

T.C. MARMARA ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ
2023-2024 EĞİTİM ÖĞRETİM YILI GÜZ DÖNEMİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
VERİTABANI YÖNETİM SİSTEMLERİ DERSİ
PROJE DÖKÜMANI

Açıklamalar

- Ödevinizi gönderirken tek bir dosya şeklinde **OğrenciNo_AdSoyad_VTYS_FinalÖdevi.pdf**, olmasına dikkat ediniz.
- Google Classroom platformlarına 27 Aralık 2023 tarihine kadar yükleyiniz.
- Tüm sorular soru metninin altındaki alanlara cevaplandırılacaktır. Bu belgenin formatını bozmayınız, sadece gerekli alanları doldurunuz.

1. Gerçekleştirdiğiniz veri tabanı projesi için grup arkadaşlarınız var ise isimlerini yazınız ve projenize ait veri tabanı/diğer yazılım bileşenleri hakkında bilgi veriniz. (10 p)

Projede geliştirici olarak yer alan üyeler Karun Acar ve Emre Okçelen'dir. Bu projeyi, çeşitli teknolojilerin entegrasyonu ile oluşturduk. Frontend tarafında HTML, CSS ve JavaScript kullanarak kullanıcı arayüzü tasarladık. HTML ile sayfaların yapısını organize ettik, CSS ile görünümü düzenleyerek kullanıcı arayüzüne estetik bir görünüm kazandırdık. JavaScript'i ise kullanıcı etkileşimlerini ve dinamik içerikleri sağlamak için kullandık. Backend tarafında ise PHP teknolojisini kullanarak, kullanıcı hesap yönetimi, sepet işlemleri vb. diğer işlemleri yönettik. Veritabanı işlemleri için ise MySQL kullanarak ürün bilgilerini, kullanıcı hesaplarını ve sipariş verilerini güvenli ve düzenli bir şekilde depoladık. Geliştirme sürecinde XAMPP, Apache sunucusu, MySQL veritabanı ve PHP ile birlikte çalışarak projenin yerel bir ortamda test edilmesine olanak sağladık. Bu teknolojilerin bir araya gelmesiyle, güvenli, hızlı ve kullanıcı dostu bir e-ticaret platformu oluşturduk.

2. Gerçekleştirdiğiniz projenin amacını detaylı bir şekilde açıklayınız. (Proje konularınızın yer aldığı classroom excel dosyası içerisindeki yazdığınız açıklama kabul edilmeyecektir.) (10 p)

Vêtement Violet; HTML, CSS, PHP, JavaScript ve MySQL teknolojilerini kullanan e-ticaret kapsamında değerlendirebilecek bir çevrimiçi alışveriş platformudur. HTML ve CSS ile estetik bir kullanıcı arayüzü oluşturuldu, PHP ve JavaScript ile ise bu arayüzü dinamik hale getirildi. Aynı zamanda MySQL veritabanı kullanarak ürünleri depolayıp yönetebilmeyi hedefledik. Bu platform belirtilen teknolojiler doğrultusunda kullanıcıların; ürün kategorileri arasında kolayca gezinebilmesini, ürünleri görüntüleyebilmesini, beğendiği ürünleri kaydedebilmesini ve sepete ürün ekleyebilmesini sağlar. Bunlara ek olarak kullanıcı dostu arayüzüyle son moda trendleri takip edenler için geniş bir ürün yelpazesi ve moda tutkunlarına trendleri takip etme ve stil sahibi olma fırsatı sunar.

Bu proje, bize farklı dilleri ve teknolojileri bir araya getirerek gerçek bir uygulama üzerinde çalışma fırsatı verdi ve bu sayede e-ticaretin arkasındaki temel yapıları anlamamıza yardımcı oldu.

3. Tasarladığınız veri tabanı mimarisinde hangi tablo ve ilişkileri kullandığınızı açıklayınız. (10 p)

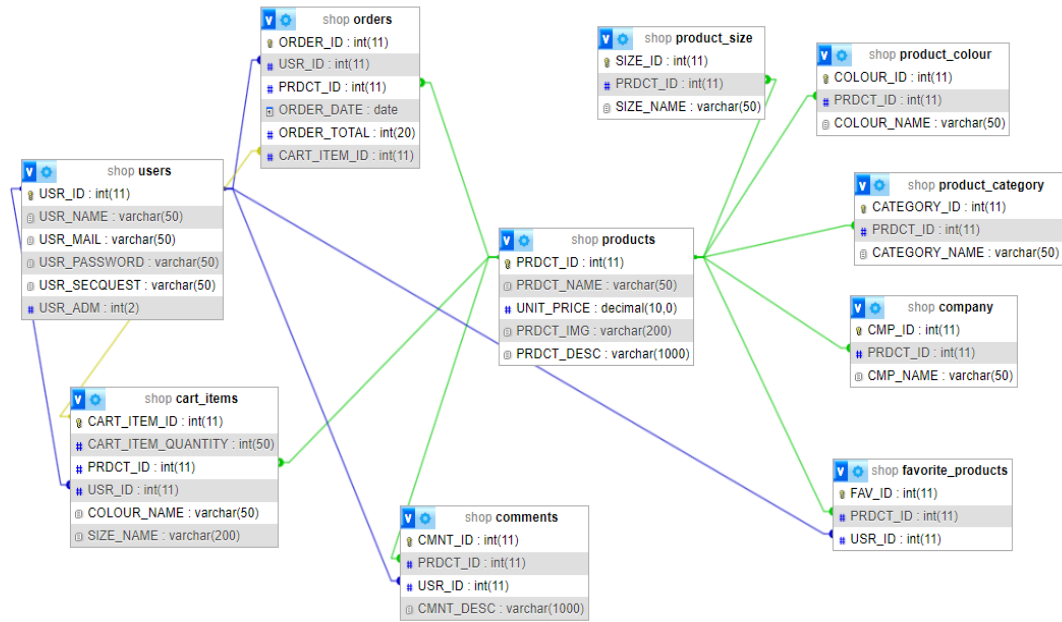
Bu projenin veri tabanı mimarisinde 'Users', 'Products', 'Orders', 'Product_Category', 'Product_Colour', 'Product_Size', 'Company', 'Favorite_Products', 'Comments', 'Cart_Items' olmak üzere 10 tane tablo kullanılmıştır. 'Users' tablosu kullanıcının adını, e-posta adresini, hesap şifresini ve güvenlik kodu bilgilerini içermektedir. Birincil anahtar olarak 'USR_ID' değişkenini kullanırken herhangi bir yabancı anahtar bulundurmamaktadır. 'Products' tablosu ürünün adını, fiyatını, resmini ve açıklama bilgilerini içermektedir. Birincil anahtar olarak 'PRDCT_ID' değişkenini kullanırken herhangi bir yabancı anahtar bulundurmamaktadır. 'Orders' tablosu satın alınmaya hazır ürünlerin satın alınma tarihini ve seçilen ürünlerden kaç adet alındığı bilgisini içermektedir. Birincil anahtar olarak 'ORDER_ID' değişkeni kullanılmıştır. 'PRDCT_ID' ve 'USR_ID' yabancı anahtarları ile 'Users' ve 'Products' tablolarına erişim sağlanılmıştır. 'Product_Category' tablosu ürünün kategori bilgilerini içermektedir. Birincil anahtar olarak 'CATEGORY_ID' değişkenini kullanırken, yabancı anahtar olarak 'PRDCT_ID' anahtarını kullanır. 'Product_Colour' tablosu ürünün renk bilgilerini içermektedir. Birincil anahtar olarak 'COLOUR_ID' değişkenini kullanırken, yabancı anahtar olarak 'PRDCT_ID' anahtarını kullanır.

'Product_Size' tablosu ürünün boyut(beden) bilgilerini içermektedir. Birincil anahtar olarak 'SIZE_ID' değişkenini kullanırken, yabancı anahtar olarak 'PRDCT_ID' anahtarını kullanır. 'Company' tablosu ürünün marka bilgilerini içermektedir. Birincil anahtar olarak 'CMP_ID' değişkenini kullanırken, yabancı anahtar olarak 'PRDCT_ID' anahtarını kullanır. 'Favorite_Products' tablosu kullanıcının beğenip kaydettiği ürünlerin bilgisini içermektedir. Birincil anahtar olarak 'FAV_ID' değişkenini kullanırken, yabancı anahtar olarak 'USR_ID' ve 'PRDCT_ID' anahtarlarını kullanır. 'Comments' tablosu ürünler hakkında kullanıcıların yaptıkları yorum bilgilerini içermektedir. Birincil anahtar olarak 'CMNT_ID' değişkenini kullanırken, yabancı anahtar olarak 'USR_ID' ve 'PRDCT_ID' anahtarlarını kullanır. 'Cart_Items' tablosu sepette bulunan ürünlerin bilgisini içermektedir. Birincil anahtar olarak 'CART_ITEM_ID' değişkeni kullanılmıştır. 'PRDCT_ID' ve 'USR_ID' yabancı anahtarları ile 'Users' ve 'Products' tablolarına erişim sağlanılmıştır.

'Users' tablosu ile 'Orders' tablosu arasında bir kullanıcı birden fazla sipariş verebileceği için bire-çok ilişki bulunmaktadır. 'Users' tablosu ile 'Comments' tablosu arasında bir kullanıcı birden fazla yorum yazabileceği için bire-çok ilişki bulunmaktadır. 'Users' tablosu ile 'Favorite_Products' tablosu arasında bir kullanıcı birden fazla ürünü ve bir ürünü birden fazla kullanıcı kaydedebileceği için çoka-çok ilişki bulunmaktadır. 'Products' tablosu ile 'Product_Category' tablosu arasında bir kategoride birden fazla ürün olabileceği için bire-çok ilişki bulunmaktadır. 'Products' tablosu ile 'Product_Colour' tablosu arasında bir ürünün birden fazla renk seçeneği bulunabileceğinden ötürü bire-çok ilişki bulunmaktadır. 'Products' tablosu ve 'Product_Size' tablosu arasında bir ürünün birden fazla beden seçeneği olacağından dolayı bire-çok ilişki bulunmaktadır. 'Products' tablosu ile 'Company' tablosu arasında bir şirkete ait birden bulunabileceğinden dolayı bire-çok ilişki barındırmaktadır. 'Cart_Items' tablosu ile 'Users' tablosu arasında bir kullanıcının bir sepette birden çok

ürün bulundurabilmesinden dolayı bire-çok ilişki barındırmaktadır. 'Cart_Items' tablosu ile 'Products' tablosu arasında ise bir ürünün birden fazla sepette bulunabileceğinden bire-çok ilişki vardır.

4. Veri tabanı ER (Entity Relationship) diagramının bilgisayar ortamında çizilmiş halini paylaşınız. (Ara raporda eksik kısımlar bu raporda giderilmelidir ve ER çizme programlarından faydalanılabilir. Elle çizim, çizip fotoğrafını çekme vb. kabul edilmeyecektir.) (10 p)



5. Herhangi iki tablonuz için DDL (create) kodları yazılmalıdır. (10 p)

```
CREATE TABLE shop.product_colour (COLOUR_ID INT(11) NOT NULL  
AUTO_INCREMENT , PRDCT_ID INT(11) NOT NULL , COLOUR_NAME  
VARCHAR(50) NOT NULL , PRIMARY KEY (COLOUR_ID)) ENGINE = InnoDB;  
ALTER TABLE product_colour ADD CONSTRAINT PRODUCT_COLOUR_FK1  
FOREIGN KEY (PRDCT_ID) REFERENCES products(PRDCT_ID) ON DELETE  
RESTRICT ON UPDATE RESTRICT;
```

```
CREATE TABLE shop.comments (CMNT_ID INT(11) NOT NULL  
AUTO_INCREMENT , PRDCT_ID INT(11) NOT NULL , USR_ID INT(11) NOT  
NULL , CMNT_DESC VARCHAR(1000) NOT NULL , PRIMARY KEY  
(CMNT_ID)) ENGINE = InnoDB;  
ALTER TABLE comments ADD CONSTRAINT COMMENTS_FK1 FOREIGN KEY  
(PRDCT_ID) REFERENCES products(PRDCT_ID) ON DELETE RESTRICT ON  
UPDATE RESTRICT; ALTER TABLE comments ADD CONSTRAINT
```

COMMENTS_FK2 FOREIGN KEY (USR_ID) REFERENCES users(USR_ID) ON DELETE RESTRICT ON UPDATE RESTRICT;

6. 5 adet DML (update, insert, delete) içeren kodları yazılmalıdır. (10 p)

- 1) \$insertQuery = "INSERT INTO favorite_products (PRDCT_ID, USR_ID) VALUES ('\$productID', '\$userID')";
- 2) "DELETE FROM product_category WHERE PRDCT_ID IN (SELECT PRDCT_ID FROM products WHERE PRDCT_NAME = ? AND UNIT_PRICE = ?)
- 3) "UPDATE products SET PRDCT_NAME=?, UNIT_PRICE=?, PRDCT_IMG=?, PRDCT_DESC=? WHERE PRDCT_NAME=?"
- 4) DELETE FROM company WHERE PRDCT_ID IN (SELECT PRDCT_ID FROM products WHERE PRDCT_NAME = ? AND UNIT_PRICE = ?)
- 5) INSERT INTO product_colour (PRDCT_ID, COLOUR_NAME) VALUES (?, ?)

7. Projenize ait kendi belirlediğiniz 10 adet SQL sorgusu yazınız, sorguların amacını ve sonuç çıktısını da lütfen ekleyiniz. (Açıklama: Sorgular Select deyimleri ve gruplama fonksiyonlarını HAVING deyimini (min, max, avg, count gibi) ve join deyimlerini (en az iki tablo ile birleştirme sorgusu) içerecek şekilde basitten karmaşığa doğru gitmelidir. Proje sunum anında veri tabanınıza ait sorular SQL ortamında gösterilecek ve açıklanacaktır. Raporunuzda ise sorgular, sorguların cevap ve sonuçlarının ekran görüntüsü olarak paylaşılması beklenmektedir. (20 p)

Örnek:

Öğrenci adı, yaşı ve aldığı dersleri döndüren SQL sorgusu aşağıdaki gibidir:

```
SELECT
    Ogrenciler.ogrenci_ad AS OgrenciAdi,
    Ogrenciler.ogrenci_yas AS OgrenciYasi,
    Dersler.ders_ad AS DersAdi
FROM
    Ogrenciler, Kayitlar, Dersler
WHERE
    Ogrenciler.ogrenci_id = Kayitlar.ogrenci_id
    AND Kayitlar.ders_id = Dersler.ders_id;
```

OgrenciAdi	OgrenciYasi	DersAdi
Ahmet	20	Matematik
Ahmet	20	Fizik
Ayşe	22	Matematik

// Bu SQL sorgusu, belirli bir kullanıcıya ait alışveriş sepetinde bulunan ürün sayısını productCount adı altında döndürmek için kullanılıyor. Sepetteki ürünleri cart_items ve products tablolarını birleştirerek ve belirli bir kullanıcı kimliğine göre filtreleyerek sayıyor.

\$sqlCount = "SELECT COUNT(*) AS productCount FROM cart_items ci JOIN products p ON ci.PRDUCT_ID = p.PRDUCT_ID WHERE ci.USR_ID = \$USR_ID";

SQL sorgunuz başarılı olarak çalıştırıldı.

```
SELECT COUNT(*) AS productCount FROM cart_items ci JOIN products p ON ci.PRDUCT_ID = p.PRDUCT_ID WHERE ci.USR_ID = 2;
```

☐ Profil çıkart [Satır içi düzenle] [Düzenle] [SQL'i açıkla] [PHP kodu oluştur] [Yenile]

Fazladan seçenekler

productCount

3

//Bu SQL sorgusu, belirli bir kullanıcıya ait alışveriş sepetindeki ürünlerin ID'si, resmi, adı, birim fiyatı ve miktar bilgilerini cart_items ve products tablolarını birleştirerek ve kullanıcı kimliğine göre filtreleyerek getiriyor.

"SELECT p.PRDUCT_ID, p.PRDUCT_IMG, p.PRDUCT_NAME, p.UNIT_PRICE, ci.CART_ITEM_QUANTITY FROM cart_items ci JOIN products p ON ci.PRDUCT_ID = p.PRDUCT_ID WHERE ci.USR_ID = \$USR_ID";

✓ Gösterilen satır 0 - 2 (toplam 3, Sorgu 0,0005 saniye sürdü.)

```
SELECT p.PRDUCT_ID, p.PRDUCT_IMG, p.PRDUCT_NAME, p.UNIT_PRICE, ci.CART_ITEM_QUANTITY FROM cart_items ci JOIN products p ON ci.PRDUCT_ID = p.PRDUCT_ID WHERE ci.USR_ID = 2;
```

☐ Profil çıkart [Satır içi düzenle] [Düzenle] [SQL'i açıkla] [PHP kodu oluştur] [Yenile]

☐ Tümünü göster | Satır sayısı: 25 | Satırları süz: Bu tabloda ara | Anahtara göre sırala: Yok

Fazladan seçenekler

PRDUCT_ID	PRDUCT_IMG	PRDUCT_NAME	UNIT_PRICE	CART_ITEM_QUANTITY
49	product-08.jpg	Cactus T-sirht	95	2
48	product-14.jpg	Black Crop-top	950	4
48	product-14.jpg	Black Crop-top	950	10

// Bu SQL sorgusu, favorite_products tablosuna belirli bir ürün ve kullanıcı ID'si eklemek için kullanılıyor. PRDUCT_ID ve USR_ID sütunlarına, belirtilen \$productID ve \$userID değerleri ekleniyor.

\$insertQuery = "INSERT INTO favorite_products (PRDUCT_ID, USR_ID) VALUES ('\$productID', '\$userID')";

✓ 1 satır eklendi.

Eklenen satır id: 8 (Sorgu 0,0006 saniye sürdü.)

```
INSERT INTO favorite_products (PRDUCT_ID, USR_ID) VALUES ('48', '2');
```

Satır içi düzenle [Düzenle] [PHP kodu oluştur]

// Bu SQL sorgusu, belirli bir kullanıcının favori ürünlerinin tamamını favorite_products tablosundan çekmek için kullanılıyor. USR_ID ile belirtilen kullanıcıya ait olan tüm favori ürün kayıtlarını getirmeyi amaçlıyor.

\$sql = "SELECT * FROM favorite_products WHERE USR_ID = '\$USR_ID'";
















✓ Gösterilen satır 0 - 4 (toplam 5, Sorgu 0,0003 saniye sürdü.)

SELECT * FROM favorite_products WHERE USR_ID = '2';

☐ Profil çıkart [Satır içi düzenle] [Düzenle] [SQL'i açıkla] [PHP kodu oluştur] [Yenile]

☐ Tümünü göster | Satır sayısı: 25 | Satırları süz: Bu tabloda ara | Anahtara göre sırala:

Fazladan seçenekler

	FAV_ID	PRDCT_ID	USR_ID
<input type="checkbox"/>   	2	44	2
<input type="checkbox"/>   	3	48	2
<input type="checkbox"/>   	4	50	2
<input type="checkbox"/>   	5	52	2
<input type="checkbox"/>   	8	48	2

//Bu SQL sorgusu, products tablosunda belirli bir ürünün bilgilerini güncellemek için kullanılıyor. PRDCT_NAME, UNIT_PRICE, PRDCT_IMG, ve PRDCT_DESC alanlarındaki verileri belirli bir ürün ismine göre güncellemek üzere kullanılıyor.

"UPDATE products SET PRDCT_NAME=?, UNIT_PRICE=?, PRDCT_IMG=?, PRDCT_DESC=? WHERE PRDCT_NAME=?"

✓ 1 satır etkilendi. (Sorgu 0,0003 saniye sürdü.)

UPDATE products SET PRDCT_NAME = 'A', UNIT_PRICE = 24, PRDCT_IMG = 'product-02.jpg', PRDCT_DESC = 'GUZEL' WHERE PRDCT_NAME = 'White Shirt';

Satır içi düzenle [Düzenle] [PHP kodu oluştur]

//Bu SQL sorgusu, product_category tablosundan, products tablosundaki belirli bir ürünün PRDCT_ID'si ve UNIT_PRICE'ı belirli bir değere eşit olan kayıtları silmek için kullanılıyor. İç içe geçmiş bir sorgu yapısıyla, products tablosundan belirli koşullara uyan ürünün ID'sini alarak, bu ID'lere ait olan kategori ilişkilerini product_category tablosundan siliyor.

"DELETE FROM product_category WHERE PRDCT_ID IN (SELECT PRDCT_ID FROM products WHERE PRDCT_NAME = ? AND UNIT_PRICE = ?)

✓ 1 satır etkilendi. (Sorgu 0,0003 saniye sürdü.)

UPDATE products SET PRDCT_NAME = 'A', UNIT_PRICE = 24, PRDCT_IMG = 'product-02.jpg', PRDCT_DESC = 'GUZEL' WHERE PRDCT_NAME = 'White Shirt';

Satır içi düzenle [Düzenle] [PHP kodu oluştur]

// Bu SQL sorgusu, products tablosundan ürün adlarına göre gruplayarak, her bir ürünün en yüksek birim fiyatını buluyor. Daha sonra bu fiyatların 200 birimden büyük olanları filtrelemek için HAVING kullanıyor.

```
SELECT PRDCT_NAME, MAX(UNIT_PRICE) FROM products GROUP BY PRDCT_NAME HAVING MAX(UNIT_PRICE) > 200;
```

✓ Gösterilen satır 0 - 4 (toplam 5, Sorgu 0,0004 saniye sürdü.)

```
SELECT PRDCT_NAME, MAX(UNIT_PRICE) FROM products GROUP BY PRDCT_NAME HAVING MAX(UNIT_PRICE) > 200;
```

☐ Profil çıkart [Satır içi düzenle] [Düzenle] [SQL'i açıkla] [PHP kodu oluştur] [Yenile]

☐ Tümünü göster | Satır sayısı: 25 | Satırları süz: Bu tabloda ara

Fazladan seçenekler

	PRDCT_NAME	MAX(UNIT_PRICE)
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	Black Crop-top	950
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	Black Jean Skirt	380
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	Blue Polo Shirt	450
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	Brown Coat	1000
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	Grey Blazer Jacket	850

// Bu SQL sorgusu, comments tablosundan kullanıcı ID'si 2 olan tüm yorumları getiriyor.

```
SELECT * FROM comments WHERE USR_ID = 2;
```

✓ Gösterilen satır 0 - 2 (toplam 3, Sorgu 0,0003 saniye sürdü.)

```
SELECT * FROM comments WHERE USR_ID = 2;
```

☐ Profil çıkart [Satır içi düzenle] [Düzenle] [SQL'i açıkla] [PHP kodu oluştur] [Yenile]

☐ Tümünü göster | Satır sayısı: 25 | Satırları süz: Bu tabloda ara | Anahtara göre sırala: Yok

Fazladan seçenekler

	CMNT_ID	PRDCT_ID	USR_ID	CMNT_DESC
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	20	49	2	o kadar da iyi değil
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	21	48	2	umarum yakın zamanda indirme girer!!!
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	22	48	2	çok güzel

// Bu SQL sorgusu, 'users' tablosundan, ismi 'A' harfiyle başlayan kullanıcıların tüm bilgilerini getiriyor. Yani, 'USR_NAME' sütunu 'A' harfiyle başlayan kullanıcıların kayıtlarını çekmeyi amaçlıyor.

```
SELECT * FROM users WHERE USR_NAME LIKE 'A%';
```

✓ Gösterilen satır 0 - 1 (toplam 2, Sorgu 0,0003 saniye sürdü.)

```
SELECT * FROM users WHERE USR_NAME LIKE 'A%';
```

☐ Profil çıkart [Satır içi düzenle] [Düzenle] [SQL'i açıkla] [PHP kodu oluştur] [Yenile]

☐ Tümünü göster | Satır sayısı: 25 | Satırları süz: Bu tabloda ara | Anahtara göre sırala: Yok

Fazladan seçenekler

	USR_ID	USR_NAME	USR_MAIL	USR_PASSWORD	USR_SECQUEST	USR_ADM
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	4	Atiye Bengü	bengu@hotmail.com	bengü123	sarısaç	0
<input type="checkbox"/> Düzenle <input type="checkbox"/> Kopyala <input type="checkbox"/> Sil	8	Aisan	Aisan@gmail.com	aisan123	123456	0

// Bu SQL sorgusu, 'comments' ve 'products' tablolarını birleştirerek 'comments' tablosundaki yorumların ve 'products' tablosundaki ürünlerin belirli alanlarını bir araya getiriyor. 'comments' tablosundan yorumların ID'si, açıklaması, kullanıcı ve ürün ID'leri, 'products' tablosundan ise ürün adları bu sorgu ile birleştirilerek çekiliyor. Bu sayede her yorumun ilgili ürün adıyla ilişkilendirilmesi sağlanıyor.

```
SELECT comments.CMNT_ID, comments.CMNT_DESC, comments.USR_ID, comments.PRDUCT_ID, products.PRDUCT_NAME FROM comments JOIN products ON comments.PRDUCT_ID = products.PRDUCT_ID;
```

✓ Gösterilen satır 0 - 6 (toplam 7, Sorgu 0,0004 saniye sürdü.)

`SELECT comments.CMNT_ID, comments.CMNT_DESC, comments.USR_ID, comments.PRDUCT_ID, products.PRDUCT_NAME FROM comments JOIN products ON comments.PRDUCT_ID = products.PRDUCT_ID;`

☐ Profil çıkart ☐ Satır içi düzenle ☐ Düzenle ☐ SQL'i açıkla ☐ PHP kodu oluştur ☐ Yenile

☐ Tümünü göster | Satır sayısı: 25 | Satırları süz: Bu tabloda ara | Anahtara göre sırala: Yok

Fazladan seçenekler

CMNT_ID	CMNT_DESC	USR_ID	PRDUCT_ID	PRDUCT_NAME
17	çok beğendim çok gzl	4	49	Cactus T-sirht
18	iade edicem	4	49	Cactus T-sirht
19	ollley	4	49	Cactus T-sirht
20	o kadar da iyi değil	2	49	Cactus T-sirht
21	umarum yakın zamanda indirimde girer!!!	2	48	Black Crop-top
22	çok güzel	2	48	Black Crop-top
23	çok beğendim çok gzl	4	54	Grey Blazer Jacket

8. Eğer gerçekleştirmiş iseniz, veri tabanı bağlama ve uygulama geliştirme aşamalarınızı kısaca açıklayarak, kullanıcı ara yüz ekranından bir örnek veriniz. Ve geliştirdiğiniz ara yüzü anlatınız. (10 p)

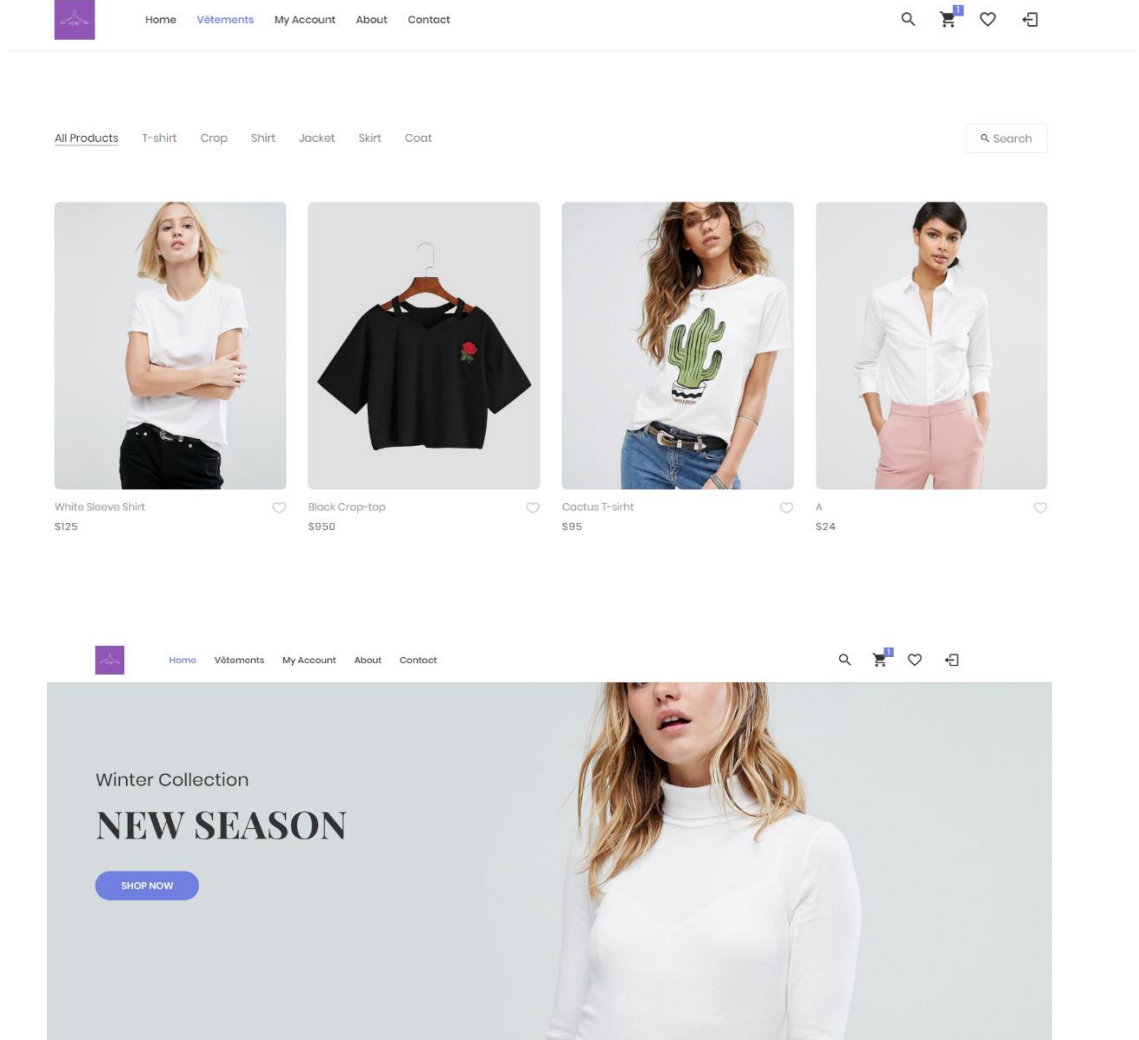
Analiz ve Planlama: İlk adım olarak, projenin gereksinimlerini belirledik ve kullanıcı ihtiyaçları doğrultusunda bir veritabanı modeli oluşturduk. Hangi tabloların olacağını ve bu tablolar arasındaki ilişkileri planladık. Bu aşama, kullanıcıların beklentilerini ve projenin genel yapısını anlamamızı sağladı.

Frontend Tasarımı (HTML, CSS, JavaScript): Kullanıcı arayüzünü oluşturmak için HTML, CSS ve JavaScript kullanarak bir frontend geliştirdik. Kullanıcı dostu ve etkileşimli bir alışveriş sitesi arayüzü tasarladık. Sayfalar arası gezinimi kolaylaştırmak ve kullanıcı deneyimini artırmak için JavaScript'in dinamik özelliklerini kullandık.

Backend Tasarımı (PHP): Veritabanı ile iletişim kurmak, kullanıcının girdiği bilgileri işlemek ve güvenliği sağlamak için PHP kullanarak backend geliştirdik. MySQL veritabanı ile bağlantı kurduk ve gerekli sorguları oluşturarak veritabanındaki verilere erişim sağladık. Kullanıcı kayıt, giriş ve alışveriş sepeti işlemlerini yönetmek için PHP skriptleri yazdık.

Veritabanı Oluşturma (MySQL): XAMPP yapısını kullanarak MySQL veritabanı oluşturduk. Analiz aşamasında belirlediğimiz tabloları ve ilişkileri MySQL komutları ile oluşturduk. Gerekli veri türleri, indeksler ve yabancı anahtarlar kullanılarak veritabanını optimize ettik.

Test ve Debug: Uygulamayı test etmek ve olası hataları gidermek için Apache üzerinde çalışan web sunucusunu kullanarak test ettik. Kullanıcı kayıt, giriş, ürün ekleme ve sepet işlemlerini kapsayan bir dizi test senaryosu oluşturduk



Bu alışveriş sayfası tasarımı, minimalist ve işlevsel bir yaklaşıma odaklanarak, kullanıcı deneyimini ön planda tutmaktadır. Temel hedef, kullanıcıların kolayca gezinmelerini sağlamak ve istedikleri ürünlere hızlı bir şekilde erişmelerini mümkün kılmaktadır. Sayfa, gereksiz detaylardan kaçınarak temiz bir görünüm sunar. Menüler ve navigasyon, kullanıcıların sayfa içinde rahatlıkla gezinmelerini sağlar. Ürün kategorileri, net bir şekilde belirtilir ve kullanıcılar istedikleri kategoriye anında erişebilir. Her ürün sayfası, temel bilgileri içeren sade bir düzenleme sahiptir. Sepet sayfası, kullanıcıların ekledikleri ürünleri görsel olarak sunar. Ürünlerin yanında miktar bilgisi ve birim fiyatı tutulmaktadır.

9. Eğer veri tabanı bağlama işlemini gerçekleştirmemiş iseniz VTYS sistemlerinde Transaction nedir açıklayınız ve çalışmanızdan bir Transaction örneği veriniz. (10 p)

Veri tabanı bağlantısı oluşturuldu.

10. View nedir açıklayınız ve bir adet view, bir adet saklı yordam (Stored Procedure) ifadesine ait SQL deyimlerinin sorgusunu ve cevabını yazınız. (10 p)

SQL VIEW, veritabanındaki birkaç tablonun birleştirilmesiyle oluşturulan bir tablonun bir görüntüsüdür. Bu, kullanıcıların veritabanındaki verileri daha kolay ve anlaşılır bir şekilde görüntülemesine yardımcı olur. SQL VIEW'ler, sorgular tarafından kullanılabilir ve güncellenebilir, ancak direkt olarak veritabanındaki verilere erişemezler. Bu nedenle, SQL VIEW'leri veritabanındaki verileri korumak için kullanışlıdır.

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0.0004 saniye sürdü.)

```
CREATE VIEW CommentProductView AS SELECT comments.CMNT_ID, comments.CMNT_DESC, comments.USR_ID, comments.PRDC_T_ID, products.PRDC_T_NAME FROM comments JOIN products ON comments.PRDC_T_ID = products.PRDC_T_ID;
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]






















✓ Gösterilen satır 0 - 6 (toplam 7, Sorgu 0.0005 saniye sürdü.)





```
SELECT * FROM `commentproductview`
```

☐ Profil çıkart [Satır içi düzenle] [Düzenle] [SQL'i açıkla] [PHP kodu oluştur] [Yenile]

☐ Tümünü göster | Satır sayısı: 25 | Satırları süz: Bu tabloda ara

Fazladan seçenekler

				CMNT_ID	CMNT_DESC	USR_ID	PRDC_T_ID	PRDC_T_NAME
<input type="checkbox"/>				17	çok beğendim çok gzl	4	49	Cactus T-sirht
<input type="checkbox"/>				18	iade edicem	4	49	Cactus T-sirht
<input type="checkbox"/>				19	ollley	4	49	Cactus T-sirht
<input type="checkbox"/>				20	o kadar da iyi değil	2	49	Cactus T-sirht
<input type="checkbox"/>				21	umarum yakın zamanda indirim girer!!!	2	48	Black Crop-top
<input type="checkbox"/>				22	çok güzel	2	48	Black Crop-top
<input type="checkbox"/>				23	çok beğendim çok gzl	4	54	Grey Blazer Jacket

↑ ☐ Tümünü işaretle Seçilileri:    

```
CREATE PROCEDURE UpdateProduct( IN newProductName VARCHAR(255), IN newUnitPrice DECIMAL(10, 2), IN newProductImage VARCHAR(255), IN newProductDescription TEXT, IN oldProductName VARCHAR(255) ) BEGIN UPDATE products SET PRDC_T_NAME = newProductName, UNIT_PRICE = newUnitPrice, PRDC_T_IMG = newProductImage, PRDC_T_DESC = newProductDescription WHERE PRDC_T_NAME = oldProductName; END;
```

✓ MySQL boş bir sonuç kümesi döndürdü (yani sıfır satır). (Sorgu 0.0024 saniye sürdü.)

```
CALL UpdateProduct('Kazak', 33.0, 'product-02.jpg', 'yünlü kaza', 'A');
```

[Satır içi düzenle] [Düzenle] [PHP kodu oluştur]