



Ondokuz Mayıs Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü

Yazılım Mühendisliği Laboratuvarı (SL 2) (22-23B)

Üniversite Kütüphane Yönetim Sistemi

Yazılım Tanıtım

Muhammed Emirhan Türkmen

20060760

TASARIM MODELİ

Bu projede Model Görüş Denetleyici MVC (Model-View-Controller) deseni kullanılmaktadır. MCV deseni uygulamayı veri (Model), kullanıcı arayüzü (View) ve işlem yönetimi (Controller) bileşenlerine ayırır. Bu, kodun modüler ve sürdürülebilir olmasını sağlar. Model, uygulamanın verilerini ve iş mantığını yönetirken, View kullanıcının arayüzü ile ilgilenir ve Controller, kullanıcının girişlerini işler ve gerektiğinde Model ve View arasındaki iletişimi sağlar.

Bu deseni tercih etmek için birçok sebep var bunlar şu şekilde sıralanabilir:

1. **Ayrı Sorumluluklar:** MVC deseni, uygulamayı farklı sorumluluklara ayırır. Model, verileri ve iş mantığını yönetirken, View kullanıcı arayüzünü temsil eder ve Controller kullanıcının girişlerini işler. Bu şekilde, her bileşenin kendine özgü bir görevi vardır ve her biri birbirinden bağımsız olarak geliştirilebilir, değiştirilebilir veya test edilebilir.
2. **Modülerlik ve Sürdürülebilirlik:** MVC deseni, kodun modüler ve sürdürülebilir olmasını sağlar. Model, veri işleme ve iş mantığı kodlarını içerir, View kullanıcı arayüzüne odaklanır ve Controller kullanıcının girişlerini işler. Bu sayede, her bir bileşeni bağımsız olarak geliştirebilir, değiştirebilir ve test edebilirsiniz. Ayrıca, bir bileşende yapılacak değişiklikler diğer bileşenlere minimum etki eder, bu da kodun daha kolay bakımını ve genişletilebilirliğini sağlar.
3. **Test Edilebilirlik:** MVC deseni, bileşenlerin bağımsız olarak test edilebilmesini sağlar. Model, iş mantığını içerdiği için bu bileşenin test edilmesi daha kolaydır. View, kullanıcı arayüzünü temsil ettiği için manuel testlere daha uygunken, Controller kullanıcının girişlerini işlediği için birim testlerle test edilebilir. Bu, kodunun daha güvenilir ve hatasız olmasını sağlar.
4. **Kullanıcı Arayüzünün Bağımsızlığı:** MVC deseni, kullanıcı arayüzünün (View) bağımsız olarak değiştirilmesini sağlar. Yeni bir arayüz tasarlandığında veya mevcut arayüzde değişiklikler yapıldığında, bu değişiklikler sadece View bileşenini etkiler ve diğer bileşenlere minimum etki eder. Bu, uygulamanın kullanıcı arayüzünü geliştirme sürecini kolaylaştırır.
5. **Paralel Geliştirme:** MVC deseni, Model, View ve Controller bileşenlerinin bağımsız olarak geliştirilebilmesini sağlar. Bu, farklı ekiplerin veya geliştiricilerin aynı anda farklı bileşenler üzerinde çalışabilmesini mümkün kılar. Bu şekilde, projenin geliştirme süresi kısılır ve paralel geliştirme avantajı elde edilir.

MCV Modelinin bazı dezavantajları vardır ancak bu sorunlar artıları düşünüldüğünde göze alınabilir eksilerdir:

1. **Karmaşıklık:** MVC deseni, bileşenlerin ayrılmasını ve sorumlulukların dağıtılmasını sağlar. Bu nedenle, daha küçük ve bağımsız bileşenlerin koordinasyonu gerekmektedir. Bu durum, tasarımın karmaşık hale gelmesine ve kodun anlaşılabilirliğinin azalmasına neden olabilir.
2. **Büyüme ve Karmaşıklık Yönetimi:** Büyük ve karmaşık projelerde, MVC deseninin yönetimi zorlaşabilir. Bileşenler arasındaki etkileşimlerin takibi, bağımlılıkların yönetimi ve koordinasyon gereksinimleri projenin büyüdükçe artabilir.

Bu eksileri önlemek için alınabilecek önlemler şu şekildedir:

1. **İyi Bir Tasarım Planı:** Projenin başlangıcında, bileşenler arasındaki ilişkileri ve sorumlulukları belirleyen iyi bir tasarım planı oluşturun. Bu, bileşenlerin ve etkileşimlerinin daha iyi bir şekilde yönetilmesine yardımcı olur.
2. **Modüler ve Bağımsız Bileşenler:** Bileşenleri mümkün olduğunca bağımsız ve modüler hale getirin. Her bir bileşenin tek bir görevi veya sorumluluğu olsun. Bu şekilde, her bileşeni ayrı ayrı yönetmek ve daha az bağımlılıkla çalışmak daha kolay olur.
3. **İyi Bir Dosya ve Klasör Yapısı:** Proje dosya ve klasör yapısını düzenli ve iyi organize edin. Bileşenlerin ve ilgili dosyaların yerlerini ve ilişkilerini net bir şekilde belirleyin. Bu, projenin büyümesi ve yönetilmesi daha kolay hale getirecektir.
4. **İyi Dokümantasyon:** Bileşenler arasındaki ilişkileri, sorumlulukları ve iletişimlerini açıkça belgeleyin. Bu, projeyi daha iyi anlamak ve sorunları daha hızlı çözmek için önemlidir. İyi bir dokümantasyon, karmaşıklığı azaltır ve ekip üyeleri arasındaki işbirliğini kolaylaştırır.
5. **İyi İletişim ve Ekip Çalışması:** Ekip üyeleri arasında iyi iletişim ve işbirliği sağlayın. Bileşenler arasındaki etkileşimleri sürekli olarak takip edin ve gerektiğinde yeniden değerlendirin. Projenin büyümesi ve karmaşıklığıyla başa çıkmak için düzenli olarak toplantılar yapın ve projeyi gözden geçirin.
6. **Otomatik Testler:** Bileşenlerin doğru çalıştığından ve birbiriyle uyumlu olduğundan emin olmak için otomatik testler kullanın. Bu, hataları erken tespit etmenize ve bileşenlerin uyumlu bir şekilde çalışmasını sağlamanıza yardımcı olur.

KULLANILAN TEKNOLOJİLER

- **PYTHON**

Bu projenin kodlama aşamasında kullanılacak programlama dili olarak Python seçilmiştir. Bu seçimin yapılmasında aşağıdaki maddeler etkili olmuştur:

1. **Okunabilir ve Kolay Yazılabilir:** Python, basit ve anlaşılır bir sözdizimine sahip olduğu için kodu okumak ve yazmak oldukça kolaydır. Bu, ekip üyeleri arasında işbirliğini kolaylaştırır ve kodun daha hızlı ve hatasız bir şekilde geliştirilmesini sağlar.
2. **Geniş Kütüphane Desteği:** Python, zengin bir standart kütüphane koleksiyonuna sahiptir. Kütüphane yönetim sistemleri için gerekli olan dosya işlemleri, ağ iletişimi, veritabanı erişimi, veri analizi vb. gibi birçok işlevi yerine getirmek için kullanılabilen hazır kütüphanelere sahiptir. Bu, geliştirme sürecini hızlandırır ve işlevselliği genişletir.
3. **Çapraz Platform Desteği:** Python, birçok işletim sistemi ve platformda çalışabilme yeteneğine sahiptir. Bu, kütüphane yönetim sisteminin farklı işletim sistemlerinde sorunsuz bir şekilde çalışabilmesini sağlar. Ayrıca, Python'un web tabanlı çözümler için popüler çerçeveleri de vardır, bu da uygulamanın tarayıcı üzerinden erişilebilir olmasını sağlar.
4. **Büyük Bir Topluluk ve Kaynak:** Python, geniş bir geliştirici topluluğuna sahiptir ve bu topluluk, dokümantasyon, sorun giderme ve yardımcı kaynaklar konusunda zengin bir kaynak havuzunu paylaşır. Bu da projenizin karşılaşılabileceği sorunlara hızlı ve etkili bir şekilde çözüm bulmanızı sağlar.
5. **Hızlı Prototip Oluşturma:** Python, dinamik ve etkileşimli bir dildir. Bu, hızlı prototipler oluşturmanızı ve fikirleri daha hızlı bir şekilde test etmenizi sağlar. Kütüphane yönetim sistemi gibi karmaşık bir projede, hızlı prototipler oluşturmak ve gereksinimleri iteratif olarak geliştirmek önemlidir.
6. **Genişletilebilirlik:** Python, C ve C++ gibi dillerle de entegrasyon yapabilme yeteneğine sahiptir. Bu, performans gerektiren kritik bölümleri C veya C++ ile yazabilir ve Python ile yüksek seviyeli kodlama yapabilirsiniz. Bu sayede performans ve esneklik arasında denge kurmanızı sağlar.

- **KÜTÜPHANELER**

- **Thinker Kütüphanesi**

Tkinter, grafik kullanıcı arayüzü (GUI) uygulamaları geliştirmek için kullanılan bir kütüphanedir. Tkinter, Tk GUI toolkit'inin Python bağlantılarından oluşur ve Python ile birlikte standart olarak gelir.

Tkinter, kullanıcı arayüzü bileşenlerini oluşturmak ve yönetmek için bir dizi fonksiyon ve sınıf sağlar. Bunlar arasında pencere, etiketler, düğmeler, metin kutuları, listeler, çerçeveler, menüler, iletişim kutuları vb. gibi birçok yaygın kullanılan bileşen bulunur.

Tkinter'in bazı önemli özellikleri ve işlevleri şunlardır:

1. **Basit ve Kolay Kullanım:** Tkinter, başlangıç düzeyindeki geliştiricilerin bile kolaylıkla GUI uygulamaları geliştirmesini sağlar. Kolay ve anlaşılır bir arayüz sunar ve çoğu bileşenin kullanımı oldukça basittir.
2. **Çok Platform Desteği:** Tkinter, çeşitli platformlarda çalışabilme yeteneğine sahiptir. Windows, macOS ve Linux gibi işletim sistemlerinde sorunsuz bir şekilde çalışır.
3. **Hızlı Prototipleme:** Tkinter, hızlı prototipleme için ideal bir araçtır. GUI bileşenlerini hızla oluşturabilir, düzenleyebilir ve test edebilirsiniz.
4. **Özelleştirme Yeteneği:** Tkinter, kullanıcı arayüzünü ihtiyaçlarınıza göre özelleştirmenizi sağlar. Fontlar, renkler, boyutlar ve yerleşim gibi birçok özelliği özelleştirebilirsiniz.
5. **Entegre Edilebilirlik:** Tkinter, Python dilinin tam bir parçasıdır ve diğer Python kütüphaneleriyle kolayca entegre edilebilir. Veritabanlarına erişim, veri analizi, ağ işlemleri gibi farklı modüllerle birlikte kullanılabilir.
6. **Geniş Dokümantasyon ve Topluluk Desteği:** Tkinter, geniş bir kullanıcı topluluğuna sahiptir ve birçok kaynak, dokümantasyon ve örnek projeler bulunmaktadır. Bu da yeni başlayanlar için kaynak bulmayı ve sorunları çözmeyi kolaylaştırır.

○ “os” Modülü

“os” modülü, işletim sistemi işlemleriyle etkileşimde bulunmak için kullanılan bir modüldür. Bu modül, işletim sistemiyle ilgili çeşitli işlevleri sağlar ve Python programının işletim sistemi düzeyinde davranışlarını kontrol etmesine olanak tanır.

os modülü, aşağıdaki gibi çeşitli işletim sistemi işlevlerini içerir:

- Dosya ve izin işlemleri: Dosya ve izin oluşturma, silme, taşıma, yeniden adlandırma gibi işlemleri yapmak için işlevler sağlar.
- Çalışma dizini: Çalışma dizinini değiştirmek veya almak için işlevler sunar.
- Çevre değişkenleri: Sistem çevre değişkenlerine erişmek veya ayarlamak için işlevler sağlar.
- Sistem çağrıları: İşletim sistemi sistem çağrılarına doğrudan erişim sağlamak için işlevler sunar.
- Kullanıcı ve grup işlemleri: Kullanıcı ve grup bilgilerine erişmek veya değiştirmek için işlevler sağlar.
- Zaman ve saat işlemleri: Zaman ve tarih işlemleri için işlevler sunar.
- İşlem yönetimi: İşletim sistemi işlemlerini başlatmak, sonlandırmak veya yönetmek için işlevler sağlar.

Bu projede os modülünün .system işlevi kullanılmıştır. “os.system()” fonksiyonu, işletim sistemi komutlarını çalıştırmak için kullanılan bir işlevdir. Bu fonksiyon, işletim sistemine bir komut dizesi gönderir ve komutun çalıştırılmasını sağlar.

○ “sys” Modülü

sys modülü, Python yürütme zamanı sistemini yönetmek için kullanılan bir modüldür. Bu modül, Python yürütme ortamıyla ilgili bilgileri sağlar ve Python yürütme zamanının davranışını değiştirmek veya kontrol etmek için işlevler ve özellikler sunar.

sys modülü, aşağıdaki gibi çeşitli işlevlere ve özelliklere sahiptir:

- Komut satırı argümanları: **sys.argv** özelliği, Python programının komut satırı argümanlarını içeren bir liste sağlar.
- Standart giriş/çıkış: **sys.stdin**, **sys.stdout** ve **sys.stderr** özellikleri, standart giriş, çıkış ve hata akışını temsil eder.
- Yürütme zamanı bilgileri: **sys.version** özelliği, Python yürütme zamanının sürüm bilgisini sağlar. **sys.platform** özelliği, Python'ın hangi işletim sisteminde çalıştığını belirtir.
- Program sonlandırma: **sys.exit()** işlevi, Python programını sonlandırmak için kullanılır.
- Yerel ve küresel değişkenler: **sys.modules** özelliği, Python yürütme zamanında yüklenmiş modüllerin bir sözlüğünü sağlar.
- Çevre değişkenleri: **sys.environ** özelliği, sistem çevre değişkenlerine erişim sağlar.
- Yolu değiştirmek: **sys.path** özelliği, Python'ın modülleri ararken kullanacağı yol listesini sağlar.

○ **mysql.connector** Kütüphanesi

mysql.connector kütüphanesi, Python programları ile MySQL veritabanı arasında iletişim kurmak için kullanılan bir Python paketidir. Bu kütüphane, MySQL veritabanına bağlanma, sorgu çalıştırma, veri alma ve yazma gibi işlemleri gerçekleştirmek için gerekli işlevleri sağlar.

mysql.connector kütüphanesi, MySQL veritabanı sunucusuna doğrudan erişim sağlamak için kullanılır. Bu kütüphane, Python programının MySQL veritabanına bağlanmasını, veritabanı işlemlerini gerçekleştirmesini ve sonuçları almasını sağlar. Bazı temel işlevleri şunlardır:

- Veritabanına bağlanma: **mysql.connector.connect()** işlevi, MySQL veritabanı sunucusuna bir bağlantı oluşturmak için kullanılır. Bağlantı parametreleri (kullanıcı adı, parola, veritabanı adı, vb.) ile birlikte kullanılır.
- Sorgu çalıştırma: **cursor.execute()** işlevi, SQL sorgularını MySQL veritabanında çalıştırmak için kullanılır. Sorgunun sonuçları, **cursor** nesnesi üzerinden alınabilir.

- Sonuçları alma: **cursor.fetchall()** veya **cursor.fetchone()** işlevleri, sorgunun sonuçlarını almak için kullanılır. Sonuçlar, Python veri yapılarına dönüştürülerek elde edilebilir.
- Veri ekleme ve güncelleme: **cursor.execute()** işlevi, INSERT, UPDATE veya DELETE gibi veritabanı işlemlerini gerçekleştirmek için kullanılabilir.
- Bağlantıyı kapatma: **connection.close()** işlevi, MySQL veritabanı sunucusuyla olan bağlantıyı sonlandırmak için kullanılır.

- **IDE/Editor**

- **Visual Studio Code**

Bu projenin geliştirme aşamasında editör olarak Visual Studio Code kullanılmıştır. Aşağıdaki maddeler bu editörün seçilmesinde etkili olmuştur.

1. VS Code, kullanımı kolay ve hafif bir metin düzenleyici ve entegre geliştirme ortamıdır. Başlangıç düzeyindeki kullanıcılar için idealdir.
2. VS Code, geniş bir programlama dilini destekler ve geliştirme sürecini kolaylaştıran bir dizi özelliğe sahiptir, örneğin kod tamamlama, hata ayıklama ve Git entegrasyonu gibi.
3. VS Code, platformlar arası destek sunar, bu da Windows, macOS ve Linux gibi farklı işletim sistemlerinde sorunsuz bir şekilde kullanabileceğiniz anlamına gelir.
4. VS Code, geniş bir uzantı ekosistemine sahiptir. Bu sayede işlevselliğini kişiselleştirebilir ve projenizin gereksinimlerine göre özelleştirilmiş bir geliştirme ortamı oluşturabilirsiniz.

- **MySQL**

Bu projede veri tabanı yönetim sistemi olarak MySQL tercih edilmiştir. Aşağıdaki maddeler veri tabanı olarak MySQL seçilmesinde etkili olmuştur.

1. MySQL, popüler ve yaygın olarak kullanılan bir ilişkisel veritabanı yönetim sistemidir. Birçok büyük ölçekli uygulama ve web sitesi tarafından tercih edilir.
2. MySQL, açık kaynaklı bir yazılımdır ve ücretsiz olarak kullanılabilir. Bu, maliyetleri düşürür ve bütçesi sınırlı olan projeler için ideal bir seçenek yapar.
3. MySQL, yüksek performanslı ve ölçeklenebilir bir veritabanıdır. Büyük veri hacimlerini etkin bir şekilde işleyebilir ve yoğun trafikli uygulamalarda performansı sağlar.

4. MySQL, geniş bir topluluğa ve kaynaklara sahiptir. Bu, kullanıcıların sorunlarını çözmek, bilgi almak ve gelişmeleri takip etmek için birçok kaynak bulabilecekleri anlamına gelir. Ayrıca, geniş bir destek ve belgelendirme ağı da vardır.

- Projenin GitHub reposu: <https://github.com/emirhan-turkmen/20060760-yaz-lab>
- Proje tanım videosu: <https://youtu.be/LKNFKuiAeP4>

- Şekil 1' de Giriş Ekranı verilmiştir.

The image shows a web browser window with the title "KUTUPHANE YONETIM SISTEMI". The main content area has a dark gray background. At the top, the text "Kutuphane Yonetim Sistemi" is displayed in a bold, black font. Below it, the word "Giris" is centered in a bold, black font. There are two input fields: the first is labeled "Kullanıcı Adı:" and the second is labeled "Şifre:". Below the password field is a button labeled "Giris".

Şekil 1

- Şekil 2'de Ana Sayfa verilmiştir.

KUTUPHANE YONETİM SİSTEMİ

OGRENCİ ARA

KITAP ID

ARA BUL

ÇIKIŞ YAP

Oğrenci Ara Kitap Ara Kitap Al Kitap İade Et

DETAYLAR

ID	Oğrenci	Kitap	Alma Tarihi	İade Tarihi
9	burak olcay	Akdeniz	2023-06-18 16	2023-06-18 16

Şekil 2

- Şekil 3'te Öğrenci Arama Ekranı verilmiştir.

Oğrenci Ara

Aranacak Kriter

Girin

Bul

Oğrenci ID	Oğrenci Adı	Numara	Adres
1	emirhan türkmen	5555555555	Bursa

Şekil 3

- Şekil 4'te Kitap Arama Ekranı verilmiştir.

Kitap Ara

Kutuphanede Ara

Kriter

Girin

Kitap ID	Kitap Adi	Yazar	Durum
8	Otomatik Portakal	Anthony Burgess	YES

Şekil 4

- Şekil 5'te Kitap Alma Ekranı verilmiştir.

Kutuphane Yonetimi

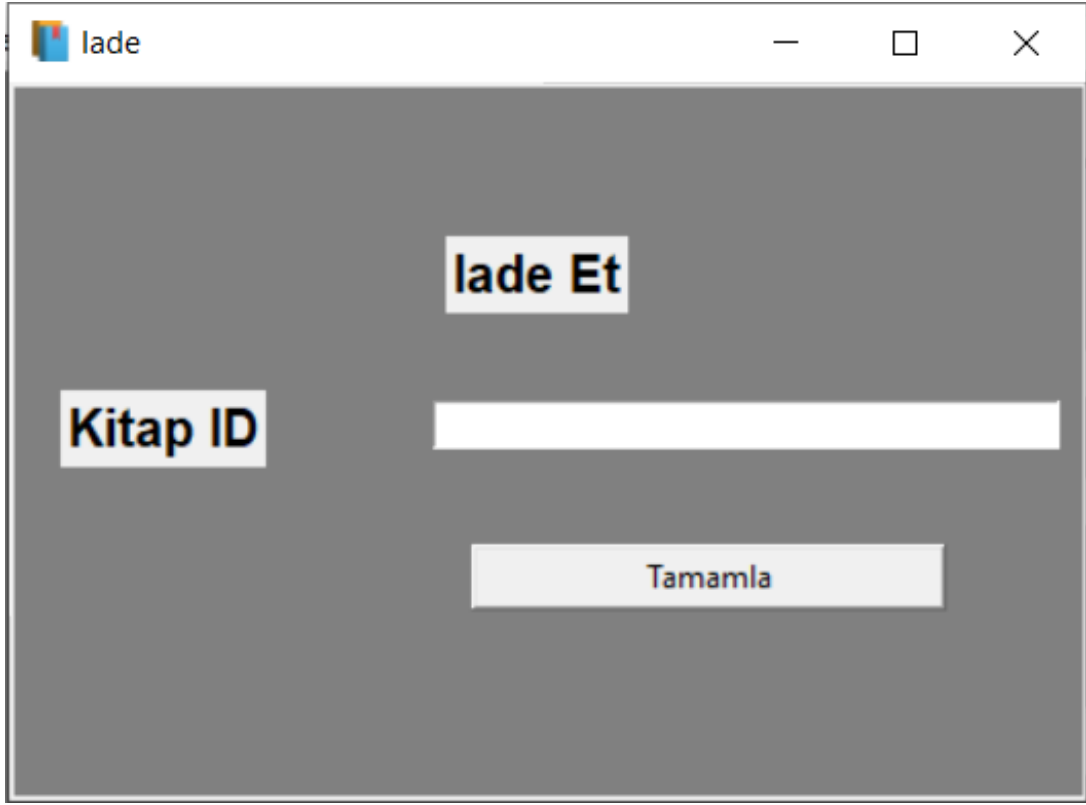
Kitap Alma

Kitap ID:

Ogrenci ID:

Şekil 5

- Şekil 6'da Kitap İade Ekranı verilmiştir.



Şekil 6

- Şekil 7’de Öğrenci Ekleme Ekranı verilmiştir.



Ogrenci Ekle

Ogrenci Bilgileri:

Ad:

Numara:

Adres:

Kaydet

Şekil 7

- Şekil 8’de Kitap Ekleme Ekranı verilmiştir.

KITAP EKLE

Kitap Detaylari:

Kitap Adi:

Yazar:

Kaydet

Şekil 8

- Şekil 9'da Kullanıcı Ekleme Ekranı verilmiştir.

Kul. Ekle

Kullanıcı Detayları:

Kul. Adi:

İsim:

Sifre:

Kaydet

Şekil 9

- Şekil 10’da Kullanıcı Silme Ekranı verilmiştir.

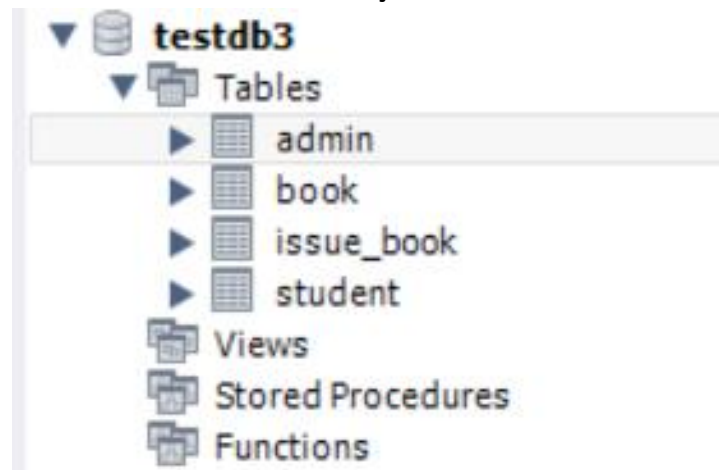
Kullanici Silme

Kullanici ID

Sil

Şekil 10

- Şekil 11’de Veri Tabanı Tabloları verilmiştir.



Şekil 11