



Tasarım Şablonları Ders İzlenesi

Dersin Amacı: Yazılım uygulamalarında kullanılan şablonların nesneye yönelik analiz ve tasarım ilkeleri ve yazılım mimarisi ile ilişkilendirerek tanımlanması, takım çalışması ile nesne yönelimli yazılım geliştirme ilkelerinin bir gerçek dünya uygulamasında gösterilmesi.

Derisin İçeriği: Bu derste yazılım tasarım şablonlarının kullanımı tanıtılmaktadır. Nesneye yönelik tasarım ilkeleri, temiz kod prensipleri, yaratıcı, yapısal ve davranışsal tasarım şablonları, mimari desenler, anti-desenler incelenmektedir. Öğrencilerin oluşturdukları bir gerçek dünya problemi uygulamasının çözümü şablonlar kullanılarak gerçekleştirilecektir.

Dersin Kaynakları: 1. Eric Freeman, Elisabeth Freeman, Kathy Sierra, Bert Bates

Head First Design Patterns, 2004, O'Reilly

2. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides Design Patterns: Elements of Reusable Object-oriented Software Addison-Wesley, 1995

3. Metseker and Wake “Design Patterns in Java”, Addison Wesley, 2006.

4. Level up code with Game Programming Patterns (e-book), Unity,2012

5. Robert C.Martin, CleanCode. A handbook of Agile Software Craftsmanship, 2009.

6. Martin Fowler, Refactoring Improving the Design of Existing Code, 2019.

Haftalık Ders Konuları

Dersin kapsamı içerisinde davetli konuşmacılar olacaktır. Çevrimiçi gerçekleşecek olan seminerler konusunda uzman mezun öğrencilerimiz tarafından verilecektir.

1	1.1Tasarım şablonu nedir? 1.1.1 Tasarım şablonlarının sınıflandırması 1.2 Nesneye yönelik tasarımda Solid İlkeleri 1.3 Refactoring (Ö01)	https://www.youtube.com/watch?v=vqEg37e4Mkw Workflows Refoctaring Martin Fowler https://www.youtube.com/watch?v=mDSSuOrarcs&t=4s Clean Code - Book Review
2	2.1 Tasarım Şablonları tasarım problemlerini nasıl çözer? 2.2 Temiz kod ve refactoring (devam) (Ö01)	https://www.youtube.com/watch?v=G08FxxwPjXE&t=3173s Clean Architecture Robert Martin https://refactoring.guru/refactoring
3	Yaratıcı şablonlar (Ö02)	https://www.youtube.com/watch?v=7EmboKQH8IM&list=P_LmmYSbUCWJ4x1GO839azG_BBw8rkh-zOj CleanCode Principles Robert Martin

4	Yaratıcı Şablonlar (devam) ve yaratıcı şablonların karşılaştırması (Ö02,Ö03) Grup çalışması projesinin önerilerinin verilmesi (Ö02, Ö03)	Araştırma Ödevi https://www.youtube.com/watch?v=6wDoopbtEqk&t=1167 Martin Fowler - Software Design in the 21st Century
5	Yapısal Şablonlar (Ö02)	
6	Yapısal Şablonlar (devam) ve Yapısal Şablonların Karşılaştırması (Ö02,Ö03) Grup çalışması projesinin ilk raporunun teslimi (Ö03)	Araştırma Ödevi
7	Teslim edilen grup projelerinin diğer gruplar tarafından değerlendirilmesi ve tartışmalar (Ö03)	
8	Ara Sınav	
9	Davranışsal Şablonlar (Ö02)	
10	Davranışsal Şablonlar (devam) ve Davranışsal Şablonların Karşılaştırması (Ö02,Ö03)	Araştırma Ödevi
11	Karmaşık Öğrenmede Tasarım Şablonları (Ö03)	https://auspace.athabascau.ca/bitstream/handle/2149/742/design_patterns.pdf?sequence=1 Design Patterns for Complex Learning
12	Yapay Zeka ile Tasarım Şablonları (Ö03)	https://arxiv.org/pdf/2303.13173 Design Patterns for AI-based Systems: A Multivocal Literature Review and Pattern Repository
13	Grup çalışması projesinin ikinci raporunun teslimi ve yıl içi çalışmalarının sunumları (Ö04)	
14	Yıl içi çalışmalarının sunumları (devam) (Ö04)	
15	Final Sınavı	

Dersin Öğrenme Çıktıları

Ö01	Nesneye yönelik temel tasarım ilkelerini yorumlar, nesne yönelimli sistemlerin tasarımında ifade eder ve örnekler
Ö02	Tasarım şablonlarına ait farklı örnekleri sınıflandırır, karşılaştırır ve anladığını ifade eder.
Ö03	Temel tasarım ilkelerini deneyimler ve bu ilkeler bağlamında bir tasarımın kalitesini değerlendirir.
Ö04	Geliştirdiği uygulamaya uygun şablonları seçer, uygular ve yazılım ürününü geliştirir.

Öğrenme Çıktısı Bloom Taksonomisi Fiilleri:

2.Anlama (Understand) Seviyesi: “Öğrenciler teorik bilgileri gerçek yaşam durumlarıyla ilişkilendirir.”

Bloom taksonomisi fiili “sınıflandırmak”, “karşılaştırmak”, “ifade etmek”, ”yorumlamak”, örneklemek”

3.Uygulama (Apply) Seviyesi: “Öğrenciler bir problem durumunda uygun çözüm yöntemini uygular.”

Bloom taksonomisi fiili “seçmek”, “uygulamak”

5.Değerlendirme (Evaluate) Seviyesi: “Öğrenciler, önerilen stratejilerin etkinliğini değerlendirir.”

Bloom taksonomisi fiili “deney yapmak”, “değerlendirmek”

6. Oluşturma Seviyesi: (Create) “Öğrenciler, yeni bir ürün veya çözüm geliştirir.”

Bloom taksonomisi fiili “geliştirmek”

Değerlendirme Ölçütleri

%25	Ara Sınav
%15	Ödev (ara sınav öncesi yapılan çalışmaların değerlendirmesi)
%20	Proje
%40	Final

AKTS Hesaplama İçeriği	Sayısı	Süresi	İşyükü Saati
Ders Süresi	13	3	39
Sınıf Dışı Ç. Süresi	13	3	39
Ödevler	1	10	12
Sunum/Seminer Hazırlama	0	0	0
Ara Sınavlar	1	15	10
Uygulama	0	0	0
Laboratuvar	0	0	0
Proje	1	0	15
Yarıyıl Sonu Sınav	1	15	10
Toplam İş Yükü			125
AKTS Kredisi			5

Yazılım Mühendisliği Programının Öğrenme Çıktıları

Sıra No	Açıklama
1	Matematik, fen bilimleri ve teknoloji ile ilgili yazılım mühendisliği konularını hatırlar ve eşleştirir .
2	Yazılım Mühendisliği alanındaki temel kuramları ve uygulamaları tanımlar ve ilişkilendirir .
3	Karmaşık yazılım mühendisliği problemlerini açık bir şekilde ifade eder ve uygun çözümleme süreçlerini belirler .
4	Karmaşık bir sistemin, sürecin veya ürünün belirli koşullardaki gereksinimlerini karşılamak üzere modern tasarım yöntemlerini açıklar , tahmin eder ve değerlendirir .
5	Yazılım mühendisliği uygulamalarında karşılaşılan karmaşık problemlerin analizi ve çözümü için gerekli teknikleri ve geliştirme araçlarını inceler , uygular ve entegre eder .
6	Karmaşık yazılım mühendisliği problemlerini incelemek üzere görev aldığı alanlardaki sorumluluğuyla takım çalışması içerisinde kendi görüşünü ifade eder , yorum yapar ve iş birliği yapar .
7	Yazılım projesindeki rolü ile ilgili bilgi ve becerileri tartışır , gözden geçirir ve yönetir .
8	Yaşam boyu öğrenme stratejisi doğrultusunda erişebildiği bilgiyi değerlendirir ve teknolojiadaki gelişmeleri izleyerek sürekli kendini geliştirir .
9	Yazılım mühendisliği alanında bilişim ve iletişim teknolojilerini etkin bir şekilde uygular , bu teknolojileri yazılım projelerine uyarlar ve ihtiyaçlara göre değiştirir .
10	Yazılım mühendisliği alanındaki paydaşlarla yazılı ve sözlü olarak görüşmeler yapar , karar ve düşüncelerini sunar ve tartışır .
11	Yazılım alanındaki gelişmeleri takip edebilmek ve meslektaşlarıyla iletişim kurabilmek için en az bir yabancı dili B1 seviyesinde ifade eder ve yorum yapar ; görüşmeler yapar ve sunar .
12	Mesleğine ilişkin etik ilkeleri toplar , iş etiği ile ilgili sorunları tespit eder ve kendi çözüm önerilerini ifade eder .
13	Kalite, proje, risk ve değişim yönetim sistemleri ile ilgili yorum yapar , sistemleri gözden geçirir ; sürekli iyileştirme süreçleri ile ilgili öneride bulunur ve inşa eder .
14	Mühendislik uygulamalarının toplumsal boyuttaki çevre bilinci, iş sağlığı ve güvenliği konularının çözümlerini uzun vadeli olarak tahmin eder , hipotezler oluşturur ve ilgili hedefe yönelik önerilerde bulunur .

Dersin Öğrenme Çıktılarının Programın Öğrenme Çıktılarına Katkıları
Katkı Düzeyi: 1: Çok Düşük 2: Düşük 3: Orta 4: Yüksek 5: Çok yüksek

	P01	P02	P03	P04	P05	P06	P07	P08	P09	P10	P11	P12	P13	P14	Total
Ort.		2,5	4	5	5	5	12	4			3	1,5	1,5		
Ö01		5	5	5											15
Ö02			3	5	5	5	3								21
Ö03					5	5	5	4	4		3				26
Ö04					5	5	5	4	4		3	3	3		32
Total		5	8	10	15	15	13	8	8		6	3	3		94