

<9999> PROJESİ

<AAAA SİSTEMİ/ALTSİSTEMİ>

<XXX YAZILIM KONFIGURASYON BİRİMİ>

YAZILIM TEST TANIMI

Onay- Approved by	YAZILIM XXX (İngilizce ve Türkçe olarak iki dilde yazılır.)					Doküman/Parça Numarası- Document/Part Number	
Kontrol- Checked by						YZ-XXXX-YYYY	
Hazırlayan- Prepared by						Rev Kodu- Rev Code	Değişiklik Tarihi- Change Date
Bölüm- Department	Yazım Ortamı- Editing Env	Dok Kodu- Doc Code	Sytl/Syflr-Pg/Pgs	Tarih- Date	Dil- Lng	Değiştiren- Changed by	Boyut- Size
UGES/<BÖLÜM>	Word xxxx	043	1 / 23	GG/AA/YY	33		A4

## Doküman Genel Kuralları

Yazılım Test Tanımı (YTET) şablonu, J-STD-016 “Standard for Information Technology Software Life Cycle Processes Software Development” standardı içinde yer alan STD (Software Test Description) şablonu temel alınarak hazırlanmıştır.

YTET, UGSD-SG-20 süreç dokümanında tanımlanan Yazılım Geliştirme Süreci’nin “22 Test Tanımlarını ve Gereksinimlerini Belirle” adımı çıktısıdır.

YTET şablonu kullanılarak hazırlanan YTET dokümanının kontrolü UGFR-SG-25 YTET Kontrol Listesi kullanılarak yapılmalıdır.

Bu bölümde şablon kullanıcıya yardımcı olması amacıyla genel açıklamalar verilmiştir.

## Şablon ile ilgili genel açıklamalar:

1. Şablonun kılavuz bölümünde, “Yazılım Konfigürasyon Birimi” yerine Yazılım ifadesi kullanılmıştır.
2. YTET şablonunun içeriği J-STD-016 standardı içinde verilen şablon ile uyumludur. J-STD-016 standardındaki açıklamalar, özgün metne uygun olarak Türkçe’ye çevrilmiş ve ilgili başlık altında verilmiştir. J-STD-016 standardından gelen metin, kırmızı (“regular”) font kullanılarak yazılmış ve çerçeve içine alınarak şablonda verilen diğer açıklama ve örneklerden farklı olması sağlanmıştır.
3. J-STD-016 standardı şablon metnini daha anlaşılır kılmak için veya Aselsan UGES Grubu’nda standart şablon metninde belirtilenlerden farklı uygulamalar varsa “Kılavuz” bölümünde açıklamalar verilmiştir. Standart şablon metni ve kılavuz metni bir bütün oluşturmakta olup birlikte değerlendirilmelidir.
4. J-STD-016 standardından gelen metin ve kılavuz kısmındaki açıklamaların yeterli görülmediği durumlarda, anlatımı güçlendirmek için aşağıdaki biçimde örnek metinler yerleştirilmiştir.

‘[Örnek]

...

[Örnek Sonu]’

Örnek metinler, şablonun ilgili bölümlerindeki açıklamaların anlaşılabilirliğini arttırmak amacıyla verilmiştir. Bu metinler dokümanın amacı ve içeriği göz önünde tutularak gerektiği şekilde düzeltilmelidir.

5. YTET şablonu için J-STD-016 standardı Türkçe’ye çevrilirken;
  - Zorunluluk (“shall”) bildiren ifadeler ‘-ecektir’ veya ‘-acaktır’ (yapacaktır vb.),
  - Yapılması istenen ifadeler (“should”) ‘-malı’ veya ‘-meli’ (yapmalı vb.),
  - Yapılabilirlik bildiren (“may”, “could”) ifadeler ise ‘-bilir’, (yapabilir vb.),
  - Açıklamalar (“will”) ‘-ar’ veya ‘-er’ (yapar vb.),

şeklinde kullanılmıştır.

## Şablonun doldurulması ile ilgili kurallar:

1. YTET şablonunda verilen bölüm isimlerinin değiştirilmemesi ve bölümlerin silinmemesi gerekmektedir. Şablondaki bölümlerden hazırlanan doküman için doldurulması uygun olmayanlar var ise başlığı atlamak veya silmek yerine, başlığın altına “Bu bölüm uygulanabilir değildir.” veya “Yoktur” gibi bir ifade yazılmalıdır.

2. Şablonda kullanılan yazım biçim özellikleri (“word style”) zorunlu olmadıkça değiştirilmemelidir.
3. Dokümanın kâğıt ortamda okunduğunda anlaşılır ve takip edilebilir olabilmesi için, doküman içinde bağlantı adresi görünmeyen “hyperlink” kullanılmamalıdır.
4. Doküman içerisinde Türkçe olmayan kelimelerin kullanılmasından mümkün olduğunca kaçınılmalı, kullanımı bir zorunluluk olarak gerekiyorsa çift tırnak içerisinde verilmelidir.

#### **Dokümanın yayınlanması ve dokümantasyona aktarma ile ilgili kurallar:**

1. Doküman yayınlanırken doküman içeriği dışında kalan tüm açıklama, standart şablon metni, kılavuz metin, örnekler v.b. dokümandan silinmelidir.
2. Dokümanın her yayınlanışında ilk sayfasındaki tanım, parça no, revizyon, onay, kontrol, hazırlayan, bölüm, ortam gibi bilgilerin kontrol edilmesi ve dokümantasyon kuralları doğrultusunda güncellenmesi gerekmektedir. Ayrıca, “header” ve “footer” alanlarının da güncellenmesi (doküman ismi, revizyon, müdürlük kısaltması, tarih, GİZLİLİK DERECESESİ) unutulmamalıdır. YTET dokümanı için GİZLİLİK DERECESESİ projenin değil, söz konusu dokümanın gizlilik derecesi olarak anlaşılmalıdır. YTET dokümanı kapsamındaki yazılıma ait planlamayı içeren Yazılım Geliştirme Planı’nda (YGP) belirlenmiş olan gizlilik seviyesine uyulmalıdır.
3. Doküman her yayınlanışında, UGTL-DK-03 (Ek-2) iş talimatına uygun olarak, şablonun 2. sayfasında, ilgili revizyon için dokümanı hazırlayan/güncelleyen, gözden geçiren ve onaylayan tüm sorumlu kişilerin isimleri bulunmalıdır. Dokümanın yeni bir revizyonu hazırlanıyor ve bu yeni revizyonu hazırlayan kişi önceki revizyonu hazırlayan kişiden farklı ise her ikisinin de isimleri bu sayfada bulunmalıdır.
4. Dokümanın her yayınlanışında değişiklik kayıtları sayfasına, o yayınlanışta (revizyonda) yapılan değişiklikler özet olarak yayınlanış nedeniyle birlikte yazılmalıdır.
5. İçindekiler (“Table of contents”) bölümü doküman tamamlandığında güncellenmelidir.
6. Dokümanın ilk yayınlanışında altlık (“footer”) bölümünde yer alan *tarih alanı* ve *antetteki tarih alanı*; daha sonraki her yayınlanışında ise altlık (“footer”) bölümünde yer alan *tarih alanı* ve antetteki *değişiklik tarihi alanı* güncellenmelidir.
7. Dokümanın Dokümantasyon Merkezi’ne ilk aktarılışında, ilk sayfadaki antette, elektronik ortamdaki doküman aktarım işlemine uygun olarak dokümanı hazırlayan, kontrol eden ve onaylayan kişilerin isimleri olmalıdır. (Bkz. UGTL-DK-03 (Ek-2) iş talimatı)
8. Antet altında yer alan “Sistemdeki güncel revizyon kodu ve tarihi taşıyorsa geçerlidir.- It is valid only if it has the current revision code and date in the system.” ifadesi ve dokümanın üretildiği “Şablon-Template: <...>” bilgisi hiçbir şekilde silinemez, değiştirilemez. Antet bilgilerinin doldurulması ile ilgili bilgiler “UGTL-DK-01 Doküman Ortak Özellikleri” iş talimatında yer alır.
9. Dokümanın DOORS gereksinim yönetim aracı kullanılarak hazırlandığı durumlarda, “4. Test Tanımları” bölümü hazırlanırken “UGTL-SG-96 DOORS Test Yönetimi” iş talimatının kullanılması önerilmektedir.

HAZIRLAYAN(LAR) / GÜNCELLEYEN(LER) <sup>1</sup>	
Ad SOYAD	Projedeki Görevi

GÖZDEN GEÇİRME EKİBİ <sup>1</sup>	
Ad SOYAD	Projedeki Görevi

ONAY <sup>1</sup>	
Ad SOYAD	Görevi

<sup>1</sup> UGTL-DK-03 Ek-2 Doküman Kodları (Sayfa Kodu) Listesi'ne uygun olarak belirlenir.

## DEĞİŞİKLİK KAYITLARI

\*E - EKLENDİ D - DEĞİŞTİ S - SİLİNDİ

Sıra No	Rev. Kodu	Tarih / Değişiklik Tarihi	Değişiklik Kararı	* E D S	Değişikliğin Yeri	Açıklama
1.	AA	GG/AA/YY		-	-	İlk yayın
2.						
3.						
4.						
5.						
6.						
7.						
8.						
9.						
10.						
11.						
12.						
13.						
14.						
15.						
16.						
17.						
18.						
19.						
20.						
21.						
22.						

## İÇİNDEKİLER

<b>1. KAPSAM</b>	<b>7</b>
1.1. KİMLİK	7
1.2. SİSTEME GENEL BAKIŞ	8
1.3. DOKÜMANA GENEL BAKIŞ	8
<b>2. REFERANS DOKÜMANLAR</b>	<b>8</b>
<b>3. TEST HAZIRLIKLARI</b>	<b>9</b>
3.1. DONANIM HAZIRLIĞI	10
3.2. YAZILIM HAZIRLIĞI	11
3.3. DİĞER TEST ÖNCESİ HAZIRLIKLAR	12
<b>4. TEST TANIMLARI</b>	<b>12</b>
4.x. (TEST GRUBUNUN PROJEYE ÖZGÜ İSMİ)	14
4.x.y. (TESTİN PROJEYE ÖZGÜ İSMİ)	14
<b>5. GEREKSİNİMLERİN İZLENEBİLİRLİĞİ</b>	<b>20</b>
<b>6. NOTLAR</b>	<b>22</b>
<b>7. EKLER</b>	<b>22</b>

## 1. KAPSAM

J-STD-016 Standardı STD Şablonu Metni:

### 1. Kapsam

Bu madde aşağıdaki alt bölümlere bölünmelidir.

1.1. Kimlik. Bu alt bölüm bu dokümanın ait olduğu sistemin ve yazılımın kimliğini (kimlik numarası, sistem içinde geçen ismi, kısaltması, versiyon ve sürüm numaraları gibi bilgilerden uygun olanları) tam olarak içerecektir.

1.2. Sisteme genel bakış. Bu alt bölüm bu dokümanın ait olduğu sistem ve yazılımın amacını açıkça ifade edecektir. Sistem ve yazılımın genel tabiatını tanımlayacak; sistem geliştirme, işletme ve bakım geçmişini özetleyecek; proje sponsorunu, tedarikçisini, kullanıcılarını, geliştiricisini ve bakım organizasyonunu belirtecek; hali hazırda ve planlanan işletme yerini belirtecek ve diğer ilgili dokümanları listeleyecektir.

1.3. Dokümana genel bakış. Bu alt bölüm bu dokümanın amacını ve içeriğini özetleyecek ve dokümanın kullanımı ile ilgili güvenlik ve gizlilik hususlarını belirtecektir.

### 1.1. KİMLİK

#### Kılavuz:

Kimlik bölümünde, J-STD-016'nın verilmesini önerdiği yazılımlar için versiyon ve sürüm numaraları, YTET'in birden fazla yazılım versiyonuna karşılık gelecek şekilde hazırlanması mümkün olduğundan girilmemelidir. Yazılım için kimlik numarası olarak parça numarası kullanılmalıdır.

Gerektiği durumlarda, YTET dokümanında yazılımın “majör” versiyon numarasından bahsedilebilir. YGÖ dokümanındaki gereksinimlere öncelik verilmiş ve yazılımların “majör” versiyon numaraları ile ilişkilendirilmişse yazılım için versiyon numarası verilebilir.

Eğer YTET gereksinim yönetim aracına girilmiş ise araç tarafından verilen temel numarası ve adresi bu bölümde verilecektir.

#### [Örnek]

Proje No	: 9999-9
Sistem/Altsistem İsmi	: AAAA Sistemi/Altsistemi
Yazılım İsmi	: XXX Yazılımı
Yazılım Kısaltması	: YKB-XXX
DOORS Temel Numarası	: 0.1
DOORS Adresi	: /9999/AAAA/XXX/XXX-YTET

#### [Örnek Sonu]

## 1.2. SİSTEME GENEL BAKIŞ

### Kılavuz:

Sisteme genel bakış bölümünde, sistem/alt sistem ve yazılım hakkında özet bilgi verilerek, dokümanın kendi içinde bütünlüğü, anlaşılabilirliği sağlanmalıdır. Bu bölümde ilgili YKB YGÖ dokümanında yer alan bilgi aynen kullanılabilir.

## 1.3. DOKÜMANA GENEL BAKIŞ

### Kılavuz:

Bu bölümde; dokümanın içeriğı, iç organizasyonu, ekleri ile ilgili özet bilgi verilmelidir.

Yeterlilik testleri kapsamında olup beyaz kutu seviyesinde testlerle doğrulanacak gereksinimler varsa, beyaz kutu seviyesindeki testlere ilişkin test hazırlıkları ve test tanımlarının dokümanın ekinde verildiğı burada belirtilebilir.

Bilgisayar destekli gereksinim yönetimi araçlarının çıktıları doğrudan ya da düzenlenerek YTET'e yerleştirilebilir. Bu durumda, oluşturulma yöntemi bu bölümde açıklanmalıdır.

Dokümanın anlaşılmasını kolaylaştırmaya yönelik olarak dokümanda kullanılan terminolojiler varsa burada bunlardan bahsedilmelidir.

### [Örnek]

Bu doküman, DOORS gereksinim yönetim aracı kullanılarak hazırlanmıştır. Her test tanımı için, YTET ile başlayan ve sayı ile devam eden numaralar verilmiştir (örnek: YTET-1).

JTD (Jiroskop Test Düzeneğı) Kullanıcı Yazılımı, doküman boyunca "Kullanıcı Yazılımı" olarak kullanılmıştır.

### [Örnek Sonu]

## 2. REFERANS DOKÜMANLAR

J-STD-016 Standardı STD Şablonu Metni:

### 2. Referans dokümanlar

Bu bölümde, referans verilen tüm dokümanların listesi; numaraları, isimleri, revizyonu, tarih ve kaynağı belirtilerek verilecektir.

### Kılavuz:

Aşağıdaki biçimde bir tablo ile referans dokümanların listesi verilmeli, doküman içinde bu referanslara atıfta bulunulduğunda "No" alanı köşeli parantez içinde belirtilerek bağlantının kurulması sağlanmalıdır.

İlgili yazılımın YGÖ dokümanı ve YATT dokümanı mutlaka listeye eklenmelidir. Varsa YTET dokümanı içinde atıfta bulunulan diğer referanslar da listeye eklenmelidir.

"Doküman No" alanında dokümanın varsa parça numarası veya ayırt edici kimlik numarası verilmelidir. "Yeri" alanında; doküman Dokümantasyon Merkezi'nde ise bu belirtilmeli, proje dizininde veya gereksinim yönetim aracında ise dosyanın tam adresi (temel numarası ile birlikte) verilmelidir. Yeri Dokümantasyon olarak belirtilen referans dokümanın gizlilik derecesi Özel ve daha üst gizlilik derecesinde ise ve bu nedenle Dokümantasyon üzerinden erişim için özel izin gerekiyorsa,



dokümanın proje dizinindeki veya gereksinim yönetim aracındaki yeri de mutlaka belirtilmelidir. Özellikle tarihi belirtilmesi gereken bir doküman ise Rev bilgisi yanına Tarih bilgisi de eklenebilir.

[Örnek]

No	Doküman İsmi	Doküman No	Rev	Yeri
1	XXX Yazılımı Yazılım Gereksinim Özellikleri Dokümanı	YZ-XXXX-0002 (040)	AA	<DOORS Dizini> 0.1 numaralı temel <Proje Dizini>
2	YYY Projesi Yazılım Test Planı			Dokümantasyon <Proje Dizini>
3	YYY Projesi Yazılım Arayüz Tasarım Tanımı Dokümanı			Dokümantasyon <Proje Dizini>

[Örnek Sonu]

### 3. TEST HAZIRLIKLARI

J-STD-016 Standardı STD Şablonu Metni:

#### 3. Test hazırlıkları

Bu bölüm aşağıdaki alt başlıklara ayrılacaktır. Gerekliyse, “UYARI” ve “DİKKAT” sözcükleri ile işaretlenmiş emniyet önlemleri ile güvenlik ve gizliliği korumaya yönelik hususlar da bu bölümde kapsanacaktır.

##### 3.x (Testin projeye özgü ismi)

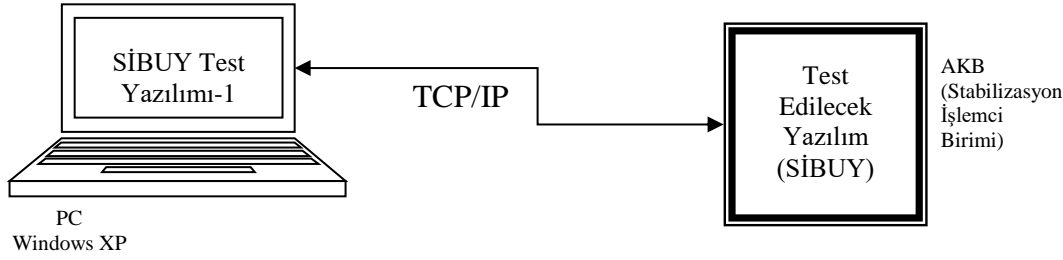
Bu bölüm testin projeye özgü ismini belirtecek, kısa bir tanımını verecek ve aşağıdaki alt bölümlere ayrılacaktır. Önceden tanımlanmış diğer bir test için verilmiş bilginin tekrarı gerektiğinde, burada yinelenmeyecek, ilgili teste referans verilecektir.

#### Kılavuz:

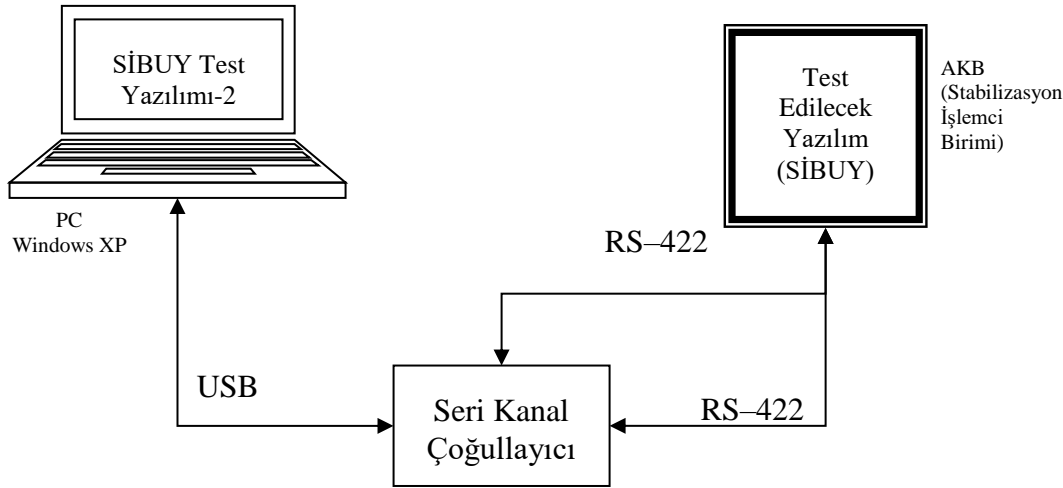
Standart şablon metni test hazırlıklarının test gruplarına göre alt başlıklar halinde gruplandırılmasını önermektedir. Ancak test grupları için yazılım ve donanım test hazırlıklarının ayrı ayrı verilmesi tekrarlara yol açtığından “3.x (Test grubunun projeye özgü ismi)” başlığı iptal edilmiştir. Yazılım ve donanım hazırlıkları gruplandırılmadan verilecektir.

Testlerde kullanılan donanım ve yazılımlardan oluşan farklı konfigürasyonların bu bölümde şemalarla verilmesi, test hazırlıklarının ve test tanımlarının anlaşılabilirliğini arttırmaktadır. Test konfigürasyon şemaları isimlendirilerek burada verilir.

[Örnek]



Test Konfigürasyonu – 1



Test Konfigürasyonu – 2

[Örnek Sonu]

### 3.1. DONANIM HAZIRLIĞI

J-STD-016 Standardı STD Şablonu Metni:

#### 3.x.1. Donanım hazırlığı

Bu bölümde test sırasında kullanılacak donanımı teste hazırlamak için gerekli işlemler tanımlanacaktır. Bu işlemlerle ilgili yayınlanmış işletme el kitapları varsa referans verilebilir. Aşağıda belirtilen türde bilgiler uygun olduğunda verilecektir:

- İsmi ve uygun olduğunda numarası ile tanımlanmış olarak kullanılacak belirli donanım
- Donanım bağlantısı için gerekli anahtar konumları (switch-settings) ve kablolama
- Donanımı, kontrol ve veri bağlantılarını gösteren şemalar
- Donanımı hazır hale getirmek için izlenecek adımlar

#### Kılavuz:

Standart şablon metninde 3.x.1 olarak belirtilen bu başlık 3.1 olarak düzenlenmiştir. Donanım hazırlıkları bütün testler için ortak olarak bu başlık altında belirtilecektir.

Kullanılacak donanımların testler açısından önemli teknik özellikleri, testlerdeki kullanım amaçları, parça numarası ve ayırt edici özellikleri (üretici firması, modeli vb.) belirtilmelidir. Parça numarası

verilen donanımlar için malzeme sayfası da olduğundan, tüm özellikler yerine testlerde kullanılan özelliklerden vurgulanmak istenenler verilecektir.

Bir prototip birimi (kasa ve kartlardan oluşan bir birim gibi) test konfigürasyonunda kullanılıyor ise her donanım parçasının tek tek açıklanması yerine, ilgili prototip birimine referans verilerek tablo halinde malzeme listesi verilebilir.

[Örnek]

### 3.1 DONANIM HAZIRLIĞI

1. **Stabilizasyon İşlemci Birimi (AB-XXXX-00YY):** Test edilecek SİBUY'un üzerinde koşacağı, ASELSAN'da üretilen 350 MHZ hızında AKB8245 kartıdır.
2. **Atış Kontrol Bilgisayarı (1234-XXXX-YYYY):** Üzerinde Stabilizasyon İşlemci Birimi montajlı ve kablajı yapılmış VME kasa ve diğer kartlardan oluşmaktadır. Kart listesi ve ayrıntılı bilgi için dokümanlarına bakınız.
3. **Rocketport seri kanal çoğullayıcı kartı ve 8 kanallı arayüz kutusu (S-XXYYY).** SİBUY'un, Test Yazılımı ile RS422 protokolünde iletişimini sağlayan çoğullayıcıdır. Rocketport seri kanal çoğullayıcı PCI yuvasından bilgisayara takılmakta ve arayüz kutusu üzerinden PC'ye RS422 ya da RS232 olarak ayarlanabilen 8 adet seri kanal eklemektedir.
4. **PC (S-XXYYY):** Windows XP işletim sistemine sahip üzerinde SİBUY Test Yazılımının koştugu PC'dir.
5. **Kablaj:** Test düzeneğinde kullanılan kablaj özellikleri Ek-C'de verilmektedir.

[Örnek Sonu]

### 3.2. YAZILIM HAZIRLIĞI

J-STD-016 Standardı STD Şablonu Metni:

#### 3.x.2. Yazılım hazırlığı

Bu bölüm, test edilecek öge(ler)i ("item(s)") ve ilişkili tüm yazılımları (verileri kapsayarak) teste hazırlamak için gerekli işlemleri tanımlayacaktır. Bu işlemler için yayınlanmış yazılım el kitaplarına da referans verilebilir. Bu bölümde aşağıda belirtilen türde bilgiler uygun olduğunda verilecektir:

- a. Testte kullanılacak belirli yazılım
- b. Test edilecek yazılım öge(ler)inin bulunduğu saklama ortamı (manyetik teyp, disket v.b.)
- c. Test sırasında kullanılacak diğer ilişkili yazılım(lar)ın bulunduğu saklama ortamı (simülörler, test sürücüler, veri tabanları, v.b.)
- d. Yazılımı yüklemek için, gerekli sıralamayı da içeren komutlar
- e. Birden fazla "test case"e ortak, yazılımı başlangıç durumuna getirme komutları

#### Kılavuz:

Standart şablon metninde 3.x.2 olarak belirtilen bu başlık 3.2 olarak düzenlenmiştir. Yazılım hazırlıkları bütün testler için ortak olarak bu başlık altında belirtilecektir.

Kullanılacak yazılımların testler açısından önemli teknik özellikleri, testlerdeki kullanım amaçları, parça numarası ve ayırt edici özellikleri (üretici firması, versiyonu vb.) belirtilmelidir. Parça

numarası verilen yazılımlar için malzeme sayfası da olduğundan, tüm özellikler yerine testlerde kullanılan özelliklerden vurgulanmak istenenler verilecektir.

[Örnek]

### 3.2 YAZILIM HAZIRLIĞI

1. **SİBUY Test Yazılımı (JY-XXXX-0000)** : SİBUY'un, SİBUY'a gönderdiği mesajları benzeterek gönderen Microsoft Visual C++ ortamında geliştirilmiş olan yazılımdır. Bu parça altında dokümente edilmiş iki farklı test yazılımı bulunmaktadır.
  - a. **(SİBUY Test Yazılımı-1)** Yazılım Ethernet üzerinden SİBUY ile haberleşme özelliğine sahiptir. PC üzerinde kořmaktadır.
  - b. **(SİBUY Test Yazılımı-2)** Yazılım ve RS422 SİBUY ile haberleşme özelliğine sahiptir. PC üzerinde kořmaktadır.
2. **Serial Monitor (TG-0000-YYYY)**: SİBUY Test Yazılımının haberleşmesinin doğrulanması için kullanılacak olan analiz yazılımıdır. PC üzerinde kořmaktadır.

[Örnek Sonu]

### 3.3. DİĞER TEST ÖNCESİ HAZIRLIKLAR

J-STD-016 Standardı STD Şablonu Metni:

#### 3.x.3. Diğer test öncesi hazırlıklar

Bu bölüm testi gerçekleřtirmek için gerekli olan test öncesi personel tarafından yerine getirilecek hareketleri, hazırlıkları veya işlemleri tanımlayacaktır.

#### Kılavuz:

Bu bölümde test öncesi yapılması gereken diğer hazırlıklar, işlemler ya da testi yapacak kiři tarafından yerine getirilecek hareketler belirtilecektir. (Örneğın, “ “Reset” düğmesini basılı tutarken XXX etiketli anahtarı “açık” konumuna getiriniz.” gibi).

[Örnek]

Testlere başlamadan önce sırasıyla;

- Diskler üzerindeki bir önceki konfigürasyona ait imzalar ve kayıt kartındaki kayıtların silinebilmesi için çalışılan konfigürasyonda kayıtları silen “master” yazılımı çalıştırılır.
- Kayıt kartına seri kanalından bağlanılarak “rx.ini” isimli dosya silinerek grup yapılarının silinmesi sağlanır.
- Çalışılacak konfigürasyona ait diskler yerlerine yerleştirilir ve o konfigürasyona ait “master” yazılımı çalıştırılarak grup ve kayıtlar oluşturulur.

[Örnek Sonu]

## 4. TEST TANIMLARI

J-STD-016 Standardı STD Şablonu Metni:

### 4. Test tanımları

Bu bölüm aşağıdaki alt başlıklara ayrılacaktır. Gerekliyse, “UYARI” ve “DİKKAT” sözcükleri ile işaretlenmiş emniyet önlemleri ile güvenlik ve gizliliği korumaya yönelik hususlar da bu bölümde kapsanacaktır.

#### Kılavuz:

Dokümanda kullanılacak test gruplarının tanımları ve gruplardaki test sayıları tablo şeklinde verilebilir.

#### [Örnek]

Taktik Kayıt (TK) YKB’sinin doğrulanması amacıyla hazırlanan testler aşağıdaki tabloda belirtilen 2 test grubu altında toplanmıştır.

**Tablo 4-1 Test Grupları**

	<i>Test Grubu İsmi</i>	<i>Test Sayısı</i>
1	<i>Yakalama Konfigürasyonu Testleri</i>	<i>10</i>
2	<i>İnceleme/Mmd Konfigürasyonu Testleri</i>	<i>20</i>
	<b>Toplam</b>	<b>30</b>

#### [Örnek Sonu]

YTET dokümanının DOORS gereksinim yönetim aracı kullanılarak hazırlandığı durumlarda, “4. Test Tanımları” bölümü, “UGTL-SG-96 DOORS Test Yönetimi” iş talimatında tarif edildiği şekilde hazırlanarak bu dokümanın eki olarak verilebilir.

#### [Örnek]

Taktik Kayıt (TK) YKB’sinin doğrulanması amacıyla hazırlanan test tanımları DOORS ortamında saklanmaktadır. DOORS ortamından PDF çıktı alınarak bu dokümanın eki haline getirilmiştir. (Bknz. Ek - Test Tanımları.pdf)

#### [Örnek Sonu]

Dokümanda, test edilmek üzere teslim alınan yazılım sürümünün, test edilebilir olgunluğa erişip erişmediğinin kontrol edilebilmesi için gerçekleştirilecek test tanımlarının listesi verilebilir. Verildiği durumda, test edilebilir olgunluğa erişme kriteri de bu listenin başında belirtilmelidir.

#### [Örnek]

Teslim alınan Taktik Kayıt (TK) YKB’si sürümünün test edilebilir olgunluğa eriştiğinin kontrolü öncelikle aşağıda listelenen testler gerçekleştirilerek yapılacaktır. Bu testlerde önem derecesi 1 veya 2 olan hata bulunması durumunda yazılım sürümü için testlere başlanmayacak, YTER hazırlanmayacak, yazılım testleri duyuru listesine durum hakkında bilgi verilerek, yazılım sorumlusundan hataların düzeltildiği yeni sürüm istenecektir. Bu önem derecelerine sahip hata olmaması durumunda ise testlere başlanacaktır.

- YTET\_1 Açılış Testi
- YTET\_15 Versiyon Numarası Kontrol Testi

3. YTET\_5 Taktik Kayıt Parametreleri Okuma Testi
4. YTET\_6 Taktik Kayıt Parametreleri Yazma Testi
5. YTET\_30 Kapanış Testi

[Örnek Sonu]

Dokümanda, test edilmek üzere teslim alınan her yazılım sürümünün regresyon testinde, mutlaka gerçekleştirilecek test tanımlarının listesi verilebilir.

[Örnek]

Teslim alınan her Taktik Kayıt (TK) YKB'si sürümünde aşağıda listelenen testler mutlaka gerçekleştirilecektir.

1. YTET\_10 Taktik Kayıt Parametreleri Gönderme Testi
2. YTET\_11 Taktik Kayıt Parametreleri Alma Testi

[Örnek Sonu]

#### 4.x. (TEST GRUBUNUN PROJEYE ÖZGÜ İSMİ)

J-STD-016 Standardı STD Şablonu Metni:

##### 4.x. (Testin projeye özgü ismi)

Bu bölüm testi projeye özgü ismi ile belirtecek ve aşağıdaki alt bölümlere ayrılacaktır. Önceden tanımlanmış diğer bir test için verilmiş bilginin tekrarı gerektiğinde, burada yinelenmeyecek, ilgili teste referans verilecektir.

#### 4.x.y. (TESTİN PROJEYE ÖZGÜ İSMİ)

J-STD-016 Standardı STD Şablonu Metni:

##### 4.x.y. ("Test case"ın projeye özgü ismi)

Bu bölüm "test case"i projeye özgü ismi ile belirtecek, amacını ve kısa tanımını verecektir. Aşağıda test case'in ayrıntılı tanımı verilecektir.

##### 4.x.y.1. Adreslenen gereksinimler

Bu bölüm, "test case" tarafından adreslenen yazılım ögesi ya da sistem gereksinimlerini belirtecektir. (Diğer bir seçenek olarak bu bilgiler 5.a bölümünde de verilebilir.)

##### 4.x.y.2. Önkoşullar

Bu bölüm, adı geçen "test case"ın uygulanması öncesinde sağlanmış olması gereken önkoşullar varsa aşağıdaki kapsamda belirtecektir:

a. Donanım ve yazılım konfigürasyonu,

b. Teste başlamadan önce belirli bir değere ayarlanması ya da sıfırlanması gereken "flag"ler, başlangıç "breakpoint"ları, "pointer"lar, kontrol parametreleri veya verilerin ilk değerleri,

c. "Test case"i koşturmak için gereken önceden belirlenmiş donanım koşulları veya elektriksel durumları,

- d. Zamanlama ölçümlemlerini yapacak cihazların başlangıç ayarları,
- e. Benzetilen ortamın (simulated environment) düzenlenmesi,
- f. “Test case”e özel diğer koşullar.

#### 4.x.y.3. Test girdileri

“Test case”in gerçekleştirilmesi için gereken test girdileri (varsa) aşağıdaki kapsamda belirtilecektir:

- a. Her bir test girdisinin adı, amacı ve tanımı (değer aralığı, doğruluk v.b. gibi),
- b. Test girdisinin kaynağı ve test girdisinin seçilmesinde kullanılan yöntem,
- c. Girdinin gerçek mi, yoksa benzetilmiş mi (simulated) olduğu,
- d. Test girdisinin (sisteme/düzeneye) uygulanma zamanı ve sırası,
- e. Girdi verinin şu durumlar için nasıl kullanılacağı:
  - 1) En az ve makul sayıda veri türü ve değeri ile öge(ler)in testi,
  - 2) Aşırı yükleme (overload), doymunluk (saturation) ya da diğer “en kötü” etkileri oluşturacak geçerli veri tipleri ve değeri aralığında öge(ler)in çalıştırılması,
  - 3) Geçersiz veri tipleri ve değeri ile öge(ler)in çalıştırılarak bozuk girdilerin uygun olarak işleme alındığının testi.
  - 4) Gerekiyorsa test tekrarına izin verme.

#### 4.x.y.4. Beklenen test sonuçları

Bu bölüm “test case” için beklenen tüm test sonuçlarını belirtecektir. Uygun olduğunda ara ve son test sonuçları birlikte verilecektir.

#### 4.x.y.5. Sonuçları değerlendirme ölçütleri

Burada “test case”in ara ve son test sonuçlarını değerlendirmede kullanılacak olan ölçütler belirtilecektir. Her test sonucu için uygun olduğunda aşağıdaki bilgiler verilecektir:

- a. Bir çıktının değişebileceği ve hala kabul edilebilir olabileceği aralık veya doğruluk
- b. Kabul edilebilir bir test sonucu oluşturmak için gerekli en az girdi ve çıktı kombinasyonunun ya da seçeneğinin sayısı
- c. İzin verilen en fazla/en az test süresi (zaman ya da olay adeti olarak belirtilebilir)
- d. Belirli bir dönemde oluşabilecek en fazla kesme (interrupt), durma ya da diğer sistem kesintilerinin sayısı
- e. İşlem hatalarının kabul edilebilir önem derecesi
- f. Sonucun yetersiz olduğu ve yeniden test yapılacağı koşullar
- g. Çıktının, girdi test verisinde, test veri tabanında/veri dosyalarında veya test yönergesinde, bozukluklar olduğu şeklinde yorumlandığı koşullar
- h. Testin kontrolü, durumu ve sonuçları için ve sonraki “test case”e hazır olduğuna dair kabul edilebilir göstergeler (yardımcı test yazılımının çıktısı olabilir)
- i. Yukarıda belirtilmeyen diğer ek ölçütler.

#### 4.x.y.6. Test yönergesi

Bu bölüm “test case” için uygulanacak test yönergesini tanımlayacaktır. Test yönergesi gerçekleştirileceği sırayla listelenmiş, tek tek numaralandırılmış bir dizi adım olarak tanımlanacaktır. Dokümanı güncelleme kolaylığı açısından, test yönergesinin ek bir doküman olarak hazırlanması ve buradan referans verilmesi de mümkündür. Test yönergesinin hangi ayrıntıda hazırlanacağı test edilen yazılım türüne bağlı olarak değişebilir. Bazı yazılımlar için her bir tuşa basma işlemi ayrı bir test yönerge adımı olabilirken, çoğu yazılımda, her adım mantıksal ilişkili bir dizi tuş basımı veya diğer hareketi içerebilir. Bu nedenle uygun ayrıntı düzeyi, beklenen sonuçları net olarak tanımlamakta ve gerçek sonuçlarla karşılaştırmada yararlı olan düzeydir. Her bir test yönergesinde mümkün olduğunca aşağıdakiler verilecektir:

a. Her adım için gereken test operatörü hareketleri ve donanım işlemleri (komutları da uygunsa içererek):

- 1) “Test case”i başlatmak ve test girdisini uygulamak
- 2) Test koşullarını gözden geçirmek
- 3) Test sonuçlarının ara değerlendirmesini yapmak
- 4) Veriyi kaydetmek
- 5) Test işlemi durdurmak ya da kesmek
- 6) İhtiyaç duyuluyorsa veri dökümlerini ya da diğer yardımcı verileri istemek
- 7) Veritabanı/veri dosyalarında değişiklik yapmak
- 8) Sonuç başarısız ise “test case”i tekrarlamak
- 9) “Test case” gerektiriyorsa öteki modları denemek
- 10) “Test case”i sonlandırmak

b. Her adım için beklenen sonuç ve değerlendirme ölçütü,

c. “Test case” birden çok gereksinim ile ilişkiliyse, yönergenin her bir adımının hangi gereksinimle ilişkili olduğunun belirtilmesi (istenirse bu bilgi 5. bölümde de verilebilir),

d. Programın hata vermesi ya da durması durumunda izlenecek yollar. Örneğin:

- 1) Referans amacıyla göstergelerden kritik verilerin kaydedilmesi,
- 2) Zamana duyarlı test destek yazılımlarının ya da test aletlerinin durdurulması ya da duraklatılması,
- 3) Test sonuçlarını içeren sistem ve operatör kayıtlarının toplanması.

e. Mümkünse, test sonuçlarını indirgeme ve çözümlemede kullanılacak aşağıdakileri yerine getirmek için işlemler:

- 1) Herhangi bir çıktının üretilip üretilmediğinin belirlenmesi
- 2) “Test case” tarafından üretilen verinin ortamının ve yerinin belirtilmesi
- 3) Test işleminin sürdürülmesinin temeli olarak çıktının değerlendirilmesi
- 4) Test çıktısının gereken çıktıya karşı değerlendirilmesi

#### 4.x.y.7. Varsayımlar ve kısıtlar



Bu bölüm; zamanlama, arayüzler, donanım, personel ve veri tabanı/veri dosyası sınırlamaları gibi; sistem veya test koşulları yüzünden “test case”in tanımını etkileyen, yapılmış varsayımları ve kısıtları ya da sınırlamaları, varsa belirtecektir. Eğer feragatler ve istisnalar için belirlenen sınırlar ve parametreler onaylanmışsa, bunlar burada belirtilecek ve “test case” üzerindeki etkileri de bu bölümde verilecektir.

#### Kılavuz:

Standart şablon metninde “test case” ifadesi, belirli bir gereksinimi doğrulamak için ön koşullar, test girdileri, test yönergesi ve beklenen test sonuçları gibi belirli bir teste ilişkin bilgilerin tümünün dokümantasyonunu ifade etmektedir. “Test case” test çerçevesi/ kutusu/kasası gibi bir anlamda kullanılmış, ancak yerleşmiş bir Türkçe karşılığı bulunmadığından şablon metninde aynen bırakılmıştır. Türkçe şablon başlığında ise “test case” için sadece “test” sözcüğü kullanılmıştır.

Beklenmeyen durum ya da veri girişi halinde yazılımın tepkisini sınavan test tanımları yapılmış olmalıdır. Çeşitli işlevlerin birlikte çalışmasını gerektiren durumlar varsa ilgili testler tanımlanmış olmalıdır.

Test tanımlarının birbirlerine göre gerçekleştirilme öncelikleri varsa bu öncelikler de belirlenmelidir. Öncelikler, DOORS aracı üzerinde tutulan YTET dosyasına kullanımı isteğe bağlı nitelik olarak girilebilir. Test tanımlarının öncelikleri buradan referans verilerek bir tablo halinde Ekler bölümünde verilebilir.

Bu bölüm, DOORS gibi Gereksinim Yönetimi araçları üzerinden yazılım gereksinimleri ve test tanımları arasında izlenebilirlik kurmayı kolaylaştırabilmek amacıyla standart metinde belirtildiğinden farklı olarak alt başlıklara ayrılmayacak, her bir test tanımı aşağıda belirtilen formatta hazırlanacaktır:

#### **4.x.y. (TESTİN PROJEYE ÖZGÜ İSMİ)**

**Adreslenen Gereksinimler:** Bknz. J-STD-016 Standardı STD Şablonu Metni 4.x.y.1

**Testin Amacı:** Testin yapılma amacı belirtilecektir. Adreslenen gereksinimlerden özet bir test amacı oluşturulmalıdır.

**Önkoşullar:** Bknz. J-STD-016 Standardı STD Şablonu Metni 4.x.y.2. Testin 3. bölümünde tanımlanan test konfigürasyonlarından hangisinde yapılacağı bu bölümde yazılacaktır. 3. bölümde belirtilen konfigürasyonla ilgili ön hazırlıkların burada tekrarlanmasına gerek yoktur.

**Test Girdileri:** Bknz. J-STD-016 Standardı STD Şablonu Metni 4.x.y.3. Test girdileri başka bir dokümandan alınıyor ise bu bölümde belirtilecektir. (Örneğin YATT) Mümkün olduğunda, test girdileri hem ortalama değerleri hem de sınır değerleri (asgari ve azami değerler) içerecek şekilde seçilmiş olmalıdır. Mümkün olduğunda, test girdileri zorlayıcı değerleri (sınır dışı değerler gibi) içerecek şekilde seçilmiş olmalıdır.

**Test Yönergesi:** Bknz. J-STD-016 Standardı STD Şablonu Metni 4.x.y.6

**Beklenen Test Sonuçları:** Bknz. J-STD-016 Standardı STD Şablonu Metni 4.x.y.4

**Test Sonuçlarını Değerlendirme Ölçütleri:** Bknz. J-STD-016 Standardı STD Şablonu Metni 4.x.y.5 Standart metin bu bilginin ayrı bir başlık altında yazılmasını istemekle birlikte, bu tür bilgilerin genellikle “Test Yönergesi” ve “Beklenen Test Sonuçları” başlıkları altında verilmesi nedeniyle bu başlık altında yazılmasına gerek duyulmayabilir.

**Test Varsayım ve Kısıtları:** Bknz. J-STD-016 Standardı STD Şablonu Metni 4.x.y.7

Test tanımlarına değışmeyen eşsiz kimlik numaraları atanmalıdır. İzlenebilirliğin araçlarla (DOORS vb.) sağlanması durumunda aracın atadığı Kimlik Numarası (ID) kullanılmalıdır.

[Örnek]

ID: KGU-YTET20

#### 4.1.1 SESLİ UYARILAR TESTİ

**Adreslenen Gereksinimler:** 514, 287, 395

**Testin Amacı:** KGÜ Yazılımı'nın:

- Yeni karıştırıcı bilgisi alındığını sesli olarak kullanıcıya bildirdiğinin,
- Komuta Kontrol mesajı alındığını sesli olarak kullanıcıya bildirdiğinin,
- Sesli uyarıların açık/kapalı yapılabilmesi için arayüz sunduğunun

doğrulanması

**Önkoşullar:** Test Şekil 1'de verilen KGÜ Yazılımı Test Konfigürasyonu'nda yapılmalıdır.

**Test Girdileri:** YATT dokümanından aşağıdaki mesajlar temel alınmıştır.

- Karıştırıcı Raporu Mesajı (Denenecek parametreleri yönergedeki tabloda verilmiştir)
- Komuta Kontrol Mesajı

**Test Yönergesi:**

- Karıştırıcı ve komuta kontrol sesli uyarıları açık konumuna getirilecektir.
- KGÜ Test Yazılımı'ndan KGÜ Yazılımı'na 1 elemanlı Karıştırıcı Raporu gönderilecektir.
- Karıştırıcı bilgisi alındığında sesli olarak kullanıcıya bildirildiği gözlenmelidir.*
- KGÜ Test Yazılımı'ndan KGÜ Yazılımı'na Komuta Kontrol mesajı tabloda belirtilen parametre değerleriyle gönderilecektir.
- KGÜ yazılımı Komuta Kontrol mesajını aldığıda sesli olarak kullanıcı uyarmalıdır. Duyulacak sesin niteliği tabloda verilmiştir.*

Komuta Kontrol Mesajının Parametresi	Sesin Niteliği
0	Hata Sesi
10	Normal Ses
20	Normal Ses
90	Normal Ses
100	Normal Ses
110	Hata Sesi

- Karıştırıcı sesli uyarısı kapalı duruma getirilecek ve 2. ve 3. adımlar tekrar edilecektir.

- *Sesli uyarının kapalı yapılması için ilgili arayüzün sunulduğu ve Karıştırıcı bilgisi alındığında sesli olarak kullanıcıya bildirilmediği gözlenmelidir.*
- 5. Komuta kontrol sesli uyarısı kapalı duruma getirilecek ve 2. ve 3. adımlar tekrar edilecektir
- *Sesli uyarının kapalı yapılması için ilgili arayüzün sunulduğu ve Komuta Kontrol mesajı alındığında sesli olarak kullanıcıya bildirilmediği gözlenmelidir.*

**Beklenen Test Sonuçları:** Test yönergesi bölümünde ilgili adımların altında beklenen test sonuçları *italik* olarak verilmiştir.

**Test Sonuçlarını Değerlendirme Ölçütleri:** Yoktur.

**Test Varsayım ve Kısıtları:** Yoktur.

[Örnek Sonu]

[Örnek]

#### 4.14 SİSTEM İMHASI TESTLERİ

...

**ID:** SIBUY\_YTET53

##### 4.14.3 SİSTEM İMHA YETENEĞİNİN AKTİFLİK DURUMUNUN BELİRLENMESİ

**Adreslenen Gereksinimler:** SIBUY\_YGO462, SIBUY\_YGO463

**Testin Amacı:** SİBUY'un BİBUY'dan sistem imha yeteneğinin aktiflik durumunu alarak NPS'ye bildirdiğinin doğrulanması

**Önkoşullar:** Test SİBUY Test Konfigürasyonunda yapılacaktır. SİBUY operasyonel modda açılmış olmalıdır.

**Test Girdileri:** Yoktur.

**Test Yönergesi:**

1. SİBUY Test Yazılımı-1'den sistem imha yeteneğinin aktif olduğu bilgisi SİBUY'a gönderilecektir.

**Beklenen Test Sonuçları:** SİBUY'un, SİBUY Test Yazılımı-1'den aldığı sistem aktiflik durumunu SİBUY Test Yazılımı-2'ye bildirdiği gözlenmelidir.

**Test Sonuçlarını Değerlendirme Ölçütleri:** Yoktur.

**Test Varsayım ve Kısıtları:** Seri kanal üzerinden gönderilmesi gereken mesaj, testlerde kullanılan donanımın kısıdından dolayı senkron haberleşme altyapısı yerine asenkron haberleşme altyapısı kullanılarak gönderilecektir.

[Örnek Sonu]

[Örnek]

#### 4.5 AKTİF SENARYO TESTLERİ

...

ID: MESSIM\_YTET\_20

#### 4.5.7 AKTİF SENARYODA TANIMLI BİR BİRLİĞİN SİLİNMESİ

**Adreslenen Gereksinimler:** 3.2.1.3.16, 3.2.1.3.17, 3.2.1.3.18

**Testin Amacı:** MESSIM yazılımının kullanıcının isteğine bağlı olarak aktif senaryoda tanımlı bir birliği bilgileri, mesajları ve hedefleri ile beraber silebildiğinin doğrulanması.

**Önkoşullar:** Test MESSIM YKB Test Konfigürasyonunda yapılmalıdır. Aktif senaryoda tanımlı en az bir birlik olmalıdır. Bu birliğe ait en az bir hedef ve mesaj bulunmalıdır.

**Test Girdileri:** Yoktur.

#### Test Yönergesi:

1. MESSIM yazılımından aktif senaryoda tanımlı birlikleri listelemesi istenecektir.
2. Gelen listedeki herhangi bir birlik seçilecek ve bu birliğe sil komutu verilecektir.
3. Silme işlemi onaylanacaktır.
4. Silinen birliğe ait mesaj ve hedeflerin varlığı gözlemlenecektir.

**Beklenen Test Sonuçları:** MESSIM Yazılımı, kullanıcı bir birliği silmek istediğinde sistemde birliğe ait mesaj ve birliğin tespit ettiği hedeflerin de silineceğine dair kullanıcıyı uyarmalı, kullanıcı birliğin silinmesini onaylarsa, birlik bilgileri ile beraber sistemde birliğe ait mesaj ve hedefleri de silmeli, sistemde silinen birliğe ait mesaj ve hedef kalmamalıdır.

**Test Sonuçlarını Değerlendirme Ölçütleri:** Yoktur.

**Test Varsayım ve Kısıtları:** Yoktur.

[Örnek Sonu]

## 5. GEREKSİNİMLERİN İZLENEBİLİRLİĞİ

J-STD-016 Standardı STD Şablonu Metni:

### 5. Gereksinimlerin izlenebilirliği

Bu bölüm aşağıdakileri içerecektir:

- a. Bu STD’de bulunan her test case’inden adreslediği sistem veya yazılım ögesi gereksinimlerine izlenebilirlik. Bir test case çok sayıda gereksinimi adresliyorsa, her test yönerge adımları setinden adreslenen gereksinim(ler)e izlenebilirlik. (Alternatif olarak, 4.x.y.1. bölümünde verilebilir.)
- b. Bu STD tarafından kapsanan her sistem veya yazılım ögesi gereksiniminden, adreslenen test case’(ler)ine izlenebilirlik. Yazılım ögesi testi için, yazılım gereksinim özellikleri (SRS) ve ilişkili arayüz gereksinim özelliklerinde (IRS’ler) bulunan her yazılım parçası gereksiniminden izlenebilirlik. Sistem testi için, sistem alt sistem gereksinim özellikleri (SSS) ve ilişkili IRS(ler)den izlenebilirlik. Eğer bir test case çok sayıda gereksinimi adresliyorsa, izlenebilirlik her gereksinimin adreslendiği belirli test yönergesi adımlarını belirtecektir.

### Kılavuz:

YGÖ’de bulunan doğrulama seviyesi YKB olarak belirlenen her bir yazılım gereksiniminin en az bir test tanımı ile kapsandığı izlenebilirlik tabloları oluşturularak kontrol edilmelidir.

YGÖ’de yer alan yazılım gereksinimleri ile doğrudan ilişkilendirilemeyen, ancak yazılımdan kullanıcı tarafından beklenebilecek (örtülü) gereksinimlerin (örneğin hatalı durumlardaki davranışların testi gibi) testine yönelik yazılım test tanımları da yazılım test sorumlusu tarafından tanımlanabilir. Bu tür durumlar oluştuğunda, genel bir “Test Gerçekleştirme” gereksinimi ile izlenebilirlik kurulmalıdır. (Bknz. Örnek) Bu tür test tanımları yapılma ihtiyacının olması bazen yazılım gereksinimlerinin eksik ifade edilmiş olmasından da kaynaklanabilir. Bu durumlarda YGÖ’de değişiklik gerekip gerekmeyeceği değerlendirilmelidir.

Bu bölümde verilecek tablolar için izlenebilirlik araçları çıktıları kullanılabilir. Kullanım kolaylığı açısından Ek olarak da verilebilir.

Eğer test tanımları “UGTL-SG-96 DOORS Test Yönetimi” iş talimatı kullanılarak hazırlanmışsa, bu dokümanın eki olacak olan PDF dokümanı test tanımlarından gereksinimlere izlenebilirliği içereceğinden burada verilmeyebilir.

#### [Örnek]

YKB seviyesinde doğrulanabilen yazılım gereksinimleri ve test tanımları için izlenebilirlik tabloları aşağıda verilmiştir. Beyaz kutu seviyesinde doğrulanabilen yazılım gereksinimleri ve test tanımları izlenebilirliği ise EK-A Beyaz Kutu Test Tanımlarında verilmiştir.

Tablo 5-1 Gereksinimlerden Test Tanımlarına İzlenebilirlik Tablosu

Gereksinimler	Test Tanımları
YGÖ-12	YTET-123
YGÖ-14	YTET-234
YGÖ-15	YTET-345
Test Gerçekleştirme	YTET-346

Tablo 5-2 Test Tanımlarından Gereksinimlere İzlenebilirlik Tablosu

Test Tanımları	Gereksinimler
YTET-123	YGÖ-12
YTET-234	YGÖ-14
YTET-345	YGÖ-15
YTET-346	Test Gerçekleştirme

#### [Örnek Sonu]

#### [Örnek]

Test Tanımlarından Gereksinimlere İzlenebilirlik bilgisi “Ek-Test Tanımları.pdf” dosyasında yer almaktadır.

#### [Örnek Sonu]

## 6. NOTLAR

J-STD-016 Standardı STD Şablonu Metni:

### 6. Notlar

Bu bölüm, bu dokümanın anlaşılmasına yardımcı olacak her türlü genel bilgiyi (örneğin geçmiş bilgi birikimi, gerekçeler, sözlük, açıklamalar) içerecektir. Bu bölüm, tüm kısaltmaların alfabetik listesini, bu dokümandaki anlamlarını ve dokümanı anlamak için ihtiyaç duyulan terim ve tanımları içerecektir.

#### Kılavuz:

Bu doküman içinde kullanılan kısaltmaların listesi alfabetik sıra ile verilir.

#### [Örnek]

Kısaltma Listesi:

TMM	Test Mühendisliği Müdürlüğü
YATT	Yazılım Arayüz Tasarım Tanımı
YGÖ	Yazılım Gereksinim Özellikleri
YKB	Yazılım Konfigürasyon Birimi
YTER	Yazılım Test Raporu
YTET	Yazılım Test Tanımı

#### [Örnek Sonu]

## 7. EKLER

J-STD-016 Standardı STD Şablonu Metni:

### 7. Ekler

Ekler, dokümanın idamesini kolaylaştırmak amacıyla; ayrı olarak derlenmiş bilgilerin (şemalar, sınıflandırılmış bilgi vb.) verilmesi için kullanılabilir. Uygunsa; her eke, dokümanın ana metni içinde, ekteki verinin normal olarak verileceği yerden, referans verilecektir. Ekler kullanım kolaylığı açısından ayrı dokümanlar olarak hazırlanabilir. Ekler (A, B gibi) alfabetik olarak numaralandırılacaktır.

#### Kılavuz:

Bu bölümde dokümanın diğer bölümlerinde verilmesi dokümanın idame ve kullanımını olumsuz etkileyecek büyük boyutlu tablo ve benzeri bilgiler yer alabilir.

Beyaz kutu seviyesinde doğrulanan YKB gereksinimleri için yapılan test tanımları yazılım sorumlusundan sağlandığında ek olarak verilmelidir.

#### [Örnek]

Ek-A Beyaz Kutu Test Tanımları

Tablo 7-1 – Beyaz Kutu Test Tanımları

YTET ID	Beyaz Kutu Test Tanımı	Doğrulanan YGÖ Gerek No
---------	------------------------	-------------------------

YTET ID	Beyaz Kutu Test Tanımı	Doğrulanan YGÖ Gerek No
YTET-BK-XXXX1	AAAA firmasının sağlamış olduğu FCDD-10-73GB disklerinden oluşan kayıt birimi, laboratuvar ortamında kurulacaktır. XXX YKB'si çalıştırılarak, kayıt kartı aracılığıyla bu disklere ulaştıktan sonra verileri kaydedebildiği görülecektir. Aynı şekilde, kayıt edilen veriler diskten geri okunarak disklere doğru şekilde kayıt yapıldığı test edilecektir.	283

[Örnek Sonu]