



Yazılım Mühendisliğine Giriş

- Yazılım mühendisliğine giriş,
- Yazılım mühendisliği ve etik,
- Yazılım mühendisliğinin önemi ve gereği,
- Yazılım geliştirme süreci
- Yazılım sürecinde araştırma, ölçme, planlama ve gereksinim analiz yöntemleri,
- Yazılım mühendisliğinde metodolojileri,
- Yazılım yaşam döngüsü sürecinde metotlar,
- Yazılımda standartlar, kalite teknikleri ve kalite prensipleri,
- Yazılımda proje yönetimi,
- Yazılım projelerinde başarı ve başarısızlık nedenleri,
- Bilgisayar destekli yazılım araçları ve örnek uygulamalar.

- Mesleki etik kuralların öğrenilmesi
- Gelişen teknolojiler ışığında, yazılım mühendisinde bulunması gereken niteliklerin belirlenmesi.
- Yazılım mühendisliği sürecini anlama, uygulama ve proje sürecini yönetebilme becerisi kazandırma.
- Geçerli yazılım mühendisliği standartlarını araştırma, öğrenme ve uygulayabilme.
- Güncel yazılımlar hakkında bilgiler vermek, uygulamaya yönelik çözümler üzerinde durmak.
- Hızla gelişen yazılım ihtiyaçlarına, kabul gören standartlarda güvenli kod geliştirebilme kabiliyeti kazandırma.

Amaç :

Yazılım mühendisliği sürecindeki

- ✓ **isterler,**
- ✓ **analiz,**
- ✓ **tasarım,**
- ✓ **gerçekleme,**
- ✓ **test,**
- ✓ **bakım,**

gibi kavramları öğrenmek, kodlama ilkelerini gözden geçirmek, günümüzün en popüler yazılım projelerinin örneklerini paylaşmak, modüler yapıların incelenmesi, kişisel gelişim ve teknolojik değişime ayak uydurmak.

Ders İçeriği ;

1. (1 .Hft) Yazılım nedir, Yazılım mühendisliği ve eğitimi,
2. (2 .Hft) Yazılım İsteklerin belirlenmesi ve geliştirme süreci,
3. (3 .Hft) Yazılım İsterleri çözümlemesi,
4. (4 .Hft) Veri Akış diyagramı geliştirme, örnek uygulama,
5. (5 .Hft) Yazılım tasarımı,
6. (6 .Hft) Arayüz tasarımı
7. (7 .Hft) Yazılım gerçekleştirimi
8. (8 .Hft) Örnek Uygulamalar
9. (9 .Hft) Yazılım testi ve bakımı
10. (10.Hft) Yazılım kalitesi ve standartları
11. (11.Hft) Proje yönetimi
12. (12.Hft) Örnek Uygulamalar
13. (13.Hft) Başarılı Projelerin Ortak Yönleri
14. (14.Hft) Bilişim Sistemlerinde Değerlendirme ve Etik.

Bu ana başlıklarla verilen ders içeriğinde kod üretme teknikleri, Yazılım Mimarileri, CobiT ve popüler yazılım proje örnekleri gibi örnek çalışmalar gösterilecektir.

- Teorik anlatım
- Uygulama sunuları
- Ödev sunumları
- Rapor

DEĞERLENDİRME SİSTEMİ		
YARIYIL İÇİ ÇALIŞMALARI	SAYISI	KATKI PAYI
Ara Sınav(online sınav)	1	70
Kısa Sınav(online sınav)	2	20
Ödev(online sınav)	1	10
TOPLAM		100
Yılıçının Başarıya Oranı		50
Finalin Başarıya Oranı		50
TOPLAM		100

- Yazılım geliştirme sürecine ait metodolojiyi öğrenmesi ve uygulaması,
- Yazılım projesi geliştirebilmeli ve yönetebilmeli,
- Literatür araştırması ve takibi yapabilmeli,
- Ödev ve rapor hazırlayabilmeli,
- Sunum becerisini geliştirmeli,
- ...

Gerçek yaşamda gereksinim duyulan basit yada karmaşık yazılımların tasarımını, üretimini ve bakımını, zaman ve maliyet kısıtlarını da göz önünde bulundurarak etik ve mühendislik yaklaşımıyla öğrencilere tanıtmak ve aynı zamanda çeşitli bireysel araştırmalar ve grup çalışmalarıyla bu süreçlere yönelik uygulamalar yapmalarına olanak sağlamak.

- Mühendislik mi ?
- Bilim mi ?
- Endüstri mi ?
- Ürün mü?
- Sanat mı ?



- ✓ Değişik ve çeşitli görevler yapma amaçlı tasarlanmış elektronik araçların, birbirleriyle haberleşebilmesini ve uyumunu sağlayarak, görevlerini ya da kullanılabilirliklerini geliştirmeye yarayan makine(dili) komutlarıdır.
- ✓ Elektronik cihazların belirli bir işi yapmasını sağlayan programların tümüne verilen isimdir.
- ✓ Var olan bir problemi çözmek amacıyla bilgisayar dili kullanılarak oluşturulmuş anlamlı ifadeler bütünüdür.

- ✓ Bir bilgisayarda donanıma hayat veren ve bilgi işlemde kullanılan programlar, yordamlar, programlama dilleri ve belgelemelerin tümü.

- Bilgisayar programlarının tasarımı, geliştirilmesi, test edilmesi ve bakımı konularını ele alan mühendislik dalıdır.
- Temel mühendislik prensiplerinin bu dalda da uygulanması, önceden tahmin edilebilir ve tekrarlanabilir sonuçların daha çok elde edilmesiyle yazılım mühendisliği gerçek bir mühendislik dalı olma yolunda ilerlemektedir.
- Diğer mühendislik dallarıyla karşılaştırıldığında çok yeni olan bu alanda sürekli yeni yöntemler geliştirilmekte ve konu yavaş yavaş belli bir olgunluğa ulaşmaktadır.
- Gelişmekte olan bu alanda halen ihtiyaç çok büyük olduğundan geleceğin mesleklerinden birisi olarak gösterilmektedir.

- Bilgisayar sistemleri artık günlük hayatın her alanında yoğun ve etkin bir şekilde kullanılmakta olduğundan, bankacılıktan otomotiv sanayisine, sağlık bilgi sistemlerinden şirket yönetimine, telekomünikasyon sistemlerinden hava taşımacılığına, çok geniş alanlarda kullanılan bilgisayar sistemlerinin çok önemli ve kritik bir parçasını oluşturuyor.
- Bilgisayar Yazılım Mühendisliği 1968 yılında NATO tarafından gerçekleştirilen bir konferans esnasında ortaya çıkan yeni bir kavram ve yeni bir mühendislik alanı olup, yazılım sistemlerinin mühendislik prensipleri çerçevesinde tasarımı, üretimi ve işletilmesini hedeflemektedir.

“Yazılım Mühendisliği tüm disiplinlerde uygulamaları olan bir alandır. ”

- Yazılım mühendisliği disiplini 1968 yıllarından bu yana oldukça gelişme kaydetmiş; yazılım geliştirme metodolojileri, programlama paradigmaları, programlama dilleri ve çeşitli araçların geliştirilmesiyle hayli ilerleme katetmiştir.
- Gelişime , IEEE (IEEE Computer Society) ve ACM (Association for Computing Machinery) gibi mesleki kuruluşların önemli etkisi olmuştur.

- Ayrıca bu kuruluşlar son yıllarda, yazılım mühendisliği çekirdek bilgisinin tanımlanması ve bu bilgilerle uyumlu yazılım mühendisliği eğitim programlarının geliştirilmesine yönelik çalışmalar da yapmaktadır.
- Bu bağlamda, diğer mühendislik dallarında olduğu gibi yazılım mühendisliği için de ayrı eğitim programlarının oluşturulması gündeme gelmiştir. Yazılım mühendisliği disiplinin olgunlaşma sürecinde yazılım mühendisliği eğitimi özel bir önem kazanmıştır.

Özel bir önem kazanan bu eğitim programı için özel projelere başlanmıştır. Genelde kısaltılmış adlarıyla karşımıza çıkabilecek olan bu projeler;

SWEBOK(Software Engineering Body of Knowledge) :

Yazılım mühendisliği çekirdek bilgisinin tanımlanması.

SWCEPP(Software Engineering Code of Ethics and Professional Practice) :

Yazılım mühendisliği etiklerinin tanımlanması.

SWEEP (Software Engineering Education Project) :

SWEBOK ile uyumlu olarak örnek bir eğitim programı tanımlanması, olarak sıralanmaktadır.

Sonuç ;



Yazılım Mühendisliği eğitiminde teknik bilgi ve beceriler yanında hukuki kavramlar, etik değerler, takım çalışması, proje yönetimi gibi soyut fakat önemli kavramların da kişiye kazandırılması amaçlanmıştır.

- ✓ Sarıdoğan , E., Yazılım Mühendisliği, Papatya Yayınevi,2004.
- ✓ Arifoğlu, A., Doğru, A., Yazılım Mühendisliği, Sas Bilişim Yayınları,2004.
- ✓ Ders Notları.