

# オープンソースのワークフローエンジンを利用して簡単なワークフローシステムを作ってみよう！

～超初心者向けです。どうぞおこしやす～

日本Activitiユーザ会

遠藤 俊一郎

OSC 2018 Kyoto

2018.8.3

# おしながき

- 自己紹介
- ワークフロー、BPMとは？
- Activitiについて
- Flowableについて
- Workflow、BPMに必要な機能
- Flowableを用いてワークフローシステムをつくってみよう！
- まとめ

# 自己紹介

- 出身地

神奈川県

- 主な出沒地

東京都内でSEをやっています。最近はETL系のオープンソースや試験の自動化といった技術検証しています。

- プライベート

聖地巡礼(京都(京都市内、舞鶴)、広島(呉、竹原)、岐阜(飛騨高山、白川郷)、...)が趣味です。

まだまだ行きたい所はたくさん。。



# ワークフロー、BPMとは？

- ワークフローとは？

<https://www.bpm-j.org/keyword/w/530/>

(日本ビジネスプロセス・マネジメント協会HP)

- BPMとは？

<https://www.bpm-j.org/bpm/>

(日本ビジネスプロセス・マネジメント協会HP)

# Activitiについて

- オープンソースのワークフロー、BPMプラットフォーム。

<<https://www.activiti.org/>>

- ライセンス

Apache 2.0 license

- 歴史

まもなくVer. 7.0公開  
(7/1現在 Early Access版公開中)

		2011	2012	2013	2014	2015	2016	2017	2018
Activiti	5.x	▲→▲ 5.0 5.6	▲→▲ 5.7 5.11	▲ ▲ 5.12 5.13	▲→▲ 5.14 5.17	▲ ▲ 5.18 5.19	▲ ▲ ▲ 5.20 5.21 5.22		
	6.x						<b>fork</b>	▲ 6.0	
Flowable	5.x						▲ 5.22	▲ 5.23	
	6.x							▲ ▲ ▲ 6.0 6.1 6.2	▲ 6.3

# Flowableについて

- オープンソースのワークフロー、BPMプラットフォーム。  
<<https://www.flowable.org/>>
- ライセンス  
Apache 2.0 license
- 歴史

		2011	2012	2013	2014	2015	2016	2017	2018
Activiti	5.x	▲→▲ 5.0 5.6	▲→▲ 5.7 5.11	▲ ▲ 5.12 5.13	▲→▲ 5.14 5.17	▲ ▲ 5.18 5.19	▲ ▲ ▲ 5.20 5.21 5.22		
	6.x						<b>fork</b>	▲ 6.0	
Flowable	5.x						▲ 5.22	▲ 5.23	
	6.x							▲ ▲ ▲ 6.0 6.1 6.2	▲ 6.3

# Flowableについて

- オープンソースのワークフロー、BPMプラットフォーム。  
<<https://www.flowable.org/>>
- ライセンス  
Apache 2.0 license
- 歴史

	2011	2012	2013	2014	2015	2016	2017	2018
--	------	------	------	------	------	------	------	------

直近ではFlowable側のリリース活動が盛ん。  
なので、今日はFlowableで説明します...

本日の利用機能の範囲ではVer6.x系はActivit,Flowableともにあまり違いはないです...

	5.x						▲ 5.22	▲ 5.23			
Flowable	6.x							▲ 6.0	▲ 6.1	▲ 6.2	▲ 6.3

# Flowableについて

- ソフトウェア要件

JDK	JDK8以上
IDE	Eclipse Mars or Neon
DBMS	主要なDBサーバ
APサーバ	一般的なAPサーバ

- 関連する言葉や技術

Workflow

BPM(Business Process Management)

BPMN(Business Process Modeling Notation)

DMN(Dicision Model and Notation)

CMMN(Case Management Model and Notation)

RPA(Robotic Process Automation)



# Workflow、BPMに必要な機能

BPM/ワークフローエンジンが具備する機能(今回ご紹介する一部機能抜粋、その他も多数ありますが…)

- ①ユーザを明確にし、役割を決める。またユーザを束ねたグループについても同様に管理する  
⇒ 仕事に関連する**ユーザの登録・管理機能**
- ②プロセス(業務のフローや処理のフロー)を設計する  
⇒ **仕事の流れ全体・実施順序・実施条件を定義する機能**
- ③データ入力のための手段(フォームを作成)を準備する  
⇒ **データ入力用画面を簡易作成する機能**

# Flowableを用いてワークフローシステムをつくってみよう！

それではFlowableを使ってワークフローシステムを作成します。

## □本日ご紹介する環境

- JDK8
- Tomcat7 or 8(8.5.31)
- Flowable6(6.3.1)

もし、本日環境がなくても非常に簡単なステップで動作・体験することが可能です。

本日の資料は後日WEB上にアップ予定ですので、「こんな感じで作成するのか」と気楽に見ていただけると幸いです。

# シナリオ

## 荷物の発送手配に関するお仕事

### ① 発送依頼表の起票



### ② 詳細情報の入力



### ③ 発送処理の実施

- 発送先の地方 (select\_region) の入力とともに発送依頼を起票する
- 誰でも実施可能



- 発送先県名(pref)、発送予定日 (shipment\_date)、荷物の3辺の長さ(sizecm)、荷物の重さ(weightkg)を入力
- 発送依頼表を起票した人が実施可能

- 荷物の3辺の長さや荷物の重さに従い、「マネージャ」もしくは「メンバーのうちの誰か」にタスクがアサインされ、荷物追跡番号を入力し、発送処理完了
- 条件に応じた担当者のみ可能



# 作成の流れ

1. **動作環境の準備**      ... Flowableのダウンロードとインストール
2. **環境の確認**      ... Flowableの起動確認
3. **flowable-idmによる作業**      ... Flowableにユーザ・グループ作成、権限設定
4. **flowable-modelerによる作業**
  - I. **ワークフローを作成**      ...ワークフローの描画
  - II. **データフォームの作成**      ... 送信先入力などの入力フォームを作成
  - III. **タスクの担当者割り当て**      ...ワークフローのタスクにアサイン先を登録
  - IV. **分岐の条件設定**      ...ワークフロー内の分岐の条件設定
  - V. **ワークフローのデプロイ**      ...作成したワークフローをランタイムへ登録
5. **flowable-taskで動かしてみよう！**

1. 動作環境の準備
2. 環境の確認
3. flowable-idmによる作業
4. flowable-modelerによる作業
  - I. ワークフローを作成
  - II. データフォームの作成
  - III. タスクの担当者割り当て
  - IV. 分岐の条件設定
  - V. ワークフローのデプロイ
5. flowable-taskで動かしてみよう！

# 1. 動作環境の準備(1/2)

- Flowable資材の入手

- コミュニティサイトよりダウンロード

<<https://www.flowable.org/>>の「Download Flowable v6.3.1」をクリックしFlowable資材(zipファイル)をダウンロード。

※2018. 5.31時点で、「flowable-6.3.1.zip」がダウンロードされる。

- 資材の展開

「flowable-6.3.1.zip」を任意の場所で解凍。

本日はこの中から、「flowable-6.3.1/wars」配下に配備されている、

- flowable-idm.war
- flowable-modeler.war
- flowable-task.war

の3つの資材を利用します。

# 1. 動作環境の準備(2/2)

- Flowable資材をTomcat上へ配備

Tomcatのwebapps配下に上記で取得した3つのwarファイルを配備しTomcat起動

- flowable-idm.war

Flowableが提供するユーザやグループ、権限管理を行うApp

- flowable-modeler.war

Flowableが提供するWebベースのモデリングApp

- flowable-task.war

Flowableが提供するWebベースのタスク管理App

1. 動作環境の準備
2. 環境の確認
3. flowable-idmによる作業
4. flowable-modelerによる作業
  - I. ワークフローを作成
  - II. データフォームの作成
  - III. タスクの担当者割り当て
  - IV. 分岐の条件設定
  - V. ワークフローのデプロイ
5. flowable-taskで動かしてみよう！



## 2. 環境の確認

Flowableの環境の正常性確認として以下のURLへアクセスしましょう！！

① <http://localhost:8080/flowable-idm/>

ユーザ : admin

パスワード : test

② <http://localhost:8080/flowable-modeler/>

③ <http://localhost:8080/flowable-task/>

まずはエラー無く画面が表示されればOKです。

1. 動作環境の準備
2. 環境の確認
3. flowable-idmによる作業
4. flowable-modelerによる作業
  - I. ワークフローを作成
  - II. データフォームの作成
  - III. タスクの担当者割り当て
  - IV. 分岐の条件設定
  - V. ワークフローのデプロイ
5. flowable-taskで動かしてみよう！

3. flowable-idmによる作業

4. flowable-modelerによる作業

動画で実際の画面を見ながら、  
操作を説明させていただきます。

1. 動作環境の準備
2. 環境の確認
3. flowable-idmによる作業
4. flowable-modelerによる作業
  - I. ワークフローを作成
  - II. データフォームの作成
  - III. タスクの担当者割り当て
  - IV. 分岐の条件設定
  - V. ワークフローのデプロイ
5. flowable-taskで動かしてみよう！

# 5. flowable-taskで動かしてみよう！

作成済みのプロセスを用いて実際にデモをさせていただきます。

## 1. Flowable-taskを用いてプロセスを開始してみよう！！

manager01でログインし開始処理

投入データ

- ・長さ:160、重さ:25
- ・長さ:161、重さ:25

member01でログインし開始処理

投入データ

- ・長さ:160、重さ:25
- ・長さ:160、重さ:26

プロセスを開始する人(今回の場合はmanager01、member01)や投入データ(今回の場合は荷物の3辺の長さと重さ)により、以降の処理を行える人や作業(フロー)が変わりますので確認してみてください。

# 5. flowable-taskで動かしてみよう！

## 2. プロセスの開始方法

1. まずは<http://localhost:8080/flowable-task/>にアクセス
2. 自身のアカウント(今回はmember01)にてログイン
3. 荷物発送手配APPを選択
4. 「Start a new process and then track its progress」をクリック
5. 荷物発送手配プロセスが開始され「発送地方選択画面」が表示されます。

# 5. flowable-taskで動かしてみよう！

## 3. タスクの確認、実行画面

① ログインユーザ

The screenshot displays the Flowable Task management interface in a web browser. The browser's address bar shows the URL `localhost:8080/flowable-task/workflow/#/apps/transport_app/tasks`. The interface includes a navigation bar with 'Flowable', 'Tasks', 'Processes', and 'Cases'. A dropdown menu in the top right corner shows the user 'めんば ぜろいち'. The main content area is divided into two sections. On the left, a list of tasks is shown under the heading 'Showing your tasks, no filter applied'. Two tasks are listed, both titled '発送詳細情報の登録' (Registration of shipping details) and assigned to 'めんば ぜろいち'. On the right, a detailed view of a task is shown, titled '発送詳細情報の登録'. It includes fields for 'Assignee' (めんば ぜろいち), 'Due date' (No due date), and 'Part of process' (荷物発送手配プロセス - June 7th 018). Below these are input fields for 'Shipping destination', 'Shipping date', 'Total length of goods', and 'Weight of goods'. The interface also features 'Save' and 'Complete' buttons.

③ タスク

② 複数タスク  
のリスト

④ タスクの  
内容

# まとめ

今回、Flowableを用いて仕事の流れを定義し利用するサンプルを紹介しました。

Flowableを用いることにより、

- **決められたルールで確実に仕事を実施**

(引越便処理: manager01、宅配便処理: kyoto-group)

- **自身の都合にあわせて仕事を実施**

(自身のタスク一覧を元に処理を実施。残作業も明確でやり忘れも防止。)

- **現在の状態把握が可能**

(起票した仕事はどこまで進んでいるか？どこで滞留しているか？といった進捗状態の把握が可能。)

といったワークフロー(エンジン)にて実現できる内容を理解頂けたかと思います。

サークルや有志による情報回覧から、全社的な組織を跨ぐ領域でもアイデア次第で利用可能です。

ご自身のおかれている立場に合わせて活用頂ければ幸いです。



ご清聴ありがとうございました。

# 参考

動画で実施している操作をスライドに整理したものです。

設定の参考にして下さい。

1. 動作環境の準備
2. 環境の確認
3. flowable-idmによる作業
4. flowable-modelerによる作業
  - I. ワークフローを作成
  - II. データフォームの作成
  - III. タスクの担当者割り当て
  - IV. 分岐の条件設定
  - V. ワークフローのデプロイ
5. flowable-taskで動かしてみよう！

# 3. flowable-idmによる作業

## 1. ユーザ (Users) の作成

- 担当内のユーザ情報を登録します
- 今回のデモにおいては、課長1名、担当者2名の計3名のユーザ情報を登録します
- 登録内容は5項目 (User id、Email、Password、First name、Last name)
- 具体的な登録情報  
次ページの表を参照願います。

# 3. flowable-idmによる作業

	User id	Email	Password	First name	Last name
マネージャ	manager01	manager01@test.co.jp	manager01	まねじゃ	ゼロいち
担当者#1	member01	member01@test.co.jp	member01	めんば	ゼロいち
担当者#2	member02	member02@test.co.jp	member02	めんば	ゼロに

Flowable-idmによる登録情報

# 3. flowable-idmによる作業

## 2. グループ (Groups) の作成

- 担当内のユーザをまとめたグループを作成・登録します
- 今回のデモにおいては、担当者2名を1つのグループに登録します ※マネージャはグループに含めない
- 具体的な登録内容

Group id	: kyoto-group
Name	: 京都グループ
構成メンバ	: member01, member02

# 3. flowable-idmによる作業

## 3. 特権 (Privileges) の設定

- manager01を全ての特権ユーザとして設定
- kyoto-groupを「Access the workflow application」の特権ユーザとして設定

flowable-idmによる各種設定のまとめ

ユーザとグループ

メンバ

manager01

member01

member02

グループ

(kyoto-group)

特権 (権限) 設定

Access identity management application

Access admin application

Access modeler application

Access the workflow application kyoto-groupへ

Access the rest application

全ての権限をmanager01へ

# 4. flowable-modelerによる作業

## I. ワークフローを作成(1/3)

- ① 「業務開始」[「開始イベント」といいます]
- ② 「発送詳細情報の登録」[「ユーザタスク」といいます]
- ③ 「条件分岐」[「ゲートウェイ」といいます]
- ④ 「引越便発送手配&発送処理」[「ユーザタスク」といいます]
- ⑤ 「宅配便発送手配&発送処理」[「ユーザタスク」といいます]
- ⑥ 「条件分岐合流」[「ゲートウェイ」といいます]
- ⑦ 「業務終了」[「終了イベント」といいます]

※上記のそれぞれの部品はフロー部品を用いて接続します。  
「シーケンスフロー」といいます

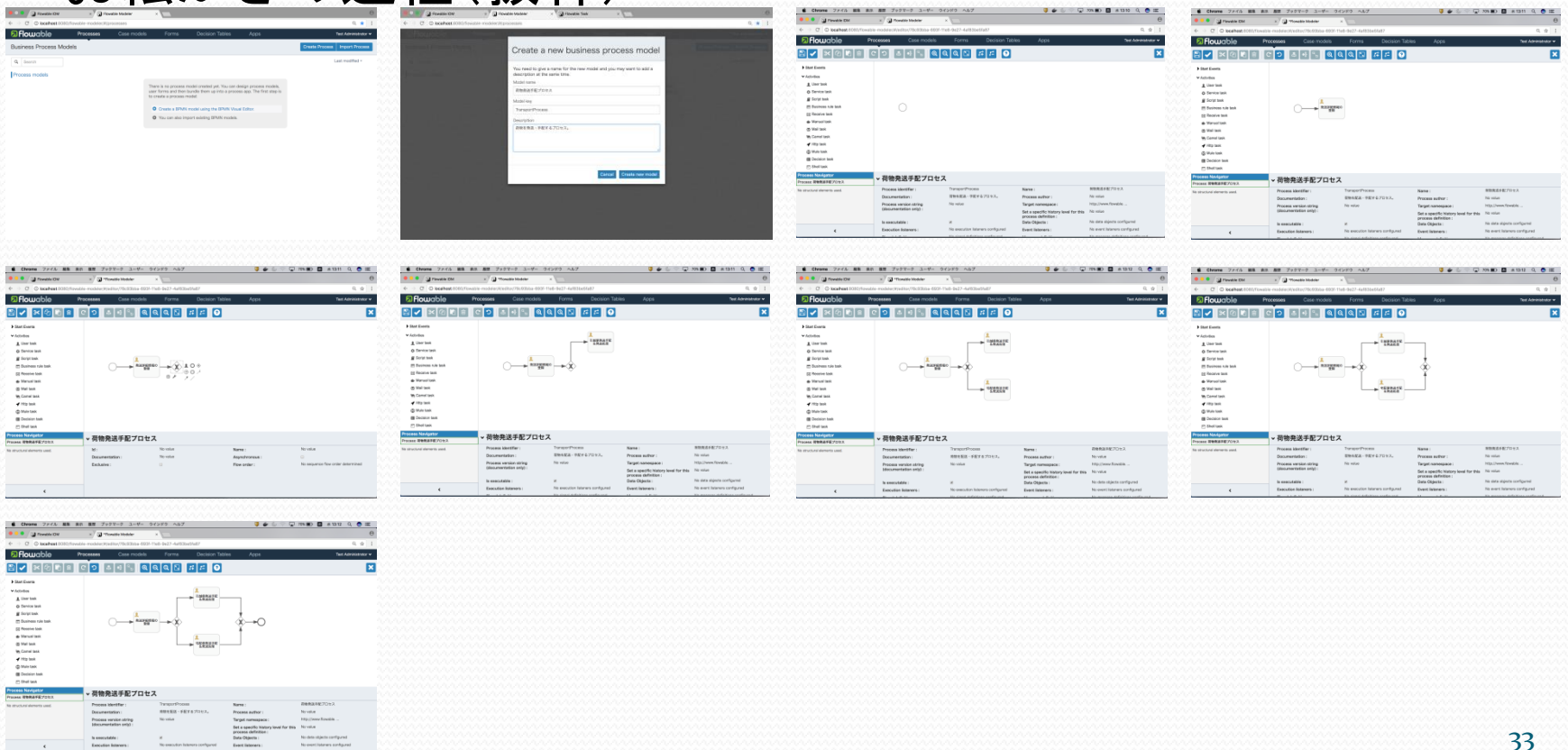
Model name : 荷物発送手配プロセス  
Model key : TransportProcess  
Description : 荷物を発送手配するプロセス



# 4. flowable-modelerによる作業

## I. ワークフローを作成(2/3)

### お絵かきの過程(抜粋)



# 4. flowable-modelerによる作業

## I. ワークフローを作成(3/3)

### お絵かきの完成イメージ

The screenshot displays the Flowable Modeler web application. The main workspace shows a BPMN diagram for a shipping process. The process starts with a start event, followed by an activity '発送詳細情報の登録' (Registration of shipping details). This leads to an XOR gateway with the condition '重量 > 160 || 重量 > 25'. Two paths emerge: one for '引越便発送手配 & 発送処理' (Relocation shipping arrangement & processing) and another for '宅配便発送手配 & 発送処理' (Parcel shipping arrangement & processing). Both paths merge at an XOR gateway, which then leads to an end event.

Process Navigator

Process: 荷物発送手配プロセス

No structural elements used.

▼ 荷物発送手配プロセス

Process identifier :	TransportProcess	Name :	荷物発送手配プロセス
Documentation :	荷物を配送・手配するプロセス。	Process author :	No value
Process version string (documentation only) :	No value	Target namespace :	http://www.flowable. ...
Is executable :	<input checked="" type="checkbox"/>	Set a specific history level for this process definition :	No value
Execution listeners :	No execution listeners configured	Data Objects :	No data objects configured
		Event listeners :	No event listeners configured
			No message definitions configured

# 4. flowable-modelerによる作業

## II. データフォームの作成(1/4)

「業務開始」に、**発送先地方の入力**を行うフォームを準備しよう

Form name : 発送先地方選択フォーム  
Form key : region  
Description : 発送先の地方を選択するフォーム

### Dropdown部品設定

Label : 発送先地方を選択してください。  
Id : select\_region  
Option : 「九州・沖縄」、「中国」、「四国」、「近畿」、「中部」、「関東」、「東北」、「北海道」

# 4. flowable-modelerによる作業

## II. データフォームの作成(2/4)

「発送詳細情報の登録」に、

- ・ 発送先県名
- ・ 発送予定日
- ・ 荷物の3辺の長さの合計[cm]
- ・ 荷物の重さ[kg]

Form name: 発送詳細情報入力フォーム  
Form key : transportDetails  
Description: 発送する荷物の詳細情報  
入力フォーム

の入力を行うフォームを準備しよう

### Text部品設定

Label : 発送先の都道府県名を入力してください。  
Id : pref

### Number部品設定①

Label : 荷物の3辺の長さの合計を入力してください。[cm]  
Id : sizecm

### Date部品設定

Label : 発送予定日を入力してください。  
Id : shipment\_date

### Number部品設定②

Label : 荷物の重さを入力してください[kg]  
Id : weightkg

# 4. flowable-modelerによる作業

## II. データフォームの作成(3/4)

「引越便発送手配&発送処理」にこれまでの既入力情報、

- ・ 発送先地方
- ・ 発送県名
- ・ 発送予定日
- ・ 荷物の3辺の長さの合計
- ・ 荷物の重さ

の情報を表示するとともに、

- ・ 引越便荷物追跡番号

の入力を行うフォームを準備しよう

Form name : 引越便荷物追跡番号登録フォーム

Form key : moving\_no

Description : 引越便の荷物追跡番号を登録するフォーム

Expression部品設定

Expression :

Text部品設定

Label : 「引越便荷物追跡番号」

Id : moving\_tracking\_no

# 4. flowable-modelerによる作業

## II. データフォームの作成(4/4)

「宅配便発送手配&発送処理」にこれまでの既入力情報、

- ・ 発送先地方
- ・ 発送県名
- ・ 発送予定日
- ・ 荷物の3辺の長さの合計
- ・ 荷物の重さ

の情報を表示するとともに、

- ・ 宅配便便荷物追跡番号

の入力を行うフォームを準備しよう

Form name : 宅配便荷物追跡番号登録フォーム  
Form key : delivery\_no  
Description : 宅配便の荷物追跡番号を登録するフォーム

Expression部品設定

Expression :

Text部品設定

Label : 「宅配便荷物追跡番号」

Id : delivery\_tracking\_no

# 4. flowable-modelerによる作業

## Ⅲ. タスクの担当者割り当て(1/2)

どのような指定が可能??

- ① プロセスを開始した人に割り当て (Assigned to process initiator)
- ② 単一ユーザに割り当て (Assigned to single user)
- ③ 複数ユーザのリストに割り当て (Candidate users)
- ④ 候補グループに割り当て (Candidate groups)

# 4. flowable-modelerによる作業

## Ⅲ. タスクの担当者割り当て(2/2)

それでは実際に割り当てます

The screenshot shows the Flowable Modeler interface with a BPMN diagram for a shipping process. The diagram includes a start event, a task '発送詳細情報の登録', a gateway 'Ea > 160 川重 > 25', and two tasks: '引越便発送手配 & 発送処理' and '宅配便発送手配 & 発送処理'. Three red callout boxes provide details on task assignments:

- Single User Assignment:** 単一ユーザに割り当て (Assigned to single user) 今回は"manager01"へ割り当て (Assigned to "manager01" this time)
- Process Initiator Assignment:** プロセスを開始した人に割り当て (Assigned to process initiator)
- Candidate Group Assignment:** 候補グループに割り当て (Candidate Groups) 今回は"kyoto-group"へ割り当て (Assigned to "kyoto-group" this time)

At the bottom, the 'Process Navigator' shows the '荷物発送手配プロセス' (Cargo Shipping Process) with the following details:

Process identifier :	TransportProcess	Name :	
Documentation :	荷物を配送・手配するプロセス。	Process author :	No value
Process version string (documentation only) :	No value	Target namespace :	http://www.flowable. ...
Is executable :	<input checked="" type="checkbox"/>	Set a specific history level for this process definition :	No value
Execution listeners :	No execution listeners configured	Data Objects :	No data objects configured
	No signal definitions configured	Event listeners :	No event listeners configured
			No message definitions configured



# 4. flowable-modelerによる作業

## IV. 分岐の条件設定(1/2)

### 条件

「3辺の長さ(sizecm)が160[cm]より大きい」

Or

「重さ(weightkg)が25[kg]より大きい」

The screenshot shows the Flowable Modeler web application. The main canvas displays a BPMN diagram with a gateway (diamond) and a task (rounded rectangle). The gateway has a flow condition: `長さ > 160 || 重さ > 25`. The task is labeled '配送詳細情報の変更'. The 'Process Navigator' panel at the bottom shows the selected flow condition: `長さ > 160 || 重さ > 25`. The 'Flow condition' field is highlighted with a red arrow pointing to the text 'Flow condition: 長さ > 160 || 重さ > 25'.

Process Navigator			
Process: 荷物発送手配プロセス			
No structural elements used.			
Id:	No value	Name:	長さ > 160    重さ > 25
Documentation:	No value	Flow condition:	<code>長さ &gt; 160    重さ &gt; 25</code>
Execution listeners:	No execution listeners configured	Default flow:	<input type="checkbox"/>
Skip expression:	No value		

① 条件を設定したい線(シーケンスフロー)をクリック(選択)する

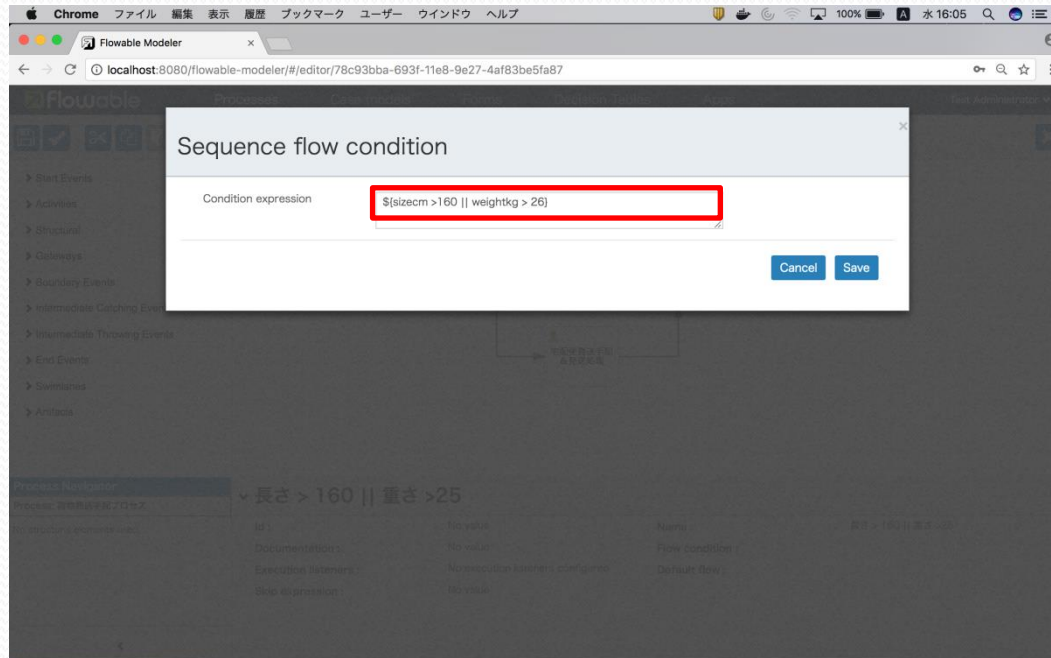
② Flow condition をクリックする(選択)する。

# 4. flowable-modelerによる作業

## IV. 分岐の条件設定(2/2)

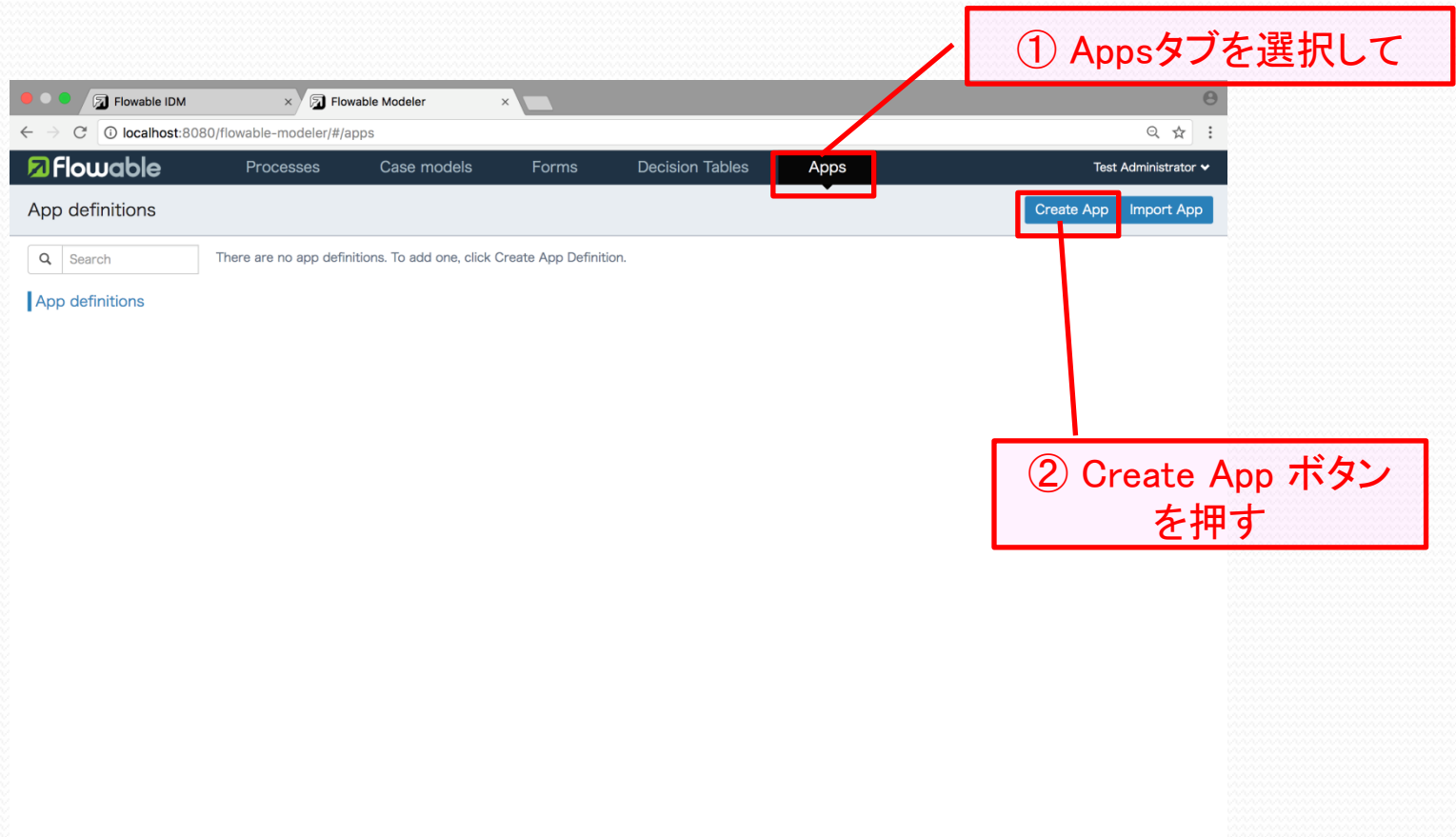
判断式を記述する。

$\{\text{sizecm} > 160 \parallel \text{weightkg} > 25\}$



# 4. flowable-modelerによる作業

## V. ワークフローのデプロイ(1/6)



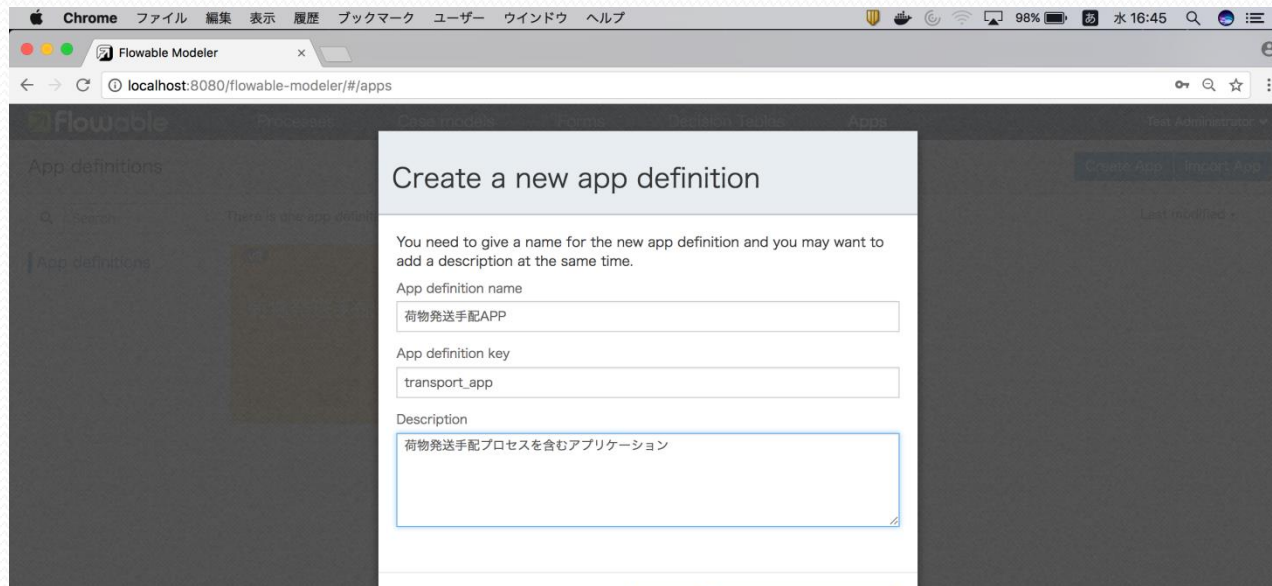
The screenshot shows the Flowable Modeler web application interface. The browser address bar displays 'localhost:8080/flowable-modeler/#/apps'. The navigation menu includes 'Processes', 'Case models', 'Forms', 'Decision Tables', 'Apps', and 'Test Administrator'. The 'Apps' tab is selected and highlighted with a red box. Below the navigation menu, the 'App definitions' section is visible, containing a search bar and the text 'There are no app definitions. To add one, click Create App Definition.' The 'Create App' button is highlighted with a red box. Two red callout boxes with arrows point to these elements: the first points to the 'Apps' tab with the text '① Appsタブを選択して', and the second points to the 'Create App' button with the text '② Create App ボタンを押す'.

① Appsタブを選択して

② Create App ボタンを押す

# 4. flowable-modelerによる作業

## V. ワークフローのデプロイ(2/6) アプリケーション情報を入力する。



App definition name : 荷物発送手配APP

App denfinition key : transport\_app

Description : 荷物発送手配プロセスを含むアプリケーション

# 4. flowable-modelerによる作業

## V. ワークフローのデプロイ(3/6)

Icon, Themeはご自由に！！

「Edit included models」ボタンをおして、既に作成しているプロセスモデル、

Model name	: 荷物発送手配プロセス
Model key	: TransportProcess
Description	: 荷物を発送手配するプロセス

を選択し、動作対象プロセスモデルとして設定する。

# 4. flowable-modelerによる作業

## V. ワークフローのデプロイ(4/6)

### APP設定画面

The screenshot shows the Flowable Modeler interface in a Chrome browser. The page title is "App definition details: 荷物発送手配APP". The interface includes a navigation bar with "Processes", "Case models", "Forms", "Decision Tables", "Apps", and "Test Administrator". The "Apps" tab is active, showing the app definition details for "荷物発送手配APP".

The main content area is divided into two columns. The left column contains a "PREVIEW" section with a placeholder image for the app icon and the text "荷物発送手配APP". The right column contains configuration options for the app, including "Icon" (with a "Change icon..." button), "Theme" (with a "Change theme..." button), "Groups access, separated by commas", and "Users access, separated by commas".

Below the configuration options, there is a section titled "Models included in the app definition" with an "Edit included models" button. Underneath, there are tabs for "BPMN models" and "CMMN models". The "BPMN models" tab is selected, showing a BPMN diagram for "荷物発送手配プロセス" (Cargo Shipping Process) with a version indicator "v1".

# 4. flowable-modelerによる作業

## V. ワークフローのデプロイ(5/6)

利用可能なように動作環境へPublishする。

The screenshot shows the Flowable Modeler web interface in a Chrome browser. The browser tabs are 'Flowable IDM' and 'Flowable Modeler'. The address bar shows 'localhost:8080/flowable-modeler/#/apps/b0ad90a3-6943-11e8-9e27-4af83be5fa87'. The main navigation bar includes 'Processes', 'Case models', 'Forms', 'Decision Tables', and 'Apps'. The 'Apps' menu is expanded, showing a list of applications. The first application is '荷物発送手配APP' (Cargo Shipping Handling APP), created by 'admin' and last updated 'Today at 1:40 PM'. The application details page is displayed, showing a preview of the app with the title '荷物発送手配APP' and a BPMN diagram. The 'Publish' button is highlighted with a red box, and a red callout box points to it with the text '「Publish」ボタンを押す' (Press the 'Publish' button).

# 4. flowable-modelerによる作業

## V. ワークフローのデプロイ(6/6)

