



T.C
KOCAELİ SAęLIK VE TEKNOLOJİ ÜNİVERSİTESİ
MÜHENDİSLİK VE DOęA BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİęİ

PROJE KONUSU: SAYISAL TASARIM LOJİK DEVRELER

HAZIRLAYANLAR

ÖęRENCİ ADI: BAŞAR BURAK ÜNAL
ÖęRENCİ NUMARASI: 220501025
ÖęRENCİ ADI: EFE ÖZTAŞ
ÖęRENCİ NUMARASI: 220501014

DERS SORUMLUSU:
DR. ÖęR. ÜYESİ ERCAN ÖLÇER

TARİH: 2.06.2024

1 GİRİŞ

1.1 Projenin amacı

- Bu projenin amacı, kullanıcıların temel mantık kapılarını kullanarak lojik devreler oluşturabileceği bir simülasyon yazılımı geliştirmektir. Bu uygulama, kullanıcıların farklı mantıksal kapıları, anahtarları ve LED'leri kullanarak lojik devreler oluşturmalarına ve bu devreleri simüle etmesine olanak tanır.
- **Projede Gerçekleştirilmesi Beklenenler:**
- - Kullanıcı dostu bir arayüz.
- - Kullanıcıların farklı türlerde mantık kapıları (AND, OR, NOT, NAND, NOR, XOR, XNOR, BUFFER) ekleyebilmesi.
- - Kullanıcıların anahtar ve LED ekleyerek devrelerini test edebilmesi.
- - Kullanıcıların doğruluk tablosunu görebilmesi.
- - Mantık kapılarının bağlantılarının (kabloların) oluşturulabilmesi.
- - Kullanıcıların oluşturduğu devreyi çalıştırabilmesi ve sonuçları gözlemleyebilmesi.
- - Kullanıcıların devrelerini sıfırlayabilmesi ve tekrar düzenleyebilmesi.

2 GEREKSİNİM ANALİZİ

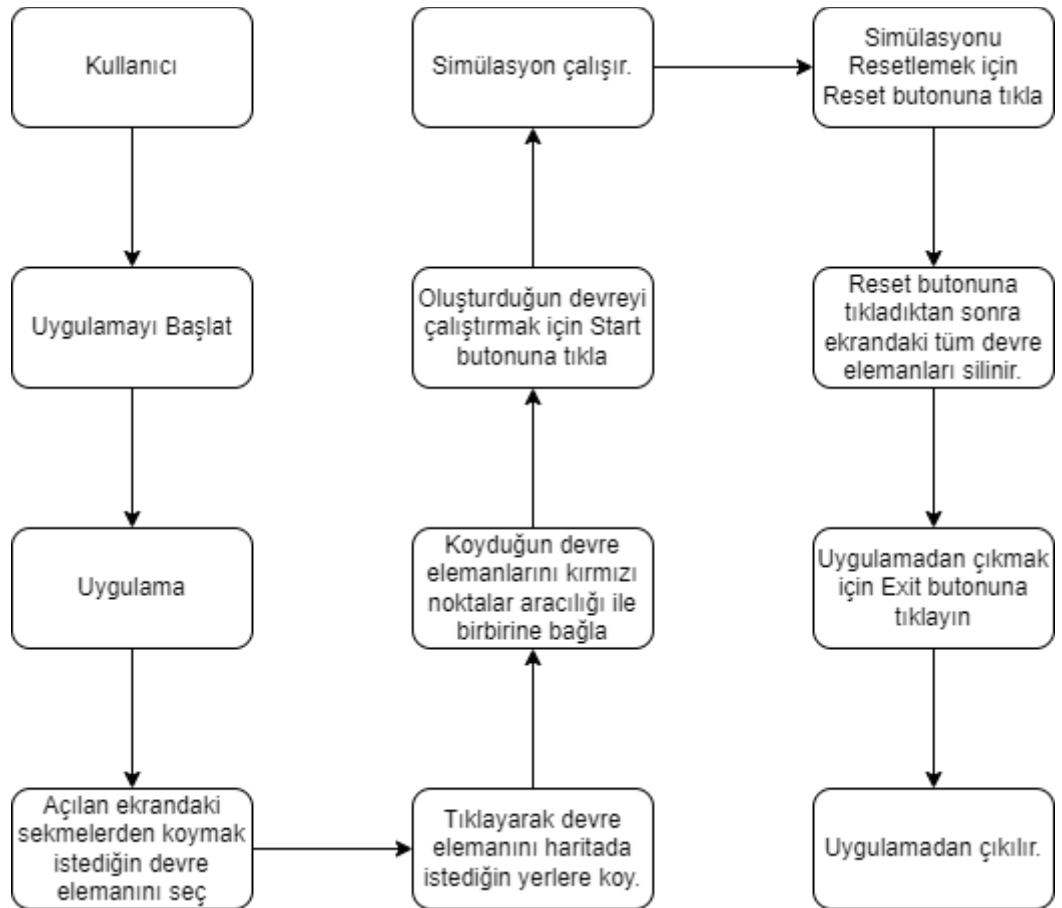
2.1 Arayüz gereksinimleri

- Ana menü: Mantık kapıları, anahtar ekleme, LED ekleme, simülasyonu başlatma/durdurma ve sıfırlama seçeneklerini içermelidir.
- Mantık kapıları menüsü: Farklı türlerde mantık kapılarının görselleri ve isimleri yer almalıdır.
- Simülasyon alanı: Kullanıcıların mantık kapılarını yerleştirebileceği, anahtar ve LED ekleyebileceği bir ızgara tabanlı arka plan sağlanmalıdır.
- Durum göstergeleri: LED'lerin ve anahtarların durumunu (açık/kapalı) görsel olarak gösteren simgeler olmalıdır.
- **Donanım Arayüzü Gereksinimleri:**
- Fare ve klavye ile tam uyumluluk.

2.2 Fonksiyonel gereksinimler

- Mantık kapılarının (AND, OR, NOT, NAND, NOR, XOR, XNOR, BUFFER) eklenmesi ve doğru çalışması.
- Anahtarların (Switch) eklenmesi ve tıklanarak durumlarının değiştirilebilmesi.
- LED'lerin eklenmesi ve mantık kapılarının çıkışları ile bağlantılı olarak durumlarının gösterilmesi.
- Kullanıcıların kapıları, anahtarları ve LED'leri bağlayabilmesi için kabloların oluşturulması.
- Simülasyonun başlatılması, durdurulması ve sıfırlanması.

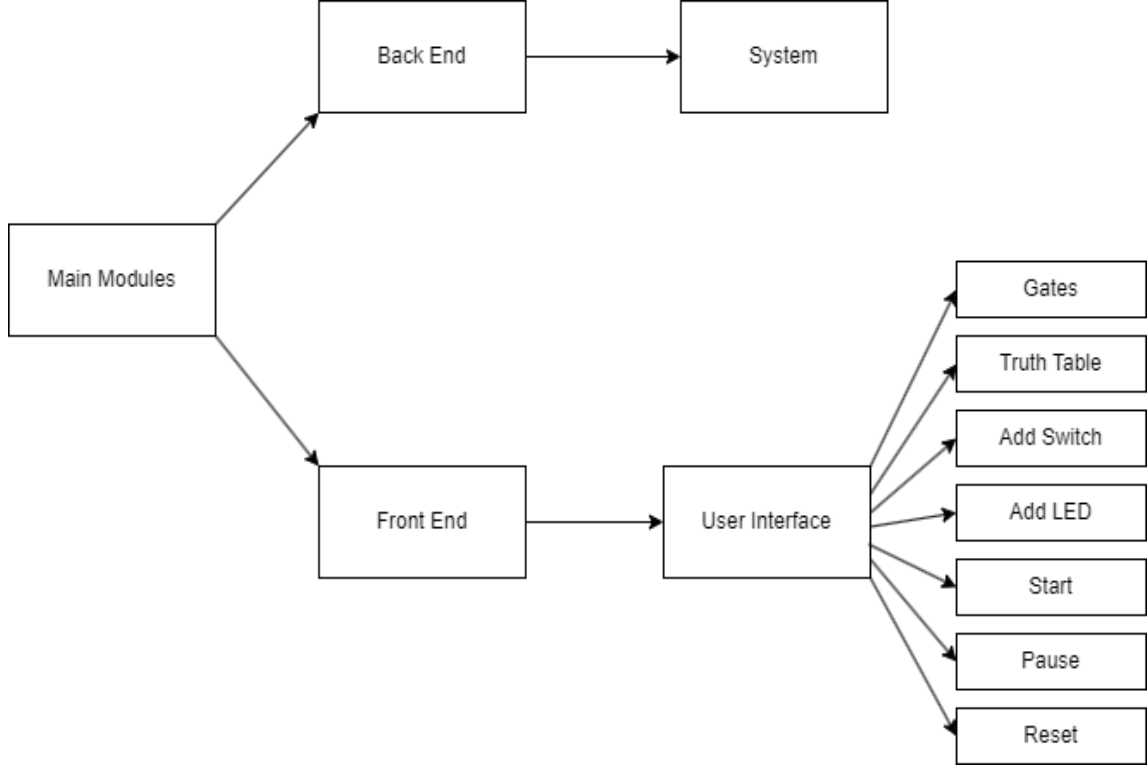
2.3 Use-Case diyagramı



3 TASARIM

3.1 Mimari tasarım

- Lojik Devreler modüler bir mimari kullanılarak tasarlanmıştır. Ana modüller; Kullanıcı Arayüzü, Mantık Kapıları, Doğruluk Tablosu, LED ve Anahtar içerir.



3.2 Kullanılacak teknolojiler

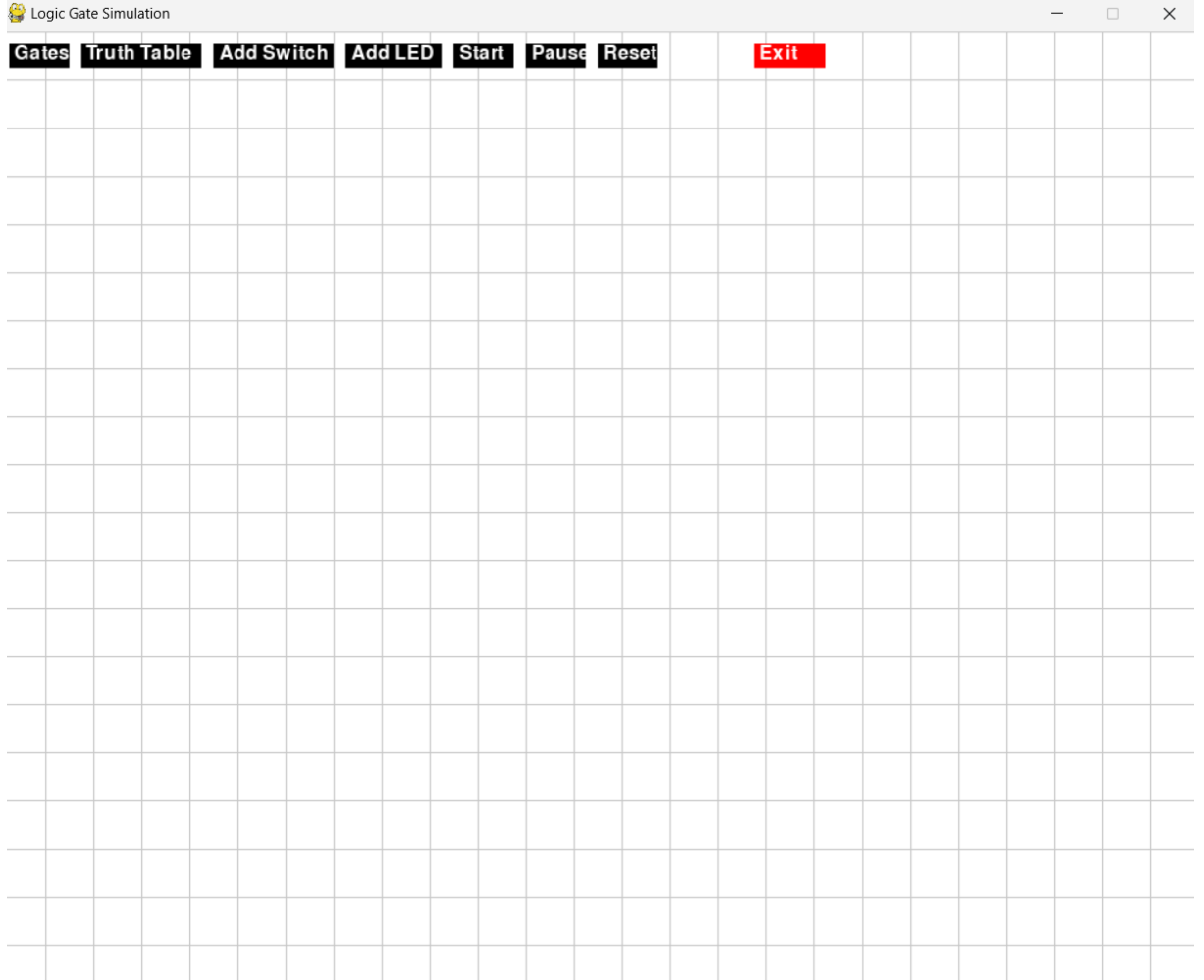
- Yazılım Dili: Python
- Kullanılan kütüphane: Pygame

3.3 Veri tabanı tasarımı

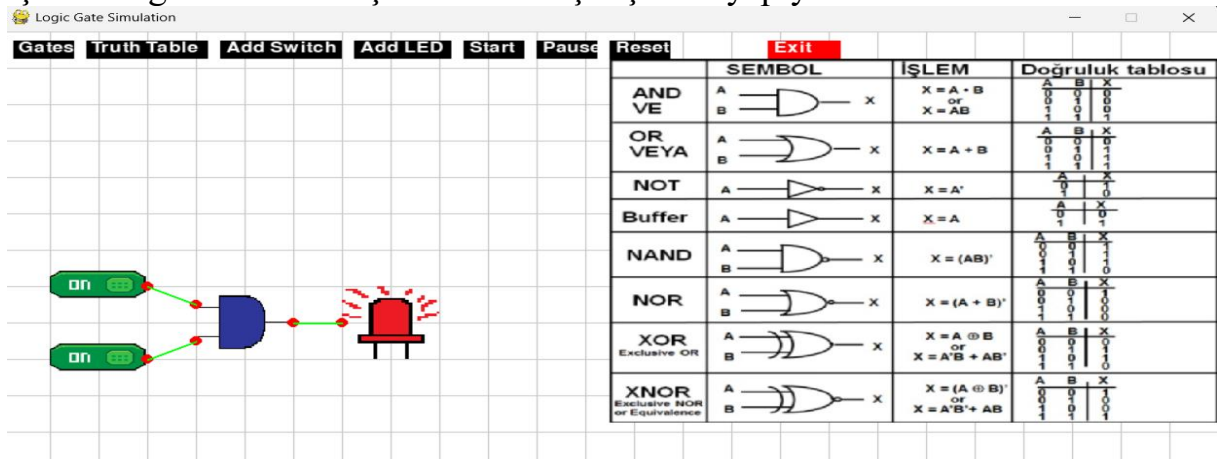
- Veri tabanı tasarımı yoktur.
- ER Diyagramı eklenmesi Veri tabanı kullanımı olmadığı için ER diyagramı oluşturulmamıştır.

3.4 Kullanıcı arayüzü tasarımı

- Programı çalıştırdığımızda karşımıza aşağıdaki görseldeki gibi bir ekran çıkacak. Bu ekranda yukarıda bulunan başlıklardan işlemleri gerçekleştireceğiz.



Öncelikle starta tıklıyoruz, Gates başlığından hangi lojik kapıyı kullanacağımızı seçiyoruz seçtikten sonra anahtarları ve LED'i ekliyoruz. Kabloları bağlayıp lojik devreyi tamamlıyoruz daha sonrasında Truth Table ile devrenin doğru çalışıp çalışmadığını kontrol edebiliyoruz. Çıkış yapmak için de sağ üstten exit tuşuna basarak çıkışımızı yapıyoruz.



4 UYGULAMA

4.1 Kodlanan bileşenlerin açıklamaları

- LogicGate Sınıfı: Bu sınıf, tüm mantık kapılarının temel sınıfıdır.
- AND, OR, NOT, NAND, NOR, XOR, XNOR, BUFFER Sınıfları: Mantıksal işlemleri gerçekleştirir.
- Switch Sınıfı: Anahtarları temsil eder.
- LED Sınıfı: LED'leri temsil eder.
- Wire Sınıfı: Bağlantıları, kabloları temsil eder.

4.2 Görev dağılımı

- Ekip olarak, sayısal tasarım lojik devreler projesinin her bir aşamasında birlikte ilerledik; Lojik devrelerin tasarımı ve geliştirilmesi gibi işlemleri birlikte gerçekleştirdik.
- Rapor ekip üyeleri arasında iletişimi koparmadan birlikte hazırlanmıştır.

4.3 Karşılaşılan zorluklar ve çözüm yöntemleri

- Lojik devrelerin tasarımı için kütüphane arayışı.
- Lojik kapıları tasarlama.
- Lojik kapılarının giriş ve çıkışlarının doğru bağlanması ve işlenmesi.

4.4 Proje isterlerine göre eksik yönler

- Bazı sınıfsal işleyişleri gerçekleyememek, Anahtarın tek tıkla bazen değişmemesi gibi.

5 TEST VE DOĞRULAMA

5.1 Yazılımın test süreci

- Herhangi bir test uygulaması geliştirilmemiştir. Yazılan kodda testler kullanıcı girişleri ile sağlanmaktadır.

5.2 Yazılımın doğrulanması

- Test sonucunda eksik ya da hatalı çalışan bileşenler:
- Anahtarın bazen tek tıkla yeşile dönmemesi.

KAYNAKÇA

<https://www.pygame.org/>

<https://www.geeksforgeeks.org/pygame-tutorial/>

<https://www.geeksforgeeks.org/logic-gates-in-python/>

<https://www.geeksforgeeks.org/python-hasattr-method/>

Github: <https://github.com/EOztas/Logic-Gates-Simulator>