

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

MOBİL İLAÇ PROSPEKTÜS UYGULAMASI

YÜKSEK LİSANS TEZİ

Barış SARIKAYA

Bilgisayar Bilimleri Anabilim Dalı

Bilgisayar Bilimleri Programı

Tez Danışmanı: Doç. Dr. Mustafa KAMAŞAK
Anabilim Dalı : Herhangi Mühendislik, Bilim
Programı : Herhangi Program

Aralık 2013

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ BİLİŞİM ENSTİTÜSÜ

MOBİL İLAÇ PROSPEKTÜS UYGULAMASI

YÜKSEK LİSANS TEZİ

**Bariş SARIKAYA
(704051017)**

Bilgisayar Bilimleri Anabilim Dalı

Bilgisayar Bilimleri Programı

**Tez Danışmanı: Doç. Dr. Mustafa KAMAŞAK
Anabilim Dalı : Herhangi Mühendislik, Bilim
Programı : Herhangi Program**

Aralık 2013

ÖNSÖZ

Mobil cihazların kullanımının arttığı son yıllarda, bir çok alanda mobil uygulamalar hayatımıza hızlı bir giriş yaptı. Bilgisayar yazılımlarında da çok önemli bir yer tutan eğitim ve sağlık alanları, mobil uygulamalarda da ipi önde göğüsleyen mobil alanlar oldular. Sağlık alanında sadece sağlık çalışanlarının değil hastaların ya da vatandaşların da ilgisini çekecek, hayatlarını kolaylaştıracak bir mobil uygulama gerçekleştirilmiştir. Cep İlaç her an yanımızda taşıyabileceğimiz bir ilaç prospektüs kitapçığı olarak hizmet vermektedir.

Cep İlaç uygulamasını destekleyen ve yorumlarını esirgemeyen sağlık personeline, ilaç sektörü çalışanlarına, arkadaşlarıma ve tüm uygulamayı kullanan herkese teşekkür ederim. Tez süresince yardımlarıyla ilerlememi ve uygulamanın gelişimine yardımcı olan, dostlarıma, sevdiklerime, tez danışmanım sayın Doç. Dr. Mustafa KAMAŞAK'a, bu tez konusunun temellerinin atılmasında bana fikir veren sayın Koray İNÇKİ'ye teşekkürü bir borç bilirim.

Aralık 2013

Barış SARIKAYA

ÖNSÖZ	v
ŞEKİL LİSTESİ	ix
ÖZET	x
SUMMARY	xii
1. GİRİŞ	1
1.1 Tezin Amacı	2
2. MOBİL UYGULAMA	1
2.1. Mobil Uygulama Nedir?.....	1
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	11
3.1. Yaşam Döngüsü	11
3.1.1. Planlama	12
3.1.2. Analiz Planlama	13
3.1.3. Tasarım.....	13
3.1.4. Gerçekleştirme	14
3.1.5. Test.....	15
3.1.6. Bakım	15
3.2. Bakım Yazılım Testi Yöntemleri	15
3.2.1. Yazılım Test Yöntemleri.....	16
3.2.1.1. Siyah Kutu Test Yöntemleri.....	16
3.2.1.2. Kullanıcı Gereksinimi Duymayan Test Yöntemleri.....	16
3.2.1.3. Kullanıcının Gerekli Olduğu Test Yöntemleri.....	17
3.2.1.4. Beyaz Kutu Test Yöntemleri.....	18
4. CEP İLAÇ YAZILIM GELİŞTİRME YAŞAM DÖNGÜSÜ	21
4.1. Planlama	21
4.2. Analiz	21
4.3. Tanıtım	23
4.4. Gerçekleştirme	23
4.5. Bakım	23
5. SAĞLIK ALANIN İNTERNET ve MOBİL UYGULAMA KULLANIMI 25	
6. CEP İLAÇ MOBİL UYGULAMA TASARIM	27
6.1. Veritabanı Tasarım.....	27
6.2. Ekran Tasarımları.....	33
6.2.1. İlaç Adı ile Prospektüs Bulma.....	33
6.2.2. Otomatik Tamamlama Özellikli Yazı Kutusu.....	35
6.2.3. Etken Madde ile İlaç Bulma	37
6.2.4. Endikasyon ile İlaç Bulma	38
6.2.5. Barkod ile İlaç Prospektüsü Bulma	40
6.2.5.1. Barkod Nedir?	40
6.2.5.2. Barkod Teknolojileri	42
6.2.5.2.1. Karekod Teknolojilerinin Kullanım Alanları.....	43
6.2.5.3. Cep İlaç Barkod Okuyucu	44
6.2.5.4. ZXing ("Zebra Crossing")	46
6.2.6. Firma ile İlaç Bulma.....	46
6.2.7. Favori İlaç Listeleri	47

6.2.7.1. Çok Arama Yapılan İlaçlar.....	47
6.2.7.2. Son Arama Yapılan İlaçlar	48
6.2.8. Reçeteni Oluştur	48
6.3. İlaç Arama İstatistikleri	49
7. MOBİL UYGULAMANIN YAYINLANMASI.....	51
8. MOBİL UYGULAMA KULLANIM SONUÇLARI.....	53
KAYNAKLAR.....	61
ÖZGEÇMİŞ	65
TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR	67

ŞEKİL LİSTESİ

Şekil 3.1 Yazılım Yaşam Döngüsü Adımları [2]	11
Şekil 3.2 Yazılım Geliştirme Döngüsü Tek Yaşam Süreci [2]	13
Şekil 3.3 Black-Box Test Metodu [3]	14
Şekil 3.4 White-Box Test Metodu [3]	15
Şekil 6.1 Cep İlaç Uygulamasının İlk Tasarlanan Veritabanı	16
Şekil 6.2 Tekrar Eden Verilerin Çıkarılması Sonrası Veritabanı Yapısı	18
Şekil 6.3 Şifrelenmiş Tablo Açık Tabloların Birlikte Olduğu Veritabanı Tasarımı	20
Şekil 6.4 İlaç Adı ile Prospektüs Arama ve İlaç Detay Bilgi Sayfası	34
Şekil 6.5 Android Standart AutoCompleteTextView Görünümü	39
Şekil 6.6 Etken Madde Arama ve Etken Maddeye ait İlaç Listesi Ekranı	40
Şekil 6.7 Endikasyon Arama ve Endikasyona Sahip İlaç Listesi Ekranı	39
Şekil 6.8 Örnek Barkod Etiketleri	44
Şekil 6.9 Barkod Etiketinde Bulunan Değerlerin Anlamları.....	45
Şekil 6.10 Barkod Okuyucu Çalışması ve Tarama Ekranı	48
Şekil 8.1 Cep İlaç Uygulaması Kullanıcı Oyları	50
Şekil 8.2 Cep İlaç Uygulamasının Kullanıcı Yorumları	51
Şekil 8.3 Cep İlaç Uygulaması Zaman Göre Yüklenme Sayısı	52
Şekil 8.4 Cep İlaç Uygulamasının Cihaz Bazlı Yüklenme Oranları	53
Şekil 8.5 Cep İlaç Uygulamasının Ülke Bazlı Kullanılma Oranları	55
Şekil 8.6 Cep İlaç Uygulamasının Operatör Bazlı Kullanım Oranları.....	57
Şekil 8.7 Andorid Google Play Sağlık Kategorisi En Çok İndirilenler Listesi.....	59

MOBİL İLAÇ PROSPEKTÜS UYGULAMASI

ÖZET

Bu yüksek lisans tezinde reçeteli ilaçlar için tasarlanan ve geliştirilen mobil uygulama (“Cepİlaç”) ve süreçleri anlatılacaktır. Başlangıçta bu uygulama çok sık kullanılan ilaçların prospektüslerini mobil ortamda daha okunabilir bir formatta sunmak için geliştirilmek üzere tasarlandı. Ancak sonradan sağlık sektörü çalışanlarından gelen yönlendirmelerle ve yapılan analizlerle birlikte 2D barkod okuma, alternatif (aynı etken maddeye) sahip ilaçları önerme, semptom ve endikasyonlarla ilaç arama, kendi reçeteni takip etme, ilaç hatırlatmaları gibi birçok özellik eklendi. Uygulamada kullanılan ilaç bilgilerinin toplanmasının ardından veritabanına yerleştirilmiştir. Veritabanına kaydedilirken kritik olarak belirlenen alanlar şifrelenerek oluşturulmuş, tek başına bir anlam taşımayan ilaç bilgileri ise şifrelenmeden veritabanına yerleştirilmiştir. Uygulama mobil cihazların kendi içerisinde tuttuğu yerel bir veritabanı kullanmakta ve internet bağlantısı ihtiyacı duymamaktadır. Şifreli ya da şifresiz olan verilerin veritabanından getirilmesi aşamasında, hızlı ve önbellekte saklanabilmeyi sağlayan algoritmalar geliştirilmiştir. Arama yapan ya da yoğun işlemci ve bellek kullanımı gerektiren işlemlerin gerçekleşmesi aşamasında ekranların kilitlenmemesi, kullanıcının ekran geçişleri yavaşlıklarından ya da görsel takılmalardan etkilenmemesi için farklı zamanlarda ve alanlarda çalışan iş parçacıkları üzerinden işlemler gerçekleştirilmiştir. Ekran tasarımlarının gerçekleştirilmesi aşamasında ise bir çok farklı çözünürlükte, boyutta ve oryantasyonda mobil cihazların olduğu düşünülerek eklenen görsel öğelerin bu durumlara göre uyumluluk sağlayabilmesi hedeflenmiştir. Bu nedenle görsel tasarımların gerçekleştirilmesinde çok yaygın olarak kullanılan görsel öğe tasarlama yöntemleri kullanılmıştır.

Sağlık bakanlığından onaylı olan Mayıs 2013’te yayınlanmış olan 11.200 ilaç şuanki veritabanında bulunmaktadır. Uygulama herhangi bir ücret ödmeden android uygulama marketinden (Google Play) “Cep İlaç” ismi ile aranarak indirilebilir. Bu uygulama, kullanıcılar tarafından çok kısa bir zamanda yaklaşık 195.000 defa indirilmiştir. Kullanıcılardan çok olumlu yorumlar alınmış, yeni özellikler ya da geliştirmeler için alınan geri bildirimler ile eklentiler gerçekleştirilmiştir. Halkın yanı sıra sağlık personelininde uygulamadan destek alıyor ve uygulamayı kullanıyor olması, uygulamanın işe yararlığına en büyük destek olmuştur.

Kullanıcıların, akıllı telefonlar ve tablet bilgisayarlar üzerinden kullanabileceği bu uygulamada reçete takibi bölümünde, ilaçların alınma zamanını ve doz miktarını hatırlatmaya yönelik hatırlatıcı sistem, önceki reçetelerin tarihsel olarak saklanması gibi birçok günlük hayata yardımcı, zamandan tasarruf sağlayıcı ve erişimi kolaylaştıran bölümler bulunmaktadır. Bu bölümlerin kullanıcılara sağladığı en faydalı alanlardan biri de ilaçların satın alındığı ve kullanılmaya başlandığı tarihlerin reçete tarihleri sayesinde biliniyor olmasıdır. Bu sayede ilaçların kullanım sonrası artması durumunda ilacın paketinin ne zaman açıldığına göre tekrar kullanılabilirliği sağlanabilir ve ilaç israfının önüne geçilebilir. Ayrıca kullanıcılar evlerinde buldukları yarım kalmış ilaçların durumları ile ilgili bilgi sahibi olabilmektedirler. Uygulama bir çok mobil platformda ve dillerde geliştirilebilir. Bu tez süresince, daha çok işletim sistemi versiyonuna ve cihaza hitap etmesi, açık kaynak kodlu bir çerçeveye sahip olması, çok yaygın kullanılan bir dil olan java ile kodlanması nedeniyle tüm bahsi geçen bölümler ve uygulama alanları android yazılım geliştirme ortamı kullanılarak tamamlanmıştır. Ortaya çıkan uygulama da android işletim sistemi kullanan mobil cihazlarda kullanılıyor olacaktır.

A MOBILE APPLICATION FOR PRESCRIPTION DRUGS

SUMMARY

This master thesis mentions about a mobile application designed and developed for prescription drugs called "Cep İlaç" and it's implementation process. At first application designed for users to find drugs' prospectuses in more readable format on mobile devices. However, subsequent health sector employees from the redirects and analysis with 2D barcode reading, alternative (same active ingredients) with a drug proposal, symptoms and indications drug-seeking, their recipe to follow, medication reminders as many features have been added. Application uses a local database that mobile devices keep their onboard storage and does not need an internet connection.

The drugs' information used in application collected from different sources were placed into the database. Information designated as critical were saved to encrypted database, drugs' information meaningless stand-alone were placed in the unencrypted database. While fetching encrypted or unencrypted data from the database, algorithms that allow you to get quickly and can be stored in cache mechanism have been developed. Under taking place doing searches or transactions requires intensive CPU and memory usage, operations were carried out through threads running different times and spaces to avoid the screen freezes, slowness of screen transitions or to prevent the interference of images.

A few different analysis has been performed on the database. With this analysis in mind, further decompilation and investigation of the database have been performed. The software library have been used that will encrypt specific attributes of databases residing on the internal secure digital card (SD card) of Android devices. The method have been developed could also benefit other Android applications requiring secure storage of data on globally readable and writable databases. For database encryption SQLCipher library were used which has a small footprint and great performance so it's ideal for protecting embedded application databases and is well suited for mobile development. It has blazing fast performance with as little as 5-15% overhead for encryption, 100% of data in the database file is encrypted, uses good security practices like key derivation, zero-configuration and application level cryptography, algorithms provided by the peer reviewed OpenSSL crypto library. AES-256 bit encryption method was used in CBC mode.

In the design phase, the visual elements aimed to ensure compatibility about screen realization of a very different resolutions, size and orientation of a mobile device were added. Therefore visual elements were used that widely used visual design method for designing. For Android UI design nine patch (9-patch) method was used to avoid different screen resolutions. Basically, 9-patch uses png transparency to do an advanced form of 9-slice or scale9. The guides are straight, 1-pixel black lines drawn on the edge of your image that define the scaling and fill of your image. By naming your image file name.9.png, Android will recognize the 9.png format and use the black guides to scale and fill your bitmaps. It's important to note that 9-patch images don't scale down they only scale up. It's not a perfect way to do scaling, but the fill-area and multi-line scale-guides does offer more flexibility than traditional 9-slice and scale9.

In this mobile application information about drugs can be gathered not only by their names and ingredients but also by there barcode numbers. For that solution application needs a barcode scanner, like image scanners. Barcode scanner designed for application should be run stand alone without a server or internet access, because the application designed not to use internet connection. There is a library for barcode scanning on mobile devices called ZXing. ZXing (pronounced "zebra crossing") is an open-source, multi-format 1D/2D barcode image processing library implemented in Java, with ports to other languages. Cep Ilac application focus is on using the built-in camera on mobile phones to scan and decode barcodes on the device, without communicating with a server. However the project can be used to encode and decode barcodes on desktops and servers as well.

There are 11.200 drugs in its current database which is approved by the Ministry of Health, which was published in May 2013. Application can be downloaded by searching by name "Cep İlaç" without paying any fees from Android application market (Google Play). This mobile application has been reduced to about 195.000 times by the users in a very short time.

Many positive comments received from users, new features or add-ons for improvements have been carried out by the feedbacks received. Public people and health personnel used the application and gave positive comments, that makes the much more help for the application.

Application security is another big issue in mobile developments. Security can be supplied with database and files on device encryption, or code obfuscations. The issued application has an encrypted database. If executable file with extension apk seized, application codes can be accessible to everyone. Code obfuscation is a good way to prevent application architecture and code know-how. ProGuard is a free Java class file shrinker, optimizer, obfuscator, and preverifier. It detects and removes unused classes, fields, methods, and attributes. It optimizes bytecode and removes unused instructions. It renames the remaining classes, fields, and methods using short meaningless names. In Cep İlaç application, creating more compact code, for smaller code archives, faster transfer across networks, faster loading, and smaller memory footprints features have been used because of the memory restrictions on mobile devices, making programs and libraries harder to reverse-engineer feature has been also used for prevent codes, and lastly listing and removing dead code feature has been used to obtain smaller size application code.

In this application that users can use on smart phones and tablet computers, follow the prescription section, taking drugs time and amount of dosage reminders for informative system , previous prescriptions historically storing as many daily life , helping you save time provider and facilitating access which sections are available. This sections help the users to remember drug's purchased date and started to use date through the date of the prescription date. In this way, the reusability of the package can be provided, in case of left of the drug after the use of drugs and can be prevented drug wastage. In addition, people found an information about status of unfinished drugs. Application can be developed in many mobile platforms and languages. This thesis throughout, more mobile operating systems and devices to appeal to the open source framework to have a very widely used language with java coding, because of all the aforementioned departments and applications which Android software development environment used to be completed. Application implemented will be used in mobile devices installed the Android operating system.

1. GİRİŞ

Haziran 2013 itibarıyla Türkiye’de yaklaşık %89,919 nüfuz etme oranına karşılık gelen toplam 68.025.878 mobil abone bulunmaktadır [1]. Temmuz 2009’da sunulmaya başlanan 3G hizmeti Haziran 2013 itibarıyla 45.341.769 milyon aboneye ulaşmıştır [1]. 3G hizmeti kullanan abonelerin 2013 Haziran itibarıyla %93 ü akıllı telefonlara sahip oldukları tahmin edilmektedir. Sonuç olarak insanlar bilgiye ve internete erişim için bilgisayarlardan çok mobil cihazları kullanmaktadır. Bankacılık işlemlerinden alışverişe oyunlara kadar neredeyse her uygulama mobil hale getirilmiştir. Son yıllarda sağlıklı yaşam ve sağlık uygulamaları için de internete duyulan ilgi artmaktadır. Mobil uygulamaların kullanılmasının yaygın olmadığı ilk dönemlerde genellikle internet bağımlı uygulamalar ön plandaydı. Ancak mobil uygulamaların daha çok alanda kullanılmaya başlanması sonrasında bağlantı gerektirmeden çalışan uygulamalarda geliştirilmeye başlanmıştır. İnteraktif anlık sunucuya bağlantı ihtiyacı duyulmayan durumlarda da uygulamaların çalışabilmesi mobil uygulamalara duyulan ilgiyi de arttırmıştır. Kullanım ve taşıma kolaylığı olması nedeniyle mobil cihazların gün geçtikçe günlük ihtiyaçları karşılamaya yönelik kullanımı yaygınlaşmaktadır. Gün içinde her an erişimine ihtiyacımız olan, yanımızda taşıyamadığımız veriler için mobil kullanımların daha rağbet gördüğü ortaya çıkmaktadır.

Birçok ülkede "Cep ilaç" benzeri uygulamalar bulunmaktadır. Ancak bazı özellikleri eksik yada farklı olarak geliştirilmiş olabilmektedir. Ya sadece ilaç prospektüs bilgilerine erişim sağlayan uygulamalar ya da ilaç bilgilerini kullanıcılara hatırlatan uygulamalar uygulama marketlerinde bulunmaktadır. Türkçe içerikli ücretsiz olarak kullanıma sunulmuş Cep ilaç dışında bu kategoride uygulama bulunmamaktadır. Sadece ilaç içeriklerinin detaylarını kullanıcılara sunan ücretli türkçe uygulamalar bulunmaktadır. İlaç prospektüs bilgilerini içeren, ilaç alım zamanı hatırlatması yapabilen, kendi reçeteni oluştur özelliği olan ve türkçe içerik sağlayan başka bir uygulama bulunmamaktadır.

1.1 Tezin Amacı

Bu uygulamanın ortaya çıkış amacı nedenlerinden ilki ilaçların içerisinde bulunan yazılı prospektüs bilgilerinin okunamaması ve özellikle evlerde ilaçların kutularının ve prospektüslerinin zamanla kaybolmasıdır. Genellikle prospektüs bilgilerinin bulunduğu kağıtlar yazıların sığması amacıyla mümkün olduğu kadar küçük punto ile yazılmaktadır. Küçük yazılar çoğu zaman okumayı zorlaştırmaktadır bunun yanı sıra zaten prospektüs okuma alışkanlığı çok az olan halkımızı bu bilgileri okumaktan daha da soğutmaktadır. Reçetesiz ilaç kullanımının çok yaygın olduğu ülkemizde prospektüs bilgilerinin ne kadar önemli olduğu net bir şekilde ortadadır. Evlerde kullanılan reçeteli ya da reçetesiz ilaçların prospektüs bilgilerine göre kullanılması yanlış kullanımlarında önüne geçmede yardımcı olacaktır. Evlerimizde kullandığımız ilaçların çoğunun kutuları veya prospektüsleri bir süre sonra erişilememekte ya da kaybolmaktadır. Bu da o ilaçların prospektüs bilgilerinin, son kullanma tarihinin, kullanım şekli ve zamanlarının erişilemez olmasına neden olmaktadır. İlaçların doktor tarafından önerilen kullanım şekli ve zamanları genellikle, eczacılar tarafından ilaçların kutularının üzerine yazılmaktadır. Kutunun kaybolması sonrası hem kullanım şeklinin yazdığı bölüm erişilemez hale gelmektedir hem de kullanım şekillerine ulaşabileceğimiz prospektüs erişilemez olmaktadır. Günlük hayatta insanların kolay bir şekilde her an ulaşabileceği, kaybolma riski olmayan, istenilen büyüklükte okunabilen bir çözüm bu sorunları giderecektir.

Ön tasarım sonrası uygulamada bulunan bölümler ortaya çıkmıştır. Ön tasarım ile birlikte uygulamada kullanılacak verilerin ve medikal bilgilerin nasıl ve nereden edinileceği hususunda bir sonuca ulaşılmış oldu. İlaçlarla ilgili bilgileri sunabilmek, onlar üzerinden özel verileri sağlayabilmek için öncelikle bu verilerin sağlıklı bir şekilde toplanması gerekmektedir. Bu nedenle geçerliliği kanıtlanmış ve otoriteler tarafından kabul görmüş sağlayıcılar tarafından yayınlanan ilaç bilgi kaynaklarından ilaçların prospektüs bilgileri toplanarak bir veritabanına kaydedildi. Sağlık Bakanlığının da yayınladığı listede bulunan ilacın barkodu, fiyatı ve ilacın Türkiye'deki firmasına ait bilgileri de ayrıca toplanarak veritabanında bulunan ilaç bilgileri ile eşleştirildi ve veritabanına kaydedildi.

İlaç bilgilerinin kaynaklardan toplanması ve veritabanı tasarımının ardından formatlı bir biçimde veritabanına kaydedilmesi işlemi yaklaşık 1 (Bir) ayda tamamlanmıştır. Bu çalışma sonrasında prototip uygulama için kullanılacak bir veritabanı elde edilmiştir. Uygulama kullanımına açılacağı zaman veritabanına erişim ve güvenlik ile ilgili bir çalışma yapılmıştır.

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 aktarıp 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, dilendiğinde 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 Geçmişi

Cep telefonlarının hayatımıza girmesi ile birlikte ilk mobil uygulamalar da ortaya çıkmış oldu. İlk mobil uygulamalar ile sadece arama yapmak, mesaj göndermek dışında başka bir şey yapılamıyordu. Bunda önemli bir etken de ilk çıkan mobil işletim 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 kişi 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 iş sektöründe hem de günlük hayatta, yaşamı daha kolay bir hale getirmeye başladı. Örneğin; yeni çıkan mobil cihazlar üzerinde daha gelişmiş rehber uygulamaları ortaya çıktı. Bu mobil uygulamalar; iletişim kurulmak istenen yüzlerce kişinin numarasını tek bir mobil cihazda kolayca saklayıp, istenen kişiye 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 paylaşmıyorlardı. 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 bildiği 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 amacıyla mobil cihazlara konmuştu. Ancak mobil cihaz kullanıcıları, zaman geçtikçe daha çok özellik ve daha çok oyun talebinde bulunmaya başladılar.

Cep telefonlarının ve mobil cihazların yaygınlaşmasından kısa bir süre sonra java tabanlı çok daha esnek olan mobil işletim sistemleri ortaya çıktı. Bunlardan en önemlisi ve günümüzde çok daha gelişmiş versiyonları bulunan Nokia şirketi tarafından geliştirilmiş “Symbian” işletim sistemidir. Bu işletim sistemleri eskisinden daha gelişmişti ve gelişmiş mobil uygulamalara kapılarını açmıştı. Yeni java tabanlı mobil işletim 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 hızda gelişerek milyonlarca insanın hayatını değiştirdi. Günlük hayatı kolaylaştıran, iş adamlarına kolaylık sağlayan, sosyal paylaşım sitelerine heran bağlı olabilmek 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 geliřtirmek istendiğinde genel olarak izlenebilecek 3 farklı yol bulunmaktadır. Bunlardan birincisi "Native" uygulama geliřtirmedir. Native uygulama geliřtirmek için belli bir iřletim sistemine belli bir programlama dili ile geliřtirme yapılması gerekmektedir. Bu řekilde uygulamanız daha güvenli ve daha performanslı olmaktadır. Diđer bir yol da "Web" tabanlı bir mobil uygulama yazarak sadece tek seferde tüm platformlarda çalışabilecek bir uygulama geliřtirebilirsiniz. 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 geliřtirme yöntemidir. Bu yolu kullanmak için Native kodlamanın içerisine web tabanlı bir sisteme erişim sağlanarak iki teknolojinin birden aynı anda kullanılması sağlanır.

Mobil uygulama geliřtirmeye 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 olacağını kararının verilmesi gerekmektedir. Uygulamanızın sadece belirli bir ya da birkaç mobil iřletim sisteminde, güvenli ve daha performanslı çalışmasını istiyorsanız native uygulama geliřtirmelisiniz. Düşük bir maliyetle her platformda çalışan ve daha çok güncel ve web odaklı bir uygulama geliřtirmek istiyorsanız web tabanlı bir uygulama geliřtirmek çok daha avantajlı olacaktır, ancak son zamanlarda popüler iřletim sistemlerinin "native olmayan" uygulamalara desteğini azaltması, gelecekte sorun yaratabilir.

Android, iOS gibi mobil platformlara uygulama geliřtirmek için bilgisayarınıza birtakım yazılımlar yüklemeniz gerekir. Bunlardan biri SDK (Software Development Kit) adıyla bilinen yazılım geliřtirme araçlarıdır. Bu araçlar, uygulama geliřtirmenize yardımcı ve kolaylaştırıcı bir çevrede yaratır. Uygulama geliřtirirken kullanacağınız kütüphaneler, debugger (hata bulucu) ve gerekli başka her şeyi içinde barındırarak geliřtiriciye her aşamada yardımcı olur. Mobil cihaz simülatörleri aracılığıyla gerçek bir cihaza gerek duymadan geliřtirdiğiniz 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.

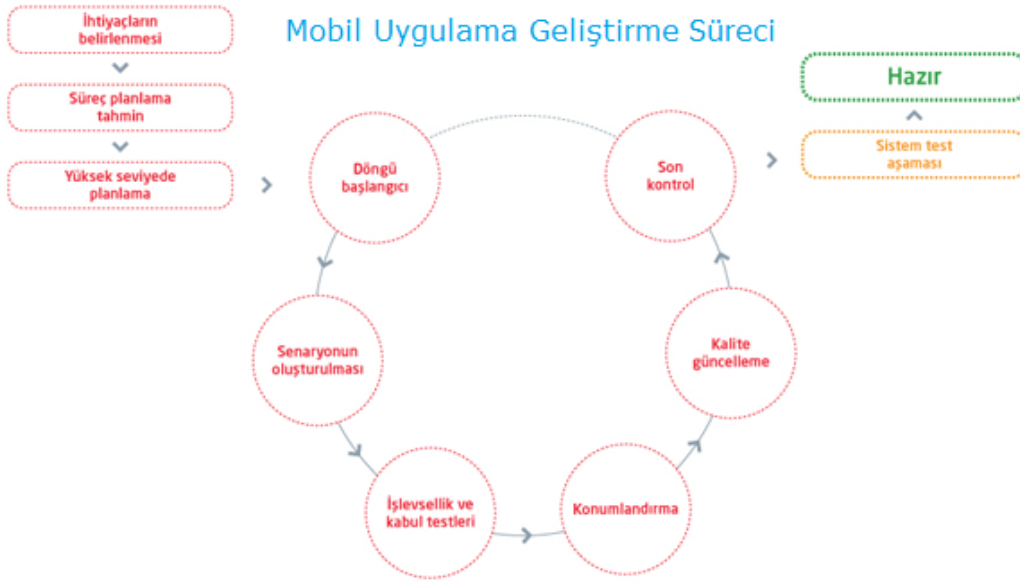
Bir mobil uygulama projesinin geliştirilmesi, sadece kodlamadan oluşmamaktadır. Basitçe bir proje geliştirilirken dahi projenin planlama, analiz, tasarım, üretim ve test aşamaları yer almaktadır ve almalıdır. Bu aşamaların bir kere gerçekleştirilmesinden sonra proje tamamlanmayabilir. Bu aşamaların bir 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 [2]

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 [2]

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 ilgilenir. 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.

3.2. Bakım Yazılım Testi Yöntemleri

Yazılım testi kullanmak, yazılım geliştirme yaşam döngüsünün önemli parçalarından biridir. Etkin bir test kodu oldukça önemlidir.

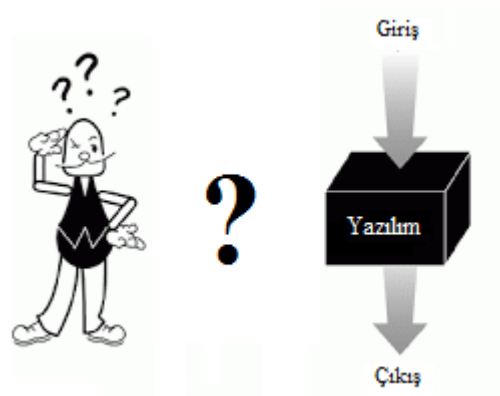
Yazılım testi, önceden belirtilen gereksinimleri karşılayıp karşılamadığını, doğru çıktıyı üretip üretmediğini kontrol eden yapıdır. Çok sayıda test durumu ve test stratejisi hazırlanır, bu testlerin hepsi belirli hedefleri (tüm sorunları kaldırmak, yazılımın hatasız çalışmasını sağlamak ve en iyi çıktıyı elde etmek) için uğraşır. Çok sayıda test tekniği ve metodolojisi bulunur. Yazılım test metodolojisi, yazılım testi tekniklerinden farklıdır.

3.2.1. Yazılım Test Yöntemleri

3.2.1.1. Siyah Kutu Test Yöntemleri

Siyah kutu (Black box) testi Şekil 3.3'de de görüldüğü üzere, kod ya da tasarım ile ilgilenmez. Testler gereksinim ve fonksiyonellik üzerinedir. Siyah kutu testi aynı zamanda, fonksiyonellik testi, closed box testi ya da opaque testi olarak adlandırılır. Siyah kutu testi, fonksiyonelliğe uygun veri seçimine ve seçilen verinin programın normal ve ya anormal davranış gösterimini izleme temeline dayanır.

Siyah kutu test stratejisinde, kullanıcı sistemin gereksinimlerini ve bu gereksinimlere nasıl cevap vereceğini bilmelidir. Siyah kutu testlerini iki gruba ayırabiliriz, bunlardan biri kullanıcı gereksinimi duymayan testler, diğeri ise kullanıcının gerekli olduğu testlerdir.



Şekil 3.3 : Siyah Kutu Test Yöntemi [3]

3.2.1.2. Kullanıcı Gereksinimi Duymayan Test Yöntemleri

Kullanıcıdan bağımsız olarak uygulamanın çalışmasının, dayanıklılığının, kullanılabilirliğinin ve uygulanabilirliğinin test edildiği metotlardır.

- Fonksiyonellik testi, fonksiyonel gereksinimleri test etmek için yazılmıştır. Eğer uygulama beklendiği gibi davranırsa, testlerin yazılmasına sıralı bir şekilde devam edilir.
- Stres testi, çok sayıda veri girişi, büyük nümerik değerler, çok sayıda sorgu olduğu zaman kullanılır ve uygulamanın dayanıklılığını belirler.

- Yükleme testi, bir sistemin performansını derecelendirmek ya da ağır yükler ve çok sayıda veri girişinde sistemin hangi noktada çakılacağını tespit etmek için kullanılır.
- Ad-Hoc testi, geçerli bir test yaratılmadan önceden kullanım için uygundur. Bu test diğer testlerin kapsama alanını ve diğer testlerin süresinin belirlenmesinde yararlı olur.
- Araştırma testi, Ad-Hoc testine benzer ve uygulamayı öğrenmemizi sağlar. Uygulama hakkında ön bilgi edinilmesini sağlar.
- Kullanılabilirlik testi, kullanıcı dostu test olarak da bilinir. Eğer kullanıcı ara yüzü önemli bir yere sahipse ve ihtiyaçlar belirli bir kullanıcıya göre belirleniyorsa uygundur.
- Duman testi, mantık testi olarak da bilinir. Bu test, uygulamanın büyük testlere hazır olup olmadığını belirler ve küçük testleri başarı ile geçtiğini gösterir.
- Yenilenme testi, uygulamanın herhangi bir hataya karşı ne kadar sürede eski haline geleceğini test eder. Sistem gereksinimlerine göre, tip ve yenileme hızı belirlenir.
- Seviye testi, uygulamanın etkinliği ile ilgilenir. Uygulama tarafından çok büyük veri işlemi yapılırken, sistemin uç limitlerini kontrol eder.

3.2.1.3. Kullanıcının Gerekli Olduğu Test Yöntemleri

Kullanıcının onayının gerektiği, kullanıcı yorumlarının dahil edildiği test metotlarıdır.

- Kullanıcı kabul testi, kullanıcının sistemi test edip, gereksinimleri karşılayıp karşılamadığının incelenmesini sağlar.
- Alfa testi, kullanıcı geliştirme merkezine çağrılır. Geliştiriciler programı kullanır ve program hakkında not alır ya da kullanıcı işlemleri gerçekleştirir.
- Beta testi, kullanıcılara beta versiyon dağıtılır ve test etmesine izin verilir. Kullanıcılar sistemi inceler, herhangi bir sorun bulduğunda, geliştiriciye bildirir.

Beyaz kutu (White box) testi, uygulamanın kodunu temel almaktadır. Kodun koşullarını, alanlarını ve açıklamalarını temel alır. Beyaz kutu testi, cam, açık kutu, temiz kutu olarak da adlandırılmaktadır. Bu testte, testi yapan kişi sorunlu kısmı bulmak için kodu incelemelidir.

Avantajları

- Kodun optimizasyonunu sağlar.
- Ekstra kod parçasını kaldırır ve tek bakışta gözükmeyen sorunları ortaya çıkarır.
- Hangi verinin kodu en iyi şekilde test edeceğini tespit eder.

Dezavantajları

- Belirli özelliklere sahip test yapıldığında maliyet artar.
- Kodun inceleyip hata bulmak çok zor bir işlemdir.

3.2.1.4. Beyaz Kutu Test Yöntemleri

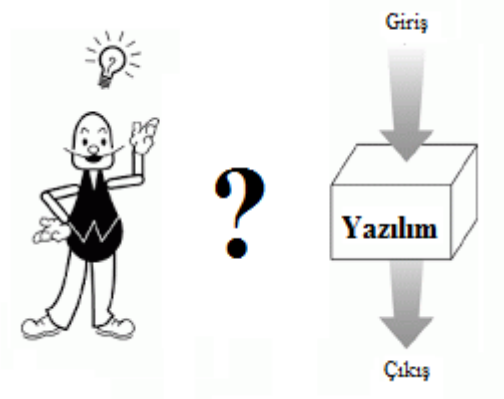
Yazılım ile ilgili en kapsamlı ve detaylı yapılan test metotlarıdır (Şekil 3.4). Kodların yapısı ve uygunluğu, birim testler, güvenlik testleri gibi bir çok test bu grup altında toplanabilir.

- Birim test, çalışan bir kod parçası ya da modül için, geliştiriciler tarafından gerçekleştirilir. Düşük seviyede işlem gerçekleştirir.
- Statik ve dinamik analiz, statik analiz kodu sıralı bir şekilde inceler ve hataları araştırır. Dinamik analiz, kodun çalışmasını ve çıktısını analiz eder.
- Açıklama kapsamı, açıklamaların test edilmesiyle ilgilidir. Her bir açıklama en az bir kez test edilir. Tüm açıklamaların sorun yaşamadan çalıştığını garanti altına alır.
- Sınıf kapsamı, hiçbir yazılım uygulaması sürekli olarak kodlanmaz, bazı noktalarda, kodun dağılışı incelemeli ve belirli bir fonksiyonu çalıştırılmalıdır. Tüm sınıflar doğrulanmasına yardım eder ve uygulamanın anormal davranış göstermesini engeller.
- Güvenlik testi, sistemin izinsiz erişimler, kod bozulması, hack işlemleri gibi olaylardan nasıl korunacağı ile ilgilidir. Karmaşık test metotları gerekir.

- Deęişim testi, belirli bir hata düzeltildikten sonra yapılan bir testtir. Hangi kodun, stratejinin geliřtirmeye yardımcı olacaęını belirlemeyi saęlar.

Bazı testler hem beyaz kutu hem de siyah kutu kapsamında incelenebilir. Bunlara örnek verecek olursak;

- Fonksiyonellik testi, kodla birlikte fonksiyonellięi test eder.
- Birleřtirme testi, uygulamaya yeni eklenen kod ile iřbirlięi yapar.
- Performans ve yükleme testi, kodun hangi kısmının sistemin kaynaklarını yönettięini ve performansını veriliřini belirler.



Şekil 3.4 : Beyaz Kutu Test Yöntemi [3]

4. CEP İLAÇ YAZILIM GELİŞTİRME YAŞAM DÖNGÜSÜ

Cep İlaç uygulamasının ortaya çıkışı, mobil ortamlarda ilaç prospektüslerinin, endikasyon ve etken maddeler üzerinden aranabilmesini sağlayan bir uygulama ihtiyacı oluşmasıdır.

4.1. Planlama

Projenin planlaması yapılırken proje içerisinde yapılacak işlemleri temel hatlarıyla ortaya çıkarmak gerekmektedir.

- İlaç araması yapılabilmesi için ilaç bilgilerinin bulunması gerekir,
- Hangi kriterler ile arama yapılabileceğinin belirlenmesi gerekir,
- Kullanıcıya sunulacak verilerin ve detaylarının belirlenmesi gerekir,
- Kullanıcıdan alınacak bilgilerin kararının verilmesi gerekir.

Projede yapılacak işlemler basitçe belirlendikten sonra projenin geliştirilmesi ve kullanımını için gerekli ortam belirlenir.

- Proje java programlama diliyle yazılacaktır ve bu proje bir android mobil uygulaması olacaktır.
- Projenin mobil cihazlar üzerinde çalışması için en az android 2.3 versiyonu özelliklerine sahip olması gerekmektedir.
- Proje bilgileri SQLite veritabanında tutulacaktır. Veritabanı mobil cihazın yerel hafızasında oluşturulacaktır.

4.2. Analiz

Proje planlamasında belirtilen maddeleri müşteriyle beraber (müşterinin kendisi ya da proje için belirlediği bir analiz ekibi/personeli ile) projenin detayları belirlenir.

- Proje 5 parçadan oluşacaktır: İlaç adıyla prospektüs arama, endikasyon ile ilaç arama, etken madde ile ilaç arama, barkodu ile ilaç arama, kendi reçeteni oluştur.
- İlaç adı işlemleri: İlaç adının girilerek prospektüs bulunması ile ilgili işlemleri kapsar.
 - Arama: İlaçların veritabanında kayıtlı olan prospektüslerine ilaçların adları ile arama yapılarak erişilir.
- Endikasyon işlemleri: Endikasyon kullanılarak yapılan arama ile ilgili işlemleri kapsar.
 - Herhangi bir rahatsızlık (endikasyon) girilerek ilaç bulunmasını sağlar.
 - Bulunan ilaç listesinden istenen ilacın detayının görülebilmesini kapsar.
- Etken madde işlemleri: Etken madde ile arama işlemlerini kapsar.
 - İlacın etken maddesi ile ilaç aranmasını ve bulunmasını sağlar.
 - Bulunan ilaç listesinden istenen ilacın detayının görülebilmesini kapsar.
- Barkod işlemleri:
 - İlaç üzerindeki barkod okutularak ilacın prospektüs bilgilerinin getirilmesini sağlar.
- Reçete işlemleri: Kullanıcının kendi reçetesini yaratabilmesi ile ilgili işlemleri belirtir.
 - Yeni reçete oluşturmayı sağlar.
 - İlaç seçerek reçeteye eklenmesini sağlar.
 - Reçetede ilaçlara kullanım senaryosu girebilir.
 - İlaçlarla ilgili hatırlatmalar ayarlanmasını sağlar.

4.3. Tanıtım

Analizde, yapılacak işlemler detaylandırıldıktan sonra yapılacak işlemler fonksiyon / ekran / işlem bazında listelenir ve hangi işlem için ne kadar zamanlama gerektiği belirtilerek proje planlaması oluşturulur.

Uygulamanın zaman planlamasının yapılabilmesi için belirlenen aşamaların süreleri verilir. Cep ilaç uygulamasında belirlenen aşamalar; ilaç veritabanının oluşturulması, veritabanı mimarisinin oluşturulması, kriptolu veritabanının oluşturulması, ilaç adı ile arama ekranı, prospektüs detay ekranı, muadil ilaçlar ekranı, endikasyon ile arama ekranı, etken madde ile arama ekranı, barkod ile ilaç arama ekranı, reçete ekleme ekranı, yardım ekranı, fonksiyonel testler, ekran testleri şeklindedir.

4.4. Gerçekleştirme

Tasarım aşamasında belirtilen detaylar, günler, başlangıç ve bitiş tarihleri kapsamında projenin, toplam başlangıç tarihinde başlanır ve bitiş tarihinde tamamlanacak şekilde kodlaması ve ekran, fonksiyonel testleri gerçekleştirilir. Tüm kodlama, iş mantığı kurallarının uygulanması ve fonksiyonelliğin yapılması bu aşamada gerçekleştirilmektedir.

4.5. Bakım

Yazılım tamamlandıktan ve yayına alındıktan (müşteri kullanmaya başladıktan) sonra yazılıma eklenmesi gereken yerler olduğu belirtildi: *Aranan ilaçların ne kadar sıklıkla arandığı ve favori ilaçların listelenmesi gereken bir ekran olması gerektiği ortaya çıktı.*

Bu durumda gelen yeni talepler için yazılım geliştirme yaşam döngüsü bu istekler için başa döner ve yeni istekler için Planlama > Analiz > Tasarım > Üretim > Test > Bakım süreçleri başlar.

Yukarıda belirtilen yazılım geliştirme yaşam döngüsü tanımlaması, genel hatlarla yazılım yaşam döngüsünün ne olduğunu anlamak için verilmiş standart bir süreçlerle benzerlik göstermektedir. Cep ilaç mobil uygulamasının yazılımının detaylarının ve işlevlerine bağlı olarak farklı standart metodolojiler de üretilmiştir. Onlarda kullanılarak eklentiler yapılarak genişletilebilir.

5. SAĞLIK ALANIN İNTERNET ve MOBİL UYGULAMA KULLANIMI

Bozulan ekolojik denge, kaynakların eşit olarak dağılmaması, küresel ısınma vb. nedeniyle oluşan sorunlar ve insanın kendi eliyle sağlığına verdiği zararların artmasıyla birlikte, bilinçlenmeye başlayan toplumlar bu sorunları çözmek için artık daha fazla çaba ve kaynak harcamaya başlamışlardır. Bunun üzerine dünyada artan yaşlı nüfusu ve buna bağlı olarak kronik hastalıklardaki artış; hastane ve sağlık sistemi üzerindeki yükü arttırmakta ve sağlık harcamalarını baş edilemez düzeylere çıkarmaktadır. Sağlık personeli yetersizliği de tabloya eklenince sağlık sektöründe teknolojinin daha etkin ve verimli olarak kullanılmasının zorunlu olduğu belirgin şekilde ortaya çıkmaktadır.

21. yüzyılda sağlıkta gerçekleşecek bu büyük dönüşüm sağlık sisteminin hastane ve doktor odaklı bir sağlık sistemden, birey odaklı bir sistem haline geleceğini ve “kişiselleşeceğini” de işaret etmektedir. Kişiselleşen sağlık kavramı; sadece hasta olduğunda sağlık sistemine başvuran ve takip edilmeyen bireyden farklı olarak, kendi sağlığıyla ilgili sorumluluk alan, sağlıklı yaşamaya, hasta olmamaya özen gösteren, sağlık verilerini yanında taşıyan, gerektiğinde hastalığıyla ilgili verilerin 7/24 kontrol altında tutulduğu proaktif bir yaklaşımı betimlemektedir.

Bireyin hastalık aşamasına geçmeden önce daha rahatsızken internette bilgi aramaya ve doktora gitmeden önce bilinçlenmeye çalışması, sosyal medyada oluşan hastalık grupları ile birlikte hastalığa yönelik sosyal destek ve bilgi paylaşımının sosyal ağlarda artması, teknolojinin gelişmesi ile orantılı olarak evde uzaktan bakım programları (internet ile doktora ulaşma ve konsültasyon), mobil ve akıllı telefonların artması ile akıllı sağlık uygulamaları ve sağlık bilgilerinizi artık sayısallaştırarak her türlü donanım ile işleyen ve yorumlayan aygıtlar (pedometreler, tansiyon şeker ölçen bilezikler vs.) bugün sağlığın geliştirilmesi adına sağlık iletişimideki yeni dijital uygulamalardır.

Özellikler, tüm dünyada insan sađlıđının geliştirilmesi ve teşvik edilmesi için sađlık hizmetlerinde yardımcı öğeler olarak kullanılan mobil sađlık uygulamaları arařtırmaları göstermektedir ki, eđer bu uygulamalar iyi dizayn edilip, iyi uygulanırsa sađlıklı davranıř üzerinde belirleyici etkilere sahip olabilmektedir.

Bireysel İnternet kullanımındaki olađanüstü artıř, İnternet'in gündelik hayatımızın vazgeçilmez bir parçası olmasına yol açmıřtır. İnternet, hekimler arası ya da hekim ve hastaları arasındaki e-posta haberleşmesinden, hemřire takip sistemlerinden sađlık bilgileri içeren ve danıřma hizmeti veren İnternet sitelerine, tele tıp hizmetlerinden, eczanelerin ilaç dađıtımına kadar, sađlıkla ilgili pek çok alanda yaygın olarak kullanılmaktadır.

Her geçen gün artan bir hızla büyüyen İnternet, giderek günlük hayatımızın vazgeçilmez bir parçası haline gelmektedir. Ekim 2010 itibarı ile dünyadaki İnternet kullanıcısı sayısının iki milyar civarında olduđu bildirilmektedir [1]. Yetkililerin yaptıkları açıklamalara göre, Türkiye'de de yaklaşık 35 milyon kiřinin İnternet kullanıcısı olduđu tahmin edilmektedir [2]. Ancak, İnternet'in bu hızlı büyümesi ve karmařık yapısı, İnternet üzerinden iřlenen suçların ve yapılan etik ihlallerin de artmasına neden olmaktadır. Bu konuya iliřkin hukuki ve etik düzenlemeler genel çerçevelerin dıřına çıkamamakta gelişmelerin gerisinde kalmakta ve bu nedenle konuya özgü çözümlerlerin hızla gerçekleştirilmesini gerekli kılmaktadır.

Pek çok alanda bilgi paylaşımına olanak sađlayan İnternet'in, günlük yaşamda önde gelen kullanım alanlarından biri de sađlıktır. Sađlık hizmeti almakta olan veya almak isteyen geniş bir kullanıcı kitlesi bilgi edinmek amacıyla İnternet'e bařvurmaktadır. Bunun yanı sıra, sađlık hizmeti veren kurum ve görevliler de mesleki bilgi ve veri paylaşımı için İnternet'ten yararlanmaktadır.

6. CEP İLAÇ MOBİL UYGULAMA TASARIM

Mobil uygulamaların tasarım aşamasının aslında hiç görünmeyen bir adımı daha bulunmaktadır. Bu adımda hangi kitleye, hangi ortama hitap edileceğinin kararının verilmesi gerekir. Ekran tasarımından yazılım alt yapının tasarımı ve kodlamasına kadar bir çok bölüme yön verecek olan en önemli karar işletim sistemi seçimidir. Yapılması planlanan mobil uygulama hangi işletim sistemi üzerinde koşacak ise o ortama özel tasarım ve kodlama yapılması gerekecektir. Açık kaynak kodlu olması, bir çok alanda yaygın olarak kullanılan java dili ile uygulamaların geliştirilebiliyor olması Android işletim sistemini seçilmesindeki en büyük nedendir.[24] Android işletim sisteminde çalışacak bir mobil uygulama geliştirmeye başlamadan önce uygulama geliştirme yöntemlerinden[25] hangisinin seçileceğinin kararının da verilerek tasarıma başlanmıştır.

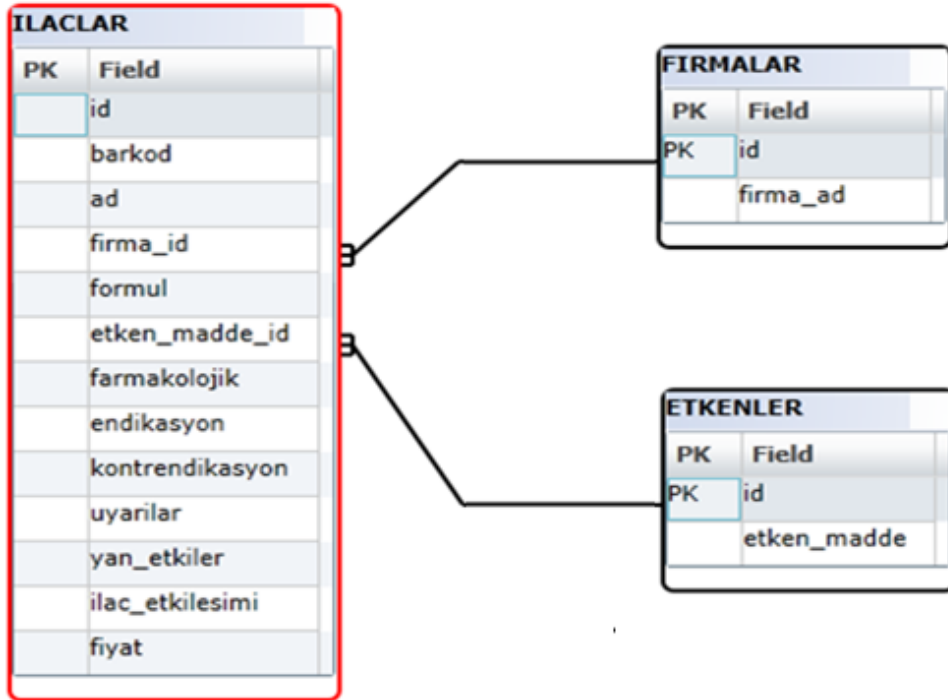
6.1. Veritabanı Tasarım

Uygulamada kullanılmak üzere toplanan ilaç bilgilerine kolay erişim ve gösterim için bir veritabanı tasarımı gerçekleştirilecektir. Toplanan tüm bilgiler Şekil 6.1'de görüldüğü şekilde tasarlanan bir tabloya aktarıldı.

Bu tablo tek başına oluşturulduğunda içerisindeki bilgiler ile çok büyük bir boyuta sahip olmaktadır. Mobil cihazlarda veri saklama alanı kısıtlı olduğu için büyük boyutlu tablolar barındırmak ve onlar üzerinde sorgular yapmak, işlemci hızının da sınırlı olduğu bu tür cihazlarda oldukça yavaş ve zor olmaktadır. Uygulamada bulunan ekranlarda herhangi bir ilaçla ilgili tüm bilgiler getirilmemektedir. Bu nedenle kullanılmayan verilerin getirilmesi sisteme gereksiz ve ağır yük getirecektir. Tek tablo yerine birden fazla tablo ile bir yapı oluşturularak erişimin daha hızlı ve kolay olması sağlanması kararı alındı. Bu sırada Sağlık Bakanlığının yayınlamış olduğu ancak halk tarafından ve sağlık personeli tarafından çok fazla ihtiyaç duyulmayan alanları da kaldırarak yeni daha çok bilinmek istenen alanlar olacak şekilde bir tasarım yapıldı.

ILACLAR	
PK	Field
	id
	barkod
	ad
	firma
	tip
	etken_madde
	atc_siniflandirma
	nfc_siniflandirma
	fiyat
	uretici_fiyat
	magaza_fiyat
	kamu_fiyat
	kamu_no
	lisans_bilgisi
	mense
	recete
	hamilelik

Şekil 6.1 : Cep İlaç Uygulamasının İlk Tasarlanan Veritabanı



Şekil 6.2 : Tekrar Eden Verilerin Çıkarılması Sonrası Veritabanı Yapısı

Şekil 6.2'de belirtien bu yeni tasarım ile alanlarda arama yapma, kolay erişim gibi işlemler kodlama tarafında daha rahat ve basite indirgenmiş oldu. Eğer herhangi bir detayı (örneğin firma adını) getirmek istemiyorsak sorgumuzda ona yer vermemiş oluyoruz. Ayrıca birden fazla aynı verinin tekrar edilmesini de bu şekilde önlemiş olmaktadır.

Ancak bu tasarımda bir ilaç hakkında arama yaparken öncelikli görmek istediğimiz alanlar yerine istemediğimiz tüm alanları getirmiş oluyoruz. Bu da geçici belleği sınırlı olan mobil cihazlarda boş yere alan işgal etmek manasına geliyor. Ekran tasarımlarımızda eğer daha basit alanları getirmek istiyorsak o alanları başka bir tabloda toplayarak o tabloya erişim sağlamak mantıklı olacaktır. Eğer detay istenirse diğer bilgileri çağırarak ve getirmek daha hızlı ve pratik olacaktır. Ekran tasarımları için olan gereksinimler aslında bizim veritabanı tasarımı için olan gereksinimlerimize de yön vermiş olacaktır.

Veritabanı yeni şekliyle tasarlandığında aslında ilaca belirginlik katan ana bilgiler bir tabloda toplanmış oldu. Tam da bu aşamada verinin gizliliği konusu ile ilgili olan gereksinimleri de düşünerek tasarımı güçlendirmek gerekmektedir. Tüm veritabanının şifrelenip bu şekilde saklanması ve sorgulandığında ise deşifre edilerek açılıp getirilmesi gerekmektedir.

Mobil işletim sistemlerinde veritabanı işlemleri dosya işlemleri ile benzerlik göstermektedir. NIST sertifikalı kriptografi algoritmaları kullanarak Android mobil cihazlar üzerinde çalışabilen dosya kriptolama uygulamaları geliştirilmiştir.[15] AES-128 ile yapılan dosya şifrelemelerinde 20 kat performans kaybı yaşandığı gözlenmiştir. Dosya işlemlerinde bu kadar fazla yükü olmasına rağmen veritabanı işlemlerinde %58 oranında daha olumlu sonuçlar alınmıştır. Dosya sisteminin blok boyutu ve I/O modu optimize edilerek %20 ile %57 oranında performans kazanıldığı gözlemlenmiştir. Veritabanında da kriptolamadan kaynaklı yükün azaltabilir olduğu görülmüştür.[14]

Dünyaca tanınmış mobil uygulamalardan biri olan Whatsapp uygulamasında bulunan mesaj, ses, görüntü gibi bir çok sosyal haberleşme ile ilgili bilgilerin şifreli tutulabilmesi üzerine bir çalışma yapılmıştır. Yapılan çalışmada AES-192 algoritması kullanılmıştır. Bu yöntem kullanılırken kriptoyu tekrar çözmek için gerekli bir şifre olmalıdır. Bu şifre sayesinde geri dönüşüm sağlanmaktadır. Zayıf olan noktası ise bu şifrenin yazılım içerisinde tutuluyor olmasıdır.[16]

Sonuç olarak veritabanı tasarımı her bilgiye hızlı erişmemizi sağlıyor ve güvenlik için veri şifrelemeyi sağlamaktadır. Veritabanı yapısı parametrik ve kolay erişim veri modeli üzerine kurulmuştur. Bir ilaç prospektüsünde bulunan tüm alanlar veritabanında bir sütun olarak yada map tabloları ile tutulmaktadır. Mobil cihazların disk alanlarının kısıtlı olduğundan bu cihazlar için büyük veri büyük problem demektir. Verinin kontrolsüz bir şekilde artması istenen bir durum değildir. Veritabanının tasarımının iyi yapılması ile verinin tekrar edilmesi de önlemiştir.

Uygulamadaki ilaç bilgileri SQLite veritabanı yapısında gömülü yerel veritabanında saklanmaktadır. Tüm veriler cihazın kendi üzerindeki diskte saklanmakta ve böylece uzaktaki herhangi bir veritabanına bağlanma gereksinimi duymamaktadır. Eğer veritabanında bir güncelleme yapılması gerekirse bunun için iki yol bulunmaktadır: Birincisi uygulamanın kendisi için uygulama marketinde çıkılacak olan bir güncelleme ile sağlanabilir ancak bu çok uygun bir yöntem değildir. İkinci yöntem ise veritabanının uzaktaki bir serverdan güncellenmesidir. Bu durumda uygulamanın güncellenmesine gerek kalmıyor ama uzaktaki sunucuda bulunan veritabanından kayıtları almak için bir web servis ihtiyacı doğuyor. Ayrıca bu web servisin gerçekleştirilmesi ve güvenliğinin sağlanması ihtiyacı doğmaktadır.

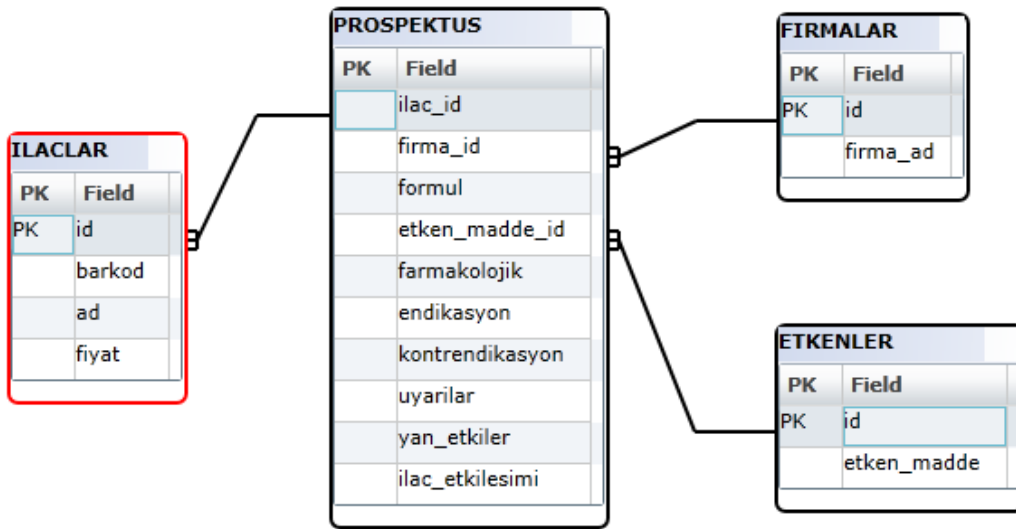
Mobil uygulamalarda karşılaşılan en yaygın problemlerden birisi veritabanı güvenliğidir. Bazı mobil cihazlar açık kaynak kodlu işletim sistemine sahiptir ya da kullanıcılar tarafından yönetici yetkisi ile erişilebilir durumdadır. Bu nedenle uygulamanızın güvenliğinden siz sorumlu olmuş oluyorsunuz. Cepİlaç uygulaması iki farklı veritabanı kullanmaktadır. Bunlardan birisi yaygın olarak bilinen ve mobil uygulamalarda kullanılan SQLite veritabanıdır, diğeri ise şifreleme gerektiren mobil uygulamalar için kullanılan SQLCipher [4] veritabanıdır.

SQLCipher açık kaynak kodlu, 256-bit AES şifreleme algoritması kullanarak SQLite veritabanı dosyalarını güvenli hale getiren bir kütüphanedir. SQLCipher birçok ticari ve açık kaynaklı ürünün kullandığı güvenli veritabanı çözümü için geliştirilmiştir ve bu yaygın kullanımı mobil ve gömülü sistemler içinde önemli hale gelmesini sağlamıştır. SQLCipher kendi şifreleme metodunu kullanmamaktadır. Bunun yerine tüm kriptolama algoritmalarında yaygın olarak kullanılan Open SSL libcrypto [5] yapısını kullanmaktadır. Veri şifreleme hem zaman hem de hafıza gibi bir çok kalemde maliyetin artmasına, bunun yanı sıra kısıtlı hafızası olan cihazlarda kilitlenmelere sebep olmaktadır. Sizin yapacağınız herhangi bir aramada eğer bir şifreleme ve çözme söz konusu ise çalışma zamanı çok artacaktır.

İlaça belirginlik katan anahtar alanları kolay ve hızlı erişim için bir tabloda toplanması uygun bulunmuştur. Bunun yanı sıra tüm verileri şifreleyerek tutmak için SQLCipher veritabanı kullanma kararı alındı. Daha önce de bahsedildiği gibi verinin büyük oluşu, şifrenmesi ve çözülmesi işlemlerinin mobil cihazlarda çok maliyetli oluşu tüm veriyi SQLCipher üzerinde tutma kararından vazgeçilmesine neden olmuştur. Verinin güvenliğinin sağlanabilmesi için şifreli veritabanından tamamen vazgeçmek mümkün olmayacaktır. Buradaki tasarımıımız performans sorunlarını da çözmek için iki veritabanını birlikte kullanmak yönünde oldu. Bu aşamada hangi alanların kriptolu veritabanında hangilerinin açık olarak bulunan veritabanında olacağını kararı verildi. Şifrenmemesi güvenlik açığı yaratabilecek anahtar alanlar SQLCipher veritabanı için seçildi. Diğer alanlar SQLite veritabanında açık olacak şekilde tasarlandı ve aralarındaki ilişkiler ikincil anahtarlar ile kuruldu. Buradaki asıl amaç prospektüste bulunan önemli alanların şifrenmesi için bir tasarım gerçekleştirmektir. Şifreli olması kararı alınan alanlar hızlı erişim için belirlenen anahtar alanlarla benzerlik göstermesi sebebiyle bu alanların bulunduğu tablo şifreli olan veritabanına taşındı. Diğer tablolar ise açık olarak saklanacak olan veritabanına tanımlandı ve bu iki tablo arasında belirli alanlar üzerinden ilişkiler kuruldu.

Veritabanı mobil uygulama kurulum dosyasında şifrelenmiş halde bulunuyor ve cihaz üzerine açılırken de şifreli halde açılması sağlanıyor. Böylece kurulmadan önce ya da kurulmdan sonra ele geçirilirse kullanılamaması sağlanmış olacaktır. Şifreleme işlemlerinde geri dönüştürme gereği olmayan ya da geri dönüştürme ihtiyacı olan durumlar söz konusudur. Geri dönüştürülemeyecek işlemler için algoritmalar (Örneğin md5) veriyi bozar ve veri o şekilde kalır. Güvenlidir çünkü dışarıdan herhangi bir şekilde veri tekrar elde edilemez. Geri dönüştürülebilir şekilde şifreleme algoritmaları (Örneğin AES) ile şifrelenen veriler için bir parola tanımlanır. Bu parolanın tanımlanması verinin tekrar geri elde edilebilmesi için gereklidir. Bir önceki yönteme göre dezavantajı ise güvenliğin sağlanabilmesi için bu parolanın iyi bir şekilde saklanması ihtiyacı olmasıdır. Özellikle mobil uygulamalarda veritabanının ve uygulamanın kodlarının cihaz üzerinde saklanması parolanın güvenliğinin sağlanmasını güçleştirmektedir.

Şifrelenen veritabanı bir parola ile tekrar açılarak gösterilmektedir. Deşifre işlemi maliyetli bir iş olması nedeniyle tüm veriler yerine belirlenen veriler şifrelenmiştir. Bu verileri tekrar açmak için AES algoritmasında bulunan parola yöntemi kullanıyor olacağız. Parolayı sabit bir değer olarak tutmak kod içerisinde ya da veritabanından ulaşılabileceğini gösterir. Bunun yerine parolayı dinamik olarak üreten bir kod parçası yazılarak parolanın direk kod içerisinde açık olarak tutulması önlenmiştir. Bu da veritabanında bulunan verilerin çalınmaması için alınan önlemlerdendir.



Şekil 6.3 : Şifrelenmiş Tablo Açık Tabloların Birlikte Olduğu Veritabanı Tasarımı

Şekil 6.3'de görülen kırmızı çerçeveli tablo SQLcipher e ait olan bir veritabanında şifreli olarak tutulmaktadır. Diğer tablolar SQLite veritabanında açık olarak tutulmaktadır. Veritabanı açıldığında hangi ilaç adına ve barkoduna ait olduğu belli olmayan onlarca prospektüs bilgisi elde edilmiş olacak ve bu da hiçbir anlam taşımayacaktır.

İlaçlarla ilgili yeni eklenecek bilgilerin açık olarak tutulan tablolara eklenmesinde hiçbir sorun olmayacaktır. Buradan yola çıkarak ilaçlara ait eş değer (muadil) ilaçları bulunabilmesi için tüm ilaçlara ait olan eş değer ilaçların eşleştirmelerini içeren bir tablo hazırlandı. bu tablo da aşağıdaki şekilde şifresiz olarak tutulan veritabanına eklendi. Şifreli veritabanında bulunan ilaç numaraları ile eşleştirilerek kolay şekilde sisteme entegre edilmiş oldu.

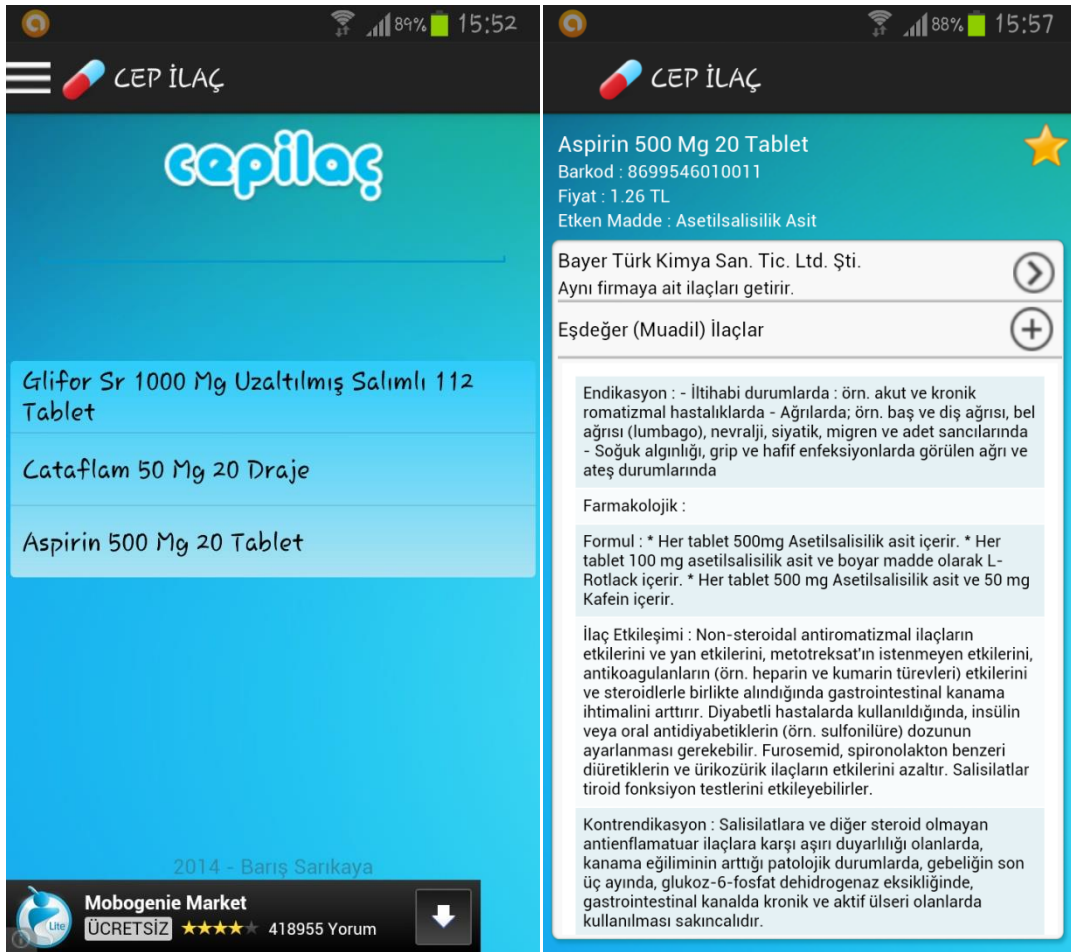
6.2. Ekran Tasarımları

6.2.1. İlaç Adı ile Prospektüs Bulma

Uygulamanın ana ekranında otomatik tamamlama özelliği olan bir ilaç adı arama alanı kullanıcıları karşılamaktadır. Bu ekran tasarımı mümkün olduğunca sadece tutulmaya çalışılmıştır. Bir arama motorunun arama yapmak için bir kutucuğa ve butona sahip olması nasıl yeterli oluyorsa ilaç aramak için de bu iki ekran elemanının yeterli olacağına karar verildi. Ancak ilaç adları bazen çok uzun isimlere sahipti bazen de çok özel isimleri olması nedeniyle hatırlanmakta zorluk çekiliyordu. Bunu düzeltebilmek için arama kutusuna yazılan kelime bir ya da iki harflik sapma ile ilaç isimlerinin içerisinde yer alıyor mu kontrolü yapılarak gerçekleştirilebilirdi. Bu çözümden kullanıcının tüm kelimeyi bir kaç harf eksikle bilmesi gerekiyordu ve bu da kullanıcıların çoğunun sağlık sektörü dışından, halktan, olacağını düşündüğümüzde çok zor olacaktı. Bunun yerine, arama yapacak kişinin ilaç adındaki ardışık gelen bir kaç harfi hatırlamasının yeterli olacağı bir çözüm düşünüldü. Bu da harfler yazıldıkça otomatik olarak tamamlamama özelliği olan ve yazılan harflere uyumluluk sağlayan ilaç isimlerini öneri listesi halinde sunan bir arama kutusu ile mümkün olacaktı. Bu arama kutusu tasarlandı ve kullanıcının önerilen ilaç isimlerinden istediğini seçerek sonuç detay sayfasına ulaşmasını sağlamaktaydı.

Ekran tasarımlarının yapılması esnasında android kolay erişilebilen, herkes tarafından yaygın olarak bilinen ve kullanılan XML teknolojisini kullanmaktadır. Bu sayede ekranda görünen objeler birer view sınıfı olarak tanımlanır ve kodlama tarafından kontrol edilmesi sağlanır. Aynı zamanda birçok obje kullanılabilmesine olanak sağlayan ViewGroup, objelerin ekrandaki pozisyonlarının ayarlanabilmesi için kullanılan layout gibi bir çok hazır kullanıcı arayüzü kalıbı Android SDK'sından erişilebilir durumdadır.[23]

Sonuç detay sayfasında ilaçla ilgili bir çok bilgiyi bulabilirsiniz. İlacın tam ismi, barkodu, fiyatı, etken maddesi, ait olduğu firma, formül, endikasyon, kontrendikasyon, uyarılar, yan etkiler, farmakolojik özellikler gibi birçok bilgisine bu detay sayfasından erişilebilmektedir. Şekil 6.4'de Cep ilaç uygulamasında bulunan otomatik ilaç adı tamamlama özelliği ile arama yapmayı sağlayan ekranı ve seçilen ilaca ait detaylı prospektüs bilgisi gösterilen ekranı bulabilirsiniz.



Şekil 6.4 : İlaç Adı ile Prospektüs Arama ve İlaç Detay Bilgi Sayfası

6.2.2. Otomatik Tamamlama Özellikli Yazı Kutusu

Mobil geliştirme ortamlarında kullanıcıdan giriş almak için kullanılan yazı kutusu elemanının değışiklik (onChange) olayını geçersiz kıldığımızda (override) ve bu metot içerisinde yazı kutusunun altına eklediğimiz liste görünümünün içerisinde belirlenen eşik değeri seviyesine göre filtrelenmiş veriler ile doldurduğumuzda istediğimiz ekran elemanını elde etmiş oluyoruz. Eğer giriş alanı boş ise alttaki listenin görünürlüğü kapatmak liste dolduğunda açmak gibi özelliklerinde eklenmesi gerekmektedir.

Eğer mobil geliştirme kitleleri bu özellikte bir elemana sahip değiller ise sorunu bu şekilde çözebiliyoruz ancak görünüm, ilave bir çok kodun yazılması ve çalışma performansı olarak tek bileşen olması daha iyi olacaktır. Bu açığı gören mobil işletim sistemi geliştiricileri tarafından (Android ve IOS) bu özellikte bir bileşen geliştirildi ve isminde otomatik tamamlanan giriş alanı verilmesi uygun görüldü (AutoCompleteEditText).[26] Bu yeni eklenen özellik Şekil 6.5'de gösterilmiştir. Cep ilaç uygulamasının ilaç arama ekranında kullanıcıları karşılayan tek kutu otomatik tamamlama özelliği olan bir giriş alanıdır.



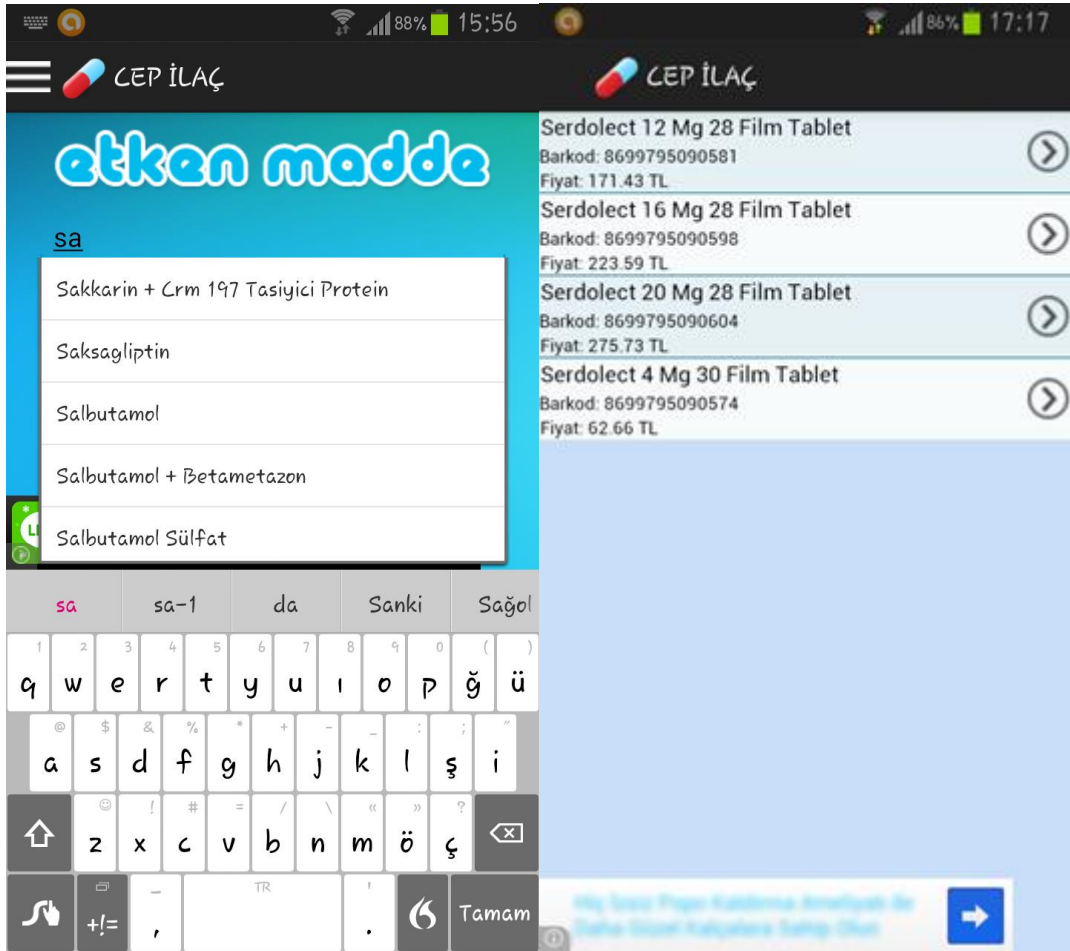
Şekil 6.5 : Android Standart AutoCompleteTextView Görünümü

Bu alan çalışma mantığı gereği öneri listesine getireceği veri listesinin sağlanmasını bekliyor ve ondan sonra sizin verdiğiniz eşik değerine göre öneri listesini dolduruyor ve sizin herhangi bir liste elemanını seçmeniz sonrasında belirtilen fonksiyona yönlendirilmektedir. Bu varsayılan özelliği veri setinin boyutu çok fazla olmadığında kullanmak çok rahat ve sorunsuz oluyor. Ancak veri seti büyüdüğünde çalıştırdığımız mobil cihazların sınırlı belleklere sahip olmaları nedeniyle kullanımda sorunlar oluşmaktaydı. Bunun yanı sıra performans sıkıntıları da ortaya çıkıyordu. Çünkü verilen liste üzerinde arama yapmak zorunda kalıyordu ve bellekteki verilere erişmeden sorunlar oluşuyordu.

Aslında mobil cihazların sahip oldukları RAM belleklerin boyutları yeterince büyük olsaydı bu problemlerle karşılaşılmayacaktı. Bu sorunu çözebilmek için otomatik tamamlama özellikli giriş alanının harf girildiğinde çağırdığı metodunda var olan listede filtre yapmak yerine girilen harflerle veritabanına giderek gerektiği kadar veriyi RAM belleğe alarak işlem yapılması sağlandı. Ancak her harf girildiğinde veritabanına gitmek de mantıklı olmayacaktı. Bu da daha önce bahsettiğimize benzeyen performans sorunlarına neden olmaktaydı. Son olarak otomatik tamamlama alanının harf değişimindeki çağrılan metodunda giriş iki harfe ulaştığında veritabanından sorgulayarak o iki kelimde olan veriler getiriliyor ve otomatik tamamlama alanının normalde kullandığı listeye doldurulmaktaydı. İki harften sonraki harfler için yapılan arama o liste üzerinden yapılıyordu. Bu şekilde ikiden fazla olan aramalar da veritabanına gidilmemiş oluyordu, ayrıca tüm veri seti çekilmeyip sadece iki harfle ilgili olanlar getirildiği için RAM bellekte yoğunluk oluşmuyordu. Öneri listesinde bulunan herhangi bir ilaç seçildiğinde ise belirtilen metot seçilen ilaca ait detaylar için veritabanına gidiyor detayları da sonuç sayfasında göstermek için yönlendirme sağlamaktadır.

6.2.3. Etken Madde ile İlaç Bulma

Etken madde, ilaç içeriğinde önemli rol üstlenen kimyasal bileşenleri ya da bileşen karışımlarını tanımlamaktadır. Etken madde ilaç üreticisi, ilacın piyasadaki adı ya da ilacın verilen/kullanılan dozajı değişse dahi belirli bir hastalık için hep aynı kalmaktadır. Doktor tavsiyesi olmadan ya da reçetesiz olarak alınan ilaçlarda ilacın adına dikkat etmek yerine içeriğindeki etken maddesinin ismine dikkat edilmelidir. Çoğu zaman doktordan önce eczacıya giderek belirttiğimiz bir rahatsızlık için eczacılardan ilaç istediğimizde hastalara verecek oldukları ilaçları etken maddesine bakarak vermektedirler. Bazen ilaçlarda, bunlar genellikle tablet olarak tanımlanan ilaçlarda olmaktadır, birden fazla etken madde yardımcı maddelerle birlikte genellikle toz şeklinde baskı ile veya kaplanarak katı halde olacak şekilde kullanılmaktadır.



Şekil 6.6 : Etken Madde Arama ve Etken Maddeye ait İlaç Listesi Ekranı

İlaç aramada kullanılan otomatik tamamlama özelliği olan giriş alanı etken madde arama ekranında da kullanılmıştır. Veritabanı tasarlanması aşamasında etken madde alanı ilacın prospektüsünün tutulduğu tabloda bir sütun olmaktan çıkarılarak ayrı tabloda tasarlanmış ve sadece referansı o tabloya yazılmıştır. Veritabanı için aynı etken maddenin yani aynı değer defalarca tekrar etmemesi iyi bir tasarımı işaret etmektedir. Ancak sadece tekrarları önlemeye fayda sağlamayıp etken madde listesini almamızda da kolaylık sağlamış oldu. Etken madde listesini veritabanından çekerek otomatik tamamlama özelliği ile kullanıcının seçmesi sağlandı. Seçilen etken madde sonrasında kullanıcı sonuç sayfasına yönlendirilmektedir.

Sonuç sayfasında birden fazla ilaç ismi dönmüş ise kullanıcı detaylı bilgilerini görmek istediği ilacı seçerek detay sayfasına yönlendirilir. Ana ekrandaki ilaç arama bölümünde ise herhangi bir ilaç listesi sayfasına yönlendirme olmaksızın bir arama yapıldığında ilacı seçip ilaçla ilgili bilgilerin bulunduğu detay sayfasına yönlendirilmektedir. Cep ilaç uygulamasında tasarlanan etken madde arama ve etken maddeye ait ilaç liste ekranları Şekil 6.6'da gösterilmiştir.

Detay bilgi sayfasında, ilacın tam ismi, barkodu, Türk Lirası cinsinden fiyatı, etken maddesi, üretici firması, eşdeğer ilaç listesi (başlangıçta kapalı olarak gelir) ve ilacın prospektüsünde bulunan tüm bilgileri bulabilirsiniz. İlaç adı ile arama yapmanız sonrası gelen ekranın aynısı kullanıcıyı karşılıyor olacaktır. Aynı etken maddeli ilaçların çok olması doktorlar için ve hastalar için avantajlı olacaktır. Çünkü seçenekler çoğalacaktır.

6.2.4. Endikasyon ile İlaç Bulma

Endikasyon kelime anlamı olarak, bir hastalıkta izlenecek tedavi yöntemi ve müdahalenin belirlenmesi, bir ilacın hangi hastalıklara ve hangi biçimde uygun olacağını veya bir operasyonun hangi koşullarda gerekli olduğunu ifade eder. Bir rahatsızlığınız olduğunda kullanacağımız ilacın prospektüsünde bulunan endikasyonlar bölümünde sizin rahatsızlığınızın, hastalığınızın adı yazıyor olacaktır.

Herhangi bir rahatsızlığınız olduğunda hangi ilacın size iyi geleceğini bilmeniz ne kadar güzel olacaktır. Ancak bir doktora danışmadan ilaç almanın sakıncaları unutulmamalıdır. Bunu yapabilmek için Cep İlaç uygulamasına endikasyon arama bölümü eklendi. Rahatsızlığın adını yazarak arama yapılabilir ve böylece bulunan ilaçlar bu rahatsızlık için kullanılabilir.

Endikasyon aramak için tasarlanan ekranda bir giriş alanı konumlandırıldı. Bu giriş alanına yazılan kelimeler ilaçların prospektüs alanlarında yazan kelimeler arasında aranıyor. Bu kelimeleri içeren ilaçların listesi kullanıcıya sunulmaktadır ve seçim yapması sağlanmaktadır. Arama ekranında bulunan giriş alanı otomatik tamamlama özelliğine sahip değildir. Çünkü bu alanda yazılabilecek seçenekler rahatsızlık çok fazladır. Kullanıcıları kısıtlamak yerine yazdıkları kelimeleri içeren kayıtları getirmek daha kullanışlı olacağı için tasarım bu yönde ilerlemiştir.



Şekil 6.7 : Endikasyon Arama ve Endikasyona Sahip İlaç Listesi Ekranı

İlaç aramada kullanılan otomatik tamamlama özelliği olan giriş alanının aksine endikasyon arama ekranında sade giriş alanı kullanılmıştır. Yazılan rahatsızlık prospektüs tablosunda bulunan endikasyon alanında aranır ve sonrasında kullanıcı sonuç sayfasına yönlendirilmektedir.

Sonuç sayfasında birden fazla ilaç ismi dönmüş ise kullanıcı detaylı bilgilerini görmek istediği ilacı seçerek detay sayfasına yönlendirilir. Ana ekrandaki ilaç arama bölümünde ise herhangi bir ilaç listesi sayfasına yönlendirme olmaksızın bir arama yapıldığında ilacı seçip ilaçla ilgili bilgilerin bulunduğu detay sayfasına yönlendirilmekteyiz. Ancak burada etken madde de olduğu gibi arada ilaçların listelendiği bir ekran gelmektedir. Cep ilaç uygulamasının Şekil 6.7'de gösterilen örnek ekranlarında bir endikasyon girilerek arama sonucuna ulaşılması gösterilmiştir.

6.2.5. Barkod ile İlaç Prospektüsü Bulma

6.2.5.1. Barkod Nedir?

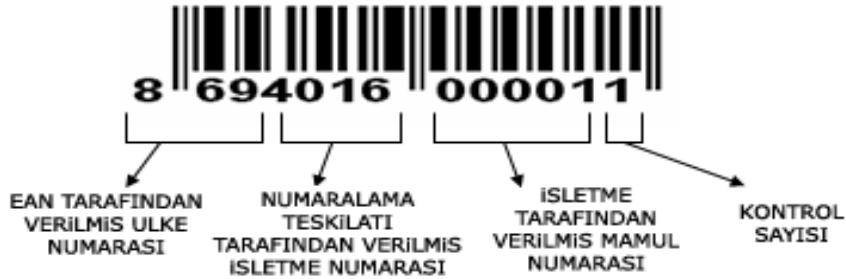
Barkod hayatımızda kullanılan bir çok ürünün üzerindeki beyaz zemine siyah bir şekilde basılmış, çizgi ve sayılardan oluşan bir özel bir sistemdir (Şekil 6.8). Barkod sistemleri kullanılmasıyla birlikte günlük hayatta ve iş hayatında büyük kolaylıklar sağlamıştır. En önemlisi alışverişlerin çileye dönmesini engelleyen barkod sistemi sayesinde ürünler barkod okuyucuya tutularak ürün fiyat girdisi otomatik bir şekilde ekrana yansımaktadır. Bu sayede onlarca, yüzlerce ürünü kısa sürede kasadan geçebilir ve ödemenizi yapıp alışverişlerden kurtulabilirsiniz. Aynı durum eczanelerde ilaç aldığımızda takibinin kolay yapılması içinde geçerlidir.



Şekil 6.8 : Örnek Barkod Etiketleri

Barkodların hayatımıza getirdiği diğer bir yenilik ise, gelişen teknolojinin de katkılarıyla ortaya çıkan barkod okuyuculardır. Barkod okuyucular sayesinde barkodlanan ürünler kolayca barkod okuyucu ile okutularak kayıt altında tutulabilir. Barkod okuyucular özel olarak hazırlanmış barkod programları sayesinde kolaylıkla çalışır. Donanımsal barkod okuyucular bir çok muhasebe, stok, satış gibi programlarla çalışmaktadırlar.

Barkodun başlangıcına göre üretim ülkelerini de öğrenebiliriz. Örneğin: “50” sayısı ile başlayan tüm ürünler İngiltere ürünüdür. Şekil 6.9'da da görüleceği üzere Türkiye'nin barkod başlangıç numarası ise 869'dur. Türkiye'den olan ürünleri tüketmek isteyen herkes 869 numarası ile başlayan ürünleri alabilirler.



Şekil 6.9 : Barkod Etiketinde Bulunan Değerlerin Anlamları

Barkod sistemi 1940'ların sonunda bulunmuş ve bu yıllara değin sürekli kullanılmıştır. Barkod sistemi geliştirilerek her alanda müşteriye kolaylık kullanılmıştır. Barkod sistemi sayesinde aynı zamanda en çok satan ürünler ve az satan ürünler rahatlıkla belirlenebilir ve güne, haftaya, aya dair dökümler rahatlıkla çıkartılabilir. Bir çok ürünün tanınması için bu kodlar gereklidir.

6.2.5.2. Barkod Teknolojileri

Barkod teknolojisi bir bilgi kodlama teknolojisidir. Barkodlar kodlanmış veriye optik okuyucular ile erişilmesini sağlayan arayüzlerdir. Günümüzde en sık kullanılan barkod türü, dik ve farklı kalınlıktaki çizgilerden oluşan doğrusal barkod türüdür. Doğrusal barkodların yoğun olarak kullanıldığı alanlar alışveriş ortamlarıdır: Ürün paketlerinde bulunan barkodlar ve optik okuyucular aracılığıyla veri tabanına aktarılan bilgi, ürünün fiyatına erişimi sağlar. Doğrusal barkodun iki boyutlu (2D) olarak geliştirilmesiyle, doğrusal barkoda kıyasla daha yüksek kapasiteye sahip 2D barkodlar elde edilmiştir. Doğrusal barkodlar 20 kadar alfanumerik karakter (harf ve rakam) kodlayabiliyorken karekodlar 4,000 civarı alfanumerik karakter kodlayabilmektedir. Günümüzde en bilinen 2D barkod teknolojisi karekod olarak isimlendirilir. Karekod teknolojisi esasen 1994 yılında otomotiv sektöründe üretim aşamasındaki taşıtların kodlanarak üretim süreçlerinin takip edilebilmesi amacıyla oluşturulmuş bir teknolojidir. Sonraki yıllarda karekod ISO tarafından belirlenmiş bir standart olarak yayınlanmıştır (ISO/IEC 18004: 2000/2006). Patenti ticari bir kuruluşa ait olmakla birlikte kullanımı özel lisansa tabi değildir, serbesttir.

Diğer firmalar benzer 2D barkod teknolojilerini farklı isimlerle geliştirmiştir (örn. Microsoft tag). İsimleri farklı olmakla birlikte, bu bildiride incelenen kullanımları açısından farklı 2D barkodların teknolojik altyapısı ve işlevleri benzerdir. Tüm barkodlar bilgiyi kodlarlar ve okuyucu cihazlar aracılığı ile bu bilgiye erişim sağlarlar. ‘Karekod’ teriminin ülkemize girişi eczacılık alanındaki kullanımı ile gerçekleşmiştir. Terim, ilk defa Sağlık Bakanlığı tarafından hazırlanan bir kılavuzda ‘datamatrix’ teriminin Türkçe karşılığı olarak önerilmiştir [6]. Diğer yandan ‘QR code’ terimi de halen kullanılmaktadır. Bu bildirinin aşağıdaki bölümlerinde terimin Türkçe karşılığı olan ‘karekod’ teriminin kullanımı benimsenmiş ve genel olarak 2D barkod teknolojileri karekod olarak adlandırılmıştır.

Barkod okutma üzerine birçok algoritma geliştirilmiştir. Bunlardan biri olan Texas Instruments üretimi olan işlemcilerde denenmiş olanıdır. Yapılan barkod okuma sistemi mobil cihazın kamerası, mobil uygulama işleyici, dijital sinyal işleme (DSP), görüntüleme cihazı kullanılarak geliştirilmiştir. 1D olarak bilinen EAN barkod ve 2D olarak isimlendirilen karekod (QRCode) şeklindeki barkodların okunabilmesi sağlanmıştır.[18] Yapılan denemeler sonrasında bu uygulama ile saniyede 66.7 çerçeve resim alınabildiği görülmüştür. Normal bir mobil cihaz kamerası ile işletim sistemi ve diğer sebeplerle nedeni ile saniyede 5-6 çerçeve resim alınabildiği bilinmektedir.[17]

6.2.5.2.1. Karekod Teknolojilerinin Kullanım Alanları

Karekod teknolojilerinin dünyada yaygınlaşmakta olan kullanım alanlarına baktığımızda ilaç reçetelerindeki kullanımlarının ötesinde kimlik ve kartvizitlerden etkinlik biletlerine, televizyon programları ve ürün broşürlerinden kütüphane uygulamalarına kadar geniş bir kullanım alanı yelpazesi ile karşılaşırız. Bu yaygın kullanımın altında yatan önemli bir etken, karekodların web sayfası adresi (URL bilgisi) gibi çok sayıda karakter içeren dizileri saklayabilecek kapasiteye sahip olması ve son kullanıcıların bu bilgiye akıllı cep telefonları ve PDA cihazları aracılığı ile erişebilmesidir.[19] Yazılı materyalde ya da TV ve bilgisayar ekranında bulunan karekodlar, kameralı cep telefonlarına yüklenen yazılımlar aracılığı ile taranabilmekte ve içerdikleri bilgi kodu çözülebilmektedir.

Örneğin, kullanıcı ürün kataloğunda gördüğü bir karekodu cep telefonu ile tarayarak ürün ile ilgili bir web sayfasına bağlanabilmektedir. Diğer bir deyişle günümüzde karekod teknolojisinin yaygınlaşmasına temel oluşturan en önemli unsurlardan birisi karekodların artırılmış gerçeklik uygulaması olarak öne çıkabilme potansiyeli ve bunun yarattığı uygulama alanlarının yaygınlığıdır. Ülkemizde de, dünyadaki gelişmeleri takiben, karekodların kullanım alanları hızla yaygınlaşmaktadır.

6.2.5.3.Cep İlaç Barkod Okuyucu

Cep İlaç uygulamasının ilaç arama bölümünün bir benzeri ilaç ismi yerine ilaç barkodunu kullanarak gerçekleştirildi. Ancak ilaç barkodlarını kullanıcıların bilmesi çok zordur. Bu nedenle ilacın barkodunu yazarak erişim sağlamak imkansızdır. İlaçların kutularını ya da şişelerini kaybetmemişsek ilacın barkodunu okutarak ilaçla ilgili bilgilere ulaşabiliriz. Bu sistemi uygulama içerisinde desteklemek için bir açık kaynak kodlu kütüphane kullanarak bir geliştirme yapıldı.

Cep İlaç uygulamasında ilaç bulmada kullanılan barkod okuyucu ZXing kütüphanesi kullanılarak yazılmıştır. Cep İlaç uygulamasının yükleneceği tüm cihazlarda gömülü kamera olduğunu ve kamerayı bir barkod tarayıcı olarak kullanabildiğimizi düşünürsek barkod görüntüsünü almamız ve okumamız çok zor olmayacaktır. Ancak bu görüntüyü almanın ardından bu görüntüyü analiz etmek ve elde edilen sonucu bir sayısal değer olarak alabilmek gerekmektedir.

ZXing kütüphanesinin mobil cihazlarda iki tip kullanım şekli bulunmaktadır. Bunlardan birincisi ZXing kütüphanesini uygulamanın geliştirildiği projeye ekleyerek gerekli ara yüz metotları ile ZXing geliştiricilerinin yazmış olduğu uygulamayı çağırmasıdır.[20] Yine ara yüz metotları ile yapılan çağırım sonrası oluşan sonucun bir şekilde alınması gerekir. Alınan sonuç değeri başka bir metoda parametre geçirilerek o ilaca ait prospektüsün bulunmasını sağlamaktadır. Bu kullanımda barkod okuyucu bölümüne girildiğinde uygulama kütüphanenin ücretsiz olarak uygulama marketinde bulunan uygulamasının mobil cihazda kurulu olup olmadığının kontrolünü gerçekleştirmektedir. Uygulama kurulu ise bu uygulamayı açarak kullanıcıyı bu uygulamaya yönlendirmektedir. Barkod tarama işleminin tamamlanmasının ardından kütüphanenin barkod okuma uygulaması otomatik olarak kapatılıp kullanıcı tekrar Cep İlaç uygulamasına yönlendirilmektedir. Uygulama kurulmamış ise kullanıcıya uygulamayı kurmak isteyip istemediği sorulmaktadır. Eğer devam etmezse bu özelliği kullanamayacağı kullanıcıya bildirilir. Kurmayı kabul ederse kullanıcı uygulama marketinde barkod okuyucunun bulunduğu sayfaya yönlendirilir.

Kurulum sonrası Cep İlaç uygulaması içerisindeki barkod okuma özelliği aktif hale gelecektir. Bu seçeneğin dezavantajı olarak başka bir uygulamanın kurulmasına bağımlı olarak çalışmasıdır. Bu da kullanıcıların bu özelliği kullanmaktan vazgeçmesine neden olabilmektedir. Kütüphanenin ikinci kullanım şekli ise açık kaynak kodlu olması nedeniyle kütüphane de kodları açık halde bulunan sınıfları kendi projemizin içerisine gömerek kullanma şeklindedir.[20] Ancak bu durumda geliştirme süreci uzamaktadır ve geliştirilmesi gereken kod miktarı artmaktadır. Bununla birlikte uygulama kendi içerisinde gömülü bir barkod okuyucuya sahip olduğu için bir başka uygulamaya ihtiyaç duymamaktadır. Bu kullanım bir çok kullanıcı açısından tercih edildiği için bu yöntem değerlendirilmiştir. Kullanıcılar bir uygulama kullanabilmek için başka bir uygulamayı cihazlarına kurmayı istememektedirler. Şekil 6.10'da Cep ilaç uygulamasının içerisinde bulunan barkod okuma özelliği sayesinde bir ilaç barkodunun nasıl tarandığı gösterilmektedir.

Barkod okuma esnasında mobil cihazı hareket ettiren kişinin el hareketleri okuma kalitesini ve performansını etkilemektedir. Eğer bir optimizasyon yapılmamış ise mobil cihazın kamerasını yönlendiren kullanıcının elini tamamen hareketsiz hale getirmesi beklenir. Ancak bu durum neredeyse imkansız denebilecek kadar zor bir durumdur. Bu nedenle bulanık olarak alınan resim çerçeveleri iyileştirmeler yapılarak değerlendirilir.[22] Yapılan görüntü iyileştirmesi sonrasında daha çabuk ve hızlı görüntü tanıma yapılabilmektedir.



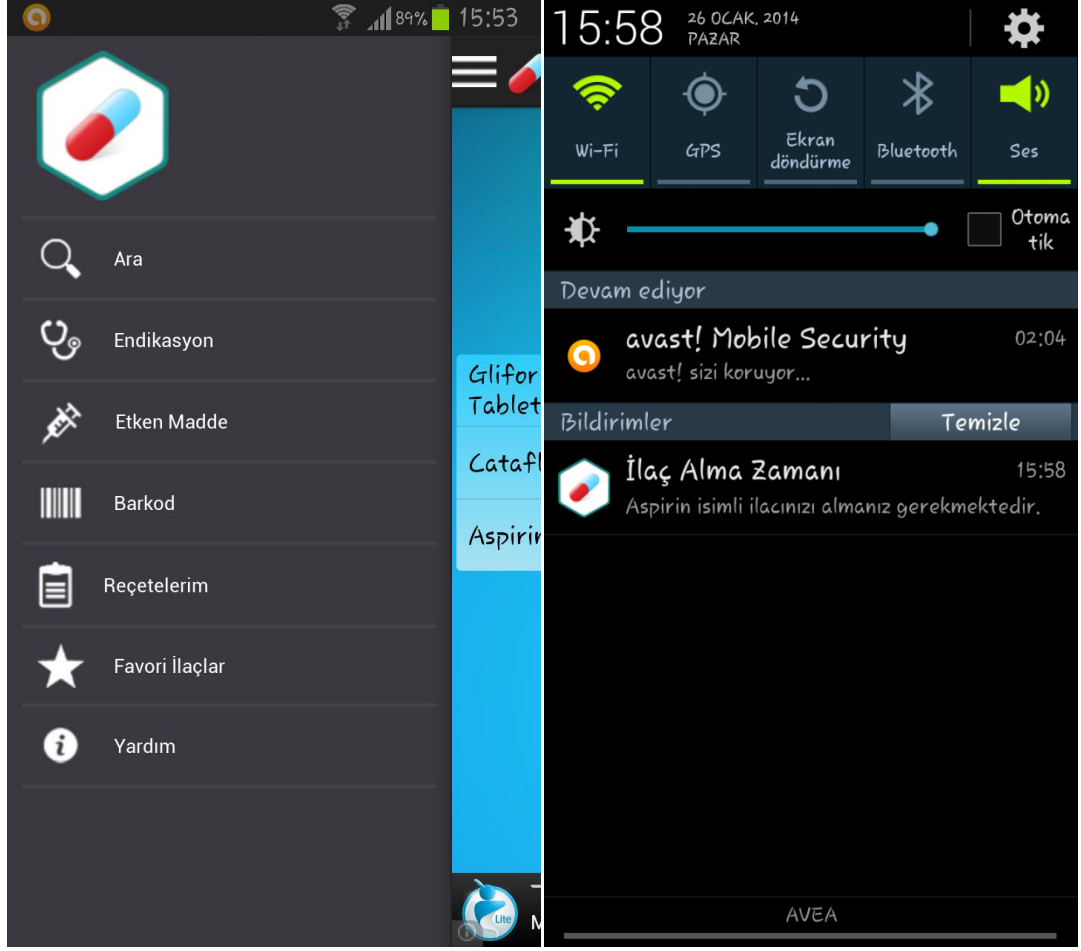
Şekil 6.10 : Barkod Okuyucu Çalışması ve Tarama Ekranı

6.2.5.4.ZXing ("Zebra Crossing")

ZXing (anlamı "Zebra Crossing") açık kaynak kodlu, çoklu 1D/2D format desteği olan Java ile geliştirilmiş ancak diğer dillere de ara yüzü olan[20] görüntü işleme kütüphanesidir. Kütüphanede herhangi bir sunucu ile haberleşmeksizin, mobil cihazlar üzerinde hali hazırda bulunan kamera kullanılarak cihaz üzerinde barkod tarama ve okuma gerçekleştirilmesi hedeflenmiştir. Ancak bu proje masaüstü ve sunucularda da barkod çözme, oluşturma işlemlerinde kullanılabilir. [21]

6.2.6.Firma ile İlaç Bulma

İlaçların ait oldukları firmalara göre genelde arama gereği duyulmaz ancak ilaçların firmalarını bilmek bazen gerekebilir. Firmalara göre arama seçeneği ya da ekranı Cep İlaç uygulamasına eklenmemiştir. Daha sonra böyle bir istek olması durumunda ekleyebilmek adına ilaç firmalarını ayrı bir tabloda tutarak hem veri tekrarının engellenmesi hemde bu alana göre arama yapılabilmesi sağlanmıştır. Artık veritabanı tasarımımızın alt yapısı firma adıyla aramayı desteklemektedir. İlaç arama ekranından seçilen ilacın, etken madde ile ilaç arama ekranında gelen ilaçların, endikasyon ile ilaç arama ekranından seçilen ilacın detaylarının gösterildiği sonuç sayfasında ilacın hangi firmaya ait olduğunu gösteren bir satır bulunuyor. Bu satır seçilerek ilacın ait olduğu firmanın sahip olduğu diğer ilaçlara erişebilirsiniz.



Şekil 6.11 : Sliding Menü ve İlaç Hatırlatma Özelliğinin Gönderdiği Bildirim Ekranı

6.2.7. Favori İlaç Listeleri

6.2.7.1. Çok Arama Yapılan İlaçlar

Genellikle kullanıcılar aradıkları ilaçların çoğunu tekrar aramaktadırlar. Eğer çok sık aranan bir ilacınız var ise bunun ismini, etken maddesini ya da endikasyonunu tekrar yazmak sizin zaman kaybetmenize ve uğraş vermenize neden olacaktır. Bu nedenle kullanıcılar arama yapıp buldukları ilaçları favori listelerine ekleyerek kendilerine özel çok aranan ilaçlar listesi yapabilirler. Daha sonra favori ilaçlar menüsünden (Şekil 6.11) istedikleri ilacın detay bilgi sayfasına hızlı şekilde erişim sağlayabileceklerdir.

6.2.7.2.Son Arama Yapılan İlaçlar

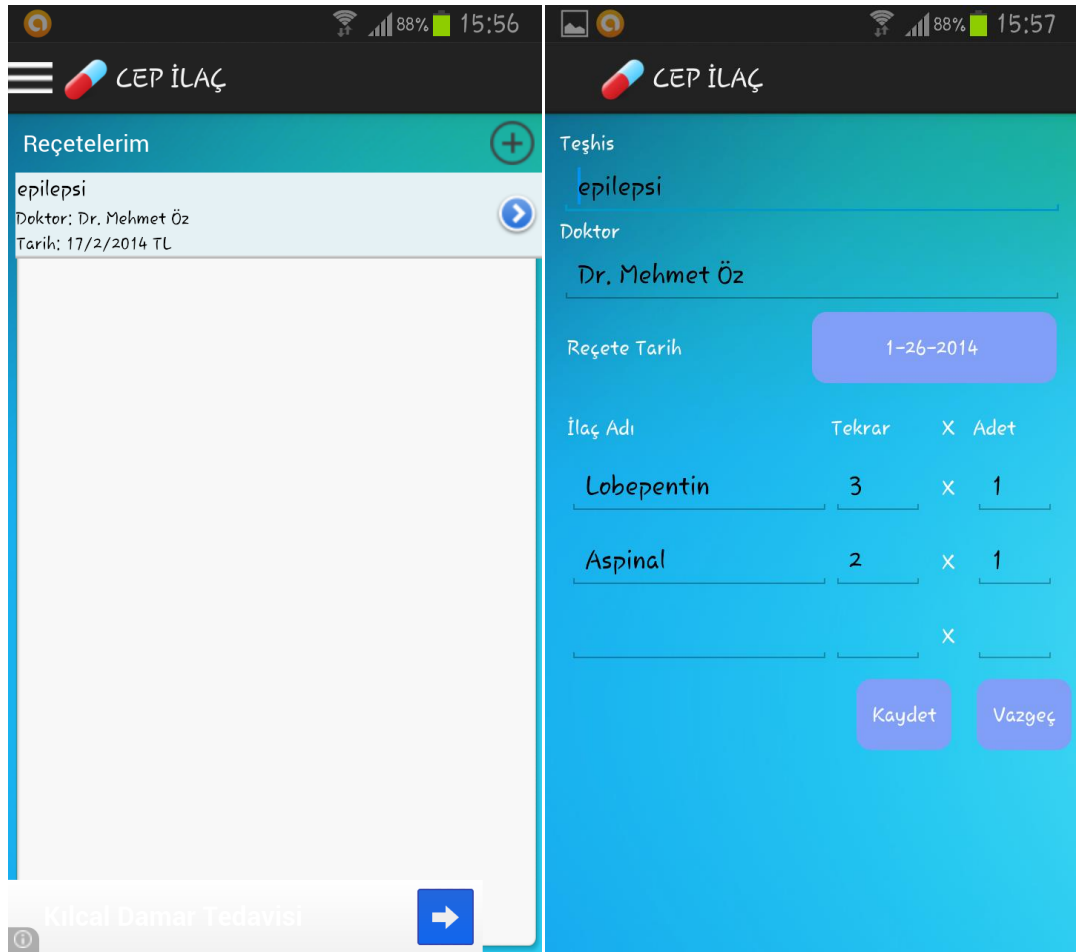
Kullanıcılar çok sık aradıkları ilaçları favori listelerine ekleyerek hızlı bir şekilde erişebiliyorlar. Ancak favori listeleri kullanıcıların insiyatifinde doldurulan bir bölüm olması nedeniyle dolu olma garantisi bulunmamaktadır. Sık aranan ilaçlara kolay erişilebilmesi için kullanıcının ekleyip eklememesine bakılmaksızın uygulamanın bunu sağlaması kullanım kolaylığı sağlayacaktır. Kullanıcılar ilaç arama ekranına geldiklerinde en son aradıkları ilaçlara erişebilirlerse aslında sık aradıkları ilaçlara da erişmiş olacaktırlar. Bu işlemi kullanıcıya bırakmaksızın son aranan belirli sayıdaki ilaç isimlerini son arananlar listesinde saklayarak kullanıcıya kolay bir erişim sağlanmış olacaktır. Bu liste tabiki favori ilaçlar listesi gibi istenildiği sayıda ve belirlenen ilaçlara göre değil arama yapılan son ilaçlara göre beslenecektir.

6.2.8.Reçeteni Oluştur

Reçete oluşturma sayfasında aslında gerçek hayatta kullandığımız reçetelerin bir benzetimi gerçekleştirilmeye çalışılmıştır. Klasik yöntemde yazılı olan reçetenin yedek nüshasının alınıp kişi tarafından kayıt altında tutulma ve farklı hastane, farklı doktor muayenesinde öngörü açısından yazılı bir belge oluşturmaktaydı. E-Reçete sistemine geçilmesiyle klasik reçete yazımı zamanından kazanç ve yapılabilecek tıbbi hata (malpractice) riskini azaltma gibi faydalar sağlasa da kişi açısından kişisel veri toplamada kısıtlılık getirmiştir. Uygulama, reçetenin hekim tarafından belirlenmesinin ardından kişi kendi reçetesini tarihi ve tanılarıyla kaydedip, daha sonra kullanılmak üzere veritabanı oluşturulmasına imkan sağlar.(Şekil 6.12) Uygulamadaki “Hatırlatıcı” sistem ile kişi isterse belirleyeceği başlangıç saati ile ilaçların bitimine kadar doz alım zamanını hatırlatma, ilaçların alınmadığı durumlarda not alınıp ileriye dönük veri kaynağı olabilme gibi faydalar sağlamaktadır.(Şekil 6.11)

6.3. İlaç Arama İstatistikleri

Yapılan aramaların oranları ve sayılar kullanıcıların kendileri için çok şey ifade etmeyebilir. Ancak mobil dünya kullanıcılarının birbirlerinden haberdar olması günümüzde çok yaygın olarak kullanılmaktadır. Bunu en çok sosyal paylaşım ağlarında görebilmekteyiz. Dünyanın herhangi bir yerindeki mobil uygulama kullanıcılarının yazdıkları, yaptıkları, paylaştıkları her veri artık ortak bir noktada toplanıp istatistiklerin tabanını oluşturmaktadır. Cep İlaç uygulamasında da aranan ilaç isimleri, endikasyon isimleri, etken madde isimleri gibi bir çok veri kayıt altında tutulabilmektedir. Bu verilerin kayıtları mobil cihazların içerisinde saklanabilmektedir. Bu istatistik kayıtlarının uzaktaki bir web server üzerinde toplanması ile anlık olarak bir internet sitesinden izlenebilir. Mobil uygulamada hem bir server ihtiyacı doğacağı için hem de sürekli internet erişimi ve cihaz veri erişim izni gerektireceği için uygulamada sadece kayıt (log) seviyesinde bırakılmıştır.



Şekil 6.12 : Kayıtlı Reçetelerim ve Reçete Tanımlama & Güncelleme Ekranı

7. MOBİL UYGULAMANIN YAYINLANMASI

Mobil uygulamaların geliřtirmesinin yani yazılım yařam dngüsünün tamamlanmasının sonrasında uygulamanın son kullanıcılar ile tanışacađı ortama tařınmasına yayınlama denilmektedir. Uygulamayı yayınlamaya hazırlarken dikkat edilmesi gereken bir ok husus bulunmaktadır. Android mobil uygulamaları Android Google Play uygulama marketi aracılıđı ile kullanıcılara sunulmaktadır. Google geliřtirdikleri uygulamaları markete koyabilmeleri iin geliřtiricilere bir sistem sunmuřtur. Bu sistem Android uygulama geliřtiricileri alanı (Android App Developer Zone) olarak adlandırılmakta ve belirli bir ücret karřılıđında üye olunarak sisteme girilebilmektedir.[27] Yayınlamak istenen uygulamanın Google politikalarına uygun olarak imzalanması (sign) gerekmektedir. Yaratılmıř olan ve uygulamamıza ait olacak sertifika ile uygulamayı imzalıyoruz.[28] Sonrasında Android iřletim sisteminde alıřtırılabilir, imzalı bir apk uzantılı dosya oluřacaktır. Bu dosya kullanıcı giriři yapılarak Android Google Play ortamına yklenip tm kullanıcıların indirilebilmesi sađlanmış olur.

Android iřletim sisteminin bulunduđu mobil cihazlarda eriřim yetkisi verilmemiř olan alanlara illegal yntemlerle eriřim sađlanabilmektedir. Bu řekilde olan cihazlara root cihazlar adı verilmektedir, bu ismin verilmesinin nedeni ise Android iřletim sisteminin linux tabanlı bir iřletim sistemi olması ve linux iřletim sistemlerinde yetkili kullanıcıya root isminin verilmiř olmasıdır. Root olarak ayarlanan cihazlarda ykl olan uygulamaların ieriklerine eriřim sađlanabildiđi bilinmektedir. Bu tr cihazlarda yklenen uygulamalara ait apk dosyalarına da eriřilebilmektedir.

Apk dosyalarının içeriğine decompile yapabilen araçlar sayesinde ulaşılabilir ve uygulamanın kodları görülebilir duruma gelebilmektedir. Uygulamanın kendisinin, verilerinin ve kodlarının güvenliği de yayınlama aşamasının dikkat edilmesi gereken önemli hususlarındandır.[29] Uygulama güvenliğinin sağlanabilmesi için kodların ve verilerin güvenliğinin alınması en önemli adımdır. Kodları karıştırarak kodlama algoritmalarının çalınması ve içerisinde bulunan erişim bilgilerinin alınmasını önlemek için ProGuard[30] isimli kütüphane kullanılmasına karar verilmiştir. Bu kütüphane sayesinde kodlar ele geçirildiğinde anlamsız karakterler ve dosyalar görüleceğinden gerçek bilgiler gizlenmiş olacaktır. Bu çalışmada oluşacak tek handikap herhangi bir sorunda hangi kod satırında hata oluştuğunu anlayamamaktır.

8. MOBİL UYGULAMA KULLANIM SONUÇLARI

Mobil Uygulama ile ilgili yüklenme durumları, kullanım istatistikleri, kullanıcı yorumları, kullanıcılar tarafından verilmiş oylar gibi kullanım ve kullanım sonrası gelen geri bildirim durumlarına bu bölümde yer verilecektir.

Cep İlaç mobil uygulaması gerçekleştirme sonrası her Android uygulamasında olduğu gibi Android Google Play Uygulama Marketi 'nde kullanıcıların hizmetine sunulmuştur. Her yazılım geliştirme sonrası son kullanıcıya ulaştırılır. Bu bölüm uygulama geliştirme yaşam döngüsünde bulunan Test ve Yaygınlaştırma adımının bir sonucudur. Sonrasında ise yine bu döngünün bir adımı olan bakım ve geri besleme diye adlandırılan işlemler devreye girer. Kullanıcıların deneyimleri, istekleri, bakış açıları sayesinde bir çok konuda geri bildirim alınmış olur.


Bahsi geçen kullanıcı verilerinden bir tanesi kullanıcı oylarıdır. Bunlar sayesinde satın almayı düşündüğümüz bir ürün hakkında fikir edinebilir, tatilde gitmeyi planladığımız yer ile ilgili bilgi edinebilir ya da burada olduğu gibi indirip kurmayı düşündüğümüz bir uygulamanın faydalı olup olmadığı konusunda bize yardımcı olabilir. Şekil 8.1'de Cep İlaç uygulamasının kullanıcı oyları bulunmaktadır. Uygulamayı mobil telefon ya da tabletine kuran kullanıcılardan 1.151 kişisi uygulama ile ilgili oy kullanmıştır. Kullanıcıların kullandıkları oyların tamamı 5 üzerinden 4,53 ortalama bir değer gelmiş ve her geçen gün bu oran yükselmektedir. Uygulamanın tez kapsamında yapılan bazı özellikleri kullanıcılar ile paylaşılan versiyona eklenmemiş sade olan kısmı kullanıma açılmıştır. Ona rağmen kullanıcılardan gelen oyların en yüksek olan 5 yıldızda yoğunlaşmış olması kullanıcıların büyük çoğunluğunun memnuniyetini kazandığını göstermektedir.



Şekil 8.1 : Cep İlaç Uygulaması Kullanıcı Oyları

Android Google Play uygulama marketinde kullanıcılar oy vermenin yanı sıra isterler ise uygulama hakkında fikirlerini de yazabilmektedirler. Kullanıcıların yorumları uygulama marketinde herhangi bir ekleme ya da silinme olmaksızın yayınlanmaktadır. Tüm dünyadaki kullanıcılar ile birlikte uygulamanın sahibide bu yorumları websitesi üzerinden görebilmektedir. Uygulamanın sahibi kullanıcıların yorumlarını silememektedir sadece o yorumlara cevap yazma hakları bulunmaktadır. Şekil 8.2'de kullanıcıların yaptıkları yorumlardan bazıları bulunmaktadır.

Android uygulama marketi uygulamasını sunmak isteyen uygulama geliştiricilere uygulamalarını yönetebilmeleri ve izleyebilmeleri için yönetim panelleri olan bir arayüz sunmaktadır. Bu arayüzde uygulamalar ile ilgili birçok bilgiye ulaşılabilir. Şekil 8.3'de uygulamanın kullanıcılar tarafından yüklenme sayılarının zamana bağlı grafiği bulunmaktadır. Bu bölümde günlük uygulama yüklenme sayısı, cihazdan silinme sayısı gibi yükleme ile ilgili birçok istatistiksel veriye ulaşılabilir.

☰  CEP İLAÇ ✓ Published ▾

REVIEWS

Filters ▾ Page 1 of 11 ◀ ▶ Go to page Go

★★★★☆
App version 2.2
Galaxy Ace2 (GT-I8160) **omer karan ayra** on 14 Dec 2013 at 16:21
[Auto-translated from Turkish](#)
Güzel Amacını yerine getirebilen bir uygulama 4 yıldız vermemin nedeni bazı ilaçların olmaması
[Reply to this review](#)

★★★★★
App version 2.2
W699 **Çiğdem Demir** on 14 Dec 2013 at 12:47
[Auto-translated from Turkish](#)
Şahane Çiğdem Demir Uzun zamandır kullanıyorum. aradığım ilaçları bu güne kadar buldum. Hastalık ismini yazınca da ilgili ilaç bulunuyor. İnternetsizde çalışır. Tavsiye ederim Yapana teşekkürler.
[Reply to this review](#)

★★★★★
App version 2.2
Galaxy S3 (m0) **Beyza Sahin** on 12 Dec 2013 at 08:19
[Auto-translated from Turkish](#)
Guzel Ben cok begendim elinize saglik. Begenmeyenler de madem daha iyisini yapsinlar. Evet eksikleri olabilir ama duzgun bi dille soylenebilir. Emeginize saglik.
[Reply to this review](#)

★★★★★
App version 2.2
LT22i (LT22i) **Samed Abdullatif** on 11 Dec 2013 at 21:28
[Auto-translated from Turkish](#)
Tesekkurler Elinize saglik, cok faydali bir uygulama
[Reply to this review](#)

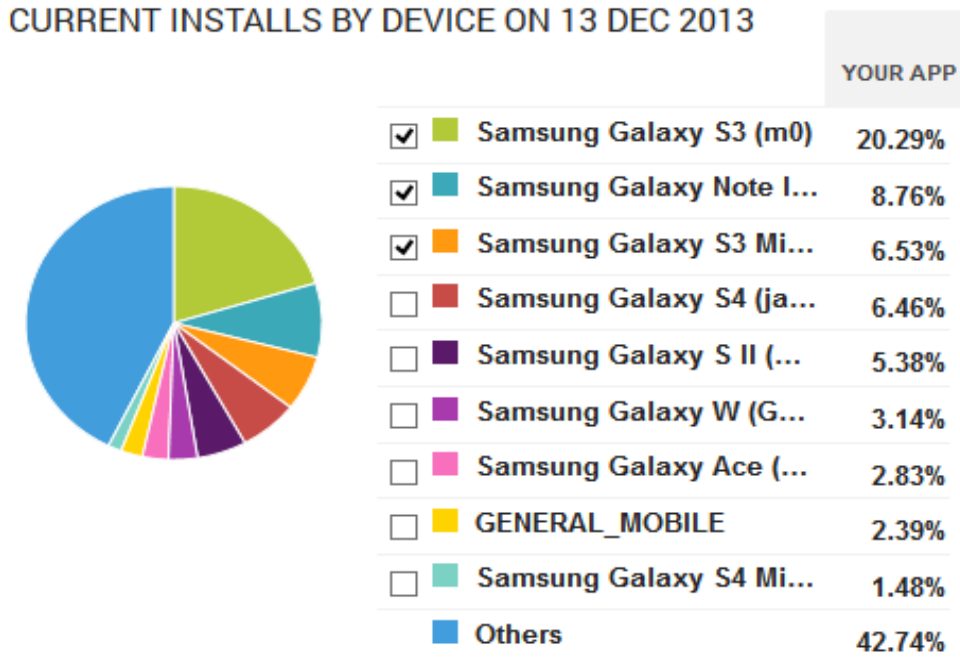
★★★★★
App version 2.2
Galaxy Note II (t03g) **seray kucuk** on 9 Dec 2013 at 15:38
Excellent ! perfect application to learn drugs. Thanks
[Reply to this review](#)

Şekil 8.2 : Cep İlaç Uygulamasının Kullanıcı Yorumları



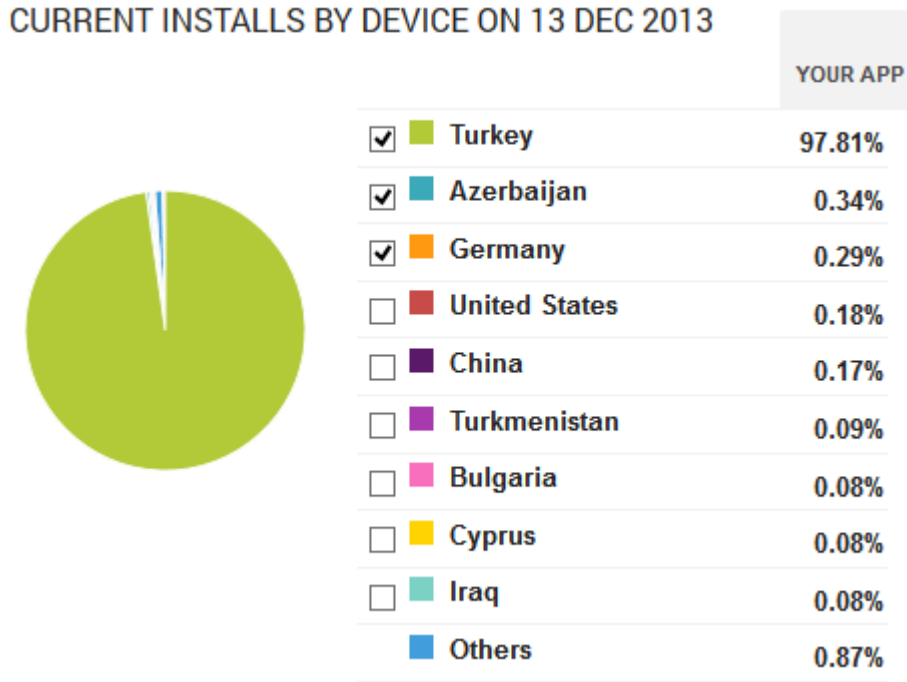
Şekil 8.3 : Cep İlaç Uygulaması Zaman Göre Yüklenme Sayısı

Android geliştirici arayüz sayfasında kullanıcıların uygulamayı yükledikleri cihazların tipleri konusunda da uygulamayı geliştiren kişiye fikir veren bir bölüm bulunmaktadır. Şekil 8.4'de gösterildiği üzere hangi cihazlarda uygulama daha yoğun kullanıldığını bildirmektedir. Bu sayede geliştirici yapacağı son kabul testlerini hangi cihazlarda yoğun olarak yapması gerektiği kararını alabilmektedir.



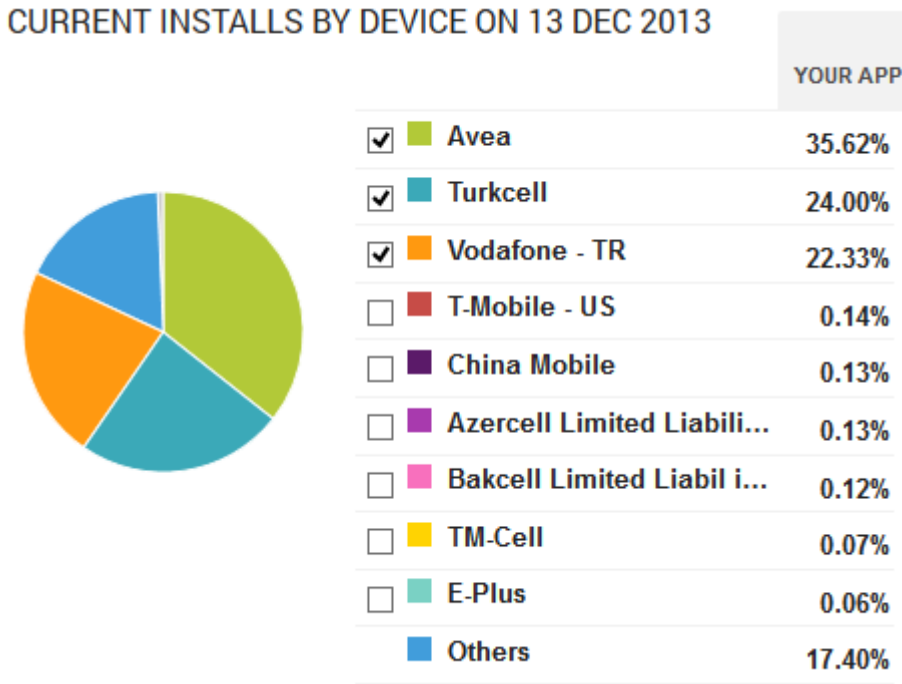
Şekil 8.4 : Cep İlaç Uygulamasının Cihaz Bazlı Yüklenme Oranları

Android geliştirici arayüz ekranında bir başka özellik kullanıcıların hangi ülkelerden bu uygulamayı indirdikleri bilgisinin gösterilmesidir. Bu özellik sayesinde uygulama geliştiriciler yükledikleri uygulamaların uluslararası kullanılıp kullanılmadığını, hangi ülkelerde kullanıldığını öğrenebilirler. Bu sayede o ülkelere özel geliştirmeler yapabilir yeni özellikler eklemek için geri bildirimler almış olabilirler. Şekil 8.5'de de görüldüğü üzere Cep ilaç uygulamasını indirip cihazına kuran kullanıcıların çok yüksek bir çoğunluğu (%97.8) Türkiye'den indiren kullanıcılardan oluşmaktadır. Bu uygulamanın içerdiği bilgiler ve dili sebebiyle Türkiye'den çok kullanıcıya sahip olması şaşırtıcı bir sonuç değildir. Uluslararası bir yapıya sahip uygulamalardaki geliştiricilere kullanıcıların durumları ile ilgili fikir vermesi açısından bu bölüm de önemli bir yer tutar.



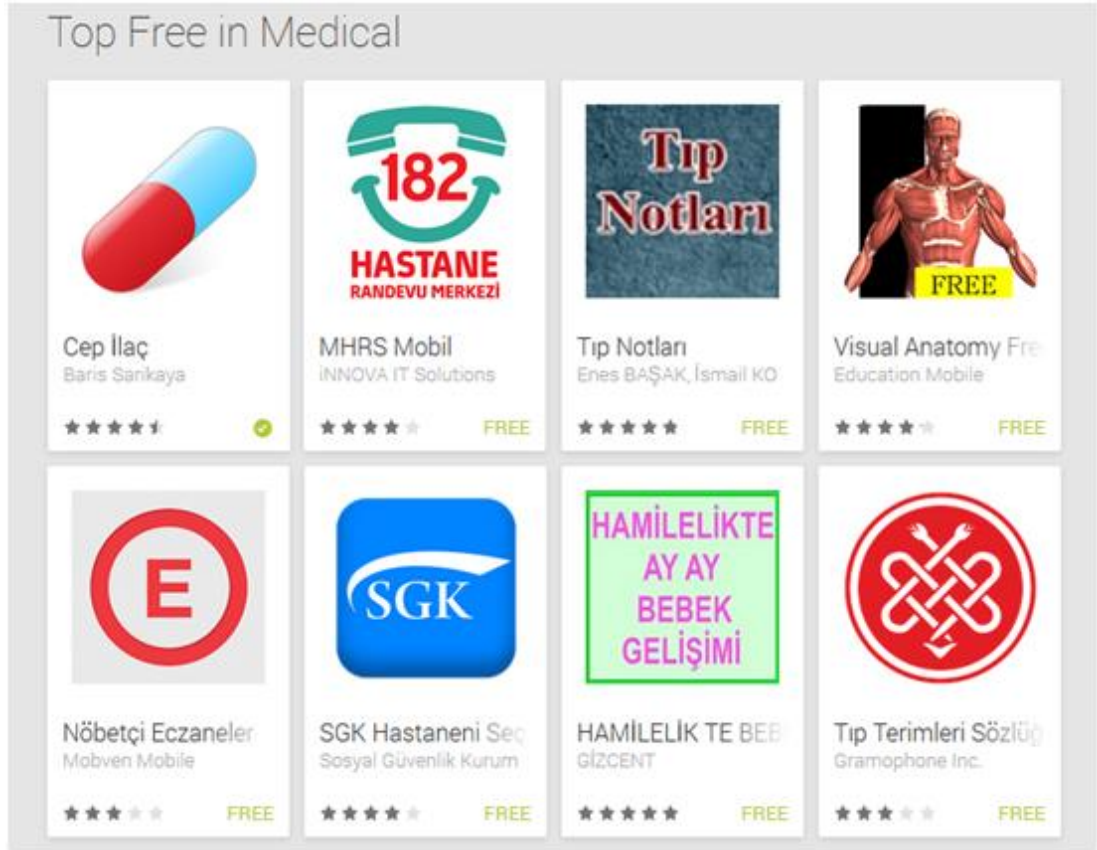
Şekil 8.5 : Cep İlaç Uygulamasının Ülke Bazlı Kullanılma Oranları

Android geliştirici arayüz ekranlarından bir diğer ekran ise kullanıcıların hangi operatör kullanarak uygulamayı indirdiklerini gösteren sayfadır. Bu sayfada kullanıcıların hangi operatörden uygulamayı kullandıkları bilgisi geliştiriciye istatistiksel bilgi sağlamaktadır (Şekil 8.6). Cep ilaç uygulaması açısından baktığımızda hangi operatörden kullanıldığı gerekli bir parametre gibi görünmemektedir. Ancak Cep ilaç uygulaması ilaç bilgilerini internet üzerinden anlık olarak alan bir uygulama olsa idi bu sayfadaki kullanım bilgileri bize fikir verecekti. Çünkü mobil cihazların anlık veri alış verişi operatörlerin internet hizmeti üzerinden olmaktadır ve bu hizmetin maddi olarak karşılığı operatöre göre değişiklik göstermektedir. Kullanım ücreti yüksek olan operatörde daha yüksek bir kullanım söz konusu ise bununla ilgili, verileri internetten alıp vermeyi durdurma ya da bir önbellek (cache) mekanizması kurma gibi önlemler almamıza yardımcı olabilmektedir.



Şekil 8.6 : Cep İlaç Uygulamasının Operatör Bazlı Kullanım Oranları

Android Google Play uygulama marketinde yüklenen uygulamalar belirli kategoriler altında tutulmaktadır. Belirlenmiş 30 civarında kategori bulunmaktadır. Cep ilaç uygulaması "sağlık" kategorisi altında yayınlanmıştır ve bu kategori altından bulunabilmektedir. Aynı zamanda her kategoride bulunan ücretli ve ücretsiz uygulamalara, bu uygulamalardan en çok indirilme oranı olanlara da marketin bu sayfalarından hem internet üzerinden hem de mobil cihazlardan erişilebilmektedir. Şekil 8.7'de Android uygulama marketinde sağlık kategorisinin en çok indirilen uygulamaları görünmektedir. Günlük olarak sıralamanın değişebildiği bu listede Cep İlaç uygulaması uzun süre ilk sırada yer almıştır.



Şekil 8.7 : Andorid Google Play Sağlık Kategorisi En Çok İndirilenler Listesi

KAYNAKLAR

1. Sectoral Research and Strategy Development Office, “Üç Aylık Pazar Verileri Raporu”, 4th Quarter of 2012, Information Technologies and Communication Agency of Turkey, March2013.
2. HP Application Lifecycle Management (ALM) Reference Document.
3. ESURVERCAD, Blog Survey Drawing, www.esurveycad.com.
4. Full Database Encryption for SQLite. <http://sqlcipher.net/>.
5. Multi format Barcode Image Processing Library. "<http://zxing.org>", accessed on May 2, 2013 .
6. Taking the Pulse. Manhattan Research Group
<http://manhattanresearch.com/Products-and-Services/Physician/Taking-the-Pulse-U-S>, 06 Ocak 2013
7. 2012 Q2 Türkiye ve dünya mobil dataları kapsamlı mobil sektör raporu, 12 Eylül 2013
8. "Bilişim ve Sağlık" Bilişim Teknolojileri Dergisi, Cilt: 2, Sayı: 2, Mayıs 2009
9. Sağlık Bakanlığı Hastane Bilgi Sistemleri Kodlama Sistemleri, 2009
10. Ulusal Tıp Bilişimi Kongresi/Medical Informatics'05 Turkey;"Elektronik Reçetenin Avantajları, Kullanımda Karşılaşılabilecek Sorunlar ve Giderilme Yöntemleri"
11. Chen, G. D., Chang, C. K., ve Wang, C. Y. “Ubiquitous Learning Website: Scaffold Learners by Mobile Devices with Information-aware Technologies”. *Computers & Education*, 50(1): 77-90 (2008).
12. Osman Gökalp and Hakan Mollaoğlu, “Uygunsuz İlaç Kullanımı”, Süleyman Demirel University, Faculty of Medicine, Faculty of Medicine Magazine, 2003.
13. Göknur Aktay, Hamit Hancı, Aysun Balseven, "İlaç Etkileşimleri ve Hekim Sorumluluğu", Turkish Medical Association, Paper works, 2003
14. Zhaohui Wang, Rahul Murmura, Angelos Stavrou, "Implementing and Optimizing an Encryption Filesystem on Android", 2013

15. http://source.android.com/tech/encryption/android_crypto_implementation
"Android honeycomb encryption,"
16. D. Cortjens A. Spruyt W.F.C. Wieringa, "WhatsApp Database Encryption Project Report", 31th of December, 2011
17. Eisaku Ohbuchi, Hiroshi Hanaizumi, Lim Ah Hock, "Barcode Readers using the Camera Device in Mobile Phones", 2011
18. R. Muniz, L. Junco, and A. Otero, "A Robust Software Barcode Reader Using the Hough Transform", In Proc. of 1999 International Conference on Information Intelligence and Systems, pages 313–319. IEEE, IEEE Computer Society Press, 1999.
19. CTIA Code Scan Action Team, "Camera-Phone Based Barcode Scanning White Paper", 9 Eylül 2008
20. ZXing resmi kaynak sitesi, <http://code.google.com/p/zxing/>, Google Şirketi tarafından desteklenmektedir, erişim Ağustos 2013
21. Octavian DOSPINESCU, Anca Alexandra DAVID, Faculty of Economics and Business Administration, Alexandru Ioan Cuza University, "From Local Data to Global Information Using ZXing Library in Android", Temmuz 2012
22. Huijuan Yang, Alex C. Kot and Xudong Jiang, "Binarization of Low-Quality Barcode Images Captured by Mobile Phones Using Local Window of Adaptive Location and Size", 2008
23. Prof. Rajkumar A. Soni, Asst. Professor, MCA Department, L. C. Institute of Technology, Bhandu, Gujarat, India. "A Study Paper on Android UI", 1 Ocak 2013
24. Why Develop for Android? [Online] / auth. M Michael // TrueSong Media. - November 12, 2009. - January 21, 2010. -
<http://www.truesongmedia.com/2009/11/why-develop-for-android/>.
25. Xi Qian, Gunagy Zhu, Xiao-Feng Li, Intel Corporation, "Comparison and Analysis of the Three Programming Models in Google Android", 2012
26. Android AutoCompleteTextView Design Resmi destek sitesi,
<http://developer.android.com/reference/android/widget/AutoCompleteTextView.html>, Erişim Temmuz 2013

27. Android Developer Registration resmi web sitesi,
<http://developer.android.com/distribute/googleplay/publish/register.html>, Erişim Ağustos 2013
28. Android Signing Your Applications resmi web sitesi,
<http://developer.android.com/tools/publishing/app-signing.html>, Erişim Ağustos 2013
29. Context on Android security, "Mobile Application Security on Android", 2009
30. Android Proguard resmi kullanım web sitesi,
<http://developer.android.com/tools/help/proguard.html>, Erişim Eylül 2013

ÖZGEÇMİŞ



Ad Soyad: Barış SARIKAYA

Doğum Yeri ve Tarihi: 08.07.1983 / DENİZLİ

Adres: Emniyetevler Mah. Semerkant Sokak No:4 D:10 Kağıthane İSTANBUL

E-Posta: sarikayabaris@gmail.com

Lisans: Yıldız Teknik Üniversitesi - Bilgisayar Mühendisliği

Mesleki Deneyim ve Ödüller:



Sigorta Bilgi Merkezi - Yazılım Geliştirme Takım Lideri

Ocak 2013 – Şuan (1 Yıl) İstanbul, Türkiye



Sigorta Bilgi Merkezi - Senior Java Developer

Mayıs 2011 – Ocak 2013 (1 Yıl 10 Ay) İstanbul, Turkey



TÜBİTAK Bilgem - Uzman Araştırmacı

Mayıs 2009 – Mayıs 2011 (2 Yıl 1 Ay) Kocaeli, Türkiye



TÜBİTAK Marmara Araştırma Merkezi - Araştırmacı

Ekim 2005 – Aralık 2008 (3 Yıl 3 Ay) Kocaeli, Türkiye

TEZDEN TÜRETİLEN YAYINLAR/SUNUMLAR

- Barış Sarıkaya, Mustafa Kamaşak : A Mobile Application for Prescription Drugs. *The International Symposium on Health Informatics and Bioinformatics (HIBIT)*, 25 - 27 Eylül, 2013 Ankara, Türkiye.
- Barış Sarıkaya, Mustafa Kamaşak : Mobil İlaç Prospektüs Uygulaması. *Tıp Teknolojileri Ulusal Kongresi, TIPTEKNO'13*, 31 Ekim - 02 Kasım, 2013 Antalya, Türkiye.