

# **HW4: Simple File Transfer using TCP**

# What to Implement

- File transfer client and server application
  - The same to HW3 (displaying file transfer status)
- Difference from HW3
  - Use TCP instead of UDP (which will significantly reduce the implementation overhead)
- Mandatory
  - Client transfer rate control
- Optional
  - Option A: support of simultaneous file transfers from multiple clients
  - Option B: Support of simultaneous file transfers from a single client

```
client> sendrate 100K
ok
client> recvrate 200K
ok
client> ratecurr
send: 100K, recv: 200K
```

# What to Implement

- Support of 'credit' command: credit [student ID]
  - There should be four or five credit\_[ID]() function in client.c
  - Each team member should write his/her own credit function and commit the change to Github repository by herself or himself

```
client> credit 20101234
```

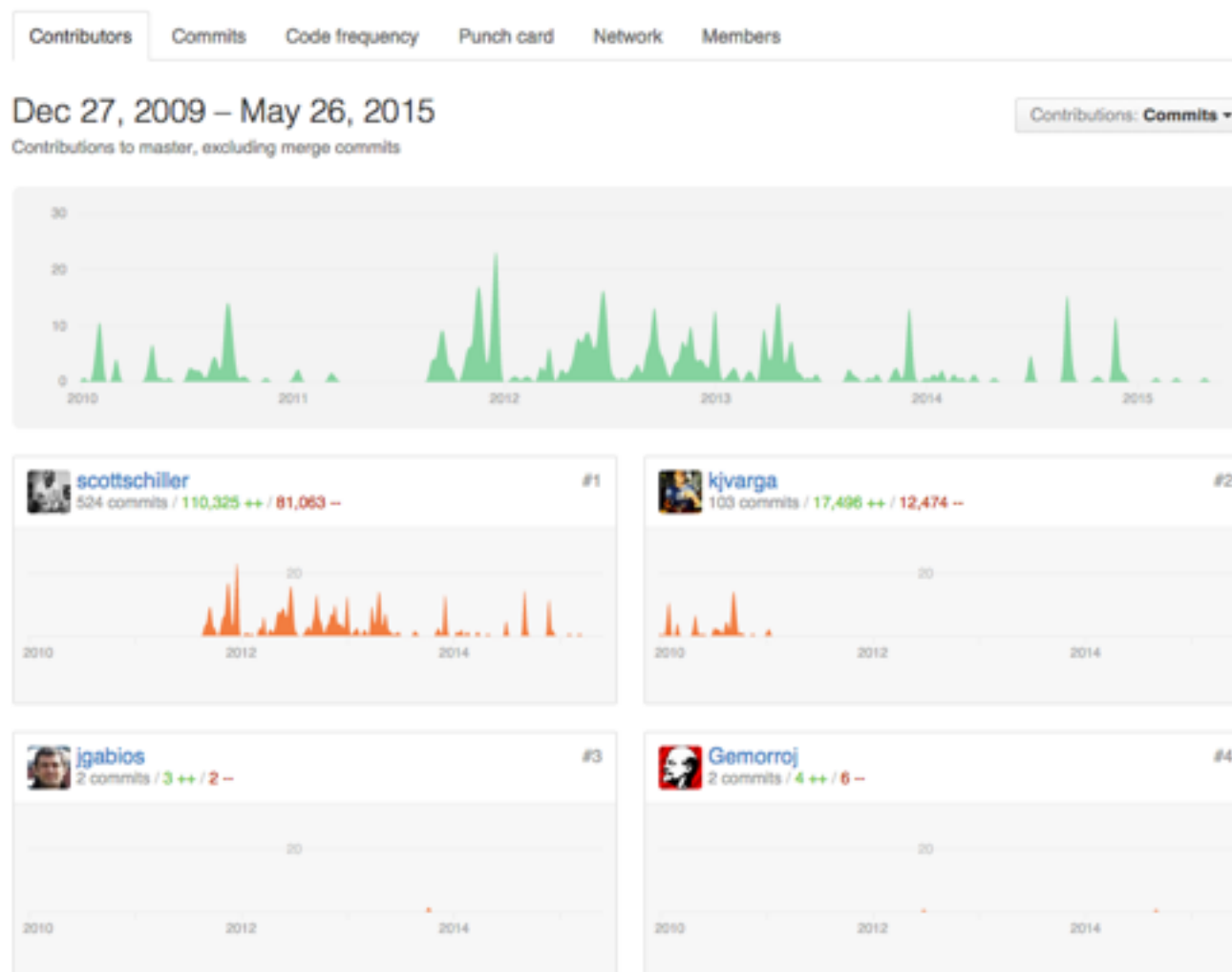
```
20101234 John Doe designed function X(), Y(), Z() and participated in  
writing the report.
```

# HW4: Simple File Transfer using TCP

- What to submit?
  - client.c and server.c (and additional header files if necessary)
  - A short report explaining
    - how to run your application (with captured images)
- Where to submit?
  - Github ([www.github.com](http://www.github.com))
- Due date
  - June 14, 2015, 23:59PM

# What to Implement

- Mandatory
  - Every student should participate in writing and commit code to Github
  - Not allowed
    - Member A: coding+discussion, Member B, C, and D: discussion



# Grading

- Single client + xfer rate
  - 10 pts
- Option A or B
  - Additional 5 pts for each
- Bonus points & policies
  - All the participating members will get the same points
  - Bonus points for each member having a commit history
    - +1pts for 4 person team + 1pts
    - +0.8pts for 5 person team + 1pts
  - Max score will be:  $10+5+5+4+1 = 25\text{pts}$
  - Non-participating members will get **no points**

# Team Assignment for HW3&4

name	Team	name	Team	name	Team
김정출	1	구본현	6	김소담	10
김희태	1	김영호	6	민권홍	10
민경민	1	신승열	6	인형민	10
이두나	1	최은주	6	최은헌	10
임기성	1	허성실	6	황주현	10
김다은	2	김민호	7	김연빈	11
박지현	2	박천호	7	김혜진	11
이성수	2	알소베히바데르나지엠	7	서동주	11
전한셈	2	유선	7	양희선	11
에기시브 오딜존	3	조광현	7	김지용	12
이소령	3	권영훈	8	성연진	12
이웅	3	김나윤	8	이경빈	12
정지만	3	김예주	8	차진원	12
허준영	3	박지윤	8	최대호	12
김상호	4	최희재	8	박형순	13
김진하	4	김지현	9	신은영	13
김학균	4	데나야로브 백하도르	9	원윤주	14
박세희	4	이지연	9	이창우	13
이형준	4	임병준	9	황인기	13
김용현	5	최현준	9	유세프	14
김윤창	5			이동우	14
이승희	5			이재열	14
임솔빈	5				
최진영	5			차동민	14

# TCP Tip

- <http://stackoverflow.com/questions/15384518/how-many-bytes-can-i-write-at-once-on-a-tcp-socket>
- How many bytes can I write at once on a TCP socket?

```
ssize_t total_bytes_written = 0;
while (total_bytes_written != 1024)
{
    assert(total_bytes_written < 1024);
    ssize_t bytes_written = write(tcp_socket,
                                  &buffer[total_bytes_written],
                                  1024 - total_bytes_written);
    if (bytes_written == -1)
    {
        /* Report failure and exit. */
        break;
    }
    total_bytes_written += bytes_written;
}
```

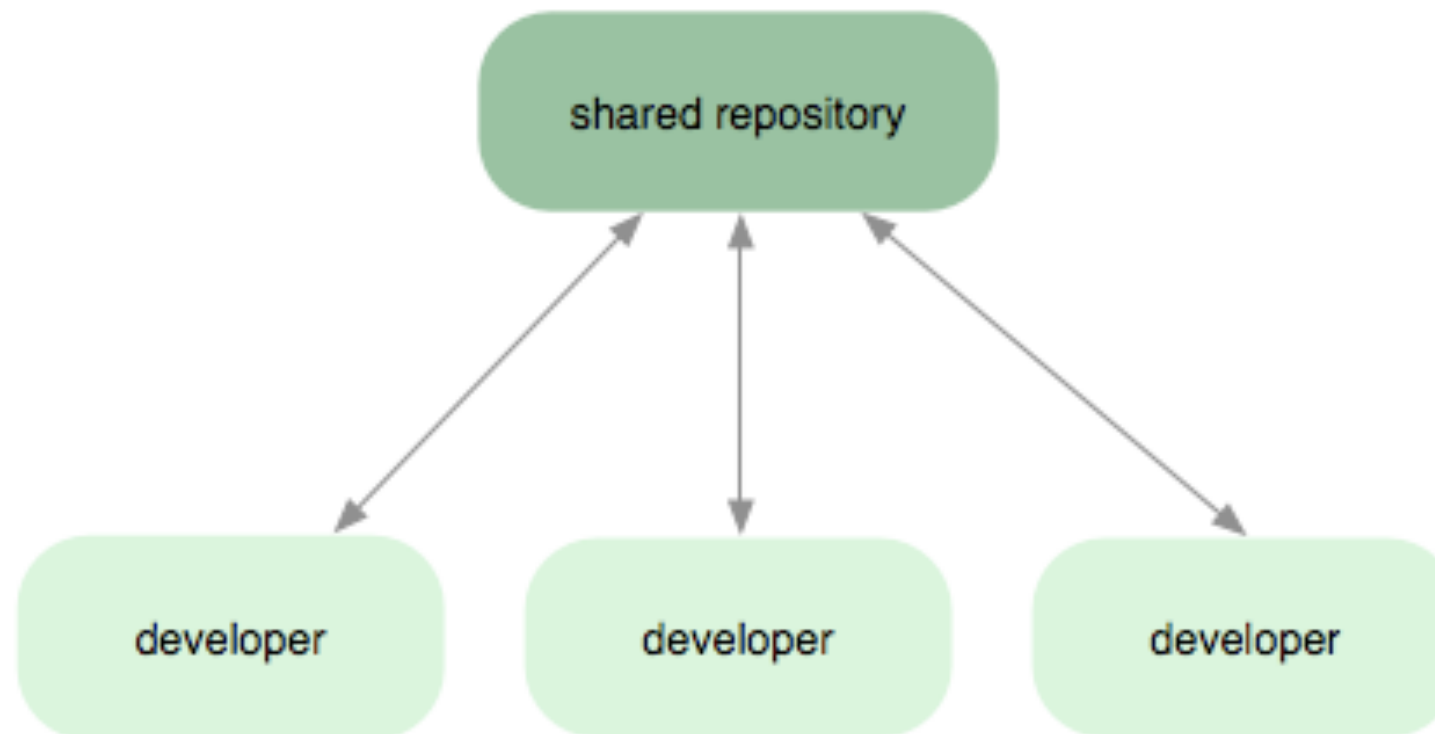


# Understanding Git

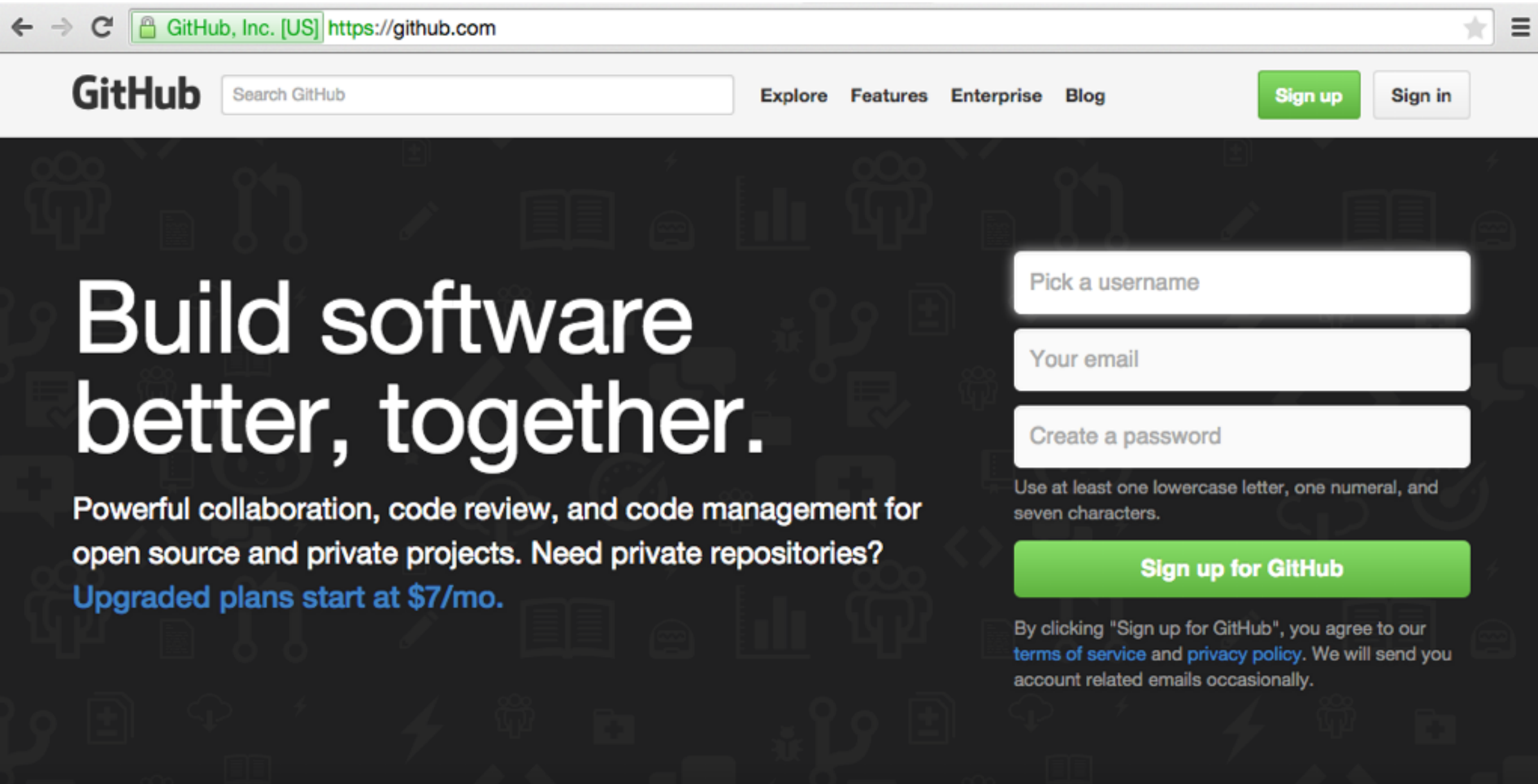
- <http://pismute.github.io/whygitisbetter/#easy-to-learn>

## Subversion-Style Workflow

매우 일반적으로 사용되는 Git 워크플로우로 중앙집중식 centralized workflow 이 있다. 특히 Subversion같은 중앙집중식 시스템을 사용하다가 넘어온 사람들이 사용한다. 내가 마지막으로 fetch한 후에 아무도 푸시하지 못하도록 할 수 있다. 그래서 모든 개발자가 같은 서버에 푸시하는 중앙집중식 모델도 가능하다.



# Github Signup



The image is a screenshot of the GitHub website's sign-up page. At the top, there's a browser address bar showing 'GitHub, Inc. [US] https://github.com'. Below this is the GitHub logo and a search bar. Navigation links for 'Explore', 'Features', 'Enterprise', and 'Blog' are present, along with 'Sign up' and 'Sign in' buttons. The main content area has a dark background with a pattern of small icons. On the left, a large white text reads 'Build software better, together.' followed by a description of GitHub's services and pricing. On the right, there's a sign-up form with three input fields: 'Pick a username', 'Your email', and 'Create a password'. Below these fields is a green 'Sign up for GitHub' button. At the bottom right, there's a disclaimer about agreeing to terms of service and privacy policy.

GitHub, Inc. [US] https://github.com

GitHub Search GitHub Explore Features Enterprise Blog Sign up Sign in

## Build software better, together.

Powerful collaboration, code review, and code management for open source and private projects. Need private repositories? Upgraded plans start at \$7/mo.

Pick a username

Your email

Create a password

Use at least one lowercase letter, one numeral, and seven characters.

**Sign up for GitHub**

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We will send you account related emails occasionally.

# Git Client Installation

- <https://git-scm.com/download/linux>

## Download for Linux and Unix

It is easiest to install Git on Linux using the preferred package manager of your Linux distribution.

### Debian/Ubuntu

```
$ apt-get install git
```

# Git Client Configuration

- <http://classic.scottr.org/presentations/git-in-5-minutes/>
- <http://guides.railsgirls.com/github/>
- <https://www.youtube.com/watch?v=mMsWq3rS6Po>

# Adding Collaborators on Github

- <https://help.github.com/articles/adding-collaborators-to-a-personal-repository/>