

**FIRAT ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**Yazılım Mühendisliği**



**YMH310 – Web Tasarımı ve Programlama Dersi**

**Proje Uygulaması ve Dokümantasyonu**

**ONLİNE AÇIK ARTIRMA SİTESİ**

*185541013* Abdullah AYGÜN

**Mayıs – 2021**

1. GİRİŞ

# ÖNSÖZ

Dünyayı çapında 2019 ‘dan bu yana çıkmış olan COVID19 virüsü ile tüm dünya el ele mücadele etmektedir. Böyle bir zamanda insanların yapacakları çoğu işlemler askıya alınmıştır. Bu nedenle birçok iş dijital ortama taşınmıştır. Bu nedenle yapılmış birçok dijital uygulama bu zorlu dönemde insanların hayatlarını kolaylaştırmaya yönelik olmuştur.

Birçok ülke kendine ait önceden çıkmış ya da son dönemlerde çıkan popüler uygulamaları kullanmaktadır. En çok kullanılan haberleşme uygulamalarıdır. Birçok büyük şirket kendi haberleşme ağını kurmuş ve dünya ile haberleşmeye başlamıştır. Bunun dışında bazı şirketler ise kendine has özellikleri olan salgın döneminden önce yüz yüze yapılan aktiviteleri dijital ortama taşımıştır.

# Projenin Amacı

Yüz yüze dönemlerde yapılan Açık Artırma(müzayede) olaylarının dijital ortama aktarılması ile proje, insanların oturduğu yerden istedikleri ürünlere teklifler vermesini sağlamaktadır. Böylece sınırlı bir zaman içinde bir ürüne en yüksek teklifi veren kişi kazanacaktır. Bir ürünü 1000 kişi ister ancak 1 kişi alır mantığına dayanmaktadır.

Açık artırma sistemi için aşağıdaki özel hedeflere yönelik bir yazılım geliştirilmesine ihtiyaç olduğu ortaya çıkmıştır:

* Açık artırmaya ürünler, başlangıç-bitiş tarihleri ve diğer özellikleri belirtilerek eklenmesi,
* Başlangıç tarihi gelmemiş ürünlerin erişime kapalı olması,
* Bitiş tarihinin geçmesi durumunda aynı şekilde ürüne erişimin kapalı olması,
* Başlangıç ve bitiş tarihleri arasında ürünün teklif verecek olanlara açık olması,
* Bir ürüne teklif verirken mutlaka başlangıç fiyatından yüksek olması,
* Art arda birden fazla teklif verilememesi,
* En son teklifin verilmesi durumunda teklifi veren kişiden teklif ettiği meblağının anında kesilmesi hedefleridir.

# Projenin Kapsamı

Her türlü ürün satışının gerçekleştirilmesi sağlanmıştır. Belli bir zaman aralığında para birimi türk lirası olmak kaydıyla müzayedeye katılanlar teklif verebilecektir.

# Tanımlamalar ve Kısaltmalar

*Açık Artırma:* Bir malın satış bedelini artırmak amacıyla o malı alacak kimseler arasında fiyat artırmaya dayanan satış yöntemi.

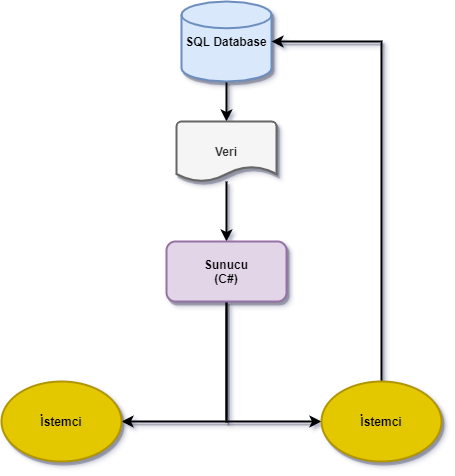
*Auction System(AS):* Açık Artırma(müzayede) Sistemi

2. PROJE PLANI

# 2.1 Giriş

Günümüz şartlarına bağlı olarak dijitalleşme çok büyük önem arz etmektedir. Bu nedenle fiziksel olarak yapabileceğimiz işlemleri dijital ortamlara aktararak daha az çaba ile daha büyük sonuçlar elde edebiliyoruz.

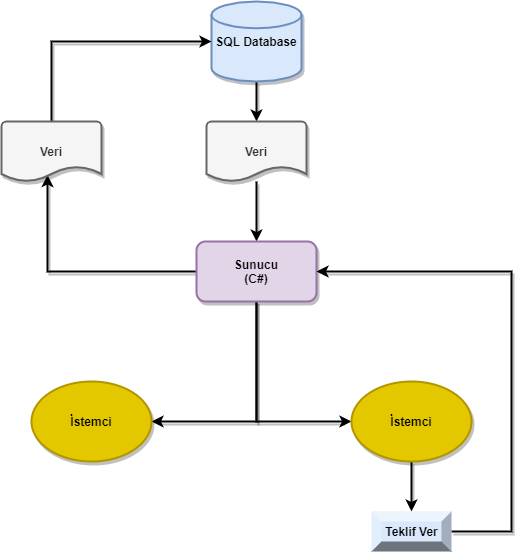
AS(Auction System) ile gerçek hayatta sanki bir Açık Artırmadaymışsınız gibi neredeyse tüm müzayede kurallarına uyarak ürünlere teklif vermenize olanak sağlamaktadır. Böylece en yüksek teklifi veren ürünün yeni sahibi olmaktadır.



AS Çalışma Şekli

# 2.2 Projenin Plan Kapsamı

Müzayede de teklif verirken sanki oradaymışsınız gibi anlık olarak veri alış-verişi sağlanmaktadır. Ancak tüm kurallar mevcut olmadığı gibi bazı kurallar da dijital ortama adapte olması için değiştirilmiştir.



Projenin Genel Yapısı

##### AYARLANMAMIŞ İŞLEV NOKTASI PUANININ (AİP) BULUNMASI

AİP = 1 \* W(1) + 2 \* W(2) + 3 \* W(3) + 4 \* W(4) + 5 \* W(5)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ölçüm Parametresi | Ağırlık Faktörü (W) | | | | | | | Parametre Değeri | |
| **Sayı** | **Yalın** | | **Ortalama** | | **Karmaşık** | |
| (1)Kullanıcı Girdi sayısı | - | | 3 | | 4 | | 6 | |  |
| (2)Kullanıcı Çıktı sayısı | 8 | | 4 | | 5 | | 7 | | 40 |
| (3)Kullanıcı Sorgu Sayısı | - | | 3 | | 4 | | 6 | |  |
| (4)Kütük Sayısı | 4 | | 7 | | 10 | | 15 | | 28 |
| (5)Dışsal Arayüz Sayısı | - | | 5 | | 7 | | 10 | |  |
| TOPLAM |  | |  | |  | |  | | **68** |

##### TEKNİK KARMAŞIKLIK FAKTÖRÜNÜN (TKF) BULUNMASI

|  |  |  |
| --- | --- | --- |
| 1. | Sistem güvenilir yedekleme ve kurtarma gerektiriyor mu? | 0 |
| 2. | Veri iletişimi gerekiyor mu? | 0 |
| 3. | Dağıtık fonksiyon var mı? | 0 |
| 4. | Performans kritik mi? | 5 |
| 5. | Sistem çok kullanılan bir işletim ortamında mı çalışacak? | 4 |
| 6. | Sistem on-line veri girişi gerektiriyor mu? | 0 |
| 7. | On-line veri giriş işlemlerinin birden fazla ekran ya da işlem üzerinden olmasını mı gerektiriyor? | 0 |
| 8. | Ana dosyalar on-line mı güncelleniyor? | 0 |
| 9. | Girdiler, çıktılar, dosyalar ve sorgular karmaşık mı? | 5 |
| 10. | Kod yeniden kullanabilir olarak mı tasarlanmış? | 5 |
| 11. | İç süreç karmaşık mı? | 4 |
| 12. | Dönüşüm ve kurulum tasarım içerisinde mi? | 5 |
| 13. | Uygulama değişik kuruluşlarda birden fazla kurulum gerektirecek şekilde mi tasarlanmış? | 0 |
| 14. | Uygulama kullanıcı tarafından kolaylıkla kullanmayı ve değiştirmek üzere mi tasarlanmış? | 5 |
|  | **0:** Hiçbir Etkisi Yok **1:** Çok Az etkisi var **2**: Etkisi Var **3**: Ortalama Etkisi Var  **4**: Önemli Etkisi Var **5**: Mutlaka Olmalı, Kaçınılamaz | 33 |

##### İŞLEV NOKTASI PUANININ (UFP) BULUNMASI

İN = AİN x (0.65 x 0.01 x TKF)

İN= 68\*(0,65\*0,01\*33)

İN = **14,586**

##### SATIR SAYISININ BULUNMASI

Proje kodlamasında kullanılacak olan dil Pascal (Delphi) olup, satır sayısı kestirim katsayısı 90 dır.

Satır Sayısı = İN \* 30 = 14,586 \* 90 **= 1312**

##### ETKİN MALİYET MODELİ: COCOMO

Projemiz, küçük ölçekli bir proje olduğundan Temel Model baz alınacaktır. Modelin uygulamasında kullanılan formüller aşağıdaki gibidir:

|  |  |  |
| --- | --- | --- |
| Ayrık Projeler | Yarı Gömülü Projeler | Gömülü Projeler |
| İş Gücü K = 2.4\*S1,05  Zaman T = 2.5\*K0,38 | İş Gücü K = 3,0\*S1,12  Zaman T = 2.5\*K0,35 | İş Gücü K = 3,6\*S1,20  Zaman T = 2.5\*K0,32 |
| S: 1000 türünden satır sayısı | | |

“Ayrık Proje” sınıfına giren projemiz için kullanacağımız COCOMO Formülü şu şekilde ortaya çıkmaktadır:

İş Gücü K = 2.4 \* S1,05 = 2.4 \* 0.51,05 = 1.16 (Aylık Kişi Başı İş Gücü)

Zaman T = 2.5 \* K0,38 = 2.5 \* 1.160.38= 2.9 = 3 Ay

# 2.3 Proje Ekip Yapısı

3Proje Ekibi ve Görevleri

# 2.4 Proje Zaman-İş Planı

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ZAMAN  İŞ | Proje Başlangıç Tarihi:06/10/2020 | | | |  |  | | | | | | | | HAFTALAR | | | | | | |
| 1 | 2 | 3 | | | | 4 | 5 | | 6 | 7 | 8 | | | 9 | 10 | | 11 | 12 |
| İhtiyaç / Gereksinim Yönetimi, Analizi |  |  |  | | | |  |  | |  |  |  | | |  |  | |  |  |
| İhtiyaç / Gereksinim Geliştirme |  |  |  | | | |  |  | |  |  |  | | |  |  | |  |  |
| Sistem Çözümleme |  |  |  | | | |  |  | |  |  |  | | |  |  | |  |  |
| Kullanıcı Arayüz Tasarımı |  |  |  | | | |  |  | |  |  |  | | |  |  | |  |  |
| Gerçekleştirim |  |  |  | | | |  |  | |  |  |  | | |  |  | |  |  |
| Sistem Doğrulanması |  |  |  | | | |  |  | |  |  |  | | |  |  | |  |  |
| Organizasyonel Eğitimler |  |  |  | | | |  |  | |  |  |  | | |  |  | |  |  |
|  | Abdullah AYGÜN | | | Abdullah AYGÜN | | | | | Abdullah AYGÜN | | | | Abdullah AYGÜN | | | | Abdullah AYGÜN | | | |

Tablo Proje İş-Zaman Çizelgesi

# 2.5 Kullanılan Özel Geliştirme Araçları ve Ortamları

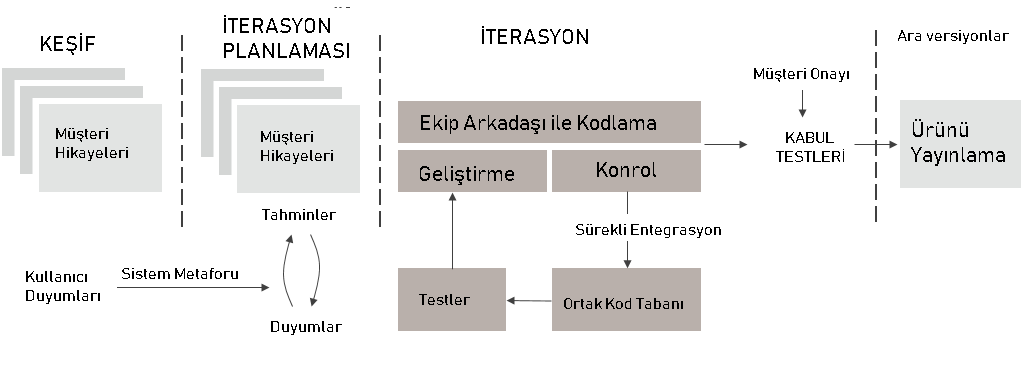
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | ÇÖZÜMLEME  TASARIM | PROGRAMLAMA | SINAMA | DESTEK |
| ORTAMLAR | Windows 10 | Visual Studio Pro 2019 | Visual Studio Pro 2019 | Google Chrome |
| ARAÇLAR | Draw.io  Microsoft Visio 2016 | ASP, C#, JQuery, CSS, HTML | Microsoft Edge | Google Chrome |

# 2.7 Proje Standartları, Yöntem ve Metodolojiler

Projemizin, 12 hafta gibi kısa sürede tamamlanması öngörülmüştür. AS’nin robust yapısı ve yazılacak olan modüllerin konsept olarak ilişkili olsalar bile kodlama açısından her birinin bağımsız küçük iş parçacıkları olması ve bunların küçük iterasyonlarla kodlanıp, üzerinde paydaşların görüş alış-verişi yapacağı toplantılarla oluşturulacak olması, doğal olarak Agile Metodolojisi’nin yaygın olarak kullanılacağına işaret etmektedir.

Bir Agile uygulaması olarak SCRUM, her ne kadar popülaritesi olan bir metodoloji olsa da sertifikalı bir SCRUM Master’ın varlığını gerektirdiğinden projemizde uygulanamayacaktır.

Extreme Programming temelinde bulunan Basitlik, Cesaret, Geri Dönüş, İletişim kullanacağımız Agile metodolojisi için projemizle uygunluk arz etmektedir. Özellikle bu 4 temel maddenin en önemli maddeleri olan Geri Dönüş ve Ekibin birbirleri ile iletişiminin sağlıklı tutulması projemizin ekip yapısına uymaktadır. Geri Dönüş yani İstekler, proje yapım aşamasındayken bile değişebilir ve bu değişim diğer yazılım metodolojilerine göre daha hızlı uygulanabilir.



4 Extreme Programlama Metodolojisi Akışı

# 2.8 Kalite Sağlama Planı

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| İş Tanımı | Zaman | | | | | | | | | | | | | İlgili Kişi | | Kalite Sağlama Unsuru | | Kullanılacak Kaynak | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |  | |  | |  | |
| Planlama |  |  |  |  |  |  |  |  |  |  |  |  | Abdullah  AYGÜN | | Projenin tüm gereksinimlerini karşılayacak şekilde yürütülmesi için gereken kalite standartlarının değerlendirilmesi | | IEEE 730 Standard for  Software Quality Assurance Plans,  Üniversitemiz Bölüm Yetkilileri | |
| Ürün gözden geçirme |  |  | | | | | | | | |  |  | Abdullah  AYGÜN | | Etkinlik  Verimlilik  Müşteri Memnuniyeti | | JSTD-  016 Standart for Information Technology Software Life  Cycle Processes Software Development | |
|  | | | | | | | | |
| Süreç/proje takip |  |  | | | | | | | | |  |  | Abdullah AYGÜN | | Tanımlama soruları  Uygulama soruları  İzleme  soruları | | ISO/IEC 15939 standardı | |
|  | | | | | | | | |
| Konfigürasyon tetkikleri |  |  |  |  |  |  |  |  |  |  |  |  | Abdullah AYGÜN | | İşlevsel Özellikler  Performans  Özellikleri  Dokümantasyon  Uyumu | | SCORM | |

# 2.9 Konfigürasyon Yönetim Planı

|  |  |  |  |
| --- | --- | --- | --- |
| Konfigüre Gerektiren Durum | İlgili Kişi | Süre | Sonuç |
| Client-Side | Abdullah AYGÜN | 1 gün | Veri işleme UI komponentlerinin kurulum ve konfigürasyonu yapılacaktır. |
| Server-Side | Abdullah AYGÜN | 1 gün | AS’nin import edeceği dosyaya ait konfigürasyon yapılacaktır. |

# 2.10 Kaynak Yönetim Planı

|  |  |  |
| --- | --- | --- |
| Kaynak Türü | Proje Kapsamında Elverişlilik | Kullanım Amacı |
| İnsan | 1 Öğrenci – Programcı | Proje Ekibi |
| Malzeme |  |  |
| Araç, gereç, donanım | 1. laptop bilgisayar | Projenin kodlanması |
| İşletme Sermayesi | - | - |

Sistemin gerçekleştirilmesi için gereken donanım mevcut olup, herhangi bir maliyete katlanmadan projenin gerçekleştirilmesini mümkün kılacaktır.

Projede herhangi bir ücretli yazılım bulunmamaktadır. Her şey C#, CSS, HTML, JQuery ile ASP yapısında yapılmıştır.

Yazılım Mühendisliği bölümü 1 öğrencinin ders projesi olarak gerçekleştireceği yazılım için, herhangi bir ücret ödemesi olmayacaktır.

# 2.11 Eğitim Planı

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Eğitim Türü | Eğitim Konusu | Eğitim Verecek Kişi | Tarih/Süre | Amaç |
| OnLine | AS Kullanımı | Abdullah AYGÜN | 26/05/2021 | Kullanıcı arayüzünün tanıtılması |

# 2.12 Test Planı

Projemizde yazılım test faaliyetleri süresince birim, entegrasyon ve sistem testleri gerçekleştirilecektir. Aşağıdaki yazılım modülleri bu plan kapsamında belirtilen testlere tabi tutulacaktır.

* Teklif verme AS kabiliyetlerine uygun çalışıyor mu?

Görsellik, istenilen düzeyde mi?

İstenilen şekilde özelleştirme yapılabiliniyor mu?

* *Görünüm menüsü*

Görsellik, istenilen düzeyde mi?

İstenilen şekilde özelleştirme yapılabiliniyor mu?

1. Derece Hatalar: Sistemin genel çalışmasını direkt etkileyen hatalar.
2. Derece Hatalar: Sistemin genel çalışmasını direkt etkilemeyen fakat işlevsel olarak bazı kısımların çalışmasını engelleyen hatalar.
3. Derece Hatalar: Sistemin çalışmasını engellemeyen, görsel yönden ortaya çıkan hatalardır.

Proje kapsamında test seviyelerine göre görev dağılımı ve sorumluluklar şöyledir:

|  |  |
| --- | --- |
| Test Seviyeleri | Görev ve Sorumluluklar |
| Birim Testleri | Geliştirme Ekibi |
| Entegrasyon Testleri | Test Ekibi – Abdullah AYGÜN |
| Sistem Testleri | Test Ekibi – Abdullah AYGÜN |
| Kabul Testleri | Müşteri ve Test Ekibi - Abdullah AYGÜN |

Projesi kapsamında test faaliyetleri yazılımcı laptop bilgisayarı üzerinde yürütülecektir. Bilgisayara ait özellikler aşağıdaki gibidir:

CPU: Intel Xeon 3.3 GHz

RAM: 8 GB DDR4 2133 MHz

HDD: 100 GB

OS: Windows 10

Yüklenecek Programlar: Microsoft Excel

|  |  |
| --- | --- |
| Test Araçları | Kullanım Amacı |
| Firebase Test Lab | Birim testleri için geliştirme ekibi tarafından |
| Winium | Sistem testleri için test ekibi tarafından test otomasyon aracı kullanılacaktır. |
| Apache JMeter | Fonksiyonel olmayan testler (Performans, Yük ve Stres) için Apache JMeter aracı kullanılacaktır. |
| TFS | Test faaliyetleri sonucunda bulunan hatalar TFS üzerinde ilgili geliştiriciye hata kaydı olarak açılacaktır. |

# 2.13 Bakım Planı

Proje ile gerçekleştirilen yazılım, aylık olarak bakımdan geçecektir. Müşterilerden gelecek olan geri bildirimler sayesinde aylık olarak yeni güncellemeler getirilerek daha iyi bir yazılım elde edilecektir.

3. SİSTEM ÇÖZÜMLEME

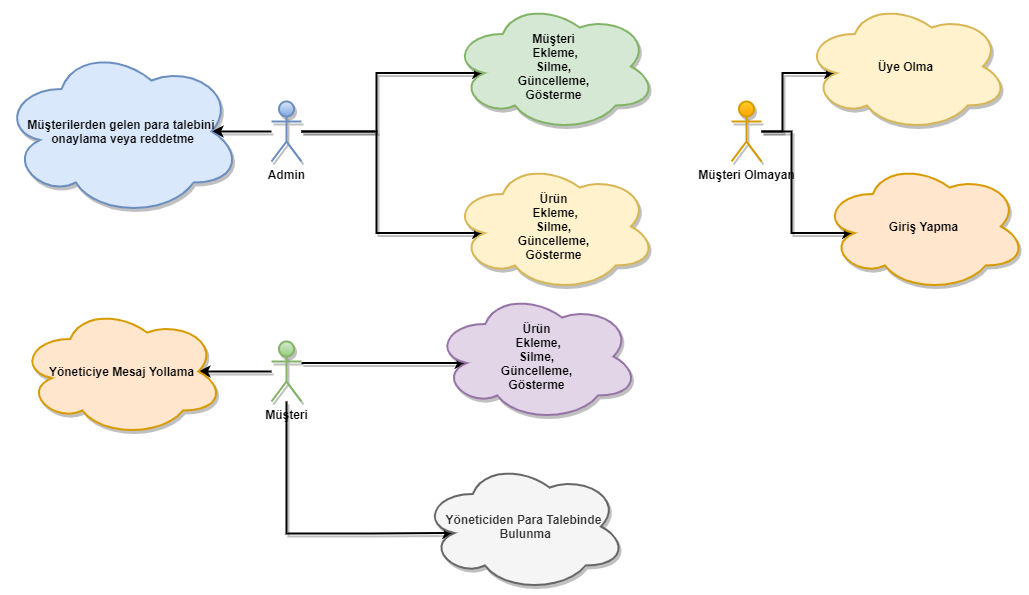
# 3.1 Mevcut Sistem İncelemesi

Proje, gerçek hayattaki açık artırma sistemindeki çoğu kuralların dijital ortama aktarılması ile oluşturulmuştur. Yüz yüze yapılan müzayede de, bizim projemizde de teklif verecek kişi oturduğu yerden teklif verebilir.

## Örgüt Yapısı

Şekil SLA Organizasyonal Yapısı

## İşlevsel Model



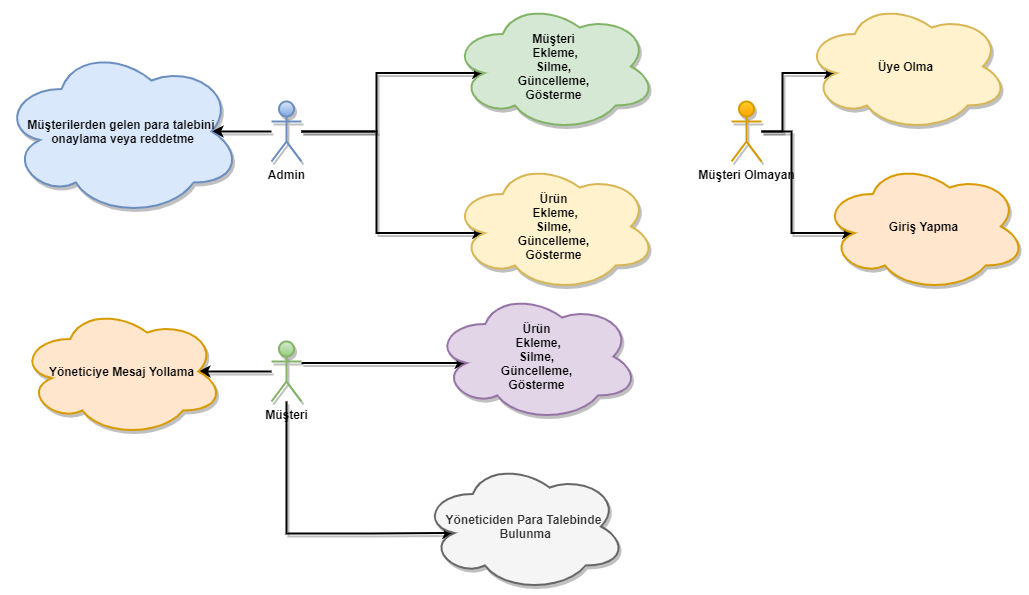
Use-Case Diyagramı

# 3.2 Gereksenen Sistemin Mantıksal Modeli

## 3.2.1 Giriş

## Günümüzde daha da popülerleşen online müzayede sistemlerinin çalışma mantıkları neredeyse hepsinin benzerdir. Kullanıcı teklif ver düğmesine bastığı anda server ‘a veri yollar. Server bu veriyi Veritabanına ekler ve Veritabanından gelen veri tekrar Server ‘a iletilir. Bunun sonucunda server diğer istemcilere veri yollar. Böylece istemciler arası haberleşme ağı kurulmuş olunur.

## 3.2.2 İşlevsel Model

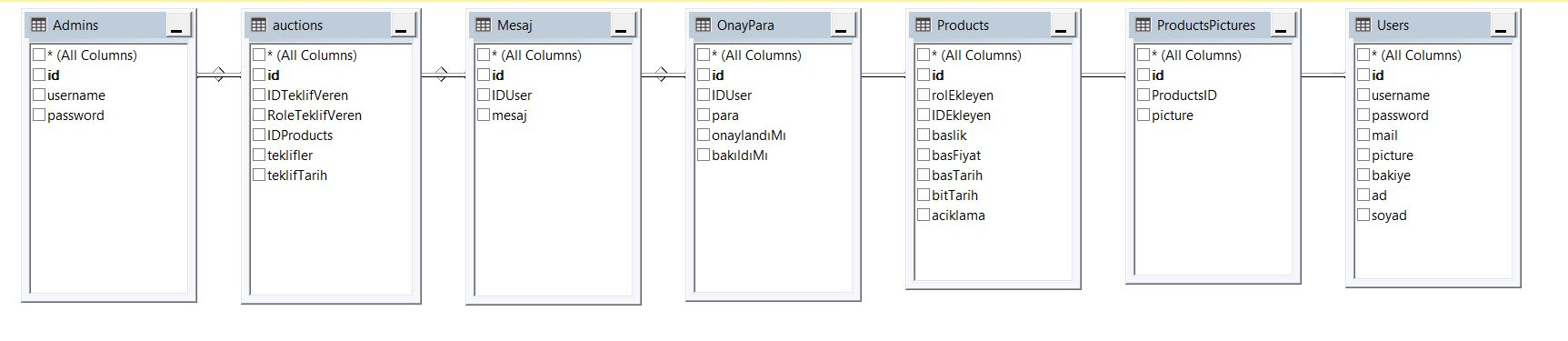


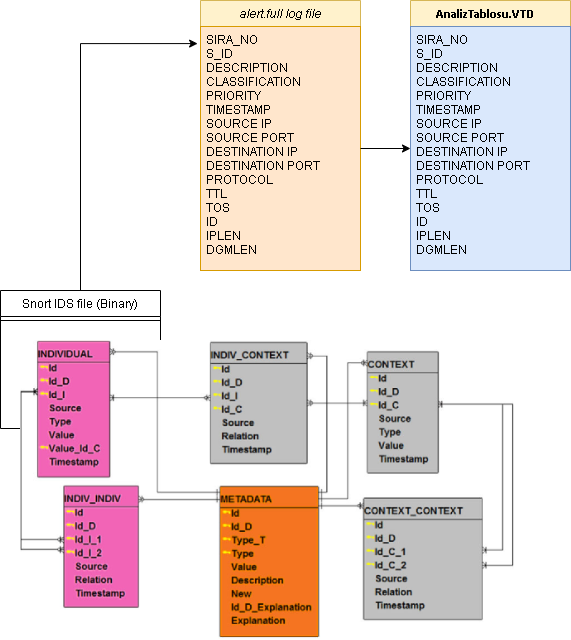
Şekil Use Case Diyagramı

## 3.2.3 Genel Bakış

Proje kapsamında çok fazla tabloya ihtiyaç yoktur. Basit bir müzayedede kullanılması geren tablolar eklenmiştir.

## 3.2.4 Bilgi Sistemleri/Nesneler



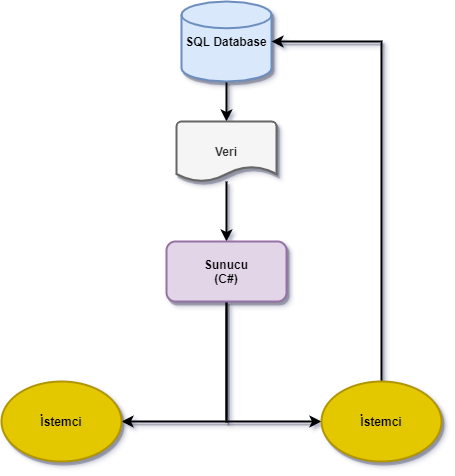


Şekil Proje Sınıf-Nesne İlişkileri

## 3.2.6 Veri Sözlüğü

Programın kullanılmasına dair özel bir veri sözlüğü bulunmamakla birlikte, ürün tipik bir veritabanı kullanıcısının rahatlıkla adapte olabileceği bir arayüze sahiptir.

## 3.2.7 İşlevlerin Sıradüzeni



Şekil İşlevlerin Sırası

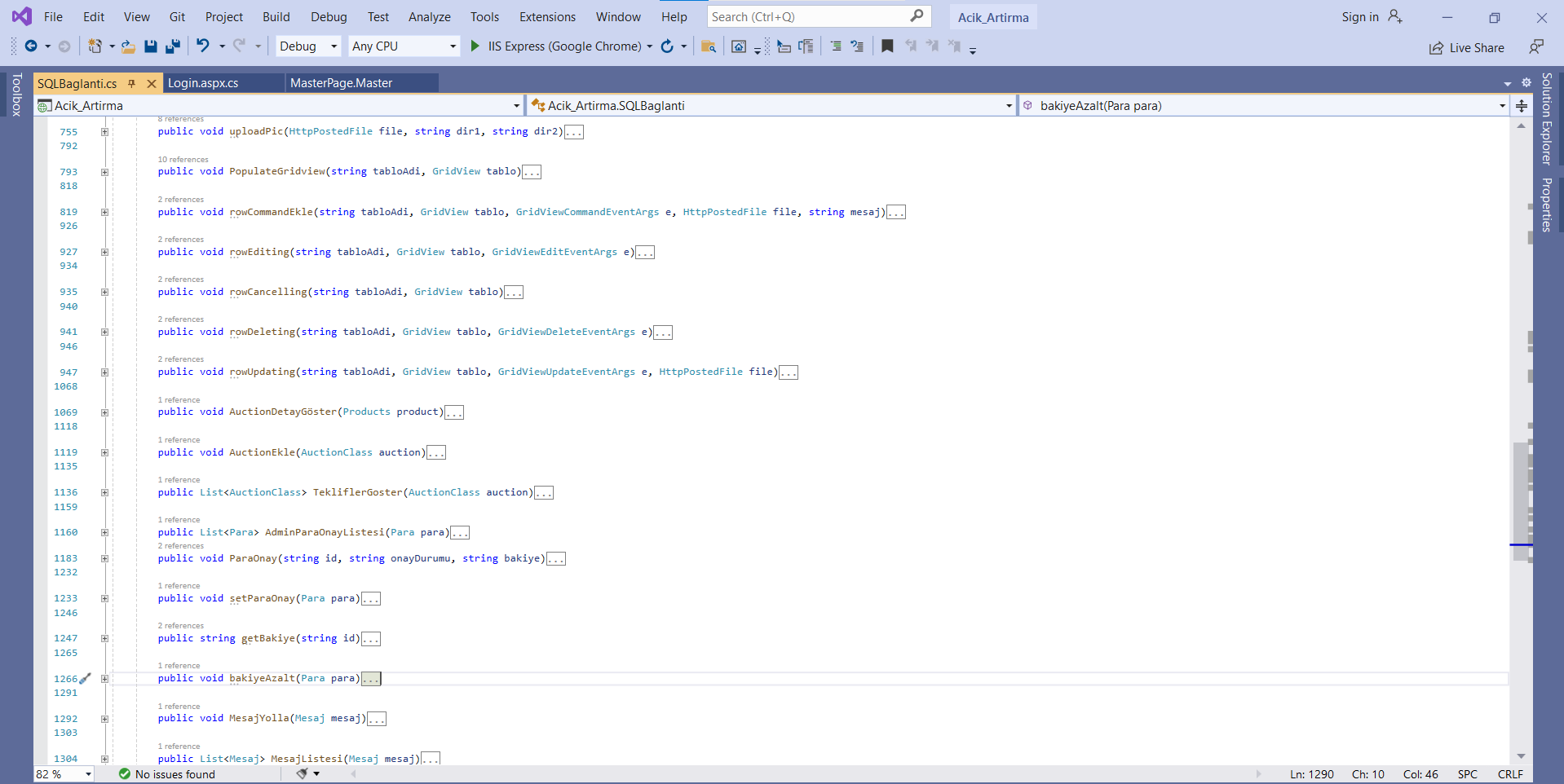
## 3.2.8 Başarım Gerekleri

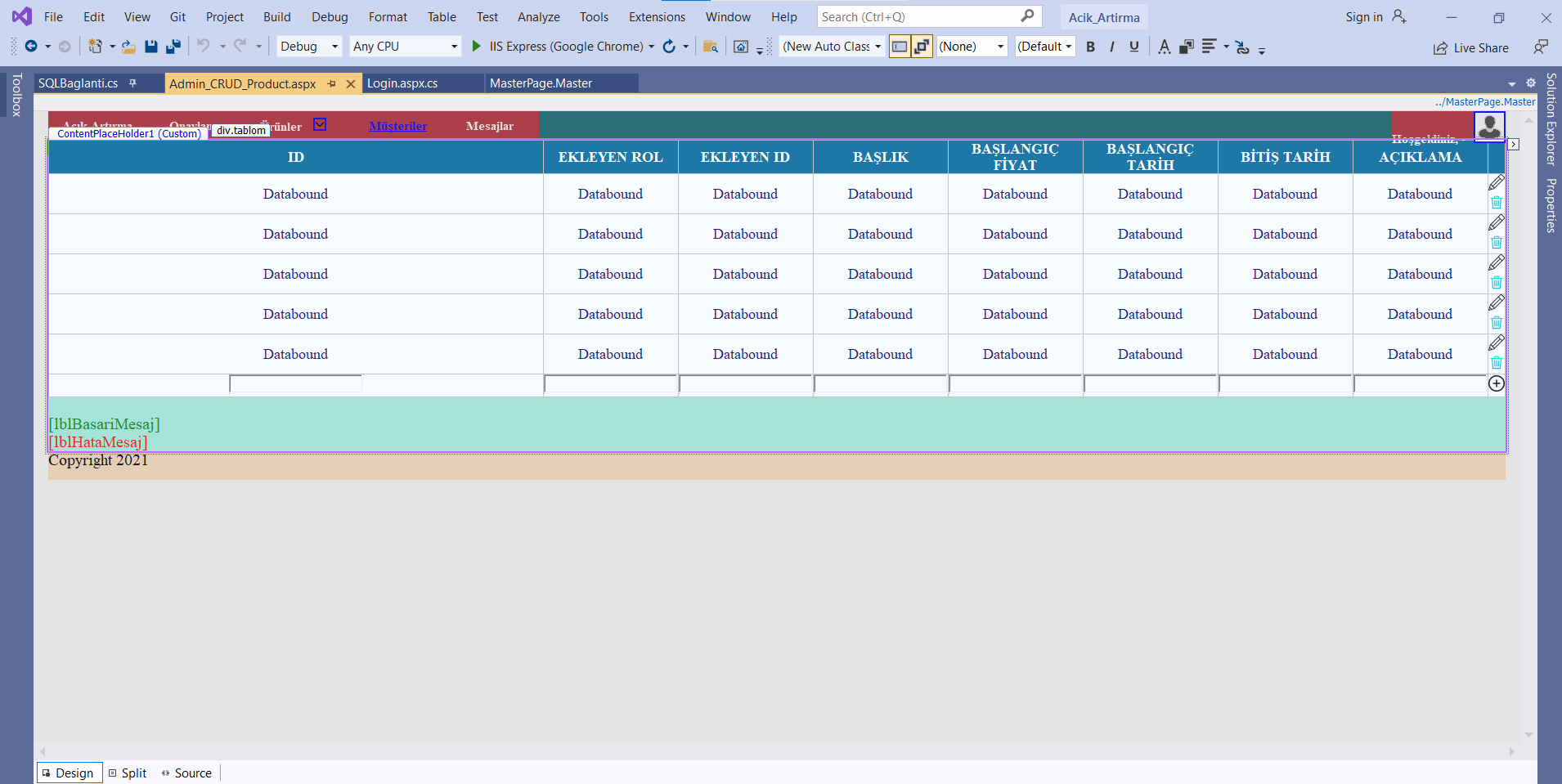
* Projenin hedeflenen 12 haftalık sürede tamamlanması,
* Gereksinim Analizinde tespit edilen hususlara ait tanımlanan tüm kabiliyetleri (sistem genelinde elverişlilik, kullanıcı dostu arayüz) sağlıyor olması

temel başarım gerekleri olarak belirlenmiştir.

# 3.3 Arayüz (Modül) Gerekleri

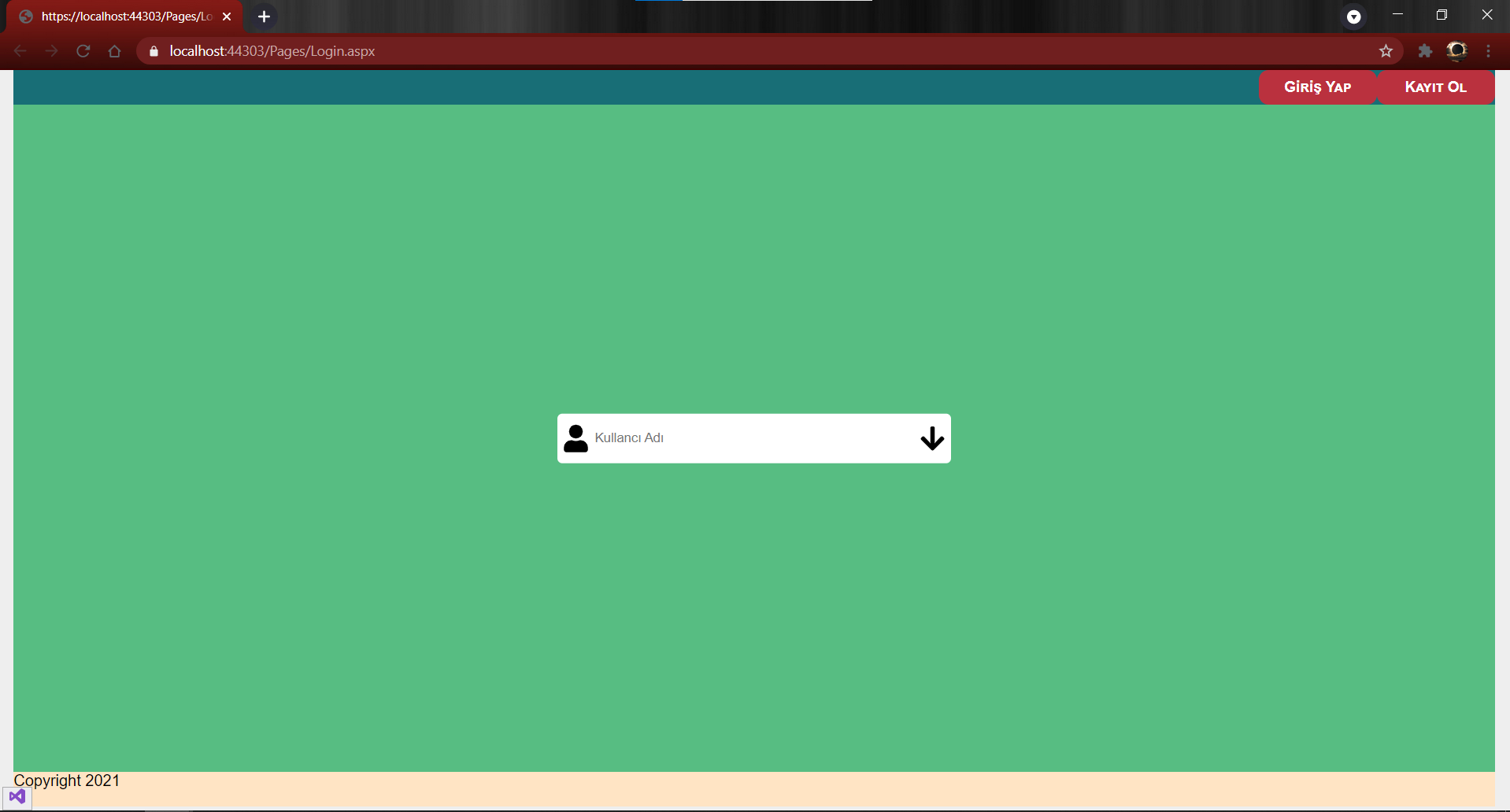
## 3.3.1 Yazılım Arayüzü

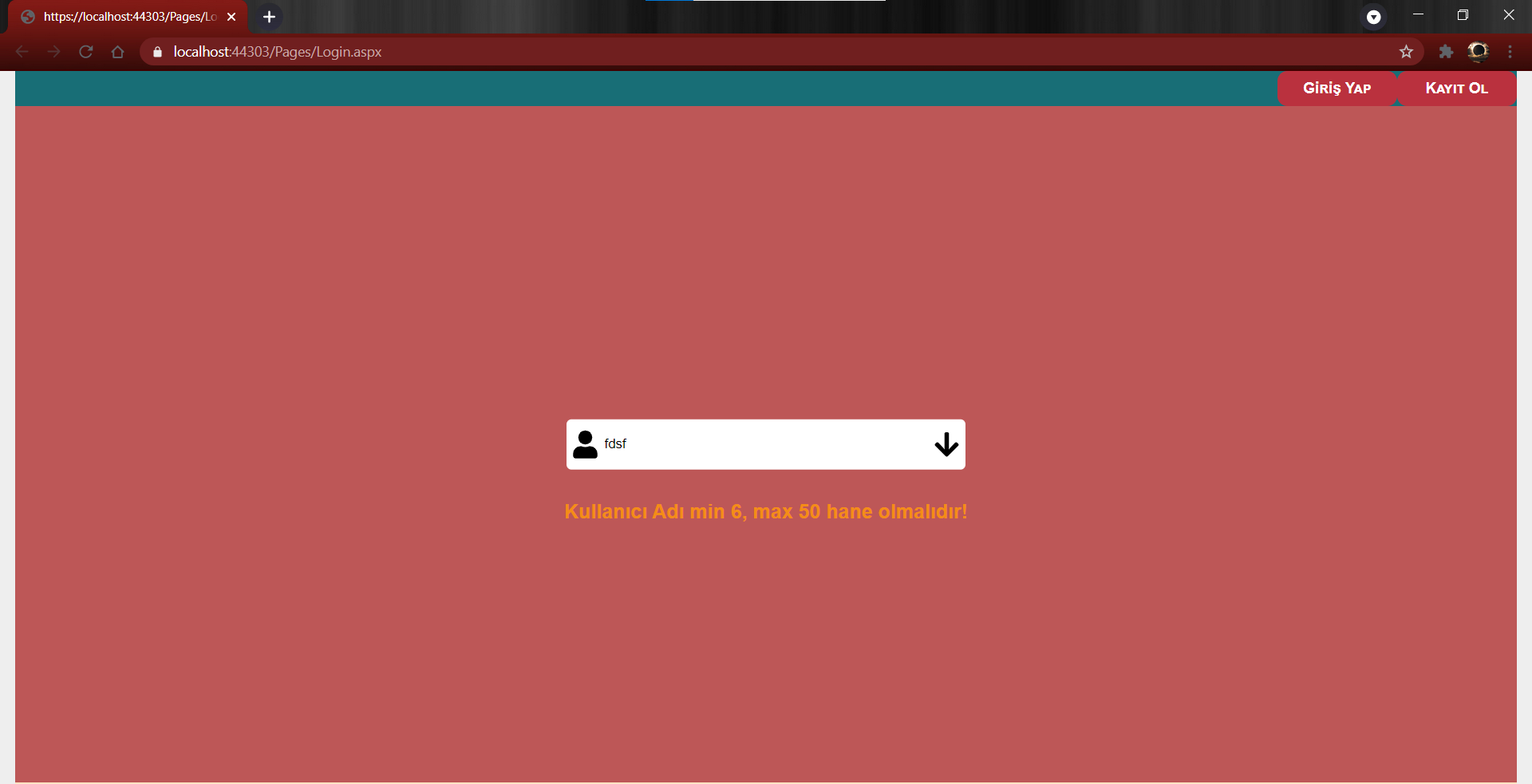


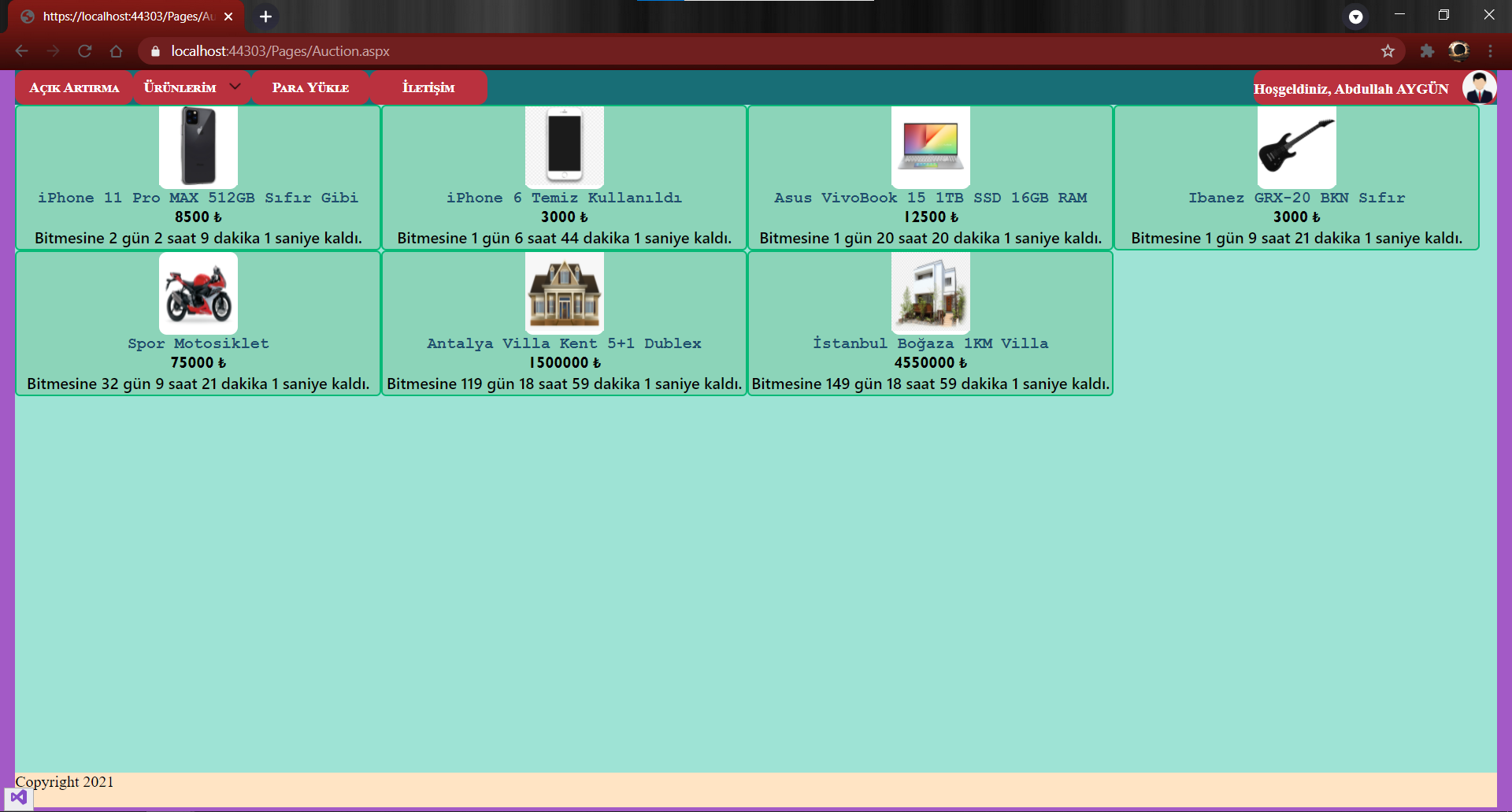


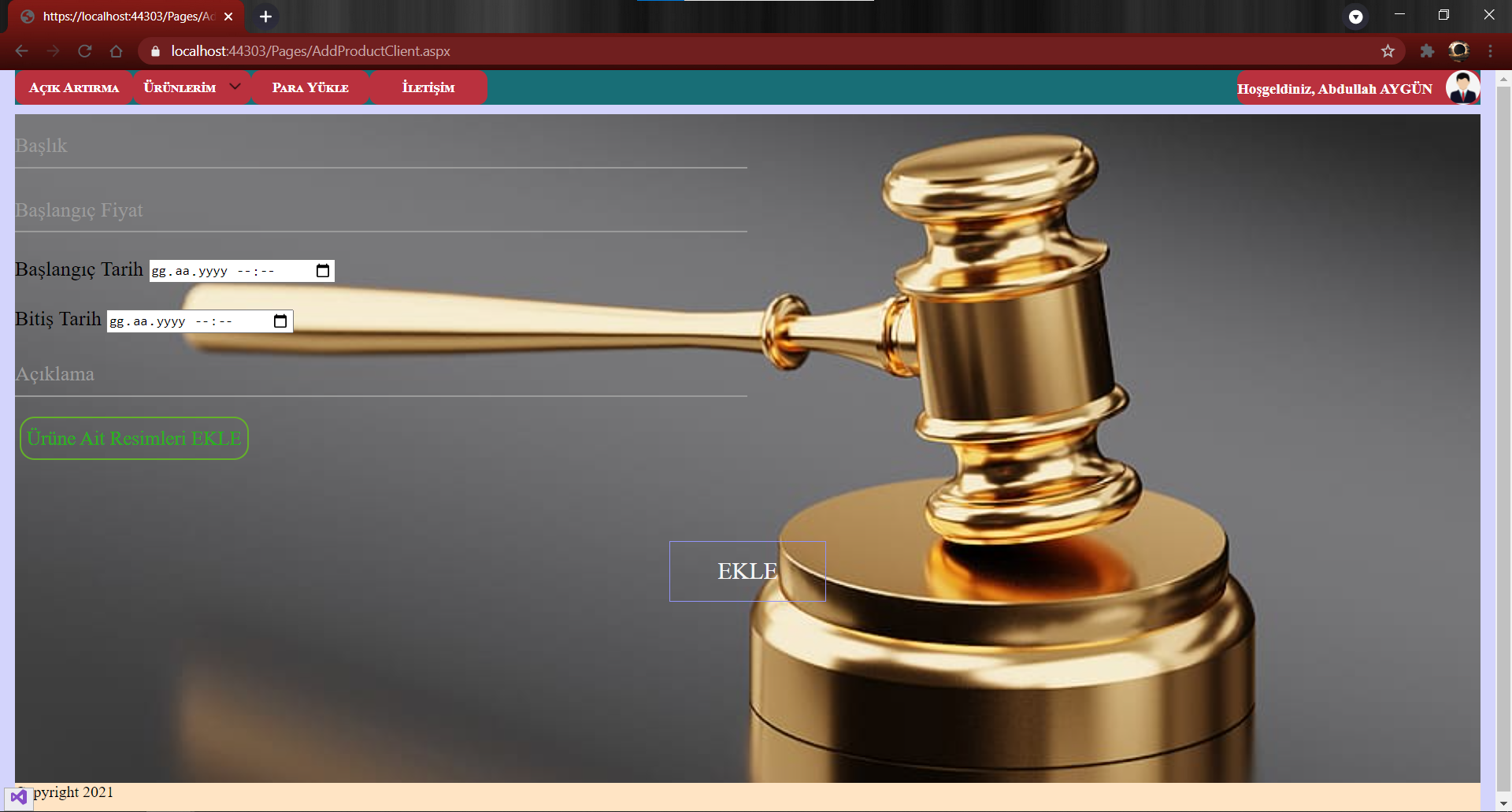
Şekil AS

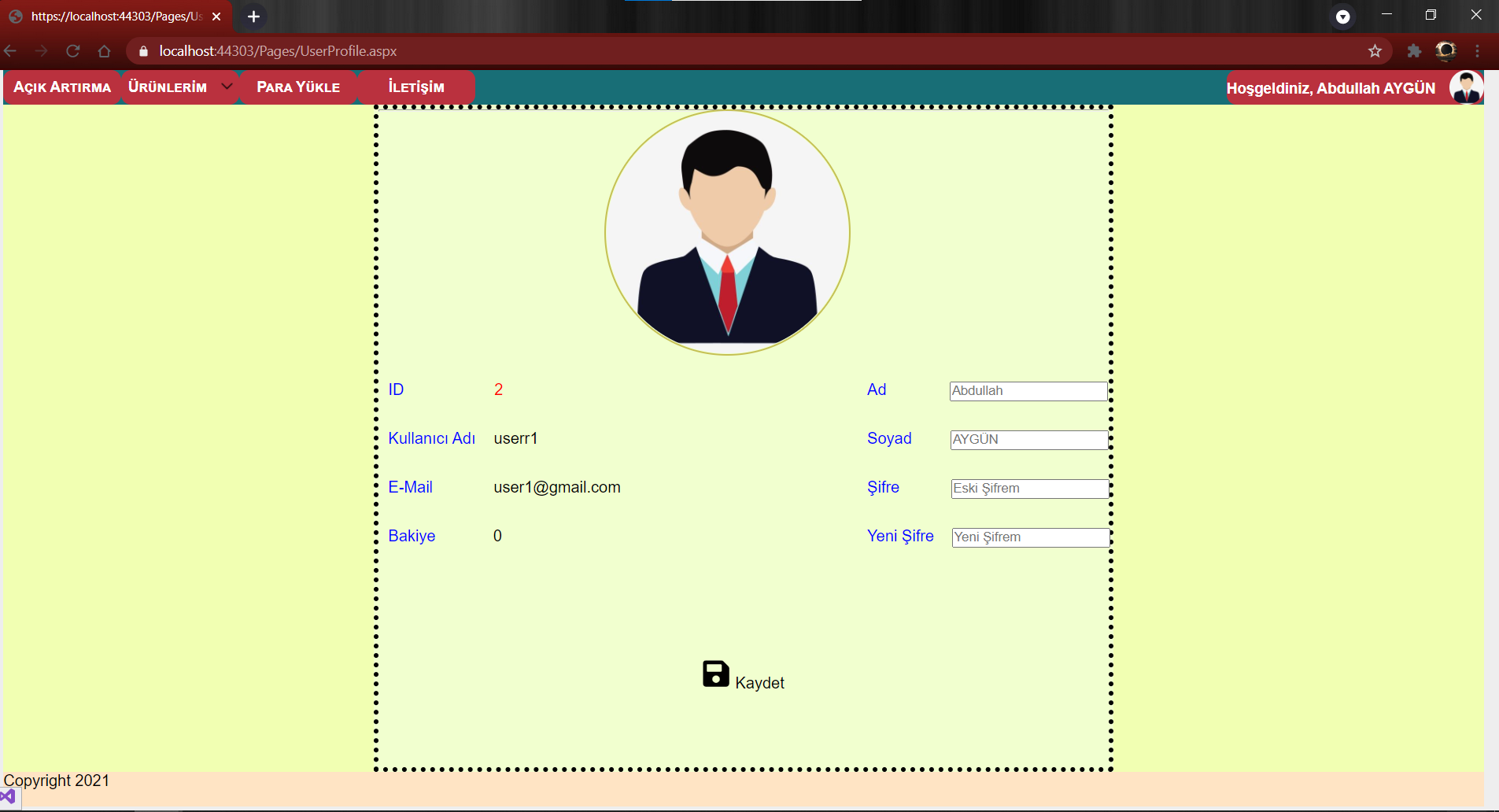
## 3.3.2 Kullanıcı Arayüzü

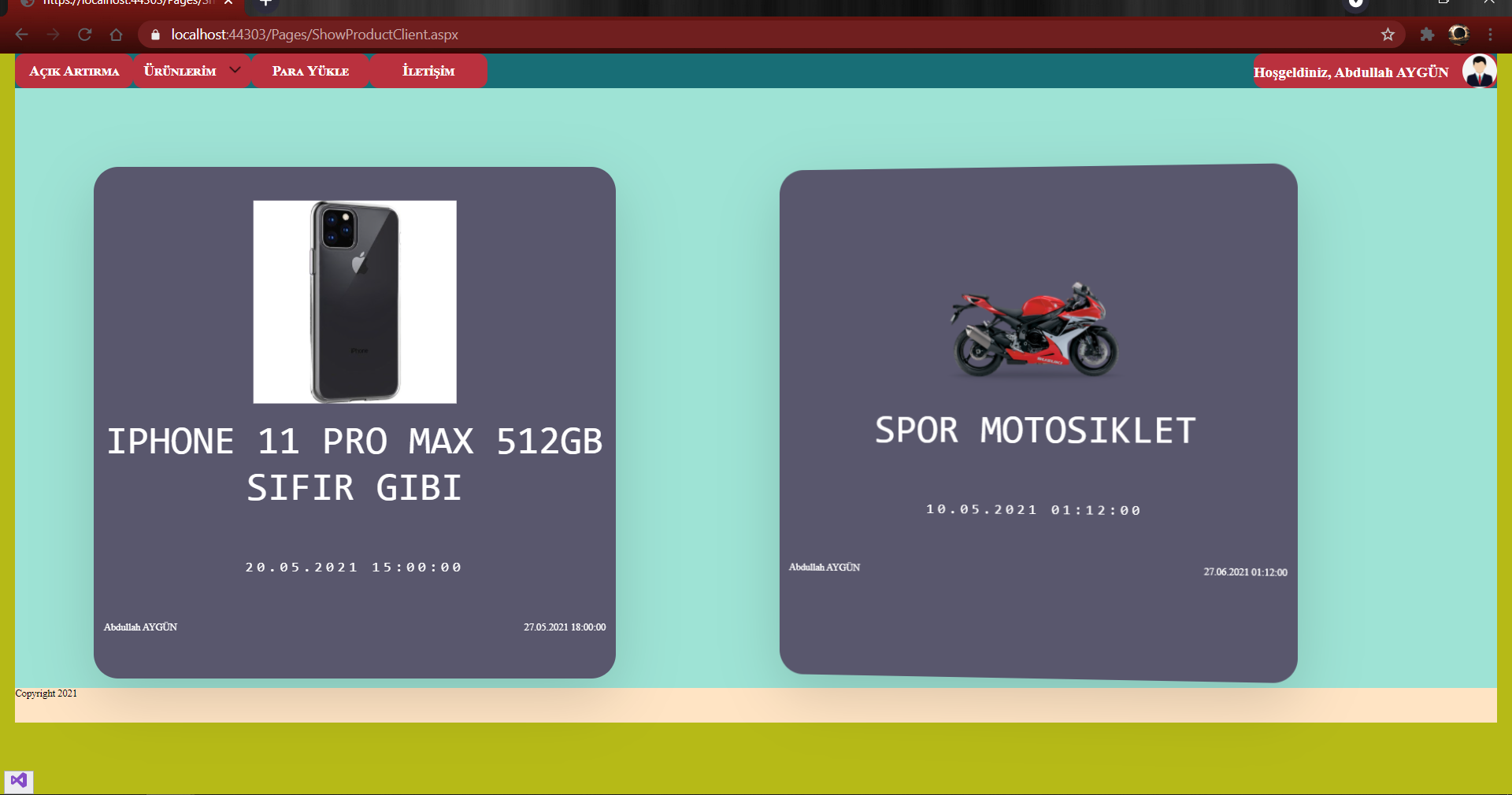


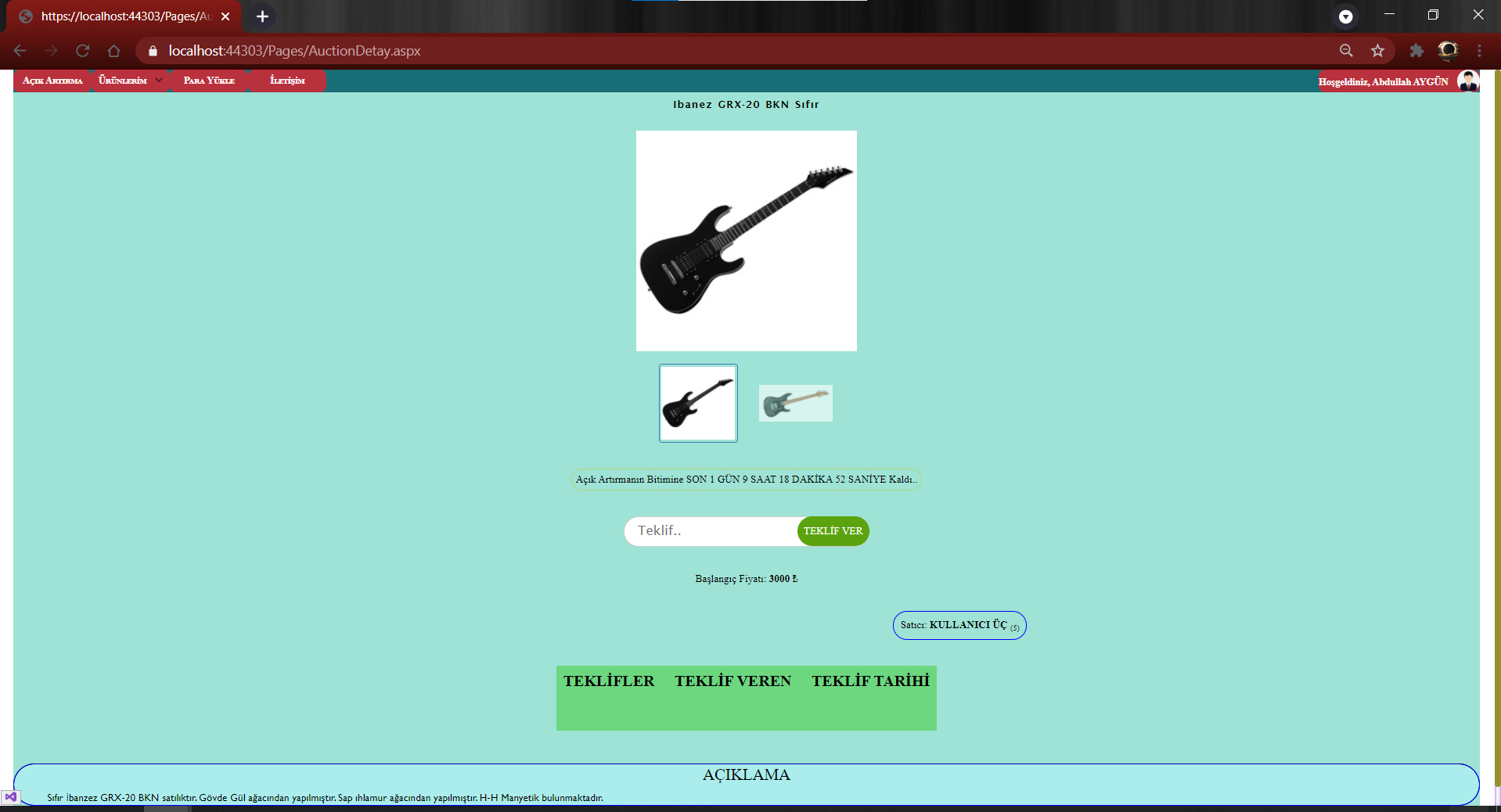












# 3.4 Belgeleme Gerekleri

## 3.4.1 Geliştirme Sürecinin Belgelenmesi

Proje Kapsam Planı, İş-Akış şemaları Proje yöneticisi tarafından yapılmıştır.

Sürecin test aşamasında karşılaşılan bir hata ile ilgili, gönderilen Rapor aşağıdadır: Rapor sonucunda yapılan düzenleme ile yazılım stabil halde çalışmasına devam etmektedir:

|  |  |  |
| --- | --- | --- |
| YAZILIM RAPORU | | |
| Yazılım Tanımı |  |  |
| Programcı Ad-Soyad | Abdulla h AYGÜN |
| Programlama Dili | ASP .NET Web Forms, JQuery |
| Bilgisayar Sistemi | Intel i5 8300H |
| İşletim Sistemi | Windows10 |
| Kaynak Kod Satırı | ~= 4500 |
| Yazılım Tasarımı | Problemin Tanımlanması | Giriş yaptıktan sonra SQLBaglanti sınıfımda Invalid Column Name “Productss” hatası alıyordum. |
| Problemin Çözümü |  |
| Sonuçların Testi | | Sütun ismini “Products” olarak değiştirdiğimde sorun çözüldü. |

## 3.4.2 Eğitim Belgeleri

Proje kapsamında, kullanıcıların 1 saatlik eğitim ile yeni sisteme adapte olmaları amaçlanır. Genel Windows ve veritabanı kontrolleri eğitimi, proje kapsamında değildir.

4. SİSTEM TASARIMI

# 4.1 Genel Tasarım Bilgileri

## 4.1.1 Genel Sistem Tanımı

Gerçek hayattaki müzayedelerin dijitalleşerek dijital ortama aktarılması hedeflenmektedir.

Proje online olarak çalışmaktadır. Gerçek zamanlı olarak teklifler verilmektedir.

## Varsayımlar ve Kısıtlamalar

Projede ek güvenlik önlemleri yazılmamıştır. Onun yerine sonuç odaklı bir proje ortaya çıkmıştır.

## Sistem Mimarisi

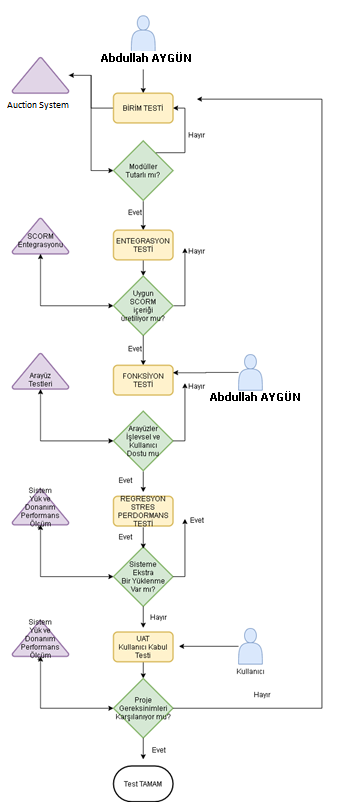
|  |
| --- |
| C:\Users\AYGUN\AppData\Local\Microsoft\Windows\INetCache\Content.Word\test (3).png  Akış Diyagramı    Şekil AS |

## 4.1.4 Dış Arabirimler

### 4.1.4.1 Kullanıcı Arabirimleri

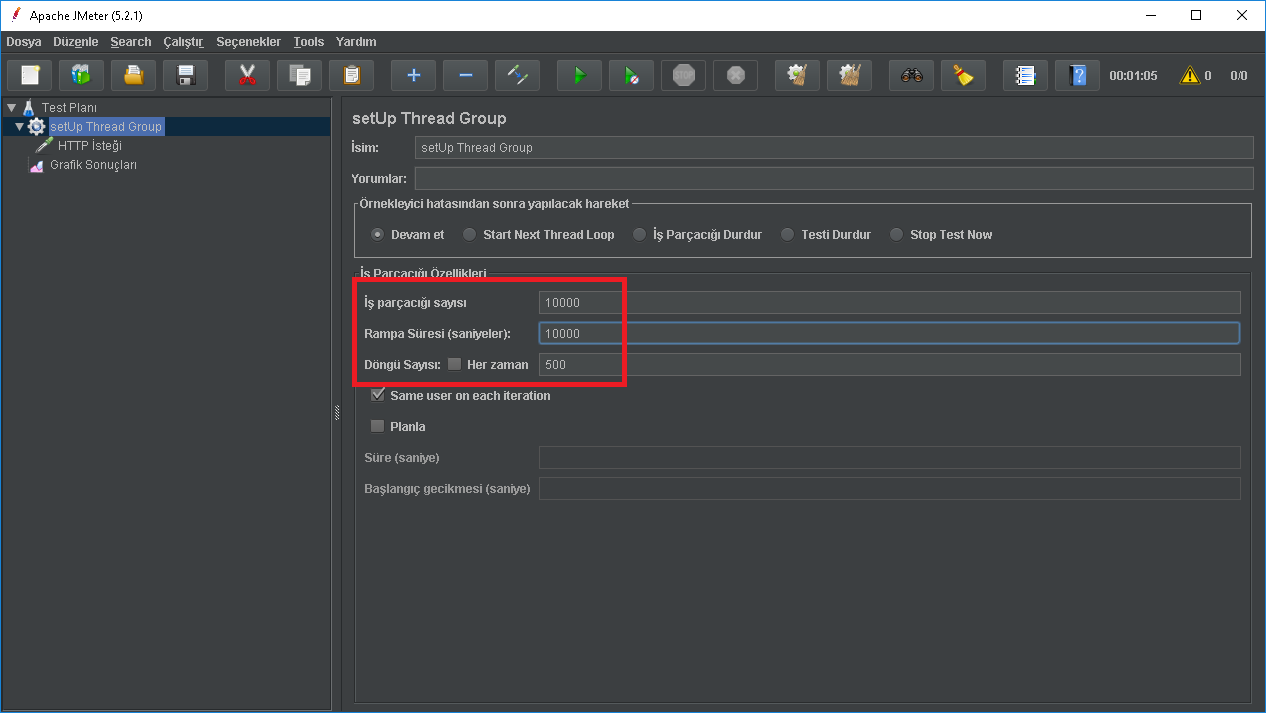
## Veri Modeli

## Testler

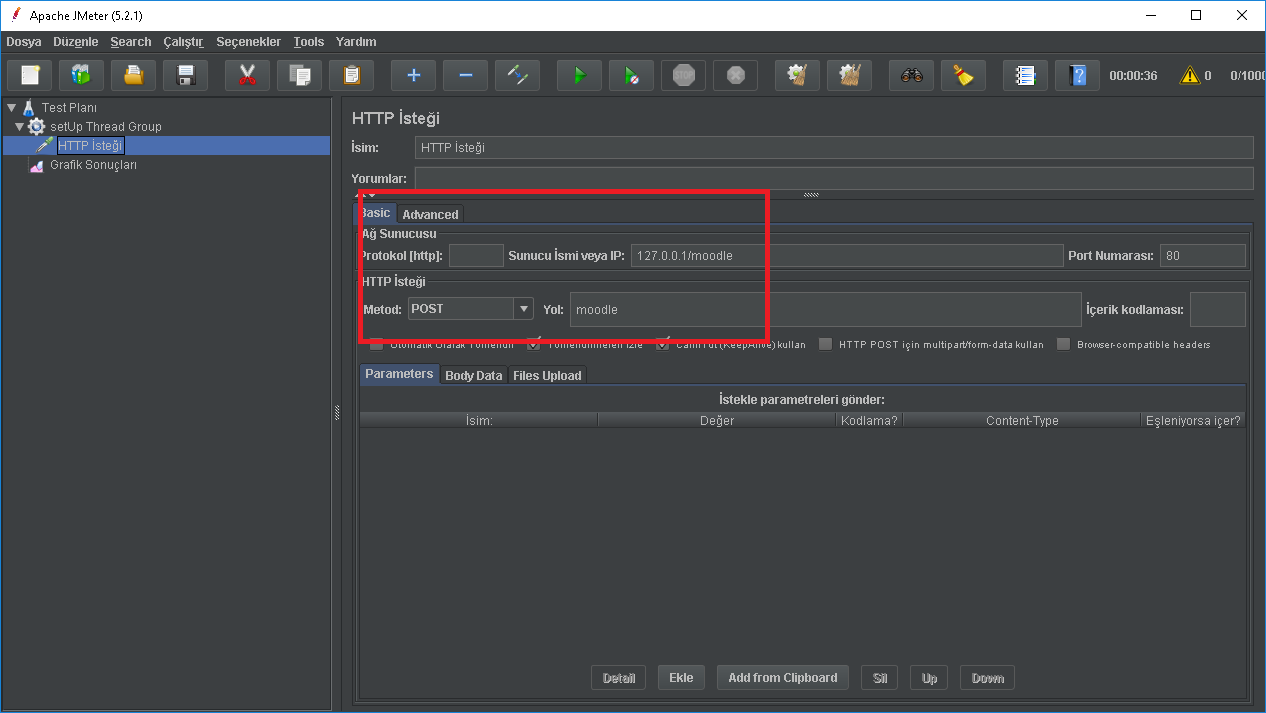


## Performans

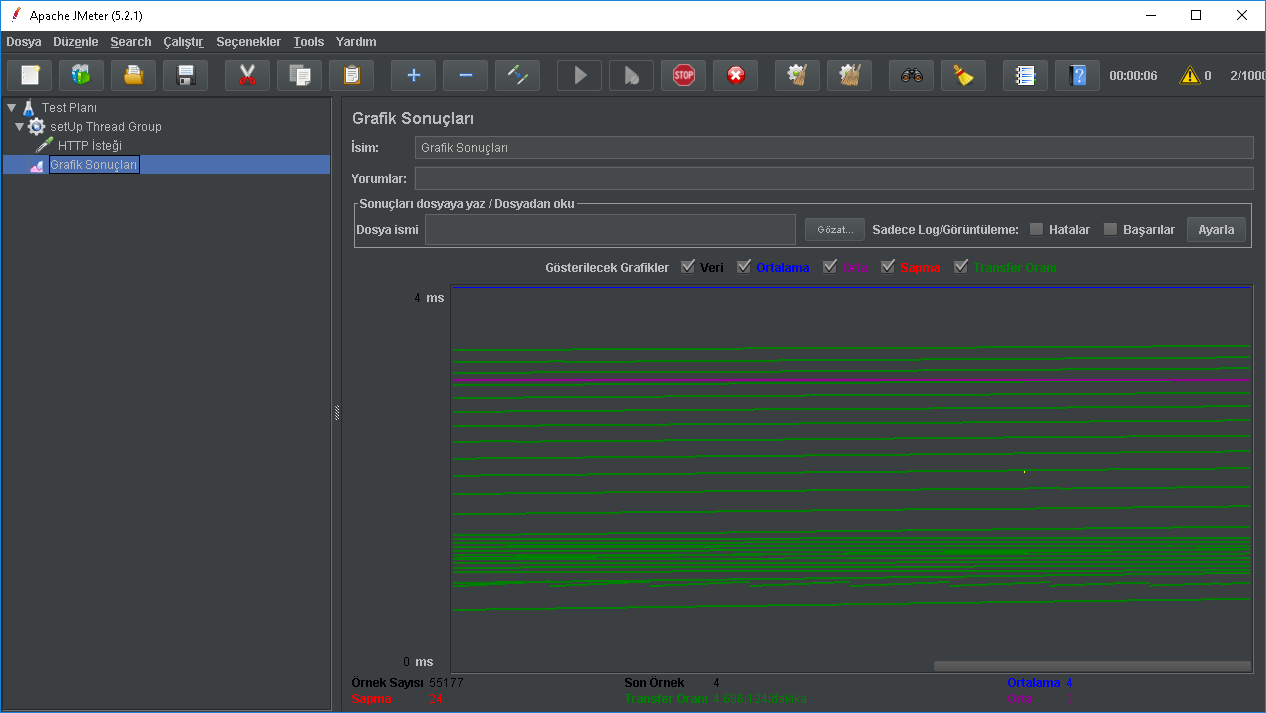
Performans test planında; özellikle stres testi için ApacheJMeter aracı kullanıldı. Test için yapılan iş tanımında “İş parçacığı sayısı” ve “Rampa süresi” = 10000, “Döngü sayısı” = 500 alındı.



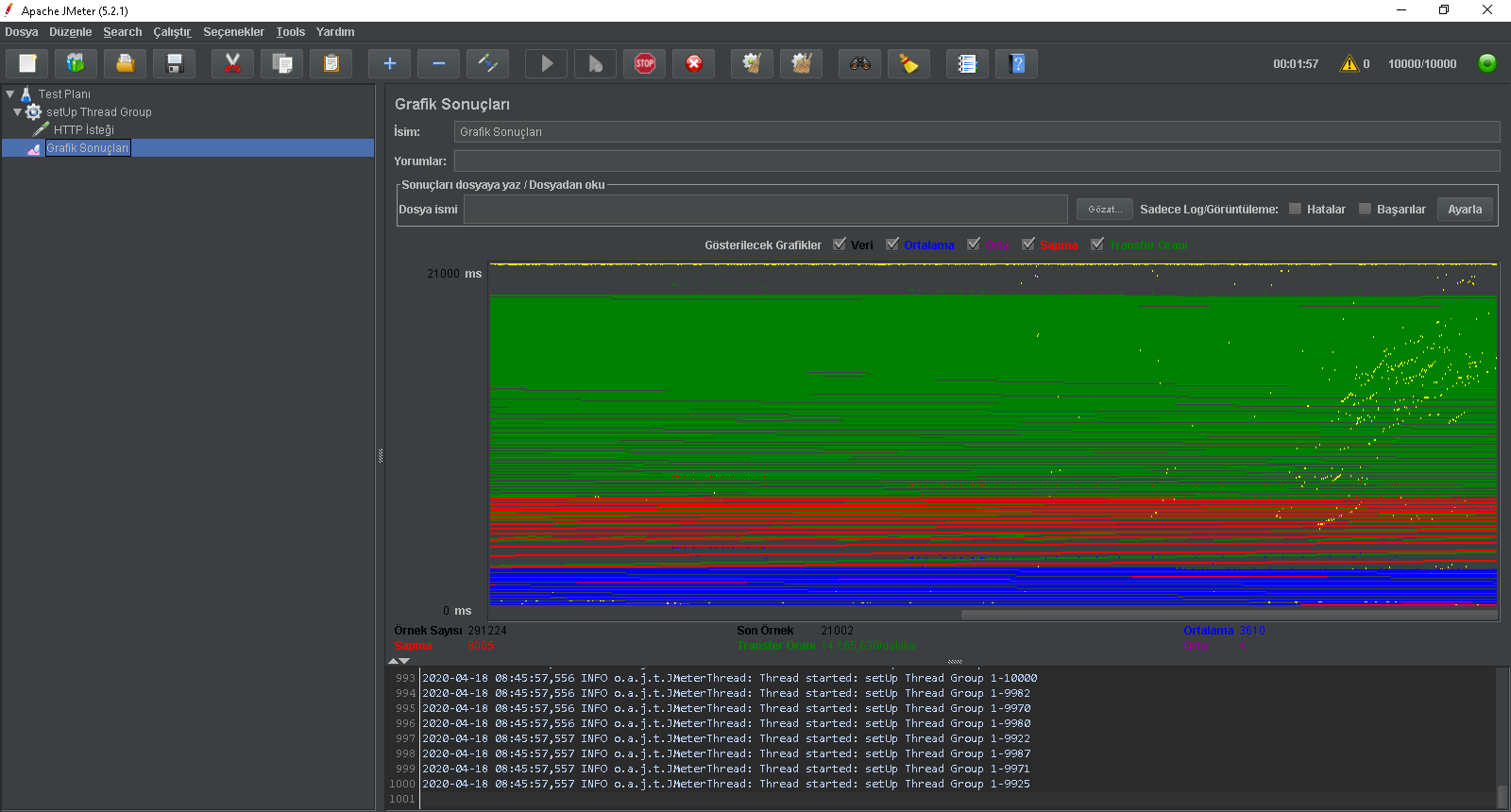
Şekil ApacheJMeter Thread Grup Ayarlaması



Şekil ApacheJMeter HTTP İsteği Oluşturulması



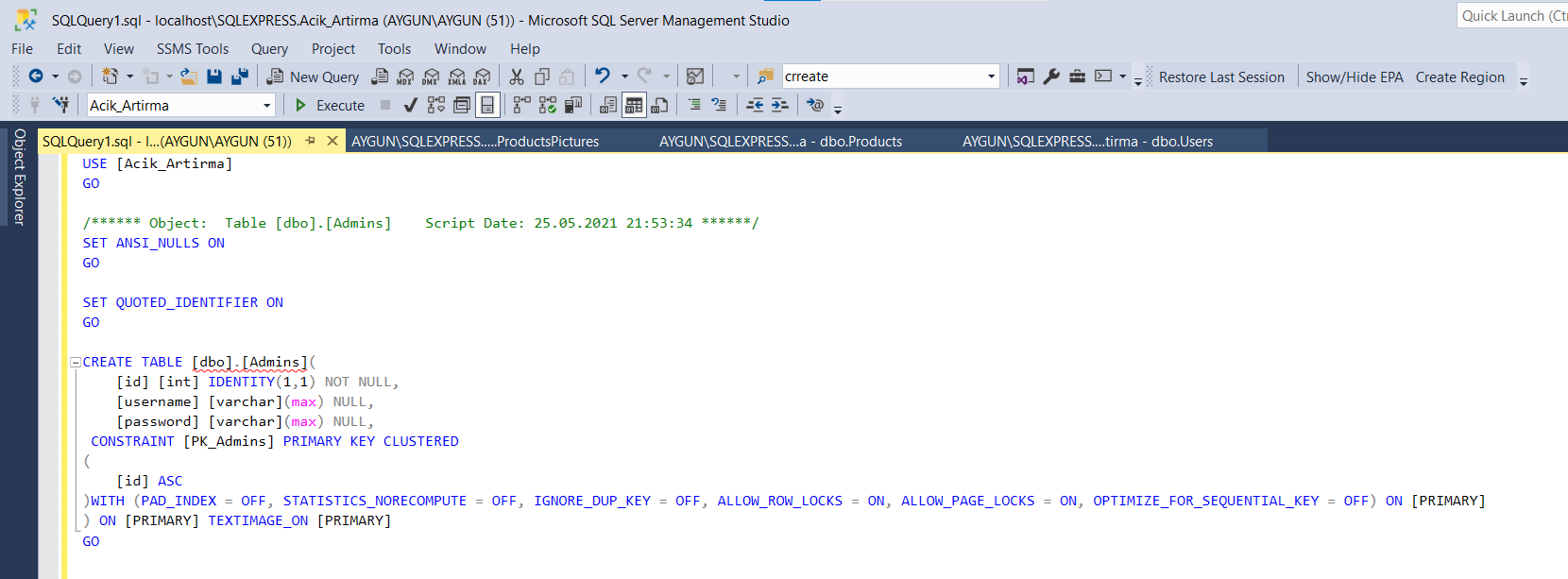
Şekil ApacheJMeter Saniyede 1 kullanıcı 10000 iş parçactğt grafik

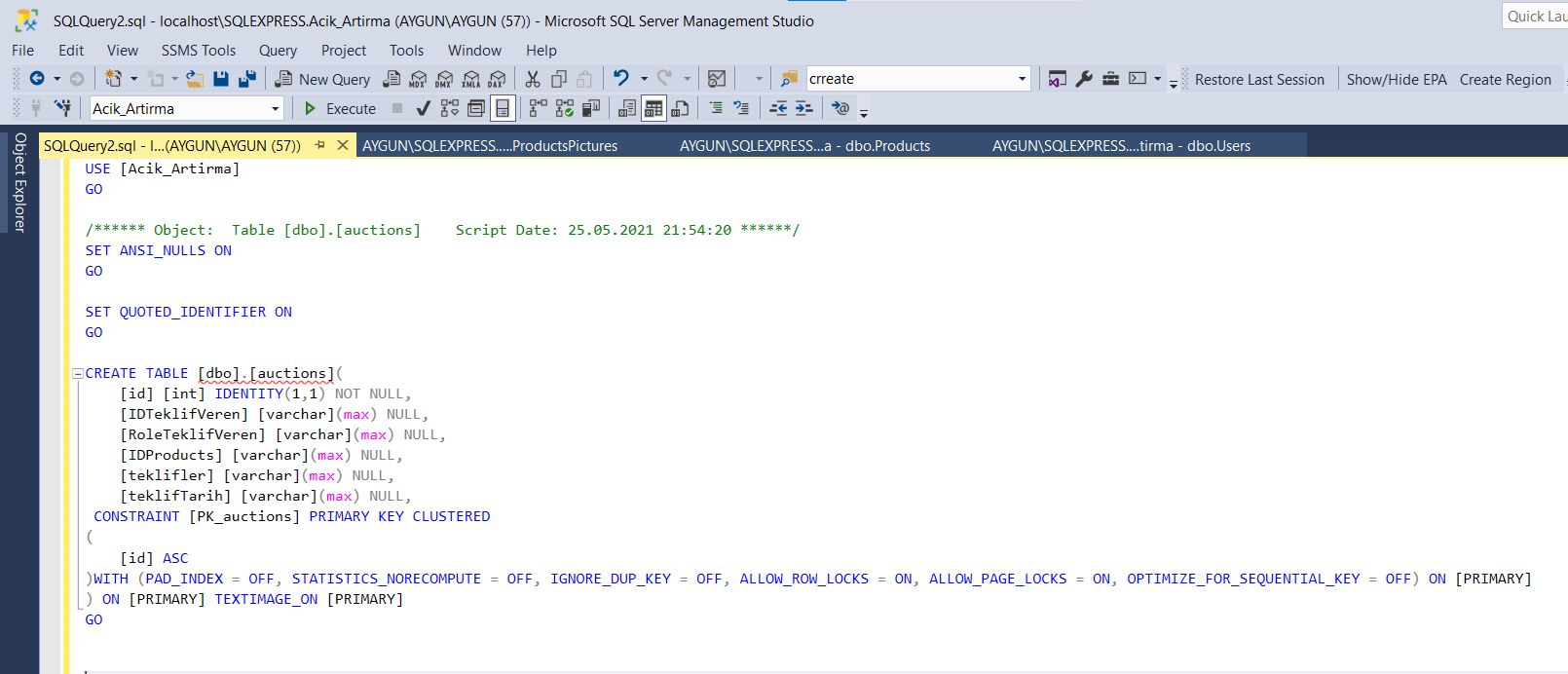


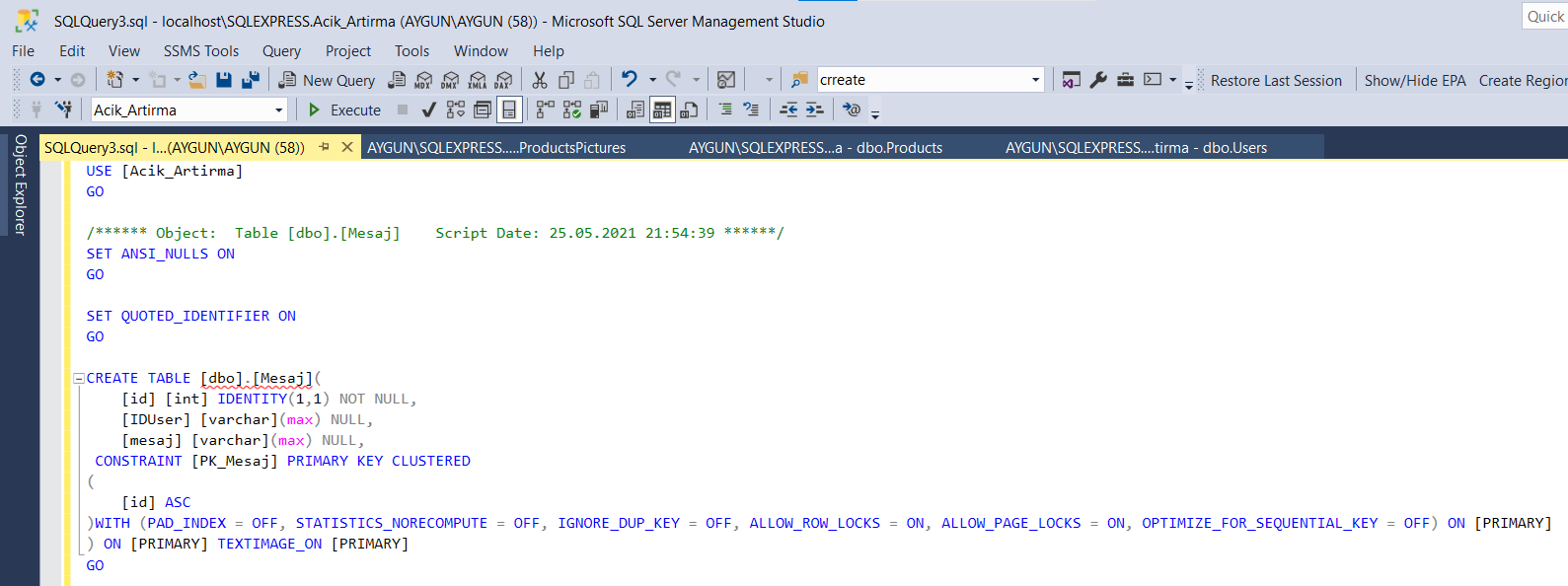
Şekil Saniyede 10000 kullanıcı 10000 iş parçacığı grafik

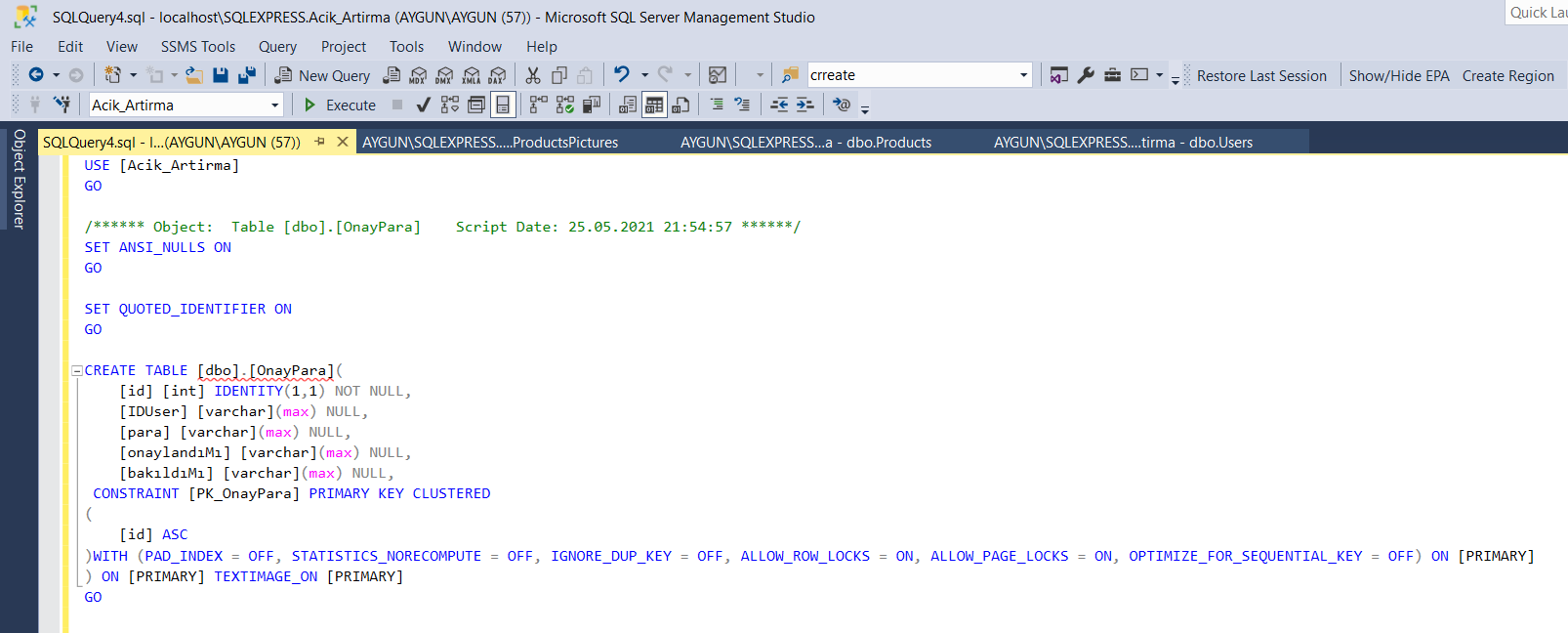
# 4.2 Veri Tasarımı

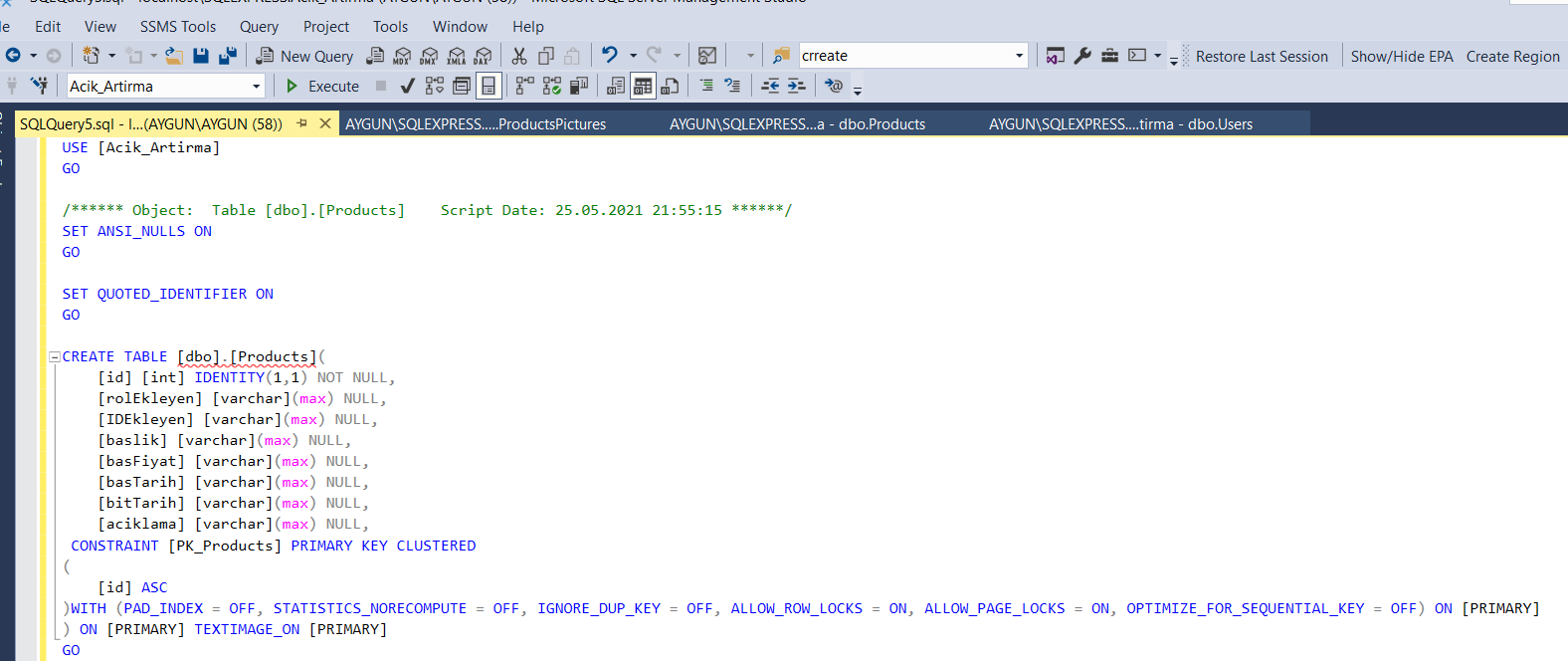
## 4.2.1 Tablo tanımları

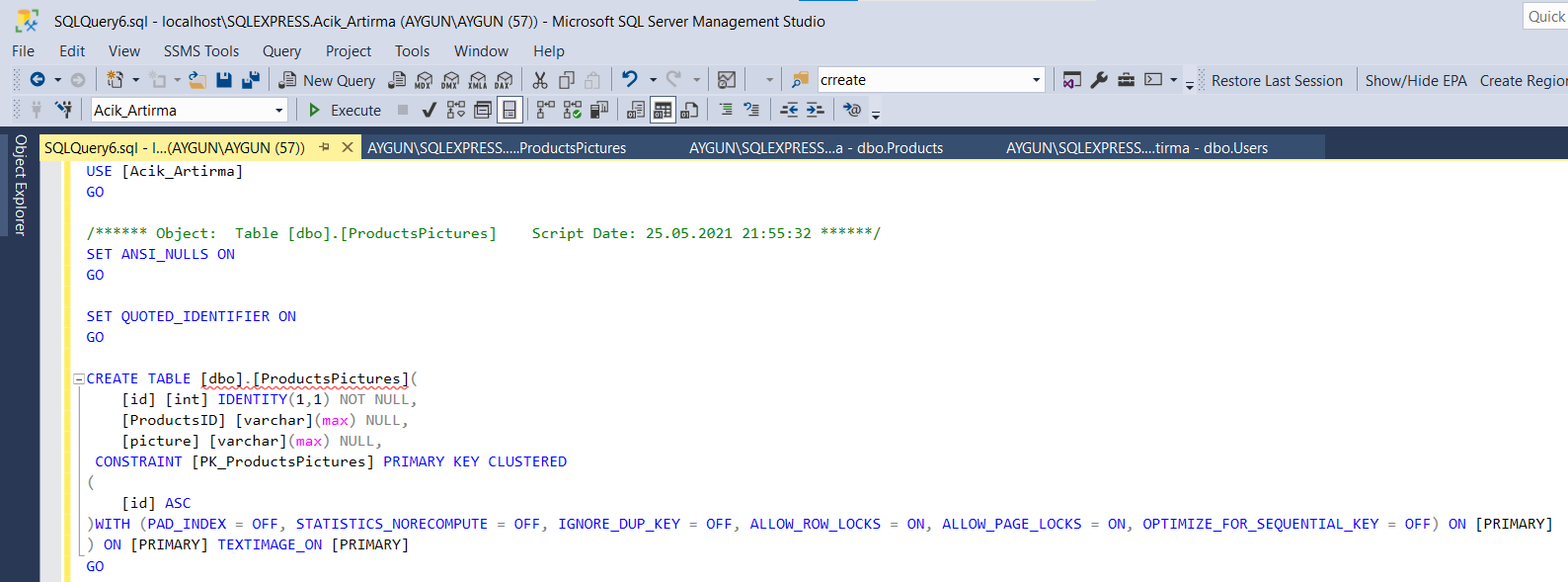


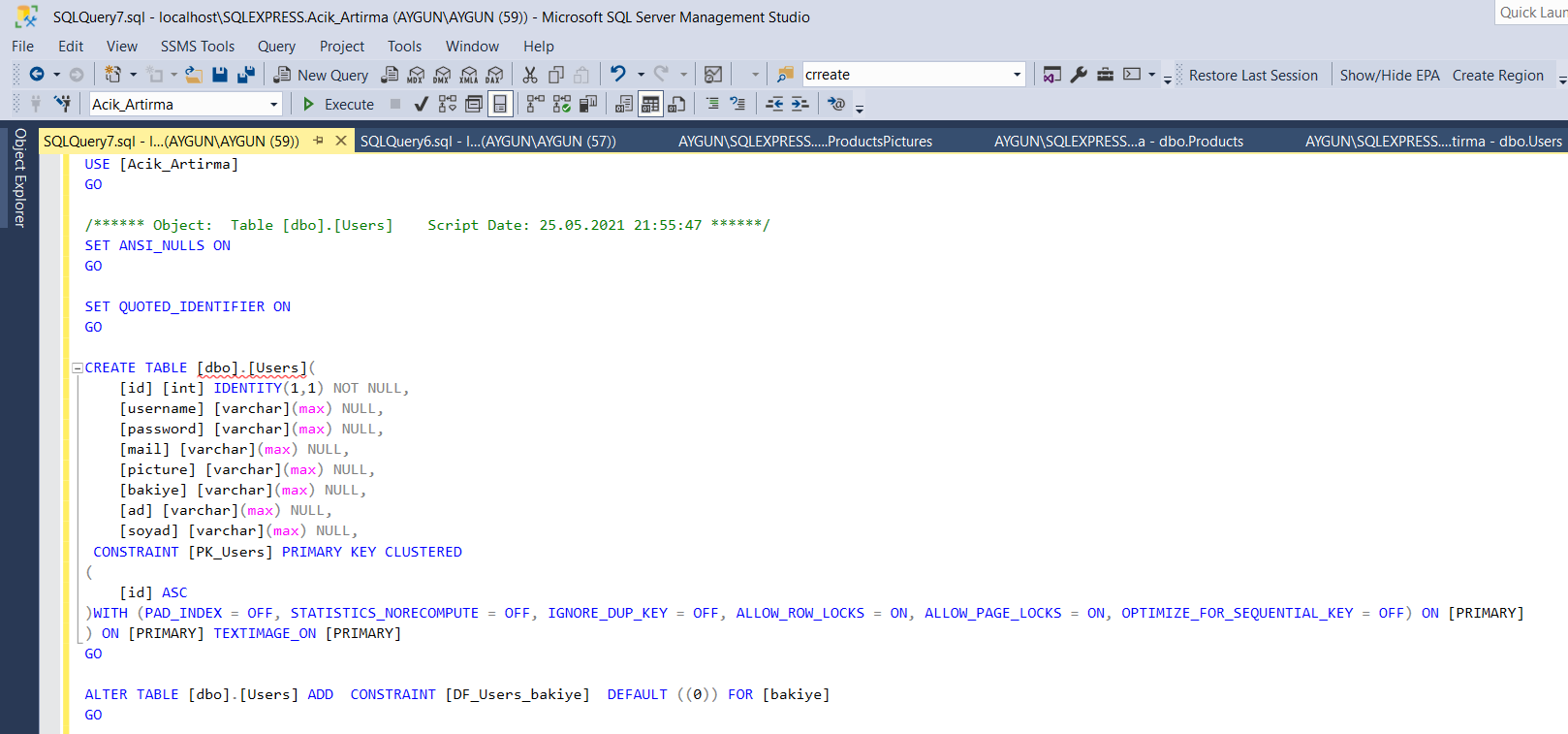










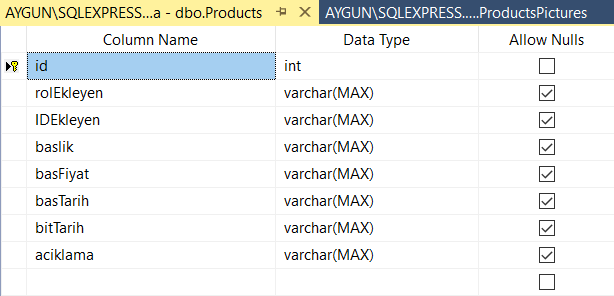


## 4.2.2 Tablo- İlişki Şemaları

Projede 1-n tablo ilişkisi kurulmuştur. “Products” tablosu ile “ProductsPicture” tabloları 1-n örneğidir. Aynı şekilde Users ile Products tabloları da 1-n şeklinde oluşturulmuştur. Yani 1 kullanıcının birden fazla ürünü olup o ürünün de birden fazla tanıtım resmi barındırmaktadır.

## 4.2.4 Değer Kümesi Tanımları

Projede her bir alan varchar(MAX) olarak oluşturulmuştur. Girilmesi gereken değerler uygulama katmanında kontrol edilmektedir.



Şekil Değer kümesi tanım

# 4.3 Süreç Tasarımı

## 4.3.1 Genel Tasarım

## Snort log dosyasının genel yapısı incelendikten sonra buna uygun bir tablo yapısı geliştirildi. Snort to Table menüsünden oluşturulan tabloya verilerinin import edilmesi sağlandı.

## 4.3.2 Modüller

* **Giriş Yapma,**
* **Üye Olma,**
* **Üye Bilgilerini Güncelleme,**
* **Üye Bilgilerini Görüntüleme,**
* **Ürün Ekleme, Çıkarma, Güncelleme ve Görüntüleme,**
* **Adminin müşteri ve ürünlere ait CRUD işlemleri,**
* **Müşterilerin ürünlere ait teklif vermesi,**
* **Ürünlerin başlangıç-bitiş tarihlerine göre durumlarını belirtme,**
* **Admine mesaj yazma,**
* **Admine para talebinde bulunma,**
* **Adminin gelen para isteklerini kabul etmesi ve reddetmesi,**
* **Çıkış yapma**

5. SİSTEM GERÇEKLEŞTİRİMİ

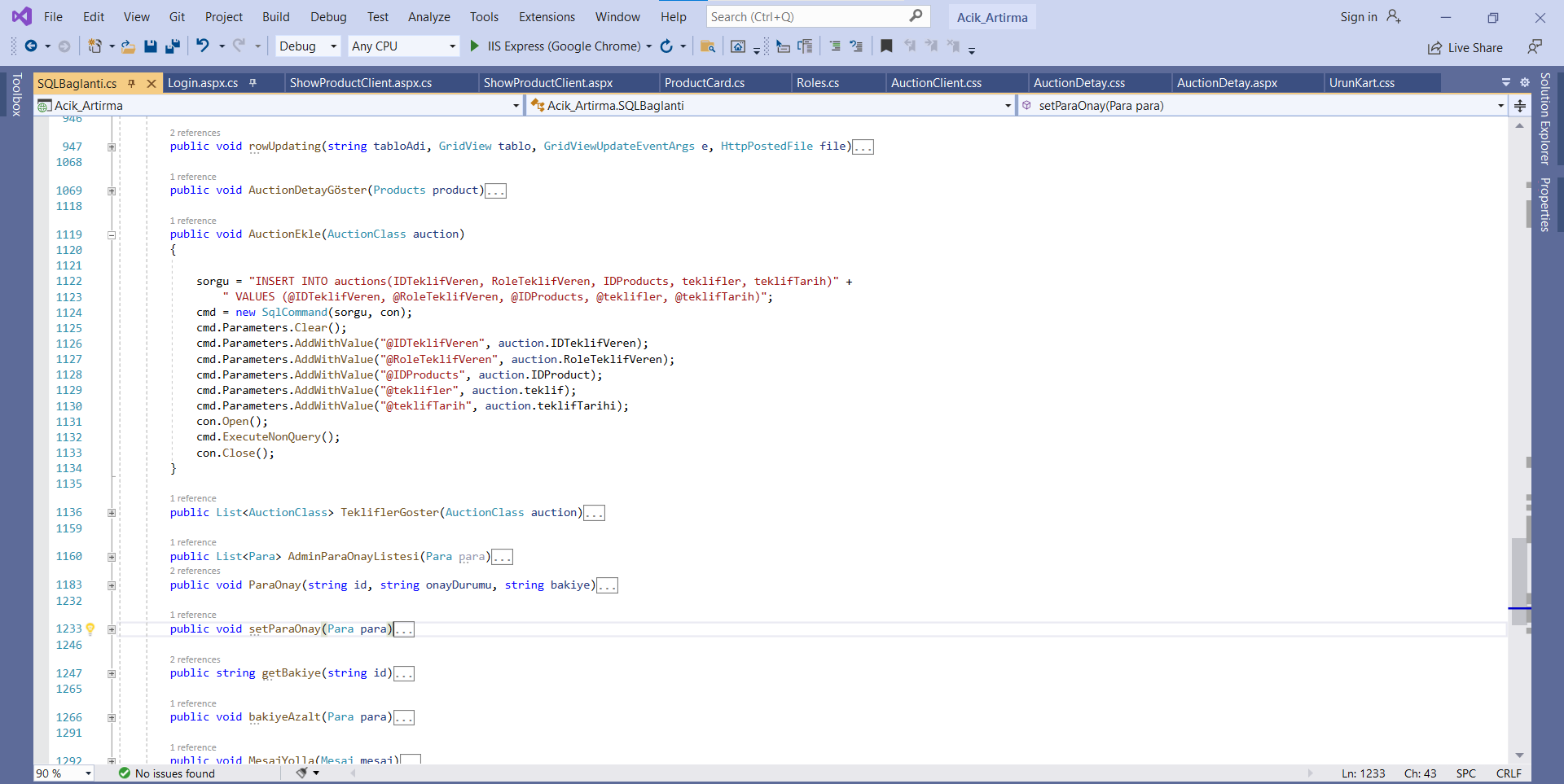
# 5.1. Giriş

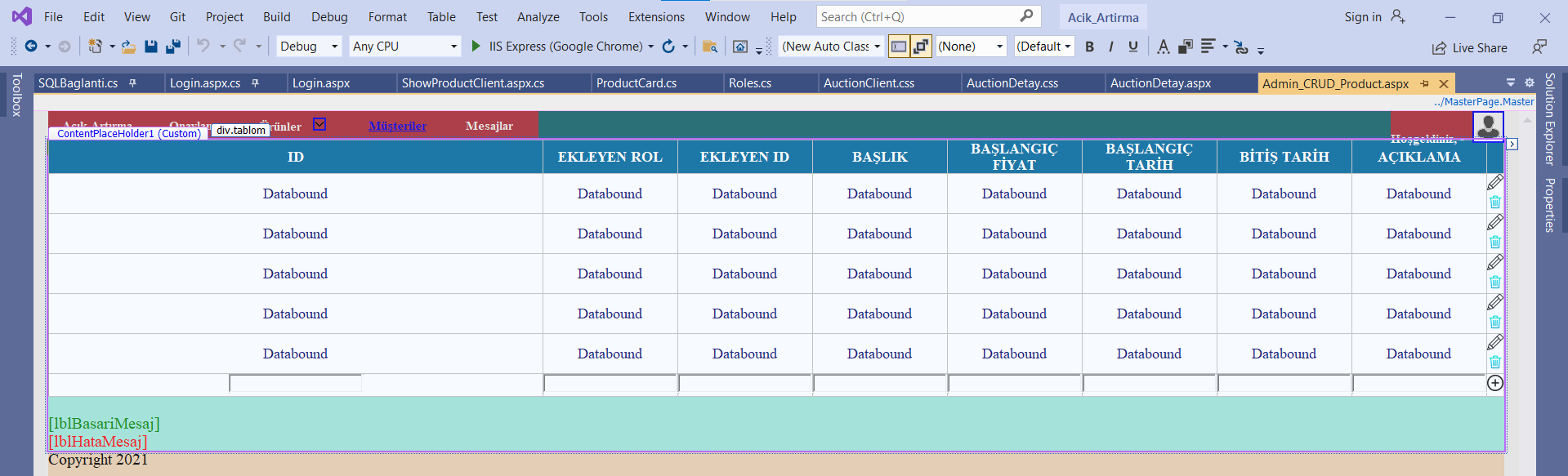
Proje, klasik, ASP .NET kodlaması ile oluşturulan 12+ modülün entegrasyonundan oluşmaktadır. ASP .NET kod yazımı için Visual Studio PRO 2019 IDE kullanılmıştır. Veritabanı kontrolleri için Microsoft SQL Server Managment Studio kullanılmıştır.

# 5.2. Yazılım Geliştirme Ortamları

## 5.2.1 Programlama Dilleri

Proje tamamen C#, JQuery, HTML, CSS Visual Studio 2019 PRO IDE kullanılarak gerçekleştirilmiştir. MSSMS yönetim sistemi kullanılmıştır.





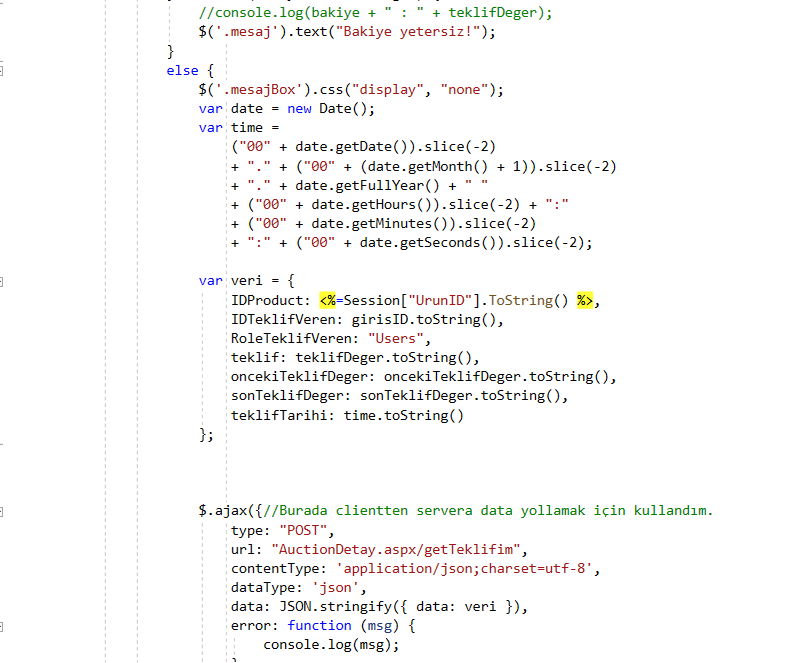
CASE diyagramları için <https://app.diagrams.net/> sitesindeki online araçlar kullanılmıştır.

# 5.3. Kodlama Stili

Kodlamamızı, Visual Studio 2019 PRO IDE ile yaptık. IDE’nin varsayılan kod stilini kullandık, herhangi bir özelleştirme yapmadık.

## 5.3.1 Açıklama Satırları

Açıklama Satırları Delphi IDE’nin varsayılan template ile yapıldı, herhangi bir özelleştirme yapılmadı.

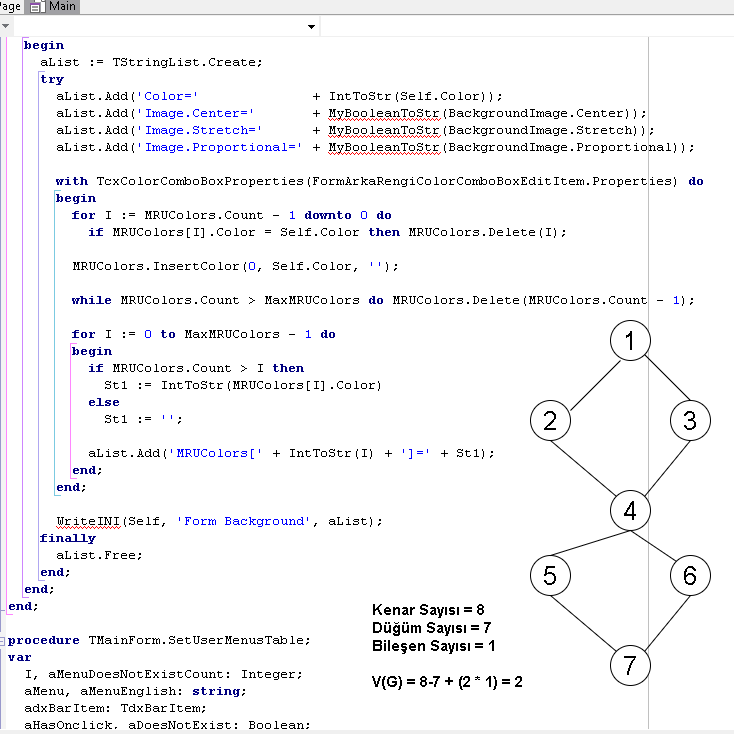


Şekil Açıklama Satırlarının Kullanımı

# 5.4. Program Karmaşıklığı

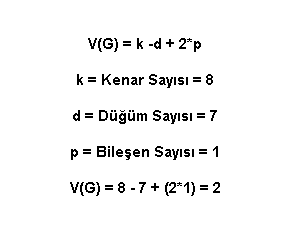
McCabe karmaşıklık ölçütü kullanılmıştır. McCabe karmaşıklık ölçütünün hesaplanmasından önce programın çizge biçimine dönüştürülmesi gerekmektedir.

## 5.4.1 Programın Çizge Biçimine Dönüştürülmesi



Şekil McCabe Metodu ile Programa karmaşıklığının bulunması

## 5.4.2 McCabe Karmaşıklık Ölçütü Hesaplama



# 5.5. Olağan Dışı Durum Çözümleme

Olağan dışı durum, bir programın çalışmasının, geçersiz ya da yanlış veri oluşumu ya da başka nedenlerle istenmeyen bir biçimde sonlanmasına neden olan durum olarak tanımlanmaktadır.

## 5.5.1 Olağandışı Durum Tanımları

Açık Artırma Site için, olağan dışı durumları bir unsurun;

* Çalışmaması
* Yavaş çalışması
* Beklendiği gibi çalışmaması

Noktalarından değerlendirebiliriz.

## 5.5.2 Farklı Olağandışı Durum Çözümleme Yaklaşımları

**Program stabil ve performanslı şekilde çalışmakta olup, olumsuz bir durum beklenmemektedir.**

## 5.6.2 Gözden Geçirme Sırasında Kullanılacak Sorular

### 5.6.2.1 Öbek Arayüzü

|  |  |  |
| --- | --- | --- |
| Gözden Geçirme Konusu | Admin | User |
| Öbek tek bir işlevsel amacı yerine getiriyor mu? | ✓ | ✓ |
| Öbek adı, işlevini açıklayacak biçimde anlamlı olarak verilmiş mi? | ✓ | ✓ |
| Öbek tek giriş ve tek çıkışlı mı? | ✓ | ✓ |
| Öbek bir işlev ise, parametrelerinin değerini değiştiriyor mu? | ✓ | ✓ |

### 5.6.2.2 Giriş Açıklamaları

|  |  |  |
| --- | --- | --- |
| Gözden Geçirme Konusu | Admin | User |
| Öbek, doğru biçimde giriş açıklama satırları içeriyor mu? | ✓ | ✓ |
| Giriş açıklama satırları, öbeğin amacını açıklıyor mu? | ✓ | ✓ |
| Giriş açıklama satırları, parametreleri, küresel değişkenleri içeren girdileri ve kütükleri tanıtıyor mu? | - | - |
| Giriş açıklama satırları, öbekte yapılan değişikliklere ilişkin tanımlamaları içeriyor mu? | - | - |
| Giriş açıklama satırları, öbekteki olağan dışı durumları tanımlıyor mu? | - | - |
| Her paragrafı açıklayan kısa açıklamalar var mı? | ✓ | ✓ |

### 5.6.2.3 Veri Kullanımı

|  |  |  |
| --- | --- | --- |
| Gözden Geçirme Konusu | Admin | User |
| İşlevsel olarak ilintili bulunan veri elemanları uygun bir mantıksal veri yapısı içinde gruplanmış mı? | ✓ | ✓ |
| Değişken adları, işlevlerini yansıtacak biçimde anlamlı mı? | ✓ | ✓ |
| Değişkenlerin kullanımları arasındaki uzaklık anlamlı mı? | ✓ | ✓ |
| Her değişken tek bir amaçla mı kullanılıyor? | ✓ | ✓ |
| Dizin değişkenleri kullanıldıkları dizinin sınırları içerisinde mi tanımlanmış? | ✓ | ✓ |
| Tanımlanan her gösterge değişkeni için bellek ataması yapılmış mı? | ✓ | ✓ |

### 5.6.2.4 Öbeğin Düzenlenişi

|  |  |  |
| --- | --- | --- |
| Gözden Geçirme Konusu | Admin | User |
| Modüller birleşimi uyumlu mu? | ✓ | ✓ |
| Modüller arası veri aktarımları sağlanıyor mu? | - | - |
| Bütün modüller birleştiğinde sistem çalışıyor mu? | ✓ | ✓ |

### 5.6.2.5 Sunuş

|  |  |  |
| --- | --- | --- |
| Gözden Geçirme Konusu | Admin | User |
| Her satır, en çok bir deyim içeriyor mu | ✓ | ✓ |
| Bir deyimin birden fazla satıra taşması durumunda, bölünme anlaşılabilirliği kolaylaştıracak biçimde anlamlı mı? | ✓ | ✓ |
| Koşullu deyimlerde kullanılan mantıksal işlemler yalın mı? | ✓ | ✓ |
| Bütün deyimlerde, karmaşıklığı azaltacak şekilde parantezler kullanılmış mı? | ✓ | ✓ |
| Bütün deyimler, belirlenen program stiline uygun olarak yazılmış mı? | ✓ | ✓ |
| Öbek yapısı içerisinde akıllı "programlama hileleri" kullanılmış mı? | - | - |

6. DOĞRULAMA VE GEÇERLEME (TEST)

# 6.1. Giriş

Geliştirilecek bilgi sistemi yazılımının doğrulanması ve geçerlenmesi, üretim süreci boyunca süren etkinliklerden oluşur. Söz konusu etkinlikler:

* Yazılım belirtimlerinin ve proje yaşam sürecindeki her bir etkinlik sonunda alınan çıktıların, tamam, doğru, açık ve önceki belirtimleri tutarlı olarak betimler durumda olduğunun doğrulanması.
* Proje süresince her bir etkinlik ürününün teknik yeterliliğinin değerlendirilmesi ve uygun çözüm elde edilene kadar aktivitenin tekrarına sebep olması. Projenin bir aşaması süresince geliştirilen anahtar belirtimlerin önceki belirtimlerle karşılaştırılması.
* Yazılım ürünlerinin tüm uygulanabilir gerekleri sağladığının gerçeklenmesi için sınamaların hazırlanıp yürütülmesi biçiminde özetlenebilir.



Şekil Doğrulama ve Geçerleme

# Sınama Kavramları

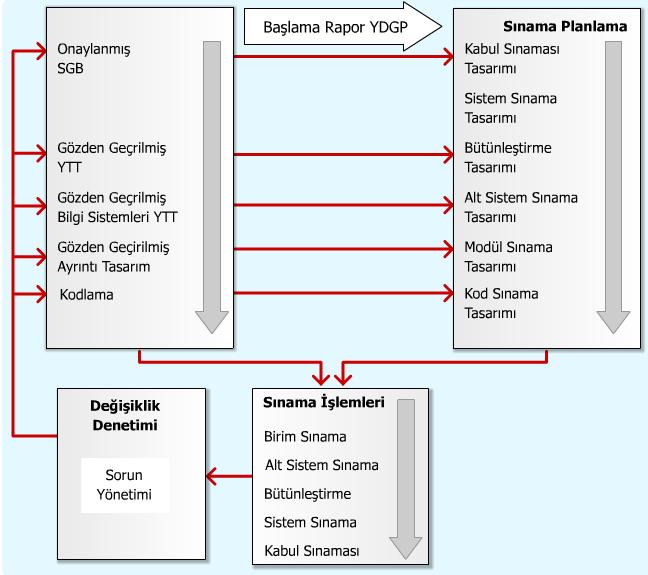
**Birim Sınama:** Proje çalışması olarak, standart AS kurulumuna eklenen 2 adet modülün bir birinden bağımsız olarak sınanması sürecidir.

**Alt Sistem Sınama:** Users, başka bir veri tablo ile işleyişi ve uyumu test edildi. Başarılı bulundu

**Sistem Sınama:** Standart AS kurulumuna, geliştirmiş olduğumuz modüllerin tümü birden etkinleştirilip sınaması yapıldı. Hata ya da uyum sorunu ortaya çıkmadı

**Kabul Sınama:** Proje gereksinimlerini karşıladığı, performansta bir düşüş olmadığı görüldü.

# 6.3. Doğrulama ve Geçerleme Yaşam Döngüsü



# 6.4. Sınama Yöntemleri

## 6.4.1 Beyaz Kutu Sınaması

|  |  |  |
| --- | --- | --- |
| Sınama Noktası | Durum | Sınamayı Yapan |
| Bütün bağımsız yolların en azından bir kere sınanması, | ✓ | Abdullah AYGÜN |
| İç veri yapılarının denenmesi | ✓ |
| Bütün döngülerin sınır değerlerinde sınanması | ✓ |
| Bütün mantıksal karar noktalarında iki değişik karar için sınamaların yapılması, | ✓ |

## Temel Yollar Sınaması

Toplam Dal Sayısı (E) = N (Toplam düğüm sayısı) + 2 🡺 E = 7 +2 = 9

# 6.5. Sınama ve Bütünleştirme Stratejileri

## 6.5.1 Yukarıdan Aşağı Sınama ve Bütünleştirme

## 6.5.2 Aşağıdan Yukarıya Sınama ve Bütünleştirme

# 6.6. Sınama Planlaması

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ZAMAN  İŞ | Proje Başlangıç Tarihi:01/03/2021 | | |  | |  | | | | | | | | HAFTALAR | | | | | |
| **10. Hafta** | | | | | | | | **11.Hafta** | | | | | | | | | | 12 |
| Sistem Doğrulanması | **Birim Sınama** | | | | | | | **Alt Sistem Sınama** | | | | **Sistem Sınama** | | | | **Kabul Sınama** | | |  |
| 1 | 2 | 3 | | 4 | | 5 | 6 | | 7 | 1 | 2 | 3 | | 4 | 5 | 6 | 7 |  |
|  |  |  | |  | |  |  | |  |  |  |  | |  |  |  |  |  |
| 1-2. gün: Admin Dizayn   1. 3-4. Gün: User Dizayn | | | | | | | Snort To Tablo modülü veri aktarımında kaybolan veriler | | | | Sistem yük, stres, performans testi uygulandı. | | | | UAT (Kullanıcı Kabul Testi) yapıldı. Sistem protipten çıkarıldı. | | |  |

# 6.7. Sınama Belirtimleri

|  |  |  |
| --- | --- | --- |
| YAZILIM SINAMA RAPORU | | |
| Yazılım Tanımı |  |  |
| Programcı Ad-Soyad | Abdullah AYGÜN |
| Programlama Dili | Classic ASP .NET Web Forms |
| Bilgisayar Sistemi | Intel i5-8300H |
| İşletim Sistemi | Windows10 |
| Sınama Türü | Beyaz Kutu |
| Problemin Çözümü | 137. satırdaki dizi index hatası düzeltildi. |

# 6.8. Yaşam Döngüsü Boyunca Sınama Etkinlikleri

7. BAKIM

# 7.1 Giriş

* Projemize ait modüllerin “Düzeltici Bakımı” Doğrulama ve Geçerleme sürecinde gerçekleştirilen testlerle yapıldı. Bu konuda modüllerimizin kodlamasında bakım ile ilgili yapmamız gereken ekstra bir işlem bulunmamaktadır.
* Modüllerin sağlıklı şekilde kurulumu, versiyon kontrolü, logging işlemleri tümüyle standalone bir exe dosya ile sağlanmakta olup, ilave işleme gerek yoktur.

# 7.2 Kurulum

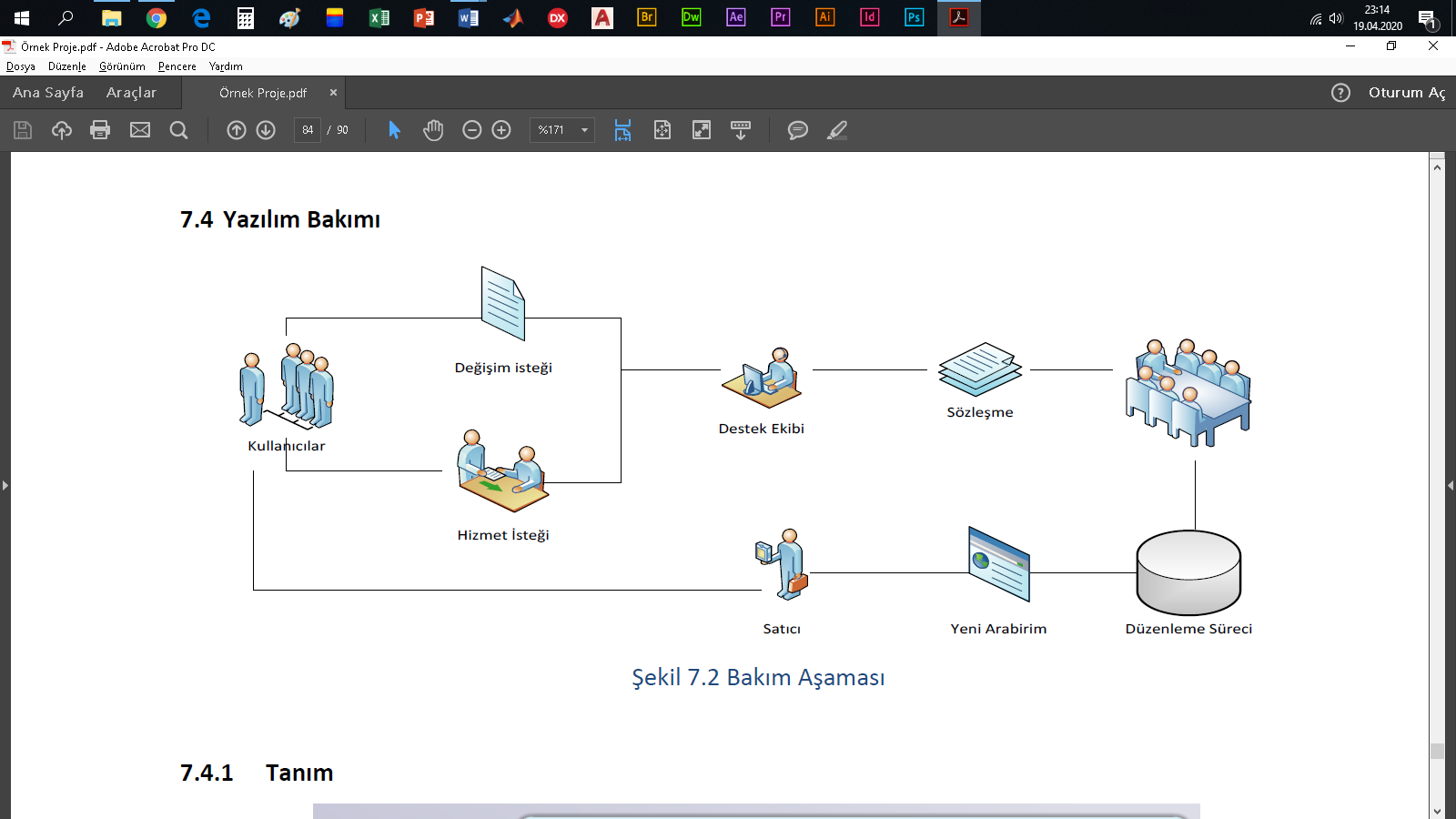
Projenin kurulumu için herhangi gib host sunucuya dosyalar upload edilir.

Veritabanı için SQL Server destekleyen bir sunucu bulunur.

# 7.3 Yerinde Destek Organizasyonu

Proje her hangi bir destek organizasyonuna ihtiyaç duymamaktadır.

# 7.4 Yazılım Bakımı



## 7.4.1 Tanım

Bir yazılımın bakımı(software maintenance), yazılımın geliştirilmesi tamamlandıktan sonraki süreçte, yazılımın ömrünü sağlıklı bir şekilde sürdürmesini sağlayan yaklaşımlar bütünüdür. Yazılım bakımı Yazılım Geliştirme Sürecinin(Software Development Process) son parçasını oluşturur. Müşteri gereksinimlerine göre oluşturulan yazılım çıktısı, değişen teknolojik araçlara ve müşteri gereksinimlerine göre sürekli bakımının yapılması, denetlenmesi gerekir. Nasıl bir arabanın yıllık bakımı yapılıyorsa, yazılım ürünün de belirli sürümler dahilinde geliştirilmesi, iyileştirilmesi ve korunması yapılmalıdır.

## 7.4.2 Bakım Süreç Modeli

IEEE 1219-1998 Yazılım Bakım Standardı esaslarına göre sağlanacaktır.

8. SONUÇ

# 8. Sonuç

Auction System(AS) projesi başarılı bir şekilde hayata geçirilmiş oldu. Bazı birkaç ek özellikler getirilerek kullanıcıların işlerinin kolaylaştırılması sağlanabilir. Ya da ek güvenlik önemleri ile sistem daha güvenli bir yer olması sağlanabilir.

9. KAYNAKLAR

<https://ue.firat.edu.tr/course/view.php?id=77916> (Ders anlatım videoları ve dokümanlar)

<https://www.youtube.com/watch?v=eJM0lchllLI&ab_channel=CodebyPROWARE>

<https://www.youtube.com/watch?v=xVgtAdxhIxk&ab_channel=DawaCIT>

<https://www.youtube.com/watch?v=nbn8c8MwsNM&ab_channel=DawaCIT>

<https://stackoverflow.com/questions/174849/asp-net-routing-with-web-forms>

<https://stackoverflow.com/questions/57057013/getting-the-id-of-the-div-on-button-click-where-div-contains-that-button>