



Adı: Selim Eren

Soyadı: KAYA

Okul Numarası: 220601052

Bölüm: Bilgisayar Mühendisliği

Ders Adı: Yazılım Mühendisliği Temelleri

Konu: Versiyon Kontrol Sistemleri

Öğretim Görevlisi: Zekeriya Anıl Güven

MENEMEN, 2023

İÇİNDEKİLER

1. Özet	3
2. Versiyon Kontrol Sistemi Nedir?	3
3. Versiyon Kontrol Sistemi Türleri	3
3.1. Yerel Versiyon Kontrol Sistemi.....	3
3.2. Merkezi Versiyon Kontrol Sistemi	4
3.3. Dağıtık Versiyon Kontrol Sistemi.....	4
4. Versiyon Kontrol Sistemi Kullanmanın Avantajları/Zorlukları.....	5
4.1. Avantajları	5
4.2. Zorlukları.....	5
5. Günümüzde Popüler Olan Versiyon Kontrol Sistemleri.....	6
5.1. CVS (Concurrent Versions System)	6
5.2. Apache Subversion (SVN)	7
5.3. Git.....	9
6. Kaynakça	11
7. İletişim Bilgileri	12

Versiyon Kontrol Sistemleri

1. Özet

Bu makalemde Versiyon Kontrol Sisteminin ne olduğunu bilmeyen insanlar için bunun ne olduğu, türleri, avantajları, zorlukları ve bazı popüler Versiyon Kontrol Sistemleri hakkında bilgi verdim. Bu makaleyi okuduktan sonra, Versiyon Kontrol Sistemleri hakkında hiçbir şey bilmiyor olsanız bile kullanabilecek hale gelebilirsiniz. Başlangıç için kolaylıkla kullanabileceğiniz "Git" adlı Versiyon Kontrol Sistemini öneririm.

2. Versiyon Kontrol Sistemi Nedir?

Versiyon Kontrol Sistemi, elimizdeki belgeleri, projeleri, yazılımları, bilgisayarınızdaki neredeyse her dosyayı kontrol altına alarak bunlarda meydana gelen değişimleri, dosyanın neresinde değişim yapıldığını, dosyaların eski sürümlerini görebilmemizi, dosyayı eski sürüme geri döndürebilmemizi ve daha nicelikleri bize sunan sistemdir. Bu sistem bulunduğundan bu yana bir sürü teknolojik gelişme yaşanmıştır. Bu gelişmelere, çağa ayak uydurmak için Versiyon Kontrol Sistemi evrim geçirmiş, bunun sonucunda üç farklı türü ortaya çıkmıştır.

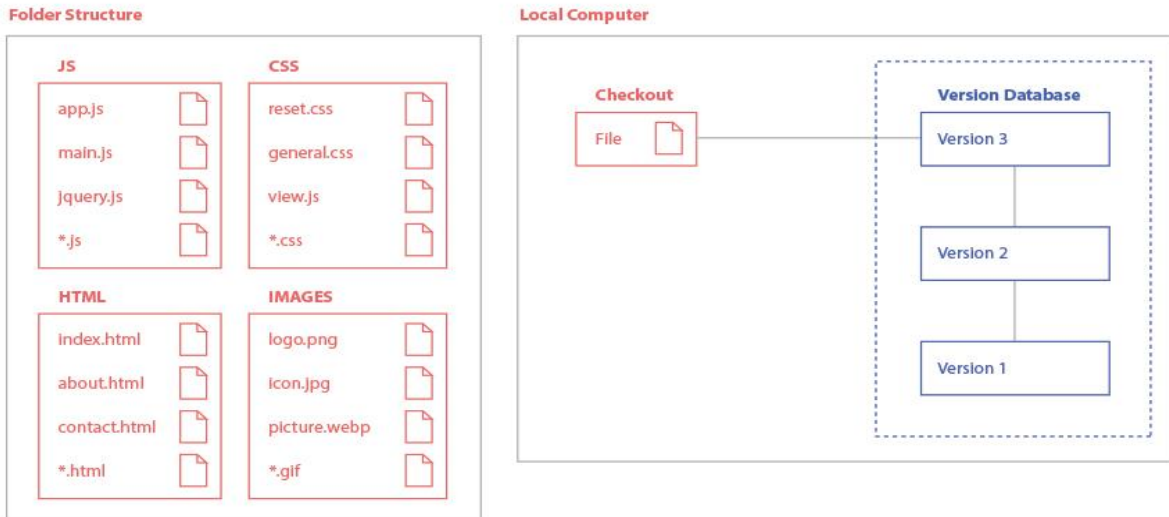
3. Versiyon Kontrol Sistemi Türleri

Günümüzde üç farklı tür Versiyon Kontrol Sistemi bulunmaktadır. Bunları gelişmişlik düzeyine göre aşağıdaki gibi sıralayabiliriz.

- *Yerel Versiyon Kontrol Sistemi (Local Version Control Systems)*
- *Merkezi Versiyon Kontrol Sistemi (Centralized Version Control Systems)*
- *Dağıtık Versiyon Kontrol Sistemi (Distributed Version Control Systems)*

3.1. Yerel Versiyon Kontrol Sistemi

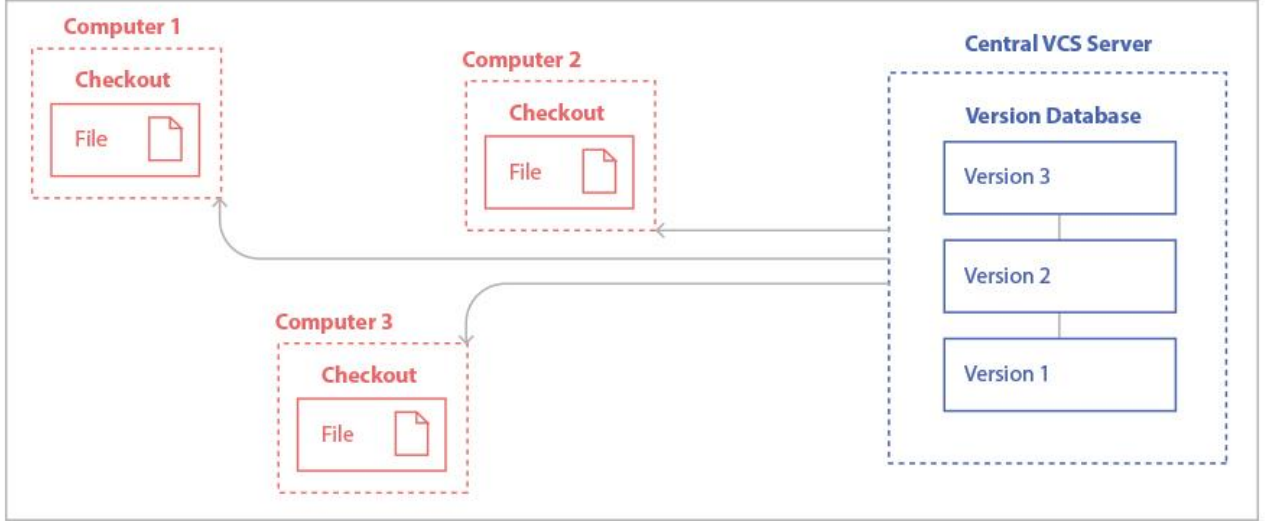
Bir proje yaptığımızı veya web sitesi tasarladığımızı düşünelim. Kullandığımız dosyaları ve oluşan versiyonları tutmak için en iyi yol türüne göre klasörlere ayırıp, tarihine göre sıralamaktır. Bu yol izlenirken ortaya çıkabilecek karışıklığı, hataları önlemek için Yerel Versiyon Kontrol Sistemi bulunmuştur. Bu sistemde dosyalar kullanıcının kullandığı makinede tutulur. Her değişiklik bir versiyon olarak tutulur ve önceki versiyonların görüntülenmesine izin verir. Ne yazık ki bu sistem sadece tek bir kullanıcının etkin olarak çalışmasına imkân verir.



3.2. Merkezi Versiyon Kontrol Sistemi

Merkezi Versiyon Kontrol Sistemi, Yerel Versiyon Kontrol Sistemindeki eksiklikleri gidermek amacıyla geliştirilmiş bir versiyon kontrol sistemidir diyebiliriz. Bu sistemde, önceki sistemde bulunmayan sunucular bulunur. Sunucular versiyon kontrolündeki bütün dosyaları kendi içinde barındırır ve kullanıcılar istediğinde bu dosyalara erişebilir.

Merkezi Sürüm Kontrol Sistemleri



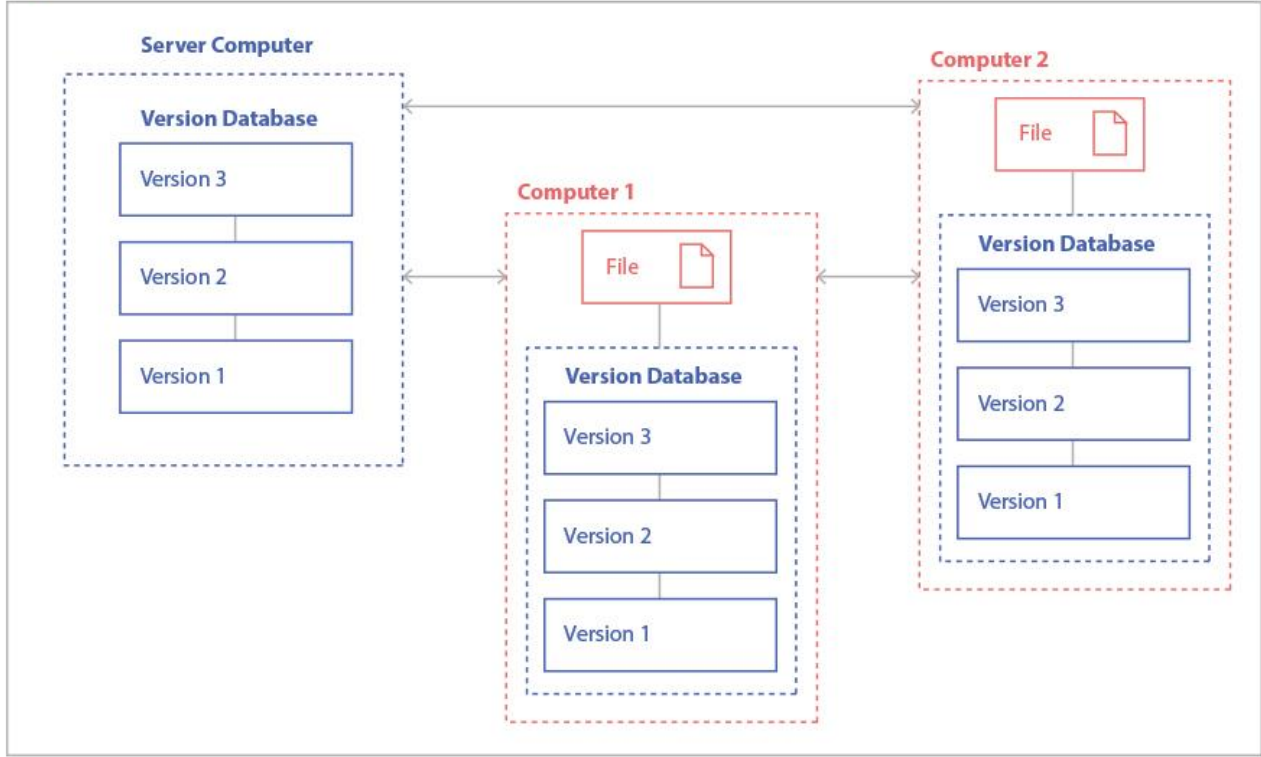
Merkezi Versiyon Kontrol Sisteminde sunucuların bulunması sayesinde sistemde sadece tek bir kişi etkin değil, birden çok kişi etkin bir biçimde projeler, belgeler, dosyalar üzerinde çalışabilir. Dosyalar üzerinde yapılan değişikliklerden herkesin haberi olur, yöneticiler kolaylıkla kimin ne kadar dosyaya erişebileceğini düzenleyebilir. Bunun gibi birçok avantajı olmasının yanı sıra büyük bir dezavantajı da mevcuttur. Sunucularda meydana gelebilecek arızalar herkesin çalışmasını sekteye uğratabilir. Sunucuların sabit disklerinde oluşabilecek bir hasar tüm dosyaların kaybedilmesine yol açabilir.

Günümüzde bu Versiyon Kontrol Sistemi türünü kullanan sistemleri saymak gerekirse; CVS ve Apache Subversion bunların en bilindik örnekleridir.

3.3. Dağıtık Versiyon Kontrol Sistemi

Merkezi Versiyon Kontrol Sistemindeki eksiklikleri, ortaya çıkan sorunların çıkmaması için Dağıtık Versiyon Kontrol Sistemi geliştirilmiştir. Bu sistemde bilgiler sunucularda versiyonlarıyla tutuldukları gibi kullanıcıların bilgisayarında da versiyonlarıyla tutulmaktadır. Bu özellik sayesinde sunucuda meydana gelebilecek olası bir hata sonucunda bilgiler yitirilmemektedir. Kullanıcılar birbirlerinin yaptığı değişiklikleri sunucu sayesinde görebilmekte, istedikleri zaman birbirlerinin dosyalarına rahatlıkla erişebilmektedir.

Dağıtık Sürüm Kontrol Sistemleri



Versiyon Kontrol Sistemi türleri arasında en gelişmiş olması ve en son piyasaya sürülmüş olmasıyla Dağıtık Versiyon Kontrol Sistemi, günümüzde en çok kullanılan türdür. Buna örnek olarak Git, GNU Bazaar, Mercurial ve BitKeeper verilebilir.

4. Versiyon Kontrol Sistemi Kullanmanın Avantajları/Zorlukları

4.1. Avantajları

- Dosyada yapılan her türlü değişikliklerin kaydı tutulur. Bunun sayesinde ihtiyaç durumunda kimin ne yaptığı rahatlıkla görülebilir.
- Kullanıcıların aynı proje, dosya üzerinde aynı anda çalışmasına imkân tanır.
- Ana dosyanın alt sürümleri oluşturup bunlar üzerinde geliştirmeler yapılabilir, uygun görüldüğü takdirde ana sürüme bunlar yüklenebilir.
- Her türlü sürüm kaydedildiği için ortaya bir sorun çıktığında bunun hangi sürümden kaynaklı olduğu rahatlıkla anlaşılıp çözülebilir.
- Dünya çapında, farklı yerlerde bulunan insanların rahatlıkla beraber çalışmasına imkân sağlar.
- Geliştirilen dosyanın, projenin, hangi belge olursa olsun daha hızlı bir şekilde geliştirilmesine olanak sağlar.

4.2. Zorlukları

- İnsanlara bu sistemi kullanma alışkanlığı kazandırmak zordur.
- Sıfırdan kullanılmaya başlandığında insanların aklı kullanırken karışabilir.

5. Günümüzde Popüler Olan Versiyon Kontrol Sistemleri

Günümüzde popüler olan Versiyon Kontrol Sistemlerinden bazıları, üstteki bölümlerde belirttiğim gibi CVS, Apache Subversion ve Git'tir. Bunların tanıtımını, özelliklerini, avantajlarını, dezavantajlarını ve nasıl kullanıldıklarını sırasıyla aşağıdaki bölümden okuyabilirsiniz.

5.1. CVS (Concurrent Versions System)

Tanıtım

CVS için Versiyon Kontrol Sistemlerinin atası demek doğru olur. CVS ilk olarak 1986 yılında Dick Grune tarafından RCS Versiyon Kontrol Sisteminin daha fazla özellik barındıran bir versiyonu olarak geliştirilmeye başlanmış ve 1990 yılında piyasa sürülmüştür.

Özellikleri

- Açık kaynak kodludur.
- Merkezi Versiyon Kontrol Sistemi türündendir.
- Kullanıcının seçtiği, kaydedilmiş sürümlerden birini tekrardan oluşturabilir.
- Birbirinden farklı yerde yaşayan kullanıcılar, beraber tek bir projede rahatlıkla çalışabilir.
- Kullanıcılar aynı projenin farklı alanlarını ayrı ayrı yapsa bile bunları birleştirip tek bir proje haline getirebilir.
- Yapılan her türlü değişiklik kayıt altında tutulur.

Önemli Komutları

- a) cvs checkout: Düzenlemek için kaynak kodu açar.
- b) cvs add: Depoya (repository) yeni dosyalar ekler.
- c) cvs remove: Depodan (repository) bir dosyayı siler.
- d) cvs status: Dosyaların durumunu gösterir.
- e) cvs log: Dosyaların değişme günlüğünü açar.
- f) cvs differ: Çalışan dosyaları depodaki (repository) dosyalarla kıyaslar.
- g) cvs update: Çalışan dosyalarla depodaki (repository) dosyaları senkronize eder.
- h) cvs commit: Depodaki (repository) dosyaları kontrol eder.
- i) cvs tag: Kaynak kodları etiketler.

Avantajları

- Farklı bölgelerdeki kullanıcılar tek proje üzerinde çalışabilir.
- Harika bir çapraz platform özelliği sunar.
- CVS topluluğu tarafından destek almak kolaydır.
- Bol özellikli ve düzgün bir komut satırı istemcisi vardır.

Dezavantajları

- Kaynak kodu depoyu (repository) için bütünlük kontrolü yoktur.
- Kontroller ve değişiklikler çok ince detaylı değildir.
- Dağıtık Versiyon Kontrol Sistemi için yetersiz destek vardır.
- CVS kontrolü dosyadan dosyaya esasına göre yürütür.

Windows Platformunda Nasıl Kullanıldığı Kurulum

Öncelikle CVS'yi kullanmak için bir tane sunucu ve bir tane arayüz programı yüklememiz gerekmektedir. Sunucuyu yüklemek için <http://www.cvsnt.org/> web adresine giriyoruz ve uygun olan sürümü seçiyoruz. İndirdikten sonra kurulum için dosyaya çift tıklayıp açıyoruz ve uygun seçenekleri seçip kuruyoruz. Kurulum bittikten sonra dosyayı kurduğunuz yerden CVSNT Control Paneli açıyorsunuz. Açılan pencerede "Repository Configuration" seçeneğine girip deponun yerini seçiyorsunuz. Bu işlemi tamamladıktan sonra "Server Settings" kısmını açıp "Default domain" yerinden "default" seçeneğini seçiyoruz. Bu işlemi de tamamladıktan sonra depo klasörünüzün ayarlarına girin ve oradan CVS sunucusunu yöneten kişiye tam erişim verin. Bütün her şeyi tamamladıktan sonra istemciyi yüklemeye geçebiliriz. İstemci yazılımı için TortoiseCVN yazılımını kullanacaksınız. İndirmek için <http://www.tortoisecvs.org/download.shtml> linkine girip uygun olanı indirebilirsiniz. Bunu da kurduktan sonra artık CVS'yi kullanabilir hale gelmiş bulunmaktasınız.

Kullanım

Öncelikle yeni bir projenizi depoya eklemek istediğinizi varsayalım. Bu işlem için projenin olduğu klasöre sağ tıklayıp CVS seçeneklerinden "Make New Module" seçeneğini seçmeniz yeterli olacaktır. Böylece projenizi deponuza eklemiş olursunuz. Bunun yanı sıra deponuzdan projelerinizden birini indirmek isterseniz, depoya girip seçtiğiniz projeye sağ tıklayın ve "CVS Checkout" seçeneğini seçin. İşlemi tamamladıktan sonra deponuzdaki dosyanın kopyasını bilgisayarınıza kaydetmiş bulunmaktasınız. Deponuza kaydetmek, silmek veya değişikliğin bilgisini yazmak için dosyalarınıza sağ tıklayıp uygun seçeneği seçebilir, istediğiniz gibi kullanmaya başlayabilirsiniz CVS'i artık.

5.2. Apache Subversion (SVN)

Tanıtım

2000 yılında CollabNet firması tarafından geliştirilmeye başlatılan ve desteklenmeye devam edilen bir Versiyon Kontrol Sistemidir. Daha öncesinde çıkmış olan CVS (Concurrent Versions System) ile olabildiğince uyumlu yapılmaya çalışılmıştır.

Özellikleri

- Açık kaynak kodludur.
- Merkezi Versiyon Kontrol Sistemi türündedir.
- Kullanıcılar aynı projenin farklı alanlarını ayrı ayrı yapsa bile bunları birleştirip tek bir proje haline getirebilir.
- Yapılan her türlü değişiklik kayıt altında tutulur.
- Kopyalama, isim değişikliği, yer değiştirme bilgileri de kayıt altında tutulur.
- Atomik değişiklik sistemi vardır. Her yapılan değişiklik üzerine sürüm alır.
- Yapılan dosya birleştirmeleri de kayıt altına alınır.
- Birden fazla depoya (repository) erişim sağlanabilir.
- Havuzda yapılan tüm değişiklikler kayıt altında tutulur.

- İstenirse dosya kilitleme mevcuttur. Anlık olarak tek bir kullanıcının dosyayı değiştirmesine izin verir.

Önemli Komutları

- a) svn admincreate: Kullanılmak üzere boş bir depo (repository) oluşturur.
- b) svn import: Sürümsüz bir dosya ağacını depoya (repository) yükler.
- c) svn checkout: Depodaki (repository) çalışan bir dosyayı kontrol eder.
- d) svn commit: Değişiklikleri SVN sunucusuna yollar.
- e) svn add: Depoya (repository) yeni bir dosya ekler.
- f) svn delete: Depodan bir dosya siler.
- g) svn list: Depodaki dosyaları görmeyi sağlar.
- h) svn diff: Kullanılan depoyla SVN sunucusundaki depodaki farkları gösterir.
- i) svn status: Çalışan dosya ve dizinlerin durumunu gösterir.
- j) svn info: Yerel veya uzaktaki bir parça hakkında bilgi verir.
- k) svn log: Kayıt mesajlarını gösterir.
- l) svn move: Bir dosyayı bir dizinden diğer dizine taşır veya yeniden adlandırır.
- m) svn merge: İki farklı sürümü çalışan dosyaya kopyalar.
- n) svn revert: Yapılan değişiklikleri geri alır.
- o) svn update: Çalışan dosyayı, depodaki değişikliklerle günceller.
- p) svn shelve: Yapılan değişiklikleri kaydetmeden kayıt altında tutar.
- q) svn help: Kullanılabilecek komutları gösterir.

Avantajları

- Kolay yüklenir.
- Çapraz platform imkânı sunar.
- Farklı bölgelerde insanlar beraber çalışabilir.
- Arayüzü anlaşılırdır.
- Versiyonları geri almak kolaylıkla yapılabilir.
- Stabildir. Neredeyse hiç sorun çıkarmaz.
- Oldukça popüler olduğu için rahatlıkla destek bulunabilir.
- Anlaması kendinden önceki sistemlere göre çok kolaydır.

Dezavantajları

- Merkezi Versiyon Kontrol sistemini kullanır.
- Offline kullanım için çok uygun değildir.
- Depolama alanı kullanımı fazladır.
- Çekme isteği (pull request) bulunmamaktadır.
- Seçilebilecek işlemler daha yalındır.
- Yavaştır.

Windows Platformunda Nasıl Kullanıldığı

Kurulum

İlk olarak Apache Subversion'ın resmi sitesine gidip oradan işletim sisteminize uygun olarak versiyonu seçip indirin. Sitenin adresi <https://subversion.apache.org/packages.html> olarak geçmektedir. İndirdikten

sonra kurulum sihirbazında istediğiniz düzenlemeleri yapıp yükleyin. Bütün bu işlemleri yaptıktan sonra ister komut satırını ister arayüzü kullanarak Apache Subversion'ı kullanabilirsiniz.

Kullanım

Kullanmak için öncelikle bir depo oluşturmanız gerekmektedir. Bunu da komut satırına "svnadmin create /path/to/repository" şeklinde gerçekleştirebiliriz. Oluşturduğunuz depoya bağlantı kurmak için bir çalışma kopyası oluşturmanız gerekmektedir. Bu eylem için şu kodu girmeniz gerekir "svn checkout file:///path/to/repository". Bunun sayesinde deponuzun yerel bir kopyasını oluşturmuş olursunuz. Üzerinde çalışacağınız dosyaları eklemek ve çıkarmak için "svn add dosya1.txt" veya "svn add klasor" gibi komutlar girebilirsiniz. Silmek içinse "svn delete dosya1.txt" komutunu kullanabilirsiniz. Yapılan değişiklikleri kaydetmek içinse "svn commit -m 'açıklama' " şeklinde komut girebilirsiniz. Başka kişilerin yaptıkları değişiklikleri almak için "svn update" yazmanı yeterli olacaktır. Dosyalarınızın tamamı üzerinde yapılan her türlü değişikliği görmek için "svn log" komutunu girebilir ve çoğunlukla kullanacağınız her komutu öğrenmiş olabilirsiniz diyebilirim.

5.3. Git

Tanıtım

2005 yılında Linux çekirdeğinin geliştirilmesinde kullanılmak için Linus Torvalds tarafından geliştirilmiş bir Versiyon Kontrol Sistemidir. Günümüzde 2021 yılı itibarıyla %73 pazar payına ulaşmıştır. Kısacası günümüzde en çok kullanılan Versiyon Kontrol Sistemidir. GitHub ve GitLab gibi yardımcı yazılımlar da mevcuttur.

Özellikleri

- Açık kaynak kodludur.
- Dağıtık Versiyon Kontrol Sistemi türündedir.
- Depoları klonlayabilir.
- Farklı bölgelerdeki insanlar beraber çok rahat bir şekilde çalışabilir.
- İstendiğinde önceki versiyonlara dönüş sağlanabilir.
- Kullanması ücretsizdir.
- Her büyüklükteki dosyayı rahatça kayıt altına alabilir.

Önemli Komutları

- a) git config: Bilgisayardaki git hesabımızı GitHub hesabımıza bağlamayı sağlar.
 - git config -global user.name "Kullanıcı adı"
 - git config -global user.email "Kullanıcı e-mail"
- b) git init: Bilgisayarda bulunan dosyayı depoya (repository) tanımlamamızı sağlar.
- c) git clone: Github'ta bulunan bir depoyu (repository) bilgisayara kopyalamayı sağlar.
- d) git add: Projede yapılan değişiklikleri depoya (repository) ekler.
- e) git push: Oluşturulan projeyi Github'taki kaynağa gönderir.
- f) git pull: Projenin ana dosyasında yapılan değişiklikleri bilgisayardaki versiyona getirilmesini sağlar.

- g) git commit: Yapılan deęişiklikleri kayıt altına almayı sağlar.
 - git commit -m "Açıklama"
- h) git status: Depo hakkında bilgi verir.
- i) git branch: Kullanılan dalları (branch) görmemizi sağlar.
- j) git merge: Üzerinde çalıştığımız iki projeyi birleştirir.

Avantajları

- Günümüzün en popüler Versiyon Kontrol Sistemi olduğu için destek bulabilmek çok kolaydır. Nasıl kullanıldığına dair bir sürü rehber vardır.
- İşlemlerini büyük ölçüde hızlı yapar.
- Depodan dosya çağırmak ve yollamak oldukça kolaydır.
- Dallar arası geçiş yapmak oldukça kolay ve hızlıdır.
- Giriş yapabilme özelliğı vardır.
- Komutları oldukça anlaşılır ve basittir.
- Bir sürü programla oldukça kolay şekilde entegre olabilir.
- Çapraz platform imkânı sunar.

Dezavantajları

- Tamamen, her şeyiyle öğrenmek uzun sürebilir.
- Kullanıcı arayüzü geri kalmıştır.
- Birleştirme işleminde sıkıntılar çıkabilir.

Windows Platformunda Nasıl Kullanıldığı Kurulum

Git'i Windows'a yüklemenin birkaç yolu vardır. En resmi hali Git'in kendi sitesinde bulunmaktadır. Bunun için <https://git-scm.com/downloads> adresine tıklıyoruz. Açılan sayfada Windows için olan yükleme seçeneğine tıklıyoruz ve inmeye başlıyor. İndirdiğimiz dosyayı çalıştırıyoruz ve kurulum başlıyor. Önünüze çıkan pencerede lisans sözleşmesi yazıyor olacaktır. Okuyup bir sonraki adıma geçiyoruz. Kurulmasını istediğimiz yeri seçiyoruz ve bir sonraki adıma geçiyoruz. Yüklennesini istediğimiz şeyleri seçip sonraki adıma geçiyoruz. Başlangıç menüsünde görünmesini istediğimiz ismini yazıp sonraki aşamaya geçiyoruz. Önümüze bu sefer kullandığımız editörü seçmemiz isteniyor. Kullandığımız editörü seçip devam ediyoruz. Önümüze Git'i nasıl kullanmak istediğimiz hakkında bir seçenek sunuluyor. 2. Seçeneğı yani "Git from the command line and also 3rd-party software" seçeneğini seçiyoruz. Böylece hem komut satırından kullanabileceğiz hem de Git Bash programından. Sıradaki aşamadaki bağlantı sağlayacağımız yöntemi seçip devam ediyoruz. Sıradaki pencerede satır sonu stilini seçmemiz isteniyor fakat varsayılan işimi görecektir. Sıradaki aşamada Git Bash'ı kullanacağımız terminali seçiyoruz ve devam ediyoruz. En son olarak ekstra ayarlardan istediklerimizi seçip devam ediyoruz ve yükleme başlıyor. Yükleme bittikten sonra Git Bash kısayolunu bulup çalıştırarak düzgün yüklenip yüklenmediğini kontrol edebiliriz. Git Bash'ı çalıştırdıktan sonra açılan pencereye "git version" komutu yazıp kontrol ediyoruz. Eğer Git sürümü hakkında bilgi veriyorsa doğru yüklediğimiz anlamına gelmektedir. Kurulumu tamamlamış ve artık istediğimiz gibi kullanabilecek hale gelmiş bulunmaktayız.

Kullanım

Öncelikle bir yerel depo oluşturmaya ihtiyacımız var. Komut satırını açın ve komut satırından proje klasörünüze ilerleyin. Projenizin ana klasörüne geldikten sonra "git init" komutunu girin. Bu işlemi gerçekleştirdikten sonra artık projenizin klasöründe ".git" adında gizli bir klasör oluşması gerekiyor. Şimdi sıradaki adım olan "git status" komutunu girin. Bu komutla projenizdeki değişiklikler ve depodaki dosyalar burada gözükecektir. Bu işlemden sonra bütün dosyalarınızı kayıt listesine ekleyip kayıtlarını yapmanız gerekmektedir. Bu işlem için "git add ." komutunu giriyorsunuz. Normalde "git add" yazıp sonuna dosya ismini yazarız ama hepsini eklemek için "." koymamız yeterli olacaktır. "." Tüm dosyaları ekle anlamına gelmektedir. Kayıt işlemi içinse "git commit -m 'kayıt' " komutunu girmeniz yeterli olacaktır. Bunlar girildikten sonra dosyalarınız sırasıyla depoya eklenir. Artık istediğiniz dosyaları ekleyip çıkartabilir, istediğiniz gibi değişiklik yapabilirsiniz. Herhangi bir yanlışınızda projenizi eski versiyonuna döndürmek de artık mümkündür.

6. Kaynakça

- ❖ <https://furkanalaybeg.medium.com/versiyon-kontrol-sistemi-nedir-2f47bb830064>
- ❖ <https://www.argenova.com.tr/versiyon-kontrol-sistemi-nedir>
- ❖ <https://argenova.com.tr/en-iyi-versiyon-kontrol-sistemleri>
- ❖ https://tr.wikipedia.org/wiki/Sürüm_kontrol_sistemi
- ❖ <https://git-scm.com/book/tr/v2/Başlangıç-Versiyon-Kontrol>
- ❖ <https://ceaksan.com/tr/versiyon-kontrol-sistemi-vcs-nedir>
- ❖ https://en.wikipedia.org/wiki/List_of_version-control_software
- ❖ https://en.wikipedia.org/wiki/Concurrent_Versions_System
- ❖ <https://blog.plasticscm.com/2010/11/version-control-timeline.html>
- ❖ <https://tr.theastrologypage.com/concurrent-versions-system>
- ❖ <https://www.quora.com/What-are-some-challenges-when-a-team-starts-using-version-control-the-first-time>
- ❖ <https://www.techtarget.com/whatis/definition/Concurrent-Versions-System-CVS>
- ❖ <https://www.codingninjas.com/codestudio/library/concurrent-version-system-guide>
- ❖ <https://www.codingninjas.com/codestudio/library/apache-subversion-guide>
- ❖ <https://subversion.apache.org/features.html>
- ❖ https://tr.wikipedia.org/wiki/Apache_Subversion
- ❖ https://svnclient.oxygenxml.com/why_subversion.html
- ❖ <https://www.perforce.com/blog/vcs/svn-commands-cheat-sheet>
- ❖ <https://www.trustradius.com/products/apache-subversion/reviews?qs=pros-and-cons#reviews>
- ❖ <https://coderspace.io/blog/git-nedir-nasil-kullanilir/#header-index-3>
- ❖ [https://tr.wikipedia.org/wiki/Git_\(yazılım\)](https://tr.wikipedia.org/wiki/Git_(yazılım))
- ❖ <https://rubicotech.com/blog/advanced-git-features/>
- ❖ <https://www.trustradius.com/products/git/reviews?qs=pros-and->

[cons#reviews](#)

- ❖ <https://www.msoguz.com/2020/03/git-nasl-kurulur.html>
- ❖ <https://medium.com/@emrullahluleci/git-ve-github-nas-l-kullan-l-r-7d3cc886b77e>
- ❖ <https://microchipdeveloper.com/mplabx:subversion-windows>
- ❖ <https://www.studocu.com/en-us/document/george-mason-university/fpga-and-asic-design-with-vhdl/tutorial-work-1-concurrent-version-system-based-on-cvs-nt-and-tortoise-cvs/757696>

7. İletişim Bilgileri

<https://github.com/selimerenkaya>

<https://medium.com/@selimerenk41>