

Appzillon CI Pipeline

Appzillon build url:<http://192.168.1.45:8080/job/AppzillonPipeLine-Test/>

Appzillon Testing url:<http://192.168.1.45:8080/job/Appzillon-RegressionTesting/>

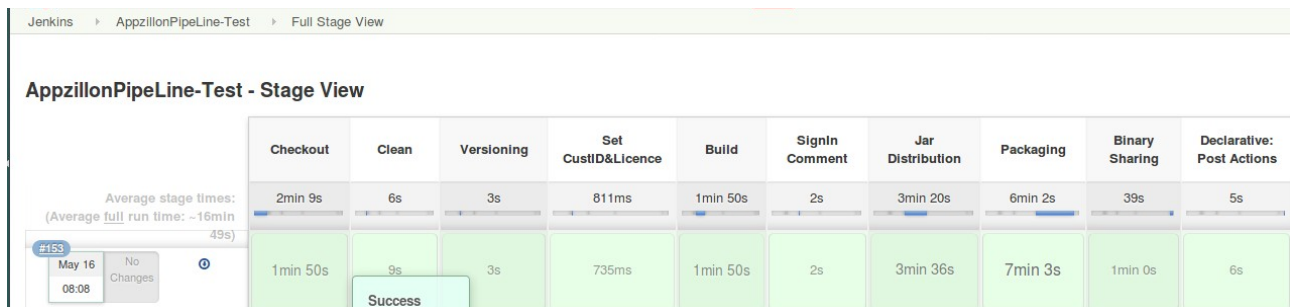
Reg App Url:<http://192.168.1.6:8081/Testing/>

Mac Server:192.168.1.206

Window Server:192.168.1.45

Linux Sever: 192.168.1.6

Binary Sharing Today's Path:\\192.168.1.6\\appzillontesting\\3.6.0\\20180516-08-08-35



```
def defaultcotpath=""
def sysdate =""
def nowdate=new Date().format( 'yyyyMMdd-HH-mm-ss' )
def pass ="p@ssw0rd"
def key ="ab:a9:b8:34:48:0b:a5:0d:e3:56:ae:5e:41:2f:41:9b"
def buildnumber = env.BUILD_NUMBER
def Macpass ="madmac"
def Mackey ="f8:bc:5c:a6:30:77:0b:61:c8:68:e7:d7:5a:25:ce:08"
//Function Notify Email
def buildstatus="Success":
```

Declaring global variables which use accross the pipeline

Email Notification Function -Start

```
def emailNotification(buildstatus,sysdate,relnum){
def emailBody="";
def recipientlist="";
def binaryPath="\\'+\\192.168.1.6\\appzillontesting\\'+\"${relnum}\"+'\\'+\"${sysdate}\""
if("${buildstatus}"=="Started"){
recipientlist="preetish.das@i-exceed.com,darshanl.sagar@i-exceed.com,shyam.kumar@i-exceed.com,amar.khan@i-exceed.com,cc:chandu.balaji@i-exceed.com"
emailBody= ""
<p>Hi All,
</p>
<p>AppzillonStudio-3.6.0 build triggered.
<br>Please find the latest build details below.
</p>
<p>Check console output at
<a href='${env.BUILD_URL}'>${env.JOB_NAME}[${env.BUILD_NUMBER}]
</a>
</p>
<p>Regards,
<br>Appzillon DevTeam
```

```

    <br>I-Exceed Technology Solutions
  </p>""
}
else if("${buildstatus}"=='Success'){
  recipientlist="preetish.das@i-exceed.com,darshanl.sagar@i-exceed.com,shyam.kumar@i-
  exceed.com,amar.khan@i-exceed.com,cc:chandu.balaji@i-exceed.com,cc:nagaraj.hiremath@i-
  exceed.com,cc:arthanarisamy@i-exceed.com,cc:prabudas.s@i-exceed.com"
  emailBody= ""
  <p>Hi All,
  </p>
  <p>Please find the latest build details below.
  <ul>
    <ui>
      <b>Path: ${binaryPath}
    </b>
    </ui>
  </ul>
  </p>
  <p>Regards,
    <br>Appzillon DevTeam
    <br>I-Exceed Technology Solutions
  </p>""
}
else{
  recipientlist="preetish.das@i-exceed.com,darshanl.sagar@i-exceed.com,shyam.kumar@i-
  exceed.com,amar.khan@i-exceed.com,cc:chandu.balaji@i-exceed.com"
  emailBody= ""
  <p style="color:red;">Hi All,
    <br>
    <br> AppzillonStudio-3.6.0 build ${buildstatus}: Job ${env.JOB_NAME} [${
  env.BUILD_NUMBER}]
    <br>
    <br> Check console output at
    <a href='${env.BUILD_URL}'>${env.JOB_NAME} [${env.BUILD_NUMBER}]
    </a>
    <br>
    <br>Regards,
    <br>
    <br>Appzillon IDE Team
    <br>I-Exceed Technology Solutions
  </p>""
}
emailtext (
  subject: "Appzillon-3.6.0.S1 Build ${buildstatus}",
  mimeType: 'text/html',
  body: "${emailBody}",
  to: "${recipientlist}",
  replyTo: "jenkinsadmin@i-exceed.com",
  from: 'jenkinsadmin@i-exceed.com',
  recipientProviders: [[class: 'CulpritsRecipientProvider']]
)
}

```

*Email Notification which
will defer in case of job
status*

1.Success
2.Fail
3.Start

Declarative Pipeline

```
pipeline {
  agent any
  triggers{cron('H 8 * * *')} -----//Setting job at 8oclock everyday morning
```

```
parameters {
  string(name: 'CheckoutPath', defaultValue: 'Change Default Path', description: 'Path for Appzillon
  source code checkout!!', trim: true)
  string(name: 'AppzillonVersion', defaultValue: '3.6.0', description: 'Appzillon Version for
  Release!!',trim: true)
  choice(choices: 'APZCUSTOMERID\nBARCLAYS\nIPPB', description: 'Select Appzillon
  Customer!!', name: 'CustId')
  // choice(choices: 'US-EAST-1\nUS-WEST-2', description: 'What AWS region?', name: 'region')
  booleanParam(
  defaultValue: false,
  description: 'For 32 bit Systems',
  name: '32 bit'
  )
  booleanParam(
  defaultValue: false,
  description: 'For 64 bit Systems',
  name: '64 bit'
  )
  booleanParam(
  defaultValue: false,
  description: 'Licence required ?',
  name: 'Licence'
  )
}
```

Setting user input parameters which could be text,checkbox,radiobutton,dropdown so user can give paramters to job while they triggres it shown in below screenshot.

Pipeline AppzillonPipeLine-Test

This build requires parameters:

CheckoutPath	<input type="text" value="Change Default Path"/>
	<small>Path for Appzillon source code checkout!!</small>
AppzillonVersion	<input type="text" value="3.6.0"/>
	<small>Appzillon Version for Release!!</small>
CustId	<div>APZCUSTOMERID ▾</div>
	<small>Select Appzillon Customer!!</small>
32 bit	<input type="checkbox"/>
	<small>For 32 bit Systems</small>
64 bit	<input type="checkbox"/>
	<small>For 64 bit Systems</small>
Licence	<input type="checkbox"/>
	<small>Licence required ?</small>
<div>Build</div>	

Setting up customer details from dropdown

Versioning for customer

```

stages {
stage('Checkout') {
  steps {
    withCredentials([string(credentialsId: 'svn-credentials-id', variable: 'TOKEN')]) {
      script {
        //def buildnumber = env.BUILD_NUMBER
        emailNotification("Started", "${sysdate}", "${params.AppzillonVersion}")
        echo "${buildnumber}"
        if("${params.CheckoutPath}"=='Change Default Path'){
          defaulcotpath="${buildnumber}"
          // bat 'svn export --username amar.khan http://192.168.1.1/svn/Appzillon/3.6.0.S1/Studio '+'${buildnumber}'+ --password '+'${TOKEN}"
        }else
        {
          defaulcotpath="${params.CheckoutPath}"
          // bat 'svn export --username amar.khan http://192.168.1.1/svn/Appzillon/3.6.0.S1/Studio '+'${params.CheckoutPath}'+ --password '+'${TOKEN}"
        }
        bat 'svn export --username amar.khan http://192.168.1.1/svn/Appzillon/3.6.0.S1/Studio '+'${defaulcotpath}'+ --password '+'${TOKEN}"
      }
    }
  }
  post {
    failure {
      script { buildstatus="Failed @Checkout Stage" }
      emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")
    }
  }
}

stage('Clean') {
  steps{
    withMaven(maven : 'mvn-3.1.0'){
      bat "cd ${defaulcotpath} && mvn clean"
    }
  }
}

stage('Versioning') {
  steps{
    script {
      def workspace = env.WORKSPACE
      def val1 = "APZPRODUCTVERSION"
      def val2 = ""+ "${params.AppzillonVersion}"+""
      def path = "${workspace}\\${defaulcotpath}\\appzillonstudio-core\\src\\main\\java\\com\\iexceed\\apz\\core\\Appzillon.java"
      def winpath = "${workspace}\\${defaulcotpath}\\appzillonstudio-

```

Svn Source code export
without .svn folder /history
into pipeline workspace

Job Start
notification

If anything fail while checkout stage
then email notification with stage name
where it got failed

Simple
mvn clean

If user has given version
option then setting up in
Appzillon.java class

```

client\package\windows\Appzillon.iss"
def command = "(gc ${path}) -replace ${val1}, ${val2} | sc ${path}"
def wincommand = "(gc ${winpath}) -replace ${val1}, ${val2} | sc ${winpath}"
def com = "" + "${command}" + ""
def wincom = "" + "${wincommand}" + ""
bat 'powershell -Command ' + "${com}"
bat 'powershell -Command ' + "${wincom}"
echo "${command}"
}
}
post {
failure {
script { buildstatus="Failed @Versioning Stage" }
emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")
}
}
}

```

Failure
notification
if happend

```

stage('Set CustID&Licence') {
steps{
script{
def val1 = "APZCUSTOMERID"
def val2 = "${params.CustId}"
if("${params.CustId}"!='APZCUSTOMERID'){
def path = "${workspace}\\${defaultcotpath}\\appzillonstudio-
core\\src\\main\\java\\com\\iexceed\\apz\\core\\Appzillon.java"
def command = "(gc ${path}) -replace ${val1}, '${val2}' | sc ${path}"
def com = "" + "${command}" + ""
bat 'powershell -Command ' + "${com}"
}
if("${params.Licence}"=="true"){
def tagstart = "private static String fffbuild"
def newtagstart = "//private static String fffbuild"
def tagstartwith4 = "////private static String fffbuild"
def paths = "${workspace}\\${defaultcotpath}\\appzillonstudio-
client\\src\\main\\java\\com\\iexceed\\apz\\ui\\core\\AppzillonUI.java"
def commands = "(gc ${paths}) -replace '${tagstart}', '${newtagstart}' | sc ${paths};(gc ${paths})
-replace '${tagstartwith4}', '${tagstart}' | sc ${paths}"
echo "${commands}"
def coms = "" + "${commands}" + ""
bat 'powershell -Command ' + "${coms}"
}
}
}
post {
failure {
script { buildstatus="Failed @Setting CustID" }
emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")
}
}
}

```

If user selected customer
id the setting up with user
requested id

Failure notification for
user if happened

```

stage('Build') {
steps{

```

In build stage we are doing maven install

```
withMaven(maven : 'mvn-3.1.0'){  
bat "cd ${defaultCotpath} && mvn install"  
}  
}
```

Post failure notification if happen

```
post {  
failure {  
script { buildstatus="Failed @Build Stage" }  
emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")  
}  
}  
}
```

To make our pipeline smooth we are commenting signin section in build.xml so that job does not fail

```
stage('SignIn Comment'){  
steps{  
script{  
def tagstart = ""  
<exec""  
def newtagstart = ""  
<!--  
<exec""  
def tagend = ""  
</exec>""  
def newtagend = ""  
</exec> -->""  
def path = "${workspace}\\${defaultCotpath}\\appzillonstudio-client\\build.xml"  
def command = "(gc ${path}) -replace ${tagstart}, ${newtagstart} | sc ${path} ;(gc ${path})  
-replace ${tagend}, ${newtagend} | sc ${path}"  
def com = "" + "${command}" + ""  
bat 'powershell -Command ' + "${com}"  
}  
}  
post {  
failure {  
script { buildstatus="Failed @SignIn Stage" }  
emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")  
}  
}  
}
```

stage('Jar Distribution')

```
steps{  
parallel(  
linux: {  
echo "JAR Distribution over linux server"  
bat 'cd d:\\softs && plink -pw '+'${pass}'+ 'superuser@192.168.1.6 '+' -batch -hostkey '+'${  
{key}"+ " "+"mkdir /AppzillonBuilds/'+"${buildnumber}"  
bat 'cd d:\\softs && pscp -r -pw '+'${pass}'+ ' -batch -hostkey '+'${key}"+ " "+"${workspace}\\$  
{defaultCotpath}\\appzillonstudio-client"+ " "+"${workspace}\\${defaultCotpath}\\ClientJars"+ " "+"$  
{workspace}\\${defaultCotpath}\\KeyStore"+ " "+"${workspace}\\$  
{defaultCotpath}\\appzillonstudio-commandbuild\\target\\Appzillon"+ '  
superuser@192.168.1.6:/AppzillonBuilds/'+"${buildnumber}"  
},  
Mac: {  
echo "JAR Distribution over MAC"
```

Jar distribution to linux server
192.168.1.6 from jenkins
server(window)

Jar distribution to Mac server
192.168.1.206 from jenkins
server(window)

```

bat 'cd d:\softs && plink -pw '+'${Macpass}'+ admin@192.168.1.206 '+' -batch -hostkey '+'$
{Mackey}"+ " "+"mkdir /AppzillonBuilds/"+"${buildnumber}"
bat 'cd d:\softs && pscp -r -pw '+'${Macpass}"+ -batch -hostkey "+"${Mackey}"+ " "+"$
{workspace}\\${defaultcotpath}\\appzillonstudio-client"+ " "+"${workspace}\\$
{defaultcotpath}\\ClientJars"+ " "+"${workspace}\\${defaultcotpath}\\KeyStore"+
admin@192.168.1.206:/AppzillonBuilds/"+"${buildnumber}"
}
)
}
post {
failure {
script { buildstatus="Failed @Jar Distribution Stage" }
emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")
}
}
}
stage('Packaging') {
steps{
parallel(
linux: {
echo "deb packaging"
bat 'cd d:\softs && plink -pw '+'${pass}'+ superuser@192.168.1.6 '+' -batch -hostkey '+'$
{key}"+ " '+'cd /AppzillonBuilds/"+"${buildnumber}"+'/appzillonstudio-client;ant'
},
window: {
echo "exe packaging"
bat "cd ${defaultcotpath}/appzillonstudio-client && ant"
},
mac: {
echo "mac packaging"
// bat 'cd d:\softs && plink -pw '+'${Macpass}'+ admin@192.168.1.206 '+' -batch -hostkey '+'$
{Mackey}"+ " '+'/Applications/apache-ant-1.9.4/bin/ant -f /AppzillonBuilds/"+"$
{buildnumber}"+'/appzillonstudio-client'
bat 'cd d:\softs && plink -pw '+'${Macpass}'+ admin@192.168.1.206 '+' -batch -hostkey '+'$
{Mackey}"+ " '+'cd /AppzillonBuilds/"+"${buildnumber}"+'/appzillonstudio-
client/Applications/apache-ant-1.9.4/bin/ant'
// bat 'cd d:\softs && pscp -r -pw '+'${pass}"+ -batch -hostkey "+"${key}"+ " "+"${workspace}\\
${defaultcotpath}\\appzillonstudio-client"+ " "+"${workspace}\\${defaultcotpath}\\ClientJars"+
"+"${workspace}\\${defaultcotpath}\\KeyStore"+ amar.khan@192.168.1.178:/tmp/test/1'
}
)
}
post {
success{
script{
sysdate="${nowdate}"
bat 'cd d:\softs && plink -pw '+'${pass}'+ superuser@192.168.1.6 '+' -batch -hostkey '+'$
{key}"+ " "+"mkdir -p /AppzillonTesting/"+"${params.AppzillonVersion}"/+"${sysdate}"
}
}
failure {
script { buildstatus="Failed @Packaging Stage" }

```

Post failure
notification

Parallel execution of packaging in all three
window,linux,mac using ant

If Success then creating folder in
share are with timestamp

Failure
notification

```

emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")
}
}
}
stage('Binary Sharing') {
steps{
parallel(
linux: {
echo "deb sharing"
//appzillonstudio-client/build/bundles
// bat 'cd d:\softs && plink -pw '+'${pass}'+ 'superuser@192.168.1.6 '+' -batch -hostkey '+'$
{key}"+ " '+cd /AppzillonBuilds/'+'${buildnumber}'+'/appzillonstudio-client;ant'
bat 'cd d:\softs && plink -pw '+'${pass}'+ 'superuser@192.168.1.6 '+' -batch -hostkey '+'$
{key}"+ " '+cp -a /AppzillonBuilds/${buildnumber}/appzillonstudio-client/build/bundles/*.deb
/AppzillonTesting/${params.AppzillonVersion}/${sysdate};cp -a /AppzillonBuilds/$
{buildnumber}/Appzillon /AppzillonTesting/${params.AppzillonVersion}/${sysdate}"
// bat 'cd d:\softs && pscp -r -pw '+'${pass}"+ " -batch -hostkey '+'${key}"+ " '+'${workspace}\\
${defaultcopath}\\appzillonstudio-client"+ " '+'${workspace}\\${defaultcopath}\\ClientJars"+
" '+'${workspace}\\${defaultcopath}\\KeyStore"+ 'superuser@192.168.1.6:/AppzillonBuilds/'+'$
{buildnumber}"
},
window: {
echo "exe sharing"
bat 'cd d:\softs && pscp -r -pw '+'${pass}"+ " -batch -hostkey '+'${key}"+ " '+'${workspace}\\$
{defaultcopath}\\appzillonstudio-client\\build\\bundles\\*.exe"+
superuser@192.168.1.6:/AppzillonTesting/${params.AppzillonVersion}/${sysdate}/'
},
mac: {
echo "mac sharing"
bat 'cd d:\softs && plink -pw '+'${Macpass}'+ 'admin@192.168.1.206 '+' -batch -hostkey '+'$
{Mackey}"+ " '+/usr/local/bin/sshpas -p '+'${pass}"+ " scp -r '+'/AppzillonBuilds/$
{buildnumber}/appzillonstudio-client/build/bundles/*.dmg
superuser@192.168.1.6:/AppzillonTesting/${params.AppzillonVersion}/${sysdate}"
// bat 'cd d:\softs && plink -pw '+'${Macpass}'+ 'admin@192.168.1.206 '+' -batch -hostkey '+'$
{Mackey}"+ " '+/Applications/apache-ant-1.9.4/bin/ant -f /AppzillonBuilds/'+'$
{buildnumber}'+'/appzillonstudio-client'
// bat 'cd d:\softs && plink -pw '+'${Macpass}'+ 'admin@192.168.1.206 '+' -batch -hostkey '+'$
{Mackey}"+ " '+sshpas -p '+'${key}"+ " scp -r"
// bat 'cd d:\softs && pscp -r -pw '+'${pass}"+ " -batch -hostkey '+'${key}"+ " '+'${workspace}\\
${defaultcopath}\\appzillonstudio-client"+ " '+'${workspace}\\${defaultcopath}\\ClientJars"+
" '+'${workspace}\\${defaultcopath}\\KeyStore"+ 'amar.khan@192.168.1.178:/tmp/test/1'
}
)
}
post {
failure {
script { buildstatus="Failed @Packaging Stage" }
emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")
}
}
}
// stage('Notify') {

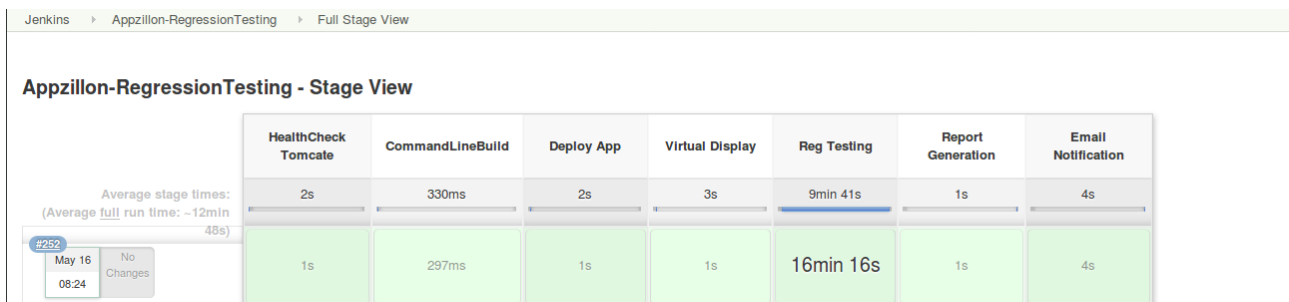
```

Binary Sharing :This section should go after testing as per suggestion so need to make changes binary should copy to common area only when automation testing success.


```
//      steps {
//      // send to email
//      emailext (
//      subject: "BUILD: Job '${env.JOB_NAME} [${env.BUILD_NUMBER}]'",
//      mimeType: 'text/html',
//      body: """
<p>build available: Job '${env.JOB_NAME} [${env.BUILD_NUMBER}]':
</p>
//
<p>Check console output at &QUOT;
  <a href='${env.BUILD_URL}'>${env.JOB_NAME} [${env.BUILD_NUMBER}]
  </a>&QUOT;
</p>""",
//      to: "amar.khan@i-exceed.com",
//      recipientProviders: [[class: 'CulpritsRecipientProvider']]
//      )
//    }
//  }
}

post {
success {
  build(job: "Appzillon-RegressionTesting", parameters: [[class: 'StringParameterValue', name:
'BuildPipelineNumber', value: "${buildnumber}" ]], propagate: true, wait: false)
  emailNotification("${buildstatus}", "${sysdate}", "${params.AppzillonVersion}")
}
}
}
```

Post success of pipeline user will get notify with binary path details



```
def pass = "Qaz#dc5gb"
def key = "ab:a9:b8:34:48:0b:a5:0d:e3:56:ae:5e:41:2f:41:9b"
def buildnumber = env.BUILD_NUMBER
def Macpass = "p@ssw0rd"
def Mackey = "ab:a9:b8:34:48:0b:a5:0d:e3:56:ae:5e:41:2f:41:9b"
```

Variable declaration

Email Notificaation function

```
def emailNotification(testingstatus){
def emailBody="";
def recipientlist="";
```

```

if("${testingstatus}"=='Started'){
recipientlist="preetish.das@i-exceed.com,darshanl.sagar@i-exceed.com,shyam.kumar@i-
exceed.com,amar.khan@i-exceed.com,cc:chandu.balaji@i-exceed.com"
emailBody= ""
<p>Hi All,
</p>
<p>AppzillonStudio-3.6.0 build triggered.
  <br>Please find the latest build details below.
</p>
<p>Check console output at
  <a href='${env.BUILD_URL}'>${env.JOB_NAME}[${env.BUILD_NUMBER}]
  </a>
</p>
<p>Regards,
  <br>Appzillon DevTeam
  <br>I-Exceed Technology Solutions
</p>""
}
else if("${testingstatus}"=='Success'){
recipientlist="preetish.das@i-exceed.com,darshanl.sagar@i-exceed.com,shyam.kumar@i-
exceed.com,amar.khan@i-exceed.com,cc:chandu.balaji@i-exceed.com"
emailBody= ""
<p style="color:#074207;">Hi All,
  <br>
  <br> AppzillonStudio-3.6.0 Testing Success:
  <br>
  <br>Click
  <a href='http://192.168.1.6/Report.html'>here
  </a>for Report
  <br>Click
  <a href='http://192.168.1.6/AppzillonTest.html'>here
  </a>for logs
  <br>Job ${env.JOB_NAME} [${env.BUILD_NUMBER}]
  <br>
  <br> Check console output at
  <a href='${env.BUILD_URL}'>${env.JOB_NAME} [${env.BUILD_NUMBER}]
  </a>
  <br>
  <br>Regards,
  <br>
  <br>Appzillon IDE Team
  <br>I-Exceed Technology Solutions
</p>""
}
else{
recipientlist="preetish.das@i-exceed.com,darshanl.sagar@i-exceed.com,shyam.kumar@i-
exceed.com,amar.khan@i-exceed.com,cc:chandu.balaji@i-exceed.com"
emailBody= ""
<p style="color:red;">Hi All,
  <br>
  <br> AppzillonStudio-3.6.0 build Success:
  <br>

```

```

<br>Click here
<a href='http://192.168.1.151/AppzillonTest.html'>
  <br>Job ${env.JOB_NAME} [${env.BUILD_NUMBER}]
  <br>
  <br> Check console output at
  <a href='${env.BUILD_URL}'>${env.JOB_NAME} [${env.BUILD_NUMBER}]
</a>
<br>
<br>Regards,
<br>
<br>Appzillon IDE Team
<br>I-Exceed Technology Solutions
</p>""

```

```

}
emailx (
subject: "Appzillon-3.6.0.S1 Testing ${testingstatus}",
mimeType: 'text/html',
body: "${emailBody}",
to: "${recipientlist}",
replyTo: "jenkinsadmin@i-exceed.com",
from: 'jenkinsadmin@i-exceed.com',
recipientProviders: [[class: 'CulpritsRecipientProvider']]
)
}

```

Email Notificaation function-End

```

pipeline {
  agent any

  parameters {
    string(name: 'Project Path', defaultValue: 'Change Default Path', description: 'Path for Appzillon
Project source code checkout!!')
    string(name: 'BuildPipelineNumber', defaultValue: 'LatestBuildNumber', description: 'Latest Build
Number for AppzillonBuildPipline!!')
  }
  /* ... unchanged ... */
  stages {
    stage ('HealthCheck Tomcate') {
      steps {
        script{
          echo 'tomcate start'
          def cmd = "curl -Is http://192.168.1.6:8081|head -n1|grep 200"
          bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ 'superuser@192.168.1.6 '+' -batch -hostkey '+'$
{Mackey}'+ ' '+'if(${cmd}); then echo 'Apache is up and running'; else sh
/home/superuser/AppzAutomationTesting/apache-tomcat-8.0.30/bin/startup.sh;fi"
          //bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ 'amar.khan@192.168.5.138 '+' -batch -hostkey
"+"${Mackey}'+ ' '+'if(${cmd}); then echo 'Apache is up and running'; else echo 'Starting
Apache';fi"
          // bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ 'amar.khan@192.168.5.138 '+' -batch -hostkey
"+"${Mackey}'+ ' '+'bash -c pwd'
          // bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ 'amar.khan@192.168.5.138 '+' -batch -hostkey
"+"${Mackey}'+ ' curl -s -o /dev/null -w %{http_code} http://localhost:8080"
        }
      }
    }
  }
}

```

Setting up UI parametrs

Checking tomcate
healthcheck if not ok then
restarting tomcate

Command line build of Appzillon reg App
project with recently genrated command
line jars utility

```

}
}
stage ('CommandLineBuild') {
  steps {
    echo 'build app'
    // bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ ' amar.khan@192.168.1.151 '+' -batch -hostkey
    '+'${Mackey}'+ ' '+'cd /home/i-exceed.com/amar.khan/Documents/testing/Appzillon/lib;java -jar
    appzillonstudio-commandbuild-ci.jar /home/i-
    exceed.com/amar.khan/Documents/testing/Appzillon/Bin/Preferences'
  }
}
stage ('Deploy App') {
  steps {
    script{
      echo 'hi'
      def cmd ="curl -Is http://192.168.1.6:8081|head -n1|grep 200"
      // bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ ' amar.khan@192.168.1.151 '+' -batch -hostkey
      '+'${Mackey}'+ ' '+'if(${cmd}); then echo 'Apache is up and running'; else sh /opt/apache-
      tomcat-8.0.30/bin/startup.sh;fi ;while(true); do if(${cmd});then break;else sleep 5;fi; done"
      //bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ ' amar.khan@192.168.1.151 '+' -batch -hostkey
      '+'${Mackey}'+ ' '+'cp /home/i-
      exceed.com/amar.khan/Documents/testing/Project/BaseApp/bin/Web/*.war /home/i-
      exceed.com/amar.khan/Documents/testing/Project/BaseApp/bin/Server/*.war /opt/apache-tomcat-
      8.0.30/webapps'
      bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ ' superuser@192.168.1.6 '+' -batch -hostkey '+'$
      {Mackey}'+ ' '+'if(${cmd}); then echo 'Apache is up and running'; else sh
      /home/superuser/AppzAutomationTesting/apache-tomcat-8.0.30/bin/startup.sh;fi ;while(true); do
      if(${cmd});then break;else sleep 5;fi; done"
    }
  }
}
stage('Virtual Display'){
  steps{
    script{
      echo 'hi'
      def startvdipcmd=""echo p@ssw0rd | sudo -S sh
      /home/superuser/AppzAutomationTesting/Xvfb.sh start""
      checkxvfbcmd="pidof /usr/bin/Xvfb"
      def checkvirtualcmd="if(${checkxvfbcmd}); then echo '*****Xvfb Server is up and running*****';
      else ${startvdipcmd};fi ;"
      bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ ' superuser@192.168.1.6 '+' -batch -hostkey '+'$
      {Mackey}'+ ' '+'${checkvirtualcmd}"
    }
    // bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ ' amar.khan@192.168.5.138 '+' -batch -hostkey
    '+'${Mackey}'+ ' '+'export DISPLAY=:99'
    // chmod +x Xvfb.sh
    // sh Xvfb.sh start
  }
}
stage ('Reg Testing') {
  steps {
    echo 'hi'

```

Deploying Testing.war and AppzillonServer.war to tomcat for testing and check until tomcat up

We are having Xvfb server for testing in virtual Display as headless so checking for its status if its not running than starting

Xvfb.sh script is attached in the end

Reg Testing is starting on virtual display :99

```

bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ 'superuser@192.168.1.6 '+' -batch -hostkey '+'$
{Mackey}'+ " "+"export PATH=$PATH:/home/superuser/AppzAutomationTesting/apache-maven-
3.5.2/bin;"+'export DISPLAY=:99;sleep 5;cd
/home/superuser/AppzAutomationTesting/appzillonstudio-
client/Appzillon/AppzillonDefaults/AppzillonAppTestProject/appzillon-test;pwd;echo
$DISPLAY'+';'+""ps aux|grep Xvfb""+';'+""mvn exec:java 2>&1 | tee /tmp/AppzillonTest.html""
}
}
stage('Report Generation'){
steps{
script{
def newline=""sed -e 's/$/
<br>/' -i /tmp/AppzillonTest.html""
def suffix=""echo '
</p>
</body>
</html>' >> /tmp/AppzillonTest.html""
def exposetestcmd=""echo p@ssw0rd | sudo -S cp /tmp/AppzillonTest.html /tmp/Report.html
/var/www/git""
def prefix=""echo '
<html>
<head>
<title>Appzillon Test Logs
</title>
</head>
<body>
<h1 style="background-color:Yellow;" align="center" >
<font color="SlateBlue">Appzillon Testcase Log
</font>
</h1>
<p style="background-color:LightGreen;"">
<button type="button" onclick="window.location.href='/tmp/test.html'" style="float:
right;">Report
</button>' | cat - /tmp/AppzillonTest.html > temp && mv temp /tmp/AppzillonTest.html""
// def prefix=""echo '
<html>
<head>
<title>Appzillon Test Logs
</title>
</head>
<body>
<h1 style="background-color:Yellow;" align="center" >
<font color="SlateBlue">Appzillon Testcase Log
</font>
</h1>
<p style="background-color:LightGreen;"">' | cat - /tmp/AppzillonTest.html > temp && mv
temp /tmp/AppzillonTest.html""
bat 'cd d:\\softs && plink -pw '+'${Macpass}'+ 'superuser@192.168.1.6 '+' -batch -hostkey
"+"${Mackey}'+ " "+"cd /home/superuser/AppzAutomationTesting;./reportGenScript.sh"+';'+"$
{exposetestcmd}"
// def workspace = env.WORKSPACE

```

Report generation script is attached in end of pipeline script

```

// def serverResultPath="${workspace}\\${buildnumber}\\robot\\report";
// bat "if not exist ${serverResultPath} mkdir ${serverResultPath}"
// bat 'cd d:\\softs && plink -pw "${Macpass}" amar.khan@192.168.5.138 '+' -batch
-hostkey "+"${Mackey} "+" '+'cat /home/i-
exceed.com/amar.khan/Documents/testing/output.xml>'+ "${serverResultPath}\\'+output.xml'
// bat 'cd d:\\softs && plink -pw "${Macpass}" amar.khan@192.168.5.138 '+' -batch
-hostkey "+"${Mackey} "+" '+'cat /home/i-exceed.com/amar.khan/Documents/testing/log.html>'+ "${
serverResultPath}\\'+log.html'
// bat 'cd d:\\softs && plink -pw "${Macpass}" amar.khan@192.168.5.138 '+' -batch
-hostkey "+"${Mackey} "+" '+'cat /home/i-
exceed.com/amar.khan/Documents/testing/report.html>'+ "${serverResultPath}\\'+report.html'
}
}}
stage('Email Notification'){
steps{
echo 'hi'
emailNotification('Success')
}
}
}
}
}
}

```

Email notification with testcase log
<http://192.168.1.6/AppzillonTest.html>
And report
<http://192.168.1.6/Report.html>

XVFB Script for Virtual Display

```

#!/bin/sh
mode=$1
case "$mode" in
'start')
echo "****Starting up the Virtual Frame Buffer on Screen 1****"
/usr/bin/Xvfb :99 -screen 0 1152x900x8 > /dev/null 2>&1 &
;;
'stop')
echo "****Stoping Active Virtual Frame****"
kill -9 $(pgrep Xvfb)
;;
*)
echo " Usage: "
echo " $0 start (start XVFB)"
echo " $0 stop (stop XVFB)"
exit 1
;;
esac
dbus-uuidgen > /var/lib/dbus/machine-id
exit 0

```

HTML Report Generation Script for Testcase results(reportGenScript.sh)

```

rm /tmp/Report.html;sed -e 's/$/<br>/' -i /tmp/AppzillonTest.html;echo '</p></body></html>'
>> /tmp/AppzillonTest.html;echo '<html><head><title>Appzillon Test
Logs</title></head><body><h1 style="background-color:Yellow;" align="center" ><font
color="SlateBlue">Appzillon Testcase Log</font></h1> <p style="background-
color:LightGreen;"><button type="button" onclick="window.location.href='/tmp/test.html'"

```

```

style="float: right;">Report</button>' | cat - /tmp/AppzillonTest.html > temp && mv temp
/tmp/AppzillonTest.html;
cat /tmp/AppzillonTest.html |sed -n '/Results for Microapp:DataBinding/,/Results for
Microapp:DataBinding<br>/p' >input.csv;while read INPUT ; do echo "<tr><td>$
{INPUT//;/</td><td>}</td></tr>" ; done < input.csv >/tmp/Report.html;
cat /tmp/AppzillonTest.html |sed -n '/Results for Microapp:Designer<br>/,/Results for
Microapp:Designer<br>/p' >input.csv;while read INPUT ; do echo "<tr><td>$
{INPUT//;/</td><td>}</td></tr>" ; done < input.csv >/tmp/Report.html;
cat /tmp/AppzillonTest.html |sed -n '/Results for Microapp:Infra<br>/,/Results for
Microapp:Infra<br>/p' >input.csv;while read INPUT ; do echo "<tr><td>$
{INPUT//;/</td><td>}</td></tr>" ; done < input.csv >/tmp/Report.html;
echo '<html><head><style>table, th, td {border: 1px solid black;border-collapse: collapse;}th, td
{ padding: 1px; text-align: left; color:white;}table#t01 { background-color:
#3b3a30;}tr:nth-child(even) {background-color: #686256;}</style><body style="background-
color:powderblue;"><h2 style="color:#4c4949;" align="center" >Appzillon Testcase Results</h2><br><table
id="t01" align="center">' | cat - /tmp/Report.html > temp && mv temp /tmp/Report.html;echo
"</table></body></head><footer><p align="center"><br>Posted by: Appzillon IDE Team
<br>Contact information: <a href="mailto:appzillondevops@i-exceed.com">appzillondevops@i-
exceed.com</a>.</p></footer></html>" >> /tmp/Report.html;sed -i 's\
[com.iexceed.utc.Appzillon.main()] INFO//g' /tmp/Report.html;sed -i 's/TestReport/\<button
type="button" onclick="window.location.href='192.168.1.6'" style="float\
center\;"\>logs\</button>/g'
/tmp/Report.html;p='<tr><td>FunctionId</td><td>Description</td><td>TestCaseNo</td><td>
SubCaseNo</td><td>Result<br></td></tr>';q='<tr><td
bgcolor="#FFA500">FunctionId</td><td bgcolor="#FFA500">Description</td><td
bgcolor="#FFA500">TestCaseNo</td><td bgcolor="#FFA500">SubCaseNo</td><td
bgcolor="#FFA500">Result<br></td></tr>';sed -i "s#\$p#\$q#g" /tmp/Report.html

```