



Bölüm 1- Giriş



Konular

- ◇ Profesyonel yazılım geliştirme
 - Yazılım mühendisliği nedir
- ◇ Yazılım mühendisliği etiği
 - Yazılım mühendisliği bağlamındaki etik konular
- ◇ Örnekler
 - Kitapta kullanılan örnekler ile ilgili genel bilgi.

Yazılım mühendisliği



Gelişmiş ülkelerin ekonomileri ile ilgili aşağıdaki olgular söylenebilir.

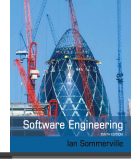
- ◇ Yazılıma bağımlıdır.
- ◇ Yazılım tarafından denetlenen dizgelerin sayısı artmaktadır.
- ◇ Yazılım mühendisliği, profesyonel yazılım geliştirme bağlamındaki kuram, yöntem ve araçlarla ilgilidir.
- ◇ Yazılım harcamaları devlet bütçesinin önemli bir bölümünü oluşturur.

Yazılım ederleri



- ◇ Yazılım ederleri bilgisayar dizgesi ederlerinin büyük bölümünü oluşturabilir. Kişisel bilgisayardaki yazılımın ederi, donanımın ederinden daha çok olabilir.
- ◇ Yazılımın sürekliliğini sağlamanın ederi, yazılım geliştirmenin ederinden daha çok olabilir. Yaşam süresi uzun olan dizgelerde, yazılımın sürekliliğini sağlamanın ederi (bakım ederi) geliştirmenin birkaç katı olabilir.
- ◇ Yazılım mühendisliği, düşük eder ile yüksek nitelikli yazılım geliştirme ile ilgilidir.

Yazılım projesi başarısızlığı

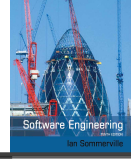


- ◇ Artan dizge karmaşıklığı
 - Yeni yazılım mühendisliği teknikleri ile daha büyük ve karmaşık dizgeler yapıldıkça, beklentiler de değişmektedir. Dizgelerin artık daha kısa sürelerde yapılması gerekmektedir. Daha büyük ve daha karmaşık dizgelere ihtiyaç duyulmaktadır. Dizgelerin, eskiden olanaksız olduğu düşünülen özellikleri olmalıdır.
- ◇ *Yazılım mühendisliği yöntemlerini kullanmama*
 - Yazılım mühendisliği yöntem ve tekniklerini kullanmadan da yazılım oluşturmak olanaklıdır. Yazılım mühendisliği yöntem ve tekniklerini kullanmayan şirketlerin yazılımları daha az güvenilir ve daha pahalıya malolmaktadır.



Profesyonel yazılım geliştirme

Yazılım mühendisliği ile ilgili sıkça sorulan sorular



Soru	Yanıt
Yazılım nedir?	Bilgisayar programları ve ilgili belgelendirim. Yazılım ürünleri belirli bir müşteri için veya (genel) pazar için tasarlanır.
İyi yazılımın özellikleri nelerdir?	İyi yazılım beklenen işlevselliği sağlamalıdır. Başarımı olmalıdır. Kolay bakımı yapılabilir, güvenilir, kullanılabilir olmalıdır.
Yazılım mühendisliği nedir?	Yazılım üretiminin bütün yönleri ile ilgilenen mühendislik disiplindir.
Temel yazılım mühendisliği etkinlikleri nelerdir?	Yazılım belirtimi, yazılım geliştirme, yazılım geçerleme ve yazılım evrimi.
Yazılım mühendisliği ve bilgisayar bilimi arasındaki fark nedir?	Bilgisayar bilimi kuram ve temellere odaklanır; yazılım mühendisliği yazılım geliştirme ve teslim etme sürecinin gerçek hayattaki olguları ile ilgilidir.
Yazılım mühendisliği ile dizge mühendisliği arasındaki fark nedir?	Dizge mühendisliği, bilgisayar tabanlı dizge geliştiriminin bütün yönleriye ilgilenir: donanım, yazılım ve süreç mühendisliğini de içerir. Yazılım mühendisliği bu genel sürecin bir parçasıdır.

30/10/2014

Chapter 1 Introduction

7

Yazılım mühendisliği ile ilgili sıkça sorulan sorular



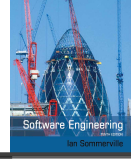
Soru	Yanıt
Yazılım mühendisliğindeki temel zorluklar nelerdir?	Artan çeşitlilik ile başa çıkma, kısalan teslim süreleri, güvenilir yazılım geliştirme
Yazılım mühendisliği ederleri nelerdir?	Yaklaşık %60 geliştirme ederi ve %40 sınaama ederi. Özel tasarım yazılım için evrim ederi geliştirme ederini geçebilir.
En iyi yazılım mühendisliği teknik ve yöntemleri nelerdir?	Bütün yazılım projelerinin profesyonel olarak yönetilmesi gerekliliğine karşın, her değişik tipteki dizge için değişik teknik uygundur. Örneğin oyunlar prototip kullanılarak geliştirilmelidir. Canguvenligi bağlamında düşünölebilen dizgelerde bütönsel ve çözümlenebilir belirtim oluşturulmalıdır. Dolayısıyla bir yöntemin diğlerinden daha iyi olduđu söylenemez.
Internet, yazılım mühendisliğini nasıl etkiledi?	Dağıtılmış hizmet tabanlı dizgelerin oluşturumunu sağlar. Web tabanlı dizge geliştirimi programlama dilleri ve yazılım yeniden kullanımında önemli gelişmelere olanak sağlar.

30/10/2014

Chapter 1 Introduction

8

Yazılım ürünleri



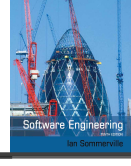
- ◇ Genelgeçer ürünler
 - Satın almak isteyen herkese yönelik ürünler.
 - Örnek – Çizim programları, proje yönetim araçları; CAD yazılımı; hasta takibi yazılımları.
- ◇ Özelleştirilmiş ürünler
 - Belirli bir müşterinin gereksinimlerini karşılamak için.
 - Örnek – gömülü denetim dizgeleri, hava trafik denetim yazılımı, trafik izleme yazılımları.

Yazılım belirtimi



- ◇ Genelgeçer ürünler
 - Yazılımın ne yapmasının gerektiğinin belirtimi geliştirici tarafından belirlenir. Yazılım bağlamındaki kararları geliştirici verir.
- ◇ Özelleştirilmiş ürünler
 - Yazılımın ne yapması gerektiğini müşteri belirler. Yazılım bağlamındaki değişiklik kararlarını müşteri verir.

İyi yazılımın temel özellikleri



Ürün özelliği	Tanım
Bakılabilirlik	Yazılım, müşterinin değişebilen isteklerine göre evrilebilecek biçimde tasarlanmalıdır. Bu oldukça önemlidir çünkü yazılımda değişiklik değişen iş ortamının doğal sonucudur.
Belbağlanırlık ve veri güvenliği	Yazılım belbağlanırlığı birçok değişik özellik içerir: bunlar arasında güvenilirlik, veri güvenliği ve can güvenliği vardır. Belbağlanırlık yazılım, dizge çökmesi durumunda fiziksel veya ekonomik zarara yol açmamalıdır. Kötü niyeti kullanıcılar dizgeye erişip zarar verememelidir.
Verimlilik	Yazılım, dizge kaynaklarını verimli kullanmalıdır. Bu kaynaklar arasında bellek ve işlemci vardır. Verimlilik; tepki verirlilik, işlem süresi ve bellek kullanımı ile ilgili olguları da içerir.
Kabul edilebilirlik	Yazılım, kullanımı için tasarlanan kullanıcılar tarafından kabul edilebilir olmalıdır. Bu bağlamda, yazılım anlaşılabilir, kullanılabilir ve kullanıcıların kullandığı diğer dizgeler ile uyumlu olmalıdır.

Yazılım mühendisliği



- ◇ Yazılım mühendisliği, dizge belirtiminin ilk aşamalarından dizgenin kullanılmaya başlandıktan sonraki dizge bakımına kadar, yazılım üretiminin bütün yönleri ile ilgilenen mühendislik disiplini.
- ◇ Mühendislik disiplini
 - Finansal ve kurumsal kısıtları göz önünde bulundurarak uygun kuram ve yöntemlerle sorun çözme
- ◇ Yazılım üretiminin bütün yönleri
 - Yalnızca geliştirmenin teknik süreci değil. Proje yönetimi, yazılım üretimini destekleyecek araç ve yöntemlerin geliştirimi.

Yazılım mühendisliğinin önemi



- ◇ Toplum ve kişiler, gelişmiş yazılım dizgelerine güvenerek yaşamlarını sürdürmektedir. Güvenilir dizgeleri düşük ederle ve çabuk üretmeliyiz.

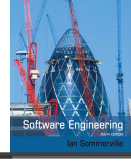
Yazılım mühendisliği yöntem ve tekniklerinin kullanımı, bireysel programlama projesiymiş gibi program yazmaya göre, uzun erimde, daha hesaplıdır. Birçok dizge için, ederin büyük bölümü, dizge kullanılmaya başladıktan sonraki ederlerdir.

Yazılım süreci etkinlikleri



- ◇ Yazılım belirtimi, müşteri ve mühendisler üretilecek yazılımı ve çalışma kısıtlarını belirler.
- ◇ Yazılım geliştirme, yazılım tasarlanır ve programlanır.
- ◇ Yazılım geçерleme, yazılımın müşterinin gereksinimlerini karşıladığı denetlenir.
- ◇ Yazılım evrimi, değişen müşteri ve pazar gereksinimlerini yansıtmak için yazılımı düzenleme.

Yazılımı etkileyen olgular



- ◇ Çoktürelilik (heterogeneity)
 - Hızla artan biçimde, dizgelerin farklı bilgisayar ve taşınır cihazlar içeren ağ üzerinde dağıtık dizgelerde çalışması gerekmektedir.
- ◇ İş ortamı ve toplumsal değişiklik
 - Ekonomi geliştikçe ve yeni teknolojiler gündeme geldikçe toplum ve iş dünyası değişmektedir. Varolan yazılımların değiştirilmesi, ve yeni yazılımlar üretilmesi gerekmektedir.

Yazılımı etkileyen olgular



- ◇ Veri güvenliği ve güven
 - Yazılım hayatımızın tüm yönleri ile iç içe olduğu için, yazılıma güvenebilmemiz çok önemlidir.
- ◇ Ölçek
 - Yazılım çok değişik ölçeklerde geliştirilmektedir: taşınabilir veya giyilebilir çok küçük gömülü dizgelerden, bulut tabanlı geniş bir kullanıcı kitlesi olan dizgelere kadar.

Yazılım mühendisliğinde çeşitlilik



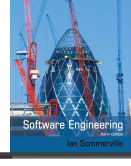
- ◇ Birçok değişik yazılım dizgesi tipi vardır. Dolayısıyla, bütün bu dizgelere uygulanabilecek evrensel yazılım teknikleri bulunmamaktadır.
- ◇ Yazılım mühendisliği yöntem ve araçları, geliştirilen uygulamaya, müşterinin gereksinimlerine ve geliştirici topluluğunun altyapısına bağlı olarak çeşitlilik gösterir.

Uygulama tipleri



- ◇ Stand-alone (tek başına) uygulamalar
 - Yerel bilgisayar (PC gibi) üzerinde çalışan uygulama dizgeleri. Bütün gerekli işlevselliği içerir ve ağa bağlı olması gerekmez.
- ◇ Etkileşimli transaction(işlembilgi)-tabanlı uygulamalar
 - Uzakta bir bilgisayar üzerinde çalışan ve kullanıcıların kişisel bilgisayarları veya terminaller üzerinden eriştiği uygulamalar. E-ticaret uygulamaları bu tipe örnektir.
- ◇ Gömülü denetim dizgeleri
 - Donanımları denetleyen ve yöneten yazılımlardır. Sayıca en çok olan dizge tipidir.

Uygulama tipleri



- ◇ Batch processing (kesikli işleme) dizgeleri
 - Veriyi büyük parçalar halinde işlemek için tasarlanmış dizgelerdir. Büyük miktarda girdiyi işleyip, karşılık gelen çıktıyı üretmek için tasarlanmışlardır.
- ◇ Eğlenceye yönelik dizgeler
 - Bu dizgeler genellikle kişisel kullanım içindir. Kullanıcının boş zamanını değerlendirmesine yöneliktir.
- ◇ Modelleme (biçeleme) ve simülasyon (benzetim) dizgeleri
 - Bilimciler ve mühendisler tarafından fiziksel süreç ve durumları biçelemek için tasarlanmış dizgelerdir.

Uygulama tipleri



- ◇ Veri toplama dizgeleri
 - Çevreden algılayıcılar aracılığıyla veri toplayan ve bu veriyi işlenmesi için başka dizgelere gönderen dizgelerdir.
- ◇ Dizge dizgeleri (systems of systems)
 - Birden çok dizgenin bir araya gelmesinden oluşan dizgelerdir.

Yazılım mühendisliği temelleri



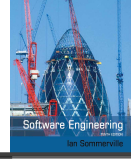
- ◇ Geliştirme tekniğinden bağımsız olarak, bazı temel ilkeler bütün tip yazılım dizgeleri için geçerlidir:
 - Dizgeler anlaşılabilir bir geliştirme süreci kullanılarak geliştirilmelidir. Değişik tipteki dizgeler için geçişik süreçler kullanılır.
 - Dependability (belbağlanırlık) ve performance (başarım) bütün dizge tipleri için önemlidir.
 - Yazılım belirtimi ve isterleri (yazılımın ne yapması gerektiği) anlamak ve yönetmek önemlidir.
 - Eğer uygunsa, yeni yazılım yapmak yerine varolan yazılımı kullanmak söz konusu olabilir.

İnternet ve yazılım mühendisliği



- ◇ Kurumlar, yerel dizgeler yerine internet tabanlı dizgeler geliştirmeyi tercih etmektedir.
- ◇ Web hizmetleri ile işlevselliğe ağ üzerinden erişilebilir.
- ◇ Cloud computing (bulut hesaplama) yaklaşımında hizmetler 'bulut' üzerinde çalışır.
 - Kullanıcılar yazılım satın almaz, kullanıma göre ödeme yapar.

Internet ve yazılım mühendisliği



- ◇ Web-tabanlı dizgelerin karmaşık dağıtık yapıları vardır. Buna karşın, yazılım mühendisliği ilkeleri bu tip dizgeler için de geçerlidir.
- ◇ Yazılım mühendisliğindeki temel düşünceler web-tabanlı dizgeler için de aynı biçimde geçerlidir.

Internet ve yazılım mühendisliği



- ◇ Yazılım yeniden kullanımı
 - Software reuse (yazılım yeniden kullanımı) web-tabanlı dizgelerde en önemli yaklaşımdır. Bu dizgeler oluşturulurken, hangi bileşen ve dizgelerin bir araya getirilebileceği düşünülmelidir.
- ◇ Artımlı ve çevik geliştirim
 - Web-tabanlı dizgeler artımlı olarak geliştirilmeli ve teslim edilmelidir. Bu tip dizgeler için bütün isterleri en başta belirtmeye çalışmak uygun değildir.

Internet ve yazılım mühendisliği

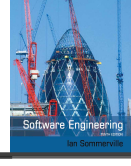


- ◇ Hizmet tabanlı dizgeler
 - Yazılım, hizmet tabanlı yazılım mühendisliği ile gerçekleşir: yazılım bileşenleri web hizmetleridir.
- ◇ Zengin arayüz
 - HTML5 ve AJAX ile tarayıcıda zengin bir arayüz oluşturmumu sağlanabilir.

Yazılım mühendisliğinde etik



Yazılım mühendisliğinde etik



- ◇ Yazılım mühendisliği, teknik konulardaki yeteneklerin kullanılmasından ibaret değildir.
- ◇ Yazılım mühendisleri, kendilerinden beklenen biçimde, dürüst olmalıdır.

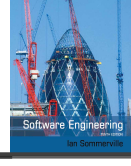
Etik davranış, yasalara uymakla sınırlı değildir. Etiksel olarak doğru ilkeleri takip etmeyi de içerir.

Profesyonel sorumluluk ile ilgili konular



- ◇ Gizlilik
 - Mühendisler, gizlilik anlaşması olmasa bile, çalışan ve müşteriler ile ilgili bilgileri paylaşmamalıdır.
- ◇ Mesleki yeterlilik
 - Mühendisler, yeterlilik düzeyleri konusunda dürüst olmalıdır. Yeterli olmadığı konularda iş almamalıdır.

Profesyonel sorumluluk ile ilgili konular



- ◇ Fikri mülkiyeti hakları
 - Mühendisler, patent gibi fikri mülkiyet düzenleyen yasalar hakkında bilgi sahibi olmalıdır. Çalışan ve müşterilerin fikri mülkiyet haklarının korunmasını sağlamalıdır.
- ◇ Bilgisayar kullanımının suistimali
 - Yazılım mühendisleri bilgisayarları doğru amaçlar için kullanmalıdır. İşverenin bilgisayarında oyun oynamak veya bilgisayar virüsü yaymak bilgisayar kullanımının suistimali alanına girmektedir.

ACM/IEEE Code of Ethics (Etik kodu)



- ◇ ABD'deki meslek kurumları etik uygulamalar ile ilgili bir metin hazırlamıştır.
- ◇ Bu kurumlara üye olanlar, üyelik aşamasında etik sözünü kabul ederler.
- ◇ Kodda toplamda sekiz ilke vardır.

Etik kodunun önemi



- *Yazılım mühendisleri, yazılım mühendisliği mesleğinin saygınlığını koruyabilmesi için, etik koda uymalıdır.*

The ACM/IEEE Code of Ethics



Software Engineering Code of Ethics and Professional Practice

ACM/IEEE-CS Joint Task Force on Software Engineering Ethics and Professional Practices

PREAMBLE

The short version of the code summarizes aspirations at a high level of the abstraction; the clauses that are included in the full version give examples and details of how these aspirations change the way we act as software engineering professionals. Without the aspirations, the details can become legalistic and tedious; without the details, the aspirations can become high sounding but empty; together, the aspirations and the details form a cohesive code.

Software engineers shall commit themselves to making the analysis, specification, design, development, testing and maintenance of software a beneficial and respected profession. In accordance with their commitment to the health, safety and welfare of the public, software engineers shall adhere to the following Eight Principles:

Etik ilkeler



1. PUBLIC - Software engineers shall act consistently with the public interest.
2. CLIENT AND EMPLOYER - Software engineers shall act in a manner that is in the best interests of their client and employer consistent with the public interest.
3. PRODUCT - Software engineers shall ensure that their products and related modifications meet the highest professional standards possible.
4. JUDGMENT - Software engineers shall maintain integrity and independence in their professional judgment.
5. MANAGEMENT - Software engineering managers and leaders shall subscribe to and promote an ethical approach to the management of software development and maintenance.
6. PROFESSION - Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.
7. COLLEAGUES - Software engineers shall be fair to and supportive of their colleagues.
8. SELF - Software engineers shall participate in lifelong learning regarding the practice of their profession and shall promote an ethical approach to the practice of the profession.

Etik ikilemler



- ◇ Yönetim ile anlaşmazlıklar.
- ◇ İşverenin, cangüvenliği açısından kritik bir sistemi, yeterince test etmeden piyasaya sürmek istemesi.



Örnek uygulamalar



Örnek uygulamalar

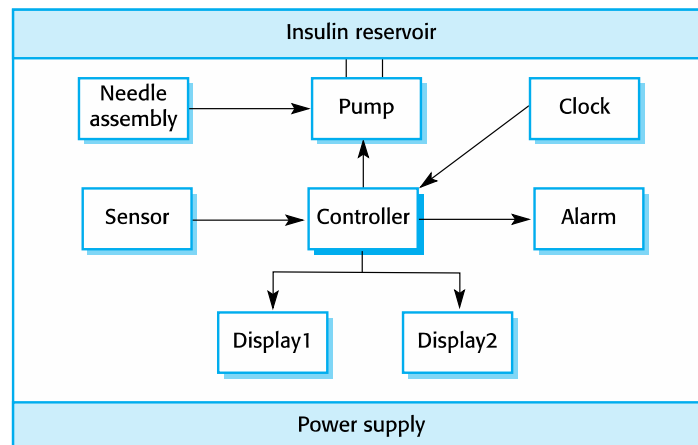
- ◇ Kişisel insülin pompası
 - Şeker hastalarının kan glüköz düzeylerini düzenlemek için kullanılan gömülü sistem.
- ◇ Ruh sağlığı için hasta bilgi sistemi
 - Mentcare. Tedavi gören sinir hastalarının kayıtlarını tutan sistem.
- ◇ Kır hava istasyonu
 - Uzak bölgelerdeki hava durum bilgilerini toplamak için veri toplama sistemi.
- ◇ iLearn: dijital öğrenme sistemi
 - Okullarda öğrenimi destekleme için sistem

İnsülin pompası kontrol sistemi

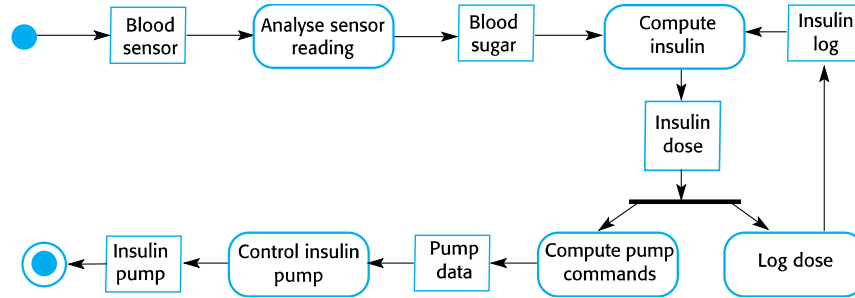


- ◇ Kan şekeri algılayıcısından veri toplar ve enjekte edilecek insulin miktarını hesaplar.
- ◇ Hesaplama, kan şekeri düzeyindeki artış hızını temel alır.
- ◇ Doğru dozda insülin göndermesi için mikro-pompaya sinyal gönderir..
- ◇ Emniyet-kritik bir sistemdir. Çünkü düşük kan şekeri beyin hasarı, koma ve ölüme neden olabilir. Yüksek kan şekeri göz ve böbrek hasarına neden olabilir.

İnsülin pompası donanım mimarisi



İnsülin pompası etkinlik modeli

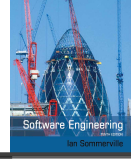


Yüksek düzey isterler



- ◇ Sistem, gerektiğinde insülin sağlayabilmeli.
- ◇ Sistem güvenilir bir sistem olmalı. Kan şekerine doğru dozda insülin verebilmeli.
- ◇ Sistem, bu isterleri sağlayabilecek biçimde tasarlanmalı ve gerçekleştirilmeli.

Mentcare: Ruh sağılıđı için hasta bilgi sistemi



- ◇ Hastaların ruh ve sinir sağılıđı sorunları ve tedavi bilgilerini tutmak için sağılık bilgi sistemi.
- ◇ Birçok hastanın hastane tedavisine gereksinimi yoktur. Özel kliniklerde, hastanın durumu hakkında bilgi sahibi olan doktor tarafından tedavi edilirler.
- ◇ Hastaların kolay ulaşımı için, bu klinikler yalnızca hastanelerde değıildir. Yerel muayenehaneler de bulunmaktadır.

Mentcare



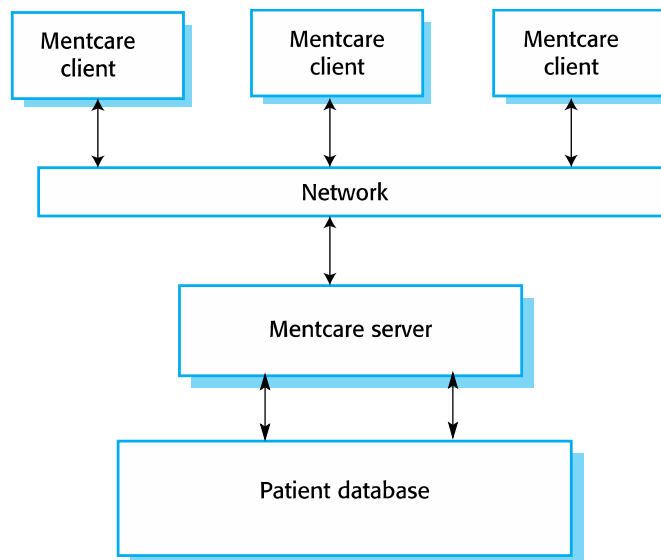
- ◇ Mentcare, kliniklerde kullanılan bir bilgi sistemidir.
- ◇ Merkezi veritabanı kullanır ama kişisel bilgisayar üzerinden çalışabilir. Dolayısıyla güvenli internet bağlantısı olmayan bölgelerde de kullanılabilir.
- ◇ Yerel sistemler, güvenli internet bağlantısı olduğunda, ana veritabanını kullanabilir. Diğer durumlarda, veritabanını indirip yerel kopya üzerinde çalışabilir.

Mentcare amacı

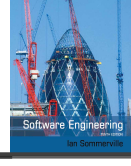


- ◇ Yerel ve ulusal hedeflere ulaşma düzeyini belirleyebilmek için yönetim bilgisi oluşturma.
- ◇ Sağlık personelinin hastaları ile ilgili bilgiye kolay ulaşımını sağlama.

Mentcare sisteminin organizasyonu



Mentcare sisteminin önemli özellikleri



- ◇ Bireysel tedavi yönetimi
 - Sağlık personele hasta kaydı oluşturabilir, sistemdeki bilgiyi güncelleyebilir, hasta tedavi geçmişini görüntüleyebilir, vs. Sistem veri özetleri oluşturur, böylece sağlık personeli önemli sorunları kolaylıkla görebilir..
- ◇ Hasta durumunun izlenmesi
 - Sistem tedavi edilen hastanın kaydını tutar ve bir sorun söz konusu ise uyarı mesajı gönderir.
- ◇ Yönetimsel raporlama
 - Sistem aylık rapor oluşturur. Raporla klinikteki hasta sayısı, yazılan ilaçlar ve ücretleri, vs. bulunur.

Mentcare sistemi: önemli noktalar



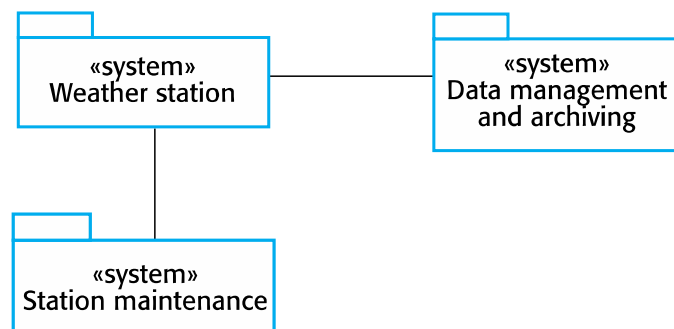
- ◇ Veri güvenliği
 - Hasta bilgileri özeldir. Yetkili sağlık personeli ve hastanın kendisi dışında erişilemez.
- ◇ Canggihüvenliği
 - Bazı akıl hastalıkları hastanın kendine ve çevresine tehdit oluşturabilmesine neden olabilir. Böyle durumlarda, sistem uyarı mesajı oluşturmaktadır.
 - Sistem her zaman erişilebilir olmalıdır. Erişilemez ise, canggihüvenliği sorunu oluşabilir: bu durumda hastalara doğru dozda ilaç verilemeyebilir.

Kır hava istasyonu



- ◇ Büyük orman alanları olan bir devlet, erişimi zor olan bölgelere birkaç yüz hava istasyonu kurmak istemektedir.
- ◇ Hava istasyonları, sıcaklık, basınç, günışığı, yağış miktarı, rüzgar hızı ve rüzgar yönü ölçen cihazlardan veri toplamaktadır.
 - Hava istasyonunda birçok algılayıcı bulunmaktadır. Bu algılayıcılar, düzenli veri toplayan ve işleyen sistemler tarafından yönetilir.

Hava istasyonunun çevresindeki sistemler



Hava bilgi sistemi



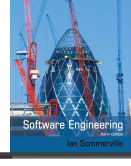
- ◇ Hava istasyonu sistemi
 - Hava durumu bilgisi toplama. Temel veri işleme ve veri yönetimi sistemine bilgi aktarma.
- ◇ Veri yönetimi ve arşivleme sistemi
 - Bu sistem, bütün hava istasyonlarından veri toplar, veriyi işler, veriyi arşivler.
- ◇ İstasyon bakım sistemi
 - Bu sistem, bütün hava durumu istasyonları ile iletişim kurar, bu sistemlerin durumu izler ve sorunları raporlar.

Diğer yazılım işlevsellikleri



- ◇ Algılayıcıları izleme, güç ve iletişim donanımını izleme, hataları yönetim sistemine aktarma.
- ◇ Sistem gücünü yönetme, bataryaların dolu olmasını sağlama, zorlu hava koşullarında, sistemin zarar görmesini engellemek için sistemi kapama.
- ◇ Dinamik ayar. Yazılımın yeni sürüm ile kolaylıkla değiştirilebilmesi, yedek donanım desteği.

iLearn: Dijital öğrenme sistemi



- ◇ Bazı genel amaçlı ve özel tasarım öğrenim araçlarının olduğu dijital öğrenme platformu.
- ◇ Sürümde olacak araçlar, öğretmen ve öğrencilerin gereksinimlerini karşılayacak biçimde özel olarak seçilecektir.
 - Spreadsheet, uzaktan öğretim sistemi (ödev gönderme ve değerlendirme, oyun ve benzetim).

Hizmet yönelimli sistem



- ◇ Her bileşen güncellenebilir bir hizmet olacaktır.
- ◇ Böylelikle, yeni hizmetler tasarlandıkça, sisteme eklenebilecektir.
- ◇ Böylelikle değişik gruplar için (okul öncesi çağındaki çocuklar vb.) sistem kişiselleştirilebilir.

iLearn hizmetleri



- ◇ Ara hizmetler: diğer hizmetlerin kullanacağı hizmetler.
- ◇ Uygulama hizmetleri: email, görüşme, resim paylaşma, belgesellere erişim.
- ◇ Ayar hizmetleri: platform ayarları. Hizmetlerin öğrenci, öğretmen ve veliler arasında nasıl kullanıldığının tanımlanması.

iLearn mimarisi



Browser-based user interface

iLearn app

Configuration services

Group
management

Application
management

Identity
management

Application services

Email Messaging Video conferencing Newspaper archive
Word processing Simulation Video storage Resource finder
Spreadsheet Virtual learning environment History archive

Utility services

Authentication Logging and monitoring Interfacing
User storage Application storage Search

iLearn hizmet entegrasyonu



- ◇ *Integrated services* are services API (application programming interface) ile diğer APIler ile iletişim. Hizmetler arası iletişim olanaklı.
- ◇ *Independent services* tarayıcı ile erişilen hizmetler. Bağımsız çalışan hizmetler. Hizmetler arasında bilgi değişimi yalnızca kullanıcı tarafından yapılabilir: copy, paste. Bu durumda yeniden kullanıcı adı ve şifre girmek gerekebilir.

Ana noktalar



- ◇ Yazılım mühendisliği, yazılım üretiminin tüm yönleri ile ilgili mühendislik dalıdır.
- ◇ Temel yazılım özellikleri, belbağlanırlık, veri güvenliği, verimlilik ve kabul edilebilirliktir.
- ◇ Belirtim, geliştirim, geçерleme ve evrim etkinlikleri bütün yazılım süreçlerinde bulunur.
- ◇ Yazılım mühendisliğinin temel olguları bütün sistem geliştirmesi tiplerinde geçerlidir.

Ana noktalar



- ◇ Birçok sistem tipi vardır. Her tipin kendine özgü yazılım mühendisliği araçları ve teknikleri söz konusudur.
- ◇ Yazılım mühendisliğindeki temel fikirler bütün yazılım sistemleri için geçerlidir.
- ◇ Yazılım mühendislerinin yazılım mühendisliği mesleğine ve topluma karşı sorumlulukları vardır. Yalnızca teknik konularla ilgilenmemelidirler.
- ◇ Meslek örgütleri, yazılım mühendislerine yön vermesi için, belirli belgeler yayımlar.