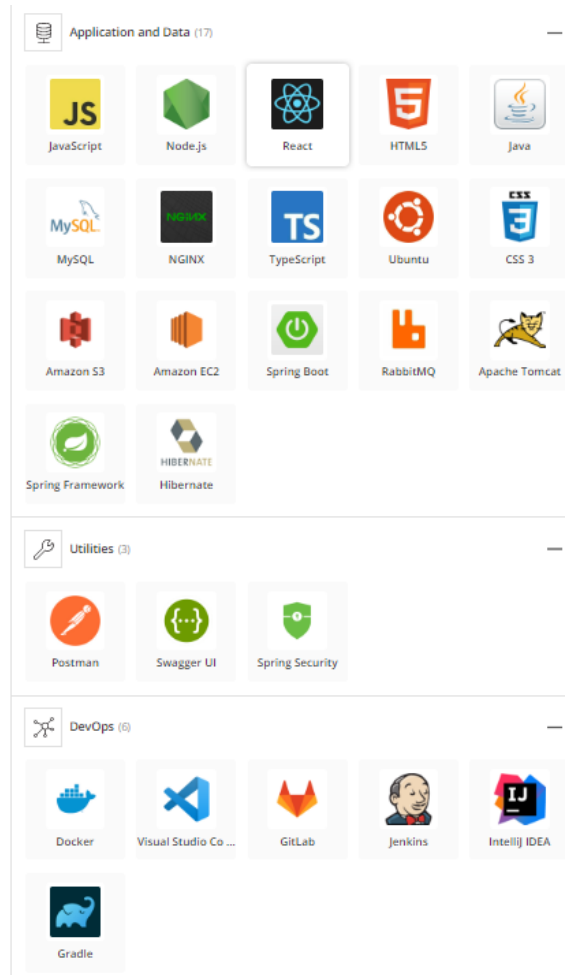


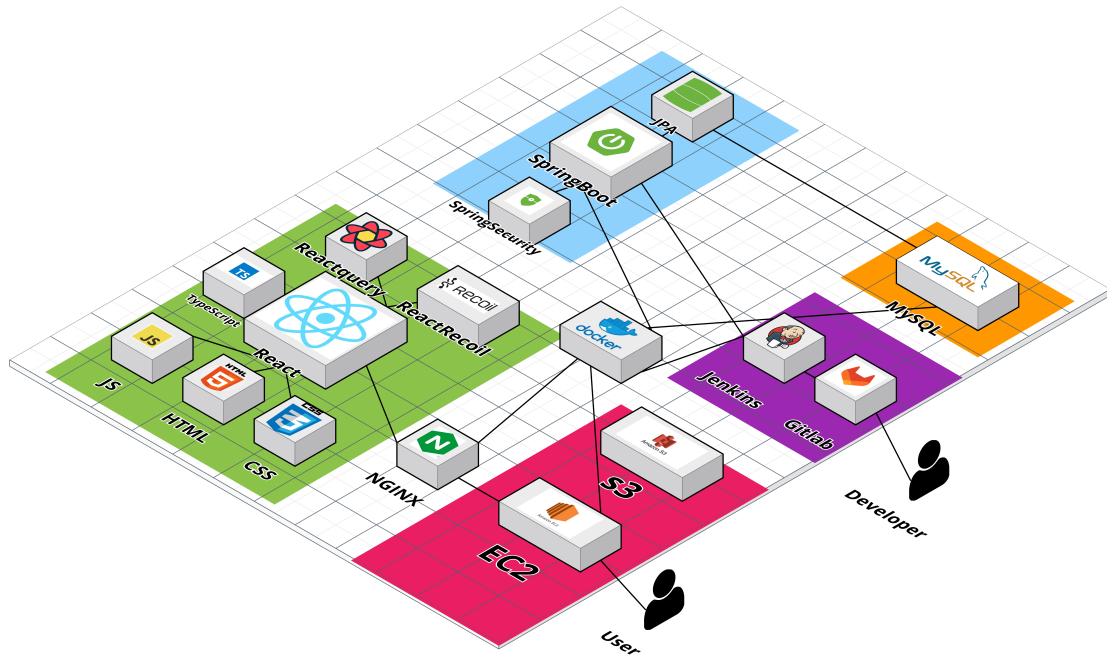
SSAFVEY 포팅 메뉴얼

1. 프로젝트 구성도

기술 스택



시스템 아키텍처



2. Spring 서버 ec2 세팅

Server spec : t2.micro

os : Amazon Linux 2

docker 설치

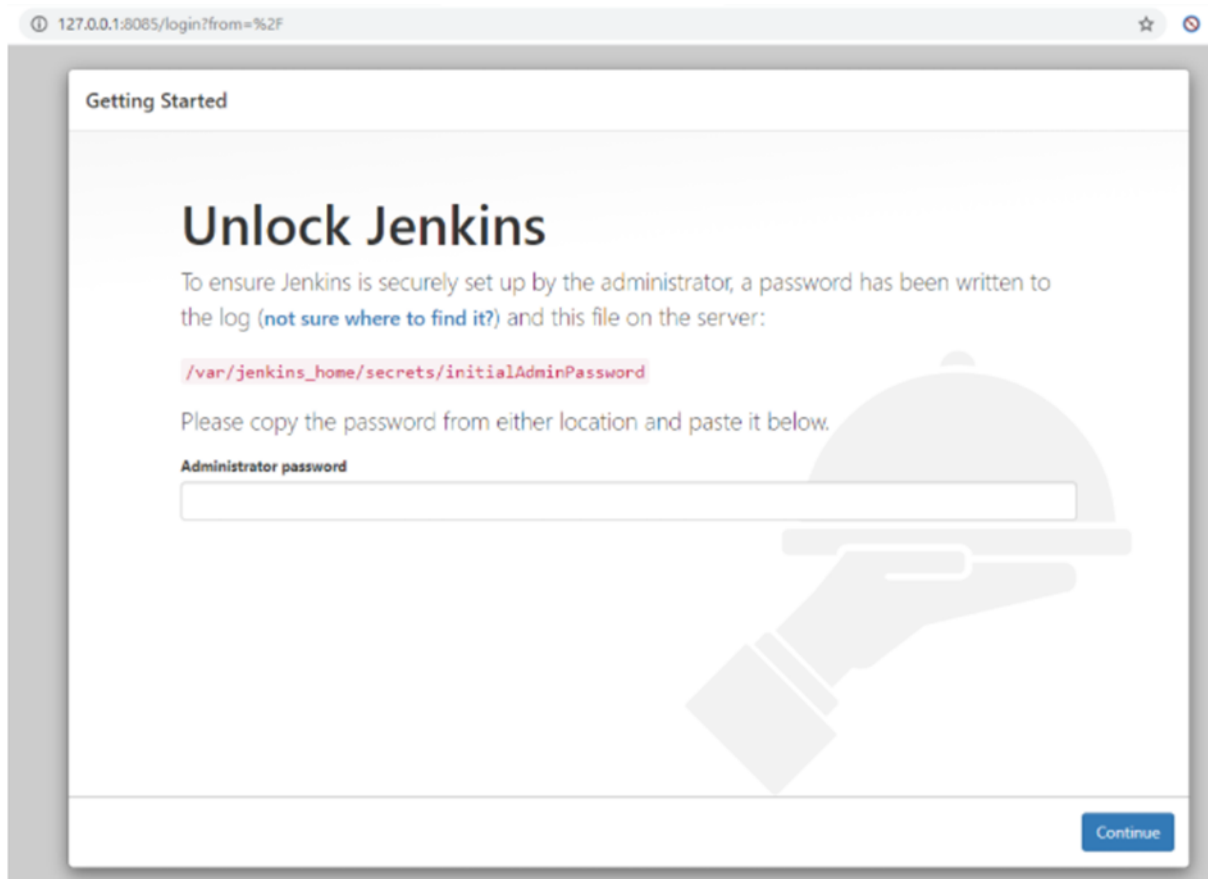
```
// docker 설치

// 패키지 업데이트
sudo apt update
// https관련 패키지 설치
sudo apt install apt-transport-https ca-certificates curl software-properties-common
// docker repository 접근을 위한 gpg 키 설정
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
// docker repository 등록
sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
// 다시 업데이트
sudo apt update
// 도커 설치
sudo apt install docker-ce
// 설치 확인
docker --version
```

3. jenkins 서버 ec2 세팅

```
// Jenkins 설치

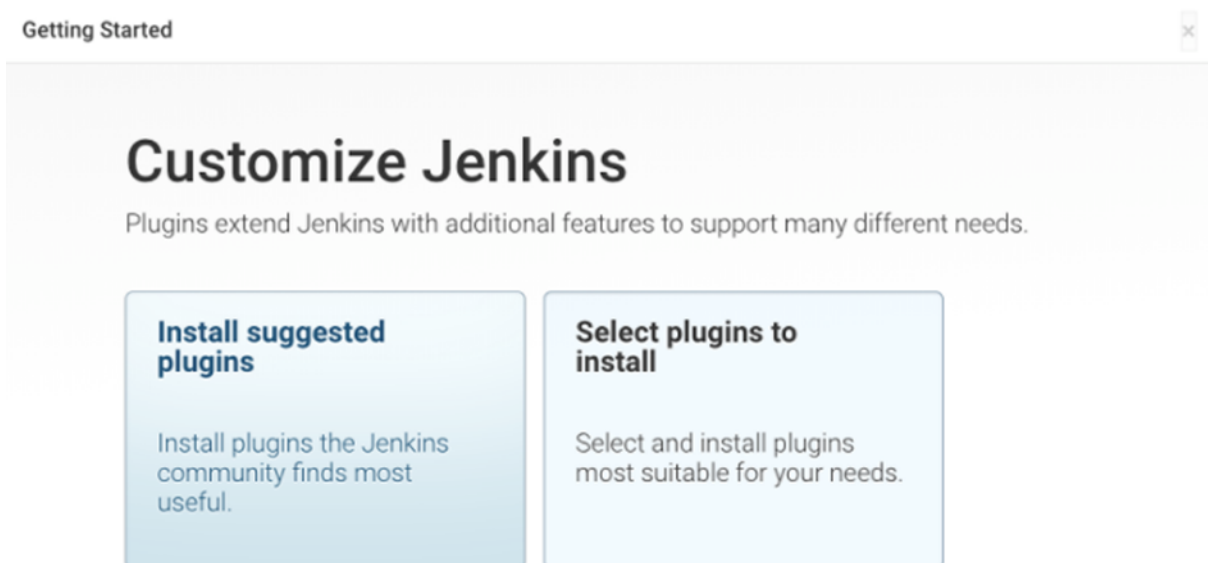
// 젠킨스 이미지 다운로드
docker pull jenkins/jenkins:lts
// 젠킨스 컨테이너 설치 및 실행
docker run -itd -p 8080:8080 -v /jenkins:/var/jenkins_home -name jenkins -u root jenkins/jenkins:lts
```



```
// 젠킨스 컨테이너에 접근해서 어드민 패스워드 찾기
docker exec -it jenkins /bin/bash

cat /var/jenkins_home/secrets/initialAdminPassword

// 혹은 jenkins 로그를 출력해서 초기 비밀번호 확인
docker logs jenkins -f
```



Install suggested plugins 설치

Getting Started

✓ Folders Plugin	✓ OWASP Markup Formatter Plugin	✓ Build Timeout	⌚ Credentials Binding	** SSH server Folders OWASP Markup Formatter ** Structs ** Pipeline: Step API ** Token Macro Build Timeout
⌚ Timestamper	⌚ Workspace Cleanup	⌚ Ant	⌚ Gradle	
⌚ Pipeline	⌚ GitHub Branch Source	⌚ Pipeline: GitHub Groovy Libraries	⌚ Pipeline: Stage View	
⌚ Git	⌚ SSH Build Agents	⌚ Matrix Authorization Strategy	⌚ PAM Authentication	
⌚ LDAP	⌚ Email Extension	⌚ Mailer		

** - required dependency

Jenkins 2.319.1

Getting Started

Create First Admin User

계정명:

암호:

암호 확인:

이름:

이메일 주소:

Jenkins 2.319.1

[Skip and continue as admin](#)

[Save and Continue](#)

설치가 완료되면 계정의 정보 입력

4. Jenkins 파이프라인 생성



The screenshot shows the Jenkins Dashboard interface. On the left is a sidebar with navigation links: '새로운 Item', '사람', '빌드 기록', 'Jenkins 관리', 'My Views', 'Lockable Resources', and 'New View'. Below these are sections for '빌드 대기 목록' (Build Queue) and '빌드 실행 상태' (Build Execution Status). The main content area displays a welcome message in Korean: 'Jenkins에 오신 것을 환영합니다.' (Welcome to Jenkins). Below this, it says 'Start building your software project' and provides two main options: 'Create a job' and 'Set up a distributed build'. The 'Set up a distributed build' section includes sub-options: 'Set up an agent', 'Configure a cloud', and 'Learn more about distributed builds'.

필요 플러그인 설치

- Generic Webhook Trigger Plugin
- GitLab

- Gitlab API Plugin
- GitLab Authentication plugin
- Mattermost Notification Plugin
- Publish Over SSH

Credentials

T	P	Store ↓	Domain	ID	Name
		System	(global)	LeeYouYoung	dldbud112@naver.com/***** (LYID)

Jenkins 관리 → Manage Credentials → Credentials 등록

Item 생성(Pipeline)

설정 셋팅하기

Build Triggers

☐ Build after other projects are built ?
☐ Build periodically ?
☒ Build when a change is pushed to GitLab. GitLab webhook URL: <http://k8a608.p.ssafy.io:8080/project/FE> ?

Enabled GitLab triggers

☒ Push Events
☐ Push Events in case of branch delete
☒ Opened Merge Request Events
☐ Build only if new commits were pushed to Merge Request ?
☐ Accepted Merge Request Events
☐ Closed Merge Request Events

Rebuild open Merge Requests

Never ▼

☒ Approved Merge Requests (EE-only)
☒ Comments

Comment (regex) for triggering a build ?

Jenkins please retry a build

고급 ▼

☐ GitHub hook trigger for GITScm polling ?

Pipeline

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

https://lab.ssfy.com/s08-final/S08P31A608

Credentials ?

dlibdud112@naver.com/***** (LYYID)

Add +

Repository browser ?

(자동)

Additional Behaviours

Add +

Script Path ?

jenkins/frontjenkinstfile

5. Webhook 연결(Build Triggers URL과 Secret token 복사)

S S08P31A608

- Project information
- Repository
- Issues 0
- Merge requests 0
- CI/CD
- Security & Compliance
- Deployments
- Packages and registries
- Infrastructure
- Monitor
- Analytics
- Wiki
- Snippets
- Settings

- General
- Integrations
- Webhooks
- Access Tokens
- Repository
- Merge requests
- CI/CD
- Packages and registries
- Monitor
- Usage Quotas

Webhook

Webhooks enable you to send notifications to web applications in response to events in a group or project. We recommend using an [integration](#) in preference to a webhook.

URL

URL must be percent-encoded if it contains one or more special characters.

Secret token

Used to validate received payloads. Sent with the request in the `X-Gitlab-Token` HTTP header.

Trigger

☒ Push events

Push to the repository.

GitLab → Settings → WebURL Secret token 복사 붙여넣기

6. 프로젝트 이미지 생성 후 컨테이너 띄우기

Backend Pipeline

```
pipeline{
  agent any
  stages{
    stage('백엔드 자동 배포') {
      stages {
        stage('gradlew 권한'){
          steps{
            dir('BE'){
              dir('ssafvey'){
                sh "chmod +x gradlew"
              }
            }
          }
        }
        stage('백엔드 이미지 생성'){
          steps{
            dir('BE'){
              dir('ssafvey'){
                sh "./gradlew bootBuildImage"
              }
            }
          }
        }
        stage('백엔드 컨테이너 삭제'){
          steps{
            catchError{
              sh "docker stop backend && docker rm backend"
            }
          }
        }
        stage('백엔드 컨테이너 생성') {
          steps {
            sh "docker run --network=ournetwork -d -p 8081:8080 -e LC_ALL=C.UTF-8 --name backend ssafvey:0.0.1-SNAPSHOT"
          }
        }
      }
    }
  }
}
```

Frontend Pipeline

```
pipeline {
  agent any
  stages {
    stage('프론트엔드 자동 배포') {
      steps {
        dir('FE') {
          sh "docker build . -t front_builder"
        }
      }
    }
  }
}
```


Frontend Dockerfile

2개의 Pipeline 생성

7. MYSQL 생성

8. Nginx 설정

SSAFVEY 포팅 메뉴얼

```

        gzip on;
        gzip_types text/plain application/x-javascript text/xml text/css application/xml application/javascript;
        gzip_vary on;
        gzip_min_length 10240;
        gzip_comp_level 5;
        gzip_disable "MSIE [1-6].(?!.*SV1)";
    }

    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/k8a608.p.ssafy.io/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/k8a608.p.ssafy.io/privkey.pem; # managed by Certbot
}
server {
    if ($host = k8a608.p.ssafy.io) {
        return 301 https://$host$request_uri;
    } # managed by Certbot
    listen 80;
    server_name k8a608.p.ssafy.io;
    return 404; # managed by Certbot
}

```