Java-Android Yazılım Mimarisi : Bir Masaüstü ile Çoklu Tablet Bilgisayar Haberleşme Uygulaması

Bilgehan Arslan[1], Süreyya Gülnar[2]

[1] Gazi Üniversitesi, Fen Bilimleri Enstitüsü, Ankara

[2] Gazi Üniversitesi, Bilişim Enstitüsü, Ankara

bilgehan.arslan@gazi.edu.tr, sureyya.gulnar@gazi.edu.tr

Özet: Java-Android yazılım mimarisi: bir masaüstü ile çoklu tablet bilgisayar haberleşme uygulaması başlığı altında yapılan çalışma; gelişen akıllı cihaz teknolojisine katkı sağlamak amacı ile Java-Android yazılım altyapısı kullanılarak geliştirilmiştir. Hızla gelişmekte olan tablet teknolojisi düşünülerek, tek bir masaüstü bilgisayar yardımı ile tabletlerle haberleşme sağlanmıştır. Bu sitemin tasarımında; temelde tek bir masaüstü bilgisayar üzerinden tablet bilgisayarlara anlık görüntünün resim formatında aktarımı hedeflenmiştir. Java-Android yazılım dili kod mimarisinin avantajları göz önünde bulundurularak hızlı haberleşme ve kayıpsız iletişim sağlanmıştır. Uygulamada; Server-Soket teknolojisi ve TCP-IP altyapısı kullanılmış ve uygulama geliştirilirken; Huffman Sıkıştırma algoritması sistem gereksinimleri için tekrar yorumlanmış ve sistemle bütünleştirilmiştir.

Anahtar Sözcükler: Server-Soket, Huffman, Java-Android Haberleşmesi, Anlık Ekran Görüntü Aktarımı

Abstract: Java android software architecture, according this study which a communication application with desktop computer and multiple tablet computer has developed for contributing smart device technology. Considering the rapidly emerging tablet technology every area in which is used tablet computer , communication each multiple tablet are provided with desktop computer. In the design of this system basically the target is transfering image which is named screenshoot from desktop computer to tablet computer. Considering the advatages of java android software language fast and loseless communication has provided in this study.Server socket technology and TCP-IP infrastructure has used this application while developing , and also Huffman compression algorthm for the system requirement have been re-interpreted and integrates into the system.

Keywords: Server-Socet Technology, Huffman, Communication of Java-Android, Screenshooting

1.Giriş

Günümüzde Java programlama dili oldukça fazla alanda, aktif olarak kullanılan nesne yönelimli bir programlama dilidir [1]. Java programlama dili ile veri tabanı erişimli uygulamalar, web tabanlı uygulamalar, akıllı telefon uygulamaları, akıllı kart uygulamaları gibi birçok uygulama geliştirmek mümkündür. Mobil cihazlar ve mobil cihazlarla iletişim ise hızla büyüyen ileri teknoloji uygulamalarındandır. Android mobil iletişim için geliştirilen yazılımlardan birisidir ve günümüzde yaygınlaşan mobil işletim sistemlerinin en önemlilerindendir. Android, linux çekirdeği üzerine inşa edilmiş bir mobil işletim sistemidir. Bu sistem ara katman yazılımı, kütüphaneler ve API C dilinden oluşur [2]. Uygulama yazılımları ise, Apache harmony üzerine kurulu java uyumlu kütüphaneler ihtiva eden uygulama iskeleti üzerinden çalışır [3].

Android, derlenmiş java kodunu çalıştırmak için dinamik çevirmeli (JIT) Dalvik sanal makinesini kullanır ve cihazların fonksiyonelliğini artıran uygulamaların geliştirilmesi için çalışan geniş bir programcı-geliştirici çevresine sahiptir [3]. Kullanım ve erişim kolaylığı çok fazla tercih edilme sebebidir.

Tablet teknolojisi, bilgisayarın masaüstü ile başlayıp dizüstü bilgisayar ile devam eden sürecinin geldiği en son noktadır [4]. Tablet bilgisayarlar klavye veya fareye ihtiyaç duymaksızın dokunmatik ekran ile kontrol edilebilir. Android mobil işletim sistemi günümüzün tablet bilgisayarlarında en çok kullanılan yazılımdır. Açık kaynak kodlu bir işletim sistemi olan Android, kısa zamanda büyük ilerleme kaydetmiş ve yoğun bir ilgi görerek yaygınlaşmıştır. Tablet teknolojisi birçok uygulama ve farklı birçok alanda karşımıza çıkmaktadır. Görüldüğü üzere yaşamımızda tablet bilgisayarlar önemli bir yer tutarken tablet bilgisayar ile masaüstü bilgisayarın haberleşmesi gerek eğitim, gerek ticaret, gerek taşımacılık olsun birçok alanda kullanılmaktadır. Bu çalışma ile hedeflenen ister sınıf ortamında olsun ister diğer ortamlar da tablet bilgisayarlar ile masaüstü bilgisayarın haberleşmesinin sağlanmasıdır.

Çalışmanın; ikinci bölümünde uygulama geliştirme süreci ve işleyiş adımları ile beraber sistem gereksinimleri hakkında bilgi verilmiştir. Üçüncü bölümde, yapılan uygulamadan bahsedilmiş, son bölümde ise sonuç ve öneriler sunulmuştur.

2. Uygulamanın Geliştirilme Süreci ve İşleyiş Adımları

Çalışmamızda temel amaç; masaüstü bilgisayarın ekran görüntüsünün, tablet bilgisayarlara aktarılmasıdır. Uygulama sırasında dikkat edilmesi gereken iki önemli unsur aşağıda belirtilmiştir.

1) Bilginin güncelliği için, alınan masaüstü bilgisayar ekran görüntüsü sık sık tekrarlanmalıdır ki gönderilen bilgi güncelliğini koruyabilsin.

2) Ekran görüntüsü gönderme süresi olabildiğince kısa tutulmalı ki tablet-masaüstü etkileşimi verimli bir şekilde sağlanabilsin.

Android altyapılı gerçekleştirilen uygulamanın çalışma mekanizması ise, aşağıdaki adımlar ile özetlenebilir.

Aşama1: Ekran görüntüsünü al.

Aşama2: Alınan görüntünün çözünürlüğünü azalt.

Aşama3: Ekran görüntüsünü, özel sıkıştırma algoritmaları sayesinde minimum kapasiteye getir.

Aşama4: Ekran görüntüsünü tabletlere aktar.

Aşama5: İstenilen durumlarda tabletlerde ki görüntüyü masaüstü bilgisayara aktar.

Eğer alınan ekran görüntüsünün kapladığı alan çok fazla ise veri yollama işlemi çok uzun sürebilir. Bu yüzden alınan ekran görüntüsü önce yeniden boyutlandırılır (resize) ve veri yollama işlemi sonra gerçekleştirilir. Böylece; veri yollama sürecinden tasarruf edilmiş olur.

Masaüstü bilgisayar görüntüsü, eş zamanlı olarak, tabletlere aktarılır. Bu sayede masaüstü bilgisayar ile tabletler arasında senkronizasyonlu görüntü alışverişi sağlanmış olur. Bu işlemler bir masaüstü ve tablet bilgisayarlar için çift taraflı olarak gerçekleştirilir.

Uygulamanın donanımsal alt yapısında ise; Java programlama dili ile hizmet sunan bir masaüstü bilgisayar ve android programlama dili tabanlı istemciler kullanılır. Windows tabanlı işletim sistemi maksimum 16 adet tablet bilgisayara izin

verdiğinden toplam 16 adet client mevcut olabilir.

Verinin türü ne olursa olsun, veri içerisinde depolama ve iletmeyi zorlayan öğeler barındırır. Veri sıkıştırma tekniklerindeki amaç; veriyi sıkıştırarak içerisindeki fazlalıkları atıp hacmini küçültmektir. Veri sıkıştırmada kullanılan yöntemler sıkıştırılacak olan verinin türüne göre seçilir. Minimum alana maksimum veri sıkıştırmak, daha hızlı aktarım sağlamak ve erişim süresini azaltmak gibi sebeplerden dolayı veri sıkıştırmaya ihtiyaç duyulur. Elias kodlaması, LZW sıkıştırma algoritması, delta sıkıştırması gibi birçok veri sıkıştırma metodu vardır. Çalışmamızda; sıkıştırma algoritması olarak, Huffman algoritması tercih edilmiştir. Huffman algoritması; bir veri kümesinde daha çok rastlanan sembolü daha düşük uzunluktaki kodla, daha az rastlanan sembolleri daha yüksek uzunluktaki kodlarla temsil etme mantığı üzerine kurulmuştur [5]. Bu mantıkla alınan ekran görüntüsüne ait her pikselin renk kodu bir veri biti dizisi oluşturup bu veri biti dizisi tekrarlanan değişkene dayanarak Huffman algoritması yardımı ile sıkıştırılır [6]. Uygulamada alınan ekran görüntüsünde aynı renk değerini tutan piksel sayısı tekrarı çok fazla olduğundan; bu çok tekrarlanan değer kayıpsız ve bozulmamış bir şekilde en iyi Huffman algoritması ile sıkıştırılabileceğinden uygulamamızda tercih edilmiştir.

Uygulamada soket programlama mantığı kullanılmıştır. Soketler aynı bilgisayardaki ya da farklı bilgisayarlardaki iki farklı sistemin haberleşmesini sağlayan sistemlerdir [7]. İstemci-sunucu mantığı ile çalışır. İstemci, veriyi isteyen soket programıdır. Sunucuya bağlanır, isteğini iletir, bağlantısını yapar. Sunucu ise veriyi veren soket programdır. İstemi alıp değerlendirir, gerekli yanıtı istemciye iletir. Sunucunun istemine herhangi bir anda yanıt verebiliyor durumda olması için devamlı çalışması gerekir. Soketler, internet bağlantınızın ve diğer bağlantıların arasında duran soyut birimlerdir.

Java programlama dilinin en önemli özelliklerinden biri de çoklu işlem (multithread) mimarisini destekliyor olmasıdır [9]. Bu mimari, bir bilgisayar programının komutlarının icrasının kontrolünü ve iyileştirilmesini sağladığı için bilgisayar bilimleri açısından çok güçlü ve hatırı sayılır bir gerçekleştirim olarak değerlendirilmelidir [9]. Çoklu işlem içinde çoklu veri iletimi ve uzaktan kontrol kavramlarını barındırır. Çalışmamızda, çoklu işlem yapısı bu yüzden tercih edilmiştir.

2.1.Sistem Gereksinimleri

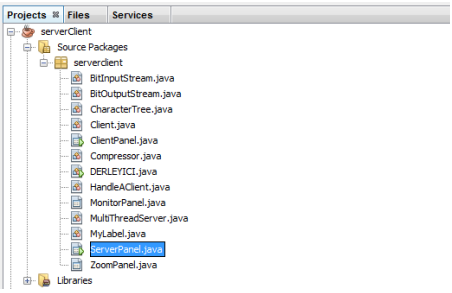
Projede kullanılan teknolojiler ve geliştirme ortamından bahsedecek olursak; uygulama Window 7 64 Bit işletim sistemine sahip bir bilgisayar üzerinde Eclipse ve Netbeans editörleri kullanılarak Java ve android programlama dilinde geliştirilir. Android sürümü olarak Android 2.3.3 işletim sistemi ve API 10 seviyesi kullanılır. Ayrıca bu uygulama Android 2.3.3 işletim sistemi ve üzeri bütün işletim sistemi sürümlerinde başarıyla çalışır. 2.3.3 sürümü öncesi sürümlerde uygulama çalışmamaktadır. Sistemde kullanılan tablet bilgisayarlar Android yazılım dili desteklidir.

3.Uygulama

Uygulamanın temeli tablet bilgisayarlar arasında anlık etkileşimli ekran görüntüsünü masaüstü bilgisayara iletme işlemine dayanır. İşlem sırasında; aktarılan görüntünün tazeliği ölçülmek istenmiştir. Farklı cihazlar (tablet ve bilgisayar), farklı programlama dilleri (java ve android) ve farklı derleyiciler (eclipse ve netbeans) veri iletim sürecine ne denli katkı sağladığını gözlemlemek amaçlanmıştır. Bu amaç doğrultusunda uygulama iki farklı şekilde gerçekleştirilmiştir.

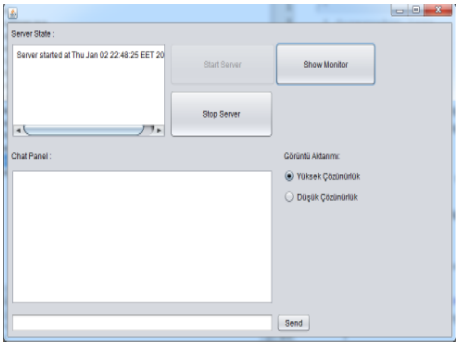
İlk olarak; iki bilgisayar arasındaki iletişim nasıl olabilir sorusu düşünülmüştür. Bu doğrultu da; uygulamaya; ilk önce ekran görüntüsü aktarma işlemi, netbeans kurulu bilgisayar ortamında alınan ekran görüntüsünün yine netbeans kurulu bir başka bilgisayar ortamına aktarılmasıyla başlanmıştır. Her iki ortamda eş cihaz yani masaüstü bilgisayar olduğunda aktarma hızının oldukça yüksek olduğu tespit edilmiştir.

Uygulamanın birinci kısmı; her iki ortamın masaüstü bilgisayara sahip olduğu varsayılarak incelenmiştir. Şekil 1 çalışmada kullanılan iki paneli gösterir. Bunlar ClientPanel (istemci paneli) ve ServerPanel (sunucu paneli) dir.



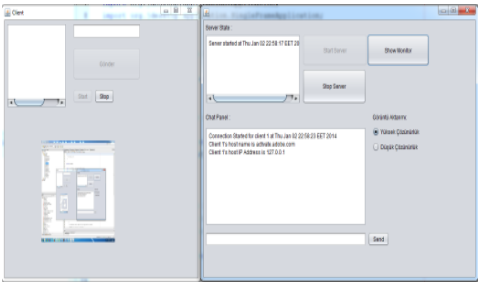
Şekil 1. ClientPanel ve ServerPanel

Şekil 2 ServerPanel'in çalışmasını gösterir ve bu durumda server (sunucu) çalışmış bulunmakta ve clientleri (istemcileri) dinlemeye hazırdır.



Şekil 2. ServelPanel.java sınıfının çalışır hali

Şekil 3 ClientPanel.java sınıfının çalışmasını gösterir. ClientPanel.java sınıfı çalıştırıldıktan sonra form üzerindeki start butonuna basıldığında server formunda cliente ait IPler ve isim bilgileri ile beraber bağlantının kurulduğu zaman bilgisi ekrana yazılmaktadır. Bu işlem gerçekleştikten yaklaşık 1 sn sonra client formuna bilgisayarın üzerindeki ekran görüntüsü düşmektedir.

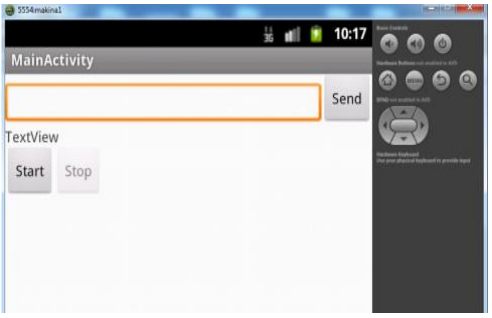


Şekil 3. ClientPanel.java sınıfının çalışması

Uygulamanın ikinci kısmı ise; bir masaüstü bilgisayar ve android işletim sistemine sahip tabletler arasında gerçekleşen ekran görüntü aktarımıdır. İşlem basamakları aşağıda belirtilmiştir.

1) Netbeans Kurulu bilgisayar üzerindeki ServerPanel formu çalıştırılır.

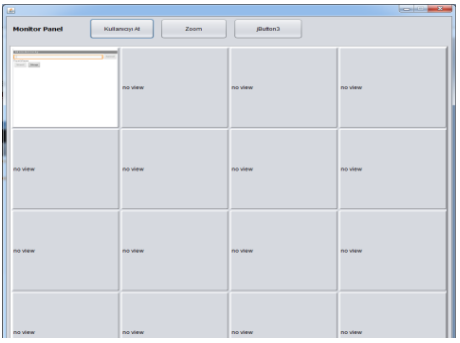
2) Server dinlemeye geçtikten sonra clientlerin bağlanma işlemi gerçekleşebilir. Bu yüzden öncelikle ServerPanel formu çalıştırılarak start butonu ile işleme başlanır.



Şekil 4. Eclipse-android ortamında client-server bağlantısı

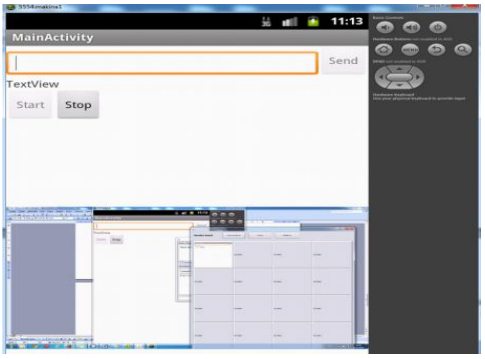
3) Şekil 4 eclipse - android ortamında client-server bağlantısını gösterir.

4) Tablet ile bilgisayar bağlantısı yani server ve client bağlantı kurulduktan sonra toplam 16 farklı tabletin ekran görüntüsünün aktarılacağı monitör paneli açılır. Aktarılan ekran görüntüsü kontrol edilebilir.



Şekil 5. Tabletlerin masaüstü üzerindeki ekran görüntüsü

5) Son olarak bilgisayar üzerindeki görüntü tabletlere aktarılarak iletişim çift taraflı olarak gerçekleştirilmiş olur.



Şekil 6. Bilgisayar ekranın tablete aktarımı

4.Sonuç ve Öneriler

Tüm açıklamalardan sonra görüldüğü üzere sistem bir masaüstü ve 16 tablet bilgisayar için başarılı bir şekilde ekran görüntüsü aktarımını gerçekleştirmiştir.

Uygulama sürecinde veri iletiminde maksimum hızı elde edebilmek amaçlı alınan ekran görüntüsü yeniden boyutlandırılarak kapladığı alan düşürülmüş Huffman algoritması ile sıkıştırılmıştır. Tüm bu eklemeler sonucunda hız değeri bakımından bir miktar artış elde edilmiş fakat yine de çok hızlı bir iletişimin sağlanmadığı gözlemlenmiştir. Bu durumun temel sebebinin TCP – IP altyapısı olduğu düşünülmektedir. Bu yapı ayrıntılı bir şekilde incelendiğinde ve bu yapının kullanımına uygun koşullar araştırıldığında karşımıza 3 ana sorun çıkar. Bunlar;

- TCP - IP kullanımı söz konusu olduğunda sistem birebir olarak işlevselidir; yani tek istemci tek sunuculu yapıya uygundur. Bu yüzden çoklu işlem söz konusu olduğunda sistem üzerinde aksamaların olması beklenilen sonuçtur. Bizim de uygulamamız da çoklu bir sistem desteği barındırdığından hız hususunda gecikmeler yaşanması normaldir.

- TCP – IP kullanıldığı sistemlerde dikkat edilmesi gereken diğer bir hususta kullanılan platformun aynı olması koşuludur. Fakat kullandığımız sistem üzerinde Windows işletim sistemi kurulu bir ana bilgisayar ve android yazılımı ile çalışan tablet bilgisayarlar mevcut olduğundan bu durumda da aksamaların olması muhtemeldir. Bu aksamalar veri iletim hızı üzerinde olumsuz sonuçlar yaratmaktadır.

- Son olarak TCP –IP tabanlı geliştirilen sistemlerde farklı geliştirme ortamları kullanılması durumda etkileşim hızında düşüş görüldüğü tespit edilmiştir.

Çalışmada; eclipse ve netbeans adlı iki farklı editör kullanılması veri iletiminde hız kaybına yol açmaktadır. Uygulamanın 16 bilgisayar ile sınırlı kalmasının sebebi ise işletim sistemi yazılımının 16'lı paylaşımdan daha fazlasına izin vermemesidir.

Sonuç olarak; yapılan uygulama da bir takım eksiklikler bulunsa da; temelde amacına ulaşmıştır. Bir bilgisayar ile 16 tane tablet bilgisayar arasında sorunsuz ekran görüntüsü aktarımı sağlanmıştır. Masaüstü bilgisayar - masaüstü bilgisayar haberleşmesinin, masaüstü bilgisayar-tablet haberleşmesinden daha hızlı sonuç verdiği gözlemlenmiştir. Gelecekte ise bu çalışma daha büyük platformlar için geliştirilebileceği gibi ekran görüntüsünün yanında ileri teknoloji uygulamaları kullanılarak hareketli görüntü akarımı da yapılabilir.

5. Kaynaklar

[1] Eckel B., "Thinking in Java", **Newjersey,** 860, (1998).

[2] Butler M., "Android: Changing the Mobile Landscape", **Pervasive Computing**, 10, 4–7, (2010).

[3] Jing Y., Ahn G-J., Hu H., "Model-based conformance testing for Android", **Proceedings of the 7th International Workshop on Security (IWSEC**), 118 (2012).

[4] Gillett F., "Why Tablets Will Become Our Primary Computing Device", **Forrester.com,** Erişim Tarihi: Mayıs 2012.

[5] Gallager, R.G. "Variations on a theme by Huffman", **IEEE Trans**., IT-24, 668-674 , (1978).

[6] Bodur T., "Durağan Görüntü Sıkıştırma Yöntemleri", **İstanbul Teknik Üniversitesi-Fen Bilimleri Enstitüsü,** Yüksek Lisans Tezi, (1995).

[7] Karasu M.,"Development techniques of socket programming" **Marmara Üniversitesi - Fen Bilimleri Enstitüsü**, Yüksek Lisans Tezi (2002).

[8] Cristiano, J.J., Liker, J.K. and White, C.C., "Customer-driven product development through quality function deployment in the US and Japan", **Journal of Product Innovation Management,** 17(4):228-308 (2000).

[9] Çöpoğlu S., Dokumacı M,, "Java Programlama Dili", Lisans Tezi, **Fırat Üniversitesi Teknik Eğitim Fakültesi, Elazığ**, (2000).