

1. Farklı doküman tipleri arasından hangisi kullanılmalıdır?

Gereksinim analizi süreci, yazılım geliştirme sürecindeki en önemli aşamalardan biridir ve projenin başarısı için kritik derecede önem taşır. Bu süreçte kullanılacak dokümanlar, projenin özelliklerine, boyutuna ve gereksinimlerine bağlı olarak değişebilir. İş analisti veya yazılım geliştirme ekibi, hangi dokümanın hangi aşamada ve hangi amaçla kullanılacağına karar vermelidir. Bazı yaygın doküman tipleri ve kullanım alanları şunlardır:

- BRD (Business Requirement Document): İş gereksinimleri belgesi, işletme çözümünün ayrıntılarını ve işletme ihtiyaçlarını belirtir.
- SRS (Software Requirement Specification): Yazılım gereksinimleri spesifikasyonu, sistem gözünden gereksinimleri dokümante eder.
- Vision and Scope Document: Vizyon ve kapsam belgesi, projenin neden gerekli olduğunu, yapılacak işin kapsamını, özellikleri ve projedeki paydaşları tanımlar.
- Product Roadmap and Product Backlog: Agile projelerde kullanılan yöntemlerdir. Product roadmap, projenin yol haritasını belirlerken product backlog ise yapılacak işlerin listesini ve önceliklendirmesini sağlar.
- RFI, RFQ, RFP: Tedarikçilerle çalışılacak projelerde kullanılan dokümanlardır. RFI (Request for Information), RFQ (Request for Quote) ve RFP (Request for Proposal) dokümanları, tedarikçilerden bilgi, teklif ve öneriler almak için kullanılır.

Hangi dokümanın hangi aşamada kullanılacağı ve nasıl hazırlanacağı, projenin özelliklerine ve gereksinimlerine bağlı olarak değişiklik gösterebilir.

2. Gereksinimleri analiz ediyorum benim için hangi format uygun?

Gereksinim analizi sürecinde farklı formatlarda çıktılar hazırlanarak iş birimlerinin ve yazılım ekiplerinin daha iyi anlaması sağlanabilir. Analiz dokümanlarının içeriğinin ve formatının, paydaşların ihtiyaçlarına ve anlama düzeylerine uygun olmalıdır. Bu doğrultuda, Data Dictionary ve Glossary gibi belgelerle veri ve

terimlerin tanımları netleştirilebilir, Decision Table ile iş kuralları açık ve okunabilir bir şekilde sunulabilir, wireframe ile ekranları tasarlarken görsel bir referans sağlanabilir. Ayrıca, iş akışları ve veri modelleri gibi daha teknik detaylar, yazılım ekiplerinin daha iyi anlaması için çizilebilir. Hazırlanan bu çalışmaları BRD, SRS gibi formatlarda dokümanlara dahil ederek de paylaşılabılır.

3. Hazır şablonlar kullandığımız dokümanlarda birçok alan var bu alanları dokümanda tutmalı mıyız?

Hazır şablonlar genellikle genel gereksinimleri karşılamak için tasarlanmış olabilir ve her şirket veya proje için tam olarak uygun olmayabilir. Şablonlardaki gereksiz veya doldurulamayan alanları çıkararak, dokümanları ihtiyaca göre revize etmek iyi bir yaklaşım olabilir. Doküman şablonunda yer alan bazı alanlar proje için gereksiz bulunursa, bu alanlar çıkarabilir ve gerekirse bu kısımların neden çıkarıldığını kısa açıklamalarla belirtilebilir.

4. Dokümanlara onay almak zor mu?

Dokümanlara onay almak genellikle zor olabilir çünkü paydaşların farklı beklentileri, bilgi seviyeleri ve zaman baskıları olabilir. İş birimleri veya yazılımcılar gibi paydaşların dokümanları okumakta zorlanması veya eksik anlaması, onay sürecini uzatabilir veya karmaşıktırabilir. Bu sorunları aşmak için şu stratejiler izlenebilir:

- Paydaşları Belirleme: Dokümanı kimlerin okuması ve onaylaması gerektiği belirlenir. Kullanıcı ve teknik gereksinimleri ayrı dokümanlarda sunarak ilgili grupların sadece kendi alanlarını incelemesi sağlanabilir.

- Anlaşılabilir Dil Kullanımı: Dokümanları mümkün olduğunca sade ve anlaşılır bir dilde yazmaya özen gösterilmelidir. Teknik terimlerin ve jargonun kullanımı minimize edilmelidir.
- Gereksiz Detaylardan Kaçınma
- Geri Bildirim Talebi
- Onay Süreci Yönetimi: Dokümanın hangi kısımlarının hangi paydaşlar tarafından okunup onaylanması gerektiğini belirleme ve bu sürecin yönetimi.
- İletişimi Güçlendirme

5. Agile ile yazılım geliştirilirken gerçekten dokümana ihtiyaç olmuyor mu?

Agile'in, "Working software over comprehensive documentation" ilkesi, çalışan bir yazılımın detaylı dokümantasyondan daha önemli olduğunu vurgular, ancak dokümantasyonunun hiç olmadığı anlamına gelmez.

Agile projelerde, gereksinimler genellikle User Story ve Acceptance Criteria gibi kısa ve net şekilde tanımlanır. Bu yaklaşım, gereksinimlerin daha esnek ve kullanıcı odaklı bir şekilde yönetilmesini sağlar. Ancak, bazı durumlarda detaylı analizler veya belgeler gerekebilir. Önemli olan, dokümanların amacına hizmet eden, gereksiz detaylardan kaçınan ve projenin ihtiyaçlarına uygun olarak hazırlanan belgeler olmasıdır.

6. IIBA nedir? Hangi doküman tiplerini standart olarak kabul etmiştir?

IIBA (International Institute of Business Analysis), iş analizi alanında uzmanlaşmış bir kuruluştur. İş analisti profesyonellerinin gelişimine katkıda bulunmak, iş analizi uygulamalarını teşvik etmek ve iş dünyasına değer katmak

amacıyla faaliyet gösterir. IIBA, iş analizi alanında standartlar oluşturarak, eğitim ve sertifikasyon programları düzenleyerek, iş analisti topluluğunu bir araya getirerek bu alandaki profesyonellerin bilgi ve becerilerini artırmayı hedefler.

IIBA tarafından kabul gören bazı doküman tipleri şunlardır:

- BRD (Business Requirement Document): İş gereksinimlerinin ayrıntılı olarak tanımlandığı bir belgedir. Müşteri ihtiyaçlarını, projenin amaçlarını ve kullanıcı gereksinimlerini içerir.
- SRS (Software / System Requirement Specification): Sistem gereksinimlerinin belgelendiği dokümandır. Sistem özellikleri, arayüzler, fonksiyonel ve fonksiyonel olmayan gereksinimler gibi detayları içerir.
- Vision and Scope Document: Proje veya ürünün kapsamını, neden gerekli olduğunu, tanımlanan özellikleri ve projedeki paydaşları tanımlayan bir belgedir. BRD yerine kullanılabilecek daha kısa ve genel bir doküman olabilir.
- Product Roadmap and Product Backlog: Agile projelerde kullanılan bir yöntemdir. Ürünün yol haritasını ve önceliklendirilmiş özellikleri içerir.
- RFI, RFQ, RFP: Tedarikçilerle çalışılacak projelerde kullanılan belgelerdir. Tedarikçilerden bilgi almak, teklif istemek ve teklif almak için kullanılır.

7. BRD nedir? Hedefleri nelerdir?

BRD (Business Requirement Document), iş gereksinimlerinin ayrıntılı olarak tanımlandığı bir belgedir. Bu belge, bir projenin neden gereksinim duyulduğunu, neyi başarmayı hedeflediğini ve projenin nasıl bir iş çözümü sağlayacağını açıklar.

BRD'nin temel hedefleri şunlardır:

- İş gereksinimlerinin tanımlanması: BRD, işletme gereksinimlerini belirler ve bu gereksinimlerin proje kapsamına dahil edilmesini sağlar. Bu, işletmenin neyi başarmak istediğini ve projenin bu hedefleri nasıl desteklediğini açıkça belirtir.

- Projenin hedeflerinin netleştirilmesi: BRD, projenin amaçlarını ve hedeflerini açıkça tanımlar. Bu, projenin neden gerçekleştirildiğini ve hangi sonuçların elde edilmesinin hedeflendiğini anlamak için önemlidir.
- Kapsamın belirlenmesi: BRD, projenin kapsamını belirler ve hangi iş süreçlerinin, fonksiyonların veya özelliklerin projeye dahil edileceğini tanımlar. Bu, projenin sınırlarını belirleyerek gereksinimlerin netleştirilmesine yardımcı olur.
- Paydaşların belirlenmesi: BRD, projenin etkilenecek paydaşlarını tanımlar ve onların beklentilerini ve gereksinimlerini belirler. Bu, paydaşların projeye doğru şekilde dahil edilmesini sağlar.
- Risklerin ve kısıtlamaların belirlenmesi: BRD, projenin karşılaşılabileceği potansiyel riskleri ve kısıtlamaları tanımlar. Bu, proje planının risklere ve kısıtlamalara uygun şekilde oluşturulmasını sağlar.
- Proje başlangıcında bir temel oluşturulması: BRD, projenin başlangıcında bir temel oluşturur ve projenin ilerlemesini izlemek için bir referans noktası sağlar.
- Proje ekibinin ve paydaşlarının yönlendirilmesi: BRD, proje ekibine ve paydaşlara projenin hedeflerini, gereksinimlerini ve kapsamını netleştirir. Bu, projenin başarılı bir şekilde tamamlanmasını sağlamak için önemlidir.

8. SRS nedir? Hedefleri nelerdir?

SRS (Software Requirement Specification), yazılımın gereksinimlerinin ayrıntılı olarak tanımlandığı bir belgedir. Bu belge, yazılımın neyi başarmayı amaçladığını, nasıl çalışacağını ve hangi gereksinimleri karşılayacağını açıklar.

SRS'nin temel hedefleri şunlardır:

- Yazılımın gereksinimlerinin tanımlanması: SRS, yazılımın işlevsel ve teknik gereksinimlerini tanımlar.
- Kullanıcı ihtiyaçlarının belirlenmesi: SRS, kullanıcıların ihtiyaçlarını ve beklentilerini belirler. Bu, yazılımın kullanıcı dostu ve işlevsel olmasını sağlar.

- Yazılımın arayüzlerinin tanımlanması: SRS, yazılımın kullanıcı arayüzünü, sistem arayüzlerini ve diğer arayüzleri tanımlar. Bu, yazılımın nasıl etkileşime gireceğini ve diğer sistemlerle nasıl entegre olacağını belirler.
- Performans ve güvenilirlik gereksinimlerinin belirlenmesi: SRS, yazılımın performans ve güvenilirlik gereksinimlerini tanımlar. Bu, yazılımın ne kadar hızlı çalışması gerektiğini ve ne kadar güvenilir olması gerektiğini belirler.
- Test gereksinimlerinin belirlenmesi: SRS, yazılımın test edilmesi gereken gereksinimlerini tanımlar. Bu, yazılımın doğru çalıştığını ve kullanıcı ihtiyaçlarını karşıladığını doğrulamak için gereken testleri belirler.
- Kısıtların ve hedeflerin belirlenmesi: SRS, yazılım geliştirme sürecinde karşılaşılabilecek kısıtları ve hedefleri belirler. Bu, yazılımın geliştirme sürecinin planlanması ve yönetilmesi için önemlidir.
- Proje ekibinin yönlendirilmesi: SRS, yazılım geliştirme sürecinde proje ekibinin ve paydaşların yönlendirilmesine yardımcı olur. Bu, yazılımın doğru şekilde geliştirilmesini ve projenin başarılı bir şekilde tamamlanmasını sağlar.

9. BRD kadar detaylı bir çalışmaya ihtiyaç duyulmayan daha küçük işlerde/projelerde kullanılabilecek başka bir doküman tipi var mıdır? Açıklayınız.

Evet bulunmaktadır. Bu dokümanlar genellikle projenin kapsamını ve gereksinimlerini daha hızlı ve basit bir şekilde belirlemeyi amaçlar. Bu tür dokümanlardan bazıları şunlardır:

- Vision and Scope Document: Bu belge, projenin neden gerektiğini, projenin kapsamını, projenin hedeflerini ve projenin paydaşlarını tanımlar.
- Project Charter: Proje başlatma belgesi olarak da adlandırılan bu belge, proje için resmi bir başlangıç yapar. Projenin amacını, kapsamını, hedeflerini, paydaşlarını ve proje yönetim ekibini tanımlar.

- Agile User Stories: Bir kullanıcının veya paydaşın bir özelliğe ilişkin bir ihtiyacını veya beklentisini ifade eder. Bu, gereksinimleri daha küçük parçalara bölerek daha yönetilebilir hale getirir.
- Product Backlog: Agile projelerde kullanılan bir başka doküman tipi de ürün gereksinimlerinin listelendiği Product Backlog'dur. Burada, projenin geliştirilmesi gereken özellikleri, hikayeleri veya görevleri içeren bir liste bulunur.

10. User story nedir? Projenizden örneklerle açıklayınız.

User Story, yazılım geliştirme sürecinde kullanıcıların ihtiyaçlarını ve beklentilerini tanımlamak için kullanılan bir tekniktir. Agile yazılım geliştirme metodolojilerinde sıkça kullanılan bir yöntemdir ve gereksinimleri daha kullanıcı odaklı ve anlaşılır bir şekilde ifade etmeyi sağlar.

User Story, genellikle şu formatta yazılır: "As a [role], I want [goal/desire] so that [benefit]."

Bu formatta, rol kullanıcının rolünü veya kimliğini tanımlar, hedef/dilek kullanıcının isteğini veya hedefini belirtir ve fayda, kullanıcının bu hedefi veya dileği gerçekleştirmesinin sağladığı faydayı açıklar.

UTM tabanlı birleştirilmiş SIEM projesi için bazı kullanıcı hikayeleri:

- Güvenlik uzmanı olarak, güvenlik duvarlarından saldırı tespit sistemlerine ve antivirüs yazılımlarından zafiyet tarama araçlarına kadar geniş bir güvenlik yelpazesini birleştirerek kurumumuzun siber tehditlere karşı direncini artırmak istiyorum.
- Güvenlik yöneticisi olarak, yönetimi basitleştiren ve güvenlik açıklarını azaltan bir UTM cihazı aracılığıyla tüm güvenlik işlemlerini tek bir yönetim noktasından kolaylıkla izleyebilme ve yönetebilme istiyorum.