Slide Number 1

Hi, this is Faraidoon Naghdali, and my todays presentation is about Source Code Control with Git. And I will show you how to use Git, the popular open-source version control software, to manage changes to your source code and text files. It doesn't matter if you're a complete beginner or if you have some prior experience I’m going to cover all the fundamentals that you need to know Git as a source code management.

Let's get stated by talking about Source Code Control, which can be an overwhelming idea when you first start out. My goal is to give you guys an overview of Git and its advantages.

Slide Number 2

What is source code control?

 Let's very quickly review the fundamentals so we're all on the same page. First just to cause a bit of confusion Source Code Control is often called Version Control. Same thing, different name. But the idea of Source Code Control or Version Control is to keep every version of your source code available to you so that if you decide to go back or something goes wrong, you still have your source code in each iteration.

This avoids losing any work.

Git has a terrific pedigree. It was created by Linus Torvalds, who you may know is the man behind Linux. It has quickly become the source code control system of choice for open source and for many others, including some of the really big players in technology. And the reason for that is it's blazingly fast. It's designed for easy and rapid branching and merging. And it's designed to ensure that nothing is ever lost.

Slide Number 2

What is repository?

Let's talk about what a repository is. And don't panic. This will be painless. It starts with repositories. Repositories are where your snapshot of your code are kept. In some version control systems repositories are centralized. You check your code out of a central repository, and you check it back in. Git is decentralized. Every Git user has his own repo. This has tremendous advantages.

You can work offline, and should anything happen to your repo anyone else working on the same project can get you up and running in no time. There is a central server, but it exists only for collaboration among teams and to get your work offsite for safe keeping. Should that central server explode your data is still safely tucked away in your own local repo. Your entire local repository is in a folder within your project folder.

This hidden repo folder is called .git. You can look in this folder, but don't touch. These centralized servers are known as remote repositories.

There are a lot of options.

You can host your own,

or you can use one of the public repositories like Bitbucket or GitHub.

We'll be using Bitbucket, because it's free, and you can keep your repositories private and it integrates nicely with SourceTree.

Slide Number 2

Slide Number 2

Slide Number 2

Slide Number 2

Slide Number 2

Slide Number 2