



ACCESSI DYS

AccessiDys

Configuration and Deployment Guide on AccessiDys servers

Version 1.1



Summary

1	INTRODUCTION	3
1.1	AIM OF THE DOCUMENT	3
1.2	REFERENCE DOCUMENTATION.....	3
1.3	GLOSSARY	3
2	CONFIGURATION MANAGEMENT	4
2.1	STRUCTURE.....	4
2.2	PROCESS	4
3	DEPLOYMENT ON ACCESSIDYS SERVERS.....	5
3.1	ENVIRONMENTS	5
3.2	ACCESS	5
3.3	DEPLOYMENT PROCESS.....	5
3.3.1	<i>Pre-requisite.....</i>	<i>5</i>
3.3.2	<i>Connect to server with cloud user and switch to node user :</i>	<i>5</i>
3.3.3	<i>Stop application.....</i>	<i>6</i>
3.3.4	<i>Update of version :</i>	<i>6</i>
3.3.5	<i>Build updated release.....</i>	<i>6</i>
3.3.6	<i>Start application</i>	<i>6</i>
3.4	KNOWN ISSUES	7



1 INTRODUCTION

1.1 AIM OF THE DOCUMENT

The document describes how is managed the configuration on Accessidys Project and how deploy a new release on the ACCESSIDYS servers.

1.2 REFERENCE DOCUMENTATION

N°	Version	Date	Document title	Detail
1	V1.0	18/05/2016	ACCESSIDYS_DAT_ENG_V1.0	Technical architecture design

Table 1-1 : reference documents

1.3 GLOSSARY

Acronyms	Definition

Table 1-2 : Acronyms description



2 CONFIGURATION MANAGEMENT

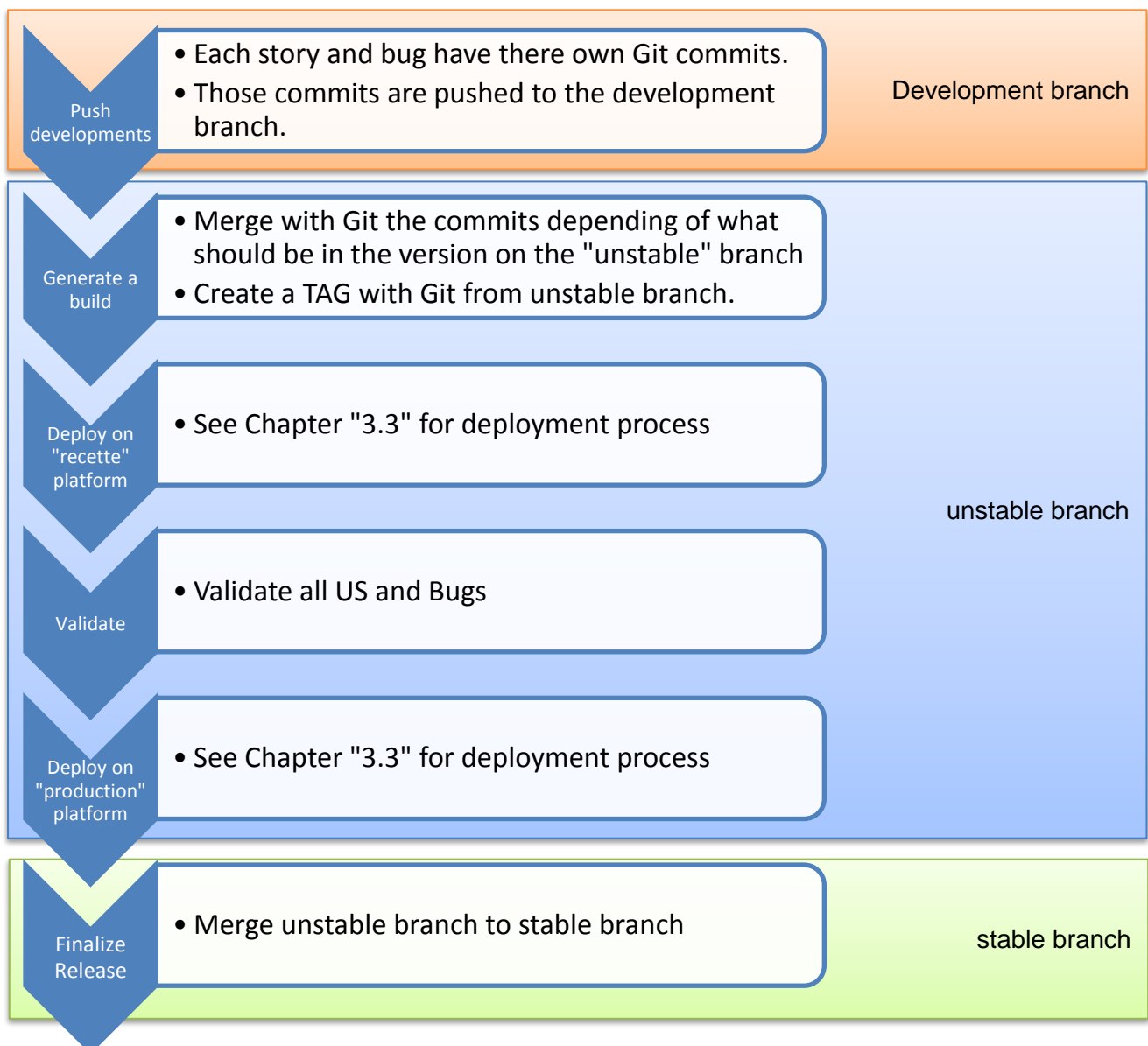
2.1 STRUCTURE

"Git" tool is used for the configuration management.

The structure of Git repository is based on three branches :

1. **Development** : All new developments are saved in this branch
2. **Unstable** : used to generate a build for validation with a list of selected developments
3. **Stable** : used to save the last release deployed on production

2.2 PROCESS





3 DEPLOYMENT ON ACCESSIDYS SERVERS

3.1 ENVIRONMENTS

ACCESSIDYS environments are :

- “**Recette**” : used to qualify the version to be installed in production.

Hostname : `recette.accessidys.com`

Unix username : `cloud`

Url accessidys website : <https://recette.accessidys.com>

- “**Production**” : is the current server for Accessidys production

Hostname : `accessidys.org`

Unix username : `cloud`

Url accessidys website : <https://accessidys.org>

3.2 ACCESS

To be able to access ACCESSIDYS environments the user need a SSH key.

All information about ssh key generation and installation are available here :

<https://support.rackspace.com/how-to/generating-rsa-keys-with-ssh-puttygen/>

3.3 DEPLOYMENT PROCESS

3.3.1 Pre-requisite

- Access to the machine (See § 3.2 Access)
- Tag on Github (See § 2.2 Process)
 - o Get the name of the tag from that page :
<https://github.com/AccessiDys/AccessiDys/tags>

3.3.2 Connect to server with cloud user and switch to node user :

```
sudo su
su node
```

Note : “cloud” is the exploitation user and “node” is the application user.



3.3.3 Stop application

Connected as “node”, execute:

```
ps -aef | grep grunt
#Résult example :  node      14530      1  4 Jul13 ?      16:35:18 grunt
#14530 is the PID
#Stop of the process (remplace <PID> with given value):
kill <PID>
```

3.3.4 Update of version :

Connected as “node”, execute:

```
cd ../node/accessidys/
#(remplace <NAME OF TAG RELEASE> with tag name requested as pre-requisite)
git pull https://github.com/AccessiDys/AccessiDys.git refs/tags/<NAME OF TAG RELEASE>
```

Merge request is execute.

Type « CTRL + x » to save and continue

If issues appears during pull request see 3.4 Known issues

Update user owner for all files :

```
exit
cd ../node/accessidys/
chown -R node:node .
su - node
cd /home/node/accessidys
```

3.3.5 Build updated release

Connected as “node”, execute:

Only for “**Recette**” platform

```
grunt build-recette
cd dist/
npm install
```

Only for “**Production**” platform

```
grunt build-prod
cd dist/
npm install
```

3.3.6 Start application

Connected as “node”, execute:



```
nohup grunt server >/dev/null 2>&1 &
```

3.4 KNOWN ISSUES

Some issues during deployment has been seen mainly regarding line endings that are different between DOS and Unix systems.

All developments are done on a Windows OS and deployments are done on a Unix systems.

You can find below the link to manage line endings issues :

<https://help.github.com/articles/dealing-with-line-endings/>