# Demirbaş Yönetim Sistemi

### Github Adresi

https://github.com/faytoncular/demirbas\_yonetim

### Grubu oluşturan öğretmenlerin listesi

Hasan ÖZÇETİN, İsmail GÖVERCİN, Selahattin ÇEKCEN

# Demirbaş Yönetim Sistemi Analiz Raporu

### Kısa Özet

### MEB’e bağlı kurumlarda demirbaş yönetiminde karşılaşılan karışıklık ve düzensizliğinin ortadan kaldırılması amacıyla tasarlanmıştır. Kullanımı kolay, her kullanıcı tarafından kolaylıkla anlaşılabilen ve etkileşime izin veren bir görsel tasarım kullanılmıştır. Demirbaşların sistematik olarak kayıt altına alınması, listelenmesi, yönetilmesi ve kullanıcılarına zimmetlenmesi gibi işlevlerin kolayca yerine getirilmesi hedeflenmiştir.

### Problem Tanımı

Farklı kurum veya anonim oluşumların tek bir yazılım üzerinden demirbaş kayıtlarının yönetim, kontrol ve transferleri gibi işlemleri hızlı ve sorunsuz bir şekilde kontrol etmesi ve yönetmesi, gerektiğinde özelleştirilmiş dökümlere ulaşması mümkün müdür?

Başarılı bir veri tabanı yönetimi ve tüm ihtiyaçların planlandığı bir ara yüz birçok kitlenin hedeflenen alanda talebine cevap verecektir. Ürünün amaçlananı vermesi için tüm ihtiyaçların belirlenmesi ve istenilen sonuca ulaşılacak ara yüz yapısının ortaya konulması sonuca ulaşılması için en önemli adımları oluşturacaktır.

### Analiz Süreci

Geliştirme ortamı olarak Python 3.8, PyCharm 2020 ve SQLite3 veritabanı kullanıldı. Bunlar dışında QT Designer, SQLite Browser vb. çeşitli 3. parti yazılımlardan gerekli hallerde faydalanıldı.

#### İhtiyaç Analizi

En üst seviyede ara yüz kullanım kolaylığı sunan, kullanıcı ihtiyaçlarını tam anlamıyla karşılayabilecek düzeyde ve verimli bir yazılım geliştirilmesi hedeflendi.

#### İçerik Analizi

Projenin Ana Bölüm Tanımları: Projemiz,

* Demirbaş İşlemleri
  + Demirbaş Ekleme/Düzenleme/Listeleme/Silme
* Kişiler
  + Kişi Ekleme/Düzenleme/Listeleme/Silme
* Kategoriler
  + Kategori Ekleme/Düzenleme/Listeleme/Silme
* Yedekleme/Yedek Geri Yükleme

bölümlerinden oluşmaktadır.

#### Durum Ortam Analizi

Geliştirme ortamı olarak Python 3.8, PyCharm 2020 ve SQLite3 veritabanı kullanıldı. Bunlar dışında QT Designer, SQLite Browser vb. çeşitli 3. parti yazılımlardan gerekli hallerde faydalanıldı. Kullanılan temel Python kütüphaneleri: PyQt5 modülü, Sqlite3 modülü, sys modülü, os modülü, datetime modülü, pickle modülü.

#### Kullanıcı Analizi

Süreç sonunda hazırlanacak yazılım, Demirbaş Yönetim Sistemine ihtiyaç duyabilecek tüm kullanıcılar tarafından kendi ihtiyaçları doğrultusunda kullanılabilecektir.

# Demirbaş Yönetim Sistemi Tasarım Raporu

### Kısa Özet

Ara yüzün kullanımı kolay ihtiyaç duyan tüm kullanıcıların rahat bir şekilde işlemlerini yapabileceği kolaylıkta olması hedeflendi.

Tasarım ara yüzü çalışmalarına başlamadan önce ara yüz planlamasının esnek ve sürece göre ana hedefleri koruyarak şekillendirilebilir olmasına özen gösterildi.

Veri tabanının minimum yük taşıması, tekrarlayan bilgileri içermeyecek yapıda olması ve ilişki yapısının en verimli şekilde kullanıma uygun olması gibi detaylara dikkat ederek planlamaları ve şeması oluşturuldu.

Ara yüzler ve veri tabanı şablonları oluşturulurken kullanıcının hata payının en aza inmesi, girdi ve çıktıları hızla düzenleyip raporlara kolayca ulaşabilmesi için kullanılabilirlik en üst düzeyde tutulmaya çalışıldı.

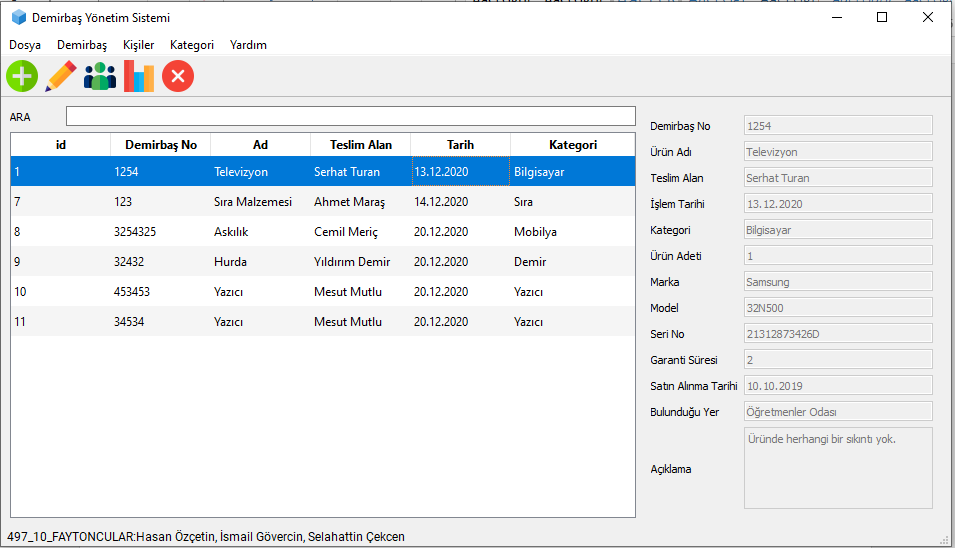
## Veri Tasarımı

Veri tabanı şeması görsel 1 de yer alan şekilde oluşturuldu.

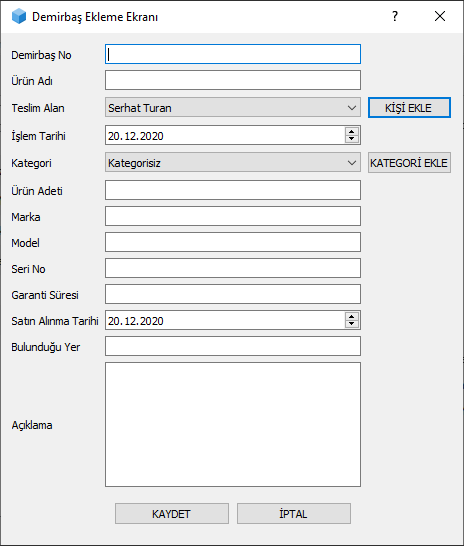
|  |  |  |
| --- | --- | --- |
| Demirbaş Tablosu | Kategori Tablosu | Kişiler Tablosu |
|  |  |  |

Görsel 1

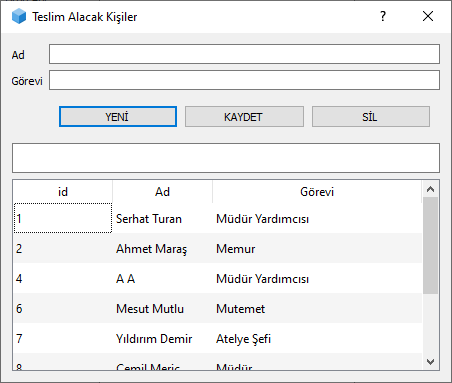
## Ara yüz Tasarımı



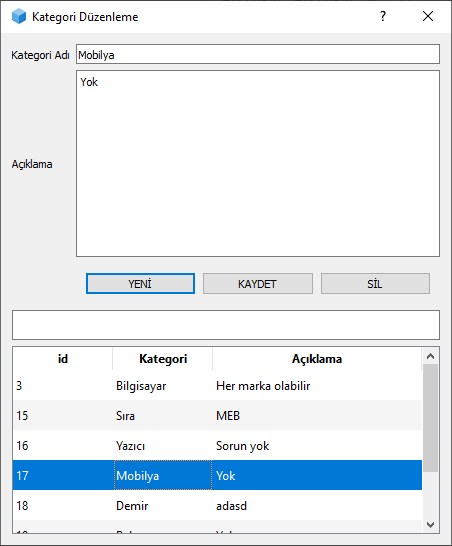
Program ana ekranı üç bölümden oluşmaktadır. Üst tarafta bulunan **Toolbar** ile sırasıyla *Demirbaş Ekle, Demirbaş Düzenle, Kişiler, Kategoriler ve Demirbaş Sil* işlemleri yapılmaktadır. Alt sol bölümde tüm demirbaşların listelendiği bir **QTableWidget** bileşeni ve üstünde yazılan her şeyi hemen filtreleyen arama bölümü vardır. Sola tarafta ise tabloya sığmayan diğer detay bilgilerin bulunduğu kısım yer almaktadır.



Demirbaş Ekleme Ekranı



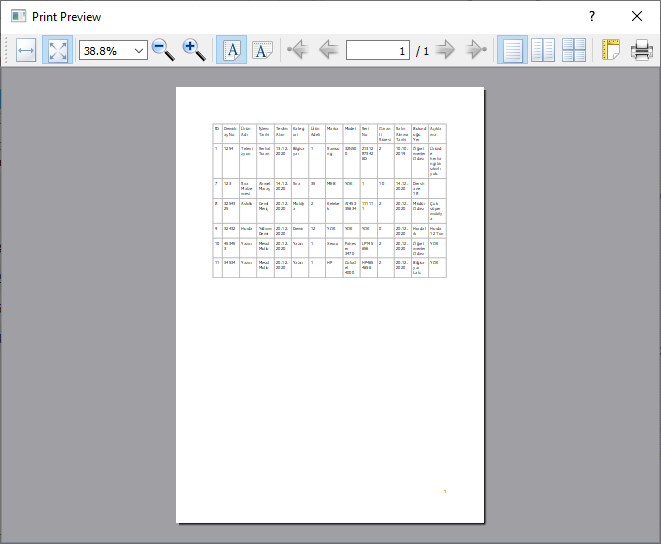
Teslim alan kişilerin yönetim ekranı



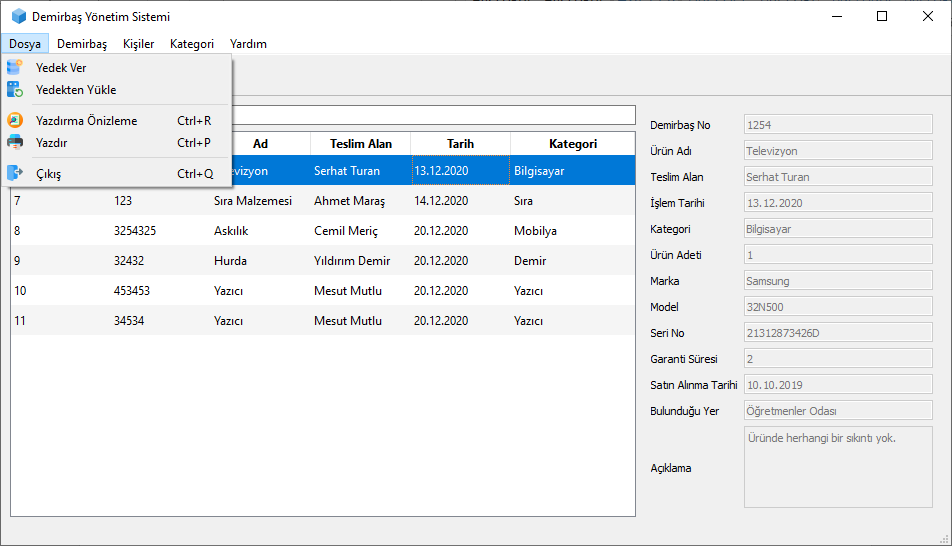
Kategori yönetim bölümü



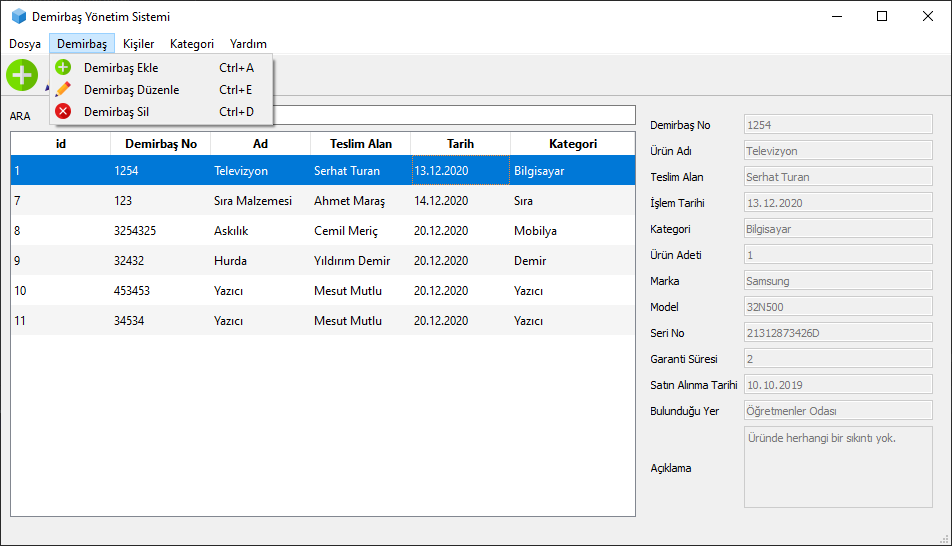
Hakkında penceresi



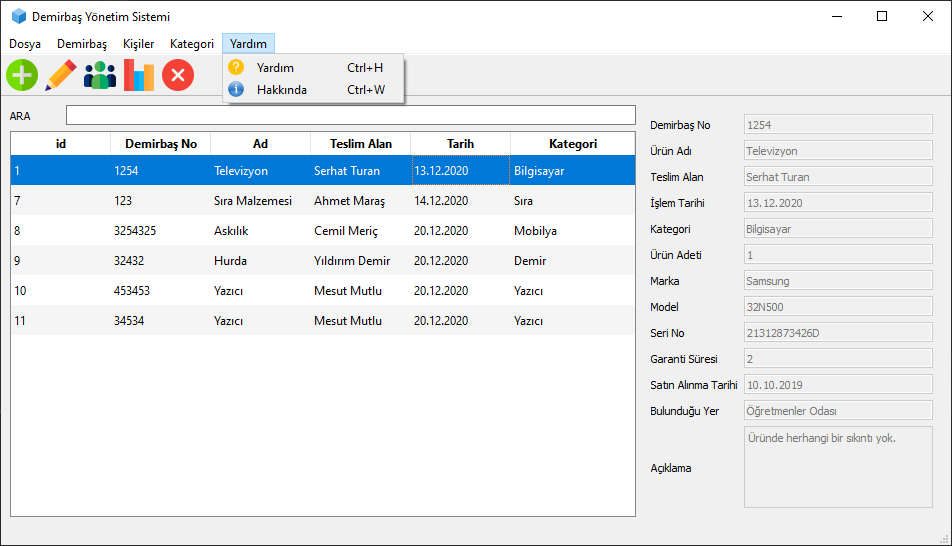
Yazdırma önizlemesi



Dosya menüsü

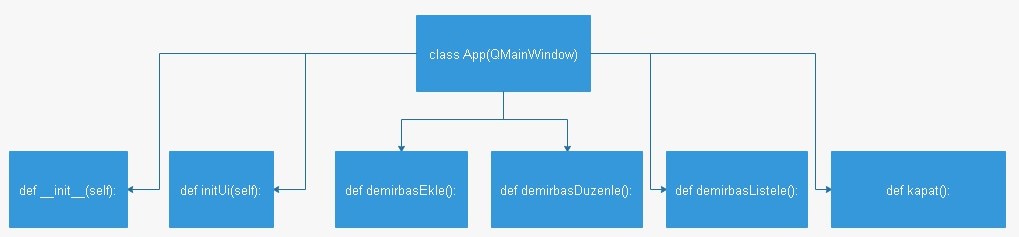


Demirbaş menüsü



Yardım menüsü

## Kod Tasarımı



Görsel 6

## Zaman Çizelgesi

|  |  |
| --- | --- |
| 01-12.2020 - 06.12.2020 | Proje iş bölümü ve veritabanı ve arayüz tasarımının tamamlanması |
| 07-12.2020 - 09.12.2020 | Demirbaş ekle- Demirbaş Düzenle işlemleri |
| 10-12.2020 - 11.12.2020 | Demirbaş Listele Kısmının tamamlanması |
| 12-12.2020 - 13.12.2020 | Eksikliklerin gözden geçirilmesi ve ilktest süreci |
| 14-12.2020 - 19.12.2020 | Github’a yükleme işlemleri, test sürecinin tamamlanması |
| 20-12.2020 - 20.12.2020 | Projenin tamamlanması ve sunum |

# Demirbaş Yönetim Sistemi Gerçekleştirme Raporu

## Karşılaşılan Sorunlar ve Uygulanan Çözümler

Projenin bazı bölümleri ayrı ayrı yazıldı. Ayrı modüllerde özellikle PyQt5 kütüphanesinin çok çeşitli bileşenlerinden bazıları kullanıldı. Birleştirme esnasında bazı bileşenler birbiri ile uyumsuzluk çıkardı. Bu sorunlar birer birer tespit edilerek gerekli dönüşümler yapılarak uyumlu bir şekilde çalışması sağlandı.

Proje çalışmalarında Sqlite3 kullanıldı. Sqlite3 ile direkt SQL kodları kullanılarak çalışılıyor. Bazı tablolarımızda çok uzun SQL ifadelerinin olması küçük yazım hatalarına neden oldu. Bu hataları tespit etmek oldukça güç. 3.parti bir veri tabanı yazılımı ile SQL ifadeleri oluşturularak sqlite3 ile sorunsuz bir şekilde çalışması sağlandı.

## Proje Bileşenleri ve Görevleri

Projede kullanılan kütüphaneler şunlar

* PyQt5 modülü
* Sqlite3 modülü
* sys modülü
* os modülü
* datetime modülü
* pickle modülü

Proje bir demirbaş uygulaması olduğu için üç ayrı bölümden oluşmaktadır.

1. Demirbaş ürünler
2. Teslim edilen kişiler
3. Demirbaş kategorileri

Her bölüm için ayrı ayrı kayıt, ekleme, silme ve düzeltme bölümleri mevcuttur.

## Github Yükleme Süreci

Projenin Github adresi <https://github.com/faytoncular/demirbas_yonetim> ‘dir. Projenin ilk yüklemesi bir hesap üzerinden yapılacaktır. Sonraki zamanlarda diğer proje ekipleri gerekli katkıyı kendi hesapları ile yapacaktır.

# Demirbaş Yönetim Sistemi Test Raporu

## Karşılaşılan Sorunlar ve Uygulanan Çözümler

Yazılım geliştirilirken karşılaştığımız en büyük sorun Python dilinin bir yorumlayıcı bir dil olmasından kaynaklandı. İşlem görmeyen satırlarda oluşan hatalar karşısında Python yorumlayıcısının herhangi bir hata vermemesi sorunların çözümünü oldukça güçleştirdi. Genellikle satır satır analiz yapılarak hatalar bulundu. Bu şekilde çözüm çok zaman aldı.

## Test Sürecinde Kullanılan Modüller (Varsa)

Proje aşamasında herhangi bir otomatik test aracı kullanılmadı. Ekip üyeleri manuel olarak hata taraması yaparak gerekli yerlerden düzenlemeler yaptılar. Hatalar ve düzenlemeler projenin Github sayfasında görüntülenebilir.

## Değerlendirme Kriterleri

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **KRİTERLER** | Ekip Üyeleri | Eğitmen 1 | Eğitmen 2 | Eğitmen 3 | Ortalama |
| 1. Analiz Raporunun Tamamlanması | Hasan ÖZÇETİN |  |  |  |  |
| 1. Tasarım Raporunun Tamamlanması | Hasan ÖZÇETİN |  |  |  |  |
| 1. Gerçekleştirim Raporunun Tamamlanması | İsmail GÖVERCİN |  |  |  |  |
| 1. Gantt Diagramı | Selahattim ÇEKCEN |  |  |  |  |
| 1. Arayüz tasarımı | Tüm Ekip |  |  |  |  |
| 1. Veri Tasarımı-Sınıf Tasarımı | Tüm Ekip |  |  |  |  |
| 1. Kullanıcı Yardım Dökümanı | Selahattin ÇEKCEN |  |  |  |  |
| 1. Programın Çalıştırılması | İsmail GÖVERCİN |  |  |  |  |
| 1. Yazılım Test Çalışması | Tüm EKip |  |  |  |  |