

Açık Kaynak Yazılım ve Tarihçesi

Açık kaynak yazılım demek kaynak kodları açık olan ve isteyen herkesin kodlara erişebildiği, okuyabildiği, gerektiğinde değiştirebildiği yazılım demektir.

Açık Kaynak yazılımı, ilk bilgisayar yaratıldığından beri var olan bilgi paylaşımı olgusuna dayanmaktadır. Açık kaynak kodlu yazılımlar bilginin paylaşarak evrilmesini ve çeşitli geliştiricilerin aynı konu üzerinde iş birliği yapmasını sağlamıştır.

Açık kaynak kodlu yazılım temelde özgür yazılım fikrine dayanmaktadır. Ancak asla aynı anlama gelmemektedir.

Özgür yazılım fikrini ortaya atan kişi ise Richard Stallman'dır. Richard Stallman tartışmasız özgür yazılım hareketini başlatan ve bugünlere gelmesini sağlayan kişidir.

1980'lerin başında, MIT AI Lab kapandıktan sonra, çeşitli yazılımlar için gizlilik anlaşmaları (NDA) yapılması artmıştı. Bu çerçevede, iş birlikleri azaldı ve laboratuvarlar yetenekli yazılım geliştiricilerini özel yazılımlar kullanan özel şirketlere kaptırmaya başladılar. 80'lerin başında üreticiler teknolojileri üzerinde daha fazla telif haklarına sahip oldular, kaynak kodlarını sakladılar ve yazılımlarının lisanslı kullanımını istediler. Tescilli yazılım teknoloji dünyasında pazar payını devraldı.

Richard Stallman bundan memnun değildi ve özgür yazılımın fiyat değil, özgürlük meselesi olduğuna inanıyordu.

Daha sonra Stallman, 1984'te GNU projesine başladı. GNU projesi serbest bir işletim sistemi yaratacaktı. Stallman, bunun özgür bir yazılım topluluğu oluşturmak için kilit basamak olduğuna inandı. Yazılım fiyat açısından değil, erişilebilirlik açısından "özgür" olmalıydı.

GNU projesi 1990'lı yılların başında Linux ile birleşti. Neredeyse on yıl önce öngörülen ücretsiz işletim sistemini Stallman, nihayet iki proje birleşiminden yarattı. GNU/Linux günümüzde tüm genel amaçlı işletim sistemlerinin altyapısını oluşturmaktadır.

Yıl 1997'yi gösterdiğinde başka bir özgür yazılım savunucusu olan Eric Raymond, GNU Projesi ve Linux Projesi'nin geliştirme stillerini birleştirmeden önce karşılaştırarak, The Cathedral and The Bazaar isimli makaleyi yayımladı. Bu iki modeli kısaca özetlemek gerekirse;

The Cathedral (GNU Emacs) : Küçük bir geliştirici grubu tarafından geliştirilmiş ve kaynak kodu her yazılım sürümünde serbestçe dağıtılmaktadır.

The Bazaar (Linux kernel): Kaynak kodu internet üzerinden genel olarak geliştirilir.

Raymond'ın bu makalede tarif ettiği prensiplerin çoğu daha sonra çevik yazılım ve DevOps hareketinin merkezi olacaktır.

1998 yılında Netscape web tarayıcısı Mozilla'nın kaynak kodunun yayınlanmasında "The Cathedral ve The Bazaar" kilit bir belge olarak kabul edildi. Bu, yaygın olarak kullanılan bir masaüstü yazılımı parçasının serbest yani açık yazılım olarak ilk büyük sürümüydü.

Netscape'in deneyinin başarısı ile Eric Raymond, açık kaynak topluluğunu büyötmek için Açık Kaynak Girişimi'ni (OSI) kurdu.

Artık açık kaynak yazılımları yükselişe geçmeye başlamıştı. Özellikle, 1999' da VA Linux ve Red Hat'in halka arzlarının gerçekleşmesi, her iki şirketin de halka açık şirketler olarak açılış günlerinde hisse fiyatlarında büyük kazanımlar elde etmesi, açık kaynağın ticari ve ana akım olacağını kanıtladı.

Ayrıca 1999'da IBM, geliştirilmesine 1 milyar dolar yatırım yaparak Linux'u desteklediklerini açıkladı, bu da kullanımını geleneksel kullanıcılar için daha az riskli bir duruma getirdi. Ertesi yıl Sun Microsystems, ofis paketi StarOffice'in kaynak kodunu yayınladı ve OpenOffice.org projesini yarattı.

Açık kaynaklı projelerinin büyük Silikon Vadisi fonları ile bir araya getirilmiş etkisi, Wall Street'in açık kaynaklı yazılım etrafında inşa edilen genç şirketler için dikkatleri çekti. IBM ve Sun Microsystems gibi teknoloji devlerinin getirdiği pazar güvenilirliği, açık kaynağın büyük ölçüde benimsenmesini sağladı. Açık kaynak modelinin benimsenmesi bugün teknoloji endüstrisinde Linux'un ve açık kaynakların hakimiyetine yol açtı.

Yıllar boyunca, açık kaynak ve kapalı kaynak geliştirme yaklaşımının artıları ve eksileri üzerine birçok tartışma yapıldı. Açık kaynaklı yazılımlar için maliyetler çok daha düşük olsa da hızlı servis garanti edilmezdi. Bu tartışmalar geçmişten günümüze kadar halen devam etmektedir.

Açık kaynak ve kapalı yazılımların nerede ve hangi koşullarda kullanıldıklarına bağlı olarak kendilerine özgü güçlü yanları ve zorlukları vardır. Ve birbirlerini dışlayan modeller değildirlir. Tam tersine günümüzde birçok kurum her iki yaklaşımı birlikte kullanarak önemli avantajlar elde etmektedir. Apple Mac OSX örneğindeki gibi işletim sisteminin alt katmanlarında açık kaynak tercih edilirken kullanıcı deneyimine yakın ön yüz katmanında kapalı yazılımları tercih etmişlerdir. Bu yaklaşımlar tabi ki kurumun stratejik avantaj sağladığı yerlere göre değişebilir. Bununla birlikte IBM ve Microsoft gibi kurumlar ticari olarak geliştirdikleri kapalı kodları açık kaynak yazılımlara dönüştürmekte, kodları çeşitli topluluklara açarak çözümlerini çok daha geniş bir pazarda aramaktadırlar.

Günümüzde açık kaynak yazılım ve ticari yazılımlar birbirlerinin alternatifi olmanın aksine yenilikçilik ve çözümlere dayalı güçlü sistemlerin oluşmasında birbirlerine destek olmaktadır.

Son yıllarda, büyük veri, yapay zeka ve nesnelerin interneti gibi teknolojiler uygulama mimarilerine büyük değişikliklere neden oldular. Başlangıçta düşük maliyet kurumlar için önemli olsa da bugün ulaşılan seviyenin temel sebebi büyük veri ve yapay zekada olduğu gibi önemli yeniliklerin neredeyse tamamının açık yazılım çevresinde gelişmiş olmasıdır. Başka bir önemli veriye göre yeni başlayan projeler de katılımcıların %66 si açık kaynağı ticari çözümlere tercih etmektedir. Bu tercihte ki en önemli etki; kalitesi yüksek, güvenli, kolay erişilebilir açık teknolojilerin karmaşık ve uzun ticari satın alma süreçlerine göre her zaman tercih ediliyor olmasıdır.

Açık Kaynak Lisanslama Türleri

Bugün, Stallman'ın öncülüğünü yaptığı GPL lisansı üçüncü versiyonu (GNU GPLv3), birkaç düzine açık kaynaklı lisans türünden sadece biridir. Açık kaynaklı yazılımı tanıtmak ve terimin kullanımını normalleştirmek amacıyla 1998 yılında kurulan Açık Kaynak Girişimi Kurumu, 80'den fazla açık kaynak lisansı onaylamıştır. Bu 80 lisans genellikle iki kategoriden birine girer: izin verilen lisanslar ve copyleft lisansları.

İzin verilen lisans basittir ve en temel açık kaynaklı lisans türüdür: Bildirim gereksinimlerine uyduğunuz sürece yazılımla istediğiniz her şeyi yapmanıza olanak tanır. İzin verilen lisanslar, yazılımı olduğu gibi ve sonucunda hiçbir garanti vermeden sağlar. Dolayısıyla izin verilen lisanslar aşağıdaki gibi özetlenebilir:

- Kod ile ne istersen yap
- Kullanım kendi sorumluluğunuzdadır
- Yazar / Katkıda bulunan kişiyi onaylayın

Copyleft lisansları, izin verilen lisansa ek gereksinimler ekler;

- Binary dosyalar dağıtırsanız, bu binary dosyalar için kaynak kodunu hazırlamanız gerekir.
- Kaynak kod, kodu aldığınız aynı copyleft koşulları altında mevcut olmalıdır.
- Lisans sahibinin lisans kullanımına ek kısıtlamalar getiremezsiniz.

Aşağıdaki tabloda, izin verilen ve copyleft çerçeveleri altında popüler açık kaynak lisansları sınıflandırılmaktadır.

İzin Verilen Lisanslar	Copyleft Lisanslar
<ul style="list-style-type: none">• BSD (Berkeley Software Distribution)• MIT• Apache 2	<ul style="list-style-type: none">• Affero GPL (AGPL)• GPL• Lesser GPL (LGPL)• Mozilla Public License (MPL)• Eclipse Public License (EPL)• Common Development and Distribution License (CDDL)

Özet olarak bu lisans sistemleri, açık kaynaklı yazılım hareketi ve altında yatan felsefeyi geliştirebilmek için hazırlanmış ve yazılım geliştiren kişiler tarafından ortak bir projeye katkı sunmadan önce üzerinde anlaşılması gereken bir yeniden uyarılma ve yeniden kullanılabilirlik politikası olarak anlaşılmalıdır.

Açık Kaynak Projelere Hizmet Veren Servisler

Aşağıda internette araştırmış olduğum açık kaynaklı projelerinizde kullanmak isteyebileceğiniz ve gene açık kaynak kodlu olan bazı servisleri paylaşıyorum.

Version Kontrolü

- BitBucket
- GitHub
- GitLab

IDE

- AppCode Objective C IDE
- IntelliJ Idea - Java IDE
- PhpStorm - PHP IDE
- PyCharm - Python IDE
- RubyMine - Ruby IDE
- Visual Studio Community
- WebStorm - JavaScript, CSS, HTML IDE

Project Hosting

- NuGet
- Python Anywhere
- SourceForge

Public Cloud

- OpenShift

Dökümantasyon

- GitBook

E-mail Gönderme

- Helpmonks - Basit ve ücretsiz.
- MailChimp - 2000 bin kullanıcı 12000 Email kadar ücretsiz.

Monitoring

- Bugsnag
- Datadog
- Healthchecks.io
- Hund
- Sentry

Code Review

- Codacy
- CodeClimate
- QuantifiedCode
- Landscape
- Snyk

Proje Yönetimi

- HuBoard
- Taiga.io
- Teamwork
- Trello
- YouTrack
- Waffle.io
- ZenHub.io

Chat

- Zulip

Tasarım ve İkonlar

- Canva

- Openclipart
- Unsplash

SSL Certificates

- Cloudflare
- GoDaddy
- Let's Encrypt

Localization

- Crowdin
- Launchpad
- POEditor
- Transifex
- Translation.io

Açık Kaynak Projelerine Destek Olurken Dikkat Edilecek Hususlar

Açık kaynaklı projelere destek olmak, bir yazılımcının kendini geliştirmesi, kariyerinde ilerlemesi için oldukça önemlidir.

Bununla birlikte bir topluluğa katkı sağlamak ve başkalarının problemlerine destek olmak bireyin kendisine çok büyük bir değer katmaktadır. Açık kaynak projelerine birçok anlamda katkıda bulunabilmektedir. Bazı özelliklerini eksik olan veya hatalı olan projelerin bulduğunuz noktalarını forklayıp istediğiniz özellikleri ekleyebilirsiniz. Bununla birlikte açık kaynaklı yazılımların dökümantasyon ihtiyaçları da bulunmaktadır. Kaliteli bir dökümantasyon hazırlayarak açık kaynak kodlarına katkıda bulunabilirsiniz.

İlk olarak bir sorunu, eksiği çözmeden önce geliştiriciler ile iletişime geçin. Durumu belirtin daha sonra forklayıp geliştirme yapmaya başlayın. Çünkü o problem ile ilgili belki sizden önce birileri geliştirme yapmaya başlamış olabilir. Topluluğun kimin ne üzerinde çalıştığından haberi olması önemlidir.

Pek çok proje README ya da CONTRIBUTING dosyalarında yeni şeyler eklemek için nelerin nasıl yapılması gerektiğinden bahseder. Katkıdan bulunmadan önce projeyi, gereksinimlerini, kod standartlarını incelemek fayda sağlayacaktır.

Pek çok proje, eklediğin kodun belirli bir standart da sahip olmasını ve kodla birlikte test ve dokümantasyonunun da yazılmasını bekler. Buna uygun bir şekilde geliştirme yapıldığından emin olunmalıdır.

Bu süreçte bazı zorluklar yaşanabilir. Yazıp düzenlemiş olduğunuz kod, topluluklarda ciddi şekilde eleştirilebilir, sorgulanabilir ve reddedilebilir. Bunlara alışkın olup kendinizi geliştirmelisiniz. Ret alan kodların neden ret aldığına dair geri bildirim muhakkak verilecektir. Bu çerçevede kendinizi ve kodunuzu geliştirerek katkıda bulunmaktan asla vazgeçmemelisiniz.

Katkıda bulunduğunuz projelerle ilgili geliştirici olarak ta geribildirim vermeyi unutmamalısınız. Karşılaşılan hataların mutlaka dökümanete etmelisiniz.

Her açık kaynaklı yazılım projesinin farklı bir build, test süreci olabilir. Projenin sadece kullanıcısı değil geliştiricide olacağınız için bu süreçleri dikkatlice incelemeli bu süreçlerde zorlandığınız konuları proje dokümantasyonuna ekleyip güncellemelisiniz. Bu sayede sizden sonrakilerde aynı dertlerden muzdarip olup benzer problemler yaşamasınlar.

En önemlisi yeni şeyler öğrenmeye her zaman açık olmalısınız.

Türkiye de Açık Kaynak Yazılımı

Türkiye’de açık kaynak kodlu yazılım yoğunlukla Linux ile özdeşleşmiştir. TUBITAK ULAKBİM tarafından bir Linux dağıtımına dönüştürülen Pardus projesi GPL Lisanslı olup açık kaynak yazılımı konusunda Türkiye’de öncüdür.

Pardus Projeler Portalı Kamu Kurum ve Kuruluşları ile KOBİ’lerin kullanımına sunulan ve açık kaynaklı olarak geliştirilen Lider Ahenk ve EnGerek gibi projelerde paylaşılmaktadır.

Lider Ahenk, kurumsal ağ üzerindeki, sınırsız sayıda farklı sistemi ve kullanıcılarını tek merkezden yönetebilmeyi, izlemeyi ve denetlemeyi sağlayan açık kaynaklı bir yazılım sistemidir.

EnGerek, açık kaynak kimlik yönetim sistemidir.

Bunlarla birlikte akla gelen ve dünyada kabul görmüş başka bir proje ise, PrimeFaces dir.

PrimeFaces açık kaynak kodlu bir yazılım olmakla birlikte Türkiye’de geliştirilip uluslararası pazarda kendine yer edinmiş nadir projelerdendir. Zengin bileşen kütüphanesine sahip olan, Ajax temelli oluşturulmuş, bir jar dosyası ile işinizi görebileceğiniz, mobil web uygulaması geliştirmek için Mobil UI Kiti’ne sahip olan, sizi HTML, Javascript, CSS ile uğraşmaktan kurtarmayı amaçlayan, açık kaynak kodlu bir JSF bileşenidir.

Son yıllarda özellikle kamu kurumlarında alt yapılara yönelik çeşitli açık kaynak kodlu yazılım uygulamalarına önem verilmektedir. Ancak Türkiye’de henüz açık kaynak kodlu yazılımların kullanılması veya geliştirilmesine yönelik düzgün bir yol haritası oluşturulamamıştır. Gerekli mevzuat çalışmalarının tamamlanması ve kurumlara bu yönde destek olunması gerekmektedir. Bu anlamda üzülerek söylemeliyim ki Amerika ve Avrupa’nın oldukça gerisinde seyretmekteyiz.

Doğuş Teknoloji'de Açık Kaynak Yazılımı

Doğuş Teknoloji olarak yazılım üreten bir firmayız. Yazılım üretip, lisanslayıp yazılım satarken bu yazılımın kodunu herkesle paylaşmak bizim gibi firmalar için ilk başta çok anlam ifade etmeyebilir. Ancak, dünyada artık şirketler yazılım satmıyor ve açık kaynak olarak hem kendilerine hem de piyasaya çok daha fazlasıyla katkıda bulunuyorlar. Doğuş Teknoloji bünyesinde projelerin açık kaynak olarak yazılması, öncelikli olarak hem bizler hem de şirket için harika bir reklamdır. Kod ve projemiz dış kaynakları (açık kaynak kodlayıcılarını) etkileyecek kadar popülerse, daha fazla işi daha hızlı ve daha ucuz hale getirmemize fayda sağlayacaktır. Akıllı insanlar, diğer akıllı insanlarla takılmayı sever. Akıllı geliştiriciler akıllı kodlarla takılmayı sever. Kaynak kodunu açtığımızda bu yetenekli insanları kendimize çekmeyi başarırız. İşe alımda, mümkün olan en iyi görüşme, yapmak zorunda olmadığınız görüşmedir; çünkü aday zaten açık kaynak projelerimizden birinde çalışmış olacaktır. Bu şekilde işe alınan birisinin bu açık kaynak koduna bağlılığı bir x kişisine göre çok daha fazla olacaktır. Bizim gibi son müşteriye hitap eden firmalarda, hangi projenin açık kaynak olarak ele alınması iyi seçilmelidir. Benim kişisel fikrim böyle bir proje seçilecekse kesinlikle son müşteriye direk olarak hitap etmeyen back-end projeler olmalıdır. Çünkü içerisinde front-end içeren açık kaynak projeleri genellikle daha özensiz ve müşteriye memnun etmeyen sonuçlar çıkarabilmektedir. Bu sebeple ürünümüz ne olursa olsun son müşteriye direk olarak hitap etmemelidir. Çünkü bizim buradaki amacımız ürünü oluşturmamıza yardımcı olacak kaliteli kişileri etrafımızda toplamak ve Doğuş Teknoloji olarak işe alımda bir portföy ve aynı zamanda şirket çevresinde yetenekli ve yaratıcı bir topluluk oluşturmaktır.

Dünyanın en büyük açık kaynaklı şirketli olarak adlandırılan Facebook'a bakarsak, Facebook'un açık kaynak şefi James Pearce açık kaynaktan elde ettikleri faydayı şu şekilde özetlemiştir;

Mühendislerimizin büyük bir yüzdesi, aramıza katılmadan önce açık kaynaklı projelerimiz hakkında bilgi sahibi olduklarını ve bu projelerin şirkete katılma kararlarına olumlu yönde etki ettiğini belirtmişlerdir.

Günümüzün en iyi yazılımlarının çoğu açık kaynak ile yazılmış olup, Hadoop, Spark, MongoDB veya Android gibi şirketler, son müşterilerle uğraşmadan çok önce geliştiricilerle etkileşime girmektedirler.

Sonuç olarak Doğuş Teknoloji olarak rekabet avantajı bırakmayan veya olmayan yazılımları biriktirmeyi bırakmalı ve paylaşmaya başlamalıyız. Bu paylaşım ile birlikte şirketimizin nihai müşterilerine daha iyi hizmet verebilecek yetenekli çalışanları aramıza katabileceğiz. Bizim gibi ARGE firmaları için önemli olan, bünyesinde arge yapabilecek, kendini güvenli alanından çıkarıp farklı işlerle uğraşabilecek cesarete ve çevikliğe sahip geliştiricileri barındırmaktır. Bu insanları, insan havuzundan en iyi şekilde seçmek için açık kaynak projelerinin çok iyi bir yöntem olduğunu düşünüyorum. Bununla birlikte bizim müşteriye dokunduğumuz çözümler içinde arka planda açık kaynak kodlu projeleri kullanmamız hem maliyet hem güvenlik hem de kendimizi geliştirme anlamında bizlere çok şey katacağını düşünüyorum.

Biz DT-Bilişsel Servisler ekibi olarak Zeyrek Diyalog Platform'u projemizin hemen hemen her modülünü açık kaynak kodlu projeler üzerinde geliştirmekteyiz. İleri de umarız Türkçe NLP Pipelin'imizin en azından belirli modüllerini açık kaynak olarak paylaşabilir bu topluluğa bizde azda olsa bir katkı sağlayabiliriz.

Kaynakça

<https://www.fsf.org/>

www.understein.net/su/docs/CathBaz.pdf

<https://www.gnu.org/gnu/manifesto.en.html>

<https://medium.com/gitcoin/a-brief-history-of-open-source-3928cb451767>

<https://opensource.com/article/17/9/open-source-licensing>

https://en.wikipedia.org/wiki/The_Cathedral_and_the_Bazaar

<https://hackernoon.com/being-a-good-open-source-citizen-9060d0ab9732>

<https://youtube.com>