**Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulmak**

### **Github Adresi**

<https://github.com/ahmetirmak06/PythonInPython>

### **Grubu Oluşturan Öğretmenlerin Listesi**

Ahmet IRMAK, Alp Eren DEMİRKAN, Bekir PALAZ, Binnur SANCAK PALAZ, Emrah ŞEKER, Murat KOÇALİ

# **Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulmak Analiz Raporu**

### **Kısa Özet**

Yapay Zekâ çalışmalarının içinde önemli bir yere sahip olan sezgisel algoritmalar yardımıyla, birçok gömülü sistem içinde küçük ama önemli bir parça olarak yer alan denklem kökü bulma/değişken değeri hesaplama işlevi için, yeni öğrenmekte olduğumuz Python dilini kullanmakta kendimizi geliştirmek istiyoruz. Bu şekilde Python kütüphanesinde yer alan modüllerden yararlanmayı öğrenmek ve kendimizin ürettiği alternatif bir modül de sunarak yeni bir modül kazandırmak şeklinde bir amacı takip edeceğiz.

### **Problem Tanımı**

Python dili, yapay zekâ çalışmalarında, özellikle de sezgisel algoritmalar tarafında hangi paket/modül kütüphanelerine sahiptir? Bunlar nasıl kullanılır? Bu kütüphanelere temel işlevler açısından kullanışlı bir modül nasıl eklenir? Açık kaynak kodlu bir dil olarak Python’ un diğer dillere göre geliştirme süreci/kodlama/performans açılarından nasıl bir üstünlüğü vardır?

### **Analiz Süreci**

#### **İhtiyaç Analizi**

Başta gömülü sistemler olmak üzere yapay zeka tekniklerinin kullanıldığı yazılımlar günümüzde hem hızla gelişmekte hem de sistem performansları açısında önemli bir yeri vardır. Bu tür yazılımların kullanıldığı sistemlerde dışarıdan girilen/okunan bir takım veriler üzerinden bazı değişkenleri tekrar hesaplamak, “an” seviyesinde kısa zamanlar içerisinde buna uygun çıktılar üretmek sıkça kullanılan bir işlemdir. Python’ un yeni ve gelişime açık bir yazılım dili olarak, bu tür temel işlevlere yüksek performanslı bir katkı sağlaması beklenir. Bu anlamda hangi kütüphanelerle çalışılabilir, geliştirdiğimiz kütüphaneleri nasıl kullanıma sunabiliriz, geçmişteki kodlamalara göre nasıl bir performans iyileştirmesi gerçekleştirebiliriz? Bu alanda çalışanlar açısından daha iyi performanslı bir algoritma geliştirmek önemli bir hedeftir. Grup olarak geliştireceğimiz bir yazılım ile konuyu farklı bir açıdan ele alacağız.

#### **İçerik Analizi**

Projemiz için oluşturduğumuz grubumuz, whatsapp üzerinden iletişim sağlarken, aynı zamanda video konferans araçları ile de bir araya gelecektir. Github hesabındaki paylaşım ile de projemizin hangi aşamada olduğu takip edilebilecektir.

Çalışma sonunda ortaya çıkacak ürünümüz, Python’un paylaşım platformlarında geliştirilmeye açık bir kütüphanenin başlangıç noktası olarak paylaşarak, proje bu aşama için sonlandırılacaktır.

#### **Durum Ortam Analizi**

Arayüz tasarımı için *PyQt5*, *Tkinter*, gerekirse veri tabanı için *Sqlite3*, temel işlemler için *sys* ve *os*, ileri düzey matematiksel işlev gerekmesi durumunda *Numpy* kullanılacaktır.

#### **Kullanıcı Analizi**

Bu proje kapsamında geliştireceğimiz yazılım, GitHub’a yüklenerek, özellikle yapay zeka alanında çalışan akademisyen ve yazılımcıların ulaşımına sunulacaktır. Umarız gelişen bir kütüphanenin başlangıcı veya gelişmekte olan bir kütüphanenin parçası olur ve çok sayıdaki projeye katkı sunar.

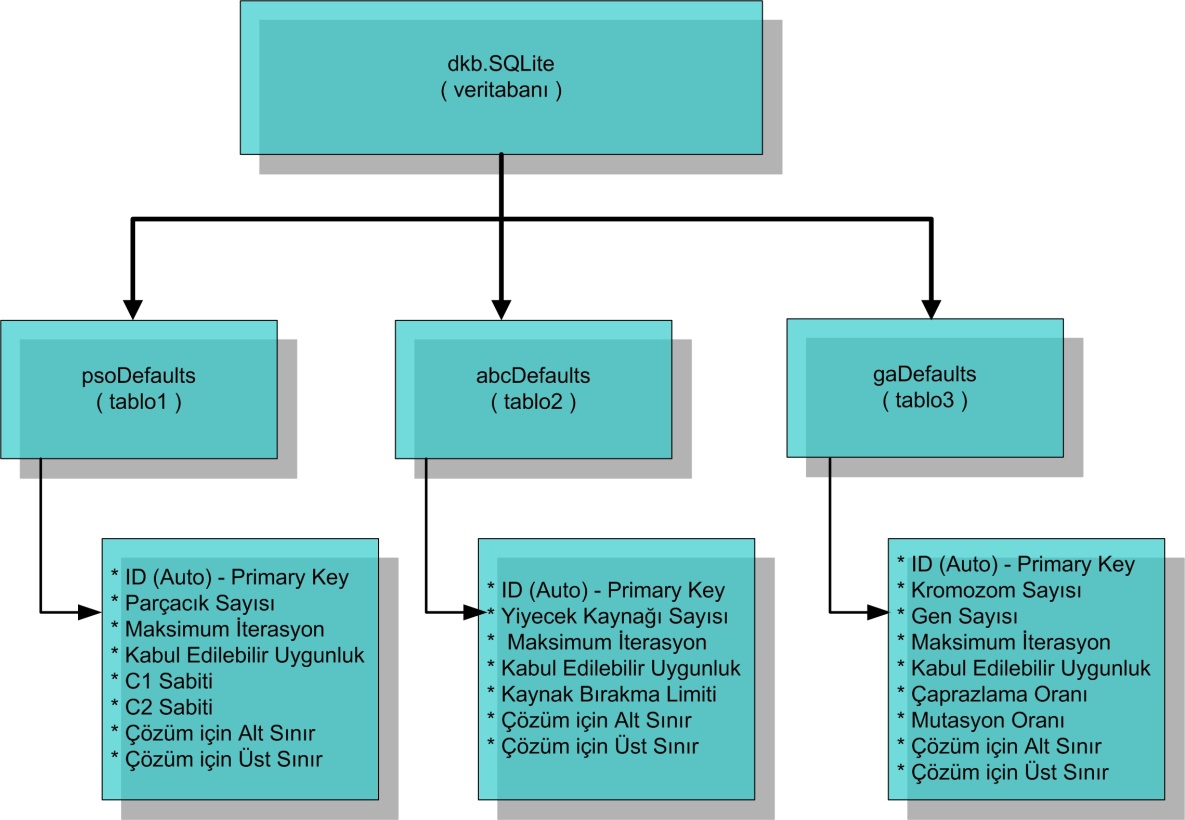
# **Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulmak Tasarım Raporu**

### **Kısa Özet**

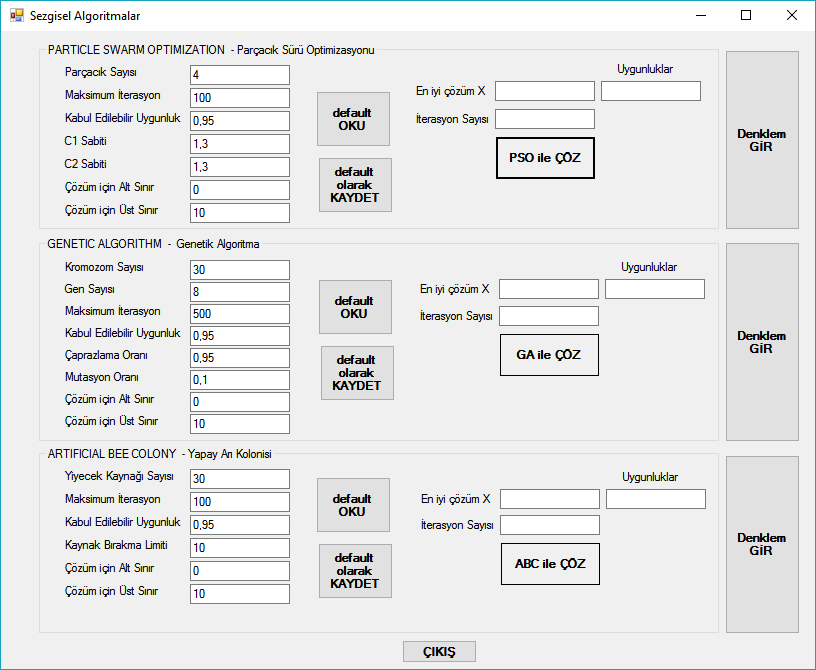
* Grup üyelerinin iletişimi için anlık mesajlaşma grubu kuruldu,
* Proje olarak ne yapılabilir konusunda beyin fırtınası yapıldı, projeye karar verildi,
* Projenin yapım aşamaları konusunda fikir alışverişinde bulunuldu,
* Grup üyeleri arasında görev paylaşımı yapıldı,
* Sürece göre zaman planlaması yapıldı,

|  |  |
| --- | --- |
| **Üye Adı** | **Görevi** |
| Ahmet IRMAK | Sezgisel Algoritma modüllerinin yazılması |
| Alp Eren DEMİRKAN | Veritabanı okuma yazma modüllerinin yazılması |
| Bekir PALAZ | Veritabanının tasarımı ve oluşturulması. **Numpy** kütüphanesinin araştırılması |
| Binnur SANCAK PALAZ | Veritabanının tasarımı ve oluşturulması. **Numpy** kütüphanesinin araştırılması |
| Emrah ŞEKER | Denklem giriş ara yüzüne ait modülün yazılması |
| Murat KOÇALİ | Genel ara yüzün tasarımı. Python’ da geliştirilmiş diğer yapay zekâ alanıyla ilgili kütüphanelerinin araştırılması |

### **Veri Tasarımı**



### **Ara** **yüz Tasarımı**



## **Kod Tasarımı**

## 

## **Zaman Çizelgesi**

|  |  |
| --- | --- |
| 19/11/2020 | Grup üyeleri ile iletişim için anlık mesajlaşma grubunun kurulması |
| 20/11/2020 | Proje konusunun araştırılması |
| 25/11/2020 | Proje konusunun tartışılması ve karar aşaması |
| 27/11/2020 | İş bölümü yapılması |
| 06/12/2020 | Projenin tasarlanması |
| 13/12/2020 | Projenin gerçekleştirilmesi ve test edilmesi |
| 20/12/2020 | Github’a yükleme, rapor teslimi ve sunum |

## **Sezgisel Algoritmalar Yardımıyla Denklem Kökü Gerçekleştirme Raporu**

## **Karşılaşılan Sorunlar ve Uygulanan Çözümler**

## Proje aşamasında karşılaşılan pek çok problem ile ilgili internet üzerinden birçok kaynak taranarak çözüme ulaşılmaya çalışıldı. Ulaşılamayan yerlerde grup üyeleri ile eşgüdüm sağlandı.

**Proje Bileşenleri ve Görevleri**

Ortam Phyton 3.5, Anaconda, PyCharm, SQLite3 veritabanı kullanıldı. SQLite browser, wt designer tasarımlar için kullanıldı. Kütüphaneler; math, pyqt5, Os kullanıldı.

Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulma programımızın kullanıcı arayüzü

* Çözüm Ekranı (PSO/GA/ABC)
* Denklem giriş ekranı
* Veritabanı kayıt ekleme/silme ekranı

|  |  |
| --- | --- |
|  |  |

**Github Yükleme Süreci**

Grup üyeleri oluşturdukları github hesapları ile 1 repostory 1 project (dkb) üzerinden yazılım kaynak kodunu birbirleri ile paylaşarak geliştirmektedir. Eğitmenlerce duyurulacak hesaplarla da paylaşım yapılacaktır.

## **Sezgisel Algoritmalar Yardımıyla Denklem Kökü Test Raporu**

## **Karşılaşılan Sorunlar ve Uygulanan Çözümler**

## Proje aşamasında karşılaşılan problemlerle ilgili internet üzerinden birçok kaynak taranarak çözüme ulaşılmaya çalışıldı. Ulaşılamayan yerlerde grup üyeleri ile eşgüdüm sağlandı. Önceki programlama dillerinden farklı yazım kuralları sebebiyle alışma döneminde güçlük çekilmiştir (İleri ket vurma). Özellikle grafik ara yüzü ile tasarım yapılmasında, program ara yüzüne eklemek istediğimiz özellikler ile ilgili yazılan kod çalışmamış; ardından wtdesigner kullanılarak aynı kodlar yazılmış ve çalışmıştır. Yaşanan zorluklar bu tip hata ayıklamalar üzerinde odaklanmıştır.

## **Test Sürecinde Kullanılan Modüller (Varsa)**

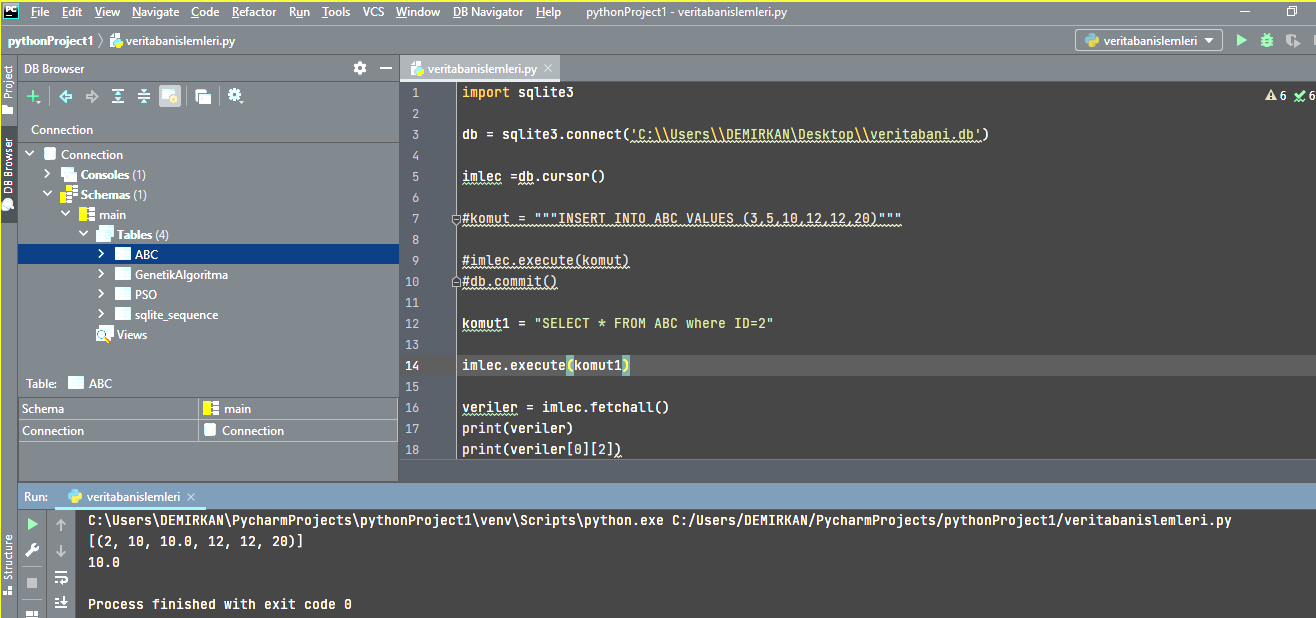
Proje test sürecinde ekstra herhangi bir modül kullanılmamıştır. Bu raporun teslimi tarihi itibariyle eksikler tamamlanmaya çalışılmaktadır. Proje son aşamasına getirilemediğinden, test aşaması henüz yapılamamıştır. Test aşaması tamamlandığında, nihai raporda bu kısma yer verilecektir.

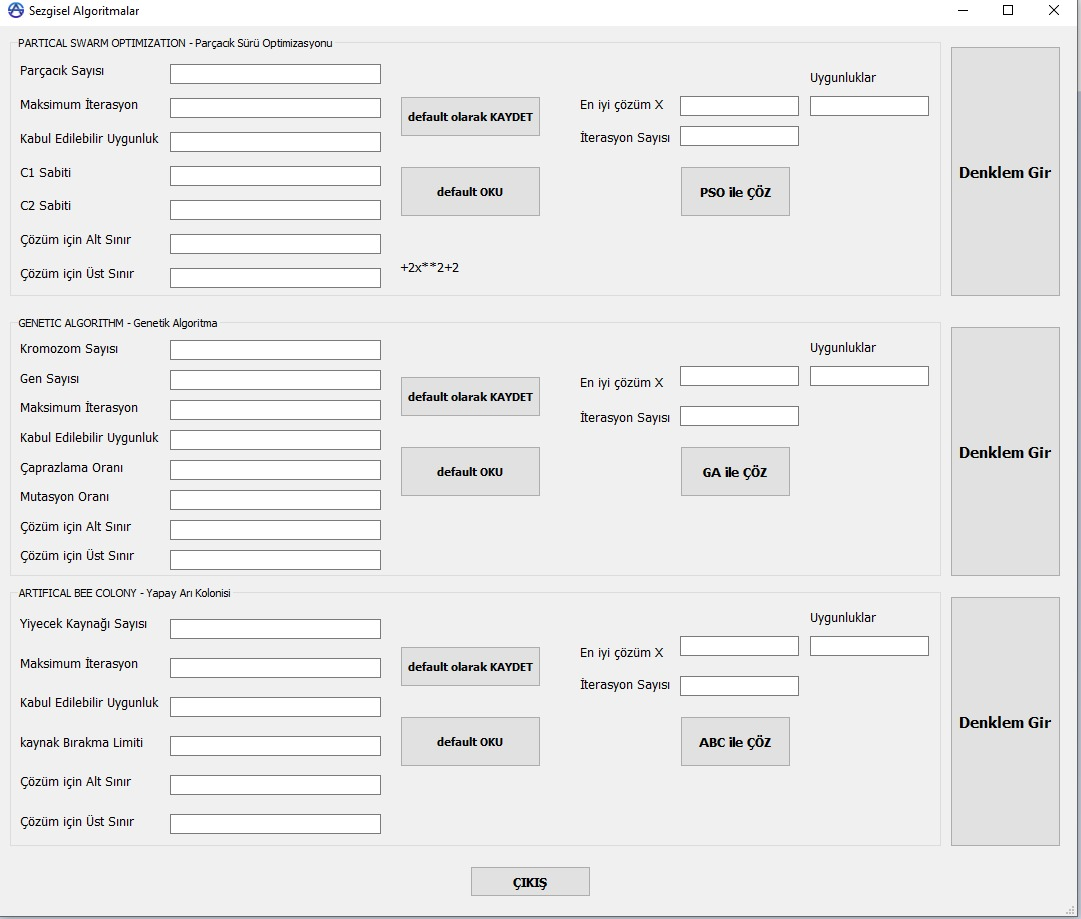
Ortam Phyton 3.5, Anaconda, PyCharm, SQLite3 veritabanı kullanıldı. SQLite browser, wt designer tasarımlar için kullanıldı. Kütüphaneler; math, pyqt5, os, random kullanıldı.

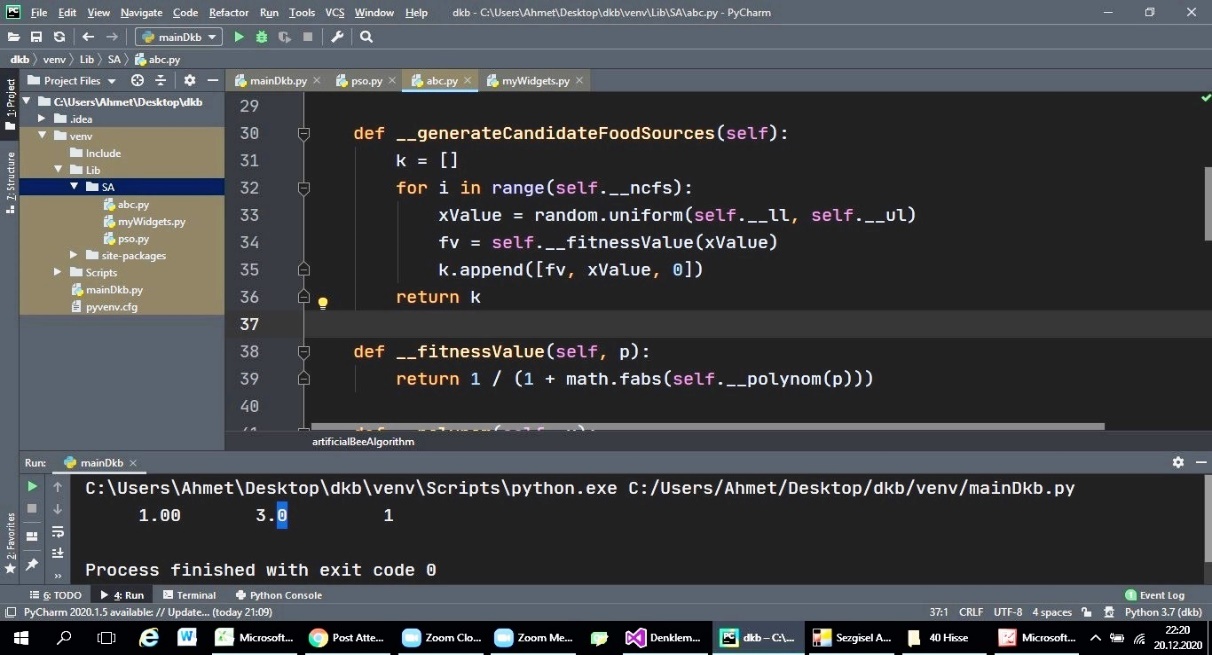
Sezgisel Algoritmalar Yardımıyla Denklem Kökü Bulma programımızın kullanıcı ara yüzü

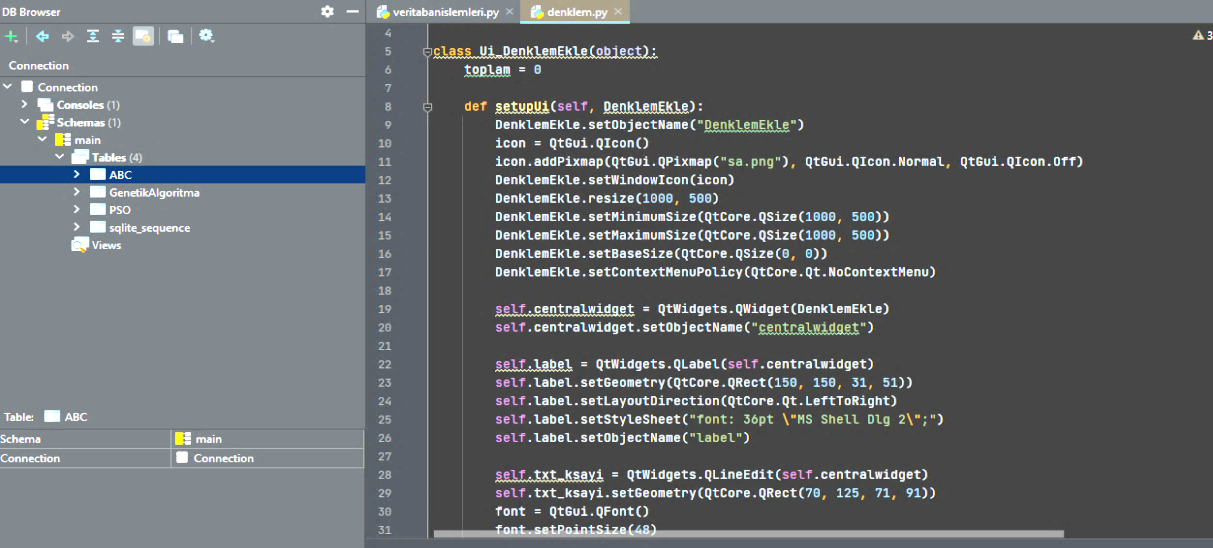
* Çözüm Ekranı (PSO/GA/ABC)
* Denklem giriş ekranı
* Veritabanı kayıt ekleme/silme ekranı

Yaptığımız projenin ekran görüntüleri ve kodlamalar:









## Değerlendirme Kriterleri

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **KRİTERLER** | Ekip Üyeleri | Eğitmen 1 | Eğitmen 2 | Eğitmen 3 | Ortalama |
| 1. Analiz Raporunun Tamamlanması |  |  |  |  |  |
| 1. Tasarım Raporunun Tamamlanması |  |  |  |  |  |
| 1. Gerçekleştirim Raporunun Tamamlanması |  |  |  |  |  |
| 1. Gantt Diagramı |  |  |  |  |  |
| 1. Arayüz tasarımı |  |  |  |  |  |
| 1. Veri Tasarımı-Sınıf Tasarımı |  |  |  |  |  |
| 1. Kullanıcı Yardım Dökümanı |  |  |  |  |  |
| 1. Programın Çalıştırılması |  |  |  |  |  |
| 1. Yazılım Test Çalışması |  |  |  |  |  |