Öğrenci Adı Soyadı : Ahmet Erdem Yabacı

Öğrenci Numarası: 170421023

T.C. MARMARA ÜNİVERSİTESİ TEKNOLOJİ FAKÜLTESİ

2024-2025 EĞİTİM ÖĞRETİM YILI

GÜZ DÖNEMİ BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

VERİTABANI YÖNETİM SİSTEMLERİ DERSİ

PROJE-FİNAL DÖKÜMANI

|  |
| --- |
| Tüm sorular soru metnini altındaki alanlara cevaplandırılacaktır. Bu belgenin formatını bozmayınız, sadece gerekli alanları doldurunuz. |

1. Gerçekleştirdiğiniz veri tabanı projesi için grup arkadaşlarınızın isimlerini yazınız ve projenize ait veri tabanı/diğer yazılım bileşenleri hakkında bilgi veriniz. (5 p)

Ahmet Erdem Yabacı

Mehmet Ali Onur Yavuz

Berkay Güner

Yusuf Ünlü

Proje Bileşenleri:

- Veri Tabanı: Microsoft SQL Server kullanılmıştır. Tablolar ve ilişkiler "Yemek Tarifi Uygulaması" ihtiyaçlarına uygun olarak tasarlanmıştır.

- Yazılım Araçları: JavaFX kullanılarak arayüz geliştirilmiştir.

- Diğer Teknolojiler: ER diyagramı için MSSQL’ten otomatik hazırlanmıştır.

1. Gerçekleştirdiğiniz veri tabanı projesi için proje dokümanınızı ve dosyalarınızı içeren herkese açık github bağlantılarınızı paylaşınız. (5 p)

[**https://github.com/ErdemYabaci/vtys\_proje**](https://github.com/ErdemYabaci/vtys_proje)

1. Gerçekleştirdiğiniz projenin amacını detaylı bir şekilde açıklayınız. (10 p)

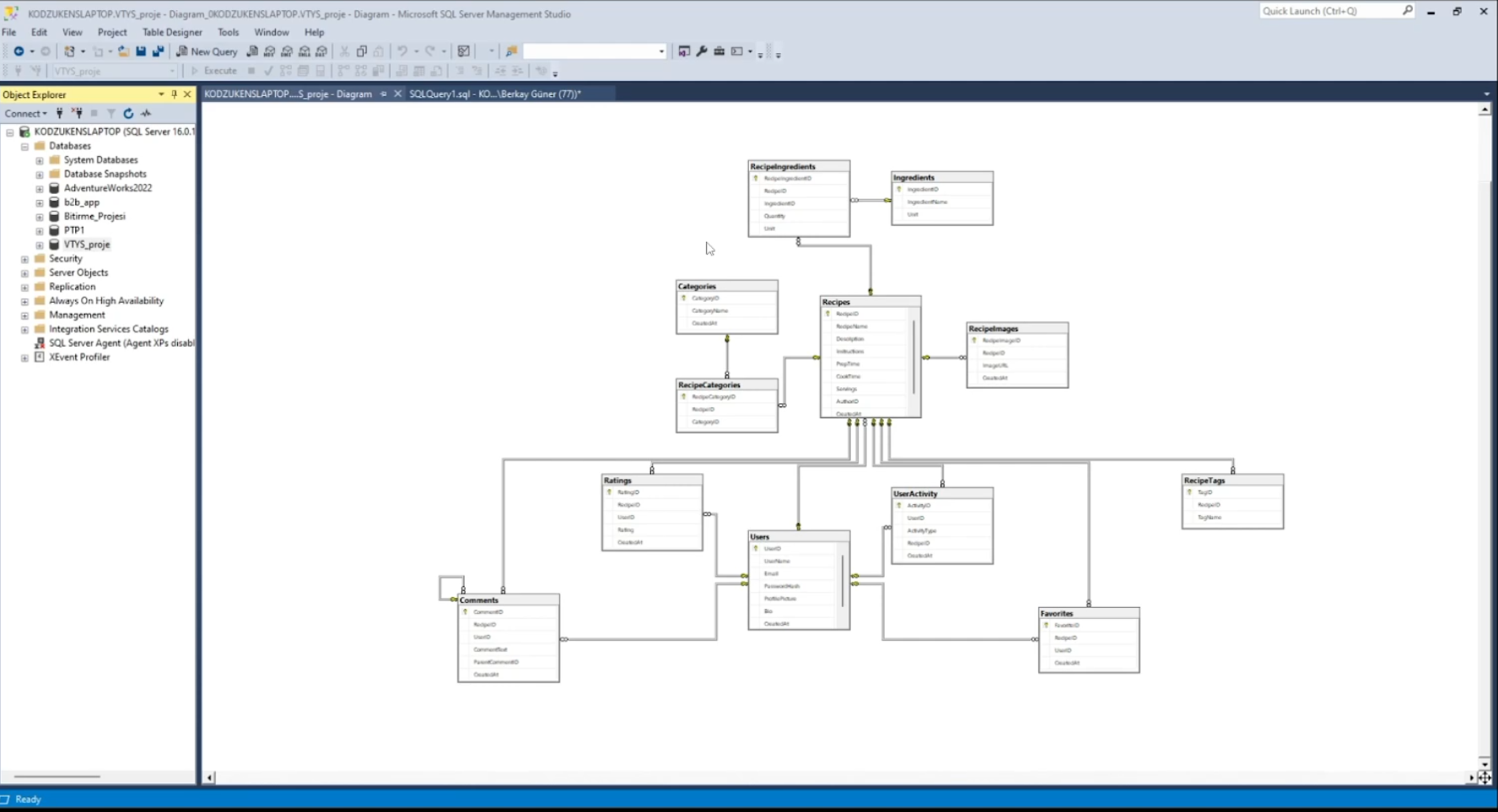
Bu proje, yemek tariflerinin kaydedilmesi, kullanıcı yorumları ve puanlama sistemleri gibi özellikleri bir araya getirerek kullanıcı dostu bir uygulama sunmayı amaçlamıştır. Kullanıcılar, tarif oluşturabilir, malzeme ekleyebilir, yorum yapabilir ve başkalarının tariflerine puan verebilir. Proje ayrıca verimli bir veri tabanı tasarımı ve modern bir arayüz ile uygulama geliştirme sürecini kapsamıştır.

1. Tasarladığınız veri tabanı mimarisinde hangi tablo ve ilişkileri kullanıldığınızı açıklayınız. (10 p)

Kullanılan Tablolar ve İlişkiler:

* **Users: Kullanıcı bilgilerini tutar. (Recipe ve Comments ile ilişkili)**
* **Recipes: Tarif bilgilerini saklar. (Users, Categories, ve Comments ile ilişkili)**
* **Ingredients: Malzeme bilgilerini tutar. (RecipeIngredients ile ilişkili)**
* **RecipeIngredients: Tariflerin malzemeleri ile ilişkisini tanımlar.**
* **Comments: Kullanıcı yorumlarını saklar. (ParentCommentID ile alt yorum sistemini destekler.)**
* **Ratings: Tarif puanlamalarını tutar.**
* **Categories: Tarif kategorilerini saklar.**
* **RecipeCategories: Tarif-kategori ilişkisini tanımlar.**
* **Favorites: Kullanıcıların favori tariflerini kayıt eder.**

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



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

Users Tablosu:

CREATE TABLE Users (

UserID INT IDENTITY(1,1) PRIMARY KEY,

UserName NVARCHAR(50) NOT NULL,

Email NVARCHAR(100) UNIQUE NOT NULL,

PasswordHash NVARCHAR(255) NOT NULL,

CreatedAt DATETIME DEFAULT GETDATE()

);

Recipes Tablosu:

CREATE TABLE Recipes (

RecipeID INT IDENTITY(1,1) PRIMARY KEY,

RecipeName NVARCHAR(100) NOT NULL,

AuthorID INT,

CreatedAt DATETIME DEFAULT GETDATE(),

FOREIGN KEY (AuthorID) REFERENCES Users(UserID) ON DELETE SET NULL

);

1. 5 adet DML (update, insert, delete) içeren kodları yazılmalıdır. (10 p)

INSERT INTO Users (UserName, Email, PasswordHash) VALUES ('Ali Demir', 'ali@example.com', 'hashed\_password');

INSERT INTO Recipes (RecipeName, AuthorID) VALUES ('Çikolatalı Kek', 1);

UPDATE Recipes SET RecipeName = 'Vegan Kek' WHERE RecipeID = 1;

DELETE FROM Comments WHERE CommentID = 10;

DELETE FROM Favorites WHERE FavoriteID = 5;

1. 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. (10 p) Örnek:

-- Tüm veritabanı: SELECT \*; -- Tüm tariflerin listesi: SELECT \* FROM Recipes; -- Tüm değerlendirmelerin listesi: SELECT \* FROM Ratings; -- Belirli bir kullanıcının yazdığı yorumlar: SELECT \* FROM Comments WHERE AuthorID = 1; -- Belirli bir kullanıcının eklediği tarifler: SELECT RecipeName FROM Recipes WHERE AuthorID = 1; -- Tariflerin pişme sürelerini listeleme: SELECT CookTime FROM Recipes; -- Malzeme bilgilerini listeleme: SELECT IngredientName FROM Ingredients; -- Kullanıcı isimlerini listeleme: SELECT UserName FROM Users; -- En yüksek puanlanan tarifler: SELECT RecipeID, AVG(Rating) AS AverageRating FROM Ratings GROUP BY RecipeID HAVING AVG(Rating) > 4; -- Kategoriye göre tarifleri listeleme: SELECT r.RecipeName, c.CategoryName FROM Recipes r JOIN RecipeCategories rc ON r.RecipeID = rc.RecipeID JOIN Categories c ON rc.CategoryID = c.CategoryID;

1. 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)
2. Eğer veri tabanı bağlama işlemini gerçekleştirmemiş iseniz VTYS sistemlerinde Transaction nedir açıklayınız ve çalışmanızdan bir Transaction örneği veriniz. (10 p)

Transaction, bir grup SQL işleminden oluşan ve hepsinin bir bütün olarak işlendiği bir birimdir.

BEGIN TRANSACTION;

INSERT INTO Users (UserName, Email, PasswordHash) VALUES ('Test User', 'test@example.com', 'test');

DELETE FROM Recipes WHERE RecipeID = 1;

IF @@ERROR <> 0

ROLLBACK TRANSACTION;

ELSE

COMMIT TRANSACTION;

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

View:

CREATE VIEW TopRatedRecipes AS

SELECT r.RecipeName, AVG(rt.Rating) AS AverageRating

FROM Recipes r

JOIN Ratings rt ON r.RecipeID = rt.RecipeID

GROUP BY r.RecipeName;

Stored Procedure:

CREATE PROCEDURE GetUserRecipes

@UserID INT

AS

BEGIN

SELECT RecipeName FROM Recipes WHERE AuthorID = @UserID;

END;