CI/CD 구축 & GHE 연동

Tangunsoft Co., Ltd.
Eric.Kim
Tel : 02-6206-2545
E-Mail : kch@tangunsoft.com

tan:gun
(주)단군소프트
소·프·트·웨·어·의·종·착·지

Agenda

- CI/CD와 Jenkins
- Build&Deploy Architecture
- Jenkins 설치
- Jenkins 환경설정
- 배포서버 구성
- 송출서버 구성
- Jenkins 프로젝트 구성
- CI/CD에서 DevOps로

CI/CD란 무엇인가

Continuous Integration / Continuous Delivery & Deployment

애플리케이션 개발 단계를 자동화하여 애플리케이션을 보다
짧은 주기로 고객에게 제공하는 방법

CI/CD 구축의 필요성

개발자들이 업무시간 중 코딩에 투자하는 시간은..?

업무 시간 중 약 48% 에만 코딩

그럼 대체, 어디에 시간을 투자하고 있을까?

- **툴 세팅**
- **미팅**
- **인프라 문제**
- **관리자 승인 대기**

여러분의 하루 일과는 어떤가요?

Jenkins란?

Jenkins

- OpenSource CI Tool
- 2004년부터 Sun Microsystems의 가와구치 코스케 에 의해 개발
- Sun Microsystems가 Oracle에 인수된 후 라이선스 이슈로 Jenkins라는 별명의 프로젝트로 분기됨
- 약 1700개(2019년 11월 기준) 의 다양한 Plug-In을 기반으로 사용자가 원하는 형태로 CI서버 구성 가능

Jenkins

Build & Deploy Automation Architecture

Jenkins 설치

Jenkins Download

- https://jenkins.io/download/
- LTS Version의 Winodws OS Installer File Download

Jenkins 설치

Jenkins Install

- Download 받은 파일의 압축을 풀고 설치 파일 실행
- 설치 경로 설정하고 Install 진행

Jenkins 설치

Jenkins 초기 설정

- Jenkins 설치경로 내 initialAdminPassword 파일을 열어 Password Text를 복사.
- Administrator password란에 복사한 Password를 삽입 후 Continue 버튼 클릭

Jenkins 설치

Jenkins 초기 설정

- Install Suggested Plugin – Most Plugin Default 설치
- Select Plugins to Install - Most Plugin Select 설치
- Install Suggested Plugin 클릭

Jenkins 설치

Jenkins 초기 설정

- Most Plugin 설치 진행

Jenkins 설치

Jenkins 초기 설정

- Jenkins 관리자 계정 생성
- 관리자 계정 정보 입력 후 Save and Continue 버튼 클릭

Jenkins 설치

Jenkins 초기 설정

- Jenkins Service URL 확인 후 Save and Finish 버튼 클릭

Jenkins 설치

Jenkins 초기 설정

- 초기 설정 완료. Start Using Jenkins 버튼 클릭

Jenkins 설치

Jenkins 초기 설정

- Jenkins 초기 화면

Jenkins 설치

Jenkins 초기 설정

- Server Local의 시스템 검색에서 “Windows Firewall with Advanced Security 검색
- Inbound Rules에서 Jenkins 등록 확인

Jenkins 환경설정

JDK 설치 확인

- 검색 -> cmd
- 명령 프롬프트 창에서 java -version

Administrator: Command Prompt

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\isb>java -version
'java' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\isb>_
```

Jenkins 환경설정

JDK 설치

- https://www.oracle.com/technetwork/java/javase/downloads/index.html
- Java SE 11.0.5 (LTS) Download
- Download 시 Oracle 계정 필요

Jenkins 환경설정

JDK 설치

- download this software. Accept License Agreement

  라디오 버튼 항목 선택

- jdk-11.0.5_windows-x64_bin.exe 클릭하여 설치 파일

  Download

Java SE Development Kit 11.0.5

You must accept the Oracle Technology Network License Agreement for Oracle Java SE to download this software.

1 Accept License Agreement Decline License Agreement

Product / File Description	File Size	Download
Linux	147.82 MB	jdk-11.0.5_linux-x64_bin.deb
Linux	154.47 MB	jdk-11.0.5_linux-x64_bin.rpm
Linux	171.62 MB	jdk-11.0.5_linux-x64_bin.tar.gz
macOS	166.73 MB	jdk-11.0.5_osx-x64_bin.dmg
macOS	167.06 MB	jdk-11.0.5_osx-x64_bin.tar.gz
Solaris SPARC	188.32 M	2 jdk-11.0.5_solaris-sparcv9_bin.tar.gz
Windows	151.39 MB	jdk-11.0.5_windows-x64_bin.exe
Windows	171.47 MB	jdk-11.0.5_windows-x64_bin.zip

Jenkins 환경설정

JDK 설치

- JDK 설치 파일 실행
- Default 설정으로 설치 진행

Jenkins 환경설정

JDK 설치

- 제어판 -> 시스템 및 보안 -> 시스템 ->

  고급 시스템 설정 클릭

Jenkins 환경설정

JDK 설치

- 제어판 -> 시스템 및 보안 -> 시스템 ->

  고급 시스템 설정 -> 고급 -> 환경변수

Jenkins 환경설정

JDK 설치

- 환경변수 창에서 시스템 변수의 "새로 만들기" 버튼 클릭

Jenkins 환경설정

JDK 설치

- 시스템 변수 추가

Virable1

Variable name : CLASSPATH

Variable value : %classpath%;.

Virable1

Variable name : JAVA_HOME

Variable value : C:₩Program File₩₩Java₩Java Version

Jenkins 환경설정

JDK 설치

- 유저 변수의 Path 선택 후 편집 버튼 클릭

Jenkins 환경설정

JDK 설치

- "새로 만들기" 버튼 클릭 후 변수값 저장

Variable : %JAVA_HOME%₩bin

Jenkins 환경설정

JDK 설치

- 추가내용 확인 후 "확인" 버튼 클릭하여 설정 저장

Jenkins 환경설정

JDK 설치

- 명령 프롬프트창에서 환경변수 정상 적용되었는지 확인

Select Administrator: Command Prompt

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\isb>java -version
java version "11.0.5" 2019-10-15 LTS
Java(TM) SE Runtime Environment 18.9 (build 11.0.5+10-LTS)
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.5+10-LTS, mixed mode)

C:\Users\isb>javac
Usage: javac <options> <source files>
where possible options include:
  @<filename>                  Read options and filenames from file
  -Akey[=value]                Options to pass to annotation processors
  --add-modules <module>(,<module>)*
        Root modules to resolve in addition to the initial modules, or all modules
        on the module path if <module> is ALL-MODULE-PATH.
  --boot-class-path <path>, -bootclasspath <path>
        Override location of bootstrap class files
  --class-path <path>, -classpath <path>, -cp <path>
        Specify where to find user class files and annotation processors
```

Jenkins 환경설정

Git 설치

- https://gitforwindows.org/ 에서 설치파일 Download

Jenkins 환경설정

Git 설치

- https://gitforwindows.org/ 에서 설치파일 Download
- Default 설정으로 Next> 설치 진행

Jenkins 환경설정

Git 설치

- 설치 완료

Jenkins 환경설정

Gradle 설치

- https://gradle.org/releases/ 에서 최신버전의 설치파일 다운로드

Getting Started Resources

The Gradle team offers free training courses each month.

There are many Gradle tutorials available to help you get started quickly. Source distributions also include many working samples for which guides are not yet written.

v6.0.1

Nov 18, 2019

- Download: binary-only or complete
- User Manual
- API Javadoc
- DSL Reference
- Release Notes

v6.0

Nov 08, 2019

- Download: binary-only or complete
- User Manual
- API Javadoc

Jenkins 환경설정

Gradle 설치

- Download 받은 Gradle 설치파일의 압축을 해제한 후 디렉토리명을 gradle로 변경
- 변경한 디렉토리를 C:₩Program Files 경로에 복사

Jenkins 환경설정

Gradle 설치

- 제어판 -> 시스템 및 보안 -> 시스템 ->

  고급 시스템 설정 -> 고급 -> 환경변수

Jenkins 환경설정

Gradle 설치

- 유저 변수의 Path 선택 후 편집 버튼 클릭

Jenkins 환경설정

JDK 설치

- "새로 만들기" 버튼 클릭 후 변수값 저장

Variable : C:₩Program Files₩gradle₩bin

Jenkins 환경설정

Gradle 설치 확인

- 검색 -> cmd
- 명령 프롬프트 창에서 gradle -v

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\isb>gradle -v

------------------------------------------------------------
Gradle 6.0.1
------------------------------------------------------------

Build time:   2019-11-18 20:25:01 UTC
Revision:     fad121066a68c4701acd362daf4287a7c309a0f5

Kotlin:       1.3.50
Groovy:       2.5.8
Ant:          Apache Ant(TM) version 1.10.7 compiled on September 1 2019
JVM:          11.0.5 (Oracle Corporation 11.0.5+10-LTS)
OS:           Windows Server 2016 10.0 amd64

C:\Users\isb>_
```

Jenkins 환경설정

Jenkins 관리툴 설정

- Jenkins 메뉴에서 Global Tool Configration 클릭

Jenkins 환경설정

Jenkins 관리툴 설정

- JDK installations

 - Name : Default_JDK

 - JAVA_HOME : C:₩Program Files₩Java₩jdk-11.0.5

- Git installations

 - Name : Default_Git

- Path to Git executable : C:₩Program Files₩Git₩bin₩git.exe

- Gradle installations

 - Name : Default_Gradle

 - GRADLE_HOME : C:₩Program Files₩gradle

- 설정 변경 후 저장

Jenkins 환경설정

Jenkins 플러그인 설치

- "Jenkins 관리" 메뉴 클릭

Jenkins 환경설정

Jenkins 플러그인 설치

- "플러그인 관리" 클릭

Jenkins 환경설정

Jenkins 플러그인 설치

- GitHub 검색하여 다음 Plug-In 설치

GitHub Authentication

GitHub Integration

- Plug-In 설치 후 Jenkins 자동 재시작
- 반응 없는 경우 Browser Refresh
- 재시작이 적용되지 않은 경우

(jenkins_url)/safeRestart

- Server Local에서는 윈도우 서비스에서

Jenkins stop & start

Jenkins 환경설정

Jenkins – GitHub Enterprise 연동

- "Jenkins 관리" 메뉴에서 시스템 설정 클릭

Jenkins 환경설정

Jenkins – GitHub Enterprise 연동

- GitHub Enterprise Servers 입력란에 다음 내용 등록

API endpoint : http://githubedu.tangunsoft.com/api/v3

Name : GitHub Enterprise Edu

- 입력 후 저장

GitHub Enterprise Servers

API endpoint	http://githubedu.tangunsoft.com/api/v3
	GitHub Enterprise server verified
Name	GitHub Enterprise Edu

배포서버 구성

SSH 서버 구성

- C:₩ 에 배포 경로로 사용될 test 디렉토리 생성

배포서버 구성

SSH 서버 구성

- http://www.freesshd.com/?ctt=download 에서

  freeFTPd.exe 설치파일 다운로드

배포서버 구성

SSH 서버 구성

- freeFTPd.exe 설치파일 실행하여 설치 진행
- 설치 설정 기본으로 하여 설치 진행

배포서버 구성

SSH 서버 구성

- freeFTPd 설치 완료

배포서버 구성

SSH 서버 구성

- freeFTPd 설치 완료

배포서버 구성

SSH 서버 구성

- 개인키 생성 Yes 클릭

(개인키를 생성하나 시나리오에서는 ID / PW 이용)

배포서버 구성

SSH 서버 구성

- freeFTPd를 시스템 서비스로 구동할 것인지에 문의
- 프로그램 불안정으로 인하여 시스템 서비스에서 정상 작동되지 않으므로 No 클릭

배포서버 구성

SSH 서버 구성

- freeFTPd를 시스템 서비스로 구동할 것인지에 문의
- 프로그램 불안정으로 인하여 시스템 서비스에서 정상 작동되지 않으므로 No 클릭

배포서버 구성

SSH 서버 구성

- 초기 설정 완료
- Finish 버튼 클릭

배포서버 구성

SSH 서버 구성

- FreeFTPd 실행

배포서버 구성

SSH 서버 구성

- 좌측 메뉴란에서 “Users” 클릭
- Users 란에서 “Add” 버튼 클릭

배포서버 구성

SSH 서버 구성

- Login : SSH로 접속할 ID 설정
- Authorization : Password 방식 설정
- Password : SSH로 접속할 PW 설정
- Home Directory : $SERVERROOT₩ 설정
- 모든 설정 입력 후 “Apply” 버튼 클릭

배포서버 구성

SSH 서버 구성

- SFTP Root directory : C:₩ 로 변경
- Server Status : Start로 변경
- Apply & Save 버튼 클릭

배포서버 구성

SSH 서버 구성

- 서버 시스템 검색에서 “Windows Firewall with Advanced Security” 검색
- Inbound Rules 선택
- New Rules 클릭

배포서버 구성

SSH 서버 구성

- Rule Type “Port” 선택

배포서버 구성

SSH 서버 구성

- Allow the connection 선택

배포서버 구성

SSH 서버 구성

- TCP 선택
- Specific local ports 선택
- Port 번호 22 지정

배포서버 구성

SSH 서버 구성

- Domain, Private, Public 선택

배포서버 구성

SSH 서버 구성

- Name : 22_port
- Description : SSH Port
- “Finish” 버튼 클릭하여 설정 저장

배포서버 구성

SSH 서버 구성

- 방화벽 설정 추가 확인

배포서버 구성

SSH Client App 설치

- https://winscp.net/eng/download.php 에 접속하여

  WinSCP 설치파일 다운로드

배포서버 구성

SSH Client App 설치

- WinSCP 설치파일 실행
- Setup Type에서 “Typical Installation” 선택

배포서버 구성

SSH Client App 설치

- User Interface style 에서 Commander 선택

배포서버 구성

SSH Client App 설치

- 설치 진행 및 완료

송출서버 구성

Tomcat 설치

- https://tomcat.apache.org/ 에 접속
- "Tomcat 9" 선택
- Binary Distributions -> Core -> zip 클릭하여 설치파일 Download

송출서버 구성

Tomcat 설치

- 설치파일 압축 해제
- 설치파일 디렉토리명 tomcat으로 변경
- Tomcat 디렉토리 C:₩Program Files 로 이동

송출서버 구성

Tomcat 설치

- C:₩Program Files₩tomcat₩conf 로 이동
- Server.xml 파일을 에디터로 열기

송출서버 구성

Tomcat 설치

- Tomcat Service Port 8080 -> 80 으로 변경
- 변경 후 저장

```
-->
<Connector port="80" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
```

송출서버 구성

Tomcat 설치

- 서버 시스템 검색에서 “Windows Firewall with Advanced Security” 검색
- Inbound Rules 선택
- New Rules 클릭

송출서버 구성

Tomcat 설치

- Rule Type “Port” 선택

송출서버 구성

Tomcat 설치

- Allow the connection 선택

송출서버 구성

Tomcat 설치

- TCP 선택
- Specific local ports 선택
- Port 번호 80 지정

송출서버 구성

Tomcat 설치

- Domain, Private, Public 선택

송출서버 구성

Tomcat 설치

- Name : 80_port
- Description : HTTP Port
- "Finish" 버튼 클릭하여 설정 저장

송출서버 구성

Tomcat 설치

- 방화벽 설정 추가 확인

송출서버 구성

Tomcat 설치

- C:₩Program Files₩tomcat₩bin

  디렉토리로 이동

- Tomat 실행

송출서버 구성

Tomcat 설치

- 브라우저에서 localhost 입력하여

  웹페이지 정상적으로 뜨는지 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 생성

- Jenkins 신규 프로젝트 생성을 위하여 메뉴에서

"새로운 Item" 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 설정

- Jenkins 프로젝트명을 입력
- 프로젝트 타입을 Freestyle Project로 선택
- OK 버튼 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 설정

- 프로젝트의 General 탭에서 설명 입력
- GitHub Project 체크
- Github Project Url 입력

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- GitHub Enterprise의 Sample 프로젝트에 접근
- Clone or download 버튼 클릭
- Url 클립보드 버튼 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- GitHub 계정정보 입력
- 입력후 "Add" 버튼 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- Credentials에서 생성한 GitHub 계정 항목 선택

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- Credentials에서 생성한 GitHub 계정 항목 선택
- 선택 후 Jenkins 프로젝트 저장

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- 프로젝트 메뉴에서 Build Now 실행

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- 빌드 번호 확인
- 빌드 번호 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- 빌드 로그 확인을 위해 Console Output 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- 빌드 로그 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 – GHE 연동

- C:₩Program Files (x86)₩Jenkins₩workspace₩tangunsoft

  경로에 정상적으로 GitHub 소스 내려 받았는지 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 빌드 구성

- 프로젝트 메뉴로 돌아가 "구성" 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 빌드 구성

- Build 항목에서 Invoke Gradle script 선택
- Invoke Gradle 라디오버튼 클릭
- Gradle Version : Default_Gradle 선택
- Tasks : build 입력
- 프로젝트 설정 저장

Jenkins 프로젝트 구성

Jenkins 프로젝트 빌드 구성

- 프로젝트 빌드 실행

Jenkins 프로젝트 구성

Jenkins 프로젝트 빌드 구성

- 해당 빌드의 콘솔 로그 확인
- 빌드 내용 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 빌드 구성

- War 파일이 정상적으로 생성되었는지 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- Jenkins 프로젝트 메뉴의 "구성" 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- 젠킨스 관리 -> 플러그인 관리 -> 설치가능 탭에서 Publish Over SSH 플러그인 선택하여 설치
- 플러그인 설치 후 Jenkins 재시작

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- Jenkins 관리 -> 시스템 설정 으로 이동
- Publish over SSH 항목에서 빌드된 파일 전송할 서버 정보 입력
- “Test Configuration” 클릭하여 Connection 정상인지 확인
- 설정 저장

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- 프로젝트 페이지로 이동하여 프로젝트 메뉴 중

  "구성" 항목 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- 프로젝트 항목 중 빌드 후 조치 항목에서 “Send build artifacts over SSH 선택
- 배포할 SSH 서버 지정
- 배포할 소스파일과 원격서버 배포경로, 실행 커멘드 입력
- 입력 후 설정 저장

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- 프로젝트 빌드 실행
- 빌드 로그 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- 파일 전송 내역 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 배포 구성

- Http://localhost/helloworld-demo 페이지 접근
- 서비스 반영 내역 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- GitHub Enterprise 에서 프로필 -> Settings

  클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Personal settings 에서 Developer settings 선택

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Personal settings 에서 Developer settings 선택

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Note에 토큰명 입력 후

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Token 생성
- 생성된 토큰값을 복사
- 리플래시되는 경우 토큰값이 사라짐

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Jenkins 프로젝트에서 빌드 유발 항목

  "GitHub hook trigger for GITScm polling"

  선택 후 저장

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- GitHub Enterprise Repo에서

Settings Tab -> Hooks -> Add webhook 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Payload URL에 다음 주소를 입력 후 저장

http(s)://{Jenkins Url}/github-webhook/

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Webhooks 설정 저장 후 Status 확인
- Webhooks 상세 Status 를 확인하고자 하는

  경우 WebHooks Url 클릭

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Webhooks 상세 Status 확인

Recent Deliveries

19fa0030-14fc-11ea-862a-1e11eb6a2697 2019-12-02 21:05:58

Request Response 200 Redeliver Completed in 0.11 seconds.

Headers

```
Request URL: http://52.141.37.132:8080/github-webhook/
Request method: POST
content-type: application/x-www-form-urlencoded
Expect:
User-Agent: GitHub-Hookshot/df81e86
X-GitHub-Delivery: 19fa0030-14fc-11ea-862a-1e11eb6a2697
X-GitHub-Enterprise-Host: githubedu.tangunsoft.com
X-GitHub-Enterprise-Version: 2.19.2
X-GitHub-Event: ping
```

Payload


```
{
  "zen": "Mind your words, they are important.",
  "hook_id": 4,
  "hook": {
    "type": "Repository",
    "id": 4,
    "name": "web",
    "active": true,
    "events": [
      "push"
    ],
    "config": {
      "url": "http://52.141.37.132:8080/github-webhook/",
      "content_type": "form",
      "insecure_ssl": "0"
    },
    "updated_at": "2019-12-02T12:05:58Z",
    "created_at": "2019-12-02T12:05:58Z",
    "url": "http://githubedu.tangunsoft.com/api/v3/repos/githubeduadmin/kch_sample/hooks/4"
    "test_url": "http://githubedu.tangunsoft.com/api/v3/repos/githubeduadmin/kch_sample/hoo
    "ping_url": "http://githubedu.tangunsoft.com/api/v3/repos/githubeduadmin/kch_sample/hoo
```

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- GitHub Repo에서 README 파일 생성 후 수정하여 Webhook 유발

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Jenkins Project Build 목록에서 Webhook을 통한 빌드가 실행되는지 확인

Jenkins 프로젝트 구성

Jenkins 프로젝트 webhook 구성

- Jenkins Project 메뉴의 GitHub Hook Log 또는 Build History에서 확인한 빌드 넘버 선택 후 Polling Log에서 WebHook Log 확인

CI/CD에서 DevOps로

DevOps란?

DevOps

- development(개발) + operations(운영) 합성어
- 신속한 고품질 서비스 제공을 통해 비즈니스 가치와 대응력을 향상시키기 위한 문화, 자동화, 플랫폼 설계에 대한 접근 방식
- 개발자와 IT 운영 팀이 밀접하게 협력하여 신뢰성에 지장을 주지 않으면서 소프트웨어 구축, 테스트 및 출시 속도를 높일 수 있습니다

기업들이 DevOps에 투자하는 이유

FORTUNE 선정 100 대 기업 분석결과

2000년 **100대** 기업 중 2018년 100대 기업으로 **선정된 기업은 단 31개 업체**!
2000년 **500대** 기업 중 2018년에 남아있는 기업은 **약 250개** 업체로 **50% 이상이 부도** 또는 인수합병

개발자는 기업 **혁신의 중심**!

✓ **Digital Disruption 이란 ?**

디지털 혁신이 일으키는 변화에 적응하지 못하는 기업들이 사라지는 현상을 일컫는 용어이다.

모든 산업은 디지털화 되어 있고, 디지털화로 변화하는 속도에 따라 경쟁에서 우위를 차지하거나, 뒤처질 수 있다.

기업들이 DevOps에 투자하는 이유

혁신에
더 많은 시간을
투자해야 합니다.

소프트웨어 분야 상위 업체는
혁신에 더 많은 시간을 투자하고
단순 작업에 적은 시간을
소비하고 있습니다.

출처: Source 2018 State of DevOps Report, presented by DORA

기업들이 DevOps에 투자하는 이유

기업들이 DevOps에 투자하는 이유

출처: Source 2017 State of DevOps Report, presented by DORA

DevOps 진입장벽

DevOps 로의 진입 장벽

- 변화에 대한 두려움
- 눈앞의 업무에 집중
- 과거의 실패
- 수직적 기업 구조
- 사업부간의 경쟁의식
- 유지보수 불감증
- 보안에 불감증
- 형상관리 부재

회사 소개

일반현황

회사명	㈜단군소프트
설립일	2002년 1월 21일
직원수	52명
사업분야	컨설팅, 소프트웨어 유통, 기술지원 서비스
소재지	서울 강남구 언주로 620 7층 (논현동, 현대인텔렉스빌딩)
자회사	㈜케이투소프트
해외지사	말레이시아
교육센터	단군소프트 교육장 (15인석)

자회사 현황

회사명	㈜케이투소프트
설립일	2007년 10월 19일
직원수	11명
사업분야	소프트웨어 개발
소재지	서울 강남구 언주로 620 7층 (논현동, 현대인텔렉스빌딩)

단군소프트의 강점(DevOps)

- 단군소프트는 **17년 이상** 해외 소프트웨어 **개발툴과 오픈소스 및 컴포넌트를 취급**한 **노하우**와 고객사를 확보하고 있습니다.
- 다양한 분야의 고객층에 대한 요구사항에 대한 **경험**으로 스타트-업부터 Enterprise 까지 **고객사 환경에 필요한 다양한 분야**의 **툴체인 (Toolchain)을 보유**하고 있습니다.

단군소프트 DevOps Enterprise Toolchain

DevOps

Issue | Prototype & Design | Develop | Version Control | Build & Test | Deploy | Operation | Monitor

Azure DevOps Toolchain

Azure Board | Visual Studio | Azure Repo | Azure Pipelines | Azure Web Apps | Azure VM Scale Set | Azure Application Insights

YT | ProtoPie | IJ | GitHub | TC | Azure Artifacts | CHEF | ZABBIX

Jira | Trello | Sketch | eclipse | Bitbucket | Jenkins | Azure Container Service | puppet | Nagios

* 200개 이상 Vendor와 파트너쉽 체결

Communication

Microsoft Teams | slack

Open Source Governance

sonatype | JFrog ARTIFACTORY

제공서비스 항목

- 단군소프트는 고객에 맞춤형 DevOps 구현을 위한 단계별 고객서비스를 제공합니다.
- DevOps 구현을 위한 오픈소스 활용 시 전문화된 분석을 통한 거버넌스 확보와 통합 및 자동화를 통한 DevOps 최적화를 제공합니다.

현황 파악 컨설팅	•개발자 및 IT 관리자와의 인터뷰를 통한 현황 파악 •Open Source 별 제품 비교 및 관리자 교육을 통한 Know-How 공유
DevOps 설계	•DevOps 단계별 Open Source 부터 다양한 DevOps 솔루션 비교/설계
구축	•요구사항 전반이나 부분적인 DevOps 단계 솔루션 환경 구축
Open Source Integration	• 단계별 DevOps 솔루션 / Open Source 연동을 위한 기술지원
보안	•Open Source의 취약점 등 위험 요소로 부터 지속적인 보호 •위험 요소를 조기 차단
최적화	•Open Source의 품질 확인 •통합 및 자동화를 통한 DevOps 최적화 제공
유지보수	•보안, 라이선스 및 품질 관련 보고서 제공 •지속적인 모니터링을 통해 발생 할 수 있는 이슈 해결

단군소프트 DevOps 컨설팅팀 구성

기술 컨설팅

- 고객사 상황에 맞는 솔루션 가이드 및 관리자 교육
- 맞춤형 Agile 방법론 제시
- 장애 또는 요청 사항에 대한 전담 인력 지원
- 서비스 자동화 및 유지 관리

영업 컨설팅

- 비즈니스 요구 사항에 충족하는 라이선스 컨설팅
- 라이선스 서비스 유지 관리
- 온/오프라인 제품 소개 및 교육 가이드

- 감사합니다 -