

YMH 213 MESLEKİ İNGİLİZCE

Dr. Öğr. Üyesi Bihter DAŞ

Software Engineering (Yazılım Mühendisliği)

Various programming languages are used to tell the computers what people want and to communicate with computers. As computers have **evolved**, demands and expectations of people have increased and therefore some issues have **emerged** other than communication.

Bilgisayarlara insanların ne istediğini söylemek ve bilgisayarlarla iletişim kurmak için çeşitli programlama dilleri kullanılır. Bilgisayarlar geliştikçe insanların talepleri ve beklentileri artmış ve bu nedenle iletişim dışında bazı sorunlar ortaya çıkmıştır.

System and Software (Sistem ve Yazılım)

A computer system consists of *hardware*, *software* and *interactive usage* required to operate them. Therefore, software should always be considered as one of the main *component* of the system. Today, software is considered not as a computer program or a package *performing* a specific job, but as «*program*», «data» and «a *collection* of documents» created within a certain logic.

Bir bilgisayar sistemi, bunları çalıştırmak için gerekli olan *donanım*, *yazılım* ve *etkileşimli kullanımdan* oluşur. Bu nedenle, yazılım her zaman sistemin ana bileşenlerinden biri olarak düşünülmelidir. Günümüzde yazılım, bir bilgisayar programı veya belirli bir işi gerçekleştiren bir paket olarak değil, belirli bir mantık içinde oluşturulan "program", "veri" ve "bir belge koleksiyonu" olarak kabul edilmektedir.

Computer System Engineering (Bilgisayar Sistem Mühendisliği)

System engineering is a problem solving activity that is implement in any kind of system. Computer system engineering is the general term that covers both hardware engineering and software engineering in system development activities.

Sistem mühendisliği, her türlü sistemde uygulanan bir problem çözme faaliyetidir. Bilgisayar sistem mühendisliği, sistem geliştirme faaliyetlerinde hem donanım mühendisliğini hem de yazılım mühendisliğini kapsayan genel bir terimdir.

System Development Process (Sistem Geliştirme Süreci)

Development, testing and maintenance of software and hardware elements, which **make up** a computerized system, should be done by following a specific **methodology**. According to the property, size, production number of the system, hardware **acquisition**, **prototype** development and production stages can also be in question together with the **process** of software development.

Bilgisayarlı bir sistemi oluşturan yazılım ve donanım unsurlarının geliştirilmesi, test edilmesi ve bakımı belirli bir metodoloji izlenerek yapılmalıdır. Sistemin özelliğine, büyüklüğüne, üretim sayısına göre donanım kabulü, prototip geliştirme ve üretim aşamaları da yazılım geliştirme süreciyle birlikte söz konusu olabilir.

Software Engineering (Yazılım Mühendisliği)

Since software engineering is not based on very **solid foundations**, its sub-processes have artistic features in many ways. Although standards are not defined, development is human-focused and there is a **great** quest to develop software more quickly and reliably.

Yazılım mühendisliği çok sağlam temellere dayanmadığından, alt süreçleri birçok yönden sanatsal özelliklere sahiptir. Standartlar tanımlanmamış olsa da, geliştirme insan odaklıdır ve yazılımı daha hızlı ve güvenilir bir şekilde geliştirmek için büyük bir arayış vardır.

Software Engineering Methodology

(Bir Algoritma İçin Gerekli Özellikler)

Methodology is a discipline that covers the whole methods related to process, specification, documentation etc. to be used throughout the software lifecycle. Until today, many different methodologies emerged and have been put into practice.

Metodoloji süreç, şartname, dokümantasyon vb. ile ilgili tüm yöntemleri kapsayan bir disiplindir, yazılım yaşam döngüsü boyunca kullanılacaktır. Bugüne kadar birçok farklı metodoloji ortaya çıkmış ve uygulamaya konulmuştur.

Software Development Models

(Yazılım Geliştirme Modelleri)

Software development phases and documents to be created depend on the chosen methodology, customer and developer. Commonly used models include:

- ▶ Classical cycle model
- ▶ V model
- ▶ Prototyping
- ▶ Spiral model
- ▶ Evolutionary development
- ▶ Incremental development
- ▶ New techniques

Software Development Process (Yazılım Geliştirme Süreci)

Software development process is one of the most important development activities in the computer systems engineering and **consists of** specific stages. Since any software is considered as a system, firstly it is determined what functions a system will perform and which hardware and software requires. After defining the scope of the software, software development process begins.

Yazılım geliştirme süreci, bilgisayar sistemleri mühendisliğinde en önemli geliştirme faaliyetlerinden biridir ve belirli aşamalardan oluşur. Herhangi bir yazılım bir sistem olarak düşünüldüğünden, öncelikle bir sistemin hangi işlevleri yerine getireceği ve hangi donanım ve yazılımın gerektirdiği belirlenir. Yazılımın kapsamını belirledikten sonra yazılım geliştirme süreci başlar.

Software Requirements Analysis (Yazılım Gereksinim Analizi)

For the success of the software development process, **customer's need** and **software requirements** need to be understood very well. No matter how well designed and written, a computer program cannot be considered successful if it does not meet **customer's needs** completely.

Yazılım geliştirme sürecinin başarısı için müşterinin ihtiyaç ve yazılım gereksinimlerinin çok iyi anlaşılması gerekir. Ne kadar iyi tasarlanmış ve yazılmış olursa olsun, bir bilgisayar programı, müşterinin ihtiyaçlarını tam olarak karşılamıyorsa başarılı sayılamaz.

Software Design (Yazılım Tasarımı)

Software design is similar a building's foundation. A building that does not have a strong foundation and consists of unplanned floors cannot be earthquake-resistant. Also it will not be possible to add new floors.

Yazılım tasarımı bir binanın temeline benzetilir. Sağlam temeli olmayan, plansız katlardan oluşan bir bina depreme dayanıklı olamaz. Ayrıca yeni katlar eklemek mümkün olmayacaktır.

Software Implementation (Yazılım Uygulama)

Analysis and design help to formulate ideas so that the machines can understand. In the phase of **implementation**, the **design** is transformed into the **programming language**, in other words it is **coded**. In order to tell the jobs to be done to the computer, an **appropriate** programming language is used.

Suitable

Analiz ve tasarım, makinelerin anlayabilmesi için fikirlerin formüle edilmesine yardımcı olur. Uygulama aşamasında tasarım, programlama diline dönüştürülür, yani kodlanır. Bilgisayara yapılacak işleri anlatmak için uygun bir programlama dili kullanılır.

Software Testing (Yazılım Deneme)

Software test is the last **control** of analysis, design and coding stages. In its most general form, a software product testing means running the software system or one of its unit in order to find **errors** before delivering the user. The presence of a possible error within the software refers to the **mistakes** in code that are done by encoder and it may be resulted in **incorrect** software product.

Yazılım testi, analiz, tasarım ve kodlama aşamalarının son kontrolüdür. En genel haliyle, bir yazılım ürünü testi, kullanıcıyı teslim etmeden önce hataları bulmak için yazılım sistemini veya bir birimini çalıştırmak anlamına gelir. Yazılım içerisinde olası bir hatanın varlığı, kodlayıcı tarafından yapılan kod hatalarını ifade eder ve hatalı yazılım ürünü ile sonuçlanabilir.

Software Maintenance

(Yazılım Bakımı)

After the design and development and then delivery of computer-based systems, maintenance stage begins. Maintenance of hardware elements of the system includes **cleaning** the elements, **replacing** worn or **malfunctioning** parts or replacing maintenance activities are performed which focus on **change**.

Bilgisayar tabanlı sistemlerin tasarımı, geliştirilmesi ve teslim edilmesinin ardından bakım aşamasına başlar. Sistemin donanım elemanlarının bakımı; elemanların temizlenmesini, aşınan veya arızalı parçaların değiştirilmesini veya değişim odaklı bakım faaliyetlerinin yapılmasını içerir.

References

- Rifat Çölkesen, Introduction to Computer Engineering
- Doç. Dr. Murat Karabatak ders notları
- <https://www.pearson.com/>
- <https://www.britishcouncil.al/>