

WEB-TABANLI GERÇEK- ZAMANLI İLETİŞİM TEKNOLOJİSİ İLE SANAL SINIF UYGULAMASI

Okt. Uğur Yıldız
Kocaeli Üniversitesi
uguryildiz@kocaeli.edu.tr

Prof. Dr. M. Melih İnal
Kocaeli Üniversitesi
minal@kocaeli.edu.tr

Öğr. Gör. Serdar Solak
Kocaeli Üniversitesi
serdars@kocaeli.edu.tr

Yrd. Doç. Dr. Umut Altınışık
Kocaeli Üniversitesi
umuta@kocaeli.edu.tr

Özet

Sanal sınıf uygulamaları e-öğrenme platformlarında eğitimci ve öğrenciler arasında ağ üzerinden, çevrim içi etkileşimin gerçekleştirildiği gelişmiş araçlar olarak tanımlanmaktadır. Sanal sınıf uygulaması olarak, e-öğrenme platformlarında genellikle ticari veya açık kaynak kodlu üçüncü parti yazılımlar kullanılmaktadır. Yapılan çalışmada, geleneksel sanal sınıf uygulamalarına alternatif, yeni bir yaklaşım olarak örgü tipi ağ topolojisini kullanılan Web-Tabanlı Gerçek-Zamanlı İletişim (WebRTC) teknolojisinin bu alandaki kullanım olanakları incelenmektedir. Ayrıca çalışmada, WebRTC teknolojisi kullanılarak geliştirilen etkileşimli bir sanal sınıf prototip uygulaması gerçekleştirilmiştir. Sonuç olarak gerçekleştirilen prototipte, e-öğrenme platformları için çok kullanıcı bir video konferans deneyimi alternatif olarak sunulmaktadır.

Anahtar Sözcükler: WebRTC, Canlı Ders, Örgü tipi ağ topolojisi, E-öğrenme.

VIRTUAL CLASS APPLICATION WITH TECHNOLOGY OF WEB-BASED REAL-TIME COMMUNICATION

Abstract

In e-learning platform, virtual classroom applications which are performed over the network between instructors and students with online interaction are described as advanced tools. Commercial or open-source third-party software as an application of virtual class is usually used in e-learning platform. In this study, a new approach as the mesh-type network topology in the Web-Based Real-Time Communication (WebRTC) technology are investigated for the usage possibilities of being an alternative to traditional virtual classroom application. Also, a prototype application of interactive virtual classroom which is developed by using WebRTC technology is implemented. As a result, experiences of multi-user video conferencing system with the implemented prototype are presented as an alternative for the e-learning platforms.

Keywords: WebRTC, Online Course, Mesh type network topology, E-learning.

GİRİŞ

Web tabanlı uygulamaların esnek ve kolay erişilebilir olması nedeniyle eğitim'de de alternatif bir platform olarak internetin kullanılması kabul görmeye başlamıştır. Üniversiteler ve çeşitli eğitim kurumları geliştirdikleri eğitim içeriklerinin sunumunu ve uygulayacakları öğrenme metodlarını kolay ve hızlıca gerçekleştirebilecekleri araçlar sunan öğrenme yönetim sistemlerini ve bu sistemlerle uyumlu çeşitli teknolojileri kullanarak e-öğrenme ortamlarını oluşturmaktadırlar. Sanal sınıf uygulamaları da eğitmen ve öğrencilerin internet üzerinden gerçek zamanlı olarak yüz yüze öğrenme yönteminin kullanılabilirdiği, özellikle uzaktan eğitim'de günümüz e-öğrenme ortamlarının vazgeçilmez bileşenlerinden birisi olmuştur.

Sanal sınıflar gerçek yüz yüze eğitimlerdeki gibi katılımcı etkileşiminin sağlanmasını hedefleyen video konferans, beyaz tahta, ekran paylaşımı, sunum ve mesajlaşma araçları gibi çeşitli araçlar sunmaktadır. Günümüzde sanal sınıflarda etkileşimi sağlayan video konferans, beyaz tahta, ekran paylaşımı, sunum ve mesajlaşma gibi araçlar farklı teknolojilerin eş zamanlı olarak kullanılmasıyla geliştirilmektedir. Bununla beraber internet teknolojilerine farklı tipteki cihazlardan erişim imkanları çoğaldıkça öğrenciler ve eğitimcilerin de günümüz sanal sınıflarına standart platformlar yerine her yerden, her zaman, aynı kalitede erişim sağlama talepleri giderek artmaktadır. Günümüzde sanal sınıf uygulamalarındaki hem işletim sistemi hemde istemci bazlı farklılıklardan doğan teknolojilerin eşzaman kullanımına dair sorunlar her tip istemci için özel eklentiler geliştirilerek çözümlenmeye çalışılmaktadır.

WebRTC üçüncü parti yazılım veya eklentilere ihtiyaç duymadan, web tarayıcılar arasında sadece HTML5 ve Javascript dilleri kullanılarak video, ses ve veri aktarım uygulamaları geliştirilmesine imkan sağlayan yeni bir açık geliştirme çatısıdır. (Sergiienko, 2014) Bu çalışmada yeni nesil web tarayıcılarının desteklediği WebRTC teknolojisinin, e-öğrenme ortamlarında gerçek yüz yüze eğitimlerdeki gibi katılımcı etkileşiminin sağlanmasını hedefleyen video konferans, beyaz tahta, ekran paylaşımı, sunum ve mesajlaşma gibi araçları barındıran sanal sınıf çözümlerinin geliştirilmesi için kullanımına dair yeni bir yaklaşım sunulmaktadır.

Mevcut Sanal Sınıf Uygulamaları

Günümüzde yaygın kullanılan sanal sınıf uygulamalarının temel bileşeni web tabanlı video konferans özelliğidir. Video konferans uygulamaları istemci ve sunucu tabanlı olmak üzere iki bölümden oluşmaktadır. İstemci tarafında genellikle Adobe Flash veya Java tabanlı bir web tarayıcı uygulaması çalışmaktadır. Sunucu tarafında ise çeşitli programlama dilleri ile geliştirilen Adobe Flash Media Server, Red5 Media Server, Wowza Streaming Engine gibi özel medya sunucu uygulamaları kullanılmaktadır. Bu medya sunucu uygulamaları video ve ses kaynaklarının isteğe bağlı veya canlı akışlarının yayınlaması, istemci akışlarının kaydedilmesi, istemci arasında paylaşılan nesnel oluşturulup yönetilebilmesini, veri aktarım işlemlerinin yönetimini sağlamakta ve bu özelliklere bağlı çeşitli gerçek zamanlı iletişim protokollerini (RTMP, RTMPT, RTMPS ve RTMPE) desteklemektedirler.

Yukarıda bahsedilen mimari çerçevesinde geliştirilmiş, e-öğrenme ortamlarında yaygın olarak kullanılan sanal sınıf uygulamaları ve genel özellikleri şöyledir:

BigBlueButton

BigBlueButton'un ilk sürümü 2007 yılında Carleton Üniversitesi "Technology Innovation Management" programı bünyesinde Richard Alam tarafından Genel Kamu Lisansı (GPL) ile yazılmıştır. (BigBlueButton, 2016) BigBlueButton temelde Java tabanlı Red5 Media Server uygulama sunucusu üzerinde geliştirilmiştir. Sunucu tarafında Red5 Media Server ile birlikte açık kaynak kodlu birçok farklı yazılım BigBlueButton alt bileşeni olarak çalışmaktadır. İstemci tarafında ise HTML5 ve Java tabanlı uygulamalar kullanılmaktadır.

Adobe Connect

Adobe firması tarafından webinar, web-görüşmesi ve e-Öğrenme ortamları için geliştirilen ticari lisanslı popüler bir web konferans platformudur. (Adobe Connect, 2016) Sunucu uygulamaları ActionScript dili ile geliştirilebilmektedir. İstemci tarafında da yine uygulama geliştirmek için ActionScript dili kullanılmaktadır.

İstemci tarafında geliştirilen uygulamalar web tarayıcısı üzerinde Adobe Flash Player eklentisi ile çalıştırılabilmektedir.

Diğer Uygulamalar

Adobe Connect ve BigBlueButton dışında e-öğrenme ortamlarında kullanılan web konferans temelli birçok uygulama bulunmaktadır. Bu uygulamaların en çok bilinenlerinden WebEx, GotoMeeting, Visimeet gibi ortak çalışma imkanı sunan web konferans sistemleri genellikle son kullanıcı bilgisayarlarına kurulan özel yazılımlar marifetiyle çalışmaktadırlar. Bunların dışında Skype ve Hangout gibi son kullanıcıların web konferans için yaygın olarak kullandıkları uygulamalar, e-öğrenme ortamlarında ve kurumsal uygulamalarda da sanal sınıf amaçlı kullanılabilmektedir.

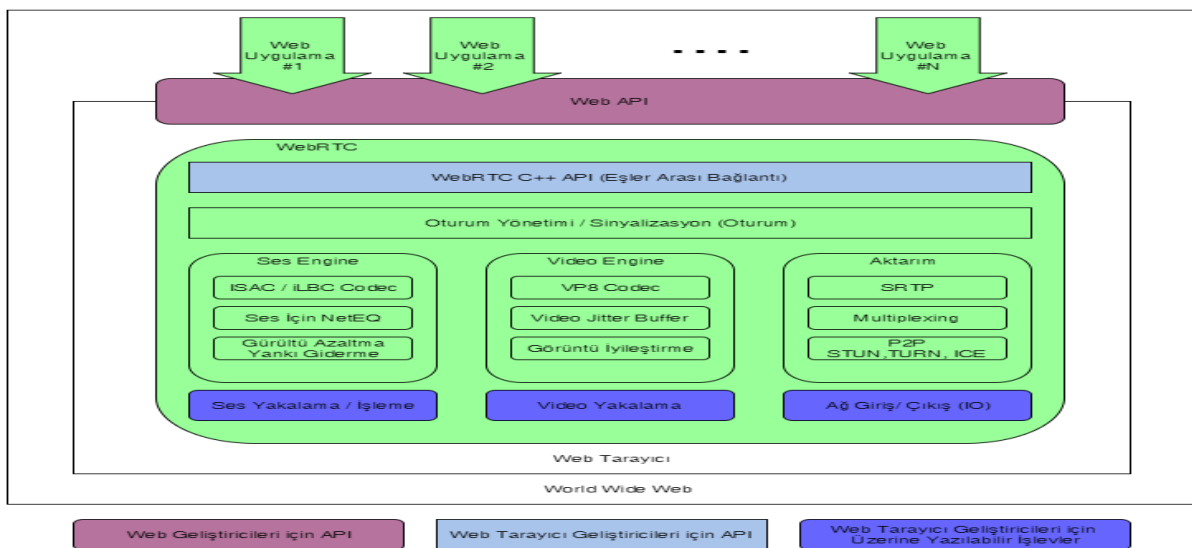
Yukarı bahsi geçen uygulamaların bazıları platformlarını WebRTC teknolojisi ile bütünleşme çalışmalarını başlatmışlardır. BigBlueButton özellikle istemci ses akışlarının kalitesini arttırmak için WebRTC teknolojisini kullanmakta ve sunum yapan kullanıcının ekran paylaşımının WebRTC ile gerçekleştirilebilmesi için çalışmalar devam etmektedir. (BigBlueButton – LABS, 2016) Bununla beraber Skype uygulaması da WebRTC uyumluluğu sağlama çalışmaları ile birlikte farklı video ve ses codec'leri kullanılan WebRTC benzeri teknolojiler üzerinde çalışmalar gerçekleştirmektedir (Office Blog, 2015).

WebRTC Teknolojisi

WebRTC, World Wide Web Consortium (W3C) tarafından hazırlanan web tarayıcı'dan web tarayıcı'ya dahili veya harici eklenti gerekmeden video konferans, ses konferans ve veri/dosya paylaşım uygulamalarının geliştirilmesine imkan sağlayan bir API'dir. (Sergiienko, 2014) WebRTC teknolojisinin sağladığı 3 ana işlev şunlardır;

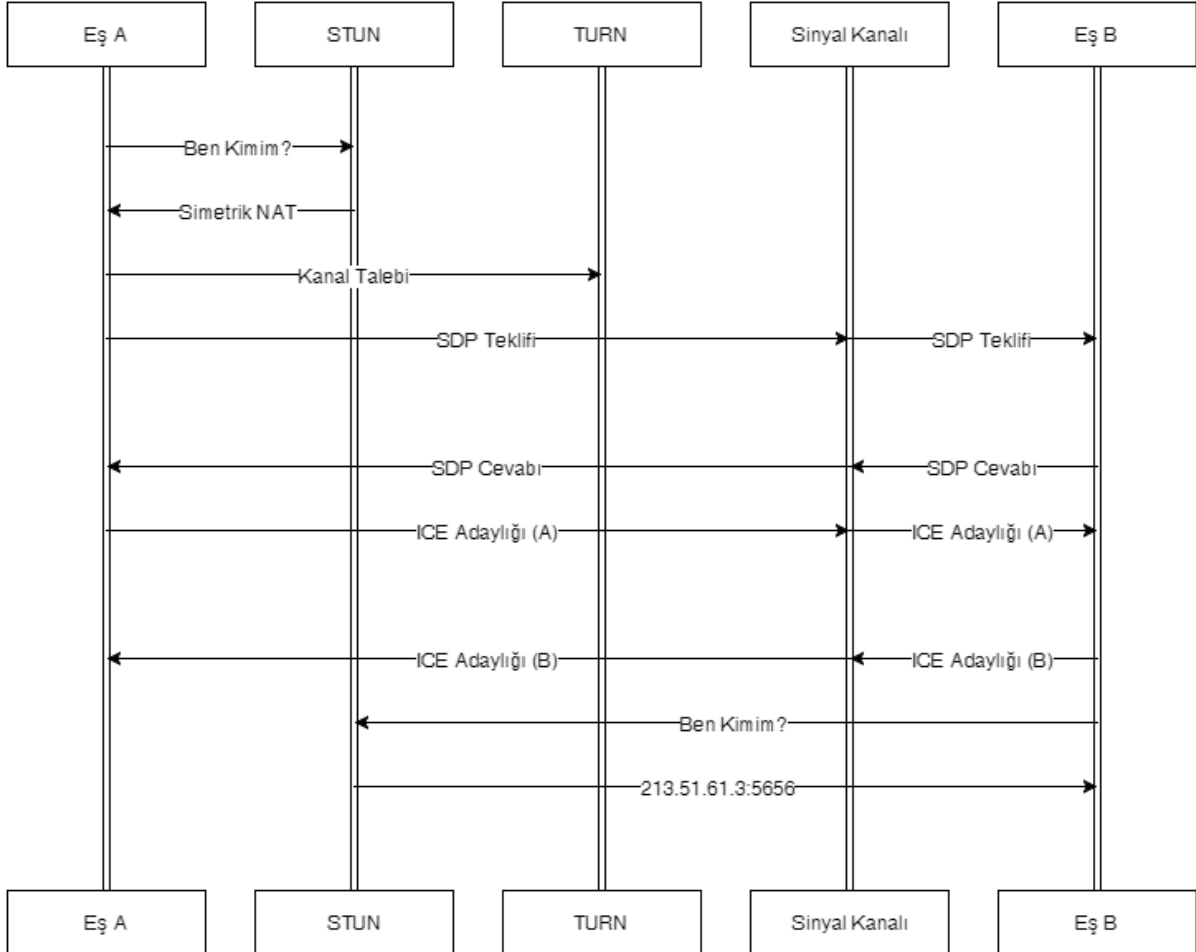
- **Ortam Akış Kaynakları Yönetimi:** Web tarayıcıların cihazda bağlı olan video ve ses kaynaklarına erişimi sağlar.
- **Veri/Dosya Aktarımı ve Paylaşımı:** Web tarayıcıların eşler arası (Peer-To-Peer) bağlantı ile oturuma katılan istemciler arasında veri/dosya aktarımı ve paylaşımı sağlar.
- **Ses / Video Aktarımı:** İstemciler arasında eşler arası bağlantı ile ses ve video görüşmeleri gerçekleştirilmesi sağlar.

WebRTC teknolojisi, kişisel bilgisayarlarda çalışan Mozilla Firefox, Google Chrome, Opera (18) ve Microsoft Edge uygulamalar tarafından, Android, iOS ve Blackberry gibi mobil işletim sistemlerindeki yerleşik web tarayıcıları tarafından ve Chrome OS, Firefox OS gibi yeni nesil işletim sistemleri tarafından desteklenmektedir. (Wikipedia, 2016) WebRTC teknolojisinin genel mimarisi Şekil 1'de görülmektedir.



Şekil 1: WebRTC Mimarisi (WebRTC.org, 2016)

WebRTC API ile gerçekleştirilecek eşler arası ağ bağlantısı öncesinde bağlantı ile ilgili bilgiler web sunucu üzerinden katılımcı diğer tüm istemciler ile değiş tokuş edilmelidir. Bu iletişim SDP (Session Description Protocol) ile gerçekleştirilir. Bu değişim başarılı bir şekilde gerçekleşirse eşler birbirlerini görebilirler. WebRTC API'de eşler arasındaki ağ iletişimi Şekil 2'de görülen Interactive Connectivity Establishment (ICE) tekniği kullanılarak gerçekleştirilmektedir.



Şekil 2: WebRTC ICE Diyarımı (Mozilla Developer Network, 2016)

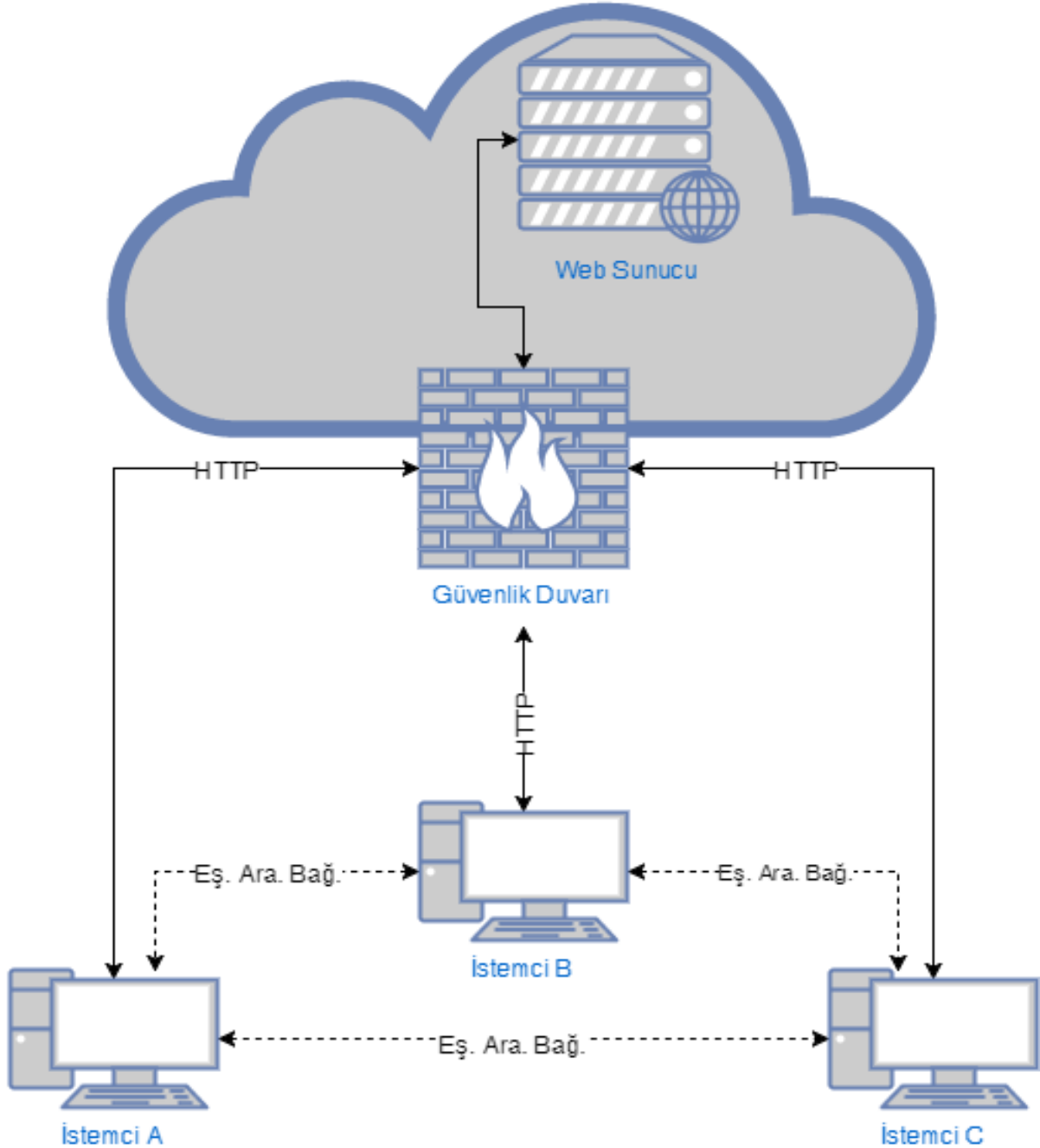
ICE'de ilk olarak işletim sisteminin veya ağ kartının sağladığı ağ adresi ile ağ iletişimi gerçekleştirilmeye çalışılır; eğer ağ adresi sağlanamazsa (NAT arkasındaki cihazlar için) STUN sunucusu kullanılarak harici bir ağ adresi alınabilir, bu da başarısız olursa TURN sunucusu kullanılarak trafik yönlendirilir. WebRTC teknolojisi ile eşler arasında çoklu bağlantı gerçekleştirilebilir. Her bir uç nokta diğer uç noktaların her biri ile tam örgü ağ topolojisi ile bir ağ yapılandırması ve iletişimi kurabilir. Bunun yanı sıra bir uç nokta diğer uç noktalara kendi çoklu ortam kaynaklarının akışlarını yıldız topolojisi ile de aktarabilir. WebRTC'nin sunucu yükünü ve maliyetleri azaltan bu ağ topolojilerine uyumluluğu; sanal sınıf uygulamalarının geliştirilebilmesi için donanım ve kaynak maliyeti daha çok olan sunucu temelli mimarilerin yerine gelecekte kullanılmasının önünü açan için bir tercih nedeni olarak öne çıkmaktadır.

Sanal Sınıf Prototip Uygulama Yapısı

Prototip uygulama temelde web uygulaması ve sinyalizasyon mekanizması olarak 2 bölümden oluşmaktadır. Eğitimciler geliştirilen web uygulaması ile bir sanal sınıf oluşturarak katılımcıları bu sanal sınıfa kaydedebilirler. Bu kayıt işlemi sistemdeki mevcut kullanıcıların oluşturulan sanal sınıf oturumuna eklenmeleri ile gerçekleştirilebilmektedir. Eğer katılacak kullanıcılar sisteme kayıtlı değilse eğitimci oluşturulacak CSV tipindeki

bir liste ile sisteme kullanıcı bilgilerini yükleyebilir ve kayıt olmaları için katılımcıların e-posta adreslerine davetiye gönderebilir.

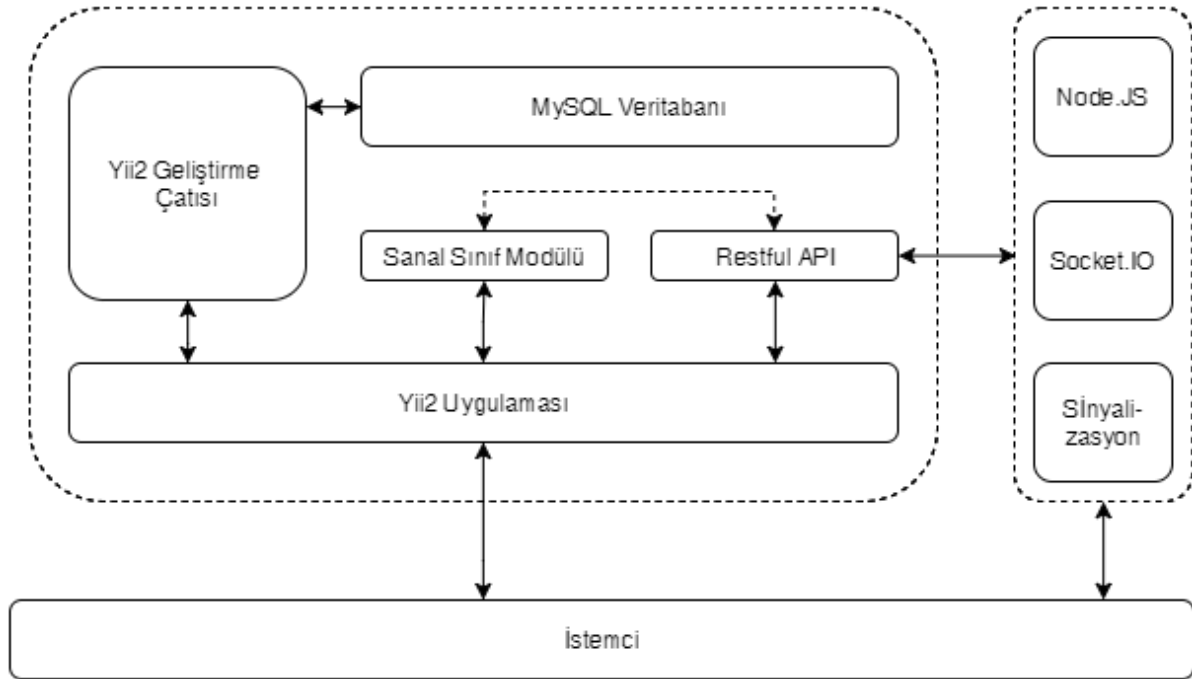
Sistem Şekil 3'de görüldüğü gibi tam örgü tipi ağ topolojisini kullanmaktadır. Bu tasarımda tüm katılımcılar şeffaf bir sinyalizasyon sunucusu kullanarak birbirleriyle bağlantı kurarlar. Her bir katılımcı web sunucusuna bağlanır ve sanal sınıf ile ilgili uygulama sayfasına giriş yaparak gerekli WebRTC Javascript API kodlarını indirir. Bu senaryo 3 katılımcı için Şekil 3'de açıklanmıştır.



Şekil 3: Örgü Tipi Ağ Topolojisi ile WebRTC Eşler Arası Bağlantı Mimarisi

Web Uygulaması

Web uygulaması Yönetim Paneli, RESTful API ve Sanal Sınıf Odaları olmak üzere 3 temel bölümden oluşmaktadır. Web uygulaması PHP dili ile Yii2 geliştirme çatısı kullanılarak yazılmıştır. Yönetim işlemleri standart web kullanıcı arayüzleri ile gerçekleştirilmektedir. Sanal Sınıf Odaları sistemdeki sinyalizasyon mekanizması ile ilgili gerekli tüm API ve betikleri içermektedir. Bu odalar sinyalizasyon mekanizmasındaki bileşenlerle web uygulamasının sağladığı RESTful API üzerinden AJAX metoduyla haberleşmektedirler. Sistem üzerindeki tüm veriler MySQL Veritabanı sunucusunda tutulmaktadır. Sistemdeki tüm bileşenlerin birbirleriyle olan etkileşimleri Şekil 4'de görülmektedir.

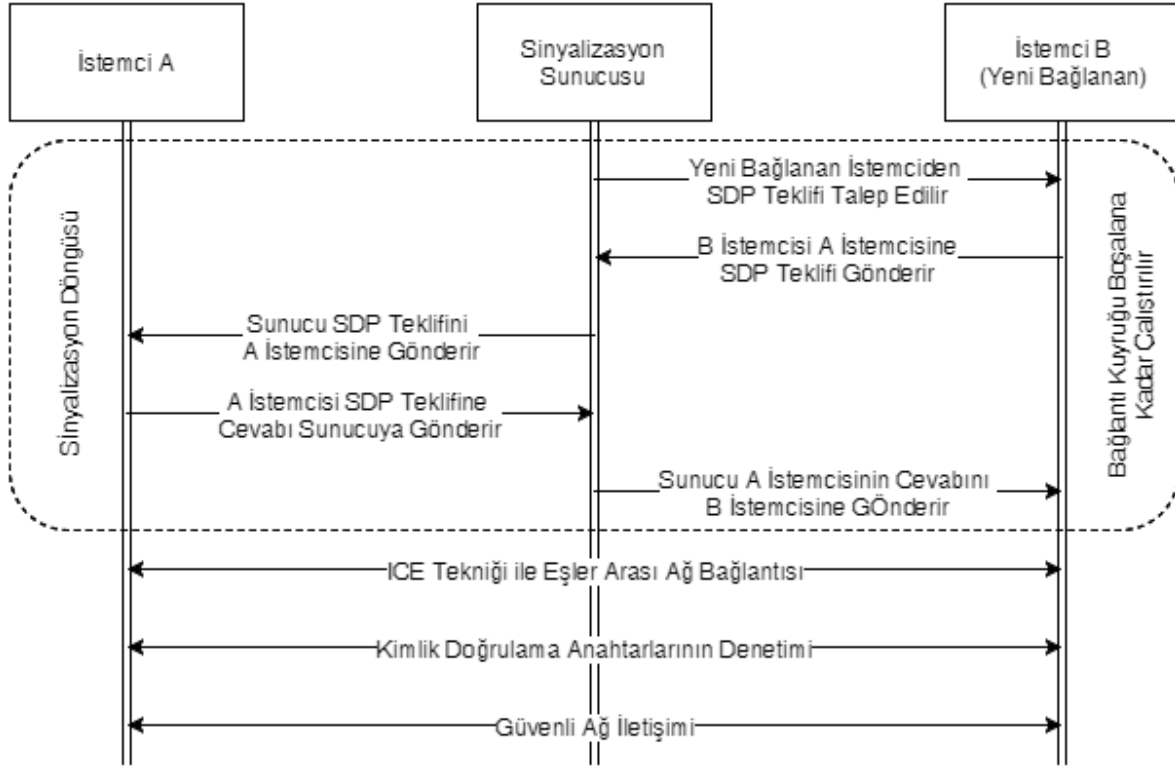


Şekil 4: Web Uygulaması

Sinyalizasyon Mekanizması

SDP temelli bağlantı verilerinin değiş tokuşuna dair sinyalizasyon işlemi her katılımcının web tarayıcısının sorumluluğundadır. Sinyalizasyon WebRTC tarafından bir standart olarak hazırlanmamıştır. Bu işlemler için üçüncü parti uygulamalar kullanılmaktadır. Bu prototiple Node.JS ve Socket.io uygulamaları sinyalizasyon sunucusu olarak kullanılmaktadır. Node.JS Javascript betik dili ile programlanabilen bir sunucu platformudur. (Node.JS, 2016) Node.JS ile bir ağ sunucu uygulaması geliştirilebilir. Socket.io ise gerçek-zamanlı sistemlerde kullanılabilen WebSocket API'dir. Socket.io, Node.JS ile entegre olarak çalışabilmektedir. Socket.io API ile Node.JS sunucusu üzerinden, sunucu ve web tarayıcılar arasında sağlıklı TCP bağlantıları kurulabilmektedir (Walsh, 2010).

Sinyalizasyon mekanizması tarayıcılar arasında bağlantı işlemini başlatmak için sunucu üzerinde tutulan sanal sınıf oda katılımcı kuyruğunu belirli aralıklarla kontrol eder. Yeni bir katılımcı bağlandığında kullanıcı için sinyalizasyon sunucusundan bir socket kimliği (socket ID) alınır ve bağlantı için uygun olan katılımcı kuyruğuna kaydedilir. Daha sonra yeni bağlanan kullanıcıdan SDP teklifi (offer) talebi ile sinyalizasyon işlemi başlatılır. Kullanıcı kendisi ile paylaşılan sanal sınıf oda katılımcı listesindeki ilk uygun kullanıcı için geriye SDP teklifini sinyalizasyon sunucusuna gönderir. SDP teklifi alan kullanıcı SDP cevabı (answer) üretir ve teklif socket ID'si saklanan alıcı ID alan bilgisiyle birlikte sinyalizasyon sunucusuna gönderir. SDP cevabını alan sinyalizasyon sunucusu, cevabı socket alıcı ID'si ile eşleşen web tarayıcısı ile paylaşır.



Şekil 5: Sinyalizasyon Süreci Genel Akışı

SDP protokolü ile gerçekleşen bağlantı bilgilerinin değiş tokuş işleminden sonra, sinyalizasyon sunucusu web tarayıcı bağlantılarında şeffaf hale gelir. Ardından tarayıcı, kimlik doğrulama anahtarlarının denetimini gerçekleştirmek ve güvenli bir ortam oluşturmak için ICE tekniği ile ağ iletişimini başlatır. Sinyalizasyon süreci döngüsü bağlantı kuyruğu boşalana kadar devam eder. Şekil 5'de sinyalizasyon işlemlerinin genel akışı görülmektedir.

SONUÇ VE ÖNERİLER

Gerçekleştirilen prototiple beraber WebRTC teknolojisinin e-öğrenme ortamlarında kullanılabilecek çok kullanıcı alternatif bir video konferans sistemi geliştirmek için uygun olduğu gözlemlenmiştir. WebRTC teknolojisi ile geliştirilen bu prototipte özellikle bir medya sunucusuna ihtiyaç duyulmaması maliyetlerin büyük oranda düşmesini sağlamaktadır. Maliyetlerin düşük olmasına rağmen bağlantının hızlı, ses gecikmelerinin düşük ve ses kalitenin hissedilir oranda artması da artı bir avantaj olarak öne çıkmaktadır. WebRTC web tarayıcılar arasındaki iletişimin güvenliğini web tarayıcı seviyesinde gerçekleştirdiği için iletişim süresini hızlandırmaktadır (Dutton, 2012).

WebRTC'nin en büyük dezavantajlarından biri ise halen kısıtlı platformlarda destekleniyor olmasıdır. Buna WebRTC API sürümünün 1.0 olmasına rağmen "World Wide Web Consortium" (W3C) ekibi tarafından halen "Taslak Çalışma" olarak başkalandırılması da eklenebilir. (World Wide Web Consortium, 2016) Teknolojinin standartları oluşurken majör değişikliklere gidilebilir olma ihtimali sektördeki büyük teknoloji firmalarının bu teknolojiyi yaygınlaştırmasını yavaşlatmaktadır.

CPU işlem performansları ve yüksek bant genişliği kullanım ihtiyacı da başka bir dezavantaj olarak karşımıza çıkmaktadır. Diğer katılımcılarla paylaşılması istenen çoklu ortam kaynaklarının kodlaması'nın (encoding) birçok kez paralel olarak yapılması, katılımcı işlemcilerinin yükünü arttırmaktadır. Bu dezavantajlarıyla beraber yankı ve gürültü iptali için halen geliştirmelere ihtiyaç duyulduğu da gözlemlenmiştir.

Bu çalışmada ayrıca WebRTC, Node.JS ve socket.io uygulamalarının birlikte kullanılarak web tabalı gerçek zamanlı uygulamaların kolay ve hızlıca gerçekleştirilebildiği gözlemlenmiştir. WebRTC halen geliştirilen bir teknoloji olmasına ve sektörde lider olan web tarayıcılar tarafından tam olarak desteklenmemesine rağmen geleneksel sanal sınıf uygulamalarına alternatif olabilecek uygulamalar geliştirmek için yeterli düzeye eriştiği söylenebilir. Bununla beraber eğer bu teknolojinin kullanımı yaygınlaşırsa e-öğrenme ortamlarını oluşturan kurumların ağ ve altyapı maliyetlerinin düşmesinde de büyük rol oynayabilir.

KAYNAKÇA

Sergiienko, A. (2014). WebRTC Blueprints. Packt Publishing.

BigBlueButton, (2016). History of BigBlueButton. 13.04.2016 tarihinde <http://bigbluebutton.org/history> adresinden alınmıştır.

Adobe Connect, (2016). Adobe Web Conferencing Software. 13.04.2016 tarihinde <http://www.adobe.com/products/adobeconnect.html> adresinden alınmıştır.

Node.JS, (2016). About Node.JS. 13.04.2016 tarihinde <https://nodejs.org/en/about> adresinden alınmıştır.

David Walsh, (2010). WebSocket and Socket.IO. 13.04.2016 tarihinde <http://davidwalsh.name/websocket> adresinden alınmıştır.

Mozilla Developer Network, (2016). WebRTC Connectivity. 13.04.2016 tarihinde https://developer.mozilla.org/en-US/docs/Web/API/WebRTC_API/Connectivity adresinden alınmıştır.

WebRTC.org, (2016). WebRTC Architecture. 13.04.2016 tarihinde <https://webrtc.org/architecture/> adresinden alınmıştır.

BigBlueButton - LABS, (2016). HTML5 Overview. 13.04.2016 tarihinde <http://docs.bigbluebutton.org/labs/html5-overview.html> adresinden alınmıştır.

Office Blog, (2015). Bringing Interoperable Real-Time Communications to the Web. 13.04.2016 tarihinde <https://blogs.office.com/2015/09/18/enabling-seamless-communication-experiences-for-the-web-with-skype-skype-for-business-and-microsoft-edge/> adresinden alınmıştır.

Wikipedia, (2016). WebRTC. 13.04.2016 tarihinde <https://en.wikipedia.org/wiki/WebRTC> adresinden alınmıştır.

Sam Dutton, (2012). Getting Started with WebRTC. 13.04.2016 tarihinde <http://www.html5rocks.com/en/tutorials/webrtc/basics/#toc-security> adresinden alınmıştır.

World Wide Web Consortium [W3C], (2016). WebRTC 1.0: Real-time Communication Between Browsers. 13.04.2016 tarihinde <https://www.w3.org/TR/webrtc/> adresinden alınmıştır.