Overall Explanation:

The design of our WTF system is as follows. The server is ran by calling ./WTFserver and then specifying the port number to use. The server will then run until SIGINT. The client is run with a variety of commands, which are configure, checkout, update, upgrade, commit, push, create, destroy, add, remove, and currentversion. While some of these client commands do not require the client to connect to the server, some do. For the ones that do, the client attempts to connect to the server and will continue to try so until SIGINT or it successfully connects. When the client connects, the server receives a buffer from the client starting with "# ", where # is a single character representing the request of the client. The server then creates a new pthread running the appropriate method to handle the client's request. This is done so multiple clients can connect to the server at once. When the requested service is completed, the process ends.

Client Functions

Below are a list of the client commands:

./WTF configure <IP Address> <port #>

Stores the IP address and port# of the server for future connections. The client cannot connect without being properly configured.

./WTF checkout <project name>

The client will receive a copy of most recent update of the specified project. If the project does not exist on the server, this command will fail.

./WTF update <project name>

For the specified project, the client will compare its own manifest for that project to the server's manifest for that project. In the case that the client's manifest is not up to date, a .Update file is created with a list of updates the project needs to make on the client side. Update will also specify if there is a conflict between the client and server in trying to update a file. This command will fail if the project does not exist on the client or server.

./WTF upgrade <project name>

This command is used to make the updates specified in the update command. Upgrade will fail if there is a conflict between the server and client, if the project does exist on the server or client, or if there is no .Update file. This is all done to prepare the project for a commit.

./WTF commit <project name>

Commit is used to prepare the client to make a push to the server. If the client version of the project is up to date, a .Commit file is created with all the changes the server will need to make during the upcoming push. If the project is not fully updated or the project does not exist on the client or server, the command will fail.

./WTF push <project name>

Push applies all of the changes that are listed in the .Commit file to the server’s version of the project. If the client does not have a .Commit file ready or the project doesn’t exist on the project or server, the command will fail.

./WTF create <project name>

Create is used to create to create a project on the server. If the project already exists on the server, create will fail.

./WTF destroy <project name>

Destroy will fully delete a project on the server. If the project does not exist on the server, this command will fail.

./WTF add <project name> <filename>

Add is used to change the client manifest to let it know the user has added a file to the project. If the file or project does not exist, add will fail.

./WTF remove <project name> <filename>

Remove is used to change the client manifest to let it know the user has removed a file from the project. If the project is not in the client manifest, remove will fail.

./WTF currentversion <project name>

Current version is used to get the current version of a project and all the files it contains on the server. If the project does not exist on the server, currentversion will fail.

Side Note:

Unfortunately, the history and rollback commands are not included.