

GENEL TANIM / GENERAL DESCRIPTION

Ders Adı / Course Name	DISTRIBUTED APPLICATION DEVELOPMENT / DISTRIBUTED APPLICATION DEVELOPMENT	
Ders Kodu / Course Code	501008122025	
Ders Türü / Course Type		
Ders Seviyesi / Course Level	First Cycle / First Cycle	
Ders Akts Kredi / ECTS	5.00	
Haftalık Ders Saati (Kuramsal) / Course Hours For Week (Theoretical)	3.00	
Haftalık Uygulama Saati / Course Hours For Week (Objected)	0.00	
Haftalık Laboratuvar Saati / Course Hours For Week (Laboratory)	0.00	
Dersin Verildiği Yıl / Year	4	
Öğretim Sistemi / Teaching System	Face to Face / Face to Face	
Eğitim Dili / Education Language	English / English	
Ön Koşulu Olan Ders(ler) / Precondition Courses	Yok	None
Amacı / Purpose	<p>Öğrencilere ölçeklenebilir, hataya dayanıklı ve yüksek performanslı dağıtık ve eş zamanlı sistemlerin tasarlanması ve geliştirilebilmesi için gerekli teorik temellerin ve pratik becerilerin kazandırılması amaçlanmaktadır.</p> <p>Ayrıca öğrencilerin; Aktör Modeli (Akka), Olay Odaklı Mimari (Kafka, RxJava) ve İletişim Kuran Sıralı İşlemler (Go'da CSP) olmak üzere üç önemli paradigmayı inceleyerek modern eş zamanlı ve dağıtık mimariler hakkında kapsamlı bir bilgi birikimi kazanması hedeflenmektedir.</p>	<p>The primary goal of this course is to equip students with theoretical foundations and practical skills to design, implement, and optimize scalable, fault-tolerant, and high-performance distributed and concurrent systems.</p> <p>By exploring three major paradigms as the Actor Model (Akka), Event-Driven Architectures (Kafka, RxJava), and Communicating Sequential Processes (CSP in Go), students will gain a comprehensive understanding of modern distributed and concurrent architectures.</p>
İçeriği / Content	<p>1. Aktör Modeli ve Mesaj Odaklı Eşzamanlılık</p> <ul style="list-style-type: none"> •Eşzamanlılık ve Dağıtık Sistemlere Giriş •Aktör Modeli İlkeleri •Akka ile Aktörlerin Uygulanması •Dağıtık Aktör Tabanlı Sistemlerin Oluşturması <p>2. Olay Odaklı Mimari ve Reaktif Programlama</p> <ul style="list-style-type: none"> •Olay Odaklı Mimariye Giriş •RxJava ile Reaktif Programlama •Kafka ve Olay Odaklı Mikroservisler •Gerçek Zamanlı Akış İşleme <p>3. İletişim Kuran Sıralı İşlemler (CSP) ve Go Eşzamanlılığı</p> <ul style="list-style-type: none"> •CSP vs Aktör Modeli •Go'da Eşzamanlılık •Dağıtık Go Uygulamaları •Dağıtık Eşzamanlılıkta Gelecek Trendleri 	<p>1. Actor Model and Message-Driven Concurrency</p> <ul style="list-style-type: none"> •Introduction to Concurrency and Distributed Systems •Actor Model Principles •Implementing Actors with Akka •Building Distributed Actor-Based Systems <p>2. Event-Driven Architectures and Reactive Programming</p> <ul style="list-style-type: none"> •Introduction to Event-Driven Architectures •Reactive Programming with RxJava •Kafka and Event-Driven Microservices •Real-Time Stream Processing <p>3. Communicating Sequential Processes (CSP) & Go Concurrency</p> <ul style="list-style-type: none"> •CSP vs. Actor Model •Concurrency in Go •Distributed Go Applications •Future Trends in Distributed Concurrency

Önerilen Diğer Hususlar / Recommended Other Considerations	Yok	None
Staj Durumu / Internship Status	Yok	None
Kitap / Malzemesi / Önerilen Kaynaklar / Books / Materials / Recommended Reading	<p>1.Sathyanarayanan, Prasanna Kumar, and Suraj Atreya. Reactive programming with Scala and Akka. Packt Publishing Ltd, 2016.</p> <p>2.Nurkiewicz, Tomasz, and Ben Christensen. Reactive programming with RxJava: creating asynchronous, event-based applications. " O'Reilly Media, Inc.", 2016.</p> <p>3.Narkhede, Neha, Gwen Shapira, and Todd Palino. Kafka: the definitive guide: real-time data and stream processing at scale. " O'Reilly Media, Inc.", 2017.</p> <p>4.Cox-Buday, Katherine. Concurrency in Go: Tools and Techniques for Developers. " O'Reilly Media, Inc.", 2017.</p>	<p>1.Sathyanarayanan, Prasanna Kumar, and Suraj Atreya. Reactive programming with Scala and Akka. Packt Publishing Ltd, 2016.</p> <p>2.Nurkiewicz, Tomasz, and Ben Christensen. Reactive programming with RxJava: creating asynchronous, event-based applications. " O'Reilly Media, Inc.", 2016.</p> <p>3.Narkhede, Neha, Gwen Shapira, and Todd Palino. Kafka: the definitive guide: real-time data and stream processing at scale. " O'Reilly Media, Inc.", 2017.</p> <p>4.Cox-Buday, Katherine. Concurrency in Go: Tools and Techniques for Developers. " O'Reilly Media, Inc.", 2017.</p>
Öğretim Üyesi (Üyeleri) / Faculty Member (Members)	Öğr. Gör. Dr. Burak YÖNYÜL	Teaching Assistant Dr. Burak YÖNYÜL

ÖĞRENME ÇIKTILARI / LEARNING OUTCOMES

1	Eş zamanlı ve dağıtık hesaplama modellerinin anlaşılması ve kıyaslanması	Understand and compare concurrency and distributed computing models
2	Ölçeklenebilir ve hataya-duyarlı dağıtık sistemlerin tasarlanması ve gerçekleştirilmesi	Design and implement scalable and fault-tolerant distributed systems
3	Endüstriyel seviyede eş zamanlı ve olay-odaklı desenlerin dağıtık uygulamalarda kullanılması ve optimizasyonu	Applying and optimizing industry-standard concurrency and event-driven patterns in distributed applications
4	Geniş ölçekli hesaplamalar için pratik ve endüstri-hazır çözümler geliştirebilme	Develop practical, industry-ready solutions for large-scale computing

HAFTALIK DERS İÇERİĞİ / DETAILED COURSE OUTLINE

Hafta / Week					
1	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Eşzamanlı ve dağıtık programlamaya giriş			Kaynak-1	
	Introduction to Distributed Application Development			Reading-1	
2	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Aktör model temelleri			Kaynak-1	
	Fundamentals of the Actor Model			Reading-1	
3	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Aktörlerin Akka ile gerçekleştirimi			Kaynak-1	
	Implementing Actors in Akka			Reading-1	
4	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Aktör tabanlı dağıtık uygulamaların ölçeklenmesi			Kaynak-1	
	Scaling Actor-Based Distributed Applications			Reading-1	
5	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Aktör tabanlı olay akışı ve kalıcılık			Kaynak-1	
	Actor-Based Event Streaming and Persistence			Reading-1	

	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
6	Olay tabanlı mimarilere giriş			Kaynak-2	
	Introduction to Event-Driven Architectures			Reading-2	
7	RxJava ile reaktif programlama			Kaynak-2	
	RxJava ile reaktif programlama			Kaynak-2	
8	Ara Sınav				
	Midterm Exam				
9	Kafka'ya ve dağıtık mesajlaşmaya giriş			Kaynak-3	
	Introduction to Kafka & Distributed Messaging			Reading-3	
10	Gelişmiş olay tabanlı desenler ve kullanım durumları			Kaynak-3	
	Advanced Event-Driven Patterns and Use Cases			Reading-3	
11	CSP'yi ve Go'nun eşzamanlılık modelini anlama			Kaynak-4	
	Understanding CSP and Go's Concurrency Model			Reading-4	

	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
12	Go'da Goroutine'ler ve kanallar			Kaynak-4	
	Goroutines and Channels in Go			Reading-4	
13	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Go ile dağıtık uygulamaların geliştirimi			Kaynak-4	
	Building Distributed Applications with Go			Reading-4	
14	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Go ile dağıtık mesajlaşmanın entegrasyonu			Kaynak-4	
	Integrating Go with Distributed Messaging			Reading-4	
15	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Dağıtık ve eşzamanlı sistemlerde gelecek trendleri			Sunum ve tartışma	
	Future Trends in Distributed and Concurrent Applications			Presentation and Discussion	
16	Teorik Dersler / Theoretical	Uygulama	Lab	Öğretim Yöntem ve Teknikleri/Teaching Methods Techniques	Ön Hazırlık / Preliminary
	Final Sınavı				
	Final Exam				

DEĞERLENDİRME / EVALUATION

Yarıyıl (Yıl) İçi Etkinlikleri / Term (or Year) Learning Activities	Sayı / Number	Katkı Yüzdesi / Percentage of Contribution (%)
Ara Sınav / Midterm Examination	1	50
Proje Hazırlama / Project Preparation	1	50
Toplam / Total:	2	100
Başarı Notuna Katkı Yüzdesi / Contribution to Success Grade(%):		60

Yarıyıl (Yıl) Sonu Etkinlikleri / End Of Term (or Year) Learning Activities	Sayı / Number	Katkı Yüzdesi / Percentage of Contribution (%)
Final Sınavı / Final Examination	1	100
Toplam / Total:	1	100
Başarı Notuna Katkı Yüzdesi / Contribution to Success Grade(%):		40

Etkinliklerinin Başarı Notuna Katkı Yüzdesi(%) Toplamı / Total Percentage of Contribution (%) to Success Grade:	100
Değerlendirme Tipi / Evaluation Type:	DDS

İŞ YÜKÜ / WORKLOADS

Etkinlikler / Workloads	Sayı / Number	Süresi (Saat) / Duration (Hours)	Toplam İş Yüğü (Saat) / Total Work Load (Hour)
Ara Sınav / Midterm Examination	1	20.00	20.00
Derse Katılım / Attending Lectures	14	3.00	42.00
Final Sınavı / Final Examination	1	20.00	20.00
Okuma / Reading	14	2.00	28.00
Proje Hazırlama / Project Preparation	1	40.00	40.00
Toplam / Total:	31	85.00	150.00

Dersin AKTS Kredisi = Toplam İş Yüğü (Saat) / 30.00 (Saat/AKTS) = 150.00/30.00 = 5.00 ~ 5.00 / Course ECTS Credit = Total Workload (Hour) / 30.00 (Hour / ECTS) = 150.00 / 30.00 = 5.00 ~ 5.00

PROGRAM VE ÖĞRENME ÇIKTISI / PROGRAM LEARNING OUTCOMES

Öğrenme Çıktıları / Learning Outcomes	Program Çıktıları / Program Outcomes										
	1.1.1	1.1.2	1.1.3	1.1.4	1.1.5	1.1.6	1.1.7	1.1.8	1.1.9	1.1.10	1.1.11
1.Eş zamanlı ve dağıtık hesaplama modellerinin anlaşılması ve kıyaslanması / Understand and compare concurrency and distributed computing models	5			5	4						
2.Ölçeklenebilir ve hataya-duyarlı dağıtık sistemlerin tasarlanması ve gerçekleştirilmesi / Design and implement scalable and fault-tolerant distributed systems	5		5	3		5	3			3	
3.Endüstriyel seviyede eş zamanlı ve olay-odaklı desenlerin dağıtık uygulamalarda kullanılması ve optimizasyonu / Applying and optimizing industry-standard concurrency and event-driven patterns in distributed applications	5	4	3	3	3	5	3	3			
4.Geniş ölçekli hesaplamalar için pratik ve endüstri-hazır çözümler geliştirebilme / Develop practical, industry-ready solutions for large-scale computing	5	3	4	4	3	5	3	3		3	

Katkı Düzeyi / Contribution Level : 1-Çok Düşük / Very low, 2-Düşük / Low, 3-Orta / Moderate, 4-Yüksek / High, 5-Çok Yüksek / Very high