

CICD: 3 weeks +

Version control : git

Build tool : maven

CI tool : jenkins

Artifactory : jfrog

Code analysis : sonarqube

- 1) Create AWS Account
- 2) Create AWS server and use
- 3) What is SDLC?
- 4) SDLC phases or stages?
- 5) SDCL model ?
- 6) Waterfall model?
- 7) Agile model?
- 8) Sprint ?
- 9) Release?
- 10) Sprint life Cycle?
- 11) Scrum ?

Instance = sever

Ssh = linux to linux

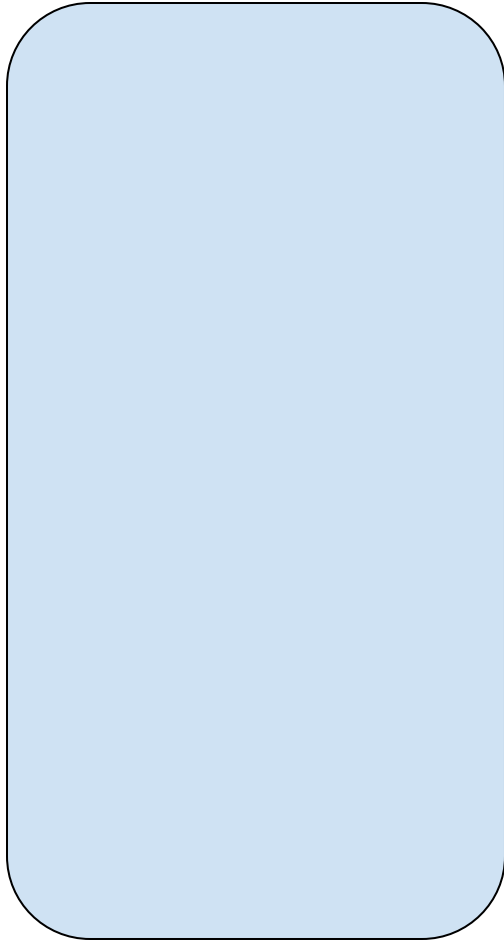
Http = we can access our webserver from internet

Create AWS server

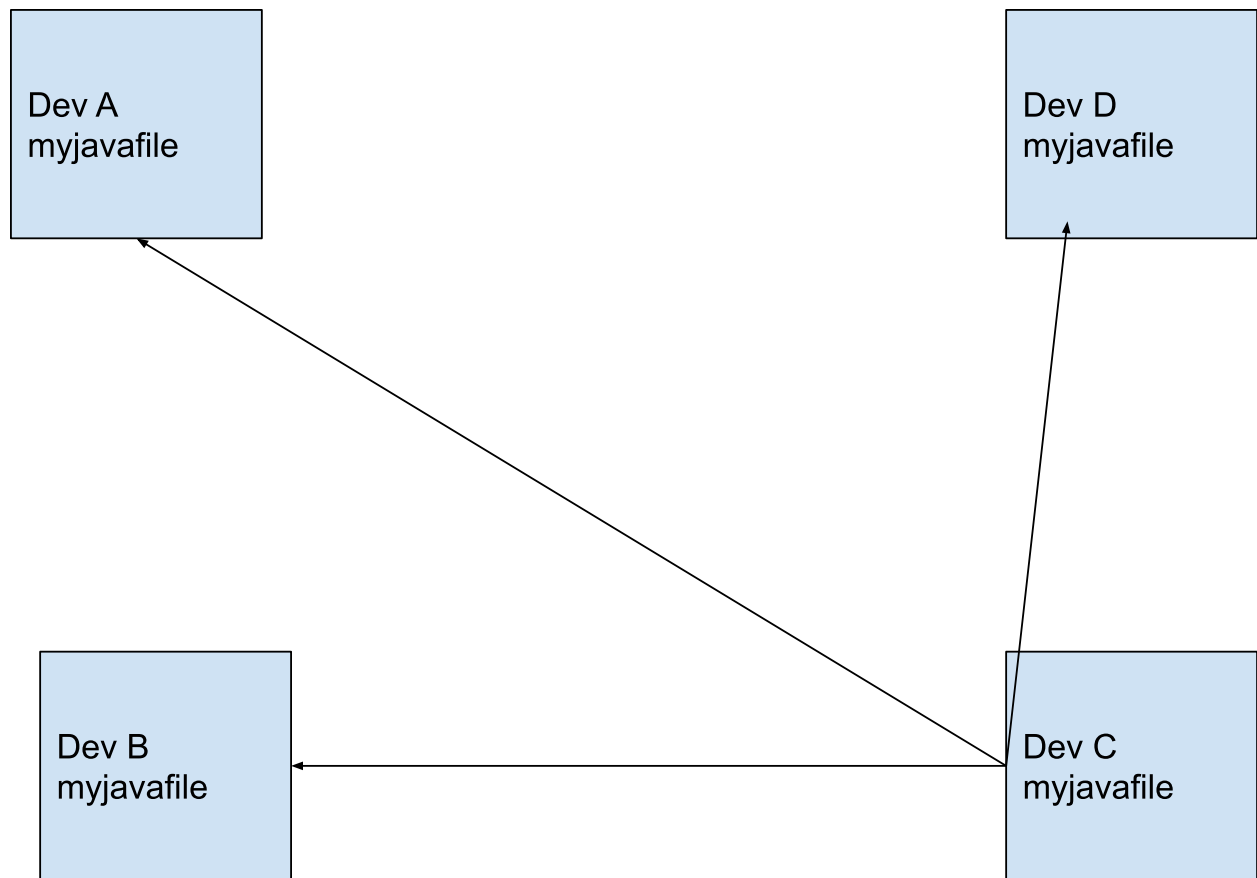
- 1) Create AWS account
- 2) Choose Ec2 service
- 3) Choose launch instance
- 4) Choose instance count and name as webserver
- 5) Choose AMI as ubuntu 18 linux instance
- 6) Choose instance type as t2.micro , it has 1 core CPU , 1 Gb ram
- 7) Create pem file ,it is just link passwd
- 8) Choose default network and Security group with SSH and http
- 9) Choose default EBS volume, it just like Hard disk
- 10) Choose launch instance , we can create server in aws

How to getting into aws server

- 1) Install git on your laptop**
- 2) After logging into server**
 - \$ sudo -i**
 - # apt-get update**
 - # apt-get install apache2**



Without Version control:

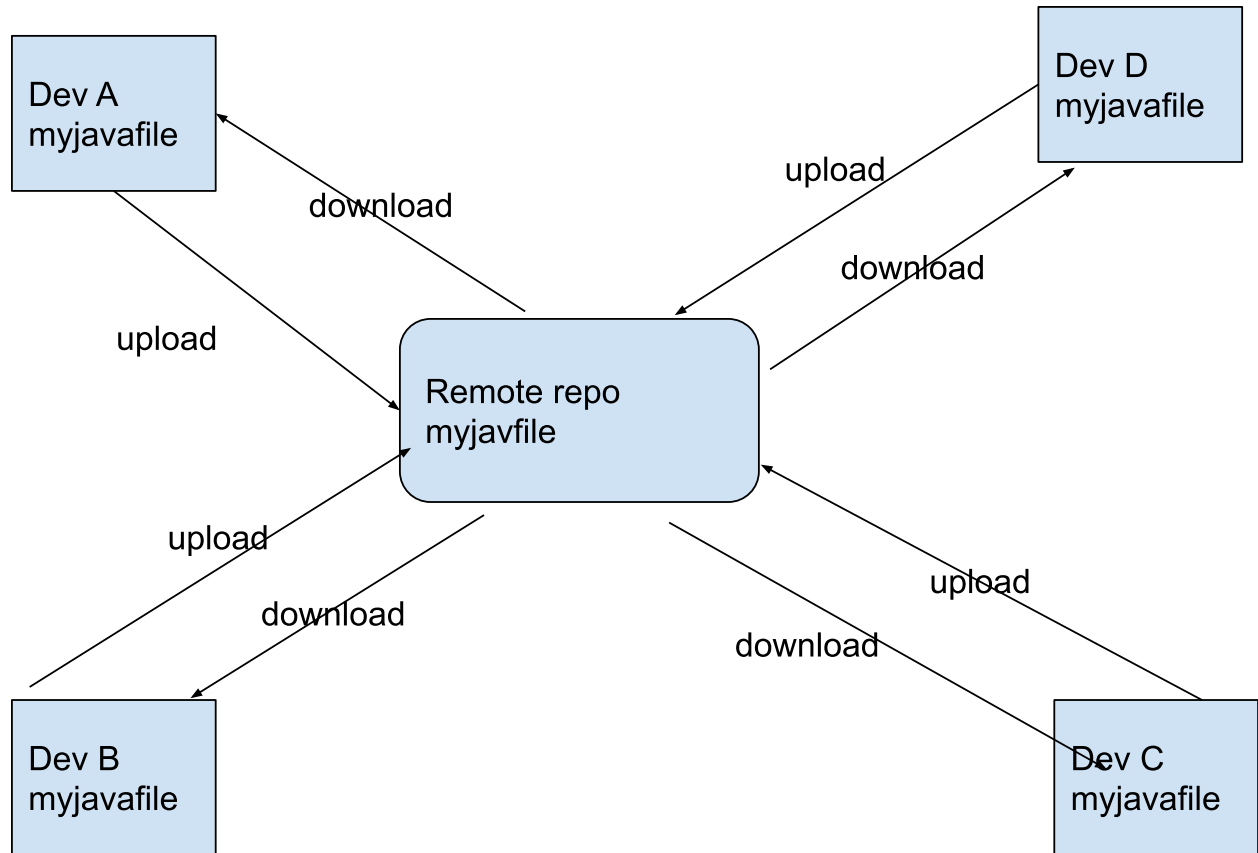


In 100% of time

80% of time going merge code

20% of time for writing code

With Version control:



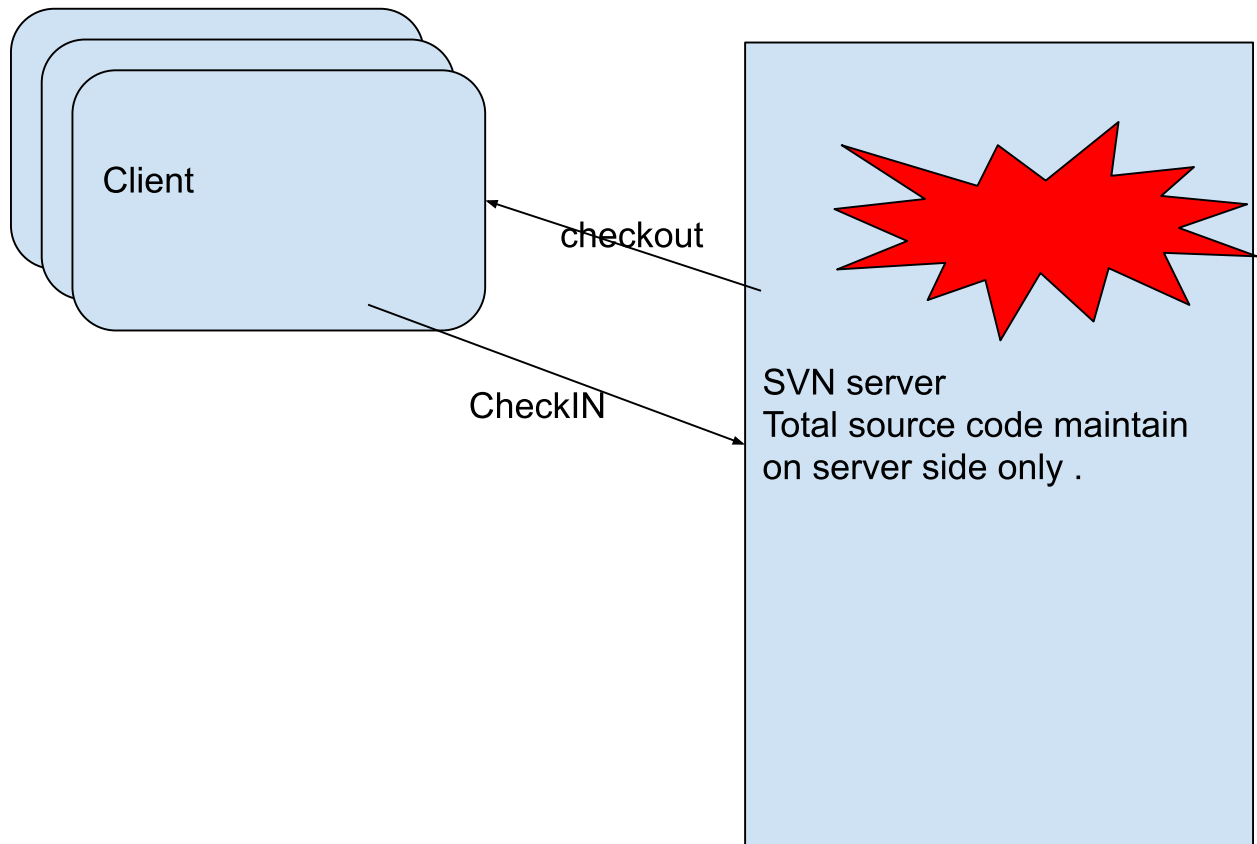
In 100% of work
80% of time for write code
20% of time for merge

- 1) We can create versions for every change with timestamp
- 2) We can be able to go back older versions at any point of time
- 3) We can do auto merge(if it is possible)
- 4) We are able to create branches.

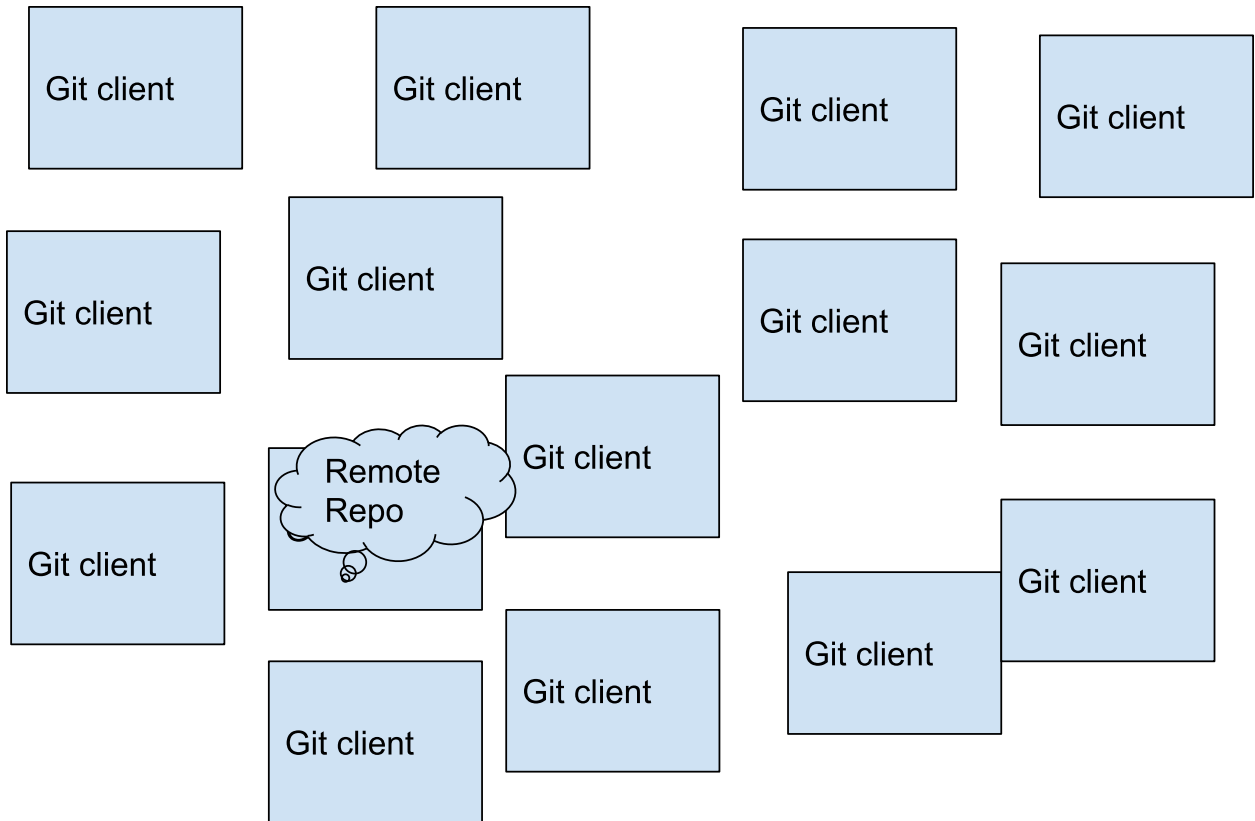
Version control:

- 1) Client and server ART
- 2) Distributed ART

Client and server ART:



Distributed ART: GIT

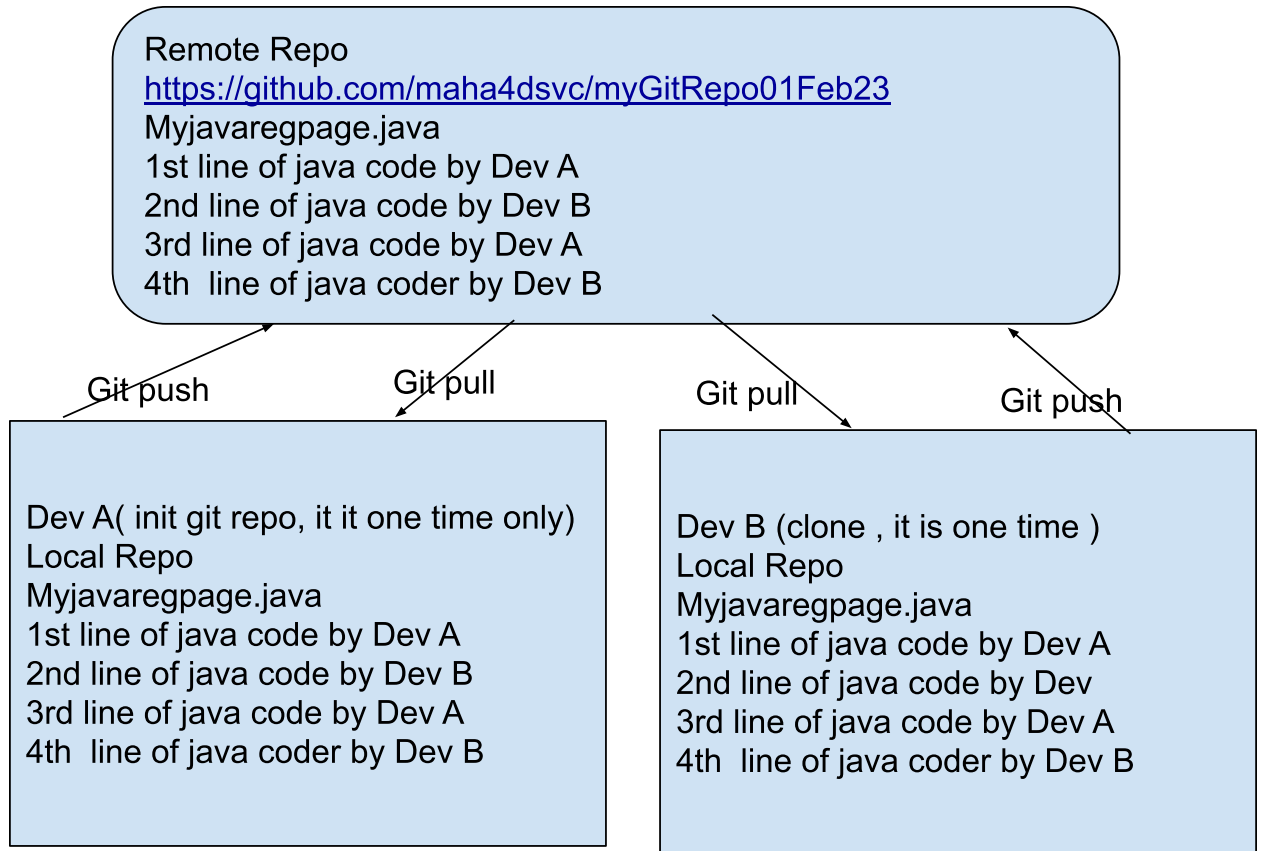


- 1) Create github account
- 2) Create a token , it is just like password
- 3) Create remote repo
- 4) Install git on your laptop
- 5) Create folder on your laptop and open git bash on same folder and execute below commands (it it one time only)

```
echo "# mygitRepo31Jan23" >> README.md
git init
git add README.md
git commit -m "first commit"
git config --global user.name "maha"
git config --global user.email "maha@gmail.com"
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/maha4dsvc/mygitRepo31Jan23.git
git push -u origin main ( it will ask token)
```

- 6) For action or change

```
git add -A
git commit -m "comment"
git push
```

Uploading:

Working directory : any change (create/delete/update/edit)

Staging area : git add -A

Local repo : git commit -m "comment"

Remote Repo : git push

Download:

Git pull= fetch + merge