

(Araştırma Makalesi)

Android Cihazlarda Kullanılabilecek Örnek Bir Okul Destek Programının Oluşturulması

Muammer AKÇAY *1, Çiğdem OKTAR²

¹Kütahya Dumlupınar Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 43100, Kütahya,
ORCID No : <https://orcid.org/0000-0003-0244-1275>

²Kütahya Dumlupınar Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, 43100, Kütahya,
ORCID No : <https://orcid.org/0000-0001-5920-3046>

Anahtar Kelimeler:

Android Studio,
Okul Destek Programı,
Konu Anlatımı,
Ders Tekrarı,
Öğrenmenin Pekiştirilmesi

Özet: Bu makalede, Android cihazlarda kullanılabilecek bir okul destek programının nasıl oluşturulduğu açıklanmaktadır. Geliştirilen Okul Destek Programı, 5. Sınıf öğrencilerinin, okulda öğrendiği Matematik, Türkçe ve Fen bilgisi derslerini, Android cihazlarında çeşitli ders anlatım videoları ve ders içeriklerine uygun hazırlanmış oyunlarla tekrar etmesi imkânını sunmaktadır. Bu uygulamada yukarıda belirtilen derslerin farklı konularına ait konu anlatım videoları bulunmaktadır. Bu sayede öğrenci okulda tam olarak anlamadığı konularla ilgili bu videoları izleyebilmekte ve bu konuları daha iyi bir şekilde öğrenebilmektedir. Bunun yanında, öğrencilerin pedagojik yaklaşımla hazırlanmış oyunlarla da konu tekrarlarını istekli bir şekilde yapmaları ve öğrenmenin içselleştirilmesi amaçlanmaktadır. Yazılımı geliştirmek amacıyla, Java programlama dilini temel alan Android Studio programı kullanılmaktadır. Yazılım geliştirme sürecinin aşamaları, projeden ekran görselleri kullanılarak ayrıntılı bir şekilde açıklanmıştır. Uygulama açılış ekranında çalışılmak istenen dersi seçmeye yönelik butonlar bulunmaktadır. Ders seçimi yapıldıktan sonra gelen ekranda seçilen derse yönelik farklı konu anlatımları ve bu konularla ilgili öğrenmeleri pekiştiren oyunlar görülmektedir. Öğrenciler bu ekrandan yapmak istedikleri aktiviteyi seçerek çalışmalarına devam edebilmektedir. Eğer istenirse, ana menüye dönerek farklı dersler için de aynı işlemleri tekrarlayabilmektedirler.

(Research Article)

Creating a Sample School Assistance Program That Can Be Used on Android Devices

Keywords:

Android Studio,
School Assistance Software,
Lectures,
Revision,
Learning Reinforcement

Abstract: This article explains how to create a school support program that can be used on Android devices. The developed School Support Program offers the 5th-grade students the opportunity to revise the Mathematics, Turkish Language and Science lessons learned at school with various lecture videos and games prepared in accordance with the course content on their Android devices. This application contains lecture videos on different topics of the above-mentioned courses. In this way, the students can watch these videos about the topics that they do not fully understand at school and learn these subjects in a better way. In addition, it is aimed that the students will willingly revise the topics of the subjects with the games prepared with a pedagogical approach and internalize the learning. To develop the software, the Android Studio program which is based on the Java programming language is used. The stages of the software development process are explained in detail using screenshots from the project. There are buttons on the application opening screen to select the course to be studied. After the course selection is made, different lectures for the selected course and games that reinforce the learning about these topics are seen on the screen. Students can continue their studies by choosing the activity they want to do from this screen. If desired, they can return to the main menu and repeat the same procedures for different courses.

1. GİRİŞ

Okulda öğrenilen bilgilerin öğrencilerin zihninde tam olarak kalıcı olmasının sağlanması için ders tekrarının belirli aralıklarla yapılması gerekmektedir. Öğretmenler genellikle öğrencilerin öğrendiklerini pekiştirmesi amacıyla onlara ödevler vermektedir. Ancak kitaplardan verilen bu ödevler öğrenciler için çoğu zaman sıkıcı olmaktadır.

Gelişen teknolojiyle birlikte eğitim daha fazla görselleştirilerek daha eğlenceli hale getirilmektedir. Artık internette çeşitli web sayfalarında çeşitli konu anlatımlarına, testlere ve hatta öğretici oyunlara rastlanmaktadır. Yaygınlaşan Android telefonlar da her zaman her yerde eğitime destek için önemli bir kaynak olarak kullanılmaktadır.

Yapılan bu çalışmada eğitime yardımcı olacak, her zaman her yerde çalışıp öğrenebilme imkânı sağlayan bir okul destek uygulama programının nasıl geliştirildiği anlatılmaktadır. Programın hedef kitlesi 5. Sınıf öğrencileridir. Bu program yardımıyla öğrenciler, Türkçe, Matematik ve Fen bilgisi derslerinin konu anlatımlarını seyrederken konu tekrarı yapabilmekte ve çeşitli oyunları oynayarak eğlenceli bir şekilde öğrendiklerini pekiştirebilmektedir.

2. LİTERATÜR VE KAVRAMSAL ÇERÇEVE

2.1. Android İşletim Sistemi

Mobil cihazların donanımsal olarak gelişmesi, onların bir işletim sistemi üzerinden kontrol edilmesi ve kullanılması ihtiyacını ortaya çıkarmıştır. Önceleri sadece iletişim aracı olarak kullandığımız mobil araçlar, artık kullanılan Android işletim sistemi aracılığıyla birden fazla özelliği içermektedir. “İnsansı robot” anlamına gelen Android, Linux çekirdeğine sahip, açık kaynak kodlu bir işletim sistemidir [1].

Temel olarak dokunmatik ekranlar için tasarlanan Android, düşük maliyetli ve kişiselleştirilebilen işletim sistemi arayan yüksek teknoloji cihazlar arasında da popülerdir. Önceleri sadece tablet ve akıllı telefonları kapsayan bu işletim sistemi günümüzde televizyonlar, arabalar, oyun konsolları, dijital kameralar ve saatler gibi cihazlarda da kullanılmaya başlamıştır [2].

İlk olarak 2003'te geliştirilmeye başlanan Android sistemler gitgide gelişmiş, bugünkü halini almıştır. Android'in bu noktalara gelmesindeki en büyük pay kurucularının kurmuş olduğu Android Inc. Şirketinin Google tarafından satın alınması ve kurucularının Google'da çalışmaya başlaması olmuştur [3].

2007 yılında Google, Android işletim sistemini geliştirdiğini açıkladıktan kısa süre sonra dev bir birlik kurulmuştur. Adı Open Handset Alliance (OHA) olan bu

birliğin amacı piyasadaki standartları belirlemek olmuştur [3]. 2008 yılında Google'ın kaynak kodları sunmasıyla telefon üreticilerinin ilk Android telefonu üretmek için birbirleriyle kıyasıya rekabet içerisine girmesi kaçınılmaz olmuştur. Açık kaynak kodlu bir işletim sistemi olan Android, kısa bir süre içerisinde büyük ilerleme kaydetmiş ve yaygınlaşmıştır [3].

Android'in kaynak kodları iki farklı lisans kullanır. Kullandığı Linux çekirdeği GPL, diğer dış bileşenler ise Apache Lisansı ile lisanslanmıştır. Bu değiştirilebilir yapı yazılımcıları Android'e katkıda bulunmaları için cesaretlendirdiğinden sürekli gelişmesini ve yeni özelliklerin eklenmesini sağlamaktadır [2].

2.2. Android İşletim Sisteminin Yetenekleri ve Özellikleri

İlk başlarda cep telefonları için düşünülen Android işletim sisteminin, farklı ürün türlerinde kullanılmasıyla bu işletim sisteminin yetenekleri geliştirilmiştir. Android, günümüzde birçok farklı ürün türlerinde kullanılmaktadır. Farklı alanlardaki bu kullanım desteği, onun hızlı bir şekilde yaygınlaşmasına ve gelişmesine katkı sağlamıştır [1].

Android işletim sistemini kullanan cihazlar, genel olarak şu şekilde gruplandırılabilir [4]:

- Akıllı telefonlar (Smartphone)
- Tablet bilgisayarlar
- Televizyonlar
- GPS (Global Positioning System ~Küresel Konumlama Sistemi) cihazları
- Notebooklar
- Ortam oynatıcıları (DVD, VCD, vb. oynatıcılar)
- Oyun konsolları
- Mutfaklarda kullanılan beyaz eşyalar
- E-Kitap okuyucular

Android işletim sisteminin hızlı bir şekilde yayılması ve farklı cihaz türlerinde kullanılması farklı sürümlerinin ortaya çıkmasına sebep olmuştur. İlk zamanlarda cep telefonlarında farklı, tablet bilgisayarlarda farklı bir sürüme sahip olan Android, 4.0 sürümünün çıkartılmasıyla hem cep telefonlarında hem de tablet bilgisayarlarda kullanılan ortak bir işletim sistemi haline getirilmiştir [1].

Android işletim sisteminin gerek güncellemelerden gerekse yeni platform uyumluluğunu sağlamak adına yeni sürümleri oluşturulmuştur. Oluşturulan bu her sürüme aynı zamanda bir kod adı verilmiştir. Oluşturulan bu sürümlere ait kod adı ve çıkış tarihi Tablo 1'de gösterilmiştir.

2022 yılının ağustos ayında kullanıma sunulan Android 13, Google'ın şimdiye kadarki en farklı olduğu iddia edilen Android sürümlerinden birisi olarak görülmektedir. Bu sürüm aşağıdaki özellikleri sunmaktadır:

- Hem tablet hem de katlanabilen telefonlar için daha büyük ekranlara daha uygun bir arayüz
- Çok kullanıcı, aynı anda çoklu işlemleri daha rahat yürütebilen yeni bir arayüz

Tablo 1. Android sürümleri [5, 14, 15].

| Sürüm | Kod Adı | Tarih |
|-----------------|--------------------|------------------------------------|
| Android 1.0 | - | Eylül 2008 |
| Android 1.1 | Petit Four | Şubat 2009 |
| Android 1.5 | Cupcake | Nisan 2009 |
| Android 1.6 | Donut | Eylül 2009 |
| Android 2.0-2.1 | Éclair | Eylül, 2009-Ocak 2010 |
| Android 2.2.X | Froyo | Mayıs 2010 |
| Android 2.3.X | Gingerbread | Aralık 2010-Şubat 2011 |
| Android 3.X | Honeycomb | Şubat 2011-Temmuz 2011 |
| Android 4.0.X | Ice cream sandwich | Ekim 2011-Aralık 2011 |
| Android 4.1 | Jelly Bean | Temmuz 2012 |
| Android 4.2 | | Kasım 2012 |
| Android 4.3 | | Temmuz 2013 |
| Android 4.4.X | KitKat | Ekim 2013-Temmuz 2014 |
| Android 5.X | Lollipop | Kasım 2014-Mart 2015 |
| Android 6 | Marshmallow | Ekim 2015 |
| Android 7.X | Nougat | Ağustos 2016-Ekim 2016 |
| Android 8.X | Oreo | Ağustos 2017-Aralık 2017 |
| Android 9 | Pie | Ağustos 2018 |
| Android 10 | Quince Tart | Eylül 2019 |
| Android 11 | Red Velvet Cake | Eylül 2020 |
| Android 12 | Snow Cone | Ekim 2021 |
| Android 12L | Snow Cone v2 | Mart 2022 |
| Android 13 | Tiramisu | Ağustos 2022 |
| Android 14 | Upside Down Cake | 2023 sonunda sürülmesi planlanıyor |

- Daha gelişmiş güvenlik, gizlilik ve performans özellikleri
- Uygulama geliştirmede bölünebilir ekranlar üzerinden aynı anda değişik işleri yürütebilme özelliği

2.3. Android İşletim Sistemi Yapısı

Android uygulamaları Android Studio programı kullanılarak java dilinde yazılmaktadır. Mobil cihazların kullanılabilmesi için birçok uygulama geliştirilmektedir. Bu aşamada uygulamaların geliştirileceği işletim sistemlerinin özelliklerinin incelenmesi gerekmektedir. İşletim sistemlerinin pazar payı, uygulama geliştirme dili,

XML editörü, ürünlerin pazarlanması, flash gibi özelliklerinin incelenmesi büyük önem arz etmektedir [6]. Tablo 2’de gösterildiği gibi bu özellikler işletim sistemi bazında karşılaştırılmalı olarak incelenmiştir. Tüm mobil işletim sistemlerinde yapılan çalışmalar, geliştirilen uygulamalar, uygulama geliştirme süreçleri ve bunun gibi birçok özellik karşılaştırıldığında, Android işletim sisteminin ve bu işletim sistemine uygun mobil uygulama geliştirmenin avantajlı olduğu görülmektedir. Android, açık kaynak kodlu olması sebebi ile çoğu yazılımcı tarafından öncelikli olarak seçilmekte ve yazılımcılara esneklik sağlamaktadır. Flash desteği sağlaması ve yazılım geliştirme araçlarının ücretsiz olması tercih edilme sebeplerinden en önemlileridir [7].

2.4. Android Mimarisi ve Sistem Özellikleri

Android Linux tabanlı bir işletim sistemi olduğundan, Linux çekirdeğini kullanmaktadır (Linux Kernel). Çekirdekte, Android için eklenen kodlar ve kütüphaneler bulunmaktadır. Bu kütüphaneler Genel Kamu Lisansı'na sahiptir. Diğer tüm bileşenler, Apache Lisansı ile dağıtılmaktadır. Güç kontrolü, paylaşılan hafıza, low memory killer ve süreçler arası iletişim, Android için özelleştirilmiş yapılar. Güvenlik, hafıza ve işlem (process) kontrolü, dosyalama ve bağlantı I/O işlemleri ve cihaz sürücülerine ise Linux çekirdeğinin doğrudan kaynak sağladığı yapılar [8].

Android işletim sistemi, katmanlardan oluşan bir yapıya sahiptir. Bu yapı Şekil 1’de gösterilmiştir [5]. Bu yapıda bulunan katmanları şu şekilde açıklamaktadır;

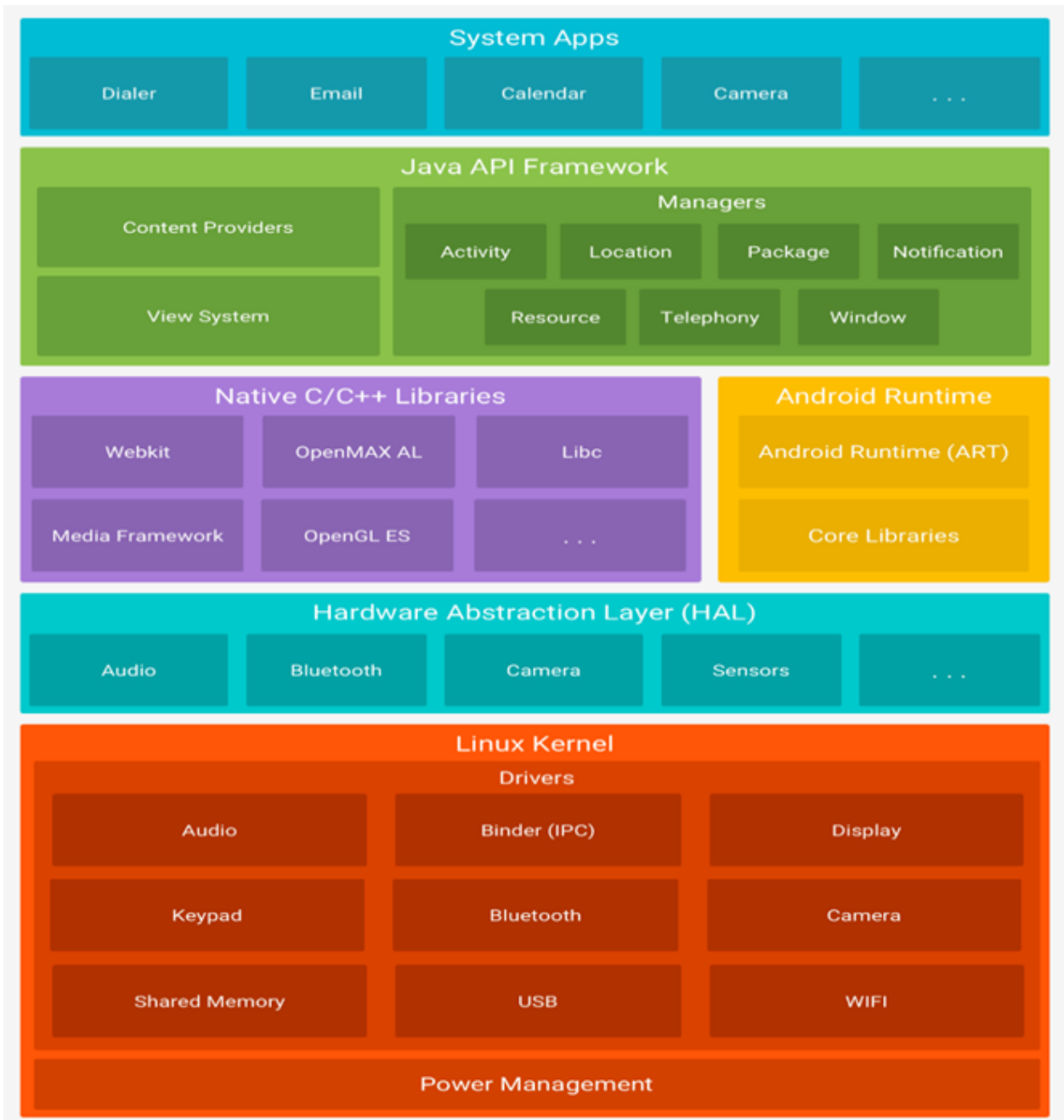
Applications (Uygulamalar): Androidle birlikte gelen e-posta yönetim uygulaması, sms uygulaması, internet tarayıcısı, telefon rehberi gibi bazı çekirdek uygulamaların bulunduğu katmandır [5].

Application Framework (Uygulama Çatısı): Android, geliştiricilerin uygulamalarına zenginlik katmak amacıyla donanım ve yazılım desteğini sağlar [5]. Android uygulamalarının yazılımını oluştururken yazılımcıya uygulama çatısı sağlayan uygulama servisleri aşağıdaki gibidir [2];

- Aktivite Yöneticisi (Activity Manager): Aktivitelerinizin yaşam çemberini kontrol eder. Aktivite yığınının yönetimini içerir.
- Görünümler (Views): Aktiviteler için kullanıcı ara yüzü yapılmasında kullanılır.
- Uyarı Yöneticisi (Notification Manager): Kullanıcılara yapılan bildirimler ve uyarılar için uyumlu ve tutarlı işlev sağlar.
- İçerik Sağlayıcılar (Content Providers): Uygulamanın veri paylaşımını sağlar. Telefon rehberi, resim, müzik vb. verilerin uygulamalarca erişimini sağlayan arabirimlerdir. SQL benzeri erişim ara yüzüne sahiptirler.
- Kaynak Yöneticisi (Resource Manager): Dışarıda tutulmak üzere diziler ve grafikler gibi kodsuz kaynakları destekler.

Tablo 2. Mobil İşletim Sistemleri.

| İşletim Sistemi | Firma | Açık Kaynak Kodlu | Flash Desteği Var Mı? | SDK (Software Development Kit) | Programlama Dili |
|-----------------|-----------|--|-----------------------|--------------------------------|------------------|
| iOS | Apple | Hayır | Hayır | Ücretli | Objective-C |
| Android | Google | Evet | Evet | Ücretsiz | Java |
| Blackberry | RIM | Hayır | Evet | Ücretsiz | C#, Java |
| Windows Phone | Microsoft | Hayır | Evet | Ücretsiz | C# |
| Symbian | Nokia | Hayır | Evet | Ücretsiz | Java, C++ |
| Bada | Samsung | Hayır | Evet | Ücretsiz | Java |
| WebOS | HP | Hayır, (Bazı kısımlar açık kaynak kodlu) | Evet | Ücretsiz | Java |

**Şekil 1.** Android işletim sistemi mimarisi [5].

Kütüphaneler (Libraries): Linux çekirdeğine erişimin sağlanması için bir takım bileşenlere ihtiyaç duyulur. Bu bileşenlerin yer aldığı bölüme libraries denir [5]. Çekirdeğin üstünde yer alan yer katman genelde C++ ve C dilleri ile yazılmış kütüphaneleri içerir (libc, SSL). Bu katmada sistem kütüphaneleri, mp3, mpeg4, jpg gibi çoklu ortam bileşenleri için medya kütüphaneleri ve 2D/3D grafikler için OpenGL/SGL içeren kütüphaneler bulunur [2]. Android İşletim Sistemi kendi ilişkisel verilerini tutabildiği bir SQLite isimli bir veri tabanına sahiptir. Kütüphaneler katmanında veri tabanı için SQLite kütüphaneleri gibi temel kütüphaneler de yer alır [2].

Android Runtime (Android Çalışma Zamanı): Dalvik virtual machine (Dalvik sanal makinesi) ile dosyaların dex formatından dönüştürülerek çalışmasını sağlar [5]. Android'i mobil Linux uygulamasından ayıran en önemli katmandır. Android alt seviye işler için (hafıza yönetimi, donanım sürücülere gibi) Linux çekirdeğini kullanmaktadır ve temel Java kütüphanelerini içerir [2].

Linux Kernel: Android mimarisindeki en alt katmandır. Bu katmanda donanımsal bilgiler ve uygulamaların çalışabilmesi için gerekli sürücüler yer alır (klavye sürücülere, ses sürücülere, Wi-fi sürücülere, kamera,

View, uygulamanın görüntülemeye sorumlu parçasıdır. Verilen örnekte, müziği hoparlöre gönderen kısımdır. Yani kullanıcıya geribildirim döndüren parçadır. Kullanıcı her zaman bu katmanla etkileşim içerisinde. Kullanıcı aksiyonları, uygulama işlemlerini tetiklemektedir [8].

Dışarıdan uygulamaya gelen kullanıcı işlemlerine cevap veren katman ise controller katmanıdır. Gelen aramaya cevap verilmesi, ekran dokunuşlarının işlenmesi, klavyedeki bir tuşa tıkladığında gerekli işlemlerin yapılması hep bu katmanın görevidir. Kullanıcıdan gelen girdiler, controller katmanından geçer ve buradan model katmanına gönderilir. Bütün işlemler yapıldıktan sonra görüntülenmek için view katmanına iletilmektedir [8]. Android 32-bit ARMv7 işlemcilerine uygun olarak geliştirilmektedir. Bunun yanında Android x-86 projesi x86 işlemciler için de destek sağlar. Google tarafından, her 6/9 ay yeni bir sürüm yayımlanmaktadır [8].

2.6. Android Hafıza Yönetimi

Android'te hafıza yönetimi ve süreçlerin (process) öldürülmesi konusu farklı bir yaklaşımla ele alınır: Android, bir sürecin içindeki aktivite kapatıldıktan sonra bile sistem daha fazla hafızaya ihtiyaç duyuncaya kadar o süreci hafızada tutar. Bu durumdaki süreçler, sistemin genel performansı üzerine kötü bir etkiye bulunmazlar; hatta aktivite (activity) tekrardan başlatıldığında ciddi performans artışı sağlamaktadır [9].

Android bunun için bir LRU listesi tutar ve "süreç öldürmeye" bu listedeki en eski süreçten başlayarak gereksiz süreçleri hafızadan kaldırmaktadır [8].

görüntü ve görüntü sürücülere, işlem ve hafıza denetimi, güç denetimi) [2]. Linux çekirdeğinin kullanılması ile güvenlik, bellek yönetimi, süreç yönetimi, ağ yığını ve sürücü modeli gibi temel sistem hizmetleri için haberleşme yapısının gerçekleştirildiği bölümdür [5].

2.5. Android Kullanıcı Arayüzü Mimarisi

Android kullanıcı arayüzü (GUI), tek işlemlerli (single threaded) ve olay tabanlı (event driven) bir özelliğe sahiptir. Yani belirli olaylara karşı aksiyon alan kod bölümleri şeklinde yazılmıştır [8].

Android kullanıcı arayüzü iç içe geçmiş bileşenlerden oluşur (Nested Components). Model-view-controller şablonuna göre tasarlanmıştır ve bu yapıya göre çalışmaktadır [8].

Model, veriyi ve veri konteynerini (data container) temsil eder. Veritabanı tarafı olarak düşünülebilir. Örneğin bir kullanıcı, herhangi bir müzik dosyasını açıp dinleyeceği zaman, oynat tuşuna tıklar ve o tuşa bağlı event'i tetikleyerek çalışır. Uygulama veri tabanından gerekli veriyi girdi olarak alır ve veriyi gerektiği gibi işleyerek kullanıcıya geri gönderir. Bu veri model olarak tanımlanabilir [9].

ActivityManagerService.java sınıfı, süreçleri önem sırası doğrultusunda takip etmekte ve bu değerleri sürecin oom_adj parametresinde sürekli olarak güncel tutmaktadır. Linux'daki standart LMK sadece bir durum altında çalışır, bu da hafızanın kritik şekilde az olmasıdır. LMK modülünü çekirdek tarafından kullanılan standart kullanımı haricinde, başka yeteneklere de sahiptir. LMK birden çok limit tanımlanmasına izin vermektedir [8].

3. MATERYAL VE METOT

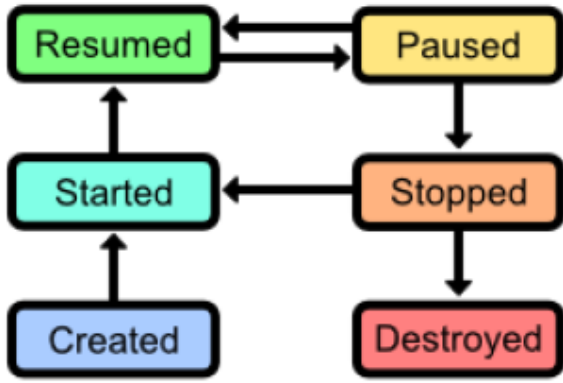
Android uygulama geliştirmek amacıyla önceden farklı yazılım geliştirme araçları kullanılmaktayken, artık sadece Android Studio programı desteklenmektedir. Android uygulaması geliştirmek için, "http://developer.android.com" adresinden ilgili program indirilerek başlanılmıştır.

3.1. Aktiviteler ve Uygulama Yaşam Döngüsü

Bu çalışmada gerçekleştirilen Android uygulamasında, birden fazla aktiviteye (activity) sahip sayfalar kullanılmıştır. Aktivite, uygulamanın kullanıcı ile etkileşimini sağlayan, kullanıcıdan gelen bilgiler doğrultusunda işlem yapan sınıf yapısına sahip arayüzlerdir. Android uygulamalarında yer alan aktivitelerine bir yaşam döngüsü vardır. Bu yaşam döngüsü bir aktivitenin çalışma durumu kontrol eder. Birden fazla aktiviteye sahip olan uygulamalarda ön tanımlı (default) gelecek olan bir aktivite bulunur.

Bir aktivitenin yaşam döngüsü Şekil 2'de görüldüğü gibi oluşturulma (Created), başlatılma (Started), devam edilme (Resumed), duraklatılma (Paused), durdurulma

(Stopped) ve yok edilme (Destroyed) olmak üzere altı durumdan oluşmaktadır. Created ile bir aktivite oluşturulur ve görüntülenmeye hazır hale gelir. Started ile aktivite artık görünür bir durumdadır ancak kullanıcı henüz etkileşime geçememiştir, resumed durumunda ile devam eder. Resumed durumunda artık aktivite ön planda çalışmaktadır ve kullanıcı etkileşime geçebilir. Paused, aktivitenin telefon aranması gibi başka bir program tarafından kesilmesi ile gerçekleşir. Program görünürdür ama kullanıcı etkileşime geçemez. Genelde paused durumunu takip eden Stopped durumunda program arka plana taşınır ve artık görünür değildir, aktivite yeniden oluşturulmadan çalıştırılabilir. Destroyed durumunda ise aktivite sistemden çıkarılır ve artık ulaşılabılır değildir [10].



Şekil 2. Aktivite Yaşam Döngüsü.

Android uygulamaları genel yaşam döngüsüne göre; bir aktiviteye, çalışma esnasında başlatılma, durdurulma, kapatılma gibi durumlar uygulanabilir. Aktivitenin kullanıcı tarafından görüntülenmesi için önce “onCreate()” komutundan geçmesi gerekir. Uygulama sırasında farklı bir aktiviteyi çalıştırmak için “onPause()” komutu kullanılarak aktivite duraklatılır. Aktiviteye, kaldığı yerden devam edilmek istendiği takdirde “onResume()” komutu kullanılarak ekrana getirilir. Bir aktivitenin durdurulması ise “onStop()” komutu ile gerçekleştirilir. Bu komuttan sonra aktivite “onCreate()” veya “onResume()” komutları kullanılarak tekrar görüntülenebilir. Sonlandırılmış aktivitelerin (“onDestroy()” komutunun çalıştırılması) yeniden görüntülenmesi için “onCreate()” komutu kullanılır [1].

Aktivitelerin durum bilgileri, çalışma sırasında yığın (stack) bölgesinde tutulur. “Son giren ilk çıkar” mantığına göre çalışan yığımda en son eklenen veri, ilk önce çekilecek olan veridir. Bir uygulamada en son başlayan aktivite bu yığın bölgesinin en üstüne yerleştirilir. Ondan önce başlamış aktiviteler ise sırasıyla bir alt bölgesinde tutulur. Bir aktivitenin kendisinden önceki aktiviteye dönmesi gerektiğinde, kendisini yok eder (“onDestroy()” komutu) ve yığın bölgesine geri dönerek önceki aktiviteyi devam ettirir. Bazen yeni bir aktivitenin çalıştırılması istendiğinde yığın bölgesinde uygun yer yoksa bu bölgede bulunan ve yok edilmesi durumunda uygulamayı etkilemeyecek aktiviteler “onDestroy()” işlemine tabi tutulur. Bu işleme tabi tutulmadan önce

“onSaveInstanceState()” komutu ile aktivitenin o an için son durumu kaydedilir. Uygulama esnasında tekrar bu aktiviteye dönmek gerekirse “onCreate()” metoduyla aktivite yeniden oluşturularak “onSaveInstanceState()” komutuyla aktiviteye ait durum bilgileri tekrar yüklenir [1].

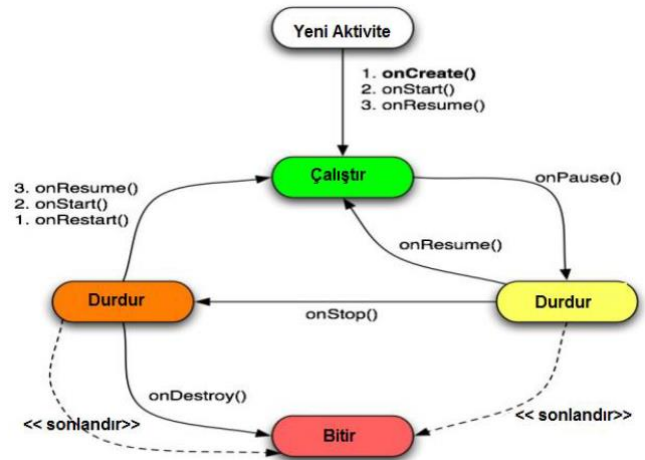
Bir Android proje yapısı aktiviteler (activities), düzen (layout), values ve drawable gibi basit parçalardan oluşmaktadır. Düzen, uygulamaların kullanıcı arayüzünü oluşturan elemanları barındırır ve bir düzen içerisinde görüntülenmesini sağlayan en alt katmandır. Layouts klasörünün içerisinde uygulamanın nasıl görüneceğini belirleyen xml dosyaları bulunur [11].

Düzen oluşturmak için kullanılan yöntemler:

- *Kullanıcı Arayüzü Elemanlarının XML Olarak Tanımlanması:* Android çok geniş bir XML yapısı sunmaktadır. Her biri değişik xml etiketi ve özellikleri ile tanımlanan çok çeşitli arayüz elemanı ve bileşeni, Android uygulamalarının görsel olarak geliştirilebilmesine ve kullanıcıya yönelik özelliklerinin artırılabilmesine olanak sağlamaktadır [12].

- *Kullanıcı Arayüzü Elemanlarının Runtime'da Oluşturulması:* Uygulamalarda View ve ViewGroup nesnelere kolaylıkla, kod tarafından da yaratılıp, yönetilmektedir [5].

Aktivite ise kullanıcı ile etkileşimi sağlayan ekrandır. Her bir düzen (layout) yani ekran için bir aktivite olması gerekir ve düzen yapıları, aktivitelerden kontrol edilmektedir. Aktivite sınıfı tarafından oluşturulan nesnelere uygulama içinde kullanılan ekranları temsil ederler. Kullanıcı yeni bir ekrana geçtiğinde Context.startActivity metoduyla bu ekran başlatılır. Eğer yeni aktivite, AndroidManifest dosyasında tanımlı değilse uygulama çökebilir. Şekil 3'te Aktivite olayları akışı gösterilmektedir [13].



Şekil 3. Aktivite Olayları.

4. BULGULAR

4.1. Uygulama Açılış Ekranı

Açılış ekranı, Android uygulamasının açılması ile birlikte gelen ilk karşılama ekranıdır. Android uygulamasında bir aktivite oluşturulurken kullanıcı ve kontrol işlemlerinin yapıldığı “java”, kontrollerin konumlandırılması ve özelliklerin belirleneceği “.xml” uzantılı arayüz dosyaları oluşturulur. Bu dosyaların etkinleştirilebilmesi için, “AndroidManifest.xml” dosyası içerisinde tanımlanması gerekmektedir.

Yapılan çalışmada açılış ekranı Şekil 4’te gösterilmektedir.



Şekil 4. Açılış Ekranı.

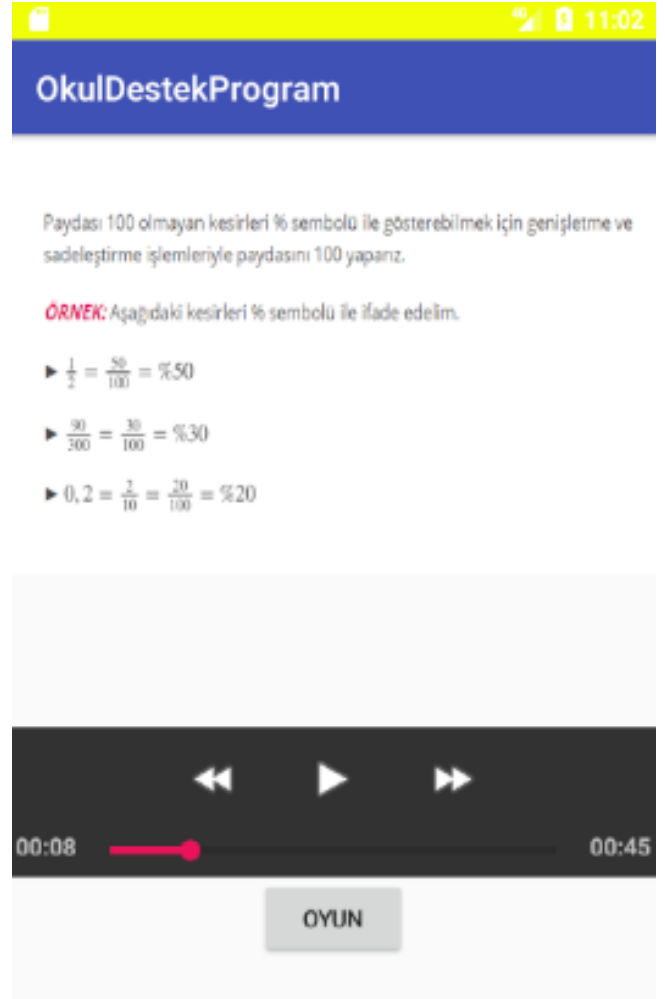
4.2. Uygulamanın Konu Anlatım Ekranı

Uygulamada farklı derslerin farklı konularına ait konu anlatım videoları bulunmaktadır. Bu sayede öğrenci anlamadığı konuları tekrar tekrar izleyerek gözden geçirebilmektedir. Eklenen videonun kontrol edilebilmesi için MediaController sınıfından ilgili metod çağırılması gerekmektedir. İlgili sayfada yazılan kodlar tez ile verilen CD’de bulunmaktadır. Projenin ekran görüntüsü ise Şekil 5’te gösterilmektedir.

4.3 Uygulamanın Oyun Ekranı

Uygulamada konu anlatımından sonra öğrencinin öğrendiklerini tekrar edebilmesi için çeşitli türde oyunlar

bulunmaktadır. Oyunların seçildiği ekran Şekil 6’da gösterilmektedir.

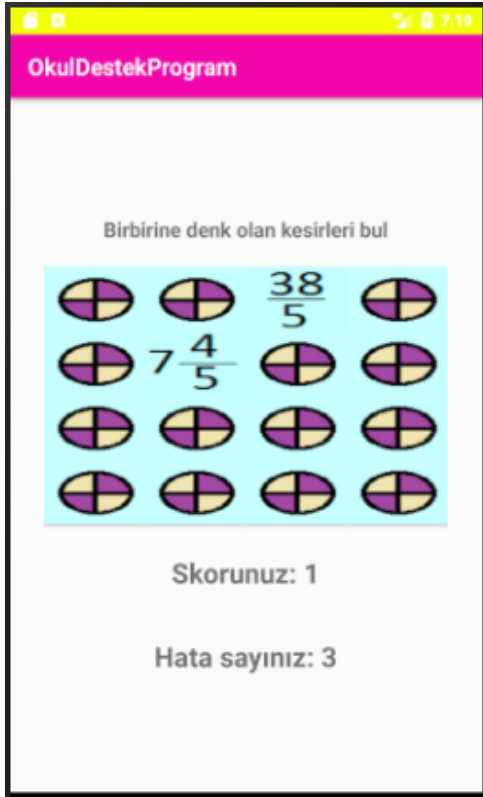


Şekil 5. Konu anlatımı ekran görüntüsü.



Şekil 6. Oyun Seçme Ekranı.

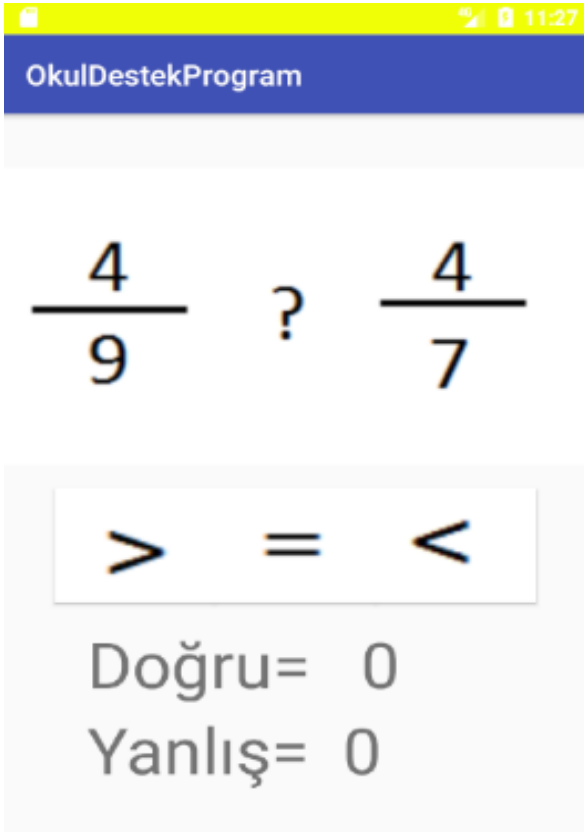
Oynanmak istenen oyunun butonu tıklandıktan sonra oyun oynanabilmektedir. Çeşitli oyun görüntüleri Şekil 7, Şekil 8, Şekil 9, Şekil 10, Şekil 11, Şekil 12 ve Şekil 13'te gösterilmektedir.



Şekil 7. Hafıza Oyunu Örneği.



Şekil 9. Toplama İşlemi Oyunu.



Şekil 8. Büyük Küçük Eşit Oyunu.



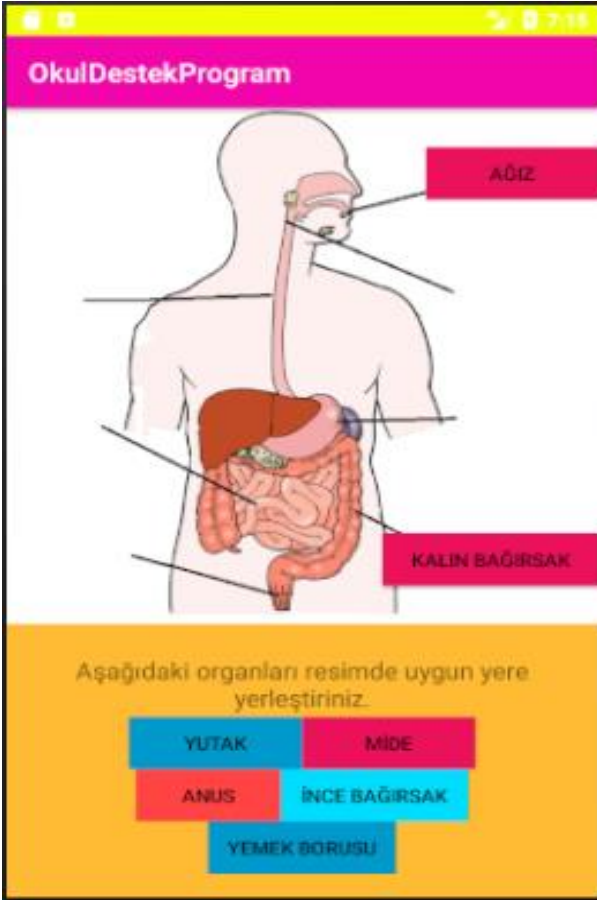
Şekil 10. Sıfat Çeşitleri Oyunu.



Şekil 11. Besinleri Uygun Kutuya Yerleştirme Oyunu.



Şekil 13. Sıfatları Uygun Kutuya Yerleştirme Oyunu.



Şekil 12. Sindirim Sistemi Organlarını Uygun Kutuya Yerleştirme Oyunu.

5. TARTIŞMA VE SONUÇ

Günümüzde cep telefonu kullanıcı sayılarının gittikçe artmasıyla birlikte mobil iletişime verilen önem hızla artmakta ve yeni teknolojilerin kullanımı tüm dünyada yaygınlaşmaktadır. Cep telefonları, küçük boyutlu, veri iletişim hızları yüksek, farklı teknolojilerle bütünleşebilir hale gelmiş ve günümüzün vazgeçilmez iletişim aracı olmuştur.

Uygulama açılış ekranında çalışılmak istenen dersleri seçmeye yönelik butonlar bulunmaktadır. Ders seçimi yapıldıktan sonra gelen ekranda seçilen derse yönelik farklı konu anlatımları ve bu konularla ilgili bilgileri pekiştiren oyunlar görülmektedir. Kullanıcı buradan yapmak istediği aktiviteyi seçerek çalışmasına devam edebilmektedir. Eğer isterse ana menüye dönerek farklı dersler için de aynı işlemleri tekrarlayabilmektedir. Bu durum kullanıcının farklı dersleri kolaylıkla tekrar edip, öğrendiklerini oyunlarla pekiştirmesini de sağlamaktadır.

Cep telefonlarının eğitim-öğretim alanında kullanımı kapsamında bu çalışmada, öğrencilerin derslerini eğlenceli bir şekilde tekrar etmelerinin sağlanması planlanmıştır. Oluşturulan yapıya istenirse daha fazla ders ve oyun eklenebilmektedir.

Etik Hususlar

Etik kurallara uyum

Yazarlar etik görev ve sorumluluklara riayet edildiğini beyan ederler.

Finansman

Yazarlar kamu, ticari veya kar amacı gütmeyen sektörlerdeki fon kuruluşlarından özel bir hibe alınmadığını beyan ederler.

Çıkar çatışması

Yazarlar herhangi bir kişi veya kurumla çıkar çatışmasının bulunmadığını beyan ederler.

Teşekkür

Projenin hazırlanması esnasında bizlere sabır gösteren değerli eşlerimiz ve çocuklarımıza teşekkür ederiz.

KAYNAKÇA

- [1] Muştı, Y. 2012. Sms ve Android tabanlı uygulamalarla sınıf içi etkileşimin artırılması. Bilecik Üniversitesi, Yüksek Lisans Tezi, Bilecik.
- [2] Kh'tour, M. 2015. Android tabanlı hastane uygulaması. Kırıkkale Üniversitesi, Yüksek Lisans Tezi, Kırıkkale.
- [3] Dalmaz, H. 2015. Android tabanlı cihazlarda web verilerinin gerçek zamanlı değişken parametrelili takibi. Erciyes Üniversitesi, Yüksek Lisans Tezi, Kayseri.
- [4] Taç, M. 2011. Android Programlama. Dikeyksen, İstanbul.
- [5] Android. <https://developer.android.com/about/versions> (Erişim Tarihi: 24.01.2023)
- [6] Karaman, S., Yıldırım, S. and Kaban, A., 2008. Öğrenme 2.0 yaygınlaşıyor: Web 2.0 uygulamalarının eğitimde kullanımına ilişkin araştırmalar ve sonuçları. XIII. Türkiye'de İnternet Konferansı Bildirileri, 22-23 Aralık, Orta Doğu Teknik Üniversitesi, Ankara, 35-40.
- [7] Keskin, N. Ö., & Kılınç, A. G. H. 2015. Mobil öğrenme uygulamalarına yönelik geliştirme platformlarının karşılaştırılması ve örnek uygulamalar. Açık öğretim Uygulamaları ve Araştırmaları Dergisi, 1(3), 68-90.
- [8] İğci, E. 2014. Android işletim sistemi ile çalışan mobil cihazlarda uygulama geliştirilmesi. Ege Üniversitesi, Yüksek Lisans Tezi, İzmir.
- [9] Aslan, B. 2013. Web 2.0 Teknikleri ve Uygulamaları artırılması. Kırklareli Üniversitesi, Yüksek Lisans Tezi, Kırklareli.
- [10] Hodson, R. 2014. Android Programming Succinctly. Syncfusion Inc.
- [11] Aliferi, C. 2016. Android Programming Cookbook. Exelisis Media P.C.
- [12] Murata, D., Kohn, C., 2009, Internet Drafts: XML Media Types, Internet Engineering, Task Force.
- [13] Ahmad, N., 2013, Requirements analysis of android application using activity theory: A case study,"

Information and Communication Technology (ICoICT), 2013 International Conference of Information and Communication Technology (ICoICT), 20-22 March, Bandung, Indonesia, 145-149.

- [14] Wikipedia, <https://en.wikipedia.org/> (Erişim Tarihi: 24.01.2023)
- [15] [Raphael, J.R. 2022. Android versions: A living history from 1.0 to 13. <https://www.computerworld.com/article/3235946/android-versions-a-living-history-from-1-0-to-today.html>? (Erişim Tarihi: 24.01.2023)