THE HOME DEPOT

Standard Operating Procedure

|  |  |
| --- | --- |
| Version Number | 1.1 |
| Creation Date | 26-Sep-2018 |
| Author | THD offshore Team |
| Owner | THD offshore Team |

Revision History

| Version | Date | Summary of changes | Revision Marks |
| --- | --- | --- | --- |
| Draft | 26-Sep-2018 | Initial draft created |  |
|  |  |  |  |

Table of Contents

[1. Overview 4](#_Toc525729643)

[2. List of Key Projects 4](#_Toc525729644)

[2.1 SyntBots Automation 4](#_Toc525729645)

[2.1.1 Business Porfolio 4](#_Toc525729646)

[2.1.2 Project Overview 4](#_Toc525729647)

[2.1.3 Technology Stack 4](#_Toc525729648)

[2.1.4 Required software & Access 4](#_Toc525729649)

[2.1.5 Steps to follow the development tasks 4](#_Toc525729650)

[2.1.6 Steps to follow testing 4](#_Toc525729651)

[2.1.7 Steps to follow deployment 4](#_Toc525729652)

[2.1.8 Version Control 4](#_Toc525729653)

[2.2 Grid to PCF Migration 5](#_Toc525729654)

[2.2.1 Business Porfolio 5](#_Toc525729655)

[2.2.2 Project Overview 5](#_Toc525729656)

[2.2.3 Technology Stack 5](#_Toc525729657)

[2.2.4 Required software & Access 5](#_Toc525729658)

[2.2.5 Steps to follow the development tasks 5](#_Toc525729659)

[2.2.6 Steps to follow testing 5](#_Toc525729660)

[2.2.7 Steps to follow deployment 5](#_Toc525729661)

[2.2.8 Version Control 5](#_Toc525729662)

[2.3 Java Migration 5](#_Toc525729663)

[2.3.1 Business Porfolio 5](#_Toc525729664)

[2.3.2 Project Overview 5](#_Toc525729665)

[2.3.3 Technology Stack 5](#_Toc525729666)

[2.3.4 Required software & Access 5](#_Toc525729667)

[2.3.5 Steps to follow the development tasks 5](#_Toc525729668)

[2.3.6 Steps to follow testing 5](#_Toc525729669)

[2.3.7 Steps to follow deployment 5](#_Toc525729670)

[2.3.8 Version Control 5](#_Toc525729671)

[2.4 Digital Mordenization – HDMS 5](#_Toc525729672)

[2.5 Production Support 5](#_Toc525729673)

[3. Review & Signoff 6](#_Toc525729674)

[4. Appendix 6](#_Toc525729675)

# Overview

This document describes the standard way of executing each key project running in THD offshore delivery. Purpose of having this document is,

* To follow unified process across the teams
* Go to document for any new team member in the team

# List of Key Projects

#### SyntBots Automation

* 1. Grid to PCF migration
  2. Java Migration
  3. Digital Mordenization - HDMS
  4. Production Support

## SyntBots Automation

### Business Porfolio

### Project Overview

### Technology Stack

### Required software & Access

### Steps to follow the development tasks

### Steps to follow testing

### Steps to follow deployment

### Version Control

## Grid to PCF Migration

### Business Porfolio

### Project Overview

### Technology Stack

### Required software & Access

### Steps to follow the development tasks

### Steps to follow testing

### Steps to follow deployment

### Version Control

## Java Migration

### Business Porfolio

### Project Overview

### Technology Stack

### Required software & Access

### Steps to follow the development tasks

### Steps to follow testing

### Steps to follow deployment

### Version Control

## Digital Mordenization – HDMS

## Production Support

## Grid to PCF Migration

### Business Porfolio

### Project Overview

Migrating all java classes into BO, DAO and Util or other helper classes using spring features. Once the conversion is done, the application will be pushed to PCF using CFCLI commands. Following are the list of applications included in migration activity.

|  |
| --- |
| ICONXPayments |
| ICONWeb |
| MyInstall |
| MyInstallBatch |
| MMHSAppointmentScheduler |
| Document Viewer |
| Purge Order Retrieval |
| HDE Document Migration Batch |
| HDE Integration |
| Document Management |
| ICONXDOCWSTC7 |
| ICONXMOBWSTC7 |
| iconxTC7 |
| SSSHWebServices |

### Technology Stack

Following are the technologies used in the Grid to Migration Project,

#### Java 1.8

#### Spring Boot

#### Maven 3.5 or above.

### Required software & Access

Following are the required softwares and accesses,

#### STS

#### Git Access.

#### Grid Stat access.

#### SonarQube access.

#### Splunk access.

#### Fortify access.

#### Postman

#### Advanced Rest Client

### Steps to follow the development tasks

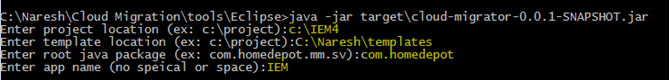
Following are the steps to proceed the development tasks.

#### Find the application name from GridStats and identify the datasource details, git branch name etc.

#### Find the git repository of the grid application and clone it using the command “**git clone repoUrl**”.

#### Once the cloning is done, the next step will be to convert it as a maven project. To perform this action follow below step:

#### Import the **“Cloud migrator**” project in STS and execute **“MigrationSevices.java”.** While executing MigrationServices.java, you have to provide the following information which is shown as a sample here,



#### Once Step-3 is completed, goto the project folder where you will see the newly generated folder “**migrationdo-delete”** after the execution of MigrationServices.java.

#### Open Migrationdo-delete->project-> Import the project into sts.

#### Goto STS. Click on File->Import->Maven->Existing Maven Project->provide project path->Click OK.

#### Update project by right clicking project-> maven ->update project

#### Add or exclude dependencies from pom.xml to resolve errors, if any.

#### Form the spring project with PCF config (with Spring boot 1.5.x).

#### Add all Datasource details in \*Application.java file as shown below. The Value attribute is mapped to VCAP services in PCF and in case the VCAP services are not available, it takes the default value



#### Migrate all java classes in BO, DAO and Util or other helper classes using Spring features.

#### Use annotations (@service, @component, @Repository) for all corresponding Classes.

#### For all service endpoints - either convert to Jersey jar v2.0 or use RestController annotation.

#### Covert all the DAO to JDBCTemplate.

#### Build the project by clicking Maven Build: clean install –DskipTests.

#### In target folder a jar will created if the build is success.

#### If build is success, run the application by right clicking project-> Run as-> Spring Boot App.

#### Once the executions are done, check the quality of the code using SonarQube. The code should pass the QualityGate of SonarQube.

#### Login to Fortify to fix the security related issues if any exist in the application.

#### Verify whether your Loggers are working in Splunk.

### Steps to follow testing

#### Hit the URL <http://localhost:8080> in postman to test the application.

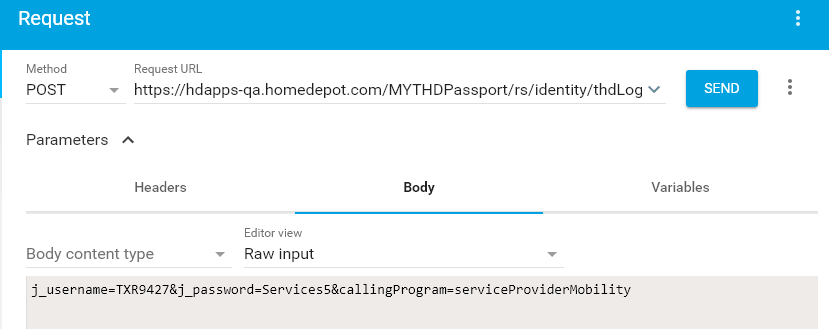
#### The key to be passed in Headers for local testing “Cookie”. While testing, QA grid Header keys to be passed are “Cookie”, “appToken”, “Authorization”.

#### Give the Request body for the endpoint.

# 

# 

#### Generate Cookie using Advanced Rest client



### Steps to follow deployment

**Pushing Application into PCF:**

1. From target folder of application, Login to the below URL from CFCLI command prompt using the command **“cf login”**.

API endpoint: <https://api.run-np.homedepot.com>

Email :

Password :

1. Push application to PCF using the below command

EX: cf push applicationname -p ./Application-0.0.1-SNAPSHOT.jar -i 1 -m 512M

1. After successful deployment of application to PCF, bind the required data sources using cup services.

Use cf cups db2-service -p "dbUrl, user, password"

Above command will prompt you to enter dbUrl, user & password

Once you enter the details it will create service with name “db2-service” (you can choose any name)

Use cf bind-service applicationname db2-service to bind the service to your application (you can do it through UI as well)

Use cf restage applicationname to restart your application

Test the application by clicking URL.

### Version Control

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Author** |
| V1.0.0 | 9/27/2018 | Sangeetha Ravi |
|  |  |  |
|  |  |  |

# Review & Signoff

|  |  |  |
| --- | --- | --- |
| Review Date | Approver Name | Comments |
|  |  |  |
|  |  |  |

# Appendix