**中国矿业大学计算机学院**

**2020 级本科生实验报告**

课程名称 软件工程实践

报告时间 2022-12-12

学生姓名 李建业

学 号 12203743

专 业 计算机科学与技术

任课教师 刘迎春

实验五 源代码版本管理实验报告

**实验目的**

学习SVN 和 Git 源代码版本管理工具，掌握源代码版本的管理采用 TortoiseSvn (管理员客户端) +AnkhSvn (VS2008 插件) +VisualSwnServer 在 Visual Studio 环境下进行代码版本管理。采用相关版本控制工具实现 Eclipse 环境下的代码版本管理。

实验步骤

**注：本实验所有的代码和仓库名字都有本人名字（lijianye）来标识**

SVN 和 Git 源代码版本管理工具

1.下载svn服务端。

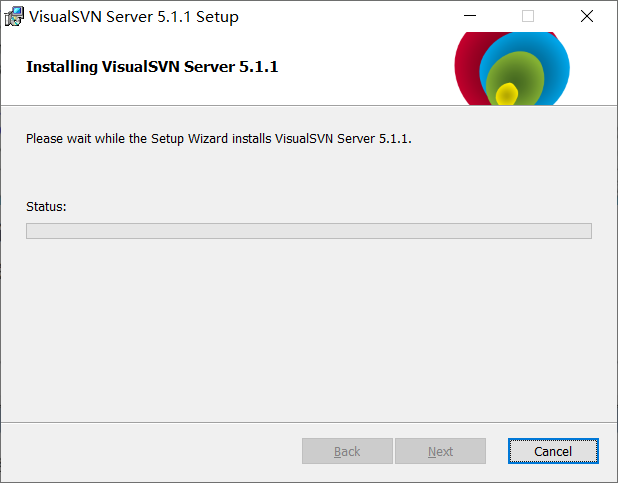


图1下载svn server

2.下载svn客户端。

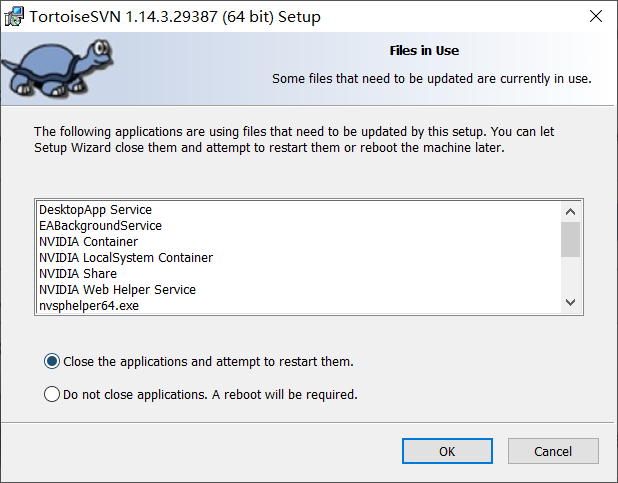


图1.2 svn客户端下载

3.创建svn仓库，名字为repositoryoflijianye。

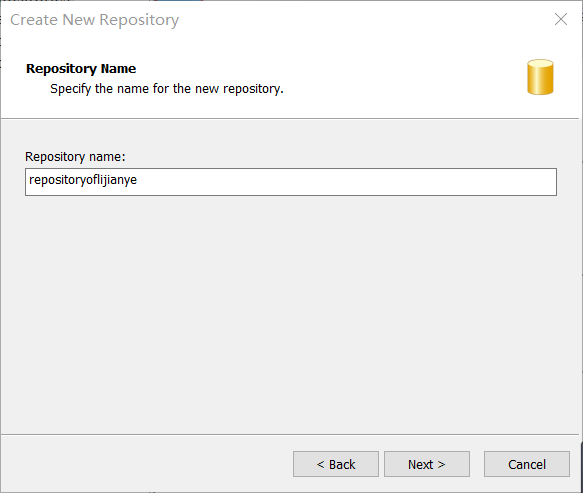


图1.3 svn仓库创建

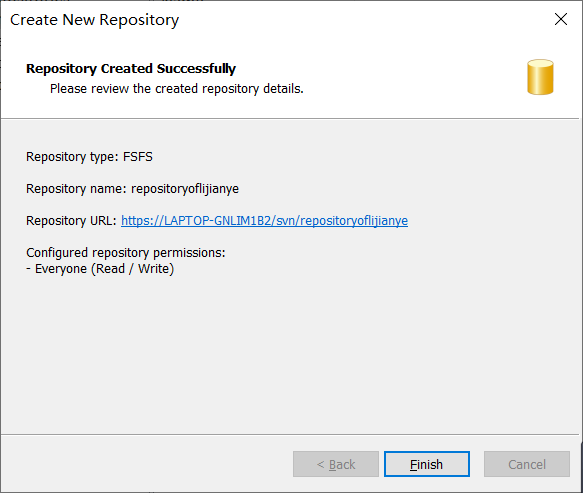


图1.4创建仓库成功

4.创建用户。

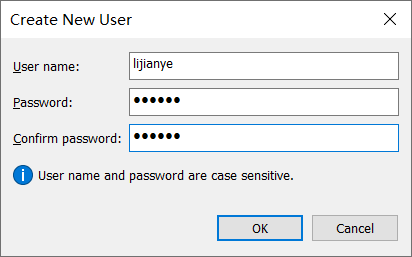


图1.5创建用户成功

5.在本地检出一个仓库文件夹。

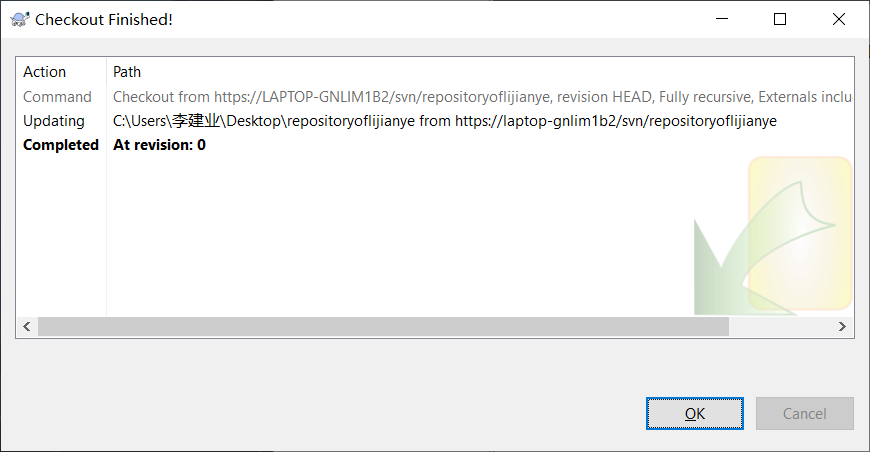


图1.6检出成功

6. 将一个项目文件放入到本地仓库中（具体操作为拖拽我的项目文件到这个文件夹下，然后右键此文件夹选择TortoiseSVN的子菜单然后选择add）。

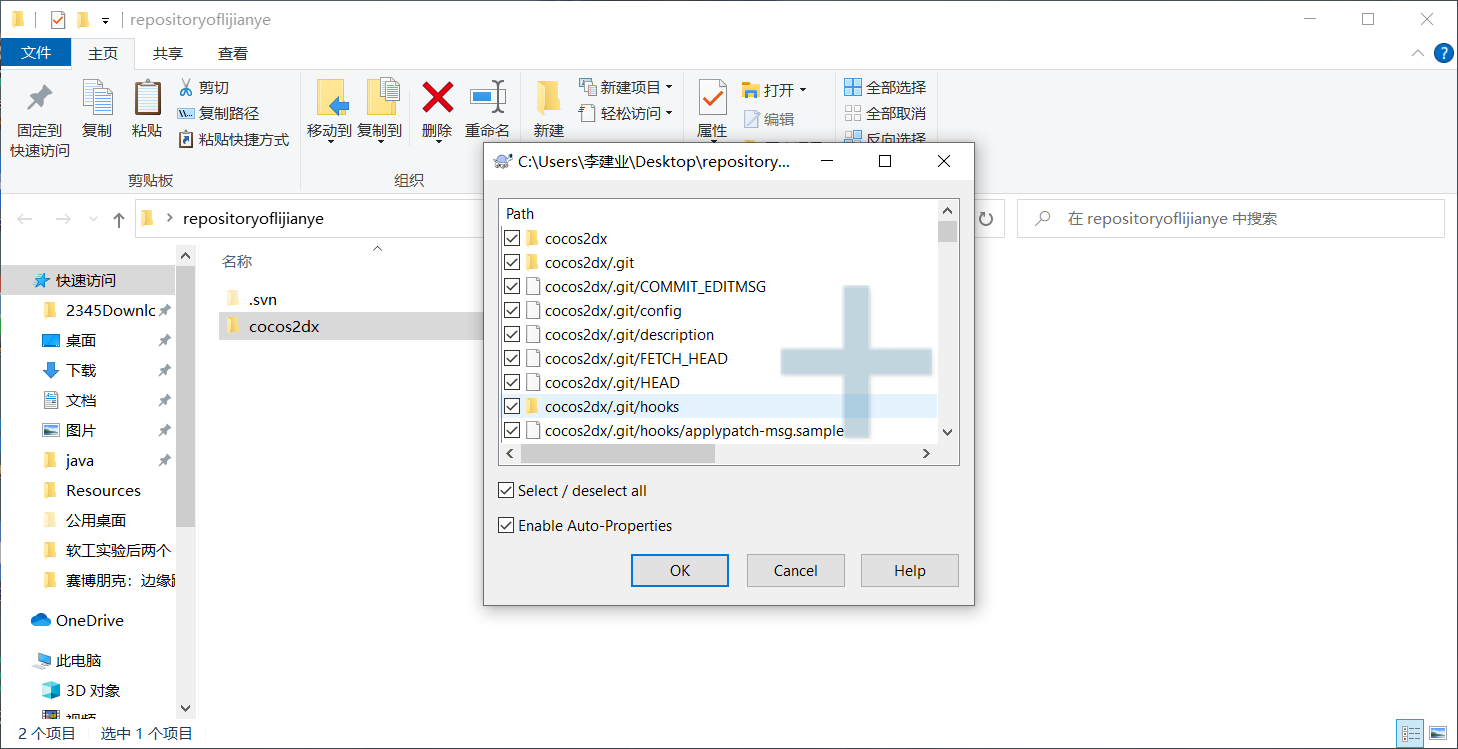


图1.7将项目添加到版本计划中

7．提交项目到服务端（具体操作为右键此文件夹 选择SVN commit）。

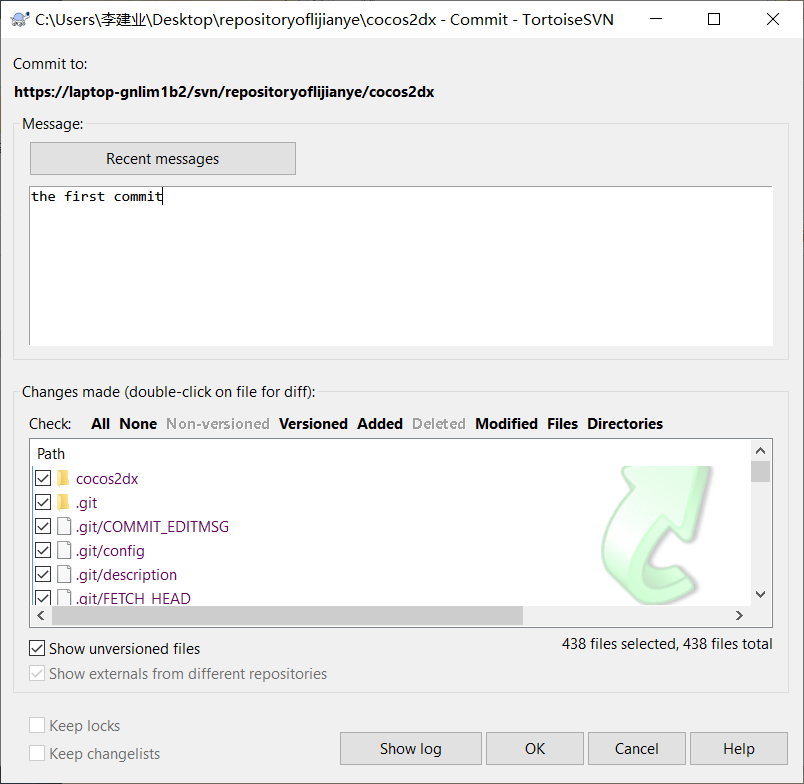


图1.8填写提交日志提交到服务端

8.在服务端的管理器中查看。

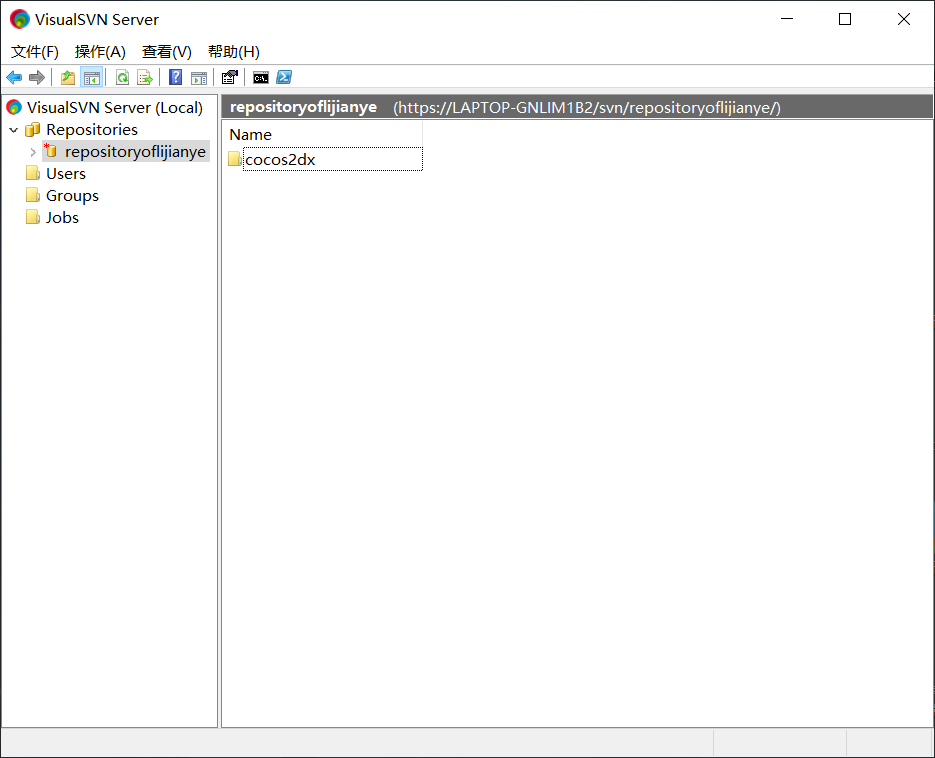


图1.9服务端的项目文件

9.当别人往仓库里提交了代码，可以通过update方式将代码与服务端同步。



图1.10 SVN Update

10.查看日志可以查看项目文件进行了怎样的修改

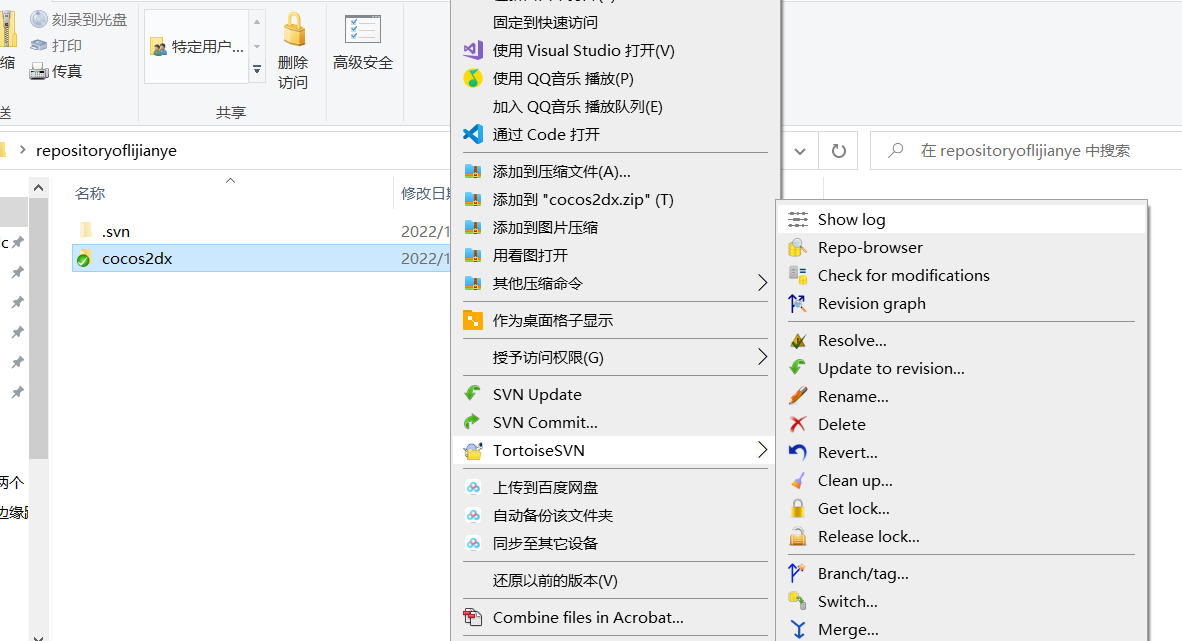


图1.11 查看日志

11.选中某个版本以更新到此版本

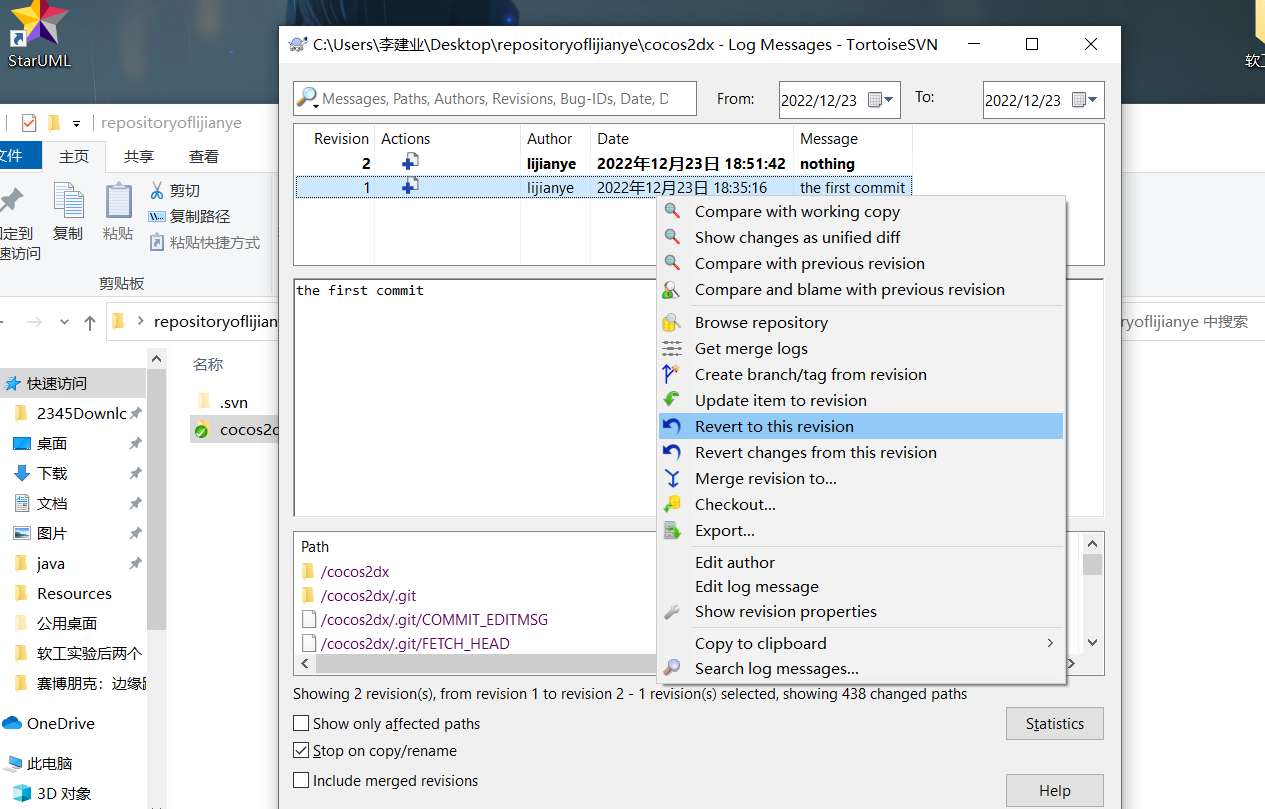


图1.12 版本更新

**Git源代码版本管理**

1.创建一个github仓库（因之前一直用git做一些项目的版本管理，所以本人已经拥有了github账号，账号注册和git安装将不做演示）

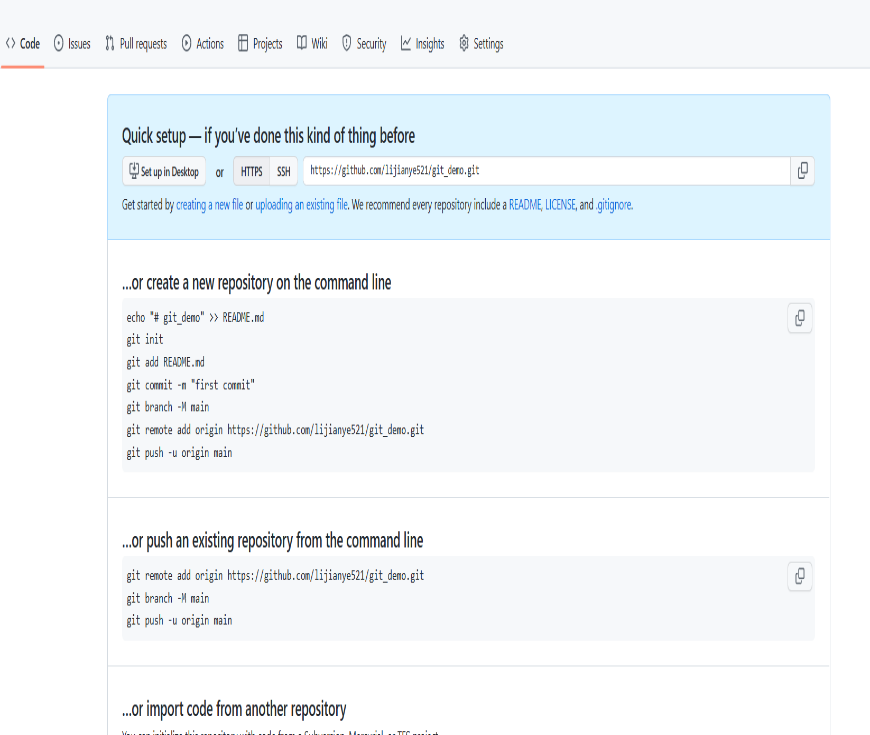


图1.13 仓库创建

2.按照github的命令行提示将代码上传到github仓库中

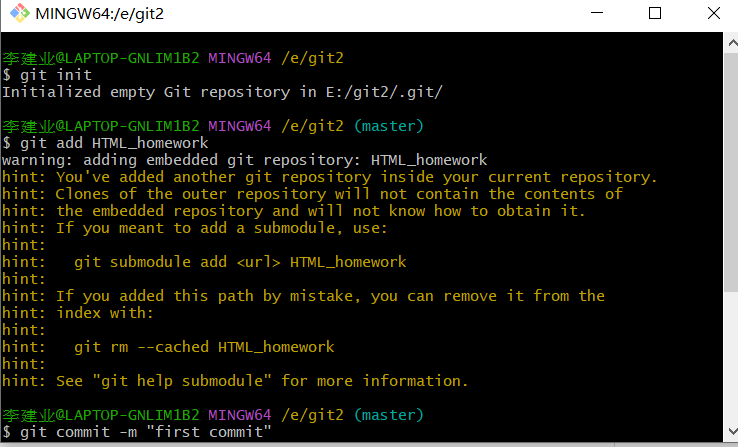


图1.14代码上传

3．Git进行源代码版本管理

克隆仓库命令 gitclone

拉取仓库的代码git pull <address>

4.这里介绍一个我自用的git神器，githubdesktop，这个省去了命令行的操作，使得所有的次操作图形化，非常的便捷。无论是代码更新还是版本回溯都很方便。

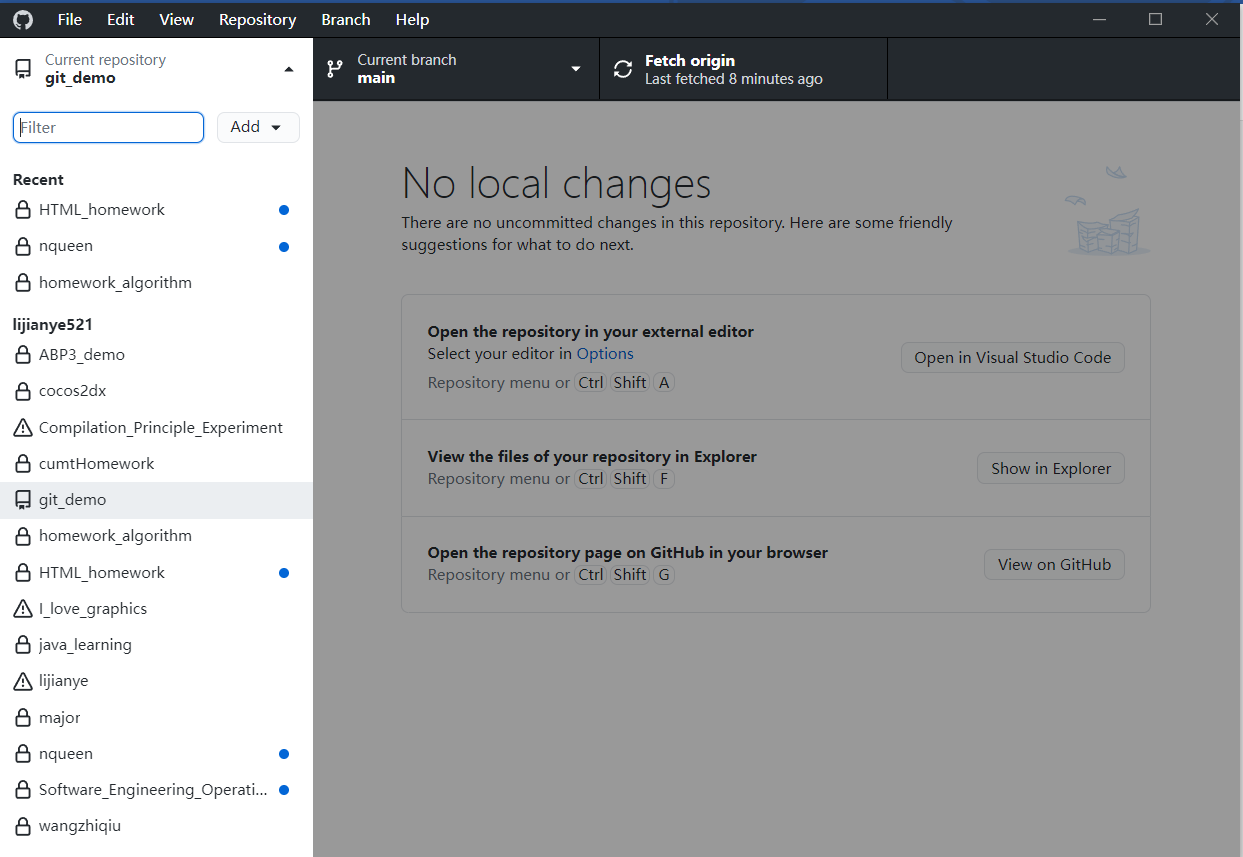


图1.15 github图形化界面

5.查看历史更改

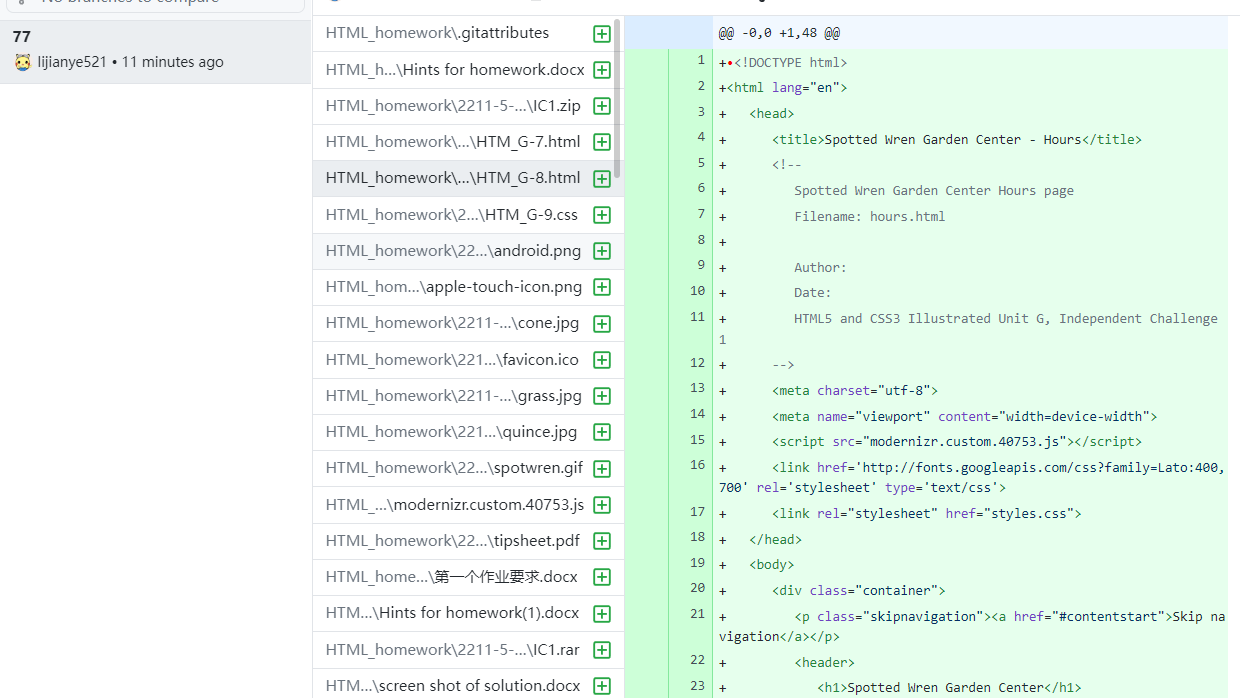


图1.16 历史更改

Visual studio环境下源代码版本管理

1.安装visual studio插件

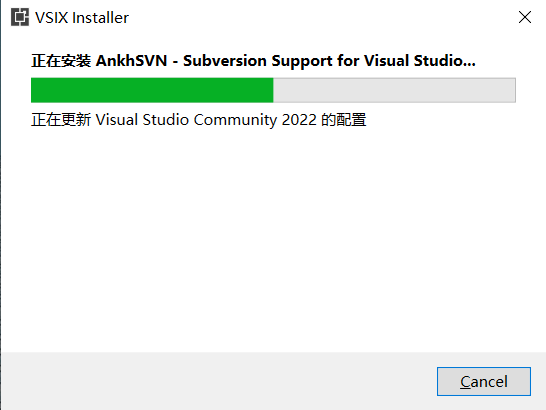


图2.1 插件安装

2.将版本管理工具改成Ankhsvn

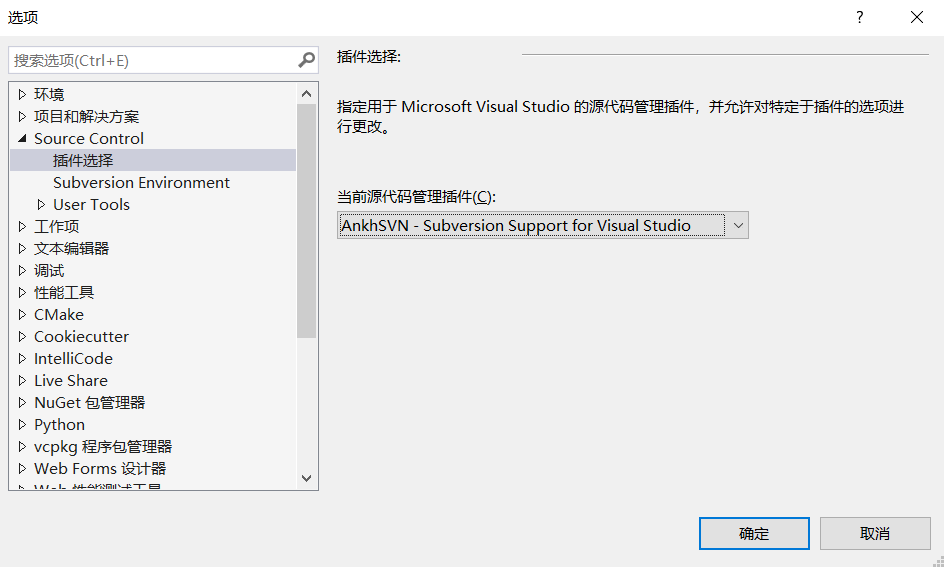


图2.2更换插件

3.将项目添加到subversion

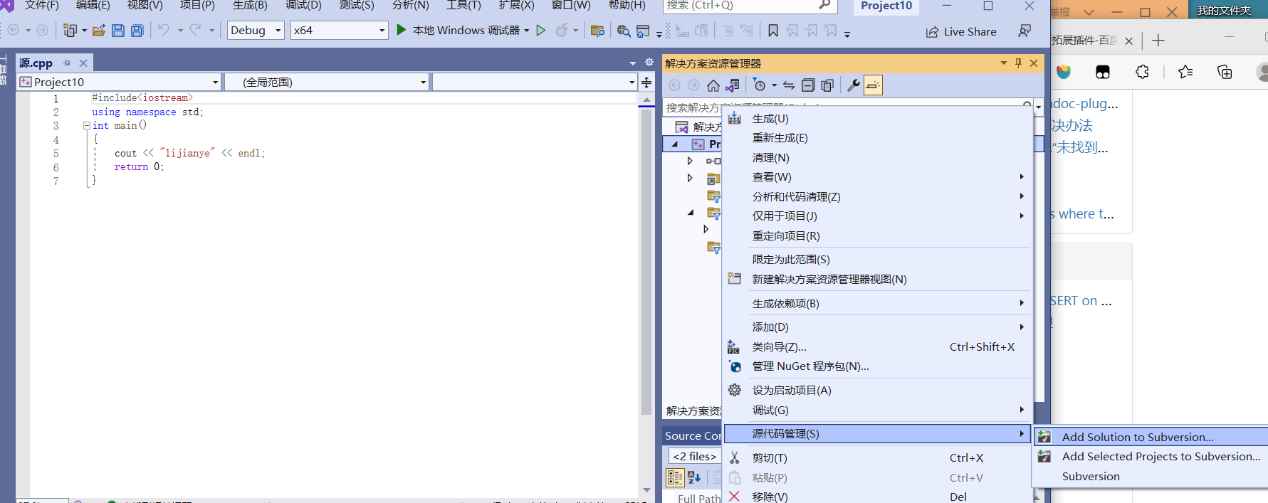


图2.3添加到subversion

4.url选择之前的仓库

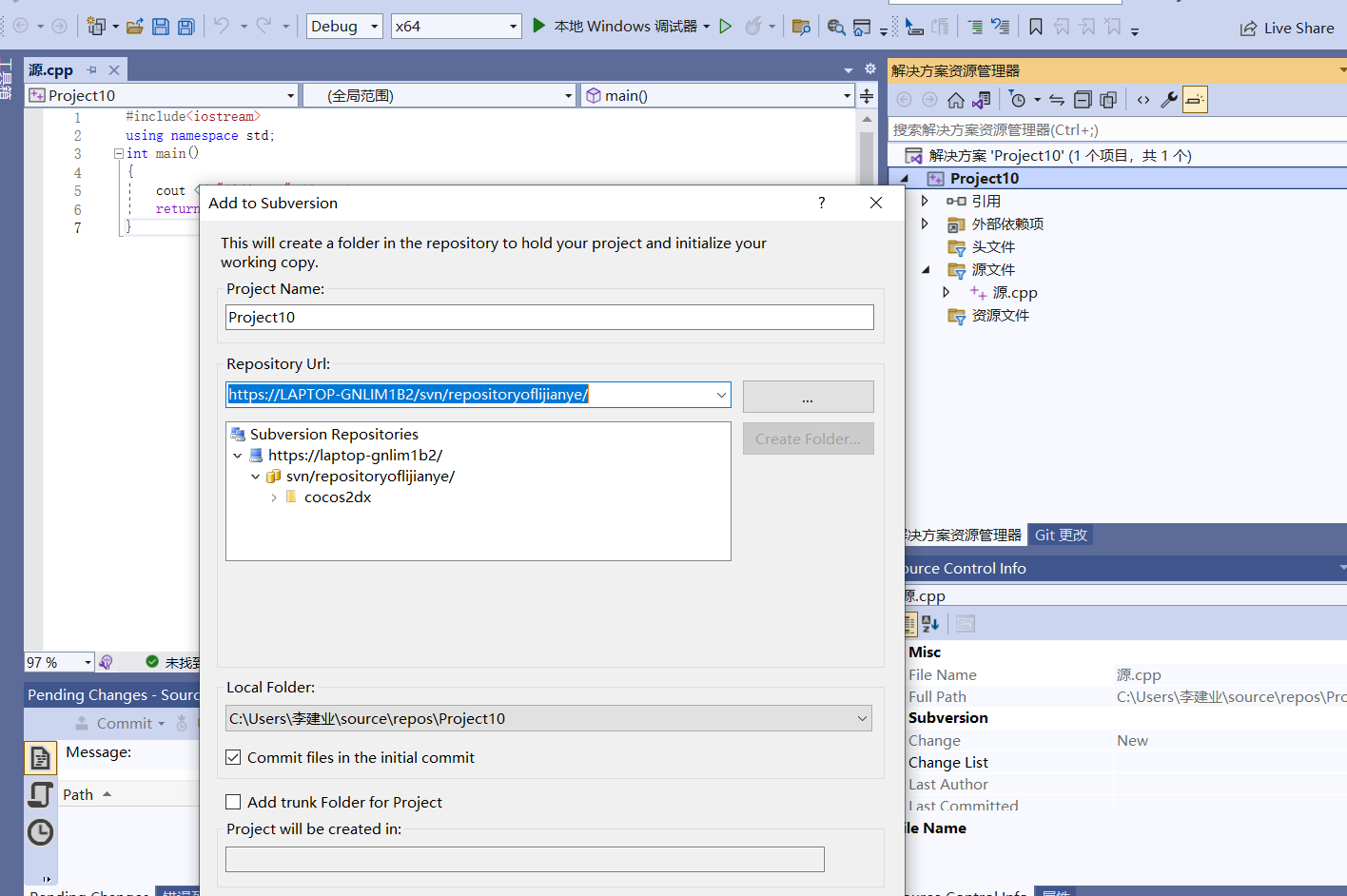


图2.4选择仓库

5.填写日志

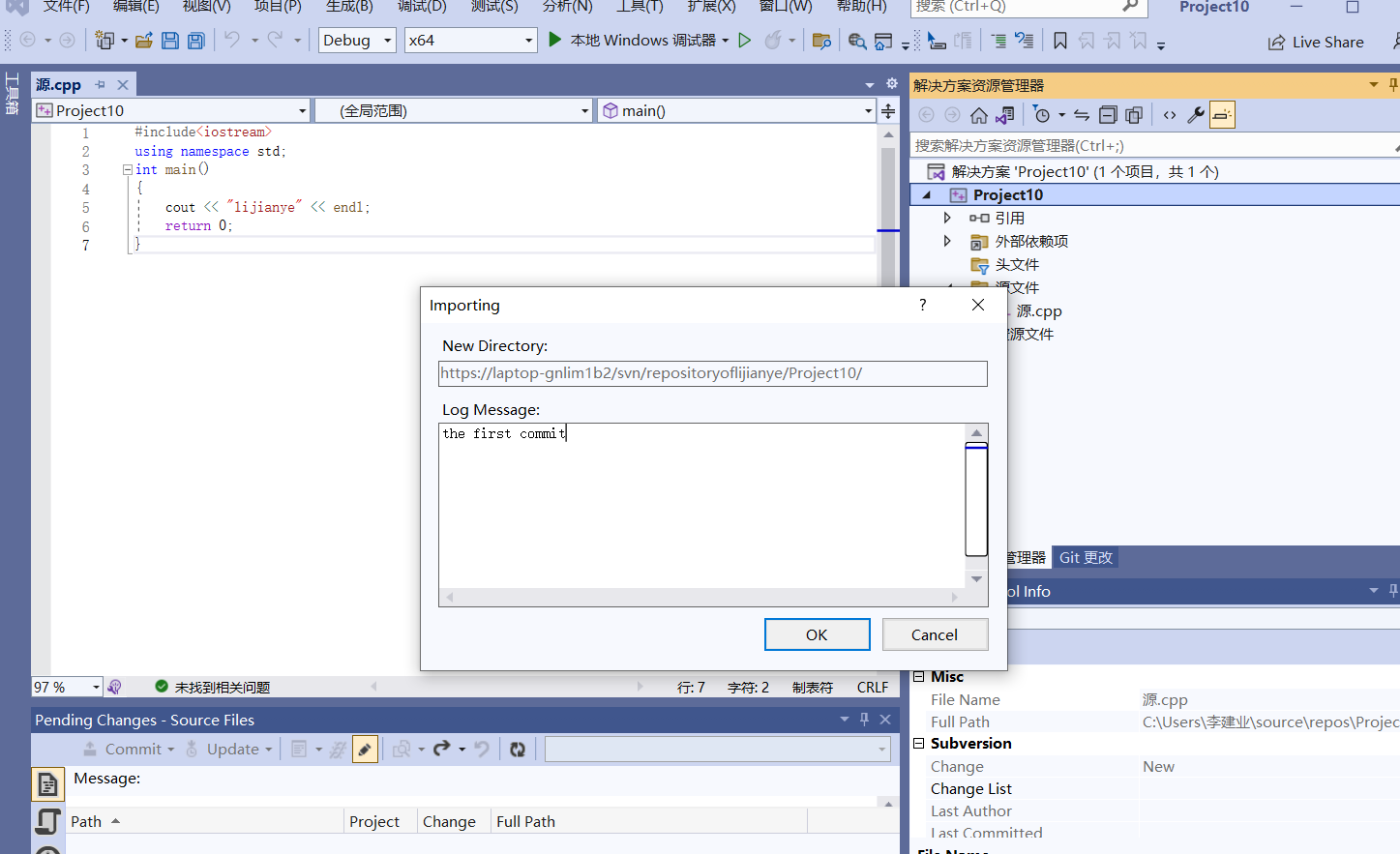


图2.5填写日志

6.代码比较（本地代码与服务端的比较）

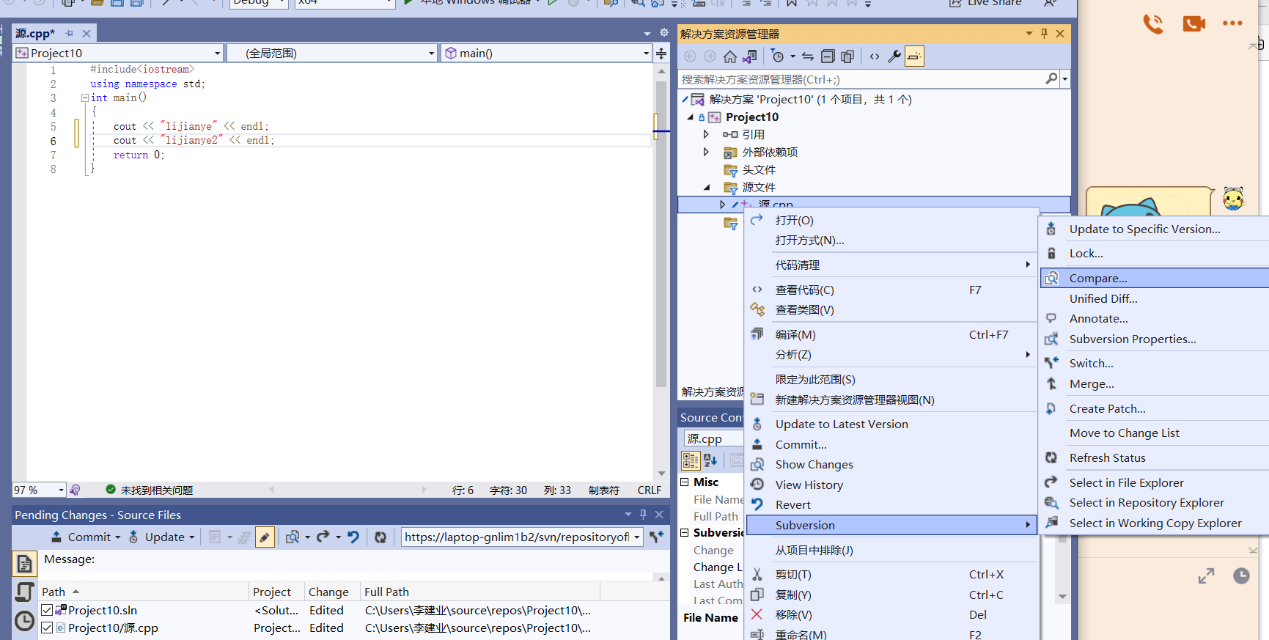


图2.6代码比较

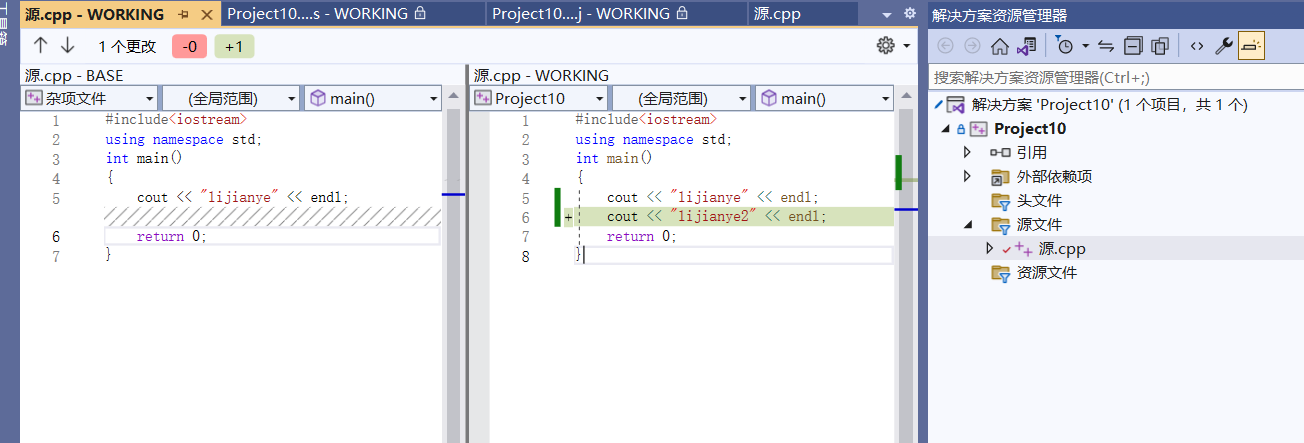


图2.7与服务端代码比较

7．查看代码历史更改

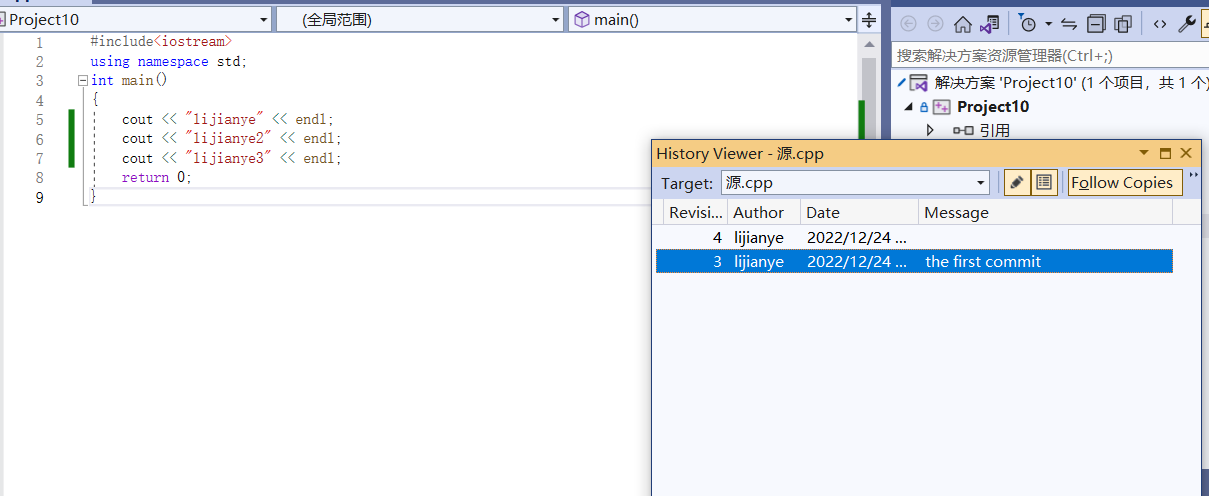


图2.8查看历史更改

Eclipse环境源代码版本管理

因为Eclipse的git方法的源代码管理与git本身的方法一模一样，这里主要演示Eclipse使用subclipse插件进行源代码管理

1. 安装Eclipse的源代码管理工具subclipse

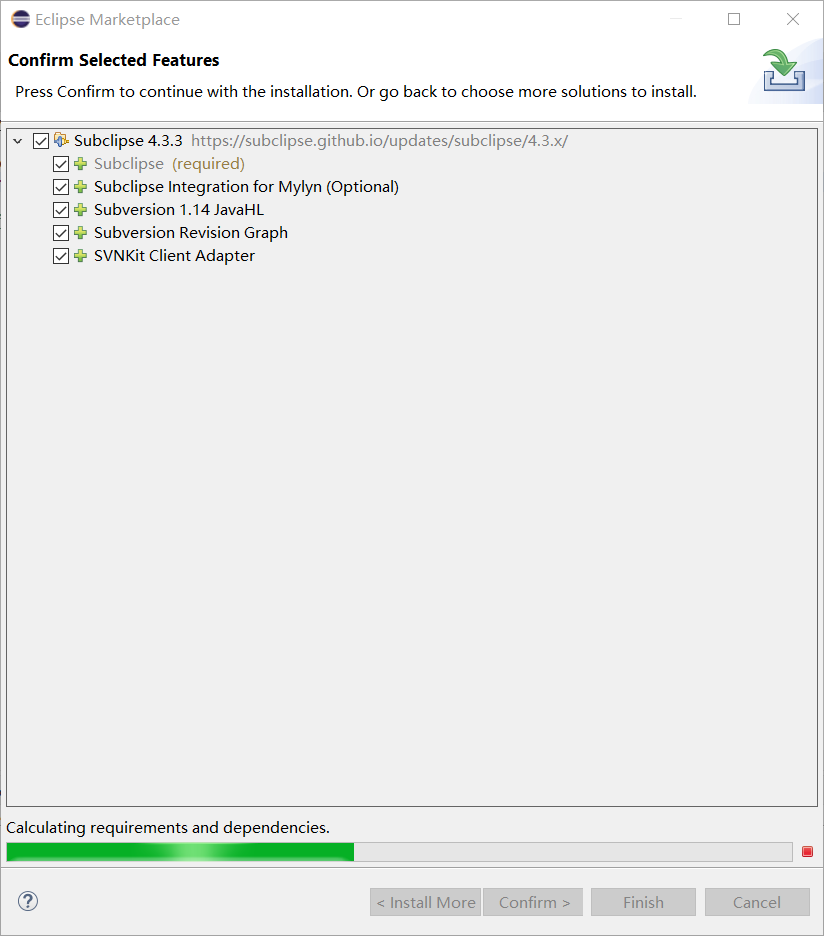


图3.1插件安装

1. 配置资源库位置

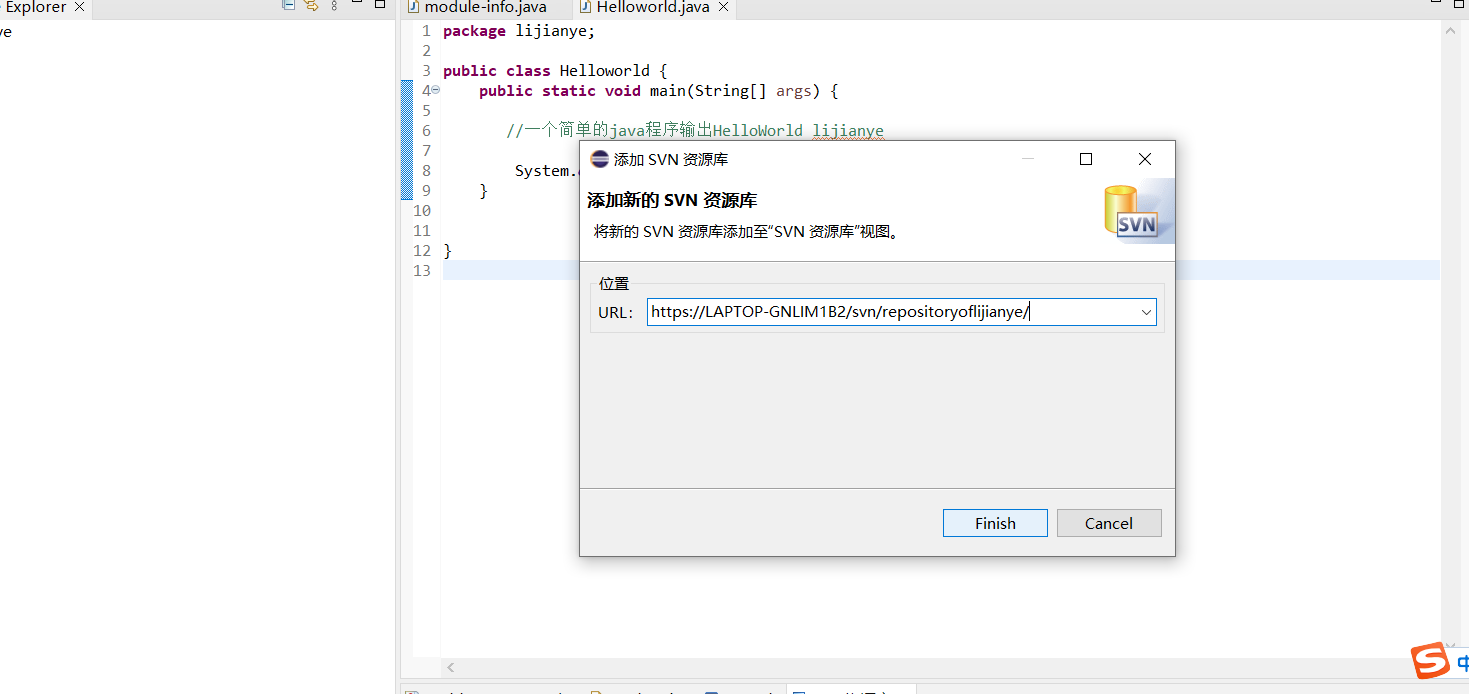


图3.2资源库配置

1. 选中工程中的share project

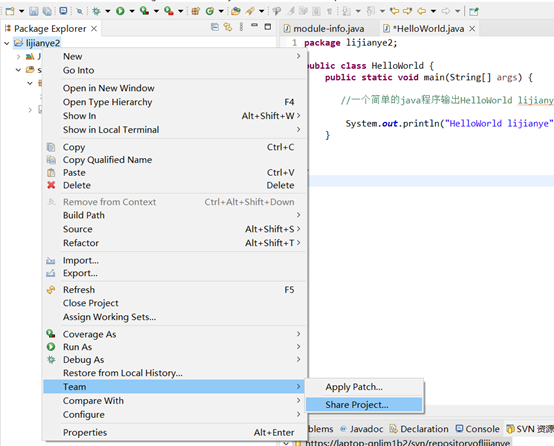


图3.3 share project

1. 选择使用svn的方式

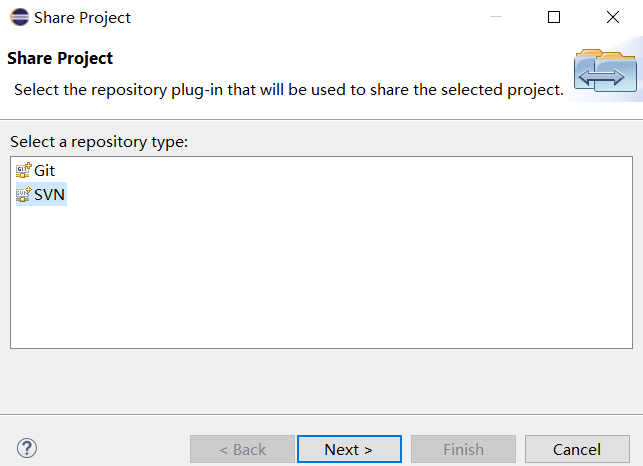


图3.4 选择svn的方式分享工程

1. 提交代码到仓库

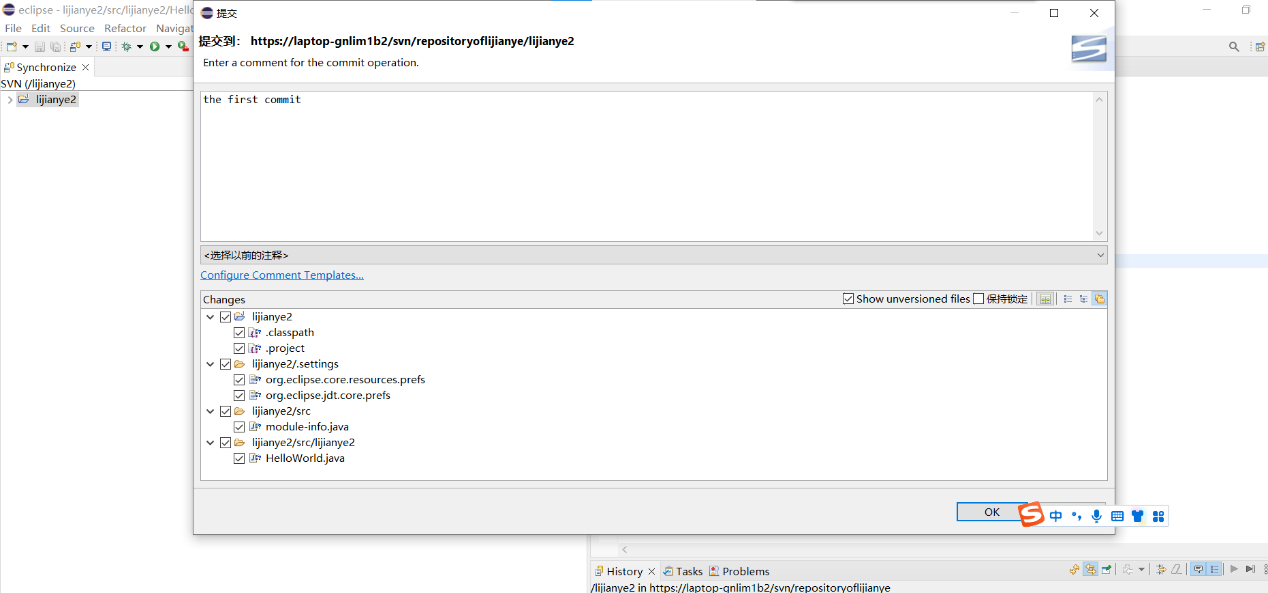


图3.5提交代码到仓库并填写提交注释

6.同步操作

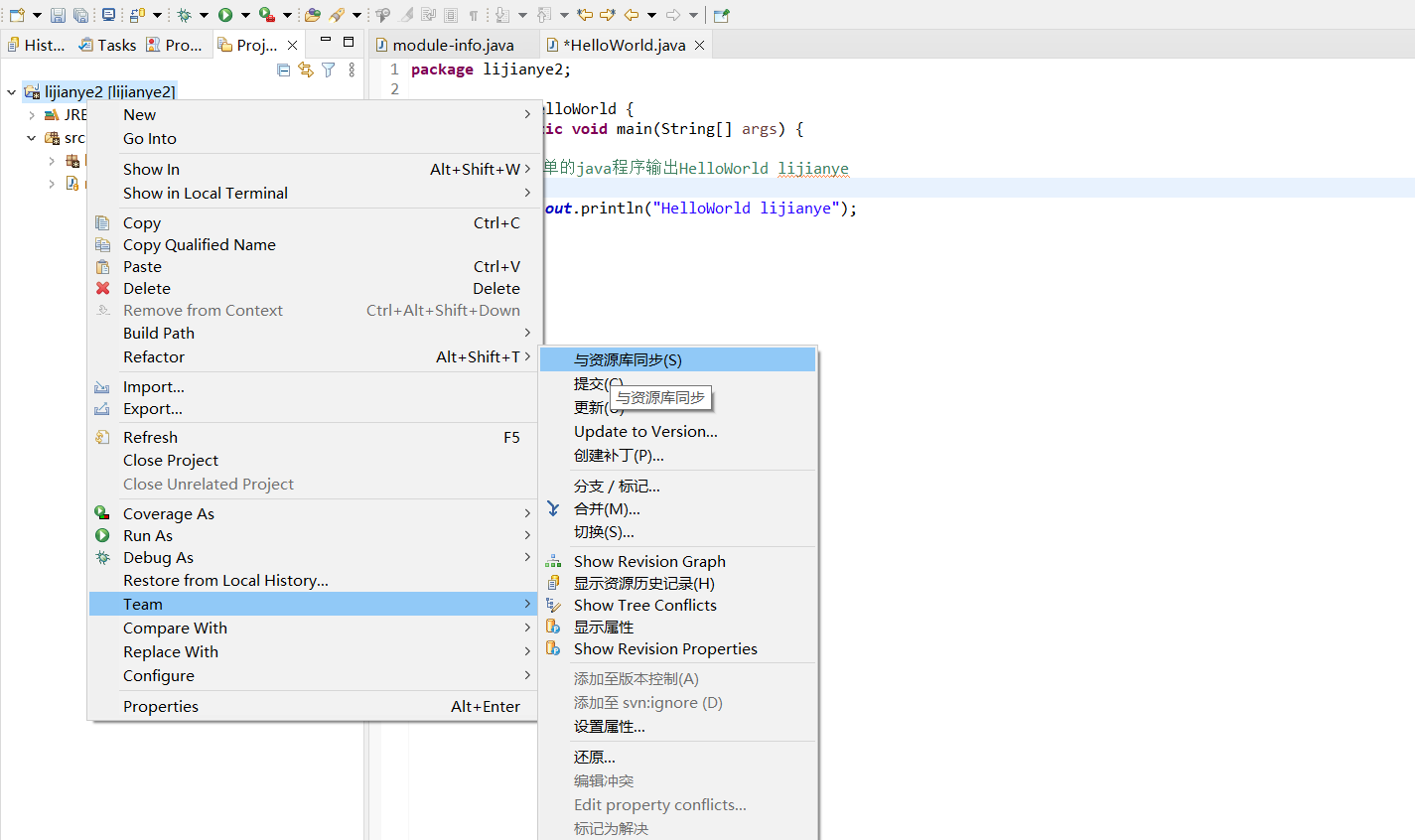


图3.6同步操作

1. 与代码仓库的代码进行比较



图3.7代码比较

8.将本地版本修改为代码仓库中最新的版本

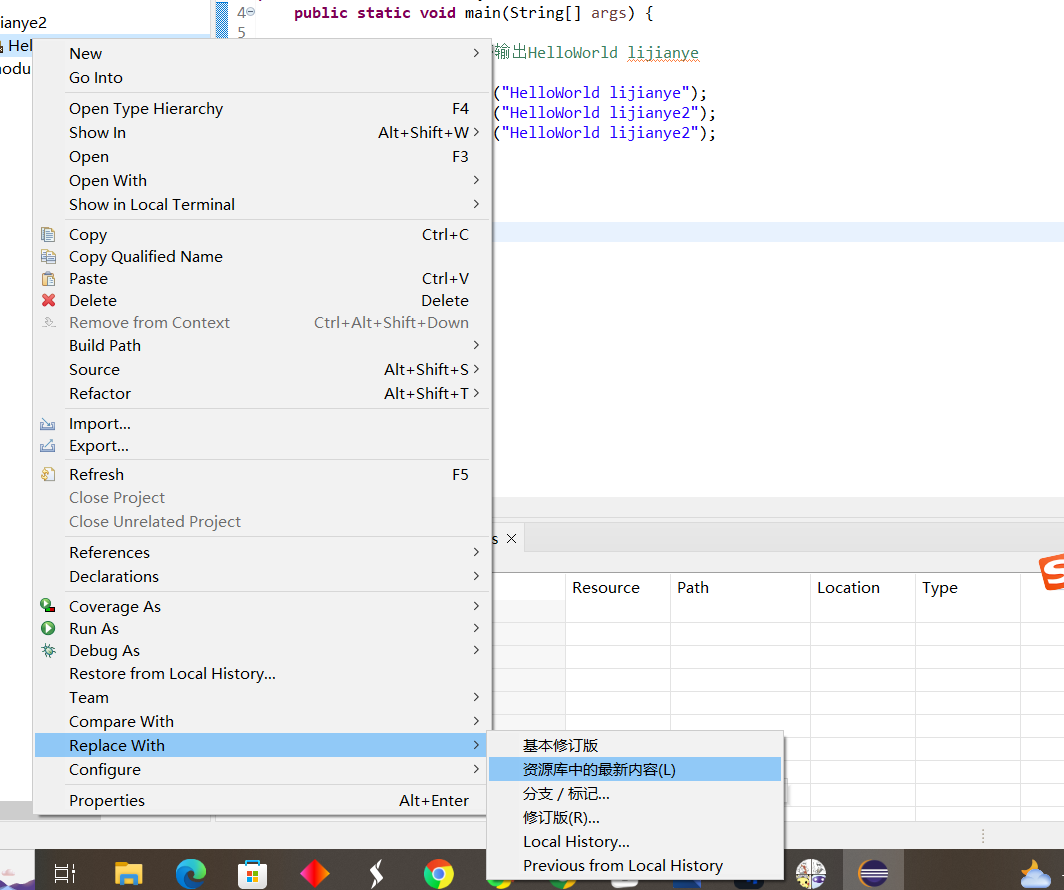


图3.8 版本修改

实验体会

通过本次实验我已经熟练的掌握了svn，git这两种版本管理工具，然后又掌握了Visual studio和Eclipse环境下如何进行源代码版本管理，如何进行代码同步，版本回溯。此外在学有余力的情况下，我还掌握了锁定当前本地代码，分支创建，查看更改日志等进阶操作。无论实在学习还是工作过程中，源代码版本管理与我们息息相关，相信日后自己在本次试验后，能够更方便快捷的管理自己的代码。