|  |  |
| --- | --- |
|  | **TÀI LIỆU** |
| **Vận hành hệ thống devops** | |

# **CÀI ĐẶT VẬN HÀNH CÁC THÀNH PHẦN CỦA HỆ THỐNG**

1. **Cài đặt Oracle JDK 8**

* Oracle JDK 8 version 221 có thể download tại link: <https://www.oracle.com/java/technologies/javase/javase8u211-later-archive-downloads.html>  
  Lưu ý: Download bản JDK 8 cho hệ điều hành tương ứng , là bản đóng gói không cần cài đặt
* Tạo folder có đường dẫn /app để lưu file đã tải về
* Chạy các command sau:

|  |
| --- |
| # cd /app  # sudo su  # mkdir /opt/jdk  # tar -zxf jdk-8u221-linux-x64.tar.gz-C /opt/jdk  # ls /opt/jdk  # update-alternatives --install /usr/bin/java java /opt/jdk/jdk1.8.0\_221/bin/java 100  # update-alternatives --install /usr/bin/javac javac /opt/jdk/jdk1.8.0\_221/bin/javac 100  # update-alternatives --display java  # update-alternatives --display javac  # java –version and javac –version |

1. **Cài Docker lên Server**
   * 1. Gỡ bỏ cài đặt cũ bằng lệnh : sudo apt-get remove docker docker-engine docker.io containerd runc
     2. Cài đặt repository:

sudo apt-get install \

apt-transport-https \

ca-certificates \

curl \

gnupg-agent \

software-properties-common

* curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add –
* sudo add-apt-repository \

"deb [arch=amd64] https://download.docker.com/linux/ubuntu \

$(lsb\_release -cs) \

stable"

* sudo apt-get update
* sudo apt-get install docker-ce docker-ce-cli containerd.io
* Mặc định chỉ user root mới chạy được lệnh docker, để user non root chạy, cần phải có lệnh sau:
* $ sudo groupadd docker
* $ sudo usermod -aG docker $USER
* $ newgrp docker
* Mặc định, docker sẽ download images, cài container và resource ở thư mục /var, nếu như thư mục cài đặt OS không đủ dung lượng, thực hiện chuyển phân vùng lưu dữ liệu của docker sang phân vùng khác, ví dụ : /nbd
  + sudo service docker stop
  + Tạo file có tên : daemon.json ở trong folder: /etc/docker có nội dung sau:  
    
  + Copy dữ liệu sang folder mới:  
    sudo rsync -aP /var/lib/docker/ /nbd/docker\_data/docker
  + Xoá bỏ thư mục cũ :  
    rm -rf /var/lib/docker
  + Khởi động lại docker service:  
    sudo service docker start

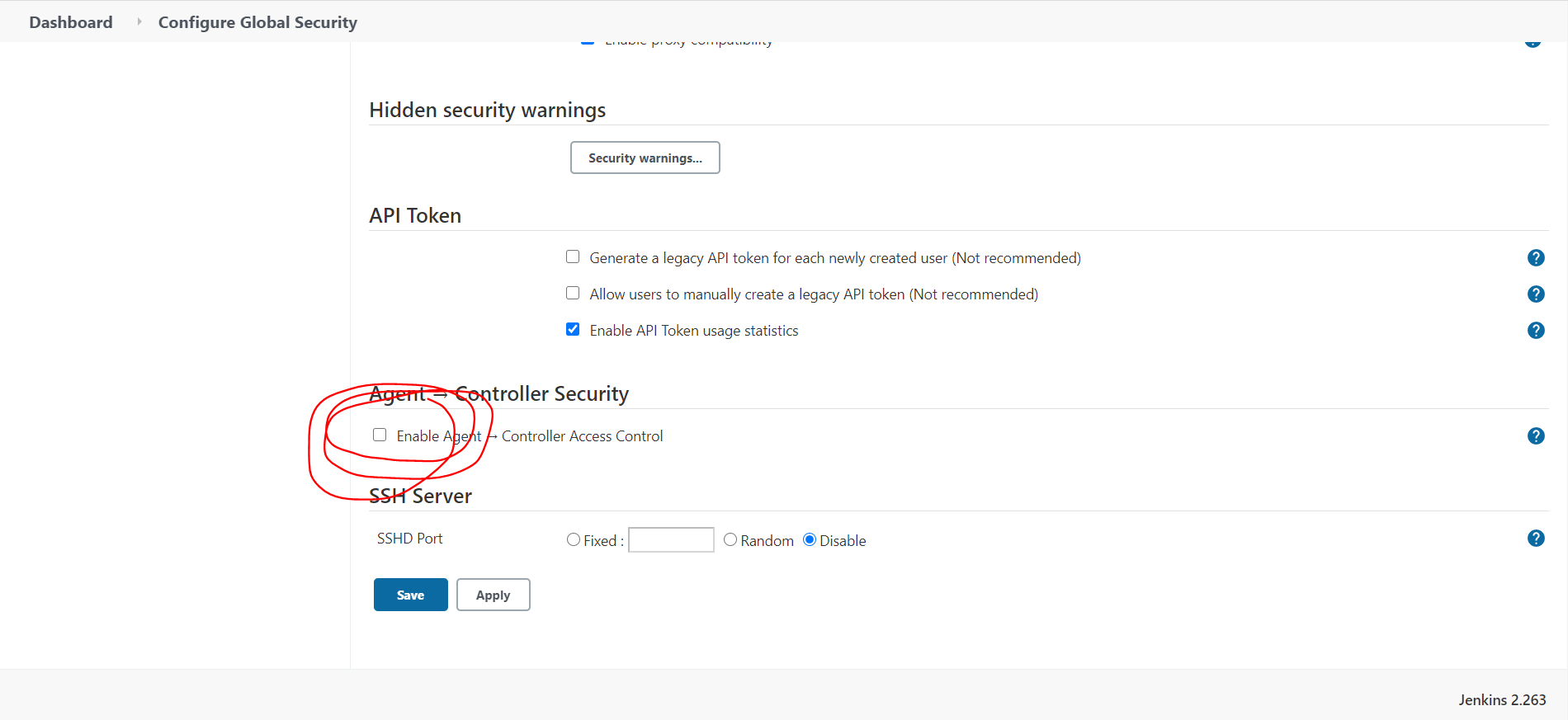
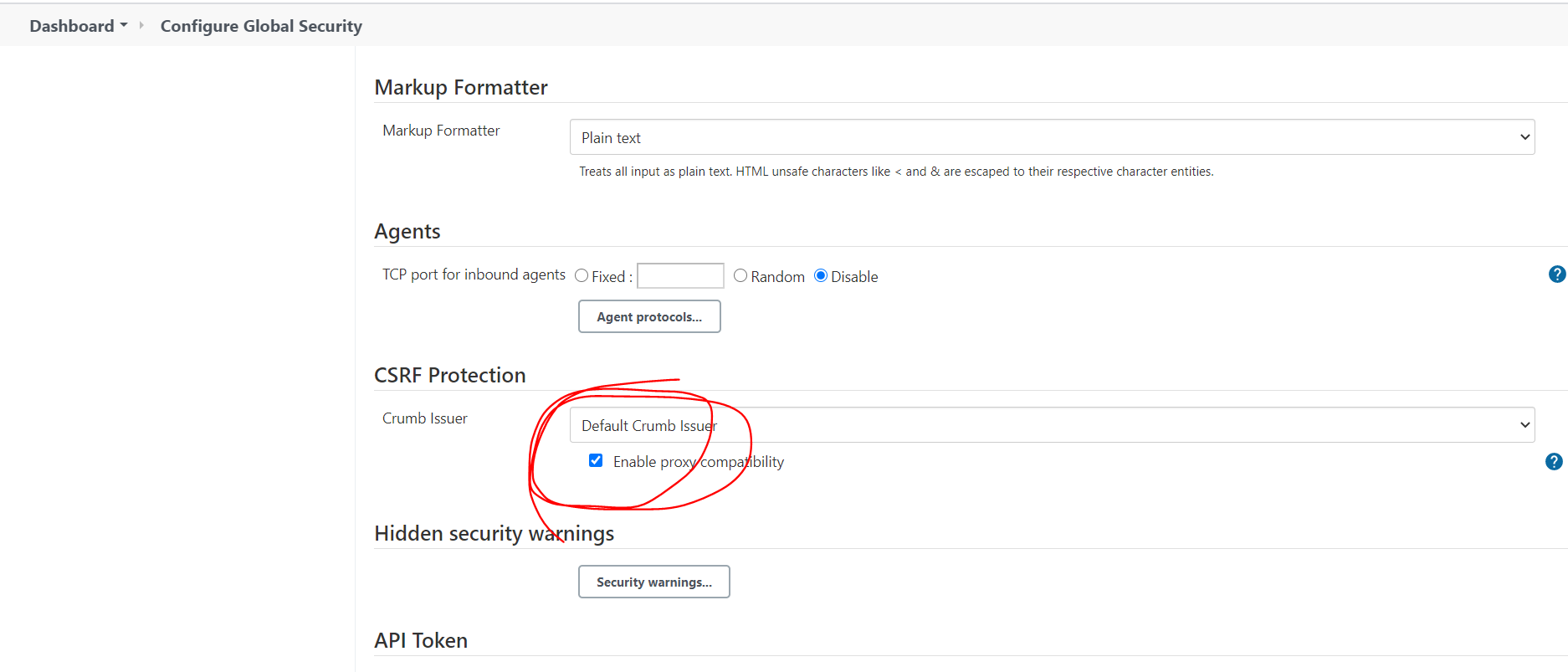
1. **Cài đặt Docker compose lên Server**
   1. sudo curl -L "https://github.com/docker/compose/releases/download/1.27.4/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
   2. sudo chmod +x /usr/local/bin/docker-compose
2. **Cài đặt Jenkins lên Server**

* wget -q -O - https://pkg.jenkins.io/debian-stable/jenkins.io.key | sudo apt-key add –
* sudo sh -c 'echo deb https://pkg.jenkins.io/debian-stable binary/ > \

/etc/apt/sources.list.d/jenkins.list'

* sudo apt-get update
* sudo apt-get install jenkins
* sudo systemctl start jenkins
* sudo systemctl status jenkins

Đổi port của jenkins từ 8080 sang 9999

* + vim /etc/default/jenkins
  + bam phim : Insert
  + change $PORT=8080 thanh port muon doi : $PORT=9999
  + ESC => :w => :q => Enter
  + sudo systemctl restart jenkins
* Mở trình duyệt : http://${IP\_SERVER}:9999 => thực hiện setting jenkins
* vim /var/lib/jenkins/secrets/initialAdminPassword
* Lấy mật khẩu được hiển thị bởi lệnh vim điền vào form trên trình duyệt
* Truy cập vào Jenkins dashboard, chọn : Manage Jenkins > Configure Global Security. Bỏ chọn phần Enable Agent và chọn phần Enable proxy compability  
    
    
  Chi tiết cài đặt : <https://www.jenkins.io/doc/book/installing/linux/>

1. **Cài đặt git lên Server**

* sudo apt install git
* git config --system http.sslVerify false
* git config –global http.sslVerify false

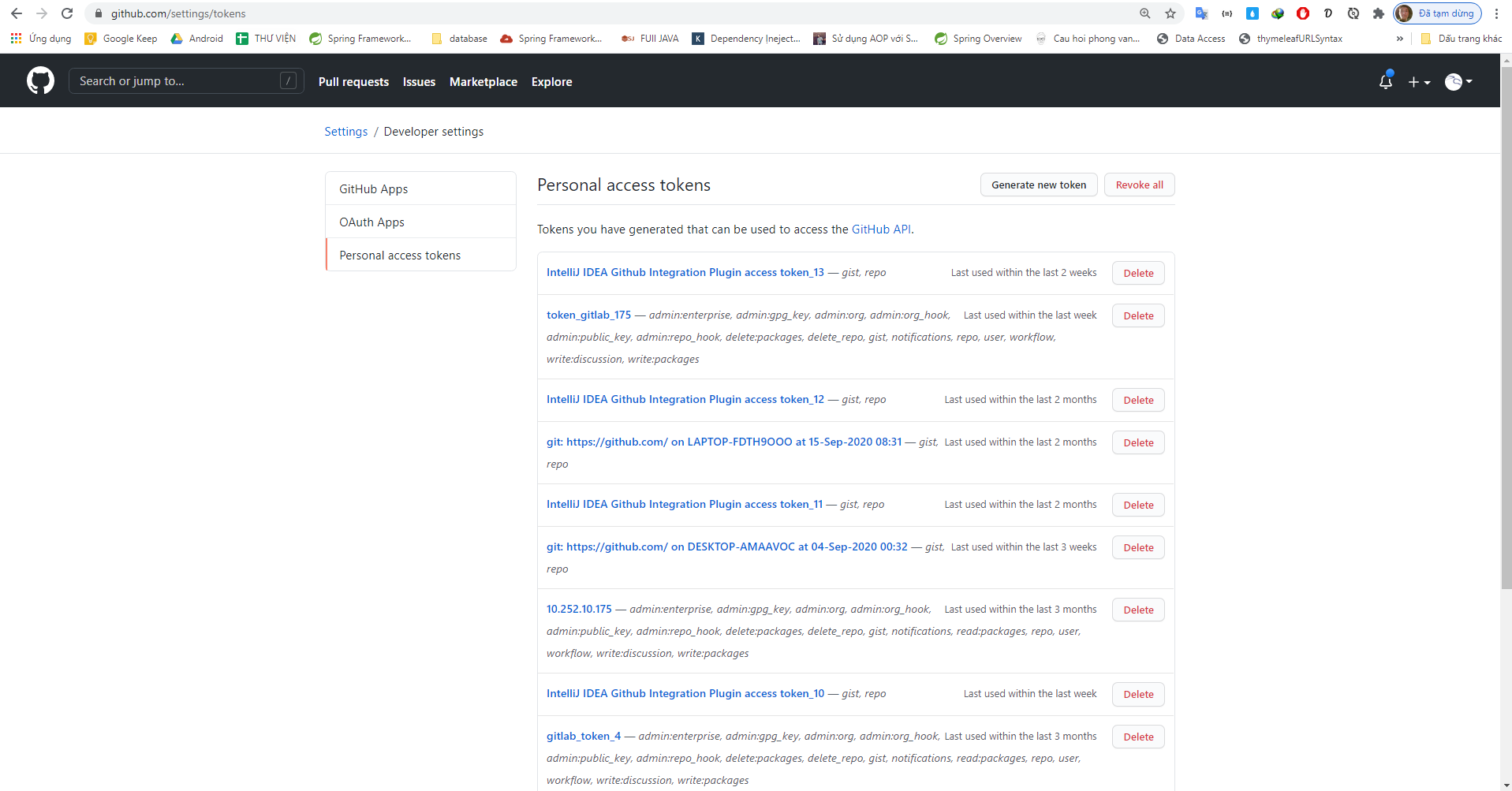
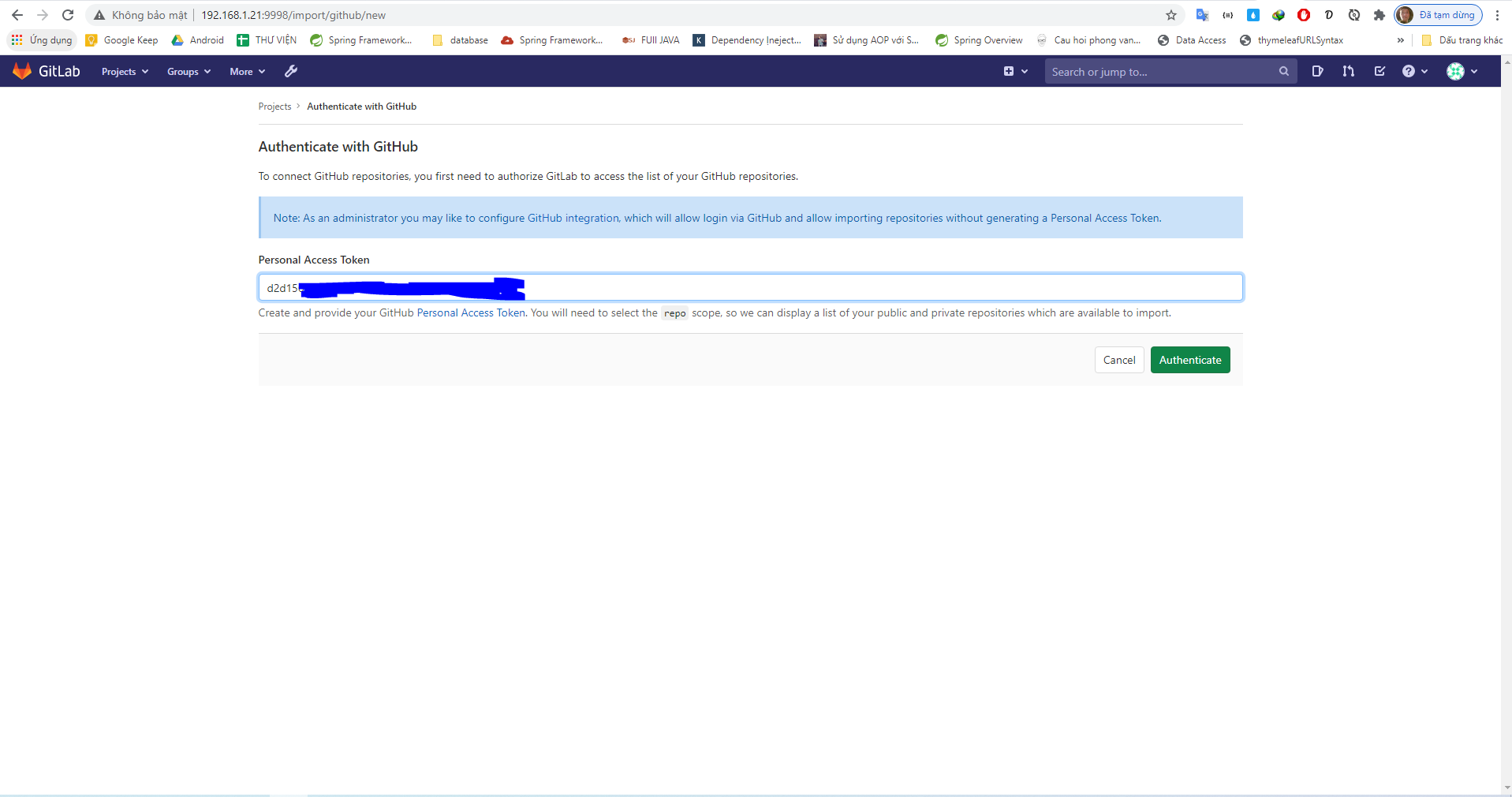
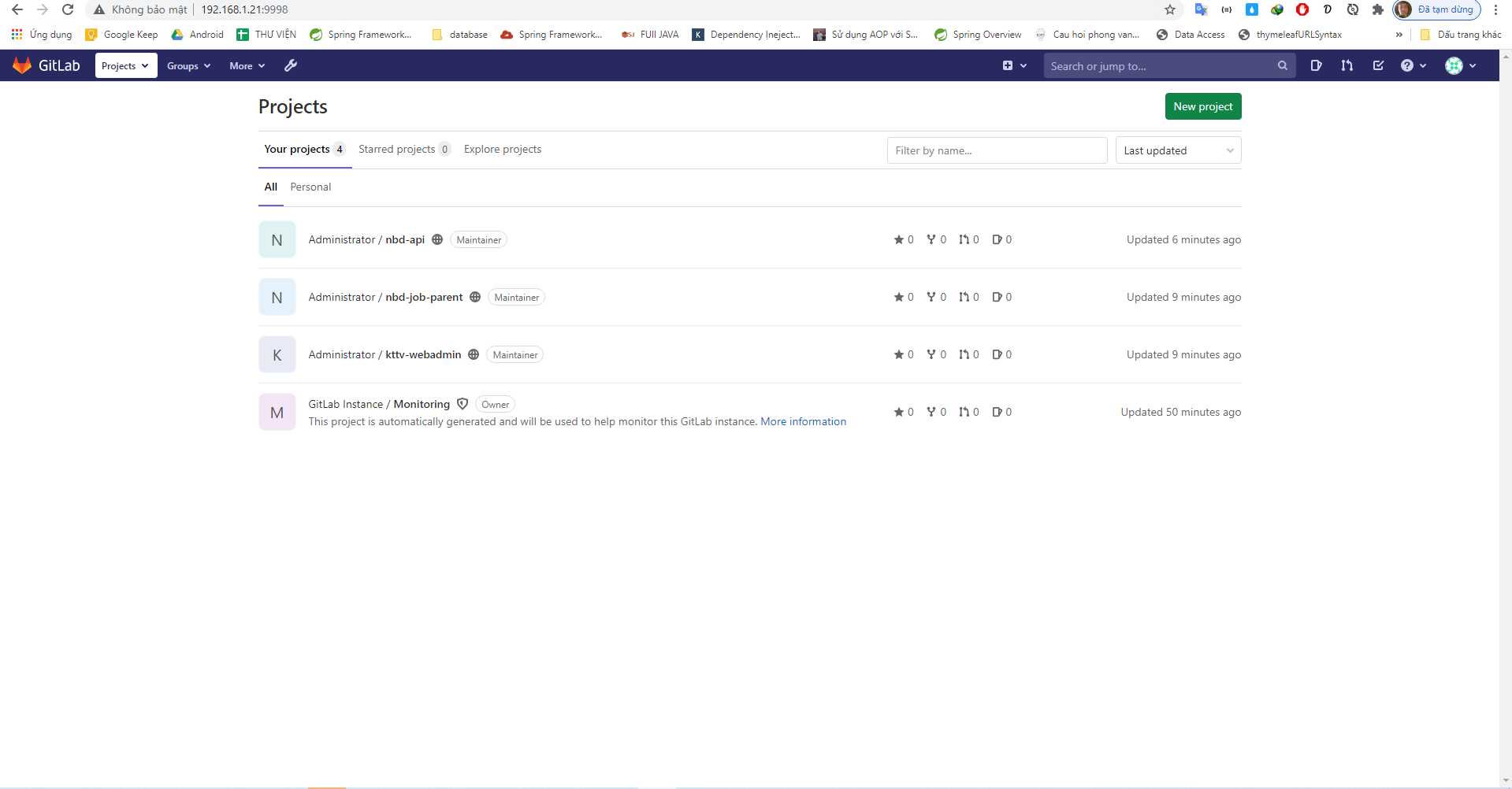
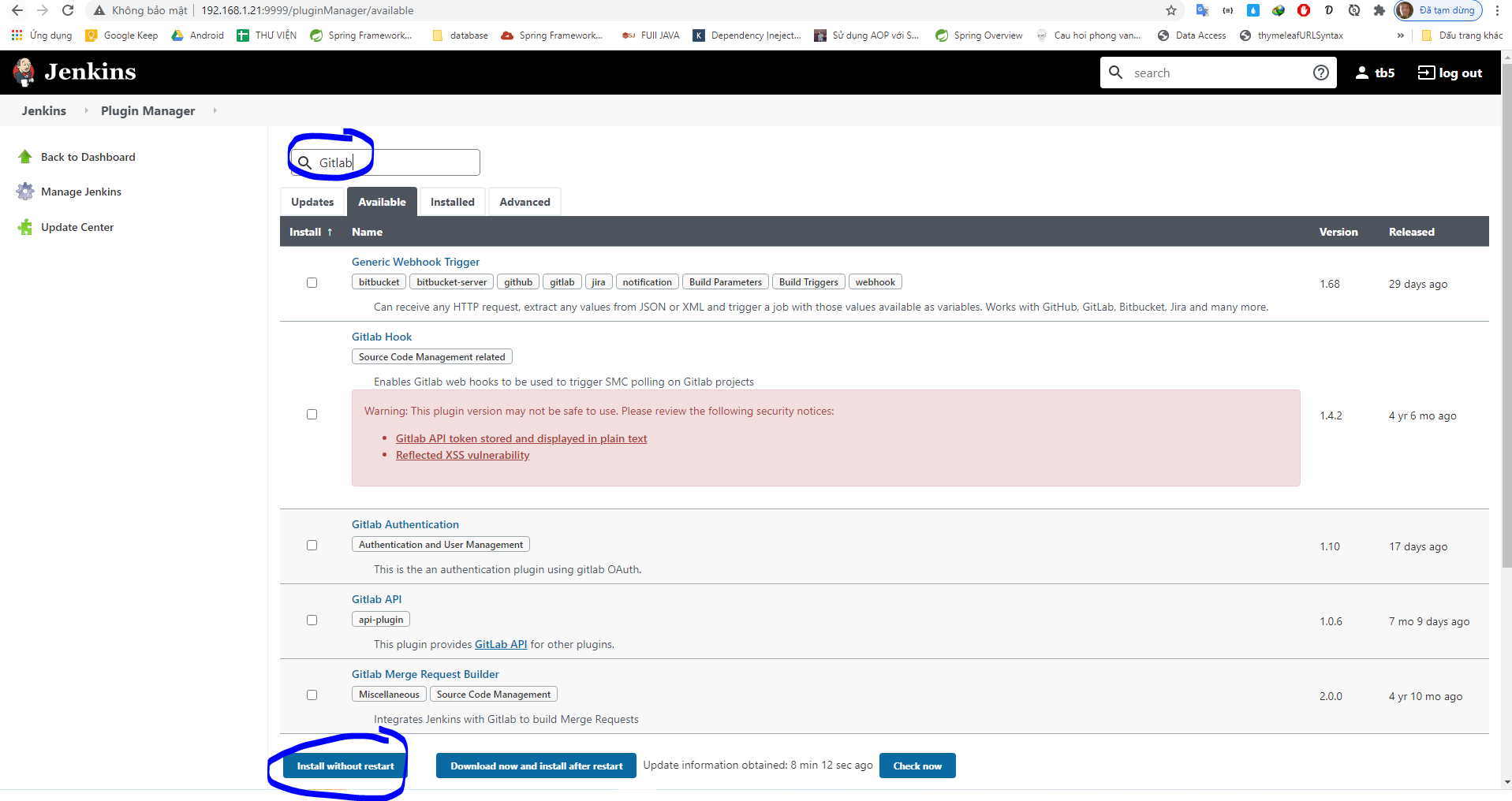
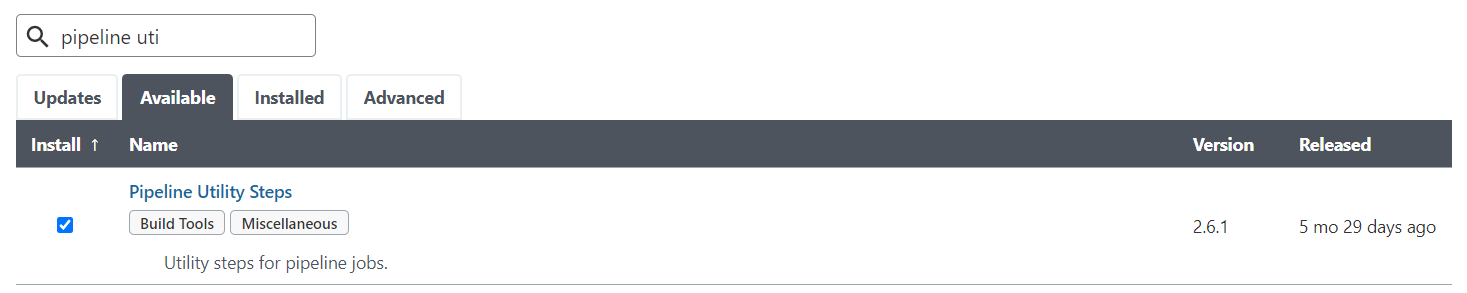
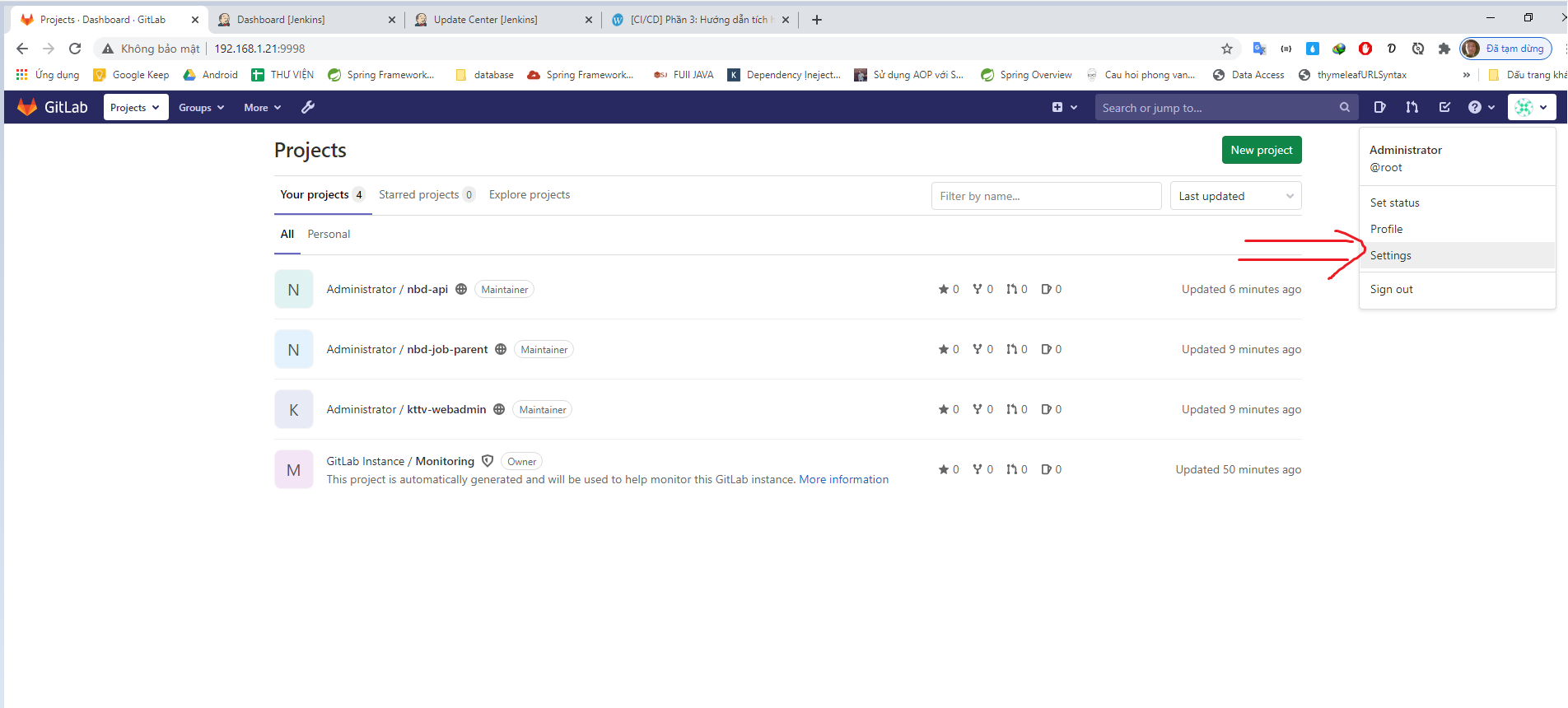
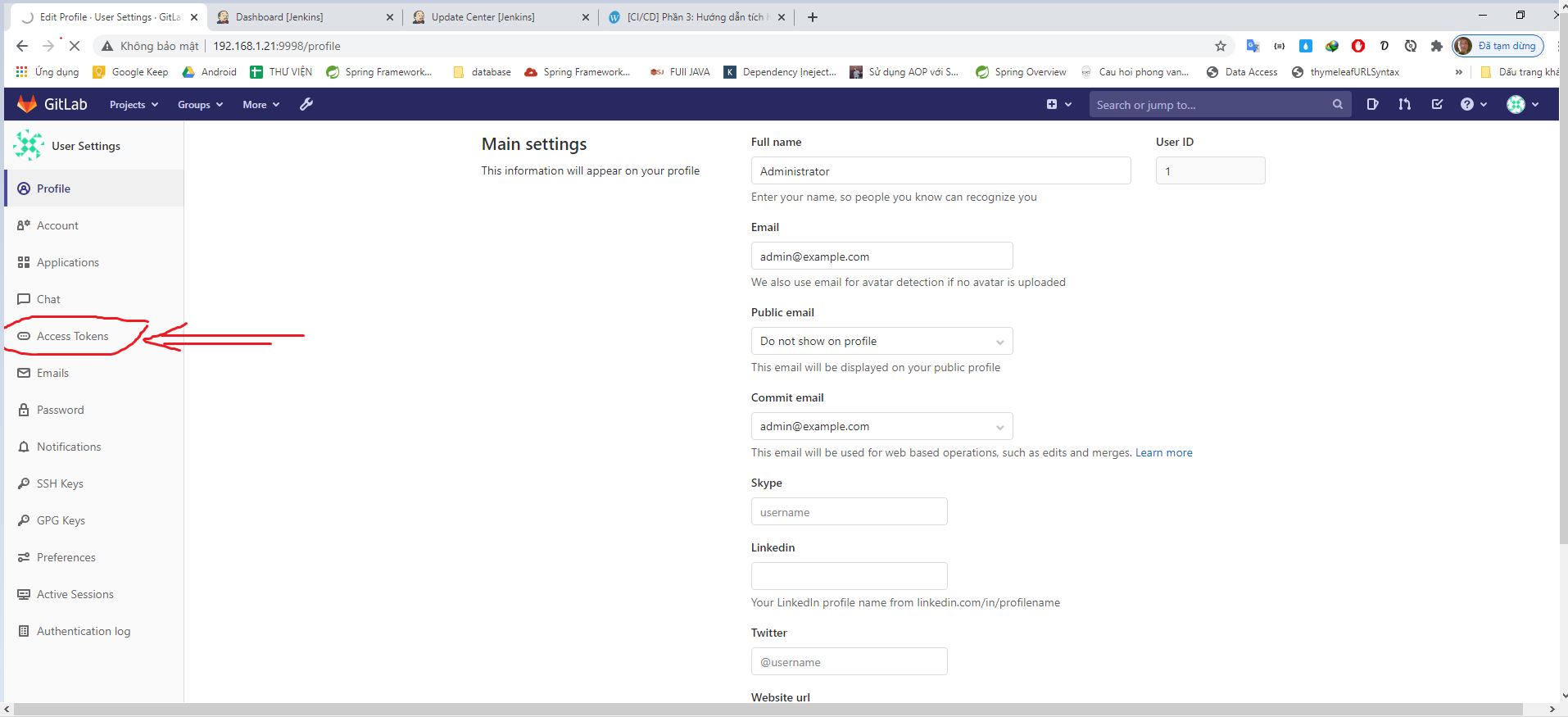
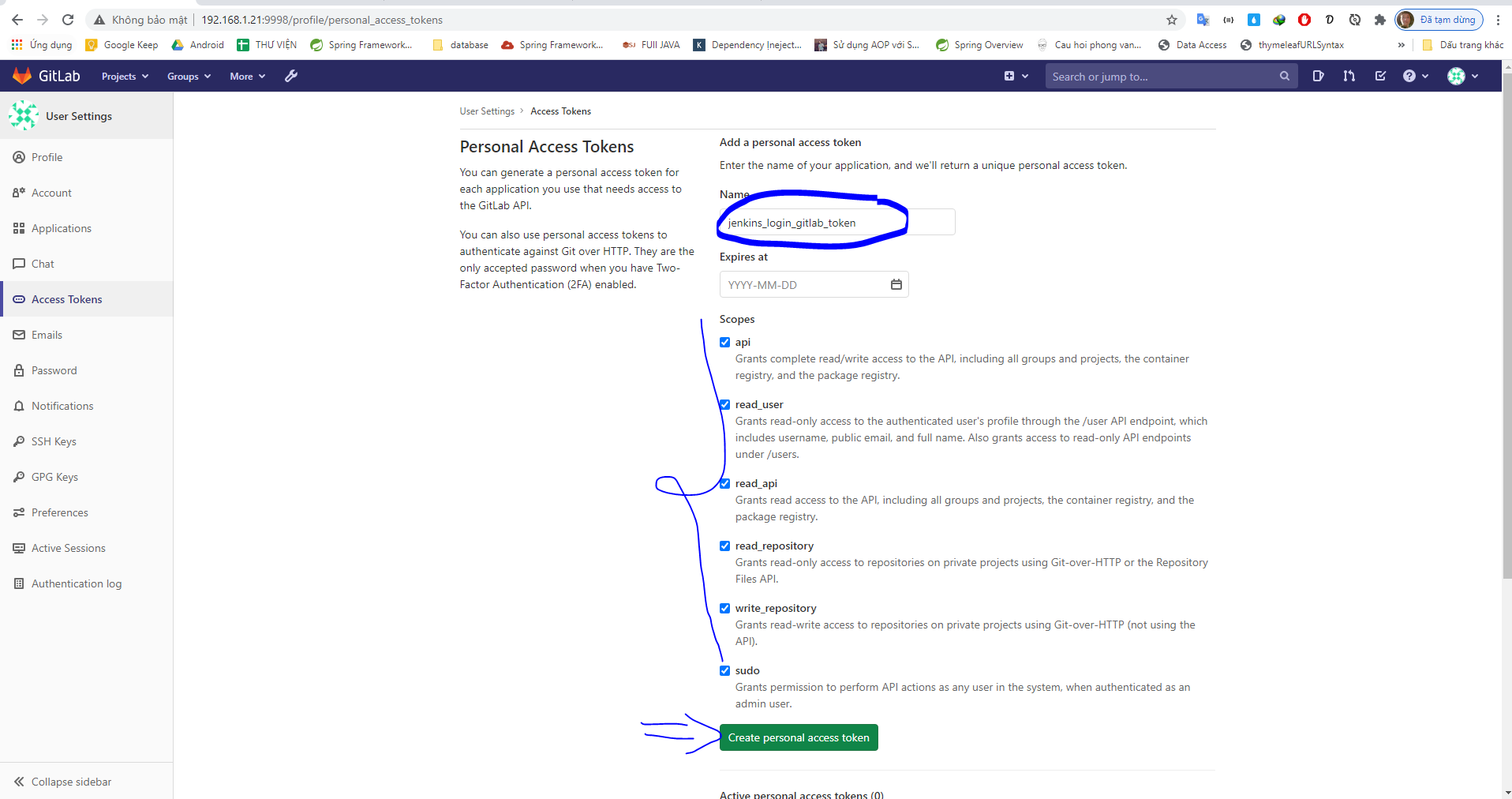
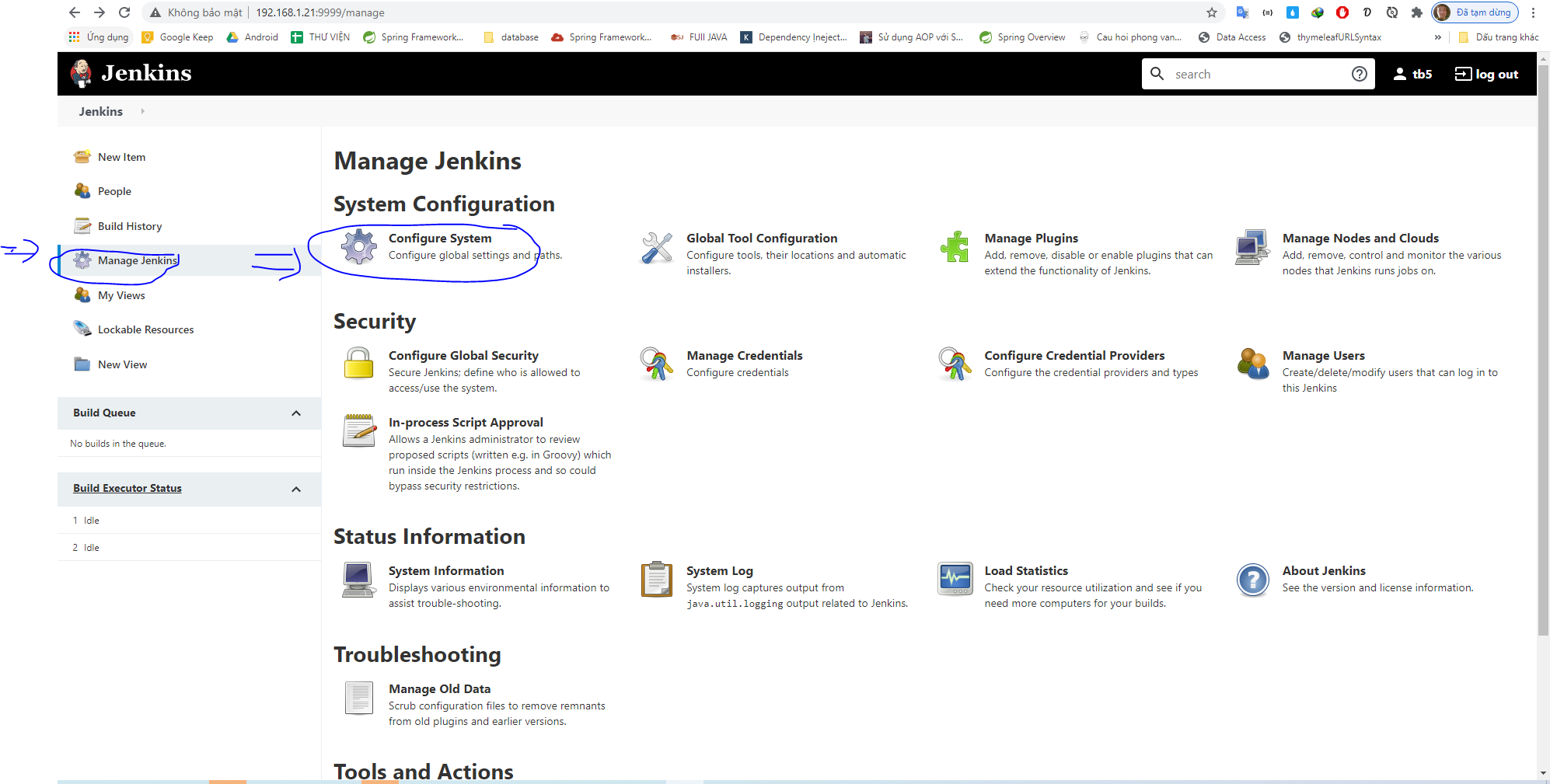
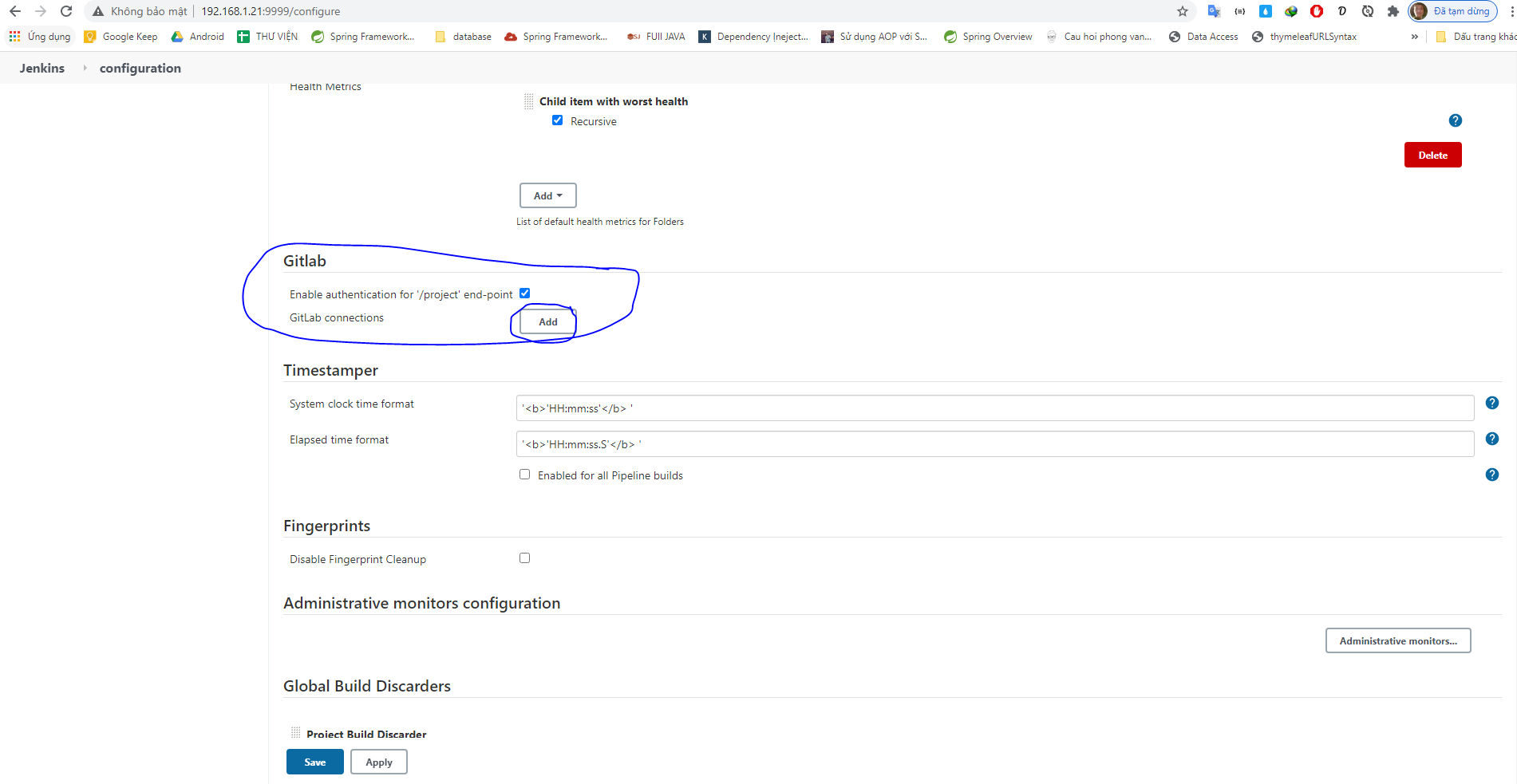
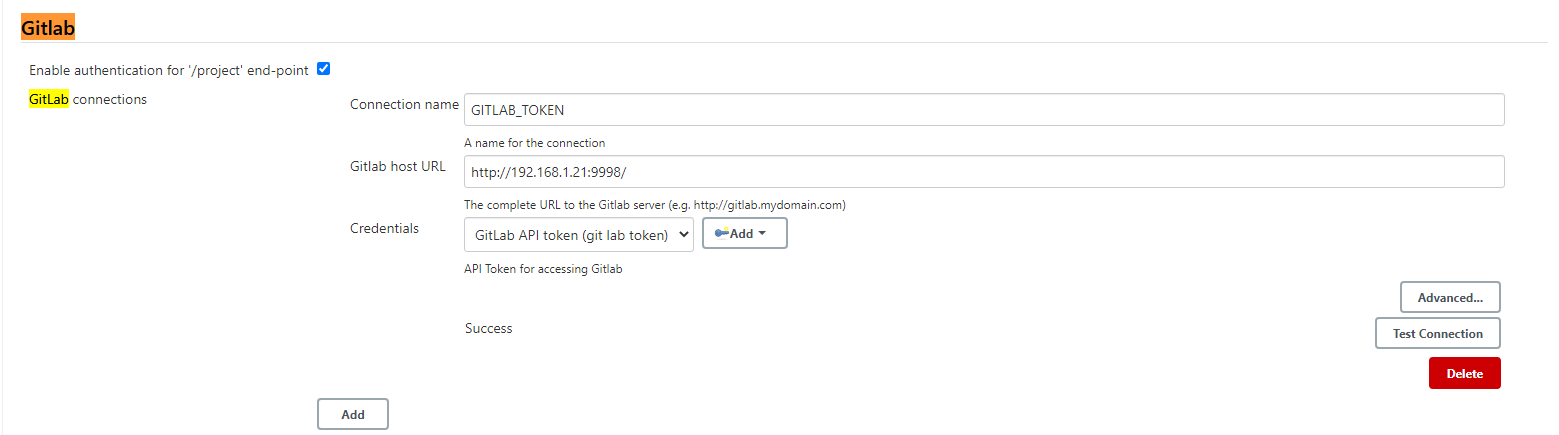
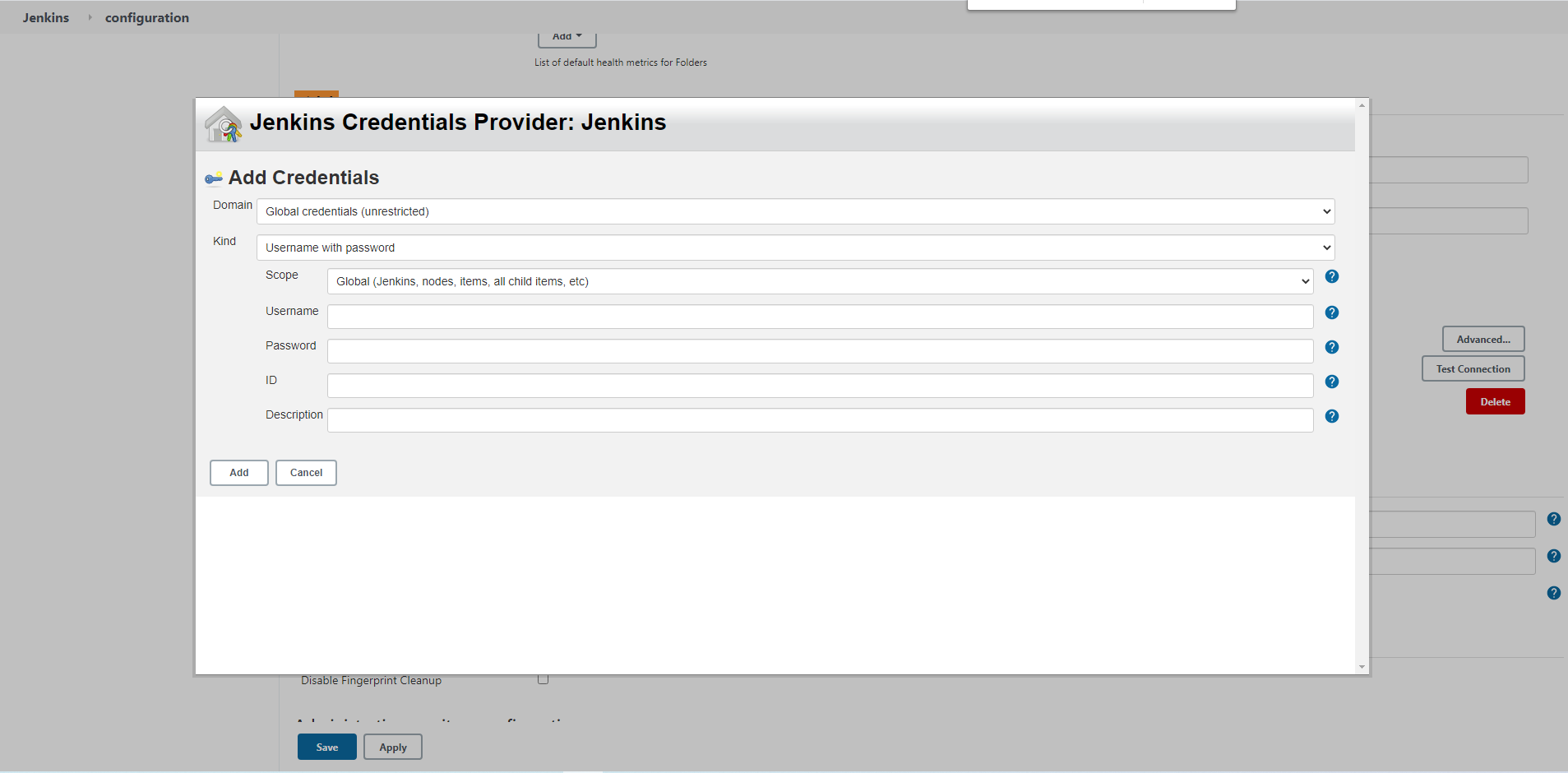
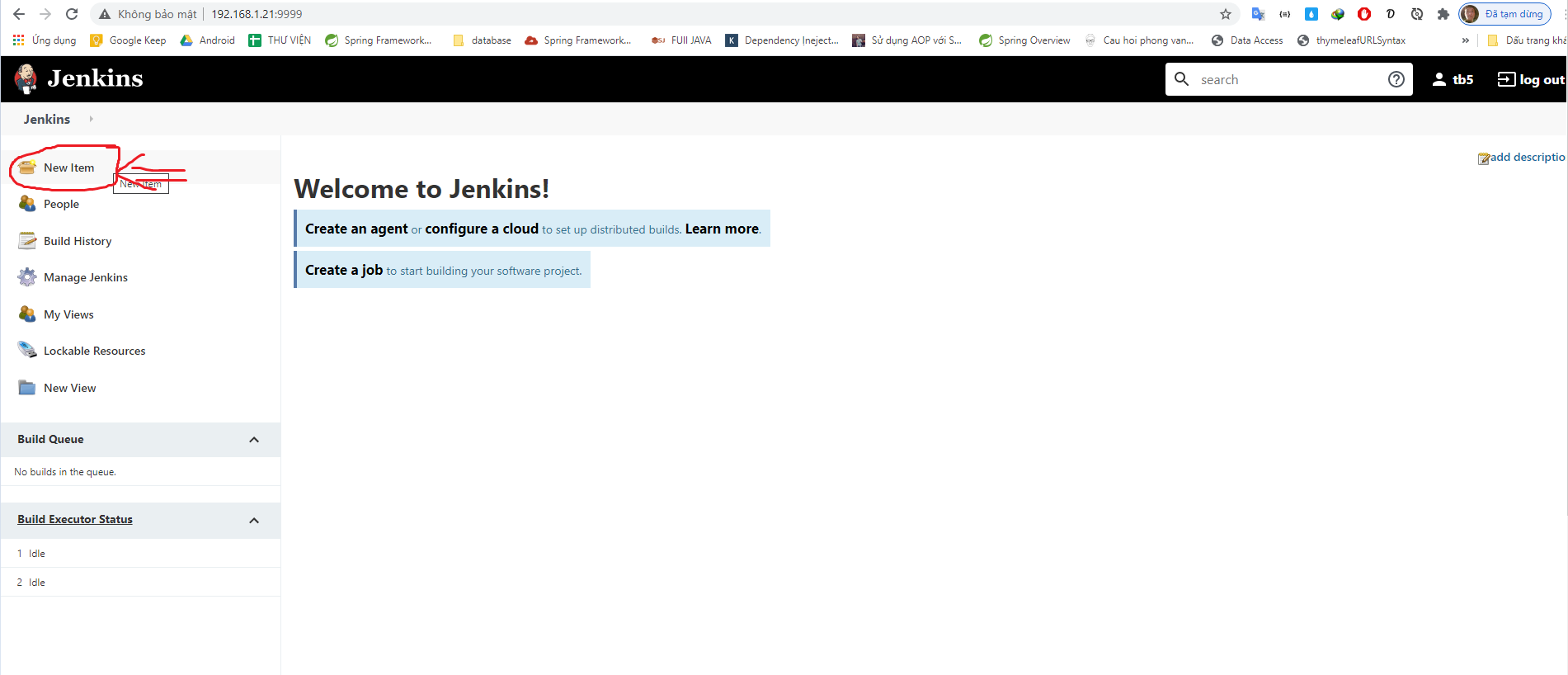
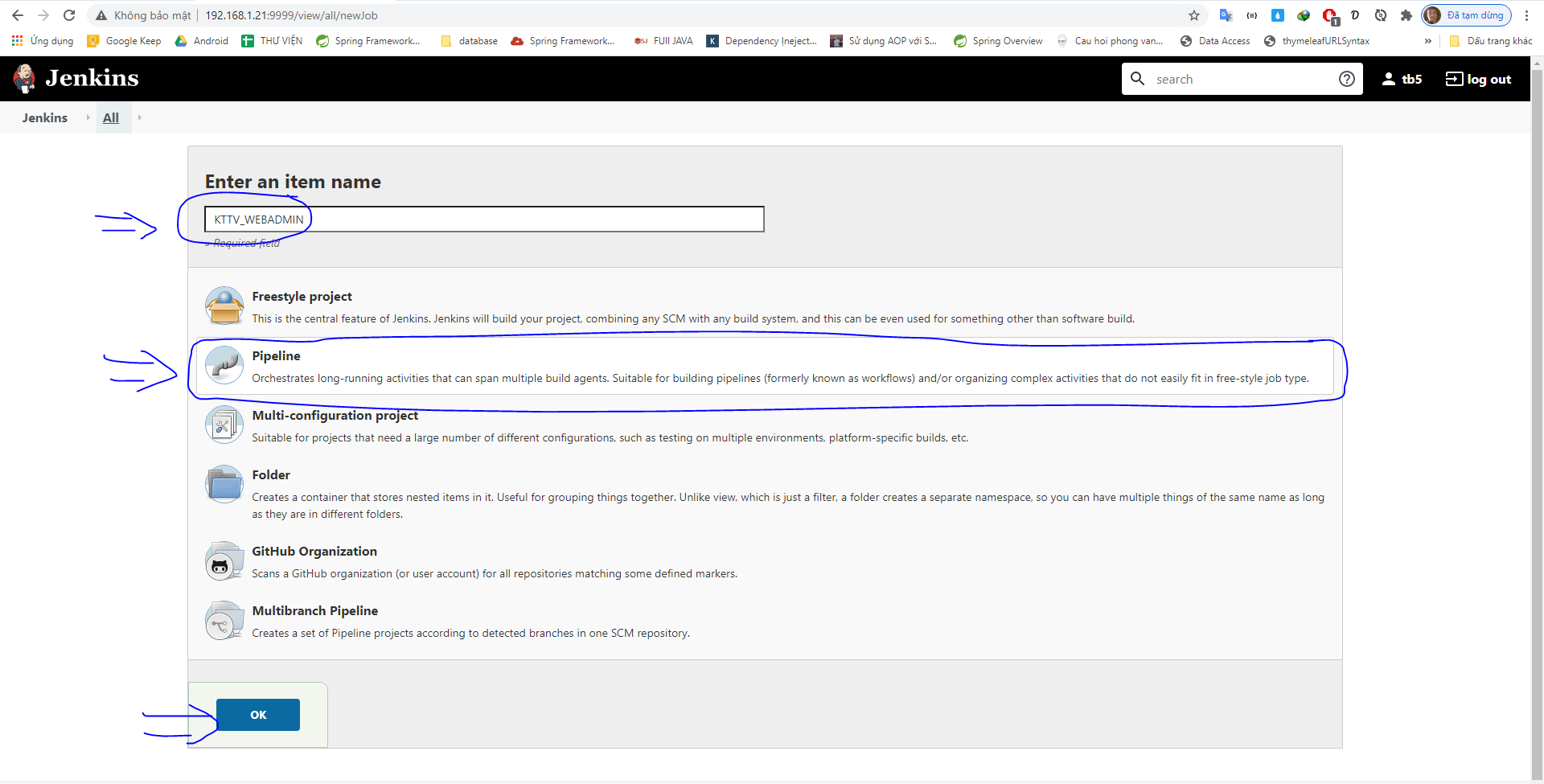
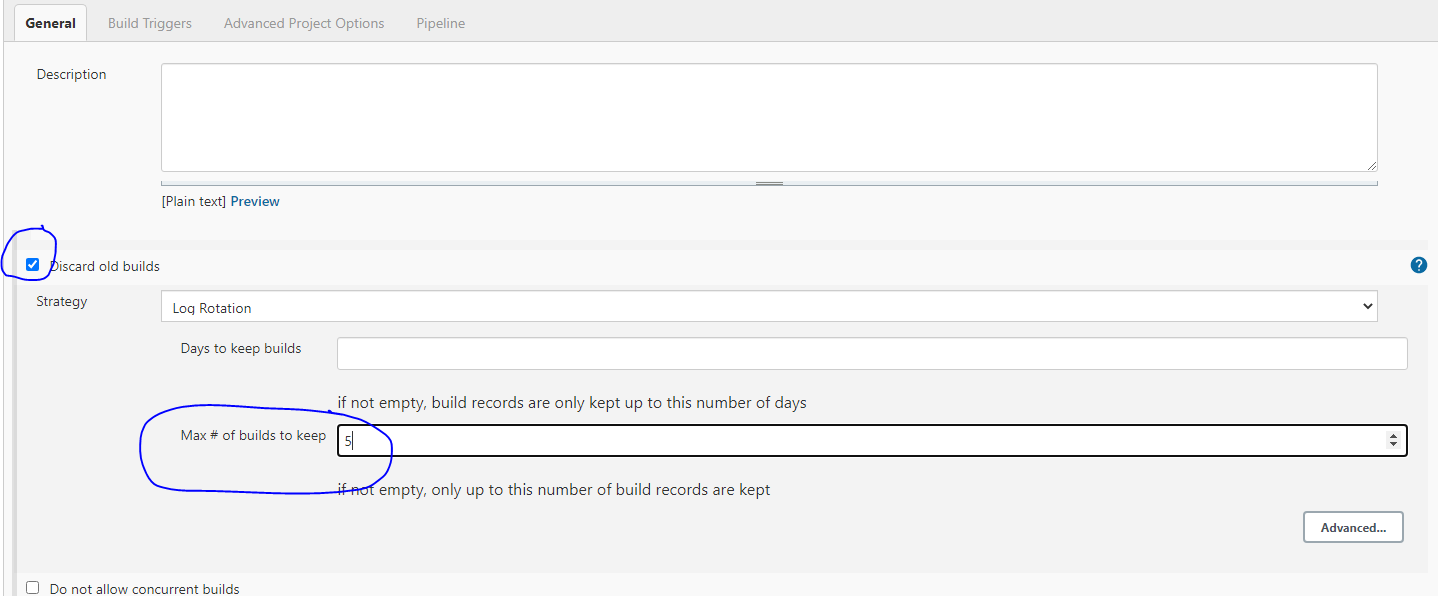
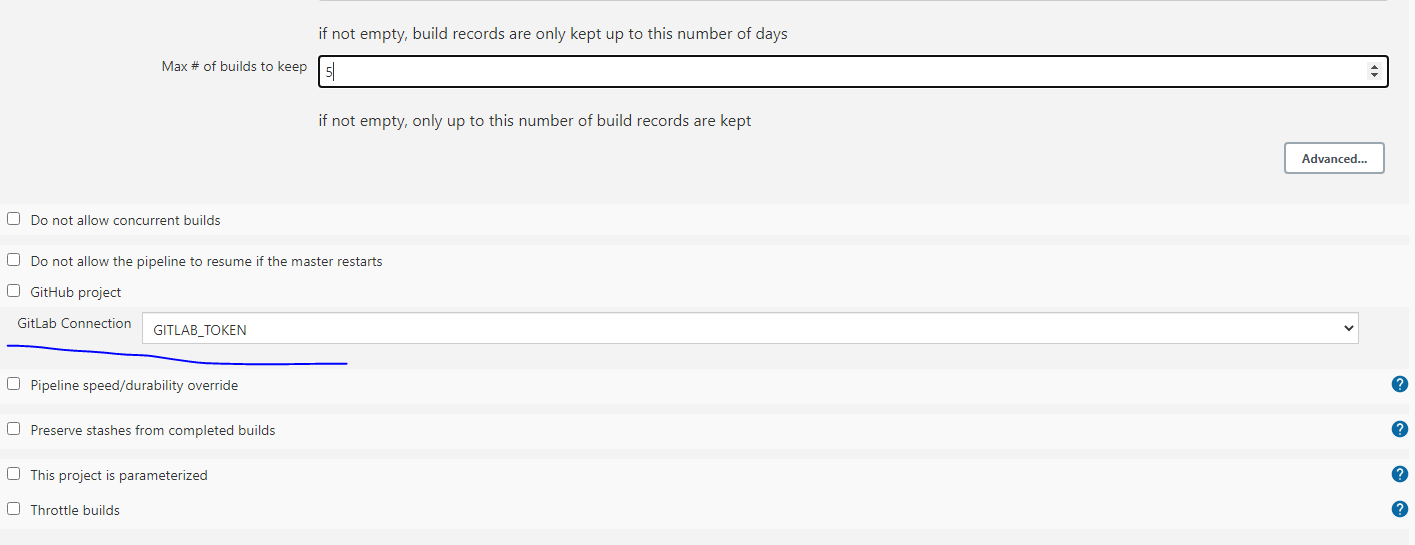
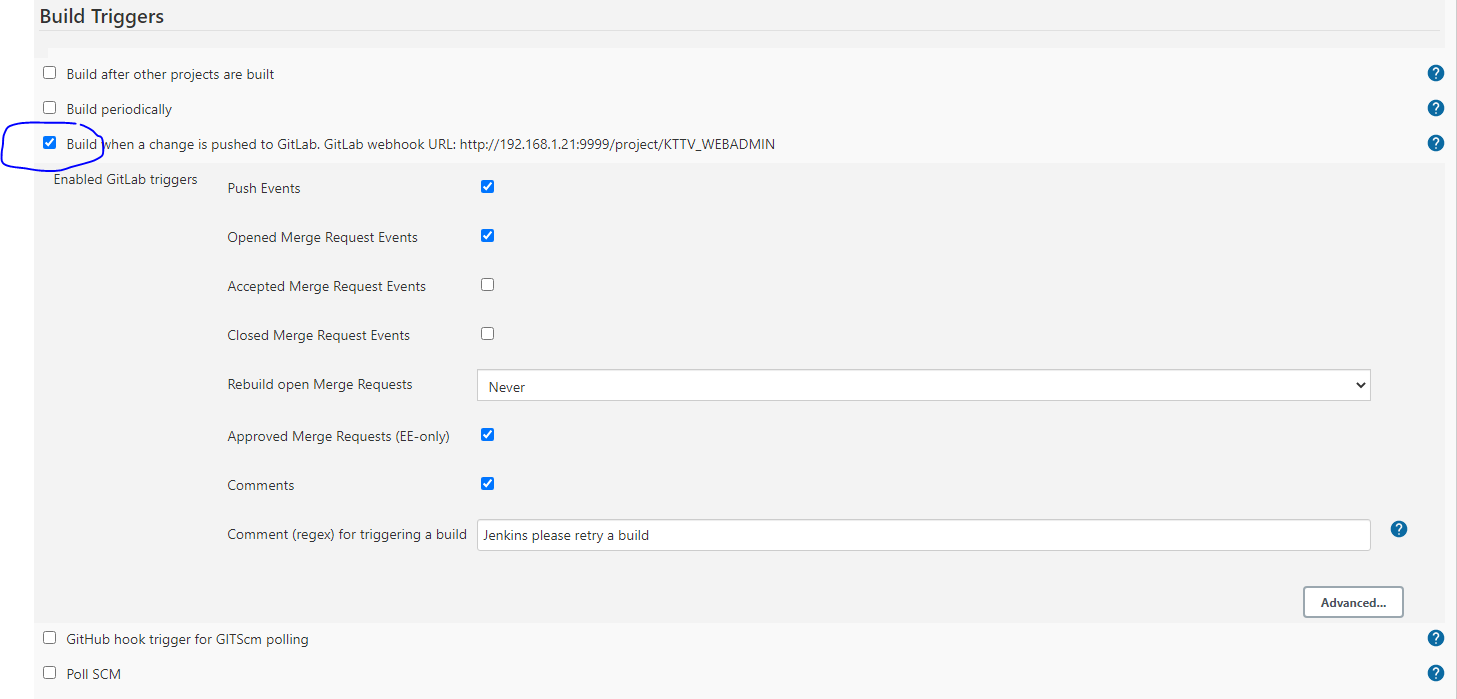
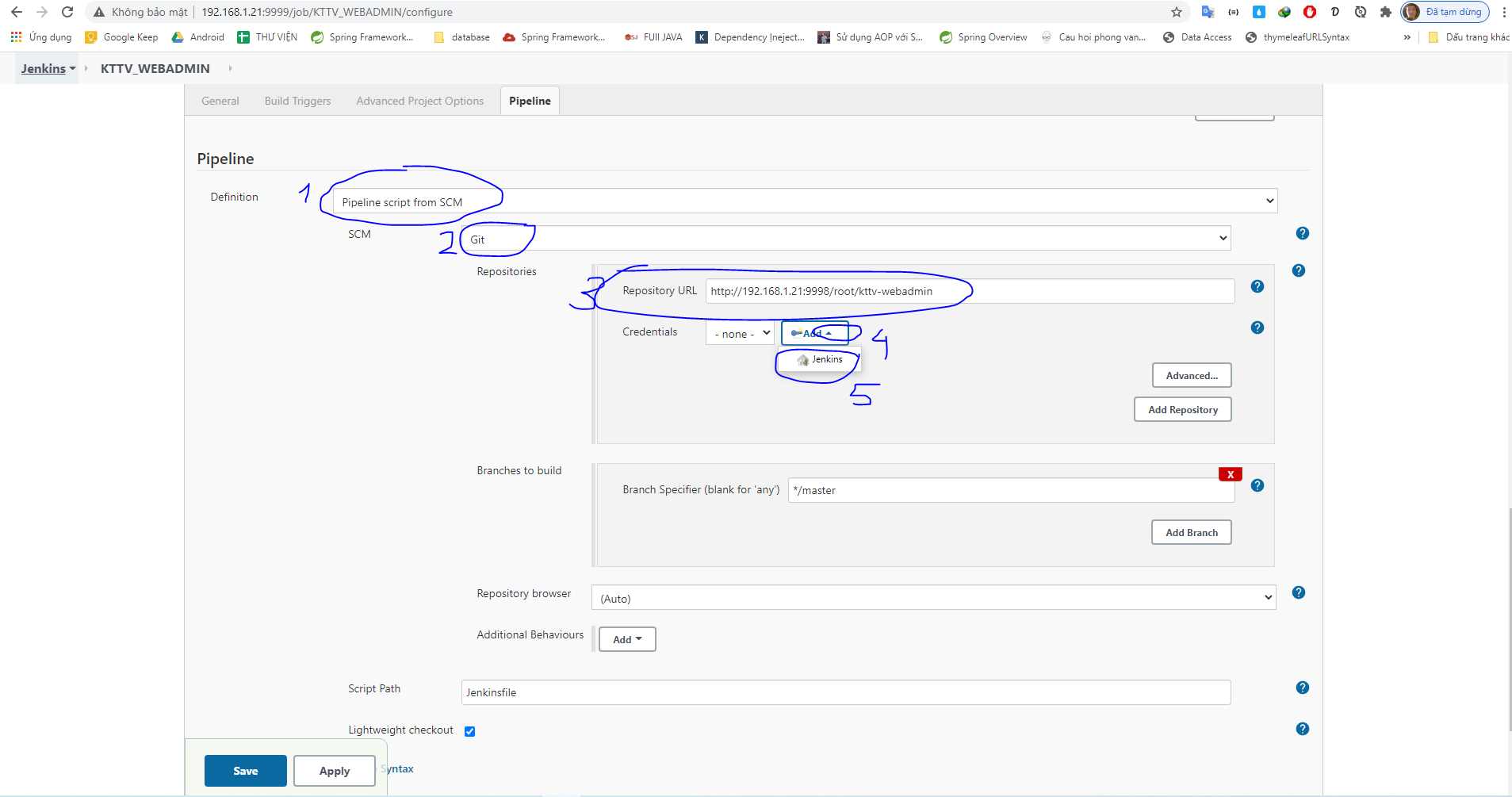
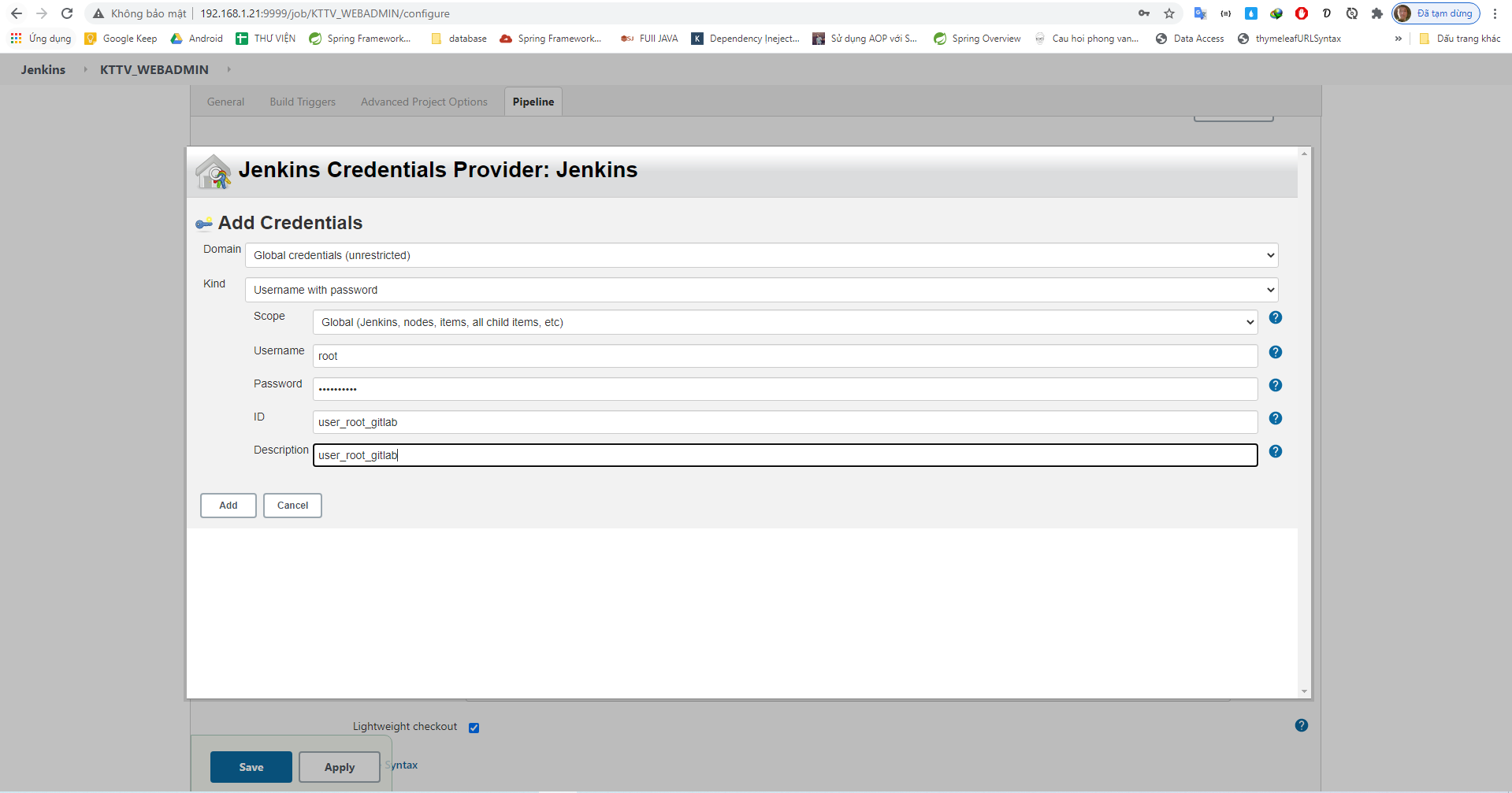
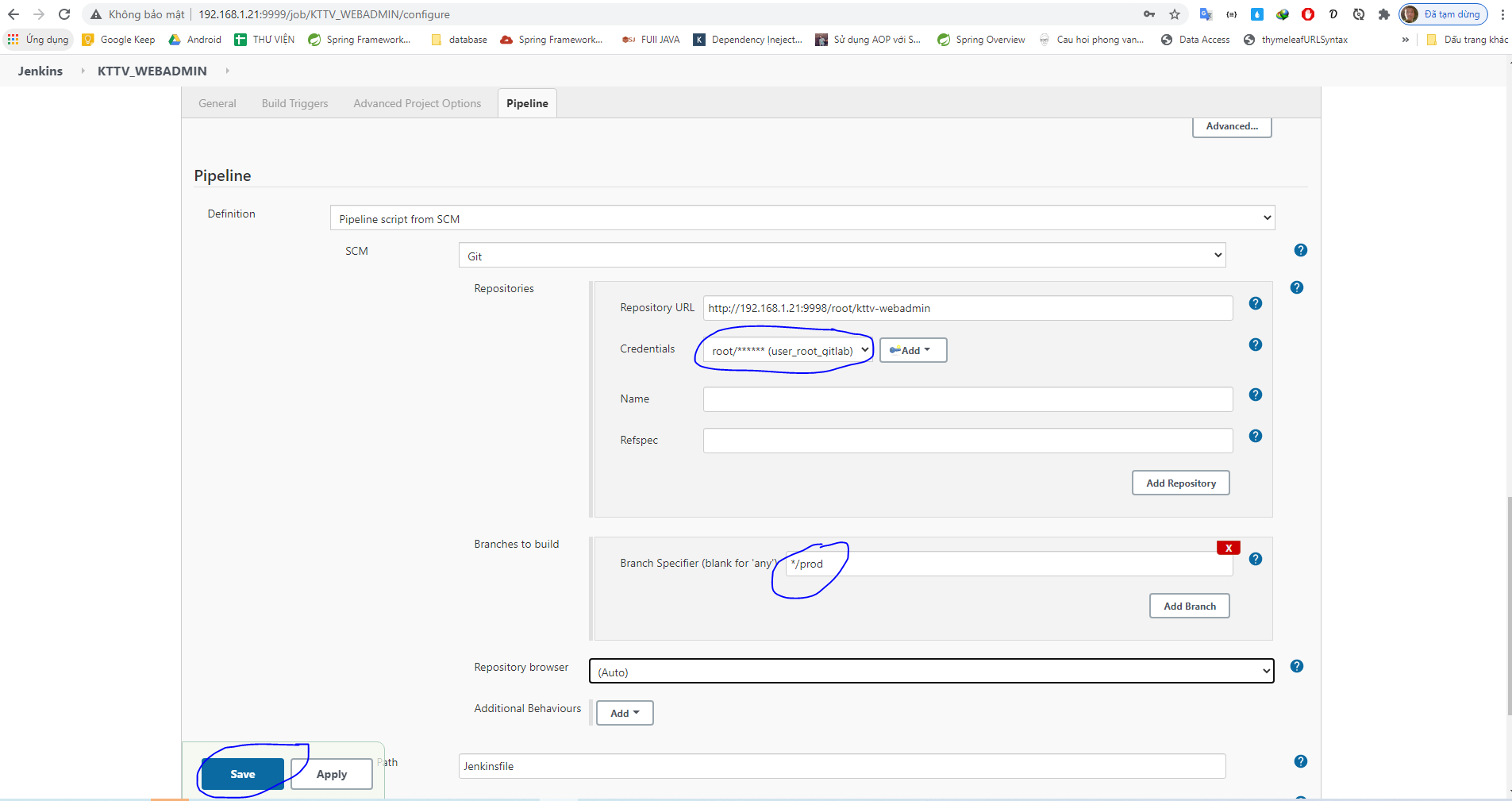
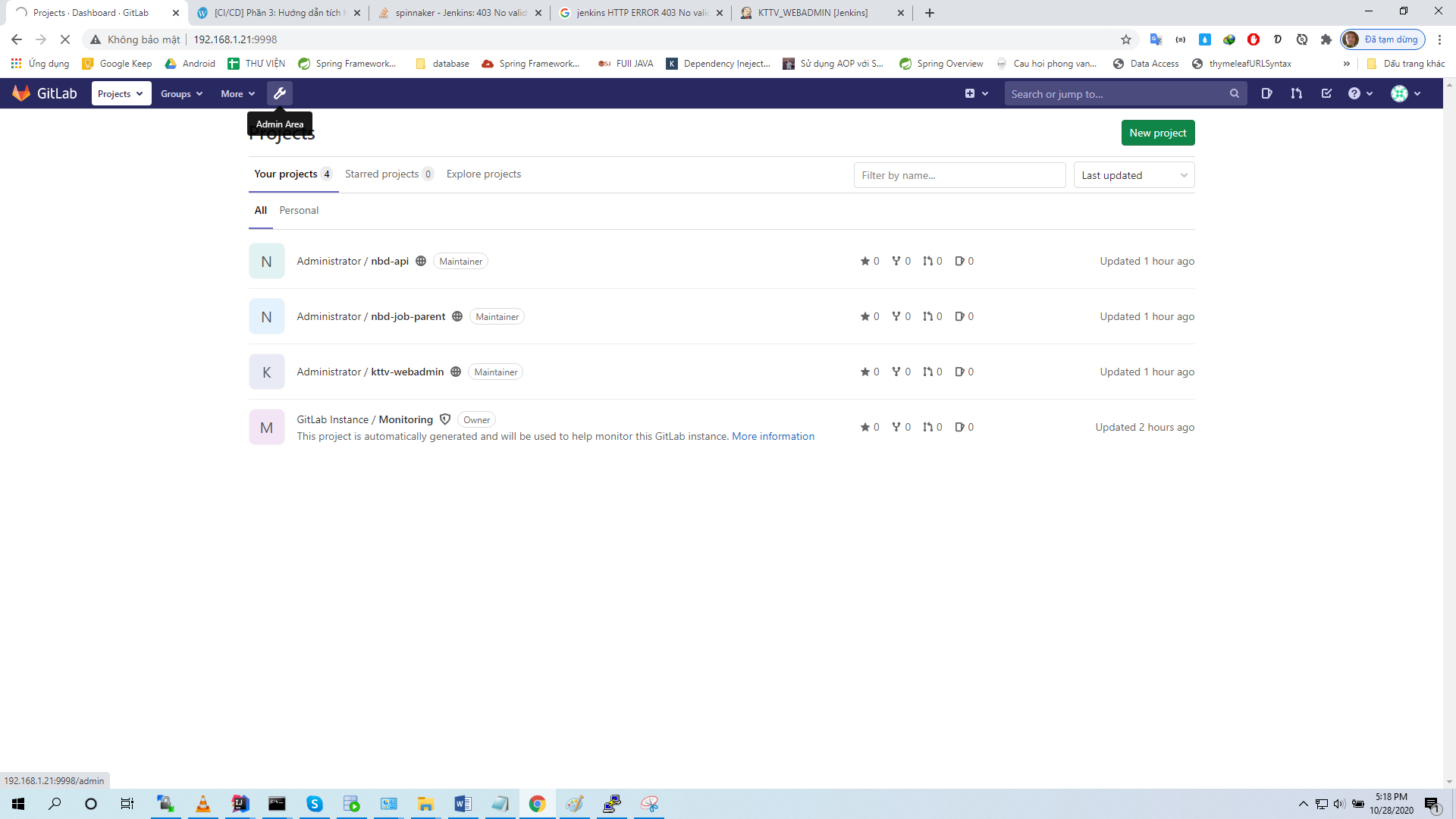
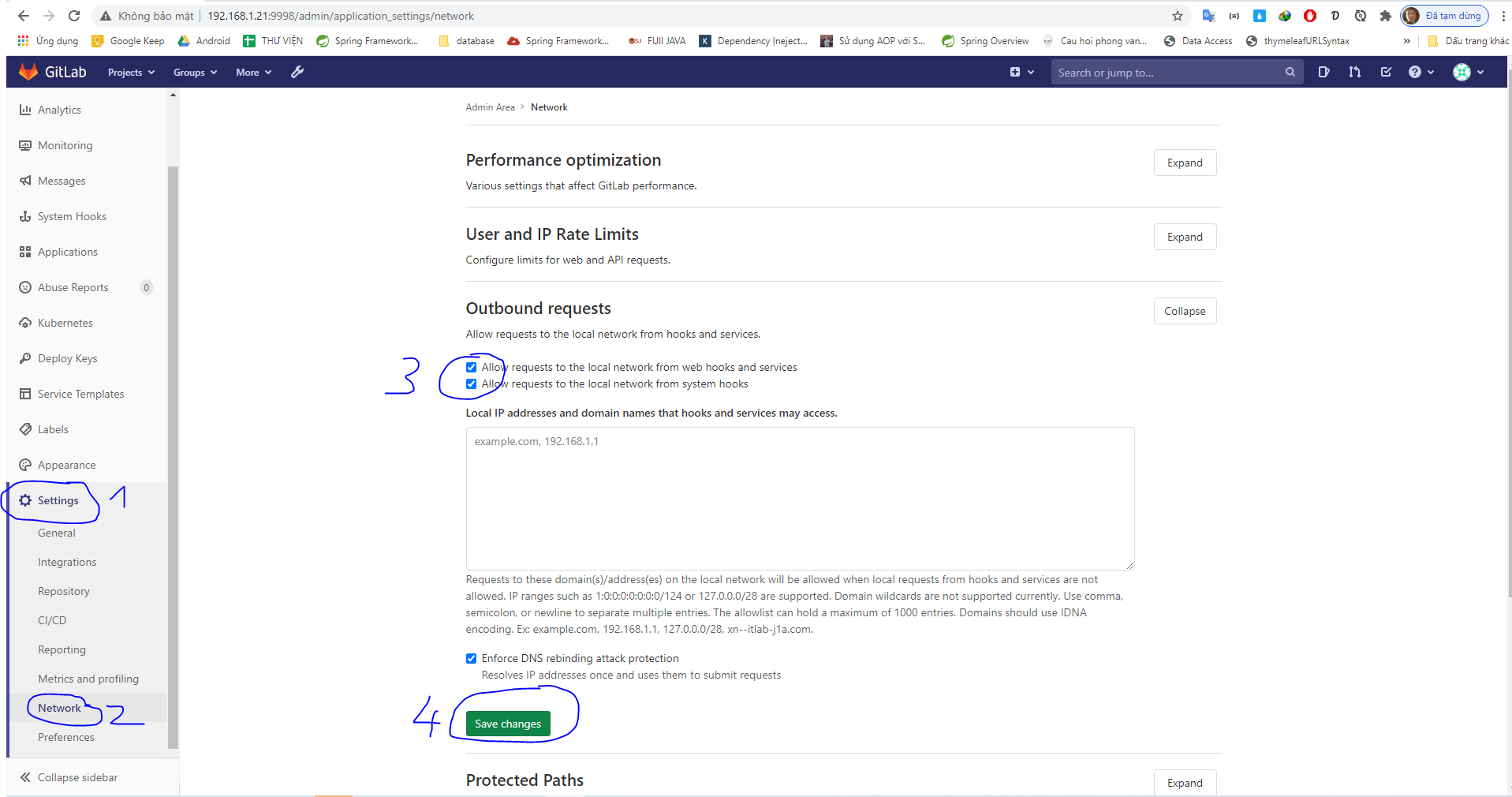
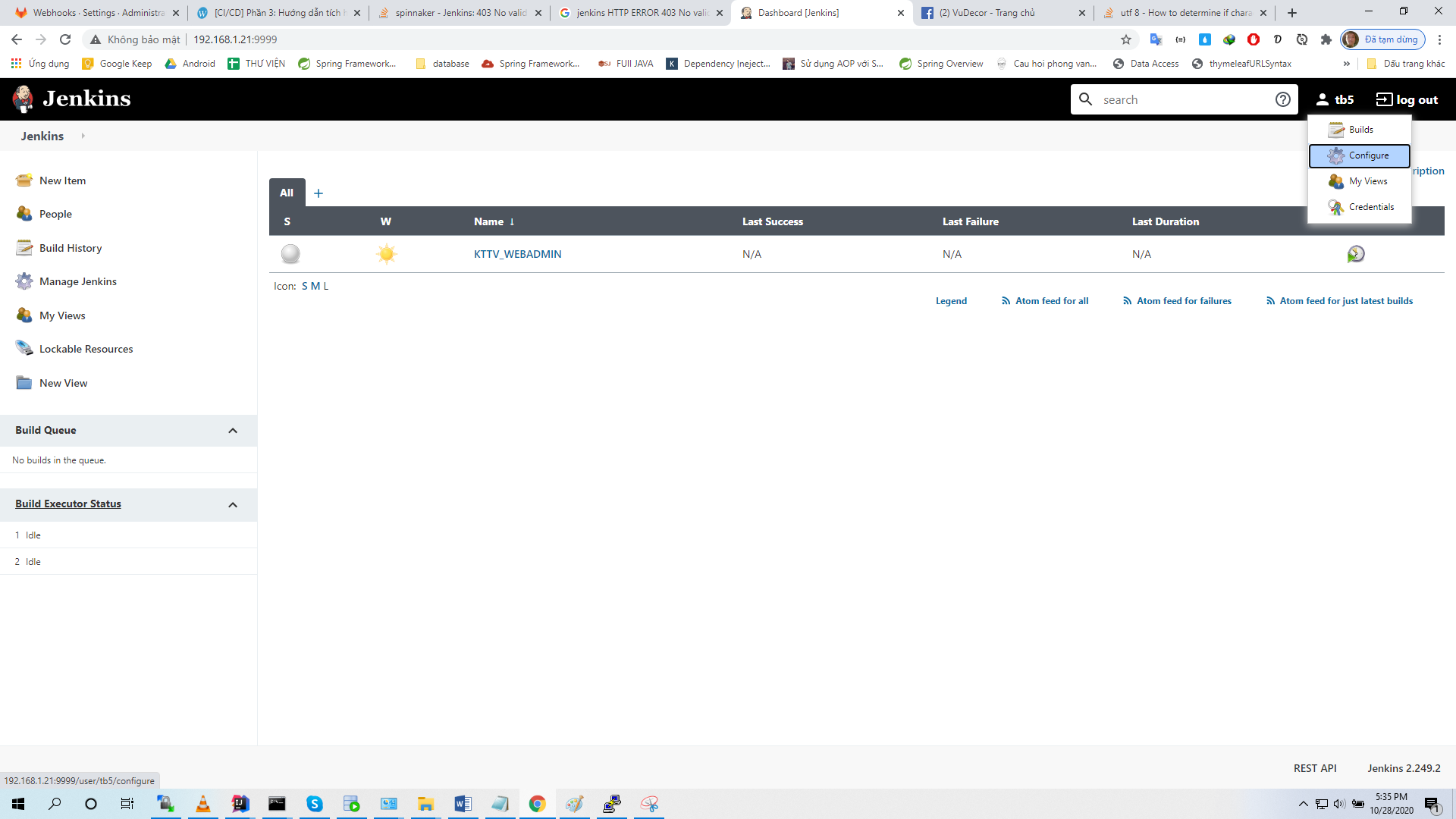
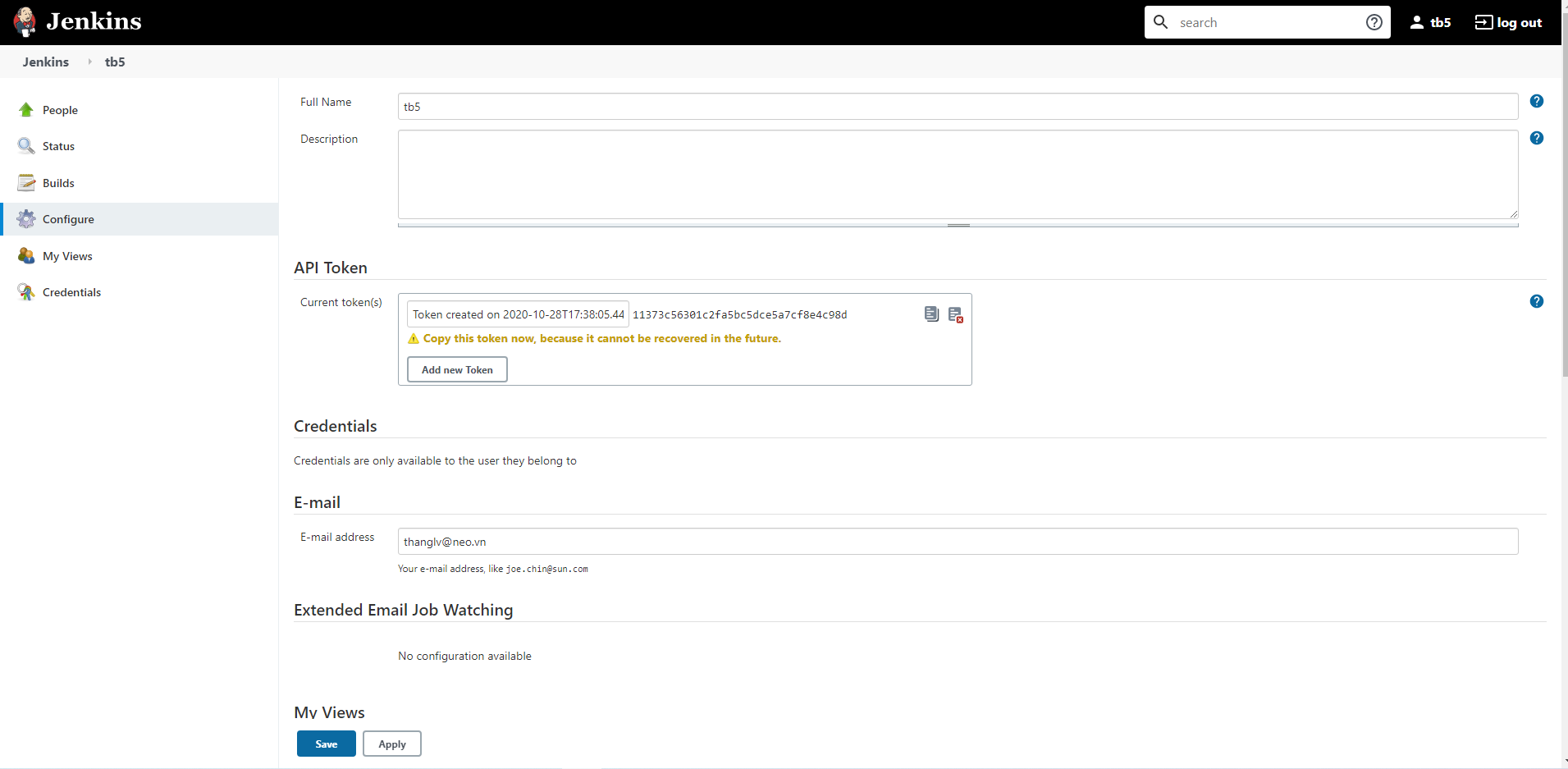
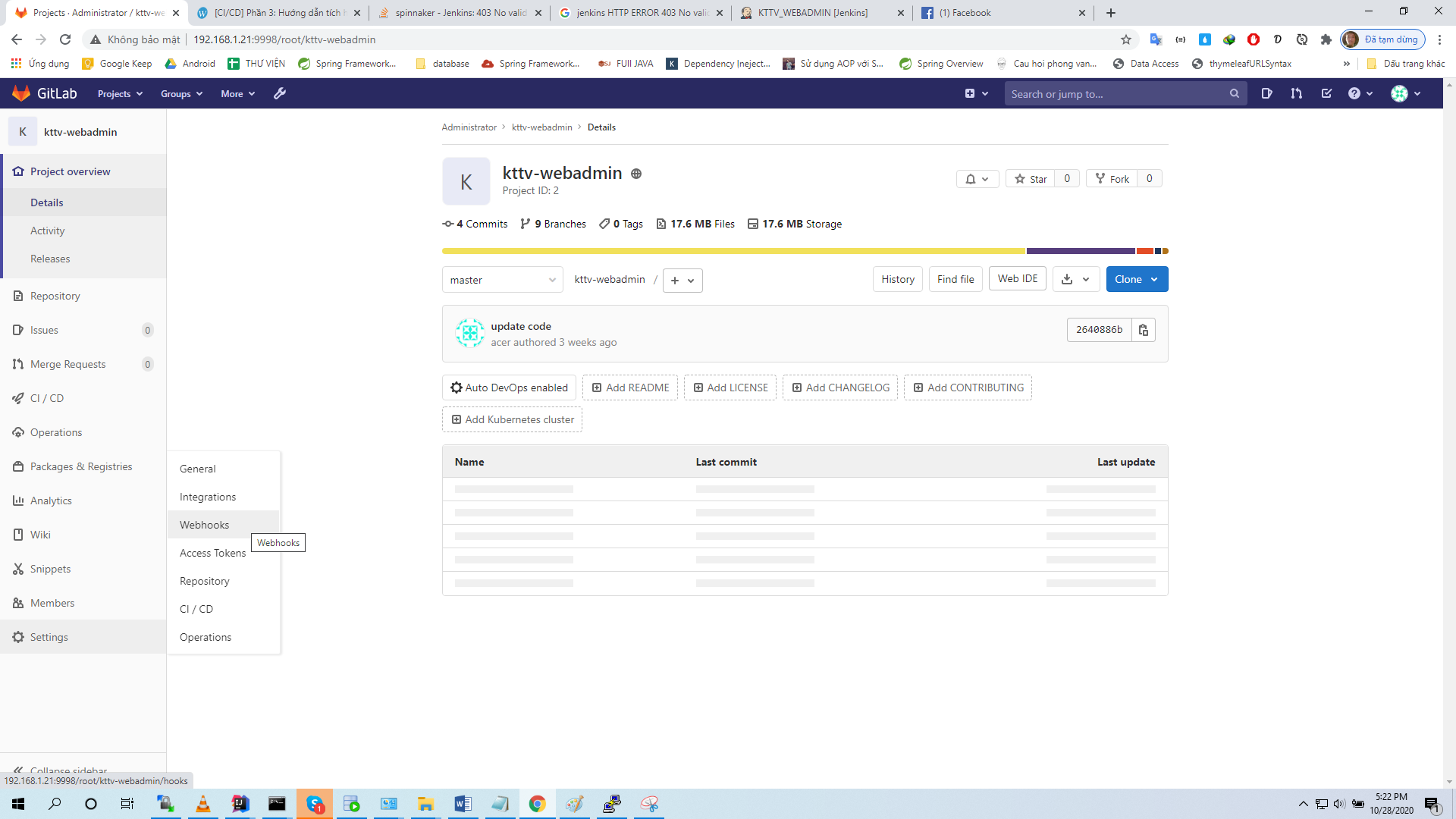
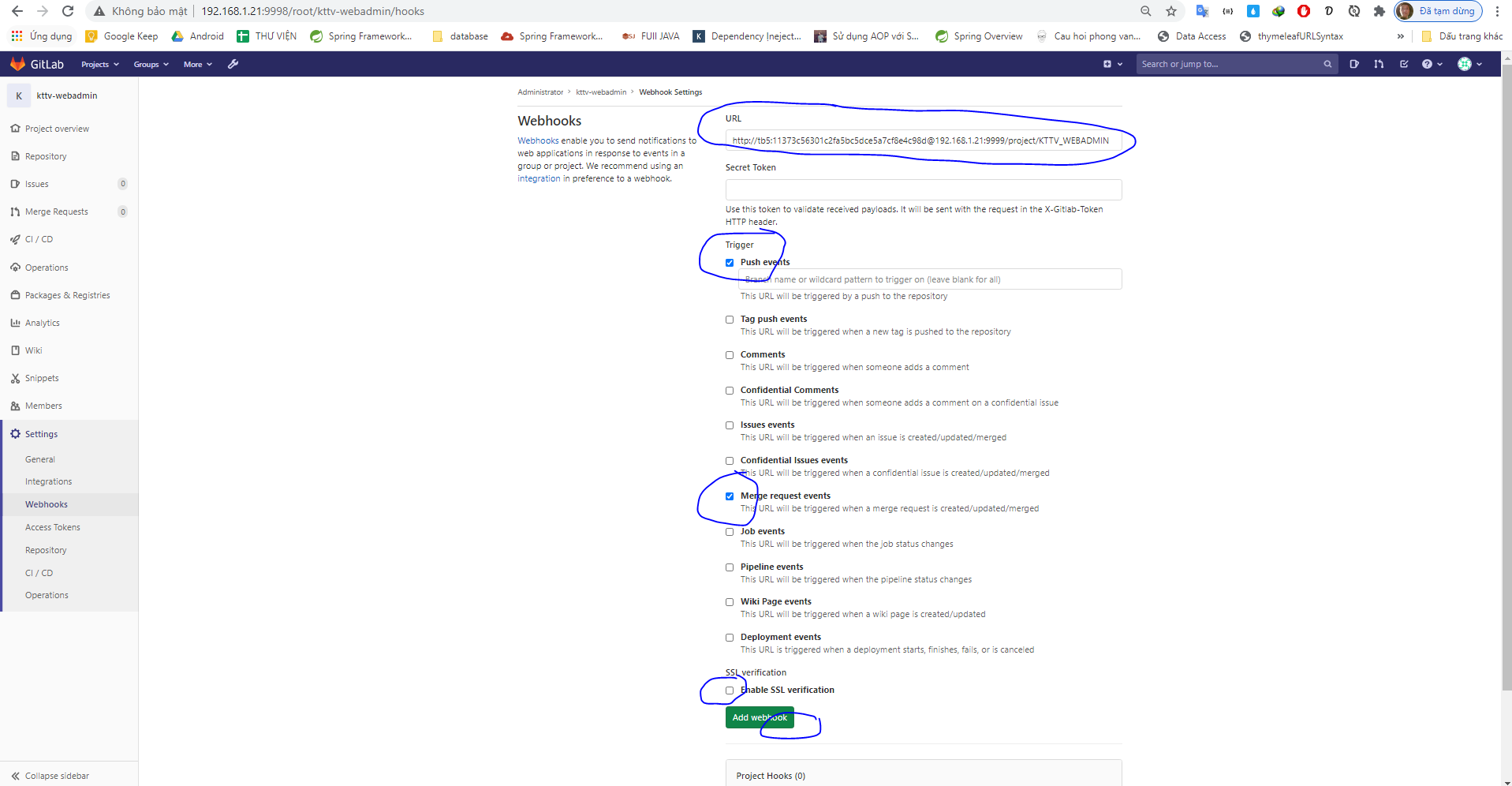
1. **Cài đặt maven lên server**

* Tải maven ở trang web: <https://maven.apache.org/download.cgi>
* Giải nén ở folder: /app với lệnh : tar –zxf apache-maven\*\*\* -C /app
* sudo nano /etc/environment
* Thêm giá trị là đường dẫn đến folder bin của apache maven vào biến PATH  
  PATH= “/app/apache-maven\*\*\*/bin”
* source /etc/environment
* mvn -version

1. **Cài đặt Gitlab lên docker**

* export GITLAB\_HOME=/srv/gitlab
* Tạo folder: /app/gitlab\_setup
* Tạo file docker-compose.yml trong folder vừa tạo với nội dung như bên dưới  
  
* docker-compose up -d
* Sau khi docker triển khai xong container, vào địa chỉ : http://${IP\_SERVER}:9998/ để cấu hình mật khẩu cho Gitlab với tên đăng nhập mặc định là : root
* Chi tiết cài đặt, tham khảo : <https://docs.gitlab.com/omnibus/docker/>

1. **Cấu hình liên kết Jenkins với Docker với Gitlab (Tự động deploy)**

* Source code đang ở trên Github, thực hiện clone source code từ Github về Gitlab server.
* Lấy Personal Access Token của Github
* 
* Ở giao diện Gitlab, chọn New Project > Import project > Github
* Điền Token vừa tạo và chọn Authenticate
* 
* Chọn Project muốn import
* 
* Kết nối Gitlab với Jenkins:
  + Truy cập vào Jenkins theo đường dẫn : <http://IP_SERVER/9999>
  + Cài plugin Gitlab cho Jenkins theo các bước : Manage Jenkins > Manage Plugins > Available > Search : Gitlab > Install without restart  
    
  + Cài plugin : Pipeline Utility Step tương tự như trên  
    
* Cấu hình Credentials:
  + Vào Gitlab, chọn user > setting  
    
  + Chọn Access Token  
    
  + Chọn những options sau và copy token được tạo ra:  
    
* Cấu hình Gitlab Authentication trong Jenkins:
  + Chọn Manage Jenkins > Configuration System  
    
  + Scroll đến gitlab và chọn add  
    
  + Điền các thông tin sau:  
    Connection name: GITLAB\_TOKEN\_AUTH  
    Gitlab host URL: IP của gitlab  
    Chọn Ignore SSL Certificates Errors  
    
  + Chọn Add ở trong phần Credentials > Chọn Jenkins > Chọn Gitlab Token và điền token vào  
    
* Cấu hình Trigger Build Gitlab cho Jenkins
  + Vào website của Jenkins, chọn New Item  
    
  + Điền tên Pipeline và chọn như bên dưới  
    
  + Trong phần General, chọn Discards Old Build và nhập Max # of builds to keep : 5  
    
  + Kiểm tra Gitlab Connection được tạo ở bước trước  
    
  + Tiếp theo, Chọn Build Trigger  
    
  + Chọn các phần 1,2,3,4 và Chọn nút 5  
    
  + Điền thông tin đăng nhập của Gitlab vào form  
    
  + Điền thông tin cuối cùng và lưu lại  
    
* Liên kết Gitlab với Jenkins:
  + Vào Setting Area của Gitlab  
    
  + Chọn các option như sau:  
    
  + Truy cập vào Web Jenkins chọn  
    
  + Chọn Add new token và lưu lại:  
    
  + Vào Repository của project, chọn Setting > Web hooks  
    
  + Điền thông tin Web hooks  
    URL webhooks: http://<jenkins-user>:<jenkin-token>@<host\_jenkins>:<port\_jenkins>/project/<project-jenkins-name>
  + Ví dụ : <http://tb5:11373c56301c2fa5bc5dce5a7cf8e4c98d@192.168.1.20:9999/project/KTTV_WEBADMIN>
  + 
  + Test thử push event

1. **Cài đặt RabbitMQ Clusters**

Tạo folder: /app/rabbitmq\_cluster

Tạo file có tên : docker-compose.yml nội dung như bên dưới bên trong folder vừa tạo  


cd /app/rabbitmq\_cluster

docker-compose up -d

docker container ls

Tìm container có tên : \*\*\_cluster\_rabbit2\_1 vừa tạo và copy id của container đó

docker exec -i -t {CONTAINER\_ID} \bash

rabbitmqctl stop\_app

rabbitmqctl join\_cluster rabbit@rabbit1

rabbitmq start\_app

1. **Cài SSH server lên ubuntu  
   -** sudo apt-get install openssh-server  
   - sudo systemctl enable ssh  
   cd tới file /etc/ssh/sshd\_config  
   - mở tuỳ chọn này lên: #PasswordAuthentication yes và #PermitRootLogin yes  
   sudo systemctl restart ssh
2. **Cài gitlab lên ubuntu server**
   1. **Install and configure the necessary dependencies  
      -** sudo apt-get update  
      - sudo apt-get install -y curl openssh-server ca-certificates tzdata perl  
      - sudo apt-get install -y postfix  
      - curl https://packages.gitlab.com/install/repositories/gitlab/gitlab-ce/script.deb.sh | sudo bash  
      **-** sudo EXTERNAL\_URL="https://gitlab.example.com" apt-get install gitlab-ce  
      - đổi mật khẩu root tại : /etc/gitlab/initial\_root\_password  
        
      - Cài đặt gitlab runner để chạy ci cd trên ubuntu  
      - curl -LJO <https://gitlab-runner-downloads.s3.amazonaws.com/latest/deb/gitlab-runner_amd64.deb>  
      - dpkg -i gitlab-runner\_amd64.deb  
      Vì Gitlab runner chỉ chấp nhận https nên ta phải cài ssl + tls cho gitlab bằng openssl chứ không phải let’s encrypt, chi tiết cài đặt xem file :   
      Register gitlab runner  
      - sudo gitlab-runner register –tls-ca-file=[/dir-crt-file]
   2. **Cài phần mềm monitor os  
      -** sudo apt install glances