

## Sistem Programlama Ödevi

- **Java** programlama diliyle **gRPC** gönderim fonksiyonlarını kullanarak dağıtık, hata-tolere bir abonelik sistemi geliştirmeniz beklenmektedir.
- Bu abonelik sistemi socket üzerinde SMTP, HTTP vb. bir protokol yerine ödev kapsamında ortaya atılan HaToKuSe (Hata-Tolere Kuyruk Servisi) isminde sınıf tabanlı ilkel bir protokol ile gerçekleşmelidir.
- Yük dağılımı, hata tolerans vb. taleplerde başvurulacak dağıtık mimaride her bir sunucu, abone bilgilerini tutmaktadır. İstemci, sadece lider sunucudan metin tabanlı mesajlaşma sistemi ile dağıtık olarak saklanacak mesajı tutmaktadır.
- Ödev güncellemeleri, bu doküman üzerinden güncellenecektir.
- Ödevi [github](#)’ da bulunan şablon üzerinden “fork” yaparak “Github Projects” altında iletiniz. Ödev boyunca takım olarak yaptığınız geliştirmeleri repo-nuza “merge” işlemi ile güncelleyiniz.
- Ödevi iletme süreciniz projenizde oluşturduğunuz “mini iş (task)” lar üzerinden incelenecektir.
- İstemci text tabanlı olarak 2 tür istek yapabilmelidir.

SET <message\_id> <message>

GET <message\_id>

- İstemciden mesajı alan lider, gelen mesajı hata tolerans sayısı kadar **üyenin diske kayıt işlemi gerçekleştirildikten/gerçekleşmedikten sonra** dönüt mesajını OK/ERROR bildirmelidir.
- Lider, periyodik olarak sistemde toplam kaç mesaj saklandığını, hangi üyelerin kaç adet mesaj tuttuğunu bastırmalıdır.
  - Lider sunucu, tolerance.conf dosyasından okuduğu hata tolerans değerine göre gelen mesajı bir o kadar aile üyesine de göndermelidir.
  - Mesajı alan aile üyeleri, bu mesajı diskte saklamalıdır.
  - Lider sunucu, istemciden (client) aldığı mesaj metin (text) tabanlı iken aile arası haberleşme .protobuf nesnesi ile olmalıdır.
  - Aile üye sistemi, **dinamik** olmalıdır. n sayıda üye, aileye girip, çıkabilmelidir.

- Üyeler periyodik olarak kendi disklerinde kaç mesaj sakladıklarını bastırmalıdır.

Test tabanlı yaklaşımla ödevi anlamak\_1:

- tolerance.conf dosyasında hata toleransı 2 seçilmiş olsun.
- 5 farklı terminalden üye başlatılmış olsun. (1’ i lider, 4’ ü üye olacaktır.)
- Lidere gönderilecek n adet mesaj, üyelere dengeli dağıtılıyor mu? (Örneğin 1000 SET mesajı gönderimi sonunda, 500 mesaj bir çiftte, diğer 500 mesaj öteki çiftte dengeli kaydedilmiş mi?)
- Lider her mesajı, id değerine göre diskinde kaydetmiş üye listesini tutuyor mu?
- İstemci 500 id değerli mesajı istediğinde, bu mesaj 3 ve 4 numaralı üyede tutuluyorsa, bu üyelerden birisi (örn 3.üye) de crash olmuşsa, lider 500 id değerli mesajı 4 numaralı üyeden alıp, istemciye verebiliyor mu?

Test tabanlı yaklaşımla ödevi anlamak\_2:

- tolerance.conf dosyasında hata toleransı 3 seçilmiş olsun.
- 7 farklı terminalden üye başlatılmış olsun. (1’ i lider, 6’ sı üye olacaktır.)
- Lidere gönderilecek n adet mesaj, üyelere dengeli dağıtılıyor mu? (Örneğin 9000 SET mesajı gönderimi sonunda, 4500 mesaj bir üçlü üyede, diğer 4500 mesaj öteki üçlü üyede dengeli kaydedilmiş mi?)
- Lider her mesajı, id değerine göre diskinde kaydetmiş üye listesini tutuyor mu?
- İstemci 4501 id değerli mesajı istediğinde, bu mesaj 3,5 ve 6 numaralı üyelere tutuluyorsa, bu üyelere biri veya ikisi de crash olmuşsa, lider 4501 id değerli mesajı, o mesajı tutan ve hayatta kalan son üyeden alıp, istemciye verebiliyor mu?

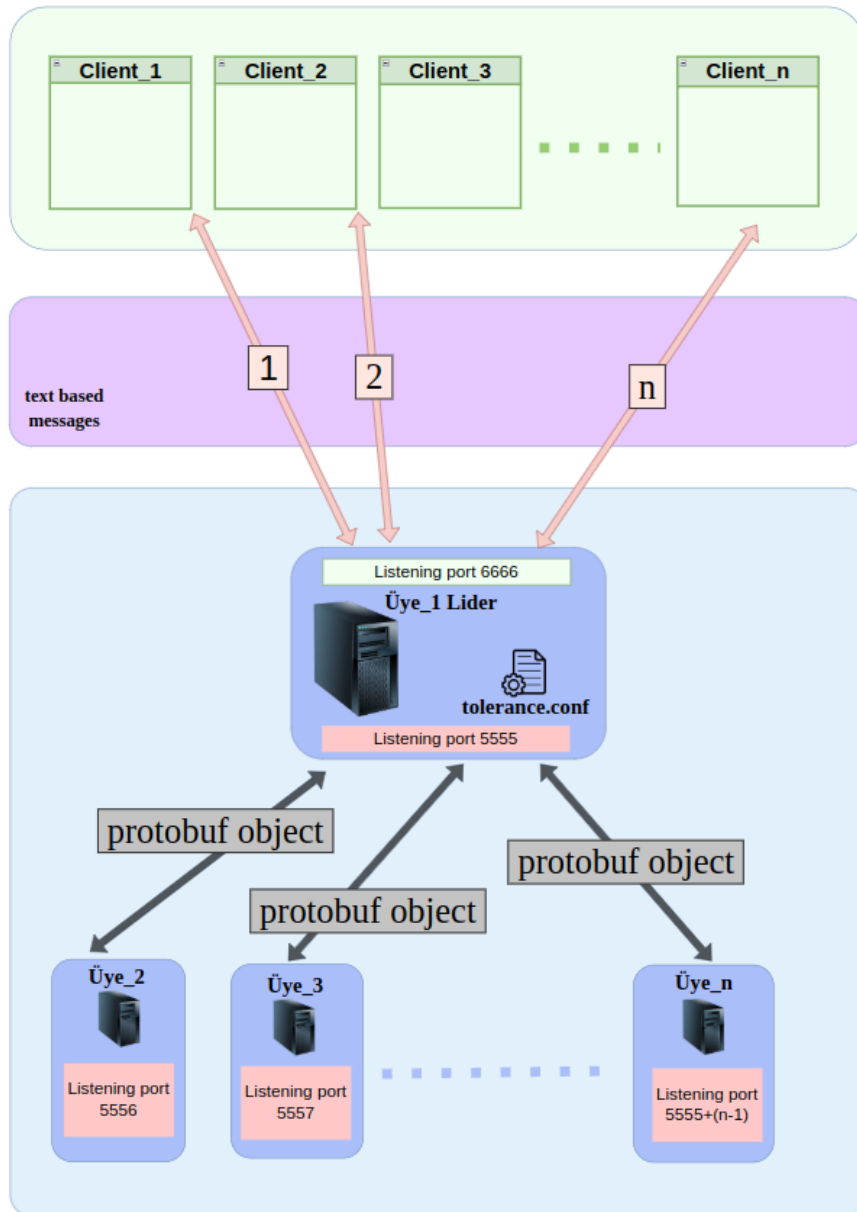
Aileye eklenen yeni üye, geçmişe dönük yük dağılımı yapmasına gerek yoktur. Zamanla gelen mesajları kaydetmede ön plana çıkarak sistemi dengeleyebilir.

## Ödev Puanlaması

İş Parçacığı	Puan
Hata toleransı 1 ve 2 olarak çalışan dağıtık kayıt sistemi (*)	50
Hata toleransı n olan dağıtık kayıt sistemi (maksimum 7 ile test edilecektir) (*)	30
Diske farklı kayıt etme biçimleri (zero copy ilkesi, buffered/unbuffered IO)	20
Arka arkaya istek yapan client paylaşılacaktır.	-
README.md dosyasında yapılanların maddelenmesi, açıklayıcı video veya sunum yapılmaması halinde caydırıcı puan kesilecektir.	<b>ZORUNLU</b>

(\*)back\_up işlemi için kalan n adet sunucudan hangisinin seçileceğine karar verilmesinin özgünlüğüne göre puanı artabilir.

## Sistem Mimarisi



Şekil.2 Dağıtık Mesaj Kayıt Servisi Sistem Mimarisi

## Örnek İş Parçacıkları

Tavsiye niteliğindedir. Link için [tıklayınız](#).

(\*) Bu ödev, [RabbitMQ](#) ve [Apache Kafka](#) çözümlerinden ilham alınarak hazırlanmıştır. Ülkemiz mühendislerinin de tüm dünyada kullanılacak yüksek teknoloji kütüphaneleri geliştirmeleri veya geliştiren takımlara katılmaları dileğiyle...

(Dersin ilgi alanında olup, ülkemizden tüm dünyaya yayılmış ve son kullanıcısı yine mühendisler olan yüksek teknoloji örnekleri [1](#), [2](#) )