

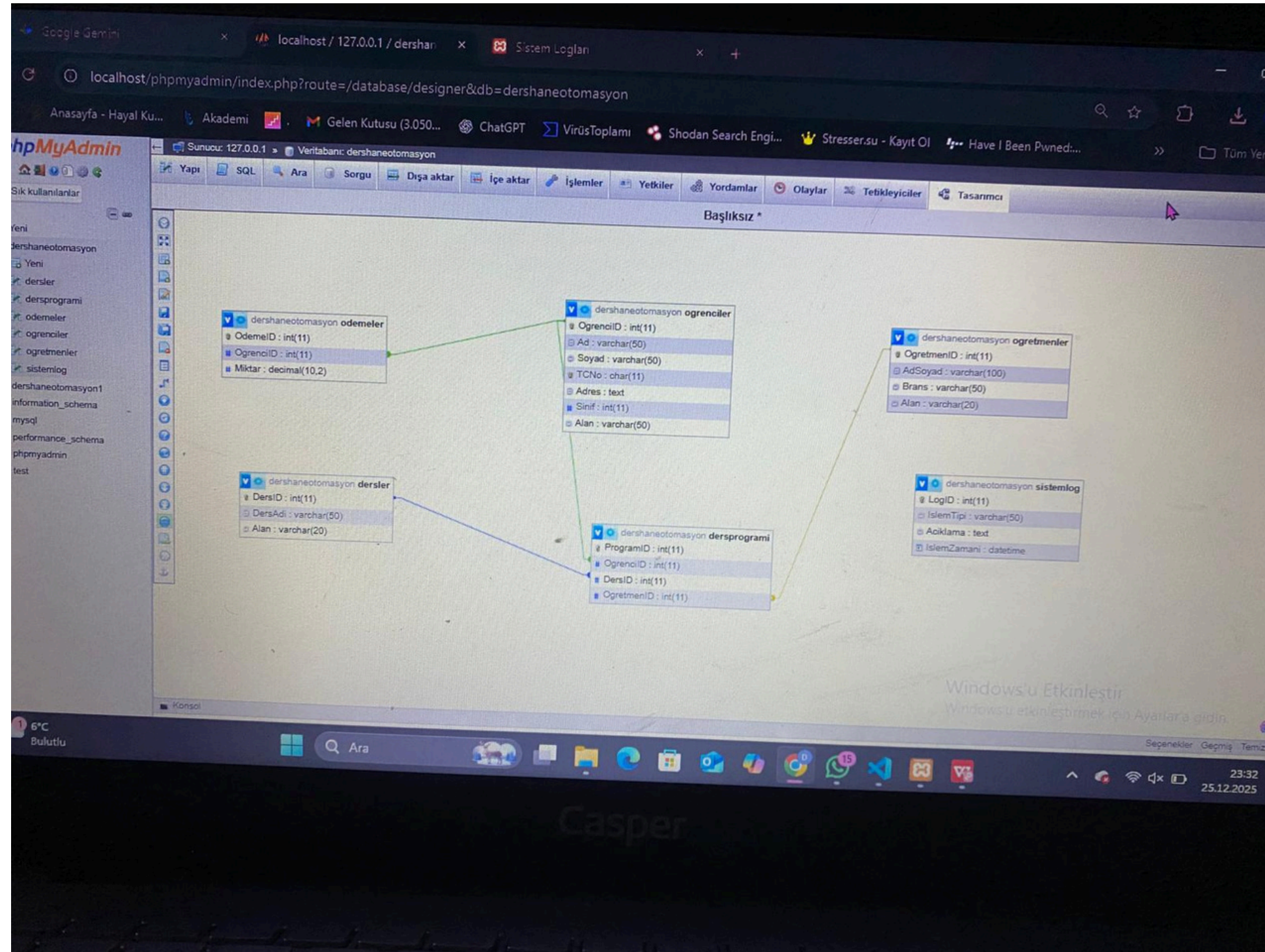
Dershane Otomasyon Sistemi + Web sitesi Raporu

Kevser Akben 242104049

Youtube Link:

Github Link:

E-R diyagramını



Burada görüldüğü üzere öğretmenletr,
öğrenciler, ders programı, ödemeler, dersler birbiri ile bağlantılıdır.
Loglar tablosu da bulunmaktadır.

Ogrenciler: Sistemin "Parent" tablosudur;
PK: OgrenciID,
benzersiz alan: TCNo.

Odemeler: Öğrenci varlığına bağlı
zayıf varlıktır; PK: OdemeID, FK: OgrenciID.

Dersler: Akademik müfredat tanımlarını içerir.

Ogretmenler: Personel envanteri ve uzmanlık alanlarını yönetir.

DersProgrami: Ogrenciler, Dersler ve Ogretmeler arasındaki Many-to-Many (N:M) ilişkiyi normalize eden tablodur.

SistemLog: Tüm DML operasyonlarını asenkron olarak kaydeden güvenlik tablosudur.

Normalizasyon Analizi (3NF Kanıtı)

Sistemin veritabanı mimarisi, veri anomalilerini (ekleme, silme, güncelleme hataları) engellemek ve veri tekrarını (Redundancy) minimize etmek amacıyla 3. Normal Form (3NF) kurallarına göre normalize edilmiştir. Aşağıda bu sürecin teknik kanıtları yer almaktadır:

- Normal Form (1NF) Analizi

Kural: Her sütun atomik (bölünemez) değerler içermeli ve tekrarlayan gruplar bulunmamalıdır.

Kanıt: Öğrenciler tablosunda isim ve soyisim gibi veriler ayrıştırılmış; bir öğrenciye ait birden fazla ödeme veya ders bilgisi tek bir hücreye yazılmak yerine ilişkili tablolara (Odemeler, DersProgramı) dağıtılmıştır.

- Normal Form (2NF) Analizi

Kural: Tablo 1NF olmalı ve anahtar olmayan tüm sütunlar Birincil Anahtara (Primary Key) tam fonksiyonel bağımlı olmalıdır.

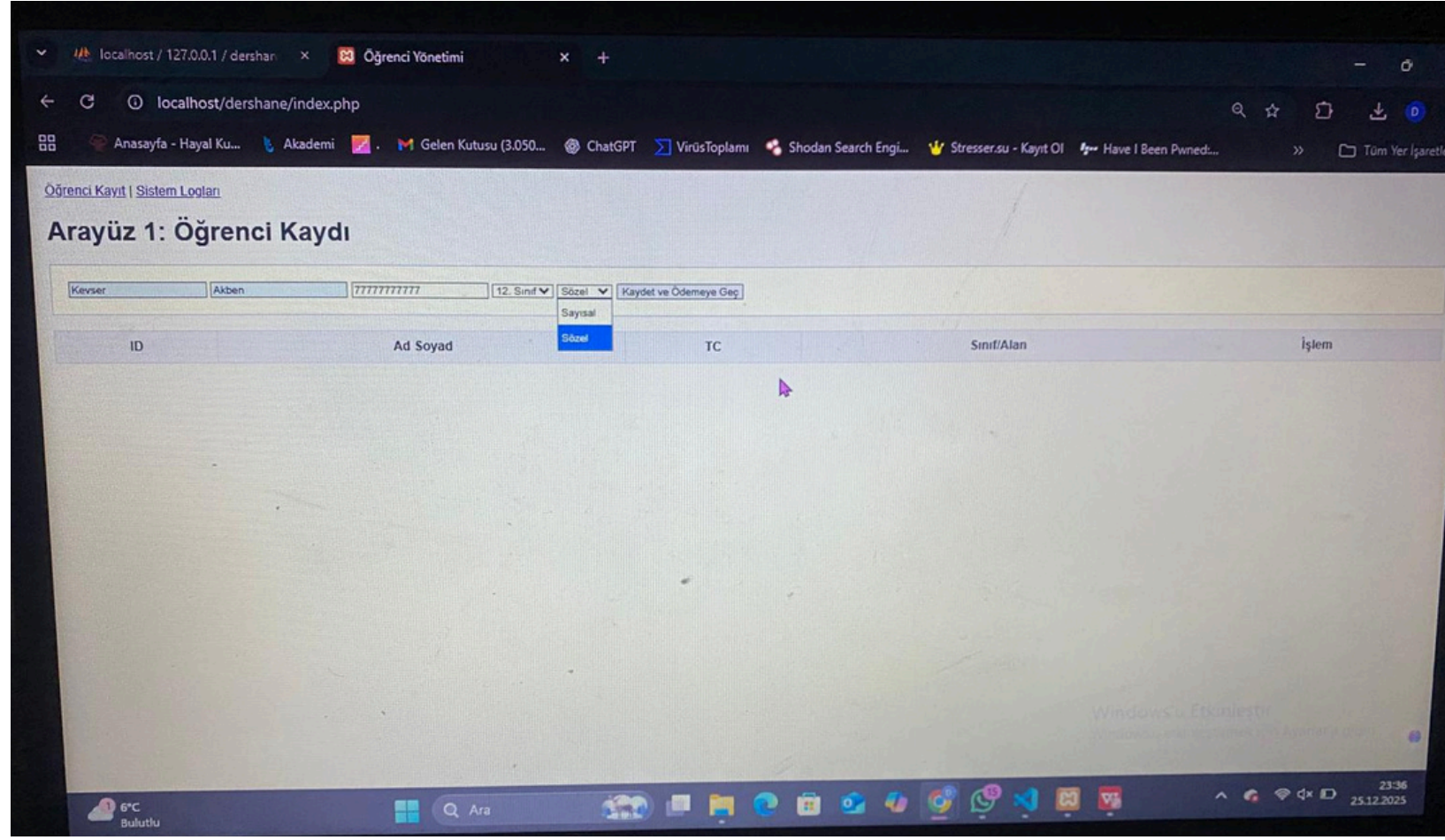
Kanıt: Odemeler tablosu 2NF standartlarındadır; çünkü ödeme miktarı doğrudan öğrenci kimliğine değil, her ödeme için benzersiz olan OdemeID anahtarına tam bağımlıdır. Kısmi bağımlılıklar giderilerek her veri kendi mantıksal kümesine yerleştirilmiştir.

- Normal Form (3NF) Analizi

Kural: Tablo 2NF olmalı ve geçişli bağımlılıklar (transitive dependencies) bulunmamalıdır. Yani anahtar olmayan bir sütun, başka bir anahtar olmayan sütuna bağımlı olmamalıdır.

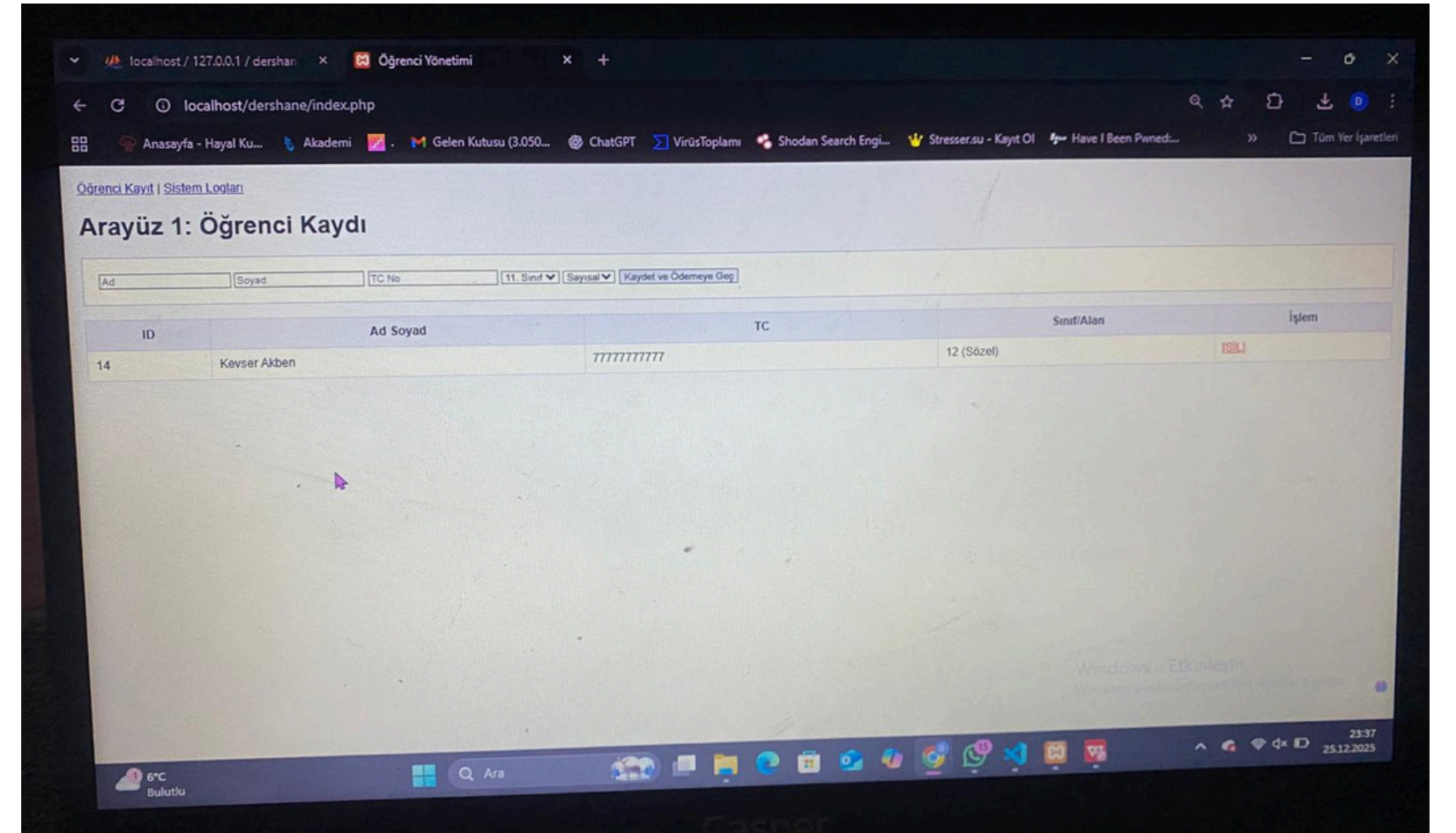
Kanıt: En kritik kanıt DersProgramı tablosundadır. Bu tabloda öğretmenin branşı veya dersin adı gibi bilgiler yer almaz; bu bilgiler ÖğretmenID ve DersID yabancı anahtarları (FK) üzerinden çekilir. Böylece anahtar olmayan sütunlar arası bağımlılık koparılarak tam normalizasyon sağlanmıştır

CRUD İşlemleri ve Arayüz Tasarımı

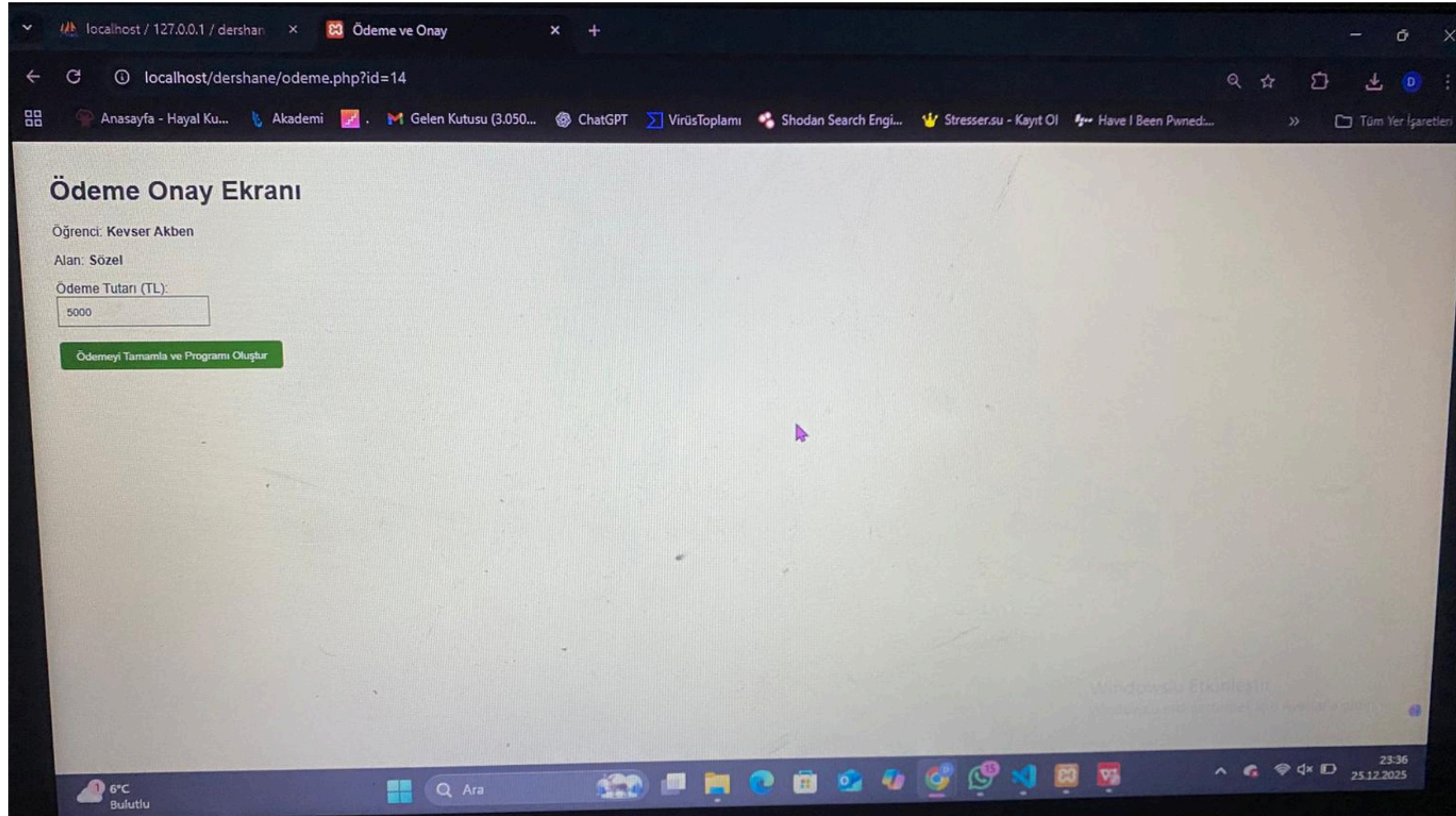


Öğrenci için önceden belirlenen fiyatı ödemek üzere ödeme ekranına ataması yani geçişi yapılmaktadır. Aynı zamanda burada listeme(read) yapısının doğru çalıştığı da gösterilmiştir.

bu iki resimde öğrenci ekleme(creat) sisteminin doğru çalıştığı gösterilmiştir. Öğrencinin, isim, soyisim, T.C., alanı ve sınıf bilgileri alınmıştır.



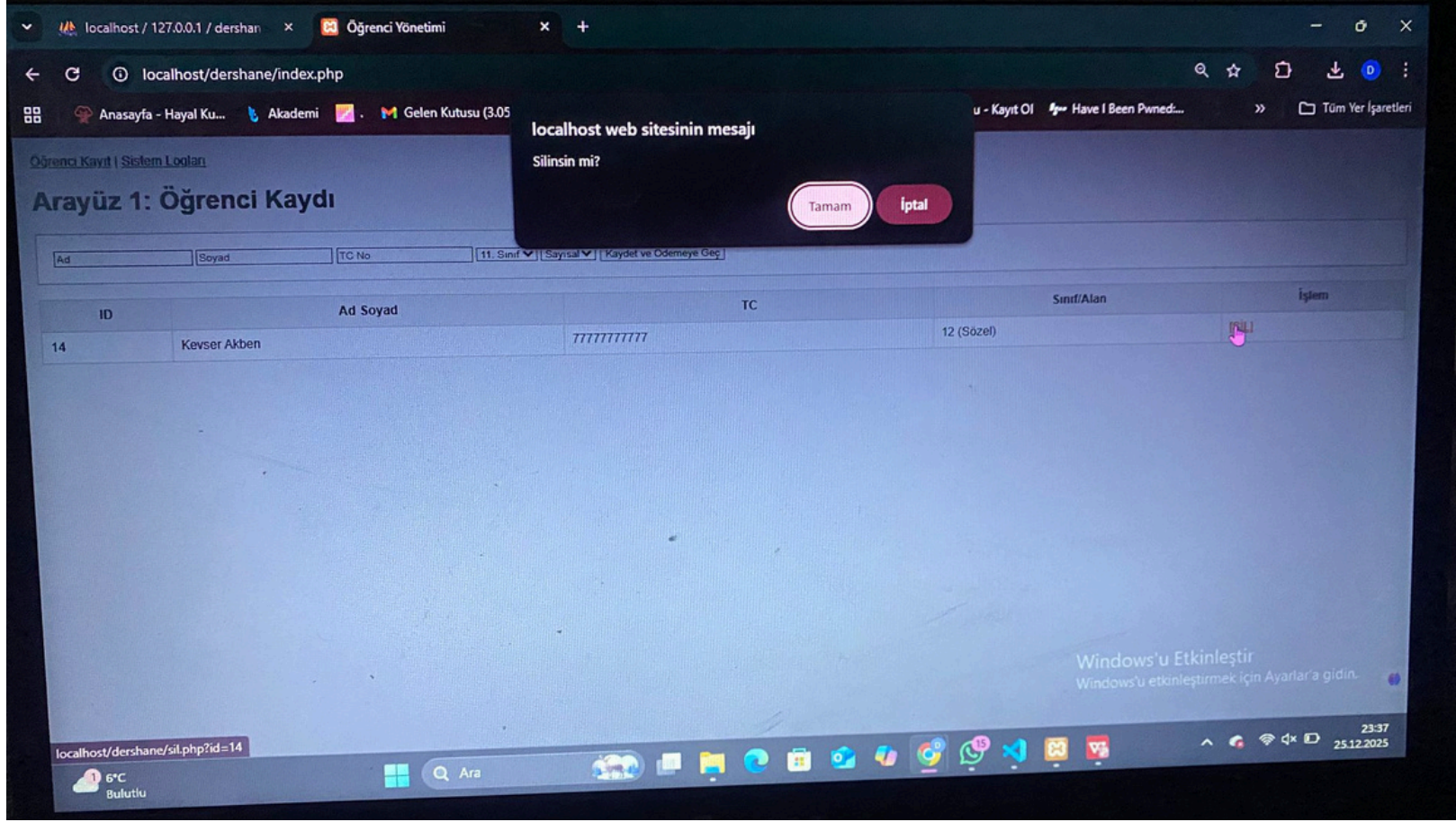
CRUD İşlemleri ve Arayüz Tasarımı



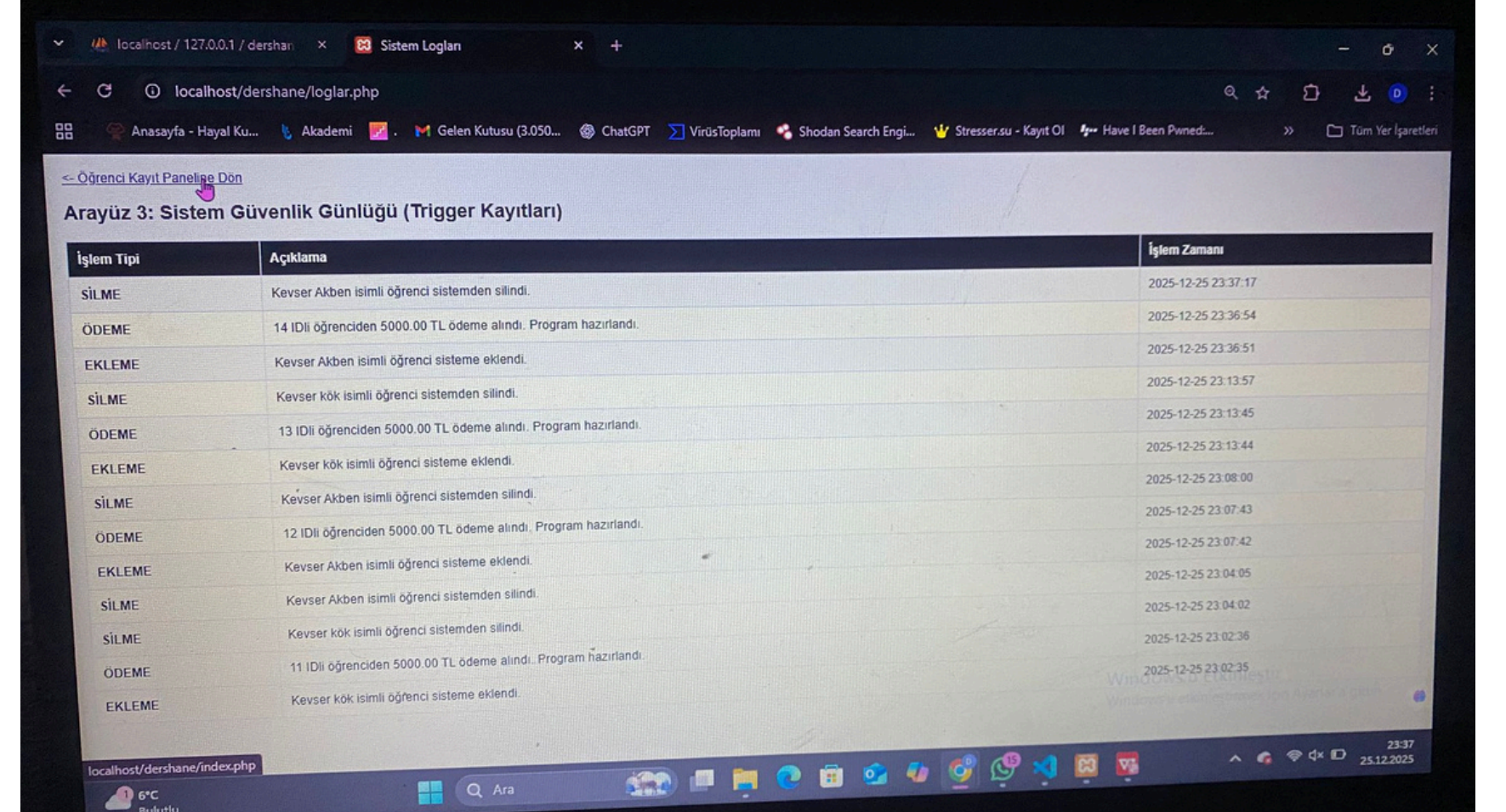
Az önce bahsedilen ödeme ekranı

CRUD İşlemleri ve Arayüz Tasarımı

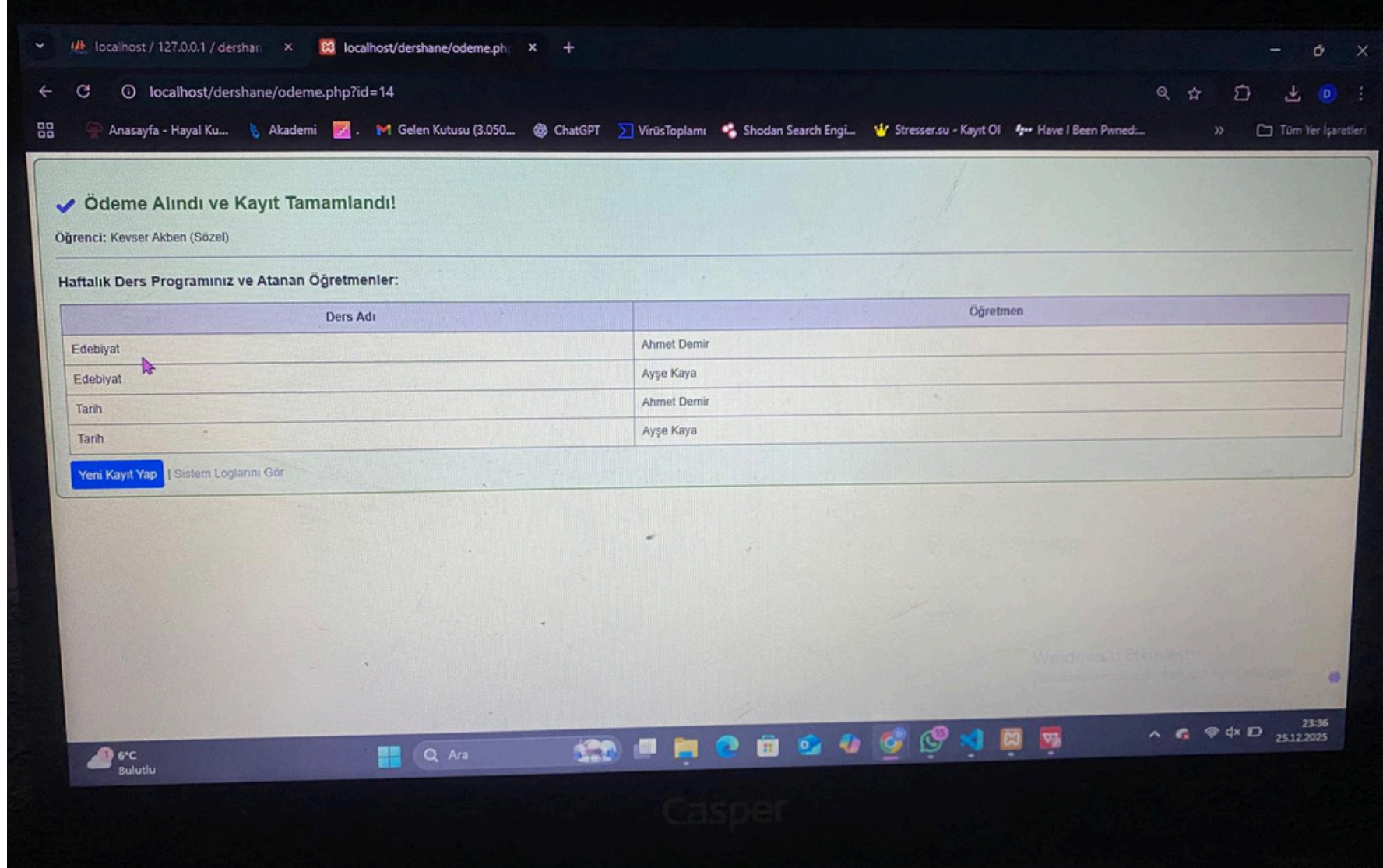
Ve bu kısımda da silme(delete) yapısının onay penceresi ile birlikte bu yapının doğru çalıştığını görmekteyiz.



Aynı zamanda loglarında düzenli olarak kayıtları tuttuğunu ve ekleme - silme işlemlerinin yapılabildiğini yine görmekteyiz.



Transaction ve Otomatik Atama



Ödeme tahsil edildiği an başlatılan Transaction süreciyle, öğrencinin alanına (Sayısal/Sözel) uygun dersler ve öğretmenler JOIN sorguları aracılığıyla otomatik olarak atanmaktadır. Bu işlem, "Atomicity" kuralı gereği ya tamamen gerçekleşmekte ya da hata anında geri alınmaktadır.

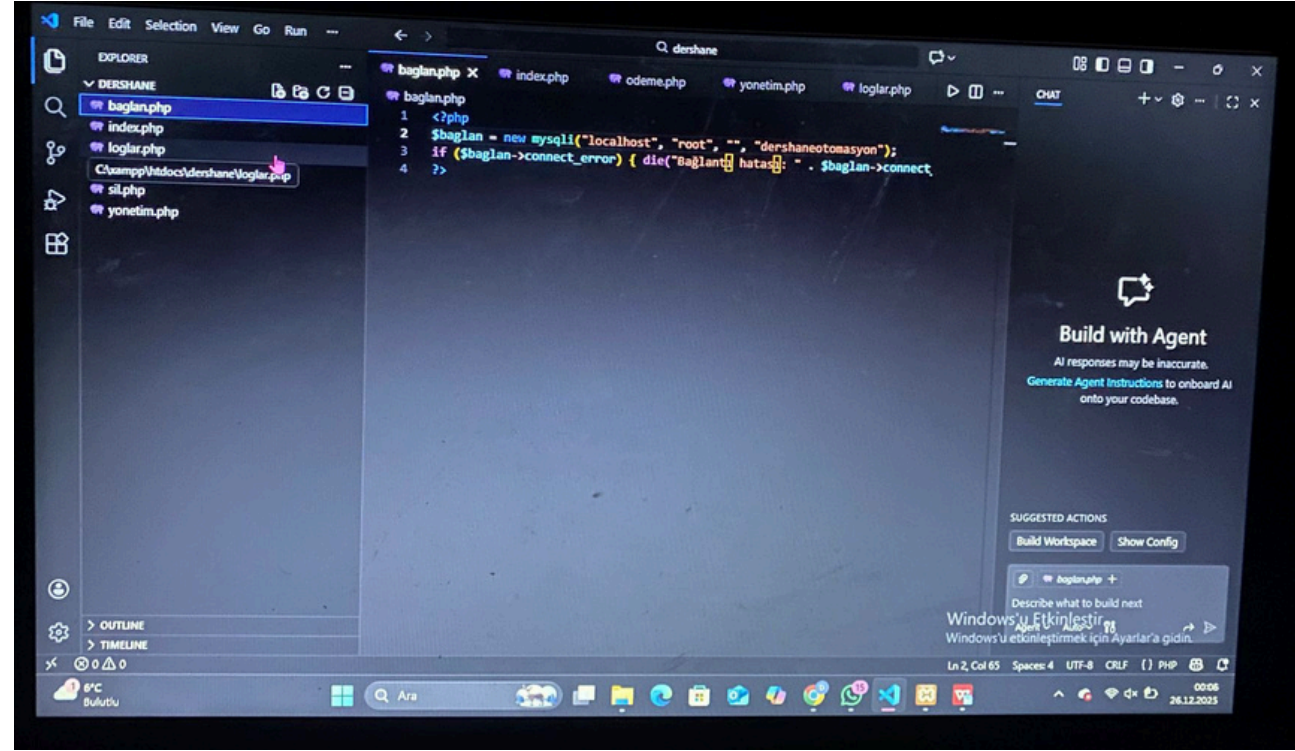
Loglama ve Trigger Mekanizması

Arayüz 3: Sistem Güvenlik Günlüğü (Trigger Kayıtları)

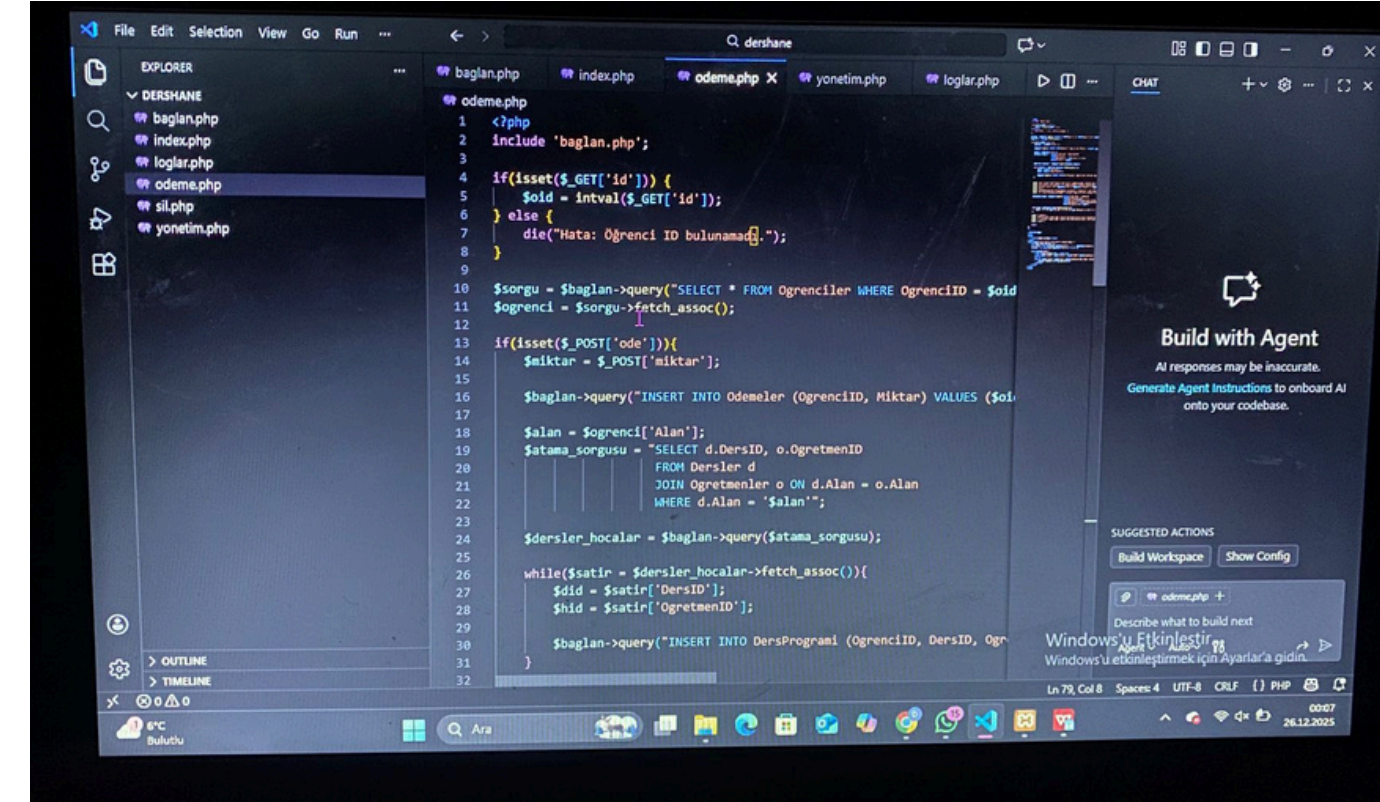
İşlem Tipi	Açıklama	İşlem Zamanı
SİLME	Kevser Akben isimli öğrenci sistemden silindi.	2025-12-25 23:37:17
ÖDEME	14 ID'li öğrenciden 5000.00 TL ödeme alındı. Program hazırlandı.	2025-12-25 23:36:54
EKLEME	Kevser Akben isimli öğrenci sisteme eklendi.	2025-12-25 23:36:51
SİLME	Kevser kök isimli öğrenci sistemden silindi.	2025-12-25 23:13:57
ÖDEME	13 ID'li öğrenciden 5000.00 TL ödeme alındı. Program hazırlandı.	2025-12-25 23:13:45
EKLEME	Kevser kök isimli öğrenci sisteme eklendi.	2025-12-25 23:13:44
SİLME	Kevser Akben isimli öğrenci sistemden silindi.	2025-12-25 23:08:00
ÖDEME	12 ID'li öğrenciden 5000.00 TL ödeme alındı. Program hazırlandı.	2025-12-25 23:07:43
EKLEME	Kevser Akben isimli öğrenci sisteme eklendi.	2025-12-25 23:07:42
SİLME	Kevser Akben isimli öğrenci sistemden silindi.	2025-12-25 23:04:05
SİLME	Kevser kök isimli öğrenci sistemden silindi.	2025-12-25 23:04:02
ÖDEME	11 ID'li öğrenciden 5000.00 TL ödeme alındı. Program hazırlandı.	2025-12-25 23:02:36
EKLEME	Kevser kök isimli öğrenci sisteme eklendi.	2025-12-25 23:02:35

Veritabanı motoru seviyesinde çalışan Trigger (Tetikleyici) mekanizması; her EKLEME ve SİLME işlemini anlık olarak SistemLog tablosuna kaydetmektedir. Görselde görüldüğü üzere, sistem üzerinden yapılan her hareket tarih ve işlem tipiyle beraber raporlanmaktadır.

KOD YAPISI VE ALGORİTMİK MANTIK



Projenin yazılım mimarisi, istemci tarafında HTML5/CSS3 ve JavaScript, sunucu tarafında ise PHP (Hypertext Preprocessor) dilleri kullanılarak geliştirilmiştir. Veritabanı yönetim sistemi olarak MySQL (MariaDB) tercih edilmiştir.



Sistem, veri bütünlüğünü korumak adına ACID prensiplerine uygun bir Transaction yapısı kullanır; özellikle ödeme onayında öğrencinin alanına göre öğretmen ve ders atamaları JOIN sorguları ile otomatik olarak gerçekleştirilir. Veritabanı seviyesinde tanımlanan Triggerlar, her türlü veri hareketini (INSERT/DELETE) yazılım katmanından bağımsız olarak asenkron şekilde loglar. Hata yönetimi tarafında kullanılan die() fonksiyonları ve veritabanı kısıtlamaları (UNIQUE, NOT NULL), mükerrer veya hatalı veri girişini engelleyerek sistemin kararlı ve güvenli çalışmasını garanti altına alır.

