

Uluslararası Geliştirilen Bir Projenin İşleyişi, Gözlemlerimiz ve Edinilen Tecrübeler

Emra Askaroglu¹, Hakan Yılmaz²

¹ASELSAN A.Ş. SST-KKYTM P.K.1 06172, Yenimahalle/Ankara, Türkiye

²ASELSAN A.Ş. SST-KKYTM P.K.1 06172, Yenimahalle/Ankara, Türkiye

¹easkaroglu@aselsan.com.tr, ²hakyilmaz@aselsan.com.tr

Özet. Günümüzde farklı şirketler, organizasyonlar ya da ülkeler ortak geliştirmeye çalıştıkları projelerde birçok güçlüklerle karşılaşmaktadırlar. NATO ülkeleri tarafından ortak olarak geliştirilen ve projelerimizde kullandığımız çekirdek yazılımları bize farklı ülkelerle ve şirketlerle aynı proje için çalışmanın getirdiği sorumlulukları ve güçlükleri göstermektedir. Bu kapsamda bir projenin farklı ülkelerle birlikte geliştirilmesi sırasında proje yönetiminin nasıl ele alındığı, geliştirme sırasında belirlenen stratejilerin neler olduğu, geliştirme sonrasında yapılacak testlerin nasıl yapılması gerektiği, konfigürasyon yönetiminin nasıl ele alındığı önem arz etmektedir. Bu makalede diğer ülkelerle ortak geliştirilen çekirdek yazılım çalışmaları kapsamındaki gözlemlerimizi ve edindiğimiz tecrübeleri paylaşacağız.

Anahtar Kelimeler: NATO, Proje Yönetimi, Yazılım Süreçleri, Yazılım Testi, Yazılım Kalitesi

1 Giriş

S4 (SG/2 Sharable Software Suite) NATO ARMY ARMAMENTS GROUP Integrated CAPABILITY Group - Indirect FIRE Sub-Group 2 (NAAG-ICG-IF-SG/2) grubunun bünyesinde müşterek olarak geliştirilen programlar takımıdır[1]. Bu makalede S4 projesi aşamalarından ve projenin geliştirilmesi sırasında edinilen tecrübelerden bahsedilecektir.

Daha önce yapılmış bir çok çalışmada projelerin yaşam döngüleriyle ilgili adımlar sunulmuştur[8][9][10]. Yazılım tasarımı, geliştirmesi, yönetimi, idame edilmesi gibi konular yazılım dünyasında her zaman üzerinde önemli bir konu olarak var olmaya devam edecektir[11]. S4 çalışması ile ilgili bu makalede projenin tüm yaşam döngüsüne odaklanmak yerine ülkelerin ortak çalışmasına ve aynı dili konuşmasına yardımcı olacak anahtar noktaların üzerine gitmeye karar verdik.

Makalenin 2. bölümünde NATO ülkeleri ile birlikte geliştirilen S4 projesi hakkında bilgi verilecek, 3. bölümde S4 projesinin geliştirme aşamaları, 4. bölümde ise bu aşamalarda alınan dersler anlatılacaktır. Makalemiz Sonuç bölümü ile sonlanacaktır.

2 S4 Projesi

S4 içerisindeki her program ayrı bir projedir ve her projede yazılım ürünleri üretilmektedir. Bu ürünler bir araya gelerek S4'ü oluşturmaktadır[2]. S4 bünyesinde geliştirilen ürünlerin temel amacı balistik hesaplama yeteneği sağlayan ateş destek çekirdek yazılımları üretmektir[3]. Bu kapsamda balistik teknolojiyi kurumsallaştırmak ve standartlaştırmak, mühimmat değişimi ve verilerini standartlaştırmak, balistik hesaplama etki eden faktörleri incelemek, STANAG¹'lar tarafından kapsanan verilerle ilgili gerçekleştirmeyi standartlaştırmak, program çerçevesinin(Framework) kurulması ve yazılım geliştirme işlevinin süreç tabanlı yapıda olmasını sağlamak gibi konular üzerine çalışmalar yapılmıştır.

S4 çalışmalarına 1994 yılında ABD ve Norveç önderliğinde başlanmıştır. Türkiye bu çalışmalara 1997 yılında dahil olmuştur. 1997 yılından itibaren gözlemci olarak dahil olmakla birlikte, bazı dönemlerde yazılım mühendisliği, yazılım kalite mühendisliği desteğinde bulunmuştur. Ayrıca geliştirilen çekirdek yazılımlarına girdi oluşturacak verilerin toplanmasıyla ilgili çalışmalara da aktif katılım sağlanmıştır[4].

3 S4 projesinin geliştirme aşamaları

3.1 Proje Organizasyonu

S4 projeleri organizasyonlarında ülkeler ve ülke temsilcileri proje lideri, ürün lideri, ürün geliştirme lideri, ürün test lideri, bağımsız yazılım denetçisi (ISA - Independent safety/software Auditor) gibi sorumluluklara sahip olabilmektedirler. Proje lideri koordinatörlüğünde projenin ürünlerinin liderleri ve diğer sorumluları ile birlikte belirli periyotlarla – ortalama 6 ayda 1 - yapılan toplantılarda projenin yeni sürüm gözden geçirme ve değerlendirme toplantıları yapılmaktadır. Bu toplantılara ilgili ürünlerin kullanıcıları da davet edilmektedir. Sonuç olarak ülkeler toplantılara “gözlemci”, “gözden geçiren”, “lider” ve “aktif kullanıcı” olarak katılım sağlarlar. Ürünler için yılda 1 kez olmak üzere yazılım geliştirme gözden geçirme ve değerlendirme toplantıları da düzenlenmektedir.

3.2 S4 Yaşam Döngüsü

S4 çalışmaları için ilk zamanlarda yazılım geliştirme süreci olarak Şelale (Waterfall) modeli kullanılmaktaydı. Daha sonraları tüm çekirdek yazılımları için Sarmal (Spiral) model uygun görülmüş ve uygulanmıştır.

Bir projenin yeni sürüm planı yapılırken öncelikle ClearQuest'e önceki sürümler için girilmiş hatalar ve istekler gözden geçirilerek başlanır(Requirements Summary List - RSL). Daha sonra hatalar ilgili yazılım geliştiricilere atanır. Yeni geliştirilecek fonksiyonallikler için gönüllülük esasına göre yazılımcılar belirlenir ve atamalar yapılır. Kodlamalar tamamlandığında her fonksiyonallite için testler planlanır ve

¹ NATO üyesi ülkelerin askeri alandaki standartlarını belirleyen bir bildirimdir.

gerçeklenir. Yazılımlardaki deęişiklikler “Software Change Request – SCR” veri tabanında tutulmaktadır. Son olarak yazılımla ilgili hata raporları (bug report), test raporları (test report) ve durum raporları (status report) yayınlanır.

Ürünlerin her yeni sürümü önceki 2 sürüm için geriye dönük uyumluluęu destekleyecek şekilde geliştirilmektedir. Geliştiriciler için S4 ADA Kodlama Standardı, S4 Ürün Geliştirme Standartları vb. referans dokümanlar bulunmaktadır. [6].

3.3 Konfigürasyon Yönetimi

S4 ürünleri geliştirilirken hafızanın sınırlı olduęu sistemlerde yaşanan sıkıntılar göz önünde bulundurularak dinamik bellek kullanımına izin verilmez. Bunun yerine nesnelere, dizilere, listelere ve benzeri veri yapılarının boyutları derleme zamanında yapılandırma dosyaları kullanılarak belirlenir. Çekirdek yazılımları katmanlarının kendine ait bir yapılandırma dosyası bulunmaktadır. Bu yapılandırma dosyalarında ilgili katmanda kullanılan veri tipleri ve bu verilerin boyutları bulunmaktadır. İlgili katman için kullanılacak kaynak önceden belirlenmiş olur. S4 çekirdek yazılımlarını kullanacak projelerin ihtiyaçlarına göre bu yapılandırma dosyaları düzenlenmeli ve çekirdek yazılımlar bu ayarlara göre derlenmelidir. Uyumlu olmayan yapılandırma dosyası verileri çalışma zamanında programın hata vermesine neden olacağı için ayarlamaların dikkatli bir şekilde yapılması gerekmektedir. Kaynak problemi olmayan projeler için çekirdek yazılımları yapılandırma dosyaları en geniş verilerle ya da varsayılan deęerlerle derlenerek kullanılabilir.

3.4 Yazılım Testleri

Bir ürünün yeni sürümü için yazılım geliştirme çalışmaları tamamlandıktan sonra yeni sürümün yayınlanması öncesi ürün için yazılım testleri yapılmaktadır. S4 ürünleri için hazırlanmış otomatik test altyapısı (Autotest) hızlı ve esnek bir test yöntemi olup yazılımların işlevsellięi, hesaplama sonuçlarının uygunluęu ve hataların uygun bir şekilde ele alınıp alınmadığı gibi konuları test eder. Autotest yazılımı çalışırken girdi olarak test betikleri (test scripts) ile dięer girdi dokümanlarını (metro raporu vb.) alır. Test yazılımı test sonuçlarını rapor şeklinde çıktı olarak verir ve yazılımın önceki sürümünün ilgili test adımları sonuçlarıyla karşılaştırır. Bu karşılaştırma sonuçları da test yazılımının ürettięi çıktılar arasında yer alır.

Autotest yazılımının kullanımı dışında yazılımın ürettięi sonuçların geçerli görülen başka yazılımlarla karşılaştırılması yöntemi de yazılım testi kapsamında kullanılmaktadır. Böylece yazılmış algoritmaların geçerlilikleri daha detaylı bir şekilde test edilmiş olmaktadır.

3.5 Kalite Yönetimi

S4 ürünleri geliştirilmesi sırasında program yönetimi, proje yönetimi, mühendislik alanlarında kalite adımları takip edilmektedir. Her adım kendi içinde bir kalite yöntemi barındırmaktadır. Genel olarak baktığımızda S4 ürünleri için aşağıdaki konuların takibi kalite açısından önem kazanmaktadır.

- Program Yönetimi: Koordinasyon, Bakım
- Proje Yönetimi: Proje Planlama, Projenin izlenmesi ve Kontrolü, Risk Yönetimi, Proje güvenliğinin izlenmesi, Ürün Değerlendirme
- Mühendislik Alanı: Gereksinim Yönetimi, Teknoloji geliştirme, Yazılım geliştirme, Konfigürasyon Yönetimi

Son zamanlarda S4 Kalite Yönetimine ISA (Independent Software/Safety Auditor) rolü eklenmiştir. ISA rolü proje yönetimi seviyesinin üstünde yer almaktadır. ISA'nın temel görevi ürünlerin yayınlanacak sürümlerinin denetimini yapmaktır. Yeni sürümün kodlama standartlarına uyumluluğu kontrol edilir. Performans ve güvenlik kriterleri değerlendirilir ve güvenlik standartlarına uygunluk kontrol edilir. Her ürün için çıkan dokümantasyonlar incelenerek uygunlukları değerlendirilir. Bu kontroller sonrasında her ürün sürümü için değerlendirme raporu ISA tarafından yayınlanır. Sonuç olarak ISA rolündeki temsilciler yazılım geliştirme ekipleriyle koordine olarak ürünlerin kalitesini denetlerler.

3.6 Eğitimler

S4 projeleri ve ürünleri için ilgili ürünlerin kullanıcılarına belli aralıklarla toplantılar ve eğitimler düzenlenmektedir. Bu eğitimler ilgili ürünlerin proje lideri tarafından organize edilerek ürünlerle ilgili farkındalığı arttırmak, eğitimler sırasında bilgi paylaşımını sağlamak ve ürünlerin kullanımının artırılmasını sağlamak amacı ile yapılmaktadır. Böylece yazılımların yanlış kullanımının önüne de geçilmiş olacaktır.

Eğitimler için planlama proje liderleri tarafından yapılarak S4 kullanıcılarına mail yolu ile bilgilendirme yapılır. Her ülkeden bu eğitimlere katılacak temsilciler kayıt formu doldurarak bilgilendirme yaparlar. Eğitimler sırasında teorik bilgilendirmenin yanı sıra bir yazılımın daha net anlaşılmasına önemli rol oynayan pratik eğitim de yapılmaktadır. İlgili çekirdek yazılımı kullanan örnek ürünler üzerinden yazılımların kullanımı daha net anlaşılacağı için pratik eğitimlere özellikle önem verilmektedir.

4 Alınan Dersler

Şimdiye kadar S4 programının nasıl bir süreçte ilerlediğini anlatmaya çalıştık. Bunun gibi farklı kullanıcı ve birden fazla ülkenin, dağıtık ekiplerin bir araya gelerek geliştirdiği projelerde nelere dikkat edilmesi gerektiğine dair birçok noktaya değinmiş olduk. Şirketlerde birden çok ekibin (şirket içi farklı ekiplerin ya da diğer firmaların ekipleriyle –alt yükleniciler vb.–) bir araya gelmesiyle yapılan projeler olduğu için bu projelerin her aşamasına katkı sağlayacak birçok şeyden bahsedebiliriz.

Projelerin geliştirmesi sırasında ekiplerin ortak paydada buluşabilmesi için toplantılar yapılması ve toplantılarda **gereksinimlerin gözden geçirilmesi, kullanılacak teknolojilerin gözden geçirilmesi ve hangi teknolojilerin kullanılacağına karar verilmesi** bunların kayıt altına alınıp takibinin düzenli olarak yapılması büyük önem arz etmektedir. Projenin farklı katmanları ile ilgili **sorumluluk**

atamasının yapılması hem geliştiricilerin projeye daha iyi adapte olmalarını hem de projenin birçok açıdan takip edildiğini garantiyecektir.

Yine ortak bir çatıda buluşmak adına projeler için her ekibin uyacağı standartlar belirlenmelidir. Örneğin S4 projelerinde yazılım ekiplerinin bir **kodlama standartları ve ürün geliştirme standartları** bulunmaktadır. Bu standartlara uygun olarak geliştirilen yazılımlar daha sonra bağımsız kalite denetçileri tarafından **gözden geçirilir**. Her yeni sürüm için gerçekleştirilen fonksiyonların **geriye dönük uygunluklarının (Backward Compatibility)** sağlanması farklı ürünlerin farklı sürümlerle çalışabilmesini sağlayacaktır. Bu durum da ilgili yazılımı servis şeklinde kullanan diğer yazılımların her yeni gelen sürüm ile kendini güncellemek zorunda kalmaması anlamına gelecektir. Ayrıca yazılımların çalışacakları platformlara ve kaynaklara uygun olarak üretilmesi diğer bir deyişle “**Güvenli**” bir yazılım olması gerekmektedir. S4 ürünlerinin çalışacakları örnek platformlar kullanıcılar tarafından belirlenip bu platformlarda çalıştıkları garantilendikten sonra sürümler çıkarılmaktadır. Bu durumu ele almak için kullanıcıların kullandıkları derleyiciler de sürüm çıkarılmadan önce S4 ürünleri için denenmektedir.

Projelerin yaşam döngüsünün son adımında **test** aşaması bulunmaktadır. S4 ürünleri için Autotest uygulaması her sürüm de yeni özellikleri de test edecek şekilde güncellenerek kullanılmaktadır. Ayrıca her sürüm testinde çıkan sonuçlar bir önceki sürümde çıkan sonuçlarla karşılaştırılmaktadır. Böylece Autotest uygulaması güvenilir, otomatik ve hızlı bir test mekanizması işletilmesine yardımcı olmaktadır.

Bir projenin yaşam döngüsünün en önemli adımlarından biri de **dokümantasyon** aşamasıdır. S4 projelerinde dokümantasyon büyük önem arz etmektedir. Her ne kadar bilgilerin bir kısmı ekiplerin bir araya gelmesiyle yapılan toplantılarla paylaşılıyor olsa da ürünlerle ilgili detaylı bilgiler ürünlerin dokümantasyonu sayesinde paylaşılmaktadır.

Bunların yanı sıra projenin her aşamasında yapılan işlerin kaliteli yürütülmesi için **kalite sorumluları** atanmalıdır. Bir projenin kaliteli olabilmesi için kalite sorumlularının projenin başlangıç aşamasında tamamlanma aşamasına kadar geçilen bütün süreçte aktif olarak görev yapmaları gerekmektedir. S4 programının en çok önem verdiği konulardan biri de bu kalite anlayışıdır.

Farklı ülkelerin bir araya gelerek ortak olarak geliştirdikleri projelerde projenin geliştirme süreci doğru tanımlandığında kaliteli, güvenli, doğruluk ve geçerlilik açısından test edilmiş/denenmiş ürünler üretilmektedir. Fakat tüm bu avantajların yanında farklı ülkelerle bir araya gelerek geliştirilen projelerde dezavantajlar da bulunmaktadır. Bu tarz projelerde bilgi aktarımının öneminden birçok kere bahsetmiştik. Fakat bilgi aktarımının zorunluluğu birçok açıdan sorun yaratabilir. Örneğin projelerin gizliliği ülkelerin bir araya gelmesini zorunlu kılmaktadır. Ülkelerin temsilcilerinin planlanan toplantılara katılımlarının zor olması, iş takvimlerinin bu programlara uymaması gibi konular sıkıntı yaratmaktadır.

Ayrıca farklı ülkelerin projelerden farklı beklentilerinin olması ve bu beklentilerin planlanamaması /geliştirilememesi diğer bir zorluktur. Bazen bir ülkenin bir ürün için isteği uzun yıllar karşılanamayabilmektedir. Ülkelerin projeler için yeterince iş gücü ayıramamaları ürünlerin gelişimini olumsuz yönde etkileyecektir. S4 projelerinde de yaşanan sıkıntılardan biri de bu kaynak ve zaman problemidir. Özellikle bu gibi

gönüllülük esasına dayanarak ilerleyen projelerde ülkelerin yeterince katkıda bulunmaları ve aktif rol oynayarak alan bilgisine detay seviyesinde sahip olmaları birçok açıdan yararlı olacaktır. Konfigürasyon çeşitliliğinin yarattığı sıkıntılar da dezavantaj oluşturmaktadır. Farklı ülkeler için farklı konfigürasyonlarla geliştirilen projelerin esneklik sağladığı bir gerçektir. Fakat bu esneklik yapılandırma ayarlarının takibini, geliştirme ve test çalışmalarını zorlaştırarak ülkelere ek maliyet getirmektedir.

5 Sonuç

S4 projesi NATO üyesi ülkelerin bir araya gelerek gönüllülük esasına dayanarak geliştirilen, balistik hesaplama başta olmak üzere Ateş İdare projelerinde kullanılabilecek birçok özelliği barındıran çekirdek yazılımlar grubudur. S4 projesi ile bir projenin geliştirilmesi aşamasında göz önüne alınması gereken birçok konu ele alınmıştır. Yapılan toplantılar, eğitimler, proje yönetim süreçleri, konfigürasyon yönetimi, test mekanizmaları, dokümantasyon, kalite yönetimi ve bunun gibi bir çok konunun önemine değindik. Ülkelerin projelere olan katkıları projelerin ilerlemesi açısından önemli bir rol oynamaktadır. Bunun gibi bu projede öğrenilenlerin yanında kaynak – zaman (iş gücü) problemleri, projenin yavaş ilerlemesi gibi dezavantajlardan da bahsettik.

Şirketlerde, yukarıda anlatılan çalışmaların projelere katkı sağlayacağı öngörüsünün uygulamaya konulması kaliteli bir proje geliştirme süresinin yaşanması açısından önem arz etmektedir.

Kaynaklar

- [1] André J. Sowa, NATO Sharable Software Developing Into True Suite Supporting National Operation, Fire Control Systems, 1-8, Eylül 2008.
- [2] [http://en.wikipedia.org/wiki/SG2_Shareable_\(Fire_Control\)_Software_Suite_\(S4\)](http://en.wikipedia.org/wiki/SG2_Shareable_(Fire_Control)_Software_Suite_(S4))
- [3] Helmut Schmidt Universität, Overview of the NATO Army Armaments Group (NAAG) AC/225 Land Capabilities Group 3 SG/2 Sharable Software Suite (S4), Almanya, Eylül 2006
- [4] Birger Rhe Hansen, Sevsay Aytar Ortaç, André J. Sowa, NATO Testing In Turkey Shows Benefit Of Meteorological Forecast Data To Indirect Fire Support, 1-8, Eylül 2008.
- [5] www.aop-37.org
- [6] John Miller NATO Armaments Ballistic Kernel User Training - NABK Contributors - NABK Coding Standard (NCS), Almanya, Eylül 2006
- [7] John Miller, NATO Armaments Support Services User Training – Autotest, Almanya, Eylül 2006
- [8] D. Schultz. Standard for the Software Life-cycle Process, *ACM SIGSOFT Software Engineering Notes*, 10(3):62–62, 1985.
- [9] I. Sommerville. Software Process Models. *ACM Computing Surveys (CSUR)*, 28(1):269–271, 1996.
- [10] Hsiang-Jui Kung, Quantitative Method to Determine Software Maintenance Life Cycle, 20th IEEE International Conference on Software Maintenance (ICSM'04), 2004

[11] Daisuke Horie, Toshio Kasahara, Yuichi Goto, and Jingde Cheng, A New Model of Software Life Cycle Processes for Consistent Design, Development, Management, and Maintenance of Secure Information Systems, Eighth IEEE/ACIS International Conference on Computer and Information Science, 2009