Türkiye'de DO-178B Uyumlu Yazılım Sertifikasyon Projelerinde Planlama Sürecinde Yaşanan Problemler

M. Umut Pişken , Burak Ata

Savunma Teknolojileri ve Mühendislik A.Ş., Mühendislik ve Sertifikasyon Müdürlüğü, Ankara, Türkiye

{mpisken, bata}@stm.com.tr

Özet. Aviyonik yazılım geliştirme projelerinde dünyada genel kabul görmüş bir standart olan DO-178B standardı, tüm yazılım yaşamdöngüsü aşamaları için amaçlar tanımlamıştır. Ülkemizde milli olarak gerçekleştirilen aviyonik yazılım sayısı günden güne artmakta, her geçen gün sektörden yeni firmalar DO-178B uyumlu yazılım geliştirme projelerine dahil olmaktadır. Projelerin DO-178B uyumunu değerlendirmek için, ilgili ülkelerin havacılık otoriteleri tarafından, "Stages of Involment (SOI)" adı verilen değerlendirmeler projenin planlama, geliştirme ve doğrulama gibi çeşitli aşamalarında yapılmaktadır. Bu değerlendirmeler esnasında, bir yardımcı doküman olan Job Aid dokümanı tarafından tanımlanmış soru listeleri kullanılmaktadır. Bu çalışmada, ülkemizde 2015 yılı içerisinde DO-178B uyumlu çeşitli yazılım geliştirme projeleri kapsamında planlama sürecini değerlendirmek üzere yapılmış olan SOI-1 değerlendirmelerinde ortaya çıkan problemler incelenecek ve problemleri önlemeye yönelik öneriler sunulacaktır.
Anahtar Kelimeler: DO-178B, SOI, Aviyonik Yazılım, Yazılım Planlama Süreci, Alınan Dersler, Emniyet Kritik Yazılımlar

Abstract. DO-178B, which is a de facto standard for developing avionics software systems, defines objectives for the entire phases of software lifecycle. Turkey's avionic industry has shown strong growth over the last decade and as a result of this trend every day new software development companies are involved in the DO-178B compliant software development projects. The certification authorities perform "Stage of Involvement (SOI)" audits at various stages of the project such as planning, development and verification to assess compliance to DO-178B. In this study, the results of SOI-1 audits in 2015 which are performed to assess planning phase of project, are analyzed. Main problem areas in planning phase are determined according to this analyze and recommendations are given to prevent these problems.
Key words: DO-178B, Avionic Software, Software Planning Process, Lessons Learned, Safety Critical Software

1 Giriş

1970'li yılların sonlarına doğru yazılım ağırlıklı ekipmanların havacılıkta yerlerini almaya başlaması beraberinde uçuşa elverişlilik açısından yazılımların nasıl değerlendirilmesi gerektiği sorusunu gündeme getirmiştir. Zaman içerisinde bu tarz ekipmanların değerlendirilmesi esnasında laboratuvar ortamında ya da hava aracı üzerinde yapılan fonksiyonel doğrulamaların yeterli olmadığı, yazılım geliştirme sürecine ilişkin incelemelerin de yapılması gerektiği anlaşılmıştır.

Bu durumun tetiklemesiyle RTCA tarafından bir komite kurulmuş ve bu komite tarafından hazırlanan DO-178, "Software Considerations in Airborne Systems and Equipment Certification" isimli rehber doküman 1982 yılında yayımlanmıştır [1]. Bu doküman DO-178A, DO-178B ve DO-178C olmak üzere üç kez versiyon atlamış olup, en güncel versiyonu DO-178C'dir [1]. Federal Aviation Administration (FAA) tarafından DO-178C AC NO:20-115C ile kabul edilebilir bir uyum yöntemi olarak kabul edilmiş olmasına rağmen yürümekte olan bir çok proje, "Advisory Circular(AC)"'nin yayım tarihinden önce başladığı için şimdilik DO-178B versiyonuna uyum sağlayacak şekilde yürütülmektedir [2].

Yazılım hataları, yazılım geliştirme esnasında ortaya çıkan sistematik hatalardan kaynaklanmaktadırlar [3]. Yazılım hataları, fiziksel parçalarda görülebilen ve kullanıma bağlı yıpranma-aşınma kaynaklı oluşan rastsal hatalardan(Random Failure) farklıdırlar. Diğer emniyet kritik yazılım geliştirme standartları gibi DO-178B standardı da bu durumu göz önüne alarak, sistematik hataları önlemek üzere yazılım geliştirme süreçlerine yönelik belirli gereksinim ve kısıtlamaları tanımlayacak şekilde geliştirilmiştir [3]. Bu yaklaşıma göre, öncelikle sistem emniyet analizleri yapılarak yazılımın içinde çalışacağı sistem açısından emniyet kritiklik seviyesi belirlenmekte, sonrasında ise bu kritiklik seviyesini sağlamak üzere kullanılacak olan standardın istediği yazılım yaşamdöngüsü süreçleri işletilmektedir. Bu yaklaşım, tasarım teminatı yaklaşımı olarak adlandırılmaktadır.[1]. Tasarım teminatı ile gereksinim, tasarım ve geliştirme aşamalarında ortaya çıkabilecek hataların sistem emniyeti açısından kabul edilebilir seviyeye indirildiğini garanti altına almak için uygulanan planlı ve sistematik aktiviteler kastedilmektedir [1]. Tasarım teminatı yaklaşımı, yazılım geliştirme esnasında uygulanacak süreçler ne kadar titiz ve sıkı olursa yazılımda o derece az hata kalacağı varsayımına dayanmaktadır [1]. Bu yaklaşımda, yazılımın kritiklik seviyesi yükseldikçe, uygulanan geliştirme ve doğrulama aktiviteleri(gözden geçirme, analiz, test gibi) nicelik ve nitelik açısından artmaktadır. DO-178B standardı da tasarım teminatı yaklaşımını temel alan bir standarttır. Örnek vermek gerekirse, DO-178B standardında tasarım teminatı seviyesi DAL A olarak belirlenmiş olan yazılımlar için, kod yapısal kapsama analizi yapılırken "Modified condition/decision coverage" (MC/DC) sağlanması gerekirken, tasarım teminatı seviyesi DAL B olarak belirlenen yazılımlarda "Decision Coverage" sağlanması yeterlidir.

DO-178B standardında, standardın son bölümünde verilmiş olan A-1'den A-10'a kadar olan tablolarda yazılım kritiklik seviyelerine göre karşılanması gereken amaçlar ve üretilmesi gereken süreç çıktıları listelenmektedir. Bu tablolara göre, yazılım

kritiklik seviyesine göre karşılanması gereken toplam amaç sayısı Tablo 1'de verilmiştir.

Tablo 1. DO-178B Standardı Yazılım Seviyelerine Göre Karşılanması Gereken Amaç Sayısı

Yazılım Seviyesi	Karşılanması İstenilen Amaç Sayısı
A	66
B	65
C	57
D	28
E	0

DO-178B uyumu otoriteler tarafından "Stages of Involvement" (SOI) adı verilen gözden geçirme aktiviteleri vasıtasıyla değerlendirilmektedir [4]. İdeal durumlarda toplamda dört adet SOI gözden geçirmesi planlanmaktadır. Bu çalışmada, Türkiye'de 2015 yılı içerisinde farklı yüklenicilere yapılan 6 adet SOI#1 denetiminin verileri incelenerek, SOI#1 gözden geçirmelerinde en çok karşılaşılan sıkıntılar ortaya konulmaya çalışılacaktır. Çalışmanın hedefi sektörün ortak yaptığı hataları ortaya koyup, yürümekte olan ya da yeni başlayacak projelere alınan dersleri aktarabilmektir.

Bu çalışmanın ikinci kısmında DO-178B uyumlu geliştirilmesi gereken yazılımların değerlendirilmesi esnasında kullanılan "Stages of Involment (SOI)" süreci hakkında genel bilgi verilecektir. Çalışmanın üçüncü kısmında, 2015 yılında ülkemizde çeşitli projeler kapsamında DO-178B uyumlu yazılım geliştirme projelerinin planlama sürecini değerlendirmeye yönelik olarak gerçekleştirilmiş olan SOI-1 değerlendirme faaliyetleri esnasında tespit edilmiş olan bulguların kaynaklandığı Job Aid soru listesi soruları irdelenecektir. Çalışmanın son bölümünde ise sonuç ve önerilere yer verilecektir.

2 SOI Gözden Geçirmeleri

FAA'in "JobAid: Conducting Software Reviews Prior to Certification" isimli dokümanında, kendi yetkisini kısmen delege ettiği uzmanlarının DO-178B uyumu amacıyla yapacakları değerlendirmelerde nasıl bir yol izlemeleri gerektiğine yönelik tavsiyeler yer almaktadır [4]. Benzer bir şekilde EASA'nın yayımladığı "CM - SWCEH – 002 Software Aspects of Certification" sertifikasyon bilgi notu da SOI gözden geçirmelerini geçerli bir DO-178B değerlendirme yöntemi olarak tanımlamaktadır [5].

Her iki kaynak dokümanda da dört adet SOI aktivitesinden bahsedilmektedir. İlk yapılan değerlendirme olan SOI#1, projenin sertifikasyon uyum çalışmalarının planlanma fazını içermektedir. SOI#2 gözden geçirmesi projenin geliştirme aşamalarına yoğunlaşır. Üst seviye yazılım gereksinimlerinin geliştirilmesinden başlayarak, tasarımın, alt seviye yazılım gereksinimlerinin, ve kaynak kodun DO-178B'ye uygun üretilip üretilmediğini sorgular. SOI#3 gözden geçirmesi doğrulama aşamalarına odaklanmaktadır. Sadece test aktiviteleri değil, doğrulama amacıyla

gerçekleştirilen tüm aktiviteler (gözden geçirmeler, analizler vs) mercek altına yatırılır. SOI#4 ise projenin sonlanmasına yakın yapılan, açıkta yapılmamış bir aktivitenin kalıp kalmadığını ve bir önceki değerlendirmelerden bu yana yeni bir sorunun ortaya çıkıp çıkmadığını sorgulayan kapanış değerlendirmesidir.

Bu çalışma kapsamında SOI#1 gözden geçirmeleri ele alınacağı için bu değerlendirmeye dair daha detaylı bilgi aktarmak faydalı olacaktır. DO-178B kapsamında yüklenicilerin beş adet plan üç adet standart hazırlaması beklenmektedir [6]. DO-178B gereği yüklenicilerden istenilen beş adet yazılım planı, Yazılım Sertifikasyon Konuları Planı (PSAC), Yazılım Geliştirme Planı, Yazlım Kalite Güvence Planı, Yazılım Doğrulama Planı ve Yazılım Konfigürasyon Yönetim Planından oluşmaktadır [6]. Üç adet standart ise Yazılım Gereksinimi Standardı, Yazılım Tasarım Standardı ve Yazılım Kodlama Standardıdır [6].

DO-178B'nın planlama aşamasında üretilen yazılım ürünlerinden beklentisi projeyi değerlendiren ile yapan taraf arasında ortak bir kural kümesinin üzerinden anlaşılması ve proje süresince bu kural setine uyulmasıdır. Bir başka deyişle yüklenici planlarını ve standartlarını yazarken sadece kendi süreçlerini belirlemek ile kalmaz, aynı zamanda sertifikasyon irtibat süreci boyunca otoritenin kendisini nasıl değerlendireceğinin de kurallarını yazmış olur. İşte bu sebepledir ki sertifikasyon otoriteleri plan ve standartları incelerken oldukça titiz davranmaktadırlar. JobAid: Conducting Software Reviews Prior to Certification dokümanındaki SOI#1 değerlendirme sorularından olan "Plan ve standartların takip edilmesi tüm uygulanabilir DO-178B amaçlarının karşılanmasını garanti ediyor mu?" sorusu (1.1.11 numaralı soru) bu sebeple çok kritiktir. Yüklenicinin planlarını inceleyen ve uygun bulan bir otorite temsilcisi 1.1.11 sorusuna "evet" cevabını vererek yüklenicinin planlarına ve standartlarına bir anlamda kefil olmuş demektir. Dolayısıyla 1.1.11 ve benzeri sorular uygun bulunmadan önce tüm DO-178B isterlerinin mevcut planlama ile karşılandığının net ispatı, değerlendirenler tarafından dikkatlice sorgulanmalıdır.

Yukarıdaki açıklamalardan da tahmin edileceği üzere SOI#1 değerlendirmesi esnasında planlar ve standartlarda yazılanların otorite temsilcisi ve yükleniciler tarafından aynı şekilde anlaşılıyor olması önem arz etmektedir. Bu sebeple planlama esnasında üretilen dokümanların yanlış anlamaya mahal vermeyecek netlikte yazılıyor olması, sadece yerel bilgiler ile anlaşılıyor olmaması gerekmektedir.

Otorite temsilcisi, SOI#1 değerlendirmesi sonunda yüklenicinin planlama süreci ürünlerinin, projenin DO-178B'ye uyumu sağlayacak adımları ve kontrolleri içerdiğine ikna olması durumunda değerlendirme başarı ile tamamlanmaktadır. Eğer eksiklikler fark edilmiş ise eksikliklerin değerlendirilmesi yapılarak, ortaya çıkan kritiklik durumuna göre gözden geçirmenin başarım durumu belirlenmektedir.

3 SOI-1 Planlama Sürecinde Sık Karşılaşılan Problemler

Bu bölümde, ülkemizde DO-178B uyumlu yazılım sertifikasyon projelerinde planlama sürecinde yaşanan problemleri tespit edebilmek adına, 2015 yılında emniyet kritiklik seviyesi B olan çeşitli projelerde gerçekleştirilen SOI-1 değerlendirmeleri

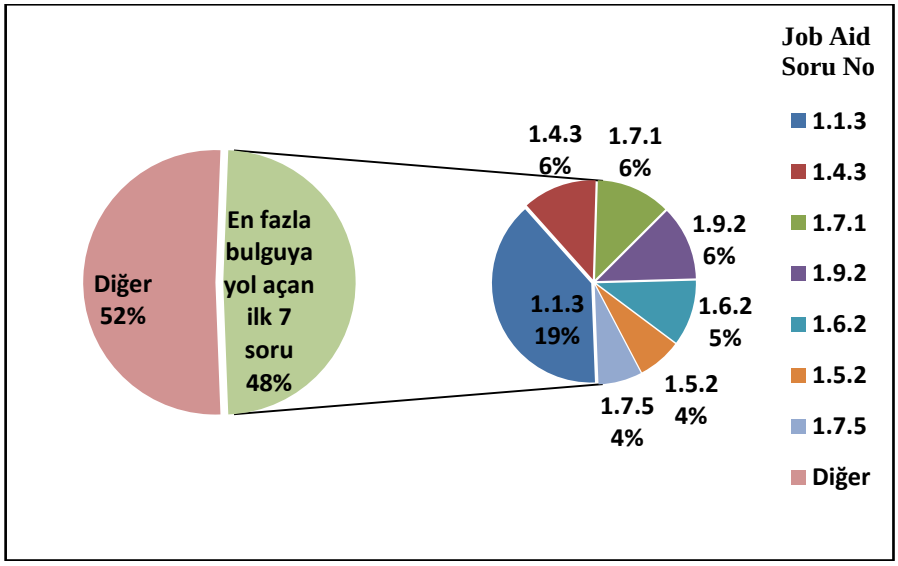
sırasında tespit edilen bulgular analiz edilmiştir. Değerlendirilmelerde "Job Aid-Conducting Software Reviews Prior To Certification"(bundan sonra kısaca "Job Aid" olarak adlandırılacaktır) dokümanının ekinde verilen soru listeleri kullanılmakta ve tespit edilen bulguların hangi sorudan kaynaklandığı kaydedilmekte ve raporlanmaktadır. SOI-1 denetimine ait soru listesinde planlama sürecinin DO-178B amaçlarını karşılayıp karşılamadığını belirlemek amacıyla toplamda 94 adet soru bulunmaktadır. 2015 yılı içerisinde, farklı firmalar tarafından geliştirilen toplam 6 adet yazılım konfigürasyon parçasına SOI-1 değerlendirme aktivitesi gerçekleştirilmiştir. Bu yazılım konfigürasyon parçalarının tamamı DO-178B uyumlu olması gereken yazılımlardır.

Veriler analiz edildiğinde, gerçekleştirilen 6 adet SOI-1 değerlendirmesi sonucunda toplam 289 adet bulgu tespit edildiği görülmüştür. Bulgular ilişkili oldukları soru listesi maddesine göre sınıflandırıldığında ise, toplam bulgu sayısının %48'ine karşılık gelen 141 tanesinin 7 adet sorudan kaynaklandığı tespit edilmiştir. Veriler incelendiğinde geriye kalan 148 adet bulgunun ise 53 adet sorudan kaynaklandığı görülmektedir. Bulguların yaklaşık yarısının tespit edilmesine yol açan 7 adet soru, ilişkilendirildikleri bulgu sayısına göre büyükten küçüğe sıralı olacak şekilde Tablo 2'de verilmiştir.

Tablo 2. SOI-1 Değerlendirmelerinde En Fazla Bulguya Sebep Olan Sorular

Job Aid Soru No	Job Aid Sorusu	İlgili Amaçlar	Bulgu Sayısı
1.1.3	Plan ve standartlar tam, açık ve tutarlı mı?	A-1, #1,7	55
1.4.3	Yazılım geliştirme süreçleri yazılım yaşam döngüsü süreçlerinin ve modelinin başarıyla uygulanmasını garanti altına alacak detay seviyesinde tanımlanmış mı? Geçiş kriterleri açık ve zorlayıcı mı?	A-1, #1-4	17
1.7.1	Yazılım Doğrulama Planının takip edilmesiyle DO-178B Tablo A-3, A-4, A-5, A-6 ve A-7 amaçlarının karşılandığı garanti ediliyor mu?	A-3 to A-7 (bütün amaçlar)	17
1.9.2	Yazılım geliştirme standartları planlarla uyumlu ve planların uygulanmasını destekler nitelikte mi?	A-1, #5	17
1.6.2	Yazılım Kalite Güvence süreci DO-178B bölüm 8.0'da tanımlanan süreçlerin içeriğini yeterli detayda karşılıyor mu?	A-1, #1	15
1.5.2	Yazılım Konfigürasyon Yönetimi süreçleri DO-178B bölüm 7.0'da tanımlanan süreçlerin içeriğini yeterli detayda karşılıyor mu?	A-8, #1-6	10
1.7.5	Yazılım Doğrulama Planı içerisinde her bir doğrulama aktivitesinde kullanılacak doğrulama metodu belirtilmiş mi?	A-1, #1-3	10

SOI-1 değerlendirme faaliyetleri esnasında yukarıdaki tabloda verilmiş olan Job Aid sorularına ilişkin tespit edilen bulguların dağılımı Şekil 1'de yer almaktadır.



Şekil 1 Bulguların Job Aid Sorularına Göre Dağılımı

Veriler incelendiğinde, SOI-1 değerlendirmeleri esnasında tespit edilen bulguların en fazla Job Aid dokümanının ekinde yer soru listesinde bulunan "Plan ve standartlar tam, açık ve tutarlı mı?" sorusundan kaynaklandığı görülmektedir. Burada veriler daha ayrıntılı incelendiğinde, tespit edilen bulguların genelde planlar arası uyumsuzluk ve tutarsızlıklardan kaynaklandığı görülmektedir. Bu uyumsuzluk ve tutarsızlıkların başlıcaları aşağıda listelenmektedir:

- Planlarda aynı rollerin farklı şekilde ifade edilmesi
- Bir planda kullanılacağı belirtilen aracın(geliştirme veya doğrulama) diğer ilgili planlarda geçmemesi
- Planlar arası bölüm referanslarındaki hatalar
- Planlarda "daha sonra belirlenecek" gibi ifadelerin geçmesi

DO-178B uyumlu yazılım geliştirme projelerinde temel olarak beş adet plan geliştirilmesi beklenilmektedir. Bu planlar projede rol alan farklı taraflar tarafından geliştirildiği için kendi aralarında tutarsızlıkların oluşması sık rastlanan bir problemdir. DO-178B uyumlu yazılım geliştiren firmaların, bu problemi önlemek adına çeşitli mekanizmalar geliştirmesi gerekmektedir. Bunlardan ilk akla gelen, plan gözden geçirmelerinin tamamına aynı ekibin katılmasıdır. Bu şekilde, sabit bir ekip tüm planları inceleyip olası tutarsızlıkları gözden geçirme sürecinde yakalayabilecektir. Ayrıca bu mekanizma sayesinde, projenin tüm paydaşları görev ve sorumluluklar hakkında hemfikir olacaktır.

SOI-1 değerlendirmelerinde en çok bulgu tespit edilmesine yol açan ikinci soru ise "Yazılım geliştirme süreçleri yazılım yaşam döngüsü süreçlerinin ve modelinin

başarıyla uygulanmasını garanti altına alacak detay seviyesinde tanımlanmış mı? Geçiş kriterleri açık ve zorlayıcı mı?" sorusudur. Burada bulgular ayrıntılı olarak incelendiğinde, ilgili soruya ait bulguların önemli kısmının "geçiş kriterlerindeki eksikliklerden" kaynaklı olduğu görülmektedir. DO-178B standardı geçiş kriterini, yazılım yaşamdöngüsünde yer alan herhangi bir alt sürece başlamak için sağlanması gerekli asgari koşullar olarak tanımlamaktadır [6]. DO-178B standardının geçiş kriterlerini zorunlu tutmasının başlıca sebebi, bir sürece başlamadan önce tamamlanması gereken diğer süreçlerin yeterli olgunlukta olduğunun teminat altına alınmak istenmesidir. Dolayısıyla geçiş kriterleri belirlenirken, mutlaka DO-178B'nin ilgili süreçlere ilişkin amaçları da göz önünde alınmalıdır. Geçiş kriterleri sadece standart istediği için konulmamalı, gerçekten ilgili süreçteki aktivitelerin düzgün şekilde ilerlemesine engel olabilecek konular geçiş kriteri olarak belirlenmelidir. Bu sorudan kaynaklı diğer bulgular ise genelde planlarda işin yapılmasına ilişkin yeterli detay verilmemesinden kaynaklanmaktadır. Her ne kadar, projenin başlangıcında gerçekleştirilecek faaliyetleri en ince detayına kadar planlamak mümkün olmasa da, aktivitelerin otorite temsilcisi dahil tüm paydaşlarca aynı şekilde anlaşılabilmesine yetecek detayda planlanması gerekmektedir. Örneğin planlarda DO-178B'nin beklediği kod ile gereksinimler arasındaki izlenebilirlik konusunda planlarda bu izlenebilirliğin hangi seviyede(method-fonksiyon seviyesi, sınıf seviyesi gibi) kurulacağı mutlaka belirtilmelidir. Aksi takdirde, sadece izlenebilirlik kurulacaktır şeklinde yazılacak genel bir ifade, otorite temsilcisi ve proje çalışanları tarafından farklı şekillerde yorumlanabilir. Otorite temsilcisinin bu cümleden beklentisi, kod satırı bazında bir izlenebilirlik olması iken, proje çalışanlarının aklındaki izlenebilirlik seviyesi modül bazında olabilir. Bu muğlak ifade SOI-1 değerlendirmesi esnasında gözden kaçarsa, ileriki değerlendirmelerde otorite ile firma arasında ciddi fikir ayrılıklarına yol açacak, projenin takvim ve bütçesini modül bazlı izlenebilirlik kurmak üzerine inşa eden firma, otoriteyi tatmin edebilmek için, önceden planlamadığı ek çalışmalar yapmak durumunda kalabilecektir. Burada sergilenebilecek en tehlikeli yaklaşım, konuyu öteleyip, ilgili aşama gelince mevcut duruma göre bir yöntem seçmektir. Bu durum hem planlamanın ruhuna hem de SOI-1 sonunda otorite ile varılması gereken mutabakata aykırılık teşkil etmektedir [1].

SOI-1 değerlendirmeleri esnasında üçüncü en çok bulgu tespit edilmesine yol açan soru "Yazılım Doğrulama Planının takip edilmesiyle DO-178B Tablo A-3, A-4, A-5, A-6 ve A-7 amaçlarının karşılandığı garanti ediliyor mu?" sorusudur. Bu soruya ilişkin tespit edilen bulgular detaylı şekilde incelendiğinde, bulguların genelde standardın istediği "en kötü çalışma zamanı analizi (worst case execution time analysis)", "bellek kullanım analizi(memory usage analysis)" gibi bazı analizlere ilişkin Yazılım Doğrulama Planında planlama yapılmamış olmasından kaynakladığı görülmektedir. DO-178B standardının amaç tablolarında bu analizlerin istenildiğine dair net ifadeler bulunmamaktadır, ancak standardın ilgili bölümlerine ayrıntılı bilgi için referanslar verilmektedir. Bu çalışma kapsamında ele alınan projelerde planlar hazırlanırken genellikle sadece amaç tabloları üzerinden hareket edildiği görülmüştür. Bu tarz problemlerin yaşanmaması adına, planlar hazırlanırken DO-178B dokümanının tamamının dikkatli şekilde okunması ve ayrıca SOI değerlendirmelerinde temel alınan Job Aid dokümanındaki soru listelerinin de

incelenmesi gerekmektedir. Bu sayede planlamanın standart tarafından istenilen ancak çok açık şekilde ifade edilmemiş konuları da kapsaması sağlanabilecektir.

SOI-1 değerlendirmeleri esnasında proje planlarının yanısıra, geliştirme aşamasında kullanılacak kodlama, tasarım ve gereksinim standartları da incelenmektedir. SOI-1 değerlendirmelerinde en çok bulgu tespit edilen dördüncü soru da bu standartlara ilişkin "Yazılım geliştirme standartları planlarla uyumlu ve planların uygulanmasını destekler nitelikte mi?" sorusudur. Bu soruya ilişkin bulgular incelendiğinde; genelde bulguların emniyetli bir yazılım geliştirmek adına kural olarak belirlenmesi gereken kısıtların, tavsiye olarak nitelenmesinden kaynaklı olduğu görülmektedir. Yazılım geliştirme standartları hazırlanırken, katı kurallar koymak yazılım geliştiricileri kısıtlamakta ve ayrıca bu kuralların uygulandığından emin olmak için de ciddi bir işgücü harcanmaktadır. Bu sebeple, geliştirme standartları hazırlanırken gerçekten kural olması gereken konular kural olarak belirlenmeli, kalan konular ise tavsiye olarak sınıflandırılmalıdır. Kısıtlayıcılığı azaltmak adına, standartlar hazırlanırken yazılımın deterministik olmayan davranışlar sergilemesine yol açabilecek bazı kullanımların tavsiye olarak sınıflandırılabilmektedir. Bu durum SOI-1 değerlendirmelerinde bulgu açılmasına yol açmaktadır. Örneğin tip dönüştürmeye ilişkin (type casting) Misra C standardında yer alan 10.3 numaralı kuralı [7], hazırlanan kodlama standardında tavsiye olarak sınıflandırmak yazılımın hatalı çalışmasına sebebiyet verebilir. Bu sebeple Misra C gibi standartların kural olarak sınıflandırdığı sınırlamalar, gerçekten çok geçerli bir sebebi ve mantıklı bir açıklaması yoksa proje kapsamında yazılan kodlama standartlarında tavsiye olarak sınıflandırılmamalıdır. Yazılım geliştirme standartları hazırlanırken, kurallar ve tavsiyeler belirlenirken bu bakış açısı ile bakılmalı ve kurallar belirlenmelidir.

Job Aid soru listesinde yer alan "Yazılım Kalite Güvence süreci DO-178B bölüm 8.0'da tanımlanan süreçlerin içeriğini yeterli detayda karşılıyor mu?" sorusu ise SOI-1 değerlendirmeleri esnasında en fazla bulgu tespit edilmesine yol açan beşinci sorudur. Bu soruya ilişkin bulgular incelendiğinde, problemlerin kaynağının planda işlerin nasıl yürütüleceğine dair yeterli detay verilmemesi olduğu görülmektedir. Örneğin Kalite Güvence Mühendisinin projedeki test, gözden geçirme gibi bazı aktivitelere hangi oranda katılacağı, Kalite Güvence Mühendisinin projede yapacağı iç denetimlerde kullacağı örneklem oranları gibi bazı detaylardan planlarda hiç bahsedilmediği sıklıkla görülmektedir. Bu durum işgücü planlamasının sağlıklı şekilde yapılamamasına yol açacaktır. Bu sebeple planlamada bu detayların verilmesi gerekmektedir. Ayrıca planlar onaylanırken, sertifikasyon otoritesinin bu tarz detayları bilmesi ve onaylaması gerekmektedir. Planlar hazırlanırken bu hususlar mutlaka dikkate alınmalı ve planlarda açık şekilde ifade edilmelidir.

SOI-1 değerlendirmelerinde en fazla bulgu tespit edilmesine yol açan altıncı soru ise "Yazılım Konfigürasyon Yönetimi süreçleri DO-178B bölüm 7.0'da tanımlanan süreçlerin içeriğini yeterli detayda karşılıyor mu?" sorusudur. Buradaki problemler de gene bir önceki paragrafta anlatılan probleme benzemekte ve konfigürasyon yönetimine ilişkin aktivitelerdeki detay eksikliğinden kaynaklanmaktadır. Bulgular incelendiğinde, konfigürasyon kontrol kurulu üyelerinin ve karar verme mekanizmasının net olarak anlatılmaması, üst ya da alt yükleniciden nasıl hata bildirimi alınacağı ve/veya verileceğine ilişkin detaylı sürecin tanımlanmaması gibi

konfigürasyon yönetimine dair detayların yeterli seviyede anlatılmamış olmasının temel sıkıntı olduğu görülmektedir. Bu detayların, projede çalışan kişilerin işlerini rahatlıkla yürütebilmelerini sağlayacak seviyede verilmesi gerekmektedir. Örneğin, konfigürasyon kontrol kurulunda kararlar verilirken, bir anlaşmazlık olması durumunda nasıl bir yöntem izlenerek (oy çokluğu, konfigürasyon kontrol kurulu başkanının nihai kararı vermesi gibi) nihai kararın verileceği planlarda net olarak açıklanmalıdır.

SOI-1 değerlendirmelerinde en fazla bulgu tespit edilmesine yol açan en son soru ise "Yazılım Doğrulama Planı içerisinde her bir doğrulama aktivitesinde kullanılacak doğrulama metodu belirtilmiş mi?" sorusudur. DO-178B yazılım projelerinde doğrulama aktivitelerinden birisi olan gözden geçirmelerde soru listeleri kullanılması zorunludur. Projelerde kullanılan bu soru listeleri genelde Yazılım Doğrulama Planının eki olarak verilmektedir. Bu soruya ilişkin bulgular incelendiğinde, bulguların çoğu zaman Yazılım Doğrulama Planının ekinde verilen ve projede gözden geçirme aktivitlerinde kullanılacak olan soru listelerinin DO-178B amaçlarını tam olarak sağlayacak yeterlilikte olmamasından kaynaklandığı görülmektedir. Bunun haricinde gene gözden geçirme gibi bazı doğrulama aktivitelerinin nasıl yürütüleceğine ilişkin yeterli detayın verilmemiş olması da bir başka ana problem kaynağı olarak görülmektedir. Bu tip bulguların önüne geçebilmek adına, planlama aşamasında kontrol listeleri DO-178B standardında yer alan amaçları tam olarak karşılayabilecek şekilde geliştirilmelidir. Hazırlanan soru listeleri ile karşılanması gereken DO-178B amaçlarının arasında izlenebilirlik kurmak ve karşılanmayan amaçların kalmadığına dair bir analiz yapmak bu sıkıntıyı çözebilecek yöntemlerden biridir. Ayrıca diğer aktivitelerde olduğu gibi yazılım doğrulama aktivitleri için de yeterli detay planlarda verilmelidir.

4 Sonuç ve Öneriler

Bu çalışmada, ülkemizde DO-178B uyumlu yazılım sertifikasyon projelerinde planlama sürecinde yaşanan problemleri tespit edebilmek adına, 2015 yılında çeşitli projelerde gerçekleştirilen SOI-1 değerlendirmeleri sırasında tespit edilen bulgular analiz edilmeye çalışılmıştır. Ülkemizde havacılık alanında yapılan sistem ve platform geliştirme çalışmaları gittikçe yoğunlaşmaktadır. Milli Muharip Uçak(TF-X) geliştirme projesi ve Yerli Bölgesel Uçak gibi projeler de göz önüne alındığında, ülkemizde ileriki yıllarda daha fazla aviyonik yazılım geliştirmesi yapılacağı öngörüsünde bulunmak yanlış olmayacaktır. Bu sebeple, özellikle bu alana yeni girecek olan yazılım üreticisi firmaların uymaları gereken DO-178B yazılım geliştirme standardına hakim olmaları büyük önem arz etmektedir. DO-178B standardını kullanarak yazılım geliştirecek olan firmaların, bu çalışmada ortaya konulmuş olan planlama sürecine ilişkin problemleri önceden görmeleri, standardın istediği planlama amaçlarını sağlamalarında katkı sağlayacaktır.

DO-178B uyumlu yazılım geliştirme projelerinde, planlama sürecinin amaçlarının oldukça iyi anlaşılması gereklidir. Bu noktada sadece standartta yer alan amaç tablolarını okumak yeterli olmayacaktır. Standardın tamamı dikkatli şekilde

incelenmelidir. Buna ek olarak, mutlaka DO-178B standardına yardımcı doküman olarak düşünülebilecek SOI değerlendirmelerinde kullanılan soru listelerini içeren Job Aid dokümanı, Certification Authorities Software Team (CAST) tarafından hazırlanmış olan "CAST Position Paper"'lar ve EASA'nın yayımladığı "CM - SWCEH–002 Software Aspects of Certification" sertifikasyon bilgi notu gibi dokümanlar da göz önüne alınıp detaylı şekilde incelenmelidir. Bu şekilde, DO-178B amaç tablolarında oldukça genel olarak yazılmış ifadelerin, aslında neyi anlatmak istediği daha iyi şekilde anlaşılabilecektir. Sonrasında bu amaçları sağlayacak şekilde bir proje planlaması yapılıp ilgili çıktılar oluşturulmalıdır.

DO-178B uyumlu yazılım geliştirme yapan firmaların, otoriteler tarafından yapılan SOI uyum değerlendirmeleri sonrasında, tespit edilen bulgulara ilişkin bir kök neden analizi çalışması yapmaları ve buradan çıkan sonuçları "alınan dersler" şeklinde dokümante ederek, firmada ilerideki projelerinde kullanılmak üzere saklamaları da önerilmektedir. Bu yolla, firmalar daha önceden yaşadıkları sıkıntı ve problemleri, ileriki dönemde yaşamayacak ve aynı hataları tekrarlamayacaklardır.

Gelecek dönemde, bu çalışmanın devamı olarak benzer şekilde Yazılım Gereksinimleri, Tasarımı ve Kodlama için yapılan SOI-2 ile Doğrulama faaliyetleri için yapılan SOI-3 uyum değerlendirmeleri sonuçlarının analiz edilmesinin faydalı olacağı değerlendirmektedir. Bu vesileyle, DO-178B standardına uyumlu yazılım gelişterecek olan firmalara dikkat etmeleri gereken hususlar konusunda faydalı bir literatür kaynağı sunulması ve firmaların SOI uyum değerlendirmelerinde yaşayabilecekleri olası başarısızlıkların önüne geçilmesi hedeflenmektedir.

5 Teşekkür

Bu bildirinin yazılması sırasındaki değerli desteklerinden dolayı SSM Sertifikasyon Müdürü Mehmet Yetiş Uysal'a, STM A.Ş. Mühendislik ve Sertifikasyon Müdürü Reşat Erhan Yüceer'e ve aynı bölüm çalışanlarından Sinem Yalçınkaya ile D. Gizem Pektaş'a teşekkür ederiz.

6 Kaynakça

1. Leanna Rierson, "Developing Safety-Critical Software", CRC Press, 1st Edition, 2013.
2. FAA, "Advisory Circular Number 20-115C - Airborne Software Assurance", 2013.
3. Ian Dodd, Ibrahim Habli, "Safety Certification of Airborne Software:An Empirical Study", Reliability Engineering and System Safety, 98, Sayfa 7–23, 2012.
4. Aircraft Certification Service, "Job Aid-Conducting Software Reviews", Rev 1 2004.
5. EASA, "CM - SWCEH – 002 Software Aspects of Certification", Rev 01, 2012.
6. RTCA, "DO-178B Software Considerations in Airborne Systems and Equipment Certification", 1992.
7. MISRA, "MISRA-C: 2004 Guidelines for the use of the C language in critical systems", MIRA Limited, Edition 2, 2008.