



- 1. 改訂情報
- 2. はじめに
 - 2.1. 本書の目的
 - 2.2. 対象読者
 - 2.3. 環境について
 - 2.4. サンプルコードおよび画面について
 - 2.5. 本書の構成
 - 2.6. チュートリアル実行ユーザ
- 3. 概要
 - 3.1. ローコード開発とは
 - 3.2. intra-martにおけるローコード開発ツール
 - 3.2.1. IM-LogicDesigner
 - 3.2.2. IM-BloomMaker
 - 3.2.3. IM-FormaDesigner
 - 3.2.4. IM-BIS
 - 3.2.5. IM-Spreadsheet
 - 3.2.6. ViewCreator
 - 3.2.7. Accel Studio
 - 3.3. Accel-Mart Quick
- 4. チュートリアル
 - 4.1. 概要
 - 4.1.1. マスタメンテナンステンプレートを利用した簡易的なToDoアプリケーション
 - 4.1.2. ワークフローテンプレートを利用した日報アプリケーション
 - 4.1.3. テンプレートを利用せずに作成するアンケートアプリケーション
 - 4.1.4. IM-Sign と連携した電子サインアプリケーション
 - 4.2. シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する
 - 4.2.1. 概要
 - 4.2.2. テンプレートからアプリケーションのベースを作る
 - 4.2.3. 登録、更新処理の修正
 - 4.2.4. 一覧での検索処理の確認
 - 4.2.5. 登録/更新/参照画面の修正
 - 4.2.6. 一覧画面の修正
 - 4.2.7. メニューの設定
 - 4.3. ワークフローを利用してアプリケーションを作成する
 - 4.3.1. 概要
 - 4.3.2. テンプレートからアプリケーションのベースを作る
 - 4.3.3. ワークフロー画面の修正
 - 4.3.4. ワークフローの修正
 - 4.3.5. アプリケーションの認可設定
 - 4.4. ゼロからアプリケーションを作成する
 - 4.4.1. 概要
 - 4.4.2. アプリケーションの基本情報を設定する
 - 4.4.3. アンケートアプリケーションの業務ロジックの作成
 - 4.4.4. 業務ロジックのURL設定
 - 4.4.5. 回答画面の作成
 - 4.4.6. 集計画面の作成
 - 4.4.7. 画面URLの設定
 - 4.4.8. メニューの設定
 - 4.5. IM-Sign と連携した電子サインアプリケーション
 - 4.5.1. 概要
 - 4.5.2. 事前準備

- 4.5.3. テンプレートからアプリケーションのベースを作る
- 4.5.4. IM-Sign 連携テンプレートの使い方
- 5. チュートリアル（応用）
 - 5.1. 概要
 - 5.1.1. アンケートアプリケーションに追加機能を作成する
 - 5.2. アンケートアプリケーションに機能を追加する
 - 5.2.1. 概要
 - 5.2.2. アンケート一覧、登録処理の業務ロジックの作成
 - 5.2.3. 業務ロジックのURL設定
 - 5.2.4. アンケート一覧、登録画面の作成
 - 5.2.5. 画面URLの設定

変更年月日	変更内容
2021-10-01	初版
2021-11-10	第2版 下記を追加・変更しました <ul style="list-style-type: none">「IM-Sign と連携した電子サインアプリケーション」を追加しました。「チュートリアル (応用)」を追加しました。「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」の以下のページに解説動画を追加しました。<ul style="list-style-type: none">「テンプレートからアプリケーションのベースを作る」「登録/更新/参照画面の修正」
2021-12-01	第3版 下記を追加・変更しました <ul style="list-style-type: none">「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」 - 「メニューの設定」にintra-mart Accel Platformでの説明を追記しました。「ゼロからアプリケーションを作成する」 - 「メニューの設定」にintra-mart Accel Platformでの説明を追記しました。
2022-01-31	第4版 一部キャプチャおよび説明を修正しました。
2022-03-31	第5版 下記を追加・変更しました <ul style="list-style-type: none">一部キャプチャおよび説明を修正しました。「はじめに」に本書チュートリアルは「テナント管理者」を付与したユーザで行う必要がある旨を追記しました。「ToDoアプリケーションに機能を追加する」 - 「「担当者」検索処理の追加」の修正および追記しました。「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」の以下のページに解説動画を追加しました。<ul style="list-style-type: none">「登録、更新処理の修正」「一覧での検索処理の確認」「一覧画面の修正」
2022-06-01	第6版 下記を追加・変更しました <ul style="list-style-type: none">一部キャプチャおよび説明を修正しました。「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」 - 「メニューの設定」にアプリケーションの権限設定の説明を追記しました。「ワークフローを利用してアプリケーションを作成する」 - 「アプリケーションの認可設定」を追加しました。「ゼロからアプリケーションを作成する」 - 「メニューの設定」にアプリケーションの権限設定の説明を追記しました。
2022-09-30	第7版 下記を追加・変更しました <ul style="list-style-type: none">「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」の解説動画を追加、修正しました。「ゼロからアプリケーションを作成する」に解説動画を追加しました。
2022-12-01	第8版 下記を変更しました <ul style="list-style-type: none">「Adobe Sign」の製品名の変更に伴い、ドキュメント内の製品名記載を「Acrobat Sign」に修正しました。
2022-12-21	第9版 下記を変更しました <ul style="list-style-type: none">「概要」の「Accel-Mart Quick」の記述を変更しました。「Accel Studio」の機能追加および仕様変更に伴う注意事項の追加しました。

変更年月日	変更内容
2023-04-01	<p>第10版 下記を変更しました</p> <ul style="list-style-type: none">「概要」の「intra-martにおけるローコード開発ツール」の記述を変更しました。「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」に以下の追記、修正を行いました。<ul style="list-style-type: none">「テンプレートからアプリケーションのベースを作る」をデータリポジトリを利用した方法に修正しました。「登録、更新処理の修正」と「一覧での検索処理の確認」をデータリポジトリを利用した方法に修正しました。「メニューの設定」の権限設定画面を最新化しました。「ワークフローを利用してアプリケーションを作成する」に以下の追記、修正を行いました。<ul style="list-style-type: none">「テンプレートからアプリケーションのベースを作る」をデータリポジトリを利用した方法に修正しました。「ワークフローの修正」のワークフロー設定画面を最新化しました。「アプリケーションの認可設定」の権限設定画面を最新化しました。「ゼロからアプリケーションを作成する」に以下の追記、修正を行いました。<ul style="list-style-type: none">「メニューの設定」の権限設定画面を最新化しました。
2023-05-31	<p>第11版 下記を変更しました</p> <ul style="list-style-type: none">「チュートリアル」全体の文言およびキャプチャを見直しました。「はじめに」の「環境について」の記述を変更しました。「回答画面の作成」の「変数の設定」の記述を修正しました。「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」 - 「メニューの設定」にAccel-Mart Quick での設定方法を追加しました。「ゼロからアプリケーションを作成する」 - 「メニューの設定」にAccel-Mart Quick での設定方法を追加しました。
2023-06-30	<p>第12版 下記を変更しました</p> <ul style="list-style-type: none">「チュートリアル」全体の文言およびキャプチャを見直しました。「IM-Sign と連携した電子サインアプリケーション」の記述を最新化しました。
2023-10-31	<p>第13版 下記を変更しました</p> <ul style="list-style-type: none">「概要」の「intra-martにおけるローコード開発ツール」の表にカスタマーサクセスライセンス向けの Advance Edition に関する記述を追加しました。「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」と「ワークフローを利用してアプリケーションを作成する」のシナリオを一部変更し、キャプチャの最新化を実施しました。メンテナンスのため「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」の解説動画を非公開にしました。メンテナンスのため「チュートリアル (応用)」の内容の一部を非公開にしました。「はじめに」のチュートリアルがベースとしているバージョンの情報を更新しました。
2023-12-22	<p>第14版 下記を変更しました</p> <ul style="list-style-type: none">「シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する」の解説動画を公開しました。
2024-04-01	<p>第15版 下記を変更しました</p> <ul style="list-style-type: none">ドキュメント全体の画像を最新化し、それに合わせて文言に軽微な修正を加えました。

本書の目的

本書は、intra-martのローコード開発ツールを利用してアプリケーションの開発を始める開発者の皆様の支援を目的としたドキュメントです。

対象読者

本書は次のような開発者を対象としています。

- intra-martのローコード開発ツールを利用して初めてアプリケーション開発を行う方
- 以下の技術について、基本的な知識をお持ちの方
 - JavaScript
 - SQL

環境について

本チュートリアルでは、Accel Studioを利用した統合的なローコード開発を行います。

本書はデータベースに PostgreSQL を利用している前提で記述しております。

利用するデータベースに合わせて適宜 SQL を変更してください。

本チュートリアルはintra-mart Accel Platform 2023 Spring(Gerbera)をベースに作成しています。

※「[シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する](#)」と「[ワークフローを利用してアプリケーションを作成する](#)」はintra-mart Accel Platform 2023 Autumn(Hollyhock)をベースにしています。

本チュートリアルは以下のアプリケーションモジュールを含んだ環境を標準とし、機能間の遷移順序を記述しています。

- Accel Studio
- IM-BloomMaker for Accel Platform
- IM-Sign for Accel Platform

サンプルコードおよび画面について

本書に掲載されているサンプルコードは可読性を重視しており、性能や保守といった観点において必ずしも適切な実装ではありません。実際の開発で本チュートリアルのサンプルコードを参考にする場合は、上記の点について十分に注意してください。



注意

本書の画面キャプチャには開発中の製品画面も含まれます。

実際にリリースされている製品や製品のバージョンによっては本書のキャプチャ画像と画面の見た目が異なる場合があります。

本書の構成

本書は以下のような構成で記述しています。

- [概要](#)
本書およびintra-martにおけるローコード開発ツールについて説明します。
- [チュートリアル](#)
ローコード開発ツールを利用した開発の流れを説明します。
- [チュートリアル \(応用\)](#)
「[チュートリアル](#)」で作成した基本的なアプリケーションの修正や機能追加を行います。

本書のチュートリアルは「テナント管理者」ロールを付与したユーザで行ってください。



コラム

Accel-Mart Quick をご利用の場合は、「Accel-Mart Quick アプリケーション管理者」ロールを付与したユーザを利用してください。



コラム

ロールの付与については、「[IM-共通マスタ 管理者操作ガイド](#)」 - 「ユーザ」 - 「[ユーザの基本設定](#)」を参照してください。

intra-martにおける、ローコード開発ツールの概要についてです。

ローコード開発とは

ローコード開発とは、最小限のソースコードでアプリケーションを開発する手法です。

基本的にはドラッグ&ドロップでの直感的操作で開発を行っていきます。

それにより、プロフェッショナルな開発者の開発生産性もさらに高まり、効率よくスピーディにアプリケーション開発を行うことができます。

intra-martにおけるローコード開発ツール

intra-martにおけるローコード開発は、intra-mart Accel Platform 上で動作する様々な機能（認証/認可、業務プロセス、ワークフロー等）を組み合わせた開発を可能とします。

例えば、業務ロジックを IM-LogicDesigner で、画面を IM-BloomMaker で作成すると、ブラウザ上の操作のみでintra-martの様々な機能を組み合わせたアプリケーションを作成できます。

intra-martが提供しているローコード開発ツールの種類、および機能の利用可能環境の組み合わせは以下の通りです。

目的	製品	スタンダード	アドバンス	エンタープライズ	Accel-Mart Plus	[CSL] ベーシック	[CSL] アドバンス	[CSL] プロフェッショナル
ビジネスロジックの作成	IM-LogicDesigner	○	○	○	○	○	○	○
表やグラフの作成	ViewCreator	○	○	○	○	○	○	○
複雑な画面の作成	IM-BloomMaker			○	○		○	○
Webフォーム画面の作成	IM-FormaDesigner			○	○		○	○
統合的な簡易アプリケーションの作成	IM-BIS			○	○		○	○
統合的なローコードアプリケーション開発	Accel Studio			○	○		○	○
Excelのような表の活用、計算	IM-Spreadsheet				○	○	○	○

IM-Spreadsheet をintra-mart Accel Platformで利用する場合は、ドメイン単位でのライセンスで提供しています。

詳細は「[IM-Spreadsheet for Accel Platform リリースノート](#)」を参照してください。

[CSL] は、カスタマーサクセスライセンス向けのエディションであることを表しています。

IM-LogicDesigner

IM-LogicDesigner は、intra-mart上で動作するビジネスロジックをWebブラウザでの操作で簡単に作成できる、ローコード開発ツールです。

作成したビジネスロジックは IM-BloomMaker やIM-BPMのビジネスロジックとして利用したり、ジョブスケジューラ機能で定時的に実行することなどが可能です。

それによりローコード開発によるビジネスソリューションの構築に寄与します。

「IM-LogicDesigner」の詳細については、「[IM-LogicDesigner チュートリアルガイド](#)」を参照してください。

IM-BloomMaker

IM-BloomMaker は、アプリケーション画面をWebブラウザ上で作成できる、画面作成に特化したローコードのビジュアル開発ツールです。

入力フォームやレイアウト設定等の豊富なコンポーネントがあり、コンポーネントをドラッグ&ドロップで配置するだけで、アプリケーション画面を誰でも、素早く、簡単に作成できます。

さらに、画面操作に合わせたアクション処理、REST APIを利用しての外部プログラム呼び出し、JSON形式での変数の入出力設定などができるため、高度な要件に対応した画面も作成できます。

「IM-BloomMaker」の詳細については、「[IM-BloomMaker for Accel Platform チュートリアルガイド](#)」を参照してください。

IM-FormaDesigner

IM-FormaDesigner は、HTML、JavaScript、XMLなどの知識がなくても、直観的な操作でデータの登録、更新、参照を行うようなアプリケーションを作成できるローコード開発ツールです。

テキストボックスやフォームボタン等の基本的な操作のパーツをはじめ、入力必須チェックや文字数チェックなどのWebアプリケーション画面の動作について、画面上で設定を行うだけで実現できます。

さらに、作成した画面の構成に基づいて、データを格納するテーブルやデータの登録、参照の処理が作成されます。

IM-FormaDesigner は特に IM-Workflow と連携する機能が豊富なため、ワークフロー処理の電子化をノンコーディングで実現できます。

「IM-FormaDesigner」の詳細については、「[IM-FormaDesigner 仕様書](#)」を参照してください。

IM-BIS

IM-BIS は、Webアプリケーション作成ツールである IM-FormaDesigner と、ワークフローモジュールである IM-Workflow を統合するためのツールです。

IM-Workflow と IM-FormaDesigner でワークフローアプリケーションを作成する場合、画面作成や各種設定をたくさん行う必要があります。

IM-BIS を利用することで、IM-Workflow と IM-FormaDesigner で行う設定を IM-BIS の画面上で行えます。そのため、簡易的なワークフローアプリケーションの作成がより簡潔かつスピーディに行えます。

「IM-BIS」の詳細については、「[IM-BIS 仕様書](#)」を参照してください。

IM-Spreadsheet

IM-Spreadsheet はintra-mart上でExcelのようなUIを実現するためのスプレッドシート機能です。

グレーシティ株式会社が開発、提供しているJavaScriptライブラリのSpreadJSを用いており、intra-mart上で、使い慣れたExcelと同じ感覚でスプレッドシートの編集・管理が可能です。

「IM-Spreadsheet」の詳細については、「[IM-Spreadsheet for Accel Platform リリースノート](#)」を参照してください。

ViewCreator

ViewCreator はドラッグ&ドロップ等での直感的操作によって、intra-mart上のデータを表やグラフの形式で参照できるようにする開発ツールです。

参照するデータは、データベースに登録されたデータをはじめ、ログファイルやCSVのようなテキストファイル、IM-LogicDesignerの実行結果なども扱えます。

「ViewCreator」の詳細については、「[ViewCreator 管理者操作ガイド](#)」を参照してください。

Accel Studio

Accel Studioはintra-mart上で動作するアプリケーションを統合的に開発、管理するためのツールです。

様々なユースケースに合わせたテンプレートを用意されており、アプリケーションに必要な画面や処理、データベースのテーブル等が自動で生成され、スムーズに開発を始められます。

画面や処理は IM-BloomMaker や IM-LogicDesigner を利用して作成されます。

作成したアプリケーションに新たな処理や画面を追加したり、公開/非公開の設定をするなど、統合的な管理が行えます。

また、アプリケーション単位でのエクスポート/インポートも可能です。

Accel-Mart Quick

Accel-Mart Quickはintra-martの各種ツールをよりシンプルに利用して業務用アプリケーションを開発できるクラウドサービスです。

テンプレートを利用したローコードアプリケーション開発も可能なため、本チュートリアルもご活用いただけます。

Accel-Mart Quickで利用可能な機能の詳細は、「[Accel-Mart Quick サービス仕様書](#)」の「[提供機能一覧](#)」を参照してください。

概要

本チュートリアルでは、以下のアプリケーションを作成します。

マスタメンテナンステンプレートを利用した簡易的なToDoアプリケーション

アプリケーション作成の「マスタメンテナンス」テンプレートを利用して、簡易的なToDoアプリケーションを作成します。作成するアプリケーションは、カテゴリや期限日などを設定したToDoの登録、更新、削除、一覧表示が行えます。

この章では、以下のような事が学べます。

- 「マスタメンテナンス」テンプレートを利用したアプリケーション作成の流れ
- テンプレートから作成したアプリケーションに独自の機能を追加する方法

ワークフローテンプレートを利用した日報アプリケーション

アプリケーション作成の「シンプルなワークフロー」テンプレートを利用して、簡易的な日報アプリケーションを作成します。

この章では、以下のような事が学べます。

- 「シンプルなワークフロー」テンプレートを利用したアプリケーション作成の流れ
- アプリケーション管理でのワークフローの編集方法
- テンプレートから作成したアプリケーションのデータベース情報を変更する方法

テンプレートを利用せずに作成するアンケートアプリケーション

アプリケーション作成のテンプレートを利用せずに簡単なアンケートアプリケーションを作成します。

アプリケーション作成では、テンプレートを利用しない「空のアプリケーション」の作成も可能です。

「空のアプリケーション」という箱に、IM-BloomMaker で作成した画面と IM-LogicDesigner で作成した業務ロジックを組み合わせ設定し、アプリケーションを作成します。

この章では、以下のような事が学べます。

- 「空のアプリケーション」を利用したアプリケーション作成の流れ
- IM-LogicDesigner での業務ロジックの作成方法
- IM-BloomMaker での画面の作成方法
- 作成したアプリケーションを呼び出す方法

IM-Sign と連携した電子サインアプリケーション

「Adobe Acrobat Sign連携 標準テンプレート」を使用し、電子サインサービスとの連携を行うアプリケーションを作成します。

本テンプレートはAcrobat Signをご利用の方向けの電子サインアプリケーションです。

この章では、以下のような事が学べます。

- 「Adobe Acrobat Sign連携 標準テンプレート」を利用したアプリケーション作成の流れ
- 作成したアプリケーションを呼び出す方法
- テンプレートから作成したアプリケーションのワークフロー設定を変更する方法

シンプルな登録、更新、削除、一覧表示を行うアプリケーションを作成する

概要

本章では、ローコード開発により簡易的なToDoアプリケーションを作成します。

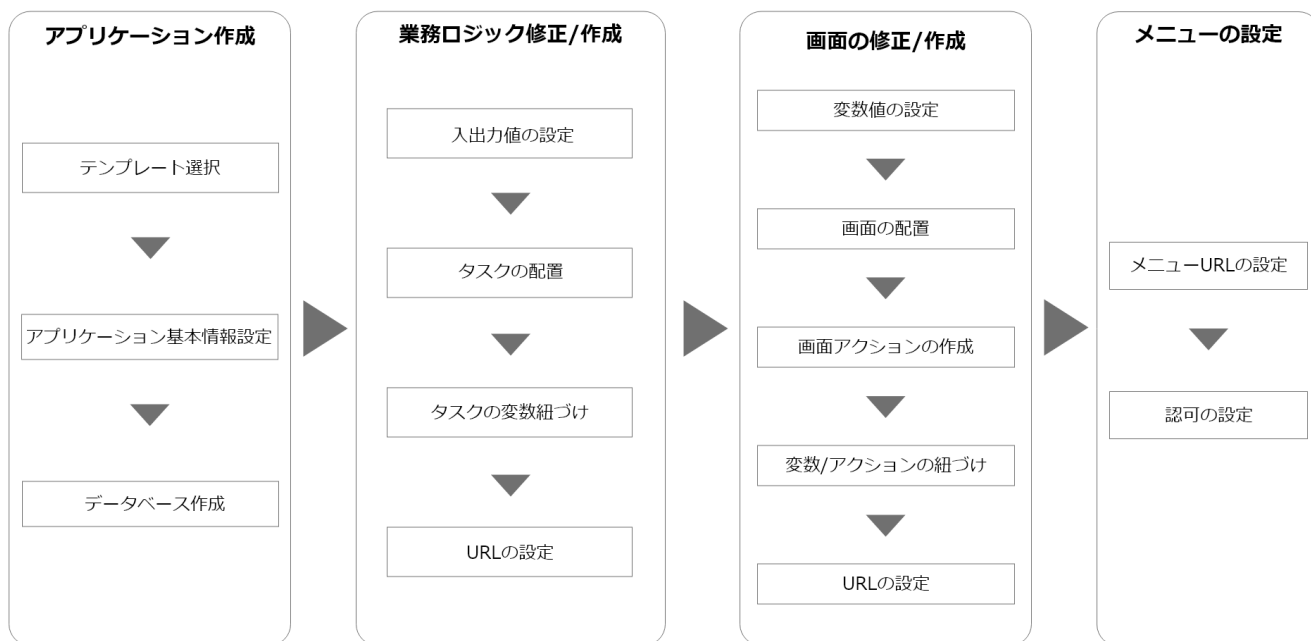
! 注意

本チュートリアルでは「[環境について](#)」に記載の環境を前提に、機能間の遷移順序を記述しています。詳しくは上記リンクからご確認ください。

作成の流れ

本チュートリアルは以下の流れでアプリケーションを作成します。

1. アプリケーション作成にてテンプレートからデータベース、処理、画面のベースを自動作成
2. IM-LogicDesignerにて、自動作成された処理の修正および追加処理の作成
3. IM-BloomMakerにて、自動作成された画面の修正
4. メニューの設定



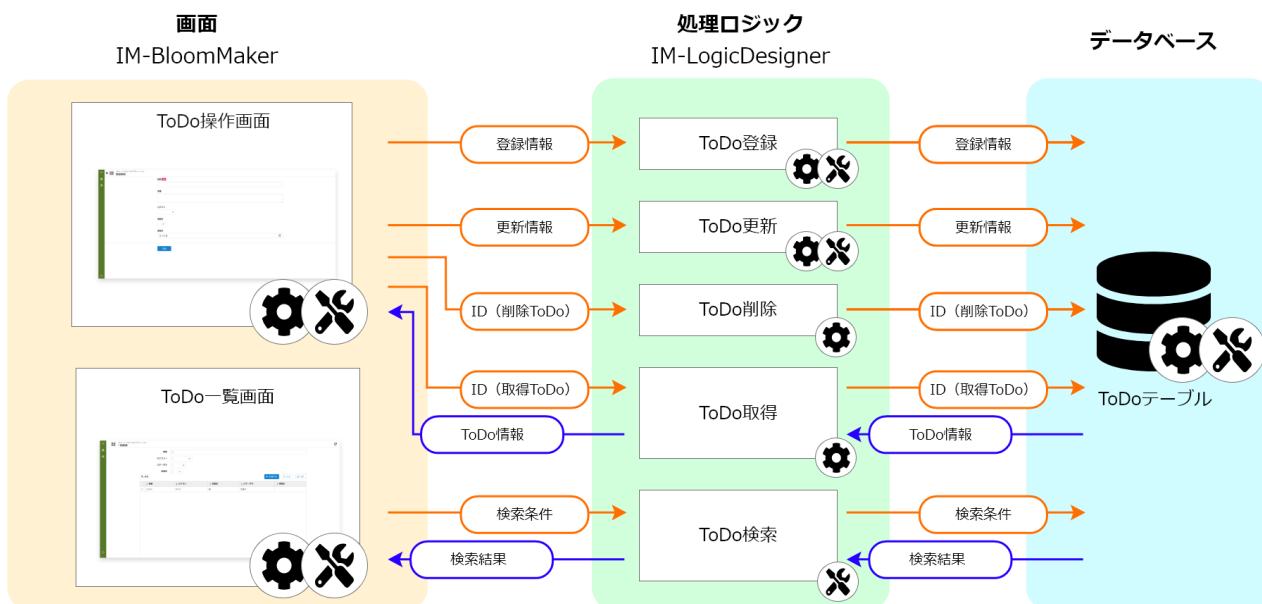
作成される処理や画面は以下の画像のようなイメージです。



チュートリアルで編集/作成



システムで自動生成



ToDoを操作する画面、一覧画面の2画面と、各画面の操作処理が自動で作成されます。
本チュートリアルでは、自動で作成された各画面、登録/更新処理の修正を行い、検索処理は新規作成します。

ToDoアプリケーションの仕様

作成するToDoアプリケーションは、カテゴリや重要度を指定して操作ユーザのToDoを登録、更新できます。
一覧画面での表示や絞り込みも行うことができ、一覧には操作ユーザのToDoのみが表示されます。
作成する画面と操作については、以下の通りです。

登録/参照/更新・削除画面

作成するToDoアプリケーションでは、登録、参照、更新、削除画面を1つの画面で実現します。
画面を呼び出すURLによって、画面の表示を切り替えます。

登録画面では以下の項目についての登録が行えます。

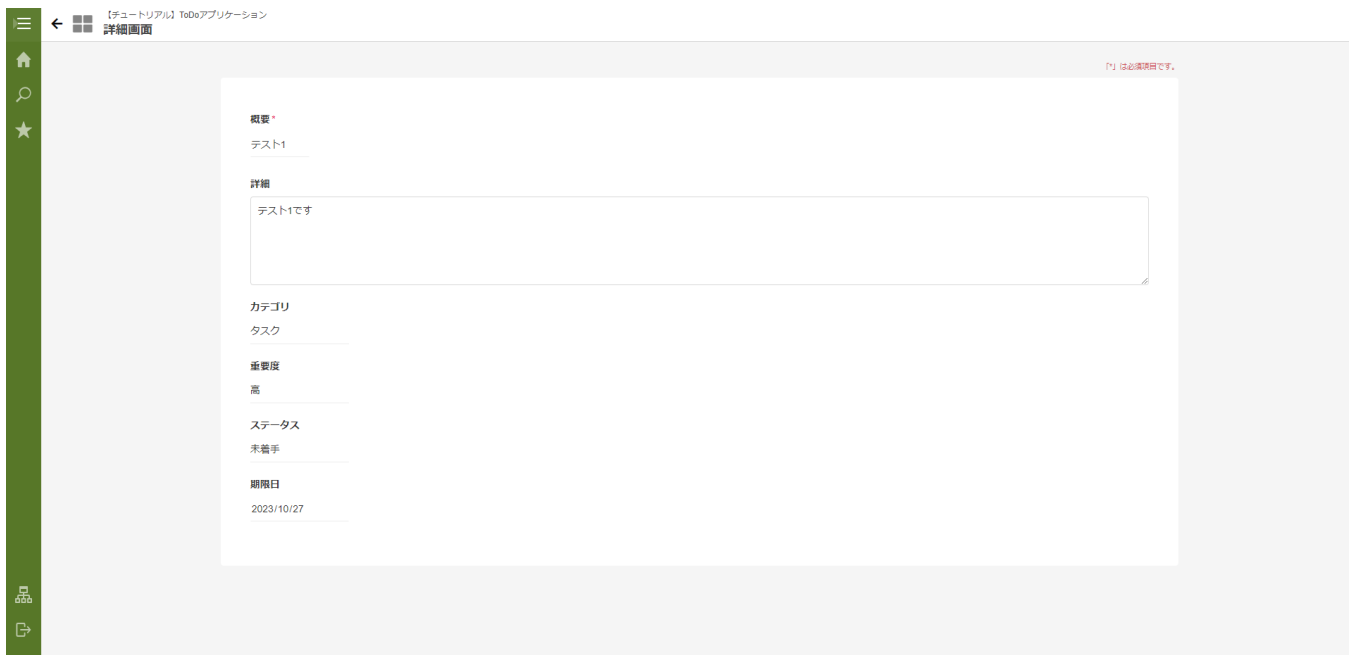
- 概要
タスクの概要を登録できます
- 詳細
タスクの詳細を登録できます
- カテゴリ
タスクのカテゴリを「タスク」、「スケジュール」、「移動」、「メモ」の中から選択して、登録できます
- 重要度
タスクの重要度を「高」、「中」、「低」の中から選択して、登録できます
- 期限日
タスクの期限日を日付で登録できます

更新・削除画面では、以下の項目の内容を更新でき、削除も可能です。

- 概要
タスクの概要を変更できます
- 詳細
タスクの詳細を変更できます
- カテゴリ
タスクのカテゴリを「タスク」、「スケジュール」、「移動」、「メモ」の中から選択して、変更できます
- ステータス
タスクのカテゴリを「未着手」、「着手中」、「完了」、「保留」の中から選択して、変更できます
- 重要度
タスクの重要度を「高」、「中」、「低」の中から選択して、登録できます
- 期限日
タスクの期限日の変更ができます

参照画面では、以下の項目の参照が行えます。編集は行えません。

- 概要
- 詳細
- カテゴリ
- 重要度
- 期限日

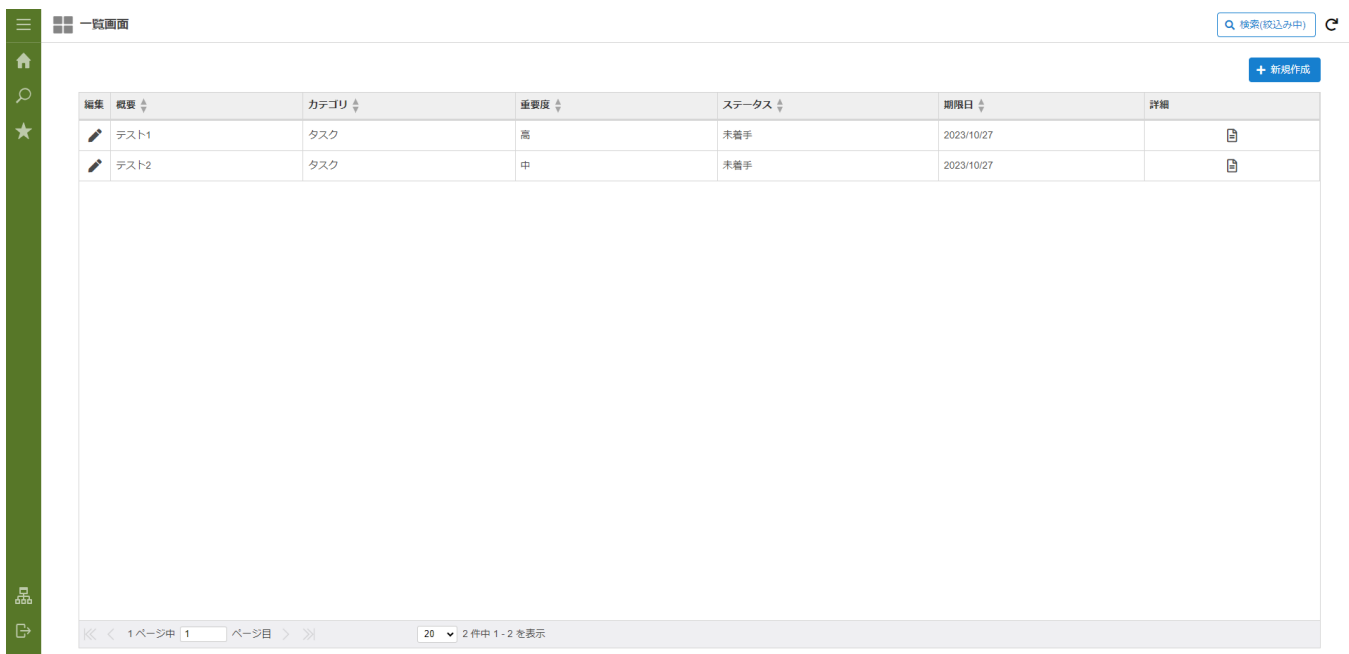


一覧画面

一覧画面では操作ユーザのToDoをテーブル形式で表示します。

概要のあいまい検索やカテゴリ、ステータスでの絞り込みも可能です。

また、一覧画面から新規登録画面へ遷移できます。



テンプレートからアプリケーションのベースを作る

アプリケーション作成のテンプレートを利用し、これから作成していくアプリケーションの元となるアプリケーションを生成します。本章では、「マスタメンテナンス」テンプレートを利用し、登録、更新などの処理をもったアプリケーションを生成します。

このページの手順について、以下の動画からも確認できます。併せてご利用ください。



注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

「マスタメンテナンス」テンプレートを利用したアプリケーションの生成


「サイトマップ」→「Accel Studio」→「アプリケーション作成」→「業務テンプレート一覧」の順に遷移します。
「ベーステンプレート」から「マスタメンテナンス」テンプレートを選択します。

「マスタメンテナンス」テンプレートを利用すると、扱うデータの定義に合わせたデータベースのテーブル、画面、処理を自動で生成できます。

「マスタメンテナンス」テンプレートを使用することで、自動生成される画面、処理は以下の通りです。

- 画面
 - 操作画面（登録/更新/参照等共通画面）
 - 一覧画面
- 処理
 - 登録処理
 - 更新処理
 - 削除処理
 - 参照用のデータ取得処理
 - 一覧用のデータ取得処理

コラム

アプリケーションを作成する際、入力項目について不明点があれば各項目名の横にある  にカーソルを合わせてください。アプリケーション利用時にどのように利用される項目であるか、設定を変えるとどのような変化があるかの説明が表示されます。



アプリケーション情報の設定

アプリケーション情報を設定します。

アプリケーションID、アプリケーション名、説明に以下の入力値を設定してください。

※アプリケーション情報に関しては、各項目に任意の入力値を設定してもかまいません。本チュートリアルは、下記の通りのアプリケーションID、アプリケーション名を入力した前提で進めます。

※「IDの決定に利用する文字列」はアプリケーションIDを入力すると自動的に入力されます。本チュートリアルでは自動入力された文字列をそのまま用いて進めます。

ラベル	入力値
アプリケーションID	tutorial_todo_app
アプリケーション名	【チュートリアル】ToDoアプリケーション
説明	マスタメンテナンステンプレートを利用して作成する簡易的なToDoアプリケーションです。

扱うデータの定義設定

次に、アプリケーションで利用するデータの定義を設定します。

ここで設定した定義を元に、データベースのテーブル、画面、処理が作成されます。

マスタメンテナンステンプレートのデータ定義は「IM-Repository」のデータリポジトリ機能を利用して作成します。

「エンティティを新規作成する」にチェックを入れ、今回作成するアプリケーションに合わせたデータが生成されるようにします。

IM-Repository資材の設定には簡易と詳細の2つの設定方法が用意されています。

本チュートリアルでの設定は詳細に設定するため、「詳細に設定する」へチェックを入れます。



エンティティについての情報を設定します。
エンティティID、エンティティ名、登録先エンティティカテゴリ、エンティティに紐づくテーブル名に以下の値を設定してください。

ラベル	入力値	備考
エンティティID	tutorial_todo_app_entity	
エンティティ名	【チュートリアル_エンティティ】ToDoアプリケーション	
登録先エンティティカテゴリ	【チュートリアル】エンティティ	任意のカテゴリで構いません。本チュートリアルでは新規作成しています。
データベース種別	TENANT	
エンティティに紐づくテーブル名	tutorial_todo_app	



上記の設定で、エンティティに紐づく「tutorial_todo_app」というテーブルが作成できます。
「IM-Repository」の機能により、エンティティの変更に合わせ、データベースのテーブルおよびテーブルを利用する登録、更新処理も変更されます。
「+エンティティ項目を追加」 - 「辞書項目を新規作成」を選択してください。

エンティティに紐づくテーブル名*

ヘルプアイコンから入力規則を確認できます。

tutorial_todo_app

エンティティ情報*

+ エンティティ項目を追加

- 辞書項目を新規作成
- 既存の辞書項目から選択
- 新規作成する辞書項目を参照

Databaseデータ型 表示形式

「辞書項目新規作成」ダイアログで、基本設定を入力しましょう。

まずは、ToDoを一意に決定する項目である「ID」を設定します。以下の表および画面を参考に設定してください。

ラベル	入力値	備考
登録先辞書項目カテゴリ	【チュートリアル】ToDo辞書	任意のカテゴリで構いません。本チュートリアルでは新規作成しています。本章で作成する辞書項目は全てこのカテゴリに設定してください。
辞書項目ID	tutorial_todo_id	
辞書項目名	ID	画面に表示される項目名にも利用されます。
データ型	TEXT	
変数名	id	画面や処理で利用する変数に利用されます。



「登録先辞書項目カテゴリ」は画面や処理では利用しないため、任意のカテゴリを設定してください。本チュートリアルでは、「【チュートリアル】ToDo辞書」というカテゴリを作成しています。本章で作成する辞書項目はすべてこのカテゴリに設定してください。

「辞書項目ID」も画面や処理では利用しませんが「IM-Repository」内で一意に設定します。重複に注意してください。

次に「データ」の項目を設定します。「データ」の項目では、エンティティの紐づけの設定を行います。本チュートリアルでは、Database設定のみを行います。

「物理名」に「id」、「最大桁数」に「100」と設定してください。



設定がすべて完了したら「設定」ボタンをクリックしてください。エンティティ情報に設定した辞書項目が1行追加されます。

「ID」の設定の最後に、追加したデータの設定を行いましょう。「表示形式」を「テキストボックス」と設定してください。「主キー」にチェックを入れ、「ID」を主キーとしてください。



これで、「ID」の設定は完了です。同様の手順で、以下の表に合わせてデータを作成しましょう。

本チュートリアルでは辞書項目に設定する「変数名」とデータの「物理名」には同様の値を設定します。

※「検索条件」のチェックボックスは、テンプレートを利用したマスタの一覧画面において、データ検索条件として利用するデータ項目のみチェックを入れます。

辞書項目ID	辞書項目名	データ型	変数名 / 物理名	最大桁数	表示形式	主キー	必須	検索条件
tutorial_todo_id	ID	TEXT	id	100	テキストボックス	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
tutorial_todo_title	概要	TEXT	title	1000	テキストボックス	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
tutorial_todo_detail	詳細	TEXT	detail	5000	複数行テキストボックス	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
tutorial_todo_category	カテゴリ	TEXT	category	1000	プルダウン	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

辞書項目ID	辞書項目名	データ型	変数名 / 物理名	最大桁数	表示形式	主キー	必須	検索条件
tutorial_todo_importance	重要度	TEXT	importance	1000	プルダウン			<input type="checkbox"/>
tutorial_todo_status	ステータス	TEXT	status	1000	プルダウン			<input type="checkbox"/>
tutorial_todo_manager	担当者	TEXT	manager	1000	テキストボックス		<input type="checkbox"/>	<input type="checkbox"/>
tutorial_todo_deadline	期限日	DATE	deadline		日付			
tutorial_todo_registered_user	登録者	TEXT	registered_user	1000	テキストボックス		<input type="checkbox"/>	
tutorial_todo_registered_date	登録日	DATE	registered_date		日付		<input type="checkbox"/>	
tutorial_todo_update_user	更新者	TEXT	update_user	1000	テキストボックス		<input type="checkbox"/>	
tutorial_todo_update_date	更新日	DATE	update_date		日付		<input type="checkbox"/>	

アプリケーション作成
マスタメンテナンス・新規作成

データベース種別
TENANT

エンティティに紐づくテーブル名
tutorial_todo_app

エンティティ項目を追加

辞書項目名	物理名	Databaseデータ型	表示形式	主キー	必須	検索条件	削除
ID	id	文字列	テキストボックス	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
概要	title	文字列	テキストボックス	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
詳細	detail	文字列	複数行テキストボックス	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
カテゴリ	category	文字列	プルダウン	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
重要度	importance	文字列	プルダウン	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
ステータス	status	文字列	プルダウン	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
担当者	manager	文字列	テキストボックス	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
期限日	deadline	日付	日付	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
登録者	registered_user	文字列	テキストボックス	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
登録日	registered_date	日付	日付	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
更新者	update_user	文字列	テキストボックス	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
更新日	update_date	日付	日付	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

アプリケーション作成

テーブル情報の入力まで完了したらテンプレートの新規作成の準備が完了です。「アプリケーション作成」をクリックし、確認ダイアログの「ビルド」をクリックします。

アプリケーション作成
マスタメンテナンス・新規作成

アプリケーション情報

扱うデータの定義

明細データの利用指定

その他情報の指定

期限日 deadline 日付 日付

登録者 registered_user 文字列 テキストボックス

登録日 registered_date 日付 日付

更新者 update_user 文字列 テキストボックス

更新日 update_date 日付 日付

明細データの利用指定

① 利用する明細データ数を指定します。

明細データの利用数
0

その他情報の指定

① このテンプレートでは、IM-LogicDesigner と IM-BloomMaker のリソースを新規作成し、アプリケーションを作成します。IM-LogicDesigner と IM-BloomMaker で利用される各種IDやルーティング情報などを自動で決定するかを指定します。自動で決定する場合は、「IDの決定に利用する文字列」が含まれた文字列が指定されます。

IM-LogicDesigner, IM-BloomMaker 資材の設定

リソースのIDやルーティング情報を手動で決定する

IDの決定に利用する文字列

※角英数字、_ (リイフ)、_ (アンダースコア) で入力してください。

tutorial_todo_app

確認
入力された情報でアプリケーションをビルドしますか？

キャンセル ビルド

アプリケーション作成

テンプレートからデータベース、画面、処理が自動で生成されます。
画面のステータス項目がCOMPLETEであれば、テンプレートからの自動作成は完了です。

アプリケーション作成
テンプレートビルドジョブ - 詳細

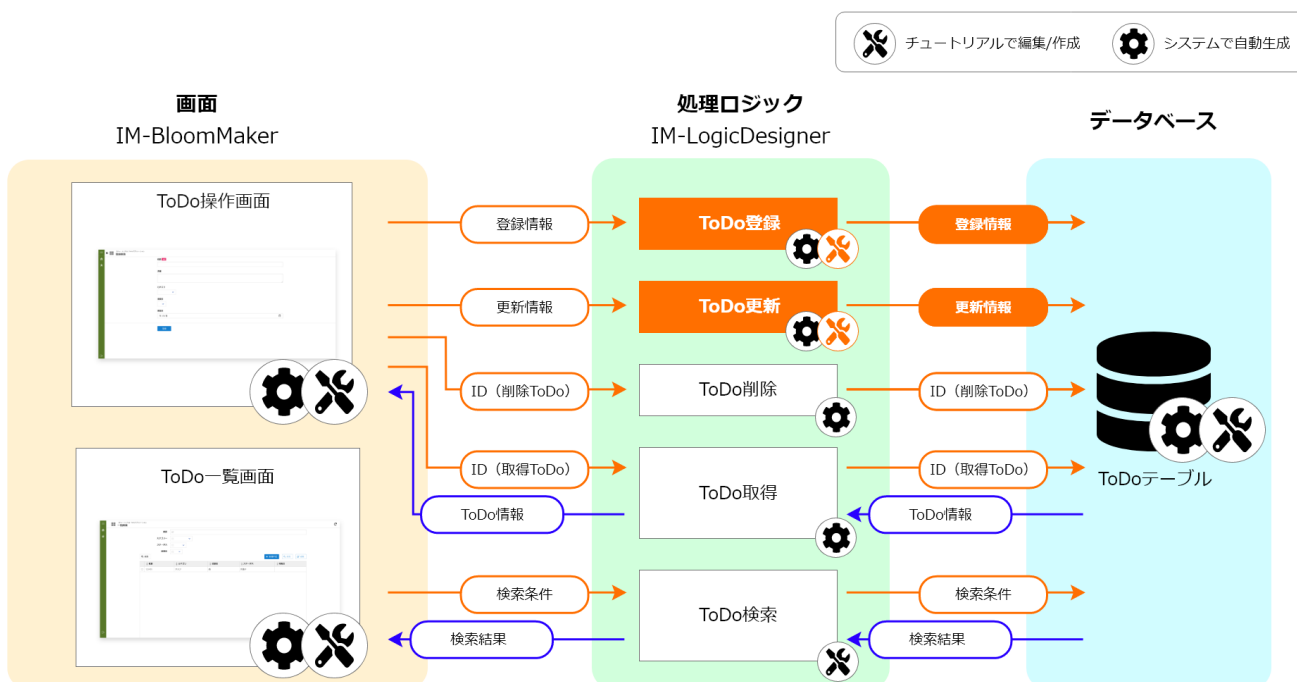
ビルドジョブID	8gpri064qzpkj1n	テンプレート名	マスタメンテナンス
ステータス	COMPLETE	作成者ユーザコード	tenant
作成日時	2023/02/24 13:56:31		

出力日時	メッセージ	詳細
2023/02/24 13:56:32	生成するリソースのIDを自動設定します。	
2023/02/24 13:56:37	一時ディレクトリに IM-LogicDesigner インポート資料を作成しました。	
2023/02/24 13:56:39	一時ディレクトリに IM-BloomMaker インポート資料を作成しました。	
2023/02/24 13:56:42	自動生成された IM-LogicDesigner 資料をインポートしました。	
2023/02/24 13:56:46	自動生成された IM-BloomMaker 資料をインポートしました。	
2023/02/24 13:56:46	作成された「一覧画面」へはこちらの詳細アイコンから遷移できます。	🔗
2023/02/24 13:56:55	アプリケーションの作成が完了しました。	🔗
2023/02/24 13:56:56	アプリケーション認可設定で テナント管理者,Accel Studio 管理者 にアクセス許可が設定されました。	

次のページでは、自動作成した処理の修正を行います。

登録、更新処理の修正

テンプレートからデータベース、IM-LogicDesigner の業務ロジック、IM-BloomMakerの画面が自動生成されました。
テンプレートから作成した業務ロジックでは、テンプレート作成時に設定したテーブル定義の項目をすべて画面上で入力して登録します。
そのため、IDや登録者、登録日など、自動で反映してほしい要素までユーザが自分で入力し登録する必要があります。
そこで、IM-LogicDesigner の業務ロジックを修正します。
なお、マスタメンテナンステンプレートでは、詳細画面表示用の業務ロジックとタスクの削除用の業務ロジックも作成されますが、本チュートリアルでは特に修正は加えません。
以下のアプリケーション全体図で色が塗られた箇所が修正対象です。



このページの手順について、以下の動画からも確認できます。併せてご利用ください。

注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

登録処理の修正

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」の順に遷移します。
アプリケーション一覧から「【チュートリアル】ToDoアプリケーション」を選択します。



アプリケーション管理画面のアプリケーション詳細で「ロジック」のタブを選択します。

「【チュートリアル】ToDoアプリケーション 登録処理」を選択します。

IM-LogicDesigner の編集画面が表示されます。

ToDoの登録処理の修正

IM-LogicDesigner 上でToDoの登録処理を修正します。

まずは、ToDoの登録処理の一部パラメータを自動で登録できるようにします。

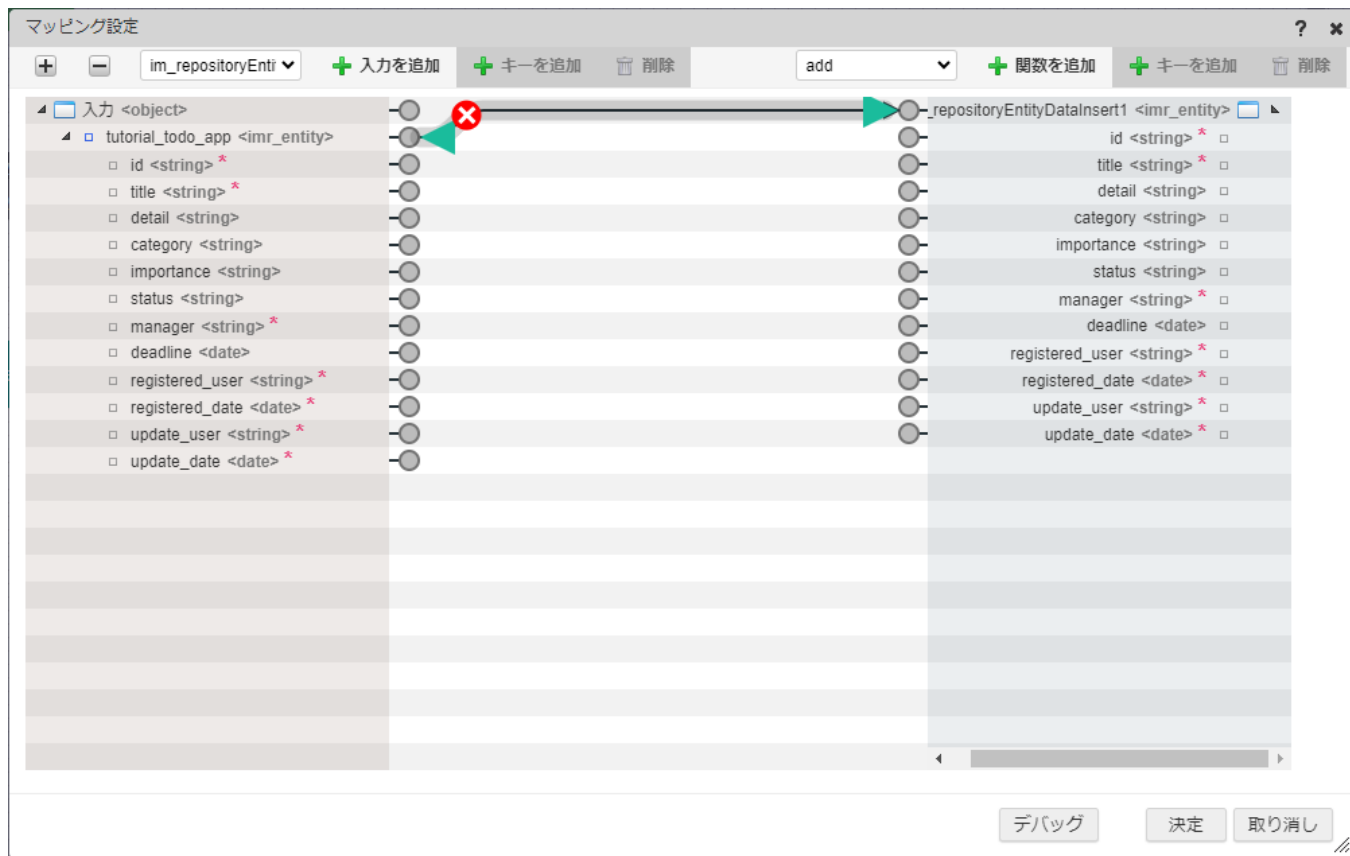
「tutorial_todo_app_entity の登録」タスクをダブルクリックし、「マッピング設定ダイアログ」を表示します。

「tutorial_todo_app_entity の登録」は、エンティティを設定し、エンティティに紐づくデータベースに対して登録処理を実行するタスクです。

自動作成された「tutorial_todo_app_entity の登録」は設定した全ての項目が入力値として設定されています。

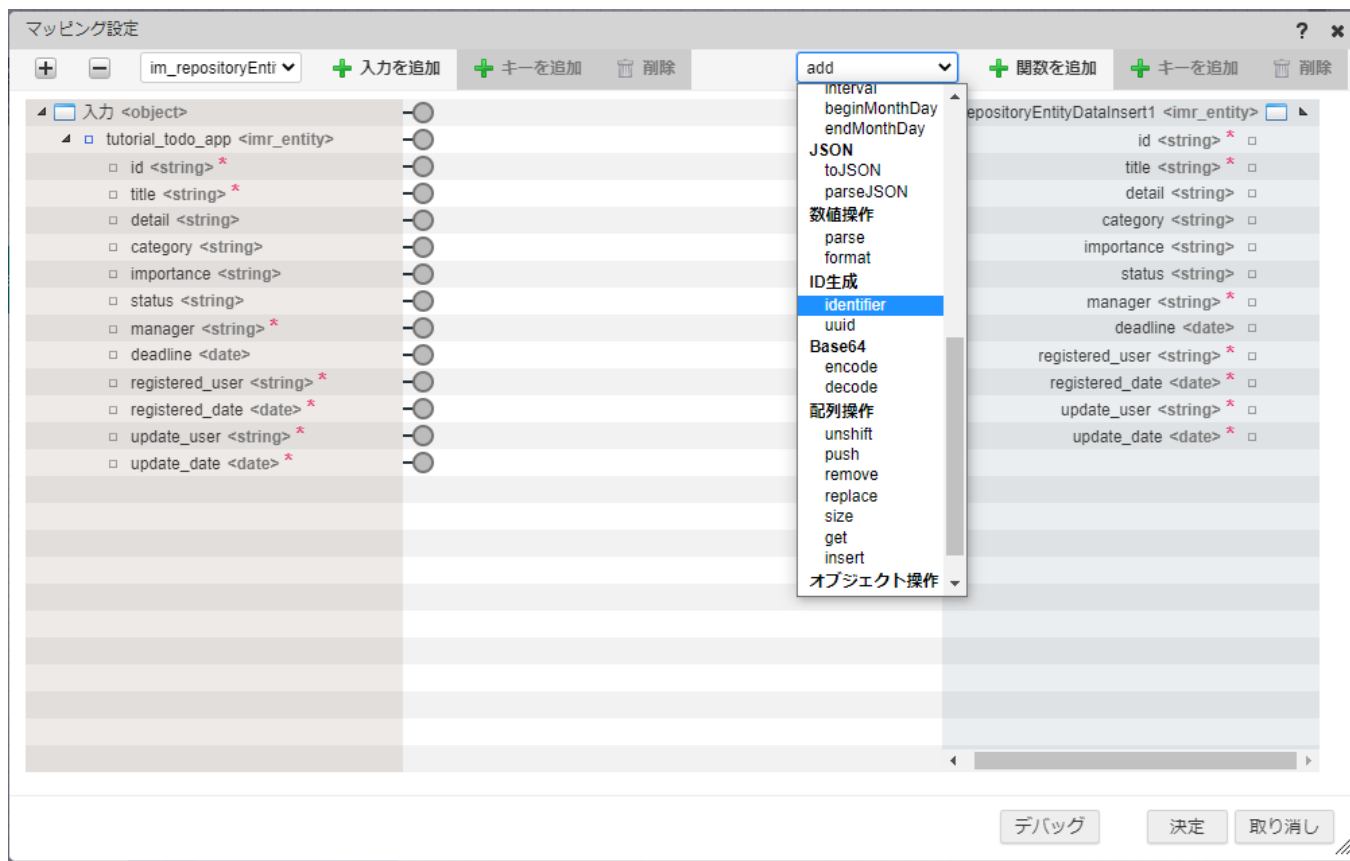
入力値 (tutorial_todo_app<imr_entity>) と出力値 (im_repositoryEntityDataInsert1<imr_entity>) を繋いでいる矢印を選択し、

✖ をクリックしてこの入力の紐づけを削除します。



まずは、IDを自動で設定するように修正します。

「マッピング設定」ダイアログ上部の関数プルダウンを選択し、ID生成から「identifier」を選択し「+関数を追加」をクリックして関数を配置します。



「マッピング設定」ダイアログに配置された「identifier」のoutから「tutorial_todo_app_entity」の登録」のidに対して線を繋げます。



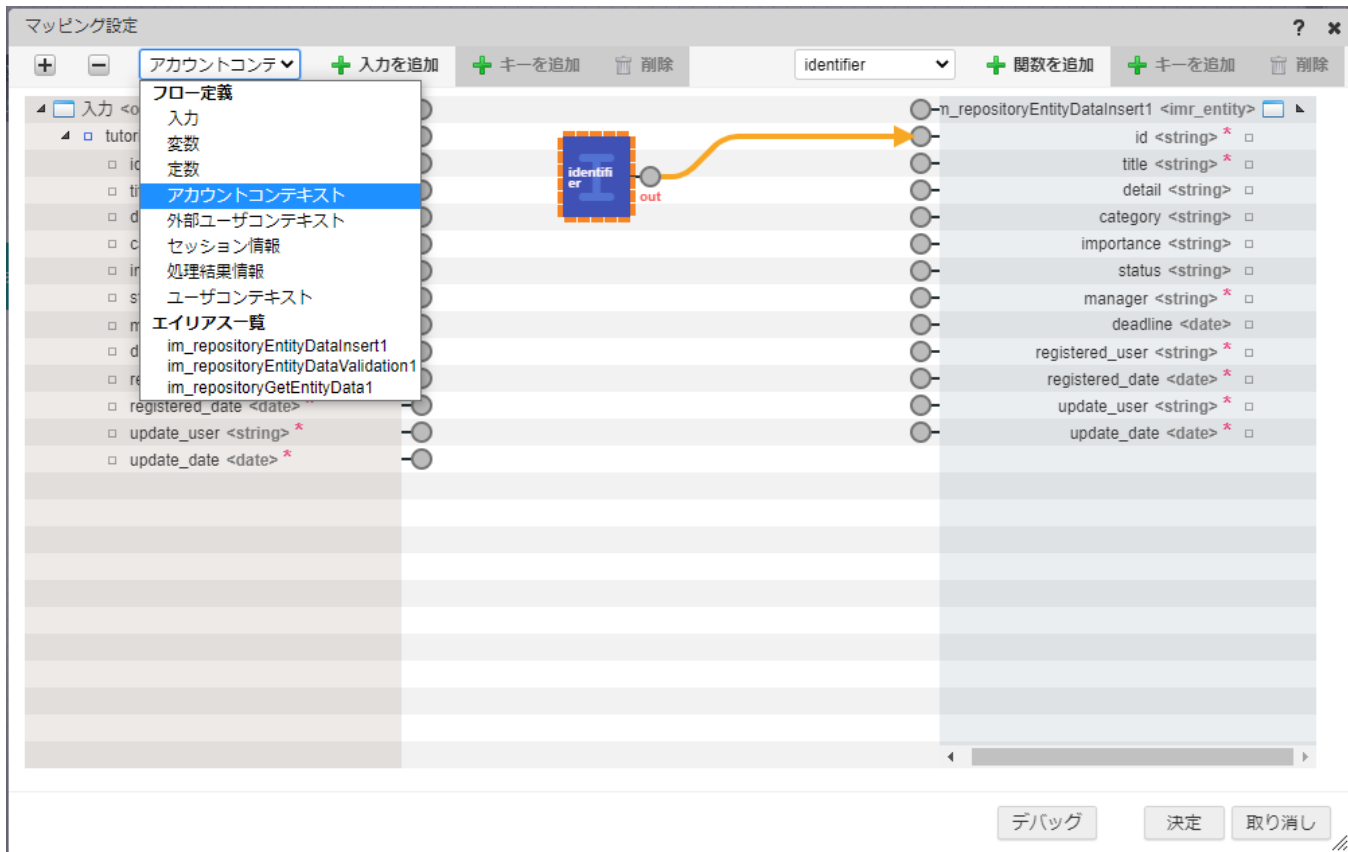
「Identifier」関数はAPI Identifierを利用して、一意となるIDを発行します。これで、IDを手入力することなく自動で割り振れるようになりました。

次に、登録者、登録日、更新者、更新日のような基本情報を自動設定させます。

登録のタイミングなので、登録者と更新者、登録日と更新日はそれぞれ同じ値を設定します。

まずは登録者を操作ユーザとするために、アカウントコンテキストを設定できるようにします。

「マッピング設定」ダイアログ上部の入力値選択欄から「アカウントコンテキスト」を選択して「+入力を追加」を選択します。



アカウントコンテキストのuserCdと「tutorial_todo_app_entity の登録」の入力値のregistered_userを紐づけます。

The screenshot shows the 'Mapping Configuration' (マッピング設定) interface. On the left, under 'Input' (入力), the 'Account Context' (アカウントコンテキスト) is expanded, and 'userCd' is selected. On the right, under 'Output' (出力), the 'tutorial_todo_app_entity' is expanded, and 'registered_user' is selected. A blue box labeled 'identifier' is positioned between the two, with an orange arrow pointing from 'userCd' to 'registered_user'. Below the mapping, there are buttons for 'Debug' (デバッグ), 'Decision' (決定), and 'Cancel' (取り消し).

同じようにアカウントコンテキストのuserCdと「tutorial_todo_app_entity の登録」の入力値のupdate_userを紐づけます。

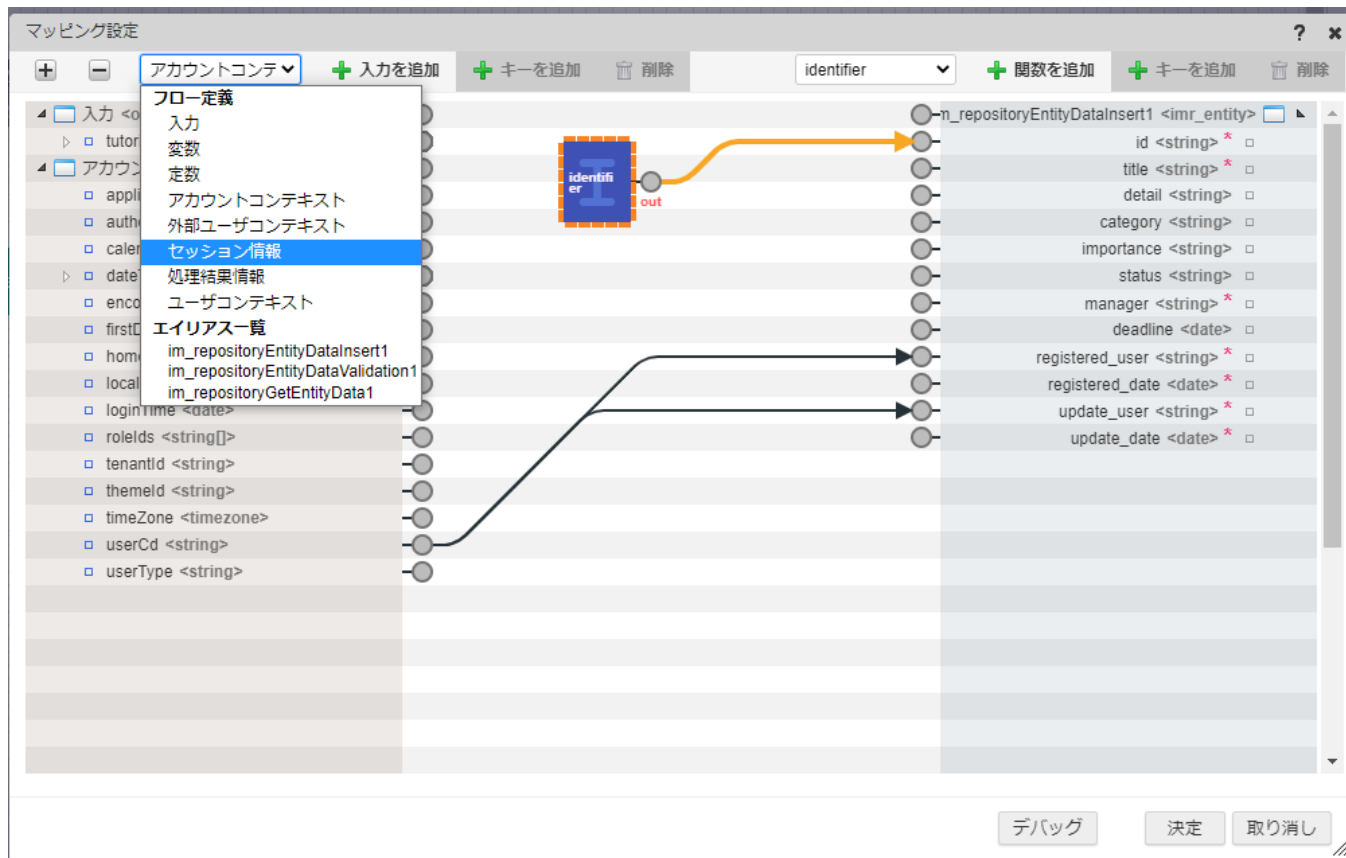
The screenshot shows the 'Mapping Configuration' (マッピング設定) interface. On the left, under 'Input' (入力), the 'Account Context' (アカウントコンテキスト) is expanded, and 'userCd' is selected. On the right, under 'Output' (出力), the 'tutorial_todo_app_entity' is expanded, and 'update_user' is selected. A blue box labeled 'identifier' is positioned between the two, with an orange arrow pointing from 'userCd' to 'update_user'. Below the mapping, there are buttons for 'Debug' (デバッグ), 'Decision' (決定), and 'Cancel' (取り消し).

これで、操作しているユーザのユーザコードが自動で「tutorial_todo_app_entity の登録」の入力値とする設定ができました。

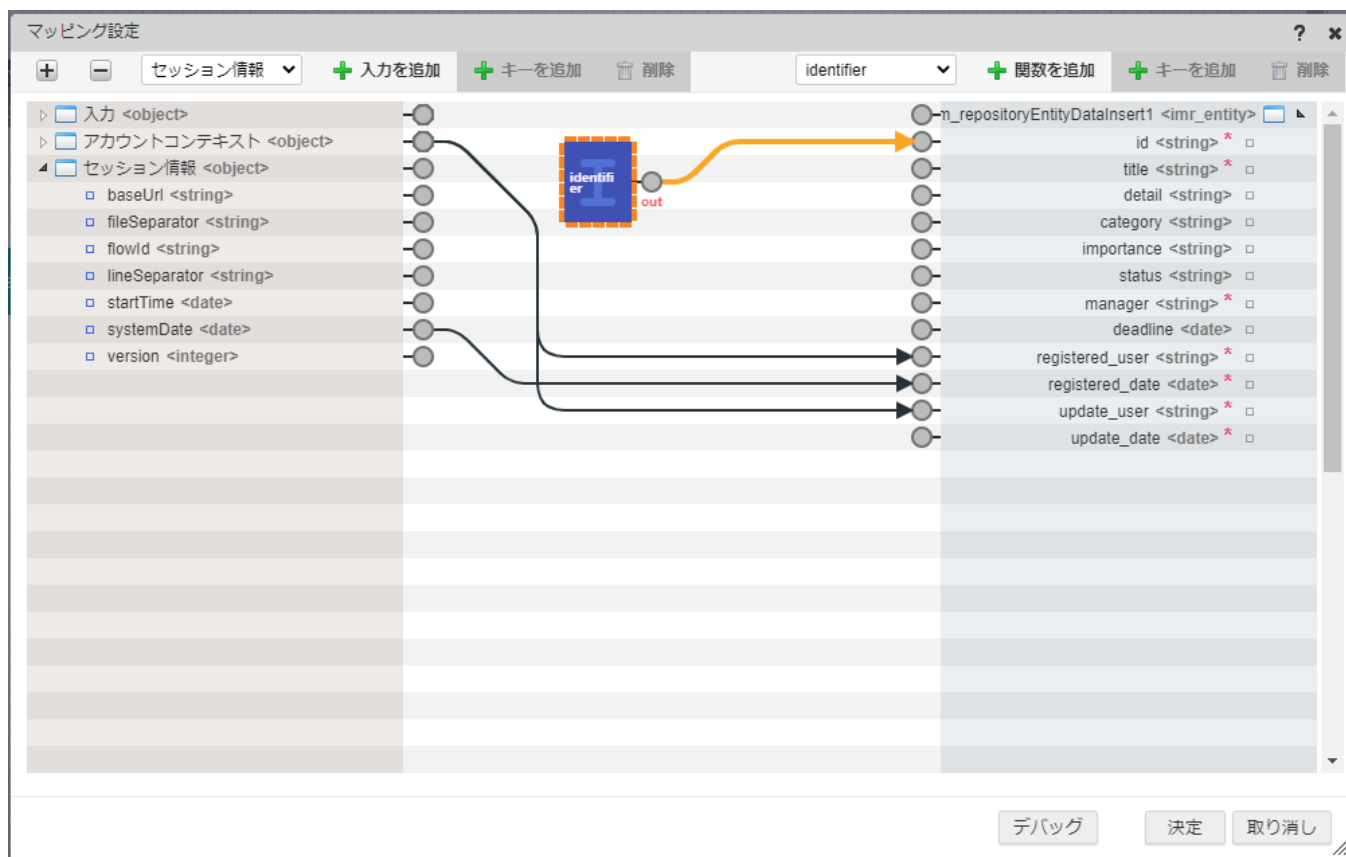
次に、登録日と更新日を操作しているタイミングの時間とします。

操作しているタイミングの時間はセッション情報のsystemDateを利用します。

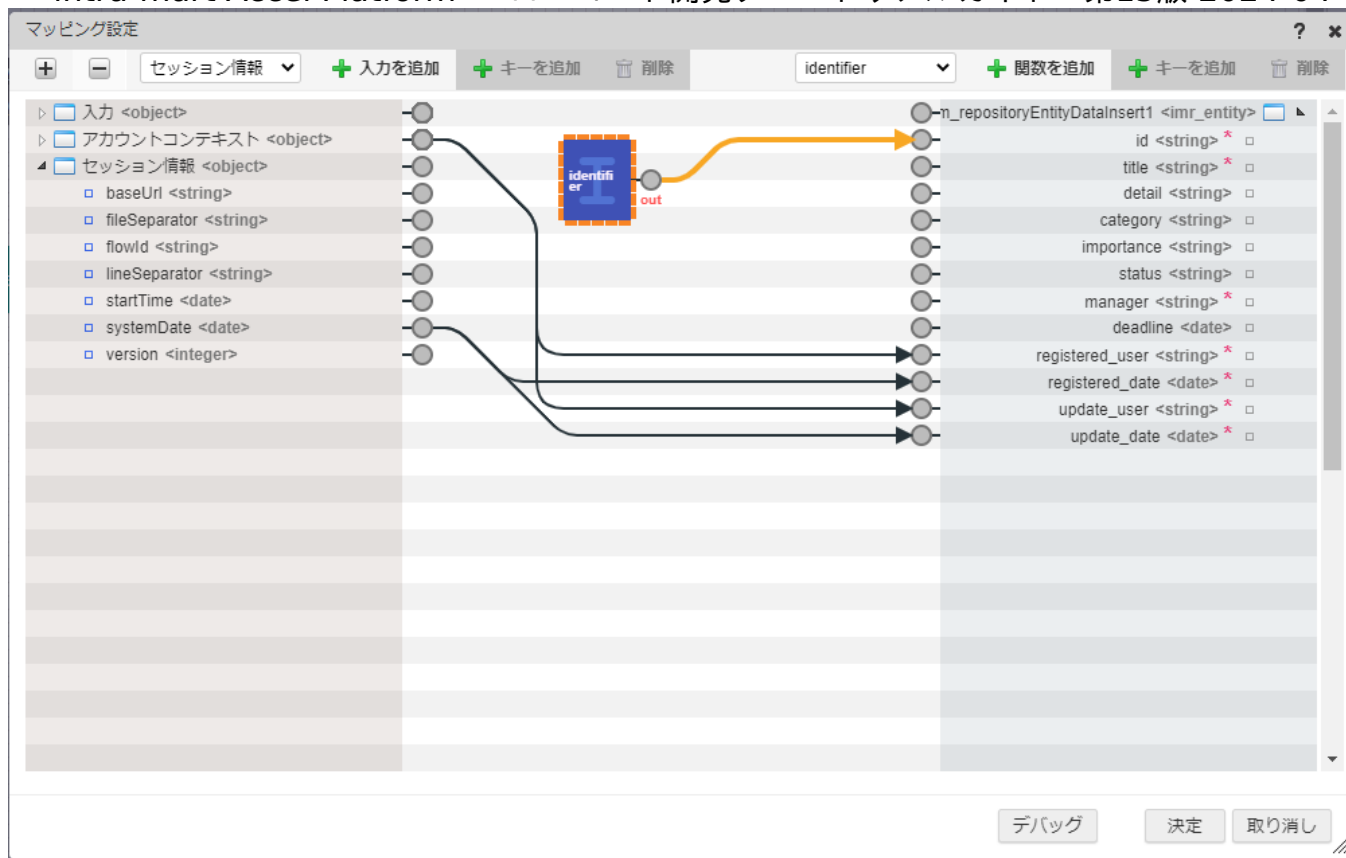
「マッピング設定」ダイアログ上部の入力値選択欄から「セッション情報」を選択して「+入力を追加」を選択します。



セッション情報のsystemDateと「tutorial_todo_app_entityの登録」の入力値のregistered_dateを紐づけます。



同じようにセッション情報のsystemDateと「tutorial_todo_app_entityの登録」の入力値のupdate_dateを紐づけます。

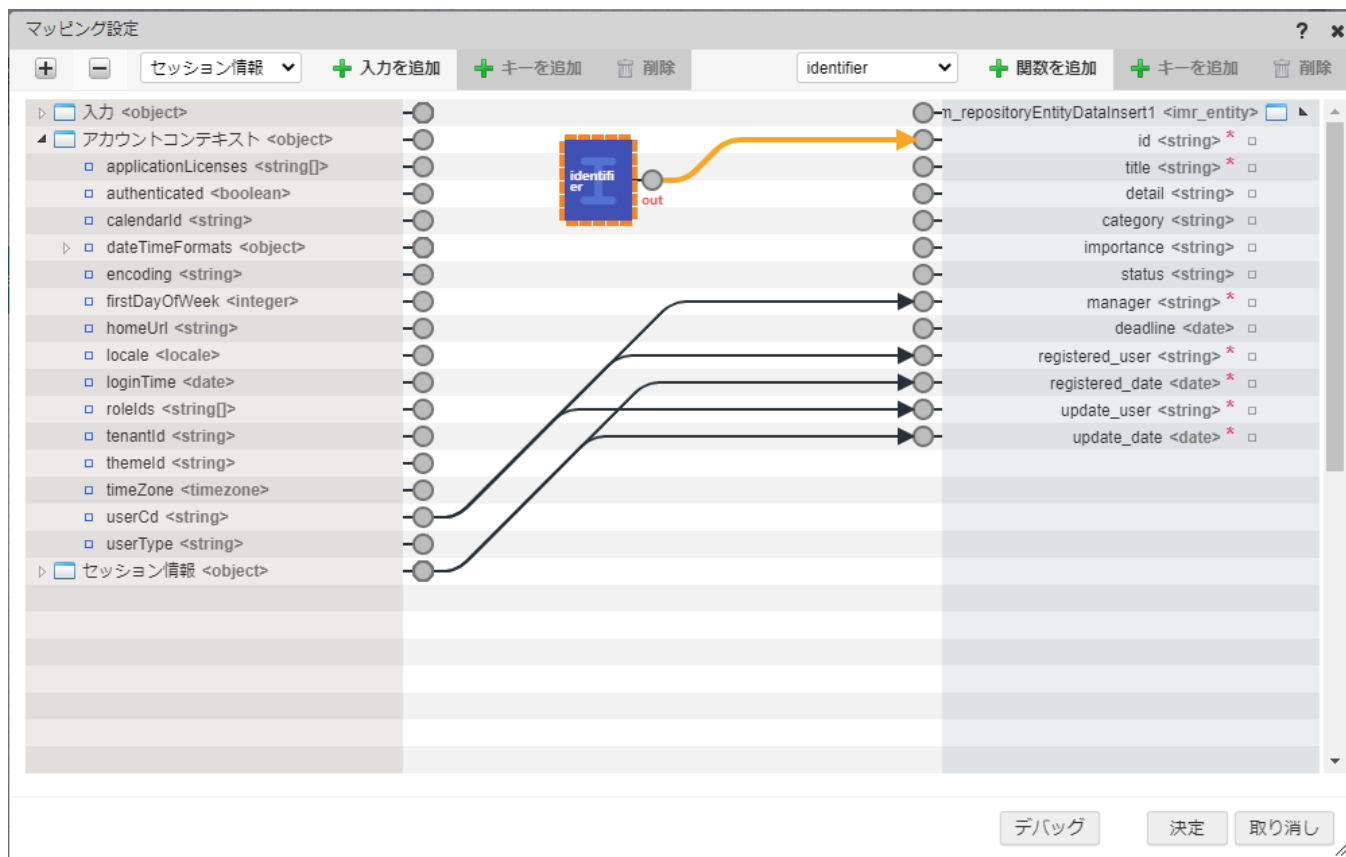


これで、登録日と更新日を操作している日時を自動で設定できます。

次に、本チュートリアルでは操作ユーザのToDoを登録するため、担当者を操作しているユーザにします。

登録者、更新者と同様にアカウントコンテキストからユーザを設定します。

アカウントコンテキストのuserCdと「tutorial_todo_app_entity の登録」の入力値のmanagerを紐づけます。



次に、statusはタスク登録時には必ず未着手のため定数を設定します。

まず、マッピング設定画面で「決定」をクリックして現在の設定を保存します。

画面上部の「定数設定」をクリックして「定数設定」ダイアログが表示されたら「+定数を追加」を選択します。

「定数ID」には「STATUS_WAITING」、「定数値」には「waiting」を入力し、「決定」をクリックしてダイアログを閉じます。

✕
+ 定数を追加
🗑️ 選択済みの定数を削除

📘 「エディタ」アイコンをクリックすると、「定数値」を複数行で入力できます。

選択	定数ID	定数値	説明	エディタ
<input type="checkbox"/>	TRUE	true		✎
<input type="checkbox"/>	STATUS_WAITING	waiting		✎

決定 取り消し


もう一度「tutorial_todo_app_entity の登録」の「マッピング設定」ダイアログを表示し、ダイアログ上部の入力値選択欄から「定数」を選択して「+入力を追加」を選択します。

定数を追加したら、入力の「STATUS_WAITING」と「tutorial_todo_app_entity の登録」のstatusを紐づけます。

? ✕
+ 入力を追加
+ 関数を追加

定数
+ キーを追加
削除
add
+ 関数を追加
+ キーを追加
削除

- ▶ 入力 <object>
- ▶ アカウントコンテキスト <object>
- ▶ セッション情報 <object>
- ▶ 定数 <object>
 - STATUS_WAITING <string>
 - TRUE <string>



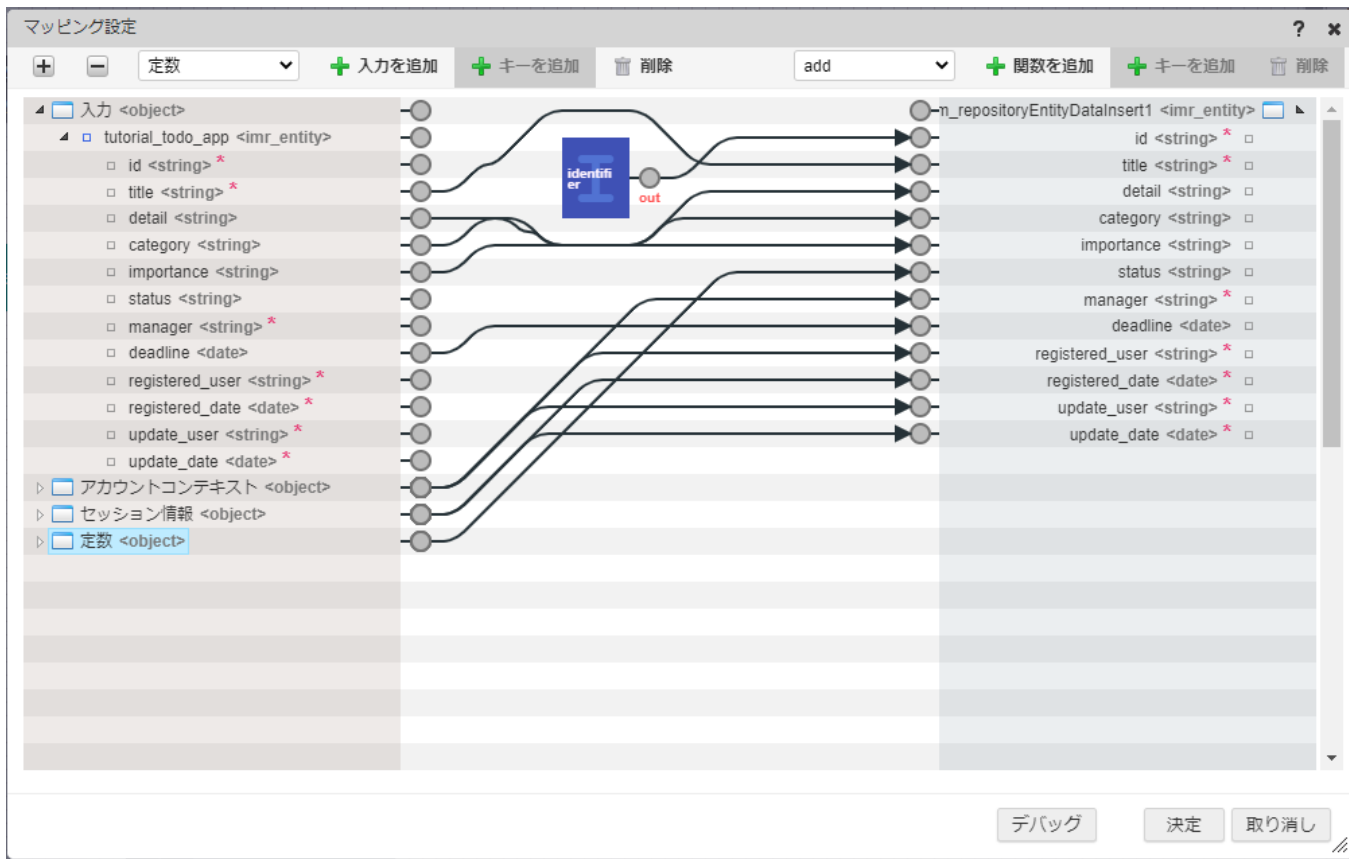
- in_repositoryEntityDataInsert1 <imr_entity>
- id <string> *
- title <string> *
- detail <string>
- category <string>
- importance <string>
- status <string>
- manager <string> *
- deadline <date>
- registered_user <string> *
- registered_date <date> *
- update_user <string> *
- update_date <date> *

デバッグ 決定 取り消し

これで自動で設定したい項目の設定が完了しました。

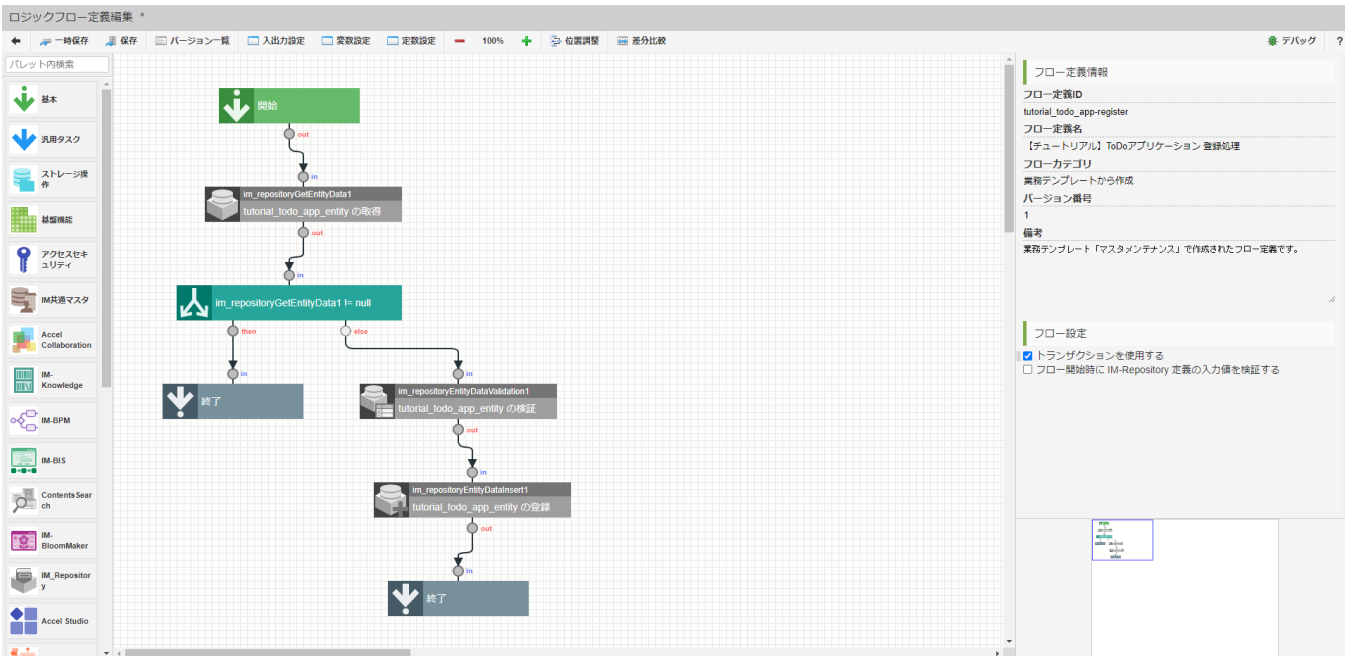
その他の変数を以下のように設定します。

入力の「title」、「detail」、「category」、「importance」、「deadline」をそれぞれ同名の入力値と紐づけ、紐づけが完了したらダ



不要な処理の削除

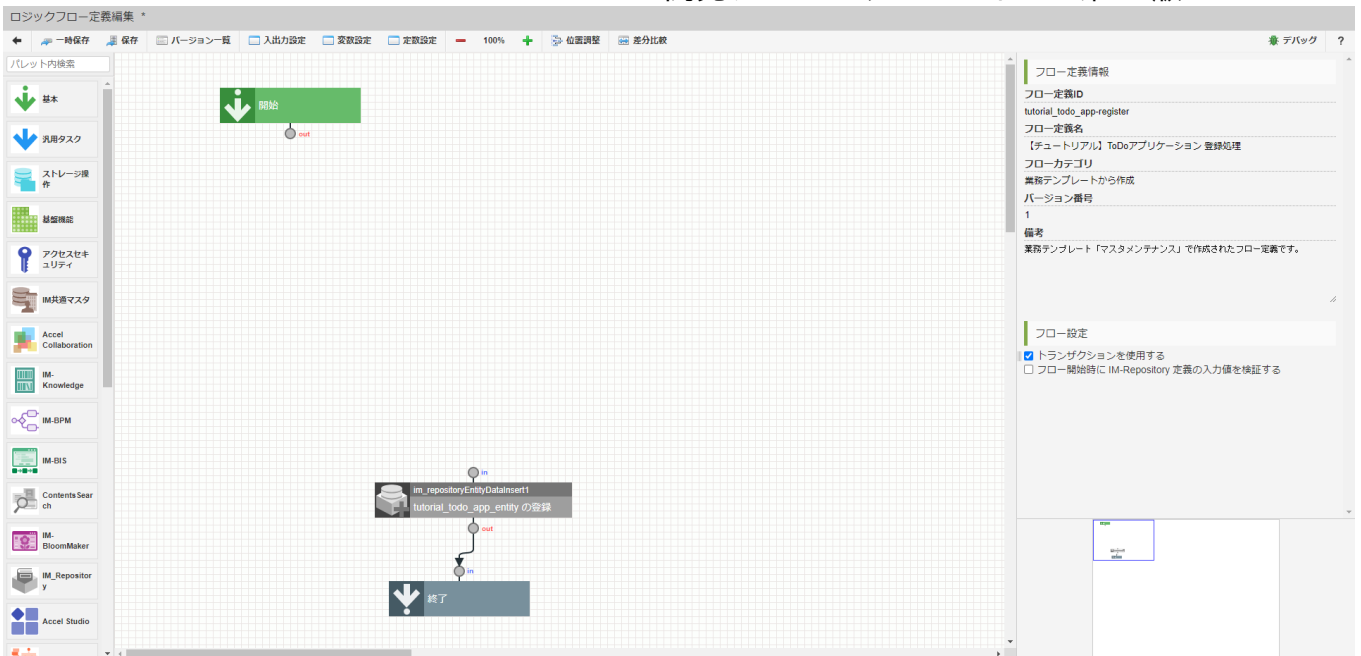
IDを自動で一意的IDが設定されるようにしたため、不要な登録時の重複チェックを削除します。
 「tutorial_todo_app_entity の取得」タスクを選択しDELETEキーを押し削除します。



「tutorial_todo_app_entity の取得」タスクが削除されました。

続けて、「im_repositoryGetEntityData1 != null」と表示されている「分岐」タスクと「終了」タスクを選択し、DELETEキーを押し削除します。

同様に、「tutorial_todo_app_entity の検証」タスクを選択し、DELETEキーを押し削除します。



「開始」タスクの「out」から「tutorial_todo_app_entity の登録」タスクの「in」まで線を繋げます。これで、登録に関するロジックフローの修正は完了です。画面上部の「保存」ボタンからロジックフローを保存してください。



3通りの保存方法がありますが、今回は「新しいバージョン番号を追加して保存する」を選択して、「決定」をクリックしてください。次に、更新フローの修正を行います。

更新処理の修正

更新処理の修正を行います。

更新処理では登録者と登録日情報を更新する必要はありません。そのため、更新処理からは登録者と登録日を更新しないようにし、その他の情報のみを更新するように修正します。

アプリケーション管理画面のアプリケーション構成情報で「ロジック」のタブを選択します。

「【チュートリアル】ToDoアプリケーション 編集処理」を選択します。

IM-LogicDesigner の編集画面が表示されます。

更新者情報の自動設定

登録と同様に更新者と更新日をアカウントコンテキストとセッション情報から設定します。

「tutorial_todo_app_entity の更新」タスクをダブルクリックし、「マッピング設定」ダイアログを表示します。

表示されたダイアログでは、入力の「tutorial_todo_app<imr_entity>」と「im_repositoryEntityDataUpdate1<imr_entity>」が紐づけられています。

マッピング設定

im_repositoryEnti + 入力を追加 + キーを追加 削除 add + 関数を追加 + キーを追加 削除

入力 <object>

- tutorial_todo_app <imr_entity>
 - id <string> *
 - title <string> *
 - detail <string>
 - category <string>
 - importance <string>
 - status <string>
 - manager <string> *
 - deadline <date>
 - registered_user <string> *
 - registered_date <date> *
 - update_user <string> *
 - update_date <date> *

im_repositoryEntityDataUpdate1 <imr_entity>

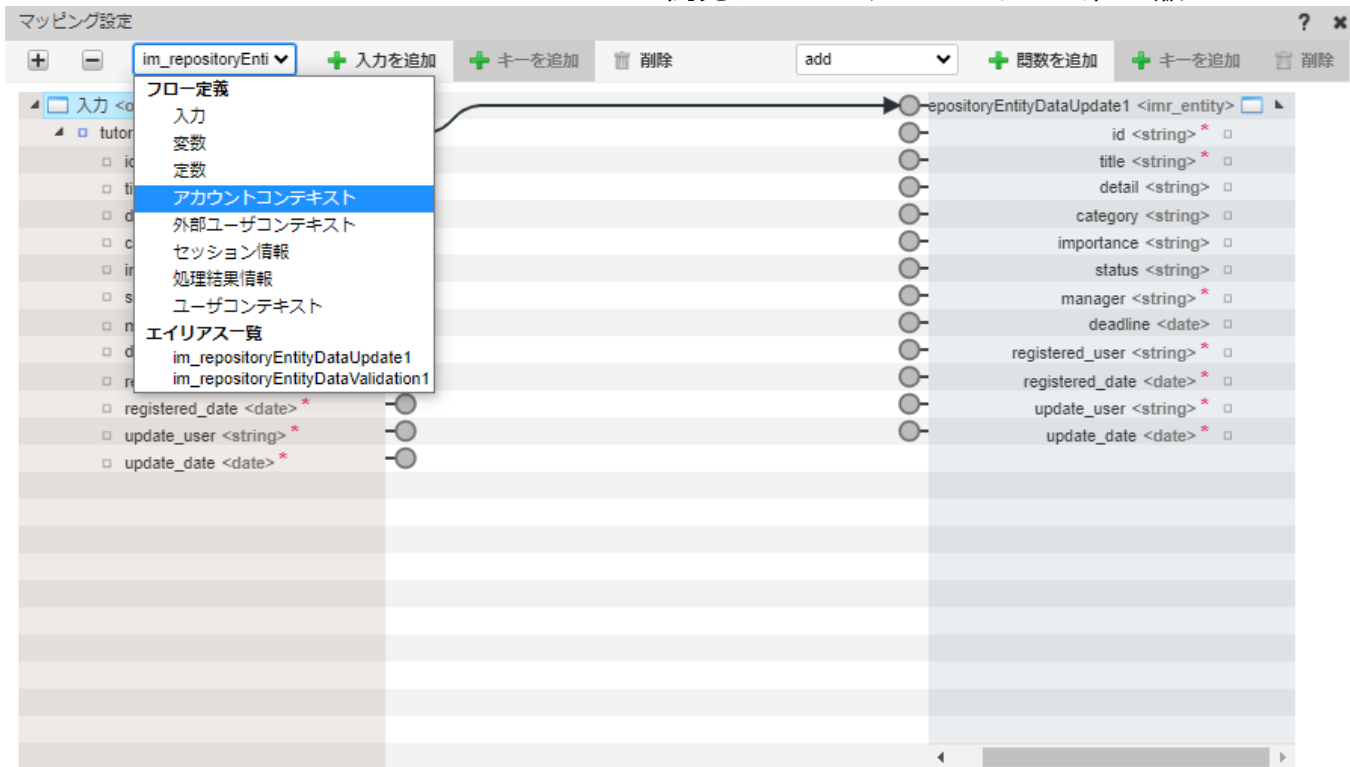
- id <string> *
- title <string> *
- detail <string>
- category <string>
- importance <string>
- status <string>
- manager <string> *
- deadline <date>
- registered_user <string> *
- registered_date <date> *
- update_user <string> *
- update_date <date> *

デバッグ 決定 取り消し

これによりオブジェクト型データ「tutorial_todo_app<imr_entity>」のプロパティと、オブジェクト型データ「im_repositoryEntityDataUpdate1<imr_entity>」のプロパティは紐づけられています。

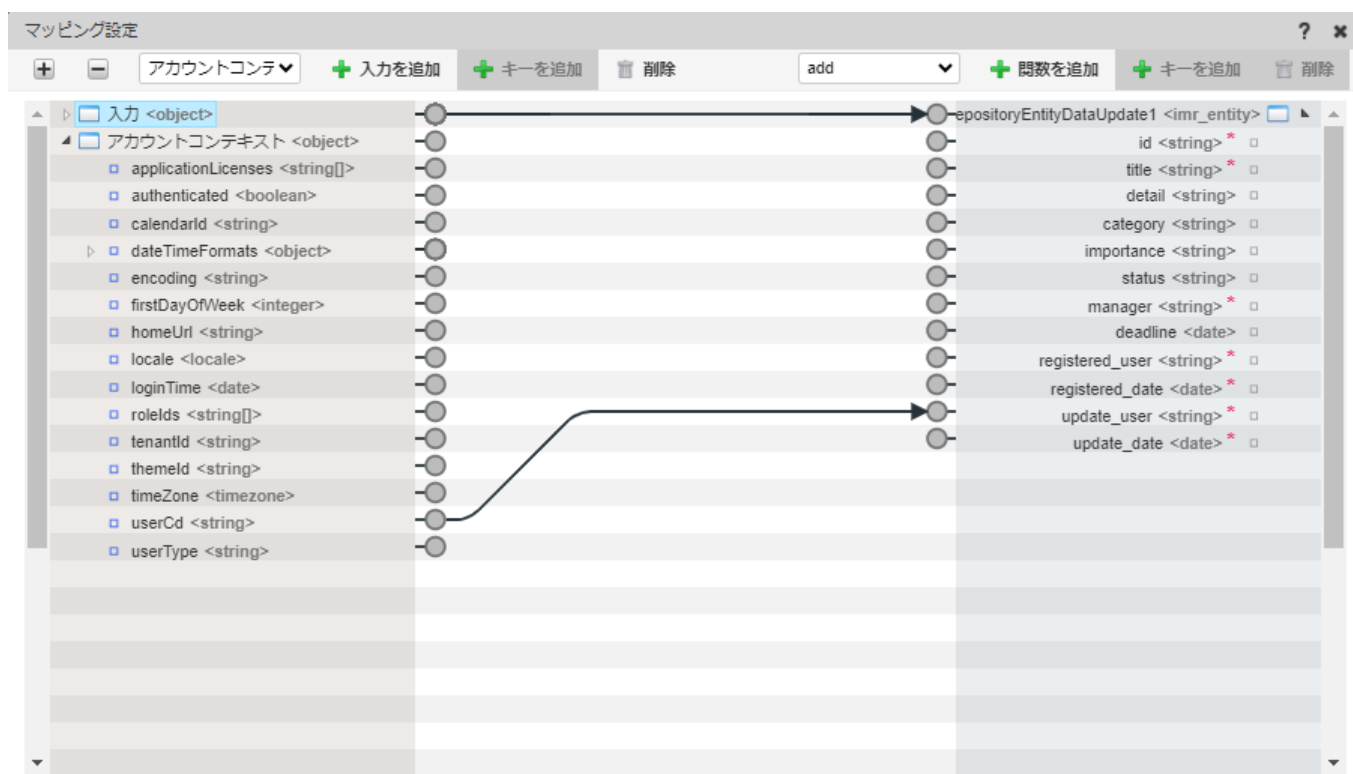
ここから更新者と更新日の情報について設定していきます。

「マッピング設定」ダイアログ上部の入力値選択欄から「アカウントコンテキスト」を選択して「+入力を追加」を選択します。



デバッグ 決定 取り消し

アカウントコンテキストのuserCdと「tutorial_todo_app_entity の更新」の入力値のupdate_userを紐づけます。



デバッグ 決定 取り消し

これで、編集操作を実施しているユーザのユーザコードが自動で「tutorial_todo_app_entity の更新」の入力値として設定されます。次に、更新日を編集操作しているタイミングの時間とします。操作しているタイミングの時間はセッション情報のsystemDateを利用します。「マッピング設定」ダイアログ上部の入力値選択欄から「セッション情報」を選択して「+入力を追加」を選択します。

セッション情報のsystemDateと「tutorial_todo_app_entityの更新」の入力値のupdate_dateを紐づけます。

! 注意

<object>型のマッピングとプリミティブ型のマッピングが混在した場合、マッピングを行った（線を引いた）順に処理が行われるため、一見、同一のマッピングに見えても動作が違うパターンが存在します。

<object>型の配下にプリミティブ型を配置するなど、階層化された値の定義をする際には、<object>型、プリミティブ型の順にマッピングを行う必要があります。

マッピング順に気を付けて操作してください。

これでマッピング設定は完了です。ダイアログ下部の「決定」をクリックしてダイアログを閉じます。

画面上部の「保存」ボタンからロジックフローを保存してください。

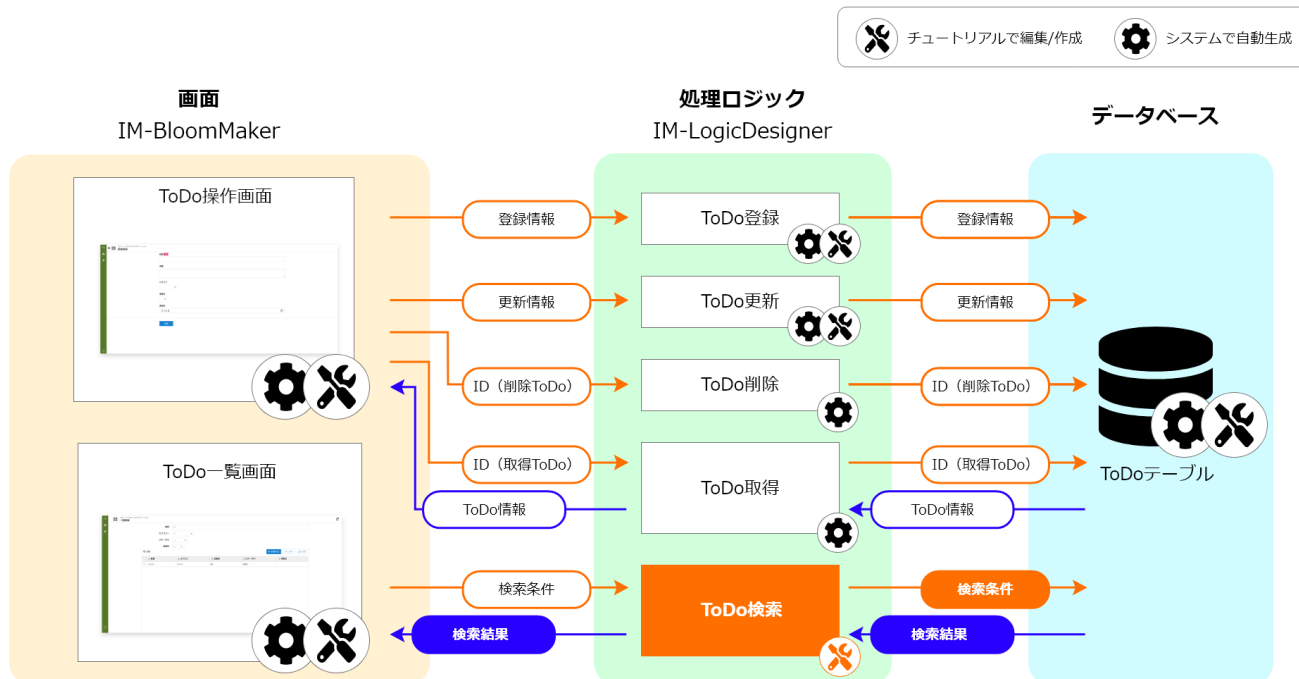
次のページでは、一覧画面での検索処理を作成します。

一覧での検索処理の確認

マスタメンテナンスプレートは、アプリケーション作成時に「検索条件」として設定した項目の検索が自動で作成されます。

検索は、IM-LogicDesignerとIM-Repositoryのデータリポジトリ機能を利用して作成されるため、設定を確認しましょう。

以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



このページの手順について、以下の動画からも確認できます。併せてご利用ください。

注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

検索処理の確認

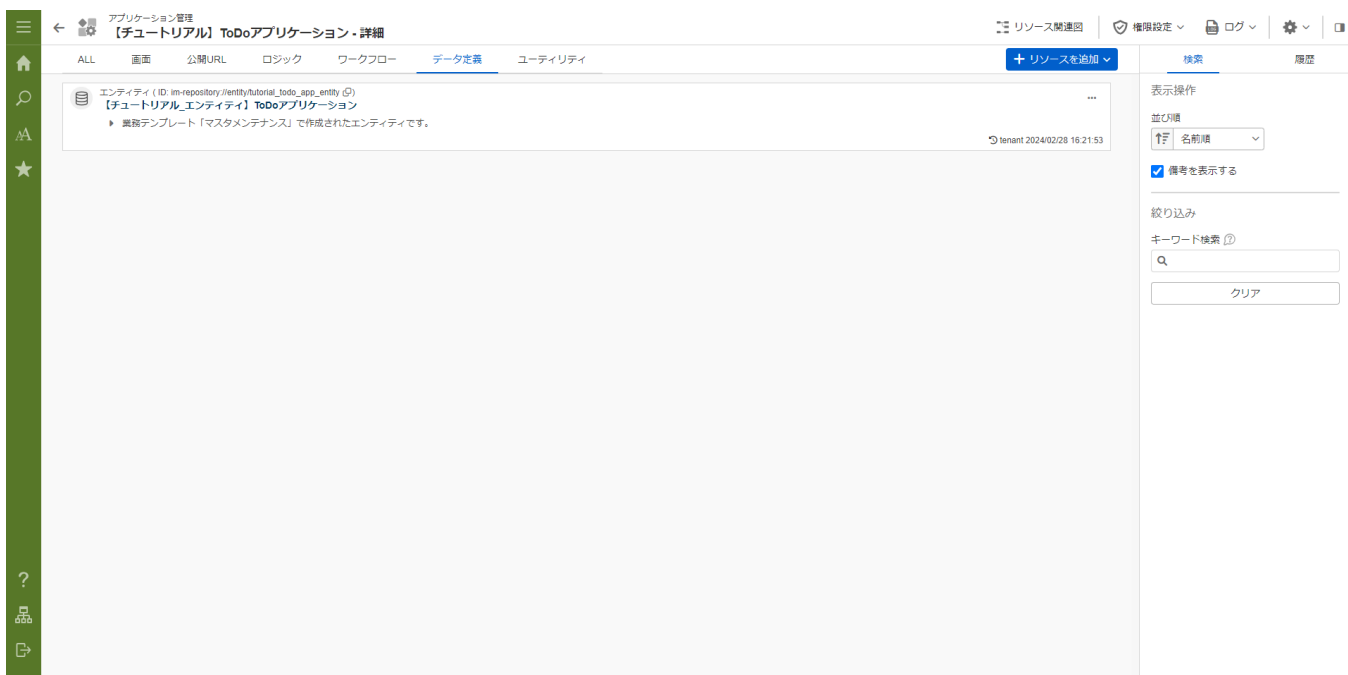
IM-LogicDesignerとIM-Repositoryのデータリポジトリ機能を利用し、以下の項目で検索を行う処理となっているか確認しましょう。

- 概要
 - タスクの概要を部分検索で検索する
- カテゴリ
 - タスクのカテゴリを一致検索で検索する
- ステータス
 - タスクのカテゴリを一致検索で検索する
- 重要度
 - タスクのカテゴリを一致検索で検索する
- 担当者
 - 操作ユーザのみ表示させるため、IM-LogicDesigner上で操作ユーザのユーザコードを自動で設定する

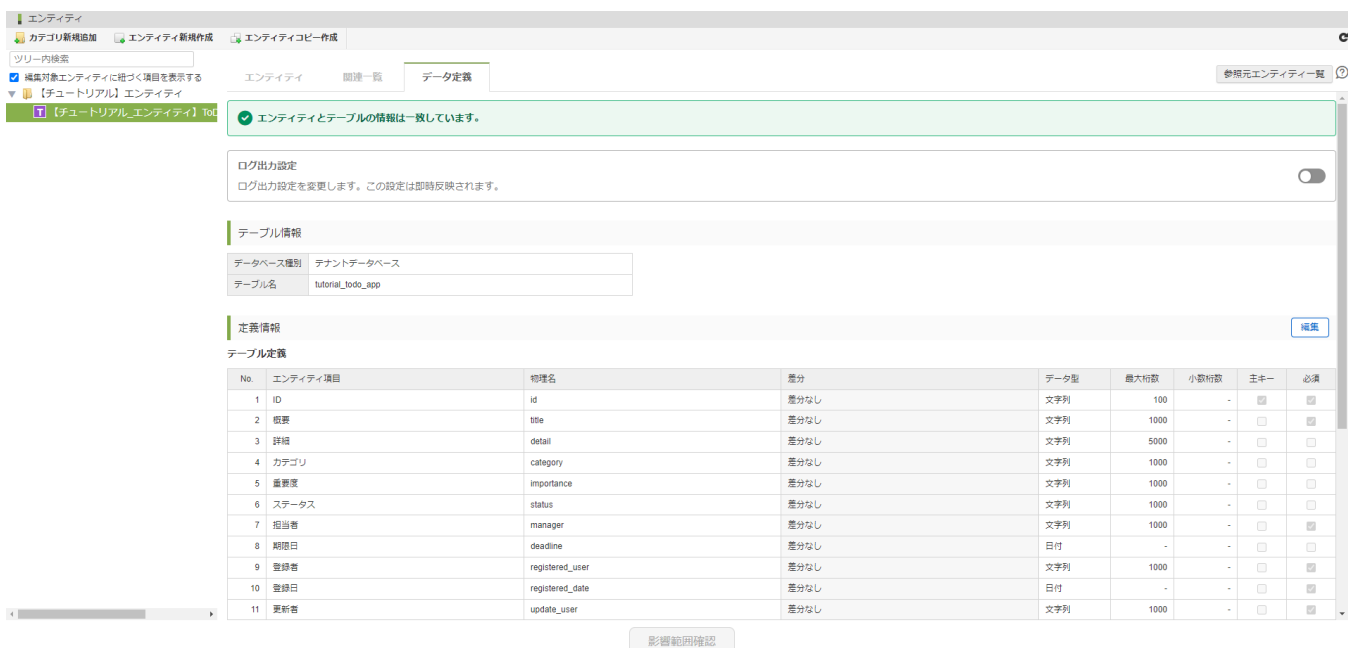
データ定義の確認

まずは、データ定義に検索条件を設定します。データ定義に検索条件を設定すると、SQLなどを書かなくてもIM-LogicDesignerで検索処理が簡単に作成ができます。

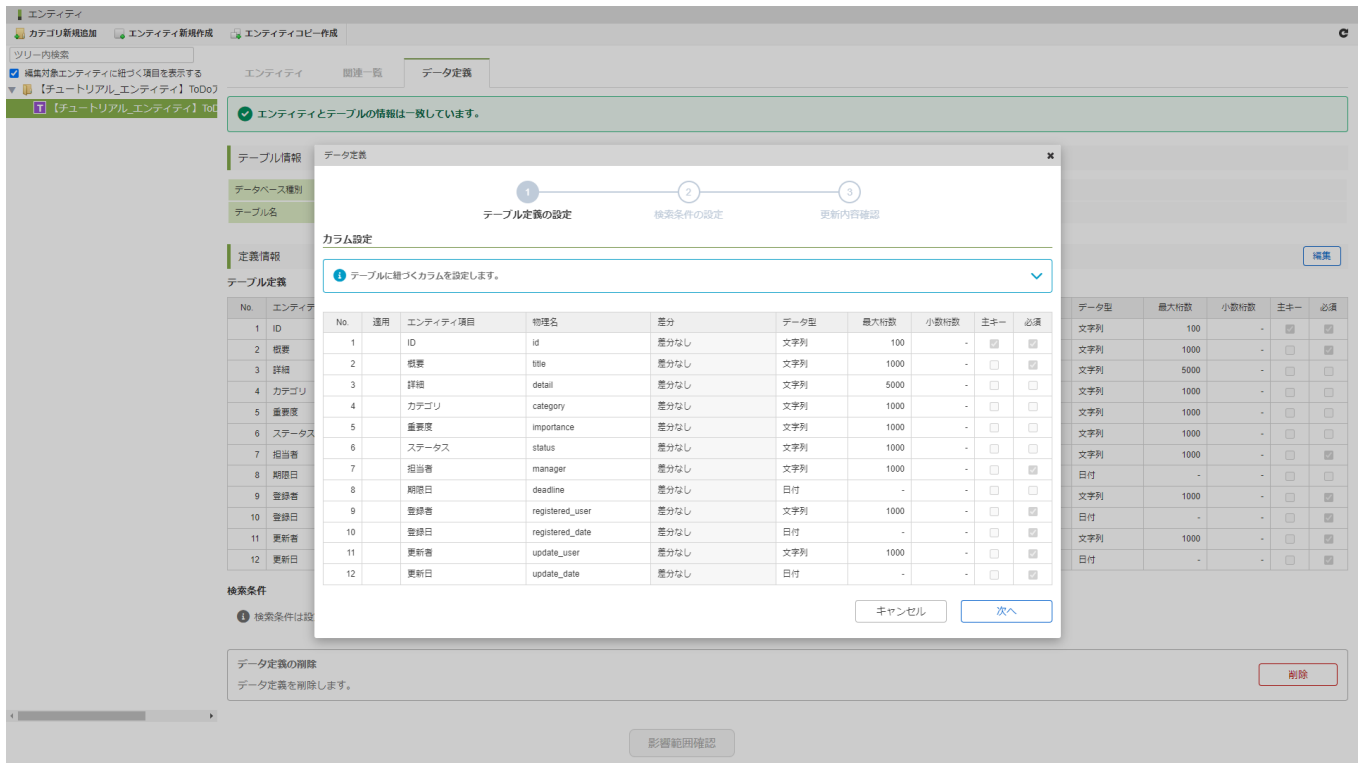
アプリケーション管理画面のアプリケーション詳細で「データ定義」のタブを選択します。



エンティティの「【チュートリアル_エンティティ】ToDoアプリケーション」を選択します。
表示されたエンティティ画面の「データ定義」タブを選択します。



アプリケーション作成時に設定したテーブル定義情報が表示されます。
「定義情報」の「編集」ボタンをクリックします。



データ定義モーダルが表示されます。
「次へ」ボタンをクリックしてください。



「検索条件の設定」が表示されます。
各検索項目について、下の表のように設定されているか確認します。
※設定が異なる場合は「条件を追加」ボタンや、各項目のプルダウン選択から、表の通りになるよう変更してください。

エンティティ項目	条件
概要	部分一致
カテゴリ	完全一致
ステータス	完全一致
重要度	完全一致
担当者	完全一致

データ定義 ×



i テーブルのレコードを絞り込むための検索条件を設定します。

検索条件の設定

条件を追加

No.	エンティティ項目	条件	削除
1	概要	部分一致	⊗
2	カテゴリ	完全一致	⊗
3	重要度	完全一致	⊗
4	ステータス	完全一致	⊗
5	担当者	完全一致	⊗

戻る

キャンセル

次へ

設定が完了したら「次へ」ボタンをクリックして、内容を確認してください。

データ定義

✓ テーブル定義の設定
 ✓ 検索条件の設定
 3 更新内容確認

テーブル情報

データベース種別	テナントデータベース
テーブル名	tutorial_todo_app

テーブル定義

No.	操作	エンティティ項目	物理名	データ型	最大桁数	小数桁数	主キー	必須
1		ID	id	文字列	100	-	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2		概要	title	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3		詳細	detail	文字列	5000	-	<input type="checkbox"/>	<input type="checkbox"/>
4		カテゴリ	category	文字列	1000	-	<input type="checkbox"/>	<input type="checkbox"/>
5		重要度	importance	文字列	1000	-	<input type="checkbox"/>	<input type="checkbox"/>
6		ステータス	status	文字列	1000	-	<input type="checkbox"/>	<input type="checkbox"/>
7		担当者	manager	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
8		期限日	deadline	日付	-	-	<input type="checkbox"/>	<input type="checkbox"/>
9		登録者	registered_user	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
10		登録日	registered_date	日付	-	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
11		更新者	update_user	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>

問題が無ければ、「反映」ボタンをクリックして、設定を反映してください。

エンティティ

カテゴリ新規追加
 エンティティ新規作成
 エンティティコピー作成

ツリー内検索

編集対象エンティティに基づく項目を表示する
 【チュートリアル】エンティティ
 【チュートリアル_エンティティ】Todo

エンティティ 関連一覧 **データ定義**

ログ出力設定
 ログ出力設定を変更します。この設定は即時反映されます。

参照元エンティティ一覧

テーブル情報

データベース種別	テナントデータベース
テーブル名	tutorial_todo_app

定義情報

テーブル定義

No.	適用	エンティティ項目	物理名	差分	データ型	最大桁数	小数桁数	主キー	必須
1		ID	id	差分なし	文字列	100	-	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2		概要	title	差分なし	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
3		詳細	detail	差分なし	文字列	5000	-	<input type="checkbox"/>	<input type="checkbox"/>
4		カテゴリ	category	差分なし	文字列	1000	-	<input type="checkbox"/>	<input type="checkbox"/>
5		重要度	importance	差分なし	文字列	1000	-	<input type="checkbox"/>	<input type="checkbox"/>
6		ステータス	status	差分なし	文字列	1000	-	<input type="checkbox"/>	<input type="checkbox"/>
7		担当者	manager	差分なし	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
8		期限日	deadline	差分なし	日付	-	-	<input type="checkbox"/>	<input type="checkbox"/>
9		登録者	registered_user	差分なし	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
10		登録日	registered_date	差分なし	日付	-	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
11		更新者	update_user	差分なし	文字列	1000	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>
12		更新日	update_date	差分なし	日付	-	-	<input type="checkbox"/>	<input checked="" type="checkbox"/>

参照項目設定

⚠ 保存されていない変更があります。

「影響範囲確認」ボタンをクリックしてください。

影響範囲確認画面が表示されますので、任意のコメントを入力し、「適用」ボタンをクリックして設定を保存しましょう。

影響範囲一覧

IM-BloomMaker コンテンツ : 4

コンテンツID	バージョン番号	コンテンツ名	変数パス
tutorial_todo_app-crud	1	【チュートリアル】 ToDoアプリケーション 登録・編集・参照画面	Svariable/endPoint/edit/request/Parameter/entity/tutorial_todo_app
tutorial_todo_app-crud	1	【チュートリアル】 ToDoアプリケーション 登録・編集・参照画面	Svariable/endPoint/refer/response/Data/entity/tutorial_todo_app
tutorial_todo_app-crud	1	【チュートリアル】 ToDoアプリケーション 登録・編集・参照画面	Svariable/endPoint/register/request/Parameter/entity/tutorial_todo_app
tutorial_todo_app-crud	1	【チュートリアル】 ToDoアプリケーション 登録・編集・参照画面	Svariable/state/entity/tutorial_todo_app

Accel Studio アプリケーション : 1

アプリケーションID	アプリケーション名	エンティティID	エンティティ名
tutorial_todo_app	【チュートリアル】 ToDoアプリケーション	tutorial_todo_app_entity	【チュートリアル_エンティティ】 ToDoアプリケーション

LogicDesigner フロー : 5

フロー定義ID	フロー定義名	バージョン番号	エンティティID	エンティティ名
tutorial_todo_app-edit	【チュートリアル】 ToDoアプリケーション 編集処理	1	tutorial_todo_app_entity	【チュートリアル_エンティティ】 ToDoアプリケーション
tutorial_todo_app-edit	【チュートリアル】 ToDoアプリケーション 編集処理	2	tutorial_todo_app_entity	【チュートリアル_エンティティ】 ToDoアプリケーション
tutorial_todo_app-refer	【チュートリアル】 ToDoアプリケーション 参照処理	1	tutorial_todo_app_entity	【チュートリアル_エンティティ】 ToDoアプリケーション
tutorial_todo_app-register	【チュートリアル】 ToDoアプリケーション 登録処理	1	tutorial_todo_app_entity	【チュートリアル_エンティティ】 ToDoアプリケーション
tutorial_todo_app-register	【チュートリアル】 ToDoアプリケーション 登録処理	2	tutorial_todo_app_entity	【チュートリアル_エンティティ】 ToDoアプリケーション

履歴情報入力

コメント* 検索条件を追加しました。

説明

適用

これで、検索条件の設定は完了です。
タスク一覧画面が表示されますので、「閉じる」ボタンで画面を閉じてください。

ロジックフローの確認

次に、検索処理のロジックフローを確認します。

アプリケーション管理画面のアプリケーション構成情報で「ロジック」タブを選択します。
「【チュートリアル】 ToDoアプリケーション 一覧取得処理」をクリックし、ロジックフロー定義編集画面を表示します。

ロジックフロー定義編集

パレット内検索

基本
汎用タスク
ストレージ操作
状態管理
アクセスセキュリティ
IM共通マスタ
Accel Collaboration
IM Knowledge
IM BPM
IM BIS
Contents Search
IM BloomMaker
IM_Repository
Accel Studio

開始

im_repositorySearchEntityCount1
tutorial_todo_app_entity 件数の取得

im_repositorySearchEntityData1
tutorial_todo_app_entity の検索

終了

フロー定義情報

フロー定義ID
tutorial_todo_app-list

フロー定義名
【チュートリアル】 ToDoアプリケーション 一覧取得処理

フローカテゴリ
業務テンプレートから作成

バージョン番号
1

備考
業務テンプレート「マスタメンテナンス」で作成されたフロー定義です。

フロー設定

トランザクションを使用する

フロー開始時に IM-Repository 定義の入力値を検証する

「tutorial_todo_app_entity の検索」タスクをダブルクリックし、マッピング設定ダイアログを表示します。

マッピング設定

im_repositorySea

+ 入力を追加 + キーを追加 削除 add + 関数を追加 + キーを追加 削除

入力 <object>

- searchCondition <map>
- sortCondition <object[]>
 - key <string>
 - order <string>
 - limit <integer>
 - offset <integer>

im_repositorySearchEntityData1 <object>

- keys <imr_entity>
 - title <string>
 - category <string>
 - importance <string>
 - status <string>
 - manager <string>
 - sort <object[]>
 - key <string>
 - order <string>
 - limit <integer>
 - offset <integer>

入力.offset <integer> ⇒ im_repositorySearchEntityData1.offset <integer>

左側の値がnullの場合、右側にnullを渡します。
左側の値が数値の場合、右側に定義されている型に変換後、右側に渡します。
変換後の数値が、右側の型が対応する範囲よりあふれている場合、有効な範囲内に丸められます。

デバッグ 決定 取り消し

ダイアログ左側の入力情報に、検索条件がマップ型データである searchCondition として渡されていることが確認できます。searchCondition は、オブジェクト型データ im_repositorySearchEntityData1 配下の keys にマッピングされています。keys の配下に「[テンプレートからアプリケーションのベースを作る](#)」の「[扱うデータの定義設定](#)」で設定した検索条件が含まれていることを確認してください。

このため、入力値の searchCondition に以下のような形で keys に対応する値を受け渡すことで、検索処理が実行できます。

```
"searchCondition": {
  "title": "ToDo01",
  "category": "task",
  "importance": "medium",
  "status": "working",
  "manager": "aoyagi"
}
```

これでロジックフローの確認は終わりました。

マッピング設定ダイアログの「× (閉じる)」ボタンで表示を終了させ、ロジックフロー定義編集画面左上の「← (戻る)」ボタンでタブの表示を終了してください。

次のページから作成した業務ロジックに合わせて画面の修正を行います。

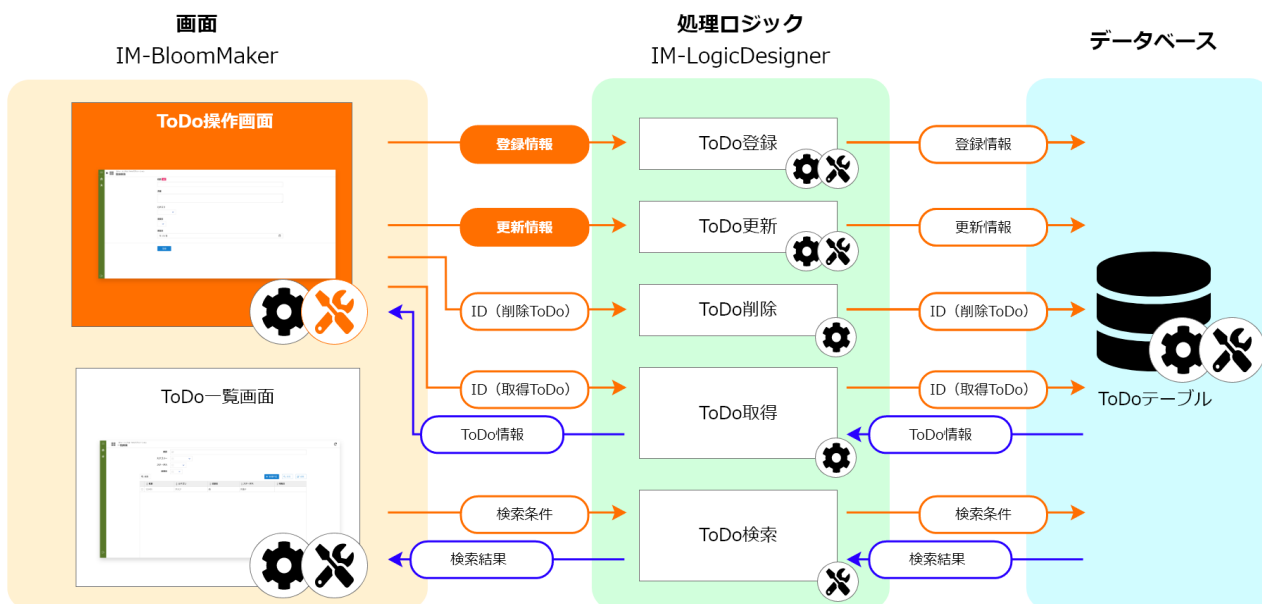
登録/更新/参照画面の修正

「マスタメンテナンステンプレート」から作成した画面は登録、更新、参照画面が1つの画面で作られており、アクセスするURLによって表示する画面種別を制御しています。

テンプレートから生成された画面では、設定した全ての項目が入力項目となっているため、修正した処理と合わなくなっています。

修正せずとも画面の利用は可能ですが、ロジックの仕様に合わせて画面を修正していきます。

以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



このページの手順について、以下の動画からも確認できます。併せてご利用ください。

! 注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

入力、表示項目の修正

まずは、操作画面の入力、表示項目の修正を行います。

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】ToDoアプリケーション」の順に遷移します。

アプリケーション管理画面のアプリケーション構成情報で「画面」のタブを選択します。

「【チュートリアル】ToDoアプリケーション 登録・編集・参照画面」を選択します。

IM-BloomMaker コンテンツ画面で「デザイン編集」を選択します。

IM-BloomMakerの「デザイン編集画面」が表示されます。

入力項目の修正

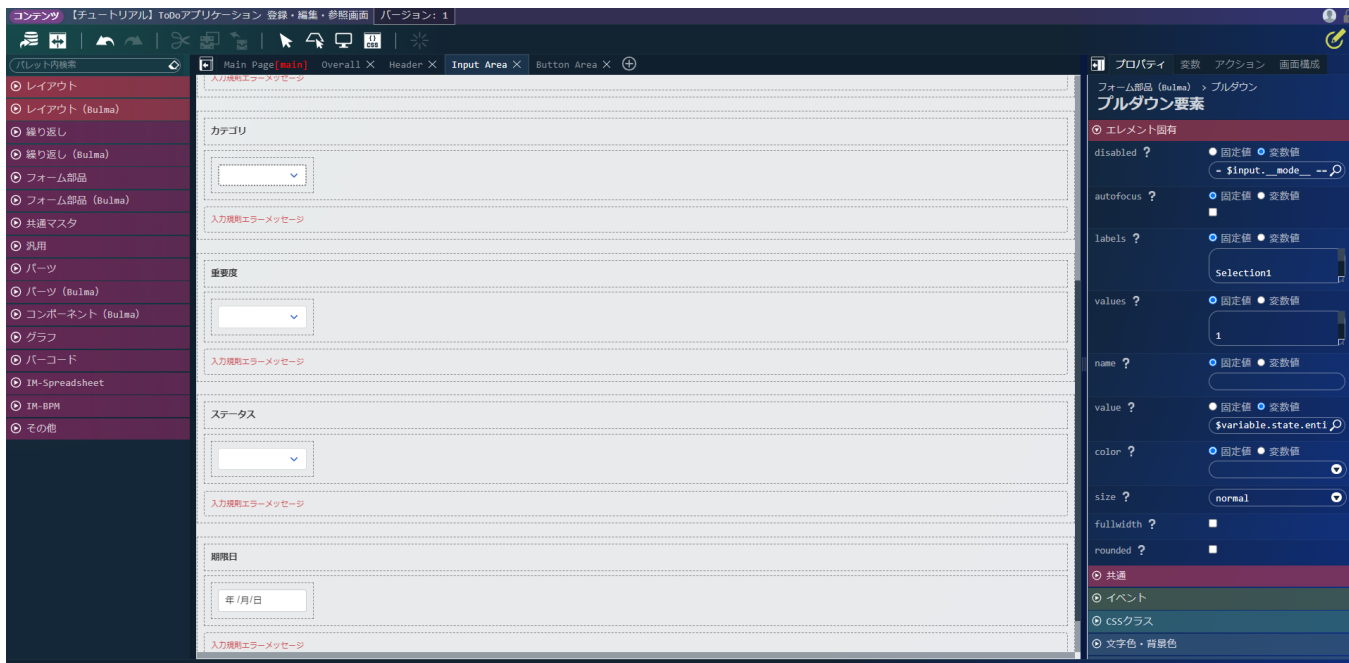
画面上部のタブの中から「Input Area」タブを選択します。

「Input Area」タブは入力項目が設定された画面です。

まずは、プルダウンの入力項目を登録したい値となるように修正します。

テンプレートから自動生成された画面のプルダウン要素のlabels、valuesには変数値が設定されています。

画面項目「カテゴリ」のプルダウンから設定します。画面に配置された「カテゴリ」のプルダウン要素を選択します。



画面右側に表示される「プルダウン要素」の「エレメント固有」 - 「Labels」のラジオボタンで固定値を選択し、以下のように設定します。「Labels」はプルダウンの表示要素です。

1行目には何も入力せずに改行するように注意してください。

- labels
 - (空欄)
 - タスク
 - スケジュール
 - 移動
 - メモ

同様に「エレメント固有」 - 「values」のラジオボタンで固定値を選択し、以下のように設定します。「values」はプルダウンを選択した際に送信される値の要素です。

「Labels」同様に、1行目には何も入力せずに改行するように注意してください。

- values
 - (空欄)
 - task
 - schedule
 - move
 - memo

「Labels」と「values」は同じ行番号の値同士が対応して選択値と送信値として設定されます。

例えば、上記の設定を行った場合、プルダウンで3行目の「スケジュール」を選択した場合、「values」の3行目である「schedule」が送信値として「value」に指定した変数に設定されます。

同様にほかのプルダウン要素も設定していきます。「重要度」の「Labels」と「values」に以下のように設定します。

- labels
 - (空欄)
 - 高
 - 中
 - 低
- values
 - (空欄)

- high
- medium
- low

次は、「ステータス」の「labels」と「values」を以下のように設定します。

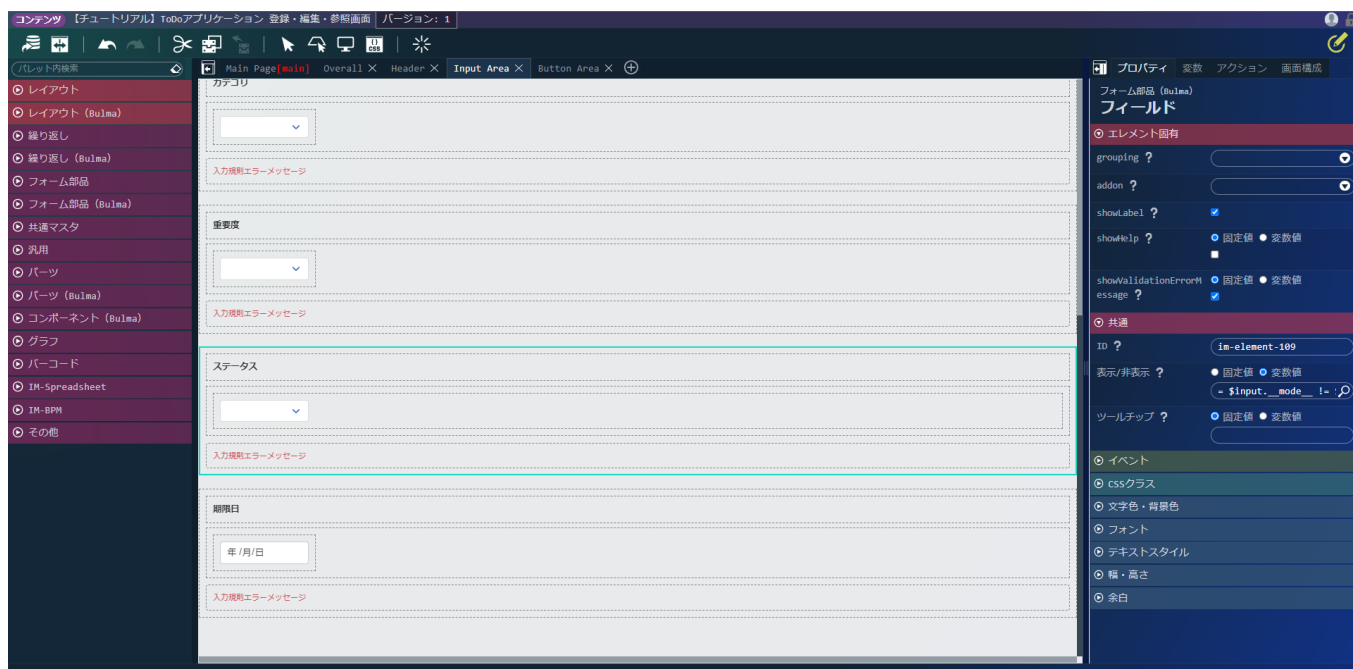
- labels
 - (空欄)
 - 未着手
 - 着手中
 - 完了
 - 保留
- values
 - (空欄)
 - waiting
 - working
 - completion
 - pending

これで、プルダウン要素の入力値の設定が完了しました。

画面の表示非表示の修正

続けて、画面の表示、非表示の設定を行います。

「ステータス」は登録のタイミングでは自動で「未着手」を設定します。
 そのため、登録画面では「ステータス」を非表示に、更新、参照画面では表示するようにします。
 画面に配置された、「ステータス」のフィールドを選択します。



画面右側の「プロパティ」タブの「共通」 - 「表示/非表示」の「変数値」を選択し、入力ボックスに以下の値を入力します。

```
「= $input.__mode__ != $constant.mode.register」
```

定数である「\$constant.mode」には、表示する画面の種別がテンプレートで以下のように設定されています。


キー	画面種別
register	登録画面
edit	編集画面
refer	参照画面

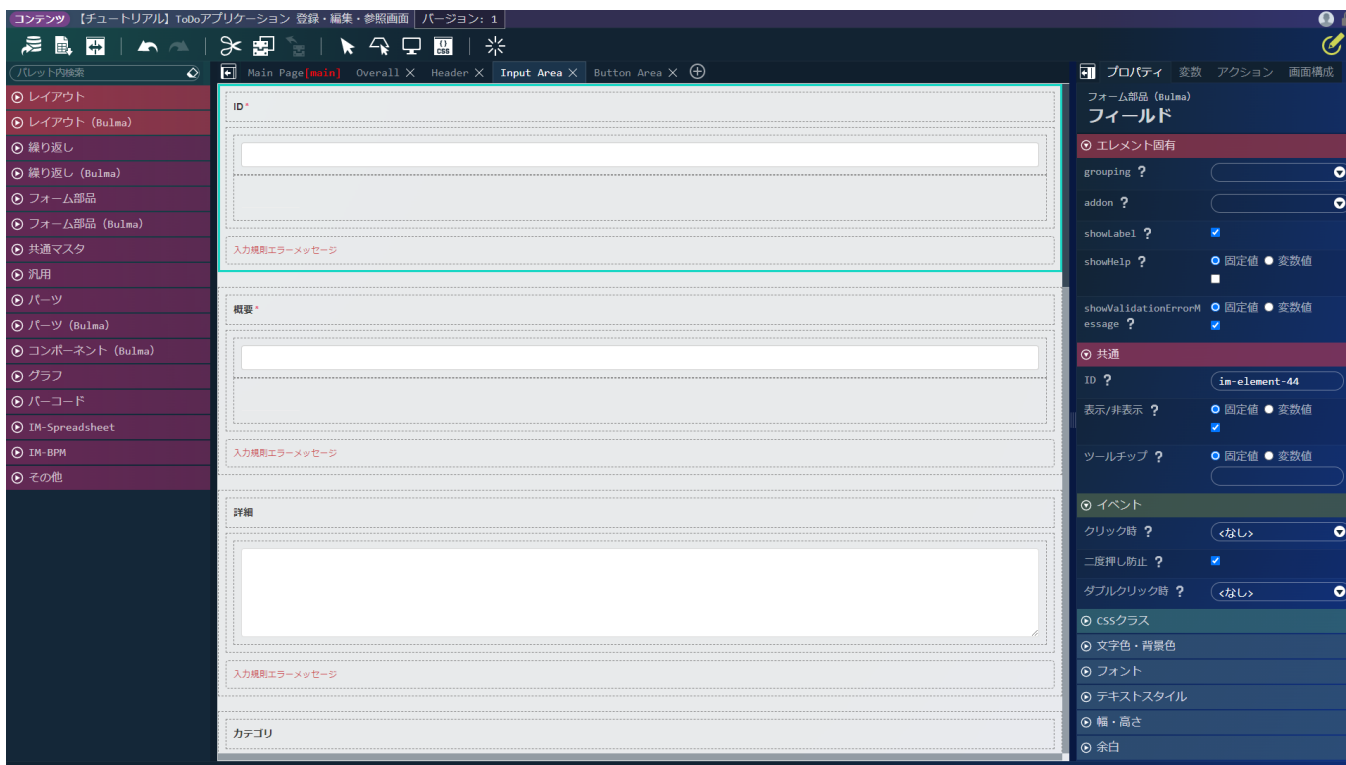
キー	画面種別
list	一覧画面

「\$input.__mode__」はテンプレートで作成された前処理によって、画面を表示した際に上記のいずれかの値が格納されます。そのため、この2つの変数の比較によって、表示画面毎の処理の切り替えが行えます。

不要な入力項目の削除

業務ロジックの修正で自動設定とした、「ID」や「登録者」などの情報を画面上の入力項目から削除します。

IDのフィールドを選択し、Deleteキーを押すか、画面上部の  をクリックして削除します。




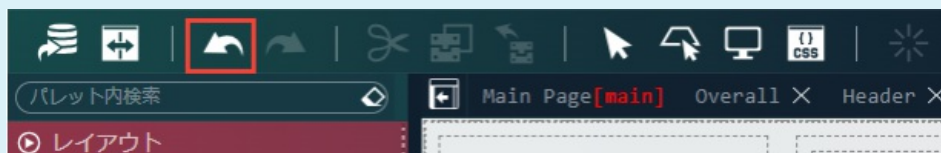
同様に、「担当者」についても今回のチュートリアルでは自動設定するため、入力項目から削除します。

以下の表示項目についてもデータベース上での管理用の項目のため削除していきます。

- 登録者
- 登録日
- 更新者
- 更新日


コラム

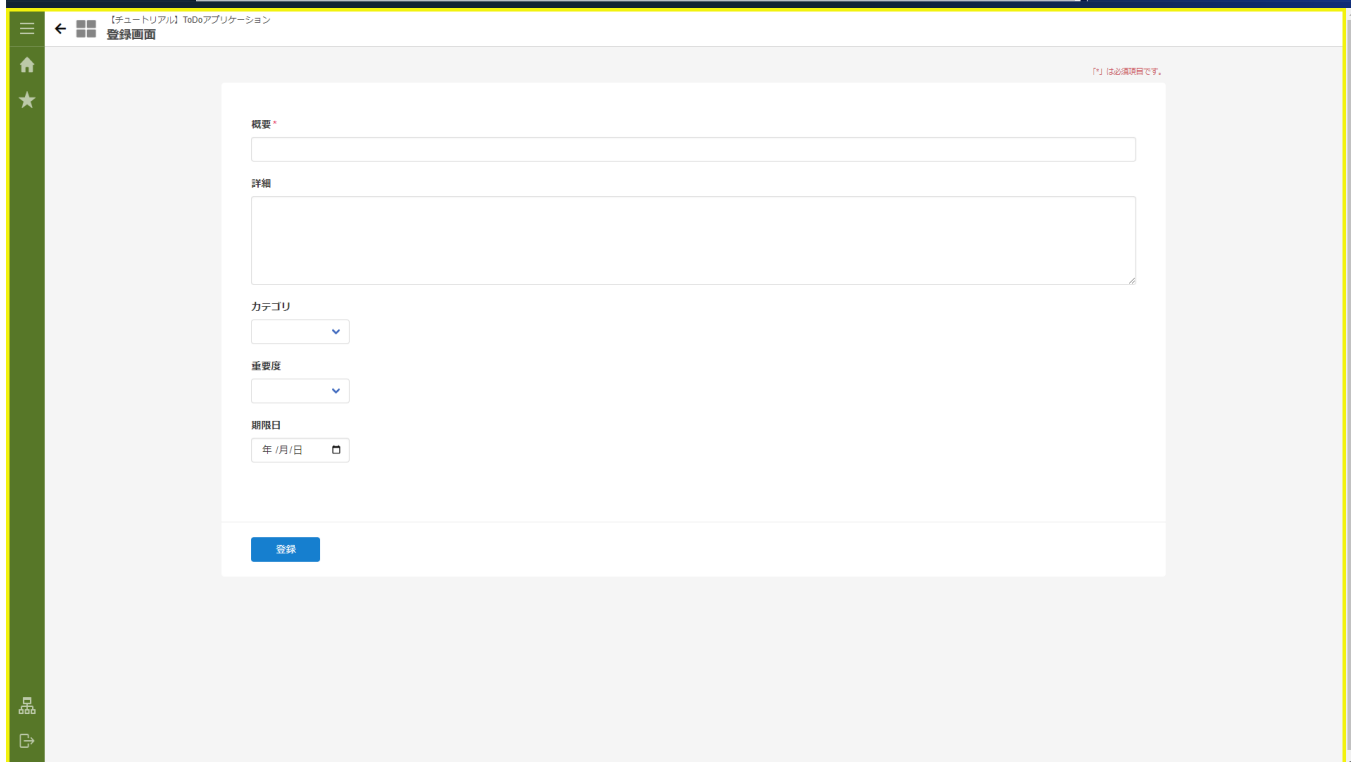
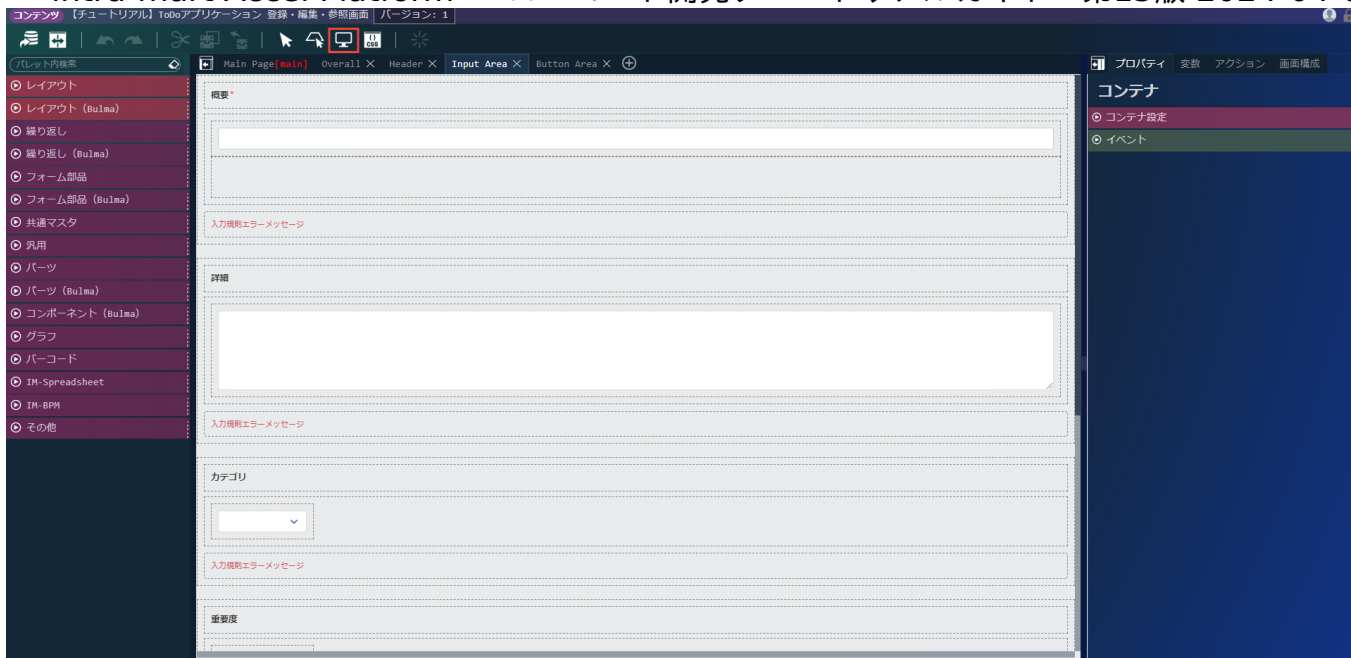
画面項目の削除で誤った項目を削除してしまった場合など、前の操作を取り消したい場合は画面上部の  か、「Ctrl+z」で直前の操作を取り消せます。



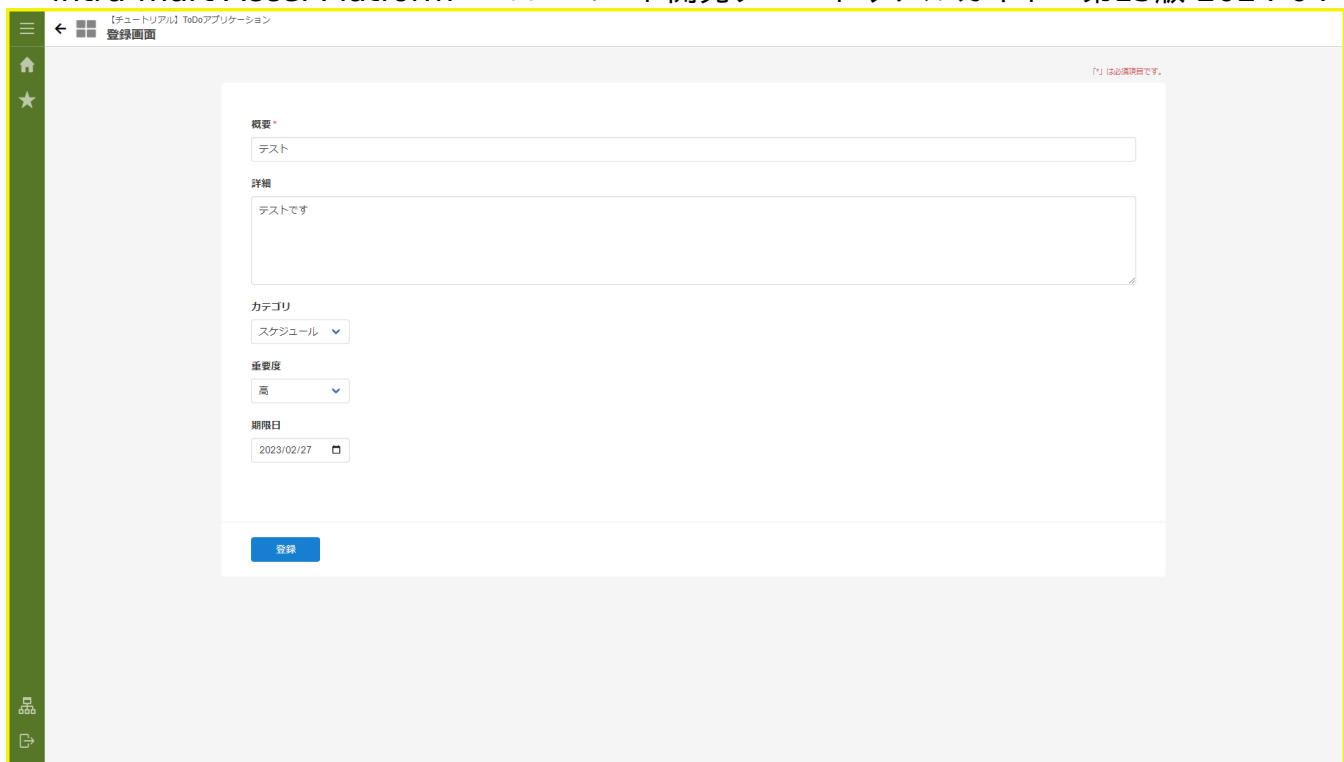
動作確認と必須設定の修正

これまでの操作で画面には必要な項目のみとなったため、動作確認を行きましょう。

画面上部の  をクリックしてください。



プレビュー画面に登録画面が表示されました。
 以下のように値を入力し、登録ボタンをクリックします。

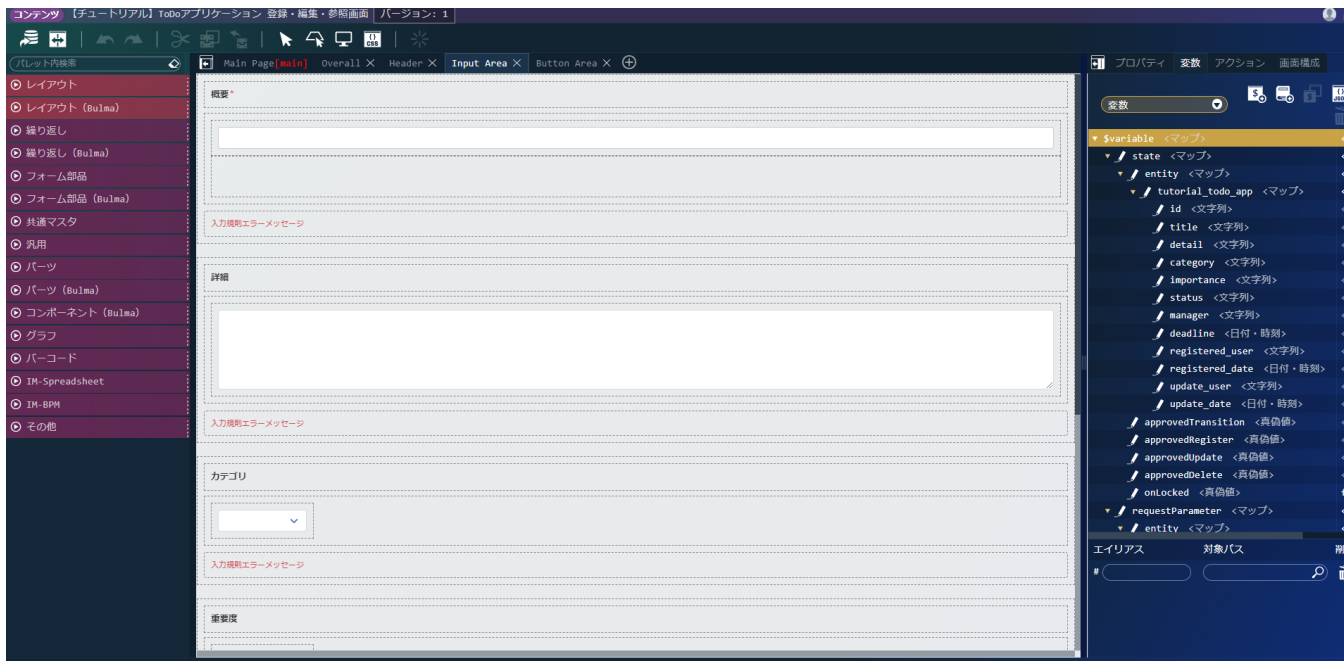


登録ボタンを押下しても、画面は動作しません。

これは、登録処理等に利用している削除対象の変数が必須設定されているためです。

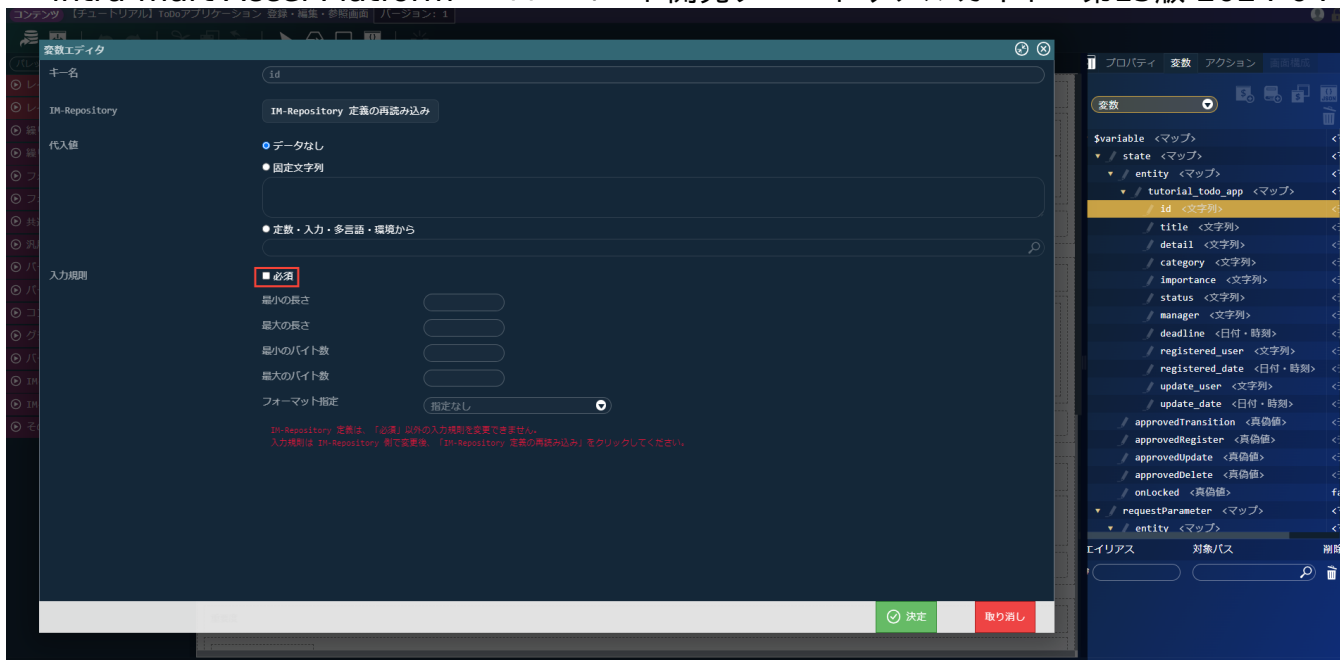
変数の必須設定を削除します。

ブラウザの「デザイン編集」タブに戻り、デザイン編集画面の右側にある変数タブを選択します。




「\$variable > state > entity > tutorial_todo_app > id」の  を選択します。

変数エディタの入力規則欄にある「必須」のチェックを外して決定を選択します。



同様に「\$variable > state > entity > tutorial_todo_app」に設定されている「manager」、「registered_user」、「registered_date」、「update_user」、「update_date」の「必須」チェックを外します。再度、動作確認を行ってみましょう。登録が完了し、一覧画面への遷移が行われました。

これで、操作画面の修正は完了です。

ブラウザの「デザイン編集」タブに戻り、画面上部の  で編集内容を保存してください。

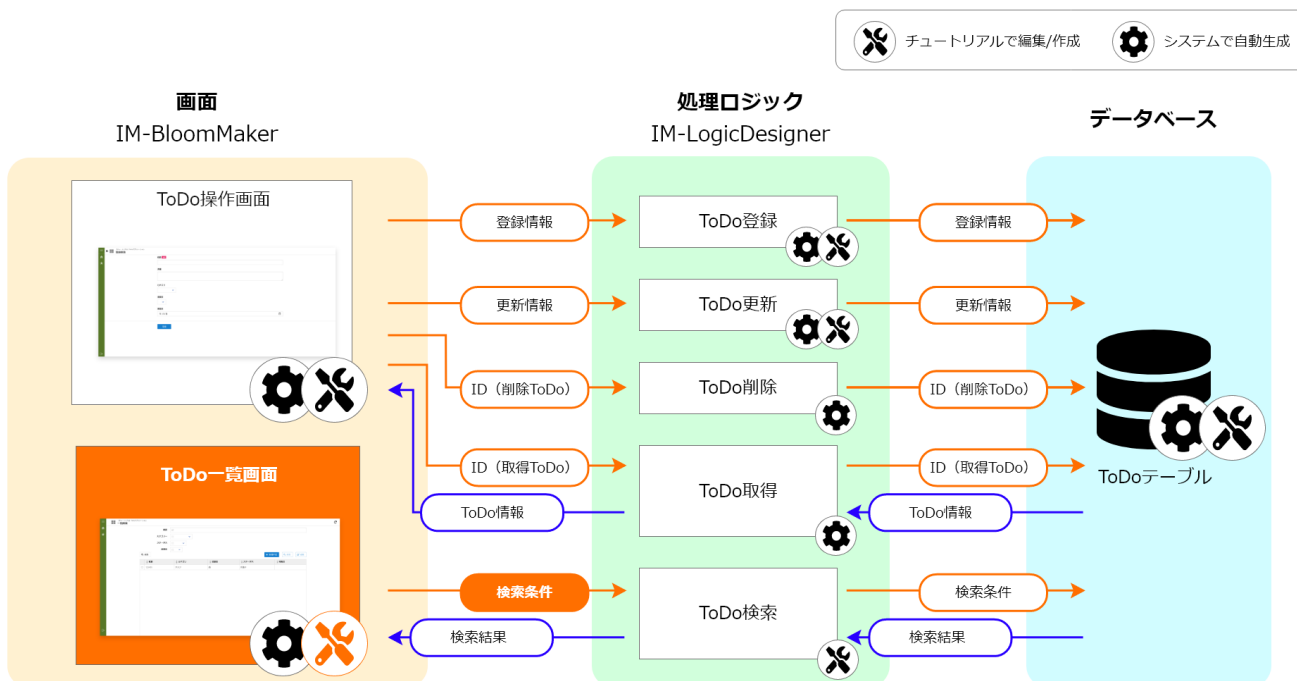
一覧画面の修正

前のページではToDoの登録や更新を行う画面から不要な項目や表示するデータについて修正を行いました。

一方で、一覧画面はテンプレートで作成されたままであるため、操作画面同様に不要な項目が表示されていたり、データベースから取得したそのままのデータが表示されてます。

ここからは、一覧画面の表示項目を変更したり、データベースから取得してきたデータを表示用に変換するための修正を行います。

以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



このページの手順について、以下の動画からも確認できます。併せてご利用ください。

**注意**

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

表示項目の修正

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】ToDoアプリケーション」の順に遷移します。

アプリケーション管理画面のアプリケーション構成情報で「画面」のタブを選択します。

「【チュートリアル】ToDoアプリケーション一覧画面」を選択します。

IM-BloomMaker コンテンツ画面で「デザイン編集」を選択します。

IM-BloomMakerの「デザイン編集画面」が表示されます。

不要な表示項目の削除

まず、操作画面の入力、表示項目の修正を行います。

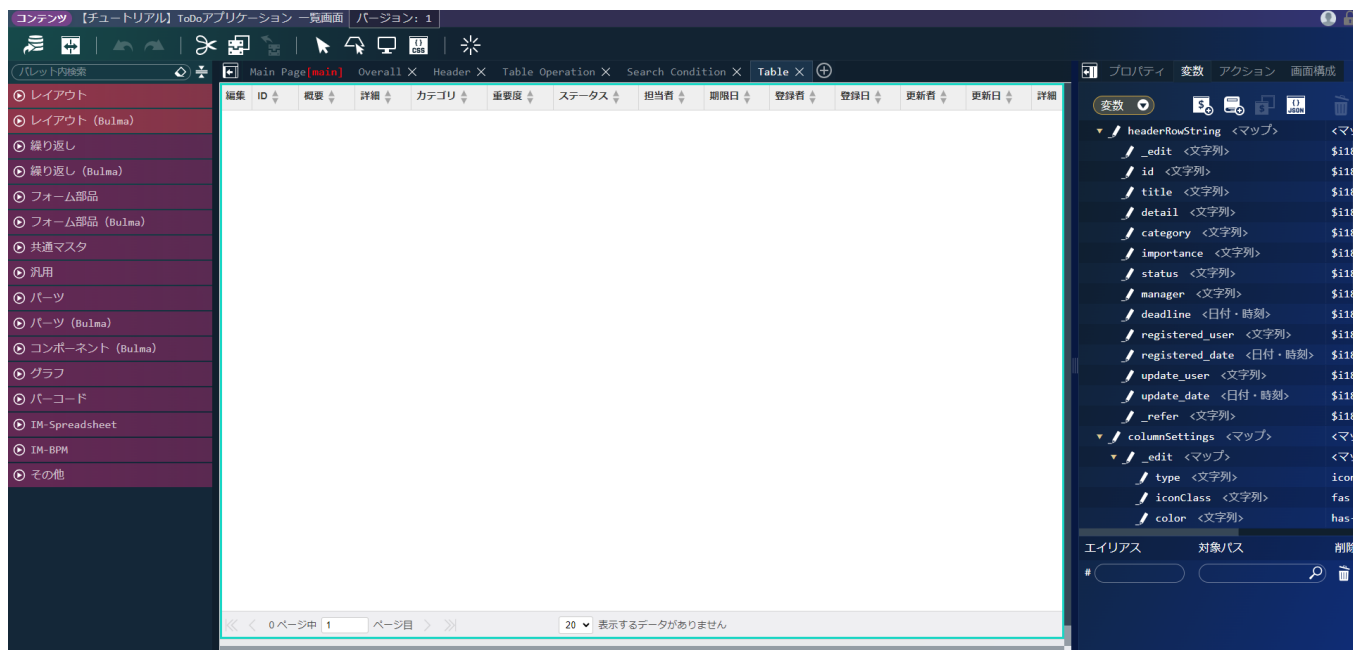
画面上部のタブから「Table」タブを選択してください。

現状の一覧画面ではすべての項目が表示されています。ここから「ID」や「担当者」といった管理用の項目、文字数の多い「詳細」を一覧に表示させないように修正します。

「マスタメンテナンスプレート」から作成した画面の一覧は「リッチテーブル」エレメントを利用して作成されます。

「リッチテーブル」はエレメント固有の設定である「headerRowString」に設定した変数によって表示項目が変わるため、テンプレートで自動作成された変数を削除します。

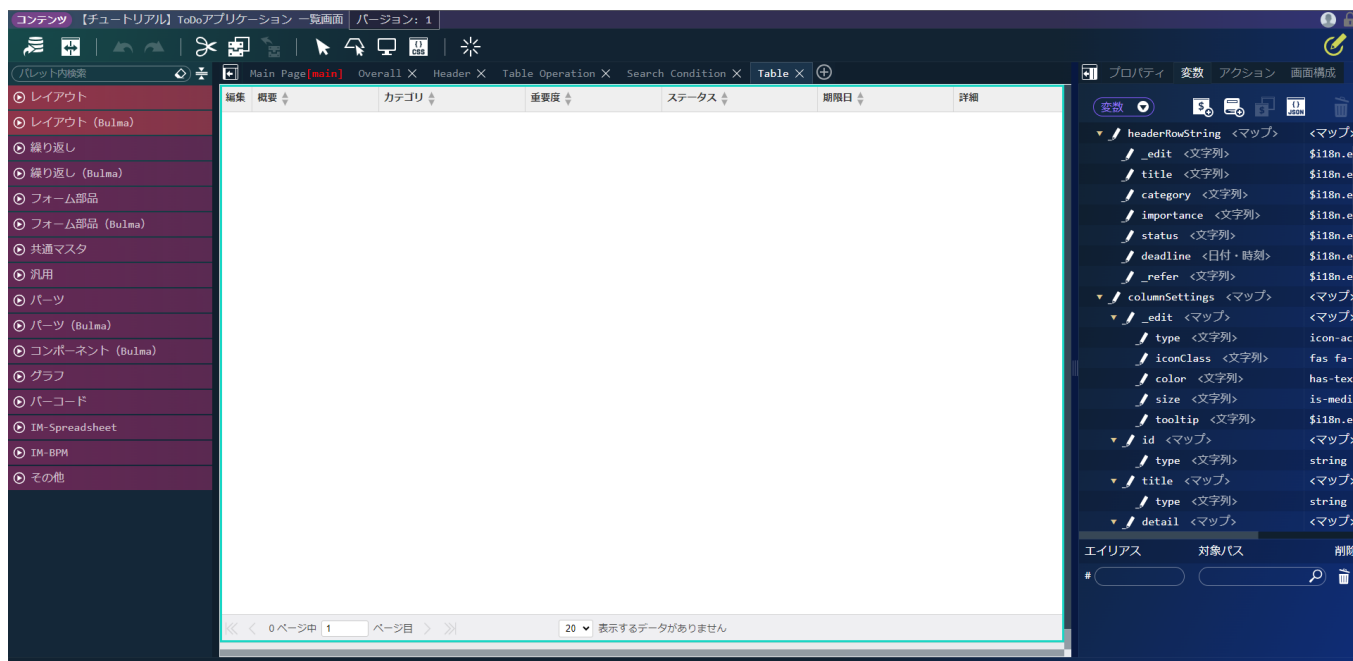
画面右側で変数タブを選択します。



変数の「\$variable > headerRowString > id」を選択し、 をクリックして削除します。

同様に、「\$variable > headerRowString」配下の以下の変数も削除します。

「detail」、「manager」、「registered_user」、「registered_date」、「update_user」、「update_date」



表示内容の修正

次に、表示する値をデータベースに格納された値ではなく、ユーザが確認しやすい形式に変換するための修正を行います。また、現状はテンプレートから生成された一覧取得処理を利用しているため、全ユーザが登録したToDoが表示されます。これについても、操作ユーザのToDoのみが表示されるよう、併せて修正します。

まず、「カテゴリ」をユーザが確認しやすい値とするために、表示用の定数を作成します。

画面右側で変数タブを選択します。

表示を「変数」から「定数」に変更します。

「\$constant」を選択し、 をクリックします。

定数エディタ画面でキー名を「category」、値の型を「マップ」に設定します。


作成した「category」に以下の定数を設定します。

キー名	値の型	代入値
task	文字列	タスク
schedule	文字列	スケジュール
move	文字列	移動
memo	文字列	メモ

定数はJSON入力で以下のコードを記載しても設定できます。

```

"category": {
  "task": "タスク",
  "schedule": "スケジュール",
  "move": "移動",
  "memo": "メモ"
}
    
```

JSON入力する場合は、定数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いように注意してください。

次に、「重要度」の表示用の定数を作成します。


「\$constant」配下に、キー名を「importance」、値の型を「マップ」とした定数を追加します。

以下の定数を設定します。

キー名	値の型	代入値
high	文字列	高
medium	文字列	中
low	文字列	低

定数はJSON入力で以下のコードを記載しても設定できます。

```
"importance": {
  "high": "高",
  "medium": "中",
  "low": "低"
}
```

JSON入力する場合は、定数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いように注意してください。

次に、「ステータス」の表示用の定数を作成します。


「\$constant」配下に、キー名を「status」、値の型を「マップ」とした定数を追加します。

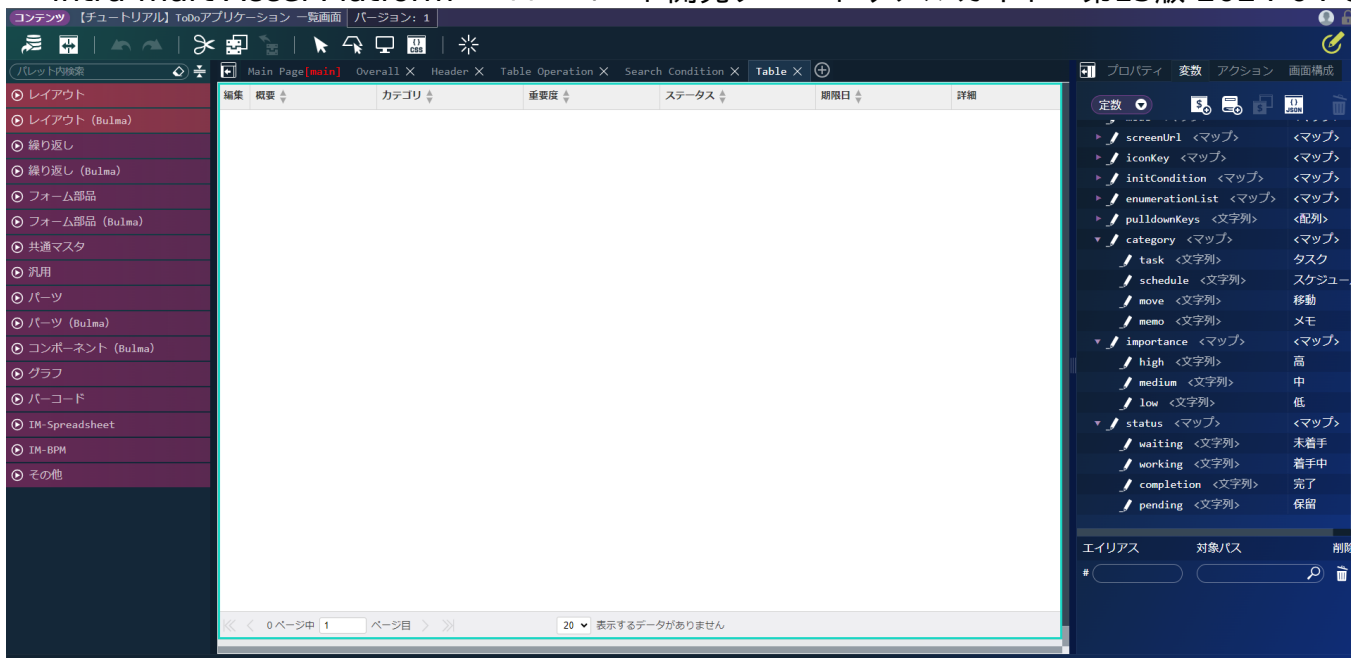
以下の定数を設定します。

キー名	値の型	代入値
waiting	文字列	未着手
working	文字列	着手中
completion	文字列	完了
pending	文字列	保留

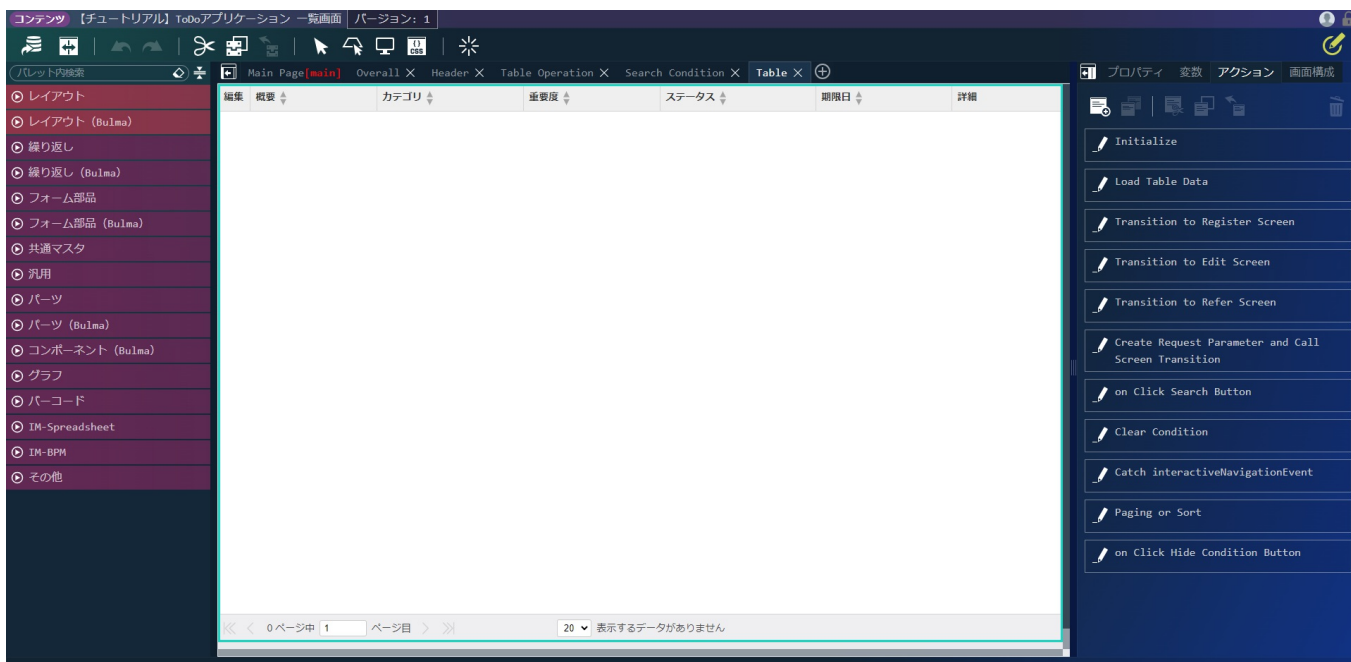
定数はJSON入力で以下のコードを記載しても設定できます。


```
"status": {
  "waiting": "未着手",
  "working": "着手中",
  "completion": "完了",
  "pending": "保留"
}
```

JSON入力する場合は、定数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いように注意してください。



次に、一覧の表示を整え、操作ユーザが登録したToDoのみが表示されるように修正します。
画面右側で「アクション」タブを選択します。



「Load Table Data」の  を選択します。

アクションエディタ画面で「標準」タブにある「変数○に○を代入する」をドラッグ&ドロップで一番上に配置します。



左側の入力エリアに変数値で「\$variable.endPoint.list.requestParameter.searchCondition.manager」と設定します。
 右側の入力エリアは「abc▼」をクリックして変数値での入力に切り替えた後、「\$env.accountContext.userCd」と設定します。



次に、表示項目をデータベースに格納された値ではなく、ユーザにわかりやすい状態に変更します。
 取得した値を元に、作成した「カテゴリ」、「重要度」、「ステータス」の定数が一覧の値として表示されるように設定します。
 アクションエディタ画面で「標準」タブにある「カスタムスクリプトを実行する」を、「IM-LogicDesigner フロールーティング○にリクエストを送信する」アクションの次（上から3番目）に追加します。



「カスタムスクリプト」に以下のコードを設定します。

```
let records = $variable.endPoint.list.responseData.data.records
for(let i = 0; i < records.length; i++){
  if(records[i].category)records[i].category = $constant.category[records[i].category]
  if(records[i].status)records[i].status = $constant.status[records[i].status]
  if(records[i].importance)records[i].importance = $constant.importance[records[i].importance]
}
```

アクションエディタの「決定」ボタンをクリックし、保存します。

検索機能の修正

「マスタメンテナンスプレート」から作成した一覧画面には、[テンプレートからアプリケーションのベースを作る](#) で検索条件に指定した項目でデータを絞り込む検索機能が実装されています。

検索機能を正常に使えるようにするため、各検索項目エレメントの修正を行います。

検索項目の修正

画面上部のタブの中から「Search Condition」タブを選択します。

「Search Condition」タブは検索項目が設定された画面です。

「カテゴリ」「重要度」「ステータス」のプルダウンの入力項目を検索に適した値となるように修正し、不要な検索項目を削除します

テンプレートから自動生成された画面のプルダウン要素のlabels、valuesには変数値が設定されています。

カテゴリのプルダウンから設定します。画面に配置された「カテゴリ」のプルダウン要素を選択します。

画面右側に表示される「プルダウン要素」の「エレメント固有」 - 「Labels」のラジオボタンで固定値を選択し、以下のように設定します。

「Labels」はプルダウンの表示要素です。

1行目には何も入力せずに改行するように注意してください。

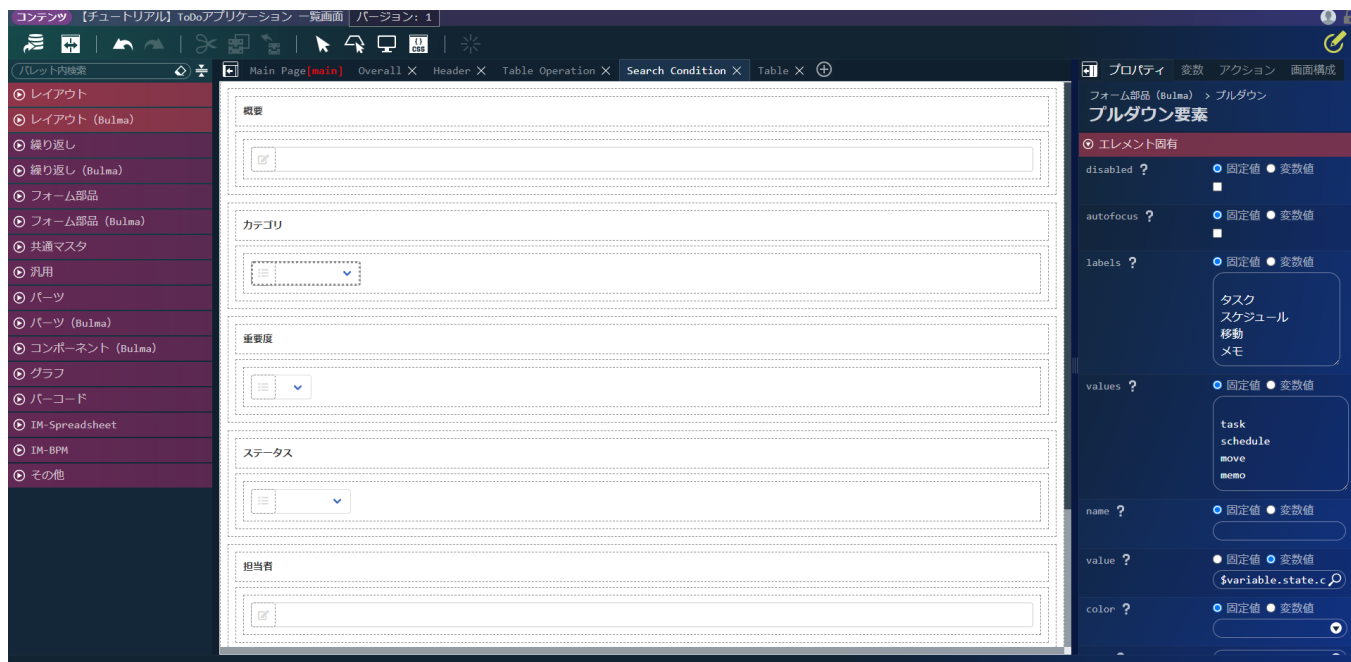
- labels
 - (空欄)
 - タスク
 - スケジュール
 - 移動

- メモ

同様に「エレメント固有」 - 「values」のラジオボタンで固定値を選択し、以下のように設定します。「values」はプルダウンを選択した際に送信される値の要素です。

「labels」同様に、1行目には何も入力せずに改行するように注意してください。

- values
 - (空欄)
 - task
 - schedule
 - move
 - memo



「labels」と「values」は同じ行番号の値同士が対応して選択値と送信値として設定されます。

例えば、上記の設定を行った場合、プルダウンで3行目の「スケジュール」を選択した場合、「values」の3行目である「schedule」が送信値として「value」に指定した変数に設定されます。

「Search Condition」タブで

同様にほかのプルダウン要素も設定していきます。「重要度」の「labels」と「values」に以下のように設定します。

- labels
 - (空欄)
 - 高
 - 中
 - 低
- values
 - (空欄)
 - high
 - medium
 - low

次は、「ステータス」の「labels」と「values」を以下のように設定します。

- labels
 - (空欄)
 - 未着手
 - 着手中
 - 完了
 - 保留
- values
 - (空欄)
 - waiting

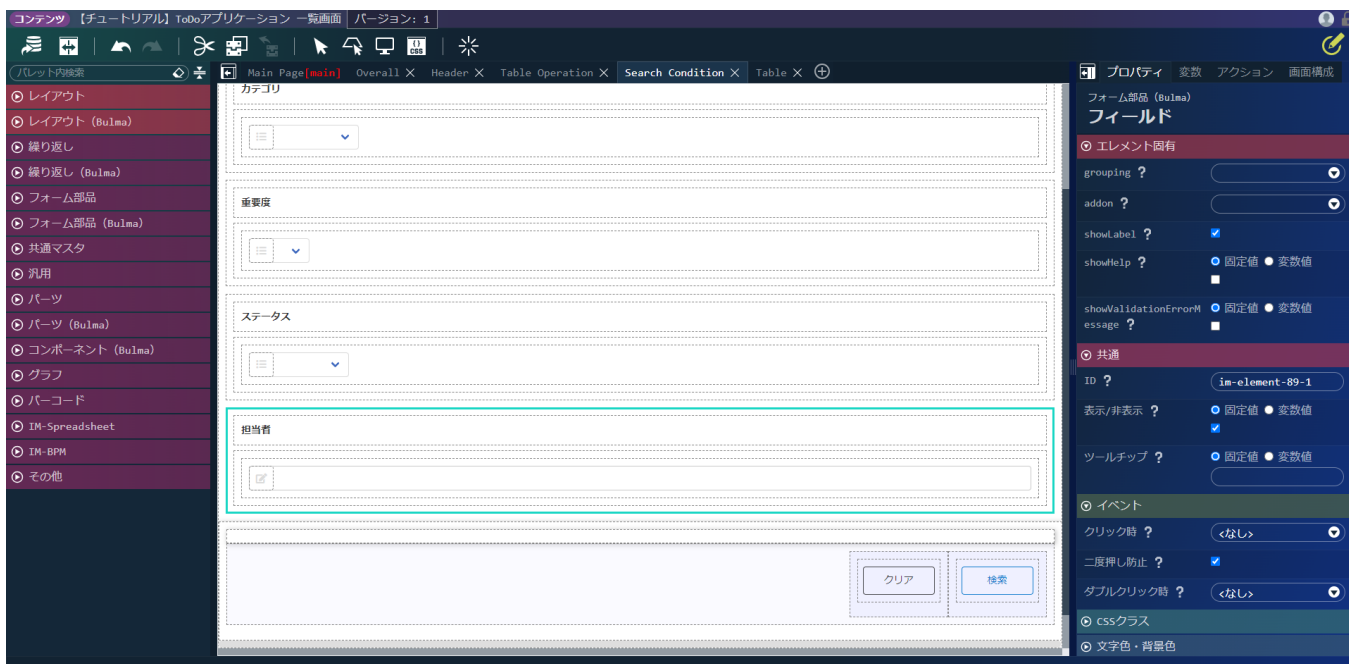
- working
- completion
- pending


これで、プルダウン要素の入力値の設定が完了しました。

不要な検索項目の削除

本アプリでは操作者が担当者として登録したデータのみを一覧に表示する仕様です。したがって、検索項目として「担当者」は不要になるため、不要な検索項目を削除します。

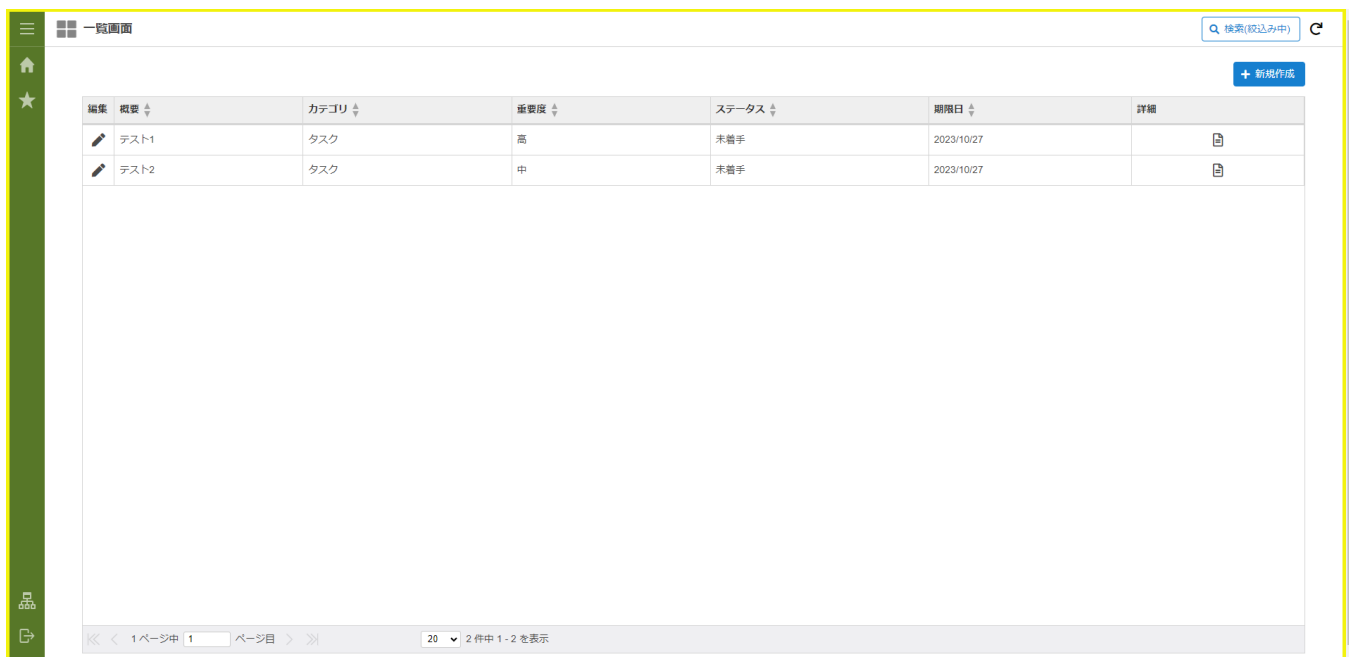
「Search Condition」タブで「担当者」項目のフィールドを選択します。



画面上部の  をクリックしてフィールドを削除します。

検索機能が動作するか確認を行きましょう。

画面上部の  を押下してください。



検索欄を埋めて検索してみましょう。

これで一覧画面の修正は完了です。

メニューの設定

作成したアプリケーションに対して、メニューからアクセスできるように設定します。
ToDoの一覧画面をメニューから呼び出せるように設定します。



注意

Accel-Mart Quick環境をご利用の場合は、メニュー設定画面への導線が異なります。また、メニューグループの新規作成は行えません。

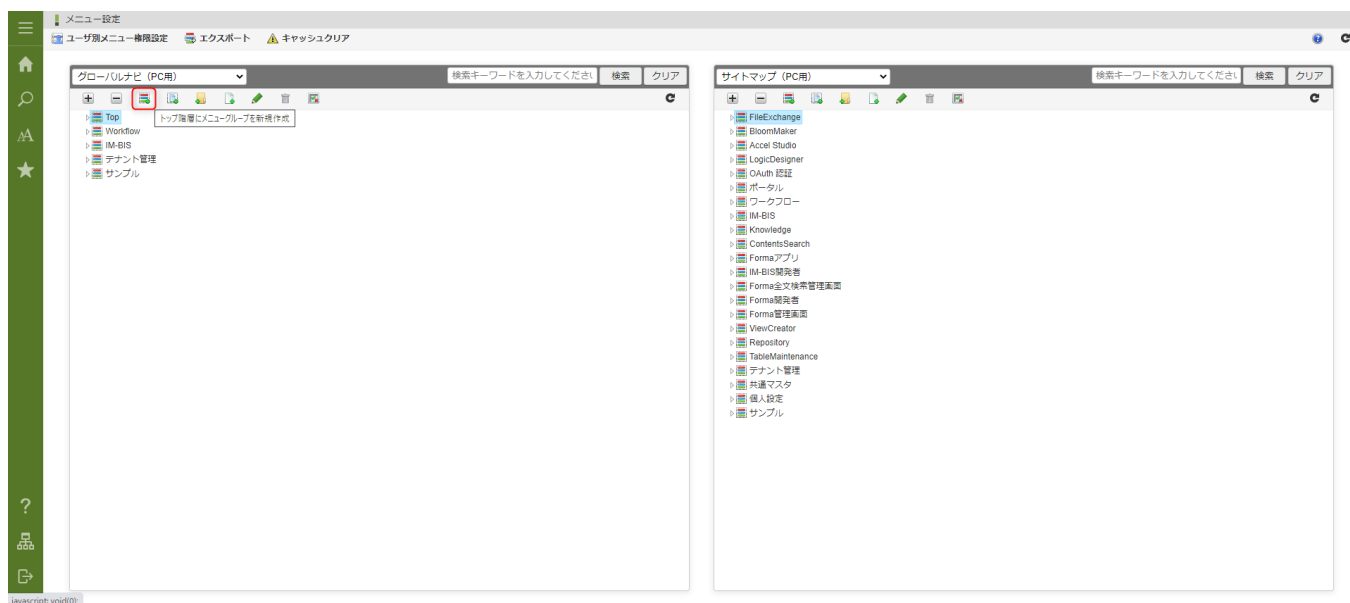
Accel-Mart Quick環境でのメニュー設定手順については、「[Accel-Mart Quick 開発者ポータル](#)」の「[メニュー設定](#)」を参照してください。

メニューの設定

「テナント管理」 - 「メニュー」 から「メニュー設定」画面を表示します。

- テナント管理
 - 認可
 - **メニュー**
 - ユーザ別メニュー権限
 - ロール
 - カレンダーメンテナンス
 - 📁 テナント情報
 - カレンダー
 - タイムゾーン
 - 日付と時刻の形式
 - テーマ
 - アカウントライセンス
 - アカウントロック
 - アプリケーションライセンス
 - ホームURL
 - メールアドレス
 - パスワードリマインダ設定
 - アナウンス設定
 - メニュー表示
 - サイトヘルプメニュー設定
 - 数値形式
 - セーフURL
 - パスワードポリシー設定
 - 📁 ジョブ管理
 - ジョブ設定
 - ジョブネット設定
 - ジョブネットモニタ
 - ファイル操作
 - データベース操作
 - アイコン管理
 - 多要素認証設定
 - 外部システムユーザ連携


 をクリックし「メニューグループの新規作成」ダイアログを表示します。

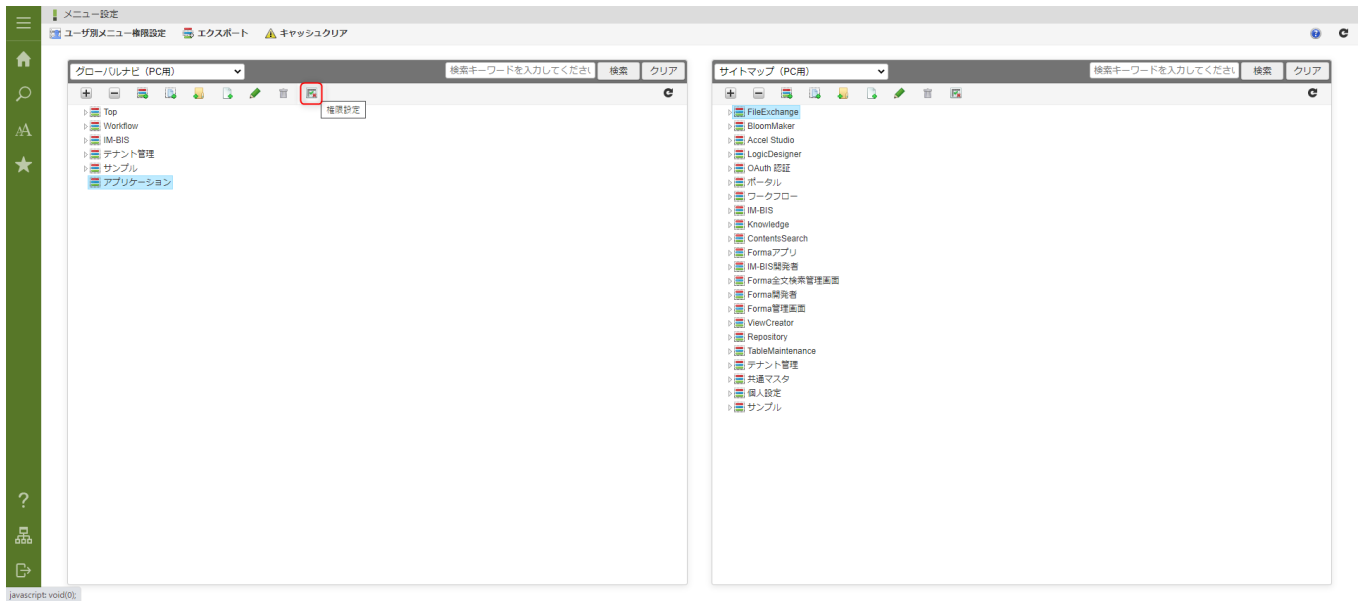


「メニューグループの新規作成」ダイアログに以下を設定します。

ラベル	入力値
メニューグループID	application
メニューグループ名	アプリケーション

入力出来たら、ダイアログ下部の「新規作成」をクリックし、ダイアログを閉じます。

作成した「アプリケーション」メニューグループを選択し  をクリックし、「権限設定」画面を表示します。

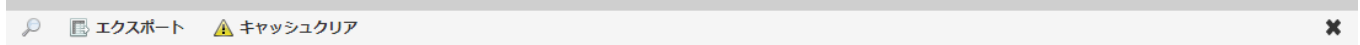


「権限設定を開始する」をクリックします。

作成した「アプリケーション」メニューグループの「認証済みユーザ」の「管理」と「参照」をクリックし許可状態に変更します。

設定出来たら「権限設定を終了する」をクリックし、右上の閉じるアイコンから「権限設定」画面を閉じます。

権限設定 (グローバルナビ (PC用))




アクションの種類: 全てのアクション

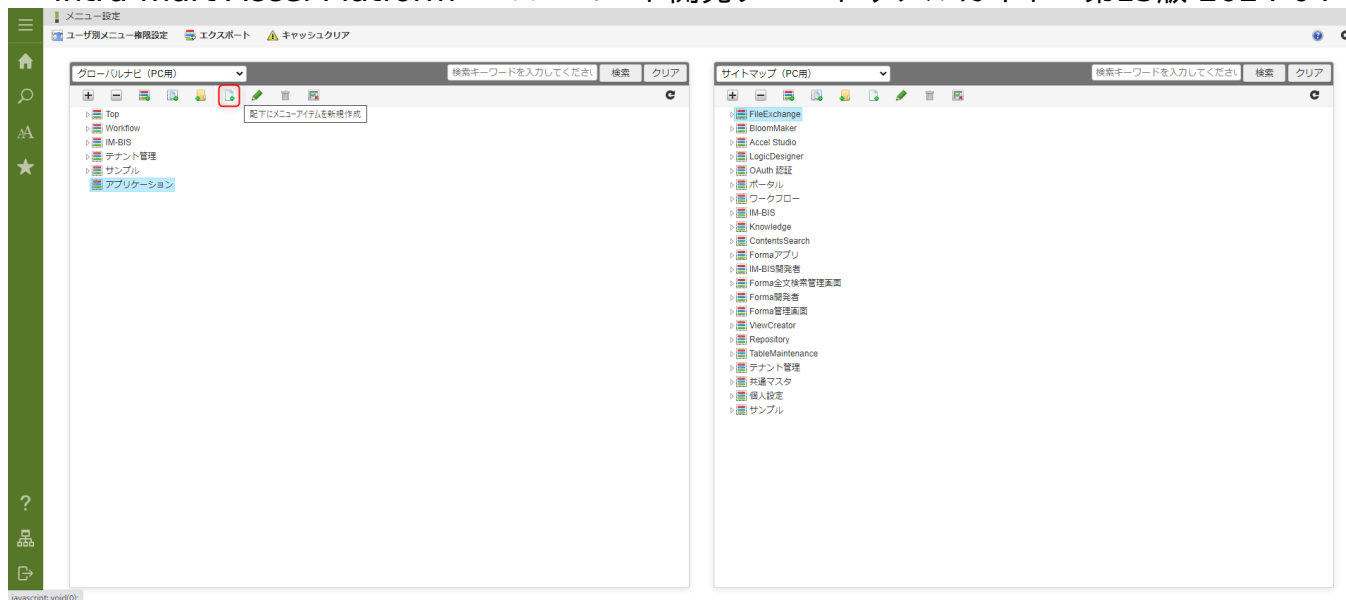
権限設定を開始する

リソース	アクション	認証		組織		ロール											
		ゲストユーザ	認証済みユーザ	サンプル会社	その他会社	テナント管理者	認可管理者	管理	メニュー管理者	メニュー通用管理者	アカウント管理者	ロール管理者	カレンダー管理者	ジョブスケジューラ管理者	IM共通管理者	IMマスタ	
メニューグループ	グローバルナビ (PC用)																
Top	管理																
	参照																
Workflow	管理																
	参照																
IM-BIS	管理																
	参照																
テナント管理	管理																
	参照																
サンプル	管理																
	参照																
アプリケーション	管理																
	参照																

注意

Accel-Mart Quick 環境ご利用の方は「アプリケーション」配下のみメニューの設定が可能です。

設定を行いたい「アプリケーション」メニューグループを選択し  をクリックし「メニューアイテムの新規作成」ダイアログを表示します。



ToDo一覧画面のメニュー設定を行います。
「メニューアイテムの新規作成」ダイアログに以下を設定します。

ラベル	入力値
メニューアイテムID	tutorial_todo_list
メニューアイテム名	ToDo一覧
URL	accel-studio-app/tutorial_todo_app/list

「メニューアイテムの新規作成」ダイアログの「新規作成」ボタンをクリックし、メニューを登録します。

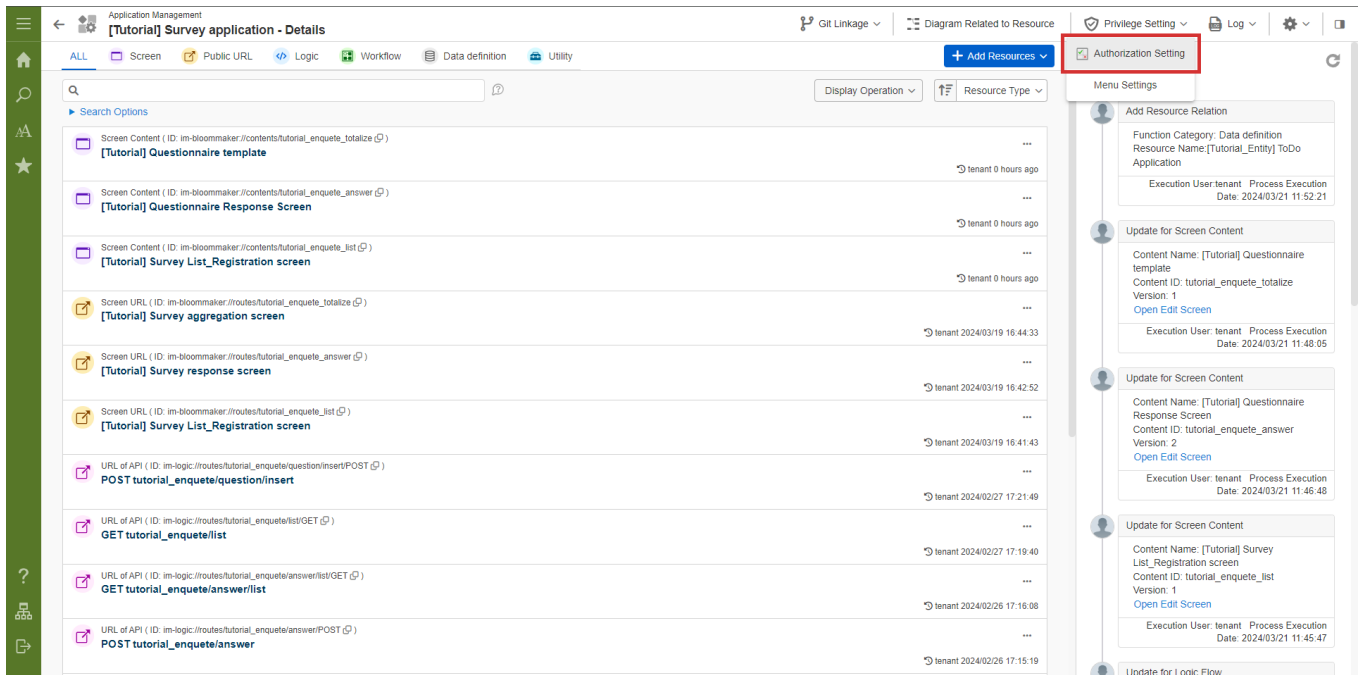


これで、メニューの設定は終了です。続けて、作成したアプリケーション全体の権限設定を行います。

作成したアプリケーションを利用するには、メニューへの権限のみではなく、アプリケーションへの認可設定も必要です。
作成したアプリケーションは、標準では「テナント管理者」ロールに対してのみ認可が設定されています。
このままでは「テナント管理者」ロールを持ったユーザ以外がアプリケーションを利用できません。以降ではアプリケーションの認可設定を行います。

権限設定

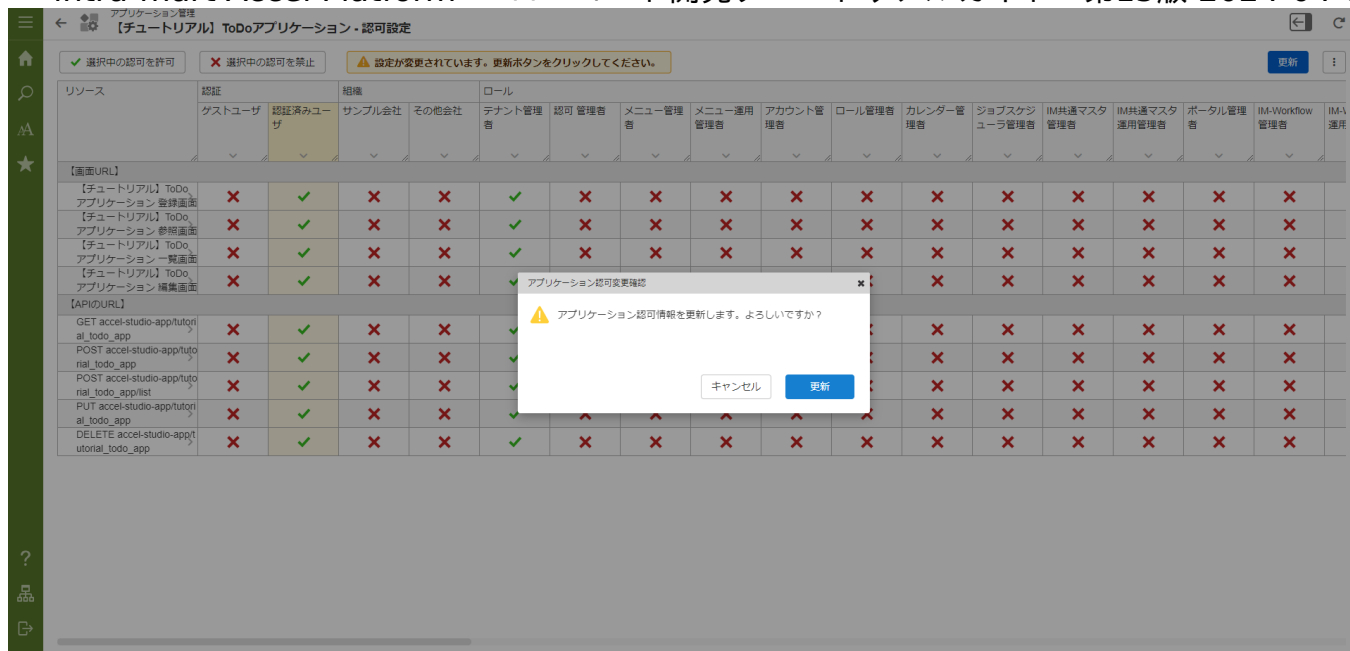
アプリケーションの権限設定はアプリケーション管理から行います。
「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」の順に遷移します。
アプリケーション一覧から、作成した「【チュートリアル】ToDoアプリケーション」を選択します。
画面右上の「権限設定」プルダウンから認可設定アイコンを選択します。



アプリケーション認可設定画面で、「認証」 - 「認証済みユーザ」の列を選択し、「選択中の認可を許可」をクリックします。

リソース	認証				権限				ロール											
	ゲストユーザ	認証済みユーザ	サンプル会社	その他会社	テナント管理者	認可管理者	メニュー管理者	メニュー運用管理者	アカウント管理者	ロール管理者	カレンダー管理者	ジョブスケジューラ管理者	共同運用管理者	共同運用運用管理者	ポータル管理者	IM-Workflow管理者	IM-Workflow運用管理者	IM-Workflow監査者	IM-Workユーザ	
【画面URL】																				
【チュートリアル】ToDoアプリケーション登録画面	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【チュートリアル】ToDoアプリケーション検索画面	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【チュートリアル】ToDoアプリケーション一覧画面	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【チュートリアル】ToDoアプリケーション編集画面	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【API/URL】																				
POST tutorial_todo_app/list/search	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
POST accel-studio-app/tutorial_todo_app	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
GET accel-studio-app/tutorial_todo_app	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
GET accel-studio-app/tutorial_todo_app/list	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
PUT accel-studio-app/tutorial_todo_app	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
DELETE accel-studio-app/tutorial_todo_app	×	✓	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×

「更新」をクリックし、更新を行います。



ログイン可能なすべてのユーザが作成したアプリケーションを利用できるようになりました。
これでToDoアプリケーションの作成、利用設定が完了しました。

ワークフローを利用してアプリケーションを作成する

概要

本章では、ローコード開発により簡易的な日報を提出するアプリケーションを作成します。



注意

本チュートリアルでは「[環境について](#)」に記載の環境を前提に、機能間の遷移順序を記述しています。
詳しくは上記リンクからご確認ください。

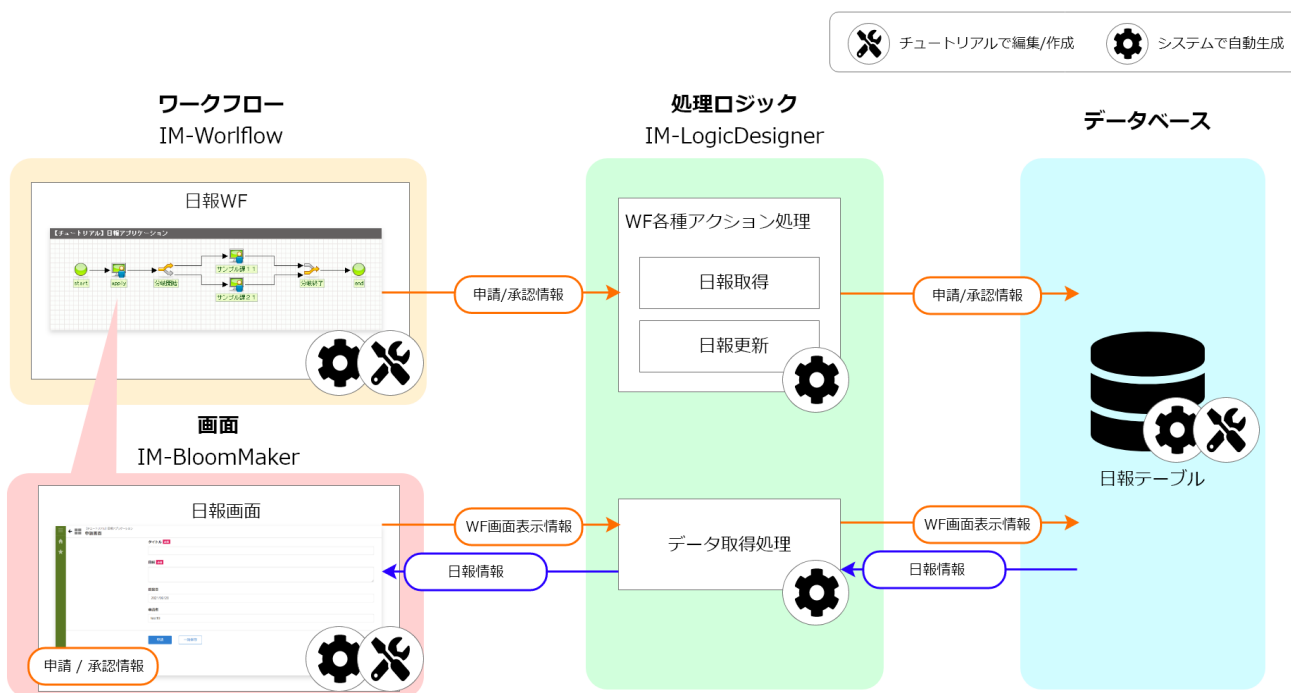
作成の流れ

本チュートリアルは以下の流れでアプリケーションを作成します。

1. アプリケーション作成にてテンプレートからデータベース、処理、画面のベースを自動作成
2. データベースの操作
3. IM-BloomMakerにて、画面の修正
4. ワークフローの修正



テンプレートから自動で作成される業務ロジックについては、本チュートリアルでは変更しません。
作成される処理や画面は以下の画像のようなイメージです。

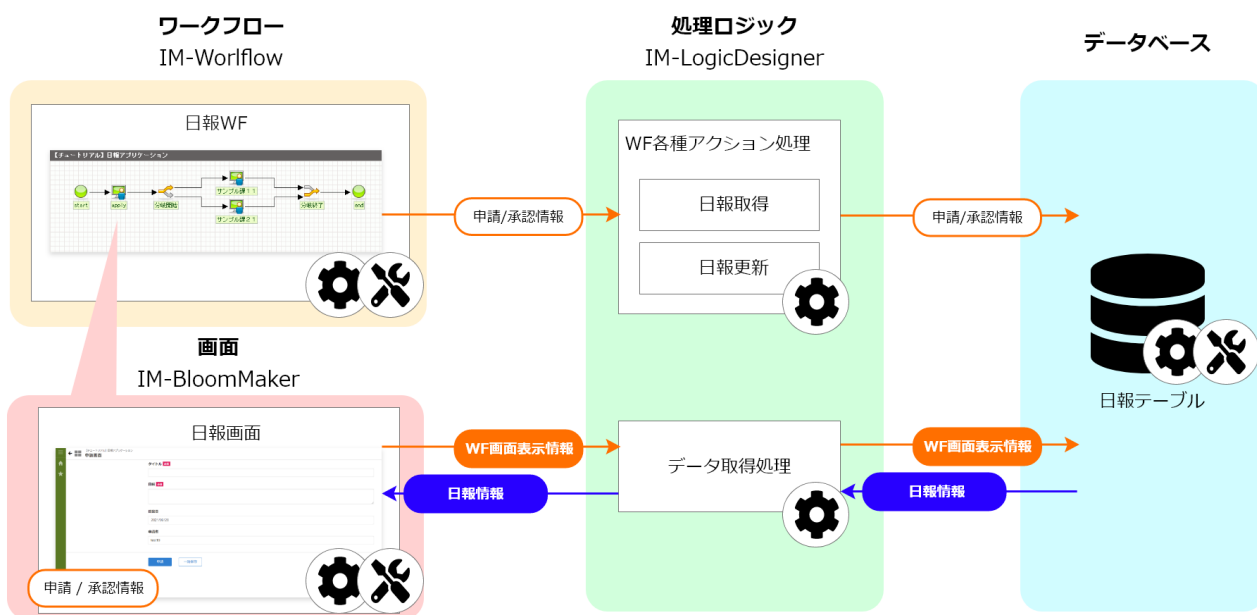


日報アプリケーションの仕様

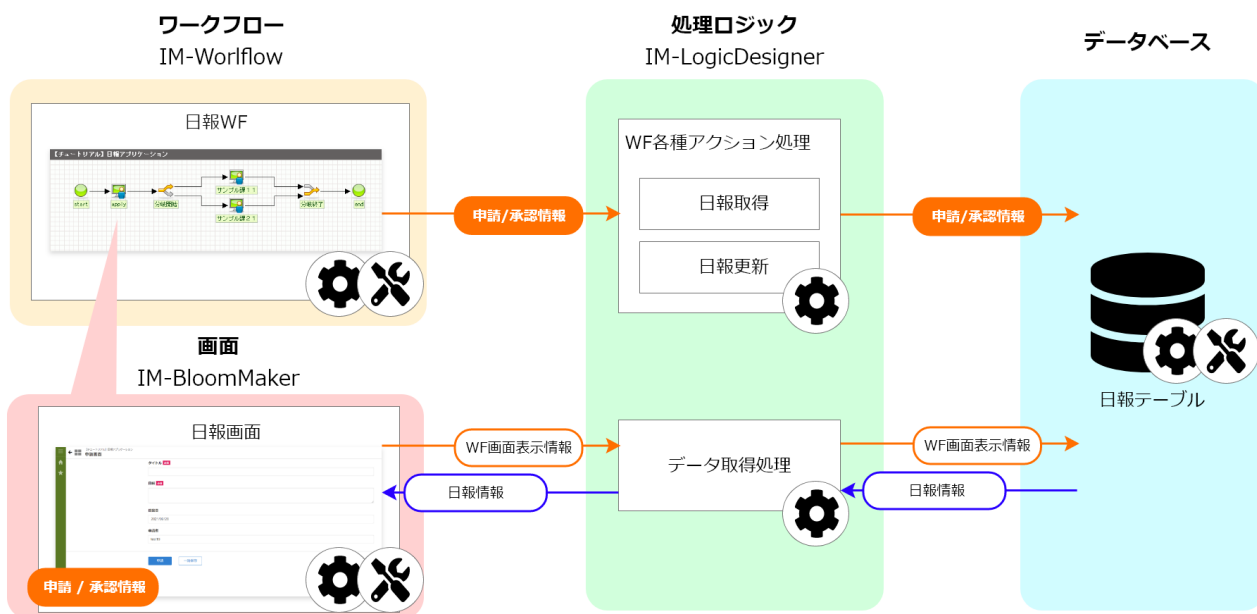
作成する日報アプリケーションは、ワークフローの申請画面から日報の作成、提出を行えます。

提出した日報はワークフローの案件一覧画面から処理、参照できます。

申請画面や承認画面表示時にどの処理画面を表示したのかを判別し、以下の図にあるように、データ取得処理を実行して表示する日報情報を取得します。



申請するタイミングでは、次のノードを選択できます。また、承認時には承認者がコメントを入力できます。
 申請、承認処理を行う際には、以下の図にあるように、画面で入力した「申請/承認情報」をワークフロー情報として処理します。
 また、ワークフロー情報を元にIM-LogicDesignerでのアクション処理により、画面に入力した日報情報のデータベースへの登録、更新を行います。



注意事項

本アプリケーションではサンプルデータのユーザや組織を利用しています。
 チュートリアル中のユーザや組織の指定は必ずしもチュートリアル通りに入力しなくとも動作します。

テンプレートからアプリケーションのベースを作る

アプリケーション作成のテンプレートを利用し、これから作成していくアプリケーションの元となるアプリケーションを生成します。
 本章では、「シンプルなワークフロー」テンプレートを利用し、登録、更新などの処理をもったアプリケーションを生成します。

「シンプルなワークフロー」のテンプレートを利用したアプリケーションの生成

「サイトマップ」→「Accel Studio」→「アプリケーション作成」→「業務テンプレート一覧」の順に遷移します。
「ベーステンプレート」から「シンプルなワークフロー」テンプレートを選択します。



「シンプルなワークフロー」テンプレートを利用すると、扱うデータの定義に合わせたデータベースのテーブル、画面、処理を自動で生成できます。

また、申請対象者を設定することで、対象者の「申請一覧」画面に作成したアプリケーションを表示できます。

「シンプルなワークフロー」テンプレートを使用することで、自動生成される画面、処理は以下の通りです。

- 画面
 - ワークフロー画面
- 処理
 - WF完了案件削除処理
 - WF過去案件削除処理
 - WF未完了案件削除処理
 - WFアクション処理
 - ユーザーデータ削除処理
 - ユーザーデータ取得処理

i コラム

アプリケーションを作成する際、入力項目について不明点あれば各項目名の横にある **?** にカーソルを合わせてください。アプリケーション利用時にどのように利用される項目であるか、設定を変えるとどのような変化があるかの説明が表示されま



アプリケーション情報の設定

アプリケーション情報を設定します。

アプリケーションID、アプリケーション名、説明に以下の入力値を設定してください。

※アプリケーション情報に関しては、各項目に任意の入力値を設定してもかまいません。本チュートリアルは、下記の通りのアプリケーションID、アプリケーション名を入力した前提で進めます。

※「IDの決定に利用する文字列」はアプリケーションIDを入力すると自動的に入力されます。本チュートリアルでは自動入力された文字列をそのまま用いて進めます。

ラベル	入力値
アプリケーションID	tutorial_daily_report_app
アプリケーション名	【チュートリアル】日報アプリケーション
説明	シンプルなワークフローテンプレートを利用して作成する日報アプリケーションです。



扱うデータの定義設定

次に、アプリケーションで利用するデータの定義を設定します。

ここで設定した定義を元に、データベースのテーブル、画面、処理が作成されます。

ワークフローテンプレートのデータ定義は「IM-Repository」のデータリポジトリ機能を利用して作成します。

IM-Repository資材の設定には簡易と詳細の2つの設定方法が用意されています。

本チュートリアルでの設定は詳細に設定するため、「詳細に設定する」へチェックを入れます。

エンティティについての情報を設定します。

エンティティID、エンティティ名、登録先エンティティカテゴリ、エンティティに紐づくテーブル名に以下の値を設定してください。

ラベル	入力値	備考
エンティティID	tutorial_daily_report_app_entity	
エンティティ名	【チュートリアル_エンティティ】日報アプリケーション	
登録先エンティティカテゴリ	【チュートリアル】エンティティ	任意のカテゴリで構いません。本チュートリアルでは新規作成しています。
データベース種別	TENANT	
エンティティに紐づくテーブル名	tutorial_daily_report_app	

上記の設定で、エンティティに紐づく「tutorial_daily_report_app」というテーブルが作成できます。

「IM-Repository」の機能により、エンティティの変更に合わせ、データベースのテーブルおよびテーブルを利用する登録、更新処理も変更されます。

「エンティティ情報」の「+エンティティ項目を追加」 - 「辞書項目を新規作成」を選択してください。



「辞書項目新規作成」ダイアログで、基本設定を入力しましょう。

まずは、日報の「タイトル」を設定します。以下の表および画面を参考に設定してください。

ラベル	入力値	備考
登録先辞書項目カテゴリ	【チュートリアル】ワークフロー辞書	任意のカテゴリで構いません。本チュートリアルでは新規作成しています。本章で作成する辞書項目は全てこのカテゴリに設定してください。
辞書項目ID	tutorial_workflow_title	
辞書項目名	タイトル	画面に表示される項目名にも利用されます。
データ型	TEXT	
変数名	title	画面や処理で利用する変数に利用されます。

i 情報はアプリケーション作成のビルド実行時に登録されます。

基本設定

登録先辞書項目カテゴリ*
この項目を登録する辞書項目カテゴリを選択します。

【チュートリアル】ワークフロー辞書

Q

辞書項目ID*

tutorial_daily_report_app

辞書項目名*
ここで入力した情報は「Database設定」の「論理名」に自動設定されます。

タイトル

データ型*
ここで入力した情報は各データ設定の「データ型」と、「Database設定」の「最大桁数」「小数桁数」に自動設定されます。

TEXT

v

変数名*
ここで入力した情報は各データ設定の「変数名」に自動設定されます。

title

データ

項目を手入力する

キャンセル
設定

「登録先辞書項目カテゴリ」は画面や処理では利用しないため、任意のカテゴリを設定してください。本チュートリアルでは、「【チュートリアル】ワークフロー辞書」というカテゴリを作成しています。本章で作成する辞書項目はすべてこのカテゴリに設定してください。

「辞書項目ID」も画面や処理では利用しませんが「IM-Repository」内で一意に設定します。重複に注意してください。

次に「データ」の項目を設定します。「データ」の項目では、エンティティの紐づけの設定を行います。本チュートリアルでは、Database設定のみを行います。

「物理名」に「title」、「最大桁数」に「1000」と設定してください。

辞書項目新規作成

title

データ

項目を手入力する

Database設定	データ型 文字列	物理名* title	論理名 タイトル
		最大桁数* 1000	小数桁数 (空)
Java設定	データ型 String	変数名 title	
JavaScript設定	データ型 String	変数名 title	
IM-LogicDesigner設定	データ型 string	変数名 title	

制約

桁数
 パンク数

設定がすべて完了したら「設定」ボタンをクリックしてください。エンティティ情報に設定した辞書項目が1行追加されます。「タイトル」の設定の最後に、追加したデータの設定を行きましょう。「表示形式」を「テキストボックス」と設定してください。「必須」にチェックを入れてください。

アプリケーション作成
シンプルなワークフロー - 新規作成

- アプリケーション情報*
- 扱うデータの定義*
- 明細データの利用指定
- ワークフロー情報の入力*
- その他情報の指定

① 作成するアプリケーションで扱うデータを IM-Repository から指定します。このアプリケーションでは新規にエンティティを作成します。

IM-Repository 資料の設定

簡易に設定する
テーブル情報を入力することで、エンティティや辞書項目を自動で作成します。エンティティや辞書項目を登録せずに扱うデータの定義を作成できます。

詳細に設定する
辞書項目を有効に利用してエンティティを作成します。辞書項目の新規作成や、既存の辞書項目の利用ができます。

エンティティID*
半角英数字、_ (アンダースコア)、. (ドット)、@ (アットマーク)、+ (プラス)、! (エクスクラメーション) で入力してください。
tutorial_daily_report_app_entity

エンティティ名*
【チュートリアル_エンティティ】日報アプリケーション

登録先エンティティカテゴリ*
【チュートリアル】エンテ

データベース種別
TENANT

エンティティに紐づくテーブル名*
ヘルプアイコンから入力規則を確認できます。
tutorial_daily_report_app

エンティティ情報*
+ エンティティ項目を追加

辞書項目名	物理名	Databaseデータ型	表示形式	主キー	必須	削除
ユーザデータID	user_data_id	文字列		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
タイトル	title	文字列	テキストボックス	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

アプリケーション作成

これで、「タイトル」の設定は完了です。同様の手順で、以下の表に合わせてデータを作成しましょう。なお、本チュートリアルでは辞書項目に設定する「変数名」とデータの「物理名」には同様の値を設定しています。

辞書項目ID	辞書項目名	データ型	変数名 / 物理名	最大桁数	表示形式	主キー	必須
--------	-------	------	-----------	------	------	-----	----

辞書項目ID	辞書項目名	データ型	変数名 / 物理名	最大桁数	表示形式	主キー	必須
tutorial_workflow_title	タイトル	TEXT	title	1000	テキストボックス		<input type="checkbox"/>
tutorial_workflow_daily_report	日報	TEXT	daily_report	20000	複数行テキストボックス		<input type="checkbox"/>
tutorial_workflow_submission_date	提出日	DATE	submission_date		日付		
tutorial_workflow_applicant	申請者	TEXT	applicant	1000	テキストボックス		
tutorial_workflow_approval_comment	承認コメント	TEXT	approval_comment	20000	複数行テキストボックス		
tutorial_workflow_approver	承認者	TEXT	approver	1000	テキストボックス		



ワークフロー情報の入力

次に、アプリケーションで利用するデータの定義を設定します。
「新規にルートを作成する」を選択してください。

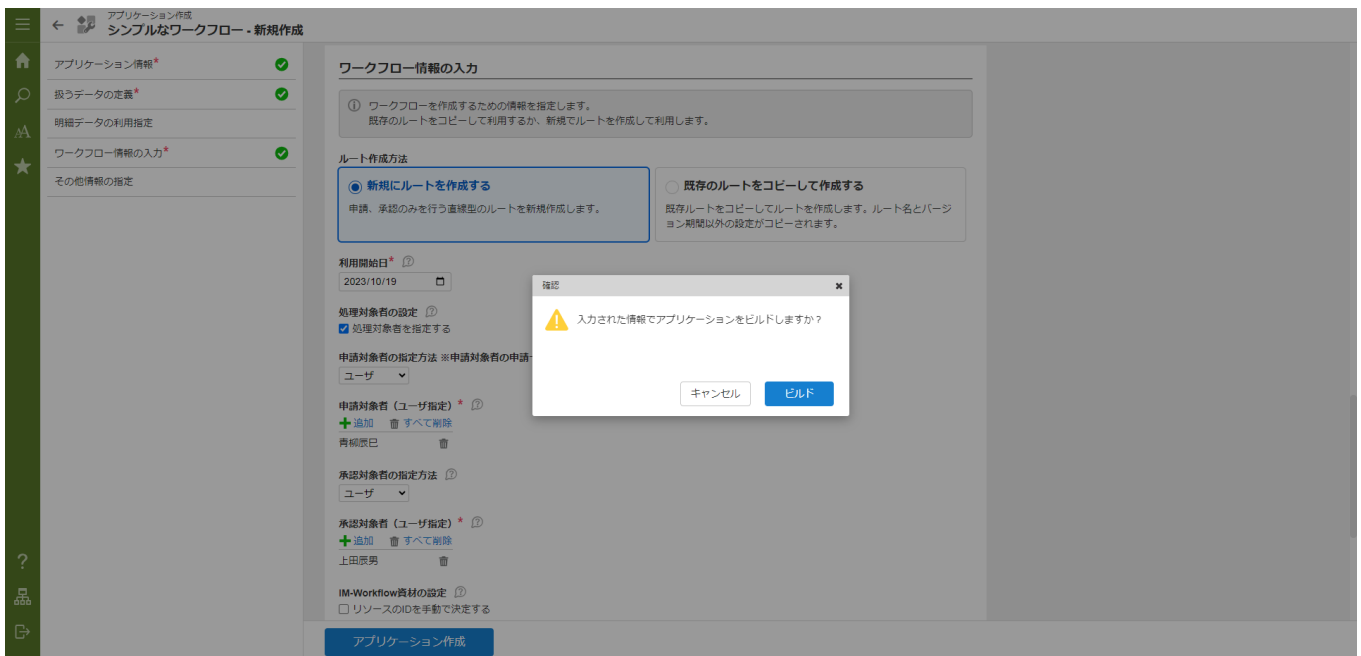
まず、「利用開始日」に現在日を入力します。

次に処理対象者を指定します。

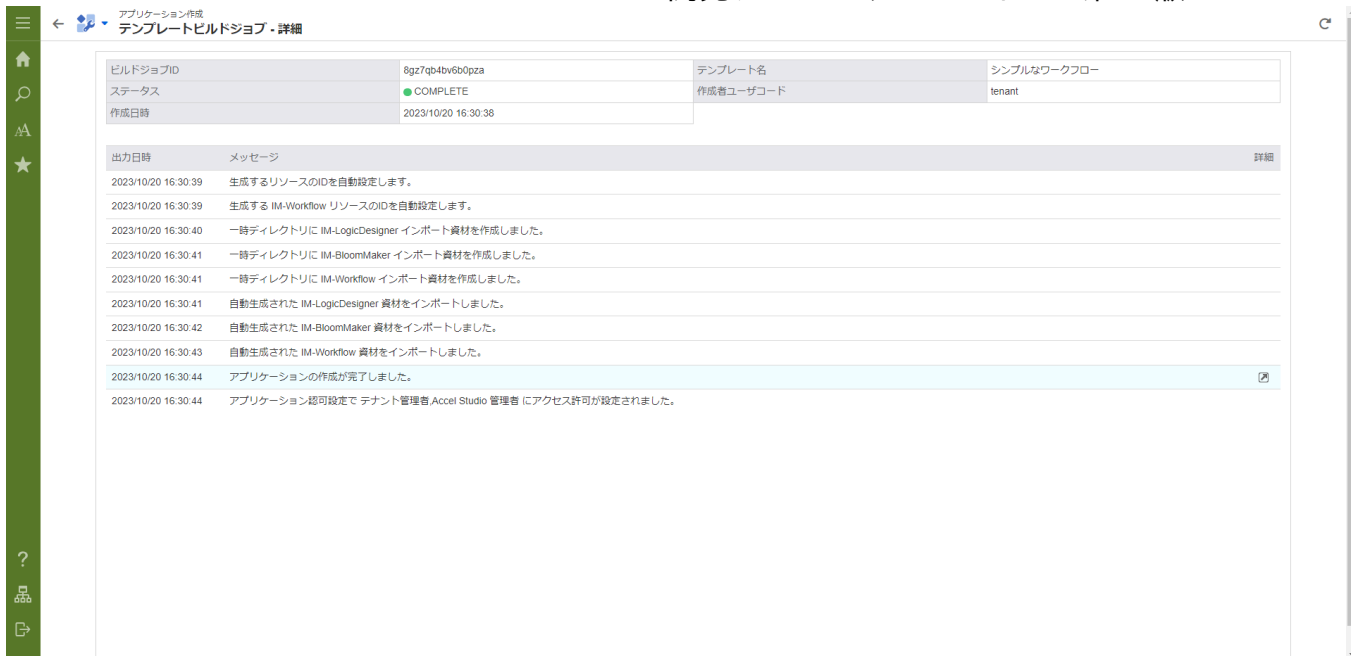
「処理対象者の設定」の「処理対象者を指定する」にチェックを入れます。
「申請対象者の指定方法」のプルダウンから「ユーザ」を選択します。「+追加」をクリックするとユーザ検索ダイアログが表示されるので、青柳辰巳ユーザを選択し、「決定」をクリックして追加します。
同じく、「承認対象者の指定方法」のプルダウンから「ユーザ」を選択します。「+追加」をクリックするとユーザ検索ダイアログが表示されるので、上田辰男ユーザを選択し、「決定」をクリックして追加します。



ワークフロー情報の入力まで完了したらテンプレートを利用したアプリケーション新規作成の準備が完了です。「アプリケーション作成」をクリックし、確認ダイアログの「ビルド」をクリックします。



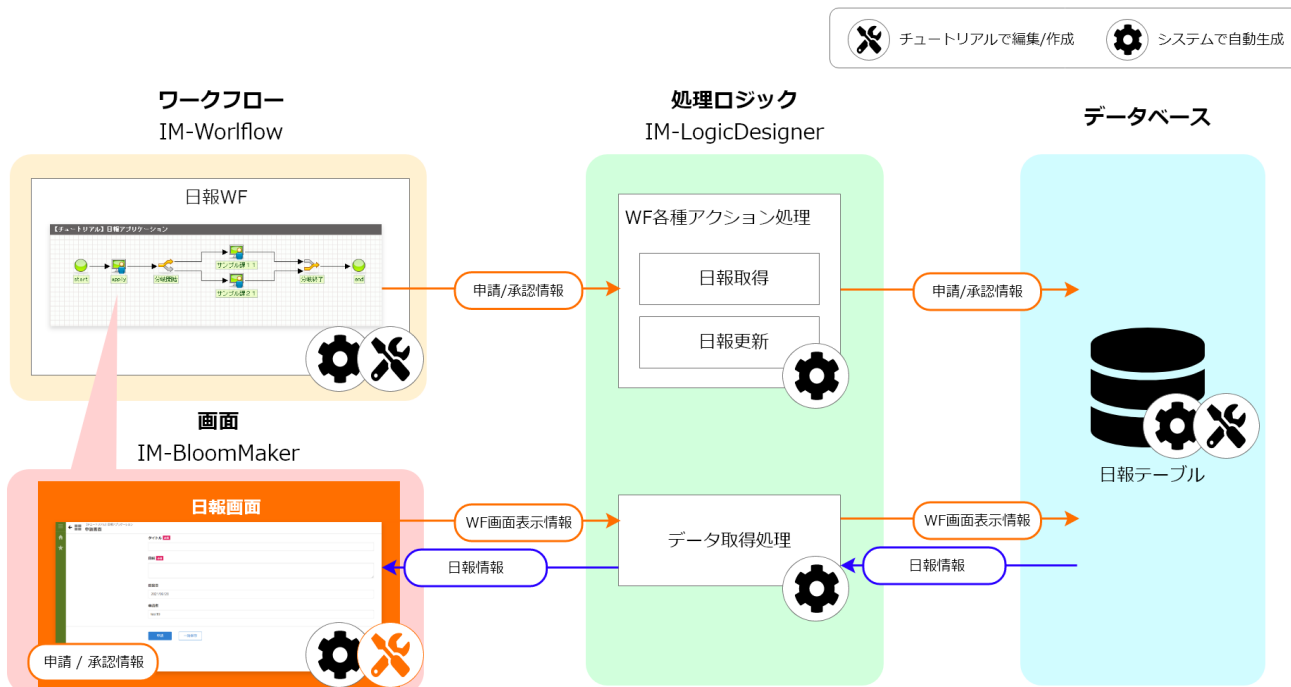
テンプレートからデータベース、画面、処理が自動で生成されます。
画面のステータス項目がCOMPLETEであれば、テンプレートからの自動作成は完了です。



次のページでは、自動作成した処理の修正を行います。

ワークフロー画面の修正

「シンプルなワークフロー」テンプレートから作成した画面はワークフローで利用する1つの画面で作られています。しかし、テンプレートから生成された画面では、申請時に承認コメントや承認者が入力できてしまいます。修正せずとも画面の利用は可能ですが、申請画面として使われている場合や処理画面として使われている場合などを判定し、必要な項目のみを表示するように画面を修正していきます。以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



入力、表示項目の修正

画面の入力、表示項目の修正を行います。

入力項目の修正

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】日報アプリケーション

アプリケーション管理画面のアプリケーション構成情報で「画面」のタブを選択します。

「[チュートリアル] 日報アプリケーションWF画面」を選択します。

IM-BloomMaker コンテンツ編集画面で「デザイン編集」を選択します。

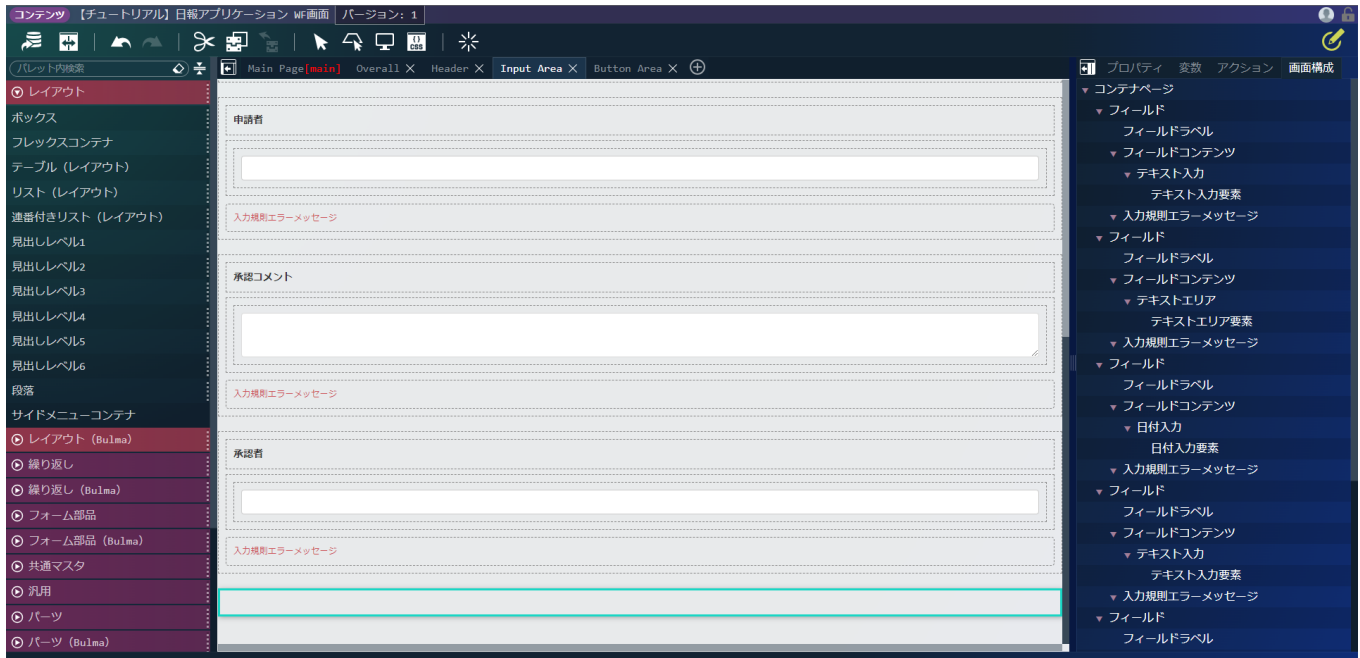
IM-BloomMakerの「デザイン編集画面」が表示されます。

画面上部のタブの中から「Input Area」タブを選択します。

まず、「承認コメント」と「承認者」の項目を申請時には非表示にするよう修正します。

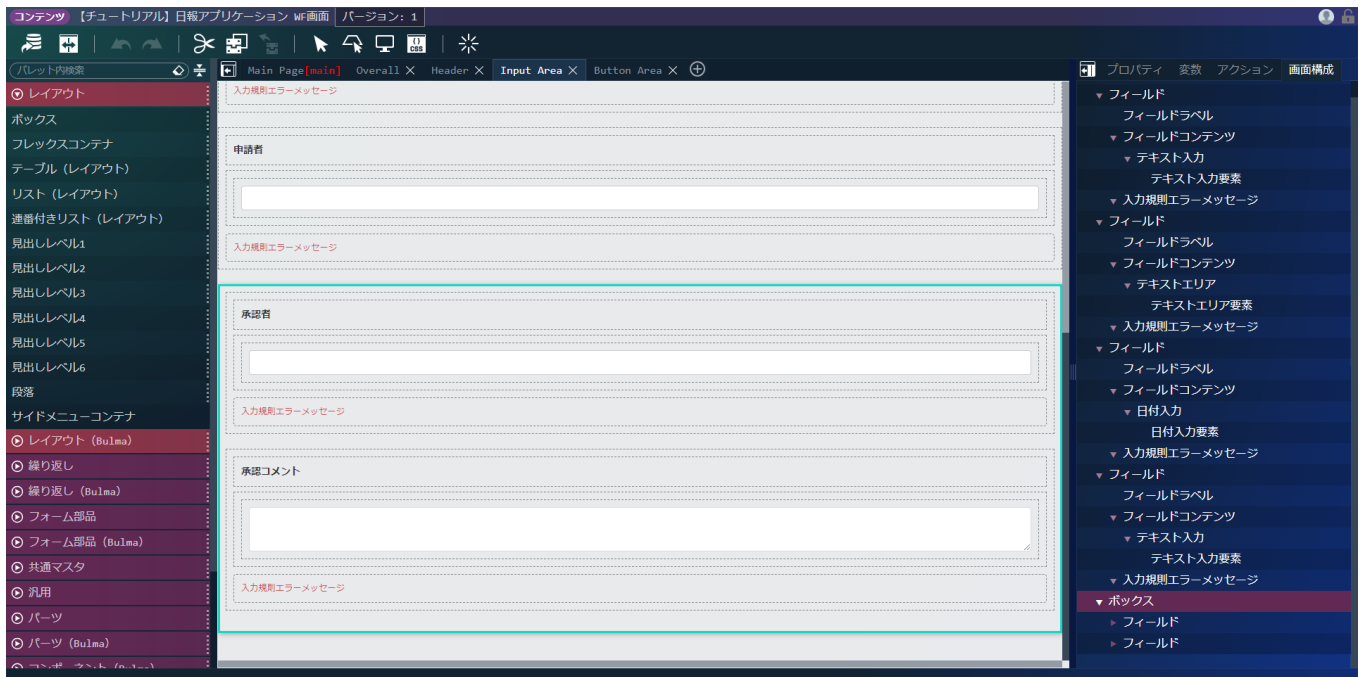
画面左側の「レイアウト」 - 「ボックス」を選択し、一番下に追加します。

画面右側に表示されている「画面構成」タブを選択します。



画面構成の一番下に先ほど追加した「ボックス」が追加されています。


追加したボックスの配下に「承認コメント」、「承認者」のフィールドをドラッグ&ドロップで移動します。

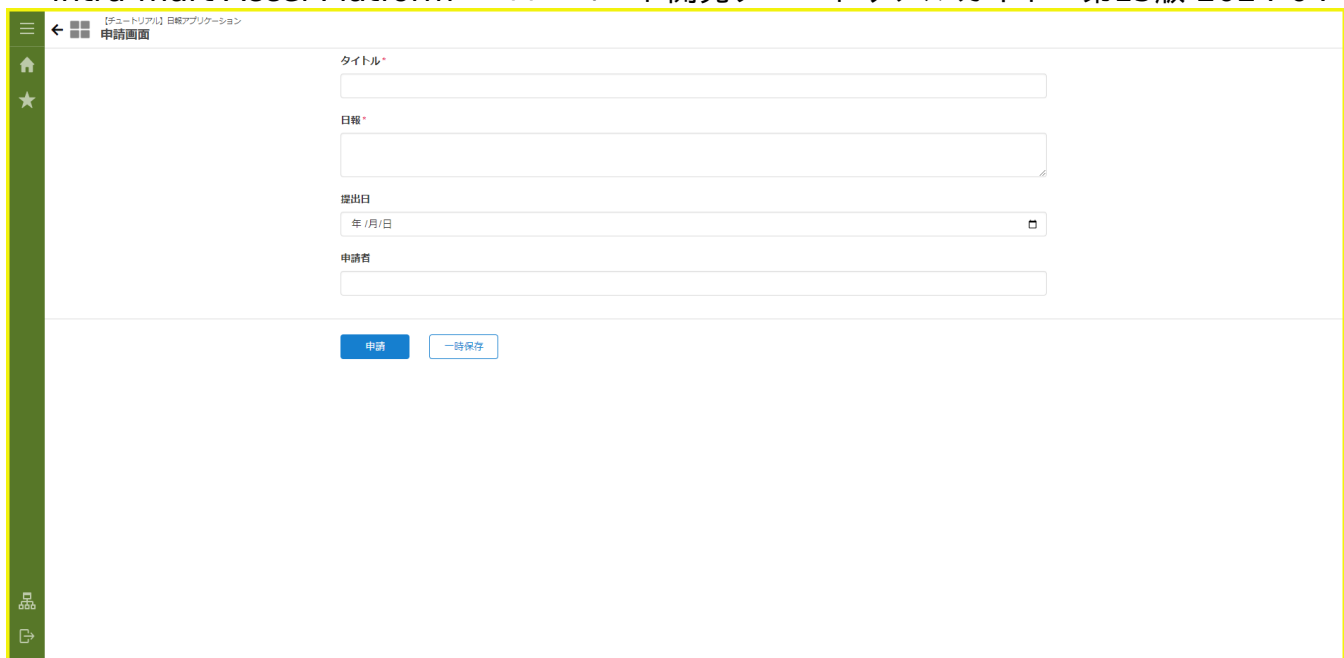


画面構成でボックスを選択した状態で「プロパティ」タブを開きます。

「共通」 - 「表示/非表示」のラジオボタンで変数値を選択し、入力ボックスに以下の値を入力します。

「= \$input.imwPageType != \$constant.pageType.apply」

 をクリックし、承認コメント、承認者が非表示になっていることを確認します。

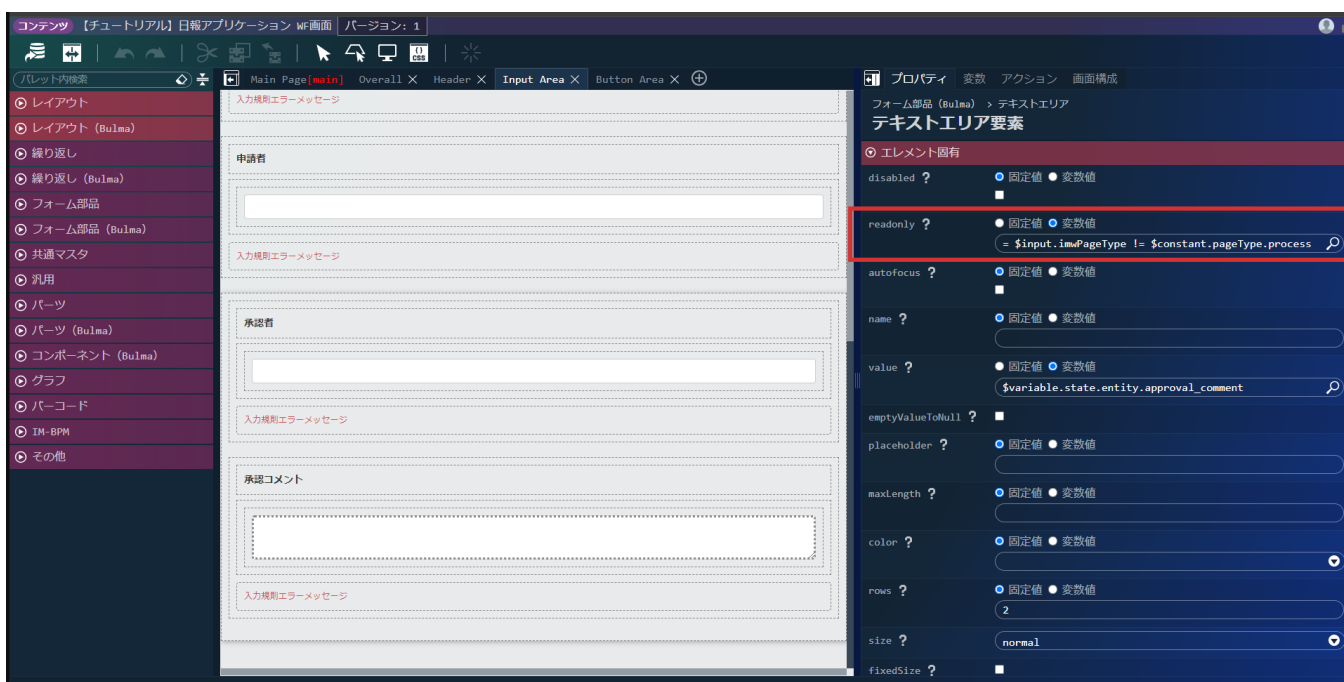


続いて、承認時にのみ「承認者」と「承認コメント」の入力ができるよう修正します。

承認コメントのテキストエリア要素を選択します。

「エレメント固有」 - 「readonly」で変数値を選択し、以下の値を入力します。

`「 = $input.imwPageType != $constant.pageType.process 」`

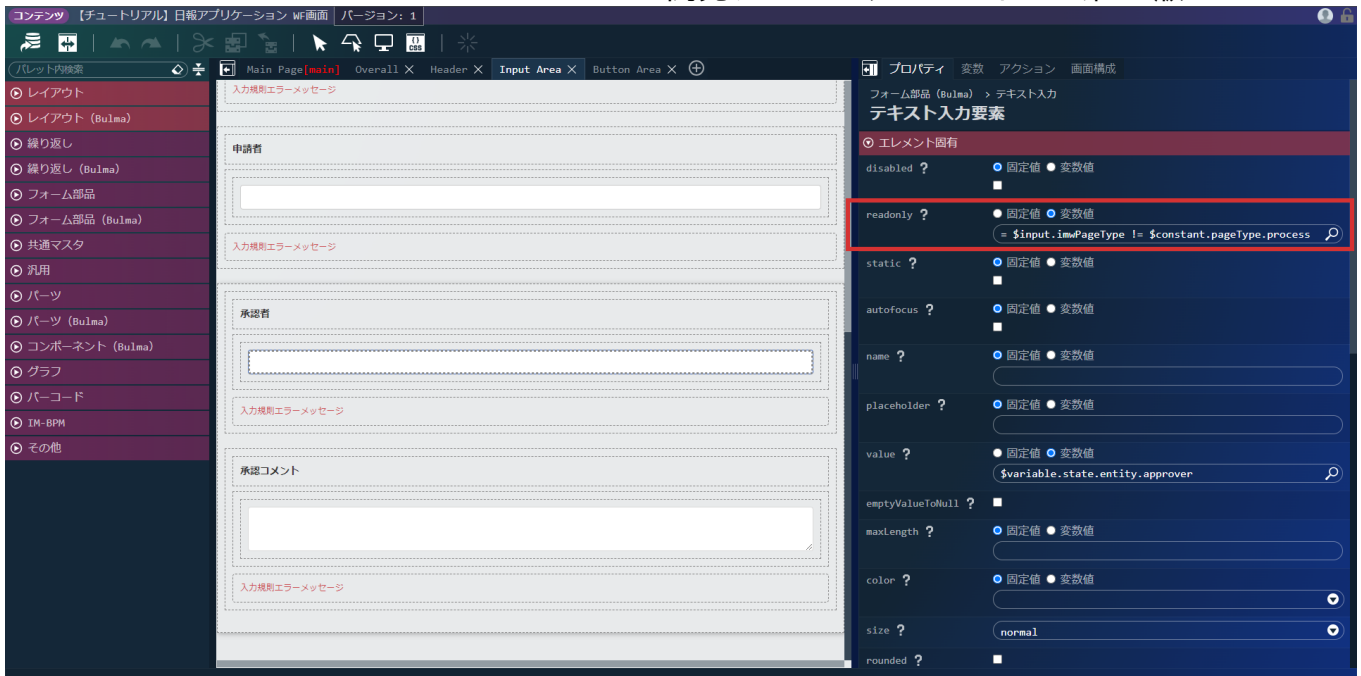


承認者のテキスト入力要素を選択します。

「エレメント固有」 - 「static」を変数値から固定値に変更します。

「エレメント固有」 - 「readonly」で変数値を選択し、以下の値を入力します。

`「 = $input.imwPageType != $constant.pageType.process 」`



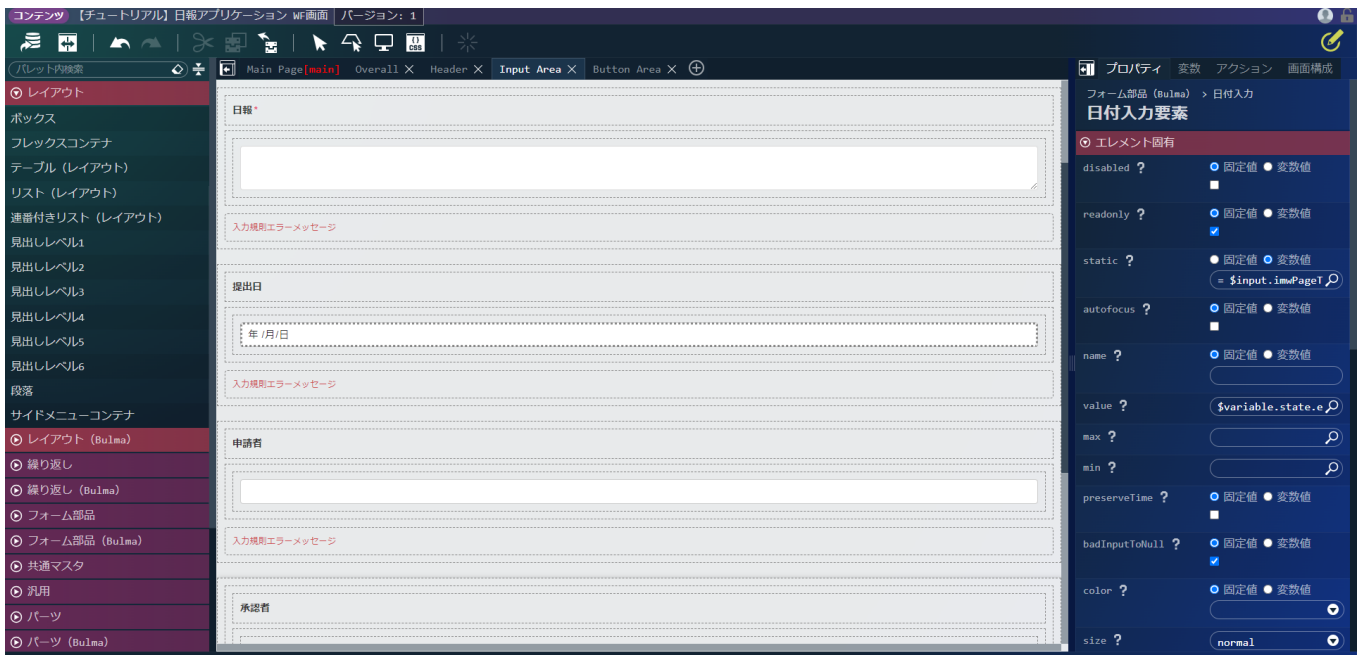
手動入力を制限する修正

次に「提出日」、「申請者」の項目の手動入力を制限します。

提出日の日付入力要素を選択し、「プロパティ」タブを開きます。

「エレメント固有」 - 「readonly」のラジオボタンで「固定値」を選択します。

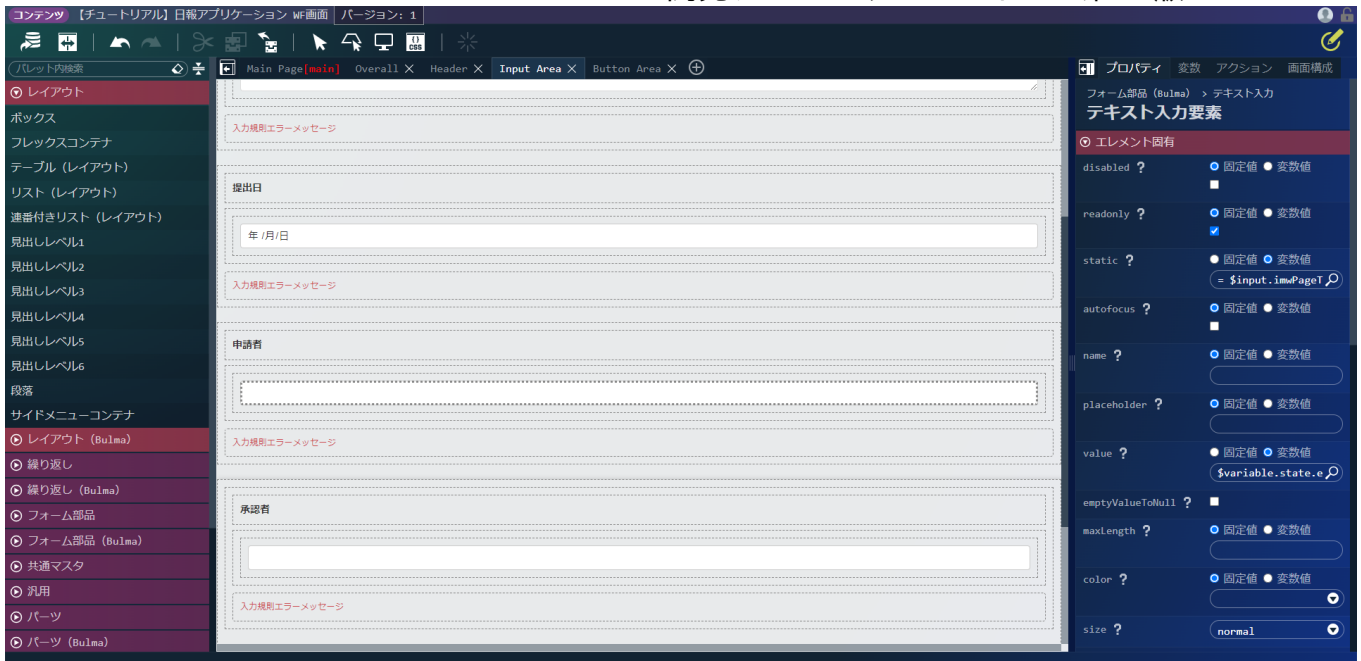
ラジオボタンの下に表示されたチェックボックスにチェックを入れます。



申請者のテキスト入力要素を選択し、「プロパティ」タブを開きます。

「エレメント固有」 - 「readonly」のラジオボタンで固定値を選択します。

ラジオボタンの下に表示されたチェックボックスにチェックを入れます。



をクリックし、提出日、申請者が入力を受け付けなくなっていることを確認します。

アクションの修正

画面のアクションを修正します。

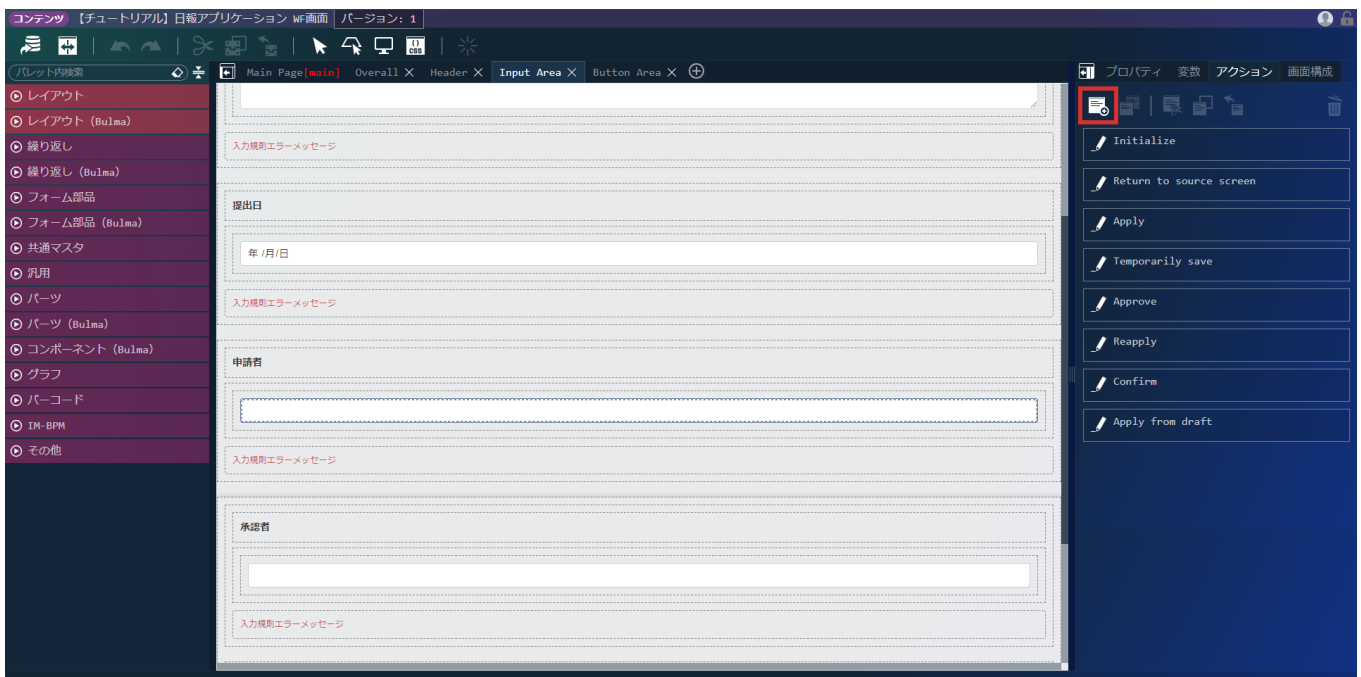
提出日の自動入力アクションを追加

提出日を自動で入力するよう修正します。

画面右側に表示されている「アクション」タブを選択します。



をクリックします。



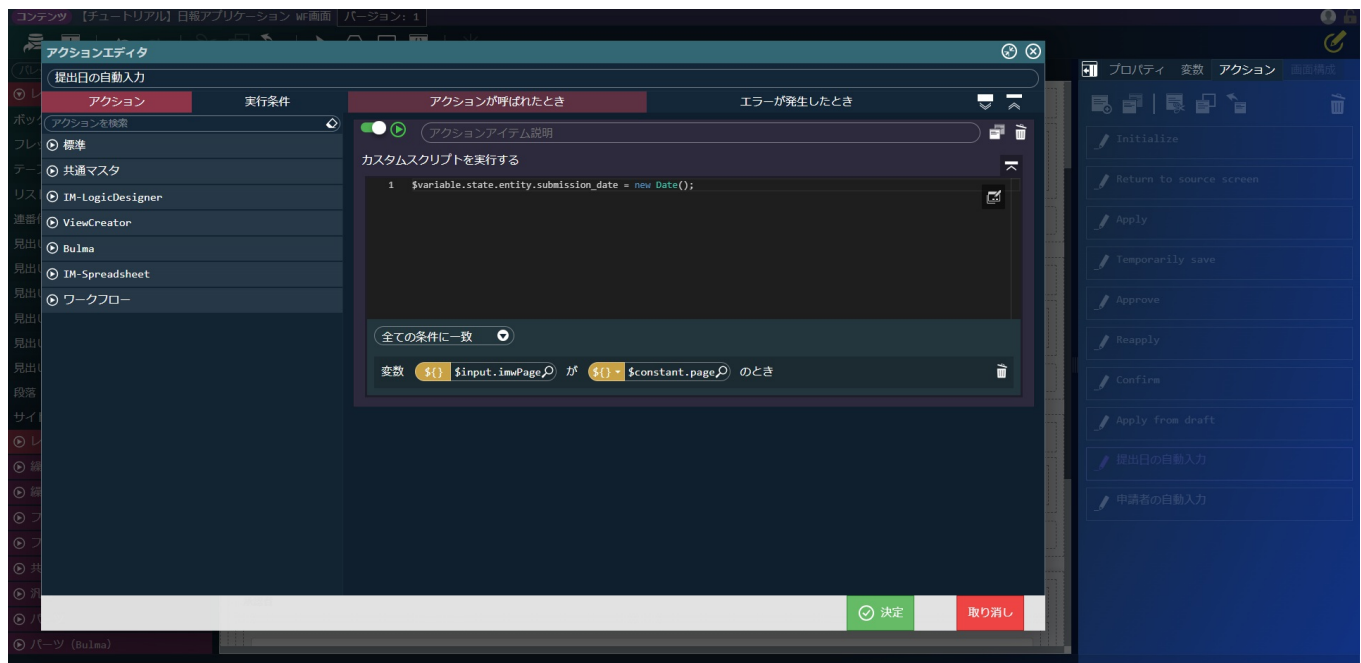
アクション名に「提出日の自動入力」と入力します。

「標準」 - 「カスタムスクリプトを実行する」を右側のスペースヘドラッグ&ドロップします。

以下の値を入力します。

```
$variable.state.entity.submission_date = new Date();
```

次に、申請時のみ動作するよう実行条件を追加します。
 アクションエディタ画面左上にある「実行条件」タブを開きます。
 「標準」 - 「変数〇が〇のとき」を先ほど追加したアクションに重ねるようにドラッグ&ドロップします。
 変数の右側にある 🔍 をクリックし、セレクタを表示させます。
 「入力」の「\$input > imwPageType」を選択し、「決定」をクリックします。
 右辺の「abc▼」をクリックし、入力値の種類を固定値から変数値へと変更します。
 表示された入力ボックス右側にある 🔍 をクリックし、セレクタを表示させます。
 「定数」の「\$constant > pageType > apply」を選択します。



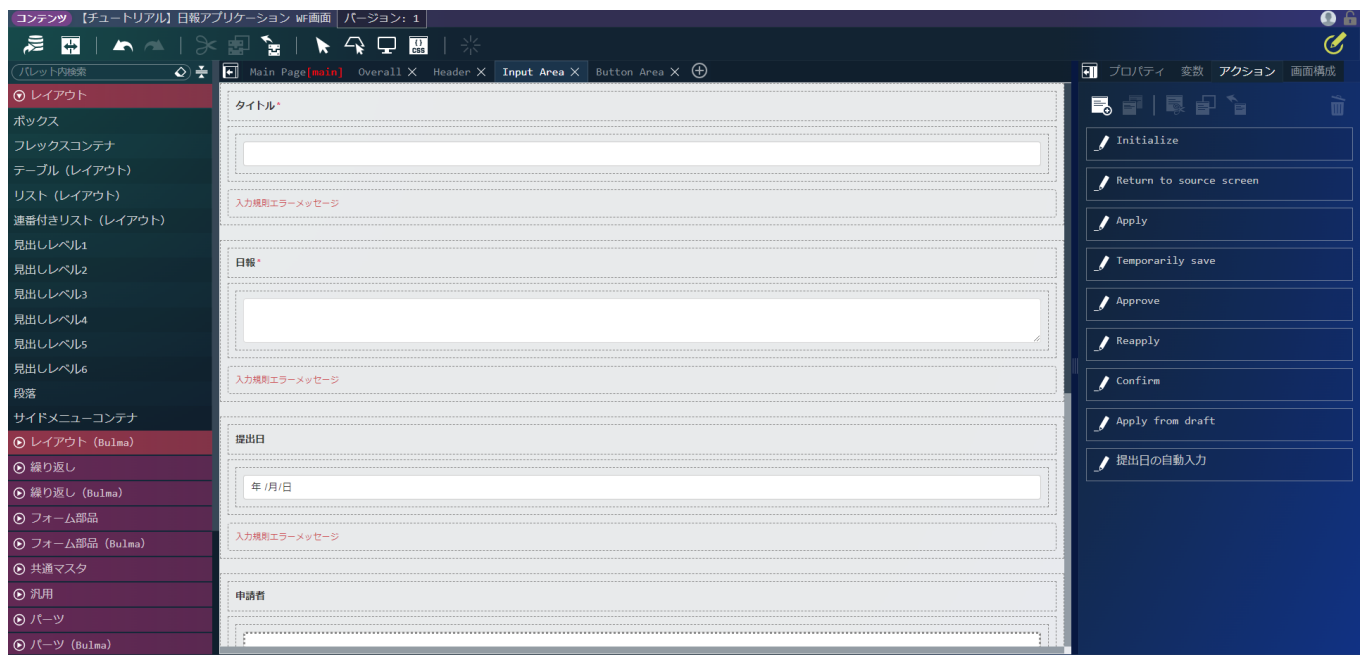
「決定」をクリックし、アクションエディタを閉じて編集を終了してください。

申請者の自動入力アクションを追加

申請者を自動で入力するように修正します。

画面右側に表示されている「アクション」タブを選択します。

☰ をクリックします。




アクション名に「申請者の自動入力」と入力します。

「標準」 - 「変数〇に〇を代入する」を右側のスペースヘドラッグ&ドロップします。

変数の右側にある  をクリックし、変数セレクタを表示させます。

申請者の入力フォームに紐づいている「\$variable > state > entity > applicant」を選択します。

右辺の「abc▼」をクリックし、入力値の種類を固定値から変数値へと変更します。


表示された  をクリックし、セレクタを表示させます。

「環境」 から「\$env > userContext > userProfile > userName」を選択します。

次に、申請時のみ動作するように実行条件を追加します。

アクションエディタ画面左上にある「実行条件」タブを開きます。

「標準」 - 「変数〇が〇のとき」を先ほど追加したアクションに重ねるようにドラッグ&ドロップします。

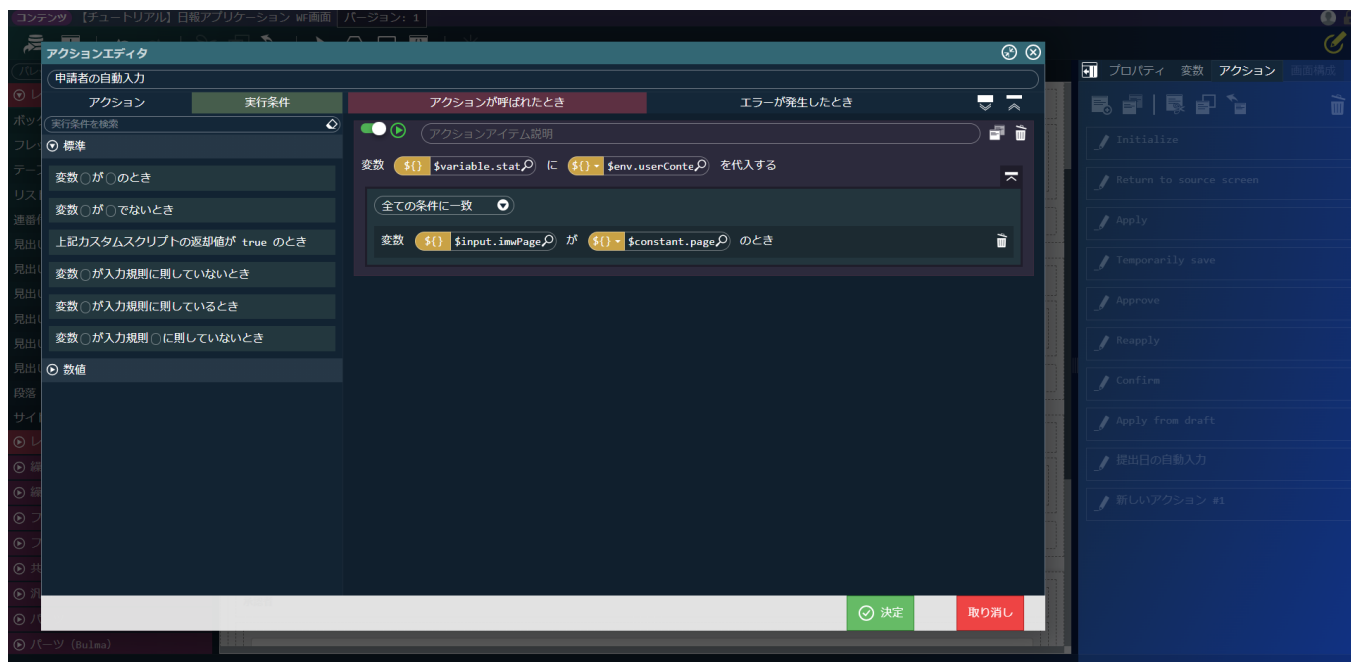
変数の右側にある  をクリックし、セレクタを表示させます。

「入力」の「\$input > imwPageType」を選択します。

右辺の「abc▼」をクリックし、入力値の種類を固定値から変数値へと変更します。

表示された入力ボックス右側にある  をクリックし、セレクタを表示させます。

「定数」の「\$constant > pageType > apply」を選択します。



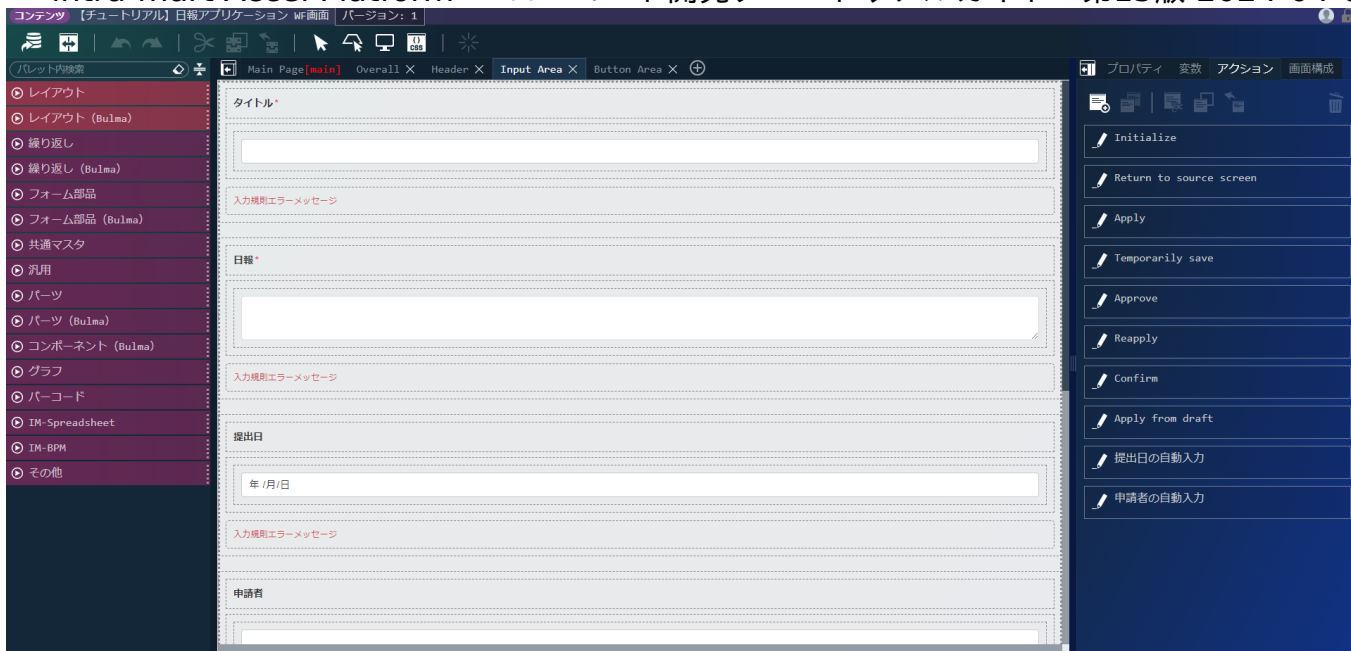
「決定」をクリックし、アクションエディタを閉じて編集を終了してください。

Initializeアクションの修正

作成した2つのアクションを画面表示時に実行されるInitializeアクションに追加します。

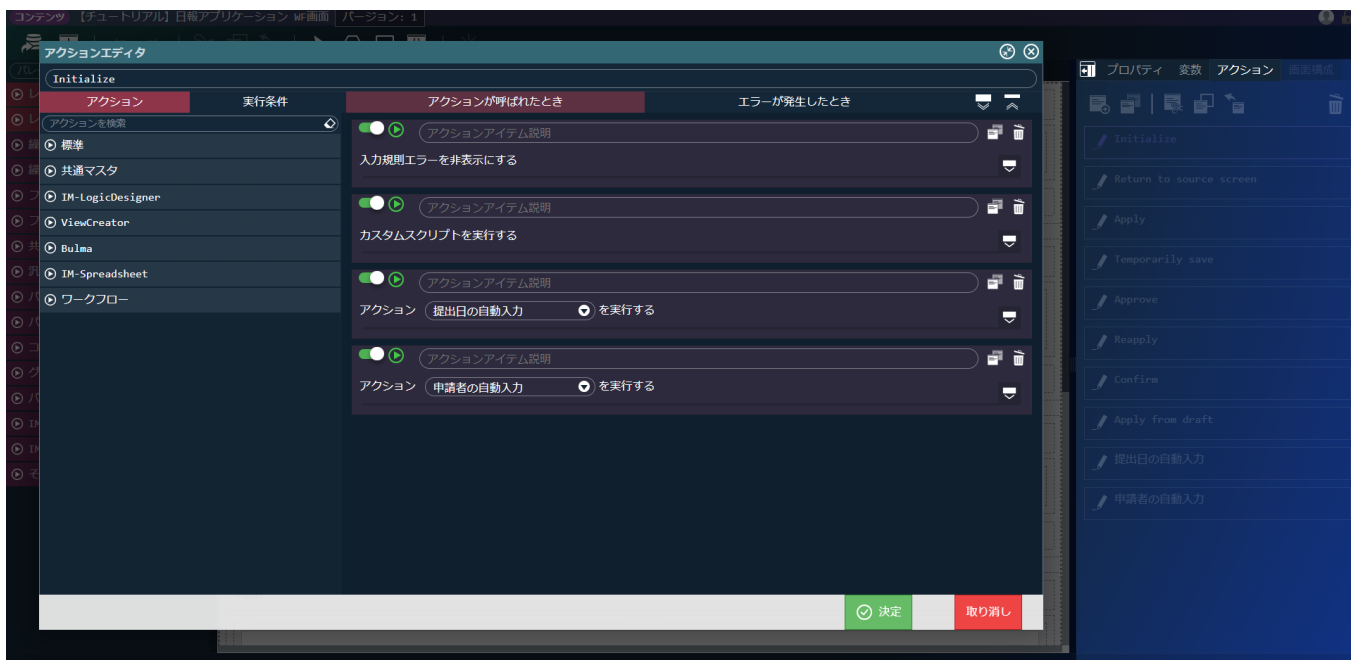
画面右側に表示されている「アクション」タブを選択します。

Initializeアクションの  をクリックします。



「標準」 - 「アクション○を実行する」を右側下部のスペースヘドラッグ&ドロップし、最後尾のアクションとして追加します。
 入力プルダウンの一覧から、先ほど作成した「提出日の自動入力」を選択します。

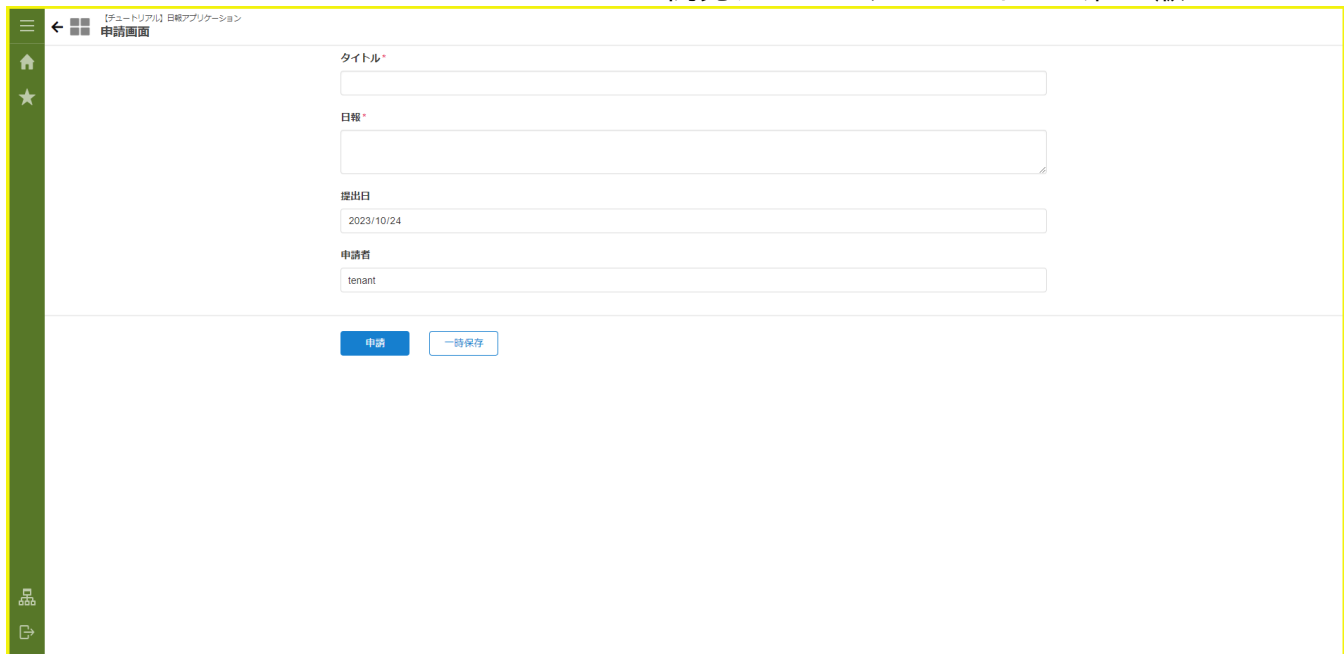
「標準」 - 「アクション○を実行する」を右側下部のスペースヘドラッグ&ドロップし、最後尾のアクションとして追加します。
 入力プルダウンの一覧から、先ほど作成した「申請者の自動入力」を選択します。



「決定」をクリックし、アクションエディタを閉じて編集を終了してください。



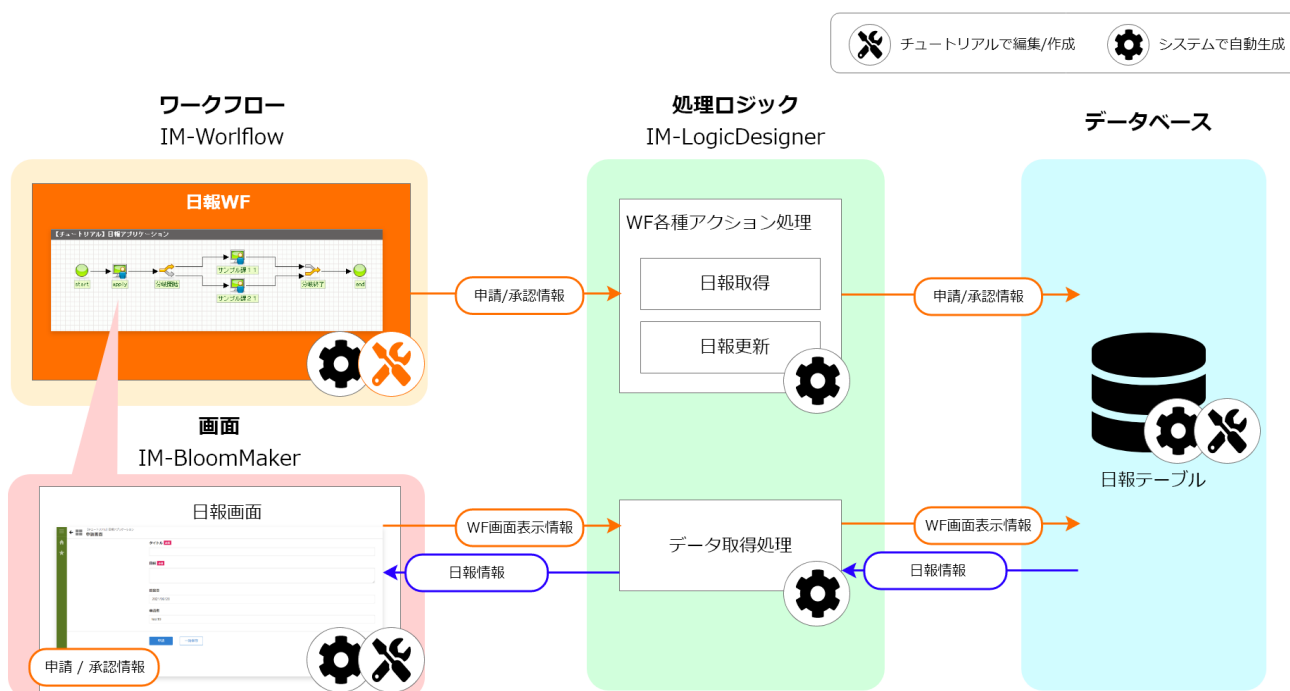
をクリックして、提出日、申請者が自動入力されていることを確認します。



これでワークフロー画面の修正は完了です。

ワークフローの修正

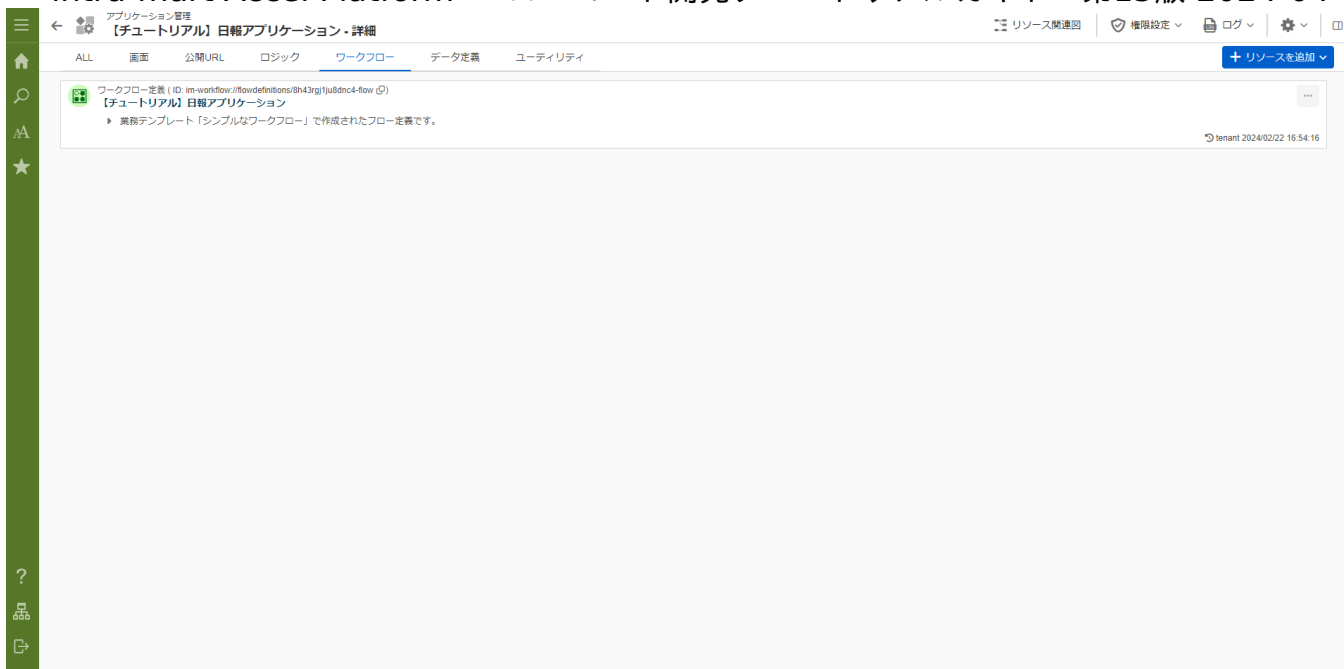
日報の承認者を申請時に選択できるように、ワークフローを修正します。
以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



「サイトマップ」→「Accel Studio」-「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】日報アプリケーション」の順に遷移します。

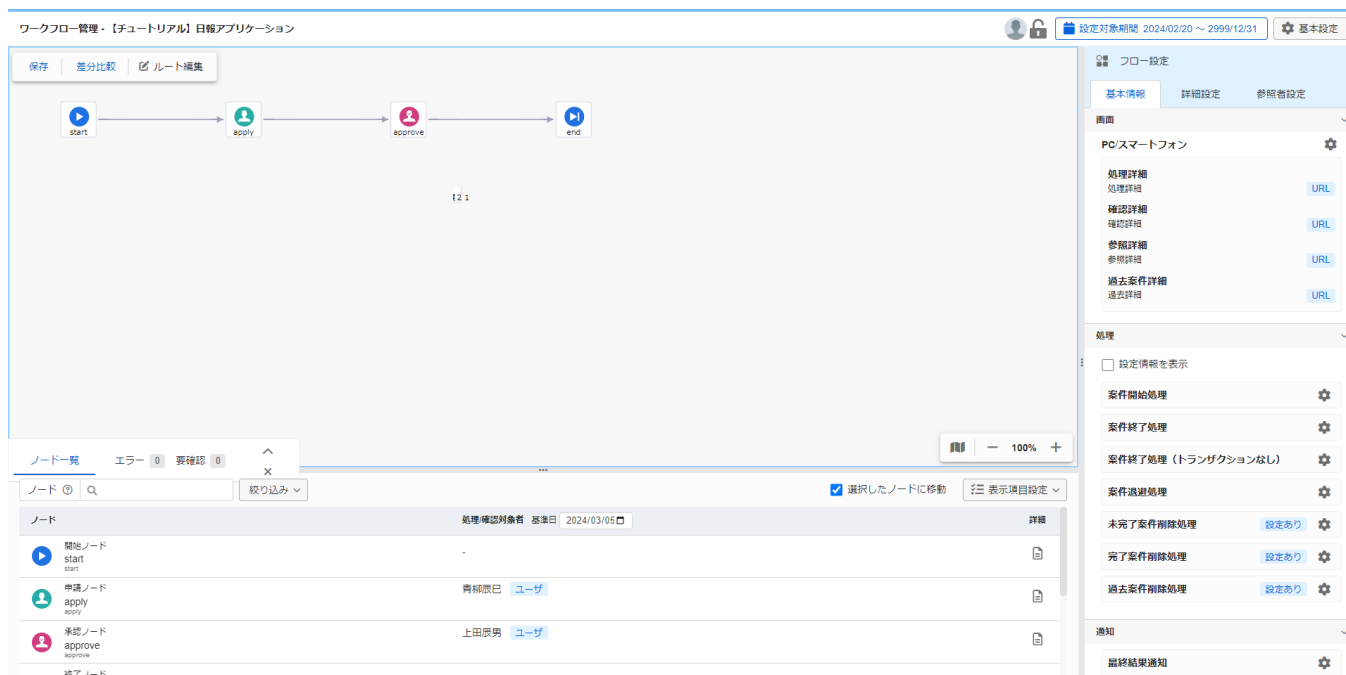
アプリケーション管理画面のアプリケーション詳細で「ワークフロー」のタブを選択します。

「【チュートリアル】日報アプリケーション」を選択します。

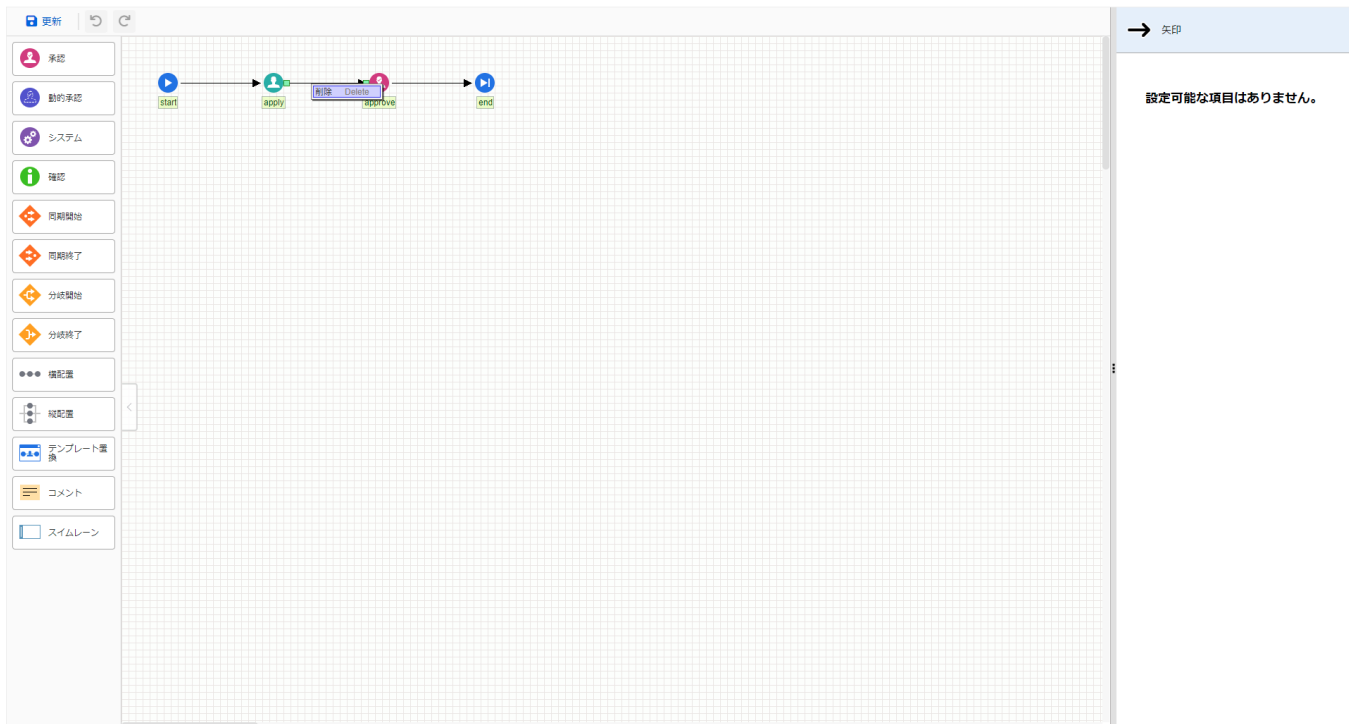


ルートの修正

まずは、ワークフローのルートを修正しましょう。
画面左上の「ルート編集」をクリックします。



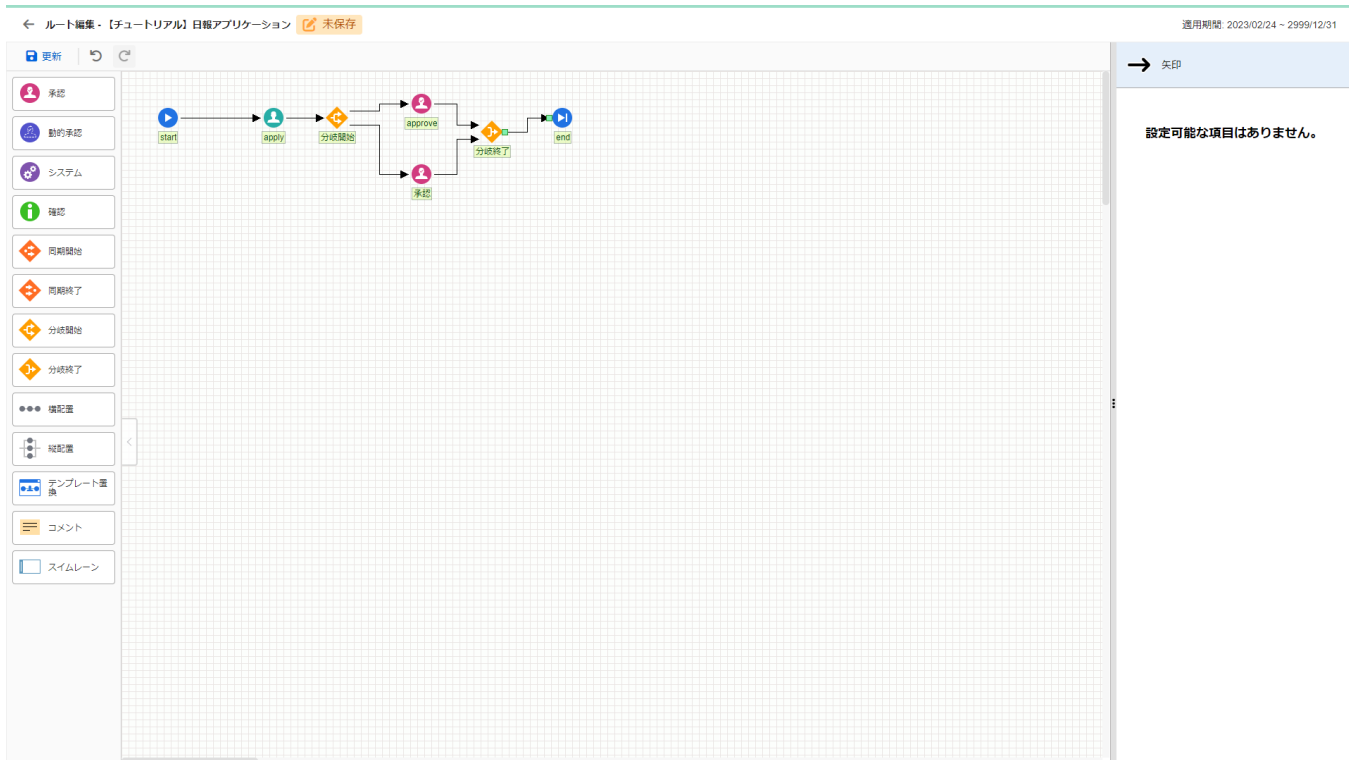
applyからapproveに繋がっている矢印を右クリックし、削除します。
approveからendに繋がっている矢印を右クリックし、削除します。



ルートに分岐開始ノード、分岐終了ノード、承認ノードを追加します。

各ノードの配置順は以下の図を参照してください。

各ノードの配置が完了したらノードのアイコン右上にある「+」マークをクリックし、各ノードを下図のように矢印で繋がめます。



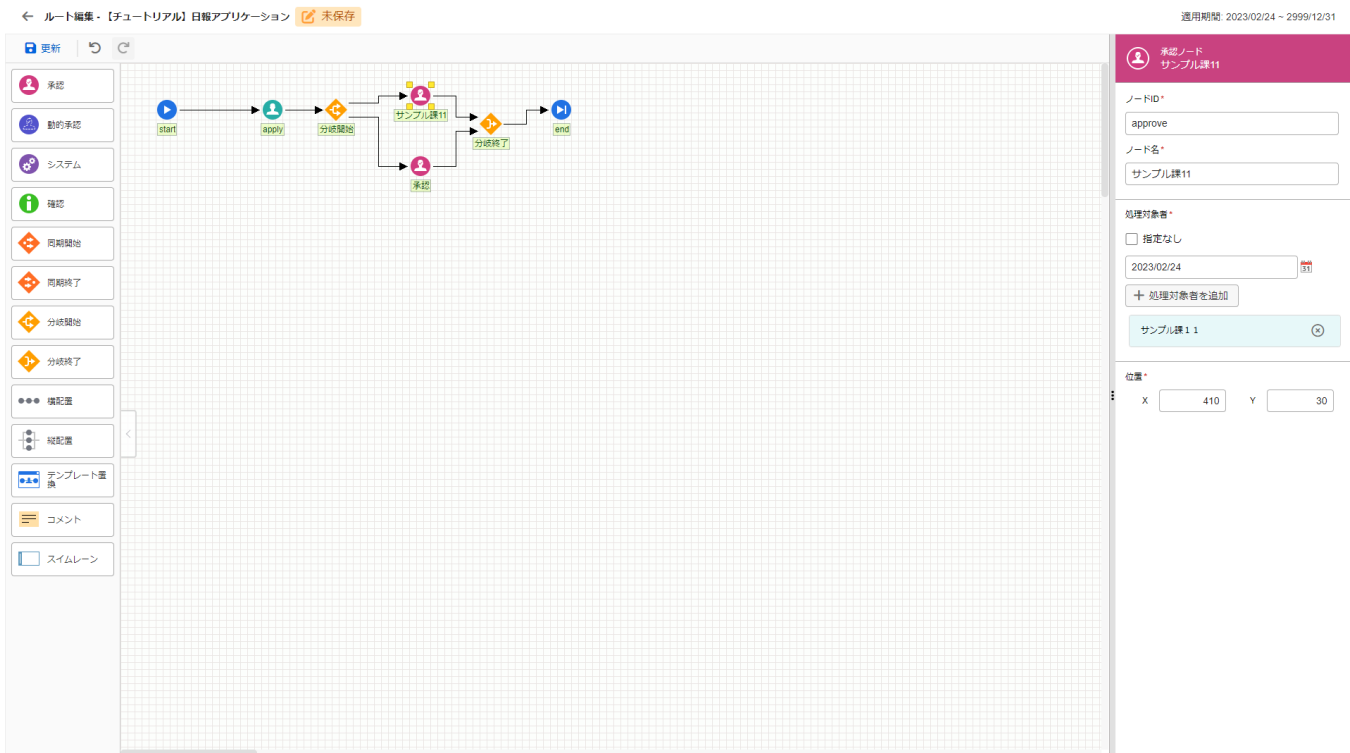
ノード詳細の修正

approveノードをクリックして選択します。

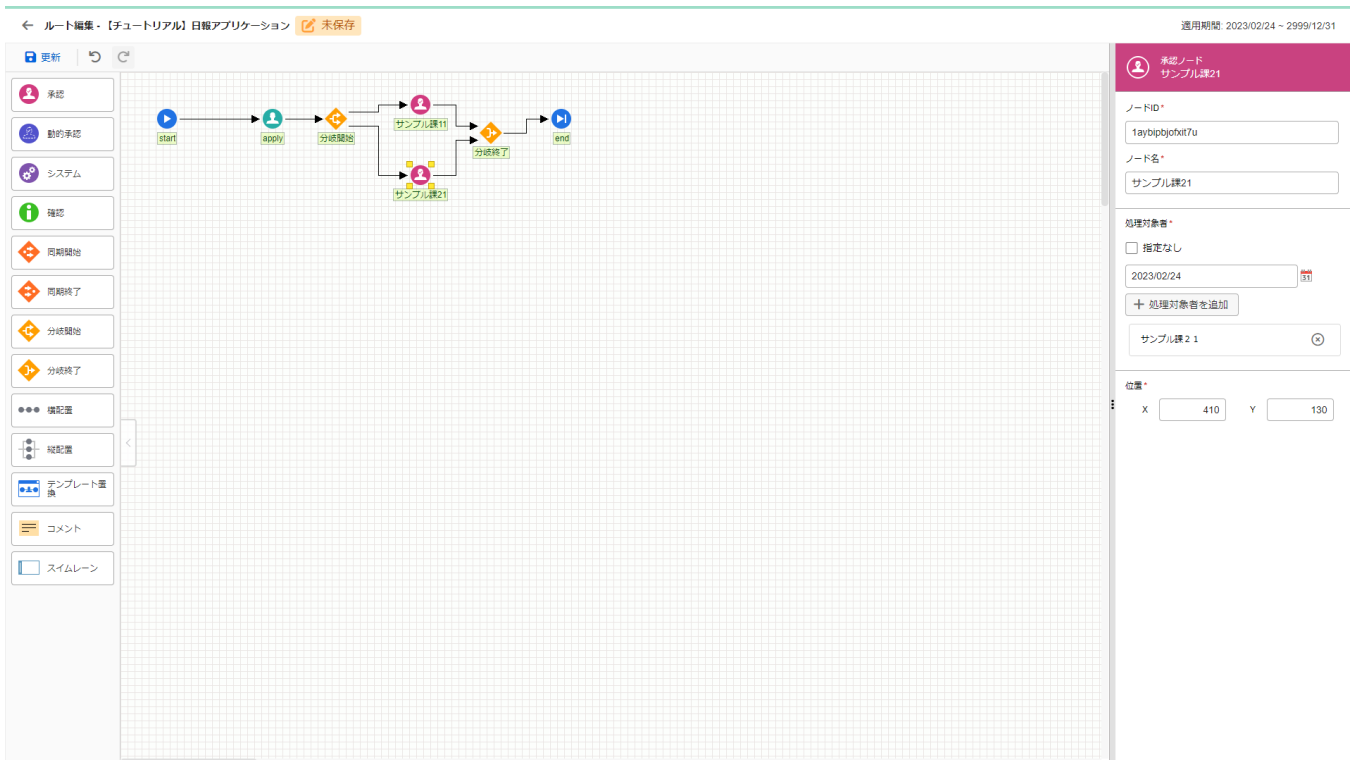
画面右側のノード詳細の「ノード名」に「サンプル課 1 1」を入力します。

「+処理対象者を追加」をクリックし、「組織」をクリックします

「サンプル会社>サンプル部門0 1」配下の「サンプル課 1 1」を選択し「決定」をクリックします。



承認ノードをクリックして選択します。
 「ノード名」に「サンプル課 2 1」を入力します。
 「+処理対象者を追加」をクリックし、「組織」をクリックします。
 「サンプル会社>サンプル部門0 2」配下の「サンプル課 2 1」を選択し「決定」をクリックします。



画面左上の「更新」をクリックし、「← (戻る)」アイコンでワークフロー管理画面に戻ります。

分岐開始方法の設定

次は、分岐開始方法の設定を行います。
 分岐開始方法の設定は、「ノード設定」で行います。
 画面下部のノード一覧から「分岐開始ノード」をクリックして選択してください。

ワークフロー管理・【チュートリアル】日報アプリケーション 未保存

設定対象期間 2024/02/20 ~ 2999/12/31 基本設定

保存 差分比較 ルート編集

分岐開始ノード
分岐開始
cmwfmea1bxdcxv

分岐開始方法

- 申請/処理時に分岐先を選択する
- 分岐処理で分岐先を選択する
- ルール定義で分岐先を選択する
- 設定しない

⚠ 分岐開始方法を設定しない場合は、分岐内の全てのノードに処理を進めます。

ノード一覧 エラー 0 要確認 1

ノード	処理確定対象者	基準日	2024/03/05	詳細
開始ノード start	-	-	-	📄
申請ノード apply	青柳宗巳	ユーザ	-	📄
分岐開始ノード 分岐開始 cmwfmea1bxdcxv	-	-	-	📄
承認ノード	サンプル数 1:1	編集	-	📄

分岐開始方法で「申請/処理時に分岐先を選択する」を選択します。

分岐開始ノード
分岐開始
cmwfmea1bxdcxv

分岐開始方法

- 申請/処理時に分岐先を選択する
- 分岐処理で分岐先を選択する
- ルール定義で分岐先を選択する
- 設定しない

分岐先選択可能ノード*

+ 分岐先設定可能ノードを追加

このノードの分岐先を設定できるノードを選択してください。

「+分岐先設定可能ノードを追加」からapplyを選択します。
ノード設定ダイアログに追加された「apply」の「分岐先複数指定」のチェックを外します。

分岐開始ノード
分岐開始
cmwfmca1bdcctv

分岐開始方法 ▼

- 申請/処理時に分岐先を選択する
- 分岐処理で分岐先を選択する
- ルール定義で分岐先を選択する
- 設定しない

分岐先選択可能ノード*

+ 分岐先設定可能ノードを追加

ノード名	分岐先複数指定
apply	<input type="checkbox"/> 有効 ✕

画面左上の「保存」から保存してください。これで、ワークフローアプリケーションの作成が完了しました。
「申請対象者」として設定した「青柳辰巳」でログインすれば、ワークフローの「申請一覧」に作成したアプリケーションが表示されます。

アプリケーションの認可設定

作成したアプリケーションは、標準では「テナント管理者」ロールに対してのみ認可が設定されています。
このままでは「テナント管理者」ロールを持ったユーザ以外がアプリケーションを利用できません。
アプリケーションの認可設定を行い、認証済みユーザであれば画面を表示できるように設定します。

権限設定

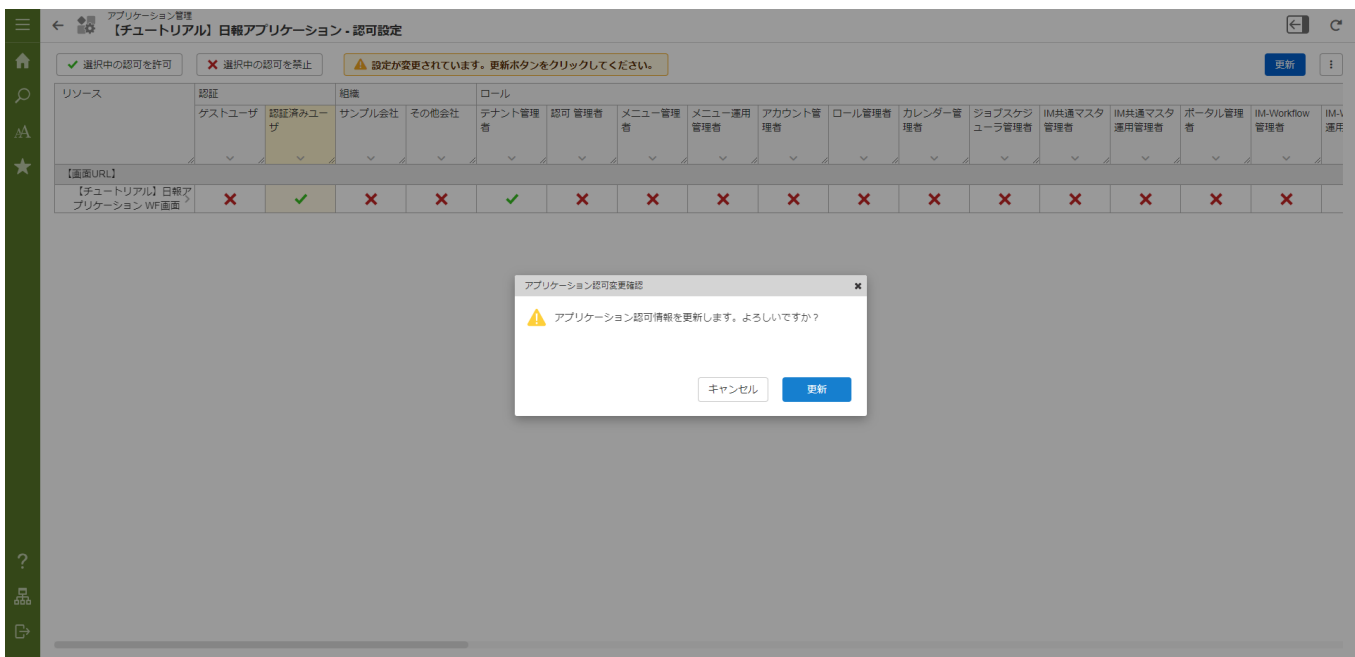
アプリケーションの権限設定はアプリケーション管理から行います。
「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」の順に遷移します。
アプリケーション一覧から、作成した「【チュートリアル】日報アプリケーション」を選択します。
画面右上の「権限設定」プルダウンから認可設定アイコンを選択します。

The screenshot shows the application management interface. The main content area lists various application components like entities, workflows, and actions. On the right side, a dropdown menu is open, showing the 'Permissions' (権限設定) option highlighted with a red box. Below it, there are options for 'Menu Settings' (メニュー設定) and 'Display Considerations' (表示考慮事項).

アプリケーション認可設定画面で、「認証」 - 「認証済みユーザ」の列を選択し、「選択中の認可を許可」をクリックします。



「更新」をクリックし、更新を行います。



これでログイン可能なすべてのユーザが作成したアプリケーションを利用できるようになりました。

ゼロからアプリケーションを作成する

概要

本章では、ローコード開発により簡易的なアンケートアプリケーションを作成します。

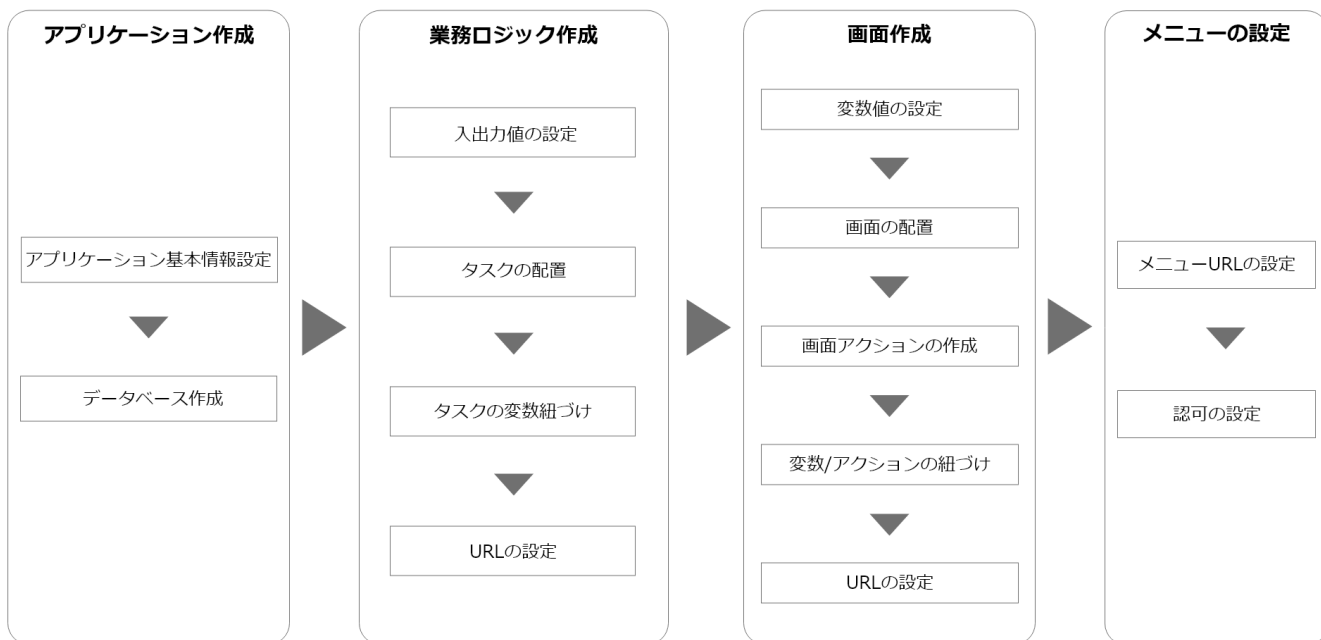
注意

本チュートリアルでは「[環境について](#)」に記載の環境を前提に、機能間の遷移順序を記述しています。詳しくは上記リンクからご確認ください。

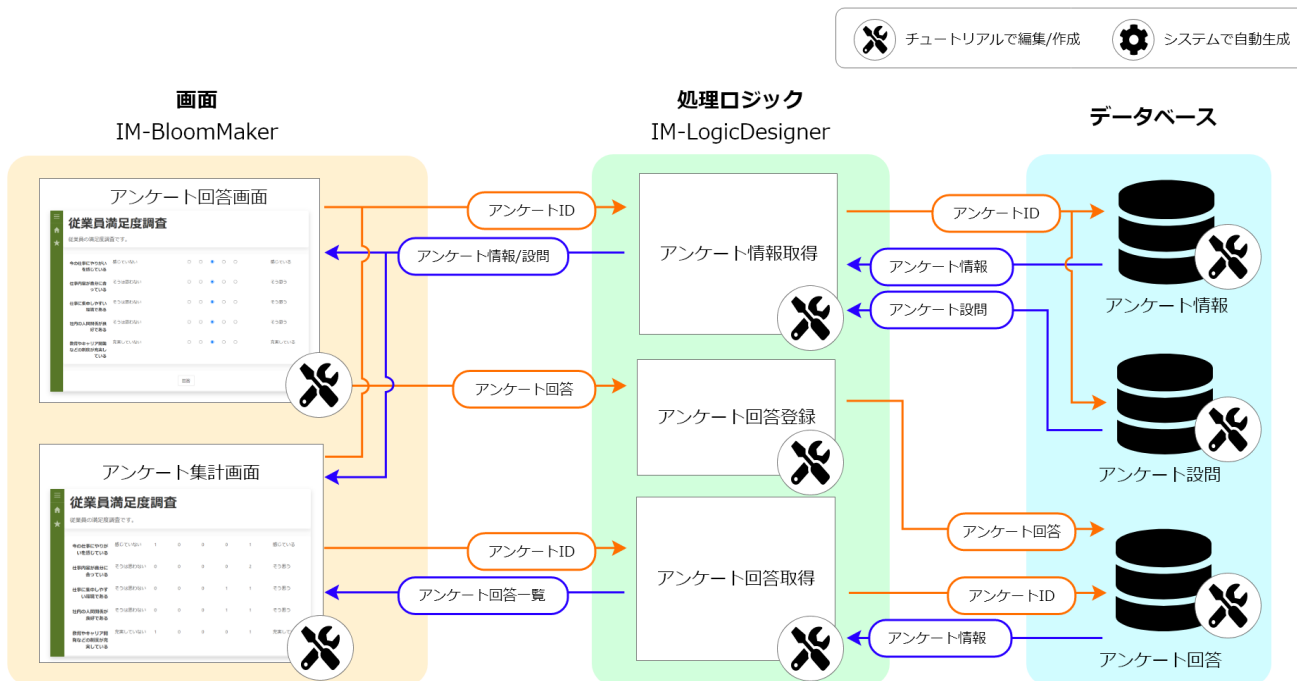
作成の流れ

本チュートリアルは以下の流れでアプリケーションを作成します。

1. アプリケーション作成で空のアプリケーションを作成
2. データベースの作成
3. IM-LogicDesignerで業務ロジックの作成
4. IM-BloomMakerで画面の作成
5. メニューの設定



作成する処理、画面は、以下の画像のようなイメージです。



作成するアンケートアプリケーションは、5スケールの選択制アンケートを利用できます。
 回答、集計画面にアクセスする際にIDを指定することにより、各アンケートを切り替えて表示します。
 設問数、選択肢の両端はデータベースに登録することで設定できます。

アンケート回答画面

アンケートアプリケーションの回答画面は、画面へアクセスする際にIDを指定し、データベースに登録した質問内容呼び出して画面に表示します。
 アンケートは5スケールの選択制で、質問内容、選択肢の両端へ任意の内容を設定できます。

従業員満足度調査
 従業員の満足度調査です。

今の仕事にやりがいを感じている	感じていない	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	感じている
仕事内容が自分に合っている	そうは思わない	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	そう思う
仕事に集中しやすい環境である	そうは思わない	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	そう思う
社内の人間関係が良好である	そうは思わない	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	そう思う
教育やキャリア開発などの制度が充実している	充実していない	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	充実している

集計画面

アンケートアプリケーションの集計画面は、画面へアクセスする際にIDを指定し、データベースに登録した質問、回答内容呼び出して画面に表示します。
 5スケールのうち、どのスケールを何人が選択したか表示します。

従業員満足度調査
 従業員の満足度調査です。

今の仕事にやりがいを感じている	感じていない	1	0	0	0	1	感じている
仕事内容が自分に合っている	そうは思わない	0	0	0	0	2	そう思う
仕事に集中しやすい環境である	そうは思わない	0	0	0	1	1	そう思う
社内の人間関係が良好である	そうは思わない	0	0	0	1	1	そう思う
教育やキャリア開発などの制度が充実している	充実していない	1	0	0	0	1	充実している

アプリケーションの基本情報を設定する

いくつかのローコード開発ツールを組み合わせ、簡易的なアンケートアプリケーションを作成します。
 本章で作成するアンケートアプリケーションはテンプレートを利用せずに作成します。
 作成する処理や画面を統合的に管理するため、アプリケーション作成から「空のアプリケーション」を作成します。

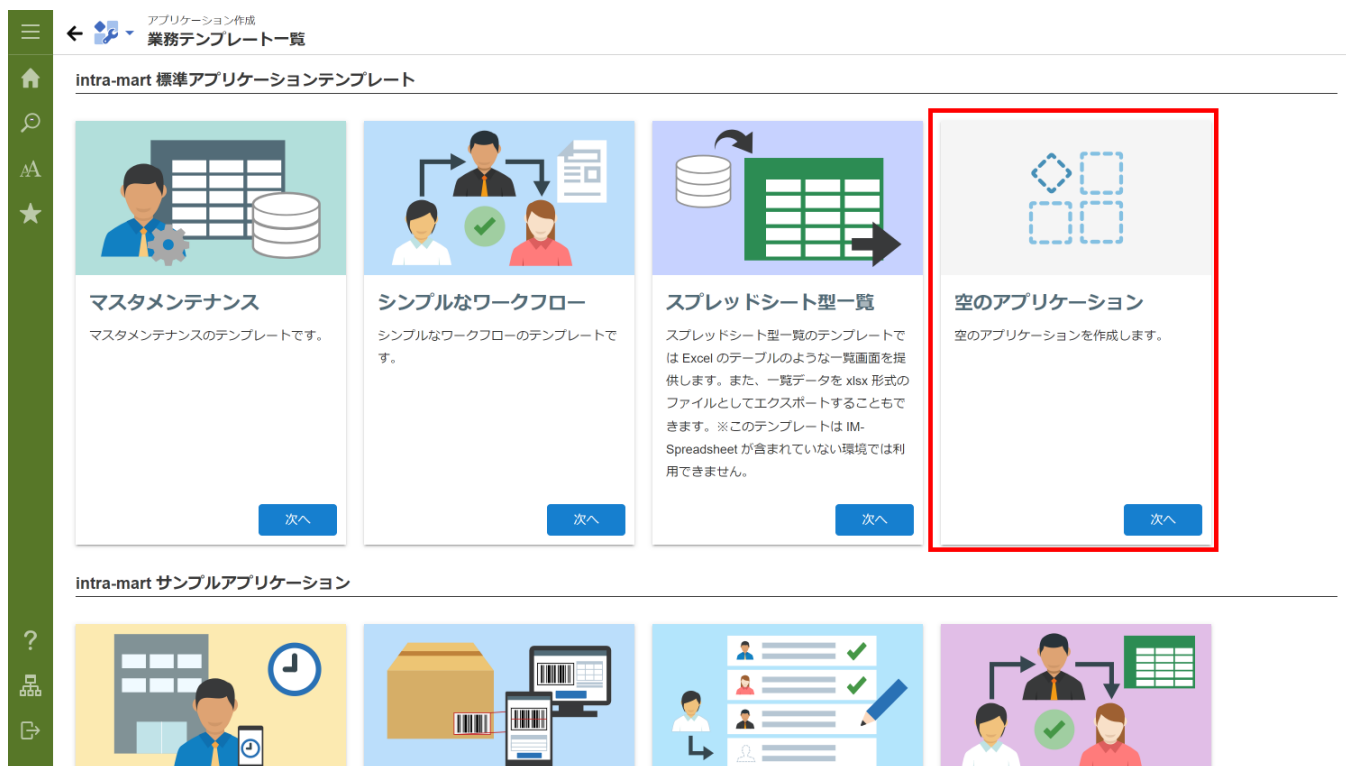
このページの手順について、以下の動画からも確認できます。併せてご利用ください。

注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

アプリケーション作成から管理用アプリケーションを作成

「ベーステンプレート」から「空のアプリケーション」を選択します。
 「空のアプリケーション」を利用するとデータベースのテーブルや処理、画面の自動作成は行われません。



i コラム

アプリケーションを作成する際、入力項目について不明点があれば各項目名の横にある **?** にカーソルを合わせてください。アプリケーション利用時にどのように利用される項目であるか、設定を変えるとどのような変化があるかの説明が表示されます。



アプリケーション情報の設定

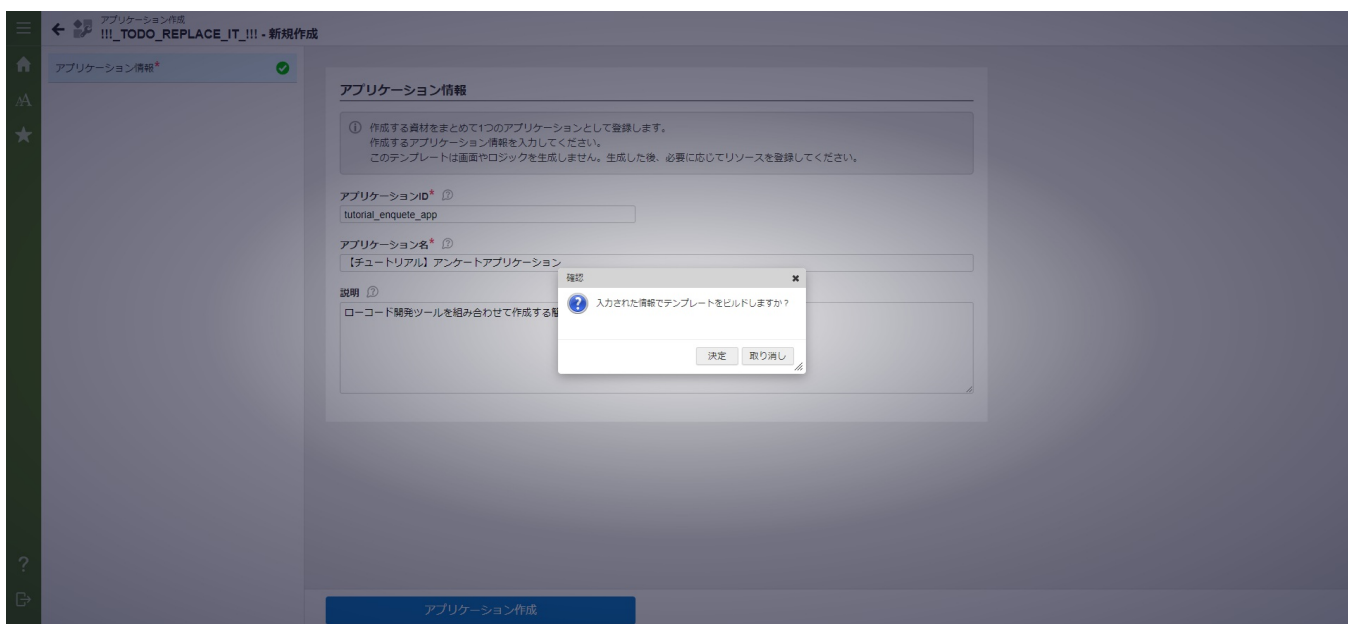
アプリケーション情報を設定します。

アプリケーションID、アプリケーション名、説明に以下の入力値を設定してください。

※アプリケーション情報に関しては、各項目に任意の入力値を設定してもかまいません。本チュートリアルは、下記の通りのアプリケーションID、アプリケーション名を入力した前提で進めます。

ラベル	入力値
アプリケーションID	tutorial_enquete_app
アプリケーション名	【チュートリアル】 アンケートアプリケーション
説明	ローコード開発ツールを組み合わせで作成する簡易的なアンケートアプリケーションです。

アプリケーション情報の設定が完了したらアプリケーション作成を押下し、確認ダイアログの「ビルド」をクリックします。



画面のステータス項目がCOMPLETEであれば、アプリケーション作成は完了です。

← アプリケーション作成
テンプレートビルドジョブ・詳細
☰

ビルドジョブID	8g5b0xt3bhjjav	テンプレート名	空のアプリケーション
ステータス	● COMPLETE	作成者ユーザコード	shinjit
作成日時	Sep 27, 2021 9:08:04 AM		

出力日時	メッセージ	詳細
Sep 27, 2021 9:08:04 AM	アプリケーションの作成が完了しました。アプリケーション詳細画面からリソースを登録してください。	🔗
Sep 27, 2021 9:08:05 AM	アプリケーション認可設定で Accel-Mart Quick アプリケーション管理者 にアクセス許可が設定されました。	

データベースの設定

空のアプリケーション作成を利用すると、データベースのテーブルは自動で作成されません。本チュートリアルでは、下記の3つのテーブルを利用して機能を作成していきます。下記の表の設定に合わせ、データベースのテーブルを作成してください。それぞれの表に合わせたテーブル作成用SQLも用意してあります。

コラム

Accel-Mart Quick環境をご利用の場合は、「Accel-Mart Quick アプリケーション管理者」のロールを持つユーザでログインし、「管理」 > 「アプリケーション管理」 > 「データベース操作」でSQLの実行が可能です。

- tutorial_enquete_info_app**
 アンケート情報テーブル
 アンケートの基本情報を保持します。

物理名	表示名	型	サイズ	主キー	必須	備考
enquete_id	ID	文字列	200	☐	☐	
enquete_title	アンケートのタイトル	文字列	2000		☐	
enquete_overview	アンケートの概要	文字列	2000		☐	
start_date	開始日	日付			☐	
end_date	終了日	日付			☐	

```
CREATE TABLE tutorial_enquete_info_app(
  enquete_id VARCHAR(200) NOT NULL,
  enquete_title VARCHAR(2000) NOT NULL,
  enquete_overview VARCHAR(2000) NOT NULL,
  start_date TIMESTAMP NOT NULL,
  end_date TIMESTAMP NOT NULL,
  PRIMARY KEY (enquete_id)
);
```

- tutorial_enquete_question_app**
 アンケート設問テーブル
 アンケートの設問を保持します。1つの設問につき1つのデータを格納し、5スケールの回答項目の両端の記述内容を指定できます。

物理名	表示名	型	サイズ	主キー	必須	備考
enquete_id	アンケートID	文字列	200	☐	☐	

物理名	表示名	型	サイズ	主キー	必須	備考
question_id	設問ID	文字列	200	□	□	
question_no	設問番号	数値	5		□	
question	設問内容	文字列	2000		□	
left_scale	回答項目の左端	文字列	2000		□	
right_scale	回答項目の右端	文字列	2000		□	

```
CREATE TABLE tutorial_enquete_question_app(
  enquete_id VARCHAR(200) NOT NULL,
  question_id VARCHAR(200) NOT NULL,
  question_no DECIMAL(5) NOT NULL,
  question VARCHAR(2000) NOT NULL,
  left_scale VARCHAR(2000) NOT NULL,
  right_scale VARCHAR(2000) NOT NULL,
  PRIMARY KEY (enquete_id, question_id)
);
```

■ tutorial_enquete_answer_app

アンケート回答テーブル

アンケートの回答を保持します。1つのアンケートの回答を1つのデータとしてまとめて保持します。

物理名	表示名	型	サイズ	主キー	必須	備考
answer_id	アンケートID	文字列	200	□	□	
enquete_id	アンケートID	文字列	200	□	□	
respondent	回答者	文字列	200		□	
answer_date	回答日時	日付			□	
answers	回答	文字列	2000		□	回答内容をカンマ区切りの文字列でひとまとめにして保持する

```
CREATE TABLE tutorial_enquete_answer_app(
  answer_id VARCHAR(200) NOT NULL,
  enquete_id VARCHAR(200) NOT NULL,
  respondent VARCHAR(200) NOT NULL,
  answer_date TIMESTAMP NOT NULL,
  answers VARCHAR(2000) NOT NULL,
  PRIMARY KEY (answer_id, enquete_id)
);
```

コラム

アンケート情報テーブルとアンケート設問テーブルに登録するサンプルデータを用意しています。動作確認やメニューの設定では、サンプルデータを利用して手順を説明します。必要に応じて、以下からダウンロードしたSQLを実行し、ご利用環境へ反映してください。

サンプルデータ

[tutorial_enquete.dml](#)

サンプルデータとして登録されるデータは以下の2種類のアンケートです。

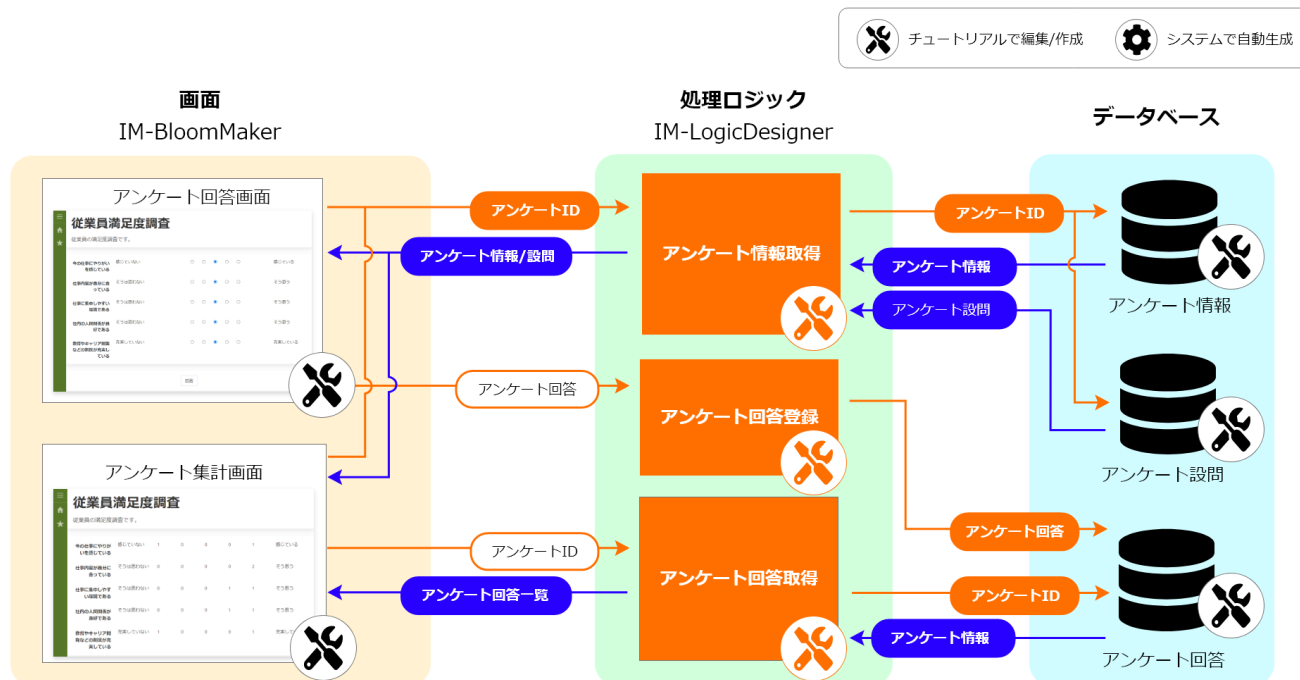
- 従業員満足度調査 (tutorial_es_enquete)
- 好きな映画ジャンル調査 (tutorial_movie_enquete)

アンケートアプリケーションの業務ロジックの作成

アンケートアプリケーションの業務ロジックとして以下の3種類の業務ロジックを作成します。

- アンケート情報取得
アンケートIDをもとに、アンケート情報テーブルとアンケート設問テーブルからデータを取得する
- アンケート回答登録
アンケートの回答をアンケート回答テーブルに登録する
- アンケート回答取得
アンケートIDをもとに、アンケート回答テーブルからアンケートの回答を一覧として取得する

以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。

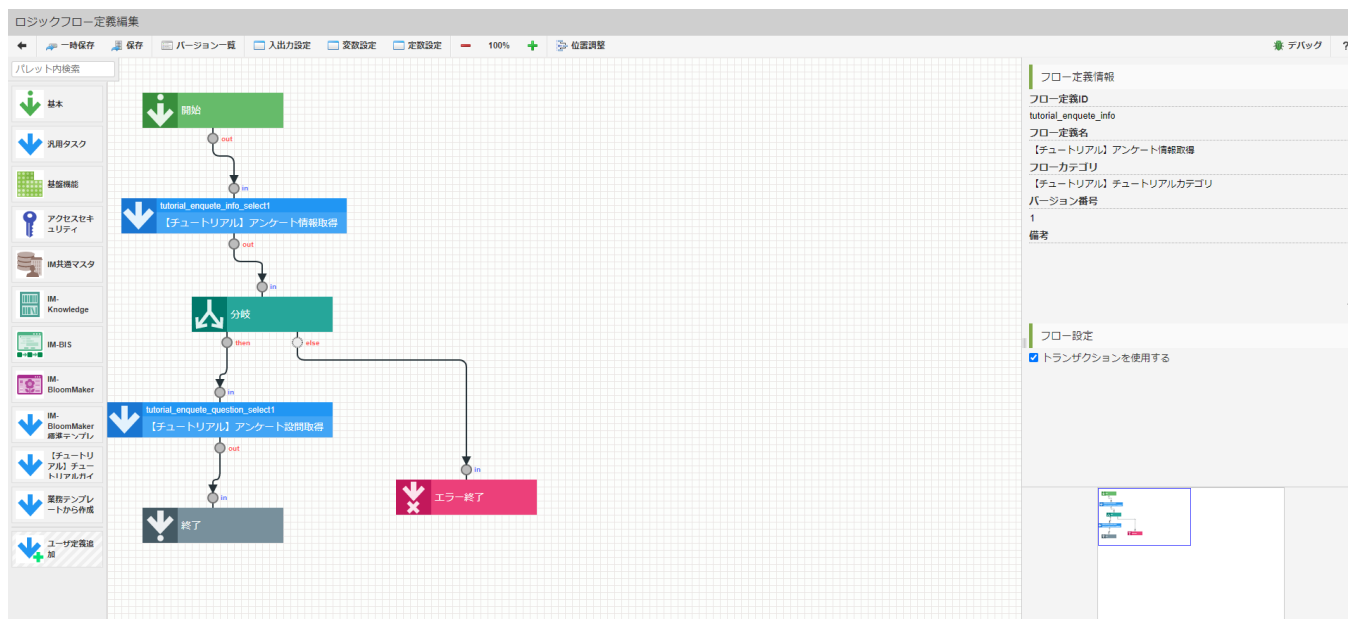


アンケート情報取得処理の作成

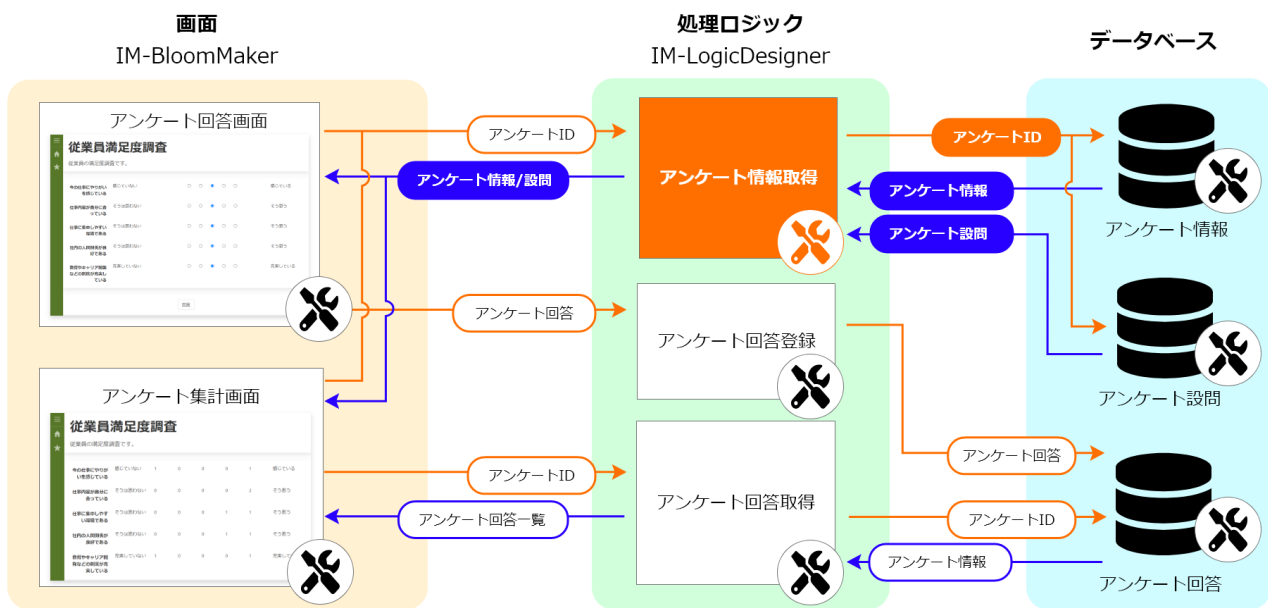
アンケート情報取得処理では、渡ってきたアンケートIDをもとに、アンケート情報テーブルとアンケート設問テーブルからデータを取得します。

アンケート情報テーブルとアンケート設問テーブルからはそれぞれ異なるSQLでデータを取得し、それらのデータを1つのデータにまとめて返却します。

データを取得できなかった場合はエラーを返します。



以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



この処理の作成手順について、以下の動画からも確認できます。併せてご利用ください。

注意

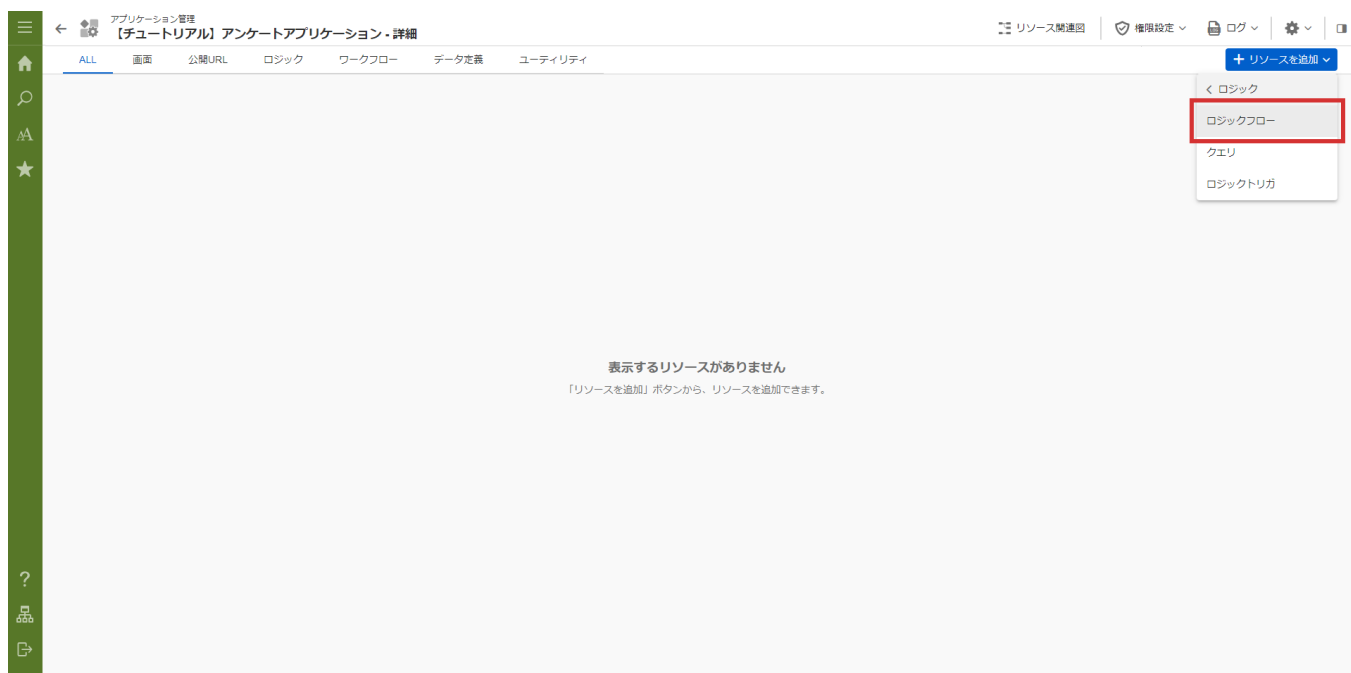
動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

ロジックの作成

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」の順に遷移します。アプリケーション一覧から先ほど作成した「【チュートリアル】アンケートアプリケーション」を選択します。



「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「ロジック」 - 「ロジックを新規作成」 - 「ロジックフロー」を選択し、「ロジックフロー定義編集画面」を表示します。



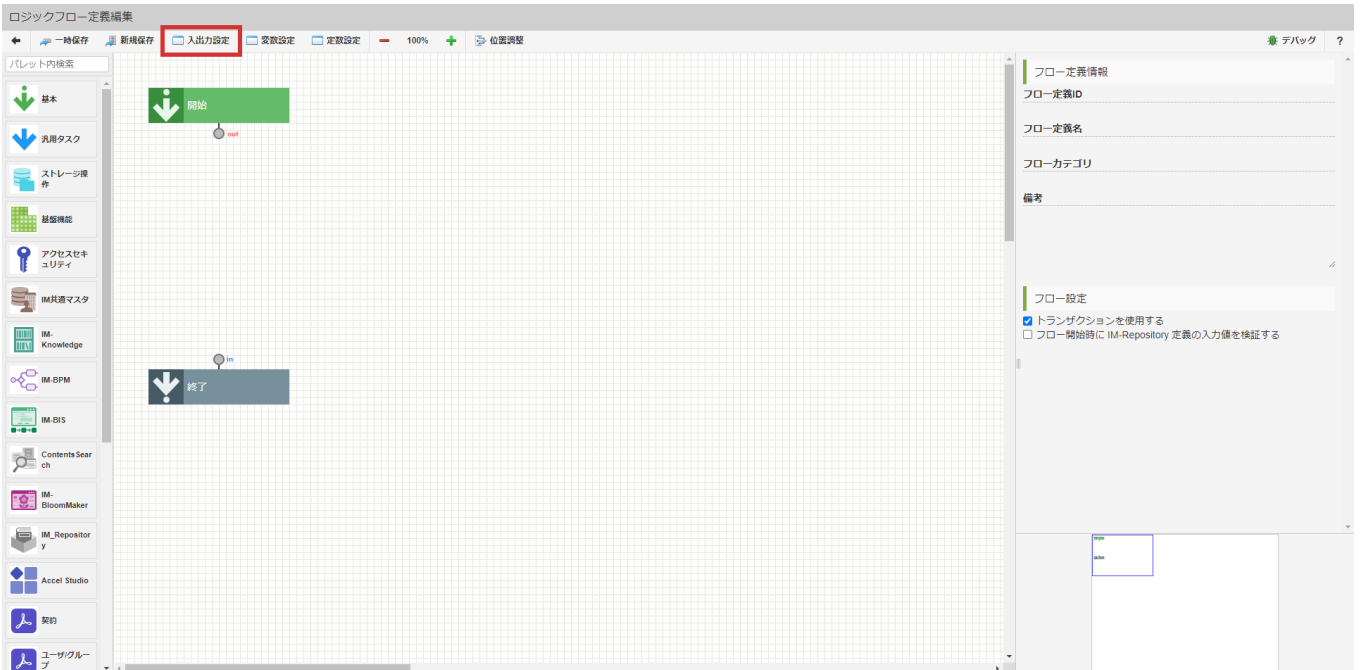
入出力値の設定

業務ロジックへの入力値と出力値を設定します。

アンケート情報取得処理は、アンケートIDを受け取り、アンケート情報とアンケート設問のデータを1つにまとめて返却します。

まず、入力値の設定を行います。

画面上部の「入出力設定」をクリックし「入出力設定」ダイアログを表示します。



入力値は「アンケートID」のみとするため、文字列の変数を追加します。
 「入力」 - 「+string」をクリックし、入力値一覧に「string1」を追加します。

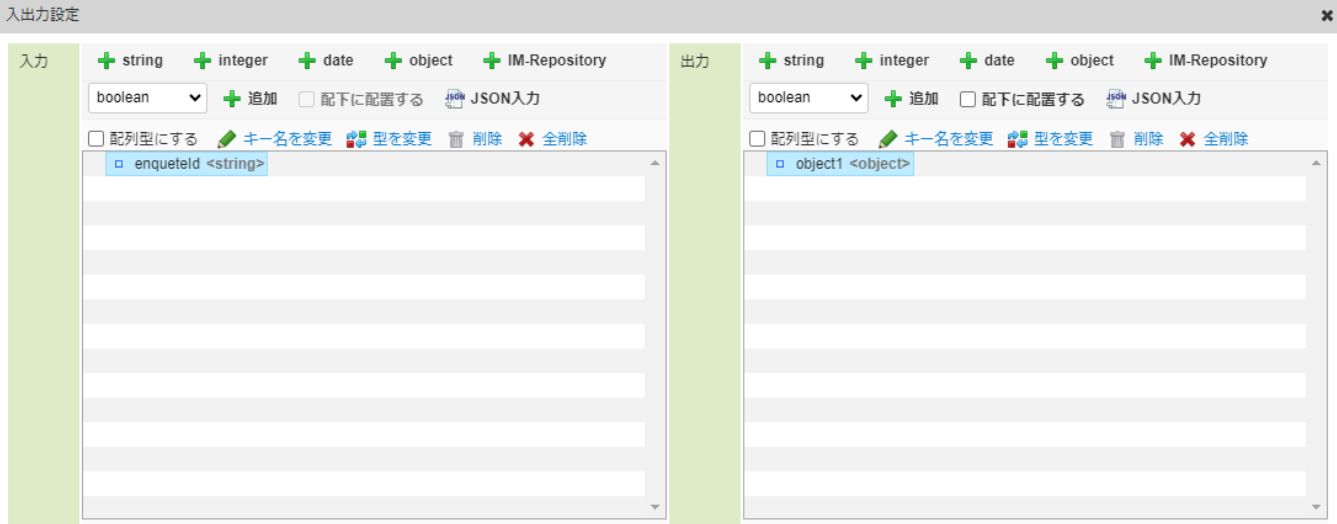


「string1」を選択した状態で入力値一覧上部の「キー名を変更」をクリックし、キー名を「enqueteld」に変更します。
 入力値はJSON入力以下のコードを記載しても設定できます。

```
{
  "enqueteld": ""
}
```

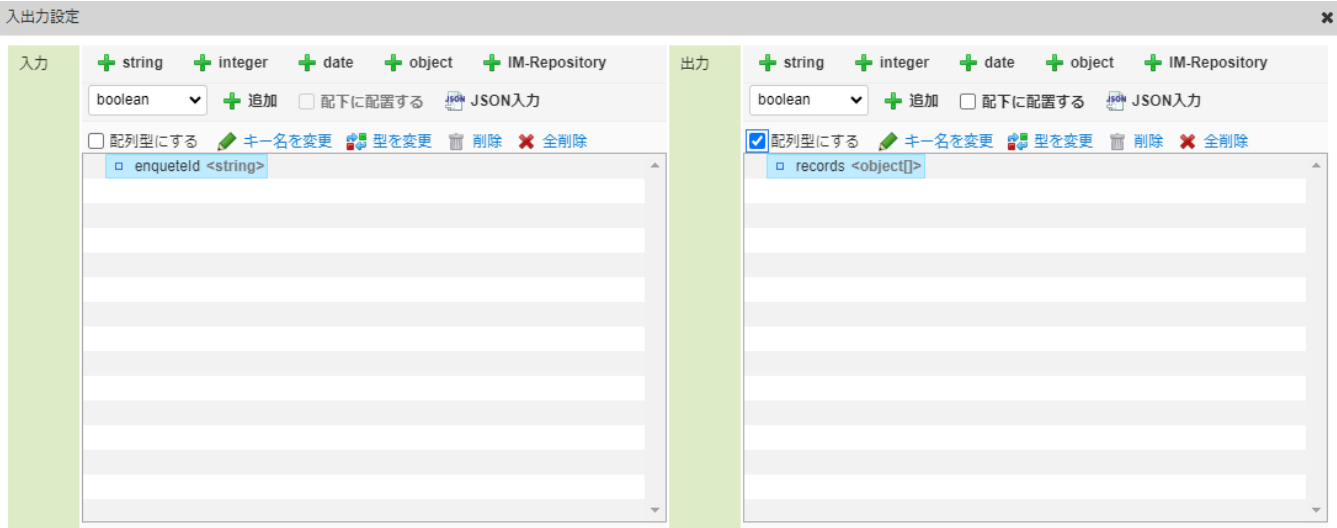
JSON入力する場合は、入力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

次に、出力値の設定を行います。
 まずは、設問内容を格納する配列を作成します。
 「出力」 - 「+object」をクリックすると出力値一覧に「object1」が追加されます。



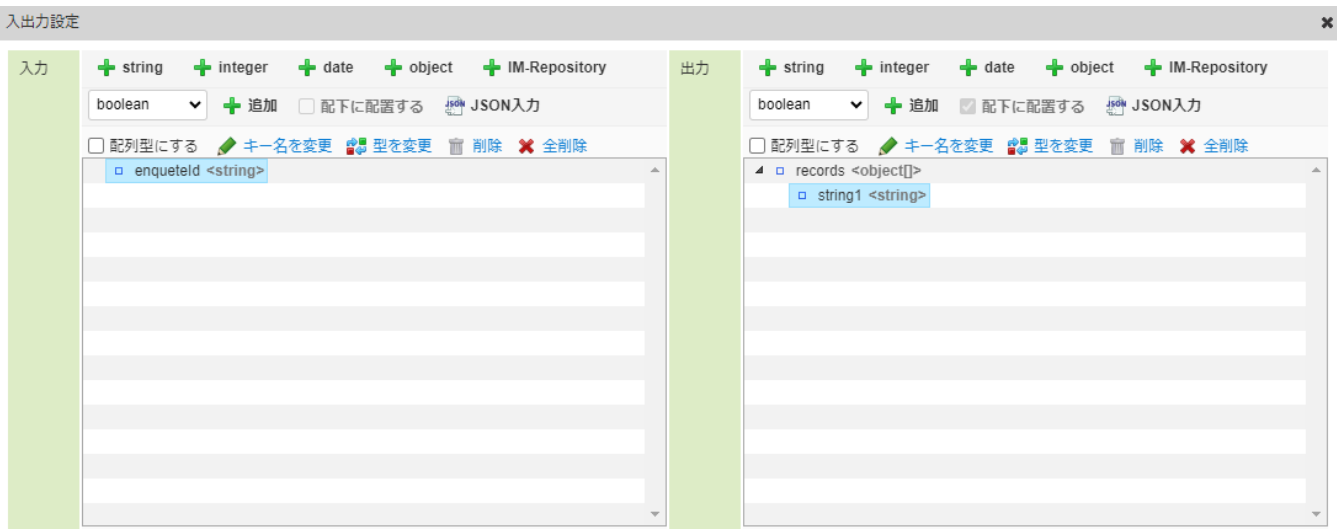
決定 取り消し

「object1」を選択した状態で出力値一覧上部の「キー名を変更」をクリックし、キー名を「records」に変更します。「records」を選択した状態で「配列型にする」にチェックをいれ、「records」を汎用オブジェクトの配列にします。

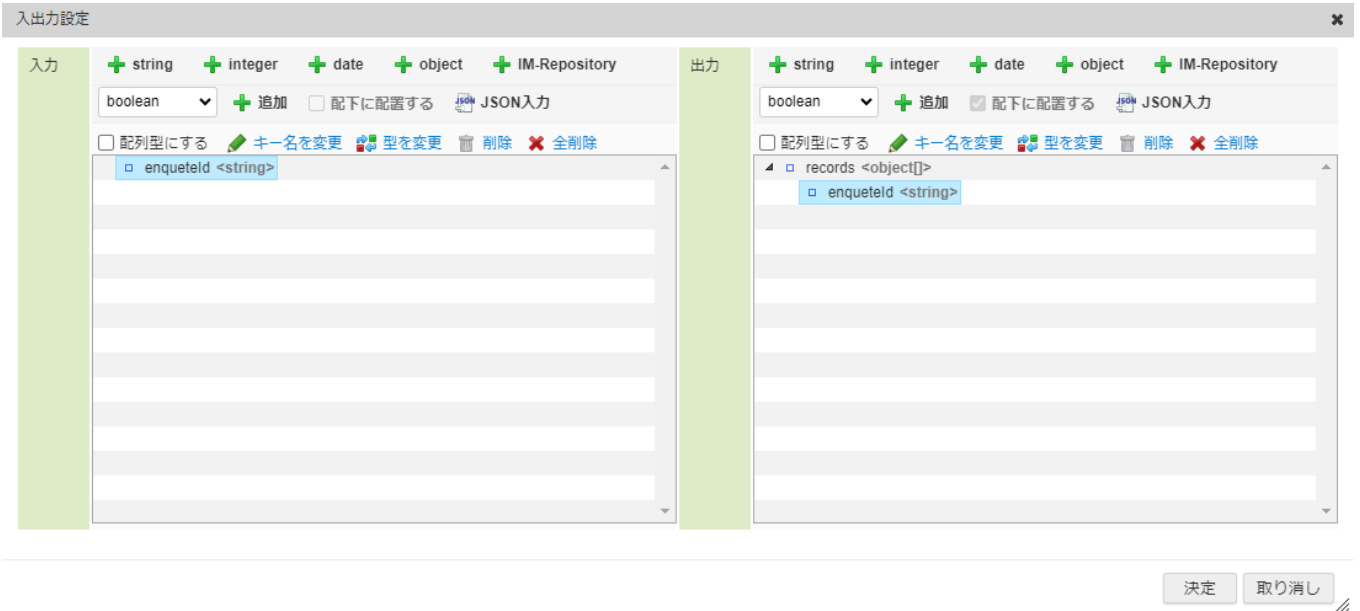


決定 取り消し

次に、各設問内容の値を格納していくための出力値を「records」配下に追加します。出力値一覧の「records」を選択した状態で「配下に配置する」にチェックを入れます。「出力」 - 「+string」をクリックし、「records」配下に「string1」という変数を追加します。



決定 取り消し



同様の手順で「records」配下に下記の設定の変数を追加します。

項目	型	キー名
アンケートID	string	enquete_id
設問ID	string	question_id
設問内容	string	question
回答項目の左端	string	left_scale
回答項目の右端	string	right_scale
設問番号	integer	question_no



次に、アンケート情報テーブルから取得した、「アンケートタイトル」と「アンケート概要」を格納するための出力値を作成します。
前記の作業で、配下に変数を追加してきた「records」を選択した状態で「配下に配置する」のチェックを外します。
「出力」 - 「+string」をクリックし、出力値一覧に「string1」を追加します。
「string1」を選択した状態で出力値一覧上部の「キー名を変更」をクリックし、キー名を「enqueteTitle」に変更します。

入出力設定

入力	出力
+ string + integer + date + object + IM-Repository boolean ▼ + 追加 <input type="checkbox"/> 配下に配置する JSON入力 <input type="checkbox"/> 配列型にする キー名を変更 型を変更 削除 全削除 □ enqueteld <string>	+ string + integer + date + object + IM-Repository boolean ▼ + 追加 <input checked="" type="checkbox"/> 配下に配置する JSON入力 <input type="checkbox"/> 配列型にする キー名を変更 型を変更 削除 全削除 ▲ □ records <object[]> □ enquete_id <string> □ question_id <string> □ question <string> □ left_scale <string> □ right_scale <string> □ question_no <double> □ enqueteTitle <string>

決定 取り消し

「出力」 - 「+string」をクリックし、出力値一覧に「string1」を追加します。

「string1」を選択した状態で出力値一覧上部の「キー名を変更」をクリックし、キー名を「enqueteOverview」に変更します。

入出力設定

入力	出力
+ string + integer + date + object + IM-Repository boolean ▼ + 追加 <input type="checkbox"/> 配下に配置する JSON入力 <input type="checkbox"/> 配列型にする キー名を変更 型を変更 削除 全削除 □ enqueteld <string>	+ string + integer + date + object + IM-Repository boolean ▼ + 追加 <input checked="" type="checkbox"/> 配下に配置する JSON入力 <input type="checkbox"/> 配列型にする キー名を変更 型を変更 削除 全削除 ▲ □ records <object[]> □ enquete_id <string> □ question_id <string> □ question <string> □ left_scale <string> □ right_scale <string> □ question_no <double> □ enqueteTitle <string> □ enqueteOverview <string>

決定 取り消し

出力値はJSON入力下記コードを記述して設定できます。

```
{
  "records": [
    {
      "enquete_id": "",
      "question_id": "",
      "question": "",
      "left_scale": "",
      "right_scale": "",
      "question_no": 0
    }
  ],
  "enqueteTitle": "",
  "enqueteOverview": ""
}
```

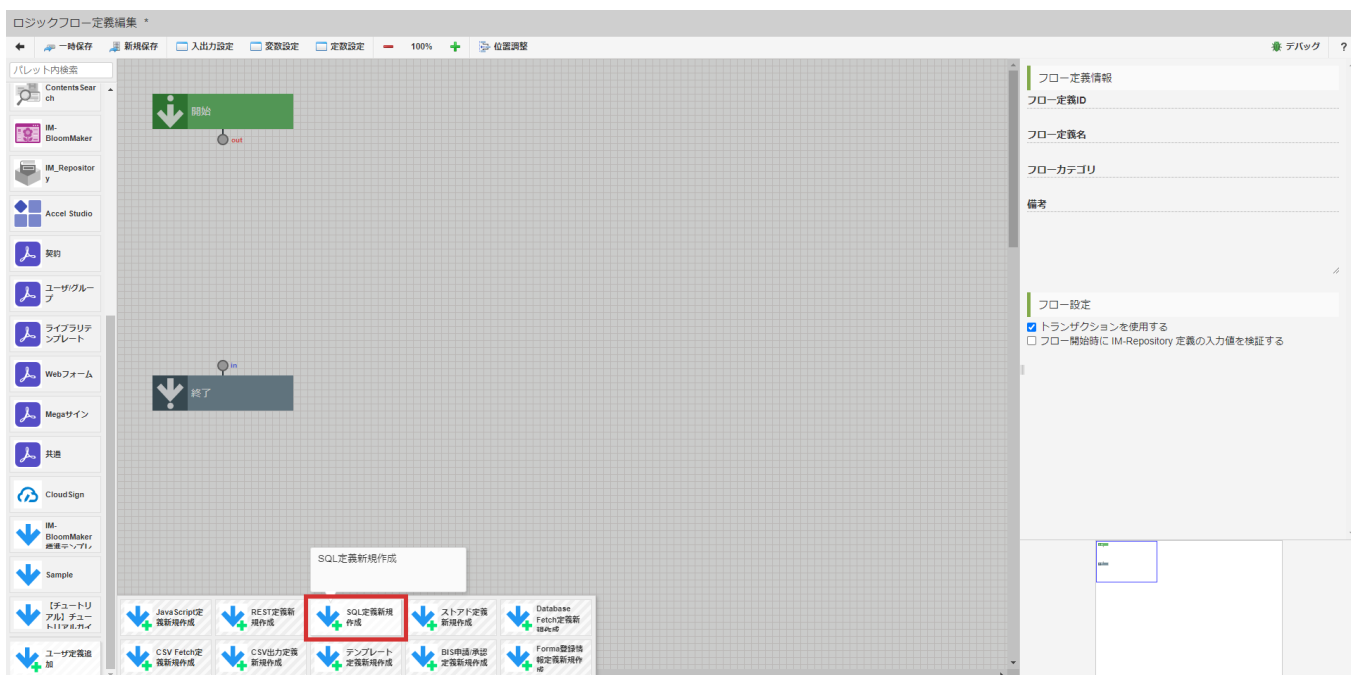
JSON入力する場合は、出力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

全ての入出力設定が完了したら、入出力設定ダイアログの「決定」をクリックし、設定を終了します。

アンケート情報取得タスクの作成

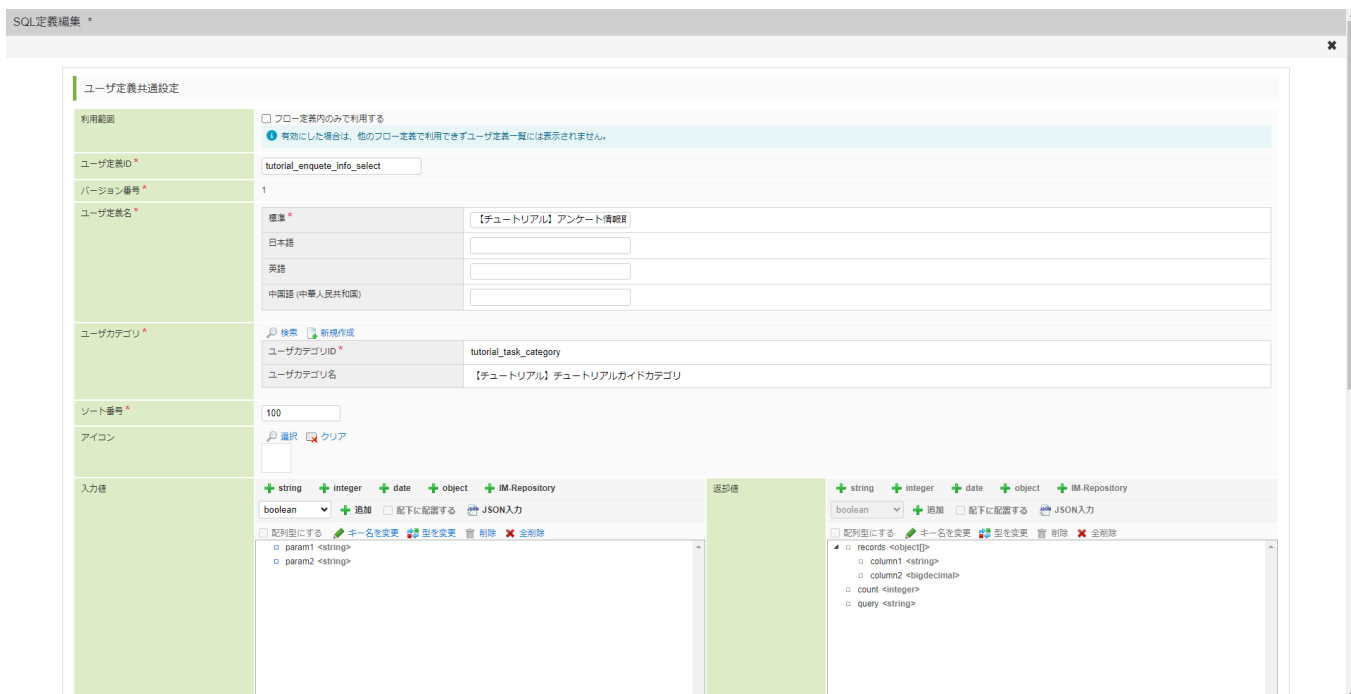
アンケートIDをもとに、アンケート情報テーブルからデータを取得するユーザ定義タスクを作成します。

パレットで「ユーザ定義追加」にカーソルを合わせ、「SQL定義新規作成」を選択します。



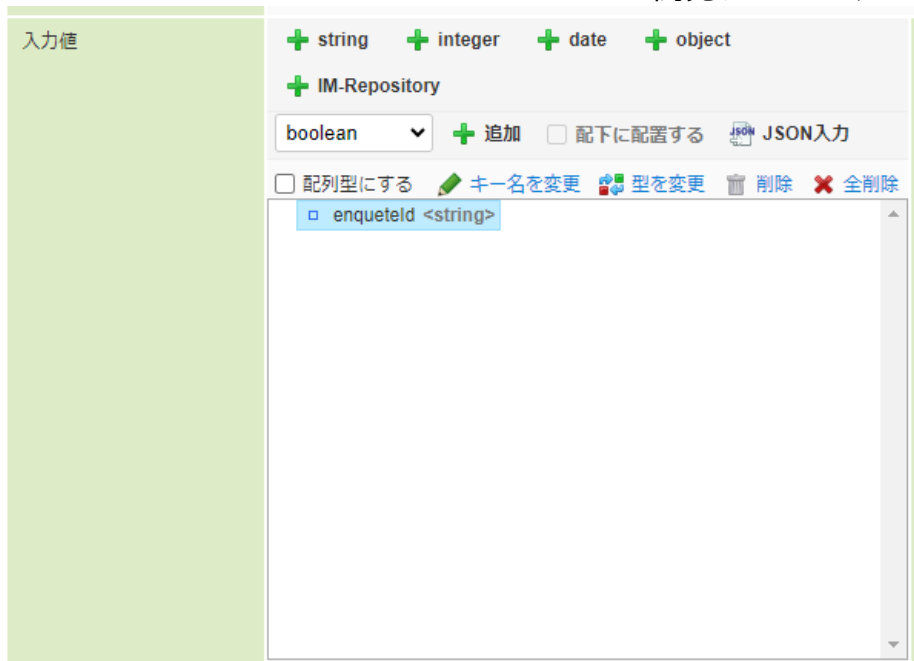
ユーザ定義共通設定を下記の通り設定します。

ラベル	入力値
ユーザ定義ID	tutorial_enquete_info_select
ユーザ定義名 (標準)	【チュートリアル】 アンケート情報取得
ユーザカテゴリ	【チュートリアル】 チュートリアルガイドカテゴリ



入力値に以下を設定します。

項目	型	キー名
アンケートID	string	enqueteld

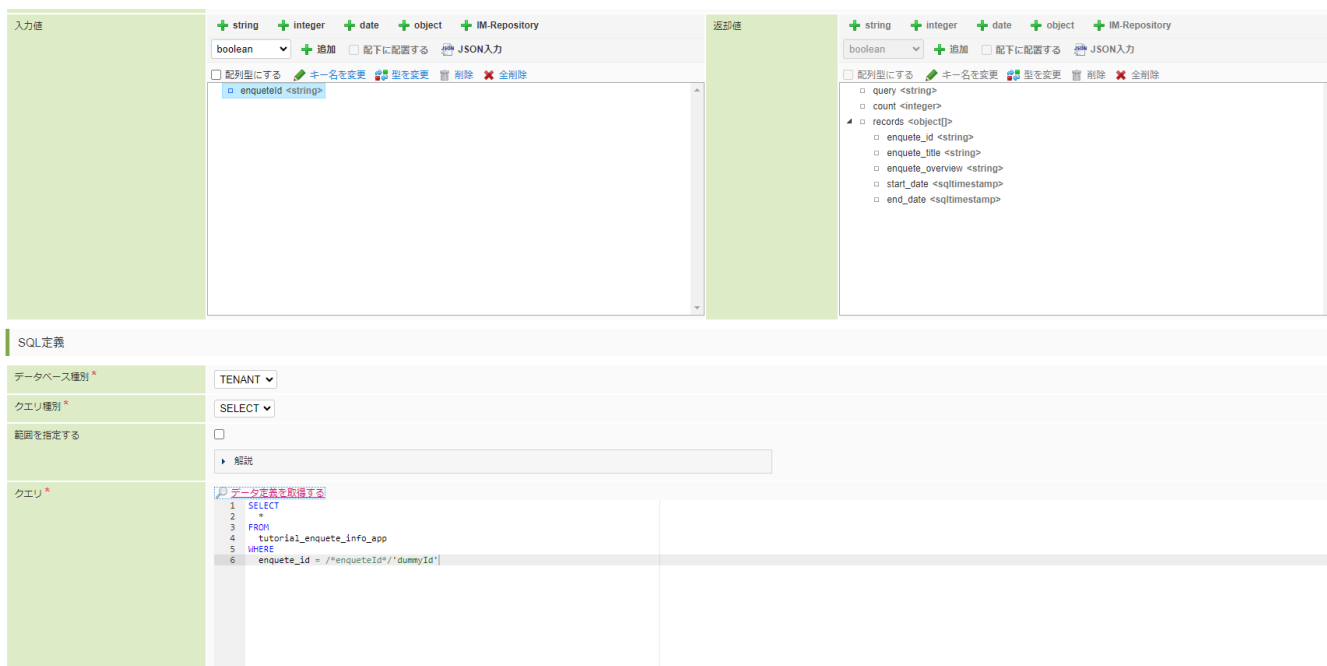


クエリに以下のSQLを設定します。

```

SELECT
*
FROM
tutorial_enquete_info_app
WHERE
enquete_id = /*enqueteId*/'dummyId'
    
```

クエリ設定上部にある「データ定義を取得する」をクリックし、タスクの出力値を設定します。



「登録」をクリックし、タスクを登録します。

アンケート設問取得タスクの作成

アンケート情報取得タスクと同様にアンケートIDをもとに、アンケート設問テーブルからデータを取得するユーザ定義タスクを作成します。

パレットで「ユーザ定義追加」にカーソルを合わせ、「SQL定義新規作成」を選択します。

ユーザ定義共通設定を下記の通り設定します。

ラベル	入力値
ユーザ定義ID	tutorial_enquete_question_select
ユーザ定義名（標準）	【チュートリアル】 アンケート設問取得
ユーザカテゴリ	【チュートリアル】 チュートリアルガイドカテゴリ

入力値に以下を設定します。

項目	型	キー名
アンケートID	string	enqueteld

クエリに以下のSQLを設定します。

```

SELECT
*
FROM
tutorial_enquete_question_app
WHERE
enquete_id = /*enqueteld*/'dummyId'
ORDER BY
question_no ASC
    
```

クエリ設定上部にある「データ定義を取得する」をクリックし、タスクの出力値を設定します。

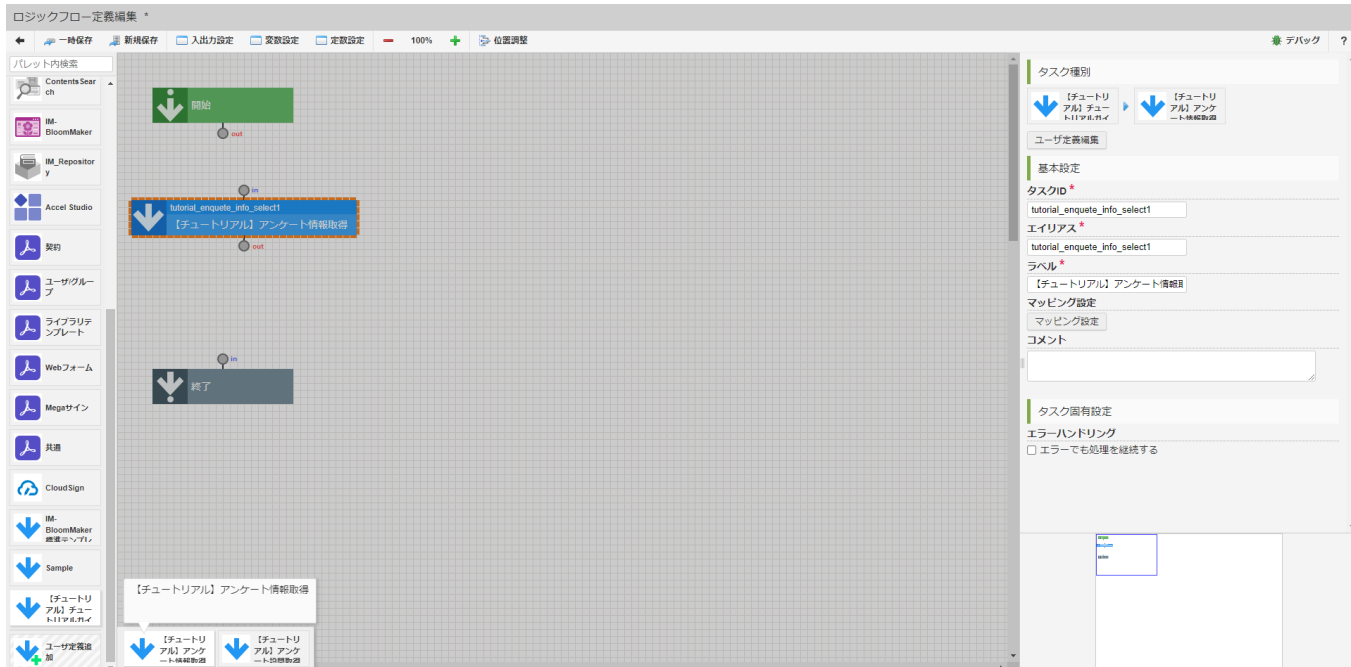
The screenshot shows the configuration interface for a task. At the top, there are two panels for '入力値' (Input Values) and '返却値' (Return Values). The '入力値' panel shows a 'boolean' type with a dropdown menu and a '+ 追加' (Add) button. The '返却値' panel shows a 'boolean' type with a dropdown menu and a '+ 追加' (Add) button. Below these panels is the 'SQL定義' (SQL Definition) section, which includes a 'データベース種別' (Database Type) dropdown set to 'TENANT', a 'クエリ種別' (Query Type) dropdown set to 'SELECT', and a 'クエリ' (Query) text area containing the SQL code from the previous block. A red box highlights the 'データ定義を取得する' (Get Data Definition) button in the query area. To the right of the SQL definition, there is a tree view showing the output structure: 'query <string>', 'count <integer>', and 'records <object[]>' containing 'enquete_id <string>', 'question_id <string>', 'question_no <bigdecimal>', 'question <string>', 'left_scale <string>', and 'right_scale <string>'.

「登録」をクリックし、タスクを登録します。

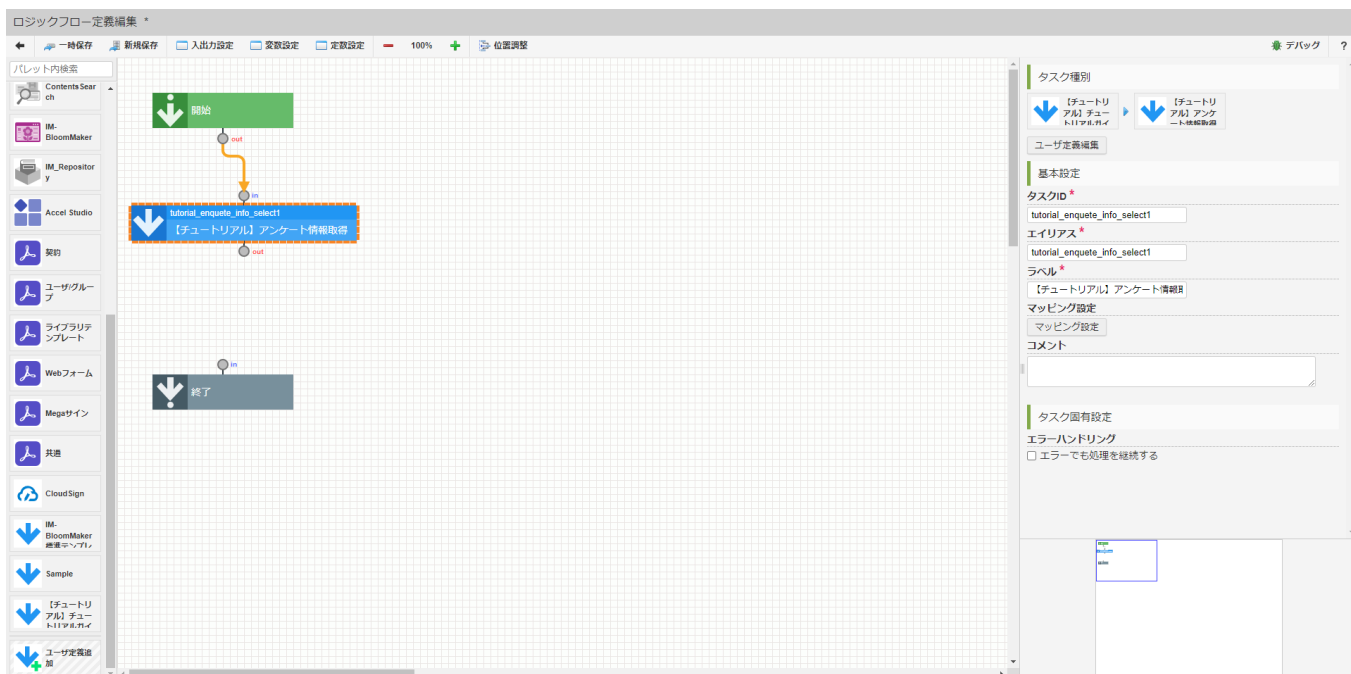
タスクの配置と変数設定

作成した「【チュートリアル】 アンケート情報取得」タスクと「【チュートリアル】 アンケート設問取得」タスクをフローに配置し、線で繋がります。

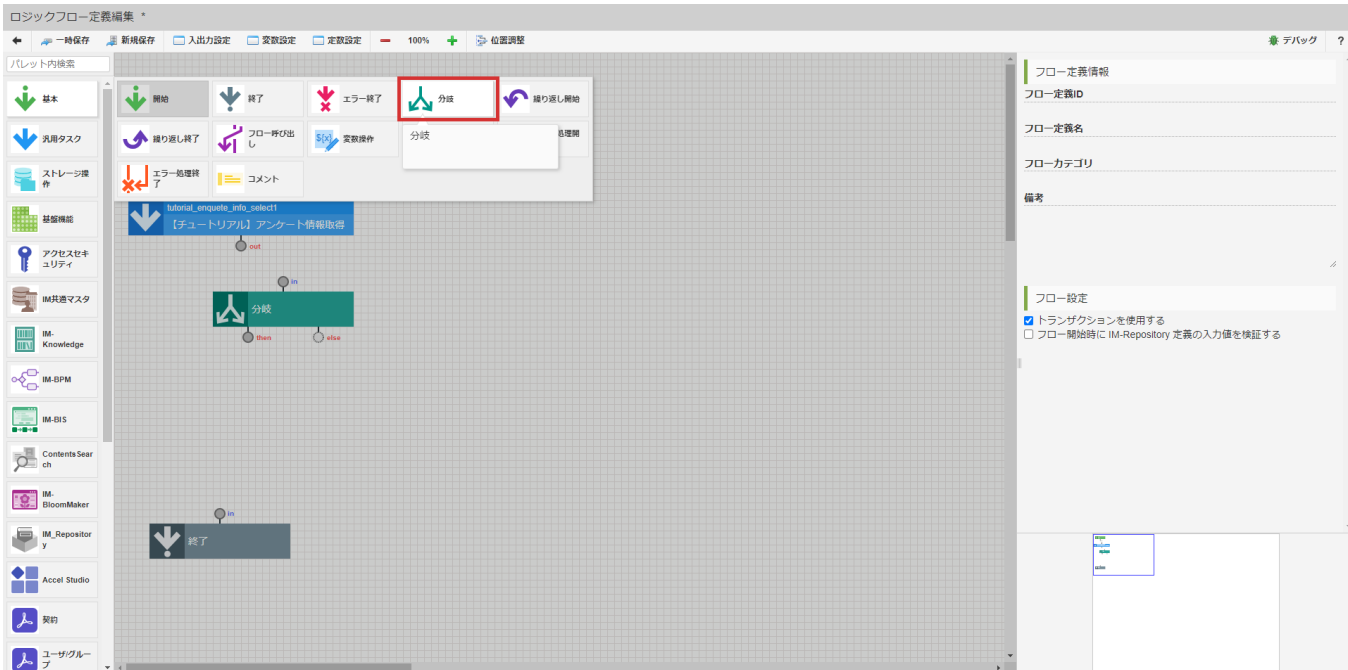
まずは、パレットから「【チュートリアル】 チュートリアルガイドカテゴリ」 - 「【チュートリアル】 アンケート情報取得」を選択し、「開始」の次のタスクとして配置します。



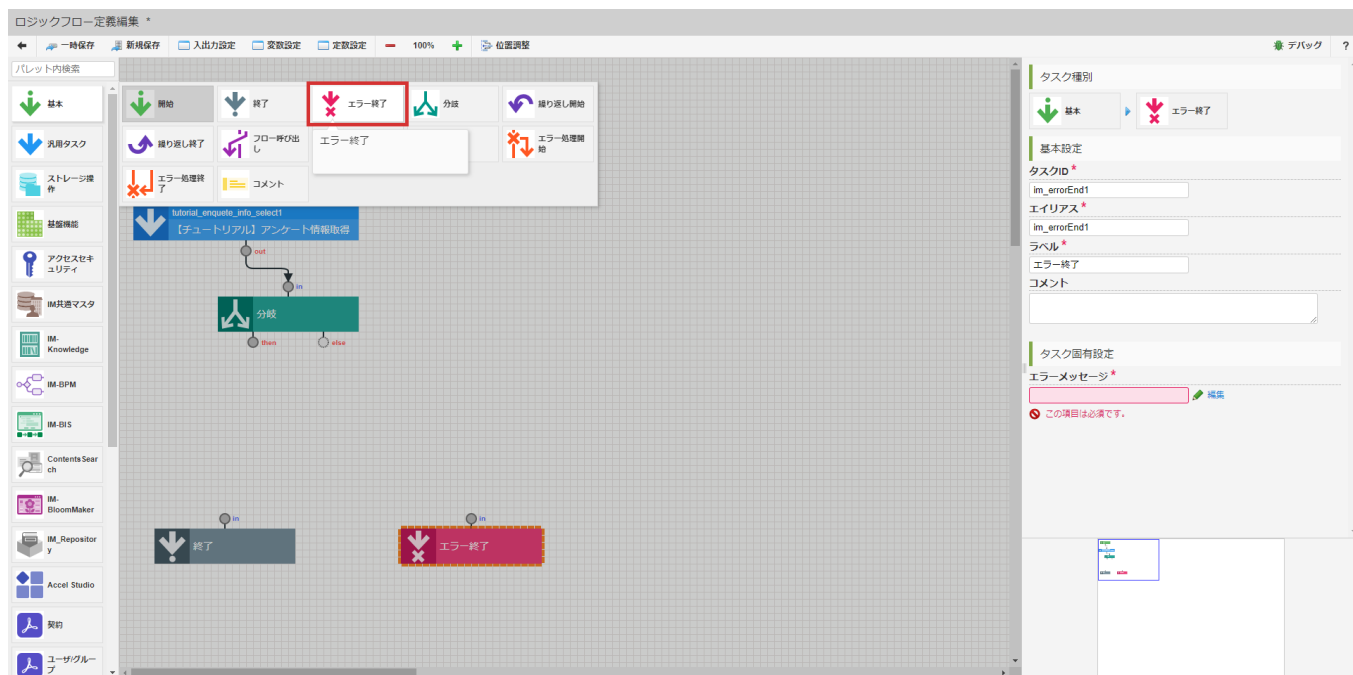
「開始」の「out」からドラッグし、「【チュートリアル】アンケート情報取得」の「in」にドロップしてタスク同士を線で繋がめます。



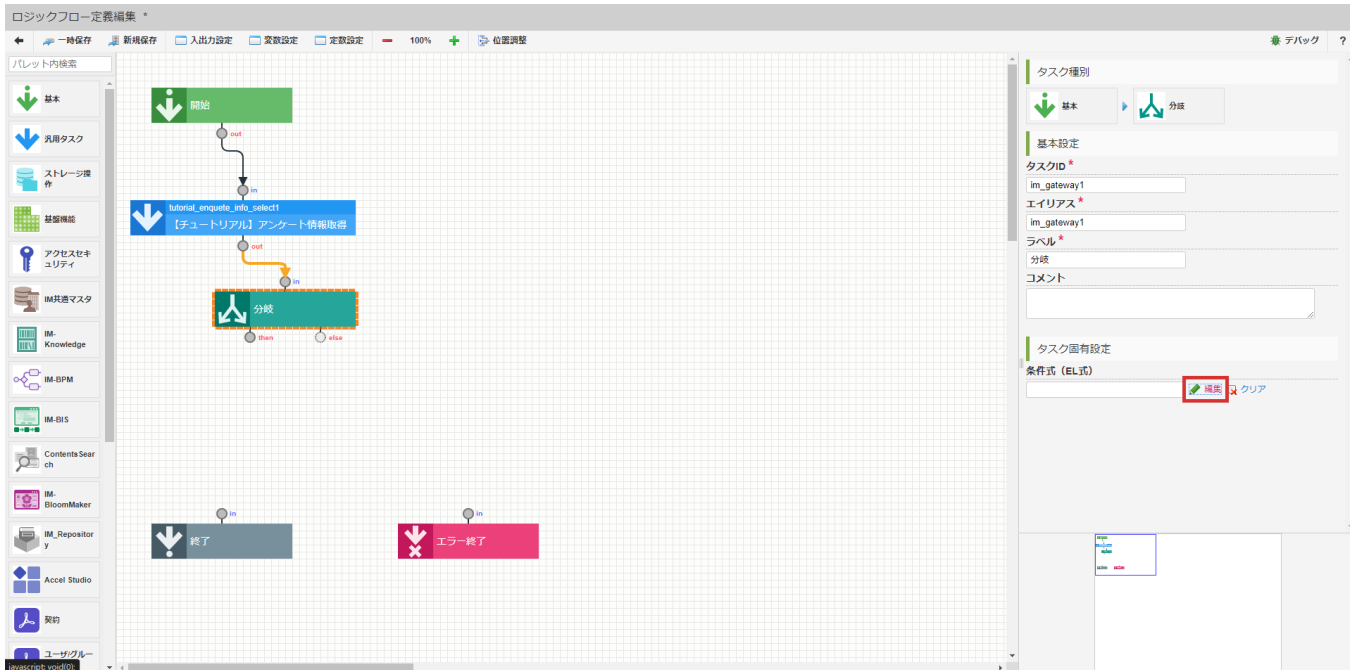
アンケートIDの誤り等でアンケート情報が取得できなかった場合、エラーを返却します。
 パレットから「基本」 - 「分岐」を選択し、「【チュートリアル】アンケート情報取得」タスクの次の要素として配置します。



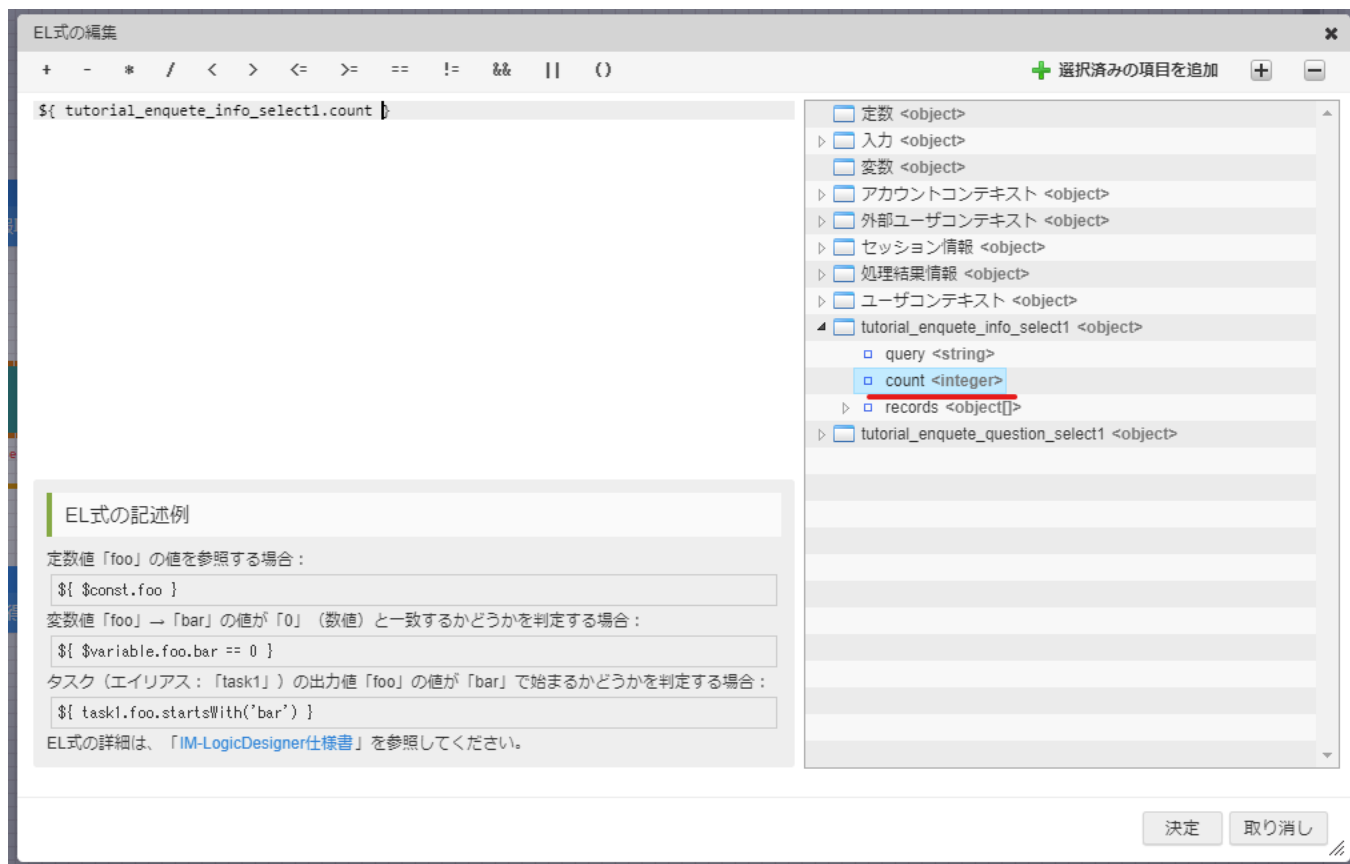
「【チュートリアル】アンケート情報取得」タスクの「out」と「分岐」の「in」を線で繋がめます。
 パレットから「基本」 - 「エラー終了」を選択し、「分岐」の次の要素として配置します。



配置した分岐をクリックして選択し、画面の右側領域で「タスク固有設定」-「条件式 (EL式)」の「編集」をクリックします。



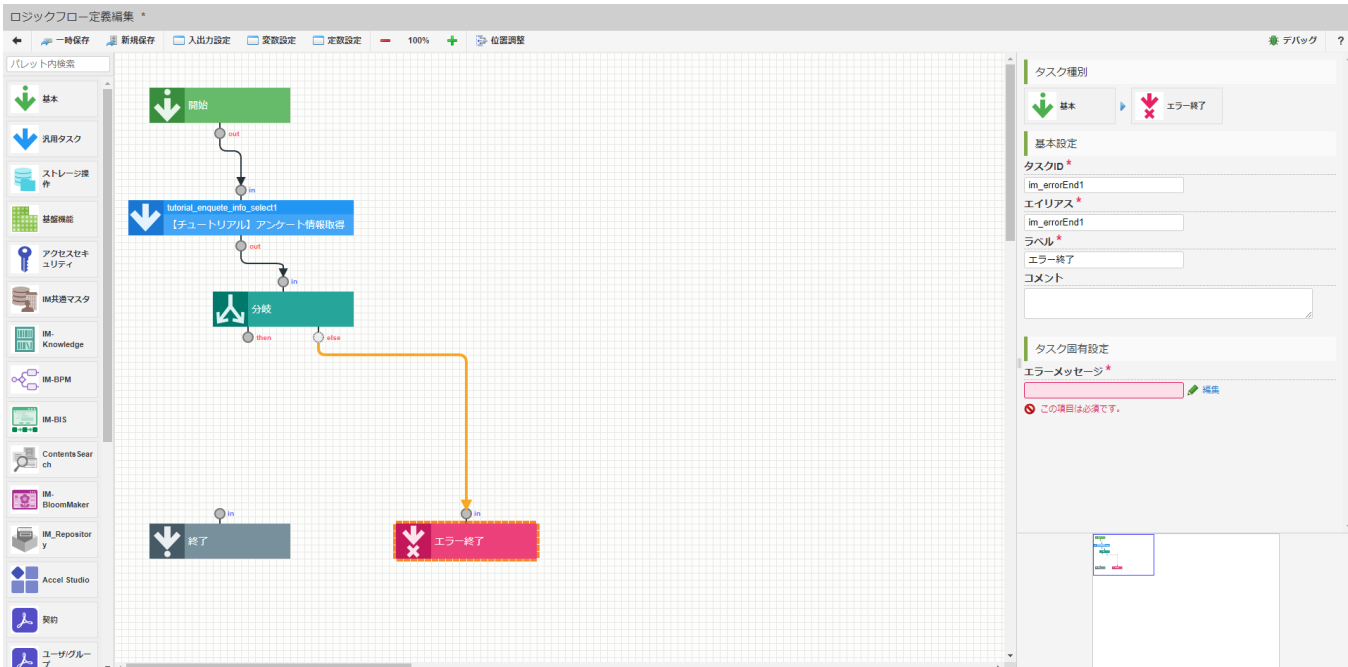
「EL式の編集」ダイアログの右側から「tutorial_enquete_info_select1 > count」を選択します。



画面上部のエディタ領域で条件式（EL式）を下記の通りに設定し、「EL式の編集」ダイアログの「決定」をクリックして編集を終了します。

`\${tutorial_enquete_info_select1.count > 0}`

「分岐」の「else」と「エラー終了」の「in」を線で繋がめます。

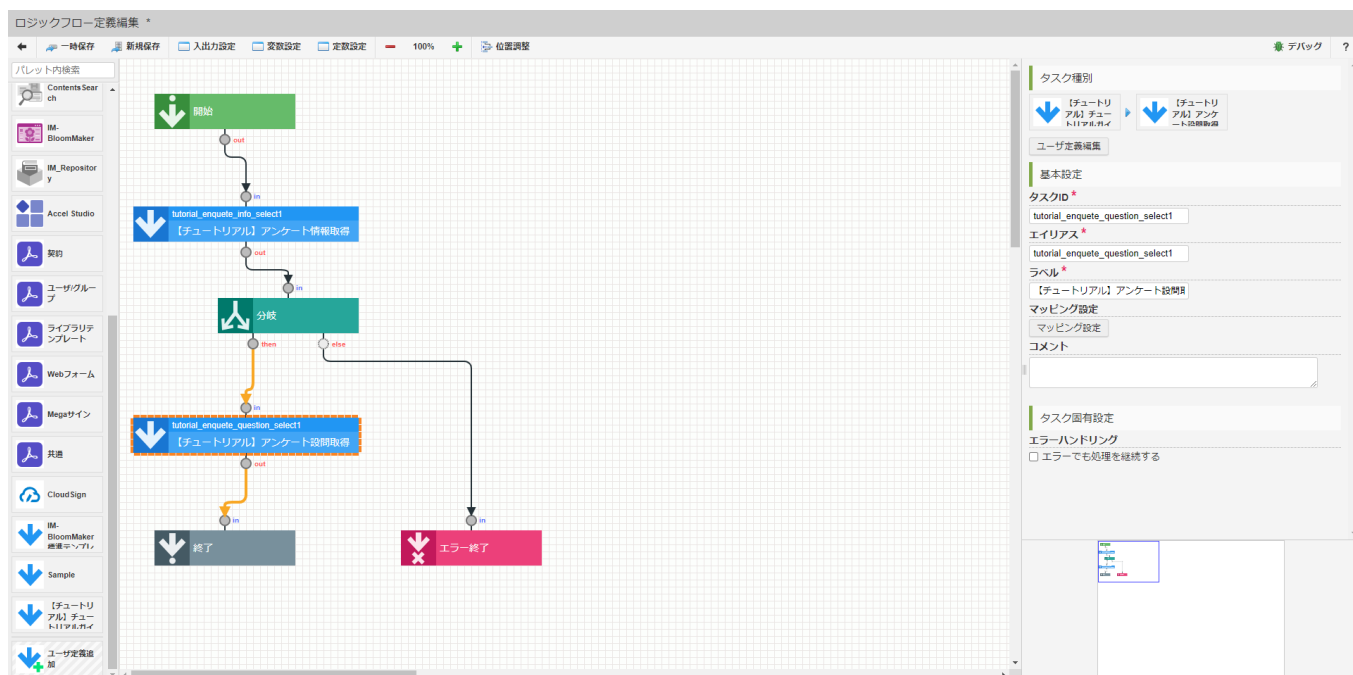


「エラー終了」をクリックし、画面の右側領域の「エラーメッセージ」の入力ボックスに「no enquete」と入力します。

パレットから「【チュートリアル】チュートリアルガイドカテゴリ」 - 「【チュートリアル】アンケート設問取得」を選択し、「分岐」の次のタスクとして配置します。

「分岐」の「then」と「【チュートリアル】アンケート設問取得」の「in」を線で繋がります。

「【チュートリアル】アンケート設問取得」の「out」と「終了」の「in」を線で繋がります。



次に、各タスクの変数設定を行います。

「【チュートリアル】アンケート情報取得」タスクをクリックし、画面の右側領域の「マッピング設定」をクリックしてマッピング設定ダイアログを表示します。

デバッグ

決定

取り消し

ダイアログ左側の入力の「enqueteld」の端子からドラッグし、と右側の「tutorial_enquete_info_select1」の「enqueteld」の端子でドロップして線で繋がります。

「決定」をクリックしてマッピング設定を終了します。

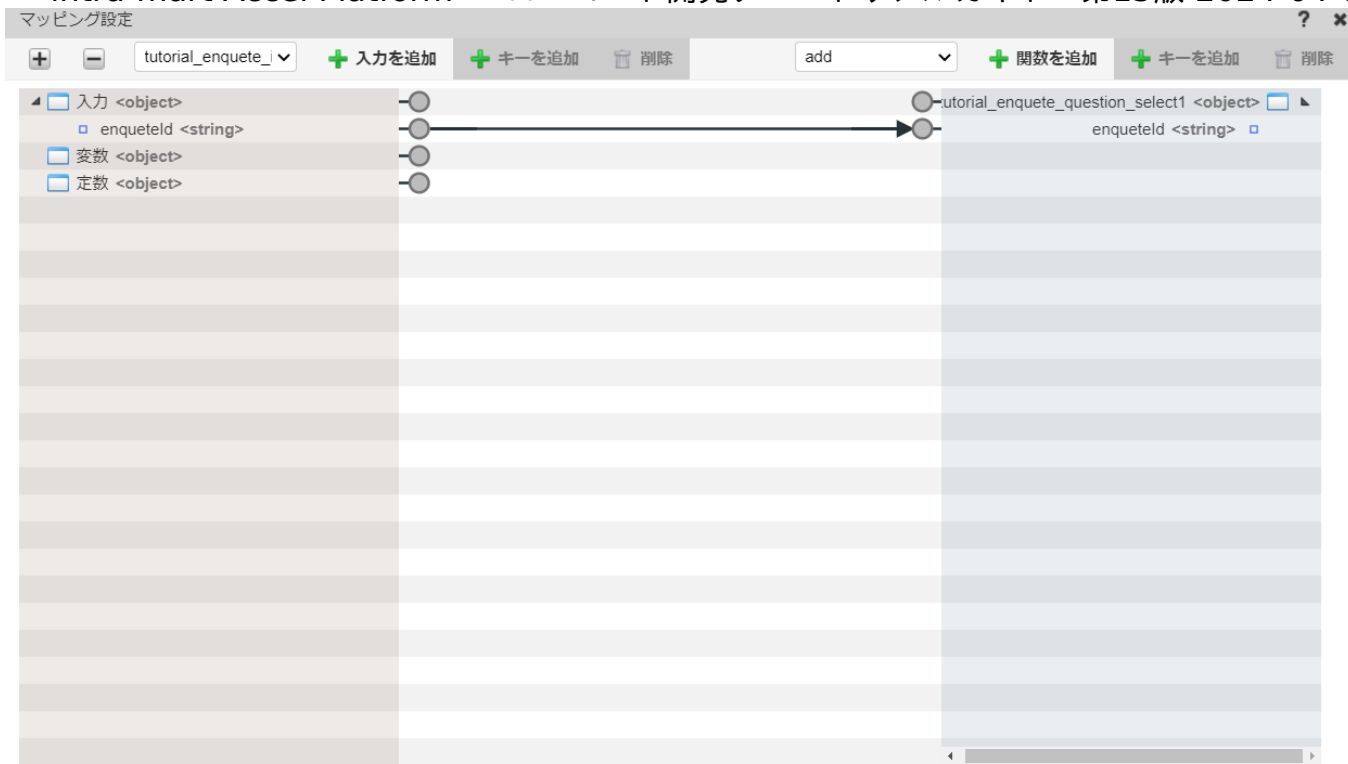
デバッグ

決定

取り消し

同様に、「【チュートリアル】アンケート設問取得」タスクのマッピング設定を行います。

ダイアログ左側の入力の「enqueteld」と右側の「tutorial_enquete_info_select1」の「enqueteld」を線で繋げ、「決定」をクリックしてマッピング設定を終了します。



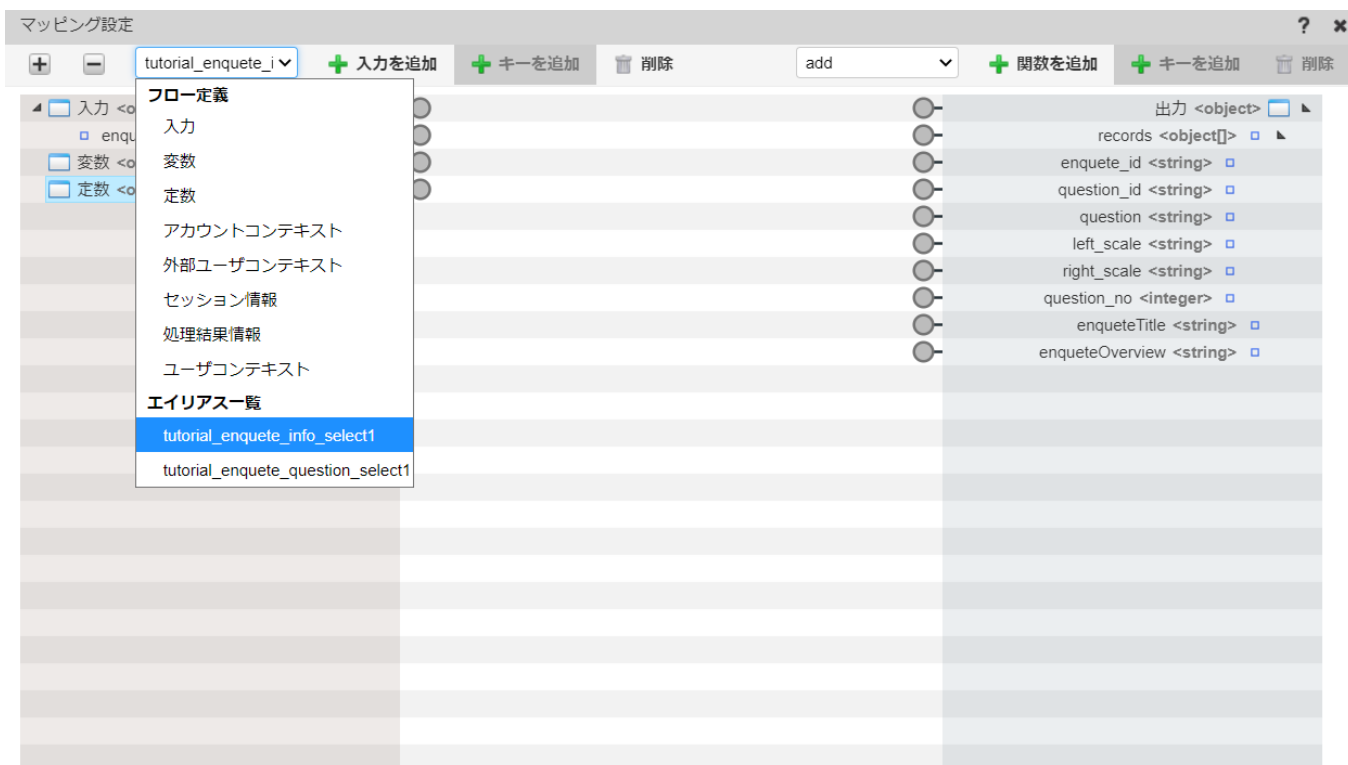
デバッグ

決定

取り消し

最後に「終了」要素のマッピング設定を行います。

「マッピング設定」ダイアログの入力側プルダウンから「エイリアス一覧」 - 「tutorial_enquete_info_select1」を選択し、「+入力を追加」をクリックします。



デバッグ

決定

取り消し

追加した「tutorial_enquete_info_select1」の「enquete_title」と出力の「enqueteTitle」を線で繋がります。

同様に「tutorial_enquete_info_select1」の「enquete_overview」と出力の「enqueteOverview」を線で繋がります。

デバッグ 決定 取り消し

次に、入力側プルダウンから「エイリアス一覧」 - 「tutorial_enquete_question_select1」を選択し、「+入力を追加」をクリックします。

追加した「tutorial_enquete_question_select1」の「records」と出力の「records」を線で繋がめます。

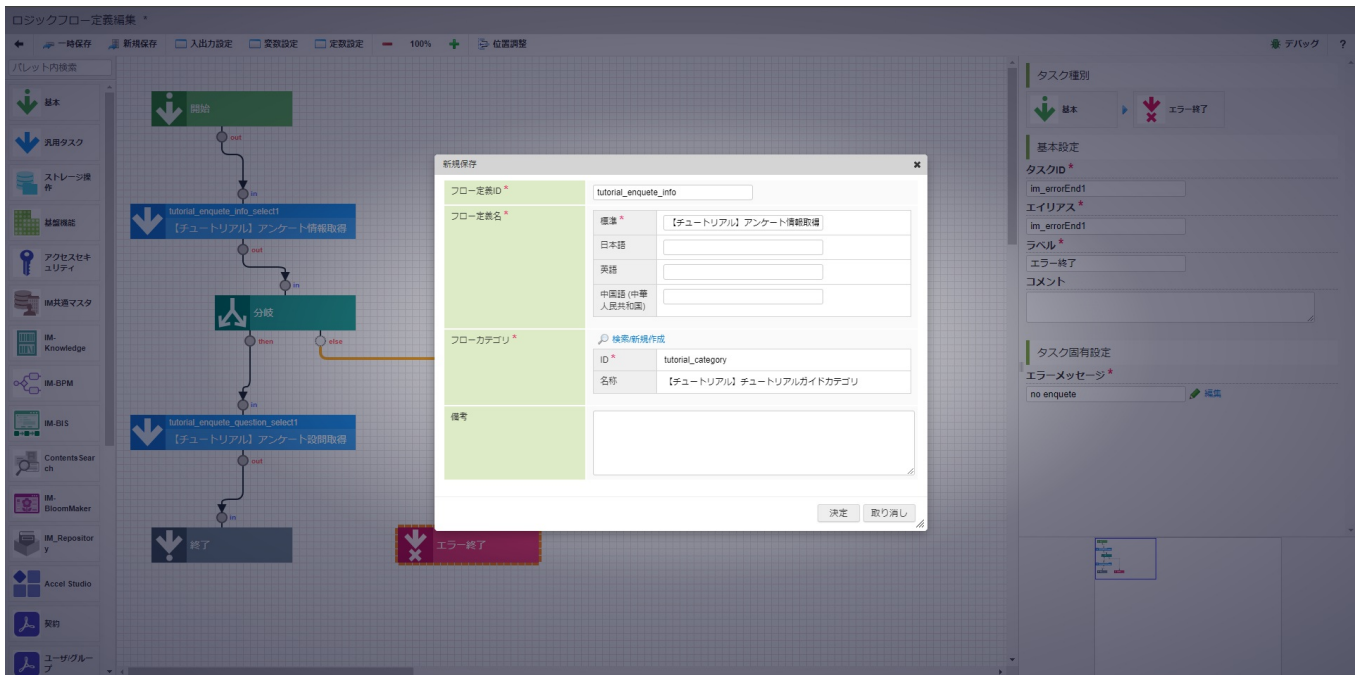
ここまでの設定ができれば「決定」をクリックし、マッピング設定を終了します。

デバッグ 決定 取り消し

これでアンケート情報取得処理は作成完了です。

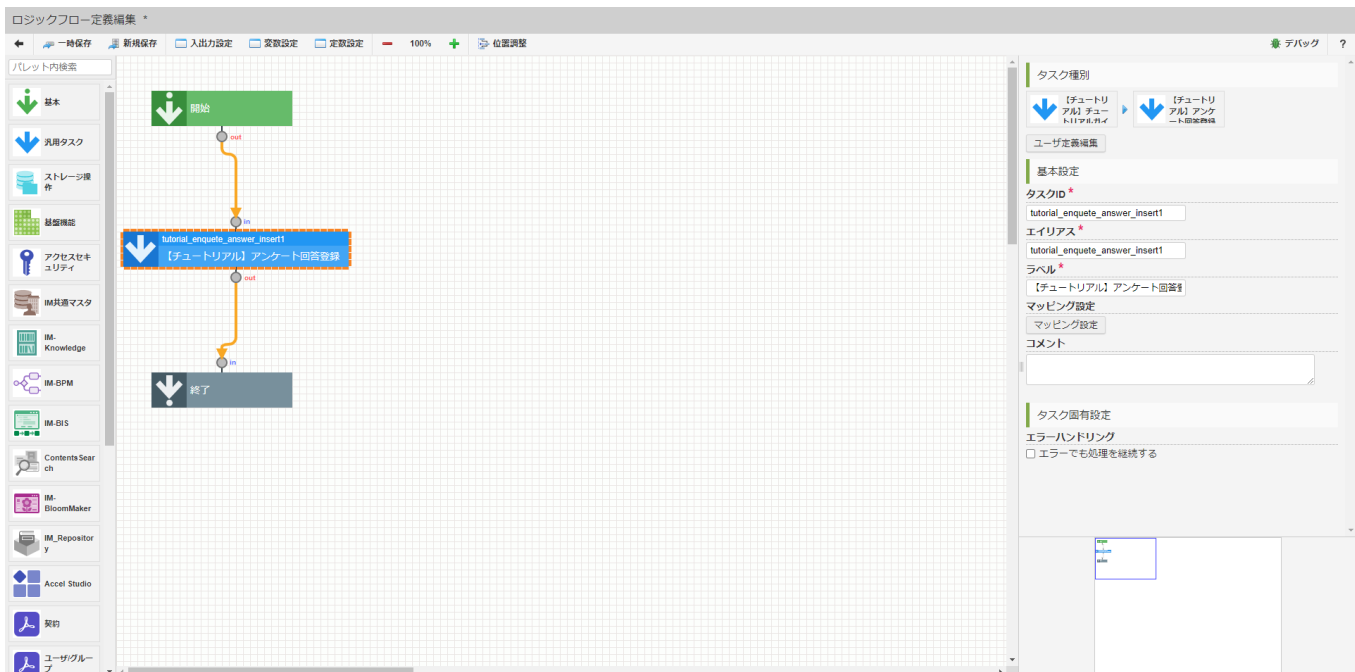
「ロジックフロー定義編集」画面上部の「新規保存」をクリックし、「新規保存」ダイアログの各項目を下記の通り入力してください。入力が完了したら「決定」をクリックして、ロジックを登録します。

ラベル	入力値
フロー定義ID	tutorial_enquete_info
フロー定義名（標準）	【チュートリアル】 アンケート情報取得
フローカテゴリ	【チュートリアル】 チュートリアルガイドカテゴリ



回答登録処理の作成

アンケート回答登録処理では、渡ってきた回答情報をアンケート回答テーブルに格納します。回答内容は配列型のデータとして渡ってくるため、配列をカンマ区切りの文字列に変換し、データベースに1つのデータとして格納します。



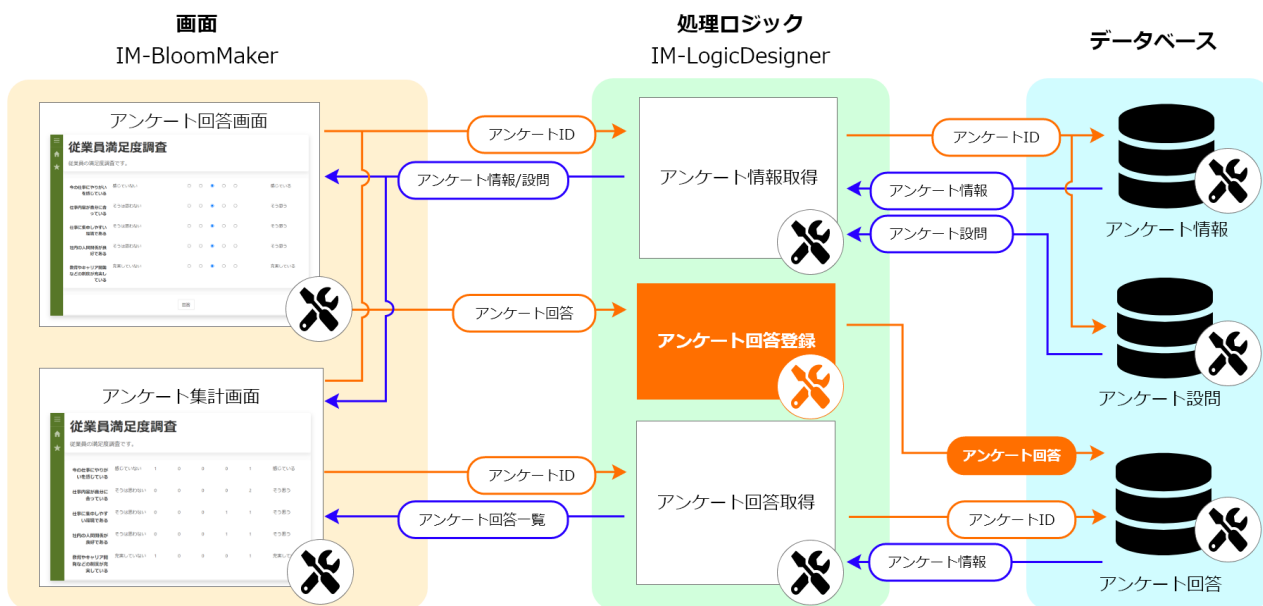
以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



チュートリアルで編集/作成



システムで自動生成



この処理の作成手順について、以下の動画からも確認できます。併せてご利用ください。



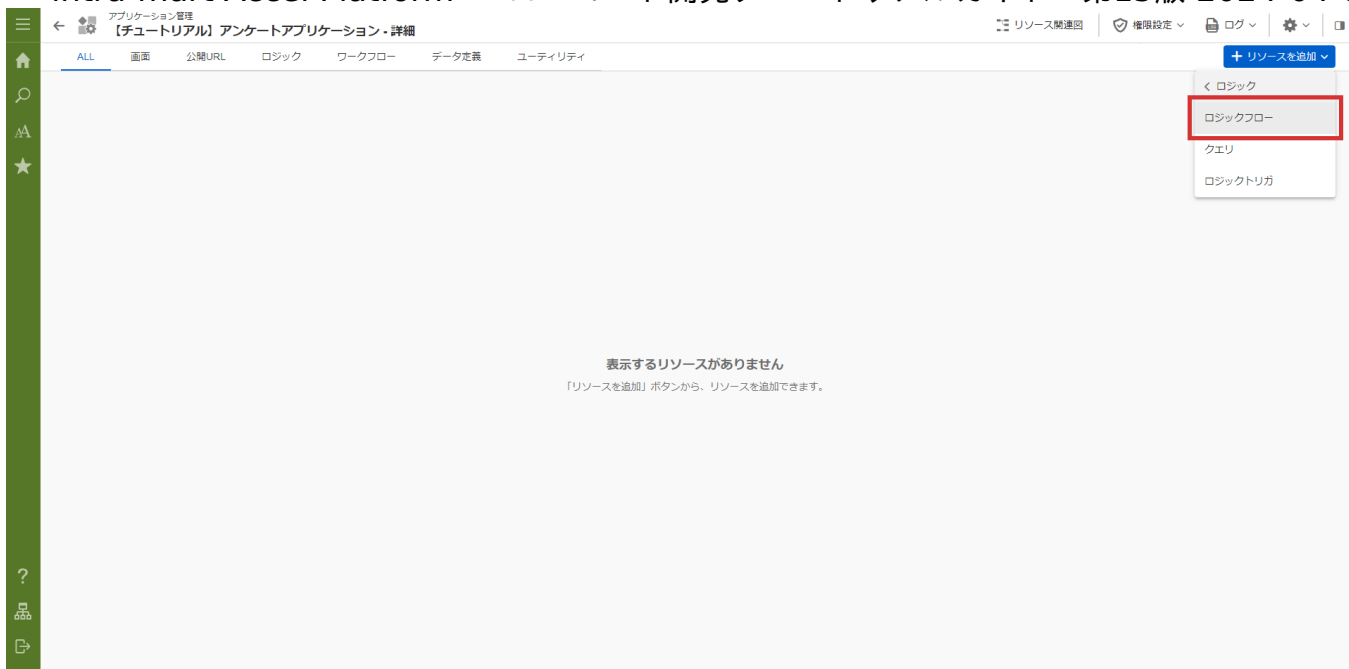
注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

ロジックの作成

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「ロジック」-「ロジックを新規作成」-「ロジックフロー」を選択し、「ロジックフロー定義編集画面」を表示します。

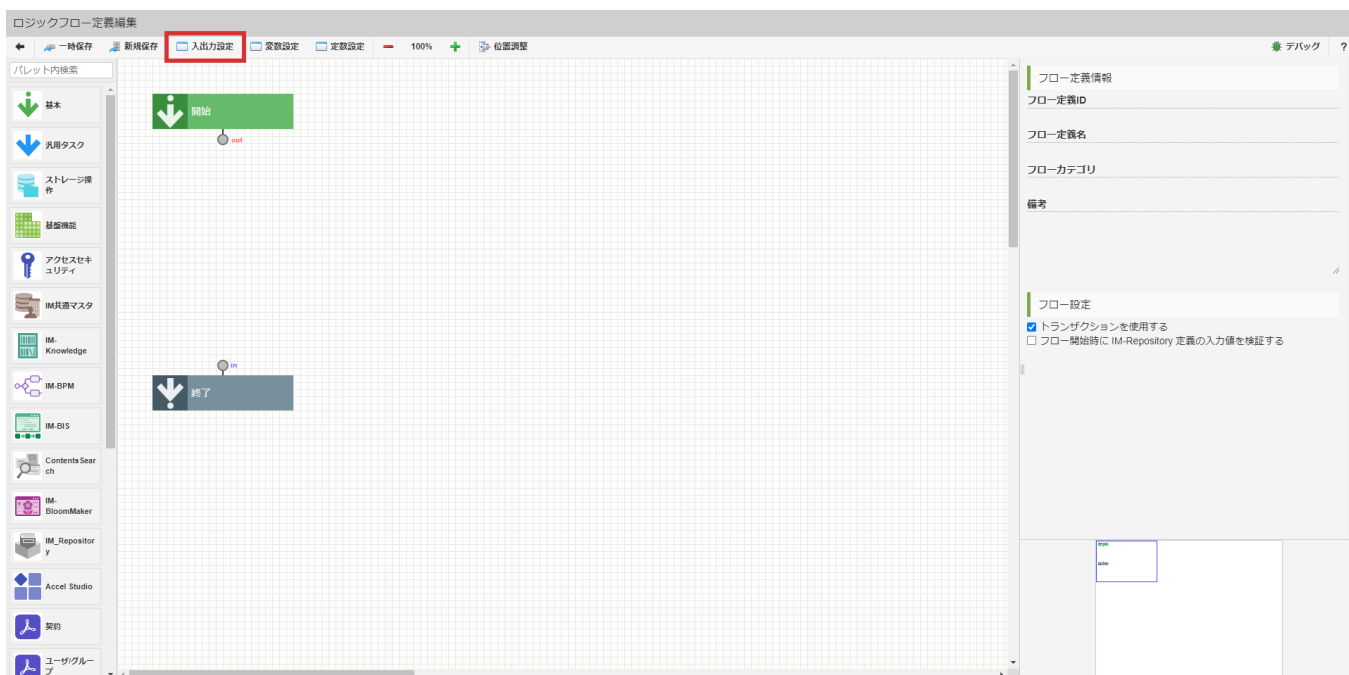


入出力値の設定

業務ロジックへの入力値と出力値を設定します。

アンケート回答登録処理は、アンケートIDとアンケートの回答内容を配列型のデータとして受け取ります。出力値は設定しません。

入力値の設定を行うため、画面上部の「入出力設定」をクリックし「入出力設定」ダイアログを表示します。



入力値は「アンケートID」と「回答情報」のみなので、文字列の変数を追加します。

「入力」 - 「+string」をクリックし、入力値一覧に「string1」を追加します。

決定 取り消し

「string1」を選択した状態で入力値一覧上部の「キー名を変更」をクリックし、キー名を「enqueteld」に変更します。同様に「入力」 - 「+string」をクリックし、入力値一覧に「string1」を追加します。

「string1」を選択した状態で入力値一覧上部の「キー名を変更」をクリックし、キー名を「answers」に変更します。追加した「answers」を選択した状態で「配列型にする」にチェックをいれます。

ここまでの設定が完了したら、入出力設定ダイアログの「決定」をクリックし、設定を終了します。

決定 取り消し

入力値はJSON入力で以下のコードを記載しても設定できます。

```
{
  "enqueteld": "",
  "answers": [
    ""
  ]
}
```

JSON入力する場合は、入力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

アンケート回答登録タスクでは、配列型データとして渡ってくる回答内容を「, (カンマ)」区切で1つの文字列として繋げて登録します。その変換処理に利用するための定数を設定します。

画面上部の「定数設定」をクリックして「定数設定」ダイアログを表示し、「+定数を追加」から下記の通り設定します。

ラベル 入力値

定数ID sep

定数値 ,

入力が完了したら「決定」をクリックして設定を終了します。

回答登録タスクの作成

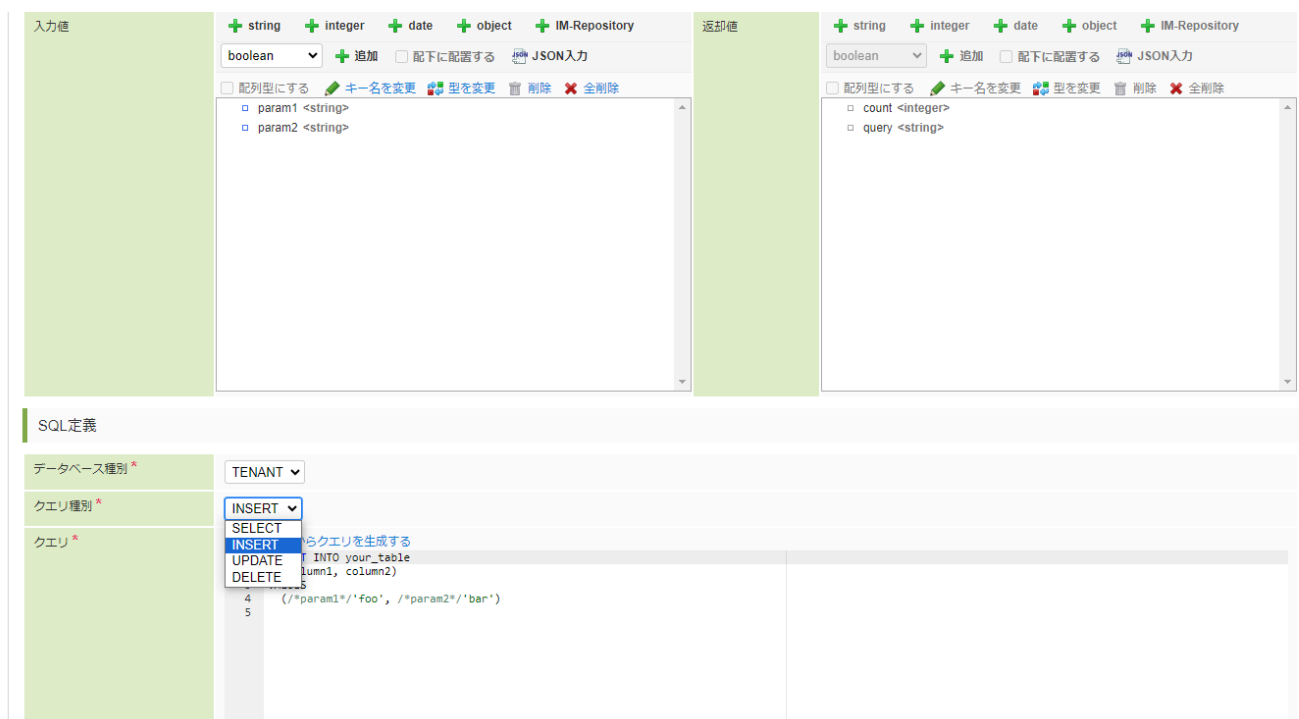
入力内容をもとに、アンケート回答テーブルにデータを登録するユーザ定義タスクを作成します。

パレットで「ユーザ定義追加」にカーソルを合わせ、「SQL定義新規作成」を選択します。

ユーザ定義共通設定を下記の通り設定します。

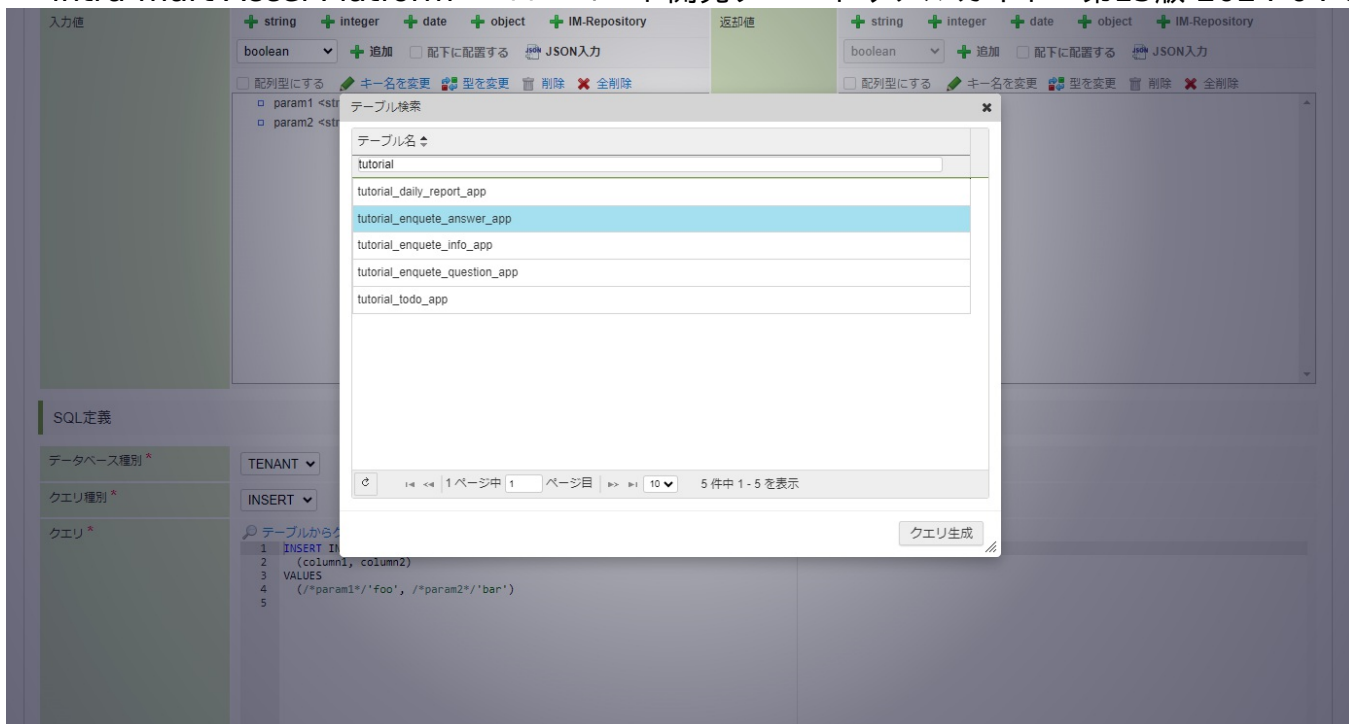
ラベル	入力値
ユーザ定義ID	tutorial_enquete_answer_insert
ユーザ定義名 (標準)	【チュートリアル】 アンケート回答登録
ユーザカテゴリ	【チュートリアル】 チュートリアルガイドカテゴリ

「SQL定義」のクエリ種別のプルダウンから「INSERT」を選択します。



「テーブルからクエリを生成する」をクリックし、テーブル検索ダイアログを表示します。

テーブル検索ダイアログで「tutorial_enquete_answer_app」を選択し、「クエリ生成」をクリックします。



SQL定義編集画面の入力値とクエリにそれぞれ以下の値が自動で設定されていることを確認してください。

- 入力値

項目	型	キー名
回答ID	string	answerId
アンケートID	string	enqueteId
回答者	string	respondent
回答日時	sqltimestamp	answerDate
回答	string	answers

- クエリ

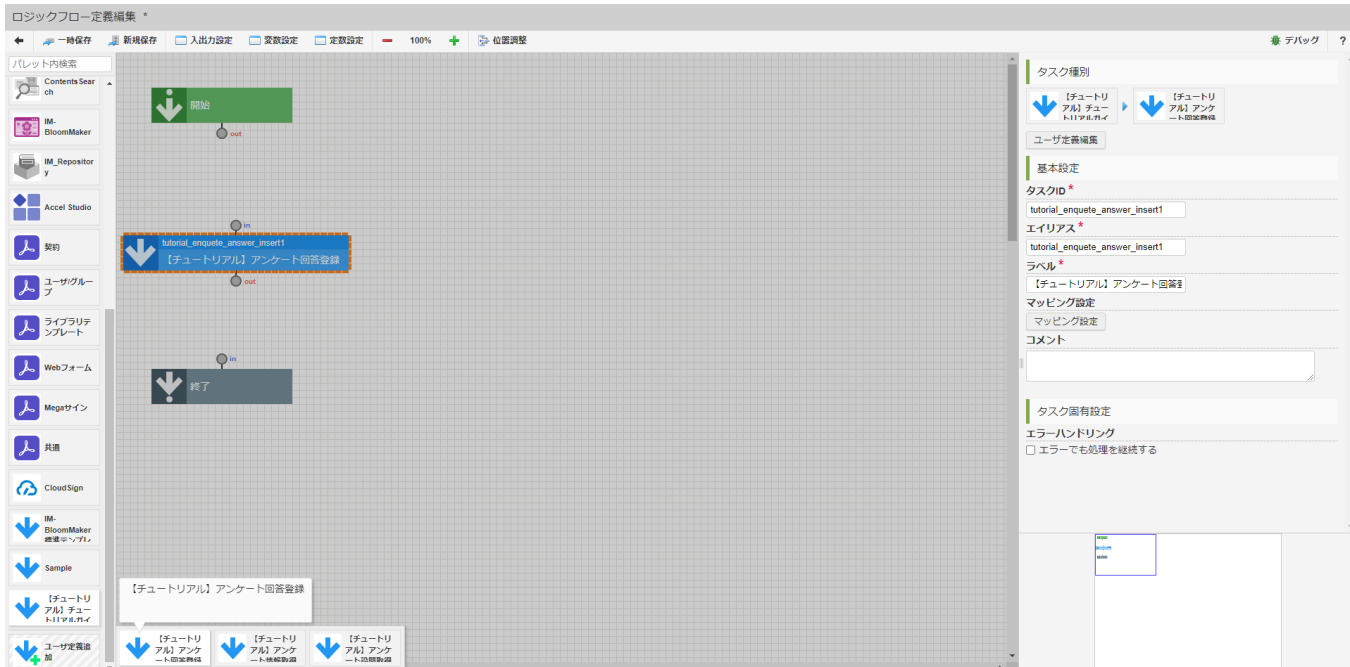
```
INSERT INTO tutorial_enquete_answer_app
(answer_id, enquete_id, respondent, answer_date, answers)
VALUES
(*answerId*/null, *enqueteId*/null, *respondent*/null, *answerDate*/null, *answers*/null)
```

内容を確認出来たら「登録」をクリックし、タスクを登録します。

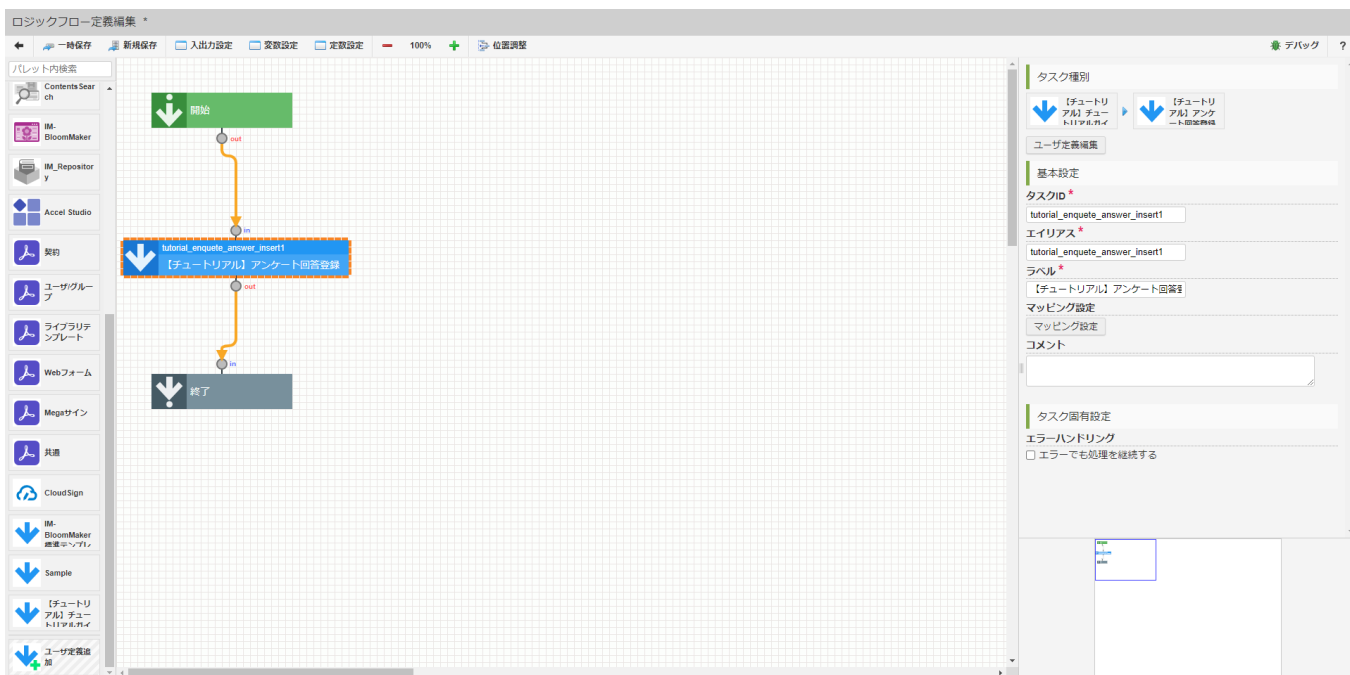
タスクの配置と変数設定

作成した「【チュートリアル】アンケート回答登録」タスクをフローに配置し、線を繋げます。

パレットで「【チュートリアル】チュートリアルガイドカテゴリ」にカーソルを合わせ、「【チュートリアル】アンケート回答登録」を選択します。

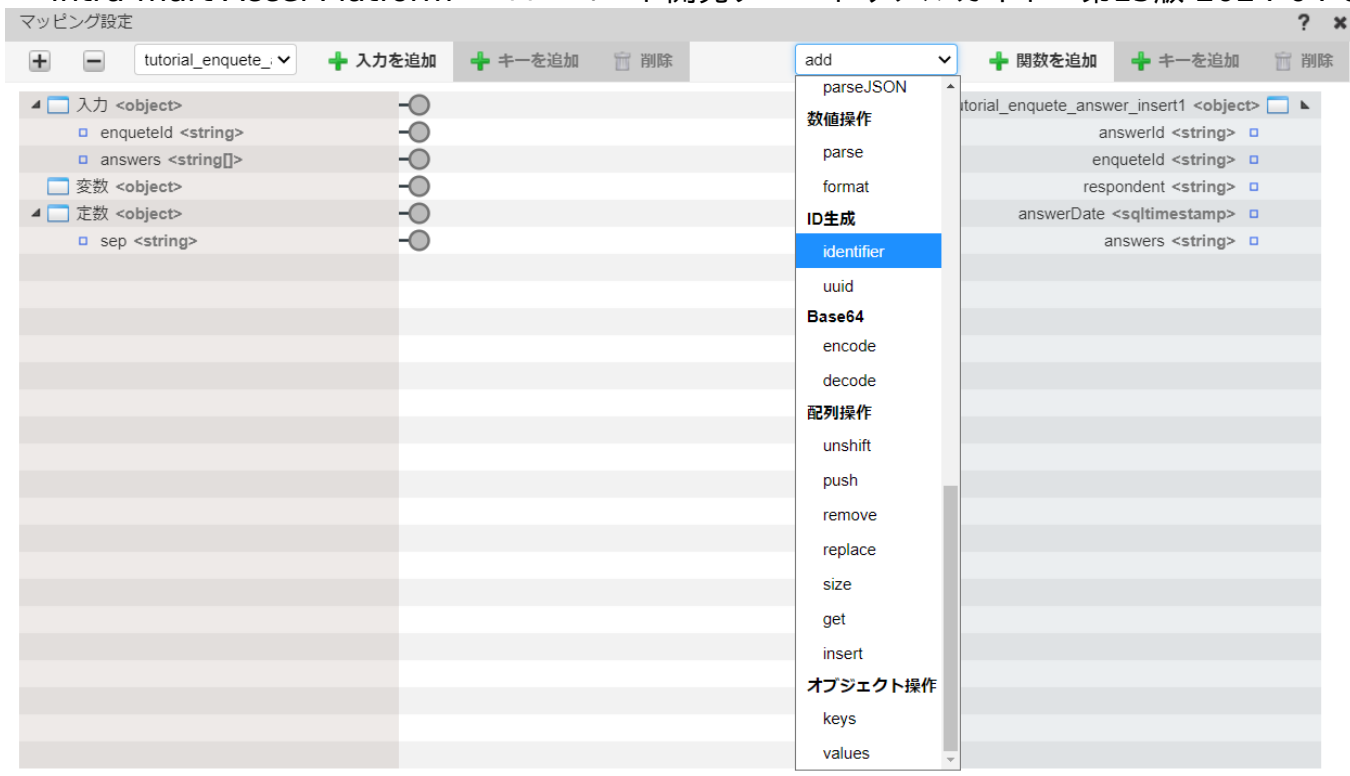


「【チュートリアル】アンケート回答登録」と「開始」、「終了」のin, outをそれぞれ線で繋げます。要素配置の順番等は以下の図を参考にしてください。



「【チュートリアル】アンケート回答登録」タスクをクリックし、画面の右側領域の「マッピング設定」をクリックしてマッピング設定ダイアログを表示します。

ダイアログ上部の関数追加プルダウンで「identifier」を選択し、「+関数を追加」をクリックして「identifier」関数を追加します。



デバッグ 決定 取り消し

追加した「identifier」の「out」と「【チュートリアル】アンケート回答登録」の入力値の「answerId」を線で繋がります。



デバッグ 決定 取り消し

入力値の「enqueteld」と「【チュートリアル】アンケート回答登録」の入力値の「enqueteld」を線で繋がります。

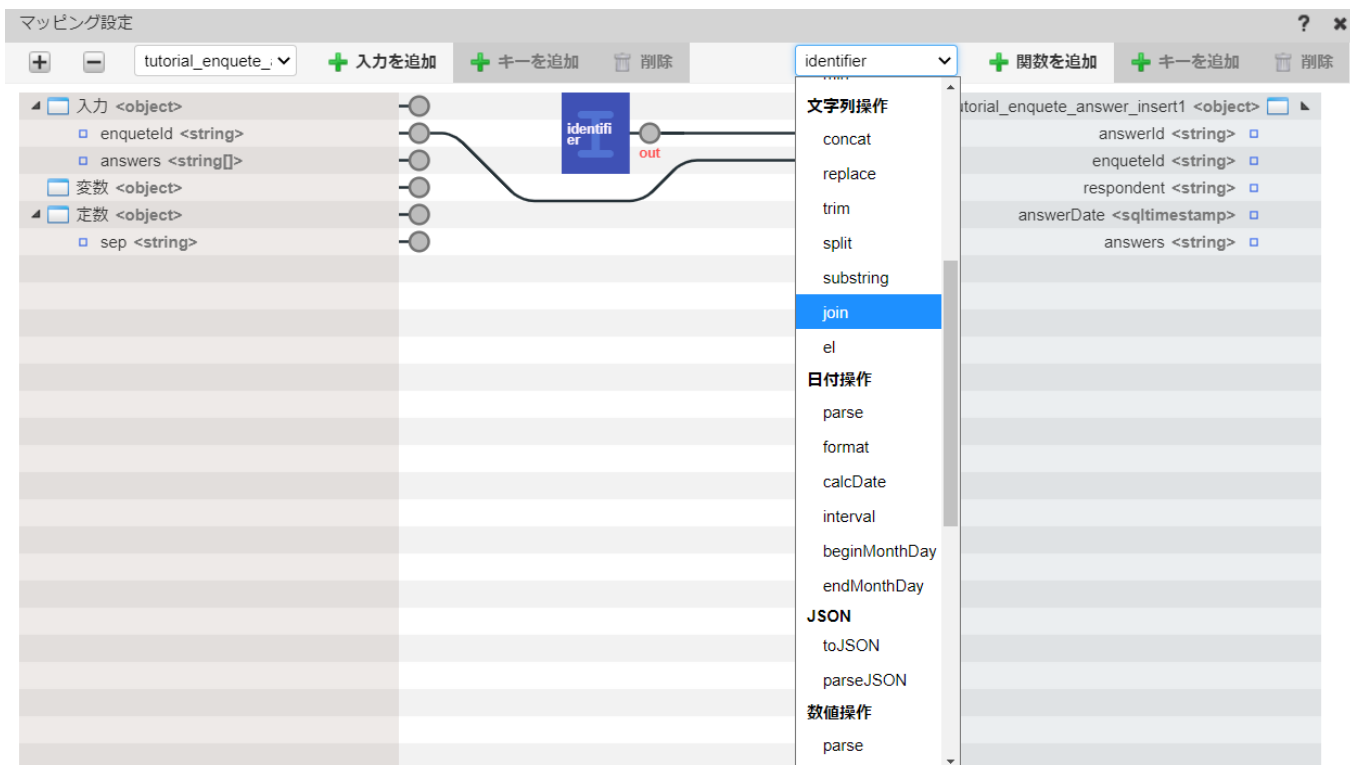


デバッグ 決定 取り消し

次に回答内容のデータについて設定していきます。

もとは文字列の配列型データである回答をデータベースに登録する際に1つの「文字列」へと変形し、「【チュートリアル】アンケート回答登録」の入力値として扱います。

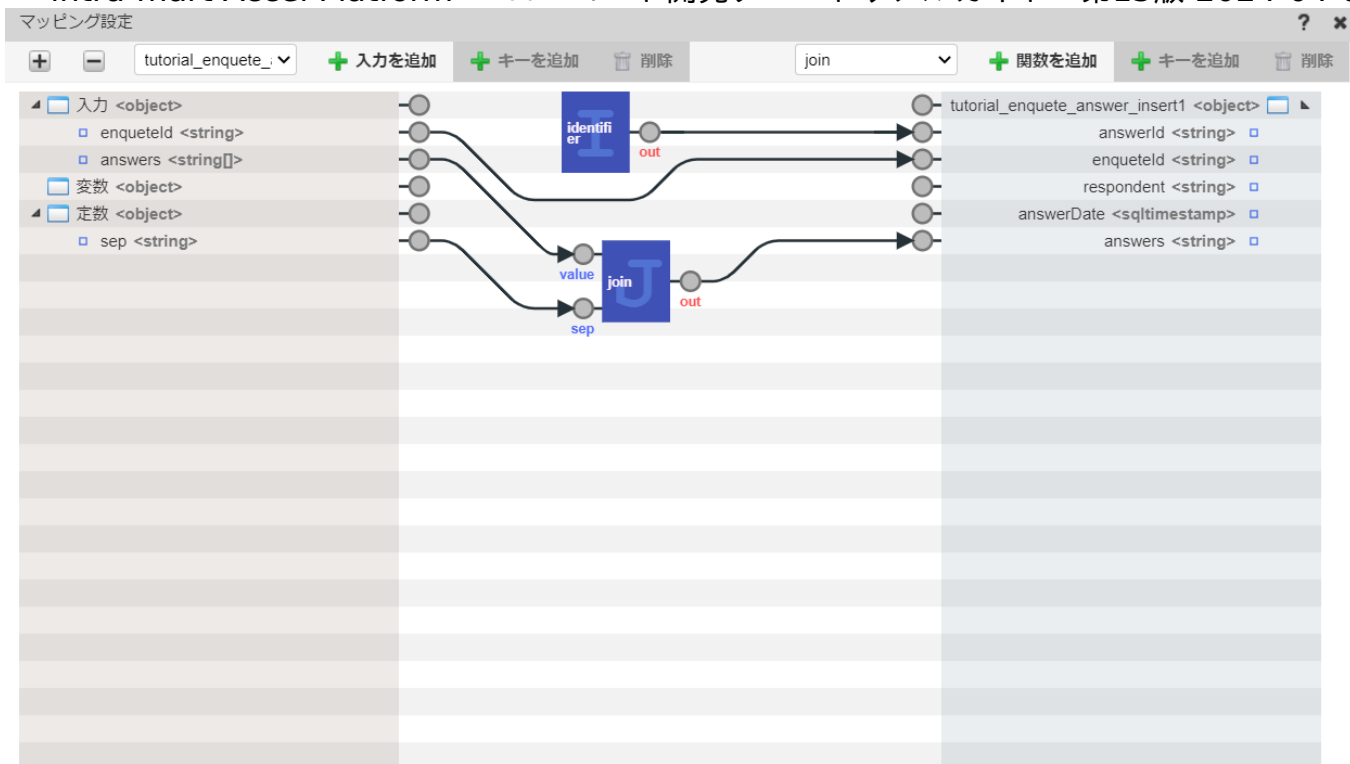
ダイアログ上部の関数追加プルダウンで「文字列操作 > join」を選択し、「+関数を追加」をクリックしてマッピング設定に「join」関数を追加します。



デバッグ 決定 取り消し

入力値の「answers」を「join」関数の「value」に、定数の「sep」を「join」関数の「sep」にそれぞれ繋がめます。

「join」関数のoutと「【チュートリアル】アンケート回答登録」のanswersを線で繋がめます。

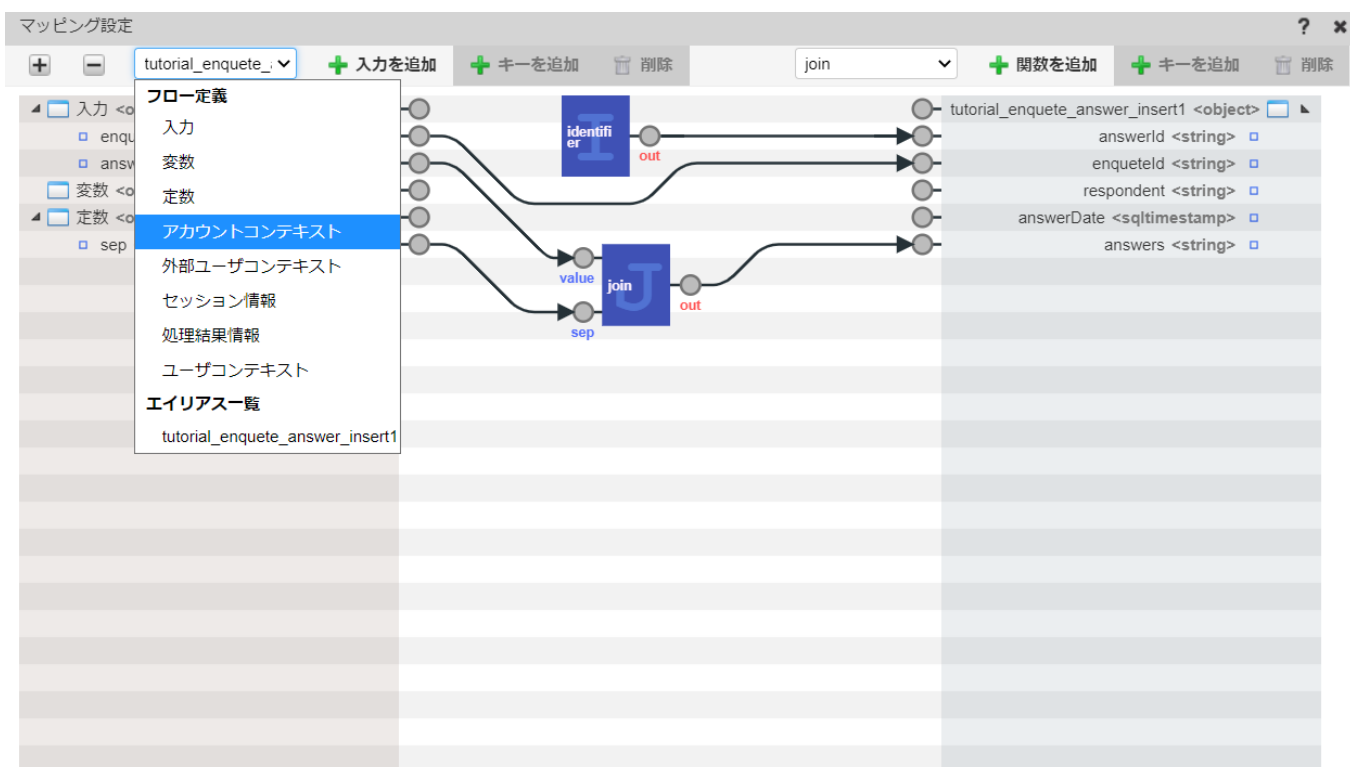


デバッグ 決定 取り消し

続けて、回答者と回答日を設定します。

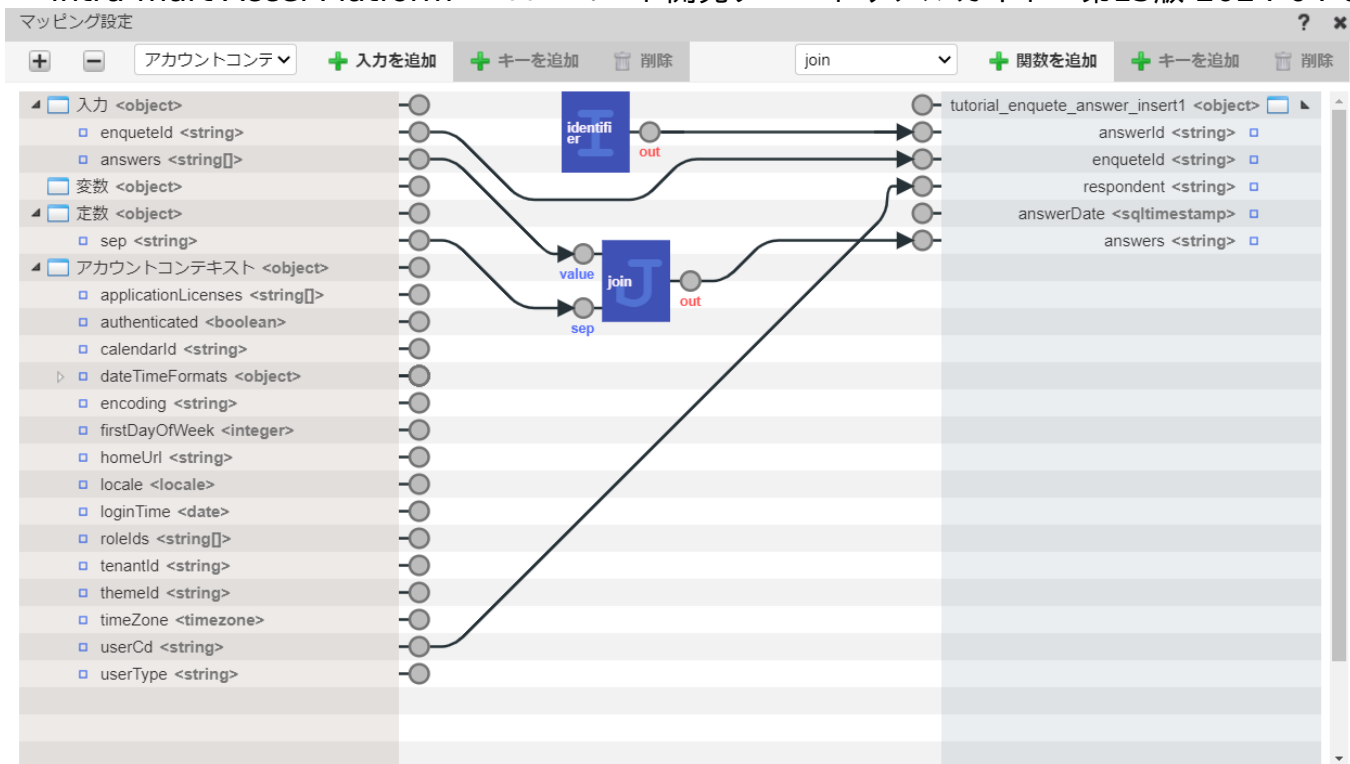
回答者はアカウントコンテキストのユーザコードを利用します。

「マッピング設定」ダイアログのプルダウンから「アカウントコンテキスト」を選択し、「+入力を追加」を選択します。



デバッグ 決定 取り消し

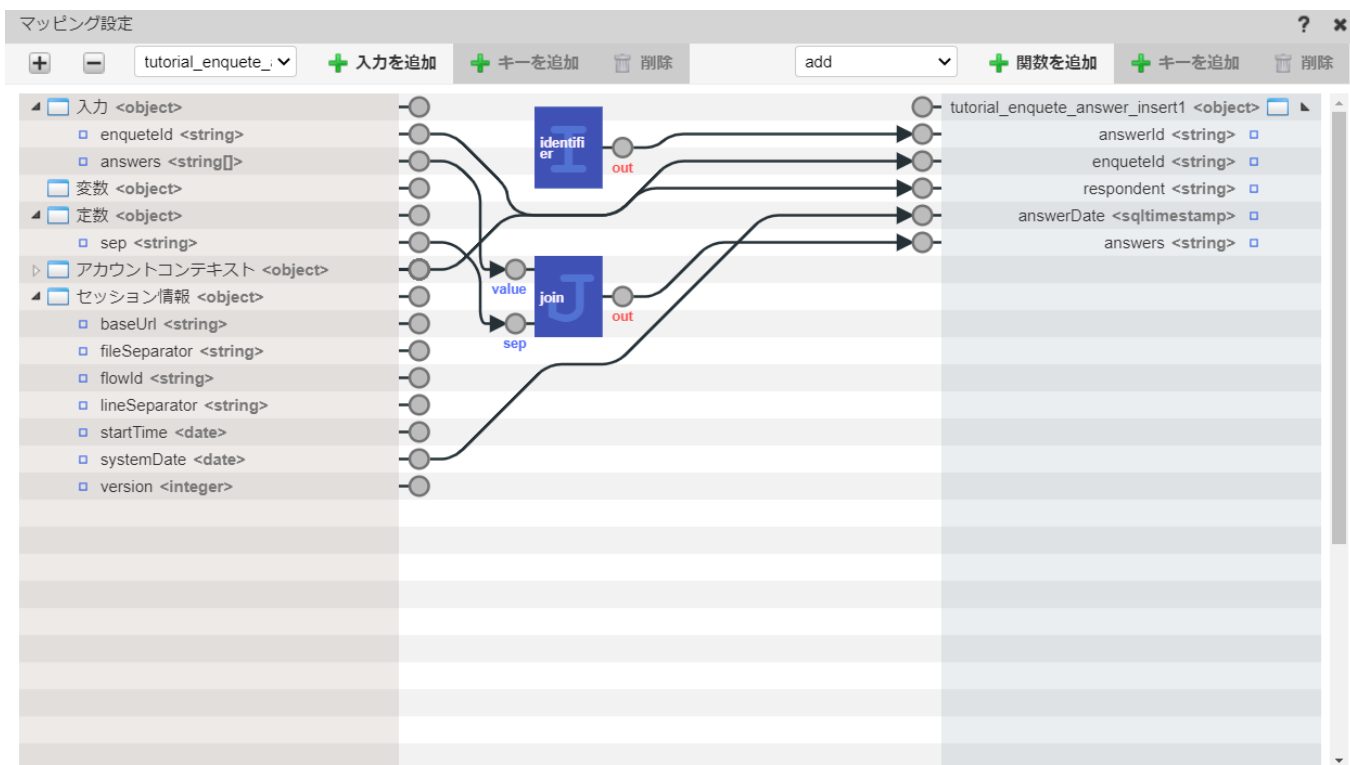
アカウントコンテキストの「userCd」と「【チュートリアル】アンケート回答登録」の「respondent」を線で繋げます。



デバッグ 決定 取り消し

回答日はセッション情報のシステム日時を利用します。アカウントコンテキストと同様の手順で「マッピング設定」ダイアログのプルダウンから「セッション情報」を選択し、「+入力を追加」を選択します。

セッション情報の「systemDate」と「【チュートリアル】アンケート回答登録」の「answerDate」を線で繋がめます。



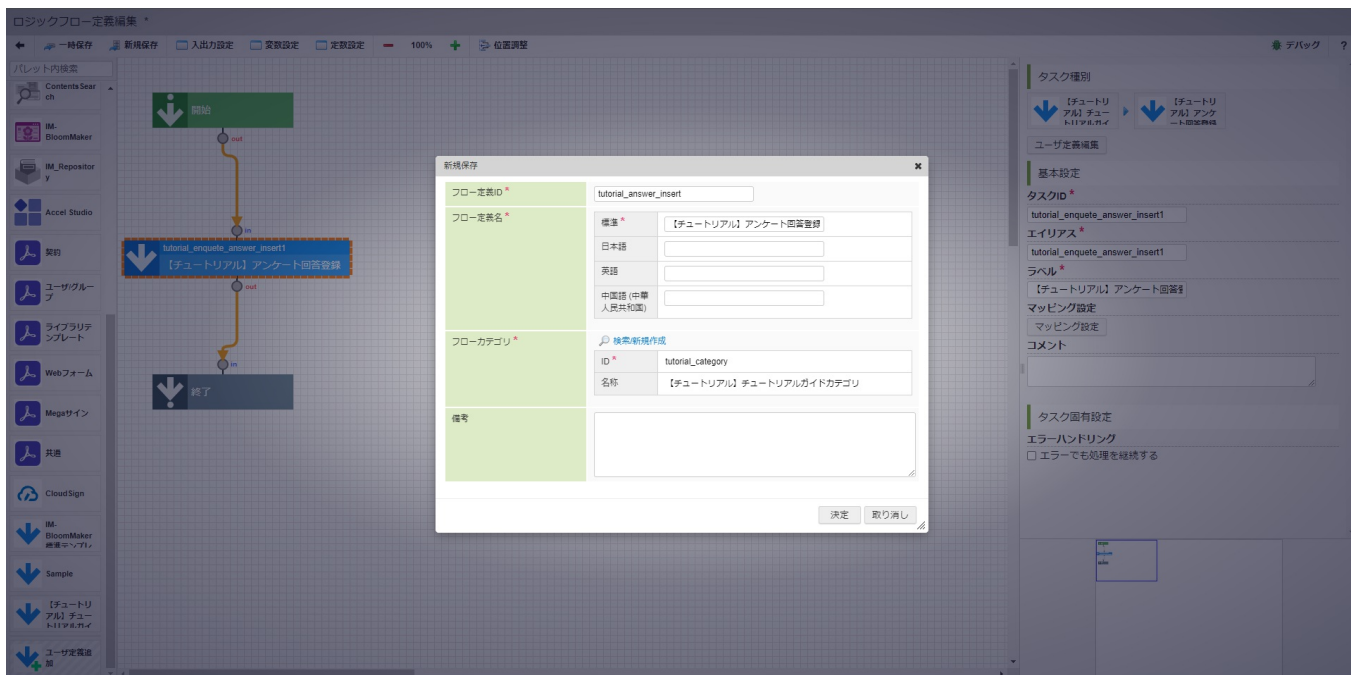
デバッグ 決定 取り消し

ここまでの設定ができれば「決定」をクリックし、マッピング設定を終了します。

これでアンケート回答登録処理の作成は終了です。

「ロジックフロー定義編集」画面上部の「新規保存」をクリックし、「新規保存」ダイアログの各項目を下記の通り入力してください。入力が完了したら「決定」をクリックして、ロジックを登録します。

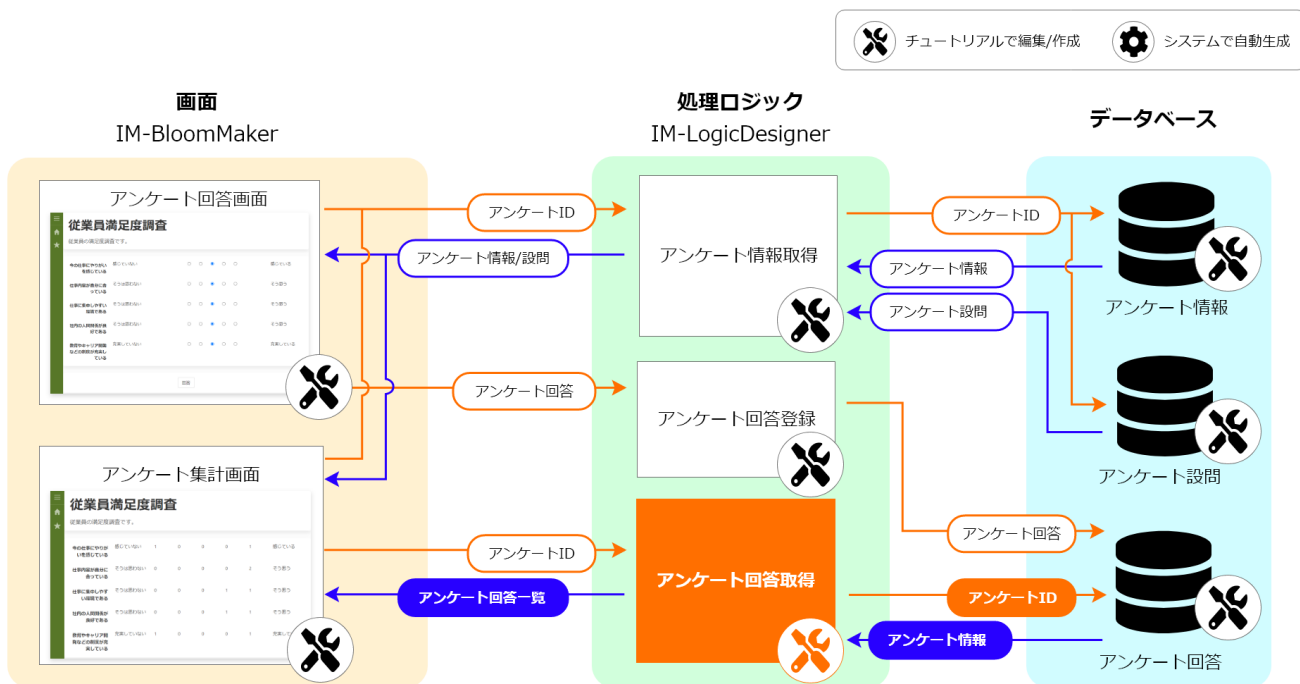
ラベル	入力値
フロー定義ID	tutorial_answer_insert
フロー定義名 (標準)	【チュートリアル】 アンケート回答登録
フローカテゴリ	【チュートリアル】 チュートリアルガイドカテゴリ



アンケート回答取得処理の作成

アンケート回答取得処理では、渡ってきたアンケートIDをもとに、アンケート回答テーブルからデータを取得します。データを取得できなかった場合はエラーを返します。

アプリケーションの全体図だと、以下の色が塗られた箇所が対象です。



この処理の作成手順について、以下の動画からも確認できます。併せてご利用ください。

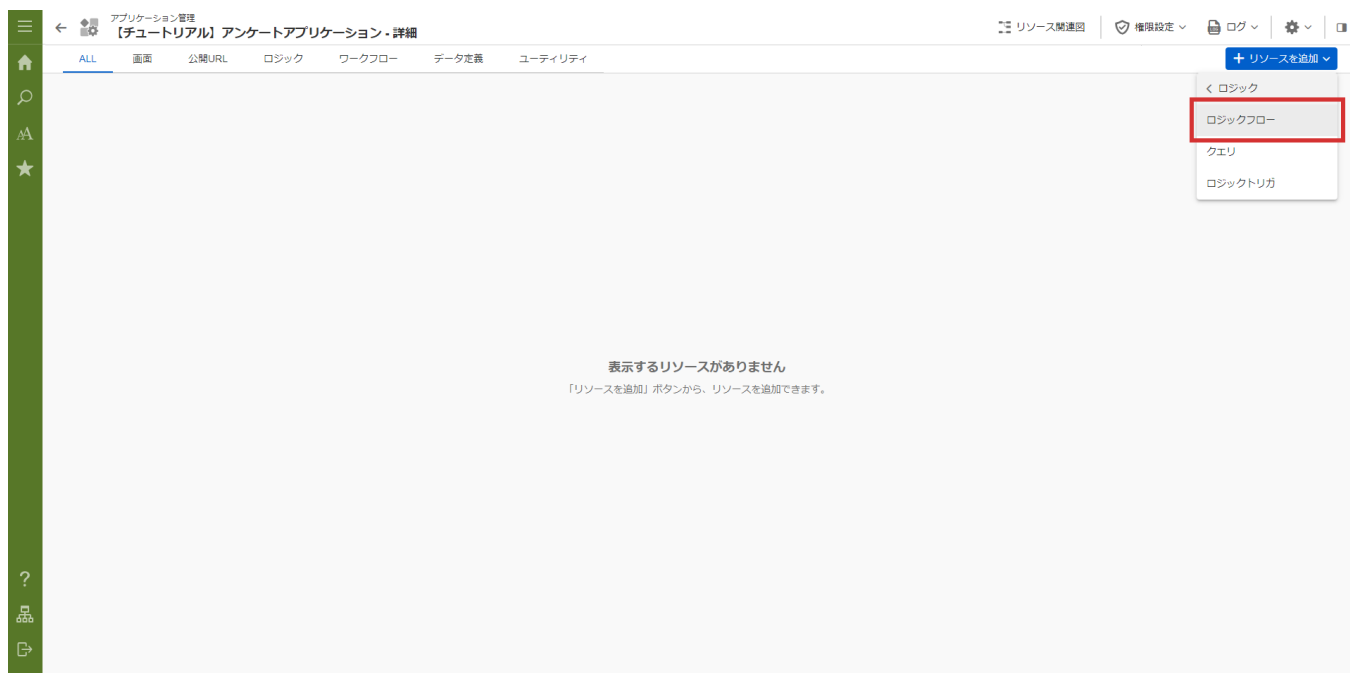
**注意**

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

ロジックの作成

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「ロジック」 - 「ロジックを新規作成」 - 「ロジックフロー」を選択し、「ロジックフロー定義編集画面」を表示します。

**入出力値の設定**

アンケート回答取得処理の入出力を設定します。

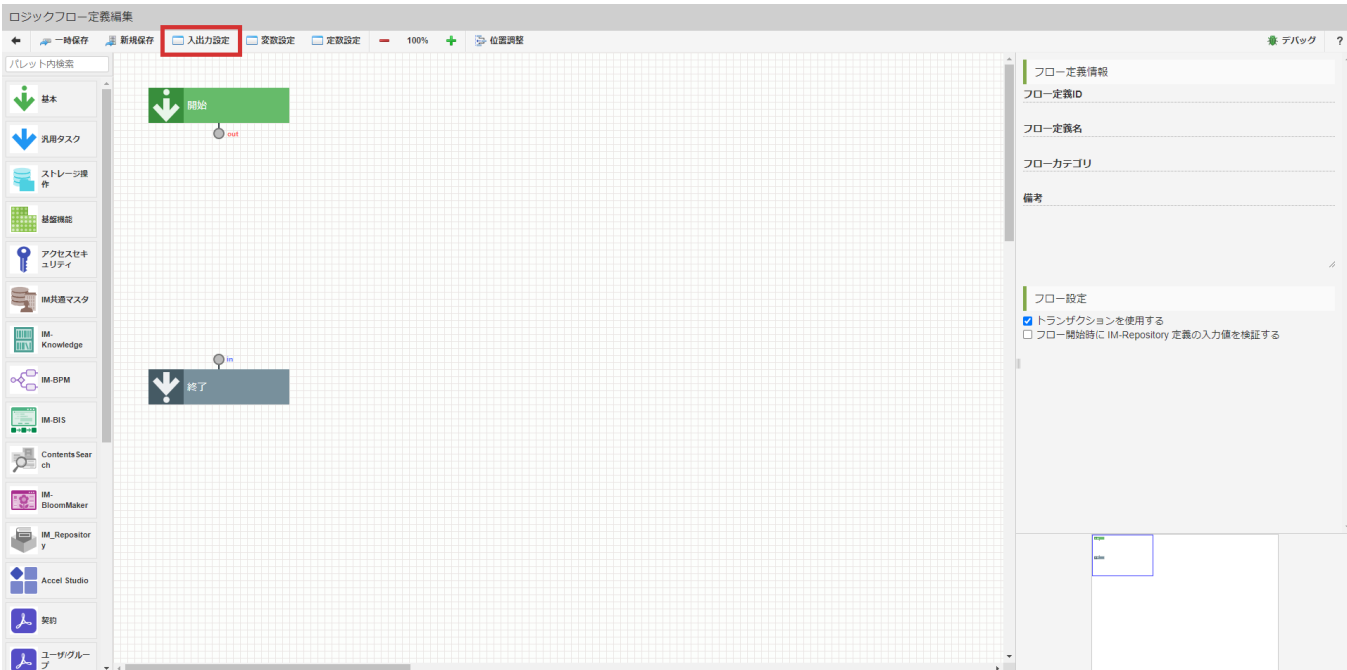
業務ロジックへの入力値と出力値を設定します。

アンケート回答取得処理は、アンケートIDを受け取り、アンケート回答を配列型データとして返却します。

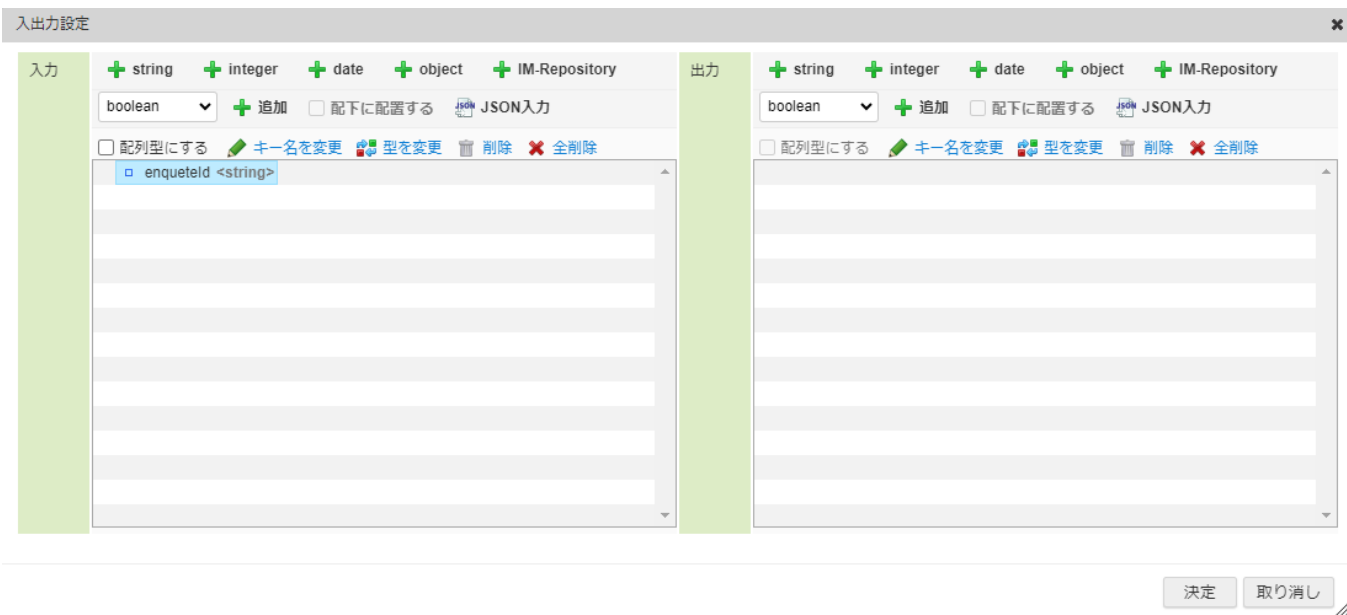
アンケートの回答は1ユーザの回答につき、配列の1データを格納します。また、回答内容についてもカンマ区切りで保持されている文字列の回答を配列に変換して格納します。

まずは、入力値の設定を行います。

画面上部の「入出力設定」をクリックし「入出力設定」ダイアログを表示します。



入力値は「アンケートID」のみとするため、文字列の変数を追加します。
 「入力」 - 「+string」をクリックし、入力値一覧に「string1」を追加します。
 「string1」を選択した状態で入力値一覧上部の「キー名を変更」をクリックし、キー名を「enqueteld」に変更します。

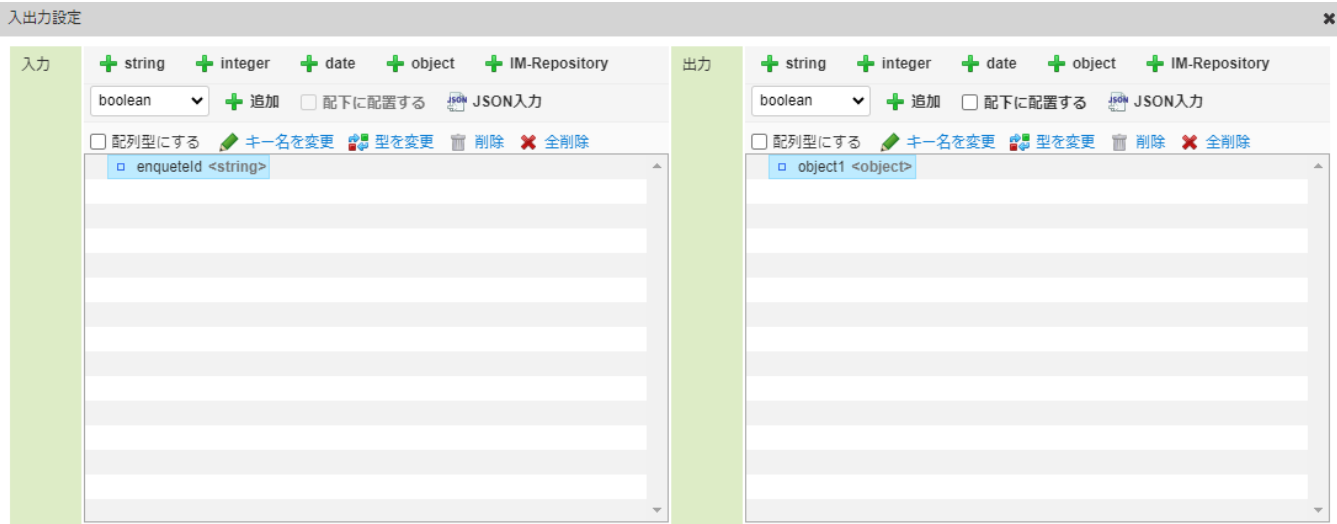


入力値はJSON入力で以下のコードを記載しても設定できます。

```
{
  "enqueteld": ""
}
```

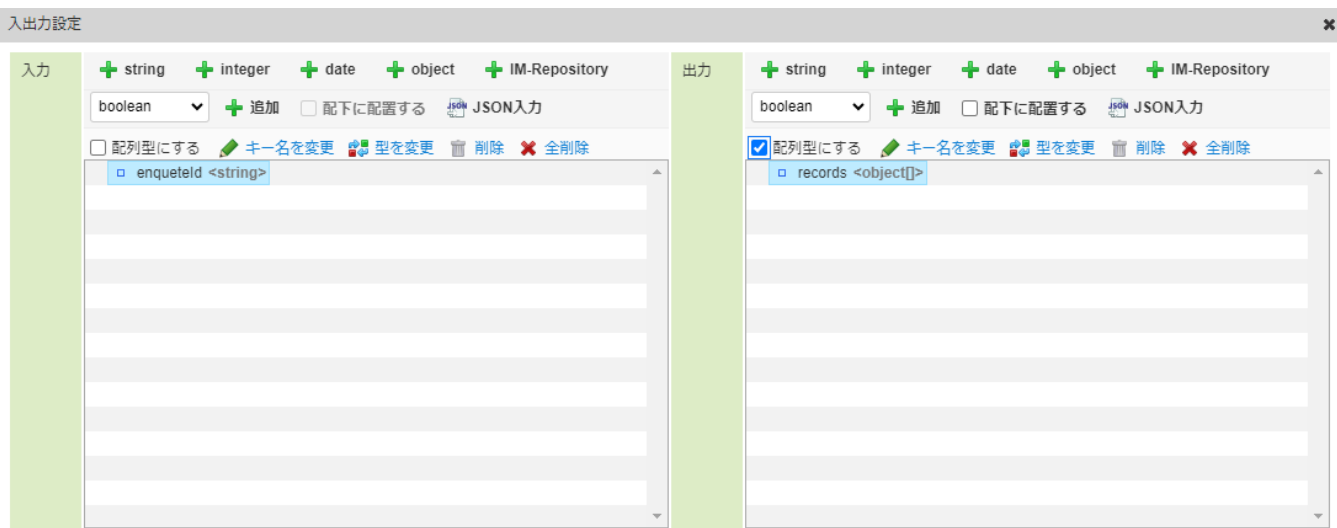
JSON入力する場合は、入力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

次に出力値の設定を行います。
 まずは、設問内容を格納する配列を作成します。
 「出力」 - 「+object」をクリックすると出力値一覧に「object1」が追加されます。



決定 取り消し

「object1」を選択した状態で出力値一覧上部の「キー名を変更」をクリックし、キー名を「records」に変更します。
配置した「records」を選択した状態で「配列型にする」にチェックをいれ、「records」を汎用オブジェクトの配列にします。



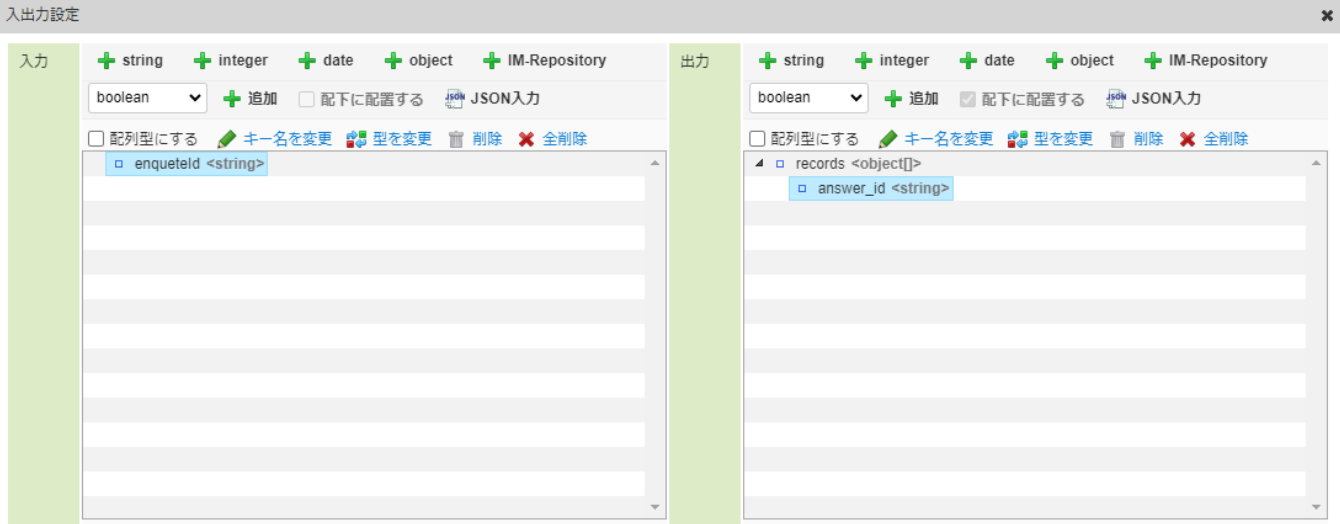
決定 取り消し

次に、設問内容の値を格納していくための出力値を「records」配下に追加します。

出力値一覧の「records」を選択した状態で「配下に配置する」を選択します。

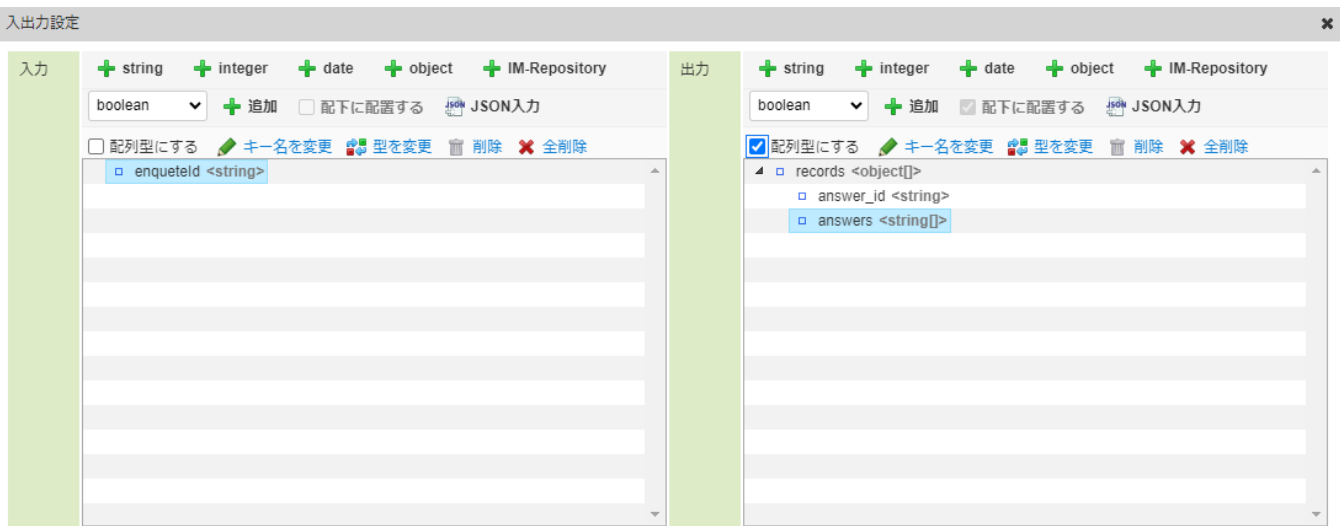
「出力」 - 「+string」をクリックし、出力値一覧の「records」配下に、「string1」を追加します。

「string1」を選択した状態で出力値一覧上部の「キー名を変更」をクリックし、キー名を「answer_id」に変更します。



決定 取り消し

次に、回答を配列型データとして格納する出力値を設定します。
 「出力」 - 「+string」をクリックし、出力値一覧の「records」配下に、「string1」を追加します。
 「string1」を選択した状態で出力値一覧上部の「キー名を変更」をクリックし、キー名を「answers」に変更します。
 「answers」を選択した状態で出力値一覧上部の「配列型にする」にチェックを入れます。



決定 取り消し

同様の手順で「records」配下に下記の設定の変数を追加します。

項目	型	キー名
回答ID	string	answer_id
回答	string[]	answers
アンケートID	string	enquete_id
回答者	string	respondent
回答日	date	answer_date

決定 取り消し

出力値も、JSON入力下記コードを記述して設定できます。

```
{
  "records": [
    {
      "answer_id": "",
      "enquete_id": "",
      "respondent": "",
      "answer_date": null,
      "answers": [
        ""
      ]
    }
  ]
}
```

JSON入力する場合は、出力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

全ての入出力設定が完了したら、入出力設定ダイアログの「決定」をクリックし、設定を終了します。

アンケート回答取得タスクでは、「, (カンマ)」区切で1つの文字列として取得した回答を配列型データに変換して返却します。その変換処理に利用するための定数を設定します。

画面上部の「定数設定」をクリックして「定数設定」ダイアログを表示し、「+定数を追加」から下記の通り設定します。

ラベル 入力値

定数ID sep

定数値 ,

入力が完了したら「決定」をクリックして設定を終了します。

アンケート回答取得タスクの作成

アンケートIDをもとに、アンケート回答テーブルからデータを取得するユーザ定義タスクを作成します。

パレットで「ユーザ定義追加」にカーソルを合わせ、「SQL定義新規作成」を選択します。

ユーザ定義共通設定を下記の通り設定します。

ラベル	入力値
ユーザ定義ID	tutorial_enquete_answer_select
ユーザ定義名 (標準)	【チュートリアル】アンケート回答取得
ユーザカテゴリ	【チュートリアル】チュートリアルガイドカテゴリ

項目	型	キー名
アンケートID	string	enqueteId

クエリに以下のSQLを設定します。

```

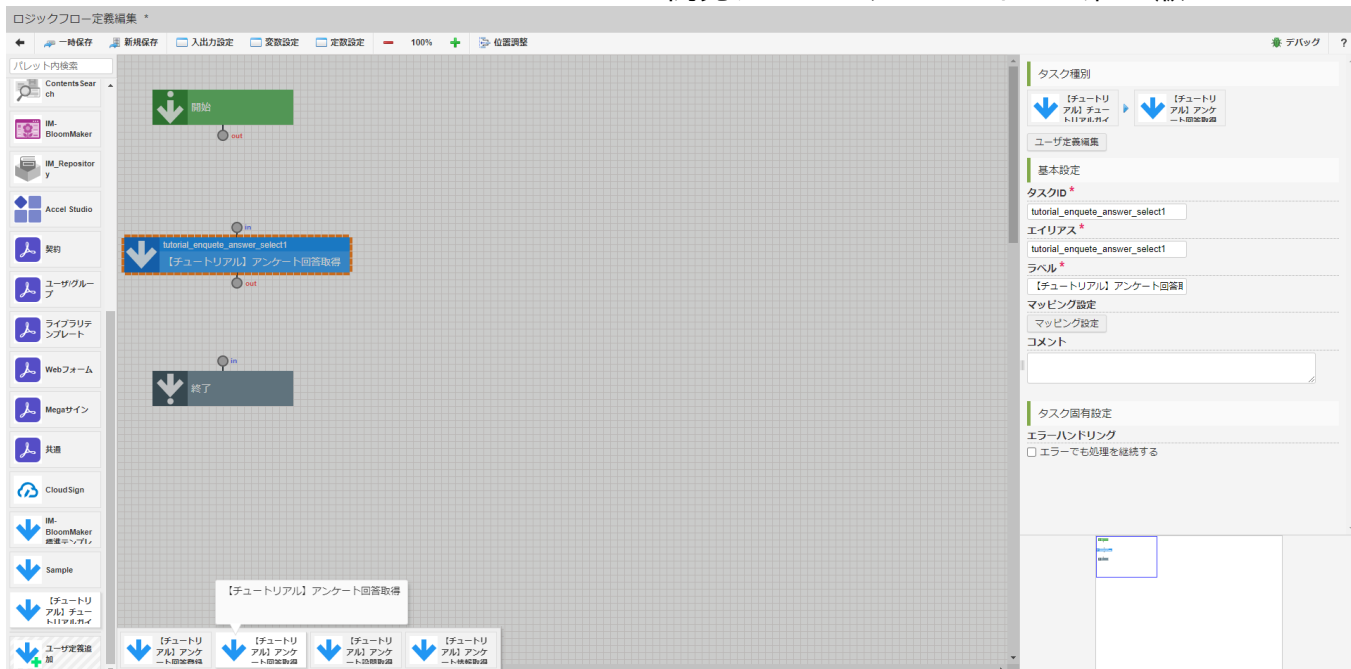
SELECT
*
FROM
tutorial_enquete_answer_app
WHERE
enquete_id = /*enqueteId*/'dummyId'
    
```

クエリ設定上部にある「データ定義を取得する」をクリックし、タスクの出力値を設定します。

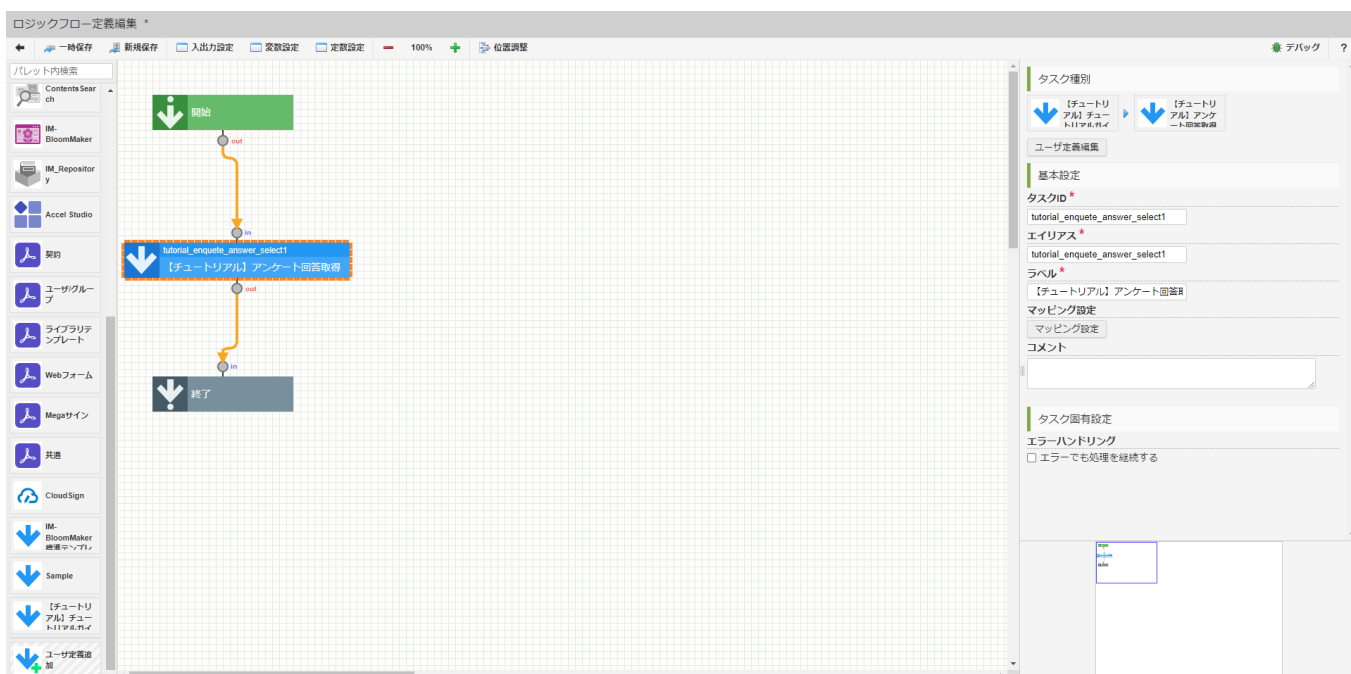
「登録」をクリックし、タスクを登録します。

タスクの配置と変数設定

作成した「【チュートリアル】アンケート回答取得」タスクをフローに配置し、線を繋げます。
 パレットで「【チュートリアル】チュートリアルガイドカテゴリ」 - 「【チュートリアル】アンケート回答取得」を選択します。

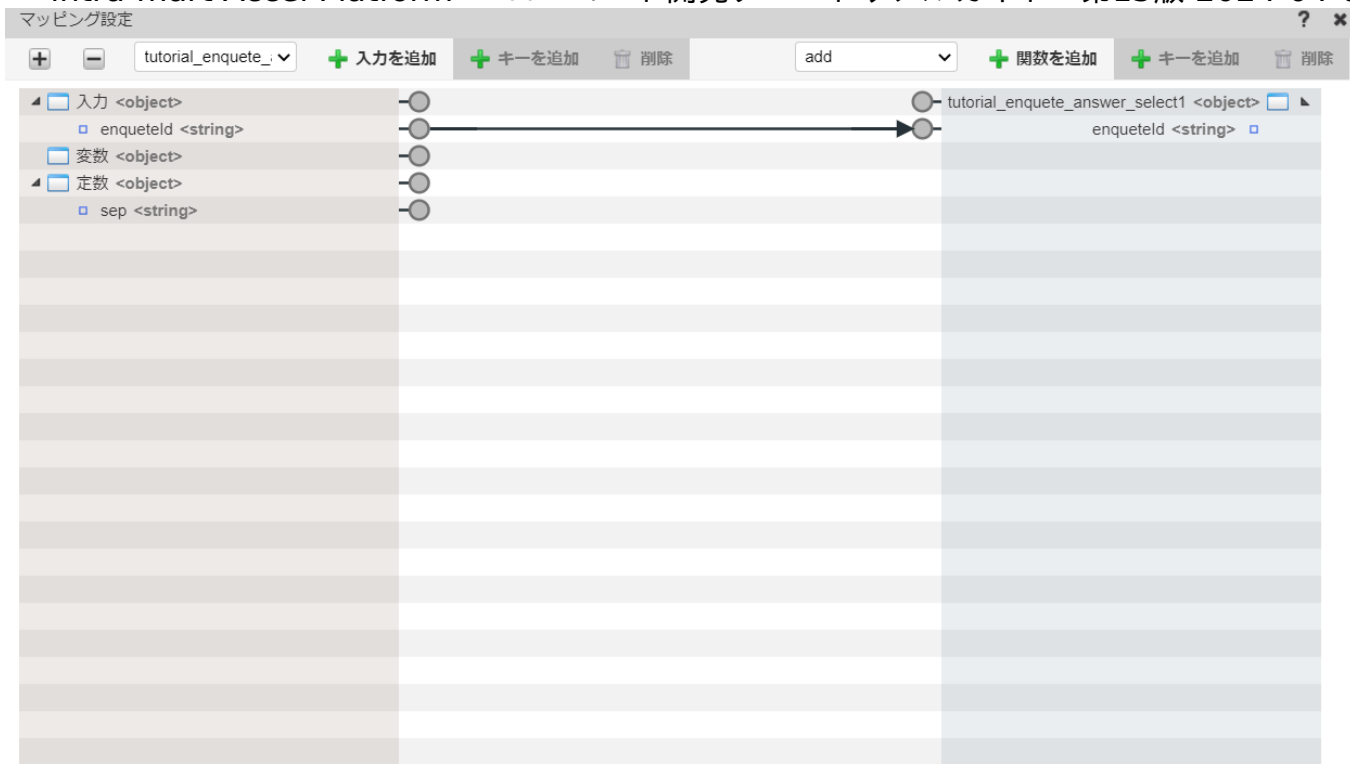


配置した「【チュートリアル】アンケート回答取得」と「開始」、「終了」のin、outをそれぞれ線で繋げます。



配置した「【チュートリアル】アンケート回答取得」タスクをクリックし、画面の右側領域の「マッピング設定」をクリックしてマッピング設定ダイアログを表示します。

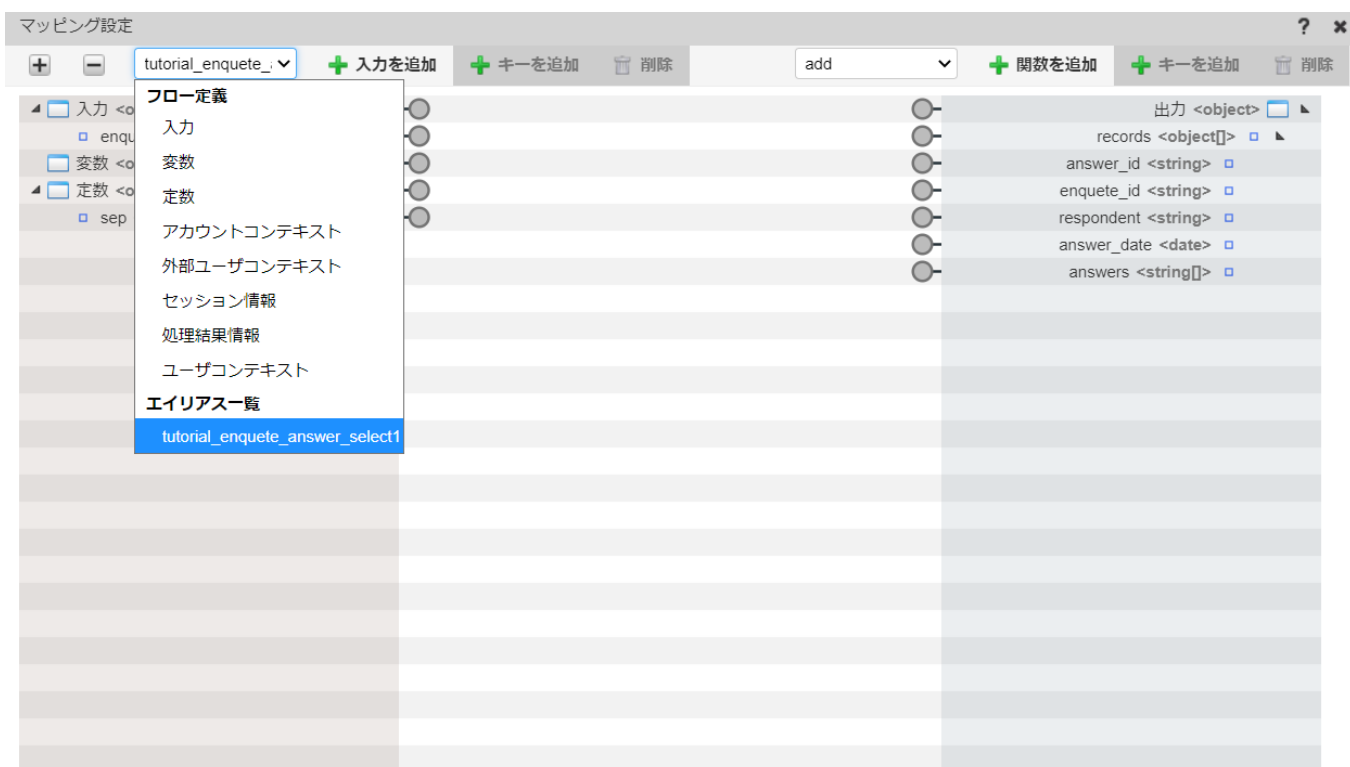
入力値の「enqueteld」と「【チュートリアル】アンケート回答取得」の入力値の「enqueteld」を線で繋げます。設定が完了したら「決定」をクリックし、マッピング設定を終了します。



デバッグ 決定 取り消し

続いて、返却値の設定を行います。「終了」をクリックし、画面の右側領域の「マッピング設定」をクリックしてマッピング設定ダイアログを表示します。

マッピング設定の上部で「tutorial_enquete_answer_select1」を選択し、「+入力を追加」をクリックします。



デバッグ 決定 取り消し

「tutorial_enquete_answer_select1」の「records」と出力の「records」を線で繋がめます。

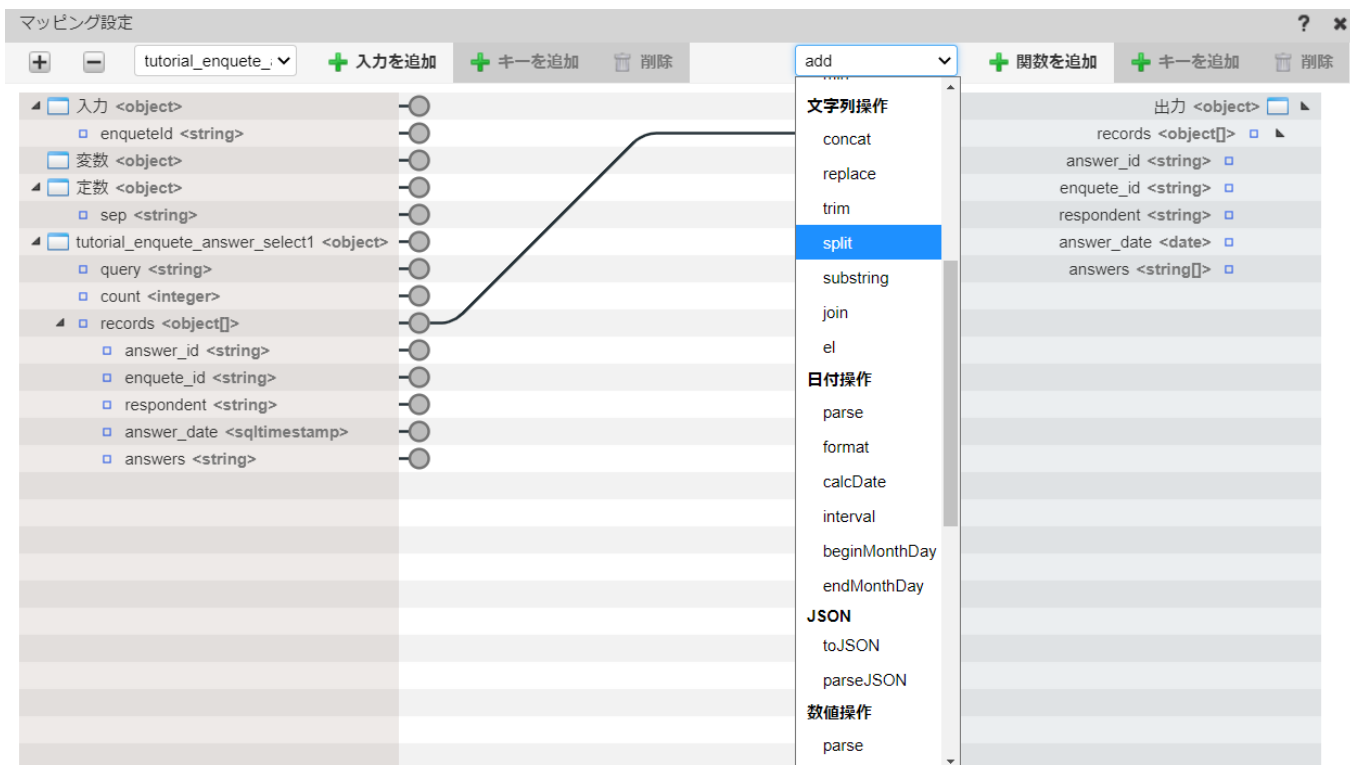


デバッグ 決定 取り消し

次に、回答を線で繋がめます。

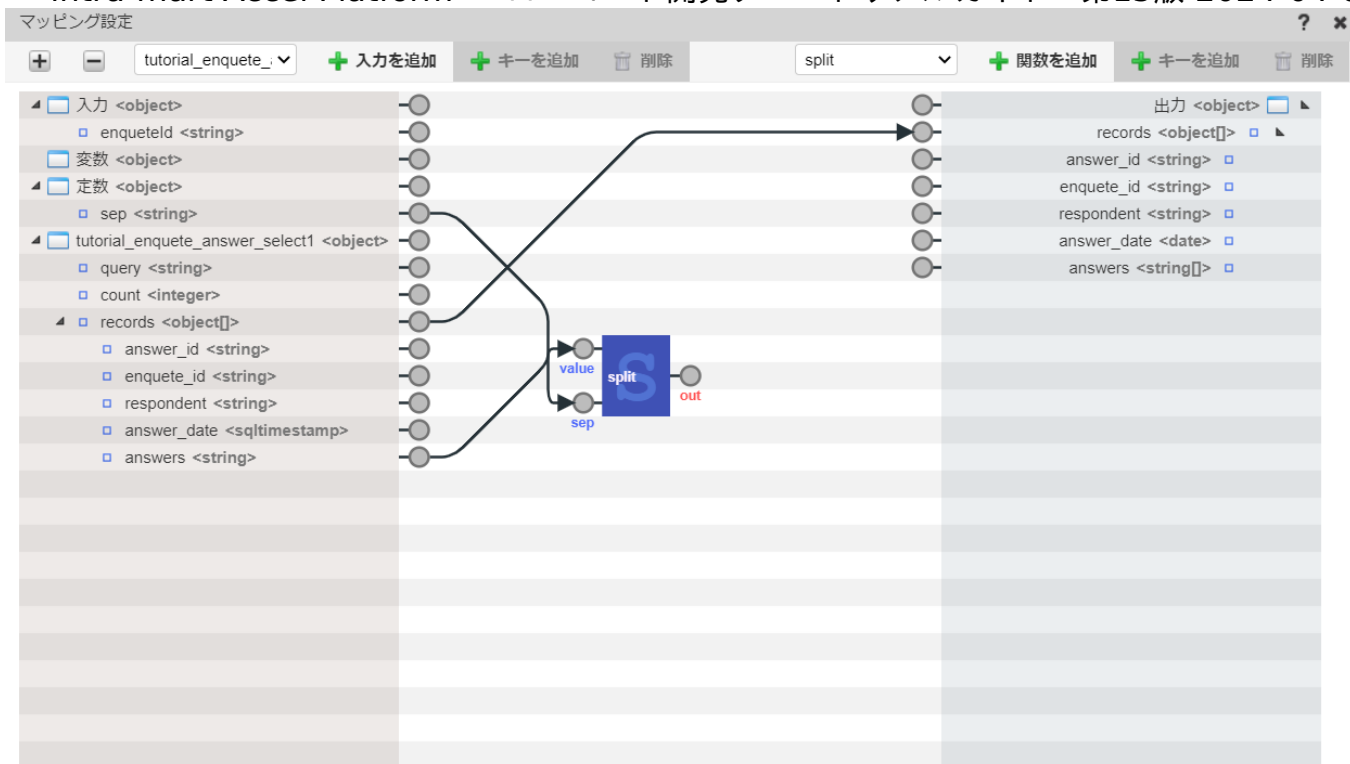
回答は「tutorial_enquete_answer_select1」でカンマ区切りの文字列データとして渡ってきますが、タスクの出力値としては配列型データとして返却するため、データの変換も行います。

画面上部の関数追加プルダウンで「文字列操作 > split」を選択し、「+関数を追加」をクリックしてマッピング設定に「split」関数を追加します。



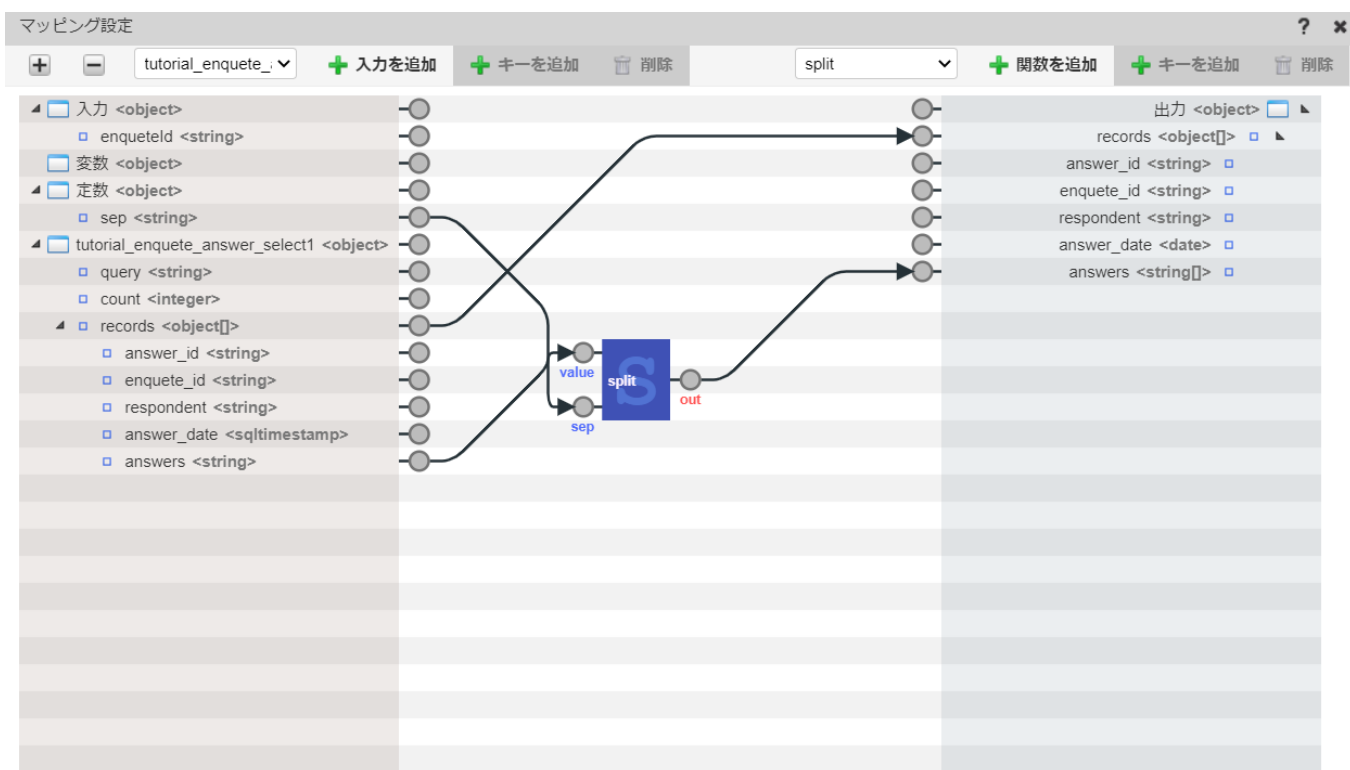
デバッグ 決定 取り消し

入力値の「answers」を「split」関数の「value」に、定数の「sep」を「split」関数の「sep」にそれぞれ繋がめます。



デバッグ 決定 取り消し

「split」関数のoutと「出力値」のanswersを線で繋がめます。



デバッグ 決定 取り消し

「決定」をクリックし、マッピング設定を終了します。

これでアンケート回答取得処理の作成は終了です。

「ロジックフロー定義編集」画面上部の「新規保存」をクリックし、「新規保存」ダイアログの各項目を下記の通り入力してください。入力が完了したら「決定」をクリックして、ロジックを登録します。

ラベル	入力値
フロー定義ID	tutorial_answers_get
フロー定義名（標準）	【チュートリアル】 アンケート回答取得
フローカテゴリ	【チュートリアル】 チュートリアルガイドカテゴリ

新規保存 ✕

フロー定義ID *	<input type="text" value="tutorial_answers_get"/>								
フロー定義名 *	<table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; padding: 2px;">標準 *</td> <td style="padding: 2px;"><input type="text" value="【チュートリアル】 アンケート回答取得"/></td> </tr> <tr> <td style="padding: 2px;">英語</td> <td style="padding: 2px;"><input type="text"/></td> </tr> <tr> <td style="padding: 2px;">日本語</td> <td style="padding: 2px;"><input type="text"/></td> </tr> <tr> <td style="padding: 2px;">中国語 (中華人民共和国)</td> <td style="padding: 2px;"><input type="text"/></td> </tr> </table>	標準 *	<input type="text" value="【チュートリアル】 アンケート回答取得"/>	英語	<input type="text"/>	日本語	<input type="text"/>	中国語 (中華人民共和国)	<input type="text"/>
標準 *	<input type="text" value="【チュートリアル】 アンケート回答取得"/>								
英語	<input type="text"/>								
日本語	<input type="text"/>								
中国語 (中華人民共和国)	<input type="text"/>								
フローカテゴリ *	<div style="margin-bottom: 5px;">🔍 検索/新規作成</div> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%; padding: 2px;">ID *</td> <td style="padding: 2px;"><input type="text" value="tutorial_category"/></td> </tr> <tr> <td style="padding: 2px;">名称</td> <td style="padding: 2px;"><input type="text" value="【チュートリアル】 チュートリアルカテゴリ"/></td> </tr> </table>	ID *	<input type="text" value="tutorial_category"/>	名称	<input type="text" value="【チュートリアル】 チュートリアルカテゴリ"/>				
ID *	<input type="text" value="tutorial_category"/>								
名称	<input type="text" value="【チュートリアル】 チュートリアルカテゴリ"/>								
備考	<div style="border: 1px solid gray; height: 40px;"></div>								

業務ロジックのURL設定

作成した業務ロジックを外部から利用できるようにするため、URLの設定を行います。

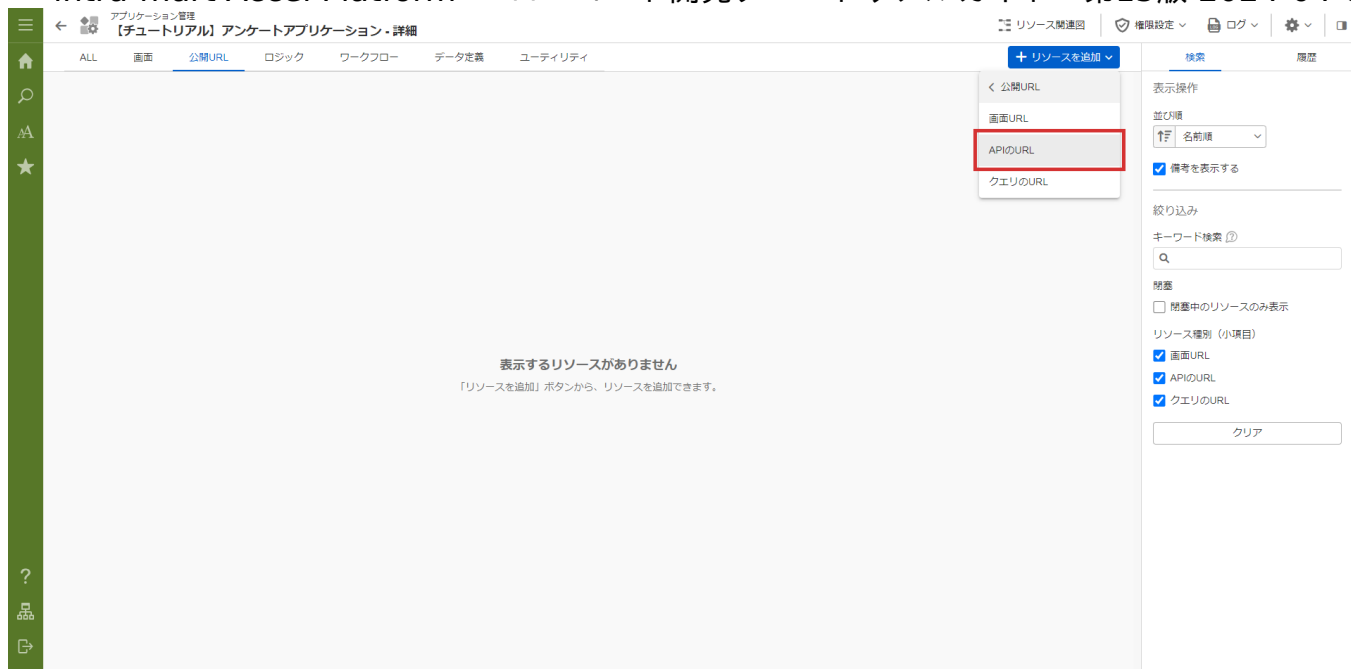
アンケート情報取得処理のURL設定

アンケート情報取得処理のURL設定を行います。

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】 アンケートアプリケーション」の順に遷移します。

「【チュートリアル】 アンケートアプリケーション」のアプリケーション管理画面で「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「APIのURL」を選択し、「ロジックフロールーティング定義編集」画面を表示します。



「ロジックフロールーティング定義編集」画面の「対象ロジックフロー定義情報」 - 「対象フロー」 - 「検索」からロジックフロー定義検索ダイアログを表示します。

ダイアログ上で「【チュートリアル】 アンケート情報取得」を選択し、「決定」をクリックします。



「ロジックフロールーティング定義情報」の「ルーティング」を「tutorial_enquete/question」と設定します。

「ロジックフロールーティング定義情報」の「認可URI」を「tutorial_enquete/question」と設定します。

その他の設定は標準のまま、「登録」をクリックしてURLを設定します。

ロジックフロールーティング定義編集

対象ロジックフロー定義情報

対象フロー *

検索

フロー定義ID * tutorial_enquete_info

フロー定義名 【チュートリアル】アンケート情報取得

バージョン番号 *

最新バージョンを利用する
 利用するバージョンを指定する

利用バージョン *

ロジックフロールーティング定義情報

ルーティング * /mart/logic/api/tutorial_enquete/question

メソッド * GET

認証方法 * IMAuthentication

認可URI * im-logic-rest/api/tutorial_enquete/question

セキュアトークンを利用する

レスポンス種別 * JSONに変換して返却

レスポンスヘッダ

ヘッダ名 *	ヘッダ値 *	削除
<input type="text"/>	<input type="text"/>	✖

登録

アンケート回答登録処理のURL設定

アンケート情報取得処理のURL設定を行います。

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」の順に遷移します。

「【チュートリアル】アンケートアプリケーション」のアプリケーション管理画面に遷移し、「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「APIのURL」を選択し、「ロジックフロールーティング定義編集」画面を開きます。

アプリケーション管理
【チュートリアル】アンケートアプリケーション - 詳細

リソース開連図 権限設定 ログ 設定

ALL 画面 公開URL ロジック ワークフロー データ定義 ユーティリティ

+ リソースを追加

公開URL
画面URL
APIのURL
クエリのURL

表示するリソースがありません
「リソースを追加」ボタンから、リソースを追加できます。

検索 履歴

表示操作
並び順
↑ ↓ 名前順

備考を表示する

絞り込み
キーワード検索
Q

閉塞
 閉塞中のリソースのみ表示

リソース種別 (小項目)
 画面URL
 APIのURL
 クエリのURL

クリア

「ロジックフロールーティング定義編集」画面の「対象ロジックフロー定義情報」 - 「対象フロー」 - 「検索」からロジックフロー定義検索ダイアログを表示します。

ダイアログ上で「【チュートリアル】アンケート回答登録」を選択し、「決定」をクリックします。

ロジックフロー定義検索

【チュートリアル】

選択	フロー定義ID	フロー定義名	プレビュー
<input checked="" type="checkbox"/>	tutorial_answer_insert	【チュートリアル】 アンケート回答登録	
<input type="checkbox"/>	tutorial_answers_get	【チュートリアル】 アンケート回答取得	
<input type="checkbox"/>	tutorial_daily_report_app-delete-activematters	【チュートリアル】 日報アプリケーション WF未済	
<input type="checkbox"/>	tutorial_daily_report_app-delete-archivematters	【チュートリアル】 日報アプリケーション WF過去	
<input type="checkbox"/>	tutorial_daily_report_app-delete-completedmatters	【チュートリアル】 日報アプリケーション WF完了	
<input type="checkbox"/>	tutorial_daily_report_app-delete-user-data	【チュートリアル】 日報アプリケーション ユーザ	
<input type="checkbox"/>	tutorial_daily_report_app-get-user-data	【チュートリアル】 日報アプリケーション ユーザ	
<input type="checkbox"/>	tutorial_daily_report_app-wf-action	【チュートリアル】 日報アプリケーション WFア	
<input type="checkbox"/>	tutorial_enquete_info	【チュートリアル】 アンケート情報取得	

1 ページ中 1 ページ目 50 9 件中 1 - 9 を表示

「ロジックフロールーティング定義情報」の「ルーティング」を「tutorial_enquete/answer」と設定します。

「ロジックフロールーティング定義情報」の「メソッド」で「POST」を指定します。

「ロジックフロールーティング定義情報」の「認可URI」を「tutorial_enquete/answer」と設定します。

その他の設定は標準のまま、「登録」をクリックしてURLを設定します。

ロジックフロールーティング定義編集

対象ロジックフロー定義情報

対象フロー

フロー定義ID

フロー定義名

バージョン番号

最新バージョンを利用する

利用するバージョンを指定する

利用バージョン

ロジックフロールーティング定義情報

ルーティング

メソッド

認証方法

認可URI

セキュアトークンを利用する

レスポンス種別

レスポンスヘッダ

ヘッダ名	ヘッダ値	削除
<input type="text"/>	<input type="text"/>	<input type="button" value="X"/>

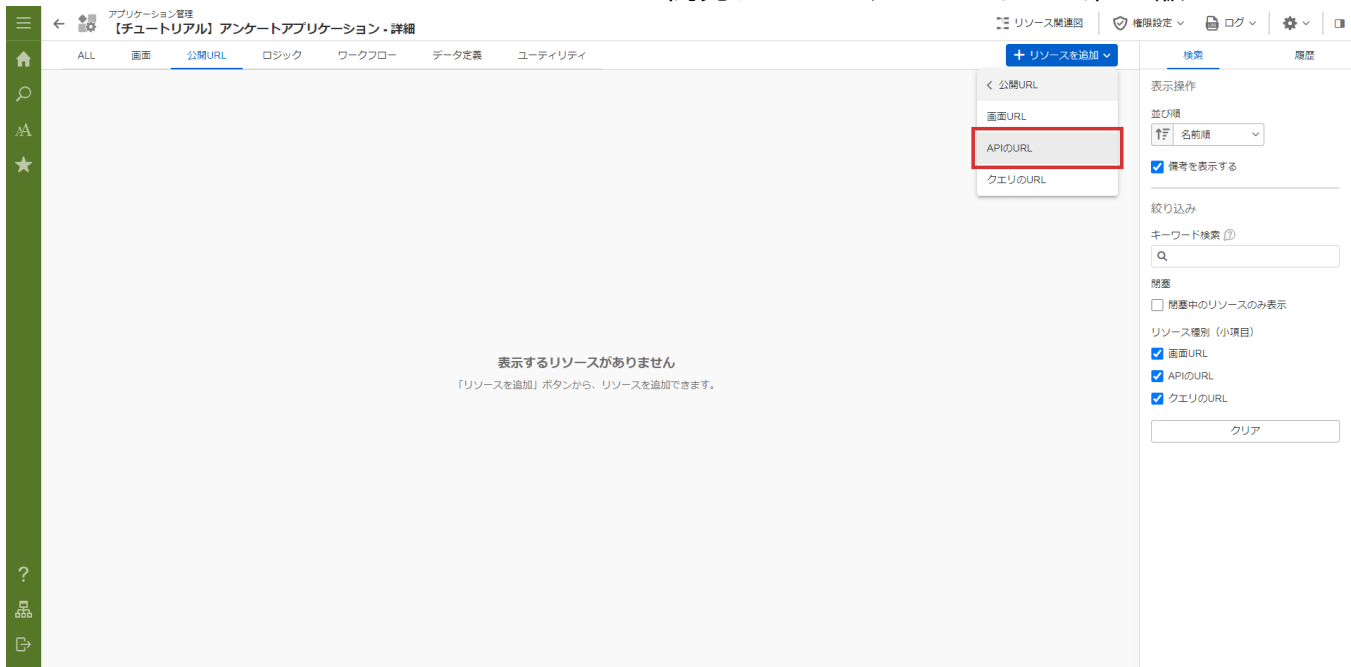
アンケート回答取得処理のURL設定

アンケート情報取得処理のURL設定を行います。

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

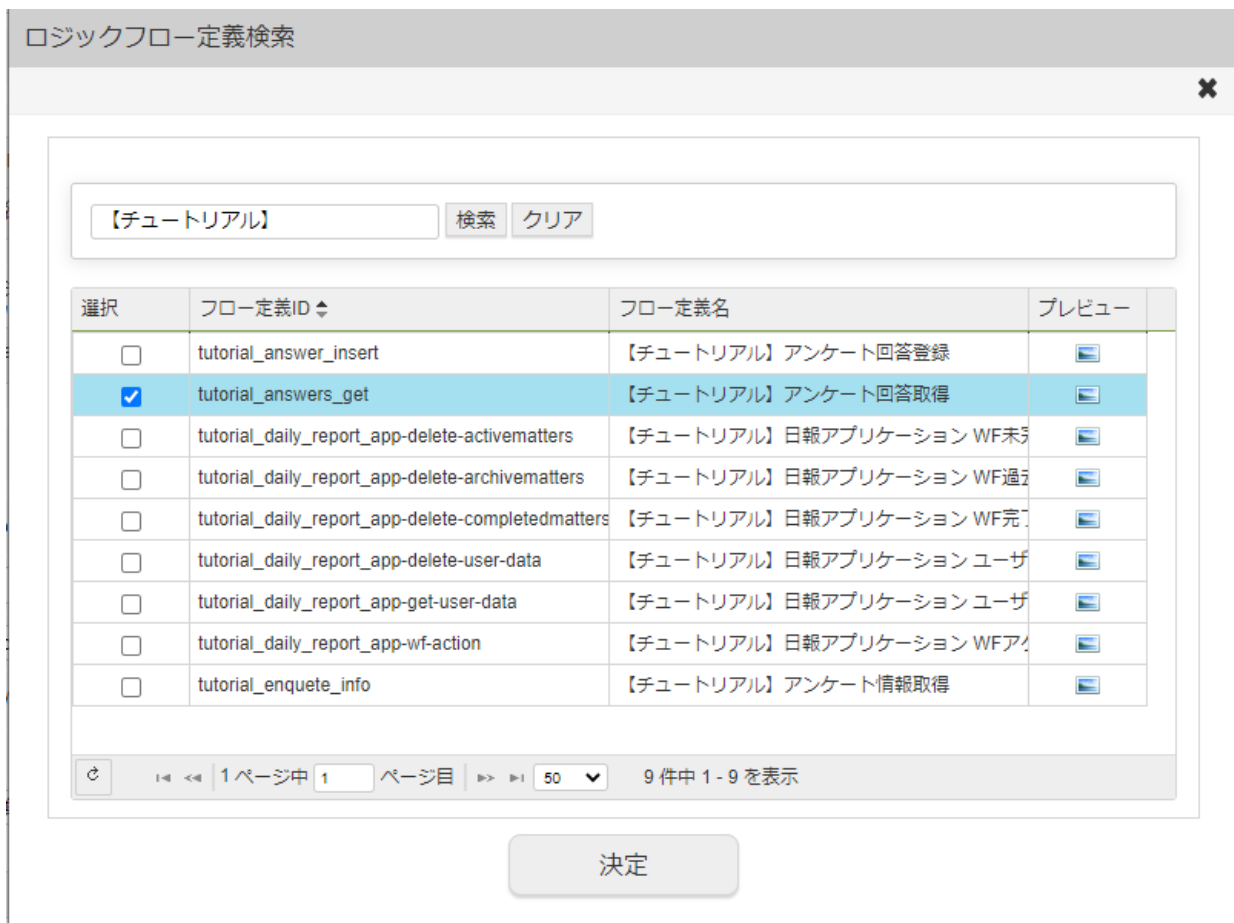
「【チュートリアル】アンケートアプリケーション」のアプリケーション管理画面で「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「APIのURL」を選択し、「ロジックフロールーティング定義編集」画面を開きます。



「ロジックフロールーティング定義編集」画面の「対象ロジックフロー定義情報」 - 「対象フロー」 - 「検索」からロジックフロー定義検索ダイアログを表示します。

ダイアログ上で「【チュートリアル】 アンケート回答取得」を選択し、「決定」をクリックします。



「ロジックフロールーティング定義情報」の「ルーティング」を「tutorial_enquete/answer/list」と設定します。

「ロジックフロールーティング定義情報」の「認可URI」を「tutorial_enquete/answer/list」と設定します。

その他の設定は標準のまま、「登録」をクリックしてURLを設定します。

対象ロジックフロー定義情報

対象フロー

フロー定義ID

フロー定義名

バージョン番号

最新バージョンを利用する

利用するバージョンを指定する

利用バージョン

ロジックフロールーティング定義情報

ルーティング

メソッド

認証方法

認可URI

セキュアトークンを利用する

レスポンス種別

レスポンスヘッダ

ヘッダ名	ヘッダ値	削除
<input type="text"/>	<input type="text"/>	<input type="button" value="✖"/>

業務ロジックのURLの認可設定

上記で設定した業務ロジックのURLを利用するためには、URLの設定とは別に、認可設定が必要です。

認可設定を行わないと、今までに作成してきたアプリケーションの業務ロジック等が動作しません。

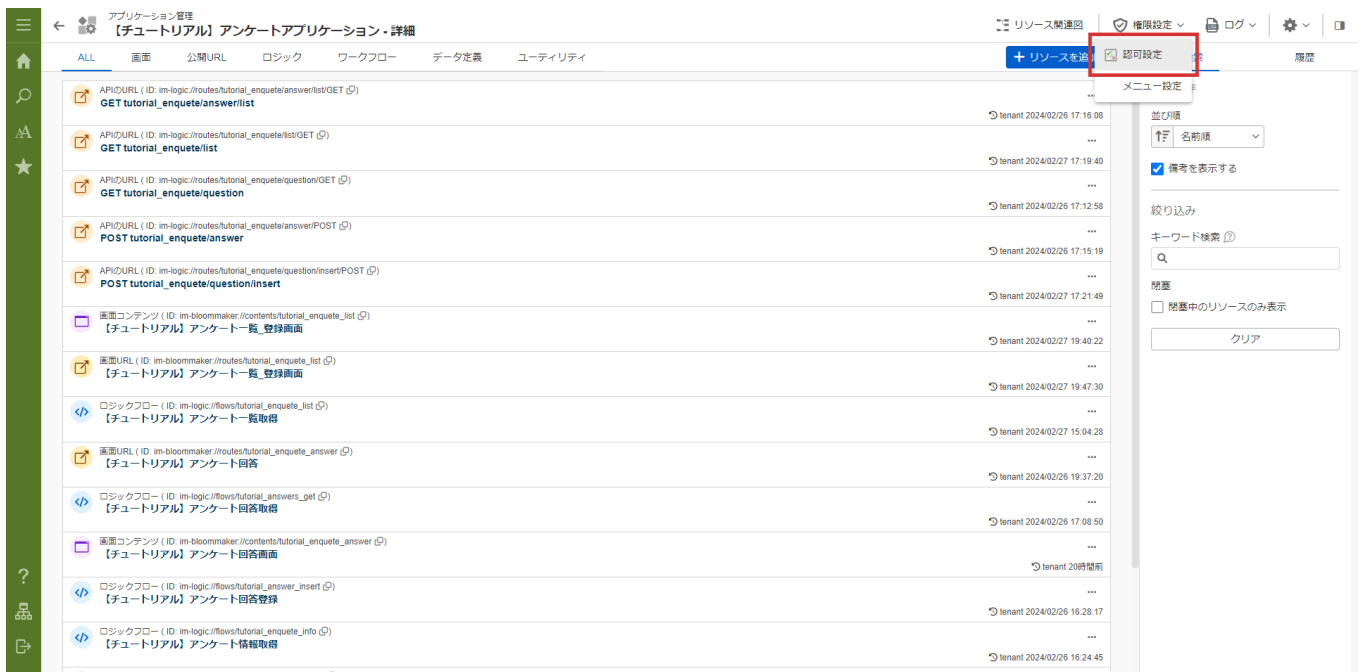
この節では認証済みユーザが業務ロジックのURLを利用できるようにするため、業務ロジックのURLの認可設定を行います。

認可設定

認可設定はアプリケーション管理画面から行います。

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

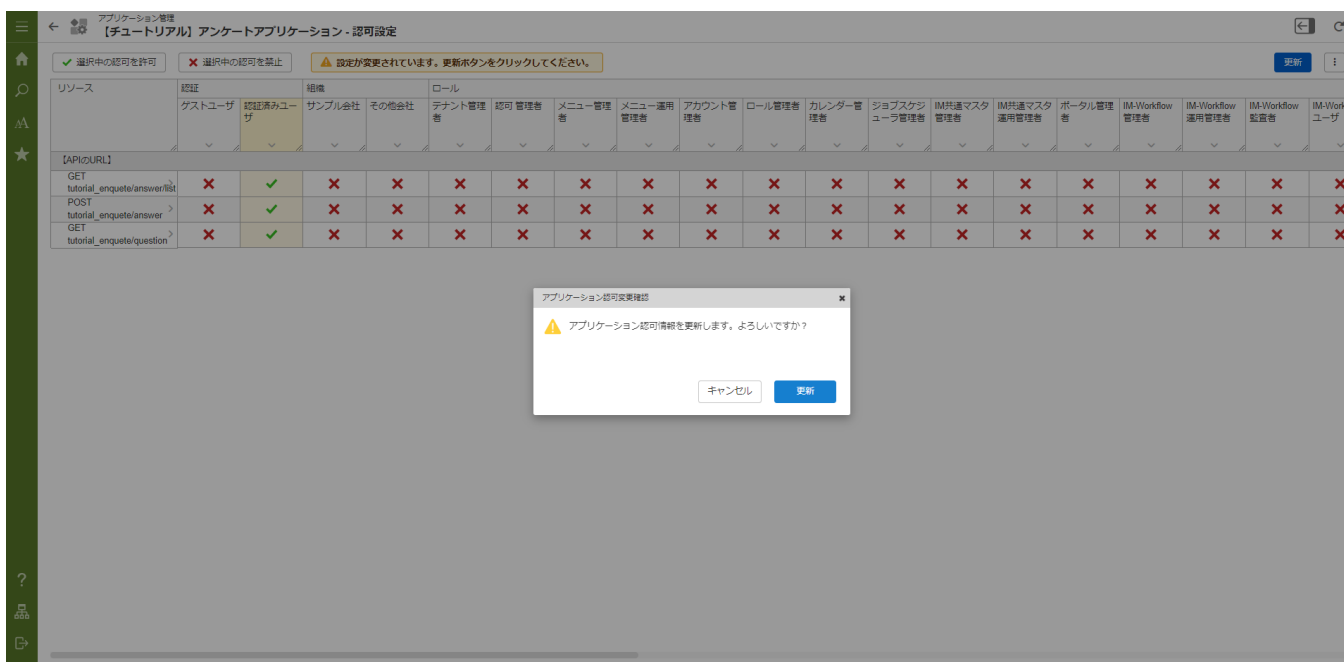
アプリケーション管理画面上部の「権限設定」プルダウンから認可設定アイコンを選択します。



アプリケーション認可設定画面で、「認証」 - 「認証済みユーザ」の列を選択し、「選択中の認可を許可」をクリックします。



「更新」をクリックし、認可設定の更新を行います。



これでログイン可能なすべてのユーザが、業務ロジックのURLを利用できるようになりました。

回答画面の作成

アンケートの回答を行うための画面を作成します。

アンケート回答画面はアンケートIDを受け取り、アンケート情報と設問情報を取得して画面に表示します。

アンケートIDが指定されていない、アンケートIDが誤っていてアンケートに関するデータが取得できない場合はエラー画面を表示します。

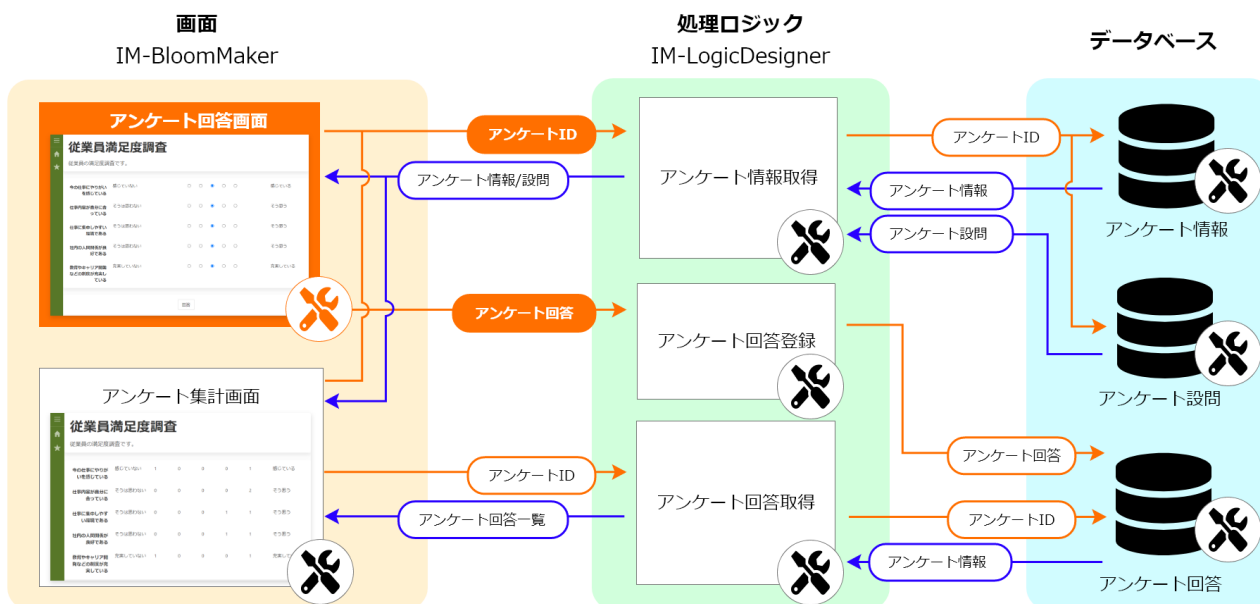
以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



チュートリアルで編集/作成



システムで自動生成



このページの手順について、以下の動画からも確認できます。併せてご利用ください。



注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

画面情報の設定

作成するアプリケーションに紐づいたアンケート回答画面を作成します。

画面コンテンツの基本情報の設定

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

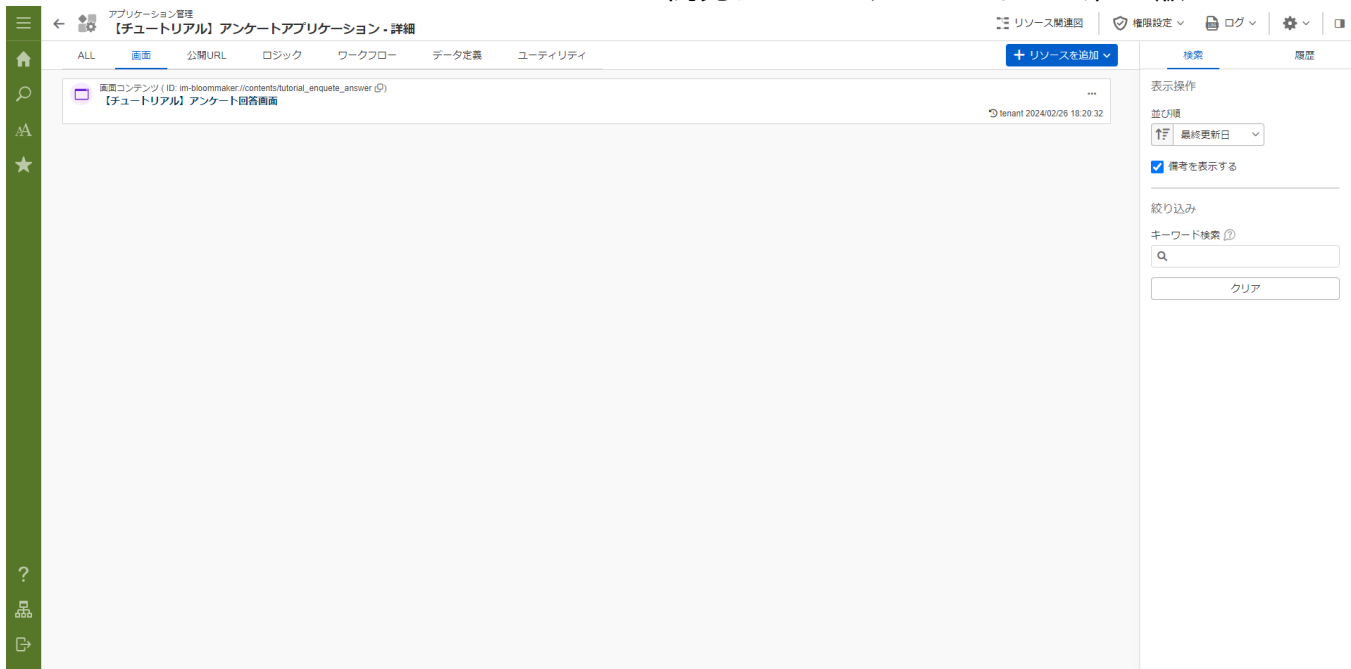
「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「画面」 - 「画面を新規作成」 - 「画面コンテンツ」を選択し、「IM-BloomMaker コンテンツ新規作成画面」を表示します。

画面コンテンツの「カテゴリ」の「検索」をクリックし、「業務テンプレートから作成」を設定します。
 コンテンツIDに「tutorial_enquete_answer」と設定します。
 コンテンツ種別で「Bulma」を選択します。
 コンテンツ名に「【チュートリアル】 アンケート回答画面」と入力し、「登録」をクリックして登録します。

アプリケーション管理画面に戻り、登録した画面コンテンツが画面一覧に表示されました。

画面コンテンツの作成

「アプリケーション管理画面」 - 「『画面』タブ」 - 「【チュートリアル】 アンケート回答画面」を選択し、IM-BloomMaker コンテンツ編集画面を開きます。



IM-BloomMaker コンテンツ編集画面の「デザイン編集」をクリックし、デザイナー画面を開きます。

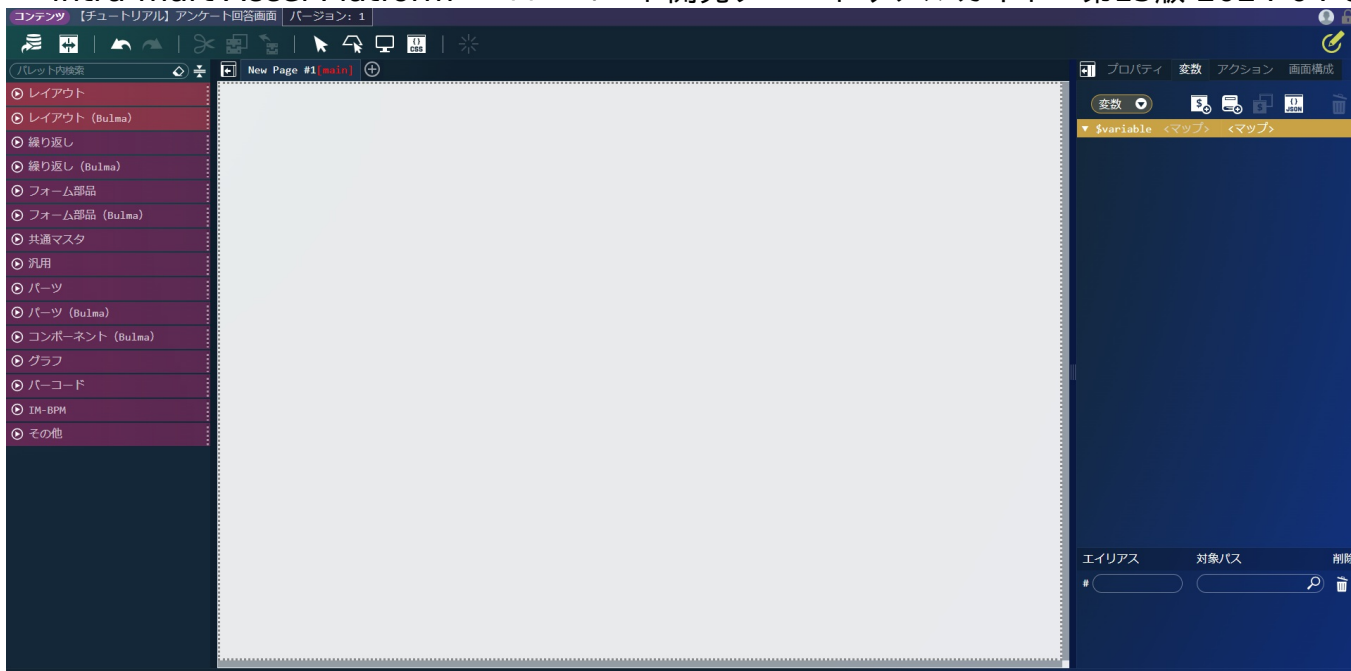


変数の設定

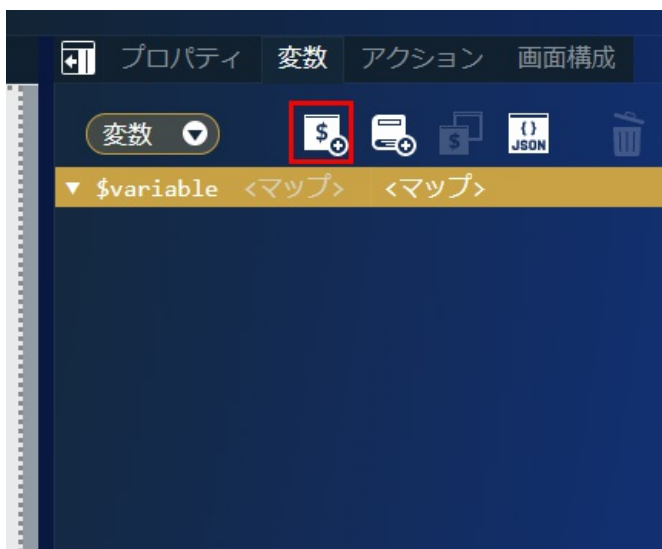
画面で利用する変数を作成します。

まず、画面に表示する設問情報を格納するための変数を作成します。この変数はアンケート情報取得処理の返却値を格納し、画面に表示するために利用します。

画面右側の「変数」タブを選択します。



変数一覧の「\$variable」を選択し、 をクリックして変数エディタを表示します。



変数エディタのキー名に「responseData」と入力し、値の型を「マップ」に変更して「決定」をクリックします。

変数エディタ

キー名: responseData

値の型: マップ

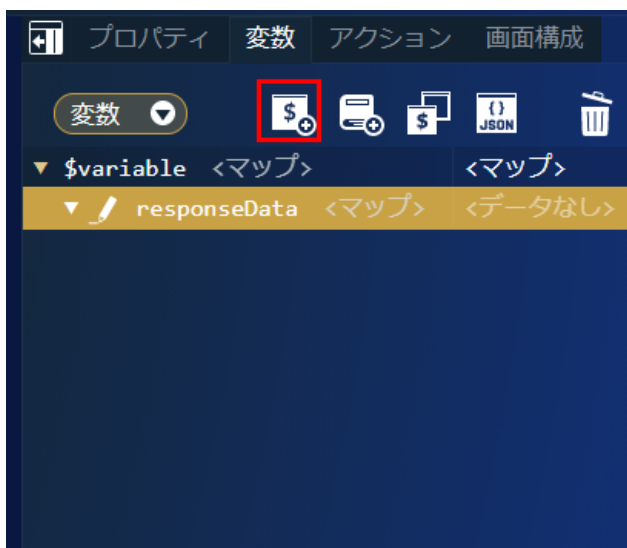
配列の次元数: 配列にしない

代入値:
 データなし
 マップ
 定数・入力・多言語・環境から


入力規則: 必須

決定 取り消し

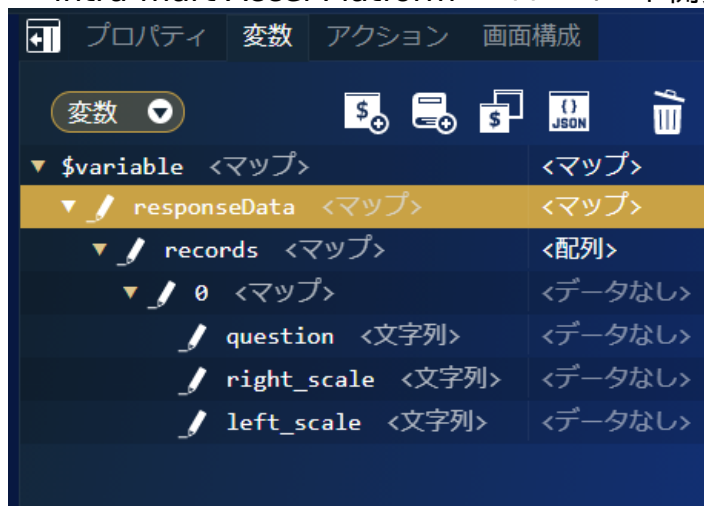
作成した「\$variable > responseData」を選択し、 をクリックして変数エディタを表示します。



変数エディタのキー名に「records」と入力し、値の型を「マップ」、配列の次元数を「1次元配列」、配列の要素数を「1」に変更して「決定」をクリックします。

作成した「\$variable > responseData > records > 0」を選択し、 をクリックして変数エディタを表示します。変数エディタのキー名に「question」と入力し、値の型を「文字列」に変更して「決定」をクリックします。

同様に「\$variable > responseData > records > 0」の配下に「right_scale」と「left_scale」を追加します。どちらの変数も値の型を「文字列」としてください。



次に、アンケートの基本情報を格納するための変数を作成します。

作成した「\$variable > responseData」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「enqueteTitle」と入力し、値の型を「文字列」に変更して「決定」をクリックします。




同様に「\$variable > responseData」の配下に「enqueteOverview」を追加します。値の型は「文字列」としてください。



変数はJSON入力で以下のコードを記載しても設定できます。

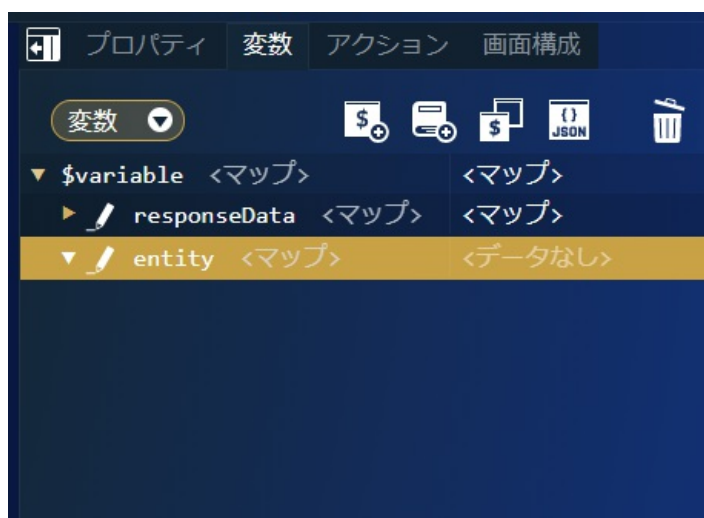
```
"responseData": {
  "records": [
    {
      "question": "",
      "right_scale": "",
      "left_scale": ""
    }
  ],
  "enqueteTitle": "",
  "enqueteOverview": ""
}
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型に誤りが無いか注意してください。

次に、アンケート情報取得処理に対して送信する値を格納するための変数を作成します。

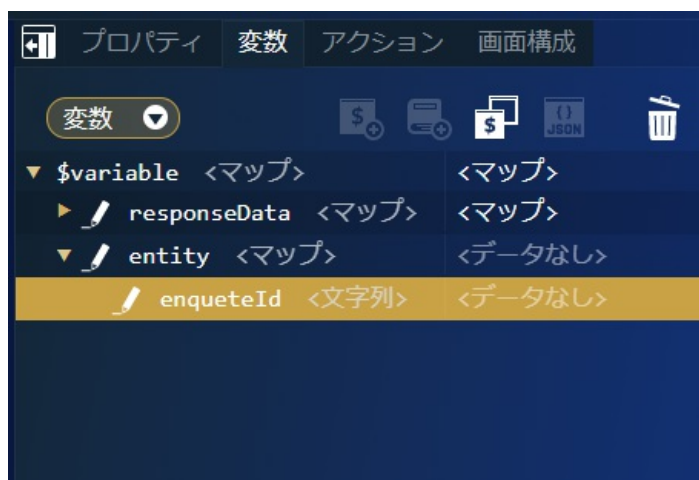
「\$variable」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「entity」と入力し、値の型を「マップ」、代入値を「マップ」に変更して「決定」をクリックします。




作成した「\$variable > entity」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「enqueteId」と入力し、値の型を「文字列」に変更して「決定」をクリックします。



変数はJSON入力で以下のコードを記載しても設定できます。

```
"entity": {
  "enqueteId": ""
}
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型に誤りが無いか注意してください。

次に、アンケート回答登録処理に対して送信する、アンケートの回答内容を格納するための変数を作成します。

「\$variable」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「answersData」と入力し、値の型を「マップ」、代入値を「マップ」に変更して「決定」をクリックします。

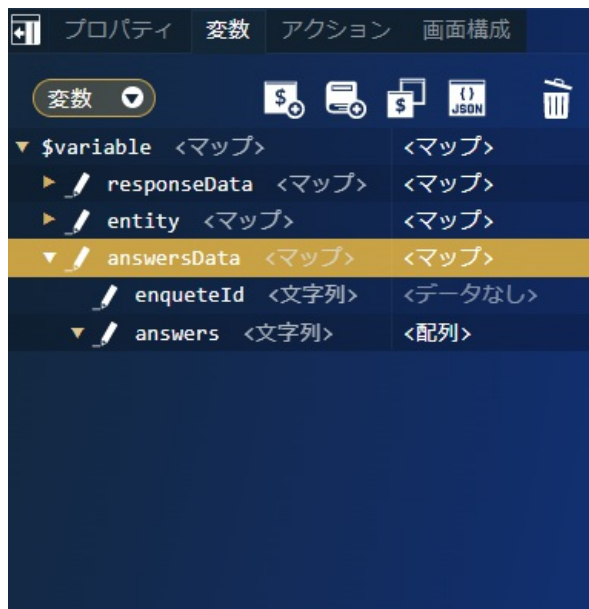


作成した「\$variable > answersData」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「enqueteId」と入力し、値の型を「文字列」に変更して「決定」をクリックします。


「\$variable > answersData」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「answers」と入力し、値の型を「文字列」、配列の次元数を「1次元配列」、配列の要素数を「0」、代入値を「配列」に変更して「決定」をクリックします。



変数はJSON入力で以下のコードを記載しても設定できます。

```
"answersData": {
  "enqueteId": "",
  "answers": []
}
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型に誤りが無いか注意してください。

エラー画面に表示するメッセージを格納するための変数を作成します。


「\$variable」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「errorMessage」と入力し、値の型を「文字列」に変更して「決定」をクリックします。



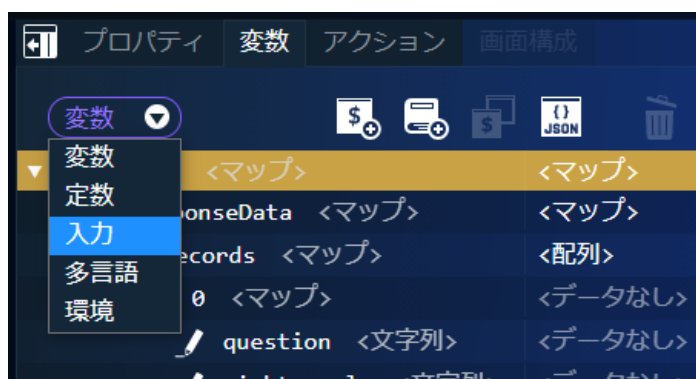
変数はJSON入力で以下のコードを記載しても設定できます。


```
"errorMessage": ""
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型に誤りが無いか注意してください。

初期表示時に設定される入力を設定します。

「変数」タブのプルダウンで「変数」を「入力」に変更します。




「\$input」を選択し、 をクリックして入力エディタを表示します。

入力エディタのキー名に「enqueteld」と入力し、値の型を「文字列」に変更して「決定」をクリックします。

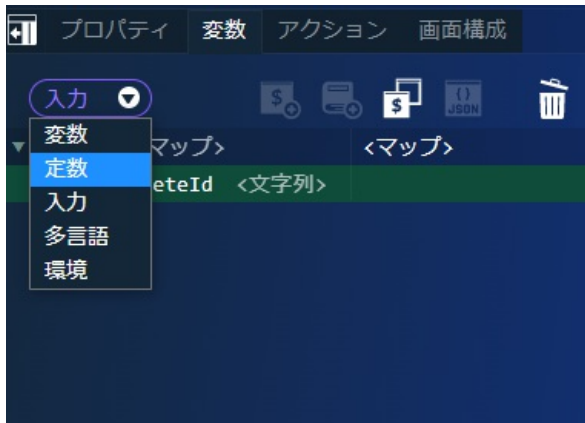
入力値はJSON入力で以下のコードを記載しても設定できます。

```
"enqueteld": ""
```


JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型に誤りが無いか注意してください。

変数設定の最後に、エラー発生時に表示するメッセージを「定数」で設定します。

「変数」タブのプルダウンで「入力」を「定数」に変更します。




まずは、アンケートIDが設定されなかった場合に表示するエラーメッセージを設定します。

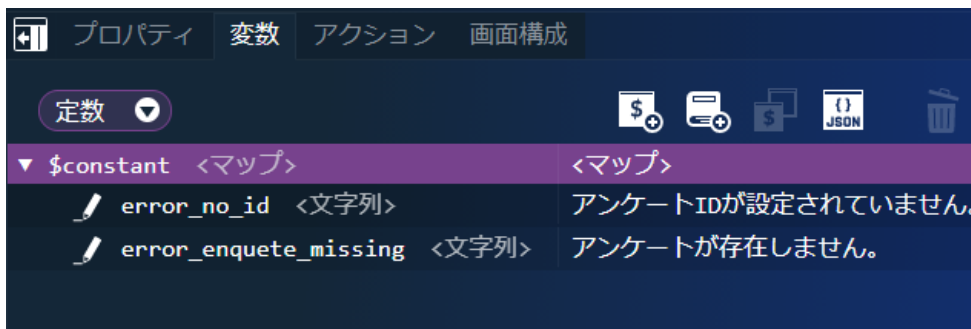
「\$constant」を選択し、 をクリックして定数エディタを表示します。

定数エディタのキー名に「error_no_id」と入力し、値の型を「文字列」、代入値のラジオボタンを「固定文字列」にして入力エリアに「アンケートIDが設定されていません。」と設定し、「決定」をクリックします。

次に、存在しないアンケートIDが指定された場合に表示するエラーメッセージを設定します。


「\$constant」を選択し、 をクリックして定数エディタを表示します。

定数エディタのキー名に「error_enquete_missing」と入力し、値の型を「文字列」、代入値のラジオボタンを「固定文字列」にして入力エリアに「アンケートが存在しません。」と設定し、「決定」をクリックします。



定数値もJSON入力で以下を記載して設定できます。

```
{
  "error_no_id": "アンケートIDが設定されていません。",
  "error_enquete_missing": "アンケートが存在しません。"
}
```

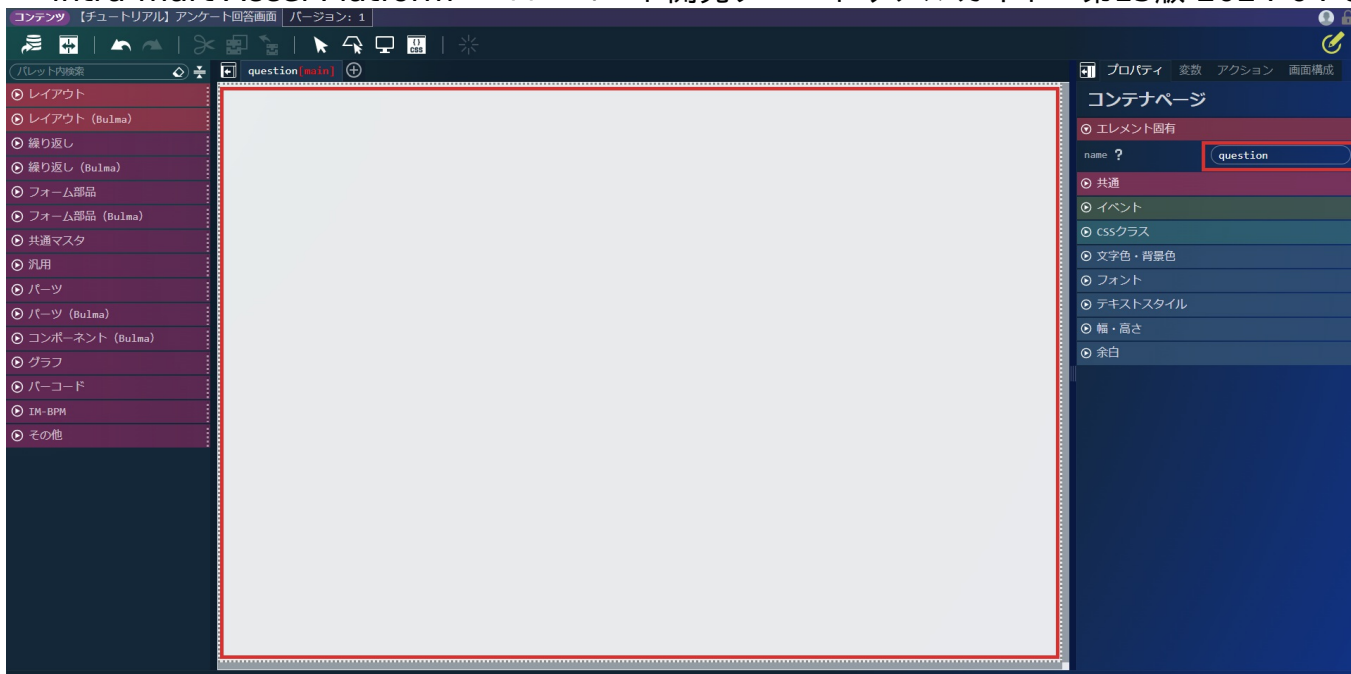
JSON入力する場合は、定数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型に誤りが無いか注意してください。

アンケート回答画面の作成

続けて、画面にアイテムを配置していきます。

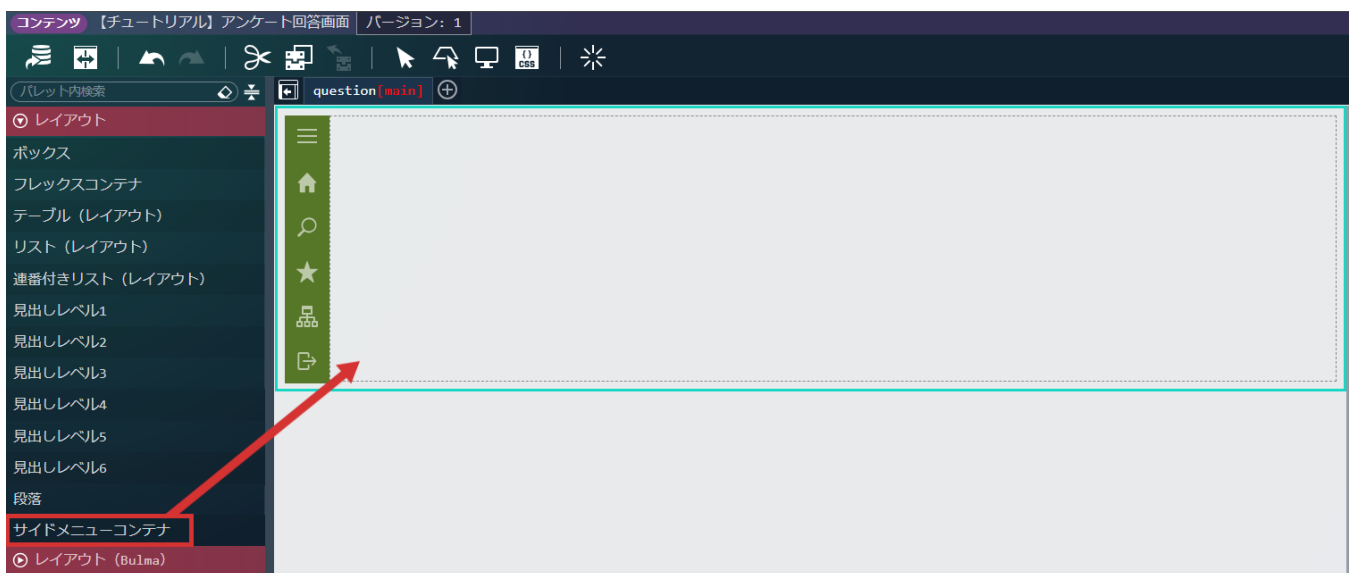
まずは、回答画面の名称を変更します。

デザイン編集エディタをクリックし、画面右側の「コンテナページ」 - 「エレメント固有」 - 「name」を「question」と設定します。



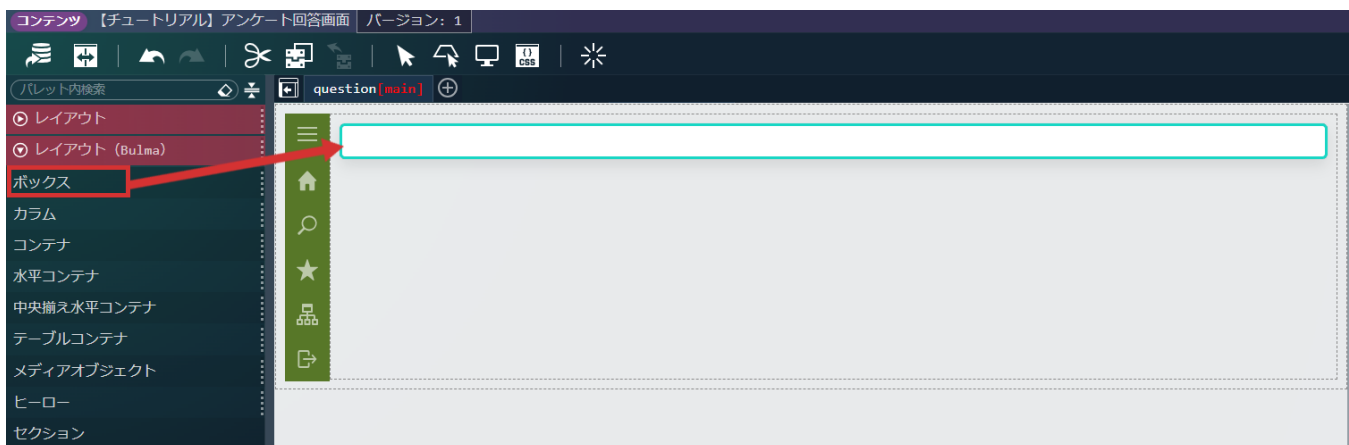
これで、タブの名称も「question」へ変更されました。次に、画面にサイドメニューを表示するため、サイドメニューコンテナを配置します。

画面左側のパレットの「レイアウト」 - 「サイドメニューコンテナ」を選択し、ドラッグ&ドロップで画面に配置します。

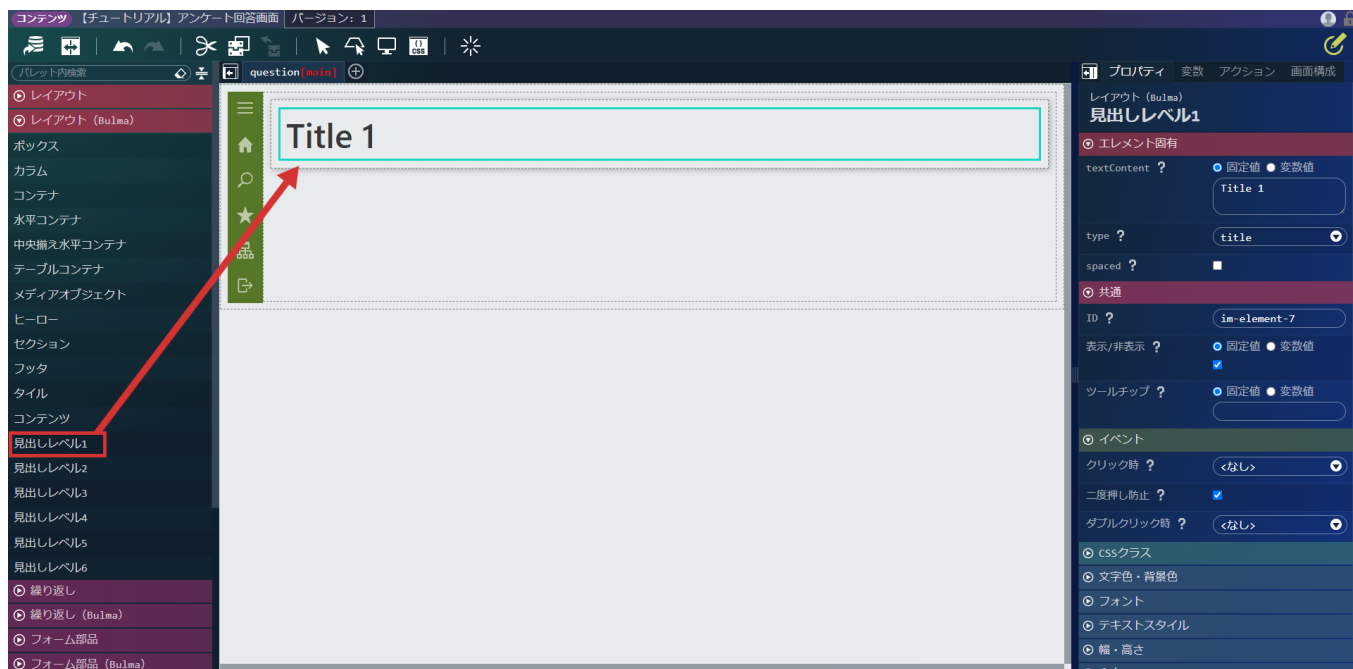


次に、アンケートのタイトルと概要の表示領域を作成します。

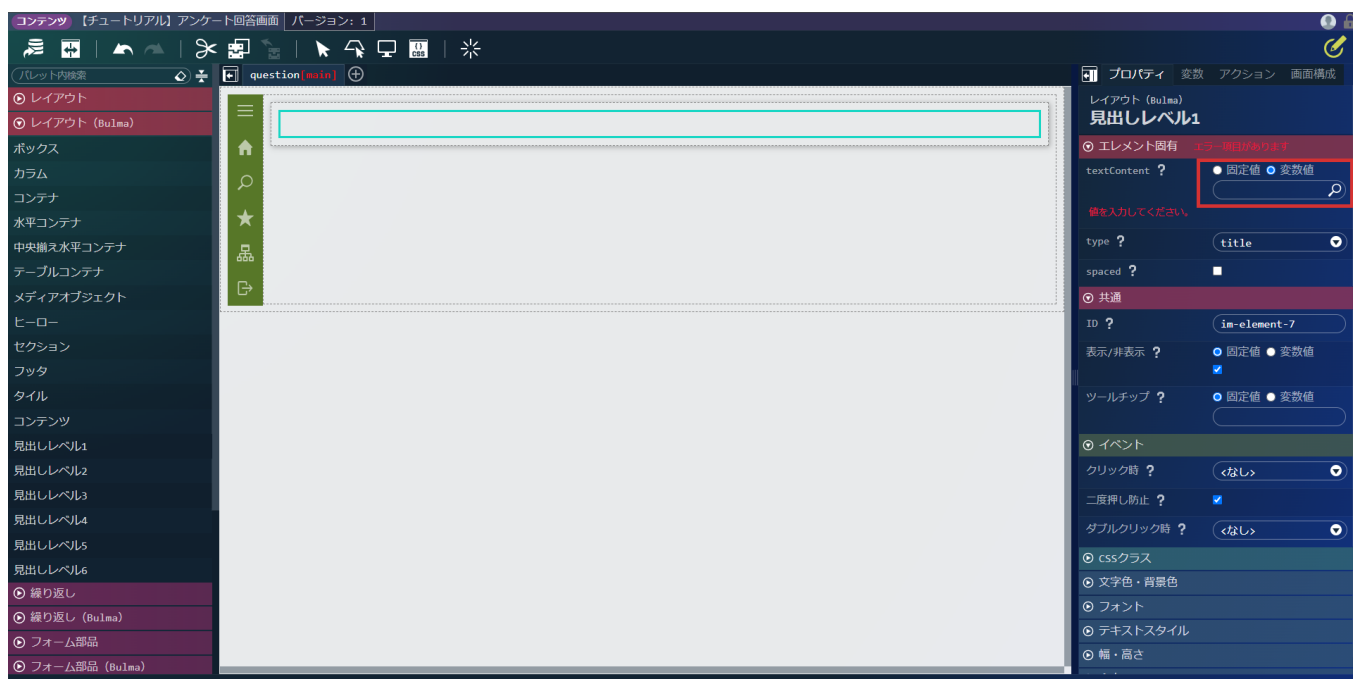
画面左側のパレットの「レイアウト (Bulma)」 - 「ボックス」を選択し、先ほど配置した「サイドメニューコンテナ」の中に配置します。



画面左側のパレットの「レイアウト (Bulma)」 - 「見出しレベル1」を選択し、先ほど配置した「ボックス」の中に配置します。



配置した「見出しレベル1」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「textContent」のラジオボタンを「変数値」へと変更し、🔍 をクリックします。



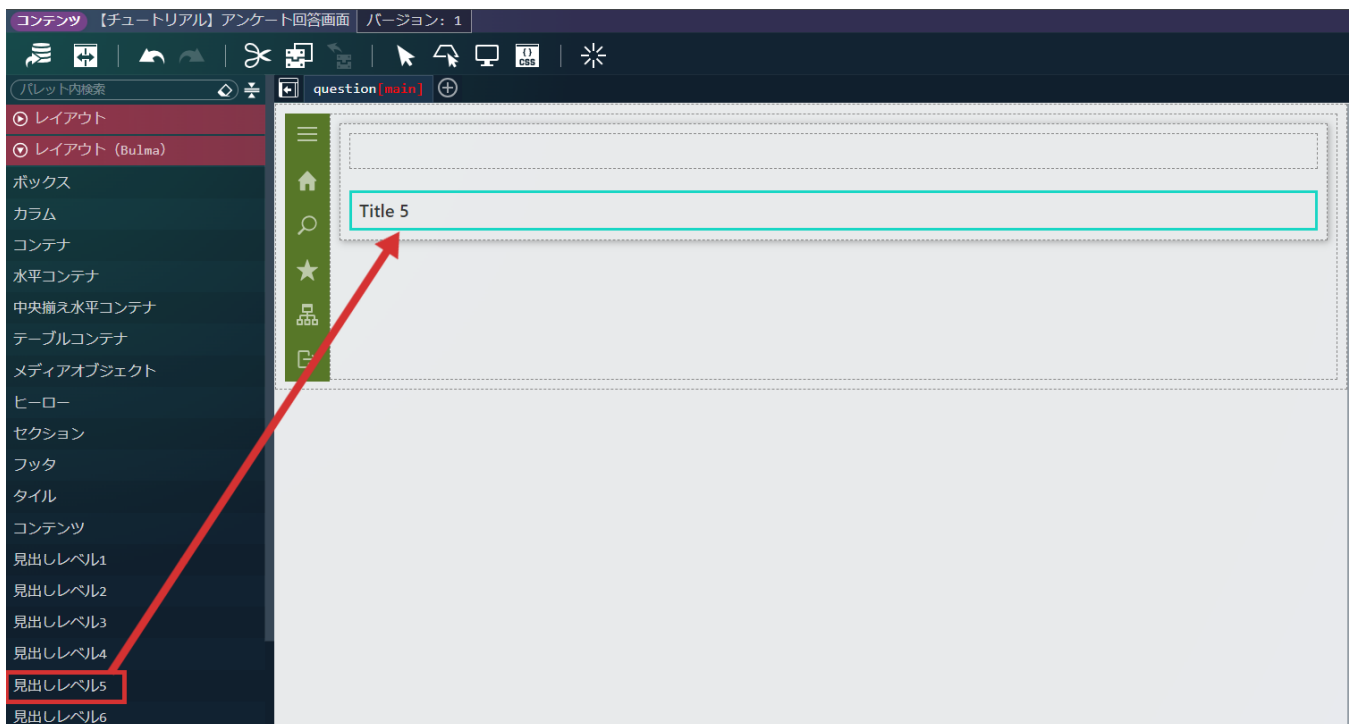
「\$variable > responseData > enqueteTitle」を選択し、「決定」をクリックして変数を設定します。



変数の設定では、textContentの入力ボックスに直接下記の文字列を入力してもかまいません。

「\$variable.responseData.enqueteTitle」

続けて、アンケート概要の表示欄を作成します。画面左側のパレットの「レイアウト (Bulma)」 - 「見出しレベル5」を選択し、先ほど配置した「ボックス」の中に配置します。



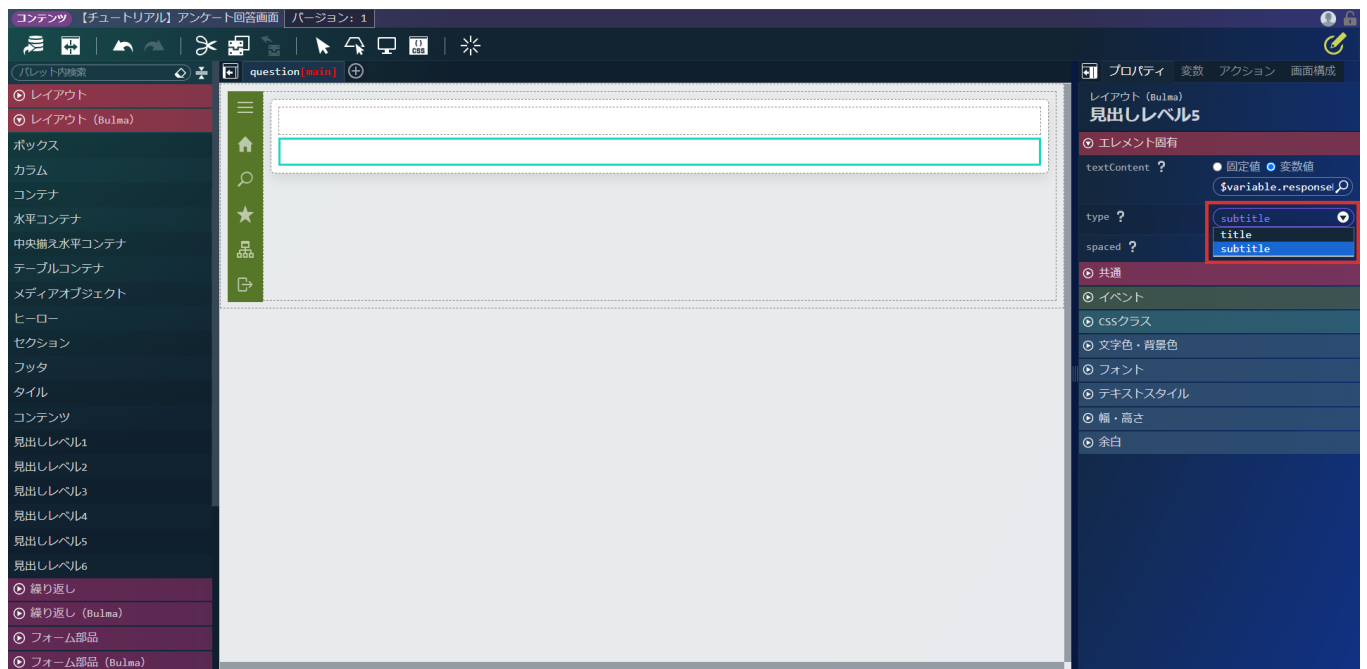
配置した「見出しレベル5」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「textContent」のラジオボタンを「変数値」へと変更し、🔍 から「\$variable > responseData > enqueteOverview」を設定します。



変数の設定では、textContentの入力ボックスに直接下記の文字列を入力してもかまいません。

「\$variable.responseData.enqueteOverview」

概要はサブタイトルとして表示するため、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「type」を「subtitle」に設定します。

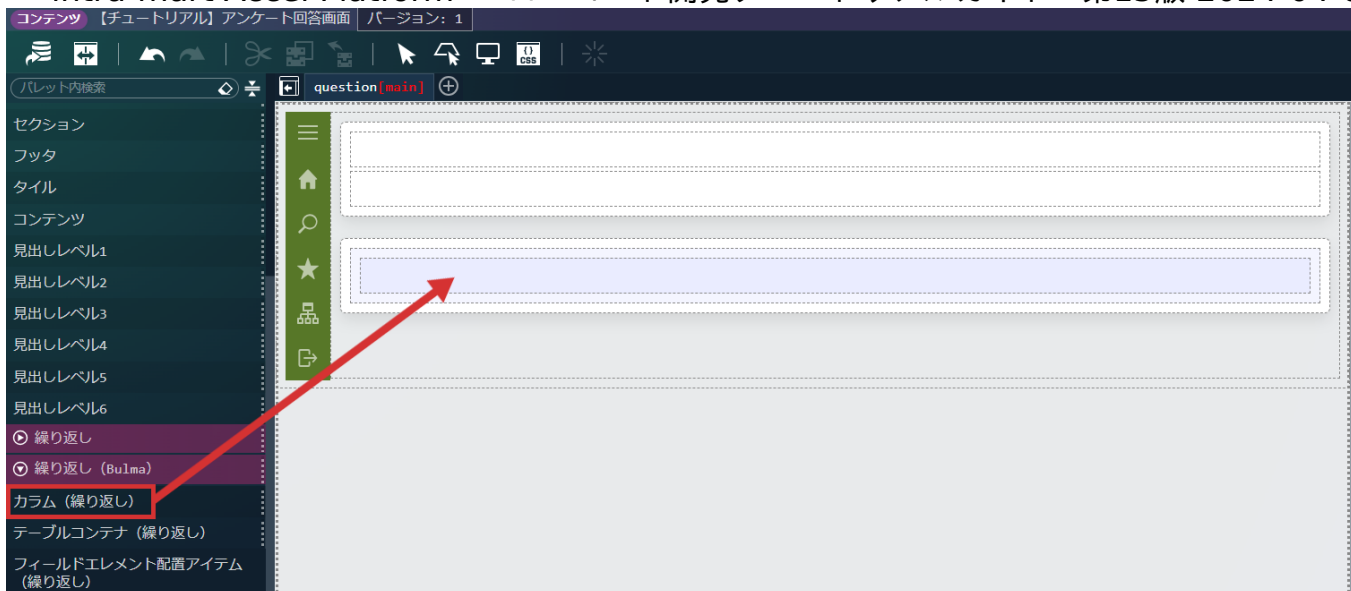


次に、設問の表示領域を作成します。

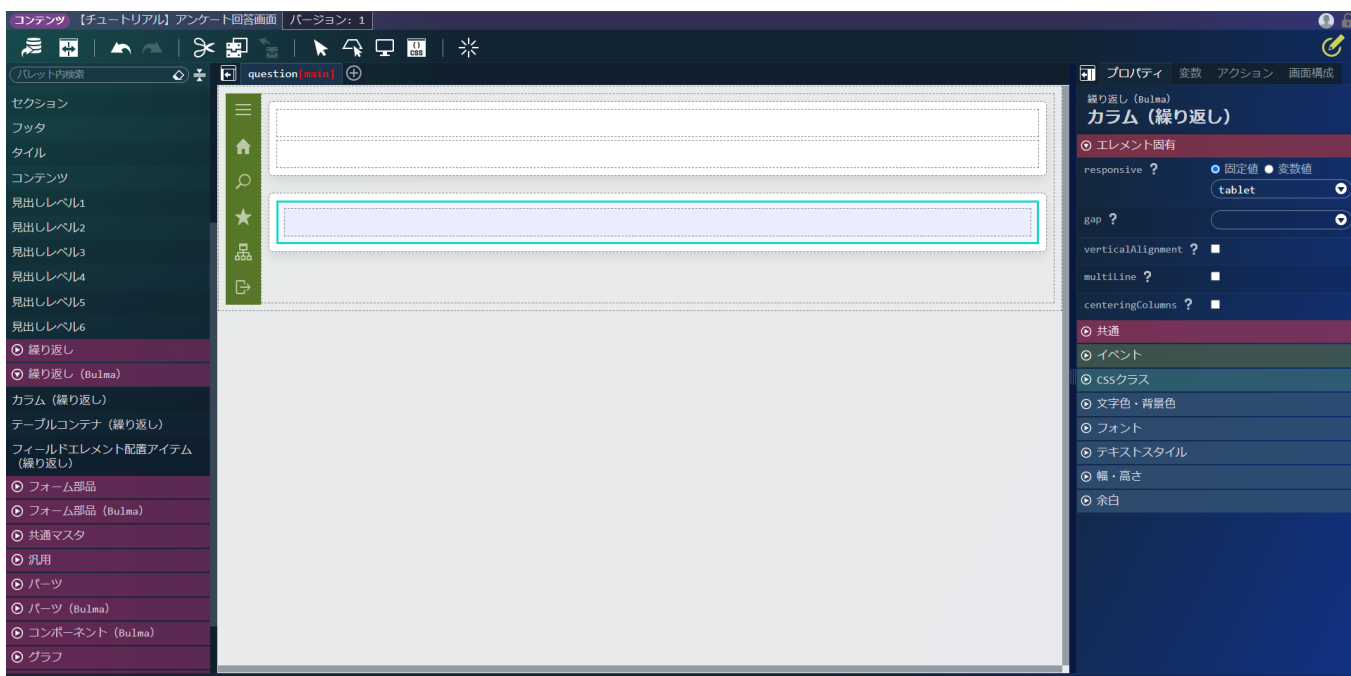
設問の表示領域はアンケートIDで取得した設問の配列くりかえすことによって表示します。

画面左側のパレットの「レイアウト (Bulma)」 - 「ボックス」を選択し、「サイドメニューコンテナ」の中に配置します。

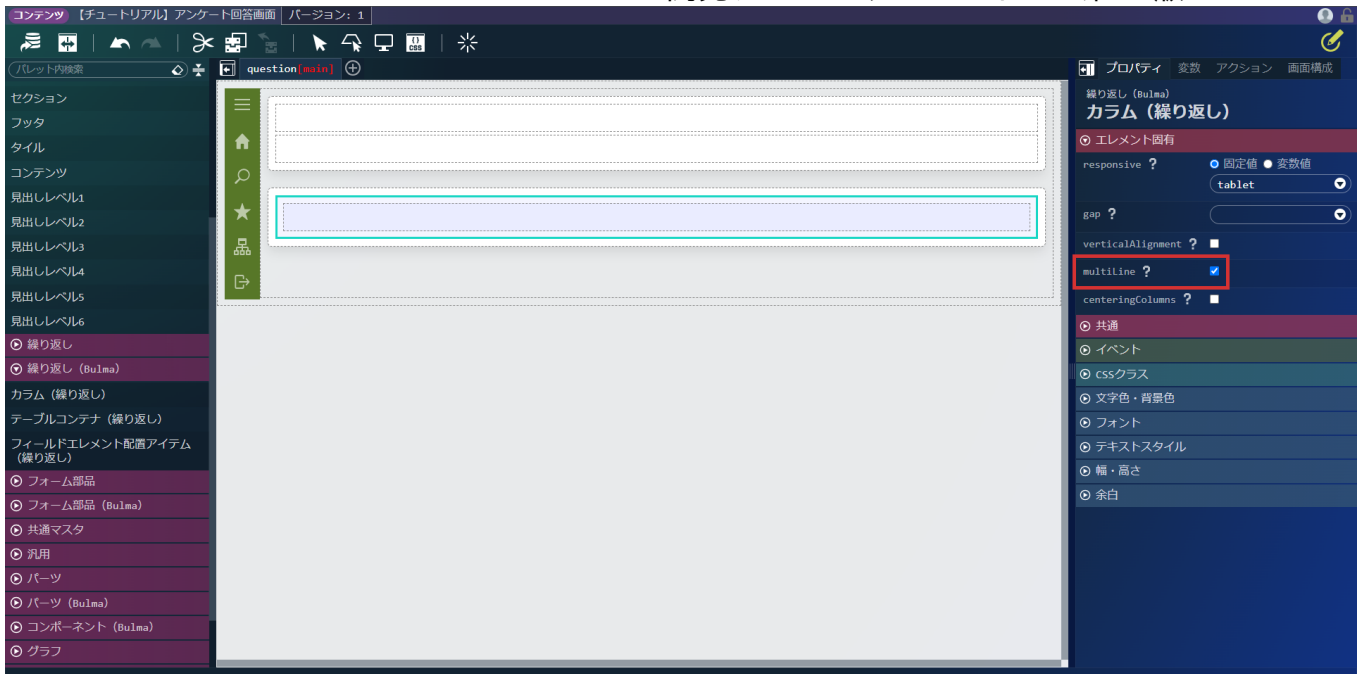
画面左側のパレットの「繰り返し (Bulma)」 - 「カラム (繰り返し)」を選択し、先ほど配置した「ボックス」の中に配置します。



配置した「カラム (繰り返し)」を選択します。



「カラム (繰り返し)」 - 「エレメント固有」 - 「multiLine」にチェックを入れます。



次に、「カラムアイテム（繰り返し）」を選択します。

「カラムアイテム（繰り返し）」 - 「エレメント固有」 - 「list」の 🔍 から「\$variable > responseData > records」を設定します。



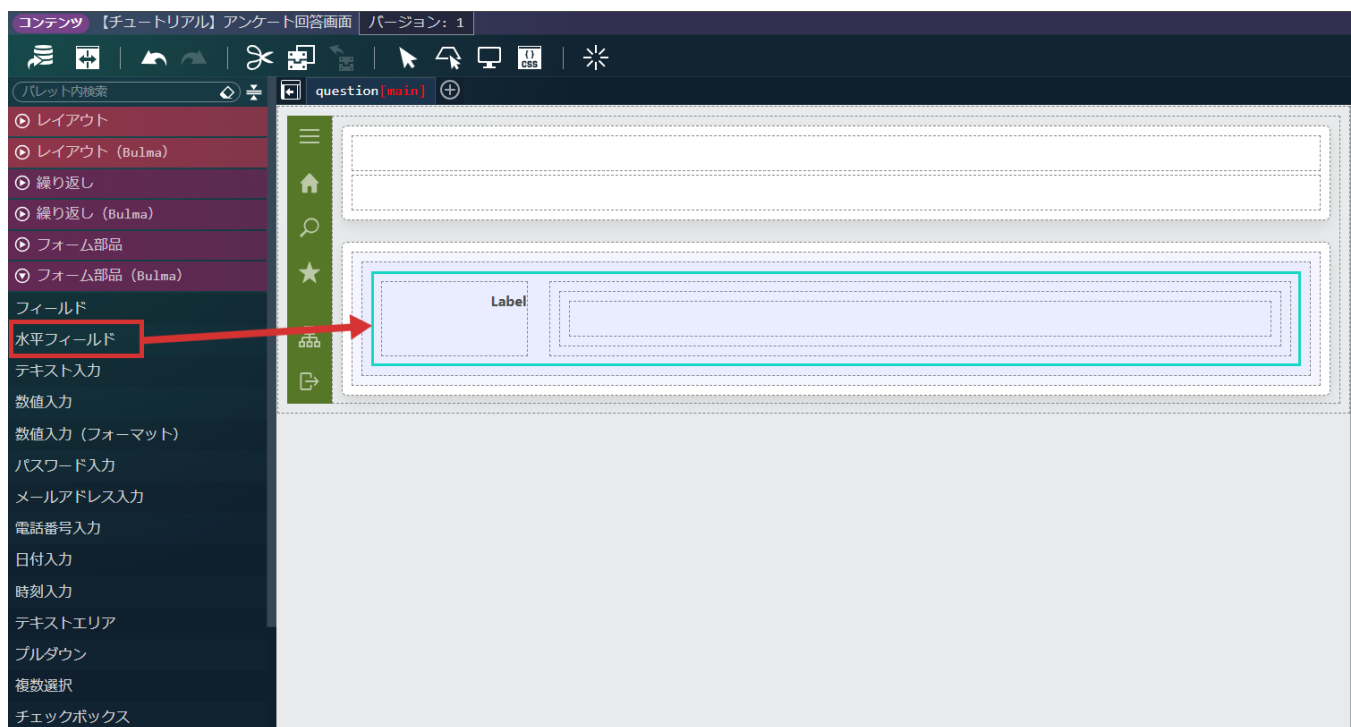
「カラムアイテム（繰り返し）」 - 「エレメント固有」 - 「size」に「full」を設定します。

「カラムアイテム（繰り返し）」 - 「エレメント固有」 - 「narrow」に「narrow」を設定します。

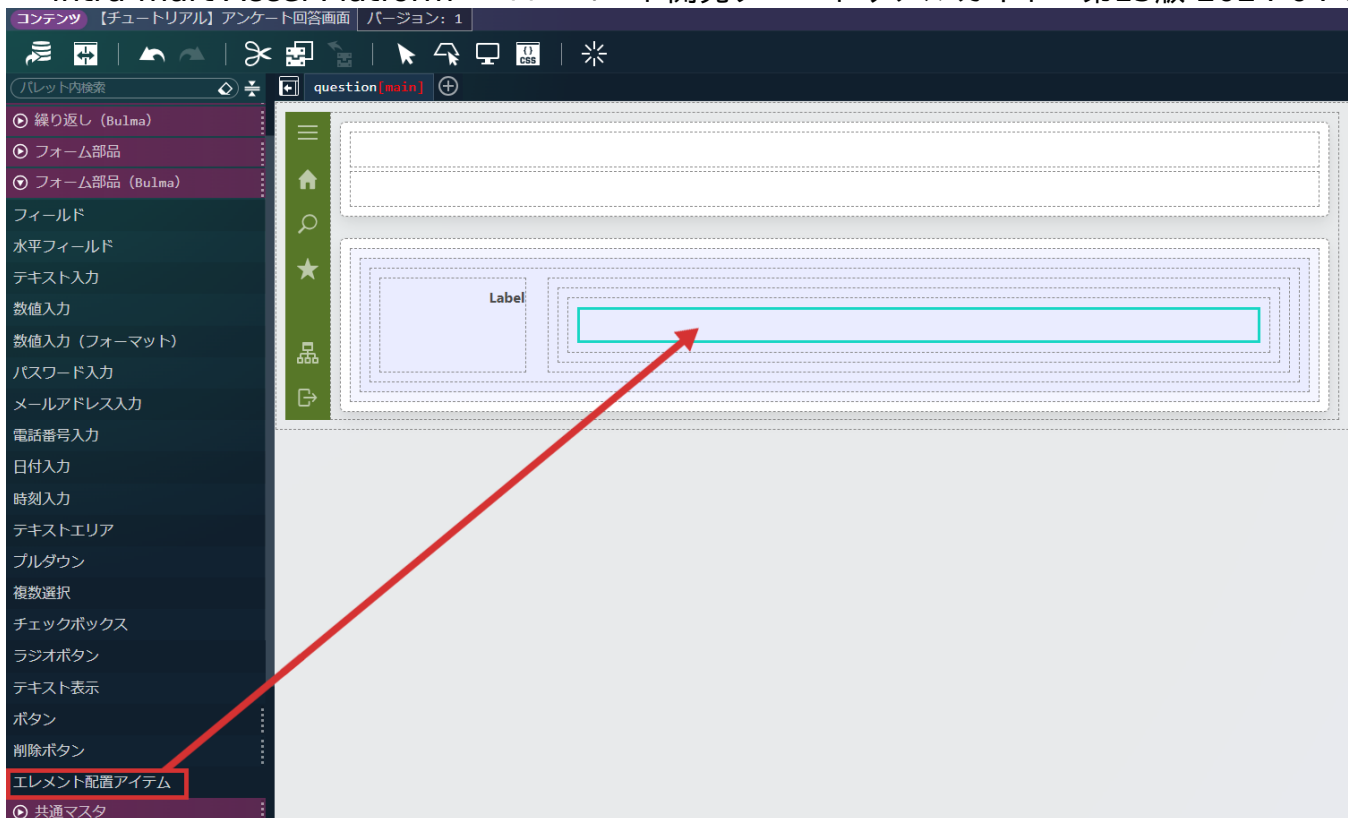


続けて、実際に回答内容を表示する領域を作成します。

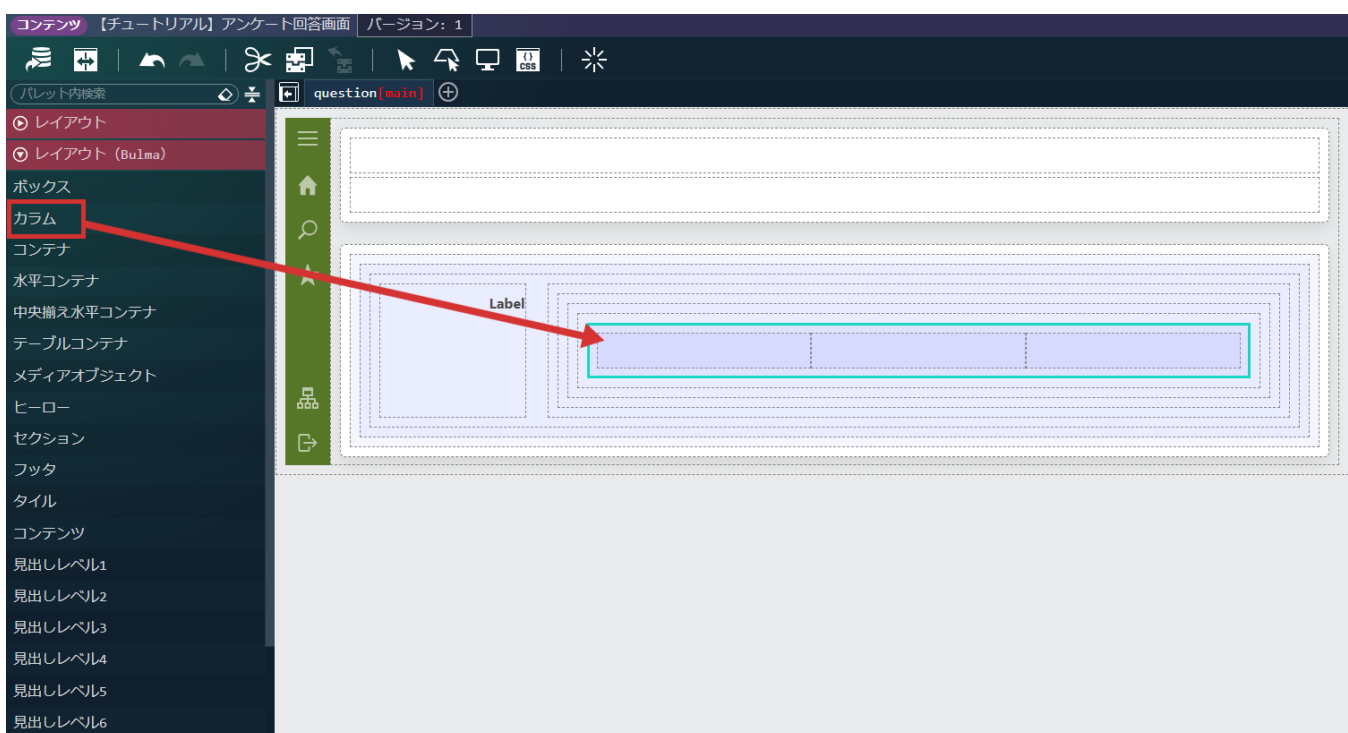
画面左側のパレットの「フォーム部品 (Bulma)」 - 「水平フィールド」を選択し、先ほど配置した「カラム (繰り返し)」内の「カラムアイテム (繰り返し)」の中に配置します。



画面左側のパレットの「フォーム部品 (Bulma)」 - 「エレメント配置アイテム」を選択し、先ほど配置した「水平フィールド」内の「フィールドコンテンツ」の中に配置します。

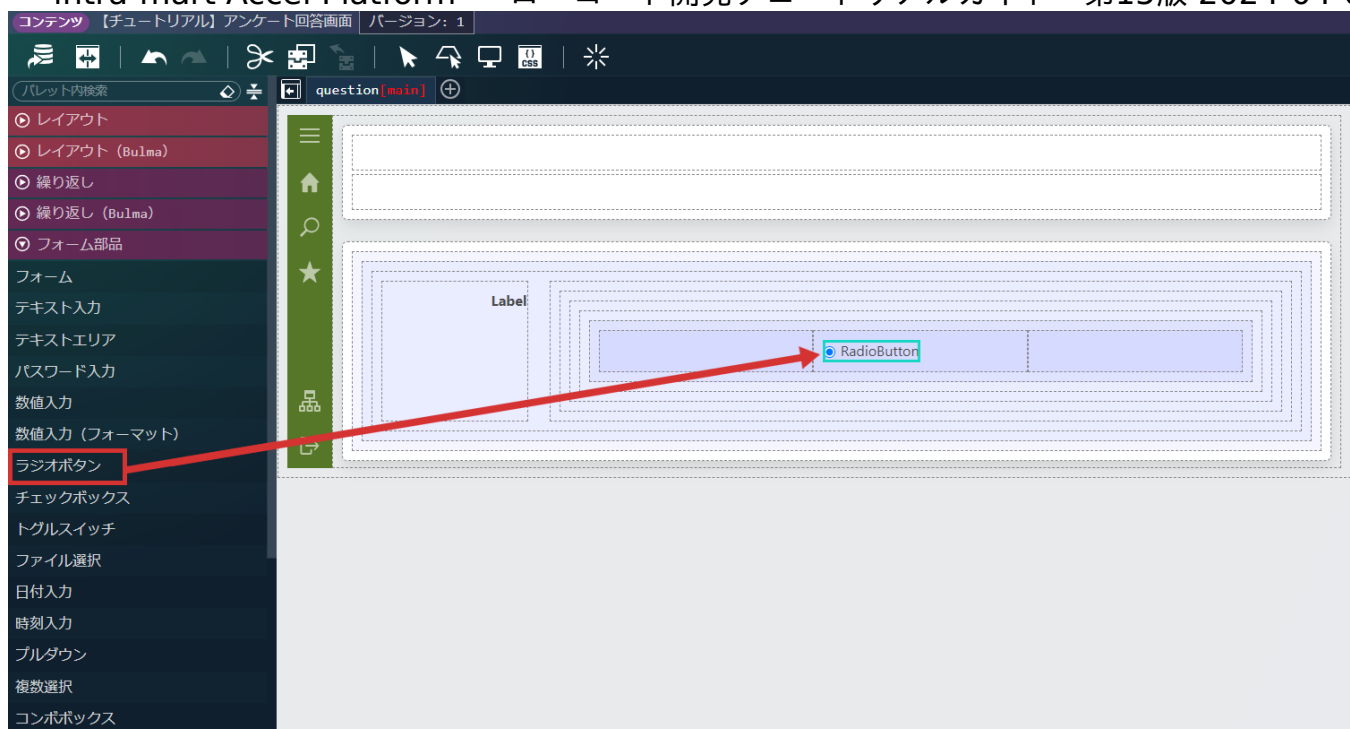


画面左側のパレットの「レイアウト (Bulma)」 - 「カラム」を選択し、先ほど配置した「エレメント配置アイテム」の中に配置します。



次に、設問の選択肢を配置します。

画面左側のパレットの「フォーム部品」 - 「ラジオボタン」を選択し、先ほど配置した「カラム」の中心のカラムに配置します。



配置したラジオボタンの設定を行います。

配置した「ラジオボタン」をクリックし、画面右側の「プロパティ」タブ - 「ラジオボタン」 - 「エレメント固有」 - 「textContent」の値を削除します。

「プロパティ」タブ - 「ラジオボタン」 - 「エレメント固有」 - 「value」に「1」と設定します。

「プロパティ」タブ - 「ラジオボタン」 - 「余白」 - 「外余白（左）」に「2rem」と設定します。

「プロパティ」タブ - 「ラジオボタン」 - 「エレメント固有」 - 「selected」のラジオボタンを「変数値」へと変更し、🔍 から「`$variable > answersData > answers`」を設定します。

「selected」のテキストエリアから、選択した変数の文字列の最後に「`[$index]`」と追記し、「`$variable.answersData.answers[$index]`」となるようにします。




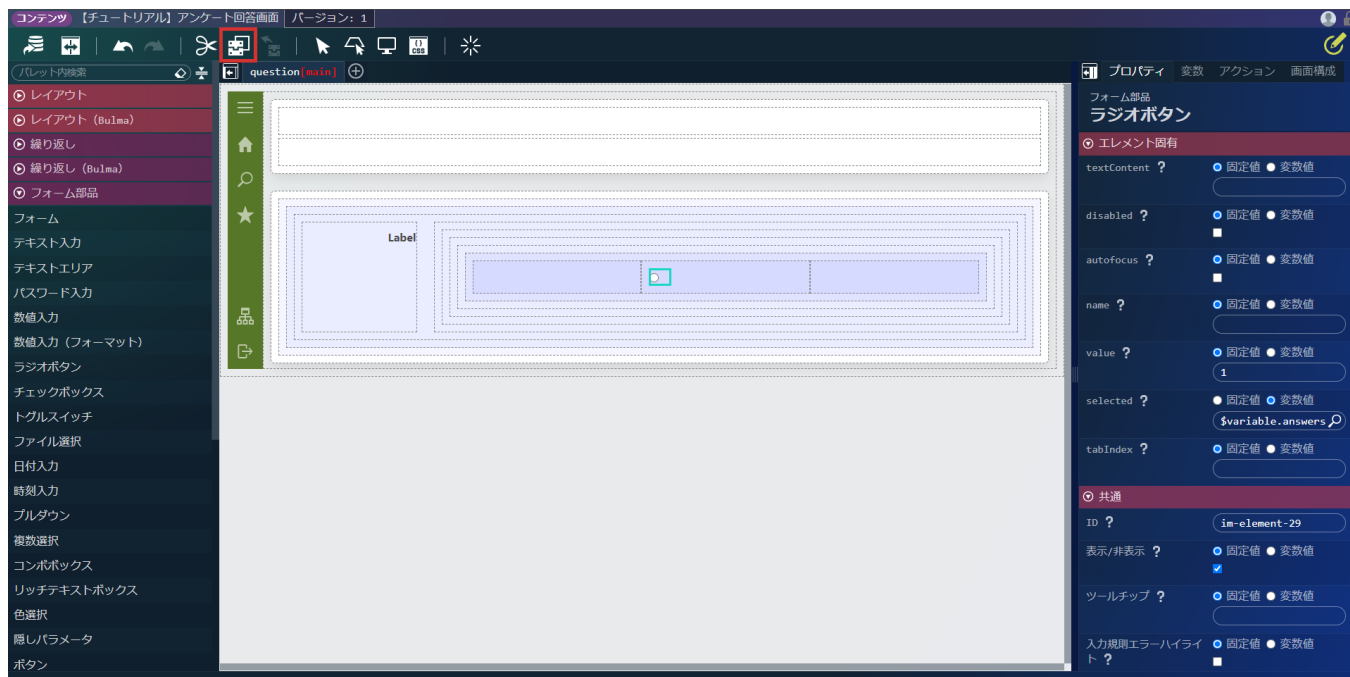
「ラジオボタン」 - 「エレメント固有」 - 「selected」の変数の設定では、selectedの入力ボックスに直接下記の文字列を入力してもかまいません。


「`$variable.answersData.answers[$index]`」

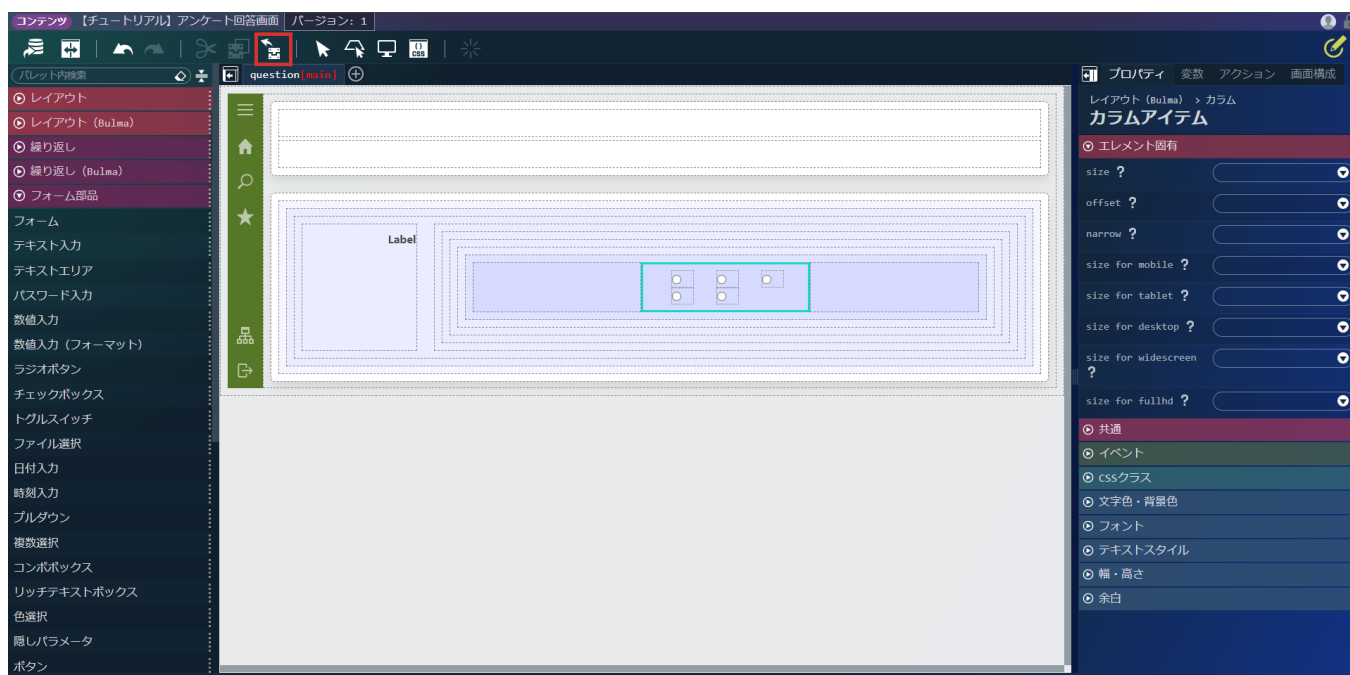
コラム

「繰り返し」の要素を利用すると、配列の変数を利用して配列の数だけ要素が表示されます。
 「繰り返し」の要素に配置した要素に対して変数を設定する場合、配列の添え字として「\$index」を指定することにより、繰り返しに合わせた変数値へと変換されます。
 詳細については、以下のCookBookを参照してください。
[「IM-BloomMaker 繰り返し要素での変数の使い方」](#)

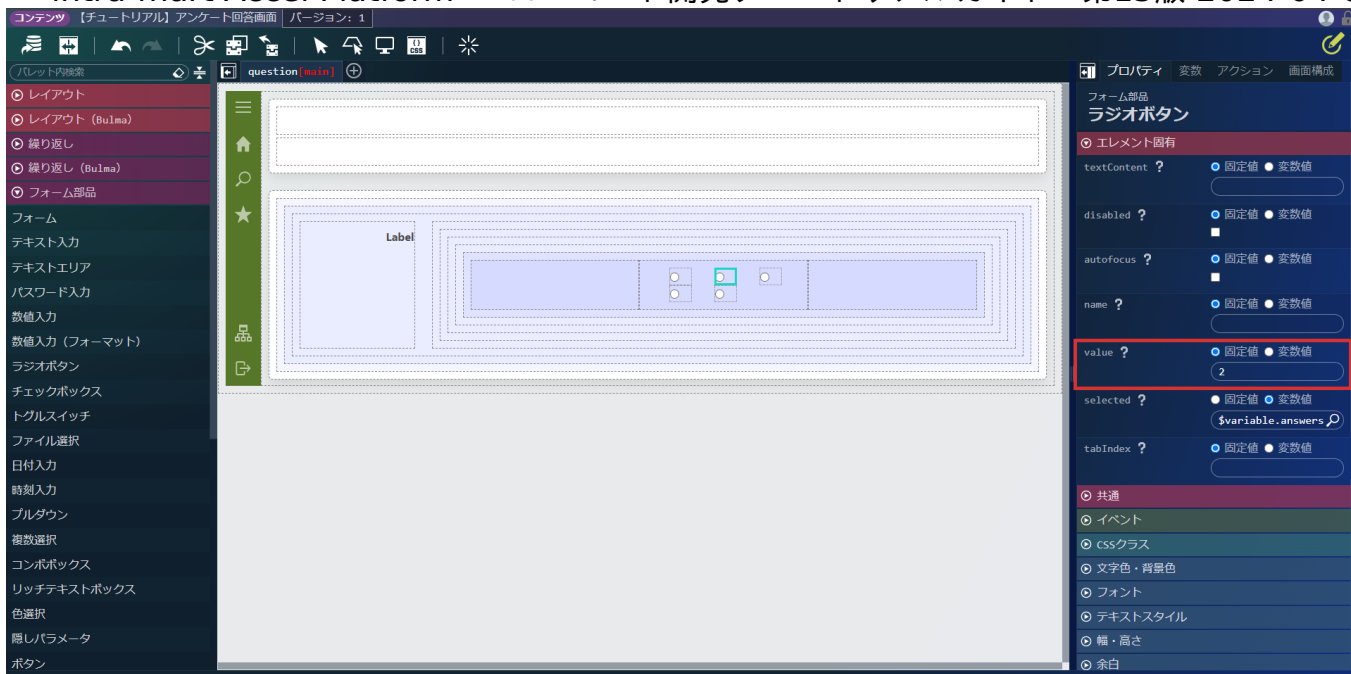
設問は5スケールでの回答を想定しているため、作成したラジオボタンをコピーします。
 ラジオボタンをクリックし、画面上部の  がキーボードのCtrl+Cでコピーします。



すでにラジオボタンが配置されているカラムアイテムをクリックし、画面上部の  がキーボードのCtrl+Vでラジオボタンが5つ並ぶまで貼り付けを繰り返します。



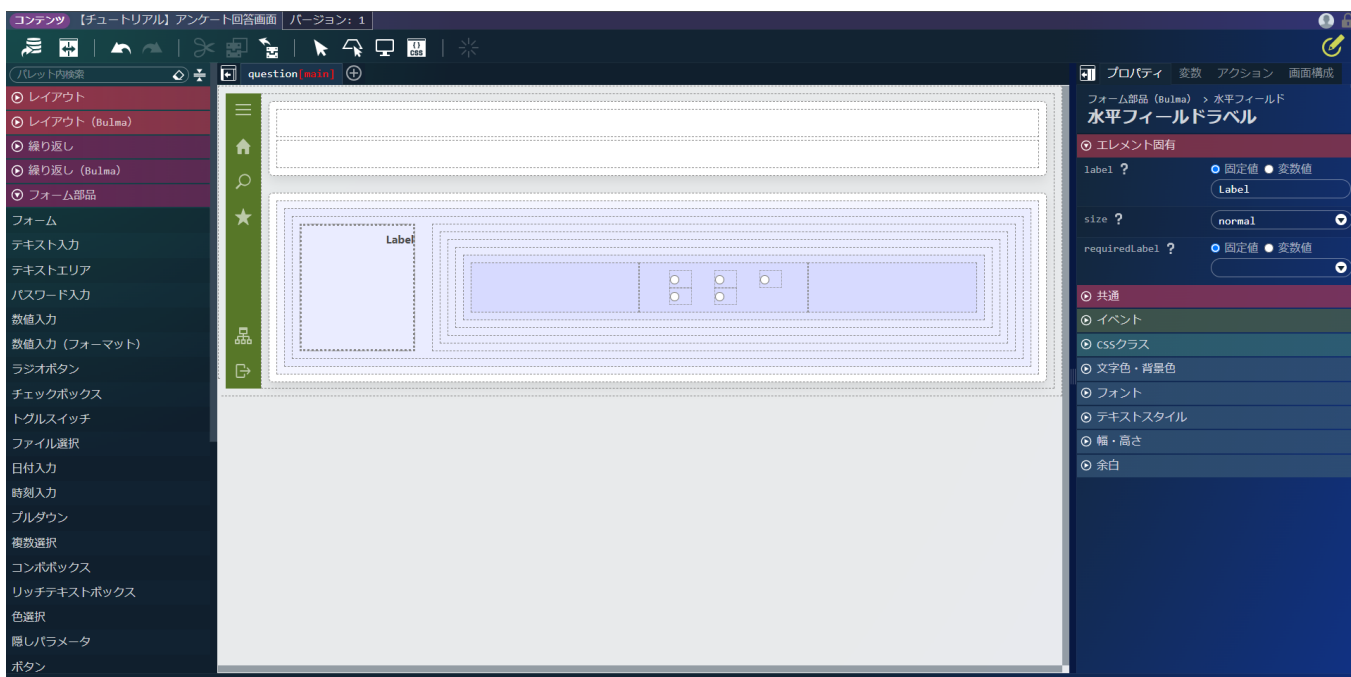
5つ並んだラジオボタンを左上から右へと順番に選択し、それぞれ画面右側の「プロパティ」タブ - 「ラジオボタン」 - 「要素固有」 - 「value」が「1」、「2」、「3」、「4」、「5」の順になるよう修正します。
 以下の画像は、2番目のラジオボタンを選択した場合の例です。




これで、設問ごとにどの選択肢を選んだかが\$variable.answersData.answersに順番に入りました。

次に設問内容を表示するラベルを配置します。

設問表示領域として配置した「水平フィールド」の左側にある「水平フィールドラベル」を選択します。



画面右側の「プロパティ」タブ - 「水平フィールドラベル」 - 「エレメント固有」 - 「label」のラジオボタンを「変数値」へと変更し、

 から「\$variable > responseData > records > 0 > question」を設定します。



「label」の入力ボックスにフォーカスし、変数文字列のrecords[0]の部分(records[\$index])へ変更します。



変数の設定では、labelの入力ボックスに直接下記の文字列を入力してもかまいません。

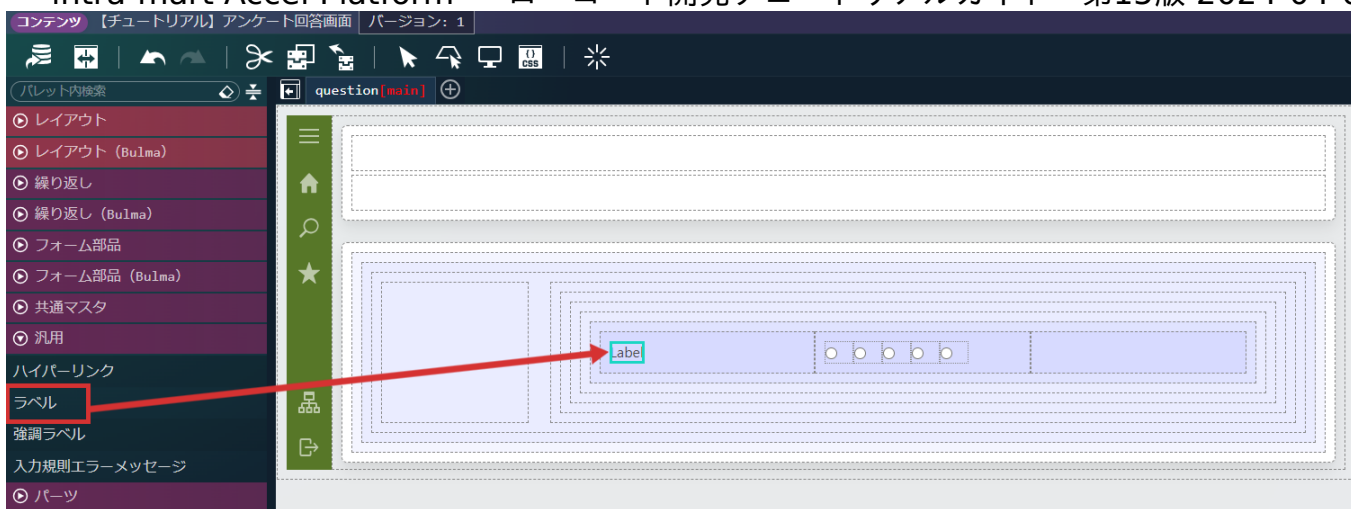
「\$variable.responseData.records[\$index].question」

これで、設問内容が表示されるようになりました。

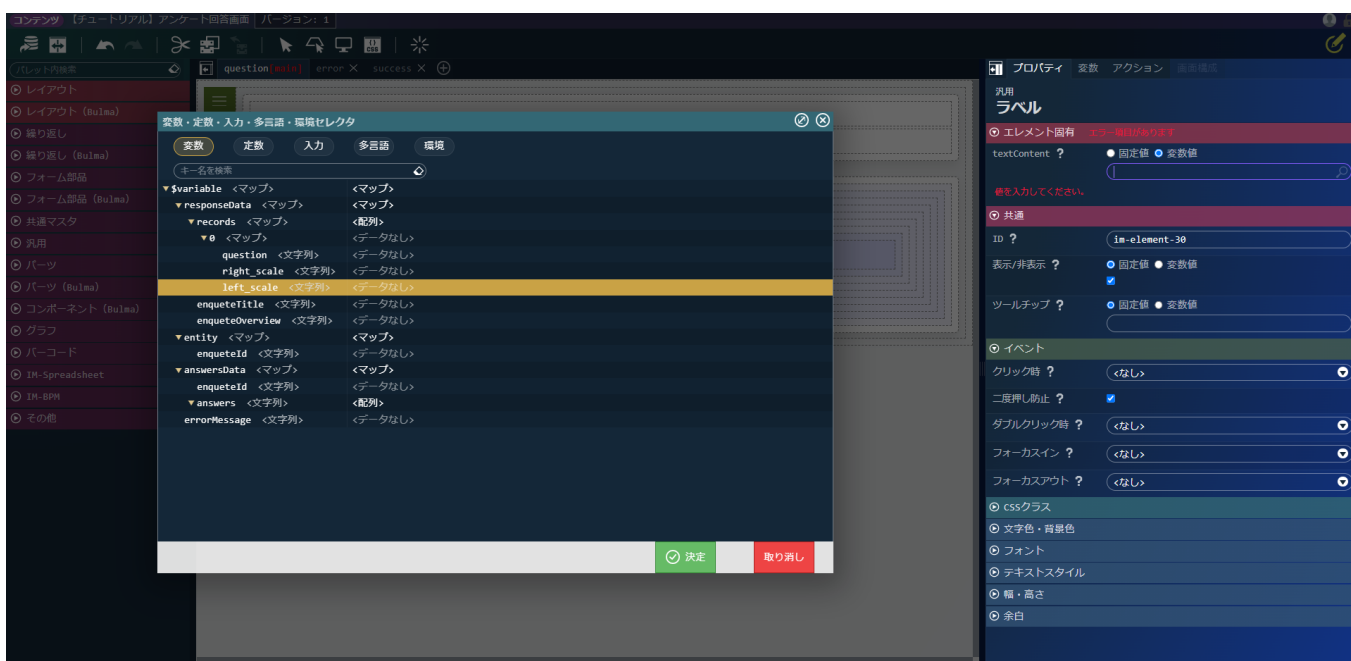
次に、各設問項目の両端を設定します。

まずは、左側の項目を設定します。

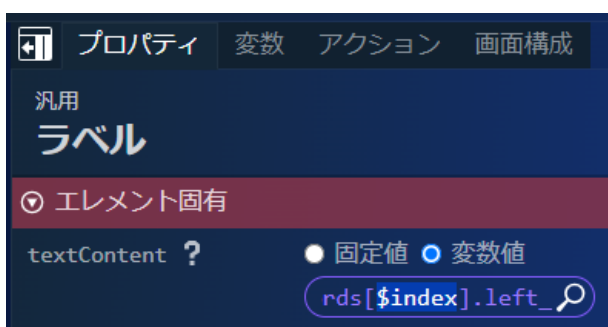
画面左側のパレットの「汎用」 - 「ラベル」を選択し、「カラム」の左カラムに配置します。



配置した「ラベル」をクリックし、画面右側の「プロパティ」タブ - 「ラベル」 - 「エレメント固有」 - 「textContent」のラジオボタンを「変数値」へと変更し、🔍 から「\$variable > responseData > records > 0 > left_scale」を設定します。



「textContent」の変数の入力ボックスにフォーカスし、変数文字列のrecords[0]の部分 records[\$index]へ変更します。

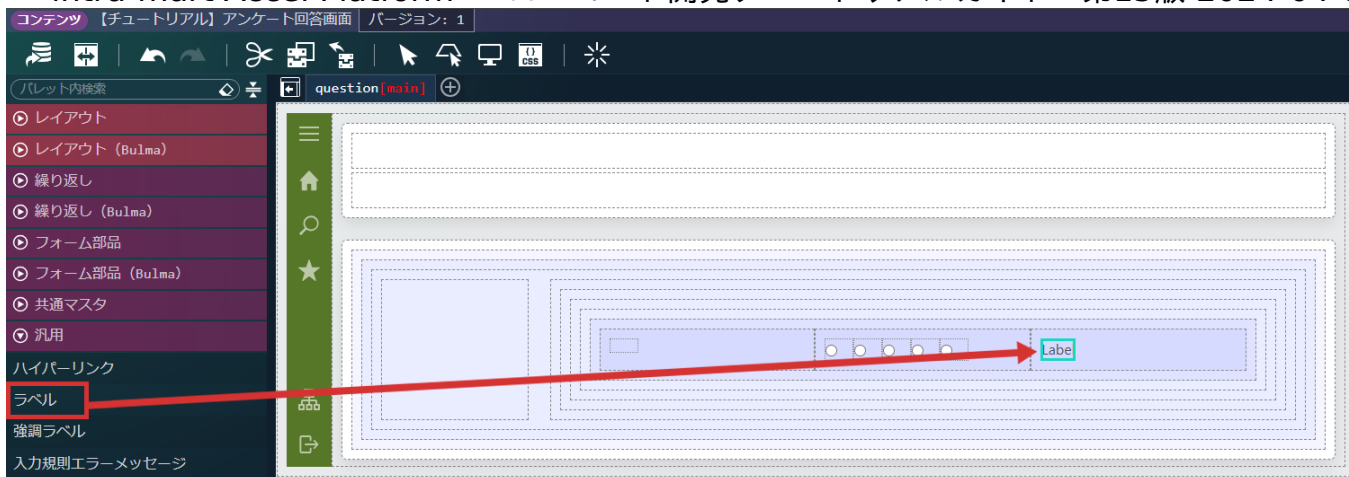



変数の設定では、textContentの入力ボックスに直接下記の文字列を入力してもかまいません。

「\$variable.responseData.records[\$index].left_scale」

同じように、右側の項目を設定します。

画面右側のパレットの「汎用」 - 「ラベル」を選択し、「カラム」の右カラムに配置します。



配置した「ラベル」をクリックし、画面右側の「プロパティ」タブ - 「ラベル」 - 「エレメント固有」 - 「textContent」のラジオボタンを「変数値」へと変更し、 から「\$variable > responseData > records > 0 > right_scale」を設定します。

「textContent」の変数の入力ボックスにフォーカスし、変数文字列のrecords[0]の部分をrecords[\$index]へ変更します。



変数の設定では、textContentの入力ボックスに直接下記の文字列を入力してもかまいません。

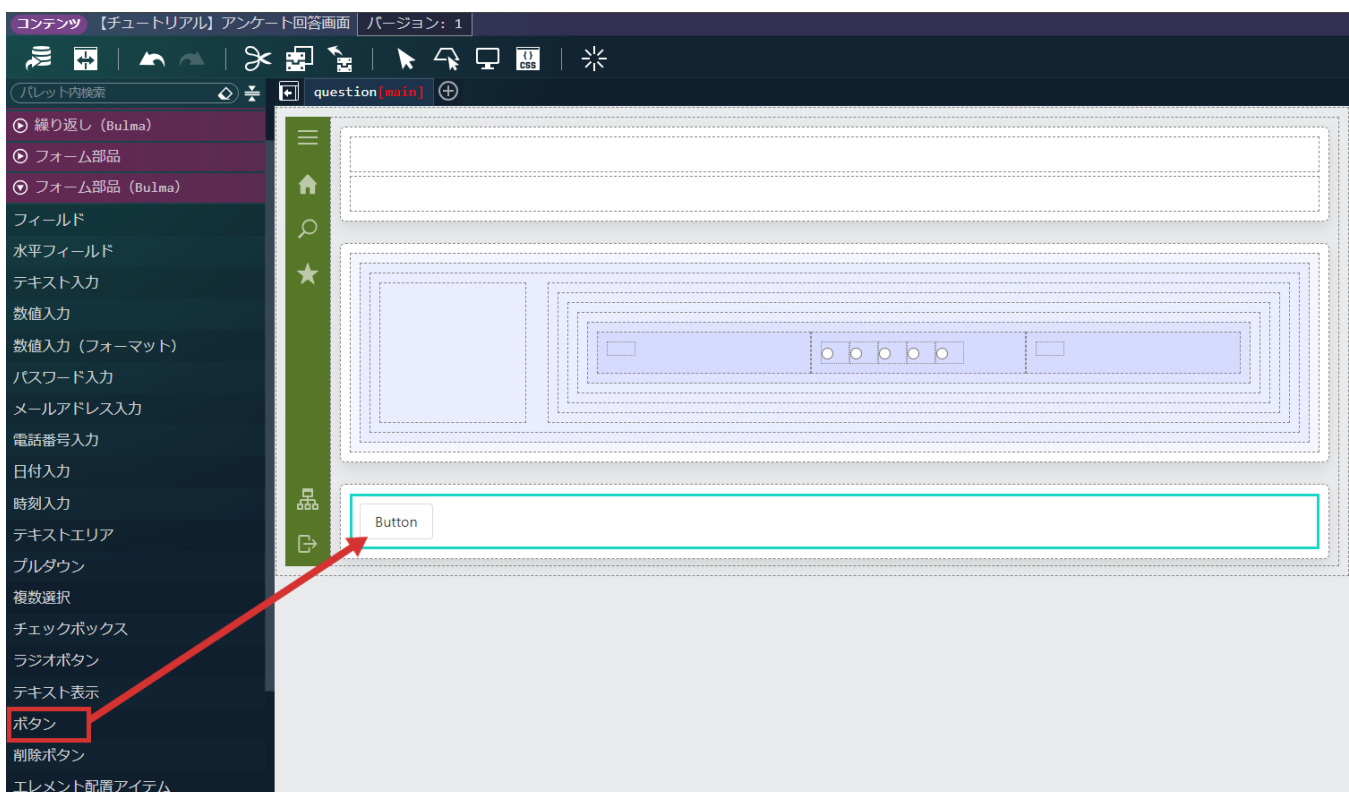
「\$variable.responseData.records[\$index].right_scale」

これで、繰り返しで設問項目を表示する領域ができました。

最後に、回答を登録するボタンを配置します。

画面左側のパレットの「レイアウト (Bulma)」 - 「ボックス」を選択し、「サイドメニューコンテナ」の中の一番下に配置します。

画面左側のパレットの「フォーム部品 (Bulma)」 - 「ボタン」を選択し、先ほど配置した「ボックス」の中に配置します。



配置した「ボタン」の「ボタン要素」をクリックし、画面右側の「プロパティ」タブ - 「ボタン要素」 - 「エレメント固有」 - 「textContent」に「回答」と設定します。



i コラム

配置したいアイテムが見つからない場合は、パレット上部にある「パレット内検索」から検索ができます。入力内容に合わせ、アイテムが絞り込まれて表示されます。

見た目の調整

続けて、見た目の調整を行います。

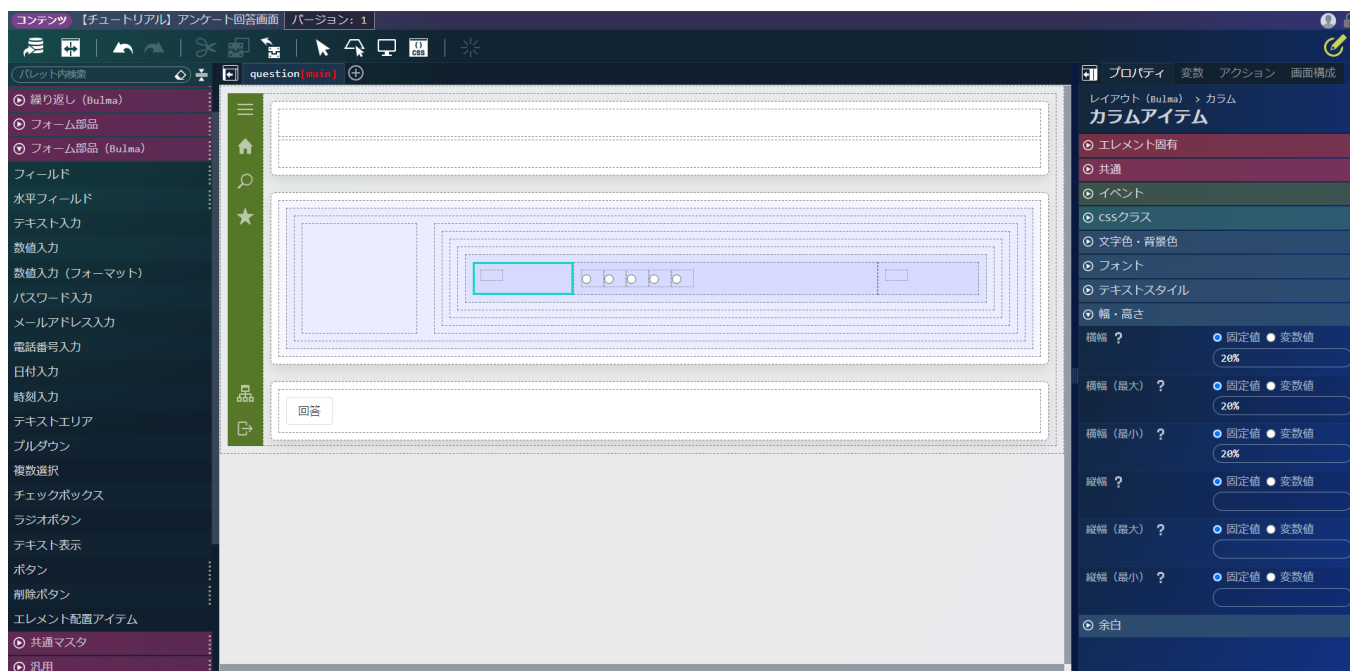
「コラム」内の左端のカラムアイテムを選択します。

「プロパティ」タブ - 「カラムアイテム」 - 「幅・高さ」 - 「横幅」に「20%」と設定します。

同じく、「横幅（最大）」、「横幅（最小）」にも「20%」と設定します。

「カラム」内の右端のカラムアイテムの「横幅」、「横幅（最大）」、「横幅（最小）」も同じく「20%」と設定します。

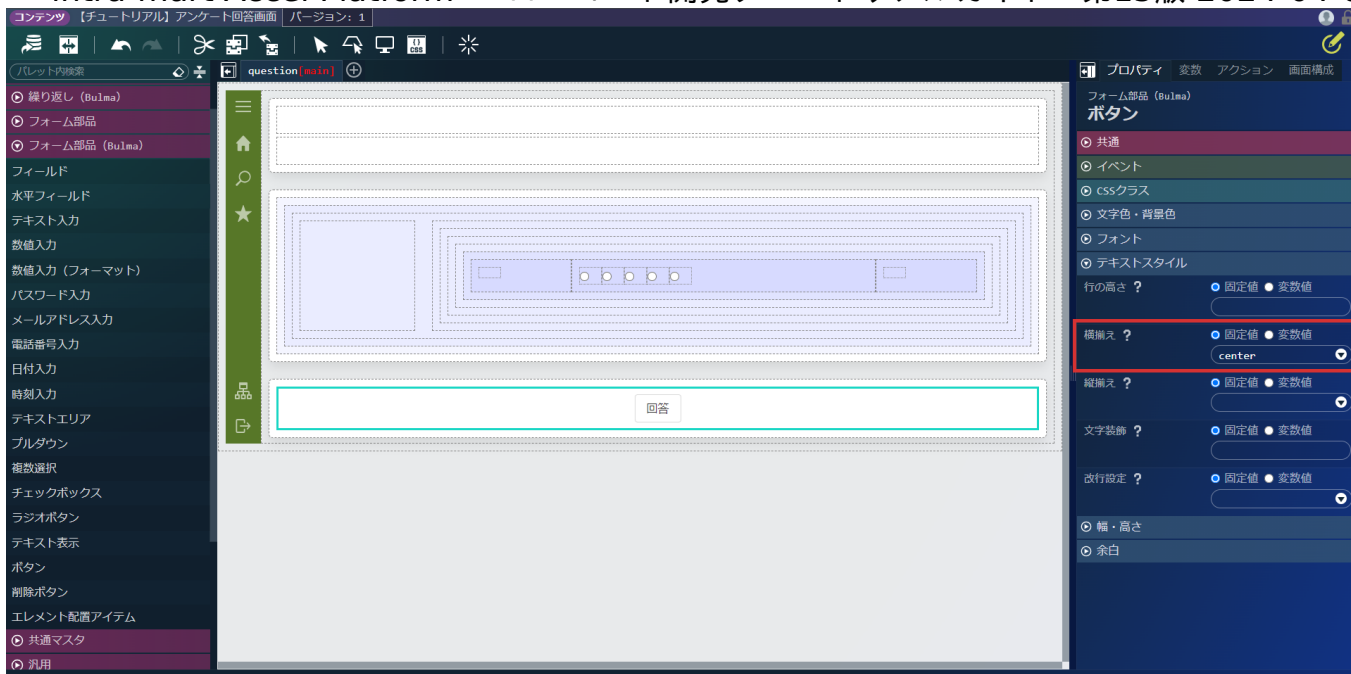
設定が完了した時点での表示領域は以下の図のような見た目です。



次に、登録ボタンの配置を変更します。


登録ボタンを配置している「ボックス」を選択します。

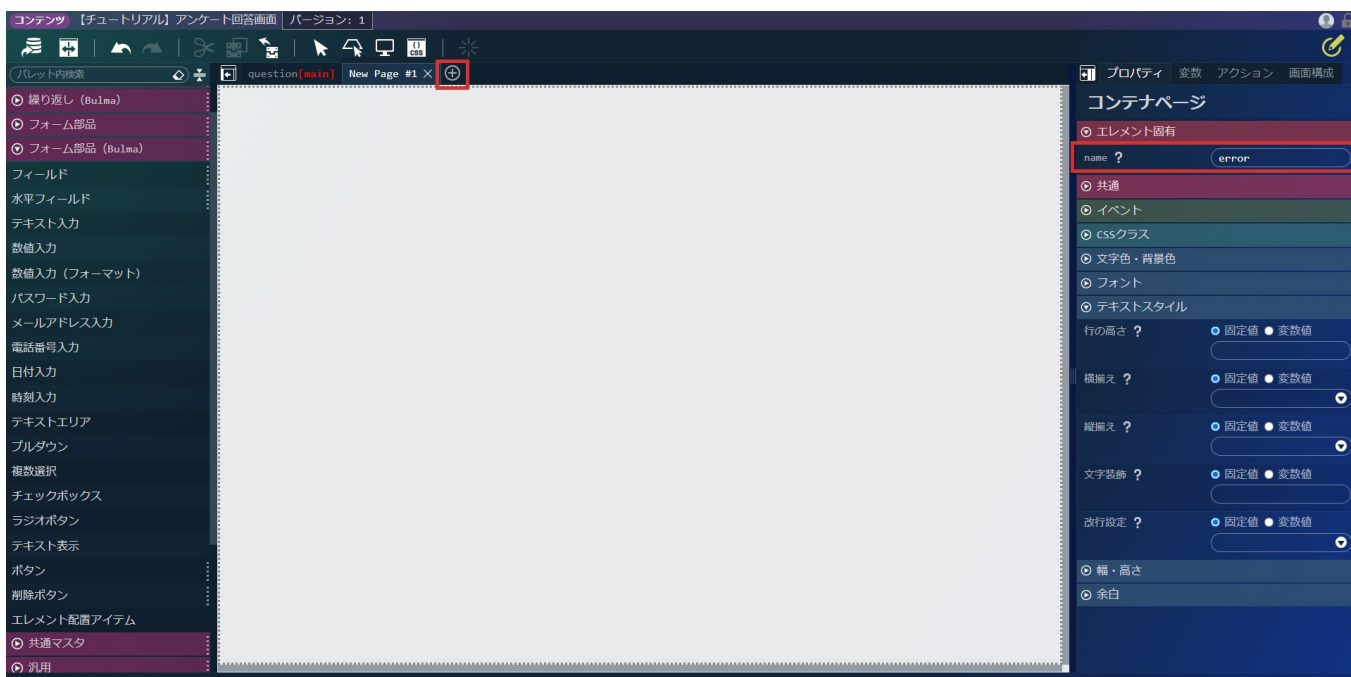
「プロパティ」タブ - 「ボックス」 - 「テキストスタイル」 - 「横揃え」に「center」と設定します。



エラー画面の作成

アンケートIDが指定されなかった場合や、誤ったIDが渡ってくることによってアンケート情報が取得できなかった場合に表示するエラー画面を作成します。

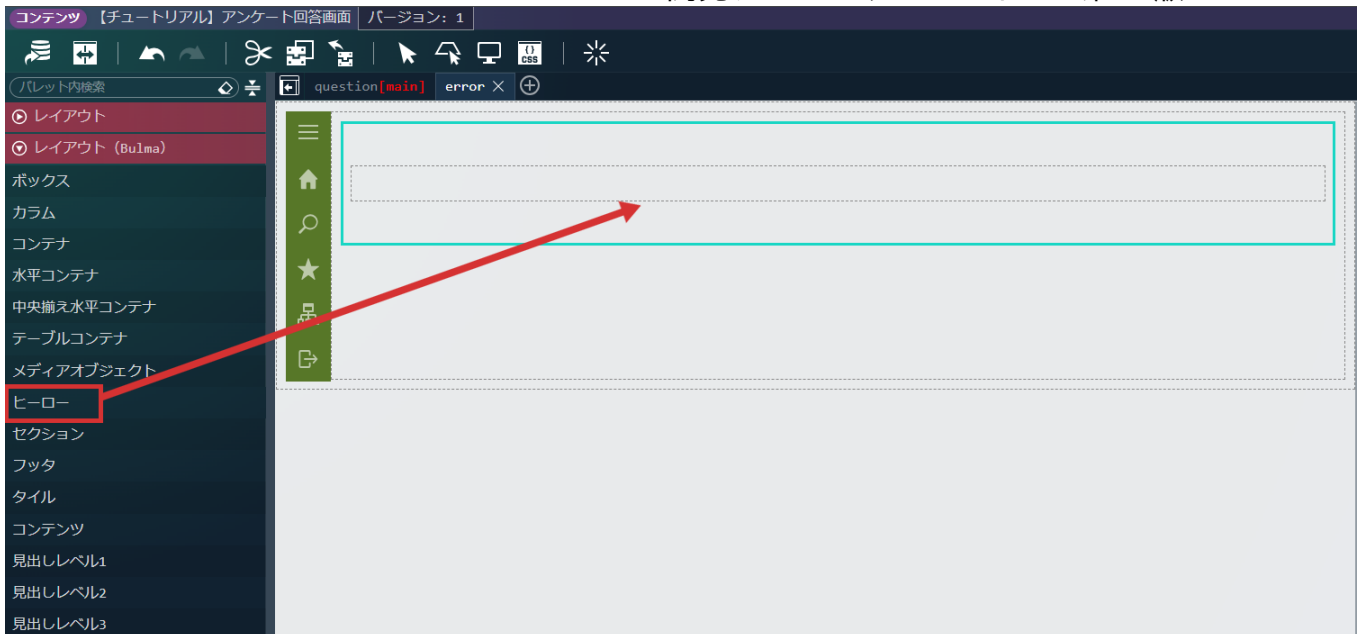
画面上部の  をクリックし、「プロパティ」タブ - 「コンテナページ」 - 「エレメント固有」 - 「name」を「error」と設定します。



画面左側のパレットの「レイアウト」 - 「サイドメニューコンテナ」を選択し、画面に配置します。

アンケートのエラーメッセージの表示領域を作成します。

画面左側のパレットの「レイアウト (Bulma)」 - 「ヒーロー」を選択し、先ほど配置した「サイドメニューコンテナ」の中に配置します。



画面左側のパレットの「レイアウト (Bulma)」 - 「見出しレベル1」を選択し、先ほど配置した「ヒーロー」の「ヒーローボディ」の中に配置します。

配置した「見出しレベル1」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「textContent」のラジオボタンを「変数値」へと変更し、🔍 から「\$variable > errorMessage」を設定します。




変数の設定では、textContentの入力ボックスに直接下記の文字列を入力してもかまいません。

「\$variable.errorMessage」

回答成功画面の作成

アンケートへ正常に回答できた場合に表示する画面を作成します。

画面上部の  をクリックし、「プロパティ」タブ - 「コンテンツページ」 - 「エレメント固有」 - 「name」を「success」と設定しま

す。

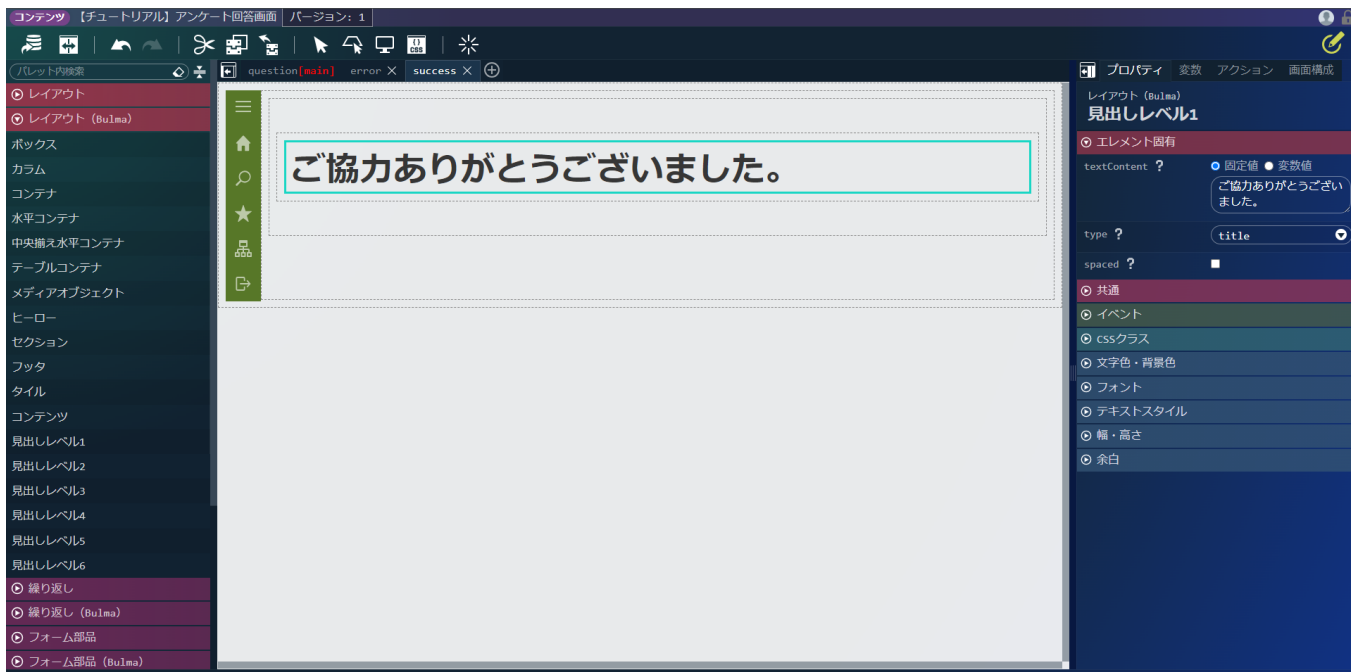
画面左側のパレットの「レイアウト」 - 「サイドメニューコンテナ」を選択し、画面に配置します。

次に、アンケートのタイトルと詳細の表示領域を作成します。

画面左側のパレットの「レイアウト (Bulma)」 - 「ヒーロー」を選択し、先ほど配置した「サイドメニューコンテナ」の中に配置します。

画面左側のパレットの「レイアウト (Bulma)」 - 「見出しレベル1」を選択し、先ほど配置した「ヒーロー」の中に配置します。

配置した「見出しレベル1」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「textContent」の「固定値」として「ご協力ありがとうございました。」と設定します。



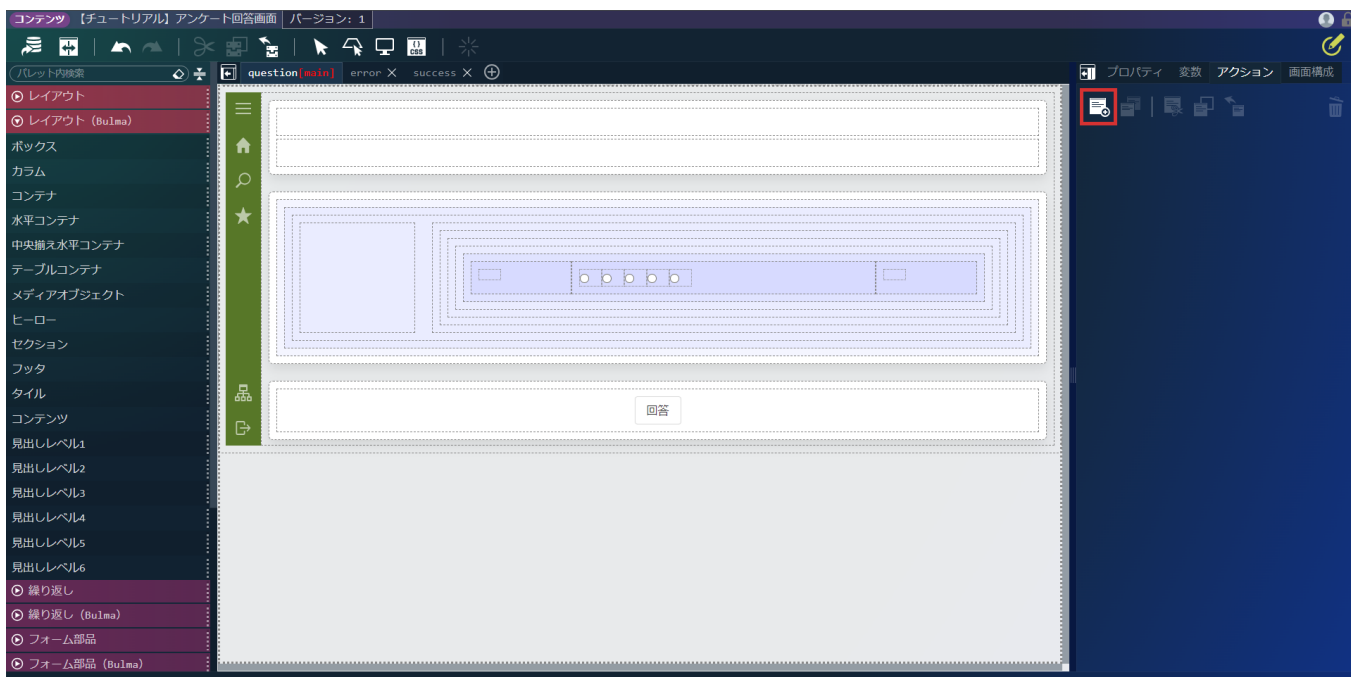
初期表示アクションの設定

次に、画面で実行されるアクションを設定します。

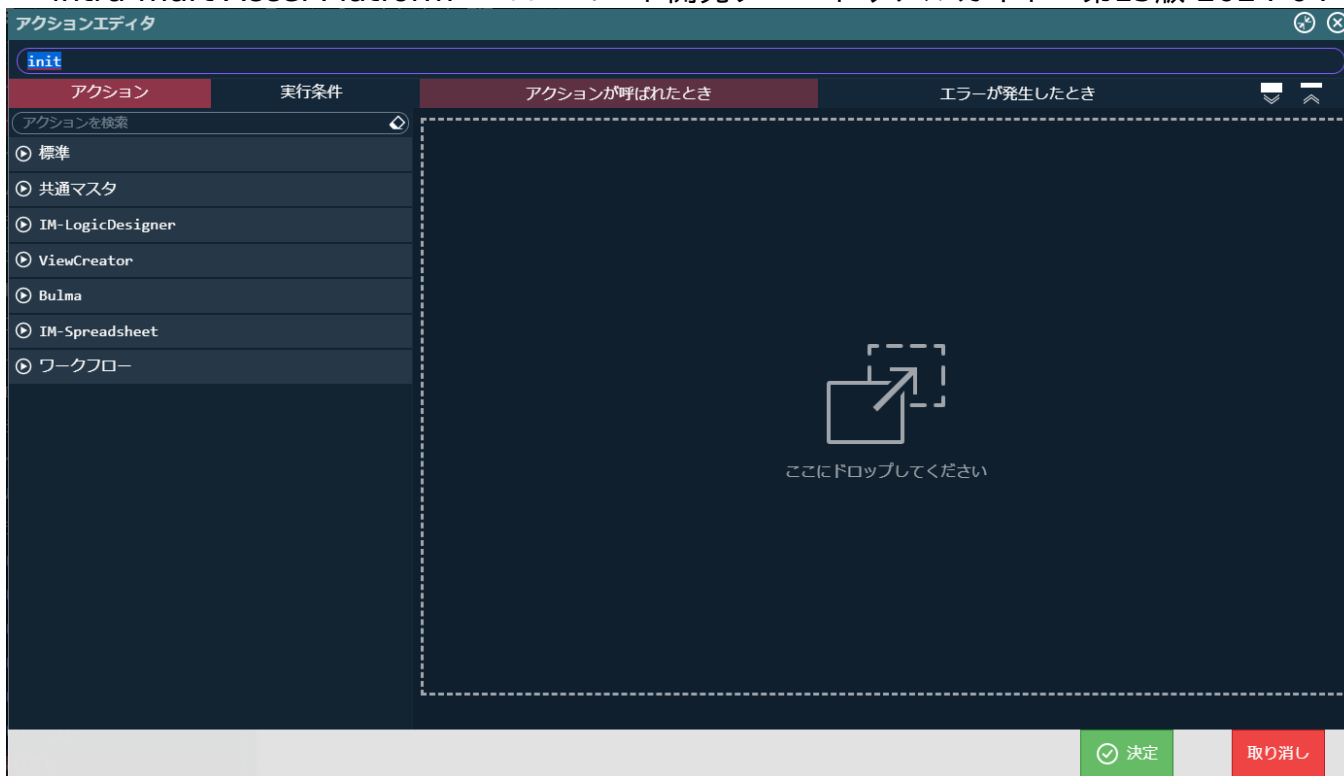
アクションは初期表示時に実行されるアクションと、回答の登録時に実行されるアクションの2種類作成します。

まずは、初期表示アクションから作成していきます。

画面右側の「アクション」タブを選択し、 をクリックしてアクションエディタを表示します。

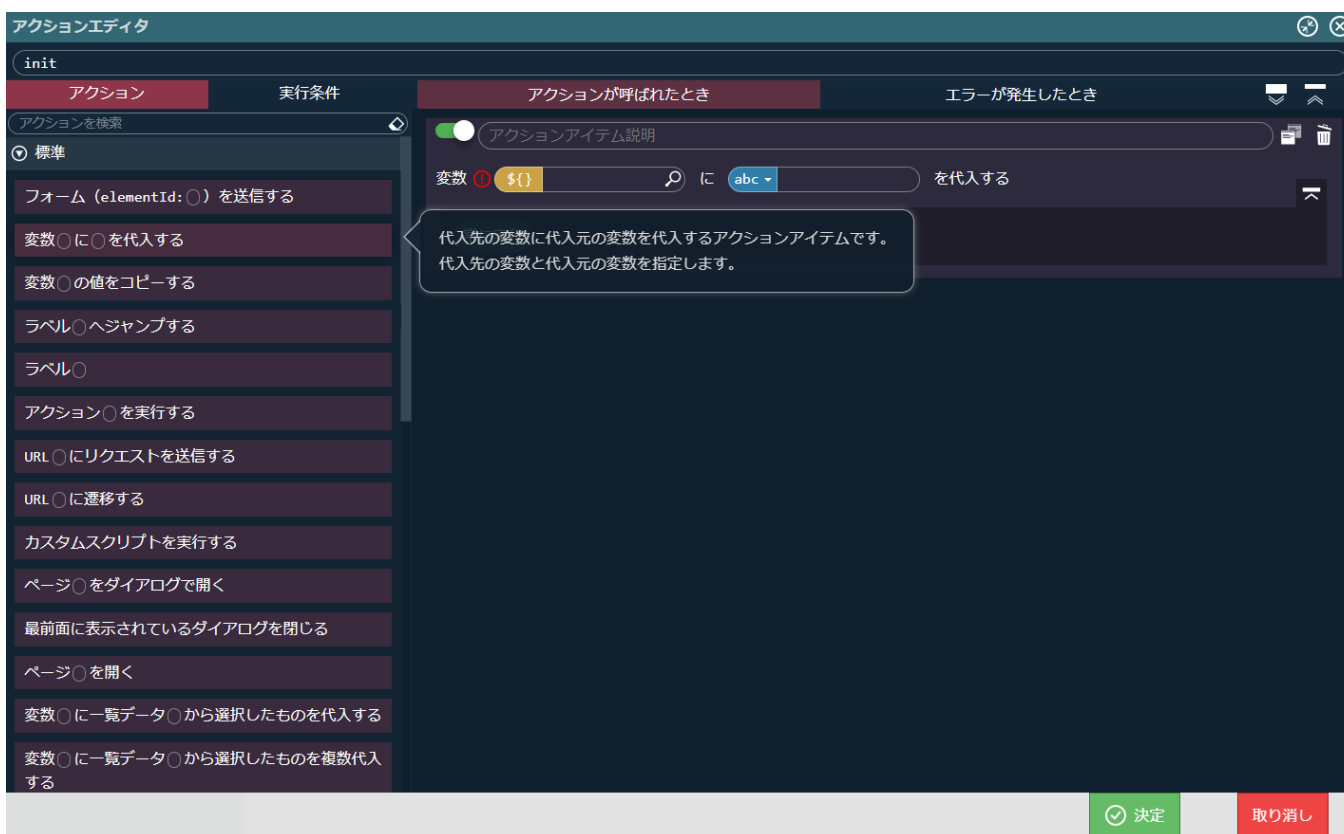



アクションエディタ上部の名称を「新しいアクション #1」から「init」に変更します。



次に、設問を取得するアクションを作成します。


「アクション」タブから「標準」 - 「変数〇に〇を代入する」を選択し、「アクションが呼ばれたとき」に配置します。

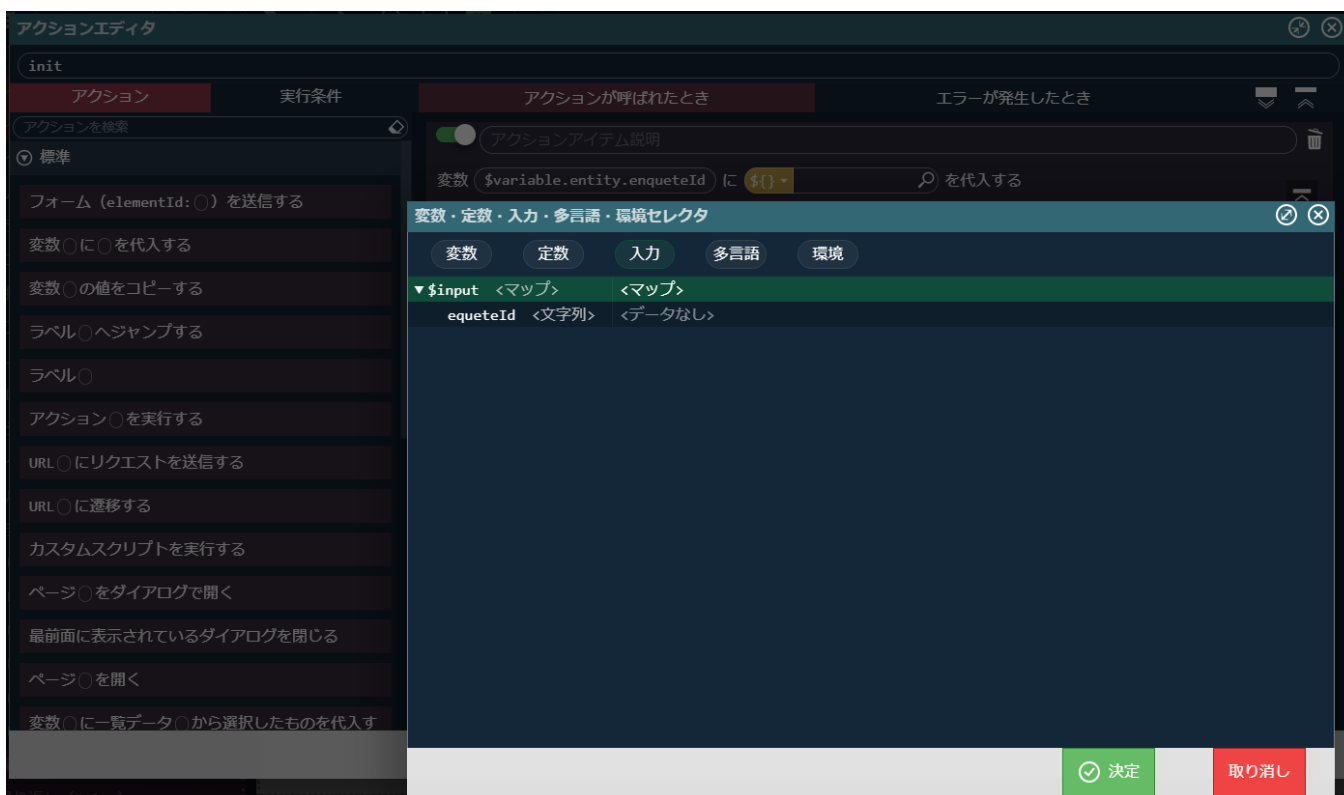


配置したアクションの変数の  から「変数 > \$variable > entity > enqueteld」を選択し、「決定」をクリックします。



代入値の「abc▼」をクリックし、入力値の種類を固定値から変数値へと変更します。

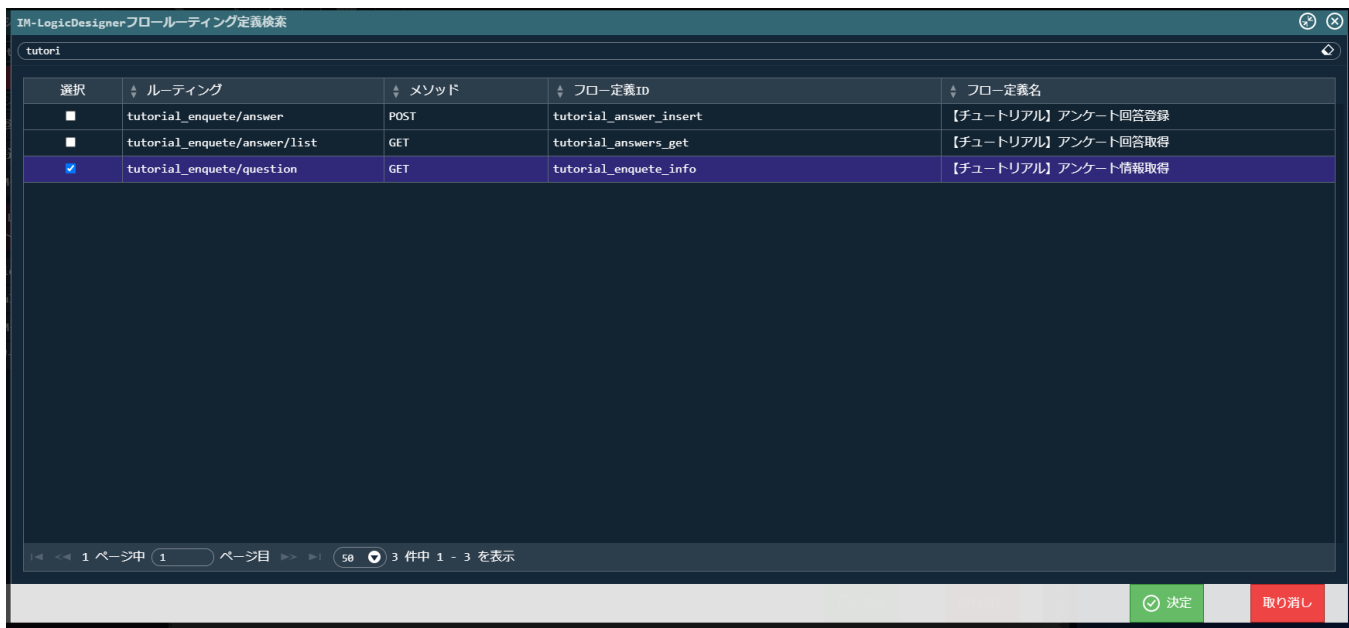
 から「入力 > \$input > enqueteId」を選択します。



左側の「アクション」タブから「IM-LogicDesigner」 - 「IM-LogicDesigner フロールーティング○にリクエストを送信する」を選択し、先ほど配置した「変数○に○を代入する」の次に配置します。



配置した「IM-LogicDesigner フロールーティング〇にリクエストを送信する」の 🔍 をクリックし、IM-LogicDesigner フロールーティング定義検索ダイアログから「tutorial_enquete/question」を選択します。



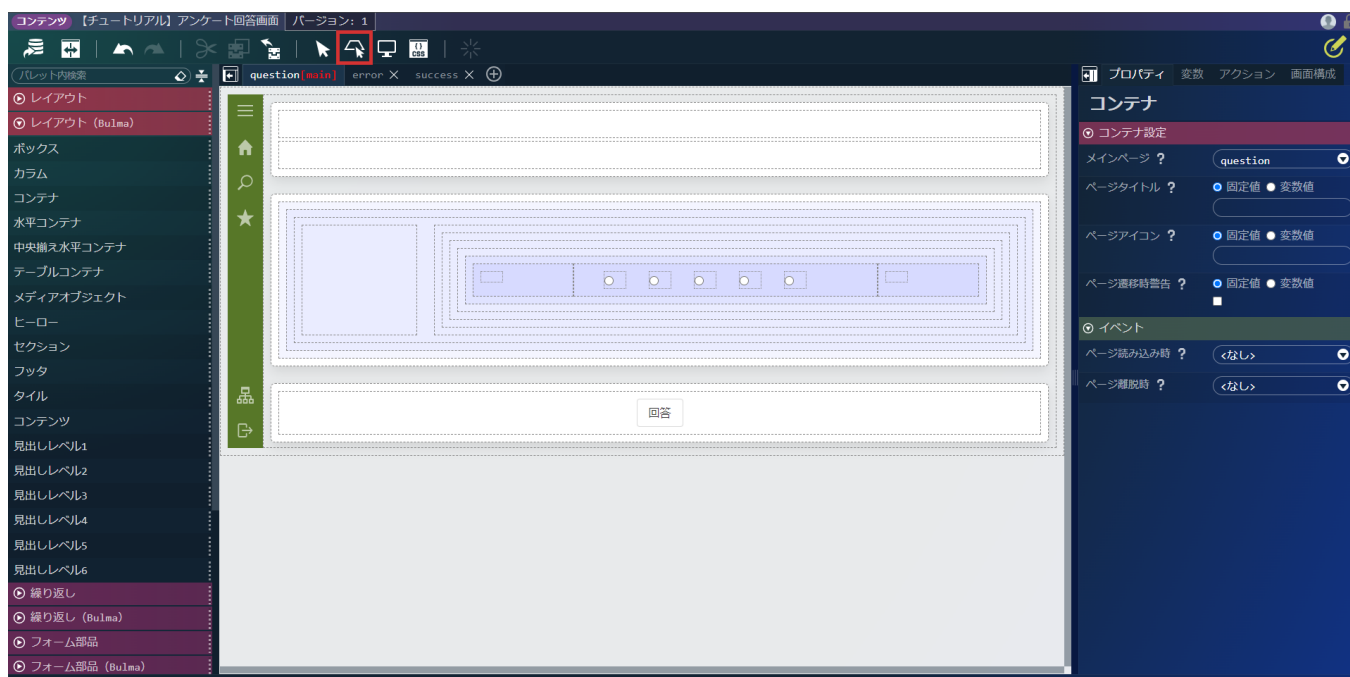
設定した業務ロジックにアンケートIDを渡すために、リクエストパラメータの 🔍 から「変数 > \$variable > entity」を設定します。業務ロジックからデータを受け取るために、レスポンスデータの 🔍 から「変数 > \$variable > responseData」を設定します。



ここまでで、アンケート情報を取得し画面に表示できるようになりました。一度、動作確認を行きましょう。アクションエディタ下部の決定ボタンをクリックし、ここまで作成したアクション処理を登録します。

「init」アクションを画面の初期表示時に呼び出す設定を行います。


画面上部の  をクリックします。



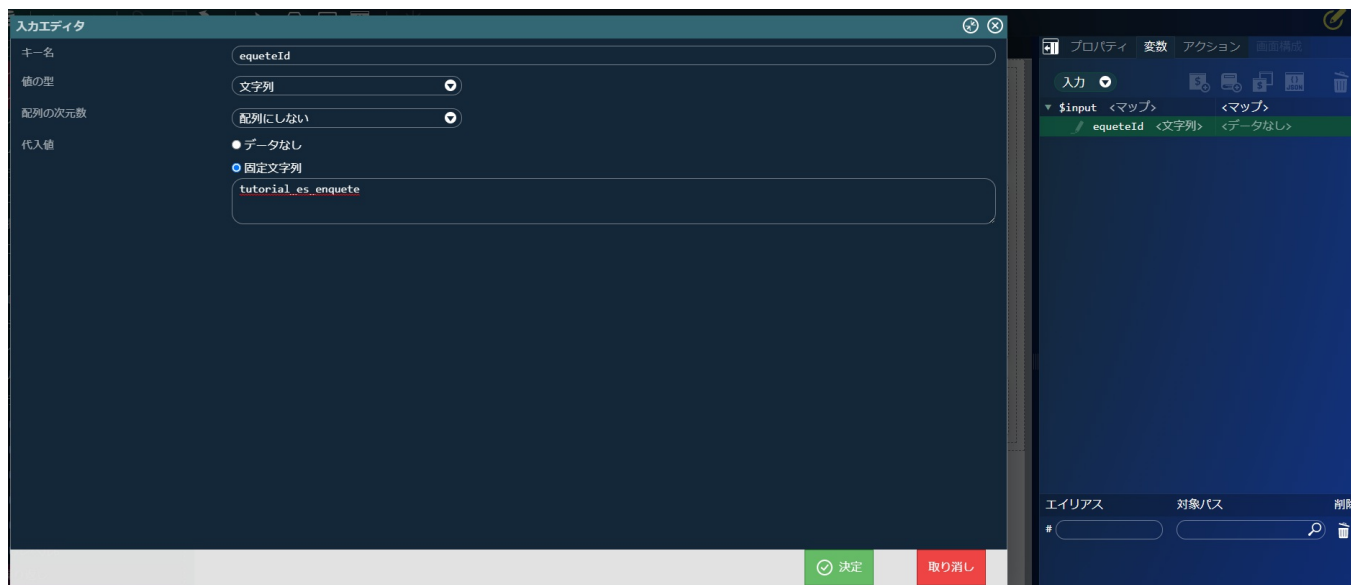
画面右側の「プロパティ」タブ - 「コンテナ」 - 「イベント」の「ページ読み込み時」に「init」アクションを設定します。




次に、試験用のデータを設定するため、画面右側の「変数」タブを選択し、プルダウンから「入力」の表示に切り替えます。

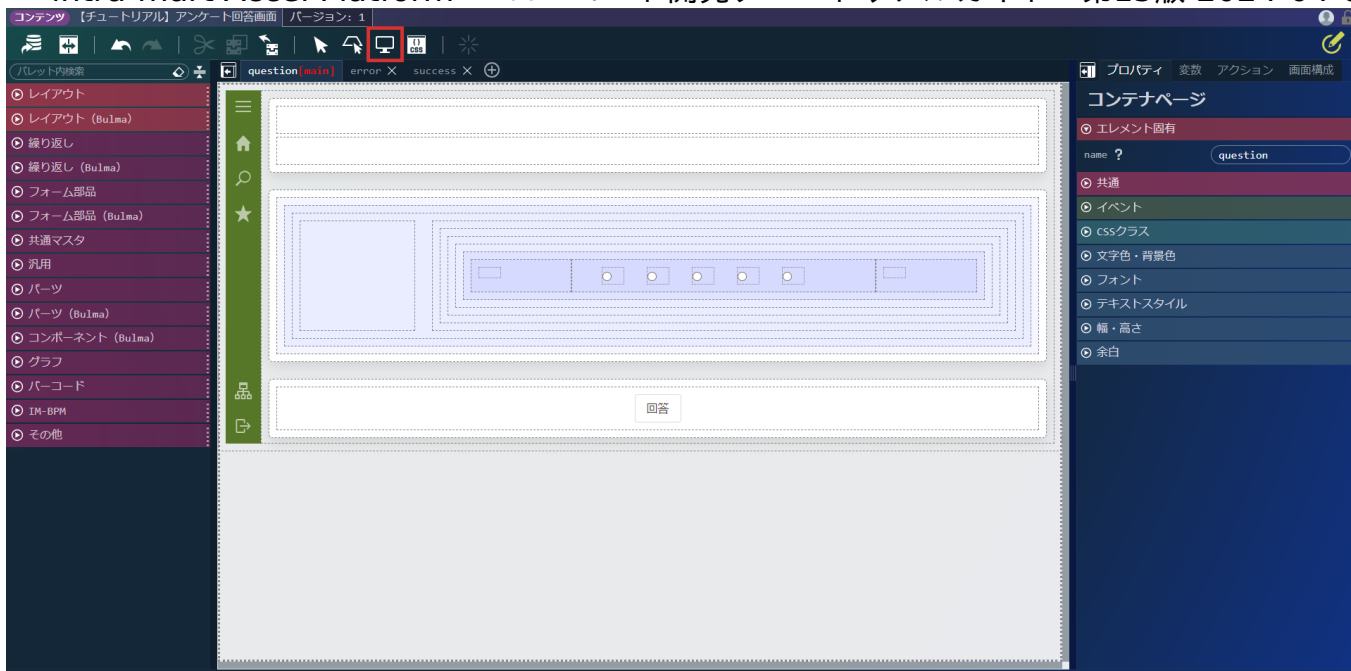
「\$input > enqueteId」の  をクリックし、代入値のラジオボタンを「固定文字列」に切り替えて「tutorial_es_enquete」と設定します。

※概要で設定したサンプルデータです。サンプルデータが存在しない場合は、tutorial_enquete_question_appテーブルにデータを登録してください。

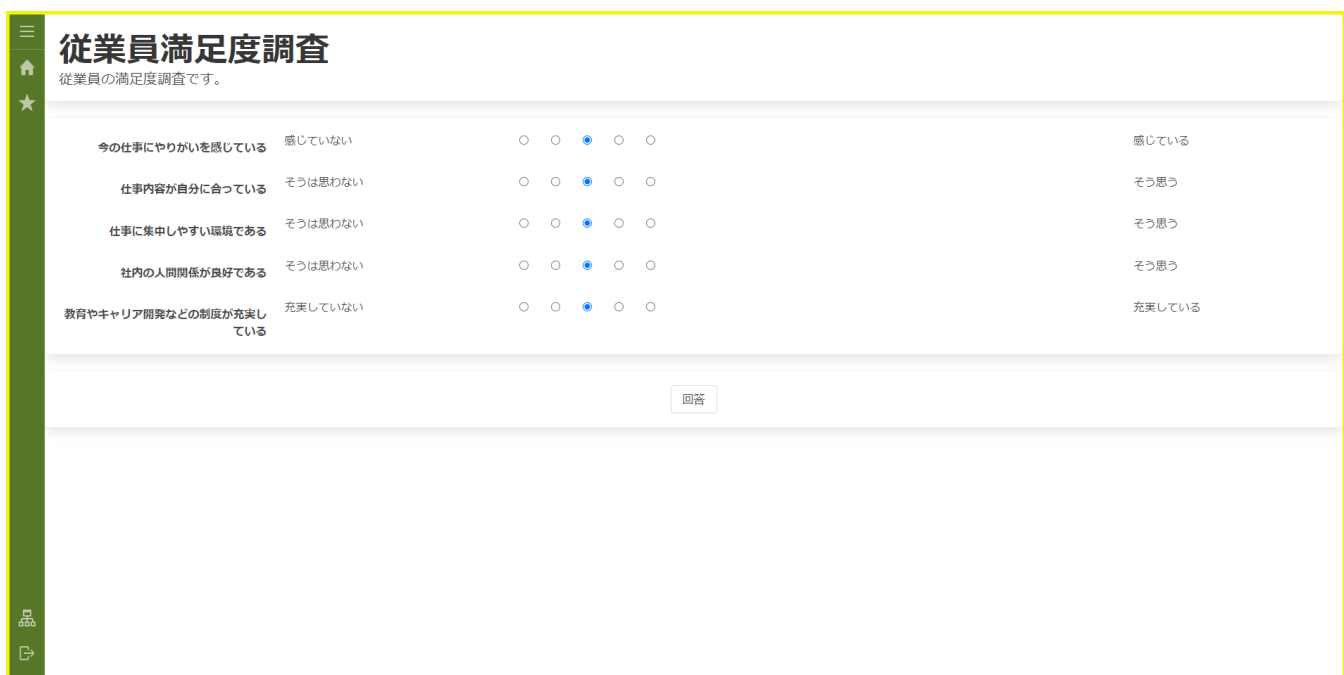


ここまでで、アンケート情報を取得し画面に表示できるようになりました。一度、動作確認を行いましょう。

入力値を登録したら、画面上部の  をクリックします。



プレビューダイアログが表示され、設問内容が表示されました。



アクション設定を続けます。

画面右側の「アクション」タブを選択し、「init」アクションの  をクリックしてアクションエディタを表示します。

回答を入れるための配列を作成します。また、初期値として中心の3が選択されるようにします。

「アクション」タブから「標準」 - 「カスタムスクリプトを実行する」を選択し、「IM-LogicDesigner フロールルーティング〇にリクエストを送信する」の次に配置します。

配置した「カスタムスクリプトを実行する」の入力欄に以下のスクリプトを設定し、「決定」をクリックしてください。これで初期表示時のアクション設定が完了です。

```
for(let i = 0; i < $variable.responseData.records.length; i++){
  $variable.answersData.answers.push(3);
}
```

回答登録アクションの設定

次に、回答を登録する際のアクション処理を作成します。

画面右側の「アクション」タブを選択し、 をクリックし、アクションエディタを表示します。

アクションエディタ上部の名称を「新しいアクション #1」から「answer」に変更します。

「アクション」タブから「標準」 - 「変数〇に〇を代入する」を選択し、「アクションが呼ばれたとき」に配置します。



配置したアクションの変数を以下のように設定します。

変数（`$variable.answersData.enqueteld`）に（`$variable.entity.enqueteld`）を代入する



左側の「アクション」タブから「IM-LogicDesigner」 - 「IM-LogicDesigner フロールーティング〇にリクエストを送信する」を選択し、先ほど配置した「変数〇に〇を代入する」の次に配置します。

配置した「IM-LogicDesigner フロールーティング〇にリクエストを送信する」の 🔍 をクリックし、IM-LogicDesigner フロールーティング定義検索ダイアログから「tutorial_enquete/answer」を選択します。

選択	ルーティング	メソッド	フロー定義ID	フロー定義名
<input checked="" type="checkbox"/>	tutorial_enquete/answer	POST	tutorial_answer_insert	[チュートリアル] アンケート回答登録
<input type="checkbox"/>	tutorial_enquete/answer/list	GET	tutorial_answers_get	[チュートリアル] アンケート回答取得
<input type="checkbox"/>	tutorial_enquete/question	GET	tutorial_enquete_info	[チュートリアル] アンケート情報取得

1 ページ中 1 ページ目 50 3 件中 1 - 3 を表示

決定 取り消し

設定した業務ロジックにアンケートIDを渡すために、リクエストパラメータに「\$variable > answersData」を設定します。

answer

アクション 実行条件 アクションが呼ばれたとき エラーが発生したとき

アクションアイテム説明

変数 $\{\}$ \$variable.answ に $\{\}$ \$variable.ent を代入する

常に実行

アクションアイテム説明

IM-LogicDesigner フロールーティング tutorial_enquete/answer にリクエストを送信する

リクエストパラメータ IN $\{\}$ \$variable.answersData

リクエストヘッダ IN $\{\}$

ステータスコード OUT $\{\}$

レスポンスデータ OUT $\{\}$

セキュアトークンを送信する

エポックミリ秒を送信しない

常に実行

決定 取り消し

登録完了時に「登録完了画面」が表示されるようにします。

「アクション」タブから「標準」 - 「ページ〇を開く」を選択し、一番最後のアクションとして配置します。

配置したアクションを以下のように設定します。

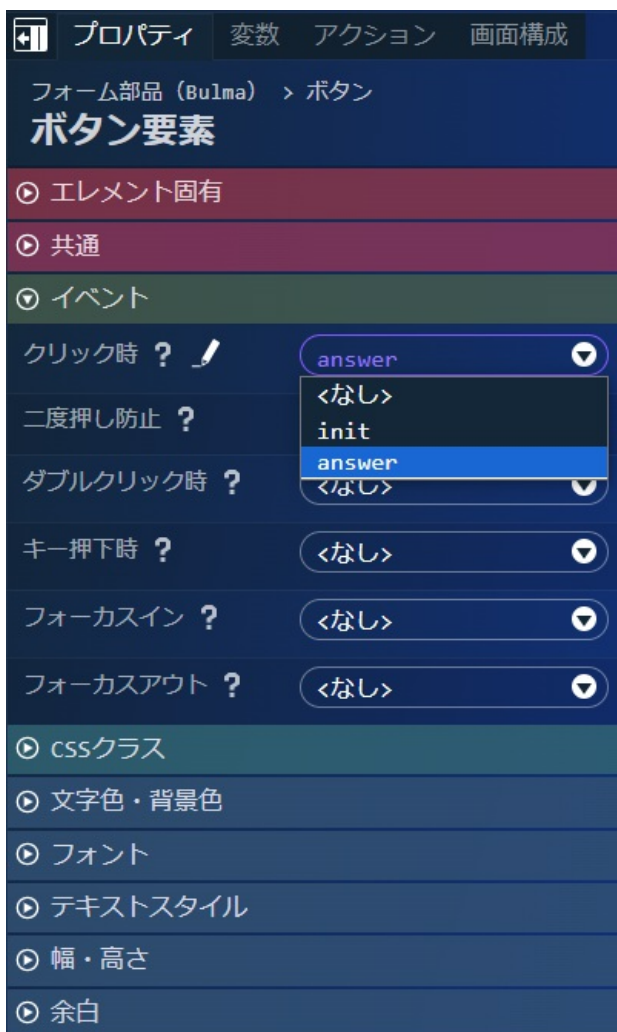
ページ(success)を開く



ここまで設定できたら「決定」をクリックし、アクションエディタを閉じます。


回答ボタンに回答登録処理を設定します。

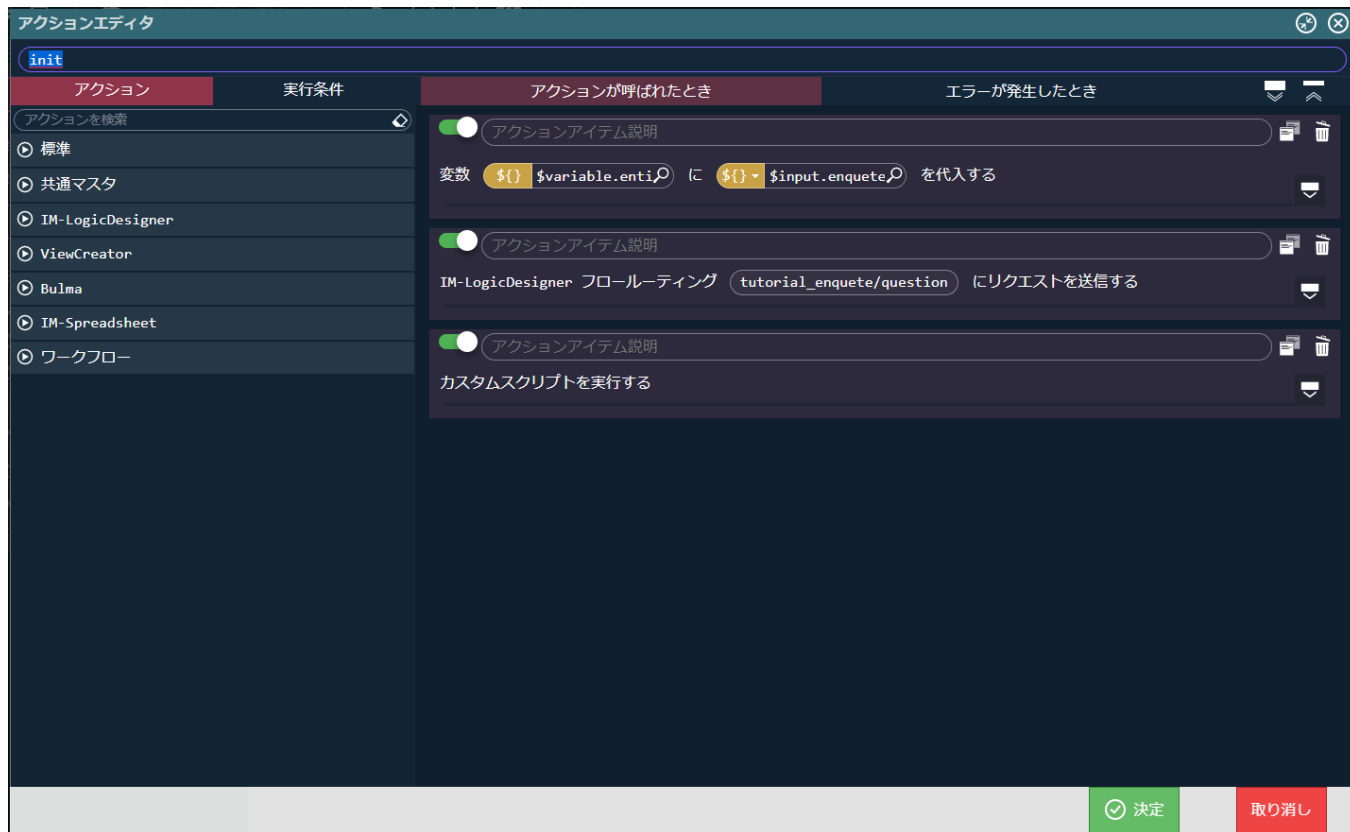
回答ボタンを選択し、「プロパティ」タブ - 「ボタン要素」 - 「イベント」 - 「クリック時」に「answer」アクションを設定します。



次に、アクション処理にエラー処理を追加します。

画面上でのエラー処理としては、画面を表示する際にアンケートIDが指定されなかった場合と、誤ったアンケートIDでアンケート情報が取得できなかった場合の2通りに対しエラーを作成します。

どちらのエラー処理もinitアクションに対して設定するので、「アクション」タブから「init」アクションの  をクリックし、アクションエディタを表示してください。



まずは、アンケートIDが指定されなかった場合のエラーを設定します。

エラー画面のメッセージを設定するため、「アクション」タブから「標準」 - 「変数〇に〇を代入する」を選択し、「アクションが呼ばれたとき」の一番上に配置します。

配置したアクションの変数を以下のように設定します。

変数 (\$variable.errorMessage) に (\$constant.error_no_id) を代入する



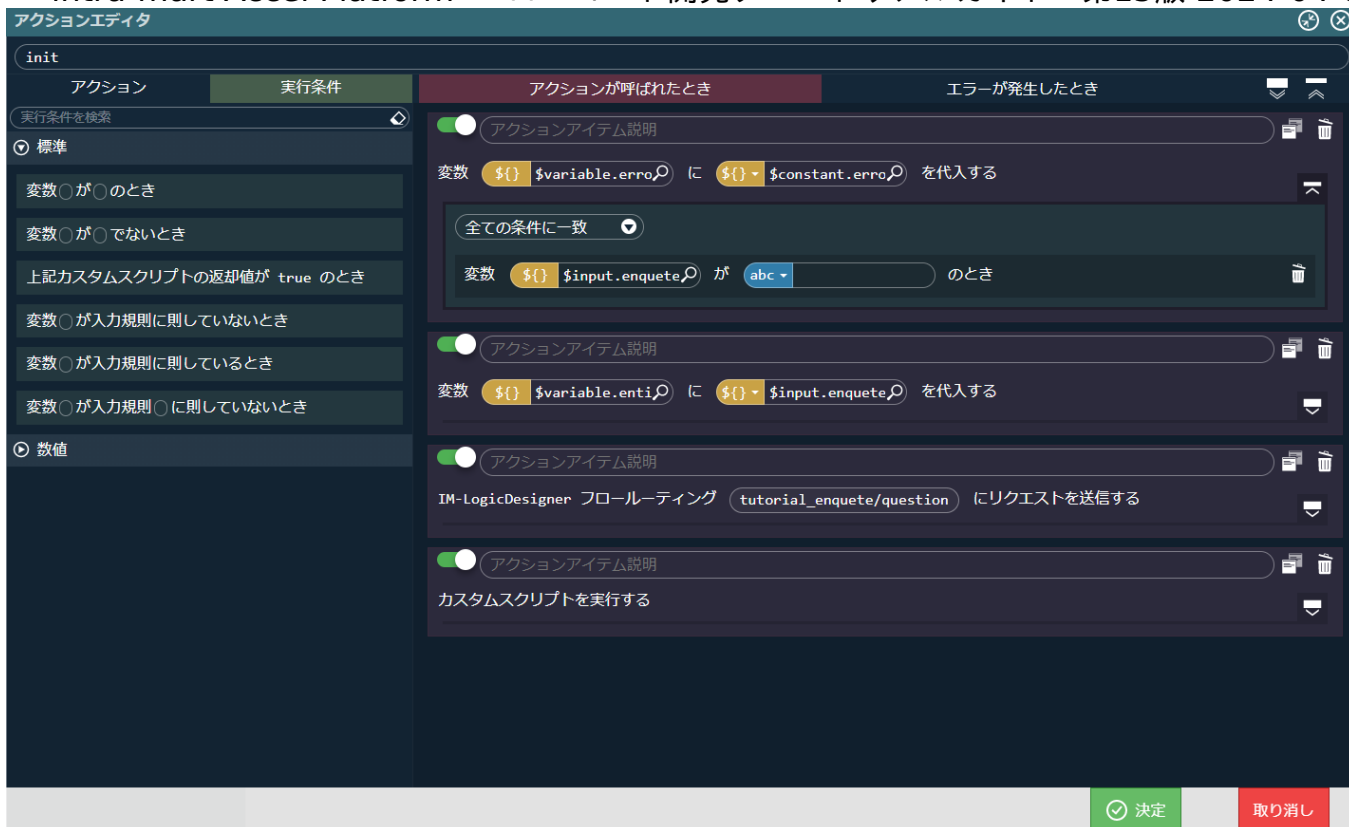
アクションエディタ左側の実行条件タブから「標準」 - 「変数〇が〇のとき」を選択肢、直前で編集していたアクションに被せるようにドラッグ&ドロップして実行条件を追加します。



配置した条件を以下のように設定します。

変数 (\$input.enqueteld) が () のとき

「〇のとき」を空にすることによって、入力値のenqueteldが指定されなかった場合の条件が指定できます。



次に、エラー画面の表示を行います。

「アクション」タブから「標準」 - 「ページ〇を開く」を選択し、先ほど設定したアクションの次に配置します。

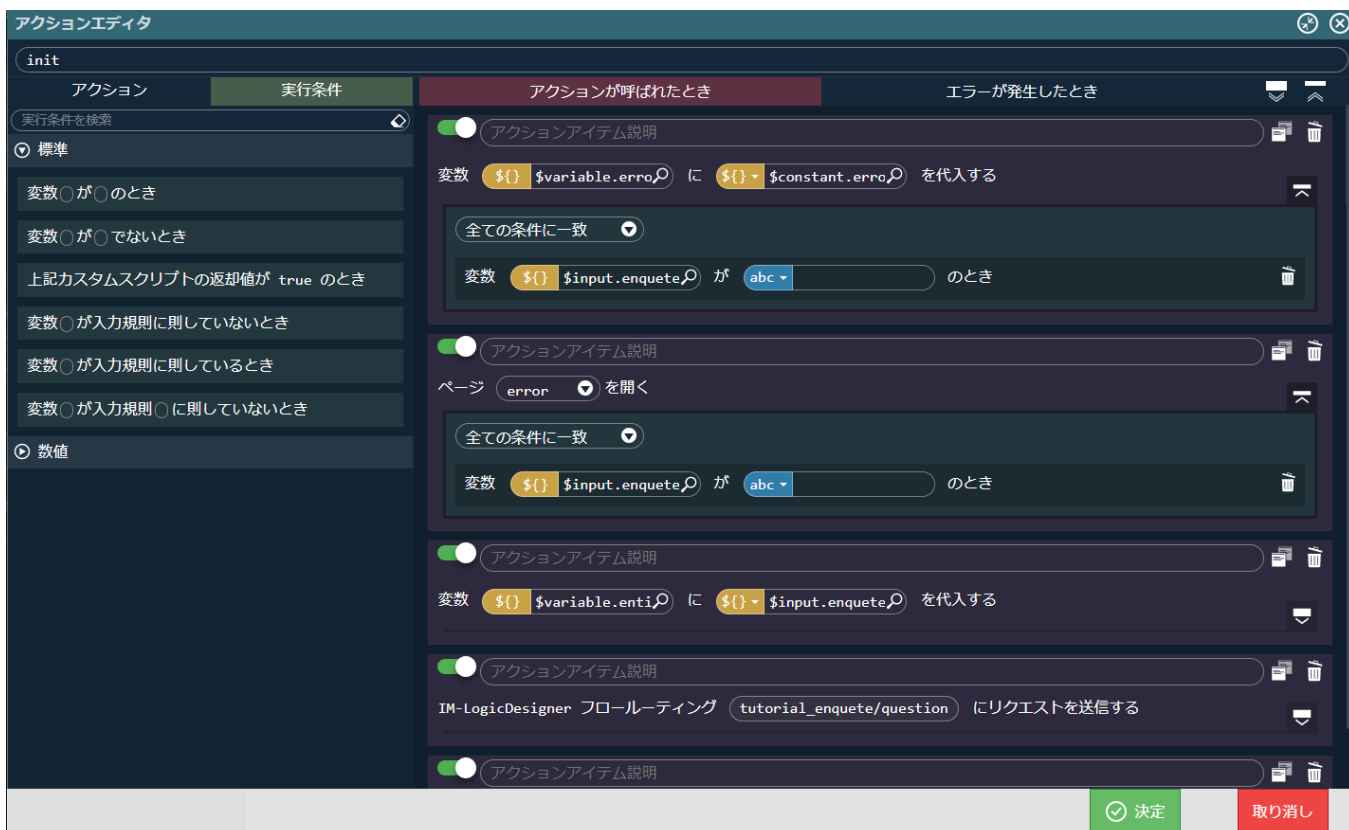
配置したアクションを以下のように設定します。

ページ(error)を開く

アクションエディタ左側の実行条件タブから「標準」 - 「変数〇が〇のとき」を直前で編集していたアクションに追加します。

配置した条件を以下のように設定します。

変数 (\$input.enqueteld) が () のとき



エラーが発生した際に、後続の処理が行われないよう設定します。

「アクション」タブから「標準」 - 「ラベル〇」を選択し、一番最後のアクションとして配置します。



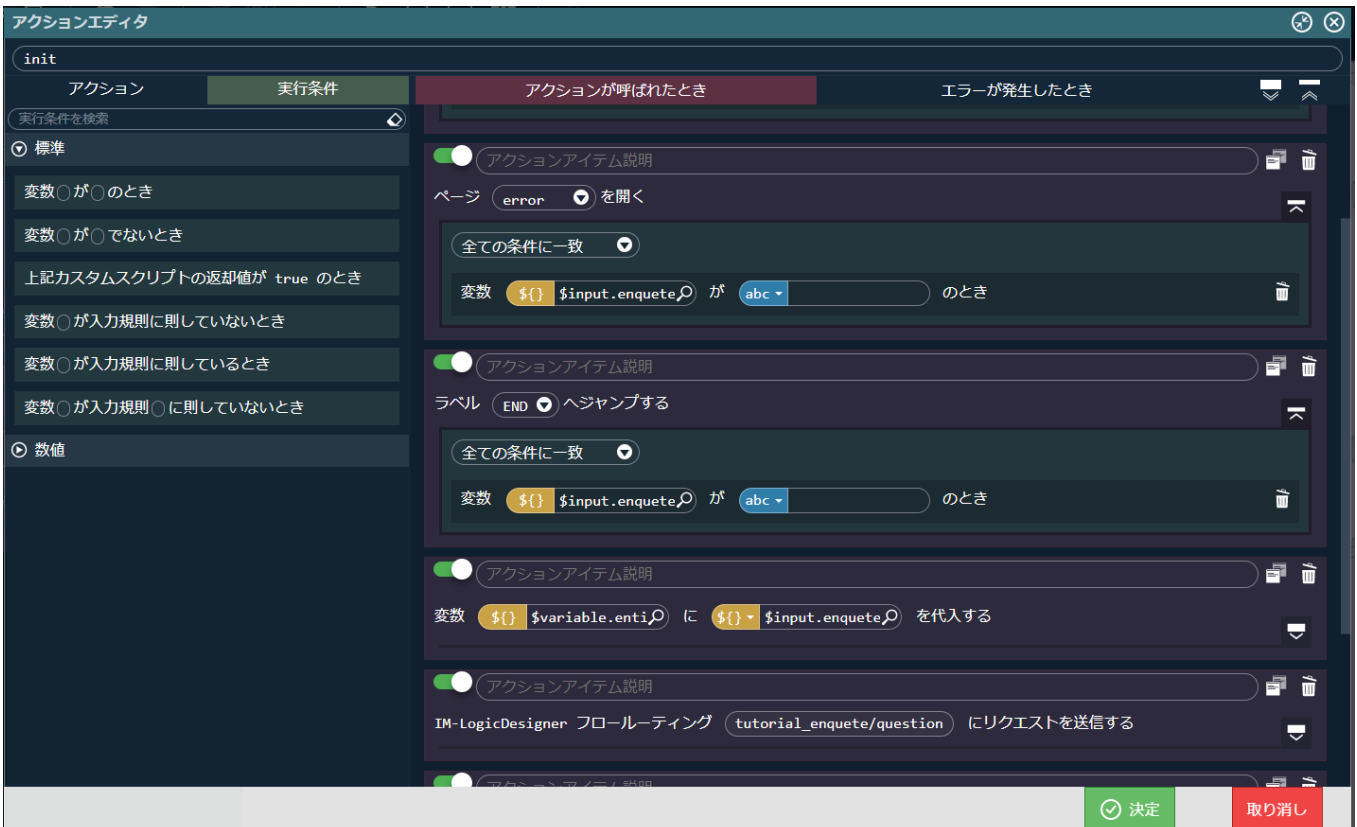
「アクション」タブから「標準」 - 「ラベル〇へジャンプする」を選択し、「ページ (error) を開く」の次のアクションとして配置します。

「ラベル (END) へジャンプする」と設定します。

アクションエディタ左側の実行条件タブから「標準」 - 「変数〇が〇のとき」を直前で編集していたアクションに追加します。

配置した条件を以下のように設定します。

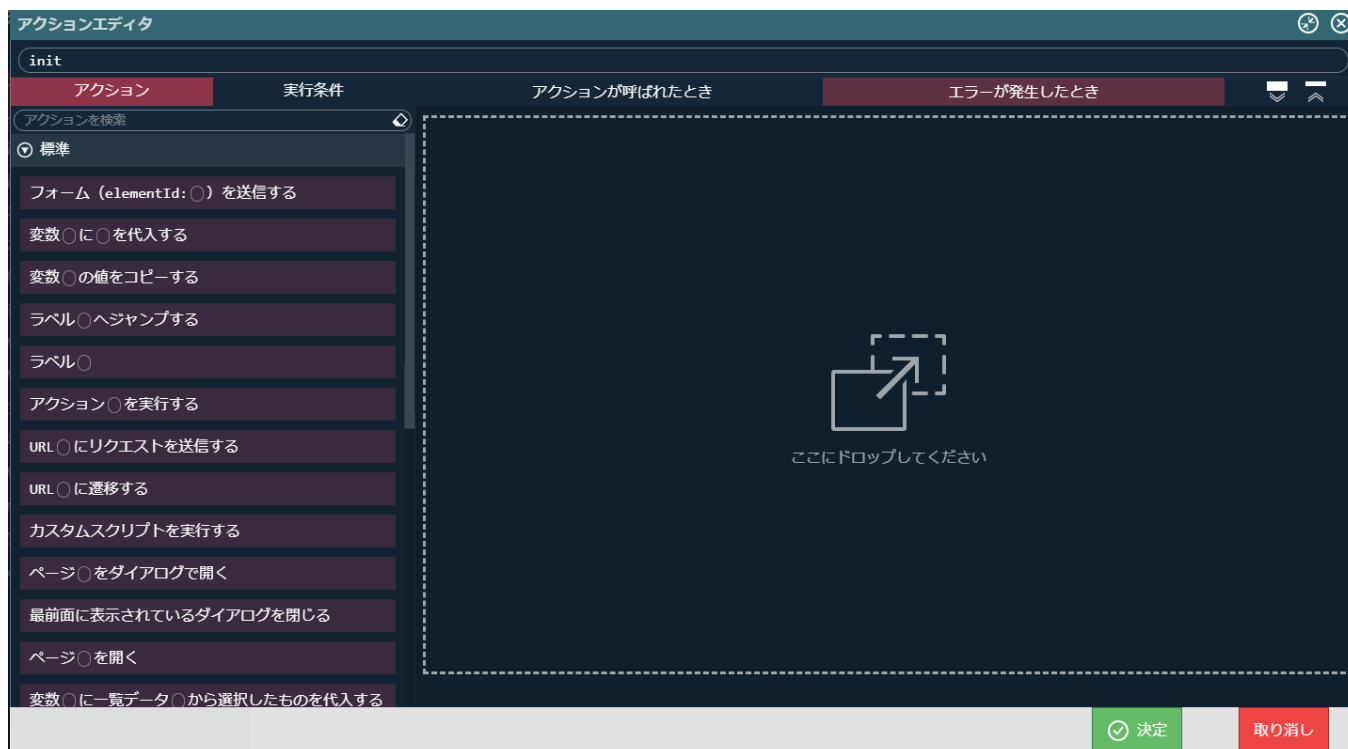
変数 (\$input.enqueteld) が () のとき



これで、画面表示時にenqueteldが指定されなかった場合にエラー画面が表示されるようになりました。

次に、アンケート情報取得で、アンケートIDに誤りがあるなどでエラーが返された場合のエラー処理を設定します。

「init」アクションのアクションエディタの「エラーが発生したとき」タブを選択します。



エラー画面のメッセージを設定するため、「アクション」タブから「標準」 - 「変数〇に〇を代入する」を選択し、「エラーが発生したとき」の一番上に配置します。

配置したアクションの変数を以下のように設定します。

変数 (\$variable.errorMessage) に (\$constant.error_enquete_missing) を代入する



次に、エラー画面の表示を行います。

「アクション」タブから「標準」 - 「ページ〇を開く」を選択し、先ほど設定したアクションの次に配置します。

配置したアクションを以下のように設定し、決定ボタンをクリックしてアクションを設定します。

ページ(error)を開く



これで回答画面の作成は完了です。

最後に画面上部の  で編集内容を保存してください。

次のページでは、回答画面をベースに集計画面を作成します。

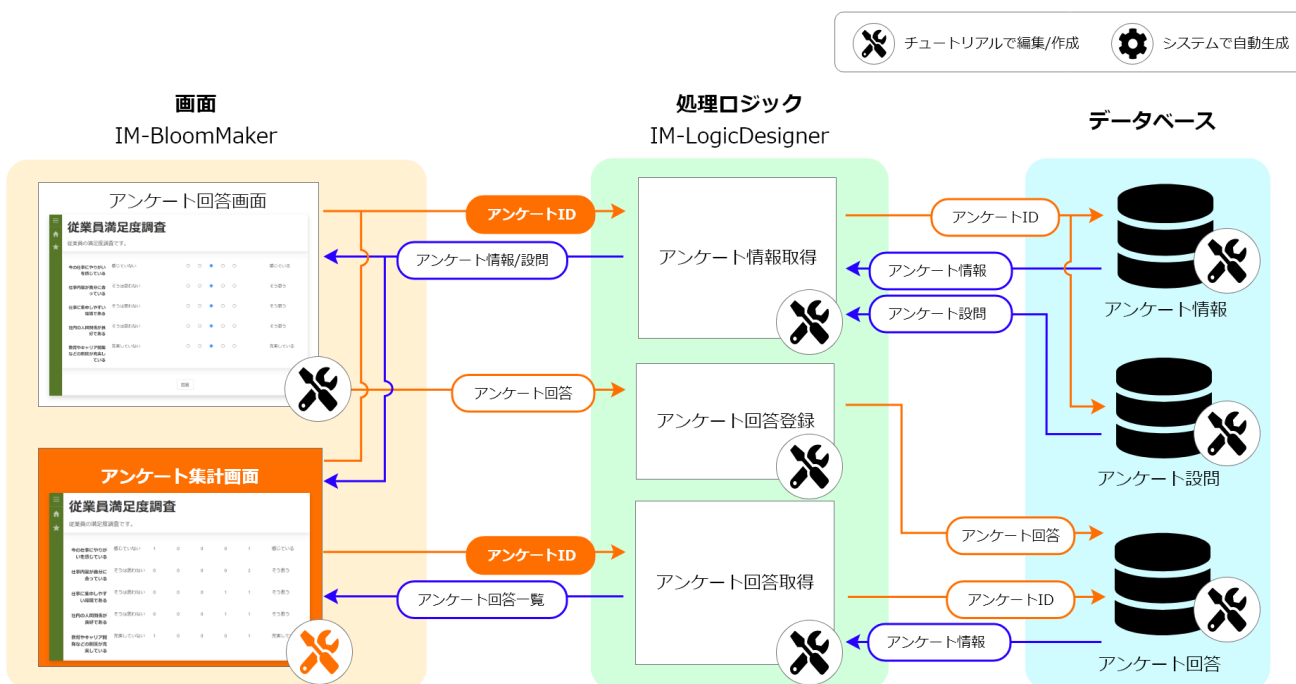
集計画面の作成

アンケート集計画面を作成します。

回答画面と同じく設問情報を取得し、それぞれの設問のどの選択肢を何人が選択して回答しているかを表示します。

回答画面と共通な処理が多いため、回答画面をテンプレートとして登録し、それをもとに編集を行います。

以下のアプリケーション全体図にある、色が塗られたところが該当箇所です。



このページの手順について、以下の動画からも確認できます。併せてご利用ください。

注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

回答画面のテンプレート登録

最初に、回答画面のテンプレート登録を行います。

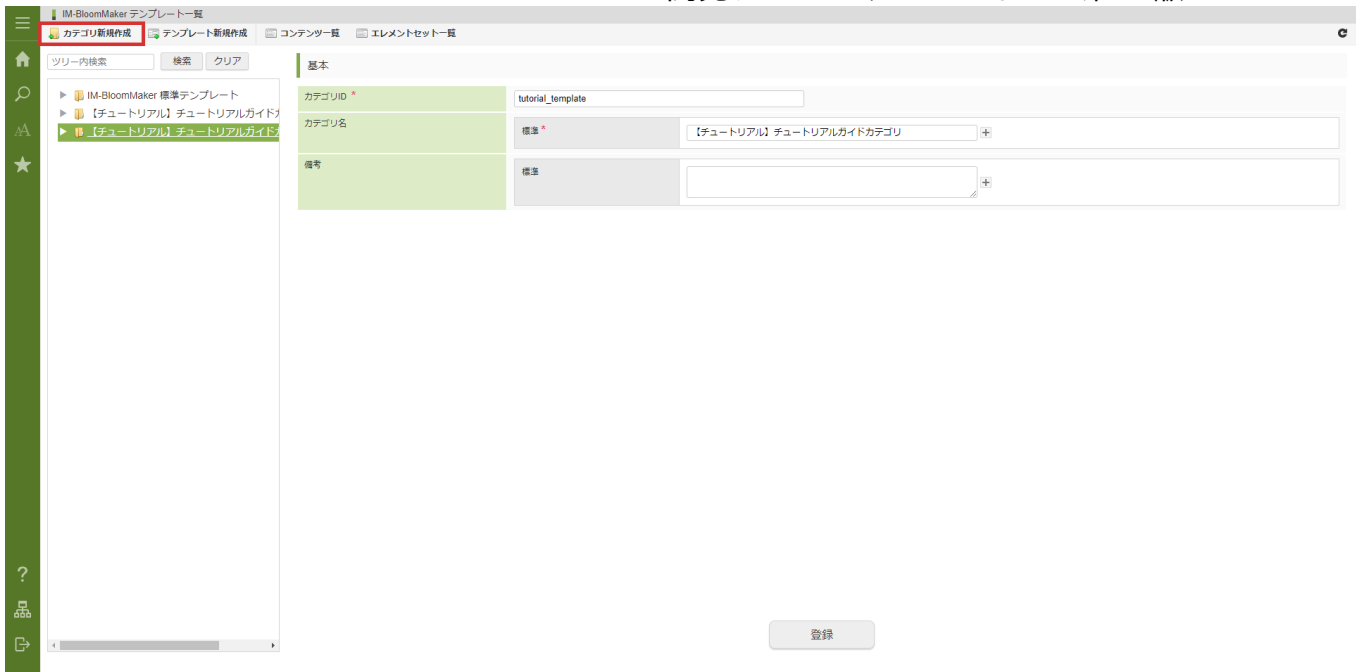
テンプレートの登録は IM-BloomMakerの機能を用いて行います。

「サイトマップ」→「BloomMaker」→「テンプレート一覧」の順に遷移します。



「IM-BloomMaker テンプレート一覧」画面上部の「カテゴリ新規作成」をクリックします。

カテゴリIDに「tutorial_template」、カテゴリ名に「【チュートリアル】チュートリアルガイドカテゴリ」と入力し「登録」をクリックします。

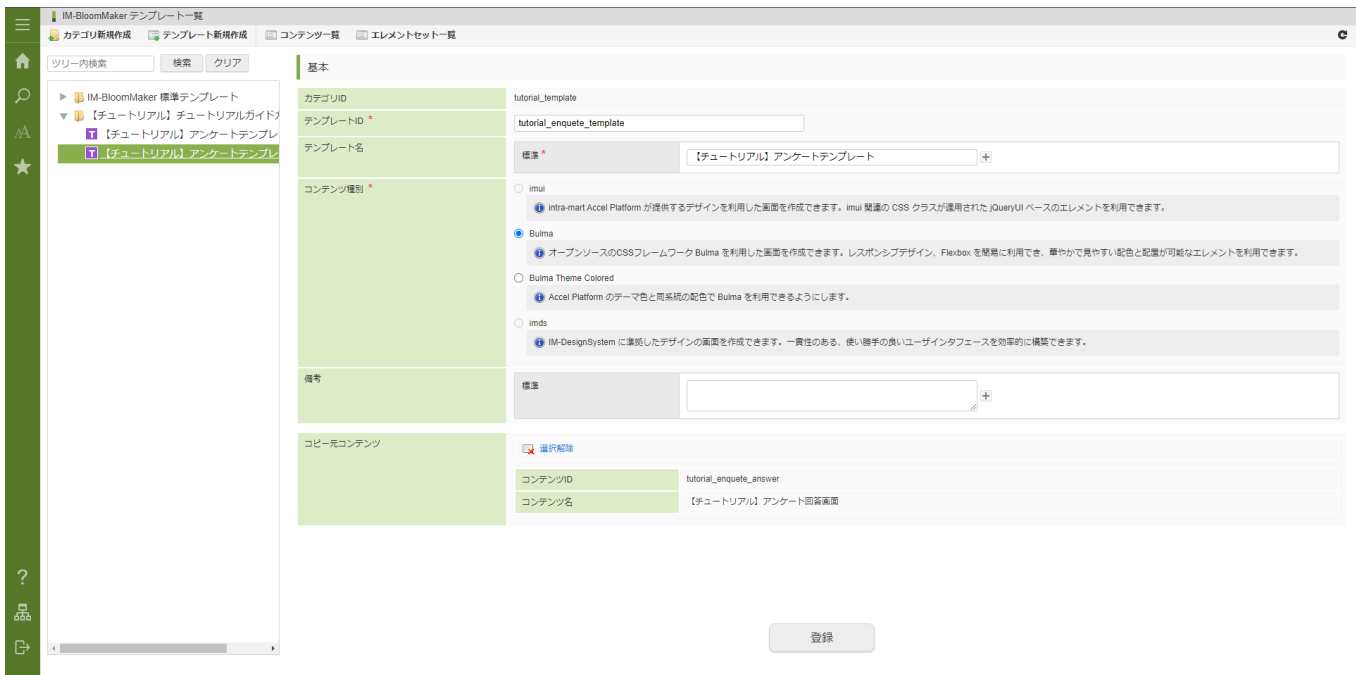


作成したカテゴリをクリックし、「テンプレート新規作成」をクリックします。

テンプレートIDに「tutorial_enquete_template」、テンプレート名に「【チュートリアル】アンケートテンプレート」と入力します。

「コピー元コンテンツ」 - 「検索」をクリックし、「コンテンツ検索」ダイアログを表示します。

「コンテンツ検索」ダイアログの中から「【チュートリアル】アンケート回答画面」を選択し、「決定」をクリックします。



「登録」をクリックし、テンプレートを登録します。

画面情報の設定

IM-BloomMakerのテンプレートを元もとに、アプリケーションに紐づいたアンケート集計画面を作成します。

テンプレートから集計画面の作成

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「画面」 - 「画面を新規作成」 - 「画面コンテンツ」を選択し、「IM-BloomMaker コンテンツ新規作成画面」を表示します。

「カテゴリ」 - 「検索」をクリックし「業務テンプレートから作成」を設定します。

「コンテンツID」に「tutorial_enquete_totalize」と設定します。

IM-BloomMaker コンテンツ新規作成

基本

カテゴリ *

検索

カテゴリID im_accel_studio

カテゴリ名 異種テンプレートから作成

コンテンツID *

tutorial_enquete_totalize

コンテンツ種別 *

imui

Bulma

Bulma Theme Colored

imds

最新バージョン番号 1

コンテンツ名

検索 *

【チュートリアル】アンケート集計画面

備考

検索

テンプレート

検索

テンプレートID

テンプレート名

登録

「テンプレート」の「検索」をクリックし、「テンプレート選択」ダイアログを表示します。



「テンプレート選択」ダイアログから、先ほど登録した「【チュートリアル】アンケートテンプレート」を選択します。
以上の設定が完了したら「登録」をクリックし、設定を終了します。

IM-BloomMaker コンテンツ新規作成

基本

カテゴリ *

検索

カテゴリID im_accel_studio

カテゴリ名 集計テンプレートから作成

コンテンツID * tutorial_enquete_totalize

コンテンツ種別 *

imui
 ① intra-mart Accel Platform が提供するデザインを利用した画面を作成できます。imui 関連の CSS クラスが適用された jQueryUI ベースのエレメントを利用できます。

Bulma
 ① オープンソースのCSSフレームワーク Bulma を利用した画面を作成できます。レスポンシブデザイン、Flexbox を簡単に利用でき、早やかで見やすい配色と配置が可能なエレメントを利用できます。

Bulma Theme Colored
 ① Accel Platform のテーマ色と同等系の配色で Bulma を利用できるようにします。

imds
 ① IM-DesignSystem に準拠したデザインの画面を作成できます。一貫性のある、使い手の良いユーザーインターフェースを効率的に構築できます。

最新バージョン番号 1

コンテンツ名

検索 *

【チュートリアル】 アンケート集計画面

検索 *

テンプレート

選択解除

テンプレートID tutorial_enquete_template

テンプレート名 【チュートリアル】 アンケートテンプレート

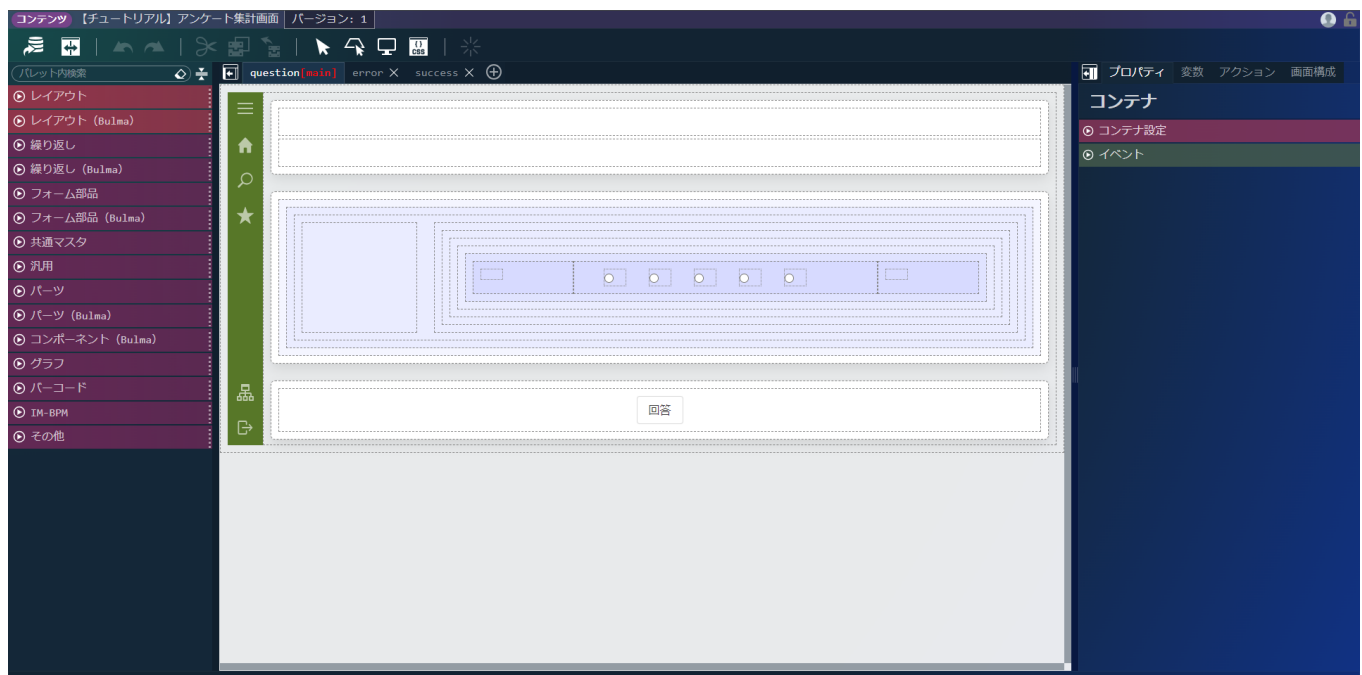
登録

画面コンテンツの作成

「アプリケーション管理画面」 - 「『画面』タブ」 - 「【チュートリアル】アンケート集計画面」を選択し、IM-BloomMaker コンテンツ編集画面を開きます。

コンテンツ画面の「デザイン編集」をクリックし、デザイナー画面を開きます。

「【チュートリアル】アンケート回答画面」と同様の画面がデザイナー上に表示されます。




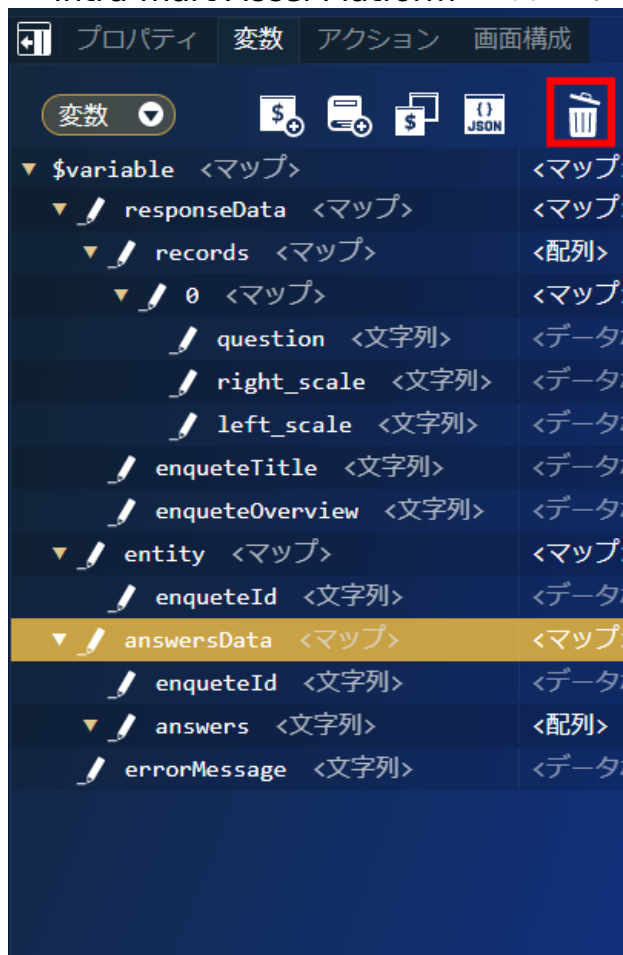
変数の設定

テンプレートで作成した変数を集計画面に必要な形式に変更します。


まずは、集計画面に不要な回答内容を格納する変数を削除します。

画面右側の「変数」タブを選択し、「\$variable > answersData」を選択します。

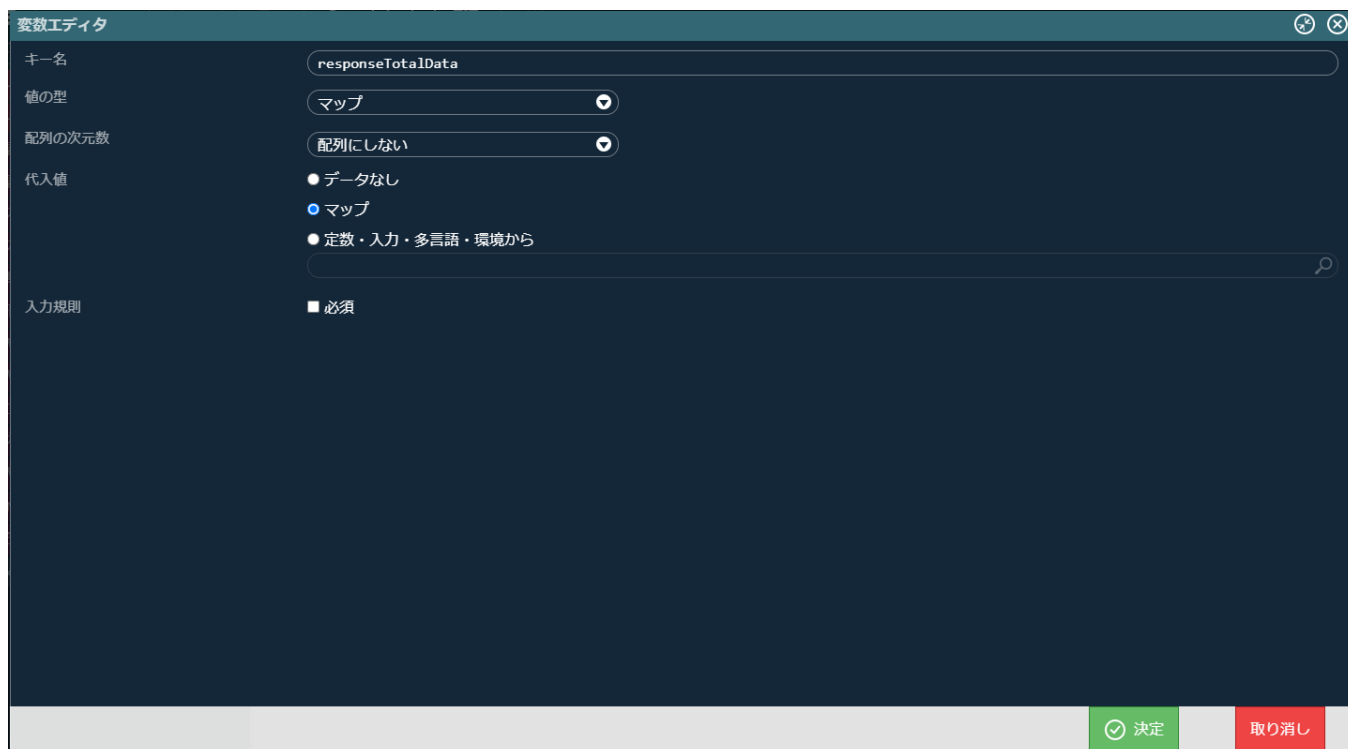
 をクリックし、変数を削除します。




次に、アンケート回答取得処理からの返却データを格納する変数を作成します。

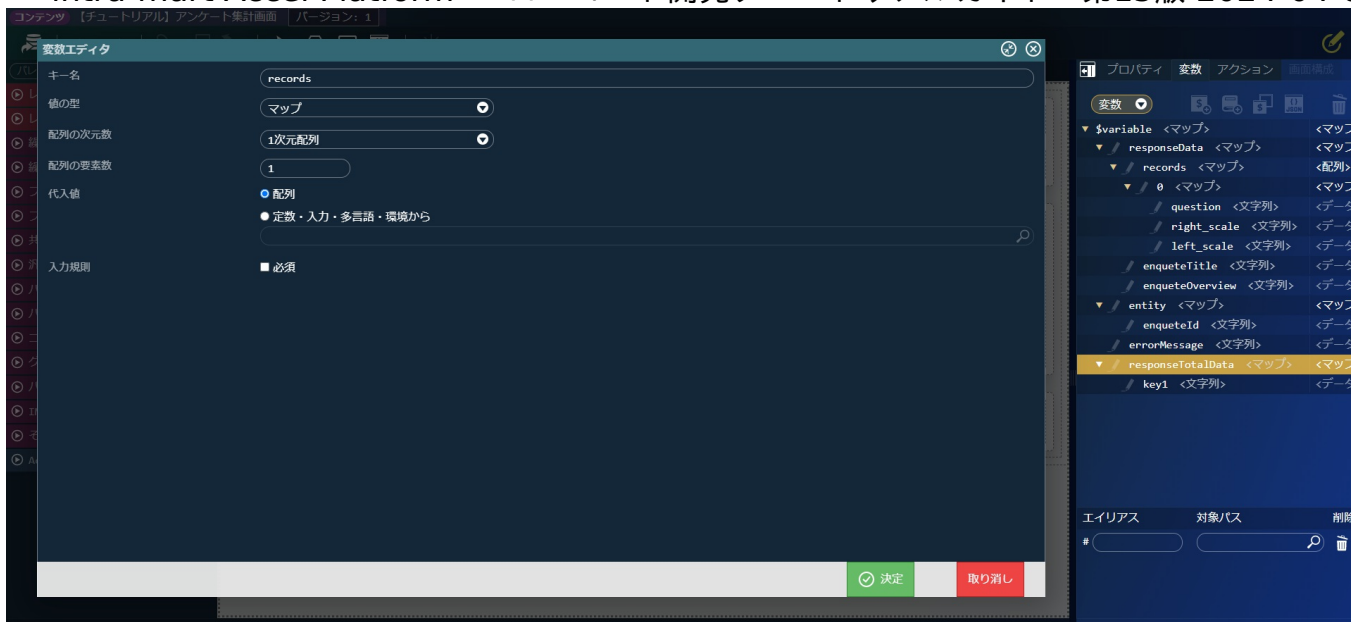
「\$variable」を選択し、 をクリックして変数エディタを表示します。


変数エディタのキー名に「responseTotalData」と入力し、値の型を「マップ」、代入値を「マップ」に変更して「決定」をクリックします。

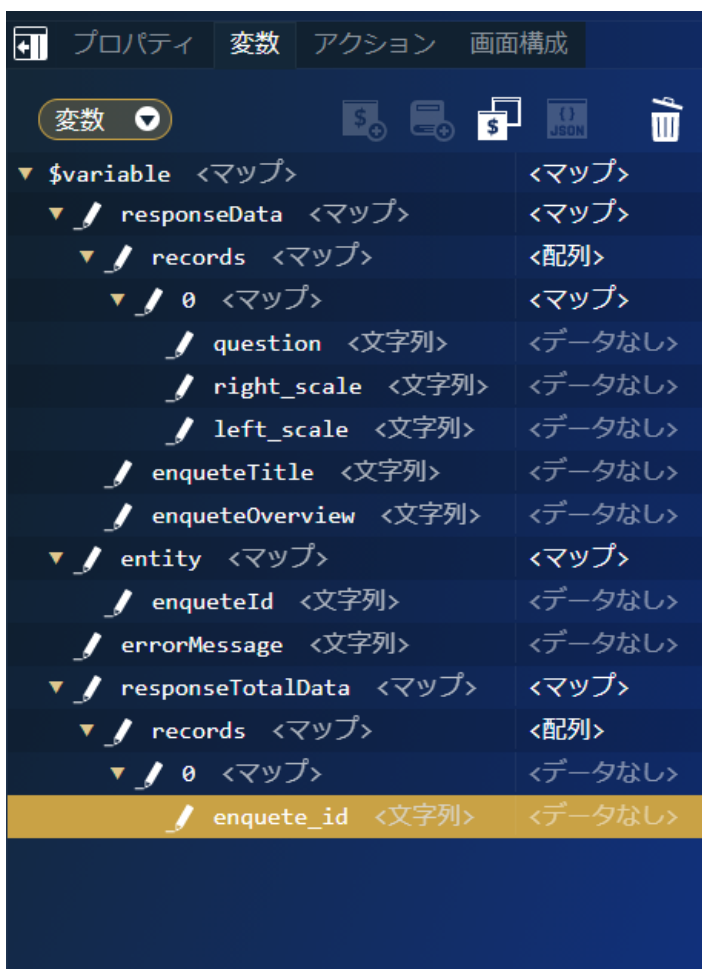


作成した「\$variable > responseTotalData」を選択し、 をクリックして変数エディタを表示します。

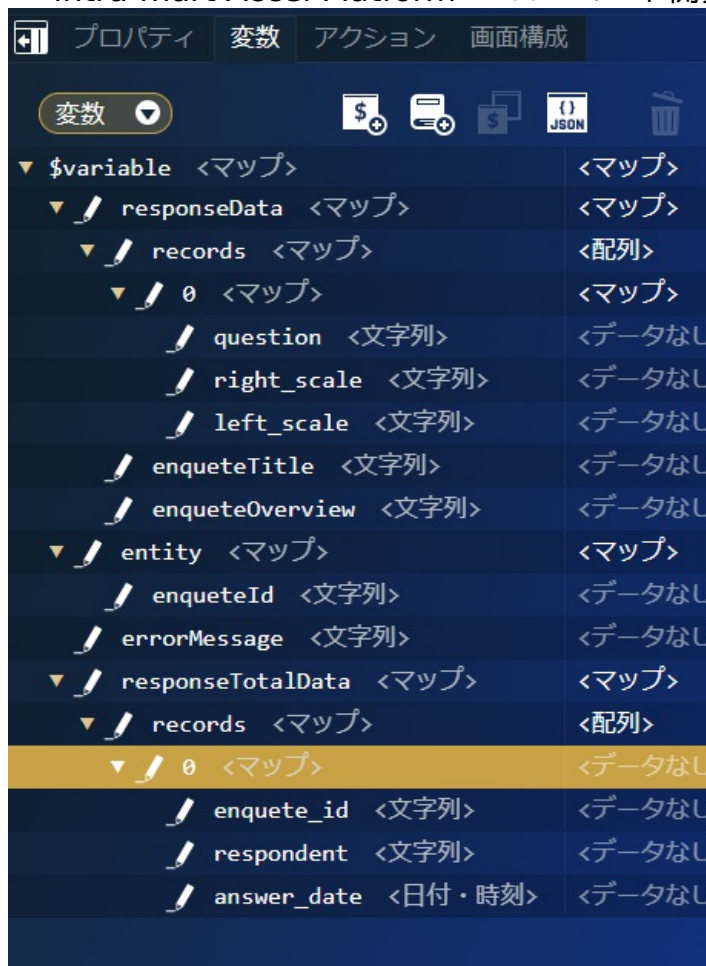
変数エディタのキー名に「records」と入力し、値の型を「マップ」、配列の次元数を「1次元配列」、配列の要素数を「1」に変更して「決定」をクリックします。




作成した「\$variable > responseTotalData > records > 0」を選択し、 をクリックして変数エディタを表示します。
変数エディタのキー名に「enquete_id」と入力し、値の型を「文字列」に変更して「決定」をクリックします。



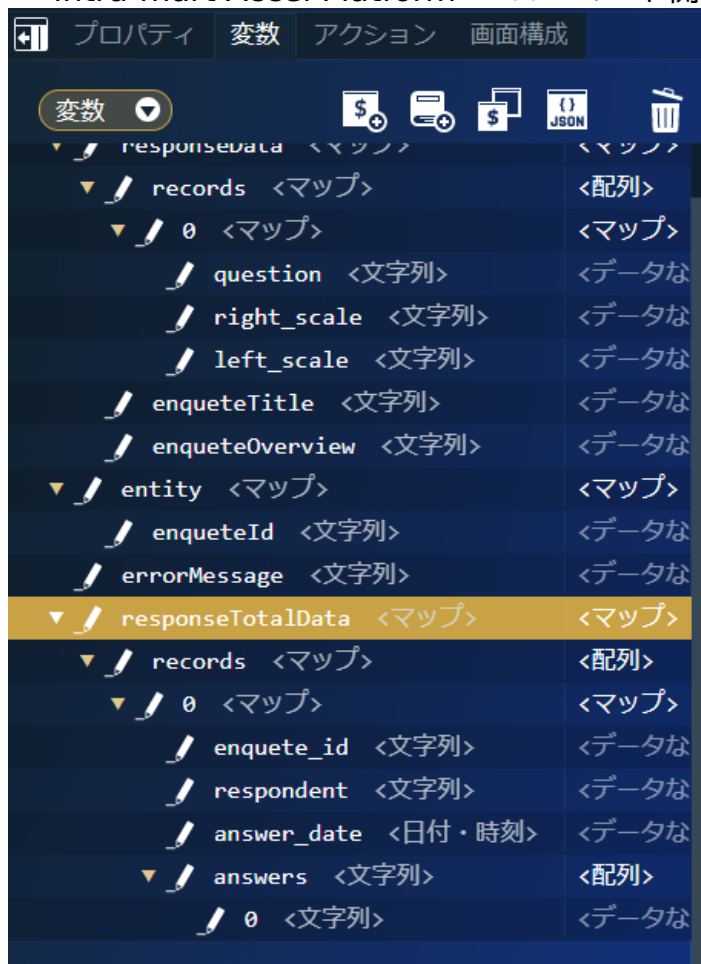
同様に「\$variable > responseTotalData > records > 0」配下に「respondent」と「answer_date」を追加します。「respondent」は値の型を「文字列」に、「answer_date」は値の型を「日付・時刻」にしてください。



最後に、回答内容を格納する変数を追加します。「\$variable > responseTotalData > records > 0」を選択し、 をクリックして変数エディタを表示します。


変数エディタのキー名に「answers」と入力し、値の型を「文字列」、配列の次元数を「1次元配列」、配列の要素数を「1」に変更して「決定」をクリックします。

これで、回答取得処理で取得したデータを格納する変数の作成は完了です。




変数はJSON入力で以下のコードを記載しても設定できます。

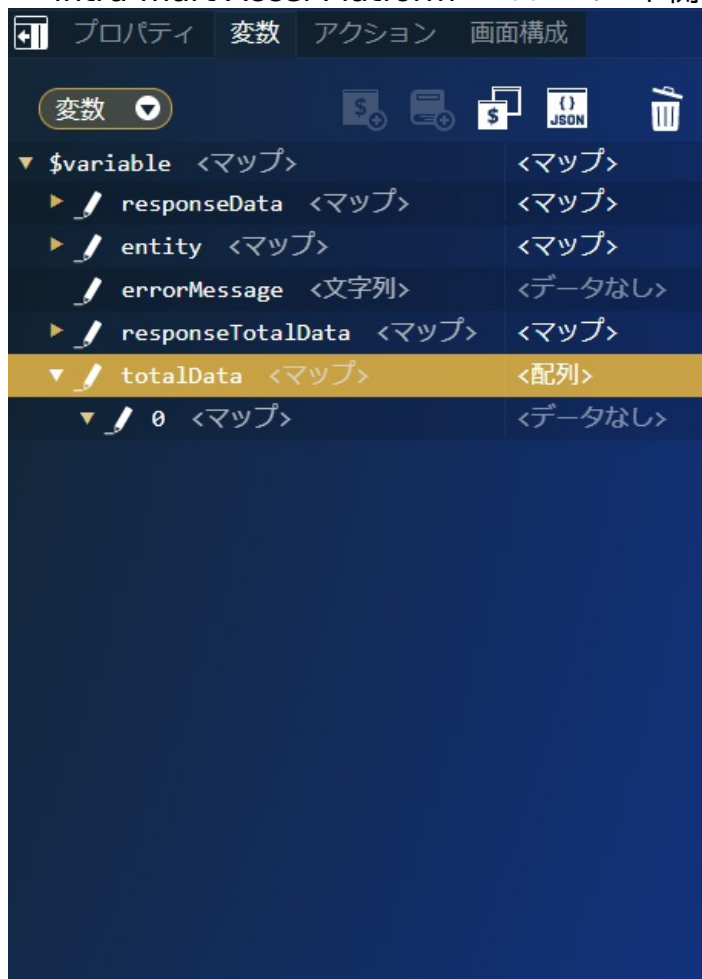
```
"responseTotalData": {
  "records": [
    {
      "enquete_id": "",
      "respondent": "",
      "answer_date": null,
      "answers": [
        ""
      ]
    }
  ]
}
```


JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

次に、どの回答に何人が回答したのかのデータを格納する変数を作成します。

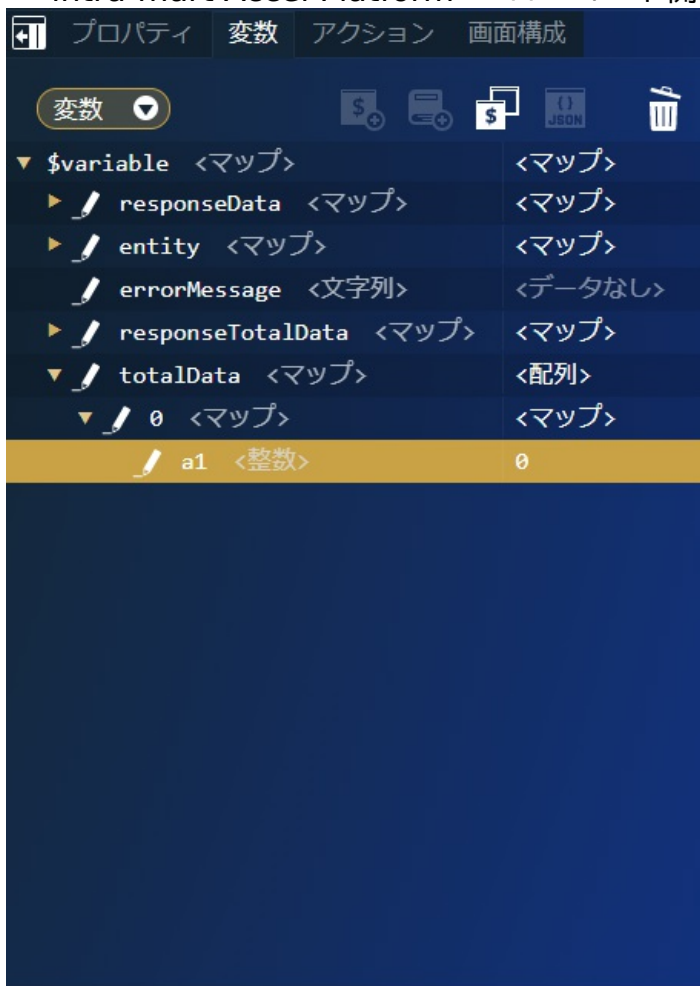
「\$variable」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「totalData」と入力し、値の型を「マップ」、配列の次元数を「1次元配列」、配列の要素数を「1」に変更して「決定」をクリックします。

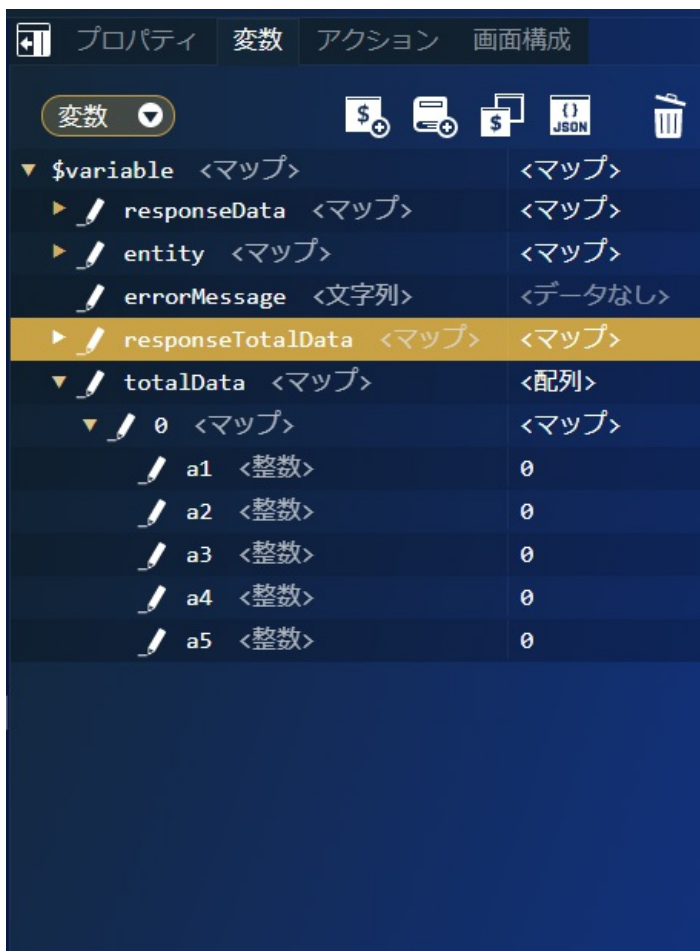


作成した「\$variable > totalData > 0」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「a1」と入力し、値の型を「整数」、代入値のラジオボタンを「固定数値」にしてから「0」を設定し、「決定」をクリックします。




同様の設定で、「a2」、「a3」、「a4」、「a5」を作成します。



変数はJSON入力で以下のコードを記載しても設定できます。

```
"totalData": [
  {
    "a1": 0,
    "a2": 0,
    "a3": 0,
    "a4": 0,
    "a5": 0
  }
]
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

これで、集計画面用の変数の設定が完了しました。

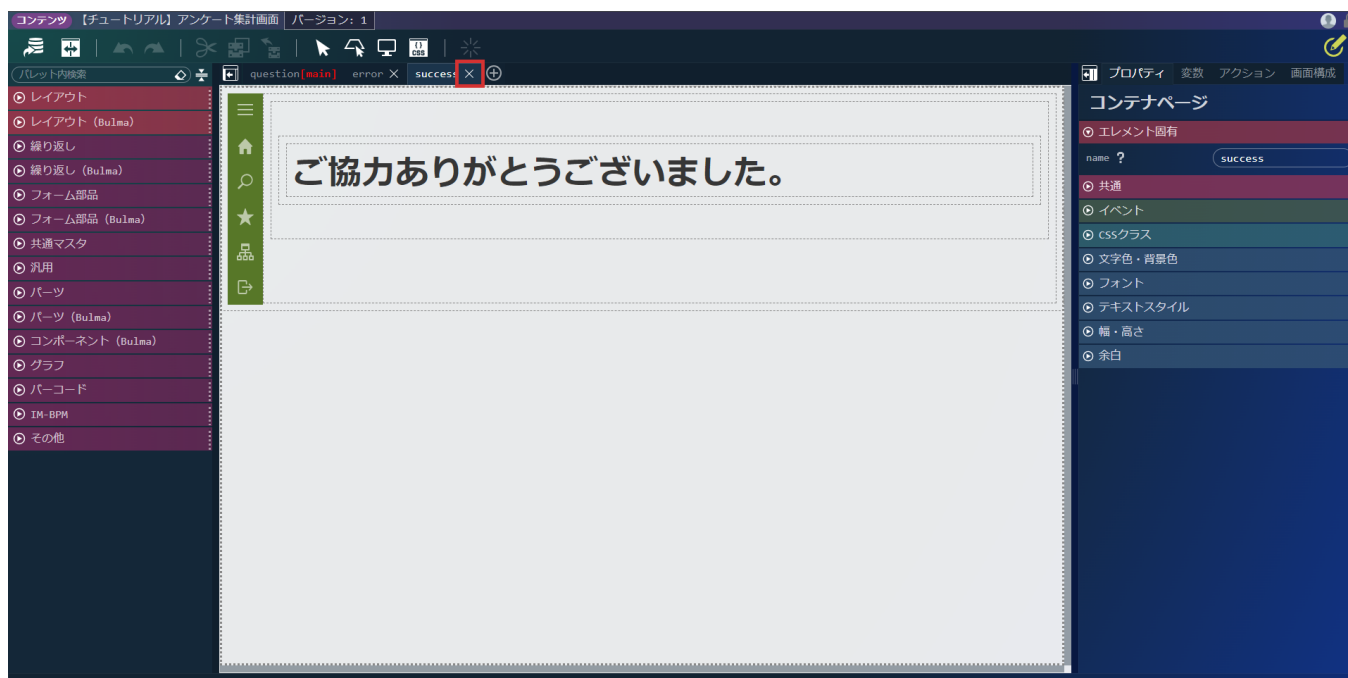
画面の修正

テンプレートで生成した各画面項目を集計画面に変更します。


まずは、集計画面として不要な画面項目を削除します。

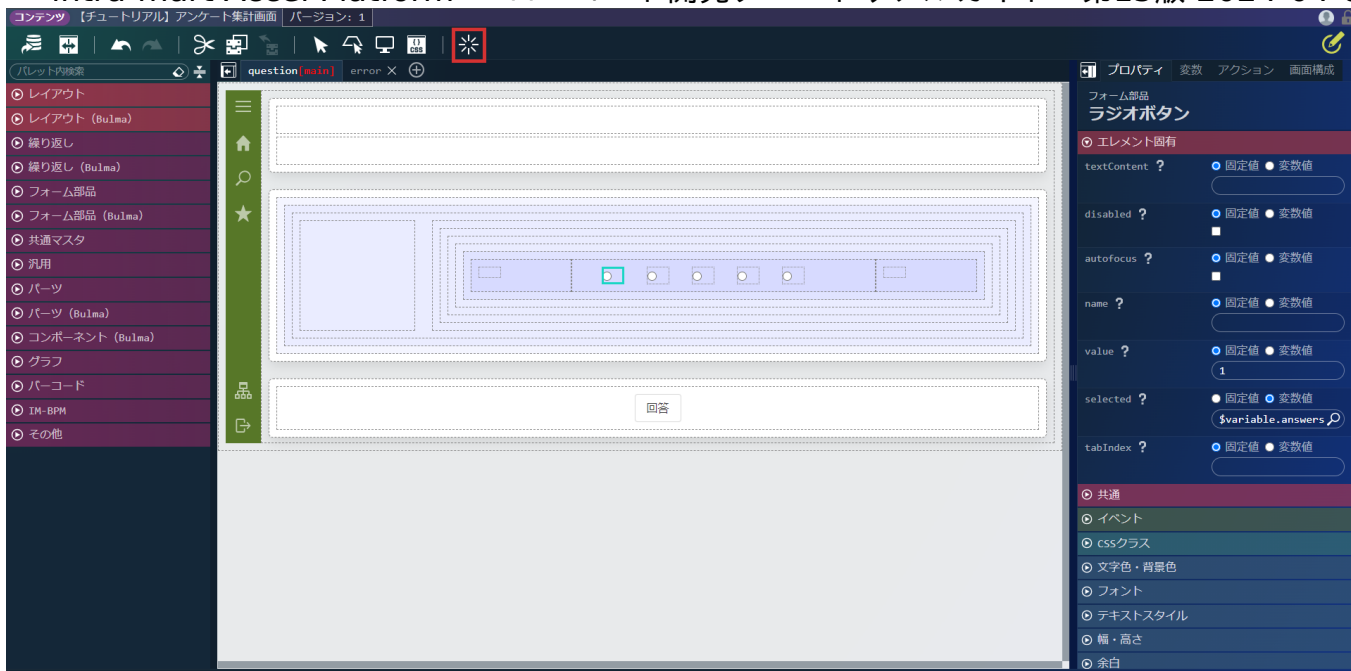
集計画面では、アンケート回答登録処理は実行されないため、回答成功画面を削除します。

エディタの「success」タブ右側に表示されている「x」をクリックし、回答成功画面を削除します。

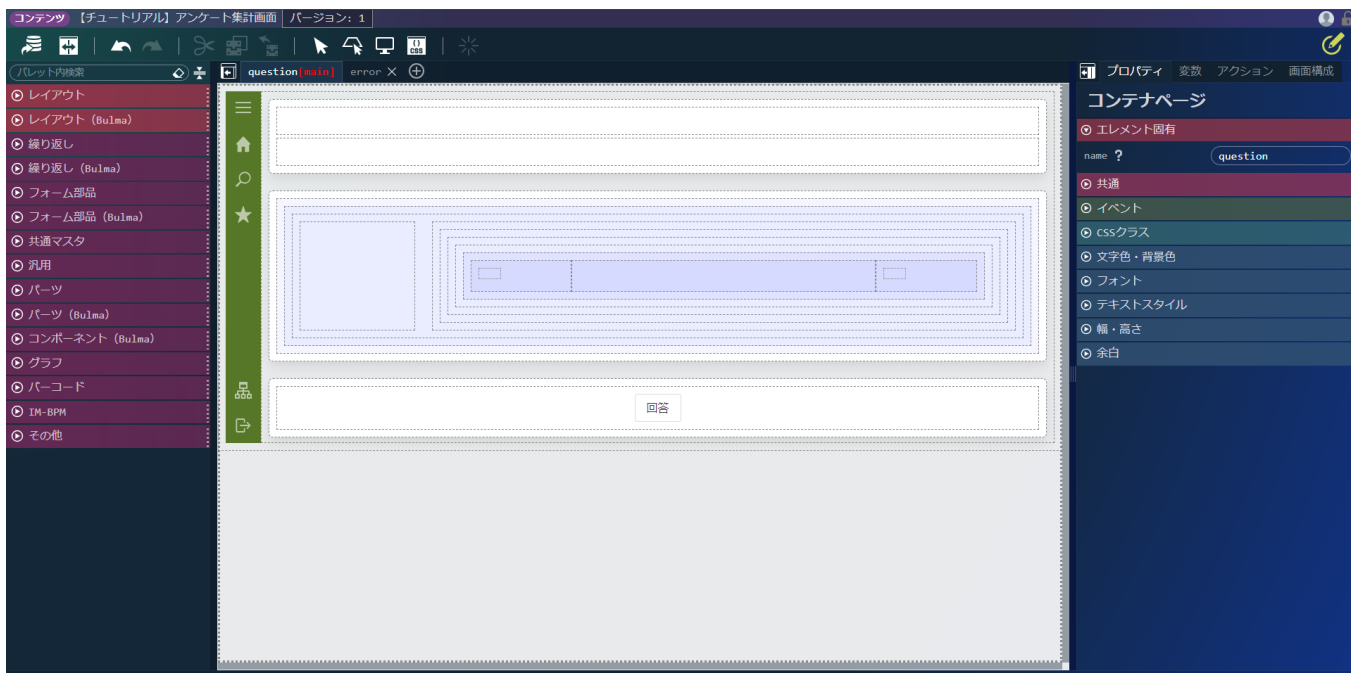


次に、回答項目のラジオボタンを削除します。

エディタの「question」タブを開き、回答項目のラジオボタンを選択し、DELETEキーを押すか画面上部の  をクリックしてください。

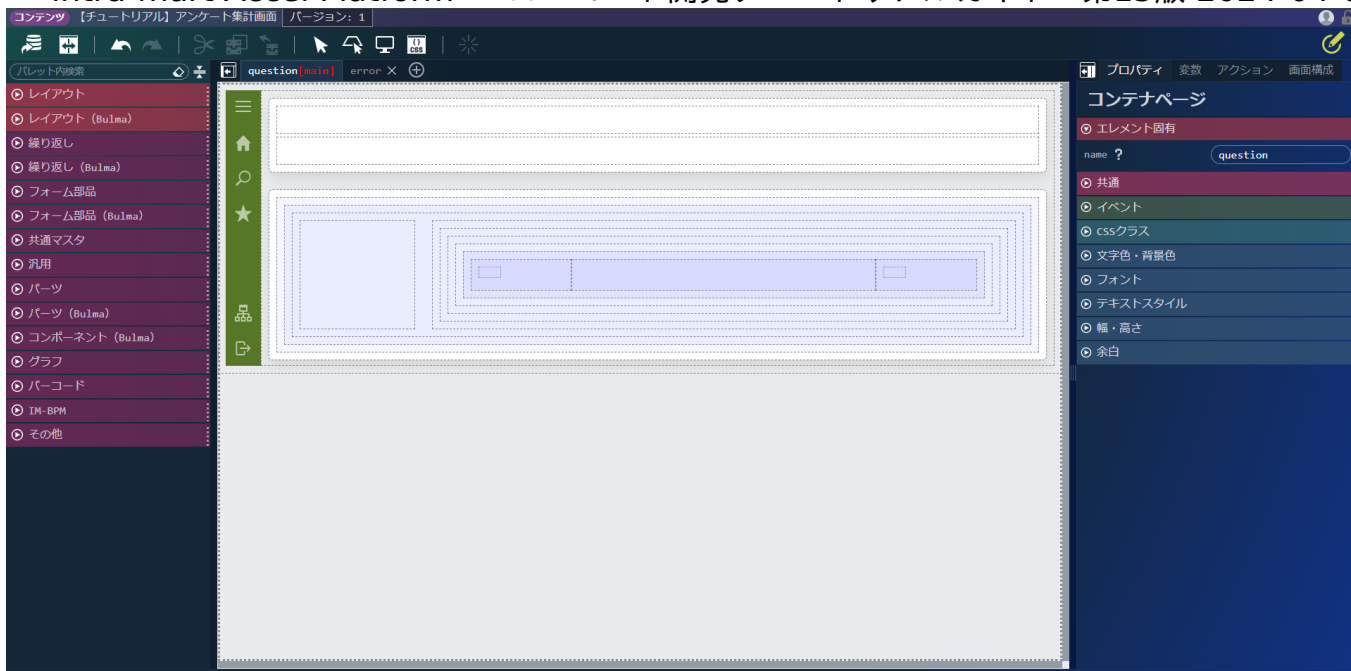


同様の手順で、ラジオボタンを全て削除します。



同じく、回答ボタンは不要なため削除します。

回答ボタンを配置しているボックスを選択し、削除します。

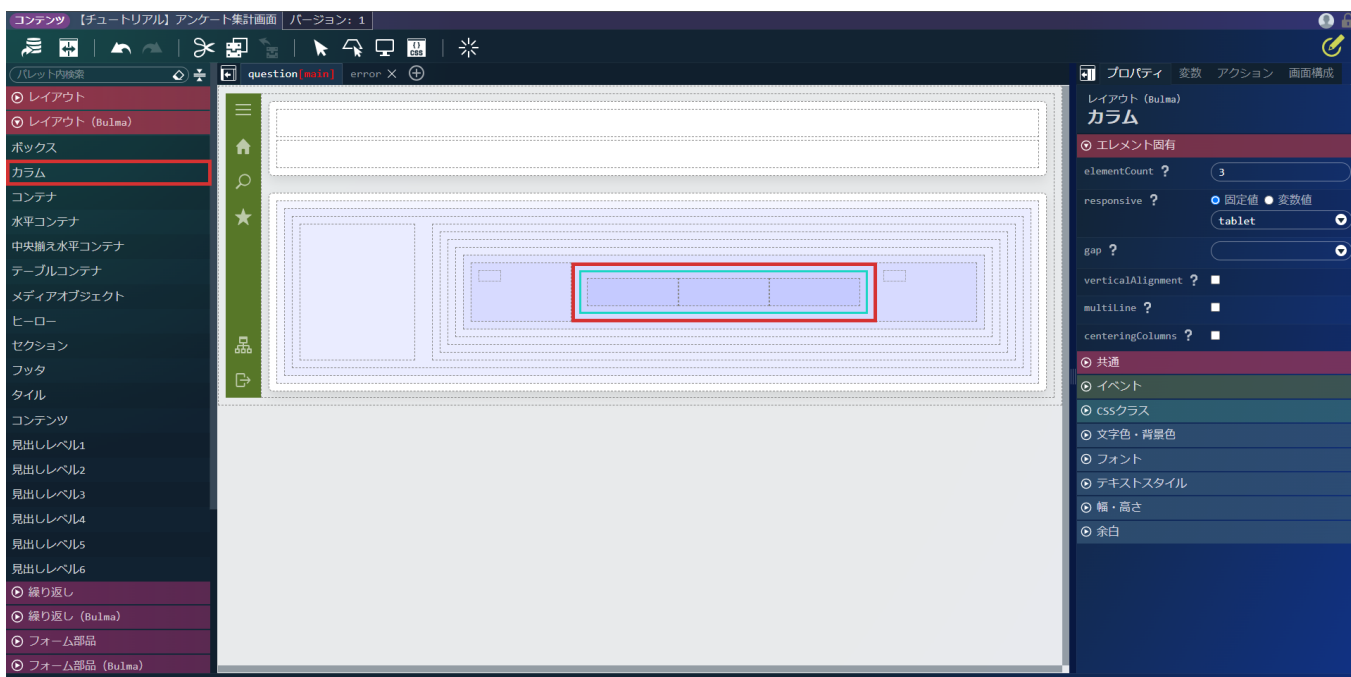


画面項目の追加

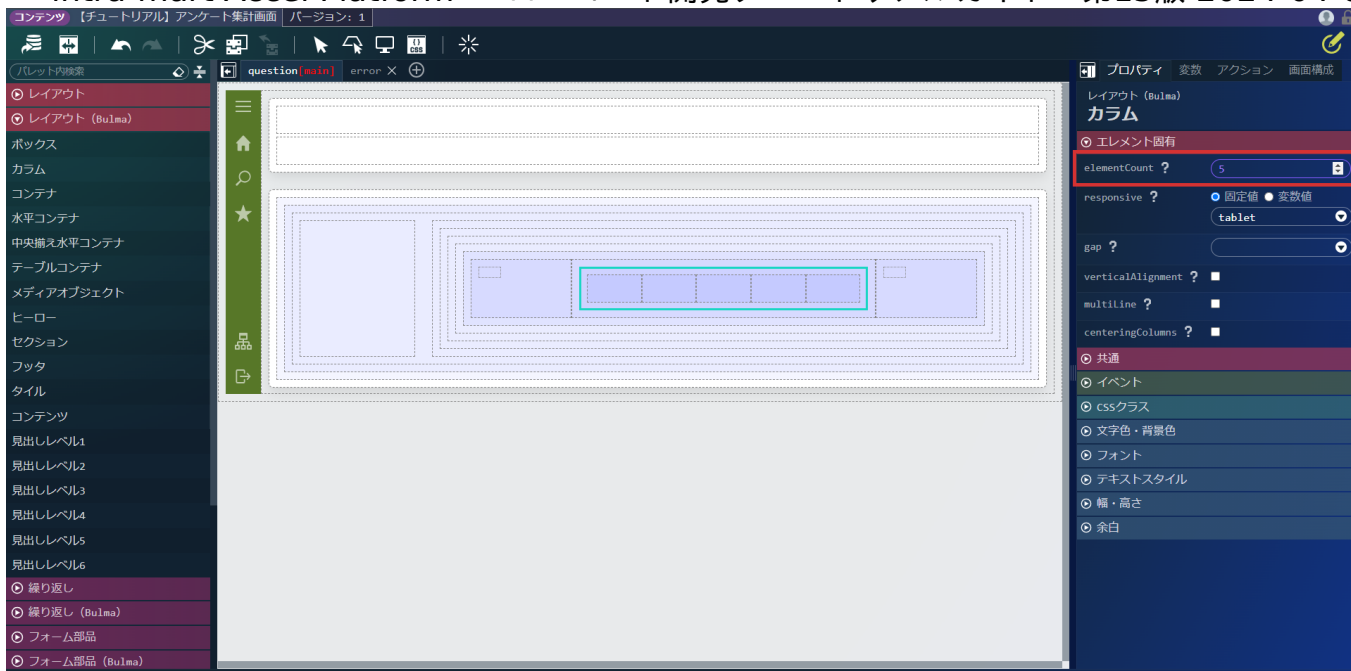
集計画面独自の画面項目を追加します。

集計画面では、どの選択肢に何人のユーザーが回答したかを数値で表示します。

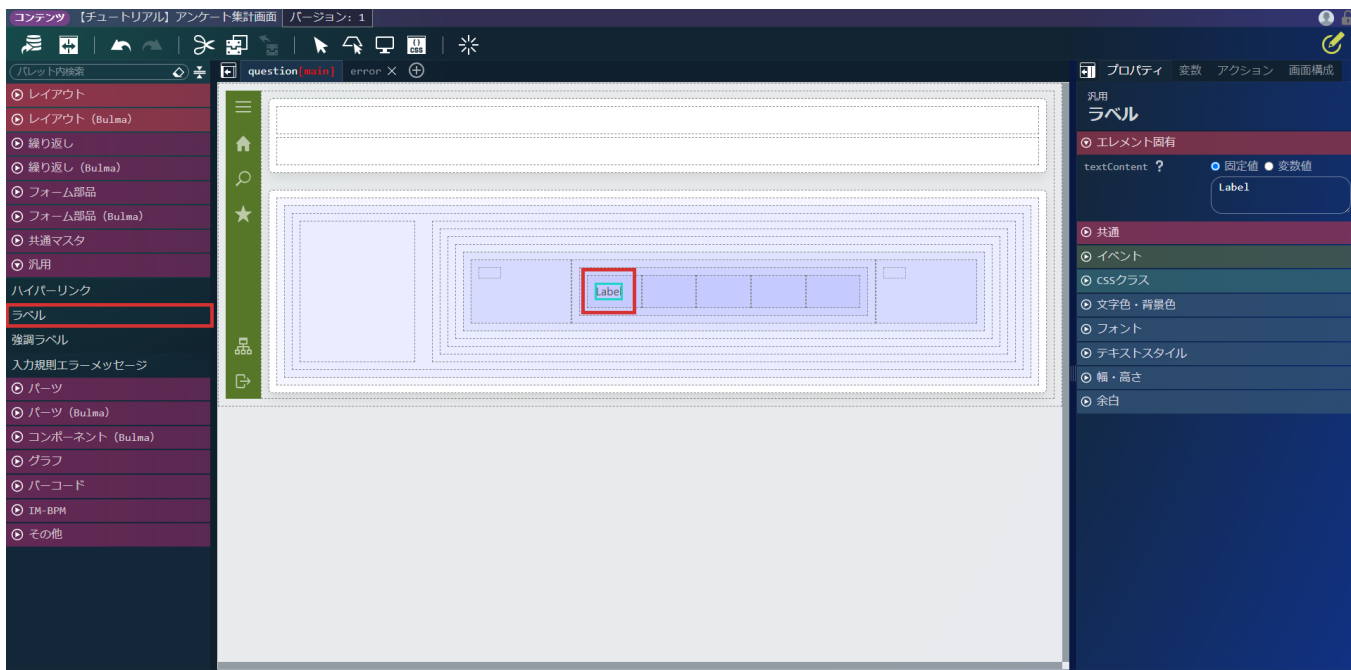
「レイアウト (Bulma)」 - 「カラム」を選択し、画面中央に配置されている「カラムアイテム (繰り返し)」の中に配置された「カラム」アイテムのうち、中央のカラムの内部に配置します。



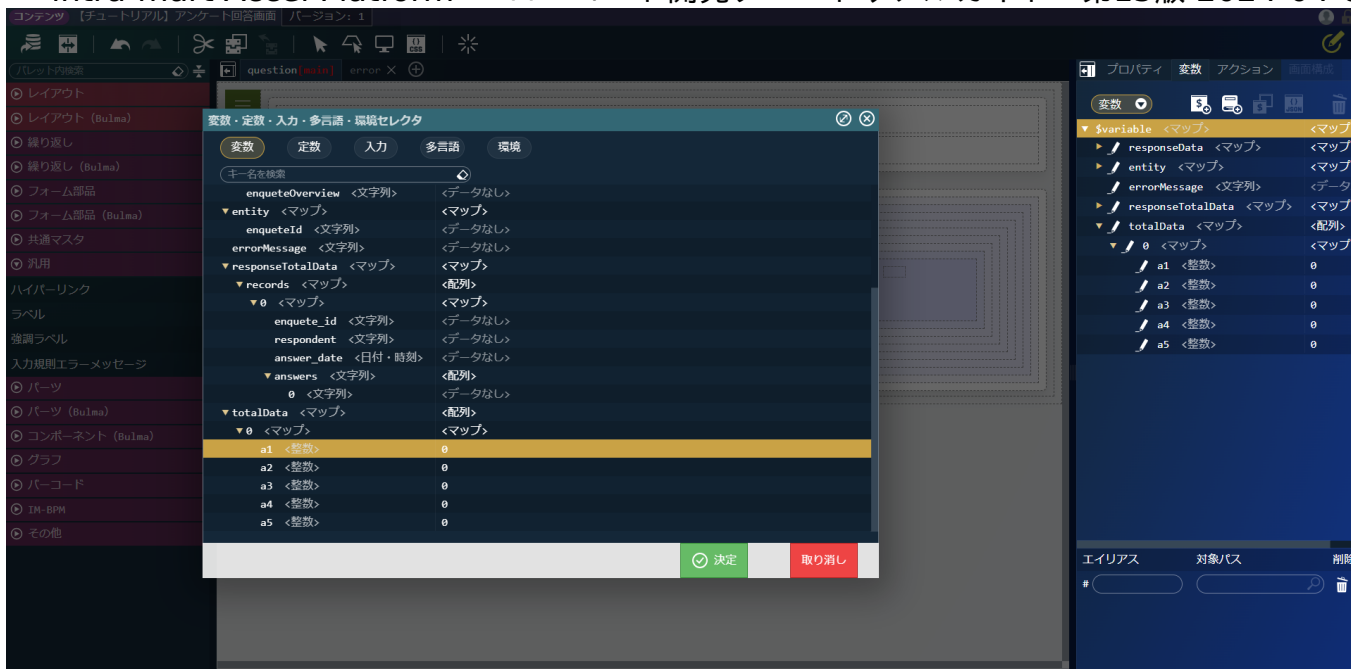
新たに配置したカラムを選択し、画面右側「プロパティ」タブの「カラム」 - 「エレメント固有」 - 「elementCount」に「5」を指定します。



次に、配置した「カラム」アイテムに選択した人が何人いるかを表示するラベルを追加します。
 「汎用」 - 「ラベル」を選択し、先ほど配置したカラムの左端に配置します。



画面右側「プロパティ」タブの「ラベル」 - 「エレメント固有」 - 「textContent」を「変数値」に設定します。
 「textContent」の 🔍 から「\$variable > totalData > 0 > a1」を選択し、「決定」をクリックします。
 上記の設定が完了したら、「textContent」の入力ボックスにフォーカスし、変数文字列のrecords[0]の部分をrecords[\$index]へ変更します。



同様に5つのカラム全てにラベルを配置していきます。

変数値は左から「``${variable}.totalData[$index].a1``」、「``${variable}.totalData[$index].a2``」、「``${variable}.totalData[$index].a3``」、「``${variable}.totalData[$index].a4``」、「``${variable}.totalData[$index].a5``」となるように設定します。

これで、画面表示の修正は完了です。

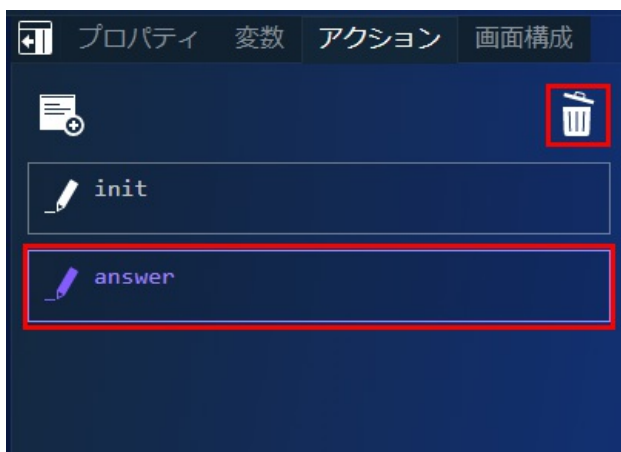
アクションの追加、修正

次は、アクションの追加や修正を行います。

まずは、集計画面には不要なアクションを削除します。

画面右側の「アクション」タブをクリックします。

アクション一覧の中から、「answer」を選択し、 をクリックしてアクションを削除します。



次に、画面表示時に集計結果を取得するアクションを作成します。



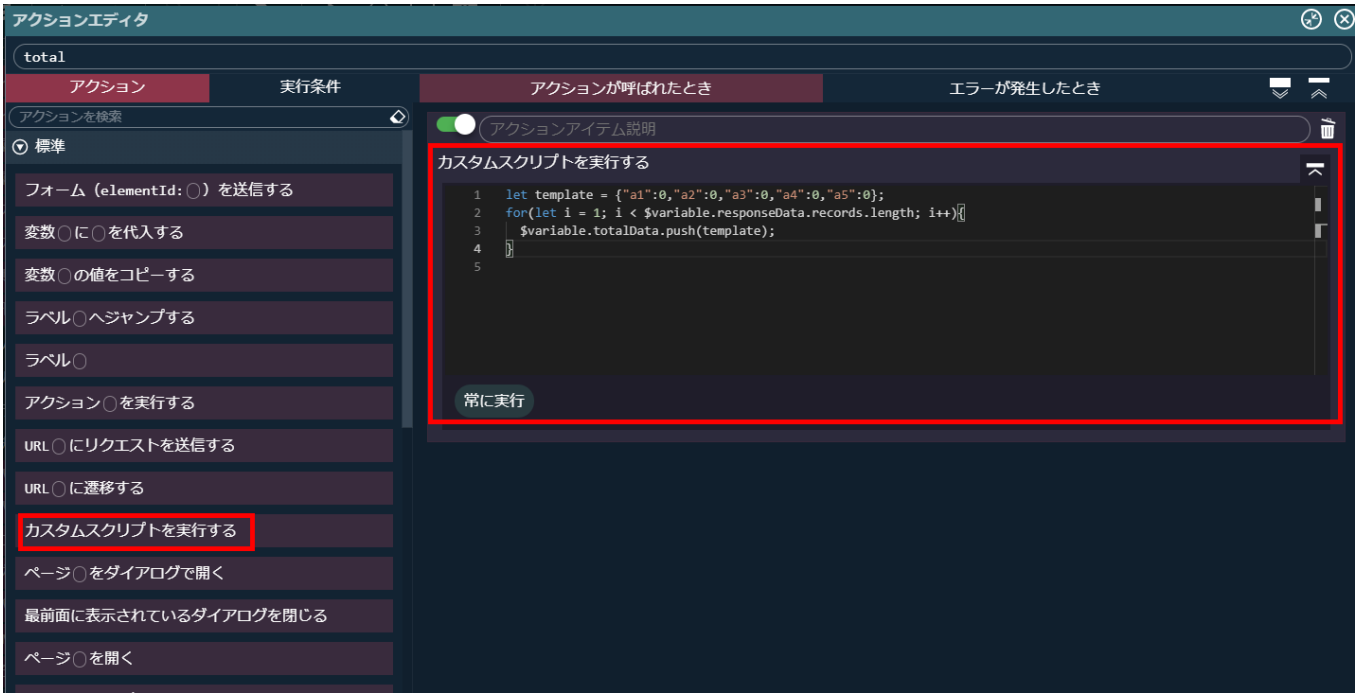
をクリックしてアクションエディタを表示し、名称を「新しいアクション #1」から「total」に変更します。

まず、設問項目に合わせて、集計項目の配列を作成するためのカスタムスクリプトを作成します。

「標準」 - 「カスタムスクリプトを実行する」を選択し、アクションの最初に配置します。

カスタムスクリプトの入力欄に以下を設定します。

```
let template = {"a1":0,"a2":0,"a3":0,"a4":0,"a5":0};
for(let i = 1; i < $variable.responseData.records.length; i++){
  $variable.totalData.push(template);
}
```



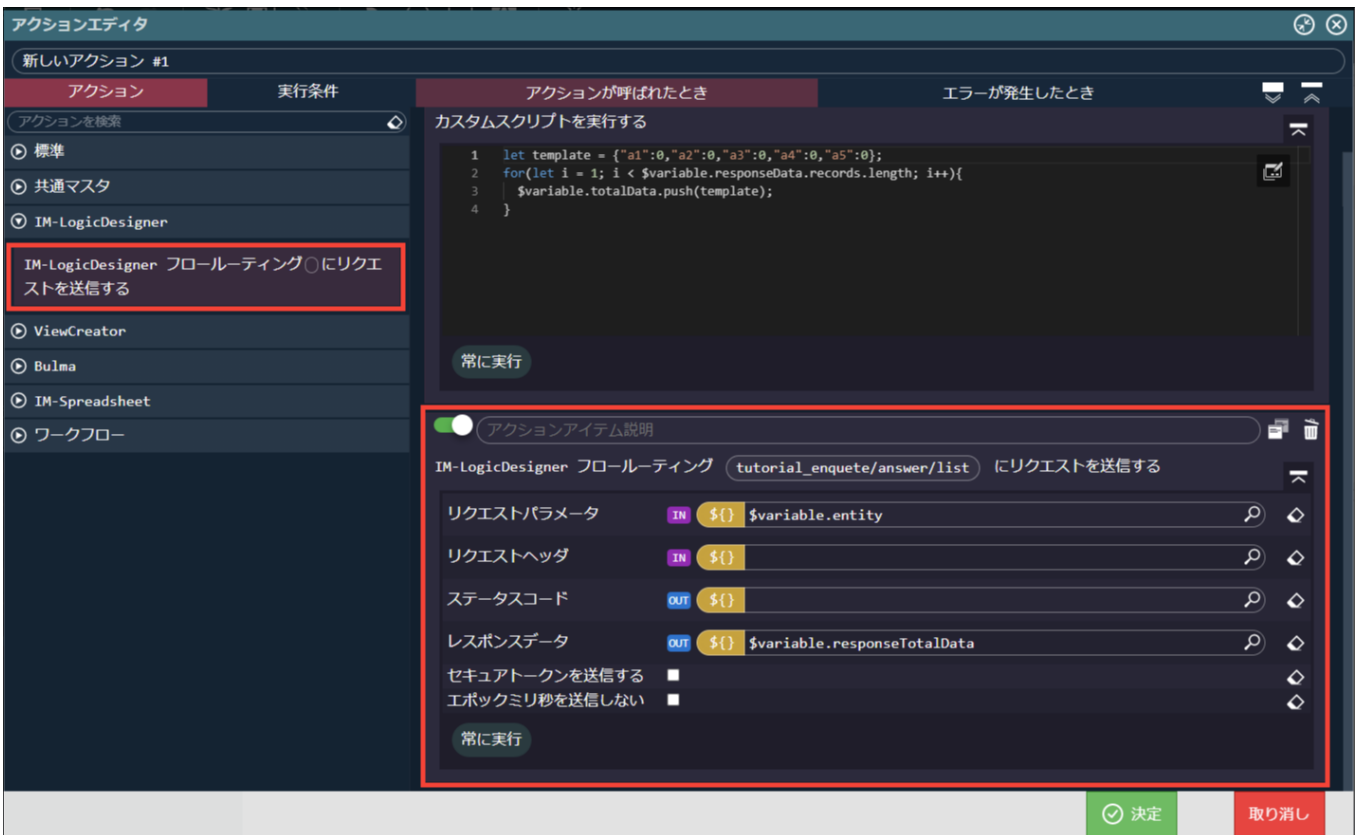
これで、取得した設問項目数に合わせて回答の集計結果を格納する変数の配列が作れるようになりました。

次に、回答結果を取得する業務ロジックを呼び出します。

「IM-LogicDesigner」 - 「IM-LogicDesigner フロールーティング〇にリクエストを送信する」を選択し、先ほど作成したカスタムスクリプトの次に配置します。

🔍 から「tutorial_enquete/answer/list」を選択します。

リクエストパラメータとして「\$variable > entity」をレスポンスデータとして「\$variable > responseTotalData」を設定します。



取得した回答結果からどの選択肢を何人選択したかを集計するカスタムスクリプトを作成します。

「標準」 - 「カスタムスクリプトを実行する」を選択し、アクションの最後に配置します。

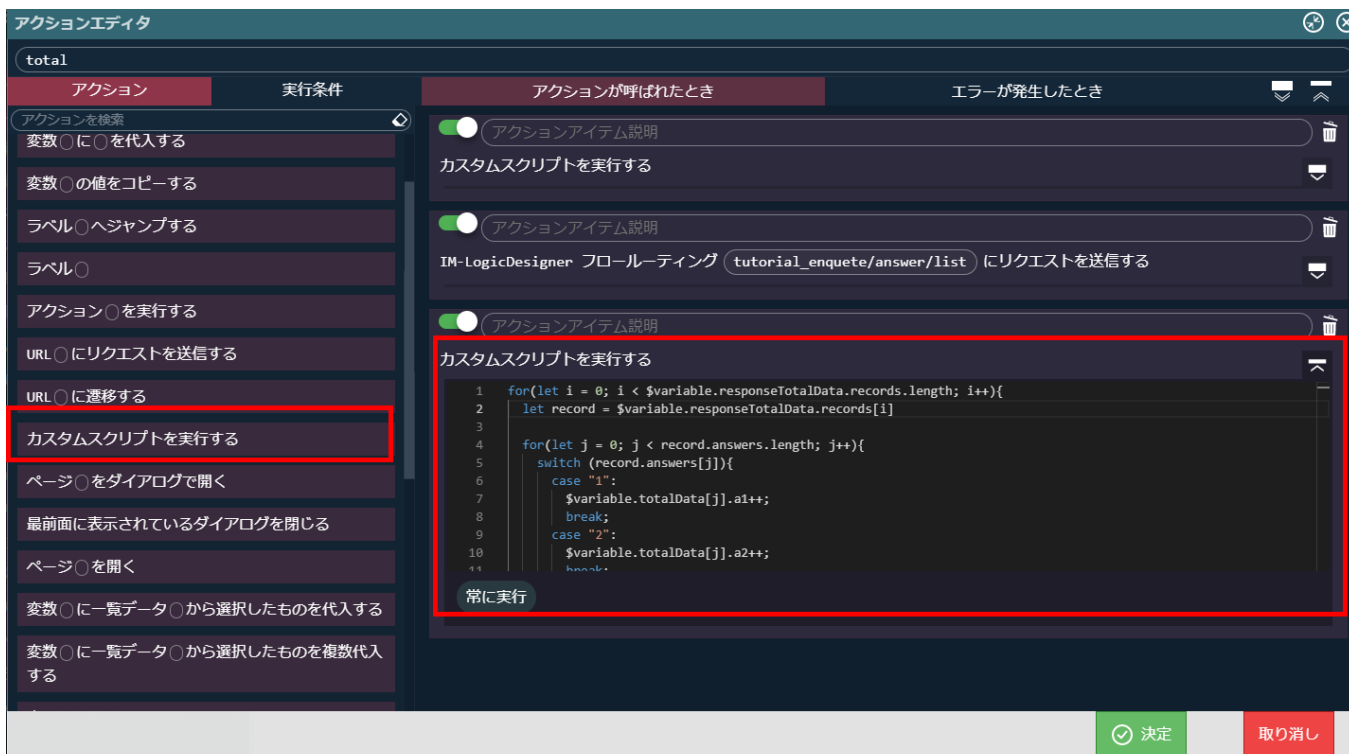
カスタムスクリプトの入力欄に以下を設定します。

```

for(let i = 0; i < $variable.responseTotalData.records.length; i++){
  let record = $variable.responseTotalData.records[i]

  for(let j = 0; j < record.answers.length; j++){
    switch (record.answers[j]){
      case "1":
        $variable.totalData[j].a1++;
        break;
      case "2":
        $variable.totalData[j].a2++;
        break;
      case "3":
        $variable.totalData[j].a3++;
        break;
      case "4":
        $variable.totalData[j].a4++;
        break;
      case "5":
        $variable.totalData[j].a5++;
        break;
    }
  }
}
}
}

```



アクションエディタ

total

アクション 実行条件 アクションが呼びされたとき エラーが発生したとき

アクションを検索

変数〇に〇を代入する

変数〇の値をコピーする

ラベル〇へジャンプする

ラベル〇

アクション〇を実行する

URL〇にリクエストを送信する

URL〇に移る

カスタムスクリプトを実行する

ページ〇をダイアログで開く

最前面に表示されているダイアログを閉じる

ページ〇を開く

変数〇に一覧データ〇から選択したものを代入する

変数〇に一覧データ〇から選択したものを複数代入する

アクションアイテム説明

カスタムスクリプトを実行する

アクションアイテム説明

IM-LogicDesigner フロールーティング (tutorial_enquete/answer/list) にリクエストを送信する

アクションアイテム説明

カスタムスクリプトを実行する

```

1 for(let i = 0; i < $variable.responseTotalData.records.length; i++){
2   let record = $variable.responseTotalData.records[i]
3
4   for(let j = 0; j < record.answers.length; j++){
5     switch (record.answers[j]){
6       case "1":
7         $variable.totalData[j].a1++;
8         break;
9       case "2":
10        $variable.totalData[j].a2++;
11        break;

```

常に実行

決定 取り消し

以上の設定ができたら「決定」をクリックして、アクション設定を終了します。

これで、集計結果を取得するアクションが作成できました。

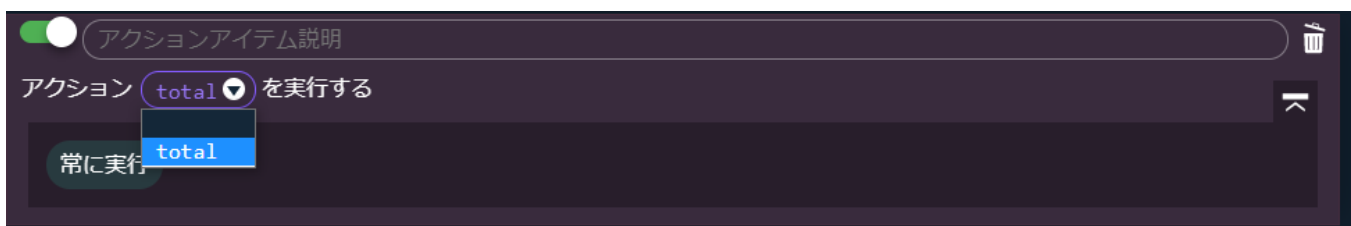
作成した「total」アクションを初期表示で呼び出すよう修正します。

「アクション」タブの「init」アクションの  をクリックし、アクションエディタを表示します。


「標準」 - 「アクション〇を実行する」を選択し、initアクションの「ラベル (END)」の上に配置します。



アクションに「total」を設定します。




回答の初期表示設定を行うカスタムスクリプトは不要なので削除します。

「IM-LogicDesigner フロールーティング (tutorial_enquete/question) にリクエストを送信する」の下に配置された「カスタムスクリプトを実行する」の  をクリックして削除します。



以上の設定ができたら、「決定」をクリックしてアクションエディタを閉じます。

最後に画面上部の  で編集内容を保存してください。

これで、集計画面の完成です。

次のページからは作成した画面にアクセスするためのURL設定を行います。

画面URLの設定

アンケート回答画面とアンケート集計画面にアクセスするためのURLを設定します。

このページの手順について、以下の動画からも確認できます。併せてご利用ください。



注意

動画は2023 Autumn(Hollyhock)環境で録画したものです。操作内容に大きな差はありませんが、アプリケーション管理画面など一部のUIに差異があります。

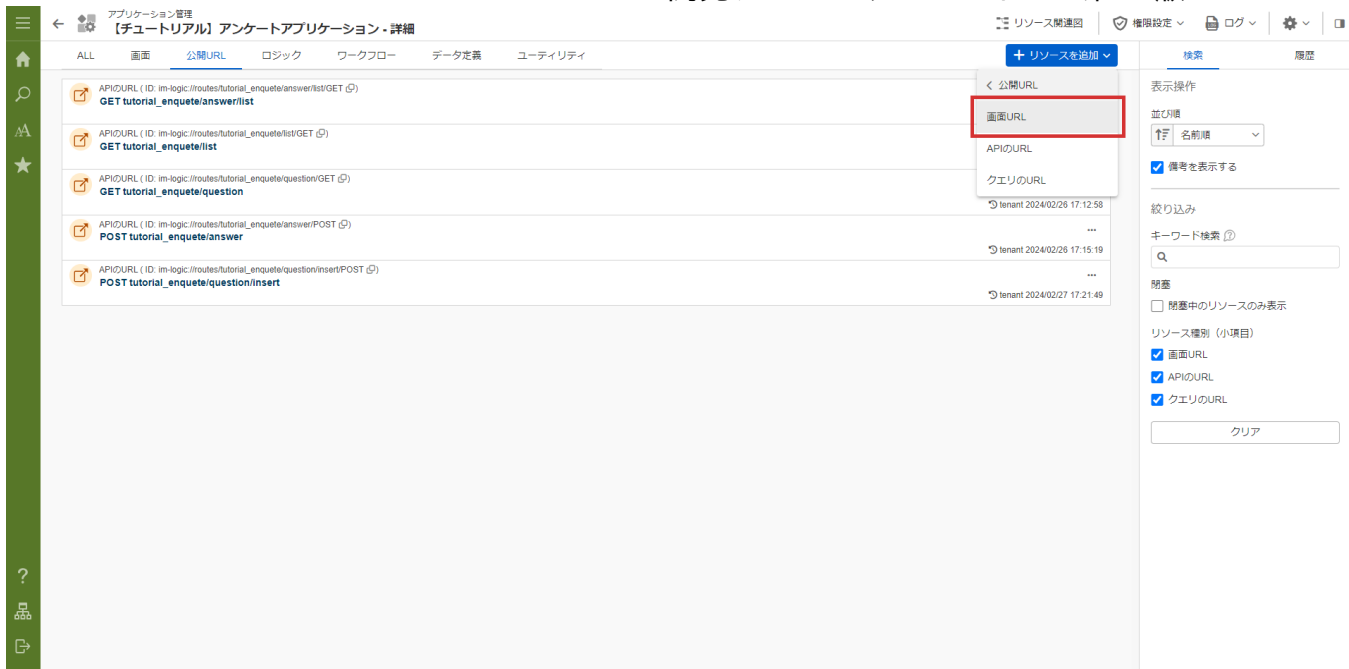
アンケート回答画面のURL設定

アンケート回答画面のURL設定を行います。

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

アプリケーション管理画面に遷移し、「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「画面URL」を選択し「IM-BloomMaker ルーティング定義新規作成」画面を開きます。



「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「カテゴリ」 - 「検索」から「ルーティングカテゴリ検索」ダイアログを表示します。

「ルーティングカテゴリ検索」ダイアログ上で「業務テンプレートから作成」を選択し、「決定」をクリックします。

「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「ルーティングID」に「tutorial_enquete_answer」と設定します。

「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「コンテンツ」 - 「検索」から「コンテンツ検索」ダイアログを表示します。

「コンテンツ検索」ダイアログから「【チュートリアル】アンケート回答画面」を選択し、「決定」をクリックします。



決定

「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「URL」に「tutorial_enquete/answer」と設定します。
 「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「ルーティング名」に「【チュートリアル】アンケート回答」と設定します。

その他の設定は標準通りとし、「IM-BloomMaker ルーティング定義新規作成」画面下の「登録」をクリックし、URL設定を登録します。

アンケート集計画面のURL設定

アンケート集計画面のURL設定を行います。

「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。

アプリケーション管理画面に遷移し、「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「画面URL」を選択し「IM-BloomMaker ルーティング定義新規作成」画面を開きます。

「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「カテゴリ」 - 「検索」から「ルーティングカテゴリ検索」ダイアログを表示します。

「ルーティングカテゴリ検索」ダイアログ上で「業務テンプレートから作成」を選択し、「決定」をクリックします。

「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「ルーティングID」に「tutorial_enquete_totalize」と設定します。

「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「コンテンツ」 - 「検索」から「コンテンツ検索」ダイアログを表示します。

「コンテンツ検索」ダイアログから「【チュートリアル】アンケート集計画面」を選択し、「決定」をクリックします。

▼ 業務テンプレートから作成

- ▶ 【チュートリアル】日報アプリケーション
- ▶ 【チュートリアル】アンケート集計画面
- ▶ 【チュートリアル】アンケート回答画面
- ▶ Accel-Mart Quick 利用者

基本

カテゴリID	im_bouquet	
コンテンツID	tutorial_enquete_totalize	
コンテンツ種別	Bulma	
最新バージョン番号	1	
コンテンツ名	標準	【チュートリアル】アンケート集計画面 +
備考	標準	 +

決定

「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「URL」に「tutorial_enquete/totalize」と設定します。
 「IM-BloomMaker ルーティング定義新規作成」画面の「ルーティング」タブ - 「ルーティング名」に「【チュートリアル】アンケート集計画面」と設定します。
 その他の設定は標準通りとし、「IM-BloomMaker ルーティング定義新規作成」画面下の「登録」をクリックし、URL設定を登録します。

IM-BloomMaker ルーティング

ルーティング

前処理

カテゴリ *	<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;">🔍 検索</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">カテゴリID</td> <td>im_bouquet</td> </tr> <tr> <td>カテゴリ名</td> <td>業務テンプレートから作成</td> </tr> </table> </div>	カテゴリID	im_bouquet	カテゴリ名	業務テンプレートから作成
カテゴリID	im_bouquet				
カテゴリ名	業務テンプレートから作成				
ルーティングID *	tutorial_enquete_totalize				
コンテンツ *	<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;">🔍 検索</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">コンテンツID</td> <td>tutorial_enquete_totalize</td> </tr> <tr> <td>コンテンツ名</td> <td>【チュートリアル】アンケート集計画面</td> </tr> </table> </div>	コンテンツID	tutorial_enquete_totalize	コンテンツ名	【チュートリアル】アンケート集計画面
コンテンツID	tutorial_enquete_totalize				
コンテンツ名	【チュートリアル】アンケート集計画面				
コンテンツバージョン番号 *	<input checked="" type="radio"/> 最新バージョンを利用する <input type="radio"/> 利用するバージョンを指定する 利用バージョン * <input style="width: 100%;" type="text"/>				
メソッド *	GET				
URL *	/imatv/tutorial_enquete/totalize				
認可URI	im-bloommaker-content://contents/route/tutorial_enquete_totalize				
ルーティング名	標準 <input style="width: 80%;" type="text" value="【チュートリアル】アンケート集計"/> +				
備考	標準 <input style="width: 80%;" type="text"/> +				

登録

これでアンケートアプリケーションへアクセスするURLの設定ができました。
 次のページでは、メニューからアプリケーションへアクセスするための設定を行います。

メニューの設定

作成したアプリケーションにメニューからアクセスできるよう設定します。
 サンプルとして登録した「従業員満足度調査」を表示します。

! 注意

Accel-Mart Quick環境をご利用の場合は、メニュー設定画面への導線が異なります。また、メニューグループの新規作成は行えません。

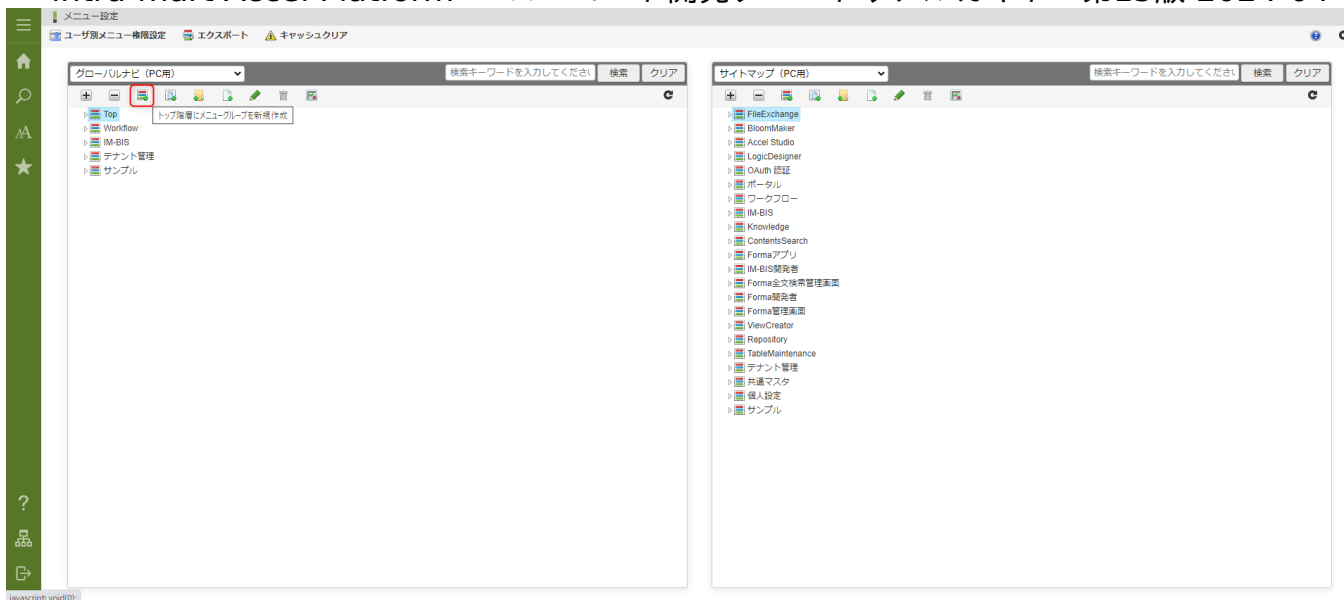
Accel-Mart Quick環境でのメニュー設定手順については、「[Accel-Mart Quick 開発者ポータル](#)」の「[メニュー設定](#)」を参照してください。

メニューの設定

「サイトマップ」→「テナント管理」→「メニュー」の順に遷移し、「メニュー設定」画面を表示します。



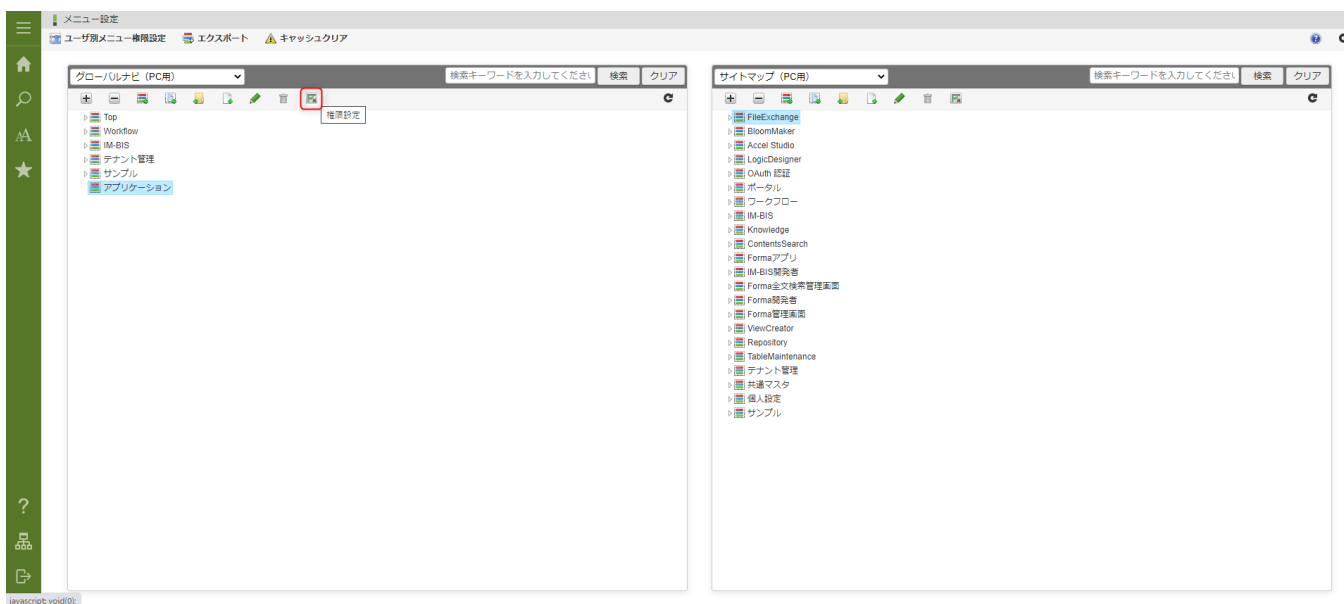
グローバルナビ (PC用) の|menu_top_create_group| をクリックし「メニューグループの新規作成」ダイアログを表示します。



「メニューグループの新規作成」ダイアログに以下を設定します。

ラベル	入力値
メニューグループID	application
メニューグループ名	アプリケーション

作成したメニューグループを選択して  をクリックし、「認可設定」画面を表示します。



「権限設定を開始する」をクリックします。


作成したメニューグループの「認証済みユーザ」の「管理」と「参照」を許可し、「権限設定を終了する」をクリックしてから「認可設定」画面を閉じます。

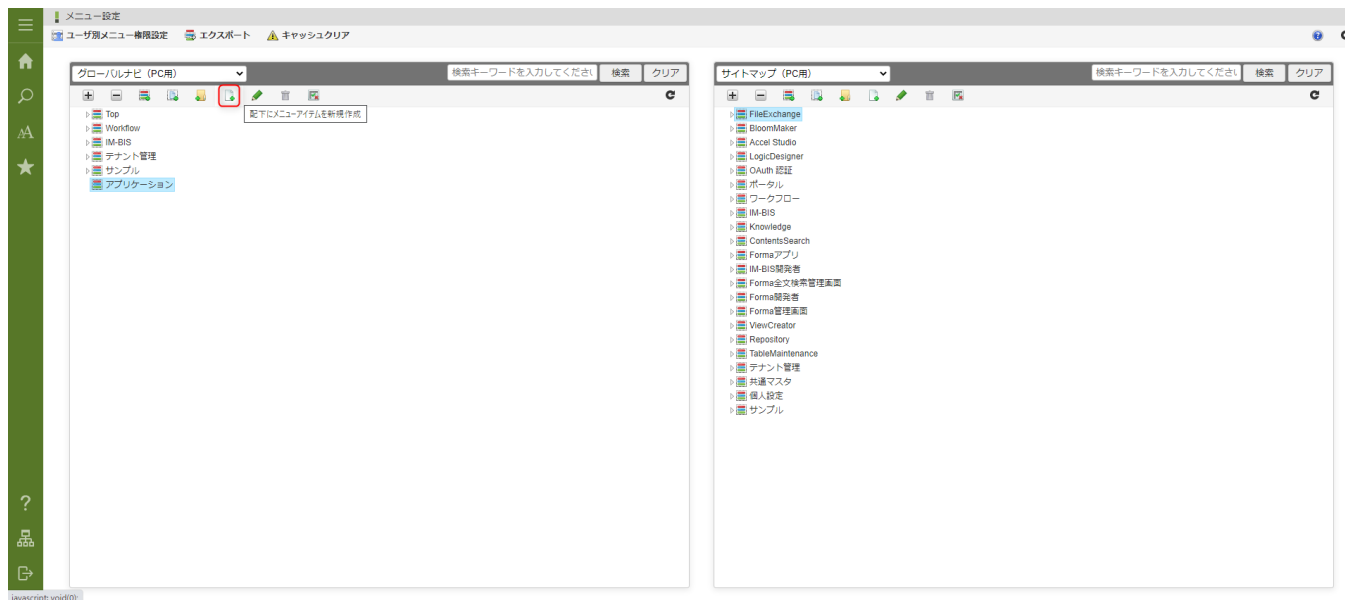
アクションの種類 全てのアクション 権限設定を開始する

リソース	アクション	認証		組織		ロール										
		ゲストユーザ	認証済みユーザ	サンプル会社	その他会社	テナント管理者	認可管理者	メニュー管理者	メニュー通用管理者	アカウント管理者	ロール管理者	カレンダー管理者	ジョブスケジューラ管理者	IM共通マスタ管理者	IMマスタ管理者	
メニューグループ グローバルナビ (PC用)	管理	✖	✖	✖	✖	✔	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖
	参照	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
Workflow	管理	✖	✖	✖	✖	✔	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖
	参照	✖	✖	✖	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
IM-BIS	管理	✖	✖	✖	✖	✔	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖
	参照	✖	✖	✖	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
テナント管理	管理	✖	✖	✖	✖	✔	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖
	参照	✖	✖	✖	✖	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
サンプル	管理	✖	✖	✖	✖	✔	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖
	参照	✔	✔	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖
アプリケーション	管理	✖	✔	✖	✖	✔	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖
	参照	✖	✔	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖	✖

注意

Accel-Mart Quick 環境ご利用の方は「アプリケーション」配下のみメニューの設定が可能です。

「アプリケーション」メニューグループを選択して  をクリックし、「メニューアイテムの新規作成」ダイアログを表示します。



アンケート回答画面のメニュー設定を行います。

「メニューアイテムの新規作成」ダイアログに以下を設定します。

ラベル	入力値
メニューアイテムID	tutorial_enquete_question
メニューアイテム名	従業員満足度調査
URL	tutorial_enquete/answer

「従業員満足度調査」のアンケートを表示するため、引数の設定を行います。

「メニューアイテムの新規作成」ダイアログの「引数」 - 「行追加」をクリックし、引数入力欄を追加します。

メニューアイテムの新規作成

メニューアイテムID * tutorial_enquete_question

メニューアイテム名 *

英語 * 従業員満足度調査

日本語 従業員満足度調査

中国語 (中華人民共和国) 従業員満足度調査

URL * tutorial_enquete/answer

呼び出し方法 GET

引数

+ 行追加 - 選択行削除

<input type="checkbox"/> キー	値
<input type="checkbox"/>	

アイコン画像

標準 ファイルパス

新規作成

追加した引数入力欄の「キー」に「enqueteld」、「値」に「tutorial_es_enquete」と入力します。

メニューアイテムの新規作成
?
✕

メニューアイテムID *

メニューアイテム名 *

英語 *	<input style="width: 80%;" type="text" value="従業員満足度調査"/>
日本語	<input style="width: 80%;" type="text" value="従業員満足度調査"/>
中国語 (中華人民共和国)	<input style="width: 80%;" type="text" value="従業員満足度調査"/>

URL *

呼び出し方法

引数

+ 行追加
 - 選択行削除

<input type="checkbox"/> キー	値
<input type="checkbox"/> enqueteld	tutorial_es_enquete

アイコン画像


標準

ファイルパス

ス

「メニューアイテムの新規作成」ダイアログの「新規作成」をクリックし、メニューを登録します。

次に、アンケート集計画面のメニュー設定を行います。

回答画面と同様に、「アプリケーション」メニューグループを選択して  をクリックし、「メニューアイテムの新規作成」ダイアログで以下を設定します。

ラベル	入力値
メニューアイテムID	tutorial_enquete_totalize
メニューアイテム名	従業員満足度調査集計画面
URL	tutorial_enquete/totalize

回答画面と同様に、「従業員満足度調査」のアンケートを表示するため、引数の設定を行います。

「メニューアイテムの新規作成」ダイアログの「行追加」をクリックし、引数入力欄を追加します。

メニューアイテムの新規作成

メニューアイテムID * tutorial_enquete_totalize

メニューアイテム名 *

英語 *	従業員満足度調査集計画面
日本語	従業員満足度調査集計画面
中国語 (中華人民共和国)	従業員満足度調査集計画面

URL * tutorial_enquete/totalize

呼び出し方法 GET

引数

+ 行追加 - 選択行削除

キー	値
<input type="checkbox"/>	
<input type="checkbox"/>	

アイコン画像

標準 ファイルパス

新規作成

追加した引数入力欄の「キー」に「enqueteld」、「値」に「tutorial_es_enquete」と入力します。

メニューアイテムの編集

メニューアイテムID * tutorial_enquete_totalize

メニューアイテム名 *

英語 *	従業員満足度調査集計画面
日本語	従業員満足度調査集計画面
中国語 (中華人民共和国)	従業員満足度調査集計画面

URL * tutorial_enquete/totalize

呼び出し方法 GET

引数

+ 行追加 - 選択行削除

キー	値
<input type="checkbox"/>	
<input type="checkbox"/> enqueteld	tutorial_es_enquete

アイコン画像

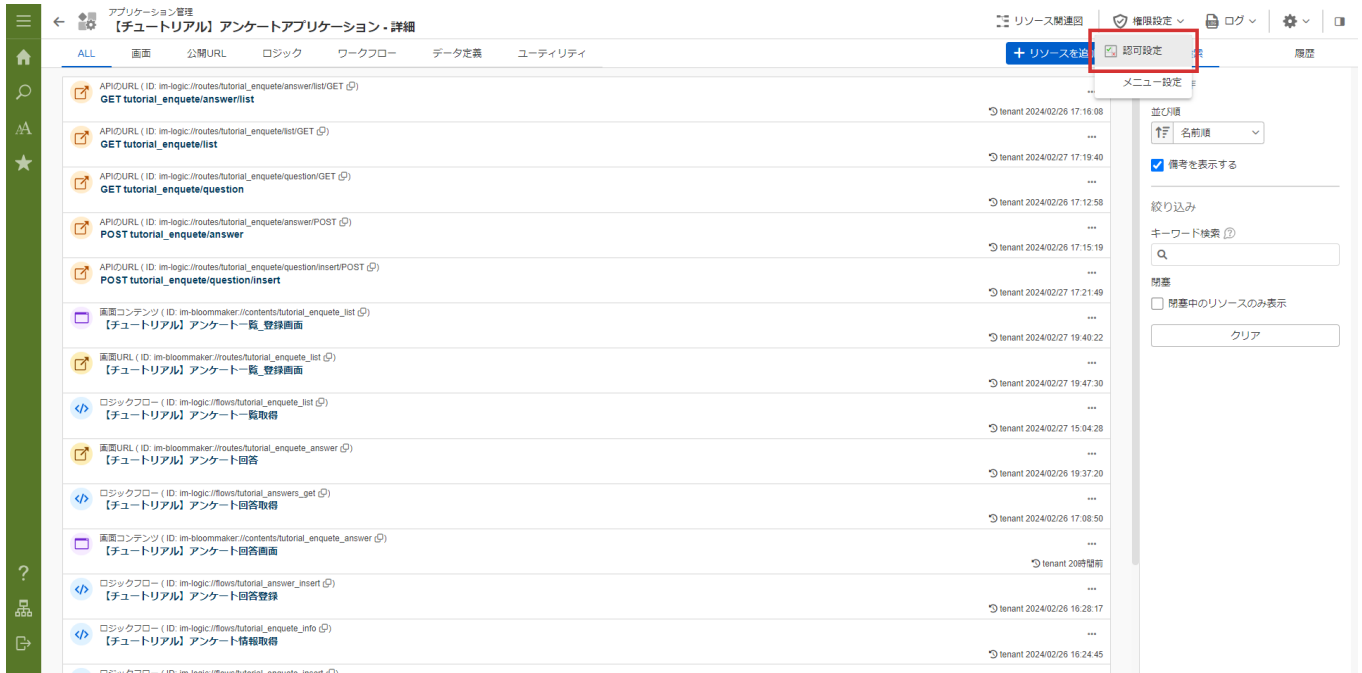
標準 ファイルパス

「メニューアイテムの新規作成」ダイアログの「新規作成」ボタンをクリックし、メニューを登録します。これで、メニューの設定は終了です。続けて、作成したアプリケーション全体の認可設定を行います。

作成したアプリケーションを利用するには、メニューへの認可設定のみではなく、アプリケーションへの認可設定も必要です。
作成したアプリケーションは標準では「テナント管理者」ロールに対してのみ認可が設定されています。
そのため、このままでは「テナント管理者」ロールを持ったユーザ以外がアプリケーションを利用できません。
テナント管理者以外もアプリケーションを利用できるように認可設定を行います。
すでに「[業務ロジックのURLの認可設定](#)」で業務ロジックのURLの認可設定を行いました。
この節ではアンケートアプリケーションの各画面を呼び出す画面URLの認可設定を行います。

認可設定

アプリケーションの認可設定はアプリケーション管理画面から行います。
「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】アンケートアプリケーション」の順に遷移します。
画面右上の「権限設定」プルダウンから認可設定アイコンを選択します。



アプリケーション認可設定画面で、「認証」 - 「認証済みユーザ」の列を選択し、「選択中の認可を許可」をクリックします。

The screenshot shows the 'Authorization Settings' table for the 'Tutorial Enquete Application'. The table has columns for 'Resource', 'Authentication', and various roles. The 'Authentication' column is highlighted in yellow, and the 'Allow selected permissions' button is visible at the top.

リソース	認証	ゲストユーザ	認証済みユーザ	組織	サンプル会社	その他会社	テナント管理者	認可管理者	メニュー管理者	メニュー運用管理者	アカウント管理者	ロール管理者	カレンダー管理者	ジョブスケジューラ管理者	IM共通マスタ運用管理者	IM共通マスタ運用管理者	ポータル管理者	IM-Workflow管理者	IM-Workflow運用管理者	IM-Workflow監査者	IM-Workflowユーザ	
【画面URL】																						
【チュートリアル】アンケート集計画面	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【チュートリアル】アンケート回答	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【API/URL】																						
GET tutorial_enquete/answerlist	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
POST tutorial_enquete/answer	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
GET tutorial_enquete/question	×	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×

「更新」をクリックし、認可設定の更新を行います。

アプリケーション設定
【チュートリアル】 アンケートアプリケーション - 認可設定

選択中の認可を許可 × 選択中の認可を禁止 ⚠ 設定が変更されています。更新ボタンをクリックしてください。

リソース	認証	組織	ロール	テナント管理 者	認可管理 者	メニュー管理 者	メニュー適用 管理者	アカウント管 理者	ロール管理 者	カレンダー管 理者	ジョブスケ ジュール管理 者	IM共通マス タスタ管理 者	IM共通マス タスタ適用 管理者	ポータル管 理者	IM-Workflow 管理 者	IM-Workflow 適用管理 者	IM-Workflow 監査 者	IM-Work ユーザ
【匿名URL】																		
【チュートリアル】 アンケ ート集計画面	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【チュートリアル】 アンケ ート回答	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
【APIのURL】																		
GET tutorial_enquete/answer/list	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
POST tutorial_enquete/answer	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
GET tutorial_enquete/question	×	✓	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×

アプリケーション認可情報更新確認
⚠ アプリケーション認可情報を更新します。よろしいですか？
キャンセル 更新

これでログイン可能なすべてのユーザが、アンケートアプリケーションを利用できるようになりました。

IM-Sign と連携した電子サインアプリケーション

概要

本章では、ローコード開発により電子サイン連携をするアプリケーションを作成します。

注意

本チュートリアルでは「[環境について](#)」に記載の環境を前提に、機能間の遷移順序を記述しています。
詳しくは上記リンクからご確認ください。

作成の流れ

本チュートリアルは以下の流れでアプリケーションを作成します。

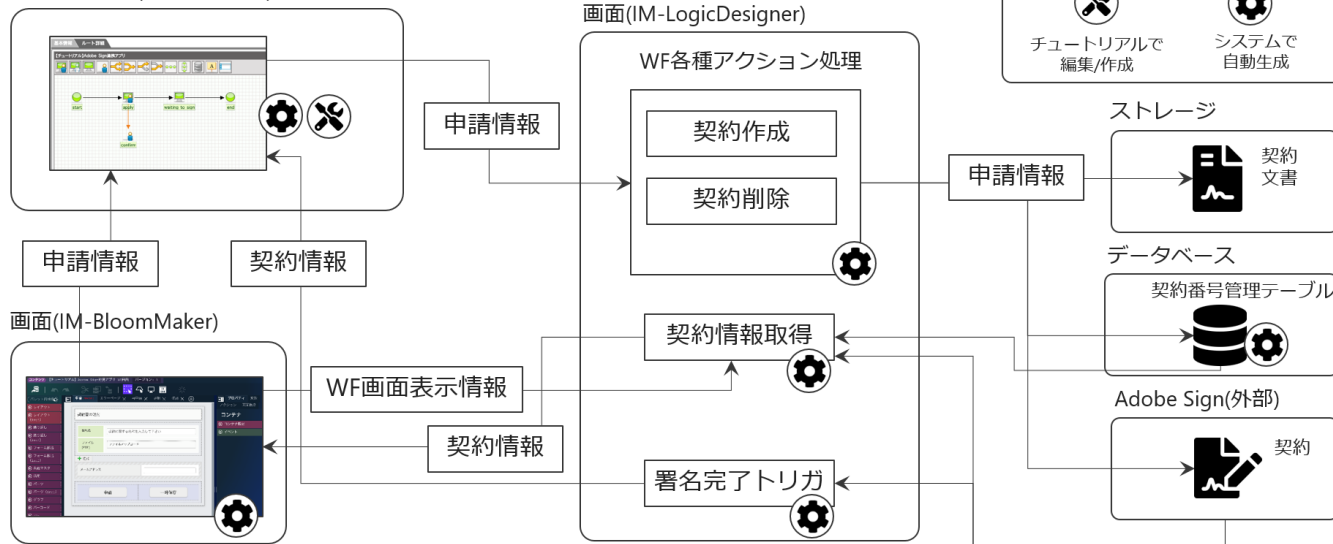
1. アプリケーション作成前に必要な準備作業
2. アプリケーション作成にてテンプレートからデータベース、処理、画面のベースを自動作成

IM-Sign 連携アプリケーションの仕様

作成する IM-Sign 連携アプリケーションは、ワークフローの申請画面から Acrobat Sign で電子サイン契約の登録を行えます。
ワークフローの申請画面から、電子サイン契約に登録する、署名文書と署名先メールアドレスを登録します。

Acrobat Sign 側で登録した電子サイン契約が完了すると、契約完了の通知を Acrobat Sign から受け取り、ワークフローを完了します。

ワークフロー(IM-Workflow)



登録・確認/参照画面

作成する IM-Sign 連携アプリケーションでは、ワークフローの申請画面よりAcrobat Signに登録する契約内容を設定します。ワークフローの状況によって、登録、確認、参照画面のどの画面なのかを切り替えて表示します。登録画面では以下の項目についての登録が行えます。

ラベル	設定値
契約名	Acrobat Signで作成する契約の契約名を設定します。
ファイル(PDF)	Acrobat Signの契約を行う書面をPDF形式でアップロードします。
メールアドレス	Acrobat Signで契約を行う際、署名を求めるメールアドレスを設定します。 署名を求めるメールアドレスは複数設定可能です。

契約書の送付

契約名

ファイル(PDF)

+ 追加

メールアドレス

確認/更新画面では、以下の項目の表示・およびダウンロードが可能です。

ラベル	設定値
契約名	Acrobat Signで作成する契約の契約名を表示します。
ファイル(PDF)	Acrobat Signの契約を行う書面をダウンロードします。
メールアドレス	Acrobat Signで契約を行う際、署名を求めるメールアドレスを表示します。

契約書の送付	
契約名	[デモ用] チュートリアル用契約
ファイル(PDF)	ダウンロード
メールアドレス	XXXXXXXXXX@XXXXXXXXXX

注意事項

本アプリケーションではサンプルデータのユーザや組織を利用しています。
 チュートリアル中のユーザや組織の指定は必ずしもチュートリアル通りに入力しなくても動作します。
 このテンプレートはAccel-Mart Quick専用のテンプレートのため、intra-mart AccelPlatformではご確認できません。

事前準備

本章では、アプリケーションの作成前に必要な、事前準備について記載しています。

Acrobat Signの開発アカウント作成

「Adobe Acrobat Sign連携 標準テンプレート」で作成したアプリケーションを利用するには、Acrobat Signの「ビジネス/エンタープライズ版アカウント」が必要です。

無償の開発者用アカウントもあります。

<https://www.adobe.com/jp/sign/developer-form.html>

事前にいずれかのAcrobat Signアカウントをご用意ください。

Acrobat Signの開発アカウント設定

以下は、Acrobat Signアカウントに対して設定する項目です。

Acrobat Sign 連携に必要な設定(API連携)

Acrobat Sign連携を行う際に必要なAPIの設定をします。

Acrobat Signサイトにログインし、アカウント設定で「APIアプリケーション」および「OAuth」の設定を追加します。

手順については、以下のドキュメントを参照ください。

「[IM-Sign利用ガイド](#)」 - 「[Acrobat Signアプリケーション設定](#)」

作成

Webhookを作成して、アプリケーションのHTTPS URLで選択したイベントに対応するリアルタイム通知を受け取ります。[セキュリティ設定](#)でクライアント証明書をアップロードして、Webhookを保護します。

名前:

スコープ:

URL:

イベント:

通知/パラメーター:

- 契約情報
- 契約文書情報
- 契約書の参加者情報
- 署名済み契約文書
- 一括送信情報
- Web フォーム情報
- Web フォーム文書情報
- Web フォーム参加者の情報

i コラム

Accel-Mart Quickにおいて、「OAuth」の「リダイレクトURI」に設定するURIは以下の通りです。
「ホスト名」はご利用の環境によって変更します。
<https://ホスト名.accel-mart.com/imart/oauth/redirect>
その他の設定は上記利用ガイドに従ってください。

Acrobat Sign 連携に必要な設定(Webhook)

Acrobat Sign連携を行う際に必要なWebhookの設定をします。
Webhookの設定を行うことで、Acrobat Sign側で署名が完了した際にAccel-Mart Quick側に通知が可能です。
Acrobat Signサイトの「Webhooks」より、Webhook登録を行います。
手順については、以下のドキュメントを参照ください。
[「IM-Sign利用ガイド」](#) - [「Acrobat Sign上のWebhook設定」](#)

OAuth を設定

クライアント ID: [REDACTED]

リダイレクト URI:

注意: OAuth 要求で指定した redirectUri は、この URI リストに含まれている必要があります。コンマ区切りのリストとして複数の URI を指定できます。

有効な範囲

OAuth プロトコルによる要求で使用する範囲を有効にする必要があります。有効にする範囲は、アプリケーションに最低限必要なセットのみにしてください。これは証明の要件の1つです。

アプリケーションの有効な範囲を変更する必要がある場合は、[サポートにご連絡ください](#)。

「group」範囲修飾子を使用する OAuth 要求はグループ管理者のみが承認でき、「account」範囲修飾子を使用する OAuth 要求はアカウント管理者のみが承認できます。

有効?	範囲	修飾子	説明
<input checked="" type="checkbox"/>	user_read	account	アカウント内のユーザーを表示
<input checked="" type="checkbox"/>	user_write	account	アカウント内のユーザーを作成または管理
<input checked="" type="checkbox"/>	user_login	account	ログインアクセス - 他のリクエストを上書きするフルアクセスをアカウント内の任意のユーザーに提供
<input checked="" type="checkbox"/>	agreement_read	account	アカウント内の任意のユーザーの代理で文書およびデータにアクセス
<input checked="" type="checkbox"/>	agreement_write	account	アカウント内の任意のユーザーの代理で文書のステータスを管理
<input type="checkbox"/>	agreement_send	account	アカウント内の任意のユーザーの代理で文書を送信
<input checked="" type="checkbox"/>	widget_read	account	アカウント内の任意のユーザーの代理で Web フォームを表示
<input checked="" type="checkbox"/>	widget_write	account	アカウント内の任意のユーザーの代理で Web フォームを作成

i コラム

Accel-Mart Quickにおいて、「Webhooks」の「Webhook用URL」に設定するURLは以下の通りです。
「ホスト名」はご利用の環境によって変更します。
<https://ホスト名.accel-mart.com/imart/adobesign/webhook>
その他の設定は上記利用ガイドに従ってください。

上記の設定ができれば、設定は完了です。

テンプレートからアプリケーションのベースを作る

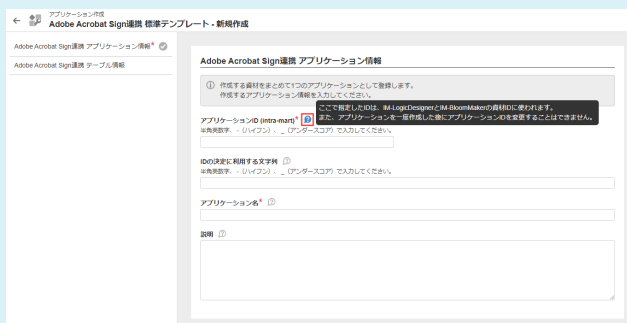
アプリケーション作成でテンプレートからアプリケーションを自動で作成します。
本章では、「Adobe Acrobat Sign連携 標準テンプレート」を利用してアプリケーションを作成します。

アプリケーション作成のテンプレートからアプリケーションの作成

「サンプルアプリケーション」から「Adobe Acrobat Sign連携 標準テンプレート」を選択します。
「Adobe Acrobat Sign連携 標準テンプレート」に利用することで、IM-Sign 連携で利用する画面、データベースのテーブル、各種処理、ワークフローが自動生成されます。

i コラム

アプリケーションを作成する際、入力項目について不明点あれば各項目名の横にある **?** にカーソルを合わせてください。アプリケーション利用時にどのように利用される項目であるか、設定を変えるとどのような変化があるかの説明が表示されます。



アプリケーション情報の設定

「サイトマップ」→「Accel Studio」→「アプリケーション作成」→「業務テンプレート一覧」の順に遷移します。

「サンプルアプリケーション」から「Adobe Acrobat Sign連携 標準テンプレート」を選択します。

アプリケーションの基本情報を設定します。

アプリケーションID、アプリケーション名、説明に以下を設定します。

※アプリケーション情報に関しては、各項目に任意の入力値を設定してもかまいません。本チュートリアルは、下記の通りのアプリケーションID、アプリケーション名を入力した前提で進めます。

ラベル	入力値
アプリケーションID	tutorial_im_sign
アプリケーション名	【チュートリアル】 Adobe Acrobat Sign 連携アプリ
説明	Adobe Acrobat Sign 連携アプリアプリケーションです。

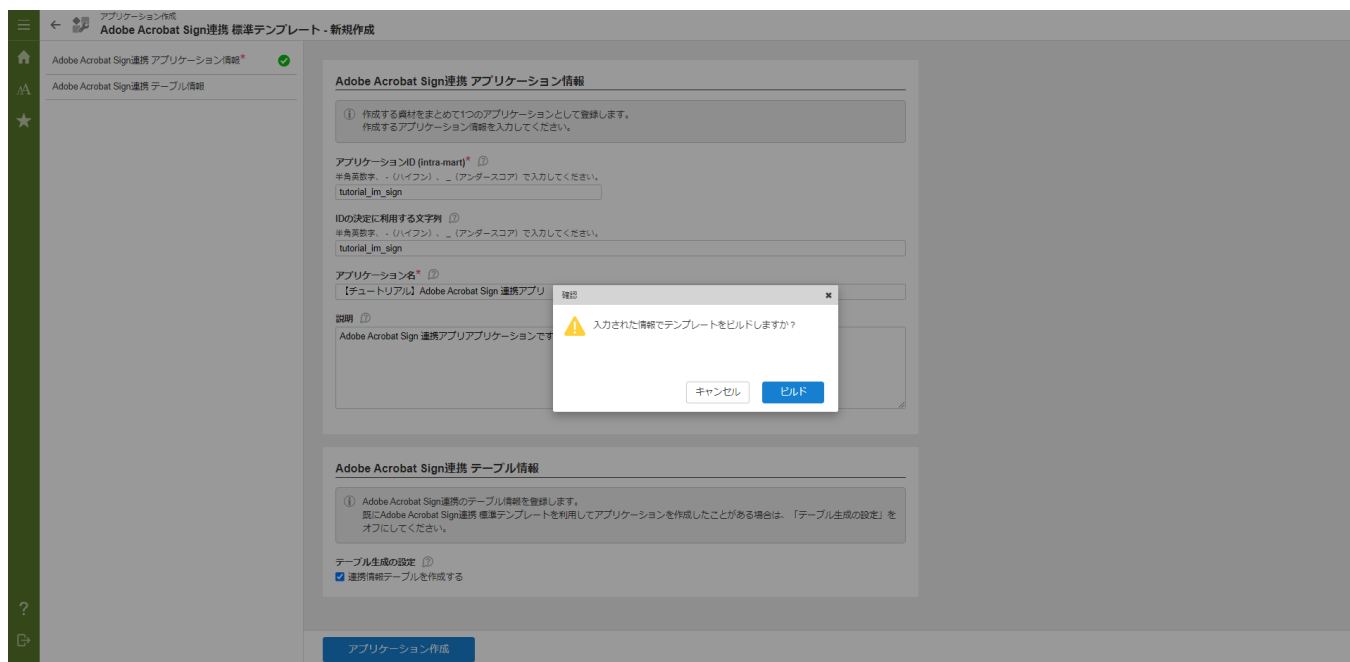


Adobe Acrobat Sign 連携テーブル情報の登録

Adobe Acrobat Sign 連携のテーブル情報を登録します。

本チュートリアル実施以前に「Adobe Acrobat Sign 連携 標準テンプレート」を利用してアプリケーションを作成したことがある場合は、「テーブル生成の設定」の「連携情報テーブルを作成する」のチェックボックスを外してください。

上記の設定が完了したら、「アプリケーション作成」をクリックし、確認ダイアログの「ビルド」をクリックします。



注意

すでに「Adobe Acrobat Sign 連携 標準テンプレート」を利用してアプリケーションを作成したことがある場合は、以下の点に注意してください。

- ・ 新しく入力した認証情報で、以前に入力した認証情報は上書きされます
- ・ 「テーブル生成の設定」をオフにしてください

テンプレートからデータベースや画面などが自動で生成されます。
画面のステータス項目がCOMPLETEになれば、テンプレートからの自動作成は完了です。

アプリケーション作成
テンプレートビルドジョブ - 詳細

ビルドジョブID	8gtmb90bglq8h1l	テンプレート名	Adobe Acrobat Sign連携 標準テンプレート
ステータス	COMPLETE	作成者ユーザコード	soyagi
作成日時	2023/06/01 17:27:24		
出日日時	メッセージ	詳細	
2023/06/01 17:27:24	生成するリソースのIDを自動設定します。		
2023/06/01 17:27:25	一括ディレクトリに IM-LogicDesigner インポート資料を作成しました。		
2023/06/01 17:27:26	一括ディレクトリに IM-BloomMaker インポート資料を作成しました。		
2023/06/01 17:27:26	一括ディレクトリに IM-Workflow インポート資料を作成しました。		
2023/06/01 17:27:26	アプリケーションで使用するテーブルを新規作成しました。		
2023/06/01 17:27:26	CREATE TABLE im_adobesign_linkage(system_matter_id varchar(20), agreement_id varchar(100) NOT NULL, PRIMARY KEY(system_matter_id));		
2023/06/01 17:27:27	自動生成された IM-LogicDesigner 資料をインポートしました。		
2023/06/01 17:27:28	自動生成された IM-BloomMaker 資料をインポートしました。		
2023/06/01 17:27:30	自動生成された IM-Workflow 資料をインポートしました。		
2023/06/01 17:27:31	アプリケーションの作成が完了しました。		
2023/06/01 17:27:31	アプリケーション認可設定で Accel-Mart Quick アプリケーション管理者 にアクセス許可が設定されました。		

ワークフロー設定

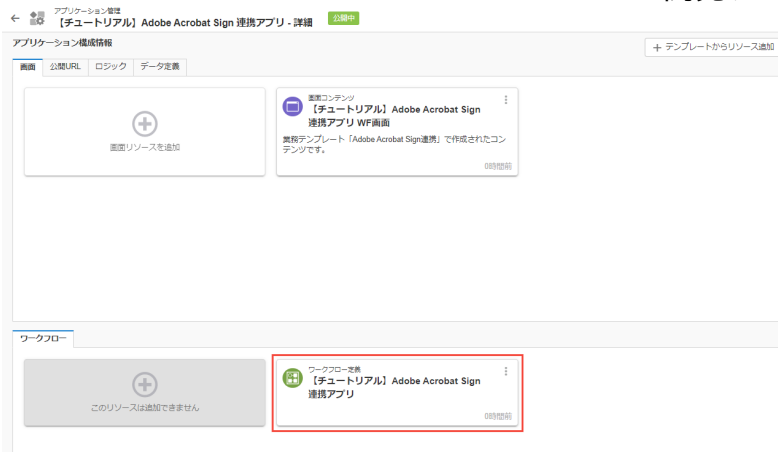
テンプレートから作成したワークフローは、利用可能なユーザやロールが割り当てられていないため、ワークフローの設定を行います。

ルート定義の修正

ワークフローのルート定義を修正します。

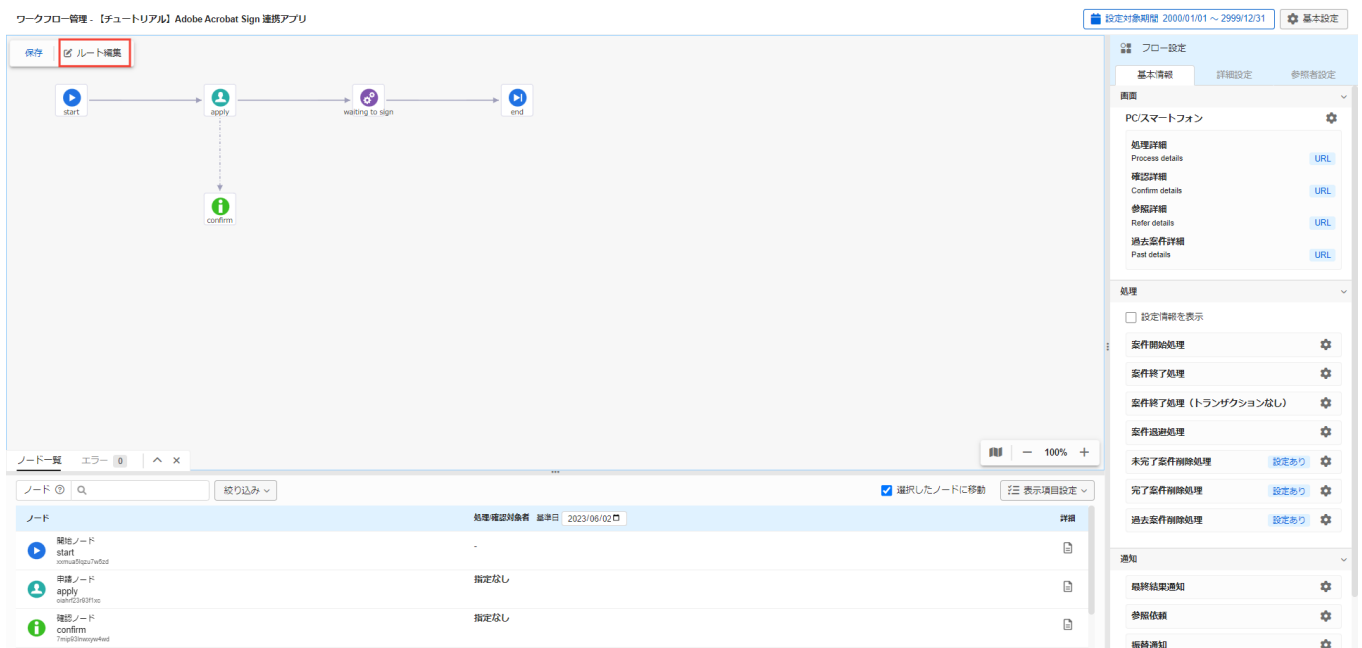
「サイトマップ」→「Accel Studio」→「アプリケーション管理」→「アプリケーション一覧」→「【チュートリアル】 Adobe Acrobat Sign 連携アプリ」の順に遷移します。

アプリケーション管理画面のワークフロータブ内にある、「ルート定義」 - 「【チュートリアル】 Adobe Acrobat Sign 連携アプリ」をクリックします。

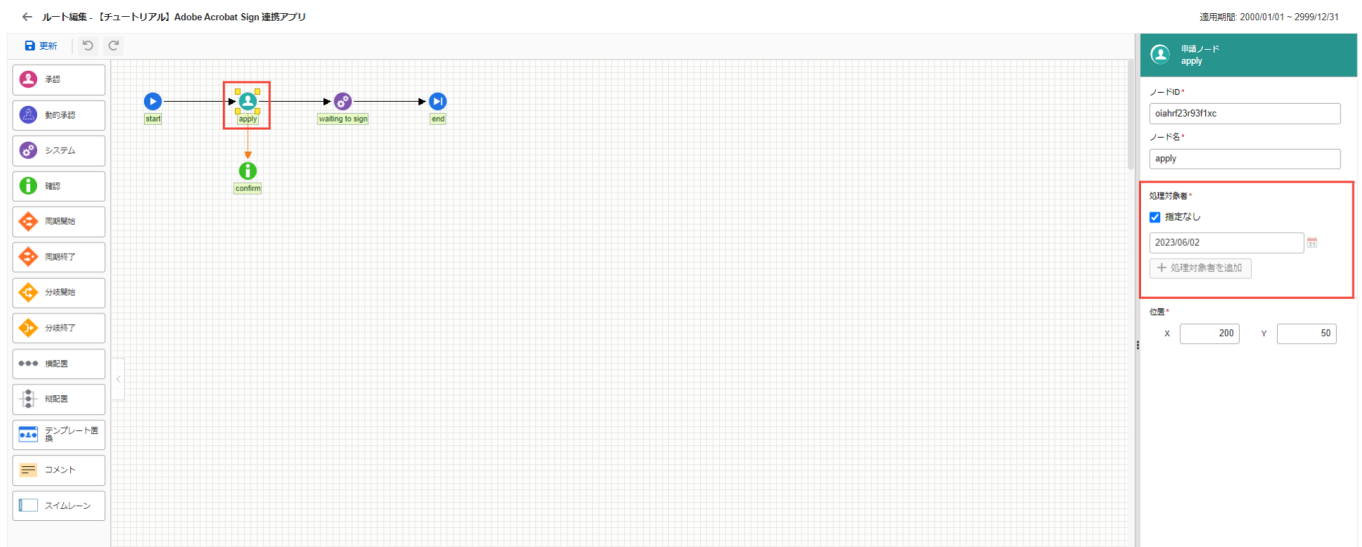


ルートの修正

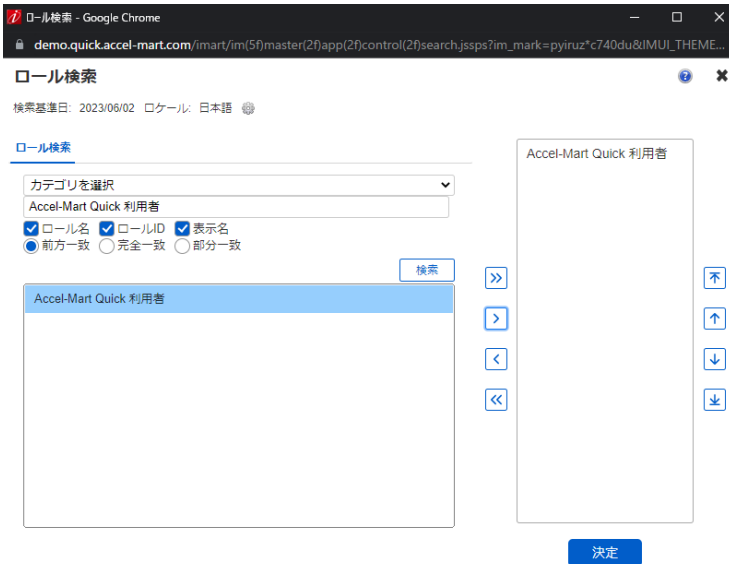
ワークフロー管理画面左上の「ルート編集」をクリックし、ルート編集画面へ遷移します。



ワークフローを利用可能なユーザを指定します。
「apply」のノードアイコンをクリックします。



画面右側領域で「処理対象者」の「指定なし」のチェックボックスからチェックを外します。
日付入力ボックスに処理対象者の処理権限開始期間を指定します。
「+処理対象者を追加」をクリックし、処理対象者選択ダイアログの「ロール」をクリックします。
ロール検索ダイアログで「Accel-Mart Quick 利用者」を選択し、「決定」をクリックします。



設定が完了したら、ルート編集画面左上の「更新」をクリックします。
これでルート編集は完了です。

ワークフローを確認

上記の設定が正常に行えているかをワークフロー申請機能経由で確認します。
「Accel-Mart Quick 利用者」ロールを持つユーザで環境にログインします。
「サイトマップ」→「ワークフロー」→「一覧」→「申請一覧」の順に遷移します。
「フロー」タブを開き、一覧に「【チュートリアル】Adobe Acrobat Sign 連携アプリ」があることを確認します。



上の図のようにアプリケーションが表示されていれば正常に設定は完了しています。

IM-Sign 連携テンプレートの使い方

本章では、「Adobe Acrobat Sign連携 標準テンプレート」を利用して作成したアプリの使い方を説明します。

外部連携アプリケーションの登録

Acrobat Sign連携を行う際に「外部連携アプリケーションの登録」が必要です。
外部連携アプリケーションの登録は、Acrobat Sign連携を利用するユーザごとに登録が必要です。
メニューの「個人設定」から「外部連携アプリケーション」をクリックします。



Acrobat Signサイトにて認証

「Acrobat Sign API」の「許可」ボタンをクリックします。



Acrobat Signサイトに移動してログイン、および認証を行ってください。



im-signテンプレートのアクセスを確認

im-signテンプレート が次の操作の権利を求めています。

- アカウント内のユーザーを表示
- アカウント内のユーザーを作成または管理
- ログインアクセス - 他のリクエストを上書きするフルアクセスをアカウント内の任意のユーザーに提供
- アカウント内の任意のユーザーの代理で文書およびデータにアクセス
- アカウント内の任意のユーザーの代理で文書のステータスを管理
- アカウント内の任意のユーザーの代理で Web フォームを表示
- アカウント内の任意のユーザーの代理で Web フォームを作成、編集または公開
- アカウント内の任意のユーザーの代理でテンプレートおよび文書ライブラリを表示
- アカウント内の任意のユーザーの代理でテンプレートおよび文書ライブラリを管理
- アカウント内の任意のユーザーの代理でワークフローを表示
- アカウント内の任意のユーザーの代理で Webhook を表示
- アカウント内の任意のユーザーの代理で Webhook を作成または編集
- アカウント内の任意のユーザーの代理で Webhook を完全に削除

im-signテンプレート があなたのアカウントに継続してアクセスします (失効させた場合は除きます)。



Acrobat Signサイトで認証後、外部連携アプリケーションの登録画面の「利用中のアプリケーション」にAcrobat SignAPIアイコンが表示されることを確認してください。

外部連携アプリケーション

「Adobe Sign API」の利用を許可しました。

利用中のアプリケーション

アプリケーション名: Adobe Sign


Adobe Sign API
 Adobe Sign API の利用を許可します。これによりAdobe Sign機能との連携が可能となります。
 解除

利用可能なアプリケーション

利用可能なアプリケーションはありません。

申請画面を使用して契約情報を登録

申請メニューより前章で作成した「【チュートリアル】Acrobat Sign連携アプリ」を選択します。
 ルート登録で権限を与えたユーザにて操作をしてください。

申請画面

申請画面でAcrobat Signに連携する契約内容を入力します。

☰
🏠
AA
★

契約書の送付

契約名

ファイル(PDF)
🗑️

+ 追加

メールアドレス

申請
一時保存

各入力項目の説明は以下の通りです。
 入力が終わりましたら申請ボタンをクリックします。

ラベル	設定値
契約名	Acrobat Signで作成する契約の契約名を設定します。
ファイル(PDF)	Acrobat Signの契約を行う書面をPDF形式でアップロードします。
メールアドレス	Acrobat Signで契約を行う際、署名を求めるメールアドレスを設定します。 署名を求めるメールアドレスは複数設定可能です。

ワークフローの案件情報を適宜設定します。
 設定後、申請ボタンをクリックします。



コラム

「Adobe Acrobat Sign連携 標準テンプレート」で作成するアプリケーションは、申請に連動してAcrobat Signに契約情報を連携するように設定をしています

承認一覧

申請を行ったユーザにて承認メニューの「処理済 未完了案件」ボタンをクリックして案件の状況を確認します。



コラム

下記のようなエラーが表示される場合、前項の「外部連携アプリケーションの登録」が設定できていない可能性があります。前章を含めて設定の確認をしてください。

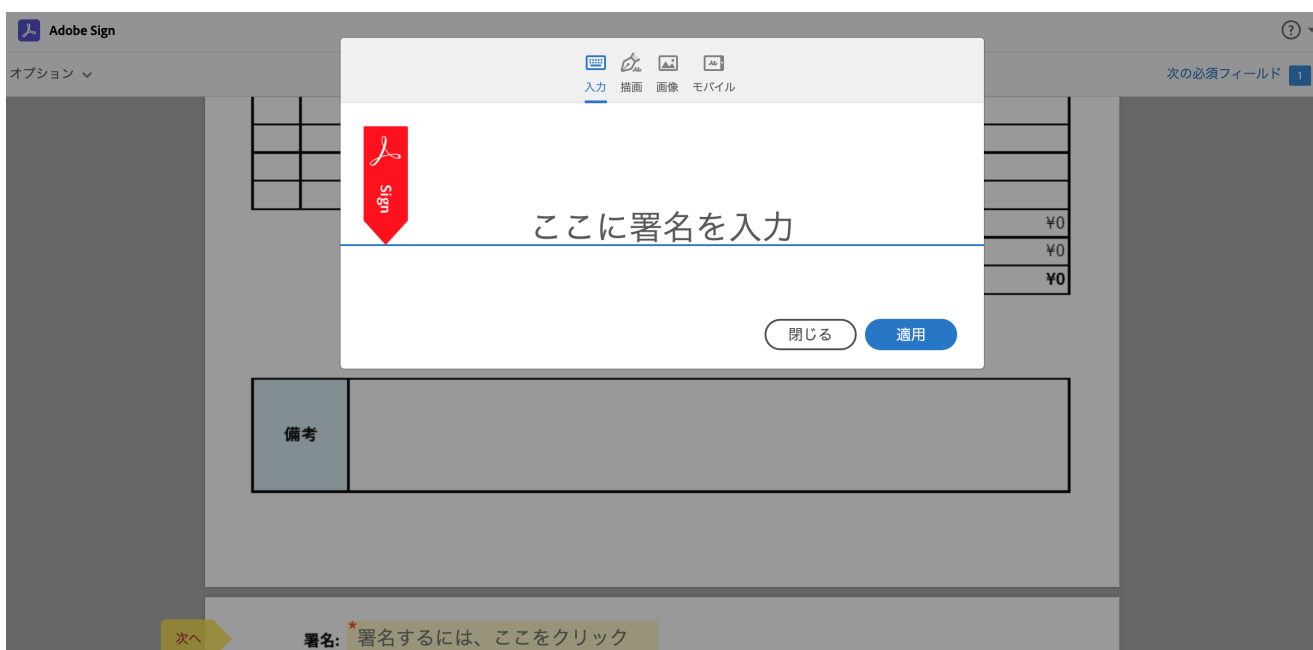


契約相手による電子サイン実施

申請後は、指定したメールアドレス宛にAcrobat Signから署名依頼を案内します。以下は、Acrobat Signから案内するメールの例です。



ワークフローに設定した契約文書に対し、契約相手に署名をしてもらいます。



契約完了を確認

全ての契約相手の署名が完了すると、ワークフローが自動的に完了状態です。

承認一覧画面の「処理済(完了案件)」にて、契約の確認、および署名済み文書のダウンロードが可能です。



署名済み文書のダウンロードをするには、「詳細」画面を表示してください。

契約書の送付	
契約名	[デモ用] チュートリアル用契約
ファイル(PDF)	ダウンロード
メールアドレス	XXXXXXXXXX@XXXXXXXXXX

本章での説明は以上です。

概要

チュートリアルで作成したアプリケーションに追加機能を作成します。
※本チュートリアルはintra-mart Accel Platform 2023 Spring(Gerbera)をベースに作成しています。

アンケートアプリケーションに追加機能を作成する

「[ゼロからアプリケーションを作成する](#)」作成した、簡単なアンケートアプリケーションに追加機能を作成します。
この章では、以下のような事が学べます。

- IM-LogicDesigner における処理の繰り返し
- IM-BloomMaker における動的な繰り返しの利用方法
- IM-BloomMaker での入力チェック

アンケートアプリケーションに機能を追加する

概要

本章では、チュートリアルで作成した簡易的なアンケートアプリケーションに追加機能を作成します。
先にチュートリアルを作成した前提で作業を行います。



注意

本チュートリアルでは「[環境について](#)」に記載の環境を前提に、機能間の遷移順序を記述しています。
詳しくは上記リンクからご確認ください。

作成の流れ

本チュートリアルはアンケートアプリケーションに「アンケート一覧画面と新規作成機能」を追加します。
本チュートリアルは以下の流れでアプリケーションを作成します。

1. IM-LogicDesignerにて業務ロジックの作成
2. IM-BloomMakerにて画面の作成

追加機能の仕様

アンケート一覧画面とアンケートの登録画面を1つのコンテンツとして追加します。
それぞれの画面の仕様は以下の通りです。

アンケート一覧画面

アンケート一覧画面では、「アンケート名」、「開始日」、「終了日」を表示します。
それぞれのアンケートの「回答画面」と「集計画面」へのリンクも用意し、アンケート登録画面へのリンクも作成します。

	新規作成					
		アンケート名	開始日	終了日	回答画面へ	集計画面へ
		従業員満足度調査	2021/01/01	2022/01/01	☑	📄
		好きな映画ジャンル調査	2021/01/01	2022/01/01	☑	📄
		好きな料理アンケート	2019/01/01	2020/01/01	☑	📄
		休日の過ごし方	2021/10/22	2021/11/05	☑	📄
		理解度チェック	2021/12/10	2021/11/04	☑	📄

アンケート登録画面

アンケート登録画面ではアンケート情報と設問を入力し、登録が行えます。
 設問数は自由に増減が可能で、全項目必須として入力チェックも行います。
 また、開始、終了日も、終了日が開始日より前にならないように入力チェックを行います。

The screenshot shows a registration form for a survey. On the left is a green sidebar with a menu icon, a home icon, and a star icon. The main form area contains the following elements:

- アンケートID**: A text input field with a red border and a small icon on the right.
- アンケートタイトル**: A text input field with a red border and a small icon on the right.
- アンケート概要**: A text input field with a red border and a small icon on the right.
- 開始日**: A date picker field with a calendar icon on the right.
- 終了日**: A date picker field with a calendar icon on the right.
- 設問**: A section with a title and two circular buttons (a green '+' and a red '-'). Below it is a table with three rows and three columns:

設問内容	左項目	右項目
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
- 登録**: A button located at the bottom center of the form.

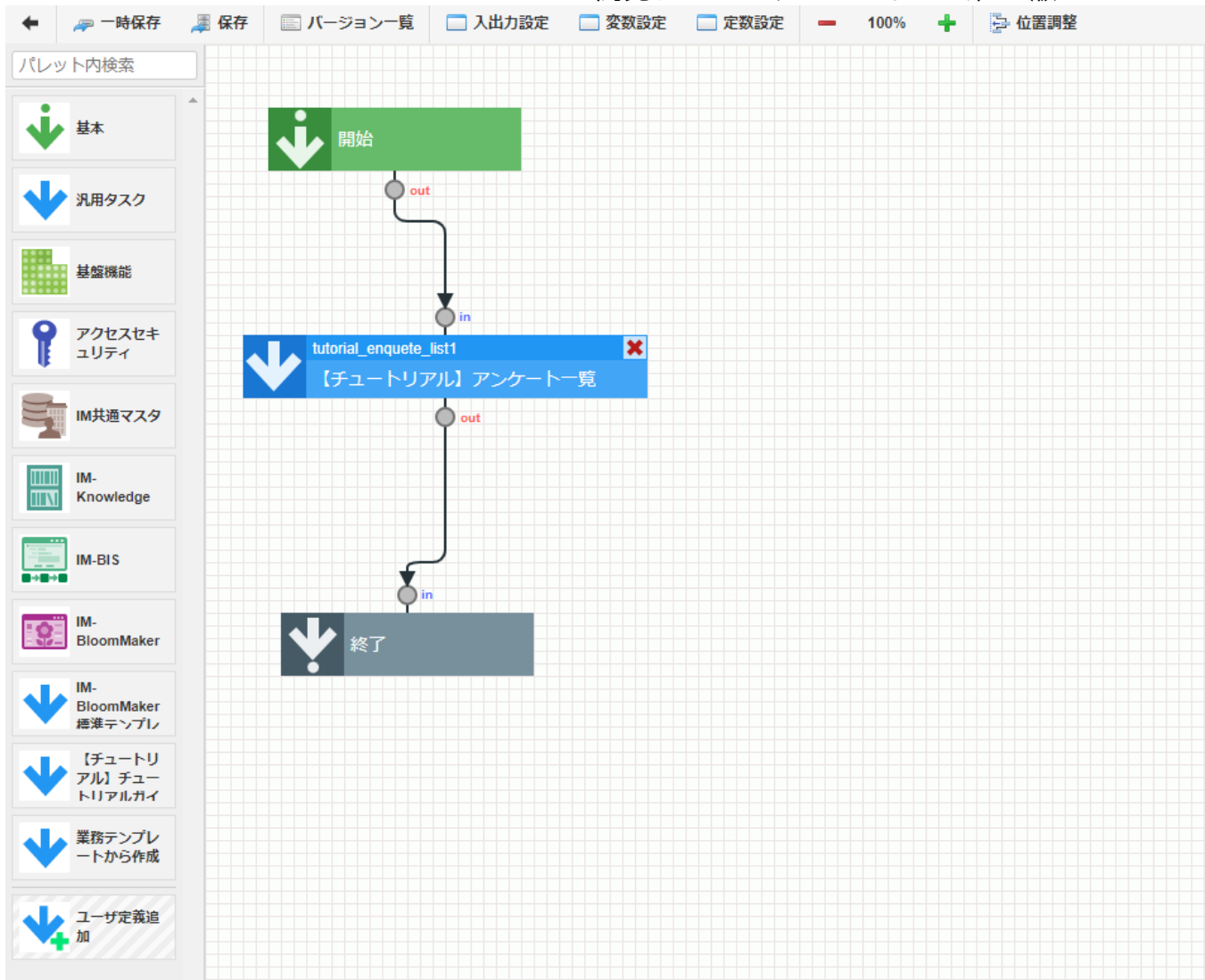
アンケート一覧、登録処理の業務ロジックの作成

アンケートアプリケーションの管理画面の業務ロジックを作成します。

- アンケート一覧取得処理
アンケート情報テーブルからアンケート一覧を取得します。
- アンケート登録処理
設定方法に合わせてアンケート情報テーブルとアンケート設問テーブルにデータを登録します。

アンケート一覧取得処理の作成

アンケート一覧取得処理では、アンケート情報テーブルからアンケートの一覧を取得します。

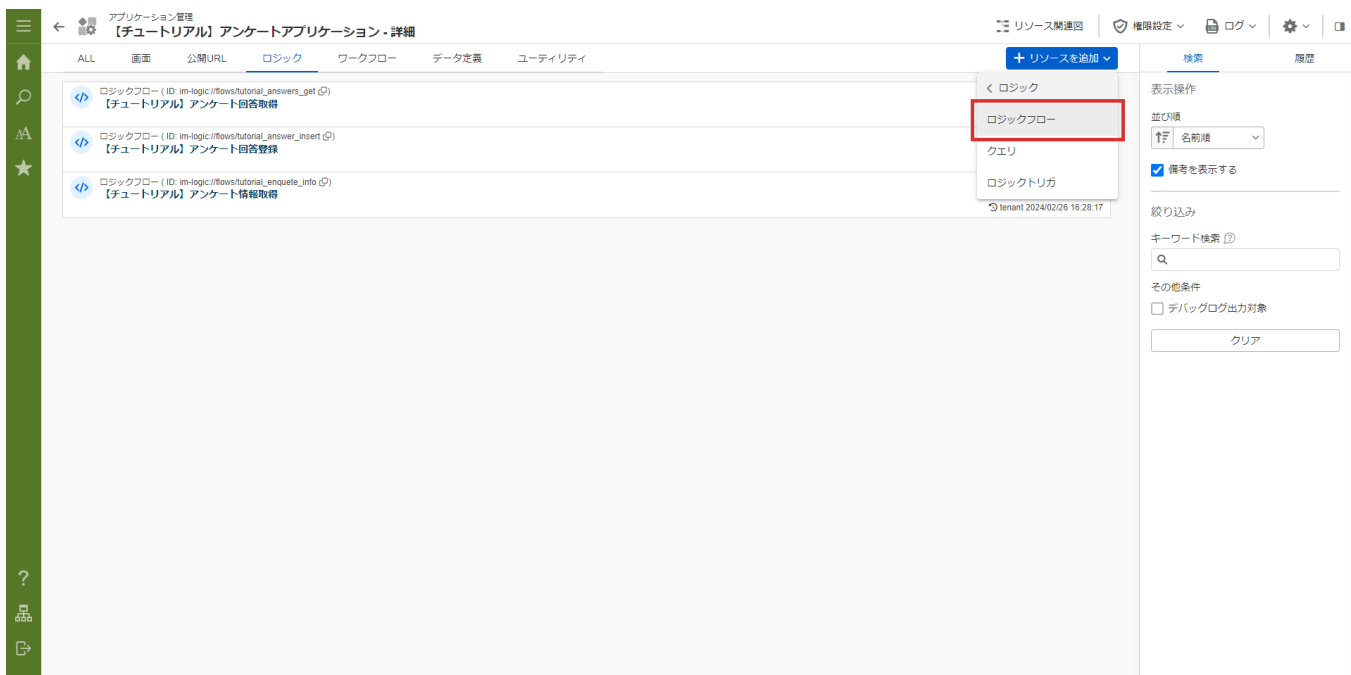


ロジックの作成

アプリケーションに紐づく業務ロジックの作成は「アプリケーション管理」から行います。

アプリケーション管理から「【チュートリアル】アンケートアプリケーション」を選択します。

「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「ロジック」 - 「ロジックを新規作成」 - 「ロジックフロー」を選択し、「ロジックフロー定義編集画面」を表示します。



入出力値の設定

業務ロジックの出力値を設定します。

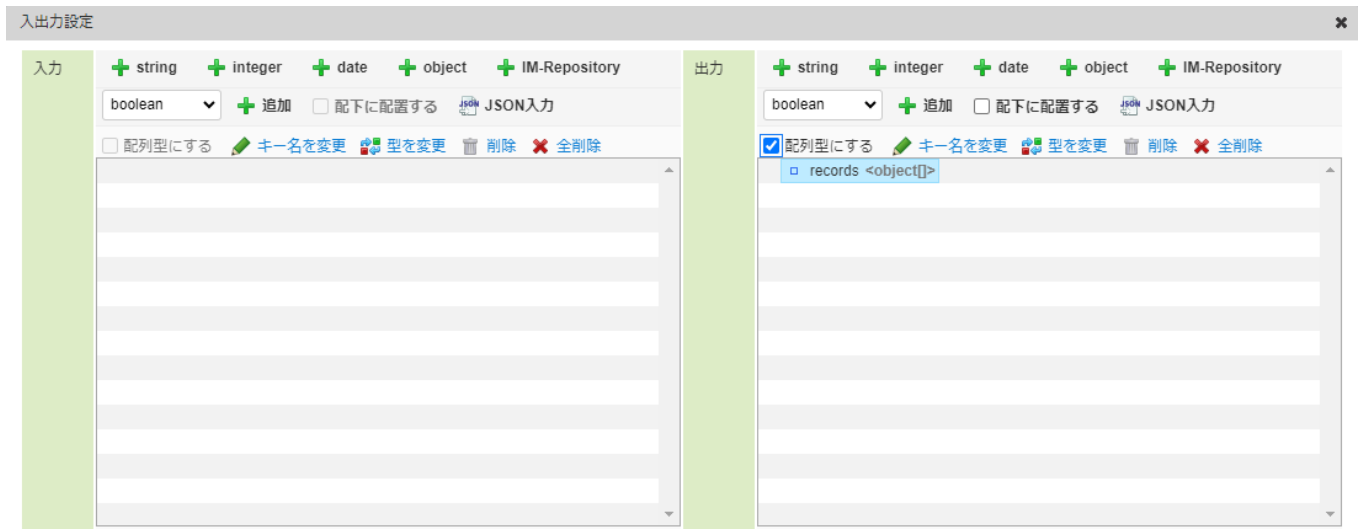
一覧の取得は検索等を行わずに全てのアンケートを表示するため、出力値のみを設定します。

まずは、アンケート一覧を格納する配列を作成します。

画面上部の「入出力設定」をクリックし「入出力設定」ダイアログを表示します。

「出力」 - 「+object」をクリックすると出力値一覧に「object1」が追加されるため、キー名を「records」と変更します。

配置した「records」を選択し、「配列型にする」を選択し、「records」を汎用オブジェクトの配列とします。

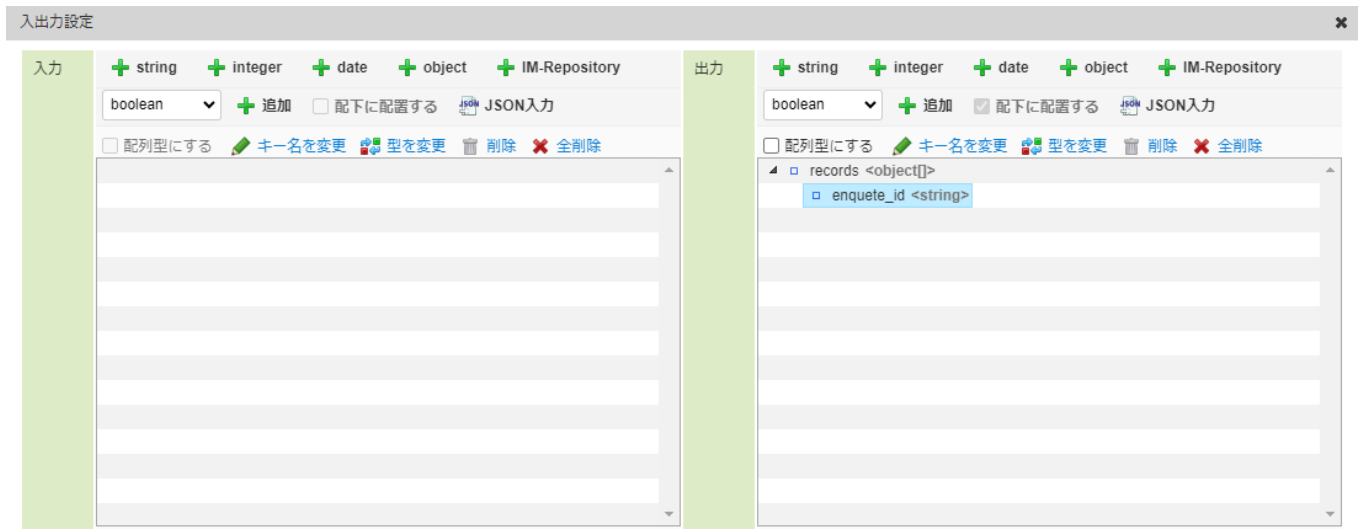


決定 取り消し

次に、アンケート情報を格納していくための出力値を「records」配下に追加します。

出力値一覧の「records」を選択し、「配下に配置する」選択します。

「出力」 - 「+string」をクリックし「records」配下に、「string1」という値を追加して、キー名を「enquete_id」と変更します。

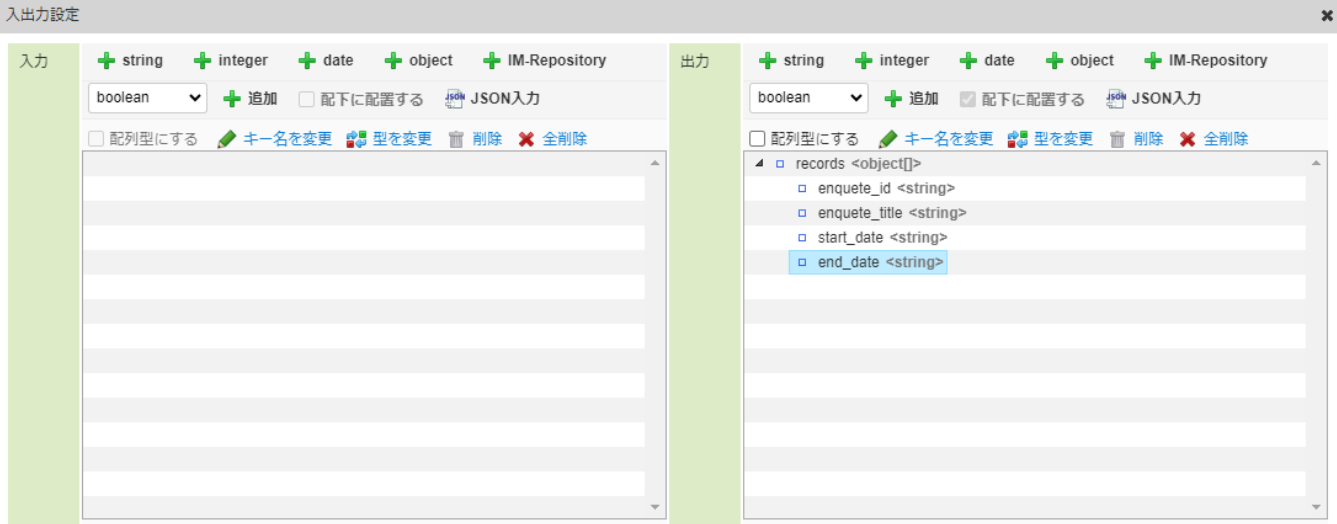


決定 取り消し

同様の手順で「records」の配下の変数が以下の値になるように設定してください。

開始、終了日は日付型でデータベースに格納されていますが、ユーザが表示しやすい形式に変換して返却するため、文字列で返却します。

項目	型	キー名
アンケートID	string	enquete_id
アンケートタイトル	string	enquete_title
開始日	string	start_date
終了日	string	end_date



決定 取り消し

出力値はJSON入力で以下のコードを記載しても設定できます。

```
"records": [
  {
    "enquete_id": "",
    "enquete_title": "",
    "start_date": "",
    "end_date": ""
  }
]
```

JSON入力する場合は、出力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

アンケート一覧取得タスクの作成

アンケート情報テーブルからデータを取得するユーザ定義タスクを作成します。
パレットから「ユーザ定義追加」にカーソルを合わせ、「SQL定義新規作成」を選択します。



ユーザ定義共通設定を以下のように設定します。

ラベル	入力値
ユーザ定義ID	tutorial_enquete_list
ユーザ定義名 (標準)	【チュートリアル】アンケート一覧
ユーザカテゴリ	【チュートリアル】チュートリアルガイドカテゴリ

クエリに以下のSQLを設定します。

SELECT

```
enquete_id,
enquete_title,
start_date,
end_date
```

FROM

```
tutorial_enquete_info_app
```

「データ定義を取得する」をクリックし、タスクの出力値を設定します。

The screenshot shows the configuration interface for a task. It is divided into two main sections: 'JSON出力値の設定' (JSON Output Value Settings) and 'SQL定義' (SQL Definition).

JSON出力値の設定: This section has two panels: '入力値' (Input Values) and '返却値' (Return Values). Both panels have a toolbar with options for adding, deleting, and changing types. The '返却値' panel is currently selected, showing a JSON structure:

```
{
  "query": <string>,
  "count": <integer>,
  "records": <object[]> [
    {
      "enquete_id": <string>,
      "enquete_title": <string>,
      "start_date": <sqltimestamp>,
      "end_date": <sqltimestamp>
    }
  ]
}
```

SQL定義: This section contains configuration fields and a query editor.

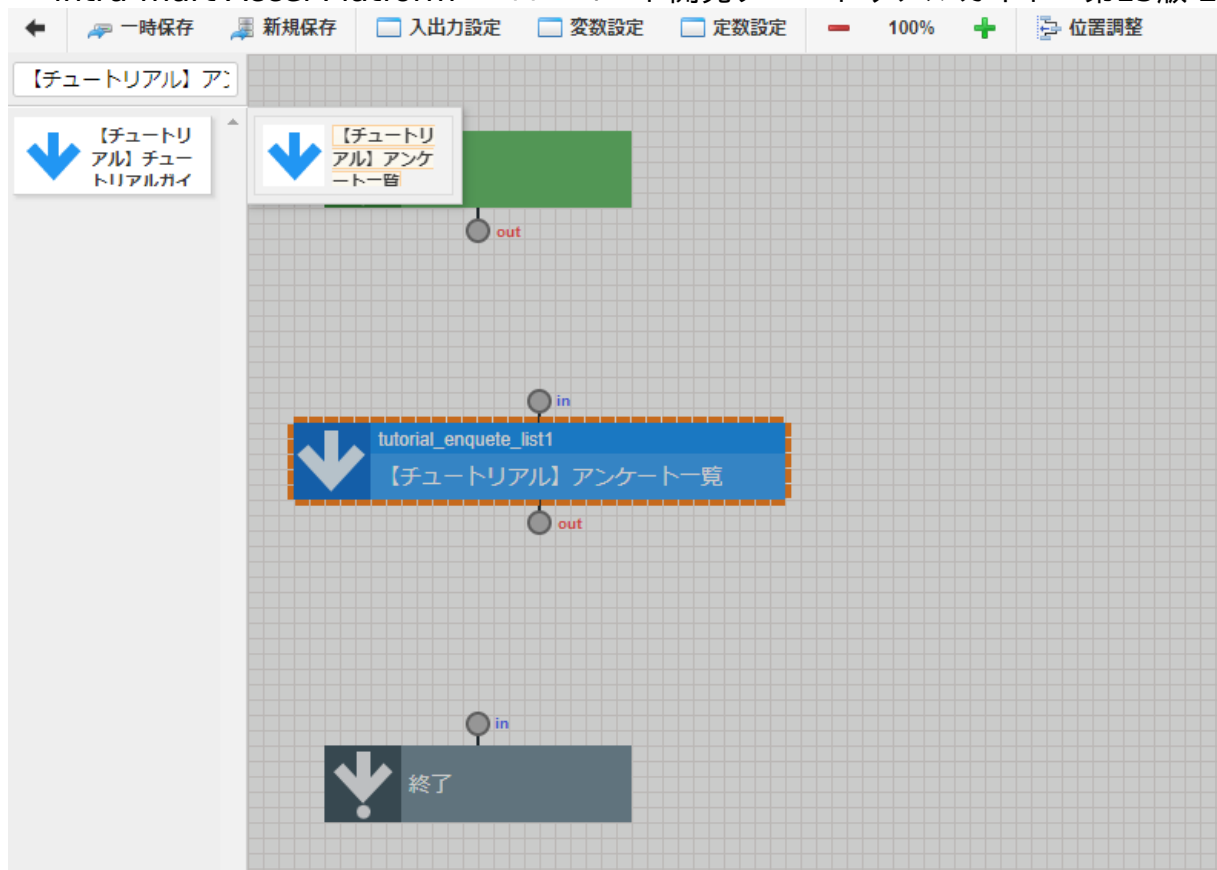
- データベース種別:** TENANT
- クエリ種別:** SELECT
- 範囲を指定する:** (unchecked)
- クエリ:**

```
1 データ定義を取得する
2 SELECT
3   enquete_id,
4   enquete_title,
5   start_date,
6   end_date
7 FROM
8   tutorial_enquete_info_app
```

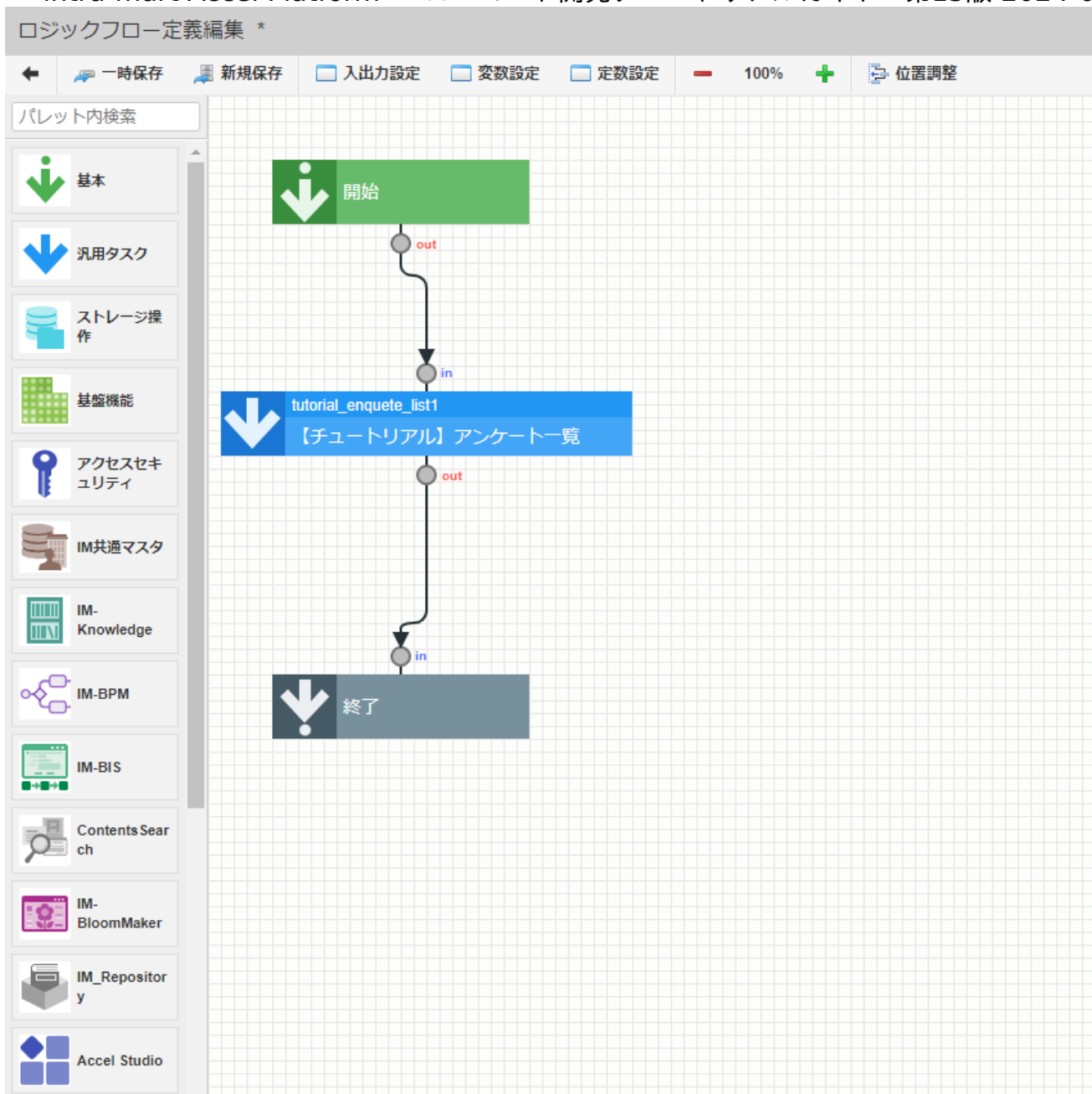
登録ボタンをクリックし、タスクを登録します。

タスクの配置と変数設定

作成した「【チュートリアル】アンケート一覧」タスクをフローに配置し、線を繋げます。
 パレットから「【チュートリアル】チュートリアルガイドカテゴリ」 - 「【チュートリアル】アンケート一覧」を選択し、「開始」タスクの下に配置します。

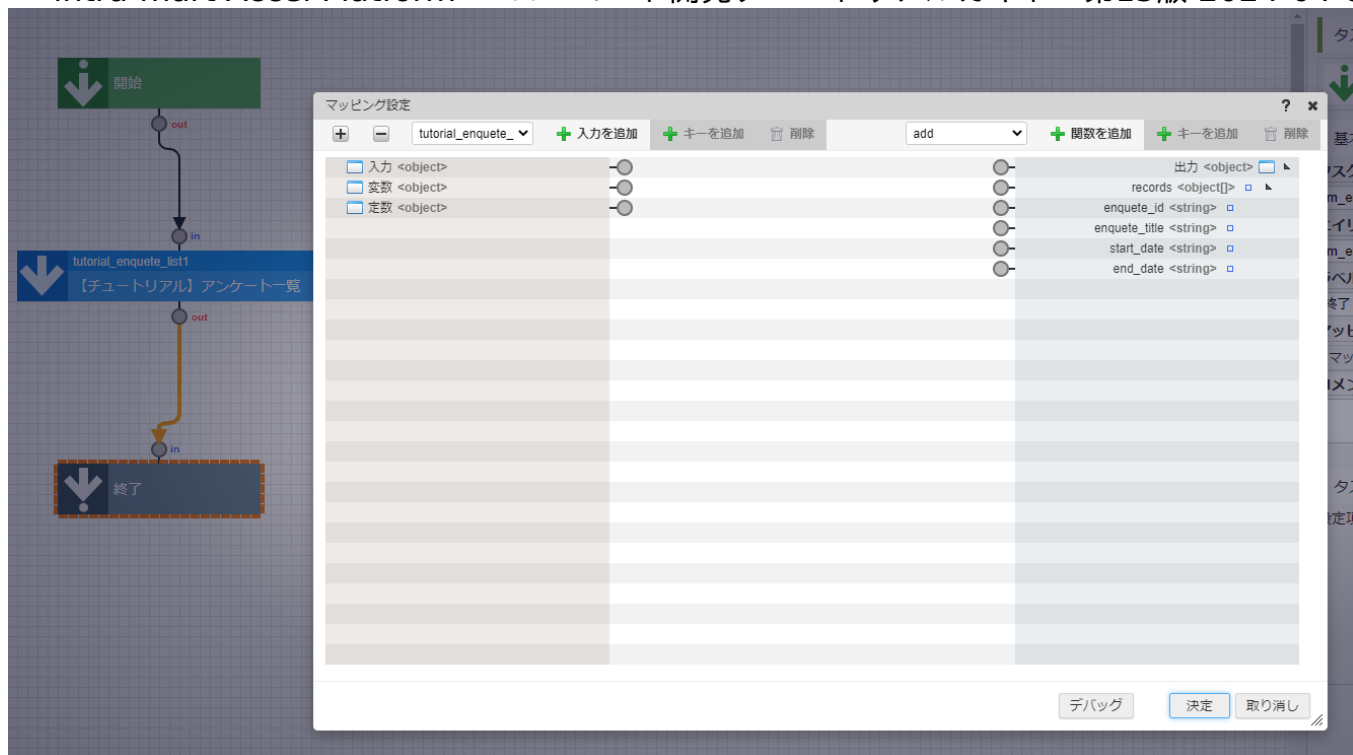


「開始」、「終了」タスクと「【チュートリアル】アンケート一覧」を線で繋がります。



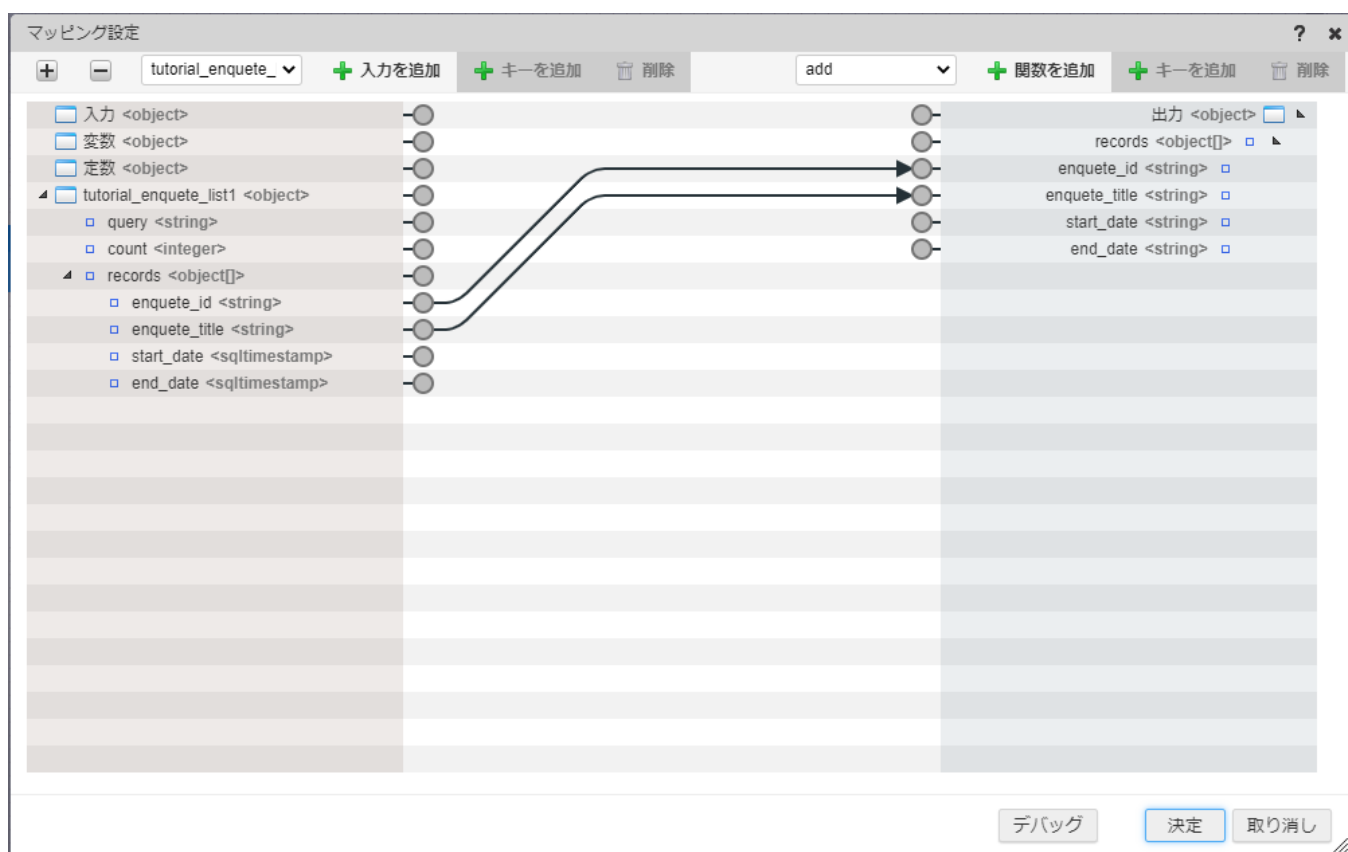
次に、配置したタスクの変数を設定していきます。

「終了」タスクを選択し、「マッピング設定」ダイアログを表示します。



「tutorial_enquete_list1」を選択し、「+入力を追加」をクリックします。

ダイアログ左側の「tutorial_enquete_list1」の「enqueteld」と右側の「出力」の「enquete_id」、「enquete_title」と右側の「出力」の「enquete_title」を線で繋げます。



次に、「開始日」を設定します。開始日は日付型で格納されており、そのままでは直感的に参照しやすい形式ではないため、「yyyy/MM/dd」の文字列に変換して返却します。

まずはフォーマットを定数で作成します。一度、決定ボタンをクリックしてマッピング設定を保存してください。

画面上部の「定数設定」をクリックして、定数設定ダイアログを表示します。

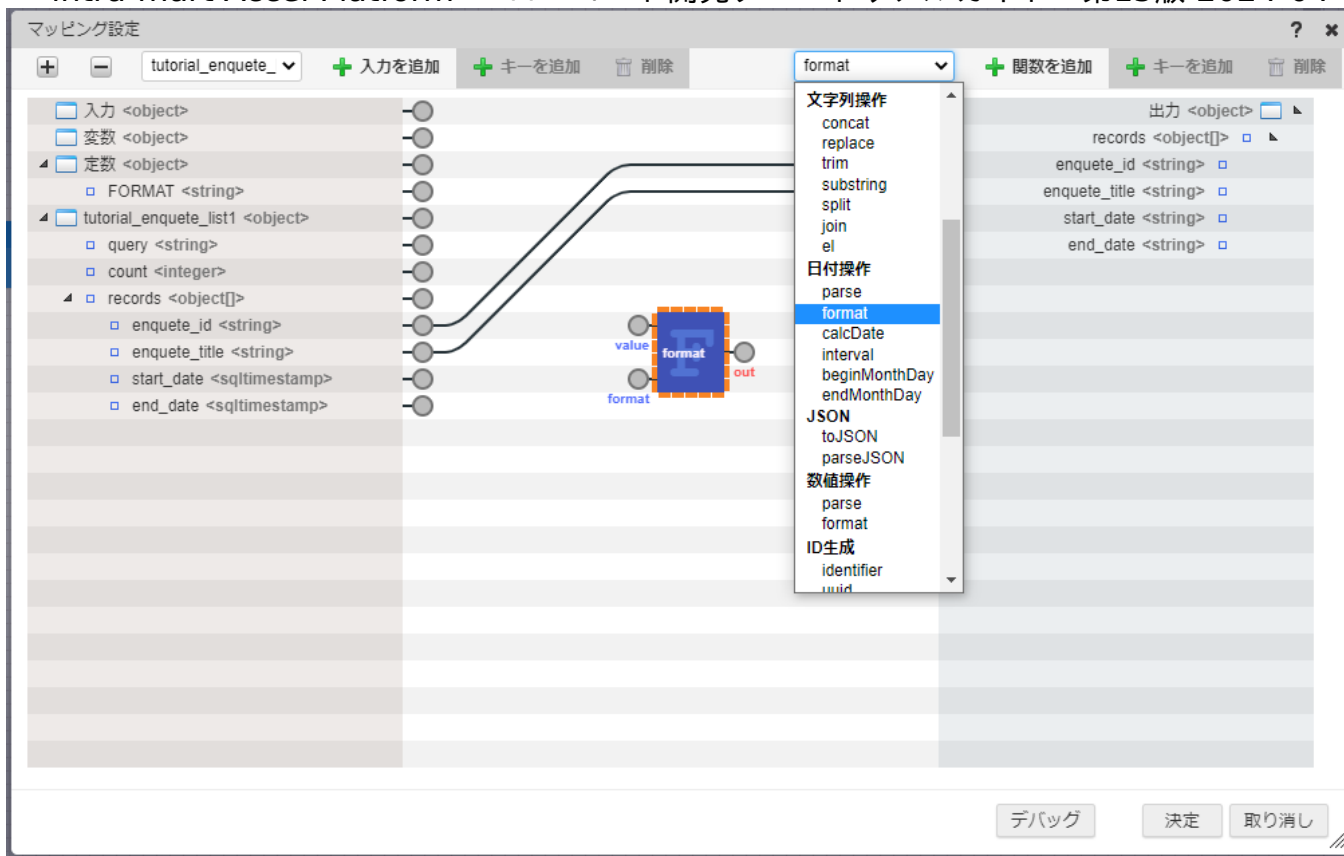


「+定数を追加」をクリックし、以下の定数を追加し、決定ボタンをクリックして定数を保存します。

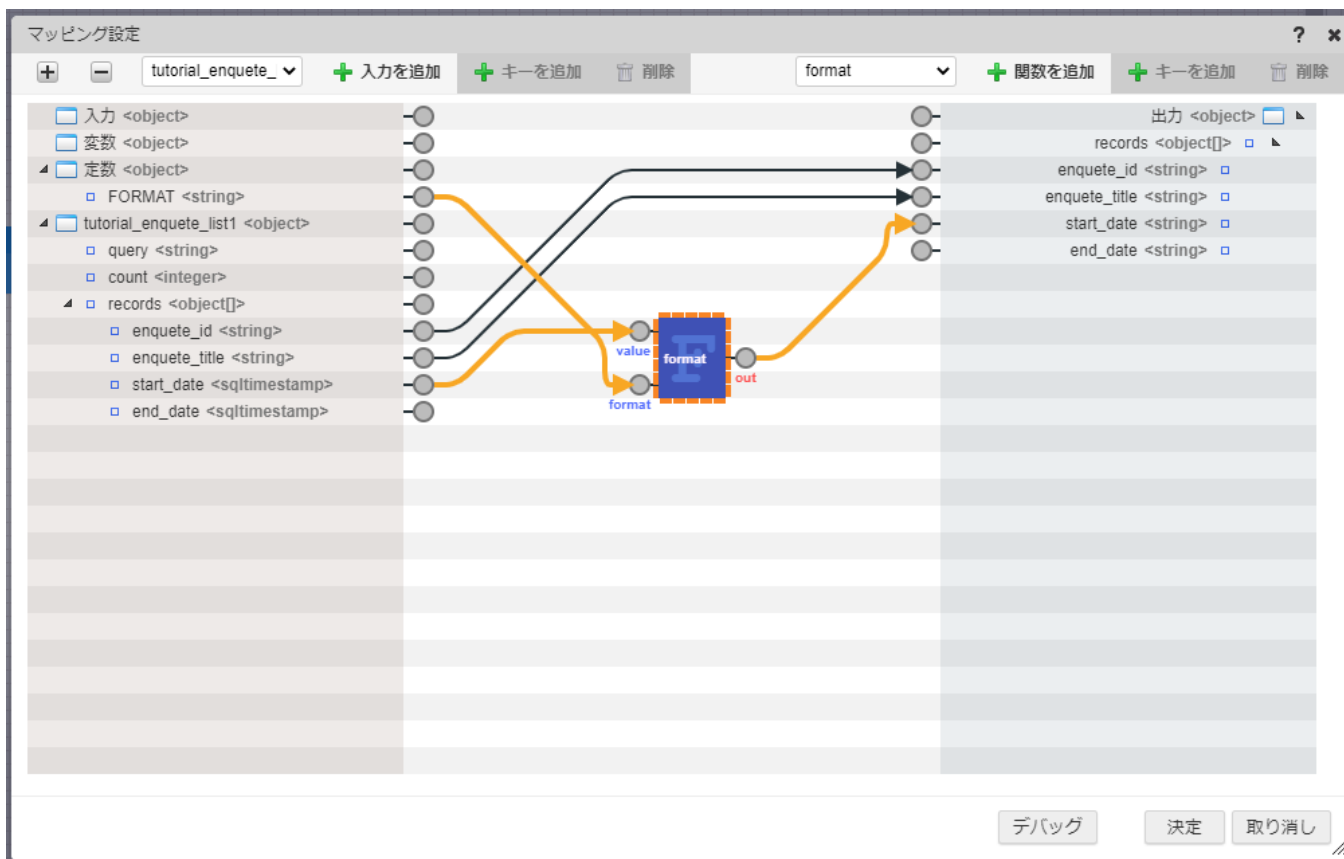
定数ID	定数値
FORMAT	yyyy/MM/dd



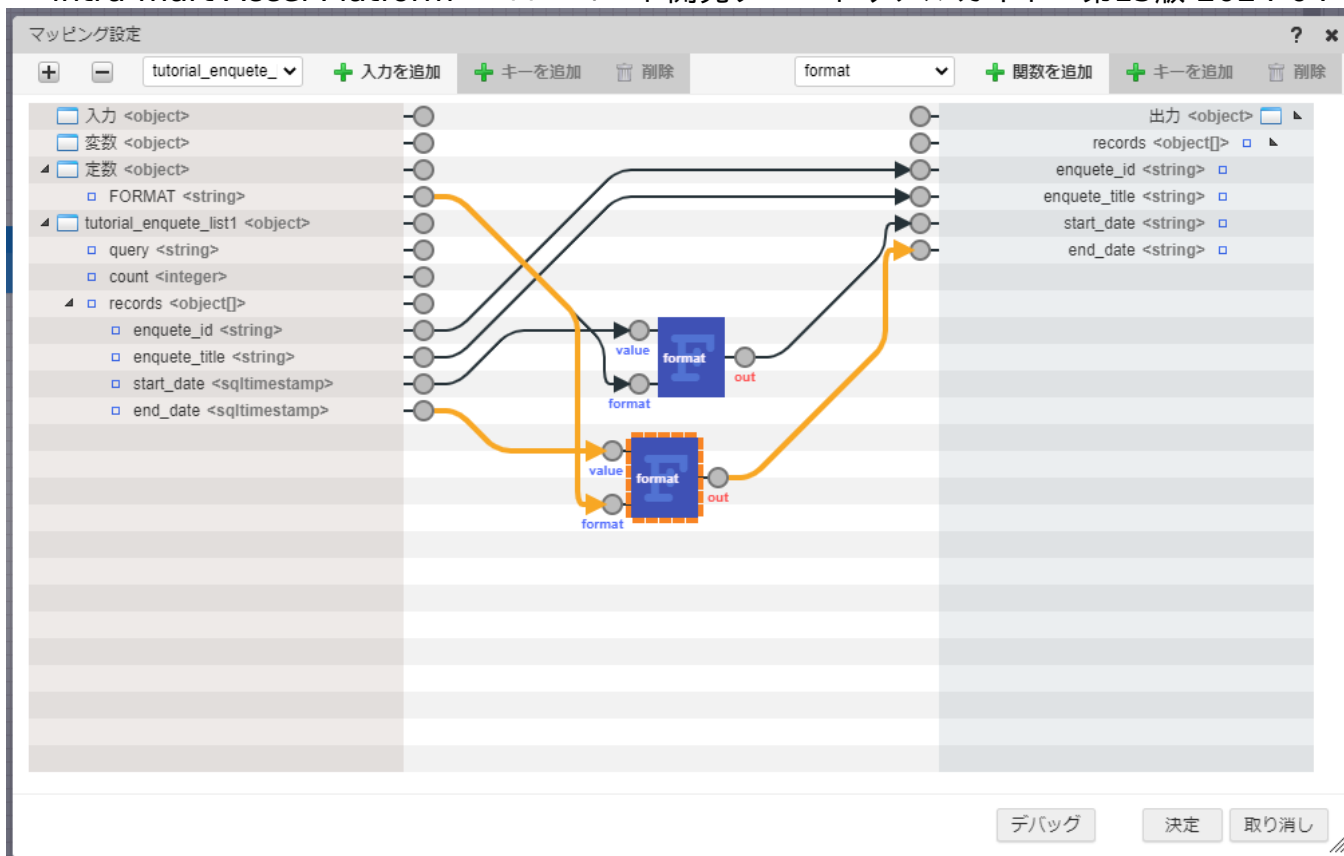
定数を保存したら再度「終了」タスクのマッピング設定を表示し、関数から「日付操作 > format」を選択し、「+関数を追加」で配置します。



ダイアログ左側の「tutorial_enquete_list1」の「start_date」を「format」の「value」に、定数の「FORMAT」を「format」の「format」に、「format」の「out」を出力の「start_date」にそれぞれ線を繋げます。



同様の手順で「end_date」のフォーマット設定を行い、決定ボタンをクリックしてマッピングを保存します。



これで、アンケート一覧取得処理の作成は終了です。

画面上部の「新規保存」をクリックし、以下のように入力し、「決定」ボタンをクリックして、ロジックの登録を行います。

ラベル	入力値
フロー定義ID	tutorial_enquete_list
ユーザ定義名 (標準)	【チュートリアル】アンケート一覧取得
ユーザカテゴリ	【チュートリアル】チュートリアルガイドカテゴリ

新規保存

フロー定義ID *

フロー定義名 *

標準 *	<input type="text" value="【チュートリアル】アンケート一覧取得"/>
英語	<input type="text"/>
日本語	<input type="text"/>
中国語 (中華人民共和国)	<input type="text"/>

フローカテゴリ *

検索/新規作成

ID *	<input type="text" value="tutorial_category"/>
名称	<input type="text" value="【チュートリアル】チュートリアルカテゴリ"/>

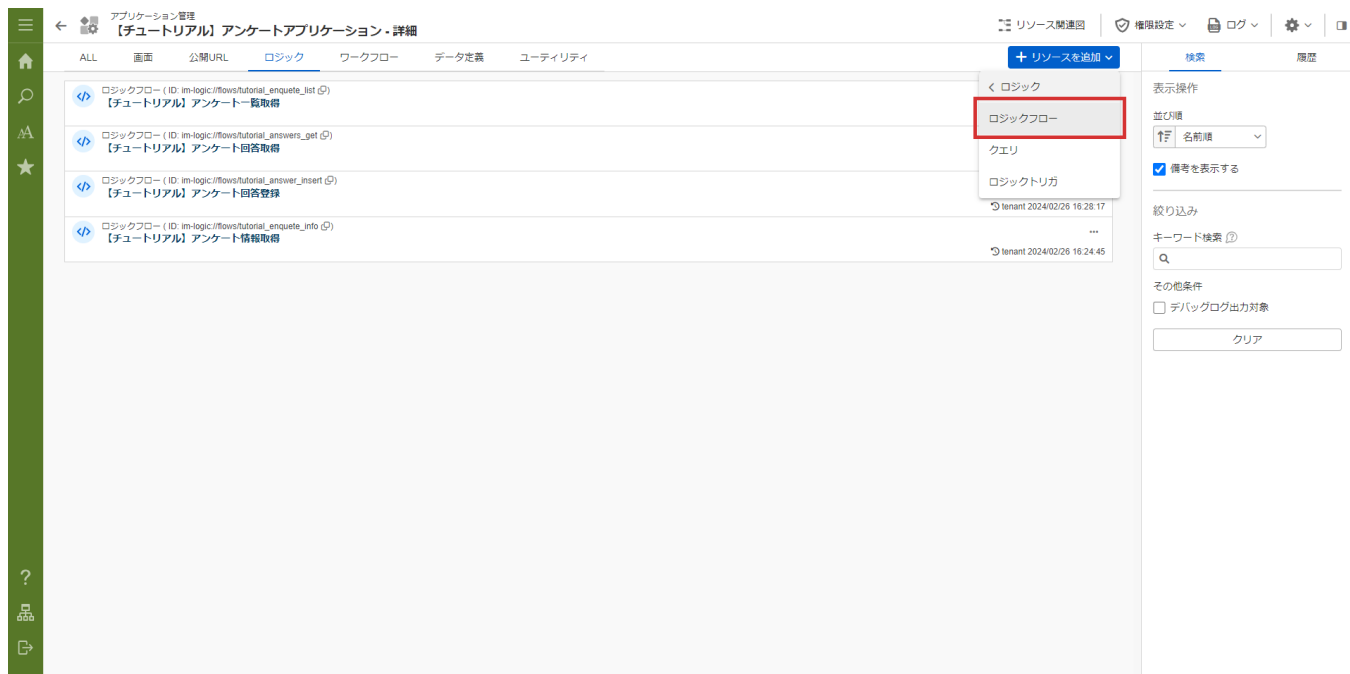
備考

次に、アンケートを登録する処理を作成します。

アンケート登録処理では、渡されたアンケート情報をアンケート情報テーブルとアンケート設問テーブルに格納します。

ロジックの作成

アンケート情報取得と同様に、「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「ロジック」 - 「ロジックを新規作成」 - 「ロジックフロー」を選択し、「ロジックフロー定義編集画面」を表示します。



入出力値の設定

業務ロジックへの入力値を設定します。

アンケート登録処理は、アンケート情報とアンケートの設問を配列で受け取ります。出力値は設定しません。

入力値の設定を行います。

画面上部の「入出力設定」をクリックし「入出力設定」ダイアログを表示します。

「入力」 - 「+string」をクリックし、キー名を「enqueteld」と設定してください。

同様の手順で、以下の表に則ってアンケート情報を格納する入力値を設定します。

項目	型	キー名
アンケートID	string	enqueteld
アンケートタイトル	string	enqueteTitle
アンケート概要	string	enqueteOverview
開始日	date	startDate
終了日	date	endDate

入出力設定

決定 取り消し

次に、アンケートの設問情報を格納する項目を追加します。「入力」 - 「+object」をクリックし、キー名を「questionList」と設定してください。

追加した「questionList」を選択し、「配列型にする」を選択します。

「questionList」を選択し、「配下に配置する」にチェックを入れ、「questionList」配下に「question」、「leftScale」、「rightScale」をすべて文字列で追加します。

入出力設定

決定 取り消し

入力値はJSON入力で以下のコードを記載しても設定できます。

```
{
  "enqueteId": "",
  "enqueteTitle": "",
  "enqueteOverview": "",
  "startDate": null,
  "endDate": null,
  "questionList": [
    {
      "question": "",
      "leftScale": "",
      "rightScale": ""
    }
  ]
}
```

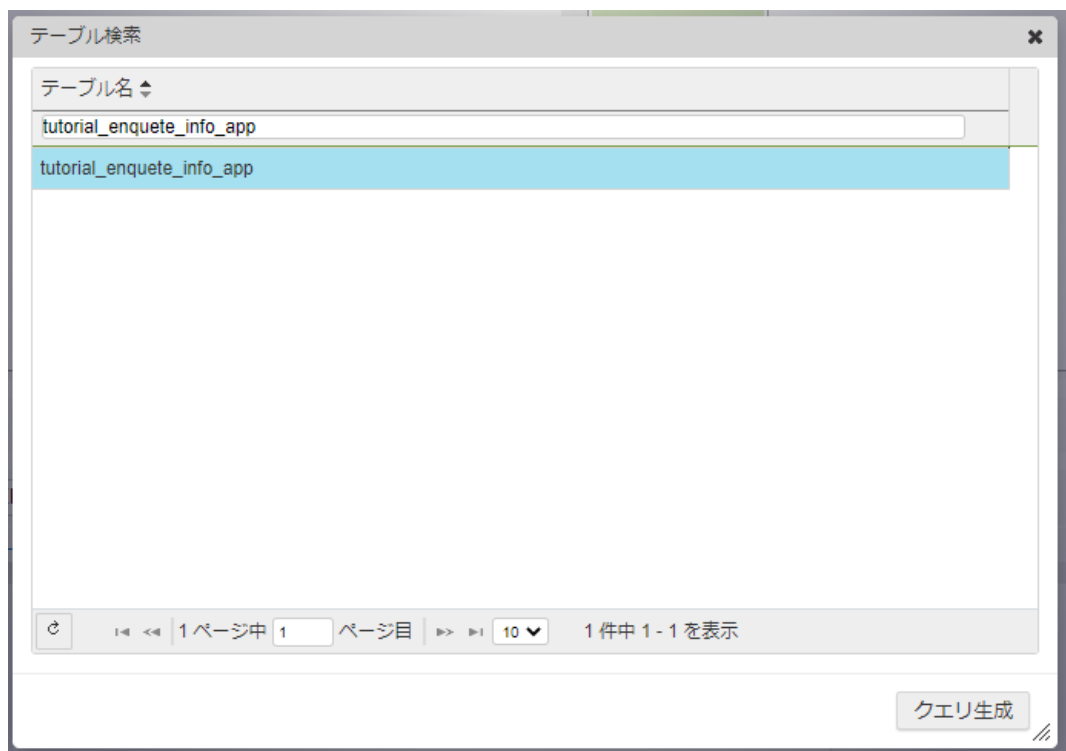
JSON入力する場合は、出力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型

アンケート情報登録タスクの作成

入力内容に基づいてアンケート回答テーブルにデータを登録するユーザ定義タスクを作成します。
パレットから「ユーザ定義追加」にカーソルを合わせ、「SQL定義新規作成」を選択します。
ユーザ定義共通設定を以下のように設定します。

ラベル	入力値
ユーザ定義ID	tutorial_enquete_info_insert
ユーザ定義名 (標準)	【チュートリアル】アンケート情報登録
ユーザカテゴリ	【チュートリアル】チュートリアルカテゴリ

SQL定義のクエリ種別のプルダウンから「INSERT」を選択します。
「テーブルからクエリを生成する」をクリックし、テーブル検索ダイアログを表示します。
テーブル検索ダイアログで「tutorial_enquete_info_app」を選択し「クエリ生成」をクリックします。



SQL定義編集画面の入力値とクエリにそれぞれ以下のような値が自動で設定されます。

- 入力値

項目	型	キー名
アンケートID	string	enqueteId
アンケートタイトル	string	enqueteTitle
アンケート概要	string	enqueteOverview
開始日	sqltimestamp	startDate
終了日	sqltimestamp	endDate

- クエリ

```
INSERT INTO tutorial_enquete_info_app
(enquete_id, enquete_title, enquete_overview, start_date, end_date)
VALUES
(/*enqueteId*/null, /*enqueteTitle*/null, /*enqueteOverview*/null, /*startDate*/null, /*endDate*/null)
```

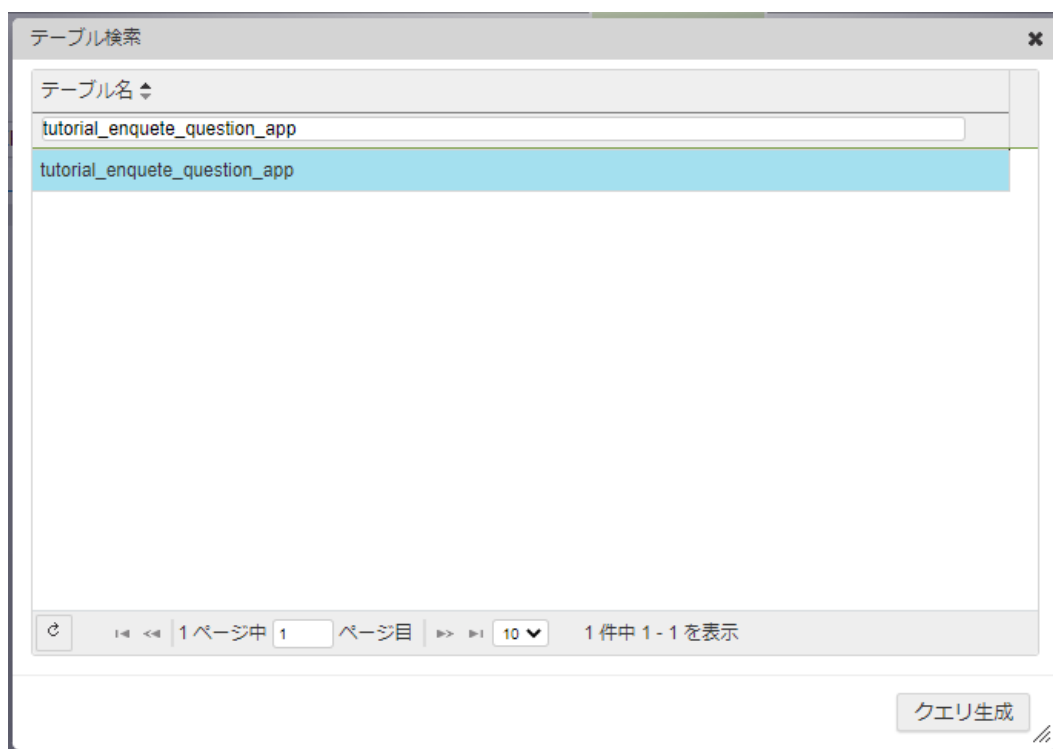
「登録」ボタンをクリックし、タスクを登録します。

アンケート設問登録タスクの作成

続けて、入力内容に基づいてアンケート設問テーブルにデータを登録するユーザ定義タスクを作成します。パレットから「ユーザ定義追加」にカーソルを合わせ、「SQL定義新規作成」を選択します。ユーザ定義共通設定を以下のように設定します。

ラベル	入力値
ユーザ定義ID	tutorial_enquete_question_insert
ユーザ定義名 (標準)	【チュートリアル】 アンケート設問登録
ユーザカテゴリ	【チュートリアル】 チュートリアルカテゴリ

SQL定義のクエリ種別のプルダウンから「INSERT」を選択します。「テーブルからクエリを生成する」をクリックし、テーブル検索ダイアログを表示します。テーブル検索ダイアログで「tutorial_enquete_question_app」を選択し「クエリ生成」をクリックします。



SQL定義編集画面の入力値とクエリにそれぞれ以下のような値が自動で設定されます。

- 入力値

項目	型	キー名
アンケートID	string	enqueteId
設問ID	string	questionId
設問番号	bigdecimal	questionNo
設問内容	string	question
左項目	string	leftScale
右項目	string	rightScale

- クエリ

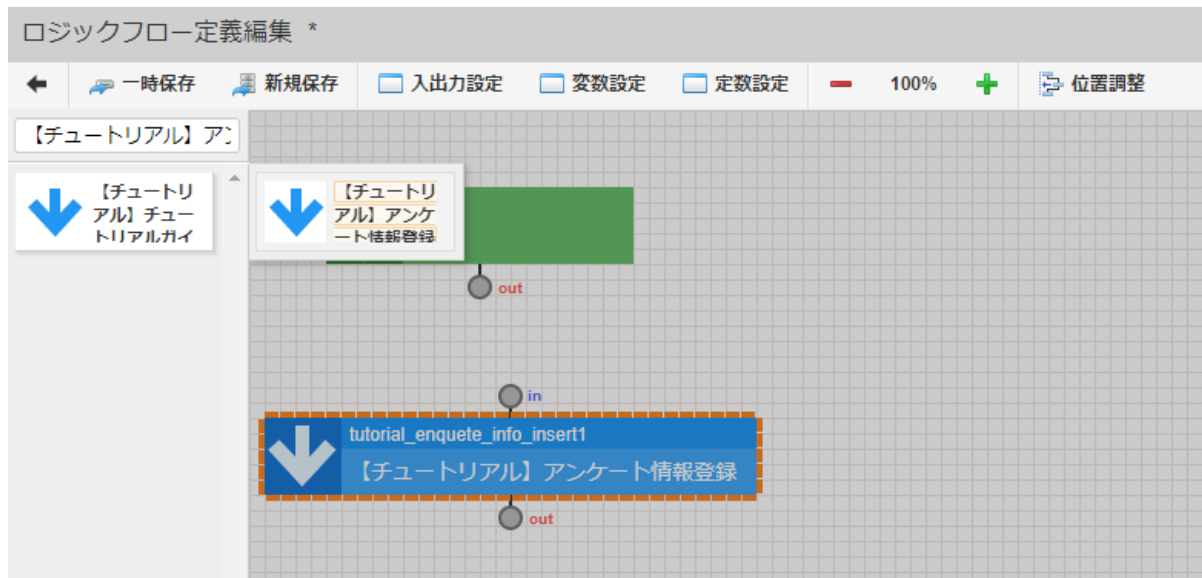

```
INSERT INTO tutorial_enquete_question_app
(enquete_id, question_id, question_no, question, left_scale, right_scale)
VALUES
(*enqueteId*/null, /*questionId*/null, /*questionNo*/null, /*question*/null, /*leftScale*/null, /*rightScale*/null)
```

「登録」ボタンをクリックし、タスクを登録します。

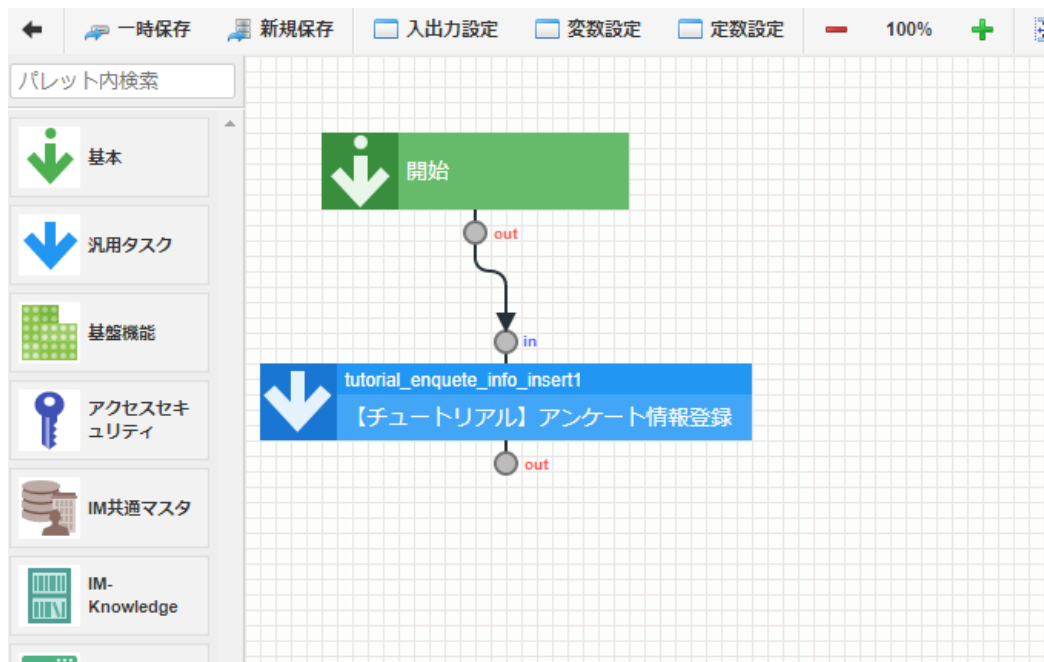
タスクの配置と変数設定

作成した「【チュートリアル】アンケート情報登録」タスクと「【チュートリアル】アンケート設問登録」タスクをフローに配置し、線を繋げます。

パレットから「【チュートリアル】チュートリアルガイドカテゴリ」 - 「【チュートリアル】アンケート情報登録」を選択し、開始の下に配置します。



配置した「【チュートリアル】アンケート情報登録」の「in」と「開始」タスクの「out」を線で繋がめます。

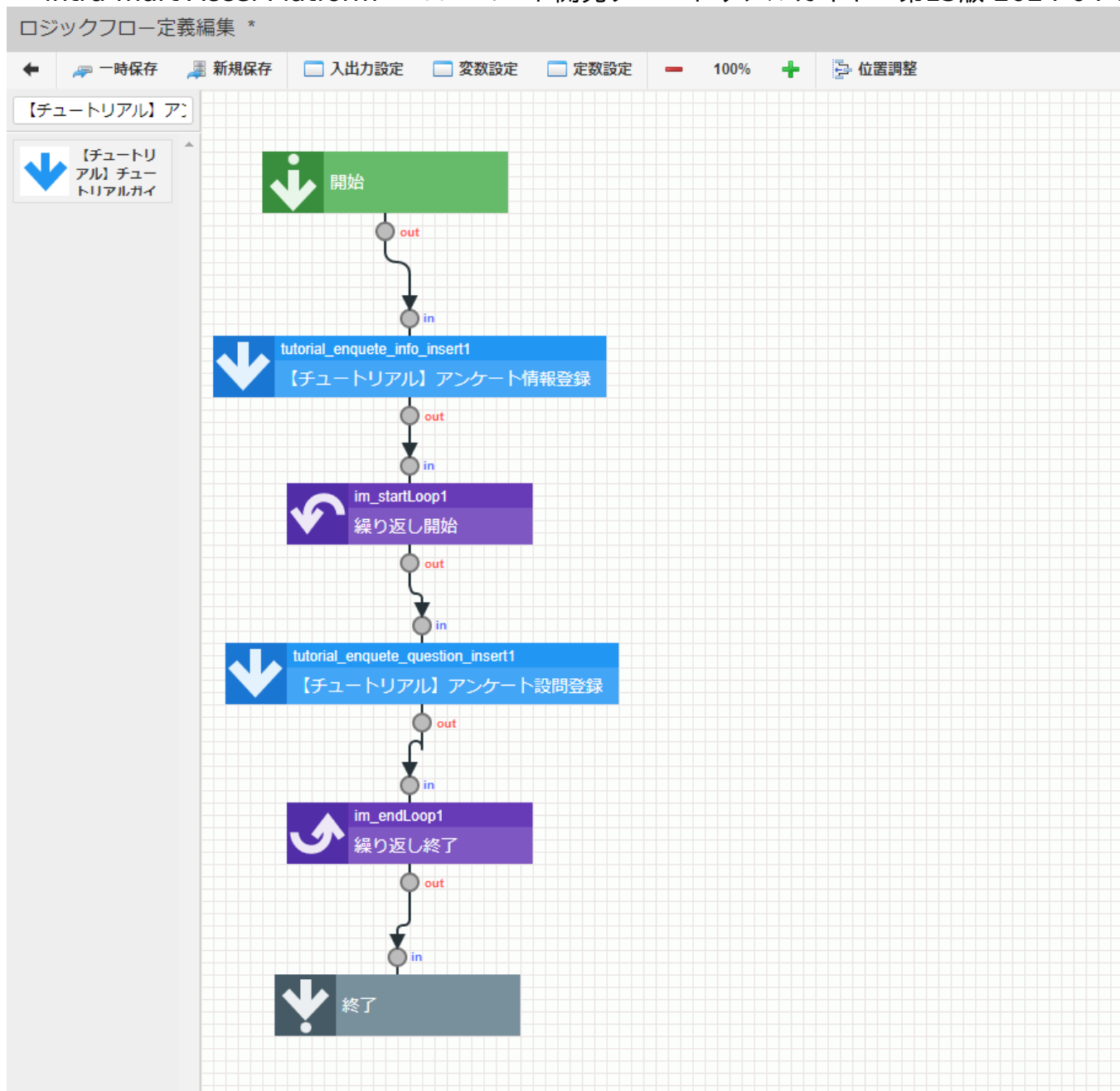


次に、アンケートの設問登録処理を配置します。アンケートの設問は入力値として渡された設問数だけ繰り返して実行します。

パレットから「基本」 - 「繰り返し開始」を選択し、「【チュートリアル】アンケート情報登録」の下に配置します。「繰り返し」タスクは「繰り返し開始」と「繰り返し終了」タスクがセットで配置されます。



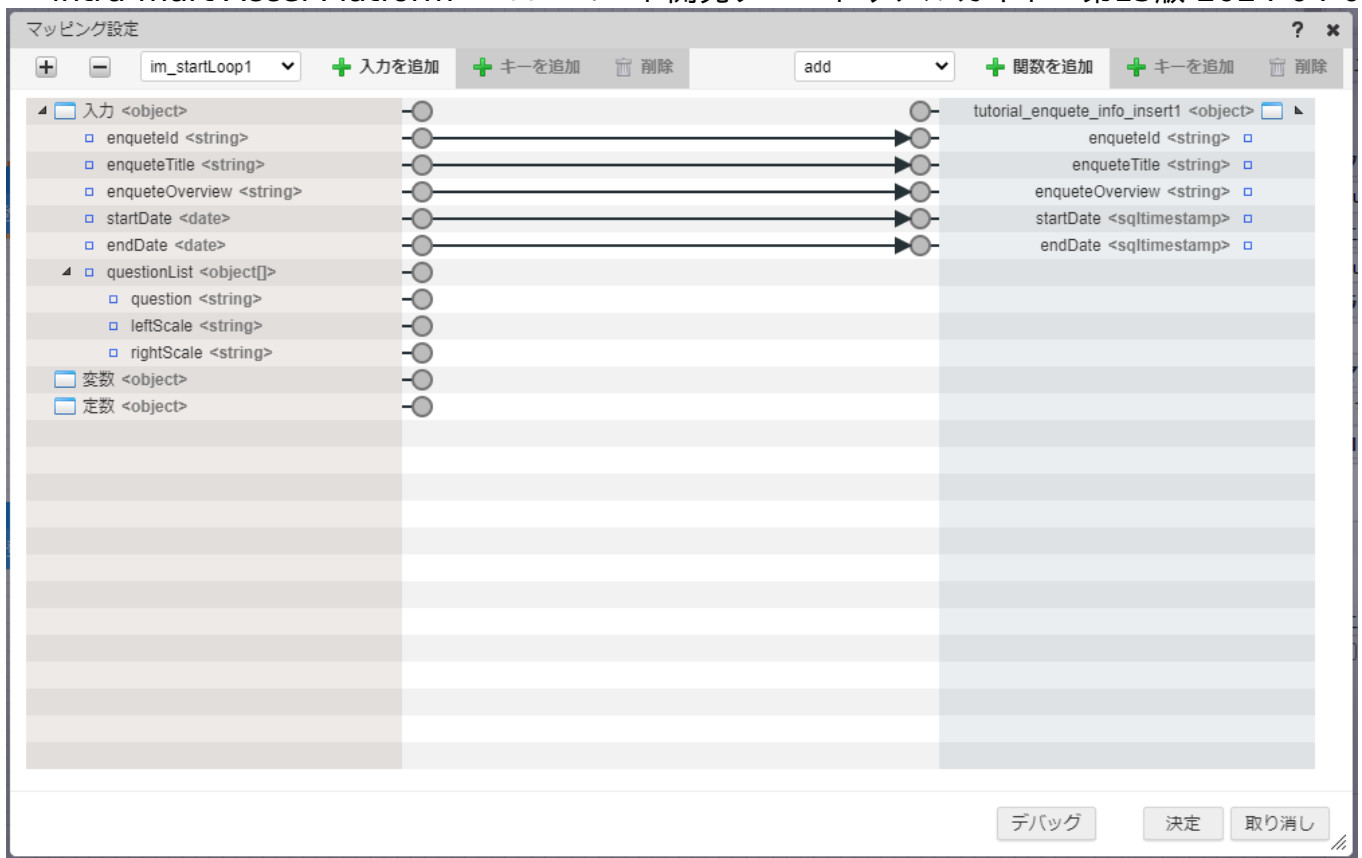
続けて、「【チュートリアル】アンケート設問登録」タスクを「繰り返し開始」と「繰り返し終了」タスクの間に配置し、全てのタスクを線で繋がめます。



次に、それぞれのタスクのマッピング設定を行います。

「【チュートリアル】アンケート情報登録」タスクを選択し、マッピング設定を表示します。

マッピング設定上で、左側の「入力」と右側の「tutorial_enquete_info_insert」の同じ値を線で繋がめます。

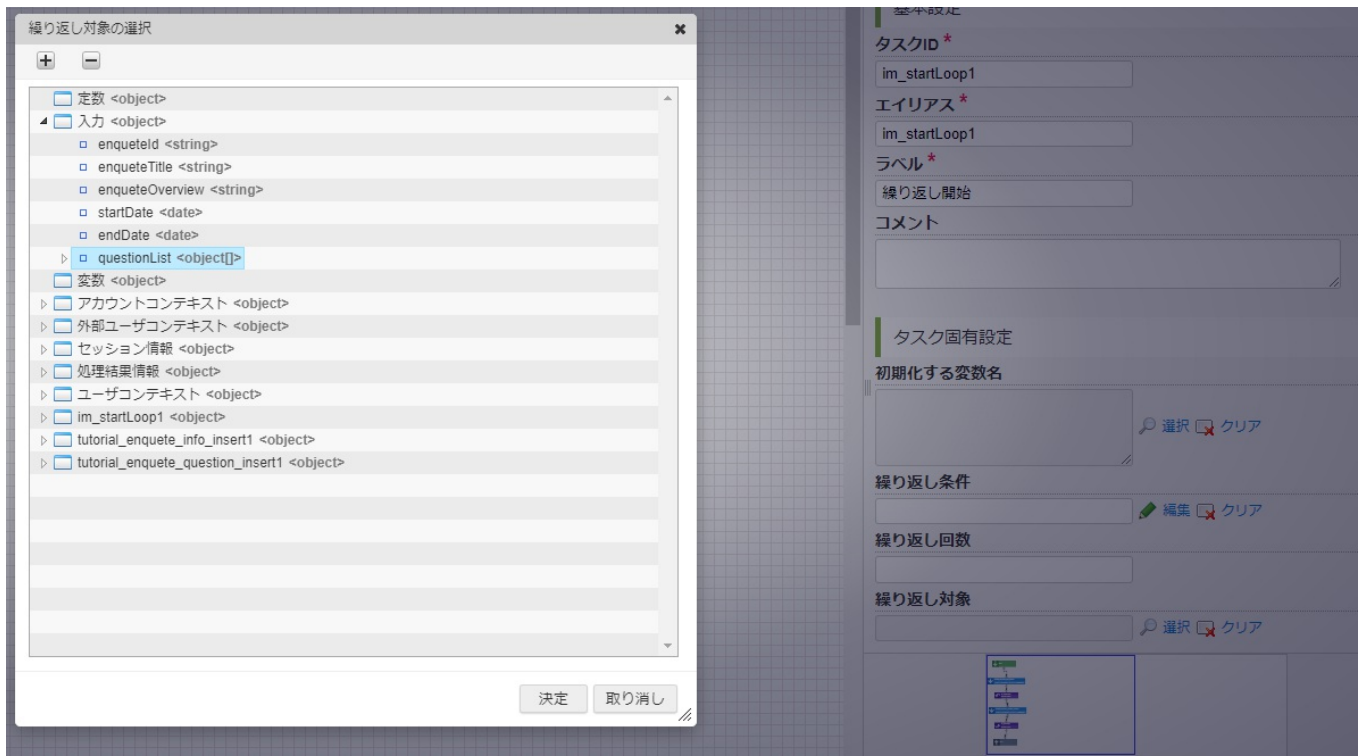


「決定」ボタンでマッピング設定ダイアログを閉じます。

「繰り返し開始」タスクを選択し、「タスク固有設定」の「繰り返し対象」 - 「選択」から「繰り返し対象の選択」ダイアログを表示します。

「入力 > questionList」を選択し決定ボタンをクリックします。

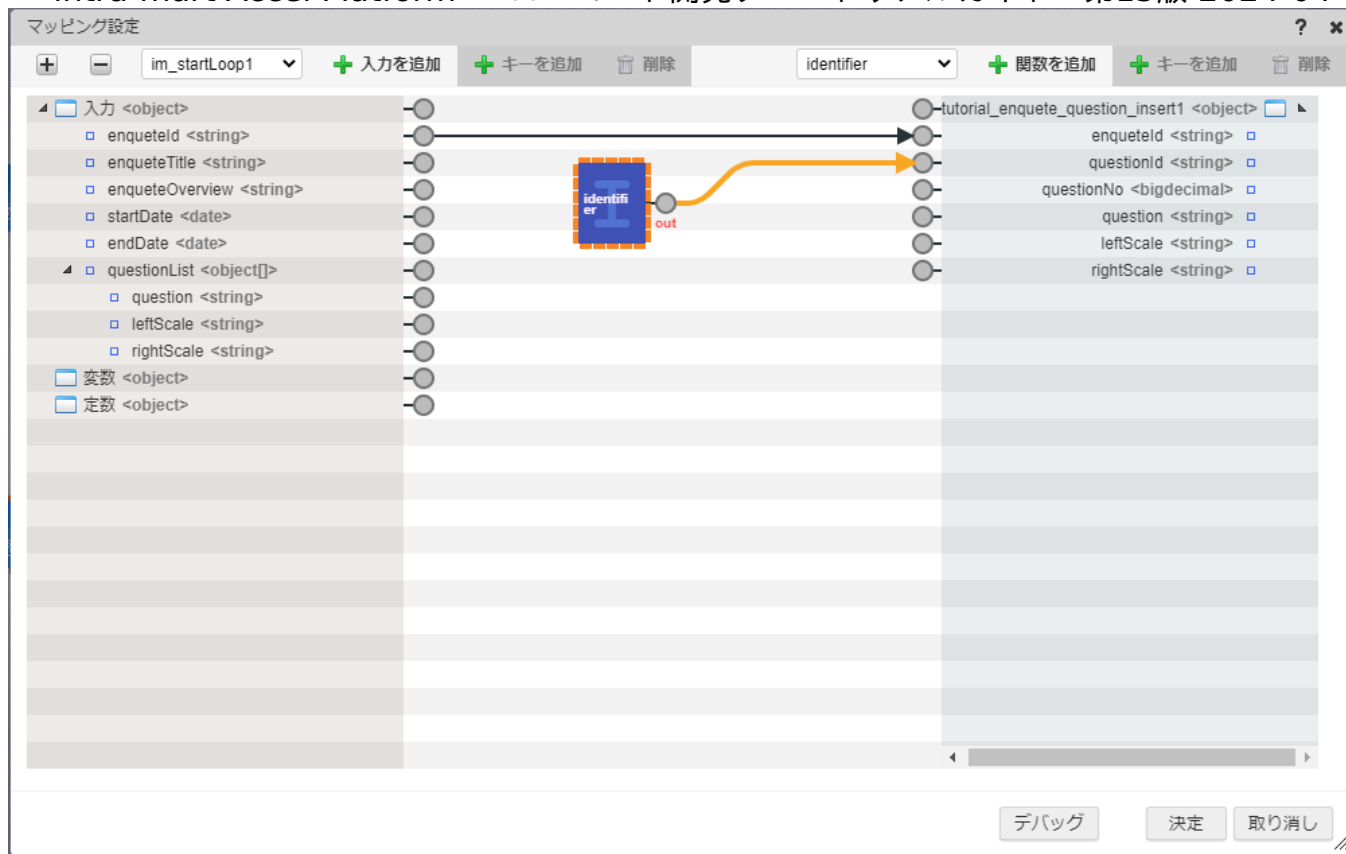
これで、入力値として渡されたquestionListの配列の長さだけ繰り返し「【チュートリアル】アンケート設問登録」が実行されます。



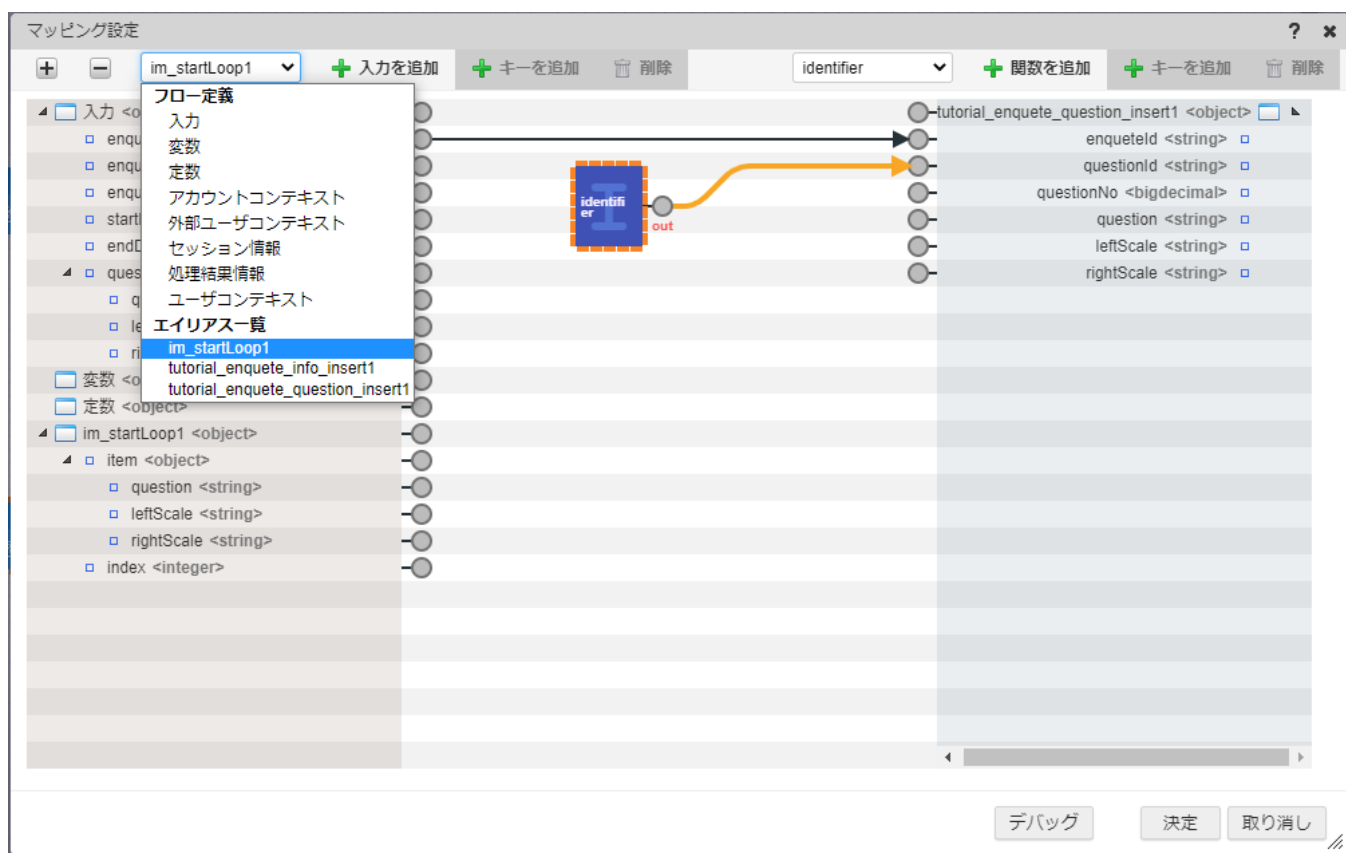
続けて、「【チュートリアル】アンケート設問登録」を選択しマッピング設定を開きます。

「入力」の「enqueteld」と「tutorial_enquete_question_insert」の「enqueteld」を線で繋がめます。

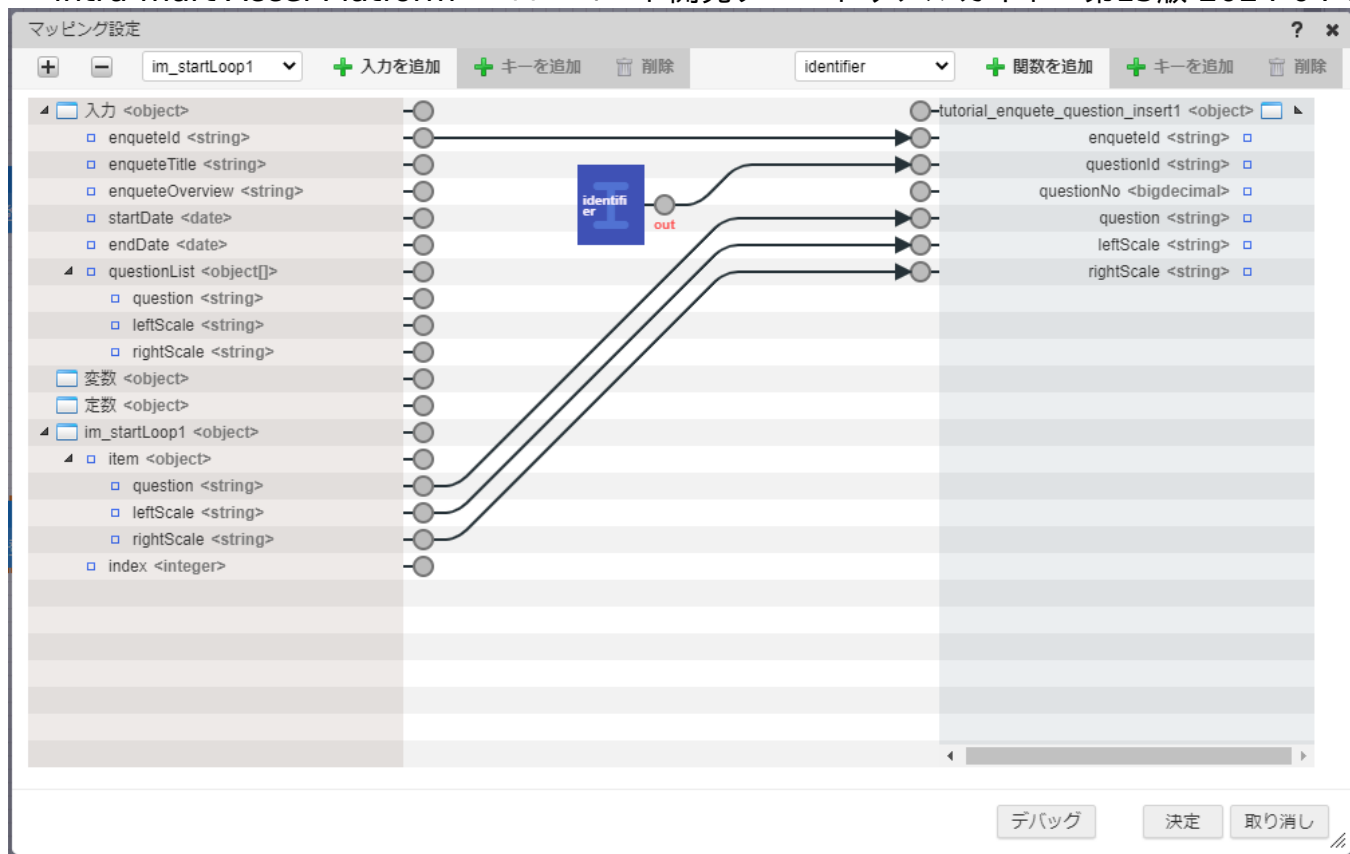
設問IDは自動設定とするため、関数から「identifier」を配置し、「tutorial_enquete_question_insert」の「questionId」を線で繋がめます。



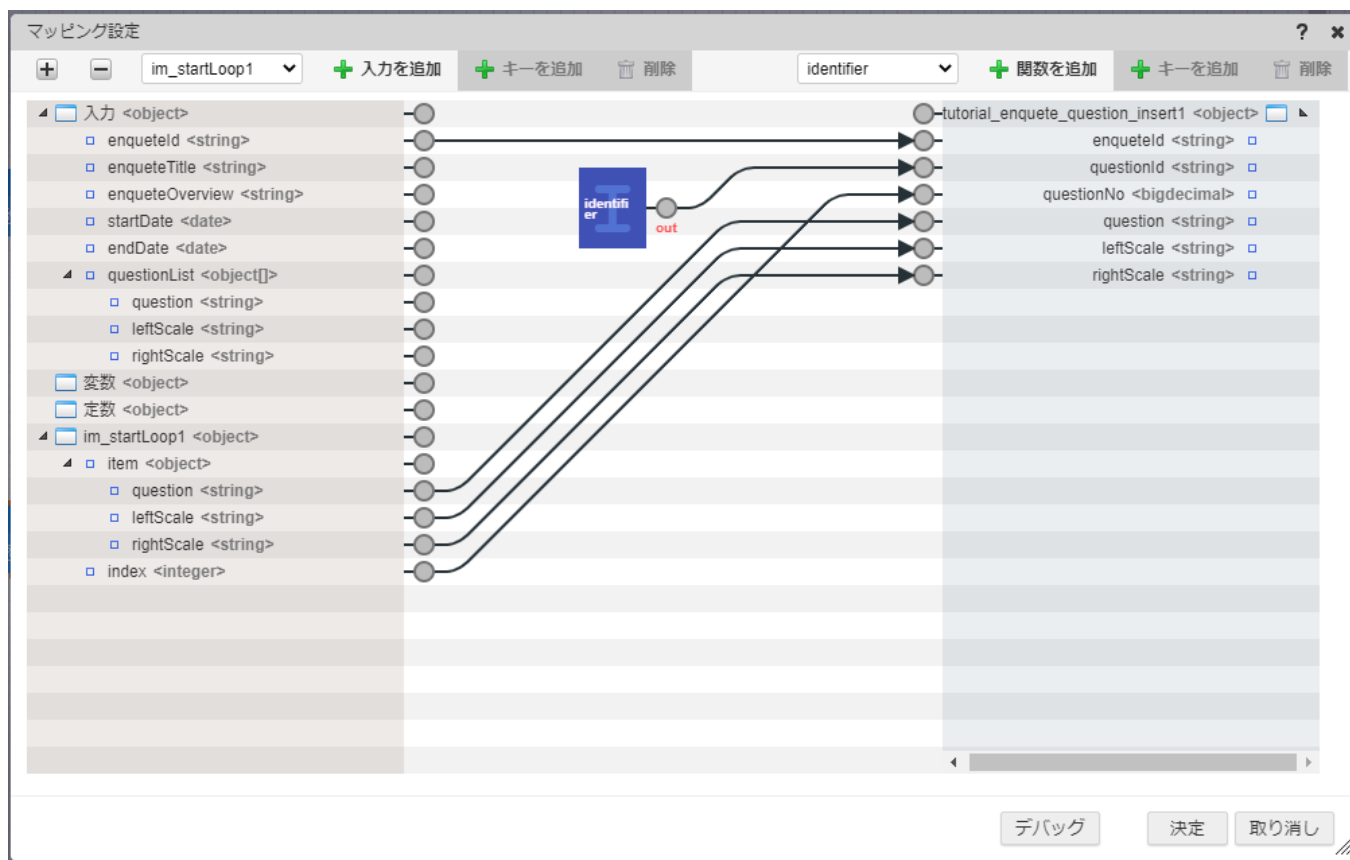
エイリアス一覧から「im_startLoop1」を選択し、「+入力を追加」を選択します。



追加された「im_startLoop」の「item」配下の値と「tutorial_enquete_question_insert1」配下の同様の変数を線で繋がめます。



最後に、「im_startLoop」の「index」と「tutorial_enquete_question_insert」の「questionNo」を線で繋ぎ、決定ボタンをクリックしてマッピング設定を保存します。



これで、アンケート設問登録処理の作成は終了です。
画面上部の「新規保存」をクリックし、以下のように入力し、「決定」ボタンをクリックして、ロジックの登録を行います。

ラベル	入力値
フロー定義ID	tutorial_enquete_insert
ユーザ定義名（標準）	【チュートリアル】 アンケート設問登録
ユーザカテゴリ	【チュートリアル】 チュートリアルガイドカテゴリ

新規保存 ✕

フロー定義ID *	<input type="text" value="tutorial_enquete_insert"/>								
フロー定義名 *	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; padding: 2px;">標準 *</td> <td style="padding: 2px;"><input type="text" value="【チュートリアル】 アンケート設問登録"/></td> </tr> <tr> <td style="padding: 2px;">英語</td> <td style="padding: 2px;"><input type="text"/></td> </tr> <tr> <td style="padding: 2px;">日本語</td> <td style="padding: 2px;"><input type="text"/></td> </tr> <tr> <td style="padding: 2px;">中国語 (中華人民共和国)</td> <td style="padding: 2px;"><input type="text"/></td> </tr> </table>	標準 *	<input type="text" value="【チュートリアル】 アンケート設問登録"/>	英語	<input type="text"/>	日本語	<input type="text"/>	中国語 (中華人民共和国)	<input type="text"/>
標準 *	<input type="text" value="【チュートリアル】 アンケート設問登録"/>								
英語	<input type="text"/>								
日本語	<input type="text"/>								
中国語 (中華人民共和国)	<input type="text"/>								
フローカテゴリ *	<div style="margin-bottom: 5px;">🔍 検索/新規作成</div> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%; padding: 2px;">ID *</td> <td style="padding: 2px;">tutorial_category</td> </tr> <tr> <td style="padding: 2px;">名称</td> <td style="padding: 2px;">【チュートリアル】 チュートリアルカテゴリ</td> </tr> </table>	ID *	tutorial_category	名称	【チュートリアル】 チュートリアルカテゴリ				
ID *	tutorial_category								
名称	【チュートリアル】 チュートリアルカテゴリ								
備考	<div style="border: 1px solid gray; height: 40px;"></div>								

業務ロジックのURL設定

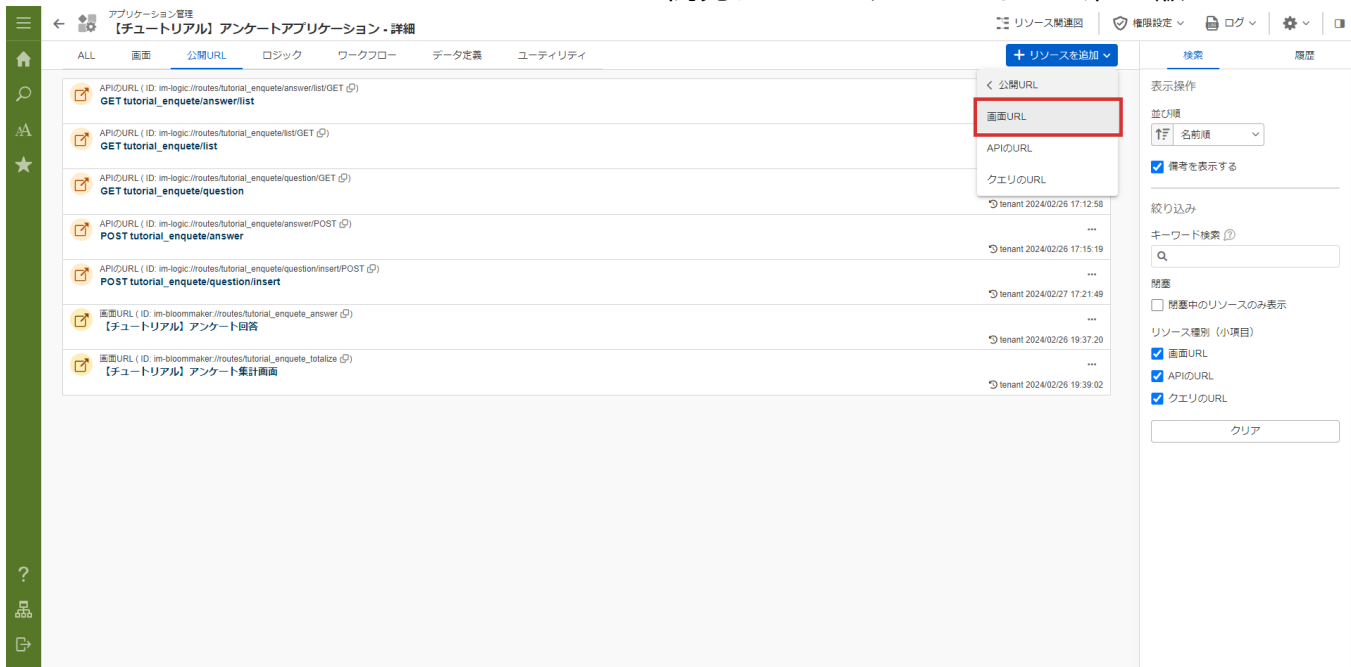
作成した業務ロジックを外部から利用できるようにするため、URLの設定を行います。

一覧取得処理のURL設定

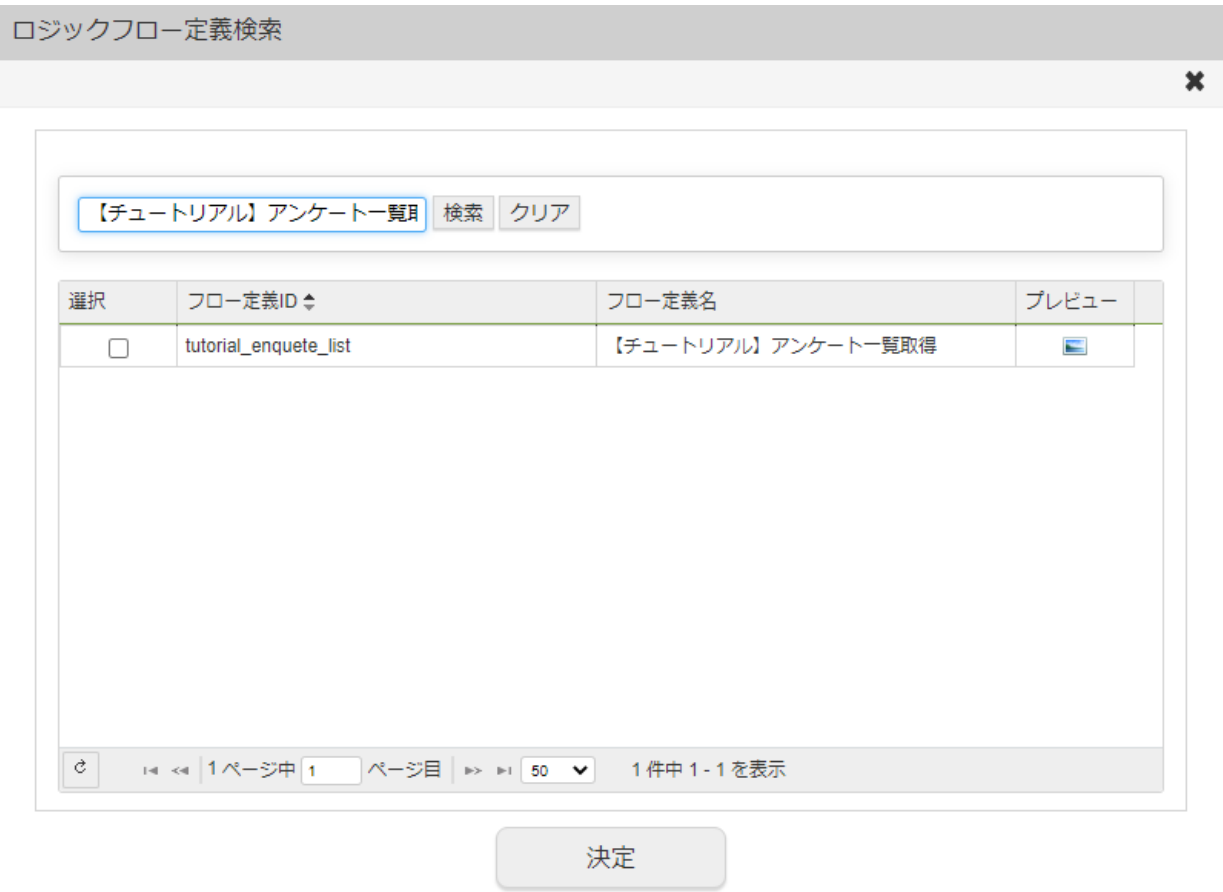
アンケート情報取得処理のURL設定を行います。

「【チュートリアル】 アンケートアプリケーション」の管理画面に遷移し、「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「APIのURL」を選択し「ロジックフロールーティング定義編集画面」を開きます。



「ロジックフロー定義編集画面」の「対象ロジックフロー定義情報」 - 「対象フロー」 - 「検索」からフロー検索ダイアログを表示します。「ロジックフロー定義検索」ダイアログ上で「【チュートリアル】アンケート一覧取得」を選択し、決定ボタンをクリックします。



「ロジックフロールーティング定義情報」の「ルーティング」に「tutorial_enquete/list」と設定します。「ロジックフロールーティング定義情報」の「認可URI」にも「tutorial_enquete/list」と設定します。その他の設定は標準のままとし、登録ボタンをクリックしてURLを設定します。

対象ロジックフロー定義情報

対象フロー フロー定義ID フロー定義名

バージョン番号 最新バージョンを利用する 利用するバージョンを指定する

ロジックフロールーティング定義情報

ルーティング メソッド 認証方法 認可URI セキュアトークンを利用する レスポンス種別 レスポンスヘッダ

ヘッダ名	ヘッダ値	削除
<input type="text"/>	<input type="text"/>	<input type="button" value="✖"/>

アンケート登録処理のURL設定

アンケート情報取得処理のURL設定を行います。

「【チュートリアル】アンケートアプリケーション」の管理画面に遷移し、「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「APIのURL」を選択し「ロジックフロールーティング定義編集画面」を開きます。

アプリケーション管理
【チュートリアル】アンケートアプリケーション - 詳細

リソース関連図 権限設定 ログ 設定

ALL 画面 公開URL ロジック ワークフロー データ定義 ユーティリティ

+ リソースを追加

検索 履歴

表示操作
並び順
↑ 名前順

備考を表示する

絞り込み
キーワード検索
Q

開基
 開基中のリソースのみ表示

リソース種別 (小項目)
 画面URL
 APIのURL
 クエリのURL

API/画面URL (ID)	メソッド	パス	作成日時
APIのURL (ID: im-logic/routes/tutorial_enquete/answerlist/GET)	GET	im-logic/routes/tutorial_enquete/answerlist	tenant 2024/02/26 17:12:58
APIのURL (ID: im-logic/routes/tutorial_enquete/list/GET)	GET	im-logic/routes/tutorial_enquete/list	tenant 2024/02/26 17:15:19
APIのURL (ID: im-logic/routes/tutorial_enquete/question/GET)	GET	im-logic/routes/tutorial_enquete/question	tenant 2024/02/27 17:21:49
APIのURL (ID: im-logic/routes/tutorial_enquete/answer/POST)	POST	im-logic/routes/tutorial_enquete/answer	tenant 2024/02/26 19:37:20
APIのURL (ID: im-logic/routes/tutorial_enquete/question/insert/POST)	POST	im-logic/routes/tutorial_enquete/question/insert	tenant 2024/02/26 19:39:02
画面URL (ID: im-bloommaker/routes/tutorial_enquete_answer)		im-bloommaker/routes/tutorial_enquete_answer	tenant 2024/02/26 19:39:02
画面URL (ID: im-bloommaker/routes/tutorial_enquete_totalize)		im-bloommaker/routes/tutorial_enquete_totalize	tenant 2024/02/26 19:39:02

「ロジックフロー定義編集画面」の「対象ロジックフロー定義情報」 - 「対象フロー」 - 「検索」からフロー検索ダイアログを表示します。「ロジックフロー定義検索」ダイアログ上で「【チュートリアル】アンケート設問登録」を選択し、決定ボタンをクリックします。



【チュートリアル】 アンケート設問登録
検索
クリア

選択	フロー定義ID	フロー定義名	プレビュー
<input type="checkbox"/>	tutorial_enquete_insert	【チュートリアル】 アンケート設問登録	

◀
1 ページ中 1
ページ目
▶▶
50
▼
1 件中 1 - 1 を表示

決定

「ロジックフロールーティング定義情報」の「ルーティング」を「tutorial_enquete/question/insert」と設定します。
 「ロジックフロールーティング定義情報」の「メソッド」に「POST」と指定します。
 「ロジックフロールーティング定義情報」の「認可URI」にも「tutorial_enquete/question/insert」と設定します。
 その他の設定は標準のままとし、登録ボタンをクリックしてURLを設定します。

対象ロジックフロー定義情報

対象フロー *	<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;"> 検索 </div> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; border: 1px solid #ccc; padding: 2px;">フロー定義ID *</td> <td style="border: 1px solid #ccc; padding: 2px;">tutorial_enquete_insert</td> </tr> <tr> <td style="border: 1px solid #ccc; padding: 2px;">フロー定義名</td> <td style="border: 1px solid #ccc; padding: 2px;">【チュートリアル】 アンケート設問登録</td> </tr> </table>	フロー定義ID *	tutorial_enquete_insert	フロー定義名	【チュートリアル】 アンケート設問登録
フロー定義ID *	tutorial_enquete_insert				
フロー定義名	【チュートリアル】 アンケート設問登録				
バージョン番号 *	<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;"> <input checked="" type="radio"/> 最新バージョンを利用する <input type="radio"/> 利用するバージョンを指定する </div> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 60%; border: 1px solid #ccc; padding: 2px;">利用バージョン *</td> <td style="border: 1px solid #ccc; padding: 2px;"><input type="text"/></td> </tr> </table>	利用バージョン *	<input type="text"/>		
利用バージョン *	<input type="text"/>				

ロジックフロールーティング定義情報

ルーティング *	/imart/logic/api/ <input style="width: 80%;" type="text" value="tutorial_enquete/question/insert"/>				
メソッド *	POST ▼				
認証方法 *	IMAuthentication ▼				
認可URI *	im-logic-rest:// <input style="width: 80%;" type="text" value="tutorial_enquete/question/insert"/>				
セキュアトークンを利用する	<input type="checkbox"/>				
レスポンス種別 *	JSONに変換して返却 ▼				
レスポンスヘッダ	<div style="border: 1px solid #ccc; padding: 2px; margin-bottom: 5px;"> + 追加 </div> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 70%; border: 1px solid #ccc; padding: 2px;">ヘッダ名 *</td> <td style="width: 30%; border: 1px solid #ccc; padding: 2px;">ヘッダ値 *</td> </tr> <tr> <td style="border: 1px solid #ccc; padding: 2px;"><input type="text"/></td> <td style="border: 1px solid #ccc; padding: 2px;"><input type="text"/></td> </tr> </table>	ヘッダ名 *	ヘッダ値 *	<input type="text"/>	<input type="text"/>
ヘッダ名 *	ヘッダ値 *				
<input type="text"/>	<input type="text"/>				

登録

アンケートの一覧表示と登録を行う画面を作成します。

アンケート一覧と登録画面は1つのコンテンツで作成します。

アンケート一覧からは登録画面とそれぞれのアンケートの回答、集計画面へのリンクを作成します。

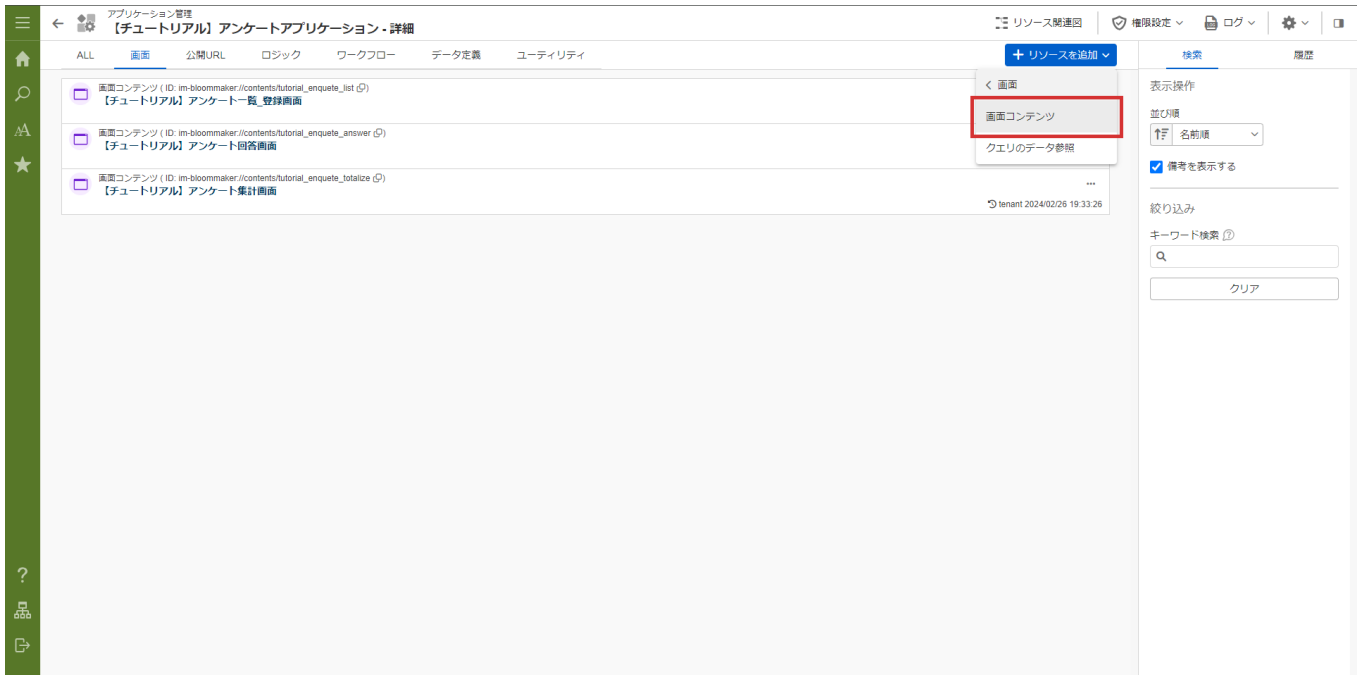
画面情報の設定

作成する画面の基本情報を設定します。

画面コンテンツの基本情報の設定

アンケート一覧、登録画面は新しい1つのコンテンツとして作成します。

「アプリケーション管理画面」右上の「リソースを追加」プルダウンから「画面」 - 「画面を新規作成」 - 「画面コンテンツ」を選択し、「IM-BloomMaker コンテンツ新規作成画面」を表示します。



画面コンテンツのカテゴリの検索ボタンをクリックし「業務テンプレートから作成」を設定します。

コンテンツIDに「tutorial_enquete_list」と設定します。

コンテンツ種別欄の「Bulma」を選択します。

コンテンツ名に「【チュートリアル】アンケート一覧_登録画面」と入力し登録ボタンをクリックして登録します。



画面コンテンツの作成

アプリケーション管理の「画面」タブから作成した「【チュートリアル】アンケート一覧_登録画面」を選択し、コンテンツ画面を開きます。

コンテンツ画面の「デザイン編集画面」をクリックし、デザイナー画面を開きます。

基本	
カテゴリ	カテゴリID: im_bouquet カテゴリ名: 業務テンプレートから作成
コンテンツID	tutorial_enquete_list
コンテンツ種別	<input checked="" type="radio"/> Bulma ① オープンソースのCSSフレームワーク Bulma を利用した画面を作成できます。レスポンシブデザイン、Flexbox を簡単に利用でき、華やかで見やすい配色と配置が可能なエレメントを利用でき、 <input type="radio"/> Bulma Theme Colored ① Accel Platform のテーマ色と同系統の配色で Bulma を利用できるようにします。
最新バージョン番号	1
コンテンツ名	標準 * <input type="text" value="【チュートリアル】アンケート一覧_登録画面"/>
備考	標準 <input type="text"/>

変数の設定

画面で利用する変数を作成します。

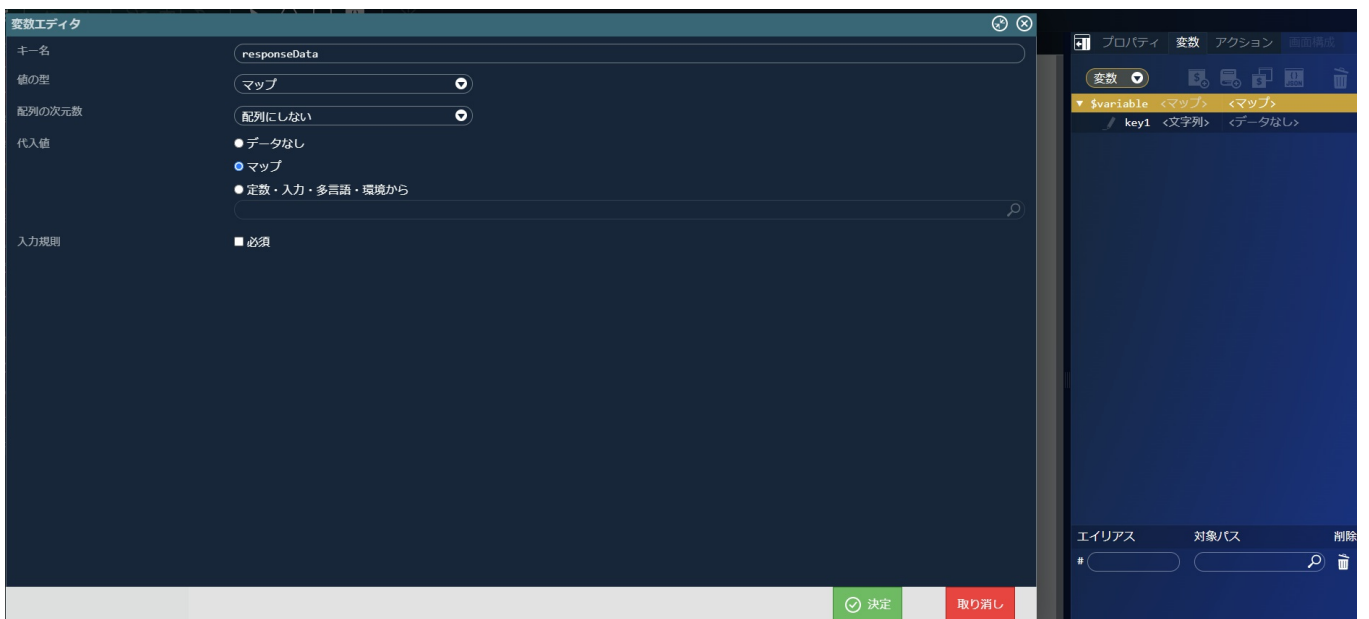
最初に、アンケート一覧取得処理画面に関連する変数を作成します。

一覧画面では、アンケート一覧取得処理からアンケート情報を取得し、それを元に一覧画面表示用に変換して利用します。

まずは、アンケート一覧取得処理からアンケート情報を取得した値を格納する変数を作成します。画面右側の「変数」タブを選択します。

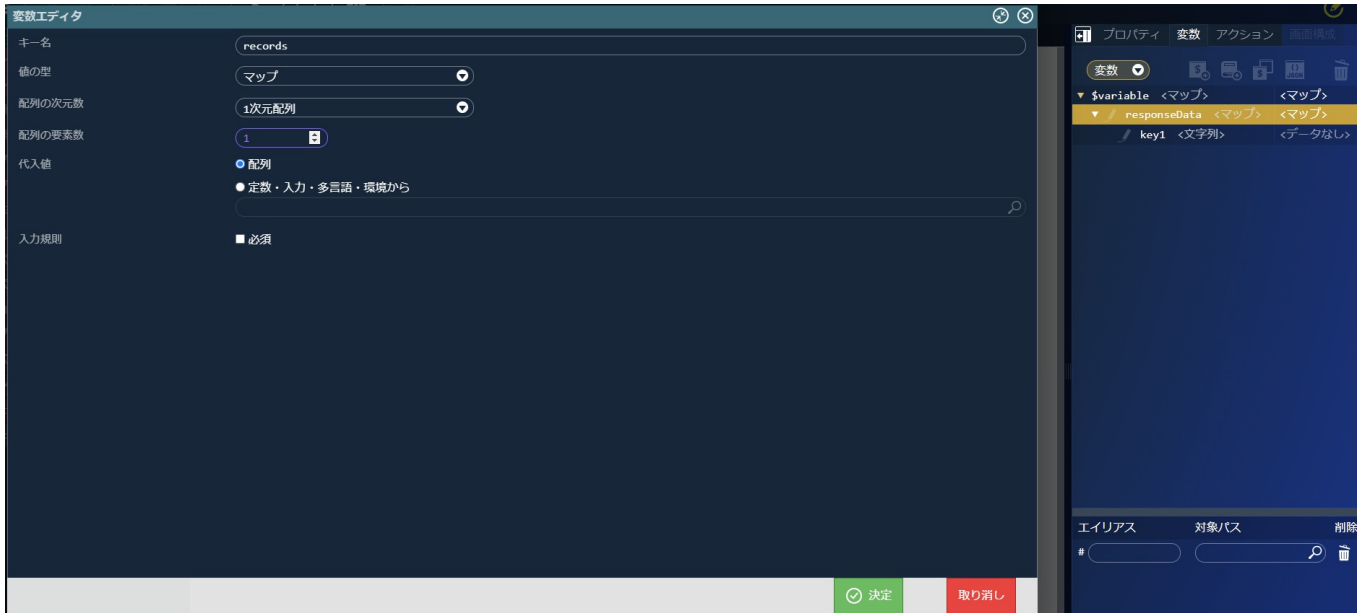
「\$variable」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「responseData」と入力し、値の型を「マップ」に、代入値も「マップ」に設定して「決定」ボタンをクリックします。

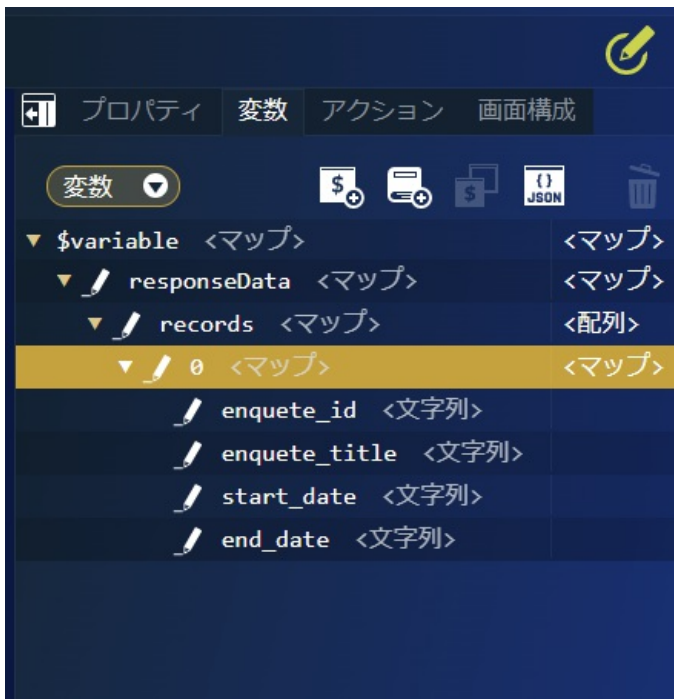


追加した「\$variable > responseData」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「records」と入力し、値の型を「マップ」に、配列の次元数を「1次元配列」、配列の要素数を「1」に設定して




追加した「\$variable > responseData > records > 0」配下に、「enquete_id」、「enquete_title」、「start_date」、「end_date」という変数を追加します。
値の型は全て「文字列」です。




変数はJSON入力でも以下のコードを記載しても設定できます。

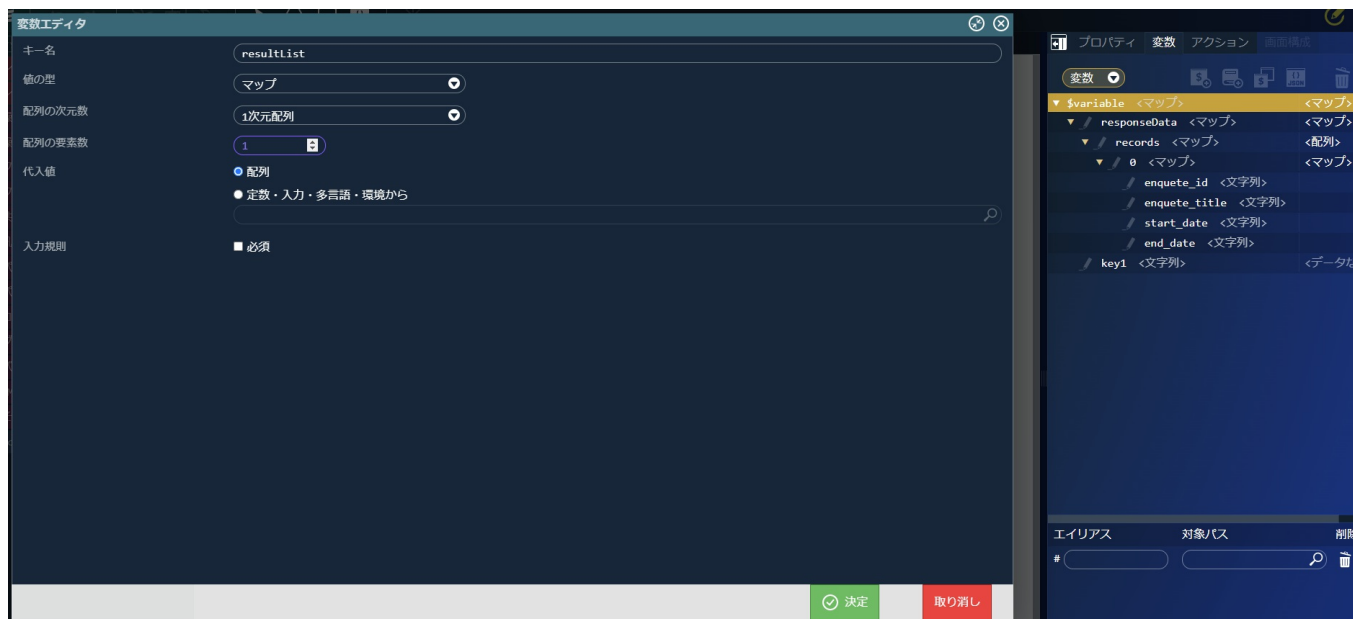
```
"responseData": {
  "records": [
    {
      "enquete_id": "",
      "enquete_title": "",
      "start_date": "",
      "end_date": ""
    }
  ]
}
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

続けて、一覧画面表示用の変数を作成します。

「\$variable」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「resultList」と入力し、値の型を「マップ」に、配列の次元数を「1次元配列」、配列の要素数を「1」に設定して「決定」ボタンをクリックします。




作成した「\$variable > resultList > 0」配下に、「title」、「startDate」、「endDate」、「answerUrl」、「totalUrl」という変数を追加します。

値の型は全て「文字列」です。




変数はJSON入力で以下のコードを記載しても設定できます。

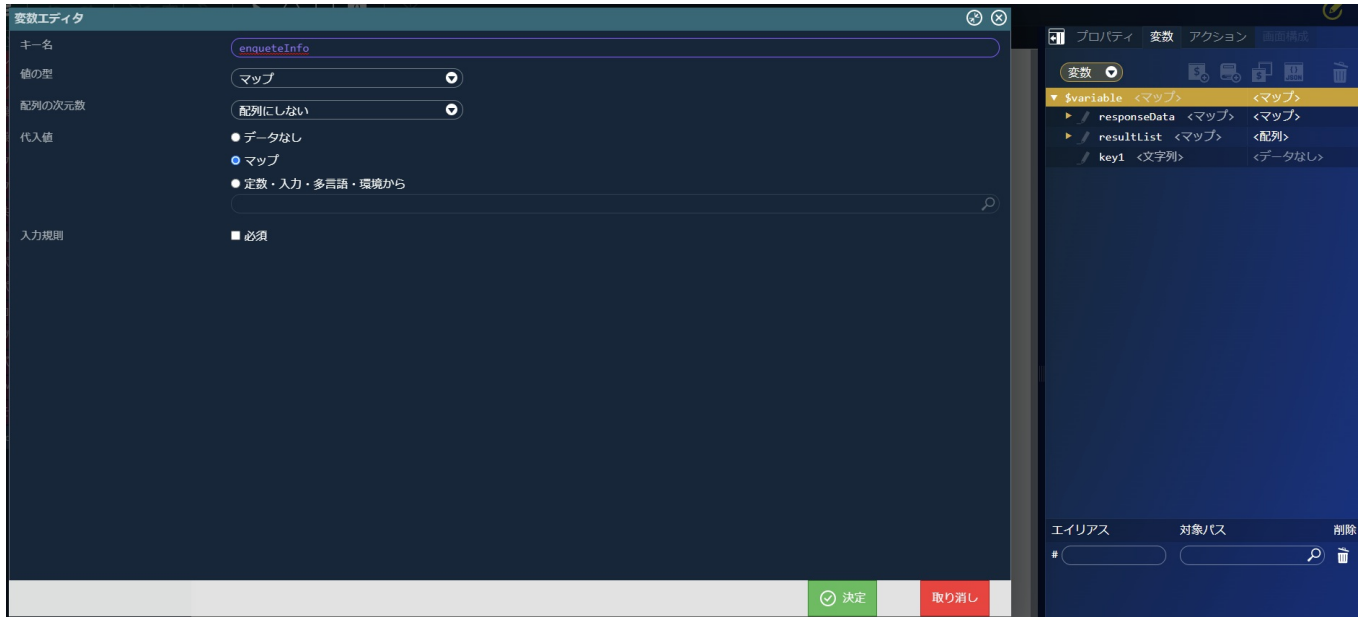
```
"resultList": [
  {
    "title": "",
    "startDate": "",
    "endDate": "",
    "answerUrl": "",
    "totalUrl": ""
  }
]
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表

次に、アンケート登録処理に対して送信するアンケートの登録内容を格納する変数を作成します。

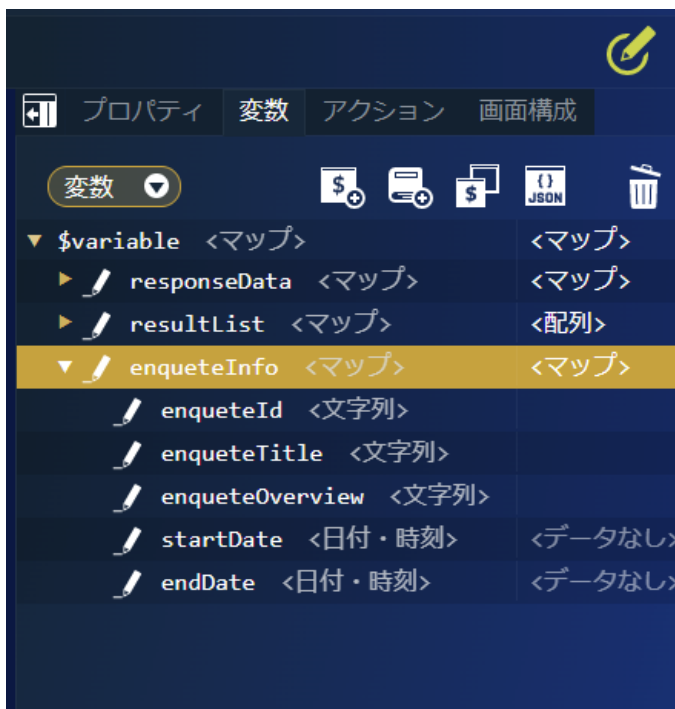
「\$variable」を選択し、 をクリックして変数エディタを表示します。

変数エディタのキー名に「enqueteInfo」と入力し、値の型を「マップ」に、代入値を「マップ」に変更して「決定」ボタンをクリックします。



作成した「\$variable > enqueteInfo」の配下に、アンケート情報を格納する値を追加します。

「enqueteId」、「enqueteTitle」、「enqueteOverview」を「文字列」で、「startDate」と「endDate」を「日付・時刻」で追加します。

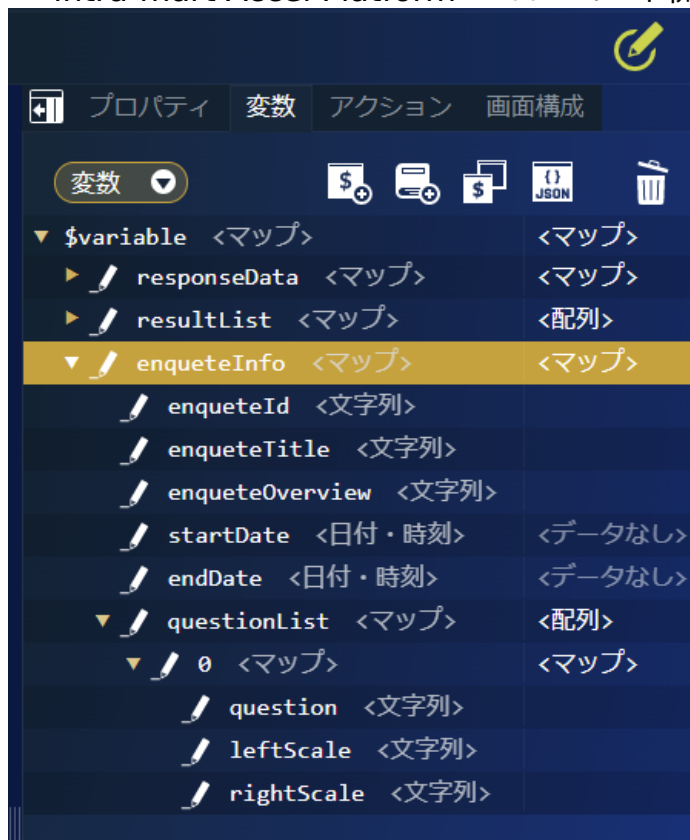


続けて、「\$variable > enqueteInfo」の配下に、アンケート設問情報を格納する値を追加します。

「\$variable > enqueteInfo」を選択し、 をクリックして変数エディタを表示します。


変数エディタのキー名に「questionList」と入力し、値の型を「マップ」に、配列の次元数を「1次元配列」、配列の要素数を「1」に変更して「決定」ボタンをクリックします。

作成した「\$variable > enqueteInfo > questionList > 0」配下に、「question」、「leftScale」、「rightScale」という変数を値の型を「文字列」で追加します。



変数はJSON入力で以下のコードを記載しても設定できます。

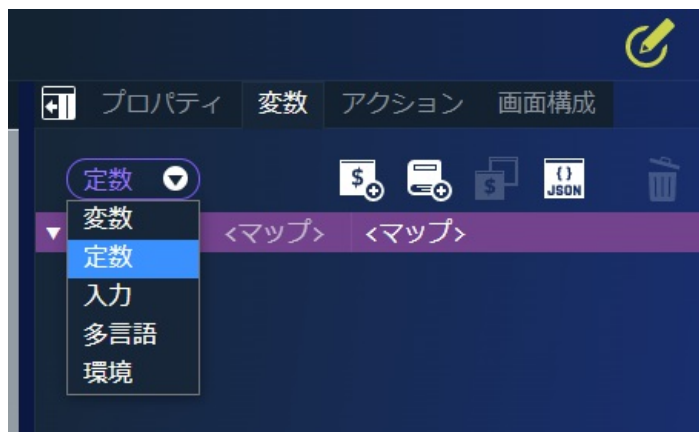
```
"enqueteInfo": {
  "enqueteId": "",
  "enqueteTitle": "",
  "enqueteOverview": "",
  "startDate": null,
  "endDate": null,
  "questionList": [
    {
      "question": "",
      "leftScale": "",
      "rightScale": ""
    }
  ]
}
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

回答画面、集計画面へアクセスするURLを定数で設定します。

回答画面、集計画面へのリンクは、定数で設定したURLに、一覧で取得したアンケートIDを繋げて作成します。

「変数」タブで「定数」を選択します。

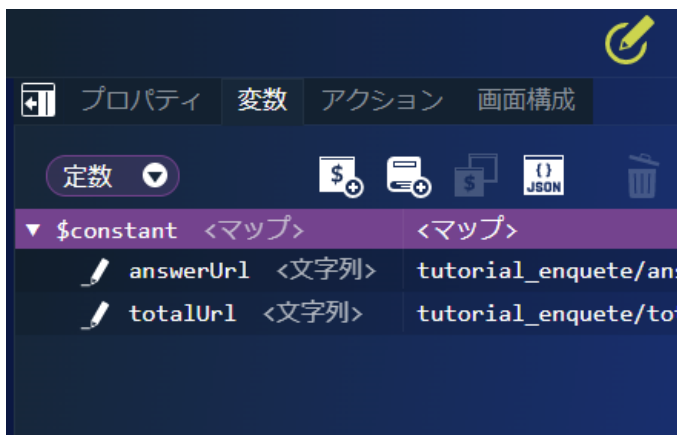


まずは、回答画面を表示するURLを設定します。「\$constant」を選択し、 をクリックして定数エディタを表示します。

定数エディタのキー名に「answerUrl」と入力し、値の型を「文字列」に、代入値に「tutorial_enquete/answer?enquetId=」と設定します。


同様に、集計画面を表示するURLを設定します。「\$constant」を選択し、 をクリックして定数エディタを表示します。

定数エディタのキー名に「totalUrl」と入力し、値の型を「文字列」に、代入値に「tutorial_enquete/totalize?enquetId=」と設定します。



定数もJSON入力以下を記載して設定できます。

```
"answerUrl": "tutorial_enquete/answer?enquetId=",
"totalUrl": "tutorial_enquete/totalize?enquetId="
```

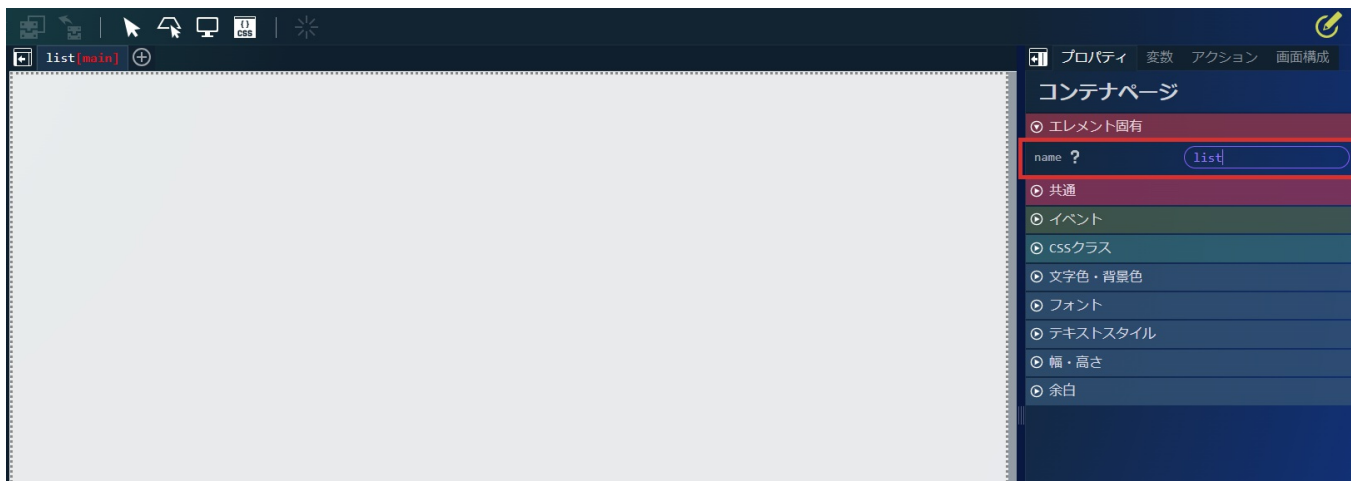
JSON入力する場合は、定数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

アンケート一覧画面の作成

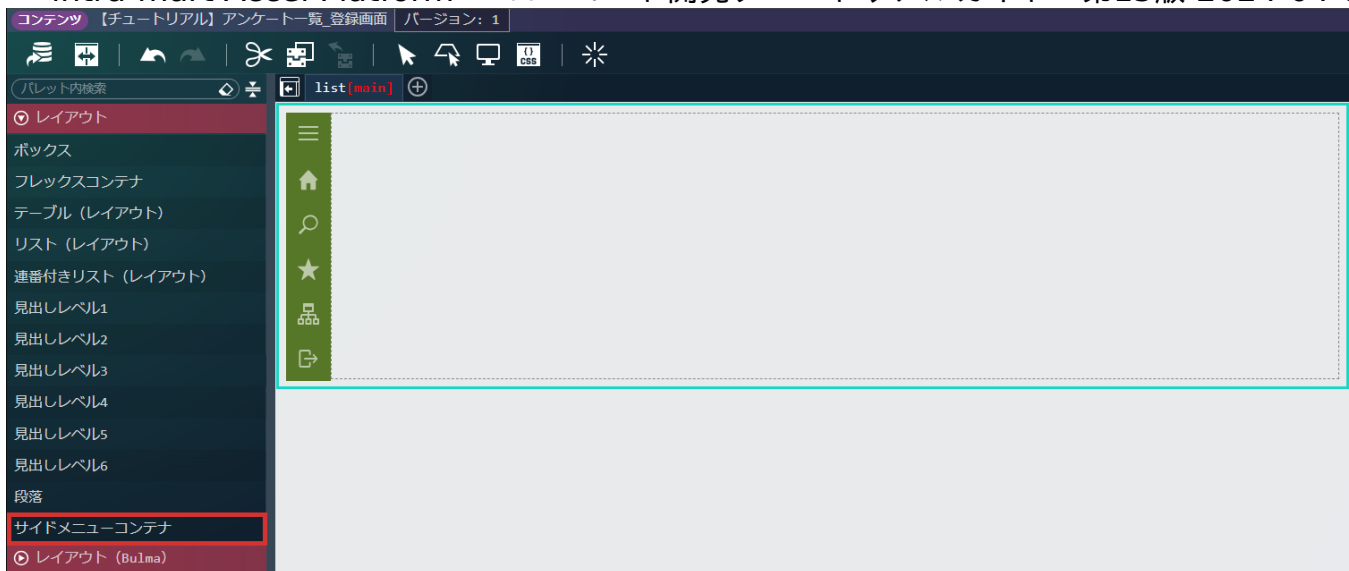
アンケートの一覧画面から作成します。

まずは、初期表示で作成された画面の名称を変更します。

デザイン編集エディタをクリックし、画面右側の「コンテナページ」 - 「エレメント固有」 - 「name」に「list」と設定します。

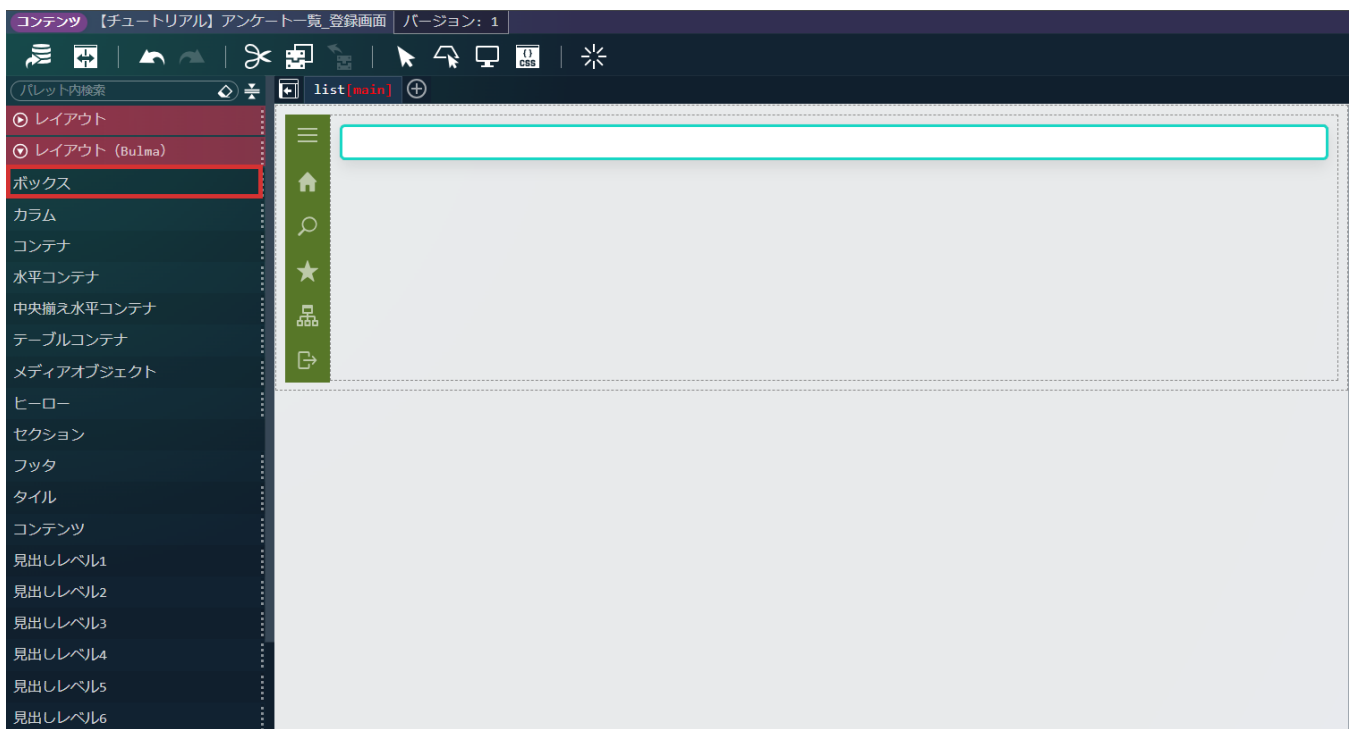


これで、タブの名称も「list」と変更されました。次に、画面にサイドメニューを表示するためにサイドメニューコンテナを配置します。画面左側のパレットの「レイアウト」 - 「サイドメニューコンテナ」を選択し、画面に配置します。

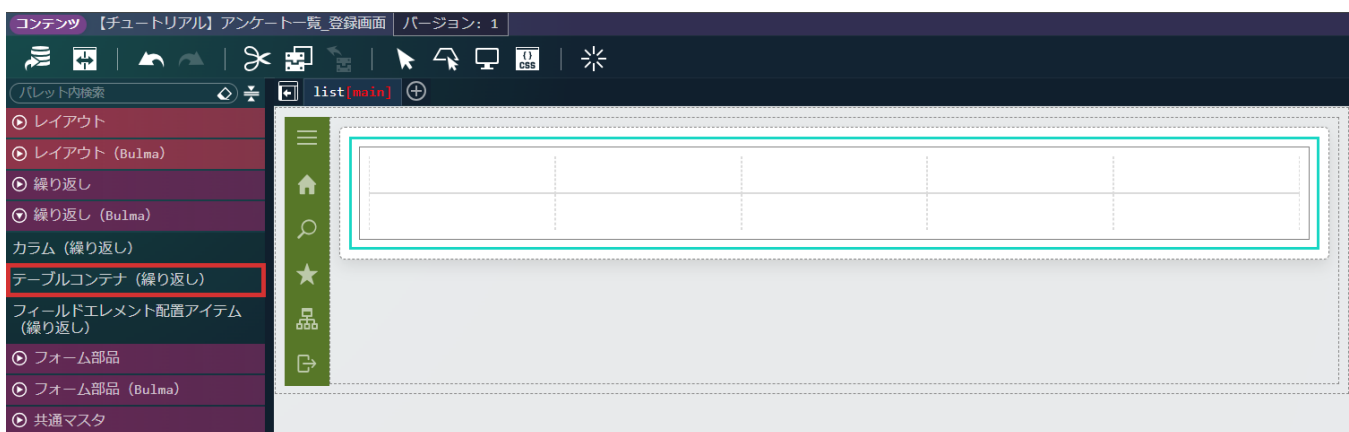


次に、アンケート一覧の表示領域を作成します。

画面左側のパレットの「レイアウト (Bulma)」 - 「ボックス」を選択し、先ほど配置した「サイドメニューコンテナ」の中に配置します。



画面左側のパレットの「繰り返し (Bulma)」 - 「テーブルコンテナ (繰り返し)」を選択し、先ほど配置した「ボックス」の中に配置します。



配置した「テーブルコンテナ（繰り返し）」配下の「テーブル（繰り返し）」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「list」の 🔍 を選択します。

プロパティ 変数 アクション 画面構成

繰り返し (Bulma) > テーブルコンテナ (繰り返し)

テーブル (繰り返し)

⊙ エレメント固有 エラー項目があります

- showHeader ? 固定値 変数値
- showFooter ? 固定値 変数値
- showBorder ? 固定値 変数値
- stripe ?
- narrow ?
- hoverable ? 固定値 変数値
- fullwidth ?
- columnCount ?
- list ?** 🔍

値を入力してください。

「\$variable > resultList」を選択し、「決定」ボタンをクリックして変数を設定します。

変数・定数・入力・多言語・環境セレクト

変数 定数 入力 多言語 環境

- ▼ \$variable <マップ> <マップ>
 - ▼ responseData <マップ> <マップ>
 - ▼ records <マップ> <配列>
 - ▼ 0 <マップ> <マップ>
 - enquete_id <文字列>
 - enquete_title <文字列>
 - start_date <文字列>
 - end_date <文字列>
 - ▼ resultList <マップ> <配列>
 - ▼ 0 <マップ> <マップ>
 - title <文字列>
 - startDate <文字列>
 - endDate <文字列>
 - answerUrl <文字列>
 - totalUrl <文字列>
 - ▼ enqueteInfo <マップ> <マップ>
 - enqueteId <文字列>
 - enqueteTitle <文字列>
 - enqueteOverview <文字列>
 - startDate <日付・時刻> <データなし>

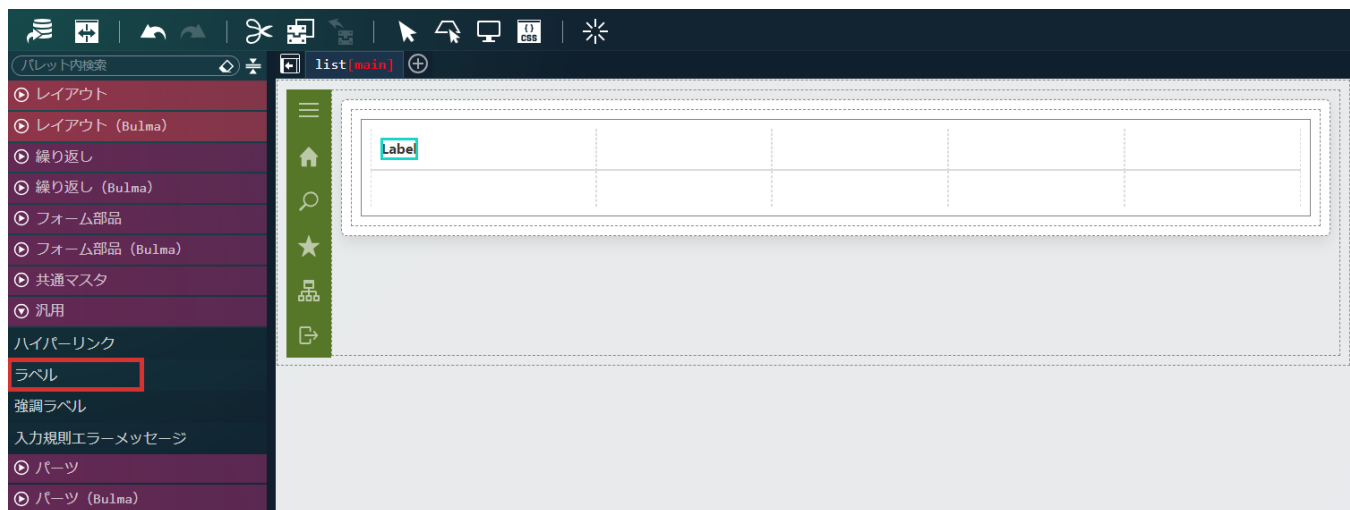
決定 取り消し

変数の設定は変数設定欄に直接以下のように入力してもかまいません。

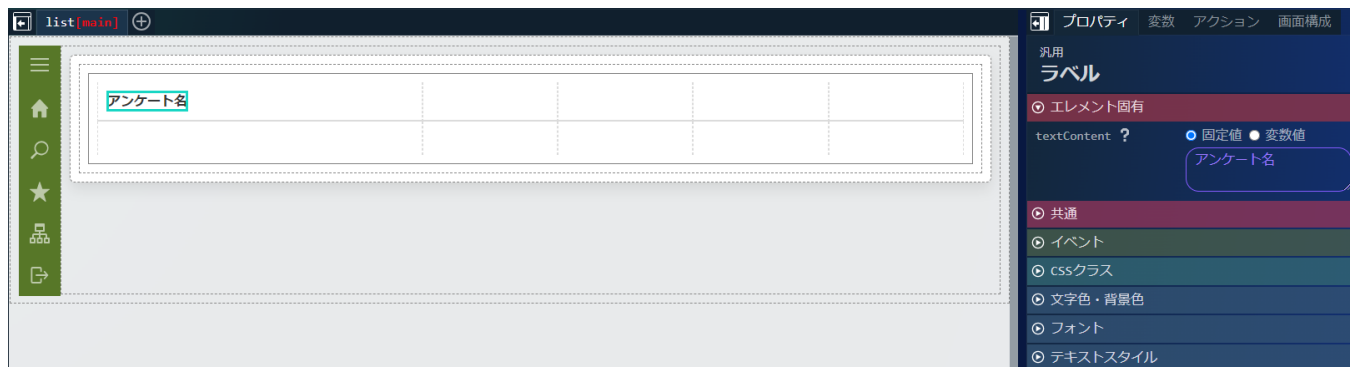
「\$variable.resultList」

続けて、テーブルヘッダに項目名を設定します。「テーブル（繰り返し）」は、「showHeader」にチェックが入っている場合、1行目がヘッダとして表示されます。

画面左側のパレットの「汎用」 - 「ラベル」を選択し、テーブル1行目の左端に配置します。



配置した「ラベル」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「textContent」に「アンケート名」と設定します。

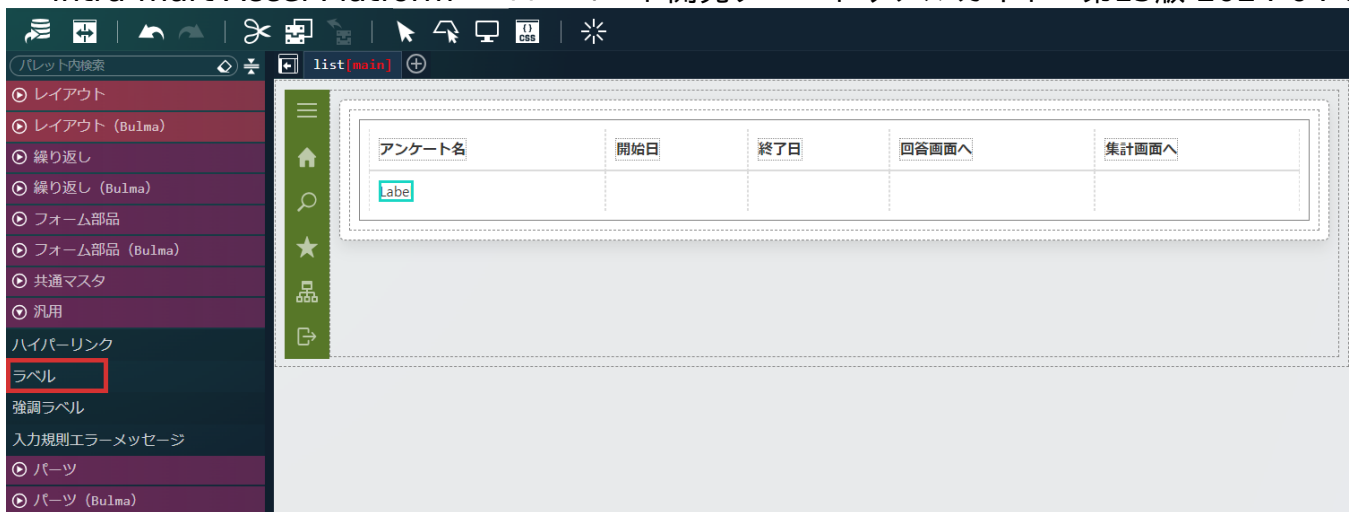



同様に、テーブルヘッダが左から「アンケート名」、「開始日」、「終了日」、「回答画面へ」、「集計画面へ」となるように設定します。



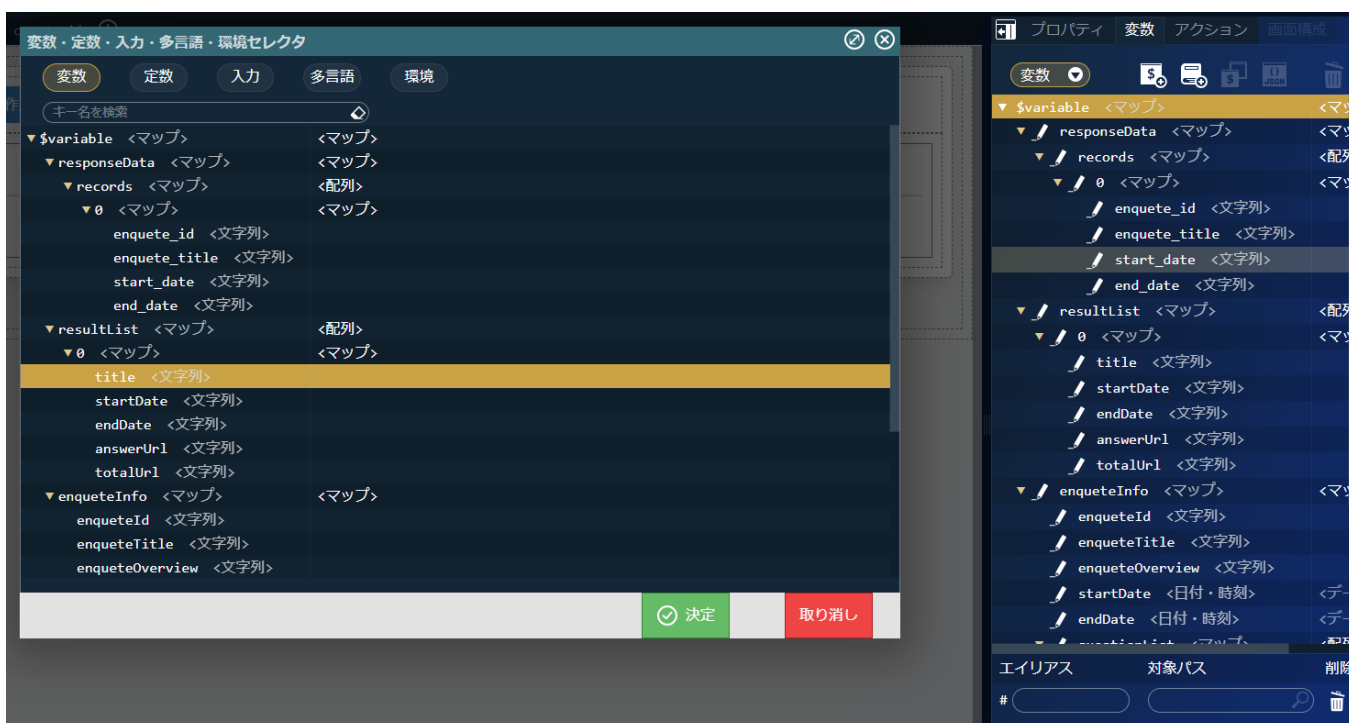
次に、一覧の表示項目を設定します。

画面左側のパレットの「汎用」 - 「ラベル」を選択し、テーブル1行目の左端に配置します。

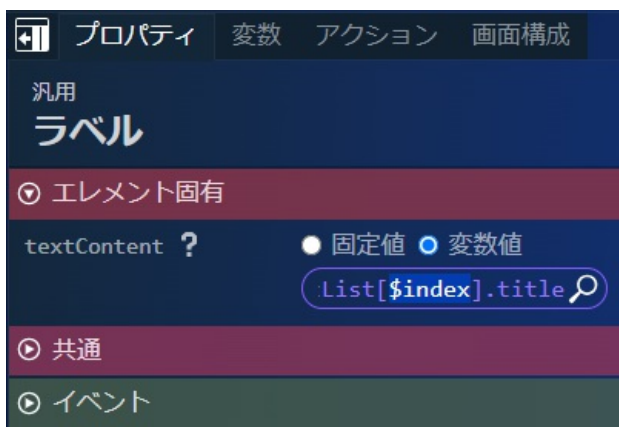


配置した「ラベル」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「textContent」を「変数値」に設定し、 を選択します。

「\$variable > resultList > 0 > title」を選択し、「決定」ボタンをクリックして変数を設定します。



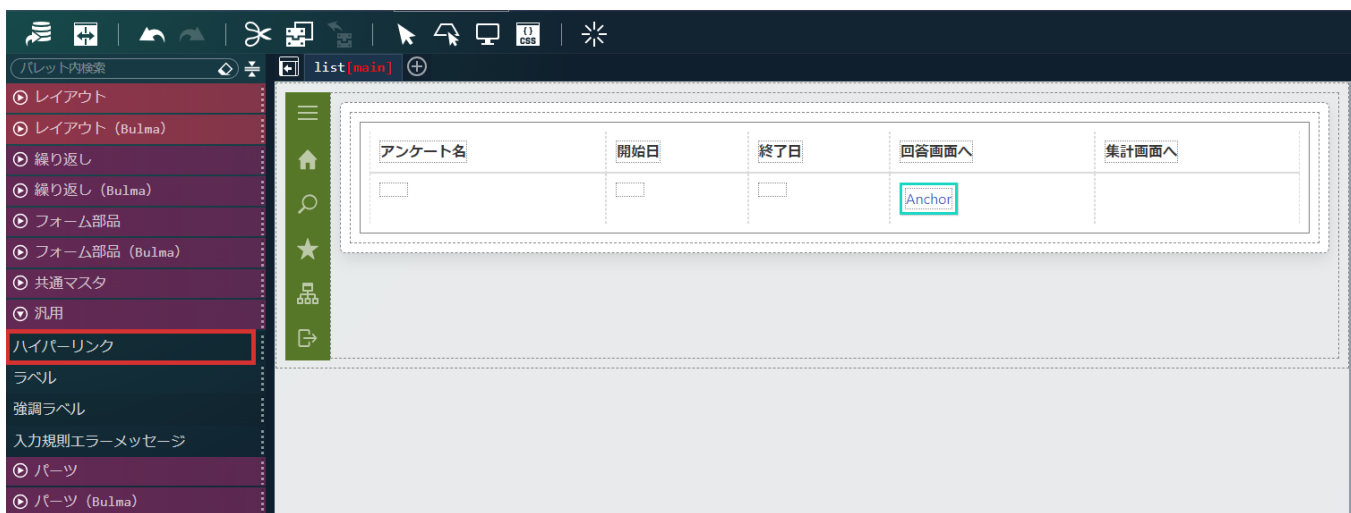
設定した変数を「\$variable.resultList[\$index].title」に変更します。



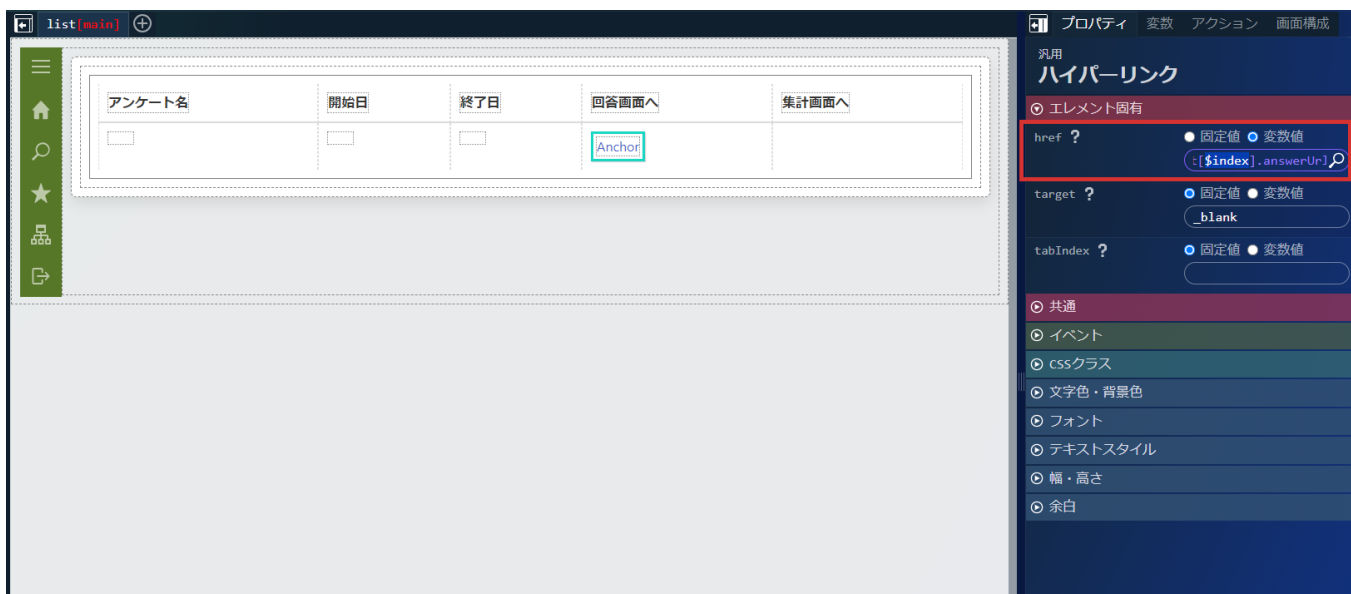
同様に、開始日の列に「ラベル」を配置し、変数値として「\$variable.resultList[\$index].startDate」を設定します。終了日の列にも「ラベル」を配置し、変数値として「\$variable.resultList[\$index].endDate」と設定します。




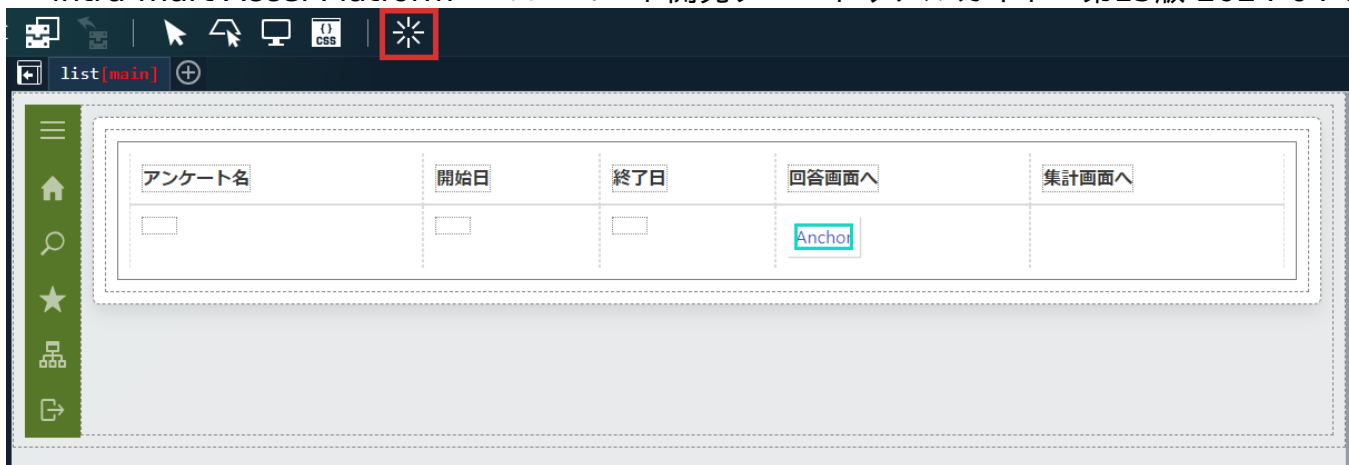
次に、回答画面へのリンクを作成します。
画面左側のパレットの「汎用」 - 「ハイパーリンク」を選択し、回答画面への列に配置します。



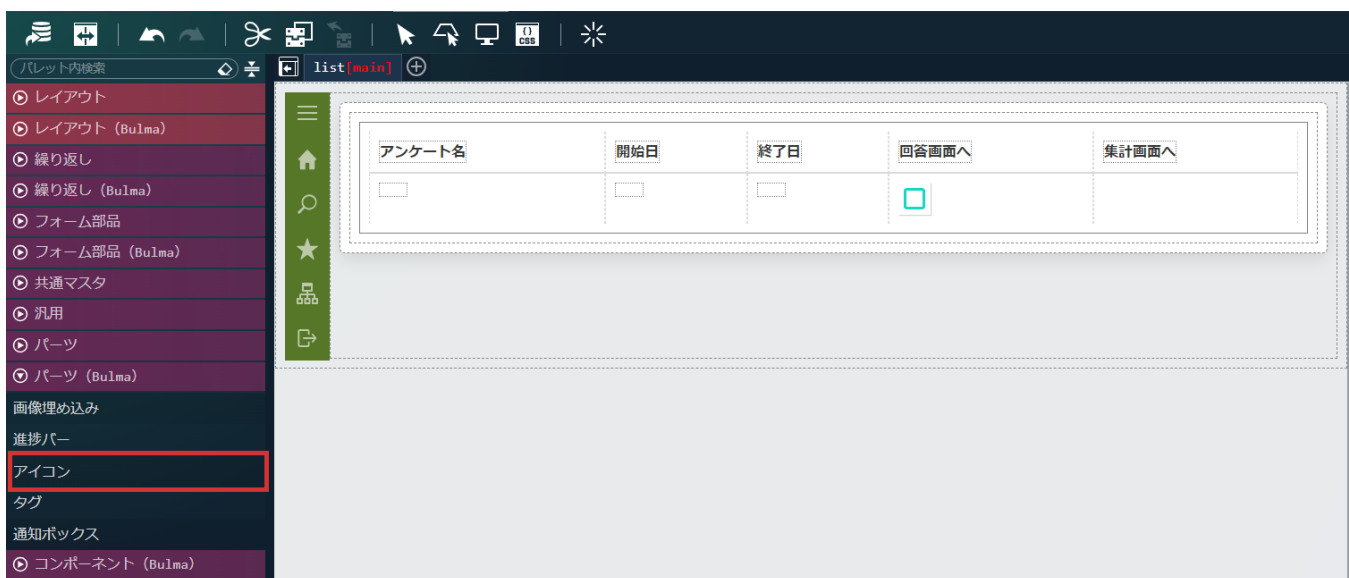
配置した「ハイパーリンク」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「href」を変数値に設定し、「`st[$index].answerUrl`」を設定します。
同様に、「プロパティ」タブ - 「エレメント固有」 - 「target」に「`_blank`」を設定します。



「ハイパーリンク」は、標準でラベルが設定されています。本チュートリアルでは、アイコンをリンクとして表示するため、「ハイパーリンク」内に設定されたラベルを選択し、かDeleteキーで削除します。



画面左側のパレットの「パーツ (Bulma)」 - 「アイコン」を選択し、「ハイパーリンク」の中に配置します。



配置した「アイコン」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「iconClass」に「fas fa-clipboard-check」を設定します。

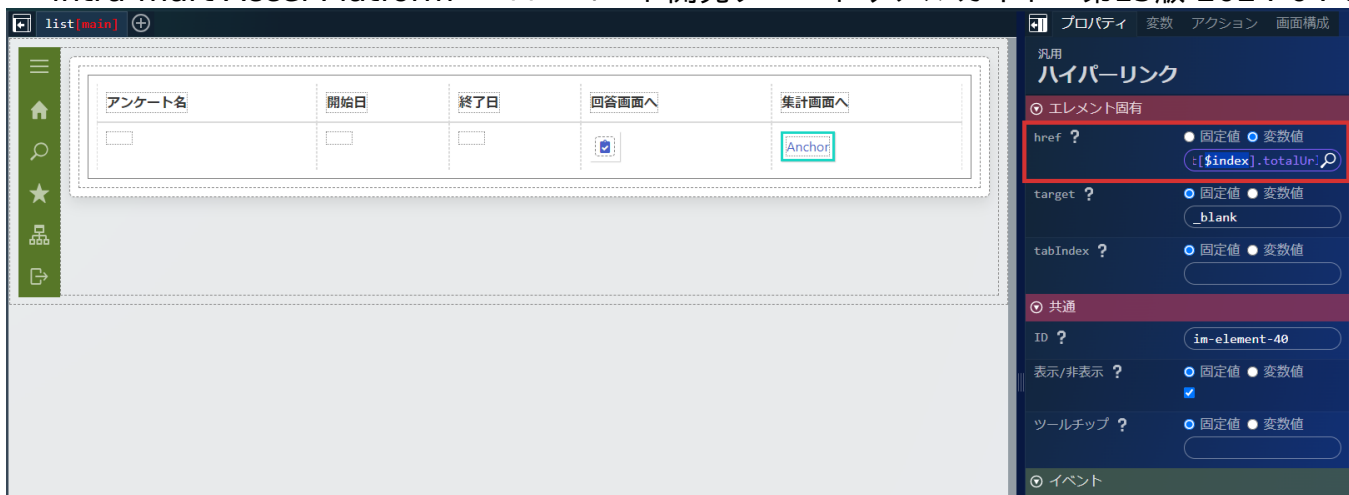



同様に、集計画面へのリンクを作成します。

画面左側のパレットの「汎用」 - 「ハイパーリンク」を選択し、集計画面への列に配置します。

配置した「ハイパーリンク」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「href」を変数値に設定し、「\$variable.resultList[\$index].totalUrl」を設定します。

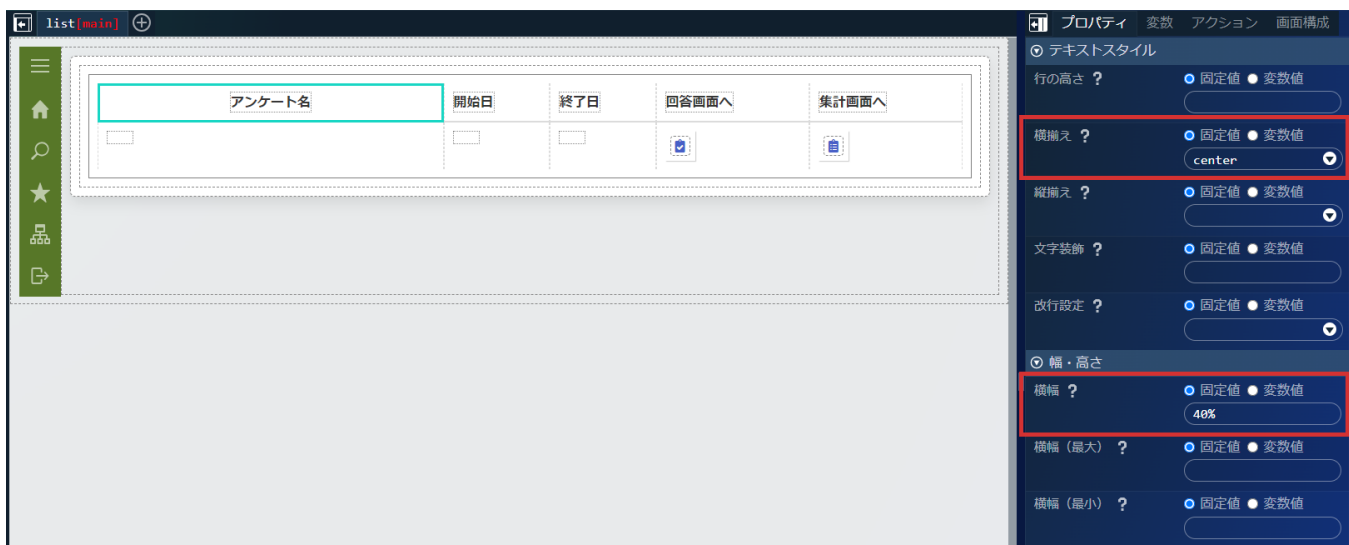
同様に、「プロパティ」タブ - 「エレメント固有」 - 「target」に「_blank」を設定します。



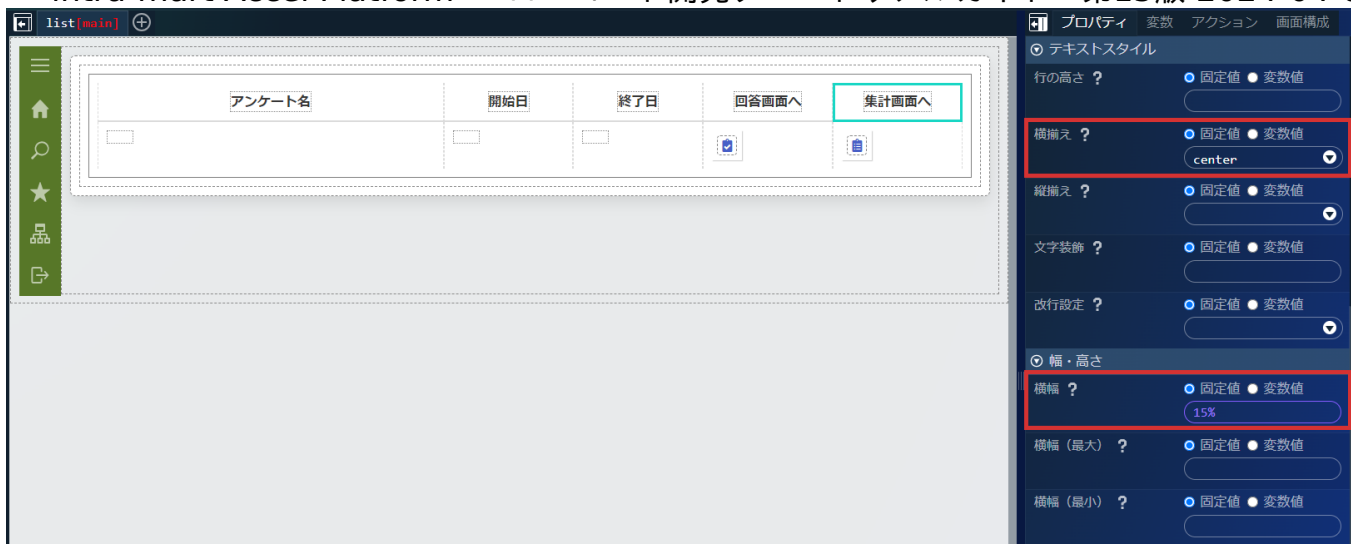
「ハイパーリンク」内に設定されたラベルを選択し、 がDeleteキーで削除し、画面左側のパレットの「パーツ (Bulma)」 - 「アイコン」を選択し、「ハイパーリンク」の中に配置します。
配置した「アイコン」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「iconClass」に「fas fa-clipboard-list」を設定します。



アンケート一覧のテーブルの表示幅や配置を調整します。
アンケート名の列のテーブルヘッダを選択し、「テキストスタイル」 - 「横揃え」を「center」に、「幅・高さ」 - 「横幅」に「40%」を設定します。



同様に、開始日、終了日、回答画面へ、集計画面への列のテーブルヘッダの「テキストスタイル」 - 「横揃え」を「center」に「幅・高さ」 - 「横幅」を「15%」に設定します。



続けて、アンケート名の列のテーブルコンテンツを選択し、「テキストスタイル」 - 「横揃え」を「center」に設定します。



同様に、開始日、終了日、回答画面へ、集計画面への列の「テキストスタイル」 - 「横揃え」を「center」に設定します。



最後に、新規作成画面へと遷移するボタンを配置します。

画面左側のパレットの「フォーム部品 (Bulma)」 - 「ボタン」を選択し、テーブルコンテナの上に配置します。



配置した「ボタン」を選択し、「プロパティ」タブ - 「エレメント固有」 - 「textContent」に「新規作成」を、「color」を「info」に設定します。



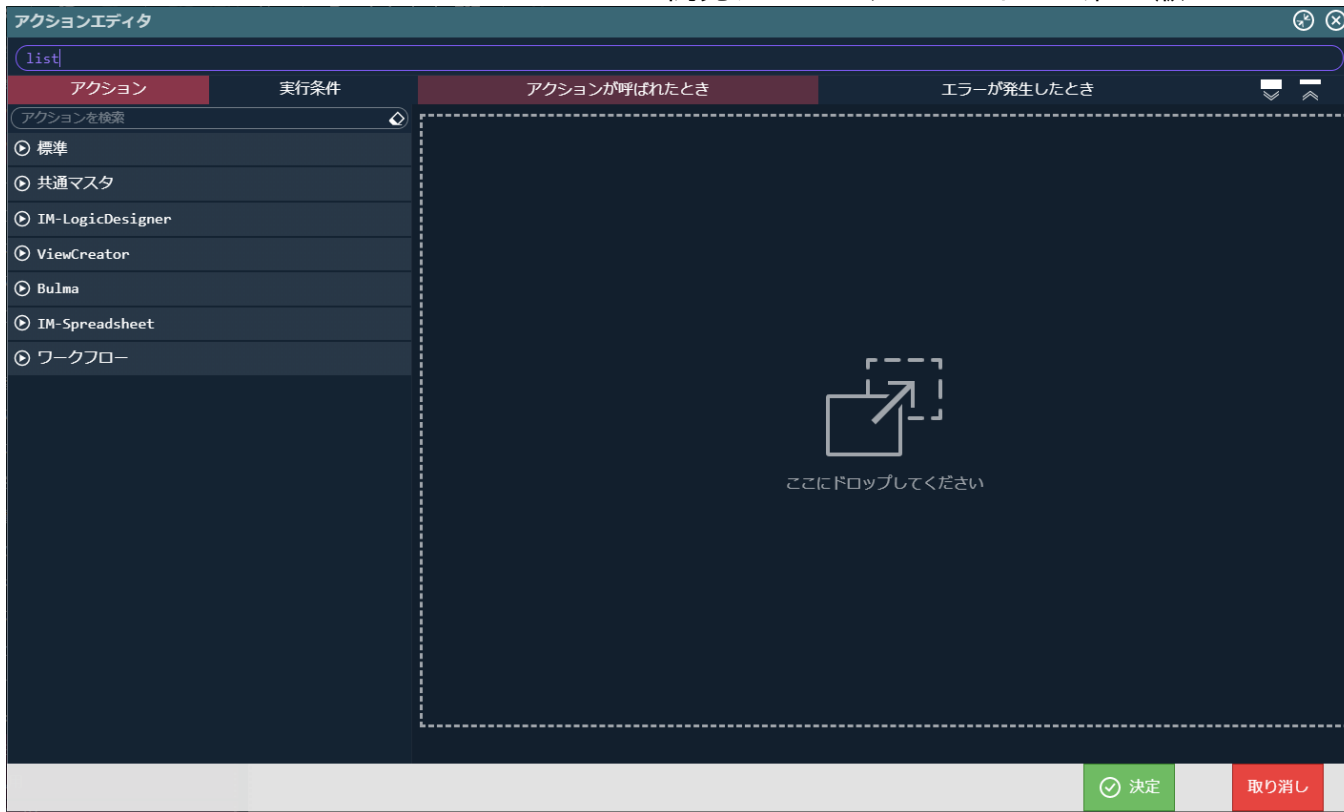
一覧画面のアクションの作成

一覧画面で利用するアクションを作成します。

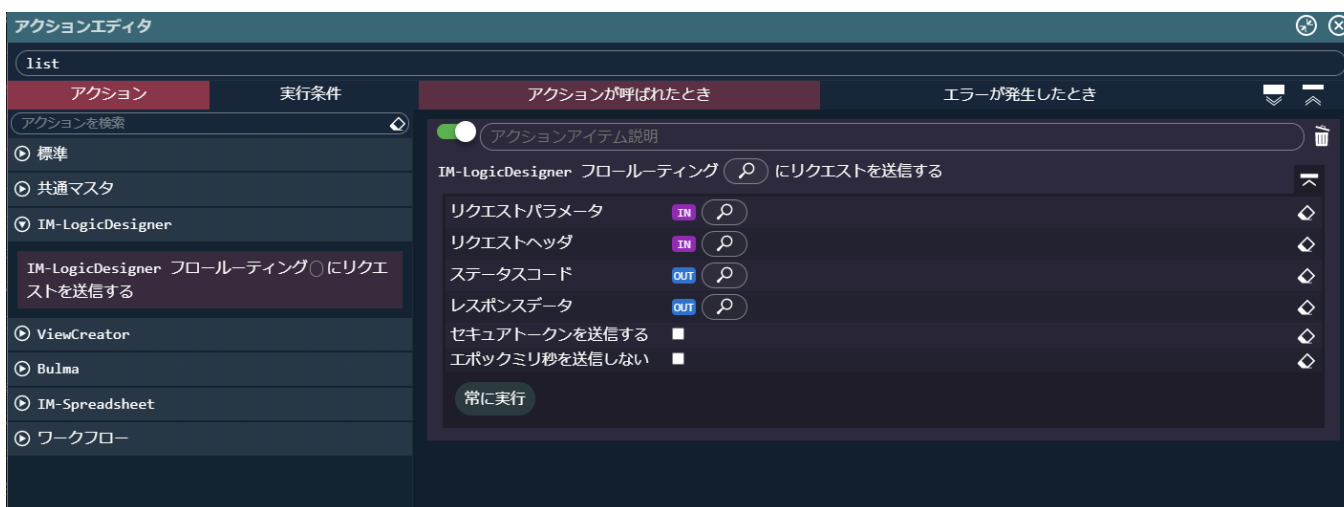
画面右側の「アクション」タブを選択し、 をクリックしアクションエディタを表示します。



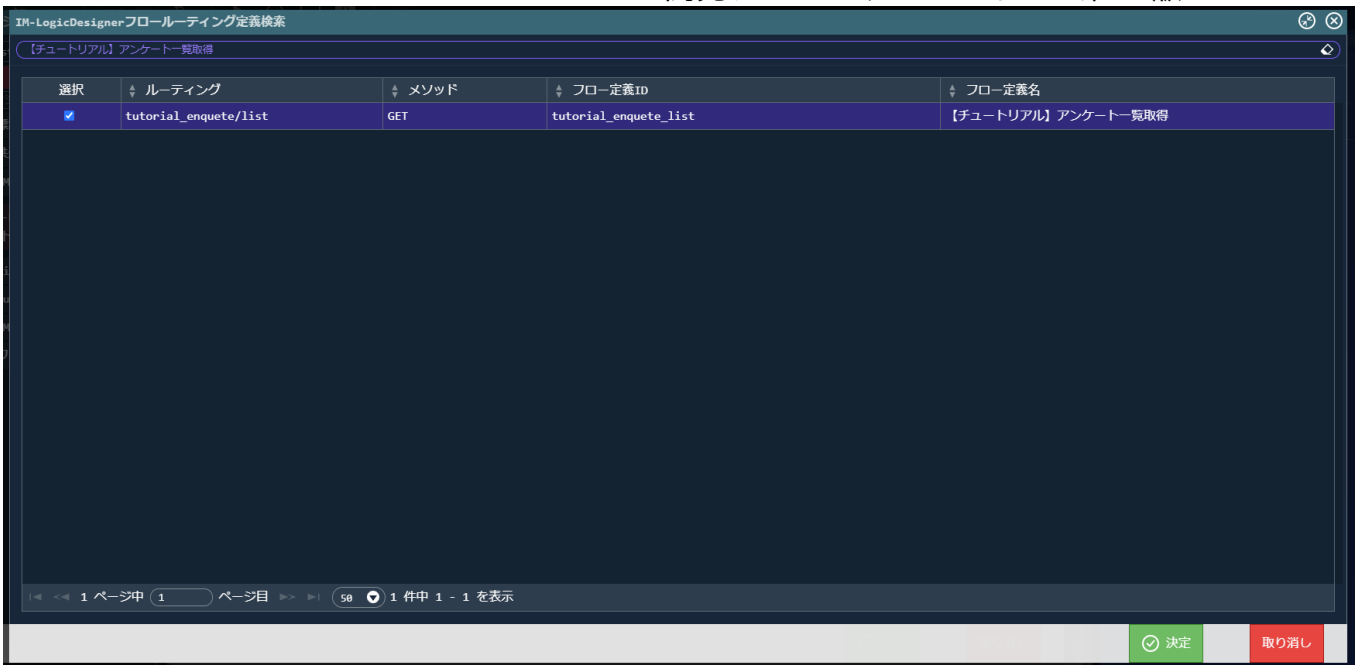
アクションエディタ上部の名称を「新しいアクション #1」を「list」に変更します。



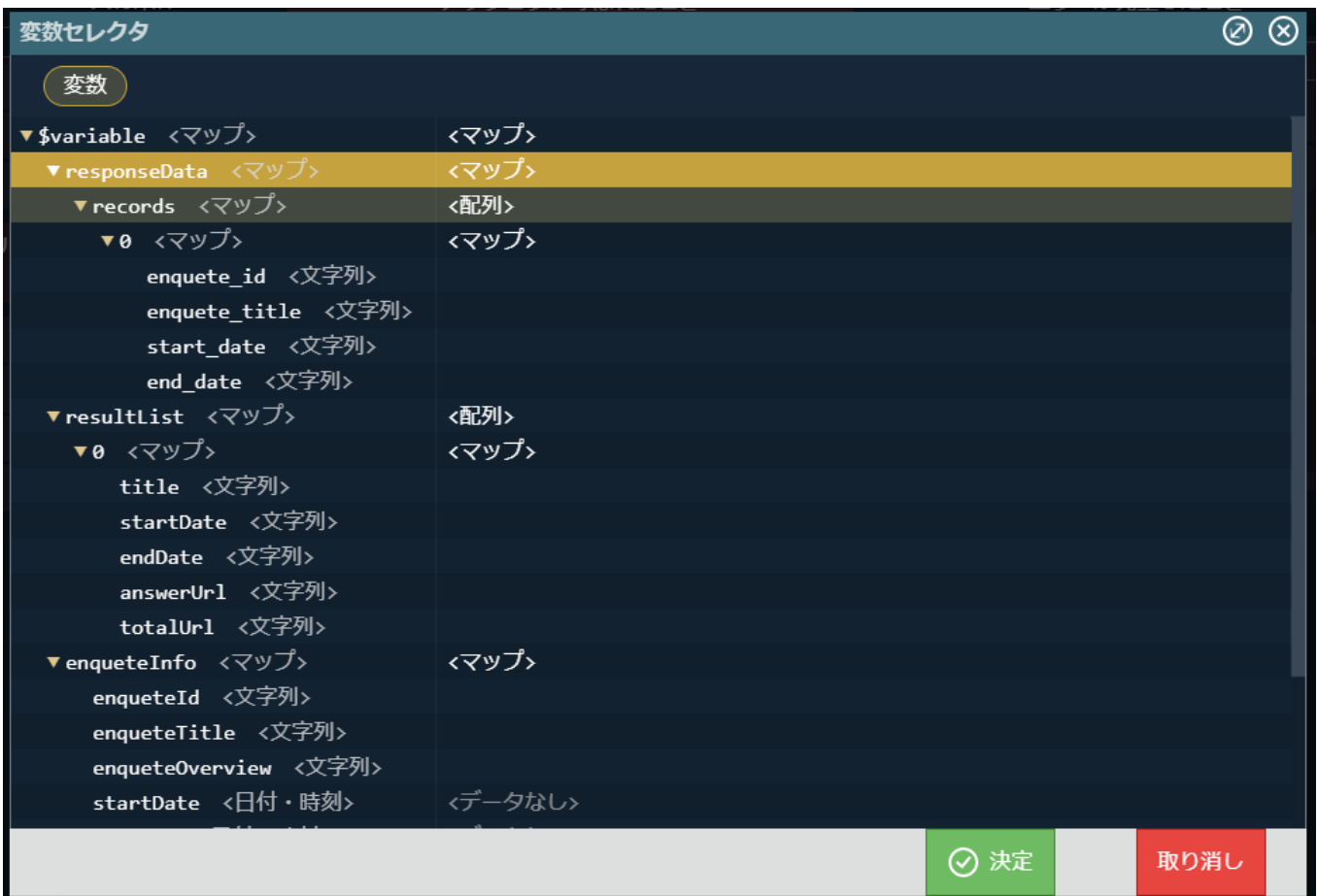
左側の「アクション」タブから「IM-LogicDesigner」 - 「IM-LogicDesigner フロールーティング○にリクエストを送信する」を配置します。



配置した「IM-LogicDesigner フロールーティング○にリクエストを送信する」の 🔍 をクリックし、フロールーティング定義検索ダイアログから「tutorial_enquete/list」を選択し、決定ボタンをクリックして設定します。

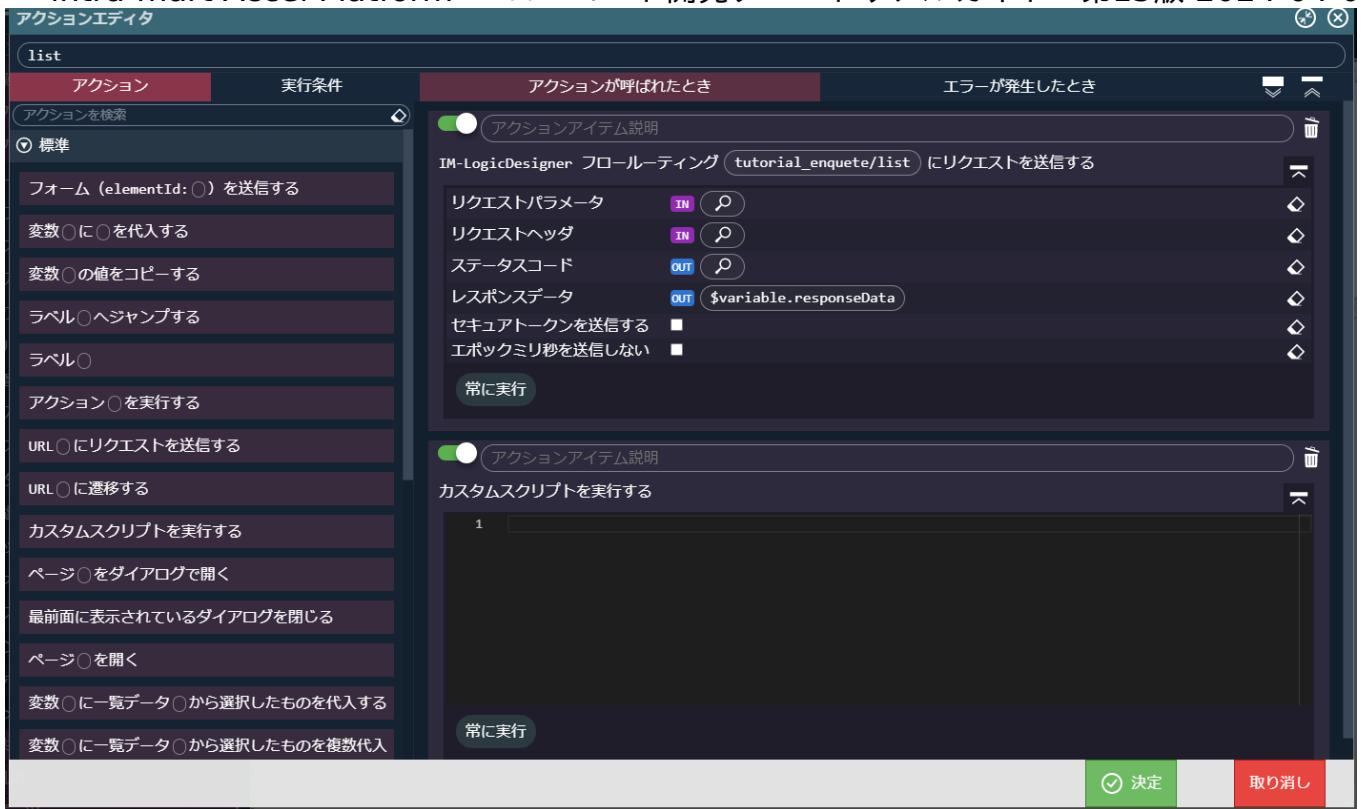


業務ロジックからデータを受け取るために、レスポンスデータの  から「変数 > \$variable > responseData」を設定します。



次に、取得したデータを表示用に加工する処理を追加します。

左側の「アクション」タブから「標準」 - 「カスタムスクリプトを実行する」を「IM-LogicDesigner フロールーティング〇にリクエストを送信する」の次に配置します。



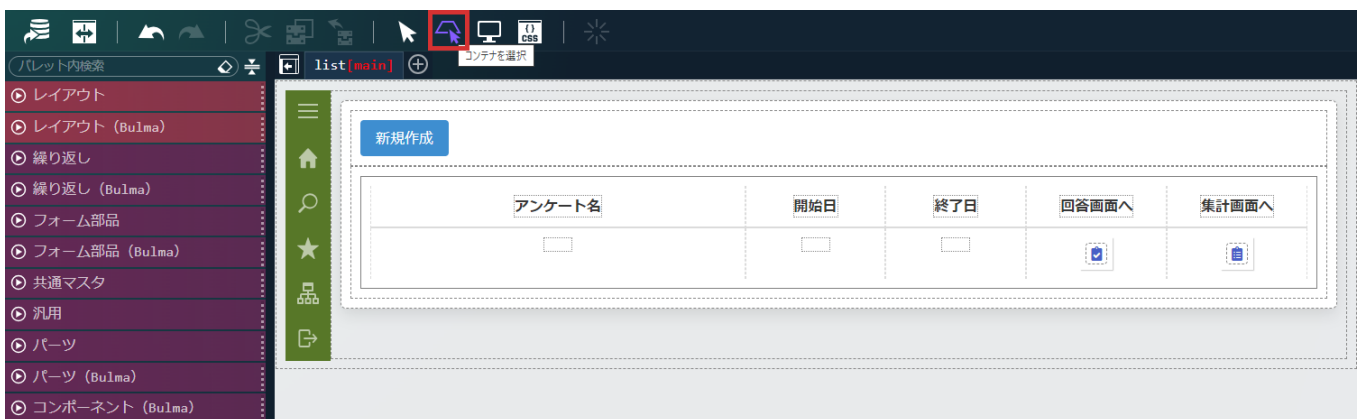
配置した「カスタムスクリプト」の入力欄に以下のスクリプトを設定し、決定ボタンをクリックして一覧取得のアクション設定が完了です。

```

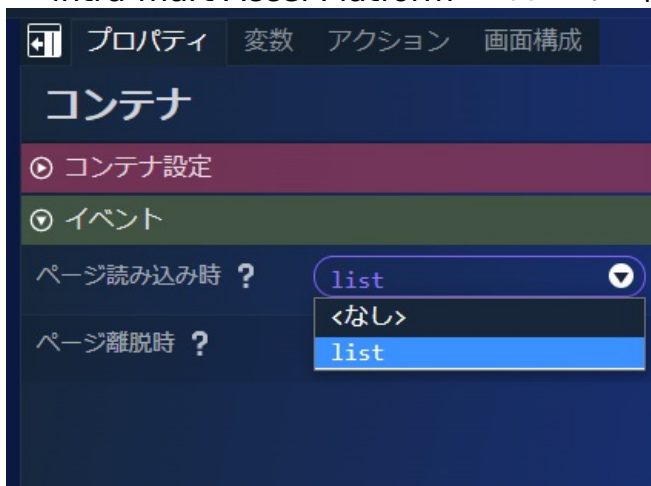
$variable.resultList = [];
for(const res of $variable.responseData.records){
  let list = {};
  list.title = res.enquete_title;
  list.startDate = res.start_date;
  list.endDate = res.end_date;
  list.answerUrl = $constant.answerUrl + res.enquete_id;
  list.totalUrl = $constant.totalUrl + res.enquete_id;
  $variable.resultList.push(list);
}
    
```

作成した「list」アクションを画面の初期表示時に呼び出す設定を行います。

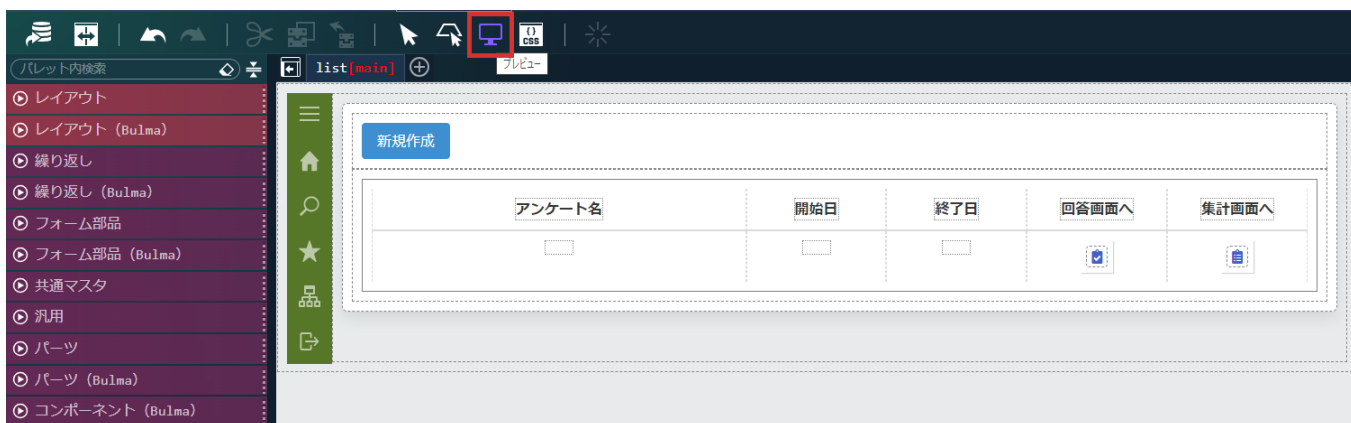
画面上部の  をクリックします。



画面右側の「プロパティ」タブ - 「コンテナ」 - 「イベント」の「ページ読み込み時」に「list」アクションを設定します。



一覧画面の動作確認を行います。画面上部の  をクリックします。




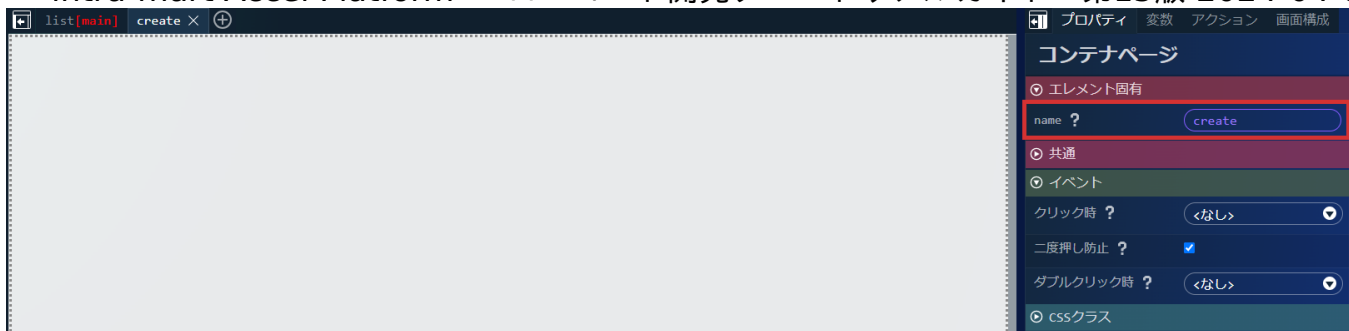
プレビューダイアログが表示され、アンケート一覧が表示されました。

アンケート名	開始日	終了日	回答画面へ	集計画面へ
従業員満足度調査	2021/01/01	2022/01/01		
好きな映画ジャンル調査	2021/01/01	2022/01/01		
好きな料理アンケート	2024/03/14	2024/03/31		
休日の過ごし方	2024/04/01	2024/04/15		
理解度チェック	2024/03/01	2024/05/31		

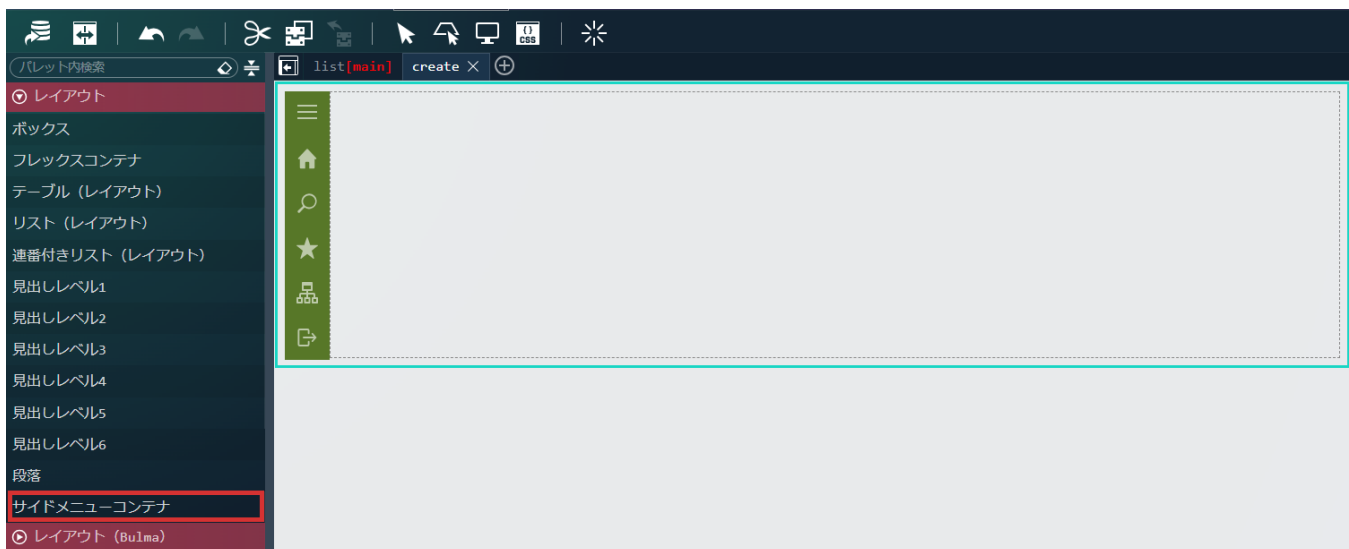
アンケート登録画面の作成

続けて、アンケートの登録画面を作成します。

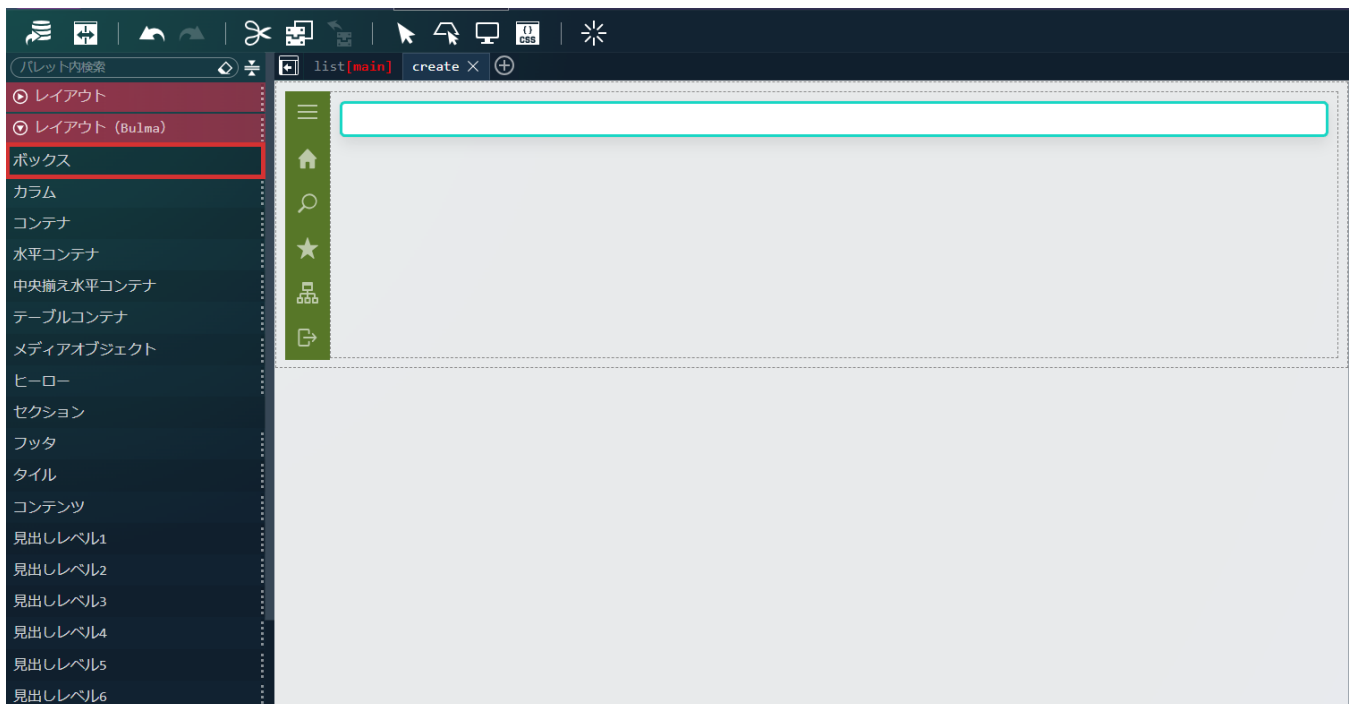
画面上部の  をクリックし、「プロパティ」タブの「コンテナページ」 - 「エレメント固有」 - 「name」に「create」と設定します。



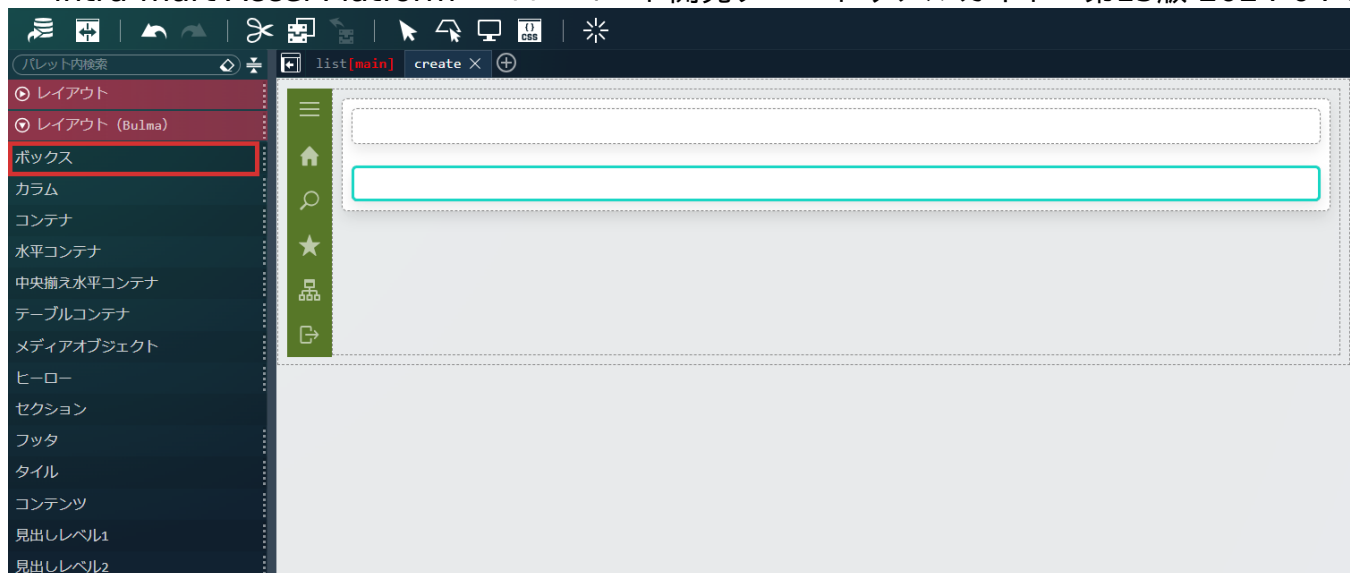
次に、画面にサイドメニューを表示するためにサイドメニューコンテナを配置します。
画面左側のパレットの「レイアウト」 - 「サイドメニューコンテナ」を選択し、画面に配置します。



次に、アンケートの登録内容の表示領域を作成します。
画面左側のパレットの「レイアウト (Bulma)」 - 「ボックス」を選択し、先ほど配置した「サイドメニューコンテナ」の中に配置します。



同様に、パレットの「レイアウト (Bulma)」 - 「ボックス」を選択し、先ほど配置した「ボックス」の中に2つ配置します。



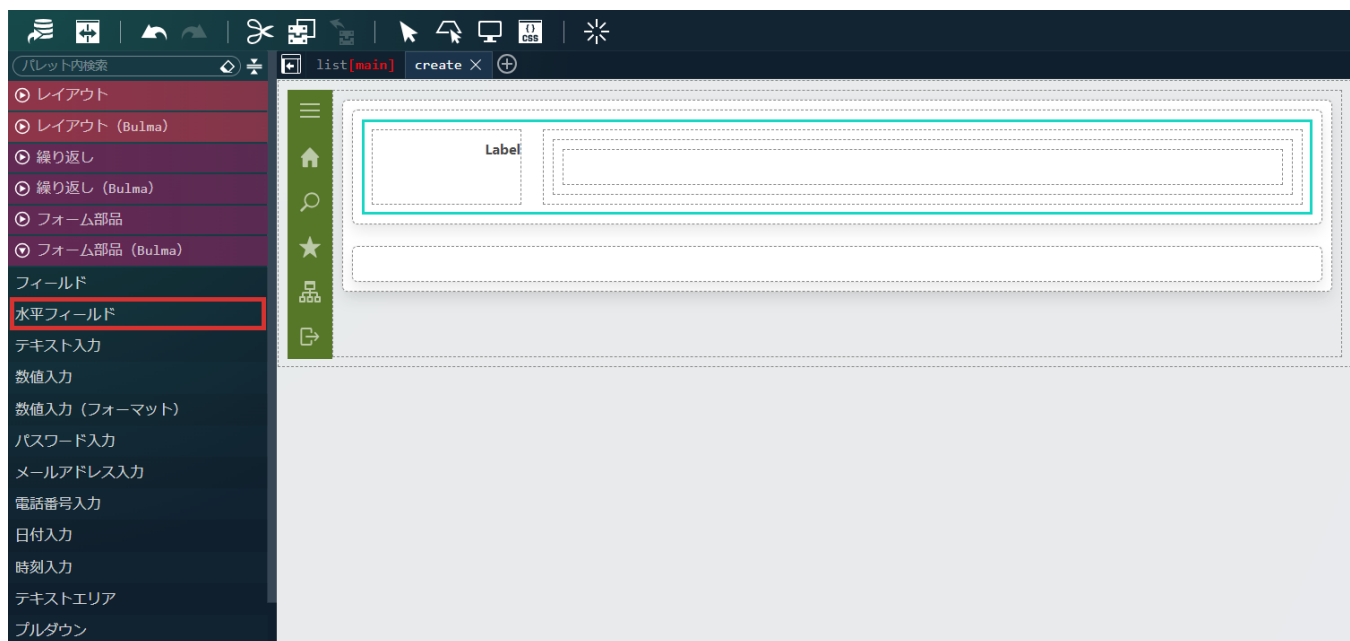
上に配置したボックスにアンケート情報の入力内容を、下に配置したボックスに設問情報の入力内容を作成していきます。

まずは、アンケート情報の入力内容を設定していきます。

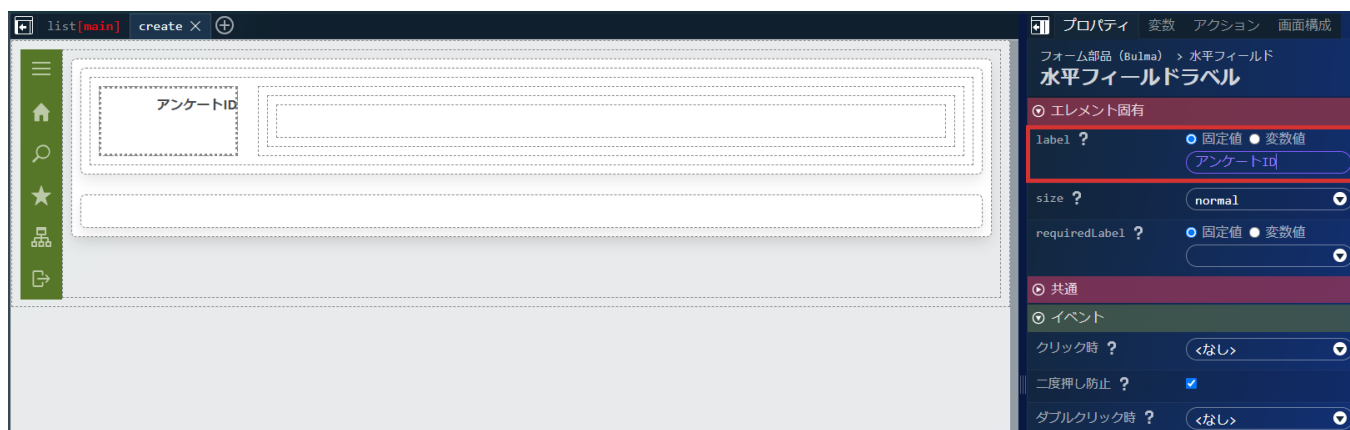
アンケート情報は「アンケートID」、「アンケートタイトル」、「アンケート概要」、「開始日」、「終了日」の入力を可能とします。

最初に、アンケートIDの入力欄を作成します。

画面左側のパレットの「フォーム部品 (Bulma)」 - 「水平フィールド」を選択し、上に配置したボックスの中に配置します。

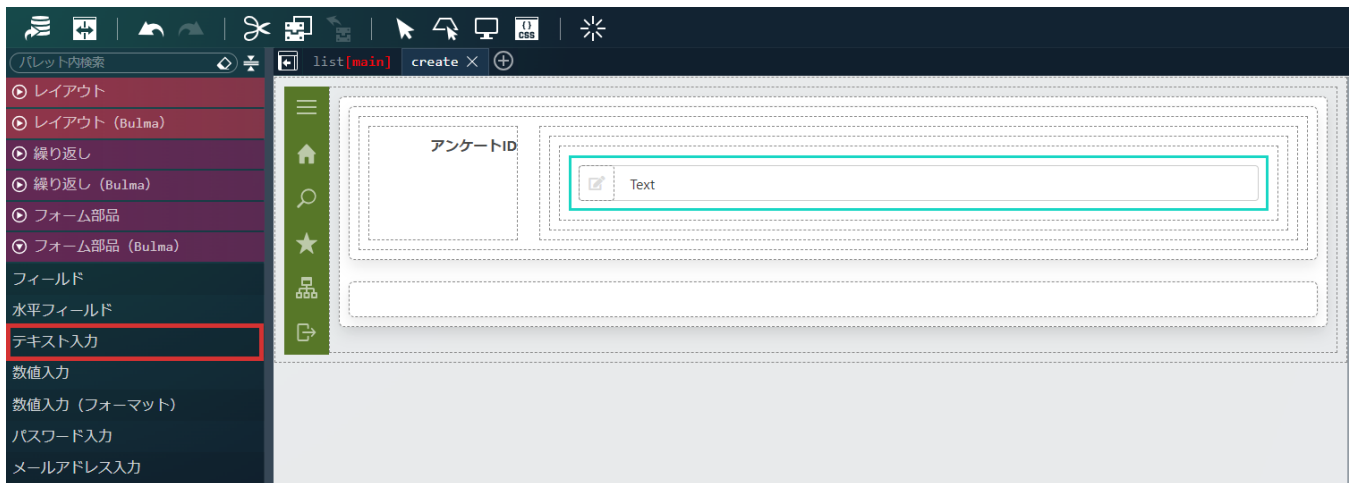



配置した「水平フィールド」の左側にある「水平フィールドラベル」を選択し、画面右側の「プロパティ」タブ - 「水平フィールドラベル」 - 「エレメント固有」 - 「label」に「アンケートID」と設定します。

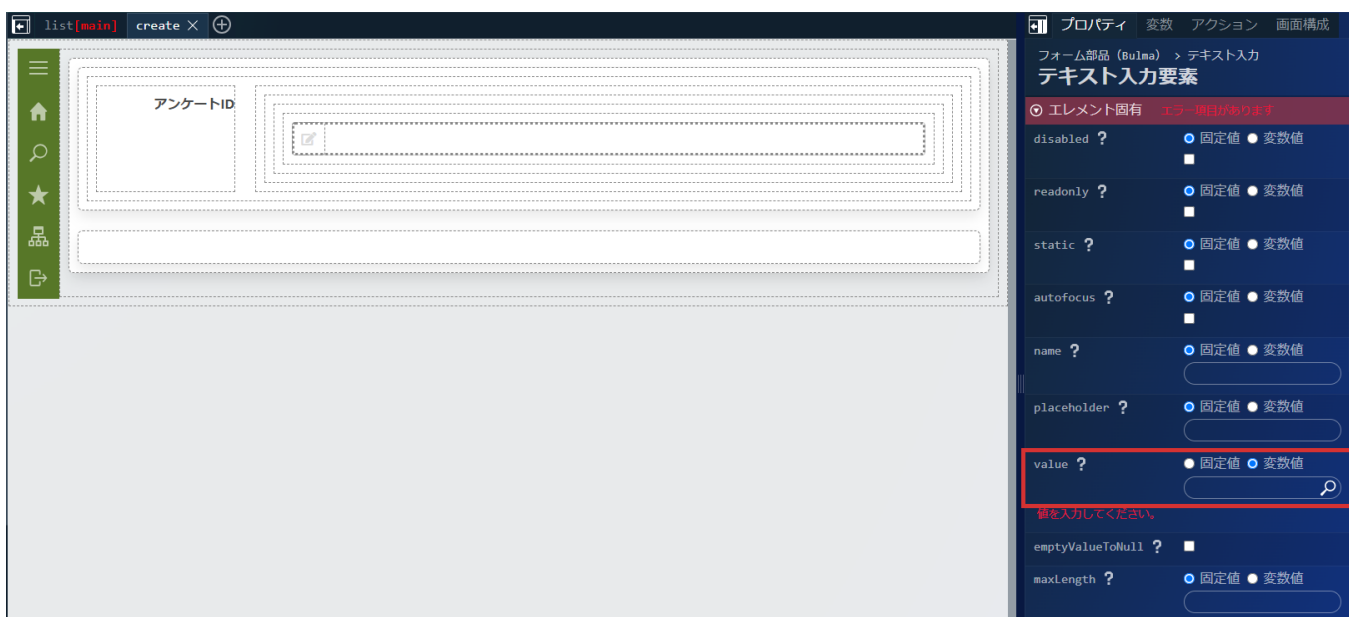


続けて、アンケートIDの入力欄用のテキストボックスを配置します。

画面左側のパレットの「フォーム部品 (Bulma)」 - 「テキスト入力」を選択し、「水平フィールド」の右側に配置します。



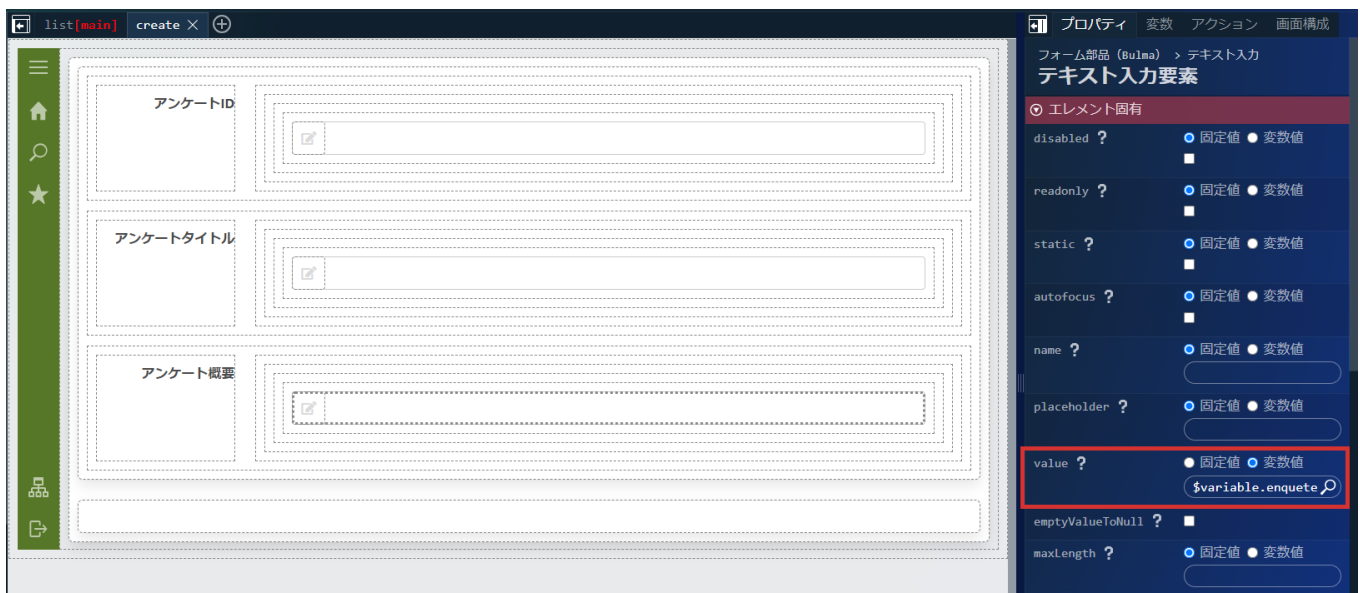
配置した「テキスト入力」の「テキスト入力要素」を選択し、「プロパティ」タブ - 「エレメント固有」 - 「value」を「変数値」にし、 を選択します。



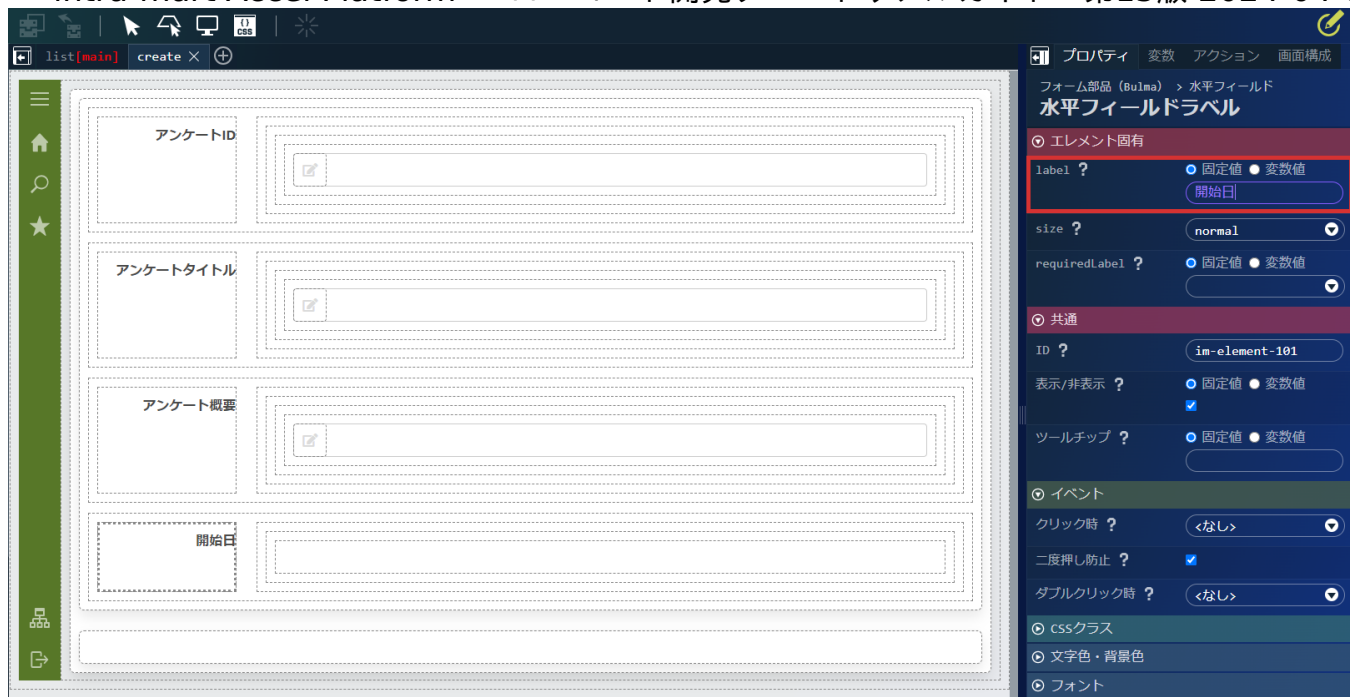
「\$variable > enquetelInfo > enquetelId」を選択し「決定」ボタンをクリックして変数を設定します。



同様の手順で、「アンケートタイトル」、「アンケート概要」の入力欄を作成します。
 テキスト入力要素の「プロパティ」タブ - 「エレメント固有」 - 「value」にはそれぞれ、「アンケートタイトル」は「\$variable.enqueteInfo.enqueteTitle」、「アンケート概要」は「\$variable.enqueteInfo.enqueteOverview」を設定してください。

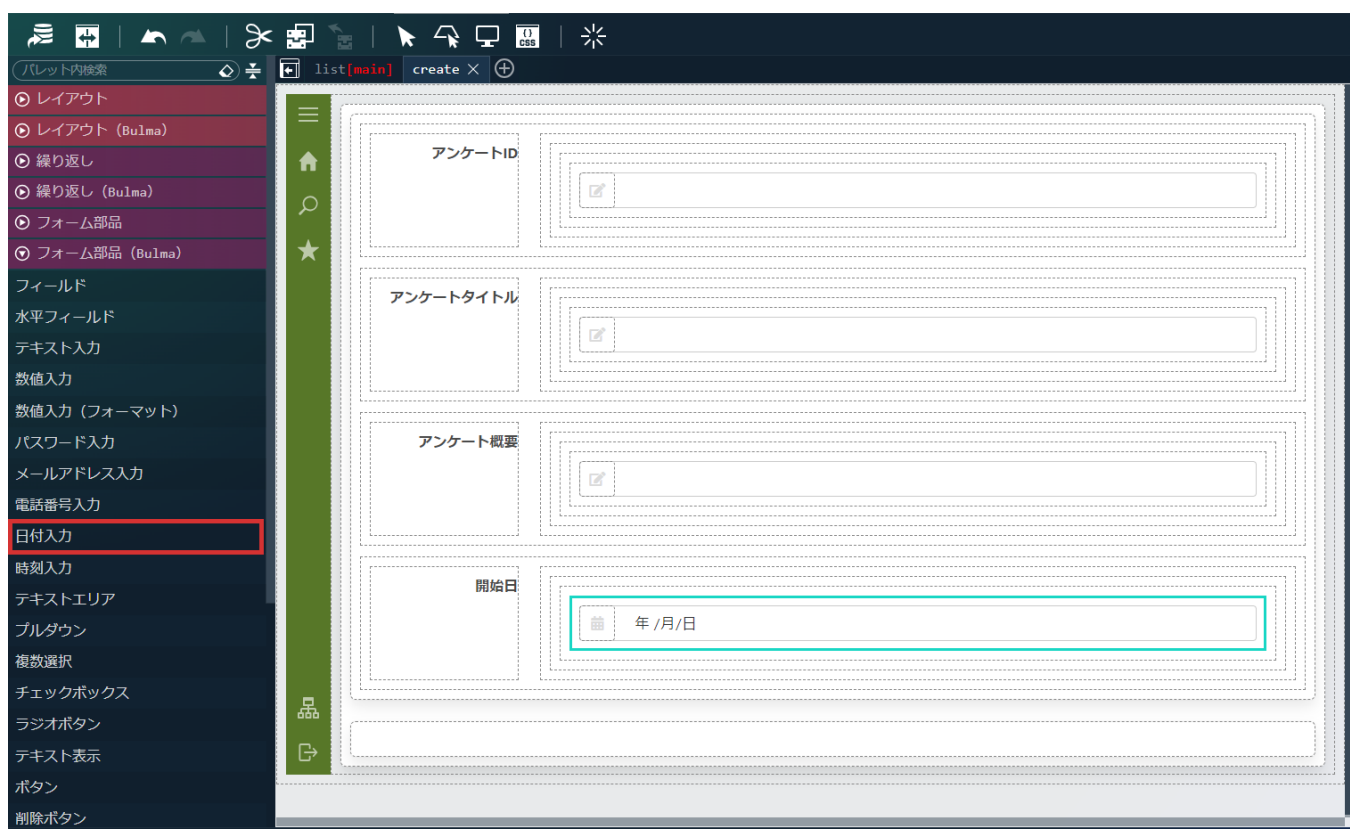


続けて、「開始日」と「終了日」の入力欄を作成します。
 画面左側のパレットの「フォーム部品 (Bulma)」 - 「水平フィールド」を選択し、「アンケート概要」の入力欄の下に配置します。
 配置した「水平フィールド」の左側にある「水平フィールドラベル」を選択し、画面右側の「プロパティ」タブ - 「水平フィールドラベル」 - 「エレメント固有」 - 「label」に「開始日」と設定します。



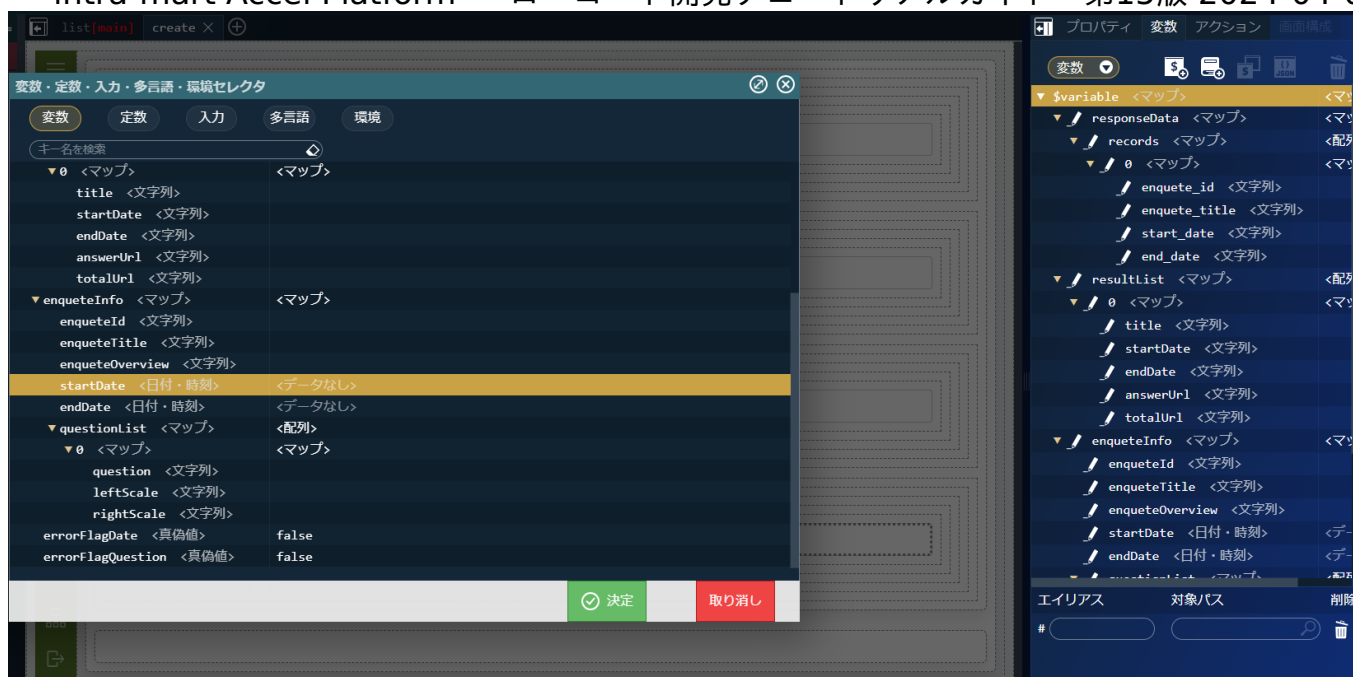
続けて、アンケートIDの入力欄用のテキストボックスを配置します。

画面左側のパレットの「フォーム部品 (Bulma)」 - 「日付入力」を選択し、「水平フィールド」の右側に配置します。



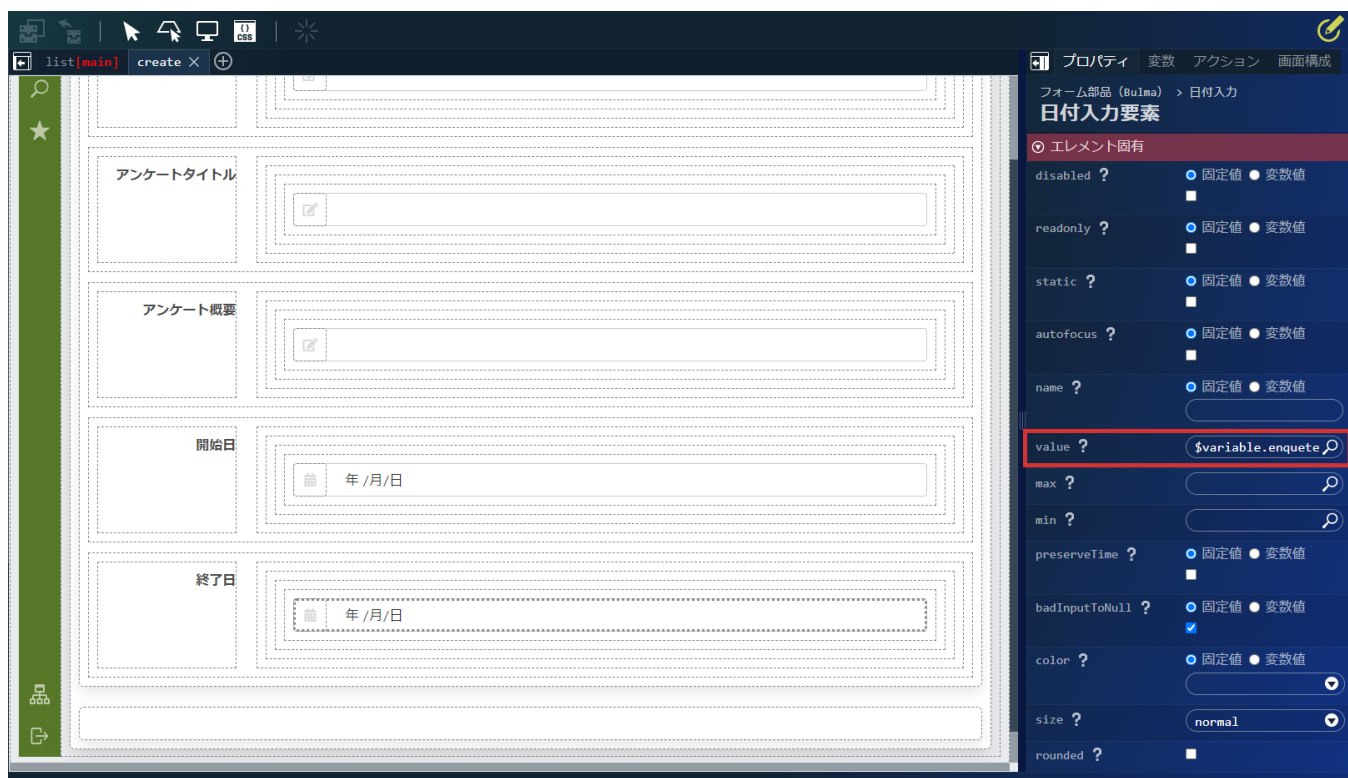
配置した「日付入力」の「日付入力要素」を選択し、「プロパティ」タブ - 「エレメント固有」 - 「value」の 🔍 を選択します。

「\$variable > enquetelInfo > startDate」を選択し「決定」ボタンをクリックして変数を設定します。



同様の手順で、「終了日」の入力欄を作成します。

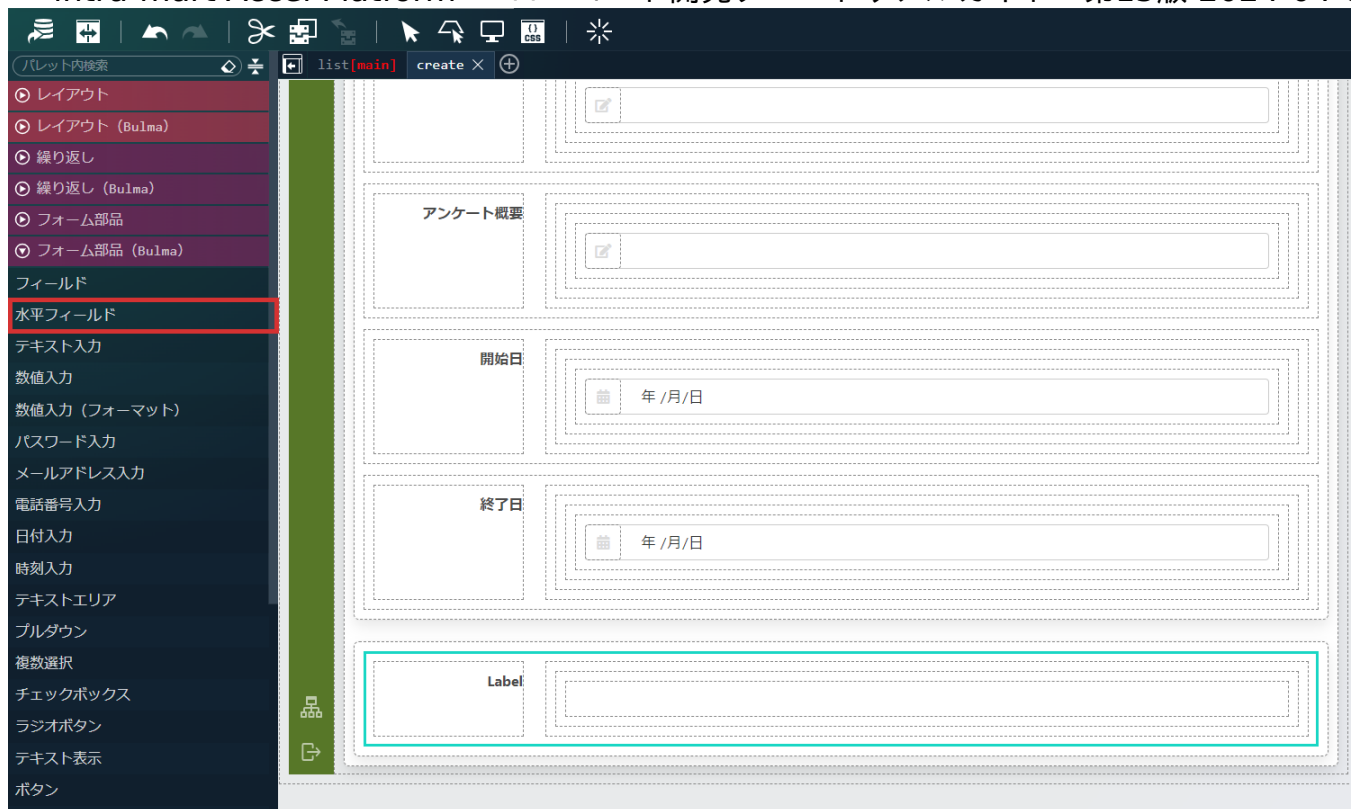
日付入力要素の「プロパティ」タブ - 「エレメント固有」 - 「value」には「\$variable.enqueteInfo.endDate」を設定してください。



次は、設問情報の入力欄を作成します。

設問情報は、設問数の増減を動的に行えるようにします。

画面左側のパレットの「フォーム部品 (Bulma)」 - 「水平フィールド」を選択し、下側の設問情報入力用の「ボックス」の中に配置します。



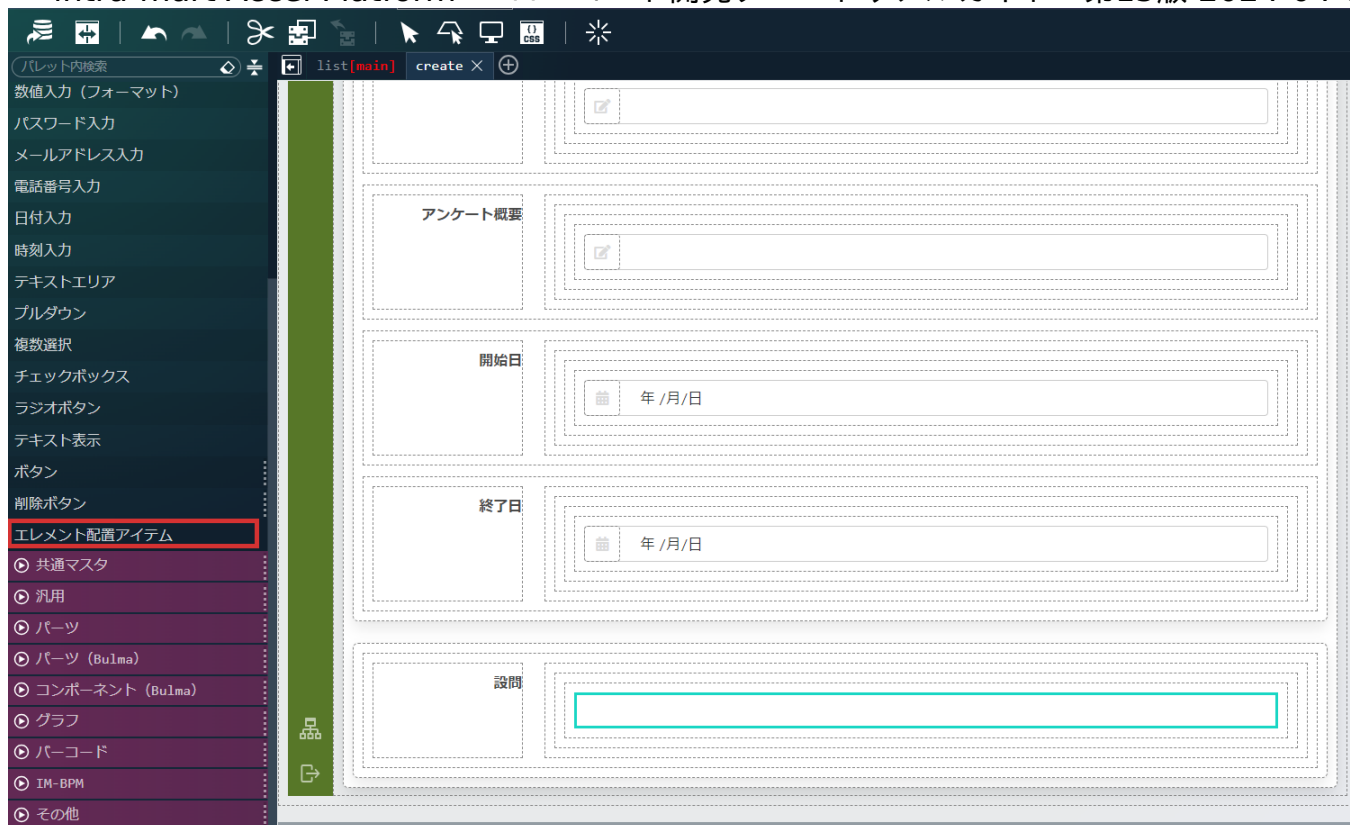
配置した「水平フィールド」の左側にある「水平フィールドラベル」を選択します。

画面右側の「プロパティ」タブ - 「水平フィールドラベル」 - 「エレメント固有」 - 「label」に「設問」と設定します。

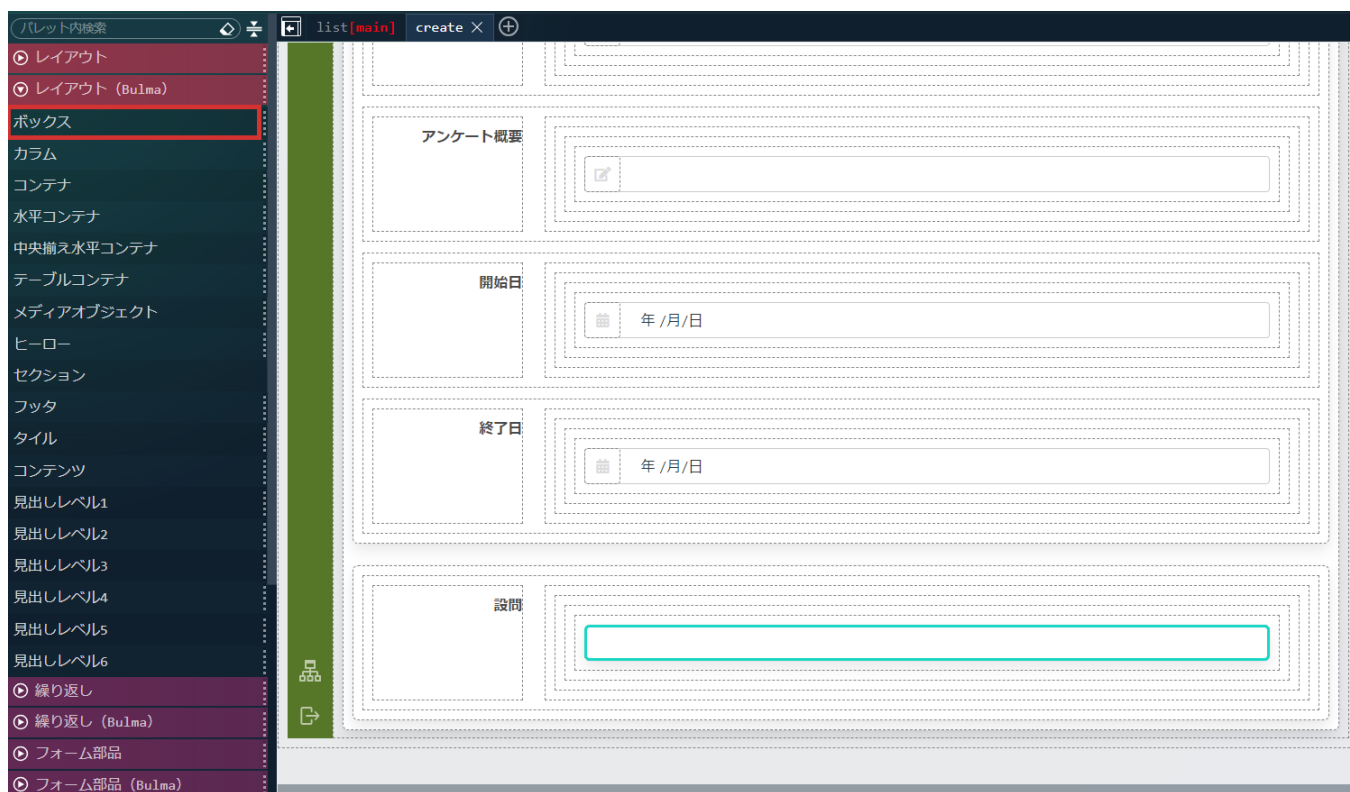


続けて、設問内容の入力欄を作成します。

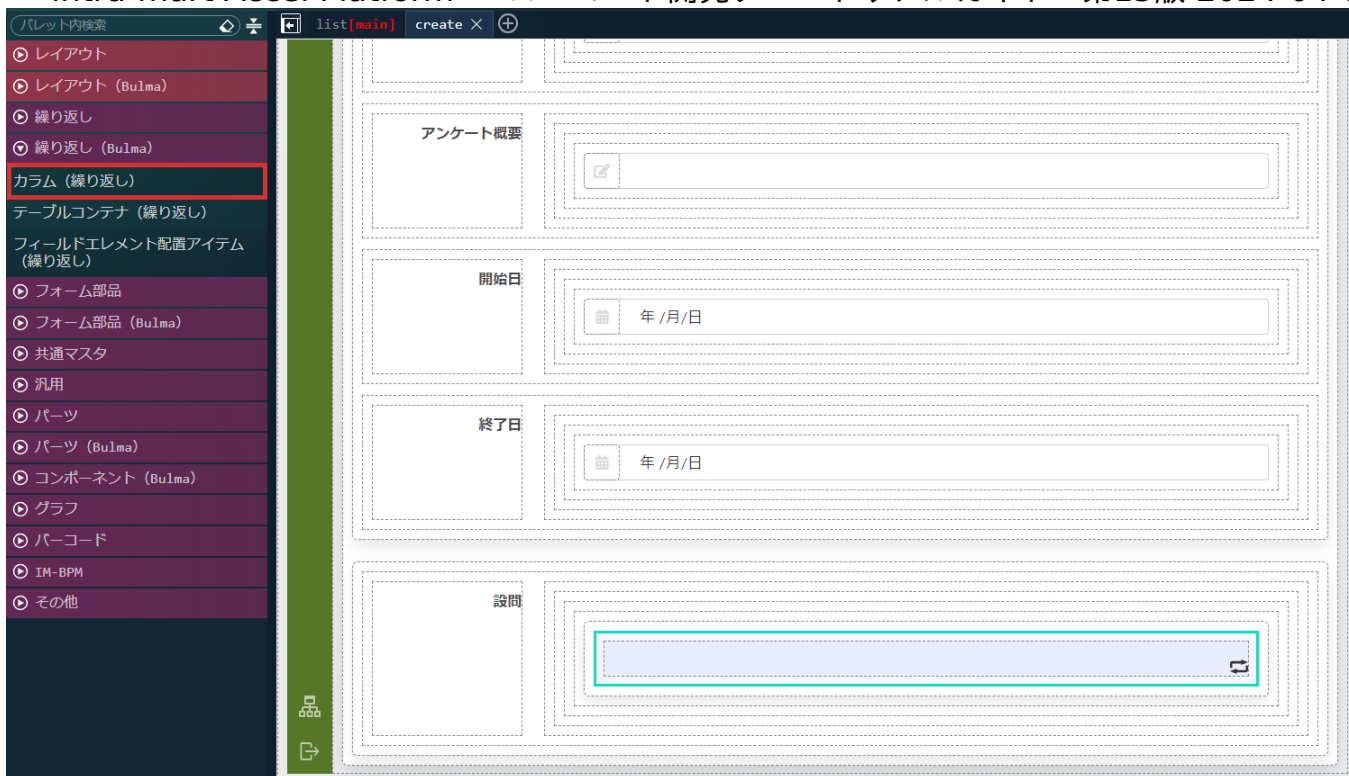
画面左側のパレットの「フォーム部品 (Bulma)」 - 「エレメント配置アイテム」を選択し、「水平フィールド」の右側に配置します。



次に、「レイアウト (Bulma)」 - 「ボックス」を選択し、先ほど配置した「エレメント配置アイテム」の中に配置します。

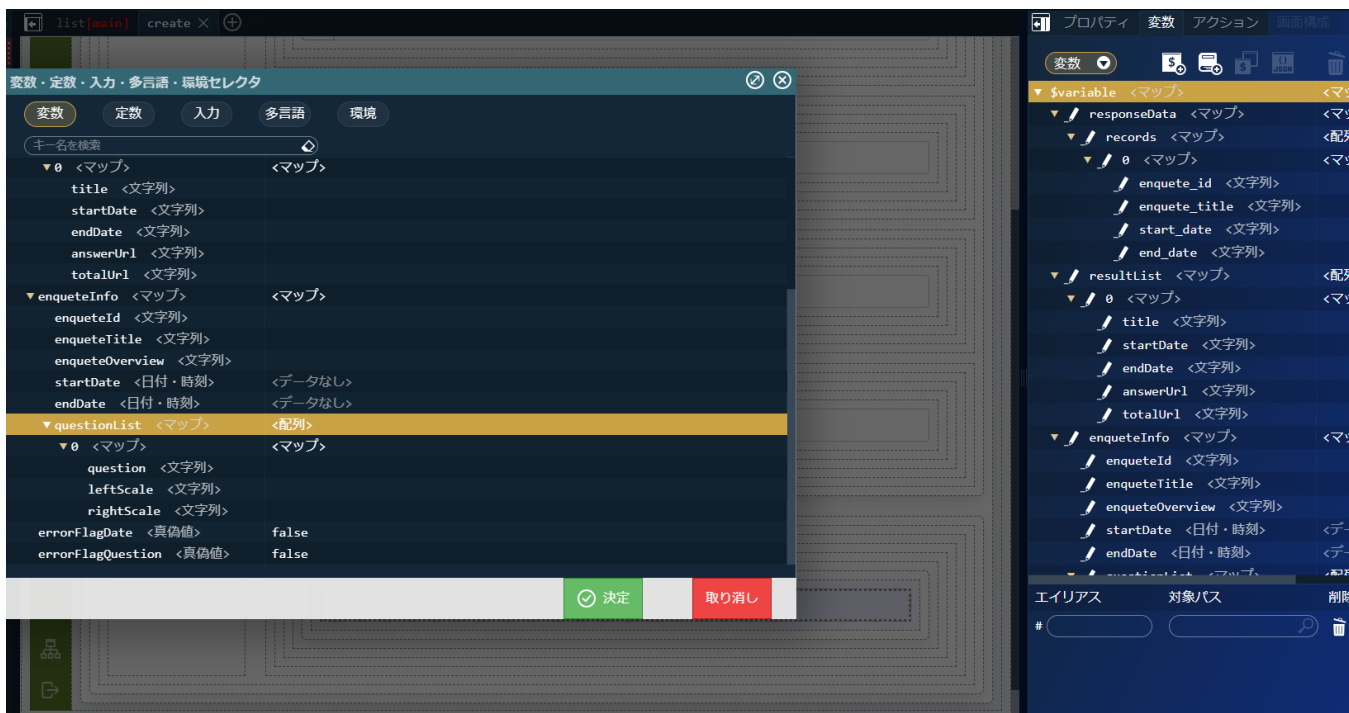


次に、「繰り返し (Bulma)」 - 「カラム (繰り返し)」を選択し、先ほど配置した「ボックス」の中に配置します。



配置した「カラム (繰り返し)」の「エレメント固有」 - 「multiLine」にチェックを入れます。

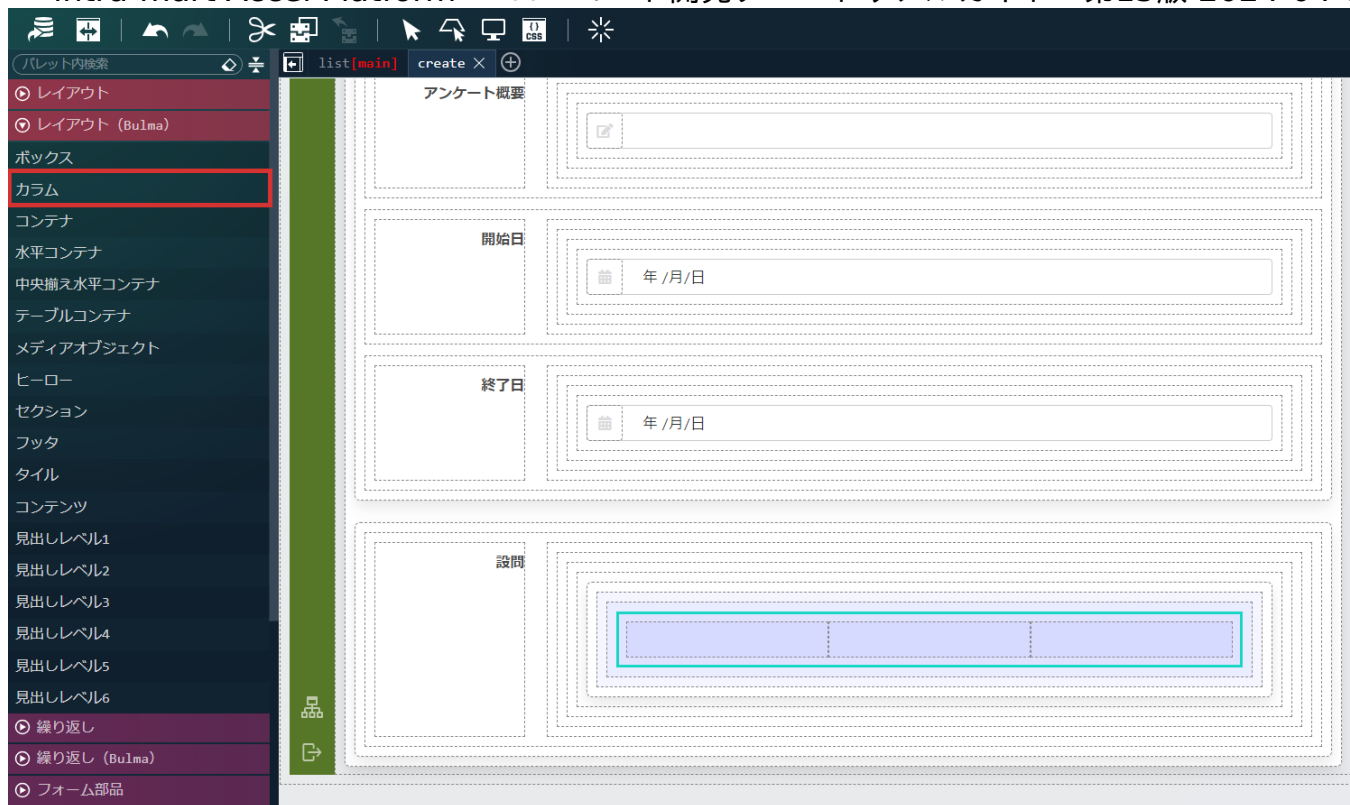
続けて、「カラム (繰り返し)」の「カラムアイテム (繰り返し)」を選択し、「エレメント固有」 - 「list」の 🔍 から「\$variable > enqueteInfo > questionList」を選択して設定します。



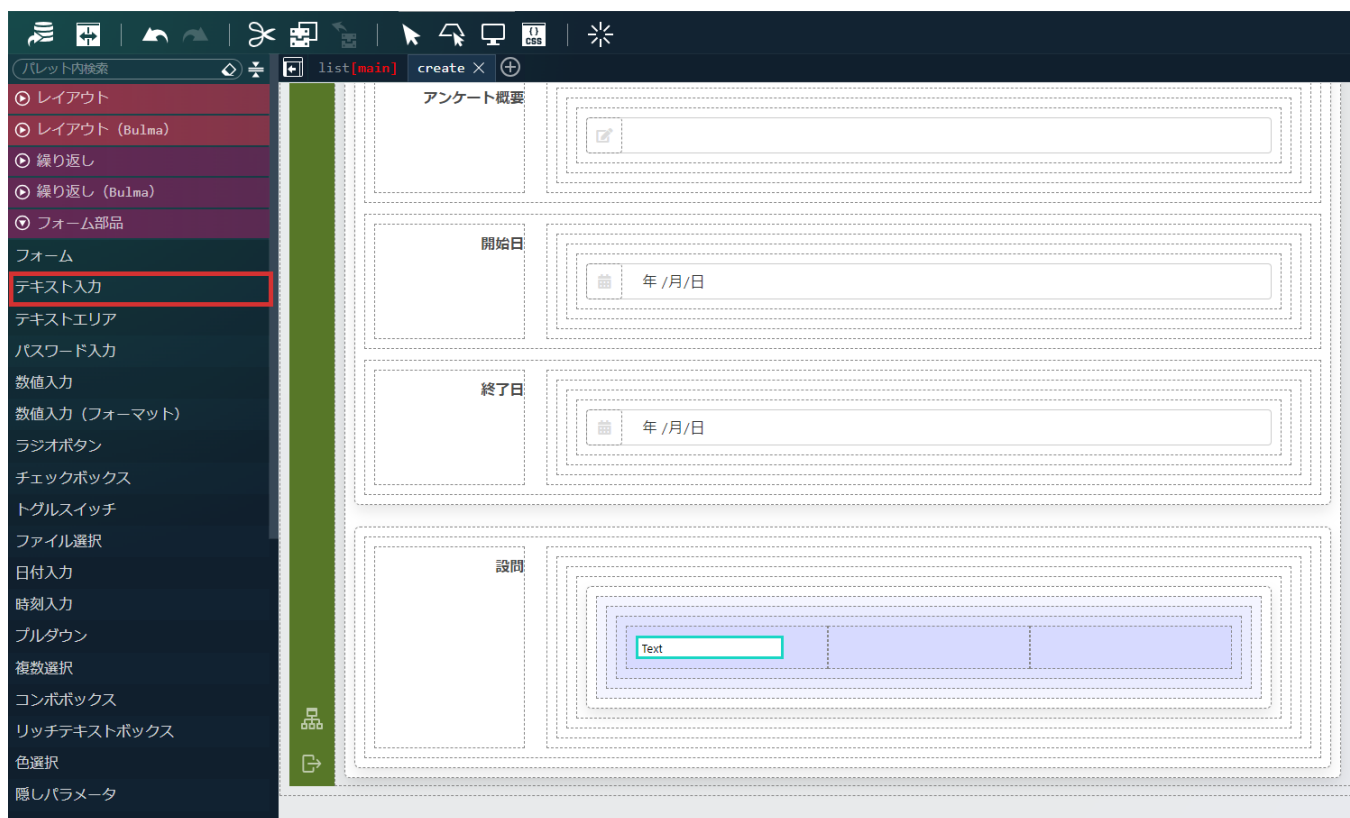
「カラム (繰り返し)」の「カラムアイテム (繰り返し)」の、「エレメント固有」 - 「size」を「full」に設定します。これで、「\$variable > enqueteInfo > questionList」の配列の長さによって、カラムが追加されるようになりました。

続けて、「設問内容」、「左項目」、「右項目」の入力欄を作成します。

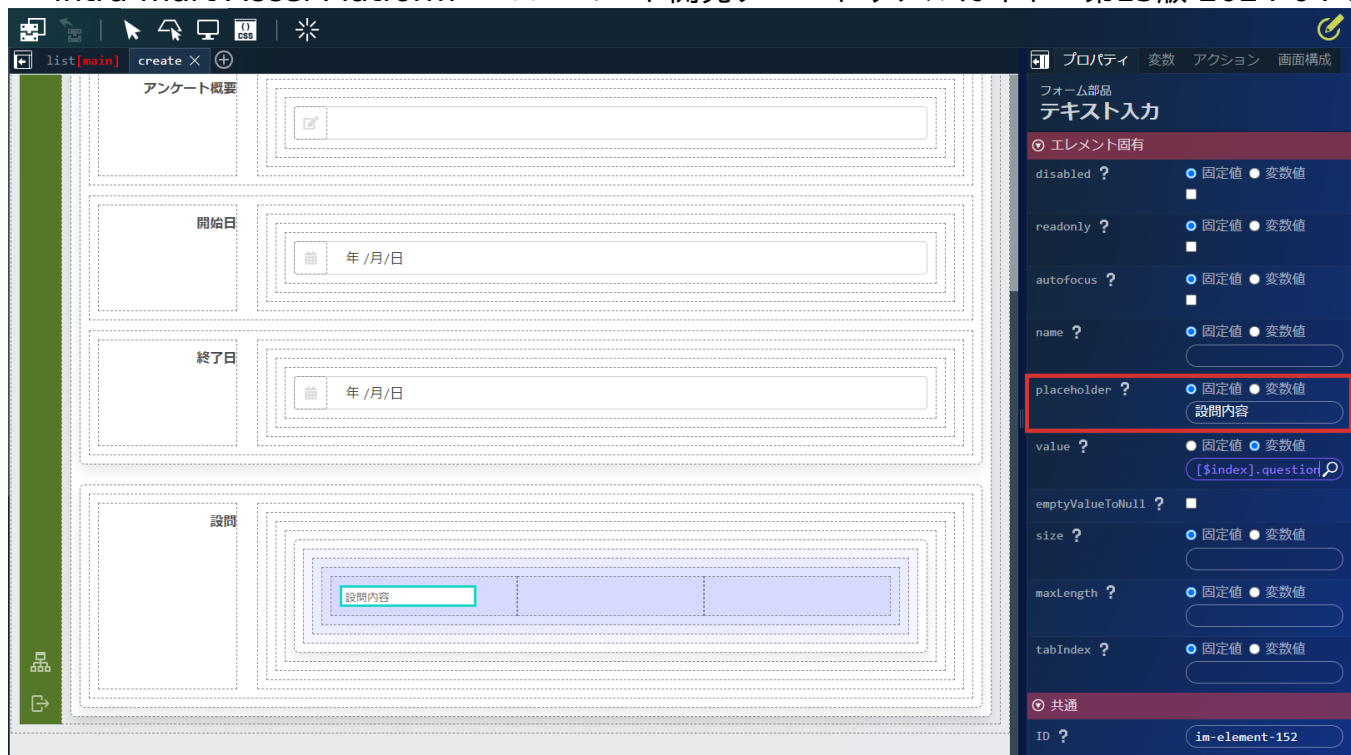
画面左側のパレットの「レイアウト (Bulma)」 - 「カラム」を選択し、「カラムアイテム (繰り返し)」の中に配置します。



画面左側のパレットの「フォーム部品」 - 「テキスト入力」を選択し、先ほど配置した「カラム」の左側に配置します。



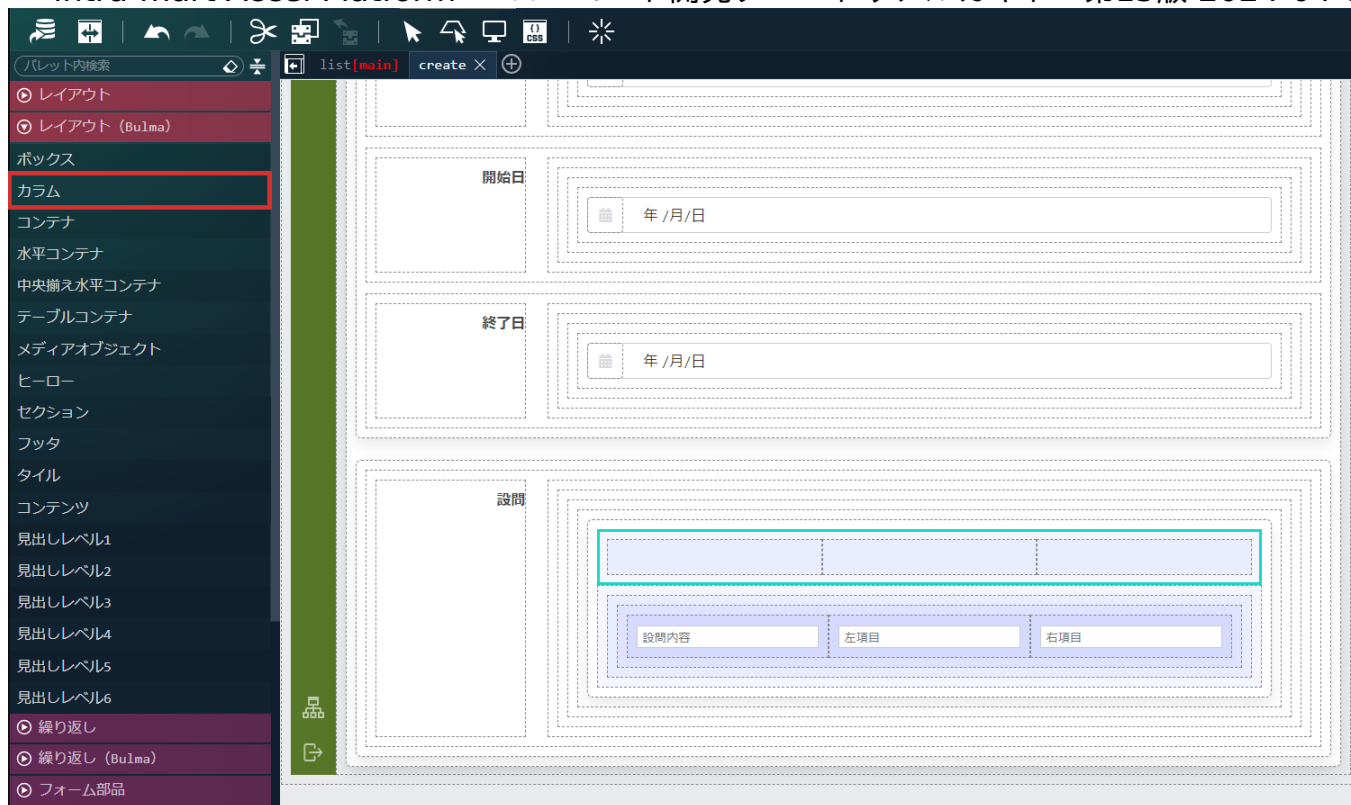
配置した「テキスト入力」を選択し、「プロパティ」タブの「エレメント固有」 - 「placeholder」に「設問内容」、「エレメント固有」 - 「value」を「変数値」にし「`$variable.enqueteInfo.questionList[$index].question`」を設定します。



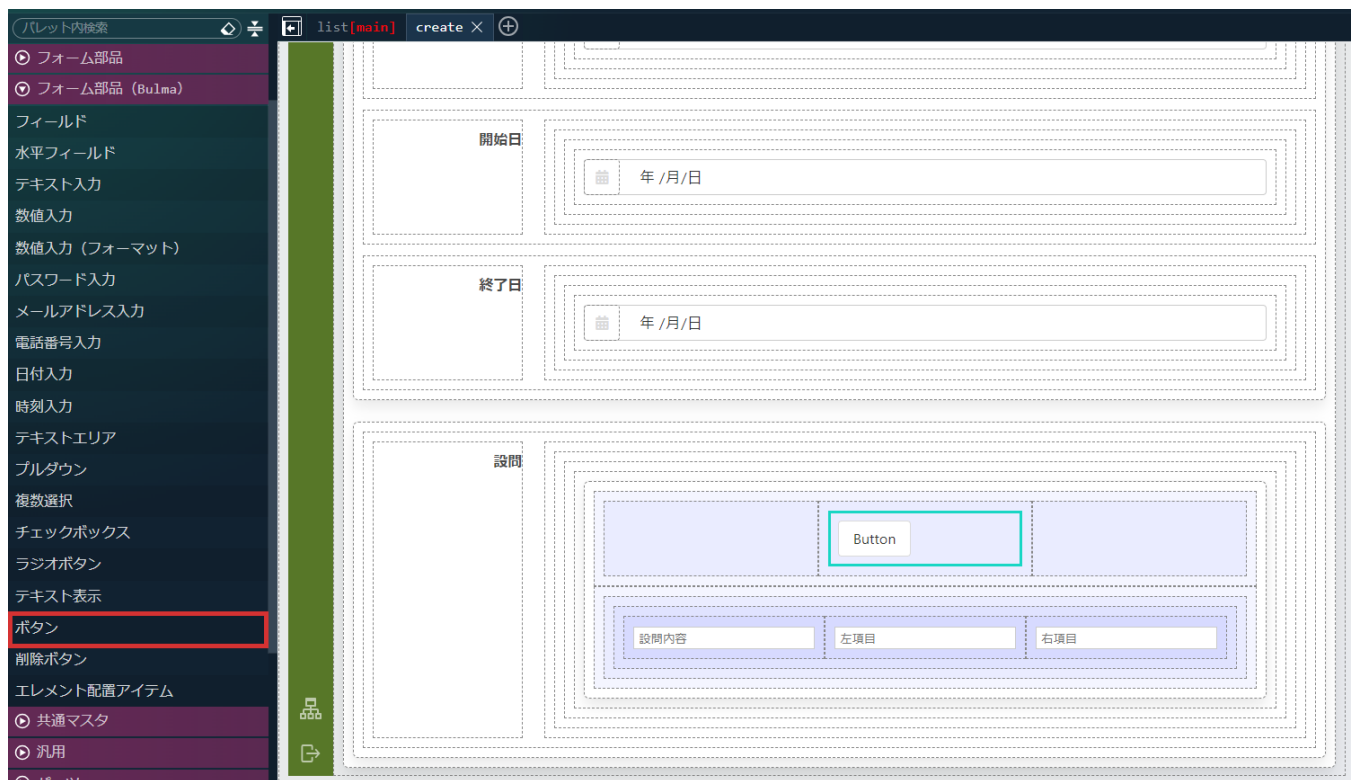
このままでは入力欄のサイズが少し小さいため、横幅の調整を行います。
配置した「テキスト入力」の「幅・高さ」 - 「横幅」に「100%」を指定します。



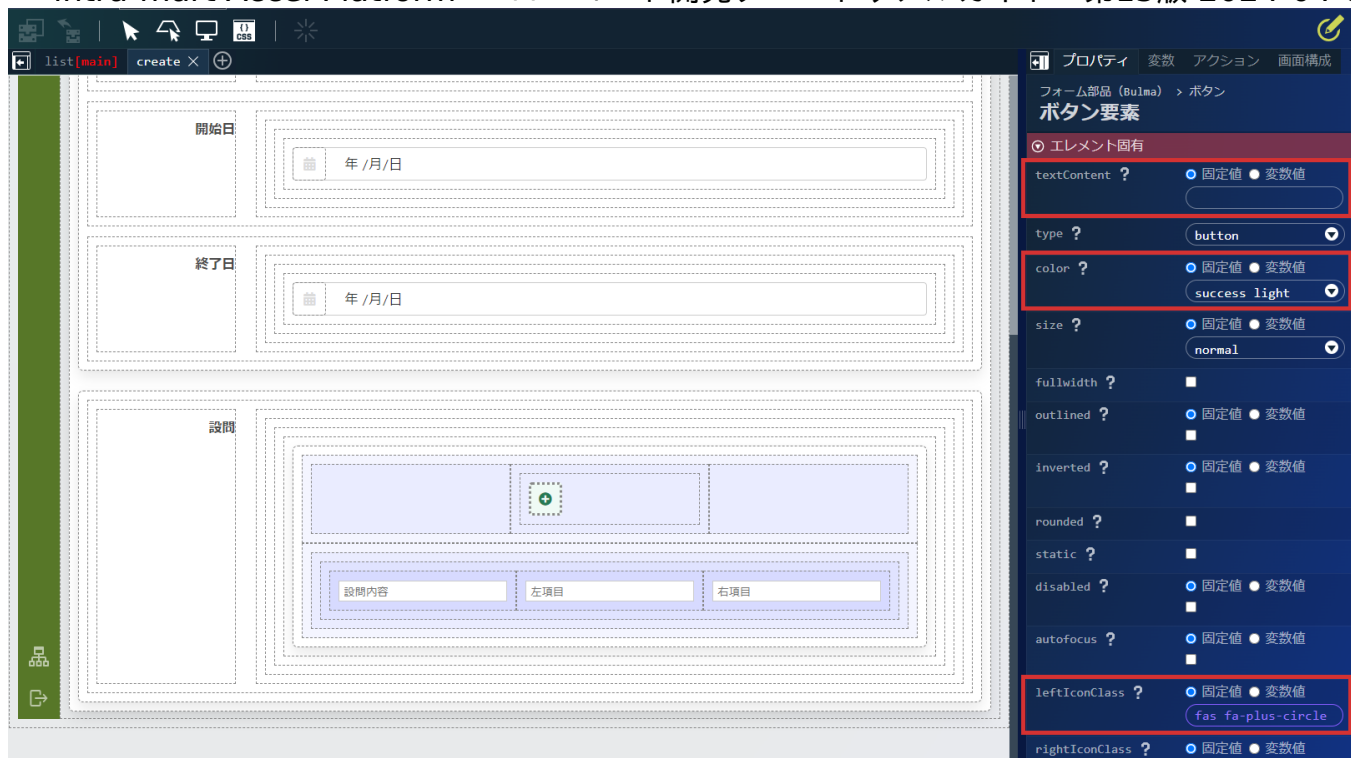
同様の手順で「左項目」、「右項目」の入力欄を作成します。
「テキスト入力」の「エレメント固有」 - 「value」にはそれぞれ、「左項目」には
「\$variable.enqueteInfo.questionList[\$index].leftScale」を「右項目」には
「\$variable.enqueteInfo.questionList[\$index].rightScale」を設定してください。
次に、設問の行の増減を行うボタンを用意します。まずは行追加ボタンを作成します。
画面左側のパレットの「レイアウト (Bulma)」 - 「カラム」を選択し、設問内容の入力欄を配置したボックスの上に配置します。



画面左側のパレットの「フォーム部品 (Bulma)」 - 「ボタン」を選択し、先ほど配置した「カラム」の中心のカラムに配置します。



配置した「ボタン」の「ボタン要素」を選択し、「プロパティ」タブの「エレメント固有」 - 「textContent」を空にします。同じく、「エレメント固有」 - 「color」を「success light」に、「leftIconClass」に「fas fa-plus-circle」をそれぞれ設定します。

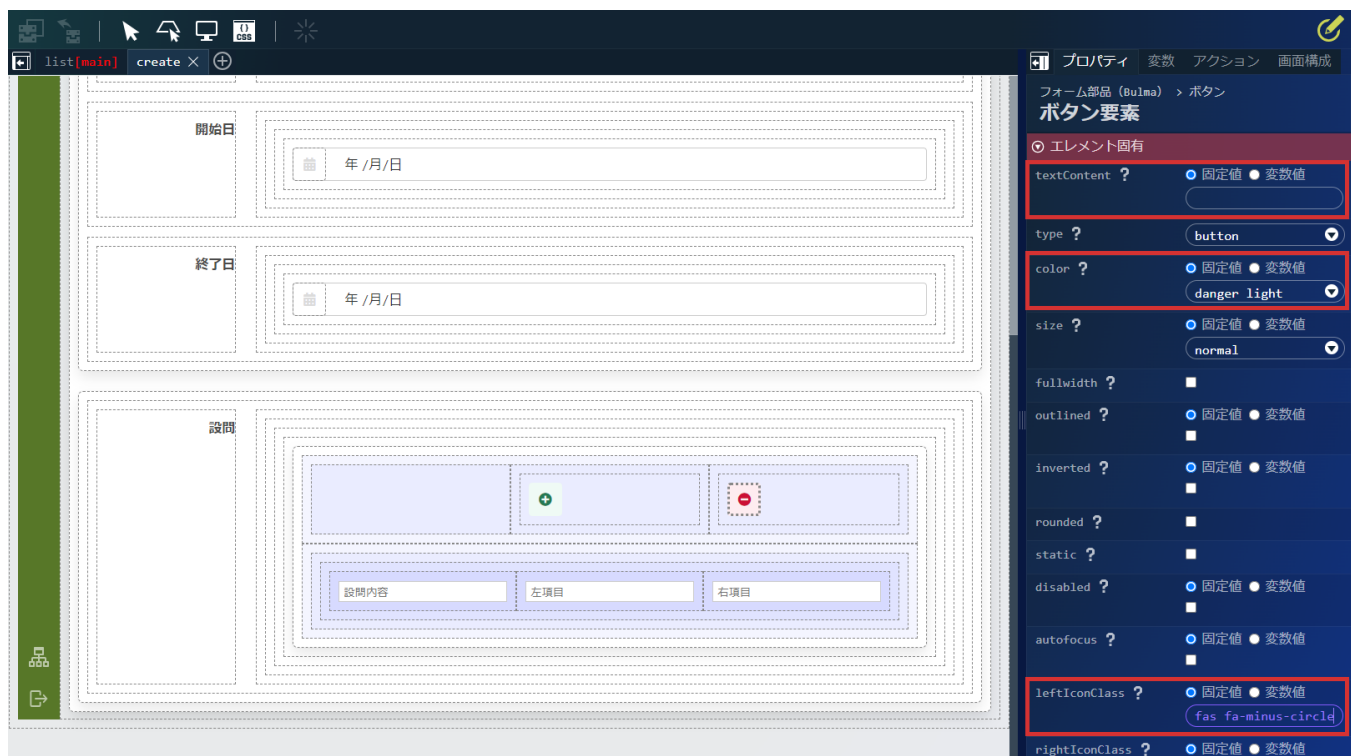


同様に削除ボタンを作成します。

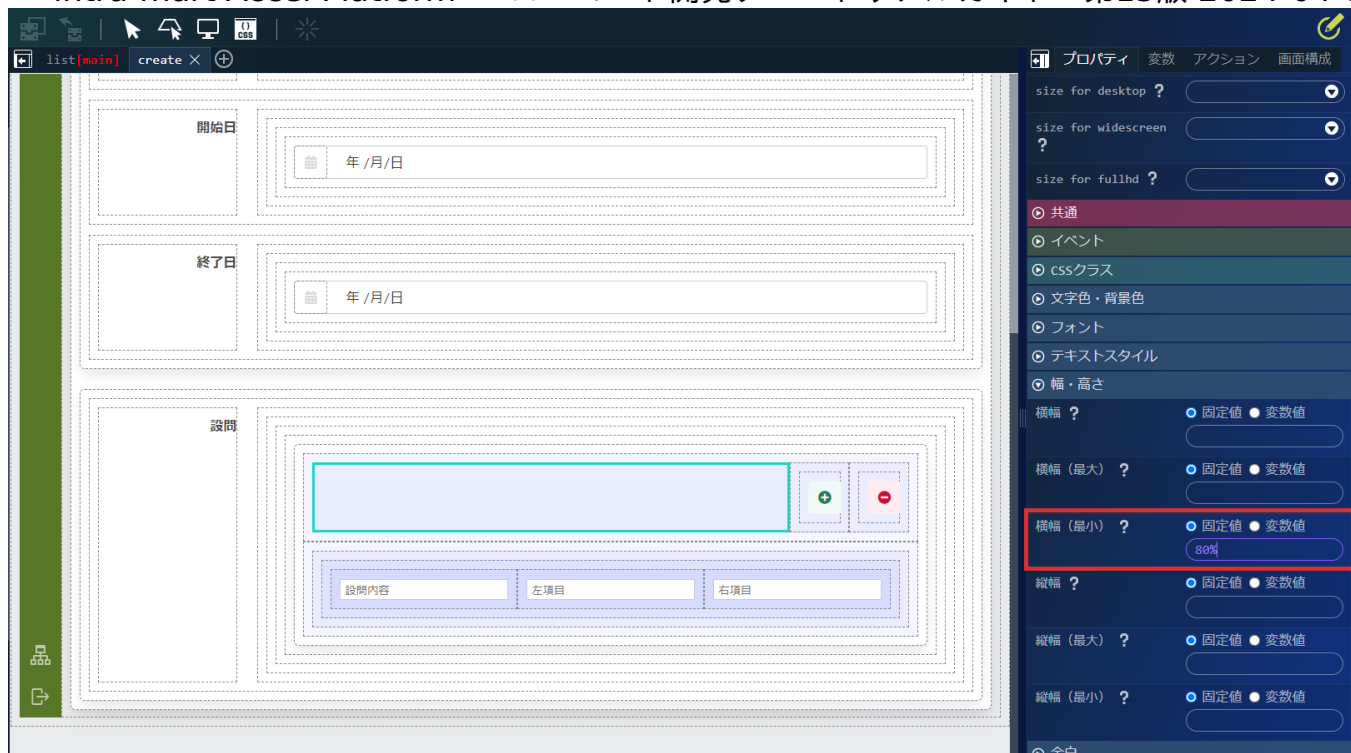
画面左側のパレットの「フォーム部品 (Bulma)」 - 「ボタン」を選択し、右端のカラムに配置します。

配置した「ボタン」の「ボタン要素」を選択し、「プロパティ」タブの「エレメント固有」 - 「textContent」を空にします。

同じく、「エレメント固有」 - 「color」を「danger light」に、「leftIconClass」に「fas fa-minus-circle」をそれぞれ設定します。

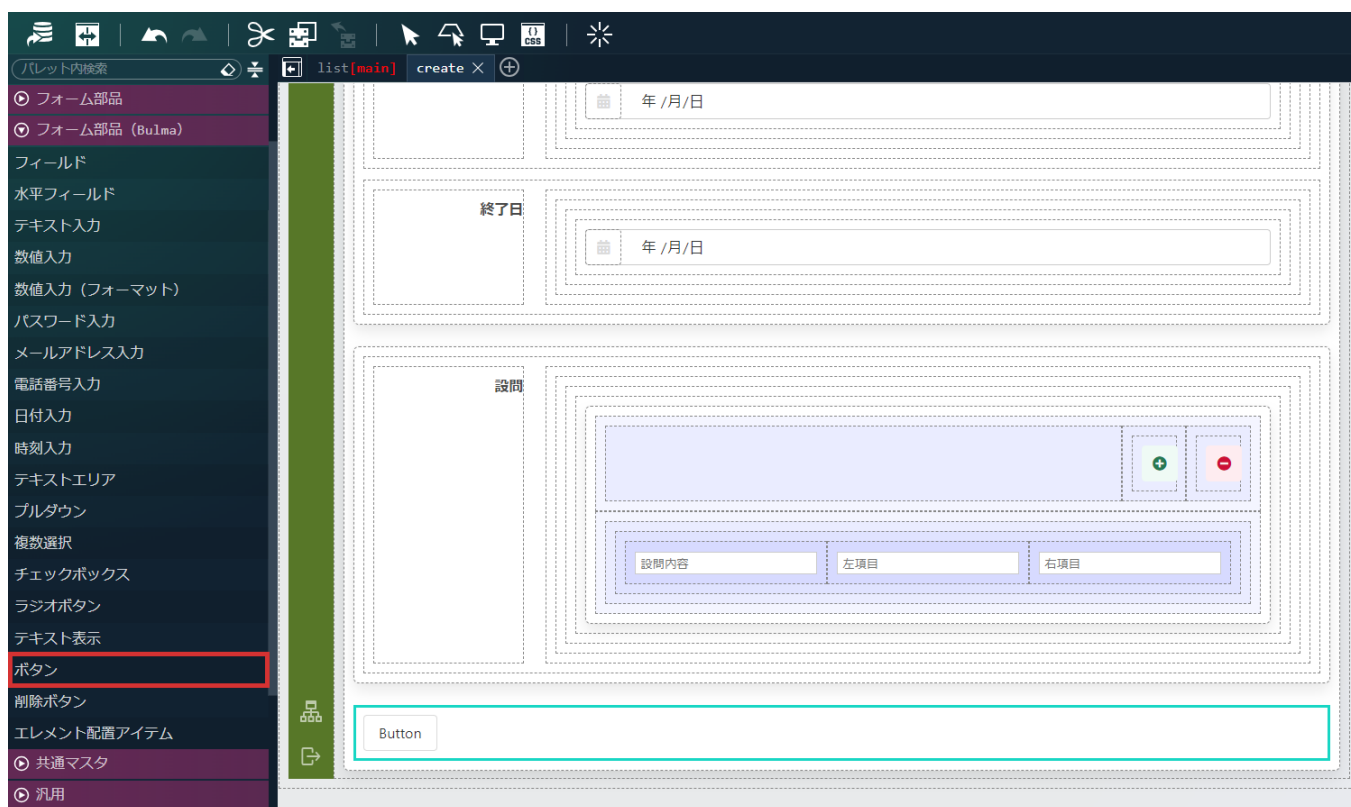


表示位置を調整するため、左端のカラムを選択し、「幅・高さ」 - 「横幅 (最小)」に「80%」を指定します。

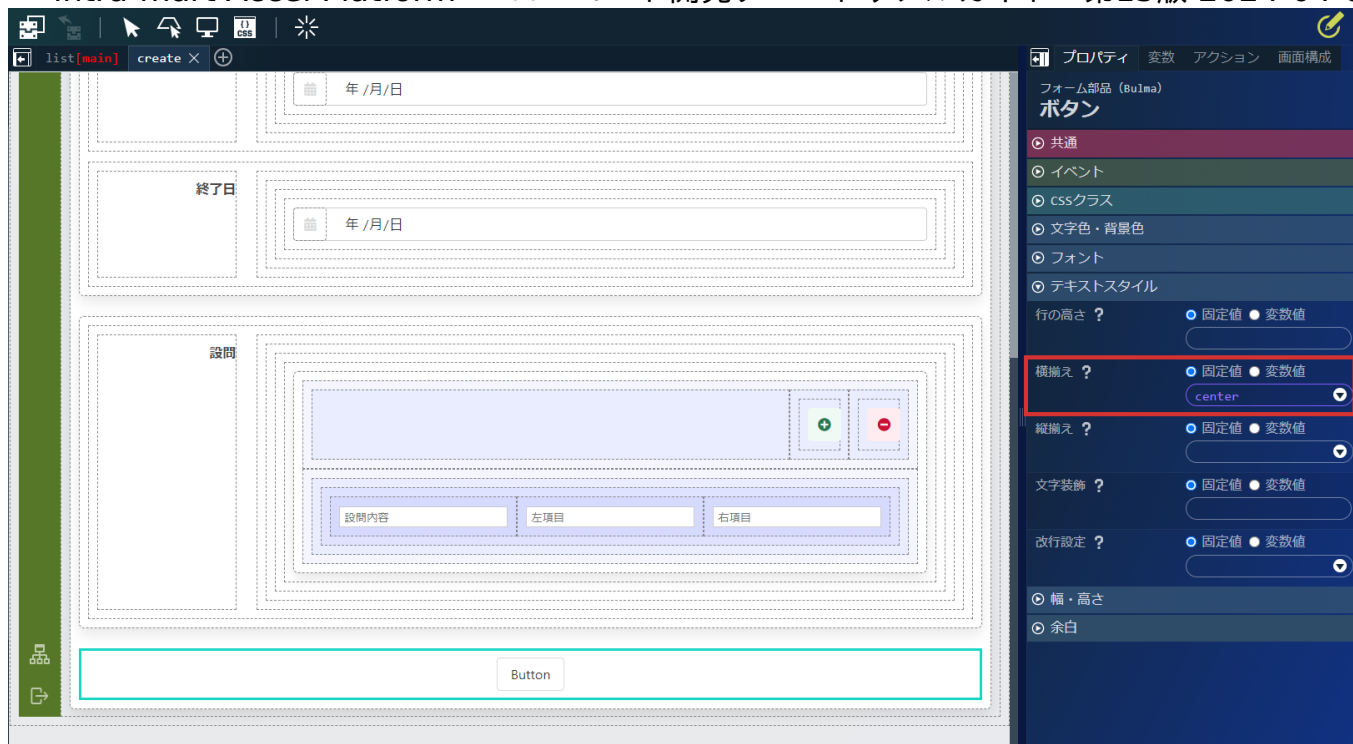


最後に、アンケートの登録ボタンを配置します。

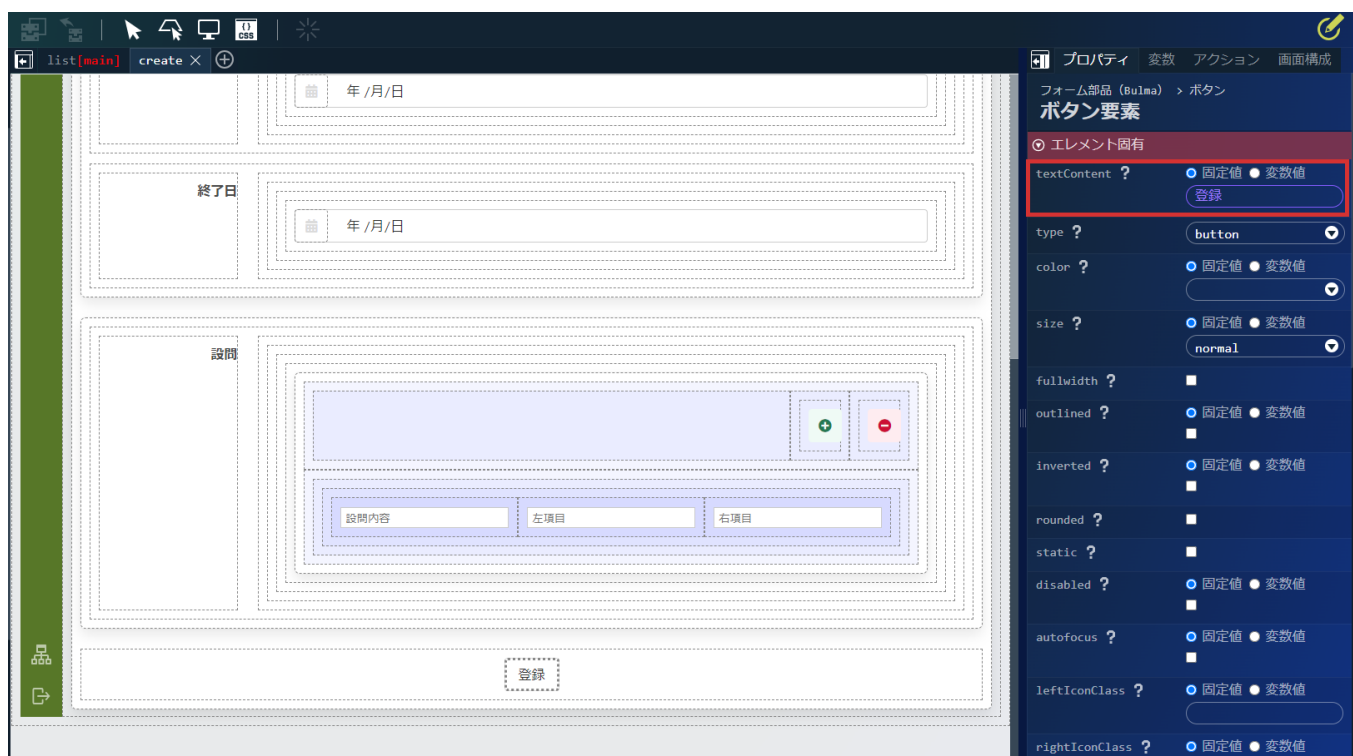
画面左側のパレットの「フォーム部品 (Bulma)」 - 「ボタン」を選択し、設問内容の入力欄を配置していた「ボックス」の一番下に配置します。



配置した「ボタン」を選択し、「プロパティ」タブの「テキストスタイル」 - 「横揃え」を「center」に設定します。




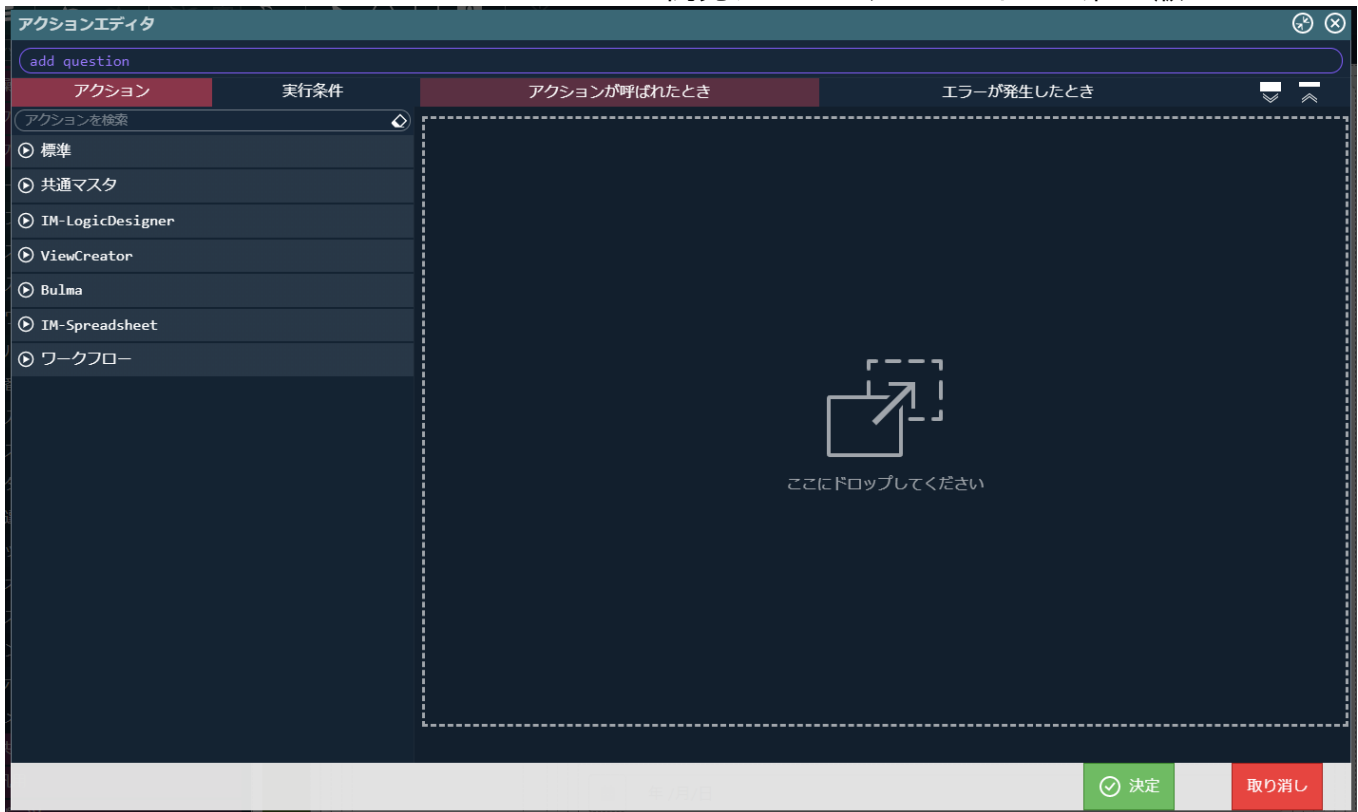
配置した「ボタン」の「ボタン要素」を選択し、「プロパティ」タブの「エレメント固有」 - 「textContent」に「登録」と設定します。



これで、入力項目の配置は完了です。
アンケート登録画面に関するアクションを作成します。

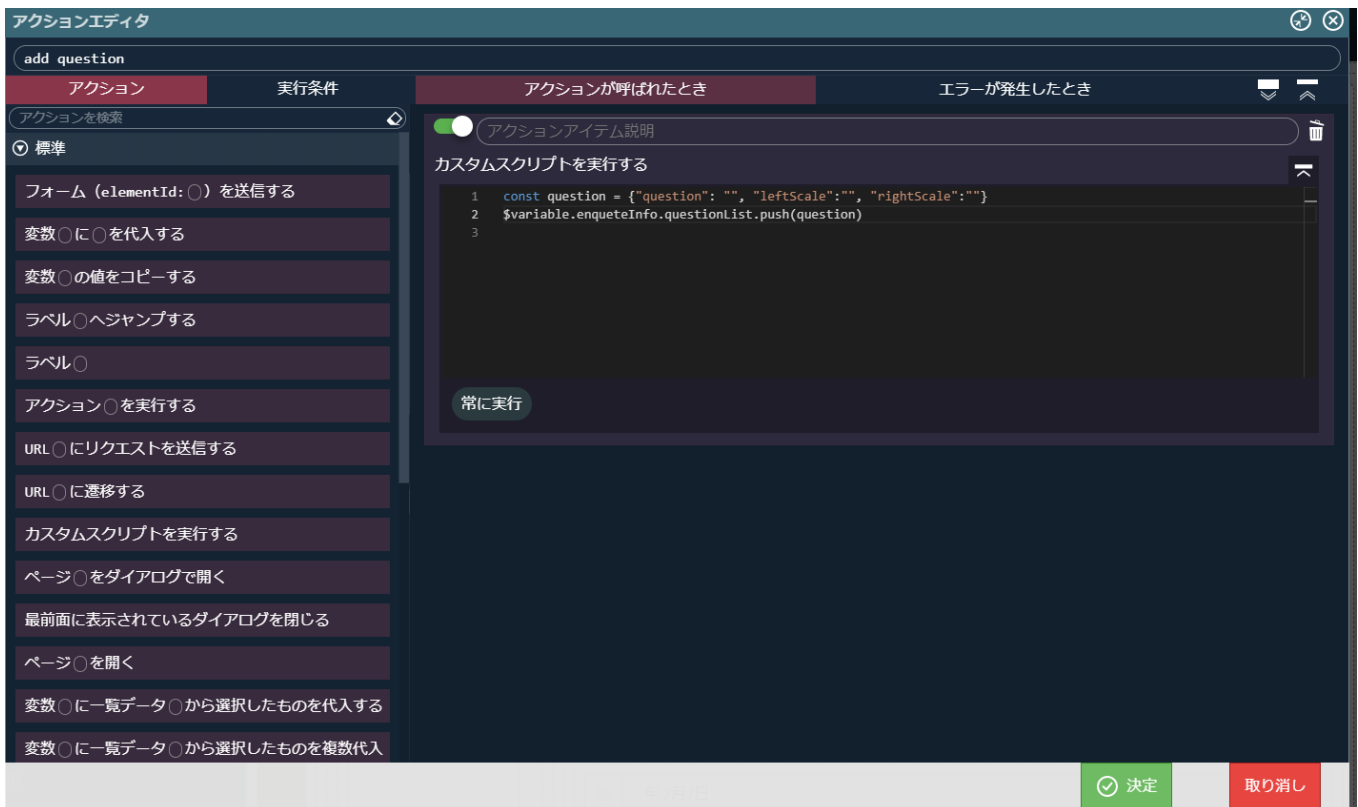
アンケート登録画面のアクションの作成


次に、画面で実行されるアクションを作成します。
最初に、設問の行追加を行うアクションを作成します。
設問の行追加は「\$variable > enqueteInfo > questionList」の配列の末尾にオブジェクトを追加して行います。
画面右側の「アクション」タブを選択し、 をクリックしアクションエディタを表示します。
アクションエディタ上部の名称を「新しいアクション #1」を「add question」に変更します。

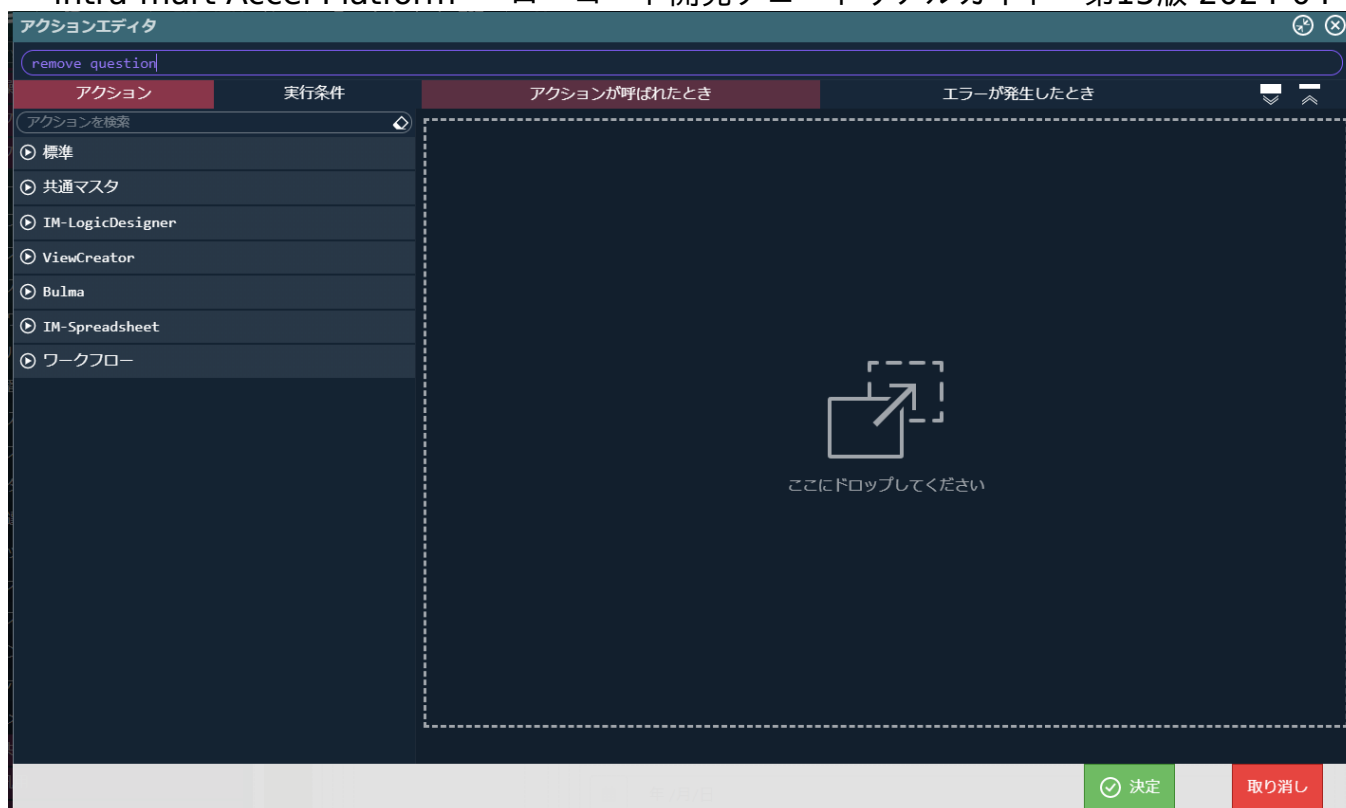


左側の「アクション」タブから「標準」 - 「カスタムスクリプトを実行する」を配置します。
配置した「カスタムスクリプト」の入力欄に以下のスクリプトを設定し、決定ボタンをクリックします。

```
const question = {"question": "", "leftScale": "", "rightScale": ""}
$variable.enqueteInfo.questionList.push(question)
```

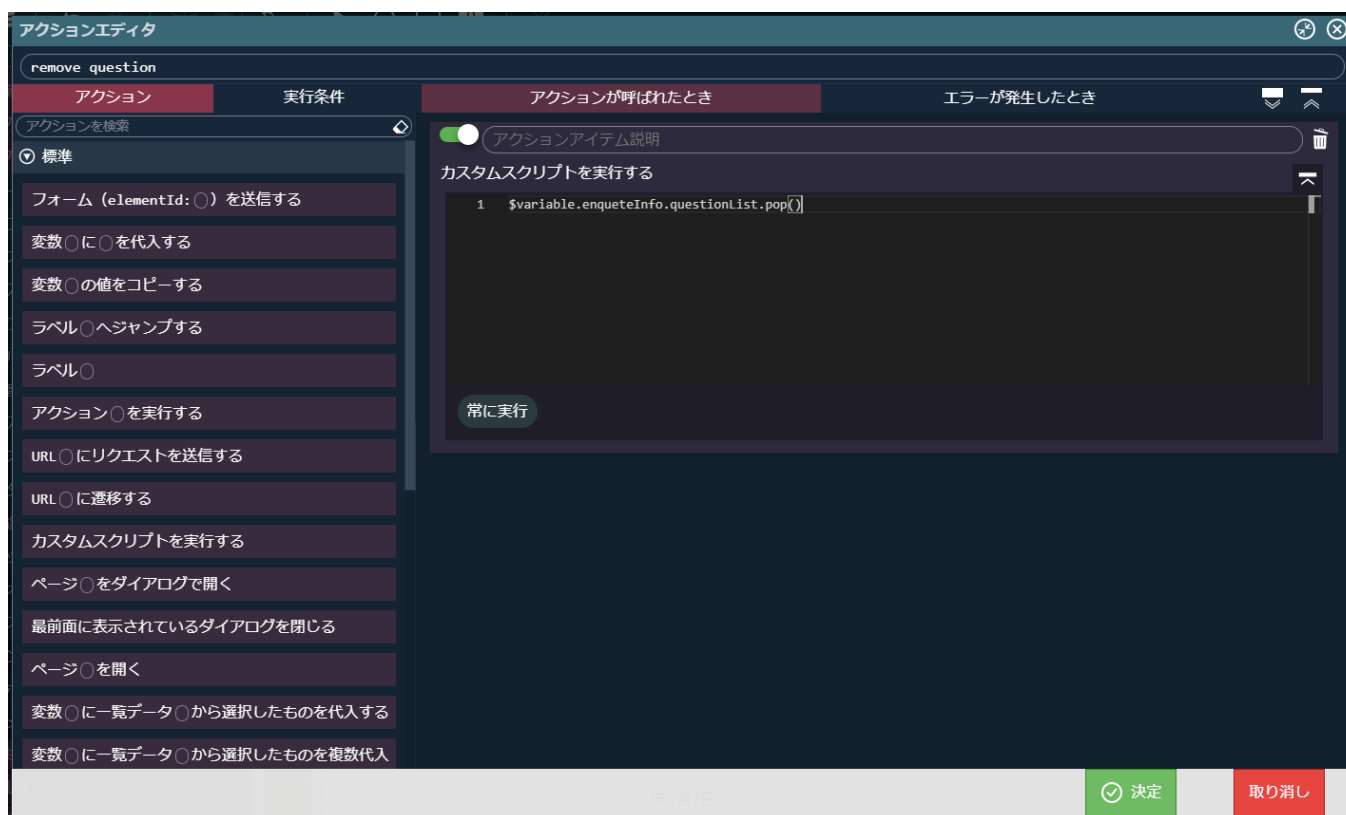


続けて、設問の行削除を行うアクションを作成します。
設問の行削除は「\$variable > enqueteInfo > questionList」の配列の末尾を削除して行います。
 をクリックしアクションエディタを表示します。
アクションエディタ上部の名称を「新しいアクション #1」を「remove question」に変更します。



左側の「アクション」タブから「標準」 - 「カスタムスクリプトを実行する」を配置します。
配置した「カスタムスクリプト」の入力欄に以下のスクリプトを設定し、決定ボタンをクリックします。

```
$variable.enqueteInfo.questionList.pop()
```



次に、アンケートを登録するアクションを作成します。



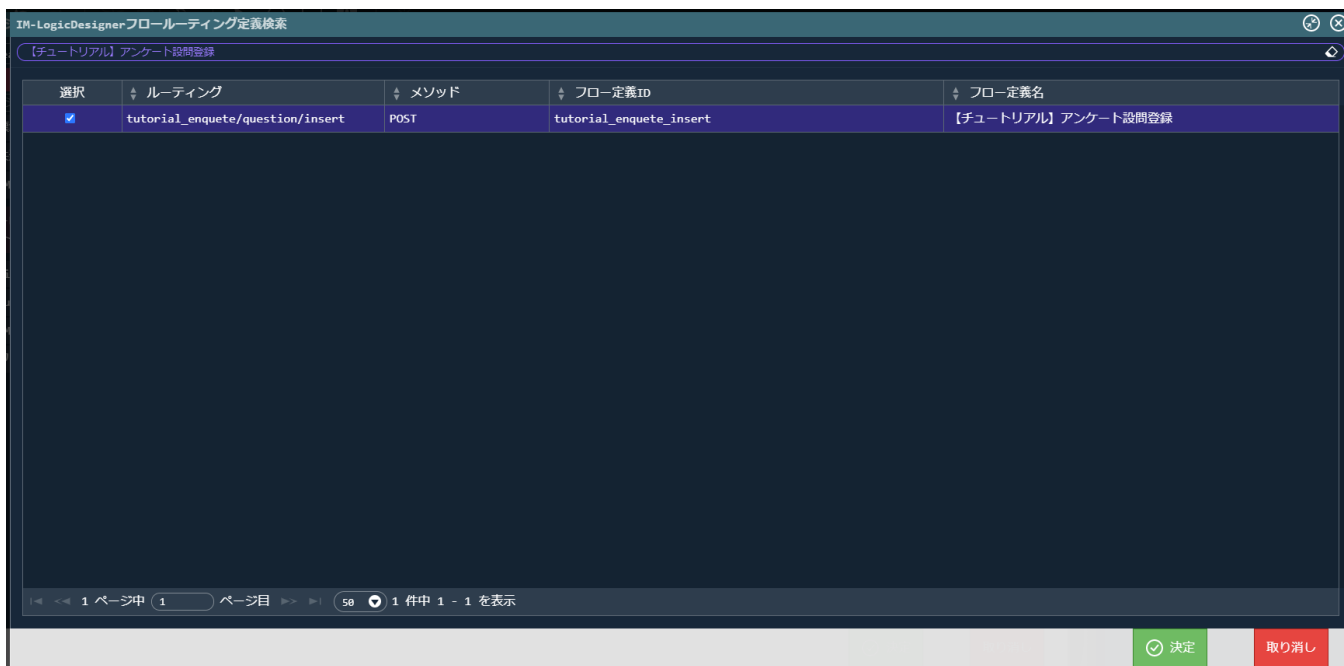
をクリックしアクションエディタを表示します。

アクションエディタ上部の名称を「新しいアクション #1」を「create enquete」に変更します。

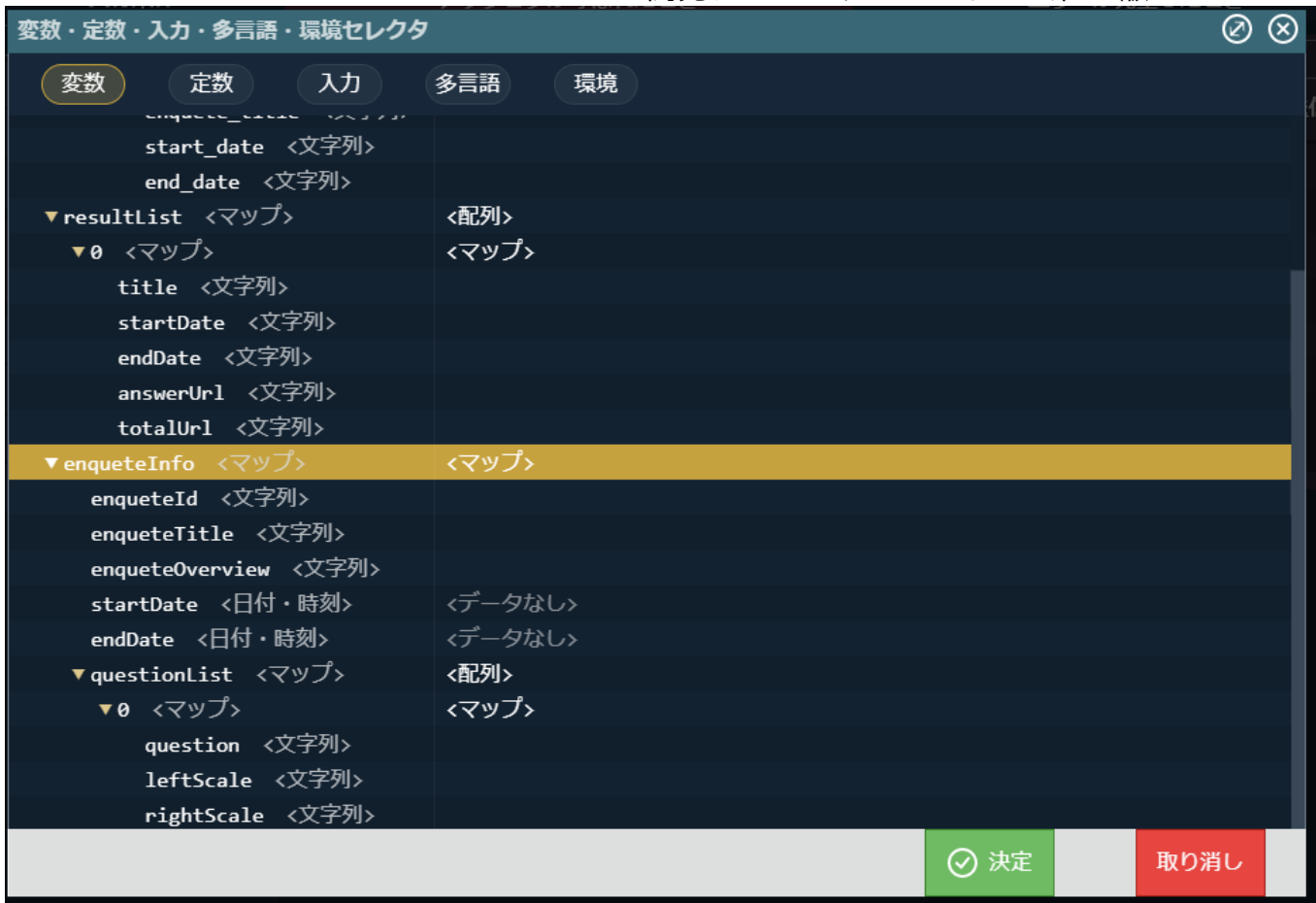
左側の「アクション」タブから「IM-LogicDesigner」 - 「IM-LogicDesigner フロールーティング○にリクエストを送信する」を配置します。



配置した「IM-LogicDesigner フロールーティング〇にリクエストを送信する」の 🔍 をクリックし、フロールーティング定義検索ダイアログから「tutorial_enquete/question/insert」を選択します。



業務ロジックヘデータを送信するために、リクエストパラメータの 🔍 から「変数 > \$variable > enqueteInfo」を設定します。



アンケート登録後、一覧画面を表示する処理を追加します。

「標準」 - 「ページ〇を開く」を先ほど配置した「IM-LogicDesigner フロールーティング〇にリクエストを送信する」の次に配置し、「ページ (list) を開く」に設定します。



登録後に一覧表示する際に、一覧を再表示するための処理を追加します。

「標準」 - 「アクション〇を実行する」を「ページ〇を開く」の次に配置し、「アクション (list) を実行する」に設定し、決定ボタンをクリックしてアクションを保存します。



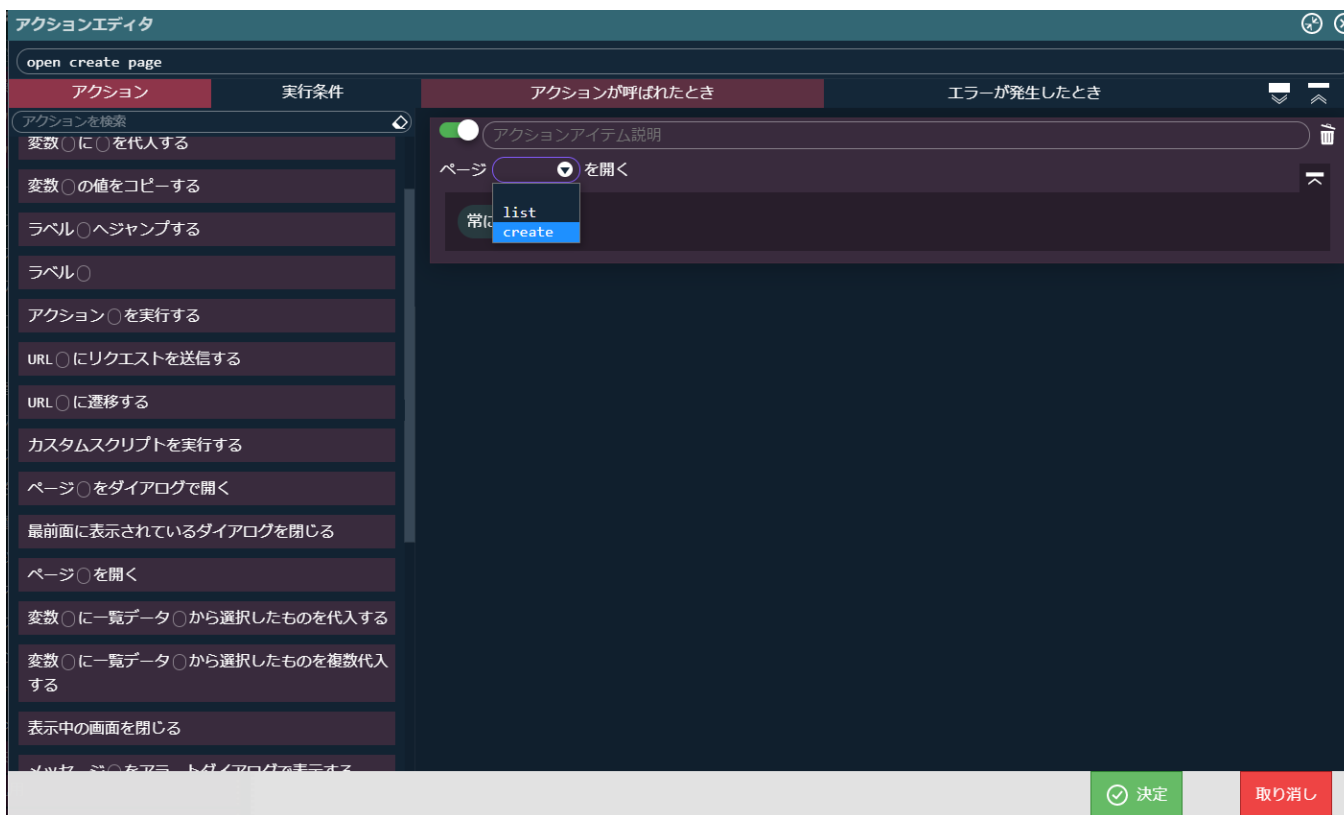
最後に、アンケート登録画面を表示するアクションを作成します。



をクリックしアクションエディタを表示します。

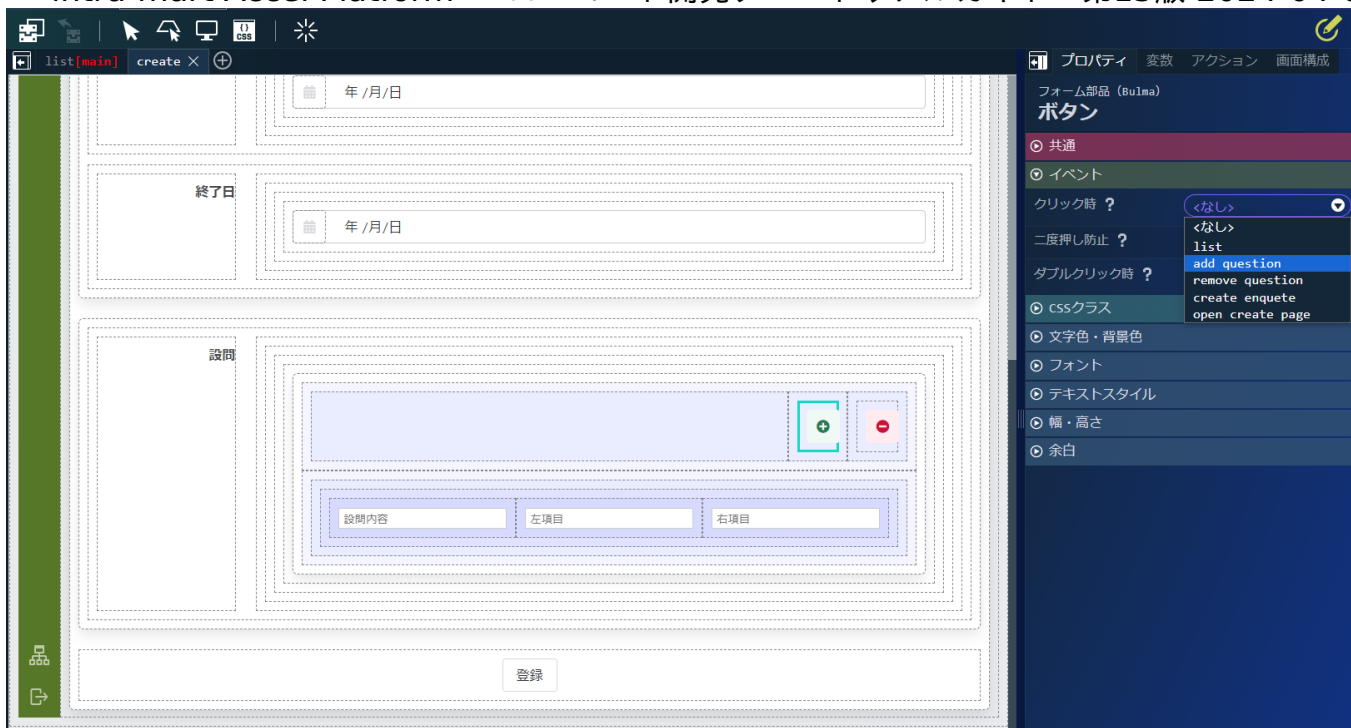
アクションエディタ上部の名称を「新しいアクション #1」を「open create page」に変更します。

左側の「アクション」タブから「標準」 - 「ページ〇を開く」を配置し、「ページ (create) を開く」に設定します。

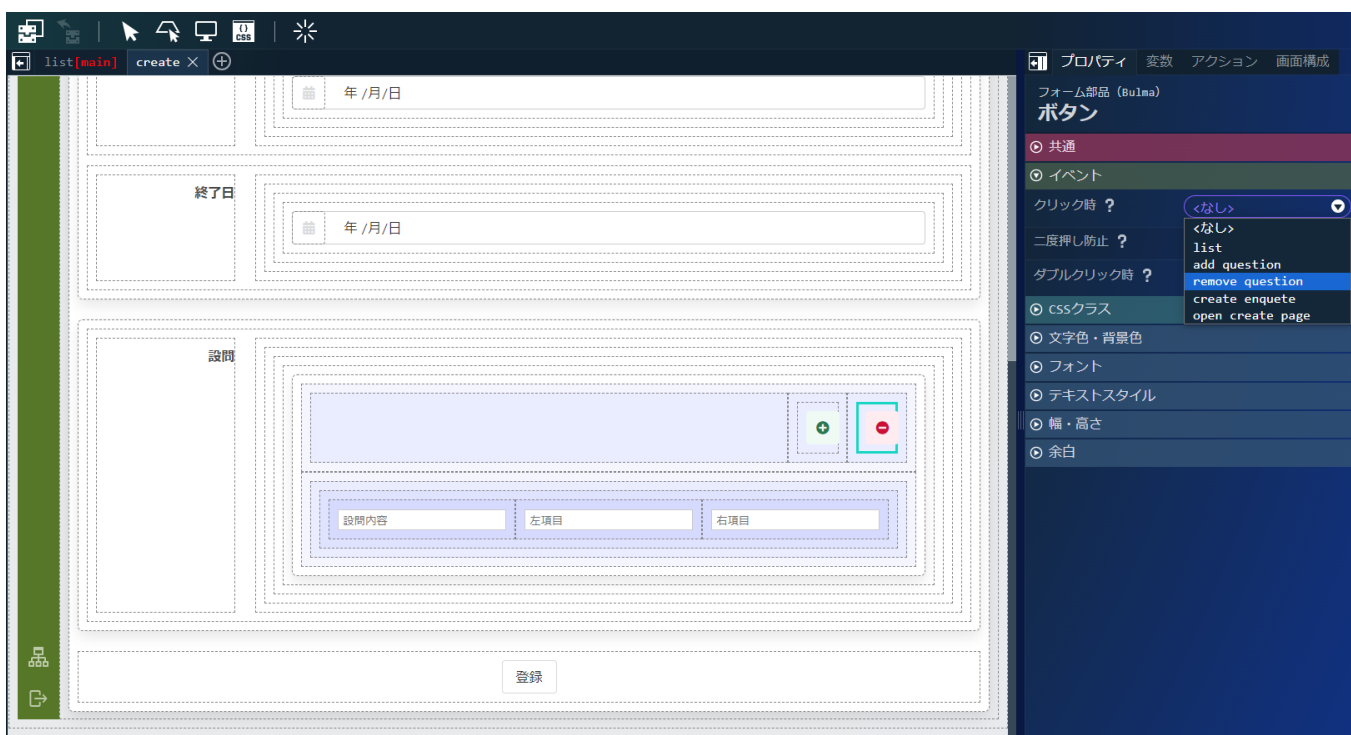


作成したアクションを各ボタンに設定します。

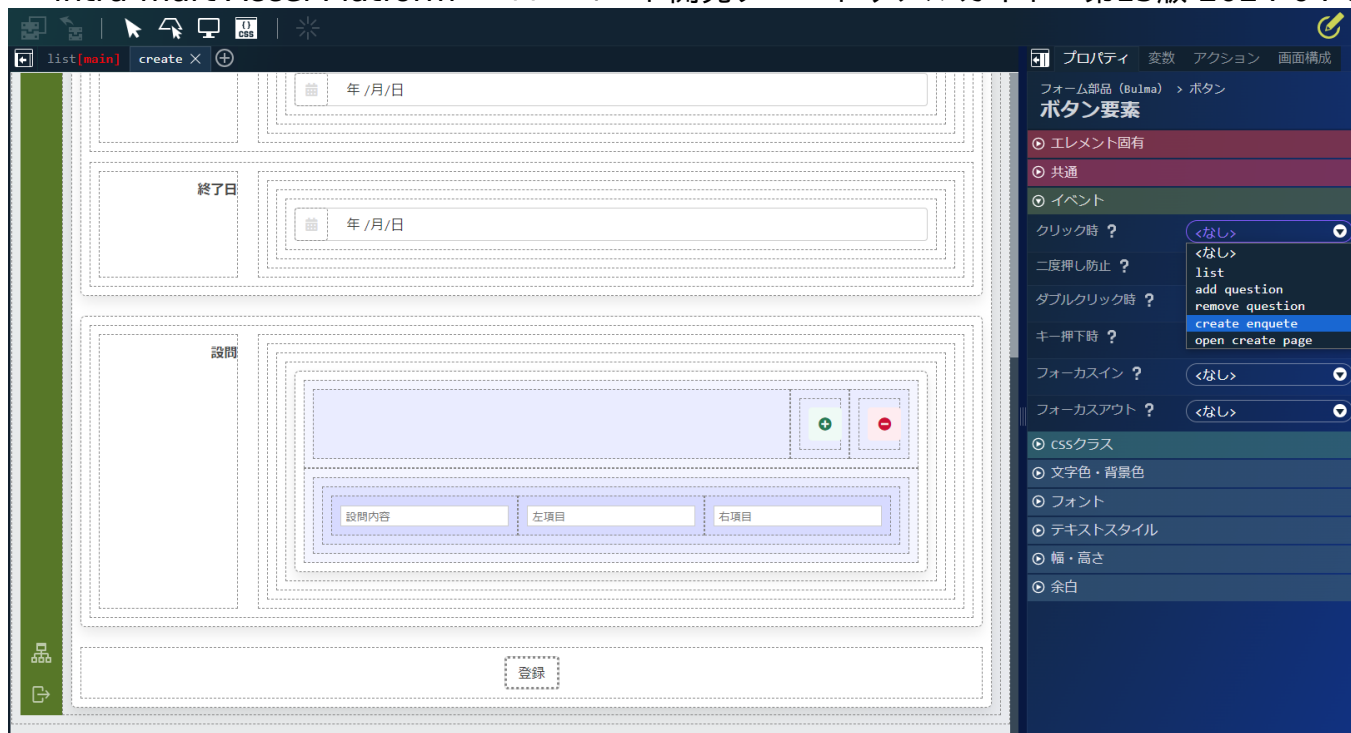
設問の入力欄の+アイコンを配置したボタンを選択し、「イベント」 - 「クリック時」に「add question」を設定します。



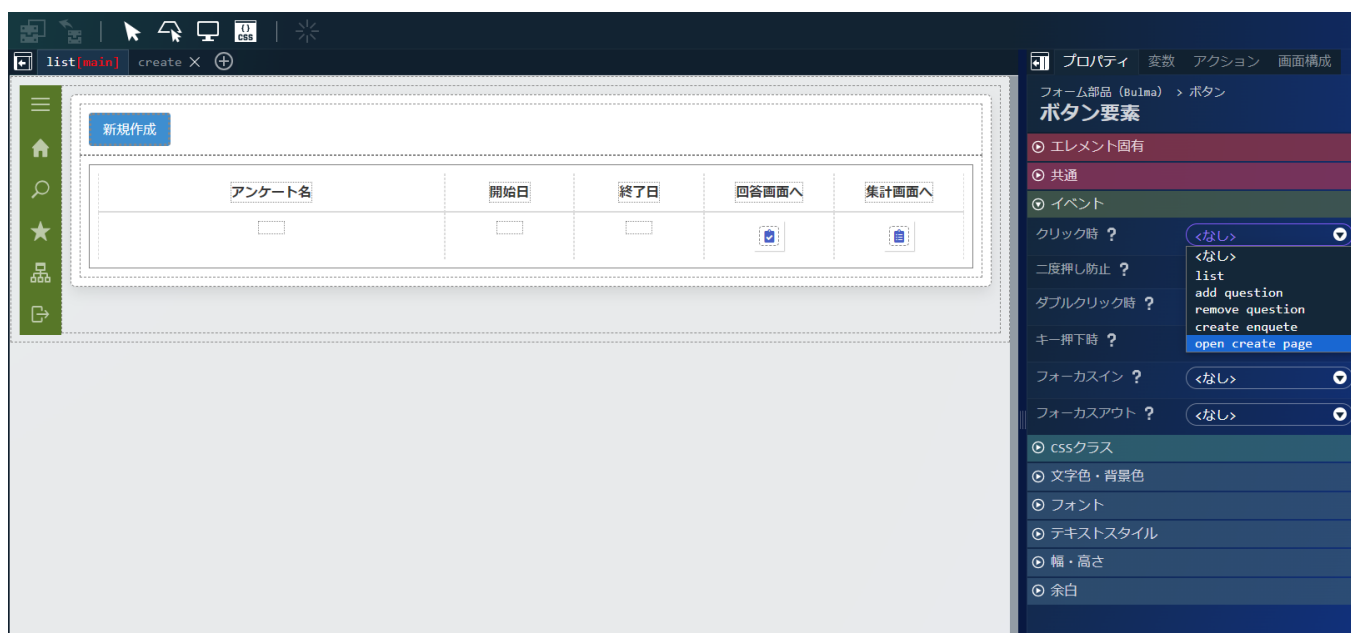
同様に、設問の入力欄の－アイコンを配置したボタンを選択し、「イベント」 - 「クリック時」に「remove question」を設定します。



アンケート登録ボタンの「ボタン要素」を選択し、「イベント」 - 「クリック時」に「create enquete」を設定します。



最後に、画面上部の「list」タブを選択して一覧画面を表示し、新規作成ボタンの「ボタン要素」 - 「イベント」 - 「クリック時」に「open create page」を設定します。

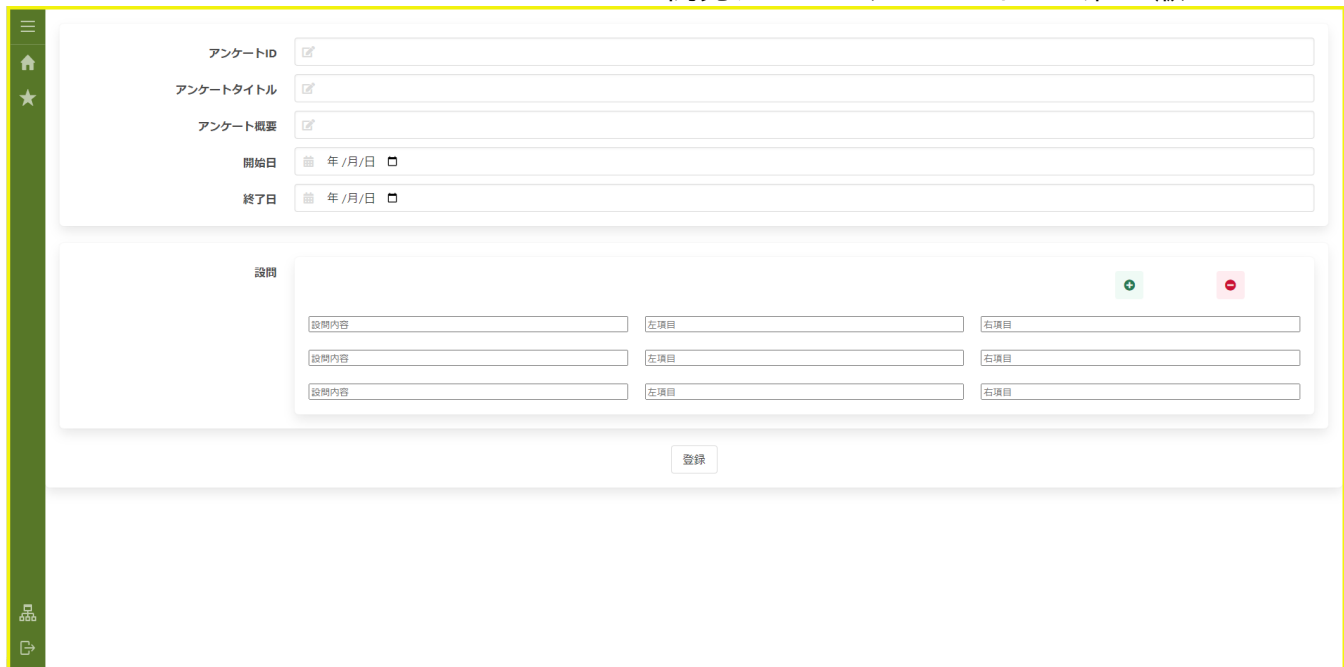


これで、作成したアクションの設定が完了しました。動作確認を行きましょう。

画面上部の  をクリックします。

プレビューダイアログが表示され、新規作成ボタンをクリックすると「アンケート登録画面」が表示されます。

アンケート登録画面では+ボタンと-ボタンでアンケートの設問の行が増減し、登録ボタンでアンケートの登録が行えます。



ここまででアンケートの一覧と登録が行える画面が作成できました。


しかし、このままでは設問項目を全て埋めなくても登録できたり、登録後再度アンケート登録画面を表示した際に前回登録した内容がそのままになっていたりします。

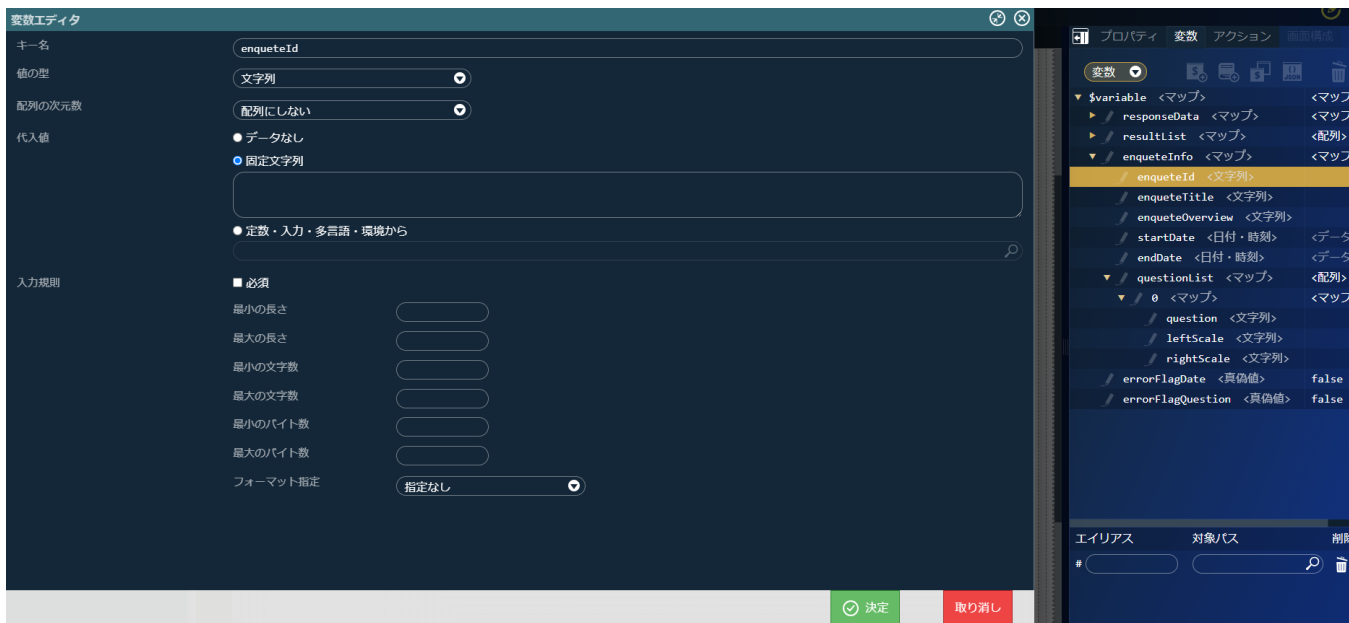
登録画面での入力チェックや、アンケート登録画面表示時の変数の初期化処理などを追加していきましょう。

入力チェックなどの追加

まずは、アンケート情報の入力チェックを追加します。

画面上部の「create」タブを選択して、アンケート登録画面を表示します。

画面右側の「変数」タブを選択し、「\$variable > enqueteInfo > enqueteId」の横の  をクリックして変数エディタを表示します。



変数エディタの「入力規則」 - 「必須」にチェックを入れ、「フォーマット指定」を「ID形式」と設定します。

変数エディタ

キー名: enqueteId

値の型: 文字列

配列の次元数: 配列にしない

代入値:

- データなし
- 固定文字列

入力規則:

- 必須
- 最小の長さ
- 最大の長さ
- 最小の文字数
- 最大の文字数
- 最小のバイト数
- 最大のバイト数
- フォーマット指定: ID形式

決定 取り消し

同様に、「\$variable > enqueteInfo > enqueteTitle」の横の  をクリックして変数エディタを表示し、「必須」にチェックを入れます。

「フォーマット指定」は標準の「指定なし」のままとします。

変数エディタ

キー名: enqueteTitle

値の型: 文字列

配列の次元数: 配列にしない

代入値:

- データなし
- 固定文字列

入力規則:

- 必須
- 最小の長さ
- 最大の長さ
- 最小の文字数
- 最大の文字数
- 最小のバイト数
- 最大のバイト数
- フォーマット指定: 指定なし

決定 取り消し

同様の手順で、「\$variable > enqueteInfo > enqueteOverview」、「\$variable > enqueteInfo > startDate」、「\$variable > enqueteInfo > endDate」を「必須」に設定します。

変数が入力規則にマッチしない場合に、アンケート登録処理を終了するようにアクションを設定します。

「アクション」タブを選択し、「create enquete」を選択してアクションエディタを表示します。

左側の「アクション」タブから「標準」 - 「ラベル〇」を一番最後のアクションとして配置し、「ラベル (END)」と設定します。



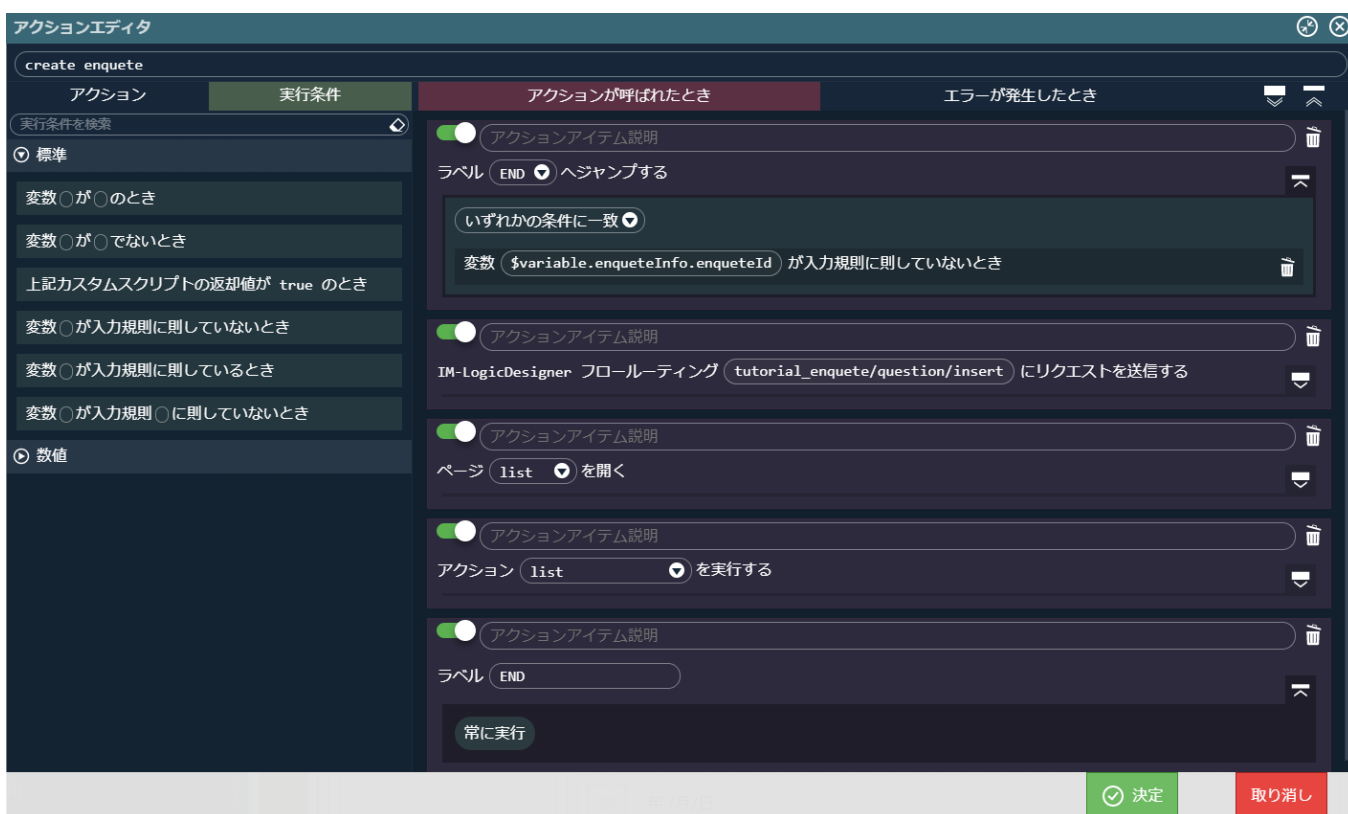
「標準」 - 「ラベル〇へジャンプする」を一番最初のアクションとして配置し、「ラベル (END) へジャンプする」と設定します。



「実行条件」タブの「標準」 - 「変数〇が入力規則に即していないとき」を選択して実行条件欄に配置し、実行条件を「いずれかの条件に一致」とします。



「変数〇が入力規則に即していないとき」の変数欄を選択し、「\$variable > enqueteInfo > enqueteId」を選択します。




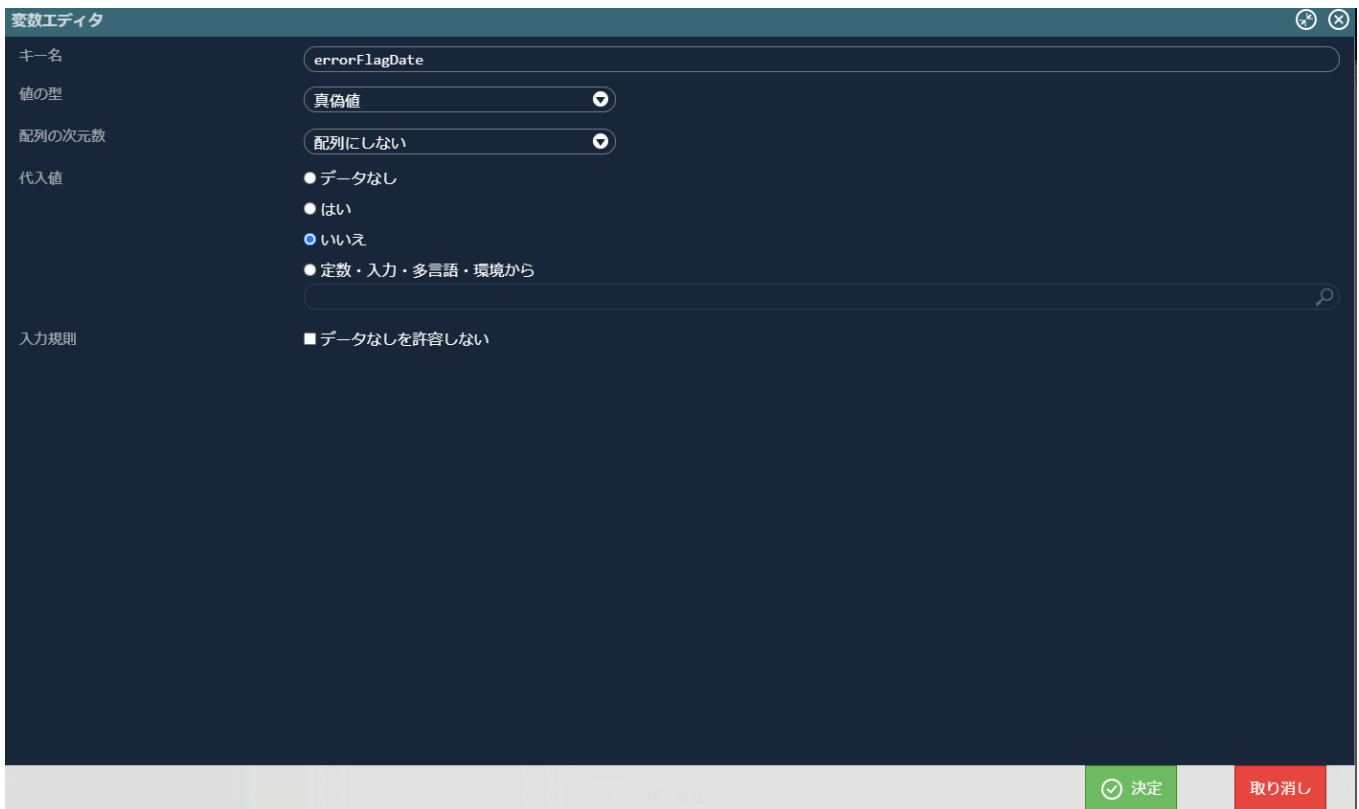
同様の手順で、「変数〇が入力規則に即していないとき」を配置し、「\$variable > enqueteInfo > enqueteTitle」、「\$variable > enqueteInfo > enqueteOverview」、「\$variable > enqueteInfo > startDate」、「\$variable > enqueteInfo > endDate」を設定します。



これで、必須として設定した各変数が空の場合に登録処理の呼び出しや一覧画面の表示をスキップされるようになりました。続けて、終了日が開始日より前に指定された場合と、設問のいずれかの項目が未指定の場合の入力チェックを行います。上記の入力チェックはカスタムスクリプトで判定を行います。

まずは、日付の入力チェック結果を格納する変数を作成します。

「決定」ボタンをクリックしてアクションを保存し、「変数」タブを選択、 から変数エディタを表示します。キー名を「errorFlagDate」、値の型を「真偽値」、代入値を「いいえ」に設定して変数を登録します。



続けて、設問の入力チェック結果を格納する変数を作成します。

 から変数エディタを表示し、キー名を「errorFlagQuestion」、値の型を「真偽値」、代入値を「いいえ」に設定して変数を登録します。

変数エディタ

キー名: errorFlagQuestion

値の型: 真偽値

配列の次元数: 配列にしない

代入値:

- データなし
- はい
- いいえ
- 定数・入力・多言語・環境から


入力規則:

- データなしを許容しない

決定 取り消し

変数はJSON入力で以下のコードを記載しても設定できます。

```
"errorFlagDate": false,
"errorFlagQuestion": false
```

JSON入力する場合は、変数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

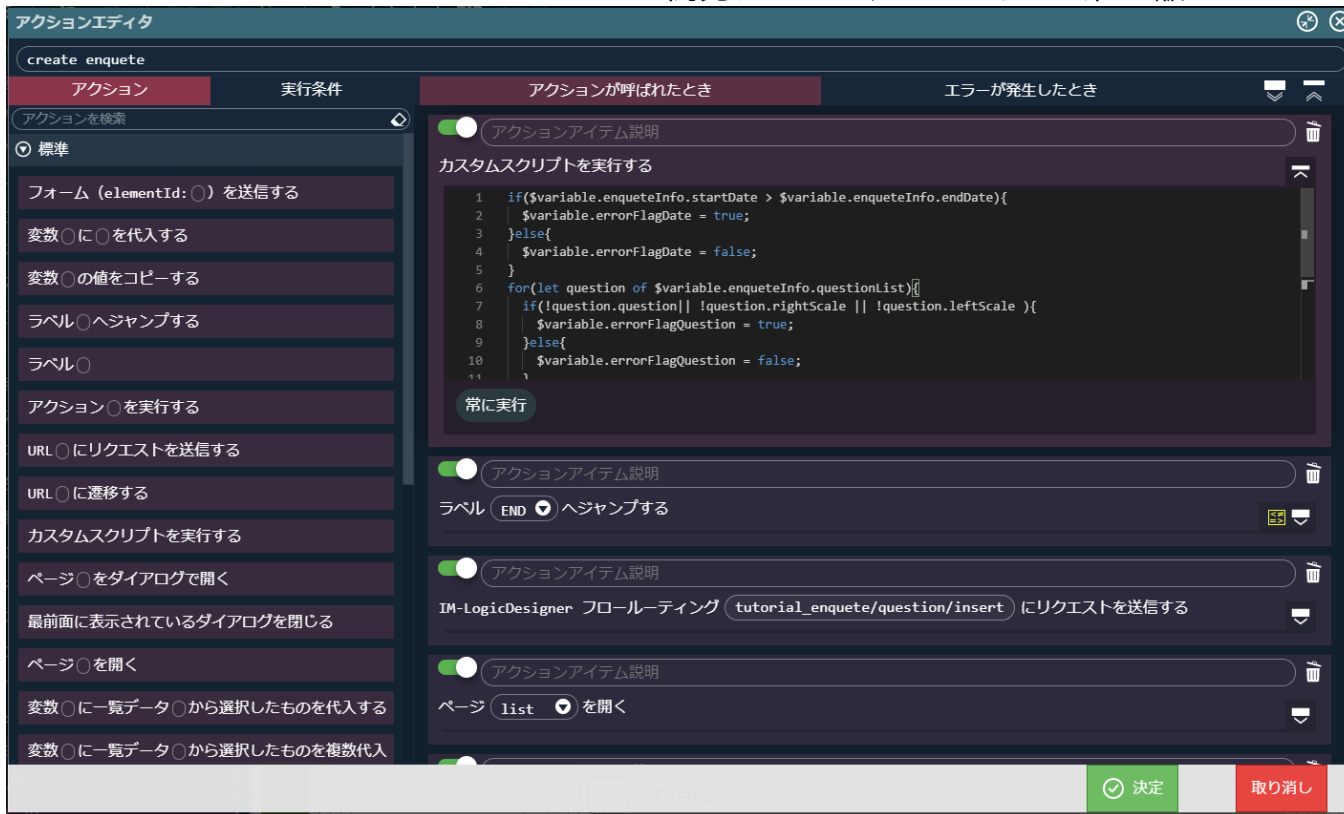
作成した変数を利用して入力チェックを行います。

「アクション」タブを選択し、「create enquete」を選択してアクションエディタを表示します。

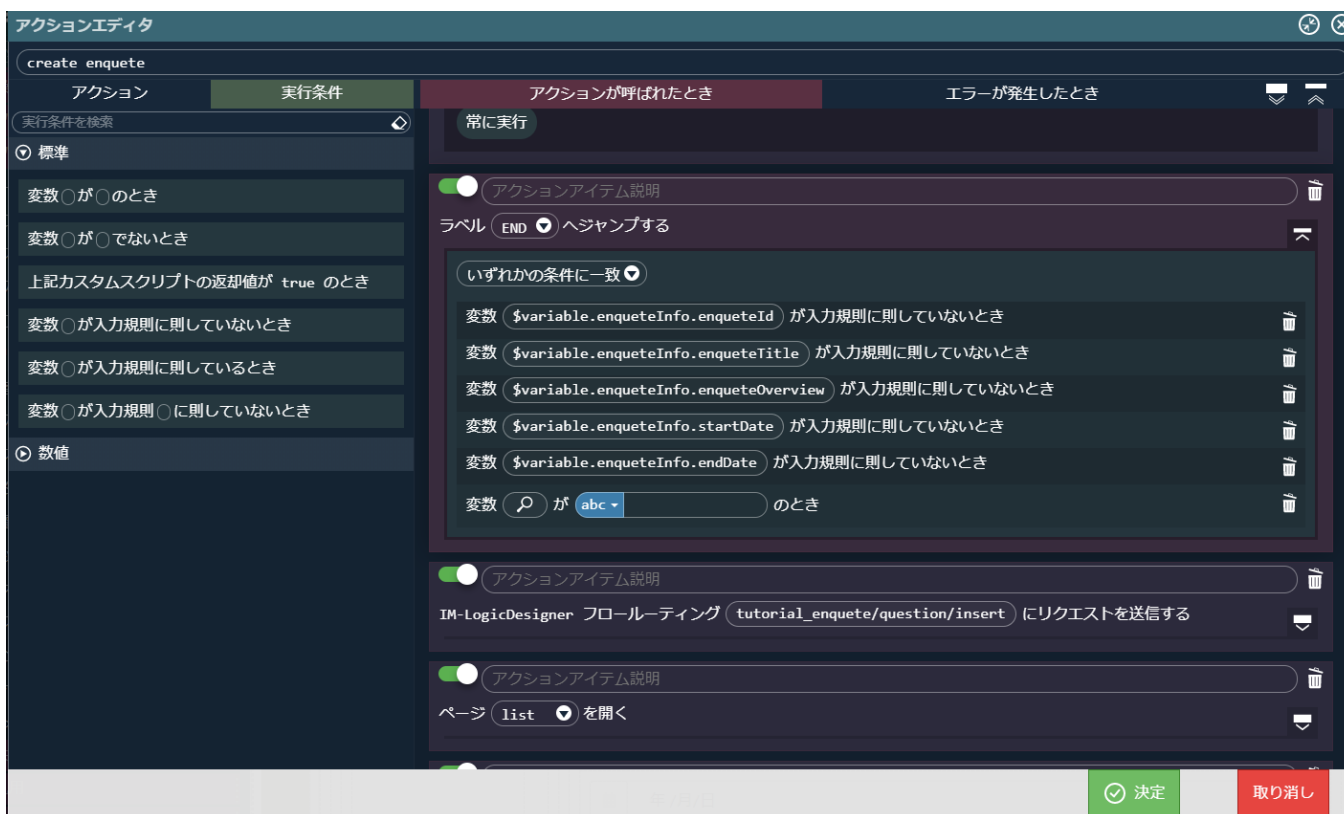
左側の「アクション」タブから「標準」 - 「カスタムスクリプトを実行する」を一番最初のアクションとして配置します。

配置した「カスタムスクリプト」の入力欄に以下のスクリプトを設定します。


```
if($variable.enqueteInfo.startDate > $variable.enqueteInfo.endDate){
  $variable.errorFlagDate = true;
}else{
  $variable.errorFlagDate = false;
}
for(let question of $variable.enqueteInfo.questionList){
  if(!question.question|| !question.rightScale || !question.leftScale ){
    $variable.errorFlagQuestion = true;
  }else{
    $variable.errorFlagQuestion = false;
  }
}
```



「ラベル (END) ヘジャンプする」の実行条件欄を表示し、「実行条件」タブの「変数〇が〇のとき」を配置します。



変数として「\$variable > errorFlagDate」を選択し、「abc▼」をクリックして入力値の種類を固定値から変数値へと変更します。

 をクリックし「環境 > \$env > const > true」を選択します。



同様の手順で、「変数（\$variable.errorFlagQuestion）が（\$env.const.true）のとき」を設定し、決定ボタンをクリックしてアクションを設定します。

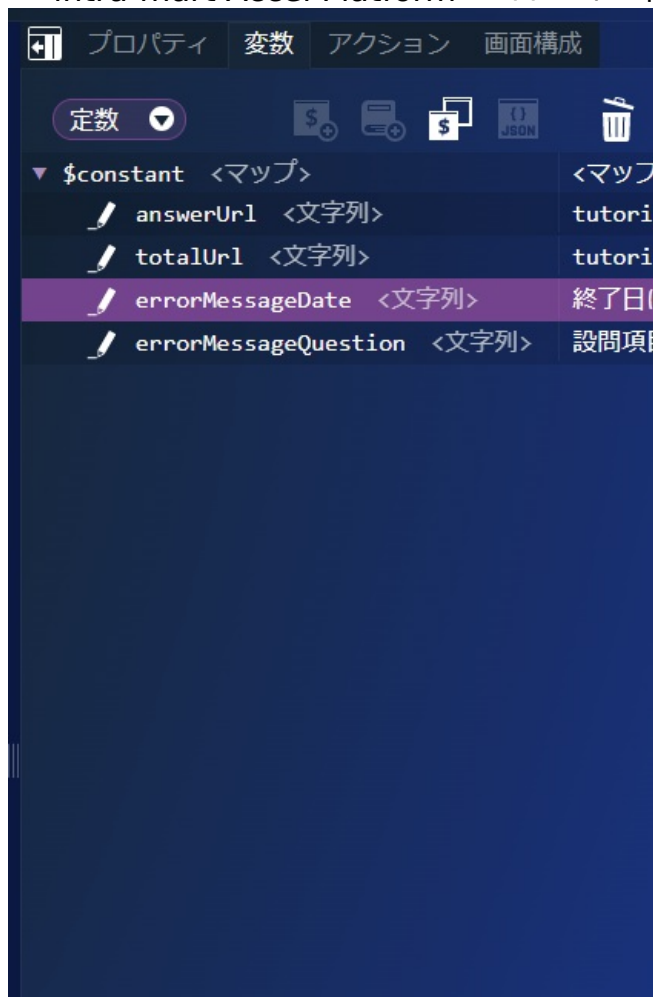


これで、登録時に入力漏れや開始、終了日に誤りがあった際には登録処理が実行されなくなりました。

エラー発生時に事象をわかりやすくするため、開始日と終了日のチェックと設問の入力チェックで失敗した場合にエラーメッセージを画面上に表示するようにしましょう。


まずは、エラーメッセージ用の定数を作成します。

「変数」タブで「定数」を選択し、「errorMessageDate」と「終了日は開始日よりも後に設定してください」、「errorMessageQuestion」「設問項目はすべて入力してください」を設定します。



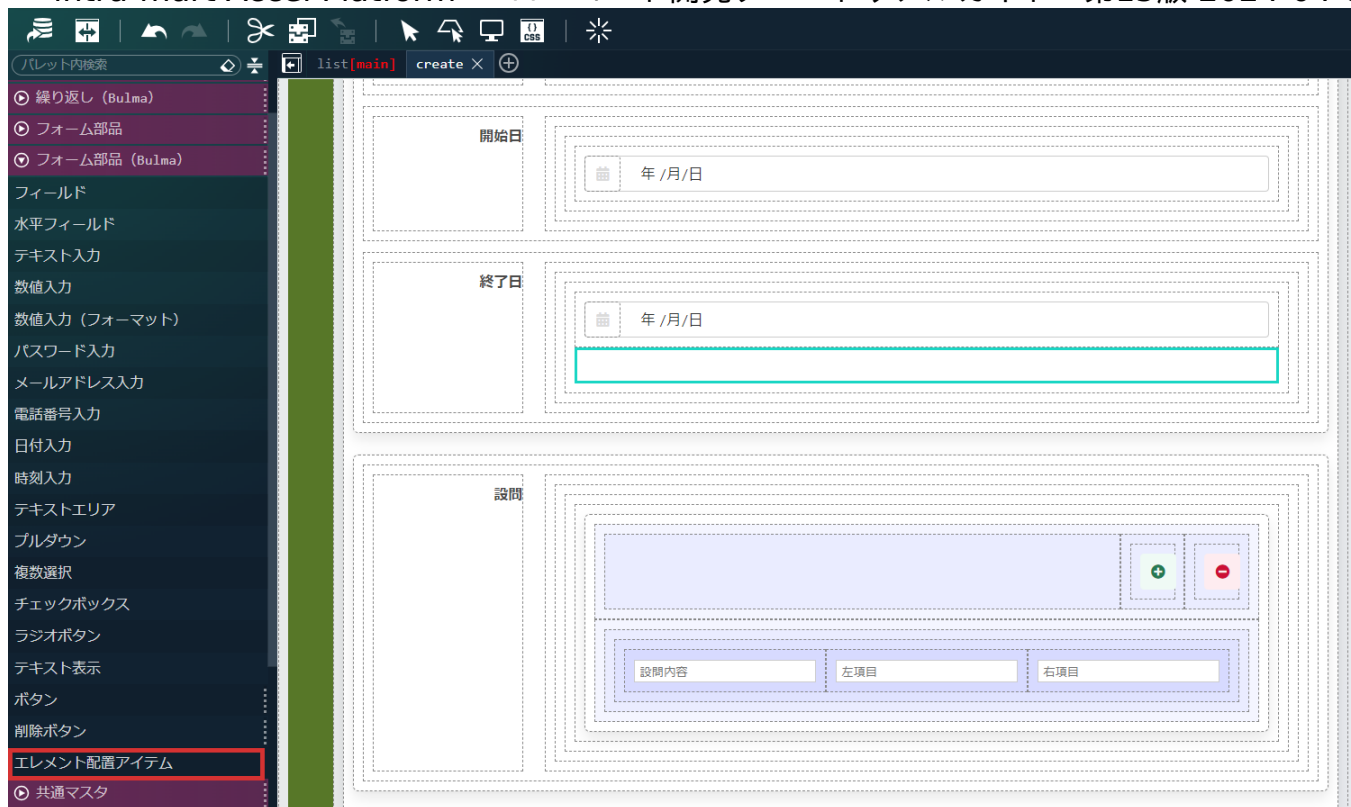
定数もJSON入力で以下を記載して設定できます。

```
"errorMessageDate": "終了日は開始日よりも後に設定してください。",
"errorMessageQuestion": "設問項目はすべて入力してください"
```

JSON入力する場合は、入力値の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

次に、開始、終了日の入力チェックのメッセージを設定します。

画面左側のパレットの「フォーム部品 (Bulma)」 - 「エレメント配置アイテム」を選択し、終了日の日付入力の下に配置します。

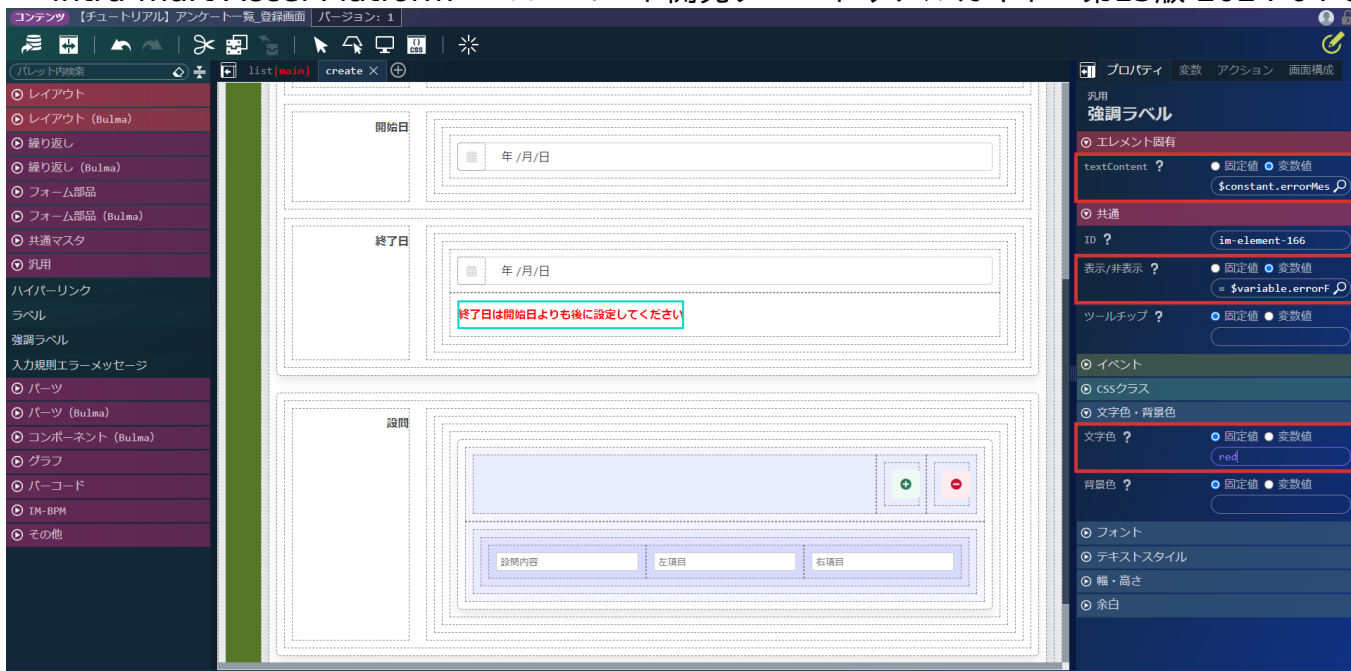


「汎用」 - 「強調ラベル」を選択し、先ほど配置した「エレメント配置アイテム」の中に配置します。



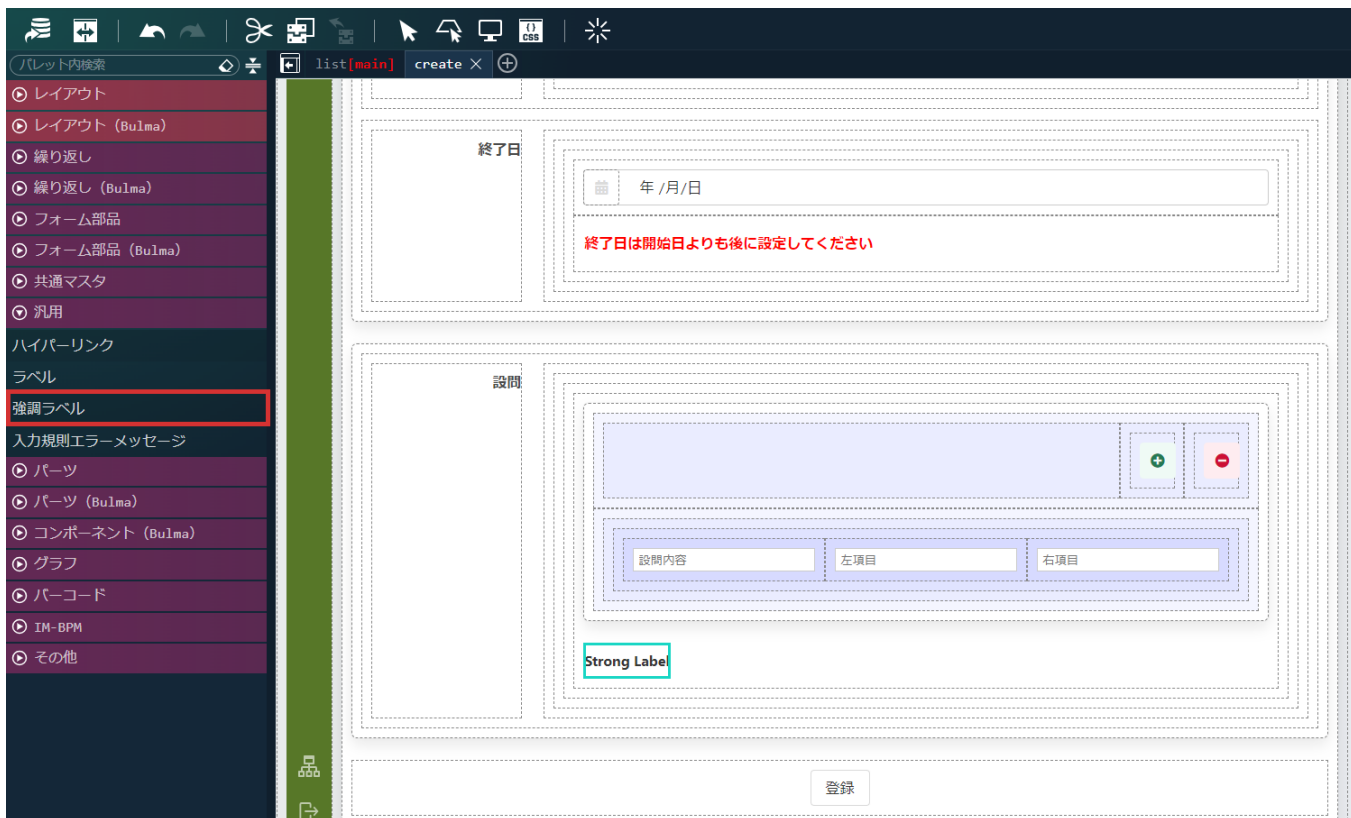
配置した「強調ラベル」を選択し、「プロパティ」タブの「エレメント固有」 - 「textContent」を変数値に変更して「`!constant.errorMessageDate`」と設定します。

「共通」 - 「表示/非表示」を変数値とし「`! = $variable.errorFlagDate == true`」、 「文字色・背景色」 - 「文字色」に「red」と設定します。



続けて、設問の入力チェックのエラーメッセージを配置します。

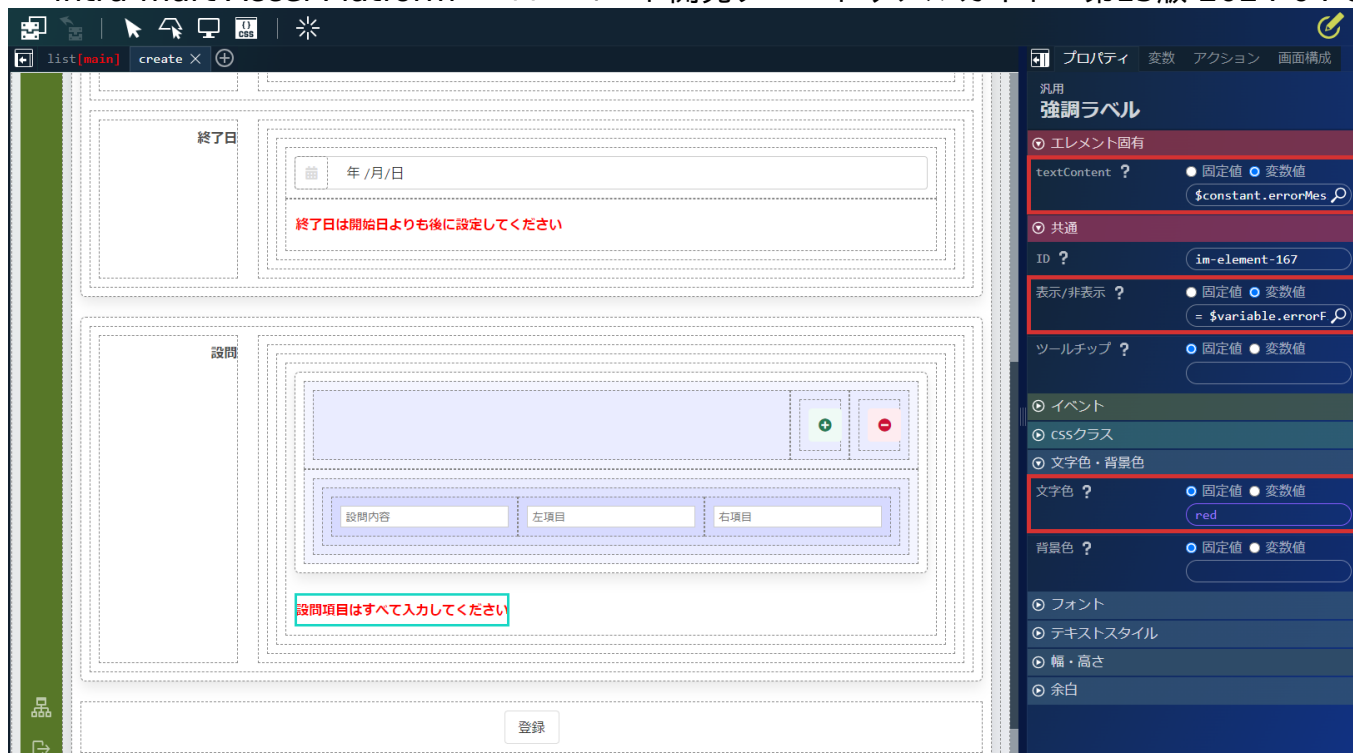
「汎用」 - 「強調ラベル」を選択し、設問内容などを配置しているエレメント配置アイテムの一番下に配置します。



配置した「強調ラベル」を選択し、「プロパティ」タブの「エレメント固有」 - 「textContent」を変数値に変更して

「\$constant.errorMessageQuestion」と設定します。

「共通」 - 「表示/非表示」を変数値とし「= \$variable.errorFlagQuestion == true」、「文字色・背景色」 - 「文字色」に「red」と設定します。



最後に、登録画面表示時に変数を初期化する処理を追加します。

本チュートリアルでは、変数の初期化は定数に登録用変数の初期状態の変数を用意しておき、上書く方法で行います。

まずは、登録用変数の初期状態の変数を用意します。

「変数」タブで「変数」を選択し、「\$variable」を選択した状態で  をクリックし、JSONエディタで以下の「enqueteInfo」のJSONをコピーします。


```
"enqueteInfo": {
  "enquetelId": "",
  "enqueteTitle": "",
  "enqueteOverview": "",
  "startDate": null,
  "endDate": null,
  "questionList": [
    {
      "question": "",
      "leftScale": "",
      "rightScale": ""
    }
  ]
}
```


JSONエディタ

```

1  {
2  "responseData": {
3    "records": [
4      {
5        "enquete_id": "",
6        "enquete_title": "",
7        "start_date": "",
8        "end_date": ""
9      }
10   ]
11  },
12  "resultList": [
13    {
14      "title": "",
15      "startDate": "",
16      "endDate": "",
17      "answerUrl": "",
18      "totalUrl": ""
19    }
20  ],
21  "enqueteInfo": {
22    "enqueteId": "",
23    "enqueteTitle": "",
24    "enqueteOverview": "",
25    "startDate": null,
26    "endDate": null,
27    "questionList": [
28      {
29        "question": "",
30        "leftScale": "",
31        "rightScale": ""
32      }
33    ]
34  },
35  "errorFlagDate": false,
36  "errorFlagQuestion": false
37  }

```

「定数」を選択し、「\$constant」を選択した状態で  をクリックし、JSONエディタでコピーした「enqueteInfo」のJSONを貼り付けます。

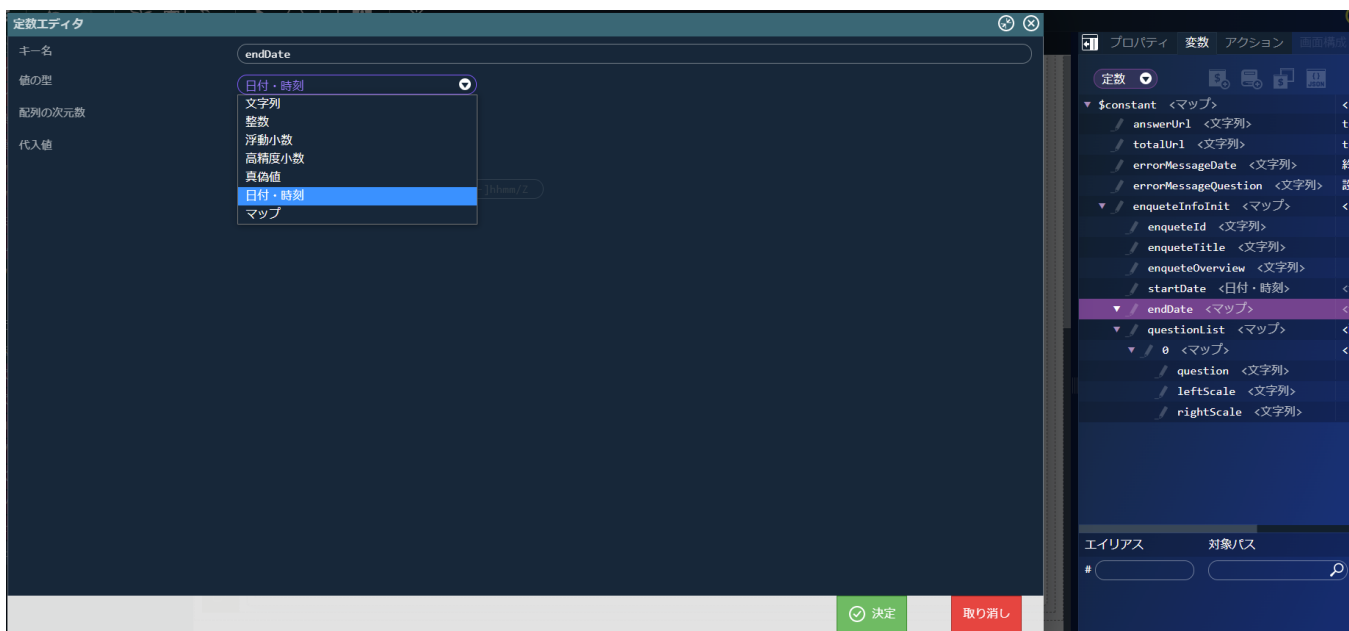
その際に、わかりやすくするために定数のキー名を「enqueteInfoInit」に変更しましょう

```

JSONエディタ
1  {
2    "answerUrl": "tutorial_enquete/answer?enqueteId=",
3    "totalUrl": "tutorial_enquete/totalize?enqueteId=",
4    "errorMessageDate": "終了日は開始日よりも後に設定してください。",
5    "errorMessageQuestion": "設問項目はすべて入力してください。",
6    "enqueteInfoInit": {
7      "enqueteId": "",
8      "enqueteTitle": "",
9      "enqueteOverview": "",
10     "startDate": null,
11     "endDate": null,
12     "questionList": [
13       {
14         "question": "",
15         "leftScale": "",
16         "rightScale": ""
17       }
18     ]
19   }
20 }

```

JSON入力の場合、初期値を「null」にした場合、値の型が「マップ」となるため、「\$constant > enqueteInfoInit > startDate」、「\$constant > enqueteInfoInit > endDate」の値の型を「日付・時刻」に変更します。



初期状態の定数を作成したため、アンケート登録画面を表示するアクションに変数の初期化処理を追加します。

「アクション」タブの「open create page」を選択し、アクションエディタを表示します。

左側の「アクション」タブから「標準」 - 「変数〇に〇を代入する」を一番最初のアクションとして配置し、「変数 (\$variable.enqueteInfo) に (\$constant.enqueteInfoInit) を代入する」に設定します。



これで画面の作成が完了しました。

画面URLの設定

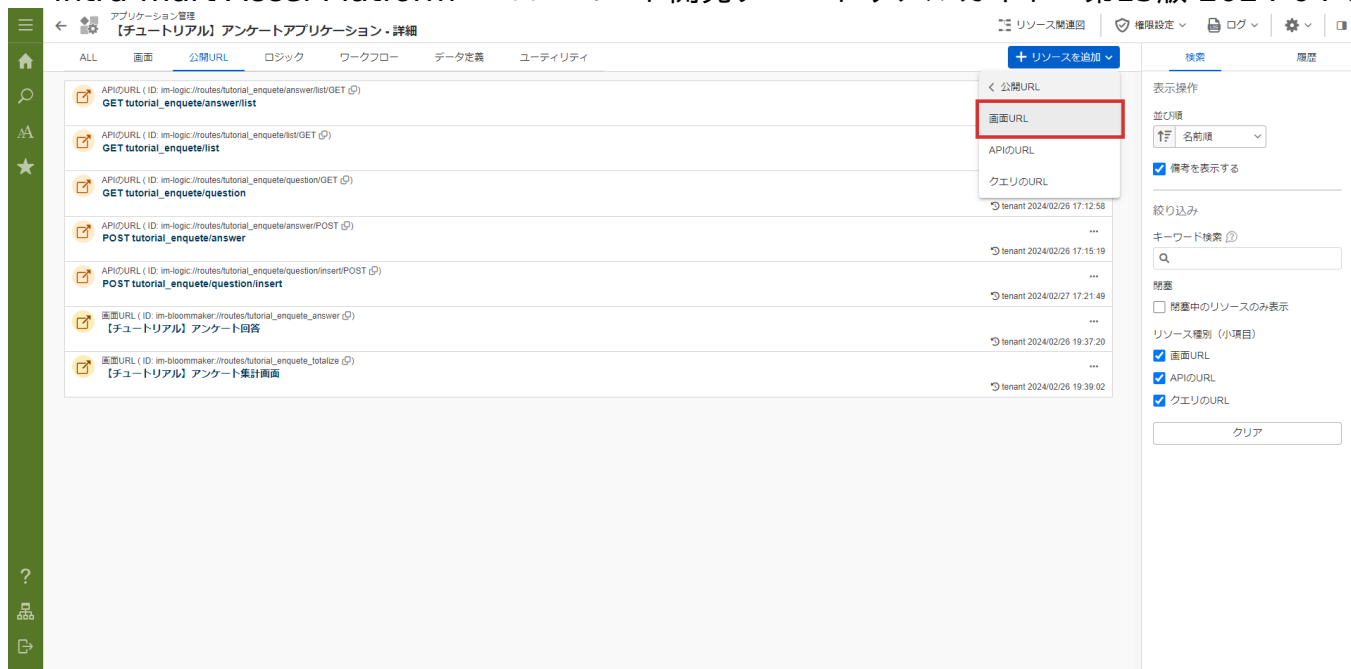
作成したアンケート回答画面と集計画面にアクセスするためのURLを設定します。

アンケート回答画面のURL設定

アンケート回答画面のURL設定を行います。

「【チュートリアル】アンケートアプリケーション」の管理画面に遷移し、「公開URL」タブを選択します。

画面右上の「リソースを追加」プルダウンから「公開URL」 - 「公開URLを新規作成」 - 「画面URL」を選択し「IM-BloomMaker ルーティング定義新規作成」画面を開きます。



「IM-BloomMaker ルーティング定義」画面の「ルーティング」タブ - 「カテゴリ」 - 「検索」から「ルーティングカテゴリ検索」ダイアログを表示します。

「ルーティングカテゴリ検索」ダイアログ上で「業務テンプレートから作成」を選択し、「決定」ボタンをクリックします。

「IM-BloomMaker ルーティング定義」画面の「ルーティング」タブ - 「ルーティングID」に「tutorial_enquete_list」と設定します。

「IM-BloomMaker ルーティング定義」画面の「ルーティング」タブ - 「コンテンツ」 - 「検索」から「コンテンツ検索」ダイアログを表示します。

「コンテンツ検索」ダイアログから「【チュートリアル】アンケート一覧_登録画面」を選択し、決定ボタンをクリックします。

ルーティング	前処理
<p>カテゴリ *</p> <p>検索</p> <p>カテゴリID: im_bouquet</p> <p>カテゴリ名: 業務テンプレートから作成</p>	
<p>ルーティングID *</p> <p>tutorial_enquete_list</p>	
<p>コンテンツ *</p> <p>検索</p> <p>コンテンツID: tutorial_enquete_list</p> <p>コンテンツ名: 【チュートリアル】アンケート一覧_登録画面</p>	
<p>コンテンツバージョン番号 *</p> <p> <input checked="" type="radio"/> 最新バージョンを利用する <input type="radio"/> 利用するバージョンを指定する </p> <p>利用バージョン *</p>	

「IM-BloomMaker ルーティング定義」画面の「ルーティング」タブ - 「URL」に「tutorial_enquete/list」と設定します。

「IM-BloomMaker ルーティング定義」画面の「ルーティング」タブ - 「ルーティング名」に「【チュートリアル】アンケート一覧_登録画面」と設定します。

その他の設定は標準通りとし、「IM-BloomMaker ルーティング定義」画面下の登録ボタンをクリックし、URL設定を登録します。

ルーティング	最終処理
カテゴリ *	<input type="text" value="検索"/> カテゴリID: im_bouquet カテゴリ名: 業務テンプレートから作成
ルーティングID *	tutorial_enquete_list
コンテンツ *	<input type="text" value="検索"/> コンテンツID: tutorial_enquete_list コンテンツ名: 【チュートリアル】 アンケート一覧_登録画面
コンテンツバージョン番号 *	<input checked="" type="radio"/> 最新バージョンを利用する <input type="radio"/> 利用するバージョンを指定する 利用バージョン * <input type="text"/>
メソッド *	GET
URL *	/imart/ tutorial_enquete/list
認可URI	im-bloommaker-content//contents/route/tutorial_enquete_list
ルーティング名	標準 * <input type="text" value="【チュートリアル】 アンケート一覧_登録画面"/>
備考	標準 <input type="text"/>

登録

ここまでで、アンケートアプリケーションへアクセスするURL設定ができました。
 これで、アンケートの一覧、登録画面の作成が完了しました。