
Restoran Yönetim Sistemi

Proje Başlangıç Tarihi 6 Aralık 2023
Proje Bitiş Tarihi 30 Aralık 2023

1 Giriş

Bu projede, öğrencilerin işletim sistemlerinin temel kavramlarını (thread yönetimi ve senkronizasyon mekanizmalarını) anlamaları ve bu kavramları bir simülasyon üzerinde uygulamaları hedeflenmektedir.

Projede bir restoran yönetim sistemi gerçekleştirmeniz beklenmektedir. Bu proje bir restoranın günlük işleyişini simüle eden bir multithread uygulamasıdır. Buradaki amaç, gerçek zamanlı bir ortamda thread senkronizasyonu ve kaynak yönetimi konseptlerini uygulamaktır. Restoranda, **müşteri grupları, garsonlar, aşçılar ve bir kasa elemanı bulunmaktadır.** Her bir rol, farklı görevleri ve sorumlulukları olan ayrı bir thread olarak işlenecektir.

Uygulama arayüzden çalıştırılmalıdır.

1.1 Süreç

- Müşterilerin sıraya girmesi
- Garsonların sipariş alması
- Aşçıların siparişleri hazırlaması
- Müşterilerin siparişleri ödemesi

1.2 Threadler

- Her müşteri, restorana giriş yaparken bir thread olarak oluşturulur.

- Her garson, müşterilere hizmet vermek için ayrı bir thread'de çalışır.
- Her aşçı, yemek hazırlama sürecini yönetmek için kendi thread'inde çalışır.
- Kasa işlemleri, ödeme alma ve hesap kapatma (masanın uygun duruma gelmesi) işlemleri için ayrı bir thread'de gerçekleştirilir.

1.3 Senkronizasyon


- Müşteri thread'leri arasında masa seçimi ve oturma sırası için senkronizasyon gerekir.
- Garson thread'leri, siparişleri alırken ve işlerken diğer garsonlarla koordineli çalışmalıdır. Aynı masaya birden fazla garsonun bakması önlenmelidir.
- Aşçı thread'leri, sınırlı sayıda ocakta yemek hazırlarken birbirleriyle uyum içinde olmalıdır. Aynı siparişi birden fazla aşçının hazırlaması engellenmelidir.
- Kasa thread'i, her seferinde yalnızca bir siparişin ödemesini işleyebilir.


2 Projenin Soruları

- Problem 1: Restoranda belirli sayıda masa, garson, aşçı ve kasa bulunduğunda, bu kaynakların etkileşimini ve işleyişi nasıl simüle edilir?
- Problem 2: Dinamik bir müşteri akışında; hangi sayıda masa, garson ve aşçı çalıştırılırsa en fazla kazanç sağlanır?

2.1 Problem 1

- Uygulama çalıştırıldığında müşterilerin gelme sırası belirlenerek projenin özellikleri ve süreler dikkate alınarak simülasyon gerçekleştirilir.
- İşlem sırası geldiğinde kullanıcı onayıyla işlemler yürütülür.
- Her adımda kullanıcıdan bir sonraki aşama için onay beklenir.

 Simülasyon başlangıcında müşteri senaryosu belirlenir. Bu aşamada, zaman sınırlamalarına bağlı kalmaksızın müşterilerin öncelik düzeylerinin belirlenmesi hedeflenmektedir.

 Restoranda başlangıçta 6 masa, 3 garson, 2 aşçı ve 1 kasa bulunmaktadır.

 Aynı anda en fazla 10 müşteri gelebilir.

- Örnek senaryo:
 - (1.adım) 5 müşteri 0 öncelikli,
 - (2.adım) 4 müşteri 2 öncelikli,
 - (3.adım) 2 müşteri 1 öncelikli

- Örnek:

 Adım 1: 8 müşteri geldi. Öncelikli müşteri yok.

- ✦ Adım 2: 6 müşteri masalara yerleştirildi. 2 müşteri beklemede.
- Adım 3: Garson 1 Müşteri 1'in, garson 2 müşteri 2'nin ve garson 3 müşteri 3'ün siparişini aldı. Müşteri 4, müşteri 5 ve müşteri 6 sipariş alımı için beklemede.
- Adım 4: Garson 1 Müşteri 1, garson 2 müşteri 2 ve garson 3 müşteri 3'ün siparişini aşçıya iletti.
- Adım 5: Aşçı 1 müşteri 1 ve müşteri 2'nin siparişini aldı ve hazırlamaya başladı. Aşçı 2 müşteri 3'ün siparişini hazırlamaya başladı ve yeni sipariş için beklemede.
-

2.2 Problem 2

- Program başlatıldığında belirli bir süre tanımlanır. Bu süre içerisinde müşteri akışı hesaplanarak restoranın kaynakları belirlenerek maksimum müşteri sayısına hizmet verme stratejisi geliştirilir.
- Bu problem için simülasyon gerçekleştirmeniz beklenmemektedir.
- Müşteri sayısı iki şekilde belirlenebilmelidir:
 - Sabit akış modeli: Toplam süre, müşteri gelme aralığı uygulamada belirlenmelidir.
 - * Örnek: Her 5 saniyede 4 müşteri gelmektedir ve 1'i öncelikli olmaktadır. 3 dakikalık sürede gelecek müşteriye göre restoranın maksimum kapasitesi hesaplanır (masa, garson, aşçı).
 - Rastgele akış modeli: Belirlenen süre içerisinde müşterilerin gelişi rastgele aralıklarla ve sayıda olacak şekilde belirlenir.
 - * Örnek:
 - 1. adım 3 müşteri
 - 2. adım (4 saniye sonrası) 5 müşteri ve 1'i öncelikli
 - 3. adım (6 saniye sonrası) 7 müşteri ve 2'si öncelikli
- Belirlenen simülasyon adımı içerisinde maksimum verimle kaynakların çalışması beklenmektedir.
 - **Masa Maliyeti:** Her masa için maliyet 1 birim olarak kabul edilir.
 - **Garson Maliyeti:** Her garson için maliyet 1 birim olarak kabul edilir.
 - **Aşçı Maliyeti:** Her aşçı için maliyet 1 birim olarak kabul edilir.
 - **Müşteri Kazancı:** Her müşteri başına elde edilen kazanç 1 birim olarak hesaplanır.
- Bir problem için bulunan çözüm senaryolarını ve en iyi çözümü göstermeniz beklenmektedir.
- Örnek: 100 müşteri gelir. 10 müşteri ayrılır. 90 müşteri için 20 masa, 8 garson ve 2 aşçı için en iyi kazanç alınmaktadır. Kazanç $90-20-8-2 = 60$ 'tır.

3 Projenin Özellikleri

- Her masaya bir müşteri oturmaktadır.
- Masalar dolu olduğunda yeni gelen müşteri bir bekleme listesine alınır. Boş masa olduğunda müşteri sırasına göre uygun müşteri masaya yerleşir.
- Proje kapsamında müşteriler iki farklı kategoride değerlendirilebilir: Normal müşteriler ve öncelikli müşteriler (65 yaş ve üzeri). Öncelikli müşteriler restorana geldiklerinde, normal müşterilerin önüne geçmelidir. Bu durum masalara oturma sırasında öncelikli müşteri avantajı sağlamalıdır.
- Boş masa varken garsonlar bekleme durumundadır. Masa doldukça garsonlar sırayla müşterilerden sipariş alır. Aynı anda gelen olursa rastgele sipariş alımı yapabilir.
- Her garson aynı anda sadece bir müşterinin siparişini alabilir.
- Aşçılar sipariş gelene kadar bekler. Sipariş geldikten sonra yemek yapmaya başlarlar. Yemek hazır olduğunda sıradaki siparişleri hazırlamaya başlarlar.
- Sipariş alındığında, her bir aşçı aynı anda en fazla 2 yemek hazırlayabilir.
- Sipariş hazır olduktan sonra müşteriler yemeğe başlar. Müşteriler yemek yerken, kasa işlemleri için beklemelidir.
- Yemek biten masanın ödemesi kasa tarafından alınır.
- Kasa her seferinde sadece 1 siparişin ödemesini gerçekleştirebilir.
- Ödeme tamamlandıktan sonra ilgili masa uygun duruma gelir.

4 Süreler

- **Müşteri Bekleme Süresi:** Yeni müşteri restorana geldiğinde tüm masalar doluysa 20 saniye süren bir bekleme süresi simüle edilir. 20 saniye sonra müşteri restorandan ayrılmaktadır.
- **Sipariş Alma Süresi:** Garsonların müşteri gruplarından sipariş alması 2 saniye sürer.
- **Yemek Hazırlama Süresi:** Aşçılar sipariş aldıktan sonra yemek hazırlamaya başlarlar ve bu süreç 3 saniye sürer.
- **Yemek Yeme Süresi:** Müşterilerin yemek yeme süresi 3 saniye olarak ayarlanır.
- **Ödeme İşlemi Süresi:** Müşteriler yemeklerini bitirdikten sonra, ödeme işleminin gerçekleşmesi 1 saniye sürer.

5 Arayüz Gereklilikleri

Proje kullanıcı dostu ve etkileşimli bir arayüzle çalışacak şekilde tasarlanmalıdır.

5.1 Genel Arayüz Özellikleri

- Bir kullanıcı arayüzü (GUI) olmalıdır.
- Arayüz garson, aşçı ve kasa işlemlerini ayrı ayrı yönetebilmelidir.
- Kullanıcı dostu bir tasarım olmalıdır.

5.2 Garson Arayüzü

- Garsonlar için müşteri siparişlerini alma ve mutfak ekibine iletmek için bir arayüz olmalıdır.
- Garsonlar, müşterilerin sipariş durumlarını ve masaların mevcut durumunu görebilmelidir.

5.3 Aşçı Arayüzü

- Aşçılar, alınan siparişleri görüntüleyebilmeli ve yemek hazırlama sürecini yönetebilmelidir.
- Hazırlanan yemeklerin durumu güncellenebilmeli ve garsonlara bildirebilmelidir.

5.4 Kasa Arayüzü

- Kasa işlemleri için ödeme alımı bulunmalıdır.
- Ödemelerin durumu görüntülenebilir.
- Toplam müşteri sayısı ve ödeme miktarı bulunur.

6 Önemli

- Tüm adımlar anlık olarak bir metin dosyasına yazdırılmalıdır.
- Uygulamada statik olarak verilen değerlerin (süreler ve sayılar) sunum sırasında güncellenmesi istenebilir.

7 Proje Teslimi

Rapor (IEEE formatında times new roman 12 pt) özet, yöntem (detaylı), katkılar ve kaynakça bölümünden oluşmalıdır.

Raporda her bir grup üyesinin katkıları belirtilmelidir.

Raporda akış diyagramı bulunmalıdır.

Proje ile ilgili sorular edestek2.kocaeli.edu.tr sitesindeki forum üzerinden Arş. Gör. Ahmet Demirtaş veya Arş. Gör. Dr. Furkan Göz'e sorulabilir.

Demo tarihleri daha sonra duyurulacaktır.

Demo sırasında algoritma, geliřtirdiđiniz kodun eřitli kısımlarının ne amaçla yazıldıđı ve geliřtirme ortamı hakkında sorular sorulabilir.

Kullandıđınız herhangi bir satır kodu açıklamanız istenebilir.

Proje grupları her đretim kendi iinde olmak zere en fazla iki kiřiden oluřturulmalıdır.