Bil 372 Projesi

Tohum Projesi

Ekrem Sönmezer, Ahmet Bahadır Eyuboğlu, Mustafa Bora Arslan, Arda Ekinci

Abstract

© 2020 Published by Tohum Projesi

Tohum, Tarım, Proje, Agriculture

1. Proje

Tohum Projesi; tarım ve çevre ile alakalı paylaşılan, açık veya kapalı kaynaklardan elde edilen verileri gruplandırma, depolama, yönetme, tarım ve çevrecilikle alakalı alt alanlarda projeksiyon oluşturma amacıyla oluşturulup, her düzeyden üreticinin rahatlıkla kullanabileceği bir web uygulaması olmak amacıyla hayata geçirilmiştir. Temel amacı düşük bütçeli üreticinin bile kolaylıkla erişebileceği, işlerini yönetirken profesyonel yaklaşabileceği bir araç olmaktır. Bu amaçla yola çıkmış olan Tohum Projesi’nin ilk versiyonu 09.12.2020 tarihinde paylaşılmıştır.

1. Gereksinim Analizi ve Tasarım
   1. Görev Dağılımı

Ekibin netleşmesinden sonra roller de kesinlik kazanmıştır. Ahmet Bahadır ve Ekrem görünümler üzerine çalışmasına, Hüseyin Arda veri tabanının oluşturulması ve oluşacak teknik sorunların çözümüne odaklanacakken Mustafa Bora da arka planda dönen operasyonları yönetip temel fonksiyonların gerçekleştirilmesinden sorumlu olacak şeklinde karar verildi. Fakat zaman ilerledikçe rol kavramları değişti ve gelişti. Bunun sonucunda herkes, hemen hemen her konuyla ilgilenmiş oldu. Örnek vermek gerekirse Ekrem ve Mustafa Bora veri tabanı için veri araştırmasında da görev alırken, Ahmet Bahadır ve Hüseyin Arda’da arka planda yazılmış fonksiyonlara yardımcı oldular.

* 1. Veri Tabanı Geliştirme

Proje süresinde Miniworld, ER/EER diyagramı ve Relational Mapping oluşturulmasının yanı sıra uygulama içerisinde kullanılacak olan veri tabanının özelliklerine karar verildi. Veri tabanının sahip olduğu özellikler ve ifade ettiği anlam Github Repository’de bulunan Resources dosyasının içerisinde Miniworld.docx dosyasında bulunmaktadır. Proje içerisinde de backend ile uyumlu bir şekilde çalıştığı bilinen PostgreSQL kurulup ekibin aynı veri tabanını kullanabilmesini sağlamak için de pythonanywhere.com üzerinden veri tabanı kiralanmıştır.

* 1. Backend Geliştirme

Süreç içerisinde yapılan veri tabanı ve ekran tasarımlarının kontrolü adına backend geliştirme adımları da veri tabanı ve ekran tasarımları ile paralel gitmesine karar verildi. Bu noktada backend için kullanılacak en uygun dilin Python, web framework’ün Django olduğuna karar verildi. İlk etap için Django ile başarılı bir şekilde ilerlenirken, oluşan zaman sıkıntısından dolayı sürecin ilk başında belirtilmiş olan risk planlamasında da çözümü belirtilen Flask ile çalışmalara devam edildi.

* 1. Frontend Geliştirme

User Interface geliştirmek için Django ya da Flask ile uyumlu hareket etmesi için günümüz web geliştiricilerin de kullandığı HTML, CSS ve Javascript kullanılmasına karar verildi. Bu ekranların son kullanıcıya aktardığı verilerin göze hitap etmesini kolaylaştırmak amacıyla da Chart.js kütüphanesinin kullanılması noktasında fikir birliği oluşturuldu.

1. Gerçekleme
   1. Veri tabanı gerçekleme

Ek kısmında yer almaktadır.

* 1. Backend gerçekleme

Backend geliştirmede öncelikle başvurulan teknolojik çözümler framework olarak Django ve Postgresql idi. Fakat Djangonun karmaşık ve detaylı yapısı yerine, hafif ve hızlı olan Flask, verilen kapsamda daha etkili bir seviyede geliştirmemize imkan sağlayacaktı. Diğer tarafta ise Postgresql ise python programları ile olan uyumluluğundan ötürü tercih edildi. Tablolar, ilişkiler, ve connection string ayarlandıktan sonra operasyonlar Flask üzerinden gerçekleştirildi.

*3.2.1 Temel Flask sayfaları*

Öncelikle Flask üzerinden login ve register sayfaları gerçekleştirildi. İlgili girdi formları ( mail, ad, soyad vb) tasarlandı ve ilgili veritabanı bağlantıları entegre edildi. Ardından ilgili yönlendirme sayfaları yapıldı, her sayfada ortak bulunan tool bar, navigation bar, form layout, nightmode ve lightmode kullanıcının daha etkili bur kullanım içerisinde bulunması için inşa edildi.

*3.2.2 Sayfalar ve içerikleri*

Login/Register: İlgili formlar içerisinde girdiler çıktılar, ilgili kişisel bilgiler istenildi ve merkezine yerleştirilecek şekilde esnetik bir html düzenlenmesi yapıldı.

Home/Dashboard: İlk girişte kullanıcıya genel takip etmek isteyecekleri bir hissiyat için belli grafikler ekranda gerçekleştirildi. Türkiye genelinde istihdam, ekili alan grafikleri, ekin veya ürün grafikleri ve son olarak platformda bulunan kullancıların son hareketlerine yönelik loglar ve gerçekleştirdikleri fonksiyonlar esnetik bir şekilde düzenlendi.

Overview: İlgili sayfada, kullanıcının navigation bar üzerinden de ulaşabileceği sayfalar merkeze getirelecek şekilde daha düzenli bir şekilde ifade edildi.

Add Product/ Add Growing: Öncelikle Add Product sayfasında, seçilen ürünün alan, miktar, yıl ve ilgili istihdam teknolojik ve biyolojik detayları girildikten sonra bu ürün, ürün verileri ile ilgili veri tablosuna eklendi.

Add Growing için, yetiştirilen ürün başlangıç tarihi ve ilgili bitiş tarihine göre yetiştirilmekte olanların tutuldu veri tablosuna eklendi ardından ilgili kullanıcının hali hazırda yetiştirmekte olduğu ürünün tablo olarak listelendiği ve buradanda hasat zamanı geldiğinde veya manuel olarak hasat edildiğinde ilgili ürün verileri tablosuna eklenebilecek şekilde gerçekleştirildi.

Tips: İlgili yıl içerisinde ekilen veya hasat edilen ürünlerden en çok kullanılanları bölgesine göre tavsiye edilip ekranda gösterildiği sayfadır.

Settings: Kullanıcının kolaylıkla anlayıp, kedisi hakkındaki bilgilerini değiştirip güncelleyebileceği bir esnetik sayfa olarak inşa edildi.

Products (Fruits / Vegetables / Grains / Legumes ): Burada ilgili ürünlerden filtreleme, arama ve tablo içerisinde gösterme özellikleri sayfa içeriği ile uyumlu bir şekilde gerçekleştirildi.

Data ( Machines / Medicines / Workers ): Burada da kullanıcılar tarafından toplanan verilerin, tarımda kullanılan makine, ilaç ve istihdam olarak gruplara ayrılarak uygun formlarda filtreleme arama ve incelemeye yönelik bir şekilde sayfaların arasına eklendi.

*3.2.3 Veri tabanı operasyonları*

İlgili sayfaların hepsinde birden fazla tablo ile etkileşim içerisinde bulunuldu. Kullanıcı oluşturma Ayarlar ve Kayıt olma sayfalarında systemlog ve farmer tabloları, Hasat ve Ürün ekleme işlemlerinde systemlog, productdata, product, farmer ve bunların aralarındaki ilişkilerin içerdiği tabloların hepsi işlendi herhangi bir id kullanılmadan güncellendi. Son olarak ise dahsboard ve diğer ürün grafikleri ve verinin raporlanmasına yönelik sayfalarda, herhangi bir güvenlik açığı vermeyecek şekilde ( idlerin sayfa içerinde kullanılması vb) işlenerek, veriler SUM, AVG gibi toplu fonksiyonlar GROUP BY, DESC gibi SELECT WHERE gibi filtereleme operasyonları ile uyumlu bir şekilde gerçekleştirildi.

*3.2.4 Frontend ile etkileşimler*

Bu veriler flask sessionları ve flask ile üzerinde jinja2 üzerinden verilerin yönetimi ve dağıtımı yapılarak ilgili tablo ve grafiklere anlamlı bir şekilde sayfanın esnetik düzenini bozmayacak şekilde gerçekleştirildi. Bu düzenlemeler esnasında verilerin hızlanması için çoğu karmaşık SQL stringleri üzerinden uygulandı, bu sayede sayfa yüklenme hızı ve verilerin anlamlaştırılması çok daha hızlı bir şekilde entegre edildi.

Frontend'den Backend'e Veri Aktarımı: Formlar submit edildikten sonra render edilen html sayfaları üzerinde çekilen girdi alanları içerisinden uygun tablolarına (systemlog, productdata vb) insert işlemleri gerçekleştirildi ve bulunan zincirleme tetiklenen işlemler çalıştırıldı.

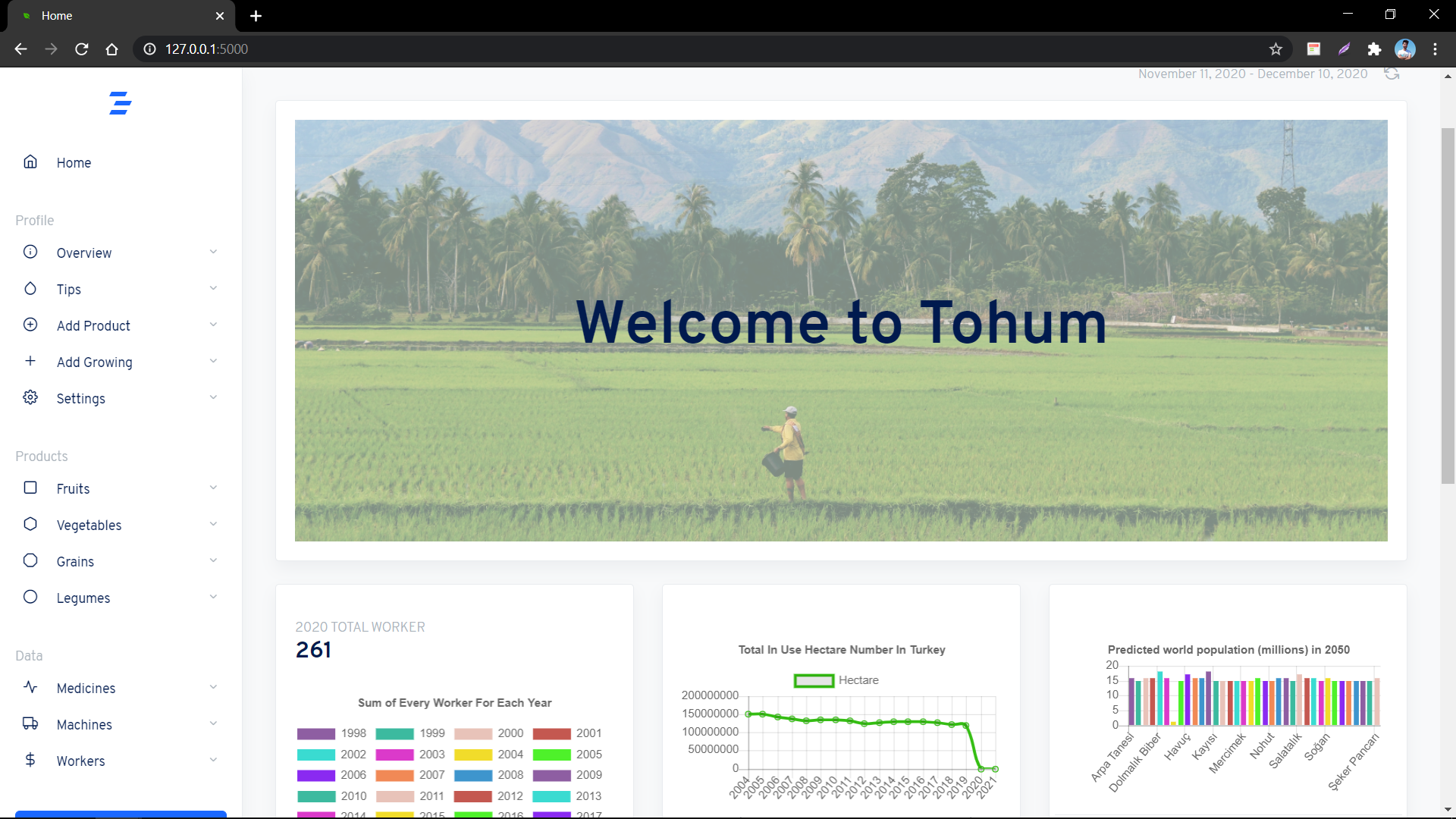
Backend'den Frontend'e Veri Aktarım: SQL üzerinden gönderilen stringler, uygun python dictionary, tuple, list gibi yapılar içerisinde düzenlenerek templateler üzerinden jinja veri yapıları ile uygun formlara tablolara ve grafiklere yerleştirildi.

* 1. Frontend Geliştirme

*3.3.1 Temel Sayfa Yapıları*

Projede hazır bir dasboard template üzerinden tasarımlar yapıldı. Bu template gerekli kısımlar değiştirilerek ve projenin amacına uygun olarak düzenlenerek kullanıldı.

Uygulamayı ilk açtığınızda sizi bir welcome ekranı karşılıyor (Şekil 4.3.1.a). Bu ekrandan giriş yapabileceğiniz veya üye olabileceğiniz ekranlara yönlendiriliyorsunuz.

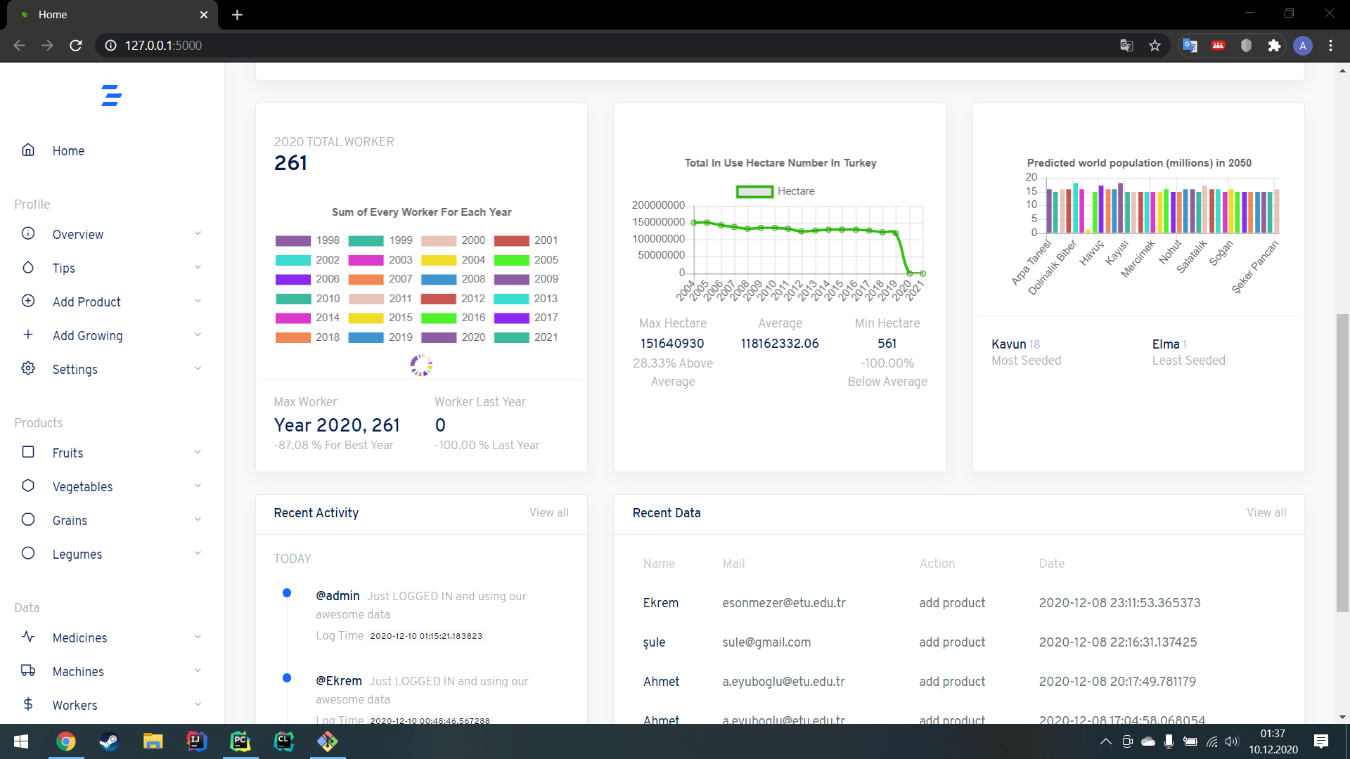


Şekil 4.3.1.a: Giriş Ekranı

Bu aşamadan sonra size üç temel menü sunuluyor. Bunlar Profile, Products ve Data. Profile sayfaları kişiye özel işlemler yapabileceğiniz ekranları içeriyor. Products, farklı ürün kategorileri için isim, yetişme bölgesi, hektar, ton gibi filtreleri ile; hangi ürünlerin kayıtlı olduğunu, yetiştiği alan miktarı ve alınan ürün miktarını size gösteren bir ekranlar sunuyor. Data bölümünde ise ilaç kullanımı, makine kullanımı ve çalışan işçi sayısını yıllara göre grafiksel olarak gösteren ekranlar bulunuyor.

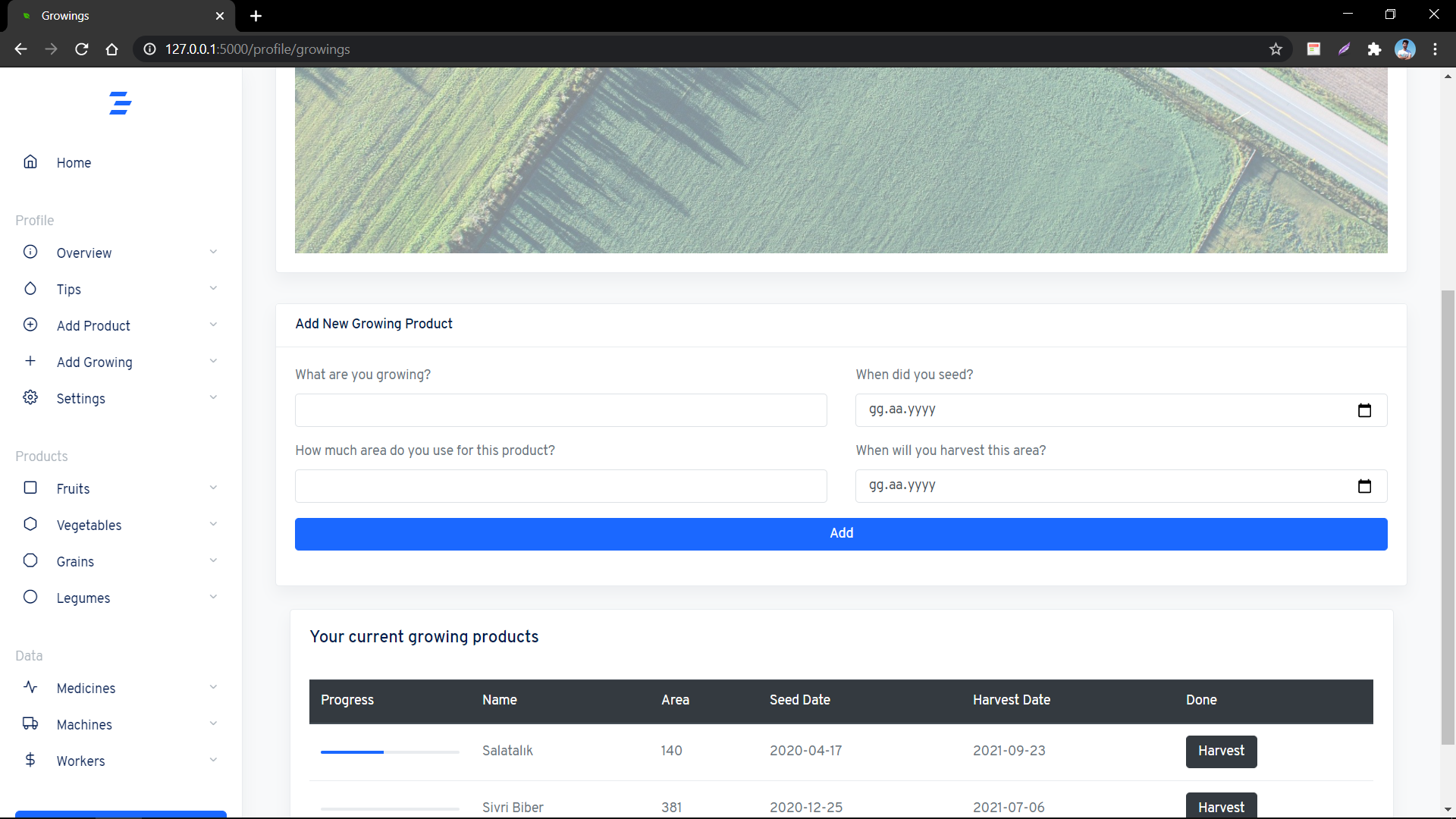
*3.3.2 Aksiyonlar*

Giriş kısmında kullanıcıyı bir Welcome ekranı karşılıyor. Kullanıcı bu ekranda yıllara göre maksimum işçi sayısını, toplam ekilen alanı ve en çok ve en az ekilen ürünü görebiliyor. Ayrıca sistemde diğer çiftçilerin son yaptığı işlemleri ve son yapılan data ekleme işlemlerini de görebiliyor (Şekil 4.3.2.a).



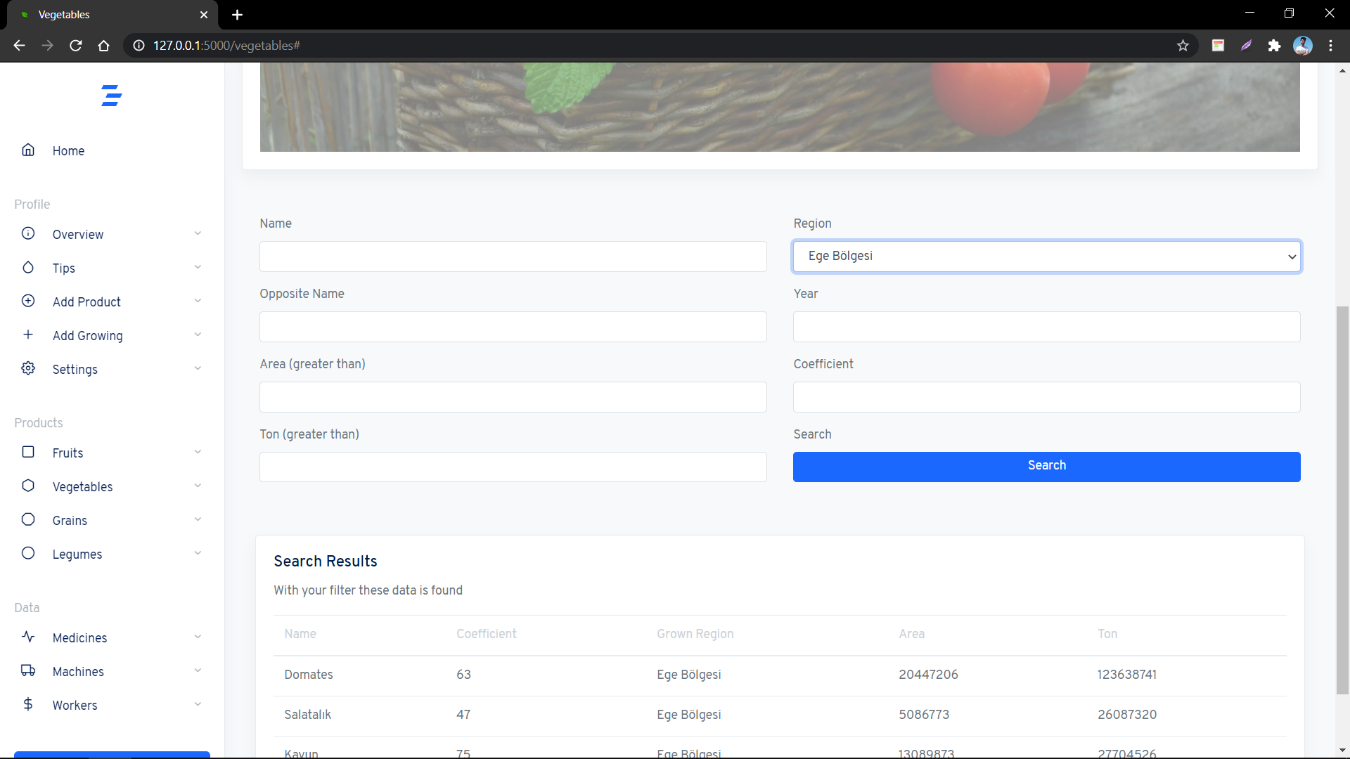
Şekil 4.3.2.a: Bazı İstatistikleri ve Son Aktiviteleri Kullanıcıya Gösteren Welcome Ekranı

Profile kısmında Overview sayfası ile giriş yapmış kullanıcı isim, soy isim ve şehir bilgisi ile yapabileceği işlemler için kısa yollar sunuluyor. Settings bölümünde kullanıcının mevcut kişisel bilgilerini görüntüleyebildiği ve düzenleyebildiği formlar bulunuyor. Kullanıcı buradan isim, mail şehir, şifre gibi bilgilerini güncelleyebilir. Add Product sayfası çiftçinin girdi yapmasını sağlıyor. Buna örnek olarak bir çiftçi o sene yetiştirdiği buğday ürünün sisteme eklemek istediğinde ürün ismi olarak buğday, kaç hektarlık bir alanda yetiştirdiği, kaç ton ürün topladığı ve bu süreçte kullandığı ilaç, makine ve işçi sayısını girerek ekleme yapabilir. Growing ekranı çiftçinin ekmeyi planladığı veya yetişen ürünlerini takip etmesi için tasarlandı. Bu ekranda çiftçi ürün ismini, ne kadar alanda ektiğini, ekme ve toplama tarihlerini girerek ekleme yapabilir (Şekil 4.3.2.b). Daha önce eklemiş olduğu ürünlerin yüzdelik olarak yetişme oranlarına bakabilir. Eğer ürünü toplanma vakti gelmişse buradan açılacak pencere ile topladığı ürünü data kısmına ekleyebilir. Tips ekranı çiftçiye bir yol gösterici olması amacıyla eklendi. Bu ekran son yıl içerisinden en çok toplanan ürünleri çiftçiye göstererek bu ürünlerin oldukça yüksek bir yetişme oranının olduğunu gösteriyor.



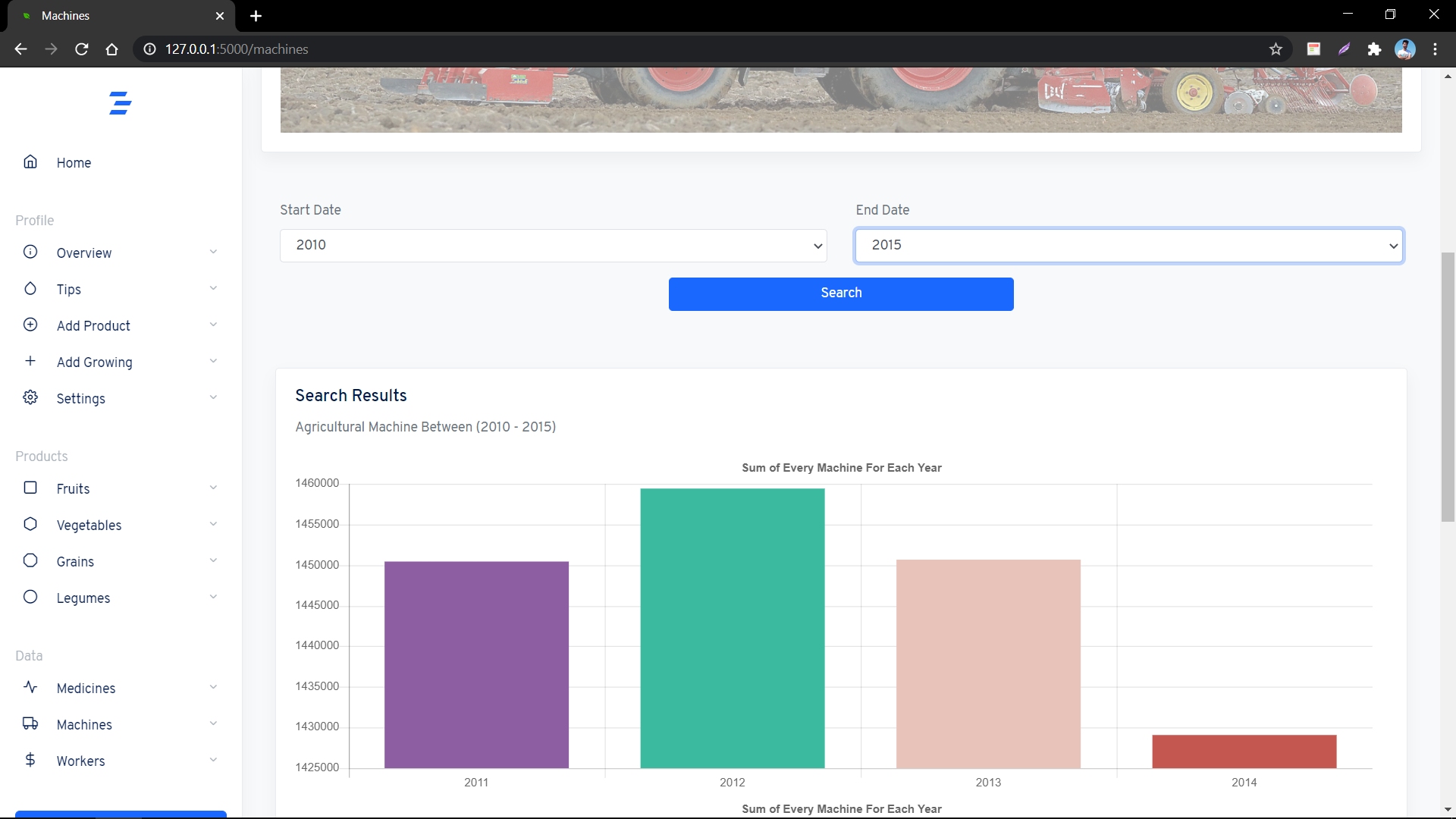
Şekil 4.3.2.b: Salatalık Ekmiş ve Sivri Biber Ekmeyi Planlayan Bir Çiftçinin Growing Ekranı

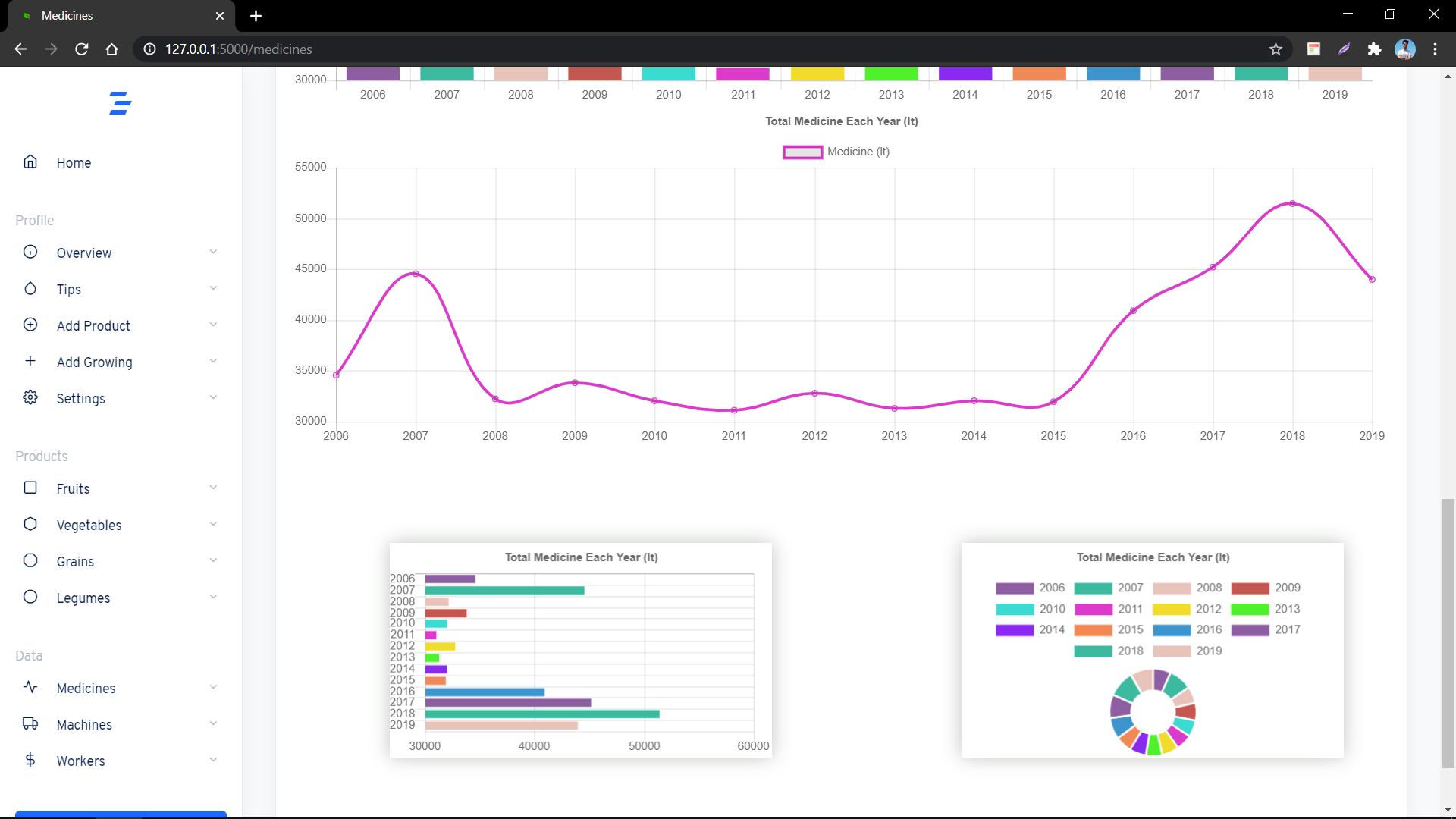
Products ekranlarında bizi bir filtreleme ekranı ve filtrelere göre sonuçlar yer alıyor. Bu ekranlar ürün kategorileri temelinde farklılaşıyorlar. Bu ekranlardan ürün kategorisine girdikten sonra filtreler ile arama yapılabilir. Ürün adı, zıtlaştığı ürün adı, yetişme bölgesi, alan miktarı (girilen alan miktarından daha fazla olanlar), ton miktarı (girilen tondan fazla olanlar), katsayı gibi filtreler kullanıcıya yardımcı olacaktır. Eğer herhangi bir filtreleme kriteri girmeden arama butonuna tıklanırsa o kategoriye ait tüm ürünler gösterilecektir (Şekil 4.3.2.c).



Şekil 4.3.2.c: Ege Bölgesi ile Verilmiş Vegetables Ekranı

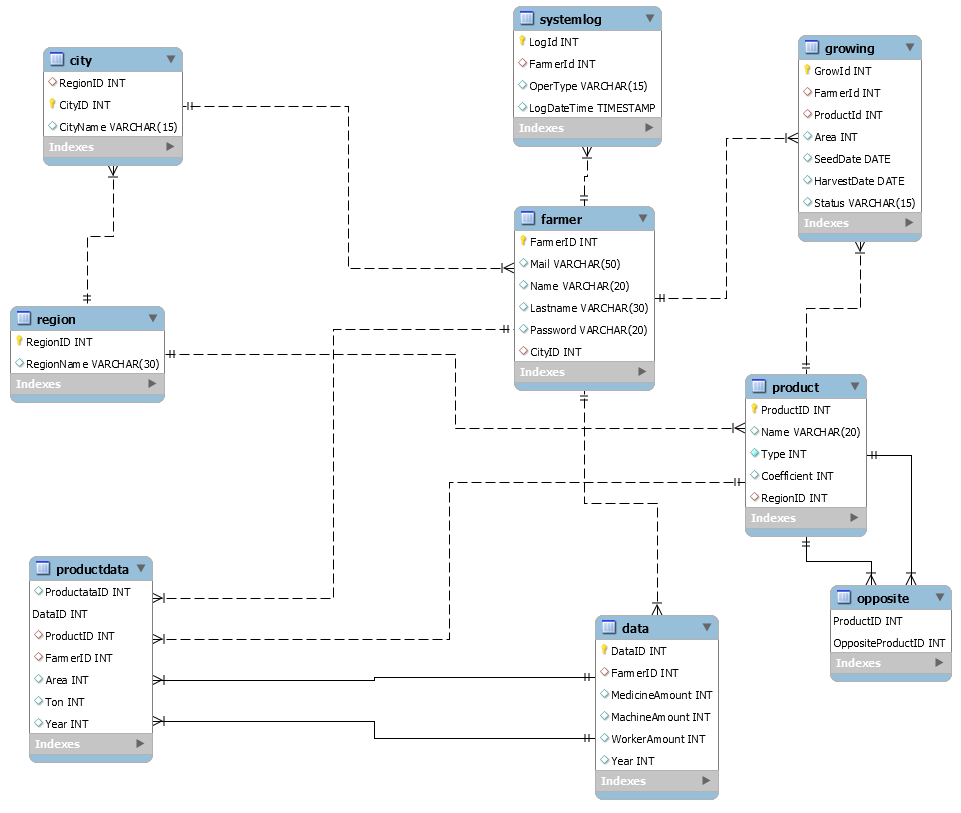
Data bölümü üç temel istatistik için geliştirildi: kullanılan ilaç miktarı, makine sayısı ve çalışan işçi sayısı. Bu ekranlardan yıl aralığı seçerek o yıllara ait verilere erişilebilir. Bu ekranlardaki veriler daha kolay bir anlaşılma sağlanması için grafikler halinde tasarlandı. Grafikler ayrıca interaktif olarak tasarlandı (Şekil 4.3.2.d ve Şekil 4.3.2.e).

Şekil 4.3.2.d: 2011-2014 Arası Makine Kullanımını Gösteren Machines ekranı



Şekil 4.3.2.e: 2006-2019 Arasında İlaç Kullanımını Gösteren Medicines Ekranı

1. Appendix



1. Kaynakça

<https://tuikweb.tuik.gov.tr/PreTabloArama.do?metod=search&araType=vt>

[https://www.tocbirsen.org.tr/uploads/documents/RAKAMLARLA\_TARIM\_SEKTÖRÜ\_WEB-min.pdf](https://www.tocbirsen.org.tr/uploads/documents/RAKAMLARLA_TARIM_SEKT%C3%96R%C3%9C_WEB-min.pdf)