

YAZILIM ÜRÜN HATTINDA YETENEK MODELİNDEN ÜRÜN KONFIGÜRASYONUNUN OLUŞTURULMASI

Mustafa Özpınar

Aselsan A.Ş. SST-MD-YMM, 06172, Yenimahalle, Ankara

mozpınar@aselsan.com.tr

Özet. Yazılım ürün hattı, belli bir yazılım ürün ailesine mensup yazılımların, bu ürün ailesinin ortaklıkları ve değişkenlikleri göz önünde bulundurularak belirlenen ve önceden oluşturulan yazılım yapıtaşlarının bir araya getirilmesiyle hızlı bir şekilde geliştirilmesi yaklaşımıdır. Ürün hattındaki yazılımlar, ortak özellikleri paylaştıkları gibi, aralarında bazı farklılıklar da vardır. Yetenek modeli, alandaki ortaklıkları ve değişkenlikleri analiz etmek için kullanılan en yaygın tekniklerdendir. Yetenek modelinde, yazılım ürün hattı ile geliştirilmesi planlanan yazılımların içerebileceği en geniş özellikler soyut olarak tanımlanır. Bu özellikler, yazılım tasarımında çoğunlukla birden çok modül tarafından karşılanmaktadır. Yazılım ürün hattındaki varlıklar ve yetenekler arttıkça, ürün hattından geliştirilecek olan ürünlerde, ortak varlıkların yeteneğe bağlı konfigürasyonlarının manuel olarak doğru bir şekilde yapılması da zorlaşmaktadır. Bu bildiride, TADES yazılım ürün hattı kapsamında yetenek modelinden yazılım konfigürasyonunun çıkarılması ile ilgili yapılan çalışmalar anlatılmıştır. Pure::Variants aracında hazırlanmış olan yetenek modeli, Eclipse geliştirme platformundaki model tabanlı dönüşüm altyapısı kullanılarak otomatik olarak ürün konfigürasyonuna dönüştürülmüştür. Bu sayede, ürün hattından çıkan ürünlerdeki konfigürasyon hata sayısı azaltılması ve ürün çıkarılmasının hızlandırılması amaçlanmıştır.

Anahtar Kelimeler: Yetenek Modeli, TADES, Yazılım Ürün Hattı, Model Tabanlı Yazılım Geliştirme

1 Giriş

Hayatın her alanında kullanımı oldukça artan yazılım yoğun sistemlerle birlikte, hızlı ve kaliteli yazılımlar ortaya çıkarmak rekabet açısından hayati önem taşıyor hale gelmiştir. Yazılım geliştirmedeki farklı yöntem arayışları neticesinde ortaya çıkan Yazılım Ürün Hattı, bir ürün ailesi göz önüne alındığında, farklılık ve ortaklıkları belirlenerek geliştirilen yazılım yapıtaşlarının entegre edilerek ürün çıkarılması yaklaşımıdır [1].

TADES, Aselsan'da Teknik Ateş Destek Sistemleri alanında komuta kontrol yazılımlarının geliştirilmesi için oluşturulan bir yazılım ürün hattı çalışmasıdır [2]. TADES ürün hattından şu ana kadar 20'den fazla ürün çıkarılmıştır.

Bu bildiriye, ikinci bölümde problemin tanımı yapılarak motivasyon anlatılmış, üçüncü ve dördüncü bölümlerde yetenek modeli ve aile modeli ile ilgili bilgilendirme yapılarak oluşturulan modellerin detayları belirtilmiştir. Beşinci bölümde modelden yapılan dönüşümler, çıktılar ve bu çıktıların kullanımları anlatılmıştır. Altıncı bölümde ise kazanımlar aktarılmaya çalışılmıştır.

2 Problemin Tanımı

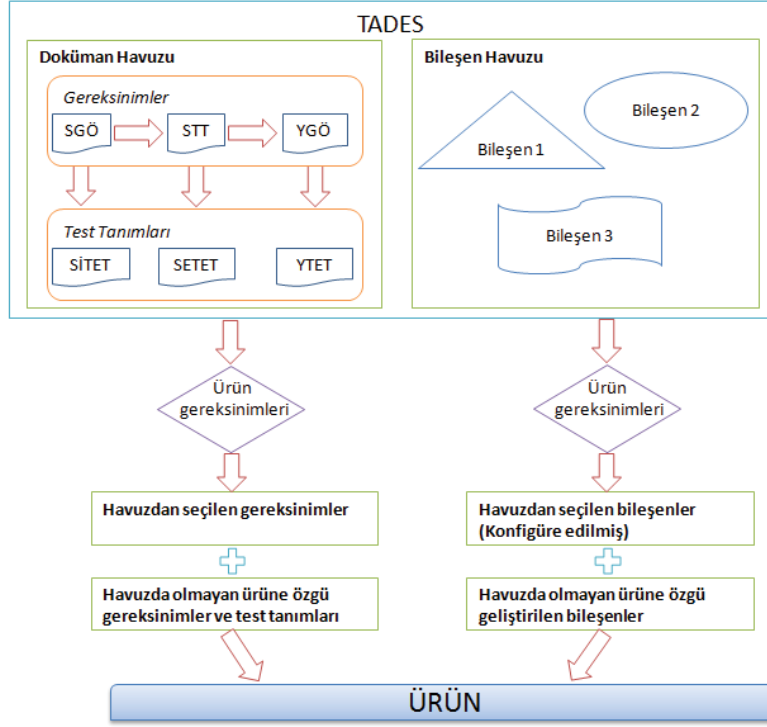
Yazılım Ürün Hattından bir ürün çıkarılırken, daha önceden oluşturulmuş varlık havuzundaki bileşenlerin ürün gereksinimlerine göre seçilmesi ve seçilen bileşenlerin konfigüre edilerek doğru bir şekilde bir araya getirilmeleri gerekir. Bileşen konfigürasyonu, yeteneklere göre bileşen kaynak kodunu yeniden derleme veya yeteneklere göre ayar dosyası ile konfigüre etme gibi yöntemlerle yapılabilir. TADES'te her bileşene ait bir konfigürasyon dosyası mevcuttur. Ürün gereksinimlerine göre bu dosyalarda ayarlamalar yapılarak bileşenin uygun yeteneklerle üründe kullanılması sağlanmaktadır. Havuzdaki bileşen sayısı ve bu bileşenlerin yetenekleri arttıkça, üründe kullanılacak bileşenlerin seçim ve manuel olarak konfigüre edilme karmaşıklığı da artmaktadır. Ayrıca konfigürasyon sırasında hata yapma riski de yüksek olmaktadır.

Şekil 1'de, TADES yazılım ürün hattından bir ürün çıkarılırken izlenen yöntem anlatılmıştır. TADES içerisinde ürün ailesine ait geliştirilen ortak bileşenler (Bileşen Havuzu) ve bu bileşenlere ait dokümanlar (Doküman Havuzu) bulunmaktadır. Doküman Havuzu'nda Aselsan süreçlerine uygun olarak dokümanlar, Bileşen Havuzu'nda ise ürün ailesinin ihtiyaçlarına göre belirlenmiş, geliştirilmiş, farklı ürünlere göre uyarlanabilir (yetenek seçilerek) bileşenler bulunmaktadır.

TADES'ten bir ürün çıkarılması esnasında öncelikle doküman ve bileşen havuzundan ürün gereksinimlerine göre ilgili üründe kullanılacak olan ortak bileşenler ve gereksinimler seçilir. Sonrasında, seçilen bileşenler konfigürasyon dosyaları kullanılarak ürüne göre konfigüre edilir. Ürün hattı bileşen havuzunda yer almayan gereksinimlere göre ürüne özgü dokümanlar oluşturulur ve bileşenler geliştirilir. Son aşamada ise havuzdan seçilen doküman ve bileşenler ile ürüne özgü oluşturulanlar birleştirilerek ürün hazır hale getirilir.

Ürünün oluşturulmasındaki en zor problemlerden birisi ürün gereksinimlerine göre havuzdan seçme ve konfigürasyon aşamasıdır. Bu aşamada yapılacak yanlışlıklar, üründe ihtiyaç olmayan bir yeteneğin üründe bulunması veya tam tersi durumu meydana getirebilir. Ayrıca birbirine bağlı olan bileşenlerin seçimindeki problemler de bileşenlerin tam ve doğru bir şekilde çalışmasını engeller. Örneğin; birbirini gerektiren iki bileşenden birinin seçilip diğerinin seçilmemesi ya da bileşenlerin yanlış konfigüre edilmeleri çıkarılacak ürünün istenilen şekilde çalışmamasına neden olur. Bundan dolayı, bu işlemlerin mümkün olduğunca kullanıcı hatasını sıfıra indirecek

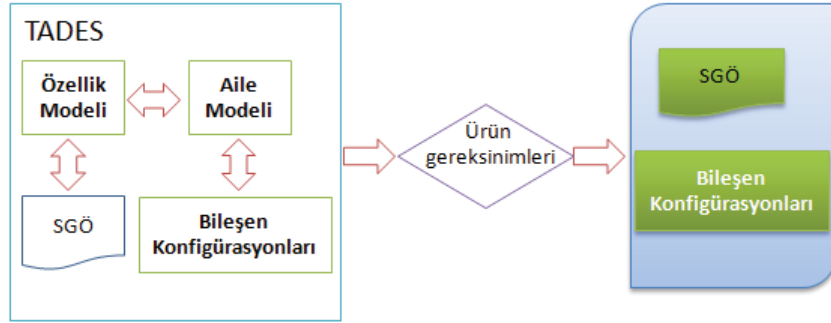
şekilde otomatik olarak yapılması ürün kalitesi, zaman ve müşteri memnuniyeti açısından önem taşımaktadır.



Yazılım ürün hattından ürün çıkarılabilmesi için ortak bileşenlere ait konfigürasyon detaylarının ve kısıtlamaların tanımlanmış olması gerekir. Sürekli değişen gereksinimler ve bileşen güncellemelerinden dolayı bu tanımların yapılması ve manuel olarak idame edilmesi oldukça zordur. Ayrıca bütün kısıtlamaları özümseyerek hatasız bir ürün çıkarmak, havuzdaki yetenek sayısı arttıkça daha fazla çaba gerektirir. Bu aşamada yapılan hataların bir kısmı testlerle bulunurken bir kısmı da ancak müşteri tarafından tespit edilmektedir. Yazılımın geliştirilmesinden sonra ortaya çıkan hatalar, bakım maliyetini yükselttiği gibi müşteri memnuniyetini de azaltmaktadır. Dolayısıyla konfigürasyon otomasyonu ile bileşen konfigürasyonunun hızlı yapılması, üründe konfigürasyondan kaynaklı hata oranını azaltarak geliştirme ve bakım maliyetini azaltma ve yazılım kalitesi ile müşteri memnuniyetinin artırılması hedeflenmektedir.

Pure::Variants [5], Eclipse geliştirme platformuna entegre çalışabilen bir modelleme aracıdır. Bu araç ile yetenek, aile modelleri hazırlanabilmekte, modeller ve model içindeki yetenekler arasında bağlantılar kurularak kısıtlamalar yapılabilmektedir.

Aşağıdaki şekilde Pure::Variants aracı kullanarak yapılan otomasyon çalışması anlatılmaktadır.



Şekil 2. Konfigürasyon otomasyon süreci

Öncelikle, TADES'e ait var olan yetenek modeli düzenlenerek Pure::Variants aracına aktarıldı. Yazılım bileşenlerindeki konfigürasyon dosyalarının modellenmesiyle bir aile modeli oluşturuldu. Daha sonra, aile modeliyle yetenek modeli arasındaki ilişkiler kuruldu ve modelden konfigürasyon dosyalarının üretilebilmesi için gerekli dönüşüm kodları (betikler) yazıldı. Son olarak da seçilen özelliklere göre doküman ve dosyaların üretilmesi sağlandı.

Pure::Variants aracı, yetenek ve aile modelleri oluşturmak için gerekli altyapıyı sağlamaktadır. Fakat alan analizi yapılarak yetenek ağacının oluşturulması, ihtiyaca göre bir aile modeli oluşturulması, yetenek modeliyle aile modeli arasında alan analize göre ilişkilerin kurulması, modelden dosyaların üretilmesi için gereken dönüşüm kodlarının yazılması işlemleri, aracı kullanan tarafından yapılmaktadır.

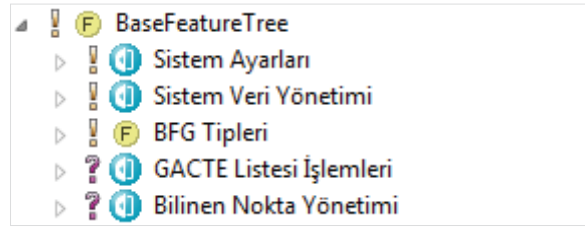
Yetenek modeli, aile modeli ve çıktılara ilişkin detaylar sonraki bölümlerde verilmiştir.

3 Yetenek Modeli

Yazılım ürün hattında yeniden kullanılabilir varlıkların geliştirilebilmesi için alandaki ürünler arasındaki ortaklık ve değişkenliklerin net bir şekilde tanımlanması gerekir. Alandaki ortaklık ve değişkenlikleri analiz etme tekniklerinden en popüler olanlarından bir tanesi de yetenek modelleme tekniğidir [7]. Yetenek modelleri, bir sisteme ait gereksinimleri üst seviye olarak tanımlamak için kullanılır [3]. Yeteneklerin belirlenerek daha küçük parçalar halinde hiyerarşik olarak listelenmesi sonrasında yetenek modeli oluşur. Yetenek modelindeki özellikler opsiyonel, alternatif veya zorunlu olabilirler.

Pure::Variants aracıyla TADES için oluşturulan yetenek modelinin bir kısmı Şekil 3'te görülmektedir. Hiyerarşik olarak oluşturulan bu modelde, yetenekler gruplanarak listelenmiştir. Her bir grubun altında bir veya birden çok yetenek veya yetenek grubu bulunmaktadır. TADES SGÖ dokümanındaki her bir gereksinim, yetenek ağacındaki

en az bir yetenek ile ilişkilendirilmiştir. Pure::Variants aracı, yetenekler arasında farklı ilişkilerin (gerektirme, önerme, çelişme, koşullu gerektirme gibi) kurulmasına imkan vermektedir. Örneğin “Bilinen Nokta listesini yazdırma” yeteneği “Yazdırma” yeteneğinin olmasını gerektirir. Bu kısıtlama yetenek modeli üzerinden yapıldığı takdirde ürün konfigürasyonundaki hatalar minimize edilebilir. Pure::Variants aracı ürün oluşturulurken yetenek seçimi sırasında var olan kısıtlamaları kontrol ederek kullanıcıya uyarı verebilmektedir. Bu sayede daha ürün oluşturulmadan yapılan konfigürasyon hatalarının önüne geçilebilmektedir.



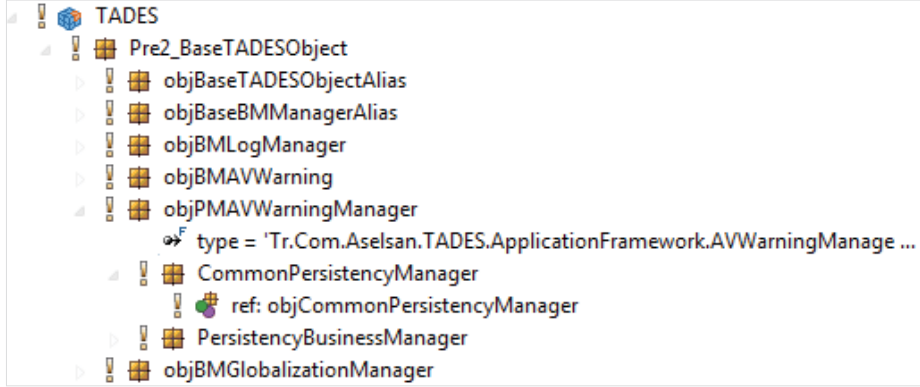
Şekil 3. TADES yetenek modelinin bir bölümü

DOORS [6] aracı ile oluşturulan SGÖ dokümanı Pure::Variants aracına aktarılarak yetenek modelinden yapılan seçim sonrasında sadece ürün tarafından kullanılacak olan gereksinimlerin havuzdan seçilebilmesi sağlanmaktadır. Böylece ürüne özgü sistem gereksinimleri dokümanının otomatik olarak oluşturulması imkânı da ortaya çıkmaktadır.

4 Aile Modeli

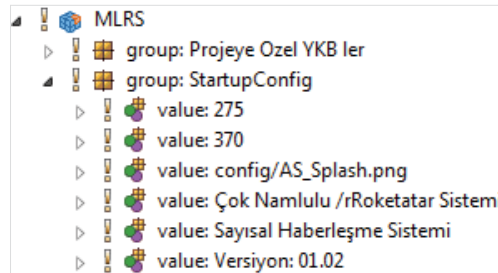
Aile modeli, yazılım bileşenleri açısından çözmek istenilen problemi ifade etmektedir [4]. Yetenek tabanlı alan analizi yapılan durumlarda aile modeli, yetenek modelinin girdi olarak kullanılmasıyla ürün oluşturma sürecindeki bir problemin çözülmesine yönelik oluşturulan modeldir. Örneğin, modelden kod parçaları oluşturulmak isteniyorsa kodun (sınıf, arayüz vs.) modellenmesi, konfigürasyon dosyası üretmek amaçlanıyorsa dosya yapısının modellenmesi gerekir. TADES yazılım ürün hattı kapsamında oluşturulan aile modeli, TADES ortak bileşenlerinin ürüne özgü çalışabilmesi için gereken konfigürasyon dosyalarının modellenmesi ile oluşturulmuştur. Bu modelleme ile çözmek istenilen problem, yazılım ürün hattından bir ürün oluşturulurken yetenek modelindeki seçimlere göre konfigürasyon ayarlarının otomatik olarak en az hata ile oluşturulmasıdır.

Şekil 4’te TADES aile modelinin bir parçası gösterilmiştir. Ağaç yapısındaki model, bileşenlerin konfigürasyonlarını ve aralarındaki ilişkileri içerecek şekilde hazırlanmıştır. Aile modelindeki zorunlu olmayan her bir eleman yetenek modelinde en az bir yetenek ile ilişkili olmak zorundadır. Bu sayede seçilen yeteneğe göre aile modelinden ilgili alanlar Pure::Variants tarafından kısıtlamalar göz önüne alınarak otomatik olarak seçilmektedir.



Şekil 4. TADES aile modeli

Ürün konfigürasyonunda, TADES bileşenlerine ait konfigürasyonlar ile ürüne özgü geliştirilen bileşenlerin konfigürasyonları iç içe geçmek zorundadır. Örneğin bir TADES bileşeninin ihtiyaç duyduğu servisi ürüne ait özel geliştirilen bir bileşen sağlayabilir. Bundan dolayı her bir ürün için bir aile modeli oluşturulması gerekmektedir.



Şekil 5. Ürün aile modeli

Şekil 5'te, bir ürüne özel oluşturulan aile modeli gösterilmektedir. Bu modelde, ürüne özgü konfigürasyon elemanları yer almaktadır. Dönüşüm sırasında, TADES aile modeli ve ürüne özgü aile modeli birleştirilerek konfigürasyon dosyaları oluşturulmaktadır.

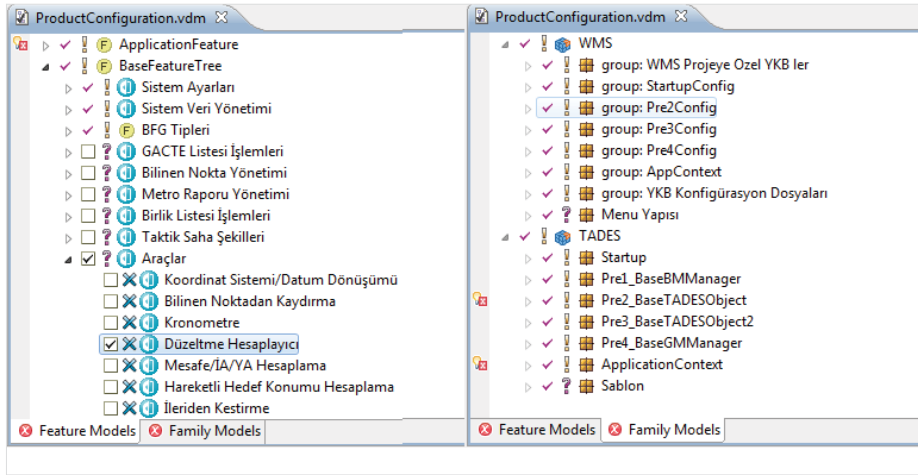
5 Dönüşüm, Çıktılar ve Kullanımları

Ürüne ait dosyaların ve gereksinimlerin oluşturulabilmesi için gerekenleri şu şekilde listeleyebiliriz:

- TADES yetenek modeli
- TADES aile modeli
- Ürün aile modeli
- Ürün gereksinimleri
- Dönüşüm betik dosyaları

Pure::Variants aracı yetenek ve aile modelleri üzerinden yapılan seçimlere göre ihtiyaç duyulan dosyaların üretilebilmesi için gerekli dönüşüm işlemlerine ait “betik” altyapısını sunmaktadır. Çalışma kapsamında, bu altyapı kullanılarak yazılan özel betik dosyaları ile araç üzerinden oluşturulan bir ürün için gerekli dönüşümler yapılabilmektedir.

Bir ürün için öncelikle Pure::Variants aracı üzerinden kullanılacak modeller seçilerek yeni bir “çözüm uzayı (configspace)” ve “variant” oluşturulur. Bu “variant”ta yetenek modeli üzerinden üründe olacak yetenekler seçilir. Araç, yetenek modelinde yapılan seçimlere göre otomatik olarak aile modelindeki seçimleri yapar. Seçim işlemi bitince “betik” ler ile dönüşümler yapılarak konfigürasyon dosyaları oluşturulur. Yetenek modelindeki seçimler sonrasında ürüne ait gereksinim dokümanında da ilişkili alanlar seçilir ve DOORS aracı ile bu seçimler göz önüne alınarak ürün gereksinim dokümanı oluşturulur. Şekil 6’da örnek olarak oluşturulmuş bir ürüne ait Yetenek Modeli ve Aile Modeli gösterilmektedir.



Şekil 6. Pure::Variants aracı ile oluşturulan bir ürüne ait yetenek ve aile modelleri

6 Sonuç

Yazılım ürün hattında, ortak bileşen havuzundaki bileşenlerin bir üründe kullanılabilmesi için, üründe talep edilen yeteneklere göre konfigüre edilebilmeleri gerekmektedir. Konfigürasyon işleminin manuel yapılması, artan bileşen ve yetenek sayısı ile ilgili olarak karmaşık hale gelmektedir. Bundan dolayı, daha kaliteli ve sorunsuz

ürünler çıkarabilmek için bileşen konfigürasyonlarının mümkün olduğunca otomatik olarak oluşturulması bir ihtiyaçtır. Bu bildiride, Aselsan'da geliştirilen TADES yazılım ürün hattı kapsamında, konfigürasyon ve gereksinim dokümanlarının otomatik olarak oluşturulabilmesi için yapılan çalışmalar aktarılmaya çalışılmıştır. Şu aşamada yetenek modelinin oluşum süreci tamamlanmış, aile modeli ise büyük bir oranda hazır hale getirilmiştir. Yetenek modelinin gereksinim dokümanı ile bağlantıları tamamıyla kurulmuş durumdadır. Bununla birlikte, modelin kendi içerisindeki ve diğer modellerle olan ilişkilerin kurulması henüz başlangıç aşamasındadır. Bundan sonraki süreçte, ilişkilerin tamamlanması ve ürün hattındaki tüm ürünlerin konfigürasyonlarının otomatik olarak yapılması hedeflenmektedir.

Yazılım ürün hattından çıkarılan bir üründe, havuzdaki bileşenlerle ürüne özgü bileşenler birlikte kullanılmak durumundadır. Örneğin, havuzdaki bileşenin ihtiyaç duyduğu konum bilgileri ürüne özgü bileşen tarafından sağlanıyor olabilir. Bundan dolayı, konfigürasyon dosyalarının üretilmesi sırasında ürün aile modeli ile TADES aile modelinin birlikte ele alınarak dönüşüm işleminin yapılması gerekmektedir. Pure::Variants aracı bu konuda herhangi bir altyapı sunmamaktadır. Ayrıca aile modelinde kullanılacak değerlerin aile modelinde yeniden tanımlamaya gerek kalmadan yetenek modelinden alınabilmesi yeteneği yoktur. Araç seçiminde bu tür kısıtların olduğunu göz önünde bulundurmak, geliştirme aşamasında bekleyen zorluklar için yol gösterici olacaktır.

Kaynakça

1. Clements, P., Northrop, L.: Software Product Lines: Practices and Patterns, Pearson Education (Addison-Wesley), ISBN 0-201-70332-7, (2001)
2. Barak, E., Erdem, S., Yılmaz, H., TADES:Komuta Kontrol Alanında bir Yazılım Ürün Hattı Çalışması, UYMK 2010.
3. Linden, F., Phol, K., Bockle, Gunter, B.: Software Product Line Engineering: Foundations, Principles, and Techniques. Springer, Heidelberg (2005)
4. Software Product Line Engineering with Feature Models, <http://www.pure-systems.com/fileadmin/downloads/pure-variants/tutorials/SPLWithFeatureModelling.pdf>
5. pure::variants, http://www.pure-systems.com/pure_variants.49.0.html
6. IBM-Rational DOORS, <http://www.ibm.com/software/products/tr/ratidoor>
7. Lee, K., Kang, K.C., Lee, J.: Concepts and guidelines of feature modeling for product line software engineering. In: Gacek, C. (ed.) ICSR 2002. LNCS, vol. 2319, pp. 62–77. Springer, Heidelberg (2002)