

Scrum Metodu Kullanılarak Bir Mobil Uygulama Geliştirme Sürecinin Gerçekleştirilmesi

Mehmet ELİBOL¹*, Çiğdem SELÇUKCAN EROL²

¹Bilgisayar ve Öğretim Teknolojileri Eğitimi Bölümü, Marmara Üniversitesi, İstanbul, Türkiye

²Enformatik Bölümü, İstanbul Üniversitesi, İstanbul, Türkiye

mehmet.elibol@marmara.edu.tr, cigdemsel@istanbul.edu.tr

(Geliş/Received:23.08.2016; Kabul/Accepted:04.04.2017)

DOI: 10.17671/gazibtd.309299

Özet— Bu çalışmanın amacı, Scrum metodu kullanılarak geliştirilen bir mobil uygulama sürecindeki deneyimlerden yola çıkılarak bu süreçle ilgili önerilerde bulunmaktır. Bu kapsamda klinik araştırmalar alanında ihtiyaç duyulan bir mobil uygulamanın, Scrum metodu ile gerçekleştirilme süreci detaylı bir şekilde anlatılmaktadır. Bu süreçte ön plana çıkan noktalar, projeye başlamak için ekiplerin nasıl oluşturulduğu, müşterinin ihtiyaçlarının nasıl tespit edildiği, süreç içinde değişen müşteri isteklerinin nasıl yönetildiği, bu süreçte şeffaflık, denetim ve adaptasyonun nasıl gerçekleştirildiği Scrum sürecindeki kavramlar (Koşu, Ürün İçeriği, vb.) baz alınarak detaylandırılmıştır.

Anahtar Kelimeler— Mobil Uygulama Geliştirme, Çevik Metodlar, Scrum

Implementation of a Mobile Application Development Process Using Scrum Method

Abstract— The purpose of this study is to suggest the process from the experiences of a mobile application developed using the Scrum method. In this context, the process of carrying out a mobile application, which is needed in the field of clinical researches, with the Scrum method is explained in detail. The concepts in the Scrum process (Sprint, Product Backlog, etc.) are based on the points that are at the forefront of this process, how the teams are set up to start the project, how customer needs are identified, how customer requests are changed in the process and how transparency.

Keywords— Mobile Application Development, Agile Methods, Scrum

1. GİRİŞ (INTRODUCTION)

Günümüzde mobil cihaz kullanımının artması ile birlikte mobil uygulama kullanımı konusunda da artış olmuş ve mobil uygulamalara olan ihtiyaç artmıştır. İhtiyaç duyulan mobil uygulamaların geliştirilmesi sırasında, mobil cihazların cihaz boyutu, ekran boyutu ve veri giriş mekanizması için fiziksel özelliklerinin olmasının yanı sıra işlem gücü, bellek alanı ve işletim sistemi gibi teknik karakterleri de olduğu dikkat edilmesi gereken noktalar [1]. Mobil uygulamaların tasarımı, değişen ihtiyaç ve beklentilerle birlikte [2], yazılım geliştirme projelerinde görünenden çok daha karmaşıktır, bu durum geliştiricileri geleneksel yazılım geliştirme metodolojilerini tekrar gözden geçirmeye zorlamaktadır [3].

Her gün binlerce uygulamanın ortaya çıktığı pazarda ihtiyaç tespit edilerek geliştirilmeye başlayan uygulama bir başka kişi ya da kurum tarafından hazırlanarak hayata geçirilebilir. Uygulama başka firma tarafından hayata geçirildikten sonra uygulamamız kullanılamaz konuma

düşebilmektedir. Bu da uygulama için harcanan zaman ve paranın boşa gitmesi anlamına gelmektedir. Görüldüğü gibi mobil uygulama geliştirme alanında ihtiyacı tespit ederek anında uygulama geliştirmeye başlamak önemli hususlardan birisidir. Bu ihtiyaçları en doğru şekilde tespit edebilmek için müşterinin ne istediğini tam olarak anlamak çok önemlidir. Hızlı gelişen ve büyüyen yazılım dünyasında ihtiyaçları tam olarak tespit edebilmenin bazı zorlukları vardır. Proje başında tespit edilen ihtiyaçlar değişen ve gelişen teknoloji ile birlikte proje sonunda değerini yitirebilmektedir ve ihtiyaç olmaktan çıkmakta hatta gereksiz konuma düşebilmektedir. Bundan dolayı ihtiyaç tespit edildikten sonra müşterinin ihtiyaçlarının ürünün her aşamasında değişebileceği göz önüne alınarak sürekli müşteri ile irtibat halinde bulunmak [4] önem arz eden diğer bir konu olmaktadır. Tüm bu hususlar mobil uygulama geliştirme sürecinde geleneksel yazılım geliştirme metodolojilerinin sınırları olmakta ve bizleri çevik metodolojilere yönlendirmektedir.

2. ÇEVİK YAZILIM GELİŞTİRME METODOLOJİSİ (AGILE SOFTWARE DEVELOPMENT METHODOLOGY)

Çevik Yazılım Geliştirme Manifestosu, projede çalışan bireyler ve aralarındaki etkileşimi ön plana çıkararak, süreçte sürekli müşteri ile işbirliği içinde değişikliklere yanıt verilmesi gerektiğini ve bu süreçte yazılımın sürekli çalışır olması gerektiğini savunan bir yazılım geliştirme metodolojisidir [4].

Çevik metodolojiler, şirketlerin doğru ürün oluşturmaya yardımcı olmak için [2] çevre ve gereksinimlerdeki değişikliklere uyum sağlamayı kolaylaştırmaktadır. Buna ek olarak süreç içindeki geliştirme tecrübelerinden yeni öğrenmeler sağlayarak yazılım geliştirme esnekliğini ve üretkenliğini artırmakta [3] ve şirketlerin pazarda mümkün olduğunca rekabetçi olmasını sağlamaktadır [2]. Örneğin; elektronik imzanın geliştirilmesinden sonra yazılım şirketleri elektronik imza ile işlem yapabilen sistemleri hayata geçirmeye başlamışlardır. Elektronik imza ile çalışan sistemler yaygınlaştıkça elektronik imza ile işlem yapan kişi sayısı da artmaya başlamıştır. Bu süreç devam ederken, teknolojiye gelişmelerle birlikte mobil imza kavramı ortaya çıkmıştır. Mobil imza kavramının ortaya çıkması ile birlikte elektronik imza ile çalışan projelerin revize edilmesi gerekmiştir [5]. Bu noktadan hareketle, görüldüğü gibi yazılım projeleri başka alanlardaki projelere kıyasla değişime daha duyarlıdır. Bu açıdan ele alındığında yazılım projeleri her zaman değişime ve gelişime açık olmalıdır.

3. MOBİL UYGULAMA GELİŞTİRME SÜREÇLERİNDE UYGULANAN ÇEVİK YAKLAŞIMLAR (AGILE APPROACHES FOR MOBILE APPLICATION DEVELOPMENT PROCESS)

Birçok çevik yazılım geliştirme metodu bulunmaktadır. Mobil uygulama geliştirme sürecinde kullanılan çevik yaklaşımlarla ilgili bilgi aşağıda yer almaktadır:

MOBİL-D mobil yazılım ürünleri ve servisleri geliştiren üç firma ile işbirliği içinde geliştirilmiştir. Mobile-D yaklaşımı kısa süre içinde tam fonksiyonlu bir mobil uygulamayı yayına hazır hale getirmek için bir ofis ortamında çalışan on kişiden daha az geliştiriciden oluşan bir ekip için optimize edilmiştir [6].

MASAM bir mobil uygulamayı hızlı şekilde geliştirmek için çevik yazılım geliştirme ilkelerini uygulayan bir yaklaşımdır [7].

Scharff ve Verma [1] Pace Üniversitesinde gerçek bir kullanıcı için mobil uygulama geliştirme sürecinde profesyonel bir Scrum Uzmanı ile birlikte Scrum Metodunu uygulamıştır.

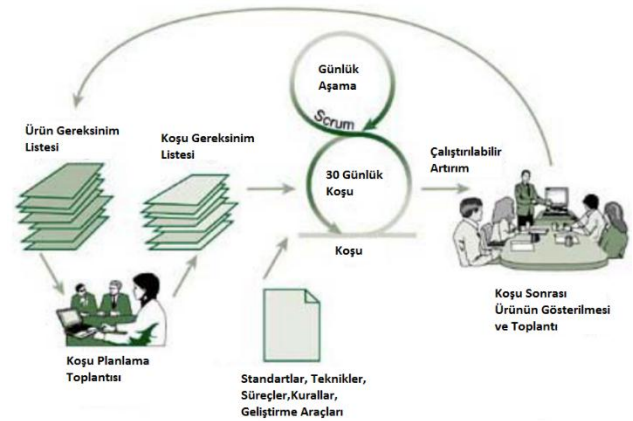
Bu çalışma kapsamında sık kullanılan metodolojilerden biri olan Scrum tercih edildiğinden sadece Scrum çevik yazılım geliştirme metodu ile ilgili bilgi verilmektedir.

3.1. Scrum (Scrum)

Schwaber ve Sutherland (1996) tarafından tanıtılan, hala desteklenen ve Beedle'nin işbirliği ile geliştirilen Scrum karmaşık ürün geliştirmeyi yönetmek için bir süreç çerçevesi sunmaktadır [8, 9]. Scrum'un mobil sistemlere uygulanması ilk defa [1] tarafından bir sınıf ortamında gerçekleştirilmiştir.

Scrum'un merkezinde geliştirme sürecinde ortaya çıkacak süreçlerin tahmin edilemez olduğu fikri yer alır. Bu nedenle, esnek bir şekilde yazılım geliştirmenin nasıl olabileceği üzerine yoğunlaşmakta [10] ve geliştirme aşaması için herhangi bir özel yazılım geliştirme tekniği tanımlanamamaktadır [11]. Scrum anlaşılması kolay, karmaşık olmayan ama uzmanlaşması zor bir süreçtir. Bu süreçte öngörülebilirliği arttırmak ve riski kontrol etmek için yinelemeli ve artırımı bir yaklaşım kullanılır [9]. Artırımı bir yaklaşım kullanarak ürün aşama aşama inşa edilirken bir sonraki ürün parçası için öngörülebilirliğin artmasına bağlı olarak süreç içinde risklerin kontrol edilebilirliği de artmaktadır. Örneğin; yüz tanıma ile bir işlem yapacak mobil uygulama geliştirme aşamasında, ilk olarak yüz tanıma özelliği sistem üzerinde gerçekleştirilebilir. Daha sonraki aşamada detay işlemlerin nasıl gerçekleştirileceği artırımı bir şekilde sisteme eklenebilir.

Scrum süreci belirli bir düzen ile kendi içinde Koşu, Ürün Gereksinim Listesi, Koşu Gereksinim Listesi, Koşu İş Takip Grafiği, Koşu Hedefi ve Kullanıcı Hikayeleri gibi birçok kavram barındırmaktadır (Şekil 1).



Şekil 1. Scrum Süreci (Scrum Process) [8]

Scrum'da yer alan bu kavramlar, Scrum Sürecindeki Genel Kavramlar, Scrum Sürecindeki Roller, Scrum Sürecindeki Toplantılar olarak özetle aşağıdaki gibi tanımlanmaktadır:

3.1.1.Scrum Sürecindeki Genel Kavramlar (General Terms for the Scrum Process)

Koşu (Sprint): 1 haftadan 4 haftaya kadar süreleri olan yinelemeli periyotlar “Koşu” olarak adlandırılmaktadır [1].

Ürün Gereksinim Listesi (Product Backlog): Temel olarak gereksinimlerin, Kullanıcı Hikâyelerinin ve özelliklerin bir listesidir [12] ve üründe yapılacak herhangi bir değişiklik için gereksinimlerin kaynağı olan bir liste şeklindedir [9].

Koşu Gereksinim Listesi (Sprint Backlog): Geliştirme Ekibinin “Koşu Hedefini” gerçekleştirmek için yapmaları gerekenleri tanımlayarak tüm işlerin görünür olmasını sağlayan bir liste şeklindedir [9].

Koşu İş Takip Grafiği (Sprint Burndown Chart): Bu grafik, ekibin her görevi tamamlaması için verdiği saat gün taahhüdünü gösterir ve ekibin ilerlemesi için bir ölçü olarak Scrum Uzmanı tarafından takip edilir [1].

Koşu Hedefi (Sprint Goal): Koşu Hedefi, Ürün Gereksinim Listesinin gerçekleştirilmesi için Koşu içinde uygulanacak olan ve Geliştirme Ekibine ürünü niçin geliştirdikleri konusunda rehberlik sağlayan bir amaçtır [9]. Koşu Hedefinin cevaplaması gereken temel soru “Niçin bu koşuyu gerçekleştiriyoruz” şeklindedir [12].

Kullanıcı Hikayeleri (User Stories): Gereksinimler genellikle Kullanıcı Hikayeleri oluşturularak açıklanmaktadır. Kullanıcı Hikayeleri müşterinin anlatımı ile oluşturulan, Ürün Sahibinin ne istediğini [1] ve sonuçta görmek istediğini ifade eden [12] ürün ile kullanıcının etkileşiminin basit bir açıklamasını içeren [13] bir liste şeklindedir. Kullanıcı Hikayeleri projede yapılacak görevleri ifade etmez, bu yüzden proje sürecindeki kişilerin ne yapacağını belirten görevlerden oluşmamaktadır [1].

3.1.2.Scrum Sürecindeki Roller (Roles in Scrum Process)

Scrum Ekibi; Ürün Sahibi, Scrum Uzmanı ve Geliştirme Ekibi olmak üzere 3 ana rolden oluşmaktadır [1]:

Ürün Sahibi (Product Owner): Geliştirilecek ürüne ihtiyacı olan ve işin gerekliliklerini başarmaktan sorumlu kişidir. Ürün Sahibi, Ürün Gereksinim Listesinde detaylandırılan ürünün vizyonunun [1], sorumluluğuna sahiptir [1]. Proje süreci ilerledikçe bu vizyon netleşir [14].

Geliştirme Ekibi (Development Team): İşlevselliğin geliştirilmesi sorumluluğuna sahip [14] ve yazılımı geliştiren 3 kişiden 15 kişiye kadar olan bir ekiptir [1].

Scrum Uzmanı (Scrum Master): Süreç boyunca takımı yönlendiren ve sorunları çözmek için ekibe yardımcı olan kişidir [1].

3.1.3.Scrum Sürecindeki Toplantılar (Meetings in the Scrum Process)

Koşu Planlama Toplantısı (Sprint Planning Meeting): Koşu boyunca gerçekleştirilecek olan işlerin planlandığı toplantıdır [9].

Günlük Scrum Toplantısı (Daily Scrum Meeting): Geliştirme Ekibinin gelecek 24 saat için bir plan oluşturması ve aktivitelere senkronize olmalarını sağlamak için 15 dakikadan oluşan olaylardır. Bu toplantılar ile bir önceki toplantıdan sonra yapılan çalışmalar görülür [9], yapılan iş miktarı [13] ve bir sonraki toplantıya kadar yapılacak çalışmalar tahmin edilir [9] ve bu şekilde ekip üyelerinin proje sürecindeki bilgiler hakkında fikir sahibi olmaları sağlanır [15].

Koşu Değerlendirme Toplantısı (Sprint Review Meeting): Gerçekleştirilen ürünü incelemek ve ihtiyaç duyulan Ürün Gereksinim Listesini düzenlemek için her Koşu sonunda gerçekleştirilen toplantılardır. Bu toplantı sırasında, Scrum Ekibi ve paydaşları Koşu sürecinde yapılan işler hakkında işbirliği içinde tartışmaktadır [9] ve paydaşlar Geliştirme Ekibinin ne yapıyor olduğunu öğrenmektedir [12].

Koşu Süreç Değerlendirme Toplantısı (Sprint Retrospective Meeting): Scrum Ekibinin kendisini değerlendirmesi için bir fırsat oluşturan ve sonraki Koşu sürecini iyileştirmek için plan oluşturulmasını sağlayan toplantıdır [9].

Bu çalışma kapsamında, Scrum metodu kullanılarak bir mobil uygulama geliştirme süreci gerçekleştirilmiştir.

4. MATERYAL VE METOD (MATERIALS AND METHODS)

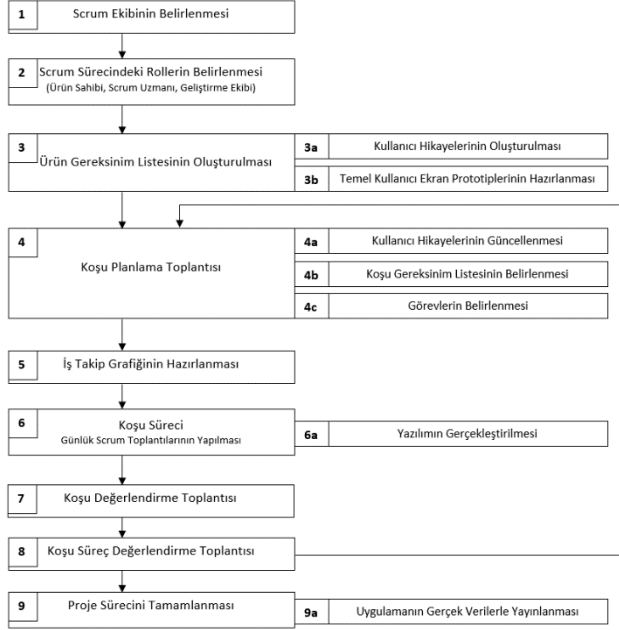
Scrum (2014) metodu ile geliştirilen mobil uygulama; verilerin sağlandığı veritabanı, veritabanı ile uygulama arasında iletişimi sağlayan servisler ve mobil uygulama olmak üzere üç kısımdan oluşmaktadır.

Veritabanı oluşturma kısmında, Microsoft SQL Server Management Studio platformu kullanılmıştır. Mobil uygulama geliştirme sürecinde Eclipse platformu üzerinde Java Tabanlı Android Yazılım Geliştirme dili kullanılmıştır. Bu iki aracın birbiri arasında iletişimi sağlayacak servislerin oluşturulması aşamasında ise Visual Studio platformu üzerinde C# ve Asp.Net yazılım geliştirme dili kullanılmıştır. Ayrıca süreç yönetimi için, Mock Up Builder [16] ve Ontime Scrum Planlama [17] yazılımlarının deneme sürümleri kullanılmıştır.

5. MOBİL UYGULAMA GELİŞTİRME SÜRECİNDE SCRUM'IN UYGULANMASI (IMPLEMENTATION OF SCRUM IN MOBILE APPLICATION DEVELOPMENT PROCESS)

Bu çalışma kapsamında; İstanbul Üniversitesi, İstanbul Tıp Fakültesi İlaç Araştırmaları Birimi için Android

tabanlı bir mobil uygulama Scrum metodu ile geliştirilmiştir. Çevik Metodolojiler içerisinde, geliştirilecek mobil uygulama için yazılım geliştirme sürecinde müşteri ihtiyaçlarını iyi analiz etmeyi sağlayan ve proje sürecinde herkesin aktif olmasına imkan veren Scrum metodu tercih edilerek, uygulama geliştirme sürecine başlanmıştır. Bu süreçte Scrum metodunun özellikleri kullanılarak mobil uygulama geliştirme süreci tamamlanmıştır. Bu çalışma kapsamında uygulanan Scrum metodu ile mobil uygulama geliştirme süreci Şekil 2’de özetlenmiştir.



Şekil 2. Scrum Metodu ile Mobil Uygulama Geliştirme Süreci (Mobile Application Development Process with Scrum Method)

Scrum metodu ile proje süreci başlarken ilk aşamada Scrum Ekibi oluşturulmuştur (1). Scrum Ekibinin oluşturulmasından sonra uygulama geliştirme sürecinde Ürün Sahibi, Scrum Uzmanı ve Geliştirme Ekibi’ nin kimler olacağı belirlenmiştir (2). Bu aşamadan sonra Ürün Sahibi ile birlikte çalışarak, Ürün Sahibinin ihtiyaçları ve analizleri değerlendirilmiş ve Ürün Gereksinim Listesi belirlenerek (3), Kullanıcı Hikayeleri oluşturulmuş (3a) ve temel kullanıcı ekran prototipleri hazırlanmıştır (3b). Sürekli güncellenen Ürün Gereksinim Listesinden öncelikli olarak hangi ihtiyaçların karşılanacağını belirlemek için Koşu Planlama Toplantıları gerçekleştirilmiş (4), Ürün Sahibinin ihtiyaçlarını ifade eden Kullanıcı Hikayeleri güncellenmiş (4a), güncellenen Kullanıcı Hikayeleri seçilerek Koşu Gereksinim Listesi belirlenmiş (4b) ve Kullanıcı Hikayelerine göre görevler oluşturulmuştur (4c). Koşu Gereksinim Listesinin belirlenmesinden sonra Scrum Uzmanı tarafından, yapılan işlerin takibini kolaylaştırmak ve zaman içindeki performansı görmek amacıyla İş Takip Grafiği hazırlanmıştır (5). Bu aşamalardan sonra Koşu Süreci (6) başlatılmıştır. Ekibin kendi kendine organize olmasını sağlayan Günlük Scrum Toplantıları ile Koşu süreci

desteklenerek yazılım ve uygulama gerçekleştirilmiştir (6a). Bu süreç sonunda Koşu Değerlendirme Toplantısı (7) yapılarak uygulamanın o ana kadar yapılan kısmı sunulmuş ve geri bildirimler alınmıştır. Koşu Süreç Değerlendirme Toplantısında (8), süreç içinde Geliştirme Ekibini olumlu veya olumsuz etkileyen yönler değerlendirilmiştir. Koşu süreci yinelemeli olarak 3 kez gerçekleştirildikten sonra proje süreci tamamlanmıştır (9) ve ürün uygulama marketinde yayınlanmıştır (9a).

5.1. Ürün Gereksinim Listesinin Oluşturulması (Creating Product Backlog) (Şekil 2,3)

Ürün Sahibi önceden hazırladığı dokümantasyon üzerinde istediği ürünü genel olarak ifade ederken, Scrum Uzmanı ve Geliştirme Ekibi sorular sorarak Kullanıcı Hikayelerinin (Figure 2, 3a) oluşturulmasını sağlamıştır. Oluşturulan Kullanıcı Hikayelerinden örnekler şu şekildedir;

•Ben bir kullanıcı olarak, uygulamayı ilk açtığımda İstanbul Tıp Fakültesi yazan bir ekranla karşılaşmak istiyorum.

•Ben bir kullanıcı olarak, giriş ekranını geçtikten sonra, Konular, Öğretici Sorular ve Sık Sorulan Sorular şeklinde olan seçim yapabileceğim bir ekranın olmasını istiyorum.

5.2. Kullanıcı Ekran Prototiplerinin Hazırlanması (Creating User Interface Prototypes) (Şekil2, 3b)

Ürün sahibinin bu isteklerine göre Geliştirme Ekibi tarafından MockUp Builder aracı kullanılarak, Şekil 3’de görüldüğü gibi yazılımın tasarımsal prototipi hazırlanmıştır.

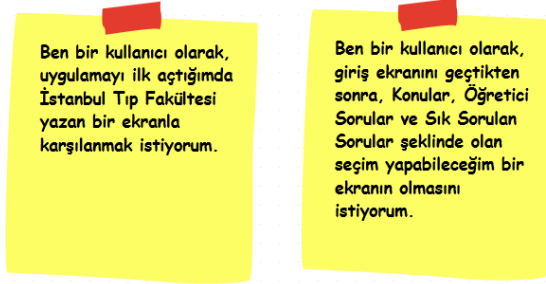


Şekil 3. Mobil uygulamanın tasarımsal prototip örnekleri (Examples of Conceptual Prototype to the Mobile Application)

5.3. Birinci Koşu Planlama Toplantısı (First Sprint Planning Meeting) (Şekil 2, 4)

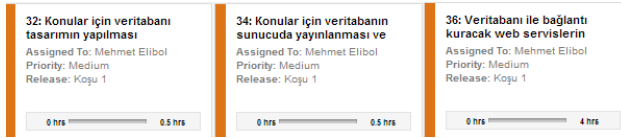
Bu toplantı öncesi Scrum Uzmanı ve Geliştirme Ekibi, Ürün Gereksinim Listesi üzerinde çalışarak Koşu Planlama Toplantısı öncesi hazırlık yapmışlardır.

Bu toplantı sırasında Ürün Gereksinim Listesinden Kullanıcı Hikayeleri güncellenmiş (Şekil 2, 4a) ve bu Kullanıcı Hikayelerinden (Şekil 4) hangilerinin Koşu sürecinde gerçekleştirileceğine karar verilerek Koşu Gereksinim Listesine (Şekil 2, 4b) aktarılmıştır.



Şekil 4. Birinci Koşu için Kullanıcı Hikayeleri Örnekleri (Examples of User Stories for the First Sprint)

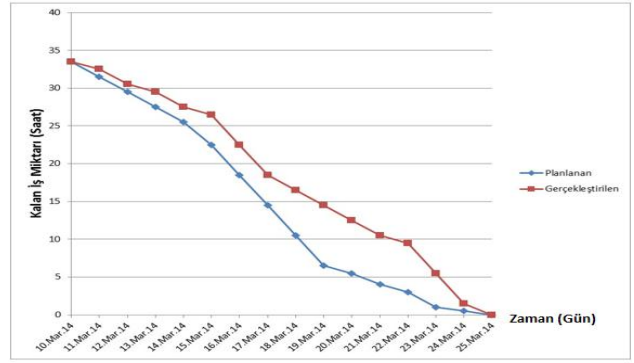
Kullanıcı Hikayelerini gerçekleştirilmek için yapılması gereken görevler (Şekil 2, 4c) ve bu görevlerin ne kadar birimde¹ gerçekleştirileceği Şekil 5'de yer alan örnekte görüldüğü gibi planlanmıştır.



Şekil 5. Birinci Koşu için Gerçekleştirilmesi Gereken Görevler (Tasks to be performed for the First Sprint)

5.4. Scrum Uzmanı Tarafından İş Bitim Grafiğinin Hazırlanması (Preparation of Sprint Burndown Chart by Scrum Master) (Şekil 2, 5)

Scrum Uzmanı tarafından Koşu sırasında işlerin ne kadarının yapıldığının rahat bir şekilde takip edilebilmesi için Koşu başlangıcında İş Takip Grafiği (Şekil 6) oluşturulmuş ve Geliştirme Ekibi ile paylaşılmıştır. Bu grafik üzerinde saat bazında yapılan işler çizilerek zamanlama hedefi ile ne kadar doğru orantılı olarak gidildiğinin görülmesi sağlanmıştır.



Şekil 6. Birinci Koşu Sırasında Takip Edilen İş Takip Grafiği (Sprint Burndown Chart for the First Sprint)

5.5. Birinci Koşu Süreci (First Sprint Process) (Şekil 2, 6)

Birinci Koşu sürecinde; veri tabanı tasarımı gerçekleştirilmiş ve sunucuya aktarılmış, web servisler oluşturulmuş ve sunucuya aktarılmış, canlı ortamda çalışan servisler aracılığıyla, tekrar kullanılabilir yapılar kullanılarak, uygulama geliştirme sürecine başlanmıştır (Şekil 2, 6a).

Birinci Koşu sürecinde ortaya çıkan bazı önemli noktalardan bir tanesi şu şekildedir;

- Ürün Sahibinden gelen dönütler sonucunda yönlendirme ekranında İletişim ve Uygulama Hakkında seçeneklerinin olmasına karar verilmiştir.

5.6. Birinci Koşu Sürecinde Günlük Scrum Toplantıları (Daily Scrum Meetings during First Sprint Process) (Şekil 2, 6)

Günlük Scrum Toplantıları gerek yüz yüze olsun, gerek iletişim teknolojilerinin getirdiği yeniliklerden yararlanılarak çeşitli yöntemlerle gerçekleştirilmiştir.

5.7. Birinci Koşu Değerlendirme Toplantısı (First Sprint Review Meeting) (Şekil 2, 7)

Geliştirme Ekibi tarafından uygulamanın o ana kadar tamamlanmış kısmının sunumu yapılmış ve Ürün Sahibinden dönütler alınmıştır. Örneğin;

- Uygulamanın ilk açıldığı ekranda, logo, belirli bir süre (3-5 saniye kadar) ekranda kaldıktan sonra Devam butonuna gerek kalmadan da yönlendirme ekranına geçilebilmelidir.

¹ Koşu süresince iş hacmi birim kavramı ile ifade edilmektedir. Uygulama geliştirme süresince birim kavramına karşılık olarak iş, saat olarak ifade edilmiştir.

5.8. Birinci Koşu Süreci Değerlendirme Toplantısı (First Sprint Retrospective Meeting) (Şekil 2, 8)

Bu toplantı sırasında üzerinde durulan noktalara ait örneklerden bir tanesi aşağıdaki şekildedir;

- Zaman Yönetimi konusunda ekibin daha dikkatli olması gerektiği üzerinde durulmuştur.

5.9. İkinci ve Üçüncü Koşu (Second and Third Sprints)

İkinci ve Üçüncü Koşu süreçleri kısaca aşağıdaki şekilde gerçekleşmiştir. Detayları bir önceki bölümde anlatılan birinci koşu sürecine benzerlik göstermektedir.

İkinci Koşu süreci 8 günlük sürede, her gün Günlük Scrum Toplantıları gerçekleştirilerek, Kullanıcı Hikayeleri temel alınarak hazırlanmış Görev Kartları ve Scrum Uzmanı tarafından hazırlanan İş Takip Grafiği takip edilerek gerçekleştirilmiştir. İkinci Koşu süreci Koşu Değerlendirme Toplantısı ve Koşu Süreç Değerlendirme Toplantısı gerçekleştirilerek tamamlanmıştır.

Üçüncü Koşu süreci 10 günlük sürede, İş Takip Grafiği takip edilerek ve Görev Kartlarındaki görevler gerçekleştirilerek tamamlanmıştır. Üçüncü Koşu sonunda Geliştirme Ekibi uygulamanın son halinin sunumunu yapmıştır ve artık uygulamanın gerçek verilerle Google Markette yayınlanmasına karar verilmiştir.

5.10. Ürünün Yayınlanması ve Proje Sürecinin Kapatılması (Launching the Production to the Market and Closing the Project Process) (Şekil 2, 9)

Ürün Sahibinden gelen gerçek veriler sisteme girildikten sonra, Ürün Sahibinin ve kullanıcıların gerçek verilerle sistemi test etmeleri sağlanmıştır. Gerçek verilerle kullanım sonucunda oluşan geri bildirimler doğrultusunda uygulama üzerinde güncellemeler yapılmıştır. Ürün Sahibinden onay alınarak uygulama Google Markette yayınlanmıştır (Şekil 2, 9a).

6. TARTIŞMA VE SONUÇ (DISCUSSION AND RESULT)

Bu çalışma sürecinde değişime en iyi şekilde cevap verebilmek amacıyla geliştirilen mobil uygulama çevik metodolojilerden olan Scrum metodu takip edilerek gerçekleştirilmiştir. Bu şekilde uygulama geliştirme sürecinde değişikliklere karşı duyarlılık oluşmuş ve değişikliklere en kısa sürede cevap verebilmek mümkün olmuştur.

Scrum Metodu proje sürecini işin ne kadarının bittiğinin görülmesi için İş Takip Grafiğinin oluşturulmasını zorunlu kılmaktadır[4]. Buna uygun olarak mobil uygulama geliştirme sürecinde İş Bitim Grafikleri oluşturulmuş, Geliştirme Ekibinin süreç içinde, her gün

işin ne kadarının yapıldığını görmeleri sağlanarak sürecin şeffaf olması sağlanmıştır. Bunun proje sürecinin takip edilmesinde son derece önemli olduğu, Scrum ile yürütülen projelerde mutlaka yapılması gerektiği sonucuna varılmıştır.

Schwaber ve Sutherland [9] Scrum Metodunda etkileşim ve iletişim kavramlarının önemi üzerinde durmaktadır. Buna uygun olarak, mobil uygulama geliştirme sürecinde Ürün Sahibi sürekli ulaşılabilir durumda olmuş, oluşan olumlu etkileşim ve pozitif iletişim ile değişikliklere karşı duyarlılık oluşmuş ve soru işareti olan noktaların hızlı şekilde çözülmesi sağlanmıştır. Özellikle mobil uygulama sektörü web sektörüne göre yeni sayılabilecek bir alan olduğu için önceden öngörülememiş ve bilgisayar teknolojilerine göre düşünülmüş noktaların (Ör: mobil işletim sistemi versiyonlarında kodların çalışmaması) hızlı şekilde çözülerek, mobil uygulama geliştirme sürecinin sağlıklı devam etmesi sağlanmıştır.

Yazılım geliştirme sürecinde önemli noktalardan birisi yazılımın süreç içinde test edilmesini sağlamaktır. Bu çalışma sürecinde Beck ve arkadaşlarının [4] belirttiği gibi mobil uygulama çalışır durumda tutulduğu için, uygulamanın sürekli test edilmiş, bu şekilde hataların anında tespit edilmesi ve çözümün hızlı şekilde bulunması sağlanmıştır. Bilgisayar teknolojilerinden farklı olarak birçok mobil cihaz ve mobil işletim sistemi versiyonunun olması ve farklı mobil cihaz ve işletim sistemi üzerinde farklı hataların çıkabileceği ve o cihaza özel çözüm üretilmesi gerektirebileceği için özellikle mobil uygulama geliştirme sürecinde daha önemli olduğu düşünülmektedir.

Scrum süreci denetim ve adaptasyonu başarı ile gerçekleştirmek için farklı toplantıların yapılmasını zorunlu kılmaktadır [9]. Bu çalışma kapsamındaki uygulama geliştirme sürecinde bu toplantılar yüz yüze ve/veya uzaktan haberleşme yöntemleriyle gerçekleştirilmiştir. Bunun sonucunda, uzaktan haberleşme yöntemleri ile yapılan toplantıların yerine, yüz yüze iletişimin sağlandığı toplantıların daha verimli geçtiği görülmüştür. Ayrıca Ürün Sahibi ile yapılan yüz yüze toplantılar sonucunda, Ürün Sahibinin ifade etmek istediklerinin kağıt kalem kullanarak, basit şekilde çizilmesinin, ne yapılacağı belirlenmesine ve prototipler hazırlanmasına büyük kolaylık sağladığı görülmüştür. Uzaktan haberleşme yöntemleri ile yapılan toplantılarda aynı verim sağlanamamıştır.

Ürün Gereksinim Listesinin belirlendiği ilk toplantı öncesi Scrum Uzmanı ve Geliştirme Ekibinin o alanda yapılmış çalışmaları inceleyerek, toplantıya gelmesinin bir gereklilik olduğu görülmüştür. Özellikle Scrum Uzmanı ve Geliştirme Ekibinin yabancı olduğu alanlarda bir iş geldiğinde bunun daha da önemli olabileceği düşünülmektedir. Klinik Araştırmalar için mobil uygulama geliştirme proje süreci başlamadan önce Scrum Uzmanı ve Geliştirme Ekibi bu alanda yapılan çalışmaları

incelemiştir. Bu şekilde, Scrum sürecinde yer alan tüm rollerin ortak bir dili konuşması sağlanmıştır.

Mobil uygulama geliştirme sürecinde, müşteriler mobil uygulama dünyasındaki bazı kısıtları düşünmeden ne istediklerini anlatmakta ve web tabanlı sistemlerdeki gibi özellikleri sistemlere ekletmeye çalışabilmektedirler. Bunun için Ürün Sahibinin de mobil uygulamaların web teknolojilerinden farklı olduğunu ve isteklerini ona göre iletmesi için örnek mobil uygulamaları incelemesinin yararlı olacağı görülmüştür. Bu çalışma sürecinde bilgilerin eklenmesi sırasında, ürün sahibi bir sayfaya çok uzun bilgilerin eklenmesini istemiş. Scrum Uzmanı devreye girerek, mobil uygulamalarda tasarimsal olarak daha iyi olacak ve kullanılabilirliği artıracığı için bir sayfada daha az bilgi verilmesinin uygun olacağını belirtmiş ve bilgilerin mobil uygulama üzerinde parça parça yapılandırılmasını sağlamıştır.

Scrum süreci yürütülürken bu sürecin çeşitli yazılımlarla desteklenmesinin önemli olduğu düşünülmekte ve süreci takip etmek ve ekran tasarımlarını yapmak için bir yazılımın kullanılması önerilmektedir. Çalışmada kullanılan web tabanlı OnTimeScrum Planlama[17] yazılımı sürecin ihtiyaçlarını karşılayarak sürecin verimli geçmesini sağlamıştır. Yine süreçte kullanılan MockUp Builder[16] ekran prototip hazırlama aracı, ekran tasarımlarının hazırlanmasında kolaylık sağlamıştır. Süreç içinde bu yazılımların deneme sürümleri kullanılmıştır.

Jeong ve ark [7] belirttiği gibi tüketicinin görüşünü yansıtan ve pazar ihtiyacını karşılayan mobil uygulamaların kar sağlama için mümkün olduğu kadar hızlı şekilde pazara çıkması önemli olmaktadır. Bu çalışma sürecinde de buna uygun olarak Scrum Metodu ile gerçekleştirilen mobil uygulama yaklaşık 2 ay gibi bir sürede Google Play Markette kullanıma sunulmuştur.

Sonuç olarak, bu çalışma kapsamında çevik metodolojilerden birisi olan Scrum kullanılarak mobil uygulama geliştirilmiştir. Değişim sürecine hızlı adaptasyon, müşteri isteklerine hızlı cevap, kısa sürede markete çıkma, test süreçlerini daha hızlı bir şekilde gerçekleştirme, uygulamanın sürekli çalışır olması, bireyler arasındaki etkileşim, Ürün Sahibi ile işbirliği sunulması gibi artılarından dolayı mobil uygulama geliştirme sürecinde Scrum Metodunun tercih edilmesi önerilmektedir. Ancak projeye başlamadan önce Scrum sürecinin ve çevik manifesto felsefesinin tüm ekip tarafından anlaşılması hatta benimsenmiş olmasının ve mobil uygulama geliştirme sürecinde çeşitli sınırlılıklar ve yeniliklerin olduğunun ekip tarafından kavranmış olmasının önemli olduğu düşünülmektedir.

Bu çalışmadan elde edilen çıktıların metodoloji seçiminde kararsız kalan araştırmacılar ve Scrum ile mobil uygulama geliştirecek kişiler için önemli bir kaynak olduğu düşünülmektedir. Özellikle Scrum Metodunun mobil oyun sektöründe aşamalı oyunlarda uygulanmasının, pazar rekabetinde avantaj sağlayacağı

düşünülmektedir. Her aşama bir Koşu sürecinde bitirilerek, bu Google Markette yayınlanabilir. Bu şekilde rekabetin olduğu pazarda uygulamanın mümkün olan en kısa sürede pazara çıkması sağlandığı için kâr etme oranı artırılabilir. Ayrıca Scrum sürecinde her Koşu sonucunda oluşan ürün parçasının yayınlanması ile uygulamanın Google Markette arama sonucunda ön sıralarda yer alması sağlanabilir. Bu da tam rekabet piyasasında ürün açısından önemli bir avantaj olarak görülmektedir.

KAYNAKLAR (REFERENCES)

- [1] C. Scharff, R. Verma, "Scrum to support mobile application development projects in a just-in-time learning context", **Proceedings of the 2010 ICSE Workshop on Cooperative and Human Aspects of Software Engineering**, Cape Town, New York, Association for Computing Machinery, ISBN:978-1-60558-966-4, 25-31, 01-08 Mayıs 2010.
- [2] H.K. Flora, S.V. Chande, "A Review and Anaysis on Mobile Application Developmnet Processes Using Agile Methodolgies", *International Journal of Research in Computer Science*, 3 (4), 8-18, 2013.
- [3] V. Rahimian, R. Ramsin, "Designing an Agile Methodology for Mobile Software Development: A Hybrid Method Engineering Approach", **2nd International Conference on Research Challenges in Informations Science**, Marrakech,, Morocco, ISBN: 978-1-4244-2273-9, 337-342, 3-6 Haziran 2008.
- [4] Internet: K. Beck, M. Beedle, A. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, D. Thomas, Manifesto for Agile Software Development, <http://agilemanifesto.org/>, 25.02.2014.
- [5] M. Elibol, and Ç. S. Erol, "Mobile Application Development With Agile Methodology", **Economics and Business Communication Challenges**, Editör: Dyduch, W. And Pankowska, M. , House of the University of Economics in Katowice, Katowice, Poland, 72-86, 2014.
- [6] P. Abrahamsson, A. Hanhineva, H. Hulkko, T. Ihme, J. Jaalinoja, M. Korkala, J. Koskela, P. Kyllonen, P. Salo, "Mobile-D:an agile approach for mobile application development", **Proceeding OOPSLA '04 Companion to the 19th annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications**, Vancouver, BC, Canada, ISBN:1-58113-833-4, 174-175, 24-28 Ekim 2004.
- [7] Y.J. Jeong, J.H. Lee, G.S. Shin, "Development Process of Mobile Application SW Based on Agile Methodology", **10th International Conference on Advanced Communications Technology**, Gangwon-Do, Institute of Electrical and Electronics Engineers (IEEE), ISBN: 978-89-5519-136-3, 362-366, , 17-20 Şubat 2008.
- [8] J. Highsmith, **Agile Software Development Ecosystems**, Addison Wesley, Boston, ISBN: 0-201-76043-6, 2002.
- [9] K. Schwaber, J. Sutherland, **The Definitive Guide to Scrum: The Rules of the Game**, Temmuz 2013.
- [10] K. Vlaanderen, S. Jansen, S. Brinkkemper, E. Jaspers, "The agile requirements refinery:Appling SCRUM principles to software product management", *Information and Software Technology*, 53 (1), 58-70, 2011.

[11] P. Abrahamsson, O. Salo, J. Ronkainen, J. Warsta, **Agile Software Development Methods: Review and Analysis**, VTT Publications 478 Espoo, Finland, 2002.

[12] H. Kniberg, **Scrum and XP from the Trenches**, C4 Media Inc, United States of America , ISBN:978-1-4303-2264-1, 2007.

[13] Internet: P. Nicolas, Introduction to SCRUM Agile Process for Global Software Development, http://www.pnextpert.com/files/SCRUM_for_Global_Development.pdf, 25 Nisan 2014.

[14] K. Schwaber, **Agile Project Management with Scrum**, Microsoft Press, Washington, , ISBN:0-7356-1993-X, 2004.

[15] M. Beedle, M. Devos, Y. Sharon, K. Schwaber, J. Sutherland, "SCRUM: An extension pattern language for hyperproductive software development", **Pattern Languages of Program**, 4, 637-651, 1999.

[16] Internet: <http://mockupbuilder.com>, 27.03.2017

[17] Internet: <https://www.axosoft.com>, 27.03.2017