olan bir hava durumu uygulaması, kullanıcıların pratik bir şekilde hava durumu bilgilerine erişmesini sağlayacaktır. Bu proje kapsamında, Android tabanlı akıllı telefonlar için bir hava durumu uygulaması geliştirilmiştir.

Srivastava vd. (2020) ve Sapundzhi ve Mladenov (2022) çalışmalarında, Android tabanlı hava durumu uygulaması geliştirmiştir. Geliştirilen uygulamalarda, kullanıcı arayüzü oluşturmak için eXtensible Markup Language (XML) dosyaları kullanılmış olup programlama dili olarak Java kullanılmıştır. Novac v.d (2019) ve Aleksandrov ve Vulović (2019) çalışmalarında Android tabanlı hava durumu uygulaması geliştirme sürecine değinmiştir. Geliştirme süreci, Java ve XML tabanlı anlatılmıştır. Bu çalışmanın en temel farkı, kullanıcı arayüzü geliştirmek için Jetpack Compose araç setinin, programlama dili olarak Kotlin'in kullanılmasıdır.

İkinci bölümde; kullanılan materyal ve metotlara, üçüncü bölümde; sonuçlar ve tartışmaya, dördüncü bölümde; sonuca yer verilmiştir.



Şekil 1.1: Nisan 2024, Dünya çapında mobil işletim sistemi pazar payı (StatCounter, 2024)

# Bölüm 2

## Materyal ve Metotlar

### 2.1 Veriler

Uygulamada kullanılan gün doğumu(Mehwish, 2024), gün batımı(Mehwish, 2024), nem(Those Icons, 2024), rüzgar(riajulislam, 2024), hissedilen sıcaklık(Vector Portal, 2024), yağış(Bqlqn, 2024) ikonları Freepik Company' nin sunduğu Flaticon dijital platformundan alınmıştır.(Freepik Company, 2024).

Hava durumu verileri OpenWeatherMap çevrim içi hava durumu servisinin sunduğu One Call API 3.0 üzerinden alınmıştır (OpenWeatherMap1, 2024). Şehre ait konum bilgisi ve konum bilgisine ait şehir bilgisini almak için OpenWeatherMap' in sunduğu Geocoding API kullanılmıştır (OpenWeatherMap2, 2024). Gökyüzü durumu ikonları OpenWeatherMap' den alınmıştır (OpenWeatherMap3, 2024).

### 2.2 Android Studio

Uygulamanın gerçekleştiriminde Integrated Development Environment (IDE) olarak Android Studio' nun Giraffe 2022.3.1 Patch 2 sürümü kullanılmıştır. Uygulamayı test etmek için Android Studio' nun içinde gelen Android Emülatörü kullanılmıştır. Android Emülatörü, bilgisayarda Android cihazlarını simüle etmektedir (Google, 2023).

### 2.3 Kotlin

Uygulama, yüzde yüz, Kotlin programlama dili kullanılarak geliştirilmiştir. Kotlin, JetBrains tarafından 2010 yılından beri geliştirilen, açık kaynaklı statik bir



programlama dilidir (JetBrains, 2024). Hem fonksiyonel programlamayı hem de nesne yönelimli programlamayı desteklemektedir (Google, 2023).

Android işletim sisteminin sahibi olan Google şirketi, 2017 yılında Kotlin programlama dilini, Android için resmi olarak destekleyeceğini duyurmuştur. Böylece, Google tarafından, Android için resmi olarak desteklenen Java ve C++ programlama dillerine Kotlin'de dahil olmuştur (Hall, 2017).

Google, 2019'da Java programlama dili kullanıcılarını desteklemeye devam ederken; dokümanlarını, örneklerini, eğitim içeriklerini, yeni kütüphanelerini ve araçlarını önce Kotlin dili için tasarlayacağını duyurmuştur (Braun, 2022).

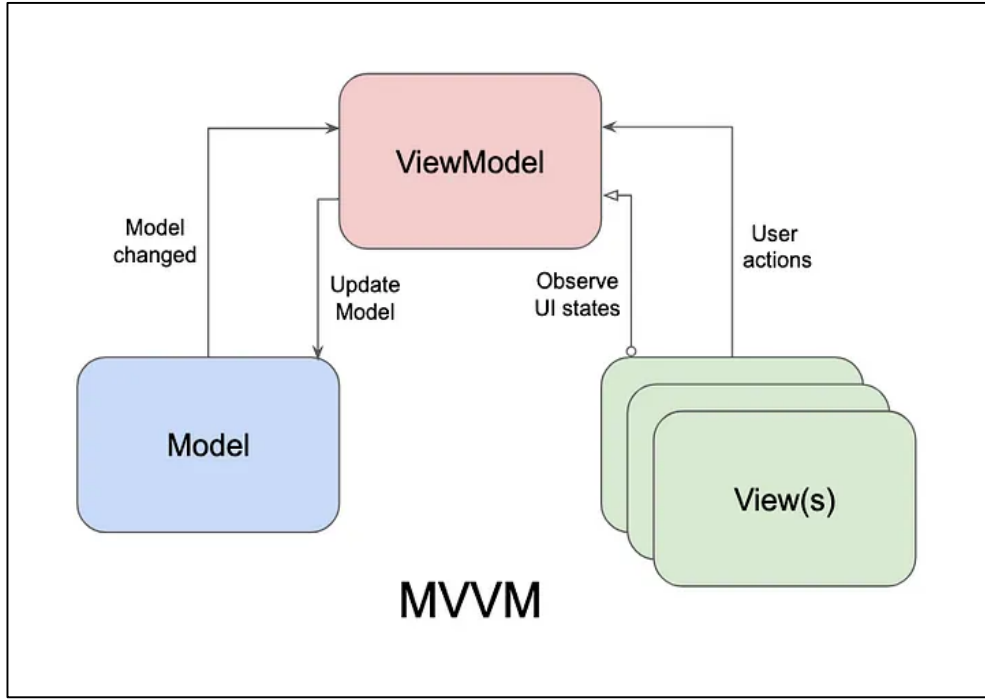
Profesyonel Android geliştiricilerin % 50'sinden fazlası Kotlin'i tercih ederken % 30'u Java'yı kullanmayı tercih etmektedir. Kotlin kullanılarak daha az kod ile daha fazla okunabilirlik elde edilebilmektedir. Kotlin, Java ile yazılmış kütüphane ve çerçevelerle doğrudan çalışabilmektedir (JetBrains, 2023). Kotlin ile geliştirilen Android uygulamaların çökme olasılığının %20'den az olduğu tespit edilmiştir (Muntenescu, 2020).

## 2.4 Jetpack Compose

Uygulamanın kullanıcı arayüzü Jetpack Compose kullanılarak geliştirilmiştir. Jetpack Compose, yerel Android kullanıcı arayüzü oluşturmak için önerilen modern bir araç setidir ve sadece Kotlin programlama dilini desteklemektedir (Google, 2023). Jetpack Compose'nin 1.0 sürümü, 2021 yılında kullanıma sunulmuştur (Bellini & Butcher, 2021). Jetpack Compose geliştirilmeden önce, Android uygulamalarında, kullanıcı arayüzü oluşturmak için sadece Android View sistem kullanılmıştır. Android View sistem'de, kullanıcı arayüzü bileşenleri XML dosyaları kullanılarak oluşturulmaktadır (Kurcheuskay, 2023). Android View sistem ile karşılaştırıldığında, Jetpack Compose kullanımı, kullanıcı arayüzü geliştirmeyi hızlandırıp basitleştirmektedir (Mbano, 2022).

## 2.5 MVVM Mimarisi

Uygulamada Model-View-ViewModel (MVVM) mimarisi kullanılmıştır. MVVM, Android uygulamaları geliştirilirken yaygın bir şekilde kullanılan bir yazılım mimarisidir. MVVM mimarisi üç temel bileşenden oluşur: Model, View ve ViewModel. Model uygulamanın verilerini ve iş mantığını temsil etmektedir. View kullanıcı arayüzünü temsil etmektedir. ViewModel, Model ile View arasında iletişimi sağlayan bir aracıdır; Model'den aldığı verileri, View'in gereksinimlerini uygun şekilde işleyip sunmaktadır. Aynı zamanda kullanıcının View üzerinde gerçekleştirdiği etkileşimleri yakalayıp gerektiğinde modeli güncelleyerek veri akışını yönetmektedir (Bhadoria, 2023). Şekil 2.1'de MVVM mimarisine yer verilmiştir.



Şekil 2.1: MVVM mimarisi (Bhadoria, 2023)

## 2.6 Kütüphaneler

Retrofit, API çağrılarını kolaylaştıran ve bunları daha verimli hale getiren güçlü bir kütüphanedir. Android ve Java platformları için tasarlanmış olan Retrofit, ağ isteklerini tip güvenli bir şekilde yapılmasına olanak tanımaktadır (Vasava, 2023).

Room, Google'nin Android Jetpack kütüphaneleri içinde yer alan, bir kalıcılık kütüphanesidir. Bu kütüphane, geliştiricilerin Kotlin ya da Java dillerini kullanarak, Android cihazlar üzerinde bir SQLite veri tabanı kurma ve bu veri tabanını yönetme süreçlerini basitleştirmektedir (Abeythilake, 2023).

'kotlinx.coroutines', JetBrains tarafından geliştirilmiş, eş yordamları destekleyen kapsamlı bir kütüphanedir. Bu kütüphane asenkron programlama yapılmasına olanak tanıyan çeşitli fonksiyonlar sunmaktadır (JetBrains, 2024).

Coil kütüphanesi, Android platformu için tasarlanmış açık kaynaklı bir görüntü yükleme kütüphanesidir. 'kotlinx.coroutines' kütüphanesi üzerine inşa edilmiştir. Android uygulamalarında görüntüleri yüklemeyi, ön belleğe almayı ve görüntülemeyi sağlamaktadır (Bharti, 2023).

## 2.7 Google Play Hizmetleri

Proje kapsamında geliştirilen uygulamada, cihazın konumunu almak için Google Play Hizmetleri'nin Konum Servisleri kullanılmıştır. Konum Servisleri, Android uygulamalarında konum tabanlı işlevler sağlamaktadır (Google, 2023).

## 2.8 Farklı Ekran Boyutlarına Uyumluluk

Proje kapsamında geliştirilen hava durumu uygulamasının kullanıcı arayüzü, farklı ekran boyutlarına sahip cihazlarda problemsiz çalışabilecek şekilde geliştirilmiştir. Uygulama geliştirme sürecinde, Android Emülatörünün sağladığı Pixel 7 cihazı kullanılmıştır. Google, farklı ekran boyutlarında, kullanıcı arayüzünün problemsiz çalışabilmesi için, testlerde ölçü birimi olarak density-independent pixels (dp) kullanılmasını önermektedir (Google, 2024). Android Emülatörü, sağladığı cihazlarla ilgili bilgileri geliştiricilere sunmaktadır. Pixel 7 cihazı, 412 dp genişliğe, 915 dp yüksekliğe sahiptir. Android'in desteklediği en küçük ekran boyutu, 320 dp genişlik ve 426 dp yüksekliktir. Kullanıcı arayüzü genelde dikey olarak kaydırabilecek şekilde tasarlanmaktadır. Farklı ekran boyutlarında problemsiz çalışabilecek bir kullanıcı arayüzü oluşturulurken daha çok genişliğe dikkat edilmelidir (Hackborn, 2011). Hava durumu uygulaması, Android Emülatörünün sağladığı, 320 dp genişlik ve 427 dp yüksekliğe sahip, 2.7" QVGA cihazında dahi problemsiz çalışmaktadır.

## 2.9 Desteklenen Android İşletim Sistemi Sürümleri

Proje kapsamında geliştirilen hava durumu uygulaması, API seviyesi 25 ve sonrası Android işletim sistemi sürümlerini destekleyecek şekilde tasarlanmıştır. Şekil 2.2’de Android işletim sistemlerinin sürümü, seviyesi, versiyon kodu, kod adı, kümülatif kullanımı ve piyasaya sürüldüğü tarih yer almaktadır.

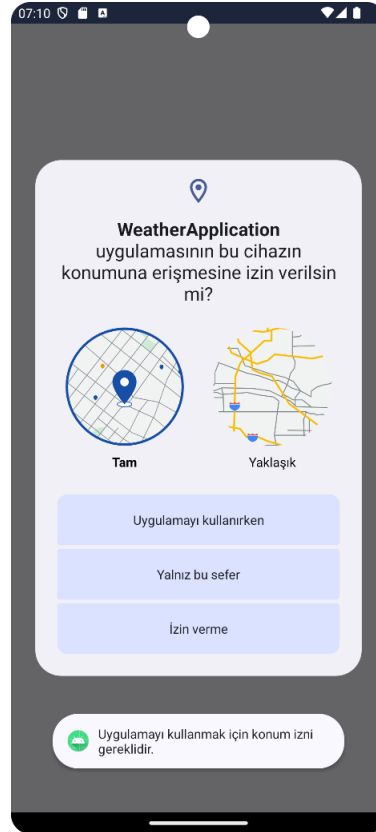
Version	SDK / API level	Version code	Codename	Cumulative usage <sup>1</sup>	Year <sup>4</sup>
Android 15 <sup>DEV</sup>	Level 35	VANILLA_ICE_CREAM	Vanilla Ice Cream <sup>2</sup>	—	TBD
Android 14	Level 34	UPSIDE_DOWN_CAKE	Upside Down Cake <sup>2</sup>	16.3%	2023
	■ targetsdk will need to be 34+ for new apps and app updates by August 31, 2024.				
Android 13	Level 33	TIRAMISU	Tiramisu <sup>2</sup>	42.5%	2022
	■ targetsdk must be 33+ for new apps and app updates since August 31, 2023.				
Android 12	Level 32 <small>Android 12L</small>	S_V2	Snow Cone <sup>2</sup>	59.5%	2021
	Level 31 <small>Android 12</small>	S			
Android 11	Level 30	R	Red Velvet Cake <sup>2</sup>	75.7%	2020
Android 10	Level 29	Q	Quince Tart <sup>2</sup>	84.5%	2019
Android 9	Level 28	P	Pie	90.2%	2018
Android 8	Level 27 <small>Android 8.1</small>	O_MR1	Oreo	92.1%	2017
	Level 26 <small>Android 8.0</small>	O		95.1%	
Android 7	Level 25 <small>Android 7.1</small>	N_MR1	Nougat	95.6%	2016

Şekil 2.2: Android API seviyeleri (API Levels, 2024)

## Bölüm 3

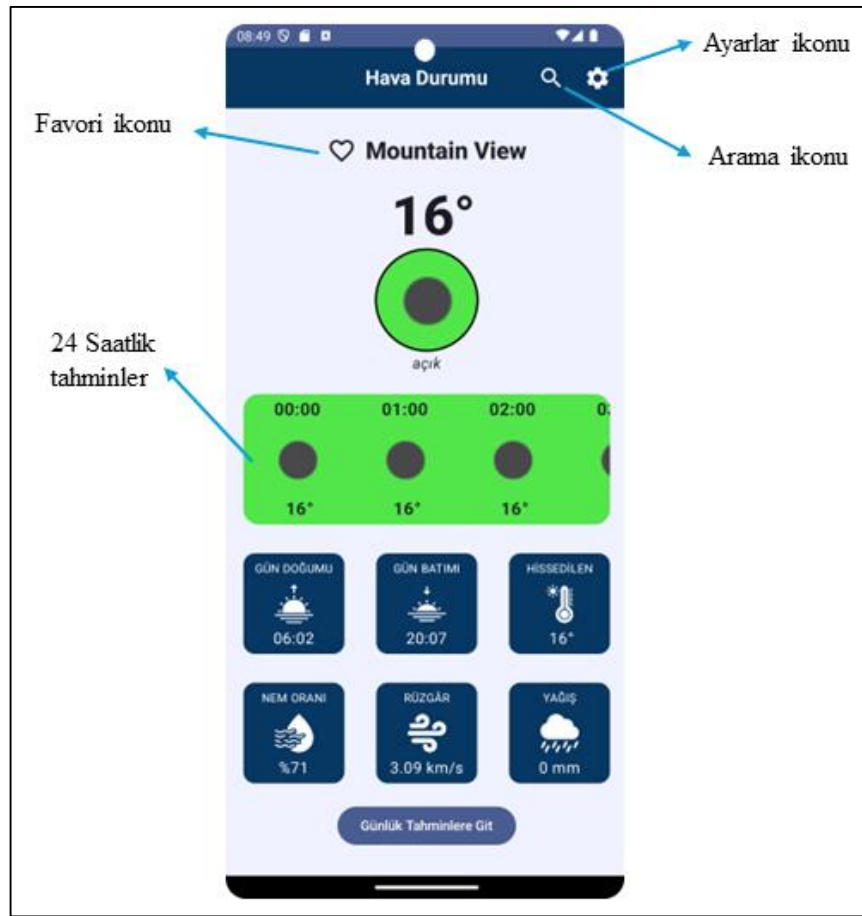
# Sonuçlar ve Tartışma

Uygulama açıldığında, kullanıcıya, Şekil 3.1’deki konum ekranı gösterilmektedir. Konum ekranında, kullanıcıdan konum izni istenmektedir. Aynı zamanda, kullanıcıya, kısa bir süre sonra kaybolan “Uygulamayı kullanmak için konum izni gereklidir” uyarısı gösterilmektedir.



Şekil 3.1: Konum ekranı

Kullanıcı, konum iznini verdikten sonra, Şekil 3.2’de yer alan, ana ekrana yönlendirilmektedir. Uygulamanın ana ekranında, cihazın mevcut konumuna ait şehir, sıcaklık, gökyüzü durumu, nem oranı, hissedilen sıcaklık, gün doğumu, gün batımı, rüzgâr ve yağış bilgileri yer almaktadır. Ayrıca, 24 saatlik, saatlik, gökyüzü durumu ve sıcaklık tahminlerine de yer verilmiştir. Şehir adının yanında yer alan favori ikonuna tıklandığında, ikonun içi kırmızı rengini alıp şehir ismi favorilere eklenmektedir. Favorilere eklenen şehir, çıkartılmak istenirse tekrar ikona tıklanabilir, ikonun içi tekrar beyaz rengini alacaktır.



Şekil 3.2: Ana ekran

Şekil 3.2’de yer alan “Günlük Tahminlere Git” butonuna tıklandığında, kullanıcı, günlük tahminler ekranına yönlendirilmektedir. Şekil 3.3’te günlük tahminler ekranına yer verilmiştir. Bu ekranda, yedi günlük, gökyüzü durumu, en yüksek sıcaklık ve en

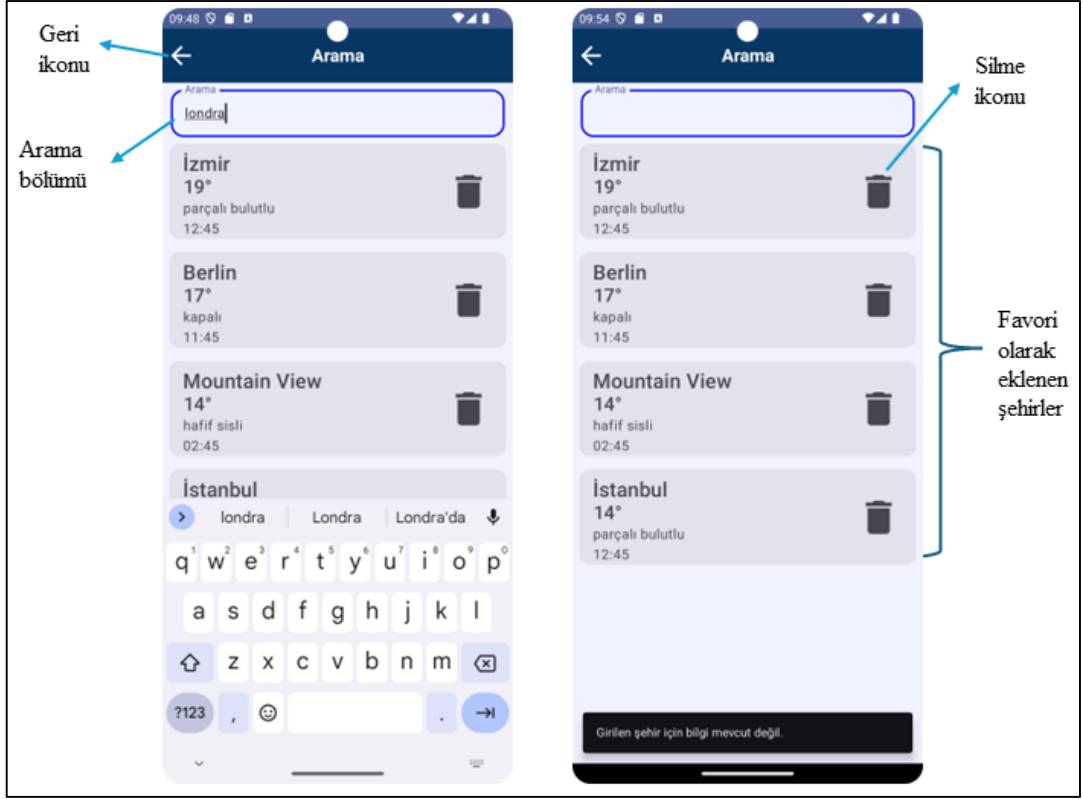
düşük sıcaklık tahminleri yer almaktadır. Kullanıcı geri ikonuna tıkladığında ana ekrana yönlendirilmektedir.



Şekil 3.3: Günlük tahminler ekranı

Şekil 3.2’de yer alan arama ikonuna tıkladığında, kullanıcı, arama ekranına yönlendirilmektedir. Şekil 3.4’te verilen arama ekranında, arama bölümüne tıkladığında, ekranda ortaya çıkan ekran klavyesi kullanılarak arama yapılabilmektedir. Arama yapıldığında kullanıcı ana ekrana yönlendirilerek arama yapılan şehir ile ilgili hava durumu bilgileri görebilmektedir. Eğer, var olmayan bir şehir ismi ile arama yapılırsa, ekranda kısa süre sonra kaybolacak olan “Girilen şehir için bilgi mevcut değil” uyarısı verilmektedir. Arama bölümünün altında favori olarak eklenen şehirler listelenmektedir. Kullanıcıya, favori olarak eklenen her bir şehrin adı, mevcut sıcaklığı, yerel saati ve gökyüzü durumu bilgisi sunulmaktadır. Kullanıcı bu

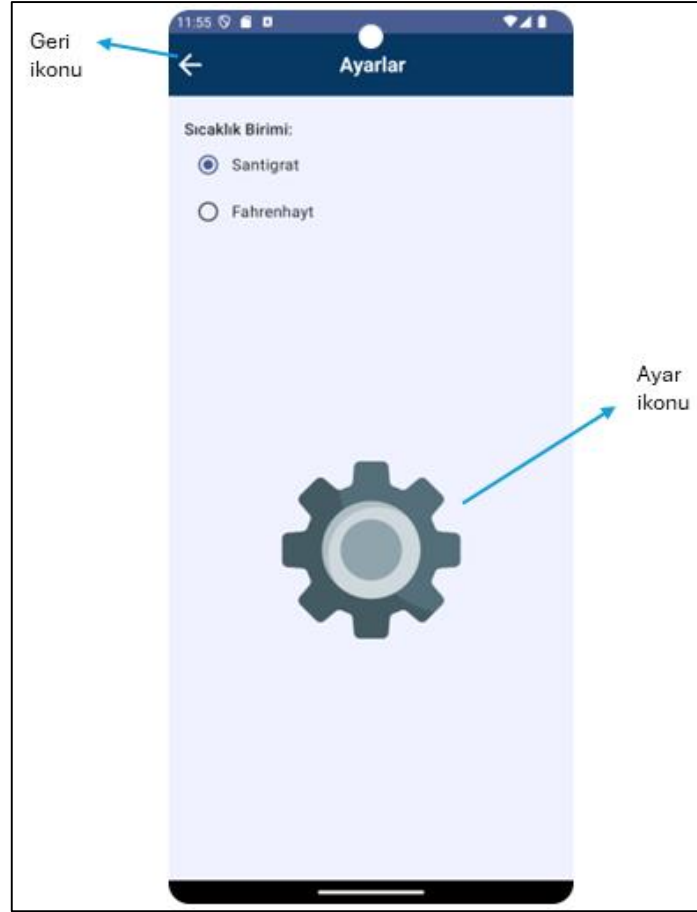
bilgilerin verildiği alana tıkladığında, ana ekrana yönlendirilerek arama yapılan şehir ile ilgili hava durumu bilgileri görülebilmektedir. Kullanıcı silme ikonuna tıkladığında, ilgili şehri favori şehirlerden kaldırabilmektedir. Geri ikonuna tıkladığında kullanıcı ana ekrana yönlendirilmektedir.



Şekil 3.4: Arama ekranı

Kullanıcı Şekil 3.2'deki ayarlar ikonuna tıkladığında, Şekil 3.5'teki ayarlar ekranına yönlendirilmektedir. Ayarlar ekranında varsayılan sıcaklık birimi Santigrat olarak seçilmiştir. Bunun nedeni, Dünya çapında, birçok ülkede sıcak birimi olarak santigrat kullanılmasıdır (Zamboni, 2020). Sıcaklık birimlerinin altında ayarlar ikonu yer almaktadır. Kullanıcı, ayarlar ekranına yönlendirildiğinde, bu ikon kendi ekseninde, saat yönünde 750 milisaniye dönmektedir. Kullanıcı geri ikonuna tıkladığında ana ekrana yönlendirilmektedir.





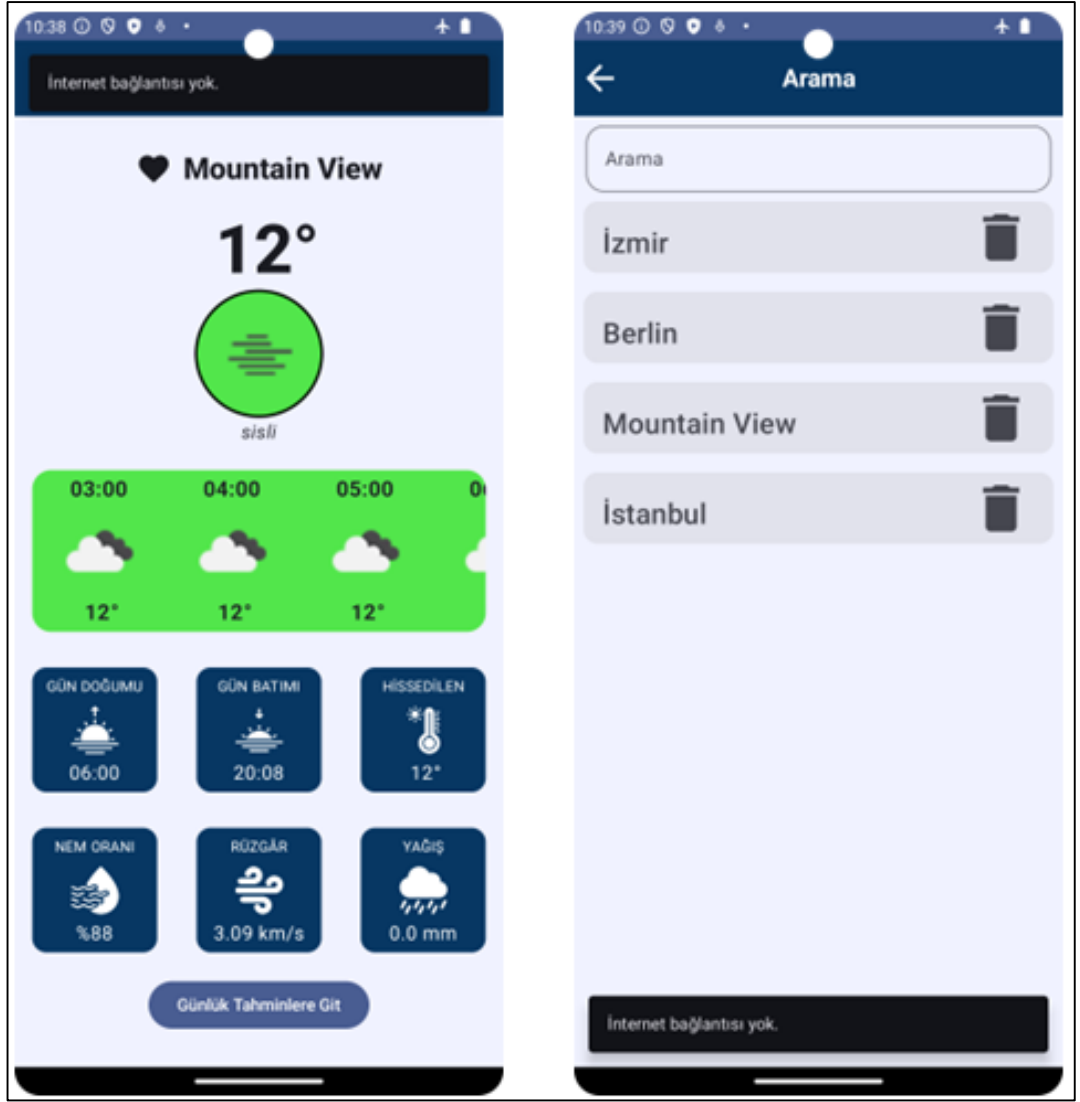
Şekil 3.5: Ayarlar ekranı

Kullanıcı seçili sıcaklık birimini değiştirdiğinde, uygulamadaki tüm sıcaklık değerleri seçili birim değerinde gösterilmektedir. Şekil 3.6’da sıcaklık değerlerinin Fahrenheit cinsinden gösterimine yer verilmiştir.

Şekil 3.7’de uygulama çevrimdışı durumdayken alınan ekran görüntülerine yer almaktadır. Uygulamada internet bağlantısı olmadığında, kullanıcıya, ana ekranın en üstünde kısa bir süre sonra kaybolan ”İnternet bağlantısı yoktur” uyarısı gösterilmektedir. İnternet bağlantısı yok iken kullanıcı, cihazın en son konumu için alınan tüm hava durumu verilerine erişebilmektedir. Kullanıcı, arama ekranında arama yapar ise veya favori şehirlerden birinin hava durumu verilerine erişmek için üzerine tıklar ise, kullanıcıya arama ekranının en altında kısa bir süre sonra kaybolan ”İnternet bağlantısı yok” uyarısı gösterilmektedir.



Şekil 3.6: Fahrenheit cinsinden sıcaklık değerleri

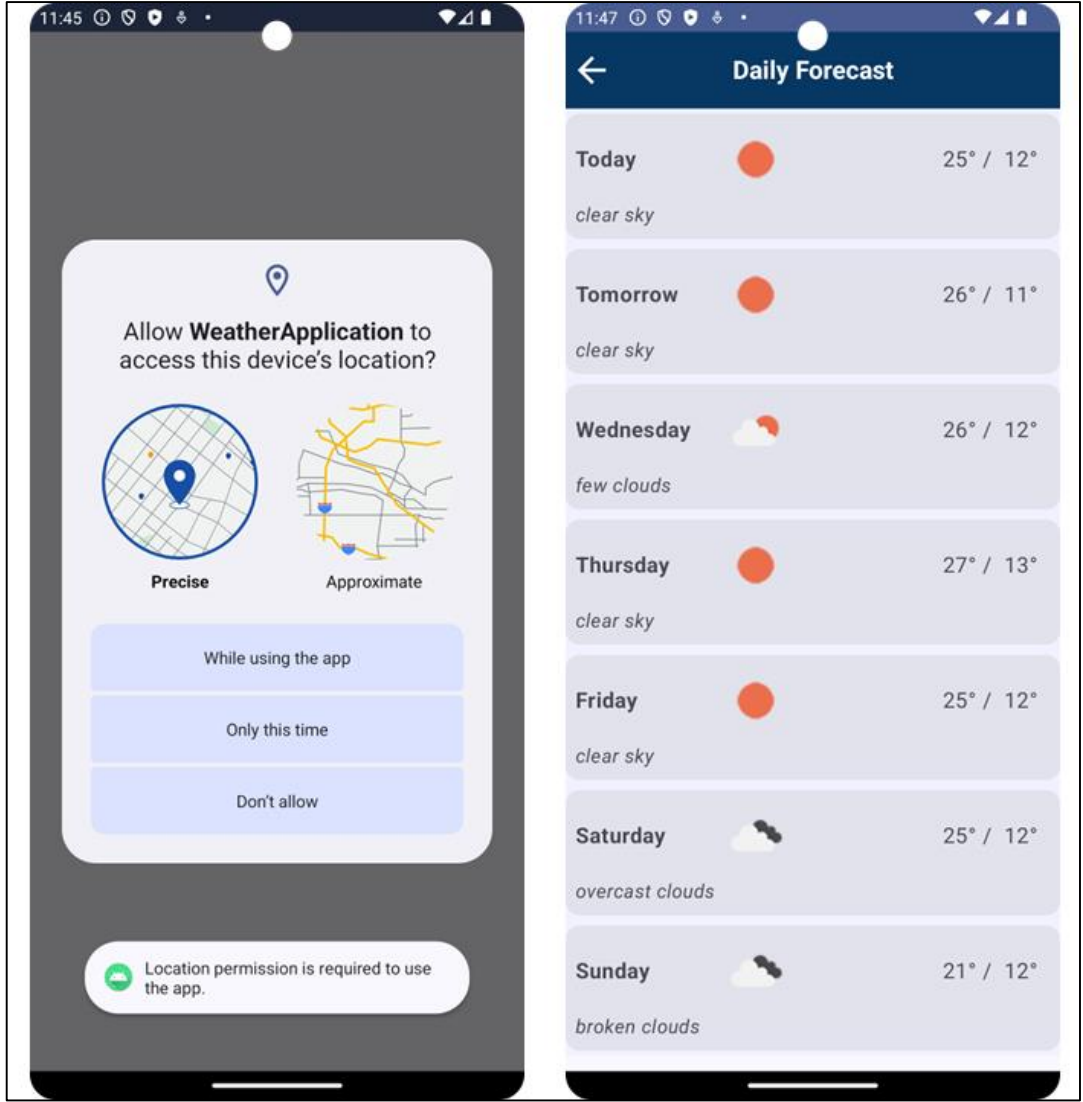


Şekil 3.7: Çevrimdışı erişim ekranları

Cihazın diline göre uygulama Türkçe ve İngilizce olmak üzere iki dil desteği sağlamaktadır. Cihazın dili, Türkçe ise Türkçe dil desteği, İngilizce veya herhangi bir dil ise İngilizce dil desteği sağlanmaktadır. Şekil 3.8 ve Şekil 3.9’da uygulamanın İngilizce dil desteği sağladığı durumlardaki ekran görüntüleri yer almaktadır.

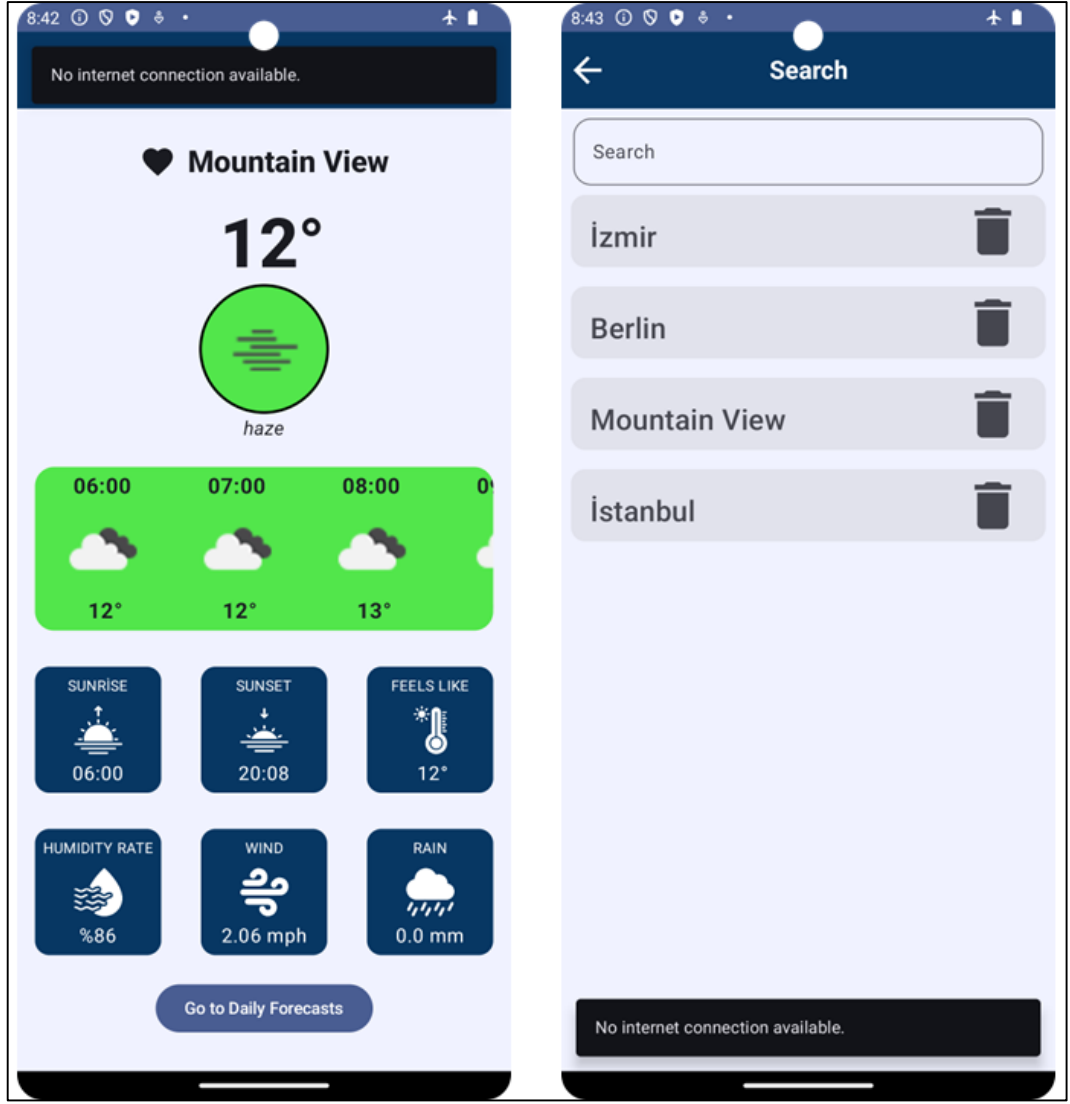


Şekil 3.8: Uygulamanın İngilizce dil desteği



Şekil 3.9: Şekil 3.8 (devamı)

Şekil 3.10'da internet bağlantısı olmadığı durumlardaki İngilizce dil desteğine yer verilmiştir. Kullanıcı, ana ekrana yönlendirildiğinde, ekranın üst bölgesinde, kısa bir süre sonra kaybolacak olan "No internet connection available" uyarısı gösterilmektedir. Aynı zamanda, arama ekranında, arama yaptığı veya favori şehirlerden birinin üzerine tıkladığında, kullanıcıya, ekranın alt bölgesinde aynı uyarı gösterilmektedir.



Şekil 3.10: İngilizce dilinde çevrimdışı erişim

Şekil 3.11 ve Şekil 3.12’de Android Emülatörünün sağladığı, 2.7” QVGA cihazından alınan ekran görüntülerine yer almaktadır.



Şekil 3.11: 2.7” QVGA ekran görüntüleri



Şekil 3.12: Şekil 3.11 (devamı)

Proje kapsamında, kullanıcıların ihtiyaç duyduğu temel hava durumu bilgilerini veren bir hava durumu uygulaması geliştirmek amaçlanmıştır. Gelecek çalışmalarda; arka uç tarafında kod optimizasyonu yapılarak uygulamanın performansını artırılabilir, kullanıcı arayüzü iyileştirilebilir, uygulamanın sağladığı dil desteğini artırılabilir, uygulamaya ek özellikler eklenebilir.



# Bölüm 4

## Sonuç

Proje kapsamında geliştirilen akıllı telefonlar için Android tabanlı hava durumu uygulaması, Jetpack Compose araç seti ve Kotlin programlama dili kullanılarak geliştirilmiştir. Uygulama, Android işletim sistemine sahip tüm akıllı telefon boyutlarına uyum sağlayacak şekilde geliştirilmiştir. API seviyesi 25 ve üzeri Android işletim sistemi sürümlerini desteklemektedir.

Uygulama, cihazın mevcut konumunun, hava durumu bilgilerini vermektedir. Aynı zamanda, kullanıcı ilgilendiği şehir ile ilgili arama yapıp hava durumu bilgilerini görüntüleyebilmektedir. Hava durumu bilgilerine hızlı bir şekilde erişilmek istenen şehirler, kaydedilip silinebilmektedir. Türkçe ve İngilizce olmak üzere, çoklu dil desteği sağlanmaktadır. Sıcaklık birimi isteğe göre Fahrenheit veya Santigrat olarak ayarlanabilmektedir. Çevrimdışı erişim özelliği, cihazın en son konumu için internetin olmadığı zamanlarda, hava durumu verilerine erişim olanağı sağlamaktadır.

Gelecekteki çalışmalarda, arka uç tarafında kod optimizasyonu yapılabilir, kullanıcı arayüzü iyileştirilebilir, dil desteği artırılabilir, ek özellikler eklenebilir.

# Kaynaklar

API Levels. (2024, 18 Mayıs). *Android API Levels*. <https://apilevels.com/>

Abeythilake, U. (2023, 30 Nisan). *Todo List App with Room Database, Kotlin MVVM architecture*. <https://www.linkedin.com/pulse/todo-list-app-room-database-kotlin-mvvm-architecture-abeythilake>

Aleksandrov, S. ve Vulović, S. (2018). Creating an android weather forecast application in the android studio. In *7th International Scientific Conference Technics and Informatics in Education, Serbia*. [https://scholar.google.com/scholar?hl=tr&as\\_sdt=0%2C5&q=Creating+an+android+weather+forecast+application+in+the+android+studio.+&btnG=](https://scholar.google.com/scholar?hl=tr&as_sdt=0%2C5&q=Creating+an+android+weather+forecast+application+in+the+android+studio.+&btnG=)

Bqlgn. (2024, 18 Mayıs). *Rainy free icon*. Flaticon. [https://www.flaticon.com/free-icon/rainy\\_3314005](https://www.flaticon.com/free-icon/rainy_3314005)

Braun, M. (2022, 17 Ağustos). *Celebrating 5 years of Kotlin on Android*. <https://android-developers.googleblog.com/2022/08/celebrating-5-years-of-kotlin-on-android.html>

Bellini, A.-C., & Butcher, N. (2021, 28 Temmuz). *Jetpack Compose is now 1.0: announcing Android's modern toolkit for building native UI*. <https://android-developers.googleblog.com/2021/07/jetpack-compose-announcement.html>

Bhadoria, D.-S. (2023, 11 Ocak). *Android MVVM — how to use MVVM in android example ?*. [https://medium.com/@dheerubhadoria/android-mvvm-how-to-use-mvvm-in-android-example-7dec84a1fb73#:~:text=Model%2DView%2DViewModel%20\(MVVM,with%20data%2Ddriven%20user%20interfaces](https://medium.com/@dheerubhadoria/android-mvvm-how-to-use-mvvm-in-android-example-7dec84a1fb73#:~:text=Model%2DView%2DViewModel%20(MVVM,with%20data%2Ddriven%20user%20interfaces).

- Bharti, V. (2023, 14 Temmuz). *Exploring Coil: A Powerful Image Loading Library for Android*. <https://medium.com/@awesomeAndroid/exploring-coil-a-powerful-image-loading-library-for-android-a5d12d96e5f6>
- Freepik Company. (2024, 18 Mayıs). *Home page*. Flaticon. <https://www.flaticon.com/>
- Google. (2024, 18 Mayıs). *Build better apps faster with Jetpack Compose*. <https://developer.android.com/develop/ui/compose>
- Google. (2023, 1 Mart). *Kotlin overview*. <https://developer.android.com/kotlin/overview>
- Google. (2023, 1 Kasım). *Build location-aware apps*. <https://developer.android.com/develop/sensors-and-location/location>
- Google. (2023, 1 Mayıs). *Run apps on the Android Emulator*. <https://developer.android.com/studio/run/emulator>
- Google. (2024, 3 Ocak). *Support different pixel densities*. <https://developer.android.com/training/multiscreen/screendensities>
- Hackborn, D. (2011, 27 Temmuz). *New Tools For Managing Screen Sizes*. <https://android-developers.googleblog.com/2011/07/new-tools-for-managing-screen-sizes.html>
- Hall S. (2017, 17 Mayıs). *Kotlin is now an officially-supported programming language for Android*. <https://9to5google.com/2017/05/17/kotlin-is-now-an-officially-supported-programming-language-for-android/>
- Howarth, J. (2023, 4 Aralık). *Time Spent Using Smartphones (2024 Statistics)*. <https://explodingtopics.com/blog/smartphone-usage-stats#top-smartphone-stats>
- JetBrains. (2024, 23 Nisan). *FAQ*. <https://kotlinlang.org/docs/faq.html>
- JetBrains. (2023, 27 Ekim). *Kotlin for Android*. <https://kotlinlang.org/docs/android-overview.html>

- JetBrains. (2024, 17 Mayıs). *Coroutines guide*. <https://kotlinlang.org/docs/coroutines-guide.html>
- Kurcheuskay, V. (2023, 15 Kasım). *Compose vs XML in Android Development: Choosing the Right UI Toolkit*. <https://hackernoon.com/compose-vs-xml-in-android-development-choosing-the-right-ui-toolkit>
- Mehwish. (2024, 18 Mayıs). *Sunrise free icon*. Flaticon. [https://www.flaticon.com/free-icon/sunrise\\_3920846](https://www.flaticon.com/free-icon/sunrise_3920846)
- Mehwish. (2024, 18 Mayıs). *Sunset free icon*. Flaticon. [https://www.flaticon.com/free-icon/sunset\\_3920949](https://www.flaticon.com/free-icon/sunset_3920949)
- Muntenescu, F. (2020, 29 Ekim). *Fewer crashes and more stability with Kotlin*. <https://medium.com/androiddevelopers/fewer-crashes-and-more-stability-with-kotlin-b606c6a6ac04>
- Mbano, U. (2022, 10 Mayıs). *What are the benefits of using Android's Jetpack Compose?*. <https://medium.com/dvt-engineering/what-are-the-benefits-of-using-androids-jetpack-compose-9d962fed0109>
- Novac, O. C., Mureşan, I. I., Novac, M. C., Oproescu, M., Gordan, C. ve Bujdosó, G. (2019, Haziran). Development of a weather application in android and IOS operating systems. In *2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)* (pp. 1-5). IEEE. <https://www.doi.org/10.1109/ECAI46879.2019.9042141>
- OpenWeatherMap1. (2024, 18 Mayıs). *One Call API 3.0*. <https://openweathermap.org/api/one-call-3>
- OpenWeatherMap2. (2024, 18 Mayıs). *Geocoding API*. <https://openweathermap.org/api/geocoding-api>
- OpenWeatherMap3. (2024, 18 Mayıs). *Weather icons*. <https://openweathermap.org/weather-conditions#Icon-list>
- riajulislam. (2024, 18 Mayıs). *Wind free icon*. Flaticon. [https://www.flaticon.com/free-icon/wind\\_9231941](https://www.flaticon.com/free-icon/wind_9231941)

- StatCounter. (2024, 18 Mayıs). *OS market share mobile worldwide*.  
<https://gs.statcounter.com/os-market-share/mobile/worldwide/#monthly-202404-202404-bar>
- Srivastava, A., Verma, R. ve Srivastav, V. (2020). *Weather Finder: An Application for weather forecasting* (No. 4823). EasyChair.  
[https://scholar.google.com/scholar?as\\_q=Weather+Finder%3A+An+Applicat+ion+for+weather+forecasting&as\\_occt=title&hl=en&as\\_sdt=0%2C31](https://scholar.google.com/scholar?as_q=Weather+Finder%3A+An+Applicat+ion+for+weather+forecasting&as_occt=title&hl=en&as_sdt=0%2C31)
- Sapundzhi, F. ve Mladenov, M. (2022). An Android-based mobile application giving information for weather in real-time. *Bul. Chem. Commun.*, 54, 89-91.  
<https://doi.org/10.34049/bcc.54.B1.0455>
- Turner, A. (2024, 13 Mart). *NUMBER OF MOBILE PHONE & SMARTPHONE USERS*.  
<https://www.bankmycell.com/blog/how-many-phones-are-in-the-world>
- Türk Dil Kurumu. (2024, 18 Mayıs). *GÜNCEL TÜRKÇE SÖZLÜK*.  
<https://sozluk.gov.tr/>
- Those Icons. (2024, 18 Mayıs). *Humidity free icon*. Flaticon.  
[https://www.flaticon.com/free-icon/humidity\\_483552](https://www.flaticon.com/free-icon/humidity_483552)
- Vector Portal. (2024, 18 Mayıs). *Temperature free icon*. Flaticon.  
[https://www.flaticon.com/free-icon/temperature\\_11813568](https://www.flaticon.com/free-icon/temperature_11813568)
- Vasava, K. (2023, 28 Kasım). *Retrofit in Android*.  
<https://medium.com/@KaushalVasava/retrofit-in-android-5a28c8e988ce#:~:text=Retrofit%20is%20a%20powerful%20and,developed%20Dagger%2C%20Okhttp%2C%20etc.>
- Zamboni, J. (2020, 5 Aralık). *Countries That Use Celsius*. [Countries That Use Celsius | Sciencing](#)