

Conner Knight: connerk, Evan Blajev: eblajev, Julius Christenson: juliusc, Jacob Ward: jward23, Sarah Zhou: sarahsz

## User Manual

### Problem & Motivation

Commit-min automatically splits up your commits to differentiate bug fixes from other commits. For example, while a developer is working on finding and fixing a bug, they may also clean up surrounding code so it is easier to read and reason about. Once the bug has been fixed, the commit will include the code that fixed the bug as well as the extra refactoring, and possibly new functionality. Our tool will automatically split up commits, separating the lines that fixed the bug from other changes.

A problem that commonly arises is when a developer needs to improve an old feature or revisit a bug fix, they search the commit history, but with refactorings included in bug fix commits they often spend a large amount of time sifting through version control system diff statements to isolate the code that is significant. Separate commits consisting of only the necessary changes for a bug fix would improve a developer's workload.

### Installation:

1. Download the project with:  

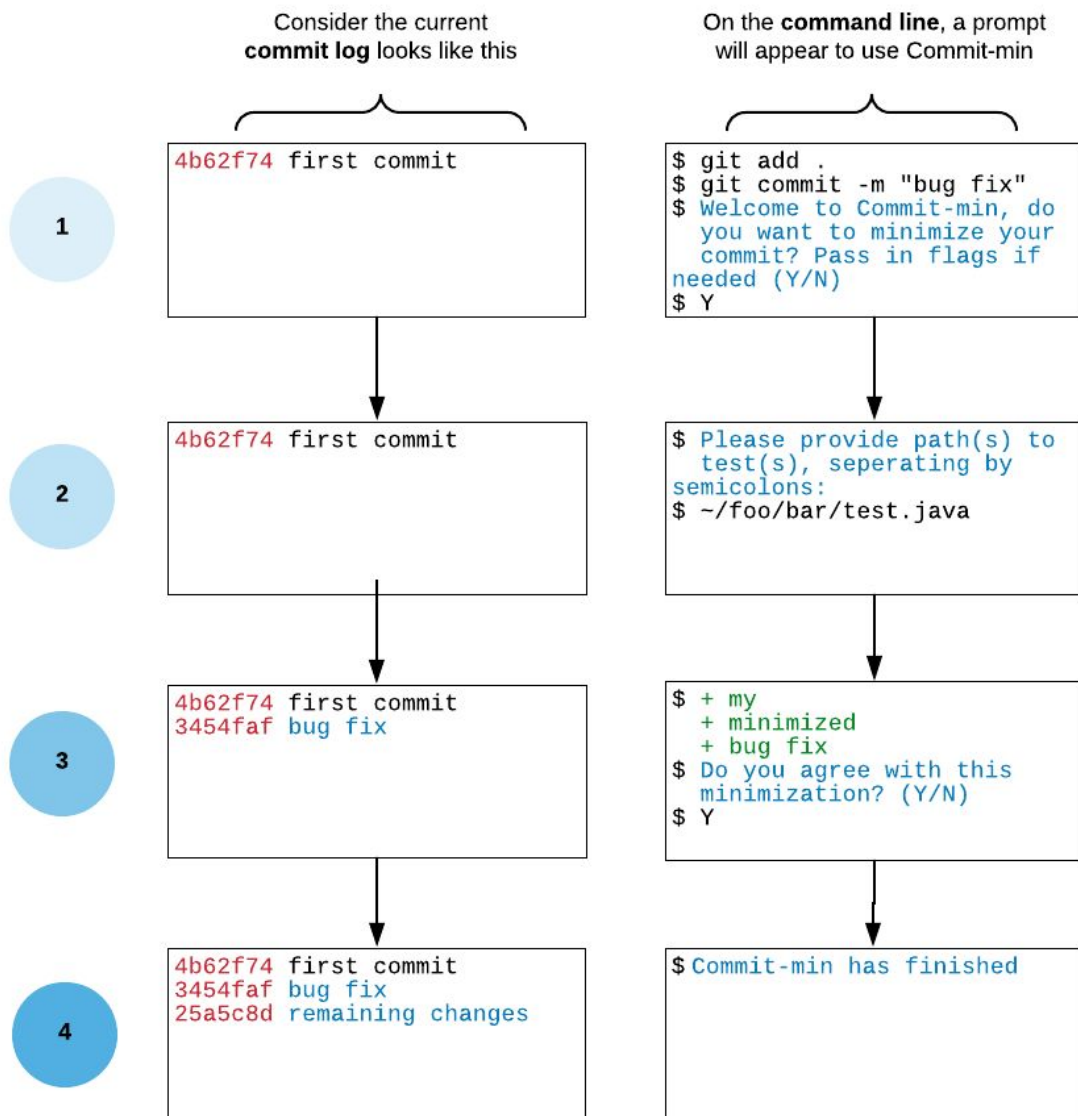
```
$ git clone https://github.com/Juliusc01/commit-min.git
```
2. Run commit-min-install with your repository directory to setup the git pre-commit hooks. This is only necessary once. Once this is done, our tool will ask to run each time the user tries to commit. NOTE: This will append to the end of the .git/hooks/pre-commit file in the repo provided.  

```
$ ./commit-min-install path/to/my/directory
```

### Usage Instructions:

1. On each invocation of 'git commit' you will be prompted to begin Commit-min. Press 'Y' to begin or 'N' to cancel. (Non case-sensitive) 'N' commits without running our program. They can also pass in flags (e.g: 'Y -a').
2. Commit-min will next ask you to provide paths to the tests that your bug fix is targeting. Commit-min will automatically minimize your bug fix and output the smallest number of lines that fixed the bug
3. Commit-min will ask the user if they agree with the minimization, and if the user types 'N', it will cancel the commit minimization and commit as it would normally. If ('Y'), then the minimized bug fix will be committed and any other changes not involved in the minimized bug fix will be committed in a separate commit message.

## Diagram Example:



## Optional flags:

These flags are passed in at the start of Commit-min

-a, --auto

Will run the tool automatically on your repo by checking the tests that failed in the previous commit and are passing in the newest commit. And will find the bug fix in your commit that caused the test to pass (Note will run all of your tests)

--time\_limit=LIMIT

Commit-min will only run for LIMIT seconds before prompting user to review changes.

NOTE: final commit not guaranteed to be minimum with this limit