BİR ARTTIRILMIŞ GERÇEKLİK UYGULAMASININ GELİŞTİRİLMESİ VE ÖĞRENCİ GÖRÜŞLERİ

Cüneyt ÖZEL[*]

Çelebi ULUYOL[**]

ÖZ

Teknoloji baş döndürücü bir şekilde çok hızlı biçimde gelişmektedir. Akıllı telefon ve tablet bilgisayarlardaki donanımsal gelişmeler beraberinde bu cihazlar üzerinde kullanılan yazılımların sayısında da artışa neden olmaktadır. Gerçek görüntü ve bilgisayar tarafından üretilmiş olan nesnelerin aynı anda bir bağlantı aracılığıyla ekranda görüntülenmesi arttırılmış gerçeklik olarak adlandırılmaktadır. Elektronik ticaret, mimari ve tasarım, eğlence, satış ve pazarlama, giyim endüstrisi ve eğitim gibi birçok alanda arttırılmış gerçeklik uygulamalarına rastlamak mümkündür. Alanyazın incelendiğinde arttırılmış gerçeklik ile ilgili daha çok teorik bilgilere rastlanmaktadır. Bununla birlikte uygulama geliştirmeye yönelik çalışmaların bulunduğu da bir gerçektir. Bu araştırma içerisinde de alanyazında eksikliği bulunan arttırılmış gerçeklik uygulamalarının nasıl geliştirilebileceğine yönelik açıklamalar yer almaktadır. Geliştirilen uygulama, Gazi Üniversitesi merkez kampüs içindeki binaların konumlarını alıp isimlerini ortaya çıkaran mobil arttırılmış gerçeklik yazılımıdır. Kodlama aşamasında Wikitude yazılım geliştirme aracından faydalanılmıştır. Bu aşamada Android işletim sistemi için uygulama geliştirme aracı olan Android Studio platformu kullanılmıştır. Kullanıcı ekranında gerçekleştirilen işlemler için Html, javascript ve CSS teknolojilerinden yararlanılmıştır. Araştırmacılar, bu araştırmada ortaya konan adımların, özellikle arttırılmış gerçeklik uygulamaları geliştirmek isteyen kişiler açısından yararlı olacağını düşünmektedirler.

Anahtar Kelimeler: Arttırılmış gerçeklik, uygulama, Android, Wikitude, Gazi

[*] Yüksek Lisans Öğrencisi, cuneytozel@msn.com

[**] GÜ GEF Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü, celebi@gazi.edu.tr

Geliş Tarihi: 29.06.2016 – Kabul Tarihi: 02.11.2016

DEVELOPMENT OF AN AUGMENTED REALITY APPLICATION AND STUDENTS' OPINIONS

ABSTRACT

Technology is evolving very rapidly. Hardware development in the smartphones and tablet computers causes an increase in the number of software use on these devices. One of the important concepts emphasized in recent years is augmented reality. In augmented reality applications, reality and computer-generated objects are presented together on a same screen via an Internet or local connection. Nowadays, augmented reality is used in different areas such as medical applications, e-commerce, advertising and marketing, entertainment, architecture and interior design, modeling and education. Most research in the literature emphasize the advantages or disadvantages of augmented reality theoretically. However, it is also true that there are some efforts to develop applications in various areas. In this study, an augmented reality application was developed by researchers and development stages were explained step by step. The application takes the location and radar data of some faculty buildings in Gazi University campus and shows the name of the buildings on the mobile device screen. For the research, Wikitude softare development kit was used for coding. The augmented reality application was designed in Android Studio platform for Android operating system. User screens were designed using Html, javascript and CSS. Researchers believe that emphasized steps in this study will be useful for researchers who want to develop augmented reality applications in the future.

Key Words: Augmented reality, application, android, Wikitude, Gazi

GİRİŞ

Bilişim teknolojilerindeki hızlı değişim ve mobil teknolojilerin kullanım oranındaki artış ile birlikte görsel teknoloji içerikleri de önem kazanmaktadır. Mobil cihazların giderek yaygınlaşması ve farklı yaş grubundaki kişiler tarafından arzu edilen mobil uygulama talepleri, mobil cihazlarda çalışan uygulamaların nitelik ve nicelik olarak da artmasına neden olmuştur. Bu sayede mobil cihazlarda çalışan uygulama kütüphaneleri çok daha zenginleşmiştir. Arttırılmış gerçeklik (Augmented reality) olarak adlandırılan teknoloji de aslında mühendislik alanında uzun bir geçmişe sahip olsa da mobil uygulama anlamında son yıllarda popüler hale gelmekte ve giderek hayatın farklı alanlarında karşımıza çıkmaktadır (Küçük, Kapakin ve Göktaş, 2015).

Bu araştırmanın amacı alanyazında birçok çalışmaya konu olmuş ve geleceğin teknolojilerinden birisi olarak gösterilen arttırılmış gerçeklik (AG) ile ilgili örnek bir uygulama geliştirmektir. Arttırılmış gerçeklik uygulamaları tıp, eğlence, mimari, tasarım ve eğitim gibi birçok disiplinde kullanılmaktadır. Bu uygulamaları gerçekleştirme imkanı sağlayan hem ticari hem de ücretsiz

açık kaynak kodlu yazılımlar mevcuttur. Özellikle son dönemde arttırılmış gerçeklik üzerine yapılan yüksek lisans ve doktora tezlerinde bir artış olduğu söylenebilir. Ancak, konu ile ilgili uygulama geliştirme anlamında çeşitli nedenlerden ötürü önemli eksiklikler bulunmaktadır. Bu makalenin özellikle yeterli teknik beceriye sahip olmayan, fakat uygulama geliştirmek isteyen araştırmacılar için yararlı olacağı düşünülmektedir. Makaledeki uygulama geliştirme adımları, kullanılan teknolojiler ve kodlar giriş düzeyinde de olsa araştırmacılara yön gösterecektir. Aşağıda ilk olarak arttırılmış gerçeklik ile ilgili temel tanımlamalar verilmiş, ardından araştırmacılar tarafından geliştirilen ortam detaylı olarak açıklanarak ortamla ilgili öğrenci görüşleri sunulmuştur.

ARTTIRILMIŞ GERÇEKLİK

Dijital nesneler olarak adlandırılabilecek metin, ses, resim, grafik, şekil ve animasyonların gerçek dünya görüntüsü üzerine yerleştirilerek hem gerçeğin hem de bilgisayar tarafından üretilen bu nesnelerin aynı anda ekranda görüntülenmesi arttırılmış gerçeklik olarak adlandırılır (Zhou, Duh ve Billinghurst, 2008). Arttırılmış gerçeklik ilk olarak üzerinde özel lenslerin bulunduğu ve HMD (Head Mounted Display) olarak adlandırılan gözlükler aracılığıyla çalıştırılmıştır. Kişi gerçek dünyaya bu gözlük aracılığıyla bakmakta ve dijital nesneler kişinin döndüğü yöne göre uygulama olarak bir ekranda veya gözlük üzerinde çalışabilmektedir (Ludwig ve Reimann, 2005). Bu gözlüklerin en büyük dezavantajlarından birisi tüm uygulamaların bir laboratuvar ortamında çalıştırılmasıdır.

Arttırılmış gerçeklik ve sanal gerçeklik akademik araştırmalarda birbirinin yerine kullanılsa da çeşitli araştırmacılar bu kavramları farklı tanımlamaktadırlar (Hsieh ve Lin, 2011). Hsieh ve Lin (2011) sanal gerçekliği, gerçeğin tamamen sanal bir ortama taşınması, arttırılmış gerçekliği ise sanal ve gerçeğin aynı anda ekranda sunulması olarak tanımlamaktadırlar. Bu tanıma göre sanal gerçeklik gerçek ortamdaki tüm değişken ve bileşenlerin sanal bir ortama taşınmasıdır. Böyle bir durumda artık gerçek yerini tamamen sanal dünyaya bırakmaktadır.

Özel tasarlanmış ortamlarda yürütülen ilk AG uygulamaları çeşitli eksikliklerine rağmen, endüstriyel düzenekler, tıp eğitimi ve oyunlar gibi geniş bir yelpazede başarılı olmuştur. Bu sistemlerin maliyetinin pahalı olması ve teknik olarak uzmanlık gerektirmesi bu teknolojilerin yaygınlaşmasını engellemiştir (Billinghurst, Kato ve Poupyrey, 2001). Özellikle 2000'li yıllarla birlikte akıllı telefonlar ve tablet bilgisayarlardaki gelişim Wikitude, Layar, Junaio gibi arttırılmış gerçeklik tarayıcılarının geliştirilmesine imkan tanımıştır. Bu gelişmelerle birlikte arttırılmış gerçeklik de laboratuvar ortamından çıkarak akıllı cihazlarda çalışan ve ulaşılması kolay uygulamalara dönüşmüştür. Günümüzde bu teknoloji hem kullanıcılar hem de uygulama geliştiriciler açısından daha ulaşılabilir hale gelmiştir.

Arttırılmış Gerçeklik Teknolojilerinin Kullanım Alanları

Ludwig ve Reimann (2005)'ın önerdiğine göre alanyazında yapılan tüm arttırılmış gerçeklik uygulamaları üç kategoriye ayrılmıştır: (1) Sunum ve görselleştirme, (2) Sanayi ve (3) Eğitici eğlence. Ek olarak, Hamilton (2011) eğlence sanayi, oyun endüstrisi, gezi ve turizm endüstrisi, pazarlama alanı, çevrimiçi sosyal ağlar ve günlük yaşamda kullanılabilirliğinin yanında, arttırılmış gerçekliğin eğitimde de geniş bir yelpazede kullanılabileceğini önermiştir. Şu reddedilemeyen bir gerçektir ki arttırılmış gerçeklik uygulamaları hızlı bilgi transferinin kritik ve hassas olduğu bütün alanlar için muazzam bir potansiyele sahiptir. Bu potansiyel özellikle eğitim için oldukça doğru ve önemlidir. Ancak dünyadaki uygulamalara bakıldığında arttırılmış gerçeklik araştırma ve gelişmelerinin eğitim yerine daha çok iş dünyası ve çevresinin ilgisini çektiği görülmektedir. Dolayısıyla bu tarz uygulamalarda herhangi bir gerçek eğitim gündemi söz konusu değildir. Bu nedenle arttırılmış gerçeklik uygulamalarının büyük bir kısmının akademik olmayan çevrelerde kullanımına rastlanmaktadır. Alanyazın incelendiğinde arttırılmış gerçekliğin reklam ve pazarlama, mimari ve inşaat, eğlence, tıp, askeri alanlar, gezi vb. farklı alanlarda kullanımının olduğu görülmektedir.

Arttırılmış gerçeklik teknolojilerinin kullanıldığı alanlar aşağıda kısaca tanıtılmaktadır (Billinghurst, Kato ve Poupyrey, 2001). Bu alanlara ilişkin bilgiler Billinghurst, Kato ve Poupyrev'in (2001) yaptığı çalışmada daha detaylı açıklanmaktadır.

Reklamcılık: Arttırılmış gerçeklik (AG) uygulamalarının en çok kullanıldığı alanlardan birisi reklamcılıktır. Çeşitli videolar ve animasyonlarla desteklenen AG kampanyaları kullanıcıların ilgisini çekmektedir. Günümüzde otomotiv, kozmetik, basın yayın, giyim, gıda, müzik, sinema vb. birçok sektöre yönelik AG reklamları geliştirilmektedir.

Tıp ve medikal: Tıp uygulamalarının çoğunluğu görüntü ve robot destekli cerrahi uygulamalardır. Günümüzde çeşitli tıp fakültelerinde eğitim ve tıbbi operasyon amaçlı AG uygulamaları kullanılmaktadır. Bu uygulamaların tıp alanındaki avantajlarından en önemlisi gelişmiş cihazlar yardımıyla verilerin anlık olarak doktor ve öğrencilere sunulmasıdır. Görüntüleme ve görüntü işleme sistemleri tıp alanında önemli bir konudur. Bu alanda görüntüleme sistemleri ve arttırılmış gerçeklik sistemleri çalıştayları düzenlenmektedir (örneğin MIAR-Medical Imaging and Augmented Reality). Google firmasının geliştirmiş olduğu Google Glass ürününden ameliyatlarda yararlanılması, tıp doktorları ve genç cerrahlar için geliştirilen rehber niteliğindeki bir ürün olan Medicar uygulaması, hemşireler için üretilen Evena Glasses akıllı gözlükleri/uygulamaları, endoskop işlemleri için üretilen başlık cihaz ve uygulamaları ve Vuzix akıllı gözlükleri için üretilen tıbbi uygulamalar AG'nin tıp alanında kullanımına örnek olarak verilebilir.

Eğitim: Teknolojideki gelişmeler eğitim ortamlarında AG kullanımına yönelik uygulamaları da beraberinde getirmektedir. Eğitim içerikleri ve AG teknolojilerinin birleşimi ile öğrencilerin gerçek hayat senaryoları içinde, yeni uygulamalar ve etkin, çekici öğrenme ve öğretim teknikleri geliştirilebilir. AG uygulamaları her alanda olduğu gibi eğitim alanında da öğrenciler için zenginleştirilmiş bir görsel olarak desteklenen içerikler oluşturularak eğitim etkinliği arttırılabilir (Rieber, 1996). Arttırılmış gerçeklik ile öğrenci ve öğretmenlerin eğitim etkinliklerinde gerçek dünyada çeşitli imkânsızlıklar sebebiyle ulaşamadıkları ya da somutlaştıramadıkları birçok nesneyi, uygulamayı, deneyi çok farklı boyutlarda daha uygun maliyetlerde ve gerçek ortamından soyutlamadan sunabilmekte ve sağladığı anlık etkileşimlerle eğitim süreci daha eğlenceli kılınabilmektedir. Öğrenen, arttırılmış gerçeklik uygulamalarıyla duyusal olarak görme ve işitmenin yanında etkileşimli uygulamalar sayesinde zenginleştirilmiş ortam ile etkileşime geçebilme olanağına sahip olmaktadır (Kerawalla, Luckin, Seljeflot ve Woolard, 2006). Anatomy 4D uygulaması, insan vücudunun farklı şekillerde üç boyutlu olarak görüntülenmesini sağlayan etkileşimli bir mobil AG uygulamasıdır. Bunun dışında Aurasma arttırılmış gerçeklik tarayıcısının alt yapısını kullanan Science AR uygulaması, bazı bilimsel konularda çocuklara yönelik olarak hazırlanmış bir eğitim içerikli uygulamadır. Ayrıca yıldız sistemlerinin konumsal olarak kameradan görüntülenmesini sağlayan Skyview ve Skymap gibi uygulamalar mevcuttur. Chemistry101, Anatomy101 ve Geometry101 gibi eğitim amaçlı uygulamalar da farklı AG örnekleridir.

Alanyazın incelendiğinde eğitimle doğrudan ilgili olan veya olmayan birçok farklı alan ve disiplindeki çeşitli uygulamaların araştırmacılar tarafından incelendiği görülmektedir. Örneğin, Heydarzadeh, Nourani ve Park (2015) genel anestezideki arttırılmış gerçeklik tıp uygulamalarını incelemiştir. Benzer biçimde, Sielhorst ve arkadaşları (2004) tıp eğitim simülasyonlarında arttırılmış gerçeklik kullanımına bakmışlardır. Aynı zamanda, Sielhorst, Feuerstein ve Navab (2008) 1990'lı yıllardaki arttırılmış gerçeklik tıbbi görüntüleme cihazlarını incelemişlerdir. Liarokapis ve arkadaşları (2004) makine mühendisliği için arttırılmış gerçekliğin bir araç olarak kullanımını incelemişlerdir. Kaufmann ve arkadaşları öğretmenler ve öğrenciler arasındaki öğrenmeyi kolaylaştırmak için matematik ve geometri dersinde kullanılan bir arttırılmış gerçeklik sistemi geliştirmişlerdir (Kaufmann, 2003; Kaufmann ve Dünser, 2007; Kaufmann ve Schmalstieg, 2003). Bütün bu farklı alanlardaki uygulamaların dışında eğitimsel arttırılmış gerçeklik uygulamaları; arttırılmış gerçeklik kullanılan kitaplar, arttırılmış gerçeklik oyunları, keşfederek öğrenme (discovery learning), nesne modelleme (objects modeling) ve beceri eğitimi (skills training) olarak da karşımıza çıkmaktadır.

Eğlence ve oyun: Mobil uygulama marketlerinde farklı platformlarda çalışan cihazlar için geliştirilmiş, farklı ilgi alanlarına yönelik pek çok arttırılmış gerçeklik oyun uygulaması bulmak mümkündür. Farklı bir örnek

olarak mobil cihaz uygulamaları ile uzaktan kumanda edilebilen bir Wi-Fi helikopter cihazı olan AR.Drone ürünü verilebilir. Bu oyuncak helikopter, ios ya da Android mobil cihaza yüklenilen kontrol uygulaması sayesinde video kaydının yanı sıra arttırılmış gerçeklik oyun deneyimini de kullanıcılara sunmaktadır. Son günlerde büyük yankı uyandıran Pokemon Go uygulaması da AG'ye verilebilecek en güzel örneklerden birisidir.

Navigasyon ve coğrafi bilgi sistemleri: Arttırılmış gerçeklik uygulamalarının ilgi çekici ve büyük potansiyele sahip kullanım alanlarından birisi de navigasyon konusudur. Yelp Monocle, Wikitude World Browser, Junaio Around You, Layar City Guides uygulamaları gibi AR uygulamaları, kullanıcılara bir şehirde bulunan restoranlar, müzeler, parklar vb. yerleri kamera vasıtası ile görüntüleme ve ilgili noktaya yönlendirme hizmeti vermektedirler. Bu tarayıcılar verilerini sosyal ağlardan, Wikipedia veya Google gibi içerik sağlayıcılardan temin etmektedirler (Craig, 2013).

Yukarıda sayılan kullanım alanlarına ek olarak arttırılmış gerçeklik teknolojisi tercüme hizmetleri, elektronik ticaret, tasarım, modelleme ve üretim gibi daha birçok alanda uygulamalar barındırmaktadır. Arttırılmış gerçeklik uygulamaları geliştirirken üç farklı uygulama çalıştırma işleminden söz edilebilir (Wojciechowski ve Cellary, 2013). Bunlar: (1) İşaretleyici tabanlı, (2) İşaretleyicisiz ve (3) Konum tabanlı AG uygulamalarıdır.

İşaretleyici tabanlı arttırılmış gerçeklik uygulamalarında ortama eklenecek sanal objelerin düzgün bir biçimde konumlandırılabilmesi için bazı işaretleyiciler kullanılmaktadır. Bir tane işaretleyici yazılım tarafından tanındığı taktirde bir eylem gerçekleştirilir. Genelde bu eylem 3 boyutlu nesnenin işaretleyicinin üzerinde belirmesi şeklinde olmaktadır. Genel olarak AG sistemi cihaz ya da işaretçinin yeri değiştirilse bile 3 boyutlu model işaretçi ile hareket ederek çeşitli açılardan da görüntü alınması sağlanabilir. Yani bu nesnenin etrafında hareket edilebileceği model için 360 derecelik bir görüş anlamına gelmektedir. Benzer şekilde işaretleyicinin kameraya olan uzaklığı da dikkate alınarak 3 boyutlu modelin boyutu buna göre ayarlanabilir.

İşaretleyicisiz AG sistemi ile canlı video akışı içerisinde tanınabilir nesneler ile AG ortamları oluşturmak mümkün olabilir. Örneğin işaretsiz bir AG uygulaması basılı bir işaretçiye ihtiyaç duymadan, insan yüzlerini ve diğer objeleri tanımak için geliştirilmiş olabilir. Ayrıca işaretleyici tabanlı arttırılmış gerçeklik uygulamaları her bir nesne gösterimi için basılı bir işaretçiye ihtiyaç duyduğundan açık alanlarda gerçekleştirilen AG ortamları için uygun olmayabilir. Böyle durumlar için işaretleyicisiz AG yaklaşımları uygulanabilir. Bu yaklaşım gelişmiş görüntü işleme ve tanımlama yöntemleri kullanmaktadır. Bunlara bir araç firmasının bir modelinin tanımlanarak o araca ait bir videonun ekranda gösterimi ya da bir hayvan resminin tanımlanarak resim üzerine hayvanın 3 boyutlu modelinin bindirilmesi gibi örnekler verilebilir.

Konum bazlı AG uygulamalarının ise en yaygın kullanımı AG tarayıcıları ile olmaktadır. Bir akıllı telefonda mevcut olan jiroskop, ivmeölçer, pusula gibi donanımlar vasıtası ile algılanabilen kamera görüntüsü üzerinde katmanlı sanal çevre bilgilerinin görüntülenmesi sağlanmaktadır. Konum bazlı AG uygulamalarının performansı mobil cihazlarda bulunan sensörlerin yetenekleri ile sınırlıdır. Yani yanlış GPS verileri, pusula ya da diğer sensör arızaları hatalı AG deneyimlerine yol açabilir. Mobil cihaz teknolojileri geliştikçe konum bazlı AG deneyimleri de kapalı ve açık alanlarda daha kararlı bir yapıya ulaşacaktır. Konum bazlı AG uygulamalarında mobil cihaz kullanıcının konumunu tanımlar ve ağ bağlantısı sayesinde çevredeki bilgileri getirir. Çevredeki parklar, eczaneler, hastaneler, restoranlar, kamu binaları ya da konum bazlı çalışan sosyal medya içerikleri sanal işaretleyicilerle cihaz ekranından canlı bir şekilde görüntülenebilir (Jackson, Angermann ve Meier, 2011).

Bu araştırmanın birincil amacı konum tabanlı bir mobil arttırılmış gerçeklik uygulaması geliştirmektir. Uygulamanın her aşaması adım adım açıklanmıştır. Ayrıca araştırmacılar tarafından geliştirilen bir anket aracılığıyla uygulama ile ilgili öğrenci görüşleri alınmıştır. Anket altı sorudan oluşmakta ve arttırılmış gerçeklik uygulaması ile ilgili temel düzeyde çeşitli soruları içermektedir. Anket maddeleri alanyazından yararlanılarak oluşturulmuş ve uygulama öncesinde Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü'nde görev yapan doktora derecesine sahip üç farklı uzman tarafından kontrol edilmiştir.

YÖNTEM

Bu bölümde araştırmanın katılımcıları, veri toplama aracı, verilerin analizi ve geliştirilen örnek uygulamanın detaylı anlatımı ele alınmaktadır.

Katılımcı Grubu

Araştırmanın katılımcıları Gazi Üniversitesi Bilişim Enstitüsü öğrencilerinden 2015-2016 Güz ve Bahar yarıyılı Arttırılmış Gerçeklik Öğrenme Ortamları Analiz ve Tasarımı dersini alan yüksek lisans ve doktora öğrencilerinden oluşmaktadır. Araştırmaya 38'i erkek 3'ü kız olmak üzere toplam 41 öğrenci katılmıştır. Öğrenciler Adli Bilişim, Bilişim Sistemleri, Bilgisayar Bilimleri ve Yönetim Bilişim Sistemleri Anabilim Dallarında öğrenimine devam eden öğrencilerdir.

Veri Toplama Aracı

Araştırmada uygulamayı deneyen katılımcıların uygulama ile ilgili görüşlerini öğrenmek amacıyla araştırmacılar tarafından bir anket tasarlanmıştır. Anket alanyazından yararlanılarak uygulama ile ilgili genel görüşleri

öğrenmeyi kapsayan maddelerden oluşmaktadır. Ankette toplam altı madde yer almaktadır. Kullanılan anket 1-5 aralığında olup, 1:Kesinlikle katılmıyorum, 2: Katılmıyorum, 3: Kararsızım, 4: Katılıyorum, 5: Kesinlikle katılıyorum şeklindedir. Verilerin analizinde frekans ve standart sapma değerleri kullanılmıştır. Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü'nden doktora derecesine sahip üç öğretim elemanının görüşleri alındıktan sonra ankete son şekli verilerek veri toplamak üzere hazır hale getirilmiştir.

Arttırılmış Gerçeklik Uygulaması ve Geliştirme Süreci

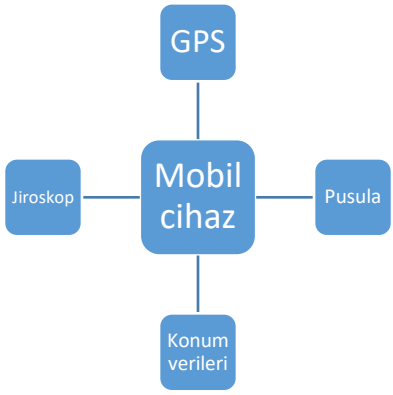
Bu bölümde geliştirilen bir AG uygulamasına ait detaylı açıklamalar sunulmaktadır.

Uygulamanın Tanımı

Araştırma içinde geliştirilen uygulama konum tabanlı bir arttırılmış gerçeklik uygulamasıdır. Uygulama, Gazi Üniversitesi'nin merkez kampüsünde bulunan fakülte binalarının konumlarını mobil telefon vasıtası ile görüntüleme işlemlerini gerçekleştirecek olan bir uygulamadır. Uygulamada kullanılan işaretçiler yardımıyla kullanıcının fakülte binalarının yerlerini mobil akıllı telefondan tespit etmesi sağlanmaktadır. Google harita (Google maps) üzerinden tespit edilen fakülte binalarının enlem boylam bilgileri uygulamanın kullanacağı bir forma dönüştürülmektedir. Uygulamanın kullanıldığı uygun bir forma dönüştürülen koordinat verileri, özel işaretçilerle işlenerek mobil cihazın ekranından görüntülenebilmektedir. Fakülte binalarının mobil cihaza olan uzaklığı da görüntülenebilmektedir.

Yazılım Araçları, Platform ve Kullanılan Teknolojiler

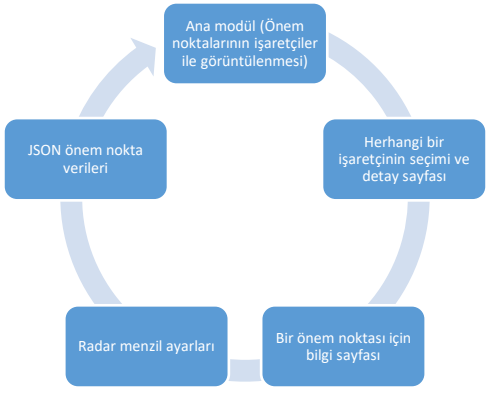
Uygulama Android 4.4.2 platformunda çalıştırılmak üzere geliştirilmiştir. Uygulamayı geliştirmek için Android Studio editörü kullanılmıştır. Uygulama geliştirme için Wikitude Android yazılım geliştirme aracından faydalanılmıştır. Aynı zamanda arayüz tasarımları için Javascript, Html5 ve CSS teknolojileri kullanılmıştır. Fakülte binalarının önem noktası verileri (konumları, isimleri ve açıklamaları) JSON data tipinde uygulama kodları içinde barındırılmaktadır. Temel sistem özellikleri Şekil 1'de gösterilmektedir.



Şekil 1: Temel sistem özellikleri

Uygulama Tasarımı

Uygulama bir ana modül ve ona bağlı olan önem noktalarının bilgilerinin alındığı detay modülü ve radar mesafe aralığı modüllerinden oluşmaktadır. Uygulamaya ait modüller Şekil 2'de görülmektedir. Programın çalıştırılması ile hazır halde bulunan önem noktalarının sisteme yüklenmesi ve işaretçiler yardımıyla uygulama ekranında görüntülenmesi ana işlemi oluşturur. Sonrasında herhangi bir işaretçinin seçimi ve detay sayfasının görüntülenmesi, detay sayfasından noktanın bilgi sayfasına ulaşılması gibi işlemler sonraki adımlarda gerçekleştirilir. Yine aynı şekilde ana ekrandan radar genişlik bilgilerine ve belirli bir mesafe aralığında kaç adet noktanın görüntüleneceği ve düzenleme işlemleri gibi işlemler yapmak mümkün olmaktadır.



Şekil 2: Ana iş akış şeması

Arayüz Tasarımları

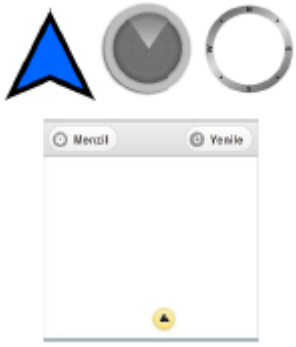
Uygulama içerisinde işaretleyici tasarımları, ana ekran tasarımı, önem noktası detay ekranı tasarımı, önem noktası bilgi ekranı tasarımı ve radar menzil ayarları ekran tasarımı yapılmıştır. Bu tasarımlar aşağıdaki bölümde tanıtılmaktadır.

İşaretleyici tasarımları: İşaretleyiciler önem noktalarının ekranda görüntülenmesini sağlayan birimlerdir. Uygulama için logo renklerine uygun iki farklı biçimde işaretleyici tasarlanmıştır. Birisi işaretleyicinin normal ekran görünümü, diğeri ise önem noktasının seçili hale geldiğinde görüntülenecek olan işaretçi nesnesidir (Şekil 3).



Şekil 3: Standart işaretçi resmi ve seçilmiş işaretçi resmi

Ana ekran tasarımı: Ana ekran tasarımında Wikitude uygulama geliştirme aracı ile gelen radar, pusula sembolleri ve seçili hale gelen işaretçinin takibi için kullanılan ok sembolü kullanılmıştır. Semboller ve ana ekran arayüz html tasarımı Şekil 4'te gösterilmektedir.



Şekil 4: Radar sembolleri ve ana sayfa tasarımı

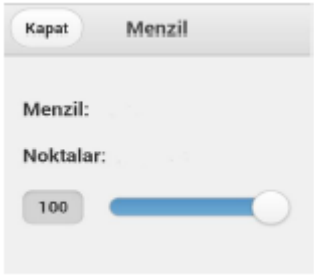
Önem noktası detay ekranı tasarımı: Herhangi bir önem noktası seçildiği taktirde kullanıcı tarafından görüntülenecek olan ekran tasarımıdır. Bu ekranda önem noktası bilgileri ve kullanıcıya olan uzaklığı gibi bilgiler yer almaktadır (Şekil 5).



Şekil 5: Detay sayfa tasarımı

Önem Noktası Bilgi Ekranı tasarımı: Önem noktası bilgi ekranı önem noktası bilgilerinin gösterildiği boş bir sayfa tasarımıdır.

Radar Menzil Ayarları Ekran Tasarımı: Radar menzil ayarlarında radarda gösterilecek olan önem noktalarının menzil ayarlarının yapıldığı tasarım ekranıdır (Şekil 6).



Şekil 6: Menzil sayfa tasarımı

Kodlama ve Kodlar

Kodlama aşamasında Wikitude yazılım geliştirme aracından faydalanılmıştır. Bu aşamada Android işletim sistemi için uygulama geliştirme aracı olan Android Studio platformu kullanılmıştır. Kullanıcı ekranında gerçekleştirilen işlemler için Html, Javascript ve CSS teknolojilerinden yararlanılmıştır. Uygulama için gerekli olan verileri bir JSON dökümanında saklanmakta ve uygulama içerisinden veriler yüklenmektedir. Veri işlemlerinin uygulama içerisinde bulundurulması bu uygulama için konum verilerinin sabit olmasından dolayıdır. Aynı veriler bir web servis vasıtasıyla da alınıp rahatlıkla işlenebilir.

Wikitude yazılım geliştirme aracı, javascript API ile birlikte Html sayfalarında javascript kodlarıyla fonksiyonel işlemlerin geçekleştirilmesine olanak tanır. Bunun için uygulamada oluşturulan Html sayfasına aşağıdaki referans kodu eklenerek, javascript fonksiyonları ile AG nesneleri oluşturulabilir, nesne özelliklerinden ve fonksiyonlarından yararlanılabilir.

```
<script src="architect://architect.js"></script>
```

Tanımlanan API nesnesi ile Wikitude kütüphanesinde tanımlı ARchitect nesneleri tanımlanarak AG ortamı sahnesinde kullanılabilir hale gelmektedir.

Konum verileri: Aşağıdaki veriler JSON data veri örnekleridir.

```
var JsonData = [{
"id": "1",
"longitude": "32.8220565",
"latitude": "39.9395254",
"description": "Rektörlük",
"altitude": "100.0",
"name": "Gazi Üniversitesi"
}, {
"id": "2",
"longitude": "32.822514",
"latitude": "39.942349",
"description": "Fen Edebiyat Fakültesi",
"altitude": "100.0",
"name": "Gazi Üniversitesi"
}
, {
"id": "3",
"longitude": "32.821964",
"latitude": "39.942455",
"description": "Eğitim Fakültesi",
"altitude": "100.0",
"name": "Gazi Üniversitesi"}];
```

Ana İşlevler ve Kullanılan Fonksiyonlar: Wikitude yazılım geliştirme aracının ARchitect nesneleri yardımıyla bir işaretçi resminin arttırılmış gerçeklik sahnesine nasıl eklendiği aşağıda bir örnekle açıklanmıştır. Sonraki bölümlerde uygulama kodlarının da bazı işletilen fonksiyonların açıklamaları yer almaktadır.

Gösterilmek istenen nesnenin lokasyonunun alınan koordinatlar ile oluşturulması aşağıdaki gibidir.

markerLocation = new AR.GeoLocation(latitude, longitude, altitude)

Nokta koordinatları enlem, boylam ve yükseklik referansıyla yeni bir konum nesnesi oluşturulur.

img = new AR.ImageResource("Dosya\img.png");

kodu ile arttırılmış gerçeklik sahnesine eklenecek olan resim dosyasının okunması ve bir resim nesnesi oluşturulması sağlanır.

```
markerDrawable = new AR.ImageDrawable(img, height, {
…..Özellikler-Opacity-OnClick-Rotation
});
```

Oluşturulan img nesnesi referans olarak verilerek ImageDrawable nesnesi oluşturulur. Bunun yanında AR.Label nesnesi ile konumu alınan verilere ait çeşitli bilgiler arttırılmış gerçeklik sahnesinde görüntülenmek üzere Label nesneleri ile oluşturulup ARObject nesnesine parametre olarak verilebilir.

```
AR.Label ( text , height , options );
markerObject = new AR.GeoObject(markerLocation, {
drawables: {
cam: [markerDrawable, Label1, Label2],
indicator: Indicator,
radar: radar
} }
```

Son olarak da yukarıda oluşturulan GeoObject nesnesi konum nesnesi ve etiket nesneleri ile oluşturularak arttırılmış gerçeklik sahnesine eklenmek için hazır hale gelir.

Ana modül ekranı: Ana Modül ekranı ve ana modül üzerinden açılan ekranlar html sayfasıdır ve index.html adındaki dosyada tutulmaktadır. MainActivity'den çağırılan openMapActivity() fonksiyonu yardımıyla index.html sayfası oluşturulur. Bu html dosyanın ilk kısmında tanımlanan marker.js, PoiProcess.js, radar.js işaretçilerin gösterim işlemleri, veri yükleme işlemleri, radar mesafe ve menzil işlemlerinin yapıldığı javascript dosyalarıdır. Marker.js dosyası alınan önem nokta bilgilerinin işaretçiler ile gösterim fonksiyonlarını barındıran dosyadır. Buradaki fonksiyonlarla işaretçiler gösterilir ve seçilme anında tetiklenecek fonksiyonları barındırır. PoiProcess.js dosyası World nesnesinin oluşturulduğu, arayüz tasarımlarında bahsedilen işaretçi sembollerinin sisteme yüklendiği, alınan veriler ile işaretçi nesnelerinin oluşturulduğu, durum mesajlarının oluşturulduğu, konum verilerinin dosyadan okunup yüklendiği, menzil mesafe hesaplamalarının yapıldığı fonksiyonların bulunduğu dosyadır.

Radar.js dosyası radar nesnesinin oluşturulduğu, sembollerinin yüklendiği ve maksimum menzil değerlerinin atandığı fonksiyonları barındıran javascript dosyasıdır. Poiprocess.js dosyasında tanımlanan World nesnesi işaretçileri ve kullanıcı nesnesini bulunduran ana obje niteliğindedir.

```
requestDataFromLocal: function requestDataFromLocalFn() {
World.loadPoisFromJsonData(FacultyJsonData);
}
```

Yukarıdaki fonksiyon önem nokta verilerinin yüklenmesi için çağırılan fonksiyondur. Bu fonksiyon vasıtası ise konum verileri;

```
World.loadPoisFromJsonData(FacultyJsonData)
```

fonksiyonuna parametre olarak verilir ve bu fonksiyon alınan önem nokta konum verilerinden işaretçi nesnesi oluşturarak ana nesne olan World nesnesine listeleyerek atama yapar. Aşağıdaki kodun işlevi alınan önem nokta verilerinden işaretçi (marker) nesneleri oluşturmaktır.

```
for (var currentPlaceNr = 0; currentPlaceNr < poiData.length; currentPlaceNr++) {
var singlePoi = {
"id": poiData[currentPlaceNr].id,
"latitude": parseFloat(poiData[currentPlaceNr].latitude),
"longitude": parseFloat(poiData[currentPlaceNr].longitude),
```

```
"altitude": parseFloat(poiData[currentPlaceNr].altitude),
"title": poiData[currentPlaceNr].name,
"description": poiData[currentPlaceNr].description
};
World.markerList.push(new Marker(singlePoi));
}
```

Oluşturulan işaretçi nesneleri function Marker(poiData) {} ile düzenlenir. İşaretçi nesnesinin değerleri atanır.

var markerLocation = new AR.GeoLocation(poiData.latitude, poiData.longitude, poiData.altitude);

Koordinat değerleriyle birlikte World nesnesinde yüklenilen işaretçilerin resim dosyalarının ataması ve özellik atamaları aşağıdaki gibi yapılır.

```
this.markerDrawable_idle = new AR.ImageDrawable
(World.markerDrawable_idle, 2.5, {
zOrder: 0,
opacity: 1.0,
onClick: Marker.prototype.getOnClickTrigger(this)
});
```

Aynı şekilde işaretçinin seçilme durumunda gösterilecek olan resim içinde nesne oluşturma işlemleri gerçekleştirilir. Ayrıca bu konum bilgilerinin yanında verilerin isim, açıklama gibi alanları da AR.Label nesnesi olarak tanımlanarak oluşturulacak olan işaretçi nesnesine atanabilir. Örnek aşağıdaki kod bloğunda bulunmaktadır.

```
this.titleLabel = new AR.Label(poiData.title.trunc(18), 0.5, {
zOrder: 1,
offsetY: 0.55,
style: {
textColor: '#FFFFFF',
fontStyle: AR.CONST.FONT_STYLE.BOLD
}
```

```
});
this.descriptionLabel = new AR.Label(poiData.description.trunc(30), 0.5, {
zOrder: 1,
offsetY: -0.55,
style: {
textColor: '#FFFFFF'
}
});
```

Burada da görüldüğü gibi etiket nesneleri farklı özellik parametreleriyle oluşturulmaktadır. Ardından aynı işaretçi nesnesinde olduğu gibi radar nesnesi oluşturulup son olarak işaretçi için AR.GeoObject nesnesi oluşturulur ve World nesnesinin MarkerList işaretçi listesine atanarak ekranda işaretçilerin gözükmesi sağlanmış olur.

```
this.markerObject = new AR.GeoObject(markerLocation, {
drawables: {
cam: [this.markerDrawable_idle, this.markerDrawable_selected, this.titleLabel, this.descriptionLabel],
indicator: this.directionIndicatorDrawable,
radar: this.radardrawables
}
});
```

Aşağıdaki html kodlarında ise kodlama bölümümün giriş kısmında bahsedilen Architects API'nin uygulama butonlarının eklenmesi gibi işlemler gerçekleşmektedir.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<meta http-equiv="Content-Type" content="application/json; charset=utf-8">
<meta content="width=device-width,initial-scale=1,maximum-scale=5, user-scalable=yes" name="viewport">
```

```
<title>World</title>
<script src="architect://architect.js"></script>
<!—radar css dosyası -->
<link rel="stylesheet" href="css/poi-radar.css" />
<link rel="stylesheet" href="jquery/jquery.mobile-1.3.2.min.css" />
<!-- required to set background transparent & enable "click through" -->
<link rel="stylesheet" href="jquery/jquery-mobile-transparent-ui-overlay.css" />
<script type="text/javascript" src="jquery/jquery-1.9.1.min.js"></script>
<script type="text/javascript" src="jquery/jquery.mobile-1.3.2.min.js"></script>
<!—işaretçilerin gösterimi-->
<script src="js/marker.js"></script>
<!—Önem Noktaları -->
<script type="text/javascript" src="js/JsonData.js"></script>
<!—Ana Modül İşlemleri-->
<script type="text/javascript" src="js/nativePoiDetail.js"></script>
<!—radar -->
<script type="text/javascript" src="js/radar.js"></script>
<script type="text/javascript">
</script>
</head>
<body>
<div data-role="page" id="page1" style="background: none;" >
<!—Ana Sayfa İçeriği -->
<!—Başlık Kısmında bulunan butonlar -->
<div id ="header-status" data-role="header" data-position="fixed" data-theme="c">
<a href="javascript: World.showRange();" data-icon="gear" data-inline="true" data-mini="true">Menzil</a>
<a href="javascript: World.reloadPlaces()" data-icon="refresh" >Yenile</a>
<h1></h1>
</div>
<!—Radar -->
<div class="radarContainer_left" id="radarContainer"></div>
```

```
<div data-role="footer" class="ui-bar" data-theme="f" data-position="fixed" style="text-align:center;">

<!—Durum Butonu -->

<a style="text-align:right;" id="popupInfoButton" href="#popupInfo" data-rel="popup" data-role="button" class="ui-icon-alt" data-inline="true" data-transition="pop" data-icon="alert" data-theme="e" data-iconpos="notext">Log</a> </p>

<!-- popup -->

<div data-role="popup" id="popupInfo" class="ui-content" data-theme="e" style="max-width:350px;">

<p style="text-align:right;" id="status-message"> </p>

</div>

</div>

<!—Önem Nokta detay Paneli -->

<div data-role="panel" id="panel-poidetail" data-position="right" data-display="overlay" style="background-color:#F0F0F0;" data-theme="c">

<!—Panel Butonları -->

<div data-role="header" data-theme="c">

<h1>Detaylar</h1>

<a href="#header" data-rel="close">Kapat</a>

</div>

<div data-role="content">

<h3 id="poi-detail-title"></h3>

<!--Açıklama -->

<h4 id="poi-detail-description"></h4>

<!-- Mesafe -->

<h4>Mesafe: <a id="poi-detail-distance"></a></h4>

<!—Bilgi Butonu-->

<a href="javascript: World.onPoiDetailMoreButtonClicked();" data-role= "button" data-icon="arrow-r" data-iconpos="right" data-inline="true">Bilgi</a>

</div>

</div>

<!—Menzil Paneli -->

<div data-role="panel" id="panel-distance" data-position="left" data-display="overlay" style="background-color:#F0F0F0;" data-theme="c">
```

```
<!—Panel Butonları -->
<div data-role="header" data-theme="c">
<h1>Menzil</h1>
<a href="#header" data-rel="close">Kapat</a>
</div>
<div data-role="content">
<!—Menzil-->
<h4> Menzil: <a id="panel-distance-value"></a></h4>
<!—Görüntülenen noktalar -->
<h4> Noktalar: <a id="panel-distance-places"></a></h4>
<!-Kaydırma Cubugu>
<input id="panel-distance-range" type="range" data-highlight="true" name="rangeSlider" min="0" max="100" value="100" data-show-value="false" step="5" data-popup-enabled="false">
</div>
</div>
</div>
</body>
</html>
```

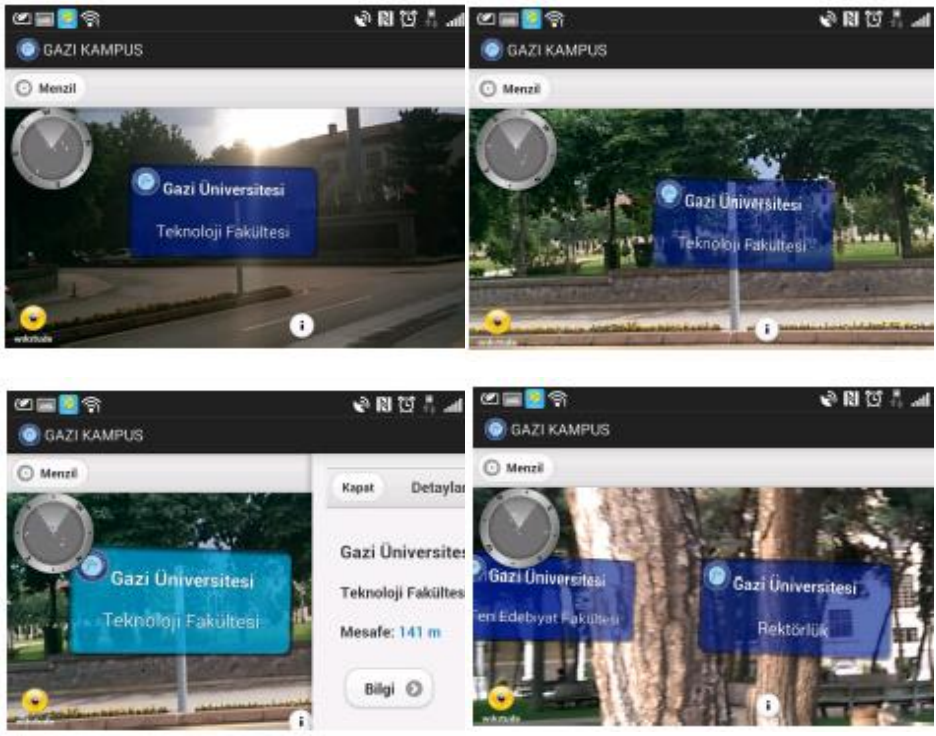
Yukarıda bazı örneklerle açıklanan javascript dosyalarına ait tüm kodlar EK 1'de sunulmuştur.

Uygulamanın Testi

Test işlemleri lokal verilerden yola çıkılarak yapılmıştır. Maddeler halinde aşağıdaki test işlemleri gerçekleştirilmiştir.

- İşaretçilerin önem nokta konumlarını göstermesi: İşaretçiler eklenen koordinatlarda genel anlamda başarılı bir şekilde görüntülenmiştir. Fakat zaman zaman mobil cihaz donanımına ve kullanıcı GPS verilerine dayanan sapmalardan dolayı işaretçilerin konumlarında değişiklikler meydana gelmektedir.
- İşaretçilerin mesafe gösterimi: Kullanıcı ile işaretçiler arasında hesaplanan mesafe verileri yine GPS verilerine göre değişimler göstermiştir. Mesela aynı noktada sabit bir şekilde durulduğunda kullanıcı ile işaretçiler arasında kullanıcı konumunun güncel-

lenmesi ile 3-10 metrelik farklılıklar meydana gelebilmektedir. Bu sapmalar iç mekanlarda yer tespitinin zor olmasından kaynaklı daha fazla olmaktadır. Aşağıdaki resimler programdan alınmış ekran görüntüleridir (Resim1,2,3,4).



Resim 1,2,3,4: Uygulamadan alınan dört farklı ekran görüntüsü

BULGULAR ve YORUM

Yukarıda tasarım ve geliştirme adımları detaylı olarak açıklanan uygulama ile ilgili olarak "Arttırılmış gerçeklik ortamları analiz ve tasarımı" dersini alan yüksek lisans ve doktora öğrencilerinin görüşleri alınmıştır. Öğrencilerin geliştirilen uygulama ile ilgili olarak görüşleri Tablo 1'de sunulmaktadır.

Tablo 1: Mobil arttırılmış gerçeklik uygulaması ile ilgili öğrenci görüşleri

İfade	Ortalama	Standart Sapma
Böyle bir uygulama yeni yerlerin keşfi (kampüs, müze, şehir merkezi, vb. yerler) için geliştirilebilir	4.55	0.59
Uygulamanın yararlı olduğunu düşünüyorum	4.88	0.67
Uygulamanın birçok avantajı bulunmaktadır	4.51	0.45
Böyle bir uygulamayı ben de geliştirebilirim	3.03	0.66
Uygulamayı geliştirmek zordur	3.21	0.61
Uygulama geliştirmek için yüksek düzeyde teknik beceri gerekmektedir	4.10	0.44

Tablo 1'de görüldüğü üzere, öğrencilerin büyük çoğunluğu uygulamanın farklı yerlerin keşfedilmesinde kullanılabileceğini düşünmektedir. Katılımcılar arttırılmış gerçeklik uygulamasının yararlı olduğunu ve çeşitli avantajlara sahip olduğunu düşünmektedirler. Bütün bu olumlu düşüncelerine karşın, katılımcılar benzer bir uygulamayı kendilerinin geliştirmesi hususunda zorluklar olduğunu düşünmektedirler. Katılımcılar uygulama geliştirmenin zorluğu hususunda ise kararsızdır. Üstelik uygulama geliştirebilmek için çeşitli teknik becerilere sahip olunması gerektiğini düşünmektedirler.

SONUÇ, TARTIŞMA ve ÖNERİLER

Bu çalışmada arttırılmış gerçeklik kavramı incelenerek örnek bir AG uygulaması geliştirme adımları açıklanmıştır. Geliştirilen AG uygulamasına ait detaylar paylaşılarak ardından öğrencilerin görüşlerine yer verilmiştir. Alanyazın incelendiğinde AG uygulamalarının nasıl geliştirildiğine dair detaylı kodları içeren bir bilgiye pek fazla rastlanmadığı görülmektedir. Arttırılmış gerçeklikten yararlanmak isteyen bireylerin kullandığı cihazlara ve planlanan arttırılmış gerçeklik çeşidine göre, eğitimde bu uygulamaları kullanmak ve oluşturmak isteyen eğitimciler için birçok araç bulunmaktadır. Bu araçlardan en belirgin ve kullanışlı olanlarına aşağıda kısaca değinilmektedir.

Daqri (http://www.daqri.com) kullanıcıların telefon kameralarını çalıştırdıklarında resim, ses, video vb. dijital içeriklere kolayca ulaşmalarını sağlayan QR kod (karekod) üretmelerini sağlamaktadır. Bunun için iPhone ya da Android cihazın bulunması yeterlidir. Herhangi bir programlama becerisi gerektirmeden dijital dünyaya kapılar açılmaktadır (Yuen, 2011). iPhone için Hololabs stüdyosu tarafından MixAR isimli bir uygulama geliştirilmektedir. Bu uygulama kullanıcının 3D model, fotoğraf ve video üretmesini kod bilmeden yapabilmesine olanak sağlamaktadır. Şu an için sadece ios işletim

sistemi kullanan cihazlarda çalışmaktadır. Ayrıca çalışmalar pilot olarak yürütüldüğünden kişilerin kullanımına şu an için henüz açılmamıştır (Gordon, 2011). ZooBurst (http://www.zooburst.com), yazarlar için geliştirilmiş bir arttırılmış gerçeklik hikaye aracıdır. Yazarlar bu uygulama aracılığıyla okuyucular için çeşitli etkileşim araçları sunabilirler. Bu uygulama ile kitap sayfaları kolaylıkla değiştirilebilir, diyalogları görmek için karakterlerin üzerine tıklanabilir, kitap sayfası farklı yönlerde hareket ettirilebilir ve kitap sayfalarına ses entegre edilebilir (Carr, 2011). ARToolKit açık kaynak kodlu, ücretsiz ve C programlama dili ile hazırlanan zengin kütüphaneye sahip bir uygulama aracıdır. 1999 yılında Hirokazu Kato tarafından geliştirilmiş, ardından birçok enstitü ve üniversite tarafından desteklenmiştir. Programlama becerisi gerektiren ARToolKit ile şu ana kadar 3D-Live, AR Groove, FaiMR, MagicBook ve PyARTK gibi birçok arttırılmış gerçeklik projesi geliştirilmiştir (Billinghurst, Kato ve Poupyrev, 2001). iPhone, Android, Symbian ve Windows Mobile işletim sistemleri için kullanıcıların arttırılmış gerçeklik uygulamaları geliştirmelerine izin veren Unifeye Mobile SDK Metaio tarafından geliştirilmiştir. Unifeye Mobile SDK kullanıcılara çok kapsamlı bir kütüphane sunmaktadır. Dolayısıyla kütüphanesi güçlü olan bu program içerisinde programcının arttırılmış gerçeklik uygulamaları geliştirmesinde esneklik sunulmaktadır (Metaio, 2011).

Arttırılmış gerçeklik ve kullanılan geliştirme araçları ilgili yukarıda özetlenen çeşitli araştırmalar derinlemesine incelendiğinde, bu araştırmaların birçoğunda amaca yönelik olarak tasarlanmış özel gözlüklerin (head mounted display - HMD) kullanıldığı görülmektedir. Bir arttırılmış gerçeklik ara yüzü içerisinde, kişi bu özel gözlük aracılığıyla gerçek dünyaya bakmaktadır. Bu gözlüklerin üzerinde özel lensler bulunmakta ve bu sayede kişinin döndüğü yöne göre özel lensler aracılığıyla gerçek dünya nesneleri dijital dünyanın nesneleri ile desteklenmektedir. Ayrıca bu gözlükler kablolar aracılığıyla bir bilgisayara bağlanmaktadır. Bu da arttırılmış gerçeklik uygulanmasının laboratuvar dışına çıkmasına engel olmaktadır. Bu engelin aşılmasına yardımcı olan gelişmeleri ise Internet, kablosuz ağ, Wi-Fi ve bunlara paralel olarak yukarıda sözü edilen ve her bireyin taşıdığı akıllı telefon, tablet bilgisayar ve diz üstü bilgisayarlardaki gelişmeler olarak sıralamak mümkündür. Günümüzdeki tüm bu gelişmeler düşünüldüğünde arttırılmış gerçekliğin laboratuvar ortamından çıkıp kişisel kullanıma da kolaylıkla açılabileceği açıkça görülmektedir.

Yazılım alanında geçmişten günümüze kadar yapılan araştırma ve geliştirmelerin sonucu olarak birçok alanda yazılım geliştiriciler için çok kullanışlı uygulama kütüphaneleri oluşmuş ve hala yeni teknolojilerle desteklenerek bireylerin kullanımı için yeni teknolojiler geliştirilmektedir. Aynı şekilde arttırılmış gerçeklik teknolojilerinde de geliştirilen araçlar vasıtasıyla hiç yazılım bilgisi olmadan bile arttırılmış gerçeklik uygulamaları geliştirmek mümkün hale gelmiştir. Bu projenin amacı da AG teknolojileri hakkın-

da genel bilgiler vermek ve var olan araç ve kütüphaneleri kullanarak AG uygulamalarının geliştirme aşamalarını göstermektir. Akıllı telefonlar üzerinde arttırılmış gerçeklik uygulamalarının tasarımını anlatan çeşitli rehberlere de alanyazında rastlanmaktadır (Butchart, 2011).

Bu çalışmaya katılan öğrenciler arttırılmış gerçeklik uygulamaları ve çeşitli disiplinlerde kullanımı ile ilgili olarak olumlu görüşlere sahiptir. Öğrenciler AG uygulamalarının yararlı olduğunu düşünmektedirler. Alanyazın incelendiğinde de benzer sonuçlar görülmektedir. El Sayed, Zayed ve Sharawy (2011)'nin geliştirdiği arttırılmış gerçeklik öğrenme kartlarının, öğrencilerin içeriğe karşı olumlu etkileri olduğu sonucuna varılmıştır. Juan, Llop, Abad ve Lluch (2010) kelime öğretiminde arttırılmış gerçekliğin yararlı olduğu sonucuna ulaşmıştır. Alanyazın incelendiğinde arttırılmış gerçeklik uygulamaları konusunda öğrencilerin tutumunun da olumlu olduğu sonucuna ulaşan çeşitli çalışmalar olduğu görülmektedir (Lee, 2012; Shelton ve Hedley, 2004; Küçük, Yılmaz ve Göktaş, 2014;).

Arttırılmış gerçeklik teknolojileri eğlenceden eğitime, coğrafi bilgi sistemlerinden tıp bilimine kadar geniş kullanım alanları sayesinde büyük bir potansiyele sahiptir. Arttırılmış gerçeklik, genel anlamda görselliğin ön planda olduğu bir teknoloji olduğu için yaygın olarak reklamcılık, oyun ve eğlence örnekleri fazlasıyla karşımıza çıksa da, eğitim alanında, turizmde, tıp biliminde, üretim teknolojilerinde de çok yararlı uygulamalara sahiptir ve durmadan gelişen teknolojilerle böyle olmaya devam edecek gibi görünmektedir. Juniper Research'un 4 Şubat 2014 tarihli raporuna göre mobil arttırılmış gerçeklik gelirlerinin yılda 1 milyar doları aşması beklendiği belirtilmiştir. Aynı rapora göre 2018 yılına kadar arttırılmış gerçeklik kullanıcılarının sayısının 200 milyona yaklaşması beklenmektedir.

Ülkemizde bu uygulamalardan yararlanacak birey sayısında artış olacağı öngörülebilir. Ülkemiz özellikle konum tabanlı arttırılmış gerçeklik uygulamaları ve etkileşimli AG müze uygulamaları geliştirilmesi için çok elverişli imkanlara sahiptir. Türkiye'de belediyeler, bazı devlet kurumları, özel kuruluşlar coğrafi bilgi sistemlerinden yararlanmaktadır. Ayrıca yaygın olarak kullanılan veri tabanı sistemleri (IBM DB2,Oracle, MS SQL Server gibi) mekansal verileri desteklemektedir. Coğrafi bilgi sistemlerinin kullanımının artmasının mobil arttırılmış gerçeklikle ilgisi, mekansal olarak saklanan veri hacminin artmasıdır. Yani ülke çapında belediyeler, devlet kurumları ya da özel telekomünikasyon şirketlerinin altyapı hizmetleriyle ilgili mekansal milyarlarca veri, veri tabanlarında tutulmakta ve işlenmektedir. Bunun anlamı konum tabanlı arttırılmış gerçeklik teknolojileri için görüntülenebilecek milyarlarca veri bulunduğudur. Arttırılmış gerçeklik teknolojilerinin gelişmesiyle kurumlar açısından farklı çözümler üretmek için ülkemizde de sadece arttırılmış gerçeklik uygulamaları geliştiren şirketlerin sayısının artması olasıdır. Bilişim teknolojilerinin hızlı gelişimi bilgi güvenliği, özel hayatın gizliği, sosyal hayatın olumsuz yönde etkilenmesi gibi konularda

endişelere sebep olsa da, yeni teknolojilerin yarar sağlayacak alanlarda kullanılması, eğitime katkısının olması ve tıbbi çözümler sunması insanlar için fazlasıyla avantaj sağlamaktadır.

Bu araştırmada ortaya konulan arttırılmış gerçeklik ortamı geliştirme adımları sadece yazılım geliştiriciler için değil özellikle öğretim teknologları açısından da oldukça önemlidir. Eğitimsel arttırılmış gerçeklik uygulamalarının geliştirilmesine yönelik çalışma yapmak isteyen öğretim teknologlarının da bu araştırmadan yararlanacakları düşünülmektedir. Kodlama becerisi gerektiren veya gerektirmeyen birçok yazılım geliştirme platformu mevcuttur. Bütün bu bilgiler ışığında bu alanda uygulama geliştirmek ve uygulamanın çeşitli değişkenler üzerindeki etkisini incelemek isteyen araştırmacılar, bu platformlardan rahatlıkla yararlanabilirler.

KAYNAKÇA

BILLINGHURST, M., KATO, H. VE POUPYREV, I. (2001). The Magic Book — Moving seamlessly between reality and virtuality. *IEEE Computers, Graphics and Applications,* S: 21(3).

BUTCHART, B. (2011). Augmented reality for smartphones: a guide for developers and content publishers. 15 Ağustos 2016 tarihinde observatory.jisc.ac.uk/docs/AR_Smartphones.pdf. adresinden erişilmiştir.

CARR, A. (2011). BMW to Launch NYC Tech Incubator with $100 million investment fund. 20 Kasım 2015 tarihinde http://www.fastcompany.com/1743933/bmw-to-launch-nyc-tech-incubator adresinden erişilmiştir.

CRAIG, A. B. (2013). Understanding Augmented Reality: Concepts and Applications, *Newnes* S: 1, (37).

EL SAYED, N. A. M., ZAYED, H. H. VE SHARAWY, M. I. (2011). ARSC: Augmented reality student card. *Computers & Education,* S: 56(4).

GORDON, K. (2011). MixAR: World's First 3D Augmented Reality Editor for iPhone. 6 Şubat 2016 tarihinde http://www.psfk.com/2011/02/mixar-worlds-first-3d-augmented-reality-editorfor-iphone.htm. adresinden erişilmiştir.

HEYDARZADEH, M., NOURANI, M. VE PARK, J. (2015). "An Augmented Reality Platform for CABG Surgery", *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)* S: 1-4.

HSIEH, M. C. VE LIN, H. C. K. (2011). A Conceptual Study for Augmented Reality E-learning System Based on Usability Evaluation, *CISME*, S: 1(8), 5-7.

JACKSON, T., ANGERMANN, F. VE MEIER, P. (2011). Survey of Use Cases for Mobile Augmented Reality Browsers. Handbook of Augmented Reality, Springer New York.

JUAN, C. M., LLOP, E., ABAD, F. VE LLUCH, J. (2010). Learning words using augmented reality. In The *10th IEEE international conference on advanced learning technologies*, Washington, DC, USA: IEEE Computer Society.

KAUFMANN, H. (2003). Collaborative augmented reality in education. *Proceedings of Imagina 2003 conference*, S: 1-4.

KAUFMANN, H. VE DUNSER, A. (2007). Summary of usability evaluations of an educational augmented reality application . *Lecture Notes in Computer Science*, 4563, 660-669.

KAUFMANN, H. VE SCHMALSTIEG, D. (2003). Mathematics and geometry education with collaborative augmented reality. *Computers & Graphics*, S: 27.

KERAWALLA, L., LUCKIN, R., SELJEFLOT, S. VE WOOLARD, A. (2006). Making it real: Exploring the potential of augmented reality for teaching primary school science. *Virtual Reality* , S: 10(3-4).

KÜÇÜK, S., KAPAKİN, S. VE GÖKTAŞ, Y. (2015). "Tıp Fakültesi Öğrencilerinin Mobil Artırılmış Gerçeklikle Anatomi Öğrenimine Yönelik Görüşleri", *Journal of Higher Education & Science / Yükseköğretim ve Bilim Dergisi*. S: 5(3).

LEE, K. (2012). Augmented reality in education and training. *TechTrends*, S: 56(2).

LIAROKAPİS, F., MOURKOUSSIS, N., WHITE, M., DARCY, J., SIFNIOTIS, M. VE PETRIDIS, P. (2004). Web 3D and augmented reality to support engineering education. *World Transactions on Engineering and Technology Education*, S: 3(1).

LUDWIG, C. VE REIMANN, C. (2005). Augmented reality: Information at focus", *Cooperative Computing & Communication Laboratory*, *4*(1), Universitat Paderborn.

METAIO. (2011). Mobile SDK. 1 Eylül 2015 tarihinde http://www.metaio.com/software/mobile-sdk/ adresinden erişilmiştir.

RIEBER, L. (1996). Seriously Considering Play: Designing Interactive Learning Environments Based on the Blending of Microworlds, Simulations, and Games. *Educational Technology Research and Development*, S: 44(2).

SHELTON, B. E. VE HEDLEY, N. R. (2004). Exploring a Cognitive Basis for Learning Spatial Relationships with Augmented Reality. *Technology, Instruction, Cognition and Learning*, 1(4), 323-357. Philadelphia, PA: Old City Publishing, Inc.

SIELHORST, T., OBST, T., BURGKART, R., RİENER, R. VE NAVAB, N. (2004). An augmented reality delivery simulator for medical training. *In International Workshop on Augmented Environments for Medical Imaging - MICCAI Satellite Workshop*. AMI-ARCS.

SIELHORST, T., FEUERSTEIN, M. VE NAVAB, N. (2008). "Advanced Medical Displays: A Literature Review of Augment Reality",. *Journal of Display Display Technology*, S: 4(4).

WOJCIECHOWSKİ, R. VE CELLARY, W. (2013). "Evaluation of Learners' Attitude Toward Learning in ARIES Augmented Reality Environments", *Computers & Education*, S: 68.

YUEN, S. C. Y. (2011). Augmented Reality Helicopter. 5 Ağustos 2016 tarihinde http://steveyuen.org/blog/?p=1082 adresinden erişilmiştir.

ZHOU, F., DUH, H. L. VE BILLINGHURST, M. (2008). Trends in augmented reality traching, iteraction and display: A review of ten years in ISMAR. Mixed and Augmented Reality, ISMAR 7th IEE/ACM International Symposium, 193-202.

K 1. Çalışmada Kullanılan Javascript Kodları

Marker.js

```
function Marker(poiData) {
this.poiData = poiData;
this.isSelected = false;
this.animationGroup_idle = null;
this.animationGroup_selected = null;
var markerLocation = new AR.GeoLocation(poiData.latitude, poiData.longitude, poiData.altitude);
this.markerDrawable_idle = new AR.ImageDrawable(World.markerDrawable_idle, 2.5, {
zOrder: 0,
opacity: 1.0,
onClick: Marker.prototype.getOnClickTrigger(this)
}); this.markerDrawable_selected = new AR.ImageDrawable(World.markerDrawable_selected, 2.5, {
zOrder: 0,
opacity: 0.0,
onClick: null
}); this.titleLabel = new AR.Label(poiData.title.trunc(18), 0.5, {
zOrder: 1,
offsetY: 0.55,
style: {
textColor: '#FFFFFF',
fontStyle: AR.CONST.FONT_STYLE.BOLD
} });
this.descriptionLabel = new AR.Label(poiData.description.trunc(30), 0.5, {
zOrder: 1,
offsetY: -0.55,
style: {
textColor: '#FFFFFF'
} });
this.directionIndicatorDrawable = new AR.ImageDrawable(World.markerDrawable_directionIndicator,
0.1, {
enabled: false,
verticalAnchor: AR.CONST.VERTICAL_ANCHOR.TOP
});
this.radarCircle = new AR.Circle(0.03, {
horizontalAnchor: AR.CONST.HORIZONTAL_ANCHOR.CENTER,
```

```
opacity: 0.8,
style: {
fillColor: "#ffffff"
} });
this.radarCircleSelected = new AR.Circle(0.05, {
horizontalAnchor: AR.CONST.HORIZONTAL_ANCHOR.CENTER,
opacity: 0.8,
style: {
fillColor: "#0066ff"
} });
this.radardrawables = [];
this.radardrawables.push(this.radarCircle);
this.radardrawablesSelected = [];
this.radardrawablesSelected.push(this.radarCircleSelected);
this.markerObject = new AR.GeoObject(markerLocation, {
drawables: {
cam: [this.markerDrawable_idle, this.markerDrawable_selected, this.titleLabel, this.descriptionLabel],
indicator: this.directionIndicatorDrawable,
radar: this.radardrawables
} );
return this;
}
Marker.prototype.getOnClickTrigger = function(marker) {
return function() {
if (!Marker.prototype.isAnyAnimationRunning(marker)) {
if (marker.isSelected) {
Marker.prototype.setDeselected(marker);


PropertyAnimation(marker.markerDrawable_idle, 'scaling', null, 1.0, kMar-
ker_AnimationDuration_Resize, new
AR.EasingCurve(AR.CONST.EASING_CURVE_TYPE.EASE_OUT_ELASTIC, {
amplitude: 2.0
}));
var selectedDrawableResizeAnimation = new AR.PropertyAnimation(marker.markerDrawable_selected,
'scaling', null, 1.0, kMarker_AnimationDuration_Resize, new
AR.EasingCurve(AR.CONST.EASING_CURVE_TYPE.EASE_OUT_ELASTIC, {
amplitude: 2.0
}));
var titleLabelResizeAnimation = new AR.PropertyAnimation(marker.titleLabel, 'scaling', null, 1.0,
kMarker_AnimationDuration_Resize, new
AR.EasingCurve(AR.CONST.EASING_CURVE_TYPE.EASE_OUT_ELASTIC, {
amplitude: 2.0
}));
var descriptionLabelResizeAnimation = new AR.PropertyAnimation(marker.descriptionLabel, 'scaling',
null, 1.0, kMarker_AnimationDuration_Resize, new
AR.EasingCurve(AR.CONST.EASING_CURVE_TYPE.EASE_OUT_ELASTIC, {
amplitude: 2.0
}));
marker.animationGroup_idle=new
AR.AnimationGroup(AR.CONST.ANIMATION_GROUP_TYPE.PARALLEL, [showIdleDrawableA-
nimation,hideSelectedDrawableAnimation, idleDrawableResizeAnima-
tion,selectedDrawableResizeAnimation, titleLabelResizeAnimation, descriptionLabelResizeAnimation]);
```

```
} marker.markerDrawable_idle.onClick = Marker.prototype.getOnClickTrigger(marker);
marker.markerDrawable_selected.onClick = null;
marker.directionIndicatorDrawable.enabled = false;
marker.animationGroup_idle.start();
}; Marker.prototype.isAnyAnimationRunning = function(marker) {
if (marker.animationGroup_idle === null || marker.animationGroup_selected === null) {
return false;
} else { if ((marker.animationGroup_idle.isRunning() === true) || (mar-
ker.animationGroup_selected.isRunning() === true)) {
return true; } else { return false; }} }; String.prototype.trunc = function(n) { return this.substr(0, n - 1) +
(this.length > n ? '...' : '');
```

}; *poiProcess.js* dosyasında verilerin yüklenme işlemleri,mesafelere göre işaretçilerin konumlandırılma işlemleri,kullanıcı cihazının konum verilerine göre yapılan mesafe ölçüm fonksiyonları yer almaktadır.

```
var World = { userLocation: null,
isRequestingData: false,
initiallyLoadedData: false,
markerDrawable_idle: null,
markerDrawable_selected: null,
markerDrawable_directionIndicator: null,
markerList: [],
currentMarker: null,
locationUpdateCounter: 0,
updatePlacemarkDistancesEveryXLocationUpdates: 10,
// Noktaların alınması ve işaretçilerin oluşturulması
loadPoisFromJsonData: function loadPoisFromJsonDataFn(poiData) {
AR.context.destroyAll();
PoiRadar.show();
$('#radarContainer').unbind('click');
$("#radarContainer").click(PoiRadar.clickedRadar);
World.markerList = [];
//İşaretçilerin yüklenmesi
World.markerDrawable_idle = new AR.ImageResource("assets/marker_idle.png");
World.markerDrawable_selected = new AR.ImageResource("assets/marker_selected.png");
World.markerDrawable_directionIndicator = new AR.ImageResource("assets/indi.png");
// Her nokta için İşaretçilerin Oluşturulması
for (var currentPlaceNr = 0; currentPlaceNr < poiData.length; currentPlaceNr++) {
var singlePoi = {
"id": poiData[currentPlaceNr].id,
"latitude": parseFloat(poiData[currentPlaceNr].latitude),
"longitude": parseFloat(poiData[currentPlaceNr].longitude),
"altitude": parseFloat(poiData[currentPlaceNr].altitude),
"title": poiData[currentPlaceNr].name,
"description": poiData[currentPlaceNr].description
}; World.markerList.push(new Marker(singlePoi));
} World.updateDistanceToUserValues();
World.updateStatusMessage(currentPlaceNr + ' nokta yüklendi');
$("#panel-distance-range").val(100);
$("#panel-distance-range").slider("refresh");
}, updateDistanceToUserValues: function updateDistanceToUserValuesFn() {
for (var i = 0; i < World.markerList.length; i++) {
World.markerList[i].distanceToUser = World.markerList[i].markerObject.locations[0].distanceToUser();
```

```
} }, updateStatusMessage: function updateStatusMessageFn(message, isWarning) {
var themeToUse = isWarning ? "e" : "c";
var iconToUse = isWarning ? "alert" : "info";
$("#status-message").html(message);
$("#popupInfoButton").buttonMarkup({
theme: themeToUse
}); $("#popupInfoButton").buttonMarkup({
icon: iconToUse
}); }, onPoiDetailMoreButtonClicked: function onPoiDetailMoreButtonClickedFn() {
var currentMarker = World.currentMarker;
var architectSdkUrl = "architectsdk://markerselected?id=" + encodeURICompo-
nent(currentMarker.poiData.id) + "&title=" + encodeURIComponent(currentMarker.poiData.title) +
"&description=" + encodeURIComponent(currentMarker.poiData.description);
document.location = architectSdkUrl; },
//Kullanıcı konumu
locationChanged: function locationChangedFn(lat, lon, alt, acc) {
World.userLocation = {
'latitude': lat,
'longitude': lon,
'altitude': alt,
'accuracy': acc
}; //Konum verilerinin yüklenmesi
if (!World.initiallyLoadedData) {
World.requestDataFromLocal();
World.initiallyLoadedData = true;
} else if (World.locationUpdateCounter === 0) {
World.updateDistanceToUserValues();
} World.locationUpdateCounter = (++World.locationUpdateCounter %
World.updatePlacemarkDistancesEveryXLocationUpdates);
}, //İşaretçinin seçilme işlemi
onMarkerSelected: function onMarkerSelectedFn(marker) {
World.currentMarker = marker;
$("#poi-detail-title").html(marker.poiData.title);
$("#poi-detail-description").html(marker.poiData.description);
var distanceToUserValue = (marker.distanceToUser > 999) ? ((marker.distanceToUser / 1000).toFixed(2)
+ " km") : (Math.round(marker.distanceToUser) + " m");
$("#poi-detail-distance").html(distanceToUserValue);
// show panel
$("#panel-poidetail").panel("open", 123);
$( ".ui-panel-dismiss" ).unbind("mousedown");
$("#panel-poidetail").on("panelbeforeclose", function(event, ui) {
World.currentMarker.setDeselected(World.currentMarker);
}); },//işaretçilerin kullanıcıya olan uzaklıklarına göre sıralanması ve en uzak noktanın alınması
getMaxDistance: function getMaxDistanceFn() {
World.markerList.sort(World.sortByDistanceSortingDescending);
var maxDistanceMeters = World.markerList[0].distanceToUser;
return maxDistanceMeters * 1.1;
}, updateRangeValues: function updateRangeValuesFn() {
var slider_value = $("#panel-distance-range").val();
// Menzil değerinin maksimum mesafeye göre ayarlanması
var maxRangeMeters = Math.round(World.getMaxDistance() * (slider_value / 100));
```

```
// metre ve kilometre menzil hesabı
var maxRangeValue = (maxRangeMeters > 999) ? ((maxRangeMeters / 1000).toFixed(2) + " km") :
(Math.round(maxRangeMeters) + " m");
// Maksimum menzilde bulunan nokta sayısı
varplacesInRange= World.getNumberOfVisiblePlacesInRange(maxRangeMeters);
$("#panel-distance-value").html(maxRangeValue);
$("#panel-distance-places").html((placesInRange != 1) ? (placesInRange + " Noktalar") : (placesInRange
+ " Nokta"));
AR.context.scene.cullingDistance = Math.max(maxRangeMeters, 1);
PoiRadar.setMaxDistance(Math.max(maxRangeMeters, 1));
}, getNumberOfVisiblePlacesInRange:function getNumberOfVisiblePlacesInRangeFn(maxRangeMeters)
{
World.markerList.sort(World.sortByDistanceSorting);
// loop through list and stop once a placemark is out of range ( -> very basic implementation )
for (var i = 0; i < World.markerList.length; i++) {
if (World.markerList[i].distanceToUser > maxRangeMeters) {
return i;
} }; return World.markerList.length;
}, handlePanelMovements: function handlePanelMovementsFn() {
$("#panel-distance").on("panelclose", function(event, ui) {
$("#radarContainer").addClass("radarContainer_left");
$("#radarContainer").removeClass("radarContainer_right");
PoiRadar.updatePosition();
}); $("#panel-distance").on("panelopen", function(event, ui) {
$("#radarContainer").removeClass("radarContainer_left");
$("#radarContainer").addClass("radarContainer_right");
PoiRadar.updatePosition();
}); }, showRange: function showRangeFn() {
if (World.markerList.length > 0) {
// update labels on every range movement
$('#panel-distance-range').change(function() {
World.updateRangeValues();
}); World.updateRangeValues();
World.handlePanelMovements();
// open panel
$("#panel-distance").trigger("updatelayout");
$("#panel-distance").panel("open", 1234);
} else { World.updateStatusMessage('Noktalar henüz yüklenmedi', true);
} }, //Nokta verilerinin tekrar yüklenmesi
reloadPlaces: function reloadPlacesFn() {
if (!World.isRequestingData) {
if (World.userLocation) {
World.requestDataFromLocal();
} else { World.updateStatusMessage('Bilinmeyen Konum.', true);
} } else { World.updateStatusMessage('Noktalar Yükleniyor...', true);
} },
// Konumların dosyadan alınması
requestDataFromLocal: function requestDataFromLocalFn() {
World.loadPoisFromJsonData(FacultyJsonData);
}, sortByDistanceSorting: function(a, b) {return a.distanceToUser - b.distanceToUser;
}, sortByDistanceSortingDescending: function(a, b) {
```

```
return b.distanceToUser - a.distanceToUser;
} }; AR.context.onLocationChanged = World.locationChanged;
AR.context.onScreenClick = World.onScreenClick;
```

radar.js dosyası radar sembollerinin görüntü ayarlarının yapıldığı fonksiyonların bulunduğu javascript dosyasıdır.

```
var PoiRadar = {
hide: function hideFn() {
AR.radar.enabled = false;
}, show: function initFn() {
AR.radar.container = document.getElementById("radarContainer");
AR.radar.background = new AR.ImageResource("assets/radar_bg.png"); AR.radar.northIndicator.image
= new AR.ImageResource("assets/radar_north.png");
AR.radar.centerX = 0.5;
AR.radar.centerY = 0.5;
AR.radar.radius = 0.3;
AR.radar.northIndicator.radius = 0.0;
AR.radar.enabled = true;
}, updatePosition: function updatePositionFn() {
if (AR.radar.enabled) {
AR.radar.notifyUpdateRadarPosition();
} }, setMaxDistance: function setMaxDistanceFn(maxDistanceMeters) {
AR.radar.maxDistance = maxDistanceMeters;
} };
```