

Open Source Proje Geliştirme

1)Open Source Proje Geliştirme Tarihçesi

1970 ve 1998 yılları arasında Open Source yazılım herhangi bir tanımlama olmadan bir varsayım olarak varlığını sürdürmüştür. Fakat zaman içerisinde GNU,BSD ve LINUX gibi çok yaygın ve stabil çalışan işletim sistemleri üretebilecek bir seviyeye evrilmiştir.

Open source yazılım ortaya atıldığı dönemde daha önceki yazılım dağıtım metotlarından çok daha farklı bir metot ortaya sunulmuştur. Daha önceki geleneksel metodda, yazılım tamamen tamamlanmış bir ürün olarak kullanıcıya uygun işletim sistemi üzerinde lisanslanmış olarak aktarılmaktaydı. Bu yazılım üzerindeki destek ve problem çözüm istekleri, üretici firma tarafından hizmet verildiği süre boyunca sağlanmaktaydı. Fakat open source yazılım, kullanıcıya direk olarak kaynak koda erişebilecek şekilde dağıtılmaktadır. Bu sayede kullanıcı ilgili ürün üzerindeki hataları veya isteklerini, yazılımın kaynak koduna doğrudan erişerek tamamlayabilmektedir. Dağıtım için bir çok kanal kullanılmaktadır; ftp, media ürünleri ve code repository'ler gibi. Open source bu açıdan bakıldığında freeware/shareware yazılım ile karıştırılmamalıdır. Freeware/shareware yazılımlar, tamamlanmış ürünlerin bedelsiz olarak kullanıcıya kaynak kodu olmadan dağıtılmaktadır. Open source yazılım için bir çok lisanslama tipi mevcuttur. Bunlar ilerleyen bölümlerde detaylı olarak incelenecektir.

Open source yazılım tarihi derinden incelendiğinde hacker kültürü ile paralellik göstermektedir. Burada hacker terimi bilinenin ötesinde, hobisi ileri derecede yazılım teknikleri ile program yazmak olan kişileri tariflemektedir. İlgilenenler için A Brief History of Hackerdom makalesi bu anlamda ilgi çekici bilgiler içermektedir.

Olgun ve yönetilen önemli bir çok open source yazılım projesinde kabul görmüş bazı kültürel normlar zaman içerisinde gelişmiştir. Bunlar:

- Her yazılımın versiyonlarını ve kullanıcılar tarafından eklenen özellik ve bug fixleri yöneten bir yaratıcısı ve sürdürücüsü bulunmaktadır.
- Her yazılımın bir tartışma ortamı bulunmaktadır. Burada fikirler, istekler ve bug fixler konuşulup tartışılır.
- Genelde bir web sitesi ya da benzer özellikleri taşıyan bir code repo sayfası bulunmaktadır.

Open source yazılım tarihi incelendiği takdirde, 1971 yılında Harvard Üniversitesinden MIT Yapay Zeka laboratuvarına birinci sınıf öğrencisi olarak katılan Richard Stallman'ın yadsınamaz önemi göze çarpmaktadır. MIT AI laboratuvarında TECO, Emacs ve LISP programlama dilinin geliştirilmesi gibi bir çok önemli projeye katkıda bulunan Stallman, labın sürekli kağıt sıkıştırıcı printer'ı yenilenme zamanı geldiğinde, yeni gelen printer'ın sürücüsünün kaynak kodlarını talep etmiştir. Kaynak kodlara sahip olan başka bir üniversite bu isteği reddedince buna hayli şaşırarak Stallman, kendisine basit bir soru yöneltir: "Toplumsal neye ihtiyacı var ?" Her alanda inovasyona ön ayak olan bu soru, Stallman için de aynı sonucu getirmiştir. Bu sorunun cevabını "Why Software Should Not Have Owners" adlı makalesinde açıklayan Stallman, kısaca şu konulara parmak basmaktadır:

-İnsanların, sadece kullanabilecekleri değil, okuyup, değiştirip uyarlayabileceği yazılımlara ihtiyacı vardır.

-Özgürlük hissi sadece bir program üzerinde bile olsa kontrolünü yitirmekle birlikte zedelenebilir.

-Bunların da ötesinde toplumun daha iyiye ve daha güzele ulaşmak için birlikte çalışmak ve iş birliği yapmaya ihtiyacı vardır.

Bu fikirler açısından bakıldığında Stallman, bedava yazılım olarak tanımladığı bu mottoyu fiyat değil özgürlük kavramı açısından ele aldığı açıkça görülebilir.

1984 yılında Stallman, MIT'deki görevinden ayrılarak GNU adı verilen işletim sistemi projesini başlamıştır. 1985 yılında projenin ilk parçası olan GNU Emacs adı verilen editörü tamamladıktan sonra bu editörün dağıtımına başlamıştır. O yıllarda internet kullanımı günümüzdeki gibi olmadığından ötürü Stallman, bu editöre ilgi duyan yazılımcılara dağıtımı kargolayarak 150 dolar karşılığında yapmaktaydı. Bu şekilde günlük masraflarını çıkaran Stallman. Free software kavramının erişime açık, özgür fakat bedava olmak zorunda değil şeklinde tasvir etmiştir. Kendi deyimi ile "bedava olan bira değil ama sohbet" şeklinde de tasvir edebiliriz. Fakat bu dağıtım şekli ortaya yeni bir problemi de çıkarmaktaydı. Herhangi bir programcı bu dağıtımı alıp istediği amaçla tekrar dağıtmadan hazır paketler ve ürünlerin içinde kullanabilir ve Stallman'ın felsefesine aykırı bir kullanıma alet edebilirdi. Bu nedenle copyleft (telif feragatı) denilen yeni bir kavramı ortaya çıkarmıştır.

Copyleft kavramı, ilgili yazılımın bir kopya olduğunu ve kopyayı alan kişinin telif haklarına sahip olmadığını tescil etmektedir. Günümüzde ise 1991 yılında son halini almış olan GNU GPL lisansının temelini oluşturan bu kavramın ilk hali ise 1989 yılında ortaya çıkmıştır.

Tüm bunların yanında 1985 yılında Free Software Foundation bu projeyi fonlamak için kurulmuştur.

1991 yılında ise University Of Finland'da lisans eğitimini alan genç mühendis Linus Torvalds Linux Kernel v0.02'yi release etmiştir. Torvalds'ın comp.os.minx haber grubuna yazdığı ilgi çekici mesajda ise "kendi işletim sisteminizi kendi isterlerinize göre yazmayı, kendi driver'larınızı ise kendiniz üretmeyi istemez misiniz" şeklinde bir çağrı bulunmaktaydı. Torvalds sıfırdan bir işletim sistemi yazmak yerine GNU parçalarını yazılımı içerisinde kullanmaktaydı. Bu nedenle Stallman, bu işletim sisteminin adının GNU-Linux olması gerekliliğini savunmuş ve bu konuyu ortaya atmıştır.

1997 yılında Richard S. Raymond, The Cathedral and the Bazaar makalesinde Linux'un başarısını derinlemesine incelemiş ve temel olarak başarısının kodun kalitesinin ötesinde arkasındaki topluluk desteği olduğunu vurgulamıştır. Bu makalede vurgulanan temeller ve kavramlar günümüzde agile ve dev-ops kavramlarının temelini oluşturmaktadır. Linux'un başarısı için "release early, release fast" mottosunu öne çıkarmıştır.

1998 yılında ise Netscape Mozilla tarayıcısının kodlarını yayınladıktan sonra, Eric Raymond, Bruce Perens, Michael Tiemann, John Hall gibi isimlerden oluşan bir grup insan Palo Alto'da yeni bir terim üzerine bir çalışma yapmışlardır. İlk olarak open source terimi burada Christine Peterson tarafından kullanılmıştır. Daha sonrasında yine aynı yıl içerisinde Bruce Perens ve Eric Raymond tarafından Open Source Institution kurulmuştur.

Tüm bu gelişmelerin yanında dünyada bir çok open source yazılım projeleri dağıtılmaya başlanmıştır.

-1990 yılında Guido van Rossum Python programlama dilini release etmiştir.

- 1995 yılında Apache web server'ı release edilmiş ve çok hızlı şekilde yayılmaya başlamıştır.
- 1990'ların ortasında ise mSql, mySql ve PostgreSQL release edilmiştir.

2) Open Source Projelerinin Gücü ve Yaygınlaşmasının Nedeni ? Nasıl İyi Bir Open Source Citizen Olunur ?

Bir önceki bölümde de belirtildiği gibi 1997 yılında Richard S. Raymond, The Cathedral and the Bazaar makalesinde open source projelerde başarının asıl anahtarının kodun kalitesinin ötesinde arkasındaki topluluk desteği olduğunu vurgulamıştır. Bunun temelinde ise Richard Stallman'ın felsefesine göre insanların birbiri ile ortaklaşa iş yapma isteği bulunmaktadır. Lakhani, Karim R., and Robert G. Wolf. "Why hackers do what they do: Understanding motivation and effort in free/open source software projects." (2003). Makalesinde ise open source projelerde yer alan programcıların ana motivasyonları bir anket ile incelenmiş ve sonuçlar Stallman'ın felsefesi ile aynı paralellikte çıkmıştır. 287 farklı open source projesinde çalışan 684 yazılım geliştiricisine yapılan anket sonuçlarına göre, geliştiricilerin yaratıcılık ve özgürlük hislerinin artması, open source projelere katkıda bulunmalarındaki en büyük faktör olarak gözlemlenmiştir. Ayrıca programlama becerilerinin artması ve entelektüel sebepler diğer bir ana faktör olarak ortaya çıkmıştır.

Bu veriler ışığında, iç ve iç eğlence motivasyonları open source proje katılımcıları arasında en büyük motivasyon kaynağı olarak görülmüştür. Aşağıda figür-1'de anket sonuçları yer almaktadır.

Table 6– Motivations to contribute to F/OSS projects

<i>Motivation</i>	<i>% of respondents indicating up to 3 statements that best reflect their reasons to contribute (%)</i>	<i>% volunteer contributors</i>	<i>% paid contributor</i>	<i>Significant difference (t statistic/p value)</i>
<i>Enjoyment based Intrinsic Motivation</i>				
Code for project is intellectually stimulating to write	44.9	46.1	43.1	n.s.
Like working with this development team	20.3	21.5	18.5	n.s.
<i>Economic/Extrinsic based Motivations</i>				
Improve programming skills	41.3	45.8	33.2	3.56 (p=0.0004)
Code needed for user need (work and/or non-work)*	58.7	-	-	-
- Work need only	33.8	19.3	55.7	10.53 (p=0.0000)
- Non-work need	29.7	37.0	18.9	5.16 (p=0.0000)
Enhance professional status	17.5	13.9	22.8	3.01 (p=0.0000)
<i>Obligation/Community based Intrinsic Motivations</i>				
Believe that source code should be open	33.1	34.8	30.6	n.s.
Feel personal obligation to contribute because use F/OSS	28.6	29.6	26.9	n.s.
Dislike proprietary software and want to defeat them	11.3	11.5	11.1	n.s.
Enhance reputation in F/OSS community	11.0	12.0	9.5	n.s.

* Aggregation of responses that indicated needing software for work and/or non-work related need . Not an actual survey question. Overlap in user needs limited to 4.9% of sample. n.s. = not significant.

Sonuçlara bakıldığında 60'ların akımı olan hippie kültürü open source yazılım geliştirme kültürüne çok benzemektedir. Daha fazla özgürlük daha fazla eğlence ve kurumlara bağlılıktan kaçınma. Bu insanları tanımlamak için open source citizen tanımı kullanılabilir. Fakat gündelik hayatta da iyi birer birey olmak için bazı kurallar geçerlidir. Bunu open source dünyası için de aynı şekilde düşünebiliriz. Peki bunlar nelerdir ?

-haber gruplarında ve forumlarda soru sormakta sorun yoktur fakat, bundan önce dokümanlar ve readme dosyaları open source katılımcısı tarafından iyice okunmalıdır,

-projenin katılım kılavuzu okunmalı ve bu kılavuza göre hareket edilmelidir,

-haber grubu veya forumda konu açmadan önce daha önce açılmış konular iyi taranmalı ve aynı konu tekrar açılmamalıdır,

-herhangi bir bug rapor edilirken, içerik ve açıklama iyice yapılmalı ve bug iyi adreslenmelidir,

-istekler kısa ve net olarak ifade edilmelidir,

-topluluk ile iletişim olabilecek her yerde sürdürülmeli ve elde edilen bilgiler ve deneyimler olabildiğince topluluk ile açık ve net bir şekilde paylaşılmalıdır,

-sabırlı olunmalı,

-projenin var olan kodlama paternine uygun olacak şekilde geliştirme yapılmalıdır.

-projede bir contribution.md varsa buna uyulmalı eğer bu yoksa projeye eklenmelidir,

-projenin lisanslamasına saygı duyulmalı ve eğer projede bir lisans dosyası yoksa yeni bir konu yaratarak bu dosyanın projeye eklenmesi sağlanmalıdır,

-diğer citizen'ları zor duruma sokmamak için yazılan kod parçası için gerekli testler yapılmalı ve paylaşılmalıdır,

-var olan kod branchler korunmalıdır,

-tutulamayacak sözler verilmemeli ve bu sayede diğer contributorlar ve maintaner'lar için oluşabilecek planlama sorunlarının önüne geçilmelidir.

3)Open Source Lisanslama Seçenekleri Nelerdir?

-MIT License (Expat) : Orijinal kodun kendisini ve lisansı kendi kodunuzun içinde bulundurduğunuz sürece istediğiniz her şeyi yapabileceğiniz lisans türüdür.

-Apache License 2.0 (Apache-2.0) : Orijinal kodun kendisini, lisansı , eğer kod içinde var olan notice dosyasını kendi kodunuzun içinde ve yaptığınız değişiklikleri belirttiğiniz taktirde istediğiniz her şeyi yapabileceğiniz lisans türüdür.

-GNU General Public License v3 (GPL-3): Orijinal kodun kendisini, lisansı , eğer kod içinde var olan notice dosyasını kendi kodunuzun içinde , yaptığınız değişiklikleri belirttiğiniz ve bir kurulum kılavuzu bulundurduğunuz taktirde istediğiniz her şeyi yapabileceğiniz lisans türüdür.

-BSD 3-Clause License: MIT License ile çok benzer özelliklere sahip olan ve neredeyse sizi lisans ve orijinal kod bulundurarak sınırsız özgürlüğe kavuşturan lisans türüdür.

-GNU General Public License v2 (GDPL-2.0): GNU General Public License v3 (GPL-3) ile çok benzer olan lisans türüdür.

-Mozilla Public License 2.0 (MPL-2.0): MPL kodlarını farklı dosyalarda sakladığınız sürece özgür olduğunuz lisans türüdür. GPL ile uyumludur.

-Microsoft Public License: Lisans ve orijinal kodu bulundurduğunuz sürece her şeyi yapabileceğiniz lisans türüdür.

Diğer Yaygın Kullanılan Lisans Türleri

-Academic Free License 3.0

-Apple Public Source License 2.0

-Fair License

-Common Public License 1.0

-IBM Public License

-4-Clause BSD

-The Json License

-Educational Community License

-Attribution Assurance License

-Free Public License 1.0

-Universal Permissive License 1.0

-Adaptive Public License 1.0

-IPA Font License

-Very Simple Public License

-Creative Commons Attribution-Non Commercial 2.0 Generic

-Mitty

-Mts

-Sendmail License

-hx711_SPI

-Zed License

-Jared M.F. Open Source Public License

-XSkat License

4)Open Source Projelere Hizmet Veren Servisler Nelerdir ?

Kaynak Kod Servisleri: (Code Repos): [Github](#), [Codeplex](#), [BitBucket](#), [Google Code](#),

Web App Hosting : [Azure](#) (10 uygulamanın temel pricing tier'larla birlikte host edilmesi bedava),[App Harbor](#).

Continious Integration ve Deployment Servisleri: [TravisCI](#) (GitHub projeleri için oldukça popüler hale gelen servis, testler kořmak ve raporlar sunmak için güzel bir alternatif sunmaktadır), [BuildHive](#), [CodeBetterCI](#), [Bamboo](#)

Test Servisleri: [SauceLabs](#) (Web ve mobil uygulamaları bir çok platform ve tarayıcı üzerinde test etmeye yaramaktadır), [BrowserStack JavaScript Testing](#) (JS testlerini kořmaya yaramaktadır)

Belli bařlı dięer Servisler

-TeamCity: Continious Integration Server

-Moqups: Mock up üretmek için kullanılan bir tool

-Jira: Online bug ve issue takip yazılımı.

-Crowdin: Lokalizasyon platformu

-BugSense: Mobil uygulamalar için bug bulma yazılımı

-Abiquo: Bulut Yönetim Servisi

-Zope: İçerik Yönetim Servisi

-Zmanda: Dosya ve veritabanı yedekleme ve geri getirme servisi

-Webix: Java Script Framework

-Vyatta: Router, Firewall ve VPN servisi

-Websphere: IBM tarafından sunulan web ve uygulama sunucusu

-SUSE: Linux dağıtımı

-SunStudio: C ve C++ dili için IDE

-Solaris : İşletim Sistemi

-StarOffice: Ofis uygulamaları

- Spring, Struts gibi Java Frameworkleri.

-Google Colab: 12 saatlik ücretsiz Machine Learning platformu

-Tensorflow, Keras gibi ML Kütüphaneleri.

-BerkeleyDB: Veritabanı motoru

- PGAdmin, MySql Enterprise: İlişkisel Veritabanı Yönetim Sistemleri
- EyeOS: Bulut tabanlı işletim sistemi
- Lotus Symphony: Ofis Uygulamaları
- NetBeans, Eclipse: Java İde'leri.

5)Türkiye'deki Open Source Uygulamaları ve Dünyadaki Güncel Uygulamalarla Karşılaştırılması

Türkiye'de open source yazılım projelerinin başlaması Dünya genelinde olduğundan biraz daha geç yaşanmıştır. Network World tarafından yapılan bir araştırma ülkelerin open source yazılım projeleri geliştirme alanında attığı adımları ortaya koymuştur. Dünya üzerinde bu alanda en büyük destek ABD, Çin, Hindistan ve Avrupa ülkeleri tarafından verilmektedir. Türkiye ise open source yazılım projeleri için yasal yaptırımın değerlendirme aşamasında olduğu ülkeler arasında gösterilmektedir.

Malezya ise bu konuda ülkemize önemli bir örnek teşkil etmektedir. 2004 yılından beri Malezya'nın uyguladığı Kamu Kurumlarına açık kaynak yazılım kullanımı programı sayesinde yazılım lisans giderlerinde yüzde seksen tasarruf, geliştirme ve danışmanlık çalışmalarında yüzde elli sekiz azalma, yazılım destek hizmetlerinde yüzde yedi azalma ve toplamda yüzde otuz tasarruf sağlanmıştır.

Tablo 1 - Malezya kamu kurumlarında AKKY'nin kullanım oranları, Haziran 2010⁴³

Uygulanma	Hedef		Gerçekleşme	
	Sayı	%	Sayı	%
Genel itibariyle AKKY'nin tercih edildiği kurumlar	724	100	705	97.38
Sunucu ve benzeri altyapılarda AKKY'nin tercih edildiği kurumlar	580	80	640	88.4
İşletim sistemi olarak AKKY'nin tercih edildiği kurumlar	218	30	496	68.51

Ülkemiz bu alanda en büyük yaptırımını Kamuda yapmayı planlamaktadır. 2012 yılında Bilgi Toplumu Dairesince yayınlanan Kamuda Açık Kaynak Kodlu Yazılım Kullanımı raporunda 2015-2018 Bilgi Toplumu Stratejisi ve Eylem Planı'nda ise AKKY açısından atılım yapılmasına dönük öneriler yer almaktadır. Bu anlamda tüm kamu kuruluşlarında Pardus işletim sistemi kullanımının yaygınlaştırılması gibi yaptırımlar başlanmıştır.

İlk olarak Adalet Bakanlığında kullanılan UYAP programı 2006 yılında başlayan bir programla açık kaynak proje haline getirilmeye çalışılmaktadır. Enerji Piyasası Düzenleme Kurumunda ise 2009 yılında Pardus işletim sistemine geçiş düzenli olarak başlamıştır. Milli Savunma Bakanlığı ise Pardus'a geçişi 2007 yılında başlatmıştır.

Ülkemizde geliştirilen en büyük open source projesi olan Pardus işletim sistemi Tübitak-Bilgem tarafından Eylül 2003'de geliştirilmeye başlanmış ve ilk sürümü Aralık 2005'de dağıtılmaya başlanmıştır. 2007'de ise yeni bir sürüme geçilmiştir. Bu sürümle birlikte, kurumsal kullanıcılar ile normal kullanıcıların

için “Kurumsal” ve “Bireysel” olmak üzere iki farklı dağıtım geliştirilmeye başlanmıştır. İlk geliştirilen kurumsal Pardus sürümüne “Kurumsal 1” adı verilmiş ve son olarak 2011 yılında “Pardus 2011” (bireysel sürüm) ve “Pardus Kurumsal 2” sürümleri dağıtılmaya başlanmıştır.

Ülkemizde geliştirilen ikinci büyük proje olan Fatih projesi ise açık kaynak yazılımına destek veren ikinci büyük projedir.

6)Doğuş Teknolojide Uygulanabilecek Alan, Proje ve Potansiyel Faydalar

Open source proje komünitesi Dünya’da olduğu gibi Türkiye’de de yazılım dünyasının gittiği trendleri takip etmektedir. Önümüzdeki yıllarda Yapay Zeka, IOT, Blockchain , Progressive Web Apps ve container teknolojisi gibi kullanım alanı genişlemesi beklenen alanlarda open source proje sayılarının hem dünyada hem de ülkemizde artması beklenmektedir.

Yapay zeka alanı altında, makine öğrenmesi, cognitive servisler, doğal dil işleme, görüntü işleme gibi bir çok konu Python ve R komünitesi tarafından gittikçe yoğunlaşan bir destekle sürdürülmektedir. Bunlarla birlikte veri etiketleme alanında kullanılan ve son yıllarda gittikçe kullanımı artan Pybossa projesi Doğuş Teknoloji için open source projelerde yer almak ve katılımcı olmak için iyi bir fırsat olabilir. Python 2.7 ile kodlanan proje tamamen açık kaynak olup GNU lisansı kullanılmaktadır. Bu anlamda ticari olarak kullanıma izin vermektedir. Ülkemizde yapılan doğal dil işlemlerinde en büyük eksikliği yaratan veri sağlanması ve bu verinin doğru şekilde etiketlenmesini daha verimli hale getirebilecek bu proje sayesinde Crowdfunder ve Mechanical Turk gibi ürünlere alternatif üretilebilir. Komünitesi daha yeni yeni büyümekte olan bu proje en başta lokalizasyon açısından desteğe açıktır. Bu sayede Türkçe metinleri veya ülke lokalizasyonlarına uygun olan verileri etiketlemek için öncelikli kullanım avantajları sunan bir yazılıma katkıda bulunulabilir.

Derin öğrenme kütüphanelerinin en popülerlerinden olan Keras ve Tensorflow üzerinde yapılabilecek geliştirmeler Doğuş Teknoloji açısından bu alanda tanınırlık ve ülkemizdeki komünitenin artması açısından fırsatlar sunabilir.

Diğer bir yandan chatbotlar günümüzde oldukça popüler hale gelmiş ve gelecek de daha da artacak bir kullanım oranına erişecektir. Chatbotların kullanılacağı framework projeleri (Microsoft Bot Framework, Webchat) gibi projelere destek vermek, lokalizasyonlarını sağlamak Doğuş Teknoloji açısından önemli bir fırsat doğurabilir. Chatbot projelerinin vermiş olduğu kanal destekleri (Facebook Messenger, Skype, Slack vb gibi) arttırılabilir. Bip üzerinden kanal desteği açmak gibi eklentiler bu tarz projelere kullanım yaygınlığı sağlayacak ve komünitesinin artmasını sağlayacaktır.

IOT alanında ise Autonomous Decentralized Peer-to-peer Telemetry çalışmaları IBM ve Samsung tarafından başlamış durumdadır. Doğuş Teknoloji bu konu üzerinde Volkswagen AG ile birlikte araçlar üzerinde çalışacak olan dock cihazlar üzerinden verileri alıp işleyecek ve bu verileri cognitive servislerle müşterilerine kampanya ve avantajlar olarak çıktı haline getirecek uygulama altyapılarını oluşturabilir. Bir çok markanın sağlayacağı farklı api servislerinin üzerine yazılabilecek olan wrapper’lar ile birlikte otonom araç teknolojisine meraklı programcılar için çok faydalı bir open source proje oluşturulabilir.

Big Data ve IOT birlikteliği ise müşteri tanıma ve kampanya sağlayan yazılım servisleri üretmek için hayli verimli çözümler sunabilmektedir. Bu anlamda turizm alanında (oteller ve restoranlar) kullanılabilecek akıllı cihazlar ve bu cihazlar ile müşterilerin özelliklerinin ve beğenilerinin çıkarılmasını

sağlayan açık kaynak yazılım projeleri önümüzdeki yıllarda artacak olan e-ticaret marketi ile birleştirilebilir.

Blockchain alanında ise ülkemizde gelişmeye açık çok geniş bir alan bulunmaktadır. Tedarik zincirleri, patent hizmetleri gibi bir çok alanda yaratılabilecek bir çok farklı open source proje çok yüksek miktarda katılımcı desteği alabilir.

7)Referanslar

<http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvT3Blbi9Tb3VyY2VfU29mdHdhcmVfSW5zdGl0dXRI>

<http://www.wikizeroo.net/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvT3Blbi9Tb3VyY2VfU29mdHdhcmVfSW5zdGl0dXRI>

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

<https://opensource.com/article/18/2/pivotal-moments-history-open-source>

<https://www.gnu.org/philosophy/why-free.html>

<https://dev.to/bekka/wrap-it-up-being-a-good-citizen-of-open-source-by-devdiscuss-8je>

<https://tldrlegal.com/licenses/tags/open%20source>

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

http://www.bilgitoplumu.gov.tr/Documents/1/Diger/Kamuda_Acik_Kaynak_Kullanimi_Calisma_Raporu.pdf

<https://opensource.com/article/17/11/10-open-source-technology-trends-2018>

Bretthauer, David. "Open source software: A history." (2001).

Lakhani, Karim R., and Robert G. Wolf. "Why hackers do what they do: Understanding motivation and effort in free/open source software projects." (2003).