

**T.C.**  
**SÜLEYMAN DEMİREL ÜNİVERSİTESİ**  
**MÜHENDİSLİK FAKÜLTESİ**

**ŞEHİR REHBERİ MOBİL UYGULAMASI**

**FARUK PINAR**  
**BUĞRA ÖZTÜRK**

**DANIŞMAN**  
**Dr.Öğr.Üyesi Gül Fatma TÜRKER**

**LİSANS TEZİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
**ISPARTA -2021**

© 2021 [Faruk PINAR-Buğra ÖZTÜRK]

## TEZ ONAYI

### ŞEHİR REHBERİ MOBİL UYGULAMASI

**Faruk PINAR ve Buğra ÖZTÜRK** tarafından hazırlanan bu tez çalışması aşağıdaki jüri üyeleri önünde Süleyman Demirel Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü'nde **LİSANS TEZİ** olarak başarı ile savunulmuştur.

**Danışman**

**Dr.Öğr.Üyesi Gül Fatma TÜRKER**  
Süleyman Demirel Üniversitesi

## **ETİK BEYANI**

Süleyman Demirel Üniversitesi Lisansüstü Eğitim Enstitüsü tez yazım kurallarına uygun olarak ve bilimsel ahlak ve geleneklere aykırı düşecek bir yol ve yardıma başvurmaksızın hazırladığım bu tez çalışmasında;

Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi, tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu, tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi, kullanılan verilerde ve ortaya çıkan sonuçlarda herhangi bir değişiklik yapmadığımı, bu tezde sunduğum çalışmanın özgün olduğunu, tezimle ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçlara katlanacağımı bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

09/06/2021

Öğrencinin Adı SOYADI

Faruk PINAR

Buğra ÖZTÜRK

İÇİNDEKİLER.....	i-ii
ÖZET.....	iii
ABSTRACT.....	iv
TEŞEKKÜR.....	v
ŞEKİLLER DİZİNİ.....	vi
SİMGELER VE KISALTMALAR DİZİNİ.....	vii
1.GİRİŞ.....	1
1.1. Tez Hakkında Bilgilendirme Ve Tezin Amacı.....	1
2.MOBİL UYGULAMA.....	2
2.1. Mobil Uygulama Nedir?.....	2
2.2. Mobil Uygulamaların Geçmişi.....	3
2.3. Mobil Uygulama Geliştirme.....	6
2.4. Mobil Uygulama Geliştirme Süreci.....	8
3. MOBİL UYGULAMA GELİŞTİRME YAŞAM DÖNGÜSÜ.....	8
3.1. Yaşam Döngüsü.....	8
3.1.1. Planlama.....	10
3.1.2. Analiz Planlama.....	10
3.1.3. Tasarım.....	11
3.1.4. Gerçekleştirme.....	12
3.1.5. Test.....	12
3.1.6. Bakım.....	12
4. MATERYAL VE YÖNTEM.....	13
4.1. Android Studio Nedir ? .....	13
4.1.1. Splash screen nedir ? .....	14
4.1.2. Neden splash screen kullanmalıyız ? .....	14
4.1.3. Component nedir ? .....	15
4.1.4. Layout nedir ? .....	15
4.1.5. Linear layout.....	16
4.1.6. Relative layout.....	16
4.1.7. GridLayout.....	16
4.1.8. RecyclerView nedir ? .....	16
4.1.9. Navigation drawer nedir ? .....	17

4.1.10. API nedir ?.....	18
4.1.11. Uygulamamızın gezi rehberi tasarımı.....	20
4.1.12. Uygulamamızın haberler kısmı tasarımı ve webview.....	21
4.1.13. Uygulamamızın ulaşım kısmı tasarımı ve tablayout.....	22
5. SONUÇ VE ÖNERİLER.....	23
KAYNAKLAR.....	24
ÖZGEÇMİŞ.....	25

## **ÖZET**

**Lisans Tezi**

### **ŞEHİR REHBERİ MOBİL UYGULAMASI**

**Faruk PINAR**

**Buğra ÖZTÜRK**

**Süleyman Demirel Üniversitesi  
Mühendislik Fakültesi  
Bilgisayar Mühendisliği Bölümü**

**Danışman:Dr.Öğr.Üyesi Gül Fatma TÜRKER**

Mobil uygulama, mobil cihazlara (akıllı telefon ve tablet) yönelik özel kodlar ve tasarımlar ile yapılan yazımlara verilen addır. Mobil uygulamalar, dünya çapında iOS ve Android işletim sistemi kullanarak yapılır. Planladığınız ve tasarladığınız her proje için mobil uygulama geliştirme yapmanız mümkün. Mobil uygulamaların web sitelerinden daha hızlı ve kolay kullanıma sahip olması kullanıcıları mobil uygulama kullanmaya teşvik ediyor. Kısacası mobil uygulamalar kullanıcılara web sitelerinden daha cazip geliyor.

Bu konuları göz önünde bulundurarak Android işletim sistemi için şehir rehberi mobil uygulaması yapmaya karar verdik. Bu uygulamanın amacı Isparta’da yaşayan insanların ya da Ispartayı ziyaret edecek insanların ihtiyaçlarını karşılayabileceği şehir rehberi uygulamasının tasarlanmasıdır.Bu uygulama günlük hava durumu bilgisi , şehirdeki gezilecek yerlerin listesi , yerel gazetedeki haberler , şehirdeki yaklaşan etkinlikler ve bilgileri , şehirdeki taksi durakları ve otobüs hatlarının bilgilerini içerecektir.

**Anahtar Kelimeler:Isparta,Şehir,Şehir Rehberi**

**2021,X SAYFA**

## **ABSTRACT**

**B.Sc.Thesis**

### **CITY GUIDE MOBILE APP**

**Faruk PINAR**

**Buğra ÖZTÜRK**

**Süleyman Demirel University  
Faculty Of Engineering  
Department Of Computer Engineering**

**Supervisor:Asst.Prof.Dr. Gül Fatma TÜRKER**

Mobile application is the name given to writings made with special codes and designs for mobile devices (smartphone and tablet). Mobile apps are made worldwide using iOS and Android operating system. It is possible to develop mobile applications for every project you plan and design. The fact that mobile applications are faster and easier to use than websites encourages users to use mobile applications. In short, mobile applications are more attractive to users than websites. Considering these issues, we decided to make a city guide mobile application for the Android operating system. The purpose of this application is to design a city guide application that can meet the needs of people living in Isparta or people who will visit Isparta. This application will include daily weather information, a list of places to visit in the city, news in the local newspaper, upcoming events and information in the city, taxi stops and bus lines in the city.

**Keywords: City,Isparta,City Guide**

**2021,x sayfa**



## **TEŞEKKÜR**

Tezimizin yürütülmesinde desteğini ve emeğini hiçbir zaman esirgemeyen tez danışmanımız sayın Dr.Öğr.Üyesi Gül Fatma Türker'e teşekkürlerimizi sunarız.

Tezimizin her aşamasında bizi yalnız bırakmayan ailelerimize sonsuz sevgi ve saygılarımızı sunarız.

**Faruk PINAR**  
**Buğra ÖZTÜRK**  
ISPARTA,2021

## ŞEKİLLER DİZİNİ

## Sayfa

Şekil 3.1 Yazılım yaşam döngüsü adımları.....	9
Şekil 3.2 Yazılım geliştirme döngüsü tek yaşam süreci .....	10
Şekil 4.1 Uygulamamızın splash screen ekranı .....	14
Şekil 4.2 Uygulamamızın anasayfa tasarımı .....	15
Şekil 4.3 Uygulamamızın navigation menü tasarımı .....	17
Şekil 4.4 Uygulamamızın kategoriler kısmı tasarımı .....	18
Şekil 4.5 Uygulamamızın hava durumu kısmı tasarımı.....	18
Şekil 4.6 Uygulamamızın etkinlikler kısmı tasarımı .....	19
Şekil 4.7 Uygulamamızın gezi rehberi kısmı tasarımı .....	20
Şekil 4.8 Uygulamamızın haberler kısmı tasarımı .....	21
Şekil 4.9 Uygulamamızın ulaşım kısmı tasarımı.....	22

## **SİMGELER VE KISALTMALAR DİZİNİ**

iOS (iPhone Operating System)

IDE (Integrated Development Environment - Entegre Geliştirme Ortamı)

API (Application Programming Interface - Uygulama Programlama Arayüzü)

SDK(Software Development Kit – Yazılım Geliştirme Kiti)

# 1.GİRİŞ

## 1.1. Tez Hakkında Bilgilendirme Ve Tezin Amacı

Proje tasarımı için Google Android Studio kullanılacak. Android Studio, Google tarafından 16 Mayıs 2013 tarihinde Google I/O etkinliğinde tanıtılan ve o tarihten bu yana Android uygulama geliştirmede kullanılan resmi Entegre Geliştirme Ortamı (IDE) olarak bilinir. Her türlü Android cihazda çalışır. Yüksek kalite, verimli sonuçlar elde edilen uygulamalar geliştirmek için en hızlı araçları sağlar.Android Studio, IntelliJ IDEA'ya dayalı olup Android uygulamaları geliştirmek için özel olarak tasarlanmıştır. Android Studio, akıllı bir kod düzenleyici ve hata ayıklayıcı, performans analiz araçları, emülatörler ve daha pek çok araç dahil olmak üzere uygulama geliştirmek için kullanıcıların ihtiyacını karşılayabilecek birçok özellik barındırır. Hava Durumu ve Konum gibi içerikler API'ler sayesinde yapılacak API (Application Programming Interface), bir uygulamaya ait yeteneklerin, başka bir uygulamada da kullanılabilmesi için, yeteneklerini paylaşan uygulamanın sağladığı arayüzdür.

Mobil uygulama herkes tarafından kolayca kullanılabilecek ve basit bir arayüze sahip olacak. İnsanlar şehir hakkındaki haberlerden ve etkinliklerden sürekli haberdar olacak,şehrin hava durumu bilgisine günlük olarak ulaşabilecekler, aradıkları yerleri konumlarına göre en yakın olanları listeleyebilecekler ,şehirde gezilebilecek yerleri uygulama sayesinde kolayca bulup bu yerler hakkında bilgi alabilecekler,şehirdeki taksi duraklarının adres ve telefon bilgilerine , otobüs hatları hakkındaki bilgilere erişebilecekler.

## **2. MOBİL UYGULAMA**

### **2.1. Mobil Uygulama Nedir?**

Bilgisayarların hayatımıza girmesinden buyana yaşanmış ve insan hayatının içine giren en önemli gelişmelerden biriside neredeyse herkesin kullandığı akıllı telefonlar, tablet bilgisayarlar olmuştur. Bu cihazlar genel olarak mobil cihazlar olarak adlandırılmaktadır ve bu cihazlarda çalışması için tasarlanmış ve geliştirilmiş yazılımlara "Mobil Uygulama" denilmektedir. Mobil uygulamalar mobil cihazların kullandığı "Mobil İşletim Sistemi" ne uygun olarak yazılmaktadır ve bunlarda her işletim sisteminin sağladığı uygulama marketlerinden indirilebilmektedir.

Günümüzde bir çok mobil işletim sistemi bulunmaktadır. Ancak bunlardan iki tanesi son 2-3 yılda dünyada ses getirmiş ve çok sayıda mobil cihaz üreticisi ve mobil cihaz kullanıcısı tarafından kullanılmaktadır. Apple firmasının geliştirdiği ve kendi ürettiği mobil cihazlarında kullandığı iOS işletim sisteminde çalışacak uygulamaları iPhone, iPod ve iPad cihazlarına Apple App Store'dan, Google firmasının geliştirdiği ve birçok mobil cihaz üreticisi, mobil cihaz kullanıcısı tarafından kullanılan android işletim sistemini kullanan cihazlar için Google Play Store'dan mobil uygulama edinebilirsiniz. Mobil uygulamaları bu uygulama marketleri aracılığı ile doğrudan mobil cihazınıza indirme imkânıyla olabildiği gibi, eğer izin verilmiş ise önce bilgisayarınıza indirip daha sonra indirdiğiniz mobil uygulamaları, mobil cihazınıza aktarip kurulumunu gerçekleştirebilirsiniz. Uygulama marketlerinde bulunan mobil uygulamalar sundukları hizmetin kalitesine, içeriğine, özelliğine ve uygulama sahibinin isteğine bağlı olarak ücretli ya da ücretsiz olarak sunulabilirler. Ücretsiz sunulan mobil uygulamaları uygulama marketinden herhangi bir ücret ödenmeden mobil cihaza indirilip, kurulumu yapılabilir, dilediğinde ise uygulama cihazdan kaldırılabilir ve tekrar yüklenebilir. Eğer mobil uygulama ücretli ise sisteme tanımlanmış olan kredi kartı ile satın alındıktan sonra cihaza yüklenebilir, sonrasında ise dilediğiniz kadar kaldırıp yükleyebilirsiniz. Bazı internet bağlantısı üzerinden ücretsiz mesajlaşma ve konuşma imkânı sunan mobil uygulamalar da yıllık ya da aylık olarak dönemlik ücretlendirme sistemi üzerinden sunulmuş olabilirler.

Apple CEO'su Tim Cook Haziran 2013'te yapılan bir tanıtımda App Store'da 900 binden fazla aktif mobil uygulama yer aldığından söz etmiştir. Ekim ayı sonunda düzenlenen tanıtımda ise bu sayısının bir milyona ulaştığı haberini mobil uygulama sektörüne duyurmuştur. Ayrıca 60 milyona yakın uygulama indirme sayısına ulaşılmıştır. Google Play Store'da ise toplam uygulama sayısı uzun süredir 1 milyonun üzerindedir. Burada da indirilme sayısı kısa bir süreçte olmasına rağmen 50 milyonun üzerine çıkmıştır. Bu da mobil sektörün ne kadar hızlı geliştiğinin bir göstergesidir. Uygulama marketlerinde bulunan mobil uygulamalar, aranan uygulamanın bulunabilmesini kolaylaştırmak için kategorilere ayrılmıştır. Mobil uygulama kategorilerine; haberler, oyunlar, müzik, bankacılık, fotoğrafçılık, spor, sosyal ağlar, eğlence gibi bir çok örnek verebiliriz. Bu kadar kategori arasından aranan mobil uygulama kolayca bulunabilir. Mobil uygulamalar ilk başlarda standart gereksinimler için geliştirilmiş olsa da genellikle hayatı kolaylaştırmaya ve pratikleştirmeye yöneliktirler. Mobil cihazlar uygulama marketlerinde bulunan oyunlar sayesinde bir eğlence merkezine dönüşebilmektedir. Ayrıca mobil uygulamalar, internetteki bir web sitesine göre çok daha az veri kullanarak aynı bilgiye ulaşmayı sağlaması gerekmektedir. Çünkü mobil cihazlarında her ne kadar yaygınlaşsa da kotalı internet bağlantısına sahip kullanıcılar çoğunluktadır ve onlar için de daha bu yapı avantaj yaratmaktadır. Son zamanlarda üretilen bir çok mobil cihazda GPS (Global Positioning System, Küresel Yer Belirleme Sistemi) özelliği bulunmaktadır. Navigasyon uygulamaları sayesinde dünyanın herhangi bir yerinde yolunuzu kaybetmeden istediğiniz yere gidebilirsiniz. Mobil Uygulamalar, mobil cihazlarda bulunan donanımsal özelliklerin alternatif şekillerde de kullanılmasına olanak sağlayabilmektedir. Buna örnek olarak, mobil cihazın fotoğraf çekmek için kullandığı LED flaşını bir başka uygulama ile aktive edip karanlık bir ortamda fener olarak kullanılması sağlanabilir. Araştırmalar akıllı telefon kullanımının dünyada giderek arttığını, özellikle tıp öğrencileri ve doktorlar tarafından çok tercih edildiğini, Amerika'da doktorların %81 oranda akıllı telefon kullandığını göstermektedir [1].

Sonuç olarak, mobil uygulamalar hayatı kolaylaştırabilir, internet bağlantısı üzerinden her türlü şekilde iletişim kurulmasını sağlayabilir, sosyal ağlarda aktif olunmasını sağlayabilmektedir. Artık bilgisayarlarda kullanılan ve ihtiyaç olan uygulamaların mobil versiyonunun olması her an bu uygulamaya ulaşılabilmesi anlamına gelmektedir. Mobil uygulamalar, bu özellikleri sayesinde insanların hayatında çok önemli bir rol oynama yolunda ilerlemektedir.

## 2.2. Mobil Uygulamaların Geçmiři

Cep telefonlarının hayatımıza girmesi ile birlikte ilk mobil uygulamalar da ortaya çıkmıř oldu. İlk mobil uygulamalar ile sadece arama yapmak, mesaj göndermek dışında başka bir řey yapılamıyordu. Bunda önemli bir etken de ilk çıkan mobil iřletim sistemlerinin oldukça ilkel ve donanımları tam olarak kullanmaya izin vermeyen yapıda olmasıydı. Mobil uygulamaların basit ve ilkel yapıda olması da büyük ölçüde, cep telefonlarının yeterli hafızaya sahip olmamasından kaynaklanıyordu. Bu düşük hafıza mobil cihazlar üzerinde rehber kaydedilen kiři sayısını ve cihaza kaydedilmek istenen veri miktarını da oldukça kısıtlamaktaydı. Mobil uygulamaların ilk çağlarında oldukları bu dönemde, mobil uygulamalar cihazla birlikte gelmekteydi ve bu uygulamaları kaldırma veya üzerine başka bir mobil uygulama yükleme olanağı da yoktu. Bu mobil uygulamalar gün geçtikçe hem iř sektöründe hem de günlük hayatta, yařamı daha kolay bir hale getirmeye başladı. Örneğın; yeni çıkan mobil cihazlar üzerinde daha gelişmiş rehber uygulamaları ortaya çıktı. Bu mobil uygulamalar; iletişim kurulmak istenen yüzlerce kiřinin numarasını tek bir mobil cihazda kolayca saklayıp, istenen kiřiye ait telefon numarasının çok kısa sürede bulmasına olanak sağlayabiliyordu. Bununla birlikte mobil cihazlarda sadece takvim, hesap makinesi, zil sesi editörü gibi çok az sayıda ve basit mobil uygulamalar da bulunmaktaydı. Bu uygulamalar sadece mobil cihazları üreten firmalar tarafından yapılıyordu ve bu mobil uygulamaların kaynaklarını kimseyle paylařmıyorlardı. Bu nedenle dışarıdan bir geliştirme yapılması gibi bir durumda söz konusu değildi. Mobil cihazlarda yapılan ilk mobil uygulama telefonlara yüklü olarak gelen oyun uygulamaları oldu. Bu oyun herkesin yakından tanıdığı ve bildiğı tetris oyunu idi. İlk kez bir cep telefonunda mobil oyun bulunması bu sektörde yeni bir çağ başlattı diyebiliriz. Bunun ardından snake gibi birçok klasik oyun da mobil uygulama olarak mobil cihazlarda yer edinmeye başladı. Bu oyunlar sadece vakit geçirme maksadıyla mobil cihazlara konmuřtu. Ancak mobil cihaz kullanıcıları, zaman geçtikçe daha çok özellik ve daha çok oyun talebinde bulunmaya başladılar. Cep telefonlarının ve mobil cihazların yaygınlařmasından kısa bir süre sonra java tabanlı çok daha esnek olan mobil iřletim sistemleri ortaya çıktı. Bunlardan en önemlisi ve günümüzde çok daha gelişmiş versiyonları bulunan Nokia řirketi tarafından geliştirilmiş “Symbian” iřletim sistemidir. Bu iřletim sistemleri eskisinden daha gelişmişti ve gelişmiş mobil uygulamalara kapılarını açmıřtı. Yeni java tabanlı mobil iřletim sistemlerinden istemediğimiz bir mobil uygulamayı

silebiliyor, hatta sınırlı sayıda mobil uygulama içeren mobil marketlerden mobil cihazımıza uygun olan başka bir mobil uygulamayı da internet aracılığı ile indirip kurabiliyorduk. Yeni ortaya çıkan java tabanlı işletim sistemleri ile birlikte mobil cihazların depolama alanları oldukça arttı ve bu depolama alanları hafıza kartı ile daha da genişletebilme özelliği kazandı. Mobil cihazlardan internet bağlantıları artık daha hızlıydı ve mobil cihazlarda internet bağlantısı ile yapılabileceklerinin sayısı oldukça arttı. Bu sayede web sayfaları üzerinde rahatça gezinmeyi ve internet üzerinden dosya alıp göndermeyi sağlayan çok daha gelişmiş mobil internet tarayıcıları ortaya çıktı. 90'lı yıllarda çok popüler olan mesajlaşma programlarının mobil uygulama versiyonları yapılarak sosyal ağlar mobil ortama taşınmaya başladı. Mobil cihazın kendi dahili hafızası ve harici takılabilen hafıza kartlarını etkin bir şekilde kullanabilmeyi sağlayan mobil dosya yöneticisi uygulamaları yapıldı. Yeni mobil uygulamalardan en çok ilgi çekenlerden biri de mobil cihaz üzerinde gelen medya oynatıcı uygulamalardı. Mobil medya oynatıcı uygulamalar sayesinde, internet üzerinden ya da mobil cihazınızın hafızasında kayıtlı olan müzik ve video dosyalarını oynatılabiliyordu. Mobil uygulama kategorisinde ayrı bir yer olan oyunlar da gelişmeye ve yaygınlaşmaya başladı. Bu mobil uygulamalar sayesinde, mobil cihazlar eğlence amaçlı kullanabilen cihazlar haline geldi. Mobil uygulamaların çoğalması ve çeşitlenmesi ile birlikte mobil cihazların da gelişmesi sayesinde kullanıcıların mobil cihazlardan, özellikle cep telefonlarından beklentileri oldukça arttı. Beklentiler arttıkça her şirketin her kullanıcıya isteğini karşılayabilecek kadar mobil uygulama geliştirme şansı kalmadı bu nedenle mobil uygulama mağazaları ortaya çıktı. Bu mobil uygulama mağazalarının ortaya çıkmasıyla birlikte şirketler kendi mobil işletim sistemlerine uygun olmak kaydıyla, diğer geliştiricilerin ve kullanıcıların mobil uygulama geliştirilmesine olanak sağladı. Mobil uygulama geliştiricilerinin kendi isteklerine ya da tüketicilerin taleplerine göre mobil uygulama geliştirmeye başlaması ile birlikte, son 5-6 yıl içerisinde mobil uygulama sayısında patlama yaşandı. Bu patlamanın sebebi olarak mobil uygulama geliştiricilerinin bu süre içerisinde sayısının oldukça hızlı bir şekilde artmasını ve bunun sebebi olan yeni nesil işletim sistemlerini gösterebiliriz. Mobil uygulamaların gelişmesinde en fazla pay sahibi olan işletim sistemleri Apple şirketinin 2007 yılında iPhone ile birlikte tanıttığı iOS mobil işletim sistemi ile birlikte, 2005 yılından beri geliştirilme aşamasında olan, 2008 yılında ilk kez bir mobil cihazla birlikte piyasaya sunulan android mobil işletim sistemidir. Günümüzde sayıları hızla artmakta olan mobil



cihazlar, mobil uygulama sektörünün de büyümesine yol açtı. Her mobil platforma özel yüz binlerce uygulama yapıldı, bununla beraber popüler olan uygulamaların her platformda bulunup, aynı işlevleri gerçekleştirebilecek şekilde geliştirilmeleri sağlandı. Sonuç olarak özetlemek gerekirse, mobil uygulamalar son birkaç yılda çok büyük bir hızla gelişerek milyonlarca insanın hayatını değiştirdi. Günlük hayatı kolaylaştıran, iş adamlarına kolaylık sağlayan, sosyal paylaşım sitelerine heran bağlı olabilme imkânı sunarak her şeyden anında haberdar olmamızı ve boş zamanımızı daha eğlenceli geçirmemizi sağlayan mobil uygulamalar sayesinde teknolojiye daha bağlı olarak yaşamaya devam ediyoruz.

### **2.3. Mobil Uygulama Geliştirme**

Günümüzde kullanıcıların sahip oldukları cep telefonları ve tablet bilgisayarlardan oluşan mobil cihazların satılmasını sağlayan en önemli iki faktör vardır. Bunlardan ilki, hangi işletim sistemine sahip olduğudur. İkincisi ise, bu işletim sisteminin sahip olduğu uygulamaların bulunduğu markette, kaç çeşit ve ne kadar kaliteli mobil uygulama bulunduğudır. Türkiye'de bu günlerde mobil uygulama pazarında üç adet mobil işletim sistemi en yaygın olarak kullanılmaktadır. Bunlar; Android, iOS işletim sistemleri ile her ne kadar pazara yeni girmiş olsa da Windows Phone 8 işletim sistemidir. Herhangi bir mobil cihaza ve kullandığı işletim sistemine yönelik, basit ya da daha karmaşık yapıda bir yazılım tasarlama ve oluşturma işlemine mobil uygulama geliştirme denilmektedir. Bu geliştirme işlemi her işletim sistemine göre farklılık gösterir. En yaygın işletim sistemleri için bahsetmek gerekirse, android işletim sistemine uygun bir mobil uygulama geliştirmek için iyi seviyede java programlama dili bilgisi gerekirken, iOS işletim sistemi için objective-c dilini bilmek gerekir. Her bir mobil işletim sistemine ait geliştirici dilleri ve geliştirme ortamları farklıdır. Mobil uygulama geliştirmek istendiğinde genel olarak izlenebilecek 3 farklı yol bulunmaktadır. Bunlardan birincisi "Native" uygulama geliştirmedir. Native uygulama geliştirmek için belli bir işletim sistemine belli bir programlama dili ile geliştirme yapılması gerekmektedir. Bu şekilde uygulamanız daha güvenli ve daha performanslı olmaktadır. Diğer bir yol da "Web" tabanlı bir mobil uygulama yazarak sadece tek seferde tüm platformlarda çalışabilecek bir uygulama geliştirebilirsiniz. Bu sayede doğrudan mobil cihazınızın tarayıcısı aracılığı ile uygulamayı kullanabilirsiniz. Masraftan ve fazla zaman harcamaktan kaçınmak için ideal bir

yöntemdir ancak sürekli internet bağlantısı gerektirmesi ve fazla performanslı çalışmaması dezavantajlarından birkaçıdır. Üçüncü yol ise "Hibrit" adıyla belirtilen yani melez uygulama geliştirme yöntemidir. Bu yolu kullanmak için Native kodlamanın içerisine web tabanlı bir sisteme erişim sağlanarak iki teknolojinin birden aynı anda kullanılması sağlanır. Mobil uygulama geliştirmeye başlamak için öncelikle uygulamanın hangi mobil platformda çalışacağına karar vermek gerekir. Sadece android ortamında mı çalışacak yoksa sadece iOS ortamında mı ya da her ikisinde mi olacağının kararının verilmesi gerekmektedir. Uygulamanızın sadece belirli bir ya da birkaç mobil işletim sisteminde, güvenli ve daha performanslı çalışmasını istiyorsanız native uygulama geliştirmelisiniz. Düşük bir maliyetle her platformda çalışan ve daha çok güncel ve web odaklı bir uygulama geliştirmek istiyorsanız web tabanlı bir uygulama geliştirmek çok daha avantajlı olacaktır, ancak son zamanlarda popüler işletim sistemlerinin “native olmayan” uygulamalara desteğini azaltması, gelecekte sorun yaratabilir. Android, iOS gibi mobil platformlara uygulama geliştirmek için bilgisayarınıza birtakım yazılımlar yüklemeniz gerekir. Bunlardan biri SDK (Software Development Kit) adıyla bilinen yazılım geliştirme araçlarıdır. Bu araçlar, uygulama geliştirmenize yardımcı ve kolaylaştırıcı bir çevrede yaratır. Uygulama geliştirirken kullanacağınız kütüphaneler, debugger (hata bulucu) ve gerekli başka her şeyi içinde barındırarak geliştiriciye her aşamada yardımcı olur. Mobil cihaz simülatörleri aracılığıyla gerçek bir cihaza gerek duymadan geliştirdiğiniz uygulamaları test etme imkânı sunar. Android için gerekli olan ve en yaygın olarak kullanılan geliştirme ortamı Eclipse’dir. Android platformuna uygulama geliştirirken Windows, MacOS ya da Linux tabanlı bir işletim sistemi kurulu olan herhangi bir bilgisayar üzerinde çalışabilirsiniz. iOS içinse durum biraz farklıdır. iOS için gerekli olan Xcode ortamı sadece MacOS işletim sistemlerinde çalışmaktadır. iOS uygulama geliştirmek için Apple MacOS içeren bir bilgisayara sahip olmanız gerekmektedir. Mobil uygulama geliştirdikten sonra yapılacak ilk şey, o uygulamanın dağıtımını sağlamaktır. Bunun için de Google Play Store, Apple AppStore, Windows Phone Store gibi mobil uygulama marketleri oluşturulmuştur. Bu mobil uygulama marketleri aracılığıyla geliştirdiğiniz uygulamalarınızı bedava olarak indirmeye sunabilir ya da isterseniz ücretli olarak sunarak, indirme başına para kazanabilirsiniz. Tabi ki bu marketler, ücretin belirli bir kısmını keserek geri kalanın size ödenmesini sağlıyorlar.

## 2.4. Mobil Uygulama Geliştirme Süreci

Her mobil uygulama projesi aynı zamanda bir yazılım projesidir. Bu nedenle mobil uygulama geliştirme süreçleri yazılım geliştirme süreçleri ile benzerlik göstermektedir. Mobil uygulama geliştirme esnasında çıkabilecek sorunlarla daha kolay başa çıkabilmek adına aşağıda bulunan mobil yazılım geliştirme süreci doğrultusunda uygulama geliştirilmiştir.

- İhtiyaç olan uygulamayı detaylı bir şekilde anlayabilmek için yapılması istenilen uygulamaya ait ihtiyaçlar bir doküman olarak işin sahibinden alınır,
- Gerekli analiz çalışması yapılır,
- Gereksinim dokümanı oluşturulur,
- Gereksinim dokümanında işin kapsamı ve modülleri belirlenir,
- İşin sahibinden kapsam ve modüller için onay alınır,
- Uygulama geliştirme süreci başlar,
- Tasarım çalışması yapılır,
- Tasarım onayından sonra geliştirme süreci başlar,
- Kodlama tamamlandıktan sonra müşteriye sunulur,
- Müşteriden onay alınır,
- Uygulama Test Case'leri oluşturulur,
- Planlanan testler yapılır,
- Testlerin yapılmasını takiben müşteriden onay alınır,
- Uygulama marketlerinde yayınlanır,
- Uygulama bakım ve güncelleme süreci başlar.

## 3. MOBİL UYGULAMA GELİŞTİRME YAŞAM DÖNGÜSÜ

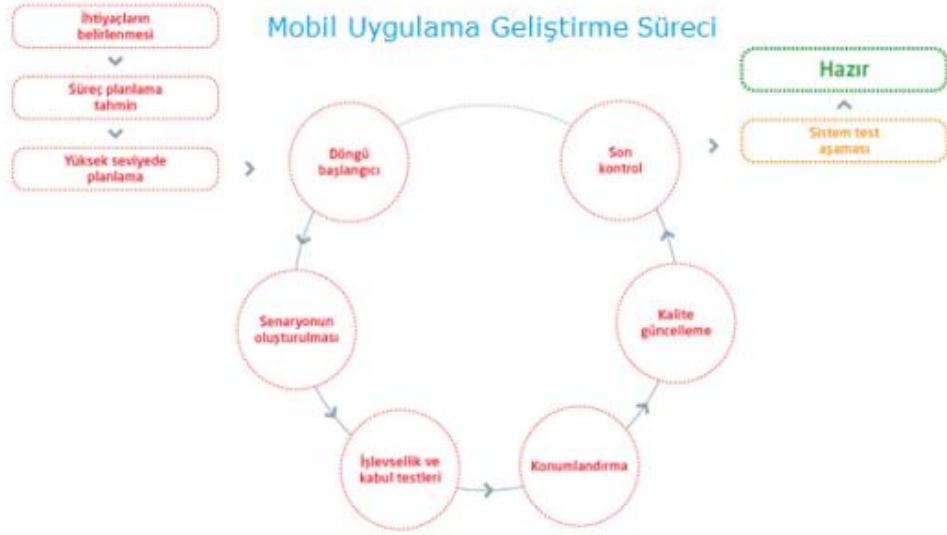
### 3.1. Yaşam Döngüsü

Geliştirdiğiniz bir mobil yazılım projesinin planlamasından başlayarak teslimatına kadar geçirmiş olduğu bütün aşamalara ve bu aşamalardan oluşan döngüye, mobil uygulama geliştirme yaşam döngüsü denir. Bir mobil uygulama projesinin geliştirilmesi, sadece kodlamadan oluşmamaktadır. Basitçe bir proje geliştirilirken dahi projenin planlama, analiz, tasarım, üretim ve test aşamaları yer almaktadır ve almalıdır. Bu aşamaların bir kere gerçekleştirilmesinden sonra proje tamamlanmayabilir. Bu aşamaların bir döngü halinde düşünülmesi gerekmektedir. Proje tamamlandıktan sonra gelecek olan istekler, hata düzeltmeleri, projeye eklenecek yeni modüller gibi konular için bu süreç devam etmektedir. Bu döngüye yazılım geliştirme yaşam döngüsü adı verilmektedir.



Şekil 3.1. Yazılım yaşam döngüsü adımları

Yazılım geliştirme süreci, zamanlamaya dayalı ve içerik olarak bölünmüş aşamalardan oluşmaktadır. Bu sayede yazılım planlı bir şekilde geliştirilmektedir. Yazılım işlevleri ile ilgili gereksinimler sürekli olarak değiştiği ve genişlediği için, söz konusu aşamalar sürekli bir döngü biçiminde ele alınır. Şekil 3.1'de de gösterildiği üzere döngü içerisinde her hangi bir aşamada geriye dönmek ve tekrar ilerlemek söz konusudur. Bir mobil uygulama projesi aslında bir yazılım projesi olarak değerlendirilmektedir. Yazılım projelerinde de planlama, analiz, tasarım, üretim ve test aşamalarının tamamlandığını ve proje içerisinde bu süreçler tamamlandıktan sonra hiçbir hata ve yeni istek olmadığını varsayalım. Bu noktada devreye projenin bakım aşaması girmektedir. Geliştirilen bütün yazılım projelerinin ileride doğabilecek hataları ve yeni istek ve talepleri için bakım sürecinin de yazılım yaşam döngüsüne eklenmesi gerekmektedir. Son haliyle yazılım yaşam döngüsü ele alındığında aşağıdaki süreç karşımıza çıkmaktadır.



Şekil 3.2. Yazılım geliştirme döngüsü tek yaşam süreci

Yazılım yaşam döngüsü içerisindeki bulunan aşamalarda gerçekleştirilen işlemlerin detayları tüm yazılım projesinin oluşmasında geçen adımları göstermektedir. Şekil 3.2 de gerçekleşen adımların bir defa işletilmesinin ilerleyişi gösterilmektedir.

### 3.1.1. Planlama

Yazılım yaşam döngüsünün başlangıç noktasıdır. Temel ihtiyaçların belirlendiği, proje için fizibilite çalışmalarının yapıldığı (maliyetlerin ve sistemin yararlarının tanımlanması) ve proje planlaması gerçekleştirildiği dönemdir.

### 3.1.2. Analiz Planlama

Yazılım yaşam döngüsünün en önemli aşamalarından biri olan analiz sürecinde projenin tüm işlevleri detaylı olarak belirlenir. Bu aşamanın amacı sistemin işlevlerini ve kesin gereksinimleri açıklığa kavuşturmak ve sonucunda bunları belirli bir formatta belgelendirmektir. Bu çalışma müşteri, yazılım mühendisi, sistem analisti, iş analisti, ürün yöneticisi vb. rollerin bir araya geldiği gruplar tarafından yapılabilir. İhtiyaçların net olmadığı durumlarda yazılım mühendisi ve müşteri arasında iletişim ve birlikte çalışmanın çok daha fazla olması gerekir. Çeşitli yazılım geliştirme metodolojilerinde bu aşamada kullanım dokümanları ve test plan dokümanları da oluşturulabilir. Kısaca, analiz sürecinde projenin tüm detayları ortaya çıkartılır.

### 3.1.3. Tasarım

Analiz çalışması sonucu ortaya çıkartılan proje detayları baz alınarak proje gerekli durumlarda bileşenlerine ayrılırlar, proje içerisinde yapılacak işlemler belirlenir ve proje planı oluşturulur. Proje planının yanı sıra tasarım dokümanı da oluşturulmalıdır. Tasarım dokümanında proje bilgileri (amaç, kapsam vs), sistem tasarım bilgileri, tasarım detayları, veri modeli, kullanıcı ara yüz tasarımları, UML diyagramları bulunmaktadır. Tasarım dokümanının amacı, yazılım geliştiricinin yazılımını geliştirirken referans alacağı ve proje sürecinde/sonrasında projeye dahil olacak yeni yazılımcıların projeyi daha kolay anlayabilmesini sağlayacak teknik bir dokümantasyona sahip olması gerekliliğidir. Gereksinimlerin tamamlanmasıyla beraber sistem tasarım aşamasına başlanır. Yazılım ürün tasarımı, müşterinin gereksinim ve isteklerini karşılamak üzere yazılım ürününün özellikleri, yetenekleri, ve ara yüzlerinin belirlenmesi etkinliğidir. İki tür tasarımdan bahsetmek mümkündür: Mimari tasarım ve detaylı tasarım. Mimari tasarım, yazılım modüllerinin genel yapıları ve organizasyon içerisindeki etkileşimleri ile ilgilidir. Sonucunda mimari tasarım dokümanları oluşturulur. Detaylı tasarım aşamasında Mimari tasarım dokümanları genelde revize edilirler. Gereksinimlerin belirlendiği analiz aşaması problemin ne olduğu ile ilgilidir. Unutmamak gerekir ki sistemdeki tüm problemler yazılım ürününün tamamlanması ile çözülmeyecektir. Maalesef çoğu zaman Ne söylemi tasarım kararı olurken Nasıl söylemi de müşterinin gereksinimi olabilmektedir. Bu duruma dikkat etmek gerekmektedir. Yazılım tasarımında kullanılan en önemli tekniklerden birisi soyutlama (Abstraction) dır. Soyutlama, problemlerin çözümlerini kolaylaştırmak için nesnelerin, olayların ve durumların bazı özelliklerin sadeleştirilerek kullanılmasıdır. Problemi basitleştirerek en önemli kısımlarına odaklanmamızı sağlar. Modelleme ise temel tasarım aracı olup statik ve dinamik modellerden bahsetmekten mümkündür. Statik model, programın çalışması sırasında değişmeyen yönlerini ifade etmek için kullanılırken (sınıf ve nesne modelleri), dinamik model programın çalışması sırasındaki işleyişi ifade etmek için kullanılır ( durum ve sıra diyagramları).

#### **3.1.4. Gerçekleştirme**

Planlama, analiz ve tasarımı tamamlanıp yapılacak işlemleri detaylı olarak belirlenen projenin geliştirme aşamasıdır. Bu noktadan sonra herhangi bir analiz işlemi yapılmamalı, proje, tasarım aşamasında oluşturulan planlama çerçevesinde ilerlemelidir. Tasarım aşamasının belirli bir olgunluğa ulaşmasıyla birlikte kodlama aşaması başlar. Müşteriye teslim edilecek ürünü programlama aşamasıdır. Kaliteli, okunabilirliği ve bakımı kolay olan basit kod yazılması her zaman olması gereken bir olgudur. İster bir şirkette çalışın ister bireysel projeler geliştirilsin mutlaka belirli bir kodlama kalite standardına göre (İsimlendirme standartları, yorum satırı kullanımları, tekrar eden kodlar, aşırı benzer işlevler, uzun metotlar vb.) geliştirme yapılmalıdır. Bu sayede sonradan koda yapılan bakımlarda zorluk çekilmemesi sağlanmış olur.

#### **3.1.5. Test**

Yazılım geliştirilmesi tamamlandıktan sonra müşteriye sunmadan önce, test ekibi tarafından beta testlerinin gerçekleştirilmesi aşamasıdır. Kodlama süresince ve kodlama sonrasında yapılan diğer önemli aşama testtir. Erken test et yaklaşımı ile hareket edip, analiz aşamasından itibaren test bakış açısına sahip olmamız hata yapma oranımızı ve maliyetleri (zaman, para, prestij vb.) düşürecektir. Birim testleri, duman testleri, yanlış değer testleri, kabul testleri, kullanım senaryo testleri, yük testleri, kullanıcı kabul testi, yoldan geçen adam testi, test otomasyonu gibi sürece ve duruma göre uygulanabilecek çok farklı kategoride ve derinlikte test türü bulunmaktadır. Bu aşama tamamlandıktan ve varsa hatalar giderildikten sonra proje yayına alınmaktadır.

#### **3.1.6. Bakım**

Proje yayına alındıktan sonra oluşabilecek hataların giderilmesi, yazılımın iyileştirilmesi ve yeni işlevlerin eklenmesi süreçleridir. Bu süreç zarfında kullanıcılardan gelen bilgiler doğrultusunda bu istekler gerçekleştirilmektedir. Tüm test aşamaları tamamlandıktan sonra yazılım ürünün sahaya teslim edilebilir bir versiyonu çıkartılır ve teslim aşaması gerçekleştirilir. Teslim çıktısı olarak ürün tek

başına yeterli değildir. Mutlaka son kullanıcılar için kullanım kılavuzu ve versiyon fark dokümanı oluşturulmalıdır. Teslim ile birlikte bakım aşaması da başlar. Hata giderici, önleyici, altyapıyı iyileştirici, ürüne yeni özellikler ekletici gibi farklı bakım faaliyetleri mevcuttur.

## **4. MATERYAL VE YÖNTEM**

### **4.1. Android Studio Nedir ?**

Android Studio, yazılımcılar için geliştirilmiş bir ortamdır. Bu ortamda yazılımcılar Android işletim sistemine sahip cihazlara uygun uygulamalar oluşturabilmektedir. Bu uygulamaların test süreci de yine Android Studio içinde gerçekleştirilebilir. Android Studio’da programlama dili olarak Java kullanılmaktadır. Ayrıca Android Studio’nun işletim sistemi Cross-platform şeklindedir. Bu, çoklu platform yazılımları demektir. Böylelikle birden fazla işletim sistemi için yazılımın dağıtımı gerçekleştirilebilmektedir.

Android Studio, yeni yazılım geliştiriciler için oldukça uygun olan bir ortamdır. Anlaşılır bir ara yüze sahiptir ve içerdiği araçlar ile yazılımcıların işlerini kolaylaştırır. Android cihazlar için, örneğin telefonlar ve tabletler, uygulama, oyun geliştirmek isteyen kişiler bu ortamdan faydalanabilirler. Ayrıca bu stüdyoda yapılan işlemlerin hata ayıklama kısmı da burada rahatlıkla gerçekleştirilebilmektedir.

Bu ortam ile sıfırdan hatasız bir oyun, uygulama üretmek mümkün olmaktadır. Biz de uygulamamızı geliştirirken Android Studioyu kullandık.





řekil 4.1. Uygulamamızın splash screen ekranı

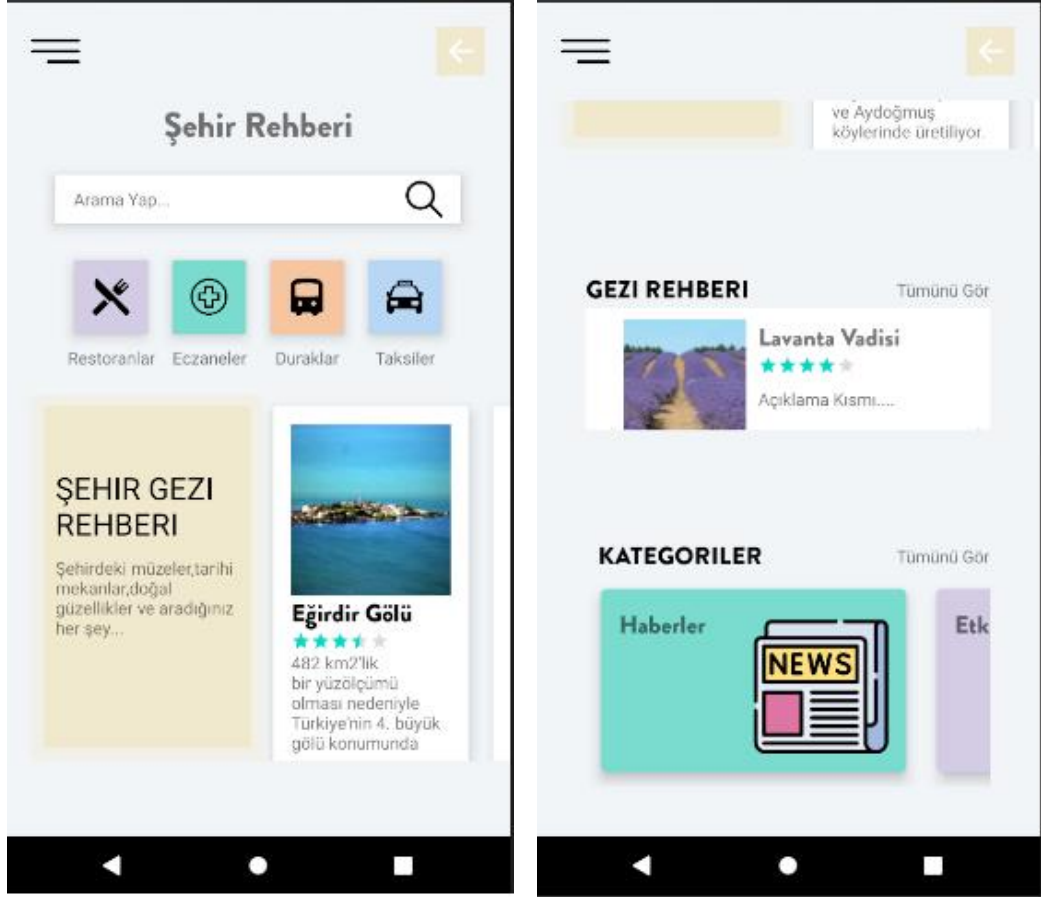
#### 4.1.1. Splash screen nedir ?

Splash Screen uygulamalarımızın açılması esnasında kullanıcıya gösterdiğimiz, genelde uygulama ile ilgili bir resmin olduęu açılıř ekranıdır.

#### 4.1.2. Neden splash screen kullanmalıyız ?

Android işletim sistemi uygulamamızın onCreate metodunu ararken belli bir süre geçer. Bu süre cihazdan cihaza, cihazın yoğunluęuna göre deęişebilir. Android işletim sistemi uygulamamızın onCreate metodunu ararken ekranda boş beyaz bir ekran gözükür.

Amacımız o süre zarfında kullanıcıyı boş beyaz ve hoş olmayan bir ekran ile karşılamak yerine uygulamamız ile ilgili bir ekran / tasarım göstermektir. Bunun içinde Splash Screen'leri kullanırız.



Şekil 4.2. Uygulamamızın anasayfa tasarımı

#### 4.1.3. Component nedir ?

Componentler basitçe, View tarafında kullandığımız nesnelerdir. Tasarımımızı componentler ile oluştururuz. Componentlere için “Widget” kavramı da kullanılmaktadır.

Ör: TextView, ImageView, EditText...

#### 4.1.4. Layout nedir ?

Android projelerimizde, Projemizin View yani arayüz kısmını .xml formatlı olarak tasarlarız. Bu arayüzler, layout dosyaları aracılığıyla oluşturulur.

Layout, yaptığımız tasarımda View componentlerin yaşama alanıdır diyebiliriz. Tasarımlarda kalıp görevi görerek sağladığı özelliklerine göre içinde kullanılan componentlere düzen sağlar, nasıl görülmeleri gerektiğini belirler.

#### **4.1.5. Linear layout**

Linear Layout, temel olarak içinde bulunan componentleri yatay (horizontal) veya dikey (vertical) olarak sırayla hizalamaya yarar.

Bunu belirlemek için xml özelliklerimizden veya daha önce söylediğimiz “Attributes” kısmından kullanılan Layout’un orientation(yön) ayarlamasını yapabiliriz.

#### **4.1.6. Relative layout**

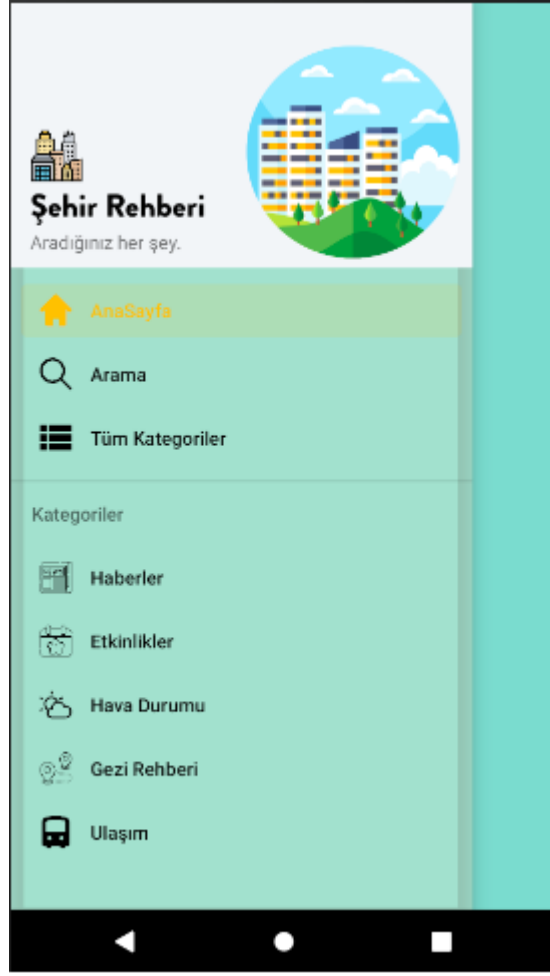
Relative Layout, temelinde programcıya tasarımda özgürlük tanır. Componentleri istediğimiz şekilde hizalayabilir ve sürükleyebiliriz. LinearLayout’tan farkı daha özgür olması ve farklı özellikler içermesidir. İçindeki componentleri, etrafındaki bileşenlere göre sağa, sola, yukarı veya aşağı hizalayabiliriz.

#### **4.1.7. GridLayout**

GridLayout, TableLayout’un daha gelişmiş halidir. TableLayout veri gösterimi için kullanılabilir fakat cansızdır ve dinamik olarak kullanışsızdır. Daha gelişmiş özellikleri içeren bir tablo görünümü için GridLayout kullanmalıyız. Çünkü GridLayout aynı zamanda ListView’e yakın mantıkta çalışır. Adapter ile kullanılabilir, gezinme, sürükleme vb. imkanlar tanır. Sütun veya satırlara istediğimiz componentleri yerleştirip onları düzene sokmamıza olanak sağlar.

#### **4.1.8. RecyclerView nedir ?**

RecyclerView, ilgili LayoutManager sınıflarını kullanarak listenin yatay(horizontal), dikey(vertical), zik-zak(staggered), ızgara(grid) görünümüne sahip olabilmesini sağlayan esnek ve performanslı bir View ögesidir.



Şekil 4.3. Uygulamamızın navigation menü tasarımı

#### 4.1.9. Navigation drawer nedir ?

Navigation Drawer , toggle menüye tıkladığımızda slide şeklinde açılıp bizi karşılayan menü diyebiliriz.

Navigation Drawer'ın, DrawerLayout isimli bir layoutu mevcuttur. Yukarıda gördüğünüz görünümü bize sunan bu layouttur. İçinde de FrameLayout bulunmaktadır. FrameLayout'lar üst üste getirilebilme özelliğine sahiptirler. Yani biz ekranda menü kapalıyken görünen yere FrameLayout koyup menüden bir alan seçtiğimizde ona ait Fragment'ı FrameLayout'a yükleyip tek bir Activity üstünden bir çok farklı işlev sunabiliriz.



Şekil 4.4. Uygulamamızın kategoriler kısmı tasarımı



Şekil 4.5. Uygulamamızın hava durumu kısmı tasarımı

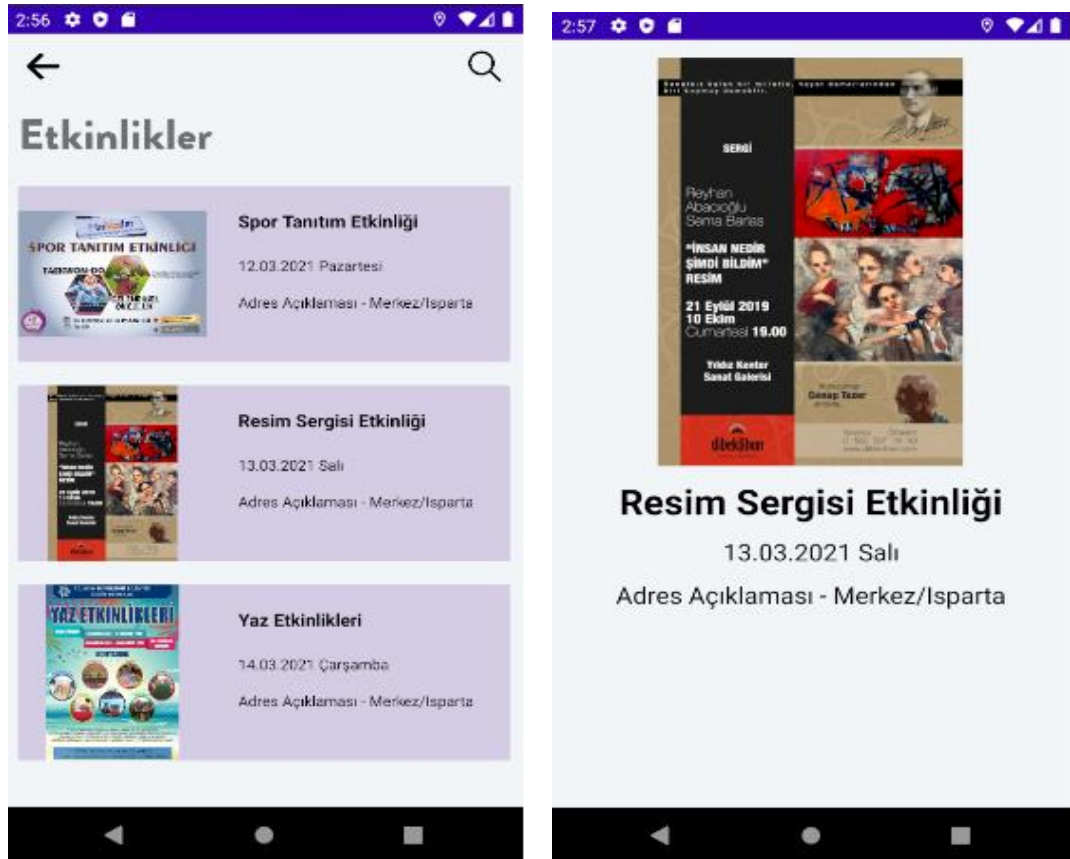
#### 4.1.10. API nedir ?

API (Application Programming Interface), bir uygulamaya ait yeteneklerin, başka bir uygulamada da kullanılabilmesi için, yeteneklerini paylaşan uygulamanın sağladığı arayüzdür.

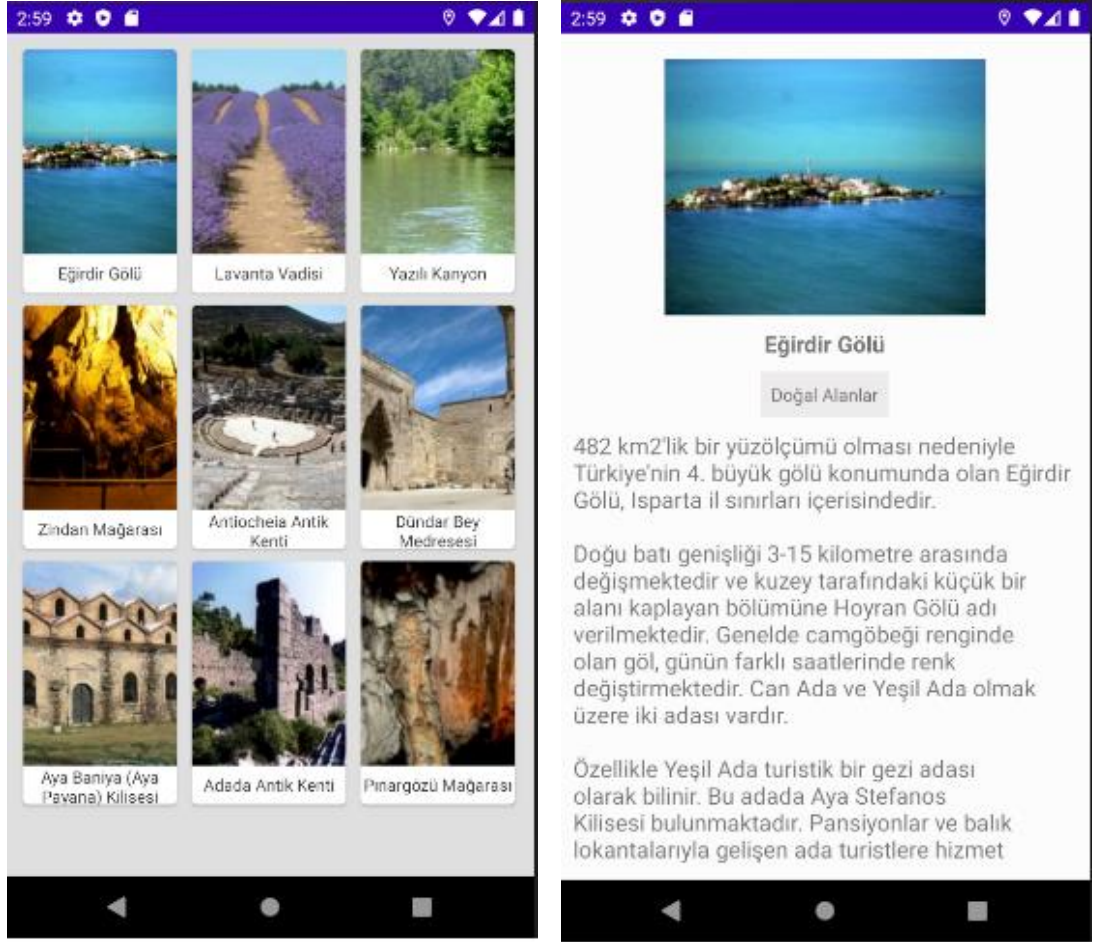
Basitçe, A uygulamasının özelliklerini B uygulamasında da kullanabilmemizi sağlayan yapıya API diyoruz.

Örneğin, ABC uygulamasından XYZ uygulamasına ait API'yi çağırarak, XYZ'nin API'sinde desteklenen tüm işlemleri ABC uygulamasında da gerçekleştirebiliriz. Yani XYZ uygulamasını açıp kullanmadan ABC uygulamasında da XYZ uygulamasına ait işlemleri gerçekleştirebilir hale gelir.

Örneğin, Geliştirdiğimiz bir projede Twitter API'yi çağırarak, Twitter'ın bize izin verdiği (tweet atma, hesap takip etme, vb.) işlemleri, API yetkimiz olan Twitter hesaplarında, kendi uygulamamız üzerinden de gerçekleştirebiliriz.



Şekil 4.6. Uygulamamızın etkinlikler kısmı tasarımı



Şekil 4.7. Uygulamamızın gezi rehberi kısmı tasarımı

#### 4.1.11. Uygulamamızın gezi rehberi tasarımı

GridLayout içinde yer alan android bileşenlerini bir dikdörtgen alana yerleştiren bir layout türüdür. İçindeki alanı hücrelere ayırdığı gibi satır ve sütun sayılarını da kendimiz belirleyebiliriz.

Şimdi de biraz CardView yapısından bahsedelim, CardView genelde tek başına kullanılabilen bir yapı değildir, RecyclerView veya ListView içerisinde bir liste elemanı olarak kullanılması yaygındır. CardView , farklı yapıda içeriklerin bir arada kullanmamıza ve görsellik olarak yeni bir deneyim kazanmamıza yardımcı olmaktadır. Uygulamanın gezi rehberi kısmında cardview ve recyclerview gridlayout ile birlikte kullanılmıştır.



Şekil 4.8. Uygulamamızın haberler kısmı tasarımı

#### 4.1.12. Uygulamamızın haberler kısmı tasarımı ve webview

WebView, hem uzak sunuculardaki web sayfalarını hem de uygulamanız içerisine koyduğunuz html sayfalarını görüntüleyebileceğiniz bir bileşendir. Bu bileşeni kullanarak uygulamanızdan çıkmadan web sayfalarını görüntüleyebilirsiniz.

Uygulamamızın web sayfasını açabilmesi için cihazın internetini kullanması gerekiyor. Bu sebeple kullanıcı uygulamayı yüklerken kullanıcıdan internet kullanımı için izin almalıyız.

Uygulamamızın haberler kısmında android studionun webview componentini kullanarak kullanıcıyı ıspartanın yerel haberlerini içeren bir web sitesine yönlendiriyoruz.





Şekil 4.9. Uygulamamızın ulaşım kısmı tasarımı

#### 4.1.13. Uygulamamızın ulaşım kısmı tasarımı ve tablayout

TabLayout, sekmeleri görüntülemek için yatay bir düzen sağlar.

Görüntülenecek sekmelerin doldurulması TabLayout.Tab örnekleri aracılığıyla yapılır. newTab() aracılığıyla sekmeler oluşturursunuz.

ViewPager entegrasyonu ;

Bu düzen ile birlikte bir ViewPager kullanıyorsanız, ikisini birbirine bağlamak için setupWithViewPager(ViewPager) ögesini çağırabilirsiniz. Bu düzen, PagerAdapter'ın sayfa başlıklarından otomatik olarak doldurulacaktır.

Bu görünüm aynı zamanda bir ViewPager dekorunun parçası olarak kullanılmasını da destekler.

TabLayout tab'lerden yani sekmelerden oluşan bir yapıdır. Biz projemizde TabLayout ve ViewPager'ı beraber kullandık. ViewPager ise sekmeler /sayfalar arası geçişi swipe yani kaydırma özelliği ile sağlar.

## **5. SONUÇ VE ÖNERİLER**

Bu tezde, şehir halkı ve turistler için Isparta şehir rehberi mobil uygulaması tasarlanmış ve uygulanmıştır. Bu mobil uygulama kullanıcıların Isparta hakkında ihtiyaç duyacakları ve işlerini kolaylaştıracakları bilgiler içermektedir ve kullanıcılar için rehber görevi görür. Uygulamanın geliştirilmesi için aşağıdaki maddeler üzerinde çalışılabilir ;

- iOS platformu için geliştirilebilir.
- Uygulamanın gezi rehberi kısmına daha fazla yer kaydedilebilir.
- Uygulamaya derecelendirme verme, favorilere ekleme, kullanıcı yorumları gibi ek özellikler eklenebilir.
- Uygulamaya turistlerin de rahatlıkla kullanabilmesi için farklı dil seçenekleri eklenebilir.

## KAYNAKLAR

Barış Sarıkaya (2013). *Mobil ilaç prospektüs uygulaması* (Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Bilişim Enstitüsü)

Anonim (2021). *Android Studio User Guide*. <https://developer.android.com/guide> (Son erişim tarihi: 02.08.2021)

Anonim (2021). <https://medium.com/> (Son erişim tarihi: 04.08.2021)

## ÖZGEÇMİŞ

Adı Soyadı : Faruk PINAR

Doğum Yeri ve Yılı : Kocaeli, 1997

Medeni Hali : Bekar

Yabancı Dili : İngilizce

E-posta : faruk\_\_\_\_@windowslive.com

### Eğitim Durumu

Lise : İzmit Atılım Anadolu Lisesi, 2015

Lisans : SDÜ, Mühendislik Fakültesi, 2021

## ÖZGEÇMİŞ

Adı Soyadı : Buğra ÖZTÜRK

Doğum Yeri ve Yılı : Çanakkale, 1998

Medeni Hali : Bekar

Yabancı Dili : İngilizce

E-posta : bugraozturk98@hotmail.com

### Eğitim Durumu

Lise : Özel Bandırma Final Akademi Anadolu Lisesi, 2016

Lisans : SDÜ, Mühendislik Fakültesi, 2021