

Realize your vision



#{solution_edition}
#{last_modified}

차례

차례	ii
서문	iv
1. 개요	1
1.1 IPA 서버 소개	1
1.2 IPA 서버 구성 요소	2
1.3 권장 사양	2
2. 사전 준비하기	3
3. 서버 설치하기	4
3.1 설치 파일 준비하기	4
3.2 IPA 서버 설치하기	5
서버 APP 패키지 설치하기	5
웹 포털 패키지 설치하기	5
DB 스키마 생성하기	6
설치된 서버 확인하기	7
4. 서비스 관리하기	8
4.1 서비스 시작하기	8
서비스 시작하기	8
각 모듈 서비스 시작하기	9
4.2 서비스 상태 확인하기	9
모든 모듈 서비스 상태 확인하기	9
4.3 서비스 중지하기	9
모든 모듈 서비스 중지하기	9
각 모듈 서비스 중지하기	10
부록 A. tomcat 설치하기	11
부록 B. Java 설치하기	14
부록 C. DBMS 설치하기	15
부록 D. Brity Works 연계 가이드	18
D.1 각 파라미터 정의	18
D.2 프로세스 카탈로그 조회(asset/api/v1/objects/type/process)	19
D.3 프로세스 실행(/scheduler/api/v1/jobs)	20

D.4 프로세스 수행 결과 전달 방식	21
D.5 챗봇과 Brity Works 연계	21

서문

법적 고지 사항

이 매뉴얼을 사용하기 전에 다음 사항을 읽어주십시오.

펴낸 곳	삼성에스디에스 주식회사
주소	서울특별시 송파구 올림픽로 35길 125
대표 전화	+82 2 1661 3388
전자 메일	global.cs@samsung.com
홈페이지	www.samsungsds.com

이 문서에서 다루는 내용은 삼성에스디에스 주식회사가 제공하는 신뢰할 수 있는 정보입니다. 그러나 부정확한 내용이나 오타로 인해 발생하는 문제는 삼성에스디에스 주식회사에서 책임지지 않습니다.

이 문서의 내용과 제품의 사양은 사전 예고 없이 변경될 수 있습니다. 개정에 관한 상세한 정보는 삼성에스디에스 주식회사의 인터넷 홈페이지(www.samsungsds.com)에서 확인할 수 있습니다.

이 문서에 대한 저작권을 포함한 지식재산권은 삼성에스디에스 주식회사에 있습니다. 삼성에스디에스 주식회사의 사전 허가 없이 설명서 내용의 일부 또는 전부를 무단 사용하거나 복제하는 것은 금지되어 있으며, 이는 삼성에스디에스 주식회사의 지식재산권 침해에 해당됩니다.

Copyright © Samsung SDS All rights reserved

사용 대상

이 매뉴얼은 설치 엔지니어를 대상으로 Samsung SDS IPA(이하 IPA) 솔루션의 서버 설치 방법을 설명합니다. 또한, 시스템 운영자를 대상으로 IPA 서버의 실행 및 중지 방법 등을 설명합니다.

설치 엔지니어 및 시스템 운영자가 IPA 서버 운영환경을 효과적으로 구성하기 위해서는 다음과 같은 지식과 경험이 필요합니다.

- 시스템 운영 업무에 대한 기본적인 지식
- 네트워크 설정에 대한 기본적인 지식
- 보안 업무에 대한 기본적인 지식
- 리눅스 서버 사용에 대한 기본적인 지식
- DB 운용 및 조작에 대한 기본적인 지식

매뉴얼 구성

이 매뉴얼은 다음과 같은 내용으로 구성되어 있습니다.

1장. 개요

IPA 서버의 개요 및 설치 환경을 설명합니다.

2장. 사전 준비하기

IPA 서버를 설치하기 전에 필요한 사전 설치 사항을 설명합니다.

3장. 서버 설치하기

셸 스크립트를 사용하여 IPA 서버를 설치하는 방법을 설명합니다.

4장. 서비스 관리하기

IPA 서비스를 시작하거나 중지하고 상태를 확인하는 방법을 설명합니다.

부록 A. tomcat 설치하기

tomcat을 설치하는 방법을 설명합니다.

부록 B. Java 설치하기

Java OpenJDK를 설치하는 방법을 설명합니다.

부록 C. DBMS 설치하기

Maria DB를 설치하는 방법을 설명합니다.

부록 D. Brity Works 연계 가이드

외부 모듈인 Brity Works 연계를 위한 설정 방법을 설명합니다.

표기 규약

문서 내용의 이해를 돕기 위해 다음과 같은 표기 규약을 사용합니다.

볼드체 활자	볼드체 활자 는 그래픽 유저 인터페이스 요소와 메뉴와 디렉토리 등을 표기할 때 사용합니다. 다만 그래픽 유저 인터페이스 중에서 포털, 창, 팝업창, 마법사, 페이지는 볼드체 활자가 아닌 큰따옴표로 표기합니다.
<i>이탤릭체</i>	<i>이탤릭체</i> 는 사용자가 입력해야 하는 변수나 파라미터값을 표기할 때 사용합니다.
고정폭 활자	고정폭 활자는 프로그래밍과 관련된 용어나 코드를 표기할 때 사용합니다. 본문에서 언급할 때는 구분을 위해 명령어, 파라미터명, 레지스트리명, 스크립트명, 프로세스명 등은 고정폭 활자 볼드체 로 표기하고, 변수와 파라미터값은 고정폭 활자 <i>이탤릭체</i> 로 표기합니다.

부연 설명 및 지침

제품을 사용할 때 알아 두면 좋은 팁이나 예외 사항, 제한 사항 등 추가적인 정보를 제공할 때 Note를 사용합니다.



프로젝트를 생성하면 자동으로 빈 프로세스가 생성됩니다.

시스템, 데이터, 단말기 등의 손상 가능성을 알릴 때 Attention을 사용합니다.



속성 화면의 입력값은 실행 시 JavaScript로 변환되므로 JavaScript의 문법에 맞게 입력해야 합니다.

개정 이력

솔루션 버전	매뉴얼 버전	매뉴얼 변경 일자	매뉴얼 변경 사항
1.0	1.0	2018.10.31	최초 제정

솔루션 버전	매뉴얼 버전	매뉴얼 변경 일자	매뉴얼 변경 사항

1. 개요

IPA(Intelligent Process Automation)는 사람이 수행하던 표준화된 업무를 봇이 대신 처리하도록 자동화하는 솔루션입니다.

설치 엔지니어 및 시스템 운영자는 IPA 서버를 설치하기 전 설치 구조를 개략적으로 이해한 후 IPA 서버의 설치 및 관리 방법을 알 수 있습니다.

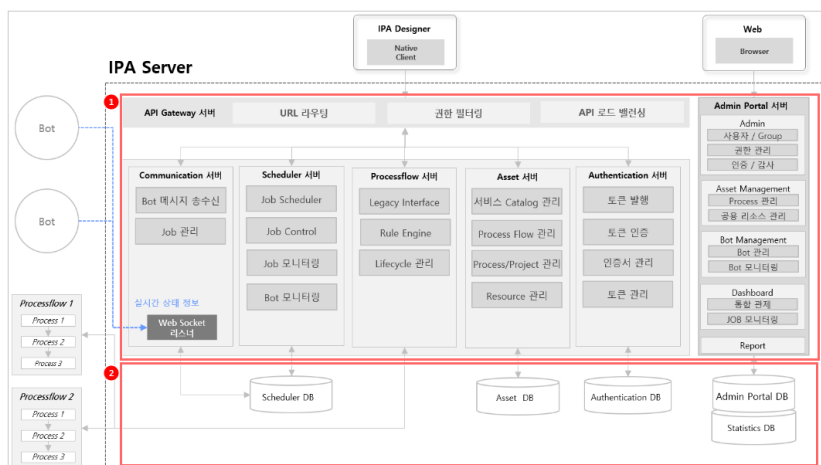
"개요"에서 다루는 내용은 다음과 같습니다.

- IPA 서버 소개
- IPA 서버 구성 요소
- 권장 사양

1.1 IPA 서버 소개

IPA 솔루션은 IPA Designer, IPA Bot, IPA 서버로 구성되어 있으며 그 구조는 다음과 같습니다.

이 매뉴얼에서는 IPA 서버를 설치하는 방법을 설명합니다. IPA 서버는 서버 모듈 영역과 DB 영역으로 구성되어 있습니다.



번호	항목	내용
①	서버 모듈 영역	총 7개 모듈(Administrator Portal, Asset, Authentication, Communication, API Gateway, Scheduler, Processflow)
②	DB 영역	서버 관리 DB(Administrator Portal, Asset, Authentication, Scheduler)

1.2 IPA 서버 구성 요소

IPA 서버를 설치하면 총 7개의 모듈이 설치됩니다. 각 모듈의 역할은 다음과 같습니다.

모듈	모듈명	역할
Admin Portal	admin	사용자, IPA Bot, IPA Designer의 정보를 등록하고 모니터링합니다.
Asset	asset	IPA Designer의 프로젝트를 관리합니다.
Authentication	auth	사용자의 토큰을 발급하고 인증합니다.
Communication	comm	IPA Bot 메시지를 송수신하고 Job의 실행, 중지, 결과 및 상태 메시지를 전달합니다.
API Gateway	gateway	요청한 URL의 경로를 설정하고 허용합니다.
Scheduler	scheduler	Job의 일정을 관리합니다.
Processflow	workflow	프로세스 플로우를 실행합니다.

1.3 권장 사양

IPA 서버를 설치하고 실행하기 위해 권장되는 하드웨어 및 소프트웨어 사양은 다음과 같습니다.

항목	권장 사양
CPU / 메모리 / 저장매체	x86 쿼드코어 프로세서 이상 / 16 GB RAM 이상 / HDD 300 GB 이상
OS	Ubuntu Server 16.04.4 LTS (64 bit)
JRE	Openjre 1.8
DBMS	Maria DB 10.2
Browser	Chrome 58 이상

2.

사전 준비하기

IPA 서버를 설치하기 전 준비해야 할 사항을 설명합니다.

사전에 확인해야 할 사항은 다음과 같습니다.

순서	확인 사항
1	서버 관리자 계정 IPA 서버 설치 시 sudo 권한이 있는 계정으로 설치를 진행해야 합니다. IPA 솔루션은 3개의 계정(ipadm, ipawas, ipadb) 생성을 권장합니다.
2	JRE 설치 여부 IPA 서버가 설치될 OS 환경에 JRE가 설치되어 있어야 합니다. 자세한 내용은 부록 B. Java 설치하기 를 참고하세요.
3	외부 방화벽 접근 규칙 공인망에서 IPA 서버로 8080, 8777, 9001 포트가 허용되어 있어야 합니다.
4	내부 방화벽 접근 규칙 IPA 서버에서 DBMS로 TCP/IP 4406 포트가 허용되어 있어야 합니다.
5	libaio 설치 여부 IPA DB가 설치되는 서버에는 libaio가 설치되어 있어야 합니다. ※ apt 설치 명령어 : apt-get install libaio1 libaio-dev

3.

서버 설치하기

IPA 서버를 설치하고, 시작 및 중단하는 방법을 설명합니다.

"서버 설치하기"에서 다루는 내용은 다음과 같습니다.

- 설치 파일 준비하기
- 서버 설치하기

설치를 위해 다음의 내용을 순서대로 진행합니다.

3.1 설치 파일 준비하기

IPA 서버를 설치하기 위한 방법을 설명합니다. 설치 시 배포되는 파일 목록은 다음과 같습니다.

파일	내용
install.sh	IPA 서버 APP 패키지 설치 스크립트
package.tgz	IPA 서버 APP 패키지 파일
Install_tomcat.sh	IPA 웹 포탈 패키지 설치 스크립트
rpa.war	IPA 웹 포탈 패키지
install_mariadb.sh	IPA Mariadb 설치 스크립트
/conf/createuser.sql	IPA DB 사용자 생성 sql
/conf/initschema.sql	IPA 초기 테이블 스키마 생성 sql
/conf/mysqlld.conf	DB 환경설정 파일
/conf/server.xml	tomcat 설정 파일
/conf/web.xml	tomcat 설정 파일

해당 파일들을 각각 설치할 서버에 업로드한 후 다음 단계로 이동하세요.

※ WAS(tomcat 서버),DBMS(Maria DB)는 바이너리 형태로 다운로드 스크립트와 동일한 위치에 업로드
 apache-tomcat-9.0.14.tar.gz,mariadb-10.3.10-linux-x86_64.tar.gz

3.2 IPA 서버 설치하기

IPA 서버는 다음 절차로 설치를 진행합니다.

서버 APP 패키지 설치하기

IPA 서버 설치 파일(install.sh)을 열어 사용 환경에 맞게 수정하고, 다음의 명령어를 입력하여 서버에 설치하세요.

IPA 서버 설치 파일(install.sh)에서 수정할 수 있는 변수는 다음과 같습니다.

```
[dfpadm@cl-drpaap:/deploy] vi install.sh
#!/bin/sh
#####
IPA_USER=ipaadm //ipa app 설치 사용자
IPA_GROUP=ipaadm //ipa app 설치 사용자 그룹

INSTALL_PATH="/rpa" //ipa app 설치 경로
GATEWAY_IP="127.0.0.1" //ipa 서버 대표 IP app 서버를 분리하지 않는 경우 그대로둠
DB_IP="127.0.0.1" //ipa DB IP db 서버를 분리하지 않는 경우 그대로둠
DB_PORT=4406 //ipa DB 포트
PACKAGE_NAME="package" //패키지 파일 이름

#####
```

웹 포털 패키지 설치하기

IPA 포털 설치 파일(Install_tomcat.sh)을 열어 환경에 맞게 수정한 후 아래의 명령어를 입력하여 설치하세요.

IPA 포털 서버 설치 파일(install_tomcat.sh)에서 수정할 수 있는 변수는 다음과 같습니다.

```
[dfpadm@cl-drpaap:/deploy] vi install_tomcat.sh
#!/bin/sh
#####
WAS_USER=ipawas //ipa 포털 사용자
WAS_GROUP=ipawas //ipa 포털 사용자 그룹
```

```
INSTALL_PATH="/rpa" //ipa 포탈 설치 경로
PACKAGE_NAME=apache-tomcat-9.0.14 //tomcat 패키지 이름
PACKAGE_FILE=apache-tomcat-9.0.14.tar.gz //tomcat 패키지 파일 이름
```

```
#####
```

```
ipa$] ./ Install_tomcat.sh
```

DB 스키마 생성하기

IPA DB 설치 파일(Install_mariadb.sh)을 열어 환경에 맞게 수정한 후 아래의 명령어를 입력하여 설치하세요.

IPA DB 설치 파일(install_mariadb.sh)에서 수정할 수 있는 변수는 다음과 같습니다.

```
[dfpadm@cl-drpaap:/deploy] vi install_mariadb.sh
#!/bin/sh
#####
DB_USER=ipadb //ipa db 사용자
DB_GROUP=ipadb //ipa db 사용자 그룹
DB_PORT=4406 //ipa db 포트

INSTALL_PATH="/rpa" //ipa db 설치 경로
PACKAGE_NAME=mariadb-10.3.10-linux-x86_64 //mariadb 패키지 이름
PACKAGE_FILE=mariadb-10.3.10-linux-x86_64.tar.gz //mariadb 패키지 파일 이름

DB_RPA_USERNAME=rpa
DB_AUTH_USERNAME=auth
DB_ASSET_USERNAME=asset
DB_ADMIN_USERNAME=admin
DB_SCHEDULER_USERNAME=scheduler

DB_RPA_PASSWORD="rpa!23"
DB_AUTH_PASSWORD="auth!23"
DB_ASSET_PASSWORD="asset!23"
DB_ADMIN_PASSWORD="admin!23"
DB_SCHEDULER_PASSWORD="scheduler!23"

DB_ROOT_PASSWORD='rpago!23'
```

```
#####
```

```
ipa$] ./install_mariadb.sh
```



IPA 서버는 5개(asset, auth, rpa, stat, scheduler)의 스키마를 사용합니다. 동일한 이름의 스키마가 mariaDB에 존재하는 경우 다른 DB 서버에 설치해야 합니다.

설치된 서버 확인하기

설치가 완료되면 /rpa 디렉토리로 이동하여 관련 파일들이 정상적으로 생성되었는지 확인하세요.

설치된 서버의 디렉토리 정보는 다음과 같습니다.

디렉토리명	설명
/apps	Asset, Authentication, Communication, Gateway, Scheduler, Processflow 등 IPA 서버가 설치된 디렉토리입니다.
/certificate	SSL 인증서, IPA 서버 인증서가 설치된 디렉토리입니다.
/logs	IPA 서버, Admin Portal의 로그가 설치되는 디렉토리입니다.
/pkgs	WAS와 DB 서버가 설치되는 디렉토리 입니다.

4.

서비스 관리하기

설치 완료 후 IPA 서비스를 이용해 서버를 관리하는 방법을 설명합니다.

시스템 운영자는 서비스를 시작 및 중지하고 상태를 확인하며 IPA 서버를 더욱 효율적으로 운영할 수 있습니다.

"서비스 관리하기"에서 다루는 내용은 다음과 같습니다.

- 서비스 시작하기
- 서비스 상태 확인하기
- 서비스 중지하기

4.1 서비스 시작하기

IPA 서버 관리 방법을 설명합니다. IPA 서버의 개별 서비스는 모듈로 구성되어 있습니다.

관리 명령어는 모듈 이름을 지정하여 실행할 수 있습니다.

개별 모듈 이름은 [1.2 IPA 서버 구성 요소](#)를 참고하세요.

서비스 시작하기

모든 IPA 서버 모듈을 실행하려면 다음 명령어를 입력하세요.

```
ipa$] /rpa/bin/run.sh //전체 서비스 시작
```

각 모듈 서비스 시작하기

IPA 서버 모듈을 각각 실행하려면 개별 모듈 이름을 입력하세요.

```
ipa$] /rpa/bin/run.sh 모듈 이름
```

4.2 서비스 상태 확인하기

서비스를 실행한 후 명령어를 입력해 IPA의 서비스 상태를 확인할 수 있습니다.

모든 모듈 서비스 상태 확인하기

모든 IPA 서버 모듈의 상태를 한번에 확인하려면 다음 명령어를 입력하세요.

```
~$ /rpa/bin/log.sh 모듈이름
```

4.3 서비스 중지하기

서비스를 실행한 후 명령어를 입력해 IPA 서비스를 중지할 수 있습니다.

모든 모듈 서비스 중지하기

모든 IPA 서버 모듈 서비스를 중지하려면 다음 명령어를 입력하세요.

```
~$ /rpa/bin/stop.sh
```


각 모듈 서비스 중지하기

IPA 서버 모듈 각각을 중지하려면 다음 개별 명령어를 입력하세요.

```
~$ /rpa/bin/stop.sh 모듈 이름
```

부록 A.

tomcat 설치하기

IPA admin서버를 구동하기 위해 설치되는 서버에 WAS(tomcat) 서버를 설치합니다.

다음은 tomcat 8.5 버전을 설치할 때의 예시입니다.

1. 설치 파일 `apache-tomcat-8.5.33.tar.gz`을 `/ipa/pkgs/tomcat/` 디렉토리에 압축을 해제하고, 아래의 명령어를 입력하여 소유자(owner) 를 tomcat 전용 계정(ipawas)으로 변경하세요.

```
~$ cd /ipa/pkgs
~$ tar xvfz ~/apache-tomcat-8.5.33.tar.gz
~$ mv apache-tomcat-8.5.33 tomcat
~$ chown -R ipawas tomcat
```

2. 설정 파일 `/ipa/pkgs/tomcat/conf/server.xml`을 다음과 같이 수정하세요.

```
<Connector port="8080" scheme="https" secure="true" SSLEnabled="true"
    keystoreFile="/ipa/certification/RPA_ADMIN_SERVER.jks"
    server="Null" keystorePass="!@#RPA_ADMIN_SERVER$%^"
    useServerCipherSuitesOrder="true" sslProtocol="TLS"
    sslEnabledProtocols="TLSv1,TLSv1.1,TLSv1.2" clientAuth="false"
    ciphers="TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
    TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,
    TLS_DHE_RSA_WITH_AES_128_GCM_SHA256,
    TLS_DHE_RSA_WITH_AES_256_GCM_SHA384,
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
    TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
    TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
    TLS_DHE_RSA_WITH_AES_128_CBC_SHA256,
    TLS_DHE_RSA_WITH_AES_128_CBC_SHA,
    TLS_DHE_RSA_WITH_AES_256_CBC_SHA256,
    TLS_DHE_RSA_WITH_AES_256_CBC_SHA,
```

```
TLS_RSA_WITH_AES_128_GCM_SHA256,
TLS_RSA_WITH_AES_256_GCM_SHA384,
TLS_RSA_WITH_AES_128_CBC_SHA256,
TLS_RSA_WITH_AES_256_CBC_SHA256,
TLS_RSA_WITH_AES_128_CBC_SHA,
TLS_RSA_WITH_AES_256_CBC_SHA,
SSL_RSA_WITH_3DES_EDE_CBC_SHA" />
```

3. 설정 파일 /ipa/pkg/tomcat/conf/server.xml의 deploy director와 access log directory를 다음과 같이 설정하세요.

```
<Engine name="Catalina" defaultHost="admin">
  <Host name="admin" appBase="/ipa/apps/admin" unpackWARs="true" autoDeploy="true">
    <Context path="" docBase="." reloadable="true"/>
    <Valve className="org.apache.catalina.valves.AccessLogValve"
      directory="/ipa/logs/admin"
      prefix="admin_access_log" suffix=".txt"
      pattern="%h %l %u %t &quot;%r&quot; %s %b" />
```

4. /ipa/pkg/tomcat/conf/logging.properties 설정의 로그 디렉토리 위치를 다음과 같이 변경하세요.

```
#####
# Handler specific properties.
# Describes specific configuration info for Handlers.
#####
1catalina.org.apache.juli.AsyncFileHandler.level = FINE
#1catalina.org.apache.juli.AsyncFileHandler.directory = ${catalina.base}/logs
1catalina.org.apache.juli.AsyncFileHandler.directory = /ipa/logs/admin
1catalina.org.apache.juli.AsyncFileHandler.prefix = catalina.
1catalina.org.apache.juli.AsyncFileHandler.maxDays = 90

2localhost.org.apache.juli.AsyncFileHandler.level = FINE
2localhost.org.apache.juli.AsyncFileHandler.directory = /ipa/logs/admin
2localhost.org.apache.juli.AsyncFileHandler.prefix = localhost.
2localhost.org.apache.juli.AsyncFileHandler.maxDays = 90

3manager.org.apache.juli.AsyncFileHandler.level = FINE
3manager.org.apache.juli.AsyncFileHandler.directory = /ipa/logs/admin
3manager.org.apache.juli.AsyncFileHandler.prefix = manager.
3manager.org.apache.juli.AsyncFileHandler.maxDays = 90
```

```
4host-manager.org.apache.juli.AsyncFileHandler.level = FINE
4host-manager.org.apache.juli.AsyncFileHandler.directory = /ipa/logs/admin
4host-manager.org.apache.juli.AsyncFileHandler.prefix = host-manager.
4host-manager.org.apache.juli.AsyncFileHandler.maxDays = 90
```

5. /ipa/pkgs/tomcat/bin/catalina.sh를 다음과 같이 변경하여 catalina.out 파일의 위치를 지정하세요.

```
if [ -z "$CATALINA_OUT" ] ; then
# CATALINA_OUT="$CATALINA_BASE"/logs/catalina.out
CATALINA_OUT=/rpa/logs/admin/catalina.out
fi
```

6. 아래의 명령어를 입력하여 /ipa/pkgs/tomcat/bin/setenv.sh를 다음과 같이 변경하세요.

```
ipa$]cat setenv.sh
export JAVA_OPTS="$JAVA_OPTS -Dspring.profiles.active=dev
```

부록 B.

Java 설치하기

IPA 서버를 구동하기 위해 설치되는 서버에 OpenJDK 1.8 버전을 설치합니다.
다음은 OpenJDK 1.8 버전을 설치할 때의 예시입니다.

1. 아래 명령어를 입력하여 OpenJDK 1.8을 설치하세요.

```
~$ sudo apt-get install openjdk-8-jre
```

2. 아래 명령어를 입력하여 기본 Java 환경을 확인하고 기본 명령어로 등록하세요

```
~$ update-alternatives --config java
```

선택	경로	우선순위	상태
0	/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java	1081	자동 모드
*1	/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java	1081	수동 모드
2	/usr/lib/jvm/java-8-oracle/jre/bin/java	1081	수동 모드

Press <enter> to keep the current choice[*], or type selection number: 1

3. 아래 명령어를 입력하여 Java가 정상적으로 설정되었는지 확인하세요.

```
~$ java -version
```

부록 C.

DBMS 설치하기

IPA 솔루션은 Maria DB 10.2 버전을 지원합니다. 다음은 mariadb 설치 예시입니다.

1. mariadb를 설치하기 위해서는 libaio가 필요합니다. 아래의 명령어를 입력하여 libaio를 설치하세요.

```
~$ apt-get install libaio1 libaio-dev
```

2. mariadb 파일을 /ipa/download 디렉토리에 다운로드한 후 압축을 해제하세요.

```
~$ tar xvfz /ipa/download/mariadb-10.2.13-linux-x86_64.tar.gz
```

3. 압축을 해제한 파일을 /ipa/pkgs/mariadb 디렉토리로 옮긴 후 ipadb 계정에 권한을 부여하세요.

```
~$ mv /ipa/download/mariadb-10.2.13-linux-x86_64 /ipa/pkgs/mariadb
~$ sudo chown -R ipadb /ipa/pkgs/mariadb
```

4. mariadb의 log가 저장될 디렉토리를 생성한 후 ipadb 계정에 권한을 부여하세요.

```
~$ mkdir /ipa/logs/mariadb
~$ sudo chown 0+rx ipadb /ipa/logs/mariadb
```

5. mariadb 실행에 필요한 기본적인 사용 환경을 설정하세요.

```
~$ mkdir /ipa/logs/mariadb/conf
~$ sudo chown ipadb /ipa/logs/mariadb/conf
~$ cp ~/mysql.conf /ipa/logs/mariadb/conf/
~$ vi /ipa/logs/mariadb/conf/
/ipa/logs/mariadb/conf/mysld.conf
port = 4406
default-character-set = utf8
```

```
socket = /rpa/pkgs/mariadb/conf/mysql.sock

[mysqld]
datadir = /rpa/pkgs/mariadb/data
basedir = /rpa/pkgs/mariadb
explicit_defaults_for_timestamp = 1
socket = /rpa/pkgs/mariadb/conf/mysql.sock

plugin_load=server_audit=server_audit.so;simple_password_check=simple_password_check.so
server_audit_file_path = /ipa/logs/mariadb/server_audit.log
server_audit_file_rotate_size = 104857600
server_audit_events = CONNECT
server_audit_logging = ON

max_connections = 500

init_connect = SET collation_connection = utf8_general_ci
init_connect = SET NAMES utf8
character-set-server = utf8
collation-server = utf8_general_ci
lower_case_table_names = 0

user = ipadb
port = 4406
pid-file = /ipa/pkgs/mariadb/conf/mysql.pid
log-output=FILE

#general-log=1
#general_log_file=/rpa/logs/mariadb/mariadb.log
slow-query-log=1
slow_query_log_file=/rpa/logs/mariadb/mariadb-slow.log
long_query_time=30
log-error=/rpa/logs/mariadb/mariadb.err

## Time Zone Setting after install
#default-time-zone='UTC'

[mysqldump]
default-character-set = utf8
socket = /rpa/pkgs/mariadb/conf/mysql.sock
```

```
[mysql]
```

6. mysql_install_db 스크립트를 실행해 데이터 디렉토리를 생성하세요.

```
~$ /ipa/pkgs/mariadb/scripts/mysql_install_db
-- defaults-file=/ipa/pkgs/mariadb/conf/mysqlld.conf
//mariadb daemon을 background로 실행
~$ /ipa/pkgs/mariadb/bin/mysqlld_safe - defaults-file=/ipa/pkgs/mariadb/conf/mysqlld.conf&
~$ su
~$ /ipa/pkgs/mariadb/bin/mysqladmin password [ROOT 패스워드] -S
/ipa/pkgs/mariadb/conf/mysqlld.sock
```


부록 D.

Brity Works 연계 가이드

Brity Works는 단일 Open API 규격을 통해 여러 다른 기종 시스템들과 직접 연동할 수 있습니다.

- 연동 포트
클라우드 서비스 이용 시 방화벽 설정의 문제로 RPA API Gateway 서버 포트만 사용합니다.
- 전송 방법
클라이언트는 HTTP 및 HTTPS를 GET 방식 또는 POST 방식을 통해 호출합니다.
- 외부 연동 시 보안 및 인증 규격
RPA Open API는 인증 정보 및 권한을 확인합니다. 호출자의 정보는 사전에 등록되어 있어야 합니다.
 - 서버 연계: 서버용 인증서를 이용하여 RPA 시스템에서 토큰을 발급받습니다.
 - 클라이언트 연계: RPA 시스템에서 발급받은 라이선스를 이용하여 토큰을 발급받고 인증을 진행합니다.



Java 사용 시 서버 인증에 필요한 토큰을 발급해주는 라이브러리는 별도로 제공됩니다.

D.1 각 파라미터 정의

프로세스를 시작하는 API의 각 파라미터 정의는 다음과 같습니다.

- **name**: 이름을 뜻하며 파라미터를 식별해 주는 ID로도 사용됩니다.
- **expression**: 챗봇 등에서 문장을 작성할 때 사용할 수 있는 한글 명칭입니다.
- **description**: 파라미터에 대한 설명입니다.
- **optional**: "yes" 또는 "no"의 값을 가지며 *optional* 혹은 *mandatory* 여부를 지정합니다.
- **type**: JavaScript의 기준으로 "*Boolean*", "*Number*", "*String*" 등이 있습니다.

D.2 프로세스

카탈로그

조회(asset/api/v1/objects/type/process)

외부 시스템에서 Brity Works에 등록된 자동화 프로세스 목록을 조회하고, 프로세스의 Input 파라미터 및 상세 정보를 조회할 수 있습니다.

다음은 카탈로그를 조회하는 API의 예시입니다.

기능	API	방식	요청사항	Input 파라미터	Output 파라미터
RPA 프로세스 카탈로그 조회	/asset/api/v1/objects/type/process	GET	process 타입의 리스트를 조회합니다.	없음	[{"filePath": "/a.dll", "name": "testAsset1", "typeName": "Asset", "id": 29748728473856, "type": 6, "category": "dll", "version": "1.00", "desc": "축하 메일 보내기", "inputs": [{"name": "sender", "expression": "발신인", "desc": "이메일을 보내는 사람", "optional": "no", "type": "String"}, {"name": "receiver", "expression": "수신인", "desc": "이메일을 받는 사람", "optional": "no", "type": "String"}, {"name": "content", "expression": "본문", "desc": "메일 본문", "optional": "no", "type": "String"}], "outputs": [{"name": "result", "expression": "발신 결과", "desc": "이메일 발신 성공 여부", "optional": "no", "type": "Boolean"}]},]
RPA 프로세스 카탈로그 조회(특정 시점 이후 변경 내용)	/asset/api/v1/objects/type/process/changed?datetime=[Datetime String]	GET	process 타입의 리스트를 조회합니다.	datetime=2018-04-30T17%3A20%3A30%2B09%3A00 Format : YYYY-MM-DDThh:mm:ssTZD	[{"outputs": "[]", "updateDatetime": 1525075163000, "inputs": "[]", "filePath": "park_3.proc",

기능	API	방식	요청사항	Input 파라미터	Output 파라미터
				(URL Encoding)	<pre> "typeName": "Process", "project": { "name": "Test_Ver1_0b", "id": "afec5113-d892-46d6-aae1-9cfad48178c9" }, "type": 2, "version": "1.80", "delYn": "Y", "isChanged": "Y", "name": "park_3", "id": "02f5a6ab-46bd-47ae-952e-2b11aa2f2499", "activation": "Y", "category": "Process", "desc": "" }] </pre>

D.3 프로세스 실행(/scheduler/api/v1/jobs)

카탈로그 조회를 통해 식별된 자동화 프로세스를 원격 봇이 수행해줍니다. 식별된 프로세스의 Input 파라미터에 따라 필요한 값을 입력하고 RPA 자동화 프로세스 실행을 요청합니다. 업무 수행 요청 후 결과는 비동기로 전송됩니다.

수행 요청의 결과 및 규격은 다음과 같습니다.

기능	API	방식	요청사항	Input 파라미터	Output 파라미터
RPA 프로세스 시작	/scheduler/api/v1/jobs	POST	scheduler 서버에 RPA 프로세스를 즉시 시작하도록 요청합니다.	<pre> { "jobType": "1", "processId": "29748728473856", "priority": "0", "userAuthentication": "{Brity Eco Token}", "jobParameter": { "sender": "joonwon7.lee@samsung.com, </pre>	<pre> [{"jobId": "1520835859498-0f3bcef2-58de7a93", "resultCode": "SUCCESS" }] </pre>

기능	API	방식	요청사항	Input 파라미터	Output 파라미터
				<pre>"receiver" : sh.sun@samsung.com, "content" : "메일 본문 내용..." }" }</pre>	

D.4 프로세스 수행 결과 전달 방식

RPA 자동화 프로세스의 수행 결과는 아래 3가지 방식으로 전달할 수 있습니다.

- 폴링(Polling): RPA 프로세스 결과 조회 API를 이용하여 주기적으로 결과를 전달하는 방식입니다.
- 콜백(Callback): 외부 시스템에서 RPA 프로세스 실행 요청 시 결과 통지를 받을 URL을 함께 전달하여 결과를 전달하는 방식입니다.
- Custom: 외부 시스템의 변경이 어려운 경우 RPA 서버에서 직접 API를 호출하여 결과를 전달하는 방식입니다. 챗봇 연계 시 이 방식을 이용합니다.



- 프로세스 시작 시 생성된 jobID는 이후에도 계속 사용되며, 프로세스 결과를 전달받는 챗봇의 API를 호출할 때 같이 전달됩니다.
- 프로세스 실행이 끝난 후 생성된 결과값은 카탈로그에 정의된 양식에 따라 챗봇에게 전달됩니다.

D.5 챗봇과 Brity Works 연계

챗봇의 사용자 대화 처리 기능을 이용해 사용자가 원하는 업무 내용을 분석합니다. 또한 RPA의 업무 자동화 기능을 이용해 AI 로봇틱스 프로세스 환경을 구축할 수 있습니다.

순서	단계	내용	연계 API
1	데이터 수집	RPA 프로세스 카탈로그 조회를 통해 대화 생성 및 RPA 프로세스를 계획합니다.	프로세스 카탈로그 조회 API(/asset/api/v1/objects/type/process)
2	파라미터 수집	챗봇 대화 진행을 통해 RPA 프로세스 실행에 필요한 Input	

순서	단계	내용	연계 API
		파라미터를 수집합니다.	
3	자동화 프로세스 실행	RPA 자동화 프로세스 시작을 요청합니다.	프로세스 실행(/scheduler/api/v1/jobs)
4	자동화 프로세스 결과 전달	RPA로부터 자동화 프로세스 수행 결과를 전달합니다.	프로세스 실행 결과를 Custom 방식으로 전달