



BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
2020-2021 EĞİTİM-ÖĞRETİM YILI GÜZ DÖNEMİ
VERİ TABANI YÖNETİM SİSTEMLERİ

Çiftlik Otomasyonu

152120161003 – Fatih ARSLAN

152120171004 – Hasan DALKILIÇ

152120171072 – Mahmut KILIÇ

152120171040 – Erkut KAYA

İletişim: fatiharslan2634@gmail.com

Telefon: 05318383701

Referanslar:

Ziraat Mühendisi Fatih ORMAN

Ziraat Mühendisi Merve ÖZGEN

İçindekiler

1. Giriş	4
1.1. Hedef	4
1.2. Doküman Standartları	4
1.3. Hedef Kitle ve Okuma Tavsiyeleri	4
1.4. Projenin Kapsamı	4
2.1. Ürün Perspektifi	4
2.2. Ürünün Özellikleri	4
2.3. Kullanıcı Sınıfı ve Özellikleri	5
2.3.1. Çiftlik Yöneticisi	5
3.1. Kullanım Kolaylığı	5
3.2. Veri Güvenliği	5
5. Veri ile ilgili İsterler	6
5.1 Proje Kapsamında Veri tabanında Oluşturulacak Tablolar	6
5.1.13. Billing	9
5.1.14. Warehouse	9
6.Tablo Özelliklerinin Tanımlanması ve İçeriği	9
6.1. Tablolar Arası İlişkiler	14
7. Sorgular	14
7.1. Sorgu Örnekleri	15
8. VTYS Araç Kullanımı	16
8.1. Saklı Yordam Kullanımı (Stored Procedure)	16
8.2. Hata Ayıklama Kullanımı (Error Handling)	17
8.3. Trigger Kullanımı	19
9. Arayüz Tasarımları	20
9.1 Giriş Arayüzü	20
9.2 Google Hesabıyla Giriş Arayüzü	20
9.3. SignUp Arayüzü	21
9.4. Mail Doğrulama Arayüzü	21
9.5. Ana Sayfa Arayüzü	22
9.6. Product Review Arayüzü	22
9.6.1 Vaccine Registration	23
9.7. Users Arayüzü	23

9.8. Warehouse Arayüzü	24
10. Diğer Entegrasyonlar.....	27
11. Proje Üyeleri ve İş Bölümü	28
12. Sonuç ve Değerlendirme	29

1. Giriş

1.1. Hedef

Çiftlik otomasyon sisteminde, hayvancılık ile uğraşan kişilerin işlerinin daha sistematik bir şekilde veri tabanı ve veri tabanına entegre olacak bir sistemle birlikte kolaylaştırılması amaçlanmaktadır. Hayvan sınıflandırılması, çiftlik çalışanları ve özelliklerinin kaydı, hayvan bazlı üretimler, gerekli yemler, kapasite takibi, müşterilerinin alışverişi gelir giderlerin işlendiği bir arayüz programı geliştirilecektir. Projenin tasarımında C# programlama dili tabanlı uygulama geliştirilmesi amaçlanmıştır.

1.2. Doküman Standartları

Dokümanda kalın puntıyla yazılmış kısımlar başlıklardır. Her başlık belirli alt başlıkları içermektedir.

1.3. Hedef Kitle ve Okuma Tavsiyeleri

Bu doküman büyük ve küçük çaplı çiftlik sahiplerini hedefliyor. İlk bölümde giriş, ikinci bölümde programın teknik özelliklerini ve yapısı hakkında bilgi veriyor. Üçüncü bölümde ise ayrıntılı bir biçimde sistemde olması gereken özellikler belirtiliyor. Son bölümde ise veri tabanı içeriğinin bahsedildi.

1.4. Projenin Kapsamı

Çiftlik sahibinin ihtiyaçlarını düzenleyip, organize etmesini ve takibi bir program vasıtasıyla sağlar. Hedefi, çiftlik sahibinin kolayca kullanabileceği, değişen ihtiyaçlara hızlı cevap veren sistematik bir yazılım oluşturmaktır.

2. Genel Tanım

2.1. Ürün Perspektifi

Kullanıcının yalnızca veri girişleri ile kontrol ve takip edebileceği, müşterilere belirli satışların yapıldığı; veteriner, müşteri ve yöneticinin gelir gider vb. birçok işlem yapılabildiği çiftlik otomasyon programıdır. Herhangi bir yazılımın alt ürünü değildir. Sistemin genel ihtiyaçlarını karşılayan bir üründür.

2.2. Ürünün Özellikleri

Program, kolay kullanılabilen birçok verinin takip edilebildiği çiftlik otomasyon programıdır. Öne çıkan özellikleri; hayvanlarla ilgili detaylı bilgi, çiftlik çalışanları takibi, süt ve süt ürünlerinin takibi, hayvanların aşısı, beslenme verimlilikleri konularını içermektedir. Müşterilerin doğal çiftlik

ürünlerini inceleyip satın alabilecek. Projenin içereceği her bir özelliğin ayrıntısını üçüncü ve dördüncü bölümde bulabilirsiniz.

2.3. Kullanıcı Sınıfı ve Özellikleri

Kullanıcının istediği herhangi bir veriye hızlıca erişebilmesi, güzel bir arayüz, karmaşıklığın minimuma indirildiği hızlı bir programdır.

2.3.1. Çiftlik Yöneticisi

Çiftlik sahiplerinden oluşur. Projenin tasarım ve dizaynı bu kullanıcı odaklı yapılmıştır. Tüm alanlara erişim hakkı vardır.

3. Sistem Gereksinimleri

Aşağıda gruplandırılmadan maddeler halinde sistemde olması gereken özellikler belirtilmiştir.

3.1. Kullanım Kolaylığı

Projede kullanım kolaylığı en temel gereksinimlerdenidir. Tüm ihtiyaçları istisnasız karşılayacak basit bir arayüzle desteklenecektir.

3.2. Veri Güvenliği

Kullanan kişinin sadece kendisine ayrılmış alanları görebilmesi, girişte kullanıcı adı ve şifre ile güvenliği sağlamak oldukça önemlidir. Kullanıcı yönetici değilse, sınırlı alanlara erişir.

4. Yazılım İsterleri

1. Çiftlik yönetici, veteriner veya müşterinin üye girişi panelini kullanarak sisteme girmesi
 - a. E- posta ile sisteme girilmesi
 - b. Sosyal medya hesabı ile sisteme girilmesi
2. Yeni üye için üye olma (sign-up) ekranı
3. Yöneticinin çiftlik çalışanlarını sisteme kayıt etmesi ve düzenleme işlemleri
4. Kullanıcıların yetkilendirilmesi
 - a. Yönetici
 - b. Çalışanlar
 - i. Çalışma saatlerinin ayarlanması
5. Hayvan isimlerinin numaralanması
6. Hayvan bilgilerinin sistemi girilmesi
 - a. Her türlü çiftlik hayvanı ve bilgileri
7. Yöneticinin hayvanları kontrol ve analiz etmesi.

- a. Memeli hayvanların kontrolü
 - b. Kümes hayvanların kontrolü
- 8. Ahırların kaydı ve görevli ilişkilendirilmesi
- 9. Ürün nakliye bilgilerinin sistemde tutulması
- 10. Çiftlik hayvanların sağlık durumu kontrolü
 - a. Aşı kontrolü
 - b. Doğurganlık oranının kontrol ve değerlendirilmesi
- 11. Üretkenlik oranının kontrolü
 - a. Günlük süt üretimi analizi.
 - b. Haftalık yumurta sayısı takibi.
- 12. Kümeslerin kaydı ve personel ilişkilendirilmesi
- 13. Müşterinin mevcut ürünleri inceleyebilmesi
- 14. Müşterinin alışveriş işlemleri
 - a. Müşteri istediği ürünleri seçer ve alışverişini tamamlar.
 - b. Fatura alabilir veya isteğe bağlı mailine e – fatura gönderilir.
 - c. Çıktı alabilir.

Özelliklerini barındıran bir uygulama oluşturulacaktır.

5. Veri ile ilgili İsterler

5.1 Proje Kapsamında Veri tabanında Oluşturulacak Tablolar

5.1.1. props_mammal_Animal

- earring_id
- animal_type
- barn_id
- age
- born_date
- gender
- breed
- weight
- price

5.1.2. coop

- coop_id
- personnel_id
- coop_name
- capacity
- egg_production

- feed_consumption

5.1.3. Animals

- animal_type
- animal_name

5.1.4. staff_members

- personnel_id
- personnel_name
- personnel_surname
- birthDate
- speciality
- email
- tel_no
- start_dateOf_employee
- salary

5.1.5. barn

- barn_id
- barn_name
- personnel_id

5.1.6. props_poultry

- bird_id
- coop_id
- animal_type
- breed
- price
- weight
- amountOf_meat

5.1.7. statusMammal

- earring_id
- daily_milk_production
- feed_water_conseption
- amountOf_meat
- vaccine_id
- vaccine_status
- last_vaccine_status
- ferility_rate

5.1.8. vaccine

- vaccine_id
- vaccine_name

5.1.9. signUp

- userId
- nameSurname
- ssn
- email
- telNo
- username
- password
- code

5.1.10. signIn

- id
- username
- password

5.1.11. propProduct

- pro_id
- pro_name
- calories
- total_fat
- cholestrel
- sodium
- carbohydrate
- protein
- vitamin_A
- productionDate
- ExpirationDate
- price

5.1.12. sellProduct

- pro_id
- product_name

- amount
- production_date
- expiration_date
- price

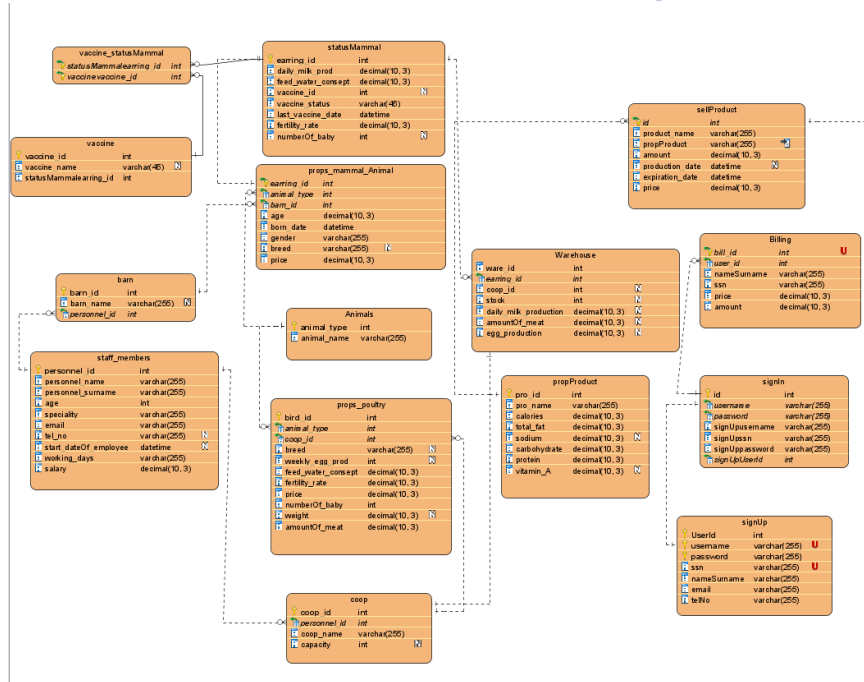
5.1.13. Billing

- bill_id
- user_id
- pro_id
- product_name
- price
- amount,

5.1.14. Warehouse

- ware_id
- earring_id
- coop_id
- stock
- daily_milk_production
- amountOf_meat
- egg_production

6.Tablo Özelliklerinin Tanımlanması ve İçeriği



props_mammal_Animal Tablosu

earring_id : INT olarak tanımlanır. props_mammal_Animal tablosunun primary key'idir. Hayvanın küpe numarasını tutar.

animal_type : INT olarak tanımlanır. Animals tablosundan foreign key olarak gelir. Hayvanın tipini gösterir.

barn_id : INT olarak tanımlanır. Barn tablosundan foreign key olarak gelir. Ahır numarasını gösterir.

born_date: DATE olarak tanımlanır. Doğduğu tarihi belirtir.

gender: VARCHAR(255) olarak tanımlanır. Cinsiyetini belirtir.

breed: VARCHAR(255) olarak tanımlanır. Irkını belirtir.

weight : DECIMAL(10,3) olarak gösterilir. Hayvanın ağırlığını gösterir.

price: DECIMAL(10,3) olarak gösterilir. Hayvanın fiyatını gösterir.

staff_members Tablosu

personnel_id: INT olarak tanımlanır. Staff_members tablosunun primary key' idir. personelin idisini tutar.

personnel_name: VARCHAR(255) olarak tanımlanır. Personelin ismini gösterir.

personnel_surname: VARCHAR(255) olarak tanımlanır. Personelin soyadını tutar.

age: INT olarak tanımlanır. Yaşı tutar.

speciality: VARCHAR(255) olarak tanımlanır. Uzmanlık alanını gösterir.

email: VARCHAR(255) olarak tanımlanır. E-maili tutar.

tel_no: VARCHAR(255) olarak tanımlanır. Telefon bilgilerini tutar.

start_dateOf_employee : DATE olarak tanımlanır. İşe başlama günü belirtilir.

salary: DECIMAL(10,3) olarak tanımlanır. Maaşları gösterilir.

statusMammal Tablosu

earring_id: INT olarak tanımlanır. props_mammal_Animal tablosunun Foreign key olarak gelir. Hayvanın küpe numarasını tutar.

daily_milk_production: DECIMAL(10,3) olarak tanımlanır. Günlük süt üretimini gösterir.

feed_water_consept: DECIMAL(10,3) olarak tanımlanır. Su tüketimini gösterir.

vaccine_status: VARCHAR(255) olarak tanımlanır. Aşı durumunu gösterir.

last_vaccine_date: DATE olarak tanımlanır. Son aşı tarihini gösterir.

Fertility_rate: DECIMAL(10,3) olarak tanımlanır. Doğurganlık oranını verir.

Vaccine Tablosu

Vaccine_id: INT olarak tanımlanır. vaccine tablosunun Primary Key'idir.

Vaccine name: VARCHAR(255) olarak tanımlanır. Aşının ismini tutar.

barn Tablosu

barn_id : INT olarak tanımlanır. barn tablosunun Primary Key'idir. Ahırın id' sini tutar.

barn_name : VARCHAR(255) olarak tanımlanır. Ahırın ismini tutar.

personnel_id : INT olarak tanımlanır. staff_members tablosundan Foreign Key olarak gelir.

Animals Tablosu

animal_type : INT olarak tanımlanır. Animals tablosunun Primary Key' idir. Hayvan isminin id karşılığını tutar.

animal_name : VARCHAR(255) olarak tanımlanır. Hayvan ismini tutar.

coop Tablosu

coop_id: INT olarak tanımlanır. coop tablosunun Primary Key' idir. Kümes id'sini tutar.

personnel_id: INT olarak tanımlanır. staff_members tablosundan Foreign Key olarak gelir.

Personel id' sini tutar.

coop_name: VARCHAR(255) olarak tanımlanır. Kümes ismini tutar.

capacity: INT olarak tanımlanır. Kümes kapasitesini tutar.

Egg_production: DECIMAL(10,3) olarak tanımlanır.

Feed_consumption: DECIMAL(10,3)

signIn Tablosu

id: INT olarak tanımlanır. signIn tablosunun primary key'idir. Kullanıcının numarasını tutar.

Username: VARCHAR(45) olarak tanımlanır. signIn tablosunun primary key'idir. signUp tablosundan foreign keyi olarak gelir. Kullanıcı adını tutar.

Password: VARCHAR(45) olarak tanımlanır. signUp tablosundan foreign keyi olarak gelir. Kullanıcı şifresini tutar.

NameSurname VARCHAR(255) olarak tanımlanır.

propProduct Tablosu

pro_id: INT olarak tanımlanır. Ürün numarasını tutar.

Ware_id: INT olarak tanımlanır. Ambar numarasını tutar.

pro_name: VARCHAR(255) olarak tanımlanır. propProduct tablosunun primary key'idir. Ürün adını tutar.

Water: DECIMAL(10,3) olarak tanımlanır. Su miktarını tutar.

calories: DECIMAL(10,3) olarak tanımlanır. Ürünün kalori miktarını tutar.

total_fat: DECIMAL(10,3) olarak tanımlanır. Ürünün toplam yağ miktarını tutar.

sodium: DECIMAL(10,3) olarak tanımlanır. Ürünün soydum miktarını tutar.

carbohydrate: DECIMAL(10,3) olarak tanımlanır. Ürünün karbonhidrat miktarını tutar.

protein: DECIMAL(10,3) olarak tanımlanır. Ürünün protein miktarını tutar.

vitamin_A: DECIMAL(10,3) olarak tanımlanır. Ürünün vitamin A miktarını tutar.

Amount: DECIMAL(10,3) olarak tanımlanır. Ürünün miktarını tutar.

ProductionDate: DATE olarak tanımlanır. Ürünün üretim tarihini tutar.

sellProduct Tablosu

id: INT olarak tanımlanır. Satılan ürünün numarasını tutar.

product_name: VARCHAR(255) olarak tanımlanır. sellProduct tablosunun primary keyi'dir. propProduct: tablosundan foreign keyi olarak gelir. Ürünün ismini tutar.

amount: DECIMAL(10,3) olarak tanımlanır. sellProduct tablosunun primary keyi'dir. Ürün miktarını tutar.

production_date: DATE olarak tanımlanır. Ürünün üretim tarihini tutar.

expiration_date: DATE olarak tanımlanır. Ürünün son kullanma tarihini tutar.

price: DECIMAL(10,3) olarak tanımlanır. sellProduct tablosunun primary keyi'dir. Ürün fiyatını tutar.

Billing Tablosu

bill_id : INT olarak tanımlanır. Billing tablosunun primary keyi'dir. Faturanın numarasını tutar.

user_id: INT olarak tanımlanır. signIn tablosunun foreign key'i olarak gelir. İsim numarasını tutar.

Pro_id: INT olarak tanımlanır.

Product_name: VARCHAR(255) olarak tanımlanır. Ürün adını tutar.

price: DECIMAL(10,3) olarak tanımlanır. sellProduct tablosundan foreign key olarak gelir. Fiyatı tutar.

amount: DECIMAL(10,3) olarak tanımlanır. sellProduct tablosundan foreign key olarak gelir. Satılan ürünün miktarını tutar.

props_poultry Tablosu

bird_id: INT olarak tanımlanmalıdır. Bu tablonun primary Key'idir. Tavuğun numarasını belirtir. animal_type: INT olarak tanımlanmalı. Hayvanın cinsini belirtmek için kullanılır. Animals isimli tablodan Foreign Key olarak gelir.

coop_id: INT olarak tanımlanmalı. Kümes numarasını belirtir. Coop tablosundan Foreign Key olarak gelir.

breed: VARCHAR(255) olarak tanımlanır. Kümes hayvanının ırkını belirtir.

price: DECIMAL(10,3) olarak tanımlanır. Tavuğun fiyatını belirtir.

weight: DECIMAL(10,3) olarak tanımlanır. Tavuğun kütlesini belirtir.

amountOf_meat: DECIMAL(10,3) olarak tanımlanır. Tavuktan çıkan et kütlesini belirtir.

signUp Tablosu

nameSurname: VARCHAR(255) olarak tanımlanır. Kaydolan kişinin isim ve soyismini tutar.

ssn: VARCHAR(255) olarak tanımlanır ve üyenin kimlik numarasını sistemde tutmaya yarar.

email: VARCHAR(255) olarak tanımlanır ve üyenin e-posta adresini sistemde tutar.

telNo: VARCHAR(255) olarak tanımlanır ve üyenin telefon numarasını sistemde tutar.

username: VARCHAR(255) olarak tanımlanır. Bu tablo için password ile birlikte Primary Key özelliği taşırlar. Üyenin kullanıcı adını sistemde tutar.

password: VARCHAR(255) olarak tanımlanır. Bu tablo için username ile birlikte Primary Key özelliği taşırlar. Üyenin şifresini sistemde tutar.

warhouse Tablosu

ware_id: INT olarak tanımlanır.

Earring_id: INT olarak tanımlanır.

Coop_id: INT olarak tanımlanır.

stock INT olarak tanımlanır.

Daily_milk_production: DECIMAL(10,3) olarak tanımlanır.

amountOf_meat: DECIMAL(10,3) olarak tanımlanır.

Egg_production DECIMAL(10,3) olarak tanımlanır.

6.1. Tablolar Arası İlişkiler

- **Animals** tablosu ile **props_mammal_Animal** tablosu arasında **ONE TO ONE**
- **barn** tablosu ile **staff_members** tablosu arasında **ONE TO MANY**
- **props_mammal_Animal** tablosu ile **barn** tablosu arasında **MANY TO ONE**
- **coop** tablosu ve **staff_members** tablosu arasında **MANY TO MANY**
- **props_poultry** tablosu ile **Animals** tablosu arasında **MANY TO ONE**
- **signIn** tablosu ile **signUp** tablosu arasında **ONE TO ONE**
- **sellProduct** tablosu ile **propProduct** tablosu arasında **MANY TO MANY**
- **Billing** tablosu ile **sellProduct** tablosu arasında **ONE TO MANY**
- **statusMammal** tablosu ile **props_mammal_Animal** tablosu arasında **ONE TO ONE**
- **vaccine** tablosu ile **statusMammal** tablosu arasında **MANY TO MANY** ilişkisi vardır.

7. Sorgular

Projede SQL (Structured Query Language) tasarımı, analizi ve yazılımı için MSQL ve MySQL programları kullanılmıştır. Bu uygulamalarda oluşturulan tabloların uygulama ile bağlantısı C# programlama dili ile gerçekleştirilmiştir. Projede kullanılan örnek sorgular bir sonraki maddede gösterilmiştir.

7.1. Sorgu Örnekleri

--Dört farklı tabloyu inner joinle birleştirerek id numaralarını, hayvanların isimlerini, ahır isimlerini, et miktarlarını ve aşı isimlerini ekrana bastırır.

```
select top 30
p.earring_id,
a.animal_name,
b.barn_name,
s.amountOf_meat,
v.vaccine_name
from
Props_Mammal_Animal p
inner join Animals a on a.animal_type = p.animal_type
inner join Barn b on p.barn_id = b.barn_id
inner join StatusMammal s on p.earring_id = s.earring_id
inner join Vaccine v on s.vaccine_id = v.vaccine_id
```

--İki farklı tabloyu inner joinle birleştirerek id numaralarını, cinsini, aşı id numralarını ve aşı durumlarını ekrana bastırır.

```
Go
Create Procedure vaccineCheck
As
SELECT
pm.earring_id,
pm.breed,
st.vaccine_id,
st.vaccine_status
FROM StatusMammal st
INNER JOIN Vaccine v ON v.vaccine_id=st.vaccine_id
INNER JOIN Props_Mammal_Animal pm ON st.earring_id=pm.earring_id
Go
```

--Ahırda çalışanın ad ve soyadını döndüren kod bloğu

```
Select personnel_name, personnel_surname, barn_name
From Staff_Members s
Inner Join Barn b on s.personnel_id = b.personnel_id
```

8. VTYS Araç Kullanımı

8.1. Saklı Yordam Kullanımı (Stored Procedure)

Saklı yordamın kullanım kolaylığından dolayı, projemizde bu kalıptan birçok yerde faydalandık. Gerekli parametrelerle kullanıma imkan tanıyan bu kalıbın kod örneklerini aşağıda göstermeye çalıştık.

--COW_Breed prosedürü memeli türünden, belirtilen ırkta kaç tane hayvan olduğu bilgisini ekrana bastırır.
CREATE PROCEDURE Mammal_Breed @breed nvarchar(30) AS SELECT COUNT (barn_id) AS "Number of Mammal Animal in Given Breed" FROM props_mammal_Animal WHERE props_mammal_Animal.breed = @breed GO
EXEC Mammal_Breed @breed = 'Holstein' EXEC Mammal_Breed @breed = 'Angus' EXEC Mammal_Breed @breed = 'Jersey' EXEC Mammal_Breed @breed = 'Merinos' EXEC Mammal_Breed @breed = 'Simental' EXEC Mammal_Breed @breed = 'Kilis'
--Bulunan kümes hayvanlarının toplam fiyatını ekrana bastıran prosedür
CREATE PROCEDURE Total_Price_in_Selected_Poultry_Type @animal_type INT AS SELECT SUM(p.price) AS "Total Price of Animals in Selectes Type" FROM Animals a INNER JOIN props_poultry p ON p.animal_type = a.animal_type WHERE @animal_type=a.animal_type GO
EXEC Total_Price_in_Selected_Poultry_Type 3
--Seçilen kümes numarasına göre, o kümede bulunan tavukların toplam fiyatının gösterilmesini sağlayan prosedür
CREATE PROCEDURE Total_Price_in_Selected_Coop @coop_id INT AS SELECT SUM (price) AS "Total Price of Animals in Selectes Coop" FROM props_poultry WHERE props_poultry.coop_id = @coop_id GO Total_Price_in_Selected_Coop seçili kümedeki kanatlıların toplam fiyatını ekrana bastırır.
EXEC Total_Price_in_Selected_Coop @coop_id = 1; EXEC Total_Price_in_Selected_Coop @coop_id = 2;
--Total_Price_in_Selected_Mammal_Type isme göre seçili memeli hayvanların toplam fiyatını ekrana bastırır.
Create PROCEDURE Total_Price_in_Selected_Mammal_Type @animal_name Varchar(100) AS SELECT SUM(m.price) AS "Total Price of Animals in Selectes Type" FROM Animals a INNER JOIN props_mammal_Animal m ON m.animal_type = a.animal_type WHERE @animal_name=a.animal_name

<pre>select * from props_mammal_Animal GO</pre>
<pre>EXEC Total_Price_in_Selected_Mammal_Type @animal_name = 'Cow'; EXEC Total_Price_in_Selected_Mammal_Type @animal_name = 'Sheep'; EXEC Total_Price_in_Selected_Mammal_Type @animal_name = 'Goat';</pre>
<pre>--Kümeste çalışan personelin ismine göre belirtilen bilgileri döndüren kod bloğu.</pre>
<pre>Create PROCEDURE coopPersonelProcedure @personnel_name VArchar(100) as select s.personnel_name, s.personnel_surname, s.tel_no, c.coop_name from Staff_Members s inner join Coop c on s.personnel_id = c.personnel_id where personnel_name = @personnel_name Go</pre>
<pre>Exec coopPersonelProcedure 'Stephany'</pre>

8.2. Hata Ayıklama Kullanımı (Error Handling)

Proje tasarımı yapılırken C# programlama dili kullanılmıştır. Kodlamanın her aşamasında hatalı girdi/çıktıları kontrol etmek için Try-Catch kalıbı kullanılmıştır. Kullanıcıya hata doğrultusunda düzeltilmesi gereken alanları ve parametreleri ekrana hata mesajı ile belirtir. Belirtilen kalıplar aşağıda ,örnek olması adına, gösterilmiştir.

```
try {
    LoginStatus loginStatus = dao.CheckLogin2(txtBoxUsername.Text,
        txtBoxPassword.Text);
    if (loginStatus == LoginStatus.OK)
    {
        MessageBox.Show("Your're logged");
        Form_Dashboard form_Dashboard = new Form_Dashboard();
        form_Dashboard.ShowDialog();
        this.Dispose();
    }
    else
    {
        MessageBox.Show("Login failed");
    }
}
catch(Exception)
{
    MessageBox.Show("Incorrect enter");
    con.Close();
}
```

```

try
{
    if (isValid())
    {
        SqlCommand cmd = new SqlCommand("INSERT INTO Staff_Members VALUES (@personnel_id, @personnel_name, @personnel_surname, @birthDate,
        @speciality, @email, @tel_no, @start_dateOf_employee, @salary)", con);
        cmd.CommandType = CommandType.Text;
        int personnel_id = int.Parse(txtPersonnelId.Text);
        cmd.Parameters.AddWithValue("@personnel_id", personnel_id);
        cmd.Parameters.AddWithValue("@personnel_name", txtPersonelName.Text);
        cmd.Parameters.AddWithValue("@personnel_surname", txtPersonnelSurname.Text);
        cmd.Parameters.AddWithValue("@birthDate", txtAge.Text);
        cmd.Parameters.AddWithValue("@speciality", txtSpeciality.Text);
        cmd.Parameters.AddWithValue("@email", txtEmail.Text);
        cmd.Parameters.AddWithValue("@tel_no", txtPhone.Text);
        cmd.Parameters.AddWithValue("@start_dateOf_employee", txtStart.Text);
        decimal salary = decimal.Parse(txtSalary.Text);
        cmd.Parameters.AddWithValue("@salary", salary);
        con.Open();
        cmd.ExecuteNonQuery();
        con.Close();
        MessageBox.Show("New Item is Adding", "Saved", MessageBoxButtons.OK, MessageBoxIcon.Information);
        getStaffRecord();
        refresh();
    }
    else
    {
        MessageBox.Show("Please Select one of Them", "Save", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
catch (Exception)
{
    MessageBox.Show("Incorrect enter");
}

```

```

try
{
    txtBirdId.Text = dataGridPoultryRegister.SelectedRows[0].Cells[0].Value.ToString();
    txtAnimalType.Text = dataGridPoultryRegister.SelectedRows[0].Cells[1].Value.ToString();
    txtCoopId.Text = dataGridPoultryRegister.SelectedRows[0].Cells[2].Value.ToString();
    txtBreed.Text = dataGridPoultryRegister.SelectedRows[0].Cells[3].Value.ToString();
    txtPrice.Text = dataGridPoultryRegister.SelectedRows[0].Cells[4].Value.ToString();
    txtWeight.Text = dataGridPoultryRegister.SelectedRows[0].Cells[5].Value.ToString();
    txtAmountofMeat.Text = dataGridPoultryRegister.SelectedRows[0].Cells[6].Value.ToString();
}
catch
{
    MessageBox.Show("Please Select one of Them", "Update", MessageBoxButtons.OK, MessageBoxIcon.Error);
}

```

8.3. Trigger Kullanımı

Trigger(Tetikleyici) , database'e yaptığımız ekleme, çıkarma vb. güncelleme işlemlerinde yapılanları takip altına almak için kullanılması gereken bir kalıp olduğundan dolayı projemizde yer verdik.

--Sisteme herhangi bir değer eklendiğinde mesaj döndürüp, gerekli bilgileri de yeni bir trigger tablosunda toplar.

```
CREATE TRIGGER triggerInsert ON Vaccine
FOR INSERT
AS
declare @vaccine_id int;
declare @vaccine_name Varchar(50);
declare @audit_action varchar(100);
select @vaccine_id=i.vaccine_id from inserted i;
select @vaccine_name=i.vaccine_name from inserted i;
set @audit_action='Inserted Record Trigger.';
insert into Vaccine_Audit
(vaccine_id, vaccine_name, Audit_Action, Audit_Timestamp)
values(@vaccine_id, @vaccine_name, @audit_action, getdate());
PRINT 'AFTER INSERT trigger get run.'
GO
```

--Status Mammal tablosunda, aşı değerleri güncellendiğinde mesaj üretir ve ilgili bilgileri yeni bir tabloda tutar.

```
CREATE TRIGGER triggerUpdate ON StatusMammal
FOR UPDATE AS
declare @earring_id int;
declare @vaccine_status varchar(100);
declare @last_vaccine_date date;
declare @audit_action varchar(100);
select @earring_id=i.earring_id from inserted i;
select @vaccine_status =i.vaccine_status from inserted i
select @last_vaccine_date =i.last_vaccine_date from inserted i
if update(vaccine_status)
set @audit_action='Updated Record - Vaccine was updated.';
if update(last_vaccine_date)
set @audit_action='Updated Record - Vaccine date was updated.';
insert into
StatusMammalTrigger(earring_id, vaccine_status, last_vaccine_date, Audit_Action, Audit_Time
stamp)
values(@earring_id, @vaccine_status, @last_vaccine_date, @audit_action, getdate());
PRINT 'AFTER UPDATE Trigger.'
GO
```

--Barn tablosunda herhangi bir değer silindiğinde sisteme trigger oluşturur ve bir tabloda bu sonuçları tutar.

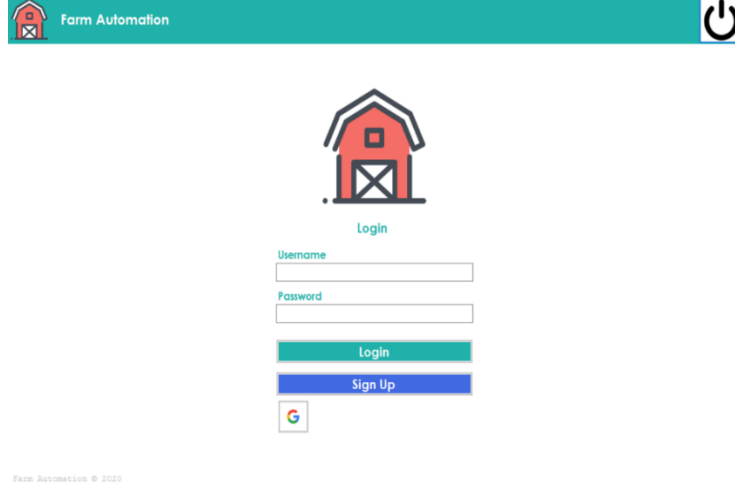
```
CREATE TRIGGER triggerBarn ON barn
AFTER DELETE
AS
declare @barn_id int;
declare @barn_name varchar(100);
declare @audit_action varchar(100);
select barn_id=d.barn_id from deleted d;
select barn_name=d.barn_name from deleted d;
set @audit_action='Deleted... Trigger.';
insert into barnTrigger
(barn_id, barn_name, Audit_Action, Audit_Timestamp)
values(@barn_id, @barn_name, @Audit_Action, getdate());
PRINT 'AFTER DELETE TRIGGER.'
Go
```

9. Arayüz Tasarımları

Programın ara yüz tasarımına ait ekran görüntüleri:

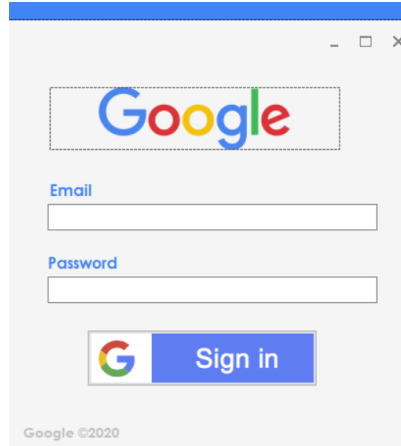
9.1 Giriş Arayüzü

Uygulama başlatıldığında kullanıcının karşısına gelen ilk ekrandır. Bu ekranda oturum açmak için isim ve şifre istenir. Google hesabı ile oturum açmak isteyen kullanıcı için Google ikonu bulunur. Ayrıca hesap oluşturabilmek için kaydol butonu bulunmaktadır. Kullanıcının girdiği bilgilere ait bir hesap veritabanında bulunmuyorsa hata mesajı alacaktır.



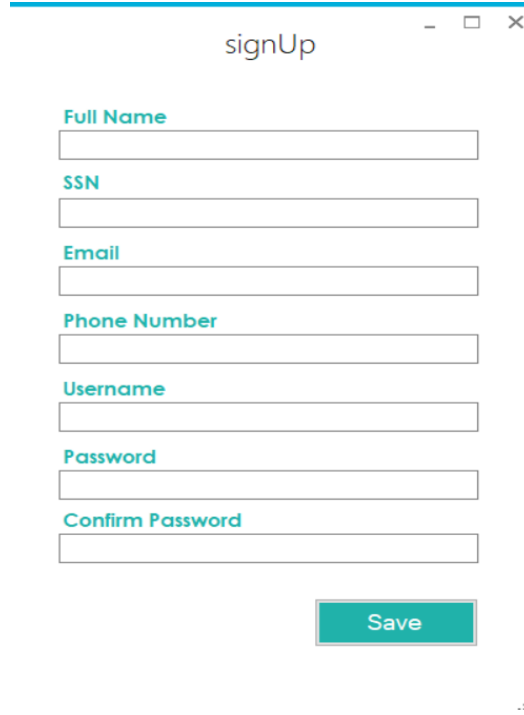
9.2 Google Hesabıyla Giriş Arayüzü

Kullanıcı istediğinde Google hesabıyla oturum açabilir. Giriş sayfasında Google ikonuna tıkladığı takdirde yeni bir sayfa açılır. Bu sayfada Google hesabıyla giriş yapılabilmesi için mail adresi ve şifre istenir.



9.3. SignUp Arayüzü

Kullanıcı kaydolmak istediğinde giriş sayfasında Sign Up butonuna basarak SignUp ekranına gelir. Bu sayfada kullanıcıdan tam isim, kimlik numarası, mail, telefon numarası, isim, şifre ve şifrenin tekrarı bilgilerini girer. Herhangi bir bilgi eksik girildiğine kullanıcı hata mesajı alacaktır.



signUp

Full Name

SSN

Email

Phone Number

Username

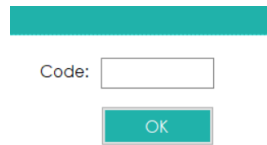
Password

Confirm Password

Save

9.4. Mail Doğrulama Arayüzü

SignUp ekranında save butonuna tıklandığında mail kısmına yazılan mail hesabına sistem tarafından 4 haneli bir sayı gönderilir. Bu sayı, maili onaylamak amacıyla gönderilir. Maile gelen sayı programda doğru girildiği takdirde kaydolma başarılı bir şekilde gerçekleşir.

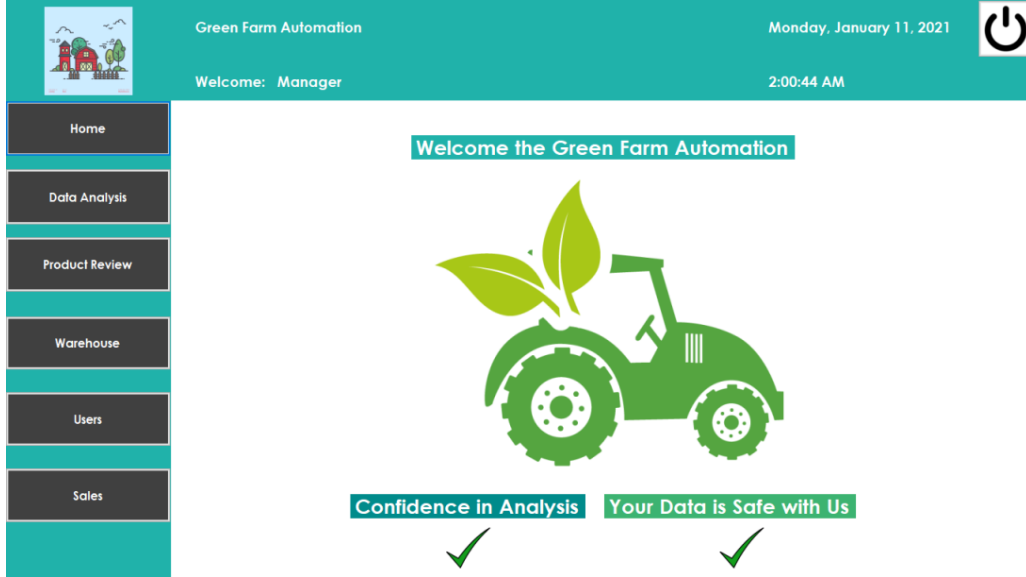


Code:

OK

9.5. Ana Sayfa Arayüzü

Kullanıcı giriş işlemi başarıyla gerçekleştiğinde ana sayfa açılır. Sayfanın sol sekmesinde seçenekler bulunur. Home, Data Analysis, Product Review, Users, Sales, Settings, Registration olmak üzere 7 seçenek bulunur.



9.6. Product Review Arayüzü

Product review seçeneğine tıklanıldığında ürünlerin besin değerlerini (su, kalori, toplam yağ, kolesterol, karbonhidrat, protein, A vitamini) değerlerini listeler. Import ve Export seçenekleriyle Excel dosyasına yazdırma/okutma işlemlerini sağlar.

	pro_id	ware_id	pro_name	water	calories	total_fat	cholestorel	carbohydrate	protein	vitamin_A	amount
▶	1	2	Chicken Egg	0.000	75.000	5.010	0.213	0.600	6.250	0.317	
	2	1	Chicken M...	0.000	234.000	17.100	0.075	0.000	18.800	0.013	
	3	3	Cow Meat	0.000	251.000	19.200	0.067	0.000	18.200	0.000	
	4	1	Cow Milk	87.500	42.300	3.500	0.050	5.000	3.400	0.700	
	5	1	Goat Milk	86.400	45.200	3.800	0.070	4.900	3.100	0.900	
	6	2	Goose Egg	0.000	185.000	13.300	0.825	0.500	13.900	0.418	
	7	2	Sheep Meat	0.000	222.000	17.200	70.000	0.000	17.200	0.000	
*											

Import XLS

Export XLS

9.6.1 Vaccine Registration

Aşı tablosunda güncelleme, ekleme ve çıkarma gibi işlemlerin yapılabildiği formdur.

Vaccine

Vaccine ID:

Vaccine Name:

	vaccine_id	vaccine_name
▶	1	Handle
	2	BRUCELLA
	3	BCD
	4	BRD
	5	CLOSTRIDIAL
	6	Newcastle
	7	Gumboro
	8	Newcastle + Bronsist

Save


Update

Delete

Refresh

9.7. Users Arayüzü

User seçeneğine tıklanıldığında açılan sayfada çalışanların bilgileri listelenir. Çalışanın ID numarası, adı, soyadı, yaşı, uzmanlık alanı, mail, telefon numarası, işe başlama tarihi ve maaş bilgileri gösterilir. ‘Add Members’ butonuna tıklanarak çalışan ekleme sayfasına ulaşılır.



Users

Add Members

personnel_id	personnel_name	personnel_surname	birthDate	speciality	email	tel_no	start_dateOf	salary
▶ 1	Stephany	Aslan	6/3/1998	Barn Manager	barn@gmail.c...	5318448877	6/3/2020	1300.000
2	Efe	Aydoğdu	10/3/1990	barn	efe@gmail.com	5135464844	10/3/1998	1000.000
3	Kim	Joan	10/3/1990	barn	jordan@gmail...	5135464844	10/3/1998	1000.000
4	Hannibal	Kingdom	11/3/1994	coop	Kingdom@qm...	5231264844	10/3/1998	1000.000
5	Hithere	Honathan	10/3/1990	barn	jordan@gmail...	5135464844	10/3/1998	1000.000
6	Veli	Hikmet	10/9/1968	Vet	vetin@gmail.c...	52132546452	11/3/1988	1100.000

9.7.1 Add Members Arayüzü

Eklenecek işçinin ID numarası, adı, soyadı, yaşı, uzmanlık alanı, mail, telefon numarası, işe başlama tarihi ve maaş bilgileri bu sayfada girilir ve ardından Save butonuna tıklanır. İşçinin herhangi bir özelliği (ID hariç) değiştirilmek istenildiğinde Update butonu kullanılır. Bir işçinin

çıkarılması gerektiğinde o işçi seçilir ve ardından Delete butonuna tıklanır. Textbox'ları sıfırlamak için Refresh butonu kullanılır. Ayrıca kayıtlı işçiler bu sayfada da listelenir.

AddMembers

Personnel ID: Email:

Personnel Name: Phone Number:


Personnel Surname: Start Date of Employee:

Age: Salary:

personnel_id	personnel_name	personnel_surname	age	email	tel_no	start_dateOf_employee	salary
*							

9.8. Warehouse Arayüzü

Registration seçeneğine tıklanıldığında Animal List sayfası açılır. Sayfanın yukarısında Add Barn (Ahır ekleme), Add Coop (Kümes Ekleme), Animal Register (Hayvan Kaydetme) ve Poultry Register (Kümes Hayvanı Ekleme) seçenekleri bulunur.



Add Barn

Add Coop

Animal List

Animal Register

Poultry Register

earring_id	animal_type	barn_id	birthDate	gender	breed	weight	price
26100001	1	1	5/15/2018	female	Holstein	724.000	9700.000
26100002	1	2	4/19/2018	female	Holstein	800.000	9700.000
26100003	1	1	5/15/2018	female	Holstein	689.000	8000.000
26100004	1	4	5/15/2018	female	Simental	827.000	10600.000
26100005	1	3	2/27/2018	male	Holstein	1004.000	11400.000
26100006	1	1	5/15/2018	female	Jersey	820.000	10500.000
26100007	1	2	5/15/2018	female	Holstein	950.000	9700.000
26100008	1	1	5/15/2018	female	Holstein	780.000	11350.000
26100009	1	1	5/15/2018	female	Simental	1068.000	9700.000
26100010	1	2	2/14/2019	female	Holstein	718.000	9250.000
26100011	1	3	3/10/2019	female	Jersey	720.000	9150.000
26100012	1	4	3/12/2019	female	Simental	648.000	9700.000
26100013	1	1	5/15/2018	male	Jersey	1200.000	11000.000
26100015	1	2	5/15/2018	female	Jersey	1068.000	9700.000
26100016	1	3	5/15/2018	female	Holstein	1068.000	9700.000
26100017	1	1	4/24/2018	female	Jersey	724.000	10550.000
26100018	1	4	4/25/2018	female	Holstein	881.000	10700.000
26100019	1	1	9/13/2018	female	Jersey	780.000	10300.000
26100020	1	1	9/15/2018	male	Holstein	814.000	9850.000
26100021	1	3	9/15/2018	female	Simental	610.000	9999.000
26100022	1	4	9/15/2018	female	Holstein	1068.000	9700.000

9.8.1. Barn Register Arayüzü

Bu sayfada eklenecek ahırın ID numarası, ismi ve işçinin ID numarası girilir ve ardından Save butonuna tıklanarak ekleme yapılır. ID numarası değiştirilmemek üzere güncelleme yapılması gerektiğinde Update butonu kullanılır. Ahırın veritabanından çıkarılması gerektiğinde o ahır seçilir ve ardından Delete butonuna tıklanır. Textbox'ları sıfırlamak için Refresh butonu kullanılır. Ayrıca kayıtlı ahırlar bu sayfada listelenir.

Barn Register

Barn ID:

Barn Name:

Personnel ID:

	barn_id	barn_name	personnel_id
*			

Save

Update

Delete

Refresh

9.8.2. Coop Register Arayüzü

Bu sayfada eklenecek kümesin ID numarası, ismi ve işçinin ID numarası, kapasitesi, yumurta üretimi ve besin üretimi girilir ve ardından Save butonuna tıklanarak ekleme yapılır. ID numarası değiştirilmemek üzere güncelleme yapılması gerektiğinde Update butonu kullanılır. Kümesin veritabanından çıkarılması gerektiğinde o kümes seçilir ve ardından Delete butonuna tıklanır. Textbox'ları sıfırlamak için Refresh butonu kullanılır. Ayrıca kayıtlı kümesler bu sayfada listelenir.

Coop Register

Coop ID:

Capacity:

Personel ID:

Egg Production:

Coop Name:

Feed Production:

	coop_id	personnel_id	coop_name	capacity	egg_production	feed_consumption
*						

Save

Update

Delete

Refresh

9.8.3. Animal Register Arayüzü

Bu sayfada eklenecek hayvanın küpe numarası, çeşidi ve bulunduğu ahırın ID numarası, doğduğu tarih, cinsiyeti, cinsi, ağırlığı, et miktarı ve fiyatı girilir ve ardından Save butonuna tıklanarak ekleme yapılır. ID numarası değiştirilmemek üzere güncelleme yapılması gerektiğinde Update butonu kullanılır. Hayvanın veritabanından çıkarılması gerektiğinde o hayvan seçilir ve ardından Delete butonuna tıklanır. Textbox'ları sıfırlamak için Refresh butonu kullanılır. Ayrıca kayıtlı hayvanlar bu sayfada listelenir.

Animal Register

Earring ID:

Breed:

Animal Type:

Weight:

Barn ID:

Amount of Meat:

Born Date:

Price:

Gender:

Save

Update

Delete

Refresh

9.8.4. Poultry Register Arayüzü

Bu sayfada eklenecek kümes hayvanının ID numarası, çeşidi, kümes numarası, cinsi, fiyatı, ağırlığı ve et miktarı girilir ve ardından Save butonuna tıklanarak ekleme yapılır. ID numarası değiştirilmemek üzere güncelleme yapılması gerektiğinde Update butonu kullanılır. Kümes hayvanının veritabanından çıkarılması gerektiğinde o kümes hayvanı seçilir ve ardından Delete butonuna tıklanır. Textbox'ları sıfırlamak için Refresh butonu kullanılır. Ayrıca kayıtlı kümes hayvanları bu sayfada listelenir.

Poultry Register

Bird ID:

Price:

Animal Type:

Weight:

Coop ID:

Amount of Meat:

Breed:

	bird_id	animal_type	coop_id	breed	price	weight	amountOf_meat
*							

Save

Update

Delete

Refresh

9.8.5. Data Analysis Arayüzü

Bu kısımda, SQL sorgularından yararlanarak uygulama içerisinde kullanıcının gerekli analiz işlemlerini yapması sağlandı ve bu analiz sonuçları görüntülenmek amacıyla tablolar ile desteklendi.

Show Personnel Barn Relationships:	Show	Show Total Daily Milk Production From Mammals:	Show
Show Personnel List Phone Number:	Show	Show Top 30 Animal Name, Vaccine Status, Amount OF Meat:	Show
Show Vaccine Status:	Show	Show Total Number of Cows:	Show
Show Total Price of Poultry:	Show	Show Total Number of Sheep:	Show
Show Total Daily Milk Production From Mammals:	Show	Show Total Number of Chickens:	Show
Show Vaccine Table Trigger:	Show	Show Total Number of Goats:	Show
Show StatusMammal Table Trigger:	Show	Show Total Price of Selected Mammal Animal:	Show
Number of Animals Selected Breed:			


10. Diğer Entegrasyonlar

10.1. Sosyal Medya ile Login Bağlantısı

Arayüzlerin gösterildiği maddede olduğu gibi kullanıcı istediğinde Google hesabıyla oturum açabilir. Giriş sayfasında Google ikonuna tıkladığı takdirde yeni bir sayfa açılır. Bu sayfada Google hesabıyla giriş yapılabilmesi için mail adresi ve şifre istenir. Kullanıcı kaydolmak istediğinde giriş sayfasında Sign Up butonuna basarak SignUp ekranına gelir. Bu sayfada kullanıcıdan tam isim, kimlik numarası, mail, telefon numarası, isim, şifre ve şifrenin tekrarı bilgilerini girer. Herhangi bir bilgi eksik girildiğine kullanıcı hata mesajı alacaktır. SignUp ekranında save butonuna tıklandığında mail kısmına yazılan mail hesabına sistem tarafından 4 haneli bir sayı gönderilir. Bu sayı, maili onaylamak amacıyla gönderilir. Maile gelen sayı programda doğru girildiği takdirde kaydolma başarılı bir şekilde gerçekleşir. Sosyal medya ile login bağlantısı ile ilgili ekran görüntüleri ‘Arayüz Tasarımları’ isimli 9. maddede gösterilmiştir.

10.1. Export/Import XLS

Export XLS ve import XLS butonları listelenen verilerde kullanılır. Veritabanında listelenen bilgileri Excel dosyasına kaydedilmesini Export XLS, Excel dosyasındaki verilerin programa kaydedilmesini Import XLS sağlar.



Farm Automation

Welcome: Manager

Home

Data Analysis

Product Review

Users

Sales

Settings

Registration

Product Review

Product Register

1

2

3

4

5

6

pro_id

pro_name

water

calories

total_fat

cholesterol

carbohydrate

protein

vitamin_A

Chicken Egg

Chicken Meat

Cow Meat

Cow Milk

Goat Milk

Goose Egg

Sheep Meat

0.000

0.000

0.000

42.300

45.200

0.000

0.000

75.000

234.000

19.200

3.500

3.800

185.000

222.000

5.010

17.100

0.067

0.050

0.070

13.300

17.200

0.213

0.075

0.087

0.050

0.070

0.825

70.000

0.800

0.000

5.000

4.900

0.500

0.000

0.000

6.250

18.800

19.200

3.400

3.100

13.900

17.200

0.317

0.913

0.900

0.700

0.900

0.418

0.900

Import XLS

Export XLS

	A	B	C	D	E	F	G	H
1	pro_id	pro_name	calories	total_fat	cholestorel	carbonhydrate	protein	vitamin_A
2		3 Chicken Egg	75	5.01	0.213	0.6	6.25	0.317
3		7 Chicken Meat	234	17.1	0.075	0	18.8	0.013
4		5 Cow Meat	251	19.2	0.067	0	18.2	0
5		1 Cow Milk	42.3	3.5	0.05	5	3.4	0.7
6		2 Goat Milk	45.2	3.8	0.07	4.9	3.1	0.9
7		4 Goose Egg	185	13.3	0.825	0.5	13.9	0.418
8		6 Sheep Meat	222	17.2	70	0	17.2	0
9								

11. Proje Üyeleri ve İş Bölümü

11.1 Proje Üyelerinin Roller

Fatih Arslan: Project Manager, Coder, Designer

Hasan Dalkılıç: Designer

Mahmut Kılıç: Project Manager, Designer

Erkut Kaya: Project Manager, Coder, Designer

11.2 Proje Adımları ve Gerçekleştiren Üyeler

İsterlerin Hazırlanması: Fatih Arslan, Hasan Dalkılıç, Mahmut Kılıç, Erkut Kaya

ER Diyagramı Tasarımı: Fatih Arslan, Mahmut Kılıç, Erkut Kaya

Arayüz Tasarımı: Fatih Arslan, Erkut Kaya

Sunumun Hazırlanması: Fatih Arslan, Hasan Dalkılıç, Mahmut Kılıç, Erkut Kaya

Raporun Hazırlanması: Fatih Arslan, Hasan Dalkılıç, Mahmut Kılıç, Erkut Kaya

11.3 Üyelerin Kişi Başına Çalışma Saatleri

Fatih Arslan: 70 Saat

Hasan Dalkılıç: 50 Saat

Mahmut Kılıç: 60 Saat

Erkut Kaya: 60 Saat

Grup Olarak Çalışılan Toplam Süre: 240 Saat

12. Sonuç ve Değerlendirme

Programlama dilindeki kararsızlığın giderilmesiyle birlikte C# programlama dili ile uygulama tasarlandı ve SQL veritabanındaki veriler entegre edildi. Kullanıcı dostu basit arayüz tasarımlarıyla bu uygulamayı kullanacak herhangi bir yönetici, kolay bir şekilde istediği verileri girebilir, güncelleyebilir ve silebilir hâle geldi. Daha fazla vakit olsaydı proje hakkında daha detaya girilebilirdi. Örneğin tablolar daha detaylandırılabilirdi, kullanıcı istediği takdirde arayüzün renk, boyutunda değişiklikler yapabilmesine izin verilebilirdi, uygulamanın kullanımında ortaya çıkabilecek hatalar hakkında geniş çaplı tespit yapılabilirdi. Sadece Türkiye içerisinde değil, İngilizce sürümü gibi dil eklentisi eklenebilirdi.

No.	Özdeğerlendirme Başlığı	% olarak (0-100)
1	Proje başındaki amaç-hedeflerde değişme oranı	%10
2	Mevcut veri yapısı (tasarım, oluşturulan tablo vb) nin proje ister ve uygulama ihtiyaçlarını karşılama oranı	%95
3	Mevcut Karmaşık sorguların tasarım ve tam entegre uygulamaya alınma oranı;	%95
4	Saklı Yordam, Hata Ayıklama ve Tetikleyicilerin tam entegre uygulamaya alınma oranı;	%100
5	Ara yüzlerin tamamlanması ve tam entegre uygulamaya alınma oranı	%100
6	Diğer Entegrasyonların (sosyal medya, e-mail/gsm, export/import) tam entegre uygulamaya alınma oranı	%90
7	Projenin tam entegre tamamlanma oranı	%95
8	Uygulamanın alanda kullanılabilirlik oranı	%100
9	Sunumun; genel akışı, teknik içerik aktarım yöntemleri, grup çalışanlarının kendi içinde dengeli şekilde konu anlatımı, zamanı etkin kullanım vb. açısından değerlendirmesi	%90
10	Proje raporunun proje çalışmalarını yansıtırma oranı	%95