

İçindekiler

A.PROJENİN TANIMI	4
B.PROJENİN SAHİP OLDUĞU NİTELİKLER.....	4
C.YAZILIM GELİŞTİRME YAŞAM DÖNGÜSÜ	7
1.Çağlayan(ŞELELE) SÜREÇ MODELİ	7
2.YOURDAN YAPISAL SİSTEM METODOLOJİSİ	9
D.YAZILIM GELİŞTİRME YAŞAM DÖNGÜSÜ TEMEL ADIMLARI	10
1.PLANLAMA	10
1.1.PROJE KAPSAMININ BELİRLENMESİ	10
1.2.PROJE KAYNAKLARININ BELİRLENMESİ	10
1.2.1.İNSAN KAYNAKLARI	11
1.2.2.DONANIM KAYNAKLARI.....	11
1.2.3.YAZILIM KAYNAKLARI	12
1.3.PROJE MALİYETİNİN BELİRLENMESİ	12
1.4. PROJE EKİP YAPISININ OLUŞTURULMASI.....	16
1.4.1.YÜKLENİCİ PROJE EKİP YAPISI	16
1.4.2.İŞ SAHİBİ PROJE EKİP YAPISI	16
1.5. AYRINTILI PROJE PLANININ OLUŞTURULMASI.....	16
1.5.1.PROJE İŞ-ZAMAN PLANI	16
1.5.2.KAYNAK YÖNETİM PLANI.....	17
1.5.3.SINAMA PLANI	18
1.5.4.EĞİTİM PLANI.....	18
1.5.5.BAKIM PLANI	18
2.SİSTEM ÇÖZÜMLEME	19
2.1.GİRİŞ	19
2.2.GEREKSİNİMLERİN BELİRLENMESİ.....	19
2.2.1.FİZİKSEL ÇEVRE	19
2.2.2.ARAYÜZLER.....	19
2.2.3.KULLANICI VE İNSAN ETMENİ.....	19
2.2.4.İŞLEVSELLİK.....	20
2.2.5.BELGELEME.....	20
2.2.6.VERİ.....	20
2.2.7.KAYNAKLAR	20

2.2.8 GÜVENLİK	21
2.2.9 KALİTE GÜVENCESİ	21
2.3 SİSTEM ÇÖZÜMLEME ÇALIŞMASI	21
2.3.1 MEVCUT SİSTEMİN İNCELENMESİ ve DEĞERLENDİRİLMESİ	21
2.4 TASARLANAN SİSTEMİN MANTIKSAL MODELİ	21
2.4.1 USE – CASE DİYAGRAMLARI	22
2.4.2 VERİ MODELİ	23
2.4.2.1 NESNE- İLİŞKİ ŞEMASI	23
2.4.2.2 VERİ SÖZLÜĞÜ	24
2.4.3 KULLANICI ARAYÜZ TASARIMI	31
2.4.3.1 OTOMASYON SİSTEMİNE GİRİŞ EKRANI	31
2.4.3.2 PERSONEL GİRİŞ ARAYÜZÜ(Dashboard)	32
2.4.3.3 ÜYE YÖNETİMİ->ÜYE EKLE ARAYÜZÜ	33
2.4.3.4 ÜYE YÖNETİMİ->ÜYE LİSTELE ARAYÜZÜ	34
2.4.3.5 ÜYE YÖNETİMİ->TOPLU ÖDEMELER ARAYÜZÜ	34
2.4.3.6 ÜYE YÖNETİMİ->TOPLU BAŞVURU ARAYÜZÜ	35
2.4.3.7 RAPOR YÖNETİMİ->DESTEK BAŞVURUSU OLANLAR ARAYÜZÜ	35
2.4.3.8 RAPOR YÖNETİMİ->DESTEK BAŞVURUSU OLMAYANLAR ARAYÜZÜ	36
2.4.3.9 RAPOR YÖNETİMİ->ÇOBAN DESTEĞİ OLANLAR ARAYÜZÜ	36
2.4.3.10 FORM-ŞABLONLAR->YENİ ŞABLON EKLE ARAYÜZÜ	37
2.4.3.11 FORM-ŞABLONLAR->ŞABLON LİSTESİ ARAYÜZÜ	37
2.4.3.12 PROJE YÖNETİMİ ->PROJE OLUŞTUR VE PROJE LİSTESİ ARAYÜZÜ	38
2.4.3.13 PERSONEL YÖNETİMİ ->PERSONEL LİSTESİ ARAYÜZÜ	38
2.4.3.14 PERSONEL YÖNETİMİ->PERSONEL EKLE ARAYÜZÜ	39
2.4.3.15 PERSONEL YÖNETİMİ-> BİLGİLERİMİ GÜNCELLE ARAYÜZÜ	39
2.4.3.16 AYARLAR->POZİSYON YÖNETİMİ ARAYÜZÜ	40
2.4.3.17 FATURALAR ARAYÜZÜ	40
2.4.3.18 HATIRLATMALAR ARAYÜZÜ.....	41
2.4.3.18 TOPLU SMS YÖNETİM ARAYÜZÜ	42
2.4.3.19 HIZLI İŞLEMLER ARAYÜZÜ.....	42
2.4.3.20 ÜYE DETAY ARAYÜZÜ	43
2.4.3.21 KULLANICI DETAY ARAYÜZÜ	43
3.TASARIM	44
3.1 GİRİŞ	44

3.2 TANIMLAR VE KISALTMALAR	44
3.3 GENEL TASARIM	44
3.4 VARSAYIMLAR VE KISITLAMALAR	45
3.5 VERİ TASARIMI.....	45
3.5.1 TABLO TANIMLARI	45
3.5.2 MYSQL DATABASE DİAGRAM	46
3.6 KULLANICI ARAYÜZ TASARIMI	47
3.7 ENTEGRASYON VE TEST GEREKSİNİMLERİ	47
3.8 ORTAK ALT SİSTEMLERİN TASARIMI	47
3.8.1 YETKİLENDİRME ALT SİSTEMİ	47
3.8.2 YEDEKLEME ALT SİSTEMİ	47
3.8.3 GÜVENLİK ALT SİSTEMİ.....	48
4.GERÇEKLEŞTİRİM	48
4.1 GİRİŞ	48
4.2 YAZILIM GELİŞTİRME ORTAMLARI	48
4.3 KODLAMA SİTİLİ.....	48
4.4 TASARLANAN SİSTEMİN KODLANMASI	49
4.5 KOD GÖZDEN GEÇİRME	53
5.YAZILIM DOĞRULAMA VE GEÇERLEME	54
5.1 GİRİŞ	54
5.2 SINAMA YÖNETİMİ	54
5.3 SINAMA VE BÜTÜNLEŞTİRME STRATEJİSİ.....	54
5.3.1 AŞAĞIDAN YUKARIYA SINAMA VE BÜTÜNLEŞTİRME	54
5.4 SINAMA RAPORU.....	55
6.KURULUM-BAKIM-EĞİTİM.....	55
6.1 GİRİŞ	55
6.2 KURULUM	55
6.3 BAKIM	55
6.4 EĞİTİM	55

A.PROJENİN TANIMI

Proje, küçükbaş hayvan yetiştiricileri birliği üye takip otomasyonudur. Ülkemiz tarım ve hayvancılık açısından büyük bir potansiyele sahiptir. Devletimiz hayvancılığın gelişmesi için çeşitli teşvikler sunmaktadır. Bu teşviklerden yararlanmak isteyen, hayvancılıkta verimliliği ve üretimi artırmak isteyen yetiştiriciler, illerde hizmet veren hayvan yetiştiricileri birliğine başvurularını yapmaktadır. Otomasyonumuz burada devreye girerek üyelerin kayıtlarını takip etmeyi ve yönetmeyi kolaylaştıracaktır.

Proje genel olarak hayvan yetiştiricileri birliğine kayıt olan ve başvuran üyelere ait detaylı üyelik bilgilerini (Adı, Soyadı, TC Kimlik No, Telefon, Adres, IBAN vs.), başvuru bilgilerini, çoban ve hayvan destek başvurularını ve bilgilerini, hibe başvuru ve bilgilerini, devir olaylarını, toplu ya da bireysel aidat bilgilerini tutmakla beraber bunlarla ilgili detaylı raporlama yapabilen, kayıtlı üyeleri toplu sms ile bilgilendirebilen, personele hatırlatma yapabilen, kullanıcı yetkilendirmesi yapılabilen, hızlı erişimleri ile genel özet ekranı ile birçok kolaylık sağlayan web tabanlı bir otomasyondur.

Bu proje sayesinde küçükbaş hayvan yetiştiricileri birliği bütün işlemleri bilgisayar ortamına taşıyarak personele düşen yükü ve yapılan hataları azaltacak ve bilgilerin saklanması ki zorlukları ortadan kaldırılmış olacaktır. Aynı zamanda verilerin güvenliği sağlanarak verilere erişmek ve takip etmek oldukça kolaylaşacaktır.

B.PROJENİN SAHİP OLDUĞU NİTELİKLER

- ❖ Proje web tabanlı olacak, internet ortamından erişilebilecektir.
- ❖ Projede otomasyon sistemine giriş için kullanıcı paneli kullanıcıyı karşılayacaktır.
- ❖ Kullanıcılara, yönetici ve personel pozisyonlarına göre yetkiler verilip, kullanıcılar yetki dâhilinde sistemin belirli alanlarını kullanabileceklerdir.
- ❖ Projede üye yönetimi, rapor yönetimi, proje yönetimi, personel yönetimi, form ve şablon yönetimi, ayarlar, hatırlatmalar, faturalar, toplu sms ve hızlı erişim menüleri bulunur.
- ❖ Üye Yönetimi menüsünde, üye ekleme, mevcut üyeleri listeleme, toplu ödemeler ve toplu başvurular bulunur.

- ❖ Rapor yönetiminde, destek başvurusu olanlar, destek başvurusu olmayanlar, çoban desteği alanları menüsü vardır.
- ❖ Form ve Şablon yönetiminde ekleme ve listeleme menüler vardır.
- ❖ Proje Yönetiminde, yeni proje oluşturma ve listeleme menüsü vardır.
- ❖ Personel Yönetiminde, yeni personel ekleme, mevcut personelleri görüntüleme ve personel bilgilerini güncelleme menüleri bulunur.
- ❖ Ayalar menüsünde pozisyonlar ile ilgili yetkilendirme ve arayüz düzenleme menüsü bulunur.
- ❖ Üye ekleme işlemleriyle üyeler ait ad, soyad, tc kimlik, birlik üye no, e-posta, cep telefonu, ev telefonu, detaylı adres bilgisi, cinsiyeti doğum tarihi, banka hesap bilgileri, üyelik durumu ve üyelik şekilleriyle ilgili bilgileri kayıt altına alabilir.
- ❖ Üyeleri adres, durum, üyelik tipi gibi alanlara göre filtreleyebilir. Bu filtrelere göre çıkan kayıtlar aynı zamanda detaylı şekilde görüntülenebilir. Yazılım aynı zamanda bu verileri Excel tablosuna aktarabilir.
- ❖ Yazılım, üye detayına girdiğimizde üyeye ait genel bilgiler dışında üyelik aidat borcu, hayvan ve çoban desteği başvurusu, , proje bilgileri, yapılan ödeme listeleri, daha önceki yıllarda yapmış olduğu başvurular hareketler, dosyalar, tyyd alanları ve bunlara ait detaylı bilgileri sunabilir.
- ❖ Üye yönetimi ile üyeler sistemden silinebilir ya da üyeliklerini bir başkasına devir edebilir.
- ❖ Giriş bedelleri ya da aidatları toplu şekilde sisteme kaydedebilir.
- ❖ Hayvan desteği almak isteyenlerin başvurularını tek tek ya da liste halinde sisteme ekleyebilir.
- ❖ Başvurular sistemde bulunan şablonlar aracılığı ile alınmaktadır. İhtiyaç halinde yeni şablon eklenip mevcut şablonlar düzenlebilir.
- ❖ Şablonlar altında; aday üyeliğe geçiş, asıl üyeliğe geçiş, üye durum belgesi, üyelikten ayrılma dilekçesi, destek başvuru belgesi, üye devir belgesi, hibe başvuru belgesi gibi şablonlar mevcuttur. Ayrıca kurum içinde kullanılmak üzere hazırlanmış izin dilekçesi ve yazışma örnekleri de bulunmaktadır.
- ❖ Yazılımın en önemli işlevlerinden birisi de raporlama kabiliyetidir. Raporlama ekranı ile adres, dönem, durum ve üyelik bilgilerine göre destek için başvuru olan, olmayan üyelere ait detaylı (üye no, ad soyad, ilçe, mahalle, sokak, tip, durum, destek anaç

hayvan sayısı, dönem gibi) bilgileri hem ekranda görüntüleyebilir hem de Excel ortamına aktarabiliriz.

- ❖ Raporlama ve listeleme ekranlarında arama çubuğu aracılığıyla kayıtlar içerisinde istenilen kayıt aranabilir.
- ❖ Zamandan kazanmak adına hazır raporlama seçenekleri(aidat, proje, üye, belde vs) kullanılabilir.
- ❖ Sistem çoban desteği almak isteyen üyelerin kayıtlarını ve başvurularını da almaktadır.
- ❖ Çoban desteği alanlar adres, dönem, durum ve üyelik bilgilerine göre filtrelenerek (üye no, ad soyad, ilçe, mahalle, sokak, tip, durum, dönem gibi) bilgileri hem ekranda görüntülenebilir hem de Excel ortamına aktarılabilir.
- ❖ Bakanlığın desteklediği projeleri yapmak isteyen üyelerin proje bilgilerini ve durumlarını sistem kayıt altına alabilir. Proje ile ilgili açıklamalar proje dosyaları ve proje üye listesini veri tabanına kaydedebilir.
- ❖ Hibe desteğinden yararlanmak isteyen üyelere ait başvuru ve kayıt bilgilerini de tutabilmektedir.
- ❖ Sistem kayıtlı üyelere toplu ya da tek sms atabilir. Sms ler daha sonra kullanılmak üzere şablon olarak kaydedilebilir. Toplu sms gönderirken alıcı alanı çeşitli filtrelemeler yapılarak düzenlenebilir.
- ❖ Hatırlatmalar menüsü ile kullanıcıların oluşturduğu hatırlatma notları ile eksik ya da hatalı işlemlerin yapılmasının önüne geçilir.
- ❖ Bekleyen ya da ödenen faturaların detaylı bilgileri de faturalar menüsünden erişilebilir.
- ❖ Sistemi ilk açtığımızda genel görünüm altında sisteme son eklenen üyeleri ve oturum açan personele ait bilgileri, tarih saat bilgilerini ve hatırlatmaları gösterir. Aynı zamanda toplam üye sayısını, toplam proje sayısını, hatırlatmaları ve faturaların sayısını da görmemiz açısından genel ekranda bizlere sunar.
- ❖ Sistem Aksiyon menüsü sayesinde yeni proje, yeni üye, yeni şablon, yeni hatırlatma, yeni destekleme ekleme şansını hızlı bir şekilde verir.
- ❖ Pozisyonlar da personel ve yönetici kullanıcıları detaylı şekilde yetkilendirilebilir. Örneğin personel üye ekler listeler ama silemez.
- ❖ Hızlı işlemler menüsü ile yeni üye, yeni proje, üye listesi, proje listesi işlemlerine anında ulaşılır.

C.YAZILIM GELİŞTİRME YAŞAM DÖNGÜSÜ

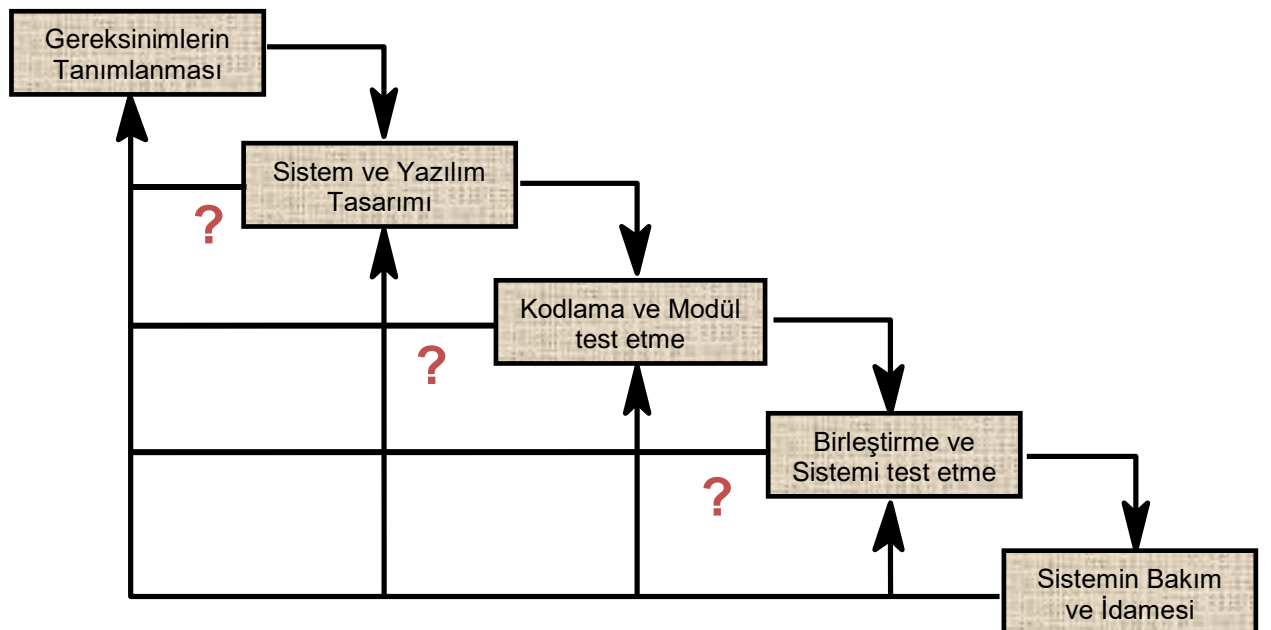
Yazılımın hem üretim, hem de kullanım süreci boyunca geçirdiği tüm aşamalar yazılım geliştirme yaşam döngüsü olarak tanımlanır. Yazılım işlevleri ile ilgili gereksinimler sürekli olarak değiştiği ve genişlediği için, söz konusu aşamalar sürekli bir döngü biçiminde ele alınır. Döngü içerisinde herhangi bir aşamada geriye dönmek ve tekrar ilerlemek söz konusudur.

Yazılım geliştirme yaşam döngüsü temel adımları;

1. PLANLAMA
2. SİSTEM ÇÖZÜMLEME
3. TASARIM
4. GERÇEKLEŞTİRİM
5. YAZILIM DOĞRULAMA VE GEÇERLEME
6. KURULUM VE BAKIM

Yukarıda yazılı olan adımların gerçekleştirilmek amacıyla süreç modelleri kullanılır. Bu modeller yazılım yaşam döngüsünde belirtilen süreçlerin geliştirme aşamasında hangi düzen ya da sırada, nasıl uygulanacağını tanımlarlar. Biz projemizde Çağlayan(Şelale) süreç modelini kullanarak geliştirmeyi tercih ettik.

1.Çağlayan(ŞELE) SÜREÇ MODELİ



Çağlayan(Şelale) modeli, analiz adımı ile başlar. Analiz adımı, tüm yazılım gereksinimleri net bir şekilde belirlenerek analiz dokümanı üretilir. Daha sonra, tasarım adımı yazılımın arayüz, veri tabanı, sınıf vb. tasarımları yapılarak tasarım dokümanı üretilir. Bir sonraki kodlama adımı yazılım; analiz ve tasarım dokümanlarında belirtilen şekilde kodlanır. Test adımı; analiz ve tasarım dokümanlarındaki tüm fonksiyonel ve fonksiyonel olmayan gereksinimler ve tasarımlar için test senaryoları yazılır ve bu test senaryoları icra edilerek yazılımın testleri yapılır. Testlerde bulunan hata hangi adımdan (analiz, tasarım veya kodlama) kaynaklı ise o adıma geri dönülerek hata giderilir. Örneğin, hata kodlamadan kaynaklı ise, sadece kodlama düzeltilir ve tekrar test adımına geçilir. Ancak, hata analiz veya tasarımdan kaynaklı ise bu adımlara gidilerek analiz ve tasarım dokümanları güncellenir, daha sonra kodlama düzeltilir ve test aşamasına geçilir. Test adımı sonunda, yazılımda herhangi bir hatası bulunamaz ise, entegrasyon adımına geçilir ve yazılım, canlı ortama entegre edilerek müşterinin kullanımına açılır.

Çağlayan(Şelale) süreç modelini tercih etmemizin sebepleri;

1. Kullanımı ve anlaması basit, yönetimi kolaydır.
2. Şelale modelinde analiz ve tasarım aşamaları oldukça detaylı yapıldığından, bu adımlar uzun sürmektedir. Ancak, analiz ve tasarım aşamalarında gereksinimlerin ve tasarımın net bir şekilde ortaya konulmasından dolayı, kodlama ve test aşamasının çok kısa sürmesi.
3. Projenin iyi tanımlı bir proje olması ve üretimi için az zaman gerektirmesi.
4. Projenin çok büyük bir proje olmaması, gereksinimlerin iyi anlaşılması.
5. Bazı modellere göre geri dönüşlerinin iyi tanımlanması.
6. Projenin safhaları ayrı olduğundan iş bölümü ve iş planının projenin en başında net bir şekilde belli olması.
7. Belgeleme işlemini ayrı bir iş olarak ele almaması üretimin doğal parçası olarak görmesi.
8. İki kere ölçüp bir kere biçmek yaklaşımının kurumda uygulanmasını sağlaması.
9. Proje dokümantasyonu hazırlanmaya zorlaması.

10. Hataların hangi aşamalarda ve kimin tarafından kaynaklandığının bulunmasını kolaylaştırması.
11. Her sonraki aşamaya geçmeden önce, önceki aşamanın tamamlanması, bu sayede önceki aşamada ayrılan kaynakların serbest kalması ve farklı projelere aktarılabilmesi.

2.YOURDAN YAPISAL SİSTEM METODOLOJİSİ

Metodoloji BT projesi ya da yazılım yaşam döngüsü aşamaları boyunca kullanılabilecek ve birbirleriyle uyumlu yöntemler bütünüdür.

Proje boyunca takip edilecek olan süreç modeline uygun, nesne yönelimli ve uygulanabilirliği kolay olan Yourdan Yapısal Sistem metodolojisi kullanılmaya karar verildi. Bu metodolojinin kullanılmasının diğer bir sebebi ise bir CASE aracı tarafından desteklenebilir olmasıdır. Aynı zamanda Çağlayan modelini temel almasıdır.

Yourdan metodolojisi aşağıdaki tablo ile açıklanabilir;

Aşama	Kullanılan Yöntem ve Araçlar	Ne için Kullanıldığı	Çıktı
Planlama	Veri Akış Şemaları, Süreç Belirtileri, Görüşme, Maliyet Kestirim Yöntemi, Proje Yönetim Araçları	Süreç İnceleme Kaynak Kestirimi Proje Yönetimi	Proje Planı
Analiz	Veri Akış Şemaları, Süreç Belirtileri, Görüşme, Nesne ilişkili şemaları Veri	Süreç Analizi Veri Analizi	Sistem Analiz Raporu
Analizden Tasarıma Geçiş	Akışa Dayalı Analiz, Süreç belirtilerinin program tasarım diline dönüştürülmesi, Nesne ilişkili şemalarının veri tablosuna dönüştürülmesi	Başlangıç Tasarımı Ayrıntılı Tasarım Başlangıç Veri tasarımı	Başlangıç Tasarım Raporu
Tasarım	Yapısal Şemalar, Program Tasarım Dili, Veri Tabanı Tabloları	Genel Tasarım Ayrıntılı Tasarım Veri Tasarımı	Sistem Tasarım Raporu

D.YAZILIM GELİŞTİRME YAŞAM DÖNGÜSÜ TEMEL ADIMLARI

1.PLANLAMA

Yazılım geliştirme yaşam döngüsünün ilk aşamasıdır. Başarılı bir proje gerçekleştirebilmek için planlama çalışmasının önemi büyüktür. Bu aşama tüm proje boyunca sürekli kullanılacak, güncellenecek ve gözden geçirilecek bir aşamadır. Birlik üye takip otomasyonu planlanması aşamasında yapılacak olan çalışmalar aşağıdaki şekildedir.

- ❖ Proje kapsamının belirlenmesi
- ❖ Proje kaynaklarının belirlenmesi
- ❖ Proje maliyetlerinin kestirilmesi
- ❖ Proje ekip yapısının oluşturulması
- ❖ Ayrıntılı proje planı yapılması
- ❖ Projenin izlenmesi

1.1.PROJE KAPSAMININ BELİRLENMESİ

Öncelikle projeyi talep eden kurum ile detaylı bir ön görüşme yapılmıştır. Bu aşamada projede olması istenilen özellikler genel hatları ile belirlenmiştir.

Genel olarak proje kapsamı hayvan yetiştiricileri birliğinde gerçekleştirilen üye, proje, destek takip işlemlerinin bilgisayar ortamında gerçekleştirilmesi ve kağıt üstünde yapılan işlemlerin dijital ortama aktarılmasını sağlama olarak belirlenmiştir. Bu sistemde bulunması gereken nitelikler proje tanımında belirtilmiştir.

1.2.PROJE KAYNAKLARININ BELİRLENMESİ

Bu aşamada aşağıda bulunan proje kaynaklarının tanımı yapılmıştır.

- ❖ İnsan kaynakları
- ❖ Yazılım kaynakları
- ❖ Donanım kaynakları

1.2.1.İNSAN KAYNAKLARI

Proje Yöneticisi	: Şevket DEMİRCİ
Yazılım Ekip Lideri	: Oğuzhan Kara
Proje Sekreteri	: Gül Ateş
Web Tasarımcısı	: Zeki Uras
Veri Tabanı Yöneticisi	: Hüma Ünlü
Sistem Çözümleyici	: Başak Eliz
Programcı	: Mahmut Erce
Sistem Tasarımcı	: Furkan Altın
Bilgisayar Mühendisi	: Şevket Demirci, Furkan Altın, Hüma Ünlü
Kalite Uzmanı	: İsmail Sancak
Eğitmen	: Tuncer Genç
Grafiker	: Abdullah Tosun

1.2.2 DONANIM KAYNAKLARI

- ❖ Web ve Database Sunucusu,
- ❖ Kişisel ve Diz Üstü Bilgisayarlar
- ❖ Projeksiyon
- ❖ Akıllı Tahta, Tv
- ❖ Renkli Yazıcı ve Tarayıcı
- ❖ İnternet alt yapısı

1.2.3.YAZILIM KAYNAKLARI

Geliştirilecek olan Hayvan Yetiştiricileri Birliği Otomasyon Sisteminde kullanılacak yazılım kaynakları şu şekilde olacaktır:

Proje Geliştirme Ortamı	: Zend Studio, XAMPP
Programlama Dili	: PHP
Veri Tabanı Yönetim Sistemi	: MySQL
Görsel Tasarım Araçları	: HTML, CSS, JQuery, CorelDraw
Bakım ve Destek Aracı	: Codeigniter, MVC Framework, Metronic Dashboard Template

1.3 PROJE MALİYETİNİN BELİRLENMESİ

Projenin süresi ve belirlenen sürede tamamlanması her proje için önemli olmakla beraber, proje ile ilgili maliyet de zaman kadar önemlidir. Projenin hedeflenen ve onaylanmış bütçe sınırları içinde tamamlanabilmesi için maliyetin belirlenmesi gerekir.

Proje maliyetinin belirlenmesinde COCOMO maliyet modeli temel alınmıştır. Bu aşamada yapılan çalışmalar proje için gerekli iş gücü ve zamanın belirlenmesi için yapılmış olan çalışmalardır. Proje maliyeti hesaplama;

Proje	Proje Sınıfı	Hesaplama Modeli
Hayvan Yetiştiricileri Birliği Üye Takip Otomasyonu	AYRIK PROJE	TEMEL MODEL

- ❖ Temel alınan modelin ilk aşaması proje sınıfının belirlenmesidir. Geliştirilecek olan Hayvan Yetiştiricileri Birliği Üye Takip Otomasyonu ayrık proje sınıfında yer almaktadır.
- ❖ COCOMO maliyet hesaplama yöntemi kullanılacak olan ayrıntı düzeyine göre üç farklı biçimde yapılabilmektedir. Yapılan maliyet çalışmalarında TEMEL Model baz alınmıştır.
- ❖ Belirlenmiş olan model ve sınıflandırmaya göre iş gücü ve zaman hesaplamalarında kullanılan formüller;

AYRIK PROJELER İÇİN:

- İş Gücü $K=2,4*S^{1,05}$ (S = bin türünden satır sayısı)
- Zaman $T=2.5*K^{0,38}$

COCOMO Modeli; girdi olarak temel satır sayısını alır, çıktı olarak iş-gücü ve zaman çıktıları verir. Satır sayısının hesaplanması için şu adımlar izlenmiştir;

S Katsayısının Hesaplanması

İlk olarak ayarlanmamış işlev noktası sayısı hesaplanır;

Ölçüm Parametresi	Sayı		Ağırlık Faktörü				
			Yalın	Ortalama	Karmaşık		
Kullanıcı Girdi Sayısı	35	X	3	4	6	=	210
Kullanıcı Çıktı Sayısı	21	X	4	5	7	=	105
Kullanıcı Sorgu Sayısı	16	X	3	4	6	=	96
Kütük Sayısı	9	X	7	10	15	=	90
Dışsal Arayüz Sayısı	2	X	5	7	10	=	14
Toplam Sayı						=	515

Buna göre AİN (Ayarlanmamış İşlev Nokta sayısı)= 515 olarak hesaplanmıştır

Projenin toplam karmaşık faktörünün hesaplanması;

TKF Soruları	Hiç etkisi yok(0)	Çok Az etkisi var(1)	Etkisi var(2)	Ortalama etkisi var(3)	Önemli etkisi var(4)	Mutlaka olmalı, kaçınılmaz (5)
Uygulama, güvenilir yedekleme ve kurtarma gerektiriyor mu?						X
Veri iletişimi gerekiyor mu?						X
Dağıtık işlem işlevleri var mı?				X		
Performans kritik mi?					X	
Sistem mevcut ve ağır yükü olan bir işletim ortamında mı çalışacak?		X				
Sistem, çevrim içi veri girişi gerektiriyor mu?						X
Ana kütükler çevrim-içi olarak mı güncelleniyor?						X
Girdiler, çıktılar, kütükler ya da sorgular karmaşık mı?					X	
İçsel işlemler karmaşık mı?				X		
Tasarlanacak kod, yeniden kullanılabilir mi olacak?					X	
Dönüştürme ve kurulum tasarımı dikkate alınacak mı?						X
Sistem birden çok yerde yerleşik farklı kurumlar için mi geliştiriliyor?	X					
Tasarlanan uygulama, kolay kullanılabilir ve kullanıcı tarafından kolayca değiştirilebilir mi olacak?					X	

TKF=48

➤ İşlev noktası sayısı (İN) = AİN * (0,65*0,01*TKF)

İN=515*(0,65*0,01*48)=160

Programlama Platformu	Satır Sayısı/İN(Ortalama)
Assembly Dili	300
COBOL	100
FORTTRAN	100
Pascal	90
C	90
Ada	70
Nesne Kökenli Diller	30
4. Kuşak Dilleri	20
Kod Üreticiler	15

İN değeri 160 bulunduktan sonra kullanılan dil grubuna göre yaklaşık satır sayısı hesaplanmıştır. Kullanılan programlama dili(PHP) nesne kökenli olduğu için yaklaşık satır sayısı kestirimi;

➤ Satır Sayısı = $160 \times 30 = 4820$

Satır sayısının bulunmasından sonra S sayısı formüllerde yerine konularak;

AYRIK PROJELER İÇİN:

➤ İş Gücü $K = 2,4 \times S^{1,05}$ (S = bin türünden satır sayısı)

➤ Zaman $T = 2,5 \times K^{0,38}$

İş Gücü $K = 2,4 \times 4,821,05 = \sim 12$ adam/ay

Zaman $T = 2,5 \times 12^{0,38} = \sim 6,5$ ay

Maliyet Analizi	
Projenin Toplam Süresi	26 hafta
Projenin Toplam Maliyeti (TL)	18000 TL
Toplam Satır Sayısı	4820
Bir satırın Parasal Maliyeti (TL)	3.75 TL

Projenin tamamlanma süresi yaklaşık 6,5 ay çıkmış olup, satır başına maliyetimizi 3,75 olarak hesaplarsak, projenin toplam maliyeti yaklaşık 18.000 TL olarak çıkacaktır.

1.4. PROJE EKİP YAPISININ OLUŞTURULMASI

1.4.1 YÜKLENİCİ PROJE EKİP YAPISI

- Proje Denetim Birimi: Şevket DEMİRCİ, Hüma Ünlü
- Proje Yönetim Birimi: Oğuzhan Kara, Şevket DEMİRCİ
- Kalite Yönetim Birimi: İsmail Sancak
- Proje Ofisi: Gül Ateş
- Teknik Destek Birimi: Adnan Şen, Mahmut Erce
- Yazılım Üretim Eşgüdüm Birimi: Mahmut Erce, Başak Eliz, Zeki Uras
- Eğitim Birimi: Tuncer Genç, Furkan Altın
- Uygulama Destek Birimi: Hüma Ünlü

1.4.2 İŞ SAHİBİ PROJE EKİP YAPISI

Proje geliştirme süresince eş güdümlü çalışılacak iş sahibi proje ekip yapısı şu şekildedir:

Birlik Başkanı: Mahmut Türk

Birlik Müdürü: Rüstem Çelebi

Birlik Büro Personeli: Fatma Aldaş

1.5. AYRINTILI PROJE PLANININ OLUŞTURULMASI

1.5.1 PROJE İŞ-ZAMAN PLANI

Proje iş-zaman planı aşağıdaki gibidir;

1. Planlama Aşaması (01.06.2019-20.06.2019)

Proje kapsamının belirlenmesi, proje kaynaklarının belirlenmesi, proje ekip yapısının oluşturulması, proje maliyetlerinin kestirilmesi, ayrıntılı proje planının yapılması.

2. Sistem çözümleme Aşaması (21.06.2019-31.08.2019)

Gerekli olan gereksinimlerin ayrıntılı belirlenmesi, mevcut sistemin çözümlenmesi, kullanıcı ara yüz modelinin oluşturulması.

3. Tasarım Aşaması (01.09.2019-25.09.2019)

Veri tasarımın, Yapısal tasarımın (genel tasarım, ayrıntı tasarım-süreç işlem tasarımı), kullanıcı ara yüz tasarımının oluşturulması, tasarım çalışmasının değerlendirilmesi, tasarım raporunun hazırlanması

4. Gerçekleştirim Aşaması (26.09.2019-30.09.2019)

Yazılım geliştirme ortamlarının belirlenmesi (Programlama dilleri, veri tabanı yönetim sistemleri, hazır program kitapları, Case araçları ve ortamları) , kod stiline belirlenmesi, yazılım kodlamasının gerçekleştirilmesi, olağan dışı durumların çözümlenmesi, yazılan kodun gözden geçirilmesi.

5. Yazılım Doğrulama ve Gerçekleme Aşaması (01.10.2019-20.10.2019)

Birim sına, alt sistem sına, sistem sına, kabul sına işlemlerinin yapılması ve sonuç raporlarının oluşturulması.

6. Kurulum-Eğitim-Bakım Aşaması (21.10.2019-15.11.2019)

Sistem yazılımlarının kurulması, veri tabanının kurulması, eğitim verilmesi, yerinde destek organizasyonu çalışmaları, düzeltici bakımın yapılması, uyarlayıcı bakımın yapılması.

1.5.2 KAYNAK YÖNETİM PLANI

Fiziksel ve insan kaynaklarının nasıl tahmin edileceğinin, edinileceğinin, yönetileceğinin ve kullanılacağına tanımlandığı süreçtir.

Projede çalışan ekip proje insan kaynaklarını oluşturmaktadır. Bu ekipte bir sekreter, beş bilgisayar mühendisi, bir programcı, bir çözümleyici, üç teknisyen ve bir grafikçi olmak üzere toplam 12 kişi çalıştırılmaktadır. Bu çalışanlar farklı görevlerde de yer almaktadırlar.

Çalışma günleri haftanın 6 günüdür. Çalışma saatleri esnektir. Kişi gündüz tamamlayamadığı işini akşam evinden de halledebilir. Önemli olan projede görev alan kişinin işi zamanında bitirmesidir. İhtiyaç duyulacak bütün fiziksel kaynaklar ofis ortamında kişiye sunulmuştur. İsteddiği zaman, istediği şekilde kullanma olanağı vardır.

1.5.3 SINAMA PLANI

Geliştirilecek olan projenin, proje talebinde bulunan kurum istekleri ve sistemin gereksinimlerini karşılayıp karşılamadığının belirlenmesi, sistemle olan uyumunun denetlenmesi, işlevlerini doğru yerine getirip getiremediğinin kontrol edilmesi için yazılım geliştirme sürecince sınamalar gerçekleştirilecektir. Bu sınamalar yazım talebinde bulunan kurumun, proje geliştirici ekip ile görüşmeleriyle sağlanacaktır.

1.5.4 EĞİTİM PLANI

Bu aşamada sistemi kullanacak olan kurum elemanlarına on günlük uygulama yazılımının kullanılabilirliğini artırma eğitimi verilecektir. Bu sistemde kullanıcılar; yöneticiler ve büro personelleri olacaktır.

Eğitimin içeriğinde üye işlemleri, raporlama işlemleri, şablon oluşturma ve yönetme işlemleri, pozisyon yetkilendirme işlemleri, hatırlatma ve profil oluşturma işlemleri, proje oluşturma işlemleri ve toplu sms konularında eğitimler verilecektir.

Eğitim içeriğinde yer almayan konularda eğitim almak isteyen kurum personeli olursa ücret karşılığında ek eğitimlerde sağlanacaktır.

1.5.5 BAKIM PLANI

Bakım ve güncelleme ile ilgili detaylar satın alma sözleşmesinde belirtilecektir. Buna göre gerçekleştirilen projenin bakım ve güncelleme süreci kurulumdan hemen sonra başlar ve iki yıl devam eder. Bu süre içerisinde meydana gelen hatalar ücretsiz düzeltilir. Talep edilen güncelleme istekleri ücretsiz şekilde karşılanır. Destek işlemleri uzaktan yapılabileceği gibi duruma göre yerinde de verilebilir. Kurum talepte bulunduktan sonra 24 saat içinde müdahale edilmelidir. İki yılın sonunda bakım ve destek paketi satın alınması şartıyla hizmet vermeye devam edilir.

2.SİSTEM ÇÖZÜMLEME

2.1 GİRİŞ

Çözümleme aşamasında kurumda ki mevcut işleyişin nasıl çalıştığı araştırılmış olup kurumun ihtiyaçları saptanmıştır. Bu işlem ardından sistem için uygun modelleme yapılmıştır.

2.2 GEREKSİNİMLERİN BELİRLENMESİ

Sistemin gereksinim duyduğu etmenlerin belirlenmesi için yapılan çalışmaları olmaktadır. Bu çalışmada gereksinimler çeşitli başlıklar altında incelenerek belirlenmiştir.

2.2.1 FİZİKSEL ÇEVRE

- Proje, uygulama ve veri tabanı sunucusu üzerinde çalışacaktır. Sunucu hizmeti hosting firmalarından sağlanacaktır. Dolayısıyla sunucu, sistemin kullanılacağı yerde bulunmayacaktır.
- Sisteme internet aracılığı ile uzaktan erişilecektir.
- Sistem elemanları herhangi bir fiziksel etmenden etkilenmemektedir. Sıcaklık, nem ve manyetik etkiler sistemi etkilemez.

2.2.2 ARAYÜZLER

- Kullanıcı ara yüzlerinde tanımlanan işlemler sisteme giriş yapan kullanıcıya göre değişecektir.
- Ara yüzler kullanıcılarının sistemi kolay kullanabilmesi için basit ve anlaşılır tasarlanacaktır.
- Dashboard ekranı bulunacaktır. Burada özet bilgiler yer alacaktır.
- Temel işlemler üye yönetimi, rapor yönetimi, form ve şablonlar, proje yönetimi, personel yönetimi, pozisyonlar, hatırlatmalar, toplu sms gönderme ve fatura işlemlerinden oluşmaktadır.

2.2.3 KULLANICI VE İNSAN ETMENİ

- Sistemi kullanacak olan kişiler iki tip kullanıcılardır. Bunlar yönetici ve büro personelidir.
- Kullanıcılar, pozisyonlar da verilen yetkiler dâhilinde sistemi kullanabilecektir.
- Sistem kullanıcıları temel bilgisayar bilgisi olan kullanıcılardır.
- Sistemi kullanacak olan kullanıcılar için tek bir eğitim söz konusudur, eğitimler yerinde uygulamalı yapılacaktır.

2.2.4 İŞLEVSELLİK

- Sistem proje tanımında belirtilen özelliklere sahip bir yapıya sahip olarak gerçekleştirilecektir.
- Sistem istenildiği zaman kullanılabilir. Hosting firması sunucu hizmeti verdiği sürece ve internet kesintisi olmadığı sürece sistem kullanılabilir durumdadır.
- Sistemin çalışmasında kesintiler yaşanmaması için uygun hosting firması seçilecektir.
- Raporlar Excel ortamına aktarılabilir.

2.2.5 BELGELEME

- Sistem kullanıcıları için takıldıklarında yardım alabilecekleri bir yönerge oluşturulacaktır.
- Aynı zamanda sistemin satın alınması sürecinde, satın alma bakım sözleşmesi ve teknik şartname oluşturulacaktır.

2.2.6 VERİ

- Sisteme girişi yapılan verilerin formatı rakam, karakter ve tarih saat olarak belirlenmiştir.
- Sistem, kullanımı sırasında sürekli olarak veri kaydetme veya veri görüntüleme işlemleri yapar.
- Veriler Excel ortamına aktarılabilir.
- Sistemde kayıtlı olan veriler üyelik iptaline kadar tutulur.

2.2.7 KAYNAKLAR

- Sistemi kurmak için ihtiyaç duyulan malzemeler: Sunucu, bilgisayar, yazıcı, tarayıcı, projeksiyon ve internet hizmeti.
- Sistemin eğitimini ve bakım hizmetini vermek için dört personel yeterlidir.
- Sistem uygulama ve veri tabanı ile birlikte başlangıçta 100GB alan kaplayacaktır.
- Sistemin geliştirme süresi ve hangi aşamanın ne kadar zaman alacağı proje zaman planlamasında gösterilmiştir.
- Projenin maliyeti ile ilgili hesaplamalar planlama aşamasında yapılmıştır.
- Güç, ısıtma ve soğutma kısıtları yoktur.

2.2.8 GÜVENLİK

- Kullanıcı yetkileri, pozisyon ayarlarında yapılarak her kullanıcı her yetkiye sahip olmayacaktır.
- Sistem yedeklemesi otomatik olarak günlük yapılacaktır.
- Yedek kopyalar sistemden ayrı bir yerde saklanacaktır.
- Bulut üzerinden hizmet alacağımız için yangın ve hırsızlık riski olmayacaktır.

2.2.9 KALİTE GÜVENCESİ

- Sunucu hizmeti alınan firmanın servis dışı kalmasından dolayı erişim sorunları yaşanabilir.
- Sistem bakımı sadece hata düzeltilmesinden değil ek olarak sistem geliştirmeyi ve yükseltmeyi de kapsayacaktır.
- Sistem internet üzerinde çalışacağı için internet bağlantısının yapılabildiği veya internet bağlantısı olan herhangi bir yere kolaylıkla taşınabilir.

2.3 SİSTEM ÇÖZÜMLEME ÇALIŞMASI

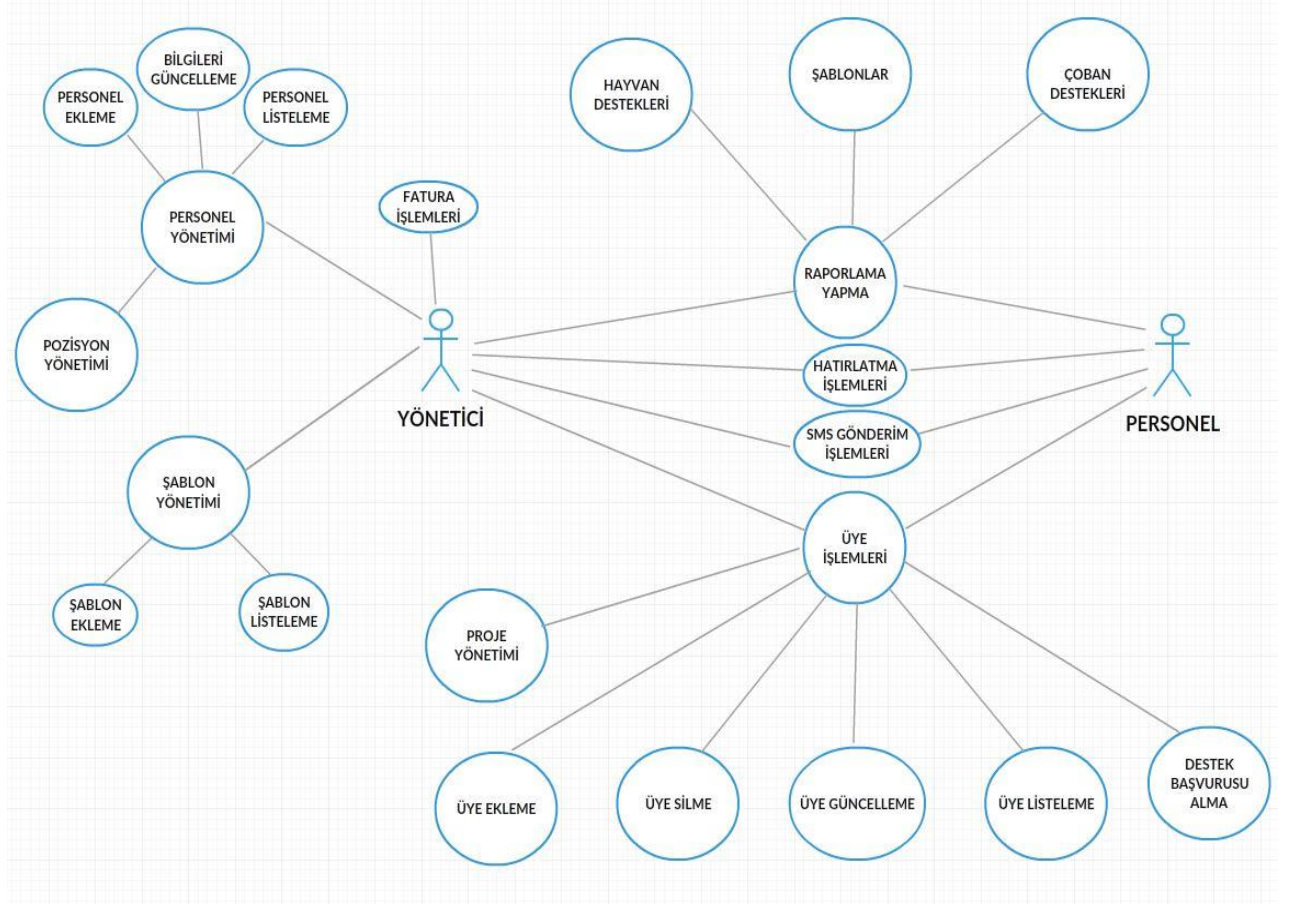
2.3.1 MEVCUT SİSTEMİN İNCELENMESİ ve DEĞERLENDİRİLMESİ

Yazılımı gerçekleştirilecek olan sistem, hayvan yetiştiricileri birliğinin işleyişini ve çalışmasını modelleyecektir. Bu sistemin işleyişi ve sistemin nasıl olması gerektiği hakkında bilgi edinildi. Gerekli görüşmeler ve incelemelerin ardından sistemin tüm işleyişinin kâğıt üzerinde yapılıyor olduğu ve verilerin sağlıklı bir ortamda tutulmadığı ve verilere erişimde sıkıntılar yaşandığı görülmüştür. Ayrıca çeşitli başvuruların takibinin yapılamadığı, üyelere iletişimde sorunlar olduğu görülmüştür. Herhangi bir aksaklığa karşı bilginin kurtarılamaması ve verilerin raporlanamaması da sistemin dijital ortama aktarılması gerçeğini ön plana çıkarmıştır. Sistem çok sayıda kullanıcının aynı anda sisteme girişinden en az şekilde etkilenecek ve web ortamının sunduğu en üst hız sınırında ve performansında olacak şekilde tasarlanacaktır. Raporlama ve takip işlemleri ayrıntılı şekilde hazırlanacaktır.

2.4 TASARLANAN SİSTEMİN MANTIKSAL MODELİ

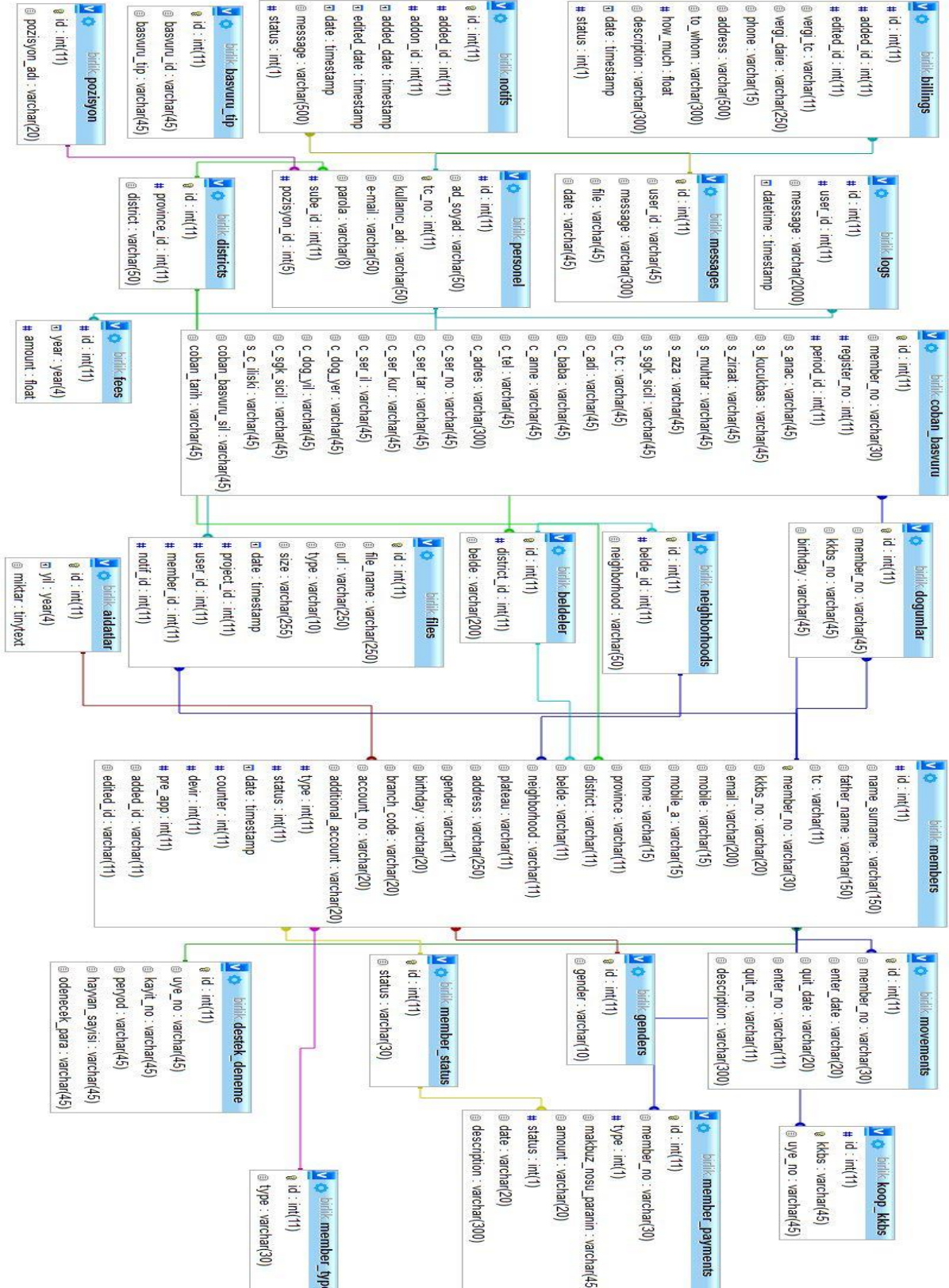
Sistemde kullanılacak olan verilerin kaydedilmesi, geri alınması ve değiştirilmesi sırasında veri tabanı kullanılacaktır. Bu veri tabanını yönetmek için ise MySQL veri tabanı yönetim sistemi kullanılacaktır. Aşağıda bu veri tabanı ile ilgili gerekli tasarımlar yapılmıştır.

2.4.1 USE - CASE DİYAGRAMLARI



2.4.2 VERİ MODELİ

2.4.2.1 NESNE- İLİŞKİ ŞEMASI



2.4.2.2 VERİ SÖZLÜĞÜ

1. MEMBER(Üye) TABLOSU

Primary Key: tc, member_no

Foreign Key: grup_no, district, belde, gender, type, status, added id, edited id

Nitelik	Tür	Açıklama
id	int(11)	Üyelere verilen otomatik id numarasıdır.
name_surname	varchar(150)	Üyelere ait adı soyad bilgisi
father_name	varchar(150)	Üyelere ait baba adı bilgisi
tc	varchar(11)	Üyeyi eşsiz belirtecek olan alandır. Bu alan üyenin tc kimlik numarasıdır.
member_no	varchar(30)	Üyelik numarasıdır
kkbs_no	varchar(20)	Üyeye ait kkbs numarasıdır
email	varchar(200)	Üyenin e-posta adres bilgisi
mobile	varchar(15)	Üyenin cep telefonu bilgisi
mobile_a	varchar(15)	Üyeye ait ikinci cep telefonu bilgisi
home	varchar(15)	Üyeye ait ev telefonu bilgisi
province	varchar(11)	Üyenin il ikamet bilgisi
district	varchar(11)	Üyenin ilçe ikamet bilgisi
belde	varchar(11)	Üyenin belde ikamet bilgisi
neighborhood	varchar(11)	Üyenin mahalle ikamet bilgisi
address	varchar(250)	Üyenin adres bilgisi
gender	varchar(1)	Üyenin cinsiyeti
birthday	varchar(20)	Üyenin doğum tarihi
branch_code	varchar(20)	Banka şube kodu
account_no	varchar(20)	Banka hesap kodu
additional_account	varchar(20)	Ek hesap kodu
type	int(11)	Üyelik şekli bilgisi
status	int(11)	Üyelik durum bilgisi
description	text	Açıklamalar bilgisi
date	timestamp	Ekleme tarihi
devir	int(11)	Devir bilgisi
added_id	varchar(11)	Üye kaydını yapan personel id
edited_id	varchar(11)	Üye kaydını düzenleyen personel id

2. MEMBER STATUS(Üye Durum) TABLOSU

Primary Key: id

Nitelik	Tür	Açıklama
id	int(11)	Durum bilgisini eşsiz belirtecek numaradır
status	varchar(30)	Üye durum bilgisi

3. MEMBER TYPES(Üye Tipi) TABLOSU

Primary Key: id

Nitelik	Tür	Açıklama
id	int(11)	Tip bilgisini eşsiz belirtecek numaradır
type	varchar(30)	Üye tipi bilgisi

4. MEMBER PAYMENT(Üye Ödemeleri) TABLOSU

Primary Key:id

Foreign Key:type,status,member_no

Nitelik	Tür	Açıklama
id	int(11)	Ödemeler bilgisini eşsiz belirtecek id no
member_no	varchar(30)	Üye no
type	int(1)	Üye tipi bilgisi
makbuz_nosu_paranin	varchar(45)	Makbuz numarası
amount	varchar(20)	Ödeme miktar bilgisi
status	int(1)	Üye durum bilgisi
date	varchar(20)	Ödeme tarih bilgisi
description	varchar(300)	Açıklamalar

5. AİDATLAR TABLOSU

Primary Key:id

Nitelik	Tür	Açıklama
id	int(11)	Aidatlar bilgisini eşsiz belirtecek id no
yil	year(4)	Aidatın ait olduğu yıl
miktar	tinytext	Aidat miktarı

6. BAŞVURU TİPİ TABLOSU

Primary Key:başvuru_id

Nitelik	Tür	Açıklama
id	int(11)	Başvuru sıra eşsiz belirtecek numara no
basvuru_id	varchar(45)	Başvuru id bilgisini
basvuru_tip	varchar(45)	Başvuru tip bilgisi

7. BELDELER TABLOSU

Primary Key:id

Foreign Key:district_id

Nitelik	Tür	Açıklama
id	int(11)	Belde bilgisini eşsiz belirtecek id no
district_id	int(11)	İlçe id bilgisi
belde	varchar(200)	Belde ad bilgisi

8. BİLLİNGS (Faturalar) TABLOSU

Primary Key:id

Foreign Key:added_id, edited_id

Nitelik	Tür	Açıklama
id	int(11)	Fatura bilgisini eşsiz belirtecek id no
added_id	int(11)	Fatura ekleyen personel id no
edited_id	int(11)	Fatura düzenleyen personel id no
vergi_tc	varchar(11)	Fatura için tc kimlik numara bilgisi
vergi_daire	varchar(250)	Fatura için vergi daire bilgisi
phone	varchar(15)	Fatura için telefon bilgisi
address	varchar(500)	Fatura için adres bilgisi
to_whom	varchar(300)	Faturanın kime (kurum-kışı)kesileceği bilgisi
how_much	float	Fatura miktarı bilgisi
description	varchar(300)	Açıklamalar
date	timestamp	İşlem tarihi
status	int(1)	Fatura durum bilgisi

9. ÇOBAN BAŞVURU TABLOSU

Primary Key: id

Foreign Key:coban_basvuru_sil,member_no

Nitelik	Tür	Açıklama
id	int(11)	Destek başvurusunu eşsiz belirtecek kayıt numarası
member_no	varchar(30)	Çoban desteğine başvuran üye numarası
register_no	int(11)	Çoban desteği başvurusunu
s_anac	varchar(45)	Anaç koyun keçi sayısı
s_kucukbas	varchar(45)	Toplam küçükbaş hayvan sayısı
s_ziraat	varchar(45)	Banka iban numara bilgisi
s_muhtar	varchar(45)	İkamet ettiği yerin muhtar bilgisi
s_aza	varchar(45)	İkamet ettiği yerin aza bilgisi
s_sgk_sicil	varchar(45)	Başvuran sgk sicil
c_tc	varchar(45)	Çoban tc kimlik numara bilgisi

c_adi	varchar(45)	Çoban Adı Soyadı
c_baba	varchar(45)	Çoban baba adı bilgisi
c_anne	varchar(45)	Çoban anne adı bilgisi
c_tel	varchar(45)	Çoban cep telefon bilgisi
c_adres	varchar(300)	Çoban adresi
c_ser_no	varchar(45)	Çobanın sertifika numara bilgisi
c_ser_tar	varchar(45)	Çobanın sertifika tarih bilgisi
c_ser_kur	varchar(45)	Çoban sertifikasının verildiği kurum bilgisi
c_ser_il	varchar(45)	Çoban sertifikasının verildiği il bilgisi
c_dog_yer	varchar(45)	Çobanın doğum yeri
c_dog_yil	varchar(45)	Çobanın doğum tarihi
c_sggk_sicil	varchar(45)	Çoban sgk sicil numara bilgisi
s_c_iliski	varchar(45)	Başuran ve çoban ilişki durum bilgisi
coban_basvuru_sil	varchar(45)	Kayıt yapan veya silen personel bilgisi
coban_tarih	varchar(45)	Çoban desteği başvuru tarihi

10.Districts(İlçeler) TABLOSU

Primary Key:id

Nitelik	Tür	Açıklama
id	int(11)	İlçe bilgisini eşsiz belirtecek id no
province_id	int(11)	İlçenin bağlı olduğu il bilgisi
district	varchar(50)	İlçe adı bilgisi

11.DOĞUMLAR TABLOSU

Primary :id

Foreign key:member_no,kkbs_no

Nitelik	Tür	Açıklama
id	int(11)	İd no bilgisi.Eşsiz belirtecek id no
member_no	varchar(45)	Üyelik numara bilgisi.
kkbs_no	varchar(45)	Üye kkbs numara bilgisi
birthday	varchar(45)	Üye doğum tarihi bilgisi

12.FEES(Harçlar) TABLOSU

Primary Key:id

Nitelik	Tür	Açıklama
id	int(11)	Harcın eşsiz olduğunu belirtecek numara bilgisi
year	year(4)	Harcın ait olduğu yıl
amount	float	Harcın miktarı

13.GENDERS(Cinsiyet) TABLOSU

Primary Key:id

Nitelik	Tür	Açıklama
id	int(11)	Cinsiyet bilgisinin eşsiz olarak tutulacağı numara bilgisi
gender	varchar(10)	Cinsiyet bilgisi

14.FİLES(Dosyalar) TABLOSU

Primary Key:id

Foreign key:user_id,member_id

Nitelik	Tür	Açıklama
id	int(11)	Dosya bilgisinin eşsiz olarak tutulacağı numara bilgisi
file_name	varchar(250)	Dosya adı
url	varchar(250)	Dosya bağlantı bilgisi
type	varchar(10)	Dosya tipi bilgisi
size	varchar(255)	Dosya boyut bilgisi
date	timestamp	Dosya yüklenme tarihi
project_id	int(11)	Proje id numarası
user_id	int(11)	İşlemi yapan personel id numarası
member_id	int(11)	Adına dosya düzenlenen üye id numarası
notif_id	int(11)	Bildirim id numarası

15.KOOP_KKBS TABLOSU

Primary Key:kkbs

Foreign key:uye_no

Nitelik	Tür	Açıklama
id	int(11)	Koyun Keçi Birliği Sistemi Kooperatif id
kkbs	varchar(45)	Koyun Keçi Birliği Sistemi Kooperatifine üye numara bilgisi. Eşsiz bilgiyi tutar.
uye_no	varchar(45)	Üye numara bilgisi

16.DESTEK BAŞVURU TABLOSU

Primary Key:kayit_no

Foreign key:uye_no

Nitelik	Tür	Açıklama
id	int(11)	Destek başvuru id no
uye_no	varchar(45)	Desteğe başvuru yapan üye id numarası
kayit_no	varchar(45)	Destek başvuru kayıt numarası. Eşsiz bilgiyi tutar.
periyod	varchar(45)	Destek periyod bilgisi
hayvan_sayisi	varchar(45)	Destek için belirtilen küçükbaş hayvan sayısı
odenecek_para	varchar(45)	Ödeme miktar bilgisi

17.MOVEMENTS (Hareketler) TABLOSU

Primary Key:id

Foreign key:member_no

Nitelik	Tür	Açıklama
id	int(11)	Hareketler için eşsiz bilgiyi tutacak id numara bilgisi
member_no	varchar(30)	Hareketi olan üye numara bilgisi
enter_date	varchar(20)	Giriş tarihi
quit_date	varchar(20)	Çıkış Tarihi
enter_no	varchar(11)	Giriş numara bilgisi
quit_no	varchar(11)	Çıkış numara bilgisi
description	varchar(300)	Açıklamalar alanı

18. NEIGHBORHOODS(Mahalleler) TABLOSU

Primary Key:id

Foreign key:belde_id

Nitelik	Tür	Açıklama
id	int(11)	Mahalle bilgisini eşsiz tutacak id numarası
belde_id	int(11)	Mahallenin bağlı olduğu belde id numarası
neighborhood	varchar(50)	Mahalle adı

19.LOGS (Loglar) TABLOSU

Primary Key:id
Foreign key:user_id

Nitelik	Tür	Açıklama
id	int(11)	Logları tutacak eşsiz id numara bilgisi
user_id	int(11)	İşlem yapan personel id bilgisi
message	varchar(2000)	Yapılan işlem ile ilgili detay açıklama bilgisi
datetime	timestamp	İşlem yapılan tarih

20.MESSAGES (Mesajlar) TABLOSU

Primary Key:id
Foreign key:member_no

Nitelik	Tür	Açıklama
id	int(11)	Hatırlatmaları eşsiz olarak tutacak id numara bilgisi
user_id	varchar(45)	Hatırlatma yapılacak personel id
message	varchar(300)	Hatırlatma mesaj içeriği
file	varchar(45)	Hatırlatma dosya bilgisi
date	varchar(45)	Hatırlatma tarihi

21.PERSONEL TABLOSU

Primary Key: tc
Foreign key:şube_id, pozisyon

Nitelik	Tür	Açıklama
id	int(11)	Personeli id numara bilgisi
ad_soyad	varchar(50)	Personel Ad Soyadı Bilgisi
tc_no	varchar(11)	Personel kimlik numarası eşsiz olarak belirtecek
kullanıcı_adı	varchar(50)	Personel ait takma ad bilgisi
e-mail	varchar(50)	Personel e-posta adresi
parola	varchar(8)	Personel parola bilgisi
sube_id	int(11)	Personelin bağlı olduğu şube (ilçe) bilgisi
pozisyon_id	int(1)	Personelin kullanıcımı yoksa yöneticimi olduğu bilgisi

22.POZİSYON TABLOSU

Primary Key:id

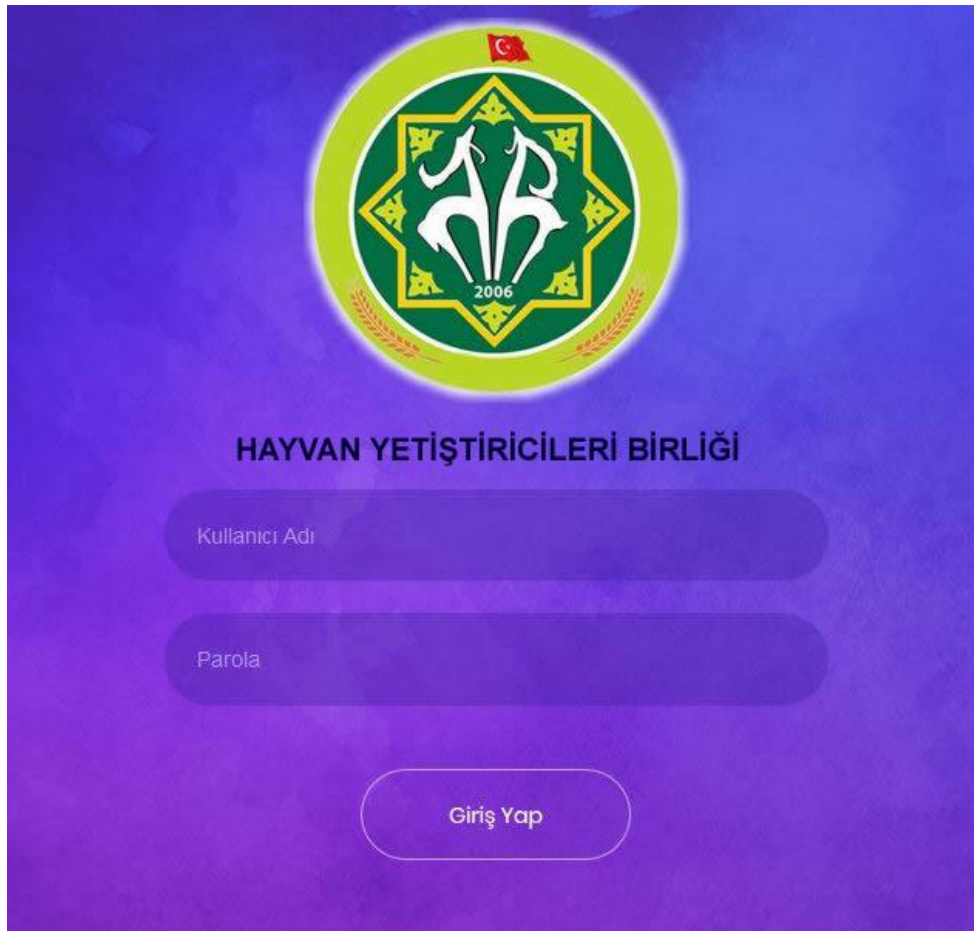
Nitelik	Tür	Açıklama
id	int(11)	Pozisyon bilgisini eşsiz tutan id numara bilgisi
pozisyon_adı	Varchar(20)	Pozisyon ad bilgisi

2.4.3 KULLANICI ARAYÜZ TASARIMI

Sistem geliştirmeye başlanmadan önce geliştirilecek olan sistem ile ilgili kullanıcı arayüz tasarımları yapılarak kullanıcıdan onay alınmıştır. İlgili arayüzler aşağıda görülmektedir.

2.4.3.1 OTOMASYON SİSTEMİNE GİRİŞ EKRANI

Hayvan yetiştiricileri birliği otomasyon sistemine giriş paneli aşağıdaki gibidir. Sisteme web üzerinden belirlenen domain ile giriş yapılacaktır. Sisteme kayıtlı kullanıcılar belirlenen kullanıcı adı ve parolaları ile belirlenen yetki düzeyinde erişim sağlayacaklardır.

The image shows a login interface for the Hayvan Yetiştiricileri Birliği (Animal Breeders' Association). At the top center is a circular logo with a green border, a yellow star, and a white silhouette of a horse and rider. Below the logo, the text "HAYVAN YETİŞTİRİCİLERİ BİRLİĞİ" is displayed in white. Underneath, there are two input fields: "Kullanıcı Adı" (Username) and "Parola" (Password). At the bottom center is a button labeled "Giriş Yap" (Login).

2.4.3.2 PERSONEL GİRİŞ ARAYÜZÜ(Dashboard)

Sisteme giriş yapan personel aşağıda bulunan ara yüz ile karşılaşır. Bu ara yüzde sistemde kayıtlı üyeler, projeler, hatırlatmalar, faturaların son durumu ile ilgili bilgiler sayısal olarak görülmektedir. Ayrıca üye yönetimi, rapor yönetimi, form-şablon yönetimi, proje yönetimi, personel yönetimi, toplu sms gönderimi, hızlı erişim menüleri, arama çubuğu ile ilgili menüler yer almaktadır. Yine bu ekranda bilgilendirme amaçlı olarak son kaydı yapılan üyeler, hatırlatmalar, tarih, saat ve kullanıcı bilgileri yer almaktadır.

The screenshot displays the D.K.K.Y.B. dashboard interface. The sidebar on the left contains navigation links: Üyeler (502), Projeler (0), Hatırlatmalar (0), Faturalar (0), Raporlar, Toplu SMS, and a YÖNETİM section with links for Üye Yönetimi, Rapor Yönetimi, Form Şablonlar, Proje Yönetimi, and Personel Yönetimi. The main content area is titled 'Son Üyeler' (Recent Members) and lists four members:

Member Name	ID	Location	Action
İbrahim ÇETİN	4880884	Merkez / Akhisar	Detay
Hacı Ahmet KOYUNCU	1248805	Eskil / Merkez	Detay
Erkan BORA	284954	Merkez / Dikmen	Detay
Ahmet ÜNLÜSOY	479088	Gülağaç / Bekarlar	Detay

To the right, the 'Hatırlatmalarım' (My Reminders) section shows 'Hiç hatırlatma yok.' (No reminders).

2.4.3.3 ÜYE YÖNETİMİ->ÜYE EKLE ARAYÜZÜ

Sisteme yeni üye kaydı yapılmak istenirse üye yönetimi altından üye ekle menüsü kullanılır. Burada üyeye ait birçok bilginin girilmesi sağlanır. Kişisel bilgisinden iletişim bilgisine kadar bütün ayrıntılar sisteme kaydedilebilir. Burada birlik üye numarası otomatik olarak sistem tarafından verilir. Sisteme üye eklendiği zaman bu üye ilk adımda aday üyedir. Kişiye ait evraklar kurul tarafından incelendikten sonra kişinin üyeliği asıl üyeliğe çekilir.

Yeni Önbaşvuru Ekle

Adı Soyadı

Adı Soyadı

Baba Adı

Baba Adı

T.C. Kimlik No / Vergi Kimlik No

T.C. Kimlik No ya da Vergi Kimlik No

Birlik Üye No

TR685065

KKBS No

Koyun Keçi Bilgi Sistemi No

E-Posta

E-Posta Adresi

Cep Telefonu

Cep Telefonu

Ek Cep Telefonu

Cep Telefonu

Adres

Adres

Cinsiyet

Bay

Doğum Tarihi

gg/aa/yyyy

Şube Kodu

Banka Şube Kodu

Hesap No

Hesap No

Ek Hesap No

Ek Hesap No

Üyelik Şekli

Asıl

Üyelik Durumu

Aktif

Açıklama

Üye hakkında açıklama

Kaydet

Üye Listesine Dön

2.4.3.4 ÜYE YÖNETİMİ->ÜYE LİSTELE ARAYÜZÜ

Sistemde kayıtlı üyeleri görebilmek için üye yönetimi altından üye listele sekmesi kullanılır. Burada kayıtlı üyelere hızlıca erişebilmek için çeşitli filtreleme yolları kullanılmıştır. Örneğin durumu aktif olanlar, asıl ya da aday üye olanlar gibi. Ayrıca bulunduğu adrese göre de filtreleme yapılabilir. Ya da arama çubuğuna kişiye ait herhangi bir bilgi yazılarak da arama yapılabilir. Sonucu bulunan adaya ait satırın sonunda detay görüntüleme butonu ile adayla ilgili her türlü detay bilgiye erişilebilir. Örneğin destek durumu, yaptığı ödemeler, proje durumu gibi.

Üye Listesi [Ana Sayfa](#) [Exceli Aktar](#) [Yeni Ekle](#)

Üyeler

İlçe: Hepsil

Mahalle: Hepsil

Sokak: Hepsil

Tip: Hepsil

Durum: Hepsil

☐ Önbaşvuru

Üye No	Adı Soyadı	TC	Cep Telefonu	Baba Adı	Doğum Tarihi	
TR680001	Mahmut	34820	5496926	Celal		Detaylı Görüntüle
TR680002	Necati KAR	28895	5426288	Halil	04/29/1968	Detaylı Görüntüle
TR680003	Barış ÇEL	28487		Ali		Detaylı Görüntüle
TR680004	Yusuf ÖZ	679182	5414705	Mevlüt	01/07/1959	Detaylı Görüntüle
TR680005	Yaşar KAÇ	649753	546241	Osman	01/13/1973	Detaylı Görüntüle

2.4.3.5 ÜYE YÖNETİMİ->TOPLU ÖDEMELER ARAYÜZÜ

Üyelere ait hizmet bedeli, yıllık aidat, giriş aidatı gibi bazı ödemelerin toplu olarak girilmesini sağlayan bölümdür. Buraya üye yönetimi altından toplu ödemeler sekmesi ile erişebilirsiniz.

Toplu Ödeme

Türü: - Seçiniz

Durumu: - Seçiniz

Üyeler: - Seçiniz

Üyeler: - Seçiniz

Miktar:

Açıklama:

2.4.3.6 ÜYE YÖNETİMİ->TOPLU BAŞVURU ARAYÜZÜ

Üyelere ait başvuruların toplu olarak sisteme kaydedildiği bölümdür. Buraya Üye yönetimi menüsü altından erişebilirsiniz.

Toplu Başvuru Onaylama

Dosya Seçiniz Gözet... Hiçbir dosya seçilmedi.

Onayla İptal

2.4.3.7 RAPOR YÖNETİMİ->DESTEK BAŞVURUSU OLANLAR ARAYÜZÜ

Sistemde kayıt olan üyeler hayvan desteği alabilmek için başvuruda bulunur. Bakanlık birimleri tarafından incelenen ve uygun görülen üyelere hayvan desteği ödenir. Sonuç durumu üyenin bilgisine işlenir. İşte hangi üyenin destek aldığını görebilmek için raporlama yönetiminden destek başvurusu alanlar menüsü kullanılır. Burada destek alan üyelere hızlıca erişebilmek için çeşitli filtreleme yolları kullanılmıştır. Örneğin durumu aktif olanlar, asıl ya da aday üye olanlar ya da dönem bazlı gibi. Ayrıca bulunduğu adrese göre de filtreleme yapılabilir. Ya da arama çubuğuna kişiye ait herhangi bir bilgi yazılarak da arama yapılabilir. Sonuçları çıkan adaylara ait detaylı bilgi ekranda görünür. Örneğin kaç adet hayvan için destek aldığı, hangi dönemde aldığı gibi. Ayrıca burada ki bilgileri Excel ortamına aktararak çeşitli düzenlemeler yapılabilir.

Dest. Başv. Olanlar veriler seçimlerinize göre değişmektedir

Hazır Rapor Seç Excel'e Kaydet

İlçe: Hepsi

Mahalle: Hepsi

Sokak: Hepsi

Tip: Hepsi

Durum: Hepsi

Ara...

Dönem: Seçiniz

#	Üye No	Ad Soyad	İlçe	Mahalle	Sokak	Tip	Durum	Dest.Anaç.Topl.	Dönem
744	TR68C	Mahmut.	Merköz	Acıpınar		Asıl	Aktif	528	2018
1	TR68C	Mahmut.	Merköz	Acıpınar		Asıl	Aktif	265	2011

2.4.3.8 RAPOR YÖNETİMİ->DESTEK BAŞVURUSU OLMAYANLAR ARAYÜZÜ

Hayvan desteği alan üyeleri raporlama yapabildiğimiz gibi yine aynı şekilde destek başvurusu olmayan adayları da detaylı şekilde raporlayabilir ve Excel ortamına aktarabiliriz.

Dest. Başv. Olmayanlar veriler seçimlerinize göre değişmektedir

Hazır Rapor Seç

Excel'e Kaydet

İlçe: Hepsil

Mahalle: Hepsil

Sokak: Hepsil

Tip: Hepsil

Durum: Hepsil

Ara...

Dönem: 2019

#	Üye No	Ad Soyad	İlçe	Mahalle	Sokak	Tip	Durum
Kayıt bulunamadı							

2.4.3.9 RAPOR YÖNETİMİ->ÇOBAN DESTEĞİ OLANLAR ARAYÜZÜ

Devletin küçükbaş hayvan yetiştiricilerine sağladığı destek yollarından biride çoban desteğidir. Üyeler devletten çoban desteği alabilmek için birlik aracılığı ile başvuruda bulunur. Başvurusu onaylanan üyeler sisteme kaydedilir. Burada destek alan üyelere hızlıca erişebilmek için çeşitli filtreleme yolları kullanılmıştır. Örneğin durumu aktif olanlar ya da hangi dönem için destek aldığı gibi. Ayrıca bulunduğu adrese göre de filtreleme yapılabilir. Ya da arama çubuğuna kişiye ait herhangi bir bilgi yazılarak da arama yapılabilir. Sonuçları çıkan adaylara ait detaylı bilgi ekranda görünür. Ayrıca burada ki bilgileri Excel ortamına aktararak çeşitli düzenlemeler yapılabilir.

Çoban Desteği Alanlar veriler seçimlerinize göre değişmektedir

Hazır Rapor Seç

Excel'e Kaydet

İlçe: Hepsil

Mahalle: Hepsil

Sokak: Hepsil

Tip: Hepsil

Durum: Hepsil

Ara...

Dönem: Seçiniz

#	Üye No	Ad Soyad	İlçe	Mahalle	Sokak	Tip	Durum	Dönem
74	TR680	Mahmut	Merkez	Acıpınar		Asil	Aktif	2018
91	TR680	Ruhi	Merkez	Acıpınar		Asil	Aktif	2018
36	TR680	Yücel	Merkez	Baymış		Asil	Aktif	2018

2.4.3.10 FORM-ŞABLONLAR->YENİ ŞABLON EKLE ARAYÜZÜ

Sisteme yeni üye kaydı, proje ya da destek başvurusu, devir gibi işlemler sistemdeki şablonlar ile yapılmaktadır. Ayrıca personelin izin ve yazışma şablon örnekleri de burada mevcuttur. Burada şablondan yararlanarak kişiye ait bilgiler doldurulup kişinin ıslak imzası alınır ve sisteme yüklenir. Bu şekilde tüm kayıtlar elektronik ortama aktarılmış olur. İşte burada gerekli olan şablon hazırlanıp istenildiğinde kullanılmak üzere sisteme eklenir.

Form Şablon Ekle (Word)

Şablon Adı

Dosya Başlığını Giriniz

Word Seç (*.doc / *.docx)

Gözet...








Hiçbir dosya seçilmedi.

Kaydet

Şablon Listesine Dön

2.4.3.11 FORM-ŞABLONLAR->ŞABLON LİSTESİ ARAYÜZÜ

Her türlü şablon ve form örneklerini görebilmek, inceleyebilmek, ekleyebilmek veya silebilmek için bu menü kullanılır.

Şablonlar (Word)			Şablon Ekle
	Üye Devir Dilekçesi Oluşturucu: Abdulkadir	2019-03-13 14:50:24	<div>Gözet</div> <div>Sil</div>
	Hibe Başvuru Dilekçesi Oluşturucu: Abdulkadir	2019-03-13 14:20:51	<div>Gözet</div> <div>Sil</div>
	yazışma örnekleri Oluşturucu: Tekin	2018-09-21 16:41:05	<div>Gözet</div> <div>Sil</div>
	Ücretsiz izin belgesi Oluşturucu: Tekin	2018-09-21 16:38:58	<div>Gözet</div> <div>Sil</div>
	izin dilekçesi Oluşturucu: Tekin	2018-09-21 16:37:45	<div>Gözet</div> <div>Sil</div>
	Aday Üyelğe Geçiş Oluşturucu: Oğuzhan	2018-09-17 07:37:13	<div>Gözet</div> <div>Sil</div>
	Asıl Üyelğe Geçiş Oluşturucu: Oğuzhan	2018-09-17 07:36:53	<div>Gözet</div> <div>Sil</div>

2.4.3.12 PROJE YÖNETİMİ ->PROJE OLUŞTUR VE PROJE LİSTESİ ARAYÜZÜ

Proje oluşturmak ve mevcut projeleri görüntüleyebilmek için bu sekmeler kullanılır.

Yeni Proje

Başlık

Başlık giriniz

Kısaca




Kaydet

İptal

Projeler	Aktif	Biten	Çöp
Aktif proje yok.			

2.4.3.13 PERSONEL YÖNETİMİ ->PERSONEL LİSTESİ ARAYÜZÜ

Birlik sisteminde iki tür kullanıcı vardır. Birincisi admin yetkisine sahip yönetici ve normal kullanıcı yetkisine sahip personel kullanıcısıdır. Bu kullanıcılar pozisyonlarına göre yetkilendirilmektedir. Sisteme erişip kullanabilen personeller bu ara yüz ile listelenir. Personelin detay bilgilerine de detay butonu ile erişilir.

Kullanıcı Listesi		Ana Sayfa		Yeni Kullanıcı Ekle
Kullanıcılar		Tümü	Yönetici	Personel
	Abdulkadir abdulkadir	Administrator abdulkadir	@hotmail.com	Detay
	Berat berat	Personel berat	@hotmail.com	Detay
	Bircan bircan	Personel bircan	@gmail.com	Detay

2.4.3.14 PERSONEL YÖNETİMİ->PERSONEL EKLE ARAYÜZÜ

Sistemi kullanmak için yeni bir kullanıcı eklenecekse personel yönetimi menüsü altından personel ekle sekmesine tıklanır. Burada personelin kişisel bilgileri dışında şube ve pozisyon bilgisi önemlidir. Hangi ilçe de çalışacaksa şube durumu ona göre seçilmeli. O şubede yönetici konumunda mı çalışacak yoksa personel konumunda mı çalışacak yine duruma göre seçilmelidir.

2.4.3.15 PERSONEL YÖNETİMİ-> BİLGİLERİMİ GÜNCELLE ARAYÜZÜ

Personel listele menüsünde detay sekmesine girildiği anda da bu sayfa açılır. Burada kurum personelleri ile ilgili her türlü işlem yapılır. Kişisel bilgilerinden şifre bilgilerine, parola bilgilerinden sosyal medya bilgilerine kadar birçok alan güncellenebilir. Ayrıca kullanıcı silinip eklenebilir.

2.4.3.16 AYARLAR->POZİSYON YÖNETİMİ ARAYÜZÜ

Sistemi kullanacak olan personellere yönetici ya da personel bazında çeşitli yetkiler verilmektedir. Örneğin personeller üye ödeme silme işlemi yapamaz. İşte bu yetkiler ve pozisyonlar burada düzenlenir. Ayrıca mevcut pozisyonların haricinde farklı pozisyonlar eklenecek ise bu menü kullanılır.

Pozisyonlar

Pozisyon Ekle

Administrator	Düzenle	Sil
Personel	Düzenle	Sil

2.4.3.17 FATURALAR ARAYÜZÜ

Kurumun tüm faturaları da bu sistem üzerinden kayıt altına alınmaktadır. Bekleyen, ödenen, siline faturalar burada görüntülenir. Ayrıca ekle butonu ile faturalar sisteme aşağıdaki ekranda olduğu gibi eklenmektedir.

Fatura Listesi

BekleyenÖdenenTümüSilinenEkle

Bekleyen fatura yok.

Yeni Fatura Ekle

Kurum yada kişi

Örnek: Mepaş elektrik

Vergi no yada Tc

Vergi dairesi

Telefon

Örnek: 5554443322

Adres

Miktar

Örnek: 500.55

Açıklama

Örnek: Nisan elektrik faturası

Durum

Ödenecek

Kaydet

İptal

2.4.3.18 HATIRLATMALAR ARAYÜZÜ

Personelin iş takibini kolaylaştırmak amacıyla ve hatırlatmalar kısmı eklenmiştir. Burada kullanıcı kendine ya da bir başka personele hatırlatma ekleyebilmektedir. Hatırlatmalar sekmesine tıklandığında bekleyen, tamamlanan veya atanan hatırlatma mesajları görülebilmektedir. Ayrıca ekle menüsü ile istenilen personele istenilen tarihli ve içerikli hatırlatma mesajları ayarlanabilmektedir.

Hatırlatmalarım

BekleyenTamamlananAtadıklarım

Ekle

Bekleyen hatırlatma yok.

Yeni Hatırlatma Ekle

Hatırlatma Kime?

☐

 Oğuzhan

☐

 Fatma

☐

 Bircan

☐

 Tekin☐☐☐☐☐☐☐

Hatırlatma Zamanı

Tarih & Saat

Bir zaman seçin

Hatırlatma Mesajı

Kaydet

İptal

2.4.3.18 TOPLU SMS YÖNETİM ARAYÜZÜ

Sisteme kayıtlı üyeleri bilgilendirmek için sms gönderme modülü hazırlanmıştır. Burada ister tek kişiye ister toplu olarak sms gönderilebilir. Ayrıca sık gönderilen mesajlar şablon olarak kaydedilir ve istenildiğinde kullanılabilir. Toplu mesajların alıcı kısmını ilçe, mahalle, tip ya da kime gibi filtreler ile daraltabilirsiniz.

SMS Yönetimi

Anasayfa

Şablon Seç

Tek SMS Toplu SMS SMS Kanunu

Kime: Üyeler

İlçe: İlçe Seçiniz

Mahalle: Önce İlçe Seçiniz

Tip: ☐ Aktif ☐ Pasif

Mesaj:
Şablon Seç

12.04.2016 tarihli DK-YED/211 sayılı BTK kararına ile 11.12.2016 tarihinden itibaren başlıklı SMS gönderimlerinde mesaj metninin sonuna operatör kodunun (8021) eklenmesi zorunlu hale getirilmiştir. Sistem otomatik olarak B021 kodunu mesaj metninin sonuna ekleyecektir. 1 SMS karakter boyu başlıklı SMS için 155 olarak güncellenmiştir. Abone no ile gönderimlerde bu durum geçerli değildir.

Temizle Gönder

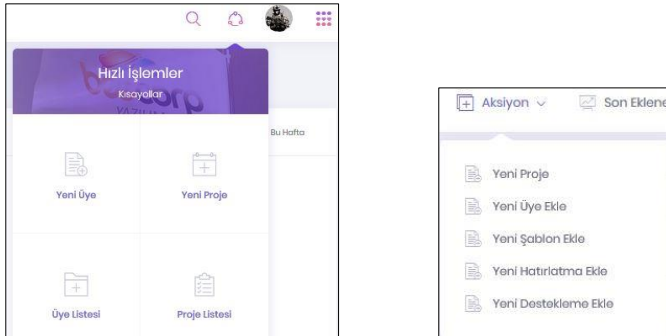
İŞLEMLER

SMS

Şablonlar

2.4.3.19 HIZLI İŞLEMLER ARAYÜZÜ

Çok sık kullanılan menülere erişebilmek için hızlı işlemler ve aksiyon gibi menüler hazırlanmış olup personele hız ve kolaylık sağlamaktadır.



2.4.3.20 ÜYE DETAY ARAYÜZÜ

Sisteme kayıtlı üyelerle ilgili her türlü iş ve işlemleri, güncellemeleri, başvuruları, hareketleri, devretme işlemlerini, üye ekleme ve silme işlemlerini ve üyeye ait bilgi görüntüleme işlemlerini üye detay ara yüzü altındaki sekmelerden gerçekleştirebiliriz. Üye çoban desteği başvurusu yapacaksa ya da hayvan desteği başvurusu yapacaksa bu menü altından işlemleri gerçekleştirilir.

Üye Detay

Üye Bilgileri

Üye Detay - İK üyeliği - 128 TL borcu var

Adı Soyadı: Mahmut

Baba Adı: Ali

T.C. Kimlik No / Vurgu Kimlik No: 12345678901234567890

Birlik Üye No: 123456

KİİS No: 123456

İlçe: Marmara

Mahalle / Köy: Açınar

Adres: marmara açınar

Cinsiyet: Bay

Doğum Tarihi:

Şube Kodu:

Hesap No:

Ek Hesap No:

2.4.3.21 KULLANICI DETAY ARAYÜZÜ

Burada kurum personelleri ile ilgili her türlü işlem yapılır. Kişisel bilgilerinden şifre bilgilerine, parola bilgilerinden sosyal medya bilgilerine kadar birçok alan güncellenebilir. Ayrıca kullanıcı silinip eklenebilir.

3.TASARIM

3.1 GİRİŞ

Bu aşamada sistem analizi çalışması sonucunda üretilen mantıksal modelin fiziksel modele dönüştürülmesi çalışmaları yapılmıştır.

3.2 TANIMLAR VE KISALTMALAR

YAZILIM: Bir bilgisayarı, belirli işlevleri yerine getirmek üzere yöneten, bilgisayara ne yapacağını söyleyen, kodlanmış komutlar dizisidir.

VERİ TABANI: Veri tabanı birbiriyle ilişkili veriler bütünüdür.

VERİ YÖNETİMİ: Verinin toplanması, verinin kaydı, verinin tekrar elde edilmesini yönetir.

MODÜLERLİK: Bir bilgisayar sisteminin veya programının birbirlerini en alt düzeyde etkileyen, ayrıca bağımsızca çalışabilen, diğer taraftan da bir bütünün parçası olan bölmelere ayrılabilmesi özelliği, yani parçalı sistemdir.

TABLO İLİŞKİ ŞEMASI: Tablolar arasında ki ilişkileri gösterildiği şemadır.

VERİ SÖZLÜĞÜ: Veri tabanı hakkındaki, kullanıcılar, nesneler, alan ayarlaması, ayrıcalıklar gibi meta dataların depolandığı yerdir.

ER DİYAGRAMI: İlişkisel veri tabanı sisteminde, tablolar arası ilişkileri göstermek için kullanılan diyagramdır.

3.3 GENEL TASARIM

Birlik sistemine web üzerinden bir domain aracılığı ile erişilecektir. Sisteme aynı anda birden fazla kişi erişebilir. Aynı anda işlem yapabilir. Sisteme giriş yapan kullanıcılar yönetici ya da personel olarak ayrılmıştır. Her türlü üye kayıt ekleme, silme güncelleme, başvuru ve devir etme gibi işlemler sistem üzerinden yapılabilmektedir. Bu işlemler ayrıntılı olarak proje tanımı ve proje niteliklerinde belirtilmiştir.

3.4 VARSAYIMLAR VE KISITLAMALAR

- Sistemde işlem yapacak personeller yetki düzeylerine göre pozisyonlara ayrılmıştır. İstenildiği takdirde bu pozisyonların sayısı artırılabilir. Şu an için yönetici ve personel olarak iki kullanıcı tanımı vardır.
- Sistem kullanıcıları şifrelerini bir başkasına vermemeli ya da kaptırmamalıdır. Aksi takdirde kendilerine verilmiş olan yetki düzeyinde işlemler gerçekleştirilebilir.
- Sistemde kayıtlı üyelerin aidat ya da hizmet bedeli ile ilgili borçları bulunuyorsa, o üyeye ait çoban ya da hayvan desteği başvurusu yapılamaz.

3.5 VERİ TASARIMI

Bu aşamada sitemde kullanılacak olan veri tabanı tablolarının tanımlamaları yapılmış ve tablolar arasındaki ilişkiler gösterilmiştir.

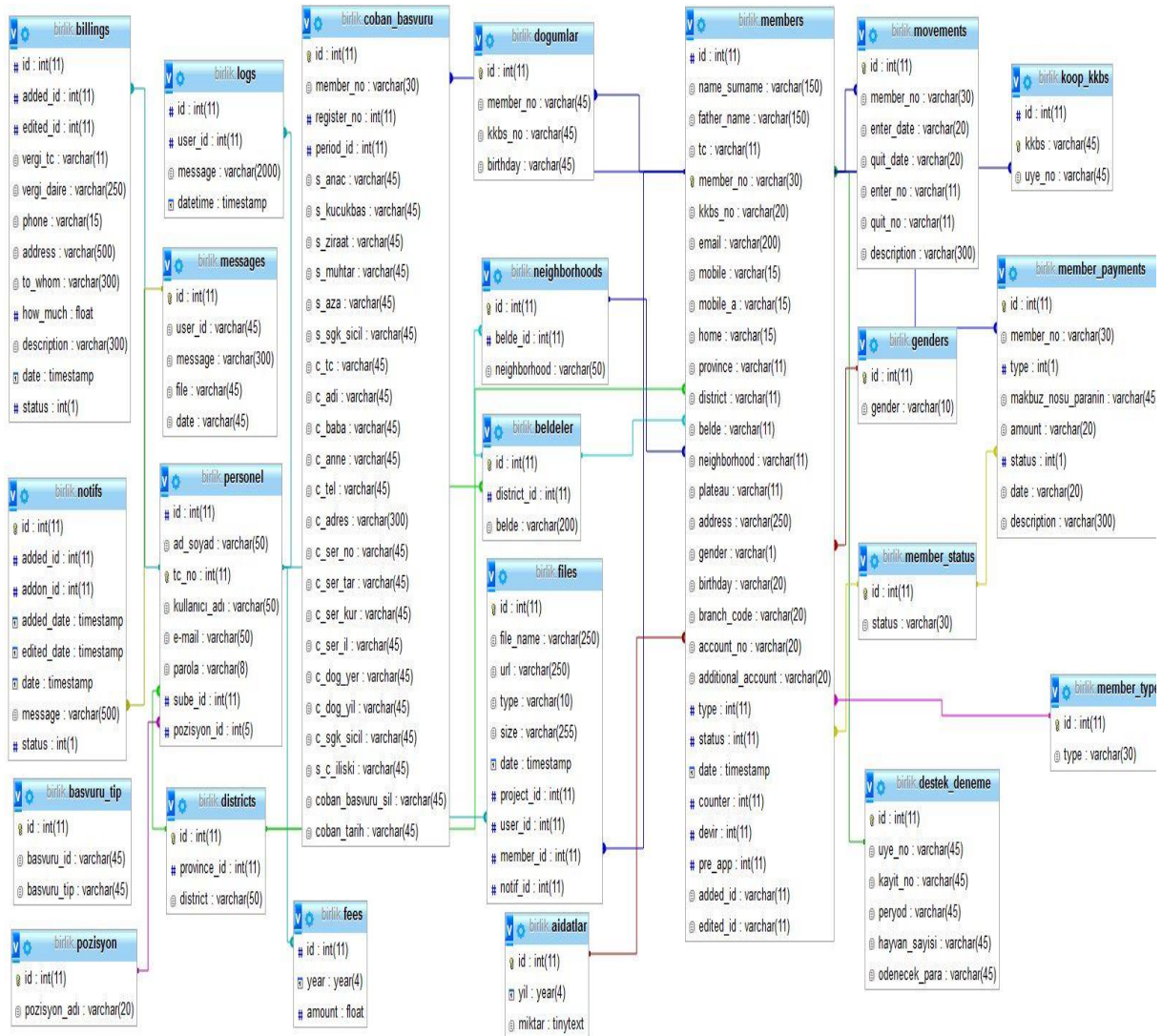
3.5.1 TABLO TANIMLARI

Tablolara ilişkin veri sözlüğü sistem çözümleme aşamasında verilmiştir. Tablo isimleri aşağıdaki gibidir. Tabloların detay bilgilerine sistem çözümleme aşamasından erişebilirsiniz.

1. Member(Üye) Tablosu
2. Member Status(Üye Durum) Tablosu
3. Member Types(Üye Tipi) Tablosu
4. Member Payment(Üye Ödemeleri) Tablosu
5. Aidatlar Tablosu
6. Başvuru Tipi Tablosu
7. Beldeler Tablosu
8. Billings (Faturalar) Tablosu
9. Çoban Başvuru Tablosu
10. Districts(İlçeler) Tablosu
11. Doğumlar Tablosu
12. Fees(Harçlar) Tablosu
13. Genders(Cinsiyet) Tablosu
14. Files(Dosyalar) Tablosu
15. Koop_Kkbs Tablosu
16. Destek Başvuru Tablosu

17. Movements (Hareketler) Tablosu
18. Neighborhoods(Mahalleler) Tablosu
19. Logs (Loglar) Tablosu
20. Messages (Mesajlar) Tablosu
21. Personel Tablosu
22. Pozisyon Tablosu

3.5.2 MYSQL DATABASE DİAGRAM



3.6 KULLANICI ARAYÜZ TASARIMI

Sistem çözümleme aşamasında gereksinimlerin ve tasarımın net bir şekilde ortaya konulmasından dolayı, kullanıcı ara yüzlerinin tamamı sistem çözümleme aşamasında tasarlanmış ve iş sahibi tarafından kabul edilmiştir. Sisteme ilave hiçbir şey eklenmeyecektir.

3.7 ENTEGRASYON VE TEST GEREKSİNİMLERİ

Sistem geliştirilirken Çağlayan Modeli uygulanmıştır. Çağlayan(Şelale) modeli test adımı; analiz ve tasarım dokümanlarındaki tüm fonksiyonel ve fonksiyonel olmayan gereksinimler ve tasarımlar için test senaryoları yazılır ve bu test senaryoları icra edilerek yazılımın testleri yapılır. Testlerde bulunan hata hangi adımdan (analiz, tasarım veya kodlama) kaynaklı ise o adıma geri dönülerek hata giderilir. Örneğin, hata kodlamadan kaynaklı ise, sadece kodlama düzeltilir ve tekrar test adımına geçilir. Ancak, hata analiz veya tasarımdan kaynaklı ise bu adımlara gidilerek analiz ve tasarım dokümanları güncellenir, daha sonra kodlama düzeltilir ve test aşamasına geçilir. Test adımı sonunda, yazılımda herhangi bir hatası bulunamaz ise, entegrasyon adımına geçilir ve yazılım, canlı ortama entegre edilerek müşterinin kullanımına açılır. Entegrasyon ve testler yapılan plana uygun olarak yürütülecektir.

3.8 ORTAK ALT SİSTEMLERİN TASARIMI

Geliştirilecek olan sistemde kullanılan alt sistemlerin tasarımı gerçekleştirilmiştir.

3.8.1 YETKİLENDİRME ALT SİSTEMİ

Sistemde işlem yapacak personeller yetki düzeylerine göre pozisyonlara ayrılmıştır. İstenildiği takdirde bu pozisyonların sayısı artırılabilir. Şu an için yönetici ve personel olarak iki pozisyon tanımı vardır. Yönetici kullanıcısının yapabileceği işlemler farklı, personel kullanıcısının yapacağı işlemler farklı olacağı için böyle bir yetkilendirme yöntemi geliştirilmiştir.

3.8.2 YEDEKLEME ALT SİSTEMİ

Sunucu hizmeti alınan hosting firması tarafından günlük yedeklemeler otomatik olarak yapılacaktır. Olağan dışı durumlara hazırlıklı olmak amacıyla sistemin komple yedeği alınacak, istenildiği zamanda son yedekten geriye dönülebilecektir. Ayrıca üye ve personele ait detay bilgilerde Excel ortamına aktarılarak manuel olarak yedeklenebilecektir.

3.8.3 GÜVENLİK ALT SİSTEMİ

Sistemde güvenliği sağlayabilmek için öncelikle kullanıcı parolaları oluşturulurken zor çözülecek şifre oluşturmalarına zorlanmıştır. Örneğin şifreleri en az sekiz karakter olacak şekilde ve içerisinde en az bir rakam, bir büyük harf ve bir özel karakter olması şartı getirilmiştir. Sisteme girişten sonra belirli bir süre hiçbir işlem yapılmaz ise oturum otomatik sonlanacaktır. Ayrıca sistemde loglama yapılarak hangi kullanıcının hangi işlemi yaptığı kayıt altına alınacaktır.

4.GERÇEKLEŞTİRİM

4.1 GİRİŞ

Bu aşama tasarım sonucu üretilen fiziksel modelin bilgisayar ortamında çalışan yazılım biçimine dönüştürülmesi çalışmalarını içermektedir.

4.2 YAZILIM GELİŞTİRME ORTAMLARI

Projede kullanılacak olan programlama dili, veri tabanı yönetim sistemi, geliştirme ortamları, görsel tasarım araçları, bakım ve destek araçlarından oluşur. Bu araçlar geliştirme ve sınamaya süresince kullanılacak olan araçlardır.

Proje Geliştirme Ortamı	: Zend Studio, XAMPP
Programlama Dili	: PHP
Veri Tabanı Yönetim Sistemi	: MySQL
Görsel Tasarım Araçları	: HTML, CSS, JQuery, CorelDraw
Bakım ve Destek Aracı	: Codeigniter, MVC Framework, Metronic Dashboard Template

4.3 KODLAMA SİTİLİ

Geliştiricilerin koda baktıkları zaman ne yapılmak istendiğini kısa sürede anlayabilmeleri yazılım geliştirme açısından önemlidir. Bunun olabilmesi için de bir kodlama stili üzerinden kodlamalar yapılmıştır. Yazılım projemiz **lower snake case** kodlama stili baz alınarak yazılmıştır.

Classlar, metodlar, değişkenler ve sabitler için mantıklı isimler verilerek kodların anlaşılabilir ve okunabilir olması sağlanmıştır.

Sistemin kodlanması aşamasında birden fazla yazılım uzmanı görev alacağından yazılımcıların birbirlerini anlamaları için açıklama satırları kullanmaları ve kod yazım düzenine dikkat etmeleri önemlidir. Bunun içinde gereken yerlerde açıklama satırları eklenmiş ve kod yazım düzenine dikkat edilmiştir. Aynı zamandan programın daha sonra ki aşaması olan bakım aşaması için de kodlama stilinin önemi büyüktür.

4.4 TASARLANAN SİSTEMİN KODLANMASI

Seçilmiş olan geliştirme ortamları kullanılarak tasarım sonrası oluşturulan fiziksel modelin bilgisayar üzerinde çalışan yazılıma dönüştürülmesi için gerekli kodlamalar yapıldı.

Bu kodlardan bazıları;

GİRİŞ KONTROLÜ;

```
<?php
```

```
class LoginController extends CI_Controller{
```

```
    public function index(){
```

```
        // $this->load->helper('cookie');
```

```
        // $remember = get_cookie('remember_me');
```

```
        // if($remember) redirect(base_url('projeler'));
```

```
        //////////
```

```
        $user = $this->session->userdata('admin_session');
```

```
        if($user){
```

```
            $flashdata = $this->session->flashdata();
```

```
            $this->session->set_flashdata($flashdata);
```

```
            redirect(base_url('yonetim'));
```

```
        }else{
```

```
            $this->load->model('AdminEventsModel');$where = ['id' => 1];
```

```
            $this->son['title'] = $this->AdminEventsModel->get_item('title', $where);
```

```
            $this->load->view('pages/login', $this->son);
```

```
}  
}
```

```
public function login(){  
    // $this->load->helper('cookie');  
  
    $user = $this->session->userdata('admin_session');  
  
    if($user){ redirect(base_url('yonetim')); }  
  
    $email = $this->input->post('email');  
  
    $password = $this->input->post('password');  
  
    if(!$email || !$password){  
        $alert_error = 'Tüm alanları doldurunuz..';  
        $this->session->set_flashdata('alert_error', $alert_error);  
    }else{  
        $email = $this->security->xss_clean($email);  
        $password = $this->security->xss_clean($password);  
        $this->load->model('UserModel');  
        $where = [  
            // 'email' => $email,  
            'username' => $email,  
            'password' => md5($password)  
        ];  
        $row = $this->UserModel->get($where);  
        if($row){  
            $user = [  
                'id' => $row->id,  
                'username' => $row->username,
```

```

        'email' => $row->email,

        'name_surname' => $row->name_surname,

        'profile_pic' => $row->profile_pic,

        'position' => $row->position

    ];

    $this->session->set_userdata('admin_session', $user);

    $alert_success = 'Hoşgeldiniz.. '.$row->name_surname.$this->input->post('remember');

    $this->session->set_flashdata('alert_success', $alert_success);

    // if($this->input->post('remember')== 'on'){

    //     set_cookie('remember_me', 'asd');

    // }

    redirect(base_url('yonetim'));

}else{

    $alert_error = 'Böyle bir kullanıcı bulunamadı..';

    $this->session->set_flashdata('alert_error', $alert_error);

}

}

redirect(base_url());

}

public function logout(){

    $this->session->unset_userdata('admin_session');

    $alert_success = 'Çıkış Yapıldı..';

    $this->session->set_flashdata('alert_success', $alert_success);

    redirect(base_url());

}

} ?>

```

ŞABLON EKLEME;

```
<div class="m-grid__item m-grid__item--fluid m-wrapper">
    <!-- BEGIN: Subheader -->
<div class="m-content">
    <div class="col-xl-12">
        <!--begin:: Widgets/User Progress -->
        <div class="m-portlet m-portlet--full-height">
            <div class="m-portlet__head">
                <div class="m-portlet__head-caption">
                    <div class="m-portlet__head-title">
                        <h3 class="m-portlet__head-text">
                            Şablonlar (Word)
                        </h3>
                    </div>
                </div>
                <div class="m-portlet__head-tools">
                    <a href="<?php echo base_url(). 'yeni_sablon'; ?>">
                        <button type="button" class="btn btn-info">
                            Şablon Ekle
                        </button>
                    </a>
                </div>
            </div>
            <div class="m-portlet__body">
                <div class="tab-content">
                    <div class="tab-pane active" id="m_widget4_tab1_content">
                        <div class="m-widget4 m-widget4--progress">
<?php if($template){ foreach($template as $temp){ ?>
                    <div class="m-widget4__item">
                        <div class="m-widget4__img m-widget4__img--">
                            
                        </div>
                        <div class="m-widget4__info">
                            <span class="m-widget4__title">
                                <?php echo $temp->file_name; ?>
                            </span>
                            <br>
                            <span class="m-widget4__sub">
                                Oluşturan: <?php echo $temp->name_surname; ?>
                            </span>
                        </div>
                        <div class="m-widget4__progress">
                            <div class="m-widget4__progress-wrapper">
                                <span class="m-widget17__progress-number">
                                    <?php echo $temp->date; ?>
                                </span>
                            </div>
                        </div>
                    </div>
                    <div class="m-widget4__ext">
```

```

        <a href="<?php echo base_url().'.templates/'. $temp->url; ?>" target="_blank"
class="m-btn m-btn--hover-brand m-btn--pill btn btn-sm btn-secondary">
        Gözet
    </a>
</div>
<div class="m-widget4__ext">
    <a href="<?php echo base_url().'.sablon_sil/'. $temp->id; ?>" onclick="myFunction()"
class="m-btn m-btn--hover-danger m-btn--pill btn btn-sm btn-secondary">
    Sil
    </a>
</div>
</div>
<?php }}else{ ?>
    Hiç şablon yok.
<?php } ?>
</div>
</div>
<div class="tab-pane" id="m_widget4_tab2_content">
<div class="m-widget4 m-widget4--progress">
</div>
</div>
<div class="tab-pane" id="m_widget4_tab3_content">
<div class="m-widget4 m-widget4--progress">
    </div>
</div>
</div>
</div>
<!--end:: Widgets/User Progress -->
</div>
</div>
</div>
</div>
<!-- end:: Body -->

```

4.5 KOD GÖZDEN GEÇİRME

Yapılmış olan kodlama çalışması sonucu kod gözden geçirme işlemi yapılmıştır. Bu sayede kodun kalitesi artmış, olası hatalar yazılım teslim edilmeden ortaya çıkarılmış, proje geliştiricileri arasında bilgi birikimi artmıştır.

Yapılacak olan sistem sınavasının ve kabul sınavasının ardında sistem kullanılmaya başlayacaktır.

5.YAZILIM DOĞRULAMA VE GEÇERLEME

5.1 GİRİŞ

- **Doğrulama** ürünü kullanacak kişilerin isteklerinin karşılanıp karşılanmadığını test eden etkinliklerden,
- **Geçerleme** ise ürünün içsel niteliğine ilişkin izleme ve denetim etkinliklerinden oluşur.

5.2 SINAMA YÖNETİMİ

Biz projemizi Kara Kutu Testi yöntemini kullanarak sınamalarımızı gerçekleştirdik.

Kara kutu test tekniği, test edilen yazılımın davranışının uygulamanın kod yapısına, kodun implementasyonuna ve kodun nasıl işlediğine bakılmaksızın uygulanır. Bu test tekniğinde tamamen uygulamanın girdileri ile çıktılarına yoğunlaşılır. Kara kutu testlerinin amacı verilen girdiler ile istenilen çıktının elde edilmesidir. Bu nedenle yazılımın iç işleyişi ile ilgili diğer bilgilerle ilgilenilmez.

Bu teknik ile aşağıdaki hata tipleri tespit edilmiş ve giderilmiştir.

- Doğru olmayan veya hiç olmayan işlevlerin tespiti
- Arayüz hataları
- Performans hataları
- Veri tabanlarına ulaşma veya veri yapılarındaki hatalar

5.3 SINAMA VE BÜTÜNLEŞTİRME STRATEJİSİ

Sinama ve bütünleştirme işlemlerinin gerçekleştirilmesinde Aşağıdan Yukarı Sinama ve Bütünleştirme yöntemi izlenmiştir.

5.3.1 AŞAĞIDAN YUKARIYA SINAMA VE BÜTÜNLEŞTİRME

Projenin bütünleştirilmesi ve sınanmasında kullanılan aşağıdan yukarıya sinama ve bütünleştirme stratejisine göre ilk olarak işçi birimleri sınanır. Bir üstteki birim ile sınanması gerektiğinde bu üst bileşen için bir sürücü program kullanılır bu aşama tamamlandıktan sonra sürücü olan parçalar gerçekleriyle yer değiştirilir ve bu sistem tamamlana kadar devam eder. Sistem bütünleştirilmesi ve sınanması bu yöntem kullanılarak gerçekleştirilmiştir.

Bu stratejiye göre sınamalar;

- Birim sinama
- Alt Sistem sinama
- Sistem sinama
- Kabul sinama

Sırasına göre gerçekleştirilir.

5.4 SINAMA RAPORU

Sinama işlemleri sonucunda sinama belirtileri, sinamayı yapan kişi, sinama tarihi, sinama yapılan yer ve açıklamalar türündeki bilgiler eklenerek sinama raporları oluşturulmuştur.

6.KURULUM-BAKIM-EĞİTİM

6.1 GİRİŞ

Geliştirilmiş olan Küçükbaş Hayvan Yetiştiricileri Birliği Otomasyon sistemi yazılım doğrulama ve geçerleme işleminin başarıyla sonuçlanması ile teslim hazır hale gelmiştir.

6.2 KURULUM

Sinama testlerinden geçirilmiş ve hatalardan arındırılmış olan ve müşterinin bütün ihtiyaçlarına cevap verecek yazılımı işletmeye almak bu safhada artık çok kolaydır. Otomasyonumuz web üzerinden çalışacağı için son kullanıcıların bilgisayarlarına herhangi bir kurulum yapılmaz. Kişinin sisteme erişebilmesi için bir bilgisayar ve internet bağlantısı yeterli olacaktır. Kurulum için hosting firmasından sunucu (yer) hizmeti ve alan adı(domain) satın alınarak hazırlanan sistem uzak sunucuya yüklenmiştir. Erişim için gerekli konfigürasyonlar yapılmıştır. Böylece müşteri belirlenen domain adı ile sisteme erişecektir. Bu şekilde otomasyon hayata geçirilmiştir.

6.3 BAKIM

Planlama aşamasında belirlenen Bakım Planına uygun olarak bakım hizmeti verilecektir.

6.4 EĞİTİM

Planlama aşamasında yapılmış Eğitim Planına uygun olarak kullanıcı eğitimleri verilmiştir.