**Socket İle İletişim Projesi Raporu**

**1. Proje Tanımı**

Bu proje, bilgisayar ağları üzerinde iletişim kurmayı sağlayan socket programlamasına yönelik basit bir örnektir. Projenin ana amacı, bir istemci ve bir sunucu arasında veri iletimini sağlamak ve bu süreci basit bir kullanıcı arayüzü ile kontrol etmektir.

**2. Kullanılan Teknolojiler**

* **Python Programlama Dili:** Projenin temelinde Python kullanılmıştır.
* **Socket Modülü:** İstemci ve sunucu arasındaki iletişimi yönetmek için socket modülü kullanılmıştır.

**3. Proje Aşamaları**

**3.1 İstemci (Client) Tarafı**

İstemci tarafı, kullanıcının bir metin girişi yapmasını sağlar ve bu girişi bir byte dizisine dönüştürerek sunucuya gönderir. Ardından, sunucudan gelen cevabı alır ve ekrana yazdırır. İstemci, kullanıcıdan gelen veriyi UTF-8 formatında gönderir ve aldığı veriyi yine bu formatta işler.

import socket

s = socket.socket()

port = 12345

s.connect(('127.0.0.1', port))

input\_string = input("Göndermek istediğiniz veriyi girin -> ")

s.sendall(bytes(input\_string, 'utf-8'))

print(s.recv(1024).decode())

s.close()

**3.2 Sunucu (Server) Tarafı**

Sunucu tarafı, belirli bir portu dinleyerek istemci bağlantılarını kabul eder. Bağlantı kurulduktan sonra istemciden gelen veriyi alır, işler ve geri gönderir. Sunucu, gelen ve gönderilen verileri UTF-8 formatında işler.

import socket

s = socket.socket()

print("Socket başarıyla oluşturuldu")

port = 12345

s.bind(('', port))

print("Soket", port, "noktasına bağlandı")

s.listen(5)

print("Soket dinleme modunda")

while True:

c, addr = s.accept()

print('Adresten bağlantı alındı:', addr)

data = c.recv(1024).decode()

if not data:

break

c.sendall(bytes(data, 'utf-8'))

c.close()

**4. Proje Sonuçları ve Analizi**

Bu basit socket iletişim projesi, istemci ve sunucu arasında güvenilir bir veri iletişimi sağlar. Proje, temel socket programlamaya giriş yaparak basit bir uygulama geliştirmeyi amaçlar. Projenin katmanlı yapısı, daha karmaşık senaryolar için uygulama geliştirmeyi kolaylaştırır.

Proje aynı zamanda şu avantajları sağlar:

* Kullanıcı dostu arayüz: İstemci tarafında kullanıcıdan veri almak için basit bir metin girişine yer verilmiştir.
* Temel güvenlik: Veri iletimi sırasında UTF-8 formatı kullanılarak veri bütünlüğü sağlanmıştır.

Bu projenin geliştirilmesi ve özelleştirilmesi, daha karmaşık ağ uygulamaları için bir temel oluşturabilir. Örneğin, şifreleme eklenerek güvenli iletişim sağlanabilir veya çoklu istemcileri destekleyen bir sunucu uygulamasına geçiş yapılabilir.

**Alperen Keleş 404709**