



MARMARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



EĞİTİM AMAÇLI ELEKTRİK PİYASASI SİMÜLATÖRÜNÜN GELİŞTİRİLMESİ

OĞUZ ALTUNTAŞ

YÜKSEK LİSANSTEZİ

Elektrik – Elektronik Mühendisliği Anabilim Dalı

Elektrik - Elektronik Mühendisliği Programı

DANIŞMAN

Prof. Dr. BülentORAL

İSTANBUL 2019



MARMARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



EĞİTİM AMAÇLI ELEKTRİK PİYASASI SİMÜLATÖRÜNÜN GELİŞTİRİLMESİ

OĞUZ ALTUNTAŞ

523114013

YÜKSEK LİSANSTEZİ

Elektrik – Elektronik Mühendisliği Anabilim Dalı

Elektrik-Elektronik Mühendisliği Programı

DANIŞMAN

Prof. Dr. Bülent ORAL

İSTANBUL 2019



MARMARA UNIVERSITY
INSTITUTE FOR GRADUATE STUDIES
IN PURE AND APPLIED SCIENCES



DEVELOPMENT OF AN ELECTRICITY MARKET SIMULATOR FOR TRAINING

OĞUZ ALTUNTAŞ

MASTER THESIS

Department of Electricity-Electronic Engineering

Thesis Supervisor

Prof. Dr. Bülent ORAL

İSTANBUL, 2019

MARMARA ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yüksek Lisans Öğuz ALTUNTAŞ'ın
"Eğitim Amaçlı Elektrik Piyasası Simülatörünün Geliştirilmesi" başlıklı tez
çalışması, 27 Haziran 2019 tarihinde savunulmuş ve jüri üyeleri tarafından başarılı
bulunmuştur.

Jüri Üyeleri

Prof.Dr. Bülent ORAL (Danışman)

Marmara Üniversitesi (İMZA)

Doç.Dr. Ümit Kemalettin TERZİ (Üye)

Marmara Üniversitesi (İMZA)

Dr. Öğr. Üyesi Süreyya KOCABEY (Üye)

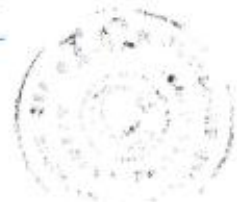
Sağlık Bilimleri Üniversitesi (İMZA)

ONAY

Marmara Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 12/7/2019 tarih ve
2019/14-23 sayılı kararı ile Öğuz ALTUNTAŞ'ın Elektrik-Elektronik Mühendisliği
Anabilim Dalı Elektrik-Elektronik Mühendisliği Programında Yüksek Lisans derecesi
alması onanmıştır.

Fen Bilimleri Enstitüsü Müdürü

Prof. Dr. Bülent EKİCİ



TEŐEKKÖR

Türk elektrik piyasası simölatör fikrinin eyleme dönüőmesi sürecini koordine eden, entelektöel yapısı ve bilgisiyle harman olmuş tecrübelerini güzel bir üslupla bana aktaran kıymetli tez danışmanım ve değeri hocam Prof. Dr. Bülent ORAL'a teşekkürü bir borç bilirim.

Haziran 2019

Oğuz ALTUNTAŐ

İÇİNDEKİLER

ÖZET.....	iii
ABSTRACT.....	iv
SEMBOLLER	v
KISALTMALAR	ix
ŞEKİL LİSTESİ.....	x
TABLO LİSTESİ.....	xiii
1. GİRİŞ VE AMAÇ.....	1
1.1.Giriş	1
1.2.Amaç	2
2.GENEL BİLGİLER.....	3
2.1.Türkiye Elektrik Piyasası	3
2.1.1.Türkiye’de elektrik piyasası tarihsel gelişimi ve yapısı	3
2.1.2. Elektrik piyasasının mevcut durumu	5
2.1.3. Piyasa yönetim sistemi	6
2.1.4. Gün öncesi piyasası	6
2.1.5. Gün içi piyasası.....	8
2.1.6. Dengeleme güç piyasası.....	9
2.1.7. Piyasa takas fiyatı belirleme yöntemi	10
2.2. ELEKTRİK PİYASA SİMÜLATÖRÜ	12
2.2.1. Simülatör kavramı	13
2.2.2. Önceki çalışmalar	13
3. TEZ ÇALIŞMASI.....	18
3.1. Piyasa Simülatörü.....	19
3.1.1. Simülatör mimarisi.....	19
3.1.2. Akış diyagramı.....	21
3.2.Simülatör Temel Bileşenleri.....	23
3.2.1. Programlama araçları.....	23
3.2.2. C# dili ve .net mimarisi	23
3.3.Uygulama Geliştirme.....	25

3.3.1. Platform oluřturma	25
3.3.3. Giriř ekranı(LoginSayfasiEkran.cs)	32
3.3.4.řeffaflık platform ana sayfası.....	34
3.3.5. Gün öncesi piyasa ekranı	35
3.3.6. Gün içi piyasa ekranı	36
3.3.7. Dengeleme güç piyasa ekranı	37
3.3.8. Kullanıcı tanımlama ekranı.....	37
3.3.9. Kullanıcı güncelleme ekranı	38
3.3.10. Piyasa mesajekranı.....	39
3.3.11.Üretim.....	40
3.3.12.Tüketim	40
3.3.13.Oyun, senaryolar ve çıktıları	41
3.4. Oyunun Çoklu Ortama Yüklenebilmesi.....	53
3.4.1. Web kanallı bilgisayara indirilmesi	54
4.SONUÇ.....	55
EKLER.....	61
ÖZGEÇMİř.....	84

ÖZET

EĞİTİM AMAÇLI ELEKTRİK PİYASASI SİMÜLATÖRÜNÜN GELİŞTİRİLMESİ

Elektrik piyasalarında serbestleşme hareketi özellikle 1990'lı yılların ikinci yarısından itibaren gelişmiş ülkelerin öncülüğünde hızlı bir değişim göstermiştir. Böylece elektrik piyasa yapıları, tekelci yapıdan içinde rekabeti barındıran piyasalar haline dönmüştür. Bu bağlamda birçok ülke yasal düzenleme ve buna bağlı mevzuatlarını hızla tamamlayarak, serbest piyasa yapısını oluşturmuştur. Türkiye, 4628 Elektrik Piyasası kanununun 2001'de çıkartılıp yürürlüğe girmesiyle beraber birçok yeniliği bünyesine katmıştır. Diğer taraftan özel sektöryatırımları da buna bağlı olarak artmıştır. 2011 yılında başlayan gün öncesi piyasa yapısı, 2015 yılından itibaren enerji borsası olarak EPIAŞ altında devam etmektedir.

Bu çalışma, yeniden yapılanan elektrik sektörünün oluşturduğu türev piyasa işleyişinin tüm paydaşları tarafından daha anlaşılabilir olmasını sağlamak için bir elektrik piyasası simülatörü geliştirilmesini amaçlamıştır. Bu kapsamda web tabanlı bir simülatör geliştirilmiş ve piyasaya yeni girecek veya piyasa içersinde bulunan kişilerin gelişimine katkı yapmayı hedeflemiştir. Bu kapsamda çalışmada Türkiye ve dünyadaki enerji piyasaları ve gelişiminden bahsedilip, özelinde elektrik enerji piyasa yapısı detaylandırılarak anlatılmıştır. Ayrıca gün öncesi piyasası, gün içi piyasası ve ikili anlaşmalar diğer bir tabirle türev piyasa kavramları irdelenerek piyasanın işleyiş biçimi ve işlevselliği aktarılmıştır.

Böylece bu çalışma lisans ve lisansüstü öğrenciler başta olmak üzere elektrik piyasasında katılmak isteyen çalışanlara, Türkiye elektrik piyasasının temelini öğretmek ve gerçek piyasa koşullarında alıştırma ve egzersiz yapma imkânı sağlayacaktır.

ABSTRACT

DEVELOPMENT OF AN ELECTRICITY MARKET SIMULATOR FOR TRAINING

The liberalization process in the electricity markets has been changing rapidly since the second half of the 1990s under the leadership of developed countries. Thus, electricity market structures, monopolistic structure has turned into a market of competition. In this context, many countries have completed their legal regulations and related legislation quickly, consisted of free market structure. Turkey, with 4628the enactment and enactment of the Electricity Market law in 2001, it has undergone several structural reforms. Private sector investments continued to increase with this momentum. Day ahead market started to service in 2011, Since 2015, it is under the energy exchange EPIAŞ.

This study aims to develop an electricity market simulator which is derivative market operation of restructured electricity industry expect to be more understandable. In this case, a web based electricity market simulator and is intended to contribute to the development of who are new to the market or in the market moment. This comprehensive work which is explained in detail from Turkey and the world energy markets and development. Also the day ahead market, day market and bilateral agreements, in other words, derivative market concepts are explained and the functioning and functionality of the market is explained.

Consequently, this study refers to undergraduate and graduate students who willing to participate in the electricity market, to teach the basics of Turkey's electricity market and will provide the opportunity to practice and exercise in real market conditions.

SEMBOLLER

I : Bir saatlik satış tekliflerinin kümesi

J : Bir saatlik dilimde alış teklifleri

B^s : Blok satış teklifi kümesi

B^d : Blok alış teklifi kümesi

F^s : Esnek satış teklifleri bütünü

P_{min} : Minimum fiyat limit

P_{max} : Maksimum fiyat limit



KISALTMALAR

EPDK	:Enerji Piyasası Düzenleme Kurumu
ETKB	:Enerji ve Tabii Kaynaklar Bakanlığı
TEİAŞ	:Türkiye Elektrik İletim A.Ş.
EÜAŞ	:Elektrik Üretim A.Ş.
PMUM	:Piyasa Mali Uzlaştırma Merkezi
TEDAŞ	:Türkiye Elektrik Dağıtım A.Ş.
TETAŞ	:Türkiye Elektrik Toptan Satış A.Ş.
YİD	:Yap-İşlet Devret
Yİ	:Yap-İşlet
İHD	:İşletme Hakkı Devri
GÖP	:Gün Öncesi Piyasası
GİP	:Gün İçi Piyasası
DGP	:Dengeleme Güç Piyasası
PTF	:Piyasa Takas Fiyatı
YAL	:Yük Alma
YAT	:Yük Atma
PYS	:Piyasa Yönetim Sistemi
YSA	:Yapay Sinir Ağları
MYTM	:Milli Yük Tevzi Merkezi

ŞEKİL LİSTESİ

Şekil 2.1. Türkiye Elektrik Piyasasının Gelişimi.	3
Şekil 2.2. Türkiye elektrik piyasa yapısı.	4
Şekil 2.3. Elektrik Piyasasının Mevcut Durumu.	6
Şekil 2.4. Gün öncesi piyasası.	7
Şekil 2.5. Genel olarak bir elektrik piyasasının bileşenleri.	8
Şekil 2.6. Piyasa yapısı.	9
Şekil 2.7. Türkiye Elektrik Piyasasındaki farklı piyasaları gün örneği.	10
Şekil 2.8. Piyasa Giriş Ekranı.	10
Şekil 2.9. Çok katmanlı fonksiyonel akıllı şebeke mimarisi.	14
Şekil 2.10. Power TAC platform yapısı.	15
Şekil 2.11. Müşterilerin kontratlarının görüldüğü sayfa.	16
Şekil 2.12. Mascem mimarisi.	16
Şekil 2.13. Simülatör akış mimarisi.	17
Şekil 2.14. Piyasa fiyat eşleşme grafik.	17
Şekil 3.1. Piyasa katılımcıları internet aracılığıyla sisteme entegre olurlar	54
Şekil 3.2. Katmanlı mimari.	20
Şekil 3.3. Sunucu/istemci mimarisi.	21
Şekil 3.4. Elektrik Piyasası Akış Diyagramı	21
Şekil 3.5. Platform tabanının oluşturulması.	25
Şekil 3.6. Visual studio database mimarisinin entegre edilmesi	26
Şekil 3.7. Tablolar arası ilişkileri gösteren uml diyagramlar	26
Şekil 3.8. Veri tabanı.	27
Şekil 3.9. Kullanıcılar tablosu.	27
Şekil 3.10. Kullanıcıtip tablosu.	27
Şekil 3.11. Üreticiteklif tablosu.	28
Şekil 3.12. Tüketiciteklif tablosu	28
Şekil 3.13. Üreticiblokteklifler tablosu	29
Şekil 3.14. Tüketiciblokteklifler tablosu.	29
Şekil 3.15. Ptfkayıtlar tablosu	30
Şekil 3.16. Piyasamesajsisitemi tablosu.	30
Şekil 3.17. Sistemyonutablosu	31

Şekil 3.18. Eşlesenteklifler tablosu	31
Şekil 3.19. İtiraz tablosu.....	31
Şekil 3.20. Entsokod tablosu.	32
Şekil 3.21. Ön yüz http protokolünden erişilebilen aspx ve winform(C#) teknolojisiyle kodlanması	32
Şekil 3.22. Kullanıcı ad veya şifresinin yanlış girilmesi durumunda verilen uyarı.	33
Şekil 3.23. Giriş ekranı hakkında bölümü.....	33
Şekil 3.24. Ön yüz http protokolünden erişilebilen aspx ve winform(C#) teknolojisiyle kodlanması(Kullanıcı işlem yapacağı ekran).	34
Şekil 3.25. Gün Öncesi Piyasa ekranı	35
Şekil 3.26. Gün içi piyasa ekranı.....	35
Şekil 3.27. Dengeleme güç piyasa ekranı	36
Şekil 3.28. Yeni kullanıcıların tanımlandığı ekran	37
Şekil 3.29. Kullanıcı güncelleme ekranı	37
Şekil 3.30. Üreticilerin taahhüt ettiği gücü aksaklıklar nedeniyle temin edememesi durumunda mazeret girebilecekleri ekran	38
Şekil 3.31. Üretim ekranı	39
Şekil 3.32. Tüketim ekranı	39
Şekil 3.33. Web giriş ekranı	40
Şekil 3.34. Yeni kayıt ekranı	41
Şekil 3.35. Yeni kullanıcı kayıt örneği.....	41
Şekil 3.36. Yeni kullanıcı kayıt mail.....	42
Şekil 3.37. Şifremi unuttum	42
Şekil 3.38. Şifremi unuttum mail	43
Şekil 3.39. Sistem operatörü mail	43
Şekil 3.40. Admin ana menü	43
Şekil 3.41. Kullanıcı tanım ekranı.....	44
Şekil 3.42. Tekil, sınıf oyun seçim ekranı	44
Şekil 3.43. Tekil oyun seçimi.....	45
Şekil 3.44. Ana ekran	45
Şekil 3.45. Gün öncesi piyasa teklifi.....	46
Şekil 3.46. Saatlik ptf izleme ekranı.....	47

Şekil 3.47. Teklif izleme ekranı	47
Şekil 3.48. Admin sınıf oluşturma ekranı	48
Şekil 3.49. Kullanıcıların tanımlanması	48
Şekil 3.50. Sınıfın tanımlanması	50
Şekil 3.51. Ptf belirleyip, teklif eşleştirmesi yapılması.	50
Şekil 3.52. Ptf oluşması.....	51
Şekil 3.53. Tekliflerin kontrolü	51
Şekil 3.54. Saat 13:00 için oluşan Ptf.....	52
Şekil 3.55. Piyasa simülatör kurulum dosyası.....	52
Şekil 3.56. Piyasa simülatör kurulum aşaması	52
Şekil 3.57. Piyasa simülatör kurulum aşaması	54

TABLO LİSTESİ

Tablo 3.1. Tüketici1 Teklifi	47
Tablo 3.2. Tüketici2 Teklifi	47
Tablo 3.3. Üretici1 Teklifi	47
Tablo 3.4. 20 tane üreticinin saat 13:00 için girdiği teklifler.....	50



1. GİRİŞ VE AMAÇ

1.1.Giriş

Hızla değişen ve ilerleyen teknolojik alt yapı ve bunun bir parçası olarak önümüze çıkan günlük yaşama olan adaptasyon göz önüne alındığında, değişen, gelişen ve küreselleşen dünyada enerji büyük önem arz etmektedir. Dış etkenlerin başında gelen küresel ısınma ve buna benzer çevresel etkenler ve enerjide dışa bağımlılık benzeri problemler, elektrik enerjisi tüketiminin verimliliği konusunu daha önemli kılmaktadır. Bu süreçte en kritik husus, elektrik üretim ve tüketim dengesini sağlanabilmesidir. Elektrik enerjisine duyulan ihtiyacın artmasıyla birlikte rekabetçi ve serbest piyasaya duyulan ihtiyaç daha belirgin hale gelmiştir [1]. Dünyada özellikle Şili’de elektrik sektöründe başlayan kapsamlı ve rekabetçi yapılanma 1980 yıllarında yoğunlaşarak İngiltere gibi gelişmiş ve gelişmekte olan ülkelerde uygulanmaya başlanmıştır [2].

Buna karşılık, zaman içerisinde elektrik üretim sektöründe rekabetten söz edilebileceği ve buna bağlı iletim ve dağıtım işlevlerinin düzenlenmesi ihtiyacının da doğmasından dolayı bu yapının düzenlenmesi gerektiği anlaşılmıştır. Yaklaşık 1990’lı yılların başında başlayan bu sürecin sonunda elektrik piyasaları tam rekabetçi hal almıştır [3].

1980’li yılların başından günümüze dek gelişmiş ülkelerin öncülüğünde yeniden yapılanma içerisinde olan elektrik piyasaları, öncesinde dikey bütünleşik yapıda işlemektedir. Tarihler 1982’i gösterdiğinde Şili ve ardından İngiltere, Galler, Norveç, Avustralya, İspanya, ABD’nin çeşitli eyaletlerinde ve Kıta Avrupasında elektrik endüstrisinde piyasa odaklı yaklaşımlara geçiş yaşanmıştır. Arz ve talebin şeffaf, dengeli ve verimliliğin maksimum düzeyde olması sağlanarak ekonomik açıdan etkinliğin artırılması hedeflenmiştir [4]. Gerek OECD gerekse de OECD dışındaki ülkeler, yaptıkları uygulamalarla nihayetinde rekabetçi elektrik piyasaları oluşturmaya çalışmışlardır. Dikey yapıda piyasanın bulunduğu yapılarda üretim, iletim, dağıtım ve perakende satış işlemleri tek bir şirketin tekelinde bulunmaktadır. Bu yapının yönetimi devlet ya da özel bir şirket tarafından işletilebilmektedir. 1990’ların ardından dikey entegre piyasaların ayrıştırılması planlanmış olsa da çoğu Afrika ülkesinde bu piyasa yapısı kullanılmaktadır. 2012 yılında Uluslararası Enerji Ajansı’nın (IEA) yaptığı çalışmada monopol elektrik piyasalarının tüketilen toplam elektrik miktarının hemen hemen %6’sını oluşturduğunu ortaya koymuştur [5].

Yeni yerleşim yerlerinin günden güne arttığı, var olanların her geçen gün yenilendiği, buna bağlı kullanıcılarının alışkanlıklarında değişmesiyle birlikte artan talebin sistematik bir yapıda üretim, dağıtım ve iletim kanallarının geleceği de göz önünde bulundurarak planlanması gerekmektedir [6]. Bu gelişmeler kapsamında elektrik enerjisinin dağıtımı ve iletiminin önemi daha da önem kazanmaktadır. Sistemde oluşabilecek arz yetersizliği durumunda alternatif enerji kaynaklarının dahil edilmesinin yanı sıra talepteki beklenmeyen artışlara karşın şebekede tutulması gereken emre amade enerji kaynaklarının minimum seviyede hazır bekletilmesi gibi durumlar elektrik tüketiminde tahminlemenin ve özellikle tüketiminin izlenmesinin önemini artırmaktadır [7]. Elektrik piyasaları kendi içinde kompleks ve birçok karakteristiği içeren bütündür. Elektrik piyasasının yeniden yapılanmasıyla birlikte karmaşıklığı ve öngörülemezliği ile beraber optimize fiyatlar ile fiyat oluşumu işleyişi de başlamıştır [8]. Yeniden yapılanmanın elektrik enerjisi sektöründe şeffaflık, rekabet ve hesap verebilirliğin kolaylaştırılması planlanmaktadır [9].

1.2.Amaç

Bu çalışmada, 2013 yılında yürürlüğe giren 6446 sayılı yeni Elektrik Piyasası Kanunu ile yeni bir hal alan elektrik piyasasının sektörde çalışan ve çalışacak tüm katılımcılar tarafından daha iyi kavratmak, hizmet içi, lisans ve lisansüstü eğitimleri sırasında alınacak öğretimler ile daha iyi analiz edilebilmesi adına özellikle Türkiye elektrik piyasasının yeni yapısını ve işleyişini kavratmak amacıyla bir elektrik piyasası simülasyon tasarımı hedeflenmiştir.

Bu kapsamda ikinci bölümde Türkiye elektrik piyasası üzerinde durulmuş ve piyasa işleyişlerinden bahsedilmiştir. Ayrıca elektrik piyasa simülasyonu kavramı açıklanmış ve yapılan önceki çalışmalar sunulmuştur. Tez çalışması bölümünde ise, geliştirilen piyasa mimarisi ve akış diyagramı verilmiş simülasyonun temel bileşenleri sunularak, oluşturulan web tabanlı platform ekranları aktarılmıştır. Kullanıcıların oyuna katılımı ile ilgili senaryolar sunulmuştur.

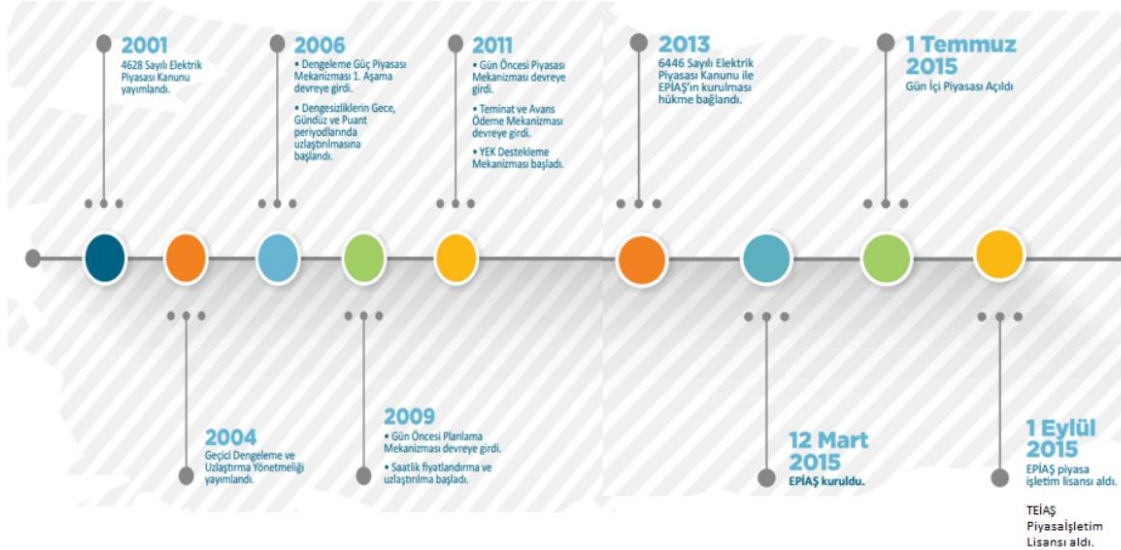
2.GENEL BİLGİLER

2.1.Türkiye Elektrik Piyasası

2000’li yılların başında 4628 sayılı Elektrik piyasa kanunun yürürlüğe girmesiyle birlikte serbestleşme süreci başlamıştır. Serbestleşme öncesinde tek elde Türkiye Elektrik Kurumunda toplanan tüm faaliyetler, düzenlemeler sonucunda iletim, üretim ve dağıtım faaliyetleri birbirlerinden ayrıldı. Türkiye 21 adet dağıtım bölgesine bölünmesiyle birlikte buna bağlı organizasyonlar da özelleştirildi. Son yapılan güncellemelerle birlikte tüketiciler elektrik tedarik şirketlerini de seçebilir hale geldi [10].

2.1.1.Türkiye’de elektrik piyasası tarihsel gelişimi ve yapısı

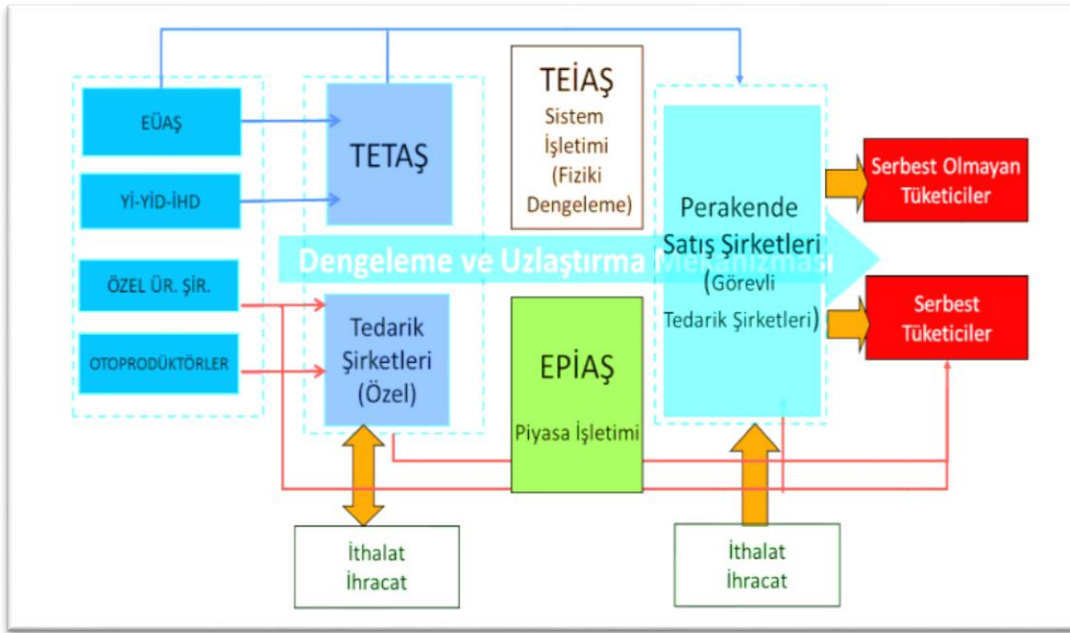
Türkiye, elektrik piyasasında köklü yapısal reformunu 2001 yılında çıkarılan 4628 sayılı Elektrik Piyasası Kanunu ile başlatarak (Şekil 2.1), öncesinde fiyatların taraflarca serbest biçimde belirlendiği ikili anlaşmalara dayanan bir piyasaydı. Gün Öncesi Piyasası’nın 2011 yılında gelmesiyle ilk etapta % 25’ler mertebesinde işlem hacmi bulunan spot piyasa, işlem hacminin %50’den fazlasının yapıldığı piyasaya dönüşmüştür [11].



Şekil 2.1. Türkiye Elektrik Piyasasının Gelişimi [12].

Spot piyasası içerisindeki alt piyasalar (gün içi piyasası, gün öncesi piyasası, dengeleme güç piyasası) piyasa fiyatlarının meydana geldiği piyasalardır. Arz ve talebin dengelendiği noktalarda uygulanan uçta fiyatlandırma ilkesine bağlı işlemektedir. Bilhassa gün öncesinde işlenen fiyatlar (piyasa takas fiyatı) spot piyasasına referans fiyat olup, hemde elektrik piyasasının önemli diğer bacağı ikili anlaşmalar (EPDK) tarafından

düzenlenen tarifelerde temel alınmaktadır. Bu kapsamda spot piyasada oluşan piyasa takas fiyatı, elektrik piyasasının belirleyici unsurudur. Piyasa takas fiyatı genellikle spot fiyat olarak bilinir. Spot elektrik piyasası genel olarak teslimattan bir gün önce işleyen ve teslimattan bir gün önce sona erer. Böyle bir sistemde teknik manada arz ve talebin değişkenliğine göre pazarın balansının dengede tutulması çok önemlidir. TSO (İletim sistem operatörü) teslimattan kısa bir süre önce dengeleme güç piyasası devreye girer. Bu teknik piyasa hızlı dakik ve hatta saniyeler içinde fiziksel teslimatı yapabilir olmaktadır [13].



Şekil 2.2. Türkiye elektrik piyasa yapısı [15].

Yeni Elektrik Piyasası Kanunun 30.03.2013'te yürürlüğe giren 6446 sayılı kanununa göre elektrik iletim faaliyetlerinin yürütülmesi, fiziki manada elektriğin dengelenmesinin yanıladengeleme güç piyasasının işletilmesini Türkiye Elektrik İletim A.Ş. (TEİAŞ) üstlenmiştir. Öncelerde TEİAŞ'ın insiyatifinde bulunan havuz piyasası 6446 sayılı kanunla birliktebağımsız bir kuruluş olan Enerji Piyasaları İşletme Anonim Şirketi (EPIAŞ) tarafından işletilmesi görevini vermiştir. Böylece EPIAŞ Mart 2015'te kurularak, piyasa işletim faaliyetlerini Eylül 2015'ten bu yana yürütmektedir [11].

Geçmişte dikey bütünleşik yapıda olan Türkiye Elektrik Kurumu, günümüzde üretimden iletme kadar dört ana şirkete ayrılmıştır. Dört ana şirketi; dağıtım, üretim, iletim ve satış ticaretlerini yapan sırasıyla, TürkiyeElektrik Dağıtım A.Ş. (TEDAŞ), Elektrik Üretim

A.Ş. (EÜAŞ), Türkiye Elektrik İletim A.Ş. (TEİAŞ) ve Türkiye Elektrik Ticaret ve Taahhüt A.Ş. (TETAŞ) şeklinde tanımlayabiliriz. Piyasanın serbestleşmesi, rekabetçi bir yapıya bürünmesi ve özel sektör yatırımlarının artırılması amacıyla bağımsız kurum olan Enerji Piyasası Düzenleme Kurumu (EPDK) yetki alanlarını kapsayan güncel bir kimlikle piyasadaki enerji oyuncularını lisans edindirmek ve buna bağlı piyasası düzenlemek ve denetlemek amacıyla kurulmuştur [14].

2.1.2. Elektrik piyasasının mevcut durumu

Türkiye’de 80’li yıllarda başlayan yeniden yapılanma hareketi 2000’li yılların başında kabul edilip yürürlüğe geçen 4628 sayılı Elektrik Piyasası Kanunu (EPK) yasal zeminde kendine yer etmiştir. Geçen zamanla birlikte serbestleşme sürecinde çeşitli adımlar kaydedilmiştir. Bu süreçte elektrik piyasasında meydana gelen bazı taleplerin karşılanması için 2013 yılında yürürlüğe giren 6446 sayılı EPK ve ilgili yönetmelikler yürürlüğe girmiştir ve bu değişikliklerle birlikte elektrik piyasasında yeni bir dönem başlamıştır. Sektörde, piyasa oyuncusu konumundaki santrallerin ürettikleri elektrik enerjilerini, elektrik piyasasının bir parçası iletim sistemi uzun mesafelere taşımaktadır. İletim sisteminden gelen elektriği dağıtım şirketleri kullandıkları dağıtım sistemi ile nihai kullanıcılara, tüketicilere iletmektedir. TETAŞ ve türevi olan elektrik toptan satışı yapan şirketler alış ve satış risklerini kontrol ederek piyasanın ticari manada ilerlemesini sağlamaktadır. Nihai tüketicilere elektriği perakende satış şirketleri temin etmektedir. Bu amaç doğrultusunda yeniden yapılanan piyasada, rekabetin geliştirilerek fiyatlandırmada gerçekçi maliyetlerin oluşturulup, yerli, ucuz ve efektif yatırımların çoğaltılması hedeflenmiştir. Kısacası yeniden yapılanan piyasada hedef, pazardaki rekabetle birlikte doğru bir pazar oluşmasıdır.

2013’te kabul gören ve yürürlüğe giren Elektrik Piyasası 6446 sayılı kanunu ile elektrik piyasasının ihtiyaçlarını karşılayabilen, dünya enerji piyasalarıyla enterekte çalışabilen, etkin ve verimli bir yapı oluşturulması hedeflenmiştir. Böylece EİAŞ ile elektrik ticaretinin mali piyasa (elektrik borsası) yapısına bürünmesi amaçlanmıştır [16].

Teknolojik gelişmelerinde etkisiyle birlikte elektrik enerjisini enterekte sistemle gerek yurtiçi gerekse de yurt dışından piyasa oyuncularının spot piyasa ya da türev piyasalar üzerinden ticaret yapabilecekleri hale getirmiştir. Hem uluslararası entereksiyonla hem de ticari işlemler ile birçok piyasa birbirine bağlanmıştır [17].



Şekil 2.3. Elektrik Piyasasının Mevcut Durumu [18].

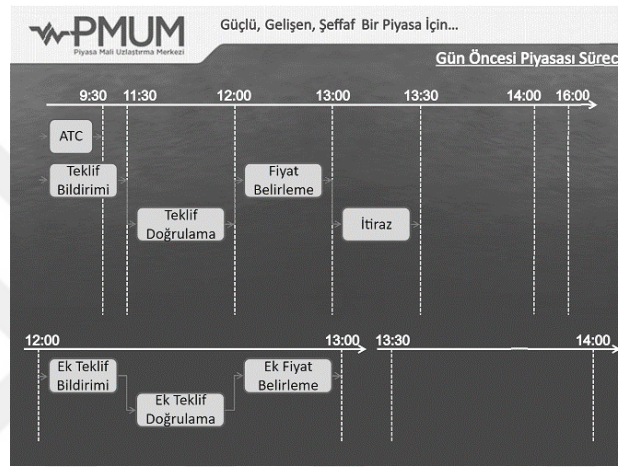
2.1.3. Piyasa yönetim sistemi

Piyasa takas fiyatları, dengeleme ve uzlaştırma sistemi kullanılarak işletilmekte olan 3 ana piyasa; gün öncesi piyasası, gün içi piyasası ve dengeleme piyasasının arz ve talebi doğrultusunda oluşan eğrilerinin doğrusal interpolasyon eğrilerinin kesişim noktasında uçta fiyatlandırma denklemlerine dayalı meydana gelmektedir. Oluşabilecek optimal maliyetle arz-talep dengesinin karşılanmasına yönelik en uygun hesaplamalar sonucunda ortaya çıkan üretim-tüketim görünümünün mali yapısı Piyasa Yönetim Sistemi (PYS) denilen yazılım aracılığıyla işletilmektedir [19-20].

2.1.4. Gün öncesi piyasası

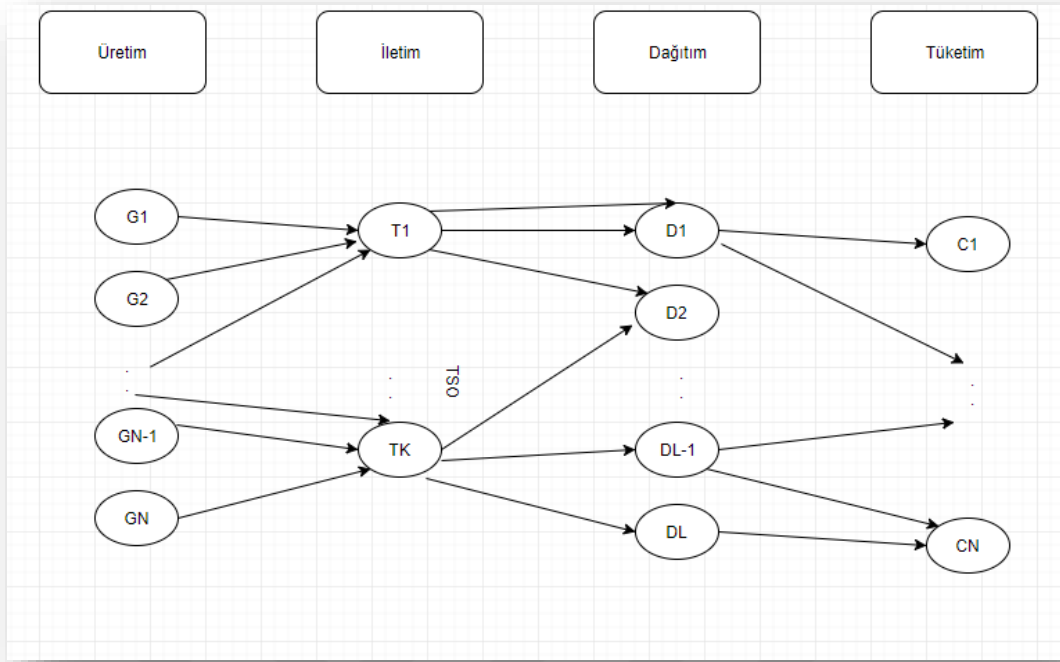
Gün öncesi piyasası, var olan günden bir sonraki gün için elektrik talebinin dengelenmesi için işletilen piyasa Gün öncesi piyasası (Göp)'tür. Göp, piyasanın önceden kestirelebilir bir yapıda olmasını sağlamasının yanı sıra katılımcıların risklerini minimize etmeleri için imkân sunmaktadır. Böylece gerçek zamanlı işleyen piyasadan kaynaklanan aksaklıkların giderilmesi için sistem işletmecisine gün öncesinden dengelenmiş piyasa bırakmaktadır. Piyasanın işleyişinde sistem işletmecisinin görevlerinden bir kaçına göz atacak olursak:

- Sistem işletmecisi hali hazırda kullanılabilir iletim kapasitelerini vebunun yanında piyasa katılımcılarının da bir sonraki gün için varsa ikili anlaşma miktarlarını ve tekliflerini bildirmesi,
- Arz-talep dengesinin optimal maliyetle belirlenmesi ve elde edilen veriler altında ilgili katılımcılara bildirilmesi,
- Katılımcıların emirlerinin kontrolü ve varsa itirazların sonuçlandırılması,
- Piyasa takas fiyatının belirlenip, piyasa katılımcılarına duyurulması [21].



Şekil2.4. Gün öncesi piyasası [22].

Gelişen ve büyüyen serbest bir piyasa hedefiyle ilerleyen Türkiye Elektrik Piyasasında katılımcıların dengeli ve etkin rol üstlenmesini sağlamak maksadıyla 1 Temmuz 2015 tarihinde Gün İçi Piyasası faaliyete geçmiştir. Gün İçi Piyasası, Gün Öncesi Piyasası ile Dengeleme Güç piyasası arasında köprü görevi görmektedir. Böylelikle Piyasanın dengelenmesine ve sürdürülebilirliğine katkıda bulunmatadır. Gün Öncesi Piyasasında teklif girişleri ile gerçekleşme zamanı arasında maksimum 36 saate varan bir fark oluşmaktadır.



Şekil 2.5. Genel olarak bir elektrik piyasasının bileşenleri [23].

2.1.5. Gün içi piyasası

Gün içi piyasası ile birlikte; üretim ve tüketim miktarlarındaki öngörülen veya öngörülemeyen değişimlerden kaynaklı dengesizliğe yol açabilecek durumlar gerçek zamana az bir süre zarfında dahi kontrol edilebilir, piyasa katılımcılarına gün içerisinde karşılaşılabilecekleri dengesizliklerin en aza indirilmesi ve dengelenmesi imkânı sunmaktadır. Akabinde piyasa oyuncuları Gün Öncesi Piyasasında kullanamadıkları kapasitelerini Gün Öncesi Piyasasının bitiminden sonra Gün İçi Piyasasında kullanabilme sunulmaktadır. Böylece piyasalardaki likiditede artış olmaktadır.

2015 yılı itibarıyla faaliyete geçen gün içi piyasasına katılım zorunlu değildir. Yenilenebilir enerji kaynaklarının ve buna bağlı santrallerin üretimlerinin doğal şartların değişebilir olmasından ötürü gün öncesinden tahmin edilebilirlik oranı güçtür. Üretimin özellikle yenilenebilir enerji kaynaklarının üretim tahminlerinden veya ani gelişen arıza durumlarından kaynaklanan dengesizlik maliyetleri gün içi piyasasıyla birlikte daha aza indirgenmesini sağlamıştır. Gün içi piyasası temelde gün öncesi ve dengeleme güç piyasası ile eşgüdümlü çalışan sürekli ticaretin yapılabildiği ve teslimat zorunluluğu bulunan bir piyasadır. Gün içi piyasası işlemleri saatlik dilimde gerçekleşip, 00.00’da başlayıp ertesi gün 00.00’da sona ermektedir. Verilen talimatlar fiziki teslimattan iki saat

öncesine kadar güncellenebilmektedir [24].

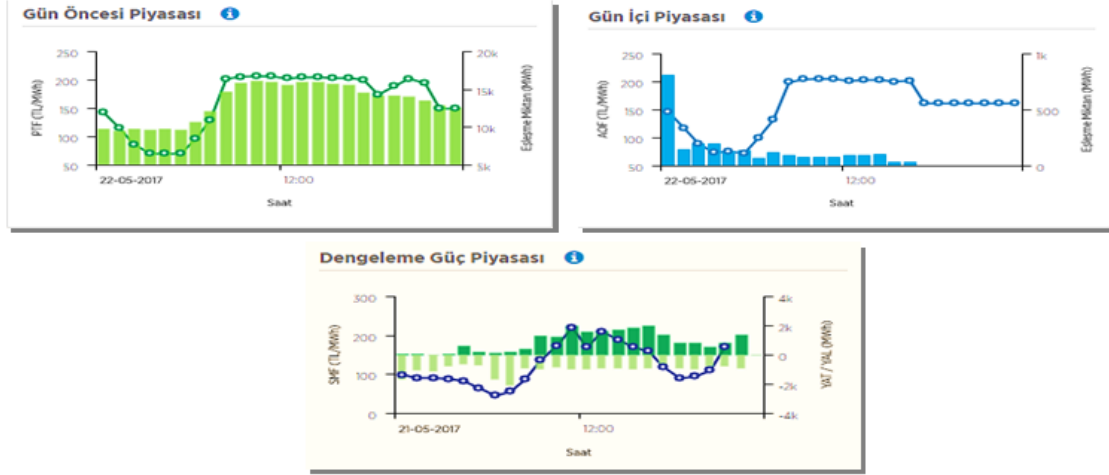
2.1.6. Dengeleme güç piyasası

Gerçek zamanlı dengeleme; yan hizmetler ve dengeleme güç piyasasının eşgüdümlü çalışmasından oluşmaktadır. Sistem işletmecisi emre amade kapasitesi bulunan üreticileri 15 dakika içerisinde sisteme girebilecek şekilde organizasyonu yapmaktadır. Buna yanında talep ve frekans kontrolü yan hizmetler tarafından yürütülmektedir. DGP’de işlemler günlük ve saatlik dilimde gerçekleştirilmektedir. DGP’ye katılan tüm piyasa oyuncuları emre amade kapasitelerini sisteme sunma zorunluluğu bulunmaktadır. Minimum 10 Mw’lık gücü alabilen veya atabilen birimler Dengeleme güç piyasasına katılımı zorunludur [18].



Şekil 2.6. Piyasa yapısı [25].

Türkiye ve benzeri elektrik piyasalarında GÖP te oluşan fiyatlar onu takip eden piyasalar için referans fiyat olmaktadır. Referans fiyatı, GÖP’te arz ve talep dengesinin kurulması için gerekli referans fiyat noktası olarak düşünebiliriz. PTF’in referans fiyat olduğu piyasalardan bir diğeri de DGP’dir. GÖP’te oluşan PTF, DGP için bir referans fiyatı kabul edersek, PTF ve SMF’nin ortalamada birbirlerine eşit olması beklenir. Ancak, her gün için böyle olması beklenmez. Arz ve talebin gün öncesindeki dengesini gerçek zamanda da yüksek oranda koruması ile mümkündür, yenilenebilir enerji kaynaklarının sistemde artışı ile bunu sağlamak gittikçe zorlaşmaktadır. Bu iki fiyatın çoğu zaman birbirinden farklılaşması beklenebilir, ancak SMF nin PTF etrafında rassal bir şekilde dağılması gerekir. Eğer bu rassallık bulunmuyorsa, SMF PTF’nin üzerinde veya altında olma eğilimi gösterir ve bu da bir etkenin sistematik olarak bu iki fiyatın farklılaşması üzerinde etkili olduğuna işarettir [26].



Şekil 2.7. Türkiye Elektrik Piyasasındaki farklı piyasaları gün örneği [27].

Kullanıcı EPIAŞ üzerinden işlem yapması için her üç piyasa için aynı piyasa girişi ekranını kullanması gerekmektedir.

Şekil 2.8. Piyasa Giriş Ekranı [27].

2.1.7. Piyasa takas fiyatı belirleme yöntemi

Kullanılan terimler;

➤ t, T : periyot indisi ve periyotlar kümesi

- I : Bir saatlik saatlik satış teklifleri bütünü
- $l, L(i)$: indisi ve segmentler bütünü, $i \in I \cup J$
- F^s : esnek satış teklifi kümesi Parametreler:

- P_{itl}^0, P_{itl}^1 : t zamanında saatlik teklifi l segmentinin ilk ve son fiyatı (Satış, $P_{min} \leq P_{itl}^0 < P_{itl}^1 \leq P_{max}$ ve alışı yönlü teklifler için $P_{max} \geq P_{itl}^0 > P_{itl}^1 \geq P_{min}$)
- Q_{itl}^0, Q_{itl}^1 : t zamanında i saatlik teklifinin l segmentinin ilk ve son miktarı ($0 \leq Q_{itl}^0 \leq Q_{itl}^1$)
- Q_b, Q_f : b ve f tekliflerinin mutlak değerleri
- δ_{bt} : b blok teklifi t periyodunda geçerli ise 1, değilse 0 olan ikili parametre
- Karar değişkenleri:
- x_{itl} : t periyodundaki i saatlik teklifinin l segmentinin kabul oranı
- p_t : t periyodundaki ptf

➤ Her periyotta arz talebe eşit olmalıdır.

$$\sum_{i \in I} \sum_{l \in L(i)} (Q_{itl}^1 - Q_{itl}^0) X_{itl} + \sum_{b \in B^S} \delta_{bt} Q_b Y_b + \sum_{f \in F^S} Q_f Z_{ft} - \sum_{f \in F^S} \sum_{l \in L(i)} (Q_{itl}^1 - Q_{itl}^0) X_{itl} - \sum_{b \in B^S} \delta_{bt} Q_b Y_b = 0, \forall t \in T \quad (2.1)$$

Satış teklifleri formül: $\forall i, t, l$

$$X_{itl} = \begin{cases} 1, & \text{if } P_{itl}^1 < p_t \\ \frac{p_t - P_{itl}^0}{P_{itl}^1 - P_{itl}^0}, & \text{if } P_{itl}^0 \leq p_t \leq P_{itl}^1, l \neq 1 \\ 0, & \text{if } p_t < P_{itl}^0 \end{cases} \quad (2.2)$$

Alış teklifleri formül: $\forall j, t, l$

$$X_{itl} = \begin{cases} 1, & \text{if } P_{itl}^1 < p_t \\ (p_t - P_{itl}^0) / (P_{itl}^1 - P_{itl}^0), & \text{if } P_{itl}^0 \leq p_t \leq P_{itl}^1, l \neq 1 \\ 0, & \text{if } P_{itl}^0 < p_t \end{cases} \quad (2.2a) [28].$$

2.2. ELEKTRİK PİYASA SİMÜLATÖRÜ

Elektrik piyasaları, çok sayıda farklı varlığı içeren, en iyi avantajları ve kazançları elde etmek için belirli özellikleri ve hedefleri olan, kararlarını veren dinamik bir sahnede oynayan karmaşık ortamlardır. Rekabetçi elektrik piyasaları için çok etkili sistem, piyasa oyuncularını modelleyen ve piyasadaki işlemlerini taklit eden çok etkenli bir elektrik pazar simülatörü ile olabilir. Piyasa oyuncuları, belirli özelliklere ve amaçlara sahip olan, kararlarını veren ve diğer oyuncularla etkileşim halindeki kişilerdir. Elektrik piyasalarında uygulanması, potansiyel bir araç olarak ortaya çıkabilmektedir [29,30].

Elektrik piyasaları, bir dereceye kadar diğer sektörlerden elektrik sektörüne geçen oyuncu sayısının artması nedeniyle daha rekabetçi hale gelmektedir. Bu, aslında, dağıtık üretime (küçük ölçekli yenilenebilir enerji üretim merkezleri) sağlanan teşviklerden kaynaklanmaktadır. Bu alandaki ilgili değişiklikler hala devam etmekte, ulusal ve uluslararası pazarlara yayılmaktadır. Karar vericiler, bu nedenle, elektrik piyasası oyuncularına müzakere sürecinde yardımcı olmak için gerekli çalışmalar yapmaktadır [30].

Elektrik piyasalarının karmaşıklığı altında özellikle yenilenebilir enerji kaynaklarının ve elektrik piyasalarının yeniden yapılandırılması sürecinin yeterli bir şekilde uyum ihtiyacı vardır. Simülasyon araç kullanımının artması, bu mekanizmaları anlamının gereği ve ilgili oyuncuların etkileşimlerinin piyasaların sonuçlarını nasıl etkilediğini görebilmektedir. Yeniden yapılandırılmış toptan elektrik piyasalarının incelenmesine yönelik çeşitli modelleme araçları ortaya çıkmıştır. Ortak bir kısıtlamaya sahip olmakla birlikte, bilgi ve bilginin değişimine izin vermek, farklı pazar modellerini test etmek ve farklı sistem oyuncularının ortak pazar ortamlarında etkileşime girmesine izin vermek için çeşitli sistemler arasında birlikte çalışabilirlik gerekmektedir [31].

Elektrik piyasalarının uygulanmasında oyun senaryolarının değerlendirilmesi ve öngörülmesi ve oyuncuların hedeflerine daha iyi uyan davranış sergilemek için stratejik olarak tepki verme imkânı tanıyan yeni bir senaryo analiz algoritmaları bazı çalışmalarda kullanılmıştır. Böylece en olası senaryoları tanımlamak için rakip oyuncuların davranışlarının modellerini oluşturmak için yaptıkları eylem tahminlerini içeren çalışmalar geliştirilmektedir. Senaryolar tanımlandıktan sonra, gerçekleştirilecek eylemin seçimini desteklemek için oyun teorisi uygulanmakta ve oyun teorisi kullanımı ile

pazardaki dengesizliđi sađlamak için deđil, belirli bir aracı desteklemek için tasarımlar yapılabilmektedir. Elektrik piyasası müzakerecileri için karar desteđi sađlamak için farklı bir yöntem sunan, farklı pazarlık görüşmeleri için farklı stratejik yaklaşımların bağdaştırılması ve her müzakere bağlamı için her zaman en uygun olanın seçilmesini sađlamak hedefleyen çalışmalardır. Bu metot ile oyuncular, hedeflerine tam olarak ulaşabilmeleri için karar desteđi sađlayan çok etkenli bir sistem olan uyarlanır öğrenme stratejik teklif sistemine bütünleşmiş edilmiştir. Bu tür müzakere varlıklarına etkin öğrenme yetenekleri sađlamak için yapay zekâ metotlar ve veri analiz algoritmaları kullanılmaktadırlar [32].

2.2.1. Simülatör kavramı

Simülasyon kavramını bir sistemin davranışlarını anlamak ya da sistemin işlemedi için belirlenecek stratejileri deđerlendirmek amacına yönelik olarak, gerçek sistem modelinin tasarımılanması ve bu model ile deneylerin yürütülmesi süreci olarak tanımlanabilir. Simülatörü işlevleri yerine getiren örnekleme dayalı deneysel ve davranışsal yöntemin uygulanabilir halidir [33].

2.2.2. Önceki çalışmalar

Yeniden yapılanan ve ticari işleyişi ile farklı bir hal alan elektrik piyasalarını bireylerin daha iyi anlaması için benzetmek amacıyla yapılmış olan çalışmalar bu bölümde sunulmuştur. Bunlar, piyasa katılımcılarının davranışlarının benzetip, geliştirilen stratejileri analiz çalışmasının yapılması için elektrik piyasalarının bir oyun şeklinde modellenmelerini içeren çalışmalardır.

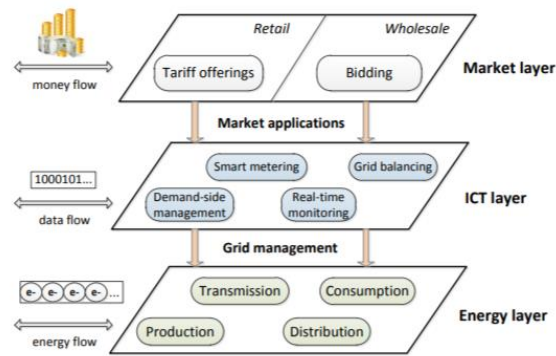
2.2.2.1. Enerji oyunu: Rekabetçi piyasada uzun dönemli elektrik piyasa simülasyon oyunu

L.J. De Vries, E. Subramahianbandve E.J.L. Chappinc'inbirlikte yürüttüğü enerji piyasası için geliştirilmiş simülasyon oyunu çalışmasında, elektrik piyasalar işleyişini sınıflaştırarak, öğrencileri üreticiye ve tüketiciye, öğretmenlere operatör rolü verilerek tekliflerin sistemde tutulması sađlanıyor. Oyuncular beş ayrı üretim şirketi pozisyonunda, ürettikleri elektriđi satarak piyasa ortamı oluşup, rekabet etmektedirler. Genellikle iki ya da üç katılımcı bir üretici şirket için rol paylaşp, uzun periyotlu planlama yapılp üretim ve tüketim dengesi oluşturuluyor. Tekliflerden oluşan takas fiyatının oluşmasının ardından arz-talep dengesine göre eşleşmeler yapılmaktadır. Eşleşen tekliflerin

sonucunda sisteme girilmiş hesaplara ilgili tutarlar aktarılmaktadır[34].

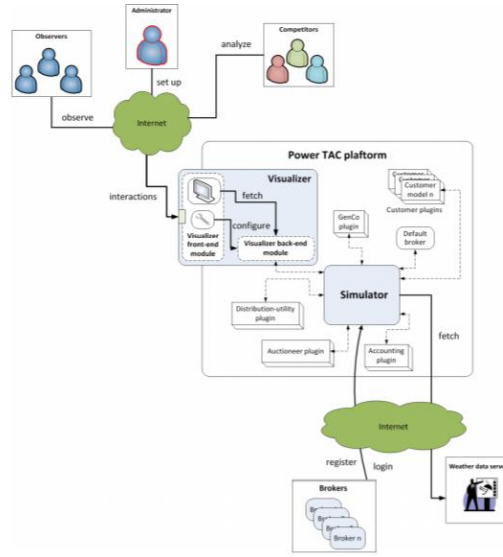
2.2.2.2. Enerji piyasası için simülasyon platformu

JuricaBabić, yüksek lisans tezinde enerji piyasası için platform geliştirmesi üzerine yaptığı çalışmasında powerTAC (Enerji ticareti) ismini verdikleri çalışmasında proje paydaşları; Selanik Aristoteles Üniversitesi, Carnegie Mellon Üniversitesi, Delft Teknoloji Üniversitesi, Minnesota Üniversitesi, Rotterdam Yönetim okulu ve modülü geliştiren Zagreb Üniversitesi'dir. Projede akıllı şebeke fonksiyonları da içine katarak 3 katmanlı mimariyi kullanarak elektrik piyasa platformu geliştirdi.



Şekil 2.9. Çok katmanlı fonksiyonel akıllı şebeke mimarisi [35].

Java teknolojisiyle geliştirilen modülde use-case senaryolarına bağlı kalarak piyasa işleyişinin olmazsa olmazı olan müşteriler için fiyatları, tekliflerin durumlarını piyasalarda meydana gelen fiyat kırılımlarını inceleyebilecekleri ekranları sağlamışlardır [35].



Şekil 2.10. Power TAC platform yapısı [35].

2.2.2.3. Kümelenmiş elektrik üreticilerine ilişkin bir oyun

Alper Günaydın, “Kümelenmiş elektrik üreticilerine ilişkin bir oyun” adlı yüksek lisans çalışmasında, elektrik piyasasını oyunlaştırarak, piyasa üzerinde analizlerin yapılabilmesini sağlamıştır. Piyasa oyuncularının davranışlarını ayırık kümeleyerek stratejik kümede çarpanlarla işlemiştir. Ayrıca birden fazla piyasa senaryoları farklı oyun değişkenleriyle birlikte test etmiştir. 2008 yılındaki ayrı dört zaman dilimi için Türkiye Elektrik Piyasasının gerçek verilerini kullanarak kümeleme yapıp, oyun senaryolarına ilişkin oyun çıktıları karşılaştırılmıştır. Senaryoların akışına göre yeterli rekabet olması, olmaması ya da denge durumuna göre yüksek ya da düşük fiyatlı yük alma, yük atma teklifleri verilmesi tavsiyeleri bulunmaktadır [36].

2.2.2.4. Türk elektrik piyasası simülasyon oyunu

Alp Eren Aydın, Türk elektrik piyasasını bazalarak yaptığı yüksek lisans tezi çalışmasında oyun yapısını kullanmış olup, Türkiye elektrik piyasasının işleyişini, ticaret temellerini, rekabet ortamını öğreterek gerçek market ortamında egzersiz yapabilme imkanını sunmuştur. Çalışmada Türk elektrik piyasasında gerçekleşen senaryoları gerçekleştirebilen yeteneklere benzer işleyişler mevcuttur. Katılımcılara gerçek piyasadaki elektrik santrallerine yatırım yaparak sahip olup ticaret yapabilmektedir [37].

SIMULATION TRAINING for TURKISH ELECTRICITY MARKET

ADONIS GENERAL PART 1 PART 2 PART 3 PART 4

Sign-off General Contract Retail Sales Hourly Price Report Dispatch Schedule

Company: Celta Energy

User: Alihan Aydin

Game: ProfitGame

Round: 1

Part: 1

RETAIL CUSTOMER CONTRACT MANAGEMENT

Retail customer contract management will be done in this page.

[ALL CUSTOMER CONTRACTS](#) [see page](#)

[Historical consumption for all customers \(round 1 to 52\)](#)

This is all-in-one screen for you to monitor all of the accepted contracts with retail customers. You are not allowed to see any offers issued by your competitors, but you can always come and visit this page to see which customers were hooked up. This might be a good indication while identifying profitable customers. Besides, different background color in the following table indicates that:

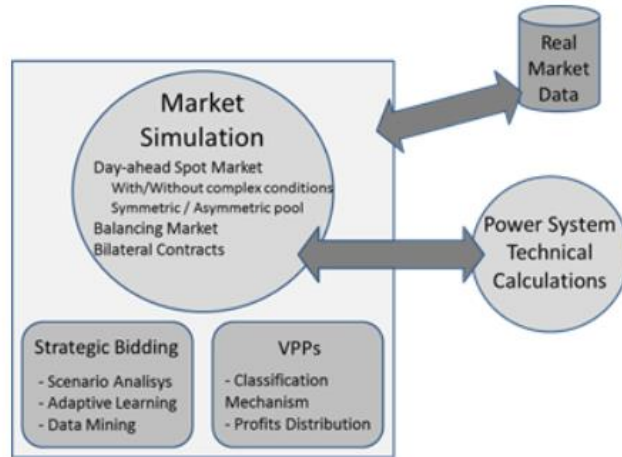
- Not contracted yet
- Has an accepted contract with other company
- Has an accepted contract with your company

Customer	T1 Price	T2 Price	T3 Price	13	14	15	16	17	18	19	20	21	22	23	24
The Don	250.00	250.00	250.00												
The Stud	250.00	250.00	250.00												
AutoFingers	250.00	250.00	250.00												
The Schmeer	250.00	250.00	250.00												
The Ridder	250.00	250.00	250.00												
Gidde	250.00	250.00	250.00												
Big Jen	250.00	250.00	250.00												
The Bully	250.00	250.00	250.00												
The Straky	250.00	250.00	250.00												
Jony Mack	250.00	250.00	250.00												
Indian Bully	250.00	250.00	250.00												
The Dark Prince	250.00	250.00	250.00												

Şekil 2.11. Müşterilerin kontratlarının görüldüğü sayfa [37].

2.2.2.5. Mascem (Rekabetçi elektrik piyasaları için simülatör)

Mascem, 4-5 üniversitenin konsorsiyum ile ortaya çıkmış elektrikpiyasa simülatörüdür. Çalışmada komplekselektrik piyasasını modelleyerek, dinamik yapıdaki piyasa bileşenlerini, piyasa oyuncularını orta/uzun vadede oluşan kullanıcı verilerini karar destek mekanizmasıyla, karakteristik yapıları sunan sanal piyasa yapısı oluşturulmuştur. Piyasa akışında, piyasa yöneticisinin yetkisinde bulunan, alıcı ve satıcıların teklifleri bir havuza düşer, piyasa fiyatı oluşur, takas fiyatına göre eşleşmeler, olumlu veya olumsuz sonuçlanır [38].

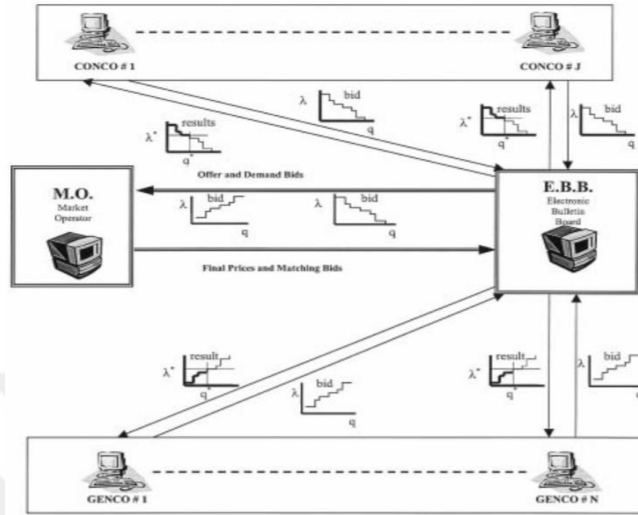


Şekil 2.12. Mascem mimarisi [39].

2.2.2.6. Enerji mühendisliği laboratuvarı: Elektrik piyasası simülatörü

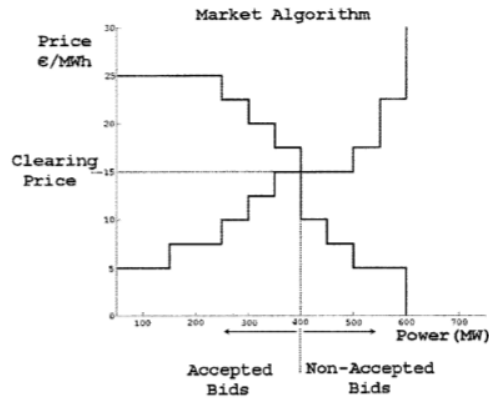
Conejove arkadaşlarınınIEEE destekli Elektrik piyasa simülatörü geliştirilmesi için yaptıkları çalışmada MATLAB üzerinden okunabilen dosya formatları yardımıyla İspanya elektrik piyasasından esinlenerek, lisans öğrencileri için senaryolar oluşturularak çıktıları incelenmiştir. Saatlik MWh cinsinden tekliflerin, beşli öğrenci gruplarından

gelen taleplere göre piyasa operatörü piyasa takas fiyatını belirler. Piyasada oluşan fiyatlandırma ve buna bağlı oluşan arz fazlası veya dengesizliği gösteren sonuçları kullanıcılar, simülasyon yardımıyla öğrenebilmektedir [40].



Şekil 2.13. Simülasyon akış mimarisi [40].

Şekil 2.13'te bahsi geçen MO piyasa operatörü, GENCOs üreticiler, CONCOs tüketiciler olup piyasa akış ve işleyişini gösterir.



Şekil 2.14. Piyasa fiyat eşleşme grafik [40].

3. TEZ ÇALIŞMASI

Yeniden yapılanan elektrik sektöründe bu pazara katılacak elektrik elektronik mühendisleri veya elektrik mühendislerinin eğitimi de farklı bir boyut kazanmaktadır. Çalışacakları yeni pazarda sektörün yeniden yapılanması gereği oluşan farklılıklar bulunmaktadır. Lisans ve lisansüstü düzeyde yapılacak katılımlarda verilecek derslerin bu kapsamda yenilenmesi ve iyileştirilmesi bir zorunluluktur.

Özellikle Türkiye Elektrik Piyasasındaki yeni durumu ve farklılıkları öğrenciye öncede kavratmak ve uygulatabilmek amacı ile eğitim amaçlı elektrik piyasası simülatörünün tasarımı bu çalışmada hedeflenmiştir. Böylece öğrencilere Türkiye elektrik piyasasının temelini öğretmek ve onlara gerçek market ortamında alıştırma yapma şansını vermek amaçlanmıştır.

Elektrik piyasası simülatörü ile gerçek bir elektrik piyasasındaki gerçekleştirilebilen tüm işlemlerin yapılması hedeflenmiştir. Yeniden yapılanma sürecinin yaşandığı Türkiye elektrik piyasası ile ilgili lisans ve lisansüstü kapsamlı bilgi kaynağını artırmak, ayrıca piyasa yapısı içinde bulunan yeni katılımcılara katkı yaparak, Türkiye elektrik piyasasının karmaşık yapısı ve bu piyasadaki güncel değişiklikler hakkında bilgi sahibi olabilmesini sağlamaktır. Böylece katılımcılar, Türkiye elektrik piyasasını değişik perspektiflerden gözlemlene şansı bulacaktır. Geliştirilecek simülatör ile katılımcılar gerekli yatırım adımlarını atarak elektrik tedarikçisi olabilecekve gerçek piyasa ile tamamen aynı olan elektrikticareti işlemlerini yapabilmesi sağlanacaktır. Böylece herhangi bir risk almadan Türkiyeelektrik piyasası ile ilgili ciddi bir deneyime sahip olabilecektir. Türkiye elektrik sektörü, yeniden yapılanma süreci sonrası elektrik piyasalarının öncülüğünde serbestleşme yönünde kararlı adımlarla ilerlemektedir. Piyasalara katılım oranlarının düşük olması ve buna bağlı altyapıdaki bazı eksiklikler de bulunmaktadır. 2023 hedeflerini de destekleyen yakın coğrafya için bir enerji merkezi olma ve enerji açığını azaltmak gibi EPIAŞ'ın kurulumu, bölge enerji borsası oluşturmak isteyen Türkiye için çok önemli bir başlangıçtır. Saatlik işlem hacminin üç milyon dört yüz binin bulduğu bu Pazar yıllık işlem hacmi ile Avrupa'da önemli bir yer almaktadır. Gelecekte bölge ülkelerinin de katılımı ile Avrupa'daki EEX (The European Energy Exchange), EPEX (European Power Exchange), PXE (Power Exchange Central Europe), Nord Pool vb. birçok spot piyasa gibi daha önemli bir piyasa olacaktır [41].

Bu çalışma, Türkiye elektrik piyasasında işlem yapmakta olan şirketlere yeni katılanlara, özellikle elektrik elektronik mühendisliği ve elektrik mühendisliği lisans ve lisansüstü öğrencilerinin Türkiye elektrik piyasası hakkında bilgi sahibi olması ve uygulama yapması noktasında yardımcı olacaktır.

3.1. Piyasa Simülatörü

Konu ile ilgili kitaplar, süreli yayınlar, elektronik ortamda bulunan dokümanlar, kongre kitapları ilgili kurum ve kuruluşlar tarafından hazırlanmış teknik raporlar, proje ve dokümanlar, kullanılacak paket program firmalarının eğitimleri ve dokümanları incelenmiş olup belli bir algoritmaya bağlı kalarak, elektronik ortamda simülatörün gerçekleştirilmesi sağlanmıştır.

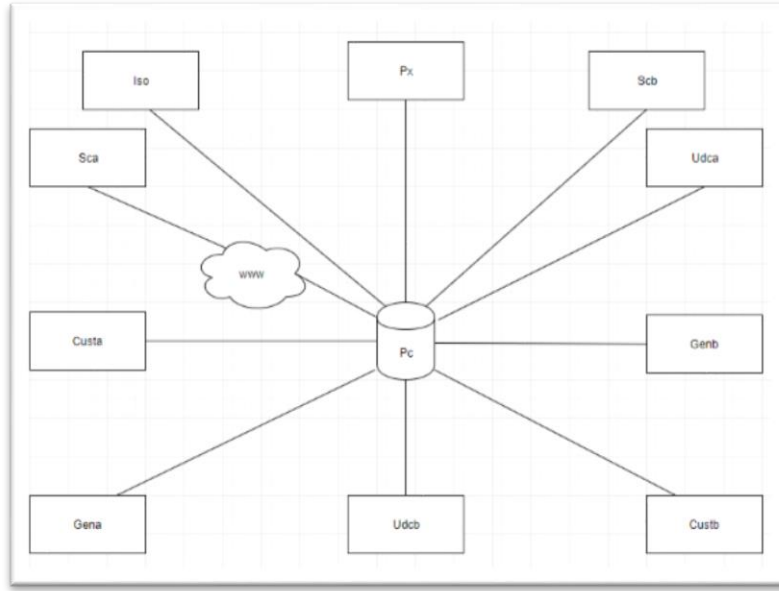
Gerçek zamanlı kontrolü gerçekleştirmeyi sağlayacak düzeyde donanıma sahip bir adet sunucu bilgisayar masa üstü bilgisayar, akabinde entegre edilebilecek üçüncü parti analiz ve performans ölçücü programlar (Matlab, Nesneye dayalı programlama tekniği kullanılarak yazılabilecek performans ölçücü programlar; Java, C# vb.)

Projemizin ismi : Elektrik Piyasa Simülatörü

3.1.1. Simülatör mimarisi

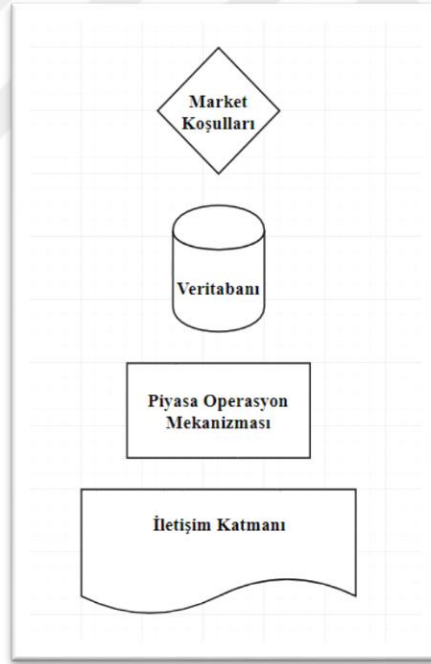
Piyasa simülatörü dört katmana ayırarak olunursa, en çok öneme sahip nokta piyasayla katılımcılar arasındaki bağlantı ve koordinasyondur. Bu yönüyle bakıldığında simülatörün iletişim yani sistemler arasında konuşmayı sağlayan yapı olması sebebiyle çoklu kullanıcı sistemi multi-agent yapısını içerir. Piyasada bulunanlar piyasa operatörü tarafından yönetilir.

İkinci katmanda ise; farklı sınıftaki kullanıcıların ve operatör sisteminin olduğu aşamadır.



Şekil 3.1. Piyasa katılımcıları internet aracılığıyla sisteme entegre olurlar [42].

Şekil 3.1’de görüldüğü üzere her bir kutucuk piyasaya katılan ve etkileşim çizelgesini göstermektedir. Diyagramda yani simülâtörde birden fazla veriden beslenmektedir. Tüm bileşenler ortak veri tabanına erişim sağlayarak işlemlerini gerçekleştirebilmektedir.

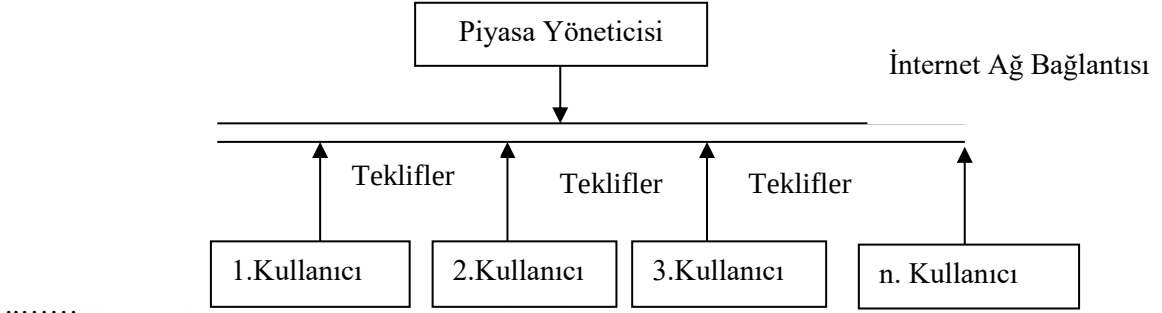


Şekil 3.2. Katmanlı mimari [42].

Elektrik piyasası simülâtörü ile çeşitli aşamalardan oluşacaktır. Oyunun tüm aşamaları kısaca nitelendirilebilecek bir zaman diliminde tamamlanacaktır. Elektrik piyasası simülâtörünün tasarımı ve tasarım sonrası aşamaları aşağıda belirtildiği gibi gerçekleştirilecektir.

- Gün Öncesi Piyasası verilerinin oluşturulması

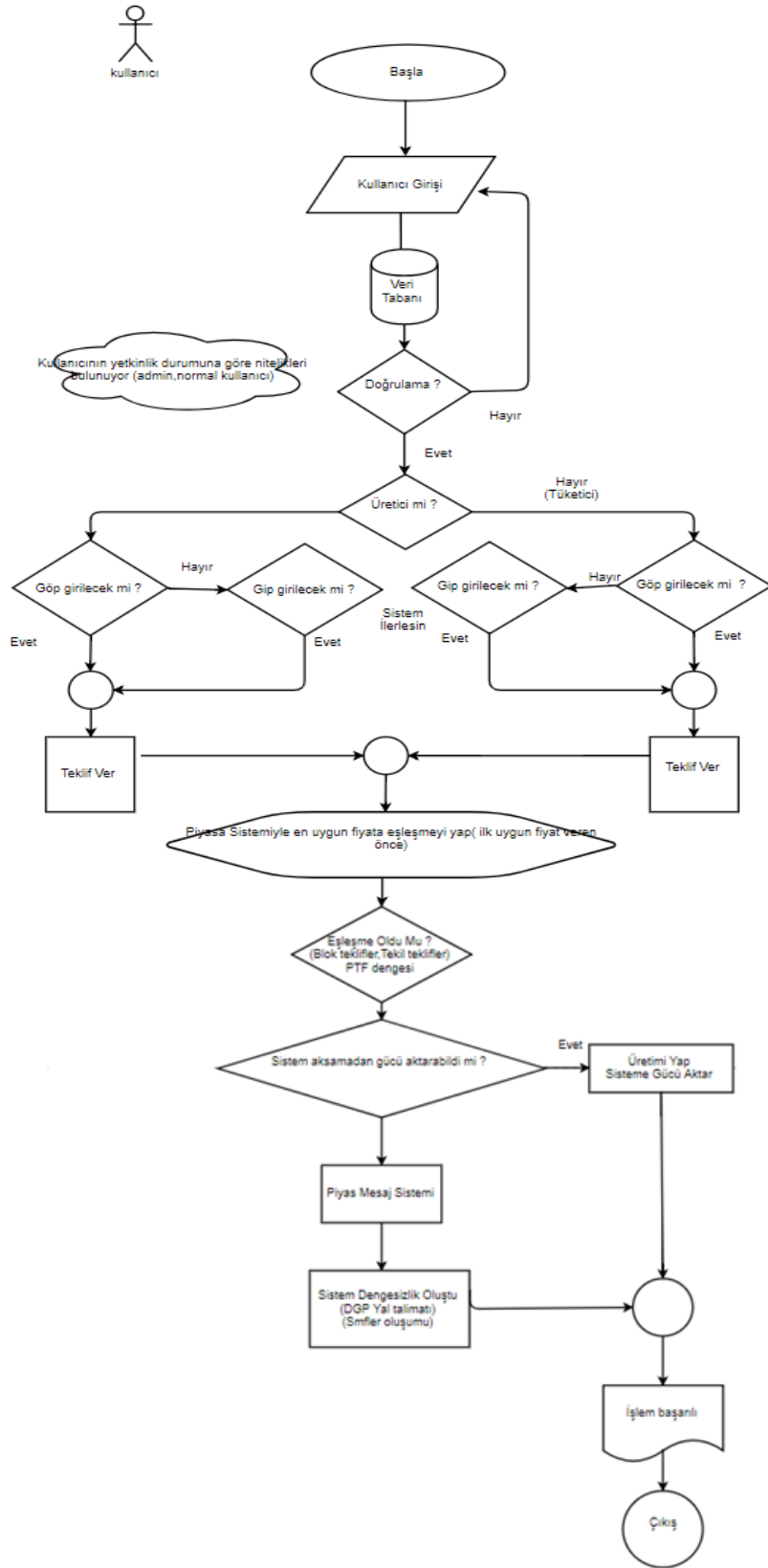
- Gün içi piyasasının oluşturulması
- Dengeleme güç piyasası oluşturulması
- Platform Microsoft. Net teknolojisi,veritabanı SQL server,MySQL üzerinde kodlanması



Şekil 3.3. Sunucu/istemci mimarisi [43].

3.1.2. Akış diyagramı

Sistemin akışını yürüten mimari aşağıdaki akışa uyacak edecek şekilde olacaktır.



Şekil 3.4. Elektrik Piyasası Akış Diyagramı

3.2. Simülatör Temel Bileşenleri

Simülatör tasarımını üç ana bileşene ayırabiliriz.

- Kullanıcıların birebir etkileşimde olacağı Microsoft .net /Form teknolojisinin kullanılacağı ara yüz.
- Simülatörün omurgası diyebileceğimiz senaryoları işleten bölüm 4.1.2.'de verdiğimiz akış diyagramı
- Girilecek olan teklif, kullanıcı bilgileri oluşan piyasa takas fiyatları vb değerlerinin tutulması, işlenmesi ve güncellenmesi için gerekli olan veri tabanı ve veri tabanı dizaynının geliştirilmesi.

3.2.1. Programlama araçları

Günümüzde yüzlerce programlama dili bulunuyor. Her biri farklı amaçlar doğrultusunda optimum düzeyde performans sağlamaktadır. Bu çalışmada, internet bağ web server dediğimiz bir adet Linux sunucu, masaüstü uygulamasını geliştireceğimiz .net framework mimarisinin ferdi olan C# dili ve MsSql veri tabanı yönetimi kullanılmaktadır.

3.2.2. C# dili ve .net mimarisi

C#, .NET Framework üzerinde koştan çeşitli güvenli ve verimli uygulamalar oluşturmaya olanak sağlayan güçlü ve güvenli nesne tabanlı bir dildir. Windows uygulamaları, XML, dağıtılmış bileşenler, istemci-sunucu uygulamaları, veritabanı uygulamaları ve çok daha fazlasını oluşturmak için C# dili kullanılmaktadır. Özellikle c# ve .net framework'e dayanan uygulamalar, yazılım geliştirmeyi kolaylaştırmak için gelişmiş kod düzenleyici, arabirim tasarımları, tümleşik hata ayıklayıcıları ve diğer birçok aracı sağlamaktadır.

Nesneye yönelimlidir olan c#, kapsülleme, kalıtım ve polimorfizm kavramlarını desteklemektedir. Uygulamaların başlangıç noktaları dahil tüm değişkenler ve yöntemler, sınıf tanımları içinde kapsüllenebilmektedir. Örnek bir sınıf doğrudan bir ana sınıftan devralabilir, ancak herhangi bir sayıda arabirimi uygulayabilir. Bir üst sınıftaki sanal yöntemleri geçersiz kılan yöntemler, yanlışlıkla yeniden tanımlamayı önlemek için geçersiz kılma anahtar sözcüğünü gerektirir. C# programları, ortak dil çalışma zamanı (CLR) ve birleşik sınıf kütüphaneleri kümesi olarak adlandırılan bir sanal yürütme sistemi

içeren, Windows'un bir parçası olan .NET Framework üzerinde çalışır. CLR, dillerin ve kütüphanelerin sorunsuz bir şekilde birlikte çalıştığı yürütme ve geliştirme ortamlarının oluşturulmasına temel oluşturan uluslararası bir standart olan ortak dil altyapısı (CLI) Microsoft'un ticari uygulamasıdır.

C # ile yazılan kaynak kodu CLI spesifikasyonuna uygun bir ara dilde (IL) derlenmiştir. IL kodları ve bitmapler ve dizeler gibi kaynaklar, diskte, genellikle .exe veya .dll uzantısıyla bir derleme adı verilen yürütülebilir dosyada depolanır. Bir derleme, derlemenin türleri, sürümü, kültürü ve güvenlik gereksinimleri hakkında bilgi veren bir bildirim içerir.

C # programı çalıştırıldığında, derleme, bildirimde yer alan bilgilere dayanarak çeşitli eylemler gerçekleştirebilen CLR'ye yüklenir. Ardından, güvenlik gereksinimleri karşılanırsa, CLR, IL kodunu yerel için tam zamanında (JIT) derleme gerçekleştirir. CLR ayrıca otomatik çöp toplama, istisna işleme ve kaynak yönetimi ile ilgili diğer hizmetleri de sağlar. CLR tarafından yürütülen kod, belirli bir sistemi hedefleyen yerel makine diliyle derlenen "yönetilmeyen kod" ın aksine bazen "yönetilen kod" olarak adlandırılır. Aşağıdaki şema C # kaynak kod dosyalarının, .NET Framework sınıf kitaplıklarının, derlemelerinin ve CLR'nin derleme zamanı ve çalışma zamanı ilişkilerini göstermektedir [44].

3.2.3. MsSql ve mysql veri tabanı sistemi

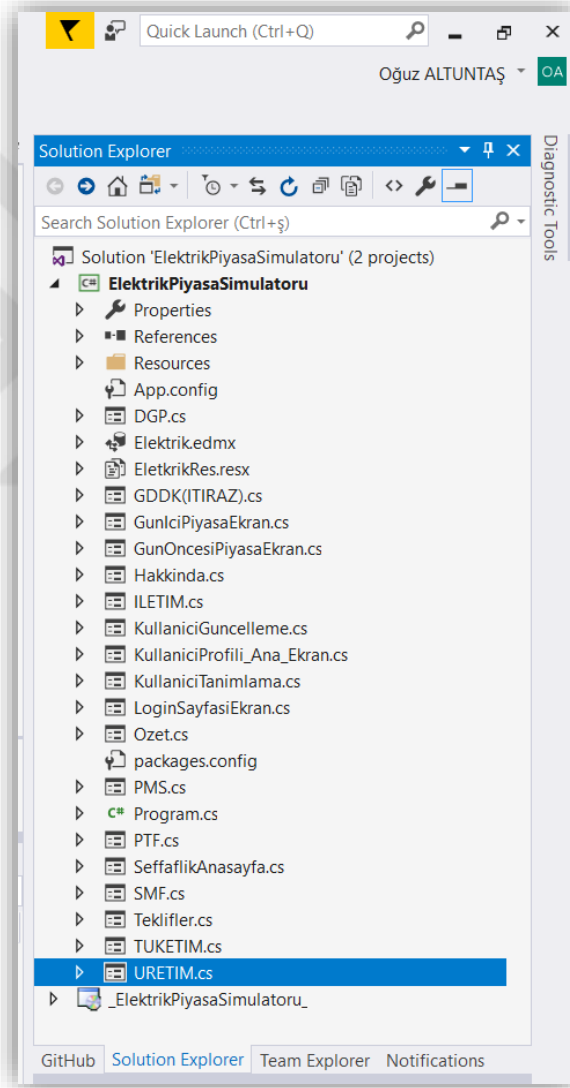
İçinde istenilen türde verileri tutan ve bu verilere erişim imkânı sağlayabilen, verilerle ilgili tüm manipölasyon işlemlerini yapabilme imkanını sunan bilgiler kümesi veri tabanı olarak nitelendirilmektedir.

MySQL, isteklere hızlı ve efektif bir şekilde cevap verebilen, çok kullanıcıli veri tabanı yönetim sistemidir. Windows, Unix ve OS/2 gibi işletim sistemleri üzerinde çalışabilen ücretsiz versiyonları mevcut olmasının yanında ticari maksatla kullanılmak üzere ücretli lisans seçeneği de mevcuttur. Mysql açık kaynak kodlu olmasının yanında ODBC gibi farklı veri tabanı türlerine belli bir standartla erişilebilir sürücüler de bulunduğu için farklı geliştirme platformlarındaki kullanılabilmektedir. Mysql ile birçok programlama türünde olduğu gibi web programlamada da verileri sunucular üzerinde eklenebilir ve güncellenebilir ve silinebilir [45].

3.3. Uygulama Geliştirme

3.3.1. Platform oluşturma

Simülâtör ekran bazlı bir uygulama olacaktır. Bu yönüyle dashboard dediğimiz bir platform üzerine çoklu ekran işlevi eklenmesiyle birlikte sistem tamamlanmış olacak. Platformumuz birçok program gerçekleşmesine dil desteği bulunan Microsoft Visual Studio IDE’sinde 2017 versiyonunda geliştirildi. Proje Elektrik Piyasa Simülâtörü şeklinde adlandırıldı.



Şekil 3.5. Platform tabanının oluşturulması

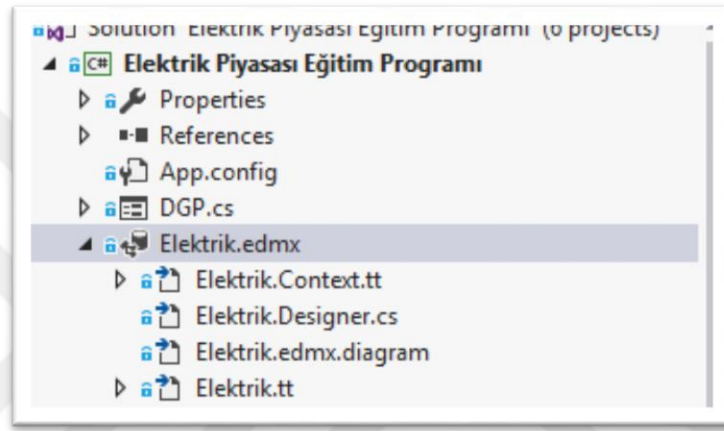
3.3.2. Veri tabanı mimarisi ve yapısı

Verilerin tutulacağı ve manipüle edileceği yer olarak da Microsoft ürünü olan Mssql 2016

database'i kullanılacaktır. Veri tabanı kısmında piyasada bulunan matematiksel işlemleri yapan ve işleyen mekanizma yürütülüyor. Sistemdeki tüm veriler burada tutuluyor. Stored procedure'lerle birlikte tekliflerin eşleşmesi, ptf gibi fiyatlar oluşturuluyor.

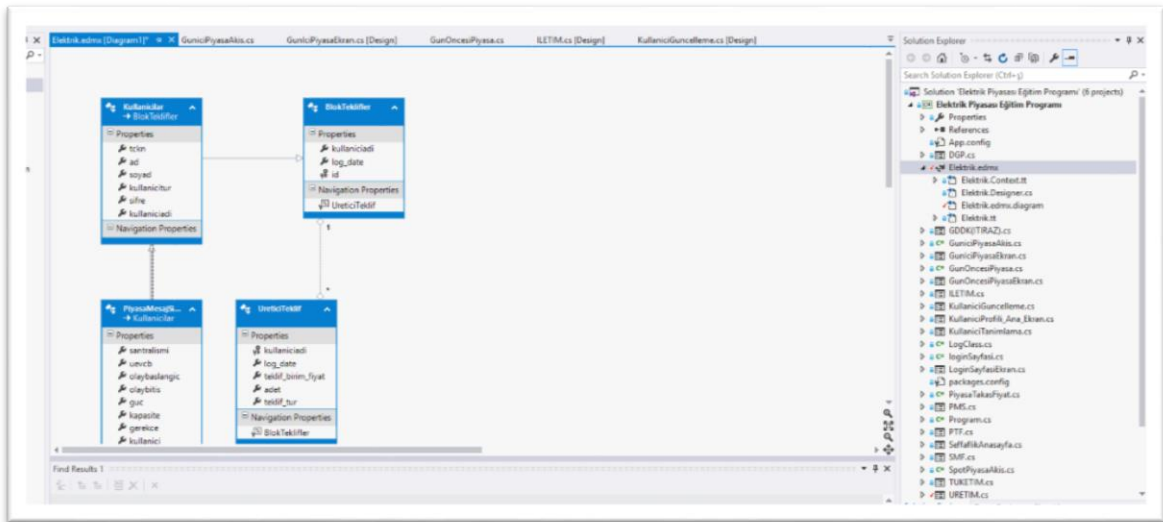
(Kullanıcı bilgileri, anlık ve geçmiş piyasa bilgileri, alım, satım işlemleri...)

C# winform projesi (Elektrik Piyasası Eğitim Programı) ile database arasındaki bağlantıyı sağlayan katman entity framework mimarisidir. Bu mimari ile birlikte verilerin saklanması, güncellenmesi ve manupule edilmesi sağlanıyor. Yapı .edmx uzantısıyla birlikte projede konumlandırılıyor.



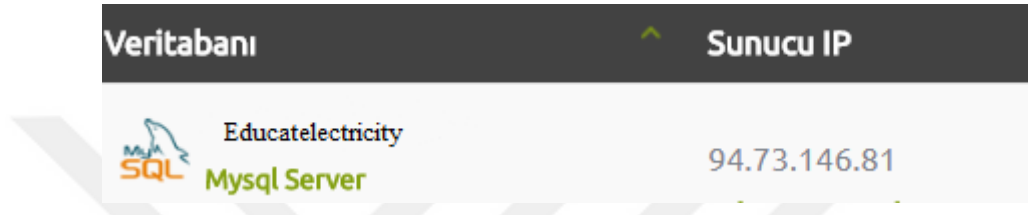
Şekil 3.6. Visual studio database mimarisinin entegre edilmesi

Geliştirme ortamında IDE'miz Visual Studio'da OOP (Object-oriented programming) nesneye yönelimli olarak programladığımız simülatörün veri tabanı ilişkilerini barındırdığı yapı şeklindeki gibi olacaktır.



Şekil 3.7. Tablolar arası ilişkileri gösteren uml diyagramlar

Veri tabanı ileride daha da geliştirilebilir bir yapı olması amacıyla Mysql ve Mssql tabanlı geliştirildi. Gerçek ortamda da ulaşılabilir olması için sunucu kiralanarak mysql veri tabanına entegrasyonu ve izinleri alındı. Mysql veri tabanı Educatelectricity olarak adlandırıldı. Bu veritabanının içersinde 12 adet tablo bulunmaktadır. Sırasıyla bu tablolar aşağıda belirtilmiştir. Tabloların alan ve yapısal tipleri değiştirilip geliştirebilir seviyededir.



Şekil 3.8. Veri tabanı

kullanıcılar tablosunun alanları Şekil 3.9’da gösterilmektedir. “kullaniciadi” alanı benzersizdir. ”tckn” alanı bir kullanıcının birden fazla hesabı olabileceğinden tekil değildir ve kimlik numarası ile kayıt yapılır ve 11 karakter kadar uzunluk ayrılmıştır.”ad” alanı kullanıcının ismi, ”soyad” alanı kullanıcı soyisminin,”kullanicitur” kullanıcının tipini admin, üretici veya tüketici gibi ve “sifre” alanı kullanıcıya özgü şifrenin tutulduğu alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
☐	1 tckn	char(11)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	2 ad	char(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	3 soyad	char(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	4 kullanicitur	varchar(10)	utf8_general_ci		Evet	NULL		Değiştir Kaldır Daha Fazla ▼
☐	5 sifre	varchar(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	6 kullaniciadi	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼

Şekil 3.9. Kullanıcılar tablosu

“kullanicitip” tablosunda kullanıcı türü ve türlerinin tanımlarının tutulacağı referans tablodur. “KulaniciTip” alanında hesap türleri tutulmaktadır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	KullaniciTip	varchar(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼

Şekil 3.10. Kullanicitip tablosu

“ureticiteklif” tablosunda üretici tipli kullanıcıların piyasaya sundukları teklifleri tutan tablodur.”kullaniciadi” teklif i veren kullanıcı isminin tutulduğu alandır. ”log_date” kullanıcı teklifinin verildiği tarih bilgisi bulunur.”teklif_birim_fiyat” 1 lot (0,1 MWh) için verilen birim fiyatı tutar.”adet” kaç adet lot için teklif verildiğini tutar. “teklif_tur” blok veya normal teklif ayrımı için kullanılan bir alandır. “saat” teklif verilen saati tutar. “durum” eslesti, aktif, pasif, iptal seçenekleri tutar. ”eslesenteklifler” eşleşen tekliflere ait olan id numaralarını tutan alan.”id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	kullaniciadi	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
2	log_date	date			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
3	teklif_birim_fiyat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
4	adet	varchar(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
5	teklif_tur	varchar(6)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
6	saat	varchar(2)	utf8_general_ci		Evet	NULL		Değiştir Kaldır Daha Fazla ▼
7	durum	varchar(10)	utf8_general_ci		Evet	NULL		Değiştir Kaldır Daha Fazla ▼
8	eslesenteklifler	varchar(50)	utf8_general_ci		Evet	NULL		Değiştir Kaldır Daha Fazla ▼
9	id	int(6)		UNSIGNED	Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.11. Ureticiteklif tablosu

“tuketiciteklif” tablosu tüketici konumunda olan kullanıcıların teklif süreçlerinin işlendiği tablodur.”kullaniciadi” teklif i veren kullanıcı isminin tutulduğu alandır. ”log_date” kullanıcı teklifinin verildiği tarih bilgisi bulunur.”teklif_birim_fiyat” 1 lot (0,1 MWh) için verilen birim fiyatı tutar.”adet” kaç adet lot için teklif verildiğini tutar. “teklif_tur” blok veya normal teklif ayrımı için kullanılan bir alandır. “saat” teklif verilen saati tutar. “durum” eslesti, aktif, pasif, iptal seçenekleri tutar. ”eslesenteklifler” eşleşen tekliflere ait olan id numaralarını tutan alan.”id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	kullaniciadi	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
2	log_date	date			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
3	teklif_birim_fiyat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
4	adet	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
5	teklif_tur	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
6	saat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
7	durum	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
8	eslesenteklifler	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
9	id	int(11)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.12. Tuketici teklif tablosu

“ureticiblokteklifler” tablosunda üretici tipli kullanıcıların piyasaya sundukları teklifleri tutan tablodur. Blok teklif yapısının en önemli ayırt edici özelliği teklif tamamıyla eşleşir ya da eşleşmez yani kısmen eşleşme diye bir durum söz konusu değildir.”kullaniciadi” teklif i veren kullanıcı isminin tutulduğu alandır. ”log_date” kullanıcı teklifinin verildiği tarih bilgisi bulunur.”teklif_birim_fiyat” 1 lot (0,1 MWh) için verilen birim fiyatı tutar.”adet” kaç adet lot için teklif verildiğini tutar. “teklif_tur” blok veya normal teklif ayrımı için kullanılan bir alandır. “saat” teklif verilen saati tutar. “durum” esleşt, aktif, pasif, iptal seçenekleri tutar. ”eslesenteklifler” eşleşen tekliflere ait olan id numaralarını tutan alan.”id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	kullaniciadi	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
2	log_date	date			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
3	teklif_birim_fiyat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
4	adet	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
5	teklif_tur	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
6	saat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
7	durum	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
8	eslesenteklifler	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
9	id	int(11)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.13. Ureticiblokteklifler tablosu

“tuketici blokteklifler” tablosu tüketici konumunda olan kullanıcıların teklif süreçlerinin işlendiği tablodur. Blok teklif yapısının en önemli ayırt edici özelliği teklif tamamıyla eşleşir ya da eşleşmez yani kısmen eşleşme diye bir durum söz konusu değildir.”kullaniciadi” teklif i veren kullanıcı isminin tutulduğu alandır. ”log_date” kullanıcı teklifinin verildiği tarih bilgisi bulunmaktadır. ”teklif_birim_fiyat” 1 lot (0,1

MWh) için verilen birim fiyatı tutar.”adet” kaç adet lot için teklif verildiğini tutar. “teklif_tur” blok veya normal teklif ayrımı için kullanılan bir alandır. “saat” teklif verilen saati tutar. “durum” esleştiri, aktif, pasif, iptal seçenekleri tutar. ”eslesenteklifler” eşleşen tekliflere ait olan id numaralarını tutan alan.”id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	kullaniciadi	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
2	log_date	date			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
3	teklif_birim_fiyat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
4	adet	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
5	teklif_tur	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
6	saat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
7	durum	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
8	eslesenteklifler	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
9	id	int(11)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.14. Tuketiciblokteklifler tablosu

“ptfkayıt” tablosunda piyasa takas fiyatları tutulur. ”date” alanında günün tarih bilgisi tutulur. ”saat” alanında ptf’nin saatlik değeri tutulur. “PTF” alanında piyasa takas fiyatı tutulur. “SMF” alanında sistem marjinal fiyat bilgisi tutulmaktadır. “AOT” alanında ağırlıklı ortalama fiyat bilgisi tutulur. ”id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	date	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
2	saat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
3	PTF	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
4	SMF	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
5	AOT	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
6	id	int(50)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.15. Ptfkayıt tablosu

“piyasamesajsistemi” tablosunda piyasada bulunan üreticilerin anlık arızalarını vs tüm piyasa oyuncularına bildirebileceği duyuruları barındıran tablodur.”santralismi” arızanın olduğu santral isminin tutulduğu alandır.”uevcb” arızanın meydana geldiği santral.”olaybaslangic” arızanın başladığı tarihi simgelemektedir..”olaybitis” arızanın bittiği tarih. “guc” işletmedeki güç. “kapasite” olay sırasındaki kapasitedir. “gerekce” sorunun gerekçesi. “kullanici” hangi kullanıcı için arıza olduğunun bilgisi. ”id” benzersiz

ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
☐	1 santralismi	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	2 uevcb	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	3 olaybaslangic	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	4 olaybitis	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	5 guc	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	6 kapasite	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	7 gerekce	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	8 kullanıcı	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	9 [id]	int(11)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.16. Piyasamesajsistemi tablosu

“sistemyonu” anlık piyasa enerji açığı veya fazlasının olup olmadığı bilgisini tutan tablodur. “log-date” günün tarih bilgisini tutmaktadır. “saat” saat bilgisini tutar. “sistemYonu” piyasanın enerji açığı veya fazlası olup olmadığı bilgisi tutulmaktadır. “id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
☐	1 log-date	date			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	2 saat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	3 sistemYonu	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
☐	4 id	int(11)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.17. Sistemyonu tablosu

“eslesenteklifler” tablosunda üretici ve tüketicilerin verdiği tekliflerden piyasa takas fiyatına göre uygun olanların eşleşip kayıtlarının atıldığı yerdir. “ureticiciID” üreticinin kullanıcı adını tutar.”tuketiciciID” tüketicinin kullanıcı adını tutar. “eslesenmiktar” eşleşen miktarı tutar. “fiyat” eşleşme yapılan fiyatı tutar. “gun” eşleşme yapılan gün bilgisini tutar. “saat” eşleşme yapılan saat bilgisini tutar. “id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.”U_adet” üreticinin toplam verdiği teklif adetini tutmaktadır. “T_adet” tüketicinin verdiği toplam teklif adetini tutmaktadır. “eslesmeID” eşleşmeye özel bir id üretilir ve bu alana ait değer tutulmaktadır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	ureticilID	char(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
2	tuketicilID	char(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
3	eslesenmiktar	char(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
4	fiyat	varchar(10)	utf8_general_ci		Evet	NULL		Değiştir Kaldır Daha Fazla ▼
5	gun	varchar(10)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
6	saat	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
7	id	int(50)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼
8	U_adet	int(50)			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
9	T_adet	int(50)			Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
10	eslemeID	varchar(50)	utf8_general_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼

Şekil 3.18. Eşlesenteklifler tablosu

“itiraz” tablosu herhangi bir teklif ya da eşleşmeye dair olacak olan itirazları tutan tablodur. “gerekce” itiraz edilen konuyu tutan alandır. “kullanici” hangi kullanıcının itiraz ettiği bilgisini tutan alandır. ”id” benzersiz ve Auto Increment (Otomatik Artma) niteliği bulunan alandır.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	gerekce	varchar(50)	latin1_swedish_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
2	kullanici	varchar(50)	latin1_swedish_ci		Hayır	Yok		Değiştir Kaldır Daha Fazla ▼
3	[id]	int(11)			Hayır	Yok	AUTO_INCREMENT	Değiştir Kaldır Daha Fazla ▼

Şekil 3.19. İtiraz tablosu

“entsokod” tablosu eic(enerji tanımlama kodu) kodlarını tutan tablodur. “EIC” eic kodunu tutan alandır. “OrganizasyonKod” koda karşılık gelen organizasyon isimlendirmesidir.

#	Sütun	Türü	Karşılaştırma	Öznitelikler	Boş	Varsayılan	Ekstra	Eylem
1	EIC	varchar(60)	utf8_general_ci		Hayır			Değiştir Kaldır Daha Fazla ▼
2	OrganizasyonKod	varchar(60)	utf8_general_ci		Evet	NULL		Değiştir Kaldır Daha Fazla ▼

Şekil 3.20. Entsokod tablosu

3.3.3. Giriş ekranı(LoginSayfasiEkran.cs)

Kullanıcıyı karşılayan ilk ekran sisteme hoş geldin diye tabir edebileceğimiz login ekranıdır. Kullanıcının kullanıcı adı ve şifresi ile birlikte girişini yapabileceği inputları kabul eden iki textbox, kontrolü sağlayacak olan Giriş adlı buton ve program ile ilgili bilgileri getirecek olan Hakkında menustrip bulunuyor.

Elektrik Piyasası Simülatörü

Hakkında

Kullanıcı Adı :

Şifre :

Giriş

Şekil 3.21. Ön yüz http protokolünden erişilebilen aspx ve winform(C#) teknolojisiyle kodlanması(Kullanıcı giriş ekranı

Kullanıcı adı ve şifreden biri hatalı girilmesi durumunda kullanıcı ekranda aşağıdaki mesajı görmektedir.

Elektrik Piyasası Simülatörü

Hakkında

Kullanıcı Adı : oguz

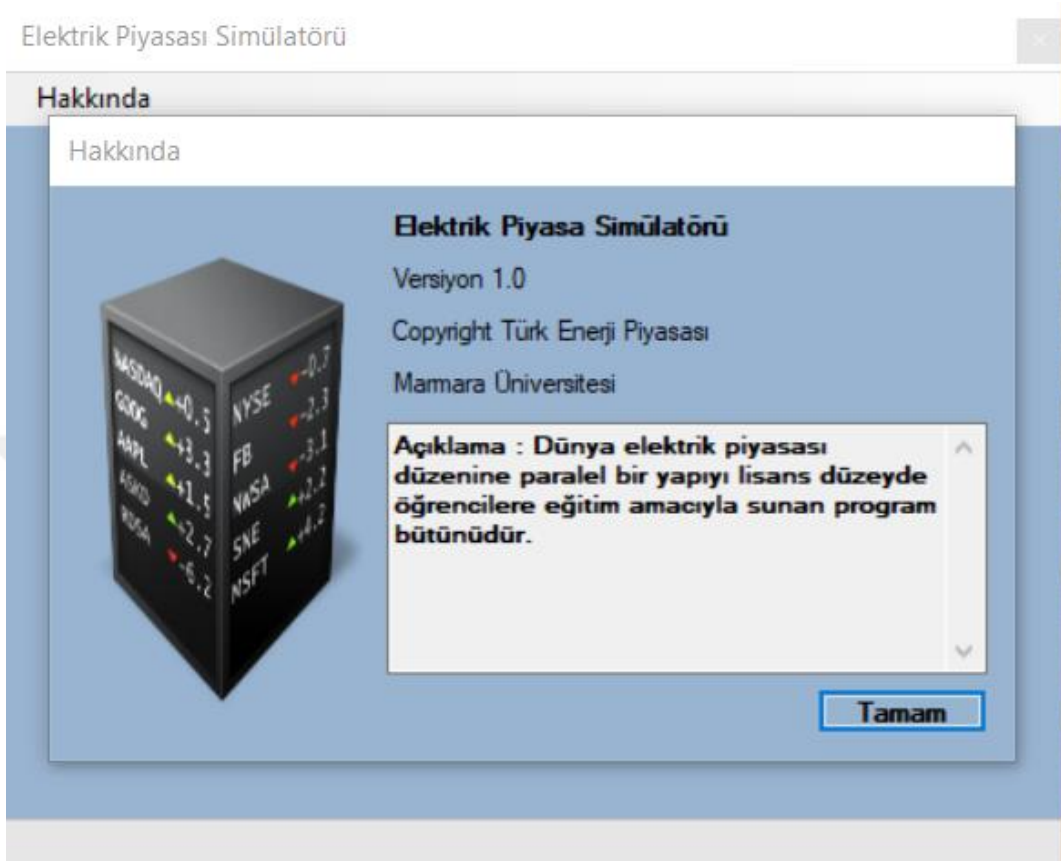
Şifre : xxxx

Hatalı Kullanıcı adı/Şifresi !

Tamam

Şekil 3.22. Kullanıcı ad veya şifresinin yanlış girilmesi durumunda verilen uyarı

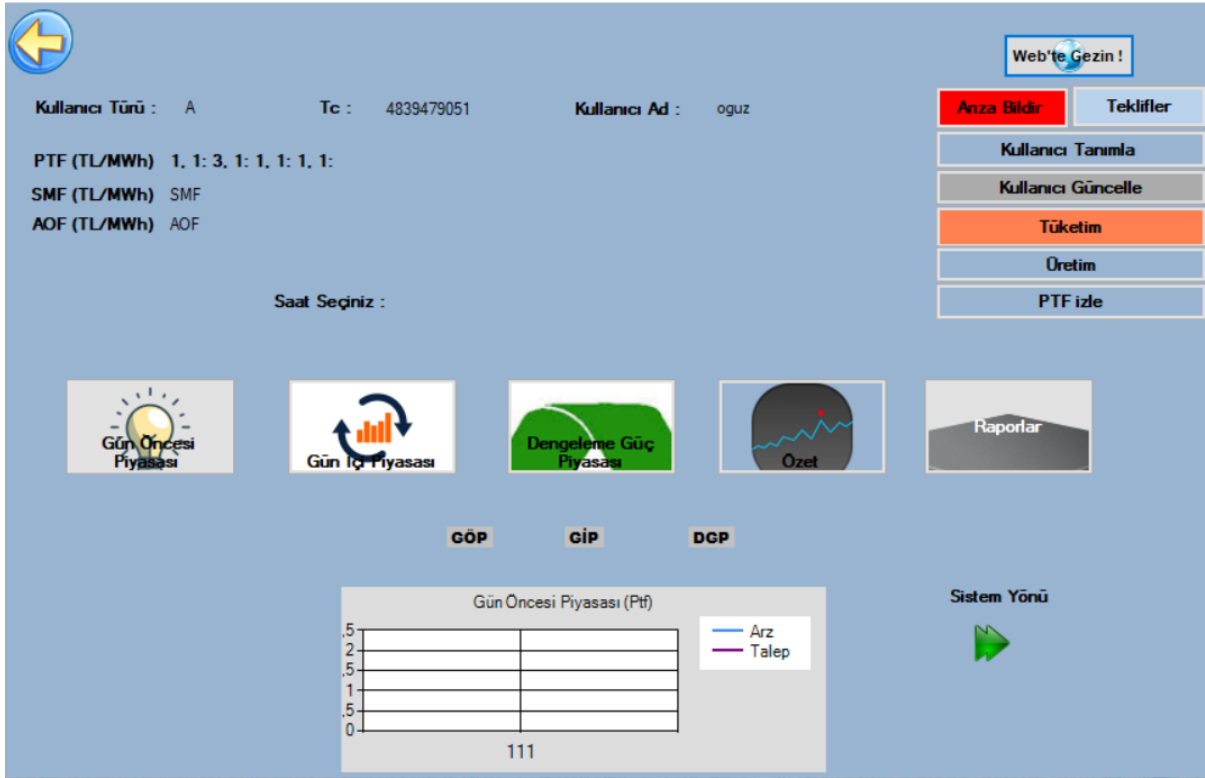
Kullanıcı Hakkında menüsüne tıklandığında aşağıdaki bilgilendirme paneline erişebilmektedir.



Şekil 3.23. Giriş ekranı hakkında bölümü

3.3.4. Şeffaflık platform ana sayfası

Kullanıcı ad ve şifre verisini doğru girmesi durumunda, başarılı login işleminden sonra ana ekran tabir edeceğimiz içersinde yönlendirmelerin bulunduğu ekran mevcuttur.



Şekil 3.24. Ön yüz http protokolünden erişilebilen aspx ve winform(C#) teknolojisiyle kodlanması(Kullanıcı işlem yapacağı ekran)

Ana sayfada kayan yazı şeklinde saatlik bazda ptf, smf ve ortalamalarını görebilmektedir. Ana ekran üzerinden naviga edilebilen butonlar aracılığıyla kullanıcı üretici/tüketici olması durumuna göre gip, göp, dgp ve yan hizmetler bölümüne entegre olabileceği ekranlara yönlenebilmektedir.

3.3.5. Gün öncesi piyasa ekranı

Gün önce piyasa ekranında gün öncesi için teklif verildiği ekrandır. Kullanıcıların tüketici/üretici mwh bazında arz/talebin alındığı ekrandır.

Gün Öncesi Piyasa Ekranı

TEKLİF EKRANI

Bilgi Notu : Bu ekran gün öncesi için teklif oluşturur her adet teklif 10 mwh'dır

DGP

YAL teklif

☐ Normal Teklif ☐ Blok Teklif

Fiyat

Verilecek Teklif Adeti

Saat

Teklif Kaydet

Fiyat

Verilecek Blok Teklif Adeti

Saat

Açık Teklif **Kesinleşen Teklif** **Teklif İptal Ettiklerim** **Eşleşmeyen Tekliflerim**

Şekil 3.25. Gün Öncesi Piyasa ekranı

3.3.6. Gün içi piyasa ekranı

Gün içi piyasa ekranı gün içinde kullanıcıların teklif vermesi için kullanılan ekrandır. Bu ekranının kodları EK’de verilmiştir.

Gün İçi Piyasa Ekranı

Teklif Ver !

Eşleşme Miktarı

Min- Maks Alış Teklif Fiyatı

Min-Maks Satış Teklif Fiyatı

Min - Maks Eşleşme Fiyat

Teklif Edilen Alış Satış Miktarları

İşlem Hacmi

İşlem Akışı

Şekil 3.26. Gün içi piyasa ekranı

3.3.7. Dengeleme güç piyasa ekranı

Dengeleme güç piyasası ekranında dgp'ye girmek isteyen kullanacağı ekrandır. Yal ve yat talimatlarının vb görülebileceği ekrandır.

Dengeleme Güç Piyasası

Temmuz 2018						
Pzt	Sal	Çar	Per	Cum	Cmt	Paz
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Bugün: 29.07.2018

Saat	Sistem Yönü
*	

Şekil 3.27. Dengeleme güç piyasa ekranı

3.3.8. Kullanıcı tanımlama ekranı

Admin yetkisine sahip olan kullanıcıların erişim sağlayabileceği ekrandır. Bu ekranda kullanıcı tanımlanabilir. Kullanıcıya ait kullanıcı adı, şifre, adı, soyadı, kimlik numarası, kullanıcı türünün ve türe ait santral bilgisinin, anlık güç bilgisinin girilebileceği ekrandır.

Kullanıcı Tanımlama



Kullanıcı adı :

Şifre :

Ad :

Soyad :

Tckn :

Kullanıcı Türü : ☐ Admin ☒ Üretici ☐ Tüketici

Kurulu Güç : Mwh

Santral Tipi :



Şekil 3.28. Yeni kullanıcıların tanımlandığı ekran

3.3.9. Kullanıcı güncelleme ekranı

Admin yetkisine sahip kullanıcının diğer kullanıcılar için bilgi güncellemesi yapabileceği ekrandır.

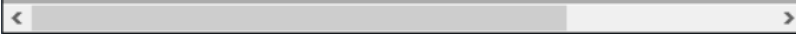
KullanıcıGuncelleme



Kullanıcı Ara

Kullanıcı Adı :

	tkn	ad	soyad	kullanicitur	sifre
	48394790510	oguz	altuntas	A	12



Şekil 3.29. Kullanıcı güncelleme ekranı

3.3.10. Piyasa mesajekranı

Üreticilerin herhangi bir şekilde arz edecekleri güç için yaşadıkları aksaklıkları bildirim etmeleri için kullanılmaktadır.

PMS



Piyasa Mesaj Sistemi

Santral İsmi

Uevcb

Olay başlangıç tarihi

Olay bitiş tarihi

İşletmedeki güç

29 Temmuz 2018

Pazar

▼

29 Temmuz 2018

Pazar

▼

Olay sırasında kapasite

Gerekçe

Santral İsmi	UEVÇB	Olay Başlangıç Tarihi	Olay Bitiş Tarihi	İşletmede Güç

<

>

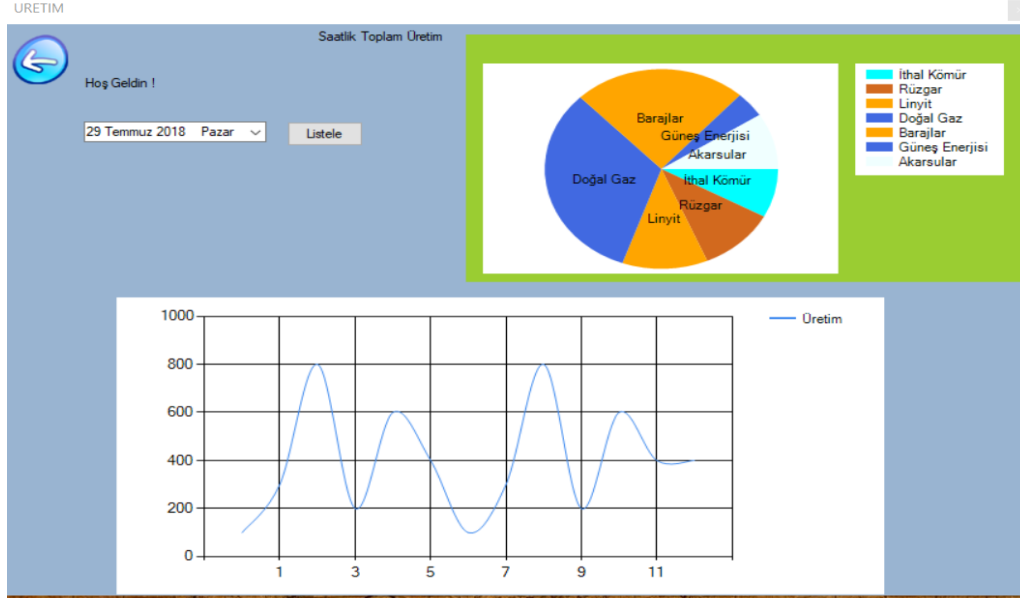
Ekle

Şekil 3.30. Üreticilerin taahhüt ettiği gücü aksaklıklar nedeniyle temin edememesi durumunda mazeret girebilecekleri ekran

Üreticilerin sisteme taahhüt ettiği yük ile ilgili problem yaşamaları durumunda piyasa mesaj sistemi aracılığıyla bildirim yapabilmektedir.

3.3.11.Üretim

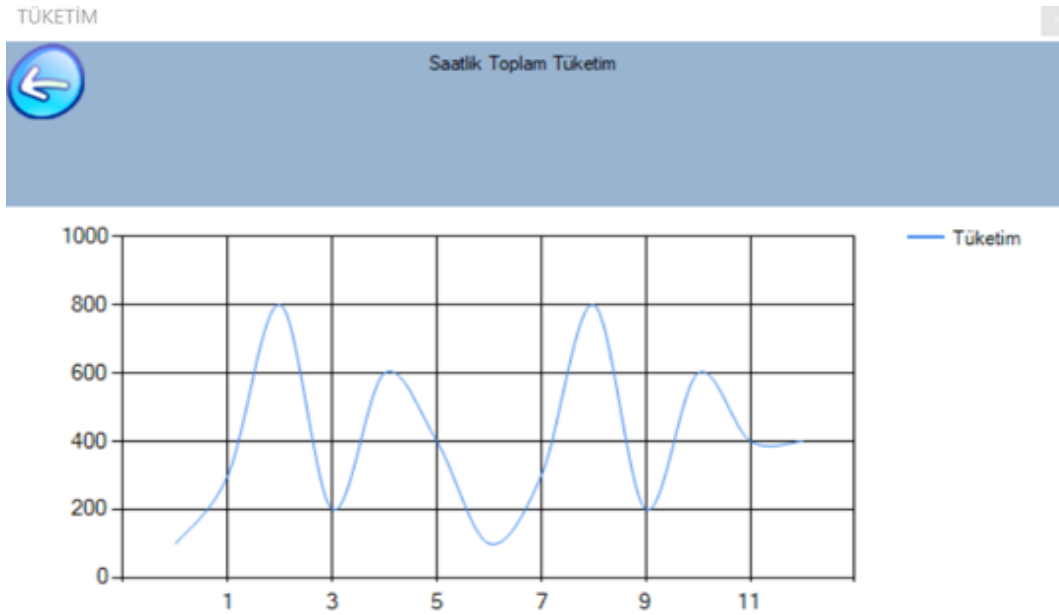
Kullanıcıların sistemde anlık üretimi görebilecekleri ekrandır.



Şekil 3.31. Üretim ekranı

3.3.12.Tüketim

Kullanıcıların piyasadaki saatlik tüketimi görebilecekleri ekrandır.



Şekil 3.32. Tüketim ekranı

3.3.13. Oyun, senaryolar ve çıktılar

Elektirik piyasa simülatörü baştan sona son kullanıcıya hitap eden, kolaylık sağlayan ihtiyaçlarla örtüşen masaüstü uygulaması olmasının yanında sadece internet erişimi baz alınarakta örnek yapılabilecek tarayıcıda çalışan mvc disiplinine sahip web uygulmasına sahiptir.

İlk aşama için kullanıcı sisteme girişini gerçekleştirebilmek adına kullanıcı adı, şifresi ve kullanıcı türünü (üretici, tüketici) belirten formu piyasa operatörüne iletebilmektedir.

Şekil 3.33. Web giriş ekranı

- Kullanıcı Giriş ekranından “Kaydol” butonuna tıklamaktadır.

Şekil 3.34. Yeni kayıt ekranı

- Doğru parametreleri girmesi için ekrandaki kontroller yönlendirme yapmaktadır.

Yeni üyelik talebiniz iletildi.Girmiş olduğunuz mail adresine gönderilecektir.

Tamam

Şekil 3.35. Yeni kullanıcı kayıt örneği

- Ardından kullanıcı ilgili talebin gönderildiğine dair bilgilendirilmektedir.
- İlgili maili piyasa operatörü görür ve uygun görmesi durumunda tanımları yapmaktadır.

Elektrik Piyasa Simülatörü| Sitenizden Gelen İletişim Formu Gelen Kutusu x

 **elektriksimulator@gmail.com**
Alıcı: ben ▾
Ad & Soyad: oguz
E-Mail Adresi: deneme@gmail.com
Konu:
Mesaj: üretici kurulu güç 100 mw

← Yanıtla

➡ Yönlendir

Şekil 3.36. Yeni kullanıcı kayıt mail

- Kullanıcının şifresini unutması durumunda, giriş sayfasından şifremi unuttum butonuna tıklamaktadır. Açılan sayfada kayıtlı olunan mail adresi girilir. Sistem operatörü ilgili maile şifreyi göndermektedir.



Şekil 3.37. Şifremi unuttum

- İlgili mail adresi girildikten sonra şifre talebinizin iletildiğine dair bilgilendirme mesajı açılmaktadır.

Şifre hatırlatma talebiniz alınmıştır. Mail adresinize şifre gönderilecektir.

Tamam

Şekil 3.38. Şifremi unuttum mail

Elektrik Piyasa Simülatörü | Sitenizden Gelen Şifre hatırlatma isteği Gelen Kutusu x

elektriksimulator@gmail.com

Alıcı: ben ▼

E-Mail Adresi: oguz@gmail.com

Konu: Şifre Hatırlatma

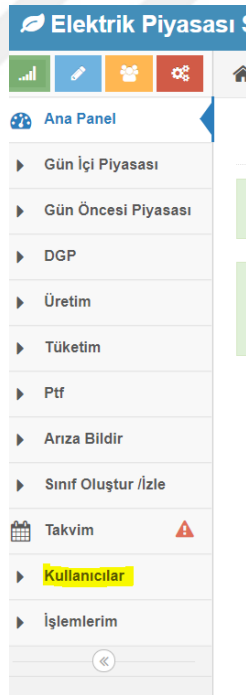
Mesaj: oguz@gmail.comMail adresli kullanıcının şifresini ilet.

Yanıtla

Yönlendir

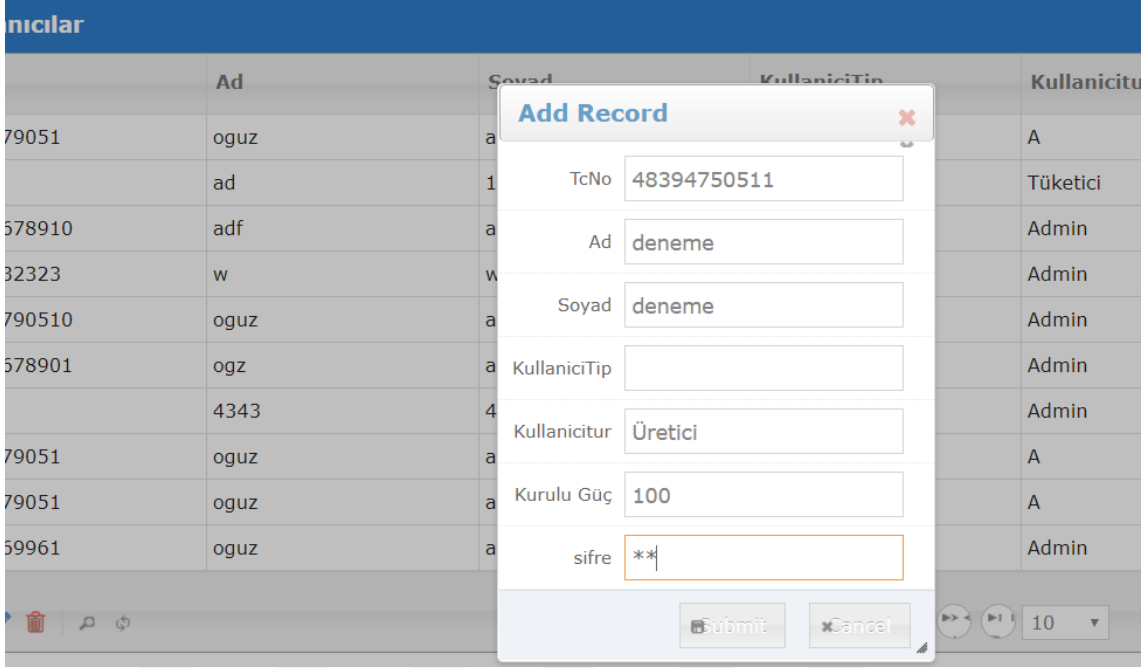
Şekil 3.39. Sistem operatörü mail

- Kullanıcı tanımı “Admin” yetkisi olan kullanıcı tarafından yapılmaktadır.



Şekil 3.40. Admin ana menü

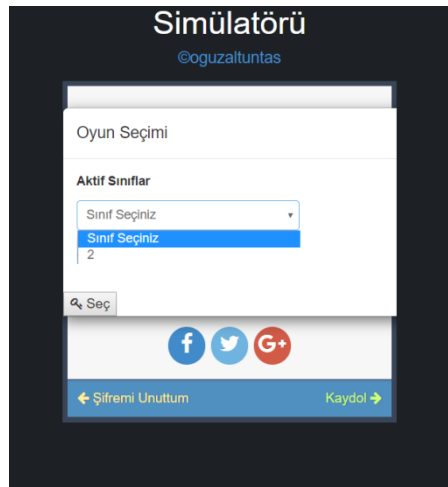
- Kullanıcının tipine göre üretici veya tüketici tanımlanmaktadır.



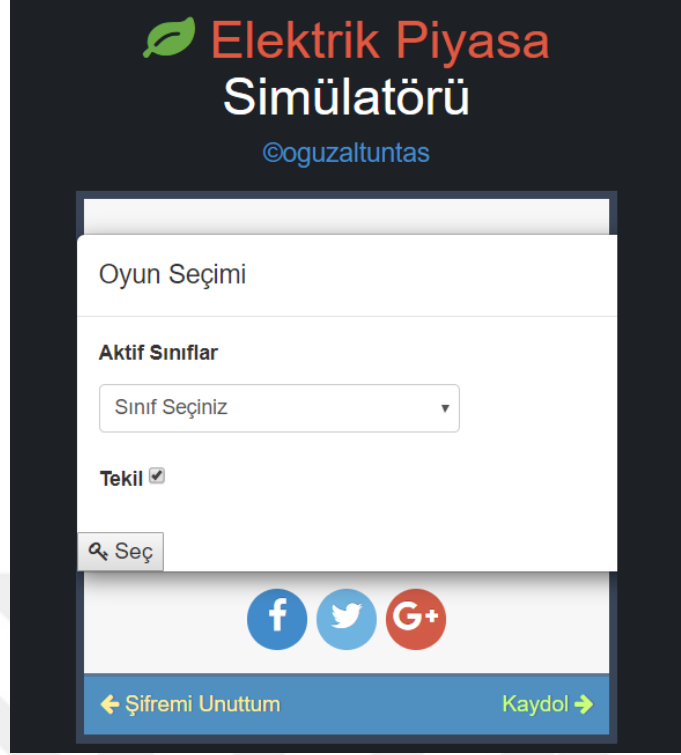
Şekil 3.41. Kullanıcı tanım ekranı

İlk senaryo için Kullanıcı üretici veya tüketici ve tekil işlem yapması durumunda;

- Kullanıcı giriş ekranından kullanıcı adı ve şifresi ile sisteme başarılı bir şekilde giriş yaparsa kullanıcıya tekil veya sınıf içinde işlem yapması seçenekleri sunulmaktadır.
- Kullanıcı tekil işlem seçeneği seçer ve ana ekran dediğimiz (dashboard) giriş ekranına ulaşır ve buradan erişim yetkisinin (rol) bulunduğu sekmelerden işlem yapabilmektedir.

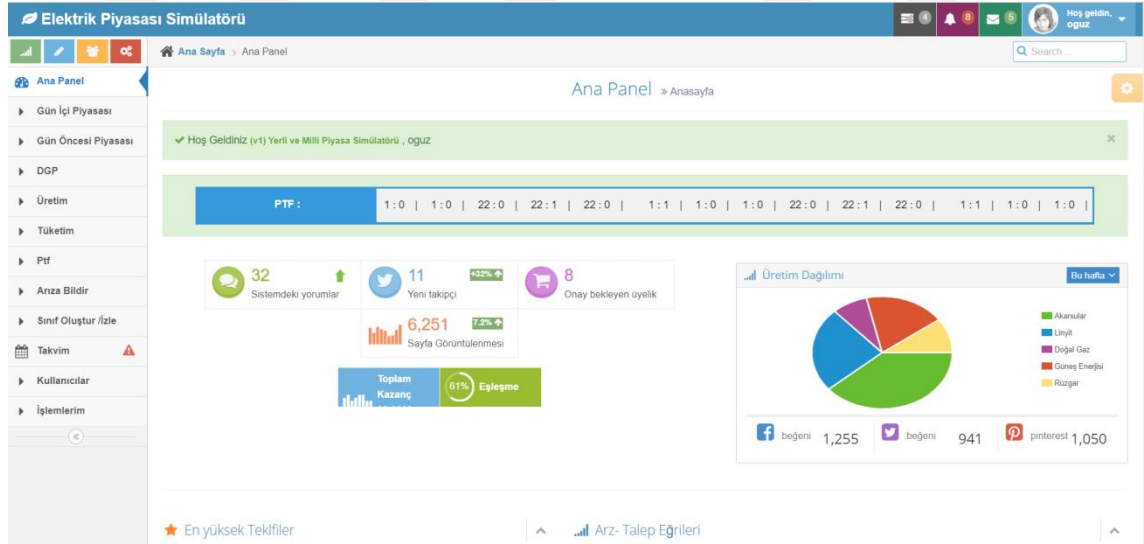


Şekil 3.42. Tekil, sınıf oyun seçim ekranı



Şekil 3.43. Tekil oyun seçimi

- Kullanıcı ana ekrana yönlendirilmektedir.



Şekil 3.44. Ana ekran

- Ana ekranda görünen sekmelere kullanıcı türlerine göre erişim sağlayabilmektedir. Admin rolü tüm sekmelere erişim yetkisi bulunurken, üretici veya tüketici sınıfı oluşturma, kullanıcılar gibi sekmelere yetkisi bulunmamaktadır.
- Ana ekrandan gün öncesi piyasası butonuna tıklayıp buradan teklif verebilmektedir.

Şekil 3.45. Gün öncesi piyasa teklifi

- Belirlenen tarih ve saat için piyasa takas fiyatı oluşmaktadır.
- Verilen teklifleri üretici ve tüketici bazında tabloda gösterimi aşağıdaki tablolarda gösterilmektedir.

Tablo 3.1. Tüketici1 teklifi

Fiyat(₺/MWh)	2000	50
Miktar(lot)	0	125

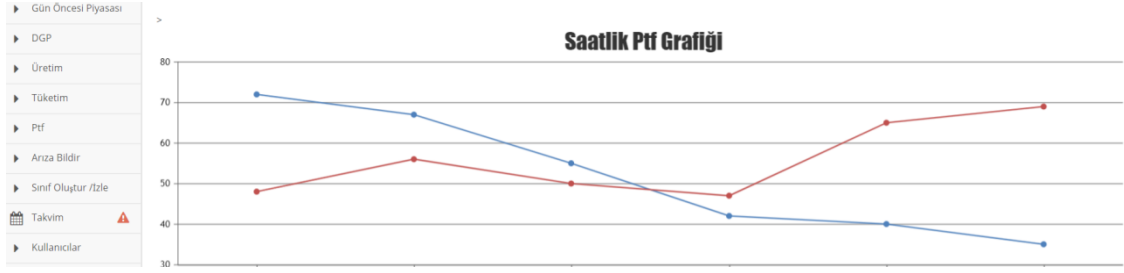
Tablo 3.2. Tüketici2 teklifi

Fiyat(₺/MWh)	0	100	200	250
Miktar(lot)	100	100	50	0

Tablo 3.3. Üretici1 teklifi

Fiyat(₺/MWh)	0	150	200	300
Miktar(lot)	0	-100	-160	-200

- Doğrusal interpolasyon ile Ptf 75₺ /MWh ve Üretici1 den 50 lot eşleşmektedir.
- Oluşan Ptf fiyatı ptf ekranından izlenebilmektedir.



Şekil 3.46. Saatlik ptf izleme ekranı

- Teklif izleme ekranında, teklif durumu kontrol edilebilmektedir.

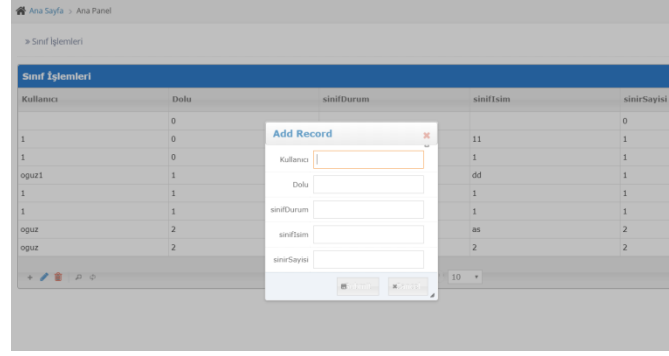
İşlemler							
adet	durum	eslesenteklifler	saat	sınıfId	teklif_birim_fiyat	teklif_tur	Tarih
2	AKTIF	8	1	0	111	A	01/01/1
2	PASIF		1	0	111	A	31/05/2018
1	P		1	0	1	Admin	03/12/2018
1	AKTIF		1	0	1	Admin	04/12/2018
1	A		1	0	11	Admin	08/12/2018
1	A		1	0	111	Admin	08/12/2018
2	A	0	3	0	100	Üretici	09/12/2018
1	P		1	0	100	Üretici	09/12/2018
1	A	0	1	0	100	Admin	10/12/2018
1	A	0	1	0	1	Uretici	27/01/2019

Şekil 3.47. Teklif izleme ekranı

İkinci senaryo için kullanıcının üretici veya tüketici ve bir sınıfa bağlı işlem yapılması durumunda;

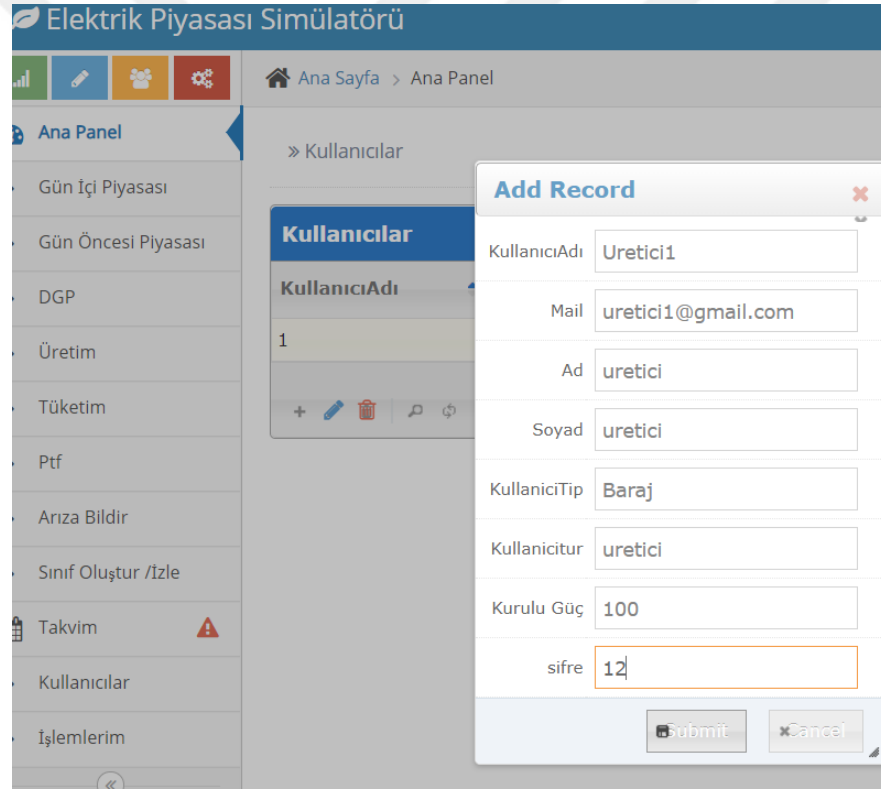
- Kullanıcı giriş ekranından kendisine ait kullanıcıadı ve şifresi ile sisteme başarılı bir şekilde giriş yapması durumunda, tekil veya sınıf içinde işlem yapması seçenekleri sunulmaktadır.
- Kullanıcı sınıf içinde işlemi seçer ve ana ekrana yönlendirilmektedir.
- Ana ekrandan gün öncesi piyasası butonuna tıklayıp kurulu gücü ve diğer durumlarını göz önüne alarak ilgili adet ve fiyatta teklifini yapabilmektedir.
- Belirlenen tarih ve saat için piyasa takas fiyatı oluşmaktadır.
- Teklif durumunu ilerleyen süreçte kontrol edebilmektedir.

Sistemde admin rolüne sahip kullanıcı kapsamlı yetkinliğe sahiptir. Ptf belirlenmesi, sınıf belirlenmesi gibi operasyonel işlemleri yapabilecek yetkilere sahip olmaktadır.



Şekil 3.48. Admin Sınıf oluşturma ekranı

- Sistem operatörü, 20 adet kullanıcıyı üretici tipinde tanımlamaktadır.



Şekil 3.49. Kullanıcıların tanımlanması

- Gün öncesi piyasası için saat 13:00 için üretici ve tüketici teklifleri aşağıdaki tabloda gösterilmektedir.

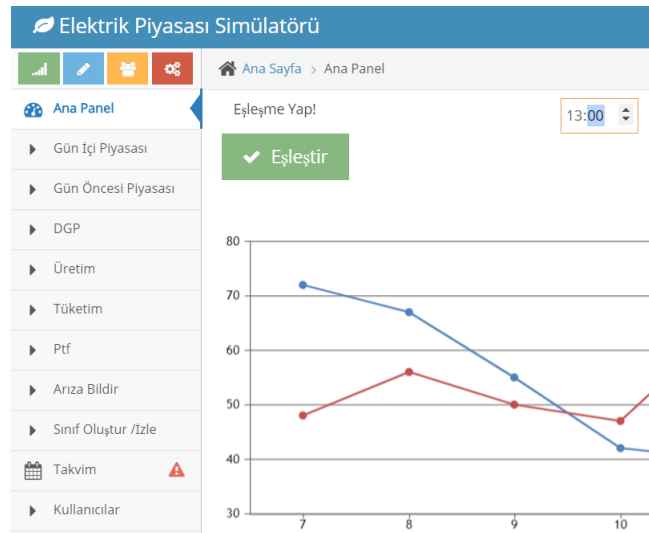
Tablo 3.4. 20 tane üreticinin saat 13:00 için girdiği teklifler

Santral	Fiyat(₺/MWh)	Miktar(lot)	Kapasite(Mwh)
Uretici1	00.01	100	100
Uretici2	50	50	120
Uretici3	55	75	100
Uretici4	70	50	75
Uretici5	100	80	700
Uretici6	45	120	150
Uretici7	60	110	150
Uretici8	80	150	170
Uretici9	90	200	210
Uretici10	100	220	250
Uretici11	130	180	200
Uretici12	200	80	100
Uretici13	90	50	75
Uretici14	110	90	100
Uretici15	120	190	210
Uretici16	75	170	200
Uretici17	80	160	200
Uretici18	50	100	150
Uretici19	40	50	75
Uretici20	95	75	100

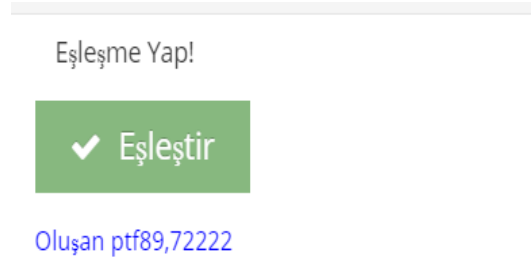
Şekil 3.50. Sınıfın tanımlanması

- Saat 13:00 için sisteme girilen tedarikçinin toplam teklifi 1000 mwh ve ptf 90 tı ve altı olursa eşleşme olması beklenmektedir.

Piyasa operatörü admin ogretmen kullanıcısı saat 13:00 için ptf oluşturulmasını tetikleyerek akabinde sistemin eşleştirmeleri yapması için ptf ekranından Eşleştir butonuna tıklar ve oluşan ptf'ye göre eşleşmeler yapılmaktadır.

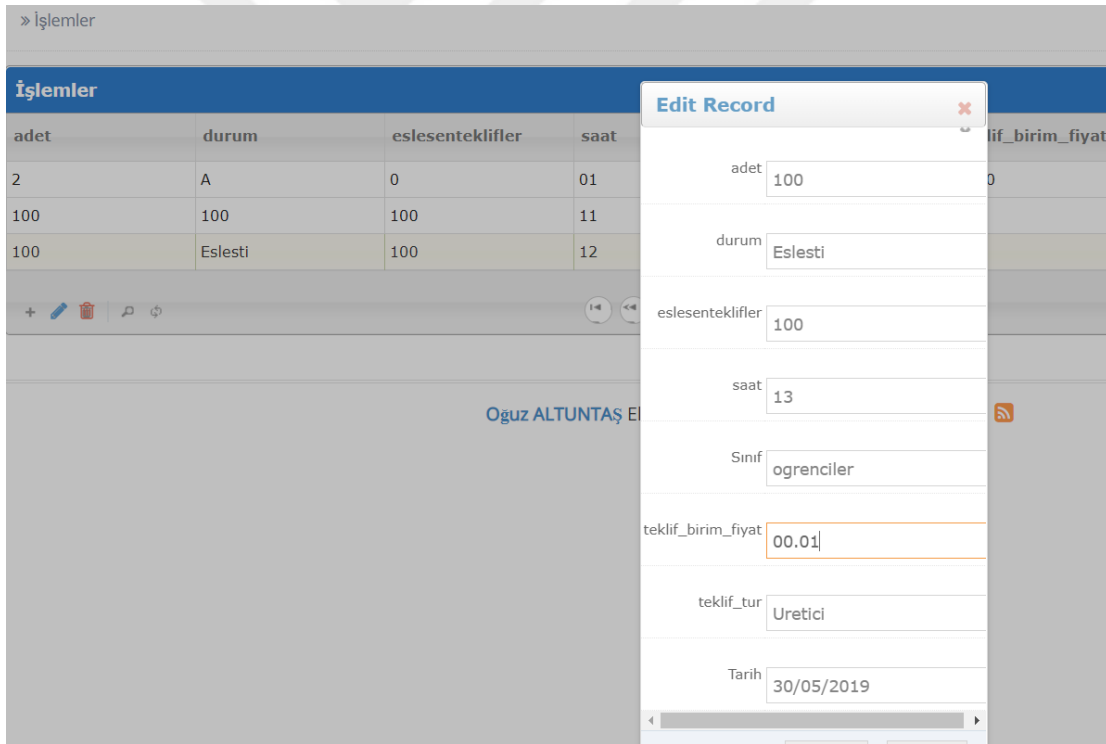


Şekil 3.51. Ptf belirleyip, teklif eşleştirmesi yapılması

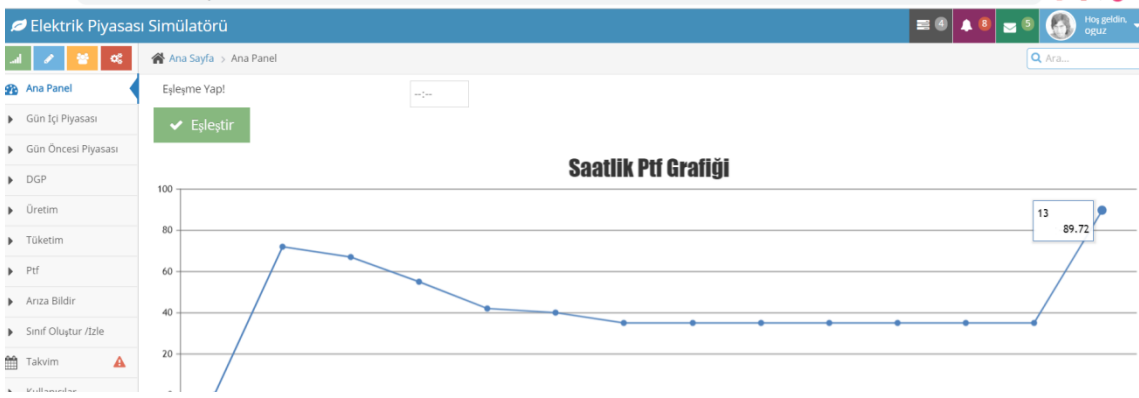


Şekil 3.52. Ptf oluşması

- Sistem girilen teklifleri optimal seviyede tutmak için teklifleri üreticiler için düşükten artarak sıralarken, tüketici teklifleri için tam tersi şekilde olacak şekilde sıralar ve piyasa takas fiyatına göre eşleştirmeleri yapabilmektedir.
- Kullanıcılar eşleşen veya eşleşmeyen tekliflerini işlemlerin menüsü altından kontrol edebilmektedir.



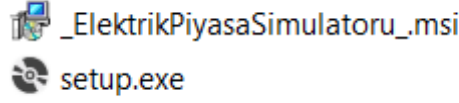
Şekil 3.53. Tekliflerin kontrolü



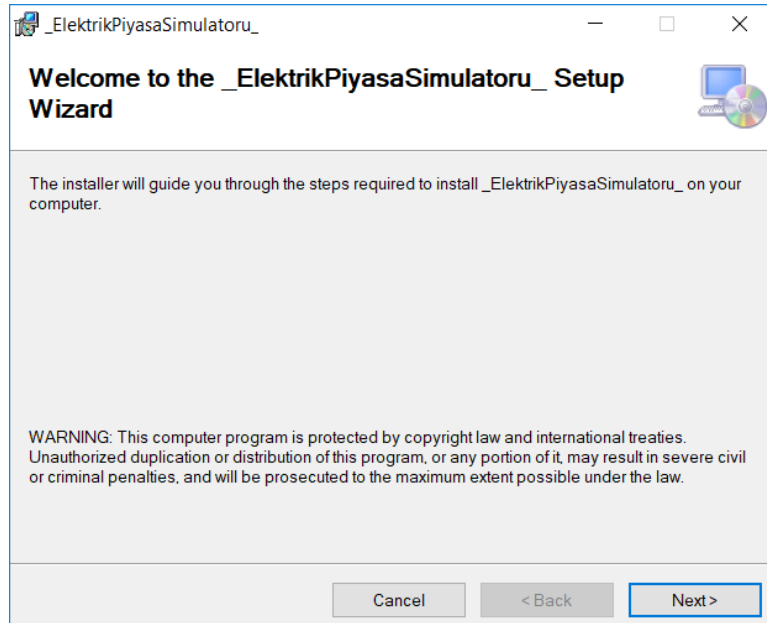
Şekil 3.54. Saat 13:00 için oluşan ptf

3.4. Oyunun Çoklu Ortama Yüklenebilmesi

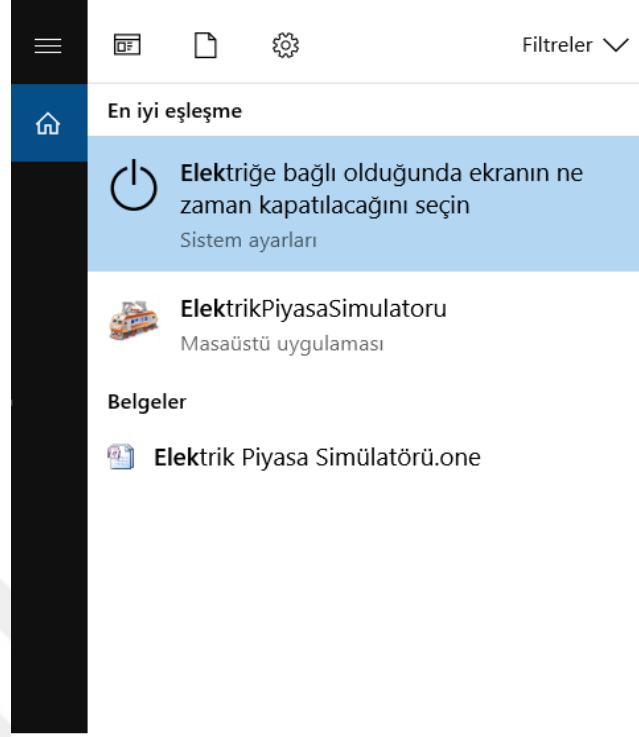
Oyun .Net framework alt yapısını kullandığı için Windows işletim sistemini barındıran güncel framework ile güncellenmiş 4.5.2 versiyon içeren bilgisayarlarda sorunsuz çalışır. Yürütülebilir kod (executable) setup şeklinde hazırlandı. Kullanıcı bu exeyi bilgisayarına yüklediğinde kullanıcı isim ve şifresiyle piyasaya katılabilir.



Şekil 3.55. Piyasa Simülator kurulum dosyası



Şekil 3.56. Piyasa simülator kurulum aşaması



Şekil 3.57. Piyasa Simülatör kurulum aşaması

3.4.1. Web kanallı bilgisayara indirilmesi

Simülatörün çağın gerektirdiği biçimde insanların kullanımına sunulması için web tabanlı bir proje oluşturulmuştur. Php programlama ve mysql veri tabanı sistemini kullanan bu web projesinde enerji ile alakadar paylaşımların yanı sıra simülatör hakkında bilgi ve deneme sürümünü denemek edinmek isteyenlerin internet ortamında erişebilecekleri web projesidir.

4.SONUÇ

Elektrik piyasalarında serbestleşme hareketi özellikle 1990'lı yılların ikinci yarısından itibaren başlamış ve gelişmiş ülkelerin öncülüğünde hızlı bir şekilde yayılmıştır. Böylece elektrik piyasa yapıları, tekeli yapıdan içinde rekabeti barındıran piyasalar haline dönmüştür. Bu bağlamda birçok ülke yasal düzenleme ve buna bağlı mevzuatlarını hızla tamamlayarak, serbest piyasa yapısını oluşturmuştur.

Bu çalışmanın amacı, Türkiye'de 2001 yılında başlayan yasal değişim sonrası 2013 yılında yürürlüğe giren 6446 sayılı yeni Elektrik Piyasası Kanunu ile yeni bir hal alan elektrik piyasasının sektörde çalışan ve çalışacak tüm katılımcılarına benimsetmek için basit ve her alanda kullanımı olan bir yazılım oluşturmaktır. Böylece Türkiye elektrik piyasasının yeni yapısını ve işleyişini kavratmak amacıyla bir elektrik piyasası simülatör tasarımı hedeflenmiştir.

Bu kapsamda çalışmanın ikinci bölümünde Türkiye elektrik piyasası üzerinde durulmuş ve piyasa işleyişlerinden bahsedilmiştir. Ayrıca elektrik piyasa simülatörü kavramı açıklanmış ve yapılan önceki çalışmalar sunulmuştur. Tez çalışması bölümünde ise, geliştirilen piyasa mimarisi ve akış diyagramı verilmiş simülatörün temel bileşenleri sunulurken, oluşturulan web tabanlı platform ekranları aktarılmıştır. Kullanıcıların oyuna katılımı ile ilgili senaryolar sunulmuştur. Bu çalışma, yeniden yapılanan elektrik sektörünün oluşturduğu türev piyasa işleyişinin tüm paydaşları tarafından daha anlaşılabilir olmasını sağlamak için bir elektrik piyasası simülatörü geliştirilmiştir.

Böylece bu çalışma lisans ve lisansüstü öğrenciler başta olmak üzere elektrik piyasasında katılmak isteyen çalışanlara, yeni yapının daha iyi kavranması ile hizmet içi, lisans ve lisansüstü eğitimleri sırasında alınacak öğretimler ile daha iyi analiz edilebilmesi sağlayacaktır.

Bu çalışmayla, kullanıcıların IP bağlantısının alınabileceği şekilde global internet bağlantısına sahip olan bilgisayardan login ekranından başarılı giriş yapmasından sonra karşısına gelen ana panelden piyasa ihtiyaçlarını karşılayacak, temel ve ekstra özellikleri barındıran zengin bir ekranla karşılaşacaktır.

Program, hali hazırda çalışan piyasa yapısını sürdürülebilir olabilmesi amacıyla kullanıcıların izlerini kayıt altına alarak çeşitli ekranlarda verilerin filtrelenerek kullanılmasını sağlar. Elektrik piyasası yapısına ilişkin teorik açıdan erişilebilen

kaynaklar olmasına rağmen teknolojik olarak geliştirilen açık kaynak kodlu egzersiz amaçlı piyasa yapısının bulunmamasına karşın bir çözüm sunar.

Bu çalışmayla birlikte Elektrik piyasası simülatörü ile gerçek bir elektrik piyasasındaki gerçekleştirilebilen tüm işlemlerin yapılması hedeflenmiştir. Yeniden yapılanma sürecinin yaşandığı Türkiye Elektrik Piyasası ile ilgili lisans ve lisansüstü kapsamlı bilgi kaynağını artırmak, ayrıca piyasa yapısı içinde bulunan yeni katılımcılara katkı yaparak, Türkiye elektrik piyasasının karmaşık yapısı ve bu piyasadaki güncel değişiklikler hakkında bilgi sahibi olabilmesini sağlamaktır. Böylece katılımcılar, Türkiye elektrik piyasasını değişik perspektiflerden gözlemleme şansı bulacaktır. Geliştirilecek simülatör ile katılımcılar gerekli yatırım adımlarını atarak elektrik tedarikçisi olabilecek ve gerçek piyasa ile tamamen aynı olan elektrik ticareti işlemlerini yapabilmesi sağlanacaktır. Böylece herhangi bir risk almadan Türkiye elektrik piyasası ile ilgili ciddi bir deneyime sahip olabilecektir.

Özellikle Türkiye Elektrik Piyasasındaki yeni durumu ve farklılıkları öğrenciye önceden kavratmak ve uygulatabilmek amacı ile eğitim amaçlı elektrik piyasası simülatörünün tasarımı, öğrencilere Türkiye elektrik piyasasının temelini öğretmek ve onlara gerçek market ortamında alıştırma yapma şansını verecektir. Çalışma daha da geliştirilerek, hem eğitim hem ticari bir yapıya oturtulabilir.

KAYNAKLAR

- [1] Yıldız, S.: “Türkiye Elektrik Piyasası Kısa Dönemli Referans Fiyat Tahmini”, İstanbul Teknik Üniversitesi, Enerji Enstitüsü, Yüksek Lisans Tezi, (2015).
- [2] Pollitt, M.(2004) Electricity Reform in Chile Lessons for Developing Countries, Center for Energy and Environmental Policy Research,04-016 WP, Cambiridge Üniversitesi
- [3]Doren,P.V.,TheRegulationof The Electictiy Industry, <http://www.cato.org/pubs/pas/pa-320.pdf>, 10 Ekim 2018.
- [4]-Çetintaş, H.,Bicil M. İ.,”Elektrik Piyasalarında Yeniden Yapılanma ve Türkiye Elektrik Piyasasında Yapısal Dönüşüm”,Optimum Ekonomi ve Yönetim Bilimleri Dergisi,2015, 2(2),1-15
- [5]-Dünya Enerji Konseyi Türkiye, Türkiye Enerji Piyasaları Araştırma Raporu, Temmuz 2018
- [6] Toker, C. A.; Korkmaz O.: “Türkiye kısa süreli elektrik talebinin saatlik olarak tahmin edilmesi” II. Elektrik Tesisat Ulusal Kongresi Bildirileri, İzmir, 24-27 Kasım, (2011).
- [7] Bölük, G. (2010). Türkiye Elektrik Piyasasında Düzenleyici Reform ve Performans, Rekabet Ekonomisi ve Politikası Sempozyumu. Alındığı tarih: 29.05.2017, adres: <http://goo.gl/BCZQ50>
- [8] Meeus, L.,Purchala, K., &Belmans, R. Development of theinternalelectricity market in Europe. *TheElectricityJournal*, 18(6), 25-35,2005
- [9]Heller,T. C.,Victor, D.G, A Political Economy Electric Power Market Restructuring:Introduction to Issues and Expectations, *Working Paper*,https://fsi-live.s3.us-west-1.amazonaws.com/s3fs-public/wp1%2C_1_May_04.pdf, Erişim Tarihi: 10.08.2018.
- [10]Gaz ve Elektrik,<https://gazelektrik.com/enerji-piyasalari/elektrik>,09 Şubat 2019.
- [11] Kölmek, F. : “Türkiye Elektrik Piyasasında Fiyat Oluşumunun Analizi ve Fiyat Tahmin Modelleri”, Hacettepe Üniversitesi Sosyal Bilimler Enstitüsü, İktisat Anabilim Dalı, Doktora Tezi, Ankara, (2016).

[12]Türkiye Elektrik İletim A.Ş.,<https://www.teias.gov.tr/tr/yayinlar-raporlar/piyasa-raporlari>, 09Aralık 2018.

[13]Weron, R.(2006) Modeling and Forecasting Electricity Loads and Prices

A Statistical Approach, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, Batı Sussex, İngiltere

[14]Arslan, S. 2008. Elektrik Enerjisi Sektöründe Serbestleşme, Yeniden Yapılanma, Özelleştirme Uygulamaları ve Dünya Örnekleri. Uzmanlık Tezi, Enerji Piyasası Düzenleme Kurumu, 172 s., Ankara.

[15]Türkiye Cumhuriyeti Enerji Piyasası Düzenleme Kurumu,<https://www.epdk.org.tr/>, 03 Mart 2019.

[16] Türkiye Elektrik Ticaret ve Taahhüt A.Ş. Genel Müdürlüğü ,2014 Yılı Sektör Raporu , Mayıs 2015

[17] Ziel F., Steinert R., Husmann S., Forecasting day ahead electricity spot prices:The impact of the EXAA to other European electricity markets. *Energ. Econ.*, V.51, p.430-444, 2015

[18] EPIAŞ, Enerji Piyasaları İşletme A.Ş., <https://www.epias.com.tr>, 21.05.2017

[19]Kölmek M.,A., Türkiye Elektrik Dengeleme ve Uzlaştırma Piyasasındaki Sistem Gün Öncesi Fiyatının Yapay Sinir Ağları Kullanılarak Modellenmesi, *Ankara Üniversitesi, Fen Bilimleri Enstitüsü, Elektronik Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi*,Ankara, 2012.

[20] Elektrik Piyasası Dengeleme ve Uzlaştırma Yönetmeliği, <https://kms.kaysis.gov.tr/Home/Goster/38953> , 08 Şubat 2019.

[21]T.C. Cumhurbaşkanlığı Resmi Gazete,<http://www.resmigazete.gov.tr/eskiler/2009/04/20090414-48.htm>, 10 Ekim 2018.

[22] PMUM,Piyasa Mali Uzlaştırma Merkezi, <https://slideplayer.biz.tr/slide/1951091/>, 25 Ağustos 2018.

[23]Vries Andreas de, Cem Kiyak Electricity Market Mechanism regarding the Operational Flexibility of Power Plants, *Modern Economy*, 2017, 8, 567-589

[24]Bicil M.I., Elektrik piyasasında fiyatlandırma ve Türkiye elektrik piyasasında fiyat

tahmini,Doktora Tezi, *Balıkesir Üniversitesi,Sosyal Bilimler Enstitüsü, İktisat Anabilim Dalı*, Doktora Tezi,2015.

[25] TMMOB Elektrik Mühendisleri Odası, http://www.emo.org.tr/ekler/dc7f3464a4839e6_ek.pdf, 15 Eylül 2017.

[26]Ceyhan G. ,Türkiye` de Elektrik Piyasa Takas Fiyatı ve Sistem Marjinal Fiyatı Farkı Üzerine İstatistiksel Bir Çalışma

[27]EPİAŞ Şeffaflık Platformu, <https://seffaflik.epias.com.tr/transparency,21> Mayıs 2017.

[28] Enerji Piyasaları İşletme A.Ş.,<https://www.epias.com.tr/gun-oncesi-piyasasi/teklifler/>,28 Mayıs 2017

[29]Tiago, P. : Zita, V. ; Tiago M. S. ; Isabel, P.; Gabriel, S.; Hugo, M.: “Adaptive learning in agents behaviour: A framework for electricity markets simulation”, Integrated Computer-Aided Engineering, Vol. 21, no. 4, pp. 399-415, (2014).

[30]Tiago, P.; Isabel, P.; Zita, V.; Hugo, M.; Tiago M. S.: “Strategic bidding in electricity markets: An agent-based simulator with game theory for scenario analysis Integrated”, Computer-Aided Engineering, Vol. 20, no. 4, pp. 335-346, (2013).

[31]Tiago, P.; Tiago M. S.; Hugo, M.; Isabel, P.; Zita, V.: “Metalearning to support competitive electricity market players’ strategic bidding” Electric Power Systems Research, V.135, pp 27–34, (2016).

[32]Aydın, A. E.: “Turkish Electricity Market Simulation Game”,Boğaziçi University, Management Information Systems, Master Thesis, (2013).

[33] Öz Yazılım Danışmanlık,<http://www.ozyazilim.com/ozgur/marmara/karar/simulasyon.htm>(Erişim 13 Şubat 2019

[34]Vrriesa L.J.,Subramahianb E. ve Chappinc E.J.L. ,A Powergame: SimulatingtheLong-Term Development of an ElectricityMarket in a Competitive Game, [Second International Conference on Infrastructure Systems and Services: Developing 21st Century Infrastructure Networks \(INFRA\)](http://www.infra2009.org/),2009

[35] Babić J.,A Simulation Platform ForPowerTrading,Zagreb Üniversitesi, Elektrik ve

Bilgisayar Mühendisliği Fakültesi, Yüksek lisans tezi, Haziran 2012.

[36] Günaydın, A., A Game Of Clustered Electricity Generators, *Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi*, Mayıs ,2008.

[37] Günaydın, A., A Game Of Clustered Electricity Generators, *Orta Doğu Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Yüksek Lisans Tezi*, Mayıs ,2008.

[38] Pinto, T., Vale, Z., Sousa, T., Praça, I., Gabriel, S., Hugo, M., Adaptive Learning in Agents Behaviour: A Framework for Electricity Markets Simulation ,*Integrated Computer-Aided Engineering*, 2014.

[39] Multi-Agent Simulator of Competitive Electricity Markets website (2018), Erişim: <http://www.massem.gecad.isep.ipp.pt/overview.php/>

[40] Contreras, J., Conejo, A. J., Torre, S., Munoz, M., Power Engineering Lab: Electricity Market Simulator, *IEEE Power Engineering Review*, 2002.

[41] Oral, B.: “Enerji Ekonomisi ve Politikaları Ders Notları”, Marmara Üniversitesi Fen Bilimleri Elektrik Elektronik Mühendisliği Lisansüstü, (2016).

[42] Lam, YC; Wu, F. F.: “Simulating electricity markets with Java” IEEE Power Engineering Society. 1999 Winter Meeting, 31 Jan- 4 Feb (1999).

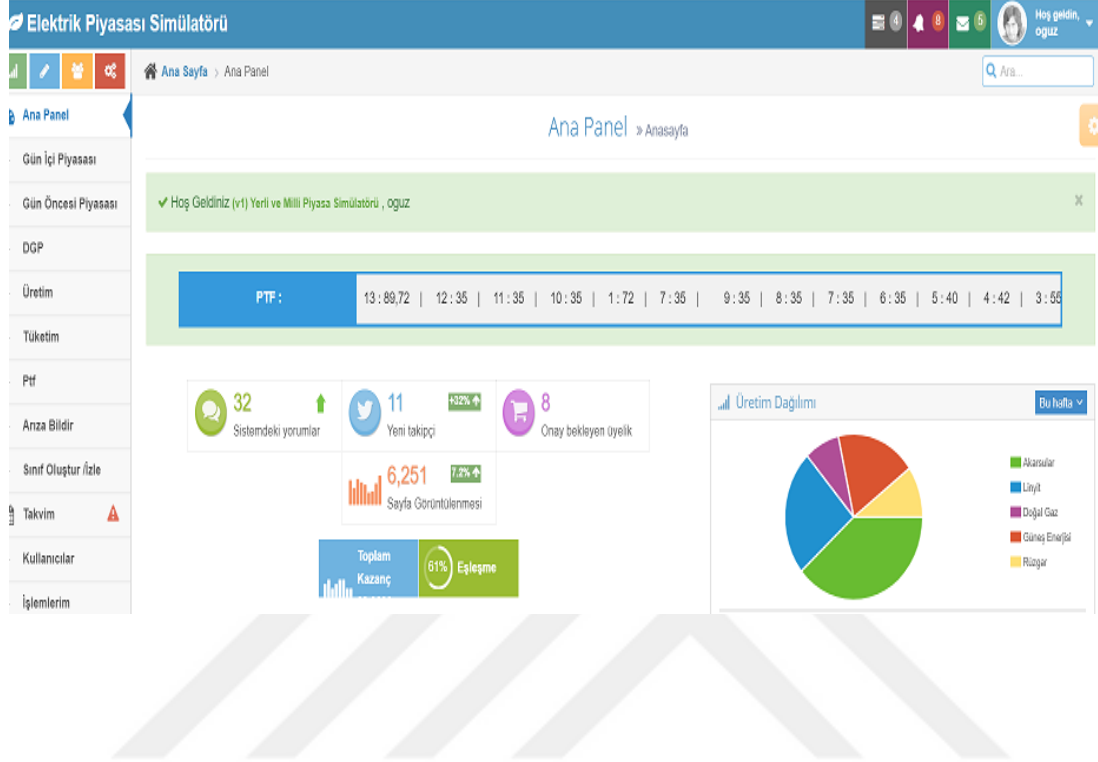
[43] Dejan Paravan, Student Member, IEEE, Andrej Sajn, Student Member, IEEE, and Robert Golob Member, IEEE, Teaching Trading Electricity with the Use of Electricity Market Simulator, Paper accepted for presentation at 2003 IEEE Bologna PowerTech Conference, June 23-26, Bologna, Italy

[44] Microsoft, <https://docs.microsoft.com/en-us/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework> ,01 Temmuz 2019.

[45] Oracle Corporation, <http://www.mysql.com>, 04 Kasım 2018.

EKLER

ELEKTRİK PİYASA SİMÜLATÖR EKRAN VE KODLARI



The screenshot displays the 'Gün içi piyasası!' (Intra-day Market) form within the 'Ana Panel' of the 'Elektrik Piyasası Simülasyonu'. The form includes fields for 'Adet' (Quantity) and 'Fiyat' (Price), both currently set to 0.00. A 'Teklif Saat Seçiniz!' (Select Offer Hour!) dropdown menu is visible, and a 'Kaydet' (Save) button is at the bottom.

Adet	Fiyat
Adet	0.00

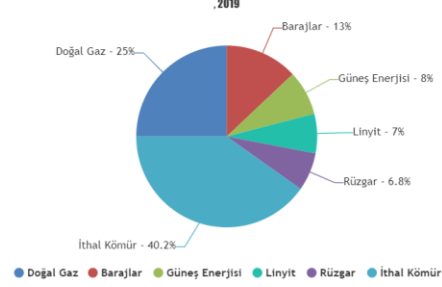
Teklif Saat Seçiniz!

Kaydet

Takvim »

May 2019							month	week	day
Sun	Mon	Tue	Wed	Thu	Fri	Sat			
28	29	30	1	2	3	4			
5	6	7	8	9	10	11			
12	13	14	15	16	17	18			

Üretim Dağılımı



```
[HttpPost]
0 references
public ActionResult Ptf(string saat)
{
    //ptf hesapla ve eşleştirme yap...
    PtfHesapla ptfHesapla = new PtfHesapla(saaf);
    string resp = ptfHesapla.PtfHesaplama(saaf);
    ViewBag.mesaj = resp;

    return View();
}

0 references
public ActionResult kayanYazi()...

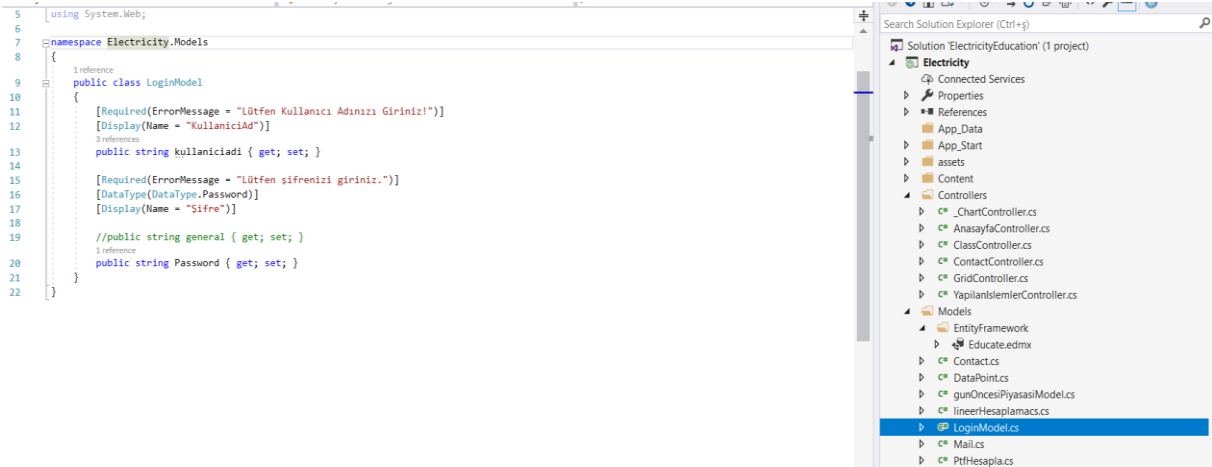
0 references
public ActionResult sınıfIslemIlk()...

[HttpPost]
0 references
public ActionResult sınıfIslem(bool tekil)...

0 references
public ActionResult DGP()...

0 references
public ActionResult üretim()...
0 references
public ActionResult tüketim()...

0 references
public ActionResult işlemlerim()...
```



```

using System.Text;
using System.Web;
using System.Web.Mvc;
using static Electricity.Models.Contactus;

namespace Electricity.Controllers
{
    0 references
    public class ContactController : Controller
    {
        // GET: Contact
        [AllowAnonymous]
        0 references
        public ActionResult Index()
        {
            return View();
        }
        //yeni üye kaydı yapıcım...
        [HttpPost]
        [AllowAnonymous]
        0 references
        public ActionResult index(Contact model)
        {
            if (ModelState.IsValid)
            {
                var body = new StringBuilder();
                body.AppendLine("Ad & Soyad: " + model.Name);
                body.AppendLine("E-Mail Adresi: " + model.Email);
                body.AppendLine("Konu: " + model.Subject);
                body.AppendLine("Mesaj: " + model.MessageMail);
                Mail.MailSender(body.ToString());
            }

            return View();
            //return View();
        }
    }

```

```
0 references
public class AnasayfaController : Controller
{

    //tüm actionların kontrol karargah merkezi...

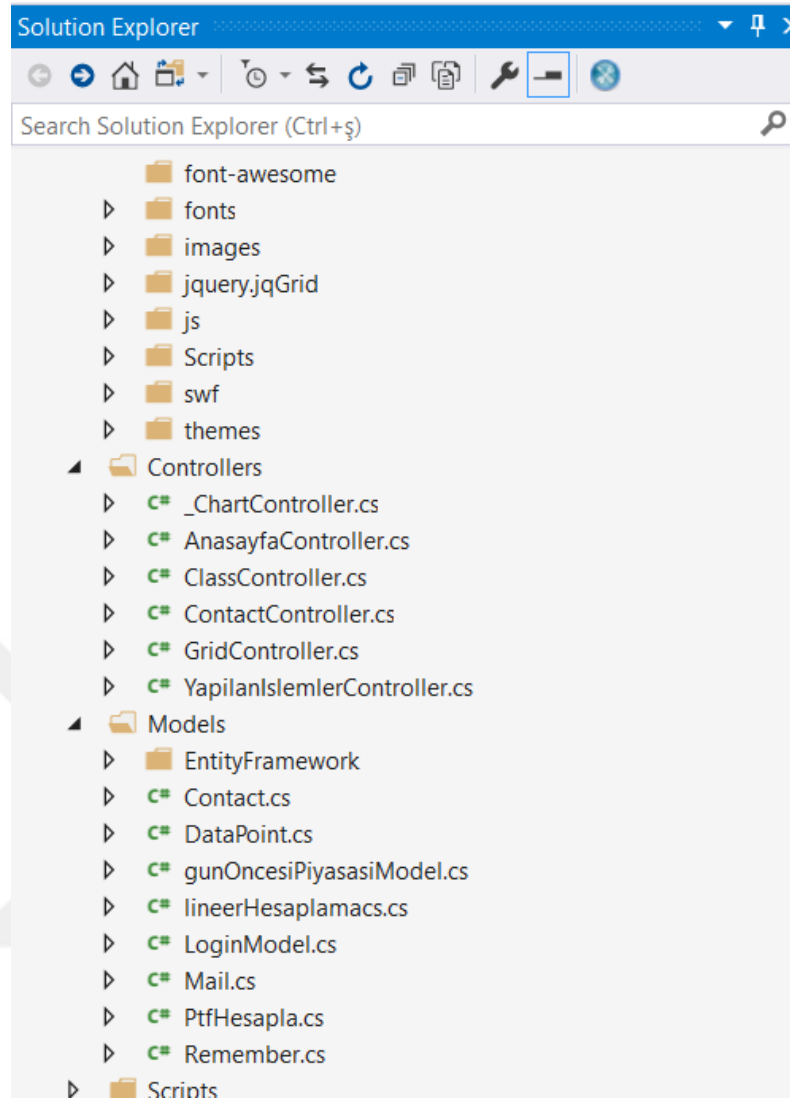
    // GET: Anasayfa
    [AllowAnonymous]
    0 references
    public ActionResult Index()
    {
        //ViewBag.partial = "_middle";

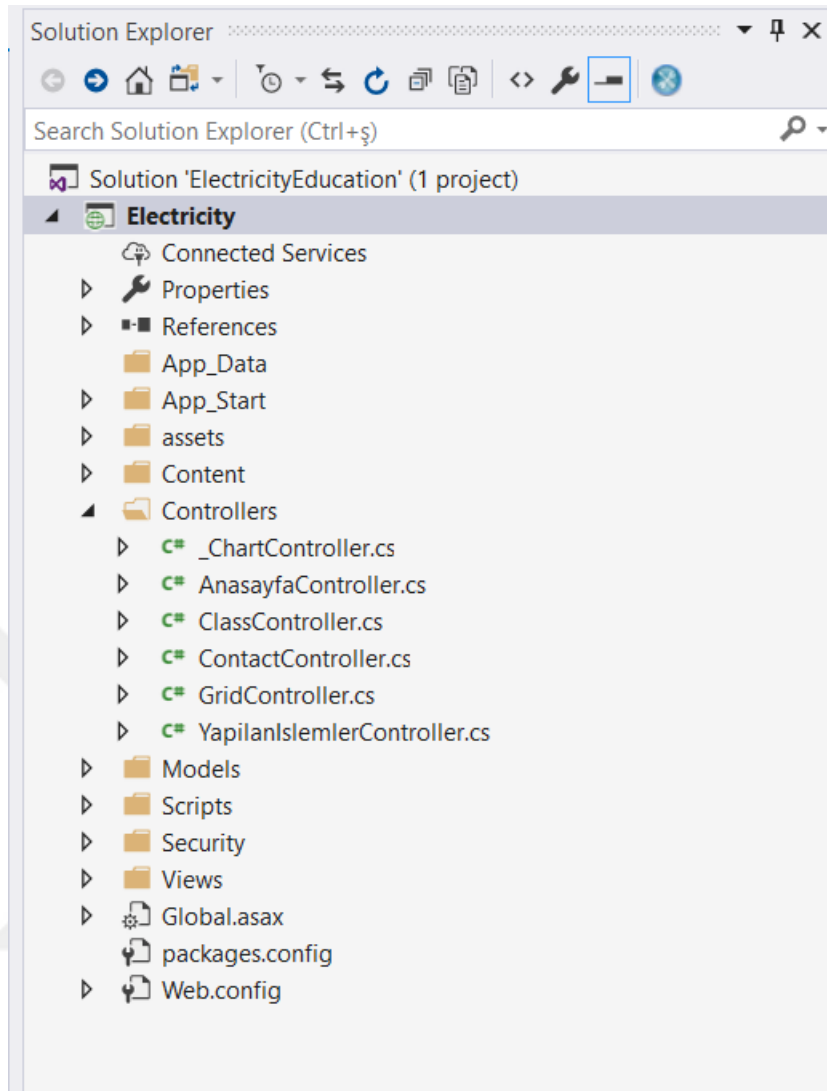
        //sistem yönünü göndereceğim...
        return View();
    }
    [AllowAnonymous]
    0 references
    public ActionResult login()
    {

        return View();
    }

    [AllowAnonymous]
    [HttpPost]
    0 references
    public ActionResult Login(models.LoginModel model)
    {

        if (ModelState.IsValid)
```





```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace ElektrikPiyasaSimulatoru
{
    public partial class LoginSayfasiEkran : Form
    {
        Educate db;
        public LoginSayfasiEkran()
        {
            InitializeComponent();
        }

        public bool giris(string kullanicadi, string kullanicisifre)
        {
            db = new Educate();
            kullanicilar kullanici = new kullanicilar();
            var context = db;
            var query = context.kullanicilar.ToList();
            var result = query.Where(e => e.kullanicadi == txtBoxKullaniciAdi.Text && e.sifre == txtSifre.Text).ToList();

            if (result.Count != 0)
            {
                return true;
            }
            else
            {
                return false;
            }
        }

        private void button1_Click(object sender, EventArgs e)
        {
            //deneme
            if (giris(txtBoxKullaniciAdi.Text, txtSifre.Text) != true)
            {
                MessageBox.Show("Hatalı Kullanıcı adı/Şifresi !");
            }
            return;
            KullaniciProfili_Ana_Ekran yeniEkran = new KullaniciProfili_Ana_Ekran();
            yeniEkran.kullanicadi = txtBoxKullaniciAdi.Text;
            yeniEkran.Show();
            this.Hide();
        }

        private void LoginSayfasiEkran_Load(object sender, EventArgs e)
        {
        }

        private void txtBoxKullaniciAdi_TextChanged(object sender, EventArgs e)
        {
        }

        private void LoginSayfasiEkran_FormClosing(object sender, FormClosingEventArgs e)
        {
        }
    }
}

```

```

        DialogResult giriskapanis = MessageBox.Show("Çıkmak İstediginizden Emin misiniz? ", "Çıkış", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
        if (giriskapanis == DialogResult.No)
        {
            e.Cancel = true;
        }
        return;
    }
    Environment.Exit(0);
}

private void hakkındaToolStripMenuItem_Click(object sender, EventArgs e)
{
    hakkında ekranHakkinda = new hakkında();
    ekranHakkinda.Show();
}
}
}

```

KullaniciProfili_Ana_Ekran.cs

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Windows.Forms;
using System.Web;
using System.Windows.Forms.DataVisualization.Charting;

namespace ElektrikPiyasaSimulatoru
{
    public partial class KullaniciProfili_Ana_Ekran : Form
    {
        Educate db;
        int sistemyon = 0;
        public KullaniciProfili_Ana_Ekran()
        {
            InitializeComponent();
        }

        public string kullanıcıAdi;
        public string sifre;
        public string tckn;
        public string kullanicitur;
        private void btnOpenBrowser_Click(object sender, EventArgs e)
        {
            System.Diagnostics.Process.Start("http://www.google.com");
        }

        private void dataGridView1_CellContentClick(object sender, DataGridViewCellEventArgs e)
        {
        }

        private void linkLabel4_LinkClicked(object sender, LinkLabelLinkClickedEventArgs e)
        {
        }
    }
}

```

```

}

private void KullaniciProfili_Ana_Ekran_Load(object sender, EventArgs e)
{
    //picture box gidip gelmesi...
    timerSistemYon.Start();
    timerSistemYon.Enabled = true;
    timerSistemYon.Interval = 1100;

    lblInputAd.Text = kullanıcıAdi;

    db = new Educate();
    kullanicilar kullanıcı = new kullanicilar();
    var context = db;
    var query = context.kullanicilar.ToList();
    var result = query.Where(b => b.kullaniciadi == kullanıcıAdi).ToList();
    lblInputC.Text = result[0].tcnk;
    lblKindOfPerson.Text = result[0].kullanicitur;
    lblKayanYaziPTF.Text = "";
    lblKayanYaziPTF.Font = new Font(lblKayanYaziPTF.Font, FontStyle.Bold);
    timerKayanYazi.Enabled = true;
    lblSmfkayanyazi.Text = "SMF";
    lblAofFiyatıKayanYazi.Text = "AOF";
    if (result[0].kullanicitur != "Admin")
    {
        btnPtf.Visible = false;
        nmrcsaat.Visible = false;
    }
    timerKayanYazi.Interval = 500;

    //timer renk değişimi
    timerRenk.Start();
    timerRenk.Enabled = true;
    timerRenk.Interval = 1000;

    //if adminse
    //sistem yönüne göre işlem yapılacak...

    var sistemContext = db.sistemyonu.ToList();
    var sistemYonuBul = sistemContext.Where(beta => beta.saat == "1" &&
beta.log_date == DateTime.Now.Date).ToList();
    if (sistemYonuBul.Count > 0)
    {
        if (sistemYonuBul[0].sistemYonu1 == "In")
        {
            pctrSistemYon.Image = ElektrikPiyasaSimulatoru.EletkrikRes.up;
            sistemyon = 1;
        }
        else if (sistemYonuBul[0].sistemYonu1 == "Out")
        {
            pctrSistemYon.Image = ElektrikPiyasaSimulatoru.EletkrikRes.down;
            sistemyon = -1;
        }
        else
        {
            //sabittir dengededir...
            pctrSistemYon.Image = ElektrikPiyasaSimulatoru.EletkrikRes.sabit;
            sistemyon = 0;
        }
    }
}

```

```

    }
}

gopuCagir();
gipicagir();
dgpCagir();
ChartGip.Visible = false;
ChartDgp.Visible = false;
}

private void timerKayanYazi_Tick(object sender, EventArgs e)
{
    db = new Educate();
    var context = db;
    var query = context.ptfkayit.ToList();
    var result = query.Where(g => Convert.ToDecimal(g.PTF) > 0).ToList();
    if (result.Count > 0)
    {
        if (string.IsNullOrEmpty(lblKayanYaziPTF.Text))
        {
            for (int i = 0; i < result.Count; i++)
            {
                //lblKayanYaziPTF.Text += result[i].saat + ": " +
result[i].PTF + "-";
                lblKayanYaziPTF.Text += result[i].saat + ": " + result[i].PTF
+ ", ";
            }
        }

        lblKayanYaziPTF.Text = this.lblKayanYaziPTF.Text.Substring(1) +
this.lblKayanYaziPTF.Text.Substring(0, 1);

        lblSmfkayanyazi.Text = this.lblSmfkayanyazi.Text.Substring(1) +
this.lblSmfkayanyazi.Text.Substring(0, 1);
        lblAofFiyatıKayanYazi.Text =
this.lblAofFiyatıKayanYazi.Text.Substring(1) +
this.lblAofFiyatıKayanYazi.Text.Substring(0, 1);
    }
}

private void lblAofFiyatı_Click(object sender, EventArgs e)
{
}

private void btnKullaniciTanimla_Click(object sender, EventArgs e)
{
    KullaniciTanimlama kullaniciTanim = new KullaniciTanimlama();
    kullaniciTanim.kullaniciadi = kullaniciAdi;
    db = new Educate();
    var context = db;
    var query = context.kullanicilar.ToList();
    var result = query.Where(b => b.kullaniciadi == kullaniciAdi).ToList();
    if (result[0].kullanicitur == "A")
    {
        kullaniciTanim.Show();
        this.Hide();
    }
    else { MessageBox.Show("Admin yetkisi olanlar görebilir"); }
}
}

```

```

private void btnGop_Click(object sender, EventArgs e)
{
    GunOncesiPiyasaEkran gunOncesiPiyasaEkran = new GunOncesiPiyasaEkran();
    gunOncesiPiyasaEkran.kullaniciadi = kullaniciAdi;
    gunOncesiPiyasaEkran.Show();
    this.Hide();
}

private void button1_Click(object sender, EventArgs e)
{
    PMS pms = new PMS();
    pms.kullaniciadi = kullaniciAdi;
    pms.Show();
    this.Hide();
}

private void btnGip_Click(object sender, EventArgs e)
{
    GunIciPiyasaEkran gunIciPiyasaEkran = new GunIciPiyasaEkran();
    gunIciPiyasaEkran.kullaniciadi = kullaniciAdi;
    gunIciPiyasaEkran.Show();
    this.Hide();
}

private void btnSeffaflikPlatform_Click(object sender, EventArgs e)
{
    Ozet ozet = new Ozet();
    ozet.kullaniciadi = kullaniciAdi;
    ozet.Show();
    this.Hide();
}

private void btnTuketim_Click(object sender, EventArgs e)
{
    TUKETIM tuketim = new TUKETIM();
    tuketim.kullaniciadi = kullaniciAdi;
    this.Hide();
    tuketim.Show();
}

private void btnuretim_Click(object sender, EventArgs e)
{
    URETIM uretim = new URETIM();
    uretim.Show();
    this.Hide();
    uretim.kullaniciadi = kullaniciAdi;
}

private void btnDgp_Click(object sender, EventArgs e)
{
    DGP dgp = new DGP();
    dgp.kullaniciadi = kullaniciAdi;
    dgp.Show();
    this.Hide();
}

private void idTekliflerim_Click(object sender, EventArgs e)
{
    Teklifler yeniEkran = new Teklifler();
}

```

```

        yeniEkran.kullaniciAdi = kullaniciAdi;
        yeniEkran.Show();
        this.Hide();
    }

    private void lblKindOfPerson_Click(object sender, EventArgs e)
    {

    }

    private void btnTakasfiyat_Click(object sender, EventArgs e)
    {
        PTF yeniEkran = new PTF();
        yeniEkran.Show();
        this.Hide();

        yeniEkran.kullaniciadi = kullaniciAdi;
    }

    private void btnPtf_Click(object sender, EventArgs e)
    {
        //ptf tablosundan kayıtları çek
        decimal ptffiyati = 0;
        int counter = 0;
        db = new Educate();
        var context = db;
        var query = context.ptfkayit.ToList();
        var queryEslemeTuketici = context.tuketiciteklif.ToList();
        var queryEslesmeUretici = context.ureticiteklif.ToList();

        int paramSaat = DateTime.Now.Hour;
        DateTime paramDate = DateTime.Now;
        var result = query.Where(b => b.saat == paramSaat.ToString() && b.date ==
paramDate.ToString() && b.PTF != "0").ToList();
        //saatlik bazda kontrol edilecek varsa tekrardan oluşturulmayacak..
        if (result.Count != 0)
        {
            MessageBox.Show("PTF fiyatı bugün ve saat için belirlenmiştir!");
            MessageBox.Show("PTF Fiyatı : " + result[0].PTF + "dir");
            return;
        }
        else
        {
            //PTF OLUŞTURUYORUZ...
            var size = 0;
            //0 ile 2000 tl arasında oynayan fiyat baz alındığında adetle
çarpılacak bir adet 1 lot =0,1 mwh denk gelir.
            tuketiciteklif tuketiciTeklif = new tuketiciteklif();
            ureticiteklif ureticiTeklif = new ureticiteklif();
            List<tuketiciteklif> ptfTeklif = new List<tuketiciteklif>();
            var resultEslemeTuketiciPtf = queryEslemeTuketici.Where(b => b.saat
== nmrcsaat.SelectedText && b.log_date == paramDate).ToList();
            var resultEslesmeUreticiPtf = queryEslesmeUretici.Where(b => b.saat ==
nmrcsaat.SelectedText.ToString() && b.log_date == paramDate).ToList();
            if (resultEslemeTuketiciPtf.Count > resultEslesmeUreticiPtf.Count)
                size = resultEslemeTuketiciPtf.Count;
            if (resultEslesmeTuketiciPtf.Count == 0 ||
resultEslesmeUreticiPtf.Count == 0)
            {
                MessageBox.Show("PTF fiyatı oluşturulamadı!");
                return;
            }
            size = resultEslesmeUreticiPtf.Count;

```

```

        for (int i = 0; i < size; i++)
        {
            var resultPtf = resultEslesmeTuketiciPtf.Where(x =>
x.teklif_birim_fiyat == resultEslesmeUreticiPtf[i].teklif_birim_fiyat).ToList();
            if (resultPtf[0] != null)
            {
                ptffiyati +=
Convert.ToDecimal(resultPtf[0].teklif_birim_fiyat);
                counter++;
            }
        }

        //fiyat oluşturu insert atacağız
        db = new Educate();
        ptfkayit = new ptfkayit();
        ptfkayit.ADT = "0"; //bu fiyatı gip oluşturulunca ortalama alınarak
oluşacak olan fiyattır.
        ptfkayit.date = Convert.ToString(DateTime.Today);
        if (counter != 0)
            ptfkayit.PTF = (ptffiyati / counter).ToString();
        ptfkayit.PTF = (ptffiyati).ToString();
        ptfkayit.saat = DateTime.Now.Hour.ToString(); // "1";
        ptfkayit.SMF = "0"; //bu fiyatı gip oluşturulunca oluşacak olan
fiyattır.
        db.ptfkayit.Add(ptfkayit);
        db.SaveChanges();
        //nutable yapıyorum..
        //PTF BAĞLI EŞLEŞTİRMELERİ OLUŞTURACAĞIMIZ YER...
        var resultEslesmeTuketici = queryEslemeTuketici.Where(b => b.saat ==
paramSaat.ToString() && b.log_date == paramDate && result[0].PTF != "0" &&
Convert.ToDecimal(result[0].PTF)
        <
Convert.ToDecimal(tuketici.teklif.teklif_birim_fiyat)).ToList();
        //üst olanlar eşleşme olacak demekki
        var resultEslesmeUretici = queryEslesmeUretici.Where(b => b.saat ==
paramSaat.ToString() && b.log_date == paramDate && result[0].PTF != "0" &&
Convert.ToDecimal(result[0].PTF)
        >
Convert.ToDecimal(ureticici.teklif.teklif_birim_fiyat)).ToList();
        //Tüketici için ptf fiyatına eşit ve üstündeki fiyatlarını eşleştir
        //öncelikle fiyata göre sıralanır fiyat eşitliğinde dakika saniye
farkına bakılır...
        //Üretici için ptf fiyatına eşit ve altındakiler eşleşir...
        //order

        //eşleşmeler o saat aralığında baştan itibaren fiyat bazında aşağıya
doğru gelerek eşleşir
        if (resultEslesmeTuketici.Count > 0 && resultEslesmeUretici.Count > 0)
        {
            var orderEslesmeTuketici =
resultEslesmeTuketici.OrderByDescending(oguz => oguz.teklif_birim_fiyat).OrderBy(oguz
=> oguz.log_date).ToList();
            var orderEslesmeUretici = resultEslesmeUretici.OrderBy(oguz =>
oguz.teklif_birim_fiyat).OrderByDescending(oguz => oguz.log_date).ToList();

            //Sistemde yük açığı var mı ? PTF ye giren yükler...

            var toplamUretim = resultEslesmeUretici.Sum(x =>
Convert.ToInt64(x.adet));
            var toplamTuketim = resultEslesmeTuketici.Sum(z =>
Convert.ToInt64(z.adet));

            long yukEslesme = 0;

```

```

        //long kalan = 0;

        var Sonuc = toplamUretim - toplamTuketim; // Toplam üretim ve
        tüketim arz veya talep fazlasını anlayacağımız yer burası...
        if (Sonuc > 0)//Sistem yük fazlası var
        {
            sistemyonu sitemYonu = new sistemyonu();
            sitemYonu.log_date = paramDate;
            sitemYonu.saat = nmrcsaat.SelectedText;
            sitemYonu.sistemYonu1 = "In";
            db.sistemyonu.Add(sitemYonu);
            db.SaveChanges();
        }
        else if (Sonuc < 0) //Sistem alış yönünde
            //dengede
        {
            sistemyonu sitemYonu = new sistemyonu();
            sitemYonu.log_date = paramDate;
            sitemYonu.saat = nmrcsaat.SelectedText;
            sitemYonu.sistemYonu1 = "Out";
            db.sistemyonu.Add(sitemYonu);
            db.SaveChanges();
        }
        else //stabil sabit düz
        {
            sistemyonu sitemYonu = new sistemyonu();
            sitemYonu.log_date = paramDate;
            sitemYonu.saat = nmrcsaat.SelectedText;
            sitemYonu.sistemYonu1 = "Notr";
            db.sistemyonu.Add(sitemYonu);
            db.SaveChanges();
        }
    }

    //if(Sonuc==0) //Uretim tüketim dengede hepsi eşleşecek

    //EŞLEŞME BAŞLASIN...

    int i = 0;
    for (int j = 0; j < orderEslesmeTuketici.Count; j++)//çünkü arz
    fazlası var dolayısıyla tüketici tarafına bakmak mantıklı...
    {
        yukEslesme = Convert.ToInt64(orderEslesmeUretici[i].adet) -
        Convert.ToInt64(orderEslesmeTuketici[j].adet);

        if (yukEslesme == 0) //o sa ilk kayıt tam eşleşme
        sağladı...Kayıt atabiliriz...
        {
            eslesenteklifler eslesenTekliflerData = new
            eslesenteklifler();
            eslesenTekliflerData.UreticiID =
            orderEslesmeUretici[i].kullaniciadi;
            eslesenTekliflerData.TuketiciID =
            orderEslesmeTuketici[j].kullaniciadi;//tüketici idsi
            eslesenTekliflerData.eslesenMiktar =
            orderEslesmeUretici[i].adet; //eşleşme miktarı
            eslesenTekliflerData.fiyat =
            orderEslesmeUretici[i].teklif_birim_fiyat; //gün için ptf fiyatıdır.
            eslesenTekliflerData.gun =
            orderEslesmeUretici[i].log_date.ToString();//gün

```

```

        eslesenTekliflerData.saat =
orderEslesmeUretici[i].saat.ToString();// saat bilgisi
        eslesenTekliflerData.U_adet =
Convert.ToInt32(orderEslesmeUretici[i].adet);//üretici adet..
        eslesenTekliflerData.T_adet =
Convert.ToInt32(orderEslesmeTuketici[j].adet);
        eslesenTekliflerData.eslemeID =
orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici
ve tüketici id ile birlikte gün ve saat ile kaydolun uniqueiddir.

        db.eslesenteklifler.Add(eslesenTekliflerData);
        db.SaveChanges();
        i++; // üretici ve tüketici için ikinci alıcı ve satıcıya
geçiliyor diyoruz...
    }
    else if (yukEslesme < 0 || yukEslesme > 0)
    {
        do
        {
            eslesenteklifler eslesenTeklifleryukKucuk = new
eslesenteklifler();
            eslesenTeklifleryukKucuk.eslemeID =
orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici
ve tüketici id ile birlikte gün ve saat ile kaydolun uniqueiddir.

            if (Convert.ToInt32(orderEslesmeUretici[i].adet) -
Convert.ToInt32(orderEslesmeTuketici[j].adet) == 0)
            {
                eslesenTeklifleryukKucuk.eslesenMiktar =
orderEslesmeUretici[i].adet; //denge var dolayısıyla hepsi eşleşecek... burdan çıkıp
fora girece dolayısıyla iki taraf içinde

                eslesenTeklifleryukKucuk.UreticiID =
orderEslesmeUretici[i].kullaniciadi;
                eslesenTeklifleryukKucuk.TuketiciID =
orderEslesmeTuketici[j].kullaniciadi;//tüketici idsi
                //eslesenTeklifleryukKucuk.eslesenMiktar =
orderEslesmeUretici[i].adet; //eşleşme miktarı
                eslesenTeklifleryukKucuk.fiyat =
orderEslesmeUretici[i].teklif_birim_fiyat; //gün için ptf fiyatıdır.
                eslesenTeklifleryukKucuk.gun =
orderEslesmeUretici[i].log_date.ToString();//gün
                eslesenTeklifleryukKucuk.saat =
orderEslesmeUretici[i].saat.ToString();// saat bilgisi
                eslesenTeklifleryukKucuk.U_adet =
Convert.ToInt32(orderEslesmeUretici[i].adet);//üretici adet..
                eslesenTeklifleryukKucuk.T_adet =
Convert.ToInt32(orderEslesmeTuketici[j].adet);
                eslesenTeklifleryukKucuk.eslemeID =
orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici
ve tüketici id ile birlikte gün ve saat ile kaydolun uniqueiddir.

                db.eslesenteklifler.Add(eslesenTeklifleryukKucuk);
                db.SaveChanges();

                // diğer kayıtlara atlayacak...
                i++;
            }
        }
    }
}

```

```

        if (Convert.ToInt32(orderEslesmeUretici[i].adet) -
Convert.ToInt32(orderEslesmeTuketici[j].adet) > 0)
        {
            eslesenTeklifleryukKucuk.eslesenMiktar =
orderEslesmeTuketici[j].adet; //tuketici teklifinden büyük olduğu için tüketilenin
hepsini

//veririz... eşleşme sağlanır bir sonraki durum için kalandan devam edilecek
dolayısıyla ona göre kalanı uretime atacagız..

//ureticini artmasına gerek yok...
            orderEslesmeUretici[i].adet =
(Convert.ToInt32(orderEslesmeUretici[i].adet) -
Convert.ToInt32(orderEslesmeTuketici[j].adet)).ToString();

            eslesenTeklifleryukKucuk.UreticiID =
orderEslesmeUretici[i].kullaniciadi;
            eslesenTeklifleryukKucuk.TuketiciID =
orderEslesmeTuketici[j].kullaniciadi; //tuketici idsi
            //eslesenTeklifleryukKucuk.eslesenMiktar =
orderEslesmeUretici[i].adet; //eşleşme miktarı
            eslesenTeklifleryukKucuk.fiyat =
orderEslesmeUretici[i].teklif_birim_fiyat; //gün için ptf fiyatıdır.
            eslesenTeklifleryukKucuk.gun =
orderEslesmeUretici[i].log_date.ToString(); //gün
            eslesenTeklifleryukKucuk.saat =
orderEslesmeUretici[i].saat.ToString(); // saat bilgisi
            eslesenTeklifleryukKucuk.U_adet =
Convert.ToInt32(orderEslesmeUretici[i].adet); //üretici adet..
            eslesenTeklifleryukKucuk.T_adet =
Convert.ToInt32(orderEslesmeTuketici[j].adet);
            eslesenTeklifleryukKucuk.eslemeID =
orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici
ve tüketici id ile birlikte gün ve saat ile kaydolun uniqueiddir.

            db.eslesenteklifler.Add(eslesenTeklifleryukKucuk);
            db.SaveChanges();

        }

        if (Convert.ToInt32(orderEslesmeUretici[i].adet) -
Convert.ToInt32(orderEslesmeTuketici[j].adet) < 0)
        {
            //demekki ikinci,üçüncü üreticinin malınada gelinecek
burada dolayısıyla tüketicinin istegini sıfırlayana kadar döneceğiz..
            {
                eslesenTeklifleryukKucuk.eslesenMiktar =
orderEslesmeUretici[i].adet; //tüketicinin üreticiden fazlası var o yüzden üretici
kadar atılıp ikinci üreticinin malını satıyoruz...

                eslesenTeklifleryukKucuk.UreticiID =
orderEslesmeUretici[i].kullaniciadi;
                eslesenTeklifleryukKucuk.TuketiciID =
orderEslesmeTuketici[j].kullaniciadi; //tuketici idsi
                //eslesenTeklifleryukKucuk.eslesenMiktar =
orderEslesmeUretici[i].adet; //eşleşme miktarı
                eslesenTeklifleryukKucuk.fiyat =
orderEslesmeUretici[i].teklif_birim_fiyat; //gün için ptf fiyatıdır.
                eslesenTeklifleryukKucuk.gun =
orderEslesmeUretici[i].log_date.ToString(); //gün

```

```

        eslesenTeklifleryukKucuk.saat =
orderEslesmeUretici[i].saat.ToString();// saat bilgisi
        eslesenTeklifleryukKucuk.U_adet =
Convert.ToInt32(orderEslesmeUretici[i].adet);//üretici adet..
        eslesenTeklifleryukKucuk.T_adet =
Convert.ToInt32(orderEslesmeTuketici[j].adet);
        eslesenTeklifleryukKucuk.eslemeID =
orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici
ve tüketici id ile birlikte gün ve saat ile kaydolan uniqueiddir.

        db.eslesenteklifler.Add(eslesenTeklifleryukKucuk);
        db.SaveChanges();

        orderEslesmeTuketici[j].adet = "0";

        i++; //ikinci kayıtla gelecek buraya ....
    }

        eslesenTeklifleryukKucuk.UreticiID =
orderEslesmeUretici[i].kullaniciadi;
        eslesenTeklifleryukKucuk.TuketiciID =
orderEslesmeTuketici[j].kullaniciadi;//tüketici idsi
        //eslesenTeklifleryukKucuk.eslesenMiktar =
orderEslesmeUretici[i].adet; //eşleşme miktarı
        eslesenTeklifleryukKucuk.fiyat =
orderEslesmeUretici[i].teklif_birim_fiyat; //gün için ptf fiyatıdır.
        eslesenTeklifleryukKucuk.gun =
orderEslesmeUretici[i].log_date.ToString();//gün
        eslesenTeklifleryukKucuk.saat =
orderEslesmeUretici[i].saat.ToString();// saat bilgisi
        eslesenTeklifleryukKucuk.U_adet =
Convert.ToInt32(orderEslesmeUretici[i].adet);//üretici adet..
        eslesenTeklifleryukKucuk.T_adet =
Convert.ToInt32(orderEslesmeTuketici[j].adet);
        eslesenTeklifleryukKucuk.eslemeID =
orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici
ve tüketici id ile birlikte gün ve saat ile kaydolan uniqueiddir.

        db.eslesenteklifler.Add(eslesenTeklifleryukKucuk);
        db.SaveChanges();

    }
    while
    (
        Convert.ToInt32(orderEslesmeUretici[i].adet) -
Convert.ToInt32(orderEslesmeTuketici[j].adet) < 0 && i <= orderEslesmeUretici.Count);
    }
}

}

else
{
    MessageBox.Show("Piyasa Takaf Fiyatı Oluşturulamadı!");
}

btnTakasfiyat_Click(sender, e);
}

```

```

    }

    private void lblInputTc_Click(object sender, EventArgs e)
    {

    }

    private void lbl_Tc_Click(object sender, EventArgs e)
    {

    }

    private void timerRenk_Tick(object sender, EventArgs e)
    {
        //btnPtF.Visible = !(btnPtF.Visible);
        if (btnPtF.ForeColor == Color.Red)
        {
            btnPtF.ForeColor = Color.Black;
        }
        else
        {
            btnPtF.ForeColor = Color.Red;
        }
        pctrSistemYon.Image = null;
    }

    private void nmrcsaat_ValueChanged(object sender, EventArgs e)
    {
        //if(nmrcsaat<0 || nmrcsaat.Value>24)
        //{ nmrcsaat.Value = 0; }
    }

    private void btnKullaniciGuncelle_Click(object sender, EventArgs e)
    {
        KullaniciGuncelleme kullaniciYanim = new KullaniciGuncelleme();
        kullaniciYanim.kullaniciadi = kullaniciAdi;
        db = new Educate();
        var context = db;
        var query = context.kullanicilar.ToList();
        var result = query.Where(b => b.kullaniciadi == kullaniciAdi).ToList();
        if (result[0].kullanicitur == "A")
        {
            kullaniciYanim.Show();
            this.Hide();
        }
        else { MessageBox.Show("Admin yetkisi olanlar görebilir"); }
    }

    private void pictureBox1_Click(object sender, EventArgs e)
    {

    }

    //int gop = 0;
    private void lblgopgrafik_Click(object sender, EventArgs e)
    {
        lblgopgrafik.Font = new Font(lblgopgrafik.Font, FontStyle.Underline);
        lblgopgrafik.BackColor = Color.Red;

        //lblGipGrafik.Font = new Font(lblGipGrafik.Font, FontStyle.Underline);
        //lblGipGrafik.BackColor = Color.Red;
    }

```

```

        Chart.Visible = true;
        ChartGip.Visible = false;
        ChartDgp.Visible = false;
        //gopuÇagir();
        //gopu çağır()
    }

    public void gopuÇagir()
    {
        //alt grafik yazı?
        lblgopgrafik.Font = new Font(lblgopgrafik.Font, FontStyle.Underline);
        lblgopgrafik.BackColor = Color.Red;

        //Kullanıcı ilk girdiğinde alttaki grafikler için çizim...
        //db = new Educatelectricity();
        ureticiteklif kullaniciTeklif = new ureticiteklif();
        db = new Educate();

        var kayitUretici = db.ureticiteklif.ToList();
        var teklifler = kayitUretici.Where(oguzEnt => oguzEnt.durum != "E" ||
oguzEnt.durum != "P").ToList();
        //for (int i = 0; i < teklifler.Count; i++)
        //var teklifUretim = teklifler.Where(q => q.teklif_birim_fiyat ==
teklifler[i].teklif_birim_fiyat).ToList();
        //tüketici teklif
        var kayitTuketici = db.tuketiciteklif.ToList();
        var kayitTuketicilist = kayitTuketici.Where(oguzTuket => oguzTuket.durum
!= "E" || oguzTuket.durum != "P").ToList();

        Series series = Chart.Series.Add("Arz");
        this.Chart.Titles.Add(" Gün Öncesi Piyasası (PtF) ");
        series.ChartType = SeriesChartType.Spline;

        for (int counter = 0; counter < teklifler.Count; counter++)
        {
            series.Points.AddXY(teklifler[counter].teklif_birim_fiyat,
teklifler[counter].adet); }

        Series S2 = new Series("Talep");
        S2.ChartType = SeriesChartType.Spline;

        for (int counter = 0; counter < kayitTuketicilist.Count; counter++)
        {
            S2.Points.AddXY(kayitTuketicilist[counter].teklif_birim_fiyat,
kayitTuketicilist[counter].adet); }

        Chart.Series.Add(S2);
        S2.Color = Color.Purple;
    }
    void gipicagir()
    {
        lblgopgrafik.Font = new Font(lblgopgrafik.Font, FontStyle.Bold);
        lblgopgrafik.BackColor = Color.Silver;

        //Kullanıcı ilk girdiğinde alttaki grafikler için çizim...
        //db = new Educatelectricity();
        ureticiteklif kullaniciTeklif = new ureticiteklif();
        db = new Educate();
    }
}

```

```

        //if (tiklandi == 0)
        //{
            var kayitUretici = db.ureticiteklif.ToList();
            var teklifler = kayitUretici.Where(oguzEnt => oguzEnt.durum != "E" ||
oguzEnt.durum != "P").ToList();
            //for (int i = 0; i < teklifler.Count; i++)
            //var teklifUretim = teklifler.Where(q => q.teklif_birim_fiyat ==
teklifler[i].teklif_birim_fiyat).ToList();
            //tuketici teklif
            var kayitTuketici = db.tuketiciteklif.ToList();
            var kayitTuketicilist = kayitTuketici.Where(oguzTuket => oguzTuket.durum
!= "E" || oguzTuket.durum != "P").ToList();

            Series seriesgip = ChartGip.Series.Add("Gip");
            this.ChartGip.Titles.Add(" Gün Öncesi Piyasası (PtF) ");
            seriesgip.ChartType = SeriesChartType.Spline;

            for (int counter = 0; counter < teklifler.Count; counter++)
            {
                seriesgip.Points.AddXY(teklifler[counter].teklif_birim_fiyat,
teklifler[counter].adet); }

            seriesgip.Color = Color.Red;

        }

        public void dgpCagir()
        {

        }

        //private void ptfChart_Click(object sender, EventArgs e)
        //{

        //}
        //int tiklandi = 0;
        private void lblGipGrafik_Click(object sender, EventArgs e)
        {

            //lblGipGrafik.Font = new Font(lblgopgrafik.Font)
            lblGipGrafik.Font = new Font(lblGipGrafik.Font, FontStyle.Underline);
            lblGipGrafik.BackColor = Color.Red;

            ChartGip.Visible = true;
            Chart.Visible = false;
            ChartDgp.Visible = false;
            //lblgopgrafik.Font = new Font(lblgopgrafik.Font, FontStyle.Bold);
            //lblgopgrafik.BackColor = Color.Silver;

            ////Kullanıcı ilk girdiğinde alttaki grafikler için çizim...
            ////db = new Educatelectricity();
            //UreticiTeklif kullaniciTeklif = new UreticiTeklif();
            //db = new Educatelectricity();

        }

        private void lblDGPGratik_Click(object sender, EventArgs e)
        {
            lblDGPGratik.Font = new Font(lblDGPGratik.Font, FontStyle.Underline);
            ChartDgp.Visible = true;
            Chart.Visible = false;

```

```

        ChartGip.Visible = false;
    }

    private void btnRaporlar_Click(object sender, EventArgs e)
    {
        SeffaflikAnasayfa seffafRapor = new SeffaflikAnasayfa();
        seffafRapor.Show();
        seffafRapor.kullaniciadi = kullanıcıAdi;
        this.Hide();
    }

    private void KullaniciProfili_Ana_Ekran_FormClosing(object sender,
    FormClosingEventArgs e)
    {
        DialogResult giriskapanis = MessageBox.Show("Çıkmak İstediginizden Emin
        misiniz? ", "Çıkış", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
        if (giriskapanis == DialogResult.No)
        {
            e.Cancel = true;
            return;
        }
        Environment.Exit(0);
    }

    private void btnRefresh_Click(object sender, EventArgs e)
    {
    }

    private void KullaniciProfili_Ana_Ekran_Load_1(object sender, EventArgs e)
    {
    }

    private void btnrefresh_Click_1(object sender, EventArgs e)
    {
        KullaniciProfili_Ana_Ekran_Load(null, null);
    }

    private void timerSistemYon_Tick(object sender, EventArgs e)
    {
        //btnPtf.Visible = !(btnPtf.Visible);
        if (sistemyon == 1)
            pctrSistemYon.Image = ElektrikPiyasaSimulatoru.EletkrikRes.up;
        else if (sistemyon == -1)
            pctrSistemYon.Image = ElektrikPiyasaSimulatoru.EletkrikRes.down;
        else
            pctrSistemYon.Image = ElektrikPiyasaSimulatoru.EletkrikRes.sabit;
    }

    private void btnGeri_Click(object sender, EventArgs e)
    {
        LoginSayfasiEkran yeniEkran = new LoginSayfasiEkran();
        //yeniEkran.kullaniciAdi = kullanıcıAdi;
        this.Hide();
    }

```

```

        yeniEkran.Show();
    }
    //fiyatı belirle eşleştirmeleri yap... ptf üstündeki alışıta önemli olan düşük
fiyattan verilmesi
    //dolayısıyla eşleşmeler yüksekte alıcı verildiğinde eşleşir

}

}

```

```

var Sonuc = toplamTucetici - toplamTuketici; // toplam alıcım ve tüketicim bir veya daha fazla tüketici bulduğumuzda yer bulamaz...
if (Sonuc > 0) //Sistem yük fazlası var
{
    sistemYonu sitemYonu = new sistemYonu();
    sitemYonu.log_date = paramDate;
    sitemYonu.saat = saat;
    sitemYonu.sistemYonul = "In";
    db.sistemYonu.Add(sitemYonu);
    db.SaveChanges();
}
else if (Sonuc < 0) //Sistem alış yönünde
//dengede
{
    sistemYonu sitemYonu = new sistemYonu();
    sitemYonu.log_date = paramDate;
    sitemYonu.saat = saat;
    sitemYonu.sistemYonul = "Out";
    db.sistemYonu.Add(sitemYonu);
    db.SaveChanges();
}
else //stabil sabit düz
{
    sistemYonu sitemYonu = new sistemYonu();
    sitemYonu.log_date = paramDate;
    sitemYonu.saat = saat;
    sitemYonu.sistemYonul = "Notr";
    db.sistemYonu.Add(sitemYonu);
    db.SaveChanges();
}
//EŞLEŞME BAŞLASIN...
if (resultEslesmeTuketici.Count < 1)
    response += "İlgili Ptf için tüketici eşleşecek kayıt bulunamadı!";

```

```

else if (yukEslesme > 0)
{
    eslesenteklifler eslesenTeklifleryukKucuk = new eslesenteklifler();
    eslesenTeklifleryukKucuk.eslemeID = orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici ve tük

    eslesenTeklifleryukKucuk.eslesenmiktar = orderEslesmeTuketici[j].adet; //tuketici teklifinden büyük olduğu için tüketilenin hep
    //veririz... eşleşme sağlanır bir sonraki durum için kalar
    //ureticini artmasına gerek yok...

    orderEslesmeUretici[i].adet = orderEslesmeTuketici[j].adet;

    eslesenTeklifleryukKucuk.uretid = orderEslesmeUretici[i].kullaniciadi;
    eslesenTeklifleryukKucuk.tuketiciID = orderEslesmeTuketici[j].kullaniciadi; //tuketici idsi
    //eslesenTeklifleryukKucuk.eslesenMiktar = orderEslesmeTuketici[j].adet;
    eslesenTeklifleryukKucuk.fiyat = orderEslesmeUretici[i].teklif_birim_fiyat; //gün için ptf fiyatıdır.
    eslesenTeklifleryukKucuk.gun = orderEslesmeUretici[i].log_date.ToString(); //gün
    eslesenTeklifleryukKucuk.saat = orderEslesmeUretici[i].saat.ToString(); // saat bilgisi
    eslesenTeklifleryukKucuk.U_adet = Convert.ToInt32(orderEslesmeUretici[i].adet); //üretici adet..
    eslesenTeklifleryukKucuk.T_adet = Convert.ToInt32(orderEslesmeTuketici[j].adet);
    eslesenTeklifleryukKucuk.eslemeID = orderEslesmeUretici[i].kullaniciadi + orderEslesmeTuketici[j].kullaniciadi; //üretici ve tük

    db.eslesenteklifler.Add(eslesenTeklifleryukKucuk);
    db.SaveChanges();
    orderEslesmeTuketici[j].adet = 0;
    orderEslesmeUretici[i].adet = orderEslesmeUretici[i].adet - orderEslesmeTuketici[j].adet;
    j++;
    //}
}
else if (yukEslesme < 0)
{
    //if (Convert.ToInt32(orderEslesmeUretici[i].adet) - Convert.ToInt32(orderEslesmeTuketici[j].adet) < 0)
    //demekki ikinci,üçüncü üreticinin malınada gelinecek burada dolayısıyla tüketicinin isteğini sıfırlayana kadar döneceğiz..
    {
        eslesenteklifler eslesenTeklifleryukKucuk = new eslesenteklifler();

        eslesenTeklifleryukKucuk.ptf_teklifi = new eslesenTeklifleryukKucuk.ptf_teklifi();
        var resultEslesmeUreticiPtf = queryEslesmeUretici.Where(b => b.saat == saat.Remove(2,3) && b.log_date==DateTime.Now.Date).ToList(); //ilgili saat
        var resultEslesmeTuketiciPtf = queryEslesmeTuketici.Where(b=> b.log_date == DateTime.Now.Date).ToList(); //var resultEslesmeTuketiciPtf = queryEs

        if (resultEslesmeTuketiciPtf.Count > resultEslesmeUreticiPtf.Count)
        {
            size = resultEslesmeUreticiPtf.Count;
            if (resultEslesmeTuketiciPtf.Count == 0) // eşleşek kayıt yok...
            {
                return response = "Tüketici teklifi yok/Ptf oluşturulamadı/Arz Fazlası var.";
            }
        }
        else if (resultEslesmeUreticiPtf.Count == 0) // eşleşek kayıt yok...
        {
            return response = "Üretici teklifi yok/Ptf oluşturulamadı/Arz eksikliği var.";
        }
        else if (resultEslesmeTuketiciPtf.Count == 0 && resultEslesmeUreticiPtf.Count == 0) // eşleşek kayıt yok...
        {
            return response = "Üretici ve tüketici teklifi yok/Ptf oluşturulamadı/Arz ve talep eksikliği var.";
        }
    }

    //ptf oluşturma
    lineerHesaplama lineer = new lineerHesaplama();
    lineerHesaplama.line line1 = new lineerHesaplama.line();
    lineerHesaplama.line line2 = new lineerHesaplama.line();
    //verilen tekliflerden bir çizgi oluşturup kesişim yerini ptf ilan edeceğiz

    //arz eğrisi
    line1.x1 = resultEslesmeUreticiPtf.Min(x => x.teklif_birim_fiyat); //100;
    line1.y1 = resultEslesmeUreticiPtf.Min(y => y.adet); //1000.

```

ÖZGEÇMİŞ

Kimlik Bilgileri	
Adı ve Soyadı	Oğuz ALTUNTAŞ
Doğum Tarihi ve Yeri	16/07/1990, Tonya / TRABZON

Eğitim Bilgileri	
Lise	KartalSüleyman Demirel Anadolu Lisesi,Fen Bilimleri, İstanbul 2004 – 2008
Lisans	Sakarya Üniversitesi, Mühendislik Fakültesi/Bilgisayar Mühendisliği, Sakarya 2008 - 2013
Lisans	Anadolu Üniversitesi, İşletme Fakültesi/İşletme,Eskişehir 2010 -2014
ÖnLisans	Anadolu Üniversitesi,Açıköğretim Fakültesi/Halkla İlişkiler ve Tanıtım 2010 -2012
Yüksek Lisans	Marmara Üniversitesi, Elektrik ve Elektronik Mühendisliği, İstanbul

İş Tecrübesi	
2011-2012	Sakarya Üniversitesi Öğrenci Asistan
2013-2015	KoçSistem Bilgi ve İletişim Hizmetleri A.Ş. Ar-ge ve Yenilik Merkezi/Yazılım Uzmanı
2015-2019	Albaraka Türk Katılım Bankası A.Ş. Kıdemli Yazılım Uzmanı
2019-Devam	Emlak Katılım Bankası A.Ş.