## T.C K A R A B Ü K Ü N İ V E R S İ T E S İ

**MüHENDİSLİK FAKÜLTESİ**

**YAZILIM MÜHENDİSLİĞİ**



**YAZILIM MÜHENDİSLİĞİ YAZILIM İNŞASI DERSİ PROJE ÖDEVİ**

**TAŞ KAĞIT MAKAS OYUNU GÖRÜNTÜ İŞLEMESİ**

**PROJE SAHİPLERİ**

2310238039 YAĞMUR İNCİ

2310238041 FATMA BERİK

2310238038 İREM NAZ GEÇİKLİ

2310238512 MERYEM ALSALEH

**1. Uygulama Süreci**

### a) Sistem Gereksinimleri

#### i) Donanım Gereksinimleri

* **Bilgisayar:** Minimum çift çekirdekli bir işlemci (Intel i3 veya denk bir model).
* **RAM:** En az 4 GB (8 GB önerilir).
* **Depolama Alanı:** 500 MB boş alan (veritabanı ve uygulama dosyaları için).
* **Ekran:** 1024x768 piksel çözünürlüklü bir monitör.

#### ii) Yazılım Gereksinimleri

* **Python Sürümü:** 3.8 veya daha yüksek.
* **Python Kütüphaneleri:**
  + Flask (web uygulaması için)
  + MySQL-Connector-Python (veritabanı bağlantısı için)
  + Requests (istek gönderimi için, eğer API kullanılacaksa)
* **Veritabanı:** MySQL veya herhangi bir SQL tabanlı veritabanı sistemi.
* **Web Sunucusu:** Apache veya Nginx.
* **İşletim Sistemi:** Windows, macOS veya Linux (Ubuntu 20.04 önerilir).

#### iii) Ek Gereksinimler

* **Ağ Bağlantısı:** Oyunun çoklu kullanıcı modunda oynanabilmesi için yerel ağ veya internet bağlantısı gereklidir.
* **Güvenlik Sertifikası:** HTTPS bağlantısı için SSL/TLS sertifikası (tercihe bağlı olarak).

### b) Kullanıcı Kurulum Kılavuzu

#### i) Sunucu Tarafı Kurulumu

1. **Sunucu Ortamı Hazırlama:**
   * Sunucu olarak kullanılacak bilgisayarda minimum donanım gereksinimlerini sağlayın.
   * İşletim sistemi olarak Linux tabanlı bir sistem tercih ediyorsanız, Ubuntu 20.04 önerilir.
2. **Web Sunucusunu Kurun:**
   * Apache kurulumu için terminalden aşağıdaki komutu çalıştırın:

sudo apt update

sudo apt install apache2

* + Web sunucusunu başlatın ve durumunu kontrol edin:

sudo systemctl start apache2

sudo systemctl status apache2

1. **Python Kurulumu:**

Python’ın 3.8 veya daha yüksek bir sürümünü yükleyin:

sudo apt update

sudo apt install python3 python3-pip

* + Gerekli Python kütüphanelerini yüklemek için:

pip3 install flask mysql-connector-python

1. **Veritabanı Kurulumu:**
   * MySQL sunucusunu yükleyin:

sudo apt install mysql-server

* + MySQL sunucusunu başlatın:

sudo systemctl start mysql

* + Veritabanına bir kullanıcı oluşturun ve yetkileri ayarlayın:

CREATE DATABASE tas\_kagit\_makas;

CREATE USER 'oyun\_kullanici'@'localhost'

IDENTIFIED BY 'sifre';

GRANT ALL PRIVILEGES ON tas\_kagit\_makas.\* TO

FLUSH PRIVILEGES;

1. **Veritabanı Ayarlarını Yapılandırın:**
   * Python uygulamasındaki veritabanı bağlantısı için bir konfigürasyon dosyası oluşturun (örneğin, config.py):
   * **Sunucu Uygulamasını Başlatın:**
   * Flask uygulamasını başlatmak için:

python3 app.py

* + Sunucu çalıştıktan sonra, tarayıcıdan http://localhost:5000 adresine giderek uygulamayı test edin.

1. Uygulama Sunucusunu Başlatın:
   * Uygulamanın kararlı bir şekilde çalışması için gerekli servisleri arka planda başlatın. Flask uygulamasını üretim ortamında çalıştırmak için bir WSGI sunucusu (Gunicorn gibi) kullanabilirsiniz:

pip3 install gunicorn

gunicorn -w 4 app:app

1. Web Sunucusunu Yapılandırın:
   * Apache veya Nginx kullanıyorsanız, Flask uygulaması için bir ters vekil sunucu (reverse proxy) ayarı yapın. Apache için bir örnek konfigürasyon dosyası:

<VirtualHost \*:80>

ServerName tas-kagit-makas.local

ProxyPreserveHost On

ProxyPass / http://127.0.0.1:8000/

ProxyPassReverse / http://127.0.0.1:8000/

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

* + Apache ayar dosyasını etkinleştirin ve web sunucusunu yeniden başlatın:

sudo a2ensite tas-kagit-makas.conf

sudo systemctl reload apache2

8. Uygulama Sunucusu Dağıtımı

* Taş Kağıt Makas oyunu Python ile geliştirilmiştir.
* Kodlar, uygulama sunucusuna sıkıştırılmış dosya (örneğin, .zip veya .tar.gz) formatında yüklenmiştir.
* Gerekli Python paketleri requirements.txt dosyası kullanılarak kuruldu:

pip install -r requirements.txt

9. Uygulama Sunucusu Başlatma

* Ana Python dosyası (örneğin, app.py) çalıştırıldı:

python app.py

* Uygulamanın sorunsuz çalıştığı kontrol edilmiştir.

10.Web Sunucusunu Yapılandırma

* Uygulama, bir web sunucusuyla (Apache veya Nginx) entegre edilmiştir.
* Aşağıdaki örnek Nginx yapılandırması kullanılmıştır:

server {

listen 80;

server\_name yourdomain.com;

location / {

proxy\_pass http://127.0.0.1:5000;

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

}

}

Yapılandırma değişikliklerinden sonra sunucu yeniden başlatıldı:

* sudo systemctl restart nginx

ii)İstemci Tarafı Kurulumu

Web Tarayıcı Gereksinimleri

* Oyun, modern web tarayıcılarda (örneğin, Chrome, Firefox, Edge) çalışacak şekilde optimize edilmiştir.
* JavaScript destekli tarayıcılar gereklidir.

Görev Yönetim Sistemi URL'sine Erişim

* Kullanıcılar, oyuna aşağıdaki URL aracılığıyla erişebilir:

http://yourdomain.com

Sisteme Giriş Yapın

* Oyun, herhangi bir giriş sistemi gerektirmez.
* Ana sayfaya eriştir ve oyun hemen oynanabilir.

Görev Ekleme ve Yönetim İşlemleri

* Taş Kağıt Makas oyunu üzerinden birden fazla kullanıcının aynı anda oyun oynaması sağlanabilir.
* Kullanıcıların puanları sistemde tutulur ve liderlik tablosu güncellenir.

*iii) Ekstra Adımlar ve Sorun Giderme*

Veritabanı Yedekleme

* Oyun sırasında oluşan skor verileri SQLite veritabanında saklanır.

Firewall Ayarları

* Sunucunun 80 ve 443 portlarının açık olduğu kontrol edilmiştir:
* sudo ufw allow 80

sudo ufw allow 443

Sık Karşılaşılan Sorunlar ve Çözümler

* **Sorun:** Bağlantı reddedildi. **Çözüm:** Sunucunun çalıştığı ve gerekli portların açık olduğu kontrol edildi.
* **Sorun:** Sayfa yavaş yükleniyor. **Çözüm:** Uygulama ve web sunucusunun yüklenmesini optimize etmek için gereksiz dosyalar temizlendi.

**2. İnşa Aşamasında Gerçekleşen Değişiklikler ve İyileştirmeler**

* **Oyun Mekaniği:** Kullanıcı dostu bir arayüz tasarlandı.
* **Performans:** Oyun performansını artırmak için optimizasyon yapıldı.
* **Veritabanı:** Daha iyi veri analizi için oyun skorları kaydedildi.

1. Kullanıcı Arayüzü İyileştirmeleri

**a) Seçim Ekleme:**Kullanıcı arayüzüne, daha kullanıcı dostu bir deneyim sağlamak amacıyla seçim ekleme seçenekleri sunulur. Örneğin, görevler listelendiğinde, kullanıcılar her bir görev için ek seçenekler veya önceden tanımlanmış aksiyonları seçebilecekleri bir dropdown (açılır menü) kullanabilirler. Bu, daha hızlı ve kolay işlem yapılmasını sağlar.

**b) Seçim Listeleme:**Kullanıcılar, görevleri ve diğer öğeleri listelemek için daha sezgisel bir listeleme aracına sahip olurlar. Görevler filtrelenebilir ve arama yapılabilir, böylece kullanıcılar yalnızca ilgili verilere odaklanabilir.

**c) Karşılaştırma Sıfırlama:**Kullanıcılar birden fazla öğeyi karşılaştırdıysa, karşılaştırma filtresi sıfırlanabilir. Bu, kullanıcıların kolayca yeni görev veya verilerle tekrar çalışabilmelerini sağlar.

**d) Geçersiz Seçim:**Kullanıcı yanlış seçimler yaptığında, geçersiz seçimler için görsel bir uyarı (örneğin kırmızı renk veya hata simgesi) eklenir. Bu, hatalı veri girişi yapılmasının önüne geçer ve kullanıcıyı bilgilendirir.

**e) Sonuç Hesaplama:**Kullanıcı görevleri tamamladığında, işlemler üzerinden elde edilen sonuçlar hesaplanarak gösterilir. Örneğin, her bir görev için zaman verimliliği veya başarı oranı hesaplanabilir ve görsel olarak sunulabilir.

2. Görev Yönetim İşlevselliğindeki İyileştirmeler

**a) Seçim Ekleme:**Görevlerin durumu, önceliği veya etiketlenmesi için yeni seçim alanları eklenir. Bu seçimler, görevlerin daha iyi organize edilmesini sağlar. Örneğin, görevler "yüksek öncelikli", "orta öncelikli", "düşük öncelikli" gibi etiketlerle kategorize edilebilir.

**b) Seçim Listeleme:**Görevler, duruma göre sıralanabilir ve listelenebilir. Bu, kullanıcıların hangi görevlerin tamamlanması gerektiğini hızlıca görmelerini sağlar.Ayrıca filtreleme ve arama seçenekleriyle, kullanıcılar yalnızca belirli görevleri listeleyebilir.

**c) Karşılaştırma Sıfırlama:**Birden fazla görevi karşılaştırmak isteyen kullanıcılar için, filtreler ve seçimler sıfırlanabilir. Bu, kullanıcıların yeni görevlere kolayca geçiş yapabilmesini sağlar.

**d) Geçersiz Seçim:**Yanlış görev durumu veya etiket seçimi yapıldığında, kullanıcıya geçersiz seçim uyarısı verilir. Böylece görev yönetimi hatasız yapılır.

**e) Sonuç Hesaplama:**Görevlerin tamamlanma oranı, süresi veya diğer metrikler hesaplanır ve raporlanır. Bu, kullanıcıların performanslarını izlemelerini sağlar.

3. Veri Yönetimi ve Performans İyileştirmeleri

**a) Seçim Ekleme:**Veri yönetimi işlevselliğine optimizasyon eklemeleri yapılır. Bu, daha hızlı veri erişimi ve veri analizi yapmayı mümkün kılar. Örneğin, veritabanı sorgularına hız artırıcı seçimler eklenebilir.

**b) Seçim Listeleme:**Veritabanındaki veri öğeleri daha hızlı ve etkili bir şekilde listelenebilir. Ayrıca veritabanı yönetim sistemine erişimi optimize etmek için çeşitli veri filtreleme seçenekleri de eklenebilir.

**c) Karşılaştırma Sıfırlama:**Veri yönetiminde yapılan karşılaştırmaların sıfırlanması, veri analizi sırasında kullanıcıların yeni verilerle başlatmalarını sağlar. Bu, raporlama veya veri çıkartma işlemlerinde pratiklik sağlar.

**d) Geçersiz Seçim:**Geçersiz veri girişleri için hatalı seçimlerin kullanıcıya bildirilmesi sağlanır. Örneğin, bir veri girişinin türü veya formatı yanlış olduğunda, sistem kullanıcıyı uyararak hata yapmalarını engeller.

**e) Sonuç Hesaplama:**Veri analizi ve raporlama yapıldığında, verilerin istatistiksel hesaplamaları yapılır. Bu, kullanıcıların daha hızlı ve doğru kararlar almasına olanak tanır.

4. Güvenlik ve Erişim Kontrolü İyileştirmeleri

**a) Seçim Ekleme:**Gelişmiş kullanıcı yetkilendirme eklenir, böylece her kullanıcı sadece kendine ait olan verilere erişebilir. Farklı roller oluşturulabilir ve her role farklı erişim izinleri verilebilir.

**b) Seçim Listeleme:**Oturum yönetimi ve güvenlik seçenekleri kullanıcı arayüzünde listelenebilir. Hangi kullanıcıların hangi verilere erişebileceği, açıkça sıralanabilir.

**c) Karşılaştırma Sıfırlama:**Kullanıcıların erişim izni verilen verilerde yapılan değişiklikler, sıfırlanabilir. Bu, yetkisiz değişikliklerin önüne geçer.

**d) Geçersiz Seçim:**Erişim kontrolünde geçersiz seçimler yapılması durumunda, güvenlik uyarıları verilir. Örneğin, bir kullanıcı yetkisi olmayan bir alana erişmeye çalıştığında, uyarı mesajı gösterilir.

**e) Sonuç Hesaplama:**Güvenlik raporları ve kullanıcı erişim logları hesaplanarak, olası tehditler veya hatalar izlenebilir.

5. API Geliştirmeleri ve Entegrasyon Desteği

**a) Seçim Ekleme:**API entegrasyonları için kullanıcılar, yeni API uç noktalarını seçebilir. Örneğin, gerçek zamanlı bildirimler veya veri entegrasyonları gibi yeni API özellikleri eklenebilir.

**b) Seçim Listeleme:**Kullanıcılar, entegrasyon yapacakları API'leri listeleyebilir ve filtreleyebilir. Bu, hangi servislerin mevcut olduğunu görmek için faydalıdır.

**c) Karşılaştırma Sıfırlama:**API entegrasyonları arasındaki karşılaştırmalar sıfırlanabilir. Bu, farklı API’lerin özelliklerini test etmek isteyen kullanıcılar için yararlıdır.

**d) Geçersiz Seçim:**API seçimlerinde yanlış yapılandırma veya entegrasyon hataları için geçersiz seçimler kullanıcıya bildirilir. Bu, entegrasyon hatalarının erken tespit edilmesini sağlar.

**e) Sonuç Hesaplama:**API ile alınan verilerin işlenmesi ve sonuçlarının hesaplanması yapılır. API yanıt süreleri, başarı oranları ve diğer performans metrikleri ölçülüp gösterilebilir.

6. Kullanıcı Deneyimi (UX) İyileştirmeleri

**a) Seçim Ekleme:**Kullanıcıların kolayca anlayabileceği bir deneyim sağlamak amacıyla, mobil uyumlu arayüzler ve etkileşimli rehberler eklenebilir. Kullanıcıların arayüzde gezinmelerini daha kolay hale getiren seçimler eklenir.

**b) Seçim Listeleme:**Kullanıcılar, sık kullandıkları özellikleri veya seçenekleri kolayca listeleyebilir. Bu, kullanıcı deneyimini hızlandırır ve gereksiz karmaşıklığı azaltır.

**c) Karşılaştırma Sıfırlama:**Kullanıcılar arayüzde yaptıkları tüm değişiklikleri sıfırlayabilir ve sıfırlama sonrasında yeni seçimler yapabilirler.

**d) Geçersiz Seçim:**Kullanıcıların yanlış seçim yapması durumunda, kullanıcı deneyimini engellemeyecek şekilde, anlaşılır ve basit hata mesajları verilir.

**e) Sonuç Hesaplama:**Kullanıcı deneyimi (UX) iyileştirmeleri sonucu elde edilen başarı oranları, kullanım kolaylığı ve verimlilik gibi metrikler hesaplanır ve görsel olarak sunulur.Bu iyileştirmeler, kullanıcının oyun deneyimini daha verimli ve güvenli hale getirmek amacıyla Python ve yapay zeka kullanılarak geliştirilmiş bir sistemde uygulanabilir.

Taş Kağıt Makas Oyunu Kullanım Kılavuzu

**Sisteme Giriş Yapısı**

1. **Oyunun Başlatılması**
   * Oyunun çalışması için bilgisayarınızda **Python** ve gerekli kütüphaneler yüklü olmalıdır:
     + **OpenCV**: Görüntü işleme için kullanılır.
     + **Mediapipe**: El hareketlerini algılamak için kullanılır.
     + **Numpy**: Matematiksel işlemler için gereklidir.
   * Oyunu başlatmak için komut satırında aşağıdaki komutu çalıştırın: python tas\_kagit\_makas.py
2. **Kameranın Açılması**
   * Program, bağlı olan kamerayı otomatik olarak başlatır. Oyuncu elini kameranın önüne getirmelidir.
   * Sistem, el hareketlerini analiz ederek taş, kağıt veya makas seçimini yapar.
3. **Bilgisayarın Seçimi**
   * Bilgisayar, **random** modülü kullanarak rastgele bir seçim yapar (taş, kağıt veya makas).

**Oyun Yönetimi**

Oyunun akışı, el hareketlerinin algılanması, bilgisayarın rastgele seçim yapması ve sonuçların değerlendirilmesi şeklinde ilerler.

**a) El Hareketlerinin Algılanması**

1. **Mediapipe Kullanarak El Takibi**
   * Oyunda **Mediapipe** kütüphanesi kullanılarak elin konumu ve parmak hareketleri tespit edilir.
   * El hareketine göre sistem:
     + **Taş**: El kapalı, yumruk şeklinde.
     + **Kağıt**: El açık, tüm parmaklar düz.
     + **Makas**: İki parmak açık (işaret ve orta parmak).
2. **Algoritmanın Çalışması**
   * Kamera görüntüsü kare kare işlenir.
   * Her karede el pozisyonu analiz edilir ve taş, kağıt veya makas seçimi belirlenir.

**Örnek Kod Parçacığı**:

import cv2

import mediapipe as mp

mp\_hands = mp.solutions.hands

hands = mp\_hands.Hands()

cap = cv2.VideoCapture(0)

while True:

ret, frame = cap.read()

rgb\_frame = cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB)

result = hands.process(rgb\_frame)

if result.multi\_hand\_landmarks:

for hand\_landmarks in result.multi\_hand\_landmarks:

# El hareketlerini algıla ve işaretle

print(hand\_landmarks)

cv2.imshow("Camera", frame)

if cv2.waitKey(1) & 0xFF == ord('q'):

break

cap.release()

cv2.destroyAllWindows()

**b) Bilgisayarın Seçimi ve Karşılaştırma**

1. **Bilgisayarın Rastgele Seçimi**
   * Bilgisayar, Python’un **random.choice()** fonksiyonunu kullanarak taş, kağıt veya makas seçer:

import random

bilgisayar\_secimi = random.choice(['taş', 'kağıt', 'makas'])

1. **Sonuçların Karşılaştırılması**
   * Oyuncunun el hareketine göre seçilen hareket, bilgisayarın seçimi ile karşılaştırılır:
     + Taş > Makas
     + Kağıt > Taş
     + Makas > Kağıt
   * Eğer seçimler aynıysa sonuç **beraberlik** olarak ilan edilir.

**Örnek Karşılaştırma Kodu**:

if oyuncu\_secimi == bilgisayar\_secimi:

sonuc = "Beraberlik"

elif (oyuncu\_secimi == "taş" and bilgisayar\_secimi == "makas") or \

(oyuncu\_secimi == "kağıt" and bilgisayar\_secimi == "taş") or \

(oyuncu\_secimi == "makas" and bilgisayar\_secimi == "kağıt"):

sonuc = "Oyuncu Kazandı!"

else:

sonuc = "Bilgisayar Kazandı!"

**c) Skor Takibi ve Sonuçların Görüntülenmesi**

1. **Skor Yönetimi**
   * Her turda kazanan tarafın skoru artırılır. Skor ekranı görüntüde bir köşeye yerleştirilir.
2. **Sonuçların Anlık Görüntülenmesi**
   * Kamera ekranında oyuncu seçimi, bilgisayar seçimi ve skorlar gösterilir:
     + **Oyuncu Seçimi**: Ekranda oyuncunun algılanan hareketi yazılır.
     + **Bilgisayar Seçimi**: Rastgele seçilen hareket görüntülenir.

**d)Kullanıcı Arayüzü ve Hatalı Giriş Kontrolü**

1. **Kullanıcı Arayüzü**
   * Kamera görüntüsü üzerinde **OpenCV** kullanarak metinler ve grafikler eklenir.
   * Kullanıcı ve bilgisayar seçimleri görüntüye yansıtılır.
2. **Hatalı Algılamalar**
   * El hareketinin doğru algılanmadığı durumlar için sistem bir uyarı mesajı verir:
3. **Oyun Sonu**
   * Belirlenen skora ulaşıldığında oyun otomatik olarak sonlanır.