

- 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. マスタメンテン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
 - 5.1.2. ToDo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
 - 5.3. ToDoアプリケーションに追加機能を作成する
 - 5.3.1. 概要
 - 5.3.2. テンプレートから機能を追加する
- 6. 付録
 - 6.1. ユーザのタイムゾーンや日付と時刻の形式に対応する
 - 6.1.1. 実装ガイド
 - 6.1.2. チュートリアル

| 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。■ 「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

第10版 下記を変更しました

- 「[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。
 - 「[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。
 - 「[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。
- 「[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

第11版 下記を変更しました

- 「[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

第12版 下記を変更しました

- 「[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

第13版 下記を変更しました

- 「[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

第14版 下記を変更しました

- 「[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](#)」の解説動画を公開しました。

2024-04-01

第15版 下記を変更しました

- 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。

2024-06-28

第16版 下記を追加しました

- 「[ユーザのタイムゾーンや日付と時刻の形式に対応する](#)」を追加しました。

| 変更年月日 | 変更内容 |
|------------|--|
| 2024-08-30 | <p>第17版 下記を追加・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」にプロコード版とローコード版に関する記載を追加しました。▪ 「ユーザのタイムゾーンや日付と時刻の形式に対応する」を変更しました。 |
| 2024-10-01 | <p>第18版 下記を追加・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」をベーシック版で IM-BIS と IM-FormaDesigner が利用可能とわかるよう変更しました。▪ 「チュートリアル（応用）」に「テンプレートから機能を追加する」を追加しました。▪ 「ゼロからアプリケーションを作成する」のデザインをimdsに変更しました。▪ 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」と「ゼロから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。▪ その他文言に軽微な修正を加えました。 |
| 2025-04-01 | <p>第19版 下記を追加・変更しました</p> <ul style="list-style-type: none">▪ 「ゼロからアプリケーションを作成する」を IM-Repository 対応のシナリオに更新しました。▪ 「ユーザのタイムゾーンや日付と時刻の形式に対応する」の「実装ガイド」に、「個別の機能要件」を追加しました。▪ 「保存されている日付を画面に表示する」と「保存されている日付一覧をリッチテーブルに一覧表示する」に、日付を絶対的な値として取り扱う場合のノートを追加しました。▪ 「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「ゼロから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。▪ その他文言に軽微な修正を加えました。 |
| 2025-04-25 | <p>第20版 下記を追加・変更しました</p> <ul style="list-style-type: none">▪ 「ゼロからアプリケーションを作成する」一部キャプチャの追加および差し替えをしました。▪ その他文言に軽微な修正を加えました。 |
| 2025-10-01 | <p>第21版 下記を追加・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」と「ゼロからアプリケーションを作成する」の一部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 2025 Spring(Kamille)をベースに作成しています。

※「[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。
- [付録](#)

チュートリアル実行ユーザ

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-共通マスター 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。

基本的にはドラック&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 | [CSL] | | | | | | | | | |
|---------------------|------------------|--------|--------|----------|------------|------------|-------------|-------------|-----------|---------------|---------------|
| | | スタンダード | アドバンスト | エンタープライズ | Accel-Mart | [CSL] Plus | [CSL] ベーシック | [CSL] アドバンス | プロフェッショナル | [ASEAN] プロコード | [ASEAN] ローコード |
| 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。

[ASEAN] は、ASEAN地域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 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トをはじめ、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。

グレープ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て設定し、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

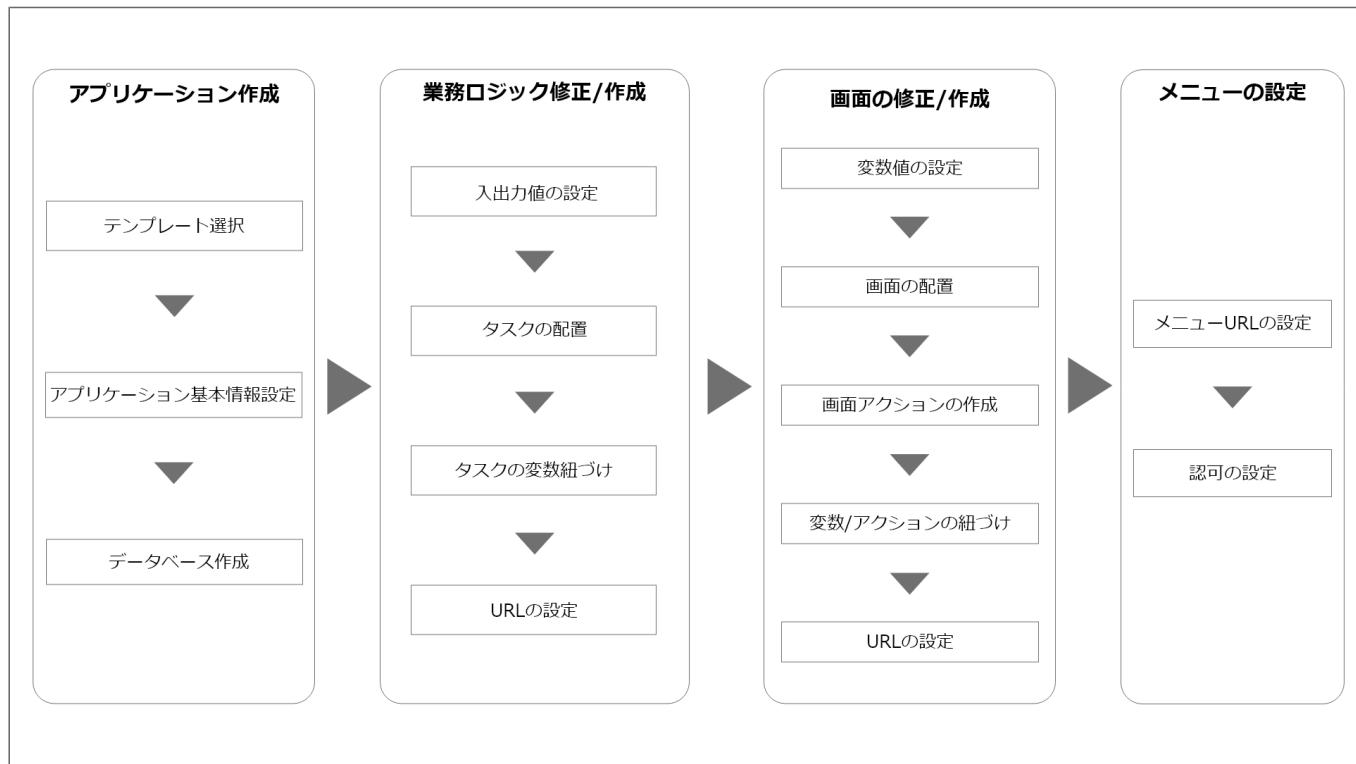
本チュートリアルでは「[環境について](#)」に記載の環境を前提に、機能間の遷移順序を記述しています。

詳しくは上記リンクからご確認ください。

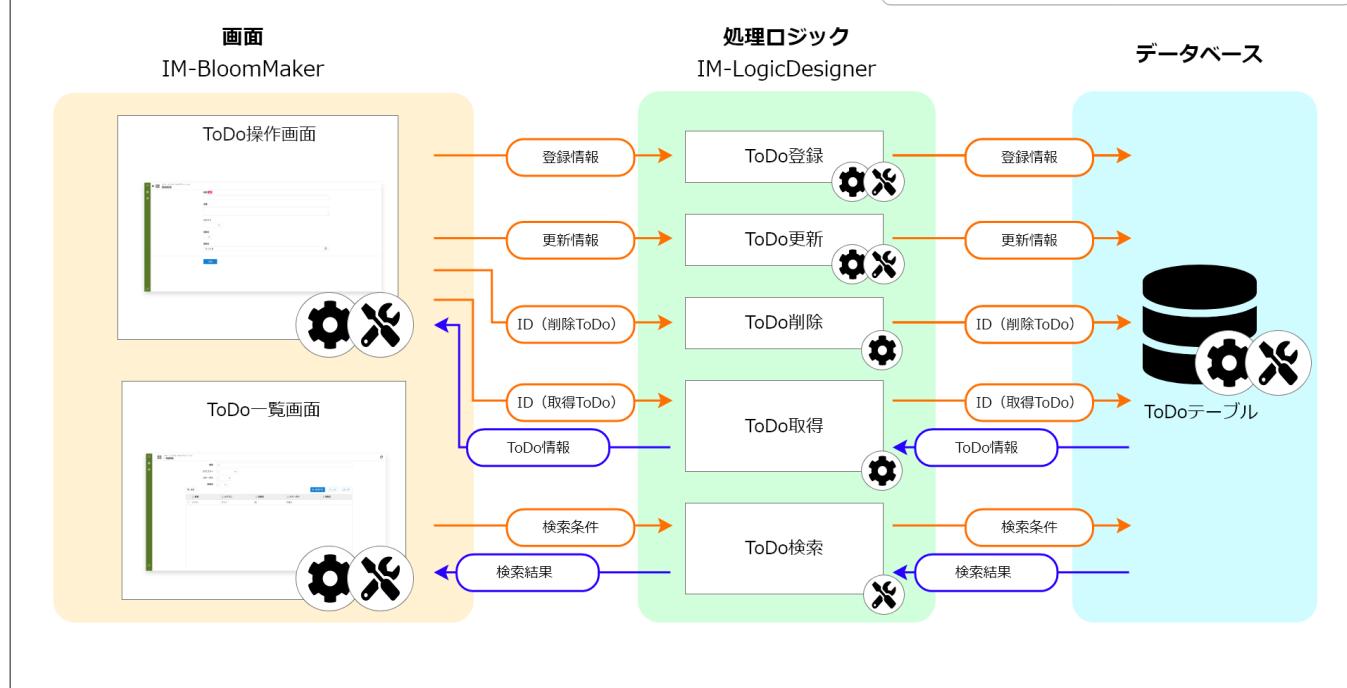
作成の流れ

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。



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・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

【デュアル】ToDoアプリケーション
登録画面

「*」は必須項目です。

概要*

詳細

カテゴリ

重要度

期限日

年 / 月 / 日

登録

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

【デュアル】ToDoアプリケーション
編集画面

「*」は必須項目です。

概要*

テスト1

詳細

テスト1です

カテゴリ

タスク

重要度

高

ステータス

未着手

期限日

2023/10/27

更新 削除

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

概要 *
テスト1

詳細
テスト1です

カテゴリ
タスク

重要度
高

ステータス
未着手

期限日
2023/10/27

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。

| 編集 | 概要 ▾ | カテゴリ ▾ | 重要度 ▾ | ステータス ▾ | 期限日 ▾ | 詳細 |
|----|------|--------|-------|---------|------------|----|
| | テスト1 | タスク | 高 | 未着手 | 2023/10/27 | |
| | テスト2 | タスク | 中 | 未着手 | 2023/10/27 | |

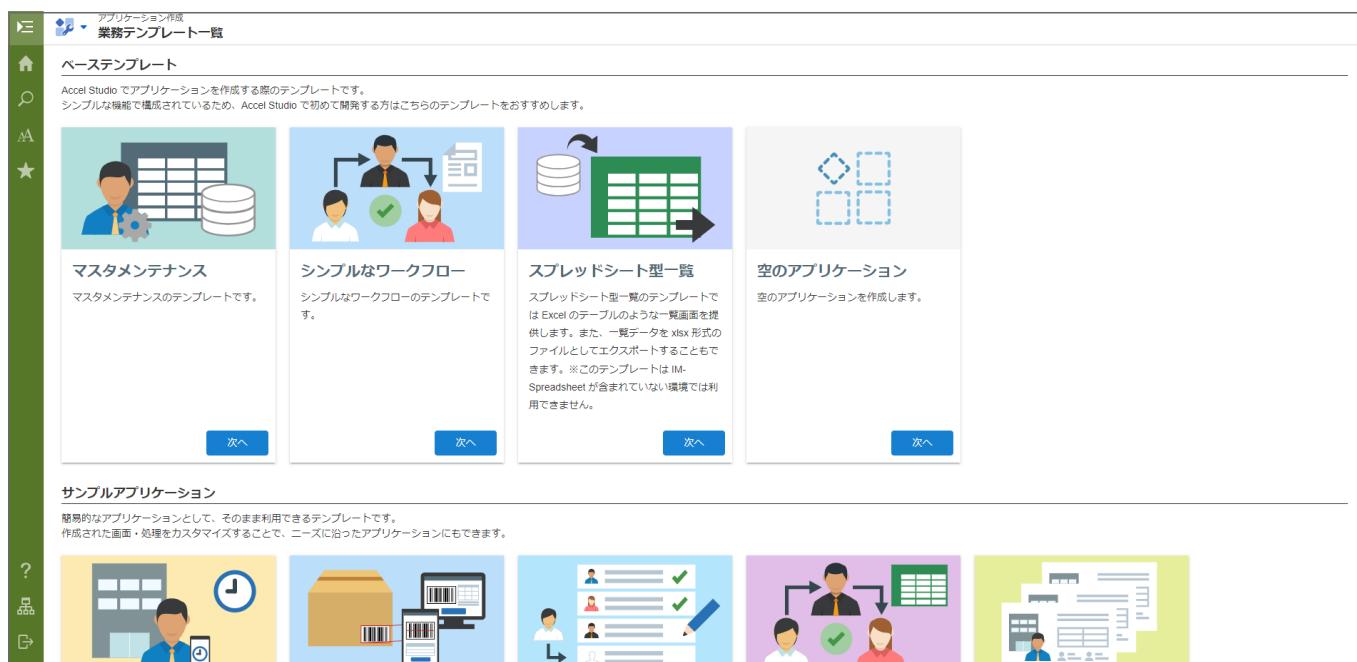
テンプレートからアプリケーションのベースを作る

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。

「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。



The screenshot shows the 'Application Creation Wizard' in Accel Studio. On the left is a vertical toolbar with icons for Home, Search, and Favorites. The main area has a header 'Application Creation Wizard' and 'Business Template Catalog'. A sidebar on the left lists 'Template Categories' like 'Customer Management', 'Order Management', etc., and 'Recent Templates'. The main content area is titled 'Base Template'.

ベーステンプレート

Accel Studio でアプリケーションを作成する際のテンプレートです。シンプルな機能で構成されているため、Accel Studio で初めて開発する方はこちらのテンプレートをおすすめします。

| | | | | | | | |
|--|---|--|---|--|--|---|--|
|  | マスタメンテナンス マスタメンテナンスのテンプレートです。 |  | シンプルなワークフロー シンプルなワークフローのテンプレートです。 |  | スプレッドシート型一覧 スプレッドシート型一覧のテンプレートでは Excel のテーブルのような一覧画面を提供します。また、一覧データを xlsx 形式のファイルとしてエクスポートすることができます。※このテンプレートは IM-Spreadsheet が含まれていない環境では利用できません。 |  | 空のアプリケーション 空のアプリケーションを作成します。 |
| 次へ | 次へ | 次へ | 次へ | 次へ | 次へ | 次へ | |

サンプルアプリケーション

簡単的なアプリケーションとして、そのまま利用できるテンプレートです。作成された画面・処理をカスタマイズすることで、ニーズに沿ったアプリケーションにもできます。

| | | | | |
|---|---|---|--|---|
|  |  |  |  |  |
|---|---|---|--|---|

「作成方法の選択」ダイアログが表示されますので、「新しくエンティティを作成」を選択し、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。

作成方法の選択

マスタメンテナンスの作成方法を選択してください。



「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。

The screenshot shows the 'Master Maintenance - New Creation' screen. On the left, there's a sidebar with 'Application Information' and 'Definition of handled data'. The main area has a 'Application Information' section with a note about the application ID and a 'Description' section.

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。

エンティティについての情報を設定します。

「詳細に設定する」はOFFのままとしてください。

エンティティ名、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 |
|-----------------|------------------------------|----|
| エンティティ名 | 【チュートリアル_エンティティ】ToDoアプリケーション | |
| エンティティに紐づくテーブル名 | tutorial_todo_app | |

The screenshot shows the 'マスタメンテナンス・新規作成' (Master Maintenance/ New Creation) screen. The main area is titled '扱うデータの定義' (Definition of handled data). It includes a note about creating a table and columns for the application. Below this is a '基本情報' (Basic Information) section with a toggle switch for '詳細に設定する' (Set in detail). The 'エンティティ名' (Entity Name) is set to '【チュートリアル_エンティティ】ToDoアプリケーション'. The 'エンティティに紐づくテーブル名' (Table name associated with the entity) is 'tutorial_todo_app'. A '項目設定' (Item Settings) section shows a table with columns for '表示名' (Display Name), '物理名' (Physical Name), 'データ型' (Data Type), and '表示形式' (Display Format). The '表示形式' column has a dropdown menu with '最大桁数' (Maximum Number of Digits) set to 100. Other sections include '明細データの利用指定' (Specify the use of detailed data) and 'その他情報の指定' (Specify other information).

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。

「項目設定」 - 「+追加」を選択してください。新規項目欄が追加されます。

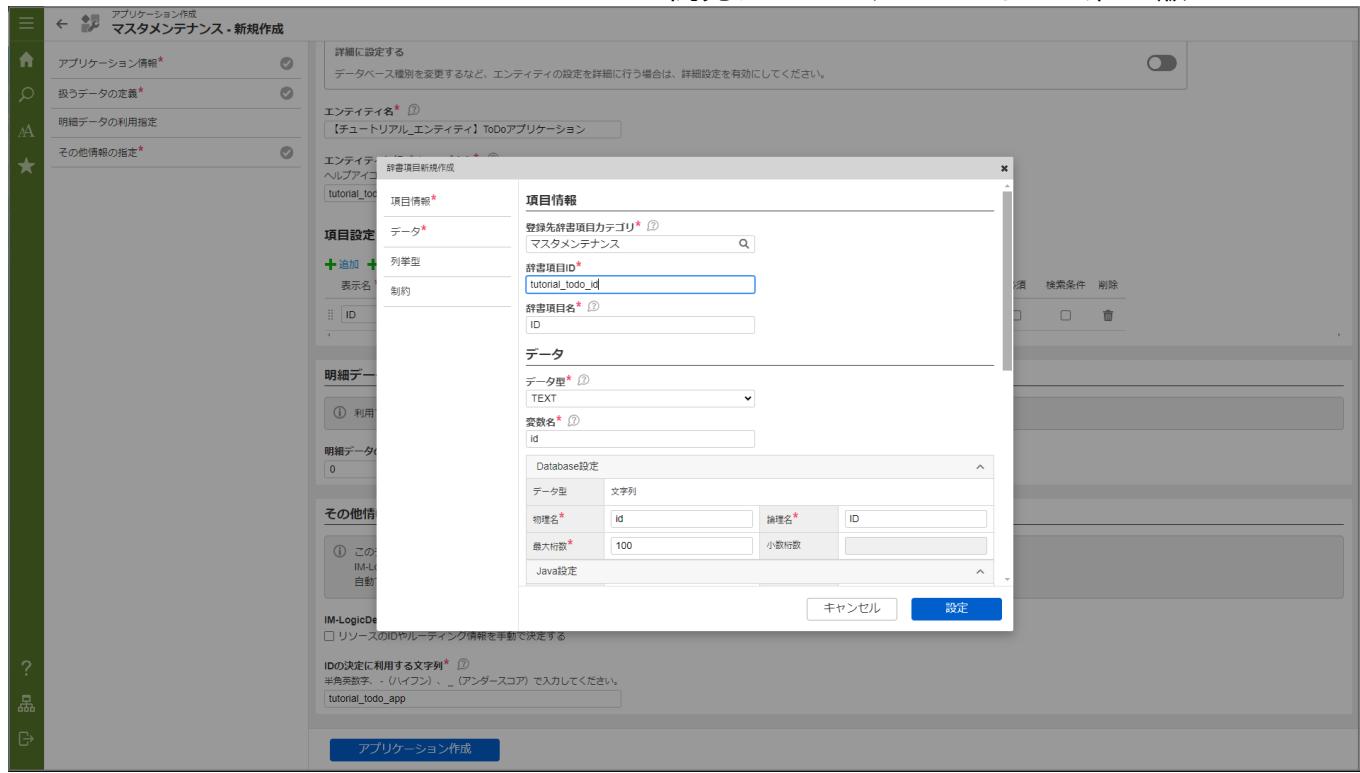
The screenshot shows the '項目設定' (Item Settings) screen. A red box highlights the '追加' (Add) button in the '既存の辞書項目から選択' (Select from existing dictionary items) section. The table below has columns for '表示名' (Display Name), '物理名' (Physical Name), 'データ型' (Data Type), and '表示形式' (Display Format). The '表示形式' column has a dropdown menu with '最大桁数' (Maximum Number of Digits) set to 100.

「表示名」に「ID」、「物理名」に「id」、「データ型」は「文字列」と設定します。

「表示形式」は「テキストボックス」を選択し、「最大桁数」に「100」と設定してください。

「詳細」の歯車アイコンをクリックして「辞書項目新規作成」ダイアログを表示し、辞書項目IDを入力しましょう。

| ラベル | 入力値 | 備考 |
|-------------|------------------|--|
| 登録先辞書項目カテゴリ | マスタメンテナンス | 任意のカテゴリで構いません。本チュートリアルでは「マスタメンテナンス」カテゴリを使用しています。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、テンプレートから自動で設定される「マスタメンテナンス」カテゴリを使用しています。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。

「辞書項目ID」の設定が完了したら「設定」ボタンをクリックしてください。

最後に、「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ー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にのみチェックを入れます。

検索条件

| 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_todo_id | ID | TEXT | id | 100 | テキストボックス | <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"/> |
| tutorial_todo_detail | 詳細 | TEXT | detail | 5000 | 複数行テキストボックス | | |
| tutorial_todo_category | カテゴリ | TEXT | category | 1000 | プルダウン | <input type="checkbox"/> | |
| 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"/> | |

アプリケーション登録

マスターデータ登録・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*

詳細に設定する
データベース種別を変更するなど、エンティティの設定を詳細に行う場合は、詳細設定を有効にしてください。

エンティティ名*

エンティティに紐づくテーブル名*

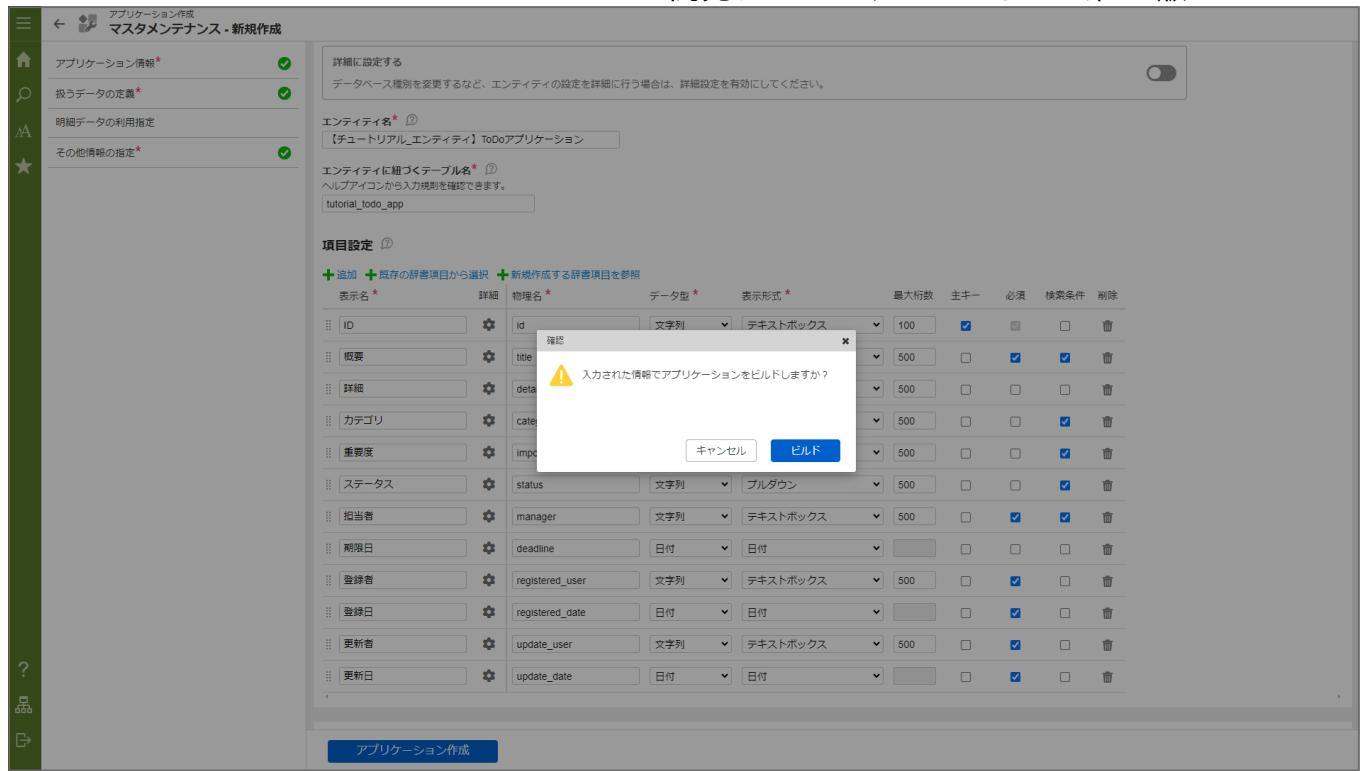
項目設定

+ 追加 + 基存の辞書項目から選択 + 新規作成する辞書項目を参照

| 表示名 * | 詳細 | 物理名 * | データ型 * | 表示形式 * | 最大桁数 | 主キー | 必須 | 検索条件 | 削除 |
|-------|--|--|--------|-------------|------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|
| ID | <input type="text" value="Id"/> | <input type="text" value="id"/> | 文字列 | テキストボックス | 100 | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 概要 | <input type="text" value="title"/> | <input type="text" value="title"/> | 文字列 | テキストボックス | 500 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 詳細 | <input type="text" value="detail"/> | <input type="text" value="detail"/> | 文字列 | 複数行テキストボックス | 500 | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| カテゴリ | <input type="text" value="category"/> | <input type="text" value="category"/> | 文字列 | プルダウン | 500 | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 重要度 | <input type="text" value="importance"/> | <input type="text" value="importance"/> | 文字列 | プルダウン | 500 | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| ステータス | <input type="text" value="status"/> | <input type="text" value="status"/> | 文字列 | プルダウン | 500 | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 担当者 | <input type="text" value="manager"/> | <input type="text" value="manager"/> | 文字列 | テキストボックス | 500 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 期限日 | <input type="text" value="deadline"/> | <input type="text" value="deadline"/> | 日付 | 日付 | | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 登録者 | <input type="text" value="registered_user"/> | <input type="text" value="registered_user"/> | 文字列 | テキストボックス | 500 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 登録日 | <input type="text" value="registered_date"/> | <input type="text" value="registered_date"/> | 日付 | 日付 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 更新者 | <input type="text" value="update_user"/> | <input type="text" value="update_user"/> | 文字列 | テキストボックス | 500 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| 更新日 | <input type="text" value="update_date"/> | <input type="text" value="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「ビルト」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。

| ビルドジョブID | 8hgcydrolc31c4 | 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 | 2024/09/18 16:02:45 | | |

| 出力日時 | メッセージ | 詳細 |
|---------------------|--|----|
| 2024/09/18 16:02:45 | 生成するリソースのIDを自動設定します。 | |
| 2024/09/18 16:02:46 | 一時ディレクトリに IM-LogicDesigner インポート資材を作成しました。 | |
| 2024/09/18 16:02:46 | 一時ディレクトリに IM-BloomMaker インポート資材を作成しました。 | |
| 2024/09/18 16:02:47 | 自動生成された IM-LogicDesigner 資材をインポートしました。 | |
| 2024/09/18 16:02:48 | 自動生成された IM-BloomMaker 資材をインポートしました。 | |
| 2024/09/18 16:02:48 | 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。 | |
| 2024/09/18 16:02:50 | アプリケーションの作成が完了しました。 | |
| 2024/09/18 16:02:50 | 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

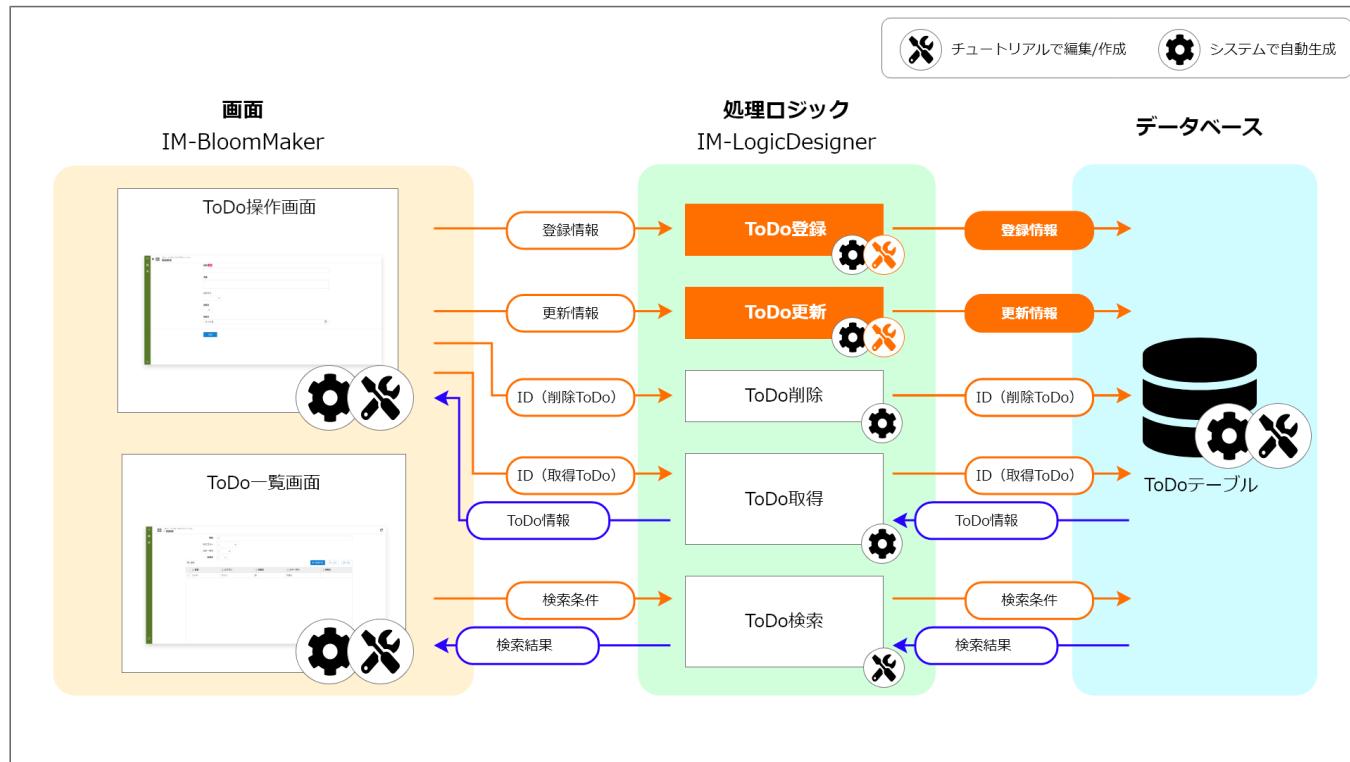
テンプレートからデータベース、IM-LogicDesigner の業務ロジック、IM-BloomMakerの画面が自動生成されました。

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ー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。

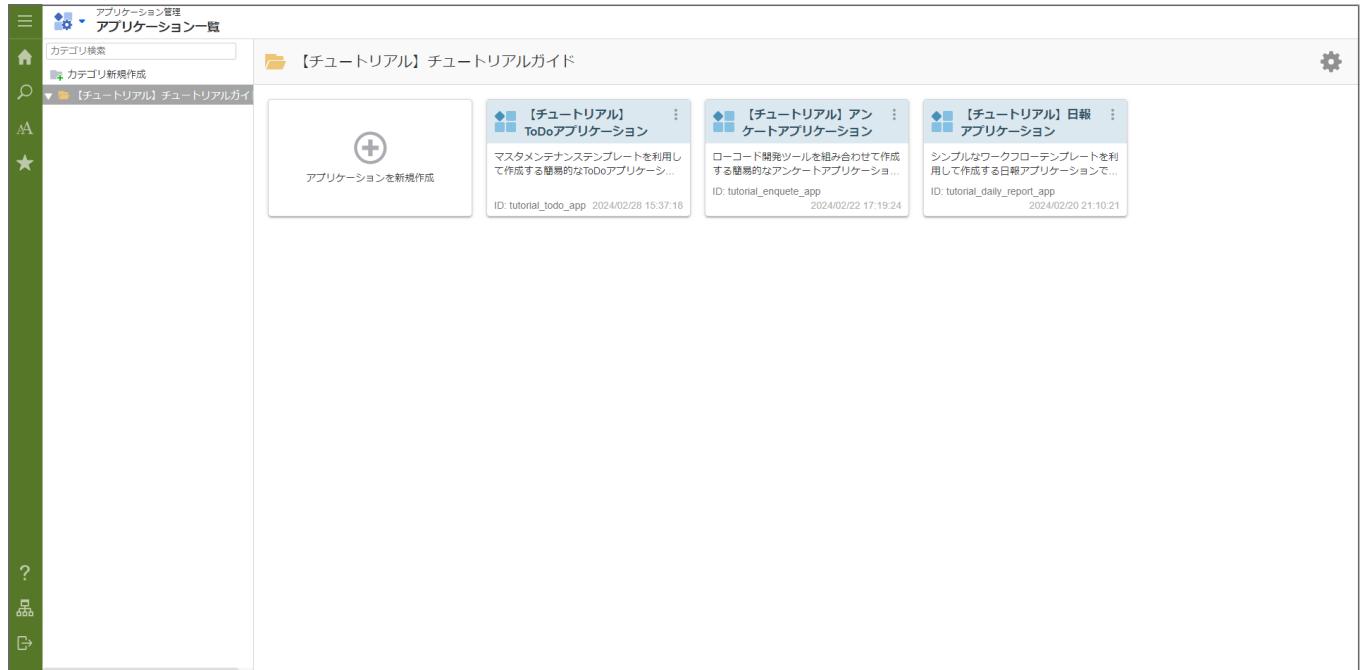
なお、マスタメンテナンステンプレートでは、詳細画面表示用の業務ロジックとタスクの削除用の業務ロジックも作成されますが、本チュートリアルでは特に修正は加えません。



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。

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。

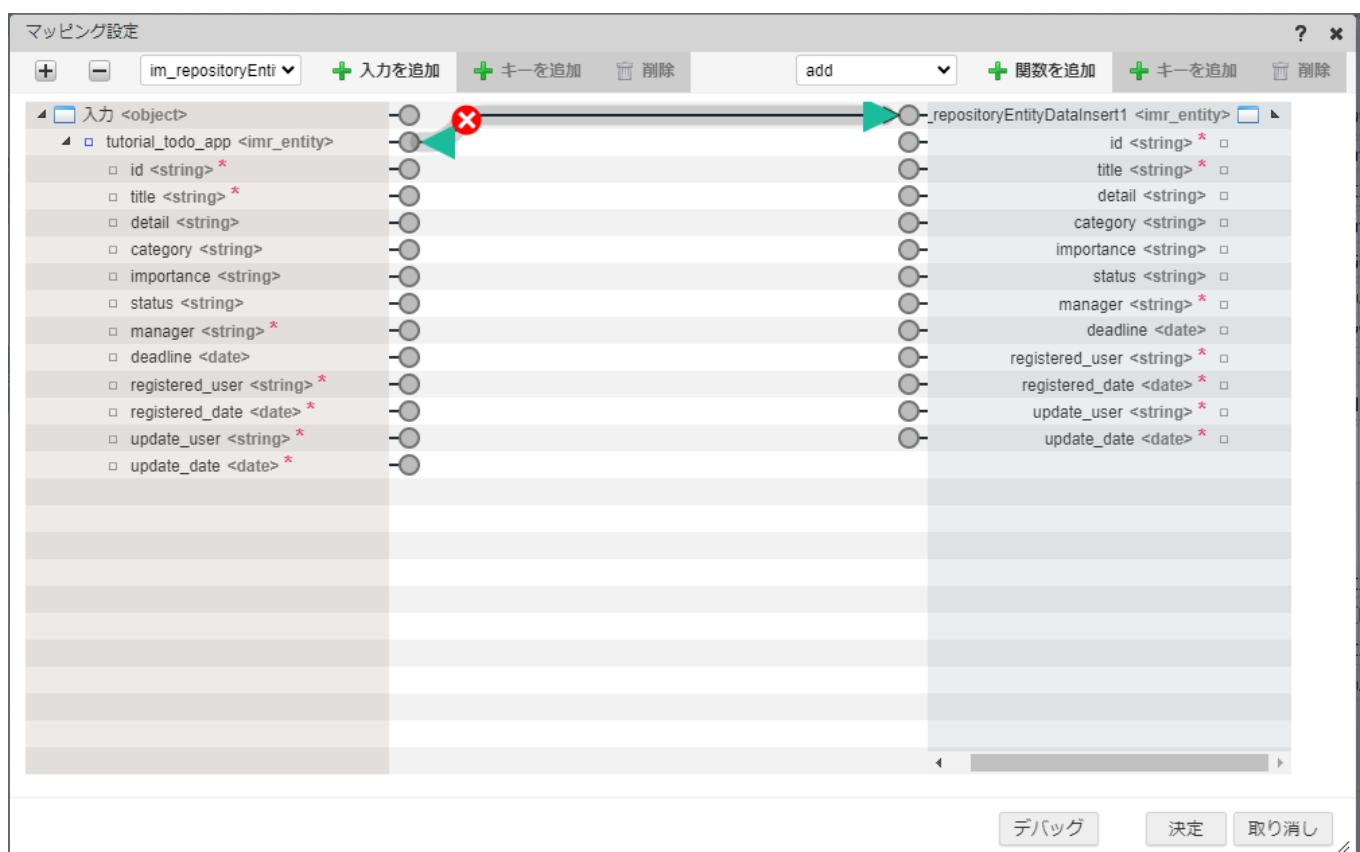
「tutorial_todo_app_entity の登録」タスクをダブルクリックし、「マッピング設定ダイアログ」を表示します。

「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。

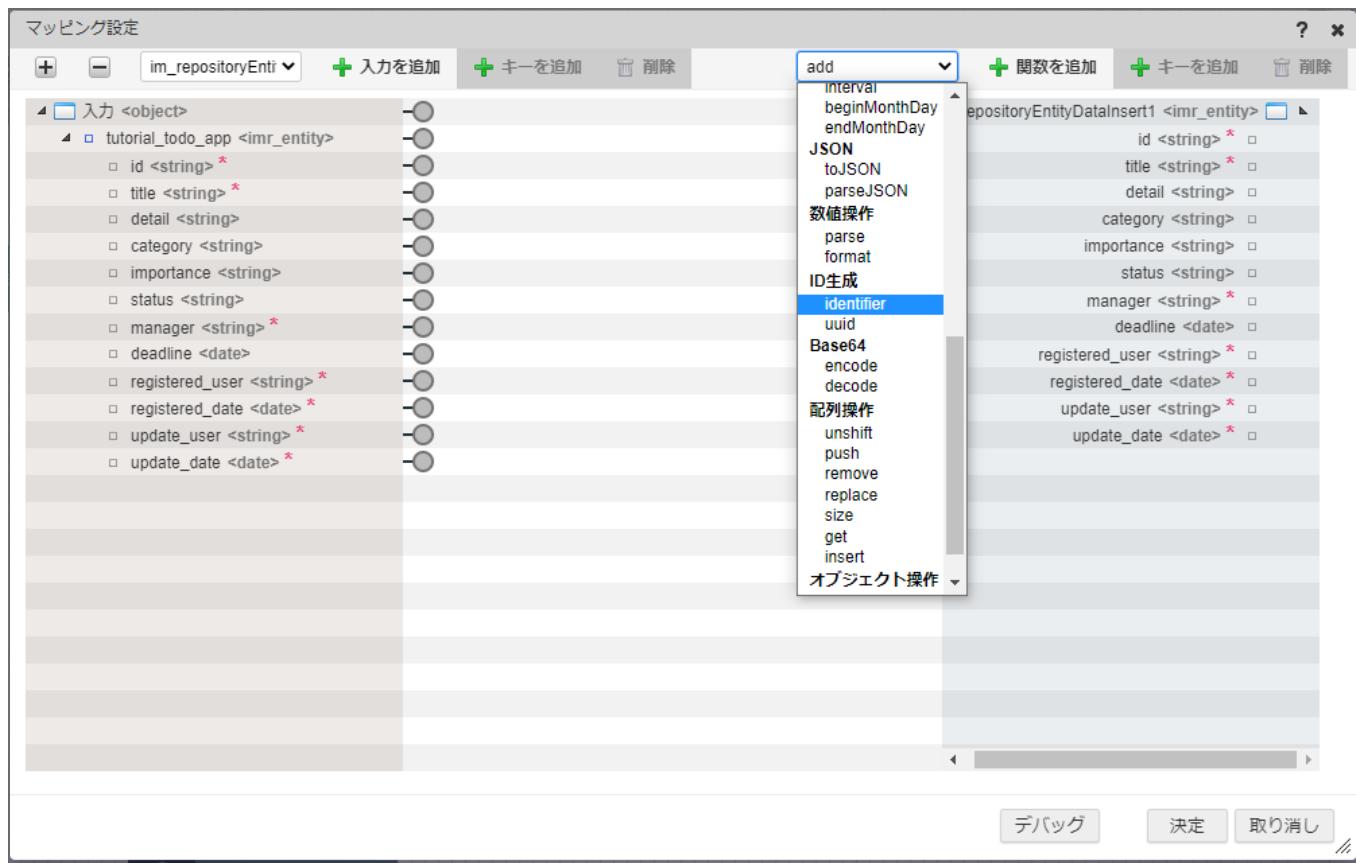
入力値 (tutorial_todo_app<imr_entity>) と出力値 (im_repositoryEntityDataInsert1<imr_entity>) を繋いでいる矢印を選択し、

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す。

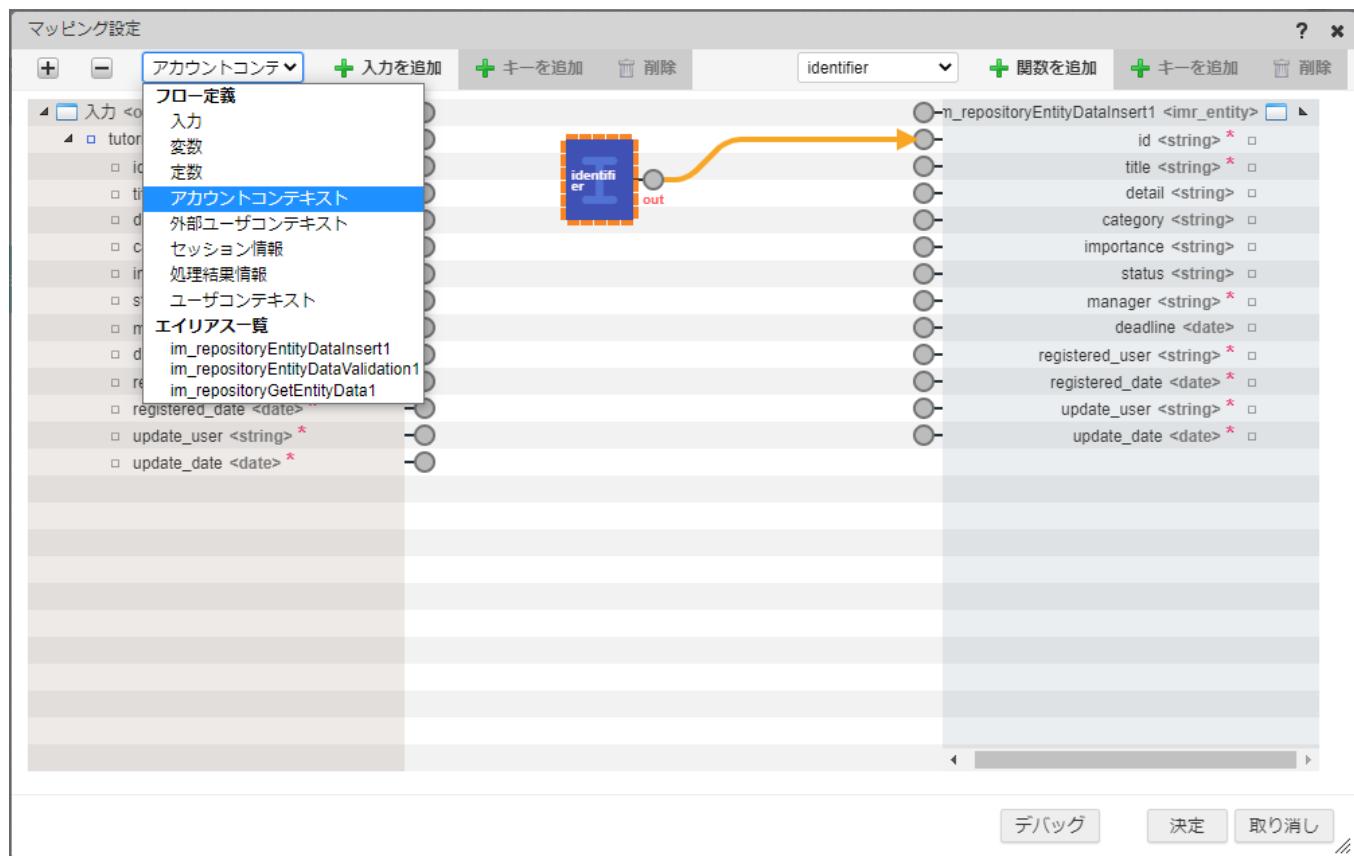
ました。

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。

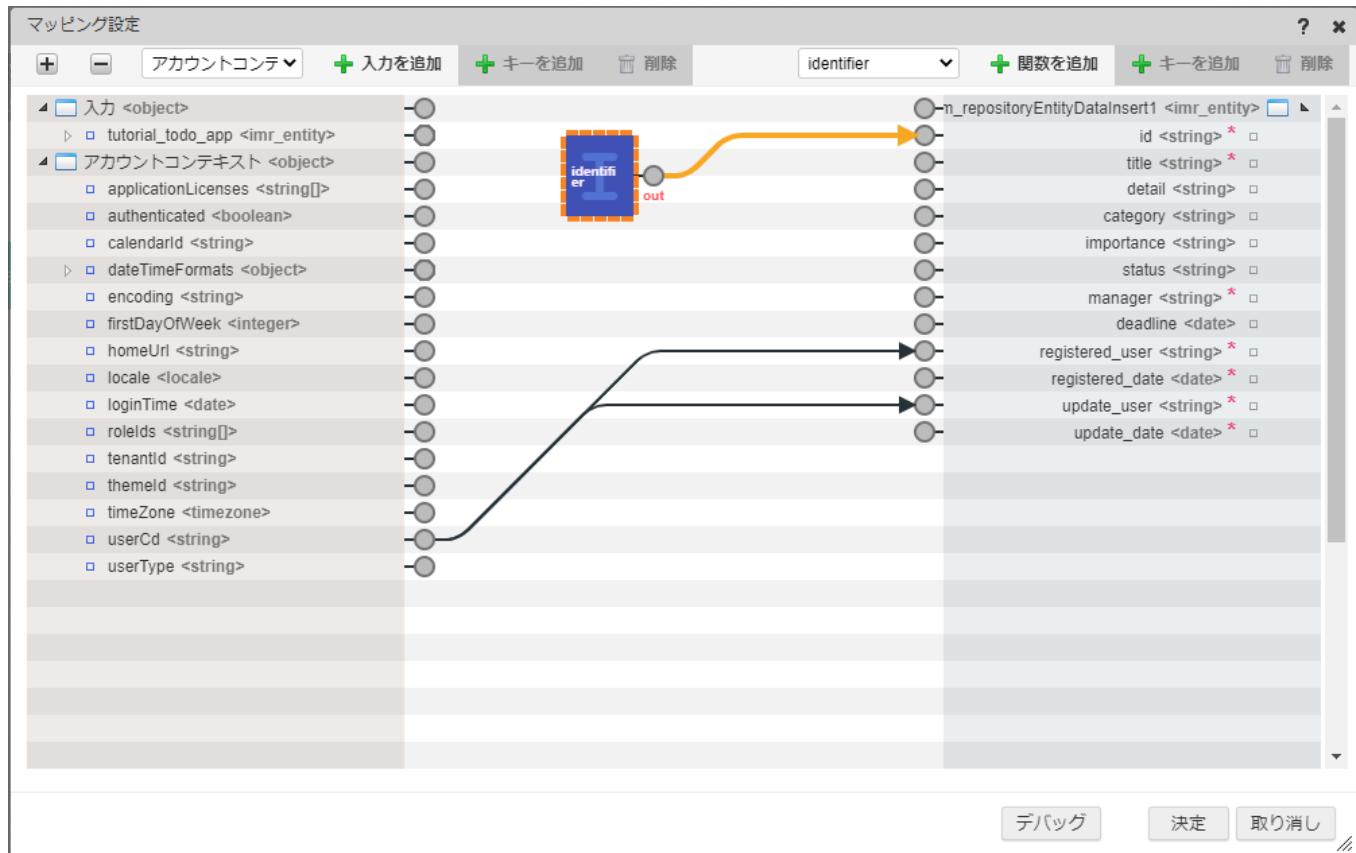
「マッピング設定」ダイアログ上部の入力値選択欄から「アカウントコンテキスト」を選択して「+入力を追加」を選択します。



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。



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。

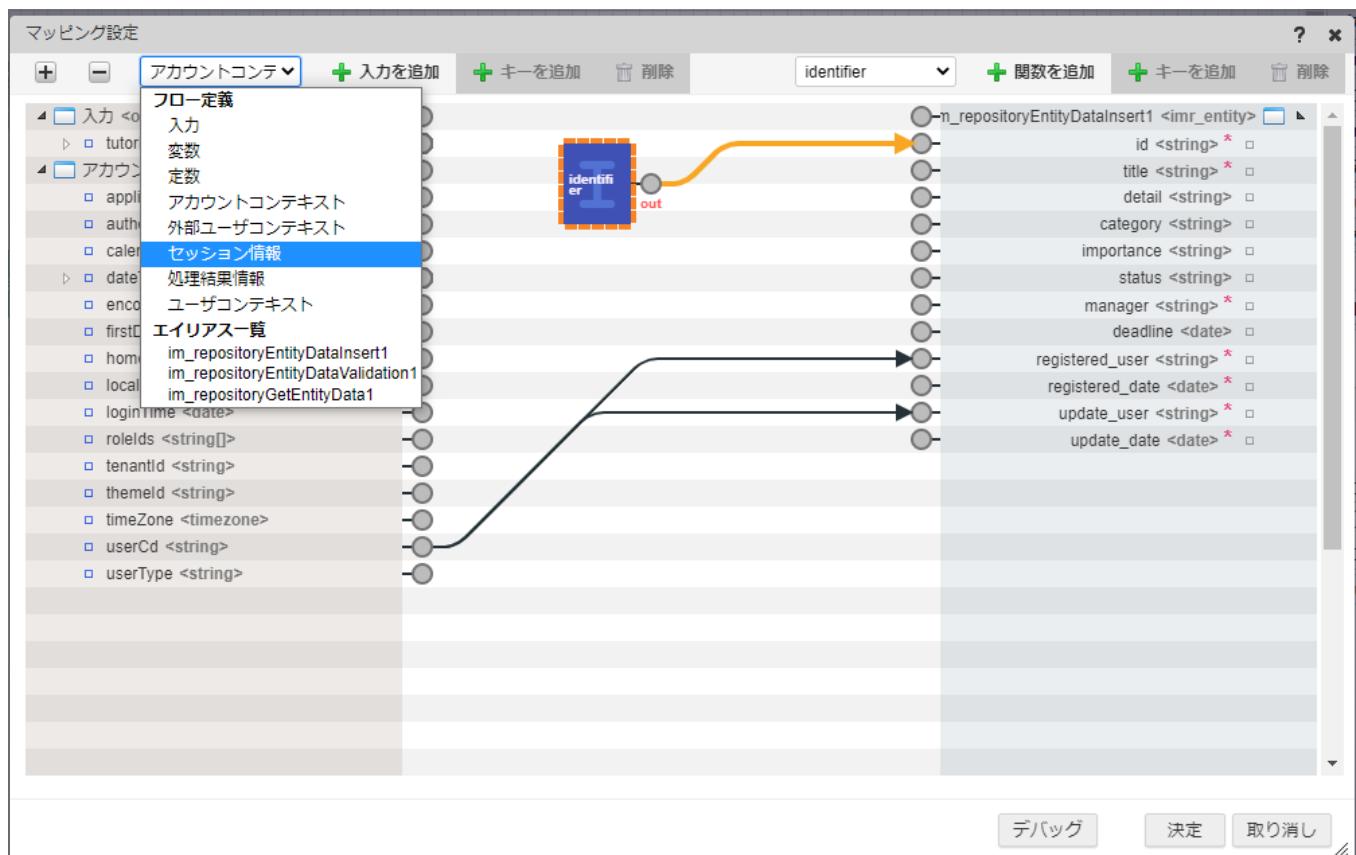


これで、操作しているユーザのユーザコードが自動で「tutorial_todo_app_entity の登録」の入力値とする設定ができました。

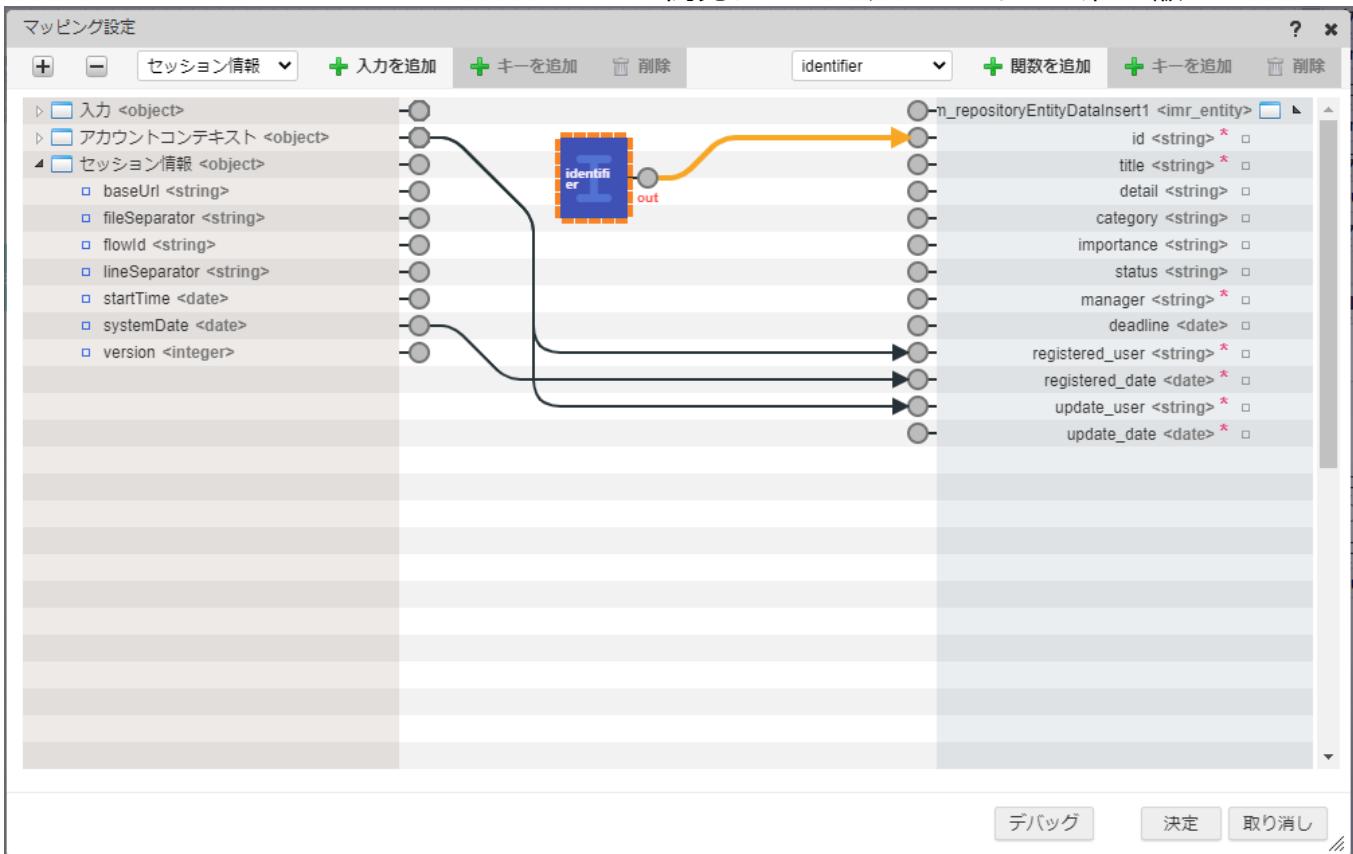
次に、登録日と更新日を操作しているタイミングの時間とします。

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。

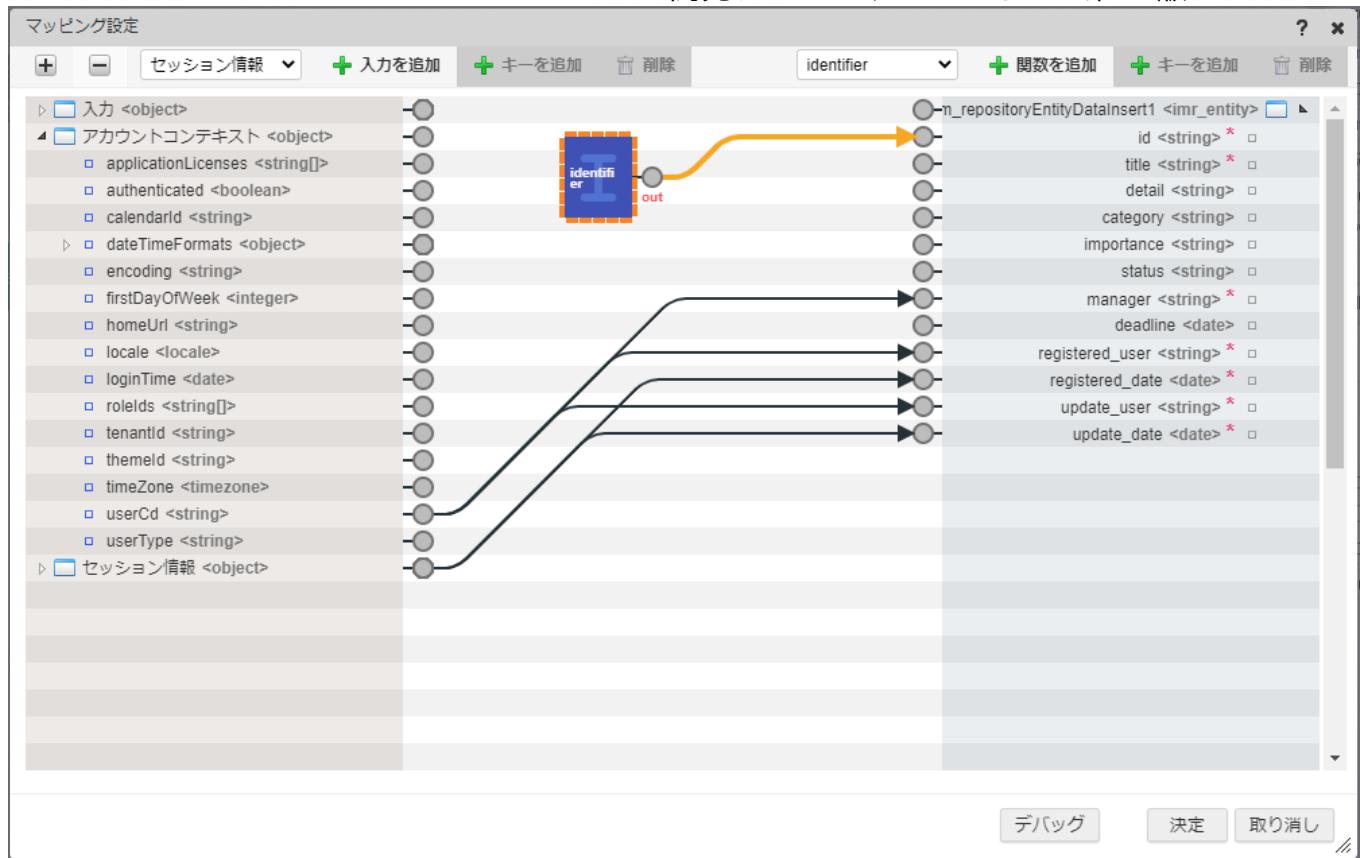


これで、登録日と更新日を操作している日時を自動で設定できます。

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。

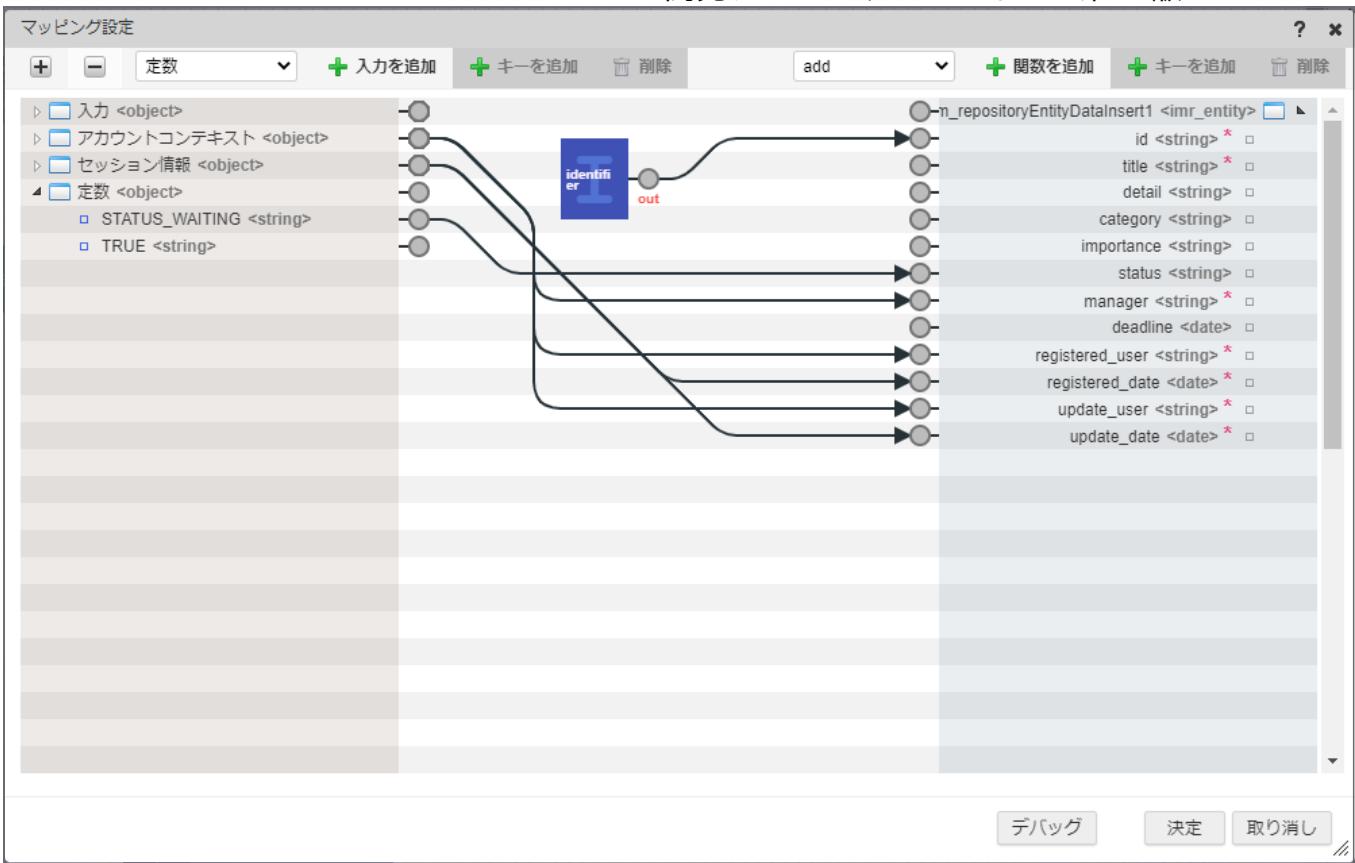
画面上部の「定数設定」をクリックして「定数設定」ダイヤログが表示されたら「+定数を追加」を選択します。

「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。

| 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 | | |

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。

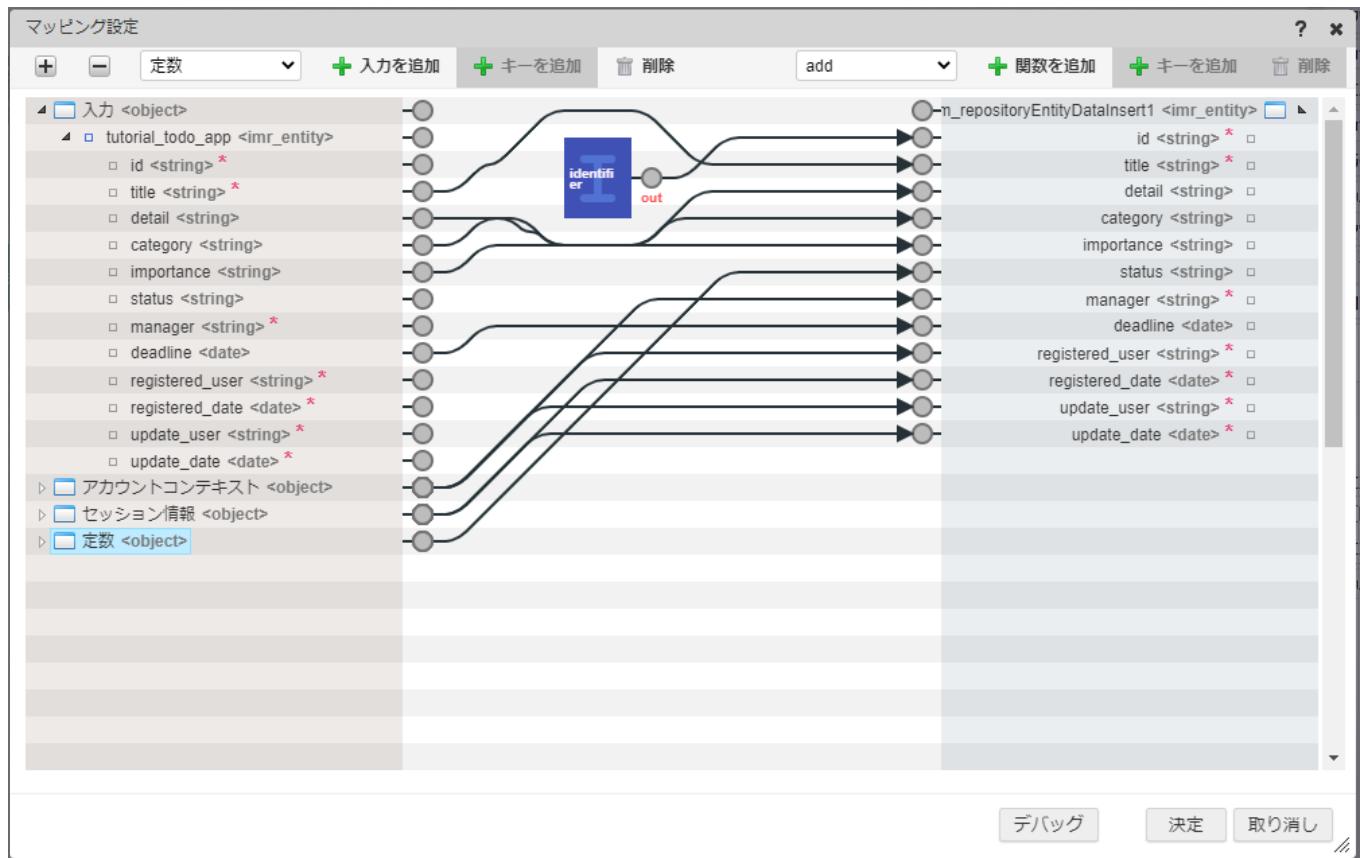
定数を追加したら、入力の「STATUS_WAITING」と「tutorial_todo_app_entity の登録」のstatusを紐づけます。



これで自動で設定したい項目の設定が完了しました。

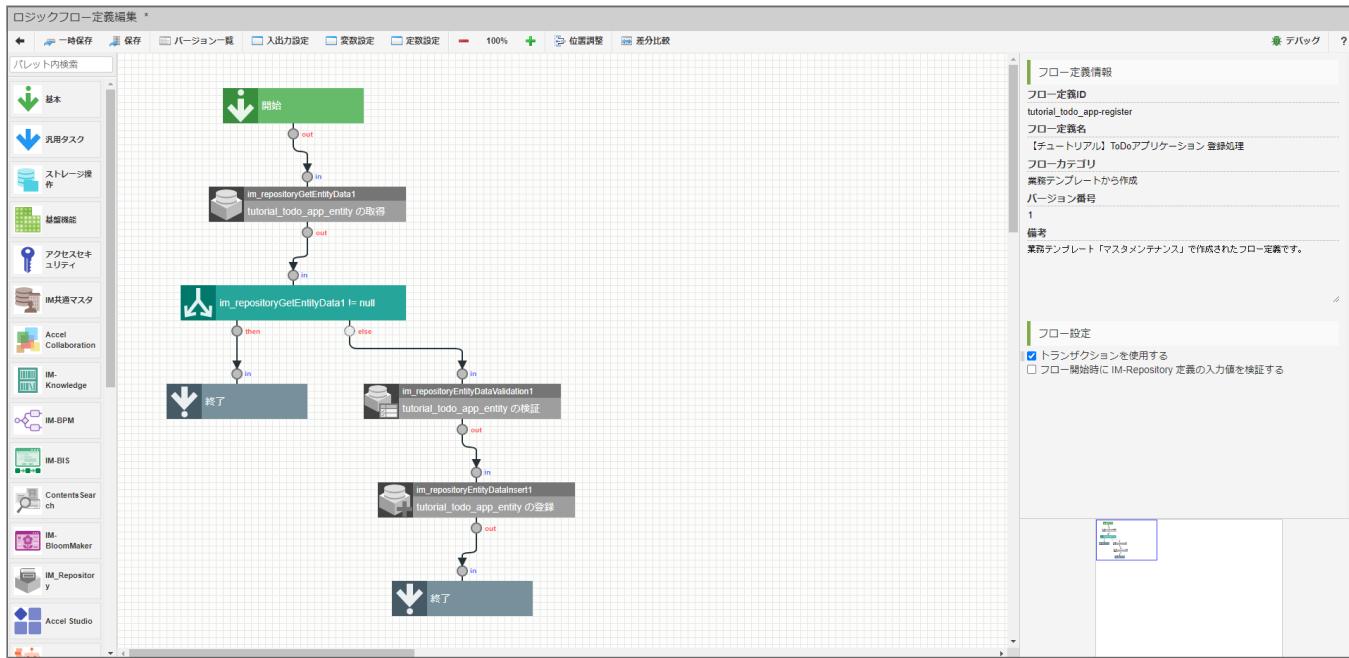
その他の変数を以下のように設定します。

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イアログ下部の「決定」をクリックして、現在の設定を保存します。



不要な処理の削除

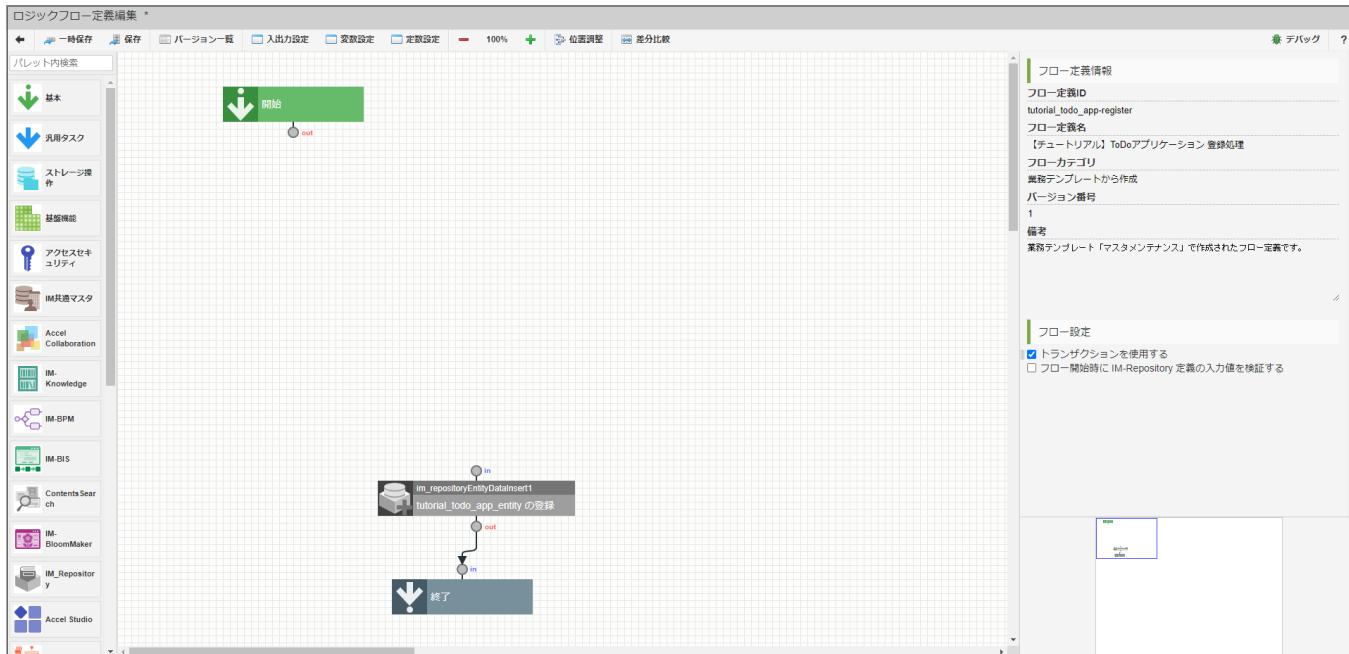
IDを自動で一意のIDが設定されるようにしたため、不要な登録時の重複チェックを削除します。



「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。

保存

| | | | | | | | | | |
|-----------|---|------|-----------------------|-----|-----------------------|----|-----------------------|---------|-----------------------|
| 保存方法 | <input checked="" type="radio"/> 新しいバージョン番号(2)を追加して保存する <input type="radio"/> 同じバージョン番号(1)に上書き保存する <input type="radio"/> フロー定義IDを新しく割り当てて複製する | | | | | | | | |
| フロー定義ID * | tutorial_todo_app-register | | | | | | | | |
| フロー定義名 * | <table border="1"> <tr> <td>標準 *</td> <td>【チュートリアル】ToDoアプリケーション</td> </tr> <tr> <td>日本語</td> <td>【チュートリアル】ToDoアプリケーション</td> </tr> <tr> <td>英語</td> <td>【チュートリアル】ToDoアプリケーション</td> </tr> <tr> <td>中国語(中国)</td> <td>【チュートリアル】ToDoアプリケーション</td> </tr> </table> | 標準 * | 【チュートリアル】ToDoアプリケーション | 日本語 | 【チュートリアル】ToDoアプリケーション | 英語 | 【チュートリアル】ToDoアプリケーション | 中国語(中国) | 【チュートリアル】ToDoアプリケーション |
| 標準 * | 【チュートリアル】ToDoアプリケーション | | | | | | | | |
| 日本語 | 【チュートリアル】ToDoアプリケーション | | | | | | | | |
| 英語 | 【チュートリアル】ToDoアプリケーション | | | | | | | | |
| 中国語(中国) | 【チュートリアル】ToDoアプリケーション | | | | | | | | |
| フローカテゴリ * | 検索/新規作成 <table border="1"> <tr> <td>ID *</td> <td>im_accel_studio</td> </tr> <tr> <td>名称</td> <td>業務テンプレートから作成</td> </tr> </table> | ID * | im_accel_studio | 名称 | 業務テンプレートから作成 | | | | |
| ID * | im_accel_studio | | | | | | | | |
| 名称 | 業務テンプレートから作成 | | | | | | | | |
| 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。 | | | | | | | | |

決定 取り消し

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。

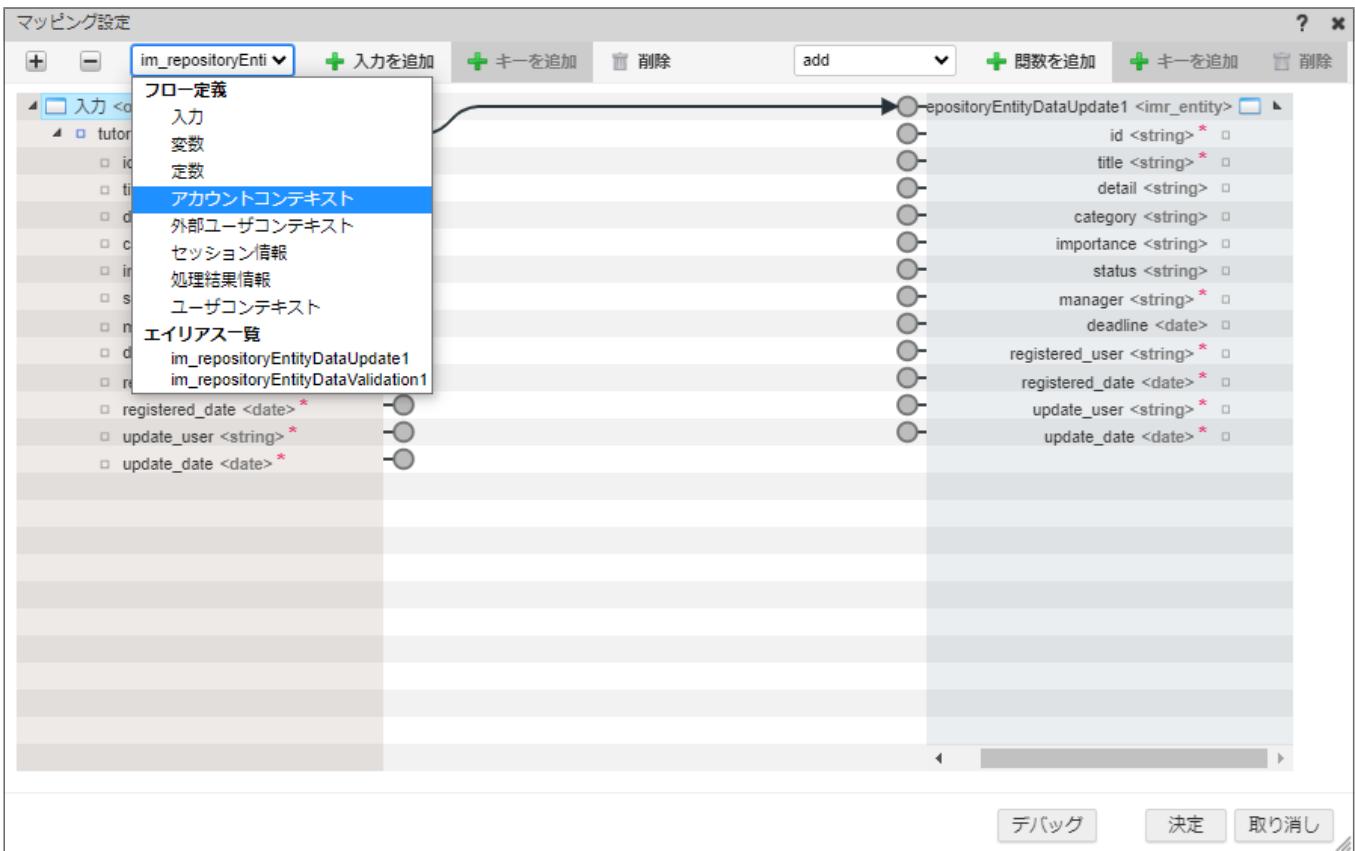


これによりオブジェクト型データ「tutorial_todo_app<imr_entity>」のプロパティと、オブジェクト型データ

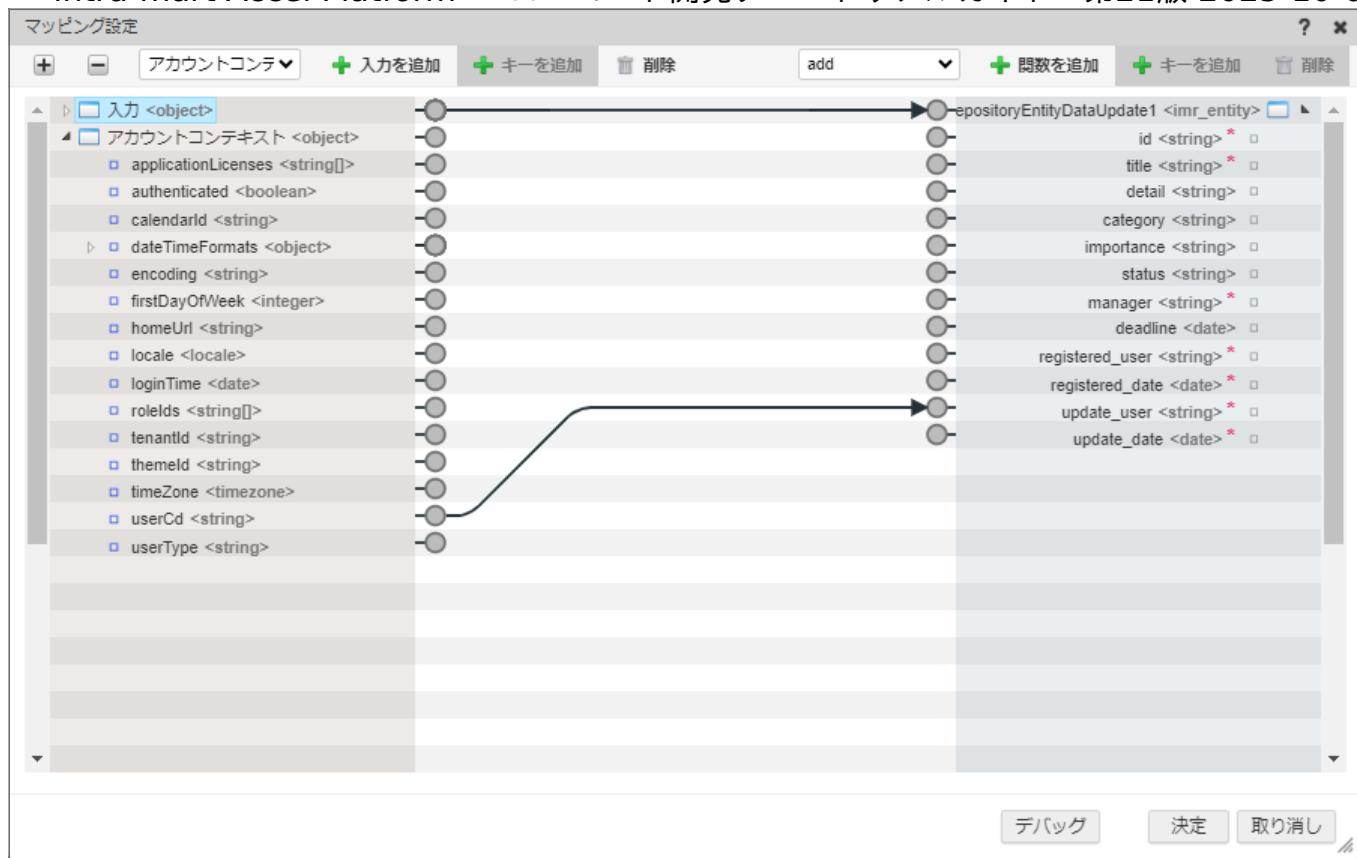
「im_repositoryEntityDataUpdate1<imr_entity>」のプロパティは紐づけられています。

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。



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。

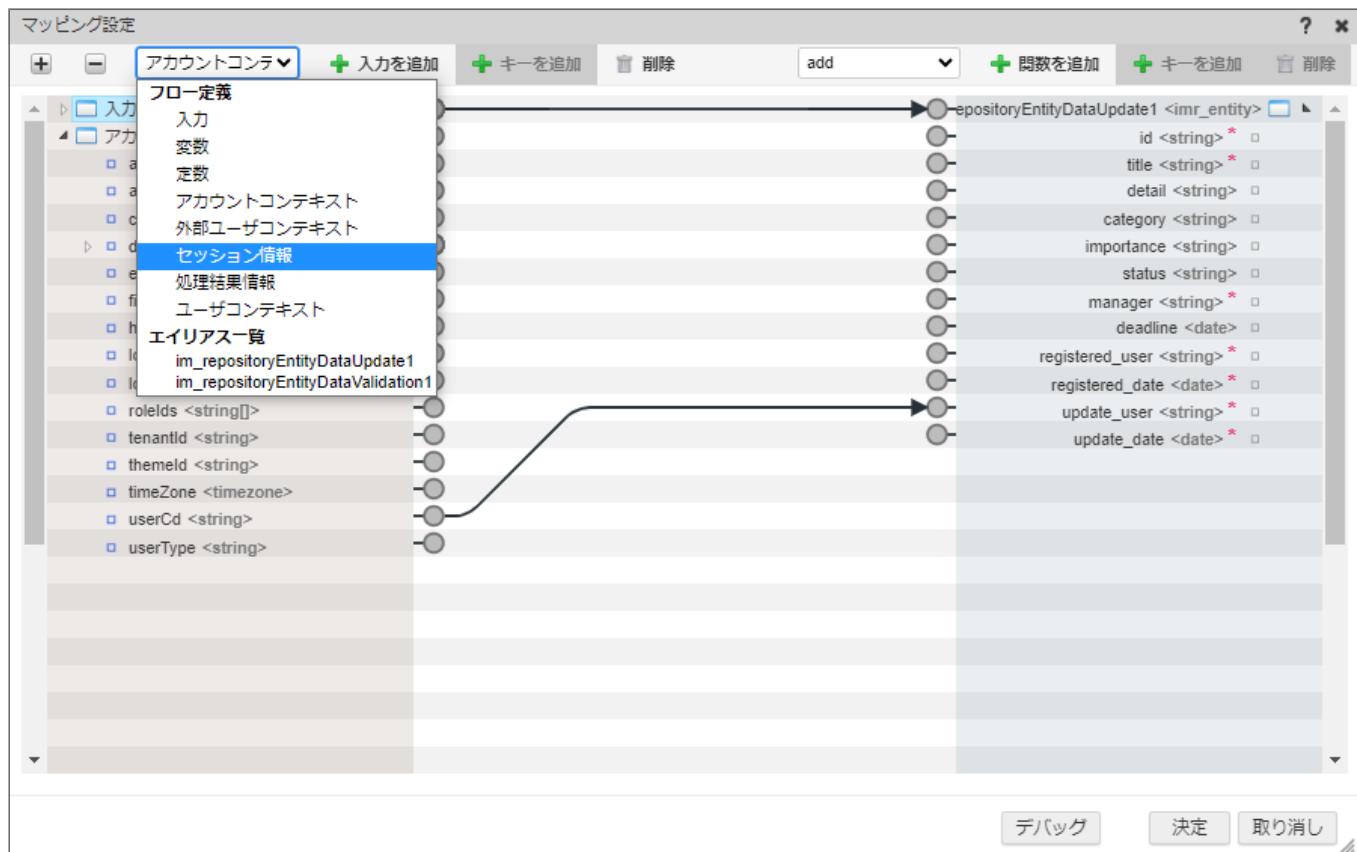


これで、編集操作を実施しているユーザのユーザコードが自動で「tutorial_todo_app_entity の更新」の入力値として設定されます。

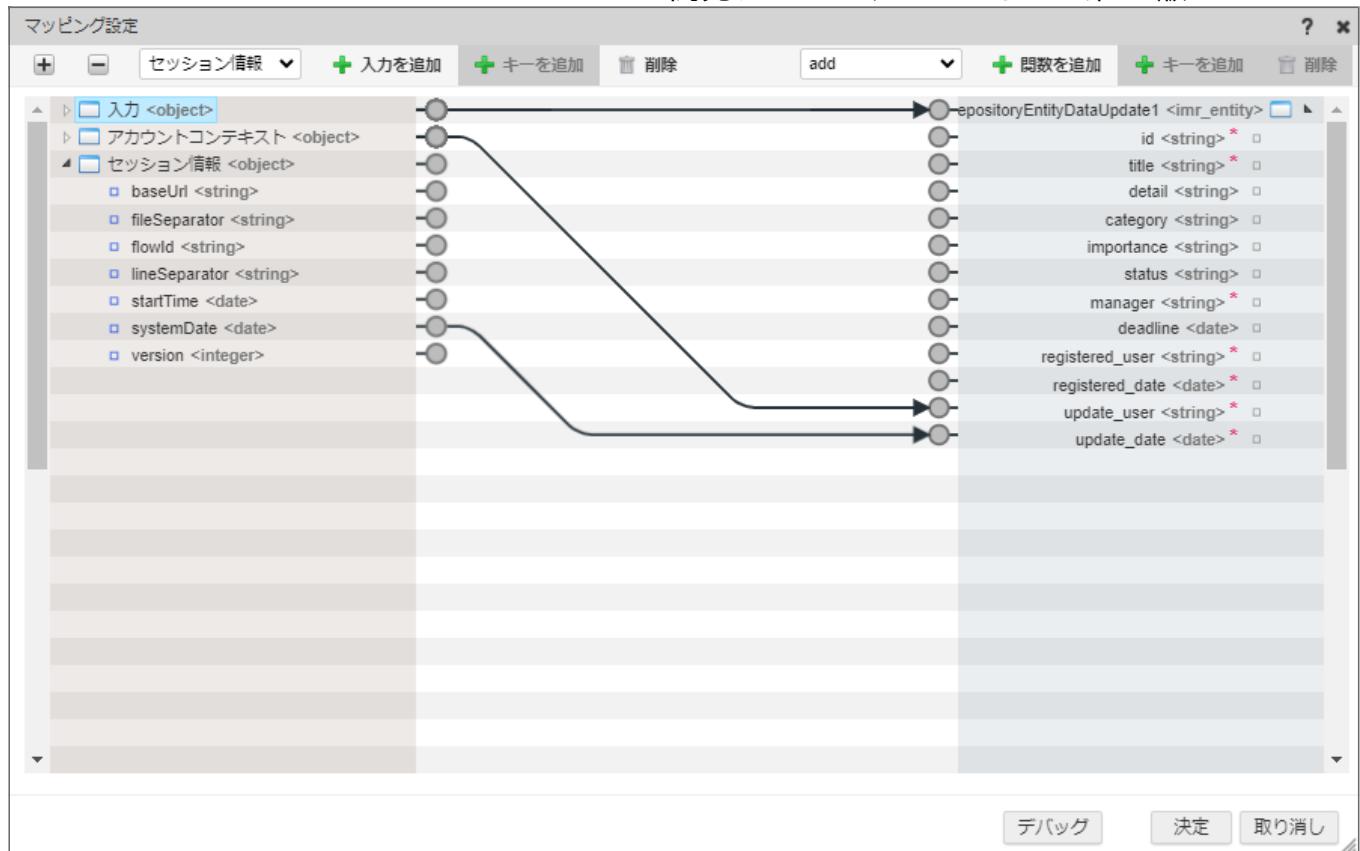
次に、更新日を編集操作しているタイミングの時間とします。

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。

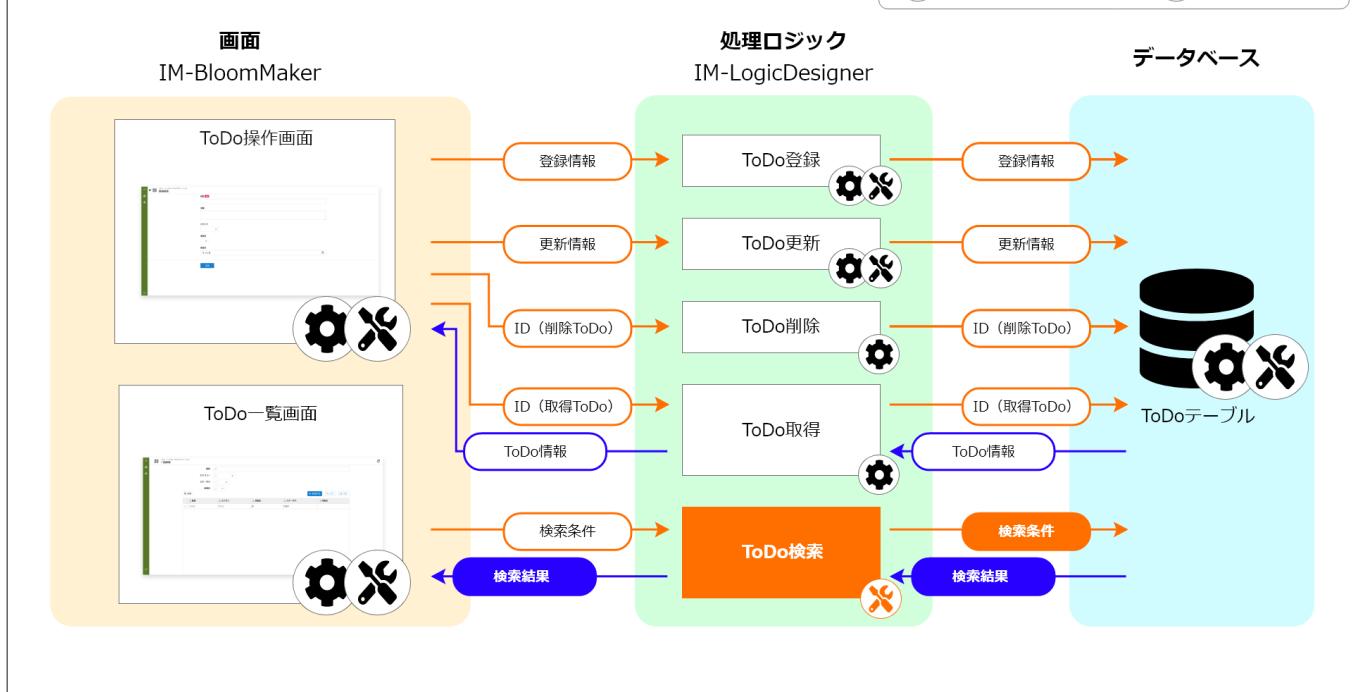
次のページでは、一覧画面での検索処理を作成します。

一覧での検索処理の確認

マスタメンテナンステンプレートは、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。

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。

The screenshot shows the 'Application Management' interface. In the top navigation bar, the 'Entity Definition' tab is highlighted. The main content area displays the entity definition for '【チュートリアル_エンティティ】ToDoアプリケーション'. On the right side, there are various search and filter options, including a date range from 'tenant 2024/02/28 16:21:53' to 'tenant 2024/02/28 16:21:53'. The sidebar on the left contains icons for Home, Search, Entity Definition, and Help.

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。

The screenshot shows the 'Entity Definition' configuration screen for the '【チュートリアル_エンティティ】ToDo' entity. The 'Data Definition' tab is selected. A green message bar at the top states 'エンティティとテーブルの情報は一致しています' (Entity and table information are consistent). Below this, there are sections for 'ログ出力設定' (Log output settings) and 'テーブル情報' (Table information), which includes a dropdown for 'データベース種別' (Database type) set to 'テナントデータベース' (Tenant Database) and 'テーブル名' (Table name) set to 'tutorial_todo_app'. The main table 'テーブル定義' (Table Definition) lists 11 columns with their respective properties and constraints. A 'Edit' button is visible next to the table header.

| No. | 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"/> |

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

エンティティ項目 条件

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

1 テーブル定義の設定 2 検索条件の設定 3 楽観的排他制御の設定 4 更新内容確認

検索条件の設定

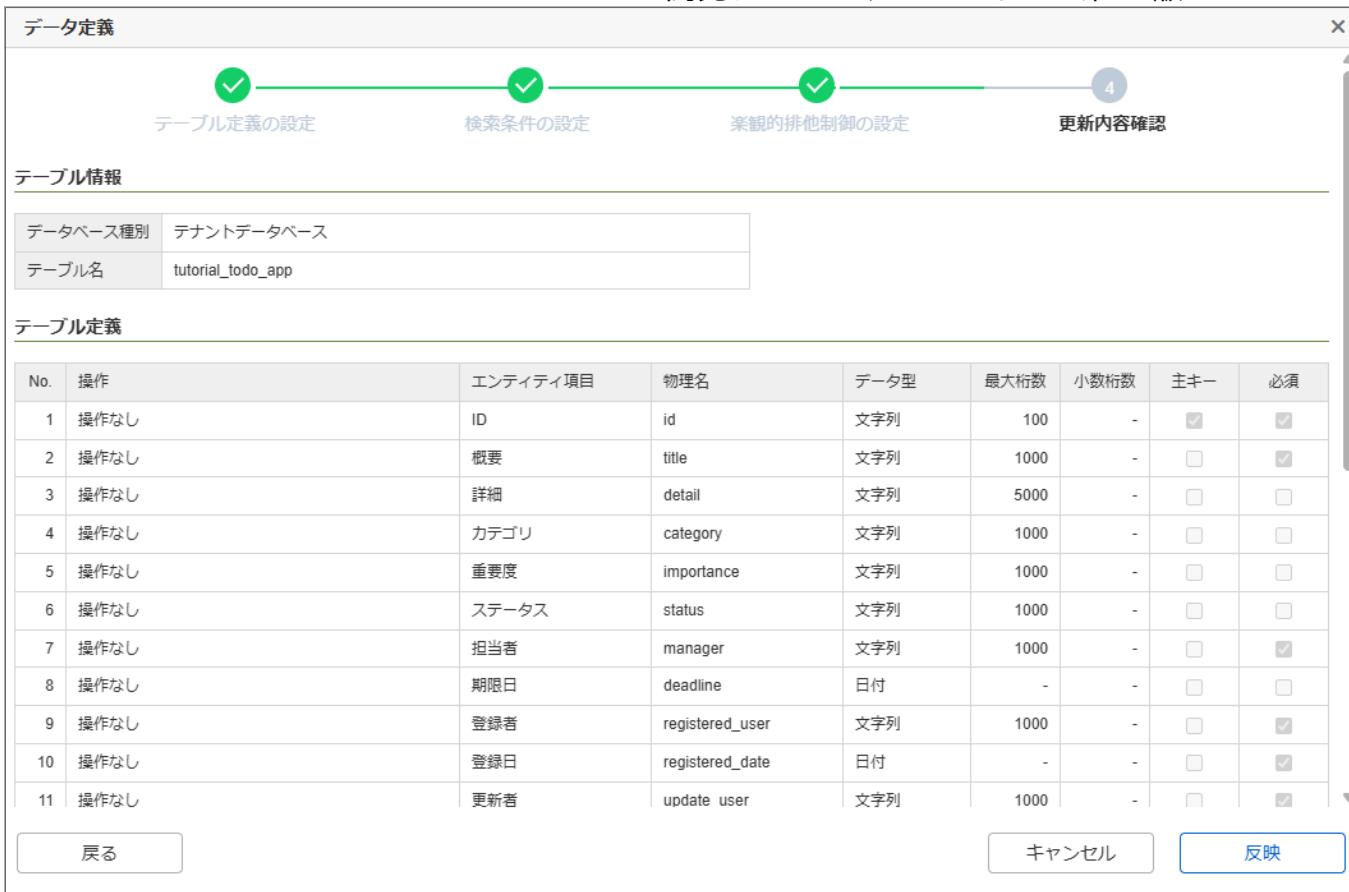
条件を追加

| 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**

検索条件を設定したら「次へ」ボタンをクリックし、「楽観的排他制御を利用しない」を選択します。

すべての設定が完了したら「次へ」ボタンをクリックして、内容を確認してください。



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

カタログ新規追加 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

選択対象エンティティに追く項目を表示する

関連一覧 データ定義 リグ出力設定

データ定義 (デュアルアソリューション) エンティティ

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 デナントデータベース

テーブル名 tutorial_todo_app

定義情報

編集

テーブル定義

| No. | 用途 | 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 checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 3 | 詳細 | | detail | 差分なし | 文字列 | 5000 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 4 | カテゴリ | | category | 差分なし | 文字列 | 1000 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 5 | 重要度 | | importance | 差分なし | 文字列 | 1000 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 6 | ステータス | | status | 差分なし | 文字列 | 1000 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 7 | 担当者 | | manager | 差分なし | 文字列 | 1000 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 8 | 期限日 | | deadline | 差分なし | 日付 | - | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 9 | 登録者 | | registered_user | 差分なし | 文字列 | 1000 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 10 | 登録日 | | registered_date | 差分なし | 日付 | - | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 11 | 更新者 | | update_user | 差分なし | 文字列 | 1000 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 12 | 更新日 | | update_date | 差分なし | 日付 | - | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

参照項目設定

参考項目設定はありません。

影響範囲確認

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/requestParameter/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/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/requestParameter/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。

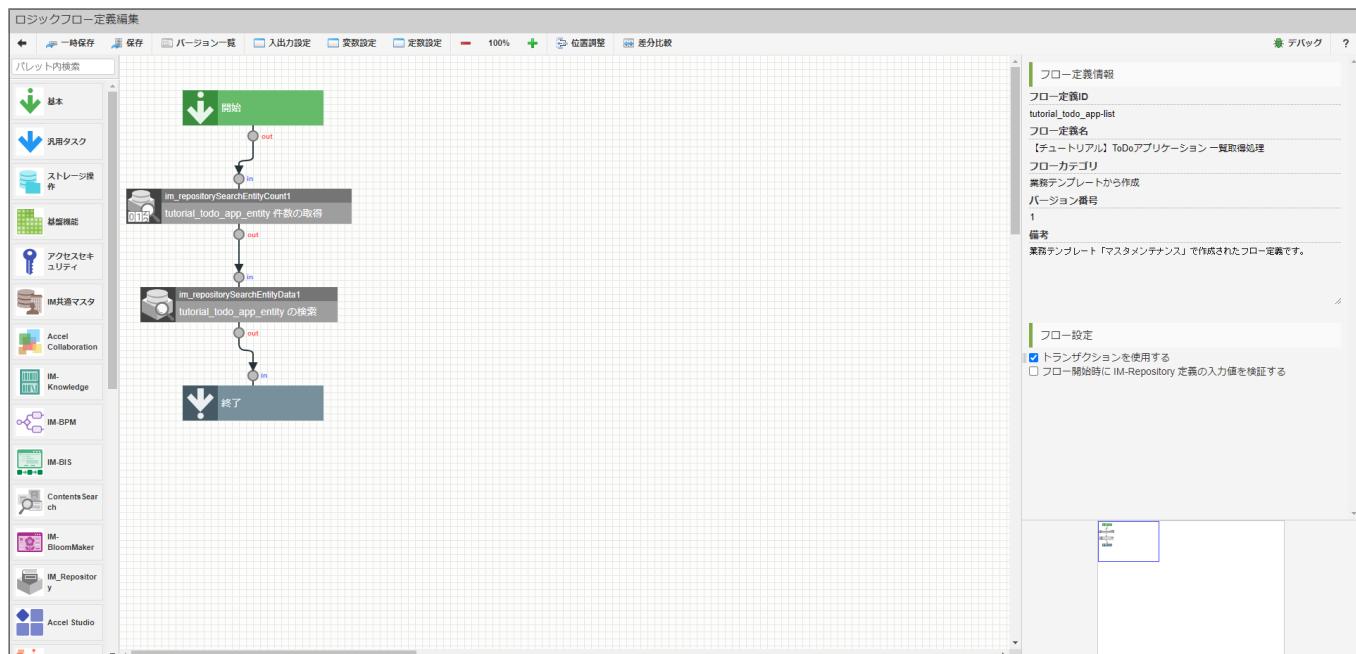
タスク一覧画面が表示されますので、「閉じる」ボタンで画面を閉じてください。

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、ロジックフロー一定義編集画面を表示します。



「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。



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 `m_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。

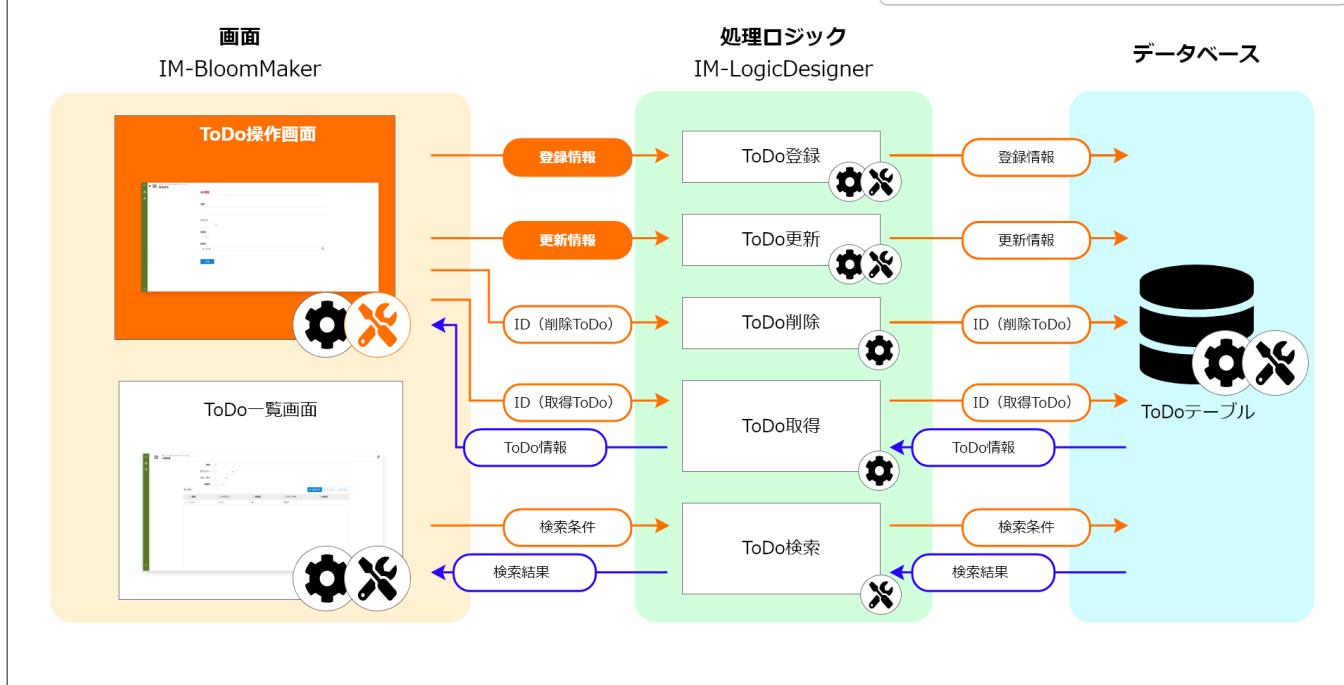
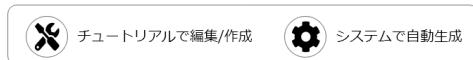
登録/更新/参照画面の修正

「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。

修正せよ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。

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

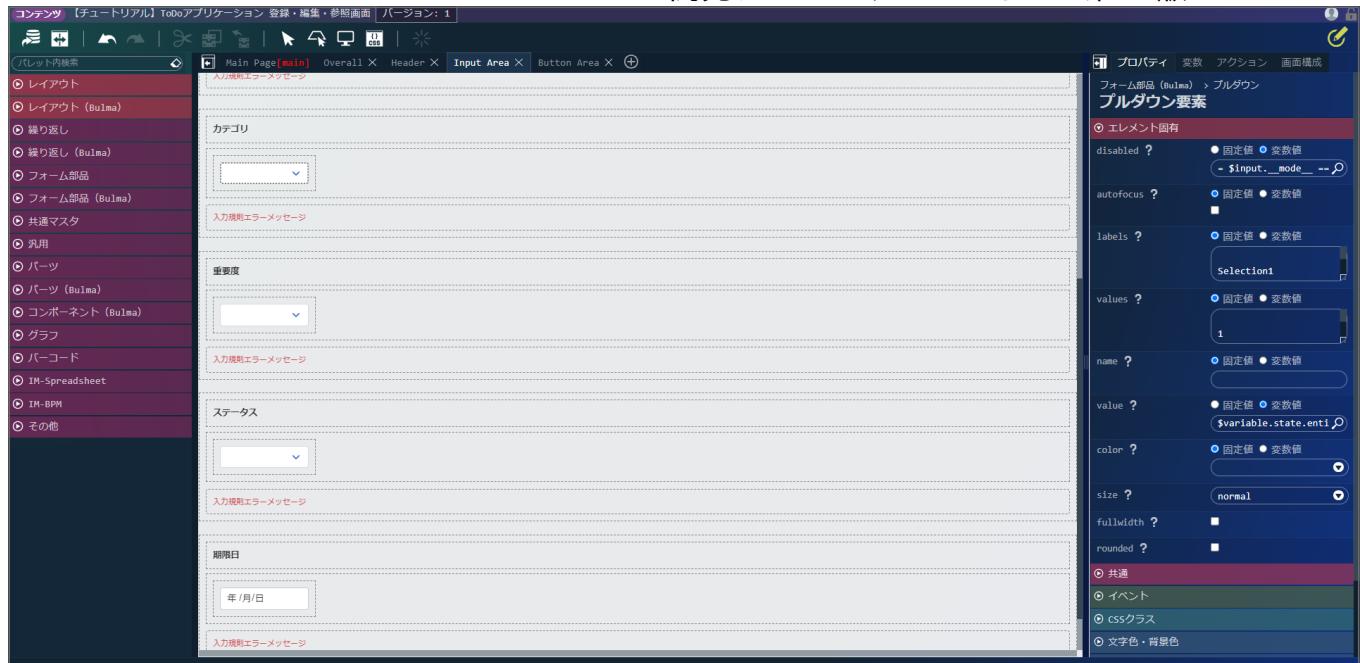
画面上部のタブの中から「Input Area」タブを選択します。

「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。

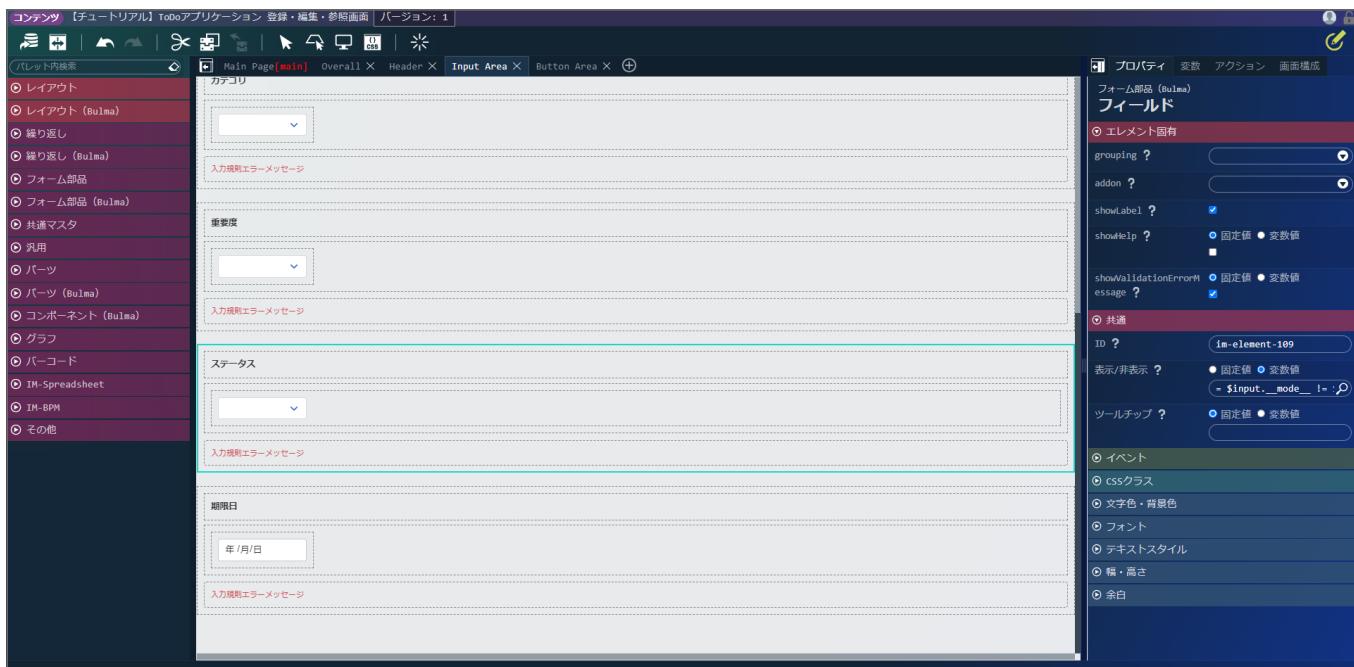
画面の表示非表示の修正

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。

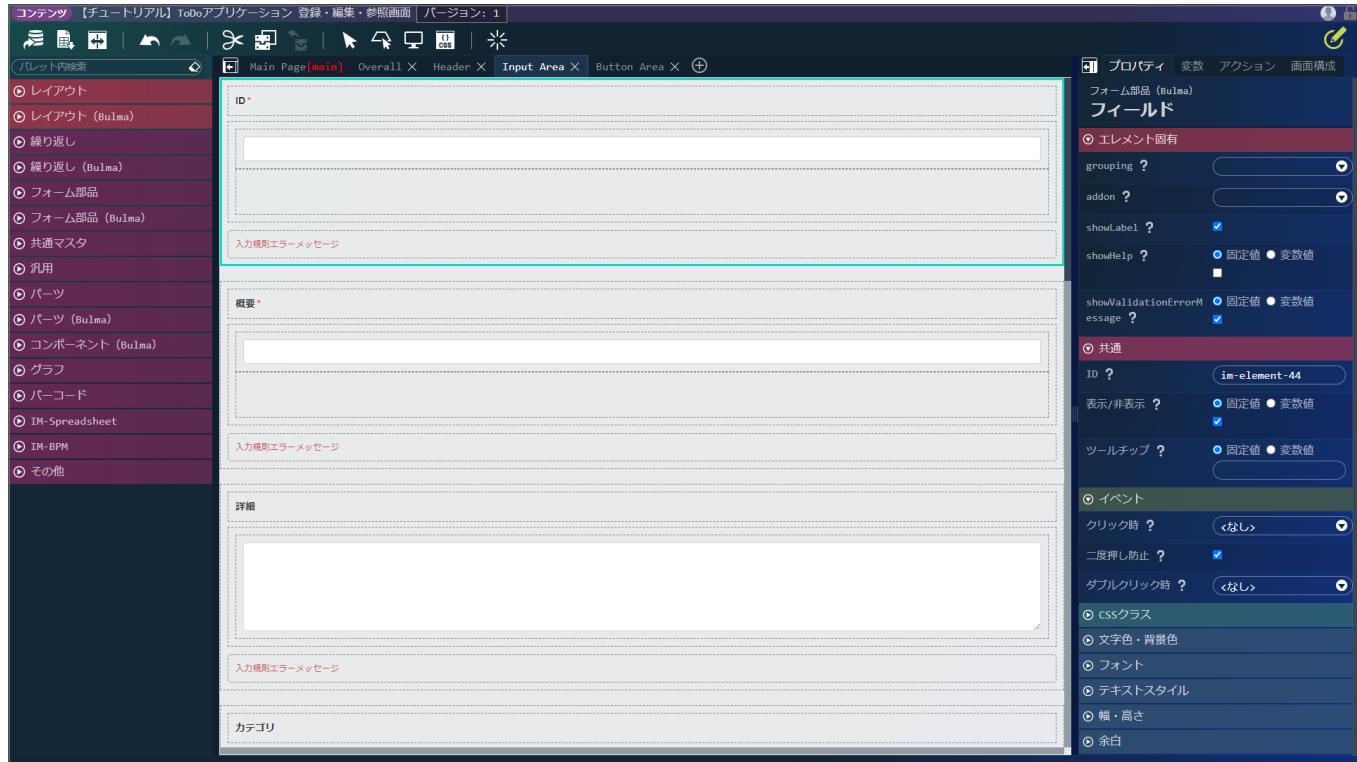
| キー | 画面種別 |
|----------|------|
| register | 登録画面 |
| edit | 編集画面 |
| refer | 参照画面 |
| list | 一覧画面 |

「\$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。



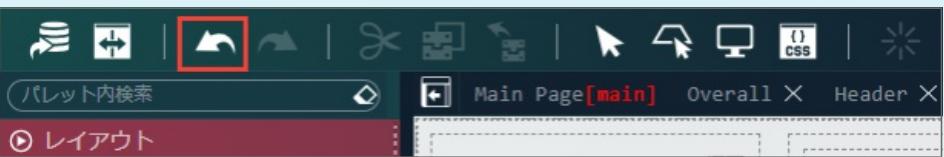
同様に、「担当者」についても今回のチュートリアルでは自動設定するため、入力項目から削除します。

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

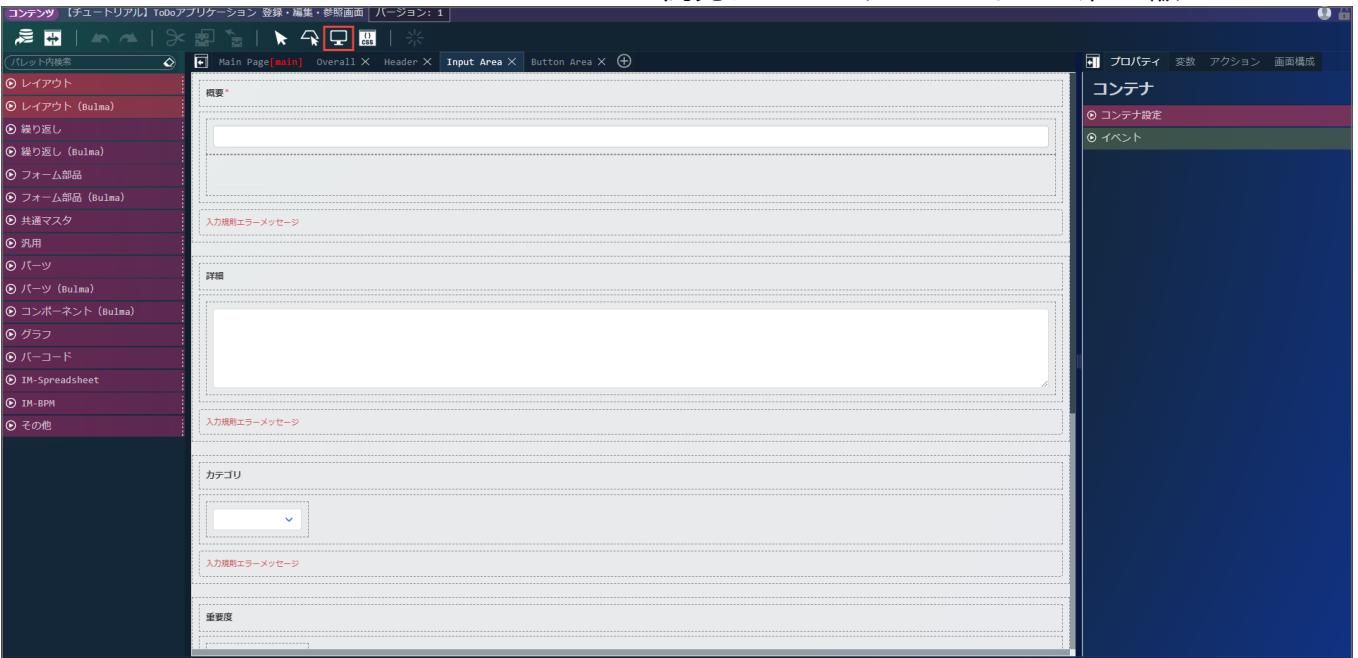
画面項目の削除で誤った項目を削除してしまった場合など、前の操作を取り消したい場合は画面上部の  か、「Ctrl+z」で直前の操作を取り消せます。



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、動作確認を行いましょう。

画面上部の  をクリックしてください。



【デュアル】ToDoアプリケーション 登録・編集・参照画面 バージョン: 1

概要*

詳細

「*」は必須項目です。

カテゴリ

重要度

期限日 年 /月/日 □

登録

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。

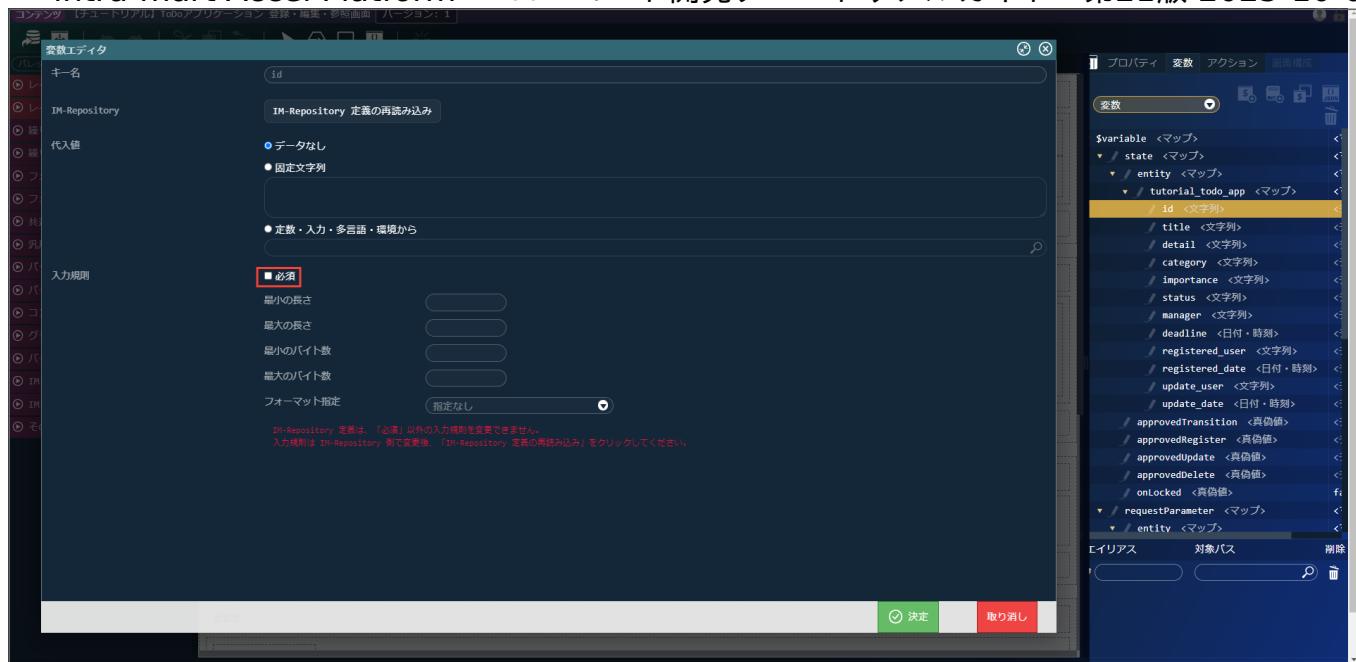
これは、登録処理等に利用している削除対象の変数が必須設定されているためです。

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。

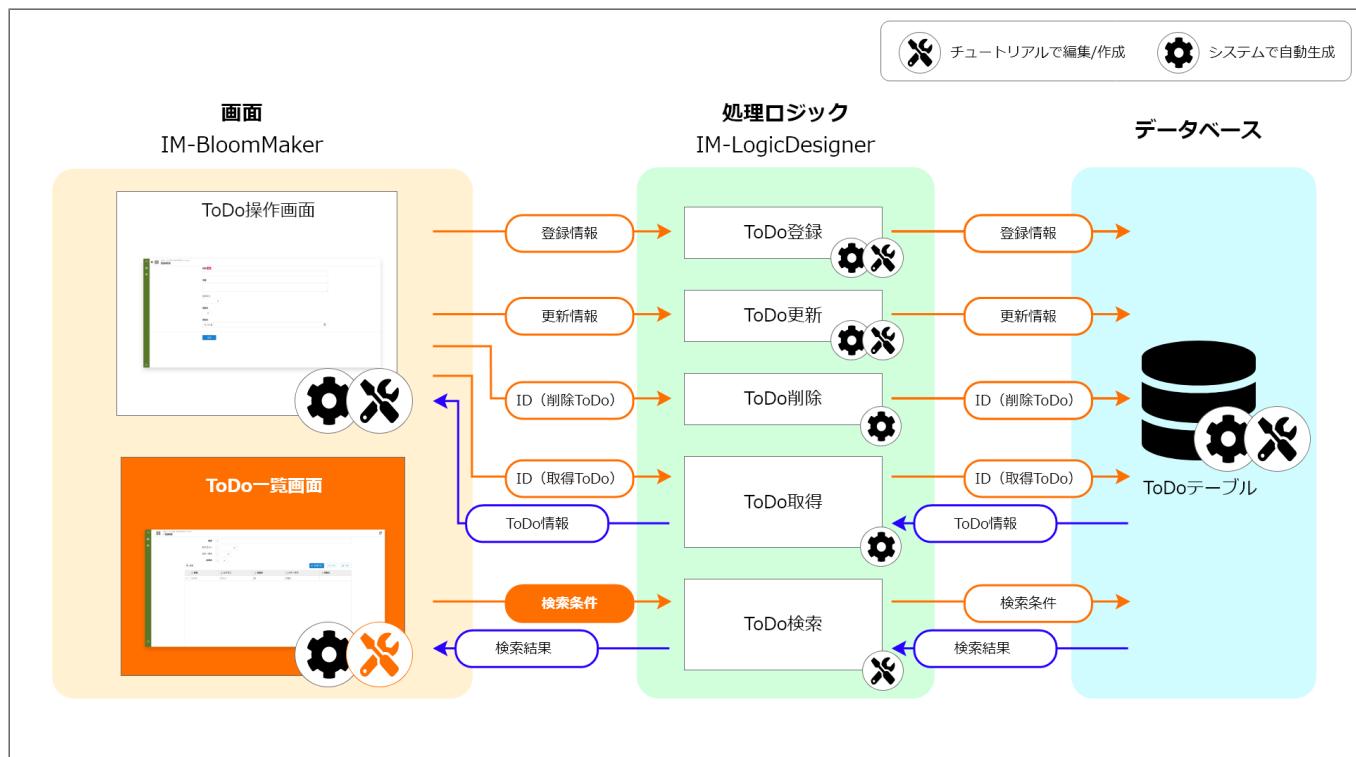
一覧画面の修正

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。

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。

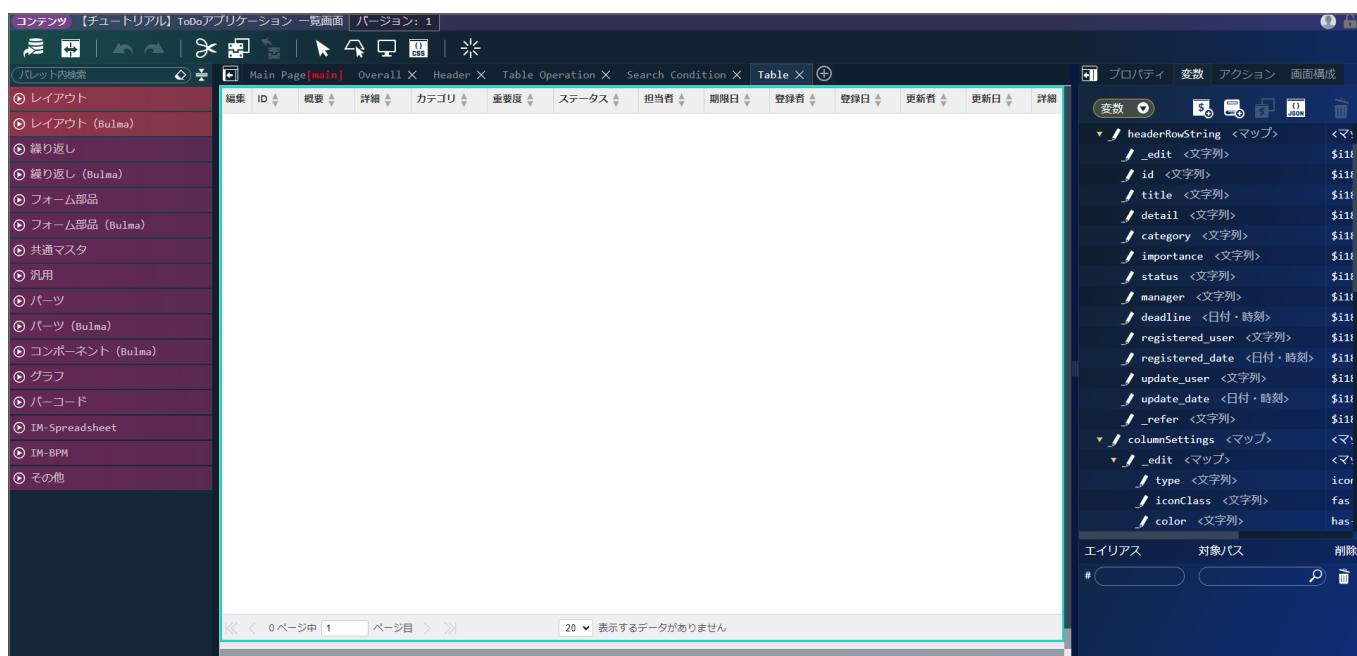
画面上部のタブから「Table」タブを選択してください。

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テンプレート」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。

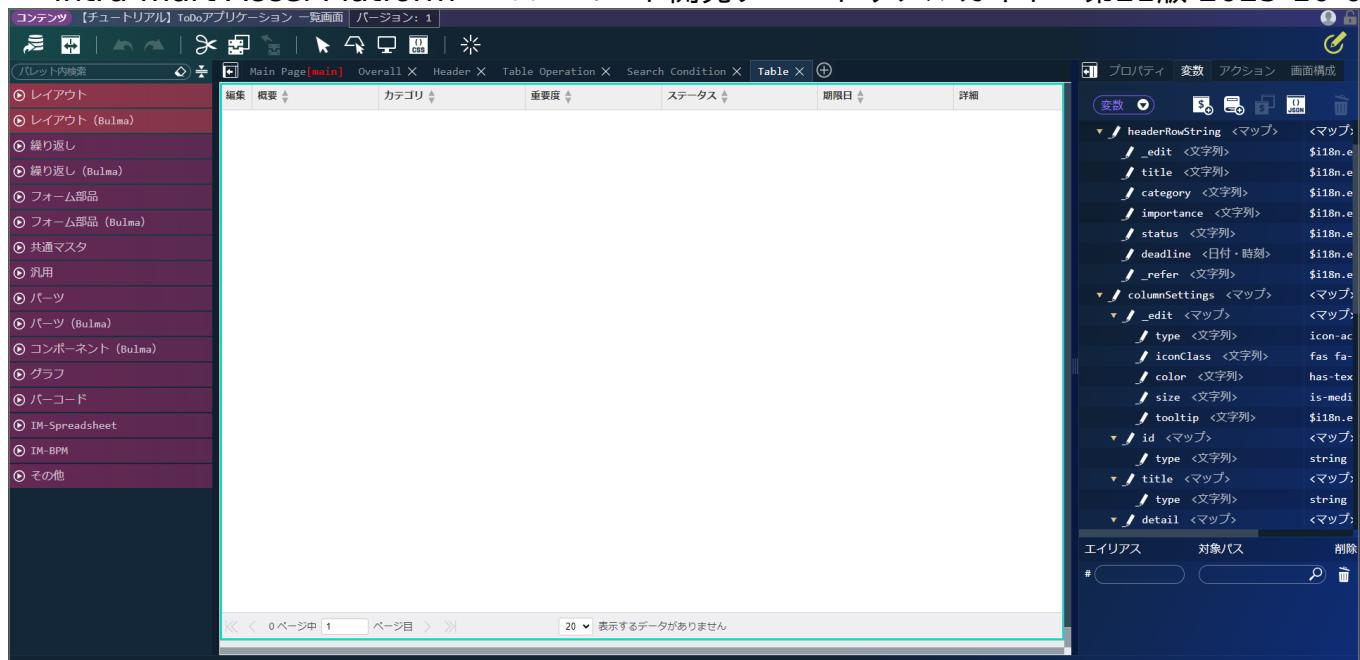
画面右側で変数タブを選択します。



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ワードを「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、キーワードを「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 |

| キーノ | 値の型 | 代入値 |
|-----|-----|-----|
|-----|-----|-----|

| | | |
|--------|-----|---|
| 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、キーノを「status」、値の型を「マップ」とした定数を追加します。

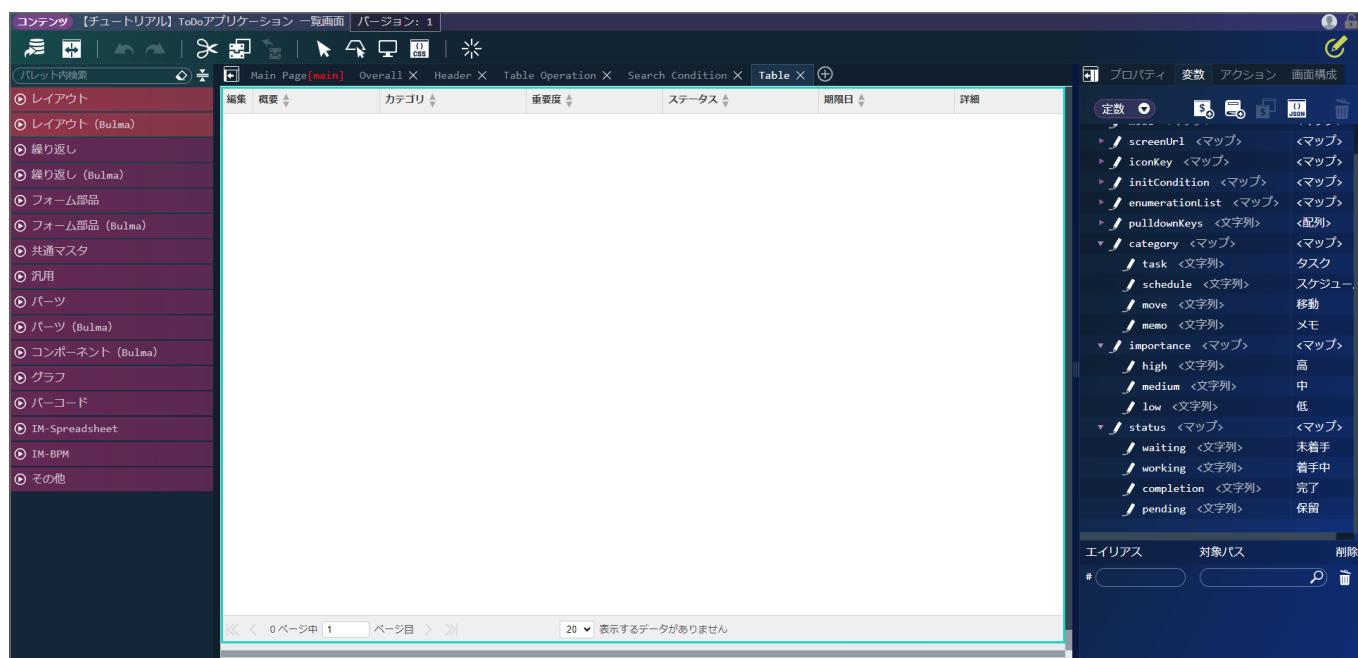
以下の定数を設定します。

| キーノ | 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"
}
```

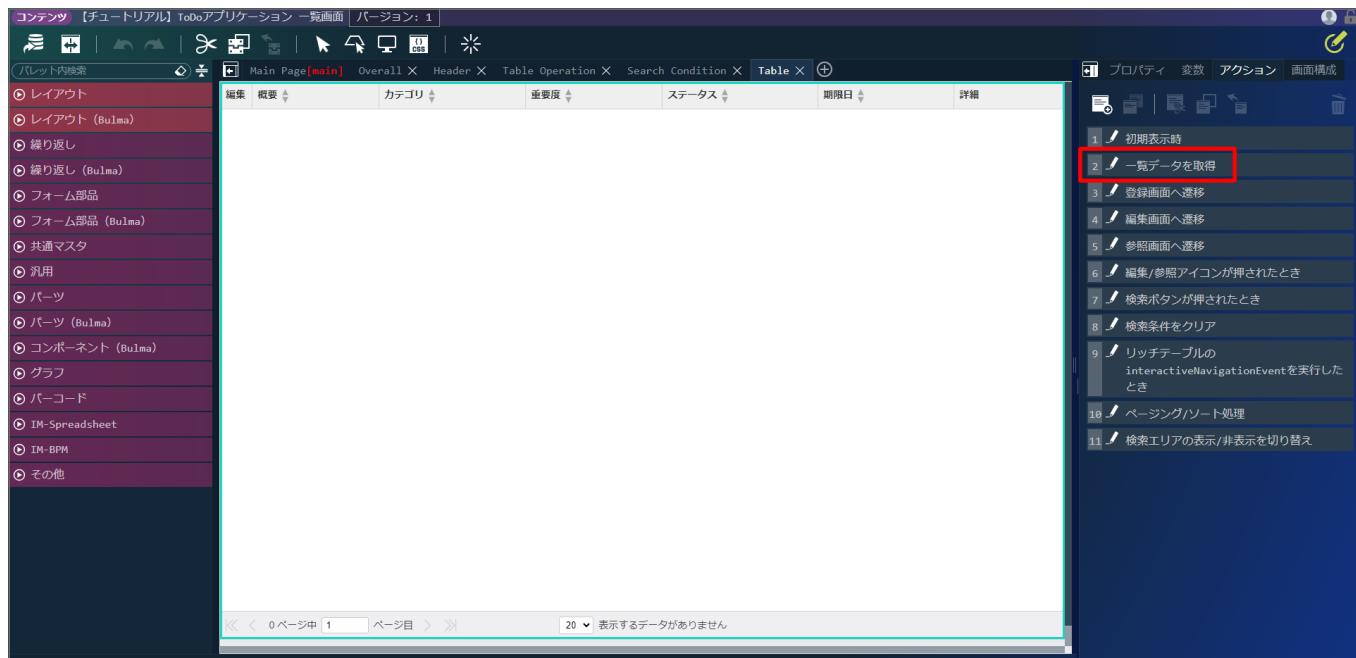
JSON入力する場合は、定数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いように注意してください。



The screenshot shows the Accel Platform's configuration interface. On the left, there's a sidebar with various application components listed under 'コンポーネント' (Components). The main area displays a table of constants. The 'status' constant is expanded, showing its mapping to four values: 'waiting' (未着手), 'working' (着手中), 'completion' (完了), and 'pending' (保留). The 'importance' constant is also expanded, showing its mapping to three values: 'high' (高), 'medium' (中), and 'low' (低). The 'Table' tab is selected in the top navigation bar.

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。



「一覧データを取得」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。

アクションエディタ

一覧データを取得

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.endP} に \${\$env.accountCo} を代入する

アクションアイテム説明

IM-LogicDesigner フロールーティング accel-studio-app/tutorial_todo_app/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

アクションアイテム説明

変数 \${variable.stat} に \${variable.endP} を代入する

アクションアイテム説明

変数 \${variable.stat} に \${\$env.const.false} を代入する

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

決定 取り消し

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。

アクションエディタ

一覧データを取得

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.endP} に \${\$env.accountCo} を代入する

アクションアイテム説明

IM-LogicDesigner フロールーティング accel-studio-app/tutorial_todo_app/list にリクエストを送信する

カスタムスクリプトを実行する

1 |

常に実行

カスタムスクリプトを実行する

決定 取り消し

「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テンプレート」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に、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。

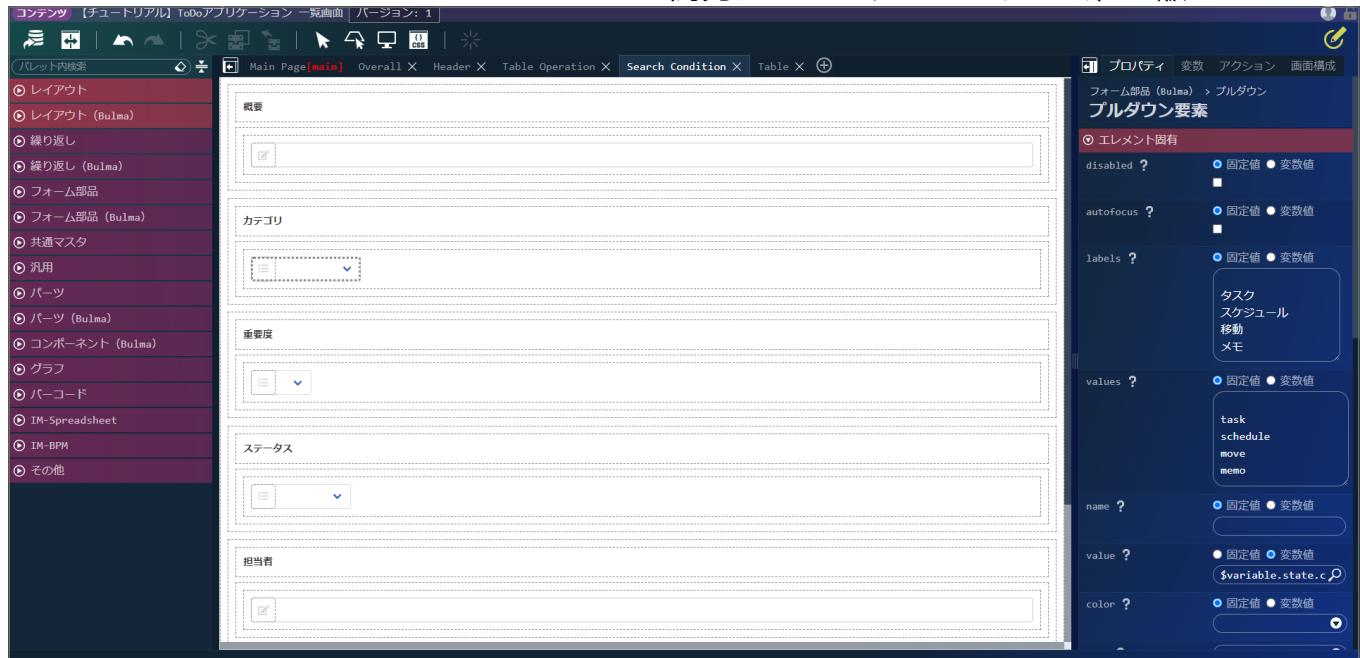
1行目には何も入力せずに改行するように注意してください。

- 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

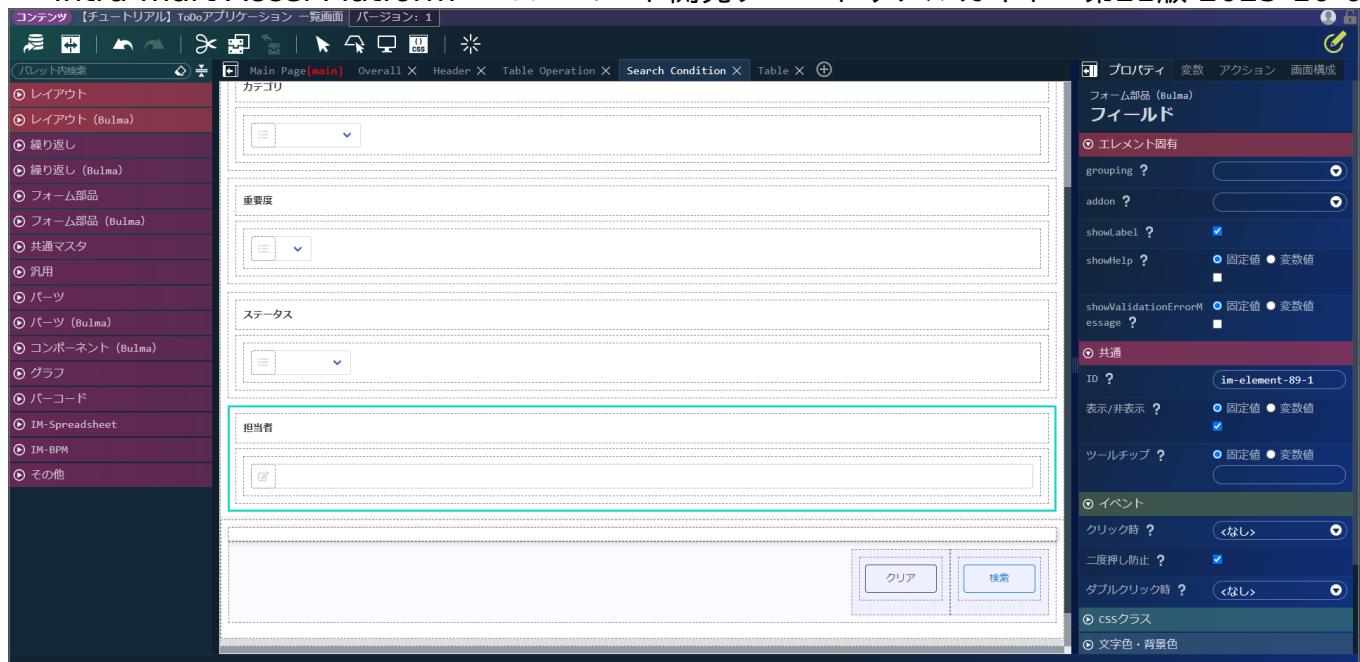
これで、プルダウン要素の入力値の設定が完了しました。

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いましょう。

画面上部の を押下してください。

| 編集 | 概要 | カテゴリ | 重要度 | ステータス | 期限日 | 詳細 |
|----|------|------|-----|-------|------------|----|
| | テスト1 | タスク | 高 | 未着手 | 2023/10/27 | |
| | テスト2 | タスク | 中 | 未着手 | 2023/10/27 | |

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。

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

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

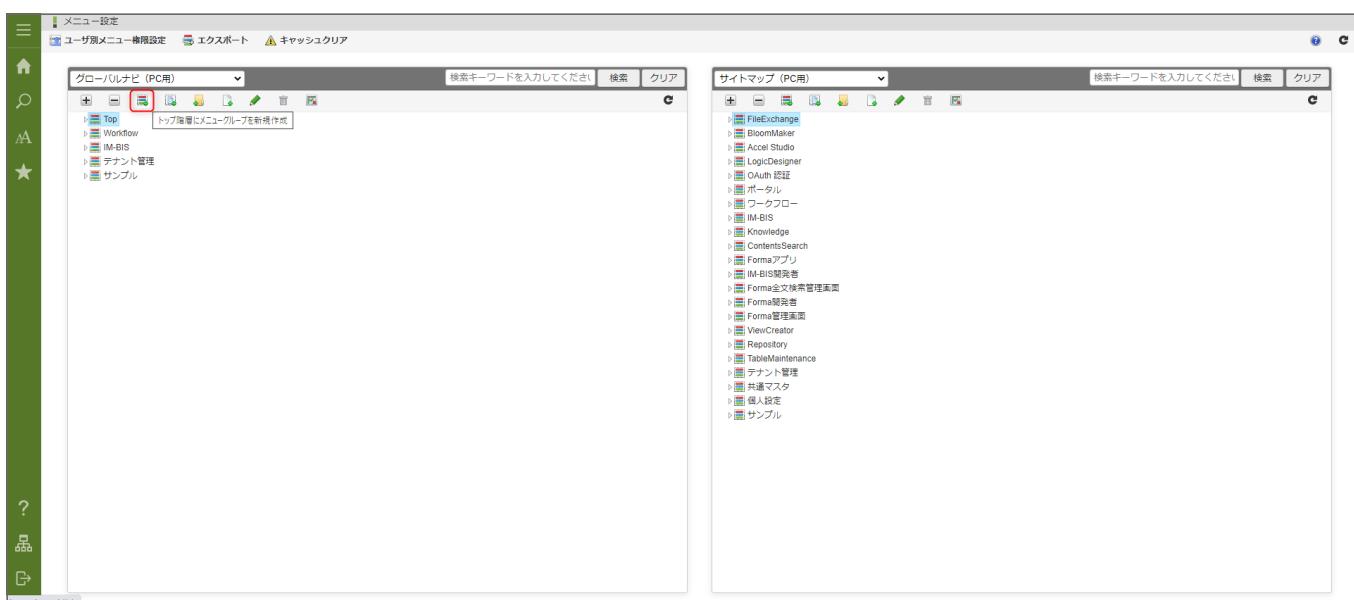
■ テナント情報

- 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。

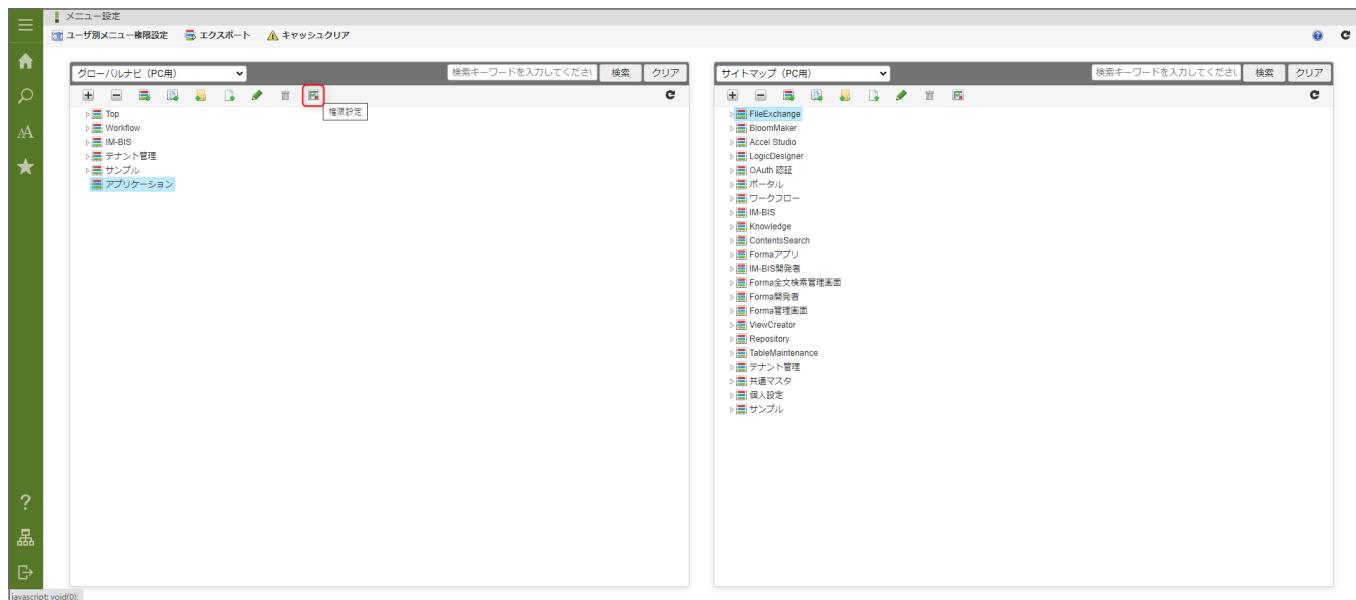


「メニューグループの新規作成」ダイアログに以下を設定します。

| ラベル | 入力値 |
|--------|-------------|
| メニューID | application |
| メニュー名 | 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))

キャッシュクリア

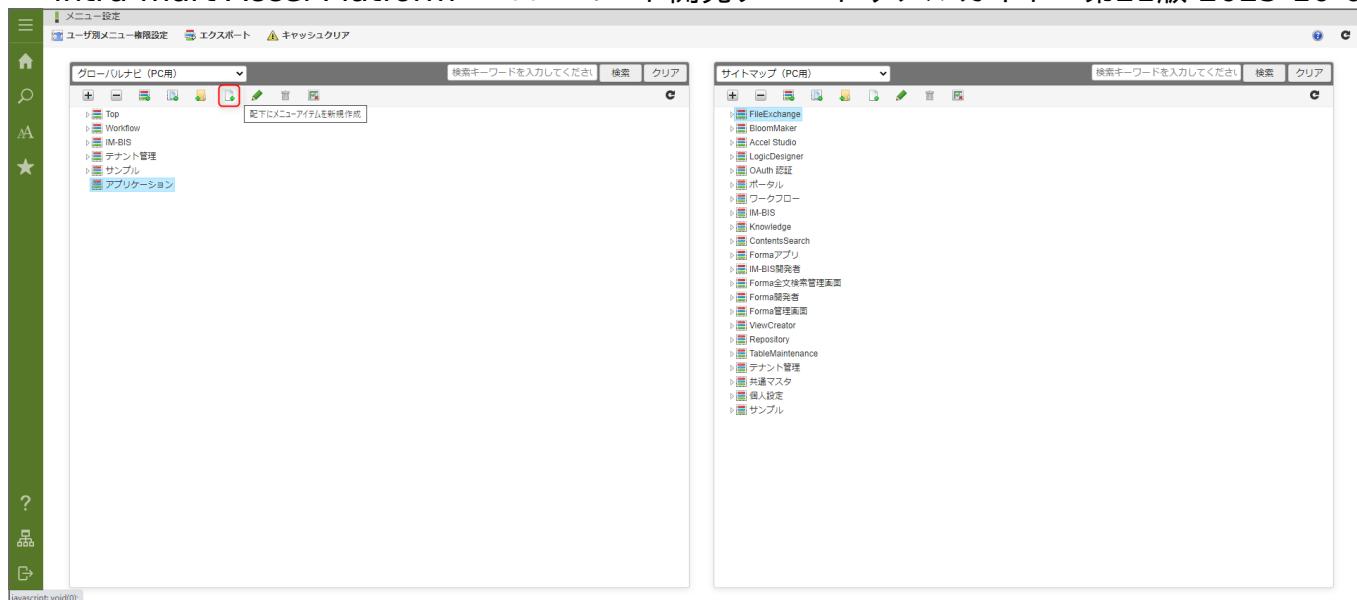
アクションの種類 全てのアクション ▾ 権限設定を開始する

| 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 | 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共通マスター管理者 | IMマネジメント |
| 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 > | | | | | | | | | | | | | | |



Accel-Mart Quick 環境ご利用の方は「アプリケーション」配下に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。

| | | | | | | | |
|-----------------------------|--|-----------------------------|---|--------------------------|--------|----------|--------|
| メニューアイテムID | tutorial_todo_list | | | | | | |
| メニューアイテム名 | <table border="1"> <tr> <td>日本語 *</td> <td>ToDo一覧</td> </tr> <tr> <td>英語</td> <td>ToDo一覧</td> </tr> <tr> <td>中国語 (中国)</td> <td>ToDo一覧</td> </tr> </table> | 日本語 * | ToDo一覧 | 英語 | ToDo一覧 | 中国語 (中国) | ToDo一覧 |
| 日本語 * | ToDo一覧 | | | | | | |
| 英語 | ToDo一覧 | | | | | | |
| 中国語 (中国) | ToDo一覧 | | | | | | |
| URL * | accel-studio-app/tutorial_todo_app/list | | | | | | |
| 呼び出し方法 | GET | | | | | | |
| 引数 | <p>+ 行追加 - 選択行削除</p> <table border="1"> <tr> <td><input type="checkbox"/> キー</td> <td>値</td> </tr> </table> | <input type="checkbox"/> キー | 値 | | | | |
| <input type="checkbox"/> キー | 値 | | | | | | |
| アイコン画像 | <table border="1"> <tr> <td>標準</td> <td><input checked="" type="radio"/> ファイルパス</td> <td>コンテキストパス配下のURLを入力してください。</td> </tr> <tr> <td></td> <td>ス</td> <td>-</td> </tr> </table> | 標準 | <input checked="" type="radio"/> ファイルパス | コンテキストパス配下のURLを入力してください。 | | ス | - |
| 標準 | <input checked="" type="radio"/> ファイルパス | コンテキストパス配下のURLを入力してください。 | | | | | |
| | ス | - | | | | | |

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。

The screenshot shows the 'Todo Application - Details' page. At the top right, there is a 'Permissions' button with a red box around it. Below the header, there are tabs for 'All', '画面' (Screens), 'Public URL', 'Log', 'Workflow', 'Data Dictionary', and 'Utility'. A search bar and a 'Search Options' button are also present. The main area lists various application URLs with their corresponding tenant IDs and creation dates.

| URL | Tenant ID | Creation Date |
|---|-----------|---------------------|
| 画面コンテンツ (ID: im-bloommaker//contents/tutorial_todo_app-list) | tenant | 2024/08/29 15:54:45 |
| 【チュートリアル】ToDoアプリケーション - 認可画面 | | |
| 「マスタメンテナンス」テンプレートから作成された一覧画面のコンテンツです。 | | |
| 画面コンテンツ (ID: im-bloommaker//contents/tutorial_todo_app-crud) | tenant | 2024/08/29 15:54:45 |
| 【チュートリアル】ToDoアプリケーション 登録・編集・参照画面 | | |
| 「マスタメンテナンス」テンプレートから作成された登録・編集・参照画面のコンテンツです。 | | |
| 画面URL (ID: im-bloommaker//routes/tutorial_todo_app-register) | tenant | 2024/08/29 15:54:45 |
| 【チュートリアル】ToDoアプリケーション 登録画面 | | |
| 「マスタメンテナンス」テンプレートから作成された登録画面のルーティング定義です。 | | |
| 画面URL (ID: im-bloommaker//routes/tutorial_todo_app-refer) | tenant | 2024/08/29 15:54:45 |
| 【チュートリアル】ToDoアプリケーション 参照画面 | | |
| 「マスタメンテナンス」テンプレートから作成された参照画面のルーティング定義です。 | | |
| 画面URL (ID: im-bloommaker//routes/tutorial_todo_app-list) | tenant | 2024/08/29 15:54:45 |
| 【チュートリアル】ToDoアプリケーション - 認可画面 | | |
| 「マスタメンテナンス」テンプレートから作成された一覧画面のルーティング定義です。 | | |
| 画面URL (ID: im-bloommaker//routes/tutorial_todo_app-edit) | tenant | 2024/08/29 15:54:45 |
| 【チュートリアル】ToDoアプリケーション - 編集画面 | | |
| 「マスタメンテナンス」テンプレートから作成された編集画面のルーティング定義です。 | | |
| PUT accel-studio-app/tutorial_todo_app/upload-files | tenant | 2024/08/29 15:54:45 |
| APIのURL (ID: im-logic//routes/accel-studio-app/tutorial_todo_app/upload-files) | | |
| PUT accel-studio-app/tutorial_todo_app/upload-files | tenant | 2024/08/29 15:54:45 |
| APIのURL (ID: im-logic//routes/accel-studio-app/tutorial_todo_app/upload-files/list) | tenant | 2024/08/29 15:54:45 |
| GET accel-studio-app/tutorial_todo_app/upload-files/list | tenant | 2024/08/29 15:54:45 |
| APIのURL (ID: im-logic//routes/accel-studio-app/tutorial_todo_app/upload-files/all/DELETE) | tenant | 2024/08/29 15:54:45 |
| DELETE accel-studio-app/tutorial_todo_app/upload-files/all | tenant | 2024/08/29 15:54:45 |

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 'Todo Application - Permissions' page. It features a grid where rows are categorized by URL type ('画面URL' or 'APIのURL') and columns represent different roles. The 'Authenticated' column is highlighted with a yellow background. A message at the top indicates that changes are being applied and suggests clicking the update button. The bottom of the page has a progress bar.

| リソース | | 認可 | | | | | | | | | | | | | | | | | | | | |
|---|--|----|----|---------|----------|--------|-------|---------|-------|---------|-----------|----------|--------|----------|------------|-------------|---------------|---------|----------------|------------------|----------------|----------------|
| | | 認証 | 組織 | ゲストユーザー | 認証済みユーザー | 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 | メニュー連用管理者 | アカウント管理者 | ロール管理者 | カレンダー管理者 | ジョブケシユラ管理者 | IM共通マスター管理者 | IM共通マスター連用管理者 | ポータル管理者 | IM-Workflow管理者 | IM-Workflow連用管理者 | IM-Workflow監査者 | IM-Workflowユーザ |
| 【画面URL】 | | | | | | | | | | | | | | | | | | | | | | |
| 【チュートリアル】ToDoアプリケーション - 認可画面 | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| 【チュートリアル】ToDoアプリケーション 登録画面 | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| 【チュートリアル】ToDoアプリケーション 参照画面 | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| 【チュートリアル】ToDoアプリケーション - 登録画面 | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| 【チュートリアル】ToDoアプリケーション - 参照画面 | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| 【APIのURL】 | | | | | | | | | | | | | | | | | | | | | | |
| POST tutorial_todo_app-list/search > | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| POST accel-studio-app/tutorial_todo_app > | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| GET accel-studio-app/tutorial_todo_app > | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| GET accel-studio-app/tutorial_todo_app-list > | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| PUT accel-studio-app/tutorial_todo_app > | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| DELETE accel-studio-app/tutorial_todo_app > | | X | ✓ | ✗ | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |

「更新」をクリックし、更新を行います。

The screenshot shows a grid-based configuration interface for application permissions. The columns represent resources (リソース), authentication methods (認証), and roles (ロール). The rows list various application URLs and API endpoints. Most entries have a red 'X' indicating they are not yet assigned or are denied. A specific row for '【チュートリアル】ToDoアプリケーション 認可設定' has a green checkmark. A modal dialog box is centered over the grid, containing a warning message: '⚠️ 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?' (Update application permission information. Is it okay?). It includes 'キャンセル' (Cancel) and '更新' (Update) buttons.

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。

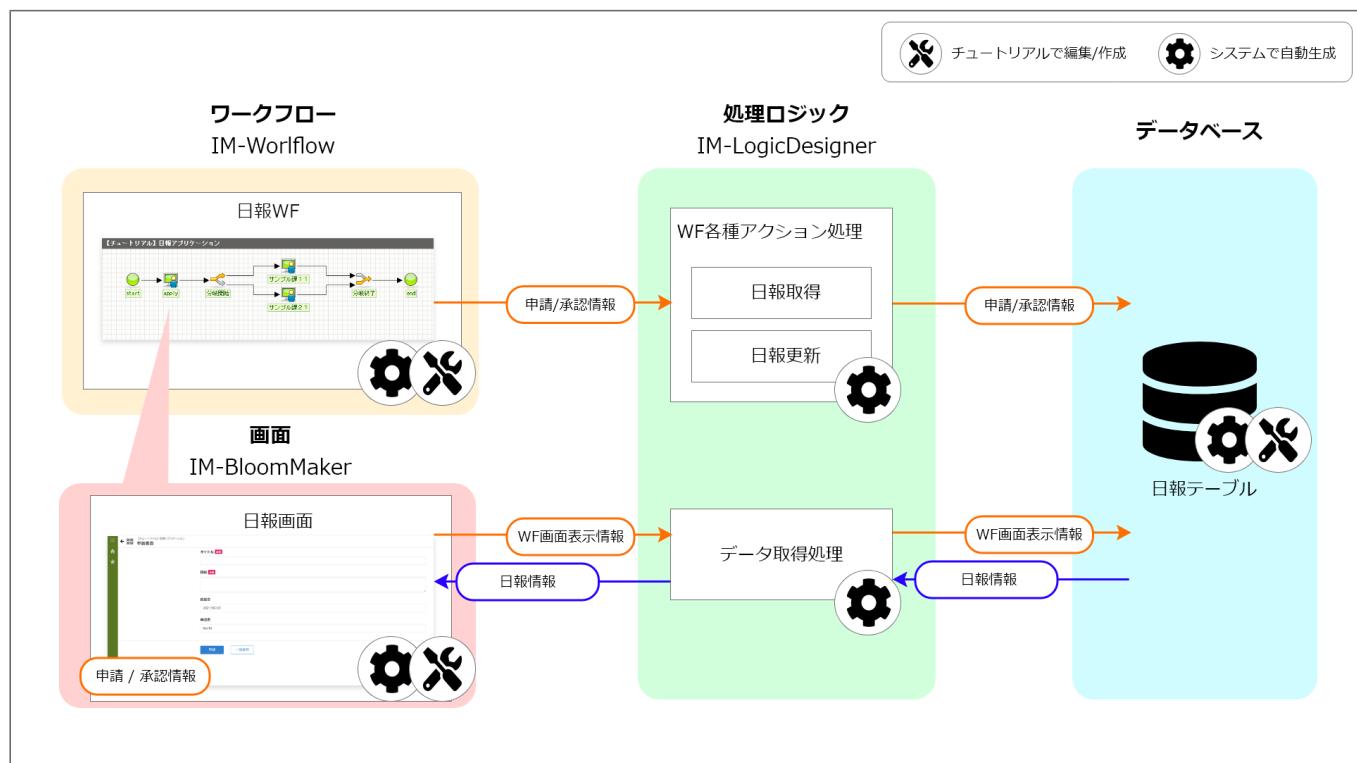
作成の流れ

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。

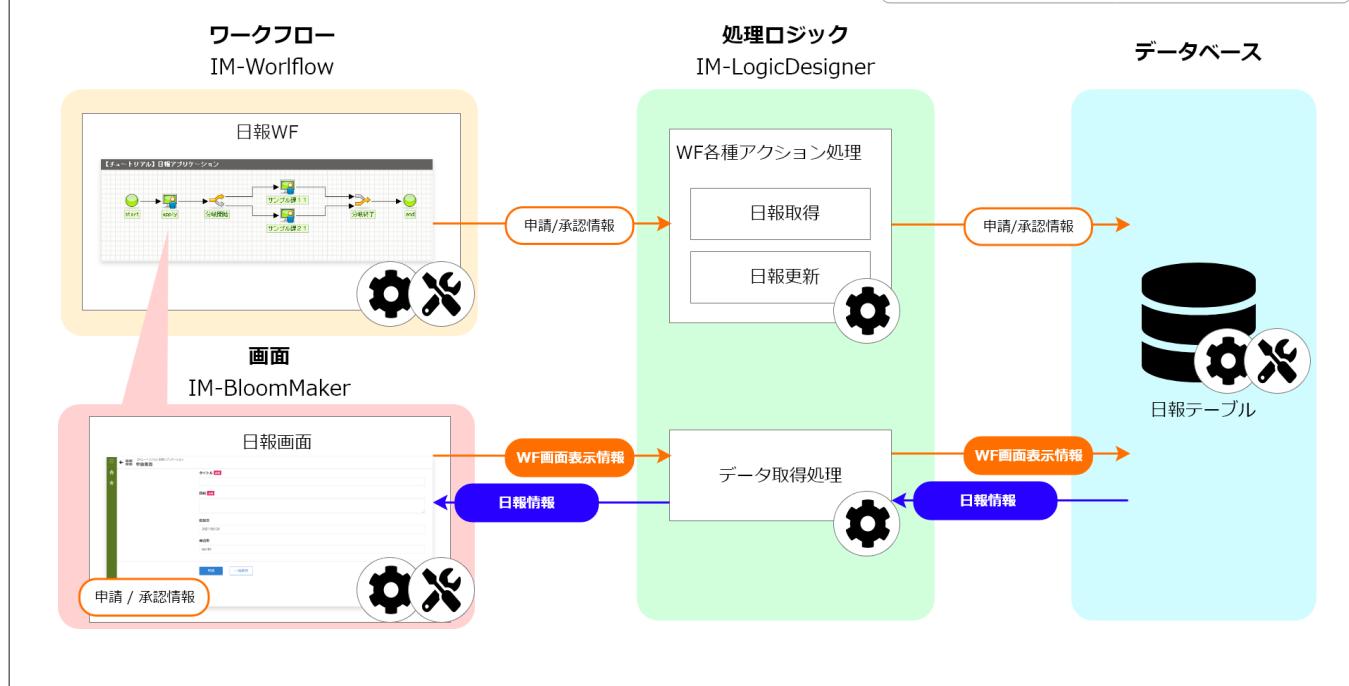


日報アプリケーションの仕様

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。

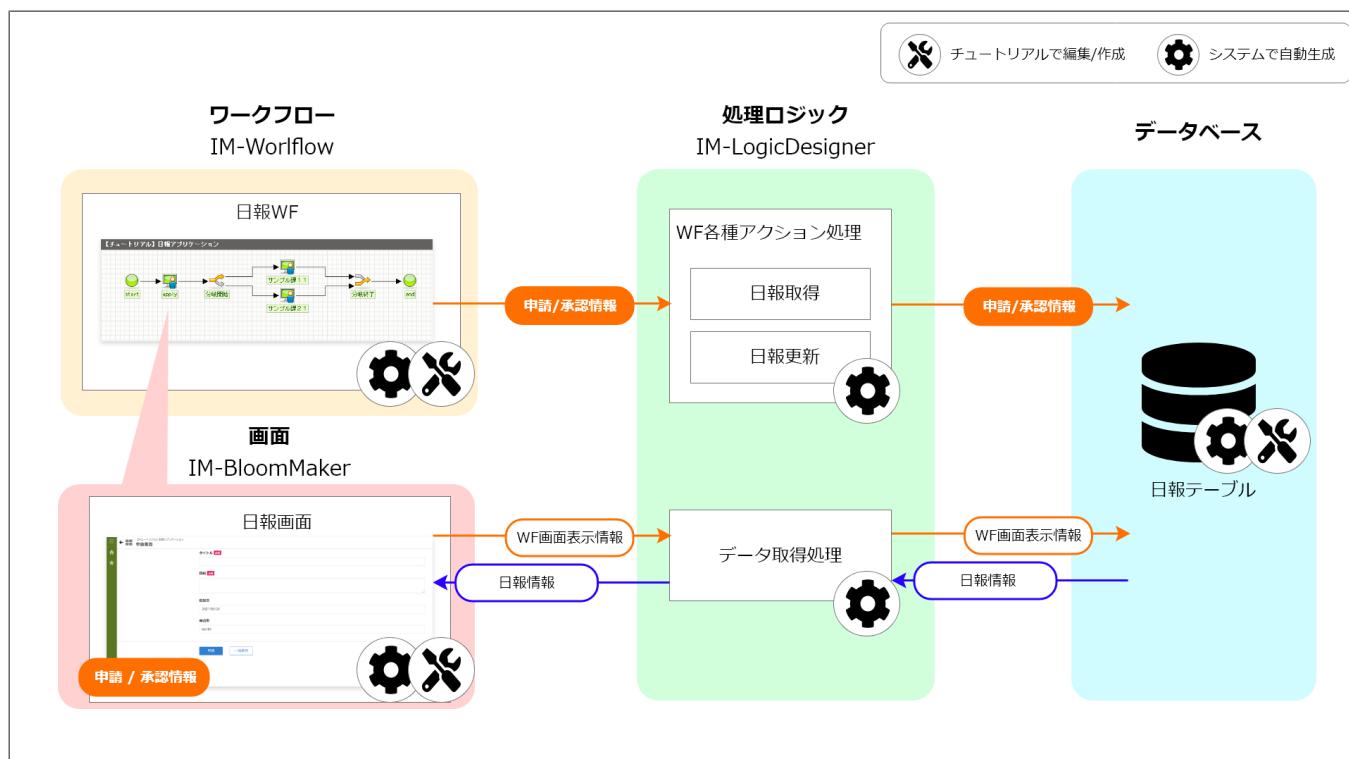
申請画面や承認画面表示時にどの処理画面を表示したのかを判別し、以下の図にあるように、データ取得処理を実行して表示する日報情報を取得します。



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。

The screenshot shows the 'Application Creation' section of the Accel Studio interface. On the left, there's a sidebar with icons for Home, Search, and Font Size. The main area has a title bar 'Application Creation Business Template List'. Below it, a section titled 'Base Template' explains that it's for creating applications and recommends it for beginners. It lists four templates: 'Master Maintenance' (マスタメンテナンス), 'Simple Workflow' (シンプルなワークフロー), 'Spreadsheet Type List' (スプレッドシート型一覧), and 'Empty Application' (空のアプリケーション). Each template has a preview icon and a 'Next' button. Below this, a section titled 'Sample Application' (サンプルアプリケーション) shows five sample application icons: a building with a person, a package with a barcode, a user profile with a checkmark, a group of people, and a document with a chart.

「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
 - ユーザデータ削除処理
 - ユーザデータ取得処理

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*

- 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

アプリケーション情報

① 作成する資材をまとめて1つのアプリケーションとして登録します。
作成するアプリケーション情報を入力してください。

アプリケーションID*  また、アプリケーションを一度作成した後にアプリケーションIDを変更することはできません。
半角英数字、- (ハイフン)、_ (アンダースコア)で入力してください。

IDの決定に利用する文字列 
半角英数字、- (ハイフン)、_ (アンダースコア)で入力してください。

アプリケーション名* 

説明 

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。 |



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 作成する資材をまとめて1つのアプリケーションとして登録します。
作成するアプリケーション情報を入力してください。

アプリケーションID* 
半角英数字、- (ハイフン)、_ (アンダースコア)で入力してください。
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

① 作成するアプリケーションで扱うエンティティの定義（テーブルおよびカラム）を作成します。
扱うデータの定義で指定したエンティティ情報をもとにアプリケーションの各画面が生成されます。

基本情報

詳細に設定する
データベース種別を変更するなど、エンティティの設定を詳細に行う場合は、詳細設定を有効にしてください。

エンティティ名* 

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

アプリケーション作成

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では「詳細に設定する」はOFFのままとしてください。

エンティティについての情報を設定します。

エンティティ名、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 |
|-----------------|----------------------------|----|
| エンティティ名 | 【チュートリアル_エンティティ】日報アプリケーション | |
| 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。

「項目設定」 - 「+追加」を選択してください。新規項目欄が追加されます。

The screenshot shows the 'Project Settings' dialog box. At the top, there are three buttons: a green plus sign labeled '追加' (Add), a green plus sign labeled '既存の辞書項目から選択' (Select from existing dictionary items), and a green plus sign labeled '新規作成する辞書項目を参照' (Refer to new dictionary items). Below these buttons, there are two input fields: '表示名 *' (Display name) containing 'ユーザデータID' and '物理名 *' (Physical name) containing 'user_data_id'. There is also a '詳細' (Details) button next to the display name field.

「表示名」に「タイトル」、「物理名」に「title」、「データ型」は「文字列」と設定します。

「表示形式」は「テキストボックス」を選択し、「最大桁数」に「1000」と設定してください。

「詳細」の歯車アイコンから「辞書項目新規作成」ダイアログを表示し、辞書項目IDを入力しましょう。

| ラベル | 入力値 | 備考 |
|-------------|-------------------------|---|
| 登録先辞書項目カテゴリ | シンプルなワークフロー | 任意のカテゴリで構いません。本チュートリアルでは「シンプルなワークフロー」というカテゴリを使用しています。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。 |

辞書項目新規作成

| | | | | | | | | | | | | | | | |
|-------|--|------|-----------------------------------|------|-----|--|--|------|------------------------------------|------|-----------------------------------|-------|-----------------------------------|------|----------------------|
| 項目情報* | 項目情報 登録先辞書項目カテゴリ* <input type="text" value="シンプルなワークフロー"/> <input type="button" value="検索"/> 辞書項目ID* <input type="text" value="tutorial_workflow_title"/> 辞書項目名* <input type="text" value="タイトル"/> | | | | | | | | | | | | | | |
| データ* | データ データ型* <input type="text" value="TEXT"/> <input type="button" value="▼"/> 変数名* <input type="text" value="title"/> | | | | | | | | | | | | | | |
| 制約 | Database設定 <table border="1"> <tr> <td>データ型</td> <td colspan="3">文字列</td> </tr> <tr> <td>物理名*</td> <td><input type="text" value="title"/></td> <td>論理名*</td> <td><input type="text" value="タイトル"/></td> </tr> <tr> <td>最大行数*</td> <td><input type="text" value="1000"/></td> <td>小数桁数</td> <td><input type="text"/></td> </tr> </table> Java設定 | | | データ型 | 文字列 | | | 物理名* | <input type="text" value="title"/> | 論理名* | <input type="text" value="タイトル"/> | 最大行数* | <input type="text" value="1000"/> | 小数桁数 | <input type="text"/> |
| データ型 | 文字列 | | | | | | | | | | | | | | |
| 物理名* | <input type="text" value="title"/> | 論理名* | <input type="text" value="タイトル"/> | | | | | | | | | | | | |
| 最大行数* | <input type="text" value="1000"/> | 小数桁数 | <input type="text"/> | | | | | | | | | | | | |
| | <input type="button" value="キャンセル"/> <input type="button" value="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、テンプレートから自動で設定される「シンプルなワークフロー」というカテゴリを使用しています。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。

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。

最後に、「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 ① 作成するアプリケーションで扱うエンティティの定義（テーブルおよびカラム）を作成します。 扱うデータの定義で指定したエンティティ情報をもとにアプリケーションの各画面が生成されます。 | | |
| 扱うデータの定義* | 基本情報 詳細に設定する <input type="checkbox"/> データベース種別を変更するなど、エンティティの設定を詳細に行う場合は、詳細設定を有効にしてください。 | | |
| 明細データの利用指定 | | | |
| ワークフロー情報の入力* | エンティティ名* <input type="text" value="【チュートリアル_エンティティ】日報アプリケーション"/> エンティティに組づくテーブル名* <input type="text" value="tutorial_daily_report_app"/> | | |
| その他情報の指定* | 項目設定 <input type="button" value="追加"/> <input type="button" value="既存の辞書項目から選択"/> <input type="button" value="新規作成する辞書項目を参照"/> 表示名* <input type="text" value="ユーザデータID"/> <input type="button" value="詳細"/> 物理名* <input type="text" value="user_data_id"/> データ型* <input type="text" value="テキストボックス"/> 表示形式* <input type="text" value="テキストボックス"/> 最大桁数 <input type="text" value="20"/> 主キー <input type="checkbox"/> 必須 <input type="checkbox"/> 削除 <input type="checkbox"/> タイトル <input type="text" value="title"/> <input type="button" value="詳細"/> 物理名* <input type="text" value="title"/> データ型* <input type="text" value="テキストボックス"/> 表示形式* <input type="text" value="テキストボックス"/> 最大桁数 <input type="text" value="1000"/> <input type="checkbox"/> <input checked="" type="checkbox"/> 必須 <input type="checkbox"/> | | |
| | 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 <input type="text" value="0"/> | | |
| | ワークフロー情報の入力 ① ワークフロー情報を入力する際の情報を入力します。 <input type="button" value="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 | 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 | | □ |
| tutorial_workflow_daily_report | 日報 | TEXT | daily_report | 20000 | 複数行テキストボックス | | □ |
| 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 | | |

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*

提交データの定義*

詳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*

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

① 作成するアプリケーションで扱うエンティティの定義（テーブルおよびカラム）を作成します。
提交データの定義で指定したエンティティ情報をもとにアプリケーションの各画面が生成されます。

基本情報

詳細に設定する
データベース種別を変更するなど、エンティティの設定を詳細に行なう場合は、詳細設定を有効にしてください。

エンティティ名* (1)
【チュートリアル_エンティティ】日報アプリケーション

エンティティに紐づくテーブル名* (1)
ヘルプアイコンから入力規則を確認できます。
tutorial_daily_report_app

項目設定 (1)

| 表示名* | 物理名* | データ型* | 表示形式* | 最大桁数 | 主キー | 必須 | 削除 |
|----------|------------------|-------|-------------|-------|-------------------------------------|-------------------------------------|-------------------------------------|
| ユーザデータID | user_data_id | 文字列 | テキストボックス | 20 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| タイトル | title | 文字列 | テキストボックス | 1000 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 日報 | daily_report | 文字列 | 複数行テキストボックス | 20000 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 提出日 | submission_date | 日付 | 日付 | | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 申請者 | applicant | 文字列 | テキストボックス | 1000 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 承認コメント | approval_comment | 文字列 | 複数行テキストボックス | 20000 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 承認者 | approver | 文字列 | テキストボックス | 1000 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

詳細データの利用指定

アプリケーション作成

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

① ワークフローを作成するための情報を指定します。
既存のルートをコピーして利用するか、新規でルートを作成して利用します。

ルート作成方法

新規にルートを作成する
申請、承認のみを行う直線型のルートを新規作成します。

既存のルートをコピーして作成する
既存ルートをコピーしてルートを作成します。ルート名とバージョン期間以外の設定がコピーされます。

利用開始日*

処理対象者の設定 処理対象者を指定する

申請対象者の指定方法 ※申請対象者の申請一覧画面にワークフローが表示されるようになります。

申請対象者 (ユーザ指定)*

+ 追加

青柳恵巳

承認対象者の指定方法

承認対象者 (ユーザ指定) *

+ 追加

上田原男

IM-Workflow資材の設定 リソースのIDを手動で決定する

添付ファイル設定 添付ファイルを利用する

承認時の添付ファイル設定 承認時に添付ファイルの追加/削除を許可する

その他情報の指定

アプリケーション作成

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「ビルト」をクリックします。

ワークフロー情報の入力

① ワークフローを作成するための情報を指定します。
既存のルートをコピーして利用するか、新規でルートを作成して利用します。

ルート作成方法

新規にルートを作成する
申請、承認のみを行う直線型のルートを新規作成します。

既存のルートをコピーして作成する
既存ルートをコピーしてルートを作成します。ルート名とバージョン期間以外の設定がコピーされます。

利用開始日*

処理対象者の設定 処理対象者を指定する

申請対象者の指定方法 ※申請対象者の申請一覧画面にワークフローが表示されるようになります。

申請対象者 (ユーザ指定)*

+ 追加

青柳恵巳

承認対象者の指定方法

承認対象者 (ユーザ指定) *

+ 追加

上田原男

IM-Workflow資材の設定 リソースのIDを手動で決定する

添付ファイル設定 添付ファイルを利用する

承認時の添付ファイル設定 承認時に添付ファイルの追加/削除を許可する

その他情報の指定

アプリケーション作成

確認

入力された情報でアプリケーションをビルトしますか?

キャンセル ビルド

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 テンプレートビルトジョブ - 詳細

| | | | |
|----------|---------------------|-----------|-------------|
| ビルトジョブID | 8hch2237cvyc4 | テンプレート名 | シンプルなワークフロー |
| ステータス | COMPLETE | 作成者ユーザコード | tenant |
| 作成日時 | 2024/09/18 17:46:09 | | |

出力日時 メッセージ 詳細

2024/09/18 17:46:09 生成するリソースのIDを自動設定します。

2024/09/18 17:46:09 生成する IM-Workflow リソースのIDを自動設定します。

2024/09/18 17:46:10 一時ディレクトリに IM-LogicDesigner インポート資材を作成しました。

2024/09/18 17:46:10 一時ディレクトリに IM-BloomMaker インポート資材を作成しました。

2024/09/18 17:46:10 一時ディレクトリに IM-Workflow インポート資材を作成しました。

2024/09/18 17:46:10 自動生成された IM-LogicDesigner 資材をインポートしました。

2024/09/18 17:46:11 自動生成された IM-BloomMaker 資材をインポートしました。

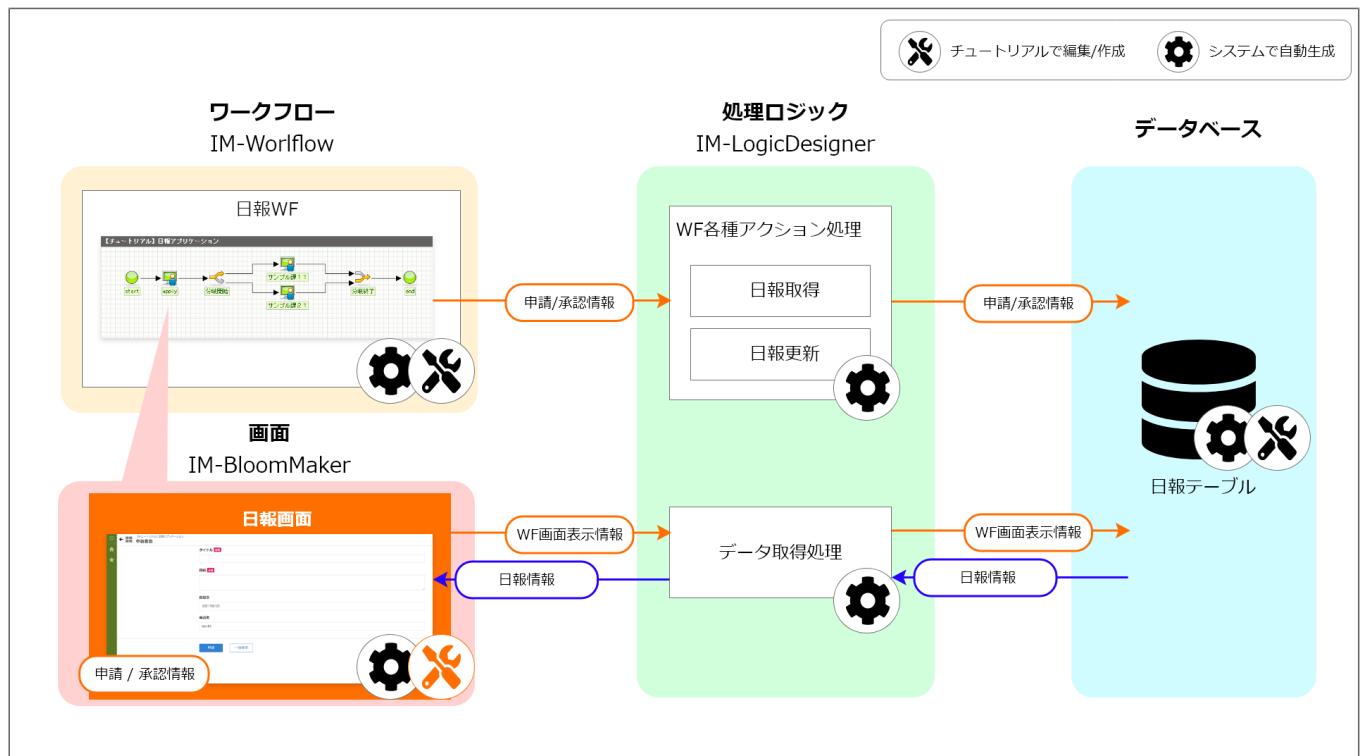
2024/09/18 17:46:12 自動生成された IM-Workflow 資材をインポートしました。

2024/09/18 17:46:12 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。

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します。修正せざる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。

「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。

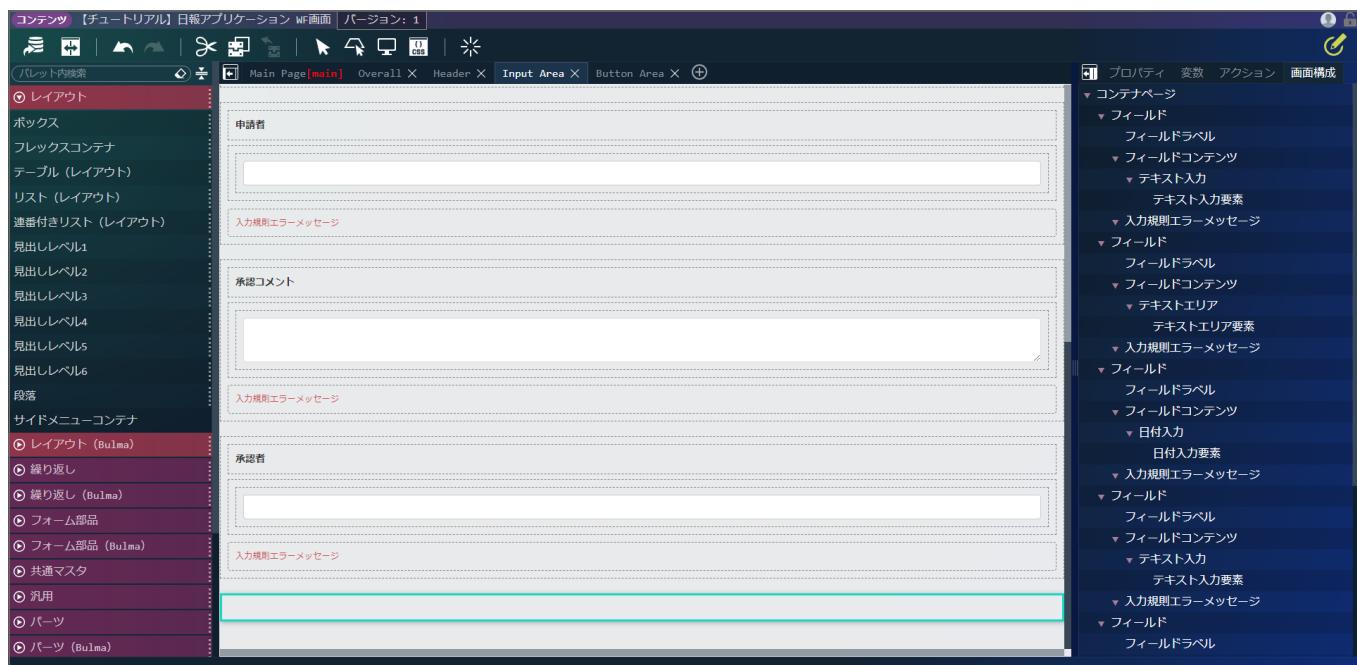
IM-BloomMaker の「デザイン編集画面」が表示されます。

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。

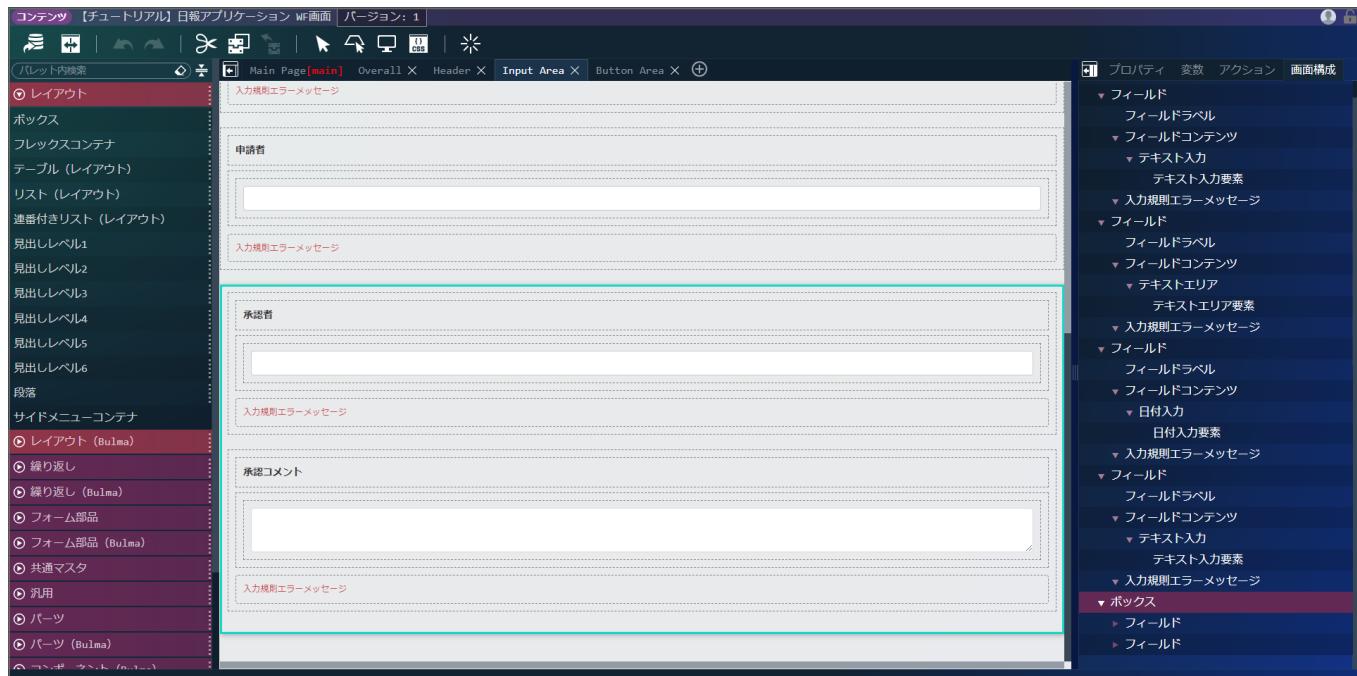
画面左側の「レイアウト」 - 「ボックス」を選択し、一番下に追加します。

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。

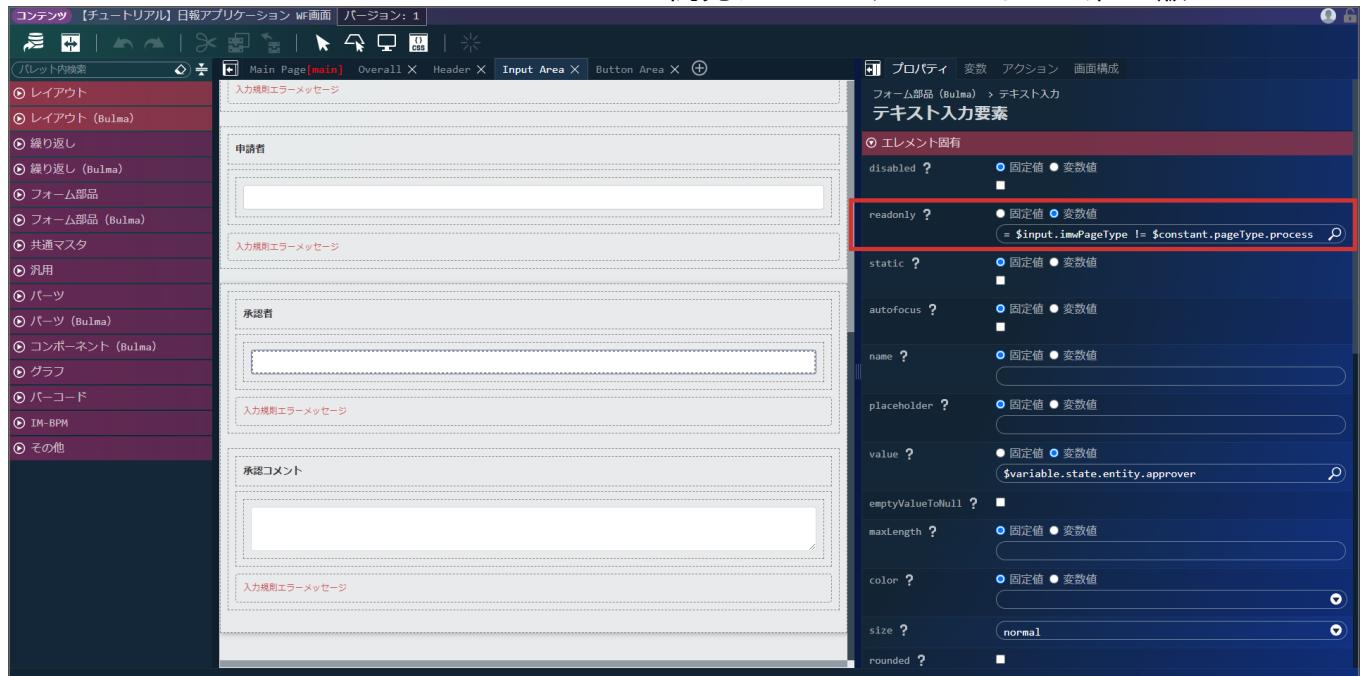
『= \$input.imwPageType != \$constant.pageType.process』

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』



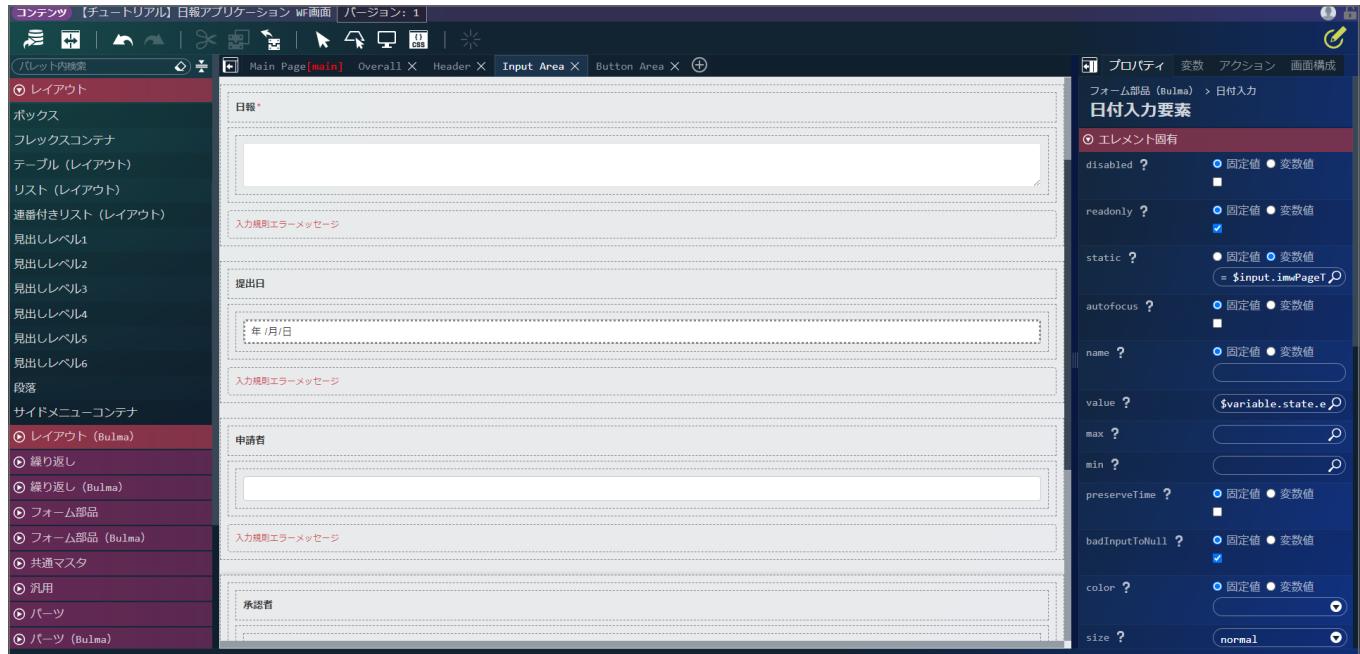
手動入力を制限する修正

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。

The screenshot shows the 'Input Area' tab selected in the top navigation bar. The left sidebar contains a tree view of components: Layout, Layout (Bulma), Wrap, Form Components, Form Components (Bulma), Common Master, Shared, Part, and Part (Bulma). The main workspace displays two text input fields. The first field is labeled '提出日' (Submission Date) and has an error message '入力規則エラーメッセージ'. The second field is labeled '申請者' (Applicant) and also has an error message. The right panel is titled 'テキスト入力要素' (Text Input Element) and shows its properties. The 'disabled' property is set to '固定値' (Fixed Value) with the value '\$inputPageT'. Other properties like 'readonly', 'static', 'autofocus', 'name', 'placeholder', 'value', 'emptyValueToNull', 'maxLength', 'color', and 'size' are also listed.

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。

The screenshot shows the 'Actions' tab selected in the top navigation bar. The left sidebar is identical to the previous screenshot. The right panel lists several actions: Initialize, Return to source screen, Apply, Temporarily save, Approve, Reapply, Confirm, and Apply from draft. The 'Initialize' action is highlighted with a red border. The main workspace shows the same two text input fields as before.

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」を右側のスペースへドラッグ&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○が○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。

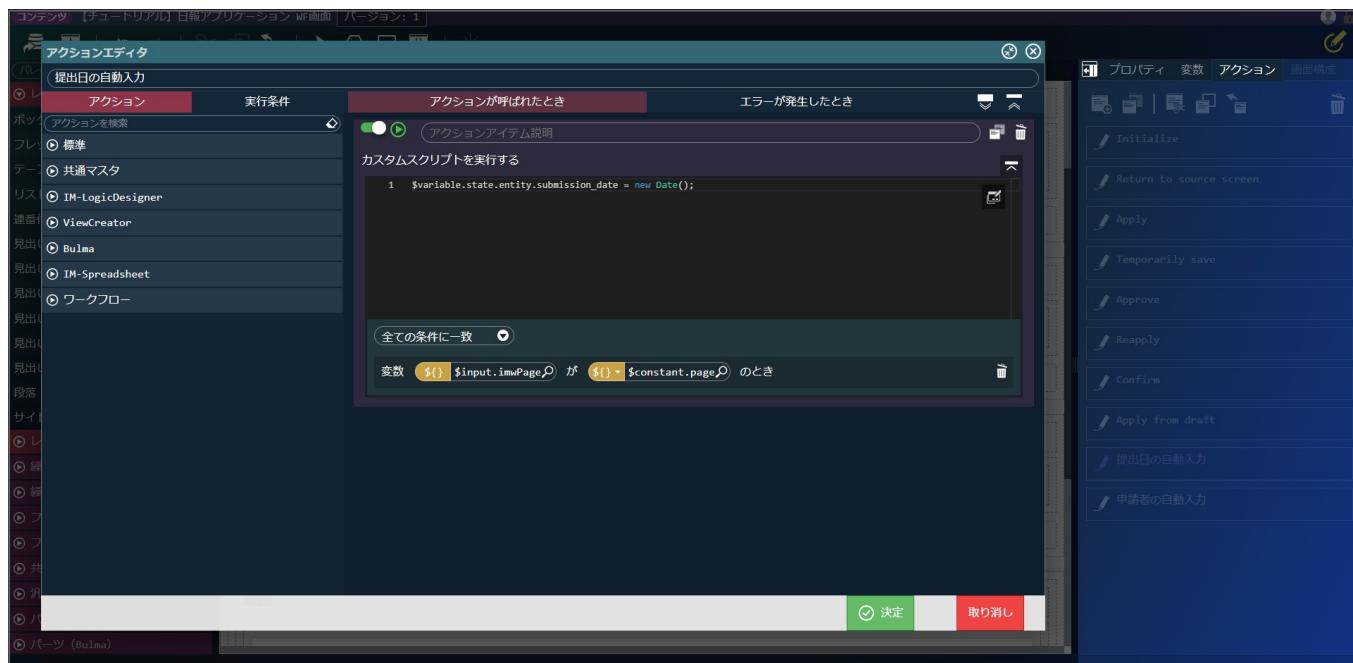
変数の右側にある  をクリックし、セレクタを表示させます。

「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。



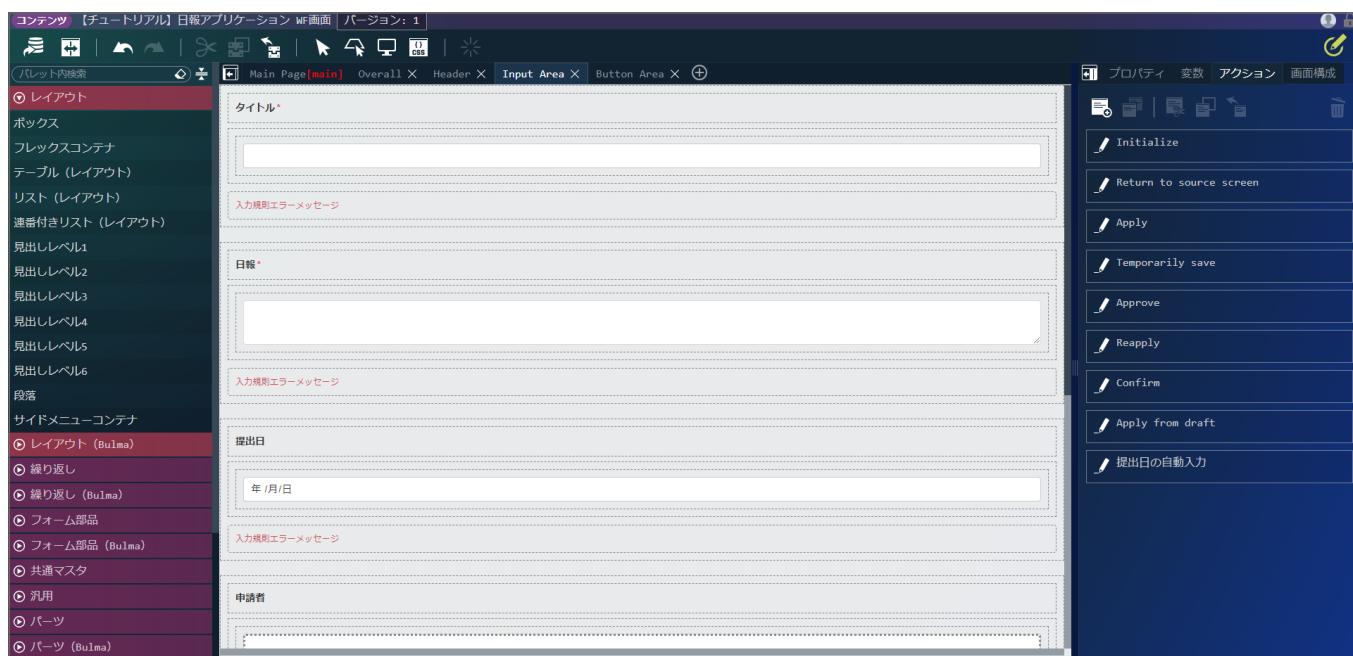
「決定」をクリックし、アクションエディタを閉じて編集を終了してください。

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」を右側のスペースへドラッグ＆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。

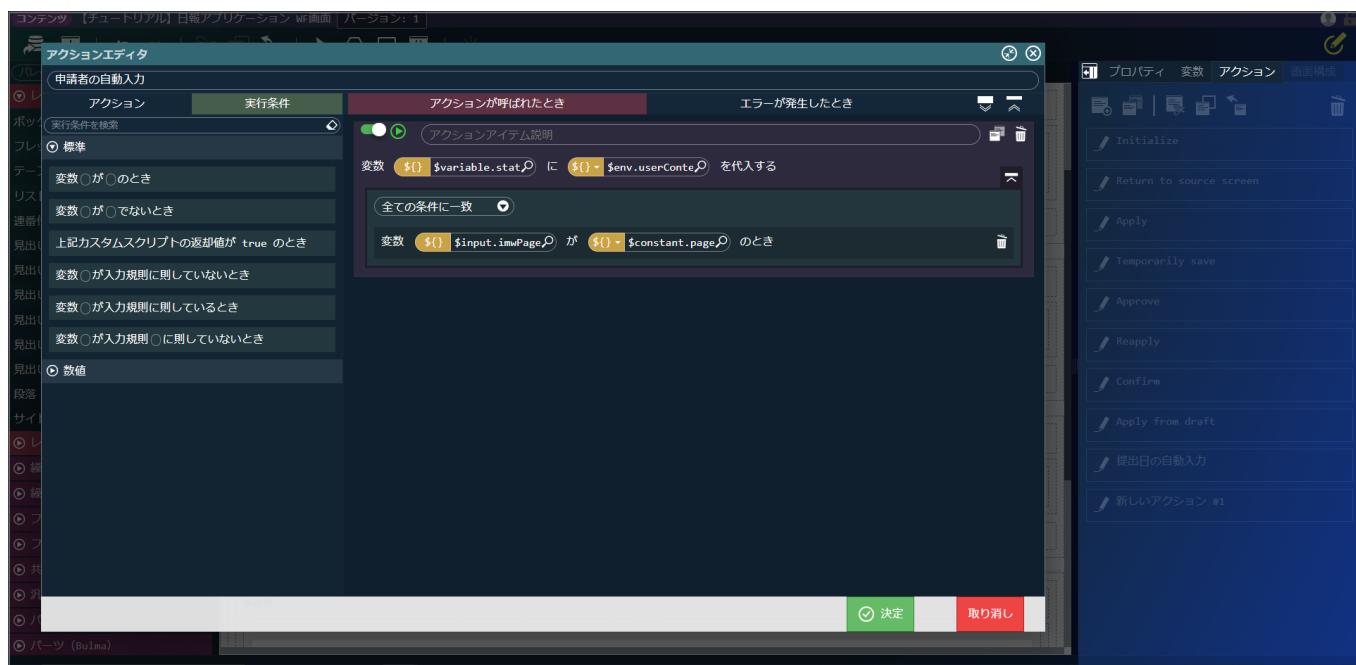
変数の右側にある  をクリックし、セレクタを表示させます。

「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。



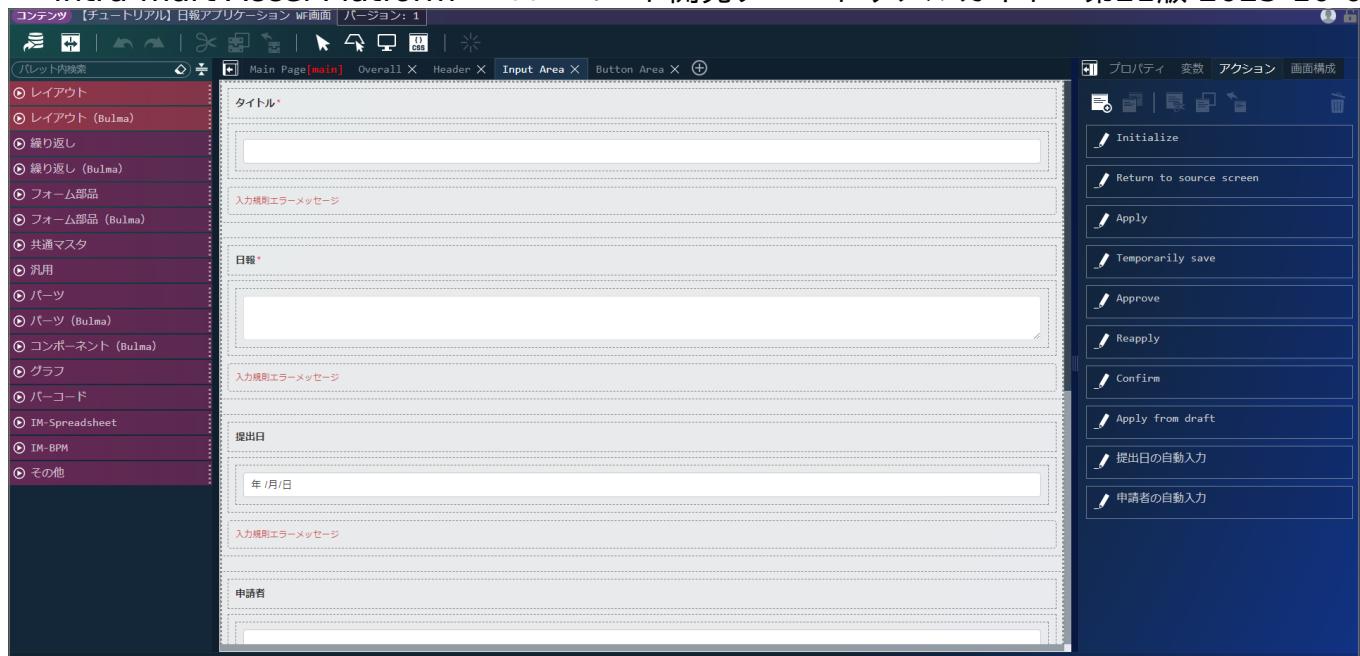
「決定」をクリックし、アクションエディタを閉じて編集を終了してください。

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。

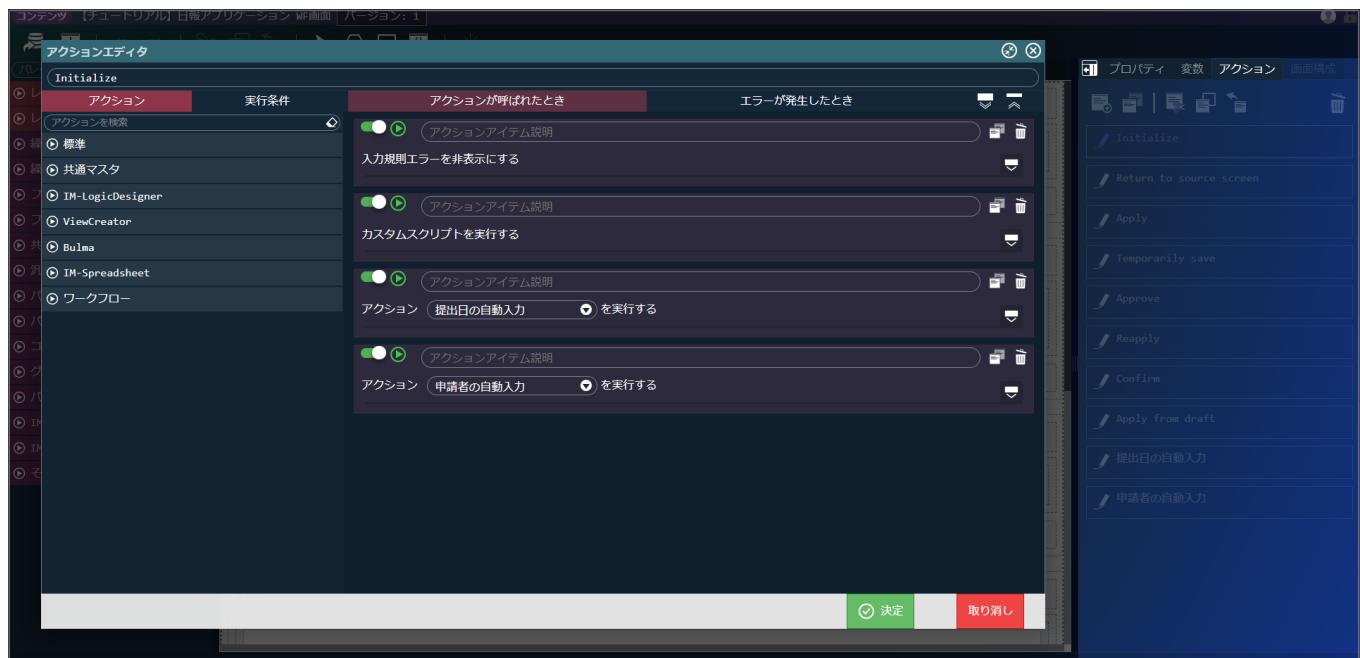
画面右側に表示されている「アクション」タブを選択します。

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へドラッグ＆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へドラッグ＆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。

申請画面

タイトル*

日報*

提出日
2023/10/24

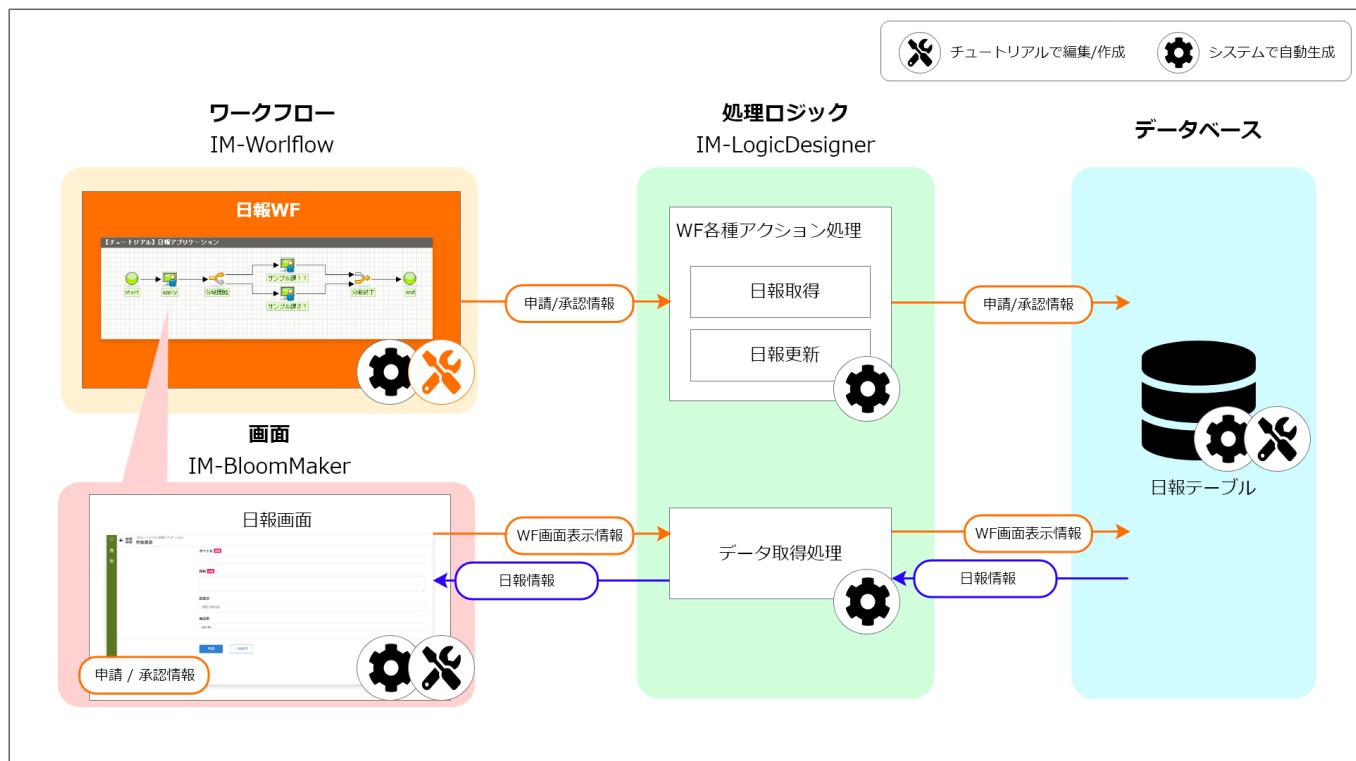
申請者
tenant

申請 一時保存

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、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。

The screenshot shows the 'Workflow Management' section of the platform. At the top, there's a breadcrumb navigation: 'アプリケーション管理' > '【チュートリアル】日報アプリケーション - 詳細'. Below the navigation, there are tabs: ALL, 画面, 公開URL, ロジック, **ワークフロー**, データ定義, and ユーティリティ. On the right side of the header, there are links for 'リソース関連図', '権限設定', 'ログ', '設定', and 'リソースを追加'. A message at the bottom right indicates the tenant was last updated on 2024/02/22 16:54:16.

ルートの修正

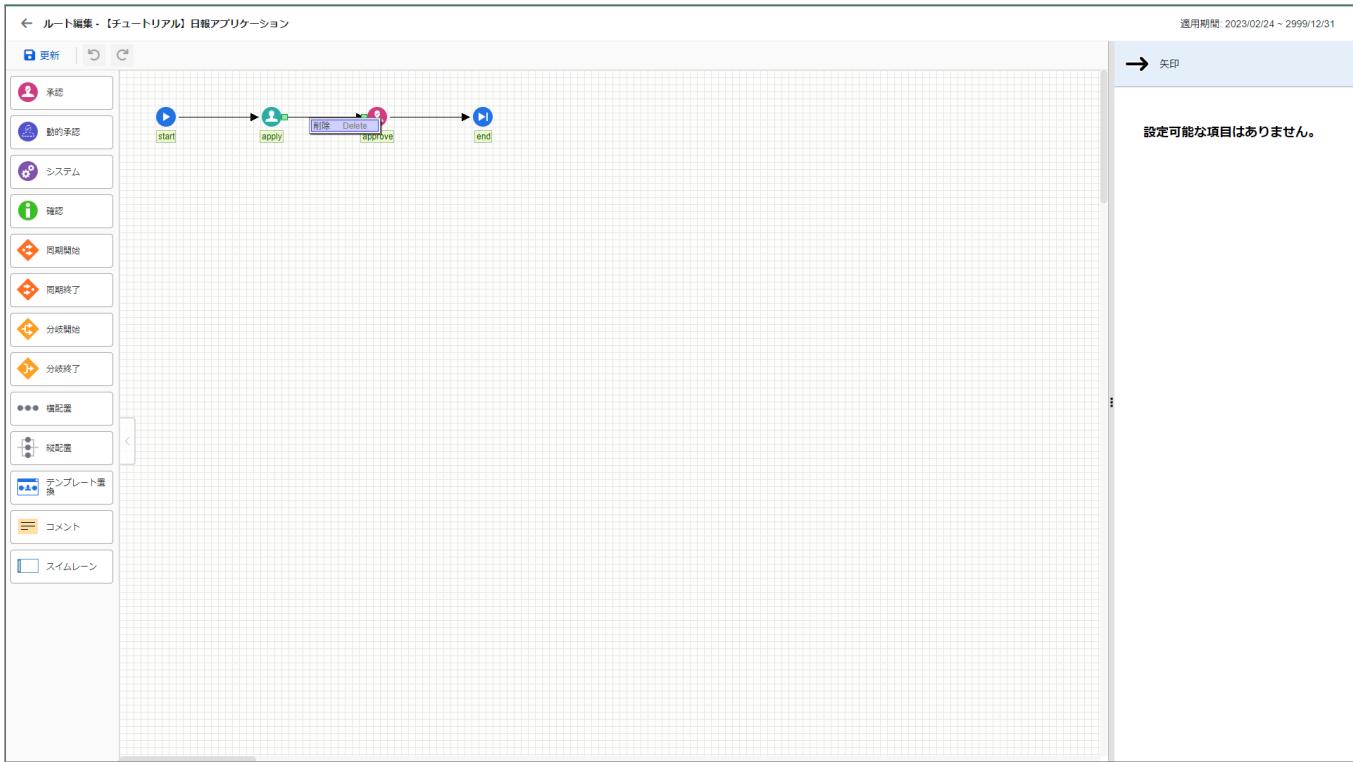
まずは、ワークフローのルートを修正しましょう。

画面左上の「ルート編集」をクリックします。

The screenshot shows the 'Route Editor' for the workflow. At the top, there's a toolbar with '保存', '差分比較', and 'ルート編集'. The main area displays a visual representation of the workflow route: start → apply → approve → end. Below this diagram, there's a 'ノード一覧' (Node List) table showing nodes like 'start', 'apply', 'approve', and 'end'. To the right of the diagram, there's a large configuration panel titled 'フロー設定' (Flow Settings) with tabs for '基本情報', '詳細設定', and '参照者設定'. The '基本情報' tab is selected, showing sections for '画面' (PC/Mobile), '処理' (Handling), and '通知' (Notification). Each section contains various configuration options and URL links.

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。

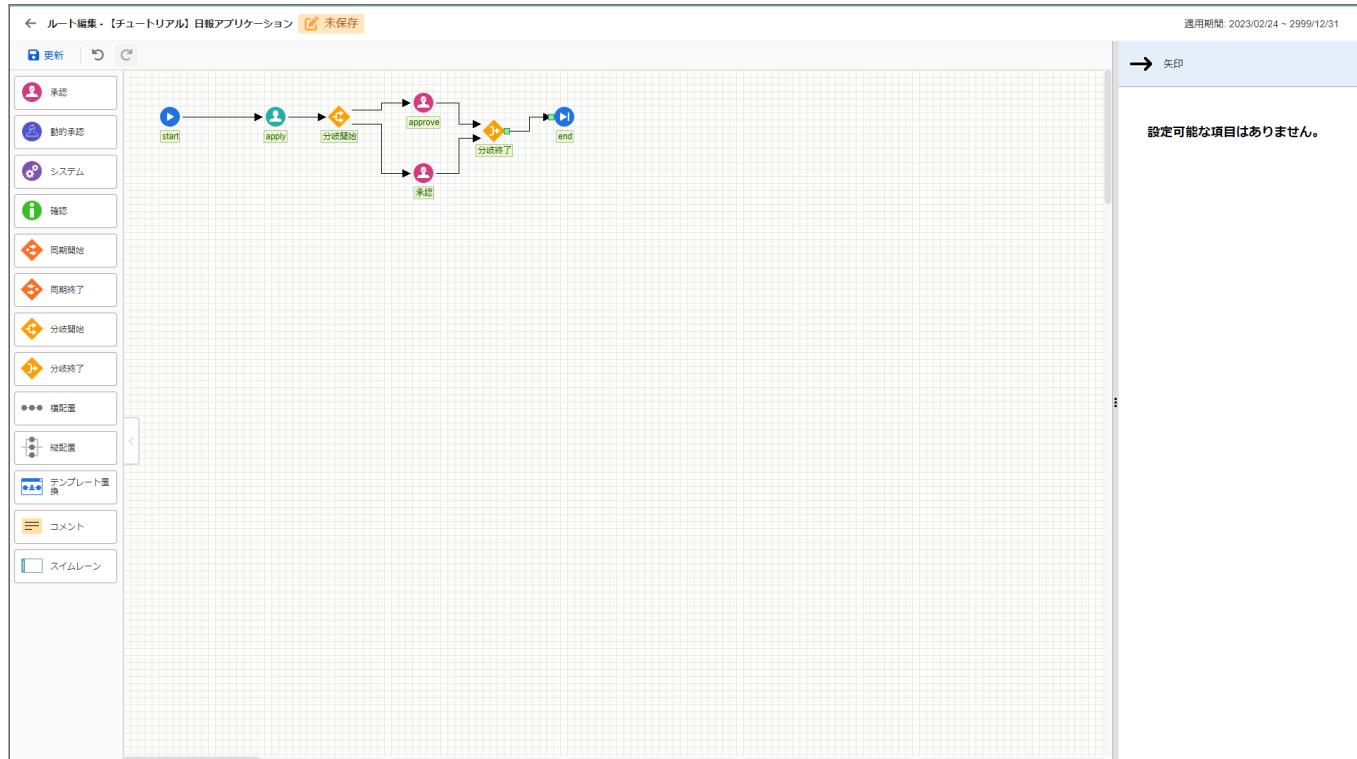
approveからendに繋がっている矢印を右クリックし、削除します。



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、各ノードを下の図のように矢印で繋ぎます。



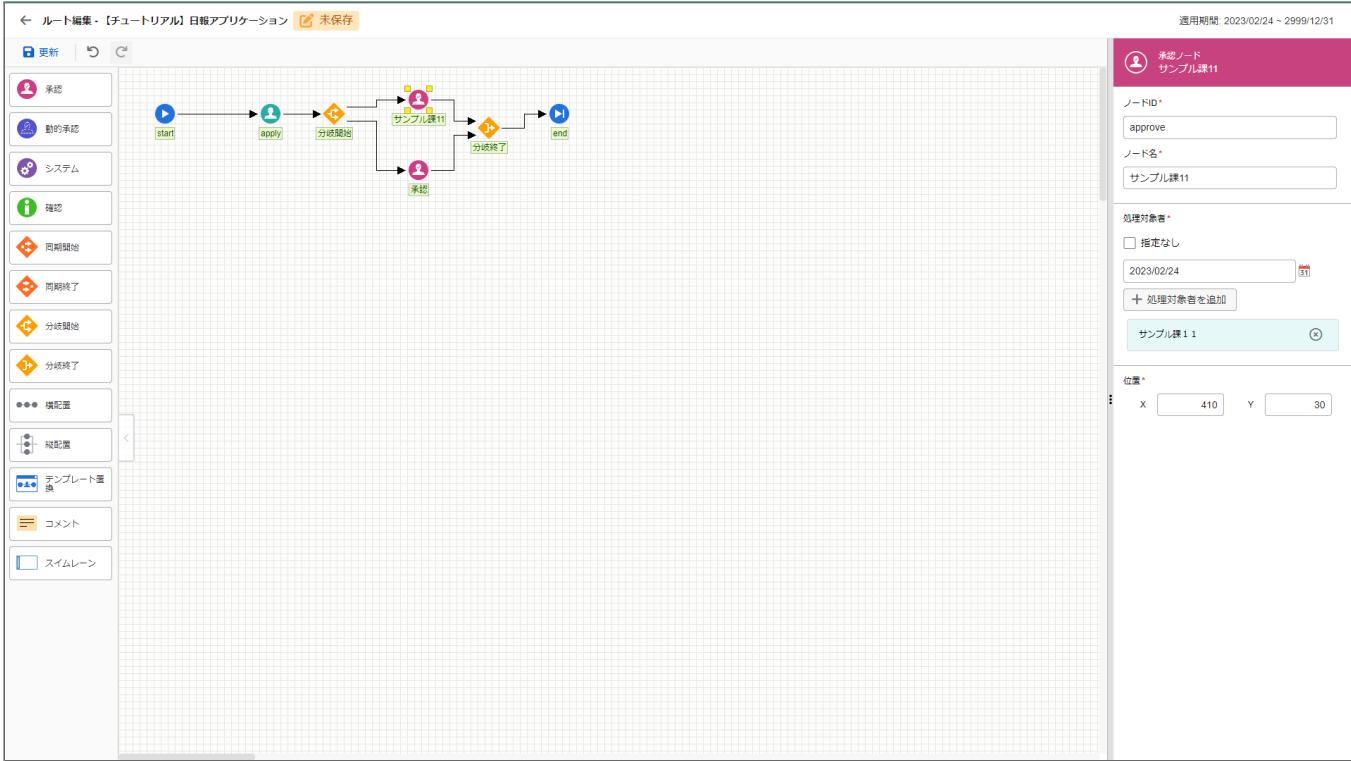
ノード詳細の修正

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11」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01」配下の「サンプル課11」を選択し「決定」をクリックします。

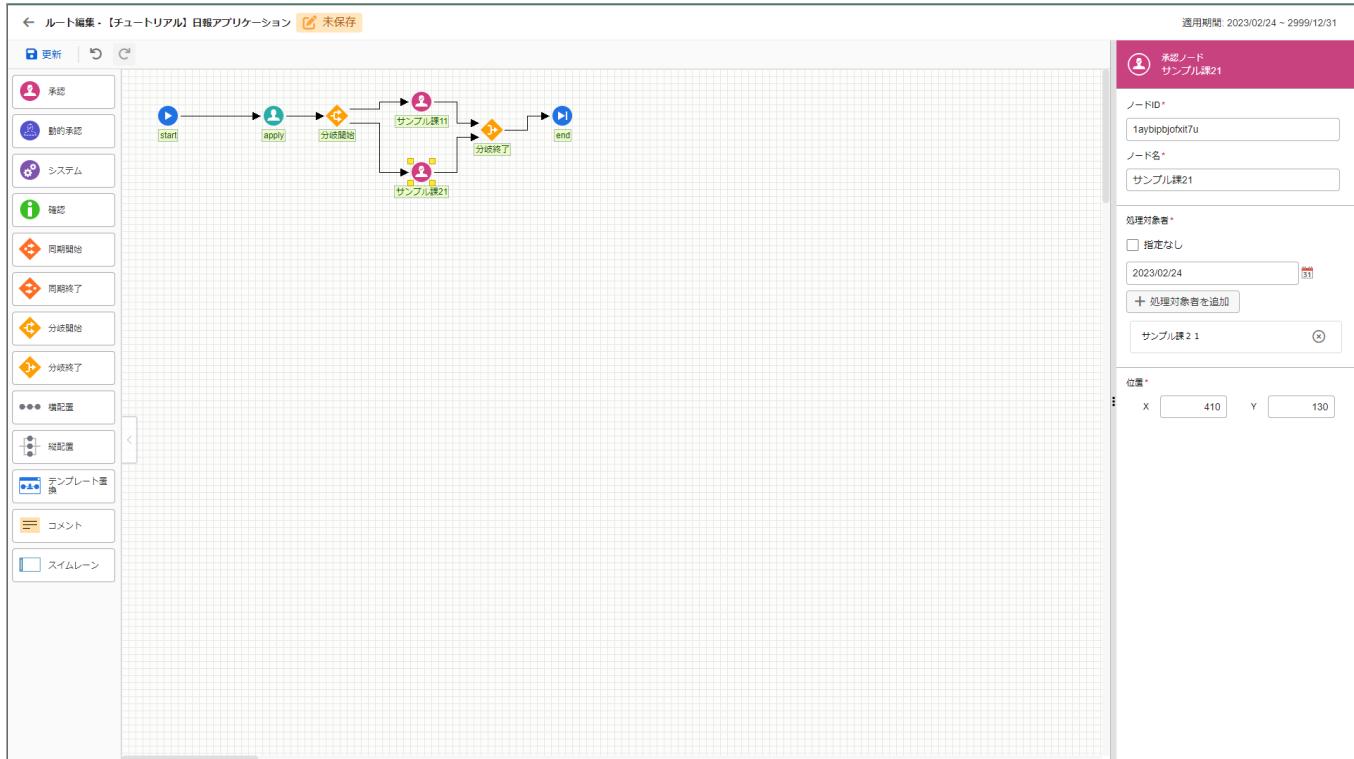


承認ノードをクリックして選択します。

「ノード名」に「サンプル課21」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02」配下の「サンプル課21」を選択し「決定」をクリックします。



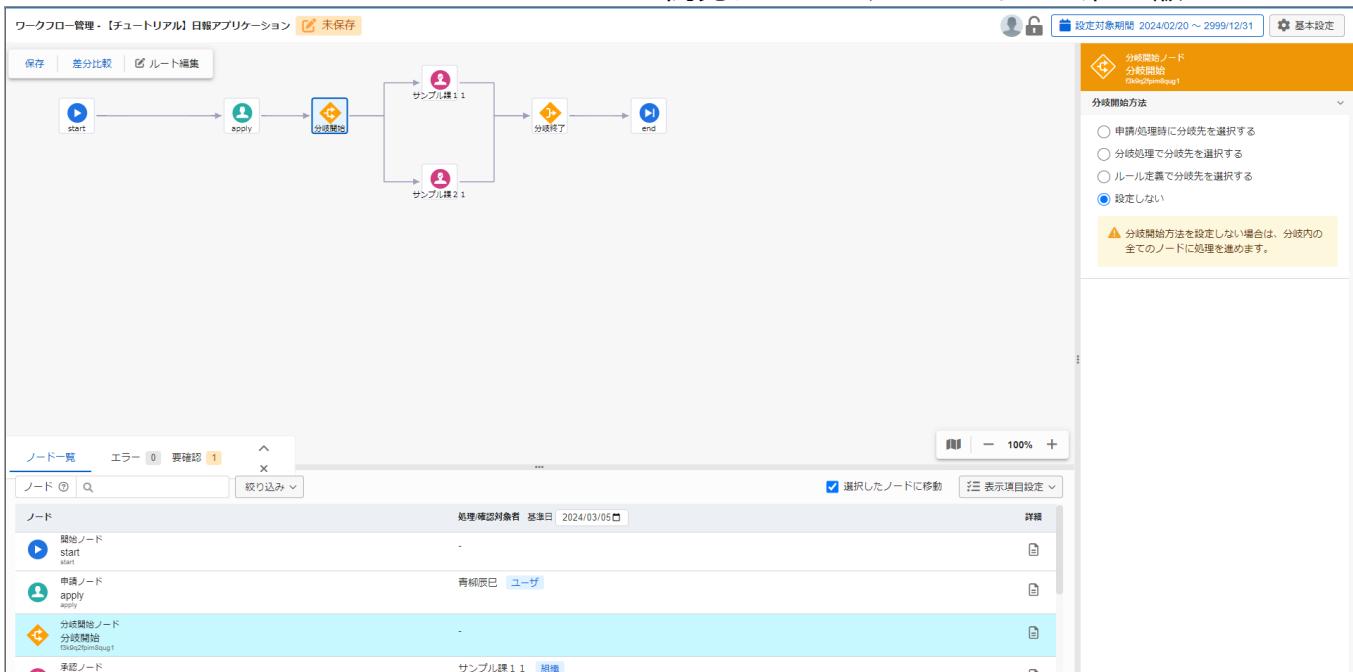
画面左上の「更新」をクリックし、「←(戻る)」アイコンでワークフロー管理画面に戻ります。

分岐開始方法の設定

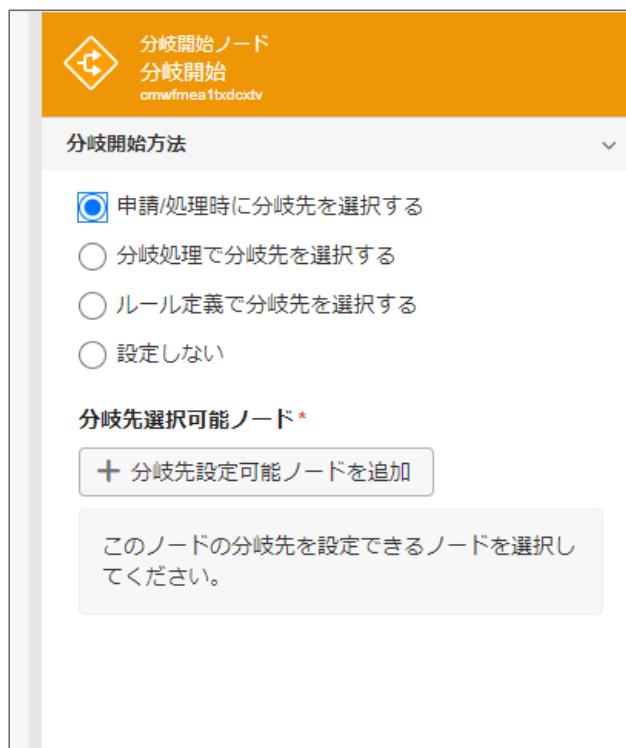
次は、分岐開始方法の設定を行います。

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。

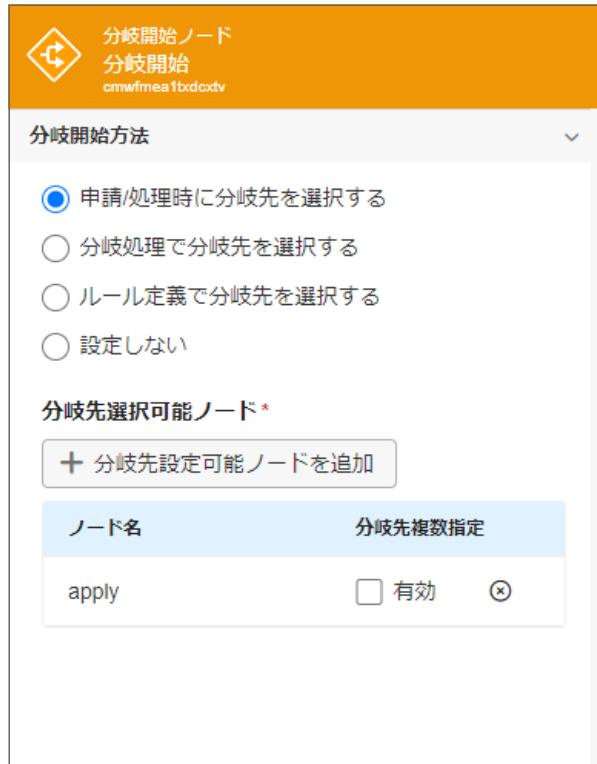


分岐開始方法で「申請／処理時に分岐先を選択する」を選択します。



「+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。



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ー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。

| 名前 | 説明 | 最終更新日 |
|---|--|----------------------------|
| エンティティ (ID: im-repository/entity/tutorial_daily_report_app_entity) | 【チュートリアル】エディティ 日報アプリケーション 業務テンプレート「シンプルなワークフロー」で作成されたエンティティです。 | tenant 2024/02/20 21:10:21 |
| ワークフロー-定義 (ID: im-workflow/flowdefinitions/8h43gj1ubdnc4-flow) | 【チュートリアル】日報アプリケーション 業務テンプレート「シンプルなワークフロー」で作成されたフロー定義です。 | tenant 2024/02/22 16:54:16 |
| ロジックフロー (ID: im-logic/flows/tutorial_daily_report_app-wf-action) | 【チュートリアル】日報アプリケーション WFアクション処理 業務テンプレート「シンプルなワークフロー」で作成されたフロー定義です。 | tenant 2024/02/20 21:10:21 |
| ロジックフロー (ID: im-logic/flows/tutorial_daily_report_app-wf-delete-completedmatters) | 【チュートリアル】日報アプリケーション WF未完了案件削除処理 業務テンプレート「シンプルなワークフロー」で作成されたフロー定義です。 | tenant 2024/02/20 21:10:21 |
| ロジックフロー (ID: im-logic/flows/tutorial_daily_report_app-wf-delete-activematters) | 【チュートリアル】日報アプリケーション WF未完了案件削除処理 業務テンプレート「シンプルなワークフロー」で作成されたフロー定義です。 | tenant 2024/02/20 21:10:21 |
| 画面URL (ID: im-bloommaker/routes/tutorial_daily_report_app-wf-route) | 【チュートリアル】日報アプリケーション WF画面 業務テンプレート「シンプルなワークフロー」で作成されたルーティング定義です。 | tenant 2024/02/20 21:10:21 |
| 画面コンテンツ (ID: im-bloommaker/contents/tutorial_daily_report_app-wf-content) | 【チュートリアル】日報アプリケーション WF画面 業務テンプレート「シンプルなワークフロー」で作成されたコンテンツです。 | tenant 2024/02/22 15:57:41 |
| ロジックフロー (ID: im-logic/flows/tutorial_daily_report_app-wf-delete-archivematters) | 【チュートリアル】日報アプリケーション WF過去案件削除処理 業務テンプレート「シンプルなワークフロー」で作成されたフロー定義です。 | tenant 2024/02/20 21:10:21 |
| ロジックフロー (ID: im-logic/flows/tutorial_daily_report_app-delete-user-data) | 【チュートリアル】日報アプリケーション ユーザー削除処理 業務テンプレート「シンプルなワークフロー」で作成されたフロー定義です。 | tenant 2024/02/20 21:10:21 |

「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。



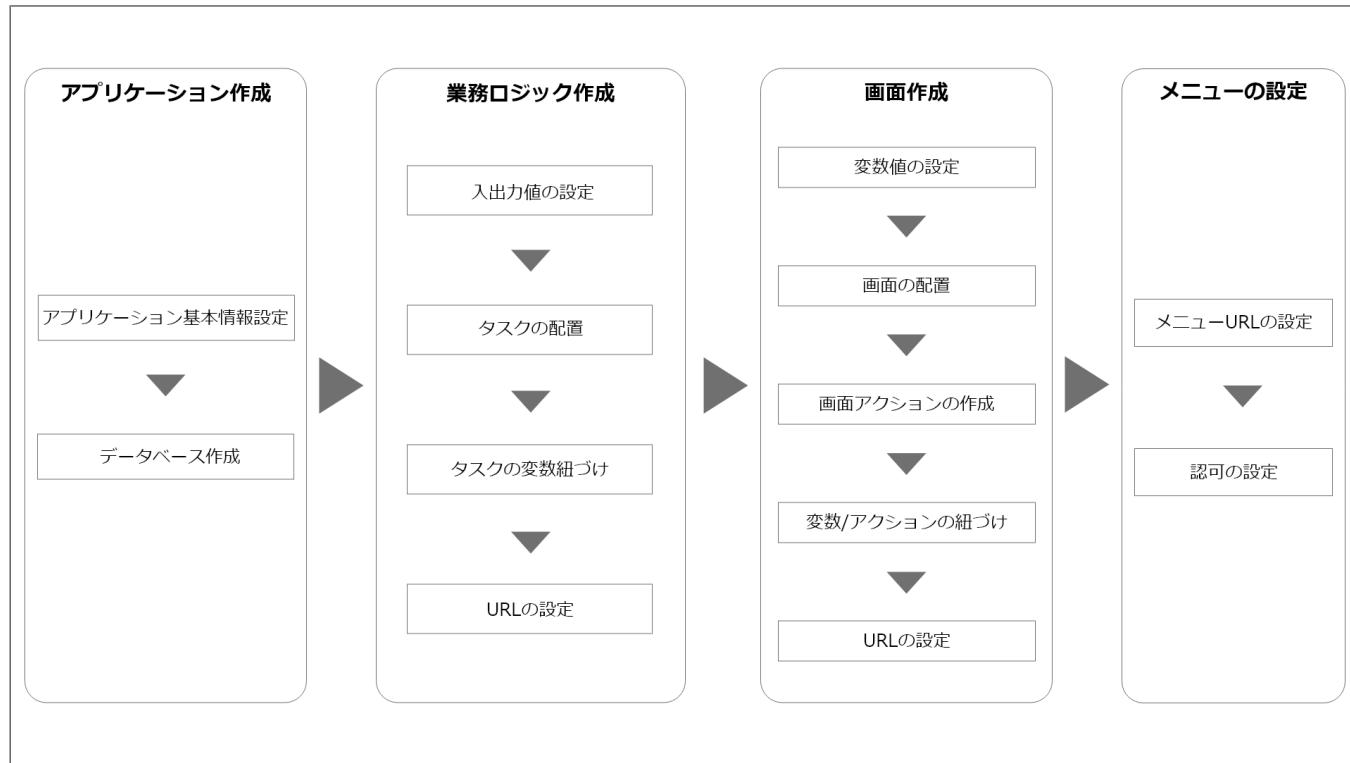
注意

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。

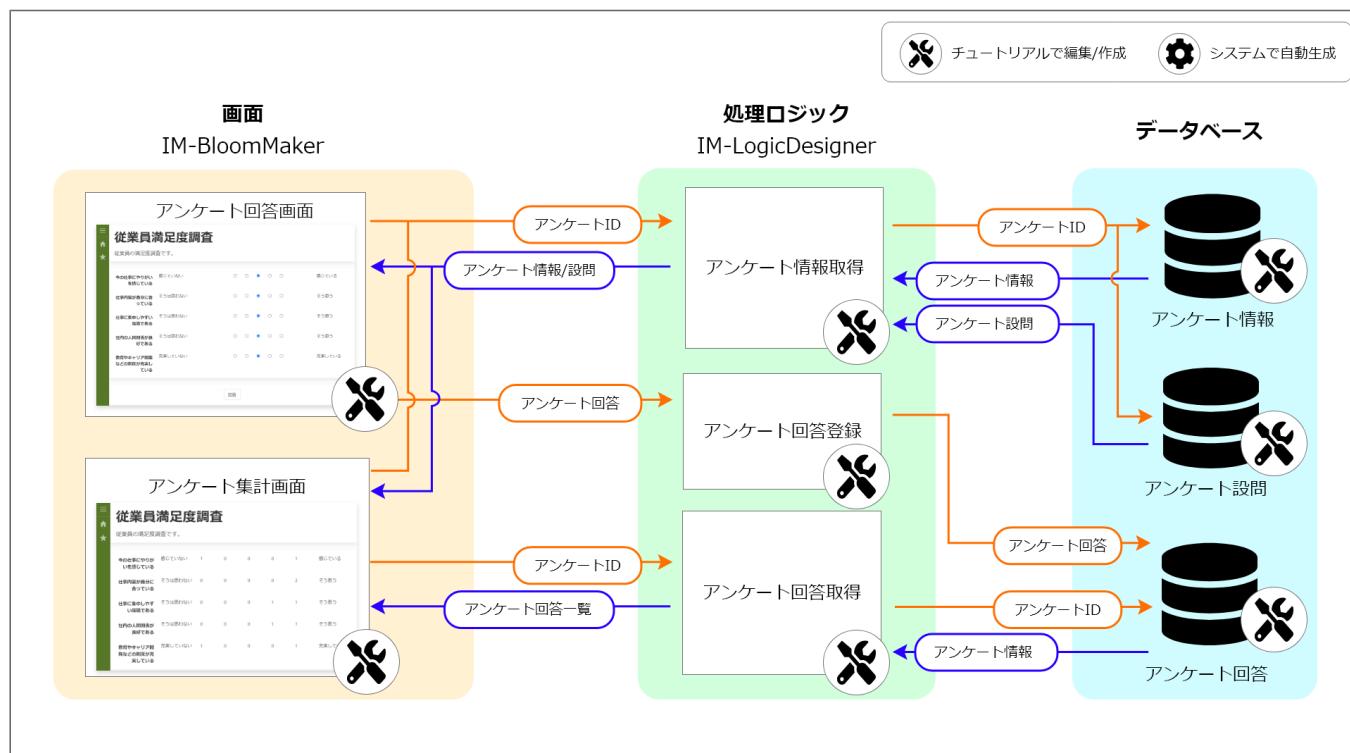
詳しくは上記リンクからご確認ください。

作成の流れ

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を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。

The screenshot shows a survey titled "従業員満足度調査" (Employee Satisfaction Survey). On the left is a vertical toolbar with icons for Home, Star, and Products. The main area displays five questions with 5-point Likert scale options from "感じていない" (Not at all) to "感じている" (Very much), with "感じている" being the selected option for all questions. A "回答" (Answer) button is at the bottom.

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

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、回答内容を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 | 0 | 0 | 2 | 3 | 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 | 1 | 3 | 1 | 1 | 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 | 1 | 3 | 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 | 1 | 0 | 2 | 2 | 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 | 0 | 0 | 5 | 1 | 0 | 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。

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

「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「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。

アプリケーション作成
業務テンプレート一覧

intra-mart 標準アプリケーションテンプレート

マスタメンテナンス
マスタメンテナンスのテンプレートです。

[次へ](#)

シンプルなワークフロー
シンプルなワークフローのテンプレートです。

[次へ](#)

スプレッドシート型一覧
スプレッドシート型一覧のテンプレートでは Excel のテーブルのような一覧画面を提供します。また、一覧データを xlsx 形式のファイルとしてエクスポートすることができます。※このテンプレートは IM-Spreadsheet が含まれていない環境では利用できません。

[次へ](#)

空のアプリケーション
空のアプリケーションを作成します。

[次へ](#)

intra-mart サンプルアプリケーション

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*

アプリケーション情報

① 作成する資材をまとめて1つのアプリケーションとして登録します。
作成するアプリケーション情報を入力してください。
このテンプレートは画面やロジックを生成しません。生成した後、必要に応じてリソースを登録してください。

作成するアプリケーションID* また、アプリケーションを一度作成した後にアプリケーションIDを変更することはできません。
半角英数字・(ハイフン)・(アンダースコア)で入力してください。

アプリケーション名*

説明

アプリケーション情報の設定

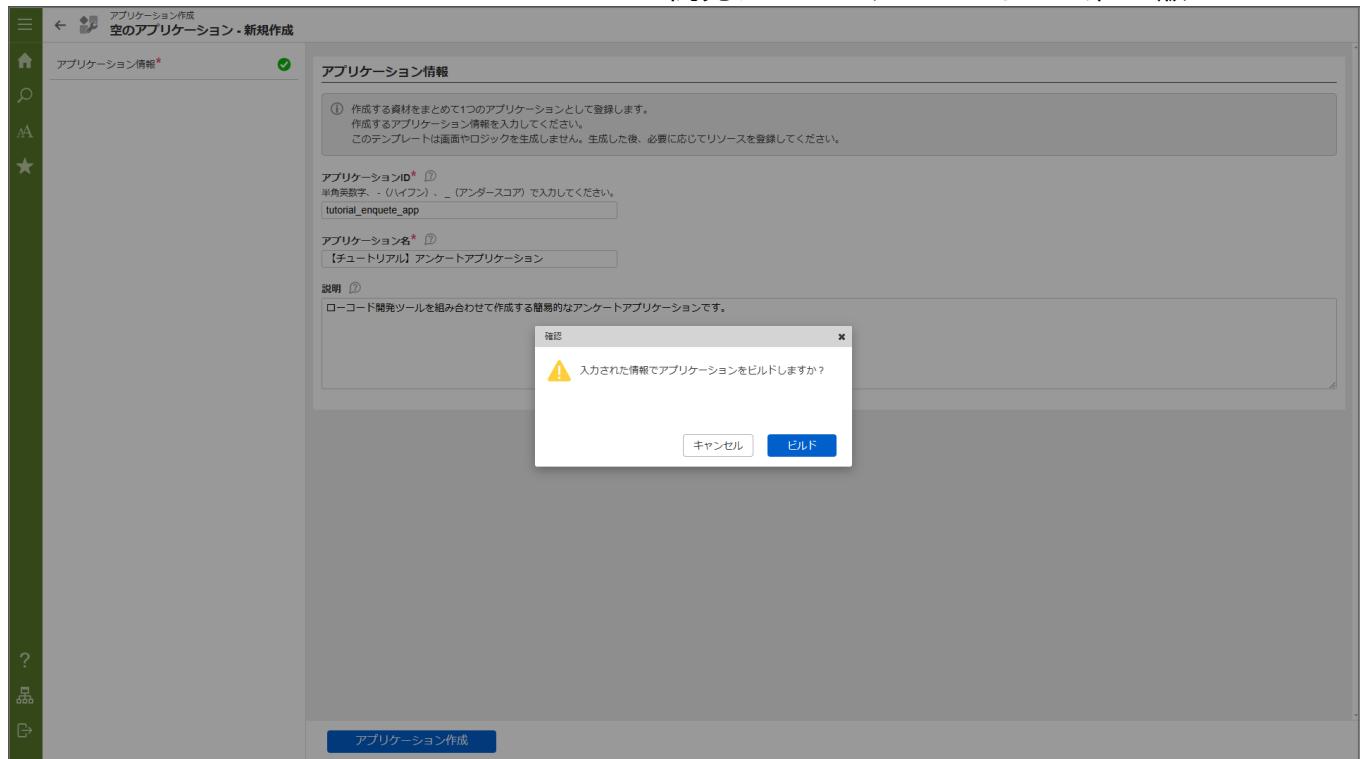
アプリケーション情報を設定します。

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。

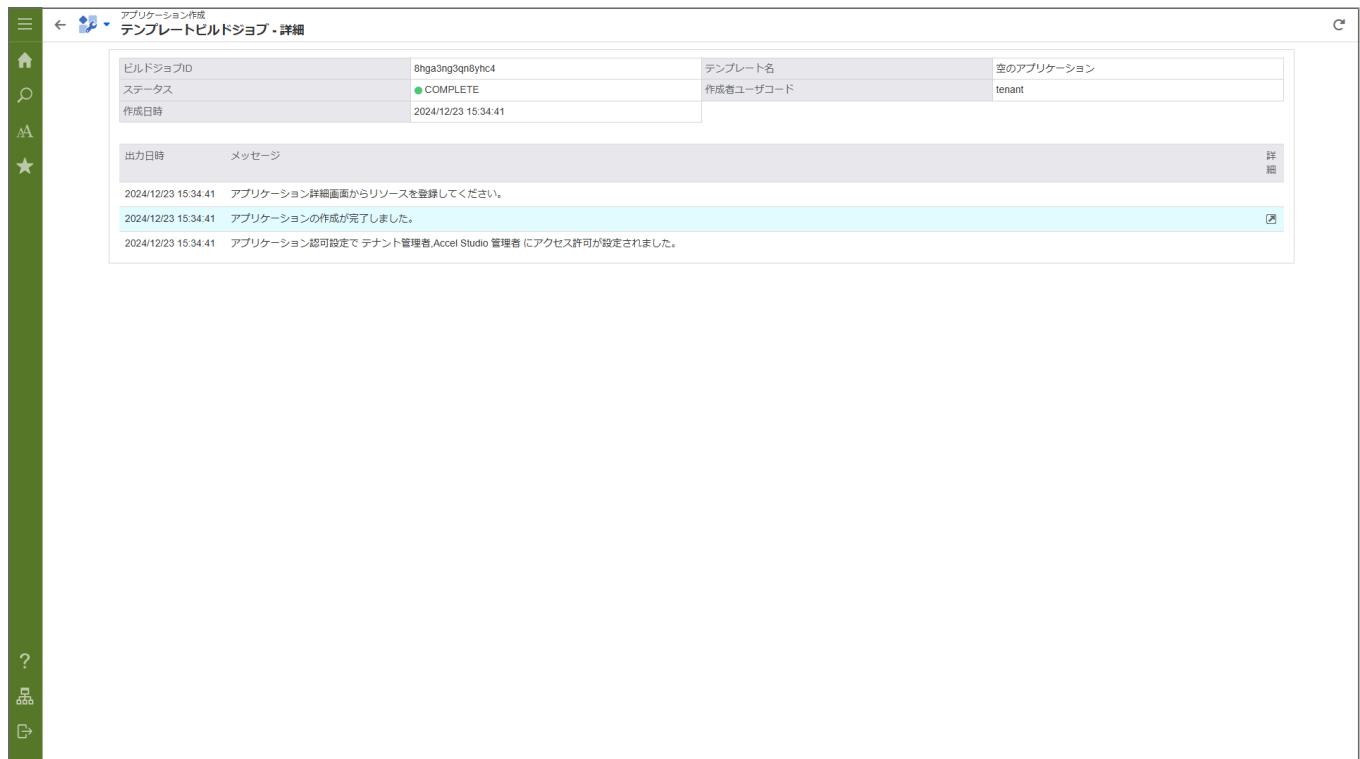
*アプリケーション情報に関しては、各項目に任意の入力値を設定してもかまいません。本チュートリアルは、下記の通りのアプリケーションID、アプリケーション名を入力した前提で進めます。

| 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て作成する簡単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。



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。

- アンケート情報テーブル
- アンケート設問テーブル
- アンケート回答テーブル



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。

エンティティを作成

アプリケーションのビルト完了画面の「アプリケーションの作成が完了しました。」横にある「詳細」アイコンをクリックして、アプリケーション管理画面に遷移してください。

右上プルダウンの「リソースを追加」 - 「データ定義」 - 「データ定義を新規作成」 - 「エンティティ」を選択し、「データ定義-新規作成画面」を表示します。

The screenshot shows the 'Application Management' section of the Accel Platform. In the top navigation bar, under 'Tutorial Application - Details', there is a dropdown menu labeled 'Resource Add' with 'Entity' selected. A red box highlights the 'Entity' button in this dropdown. On the right side of the screen, there is a sidebar titled 'Application History' containing details about the application, such as its name, description, and last run information.

まずは、アンケートの基本情報を保持する「アンケート情報」エンティティを作成しましょう。

エンティティID、エンティティ名、エンティティに紐づくテーブル名に以下の入力値を設定してください。

| ラベル | 入力値 |
|-----------------|----------------------------------|
| エンティティID | tutorial_enquete_info_app_entity |
| エンティティ名 | アンケート情報 |
| エンティティに紐づくテーブル名 | tutorial_enquete_info_app |

「登録先エンティティカテゴリ」の虫眼鏡アイコンをクリックして「エンティティカテゴリ検索」ウインドウを開き、「カテゴリ新規追加」から以下の入力値でカテゴリを新規作成して設定します。

| ラベル | 入力値 |
|--------|------------------------|
| カテゴリID | tutorial_entity |
| カテゴリ名 | 【チュートリアル】チュートリアルエンティティ |

IM-Repository
データ定義・新規作成

エンティティの定義*

① 指定したデータでエンティティを作成します。

基本情報

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

エンティティ名*
アンケート情報

登録先エンティティカテゴリ*
【チュートリアル】チュートリアル

データベース種別 ②
 TENANT
 SHARED

エンティティに紐づくテーブル名* ③
ヘルプアイコンから入力規則を確認できます。
tutorial_enquete_info_app

項目設定

+ 追加 + 既存の辞書項目から選択 + 新規作成する辞書項目を参照

| 表示名 * | 詳細 | 物理名 * | データ型 * | 最大桁数 | 主キー | 必須 | 削除 |
|-----------------------|----|-------|--------|------|-----|----|----|
| ④ エンティティ情報が登録されていません。 | | | | | | | |

サブエンティティの利用指定

⑤ メインエンティティに紐づくサブエンティティの数を指定します。

サブエンティティの利用数

新規作成

上記の設定で、エンティティに紐づく「tutorial_enquete_info_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て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。

次に、エンティティの項目を設定していきましょう。「項目設定」 - 「+追加」を選択してください。新規項目欄が追加されます。

項目設定

+ 追加 + 既存の辞書項目から選択 + 新規作成する辞書項目を参照

| 表示名 * | 詳細 | 物理名 * | データ型 * | 最大桁数 | 主キー | 必須 | 削除 |
|----------------------|---------------------------------|----------------------|---------------------------------|---------------------------------|--------------------------|--------------------------|---------------------------------|
| <input type="text"/> | <input type="button" value=""/> | <input type="text"/> | <input type="button" value=""/> | <input type="button" value=""/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="button" value=""/> |

「表示名」に「アンケートID」、「物理名」に「enquete_id」、「データ型」は「文字列」、「最大桁数」に「200」と設定してください。

続けて、「詳細」の歯車アイコンをクリックして「辞書項目新規作成」ダイアログを表示し、辞書項目IDを入力しましょう。

| 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】アンケート辞書 | 任意のカテゴリで構いません。本チュートリアルでは「【チュートリアル】アンケート辞書」カテゴリを作成しています。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_enquete_id | |
| 辞書項目名 | アンケート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 | enquete_id | 画面に表示される項目名にも利用されます。 |

「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】アンケート辞書」カテゴリを新規作成しています。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。
 「辞書項目ID」の設定が完了したら「設定」ボタンをクリックしてください。

最後に、「主キー」にチェックを入れ、「アンケートID」を主キーとしてください。

これで、1つ目の項目「アンケート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。

| 物理名/変数名 | 表示名/辞書項目名 | 辞書項目ID | 型 | サイズ | 主キー | 必須 |
|------------------|------------|-----------------------------|-----|------|--------------------------|--------------------------|
| enquete_id | アンケートID | tutorial_enquete_id | 文字列 | 200 | <input type="checkbox"/> | <input type="checkbox"/> |
| enquete_title | アンケートのタイトル | tutorial_enquete_title | 文字列 | 2000 | <input type="checkbox"/> | |
| enquete_overview | アンケートの概要 | tutorial_enquete_overview | 文字列 | 2000 | <input type="checkbox"/> | |
| start_date | 開始日 | tutorial_enquete_start_date | 日付 | | <input type="checkbox"/> | |
| end_date | 終了日 | tutorial_enquete_end_date | 日付 | | <input type="checkbox"/> | |

エンティティの定義

① 指定したデータでエンティティを作成します。

基本情報

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

エンティティ名*
アンケート情報

登録先エンティティカテゴリ*
【チュートリアル】チュートリアル

データベース種別 ②
● TENANT
○ SHARED

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_enquete_info_app

項目設定

| 表示名* | 詳細 | 物理名* | データ型* | 最大桁数 | 主キー | 必須 | 削除 |
|------------|------------------|------|-------|-------------------------------------|-------------------------------------|--------------------------|----|
| アンケートID | enqueue_id | 文字列 | 200 | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | 有 |
| アンケートのタイトル | enqueue_title | 文字列 | 2000 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | 有 |
| アンケートの概要 | enqueue_overview | 文字列 | 2000 | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | 有 |
| 開始日 | start_date | 日付 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | 有 |
| 終了日 | end_date | 日付 | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | 有 |

新規作成

テーブル情報の入力まで完了したらエンティティの新規作成の準備が完了です。「新規作成」をクリックして設定を保存してください。

同様に、「アンケート設問」と「アンケート回答」のエンティティも作成しましょう。

「リソースを追加」 - 「データ定義」 - 「データ定義を新規作成」 - 「エンティティ」を選択し、「データ定義-新規作成画面」から、以下の表の通りにそれぞれエンティティを作成してください。

なお、「アンケートID」は「+追加」ではなく「+既存の辞書項目から選択」から、先ほど作成した項目を指定します。

■ 「アンケート設問」エンティティ

| ラベル | 入力値 |
|-----------------|--------------------------------------|
| エンティティID | tutorial_enquete_question_app_entity |
| エンティティ名 | アンケート設問 |
| 登録先エンティティカテゴリ | 【チュートリアル】チュートリアルエンティティ |
| エンティティに紐づくテーブル名 | tutorial_enquete_question_app |

| 物理名/変数名 | 表示名/辞書項目名 | 辞書項目ID | 型 | サイズ | 主キー | 必須 |
|-------------|-----------|------------------------------|-----|------|--------------------------|--------------------------|
| enqueue_id | アンケートID | tutorial_enquete_id | 文字列 | 200 | <input type="checkbox"/> | <input type="checkbox"/> |
| question_id | 設問ID | tutorial_enquete_question_id | 文字列 | 200 | <input type="checkbox"/> | <input type="checkbox"/> |
| question_no | 設問番号 | tutorial_enquete_question_no | 数値 | 5 | <input type="checkbox"/> | <input type="checkbox"/> |
| question | 設問内容 | tutorial_enquete_question | 文字列 | 2000 | <input type="checkbox"/> | <input type="checkbox"/> |
| left_scale | 回答項目の左端 | tutorial_enquete_left_scale | 文字列 | 2000 | <input type="checkbox"/> | <input type="checkbox"/> |
| right_scale | 回答項目の右端 | tutorial_enquete_right_scale | 文字列 | 2000 | <input type="checkbox"/> | <input type="checkbox"/> |

■ 「アンケート回答」エンティティ

| ラベル | 入力値 |
|----------|------------------------------------|
| エンティティID | tutorial_enquete_answer_app_entity |
| エンティティ名 | アンケート回答 |

| ラベル | 入力値 |
|-----------------|-----------------------------|
| 登録先エンティティカテゴリ | 【チュートリアル】チュートリアルエンティティ |
| エンティティに紐づくテーブル名 | tutorial_enquete_answer_app |

| 物理名/変数名 | 表示名/辞書項目名 | 辞書項目ID | 型 | サイズ | 主キー | 必須 |
|-------------|-----------|------------------------------|-----|------|-----|----|
| answer_id | 回答ID | tutorial_enquete_answer_id | 文字列 | 200 | □ | □ |
| enquete_id | アンケートID | tutorial_enquete_id | 文字列 | 200 | □ | □ |
| respondent | 回答者 | tutorial_enquete_respondent | 文字列 | 200 | | □ |
| answer_date | 回答日時 | tutorial_enquete_answer_date | 日付 | | | □ |
| answers | 回答 | tutorial_enquete_answers | 文字列 | 2000 | | □ |

検索条件の設定

作成したエンティティには、検索条件が設定されていません。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できます。

「アンケート設問」「アンケート回答」の2つのエンティティに検索条件を設定します。

まずは、アプリケーション管理画面のアプリケーション詳細で「アンケート設問」を選択してください。

The screenshot shows the application management interface for the 'Tutorial Application' (チュートリアル アプリケーション). The left sidebar has icons for Home, Search, and Application Management. The main area is titled 'Application Management [Tutorial Application] Application Details'. The top navigation bar includes 'Git連携', 'リソース関連図', '権限設定', 'ログ', and a settings gear icon.

The central part of the screen displays the 'Data Definition' tab under the 'Entity' section. It lists three entities:

- アンケート回答 (tutorial_enquete_answer_app_entity)
- アンケート設問 (tutorial_enquete_question_app_entity) - This entity is highlighted with a red box.
- アンケート情報 (tutorial_enquete_info_app_entity)

To the right of the entities, there is a 'Application History' panel showing the creation of the 'アンケート設問' entity. The details are as follows:

- Entity Name: アンケート設問
- Entity ID: tutorial_enquete_question_app_entity
- Type: データベース種別 (Database Type)
- Table Name: tutorial_enquete_question_app
- Action: 編集画面を開く (Open Edit Form)
- Execution User: tenant
- Execution Date: 2024/12/23 17:33:25

Below the history, there are sections for 'Resource Association' and 'Table Definition' creation, each with their own entity details.

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。

The screenshot shows the 'Data Definition' screen. At the top, there are tabs for 'エンティティ' (Entity), 'カテゴリ新規追加' (New Category Addition), 'エンティティ新規作成' (New Entity Creation), 'エンティティコピー作成' (Entity Copy Creation), and '辞書項目一覧' (Dictionary Item List). Below these are search and filter fields, and a message indicating that entity and table information are consistent. A large table lists table definitions with columns for No., 適用 (Scope), エンティティ項目 (Entity Item), 物理名 (Physical Name), 差分 (Difference), データ型 (Data Type), 最大桁数 (Maximum Digits), 小数桁数 (Decimal Digits), 主キー (Primary Key), and 必須 (Must). A red box highlights the 'Edit' button in the top right corner of the table area.

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。

The screenshot shows the 'Data Definition' screen with the 'Edit' dialog open. The dialog has four steps: 1. テーブル定義の設定 (Table Definition Setting), 2. 検索条件の設定 (Search Condition Setting), 3. 業務的排他制御の設定 (Business Exclusive Control Setting), and 4. 更新内容確認 (Update Content Confirmation). Step 1 is currently active. A red box highlights the 'Edit' button in the top right corner of the dialog.

データ定義モーダルが表示されますので、「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。

データ定義



テーブル定義の設定

2

検索条件の設定

3

楽観的排他制御の設定

4

更新内容確認

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 | アンケートID | 完全一致 | |

完全一致
部分一致
前方一致
後方一致
複数条件

[戻る](#)[キャンセル](#)[次へ](#)

「条件を追加」をクリックし、「エンティティ項目」プルダウンから「アンケートID」を、「条件」プルダウンから「完全一致」をそれぞれ選択します。

検索条件を設定したら「次へ」ボタンをクリックし、「楽観的排他制御を利用しない」を選択します。

すべての設定が完了したら「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



楽観的排他制御の設定



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_enquete_question_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 | 操作なし | アンケートID | enquete_id | 文字列 | 200 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 2 | 操作なし | 設問ID | question_id | 文字列 | 200 | - | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| 3 | 操作なし | 設問番号 | question_no | 数値 | 5 | 0 | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| 4 | 操作なし | 設問内容 | question | 文字列 | 2000 | - | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| 5 | 操作なし | 回答項目の左端 | left_scale | 文字列 | 2000 | - | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| 6 | 操作なし | 回答項目の右端 | right_scale | 文字列 | 2000 | - | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

参照項目設定

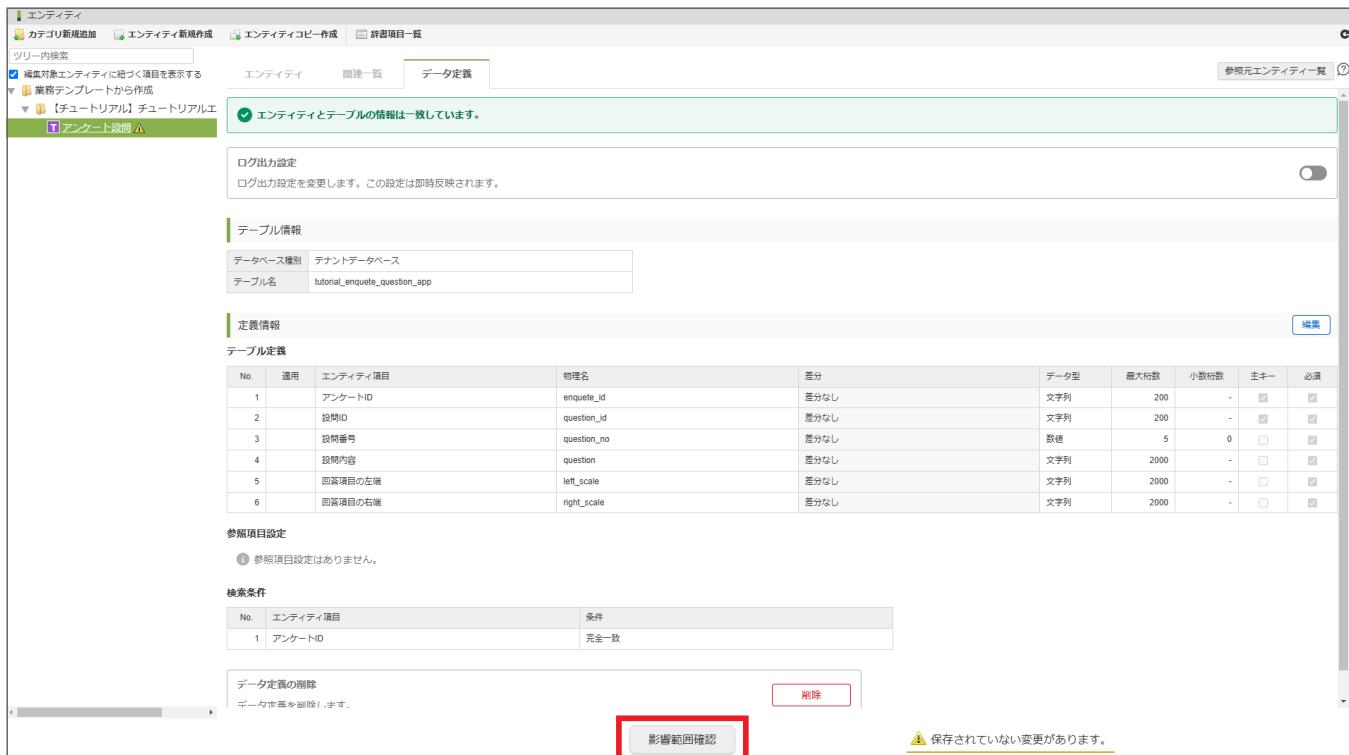
i 参照項目設定は設定されていません。

検索条件

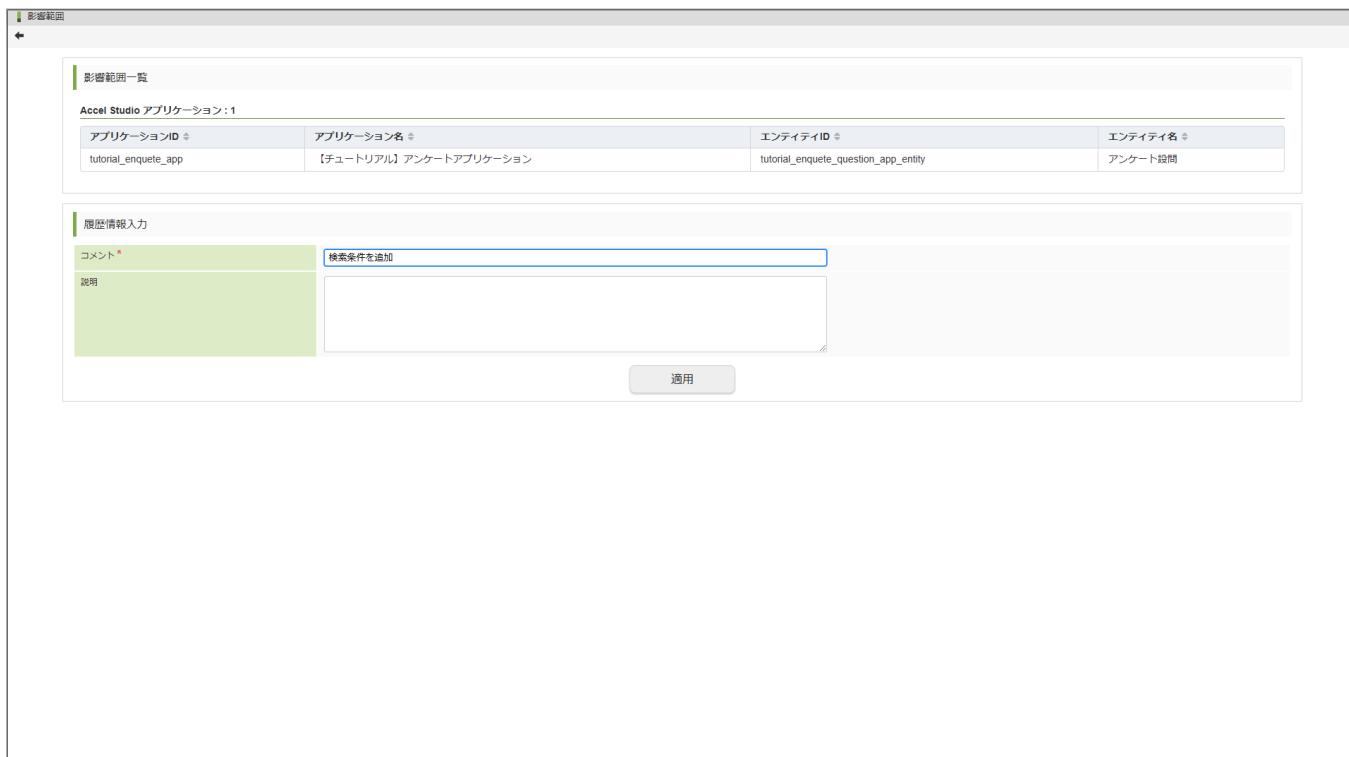
[戻る](#)[キャンセル](#)[反映](#)

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。



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。

同様の手順で、「アンケート回答」のエンティティにも検索条件「アンケートID」 - 「完全一致」を設定してください。
これで、検索条件の設定は完了です。

サンプルデータの読み込み

このチュートリアルでは、動作確認やメニューの設定に、サンプルデータを利用した手順を説明しています。

アンケート情報テーブルとアンケート設問テーブルに登録するサンプルデータを用意していますので、以下からダウンロードしたSQLを「サ

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。

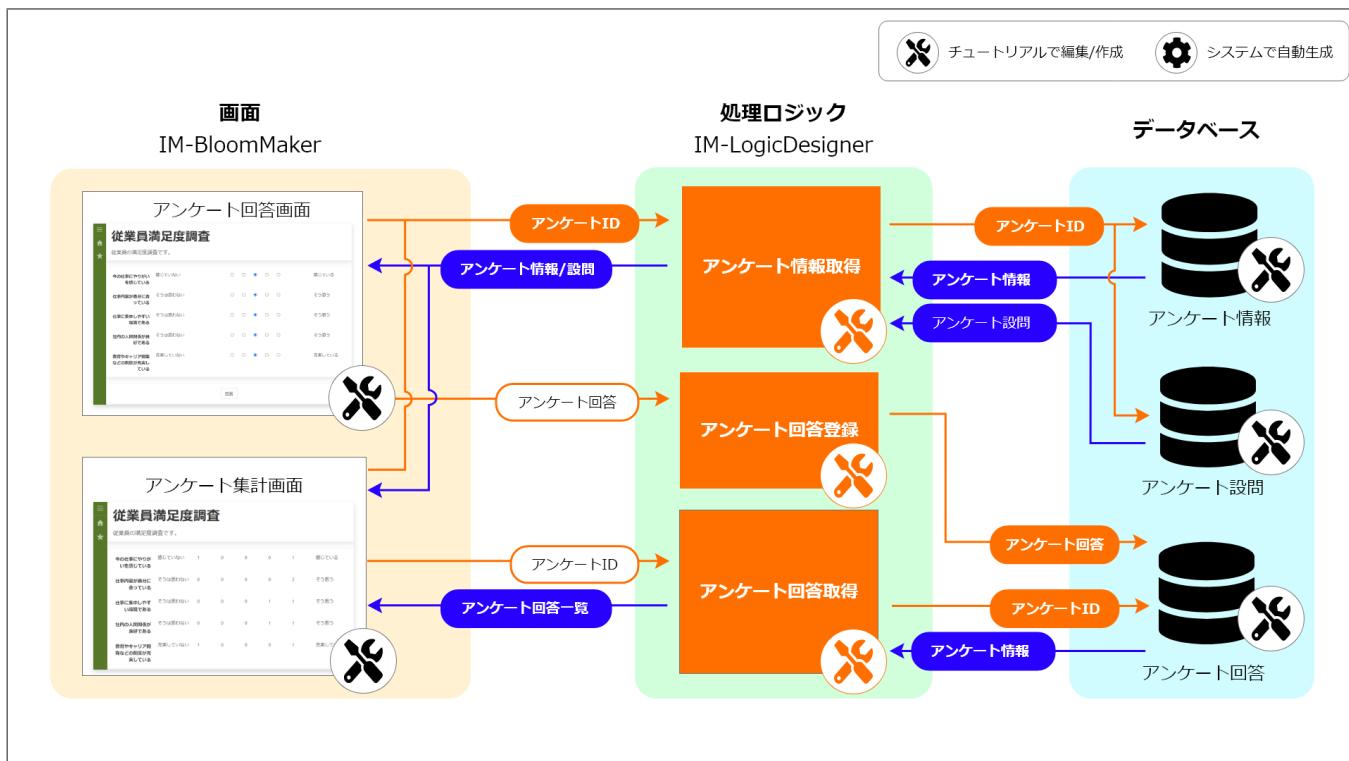
- 従業員満足度調査 (tutorial_es_enquete)
- 好きな映画ジャンル調査 (tutorial_movie_enquete)

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。



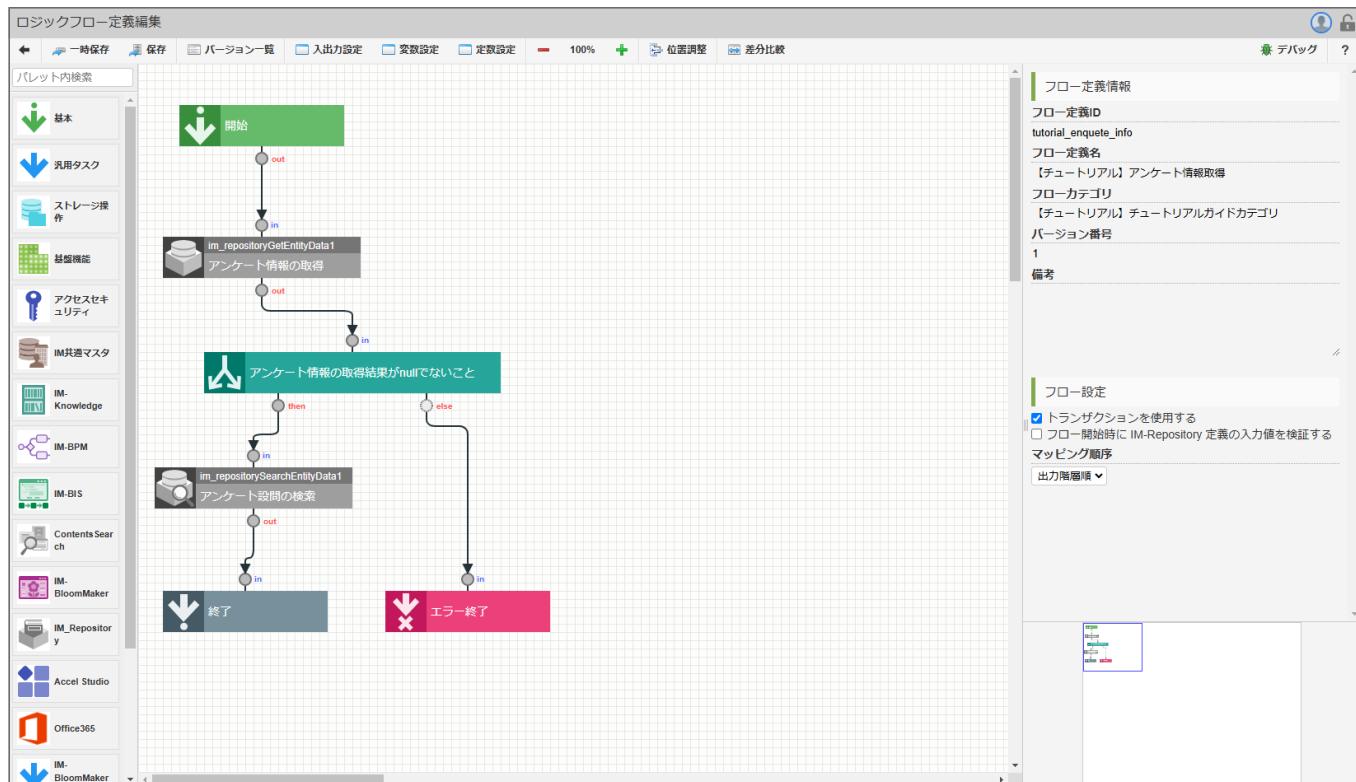
この処理の作成手順について、以下の動画からも確認できます。併せてご利用ください。

アンケート情報取得処理の作成

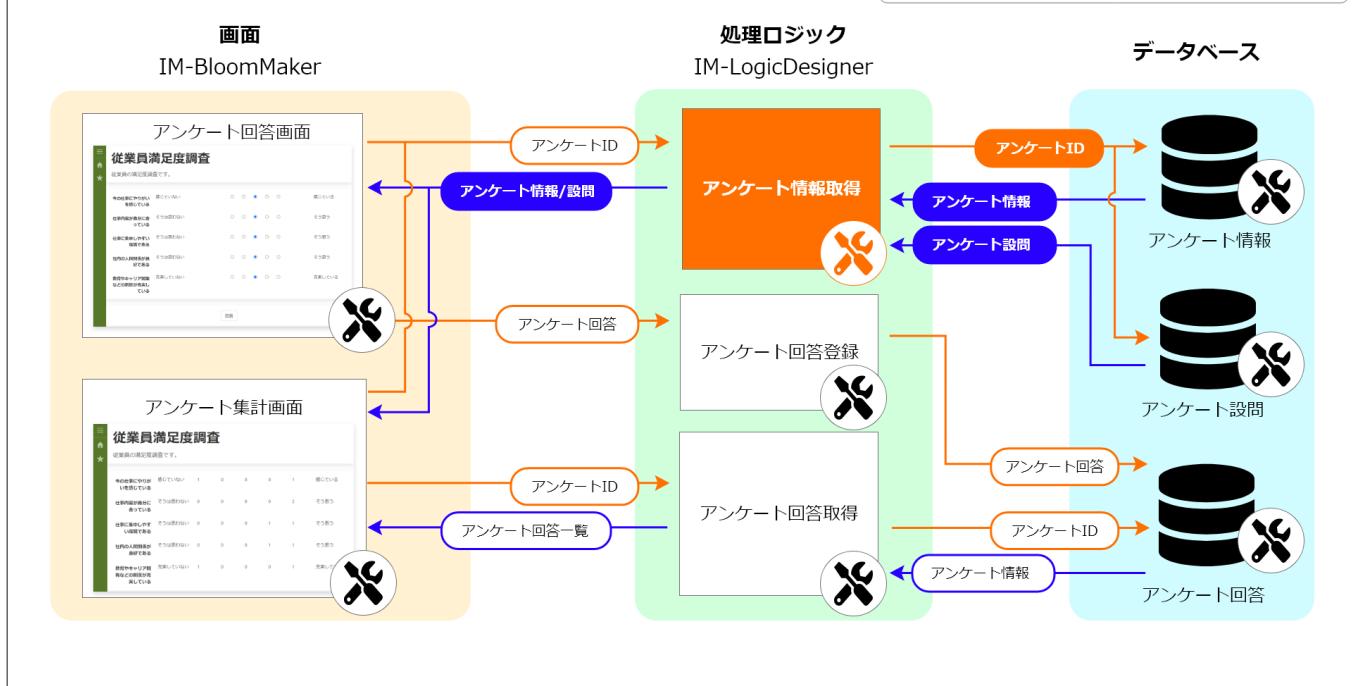
アンケート情報取得処理では、渡ってきたアンケートIDをもとに、アンケート情報テーブルとアンケート設問テーブルからデータを取得します。

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エンティティでデータを取得し、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。



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。

The screenshot shows the 'Application Management' interface. On the left, there's a sidebar with various icons. The main area displays a list of applications under the heading '【チュートリアル】 チュートリアルガイド'. One item, '[チュートリアル] アンケートアプリケーション' (Survey Application), is highlighted with a red box. This item has a detailed description: 'マスタメンテンナンステンプレートを利用して作成する簡単なアンケートアプリケーション' (A simple survey application created using the Master Maintenance Template) and an ID: 'tutorial_enquete_app'. The timestamp '2024/02/22 17:19:24' is also visible.

「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。

The screenshot shows the application management interface of the intra-mart Accel Platform. On the left, there's a sidebar with icons for home, search, and user. The main area has a header with tabs like '画面' (Page), 'ログ' (Log), 'ワークフロー' (Workflow), etc. A search bar is at the top. Below it is a list of entities with columns for ID, Name, Type, and Last Update. A context menu is open over the first entity in the list, with the 'ログ' (Log) option highlighted with a red box. Other options in the menu include 'クエリ' (Query) and 'ロジックトリガ' (Logic Trigger). To the right of the list, there are several panels showing detailed information about specific logs or entities.

入出力値と定数の設定

業務ロジックへの入力値と出力値、定数を設定します。

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。

The screenshot shows the 'ロジックフロー定義編集' (Logic Flow Definition Editor) window. On the left is a palette with various task icons. In the center, there's a flowchart with a green '開始' (Start) node at the top and a grey '終了' (End) node at the bottom, connected by a blue arrow labeled 'in' and 'out'. At the top of the editor, there are tabs: '一時保存' (Temporary Save), '新規保存' (New Save), '入出力設定' (Input/Output Setting) (which is highlighted with a red box), '変数設定' (Variable Setting), '定数設定' (Constant Setting), and '位置調整' (Position Adjustment). On the right, there are two main sections: 'フロー定義情報' (Flow Definition Information) and 'フロー設定' (Flow Settings). The 'フロー定義情報' section contains fields for 'フロー定義ID' (Flow Definition ID), 'フロー定義名' (Flow Definition Name), 'フローカテゴリ' (Flow Category), and '備考' (Remarks). The 'フロー設定' section contains a checkbox for 'トランザクションを使用する' (Use Transaction) and another for 'フロー開始時に IM-Repository 定義の入力値を検証する' (Validate input values from IM-Repository definition when flow starts).

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「キーネームを変更」をクリックし、キーネームを「enquetedId」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。

```
{
  "enquetedId": ""
}
```

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な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」 - 「+IM-Repository」をクリックして「エンティティ検索」ウインドウを開きます。



「アンケート設問」を選択して「決定」をクリックしてください。



出力値一覧に「tutorial_enquete_question_app_entity」が追加されます。



「tutorial_enquete_question_app_entity」を選択した状態で出力値一覧上部の「キーナを変更」をクリックし、キーナを「records」に変更します。

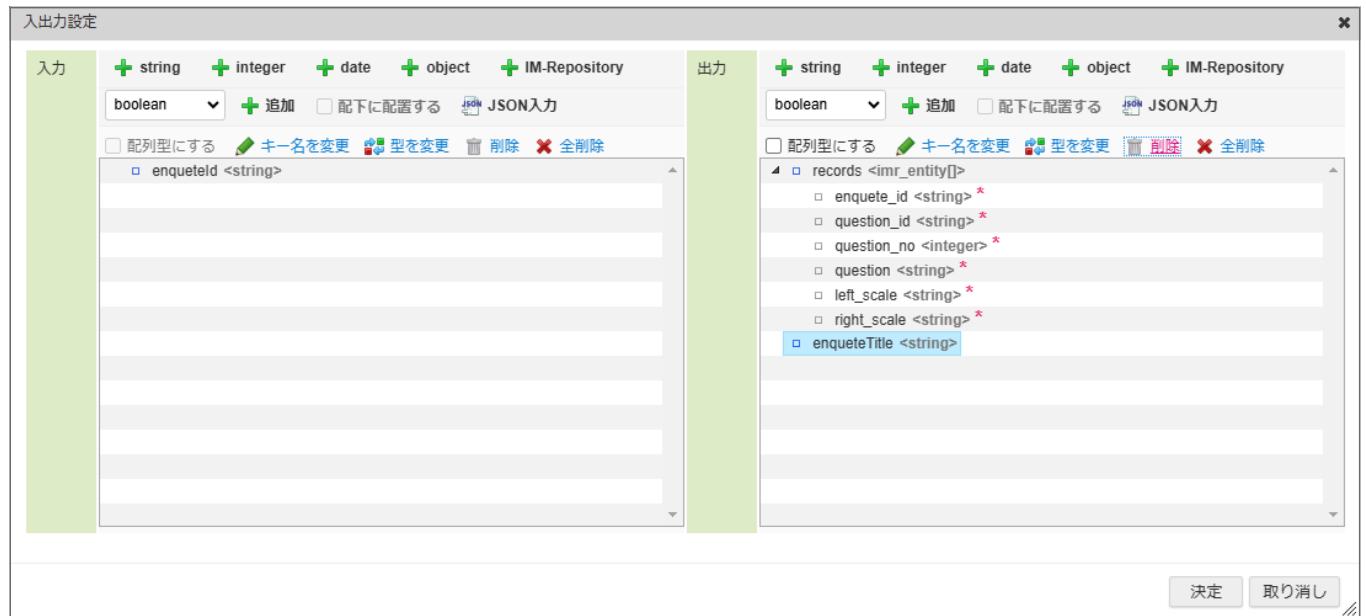
「records」を選択した状態で「配列型にする」にチェックをいれ、配列型に変更します。



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。

「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「キーナーを変更」をクリックし、キーナーを「enqueueTitle」に変更します。



「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「キーナーを変更」をクリックし、キーナーを「enqueueOverview」に変更します。



出力値はJSON入力で下記コードを記述しても設定できます。

```
{
  "records": [
    {
      "enquete_id": "",
      "question_id": "",
      "question_no": 0,
      "question": "",
      "left_scale": "",
      "right_scale": ""
    }
  ],
  "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ないか注意してください。

全ての出入力設定が完了したら、出入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。

次に、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。

| 定数ID | 定数值 |
|-------------|-------------|
| ASC | asc |
| QUESTION_NO | question_no |

定数設定

+ 定数を追加 + 共通定数を追加 - 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 | データ型 | 共通定数 | 説明 | エディタ |
|--------------------------|-------------|-------------|--------|------|----|------|
| <input type="checkbox"/> | ASC | asc | string | | | |
| <input type="checkbox"/> | QUESTION_NO | question_no | string | | | |

決定 取消し

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。

IM-Repositoryタスクの配置と変数設定

アンケートの情報を取得するタスクを設定します。

パレットから「IM-Repository」 - 「エンティティデータの取得」を選択し、「開始」の次のタスクとして配置します。

ロジックフロー定義編集 *

一時保存 新規保存 入出力設定 変数設定 定数設定 100% 位置調整

パレット内検索

■ 基本 洗濯タスク ストレージ操作 設備機能 アクセスセキュリティ IM共通マスター IM-Knowledge IM-BPM IM-BIS Contents Search IM-BloomMaker IM-Repository Accel Studio Office365 IM-BloomMaker Sample ユーザ定義追加

タスク種別 IM_Repository エンティティデータの取得

基本設定

タスクID * im_repositoryGetEntityData1
エイリアス * im_repositoryGetEntityData1
ラベル * エンティティデータの取得
マッピング設定
マッピング設定
エンティティを識別してください。
コメント

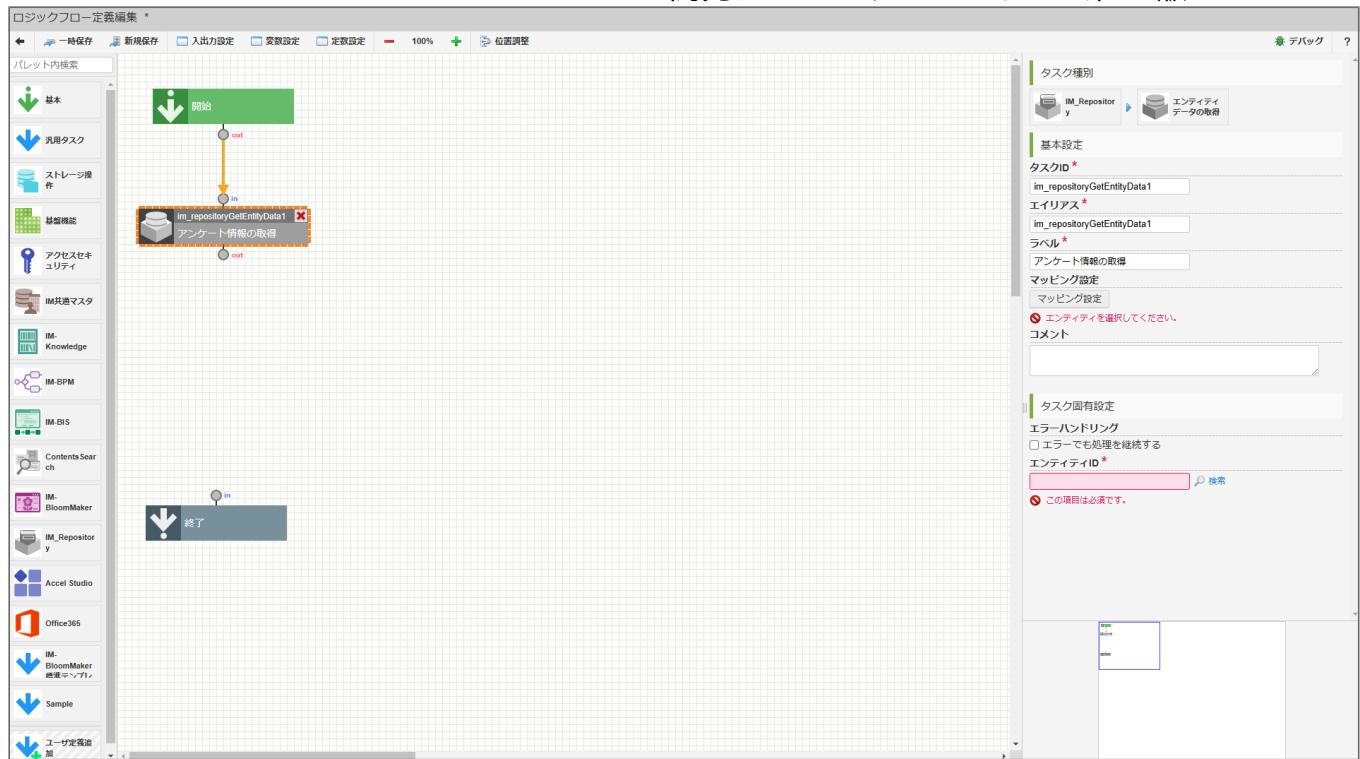
タスク固有設定

エラー処理
エラーでも処理を継続する
エンティティID *

この項目は必須です。

配置した「エンティティデータの取得」タスクの「ラベル」を「アンケート情報の取得」に変更します。

「開始」の「out」からドラッグし、「アンケート情報の取得」の「in」にドロップしてタスク同士を線で繋げます。

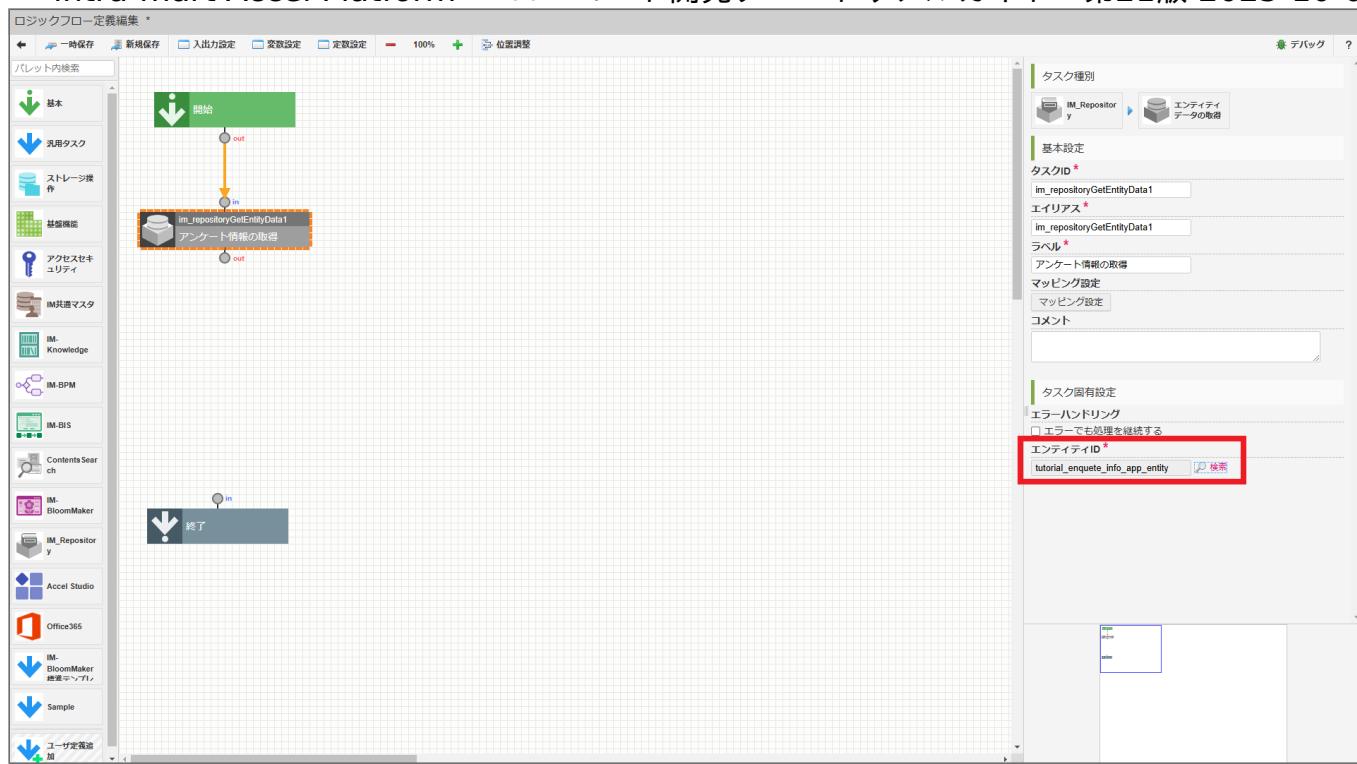


「タスク固有設定」から「エンティティID」を設定します。「検索」アイコンをクリックしてエンティティ検索ウインドウを開いてください。

「アンケート情報」を選択し、「決定」をクリックします。

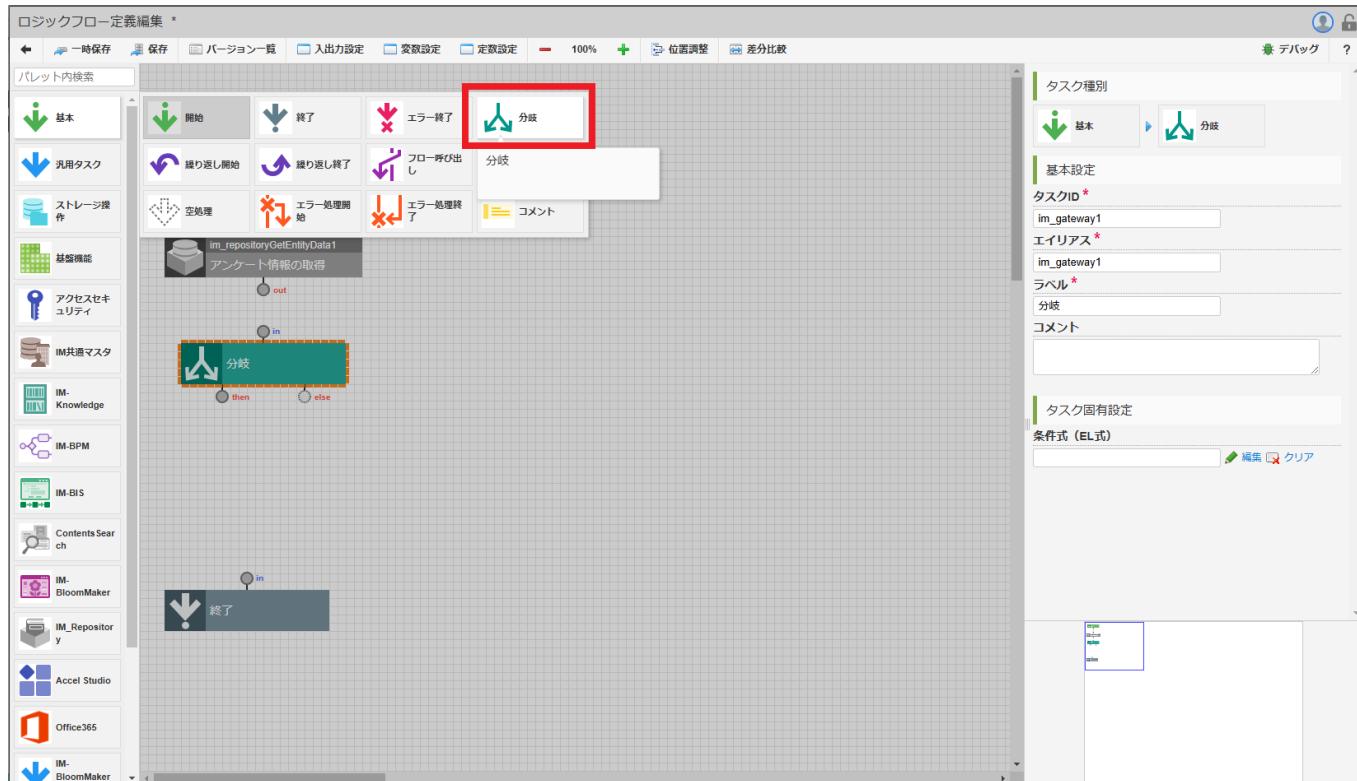


「エンティティID」に「tutorial_enquete_info_app_entity」が入力されました。



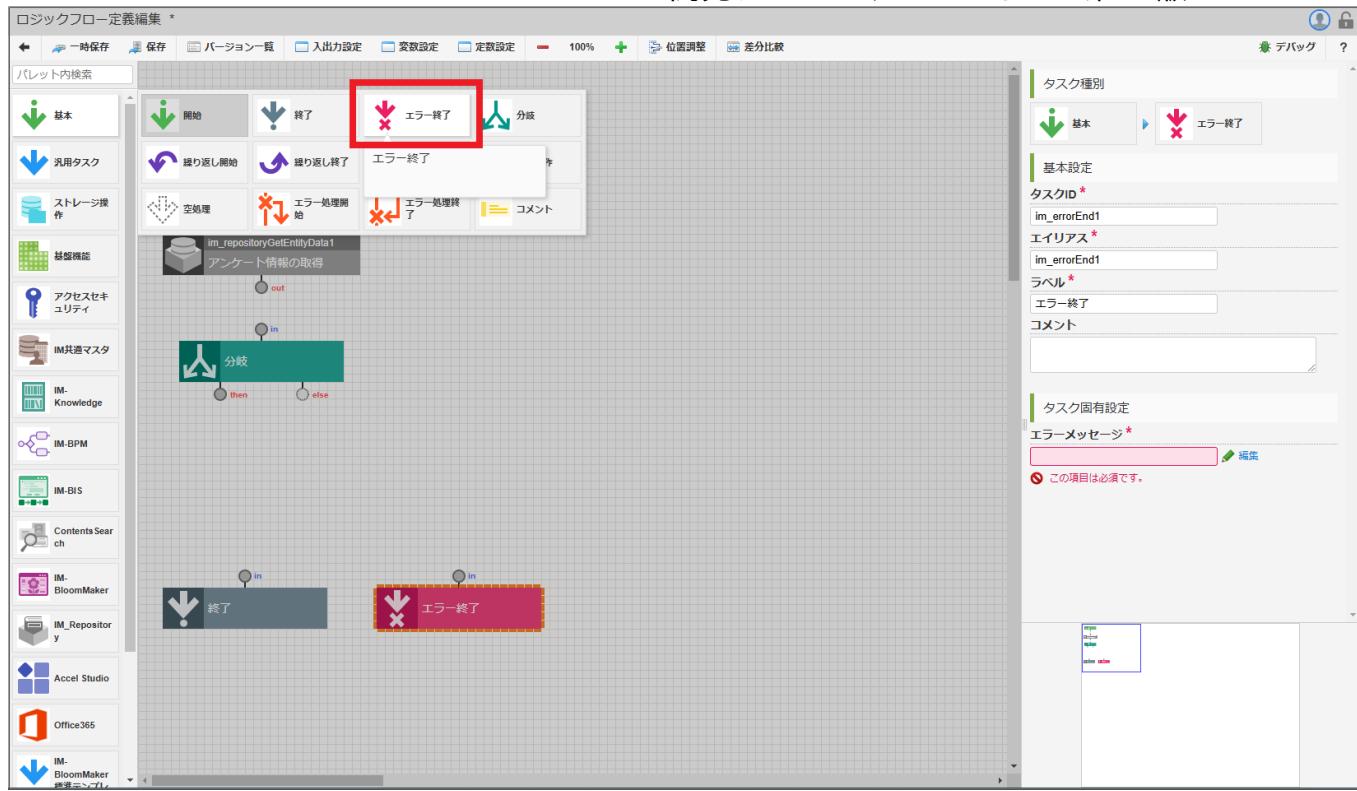
アンケートIDの誤り等でアンケート情報が取得できなかった場合、エラーを返却します。

パレットから「基本」 - 「分岐」を選択し、「アンケート情報の取得」タスクの次の要素として配置します。

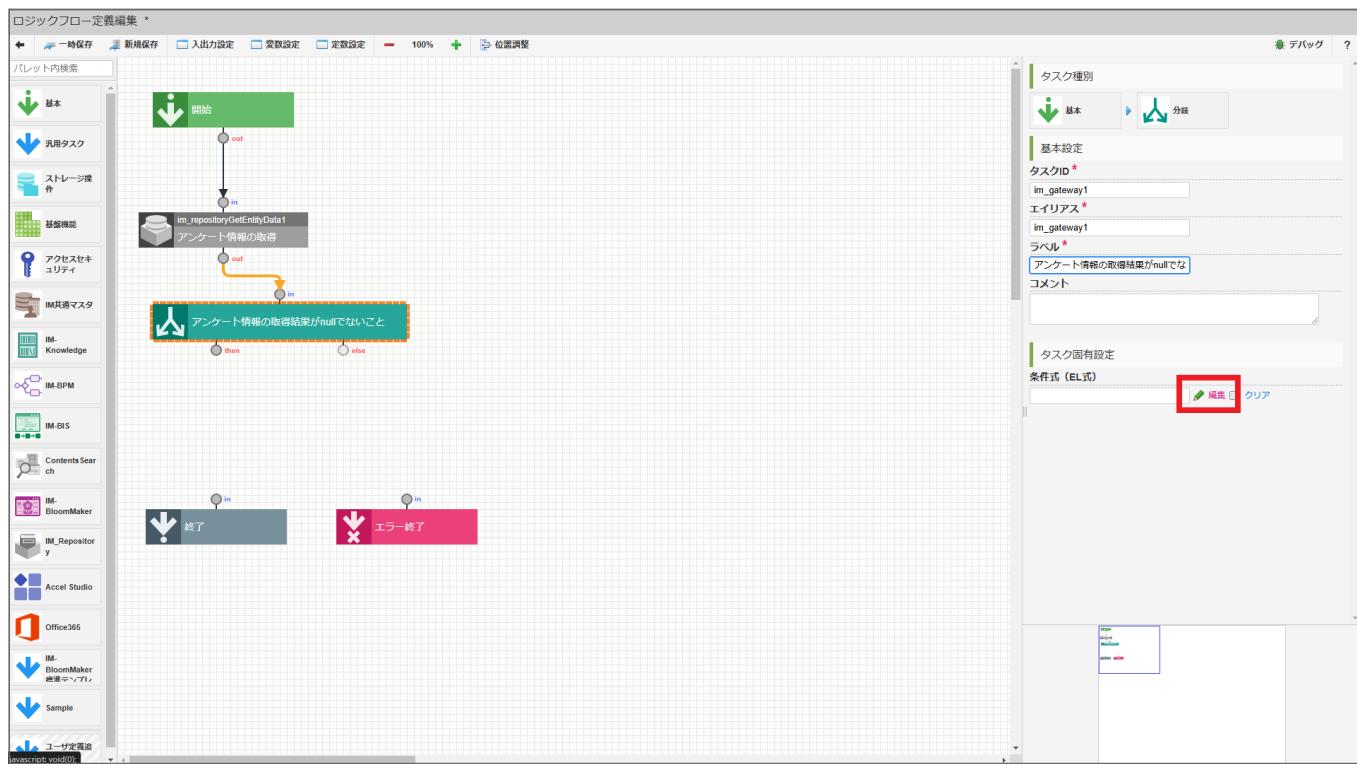


「アンケート情報の取得」タスクの「out」と「分岐」の「in」を線で繋げます。

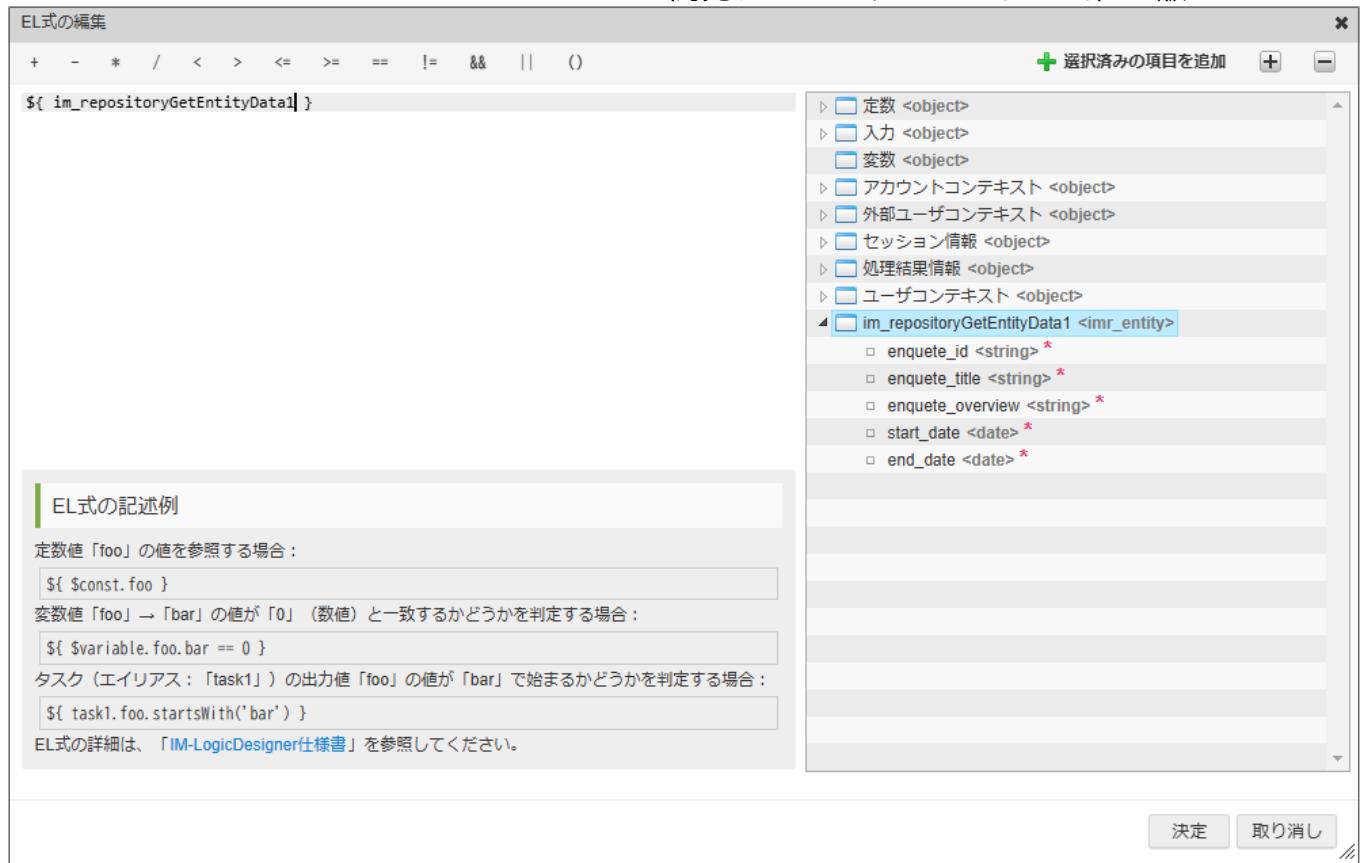
パレットから「基本」 - 「エラー終了」を選択し、「分岐」の次の要素として配置します。



配置した「分岐」をクリックして選択し、画面の右側領域の「ラベル」を「アンケート情報の取得結果がnullでないこと」に変更します。次に、「タスク固有設定」 - 「条件式 (EL式)」の「編集」をクリックします。



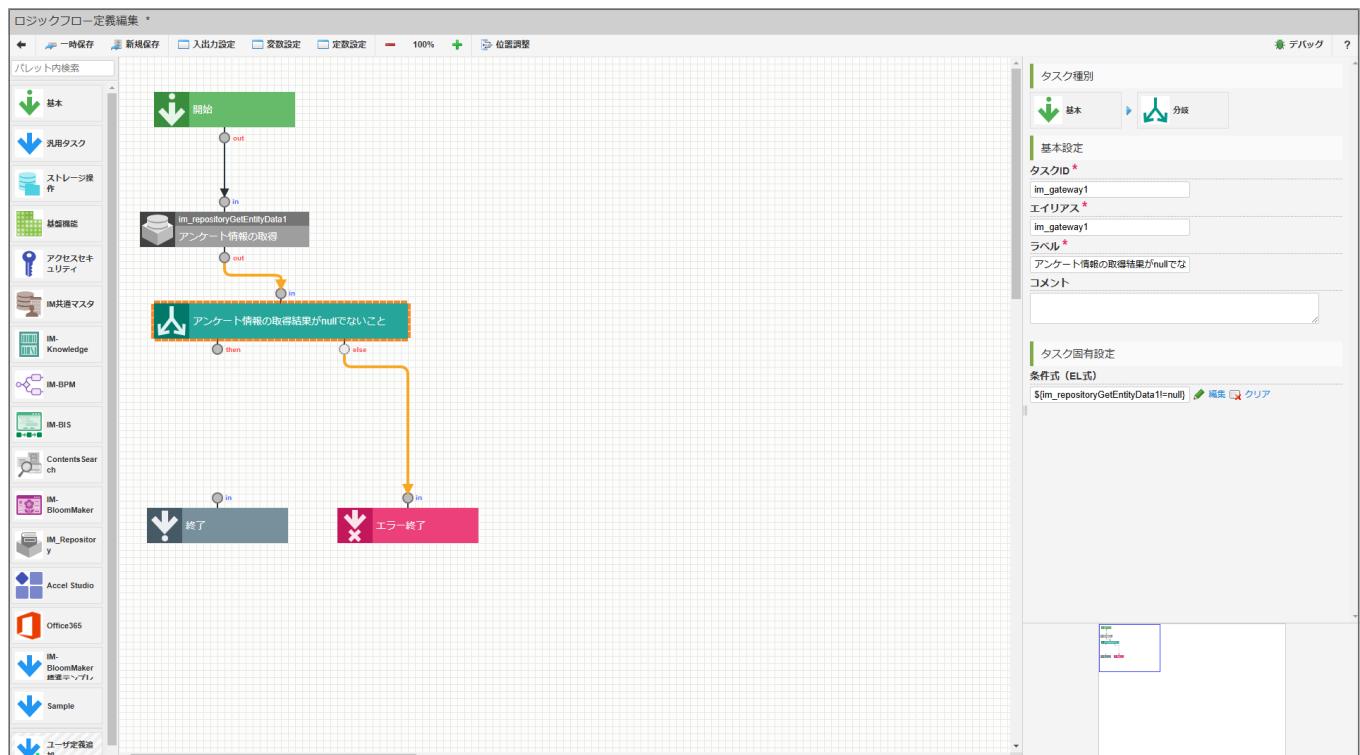
「EL式の編集」ダイアログの右側から「im_repositoryGetEntityData1」を選択します。



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。

「\${ im_repositoryGetEntityData1!= null }」

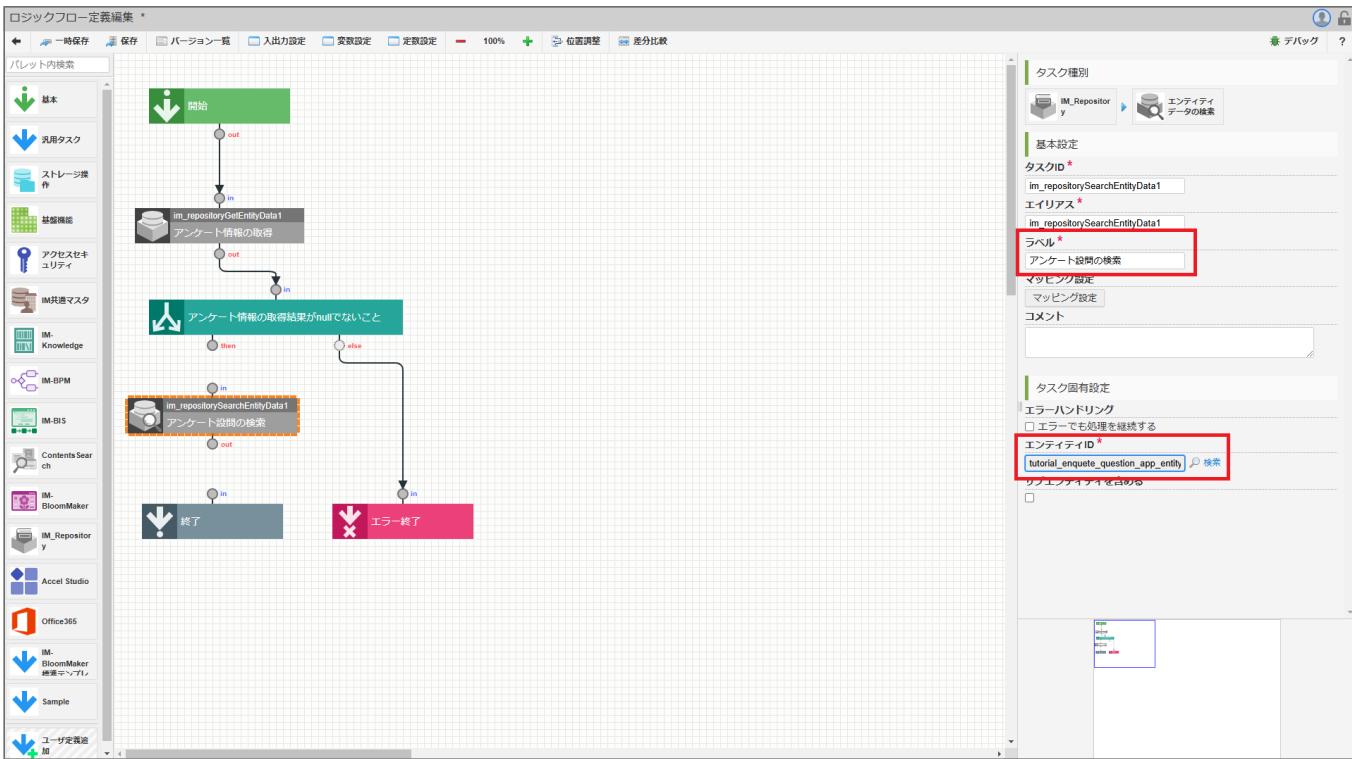
「分岐」の「else」と「エラー終了」の「in」を線で繋げます。



「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 enqueue」と入力します。

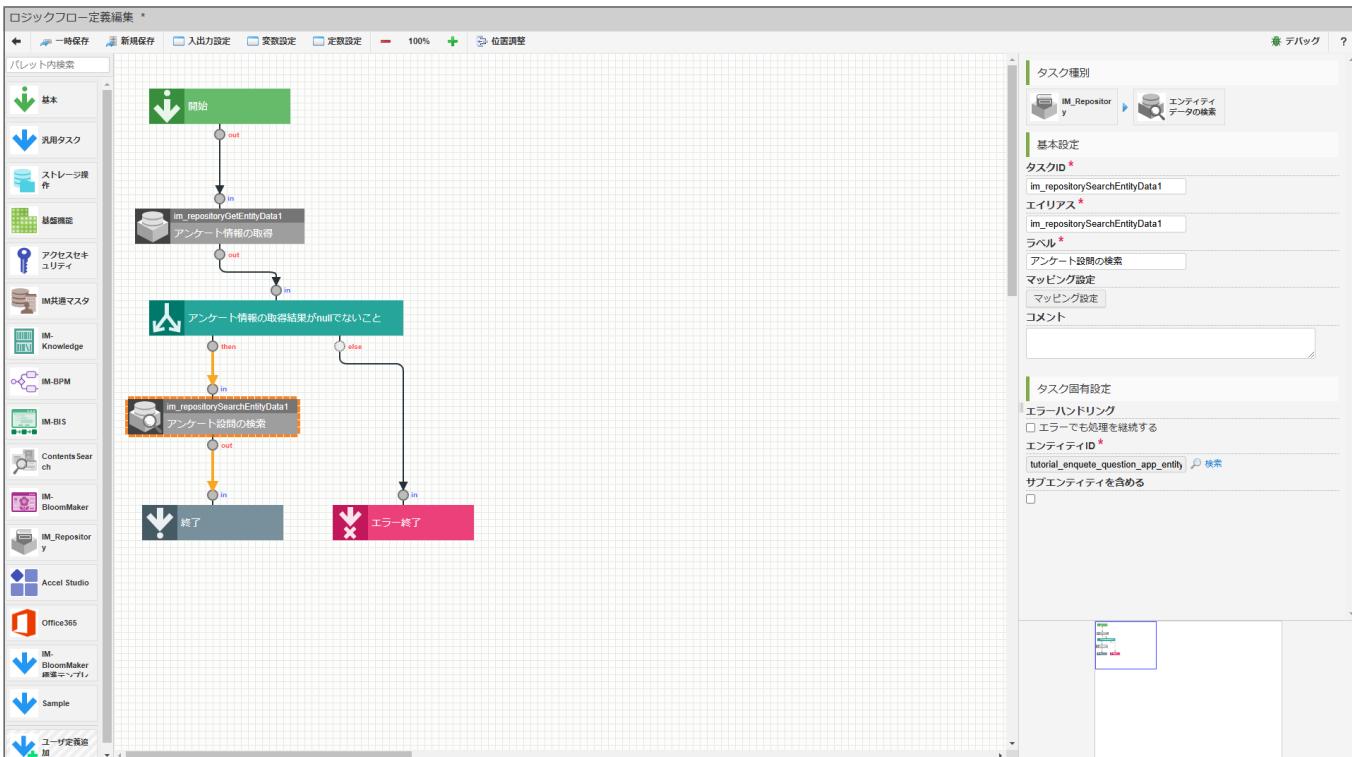
パレットから「IM-Repository」 - 「エンティティデータの検索」を選択し、「分岐」の次のタスクとして配置します。配置した「エンティティデータの検索」の「ラベル」を「アンケート設問の検索」に変更します。

「タスク固有設定」 - 「エンティティID」の「検索」アイコンをクリックしてエンティティ検索ウインドウを開いてください。「アンケート設問」を選択し、「tutorial_enquete_question_app_entity」を設定します。



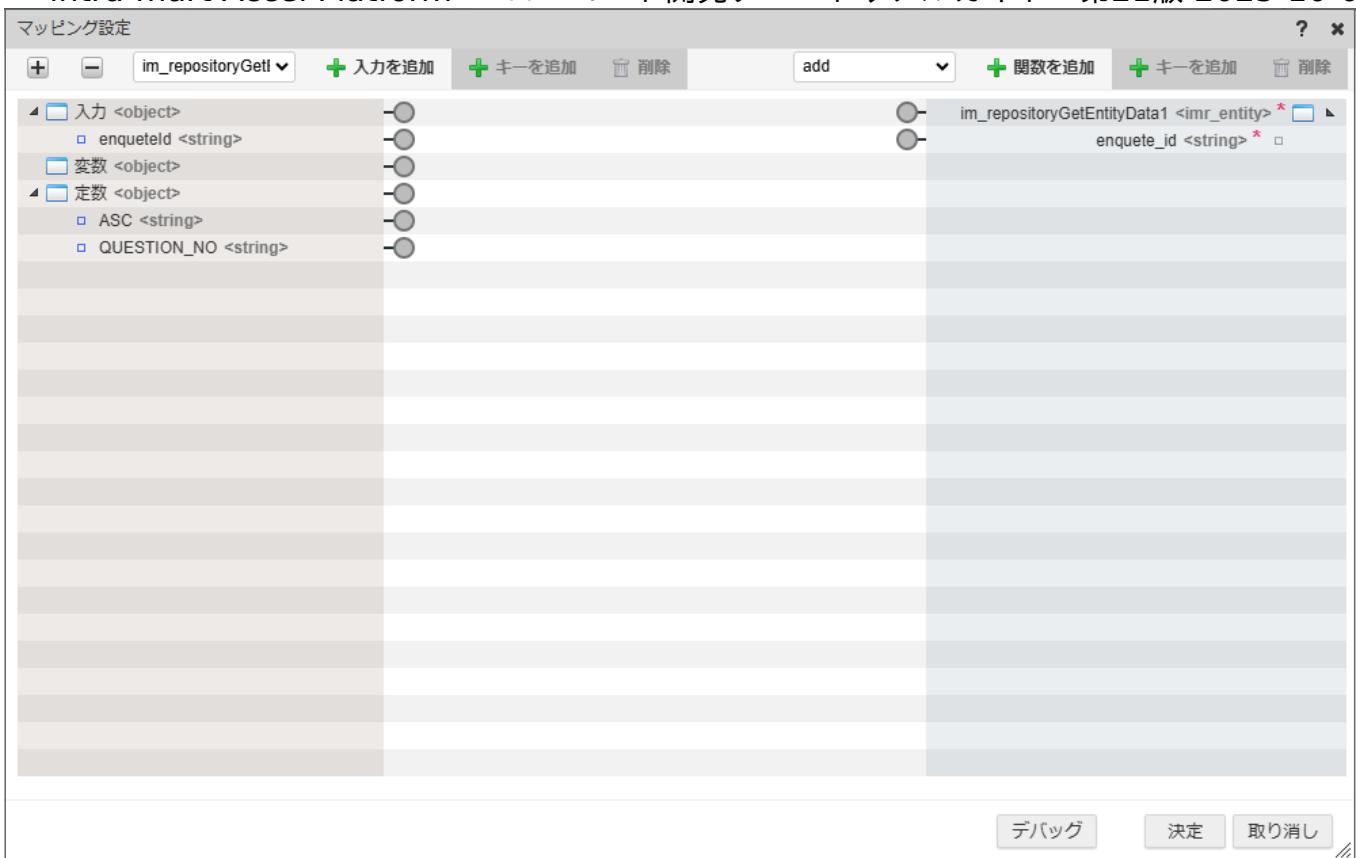
「分岐」の「then」と「アンケート設問の検索」の「in」を線で繋げます。

「アンケート設問の検索」の「out」と「終了」の「in」を線で繋げます。



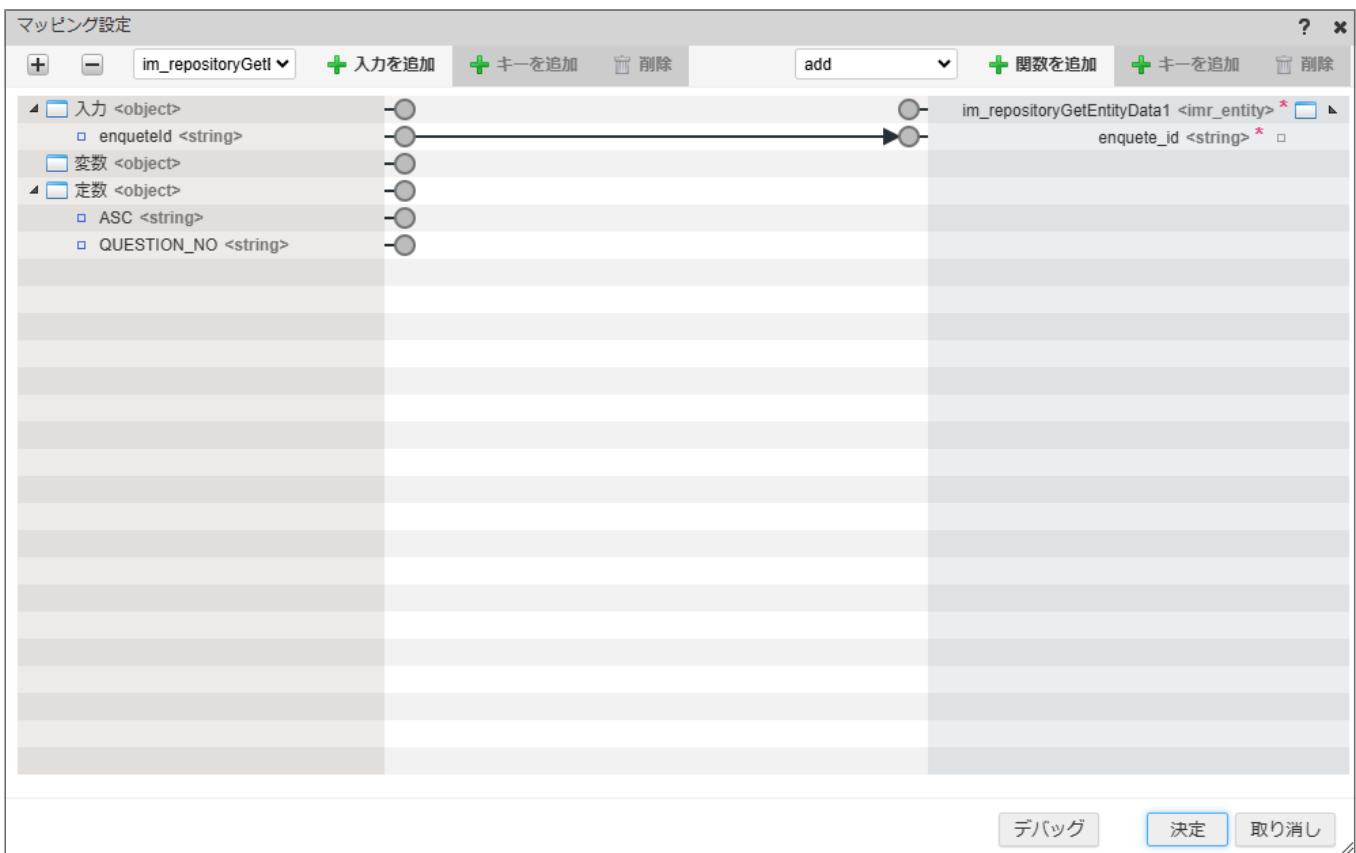
次に、各タスクの変数設定を行います。

「アンケート情報の取得」タスクをダブルクリックしてマッピング設定ダイアログを表示します。



ダイアログ左側の入力の「enqueteid」の端子からドラッグし、右側の「im_repositoryGetEntityData1」の「enquete_id」の端子でドロップして線で繋げます。

「決定」をクリックしてマッピング設定を終了します。

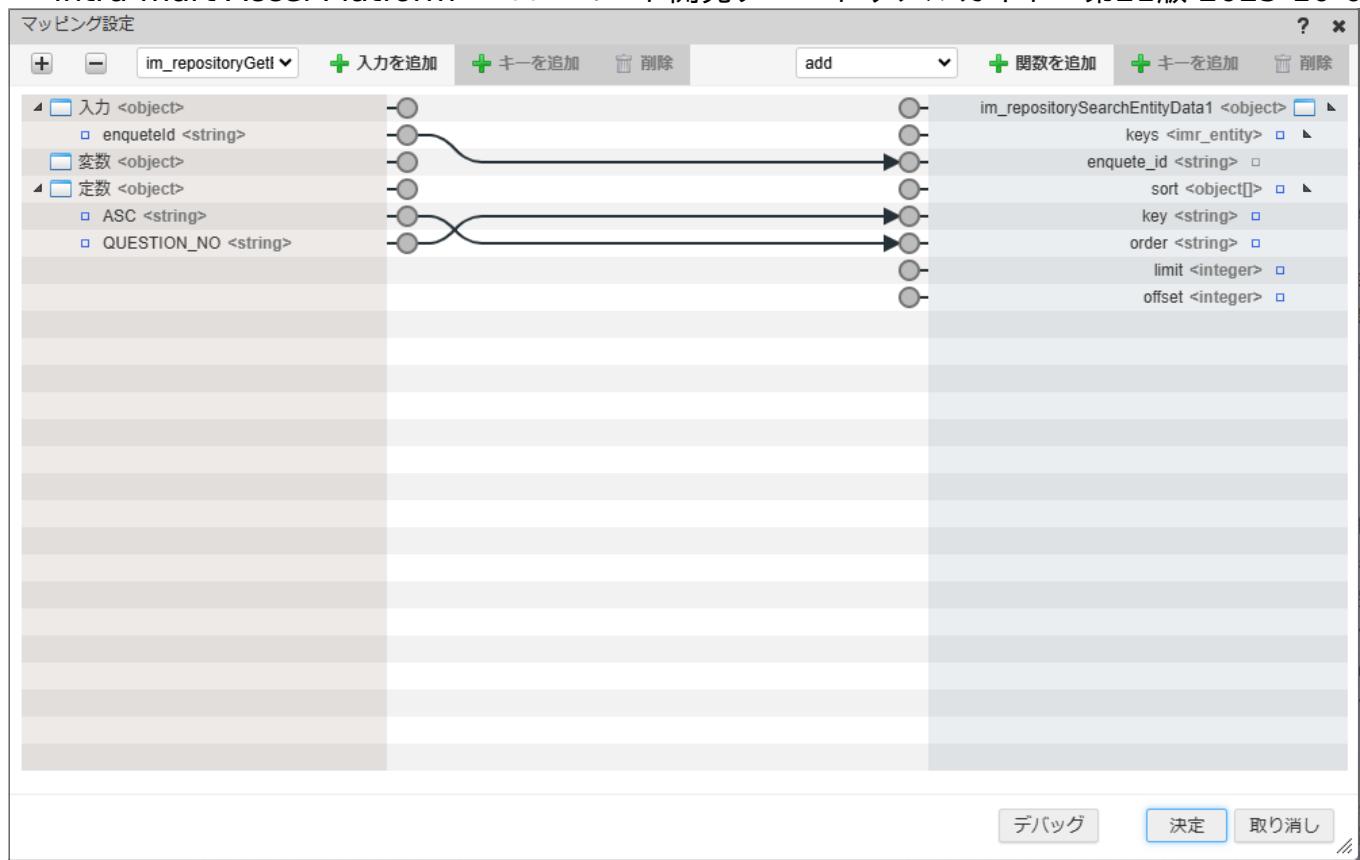


次に、「アンケート設問の検索」タスクのマッピング設定を行います。

ダイアログ左側の入力の「enqueteid」と右側の「im_repositorySearchEntityData1」の「enquete_id」を線で繋げます。

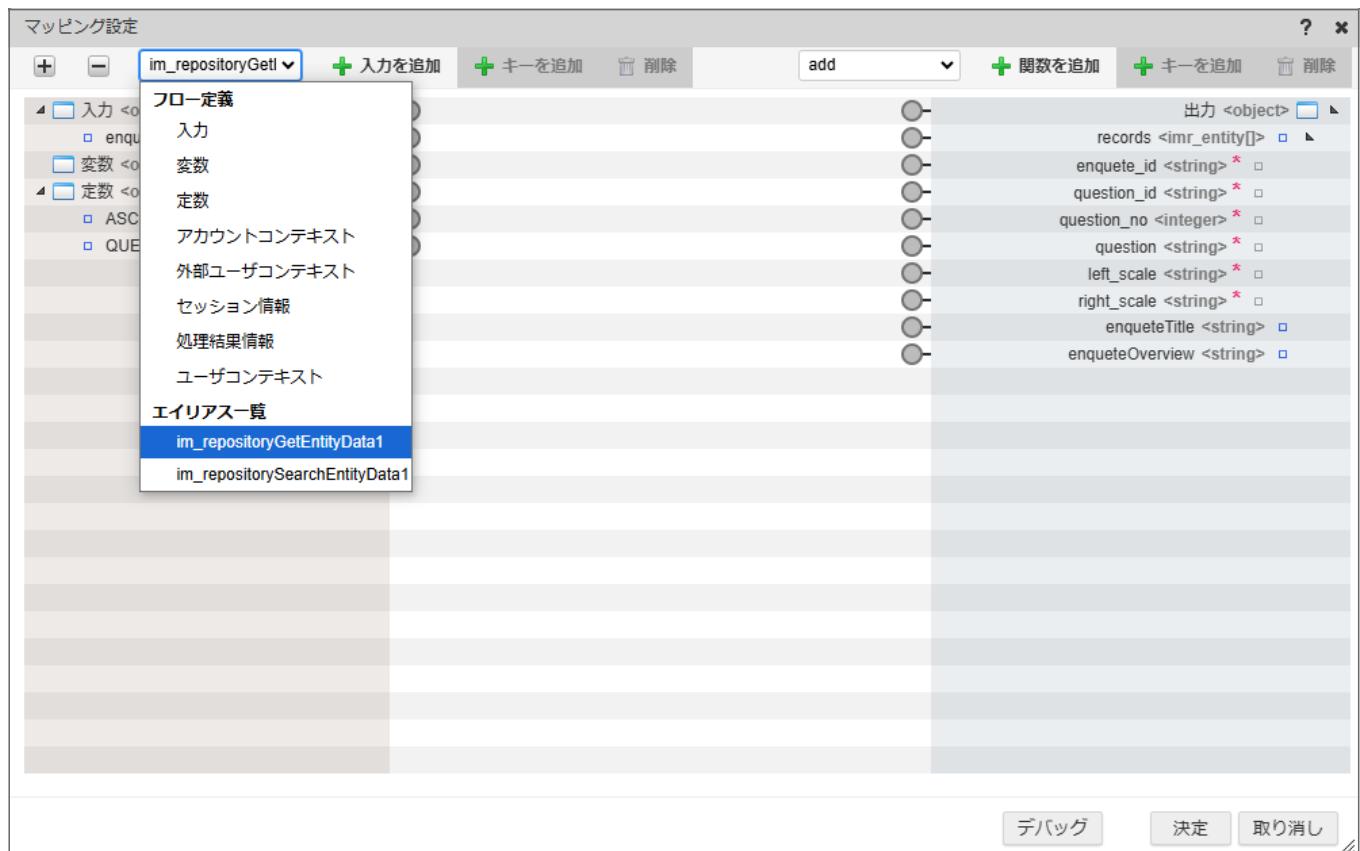
同様に、定数の「ASC」と出力の「order」、「QUESTION_NO」と「key」をそれぞれ線で繋げてください。

ここまで設定ができたら「決定」をクリックし、マッピング設定を終了します。



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」 - 「im_repositoryGetEntityData1」を選択し、「+入力を追加」をクリックします。



追加した「im_repositoryGetEntityData1」の「enqueue_title」と出力の「enqueueTitle」を線で繋げます。

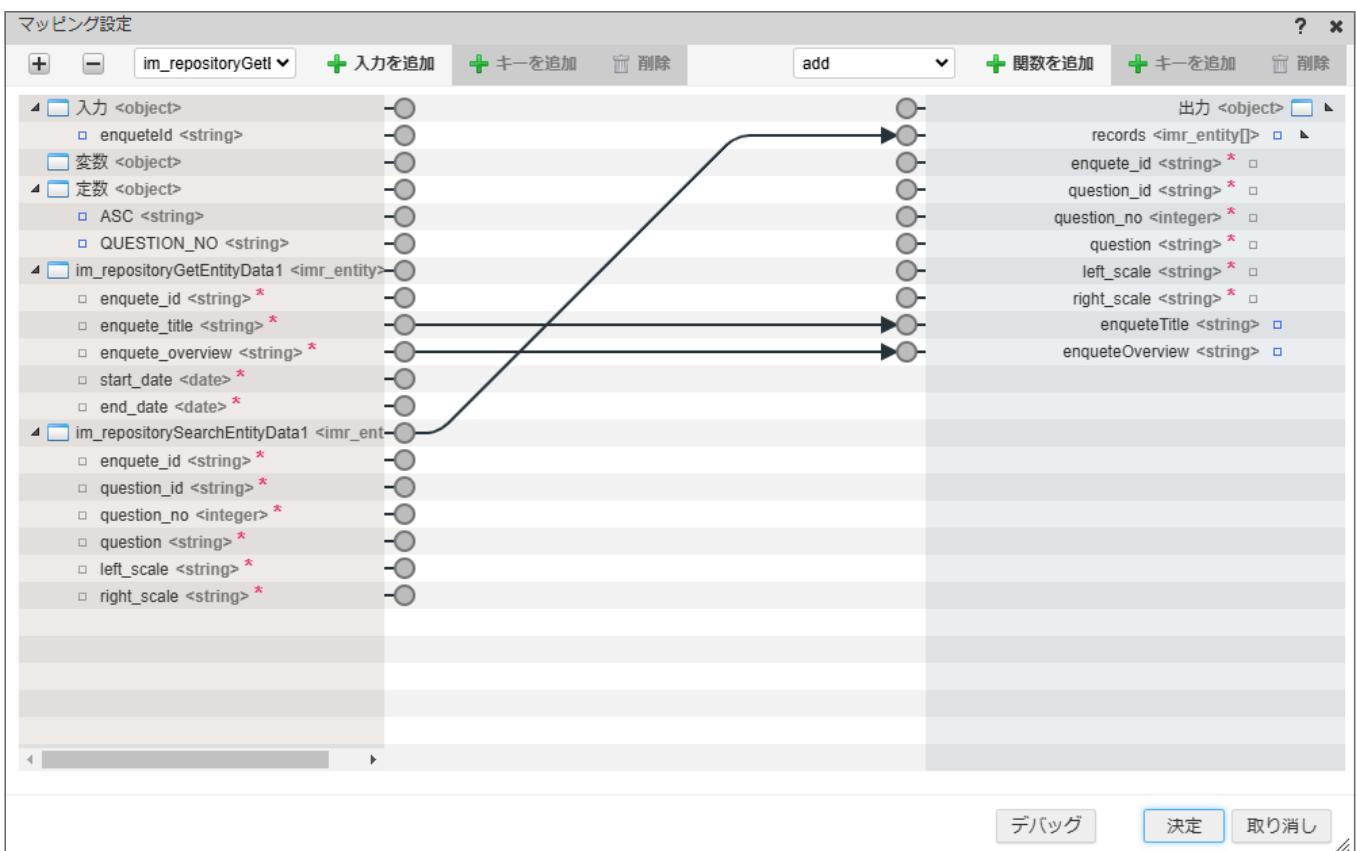
同様に「im_repositoryGetEntityData1」の「enqueue_overview」と出力の「enqueueOverview」を線で繋げます。



次に、入力側プルダウンから「エイリアス一覧」 - 「im_repositorySearchEntityData1」を選択し、「+入力を追加」をクリックします。

追加した「im_repositorySearchEntityData1」と出力の「records」を線で繋げます。

ここまで設定ができたら「決定」をクリックし、マッピング設定を終了します。

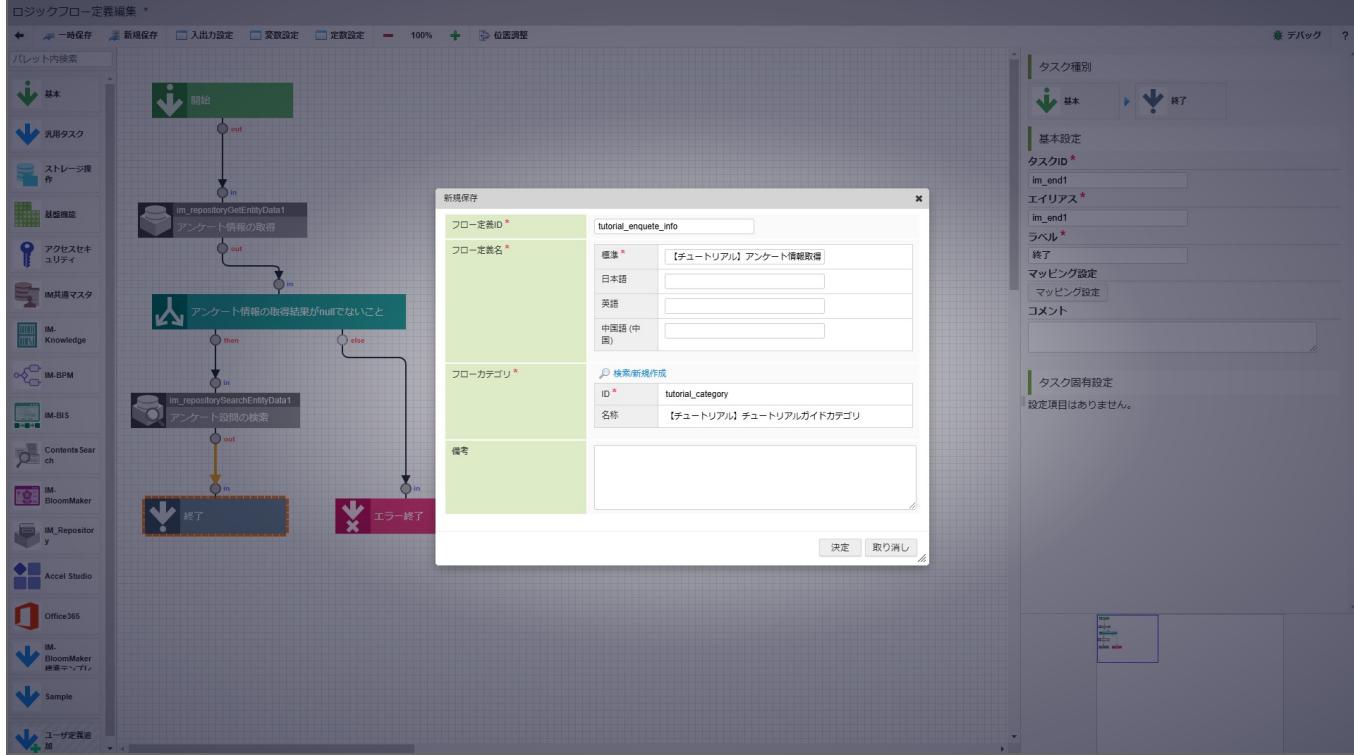


これでアンケート情報取得処理は作成完了です。

「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。

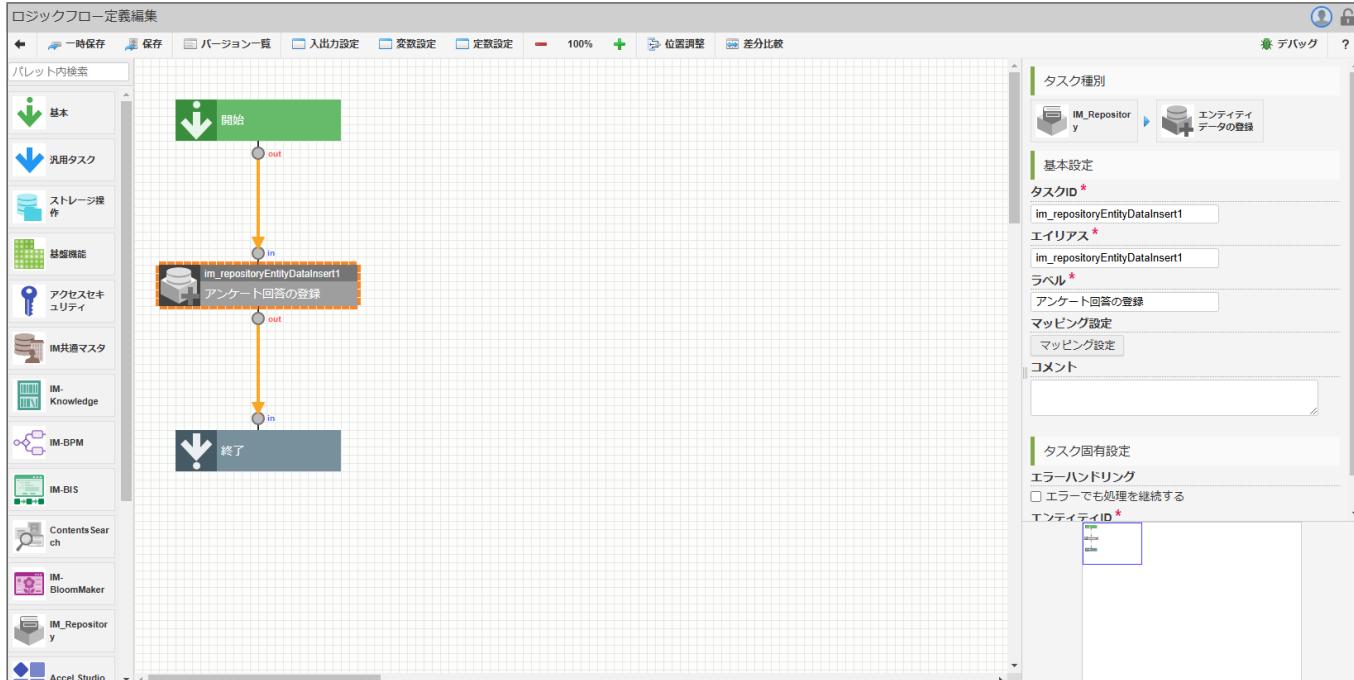
| ラベル | 入力値 |
|------------|-------------------------|
| フロー定義ID | tutorial_enquete_info |
| フロー定義名（標準） | 【チュートリアル】アンケート情報取得 |
| フローカテゴリ | 【チュートリアル】チュートリアルガイドカテゴリ |



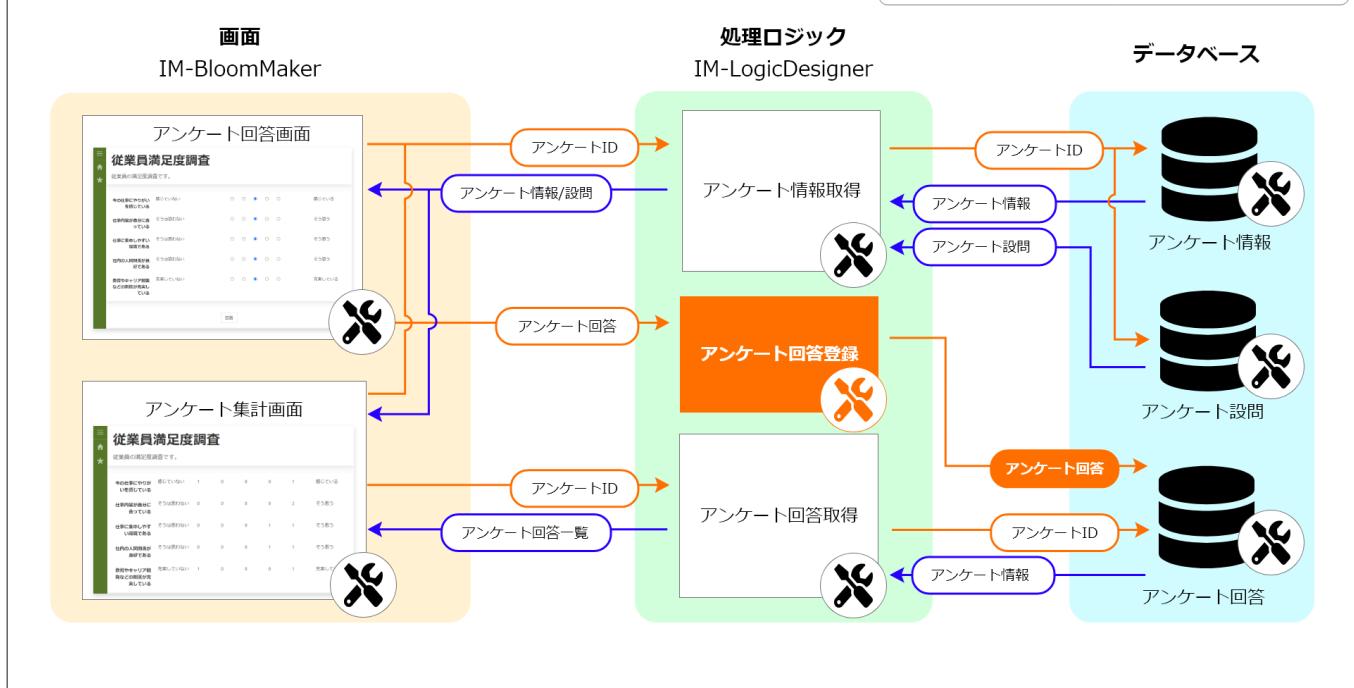
アンケート回答登録処理の作成

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。



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。

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。

The screenshot shows the application management interface of the intra-mart Accel Platform. On the left, there's a sidebar with various icons. The main area displays a search results list for 'アンケート回答' (Survey Answer) entries. A context menu is open over one of the entries, with the 'ログフロー' (Log Flow) option highlighted by a red box. To the right, there are three detailed panels showing log flow definitions for different entities: 'アンケート回答' (Survey Answer), 'アンケート設問' (Survey Question), and 'アンケート情報' (Survey Information). Each panel includes fields for entity name, ID, table name, search conditions, execution user, and execution date.

入出力値と定数の設定

業務ロジックへの入力値と定数を設定します。

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。

The screenshot shows the 'Log Flow Definition Editor' window. On the left is a palette with various task icons. In the center, there's a canvas with a green '開始' (Start) node at the top and a grey '終了' (End) node at the bottom, connected by a flow. A red box highlights the '入出力設定' (Input/Output Settings) tab in the top navigation bar. To the right, there are two tabs: 'フロー定義情報' (Flow Definition Information) and 'フロー設定' (Flow Settings). The 'フロー設定' tab is active, showing checkboxes for 'トランザクションを使用する' (Use Transaction) and 'フロー開始時に IM-Repository 定義の入力値を検証する' (Validate input values from IM-Repository definition when flow starts).

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。



「string1」を選択した状態で入力値一覧上部の「キーナを変更」をクリックし、キーナを「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「キーナを変更」をクリックし、キーナを「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。



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。

ここまで設定が完了したら、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

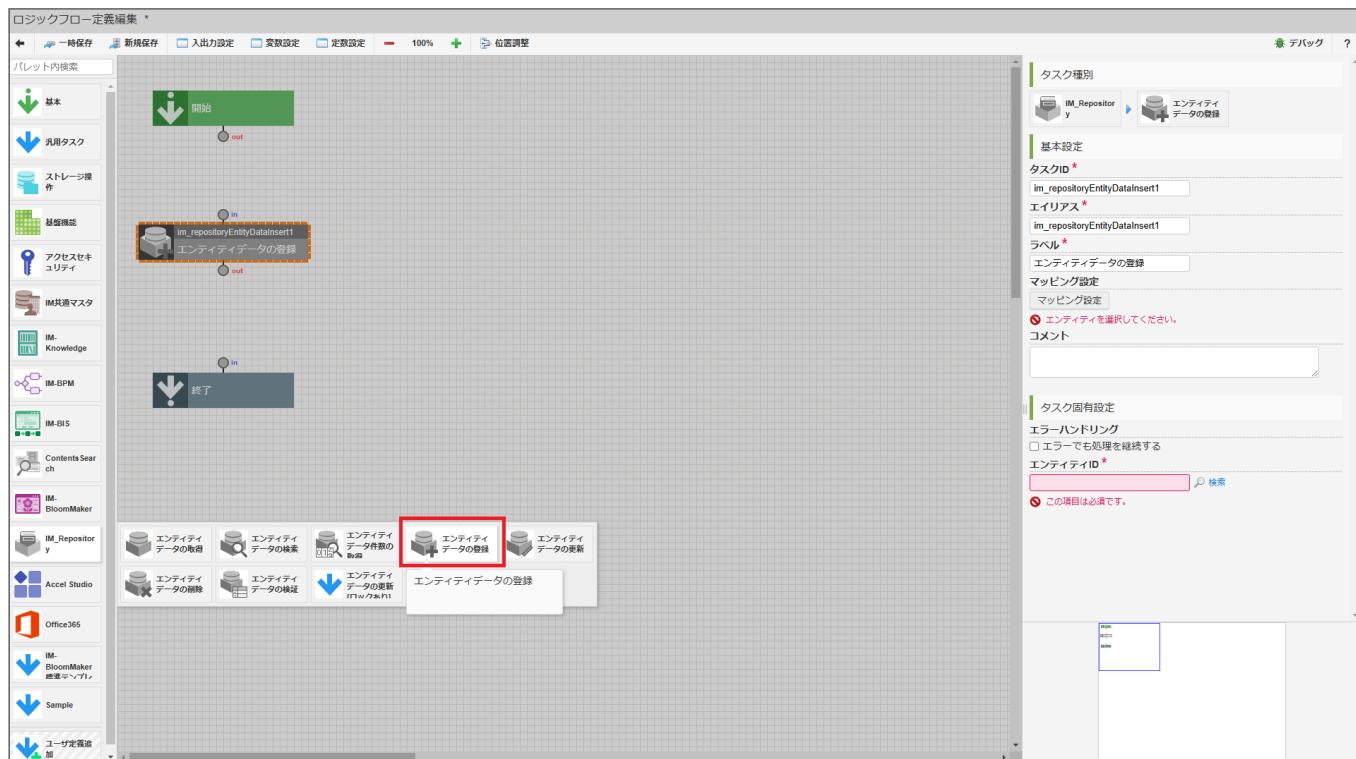
定数値 ,

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。

IM-Repositoryタスクの配置と変数設定

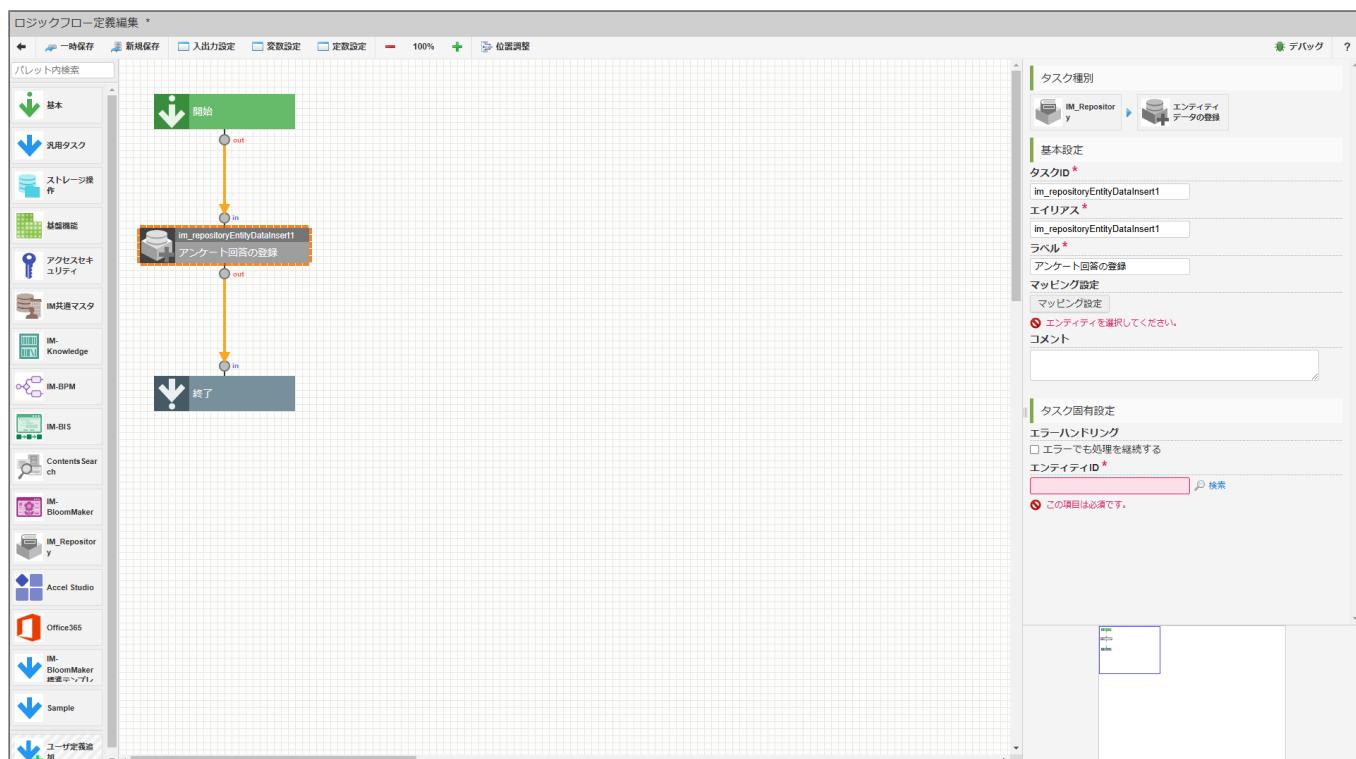
アンケートの回答を登録するタスクを設定します。

パレットから「IM-Repository」 - 「エンティティデータの登録」を選択し、「開始」の次のタスクとして配置します。



配置した「エンティティデータの登録」タスクの「ラベル」を「アンケート回答の登録」に変更します。

「アンケート回答の登録」と「開始」、「終了」のin, outをそれぞれ線で繋げます。

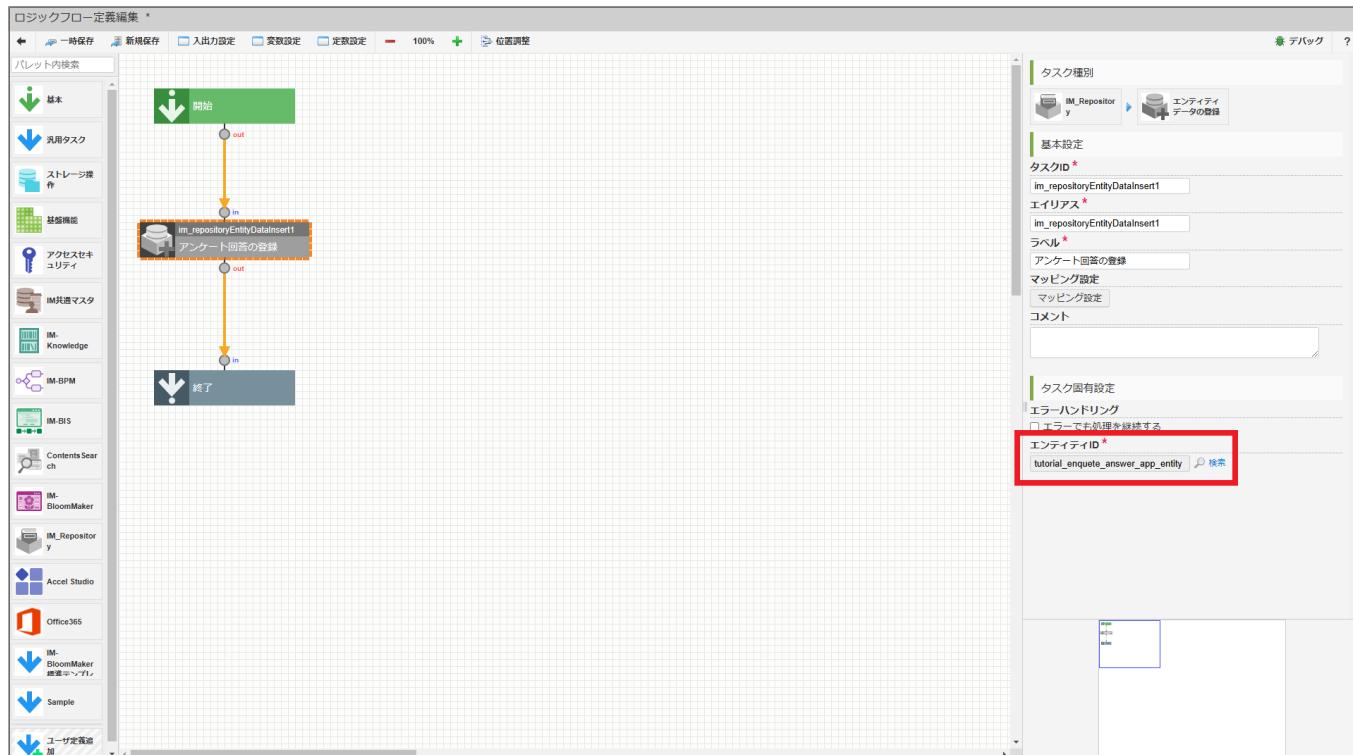


「タスク固有設定」から「エンティティID」を設定します。「検索」アイコンをクリックしてエンティティ検索ウインドウを開いてください。

「アンケート回答」を選択し、「決定」をクリックします。



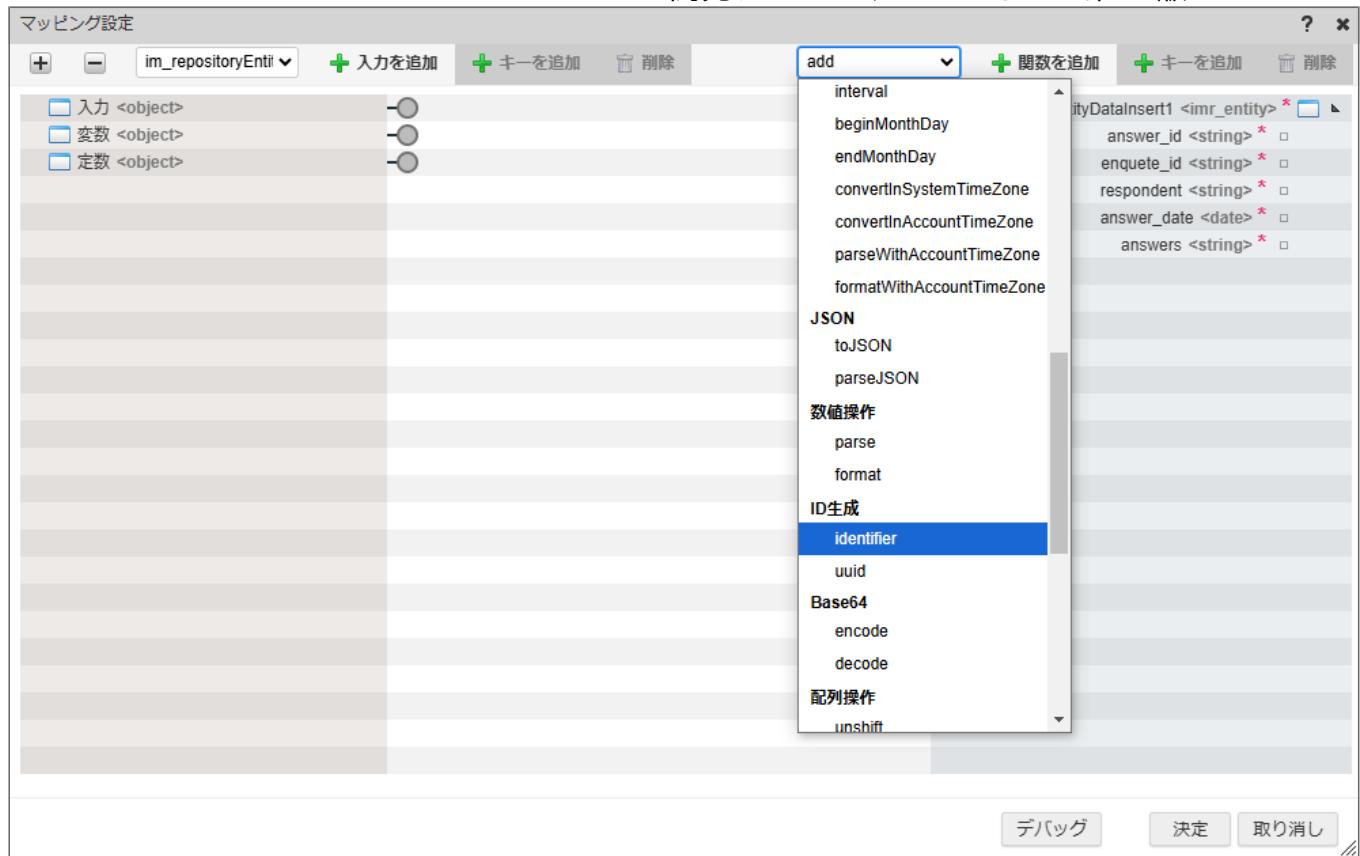
「エンティティID」に「tutorial_enquete_answer_app_entity」が入力されました。



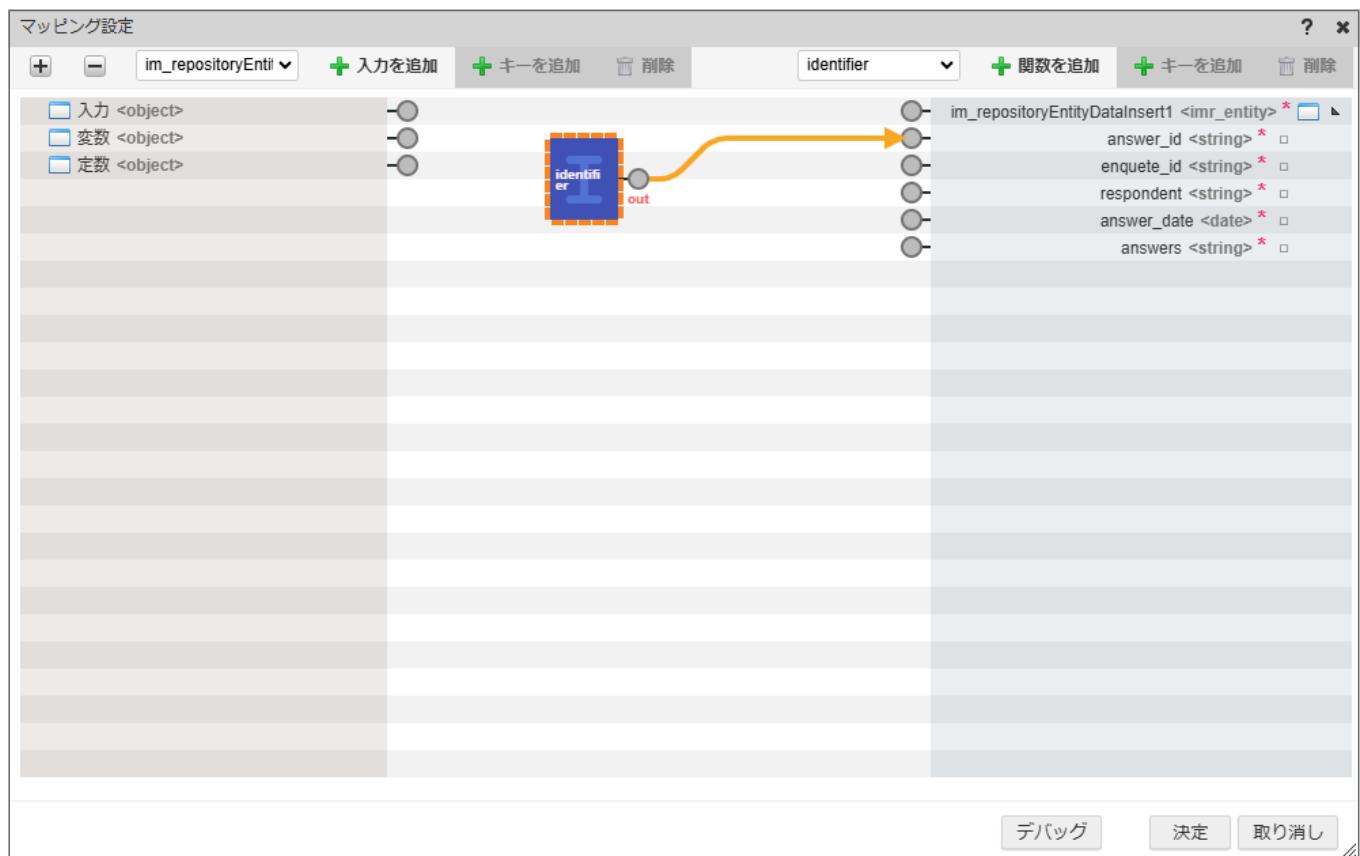
次に、各タスクの変数設定を行います。

「アンケート回答の登録」タスクをダブルクリックしてマッピング設定ダイアログを表示します。

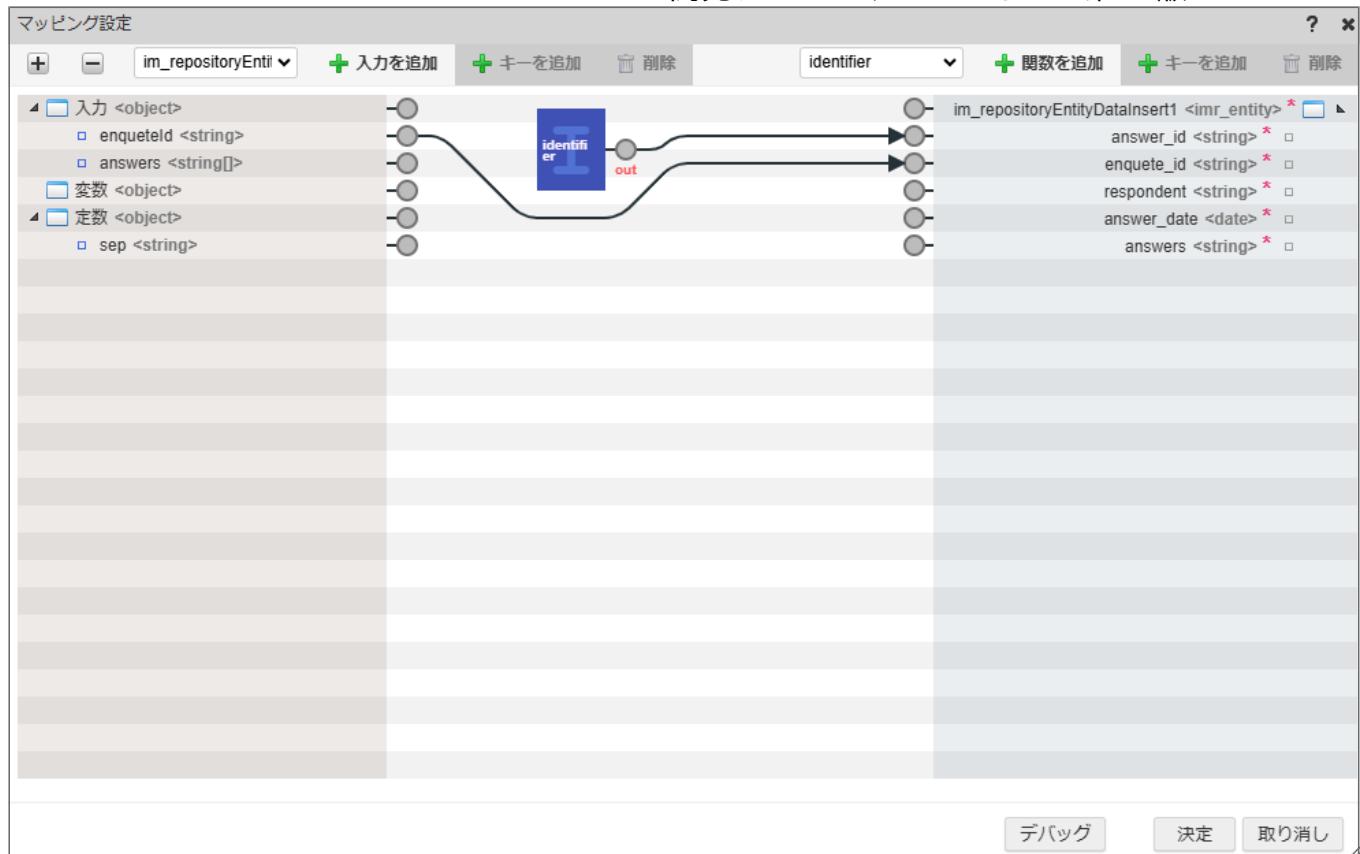
ダイアログ上部の関数追加プルダウンで「identifier」を選択し、「+関数を追加」をクリックして「identifier」関数を追加します。



追加した「identifier」の「out」と「アンケート回答の登録」の入力値の「answer_id」を線で繋げます。



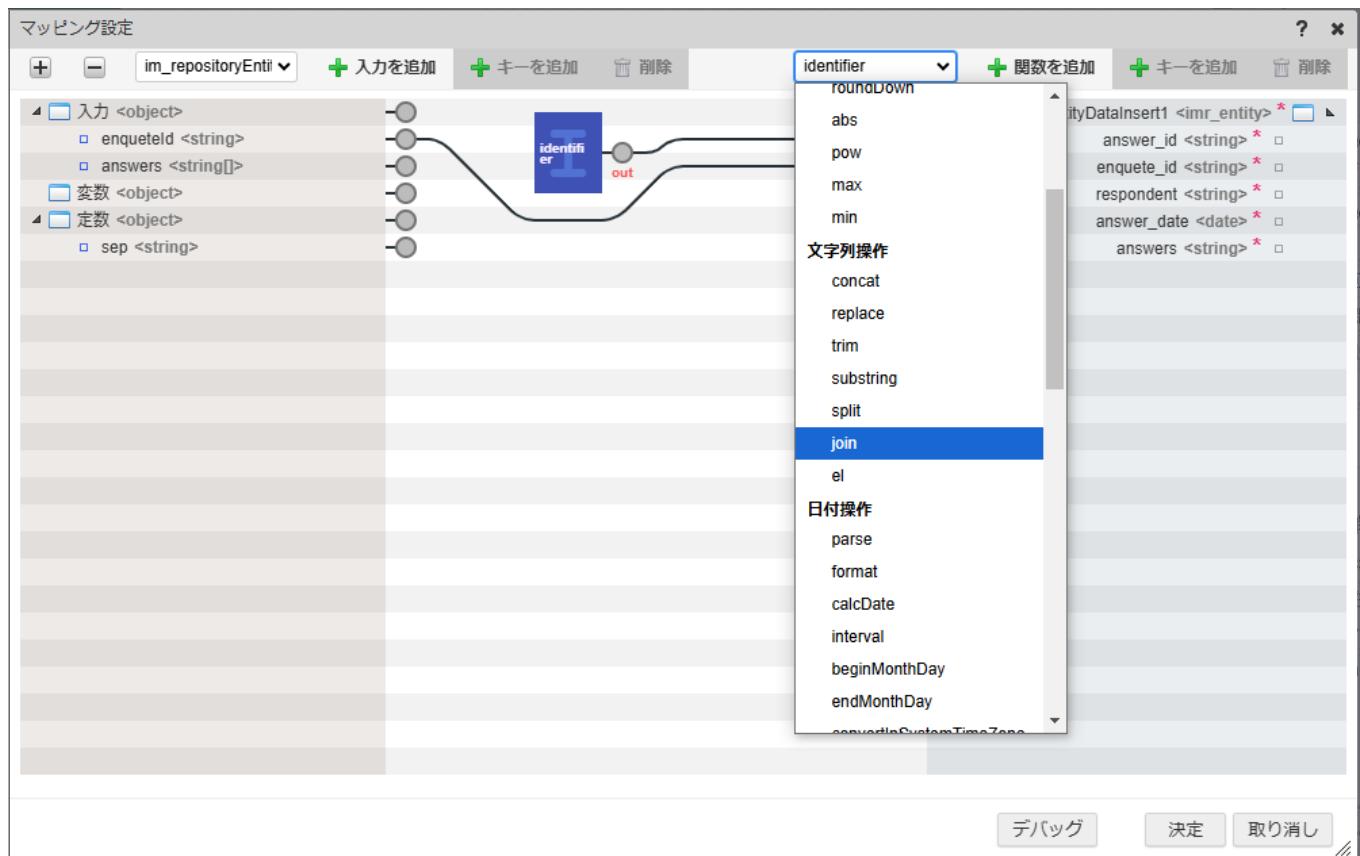
入力値の「enquetelId」と「アンケート回答の登録」の入力値の「enquete_id」を線で繋げます。



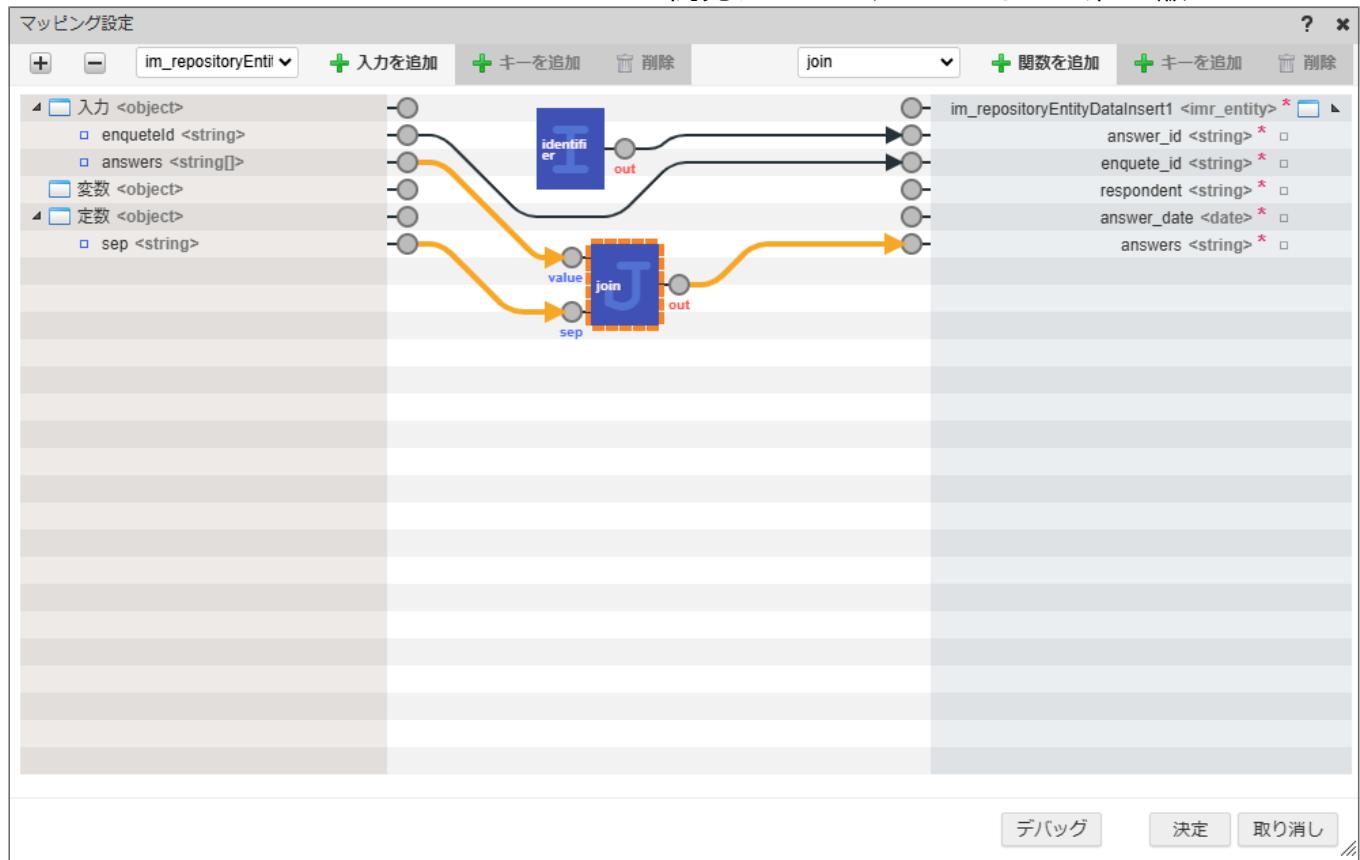
次に、回答内容のデータについて設定していきます。

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、「アンケート回答の登録」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。



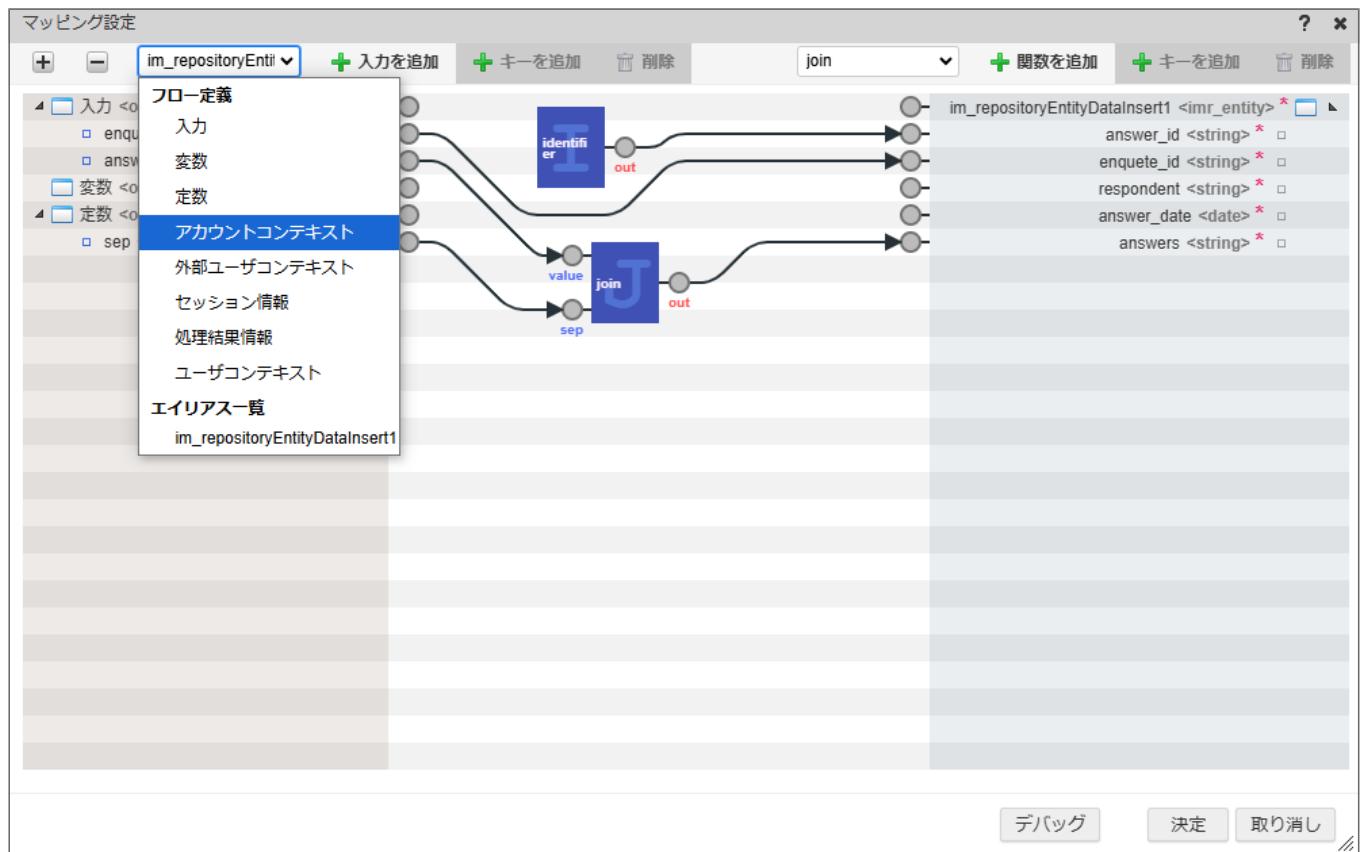
入力値の「answers」を「join」関数の「value」に、定数の「sep」を「join」関数の「sep」にそれぞれ繋げます。
「join」関数のoutと「アンケート回答の登録」のanswersを線で繋げます。



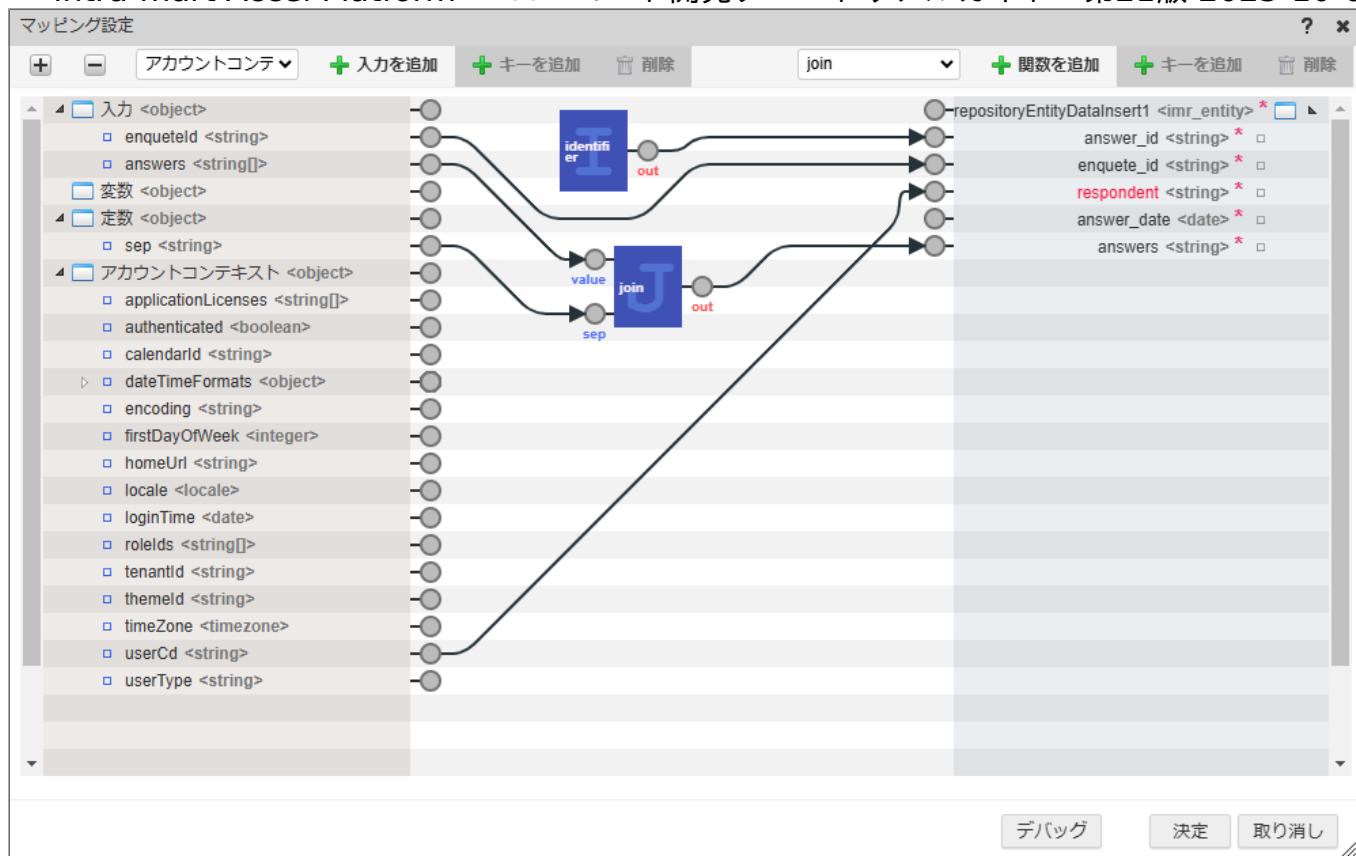
続けて、回答者と回答日を設定します。

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。

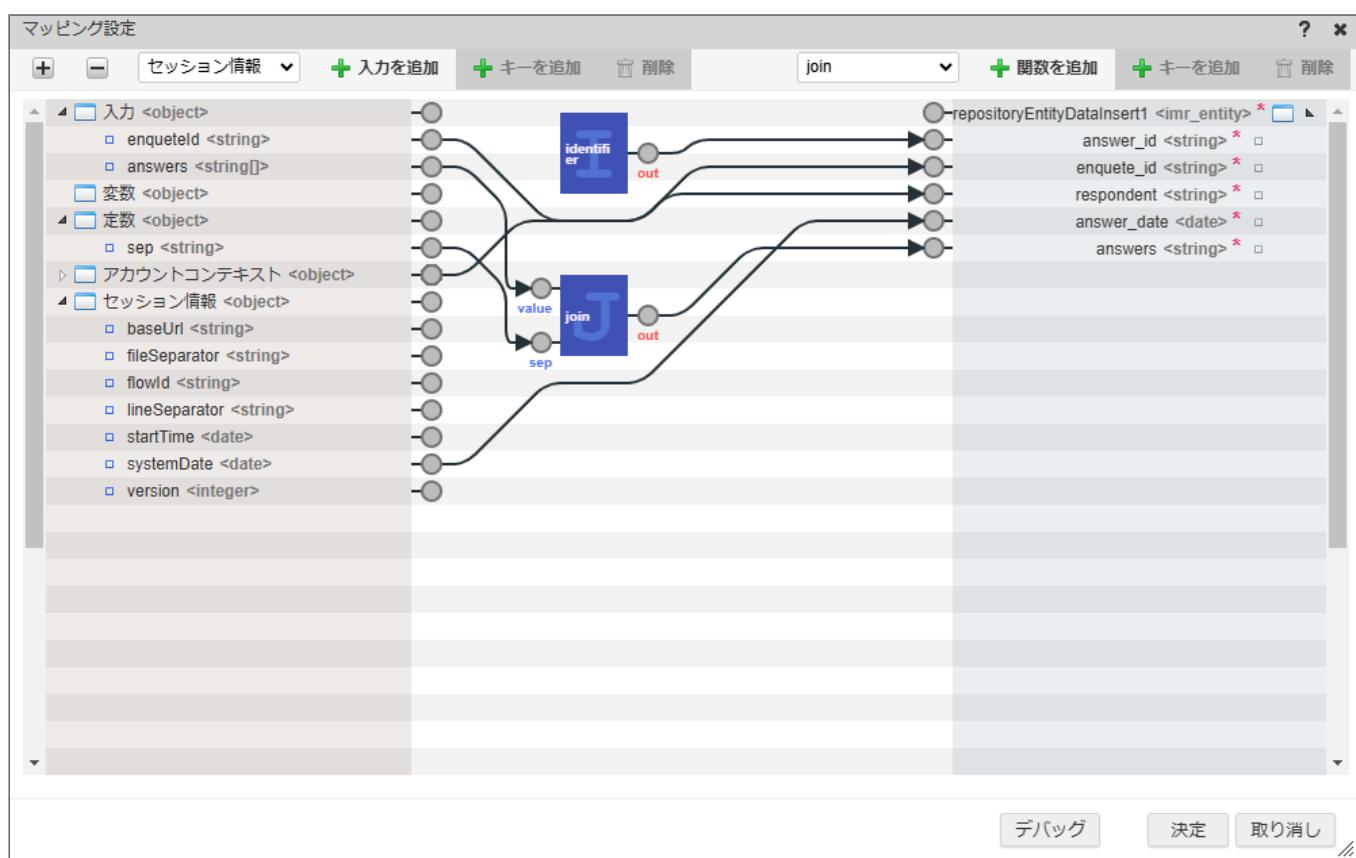


アカウントコンテキストの「userCd」と「アンケート回答の登録」の「respondent」を線で繋げます。



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「アンケート回答の登録」の「answer_date」を線で繋げます。

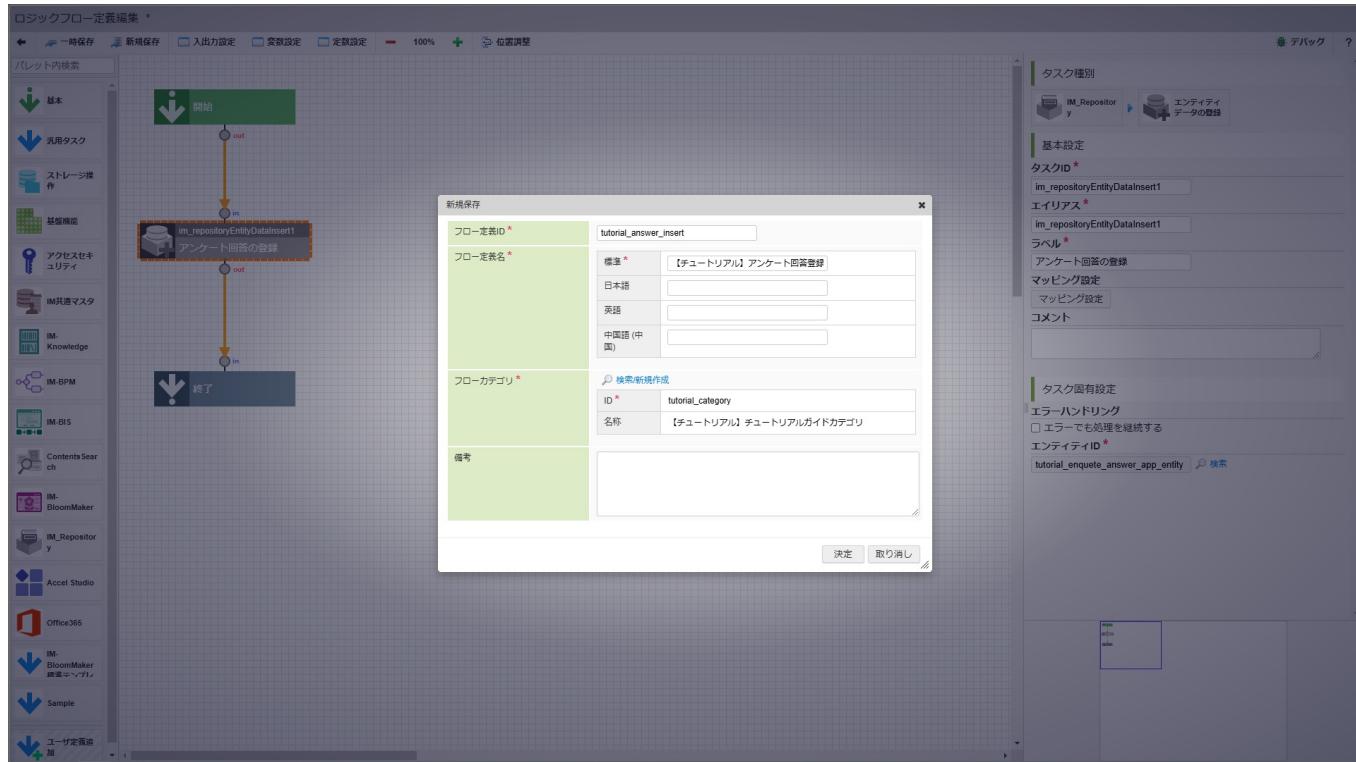


ここまで設定ができたら「決定」をクリックし、マッピング設定を終了します。

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。

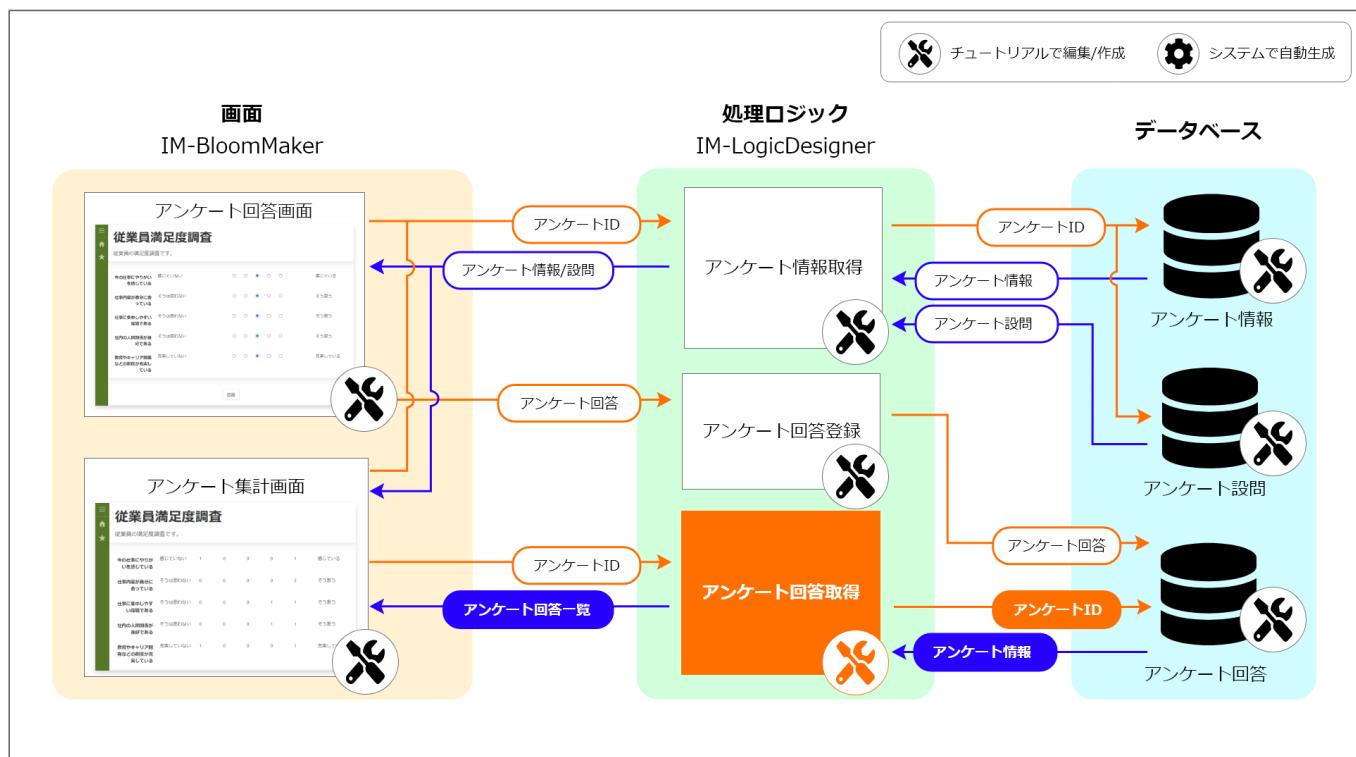
| ラベル | 入力値 |
|------------|-------------------------|
| フロー定義ID | tutorial_answer_insert |
| フロー定義名（標準） | 【チュートリアル】アンケート回答登録 |
| フローカテゴリ | 【チュートリアル】チュートリアルガイドカテゴリ |



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。

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、「ロジックフロー一定義編集画面」を表示します。

The screenshot shows the Accel Studio application management interface. In the top navigation bar, there is a dropdown menu labeled 'リソースを追加' (Add Resource) with a sub-menu open. The sub-menu includes options like 'ロジック' (Logic), 'ワークフロー' (Workflow), 'データ定義' (Data Definition), 'ユーティリティ' (Utility), 'ドキュメント' (Document), and 'テスト' (Test). The 'ロジック' option is highlighted with a red box. On the right side of the screen, there are several panels showing entity definitions for 'アンケート回答' (Survey Answer), 'アンケート設問' (Survey Question), and 'アンケート情報' (Survey Information). Each panel displays the entity ID, name, and a detailed description of the entity's purpose and data structure.

入出力値の設定

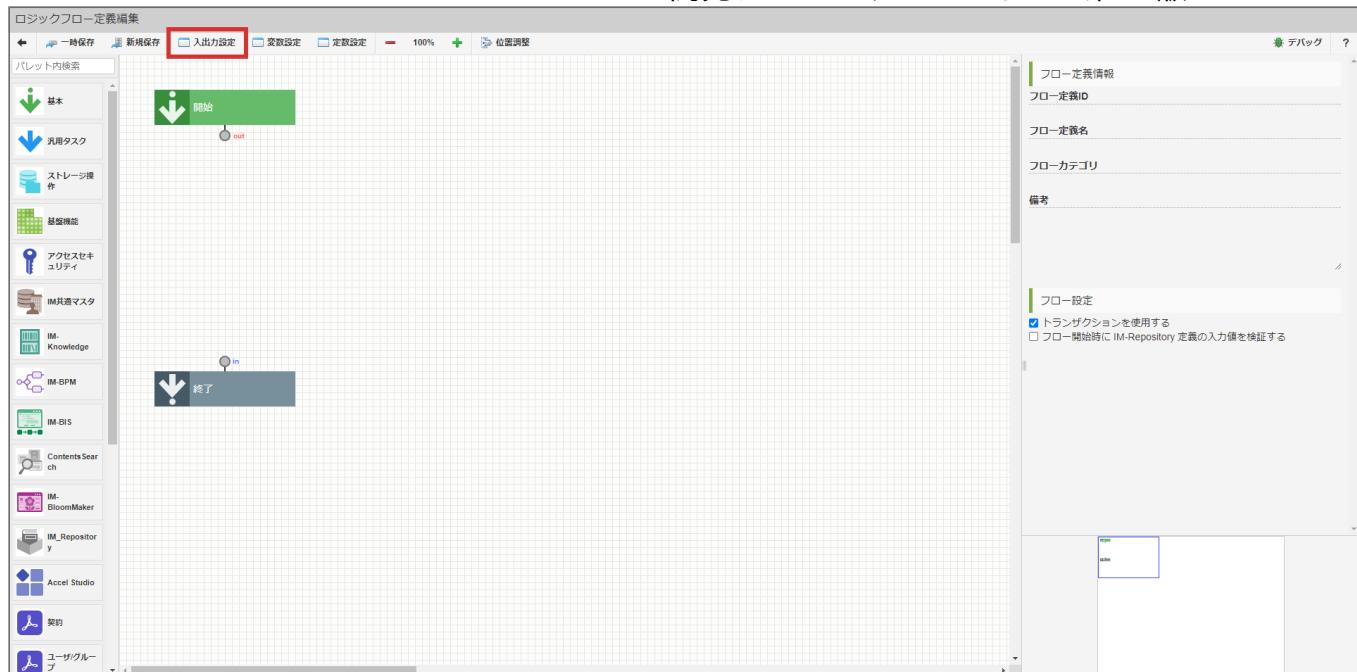
業務ロジックへの入力値と出力値を設定します。

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「キーネームを変更」をクリックし、キーネームを「enquetetId」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。

```
{
  "enquetetId": ""
}
```

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」 - 「+IM-Repository」をクリックして「エンティティ検索」ウインドウを開きます。



「アンケート回答」を選択して「決定」をクリックしてください。



出力値一覧に「tutorial_enquete_answer_app_entity」が追加されます。



「tutorial_enquete_answer_app_entity」を選択した状態で出力値一覧上部の「キーナを変更」をクリックし、キーナを「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、配列型に変更します。



出力値も、JSON入力で下記コードを記述して設定できます。

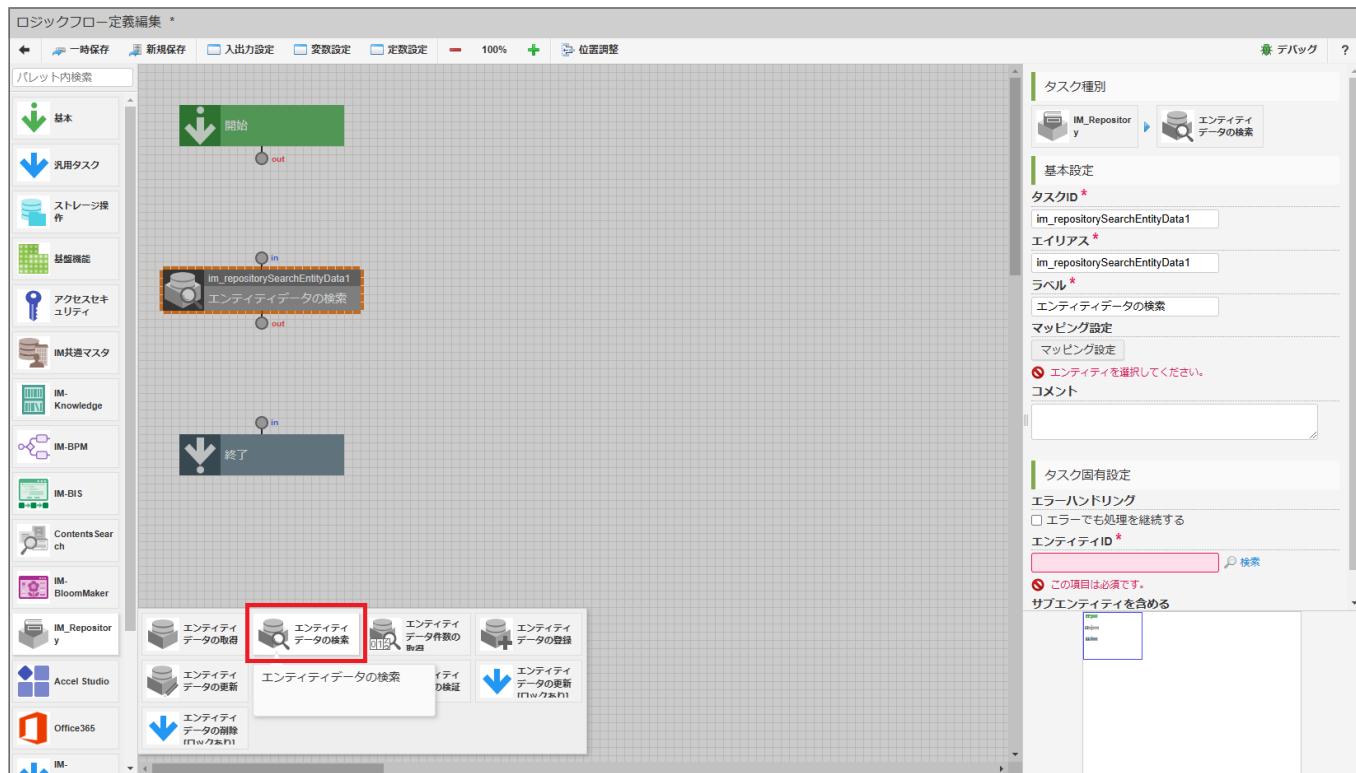
```
{
  "records": [
    {
      "answer_id": "",
      "enquete_id": "",
      "respondent": "",
      "answer_date": null,
      "answers": ""
    }
  ]
}
```

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。

全ての出入力設定が完了したら、出入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。

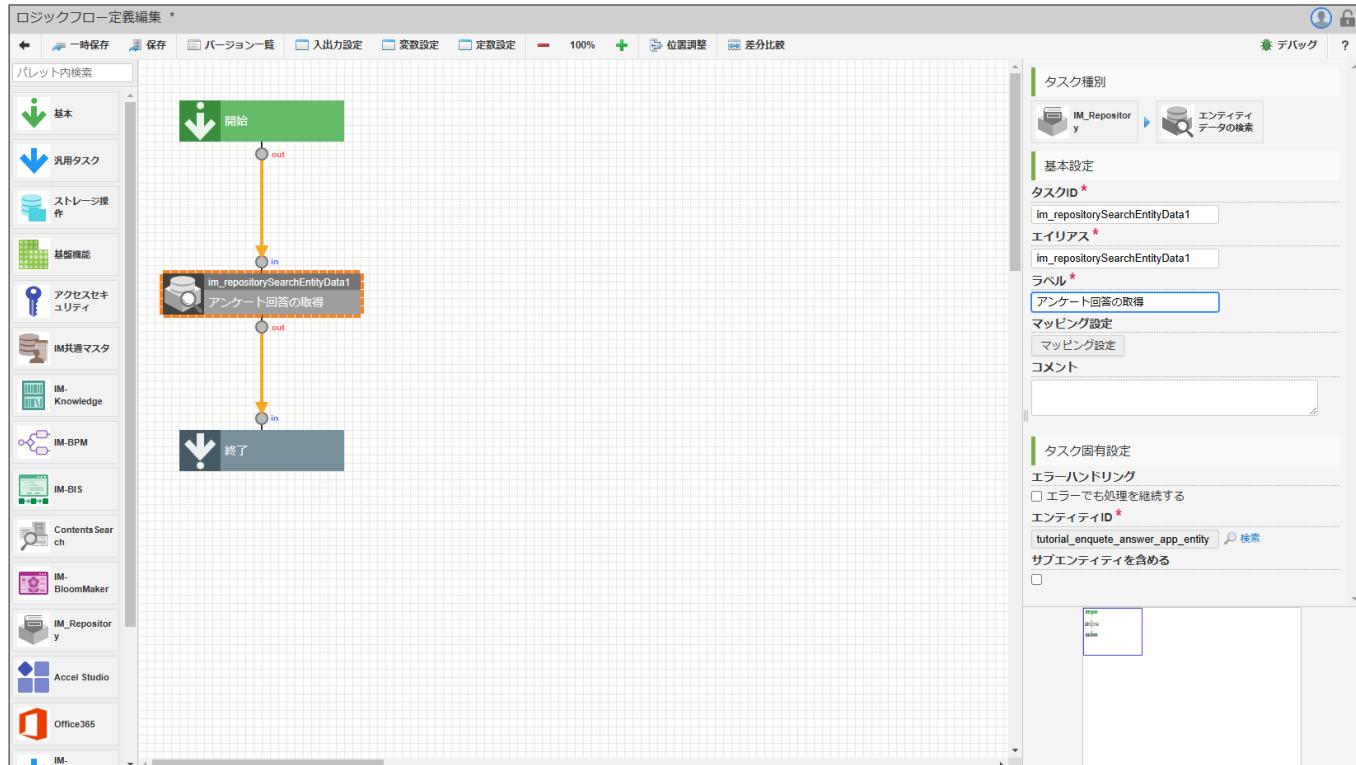
アンケートの回答を取得するタスクを設定します。

パレットから「IM-Repository」 - 「エンティティデータの検索」を選択し、「開始」の次のタスクとして配置します。



配置した「エンティティデータの検索」タスクの「ラベル」を「アンケート回答の取得」に変更します。

「アンケート回答の取得」と「開始」、「終了」のin、outをそれぞれ線で繋げます。



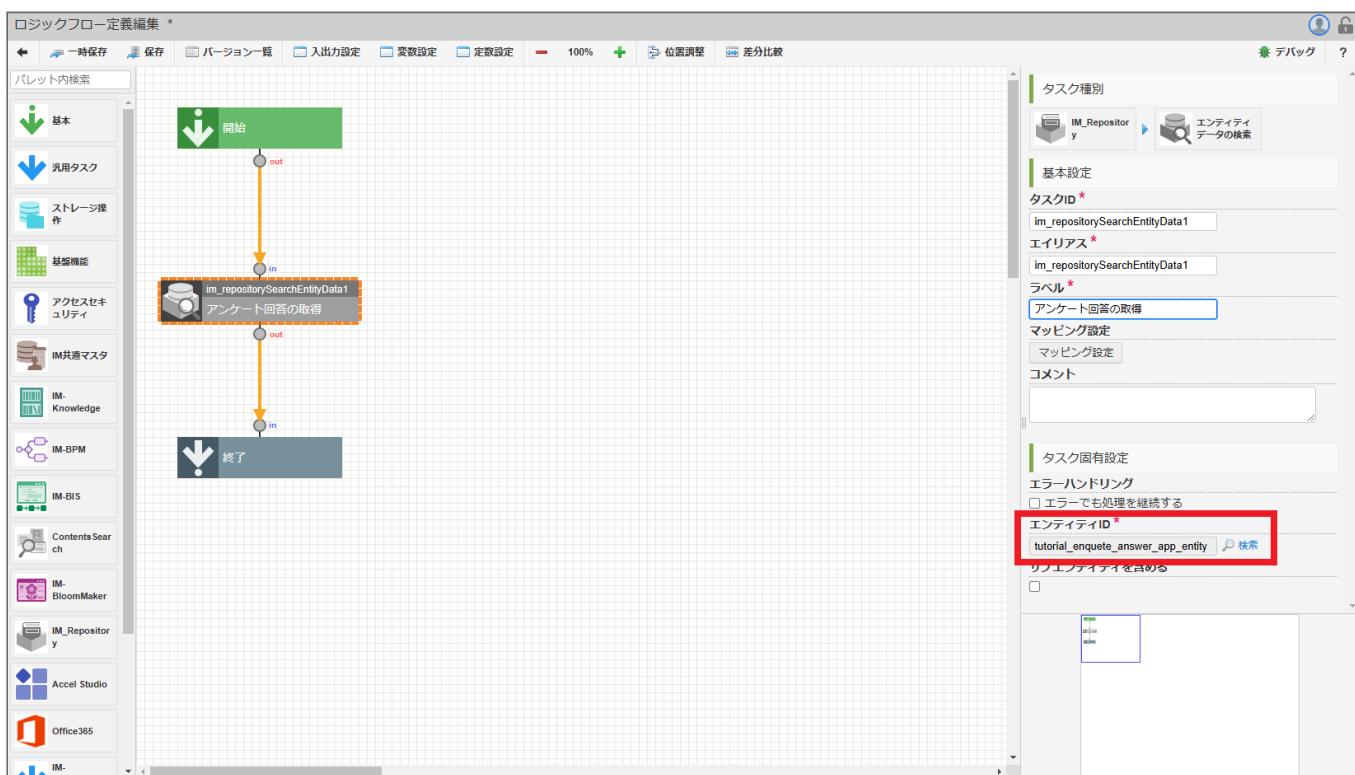
「タスク固有設定」から「エンティティID」を設定します。「検索」アイコンをクリックしてエンティティ検索ウインドウを開いてください。

「アンケート回答」を選択し、「決定」をクリックします。

エンティティ検索



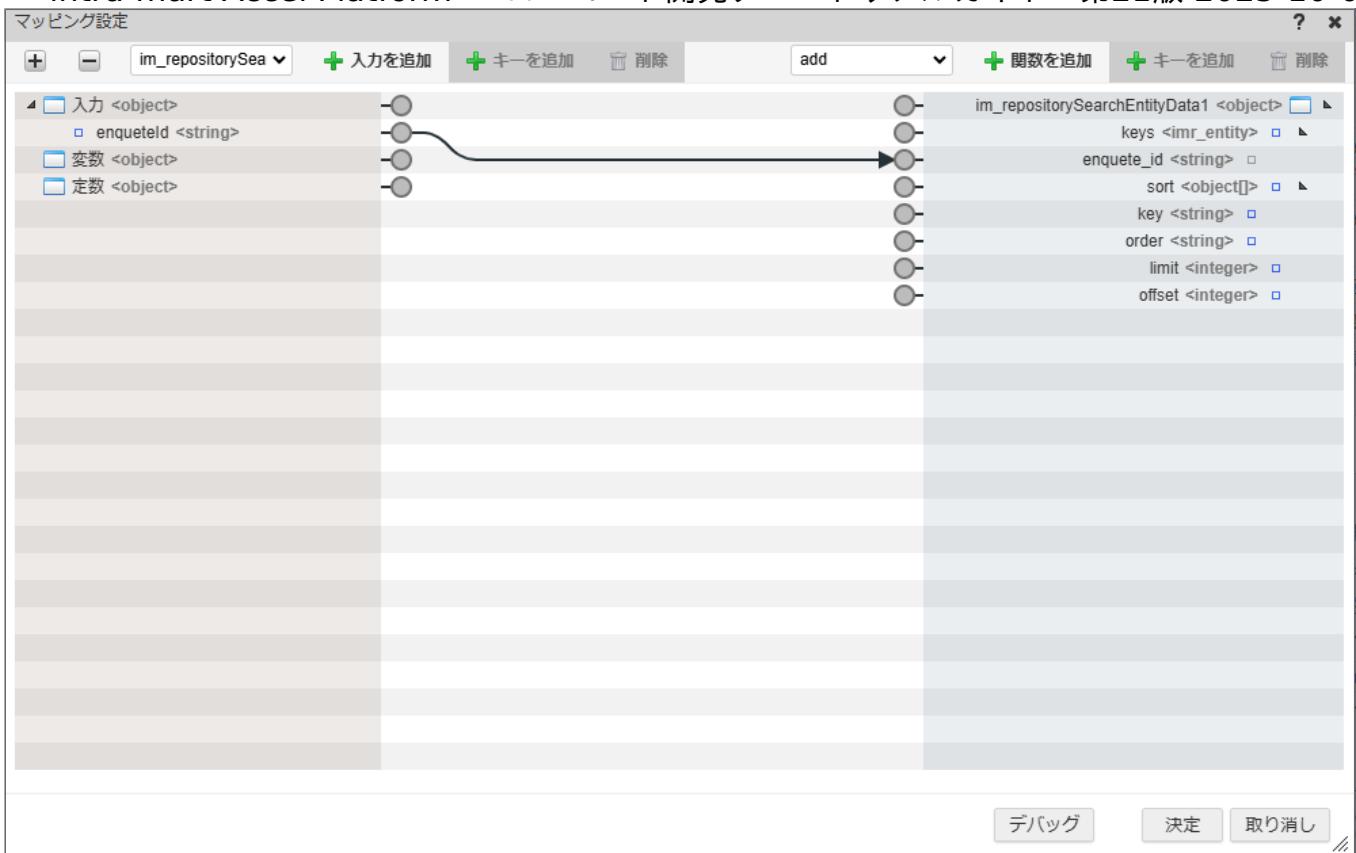
「エンティティID」に「tutorial_enquete_answer_app_entity」が入力されました。



次に、各タスクの変数設定を行います。

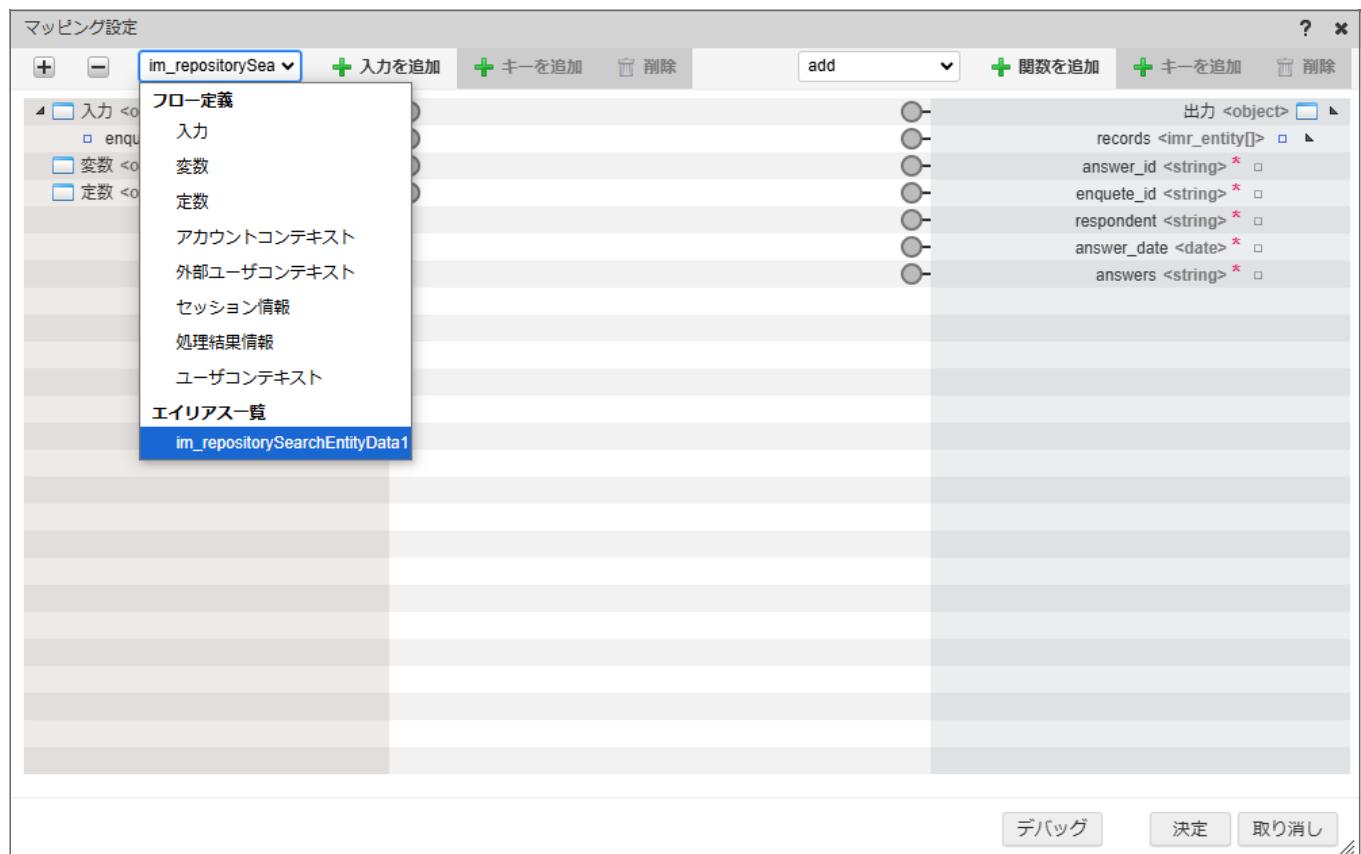
「アンケート回答の取得」タスクをダブルクリックしてマッピング設定ダイアログを表示します。

入力値の「enquetedId」と「アンケート回答の取得」の入力値の「enquete_id」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。

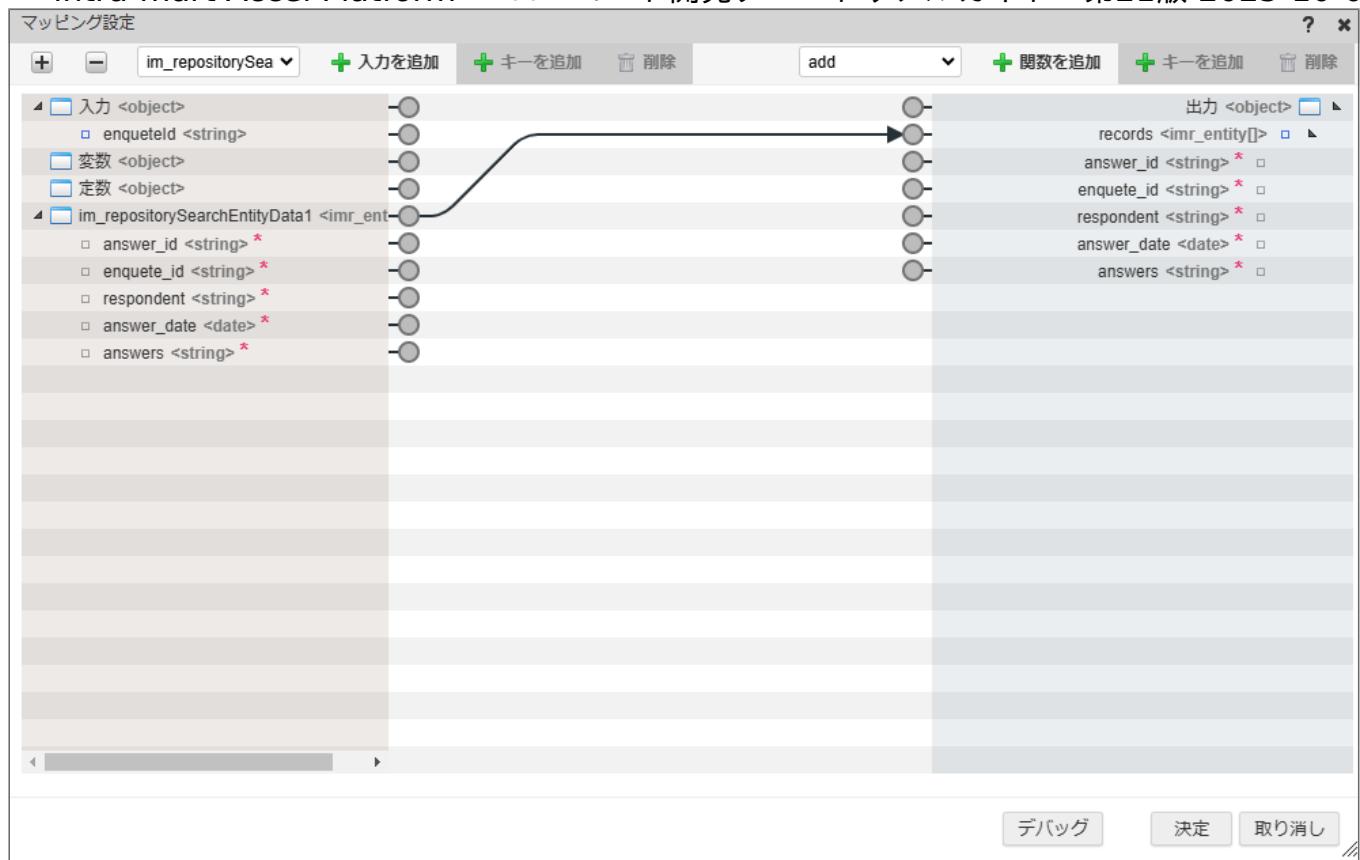


続いて、返却値の設定を行います。「終了」をクリックし、画面の右側領域の「マッピング設定」をクリックしてマッピング設定ダイアログを表示します。

マッピング設定の上部で「im_repositorySearchEntityData1」を選択し、「+入力を追加」をクリックします。



「im_repositorySearchEntityData1」と出力の「records」を線で繋げます。

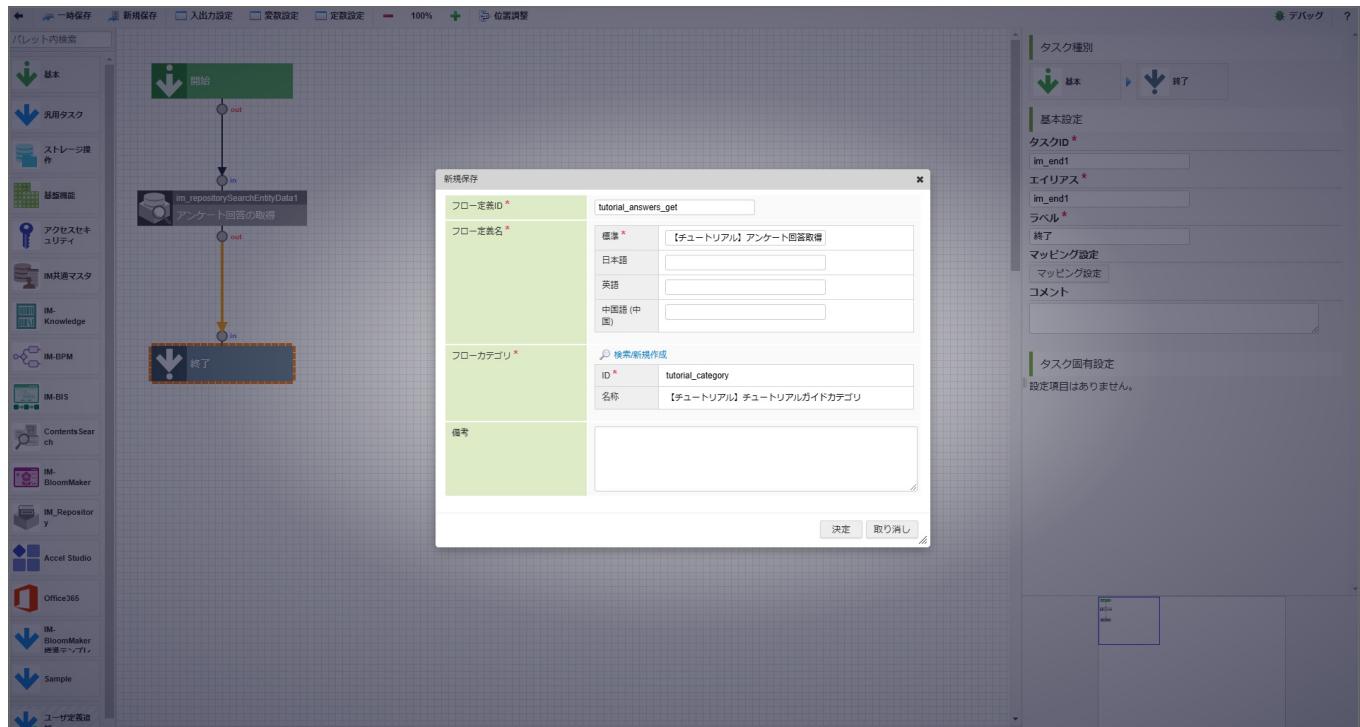


「決定」をクリックし、マッピング設定を終了します。

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 |



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」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、「ロジックフローリング定義編集」画面を表示します。

「ロジックフローリング定義編集」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。

ロジックフロー定義検索

【チュートリアル】 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_answer_insert | 【チュートリアル】アンケート回答登録 | [Preview] |
| <input type="checkbox"/> | tutorial_answers_get | 【チュートリアル】アンケート回答取得 | [Preview] |
| <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未完 | [Preview] |
| <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 | [Preview] |
| <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 | [Preview] |
| <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 ユーザデータ | [Preview] |
| <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 ユーザデータ | [Preview] |
| <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アクション | [Preview] |
| <input checked="" type="checkbox"/> | tutorial_enquete_info | 【チュートリアル】アンケート情報取得 | [Preview] |

1 ページ中 1 ページ目 50 9 件中 1 - 9 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」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 * 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 * /imart/logic/api/tutorial_enquete/question
メソッド * GET
認証方法 * IMAuthentication
認可URI * im-logic-rest://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
+ 追加 ヘッダ名 * ヘッダ値 * 削除

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」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】アンケート回答登録」を選択し、「決定」をクリックします。

| 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 | [Preview icon] |
| <input type="checkbox"/> | tutorial_answers_get | 【チュートリアル】アンケート回答取得 | [Preview icon] |
| <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未実行 | [Preview icon] |
| <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 | [Preview icon] |
| <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 | [Preview icon] |
| <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 ユーザー | [Preview icon] |
| <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 ユーザー | [Preview icon] |
| <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アクション | [Preview icon] |
| <input type="checkbox"/> | tutorial_enquete_info | 【チュートリアル】アンケート情報取得 | [Preview icon] |

「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。

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」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、「ロジックフローリーティング定義編集」画面を開きます。

「ロジックフローリ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。

ロジックフロー定義検索

| 選択 | フロー定義ID | フロー定義名 | プレビュー |
|-------------------------------------|---|-----------------------------|-------|
| <input type="checkbox"/> | tutorial_answer_insert | 【チュートリアル】アンケート回答登録 | |
| <input checked="" 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未実行 | |
| <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 ユーザー情報 | |
| <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 ユーザー情報 | |
| <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アクション | |
| <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

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/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。

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を利用できるようにするために、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。

The screenshot shows the 'Application Management' screen with the 'Permission Setting' dropdown menu open. The 'Permission Setting' option is highlighted with a red box.

| リソース | 認証 | 組織 | ロール |
|------------------|-------------------------------------|-------------------------------------|------------------|
| ゲストユーザー | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | サンプル会社 |
| 認証済みユーザー | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | その他会社 |
| デナント管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | 認可管理者 |
| メニュー管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | メニュー運用管理者 |
| アカウント管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | アカウント管理者 |
| ロール管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | ロール管理者 |
| カレンダー管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | カレンダー管理者 |
| ジョブスケジューリング管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | ジョブスケジューリング管理者 |
| IM共通マスター運用管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM共通マスター運用管理者 |
| ポータル管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | ポータル管理者 |
| IM-Workflow運用管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM-Workflow運用管理者 |
| IM-Workflow監査者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM-Workflow監査者 |
| IM-User ユーザー | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM-User ユーザー |

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 'Permission Setting' screen for the 'Tutorial Application'. The 'Authentication' column for 'Authenticated User' is selected, and the 'Allow selected authentication' button is highlighted with a red box.

| リソース | 認証 | 組織 | ロール |
|------------------|-------------------------------------|-------------------------------------|------------------|
| ゲストユーザー | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | サンプル会社 |
| 認証済みユーザー | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | その他会社 |
| デナント管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | 認可管理者 |
| メニュー管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | メニュー運用管理者 |
| アカウント管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | アカウント管理者 |
| ロール管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | ロール管理者 |
| カレンダー管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | カレンダー管理者 |
| ジョブスケジューリング管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | ジョブスケジューリング管理者 |
| IM共通マスター運用管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM共通マスター運用管理者 |
| ポータル管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | ポータル管理者 |
| IM-Workflow運用管理者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM-Workflow運用管理者 |
| IM-Workflow監査者 | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM-Workflow監査者 |
| IM-User ユーザー | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | IM-User ユーザー |

「更新」をクリックし、認可設定の更新を行います。

The screenshot shows the 'API権限' (API Permissions) configuration page for the '【チュートリアル】アンケートアプリケーション - 認可設定' (Tutorial Survey Application - Authorization Configuration). The main table lists various roles and their permissions for different API endpoints. A modal dialog titled 'アプリケーション認可変更確認' (Application Authorization Change Confirmation) is displayed, asking if the user wants to update application authorization information.

| リソース | 認証 | 組織 | ロール | ゲストユーザー | 認証済みユーザー | 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 | メニュー運用 | アカウント管理 | ロール管理 | カレンダー管理 | ジョブスケジューラ管理 | IM共通マスタ管理 | IM共通マスター運用管理者 | 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-User |
|----------------------------------|----|----|-----|---------|----------|--------|-------|---------|-------|--------|--------|---------|-------|---------|-------------|-----------|---------------|---------|----------------|------------------|----------------|---------|
| [APIのURL] | | | | | | | | | | | | | | | | | | | | | | |
| GET tutorial_enquete/answer/list | × | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| POST tutorial_enquete/answer | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| GET tutorial_enquete/question | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |

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。

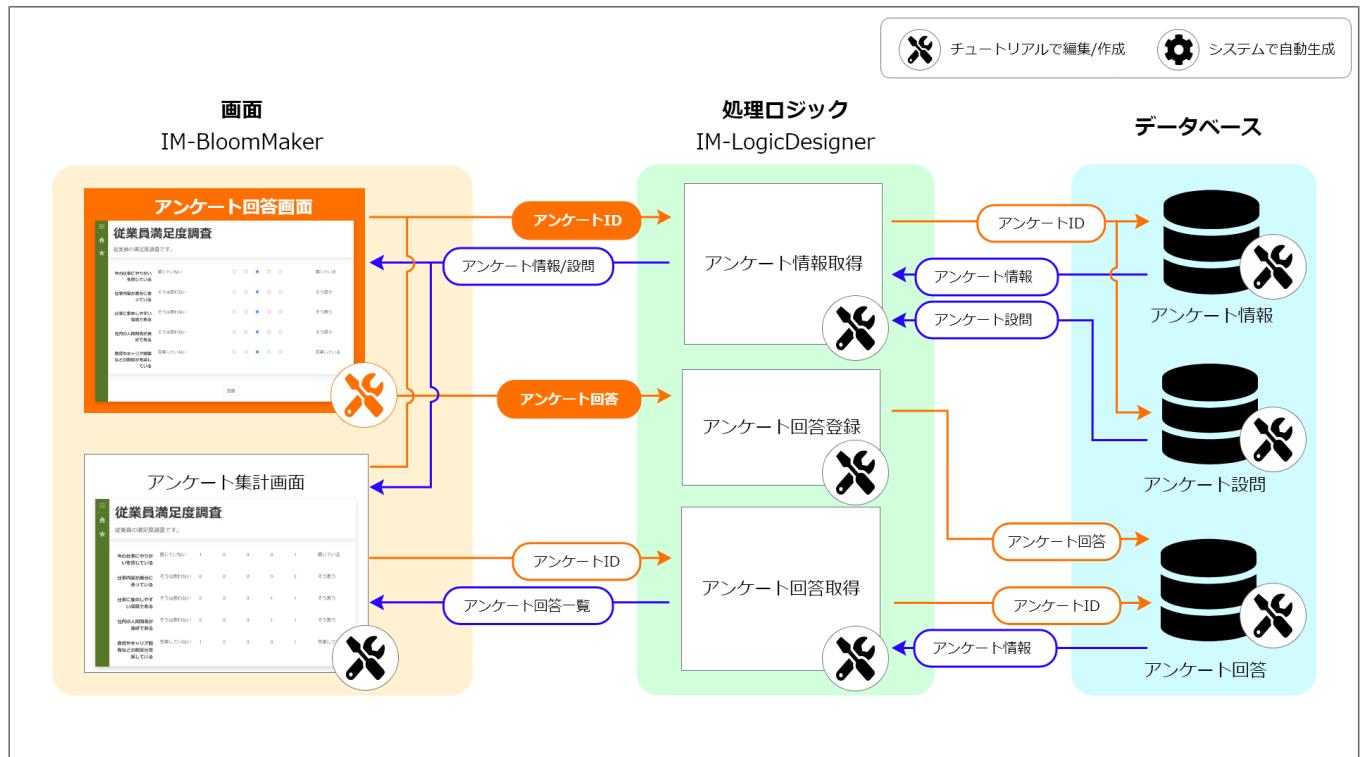
回答画面の作成

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。



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。

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。

The screenshot shows the BloomMaker application management interface. On the left is a vertical sidebar with icons for Help, Home, Applications, and Logout. The main area has a header bar with tabs for Application Management, Application Details, and various logs and documents. Below the header is a search bar and a 'Search Options' dropdown. The main content area displays a list of resources under the 'Tutorial' category. A red box highlights the 'Content' option in the 'Create Resource' dropdown menu. The list includes:

- ロジックフロー (ID: im-logic://flows/tutorial_answers_get) [チュートリアル] アンケート回答取得
- ロジックフロー (ID: im-logic://flows/tutorial_answer_insert) [チュートリアル] アンケート回答登録
- ロジックフロー (ID: im-logic://flows/tutorial_enqueue_info) [チュートリアル] アンケート情報取得
- エンティティ (ID: im-repository://entity/tutorial_enqueue_answer_app_entity) アンケート回答
- エンティティ (ID: im-repository://entity/tutorial_enqueue_question_app_entity) アンケート設問
- エンティティ (ID: im-repository://entity/tutorial_enqueue_info_app_entity) アンケート情報

Each item has a timestamp and a 'tenant' indicator.

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。カテゴリが作成されていない場合は、「サイトマップ」→「BloomMaker」→「コンテンツ一覧」画面の「カテゴリ新規作成」から、カテゴリIDを「im_accel_studio」として作成してください。

コンテンツIDに「tutorial_enqueue_answer」と設定し、コンテンツ種別で「imds」を選択してください。

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。

IM-BloomMaker コンテンツ新規作成

基本

| | | |
|------------|---|---|
| カテゴリ * | <input type="text"/> 検索 カテゴリID: im_accel_studio カテゴリ名: 業務テンプレートから作成 | |
| コンテンツID * | <input type="text"/> tutorial_enquete_answer | |
| コンテンツ種別 * | <input type="radio"/> imui <small>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 開発の 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。</small> <input type="radio"/> Bulma <small>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ます。</small> <input type="radio"/> Bulma Theme Colored <small>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。</small> <input checked="" type="radio"/> imds <small>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、使い勝手の良いユーザインターフェースを効率的に構築できます。</small> | |
| 最新/バージョン番号 | 1 | |
| コンテンツ名 | 標準 * | <input type="text"/> 【チュートリアル】アンケート回答画面 <input style="border: none; background-color: inherit; color: inherit; font-size: inherit;" type="button" value="+"/> |
| 備考 | 標準 | <input type="text"/> |
| テンプレート | <input type="text"/> 検索 テンプレートID テンプレート名 | |

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 デザイナ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

画面

画面コンテンツ (ID: im-bloommaker /contents/tutorial_enquete_answer)

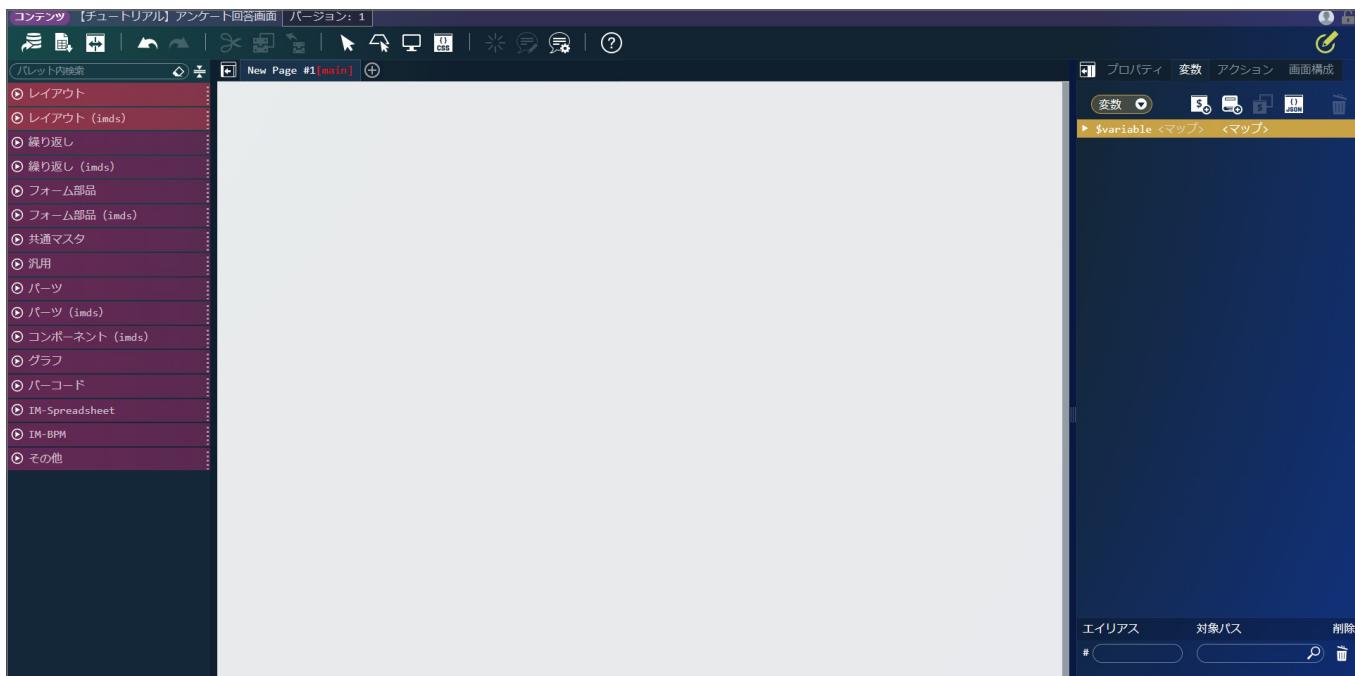
【チュートリアル】アンケート回答画面

tenant 0時間前

変数の設定

画面で利用する変数を作成します。

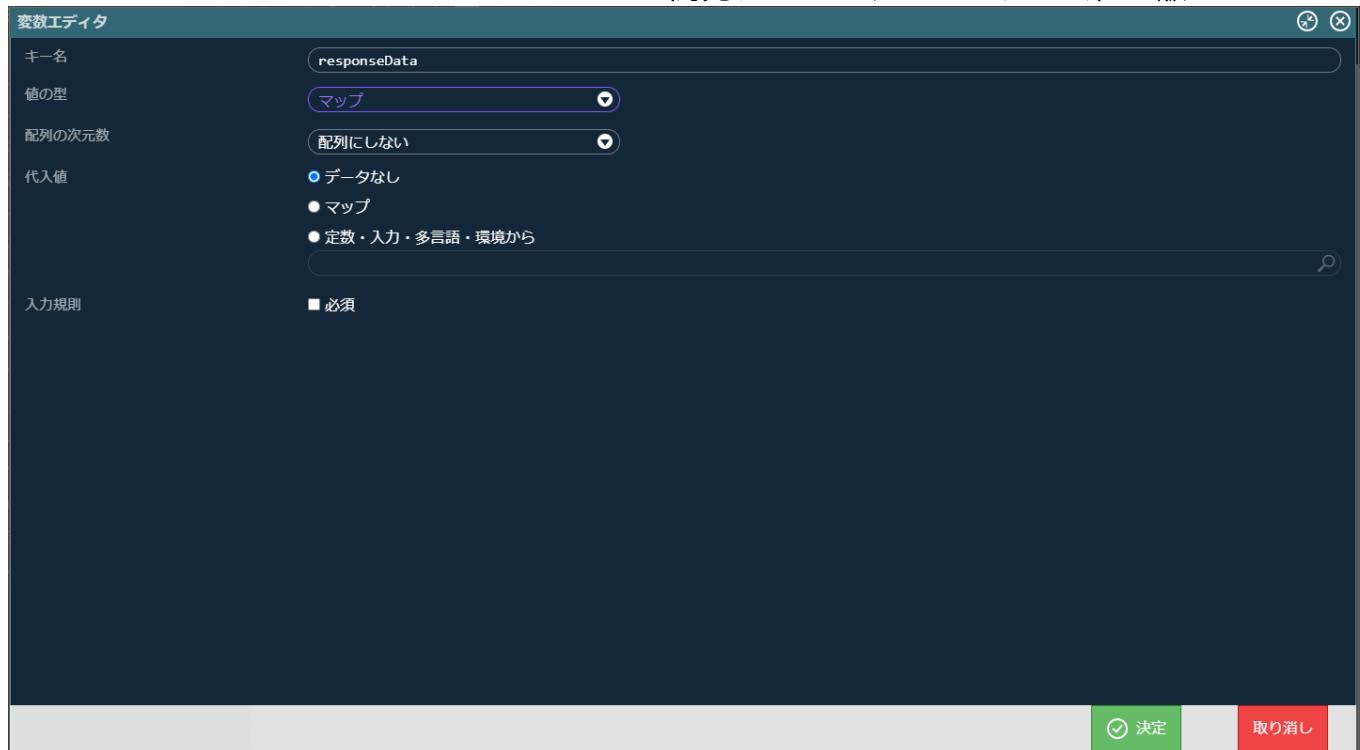
まず、画面に表示する設問情報を格納するための変数を作成します。この変数はアンケート情報取得処理の返却値を格納し、画面に表示するために利用します。



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。



作成した「\$variable > responseData」を選択し、「IM-Repositoryから作成」アイコンをクリックしてプルダウンから「エンティティをインポート」を選択します。



表示された「エンティティ検索」画面から、「アンケート設問」を選択して決定をクリックしてください。

エンティティ検索

【チュートリアル】チュートリアル
アンケート回答
アンケート設問 ✓
アンケート情報

詳細情報

基本 エンティティ項目 関連一覧 データ定義

| | | | |
|------------|--------------------------------------|---------|--|
| エンティティID | tutorial_enquete_question_app_entity | | |
| エンティティ名 | 標準 | アンケート設問 | |
| | 日本語 | | |
| | 英語 | | |
| | 中国語 (中国) | | |
| 説明 | 標準 | | |
| | 日本語 | | |
| | 英語 | | |
| | 中国語 (中国) | | |
| エンティティの有効化 | <input checked="" type="checkbox"/> | | |

選択済み項目
アンケート設問

クリア 決定

「\$variable > responseData」配下に「tutorial_enquete_question_app_entity」が追加されました。

プロパティ 変数 アクション 画面構成

変数

\$variable <マップ>

responseData <マップ>

tutorial_enquete_question_app_entity <マップ>

- enqueuee_id <文字列>
- question_id <文字列>
- question_no <浮動小数点数>
- question <文字列>
- left_scale <文字列>
- right_scale <文字列>

次に、「tutorial_enquete_question_app_entity」の をクリックして変数エディタ画面を表示します。

キー名を「records」に変更し、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。

変数エディタ

キーナム
records

コメント

IM-Repository 定義の再読み込み

IM-Repository 定義の再読み込み

配列の次元数
1次元配列

配列の要素数
1

代入値
配列

● 定数・入力・多言語・環境から

検索

■ 必須

IM-Repository 定義は、「必須」「フォーマットに則していないときのエラーメッセージ」以外の入力規則を変更できません。
入力規則は IM-Repository 側で変更後、「IM-Repository 定義の再読み込み」をクリックしてください。

決定 取り消し

「records」配下に作成された「0」の  をクリックして、変数エディタ画面で代入値を「マップ」に変更します。

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ナムに「enqueteTitle」と入力し、値の型を「文字列」として「決定」をクリックします。

プロパティ 变数 アクション 画面構成

変数

| | |
|----------------------|---------|
| \$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> | <データなし> |
| question_id <文字列> | <データなし> |
| question_no <浮動小数点数> | <データなし> |
| question <文字列> | <データなし> |
| left_scale <文字列> | <データなし> |
| right_scale <文字列> | <データなし> |
| enqueteTitle <文字列> | <データなし> |

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": [
    {
      "enquete_id": "",
      "question_id": "",
      "question_no": "",
      "question": "",
      "left_scale": "",
      "right_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な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「enquetedId」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": {
  "enquetedId": ""
}
```

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。

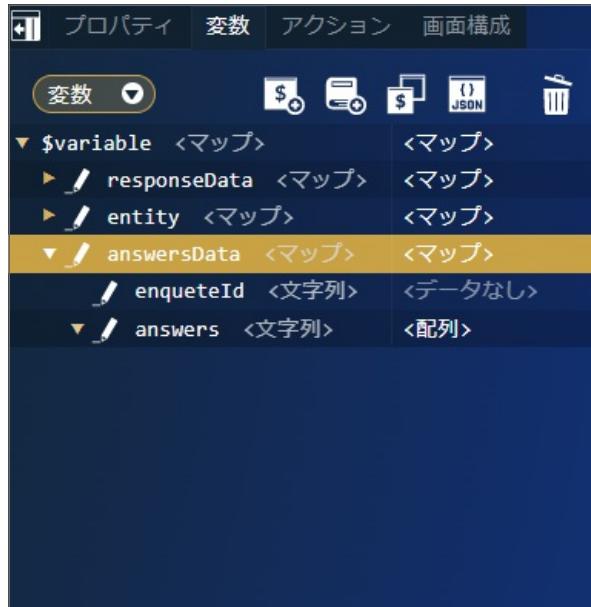


作成した「\$variable > answersData」を選択し、 をクリックして変数エディタを表示します。

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

同様に、「\$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」として「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。

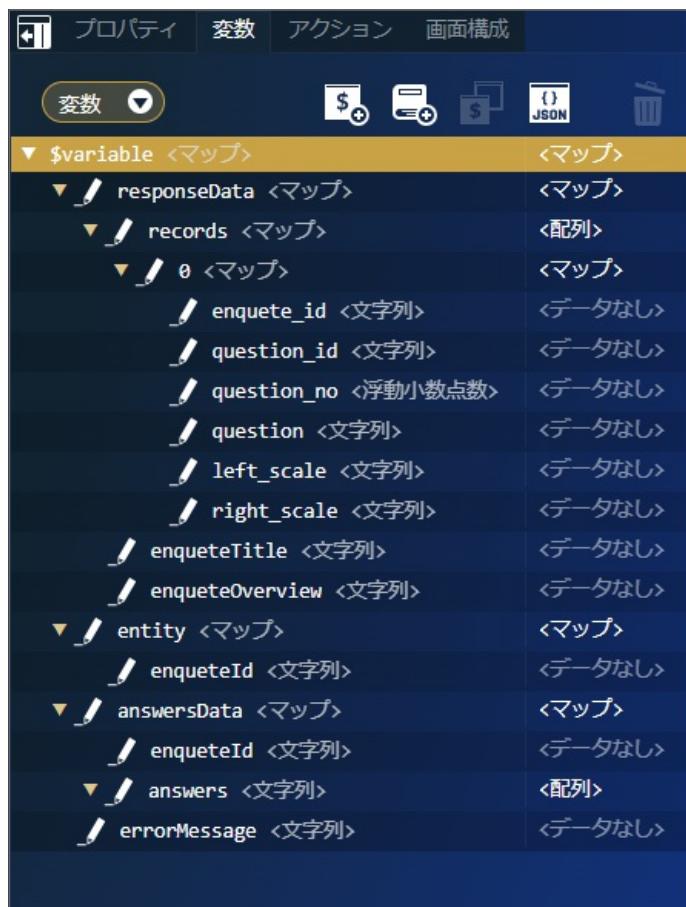
```
"answersData": {  
    "enquetelId": "",  
    "answers": []  
}
```

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。

次に、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」として「決定」をクリックします。

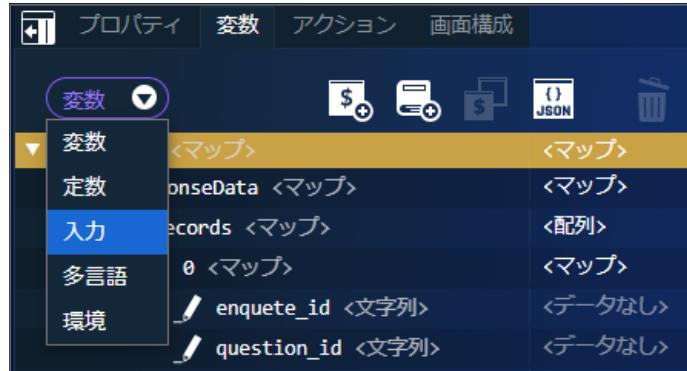


```
"errorMessage": ""
```

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「enquetetId」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。

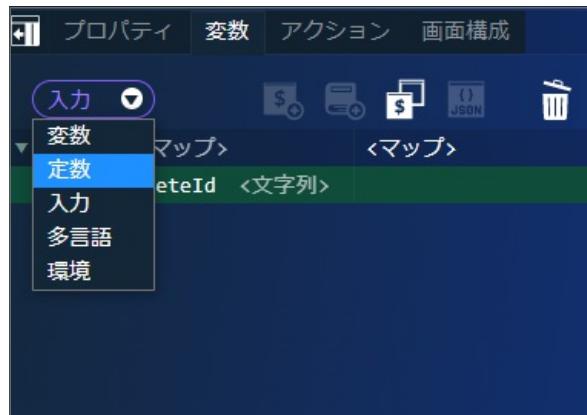
入力値はJSON入力で以下のコードを記載しても設定できます。

```
"enquetetId": ""
```

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。

The screenshot shows a configuration interface for constants. At the top, there are tabs for 'プロパティ' (Properties), '変数' (Variables), 'アクション' (Actions), and '画面構成' (Screen Configuration). The '変数' tab is selected. Below the tabs, there's a toolbar with icons for '定数' (Constant), '\$' (Currency), '+', '\$+' (Currency +), 'JSON', and a trash bin. A dropdown menu labeled '\$constant <マップ>' is open, showing two entries: '_error_no_id <文字列>' with the value 'アンケートIDが設定されていません。' and '_error_enquete_missing <文字列>' with the value 'アンケートが存在しません。'.

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。

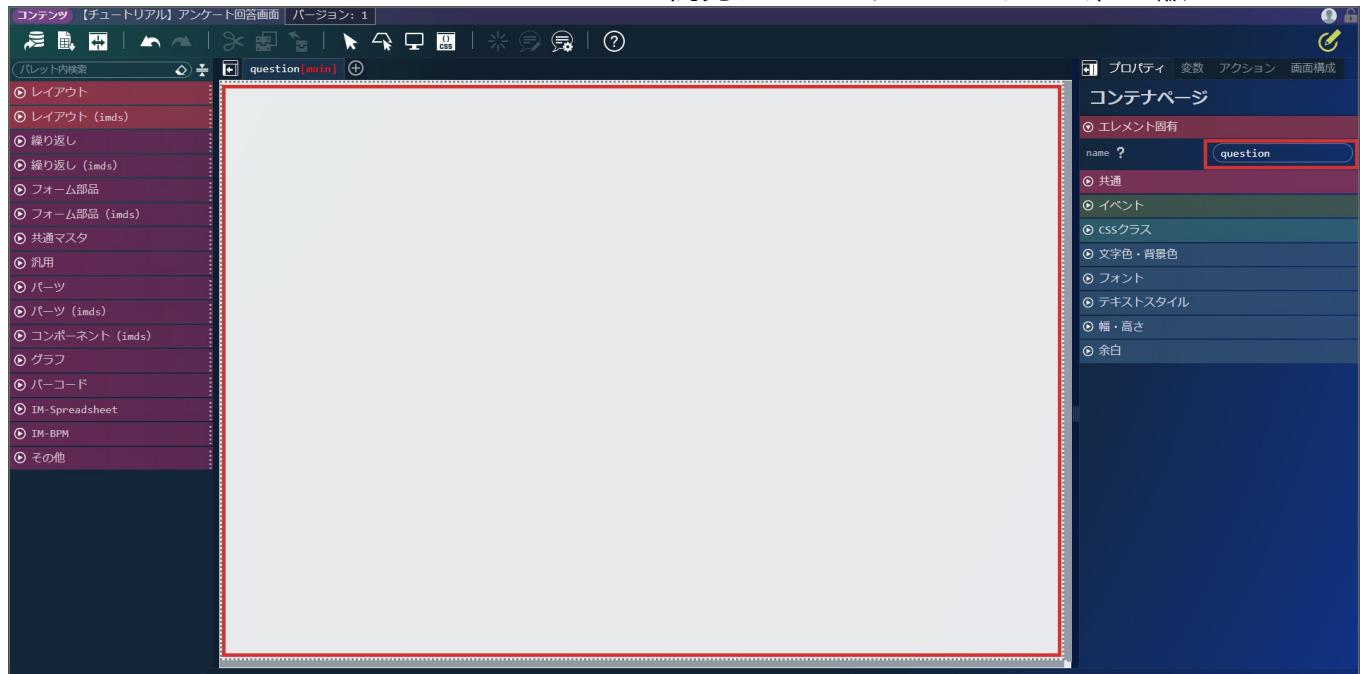
アンケート回答画面の作成

続けて、画面にアイテムを配置していきます。

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。

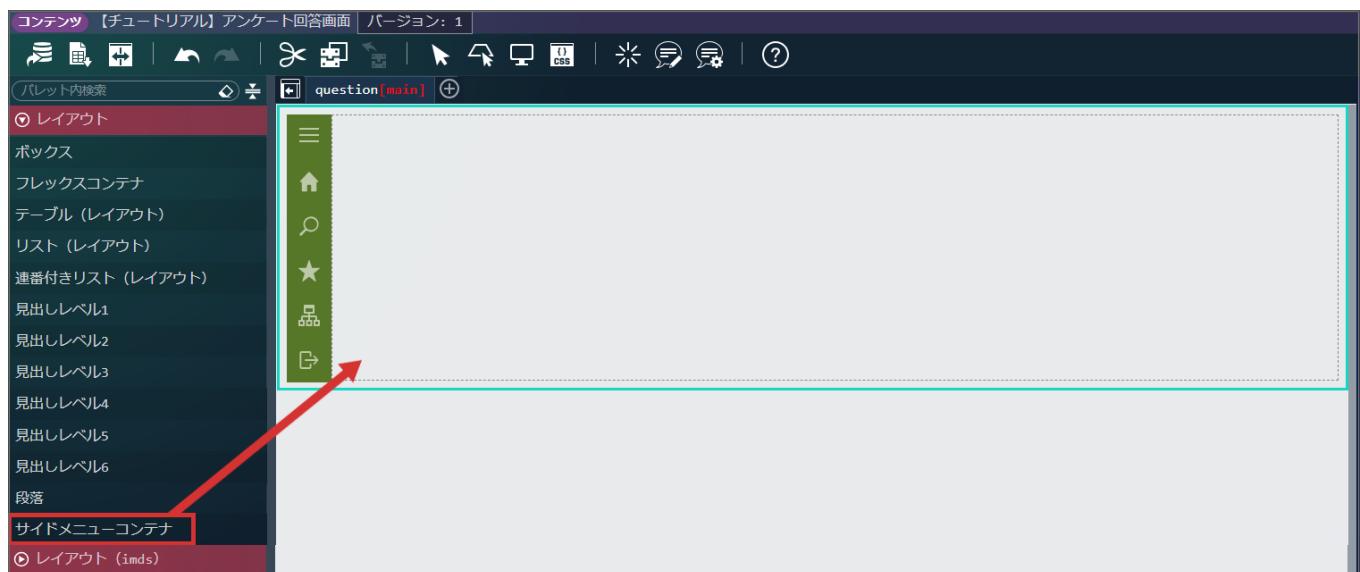
まずは、回答画面の名称を変更します。

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。



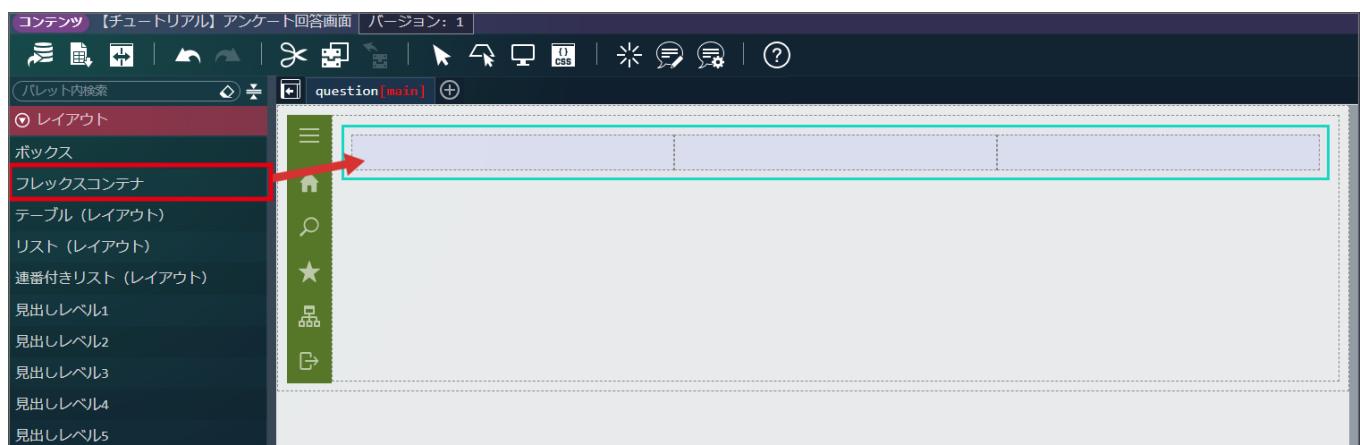
これで、タブの名称も「question」へ変更されました。次に、画面にサイドメニューを表示するため、サイドメニュー コンテナを配置します。

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。

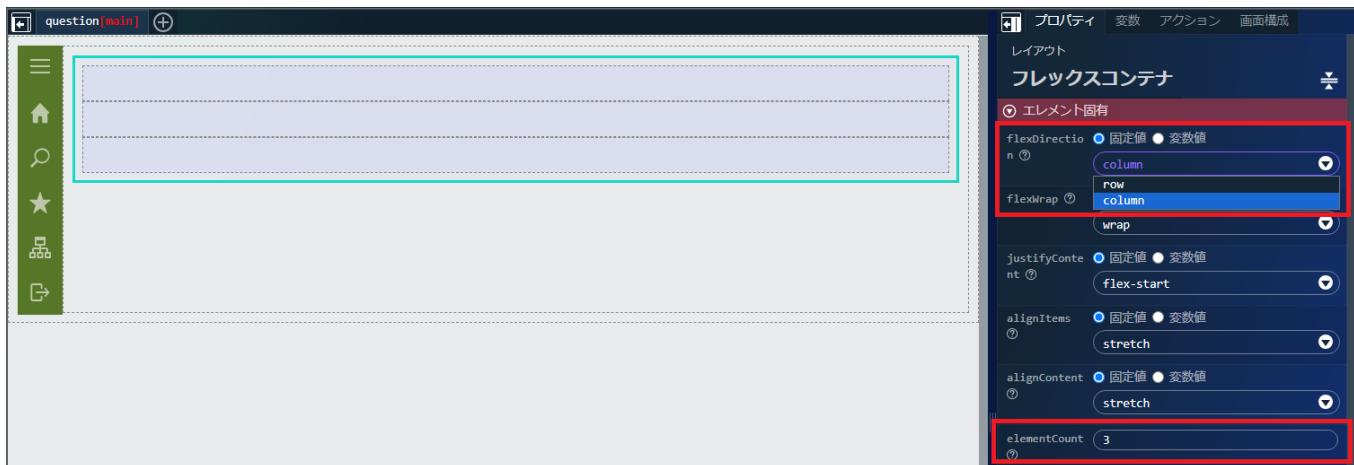


次に、アンケートのタイトルと概要の表示領域を作成します。

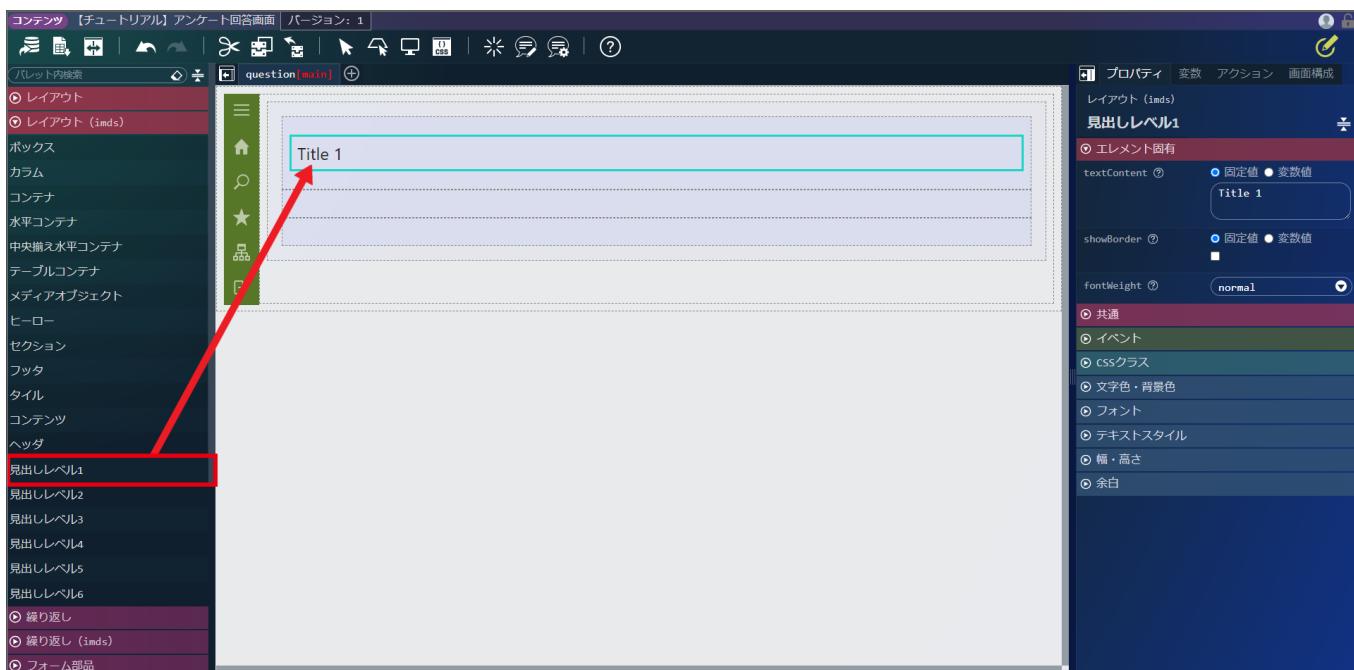
画面左側のパレットの「レイアウト」 - 「フレックスコンテナ」を選択し、先ほど配置した「サイドメニュー コンテナ」の中に配置します。



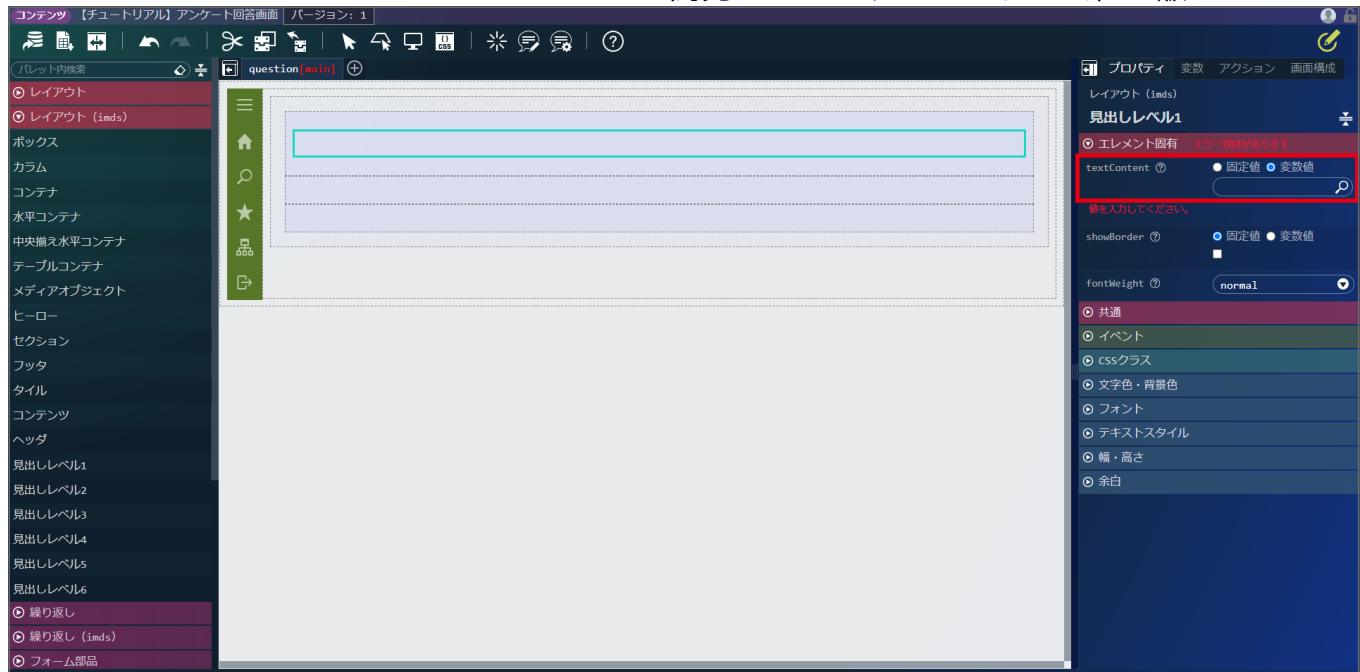
「エレメント固有」 - 「flexDirection」を「column」に変更してください。



画面左側のパレットの「レイアウト (imds)」 - 「見出しレベル1」を選択し、先ほど配置した「フレックスコンテナ」の一番上の「フレックスアイテム」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・定数・入力・多言語・環境セレクタ

変数 (selected) 定数 入力 多言語 環境

キー名を検索

| | |
|-----------------------|---------|
| ▼ \$variable <マップ> | <マップ> |
| ▼ responseData <マップ> | <マップ> |
| ▼ records <配列> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| enquete_id <文字列> | <データなし> |
| question_id <文字列> | <データなし> |
| question_no <浮動小数点数> | <データなし> |
| question <文字列> | <データなし> |
| left_scale <文字列> | <データなし> |
| right_scale <文字列> | <データなし> |
| enqueteTitle <文字列> | <データなし> |
| enqueteOverview <文字列> | <データなし> |
| ▼ entity <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answersData <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answers <配列> | <配列> |
| errorMessage <文字列> | <データなし> |

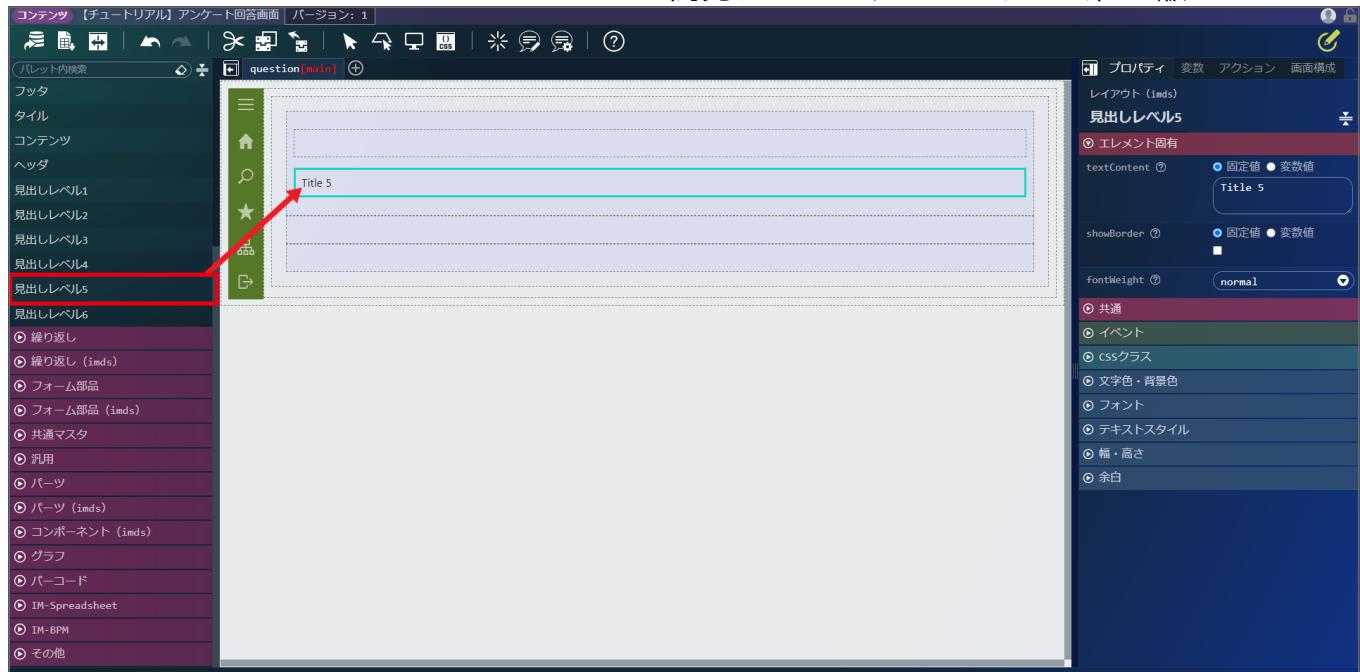
繰り返しエレメントを考慮 ②

決定 取り消し

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 (imds)」 - 「見出しレベル5」を選択し、先ほど配置した「見出しレベル1」の下に配置します。



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・定数・入力・多言語・環境セレクタ

変数 定数 入力 多言語 環境

キー名を検索 ×

繰り返しエレメントを考慮 ?

| | |
|-----------------------|---|
| ▼ \$variable <マップ> | <マップ> |
| ▼ responseData <マップ> | <マップ> |
| ▼ records <配列> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| enquete_id <文字列> | <データなし> |
| question_id <文字列> | <データなし> |
| question_no <浮動小数点数> | <データなし> |
| question <文字列> | <データなし> |
| left_scale <文字列> | <データなし> |
| right_scale <文字列> | <データなし> |
| enqueteTitle <文字列> | <データなし> |
| enqueteOverview <文字列> | <データなし> |
| ▼ entity <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answersData <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answers <配列> | <配列> |
| errorMessage <文字列> | <データなし> |

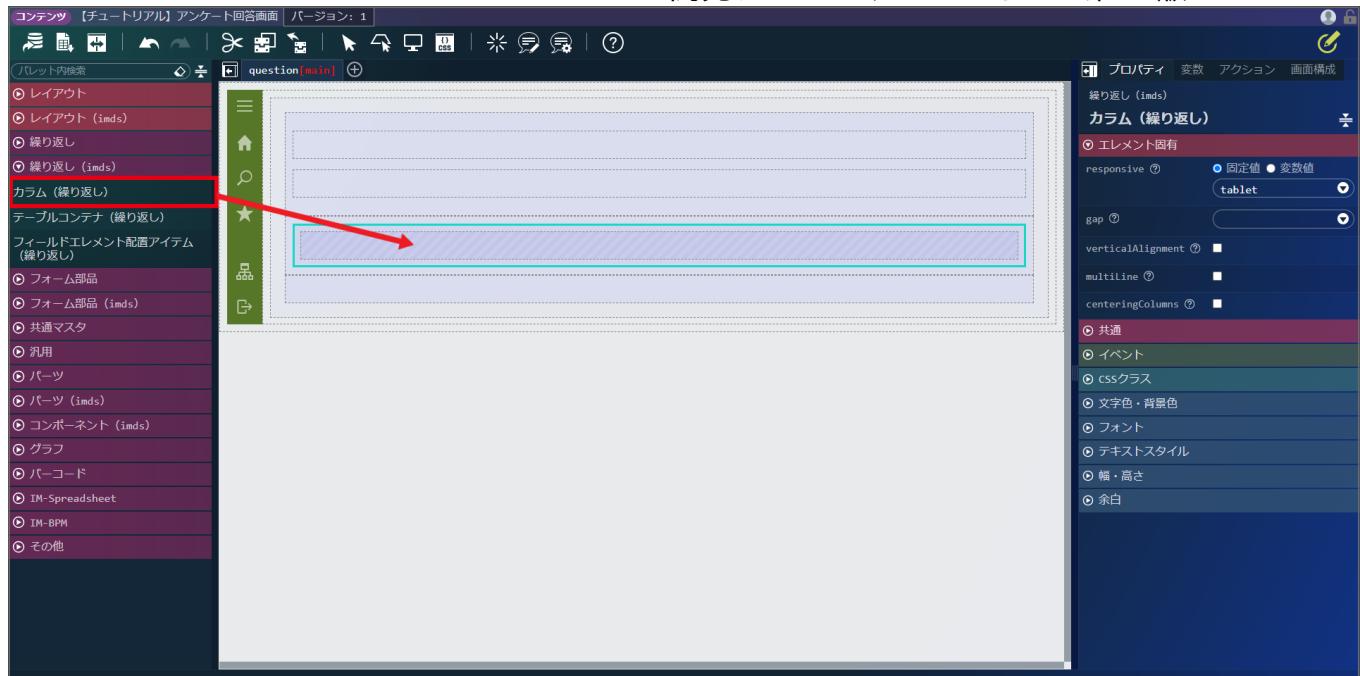
決定 取り消し

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」

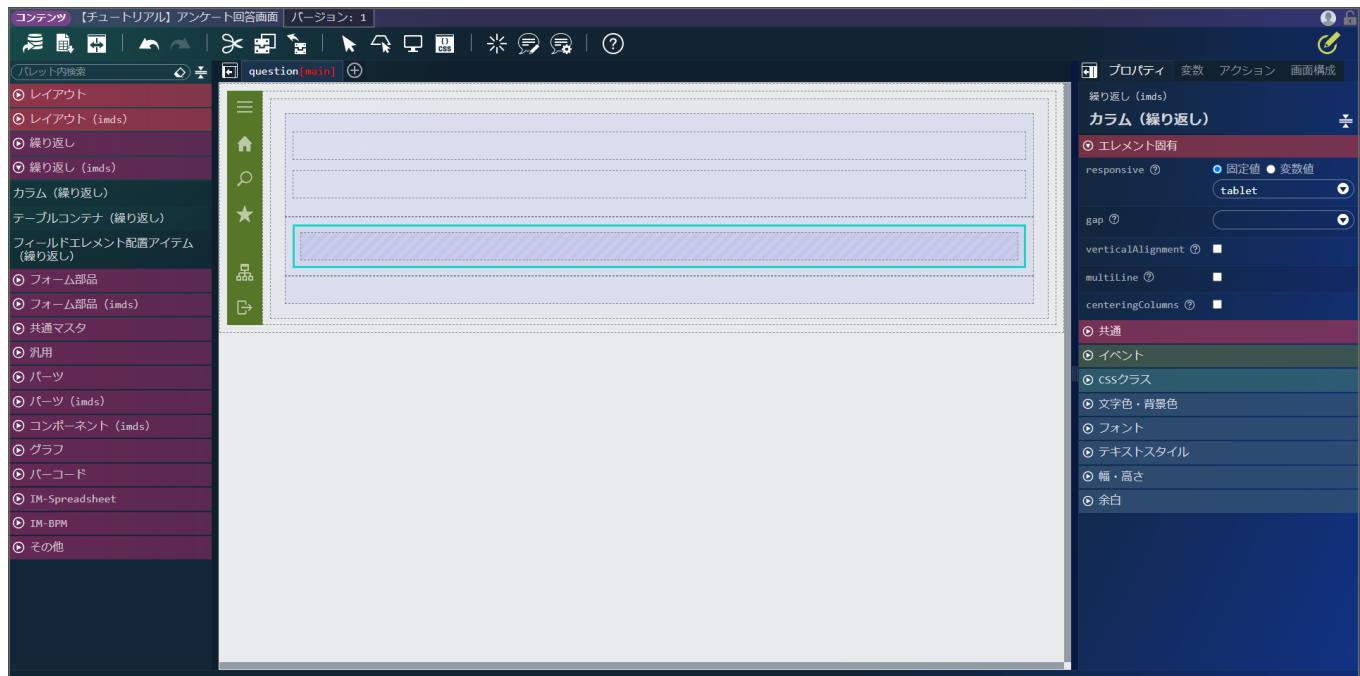
次に、設問の表示領域を作成します。

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

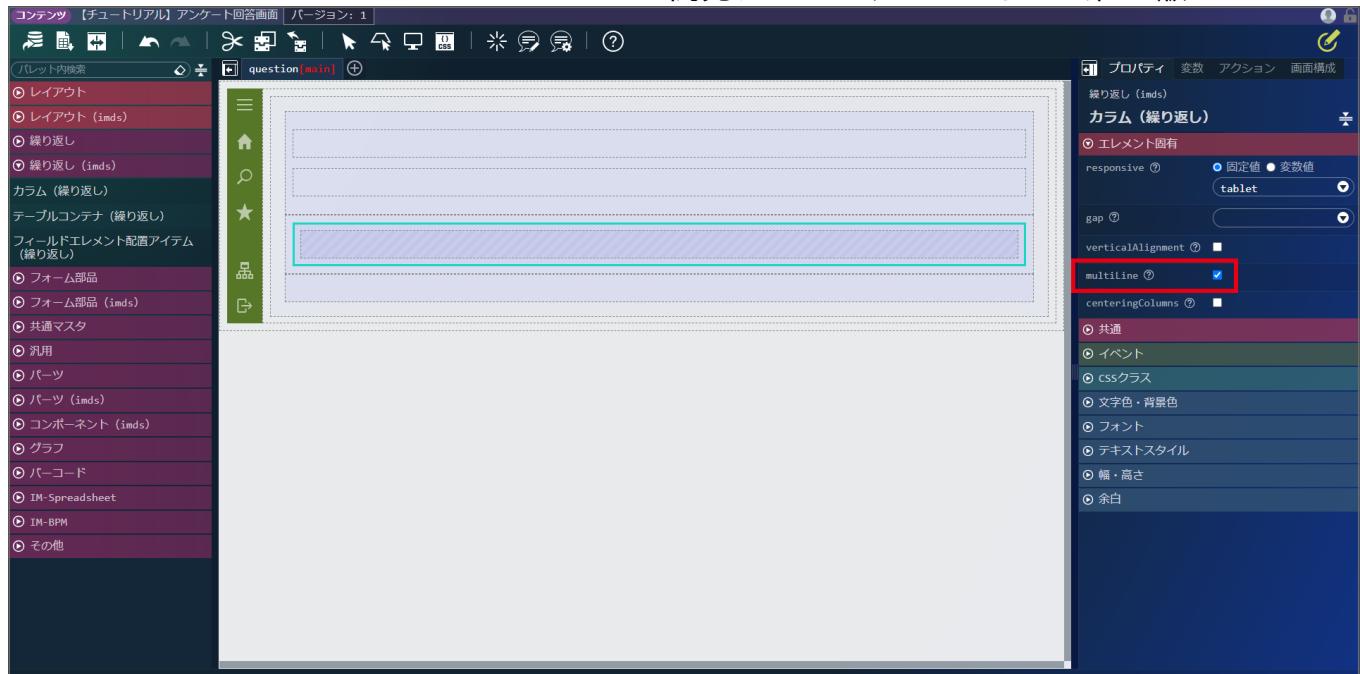
画面左側のパレットの「繰り返し (imds)」 - 「カラム（繰り返し）」を選択し、先ほど配置した「フレックスコンテナ」の中央の「フレックスアイテム」の中に配置します。



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。

変数・定数・入力・多言語・環境セレクタ

繰り返しエレメントを考慮 ②

| 変数 | 定数 | 入力 | 多言語 | 環境 |
|---|----|----|-----|----|
| キーワード検索 | | | | |
| <ul style="list-style-type: none"> ▼ \$variable <マップ> <マップ> ▼ responseData <マップ> <マップ> ▼ records <マップ> <配列> <ul style="list-style-type: none"> ▼ 0 <マップ> <マップ> enquete_id <文字列> <データなし> question_id <文字列> <データなし> question_no <浮動小数点数> <データなし> question <文字列> <データなし> left_scale <文字列> <データなし> right_scale <文字列> <データなし> enqueteTitle <文字列> <データなし> enqueteOverview <文字列> <データなし> ▼ entity <マップ> <マップ> enqueteId <文字列> <データなし> ▼ answersData <マップ> <マップ> enqueteId <文字列> <データなし> ▼ answers <文字列> <配列> errorMessage <文字列> <データなし> | | | | |
| <input type="button" value="決定"/> <input type="button" value="取り消し"/> | | | | |

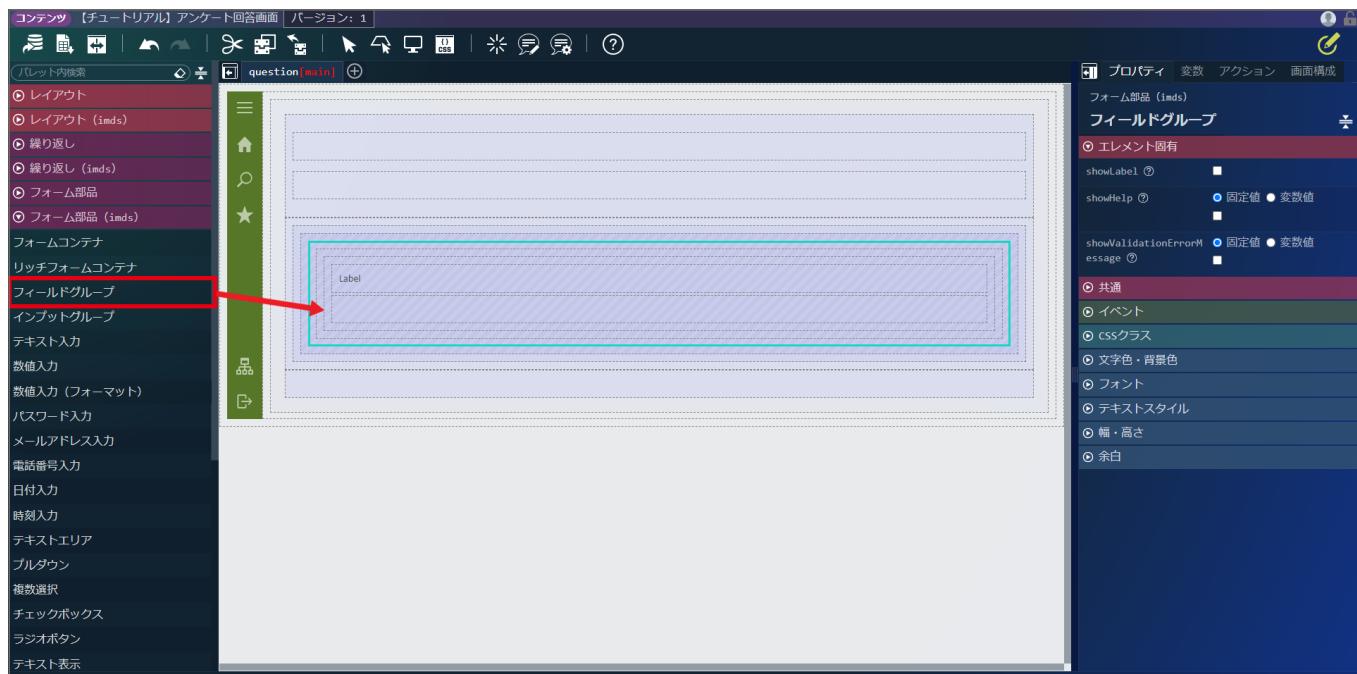
「カラムアイテム（繰り返し）」 - 「エレメント固有」 - 「size」 に「full」を設定します。

「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。



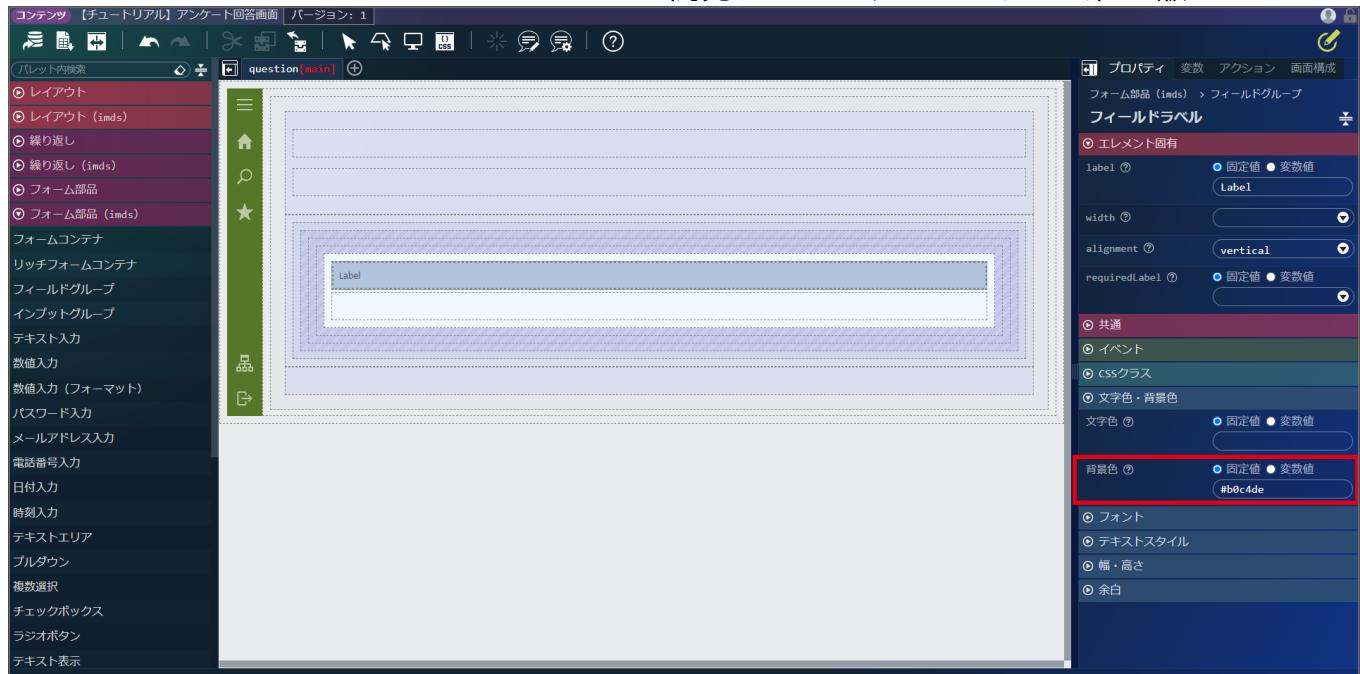
続けて、実際に回答内容を表示する領域を作成します。

画面左側のパレットの「フォーム部品 (imds)」 - 「フィールドグループ」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。

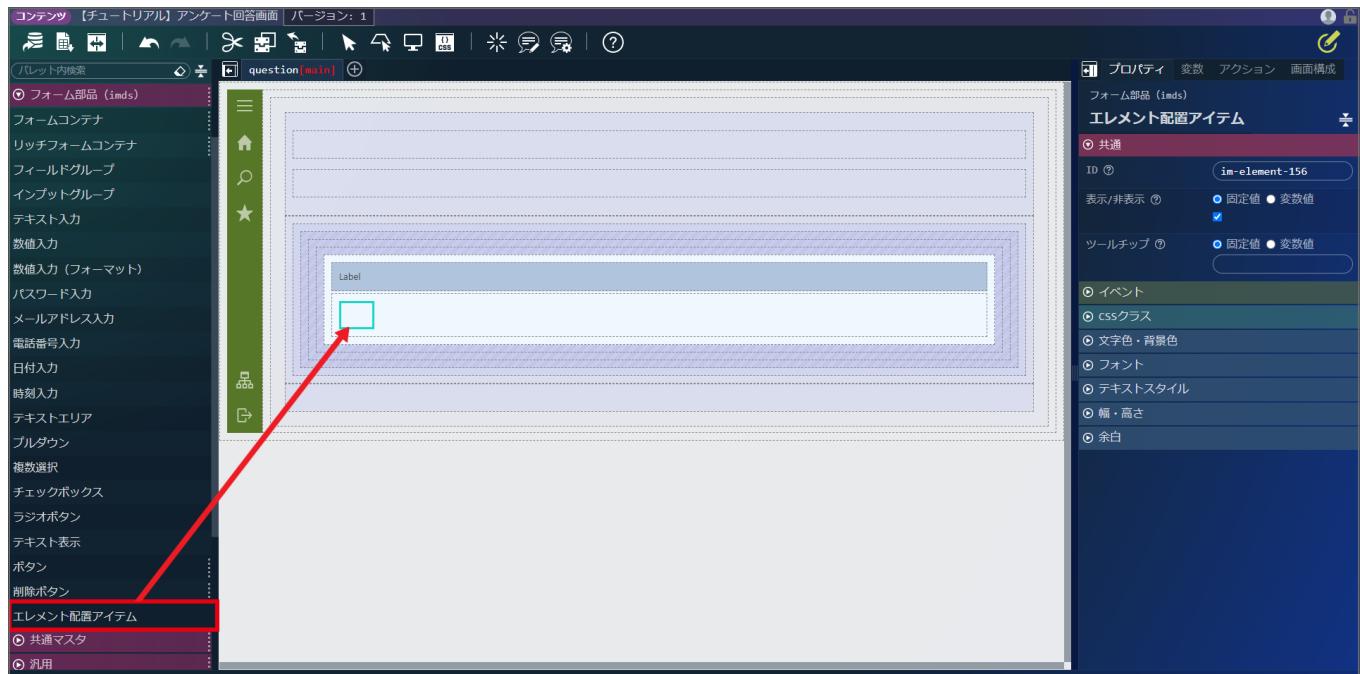


配置した「フォーム部品 (imds)」 - 「フィールドグループ」 - 「フィールド」を選択し、「文字色・背景色」の「背景色」に「#f0f8ff」を設定します。

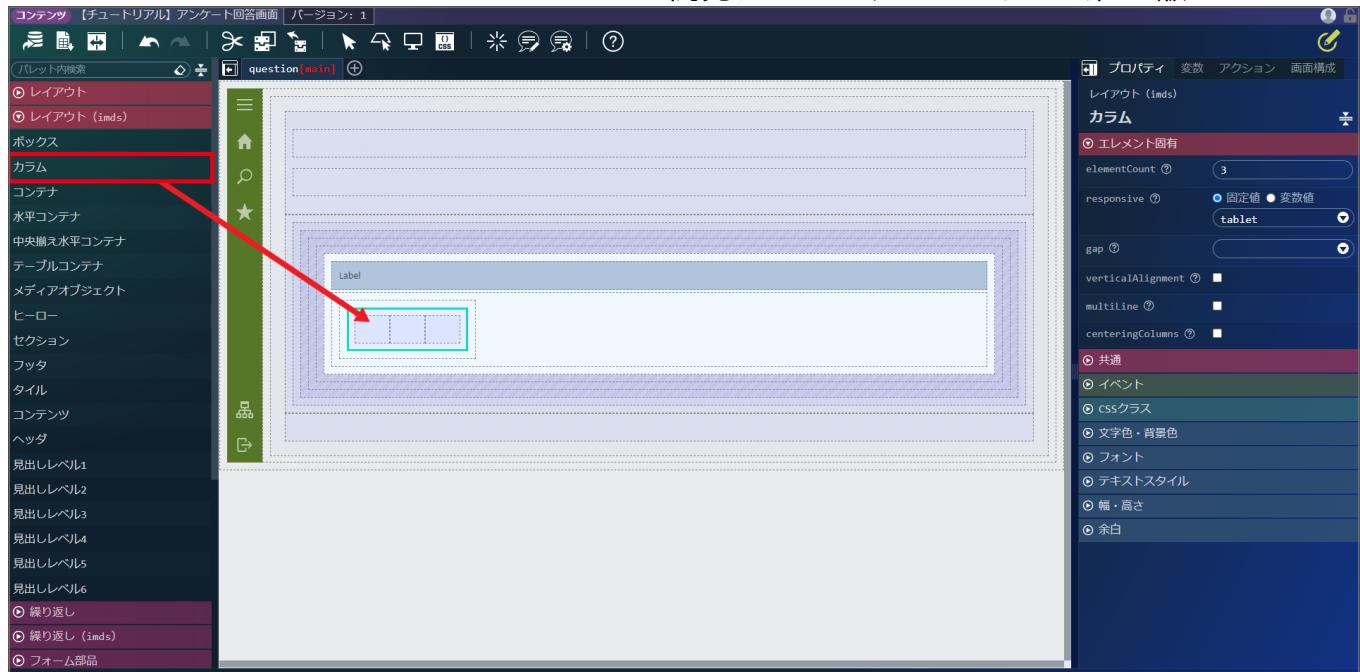
同様に、「フォーム部品 (imds)」 - 「フィールドグループ」 - 「フィールドラベル」の背景色を「#b0c4de」としてください。



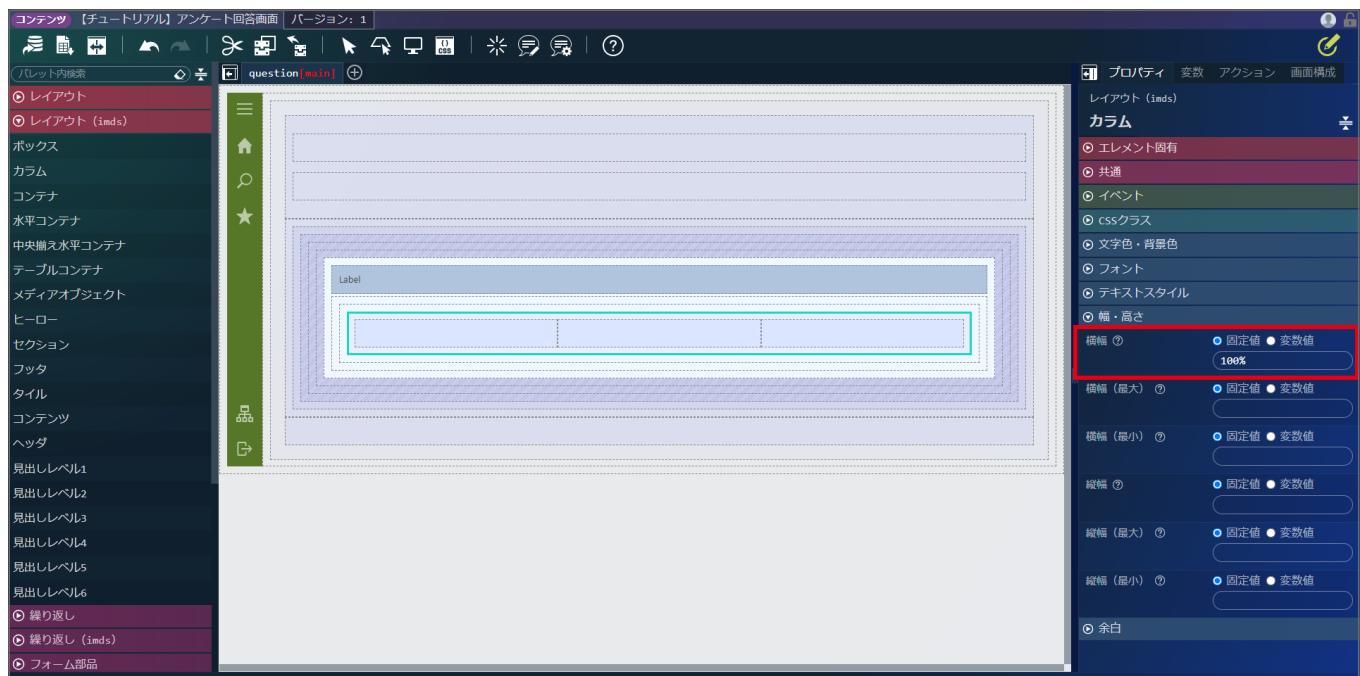
画面左側のパレットの「フォーム部品（imds）」 - 「エレメント配置アイテム」を選択し、「フィールドコンテンツ」の中に配置します。



画面左側のパレットの「レイアウト（imds）」 - 「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。

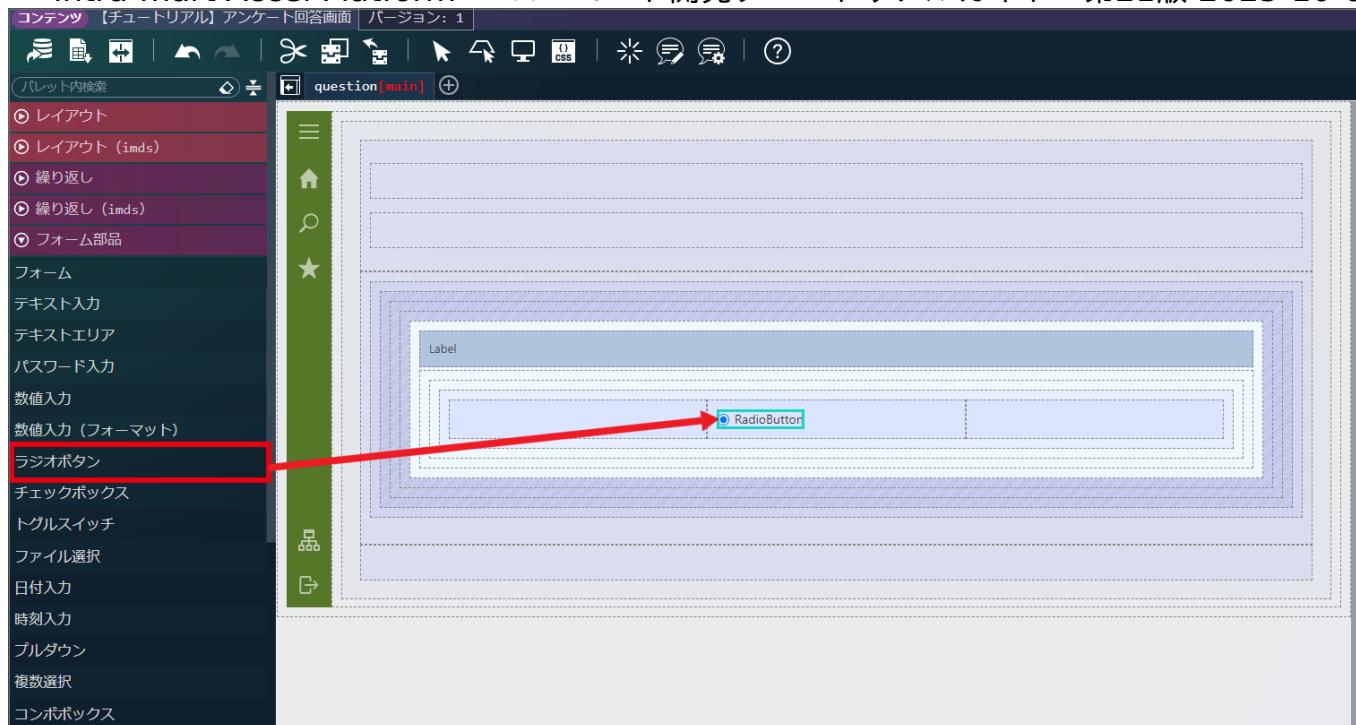


「エレメント配置アイテム」および「カラム」の「幅・高さ」 - 「横幅」を「100%」に設定します。



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。

| Property | Type | Value |
|---------------|-----------|-----------------------------|
| textContent ? | 固定値 ● 変数値 | 1 |
| disabled ? | 固定値 ● 変数値 | |
| autofocus ? | 固定値 ● 変数値 | |
| name ? | 固定値 ● 変数値 | |
| value ? | 固定値 ● 変数値 | 1 |
| selected ? | 固定値 ● 変数値 | \$variable.answers[\$index] |

「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]」

i コラム

「繰り返し」のエレメントを利用すると、配列の変数を利用して配列の数だけエレメントが表示されます。

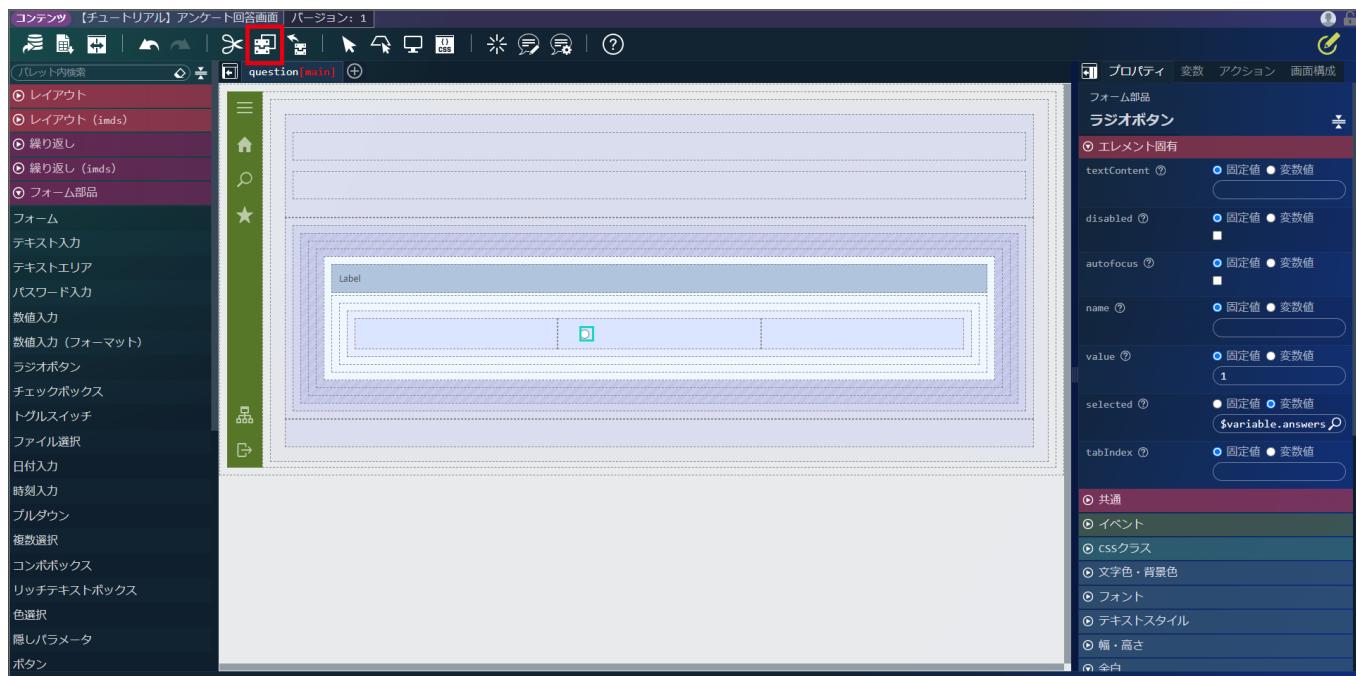
「繰り返し」のエレメントに配置した要素に対して変数を設定する場合、配列の添え字として「\$index」を指定することにより、繰り返しに合わせた変数値へと変換されます。

詳細については、以下のCookBookを参照してください。

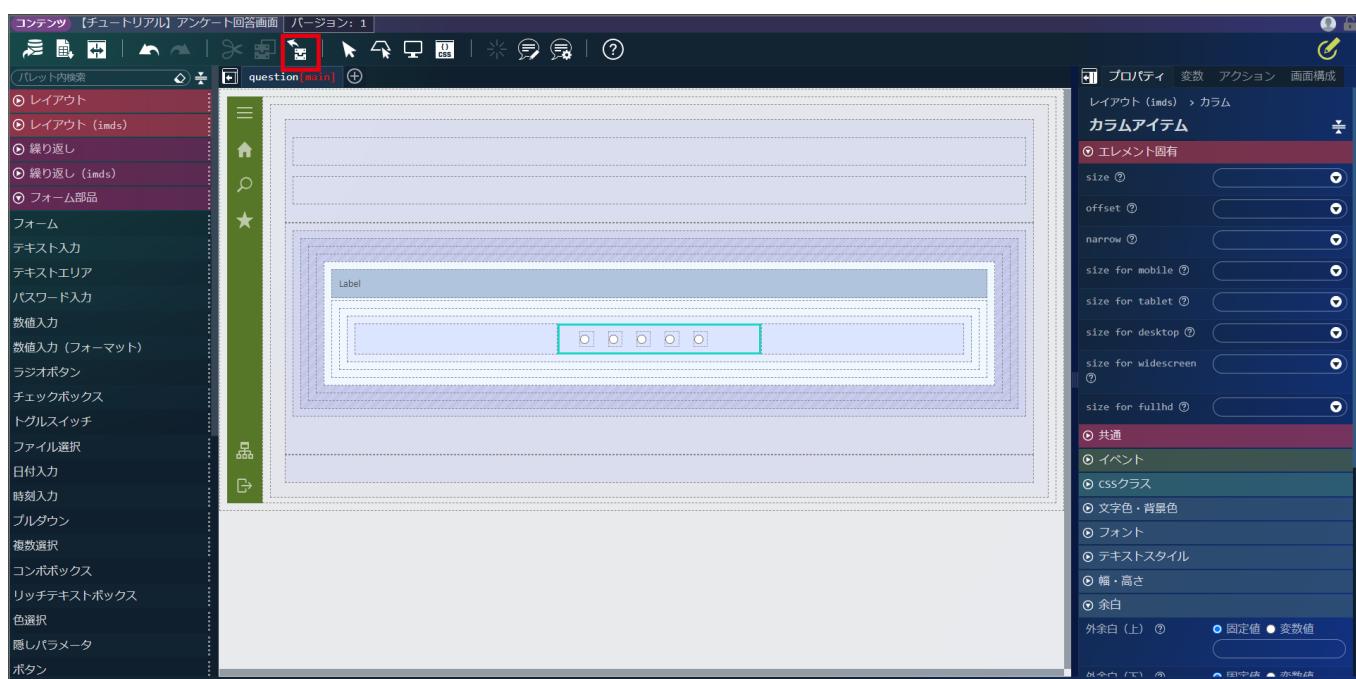
[「IM-BloomMaker 繰り返しエレメントでの変数の使い方」](#)

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  かキーボードのCtrl+Cでコピーします。

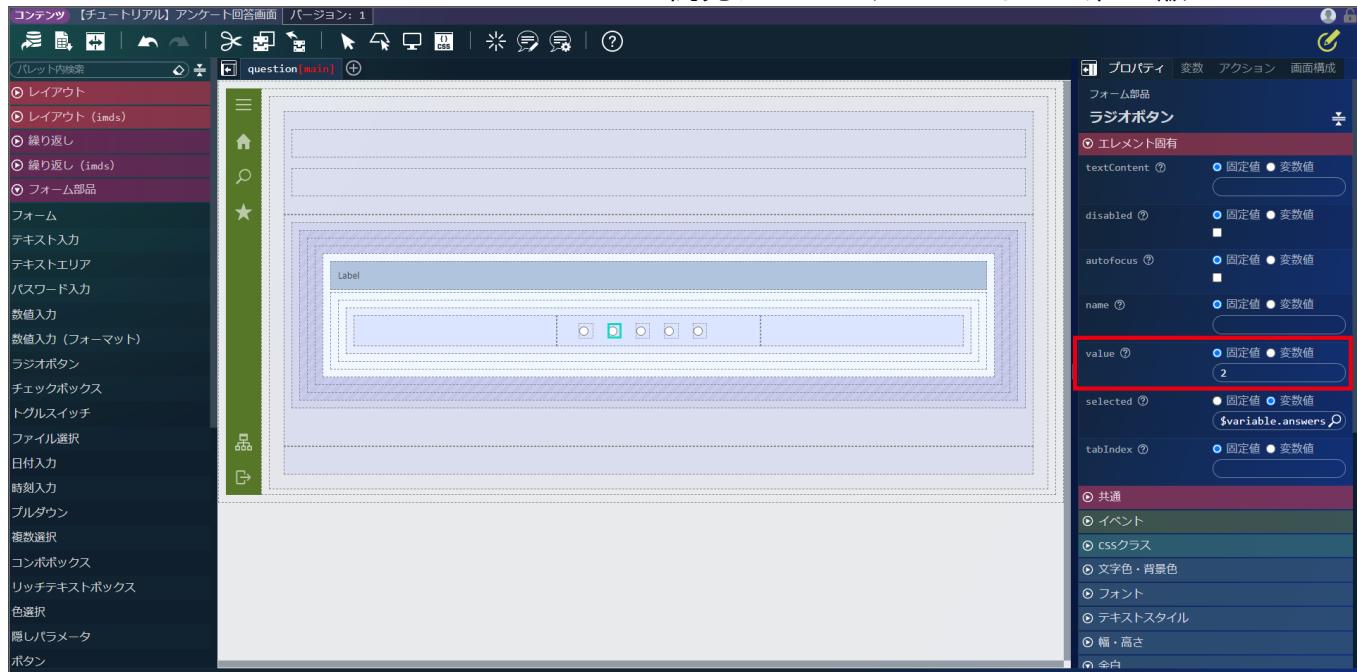


すでにラジオボタンが配置されているカラムアイテムをクリックし、画面上部の  かキーボードのCtrl+Vでラジオボタンが5つ並ぶまで貼り付けを繰り返します。



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」 - 「エレメント固有」 - 「value」が「1」、「2」、「3」、「4」、「5」の順になるよう修正します。

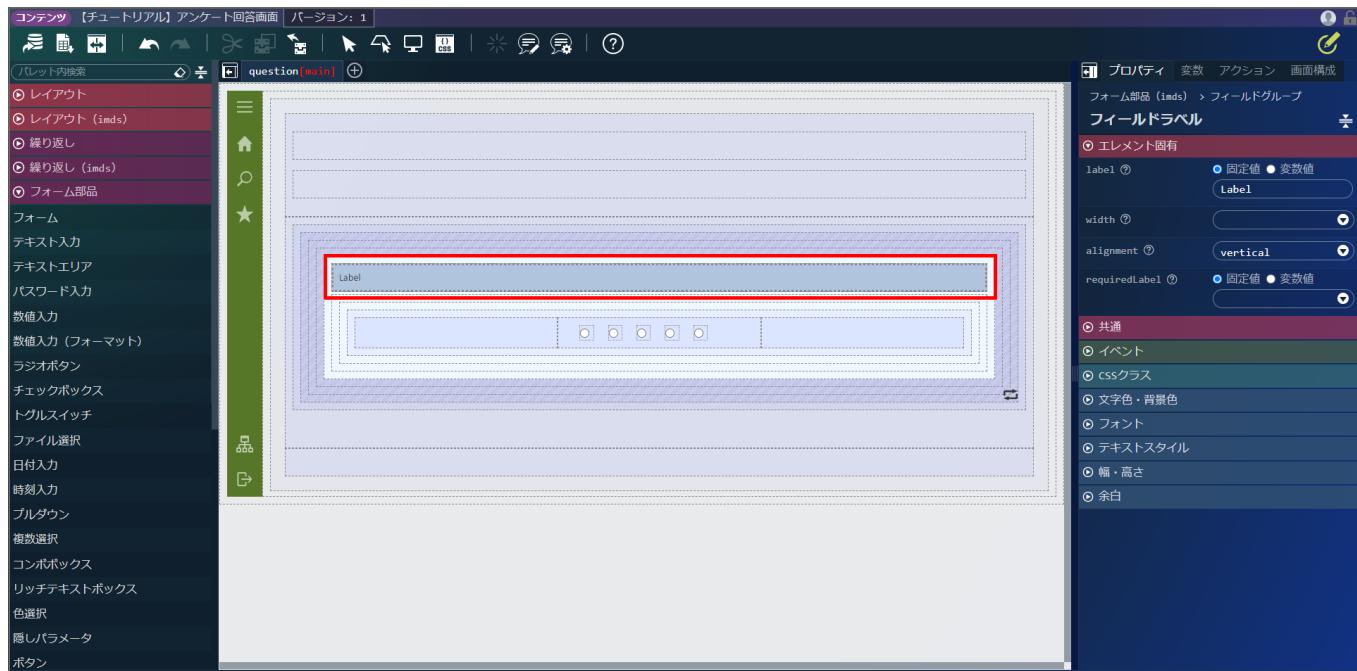
以下の画像は、2番目のラジオボタンを選択した場合の例です。



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使用する「フィールドグループ」 - 「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 - 「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。

変数・定数・入力・多言語・環境セレクタ



変数 定数 入力 多言語 環境

キーワード検索


 繰り返しエレメントを考慮 ②

| | |
|-----------------------|---------|
| ▼ \$variable <マップ> | <マップ> |
| ▼ responseData <マップ> | <マップ> |
| ▼ records <配列> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| enquete_id <文字列> | <データなし> |
| question_id <文字列> | <データなし> |
| question_no <浮動小数点数> | <データなし> |
| question <文字列> | <データなし> |
| left_scale <文字列> | <データなし> |
| right_scale <文字列> | <データなし> |
| enqueteTitle <文字列> | <データなし> |
| enqueteOverview <文字列> | <データなし> |
| ▼ entity <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answersData <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answers <配列> | <配列> |
| errorMessage <文字列> | <データなし> |

決定

取り消し

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。

変数 定数 入力 多言語 環境

キーワード検索

繰り返しエレメントを考慮 ②

| | |
|-----------------------|---------|
| ▼ \$variable <マップ> | <マップ> |
| ▼ responseData <マップ> | <マップ> |
| ▼ records <配列> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| enquete_id <文字列> | <データなし> |
| question_id <文字列> | <データなし> |
| question_no <浮動小数点数> | <データなし> |
| question <文字列> | <データなし> |
| left_scale <文字列> | <データなし> |
| right_scale <文字列> | <データなし> |
| enqueteTitle <文字列> | <データなし> |
| enqueteOverview <文字列> | <データなし> |
| ▼ entity <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answersData <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answers <配列> | <配列> |
| errorMessage <文字列> | <データなし> |

決定 取消し

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。

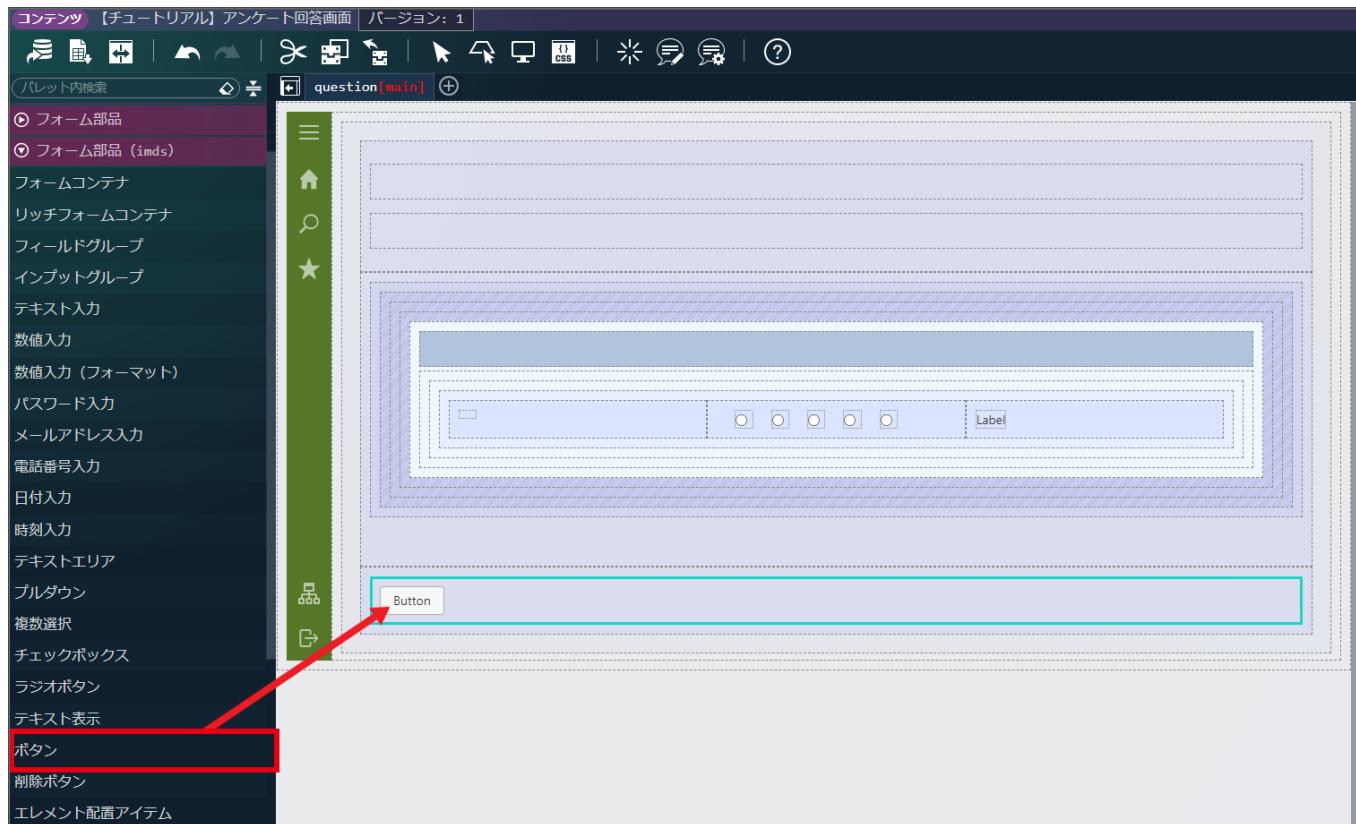
変数の設定では、textContentの入力ボックスに直接下記の文字列を入力してもかまいません。

「\$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。

「フレックスコンテナ」の一番下の「フレックスアイテム」の中に、画面左側のパレットの「フォーム部品 (imds)」 - 「ボタン」を配置します。



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。



コラム

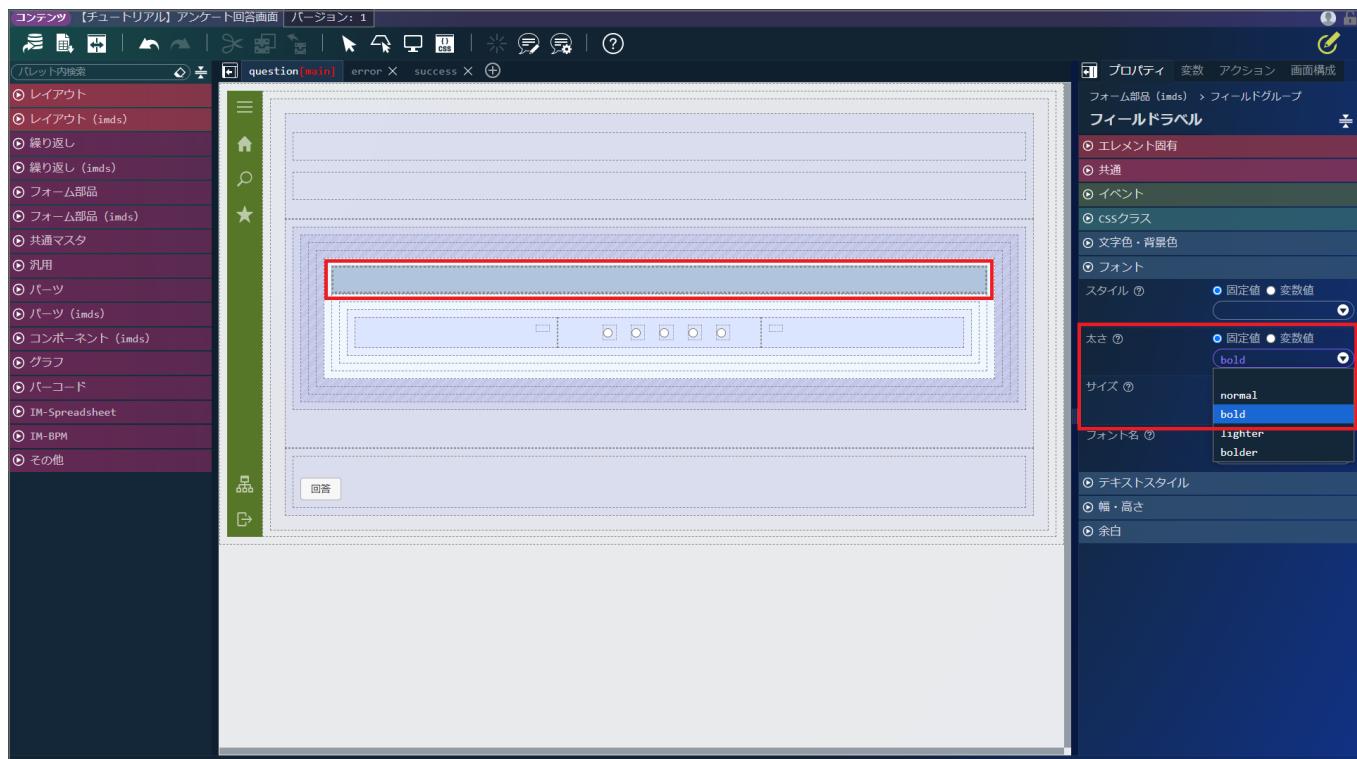
配置したいアイテムが見つからない場合は、パレット上部にある「パレット内検索」から検索できます。
入力内容に合わせ、アイテムが絞り込まれて表示されます。

それぞれの項目が見やすくなるように、見た目の調整を行いましょう。

「フレックスアイテム」内に配置した「見出しレベル1」を選択します。

「プロパティ」タブ - 「見出しレベル1」 - 「fontweight」を「bold」に設定してください。

同様に、「フィールドグループ」を選択し、「プロパティ」タブ - 「フィールドラベル」 - 「フォント」 - 「太さ」を「bold」に設定します。

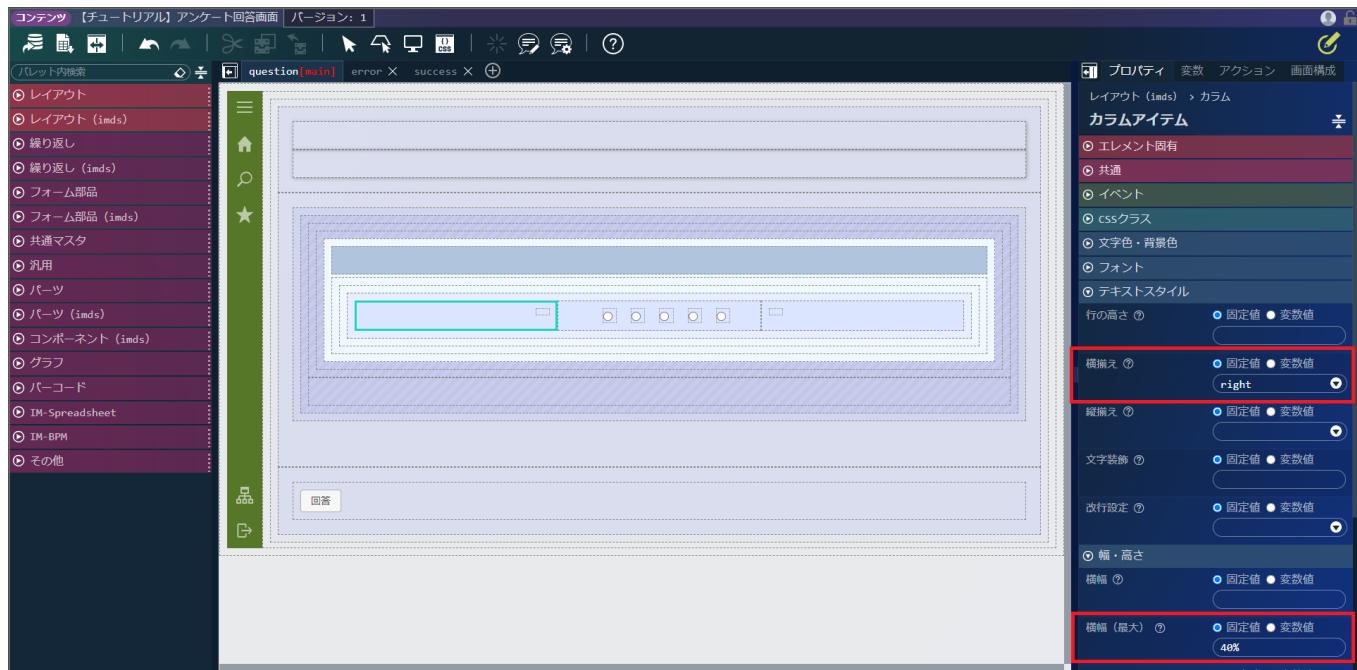


「カラム」内の左端のカラムアイテムを選択します。

「プロパティ」タブ - 「カラムアイテム」 - 「テキストスタイル」 - 「横揃え」を「right」と設定してください。

「プロパティ」タブ - 「カラムアイテム」 - 「幅・高さ」 - 「横幅（最大）」に「40%」と設定します。

「カラム」内の右端のカラムアイテムも同様に、「テキストスタイル」 - 「横揃え」は「left」とし、「幅・高さ」 - 「横幅（最大）」を「40%」と設定してください。



続けて、モバイル端末で表示した場合の調整をしましょう。

「プロパティ」タブ - 「カラム」 - 「エレメント固有」 - 「responsive」に「mobile」を設定してください。

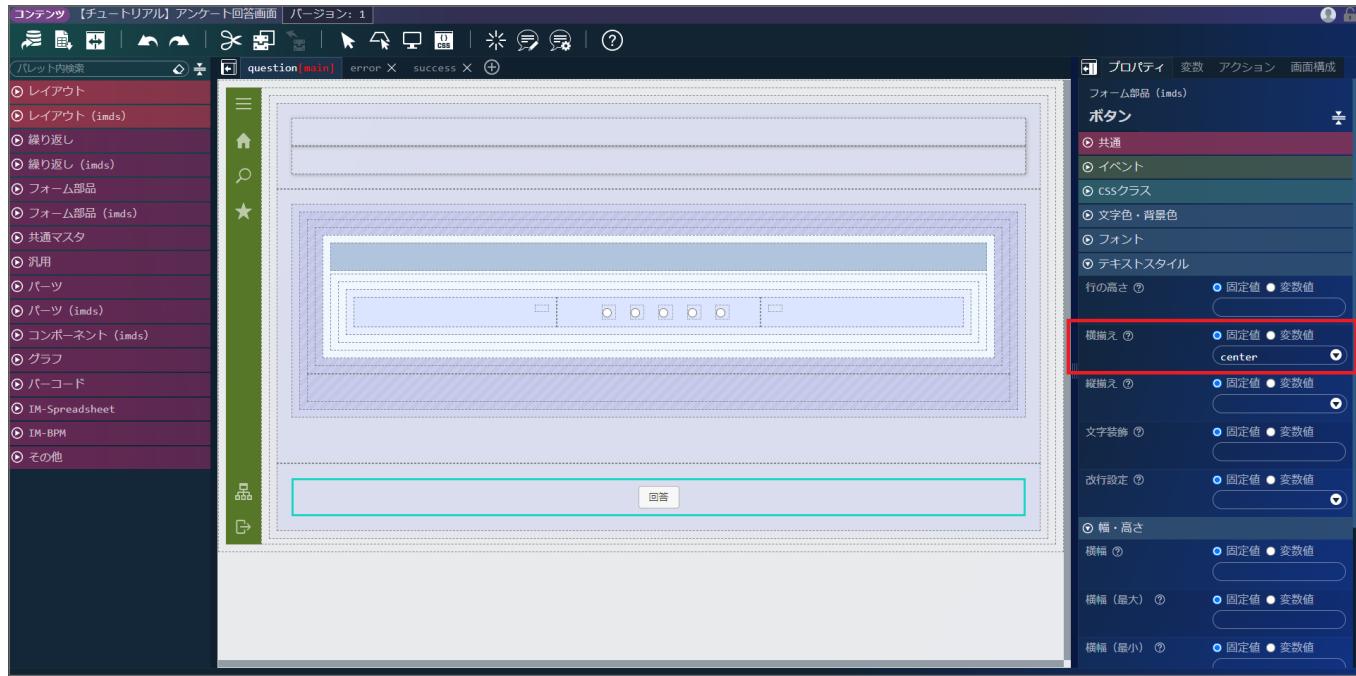
「カラム」内の中央のカラムアイテムの「エレメント固有」 - 「narrow」を「narrow mobile」とし、「テキストスタイル」 - 「横揃え」に「center」を設定します。

次に、登録ボタンの配置を変更します。

一番下の「フレックスアイテム」に配置した「ボタン」を選択します。

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

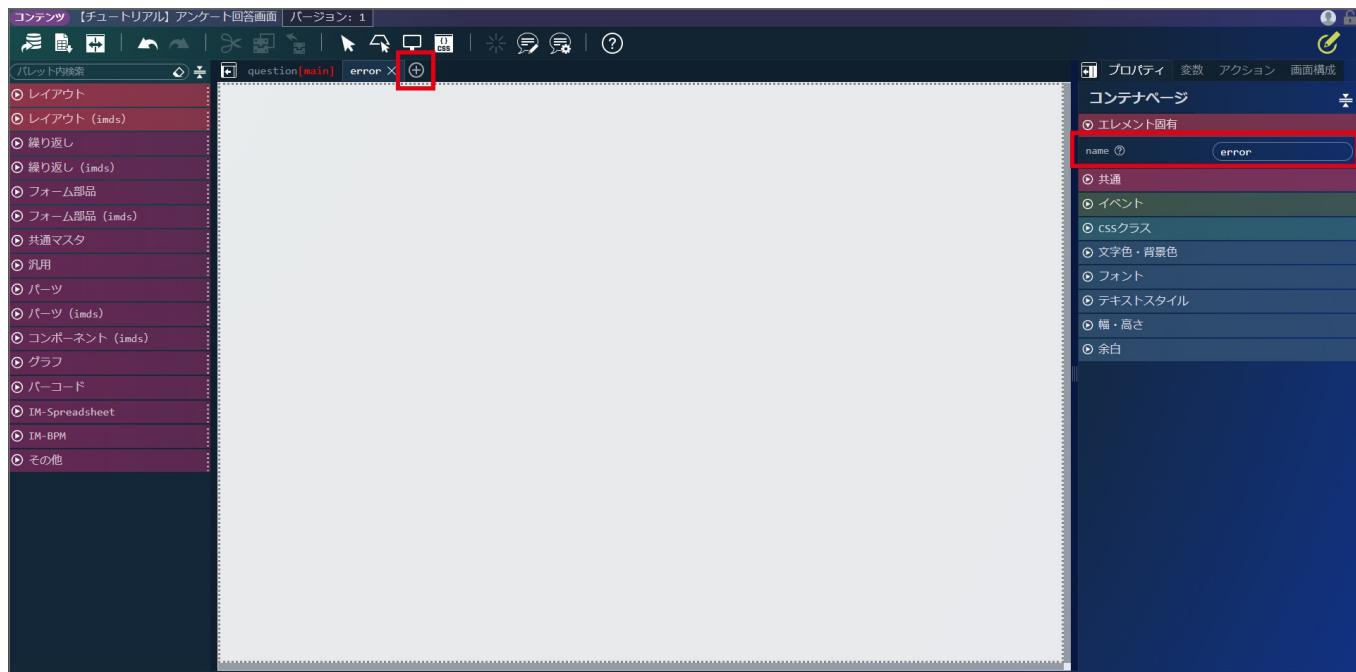
「余白」 - 「内余白（上）」に「3rem」と入力してください。



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。

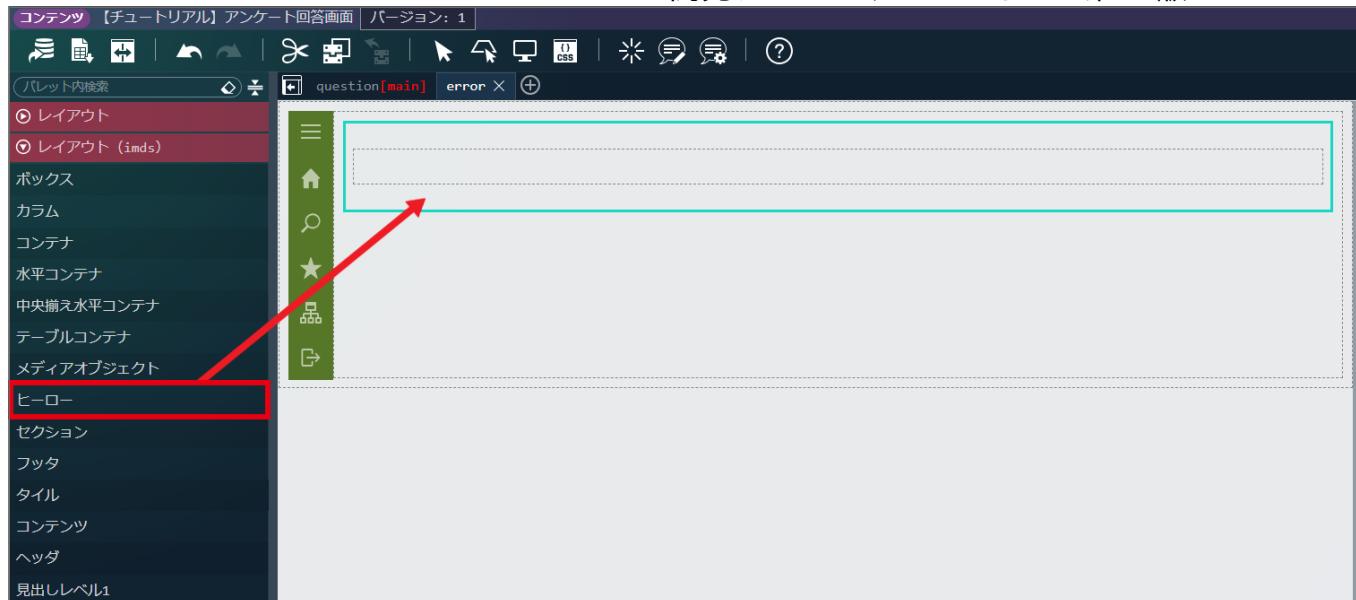
画面上部の をクリックし、「プロパティ」タブ - 「コンテナページ」 - 「エレメント固有」 - 「name」を「error」と設定します。



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



画面左側のパレットの「レイアウト (imds)」 - 「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。

The screenshot shows the '変数・定数・入力・多言語・環境セレクタ' dialog. The '変数' tab is selected. The search bar contains '\$variable'. The results list shows several variables, with 'errorMessage' highlighted in yellow at the bottom. The dialog includes tabs for '定数', '入力', '多言語', and '環境'. At the bottom are '決定' (Decision) and '取り消し' (Cancel) buttons.

| 変数 | 説明 |
|-----------------------|---------|
| \$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> | <データなし> |
| question_id <文字列> | <データなし> |
| question_no <浮動小数点数> | <データなし> |
| question <文字列> | <データなし> |
| left_scale <文字列> | <データなし> |
| right_scale <文字列> | <データなし> |
| enqueteTitle <文字列> | <データなし> |
| enqueteOverview <文字列> | <データなし> |
| ▼ entity <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answersData <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| ▼ answers <文字列> | <配列> |
| errorMessage <文字列> | <データなし> |

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 - 「コンテナページ」 - 「エレメント固有」 - 「name」を「success」と設定します。

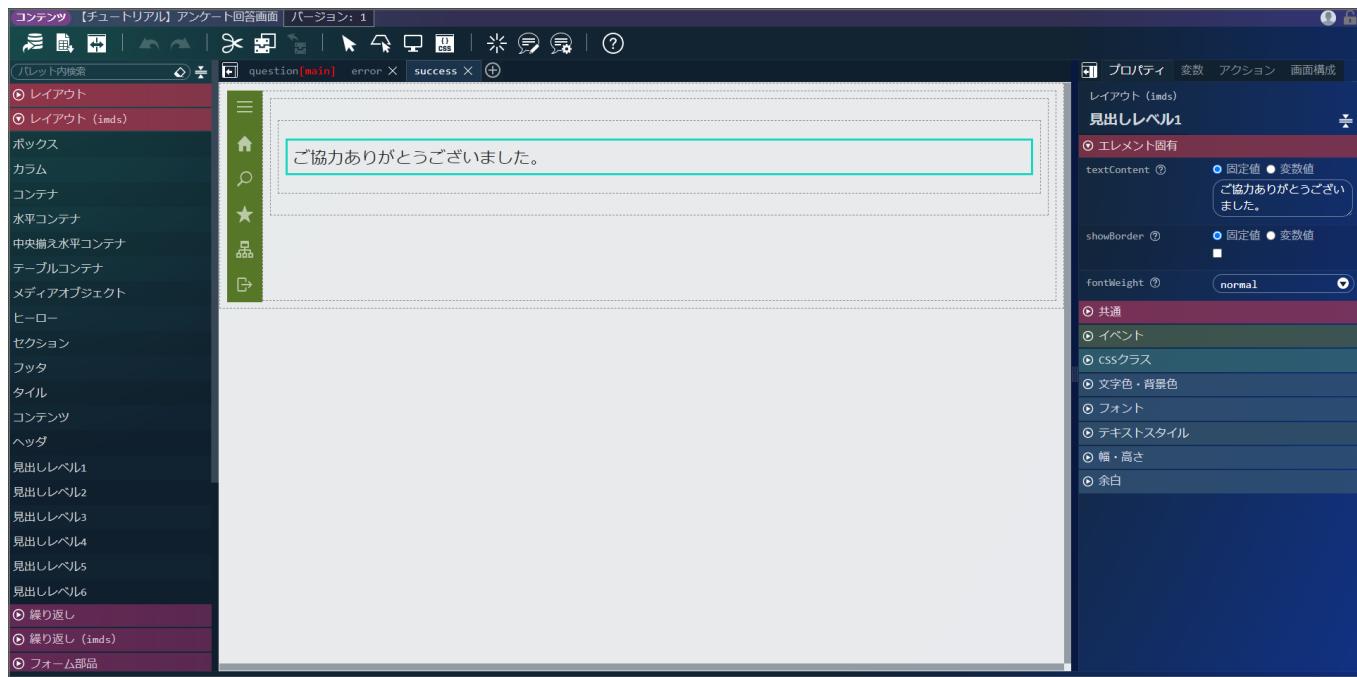
画面左側のパレットの「レイアウト」 - 「サイドメニュー・コンテナ」を選択し、画面に配置します。

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 (imds)」 - 「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 (imds)」 - 「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

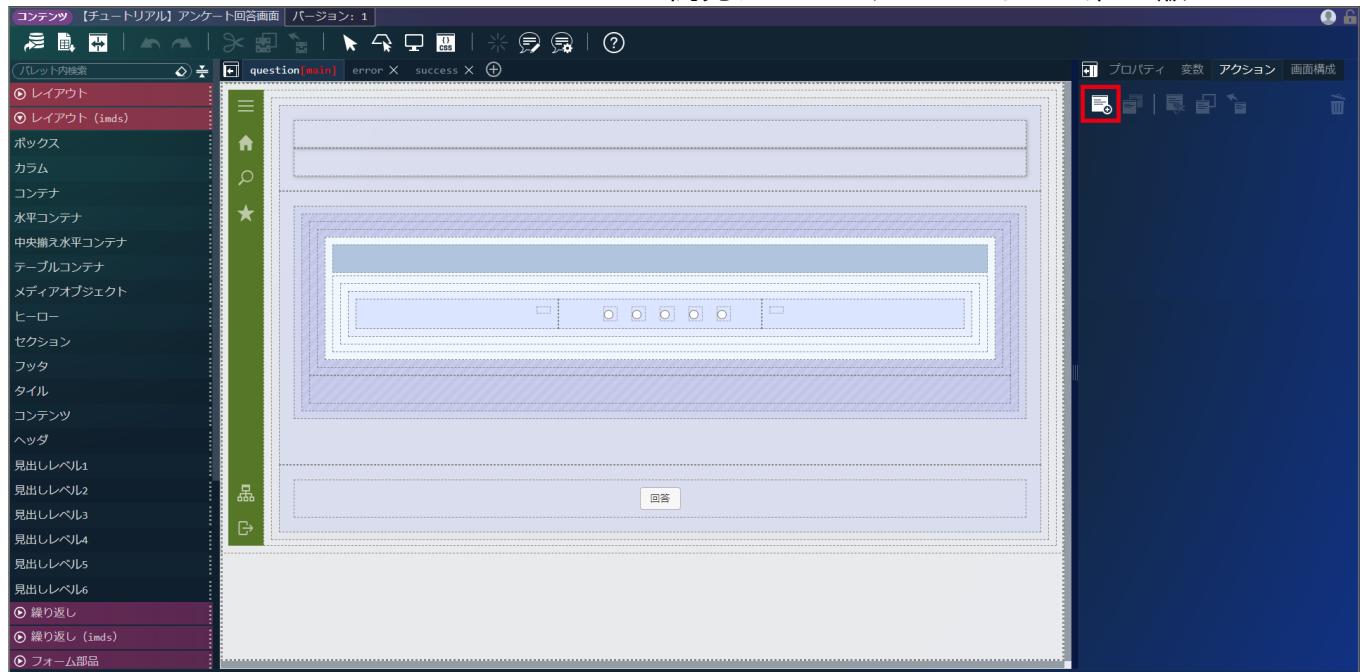
次に、画面で実行されるアクションを設定します。

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。

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。

| 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 |
|--|------|--------------|------------|
| アクションを検索 | | | |
| <input checked="" type="radio"/> 標準 | | | |
| <input type="radio"/> 共通マスター | | | |
| <input type="radio"/> IM-LogicDesigner | | | |
| <input type="radio"/> ViewCreator | | | |
| <input type="radio"/> Bulma | | | |
| <input type="radio"/> IM-Spreadsheet | | | |
| <input type="radio"/> ワークフロー | | | |

ここにドロップしてください

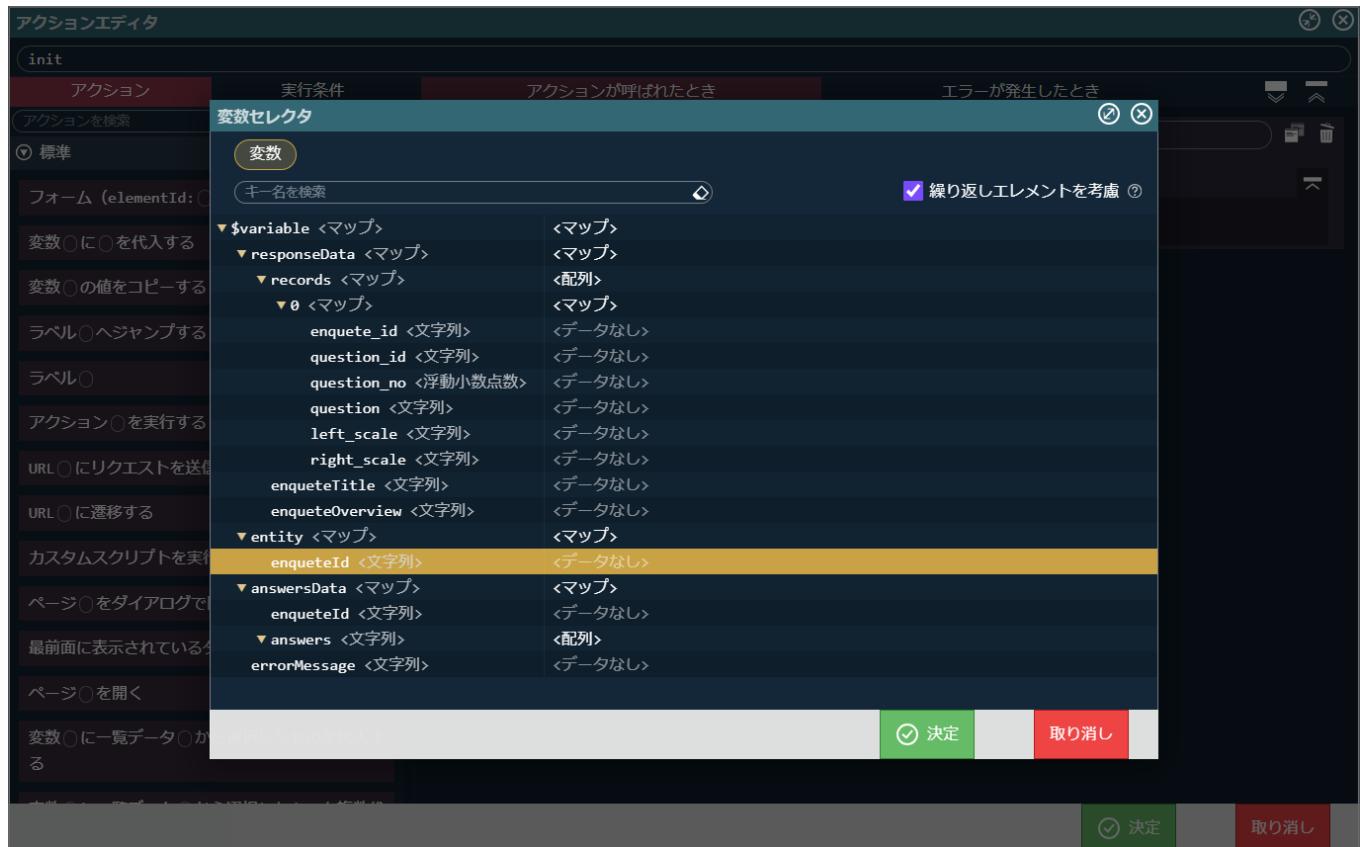
決定 取り消し

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○に○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。

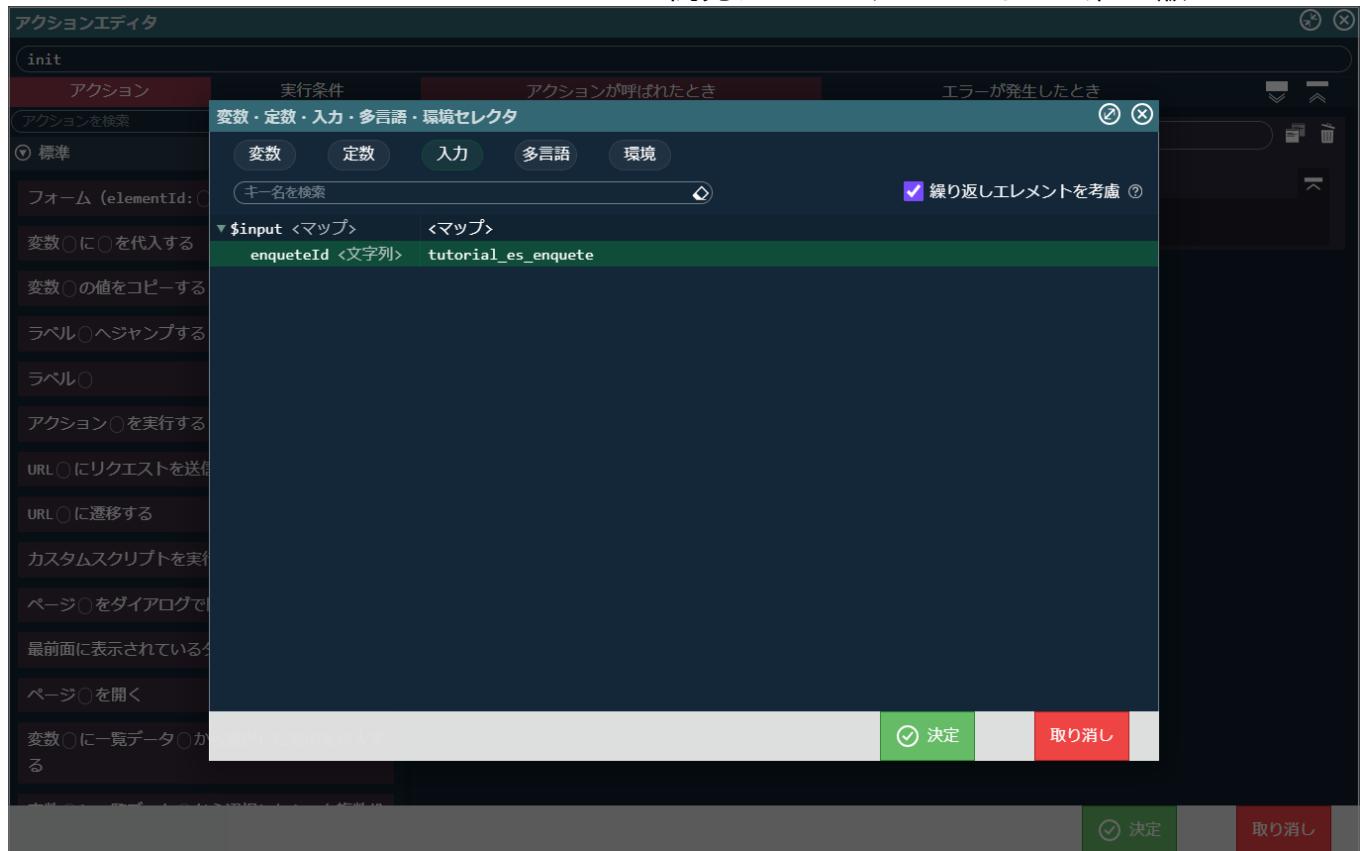


配置したアクションの変数の から「変数 > \$variable > entity > enquetetId」を選択し、「決定」をクリックします。



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 > enquetetId」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○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。

IM-LogicDesigner フローリーティング定義検索

| 選択 | ルーティング | メソッド | フロー定義ID | フロー定義名 |
|-------------------------------------|------------------------------|------|------------------------|--------------------|
| ■ | tutorial_enquete/answer | POST | tutorial_answer_insert | 【チュートリアル】アンケート回答登録 |
| ■ | 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 checked="" 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。

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

1 変数 \${} \$variable.entity  に \${} + \$input.enquete  を代入する

アクションアイテム説明

2 IM-LogicDesigner フローリーティング `tutorial_enquete/question`  にリクエストを送信する

リクエストパラメータ IN \${} \$variable.entity 

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 \${} \$variable.responseData 

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 ■ 

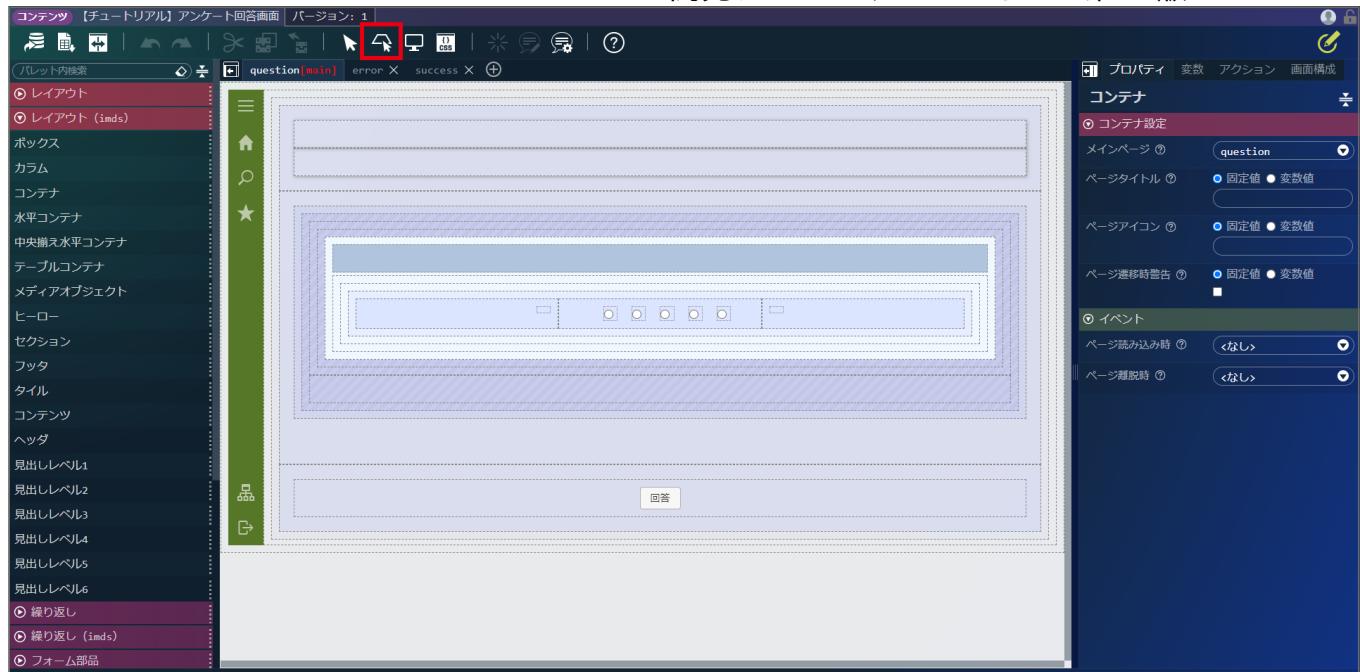
アカウントコンテキストのタイムゾーンを強制的に使用する ■ 

常に実行

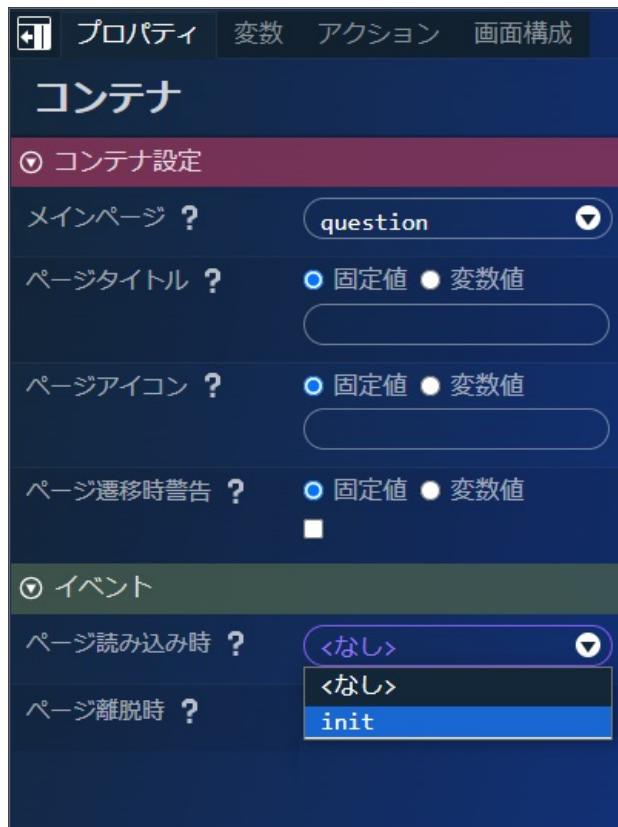
ここまでで、アンケート情報を取得し画面に表示できるようになりました。一度、動作確認を行いましょう。
アクションエディタ下部の決定ボタンをクリックし、ここまで作成したアクション処理を登録します。

「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。



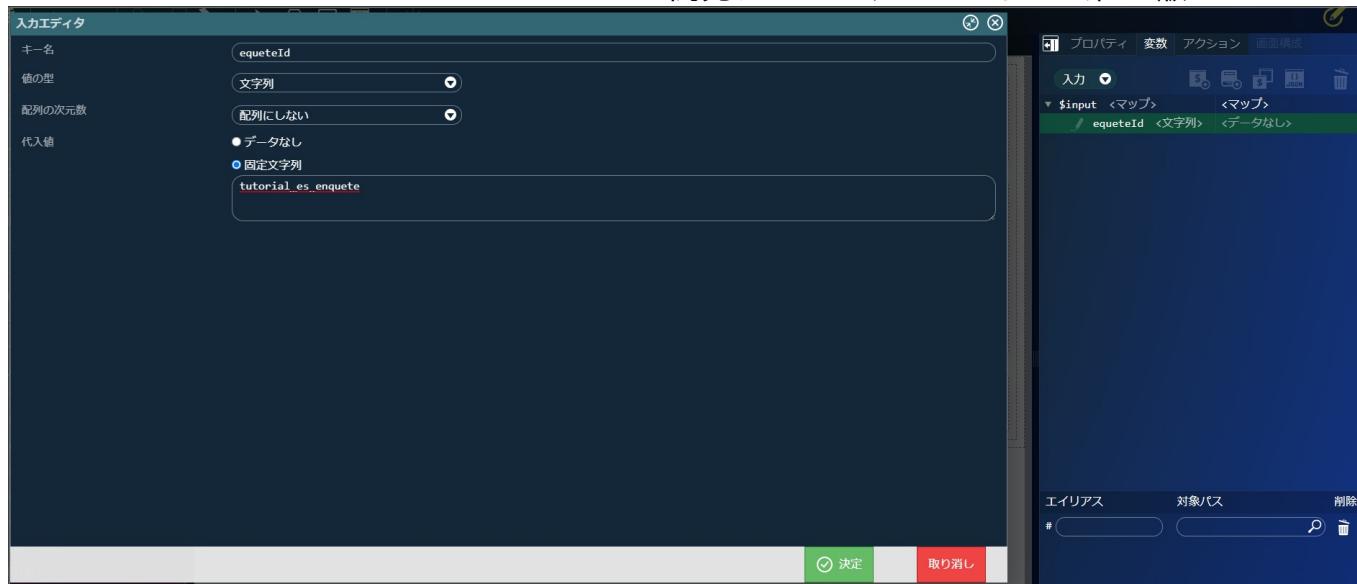
画面右側の「プロパティ」タブ - 「コンテナ」 - 「イベント」の「ページ読み込み時」に「init」アクションを設定します。



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。

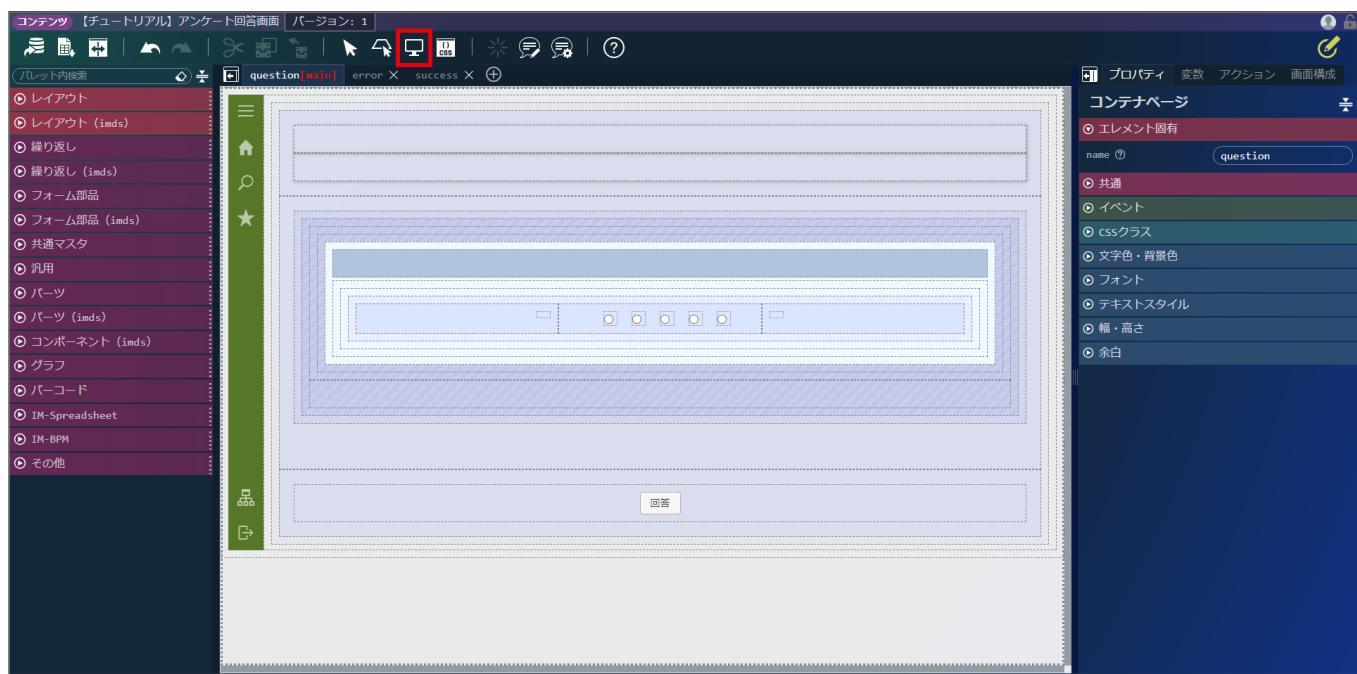
「\$input > enqueteld」の をクリックし、代入値のラジオボタンを「固定文字列」に切り替えて「tutorial_es_enquete」と設定します。

※ 4.4.2.2.3. サンプルデータの読み込み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。

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

を感じていない ○ ○ ○ ○ ○ 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 ○ ○ ○ ○ ○ 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 ○ ○ ○ ○ ○ そう思う

社内の人間関係が良好である

そうは思わない ○ ○ ○ ○ ○ 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 ○ ○ ○ ○ ○ 充実している

回答

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 フロールーティング○にリクエストを送信する」の次に配置します。

アクションエディタ

init

アクション 実行条件

アクションを検索

標準

フォーム (elementId:○) を送信する

変数○に○を代入する

変数○の値をコピーする

ラベル○へジャンプする

ラベル○

アクション○を実行する

URL○にリクエストを送信する

URL○に遷移する

カスタムスクリプトを実行する

ページ○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○を開く

変数○に一覧データ○から選択したものを代入する

変数○に一覧データ○から選択したものを複数代入する

メッセージ○をアラートダイアログで表示する

アクションが呼ばれたとき エラーが発生したとき

1 変数 \${} \$variable.entity に \${} \$input.enquete を代入する

2 IM-LogicDesigner フローリーティング (tutorial_enquete/question) にリクエストを送信する

3 カスタムスクリプトを実行する

常に実行

決定 取り消し

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.enquetetd) に (\$variable.entity.enquetetd) を代入する

The screenshot shows the IM-LogicDesigner Logic Editor interface. On the left, there is a sidebar titled 'アクションエディタ' (Action Editor) with a 'answer' tab selected. Below it is a list of actions under the '標準' (Standard) category, including: 'フォーム (elementId:○) を送信する', '変数○に○を代入する', '変数○の値をコピーする', 'ラベル○へジャンプする', 'ラベル○', 'アクション○を実行する', 'URL○にリクエストを送信する', 'URL○に遷移する', 'カスタムスクリプトを実行する', 'ページ○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○を開く', '変数○に一覧データ○から選択したものを代入する', '変数○に一覧データ○から選択したものを複数代入する', and 'メッセージ○をアラートダイアログで表示する'. The main workspace shows a configuration for a 'アクションが呼ばれたとき' (Action Called) event. It specifies an action '変数 \${} \$variable.answer' to be assigned to variable '\$variable.entity' with the value '\$variable.answersData'. This configuration is set to run '常に実行' (Always). At the bottom right are '決定' (Decision) and '取り消し' (Cancel) buttons.

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○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。

The screenshot shows the 'IM-LogicDesigner フロールーティング定義検索' (IM-LogicDesigner Routing Definition Search) dialog. The search term '[チュートリアル]' is entered. The results table lists three entries:

| 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 |

At the bottom, there is a page navigation bar showing '1 ページ中 (1 / 58)' and a message '3 件中 1 - 3 を表示'.

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。

The screenshot shows the Action Editor interface with two actions being configured:

- Action 1: アクションが呼ばれたとき**
- Action 2: エラーが発生したとき**

Action 1 Configuration:

- アクション:** IM-LogicDesigner フローリーティング (tutorial_enquete/answer) にリクエストを送信する
- 実行条件:** 常に実行
- 説明:** 変数 \${variable.answer} に \${variable.entity} を代入する

Action 2 Configuration:

- アクション:** IM-LogicDesigner フローリーティング (tutorial_enquete/answer) にリクエストを送信する
- 実行条件:** 常に実行
- 説明:** 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 ■, アカウントコンテキストのタイムゾーンを強制的に使用する ■

Action Editor Buttons:

- 決定 (Green)
- 取り消し (Red)

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

The screenshot shows the Action Editor interface with three actions defined:

- Action 1:** Triggered when the action is called. It sets the variable `$variable.answer` to `$variable.answers[0].entityId`.
- Action 2:** Triggered when the action is called. It sends a request to the `IM-LogicDesigner` role-timing `tutorial_enquete/answer` with the URL `tutorial_enquete/answer`. The request parameters are `$variable.answersData`.
- Action 3:** Triggered when the action is called. It opens the success page.

Buttons at the bottom right: 決定 (Decision) and 取消 (Cancel).

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。

The screenshot shows the 'Button Element' configuration screen in the intra-mart Accel Platform. The top navigation bar includes tabs for 'プロパティ' (Properties), '変数' (Variables), 'アクション' (Actions), and '画面構成' (Page Configuration). The current view is under the 'Actions' tab, specifically for the 'ボタン' (Button) section.

ボタン要素

- ◎ エレメント固有
- ◎ 共通
- ◎ イベント

Event settings:

- クリック時: answer (selected)
- 二度押し防止: <なし>
- ダブルクリック時: answer (selected)
- キー押下時: <なし>
- フォーカスイン: <なし>
- フォーカスアウト: <なし>

Other configuration sections:

- ◎ CSSクラス
- ◎ 文字色・背景色
- ◎ フォント
- ◎ テキストスタイル
- ◎ 幅・高さ
- ◎ 余白

エラー処理の設定

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通りに対してエラー処理を作成します。

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。

アクションエディタ

✖️ ✎

The screenshot shows the Action Editor interface with the 'init' tab selected. On the left, a sidebar lists various action types: Standard, Common Master, IM-LogicDesigner, ViewCreator, imds, IM-Spreadsheet, and WorkFlow. The 'Standard' action is currently selected. The main workspace displays three actions:

- Action 1: 'アクションが呼ばれたとき' (When action is called) - '変数 \${variable.entity} に \${input.enquete} を代入する' (Set variable \${variable.entity} to \${input.enquete}).
- Action 2: 'エラーが発生したとき' (When error occurs) - 'IM-LogicDesigner フローラーティング (tutorial_enquete/question) にリクエストを送信する' (Send request to IM-LogicDesigner FlowRouter (tutorial_enquete/question)).
- Action 3: 'アクションアイテム説明' (Action item description) - 'カスタムスクリプトを実行する' (Execute custom script).

At the bottom right, there are '決定' (Decision) and '取り消し' (Cancel) buttons.

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。

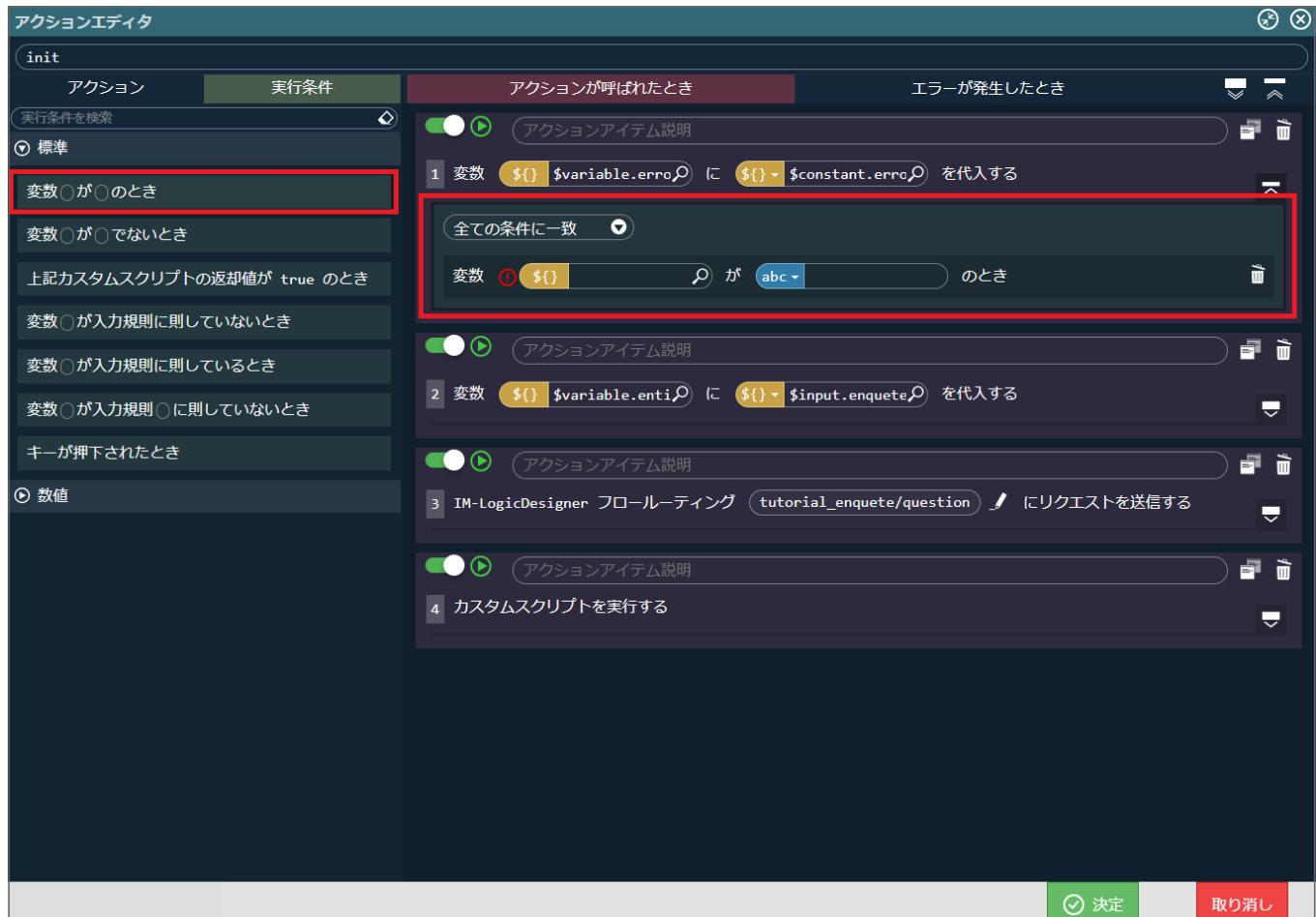
エラー画面のメッセージを設定するため、「アクション」タブから「標準」 - 「変数〇に〇を代入する」を選択し、「アクションが呼ばれたとき」の一一番上に配置します。

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



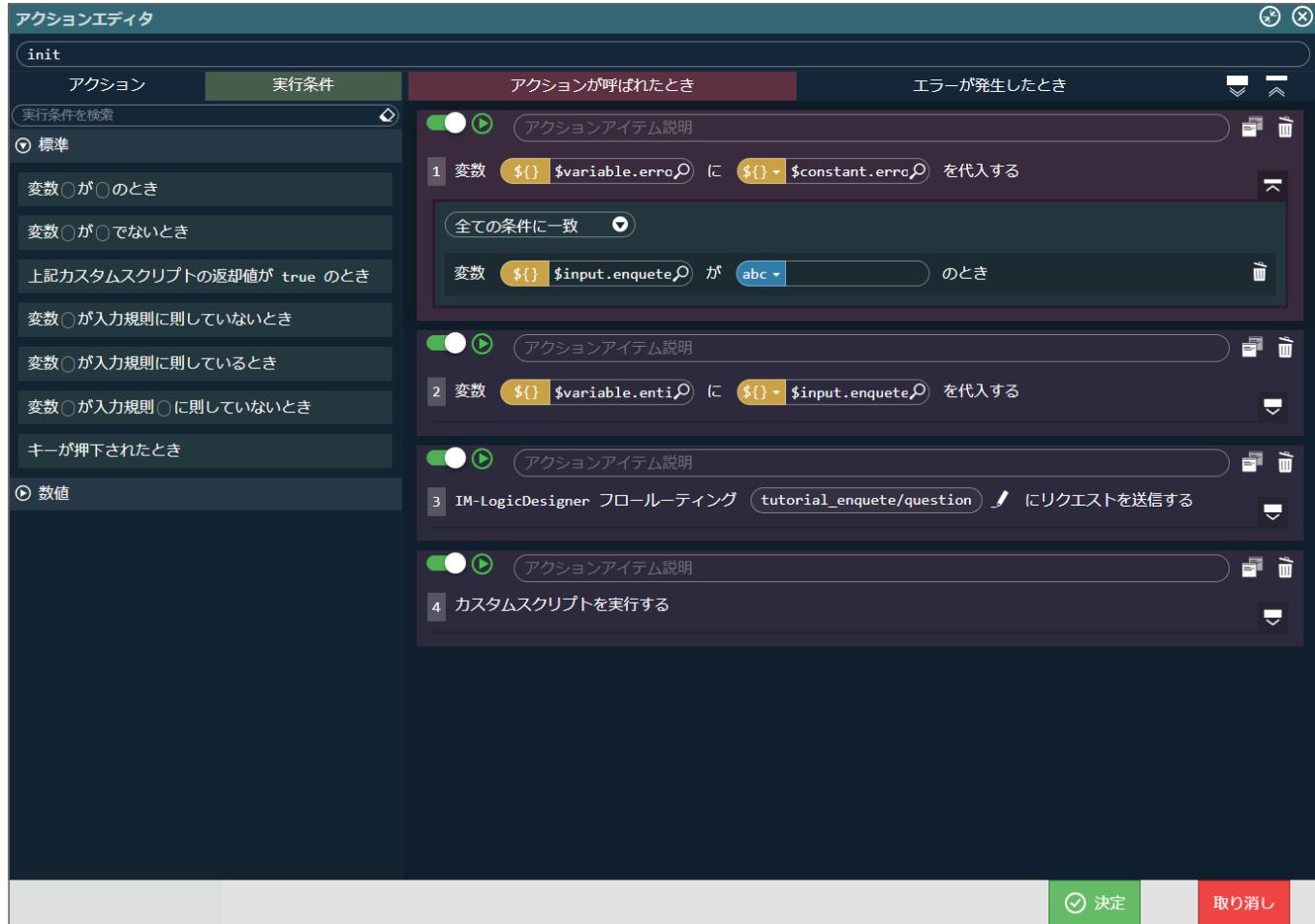
アクションエディタ左側の実行条件タブから「標準」 - 「変数○が○のとき」を選択肢、直前で編集していたアクションに被せるようにドラッグ & ドロップして実行条件を追加します。



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

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



次に、エラー画面表示の設定を行います。

「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後に配置します。

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」を直前に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

アクションエディタ

init

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

1 変数 \${variable.error} に \${constant.error} を代入する

2 ページ error を開く

3 変数 \${variable.entity} に \${input.enquete} を代入する

4 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

5 カスタムスクリプトを実行する

決定 取り消し

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。

「ページ(error)を開く」アクション左上の「実行後に後続処理を続行する」をクリックして「実行後にアクションを終了する」に変更します。

アクションエディタ

init

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

1 全ての条件に一致 のとき

2 ページ error を開く

実行後にアクションを終了する

3 変数 \${variable.entity} に \${input.enquete} を代入する

4 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

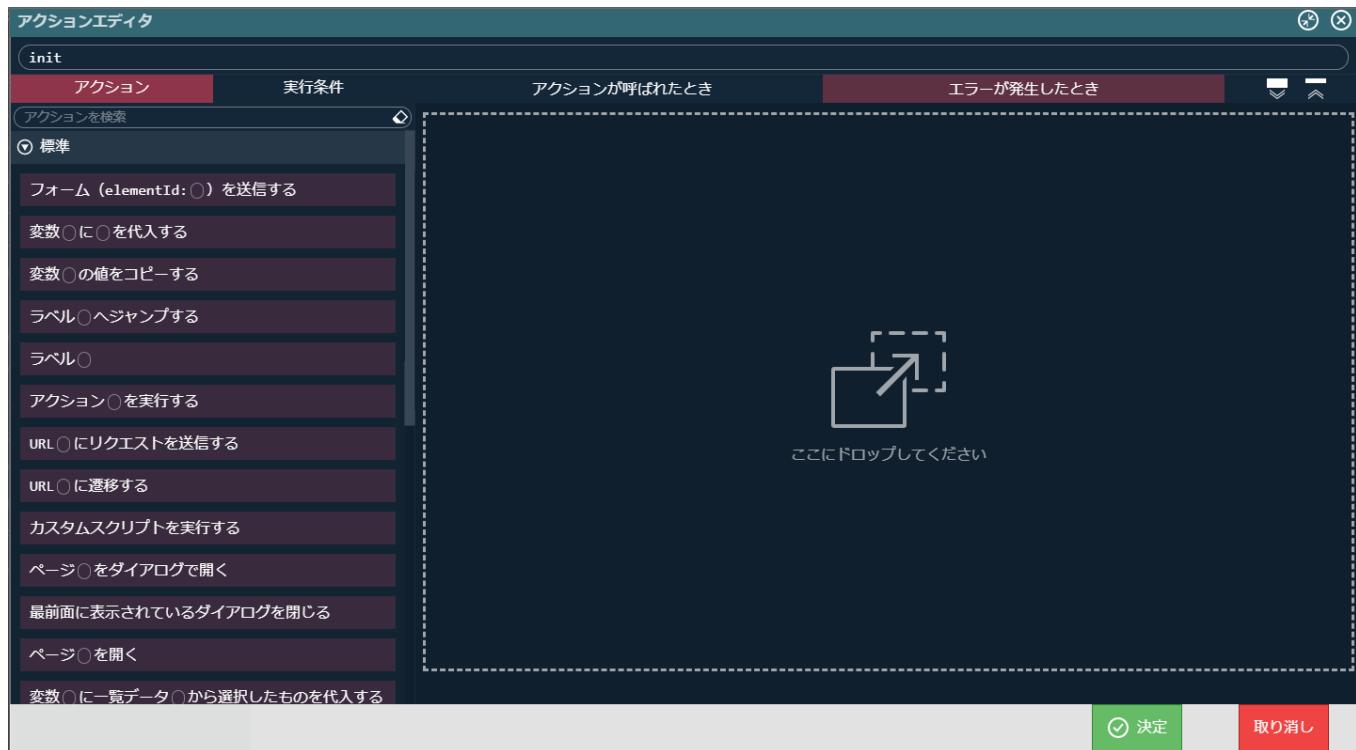
5 カスタムスクリプトを実行する

決定 取り消し

これで、画面表示時にenqueteid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○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。

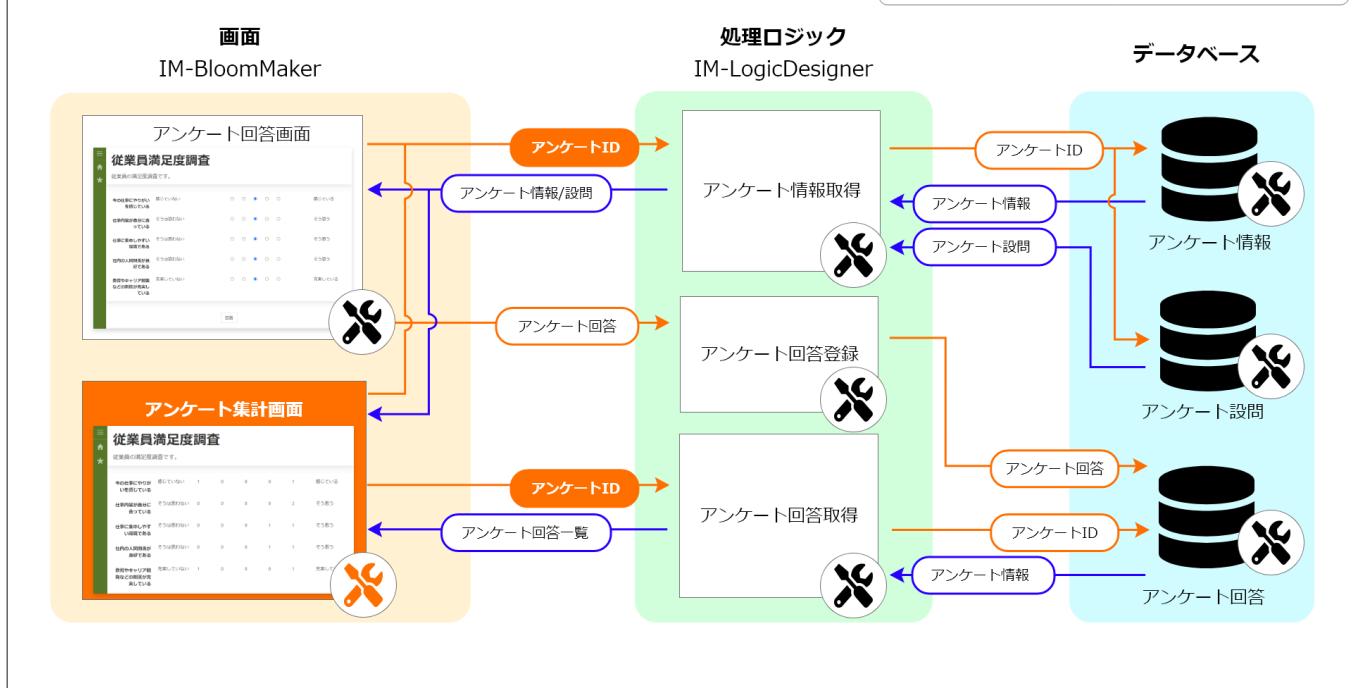
集計画面の作成

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。

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。

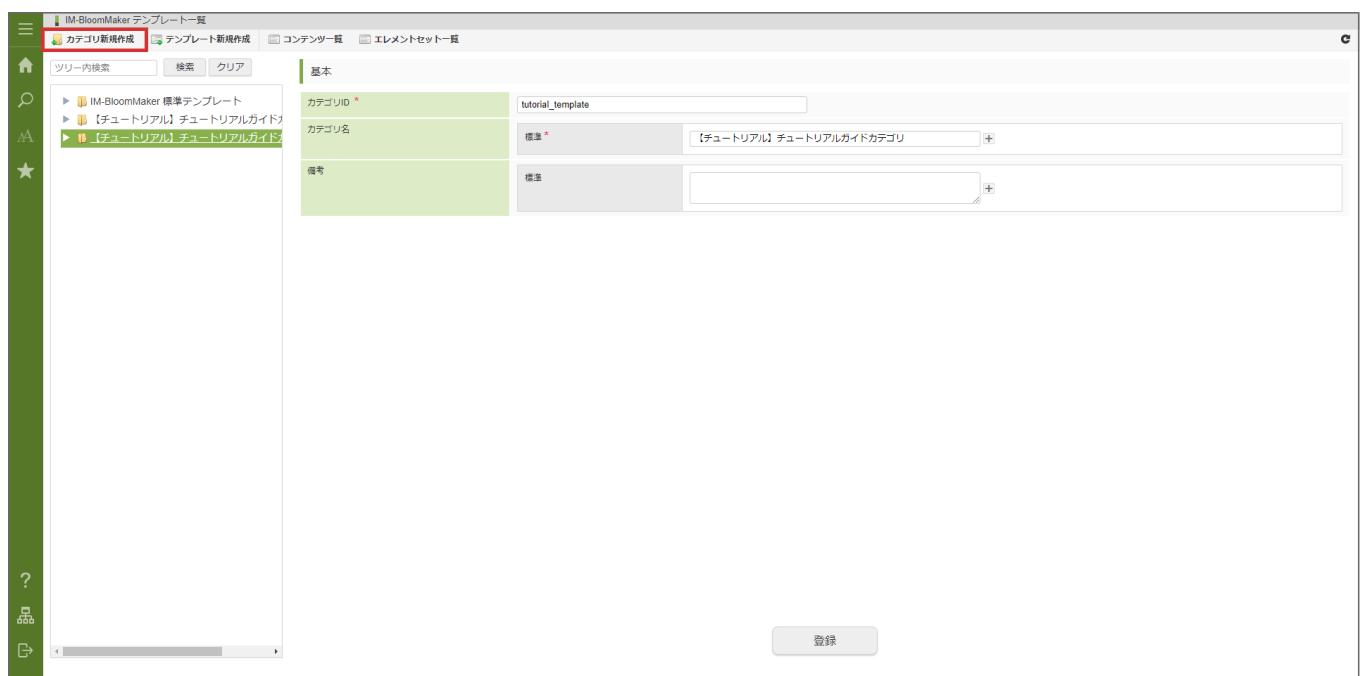
テンプレートの登録は IM-BloomMaker の機能を用いて行います。

「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。

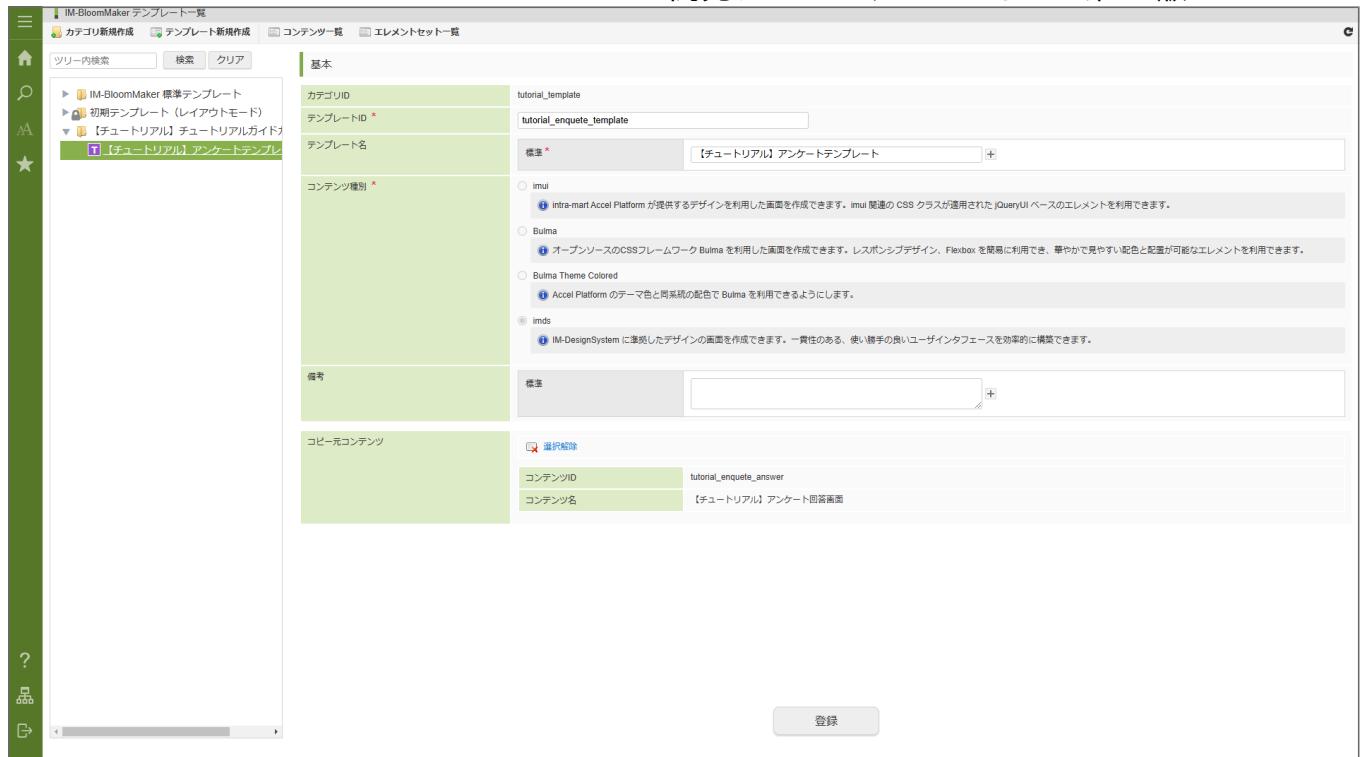


作成したカテゴリをクリックし、「テンプレート新規作成」をクリックします。

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。

「コンテンツ名」に「【チュートリアル】アンケート集計画面」と入力します。

IM-BloomMaker コンテンツ新規作成

基本

| | |
|-----------|---|
| カテゴリ * | <input type="text" value="im_accel_studio"/> 検索 カテゴリID カテゴリ名 【チュートリアル】アンケート集計画面 |
| コンテンツID * | <input type="radio"/> imui 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 開発の 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。 |
| コンテンツ種別 * | <input type="radio"/> Bulma 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ます。 |
| | <input type="radio"/> Bulma Theme Colored 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。 |
| | <input type="radio"/> imds 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、使い勝手の良いユーザ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 | <input type="text" value="【チュートリアル】アンケート集計画面"/> 標準 |
| 備考 | <input type="text"/> 標準 |
| テンプレート | <input type="text"/> 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

テンプレート名を入力してください。

| | | |
|---|--|---|
| 【チュートリアル】アンケートテンプ... カテゴリ名：【チュートリアル】チュートリ... 最終更新者：tenant 最終更新日：2024/07/11 17:07:55 | 一覧テーブルテンプレート カテゴリ名：初期テンプレート（レイアウト... 最終更新者： 最終更新日：2024/07/10 17:40:50 | 入力フォームテンプレート カテゴリ名：初期テンプレート（レイアウト... 最終更新者： 最終更新日：2024/07/10 17:40:49 |
| 空白テンプレート カテゴリ名：初期テンプレート（レイアウト... 最終更新者： | IM-BloomMaker テンプレート一覧... カテゴリ名：IM-BloomMaker 標準テンプレート ユーザ検索 最終更新者： | IM-BloomMaker テンプレート一覧... カテゴリ名：IM-BloomMaker 標準テンプレート 最終更新者： |

「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。

The screenshot shows the IM-BloomMaker Content creation interface. The left sidebar has a 'Basic' tab selected. The main area contains several input fields and dropdown menus:

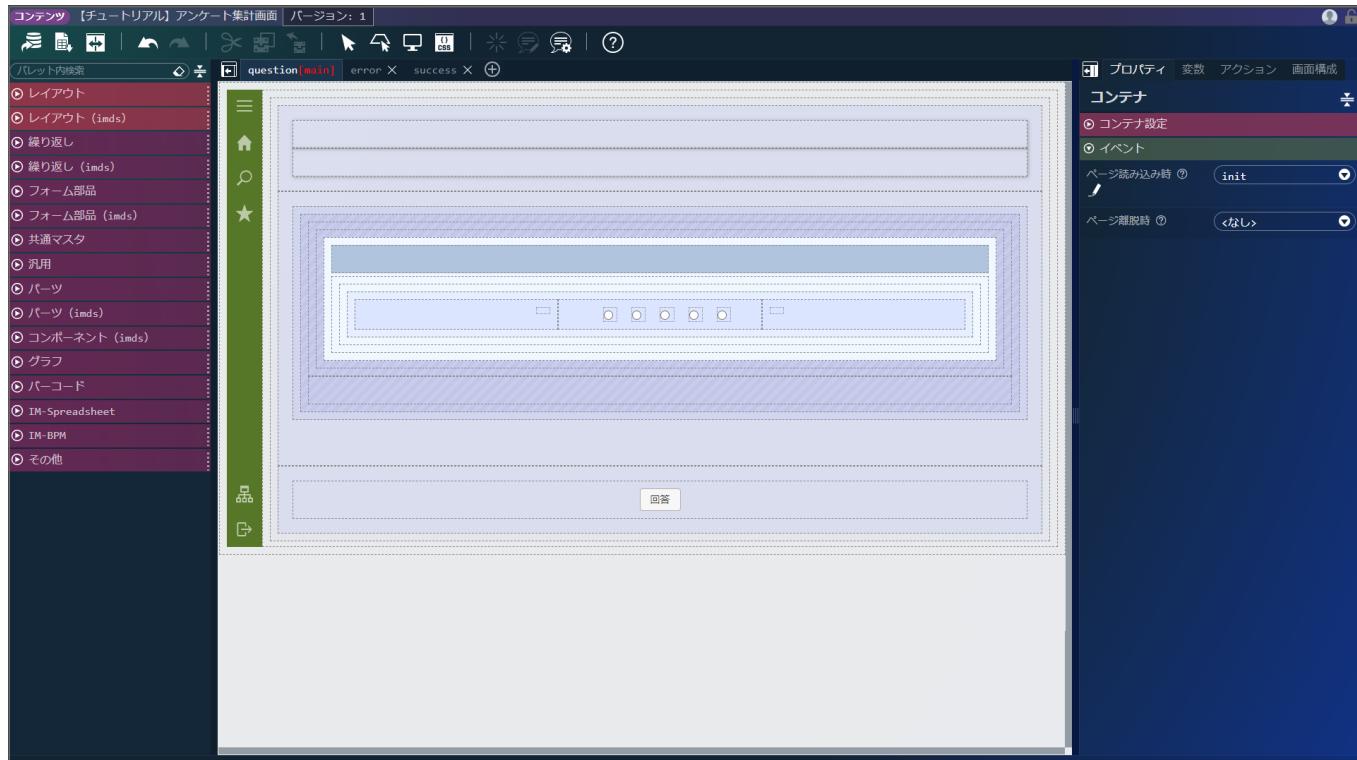
- Category ***: A dropdown menu with options like 'imui', 'Bulma', 'Bulma Theme Colored', and 'imds'. The 'imui' option is currently selected.
- Content ID ***: A dropdown menu with options like 'tutorial_enquete_totalize'.
- Content Type ***: A dropdown menu with options like 'imui', 'Bulma', 'Bulma Theme Colored', and 'imds'.
- Latest Version Number**: A text input field containing '1'.
- Content Name**: A dropdown menu with options like '【チュートリアル】アンケート集計画面'.
- Notes**: A dropdown menu with options like '標準'.
- Template**: A dropdown menu with options like '選択解除', 'Template ID: tutorial_enquete_template', and 'Template Name: 【チュートリアル】アンケートテンプレート'.

A large '登録' (Register) button is located at the bottom right of the form.

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 デザイナ編集画面を開きます。

「【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上に表示されます。



変数の設定

回答画面作成時に設定した変数を、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。

The screenshot shows the variable editor interface. At the top, there are tabs for 'プロパティ', '変数' (selected), 'アクション', and '画面構成'. Below the tabs are icons for '\$' (add variable), file (add action), JSON, and delete. The main area displays a hierarchical tree of variables:

- \$variable <マップ>
- responseData <マップ>
 - records <配列>
 - 0 <マップ>
 - enquete_id <文字列> <データなし>
 - question_id <文字列> <データなし>
 - question_no <浮動小数点数> <データなし>
 - question <文字列> <データなし>
 - left_scale <文字列> <データなし>
 - right_scale <文字列> <データなし>
 - enqueteTitle <文字列> <データなし>
 - enqueteOverview <文字列> <データなし>
 - entity <マップ>
 - enqueteId <文字列> <データなし>- answersData <マップ> (highlighted with yellow background)
 - enqueteId <文字列> <データなし>
 - answers <配列>
 - errorMessage <文字列> <データなし>

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。

The screenshot shows the '変数エディタ' (Variable Editor) dialog box. The configuration is as follows:

- キー名: responseTotalData
- 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

At the bottom right, there are buttons for '決定' (Decision) and '取り消し' (Cancel).

作成した「\$variable > responseTotalData」を選択し、「IM-Repositoryから作成」アイコンをクリックしてプルダウンから「エンティティをインポート」を選択します。

The screenshot shows the 'Variable' configuration screen. At the top, there are tabs for 'プロパティ' (Properties), '変数' (Variables), 'アクション' (Actions), and '画面構成' (Page Configuration). Below the tabs, a toolbar includes icons for '変数' (Variables), '\$+' (New Variable), 'Import' (highlighted with a red box), '\$-' (Delete Variable), 'JSON', and a trash bin. The main area displays a hierarchical list of variables:

- \$variable <マップ> 辞書項目をインポート <マップ>
- responseData <エンティティをインポート> <マップ>
 - records <マップ> <配列>
 - 0 <マップ> <マップ>
 - enquete_id <文字列> <データなし>
 - question_id <文字列> <データなし>
 - question_no <浮動小数点数> <データなし>
 - question <文字列> <データなし>
 - left_scale <文字列> <データなし>
 - right_scale <文字列> <データなし>
 - enqueteTitle <文字列> <データなし>
 - enqueteOverview <文字列> <データなし>
 - entity <マップ> <マップ>
 - enqueteId <文字列> <データなし>
 - errorMessage <文字列> <データなし>- responseTotalData <マップ> <マップ>

表示された「エンティティ検索」画面から、「アンケート回答」を選択して決定をクリックしてください。

The screenshot shows the 'Entity Search' dialog box. On the left, a sidebar lists categories: 'エンティティIDまたはエンティティ名' (Entity ID or Entity Name), '[チュートリアル] チュートリアル' (Tutorial Tutorial), and three items under 'アンケート': 'アンケート回答' (Survey Answer) which is selected (highlighted in green), 'アンケート設問' (Survey Question), and 'アンケート情報' (Survey Information). The main area has tabs for '詳細情報' (Detailed Information) and '選択済み項目' (Selected Items). The 'Selected Items' tab on the right shows the selected item 'アンケート回答'. The 'Basic' tab in the center displays entity information:

| エンティティ項目 | | 関連一覧 | データ定義 |
|--|--|-------------------------------------|-------|
| エンティティID tutorial_enquete_answer_app_entity | | | |
| エンティティ名 | | 標準 アンケート回答 | |
| | | 日本語 | |
| | | 英語 | |
| | | 中国語 (中国) | |
| 説明 | | 標準 | |
| | | 日本語 | |
| | | 英語 | |
| | | 中国語 (中国) | |
| エンティティの有効化 | | <input checked="" type="checkbox"/> | |

At the bottom right are buttons for 'クリア' (Clear) and '決定' (Decision).

「\$variable > responseTotalData」配下に「tutorial_enquete_answer_app_entity」が追加されました。

The screenshot shows the 'Variables' tab in the Accel Platform interface. A variable named '\$variable' is expanded, showing its properties: responseData (Map), entity (Map), errorMessage (String), and responseTotalData (Map). The 'responseTotalData' property is selected and highlighted with a yellow background.

次に、「tutorial_enquete_answer_app_entity」の鉛筆アイコンをクリックして変数エディタ画面を表示します。

キー名を「records」に変更し、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。

The screenshot shows the 'Variable Editor' dialog box. The 'Key Name' field is set to 'records'. The 'Array Dimension' dropdown is set to '1-dimensional array' and the 'Element Count' dropdown is set to '1'. The 'Type' section shows 'List' selected. The 'Input Rule' section has '必修' (Required) checked. At the bottom right, there are '決定' (Decision) and '取り消し' (Cancel) buttons.

「records」配下に作成された「0」の をクリックして、変数エディタ画面で代入値を「マップ」に変更します。

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": [
    {
      "answer_id": "",
      "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。

The screenshot shows the variable editor interface. At the top, there are tabs for 'プロパティ' (Properties), '変数' (Variables), 'アクション' (Actions), and '画面構成' (Page Configuration). Below the tabs is a toolbar with icons for adding a variable (\$+), adding an action (document+), a JSON icon, and a delete icon.

The main area displays a hierarchical list of variables:

- \$variable <マップ>
- responseData <マップ>
- entity <マップ>
- errorMessage <文字列>
- responseTotalData <マップ>
- totalData <マップ> (This item is selected and highlighted in yellow.)
 - 0 <マップ>

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。

The screenshot shows the variable editor interface with the '変数' (Variables) tab selected. A single entry is listed under the variable '\$variable':

| ▶ <i>responseData</i> <マップ> | <マップ> |
|----------------------------------|---------|
| ▶ <i>entity</i> <マップ> | <マップ> |
| <i>errorMessage</i> <文字列> | <データなし> |
| ▶ <i>responseTotalData</i> <マップ> | <マップ> |
| ▼ <i>totalData</i> <マップ> | <配列> |
| ▶ <i>0</i> <マップ> | <マップ> |
| <i>a1</i> <整数> | 0 |

同様の設定で、「a2」、「a3」、「a4」、「a5」を作成します。

The screenshot shows the variable editor interface with the '変数' (Variables) tab selected. Multiple entries are listed under the variable '\$variable':

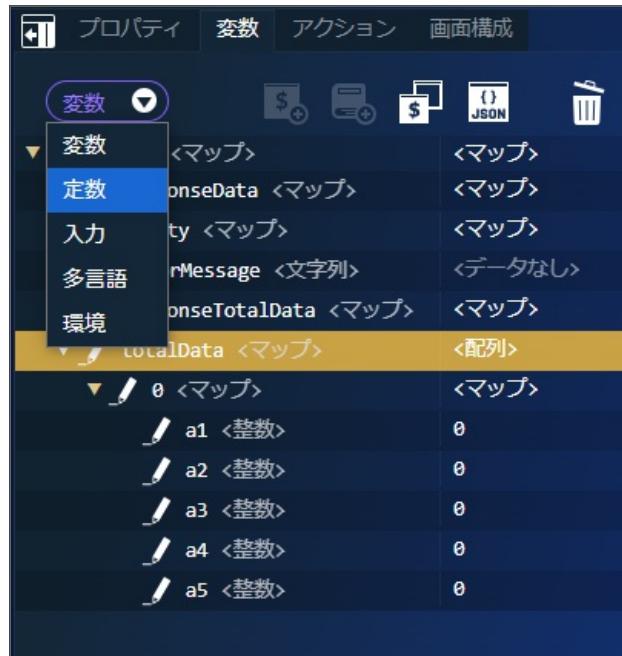
| ▶ <i>responseData</i> <マップ> | <マップ> |
|----------------------------------|---------|
| ▶ <i>entity</i> <マップ> | <マップ> |
| <i>errorMessage</i> <文字列> | <データなし> |
| ▶ <i>responseTotalData</i> <マップ> | <マップ> |
| ▼ <i>totalData</i> <マップ> | <配列> |
| ▶ <i>0</i> <マップ> | <マップ> |
| <i>a1</i> <整数> | 0 |
| <i>a2</i> <整数> | 0 |
| <i>a3</i> <整数> | 0 |
| <i>a4</i> <整数> | 0 |
| <i>a5</i> <整数> | 0 |

```
"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。

次に、対応する定数を作成します。

「変数」タブのプルダウンで「変数」を「定数」に変更します。



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

キー名を「template」、値の型と代入値を「マップ」と設定し、「決定」をクリックします。

定数エディタ

| | |
|--------|---|
| キー名 | template |
| コメント | |
| 値の型 | マップ |
| 配列の次元数 | 配列にしない |
| 代入値 | <input checked="" type="radio"/> データなし <input type="radio"/> マップ |

決定

作成した「template」を選択し、 をクリックして定数エディタを表示します。

キー名に「a1」と入力し、値の型を「整数」、代入値の「固定数値」に「0」を設定して「決定」をクリックします。

定数エディタ

| | |
|--------|---|
| キー名 | a1 |
| コメント | |
| 値の型 | 整数 |
| 配列の次元数 | 配列にしない |
| 代入値 | <input checked="" type="radio"/> データなし <input type="radio"/> 固定数値 0 |

決定

同様の設定で、「a2」、「a3」、「a4」、「a5」を作成します。

The screenshot shows the variable configuration screen. At the top, there are tabs for 'プロパティ' (Properties), '変数' (Variables), 'アクション' (Actions), and '画面構成' (Page Configuration). The '変数' tab is selected. On the left, a tree view shows variables categorized by type: '\$constant' (constant) and 'template' (map). Under 'template', there are five variables: 'a1' through 'a5', all of which are mapped to the value '0'. On the right, there is a detailed description of each variable, such as 'アンケートIDが設定されていません。' (Survey ID is not set) for 'error_no_id'.

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。

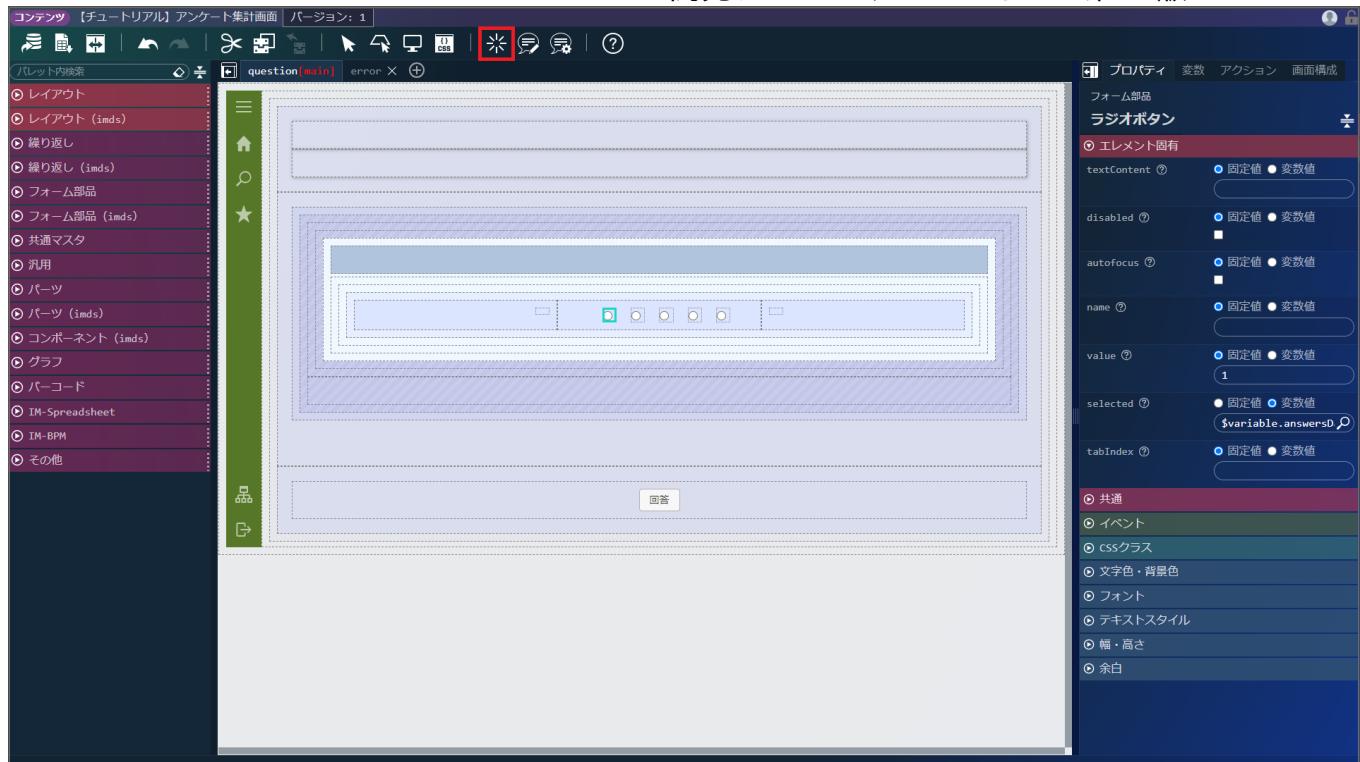
エディタの「success」タブ右側に表示されている「×

をクリックし、回答成功画面を削除します。

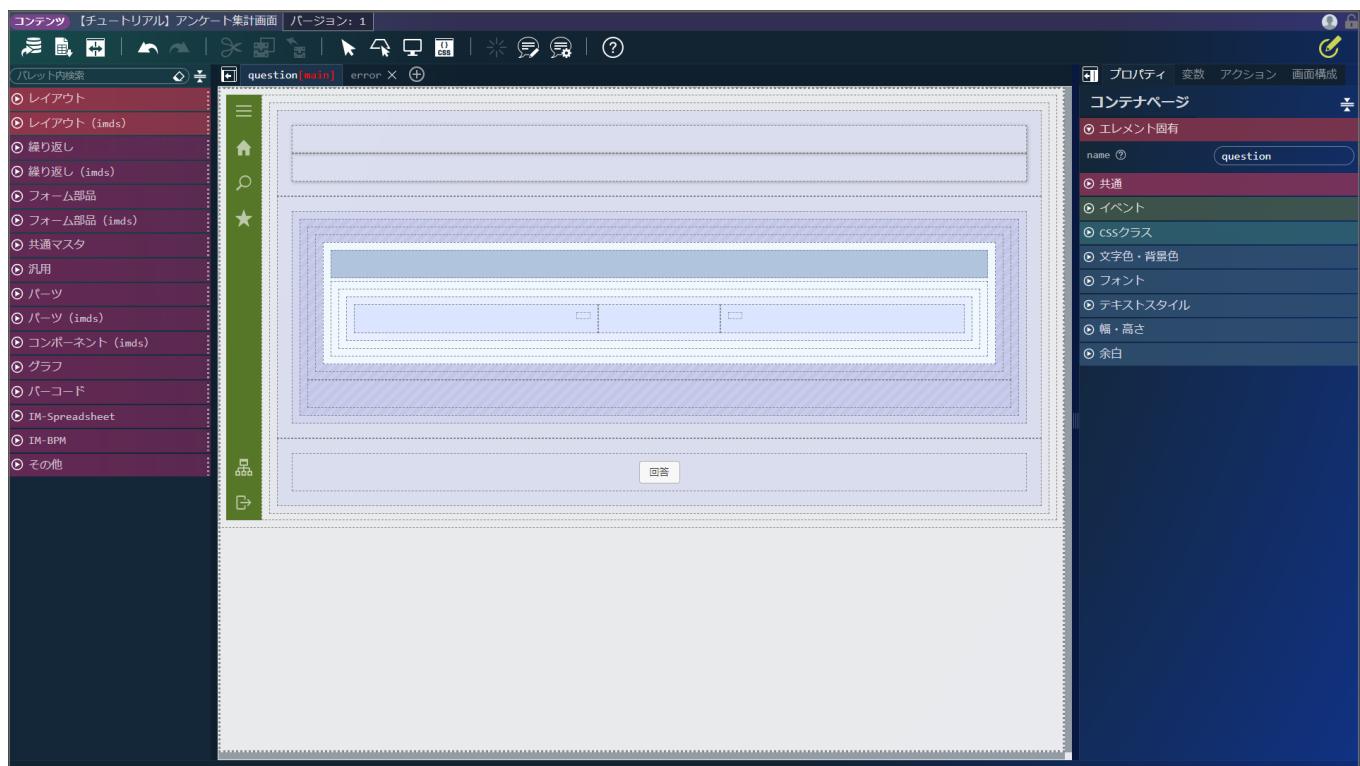
The screenshot shows the visual editor interface. The top bar shows the title '【チュートリアル】アンケート集計画面 | バージョン: 1' and various tool icons. The main workspace contains a component labeled 'question[main] error X success X'. A red box highlights the 'success' tab, which is currently active. To the left is a sidebar with a list of components: 'レイアウト', 'レイアウト (imds)', '繰り返し', '繰り返し (imds)', 'フォーム部品', 'フォーム部品 (imds)', '共通マスク', '汎用', 'パート', 'パート (imds)', 'コンポーネント (imds)', 'グラフ', 'バーコード', 'IM-Spreadsheet', 'IM-BPM', and 'その他'. On the right, the properties panel is open, showing settings for the 'success' tab under the 'conteiner' section. The 'name' field is set to 'success' with a red box around it. Other settings include 'イベント' (Event) set to 'success', 'CSSクラス' (CSS Class), '文字色・背景色' (Text Color・Background Color), 'フォント' (Font), 'テキストスタイル' (Text Style), '幅・高さ' (Width・Height), and '余白' (Margin).

次に、回答項目のラジオボタンを削除します。

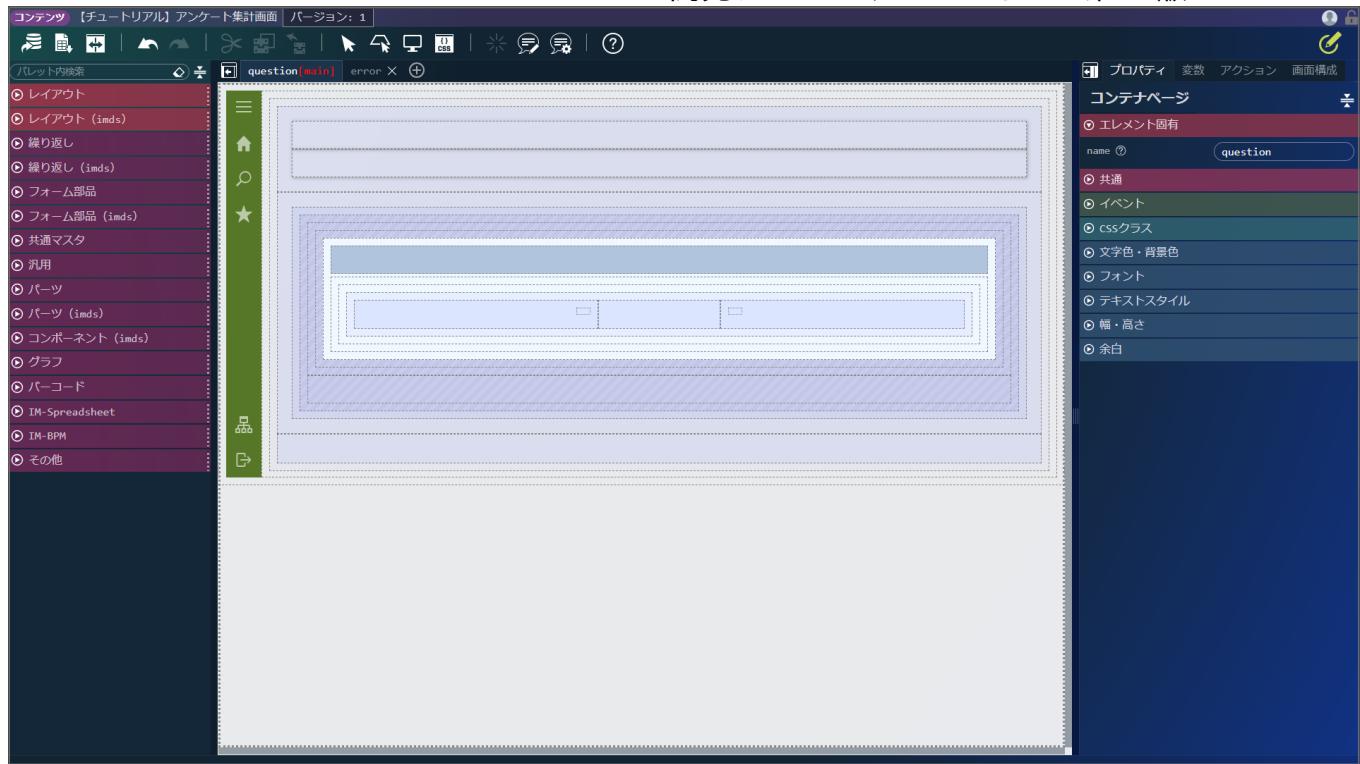
エディタの「question」タブを開き、回答項目のラジオボタンを選択し、DELETEキーを押すか画面上部の をクリックしてください。



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。

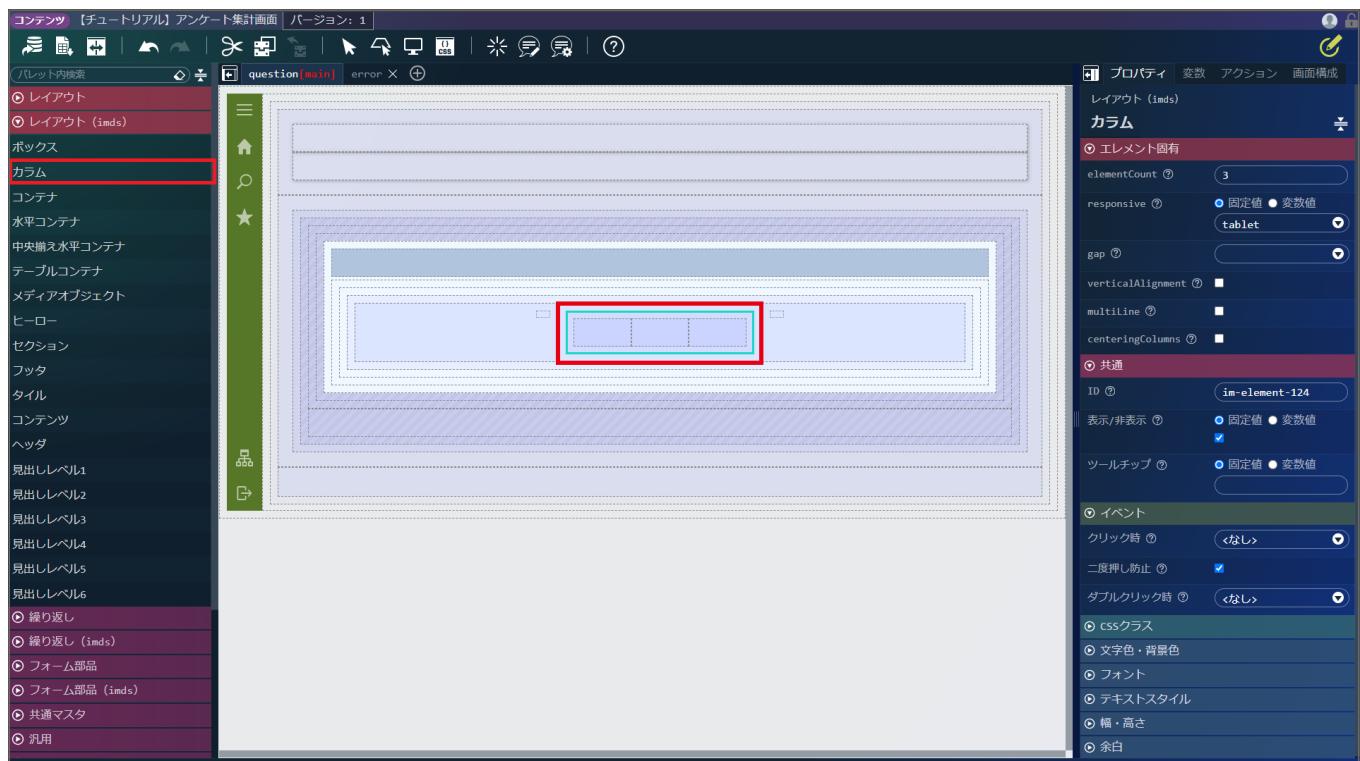


画面項目の追加

集計画面独自の画面項目を追加します。

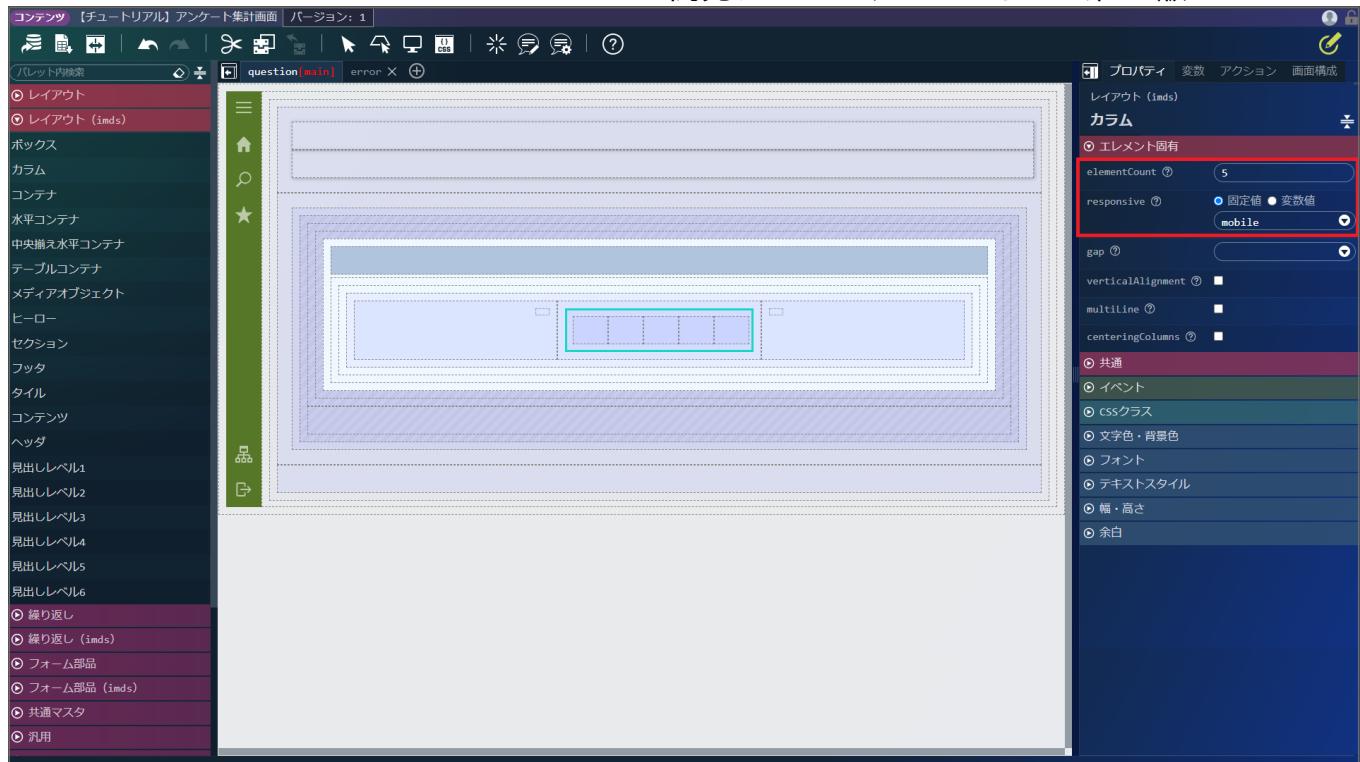
集計画面では、どの選択肢に何人のユーザが回答したかを数値で表示します。

「レイアウト (imds)」 - 「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。



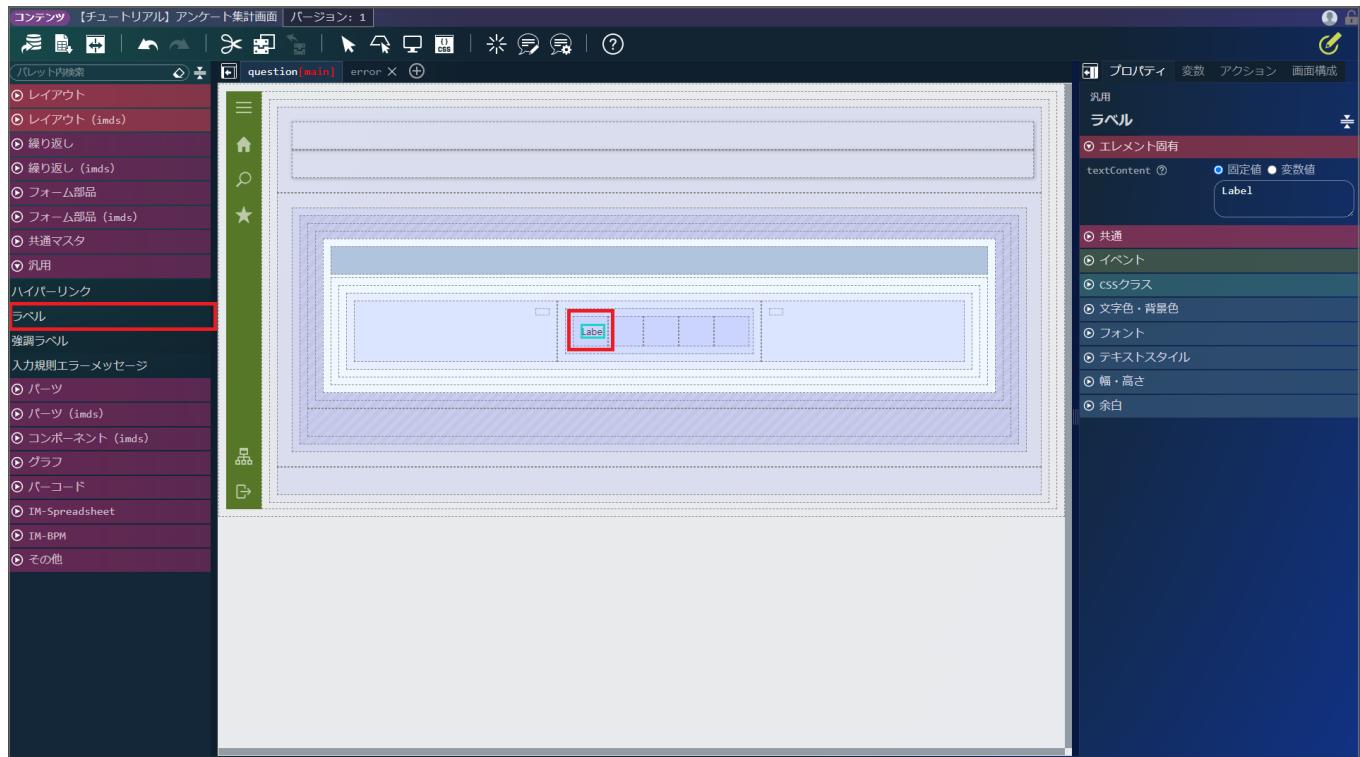
新たに配置したカラムを選択し、画面右側「プロパティ」タブの「カラム」 - 「エレメント固有」 - 「elementCount」に「5」を指定します。

「エレメント固有」 - 「responsive」のプルダウンから「mobile」を選択します。



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。

変数 定数 入力 多言語 環境

キーワード検索 ◇

繰り返しエレメントを考慮 ②

| | |
|---------------------------|---------|
| enqueteOverview <文字列> | <データなし> |
| ▼ entity <マップ> | <マップ> |
| enqueteId <文字列> | <データなし> |
| errorMessage <文字列> | <データなし> |
| ▼ responseTotalData <マップ> | <マップ> |
| ▼ records <マップ> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| answer_id <文字列> | <データなし> |
| enquete_id <文字列> | <データなし> |
| respondent <文字列> | <データなし> |
| answer_date <日付・時刻> | <データなし> |
| answers <文字列> | <データなし> |
| ▼ totalData <マップ> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| a1 <整数> | 0 |
| a2 <整数> | 0 |
| a3 <整数> | 0 |
| a4 <整数> | 0 |
| a5 <整数> | 0 |

① 決定 取り消し

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。

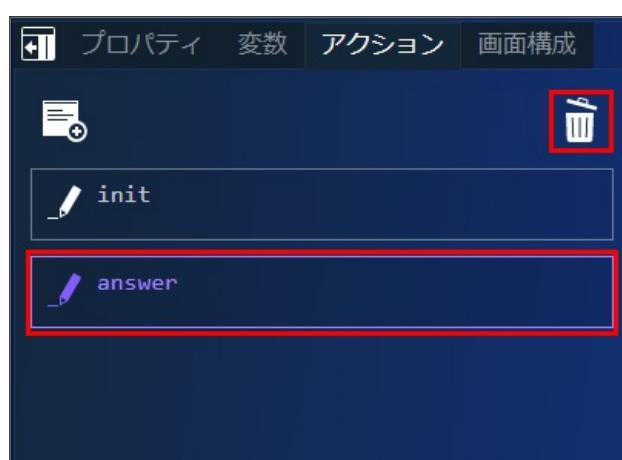
アクションの追加、修正

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。

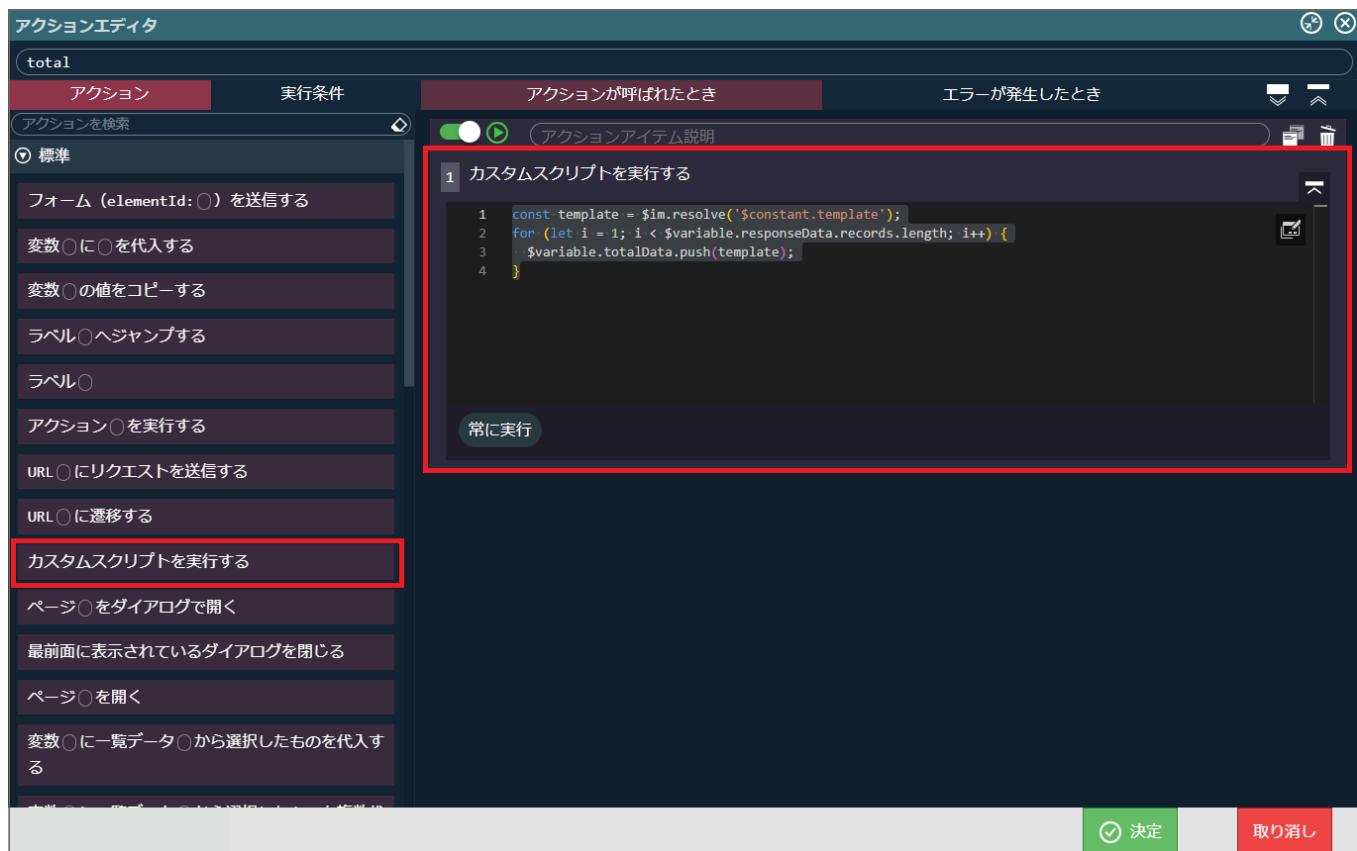
 をクリックしてアクションエディタを表示し、名称を「新しいアクション #1」から「total」に変更します。

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。

```
const template = $im.resolve("$constant.template");
for(let i = 1; i < $variable.responseData.records.length; i++) {
    $variable.totalData.push(template);
}
```

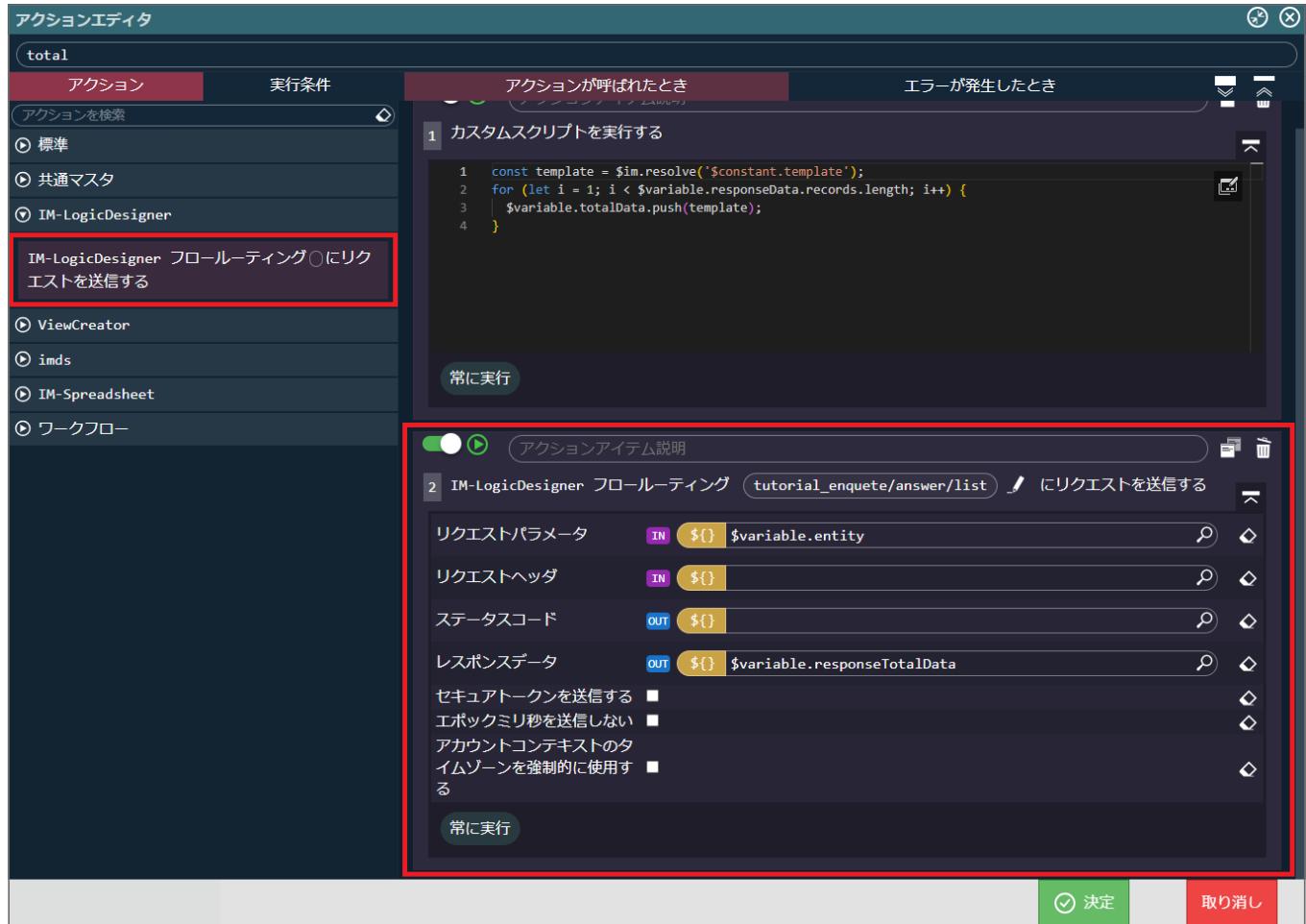


これで、取得した設問項目数に合わせて回答の集計結果を格納する変数の配列が作れるようになりました。

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++){
  const record = $variable.responseTotalData.records[i]

  // 回答の配列化
  const answers = record.answers.split(',');

  // 各回答数をカウント
  for(let j = 0; j < answers.length; j++){
    switch (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

アクションを検索

標準

- フォーム (elementId:○) を送信する
- 変数○に○を代入する
- 変数○の値をコピーする
- ラベル○へジャンプする
- ラベル○
- アクション○を実行する
- URL○にリクエストを送信する
- URL○に遷移する
- カスタムスクリプトを実行する** (Red box)
- ページ○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○を開く
- 変数○に一覧データ○から選択したものを代入する

アクションが呼ばれたとき エラーが発生したとき

- カスタムスクリプトを実行する
- 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
- 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   const record = $variable.responseTotalData.records[i]
3
4   // 回答の配列化
5   const answers = record.answers.split(',')
6
7   // 各回答数をカウント
8   for(let j = 0; j < answers.length; j++){
9     switch (answers[j]){
10       case "1":
11         $variable.totalData[1] += 1;
12       break;
13     }
14   }
15 }

```

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○を実行する」を選択し、initアクションの一番下に配置します。

アクションエディタ

init

アクション 実行条件

アクションを検索

標準

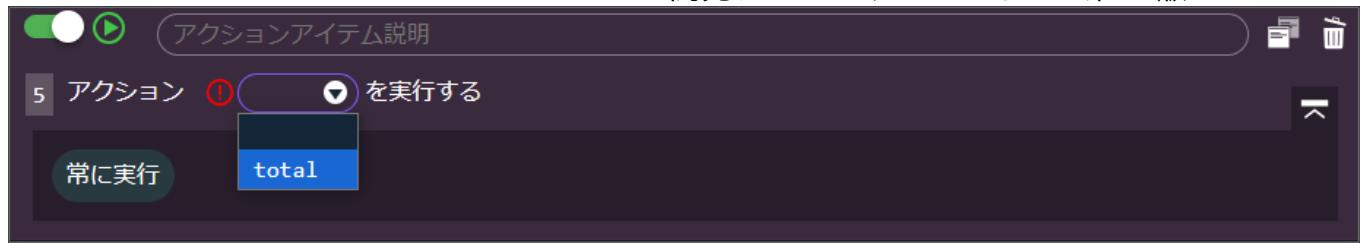
- フォーム (elementId:○) を送信する
- 変数○に○を代入する
- 変数○の値をコピーする
- ラベル○へジャンプする
- ラベル○
- アクション○を実行する** (Red box)
- URL○にリクエストを送信する
- URL○に遷移する
- カスタムスクリプトを実行する
- ページ○をダイアログで開く
- 最前面に表示されているダイアログを閉じる
- ページ○を開く
- 変数○に一覧データ○から選択したものを代入する

アクションが呼ばれたとき エラーが発生したとき

- 変数 \${} \$variable.error に \${} \$constant.error を代入する
- ページ error を開く
- 変数 \${} \$variable.entity に \${} \$input.enquete を代入する
- 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
- カスタムスクリプトを実行する
- アクション (Red box)

決定 取り消し

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。

アクションエディタ

init

アクション 実行条件

3. 変数 `variable.entity` に `$input.enquete` を代入する

4. 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

5. カスタムスクリプトを実行する

```

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

```

6. アクション (total) を実行する

eslint 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。

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」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。

The screenshot shows the 'Application Management' interface. In the center, there's a table listing various API URLs under the 'Tutorial Application'. On the right side of the table, there are several buttons and dropdown menus. One of these buttons is highlighted with a red box and labeled '画面URL' (Page URL). Below this button, there are other options like 'APIのURL' (API URL) and 'クエリのURL' (Query URL). To the right of the table, there are more settings and filters, including a search bar and a checkbox for '表示操作' (Display Operations).

「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。

ルーティングカテゴリが存在しない場合は、この手順の前に「サイトマップ」→「BloomMaker」→「ルーティング定義一覧」から任意のルーティングカテゴリを新規作成してください。

「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」を設定します。

「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。

ツリー内検索 検索 クリア

▼ 業務テンプレートから作成

【チュートリアル】アンケート回答画面
【チュートリアル】アンケート集計画面

| 基本 | |
|-----------|-------------------------|
| カテゴリID | im_accel_studio |
| コンテンツID | tutorial_enquete_answer |
| コンテンツ種別 | imds |
| 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】アンケート回答画面 |
| 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/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。

IM-BloomMaker ルーティング

| | |
|----------------|--|
| カテゴリ * | im_bouquet |
| ルーティングID * | tutorial_enquete_answer |
| コンテンツ * | tutorial_enquete_answer |
| 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 |
| メソッド * | GET |
| URL * | /imart/tutorial_enquete/answer |
| 認可URI | im-bloommaker-content://contents/route/tutorial_enquete_answer |
| ルーティング名 | 標準 【チュートリアル】アンケート回答 |
| 備考 | 標準 |

登録

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」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。

The screenshot shows a list of API URLs in a browser-like interface. A context menu is open over the third item, which is highlighted with a red box. The menu items shown are '画面URL', 'APIのURL', and 'クエリのURL'. The list includes entries like 'APIのURL (ID: im-logic://routes/tutorial_enquete/answer/list/GET) GET tutorial_enquete/answer/list' and 'APIのURL (ID: im-logic://routes/tutorial_enquete/question/GET) GET tutorial_enquete/question'.

「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。

| 項目 | 値 |
|-----------|---------------------------|
| カテゴリID | im_accel_studio |
| コンテンツID | tutorial_enquete_totalize |
| コンテンツ種別 | imds |
| 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 【チュートリアル】アンケート回答画面 |

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」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。

The screenshot shows the 'IM-BloomMaker ルーティング' configuration page. It includes fields for:

- カテゴリ ***: im_bouquet
- ルーティングID ***: tutorial_enquete_totalize
- コンテンツ ***: tutorial_enquete_totalize
- コンテンツバージョン番号 ***:
 - 現行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: [選択]
- メソッド ***: GET
- URL ***: /imart/ tutorial_enquete/totalize
- 認可URI**: im-bloommaker-content://contents/route/tutorial_enquete_totalize
- ルーティング名**: 【チュートリアル】アンケート集計
- 権限**: 【チュートリアル】アンケート集計

A '登録' (Register) button is at the bottom right.

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」→「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「権限設定」プルダウンから「メニュー設定」を選択し、「メニュー設定」画面を表示します。

The screenshot shows a list of resources or URLs in the 'メニュー設定' (Menu Setting) section. The 'メニュー設定' button is highlighted with a red box in the top right corner of the header.

| リソース名 | 説明 | 操作 |
|--|--|-----------------------------------|
| 画面 [チュートリアル] アンケートアプリケーション - 詳細 | 画面 [チュートリアル] アンケートアプリケーション - 詳細 | ... tenant 1時間前 |
| 画面 [チュートリアル] アンケート回答画面 | 画面 [チュートリアル] アンケート回答画面 | ... tenant 23時間前 |
| 画面URL [チュートリアル] アンケート集計 | 画面URL [チュートリアル] アンケート集計 | ... tenant 0時間前 |
| 画面URL [チュートリアル] アンケート回答 | 画面URL [チュートリアル] アンケート回答 | ... tenant 0時間前 |
| APIのURL (ID: im-logic://routes/tutorial_enquete/answer/list/GET) | APIのURL (ID: im-logic://routes/tutorial_enquete/answer/list/GET) | ... tenant 2024/12/25 15:37:15 |
| APIのURL (ID: im-logic://routes/tutorial_enquete/answer/POST) | APIのURL (ID: im-logic://routes/tutorial_enquete/answer/POST) | ... tenant 2024/12/25 15:38:29 |
| APIのURL (ID: im-logic://routes/tutorial_enquete/question/GET) | APIのURL (ID: im-logic://routes/tutorial_enquete/question/GET) | ... tenant 2024/12/25 15:39:00 |
| ロジックフロー (ID: im-logic://flows/tutorial_answers_get) | ロジックフロー (ID: im-logic://flows/tutorial_answers_get) | ... tenant 2024/12/24 17:48:17 |
| ロジックフロー (ID: im-logic://flows/tutorial_answer_insert) | ロジックフロー (ID: im-logic://flows/tutorial_answer_insert) | ... tenant 2024/12/24 17:01:35 |
| ロジックフロー (ID: im-logic://flows/tutorial_enquete_info) | ロジックフロー (ID: im-logic://flows/tutorial_enquete_info) | ... tenant 2024/12/24 16:32:03 |
| エンティティ (ID: im-repository/entity/tutorial_enquete_answer_app_entity) | エンティティ (ID: im-repository/entity/tutorial_enquete_answer_app_entity) | ... tenant 2024/12/23 17:57:52 |
| エンティティ (ID: im-repository/entity/tutorial_enquete_question_app_entity) | エンティティ (ID: im-repository/entity/tutorial_enquete_question_app_entity) | ... tenant 2024/12/23 17:57:52 |

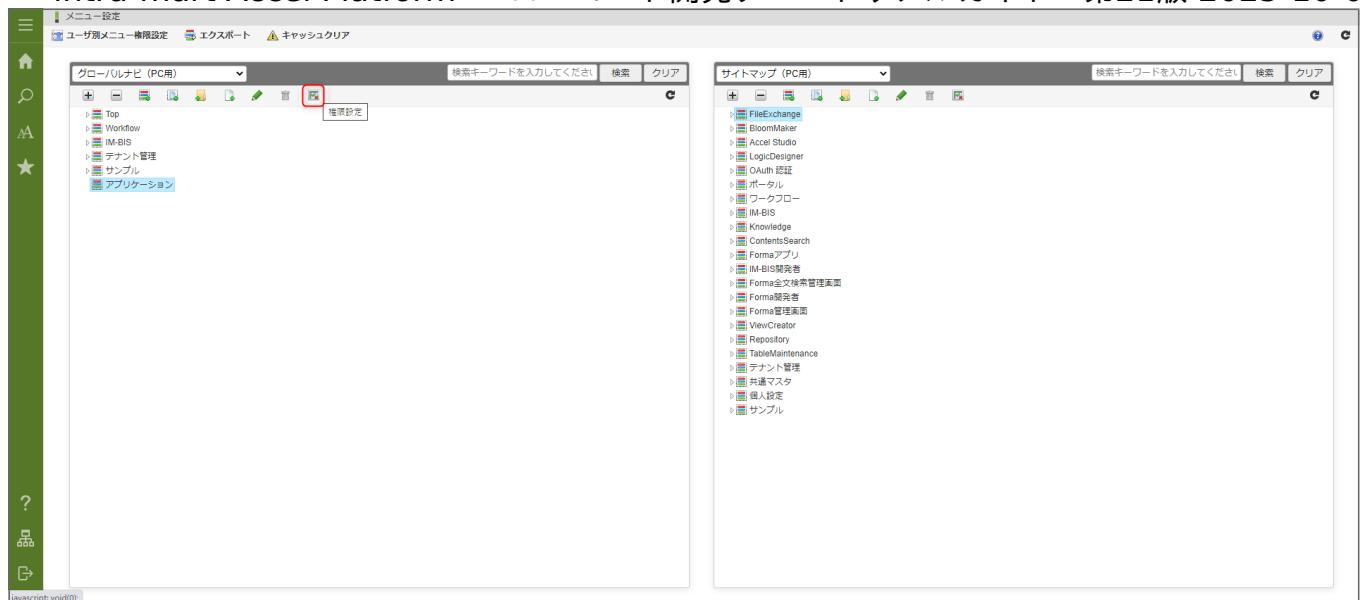
グローバルナビ（PC用）の をクリックし「メニューグループの新規作成」ダイアログを表示します。

The screenshot shows the 'メニュー設定' (Menu Setting) screen with two main sections: 'グローバルナビ (PC用)' and 'サイトマップ (PC用)'. The 'グローバルナビ (PC用)' section has a new menu group icon highlighted with a red box. The 'サイトマップ (PC用)' section shows a list of site map items.

「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ID | application |
| メニュー名 | アプリケーション |

作成したメニューを選択して 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。

権限設定 (グローバルナビ (PC用))

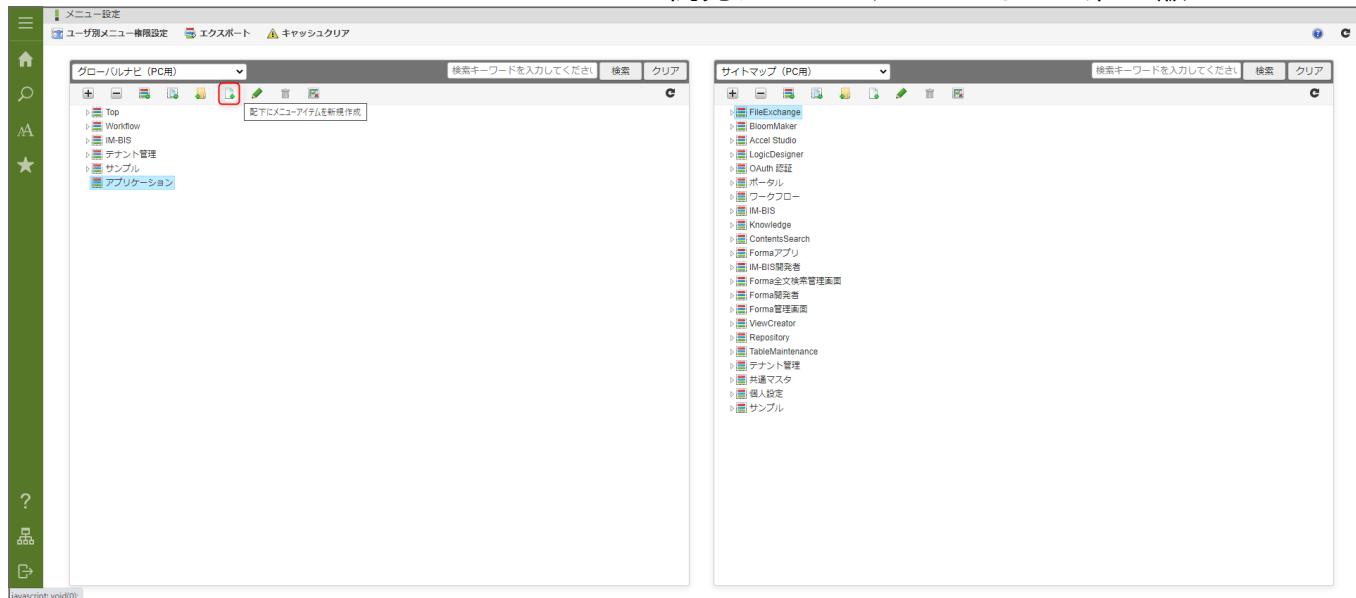
アクションの種類 全てのアクション ▾ 権限設定を開始する

| リソース | アクション | 認証 | | 組織 | | ロール | | メニュー管理 | 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 | ジョブスケジューラ管理者 | IM共通マスター管理者 | IMマネジメント |
|----------|---------------|--------|---------|--------|-------|---------|-------|--------|-----------|----------|--------|----------|--------------|-------------|----------|
| | | 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 | | | | | | | | |
| メニュー群 | グローバルナビ (PC用) | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | ▼ | |
| | Top | 管理 > | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | ☒☒ | |
| 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」配下にのみメニューの設定が可能です。

「アプリケーション」メニュー群を選択して 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 <table border="1" style="width:100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align:left; padding: 2px;"><input type="checkbox"/></th> <th style="text-align:left; padding: 2px;">キー</th> <th style="text-align:left; padding: 2px;">値</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;"><input type="checkbox"/></td> <td style="padding: 2px;"></td> <td style="padding: 2px;"></td> </tr> </tbody> </table> | | <input type="checkbox"/> | キー | 値 | <input type="checkbox"/> | | |
| <input type="checkbox"/> | キー | 値 | | | | | | |
| <input type="checkbox"/> | | | | | | | | |
| アイコン画像 | 標準 | <input checked="" type="radio"/> ファイル <input type="radio"/> コンテキストパス配下のURLを入力してください。 ス | | | | | | |

新規作成 戻る

追加した引数入力欄の「キー」に「enquetelId」、「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 * | 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 <table border="1" style="width:100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align:left; padding: 2px;"><input type="checkbox"/></th> <th style="text-align:left; padding: 2px;">キー</th> <th style="text-align:left; padding: 2px;">値</th> </tr> </thead> <tbody> <tr> <td style="padding: 2px;"><input type="checkbox"/></td> <td style="padding: 2px; border-bottom: 1px solid red;">enquetelId</td> <td style="padding: 2px; border-bottom: 1px solid red;">tutorial_es_enquete</td> </tr> </tbody> </table> | | <input type="checkbox"/> | キー | 値 | <input type="checkbox"/> | enquetelId | tutorial_es_enquete |
| <input type="checkbox"/> | キー | 値 | | | | | | |
| <input type="checkbox"/> | enquetelId | tutorial_es_enquete | | | | | | |
| アイコン画像 | 標準 | <input checked="" type="radio"/> ファイル <input type="radio"/> コンテキストパス配下のURLを入力してください。 ス | | | | | | |

新規作成 戻る

「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ID * | tutorial_enquete_totalize | | | | | | |
| メニューアイテム名 * | <table border="1"> <tr> <td>英語 *</td> <td>従業員満足度調査集計画面</td> </tr> <tr> <td>日本語</td> <td>従業員満足度調査集計画面</td> </tr> <tr> <td>中国語 (中華人民共和国)</td> <td>従業員満足度調査集計画面</td> </tr> </table> | 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 |
| 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 <table border="1"> <tr> <td><input type="checkbox"/> キー</td> <td>値</td> </tr> <tr> <td><input type="checkbox"/></td> <td></td> </tr> </table> | <input type="checkbox"/> キー | 値 | <input type="checkbox"/> | | | |
| <input type="checkbox"/> キー | 値 | | | | | | |
| <input type="checkbox"/> | | | | | | | |
| アイコン画像 | <table border="1"> <tr> <td>標準</td> <td><input checked="" type="radio"/> ファイルパス</td> <td>コンテキストパス配下のURLを入力してください。</td> </tr> <tr> <td></td> <td>ス</td> <td></td> </tr> </table> | 標準 | <input checked="" type="radio"/> ファイルパス | コンテキストパス配下のURLを入力してください。 | | ス | |
| 標準 | <input checked="" type="radio"/> ファイルパス | コンテキストパス配下のURLを入力してください。 | | | | | |
| | ス | | | | | | |
| <input type="button" value="新規作成"/> | | | | | | | |

追加した引数入力欄の「キー」に「enquetetId」、「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 ＊ | <table border="1"> <tr> <td>英語 ＊</td> <td>従業員満足度調査集計画面</td> </tr> <tr> <td>日本語</td> <td>従業員満足度調査集計画面</td> </tr> <tr> <td>中国語 (中華人民共和国)</td> <td>従業員満足度調査集計画面</td> </tr> </table> | 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 |
| 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 | <p>+ 行追加 - 選択行削除</p> <table border="1"> <thead> <tr> <th>□ キー</th> <th>値</th> </tr> </thead> <tbody> <tr> <td>enqueteid</td> <td>tutorial_es_enquete</td> </tr> </tbody> </table> | □ キー | 値 | enqueteid | tutorial_es_enquete | | |
| □ キー | 値 | | | | | | |
| enqueteid | tutorial_es_enquete | | | | | | |
| アイコン画像 | <p>標準 <input checked="" type="radio"/> ファイル <input type="radio"/> コンテキストパス配下のURLを入力してください。</p> <p>ス</p> | | | | | | |

「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。

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もメニュー設定と同様に、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。

The screenshot shows the 'Application Management' interface with a search bar and various navigation tabs at the top. A red box highlights the '認可設定' (Access Control) button in the top right corner of the main content area. Below it, a table lists numerous resources, each with a status icon, a name, and a timestamp indicating when the tenant last accessed it.

| リソース | 状況 | 名前 | 最終アクセス |
|---------|----|-------------------------------|----------------------------|
| 画面 | OK | 【チュートリアル】アンケート集計画面 | tenant 1時間前 |
| 画面 | OK | 【チュートリアル】アンケート回答画面 | tenant 23時間前 |
| 画面URL | OK | 【チュートリアル】アンケート回答 | tenant 0時間前 |
| 画面URL | OK | 【チュートリアル】アンケート回答 | tenant 0時間前 |
| APIのURL | OK | GET tutorial_enquete/list | tenant 2024/12/25 15:37:29 |
| APIのURL | OK | POST tutorial_enquete/answer | tenant 2024/12/25 15:38:29 |
| APIのURL | OK | GET tutorial_enquete/question | tenant 2024/12/25 15:38:29 |
| ロジックフロー | OK | 【チュートリアル】アンケート回答取得 | tenant 2024/12/25 15:35:29 |
| ロジックフロー | OK | 【チュートリアル】アンケート回答登録 | tenant 2024/12/24 17:48:17 |
| ロジックフロー | OK | 【チュートリアル】アンケート情報取得 | tenant 2024/12/24 17:01:35 |
| エンティティ | OK | アンケート回答 | tenant 2024/12/24 16:32:03 |
| エンティティ | OK | アンケート設問 | tenant 2024/12/23 17:57:52 |

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 'Access Control' configuration page for the 'Survey Application'. It features a header with checkboxes for '選択中の認可を許可' (Allow selected access) and '選択中の認可を禁止' (Deny selected access), and a note about changing settings. The main area is a grid where rows represent resources and columns represent users and roles. The '認証済みユーザ' (Authenticated User) column is highlighted in yellow.

| リソース | 認証 | 組織 | ロール | ゲストユーザー | 認証済みユーザー | 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 | シヨブスケシユラ管理者 | IM共通マスター管理者 | IM共通マスター通用管理者 | ポータル管理者 | IM-Workflow管理者 | IM-Workflow通用管理者 | 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 | X | ✓ | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 【チュートリアル】アンケート回答 | X | ✓ | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 【APIのURL】 | | | | | | | | | | | | | | | | | | | | | | |
| GET tutorial_enquete/list | X | ✓ | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| POST tutorial_enquete/answer | X | ✓ | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| GET tutorial_enquete/question | X | ✓ | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

「更新」をクリックし、認可設定の更新を行います。

The screenshot shows a configuration page for an application named "【チュートリアル】アンケートアプリケーション - 認可設定". It displays a grid of permissions for different roles across various URLs. A modal dialog box is open in the center, asking for confirmation to update application permission settings.

| リソース | 認可 | | 拒否 | | ロール | | | | | | | | | | | | | | |
|----------------------------------|---------|----------|--------|-------|---------|-------|-----------|-----------|----------|--------|----------|--------------|--------------|--------------|---------|-------------------|-------------------|-----------------|-----------------|
| | ゲストユーザー | 認証済みユーザー | サンプル会社 | その他会社 | テナント管理者 | 認可管理者 | メニュー運用管理者 | 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 運用管理者 | IM-Workflow 運用管理者 | IM-Workflow 監査者 | IM-Workflow ユーザ |
| 【画面URL】 | [確認] | | | | | | | | | | | | | | | | | | |
| 【チュートリアル】アンケート実行画面 | × | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| 【チュートリアル】アンケート回答 | ✗ | ✓ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | ✗ | |
| 【API(URL】 | [確認] | | | | | | | | | | | | | | | | | | |
| 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

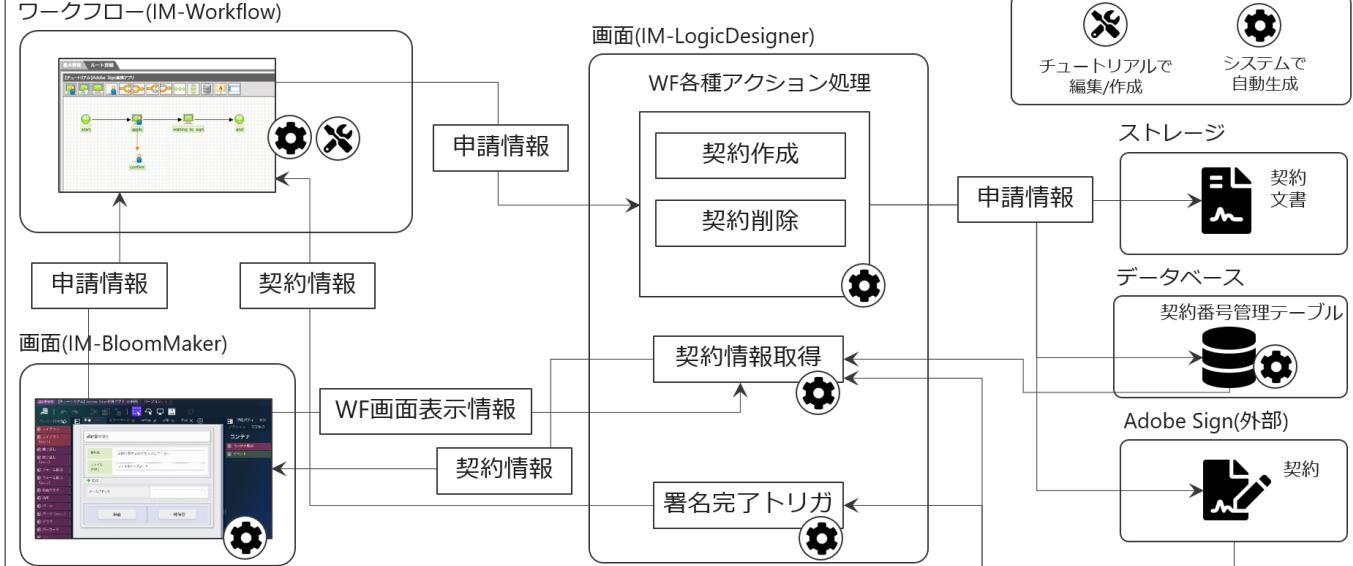
IM-Sign 連携アプリケーションの仕様

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。 |

The screenshot shows the 'Contract Submission' screen with the following fields:

- 契約名:** Contract Name input field.
- ファイル(PDF):** File selection button.
- + 追加:** Add button for multiple attachments.
- メールアドレス:** Email address input field.
- 申請:** Submit button.
- 一時保存:** Temporary Save button.

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 | @ |

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。
 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」](#)



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」



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」を利用することで、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。

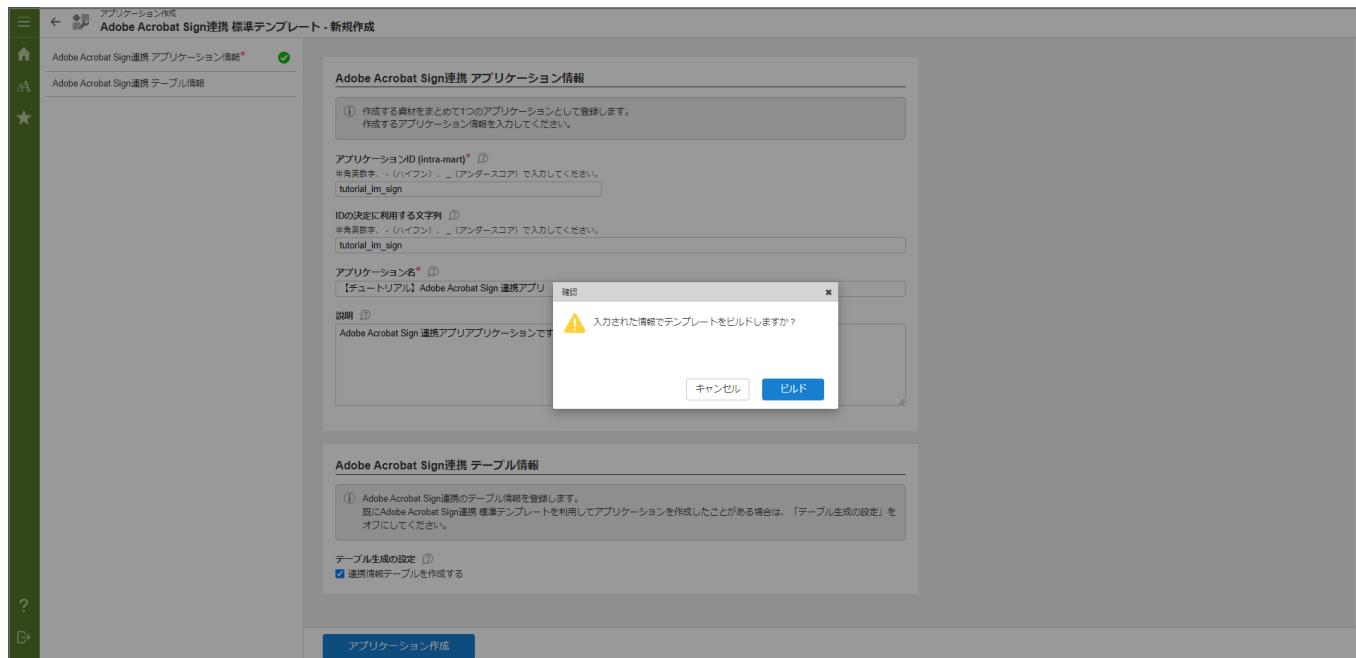
| ラベル | 入力値 |
|------------|-------------------------------------|
| アプリケーションID | tutorial_im_sign |
| アプリケーション名 | 【チュートリアル】 Adobe Acrobat Sign 連携アプリ |
| 説明 | Adobe Acrobat Sign 連携アプリアプリケーションです。 |

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。

| テンプレートビルトショット - 詳細 | | | |
|---------------------|--|-----------|-------------------------------|
| ビルトショットID | 8glmb90bgqlq8h11 | 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 | aoyagi |
| 作成日時 | 2023/06/01 17:27:24 | 詳細 | |
| 出力日時 | メッセージ | | |
| 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 | 一時ディレクトリに IM-LogicDesigner インポート資材を作成しました。 | | |
| 2023/06/01 17:27:26 | 一時ディレクトリに IM-BloomMaker インポート資材を作成しました。 | | |
| 2023/06/01 17:27:26 | 一時ディレクトリに IM-Workflow インポート資材を作成しました。 | | |
| 2023/06/01 17:27:26 | アプリケーションで利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 資材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 資材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 資材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

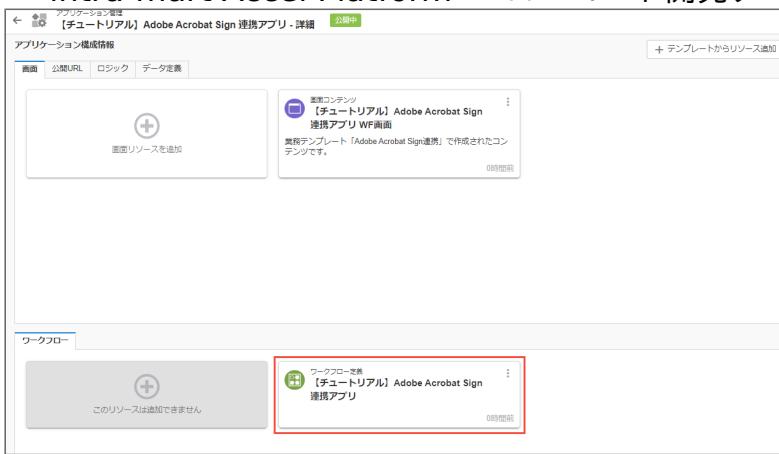
テンプレートから作成したワークフローは、利用可能なユーザやロールが割り当てられていないため、ワークフローの設定を行います。

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。

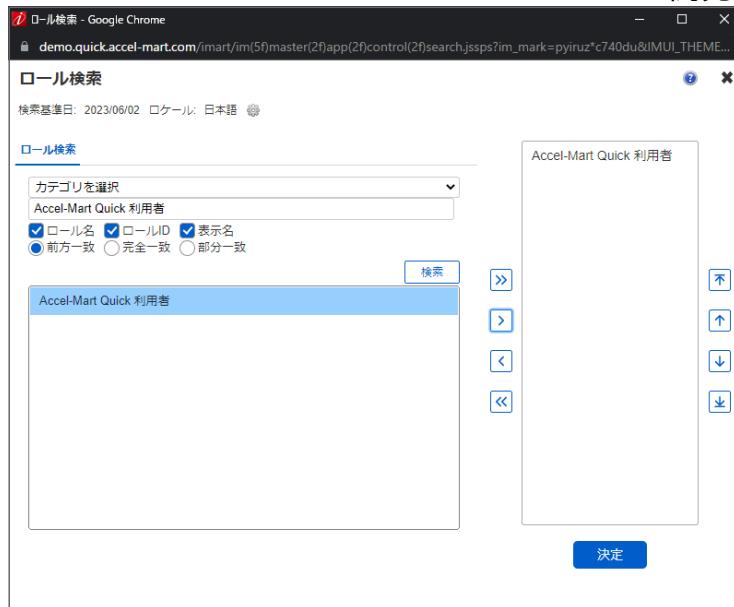
「apply」のノードアイコンをクリックします。

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。



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。



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。

The screenshot shows the 'Contract Submission' (契約書の送付) page. On the left is a vertical toolbar with icons for three, home, font size, and star. The main area has fields for 'Contract Name' (契約名) containing 'チュートリアル用契約', 'File (PDF)' (ファイル(PDF)) with a download link to 'テンプレート用契約書.pdf (66.8KB)' and a delete icon, and 'Email Address' (メールアドレス) with the value 'XXXXX@XXXX.XXXX'. There are 'Add' (追加) and 'Save Draft' (一時保存) buttons at the bottom.

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。

The screenshot shows a form titled '【チュートリアル】Adobe Sign連携アプリ - apply'. It includes fields for '申請者' (Kōsho), '申請基準日' (Kōsho基準日), '案件名' (Case Name), '担当組織' (Organization), '優先度' (Priority), and 'コメント' (Comment). There are sections for '添付ファイル' (Attached Files) and '根回し' (Follow-up). A large blue '申請' (Apply) button is at the bottom.

i 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ー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。



i 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。



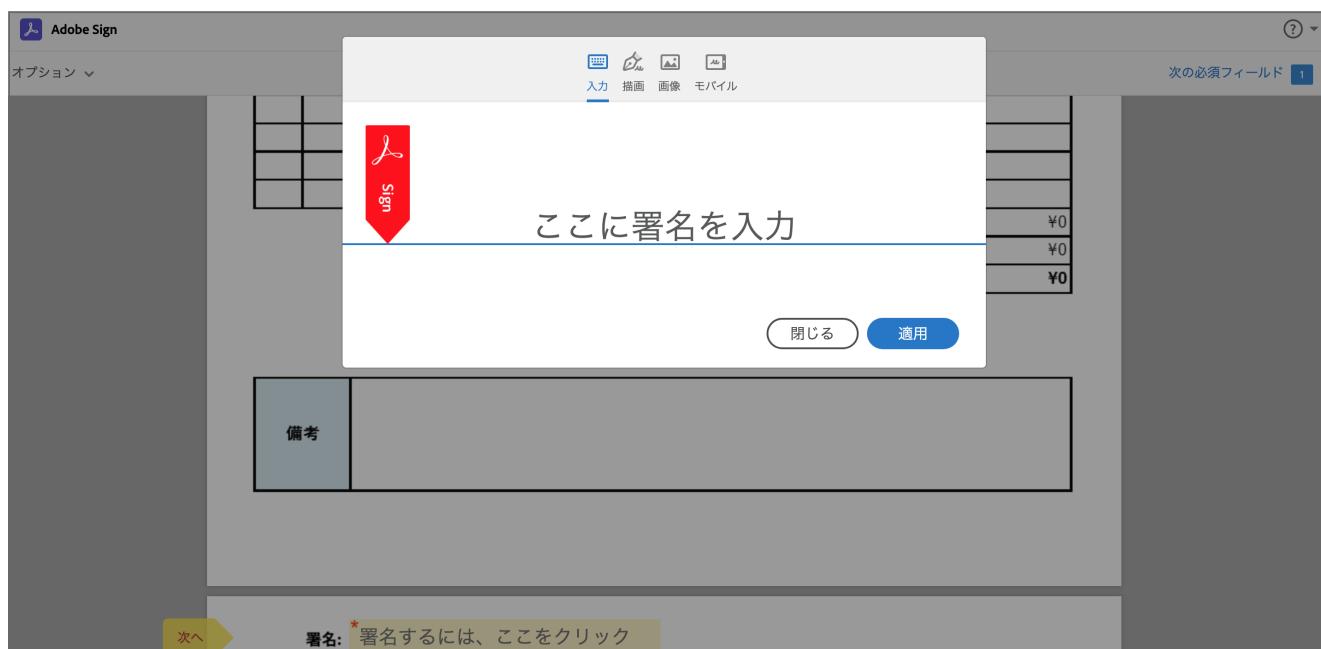
契約相手による電子サイン実施

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

@

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](#)」で作成した、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

ToDoアプリケーションに追加機能を作成する

「[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](#)」で作成した、簡易的なToDoアプリケーションに「[intra-mart Accel Platform Accel Studio テンプレートカタログ](#)」から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。

- Accel Studio における処理の繰り返し
- IM-BloomMaker における変数の利用方法
- IM-BloomMaker でのコンテナページの利用方法

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 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 好きな映画ジャンル調査 | 2021/01/01 | 2022/01/01 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 好きな料理アンケート | 2019/01/01 | 2020/01/01 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 休日の過ごし方 | 2021/10/22 | 2021/11/05 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 理解度チェック | 2021/12/10 | 2021/11/04 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

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も前にならないように入力チェックを行います。



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 年 /月/日

終了日 年 /月/日

設問

| | | |
|------|-----|-----|
| 設問内容 | 左項目 | 右項目 |
| 設問内容 | 左項目 | 右項目 |
| 設問内容 | 左項目 | 右項目 |

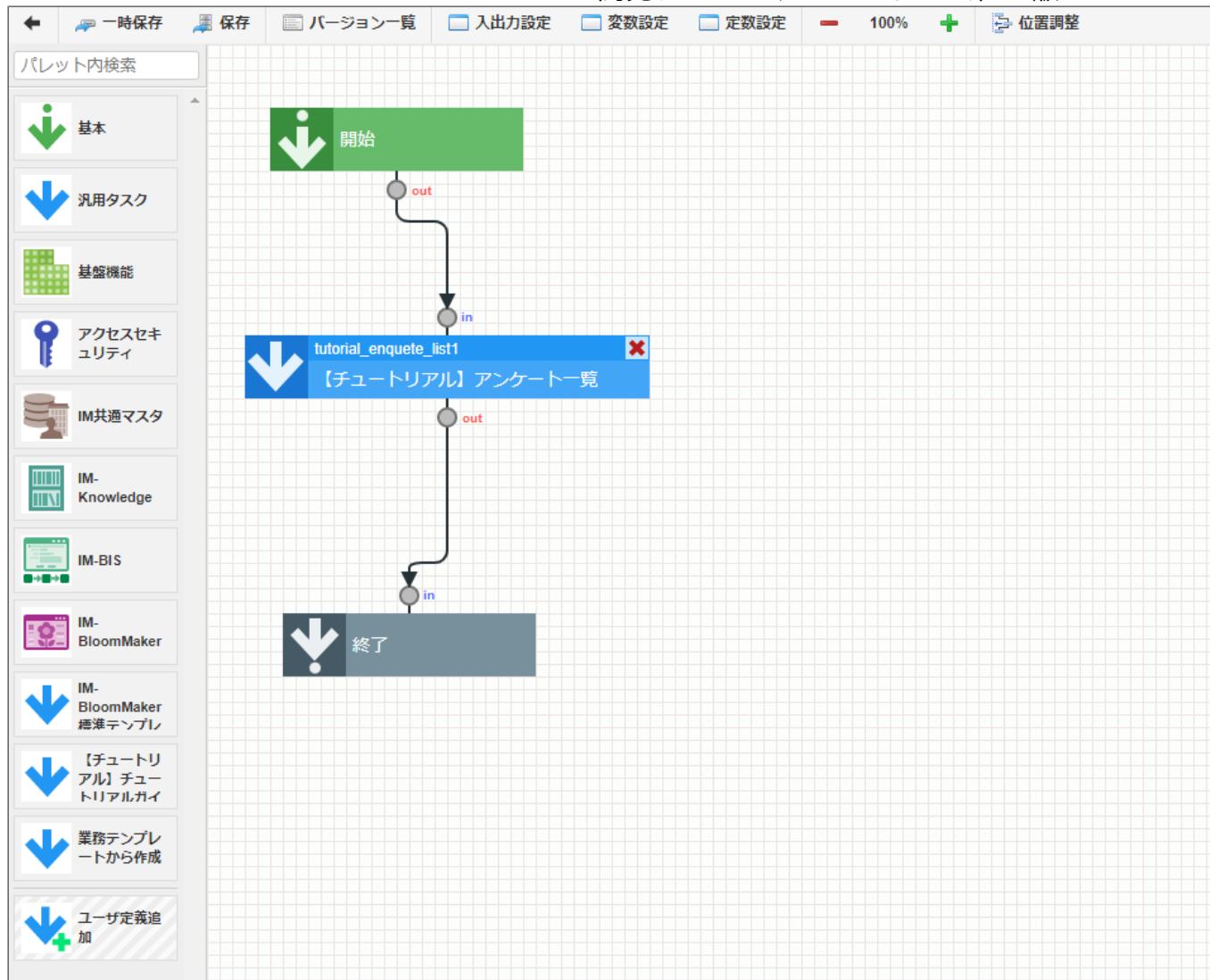
アンケート一覧、登録処理の業務ロジックの作成

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。

業務ロジックの出力値を設定します。

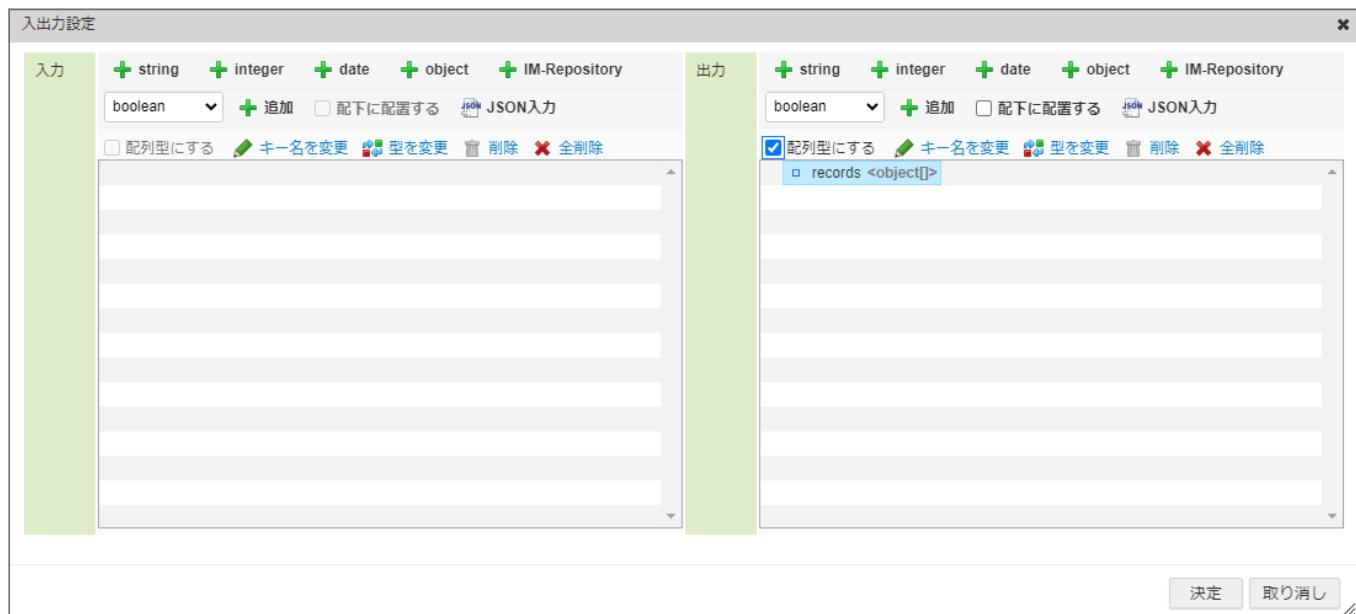
一覧の取得は検索等を行わずに全てのアンケートを表示するため、出力値のみを設定します。

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。



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」を選択します。

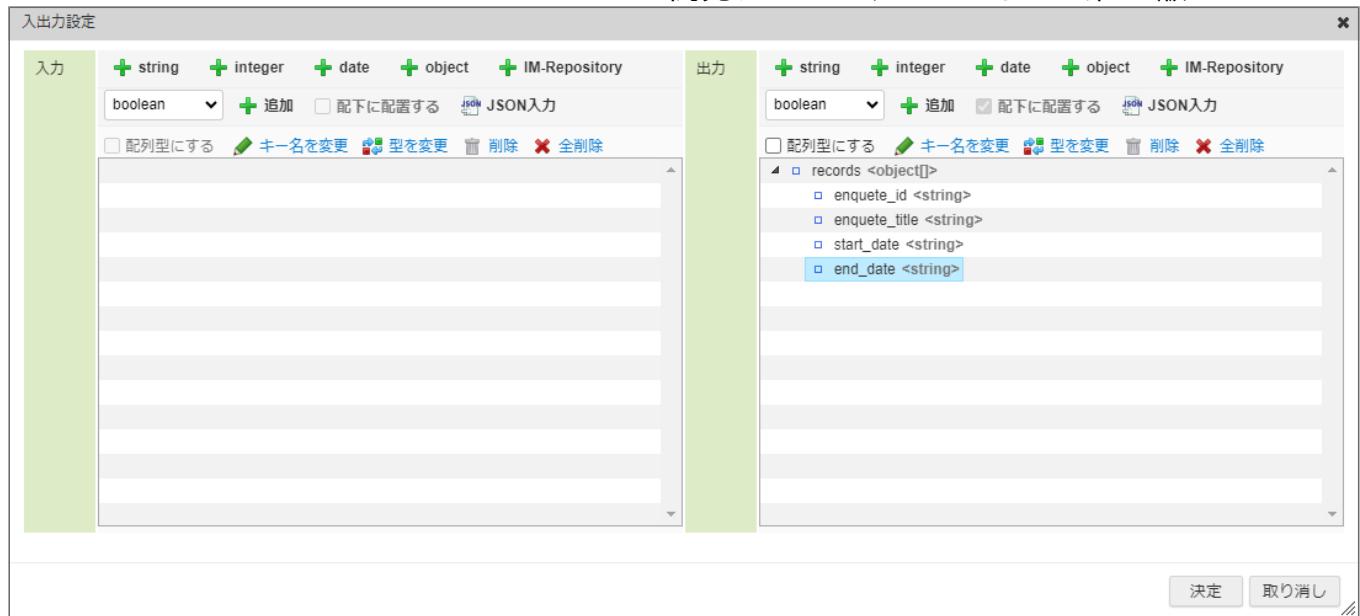
「出力」 - 「+string」をクリックし「records」配下に、「string1」という値を追加して、キー名を「enquete_id」と変更します。



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 |



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": [
  {
    "enqueue_id": "",
    "enqueue_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_enqueue_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】チュートリアルカテゴリ |

クエリに以下のSQLを設定します。

SELECT

```

enquête_id,
enquête_title,
start_date,
end_date
FROM
tutorial_enquête_info_app

```

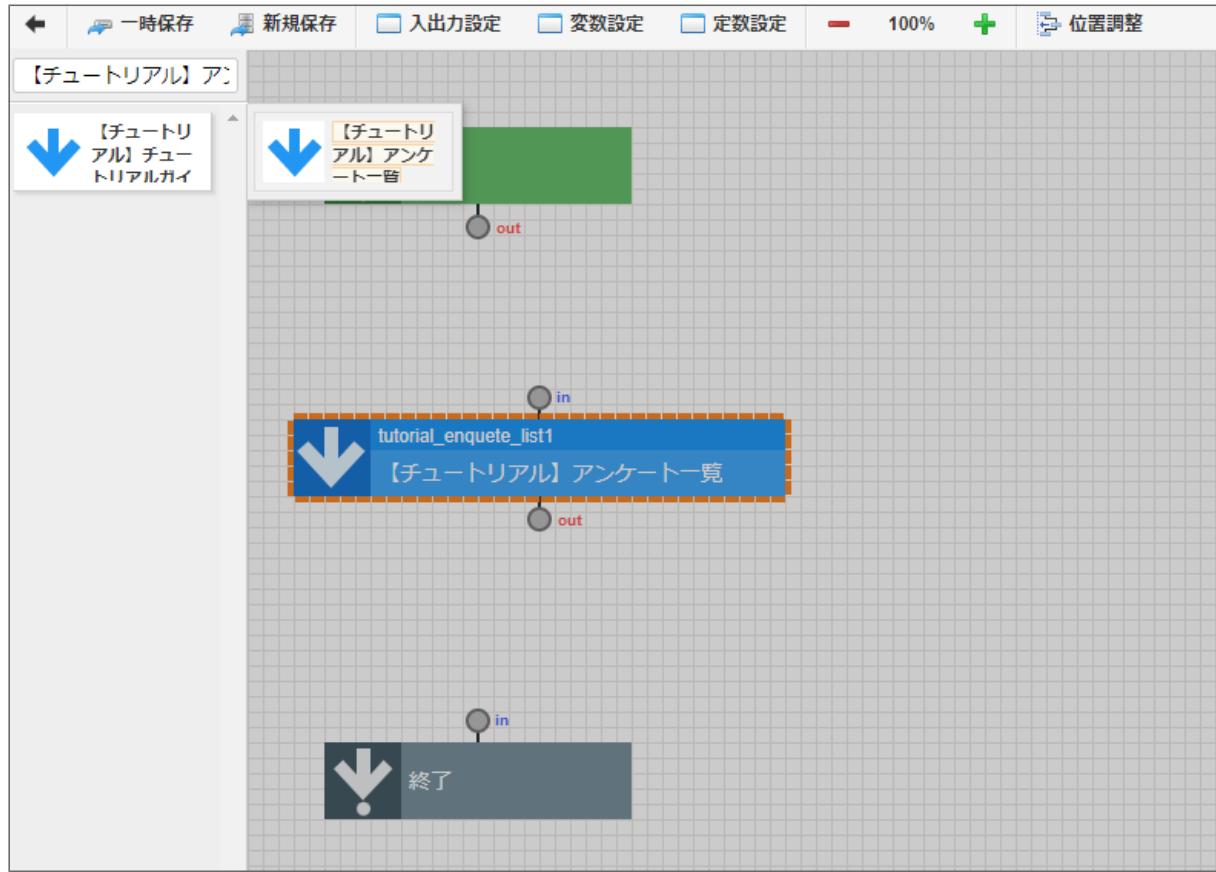
「データ定義を取得する」をクリックし、タスクの出力値を設定します。

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

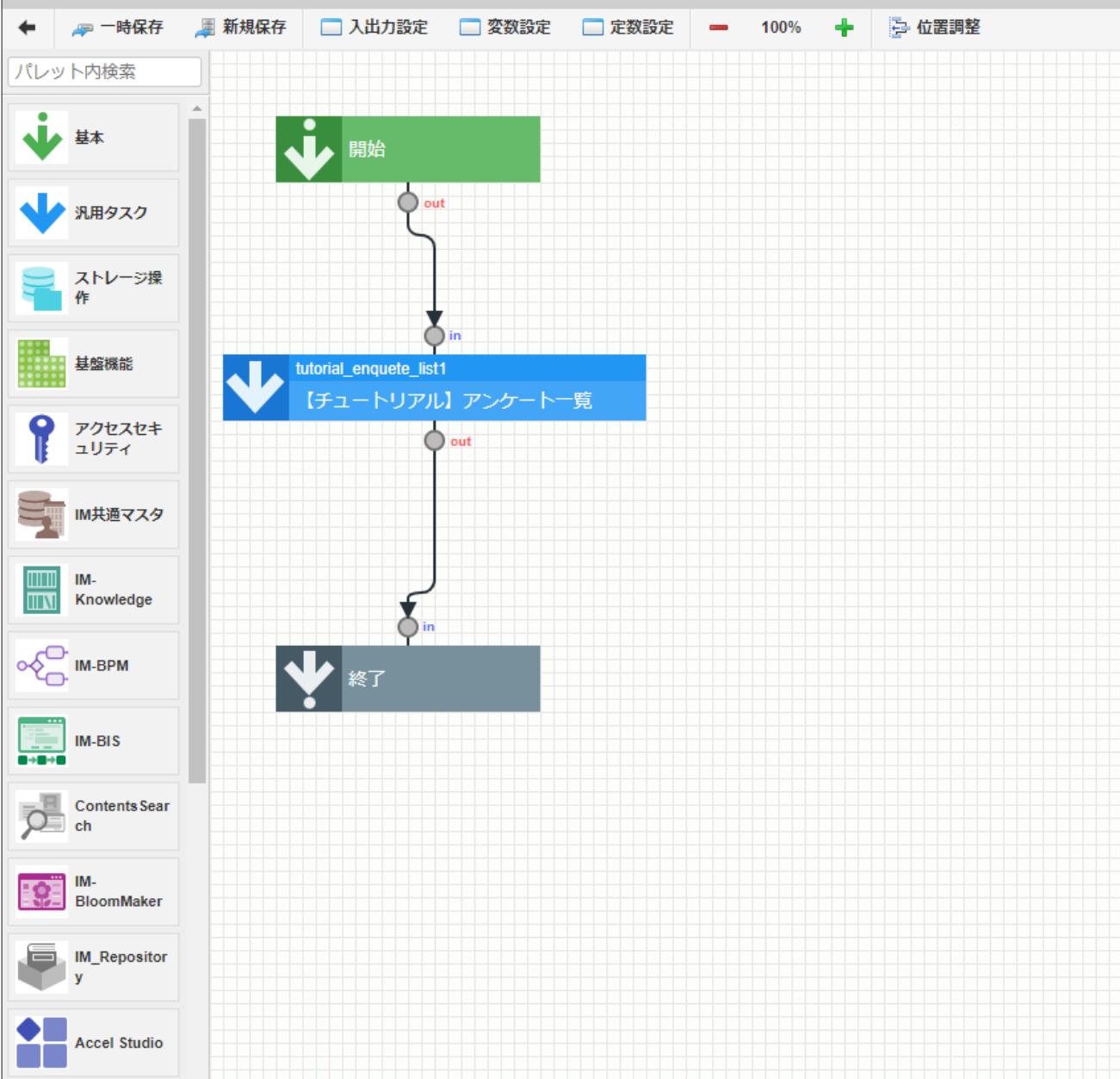
作成した「【チュートリアル】アンケート一覧」タスクをフローに配置し、線を繋げます。

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。



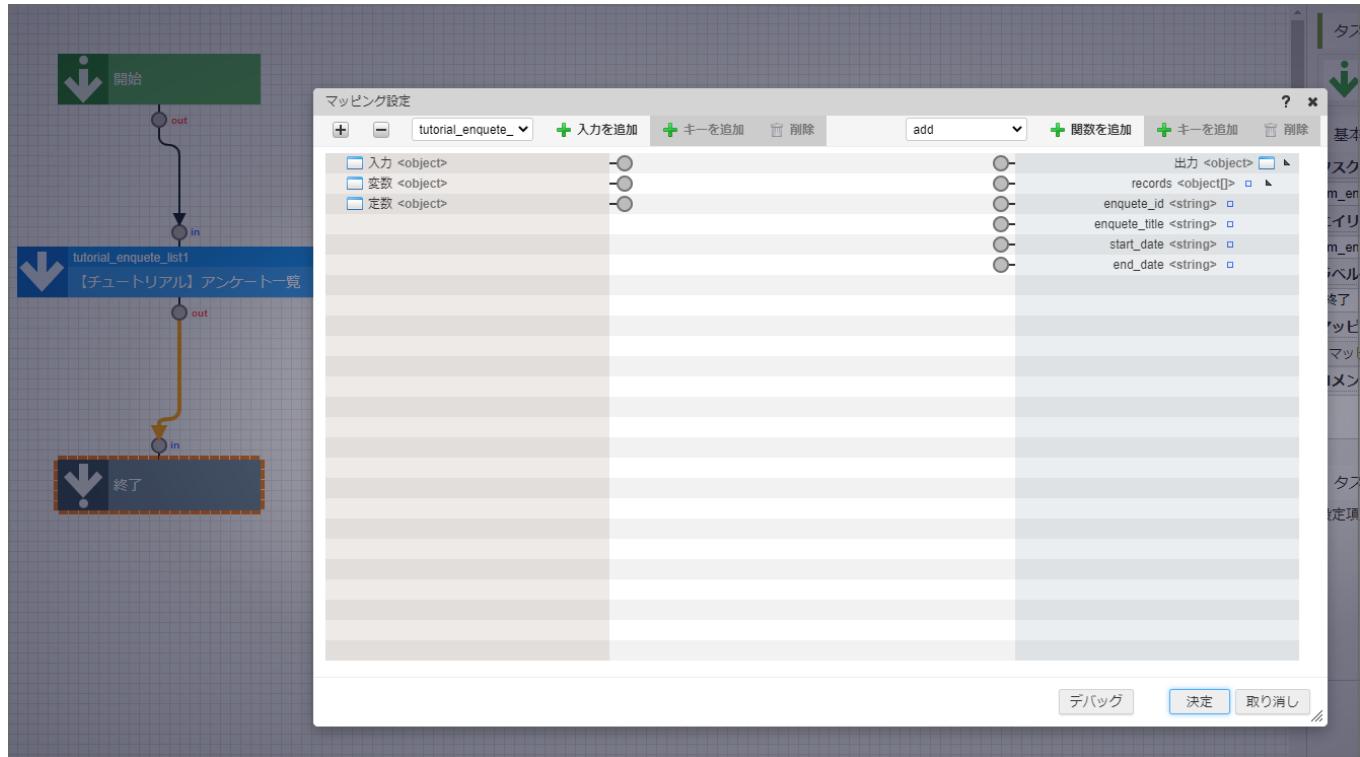
「開始」、「終了」タスクと「【チュートリアル】 アンケート一覧」を線で繋げます。

ロジックフロー定義編集 *



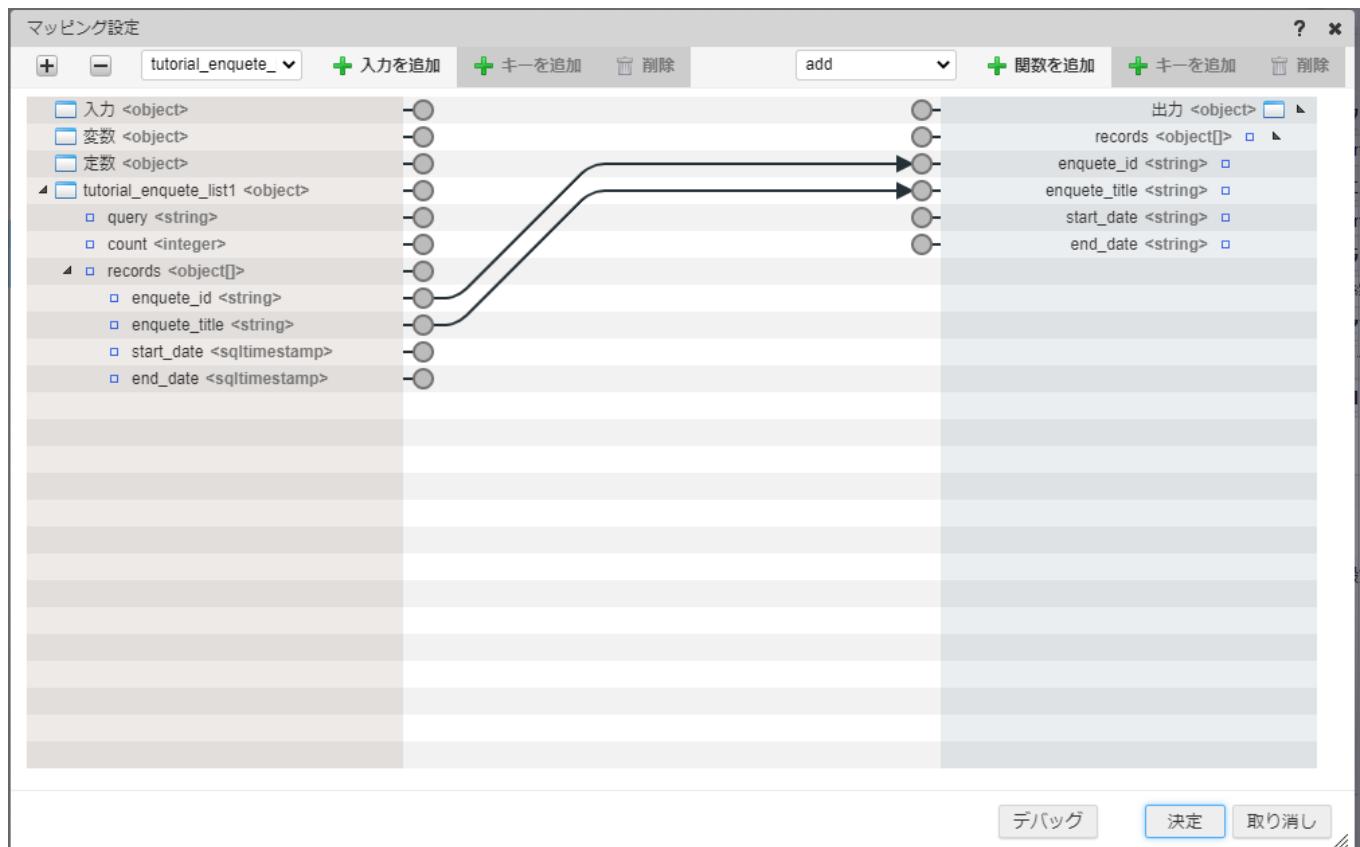
次に、配置したタスクの変数を設定していきます。

「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「enquetelist」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。

定数設定

+ 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 |
|----|------|-----|----|------|
| | | | | |

決定 **取り消し**

「+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 |

定数設定

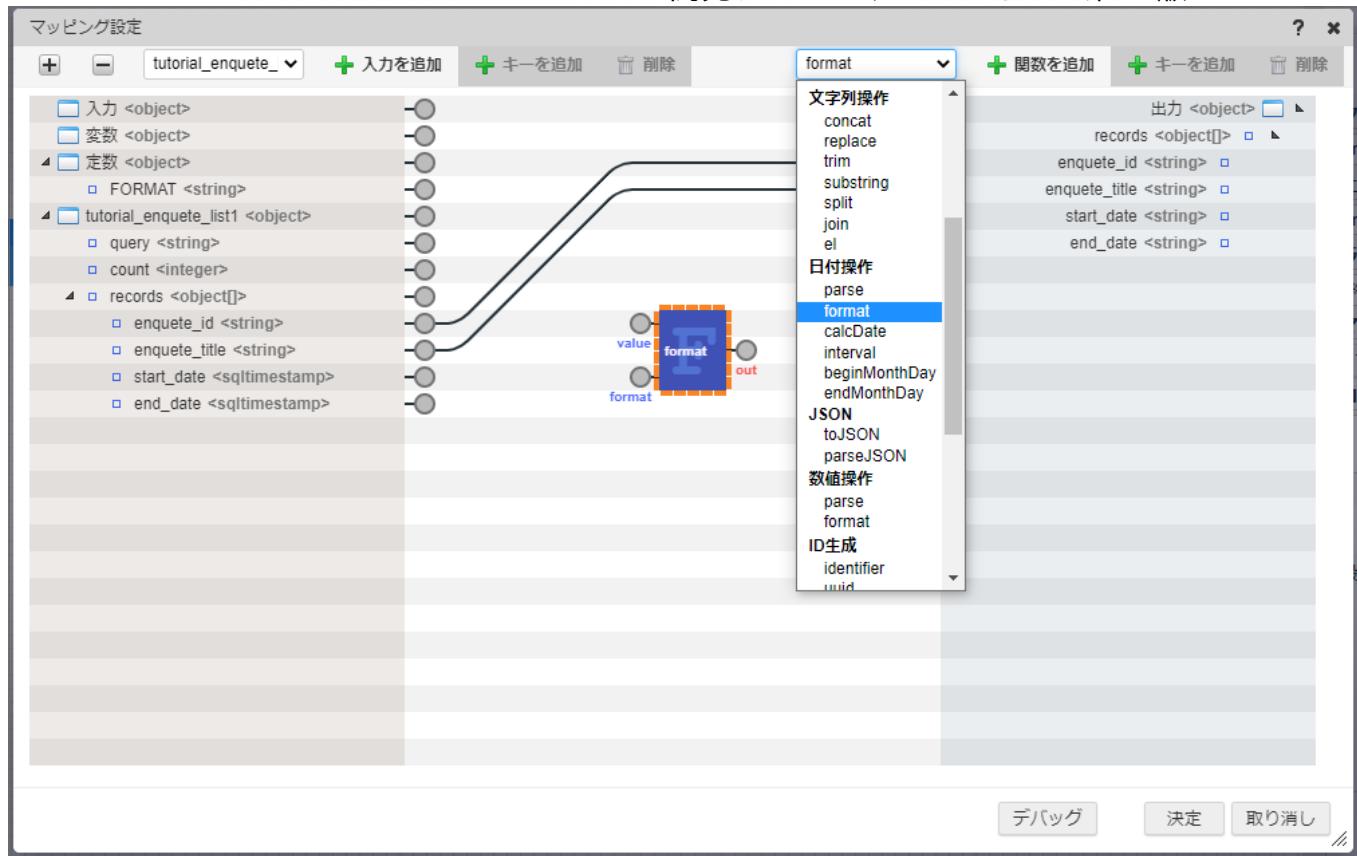
+ 定数を追加 **選択済みの定数を削除**

「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。

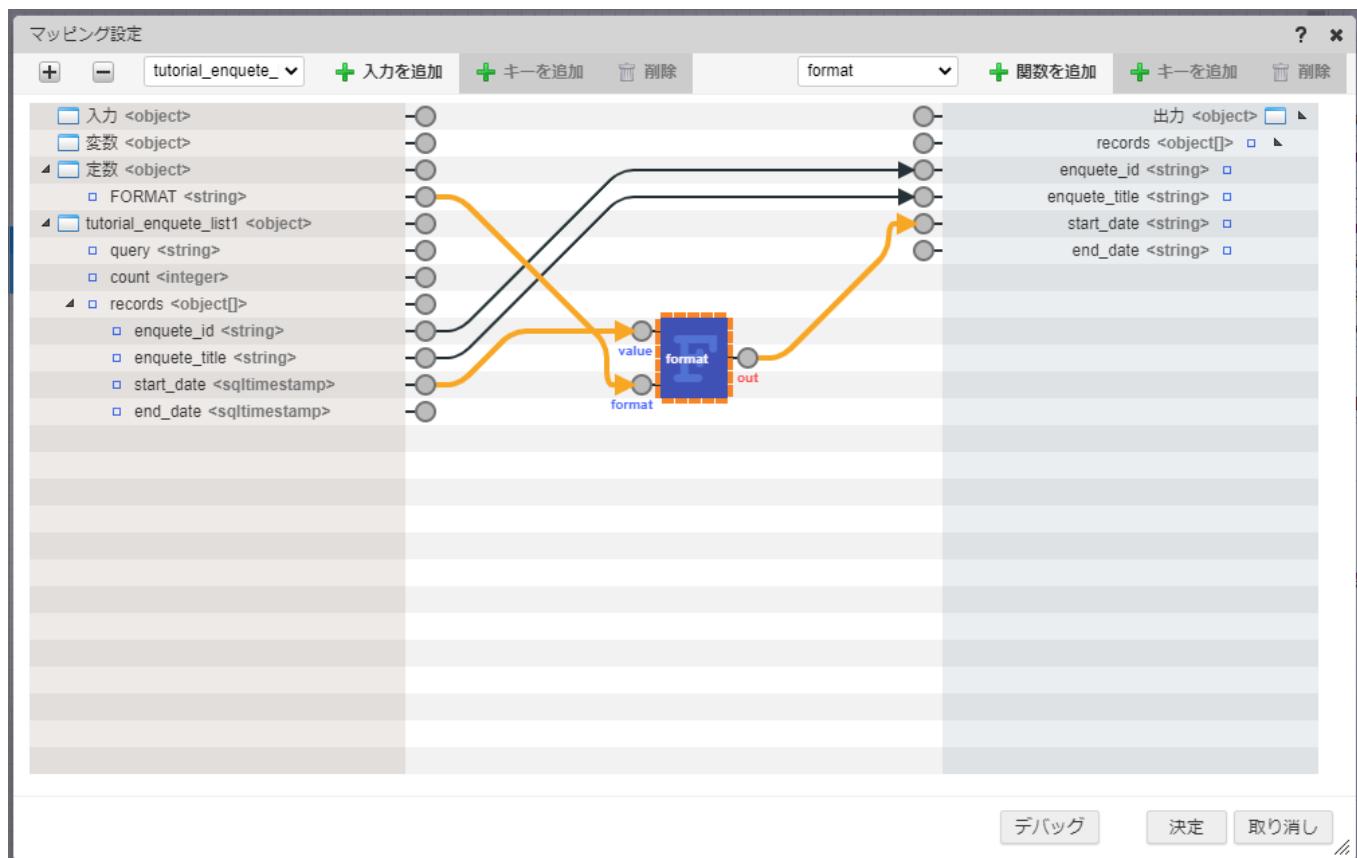
| 選択 | 定数ID | 定数値 | 説明 | エディタ |
|--------------------------|--------|------------|----|---|
| <input type="checkbox"/> | FORMAT | yyyy/MM/dd | |  |

決定 **取り消し**

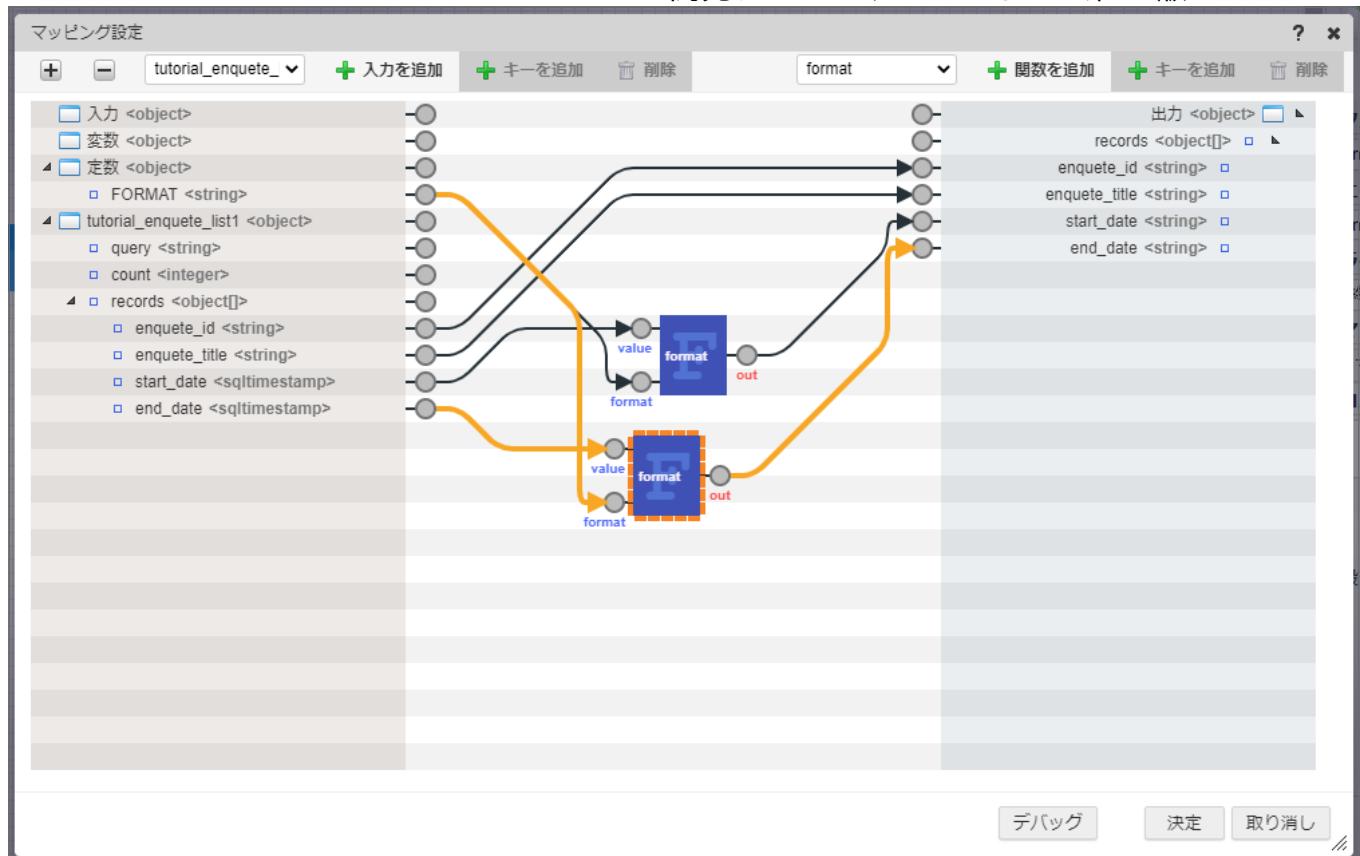
定数を保存したら再度「終了」タスクのマッピング設定を表示し、関数から「日付操作 > format」を選択し、「+関数を追加」で配置します。



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 |
| フロー定義名（標準） | 【チュートリアル】アンケート一覧取得 |
| フローカテゴリ | 【チュートリアル】チュートリアルカテゴリ |

新規保存

| | | | | | | | | | |
|--------------|---|------|--------------------|----|----------------------|-----|--|--------------|--|
| フロー定義ID * | tutorial_enquete_list | | | | | | | | |
| フロー定義名 * | <table border="1"> <tbody> <tr> <td>標準 *</td> <td>【チュートリアル】アンケート一覧取得</td> </tr> <tr> <td>英語</td> <td></td> </tr> <tr> <td>日本語</td> <td></td> </tr> <tr> <td>中国語(中華人民共和国)</td> <td></td> </tr> </tbody> </table> | 標準 * | 【チュートリアル】アンケート一覧取得 | 英語 | | 日本語 | | 中国語(中華人民共和国) | |
| 標準 * | 【チュートリアル】アンケート一覧取得 | | | | | | | | |
| 英語 | | | | | | | | | |
| 日本語 | | | | | | | | | |
| 中国語(中華人民共和国) | | | | | | | | | |
| フローカテゴリ * | <table border="1"> <tbody> <tr> <td>ID *</td> <td>tutorial_category</td> </tr> <tr> <td>名称</td> <td>【チュートリアル】チュートリアルカテゴリ</td> </tr> </tbody> </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 | | | | | | | | | |

決定 取り消し

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、キーナーを「enquetefld」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 | キーナー |
|-----------|--------|-----------------|
| アンケートID | string | enquetefld |
| 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 |



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、キーナー「questionList」を設定してください。

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。

```
{
  "enqueueId": "",
  "enqueueTitle": "",
  "enqueueOverview": "",
  "startDate": null,
  "endDate": null,
  "questionList": [
    {
      "question": "",
      "leftScale": "",
      "rightScale": ""
    }
  ]
}
```

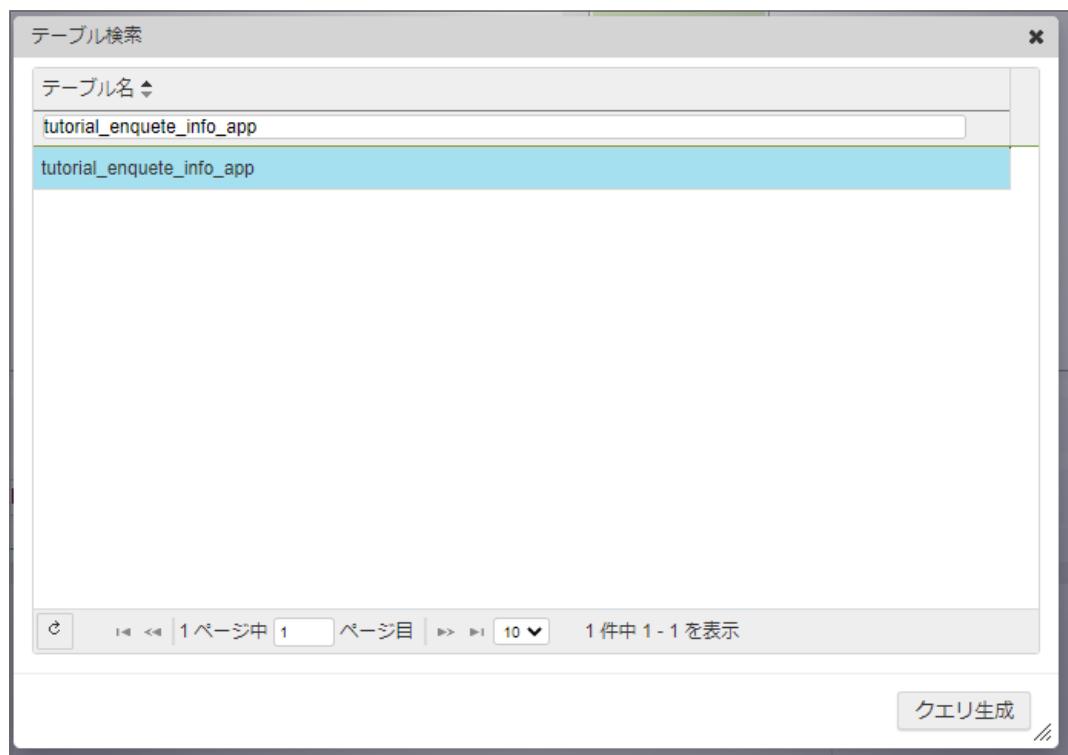
JSON入力する場合は、出力値の「JSON入力」をクリックして「JSON入力画面」を表示してください。また、JSON形式での入力の際には型

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 | キーナ |
|-----------|--------------|-----------------|
| アンケートID | string | enquetelid |
| 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
(/*enquetelid*/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。

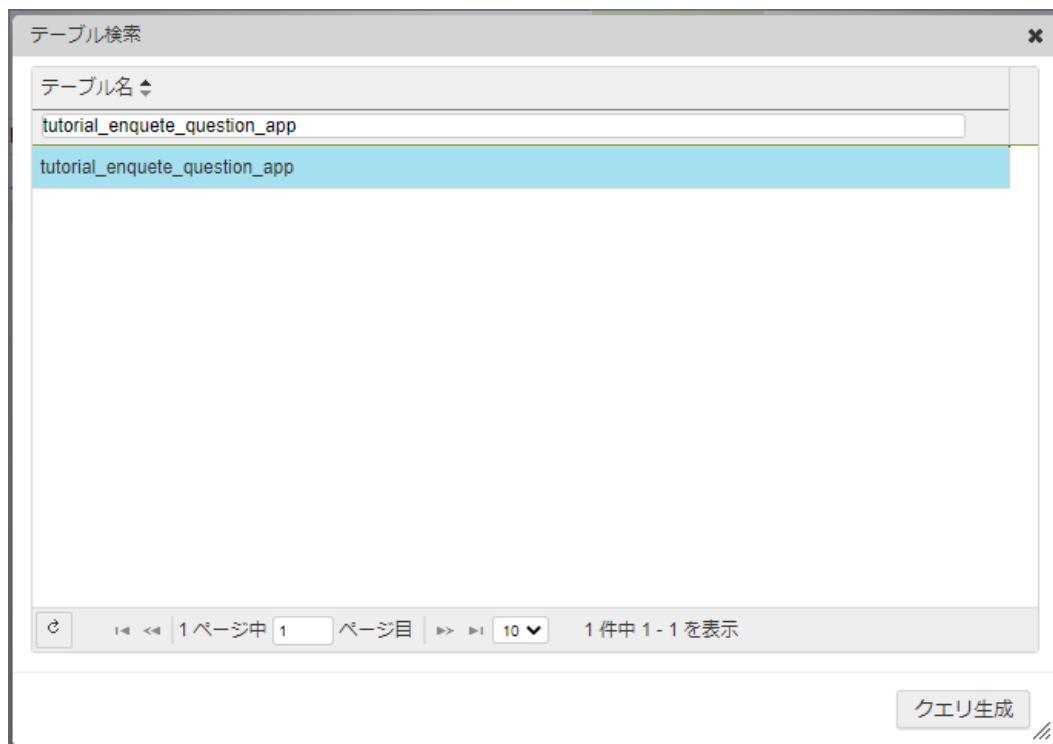
ユーザ定義共通設定を以下のように設定します。

| 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 | キーナ |
|---------|------------|------------|
| アンケートID | string | enquetedId |
| 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

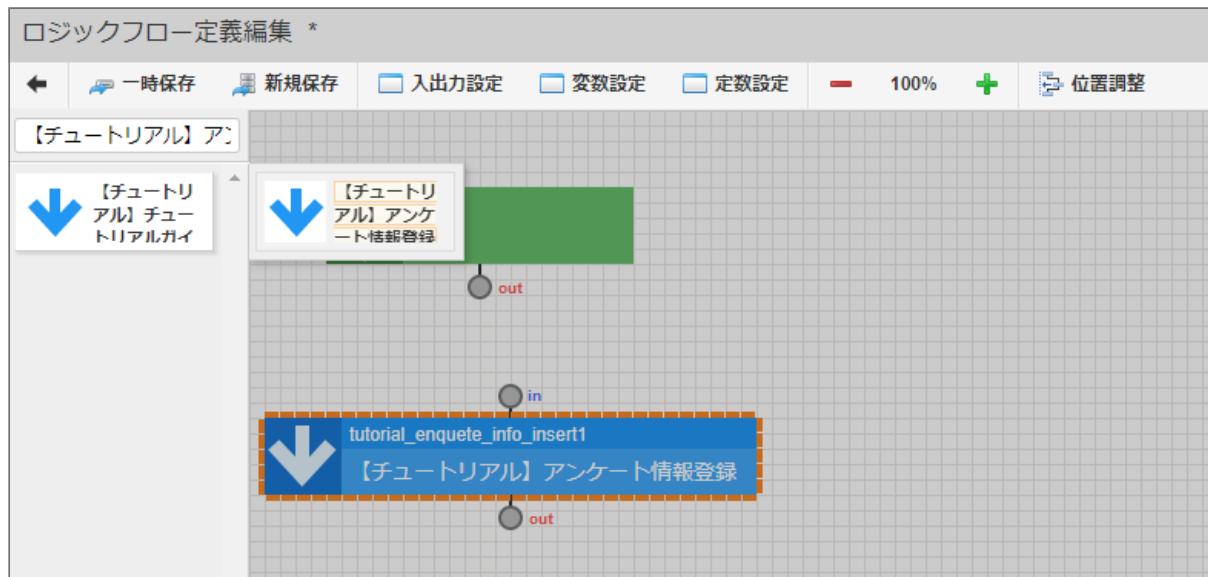
```
INSERT INTO tutorial_enquete_question_app
(enquete_id, question_id, question_no, question, left_scale, right_scale)
VALUES
(/*enquetelid*/null, /*questionId*/null, /*questionNo*/null, /*question*/null, /*leftScale*/null, /*rightScale*/null)
```

「登録」ボタンをクリックし、タスクを登録します。

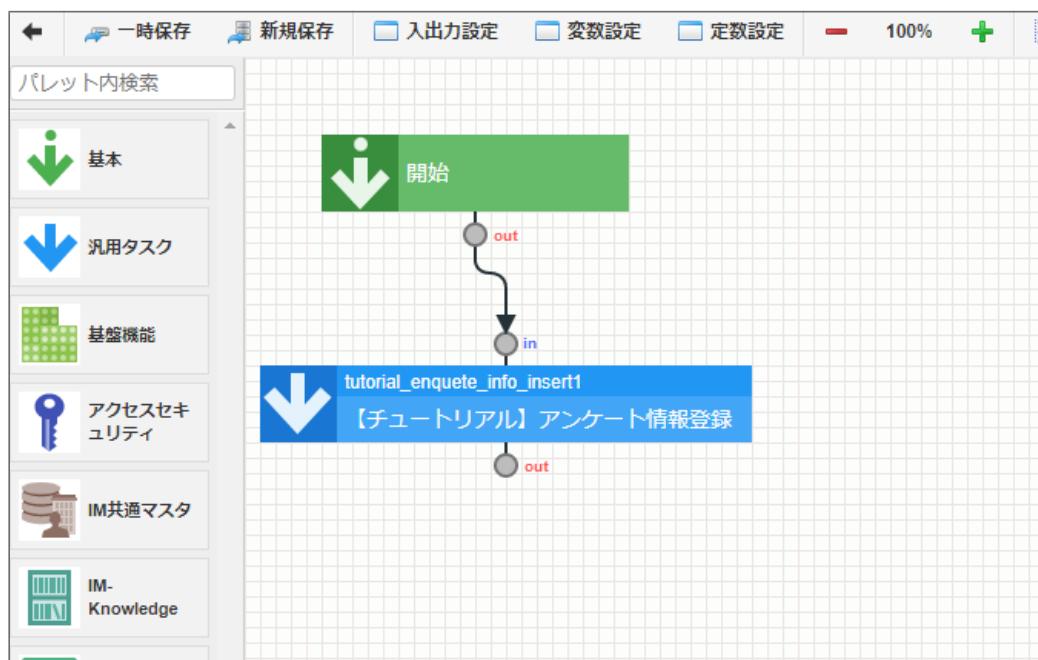
タスクの配置と変数設定

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」を線で繋げます。



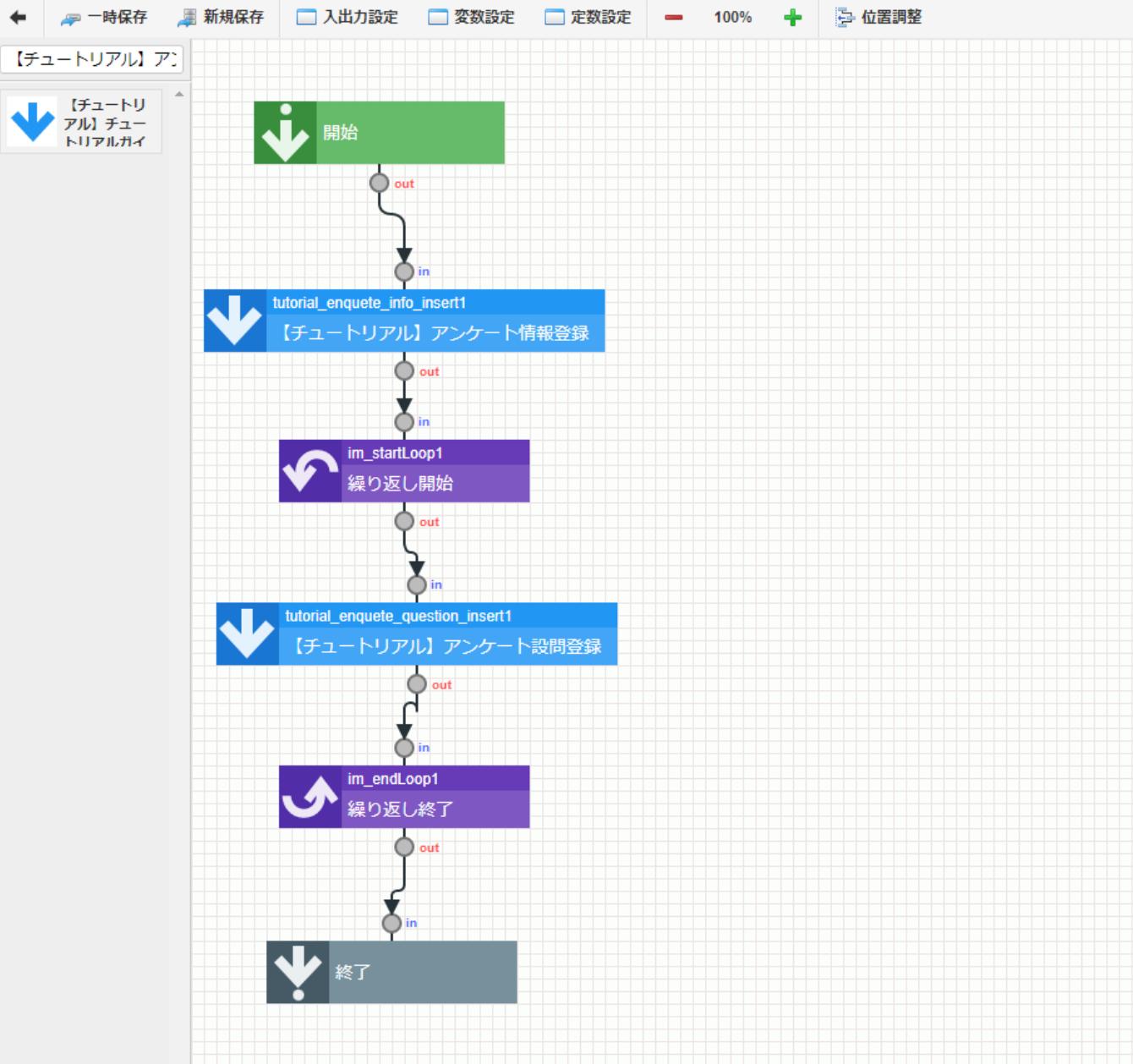
次に、アンケートの設問登録処理を配置します。アンケートの設問は入力値として渡された設問数だけ繰り返して実行します。

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ぎます。

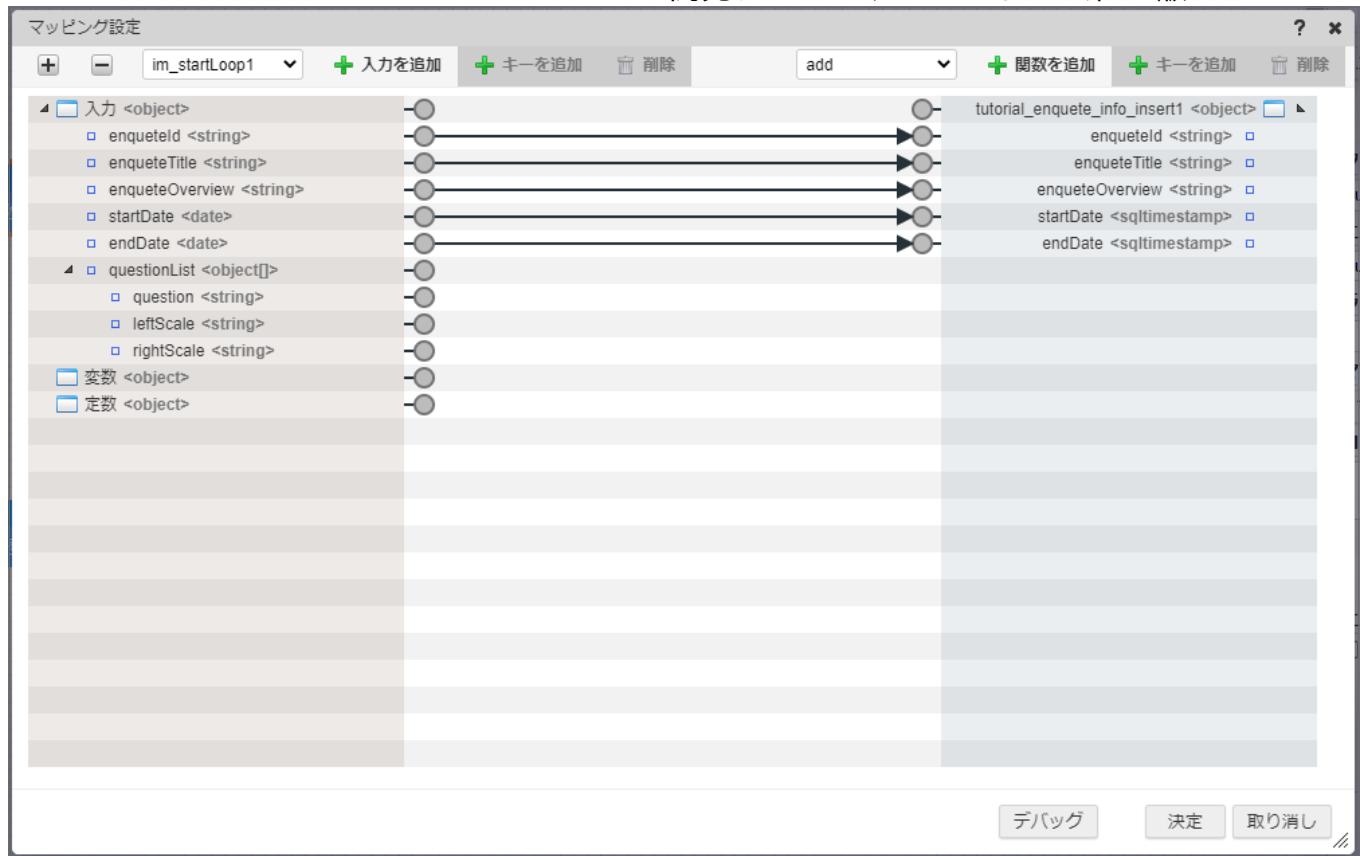
ロジックフロー定義編集 *



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」の同じ値を線で繋げます。

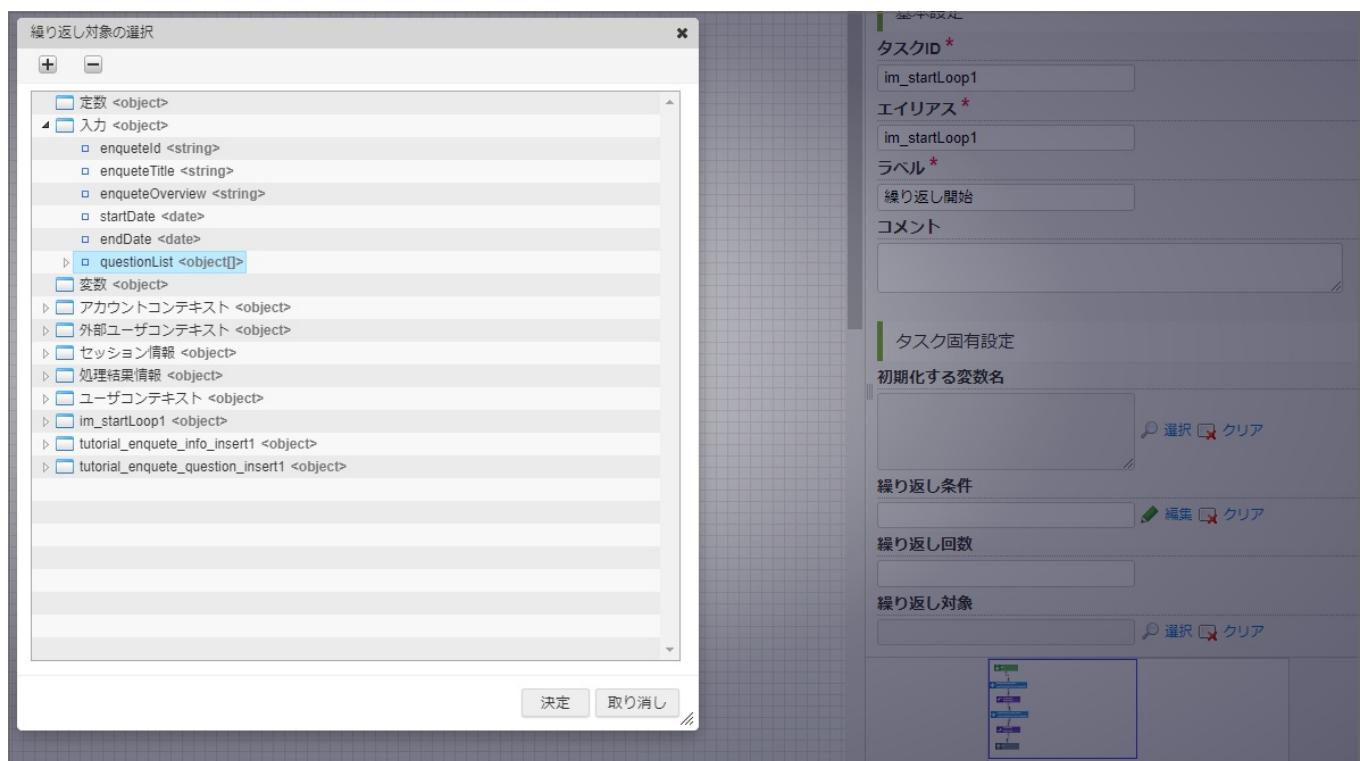


「決定」ボタンでマッピング設定ダイアログを閉じます。

「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。

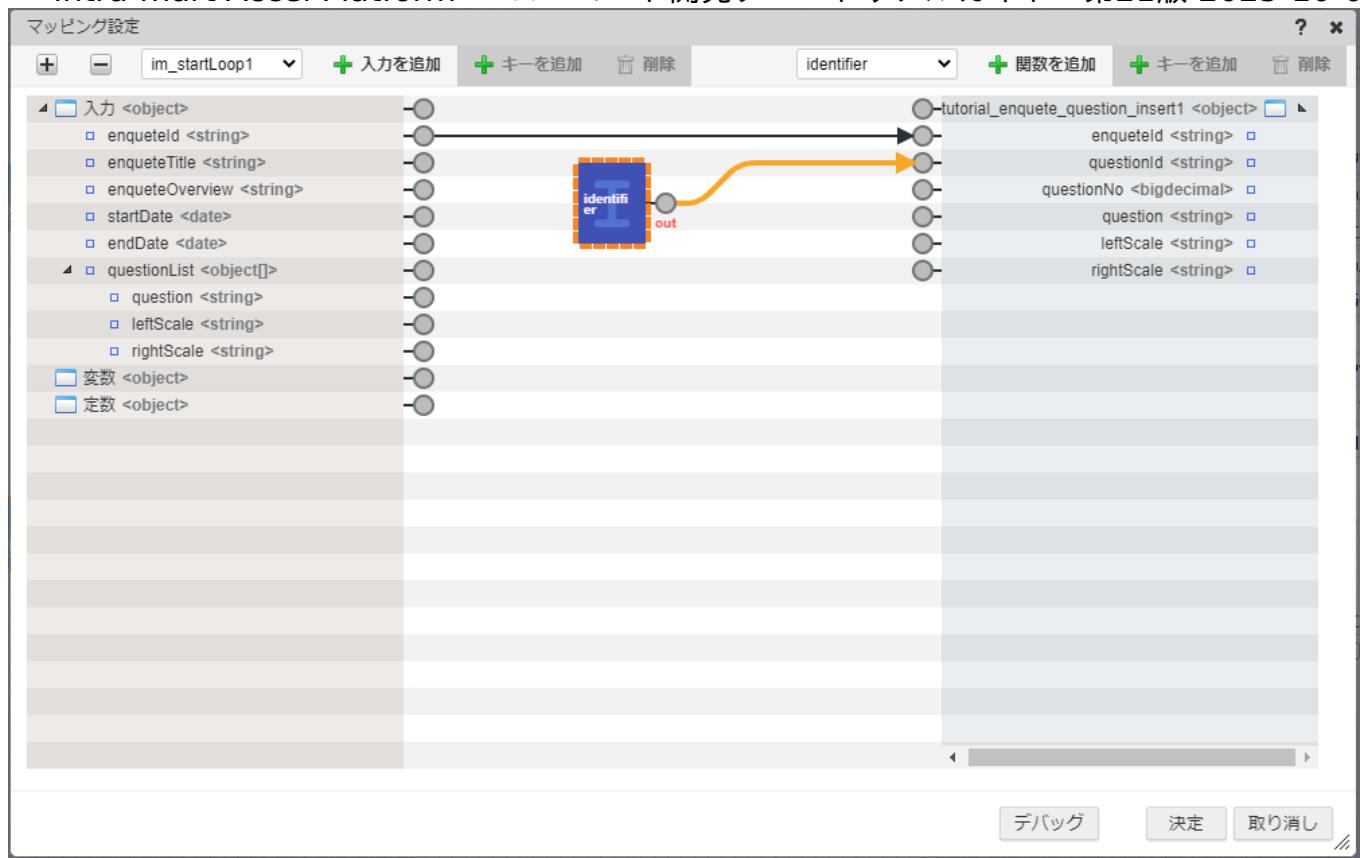
これで、入力値として渡されたquestionListの配列の長さだけ繰り返し「【チュートリアル】アンケート設問登録」が実行されます。



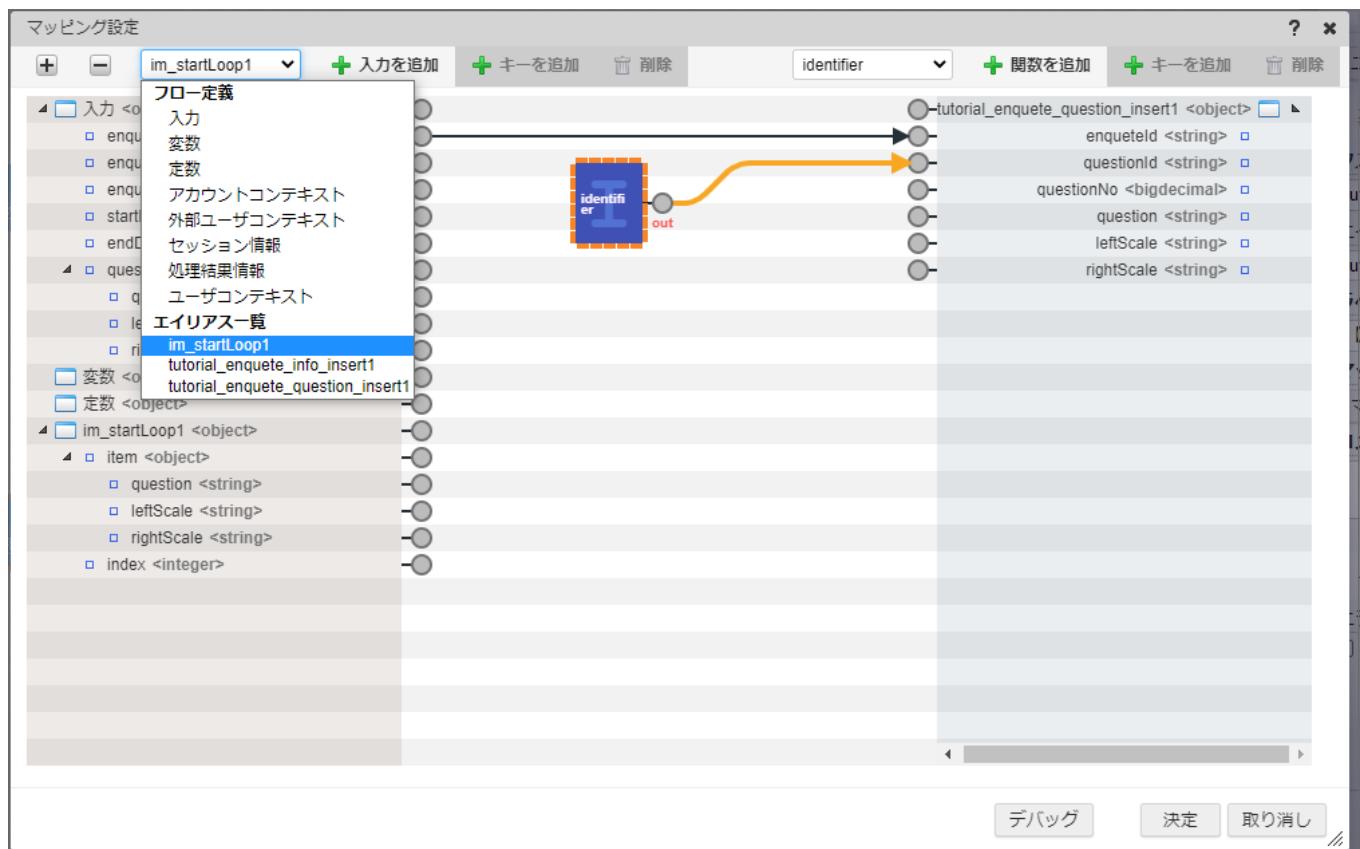
続けて、「【チュートリアル】アンケート設問登録」を選択しマッピング設定を開きます。

「入力」の「enqueteid」と「tutorial_enquete_question_insert」の「enqueteid」を線で繋ぎます。

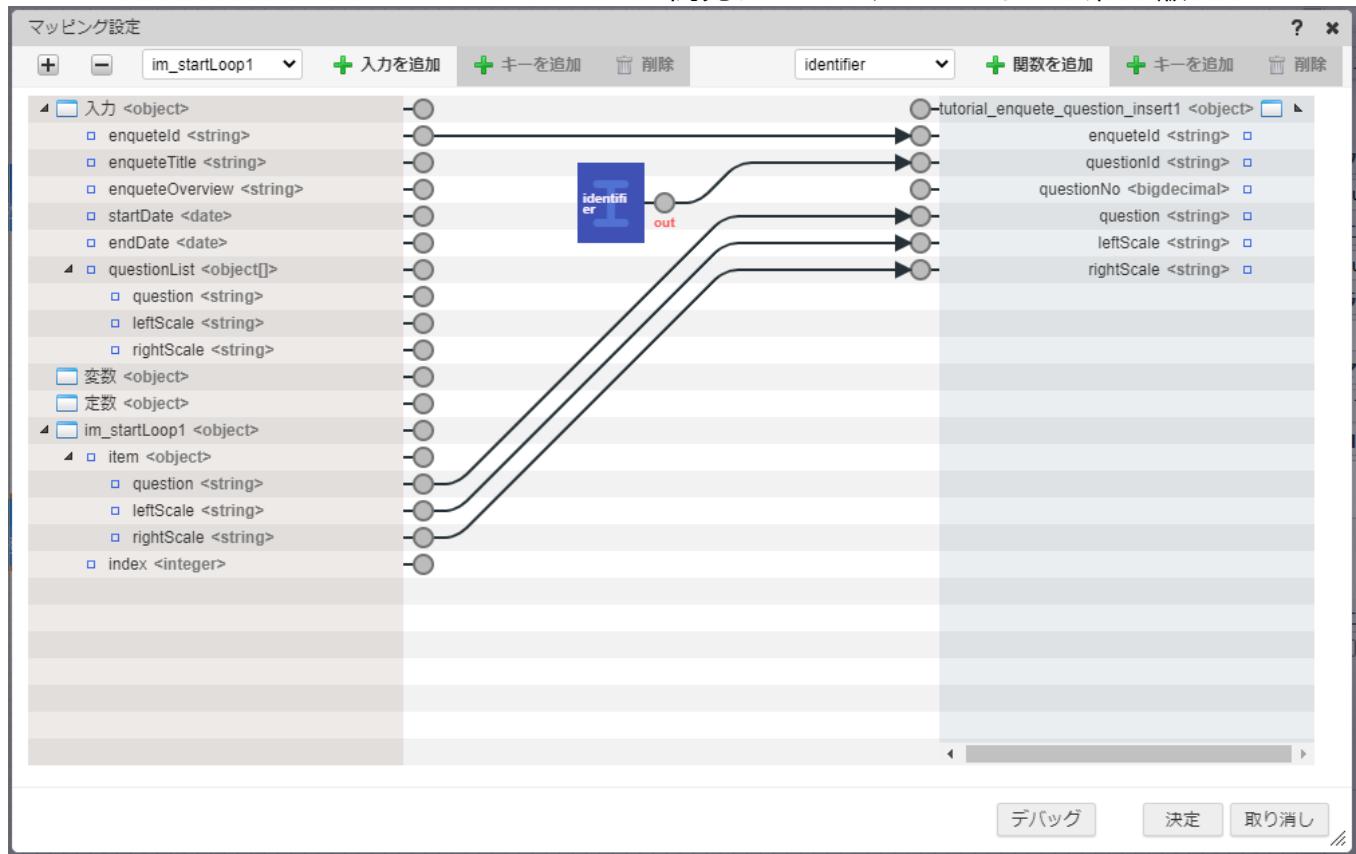
設問IDは自動設定とするため、関数から「identifier」を配置し、「tutorial_enquete_question_insert」の「questionId」を線で繋ぎます。



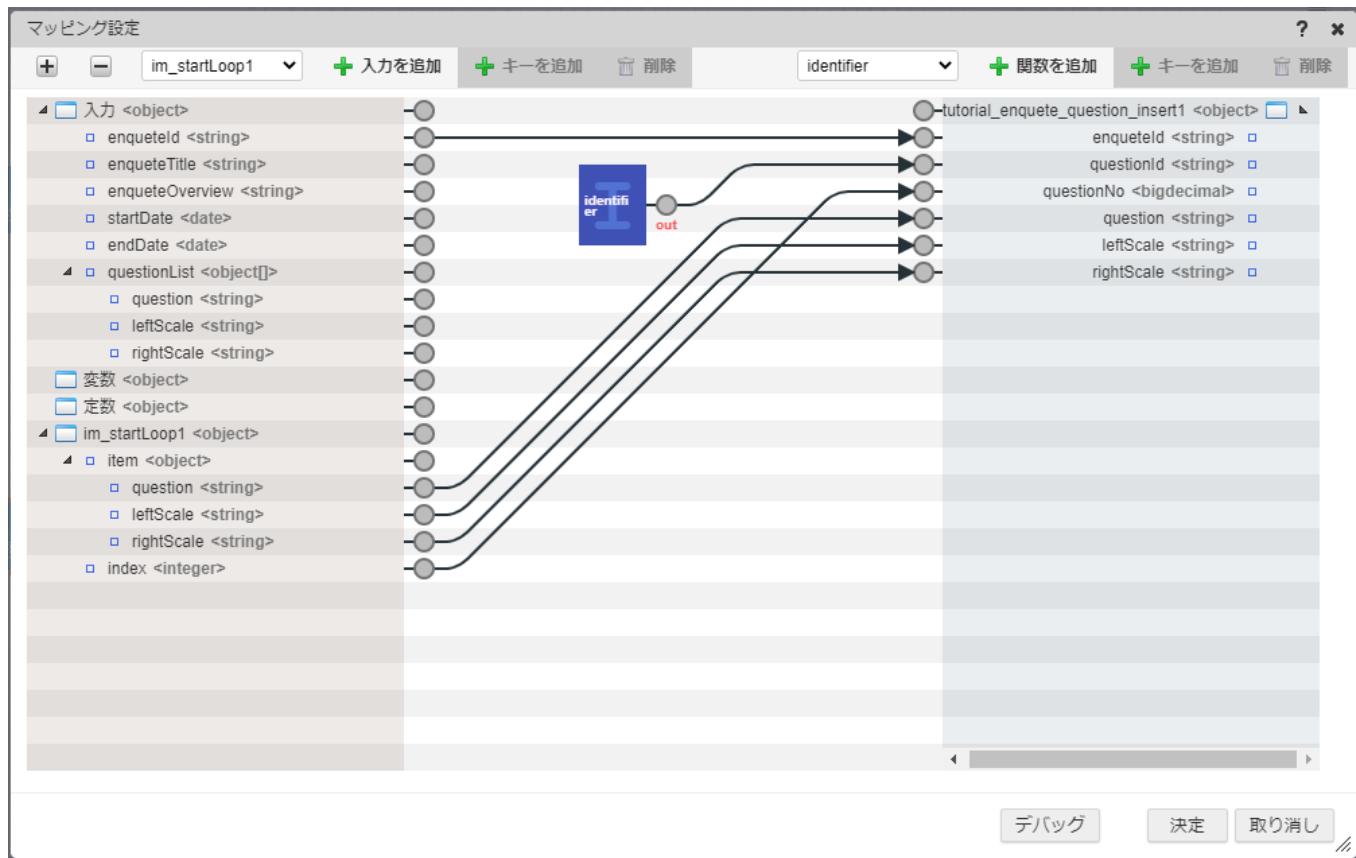
エイリアス一覧から「im_startLoop1」を選択し、「+入力を追加」を選択します。



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ぎます。



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 |
| フロー定義名（標準） | 【チュートリアル】アンケート設問登録 |
| フローカテゴリ | 【チュートリアル】チュートリアルカテゴリ |

新規保存

| | | | | | | | | | |
|---|---|------|--------------------|----|----------------------|-----|--|---------------|--|
| フロー定義ID * | tutorial_enquete_insert | | | | | | | | |
| フロー定義名 * | <table border="1"> <tr> <td>標準 *</td> <td>【チュートリアル】アンケート設問登録</td> </tr> <tr> <td>英語</td> <td></td> </tr> <tr> <td>日本語</td> <td></td> </tr> <tr> <td>中国語 (中華人民共和国)</td> <td></td> </tr> </table> | 標準 * | 【チュートリアル】アンケート設問登録 | 英語 | | 日本語 | | 中国語 (中華人民共和国) | |
| 標準 * | 【チュートリアル】アンケート設問登録 | | | | | | | | |
| 英語 | | | | | | | | | |
| 日本語 | | | | | | | | | |
| 中国語 (中華人民共和国) | | | | | | | | | |
| フローカテゴリ * | 検索/新規作成 <table border="1"> <tr> <td>ID *</td> <td>tutorial_category</td> </tr> <tr> <td>名称</td> <td>【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 | | | | | | | | | |
| <input type="button" value="決定"/> <input type="button" value="取り消し"/> | | | | | | | | | |

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。

The screenshot shows a search results table for API URLs. The table has columns for URL, Method, Description, and Last Modified. A context menu is open over the first row, with the '画面URL' option highlighted and surrounded by a red box.

| URL | Method | Description | Last Modified |
|--|--------|-------------|----------------------------|
| APIのURL (ID: im-logic://routes/tutorial_enquete/answer/list) GET tutorial_enquete/answer/list | GET | | tenant 2024/02/26 17:12:58 |
| APIのURL (ID: im-logic://routes/tutorial_enquete/list) GET tutorial_enquete/list | GET | | tenant 2024/02/26 17:15:19 |
| APIのURL (ID: im-logic://routes/tutorial_enquete/question/GET) GET tutorial_enquete/question | GET | | tenant 2024/02/27 17:21:49 |
| APIのURL (ID: im-logic://routes/tutorial_enquete/answer/POST) POST tutorial_enquete/answer | POST | | tenant 2024/02/26 19:37:20 |
| APIのURL (ID: im-logic://routes/tutorial_enquete/question/insert/POST) POST tutorial_enquete/question/insert | POST | | tenant 2024/02/26 19:39:02 |
| 画面URL (ID: im-bloommaker://routes/tutorial_enquete_answer) 【チュートリアル】アンケート回答 | | | tenant 2024/02/26 19:37:20 |
| 画面URL (ID: im-bloommaker://routes/tutorial_enquete_totalize) 【チュートリアル】アンケート集計画面 | | | 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】アンケート一覧取得」を選択し、決定ボタンをクリックします。

The screenshot shows a search dialog box titled 'ロジックフロー定義検索'. The search bar contains '【チュートリアル】アンケート一覧'. The table has columns for 選択 (Selected), フロー定義ID (Flow Definition ID), フロー定義名 (Flow Definition Name), and プレビュー (Preview). One row is selected, showing 'tutorial_enquete_list' and '【チュートリアル】アンケート一覧取得'. At the bottom is a large '決定' (Decision) button.

| 選択 | フロー定義ID | フロー定義名 | プレビュー |
|--------------------------|-----------------------|--------------------|-------|
| <input type="checkbox"/> | tutorial_enquete_list | 【チュートリアル】アンケート一覧取得 | |

「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ID * tutorial_enquete_list
フロー定義名 【チュートリアル】アンケート一覧取得

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 * /im-logic/api/tutorial_enquete/list
メソッド * GET
認証方法 * IMAuthentication
認可URI * im-logic-rest://tutorial_enquete/list
セキュリティ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 + 追加 ヘッダ名 * ヘッダ値 * 削除

登録

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

リソース閲覧図 権限設定 ログ モード

リソースを追加

検索 履歴

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

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

開基
閉基
リソース種別 (小項目)
画面URL
APIのURL
クエリのURL

画面URL

APIのURL

クエリのURL

tenant 2024/02/26 17:12:58

tenant 2024/02/26 17:15:19

tenant 2024/02/27 17:21:49

tenant 2024/02/28 19:37:20

tenant 2024/02/28 19:39:02

クリア

「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。

ロジックフロー定義検索

| 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」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」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 * | <input type="button" value="検索"/> 検索 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 * | <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 * |

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/ tutorial_enquete/question/insert |
| メソッド * | POST |
| 認証方法 * | IMAuthentication |
| 認可URI * | im-logic-rest:// 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 | + 追加 ヘッダ名 *: <input type="text"/> ヘッダ値 *: <input type="text"/> |

登録

アンケート一覧、登録画面の作成

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、デザイナ画面を開きます。

変数の設定

画面で利用する変数を作成します。

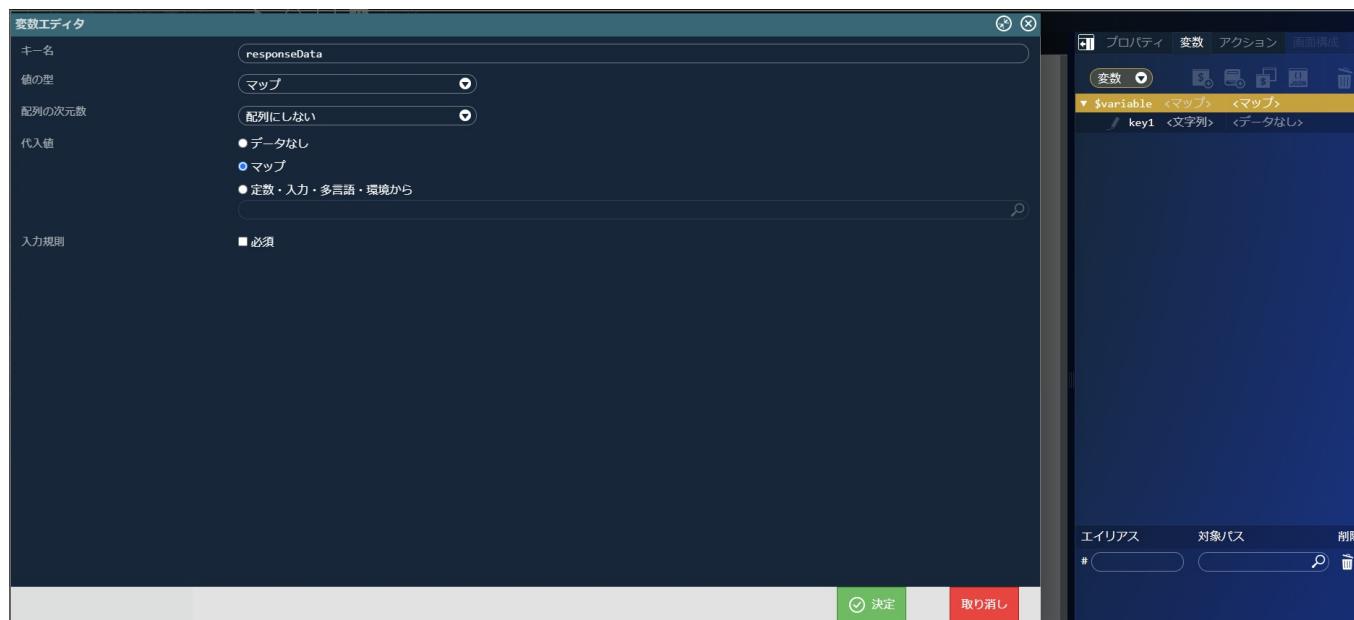
最初に、アンケート一覧取得処理画面に関連する変数を作成します。

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。

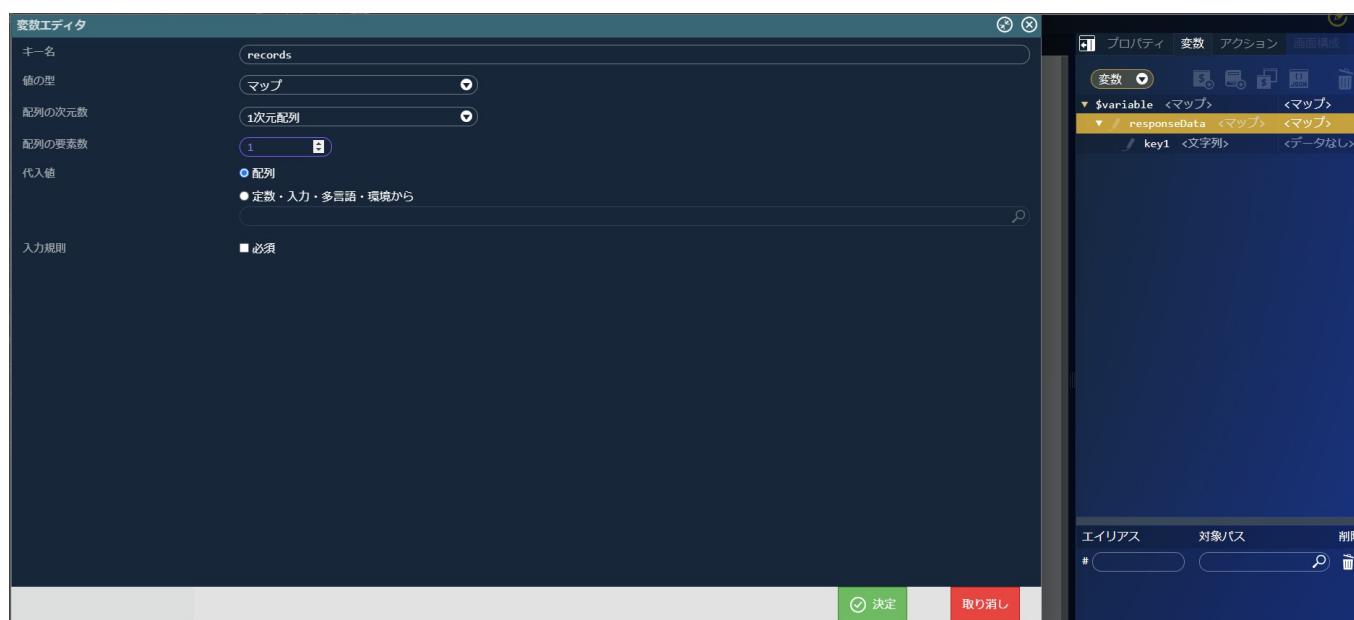
「\$variable」を選択し、 をクリックして変数エディタを表示します。

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。

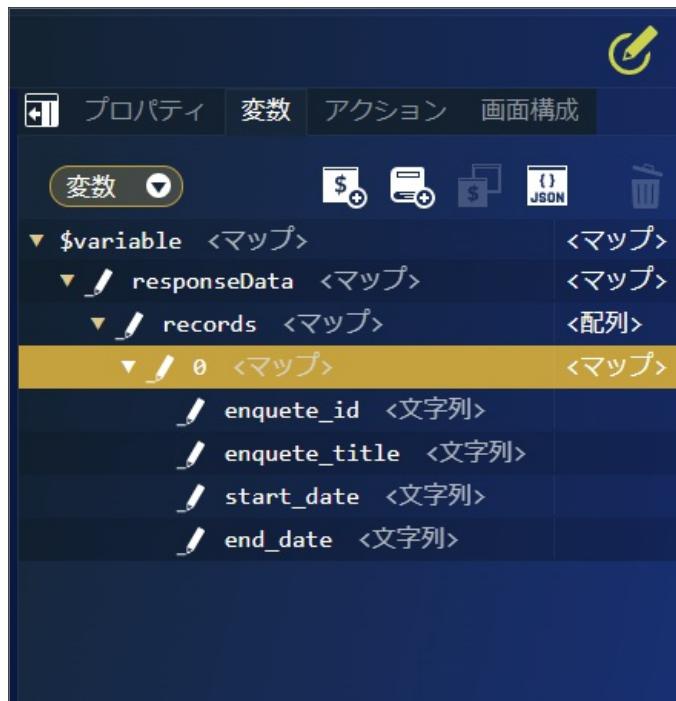
変数エディタのキー名に「records」と入力し、値の型を「マップ」に、配列の次元数を「1次元配列」、配列の要素数を「1」に設定して「決定」ボタンをクリックします。



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。

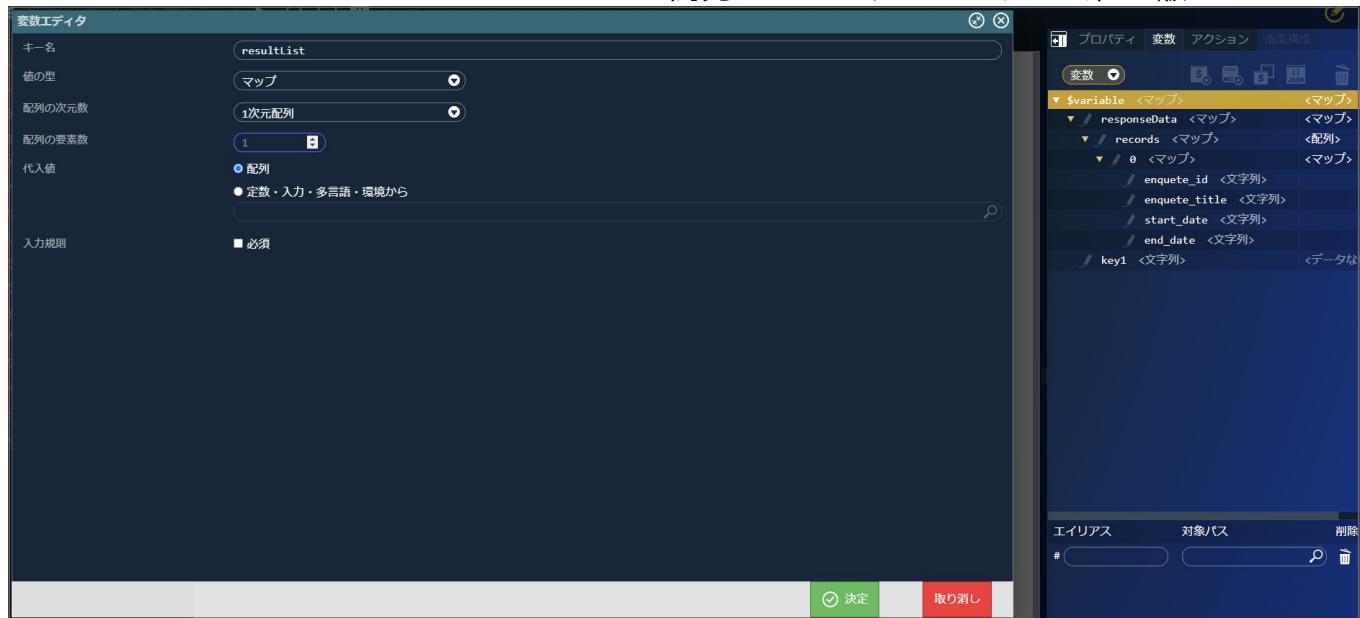
```
"responseData": {
  "records": [
    {
      "enquete_id": "",
      "enquete_title": "",
      "start_date": "",
      "end_date": ""
    }
  ]
}
```

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ナに「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。

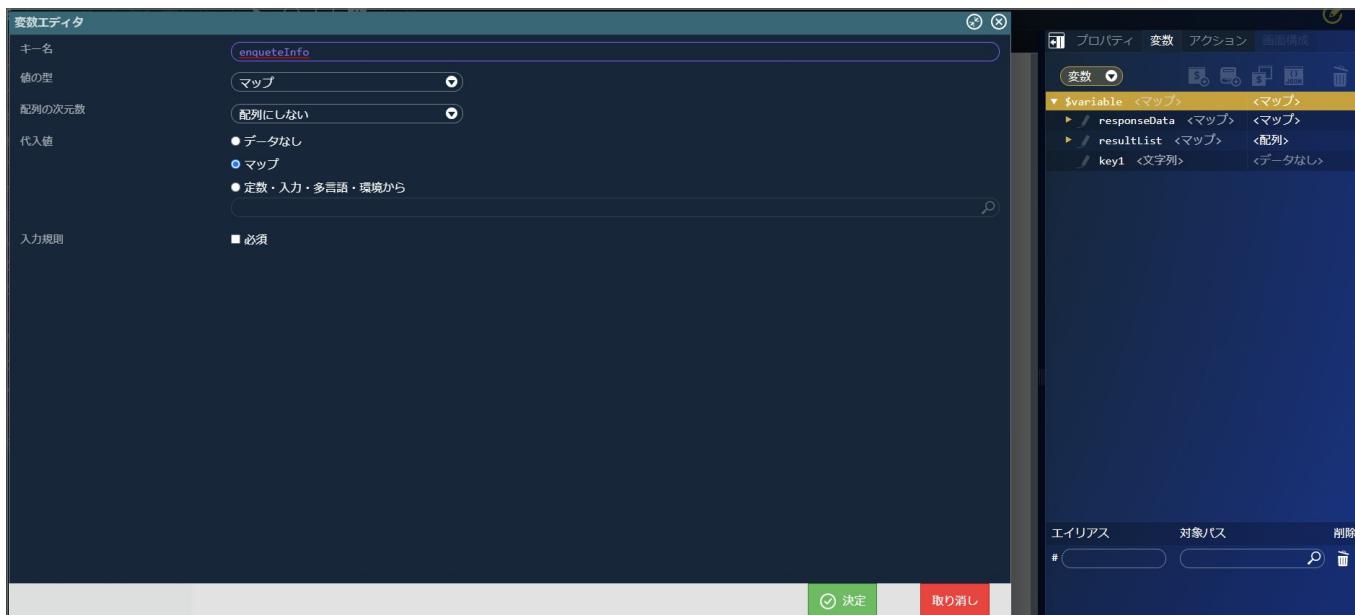
```
"resultList": [
{
  "title": "",
  "startDate": "",
  "endDate": "",
  "answerUrl": "",
  "totalUrl": ""
}]
```

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と誤りが無いか注意してください。

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。

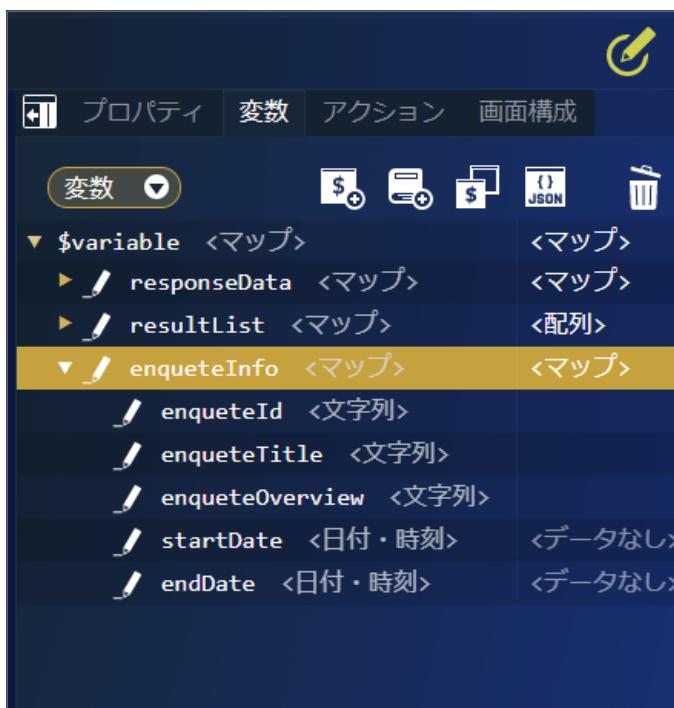
「\$variable」を選択し、 をクリックして変数エディタを表示します。

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



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。

「enquetId」、「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。

The screenshot shows the 'Properties' tab selected in the top navigation bar. The 'Variables' tab is active, indicated by a yellow background. The tree view shows the following structure:

- \$variable <マップ>
- responseData <マップ>
- resultList <配列>
- enqueteInfo <マップ> (selected)
 - enqueteId <文字列>
 - enqueteTitle <文字列>
 - enqueteOverview <文字列>
 - startDate <日付・時刻> <データなし>
 - endDate <日付・時刻> <データなし>
 - questionList <マップ> <配列>
 - 0 <マップ> <マップ>
 - question <文字列>
 - leftScale <文字列>
 - rightScale <文字列>

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。

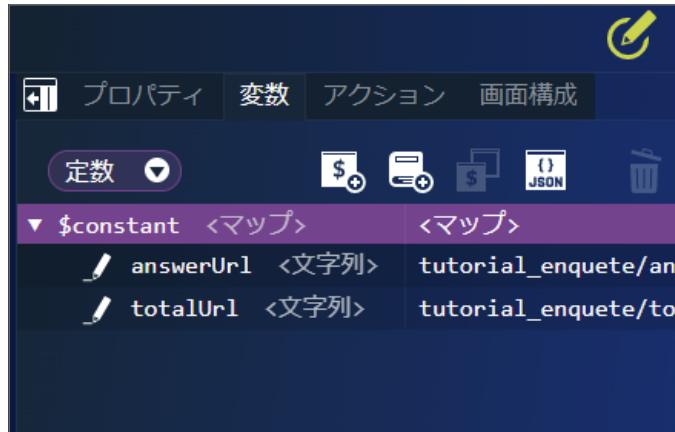
「変数」タブで「定数」を選択します。

The screenshot shows the 'Properties' tab selected in the top navigation bar. The 'Variables' tab is active, indicated by a yellow background. The tree view shows the following structure:

- 定数 <マップ> <マップ> (highlighted)
- 変数
- 定数
- 入力
- 多言語
- 環境

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ナに「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ナに「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入力で以下を記載して設定できます。

```
"answerUrl": "tutorial_enquete/answer?enquetId=",
"totalUrl": "tutorial_enquete/totalize?enquetId="
```

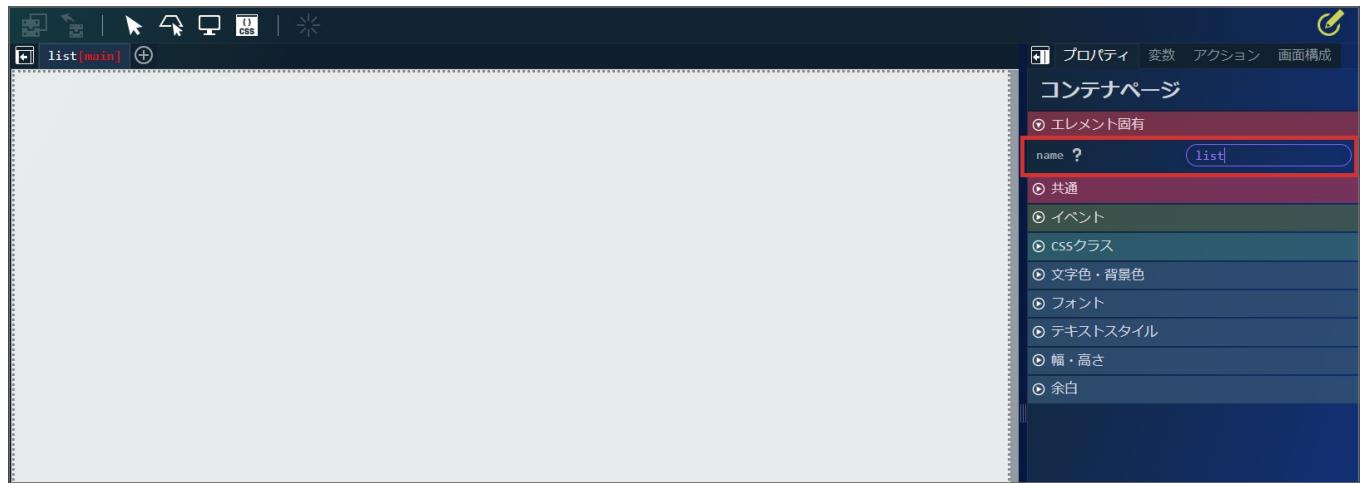
JSON入力する場合は、定数の  をクリックして「JSONエディタ」を表示してください。また、JSON形式での入力の際には型が上記の表と誤りが無いか注意してください。

アンケート一覧画面の作成

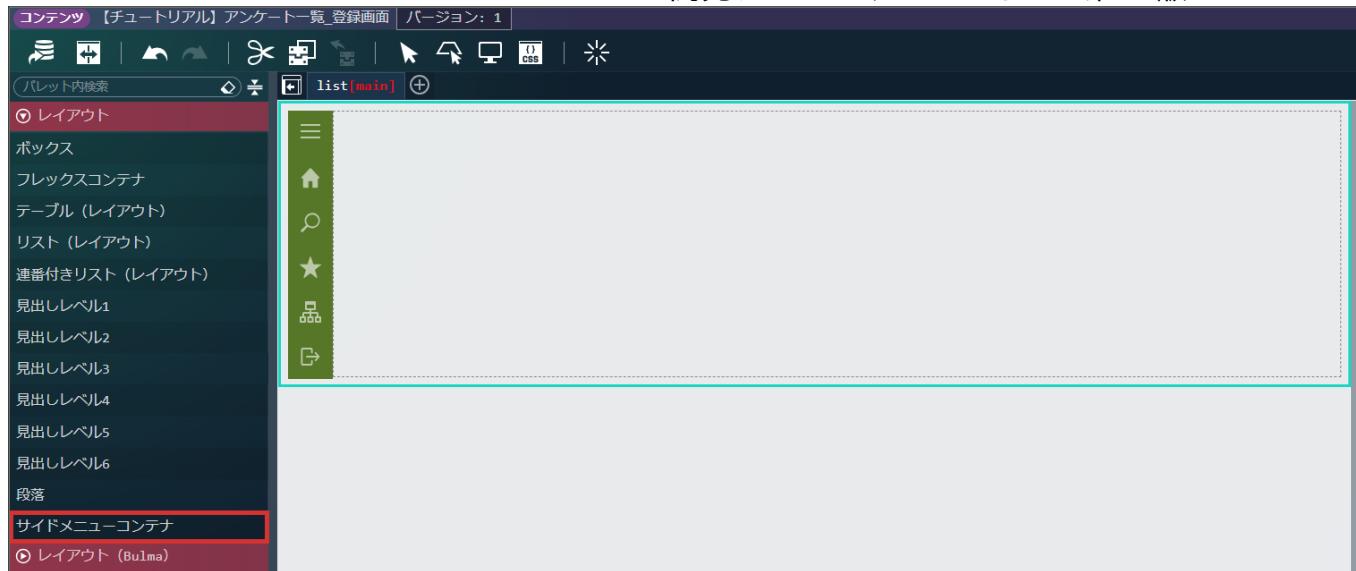
アンケートの一覧画面から作成します。

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。

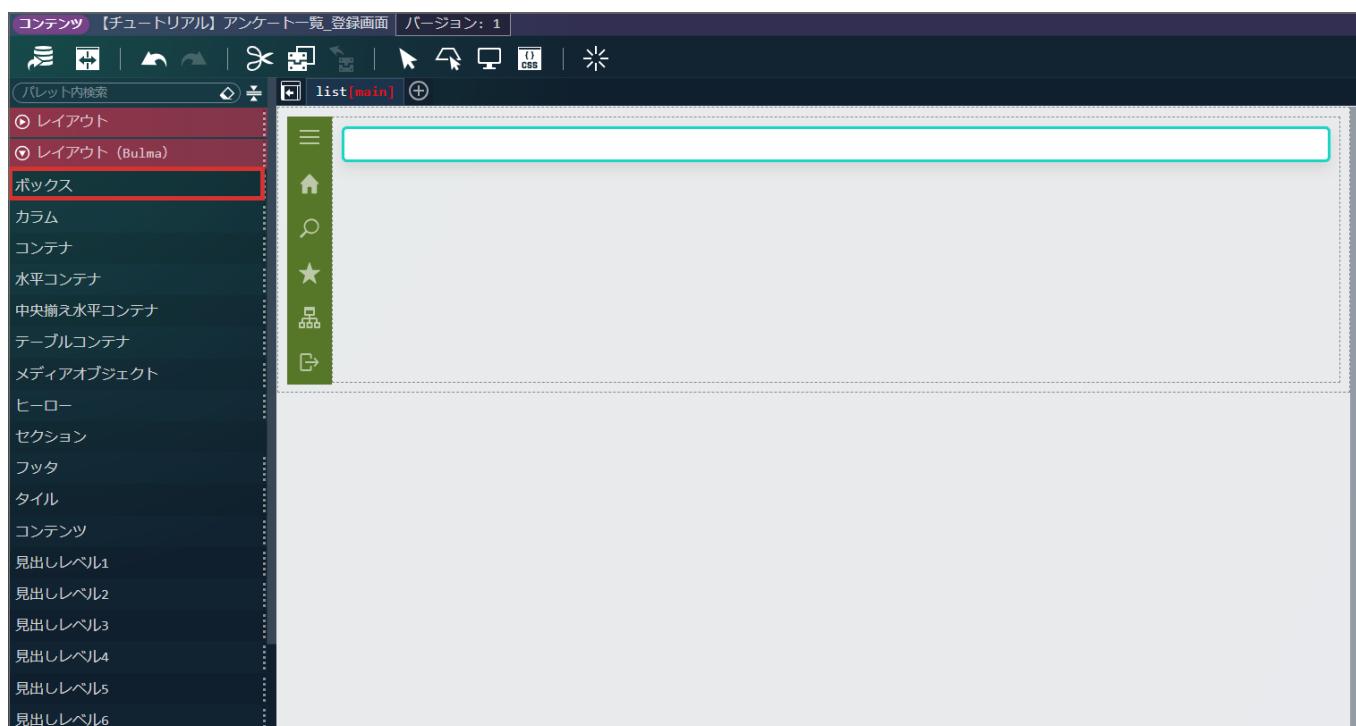


これで、タブの名称も「list」と変更されました。次に、画面にサイドメニューを表示するためにサイドメニューコンテナを配置します。画面左側のパレットの「レイアウト」 - 「サイドメニュー」を選択し、画面に配置します。

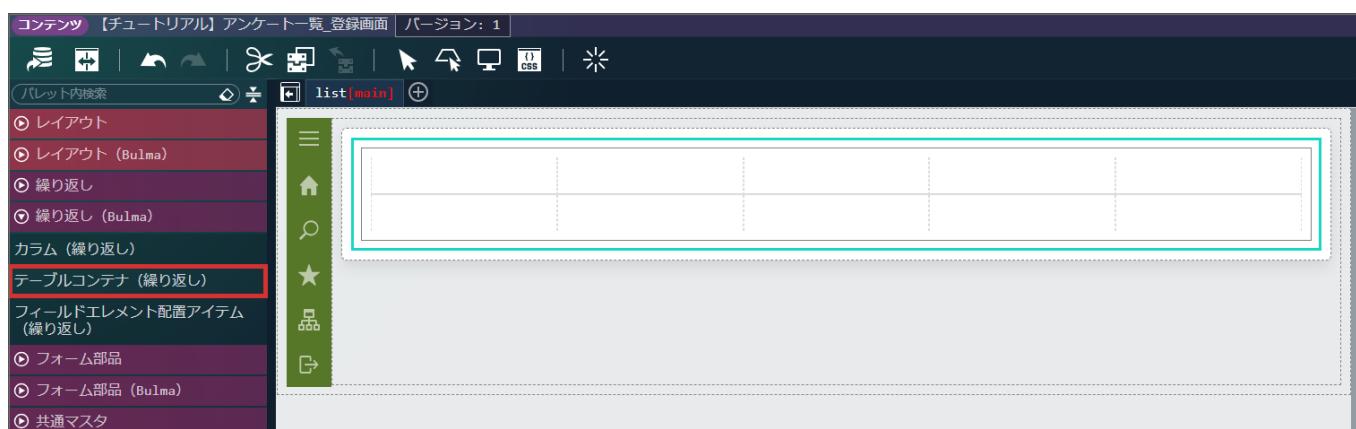


次に、アンケート一覧の表示領域を作成します。

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「サイドメニューインテナ」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。



The screenshot shows the configuration interface for a 'Table (Repeat)' component. At the top, there are tabs: 'Properties' (selected), 'Variables', 'Actions', and 'Screen Structure'. Below the tabs, the breadcrumb navigation shows '繰り返し (Bulma) > テーブルコンテナ (繰り返し)'. The main area is titled 'Table (Repeat)'. A red box highlights the 'list' property, which has a value input field containing '5' and a search icon. Below the input field, a message says '値を入力してください。' (Please enter a value.). Other properties shown include 'showHeader' (checkbox checked), 'showFooter' (checkbox unchecked), 'showBorder' (checkbox unchecked), 'stripe' (checkbox unchecked), 'narrow' (checkbox unchecked), 'hoverable' (checkbox checked), 'fullwidth' (checkbox checked), and 'columnCount' (input field with value '5').

「\$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。



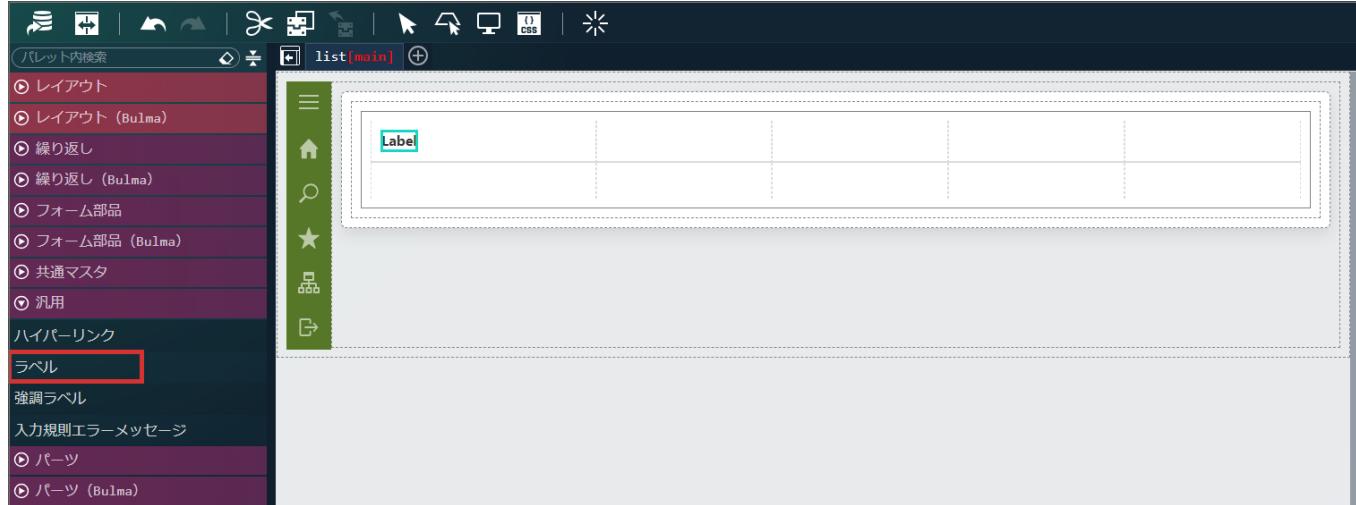
The screenshot shows the 'Variables · Constants · Input · Multilingual · Environment Selector' dialog. The 'Variables' tab is selected. On the left, a tree view shows variables like '\$variable' (mapped), 'responseData' (mapped), 'records' (mapped), '0' (mapped), 'resultList' (mapped), '0' (mapped), 'enqueteInfo' (mapped), and 'enqueteId' (string). The 'resultList' node is highlighted with a yellow background. On the right, detailed information for 'resultList' is shown: type '配列' (Array), value '[]', and description '<マップ>'. Below this, the '0' node under 'resultList' is also highlighted with a yellow background, showing type 'マップ' (Map), value '{}', and description '<マップ>'. At the bottom, there are 'Decision' and 'Cancel' buttons.

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。

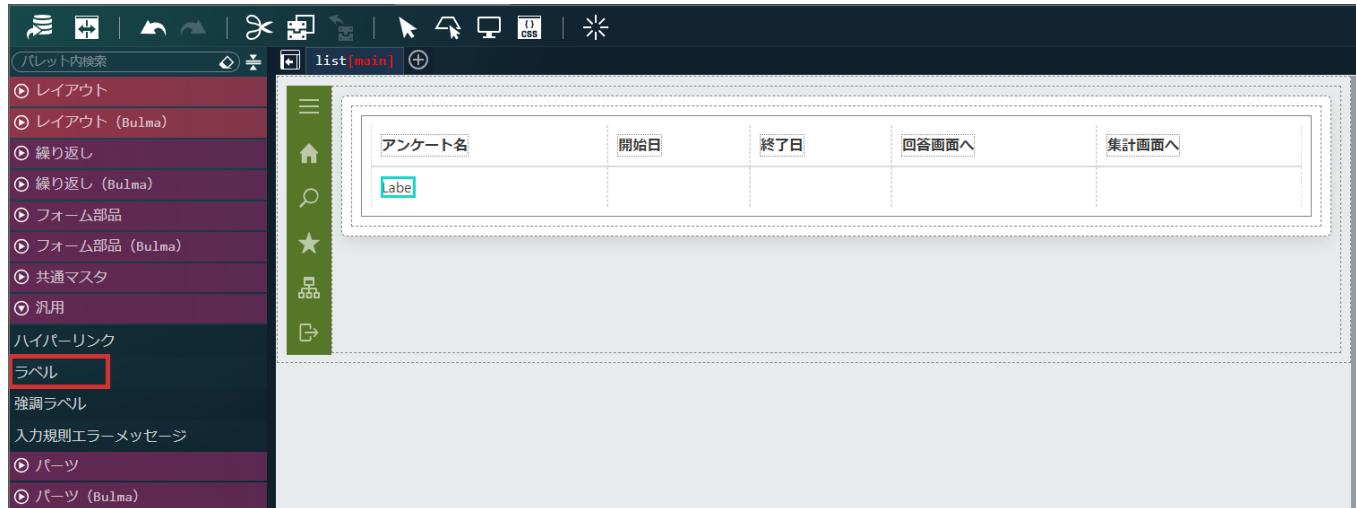


同様に、テーブルヘッダが左から「アンケート名」、「開始日」、「終了日」、「回答画面へ」、「集計画面へ」となるように設定します。



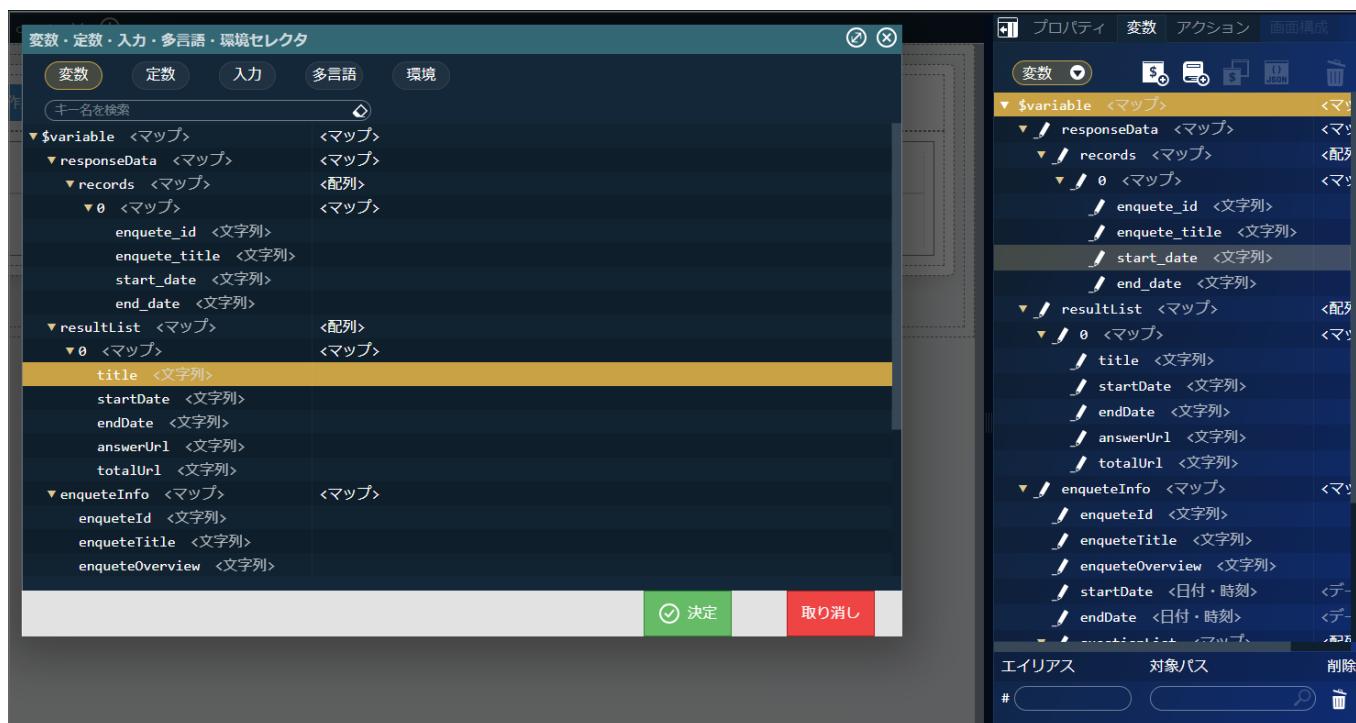
次に、一覧の表示項目を設定します。

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。

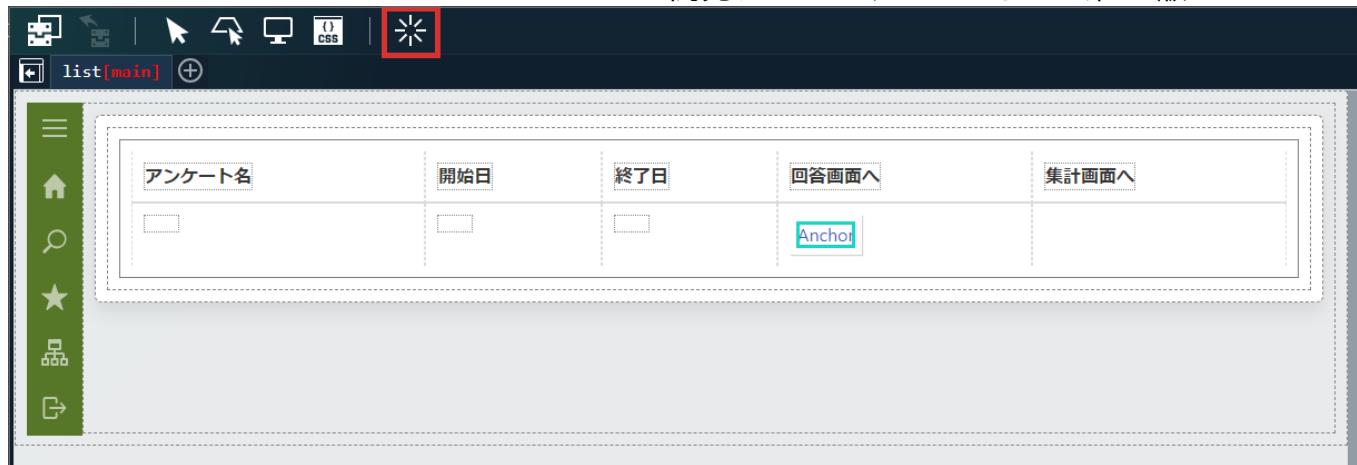
次に、回答画面へのリンクを作成します。

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。

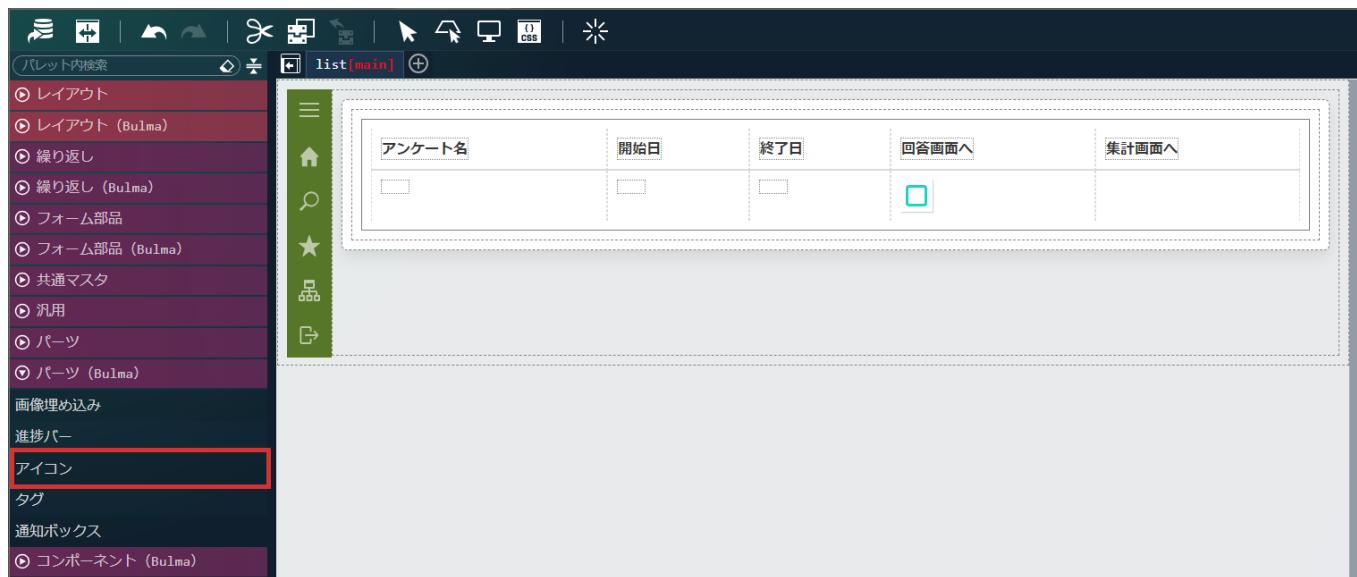
配置した「ハイパーリンク」をクリックし、画面右側の「プロパティ」タブ - 「エレメント固有」 - 「href」を変数値に設定し、「\$variable.resultList[\$index].answerUrl」を設定します。

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、を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

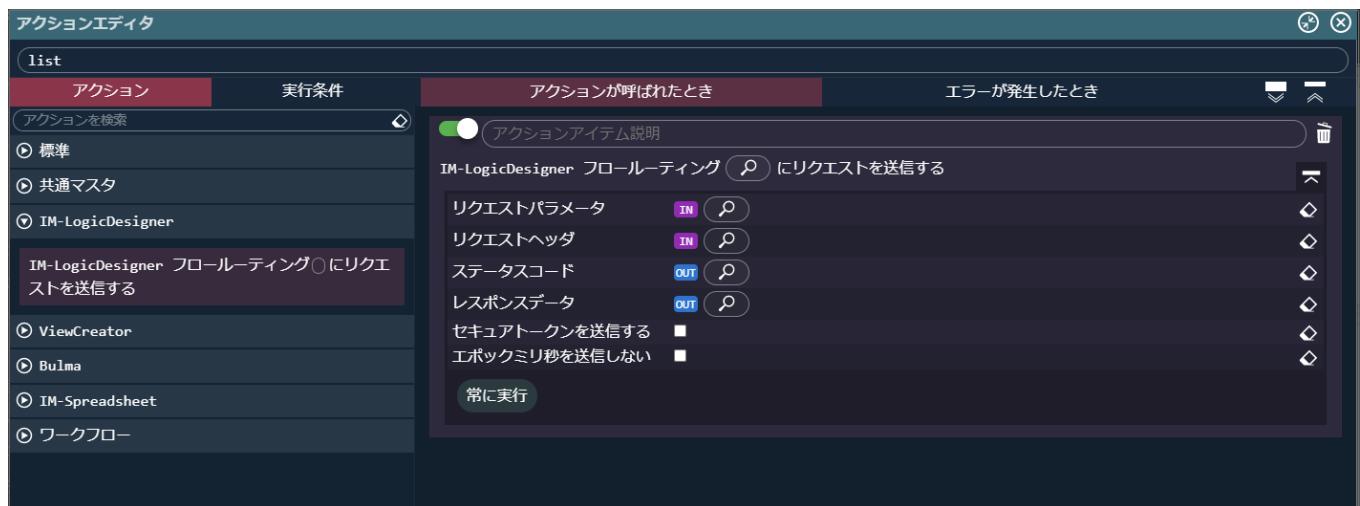
一覧画面で利用するアクションを作成します。

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。

IM-LogicDesigner フロールーティング定義検索

【チュートリアル】アンケート一覧取得

| 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/list | GET | tutorial_enquete_list | 【チュートリアル】アンケート一覧取得 |

1 ページ中 (1) ページ目 << >> (50) 1 件中 1 - 1 を表示

決定 取り消し

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。

変数セレクタ

変数

| | |
|-----------------------|---------|
| ▼ \$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

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○にリクエストを送信する」の次に配置します。

The screenshot shows the Action Editor interface. On the left, a sidebar lists various actions like 'Send Form', 'Copy Variable Value', etc. The main area is divided into two sections: 'When Action is Called' and 'When Error Occurs'. The first section contains a configuration for an 'IM-LogicDesigner' role-tapping action to send a request to 'tutorial_enquete/list'. The second section contains a configuration for a 'Custom Script' action to execute the provided JavaScript code.

```

$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);
}

```

決定 | 取り消し

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);
}

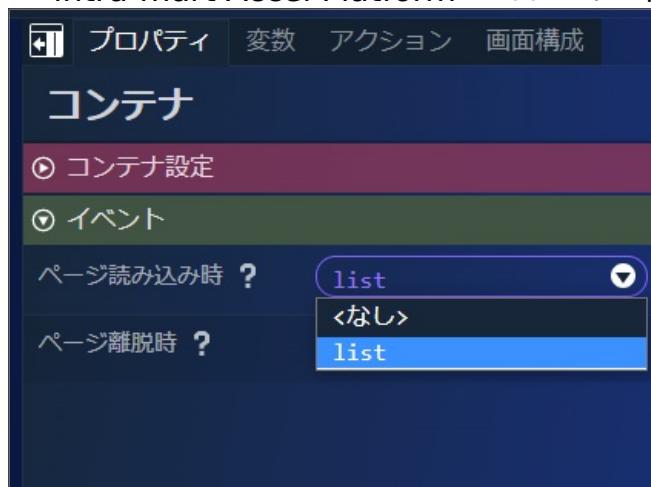
```

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。

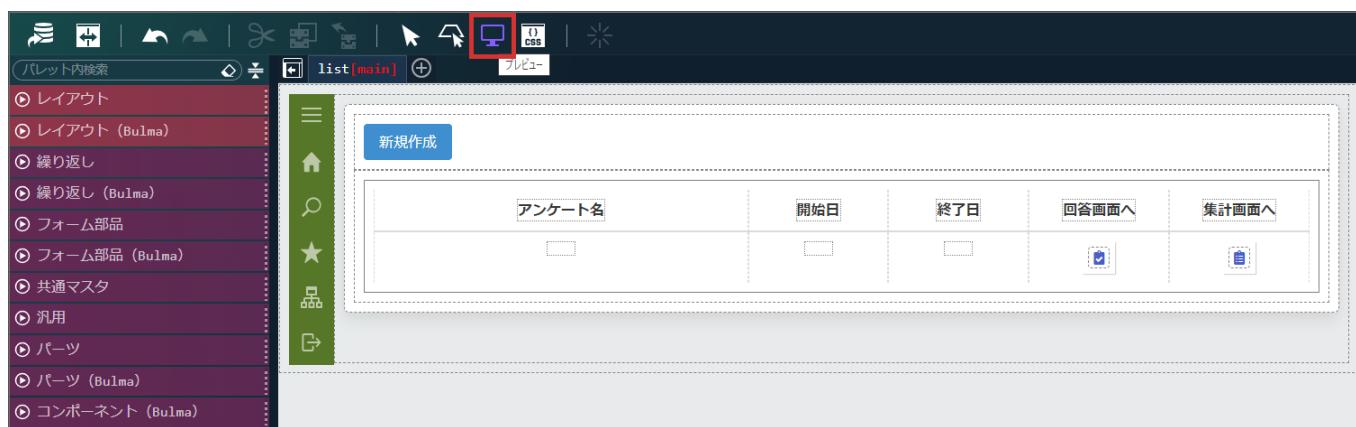
画面上部の をクリックします。

The screenshot shows the Layout Editor. A sidebar on the left lists layout components like 'Layout', 'Layout (Bulma)', 'Loop', etc. The main area shows a 'list' component with a 'New Create' dialog open. The dialog has fields for 'アンケート名' (Survey Name), '開始日' (Start Date), '終了日' (End Date), '回答画面へ' (Answer Page), and '集計画面へ' (Summary Page). The 'list' component is highlighted with a red box.

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 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 好きな映画ジャンル調査 | 2021/01/01 | 2022/01/01 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 好きな料理アンケート | 2024/03/14 | 2024/03/31 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 休日の過ごし方 | 2024/04/01 | 2024/04/15 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| 理解度チェック | 2024/03/01 | 2024/05/31 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

アンケート登録画面の作成

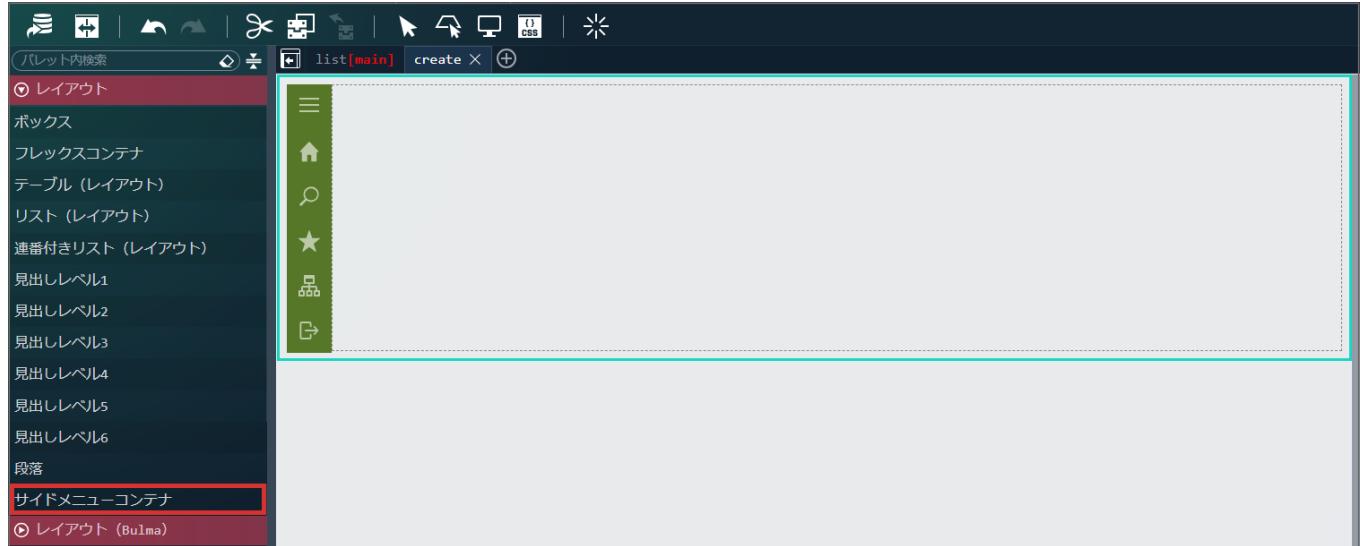
続けて、アンケートの登録画面を作成します。

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。



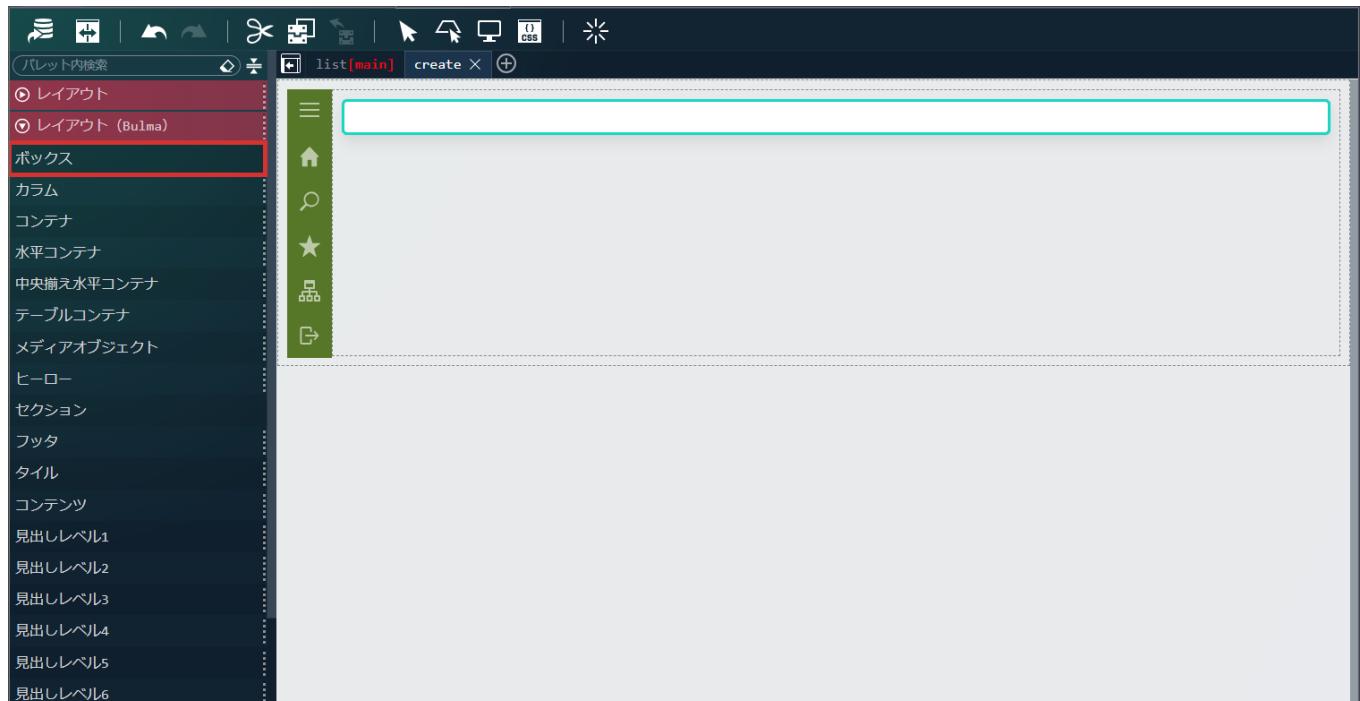
次に、画面にサイドメニューを表示するためにサイドメニュー コンテナを配置します。

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。

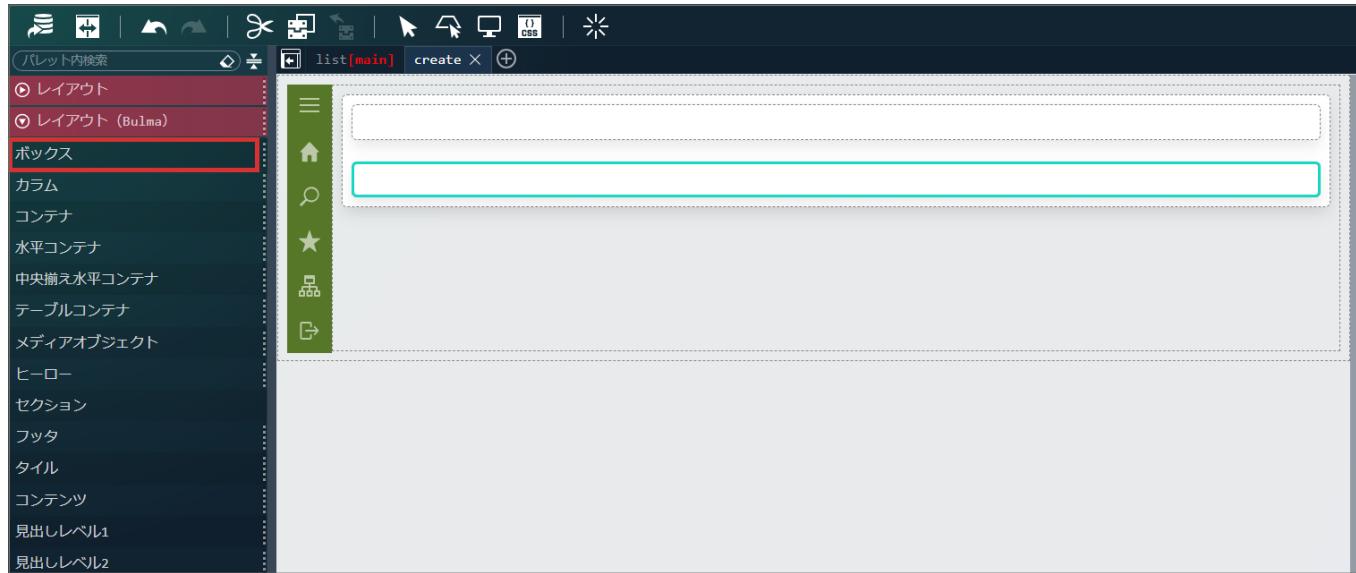


次に、アンケートの登録内容の表示領域を作成します。

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。



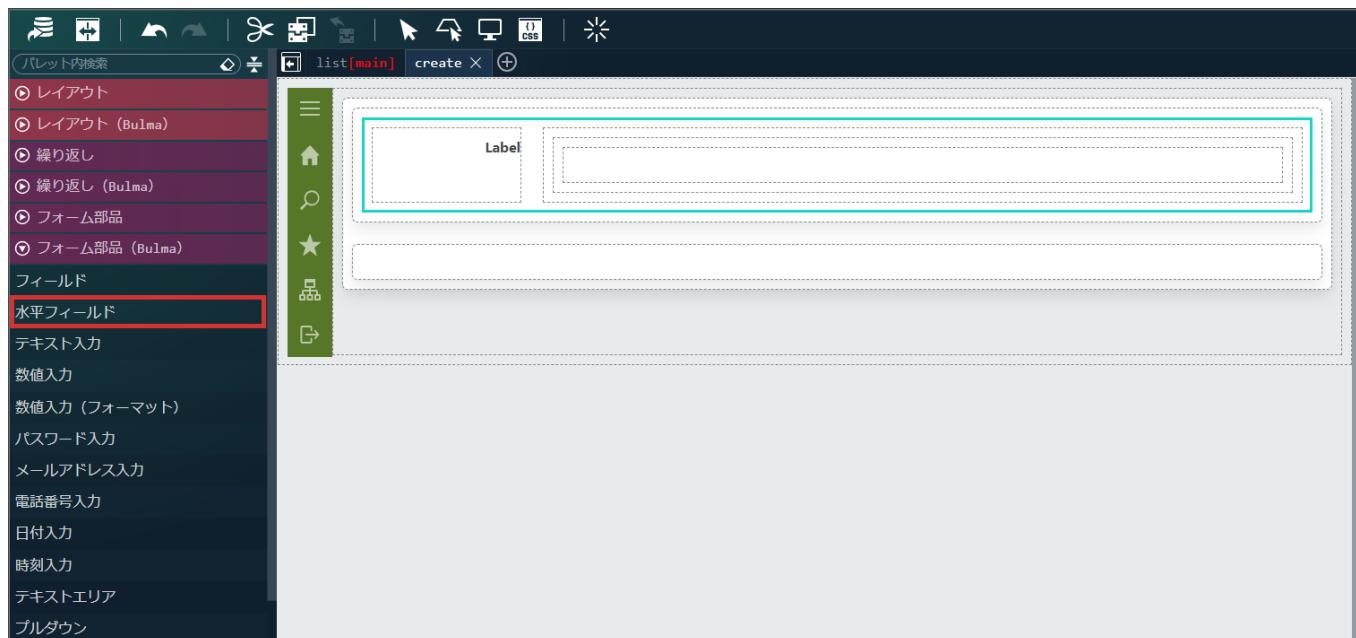
上に配置したボックスにアンケート情報の入力内容を、下に配置したボックスに設問情報の入力内容を作成していきます。

まずは、アンケート情報の入力内容を設定していきます。

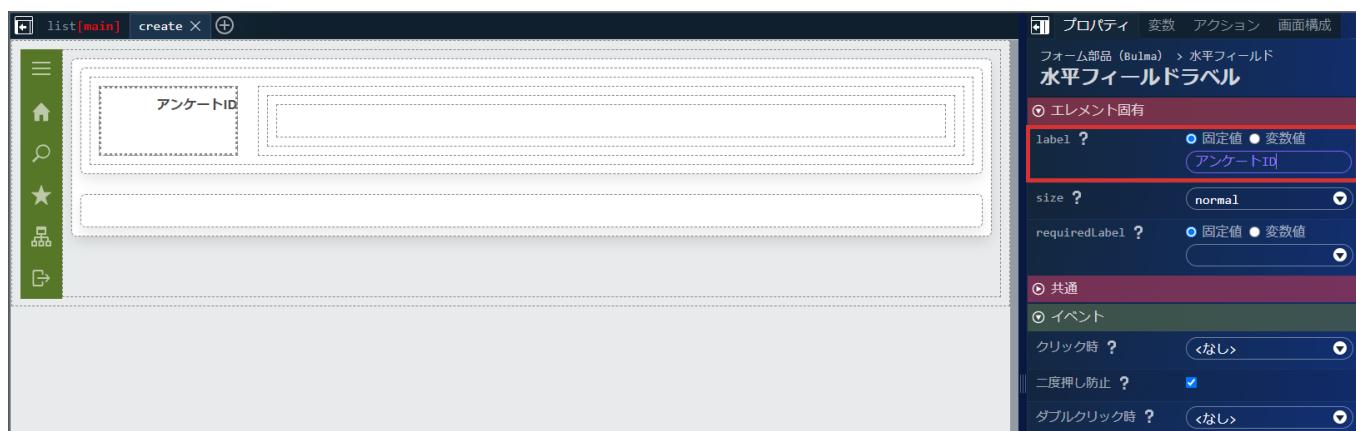
アンケート情報は「アンケートID」、「アンケートタイトル」、「アンケート概要」、「開始日」、「終了日」の入力を可能とします。

最初に、アンケートIDの入力欄を作成します。

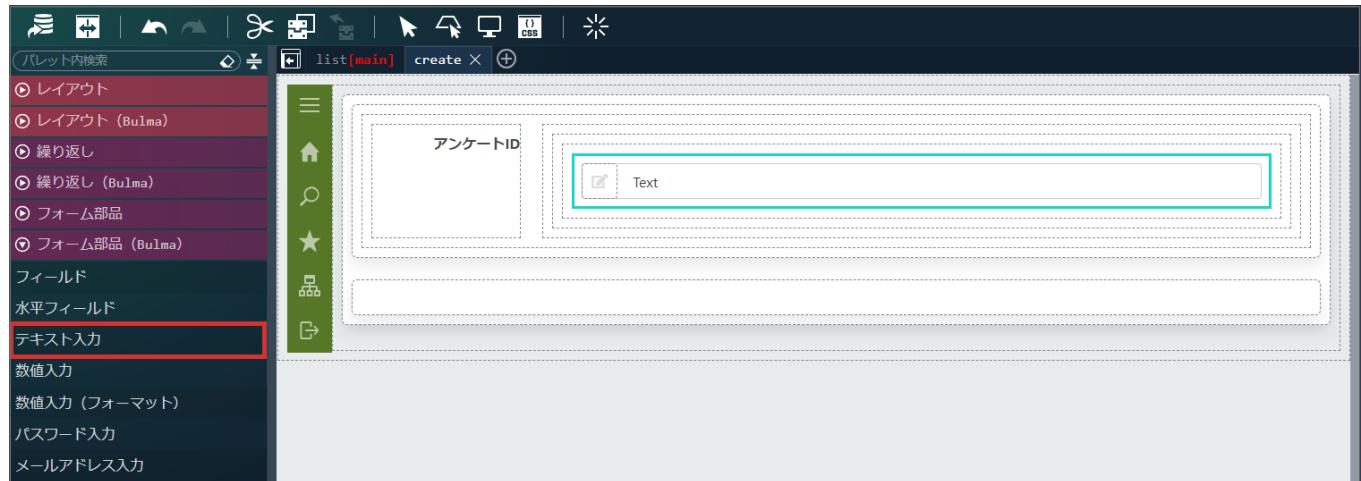
画面左側のパレットの「フォーム部品 (Bulma)」 - 「水平フィールド」を選択し、上に配置したボックスの中に配置します。



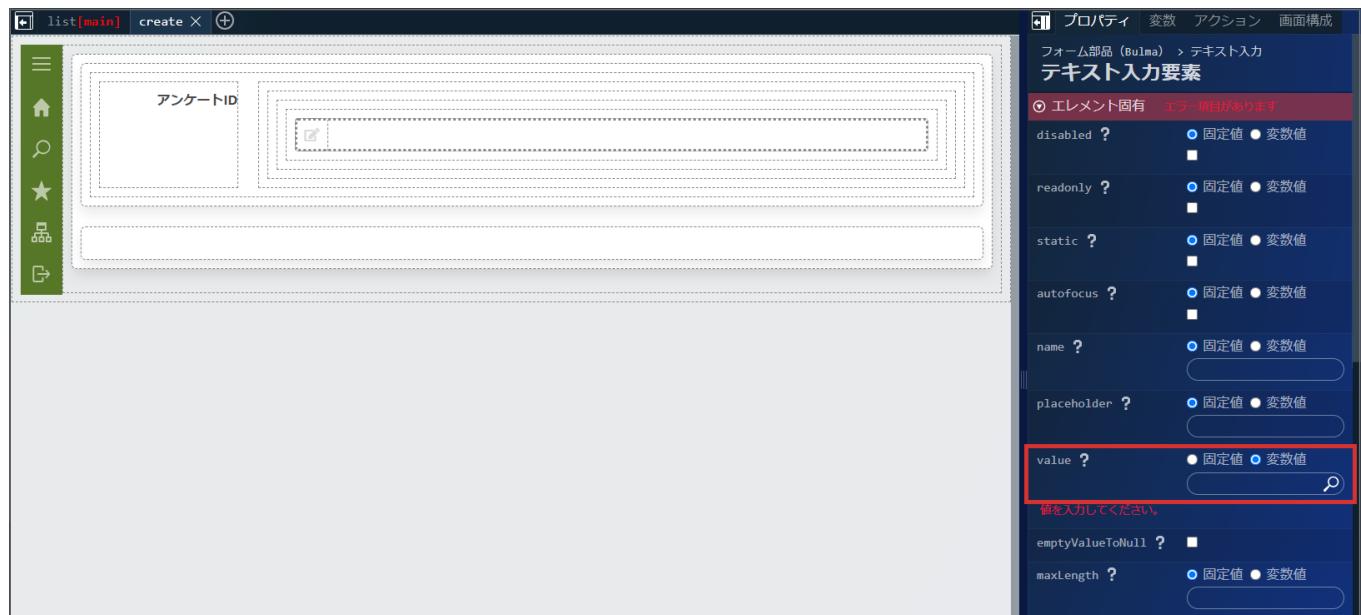
配置した「水平フィールド」の左側にある「水平フィールドラベル」を選択し、画面右側の「プロパティ」タブ - 「水平フィールドラベル」 - 「エレメント固有」 - 「label」に「アンケートID」と設定します。



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。



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 > enqueteInfo > enqueteId」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

```

start_date <文字列>
end_date <文字列>
▼ resultList <マップ>
  ▼ 0 <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>
    enqueteId <文字列>
    enqueteTitle <文字列>
    enqueteOverview <文字列>
    startDate <日付・時刻>
    endDate <日付・時刻>
  ▼ questionList <マップ>
    ▼ 0 <マップ>
      question <文字列>
      leftScale <文字列>
      rightScale <文字列>

```

決定 取り消し

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。

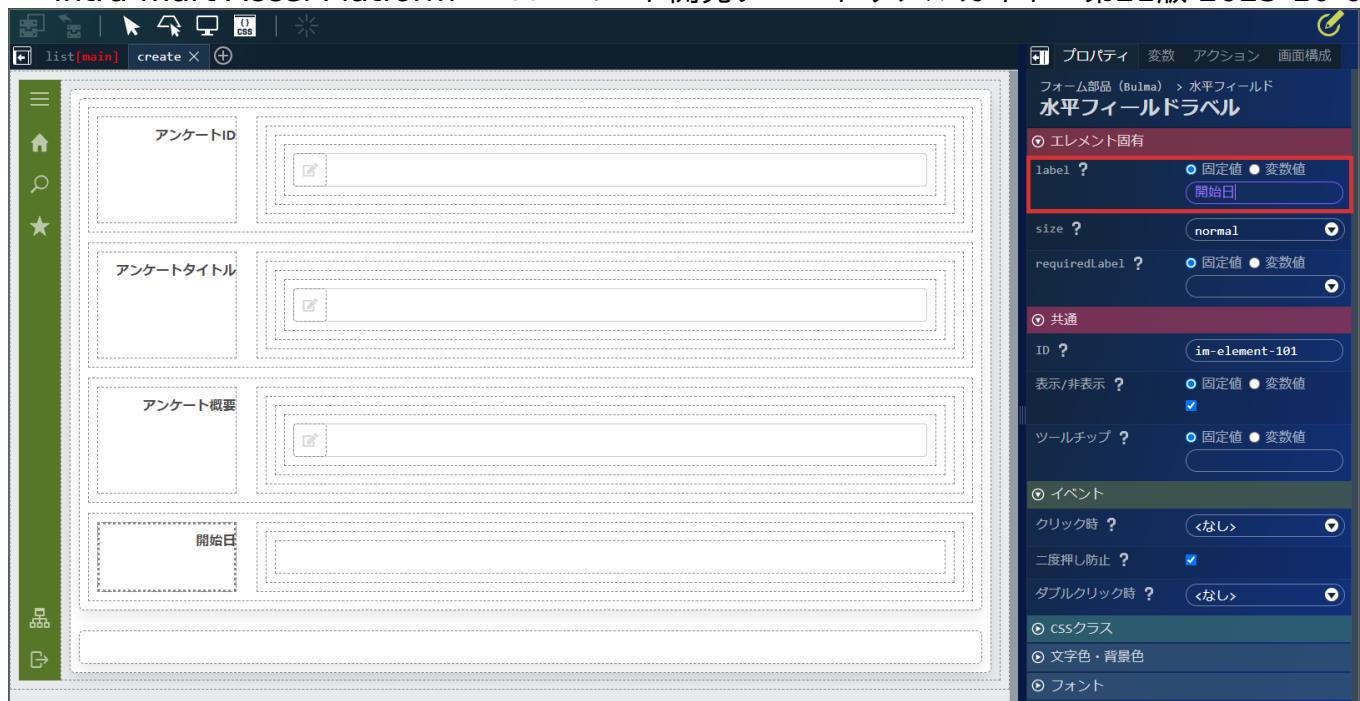
テキスト入力要素の「プロパティ」タブ - 「エレメント固有」 - 「value」にはそれぞれ、「アンケートタイトル」は「\$variable.enqueteInfo.enqueteTitle」、「アンケート概要」は「\$variable.enqueteInfo.enqueteOverview」を設定してください。

The screenshot shows the Accel Platform's configuration interface for creating form inputs. On the left, there's a sidebar with icons for list, create, and search. The main area displays three input fields labeled 'アンケートID', 'アンケートタイトル', and 'アンケート概要'. Each field has a small icon in its top-right corner. To the right, a detailed properties panel for the 'テキスト入力要素' (Text Input Element) is open. Under the 'エレメント固有' (Element Specific) tab, the 'value' property is highlighted with a red border. It shows two radio button options: '固定値' (Fixed Value) and '変数値' (Variable Value). The '変数値' option is selected, and the variable '\$variable.enquete...' is entered into the input field next to it.

続けて、「開始日」と「終了日」の入力欄を作成します。

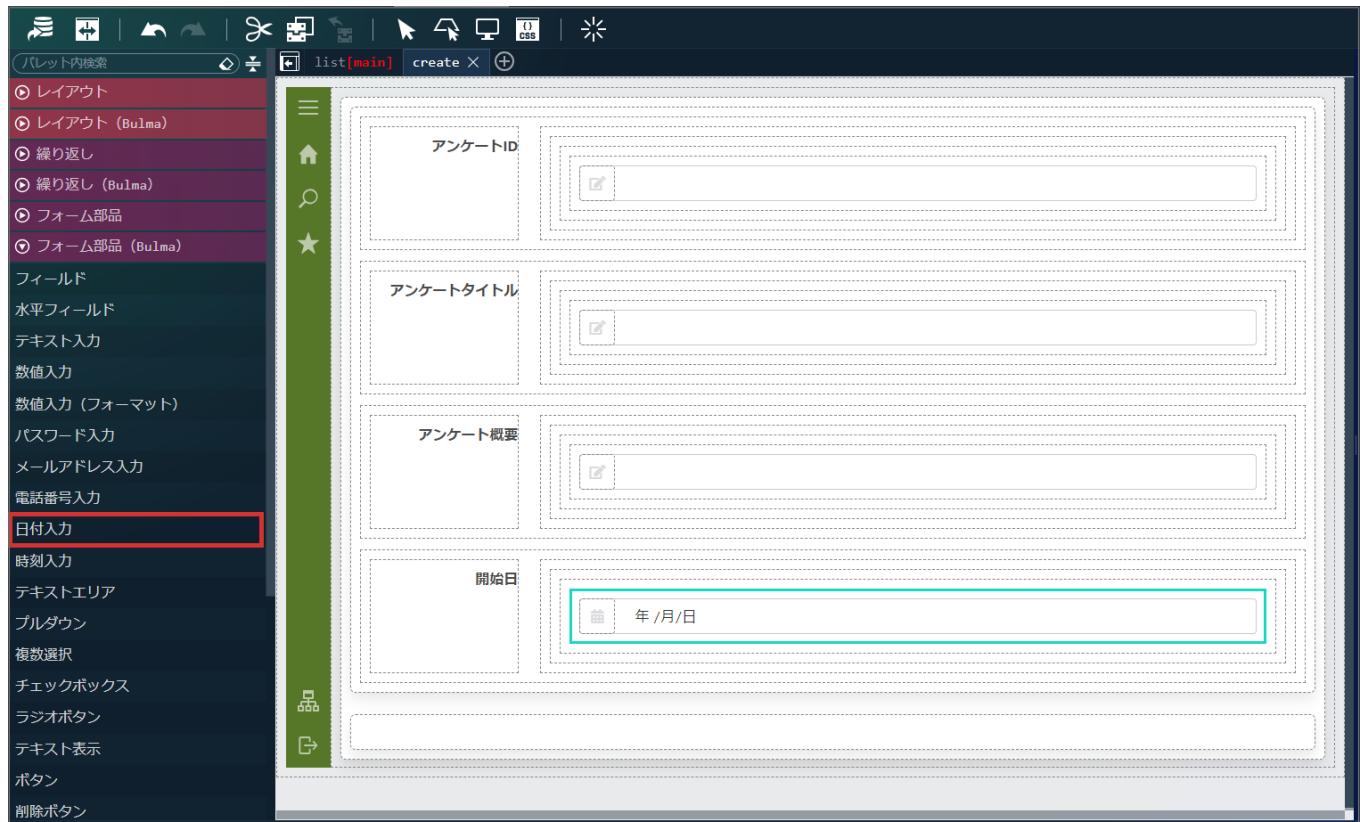
画面左側のパレットの「フォーム部品（Bulma）」 - 「水平フィールド」を選択し、「アンケート概要」の入力欄の下に配置します。

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。

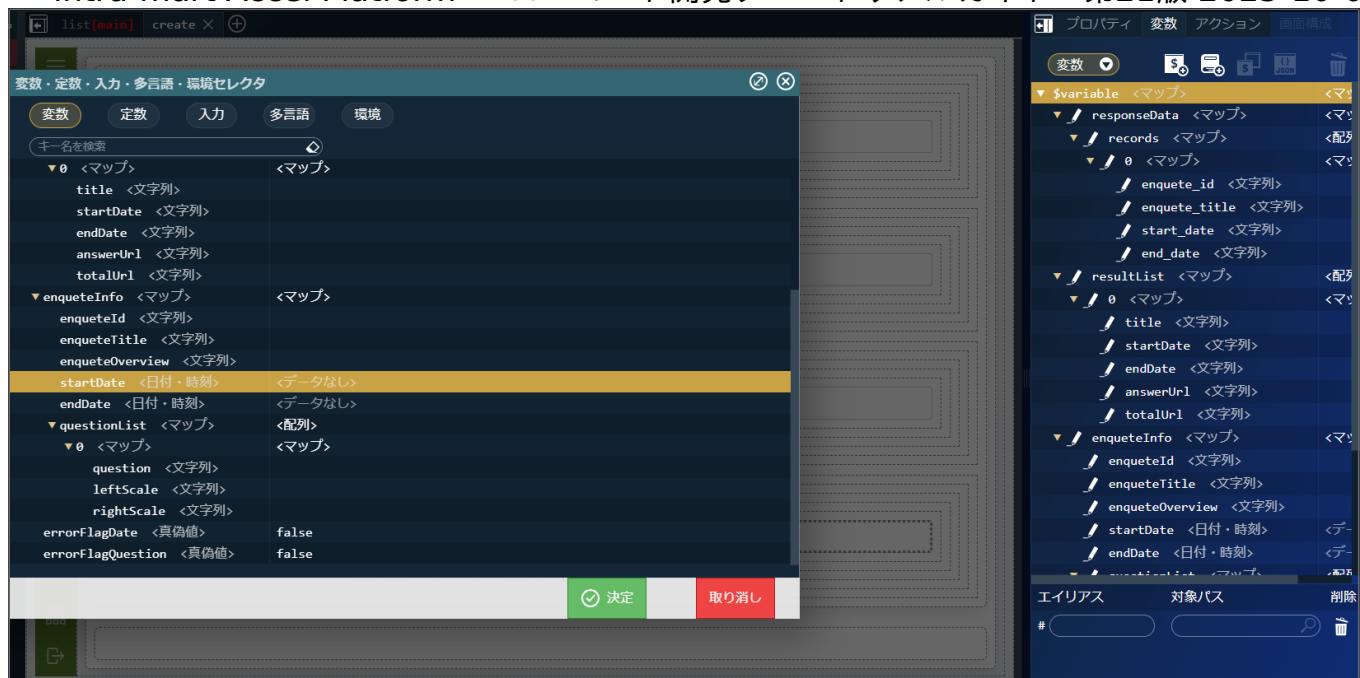


続けて、アンケートIDの入力欄用のテキストボックスを配置します。

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。

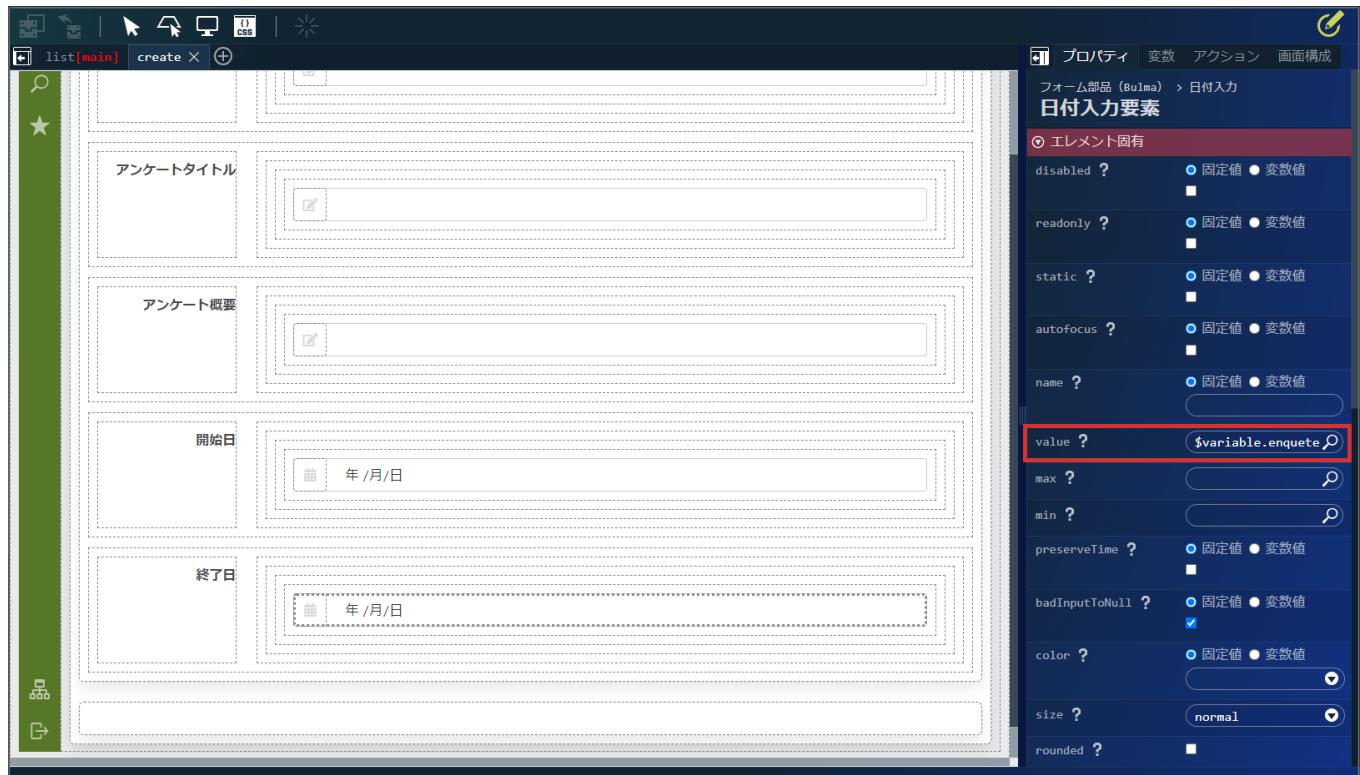


配置した「日付入力」の「日付入力要素」を選択し、「プロパティ」タブ - 「エレメント固有」 - 「value」の を選択します。
'\$variable > enqueteInfo > startDate' を選択し「決定」ボタンをクリックして変数を設定します。



同様の手順で、「終了日」の入力欄を作成します。

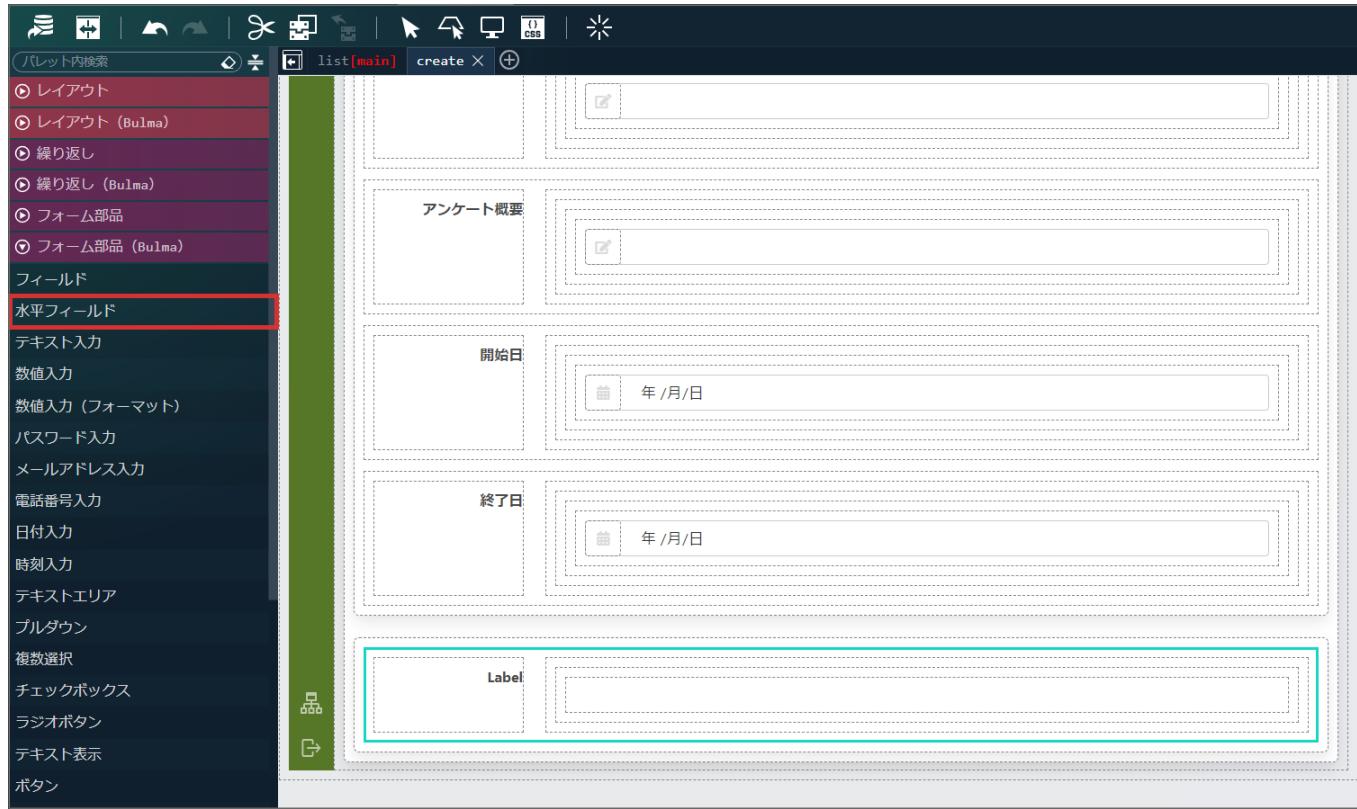
日付入力要素の「プロパティ」タブ - 「エレメント固有」 - 「value」には「\$variable.enqueueInfo.endDate」を設定してください。



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。



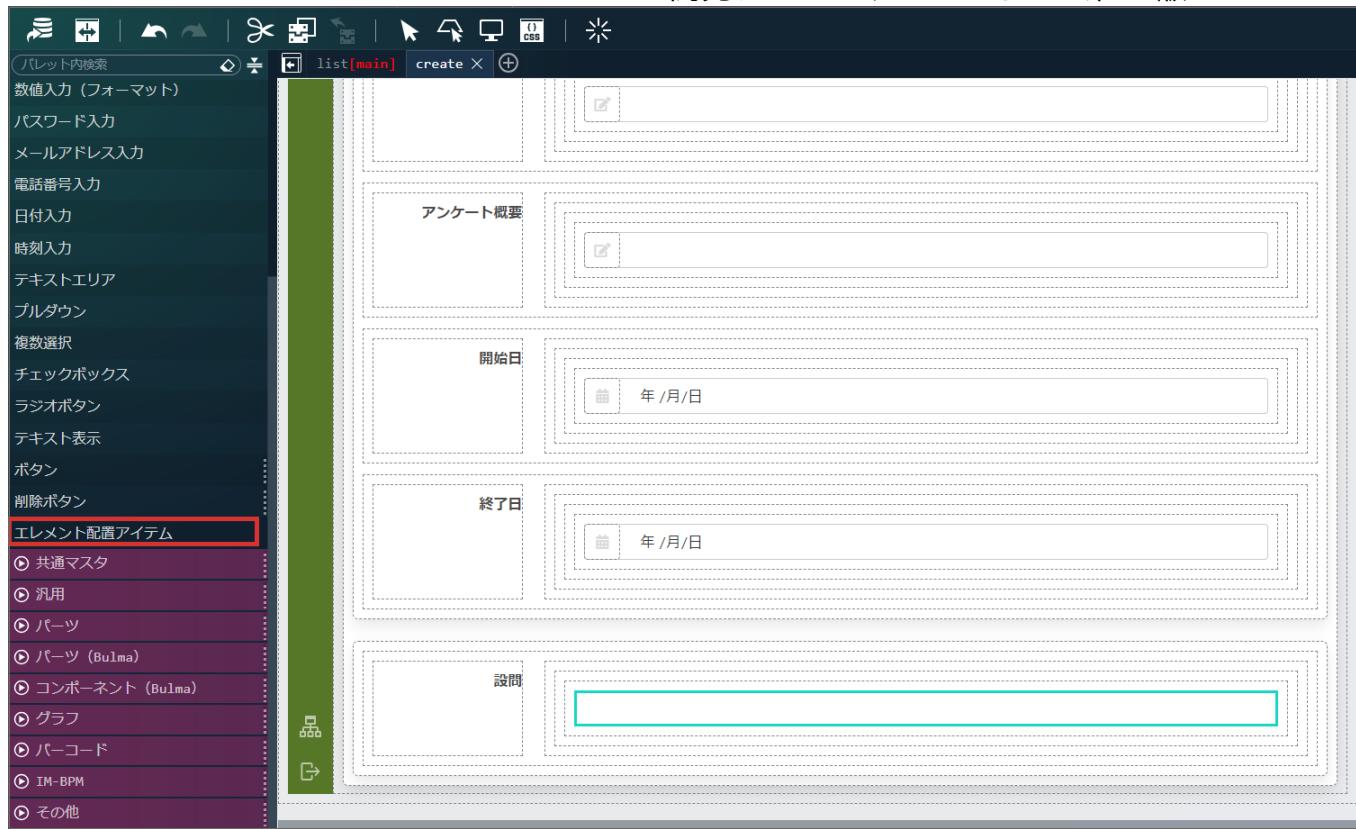
配置した「水平フィールド」の左側にある「水平フィールドラベル」を選択します。

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」を設定します。

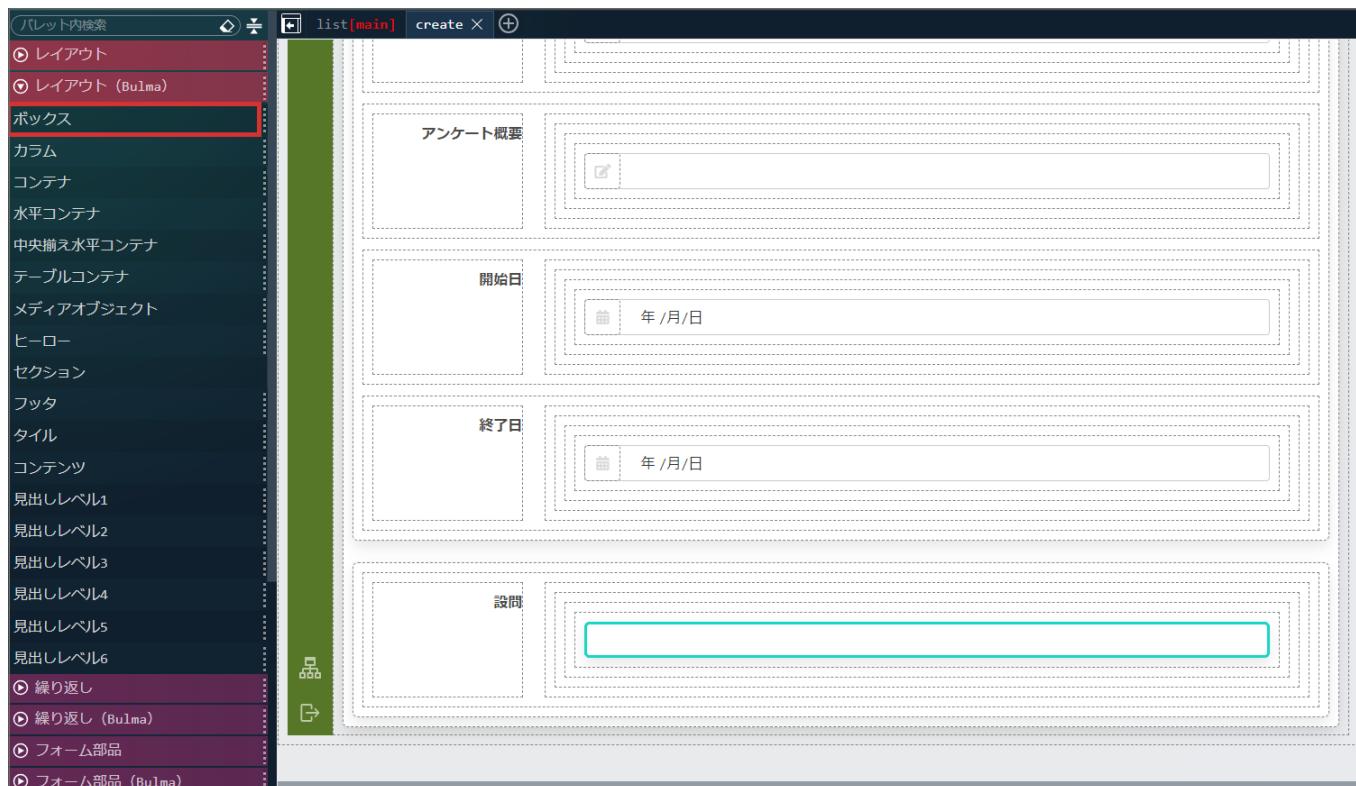


続けて、設問内容の入力欄を作成します。

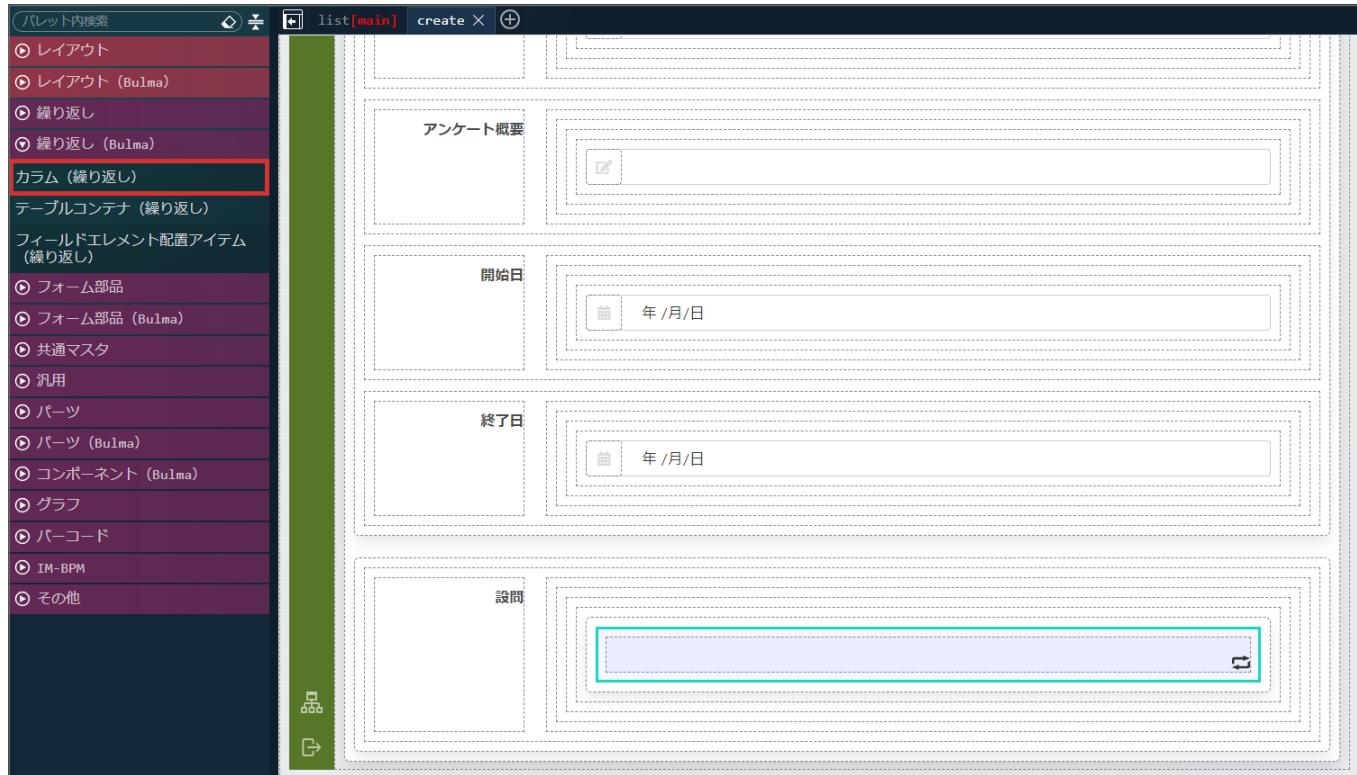
画面左側のパレットの「フォーム部品 (Bulma)」 - 「エレメント配置アイテム」を選択し、「水平フィールド」の右側に配置します。



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。

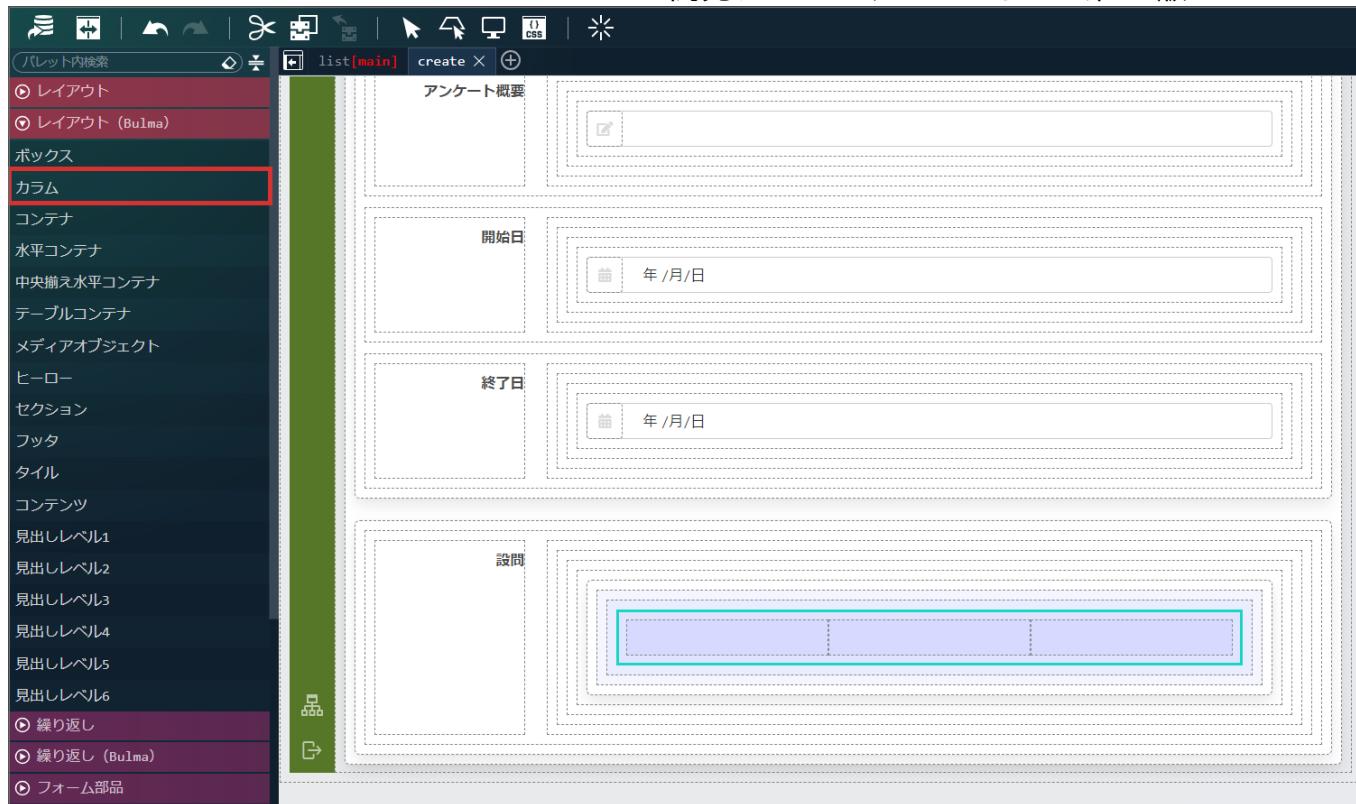
続けて、「カラム（繰り返し）」の「カラムアイテム（繰り返し）」を選択し、「エレメント固有」 - 「list」の から「\$variable > enqueteInfo > questionList」を選択して設定します。

「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。

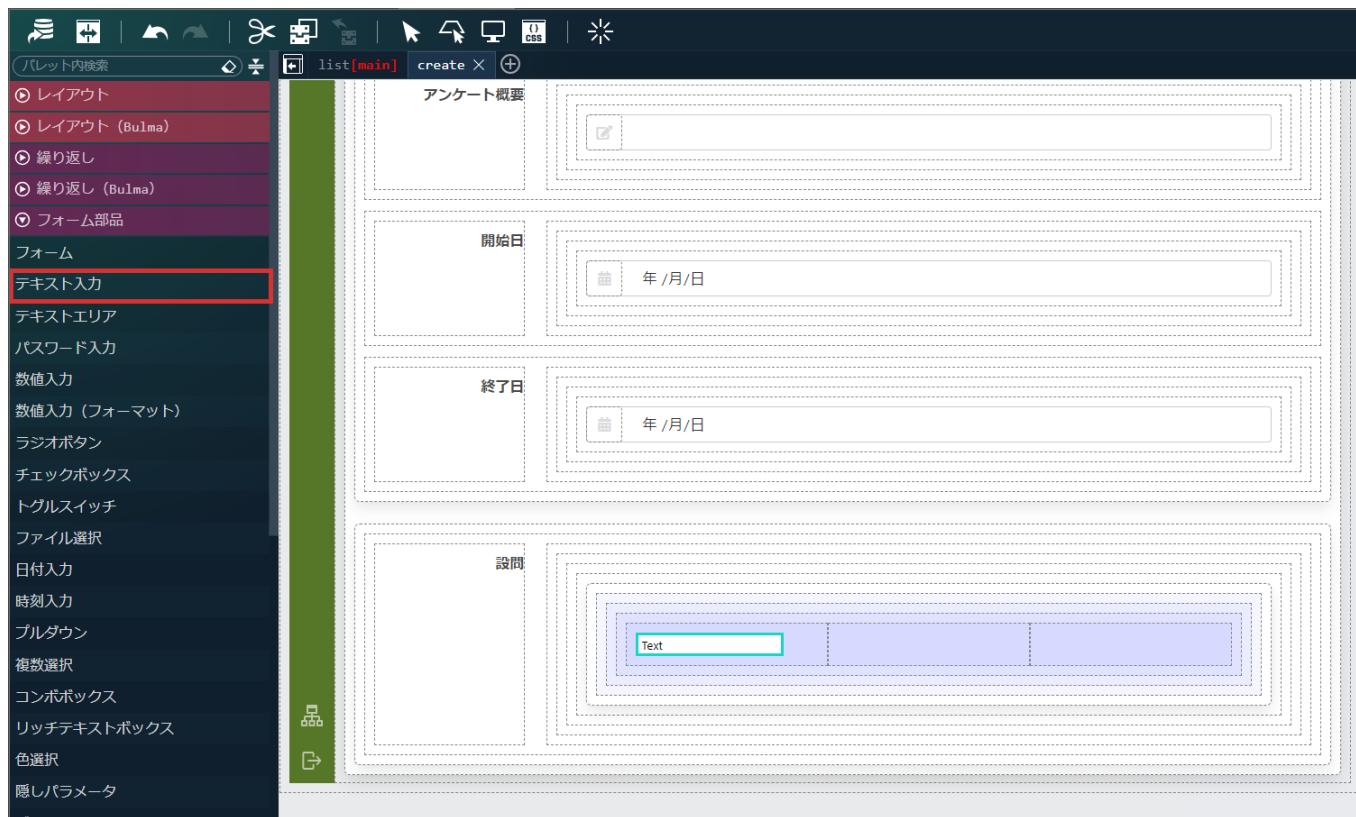
これで、「\$variable > enqueteInfo > questionList」の配列の長さによって、カラムが追加されるようになりました。

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。

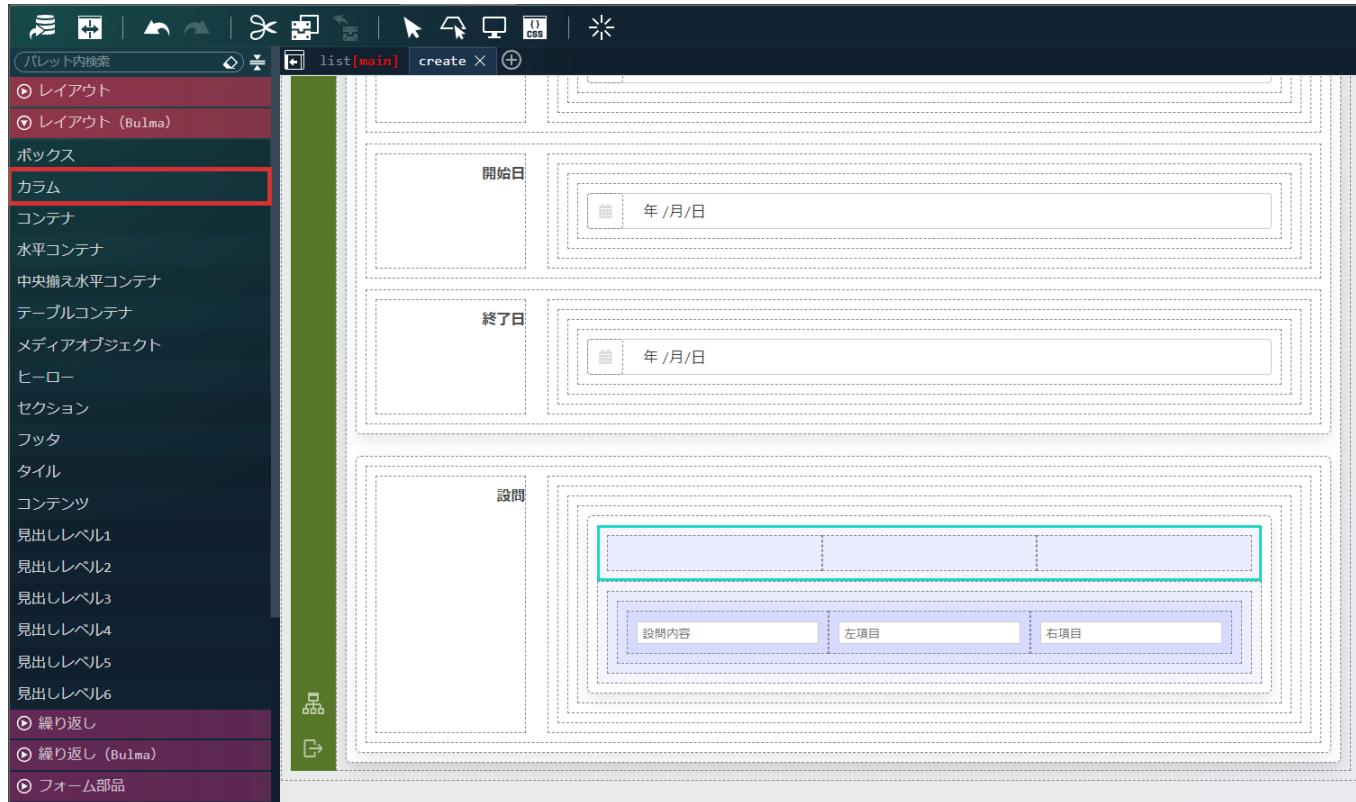
配置した「テキスト入力」の「幅・高さ」 - 「横幅」に「100%」を指定します。

同様の手順で「左項目」、「右項目」の入力欄を作成します。

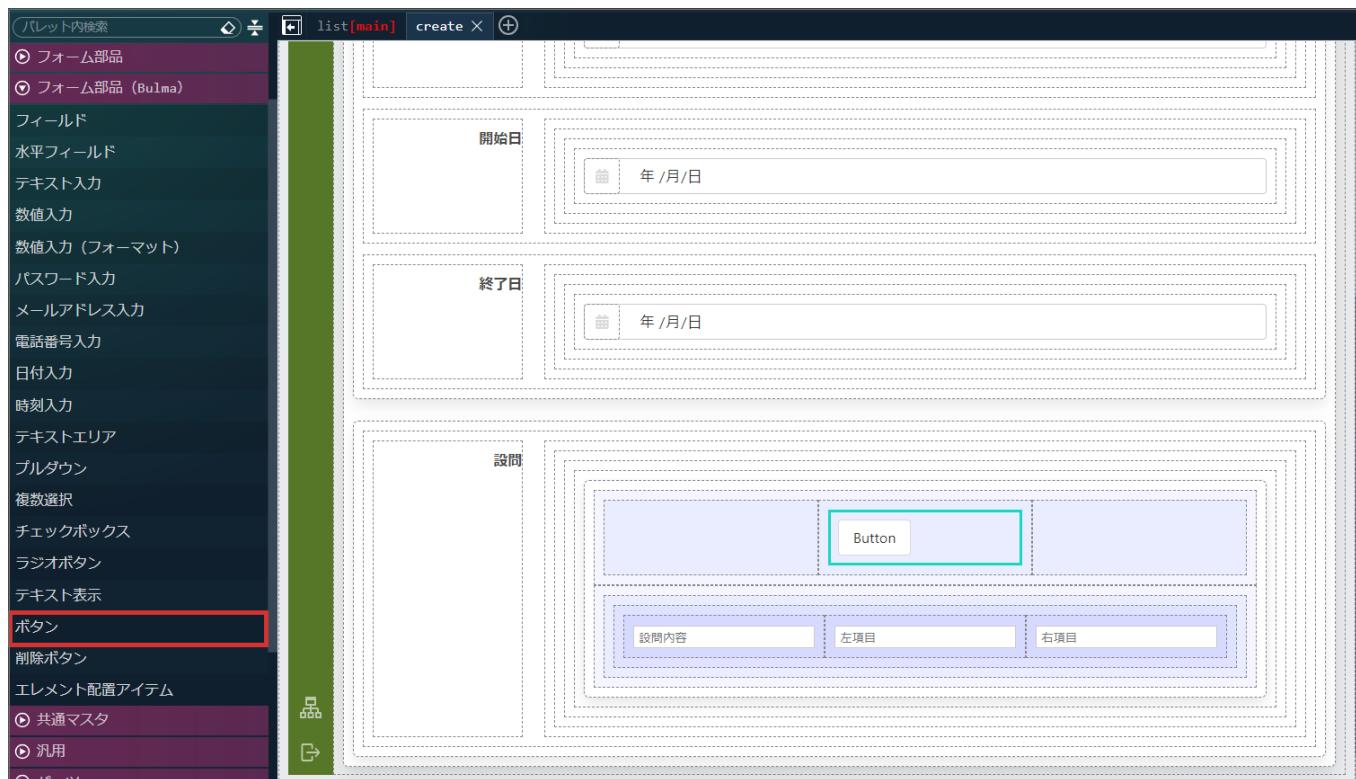
「テキスト入力」の「エレメント固有」 - 「value」にはそれぞれ、「左項目」には
「\$variable.enqueteInfo.questionList[\$index].leftScale」を「右項目」には
「\$variable.enqueteInfo.questionList[\$index].rightScale」を設定してください。

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。

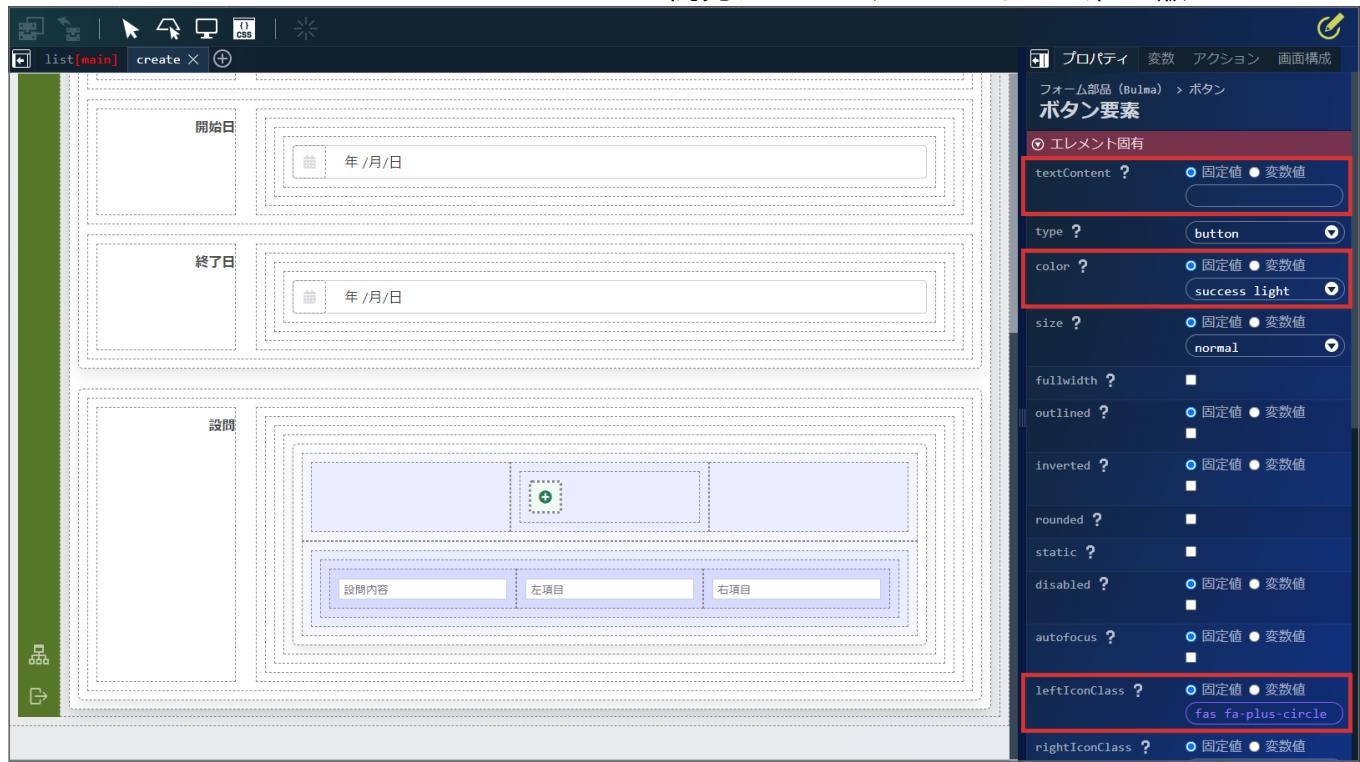
画面左側のパレットの「レイアウト (Bulma)」 - 「カラム」を選択し、設問内容の入力欄を配置したボックスの上に配置します。



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。

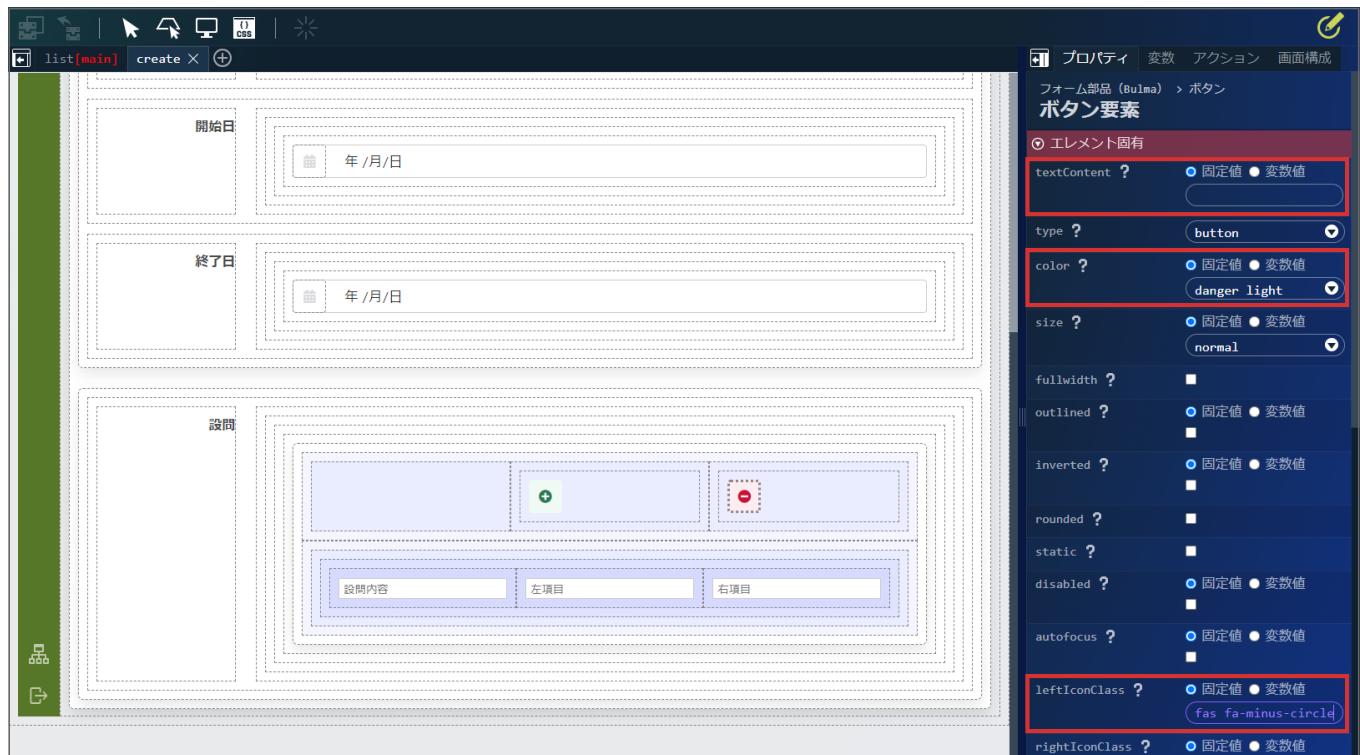


同様に行削除ボタンを作成します。

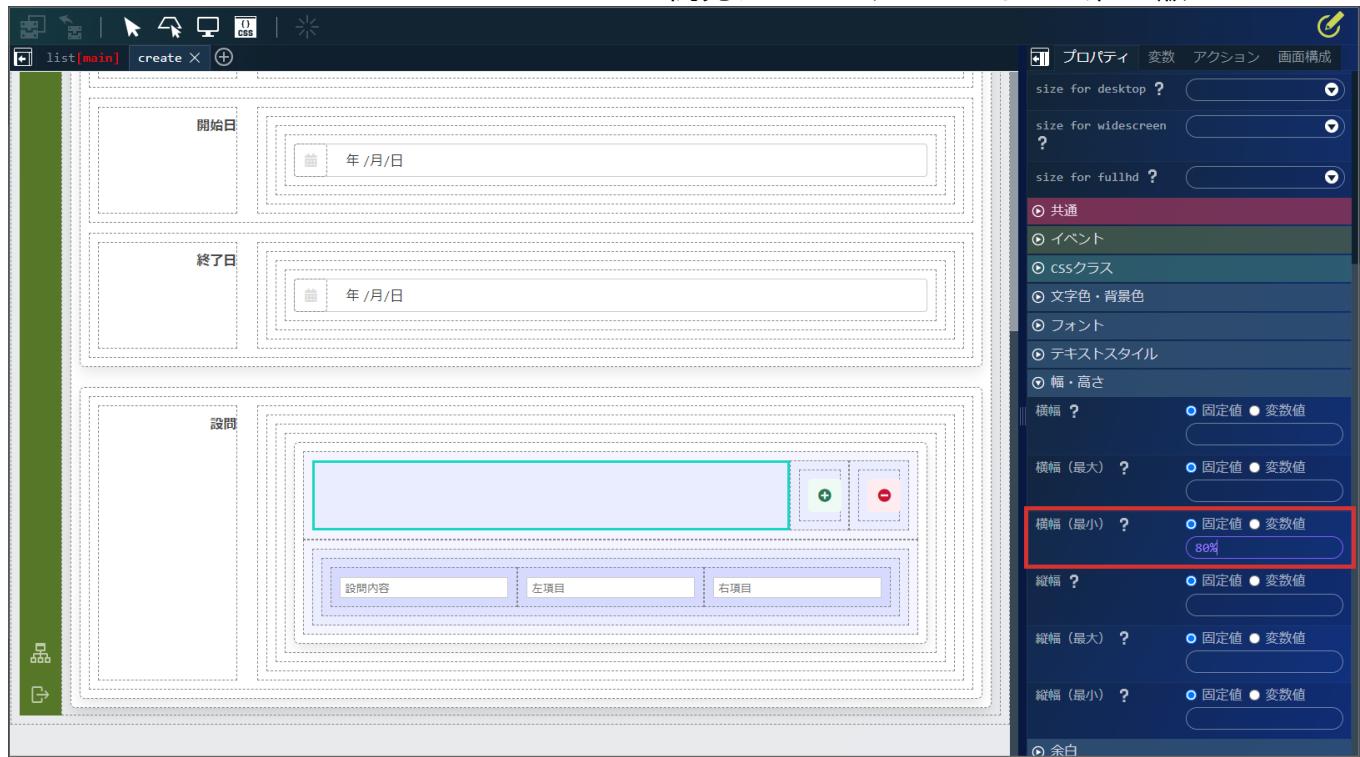
画面左側のパレットの「フォーム部品（Bulma）」 - 「ボタン」を選択し、右端のカラムに配置します。

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。

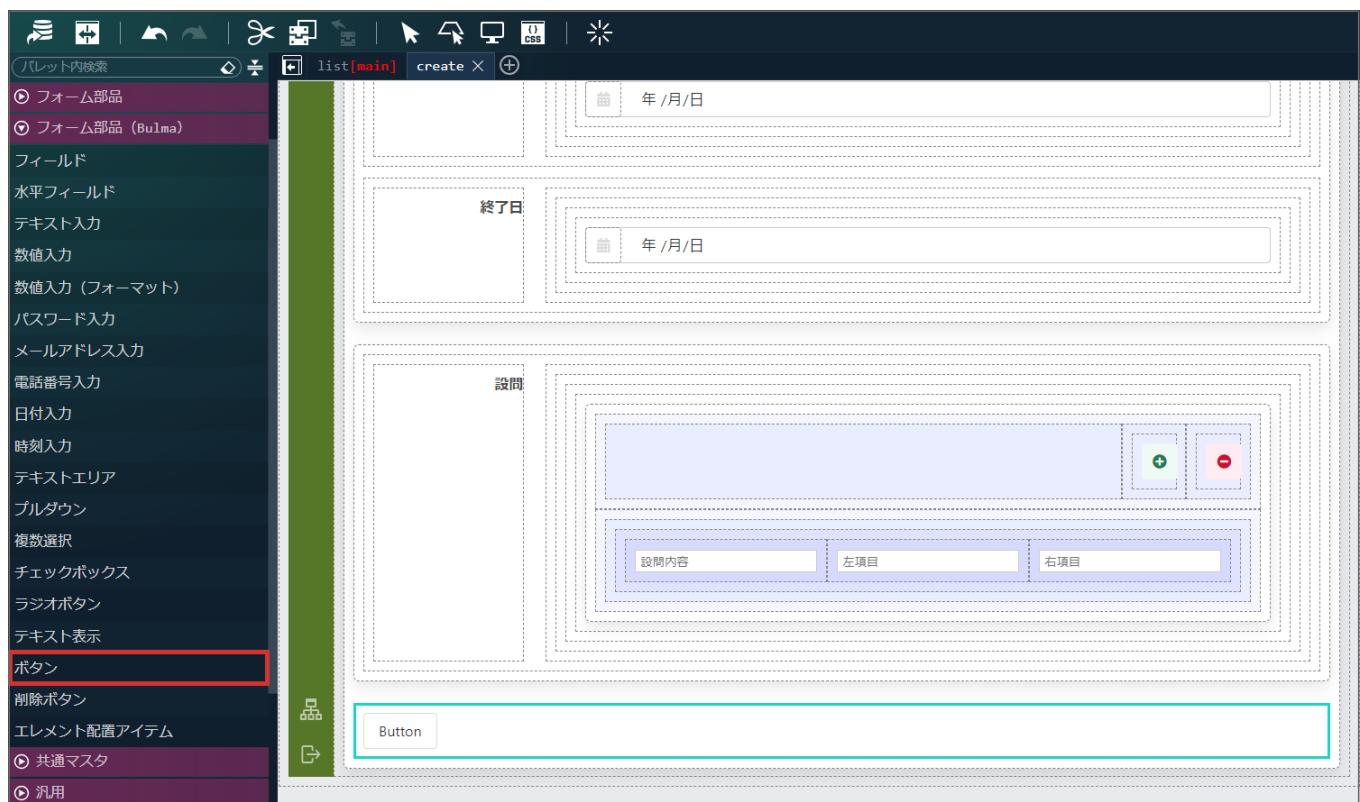


表示位置を調整するため、左端のカラムを選択し、「幅・高さ」 - 「横幅（最小）」に「80%」を指定します。

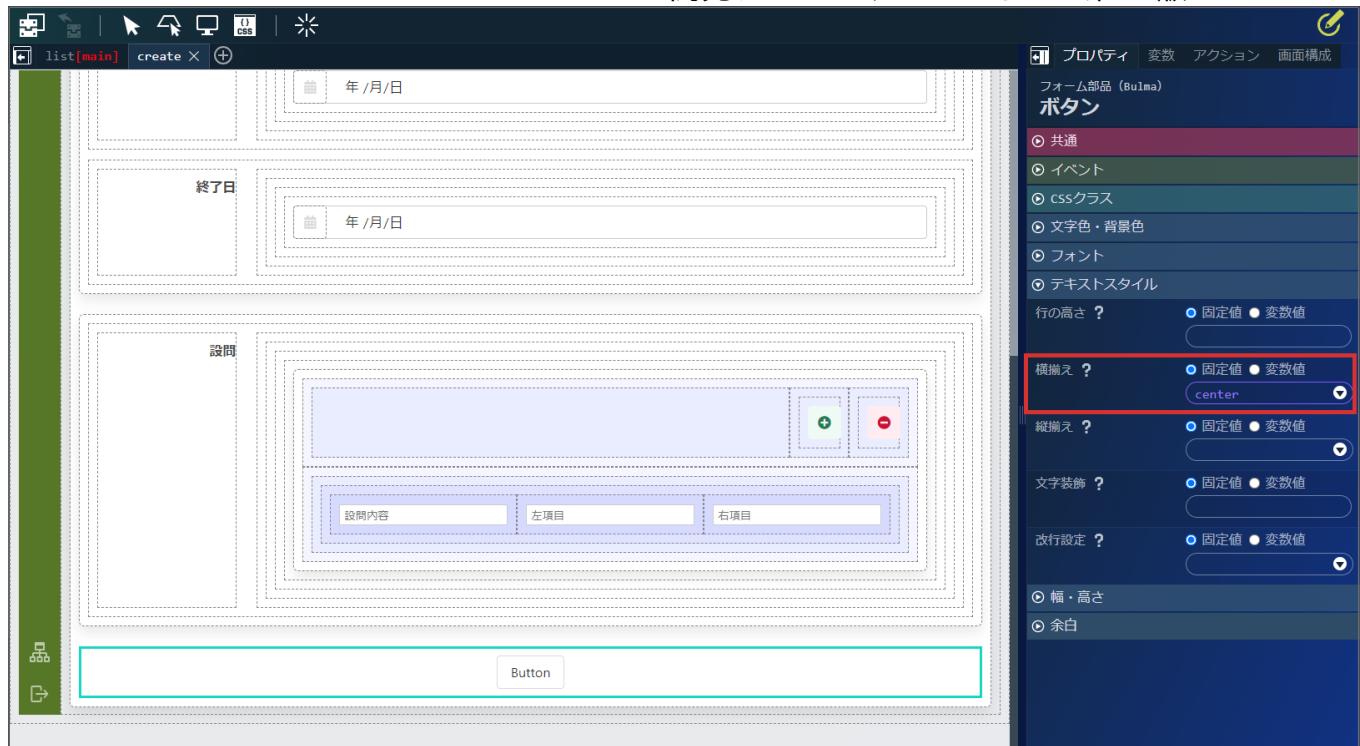


最後に、アンケートの登録ボタンを配置します。

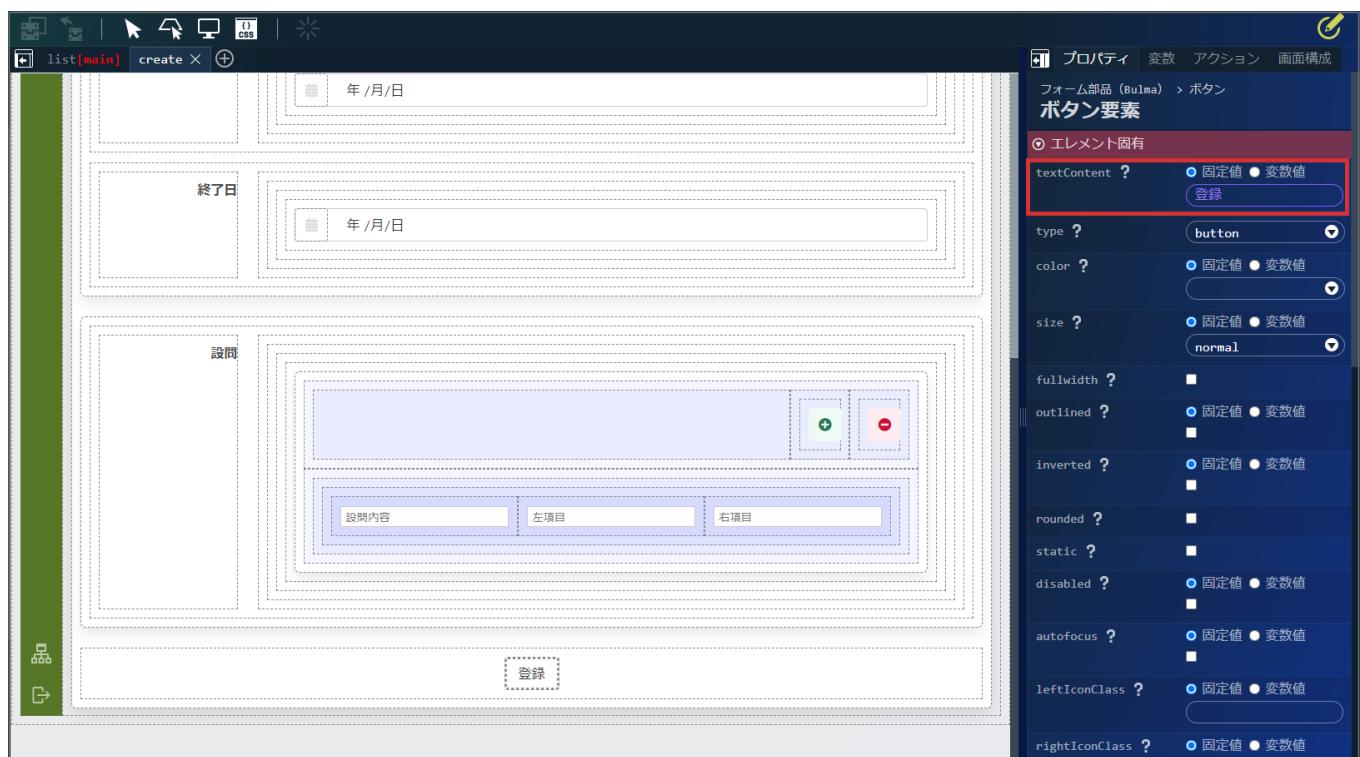
画面左側のパレットの「フォーム部品 (Bulma)」 - 「ボタン」を選択し、設問内容の入力欄を配置していた「ボックス」の一番下に配置します。



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

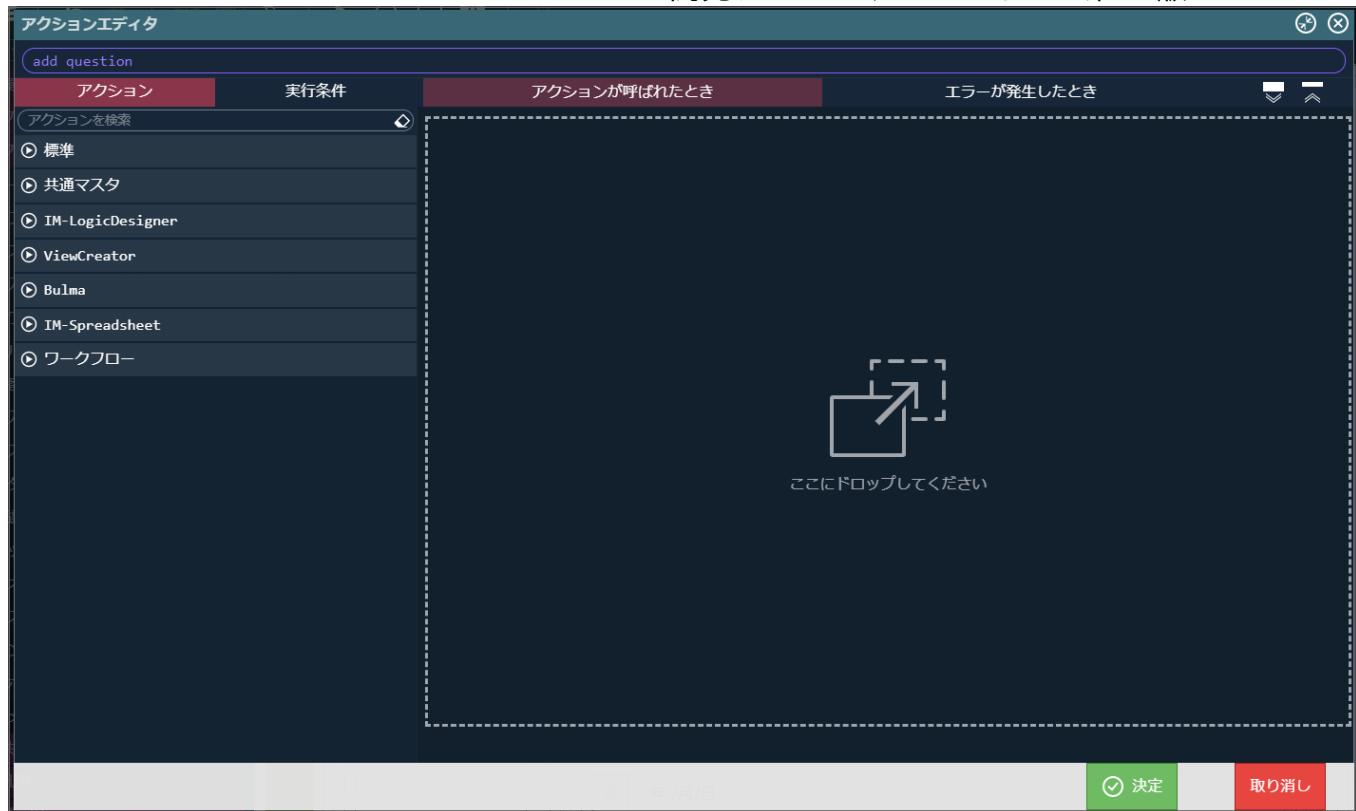
次に、画面で実行されるアクションを作成します。

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。

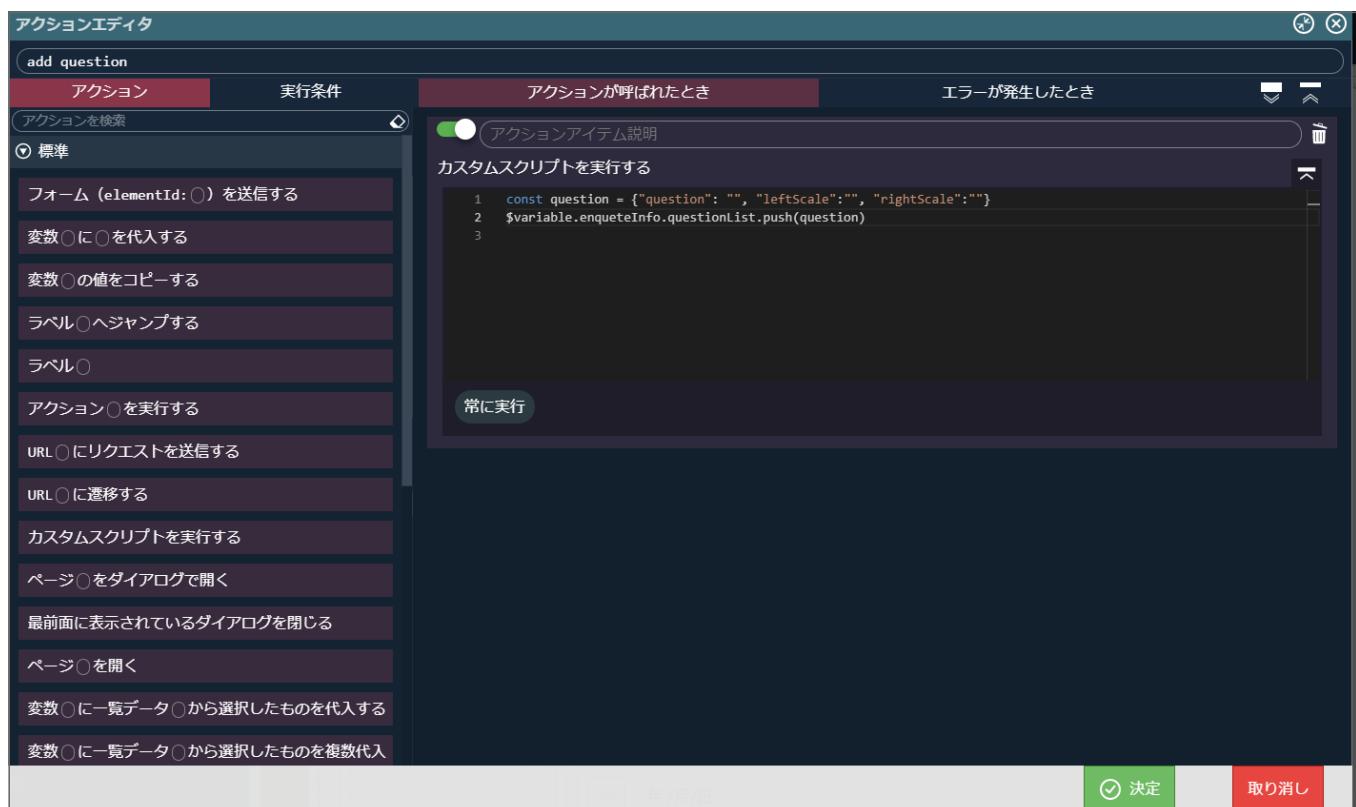
アクションエディタ上部の名称を「新しいアクション #1」を「add question」に変更します。



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。

アクションエディタ

remove question

アクション 実行条件

アクションを検索

標準 共通マスター IM-LogicDesigner ViewCreator Bulma IM-Spreadsheet ワークフロー

アクションが呼ばれたとき エラーが発生したとき

ここにドロップしてください

決定 取消し

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()
```

アクションエディタ

remove question

アクション 実行条件

アクションを検索

標準

カスタムスクリプトを実行する

1 \$variable.enqueteInfo.questionList.pop()

常に実行

決定 取消し

次に、アンケートを登録するアクションを作成します。

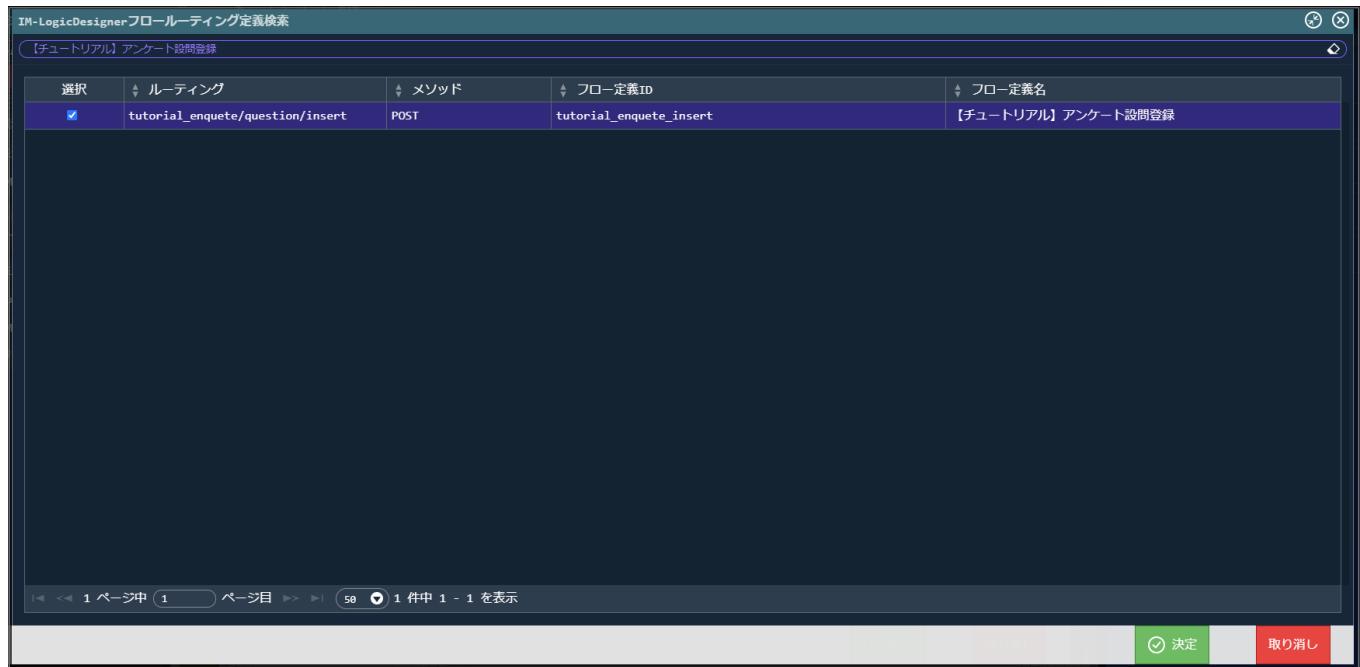
をクリックしアクションエディタを表示します。

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 フローリーティング○にリクエストを送信する」の をクリックし、フローリ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。



業務ロジックへ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。

変数・定数・入力・多言語・環境セレクタ



変数 定数 入力 多言語 環境

| | |
|------------------------------------|---------|
| <code>start_date</code> <文字列> | |
| <code>end_date</code> <文字列> | |
| ▼ resultList <マップ> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| <code>title</code> <文字列> | |
| <code>startDate</code> <文字列> | |
| <code>endDate</code> <文字列> | |
| <code>answerUrl</code> <文字列> | |
| <code>totalUrl</code> <文字列> | |
| ▼ enqueteInfo <マップ> | <マップ> |
| <code>enqueteId</code> <文字列> | |
| <code>enqueteTitle</code> <文字列> | |
| <code>enqueteOverview</code> <文字列> | |
| <code>startDate</code> <日付・時刻> | <データなし> |
| <code>endDate</code> <日付・時刻> | <データなし> |
| ▼ questionList <マップ> | <配列> |
| ▼ 0 <マップ> | <マップ> |
| <code>question</code> <文字列> | |
| <code>leftScale</code> <文字列> | |
| <code>rightScale</code> <文字列> | |

決定 取り消し

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○を開く」を先ほど配置した「IM-LogicDesigner フローリーティング○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。

アクションエディタ

create enquete

| 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 |
|---|------|--------------|------------|
| アクションを検索 | | | |
| 変数○に○を代入する | | | |
| 変数○の値をコピーする | | | |
| ラベル○へジャンプする | | | |
| ラベル○ | | | |
| アクション○を実行する | | | |
| URL○にリクエストを送信する | | | |
| URL○に遷移する | | | |
| カスタムスクリプトを実行する | | | |
| ページ○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○を開く | | | |
| 変数○に一覧データ○から選択したものを作成する | | | |
| 変数○に一覧データ○から選択したものを作成する | | | |
| 表示中の画面を閉じる | | | |
| メッセージ○をアラートダイアログで表示する | | | |
| アクションアイテム説明 | | | |
| IM-LogicDesigner フローリーティング [tutorial_enquete/question/insert] にリクエストを送信する | | | |
| リクエストパラメータ IN \$variable.enqueteInfo | | | |
| 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 | | | |
| アクションアイテム説明 | | | |
| ページ list [] を開く | | | |
| 常に実行 | | | |

決定 取り消し

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○を実行する」を「ページ○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。

アクションエディタ

create enquete

| アクション | 実行条件 |
|---------------------------|------|
| アクションを検索 | |
| 標準 | |
| フォーム (elementId:○) を送信する | |
| 変数○に○を代入する | |
| 変数○の値をコピーする | |
| ラベル○へジャンプする | |
| ラベル○ | |
| アクション○を実行する | |
| URL○にリクエストを送信する | |
| URL○に遷移する | |
| カスタムスクリプトを実行する | |
| ページ○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○を開く | |
| 変数○に一覧データ○から選択したものを代入する | |
| 変数○に一覧データ○から選択したものを複数代入する | |

アクションが呼ばれたとき

IM-LogicDesigner フローリーティング (tutorial_enquete/question insert) にリクエストを送信する

- リクエストパラメータ IN \$variable.enqueteInfo
- 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

アクションアイテム説明

ページ list ○を開く

常に実行

アクションアイテム説明

アクション ○ を実行する

常に実行 list add question remove question

決定 取り消し

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○を開く」を配置し、「ページ (create) を開く」に設定します。

アクションエディタ

open create page

| アクション | 実行条件 |
|---------------------------|------|
| アクションを検索 | |
| 変数○に○を代入する | |
| 変数○の値をコピーする | |
| ラベル○へジャンプする | |
| ラベル○ | |
| アクション○を実行する | |
| URL○にリクエストを送信する | |
| URL○に遷移する | |
| カスタムスクリプトを実行する | |
| ページ○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○を開く | |
| 変数○に一覧データ○から選択したものを代入する | |
| 変数○に一覧データ○から選択したものを複数代入する | |
| 表示中の画面を閉じる | |
| メッセージ: 他のアクションを実行する | |

アクションが呼ばれたとき

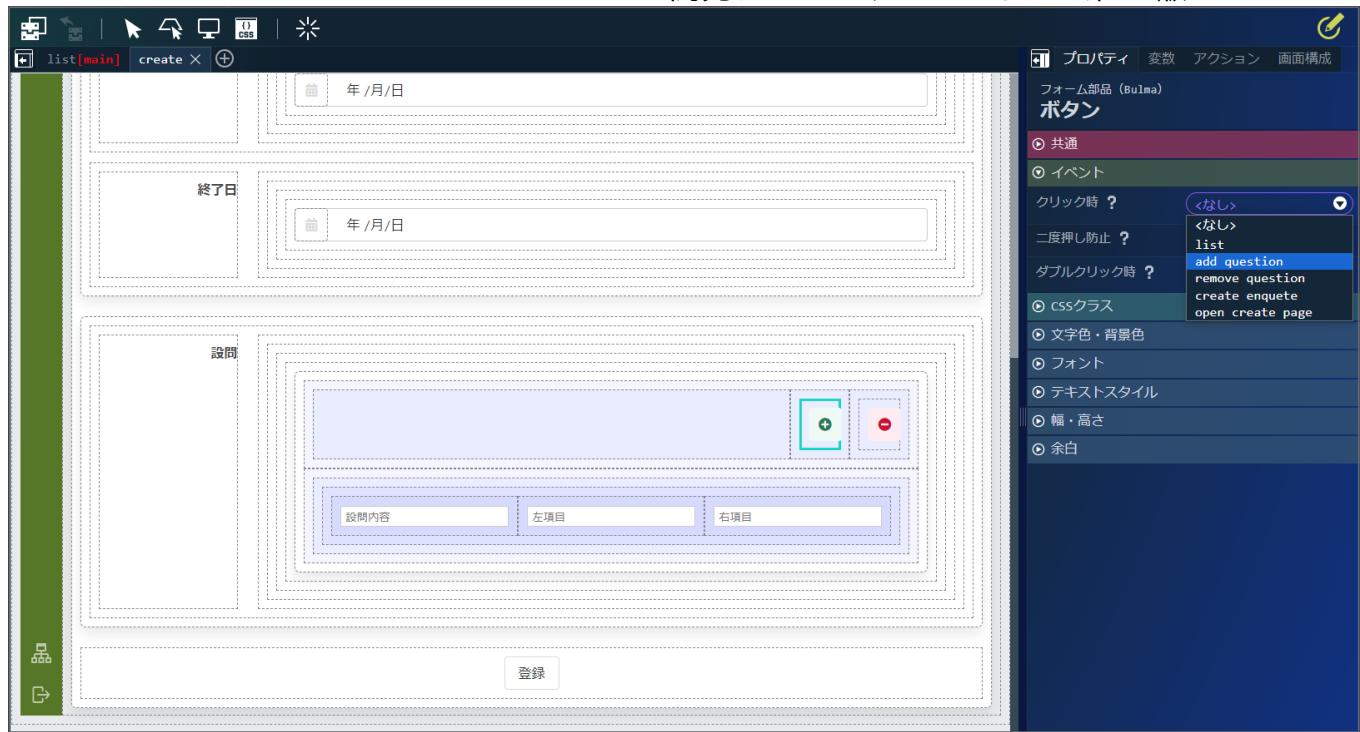
ページ ○ を開く

常に list create

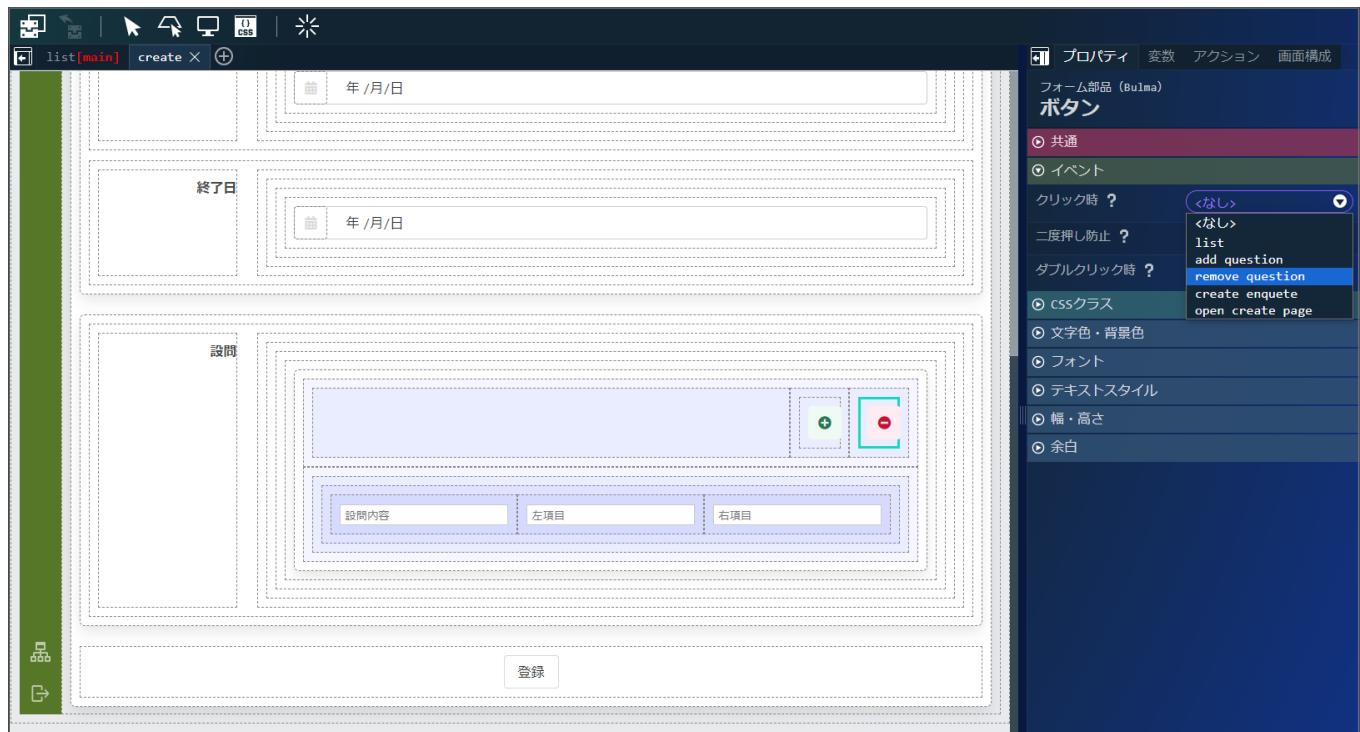
決定 取り消し

作成したアクションを各ボタンに設定します。

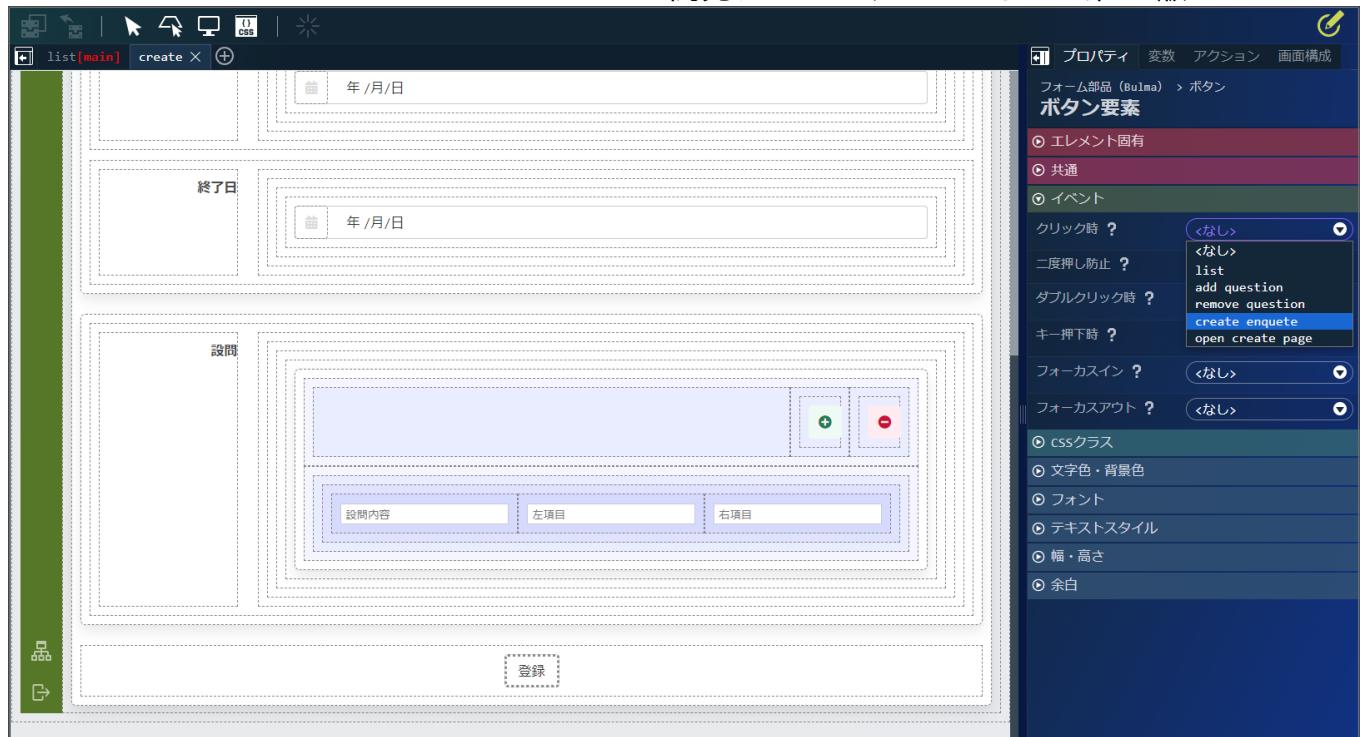
設問の入力欄の+アイコンを配置したボタンを選択し、「イベント」 - 「クリック時」に「add question」を設定します。



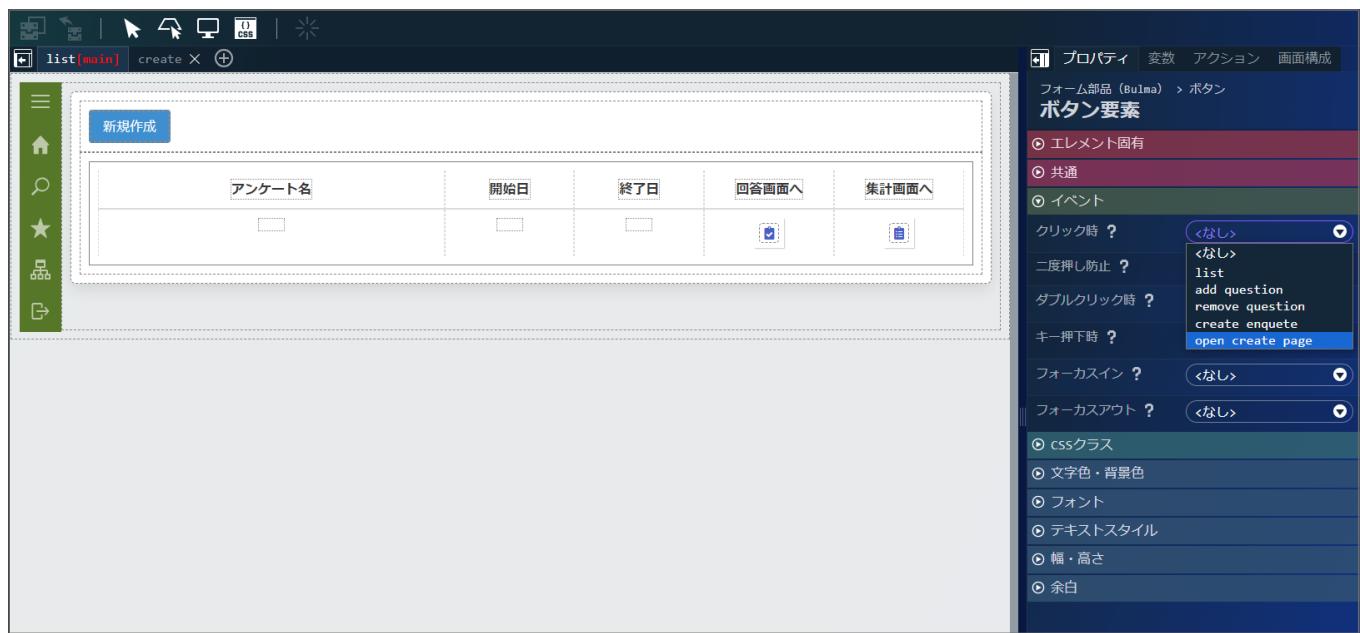
同様に、設問の入力欄のーアイコンを配置したボタンを選択し、「イベント」 - 「クリック時」に「remove question」を設定します。



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。動作確認を行いましょう。

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。

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 年/月/日

終了日 年/月/日

設問

登録

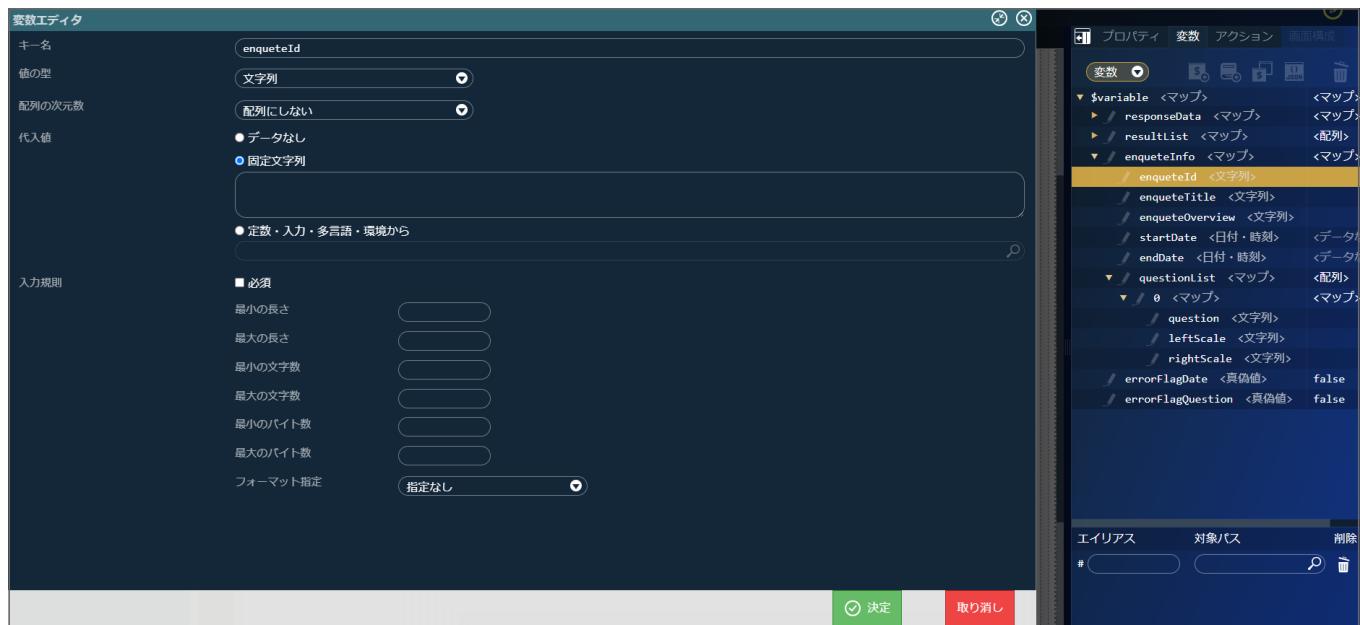
ここまででアンケートの一覧と登録が行える画面が作成できました。
しかし、このままでは設問項目を全て埋めなくても登録できたり、登録後再度アンケート登録画面を表示した際に前回登録した内容がそのままになっているたりします。
登録画面での入力チェックや、アンケート登録画面表示時の変数の初期化処理などを追加していきましょう。

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 | <input type="radio"/> データなし <input checked="" type="radio"/> 固定文字列 <div style="border: 1px solid #ccc; padding: 5px; height: 40px; margin-top: 10px;"></div> |
| <input type="radio"/> 定数・入力・多言語・環境から | |
| <input checked="" type="checkbox"/> 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 | |
| <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin-left: auto; margin-right: 0;"> 指定なし 数値形式 アルファベット+数値形式 ID形式 ユーザコード形式 Eメールアドレス形式 URL形式 電話番号形式 郵便番号形式 選択候補のみ 正規表現にマッチする形式 指定なし </div> | |
| <input checked="" type="button"/> 決定 <input type="button"/> 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 | <input type="radio"/> データなし <input checked="" type="radio"/> 固定文字列 <div style="border: 1px solid #ccc; padding: 5px; height: 40px; margin-top: 10px;"></div> |
| <input type="radio"/> 定数・入力・多言語・環境から | |
| <input checked="" type="checkbox"/> 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 | |
| <div style="border: 1px solid #ccc; padding: 5px; width: fit-content; margin-left: auto; margin-right: 0;"> 指定なし 数値形式 アルファベット+数値形式 ID形式 ユーザコード形式 Eメールアドレス形式 URL形式 電話番号形式 郵便番号形式 選択候補のみ 正規表現にマッチする形式 指定なし </div> | |
| <input checked="" type="button"/> 決定 <input type="button"/> 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○」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。

「変数〇が〇のとき」の変数欄を選択し、「\$variable > enqueteInfo > enqueteId」を選択します。

同様の手順で、「変数〇が〇のとき」を配置し、「\$variable > enqueteInfo > enqueteTitle」、「\$variable > enqueteInfo > enqueteOverview」、「\$variable > enqueteInfo > startDate」、「\$variable > enqueteInfo > endDate」を設定します。

The screenshot shows the Logic Designer interface with a workflow titled "create enquete". The workflow consists of several steps:

- アクション**: "アクションアイテム説明" (Action Item Description) with condition "標準" (Standard).
- 実行条件**: "実行条件を検索" (Search execution conditions), showing filters for "変数〇が〇のとき" (Variable〇 is〇), "変数〇が〇でないとき" (Variable〇 is not〇), and "上記カスタムスクリプトの返却値が true のとき" (Return value of the above custom script is true).
- アクションが呼ばれたとき**: "アクションアイテム説明" (Action Item Description) with condition "標準" (Standard). It includes a label "END" and a "ヘジャンプする" (Jump) action.
- エラーが発生したとき**: "アクションアイテム説明" (Action Item Description) with condition "標準" (Standard).
- 標準**: "いつれかの条件に一致" (One or more conditions are met) with multiple conditions for variable values.
- IM-LogicDesigner**: "IM-LogicDesigner フロールーティング [tutorial_enquete/question/insert] にリクエストを送信する" (Send request to IM-LogicDesigner flow [tutorial_enquete/question/insert]).
- 標準**: "ページ [list] を開く" (Open page [list]).
- 標準**: "アクション [list] を実行する" (Execute action [list]).

At the bottom right are "決定" (Decision) and "取り消し" (Cancel) buttons.

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も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。

キーナーを「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。

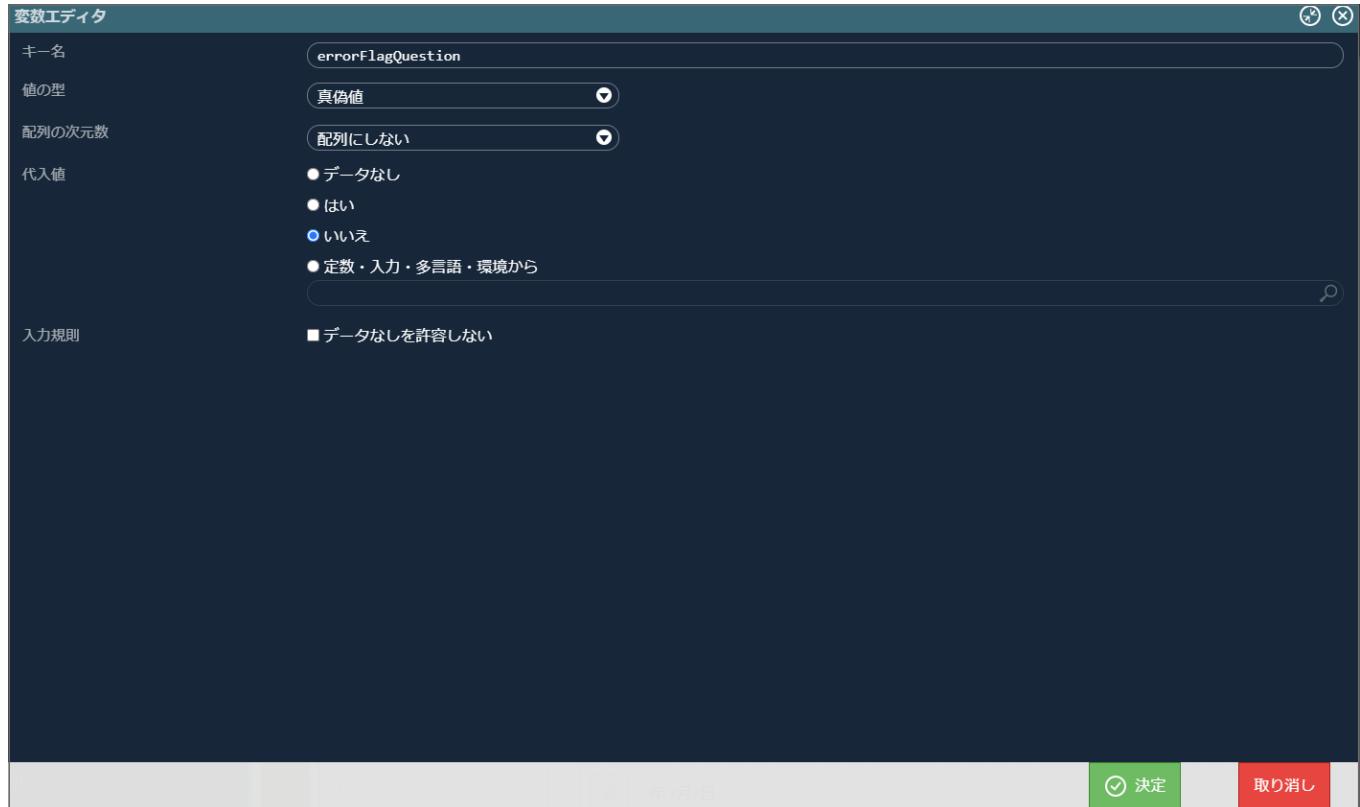
The screenshot shows the Variable Editor interface with the following settings:

- キー名**: "errorFlagDate"
- 値の型**: "真偽値" (Boolean)
- 配列の次元数**: "配列にしない" (Do not use arrays)
- 代入値**: "データなし" (No data) selected.
- 入力規則**: "データなしを許容しない" (Do not allow no data).

At the bottom right are "決定" (Decision) and "取り消し" (Cancel) buttons.

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、キーナーを「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。

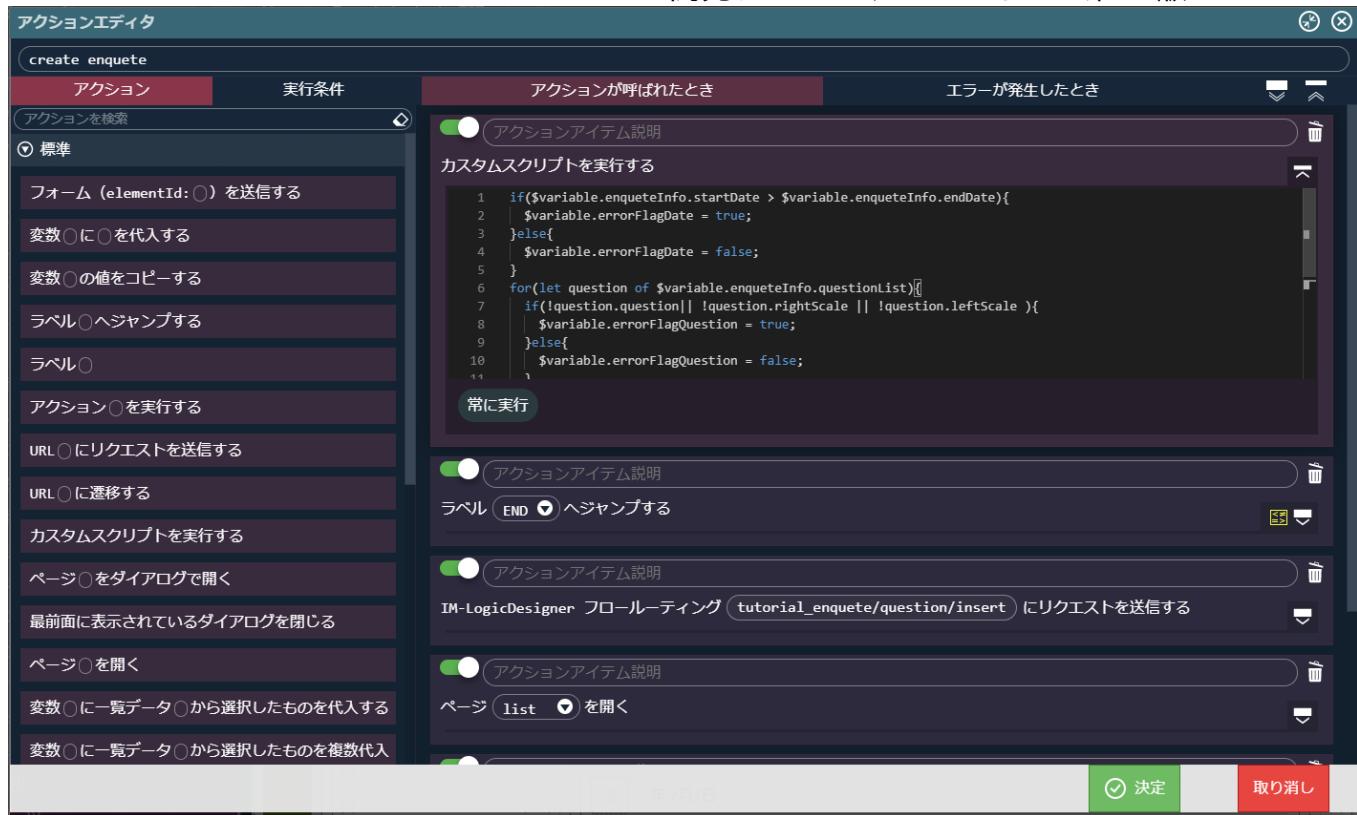
作成した変数を利用して入力チェックを行います。

「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。

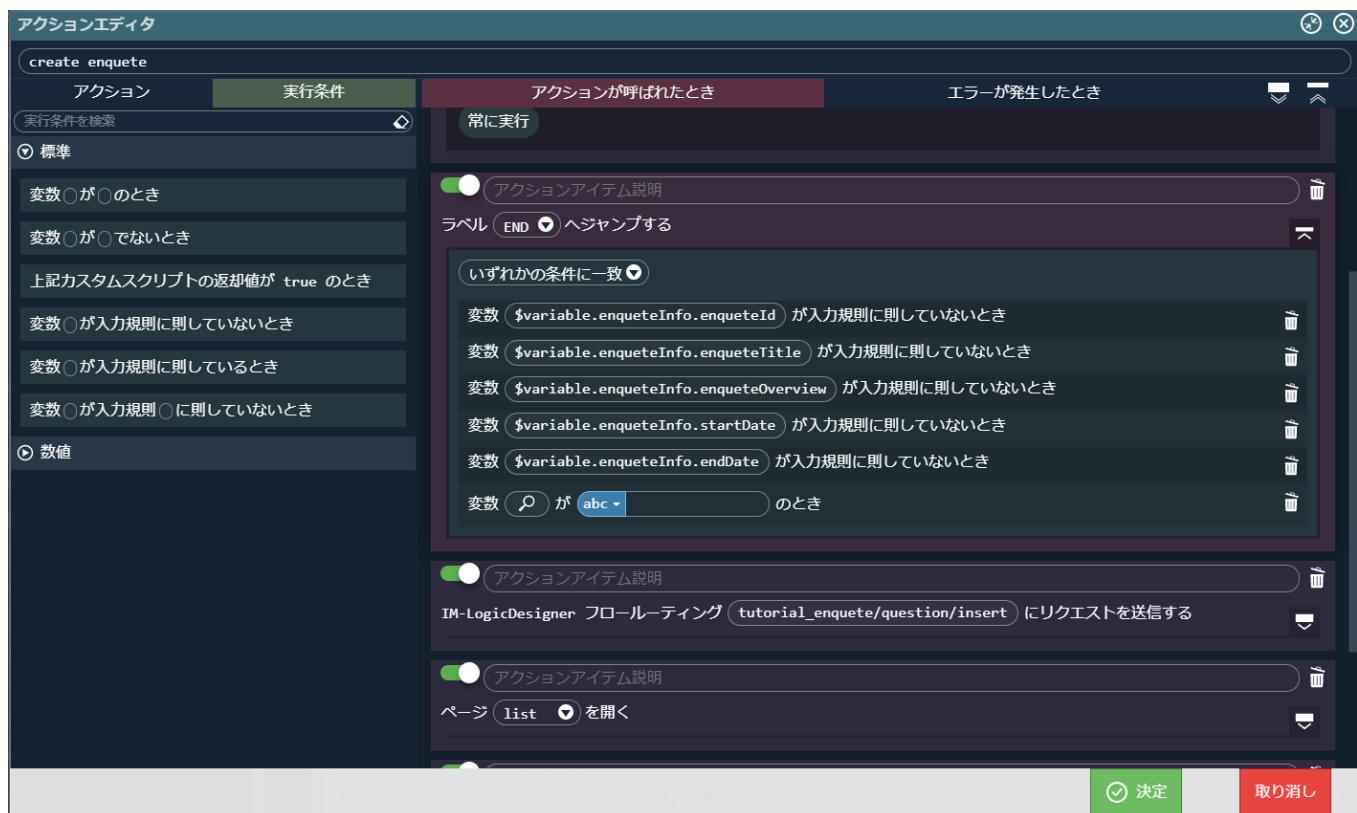
左側の「アクション」タブから「標準」 - 「カスタムスクリプトを実行する」を一番最初のアクションとして配置します。

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。

アクションエディタ

create_enquete

実行条件

常に実行

アクションが呼ばれたとき

変数 \$variable.errorFlagQuestion が \${env.const.true} のとき

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

ページ list を開く

決定 取り消し

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。

アクションエディタ

create_enquete

実行条件

常に実行

アクションが呼ばれたとき

変数 \$variable.errorFlagQuestion が \${env.const.true} のとき

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

ページ list を開く

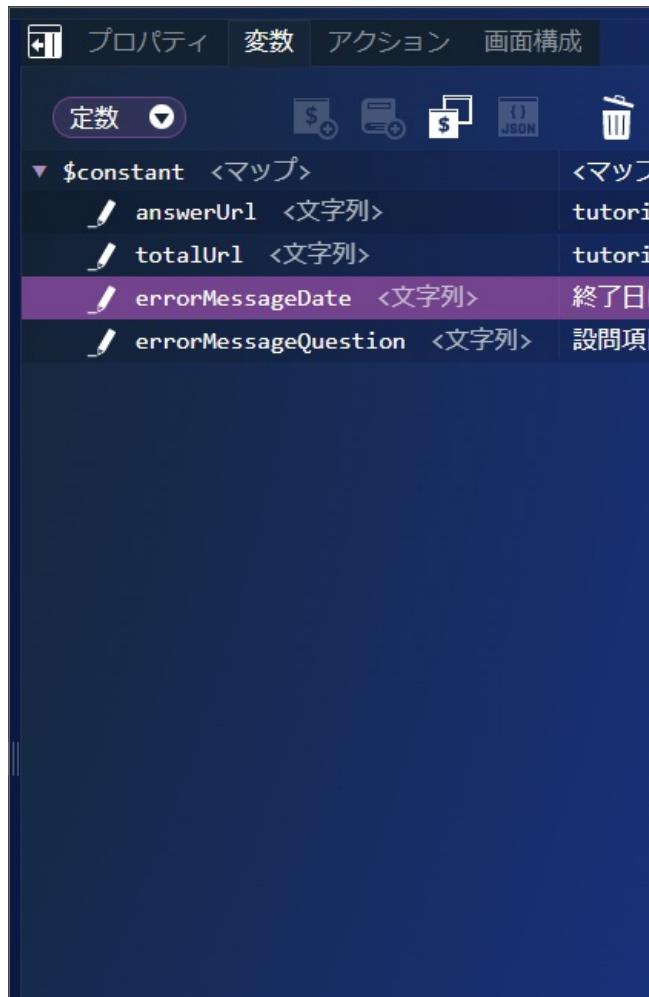
決定 取り消し

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いようになりました。

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。



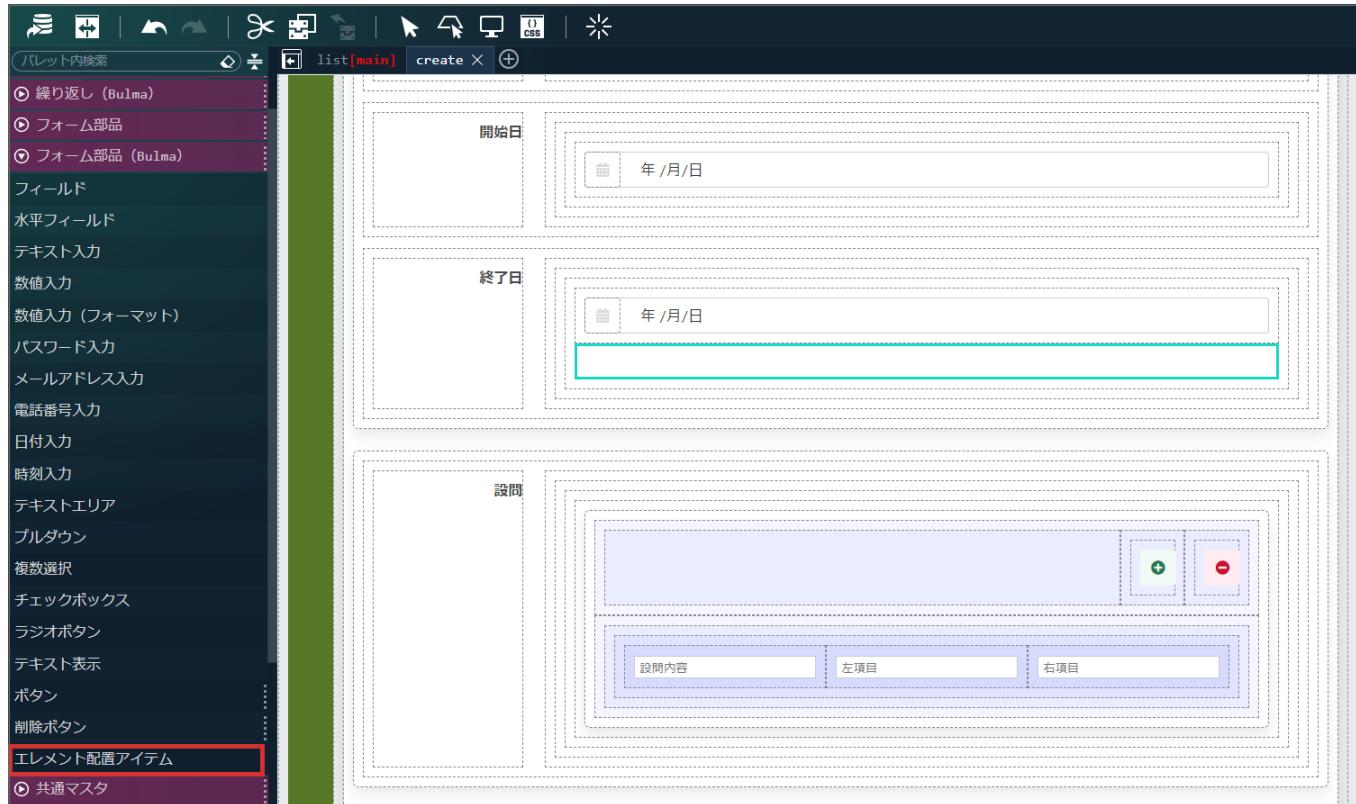
定数もJSON入力で以下を記載して設定できます。

```
"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。



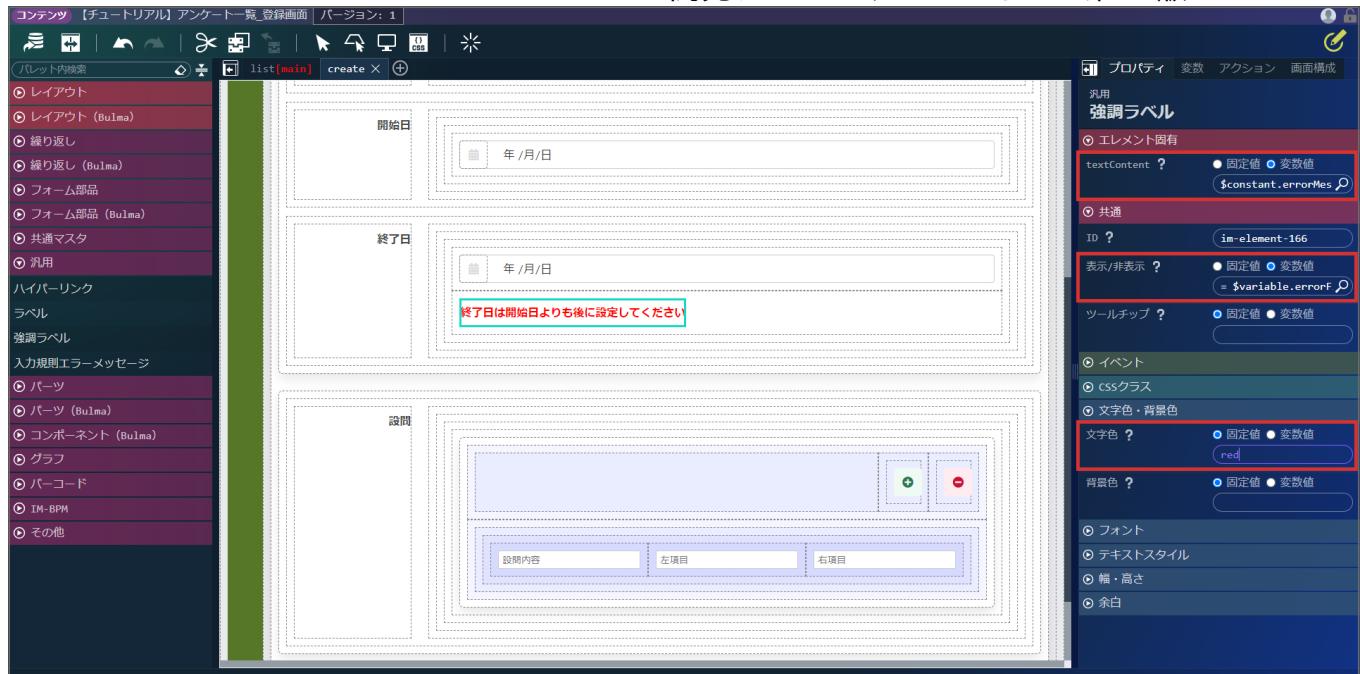
「汎用」 - 「強調ラベル」を選択し、先ほど配置した「エレメント配置アイテム」の中に配置します。



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

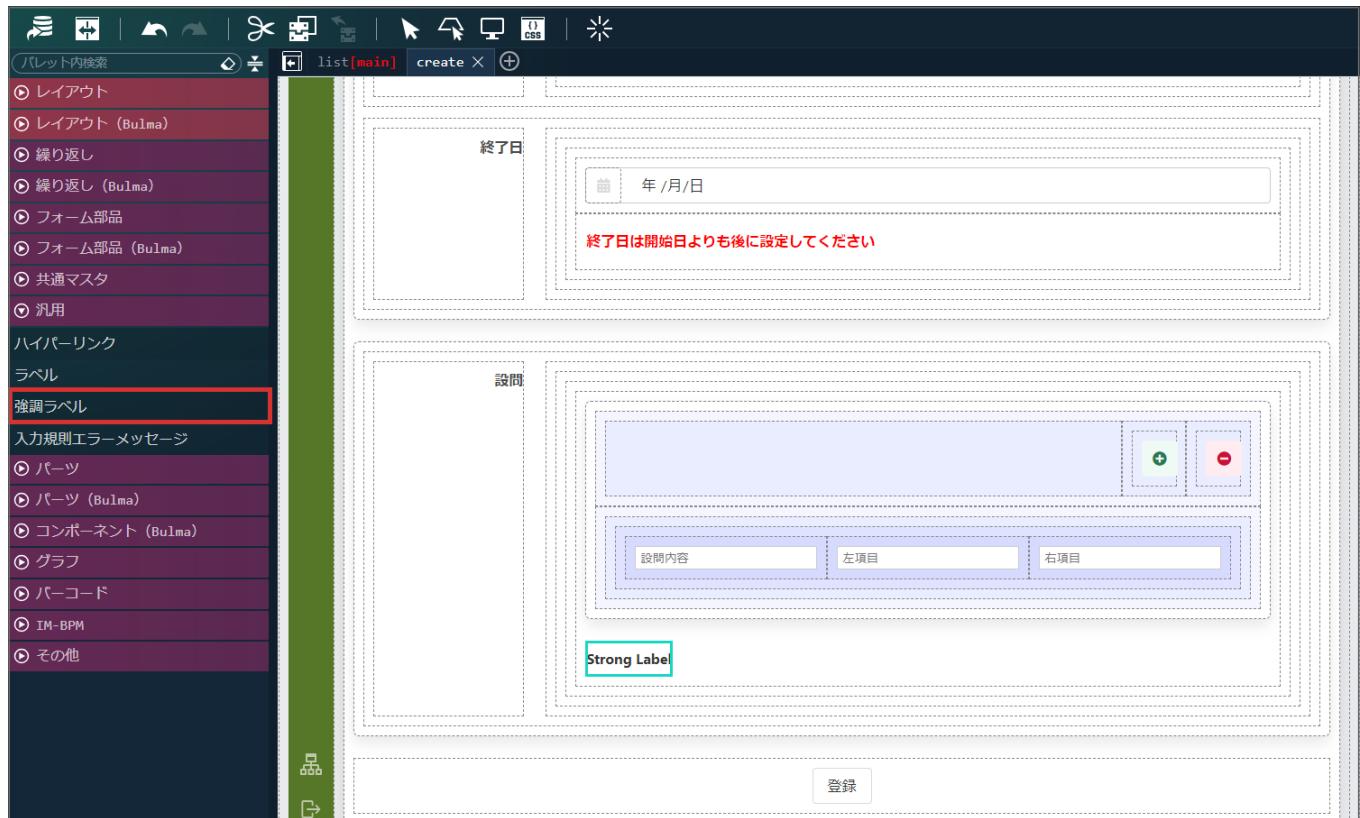
「\$constant.errorMessageDate」と設定します。

「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。



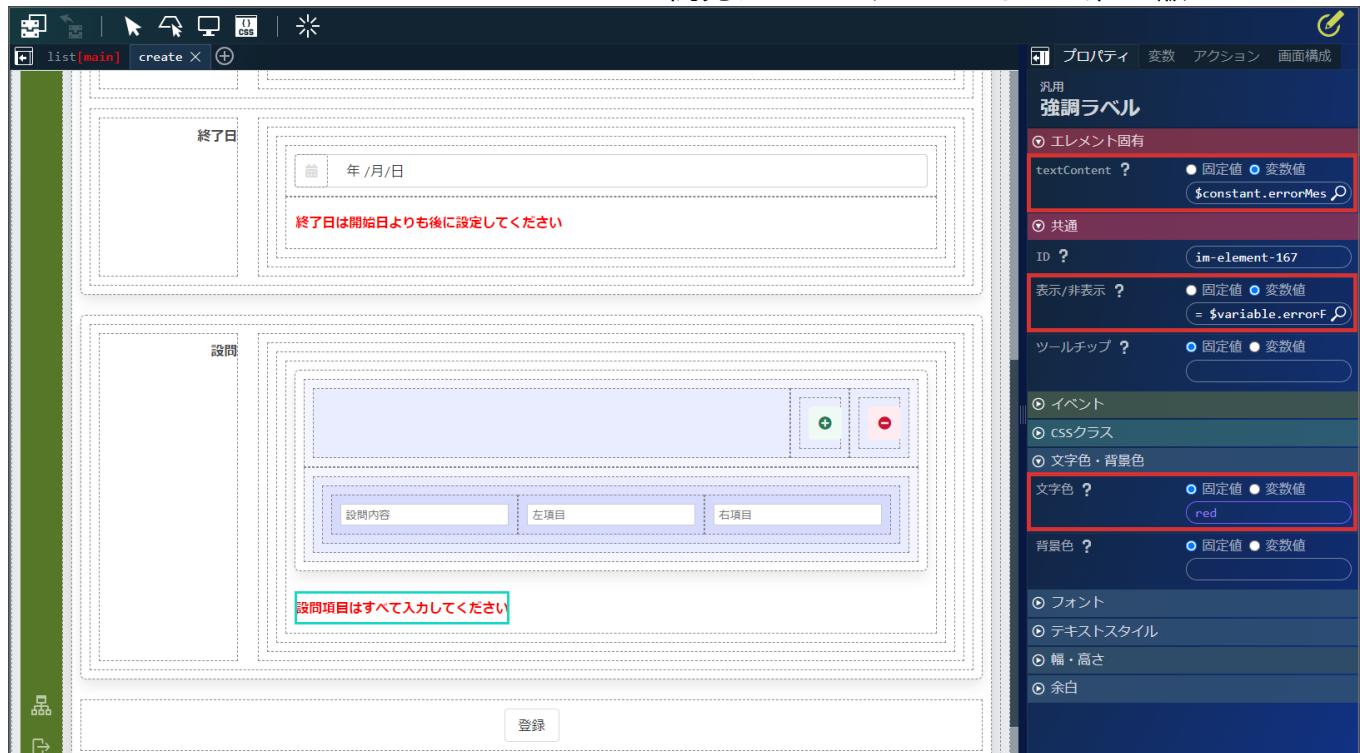
続けて、設問の入力チェックのエラーメッセージを配置します。

「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": {
  "enqueteid": "",
  "enquetteTitle": "",
  "enquetteOverview": "",
  "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。

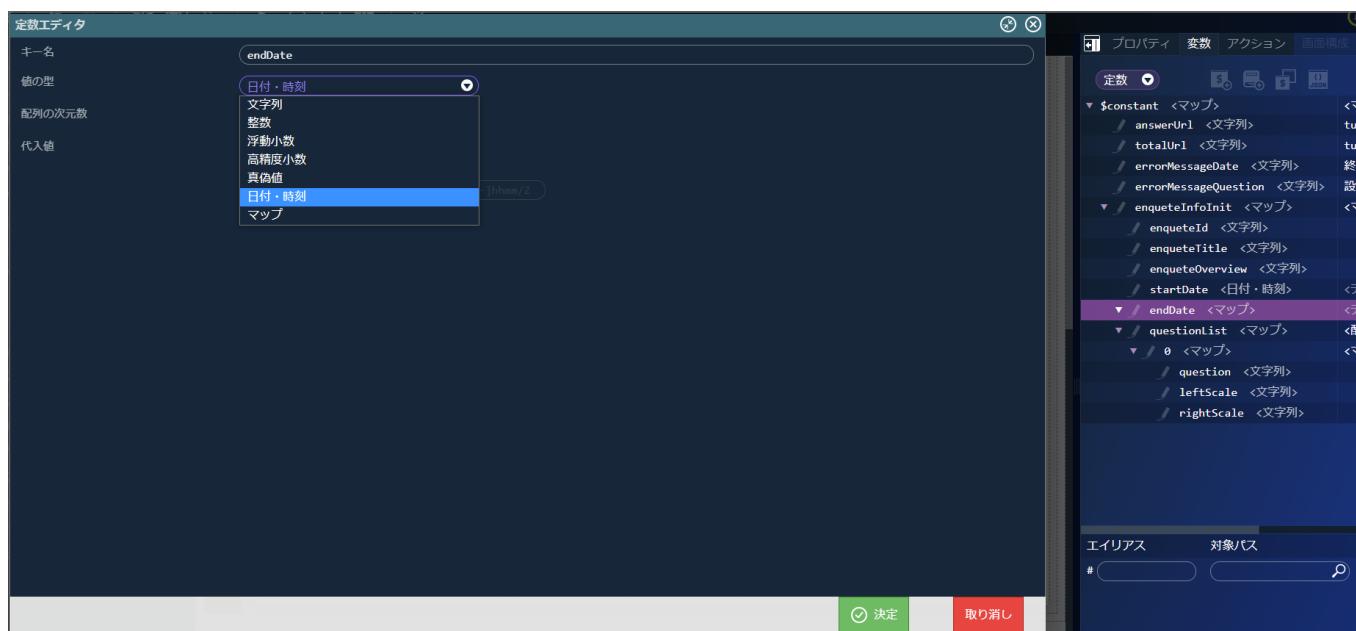
その際に、わかりやすくするために定数のキー名を「enqueteInfoInit」に変更しましょう

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 }
```

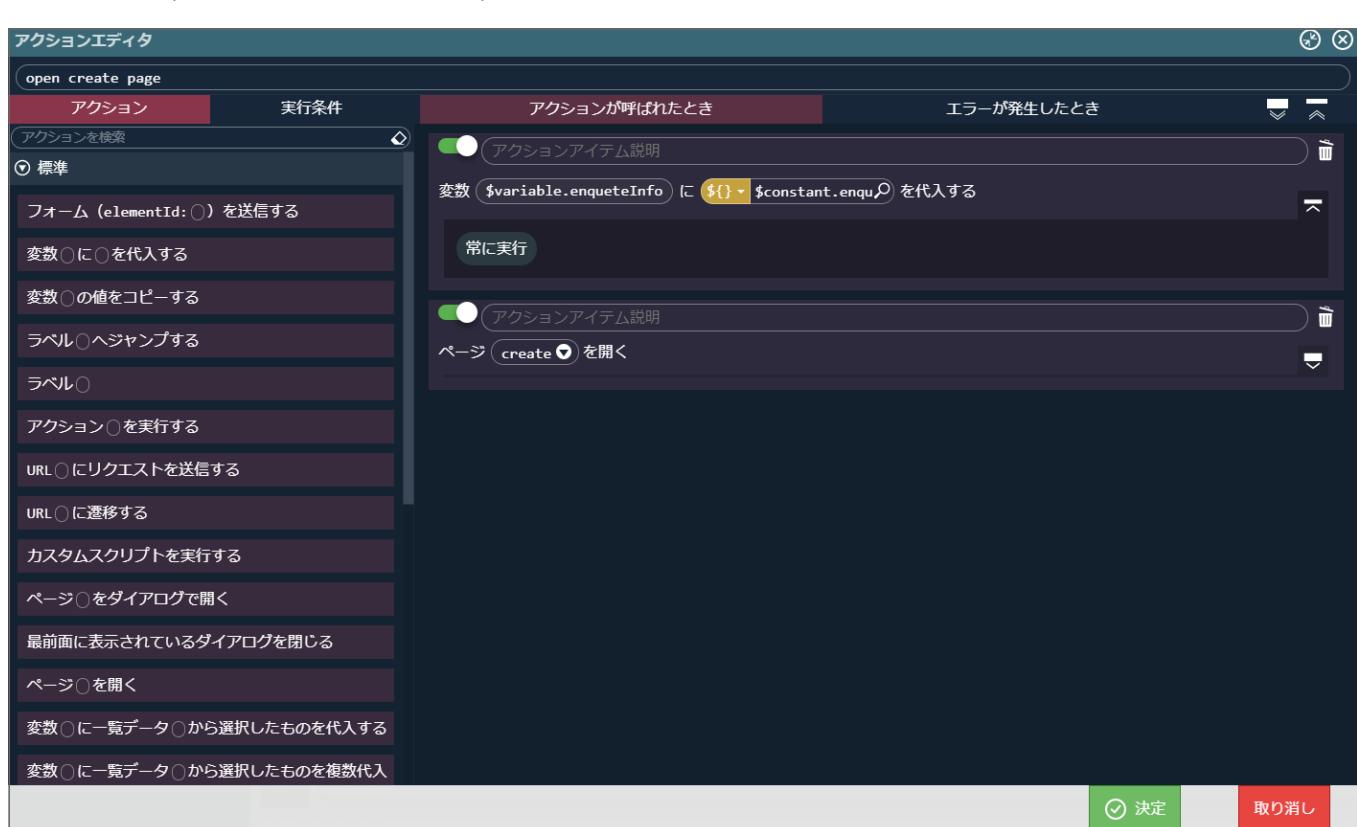
JSON入力の場合、初期値を「null」にした場合、値の型が「マップ」となるため、「\$constant > enqueteInfoInit > startDate」、
「\$constant > enqueteInfoInit > endDate」の値の型を「日付・時刻」に変更します。



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。

The screenshot shows a list of API URLs in a browser-like interface. One item in the list is highlighted with a red box, and a context menu is open over it. The menu items include '画面URL' (Page URL), 'APIのURL' (API URL), and 'クエリのURL' (Query URL). The '画面URL' option is specifically highlighted.

「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。

「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。

The screenshot shows the configuration of a new route in the intra-mart Accel Platform. The route is named 'tutorial_enquete_list'. It includes fields for Category (im_bouquet), Content ID (tutorial_enquete_list), Content Name ([チュートリアル] アンケート一覧_登録画面), Version (最新バージョンを利用する), Method (GET), URL (/mart/tutorial_enquete/list), and Access URI (im-bloommaker-content://contents/route/tutorial_enquete_list). The route name is also specified in the 'ルーティング名' and '備考' sections. A '登録' (Register) button is at the bottom.

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。

ToDoアプリケーションに追加機能を作成する

概要

本章では、「[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](#)」で作成した簡易的なToDo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。



コラム

オプションテンプレートについての詳細は、以下のドキュメントを参照してください。

[「intra-mart Accel Platform Accel Studio テンプレートカタログ」 - 「オプションテンプレート」](#)

作成の流れ

本チュートリアルは「[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](#)」で作成したToDoアプリケーションに「ファイルアップロード機能」を追加します。
「ファイルアップロード機能」はAccelStudioのオプションテンプレートである「ファイルアップロードテンプレート」を利用して追加します。

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. 「テンプレートからリソース追加」にて「ファイルアップロードテンプレート」のリソース追加
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](#)」資料の編集

追加機能の仕様

ToDoアプリケーションの登録画面に、任意のファイルを添付可能なファイルアップロード機能を追加します。
画面の仕様は以下の通りです。

ファイルアップロード機能

登録画面の入力項目に、添付するファイルを選択できるボタンを配置します。
添付したファイルの削除、および備考欄へのテキスト追加も可能です。

【チュートリアル】ToDoアプリケーション
登録画面

概要 *

詳細

カテゴリ

重要度

期限日 年 /月/日

ファイル名 ここをクリックしてファイルを選択

参考

登録



コラム

「ファイルアップロード」テンプレートについての詳細は、以下のドキュメントを参照してください。

本チュートリアルは以下のドキュメントの手順に則ってファイルアップロード機能を追加します。

[「intra-mart Accel Platform Accel Studio テンプレートカタログ」 - 「ファイルアップロード」](#)

テンプレートから機能を追加する

チュートリアルで作成したToDoアプリケーションにオプションテンプレートからファイルアップロードテンプレートを追加します。

オプションテンプレートは作成したアプリケーションに機能を追加できるテンプレートです。

このテンプレートを利用すればファイルアップロードに関する処理が作成されます。ToDoアプリケーションの処理や画面を修正して、ファイルを一つアップロードできるようにしましょう。

テンプレートの追加

AccelStudioでは、作成したアプリケーションに対して、別で作成した画面や処理の追加が行えます。

本チュートリアルでは、オプションテンプレートの「ファイルアップロード」テンプレートを利用します。

オプションテンプレートの選択

アプリケーション管理から「【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」右上の「リソースを追加」プルダウンから「テンプレートからリソース追加」を選択し、「業務テンプレート一覧画面」を表示します。

【チュートリアル】ToDoアプリケーション - 詳細

+ リソースを追加 ▾

テンプレートからリソース追加

画面 公開URL ロジック データ定義 ユーティリティ

検索オプション

画面コンテンツ (ID: im-bloommaker//contents/tutorial_todo_app-list) 【チュートリアル】ToDoアプリケーション - 詳細 「マスタメンテナンス」テンプレートから作成された一覧画面のコンテンツです。

画面コン텐ツ (ID: im-bloommaker//contents/tutorial_todo_app-crud) 【チュートリアル】ToDoアプリケーション 登録・編集・参照画面 「マスタメンテナンス」テンプレートから作成された登録・編集・参照画面のコンテンツです。

画面URL (ID: im-bloommaker//routes/tutorial_todo_app-register) 【チュートリアル】ToDoアプリケーション 登録画面 「マスタメンテナンス」テンプレートから作成された登録画面のルーティング定義です。

画面URL (ID: im-bloommaker//routes/tutorial_todo_app-refer) 【チュートリアル】ToDoアプリケーション 参照画面 「マスタメンテナンス」テンプレートから作成された参照画面のルーティング定義です。

画面URL (ID: im-bloommaker//routes/tutorial_todo_app-edit) 【チュートリアル】ToDoアプリケーション 編集画面 「マスタメンテナンス」テンプレートから作成された編集画面のルーティング定義です。

APIのURL (ID: im-logic//routes/accel-studio-app/tutorial_todo_app/POST) POST accel-studio-app/tutorial_todo_app

APIのURL (ID: im-logic//routes/accel-studio-app/tutorial_todo_app/GET) GET accel-studio-app/tutorial_todo_app

APIのURL (ID: im-logic//routes/accel-studio-app/tutorial_todo_app/POST) POST accel-studio-app/tutorial_todo_app/list

権限設定 リソース関連図 ログ ログイン ヘルプ

認可の変更

【ルーティング名: 【チュートリアル】ToDoアプリケーション 参照画面】 メソッド: GET 公開URL: accel-studio-app/tutorial_todo_app/refer 以下のアクセスが許可されました。認証済みユーザー

【ルーティング名: 【チュートリアル】ToDoアプリケーション 編集画面】 メソッド: GET 公開URL: accel-studio-app/tutorial_todo_app/edit 以下のアクセスが許可されました。認証済みユーザー

【ルーティング名: 【チュートリアル】ToDoアプリケーション 参照画面】 メソッド: GET 公開URL: accel-studio-app/tutorial_todo_app/list 以下のアクセスが許可されました。認証済みユーザー

【ルーティング名: POST accel-studio-app/tutorial_todo_app】 メソッド: POST 公開URL: logic/api/accel-studio-app/tutorial_todo_app 以下のアクセスが許可されました。

「業務テンプレート一覧」が表示されるので、「ファイルアップロード」テンプレートを選択します。

既存アプリケーションの更新 業務テンプレート一覧

社内FAQ管理 休暇申請ワークフロー 自己紹介 メール配信ワークフロー

次へ 次へ 次へ 次へ

オプションテンプレート

作成したアプリケーションに機能を追加します。このテンプレートからアプリケーションを作成することも可能です。
※ 基本的な機能を含まないため Accel Studio での開発経験がある方向けのテンプレートです。

ファイルアップロード CSVインポート CSVエクスポート ガントチャート

IM-BloomMaker のファイルアップロードエレメントを利用した、ファイルアップロード処理を作成します。このテンプレートでは画面資材は作成されません。

CSV形式のファイルをインポートする処理を作成します。このテンプレートでは画面資材は作成されません。

CSV形式でファイルをエクスポートする処理を作成します。このテンプレートでは画面資材は作成されません。

指定した開始・終了日をもとにガントチャートを表示するテンプレートです。データの取得部分を変更することで利用できます。※アプリケーション作成時点ではサンプルデータが表示されます。

次へ 次へ 次へ 次へ

「既存アプリケーションの更新」画面が表示されます。

「既存アプリケーションの更新」画面では、既存アプリケーションの基本情報が自動入力されます。なお、アプリケーションIDとアプリケーション名は変更できません。

The screenshot shows the 'File Upload - Resource Addition' screen. On the left is a vertical toolbar with icons for Home, Search, and Help. The main area has tabs for 'Application Information' (checked), 'Data Definition' (checked), and 'Other Information' (checked). The 'Application Information' tab contains fields for 'Application ID' (tutorial_todo_app), 'Application Name' (【Tutorial】ToDo Application), and a 'Description' text area. The 'Data Definition' tab contains a note about defining data and a section for table generation settings (auto-generate table) and database type (TENANT). A blue button at the bottom right says 'Existing Application Update'.

IDの決定に利用する文字列の設定

「IDの決定に利用する文字列」を設定します。自動的にアプリケーションIDと同じものが入力されていますが、任意のIDを設定してください。

「IDの決定に利用する文字列」は、各処理やIDなどに利用されます。

This screenshot is identical to the one above, but the input field for 'IDの決定に利用する文字列' (tutorial_todo_app) is highlighted with a red rectangle. The rest of the interface, including the sidebar, tabs, and other input fields, remains the same.

利用するデータベース種別とテーブルの名称の設定

次に、利用するデータベース種別とテーブルの名称を設定します。

「ファイルアップロード」テンプレートでは、新規にテーブルを作成する必要があります。任意のテーブル名を入力してください。アプリケーションの基本情報とテーブル情報を設定した後、「既存アプリケーションの更新」ボタンをクリックします。

The screenshot shows the 'Template Build Job - Details' screen in the Accel Platform. At the top, there's a header with icons for back, forward, and search, followed by the title 'テンプレートビルトジョブ - 詳細'. On the left is a vertical toolbar with icons for home, search, and others. The main area has two tabs: 'ビルトジョブID' (Build Job ID) and 'ログ' (Log). The 'ビルトジョブID' tab shows details: ID '8h9503st8gql9c4', Status 'COMPLETE', and Creation Date '2024/06/26 17:18:38'. The 'ログ' tab displays a log of build events:

| 出力日時 | メッセージ | 詳細 |
|---------------------|--|----|
| 2024/06/26 17:18:38 | テンプレートをビルドし、「【チュートリアル】ToDoアプリケーション」(アプリケーションID: tutorial_todo_app)にリソースを追加します。 | |
| 2024/06/26 17:18:38 | 生成するリソースのIDを自動設定します。 | |
| 2024/06/26 17:18:39 | 一時ディレクトリに IM-LogicDesigner インポート資材を作成しました。 | |
| 2024/06/26 17:18:39 | アプリケーションで利用するテーブルを新規作成しました。 | |
| 2024/06/26 17:18:39 | CREATE TABLE tutorial_todo_file_upload (file_id VARCHAR(100), file_key VARCHAR(128), business_key VARCHAR(1000), create_user_cd VARCHAR(100), create_date TIMESTAMP, record_user_cd VARCHAR(100), record_date TIMESTAMP, note VARCHAR(1000), PRIMARY KEY (file_id)) | |
| 2024/06/26 17:18:40 | 自動生成された IM-LogicDesigner 資材をインポートしました。 | |
| 2024/06/26 17:18:42 | アプリケーションにテンプレートから生成されたリソースを追加しました。 | |
| 2024/06/26 17:18:42 | 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。 | |

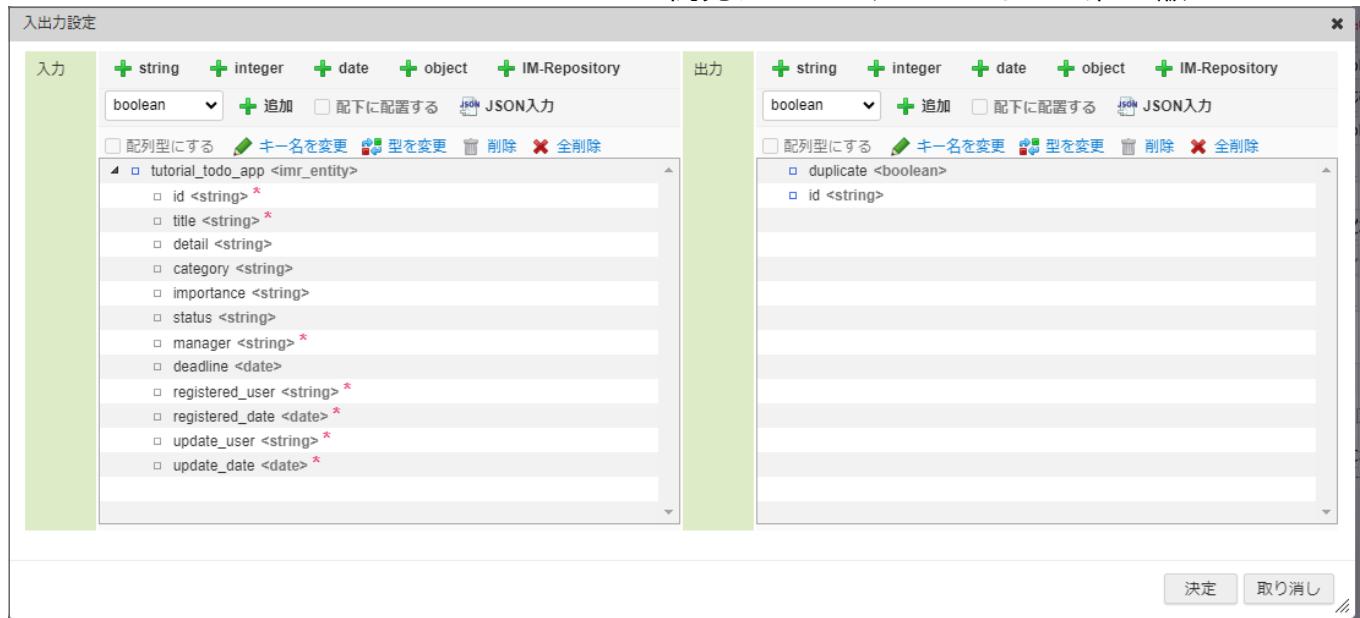
アプリケーションの更新が完了すると、ステータスが「COMPLETE」と表示されます。

登録処理の修正

「ファイルアップロード」テンプレートでは、変数のbusinessKeyという値に紐づく添付ファイルの登録、更新、削除が行えます。本チュートリアルでは、ToDoのIDをbusinessKeyに設定して、ToDoに紐づく添付ファイルの登録、更新、削除を行うようにします。「アプリケーション管理画面」から「【チュートリアル】ToDoアプリケーション 登録処理」を選択してください。「ロジックフロー定義編集」画面が表示されます。「【チュートリアル】ToDoアプリケーション 登録処理」では、エンティティデータの登録タスク(im_repositoryEntityDataInsert1)内でidentifier関数を利用して一意のIDを作成しています。変数機能を活用して、登録時に設定されたIDを画面でも利用できるようにしましょう。

入出力・変数の追加

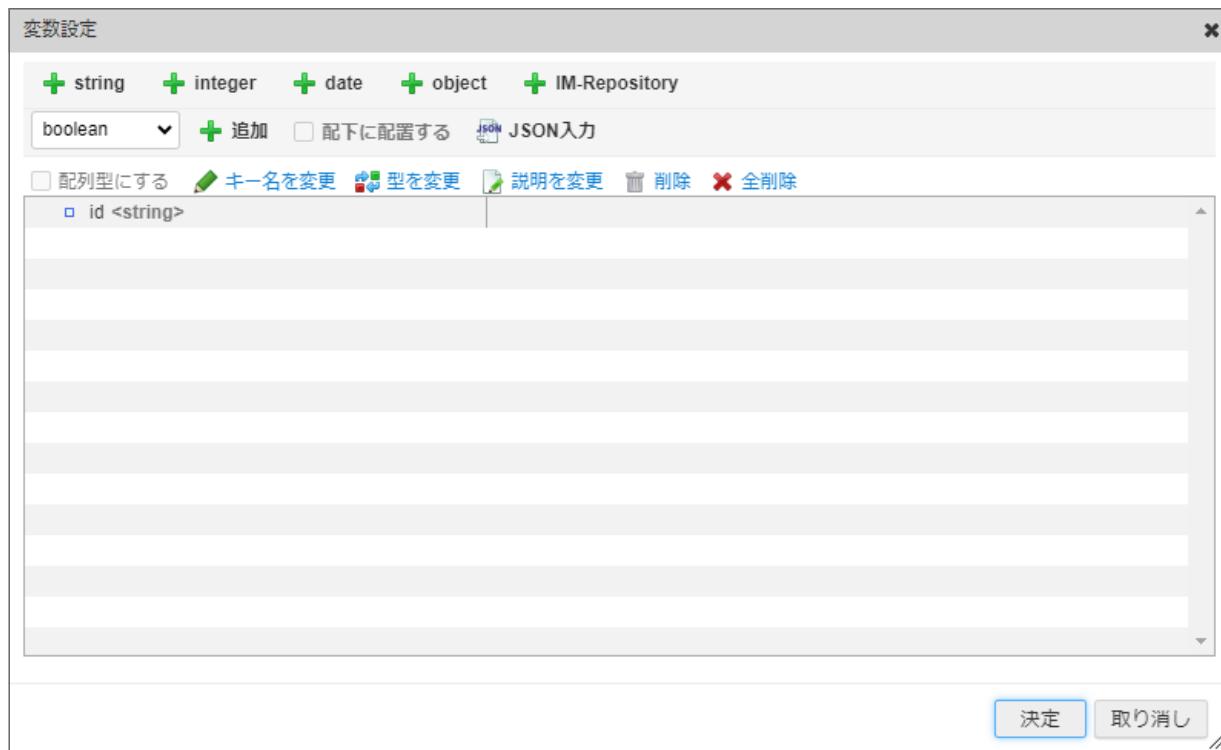
まずは、ロジックフローの返却値として利用できるように、入出力設定を行いましょう。画面上部の「入出力設定」を選択してください。「入出力設定」の「出力」側の「+string」を選択してください。追加された出力値のキー名を「id」としてください。「決定」ボタンで入出力設定を閉じてください。



続けて「変数設定」を選択してください。

「+string」を選択し、追加された変数のキーナムを「id」としてください。

「決定」ボタンで変数設定を閉じてください。



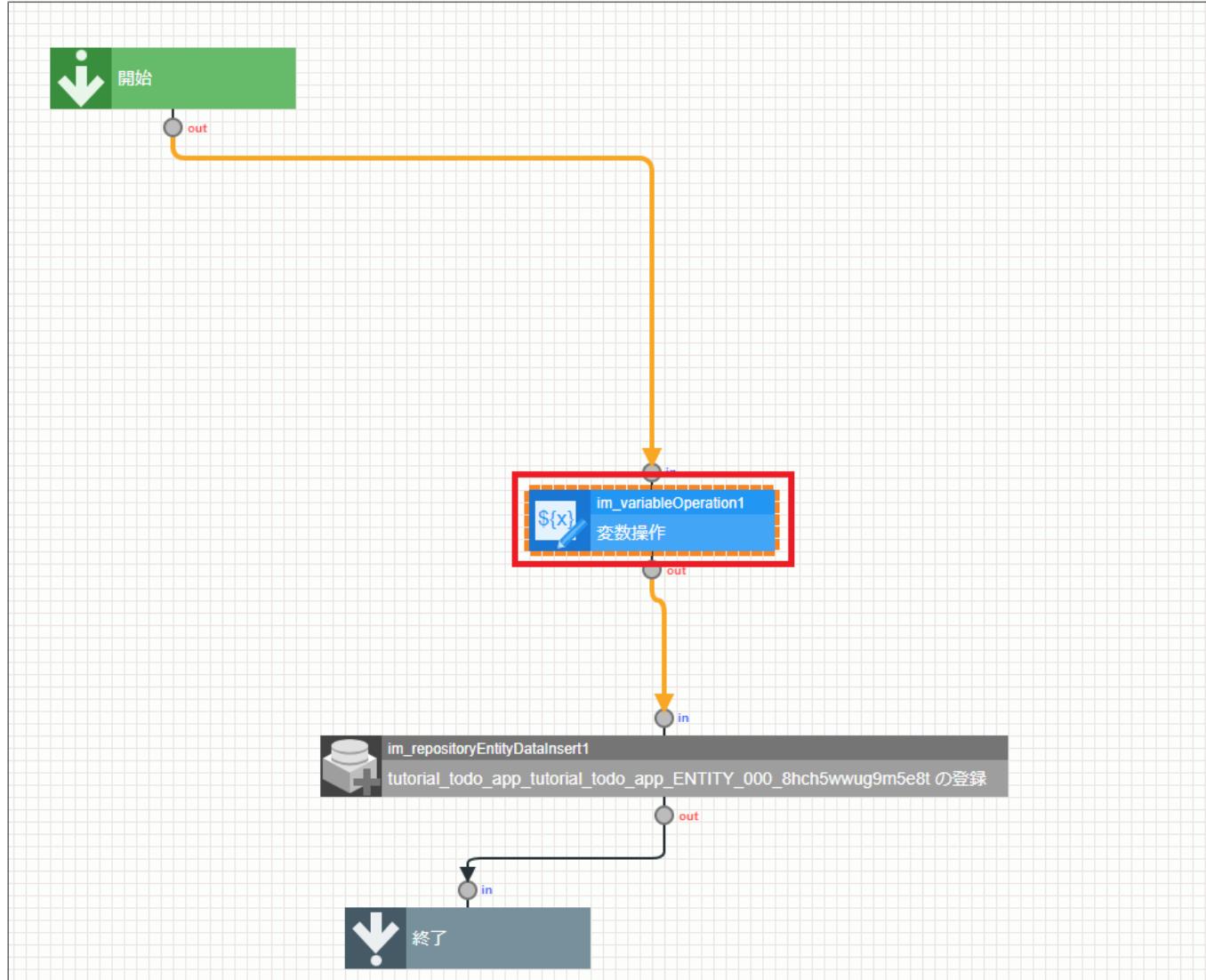
これで、入出力、変数の設定は完了です。ロジックフローの修正も行いましょう。

ロジックフローの修正

作成した変数「id」に一意の値を設定して、出力するようにフローを修正しましょう。

まずは、パレットから「基本」、「変数操作」を選択し、開始と登録タスク (im_repositoryEntityDataInsert1) の間に配置しましょう。





配置した変数操作をダブルクリックし、マッピング設定を表示してください。

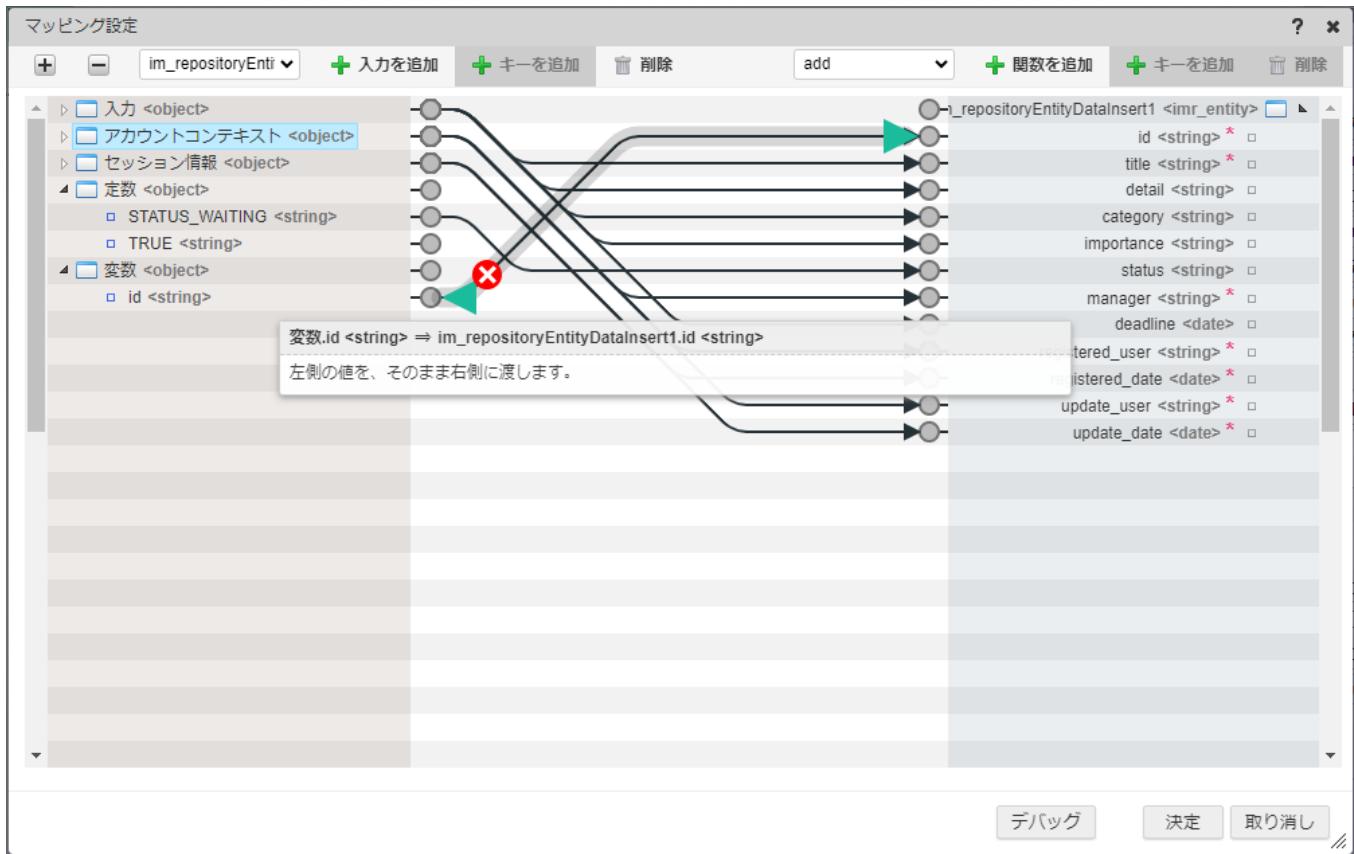
関数から「identifier」を追加し、変数欄の「id」に線を繋げて「決定」でマッピング設定を閉じてください。



続けて、登録タスクの修正を行いましょう。登録タスクをダブルクリックしてマッピング設定を表示してください。

「フロー定義」 - 「変数」を選択して、「+入力を追加」を行い、変数を入力値として利用できるようにします。

登録タスクのidentifierに紐づいているidを削除し、「変数」の「id」と「登録タスク」の「id」を線で繋げて「決定」でマッピング設定を閉じてください。



最後に、ロジックフローの出力値としてもidを設定しましょう。

「終了」をダブルクリックして、マッピング設定を表示してください。

「変数」の「id」と「出力」の「id」を線で繋げて「決定」でマッピング設定を閉じてください。



これでロジックフローの修正は完了です。「保存」でロジックフローを保存してください。

ファイルアップロード機能を、ToDoアプリケーションの画面に追加します。
「アプリケーション管理画面」から「【チュートリアル】登録・編集・参照画面」を選択してください。
「デザイン編集」画面が表示されます。

変数の作成

まずは、ファイルアップロードに必要な変数を追加しましょう。

「変数」タブから  をクリックして、\$variable > endPoint 配下に、以下の変数を追加してください。

```
"list": {
  "requestParameter": {
    "limit": 10,
    "offset": 1,
    "businessKey": ""
  },
  "responseData": {
    "records": [
      {
        "fileKey": "",
        "createUserCd": "",
        "createUserName": "",
        "createDate": null,
        "createDateStr": "",
        "recordUserCd": "",
        "recordUserName": "",
        "recordDate": null,
        "recordDateStr": "",
        "note": ""
      }
    ],
    "totalCount": 0
  }
},
"deleteAll": {
  "requestParameter": {
    "businessKey": ""
  },
  "responseData": {
    "error": "",
    "errorMessage": ""
  }
},
"crud": {
  "requestParameter": {
    "records": [
      {
        "fileKey": "",
        "note": ""
      }
    ],
    "businessKey": ""
  },
  "responseData": {
    "error": "",
    "errorMessage": ""
  }
}
```

追加後に、「createDate」と「recordDate」の  をクリックして、「値の型」を「マップ」から「日付・時刻」に変更してください。

変数

\$variable <マップ>

- state <マップ>
- endPoint <マップ>
- refer <マップ>
- edit <マップ>
- register <マップ>
- delete <マップ>
- list <マップ>
 - requestParameter <マップ>
 - responseData <マップ>
 - records <マップ>
 - 0 <マップ>
 - fileKey <文字列>
 - createUserCd <文字列>
 - createUserName <文字列>
 - createDate <日付・時刻>
 - createDateStr <文字列>
 - recordUserCd <文字列>
 - recordUserName <文字列>
 - recordDate <日付・時刻>
 - recordDateStr <文字列>
 - note <文字列>
 - totalCount <整数>
 - deleteAll <マップ>
 - crud <マップ>
- businessKey <マップ>
 - id <文字列>

エイリアス 対象パス 削除

次に、もう一度 をクリックして、\$variable > state 配下に以下の変数を追加してください。

```
"uploadFileInfoList": [
  {
    "fileKey": "",
    "note": ""
  }
]
```

```

1  {
2      "entity": {
3          "tutorial_todo_app": {
4              "id": null,
5              "title": null,
6              "detail": null,
7              "category": null,
8              "importance": null,
9              "status": null,
10             "manager": null,
11             "deadline": null,
12             "registered_user": null,
13             "registered_date": null,
14             "update_user": null,
15             "update_date": null
16         }
17     }
18     "key": {
19         "id": null
20     },
21     "uploadFileInfoList": [
22         {
23             "fileKey": "",
24             "note": ""
25         }
26     ],
27     "isChanged": false,
28     "isLocked": false,
29     "confirmResult": false
30 }

```

最後に、修正した登録タスクからIDを受け取れるように変数を修正しましょう。

\$variable > endPoint > register > responseData 配下に文字列の「id」を追加してください。

| | |
|----------------------------|---------|
| ▼ ↴ endPoint <マップ> | <マップ> |
| ▶ ↴ refer <マップ> | <マップ> |
| ▶ ↴ edit <マップ> | <マップ> |
| ▼ ↴ register <マップ> | <マップ> |
| ▼ ↴ requestParameter <マップ> | <マップ> |
| ▶ ↴ entity <マップ> | <マップ> |
| ▼ ↴ responseData <マップ> | <マップ> |
| ↗ ↴ duplicate <真偽値> | <データなし> |
| ↗ ↴ id <文字列> | |

新規アクションの追加

変数を追加したら、次はアクションを追加します。

「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開きます。

Save Upload File アクションの追加

最初に、Save Upload Fileアクションを作成します。このアクションは、アップロードしたファイルをデータベースに保存するアクションです。

アクション名には「Save Upload File」と入力してください。

「標準」から「変数〇に〇を代入する」アクションアイテムを配置して、以下のように設定します。

| 項目 | 指定する値 | 備考 |
|------|---|------------------------|
| 変数値1 | \$variable.endPoint.crud.requestParameter.records | |
| 変数値2 | \$variable.state.uploadFileInfoList | プルダウンから「変数値」を選択してください。 |

「標準」から「変数〇に〇を代入する」アクションアイテムを、先ほど配置したアクションの下に配置して、以下のように設定します。

| 項目 | 指定する値 | 備考 |
|------|---|------------------------|
| 変数値1 | \$variable.endPoint.crud.requestParameter.businessKey | |
| 変数値2 | \$variable.state.entity.tutorial_todo_app.id | プルダウンから「変数値」を選択してください。 |

「IM-LogicDesigner」から「IM-LogicDesigner フロールーティング〇にリクエストを送信する」アクションアイテムを配置して、以下のように設定してください。

| 項目 | 指定する値 | 備考 |
|-------------|---|------------------------------|
| ルーティング | accel-studio-app/tutorial_todo_app/upload-files | 「ファイル情報登録・更新・削除処理」で検索してください。 |
| リクエストパラメータ | \$variable.endPoint.crud.requestParameter | |
| セキュアトークンを送信 | チェックを入れる | |

アクションアイテムの設定後、「決定」をクリックしてアクションを保存します。

Load Upload File Table アクションの追加

次に、Load Upload File Table アクションを作成します。このアクションは、アップロードしたファイルの一覧テーブルを読み込み、画面に反映させるアクションです。

 をクリックしてアクションエディタを開いてください。

アクション名を「Load Upload File Table」とします。

「標準」 - 「変数〇に〇を代入する」アクションアイテムを配置して、以下のように設定します。

| 項目 | 指定する値 | 備考 |
|------|---|------------------------|
| 変数値1 | \$variable.endPoint.list.requestParameter.businessKey | |
| 変数値2 | \$variable.state.entity.tutorial_todo_app.id | プルダウンから「変数値」を選択してください。 |

「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」アクションアイテムを先ほどのアクションの次に配置して、以下のように設定してください。

| 項目 | 指定する値 | 備考 |
|-------------|--|--------------------------|
| ルーティング | accel-studio-app/tutorial_todo_app/upload-files/list | 「ファイル情報一覧取得処理」で検索してください。 |
| リクエストパラメータ | \$variable.endPoint.list.requestParameter | |
| レスポンスデータ | \$variable.endPoint.list.responseData | |
| セキュアトークンを送信 | チェックを入れない | |

一番下に「標準」 - 「変数〇に〇を代入する」アクションアイテムを配置して、以下のように設定します。

| 項目 | 指定する値 | 備考 |
|------|---|------------------------|
| 変数値1 | \$variable.state.uploadFileInfoList | |
| 変数値2 | \$variable.endPoint.list.responseData.records | プルダウンから「変数値」を選択してください。 |

アクションアイテムの設定後、「決定」をクリックしてアクションを保存してください。

Delete All Upload File アクションの追加

最後に、Delete All Upload File アクションを作成します。このアクションは、アップロードした紐づけられているファイルをすべて削除する際に使用します。

 をクリックしてアクションエディタを開いてください。

アクション名を「Delete All Upload File」とします。

「標準」 - 「変数〇に〇を代入する」 アクションアイテムを配置して、以下のように設定します。

| 項目 | 指定する値 | 備考 |
|------|--|------------------------|
| 変数値1 | \$variable.endPoint.deleteAll.requestParameter.businessKey | |
| 変数値2 | \$variable.state.entity.tutorial_todo_app.id | プルダウンから「変数値」を選択してください。 |

「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」 アクションアイテムを配置して、以下のように設定してください。

| 項目 | 指定する値 | 備考 |
|-------------|---|--------------------------|
| ルーティング | accel-studio-app/tutorial_todo_app/upload-files/all | 「ファイル情報一括削除処理」で検索してください。 |
| リクエストパラメータ | \$variable.endPoint.deleteAll.requestParameter | |
| レスポンスデータ | \$variable.endPoint.deleteAll.responseData | |
| セキュアトークンを送信 | チェックを入れる | |

アクションアイテムの設定後、「決定」をクリックしてアクションを保存します。

既存アクションの編集

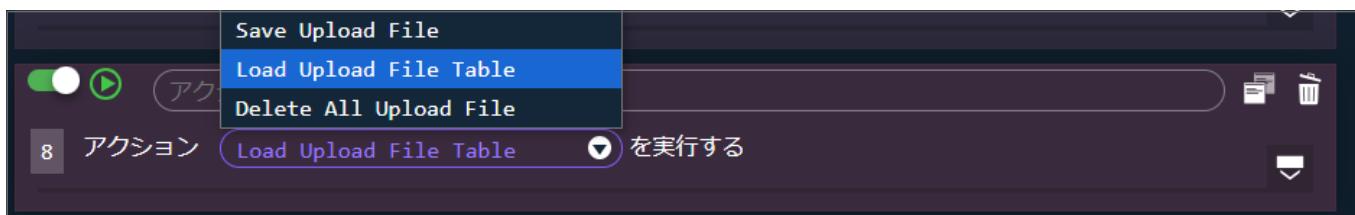
「初期表示時」 アクション

ここからは、ToDoアプリケーションの既存アクションの編集を行います。

「初期表示時」 アクションの  をクリックしてアクションエディタを開いてください。

「標準」から「アクション〇を実行する」 アクションアイテムを選択し、一番下に配置してください。

プルダウンから「Load Upload File Table」を選択し、「決定」をクリックしてアクションを保存してください。



「登録ボタンが押されたとき」 アクション

「登録ボタンが押されたとき」 アクションの  をクリックしてアクションエディタを開いてください。

「IM-LogicDesigner フローリーティングにリクエストを送信する」の後に、「標準」 - 「変数○に○を代入する」アクションアイテムを配置して、以下のように設定します。

| 項目 | 指定する値 | 備考 |
|------|--|------------------------|
| 変数値1 | \$variable.state.entity.tutorial_todo_app.id | |
| 変数値2 | \$variable.endPoint.register.responseData.id | プルダウンから「変数値」を選択してください。 |

これで、ToDoの登録時に登録したIDが\$variable.state.entity.tutorial_todo_app.idに設定されました。

「標準」から「アクション○を実行する」アクションアイテムを選択し、「アクション一覧画面へ遷移を実行する」の前に配置します。プルダウンから「Save Upload File」を選択し、「決定」をクリックしてアクションを保存してください。



「更新ボタンが押されたとき」アクション

「更新ボタンが押されたとき」アクションの をクリックしてアクションエディタを開いてください。

「標準」から「アクション○を実行する」アクションアイテムを選択し、「URL \$constant.screenUrl.list に遷移する」の前に配置します。

プルダウンから「Save Upload File」を選択し、「決定」をクリックしてアクションを保存してください。

「削除ボタンが押されたとき」アクション

「削除ボタンが押されたとき」アクションの をクリックしてアクションエディタを開いてください。

「標準」から「アクション○を実行する」アクションアイテムを選択し、「アクション一覧画面へ遷移を実行する」の前に配置します。

プルダウンから「Delete All Upload File」を選択し、「決定」をクリックしてアクションを保存してください。

ファイルアップロードエレメントの追加

ファイルアップロードエレメントを配置するコンテナページを作成し、ToDoアプリケーションの登録画面にファイルを添付できるエリアを追加します。

コンテナページの追加

をクリックして新規コンテナページを追加し、コンテナページのエレメント固有の「name」に「File Upload Area」と入力してください。

「繰り返し（Bulma）」から「テーブルコンテナ（繰り返し）」エレメントを配置します。

「テーブル（繰り返し）」を選択し、エレメント固有 - プロパティの「columnCount」を「2」に、「list」の変数値を「\$variable.state.uploadFileInfoList」と設定してください。

「汎用」 - 「ラベル」エレメントを選択し、画像のように「テーブルヘッダセル」内に配置します。

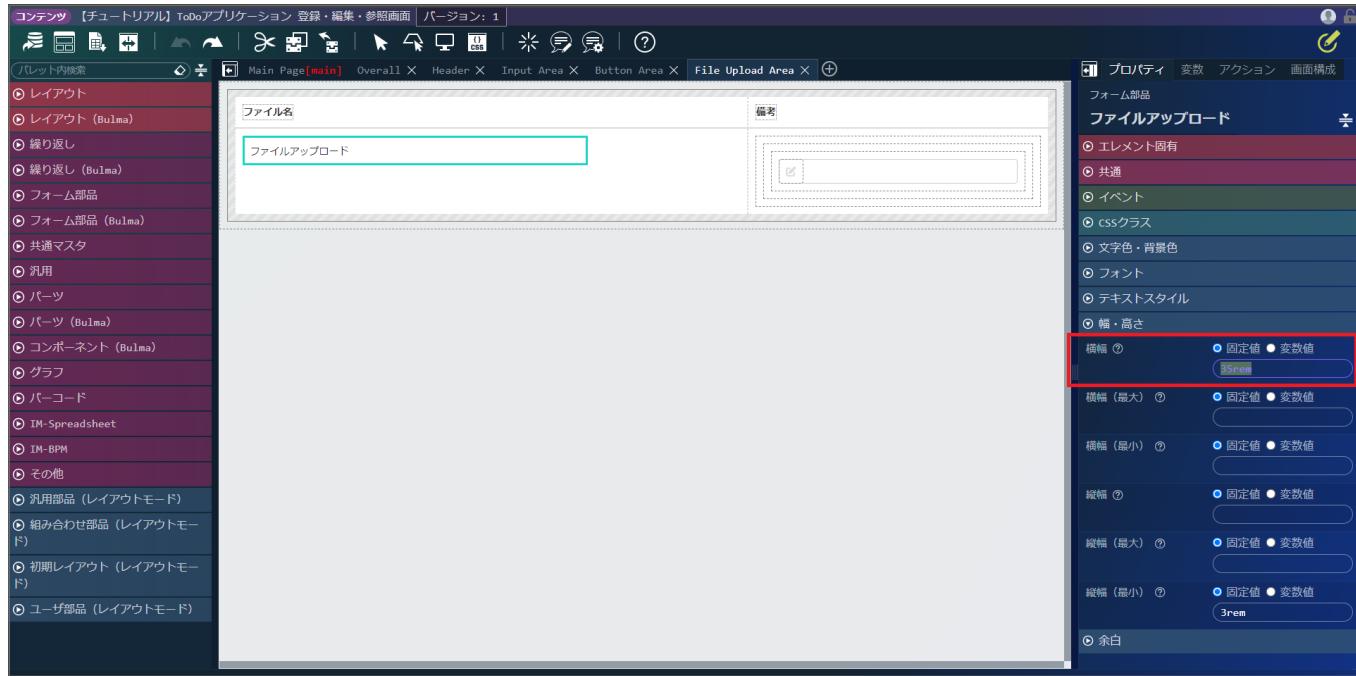
「ラベル」のエレメント固有の「textContent」には「固定値」でそれぞれ「ファイル名」と「備考」を入力してください。

次に、「フォーム部品」から「ファイルアップロード」エレメントを選択し、画像のように「テーブルセル」内に配置してください。

「ファイルアップロード」のエレメント固有のプロパティは以下のように設定してください。

| 項目 | 指定する値 | 備考 |
|----------|--|---------------|
| value | \$variable.state.uploadFileInfoList[\$index].fileKey | |
| readonly | = \$input.__mode__ == \$constant.mode.refer | 変数値を選択してください。 |

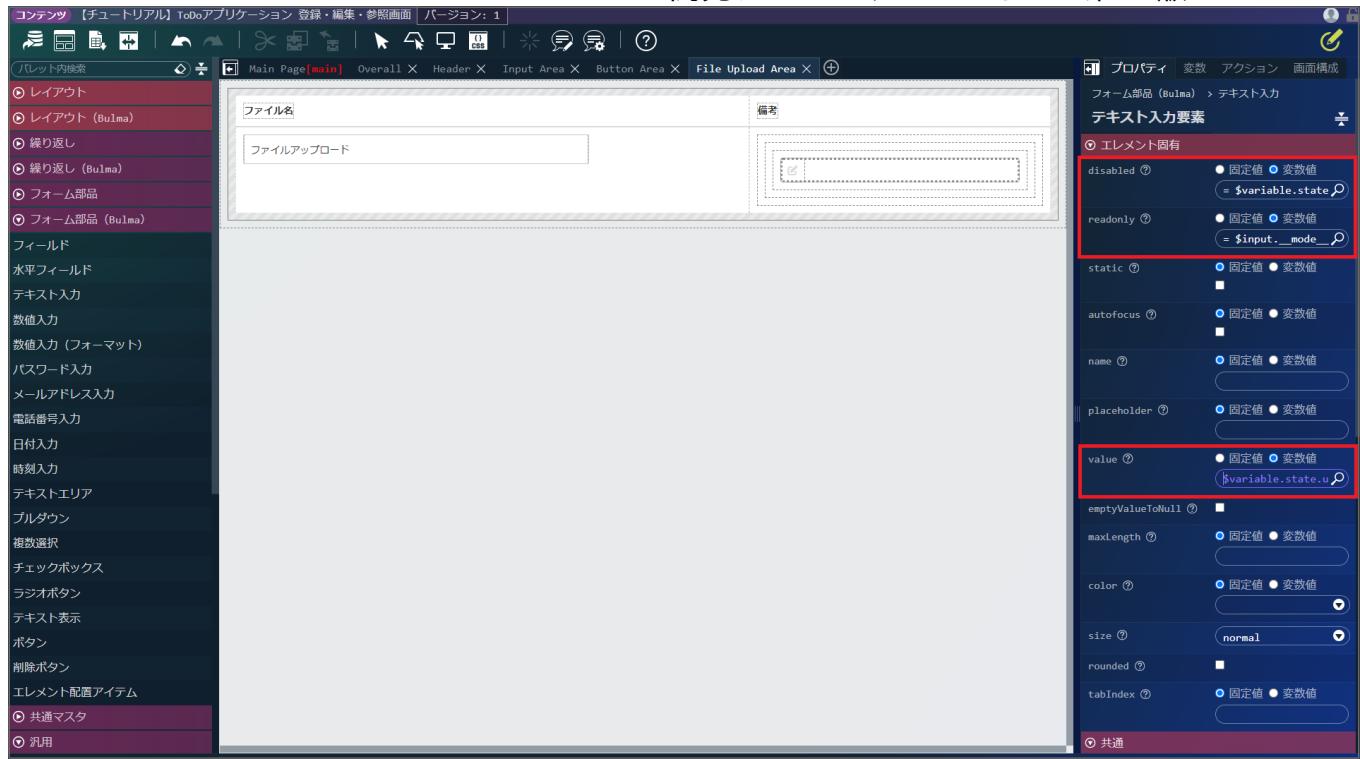
「ファイルアップロード」 - 「幅・高さ」 - 「横幅」に設定されている「3rem」を削除します。



「フィールド」エレメントを配置した後、「フォーム部品（Bulma）」から「テキスト入力」エレメントを選択し、「フィールド」内に配置してください。

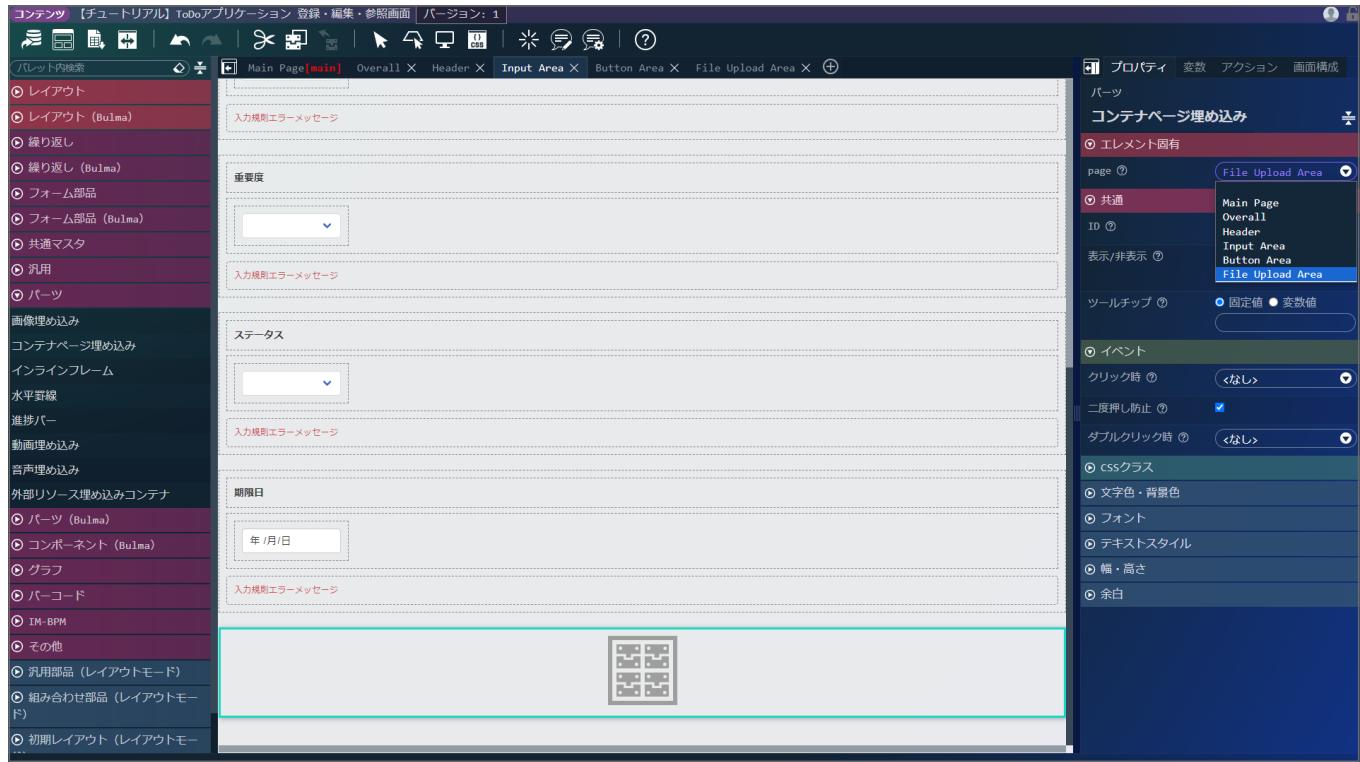
「テキスト入力要素」のエレメント固有のプロパティは以下のように設定してください。

| 項目 | 指定する値 | 備考 |
|----------|--|---------------|
| disabled | = \$variable.state.uploadFileInfoList[\$index].fileKey == null | 変数値を選択してください。 |
| readonly | = \$input.__mode__ == \$constant.mode.refer | 変数値を選択してください。 |
| value | \$variable.state.uploadFileInfoList[\$index].note | 変数値を選択してください。 |



追加したコンテナページの配置

「パート」から「コンテナページ埋め込み」エレメントをコンテナ「Input Area」の「期限日」の下に配置します。



「コンテナページ埋め込み」エレメントの「page」プロパティには、作成した「File Upload Area」を指定してください。

動作確認

ここまで作成したら、プレビュー画面から動作確認をしてみましょう。

登録画面でファイルアップロードが行え、更新画面ではアップロードしたファイルの確認や削除が行えます。

The screenshot shows a registration form for a Todo application. The form includes fields for a summary, detailed description, category selection, priority level, due date, file upload, and a note section. A red box highlights the file upload field, which contains a placeholder text: '+ ここをクリックしてファイルを選択' (Click here to select a file). A blue '登録' (Register) button is located at the bottom left of the form area.

i コラム

本チュートリアルでは、一つのファイルのアップロードに関する手順を説明しました。複数のファイルをアップロードについては、以下のドキュメントも参考にしてください。

[「intra-mart Accel Platform Accel Studio テンプレートカタログ」 - 「ファイルアップロード」](#)

ユーザのタイムゾーンや日付と時刻の形式に対応する

実装ガイド

本章では、ユーザが設定したタイムゾーンや日付と時刻の形式に従って、画面表示したり、画面から入力できるように実装するための対応方法を紹介します。

項目

- 前提
- 日付
 - 日付を保存する
 - 保存されている日付を画面に表示する
 - 保存されている日付一覧を画面に一覧表示する
 - 保存されている日付一覧をリッチテーブルに一覧表示する
- 日時
 - 日時を保存する
 - 保存されている日時を画面に表示する
 - 保存されている日時一覧を画面に一覧表示する
 - 保存されている日時一覧をリッチテーブルに一覧表示する
- 個別の機能要件
 - 日時を保存する
 - 保存されている日時を画面に表示する
 - 保存されている日時一覧を画面に一覧表示する
 - 保存されている日時一覧をリッチテーブルに一覧表示する

本章で紹介する対応方法は以下の考え方に基づいています。

- 「intra-mart Accel Platform タイムゾーン仕様書」 - 「動作仕様を実現するための考え方」

前提

紹介する対応方法は、以下のテーブルが作成されていることを前提としています。

- テーブル定義

```
CREATE TABLE example_table (
    user_cd    VARCHAR(100) NOT NULL,
    update_date TIMESTAMP NOT NULL,
    PRIMARY KEY (user_cd)
);
```

また、入力値や出力結果は以下の設定を前提として記載します。

- 各種設定値

| 設定項目 | 設定値 |
|-------------------------|---------------------|
| アカウントタイムゾーン（ユーザのタイムゾーン） | (GMT+09:00) 日本 / 東京 |
| システムタイムゾーン | (GMT+00:00) UTC |
| 日付と時刻の形式 | 英語形式 |
| | 日付（標準表示） |
| | 日付（簡易表示） |
| | 日付（入力） |

設定項目

設定値

| | |
|---------------|-----------|
| 時刻（標準表示） | h:mm a |
| 時刻（タイムスタンプ表示） | h:mm:ss a |
| 時刻（入力） | HH:mm |



コラム

IM-BloomMaker の一部のエレメントを利用した日付と時刻の表示形式は、ブラウザの言語設定によって変わります。本章の画面表示例は、ブラウザの言語設定が英語の場合の表示例です。



コラム

本章で利用している IM-LogicDesigner の以下のマッピング関数は intra-mart Accel Platform 2024 Autumn(Jasmine) 以降で利用可能です。

- `convertInSystemTimeZone`
- `convertInAccountTimeZone`
- `formatWithAccountTimeZone`

日付

以下のように日付を相対的に扱うための流れを説明します。

- 「intra-mart Accel Platform タイムゾーン仕様書」 - 「日付は相対的」

日付を保存する

IM-BloomMaker で日付を編集する画面を作成する場合、日付入力エレメントを利用します。

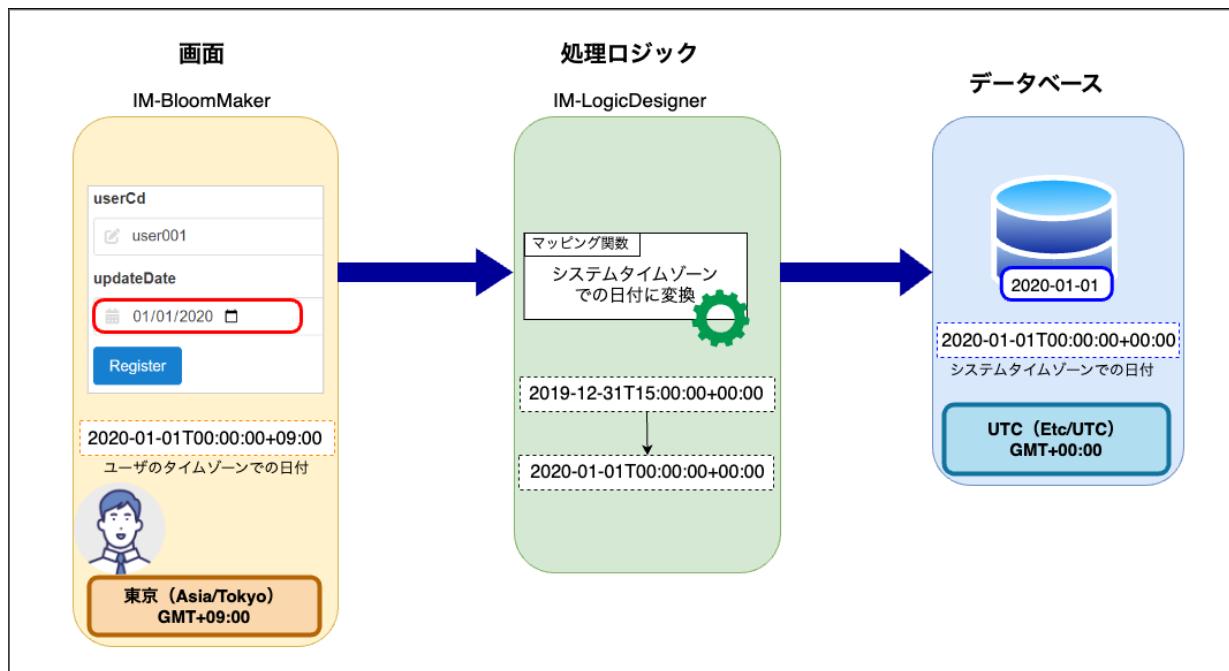
日付入力エレメントには、日付・時刻型の変数を設定します。

IM-BloomMaker で選択した日付はユーザーのタイムゾーンでの日付です。ユーザーのタイムゾーンでの日付を IM-LogicDesigner に送信します。

IM-LogicDesigner で日付は `date` 型で扱います。

日付を相対的に扱うためには、ユーザーのタイムゾーンでの日付をシステムタイムゾーンでの日付に変換する必要があります。

日付をデータベースに保存する際は、システムタイムゾーンでの日付をデータベースに保存します。



IM-BloomMaker から日付の送信する

日付入力エレメントを利用して日付を扱うために変数を設定します。

日付である「`updateDate`」のデータ型を「日付・時刻」に設定します。

変数 \$+ \$ JSON

\$variable <マップ> <マップ>

- requestData <マップ> <データなし>
- userCd <文字列> <データなし>
- updateDate <日付・時刻> <データなし>

日付を扱うために日付入力エレメントを設定します。

電話番号入力
日付入力
時刻入力
テキストエリア
プルダウン
複数選択
チェックボックス
ラジオボタン

日付の入力フォームを表示するエレメントです。
「フィールド」エレメントの配下に配置します。
このエレメントは Internet Explorer では動作しません。
また、ブラウザごとに動作が異なります。

2020/01/01

#Enter Date #input #カレンダー

日付入力エレメントの value プロパティには「日付・時刻」型である「updateDate」を設定します。

日付入力要素

⌚ エレメント固有

disabled ② 固定値 変数値

readonly ② 固定値 変数値

static ② 固定値 変数値

autofocus ② 固定値 変数値

name ② 固定値 変数値

value ② \$variable.requestData.updateDate

アクション「IM-LogicDesigner フロールーティング○にリクエストを送信する」のリクエストパラメータとして日付を含んだマップ型の変数を設定します。

IM-LogicDesigner フロールーティング (tutorial/sample/bm_ld_datetime/date) にリクエストを送信する

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

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

作成したコンテンツにおける日付の流れ

日付入力エレメントで以下の日付を入力されたとします。

2012/09/19

この場合、IM-LogicDesigner に送信される日付は以下のようにユーザのタイムゾーンでの日付として送信されます。

2012-09-19T00:00:00+09:00

IM-LogicDesigner での日付の受信とデータベースへの保存

ロジックフローの「入出力設定」で IM-BloomMaker から送信される情報に応じて入力を設定します。

日付を扱う「updateDate」のデータ型「date」に設定します。

入出力設定

入力

boolean + 追加 □ 配下に配置する JSON入力

□ 配列型にする キー名を変更 型を変更 削除 全削除

- userCd <string>
- updateDate <date>

出力

boolean + 追加 □ 配下に配置する JSON入力

□ 配列型にする キー名を変更 型を変更 削除 全削除

決定 取消し

入力値の情報をデータベースに保存するSQLユーザ定義タスクを用意します。

SQLユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|-------|--|
| クエリ種別 | INSERT |
| 入力値 | userCd <string> updateDate <sqltimestamp> |
| 返却値 | なし |

設定項目

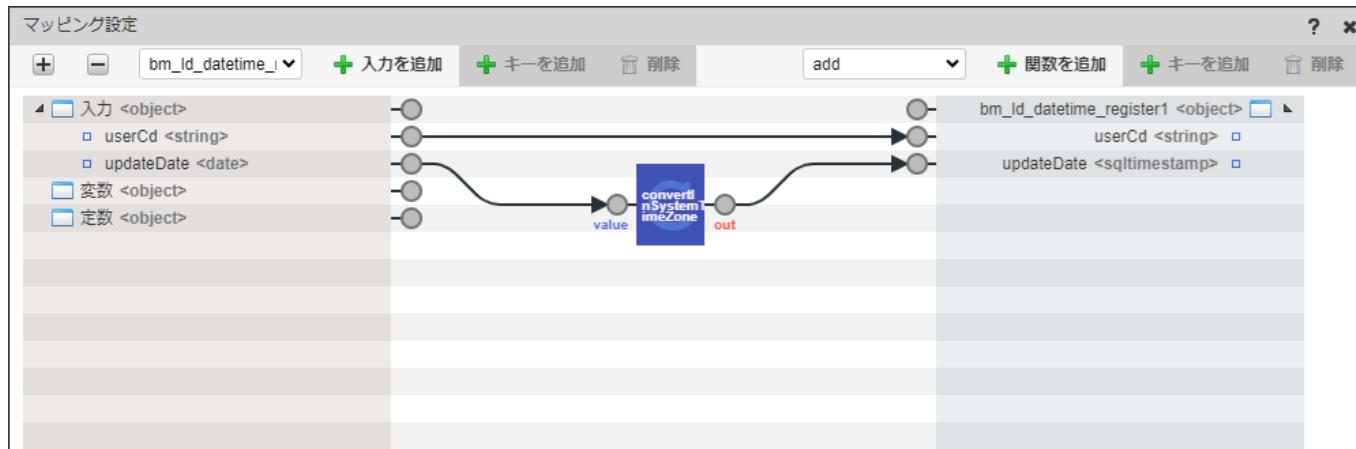
値

クエリ

```
INSERT INTO example_table (user_cd, update_date)
VALUES (/*userCd*/null, /*updateDate*/null)
```

用意したSQLユーザタスクのマッピング設定で日付をシステムタイムゾーンでの日付に変換します。

マッピング設定で日付をシステムタイムゾーンでの日付に変換するためにマッピング関数「convertInSystemTimeZone」を利用します。変換後の日付をSQLユーザ定義タスクの入力値として設定します。



作成したロジックフローにおける日付の流れ

IM-BloomMaker から以下の日付が送信されたとします。

```
2012-09-19T00:00:00+09:00
```

ロジックフロー内ではオフセット情報を持たないデータ型「date」で日付を扱っているため、送信された日付は以下の値として受け付けます。

```
2012-09-18T15:00:00+00:00
```

日付を相対的に扱うために、SQLユーザ定義タスクのマッピング設定で送信された上記の日付をマッピング関数「convertInSystemTimeZone」を利用してシステムタイムゾーンでの日付に変換しています。

日付は以下に変換されます。

```
2012-09-19T00:00:00+00:00
```

SQLユーザ定義タスクの入力値として変換後の日付が設定されているため、データベースにはシステムタイムゾーンでの日付が保存されます。

保存されている日付を画面に表示する

IM-LogicDesigner で日付は date 型で扱います。

IM-BloomMaker で日付を参照する際に日付を相対的に扱うためには、システムタイムゾーンでデータベースに保存されている日付をユーザのタイムゾーンでの日付に変換する必要があります。

データベースから取得した日付をユーザのタイムゾーンでの日付に変換し、ロジックフローの出力値として設定します。

IM-BloomMaker では日付入力エレメントを利用して日付を表示します。

日付入力エレメントには、日付・時刻型の変数を設定します。

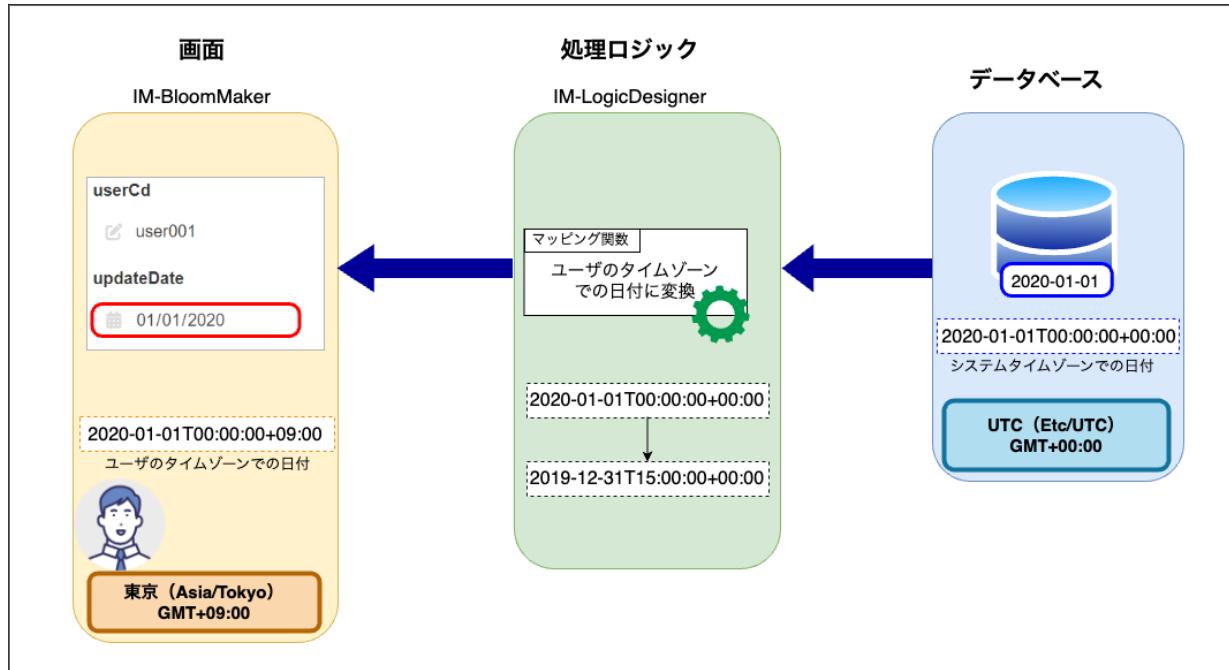
IM-LogicDesigner から取得した情報を「日付・時刻」型の変数に設定することで日付を画面に表示します。

i コラム

ユーザのタイムゾーンに依存せず、どの地域でも同一の日付を表示したい場合（日時と同様に絶対的として取り扱いたい場合）は、日付を文字列として扱うことを推奨します。

日付入力エレメントには、文字列型の変数を設定します。

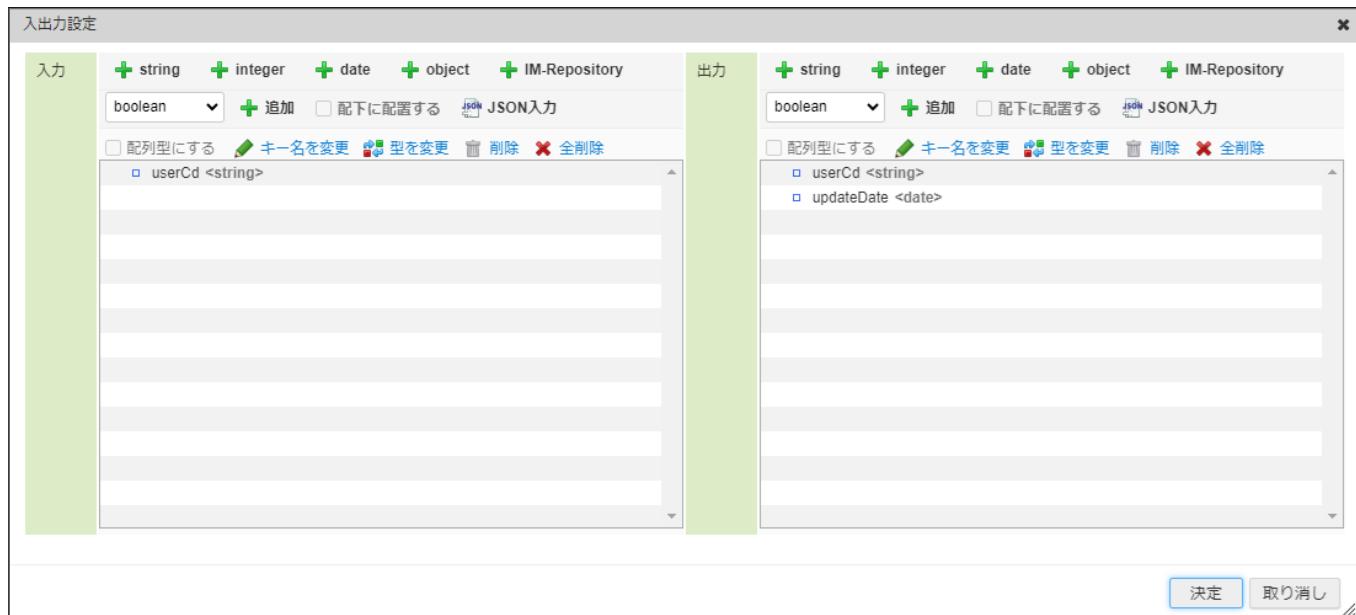
IM-LogicDesigner 側では string 型を使用し、「yyyy-MM-dd」フォーマットで返却してください。



IM-LogicDesigner で保存されている日付を返却する

ロジックフローの「入出力設定」でデータベースから取得した情報を返却する出力を設定します。

日付を扱う「updateDate」のデータ型「date」に設定します。



データベースから取得した日付を返却するSQLユーザ定義タスクを用意します。

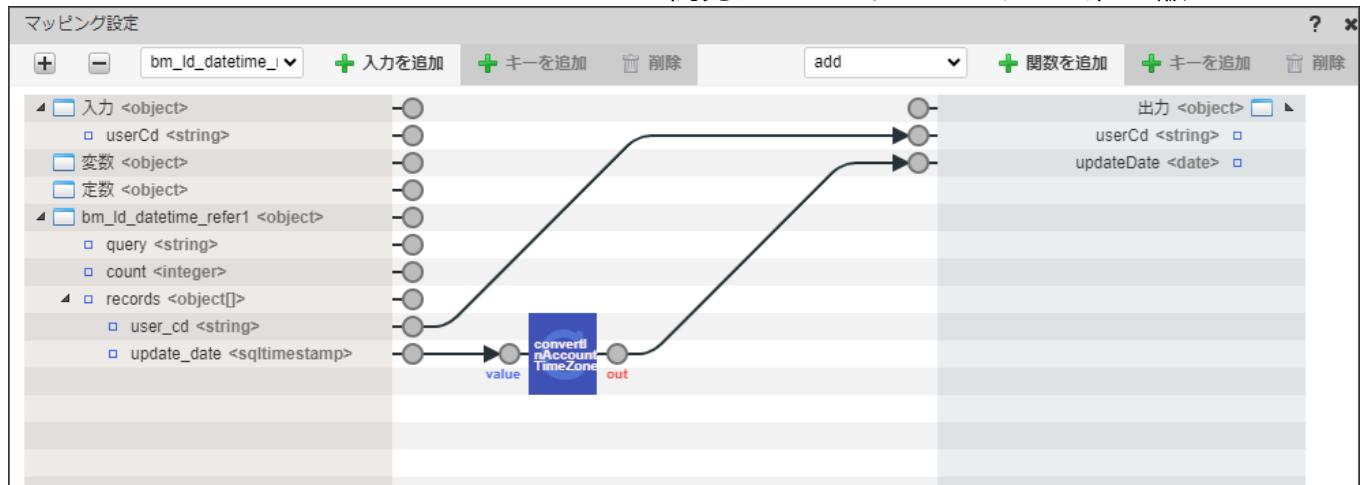
SQLユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|--------------------------------|---|
| クエリ種別 | SELECT |
| 入力値 | userCd <string> |
| 返却値 (records <object[]> の中の要素) | user_cd <string> update_date <sqltimestamp> |
| クエリ | SELECT user_cd, update_date FROM example_table WHERE user_cd = /*userCd*/'aoyagi' |

用意したSQLユーザ定義タスクから取得した日付をマッピング設定でユーザのタイムゾーンでの日付に変換します。

マッピング設定でSQLユーザ定義タスクから取得した日付をマッピング関数「convertInAccountTimeZone」を利用してユーザのタイムゾーンでの日付に変換します。

変換後の日付をロジックフローの出力値として設定します。



作成したロジックフローにおける日付の流れ

データベースに保存されている日付が以下の日付であるとします。

2012-09-19T00:00:00+00:00

日付を相対的に扱うために、マッピング設定でSQLユーザ定義タスクから取得した上記の日付をマッピング関数「convertInAccountTimeZone」を利用してユーザーのタイムゾーンでの日付に変換しています。日付は以下に変換されます。

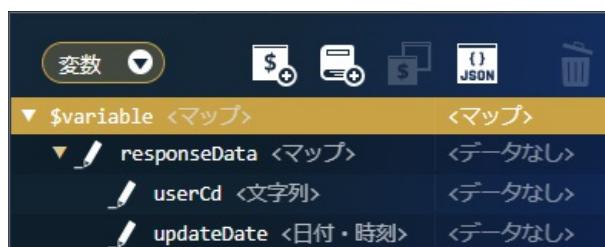
2012-09-19T00:00:00+09:00

変換された日付がロジックフローの出力値として設定されており、出力の型が *date* のため、IM-BloomMaker にはシステムタイムゾーン (GMT) での日付を返却します。

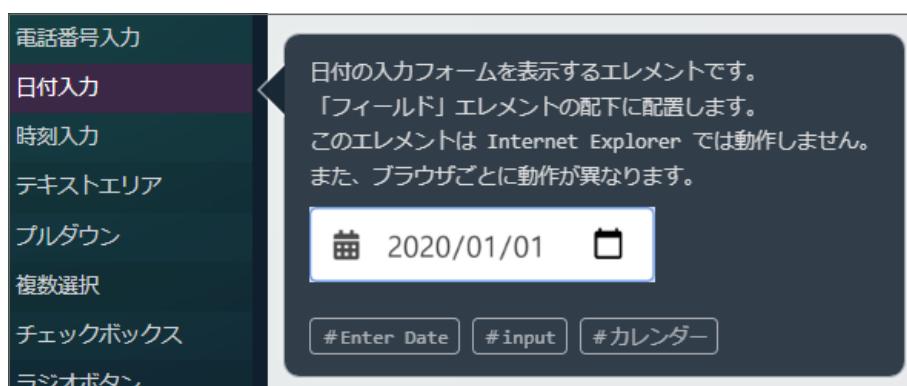
2012-09-18T15:00:00+00:00

IM-BloomMaker で日付を表示する

日付入力エレメントを利用して日付を扱うために変数を設定します。
日付である「updateDate」のデータ型を「日付・時刻」に設定します。



日付を表示するために日付入力エレメントを設定します。



日付入力エレメントの value プロパティには「日付・時刻」型である「updateDate」を設定します。

日付入力要素

⌚ エレメント固有

disabled ②
 固定値 変数値
■

readonly ②
 固定値 変数値
✓

static ②
 固定値 変数値
✓

autofocus ②
 固定値 変数値
■

name ②
 固定値 変数値

value ②
\$variable.responseData.updateDate

アクション「IM-LogicDesigner フロールーティング○にリクエストを送信する」のレスポンスデータとして日付を含んだマップ型の変数を設定します。

⌚ アクションアイテム説明

IM-LogicDesigner フロールーティング `tutorial/sample/bm_ld_datetime/date` にリクエストを送信する

| | | | | |
|---------------|--------------------------|------------------------------|----------------------|-------|
| リクエストパラメータ | IN | \${} \$input | | |
| 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 | \${} \$variable.responseData | | |
| セキュアトークンを送信する | <input type="checkbox"/> | | | |
| エポックミリ秒を送信しない | <input type="checkbox"/> | | | |
| (全ての条件に一致) | | | | |
| 変数 | \${} \$input.userCd | が | \${} \$env.const.nul | でないとき |

作成したコンテンツにおける日付の流れ

IM-LogicDesigner から以下の日付が返却されたとします。

2012-09-18T15:00:00+00:00

日付入力エレメントではユーザのタイムゾーンでの日付を表示するため以下のように表示されます。

2012/09/19

保存されている日付一覧を画面に一覧表示する

IM-LogicDesigner で日付は date 型で扱います。

IM-BloomMaker で日付を参照する際に日付を相対的に扱うためには、システムタイムゾーンでデータベースに保存されている日付をユーザ

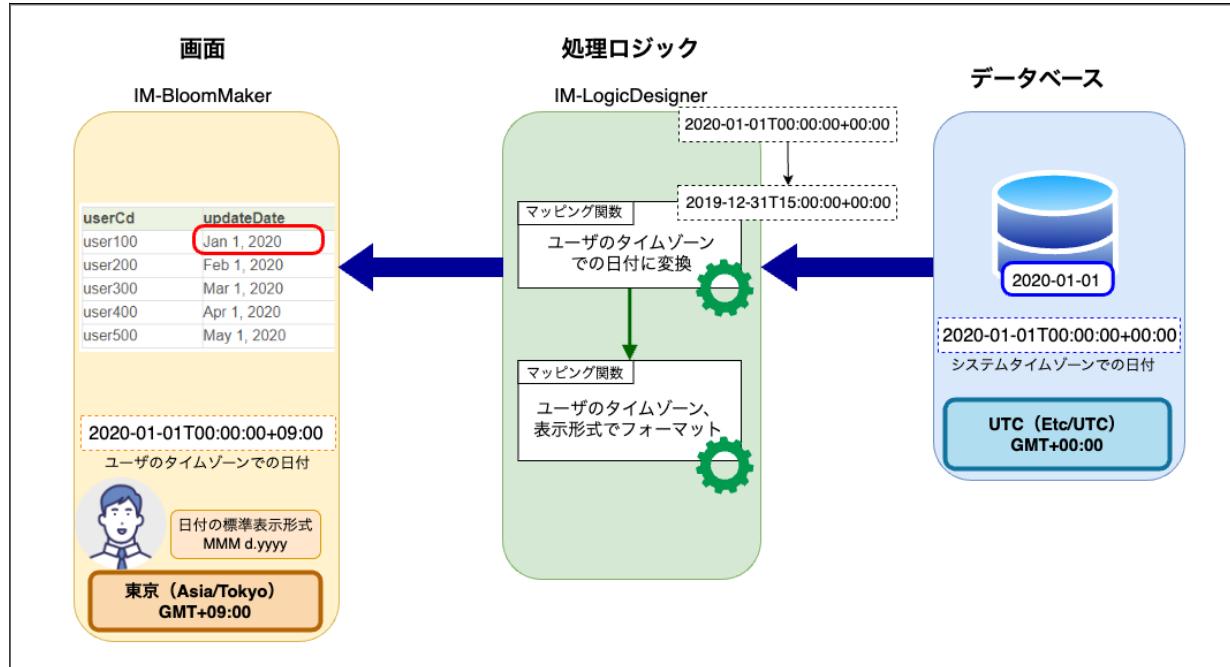
また、日付を一覧する際には日付をユーザの日付の時刻の表示形式で文字列に変換する必要があります。

データベースから取得した日付をユーザのタイムゾーンでの日付に変換した後に日付文字列に変換し、ロジックフローの出力値として設定します。

IM-BloomMaker で日付を一覧表示する画面を作成する場合、テーブル（繰り返し）エレメントを利用します。

テーブル（繰り返し）エレメントで日付を表示する列には、文字列型の変数を設定します。

IM-LogicDesigner から取得した情報を「文字列」型の変数に設定することで日付を画面に一覧表示します。



IM-LogicDesigner で保存されている日付文字列を返却する

ロジックフローの「入出力設定」でデータベースから取得した情報を返却する出力を設定します。

日付を扱う「`updateDate`」のデータ型「`string`」に設定します。



データベースから取得した日付を返却するDatabase Fetchユーザ定義タスクを用意します。

Database Fetchユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|-------|--|
| クエリ種別 | SELECT |
| 入力値 | なし |
| 返却値 | <code>user_cd <string></code> <code>update_date <sqltimestamp></code> |

設定項目

値

クエリ

```
SELECT user_cd, update_date FROM example_table ORDER
BY user_cd
```

用意したDatabase Fetchユーザタスクから取得した日付をマッピング設定でユーザの日付の時刻の表示形式で文字列に変換します。

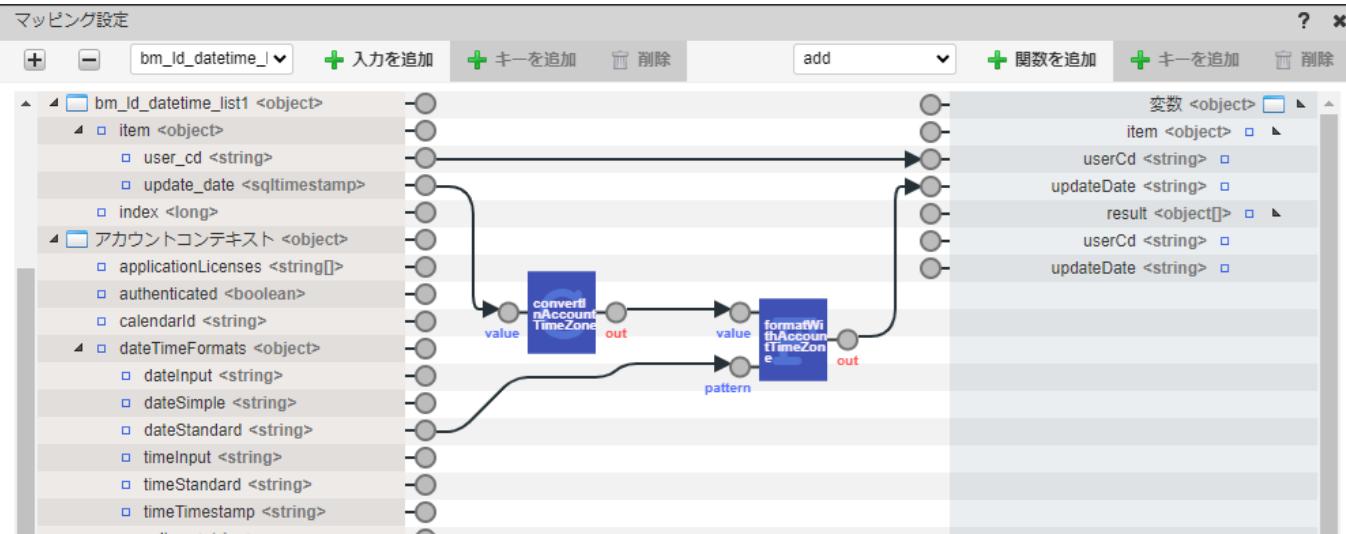
Database Fetchユーザタスクから取得した日付をマッピング関数「convertInAccountTimeZone」を利用してユーザのタイムゾーンでの日付に変換します。

マッピング関数「formatWithAccountTimeZone」を利用してユーザの日付の時刻の表示形式で文字列に変換します。

マッピング関数「formatWithAccountTimeZone」の入力の「value」にユーザのタイムゾーンに変換した日付を設定します。

日付形式文字列のフォーマットとして、ユーザが設定している「日付（標準表示）」形式とするためにマッピング関数

「formatWithAccountTimeZone」の「pattern」に「アカウントコンテキスト - dateTimeFormats - dateStandard」を設定します。



作成したロジックフローにおける日付の流れ

データベースに保存されている日付が以下の日付であるとします。

```
2012-09-19T00:00:00+00:00
```

日付を相対的に扱うために、マッピング設定でDatabase Fetchユーザ定義タスクから取得した上記の日付をマッピング関数「convertInAccountTimeZone」を利用してユーザのタイムゾーンでの日付に変換しています。

日付は以下に変換されます。

```
2012-09-19T00:00:00+09:00
```

マッピング関数「formatWithAccountTimeZone」を利用してユーザのタイムゾーンでの日付をユーザの日付の時刻の表示形式で文字列に変換しています。

マッピング関数「formatWithAccountTimeZone」のフォーマットとしてユーザの「日付（標準表示）」形式を指定しているため日付は以下の文字列に変換されます。

```
Sep 19, 2012
```

IM-BloomMaker で日付一覧を表示する

テーブル（繰り返し）エレメントを利用して日付を一覧表示するために変数を設定します。

日付である「updateDate」のデータ型を「文字列」に設定します。

変数

- \$variable <マップ>
- ▶ headerRowString <マップ>
- ▼ responseData <マップ> <マップ>
 - ▶ result <マップ> <配列>
 - ▶ ① 0 <マップ>
 - ▶ userCd <文字列> <データなし>
 - ▶ updateDate <文字列> <データなし>

日付を一覧表示するためにテーブル（繰り返し）エレメントを設定します。

ス（繰り返し）

- テーブル（繰り返し）
- リスト（繰り返し）
- 連番付きリスト（繰り返し）
- ① 繰り返し（Bulma）**
- ② フォーム部品
- ③ フォーム部品（Bulma）
- ④ 共通マスタ
- ⑤ 逆用

表形式でエレメントを繰り返し配置するエレメントです。
listプロパティで指定した配列の要素数分エレメントが配置されます。
他のエレメントと比較して読み込みが遅いため注意してください。
インラインフレックス（繰り返し）エレメントを代替エレメントとして使用することを推奨します。

| | | |
|---------|---------|---------|
| Element | Element | Element |
| Element | Element | Element |
| Element | Element | Element |

#Table (repeat) #table

テーブル（繰り返し）エレメントの list プロパティには日付文字列を含んだマップ型の配列を設定します。

テーブル（繰り返し）

エレメント固有

showBorder 固定値 変数値
②

showHeader 固定値 変数値
②

showFooter 固定値 変数値
②

columnCount 2

list ② \$variable.responseData.result

アクション「IM-LogicDesigner フロールーティング○にリクエストを送信する」のレスポンスデータとして日付を含んだマップ型の配列変数を設定します。

The screenshot shows the IM-LogicDesigner interface with a flow titled "IM-LogicDesigner フロールーティング" (tutorial/sample/bm_ld_datetime/date_list). The flow has the following steps:

- リクエストパラメータ (IN): \${}
- 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): \${} \$variable.responseData
- セキュアトークンを送信する (checkbox)
- エポックミリ秒を送信しない (checkbox)

A button at the bottom left says "常に実行" (Always Run).

保存されている日付一覧をリッチテーブルに一覧表示する

IM-LogicDesigner で日付は *date* 型で扱います。

IM-BloomMaker で日付を参照する際に日付を相対的に扱うためには、システムタイムゾーンでデータベースに保存されている日付をユーザのタイムゾーンでの日付に変換する必要があります。 (マッピング関数)

リッチテーブルでは、日付入力エレメントと異なり、タイムゾーンを変換せず IM-LogicDesigner が返却した日付をそのまま表示します。したがって、日付を一覧表示する際には、あらかじめ IM-LogicDesigner 側で、ユーザタイムゾーンの日付で IM-BloomMaker に返却する必要があります。 (アクションアイテムでオプションを指定)

IM-BloomMaker では IM-LogicDesigner から取得した情報を「日付・時刻」型の変数に設定します。

リッチテーブルエレメントで日付を扱う場合は、columnSettingsプロパティで「type=dateOnlyDate」を設定します。

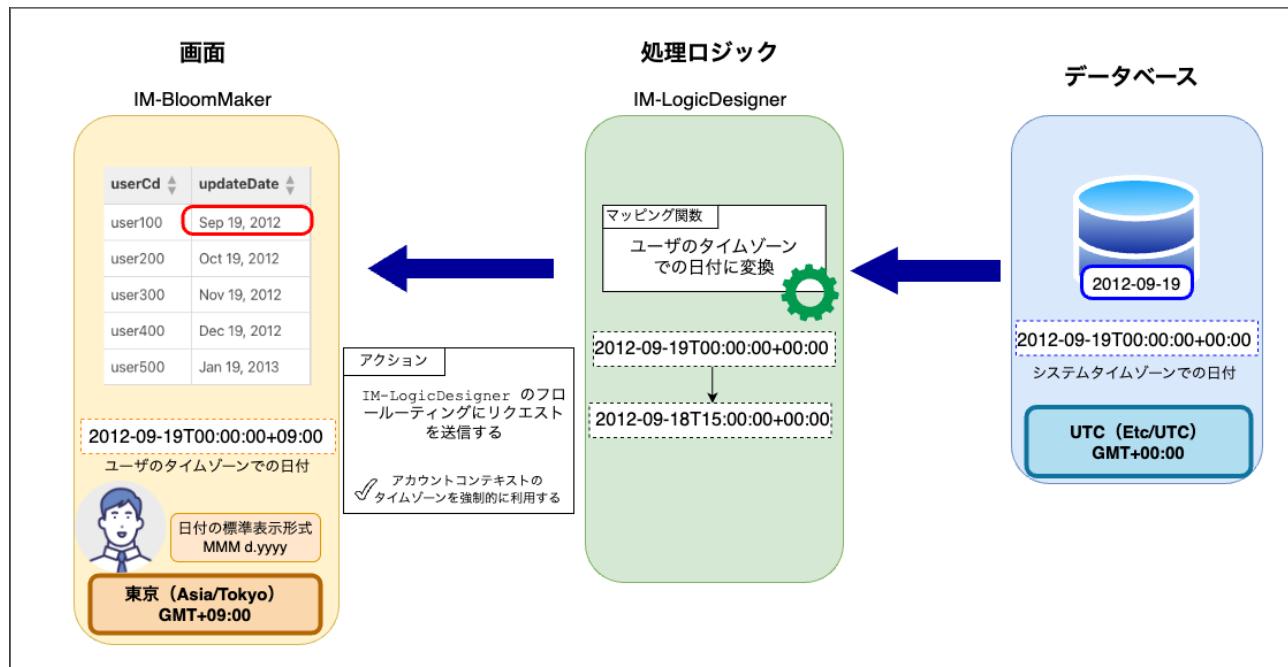
リッチテーブルで日付を扱う場合、allowEditingプロパティをTrueに設定することで日付を編集できます。

i コラム

ユーザのタイムゾーンに依存せず、どの地域でも同一の日付を表示したい場合（日時と同様に絶対的として取り扱いたい場合）は、日付を文字列として扱うことを推奨します。

columnSettingsプロパティで「type=dateOnlyString」を設定すると、リッチテーブル上では「yyyy-MM-dd」形式の文字列で取り扱います。

IM-LogicDesigner 側では *string* 型を使用し、同じ日付フォーマットで返却してください。



IM-LogicDesigner で保存されている日付を返却する

ロジックフローの「入出力設定」でデータベースから取得した情報を返却する出力を設定します。

日付を扱う「updateDate」のデータ型「date」に設定します。

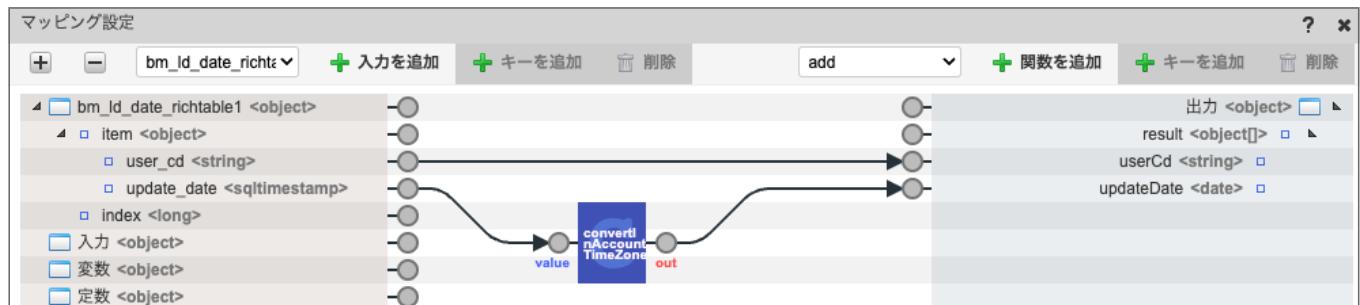


データベースから取得した日付を返却するDatabase Fetchユーザ定義タスクを用意します。

Database Fetchユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|-------|---|
| クエリ種別 | SELECT |
| 入力値 | なし |
| 返却値 | user_cd <string> update_date <sqltimestamp> |
| クエリ | SELECT user_cd, update_date FROM example_table ORDER BY user_cd |

Database Fetchユーザタスクから取得した日付をマッピング関数「convertInAccountTimeZone」を利用してユーザのタイムゾーンでの日付に変換します。



IM-BloomMaker で日付一覧をリッチテーブルに表示する

リッチテーブルエレメントを利用して日付を一覧表示するために変数を設定します。

日付である「updateDate」のデータ型を「日付・時刻」型に設定します。



日付を一覧表示するためにリッチテーブルエレメントを設定します。

プロトタイプ

- リッチテーブル
- 履歴・コメント
- ④ フォーム部品 (Bulma)
- ④ 共通マスタ
- ④ 汎用

ソート、ページング、簡易検索、データ編集機能を持つテーブルエレメントです。

| User code | User name | Membership department |
|-----------|-----------------|-----------------------|
| Analyst | Tatsuya Analyst | Sample section 11 |
| Works | Tatsuya Works | Sample section 23 |

Page: 1 of 1 13 | Display 1 - 2 out of 2 items

#Rich Table #table

リッチテーブルエレメントの datasource プロパティには日付を含んだマップ型の配列を設定します。

リッチテーブルエレメントの headerRowString プロパティ、 columnSettings プロパティはそれぞれ以下のように設定します。

columnSettings プロパティの日付を表示する column (列) には、「type=dateOnlyDate」を設定します。

フォーム部品

リッチテーブル

④ エレメント固有

- headerRowString ②
\$variable.headerRowString
- dataSource ②
\$variable.responseData
- columnSettings ②
\$variable.columnSettings

アクション「IM-LogicDesigner フロールーティング○にリクエストを送信する」のレスポンスデータとして日付を含んだマップ型の配列変数を設定します。

リッチテーブルエレメントで日付を正しく表示するためには、「アカウントコンテキストのタイムゾーンを強制的に使用する」プロパティにチェックを入れ、ユーザタイムゾーンの日付でIM-BloomMakerに返却する必要があります。

アクションアイテム説明

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

リクエストパラメータ IN \${}

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 \${} \$variable.responseData

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 ■

アカウントコンテキストのタイムゾーンを強制的に使用する

常に実行

ロジックフローからBloomMakerに返却される日付の流れ

データベースに保存されている日付が以下の日付であるとします。

2012-09-19T00:00:00+00:00

日付を相対的に扱うために、マッピング設定で Database Fetch ユーザ定義タスクから取得した上記の日付をマッピング関数「convertInAccountTimeZone」を利用してユーザのタイムゾーンでの日付に変換しています。
日付は以下に変換されます。

2012-09-19T00:00:00+09:00

変換された日付がロジックフローの出力値として設定されており、出力の型が *date* のため、以下のようにシステムタイムゾーン（GMT）に変換されます。

2012-09-18T15:00:00+00:00

ただし「アカウントコンテキストのタイムゾーンを強制的に使用する」プロパティにチェックを入れているため、最終的には IM-BloomMaker にはユーザのタイムゾーンでの日付を返却します。

2012-09-19T00:00:00+09:00

リッチテーブルエレメントのその他の設定

リッチテーブルには、日付・時刻に関連する *type* を4つ用意しています。以下に使い分けについて示します。

| type | ユースケース | BloomMakerの変数型 | 備考 |
|----------------|----------------------------------|-----------------------|---|
| date | 日時を扱いたい場合に使用します。 | 日付・時刻型 | |
| dateOnlyDate | 日付を扱いたい場合に使用します。 | 日付・時刻型 | 日付入力エレメントと互換性があります。 互換したい場合は、 <i>preserveTime</i> プロパティを <i>false</i> に、 <i>appliedTimeZone</i> プロパティを <i>account-context</i> に設定してください。 |
| dateOnlyNumber | javaScriptのDate型を扱いたい場合を想定しています。 | 整数型 | |
| dateOnlyString | 文字列型で日付を扱いたい場合に使 | 文字列型 | 用します。 |

さらに詳細な仕様は以下のページでご確認ください。

https://document.intra-mart.jp/library/bloommaker/public/im_bloommaker_user_guide/texts/appendix/elements/regular.html#imrichtable

日時

以下のように日時を絶対的に扱うための流れを説明します。

- 「intra-mart Accel Platform タイムゾーン仕様書」 - 「日時は絶対的」

日時を保存する

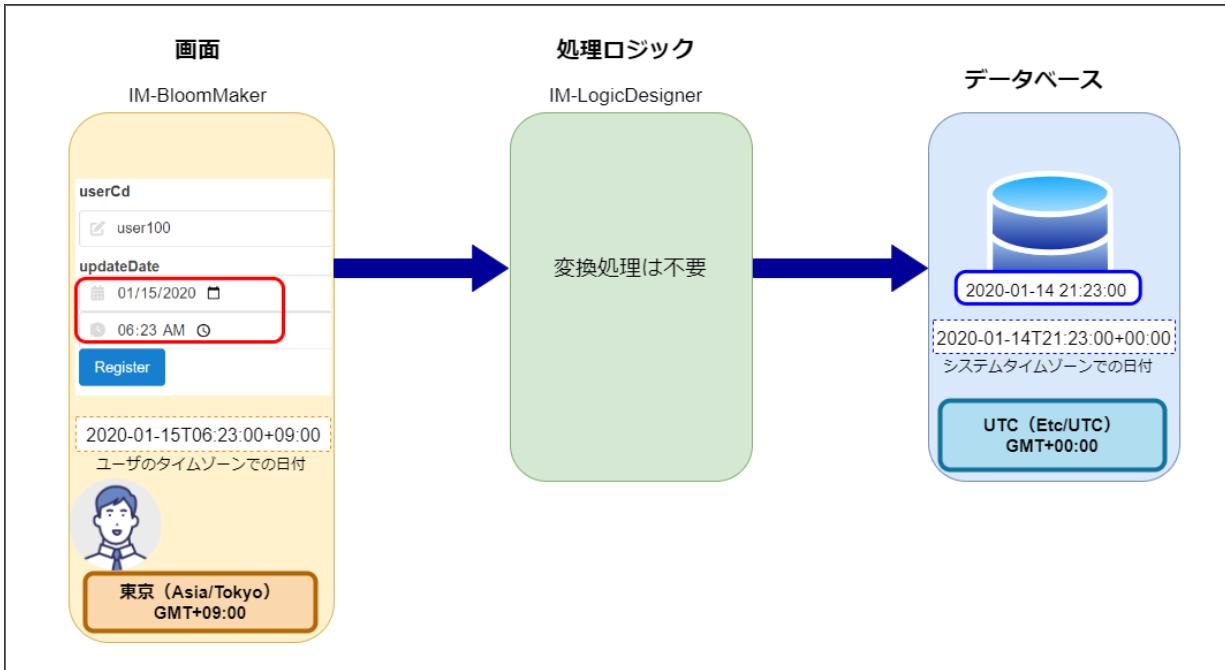
IM-BloomMaker で日時を編集する画面を作成する場合、日付入力エレメントと時刻入力エレメントを利用します。

日時入力エレメント・時刻入力エレメントには、日付・時刻型の変数を設定します。

IM-BloomMaker で選択した日時はユーザのタイムゾーンでの日時です。ユーザのタイムゾーンでの日時を IM-LogicDesigner に送信します。

IM-LogicDesigner で日時は *date* 型で扱います。

日時を絶対的に扱うためには、IM-BloomMaker から送信される日時をそのままデータベースに保存します。



IM-BloomMaker から日時の送信する

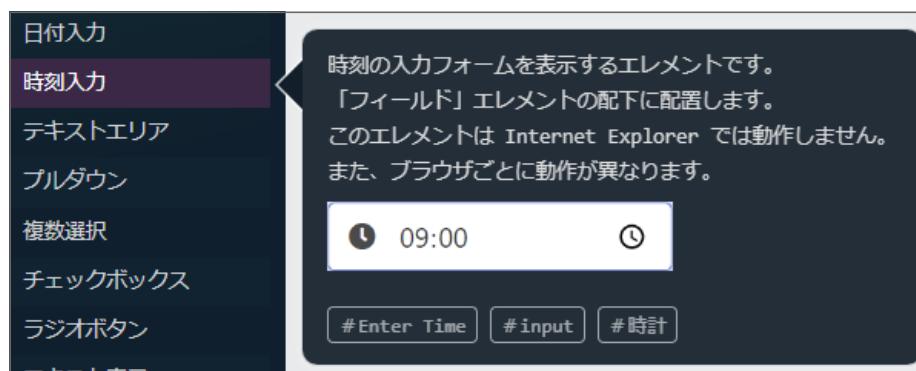
日付入力エレメントと時刻入力エレメントを利用して日時を扱うために変数を設定します。
日時である「`updateDate`」のデータ型を「日付・時刻」に設定します。



日時の日付を扱うために日付入力エレメントを設定します。



日時の時刻を扱うために時刻入力エレメントを設定します。



日付入力エレメントの `value` プロパティには「日付・時刻」型である「`updateDate`」を設定します。`preserveTime` プロパティを有効に設定します。

日付入力要素

⌚

⌚ エレメント固有

disabled ② 固定値 変数値
■

readonly ② 固定値 変数値
■

static ② 固定値 変数値
■

autofocus ② 固定値 変数値
■

name ② 固定値 変数値
[入力欄]

value ② \$variable.requestData.updateDate

max ② [入力欄]

min ② [入力欄]

preserveTime ② 固定値 変数値
②

時刻入力エレメントの value プロパティにも「日付・時刻」型である「updateDate」を設定します。

時刻入力要素

⌚

⌚ エレメント固有

disabled ② 固定値 変数値
■

readonly ② 固定値 変数値
■

static ② 固定値 変数値
■

autofocus ② 固定値 変数値
■

name ② 固定値 変数値
[入力欄]

value ② \$variable.requestData.updateDate

アクション「IM-LogicDesigner フロールーティング○にリクエストを送信する」のリクエストパラメータとして日時を含んだマップ型の変数を設定します。

The screenshot shows the IM-LogicDesigner interface with a flow configuration. The top bar includes a toggle switch, a circular icon with a play button, and a search bar labeled "アクションアイテム説明". Below the bar, the title is "IM-LogicDesigner フロールーティング" and the URL is "tutorial/sample/bm_ld_datetime/datetime". A button "にリクエストを送信する" (Send Request) is visible. The main area displays several configuration items:

- リクエストパラメータ (Request Parameter): IN \${variable.requestData}
- リクエストヘッダ (Request Header): IN \${}
- ステータスコード (Status Code): OUT \${}
- レスポンスデータ (Response Data): OUT \${}
- セキュアトークンを送信する (Send Secure Token): checked
- エポックミリ秒を送信しない (Do not send epoch milliseconds): checked

A green button at the bottom left says "常に実行" (Always Execute).

作成したコンテンツにおける日時の流れ

日付入力エレメントと時刻入力エレメントで以下の日時を入力されたとします。

2012/09/19 03:46

この場合、IM-LogicDesigner に送信される日時は以下のようにユーザのタイムゾーンでの日時として送信されます。

2012-09-19T03:46:00+09:00

IM-LogicDesigner での日時の受信とデータベースへの保存

ロジックフローの「入出力設定」で IM-BloomMaker から送信される情報に応じて入力を設定します。

日時を扱う「updateDate」のデータ型「date」に設定します。

The screenshot shows the "Input/Output Settings" dialog. It has two main sections: "Input" and "Output".

Input: Shows a list of fields:

- boolean
- 追加 (Add)
- 配下に配置する (Configure under)
- JSON Input
- 配列型にする (Change type to array)
- キー名を変更 (Change key name)
- 型を変更 (Change type)
- 削除 (Delete)
- 全削除 (Delete all)

 Below this is a list of specific fields:

- userCd <string>
- updateDate <date>

Output: Similar structure to the input section, showing the same list of fields and configuration options.

At the bottom right are buttons: "決定" (OK) and "取り消し" (Cancel).

入力値の情報をデータベースに保存するSQLユーザ定義タスクを用意します。

SQLユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|-------|---|
| クエリ種別 | INSERT |
| 入力値 | userCd <string> updateDate <sqltimestamp> |
| 返却値 | なし |
| クエリ | <pre>INSERT INTO example_table (user_cd, update_date) VALUES (/*userCd*/null, /*updateDate*/null)</pre> |



作成したロジックフローにおける日時の流れ

IM-BloomMaker から以下の日時が送信されたとします。

2012-09-19T03:46:00+09:00

ロジックフロー内ではオフセット情報を持たないデータ型「date」で日時を扱っているため、送信された日時は以下の値として受け付けます。

2012-09-18T18:46:00+00:00

日時は絶対的に扱うため IM-BloomMaker から送信された日時がそのままデータベースに保存されます。

保存されている日時を画面に表示する

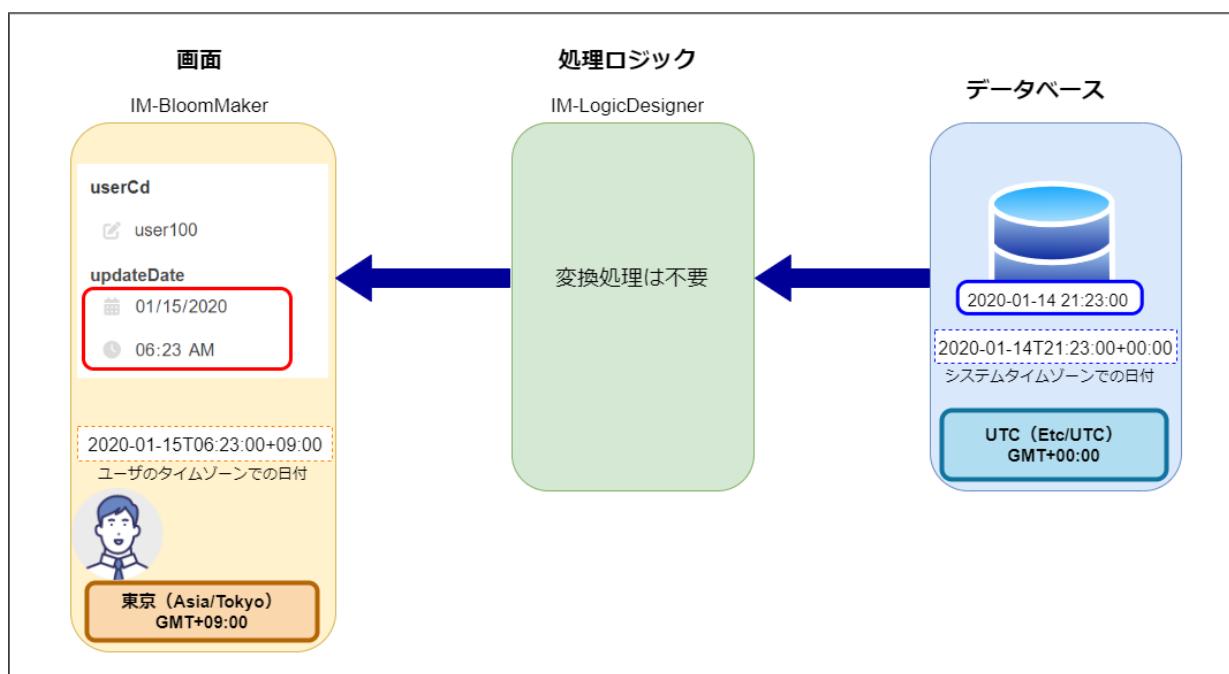
IM-LogicDesigner で日時は date 型で扱います。

IM-BloomMaker で日時を参照する際に日付を絶対的に扱うためには、データベースに保存されている日時をそのままロジックフローの出力値として設定します。

IM-BloomMaker では日付入力エレメントと時刻入力エレメントを利用して日時を表示します。

日付入力エレメントには、日付・時刻型の変数を設定します。

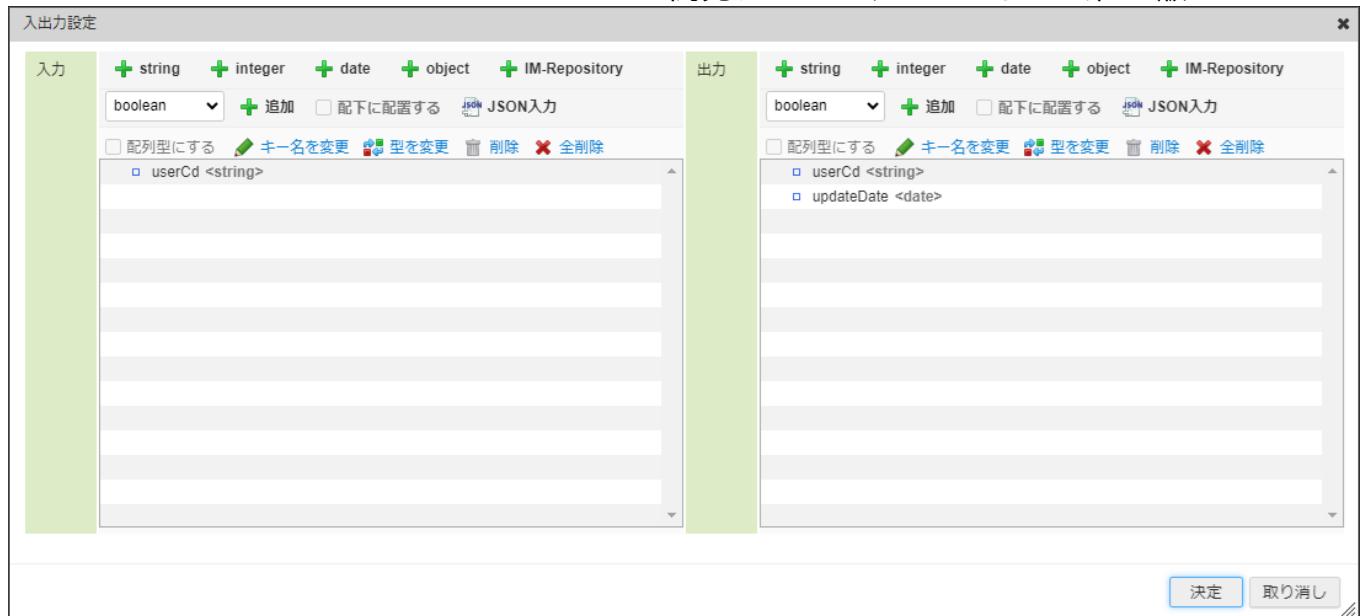
IM-LogicDesigner から取得した情報を「日付・時刻」型の変数に設定することで日時を画面に表示します。



IM-LogicDesigner で保存されている日時を返却する

ロジックフローの「入出力設定」でデータベースから取得した情報を返却する出力を設定します。

日時を扱う「updateDate」のデータ型「date」に設定します。

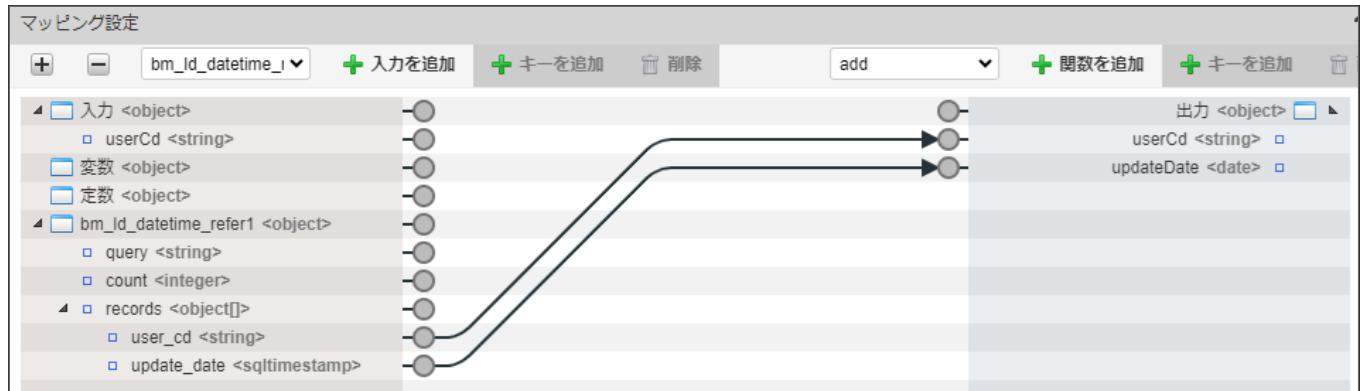


データベースから取得した日時を返却するSQLユーザ定義タスクを用意します。

SQLユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|--------------------------------|---|
| クエリ種別 | SELECT |
| 入力値 | なし |
| 返却値 (records <object[]> の中の要素) | user_cd <string> update_date <sqltimestamp> |
| クエリ | SELECT user_cd, update_date FROM example_table WHERE user_cd = /*userCd*/'aoyagi' |

用意したSQLユーザタスクから取得した日時をロジックフローの出力値として設定します。



作成したロジックフローにおける日時の流れ

データベースに保存されている日時が以下の日時であるとします。

2012-09-18T18:46:00+00:00

ロジックフローの出力値として設定された日時が IM-BloomMaker に返却されます。

IM-BloomMaker で日時を表示する

日付入力エレメントと時刻入力エレメントを利用して日時を扱うために変数を設定します。

日時である「updateDate」のデータ型を「日付・時刻」に設定します。

The screenshot shows the variable mapping configuration. A variable named '\$variable' is mapped to 'responseData'. Within 'responseData', the 'userCd' field is mapped to 'データなし' (Data Not Available) and the 'updateDate' field is also mapped to 'データなし'.

日時を表示するために日付入力エレメントを設定します。

The screenshot shows the element selection interface. The '日付入力' (Date Input) element is selected and highlighted in purple. A detailed description box provides information about the date input element, stating it displays a date input form, is a 'field' element, and is not supported in Internet Explorer. It also notes browser-specific differences. Below the description is a preview of the date input field showing '2020/01/01' and three associated buttons: '#Enter Date', '#input', and '#カレンダー' (Calendar).

日時を表示するために時刻入力エレメントを設定します。

The screenshot shows the element selection interface. The '時刻入力' (Time Input) element is selected and highlighted in purple. A detailed description box provides information about the time input element, stating it displays a time input form, is a 'field' element, and is not supported in Internet Explorer. It also notes browser-specific differences. Below the description is a preview of the time input field showing '09:00' and three associated buttons: '#Enter Time', '#input', and '#時計' (Clock).

日付入力エレメントの value プロパティには「日付・時刻」型である「updateDate」を設定します。

The screenshot shows the configuration interface for the date input element. The 'value' property is set to '\$variable.responseData.updateDate'. The configuration interface includes sections for 'disabled', 'readonly', 'static', 'autofocus', and 'name', each with '固定値' (Fixed Value) and '変数値' (Variable Value) options, with '固定値' selected for all.



アクション「IM-LogicDesigner フロールーティング○にリクエストを送信する」のレスポンスデータとして日時を含んだマップ型の変数を設定します。

IM-LogicDesigner フロールーティング (tutorial/sample/bm_ld_datetime/datetime) にリクエストを送信する

| | | | | |
|--|--------------------------|-----------------------------------|--|-----------------------------------|
| リクエストパラメータ | IN | \$input | <input type="button" value="検索"/> | <input type="button" value="削除"/> |
| リクエストヘッダ | IN | \$input | <input type="button" value="検索"/> | <input type="button" value="削除"/> |
| ステータスコード | OUT | \$input | <input type="button" value="検索"/> | <input type="button" value="削除"/> |
| レスポンスデータ | OUT | \$variableresponseData | <input type="button" value="検索"/> | <input type="button" value="削除"/> |
| セキュアトークンを送信する | <input type="checkbox"/> | | <input type="button" value="削除"/> | |
| エポックミリ秒を送信しない | <input type="checkbox"/> | | <input type="button" value="削除"/> | |
| 全ての条件に一致 <input type="button" value="検索"/> | | | | |
| 変数 | \$input.userCd | <input type="button" value="検索"/> | が \$env.const.null <input type="button" value="検索"/> でないとき | <input type="button" value="削除"/> |

作成したコンテンツにおける日時の流れ

IM-LogicDesigner から以下の日時が返却されたとします。

2012-09-18T18:46:00+00:00

日付入力エレメントと時刻入力エレメントではユーザのタイムゾーンでの日付を表示するため以下のように表示されます。

2012/09/19 03:46

保存されている日時一覧を画面に一覧表示する

IM-LogicDesigner で日時は date 型で扱います。

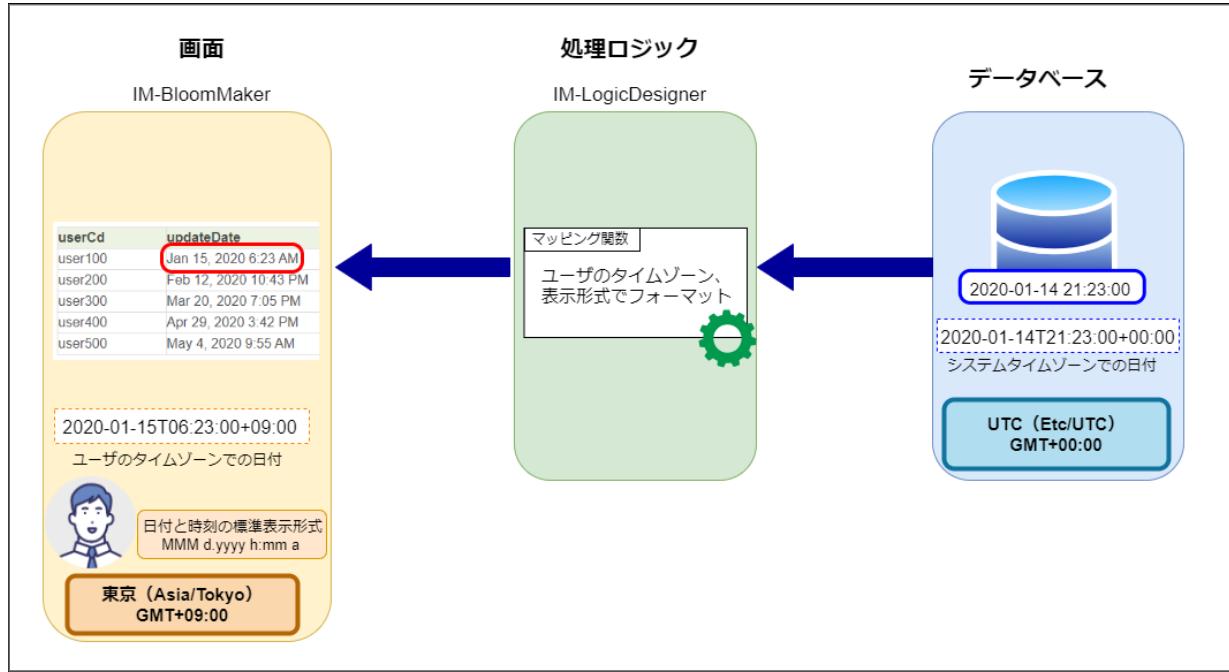
IM-BloomMaker で日時を一覧する際には日時をユーザの日時の時刻の表示形式で文字列に変換する必要があります。

データベースから取得した日時を相対的な時間での日時文字列に変換し、ロジックフローの出力値として設定します。

IM-BloomMaker で日時を一覧表示する画面を作成する場合、テーブル（繰り返し）エレメントを利用します。

テーブル（繰り返し）エレメントで日時を表示する列には、文字列型の変数を設定します。

IM-LogicDesigner から取得した情報を「文字列」型の変数に設定することで日時を画面に一覧表示します。



IM-LogicDesigner で保存されている日時文字列を返却する

ロジックフローの「入出力設定」でデータベースから取得した情報を返却する出力を設定します。

日時を扱う「`updateDate`」のデータ型「string」に設定します。



データベースから取得した日時を返却するDatabase Fetchユーザ定義タスクを用意します。

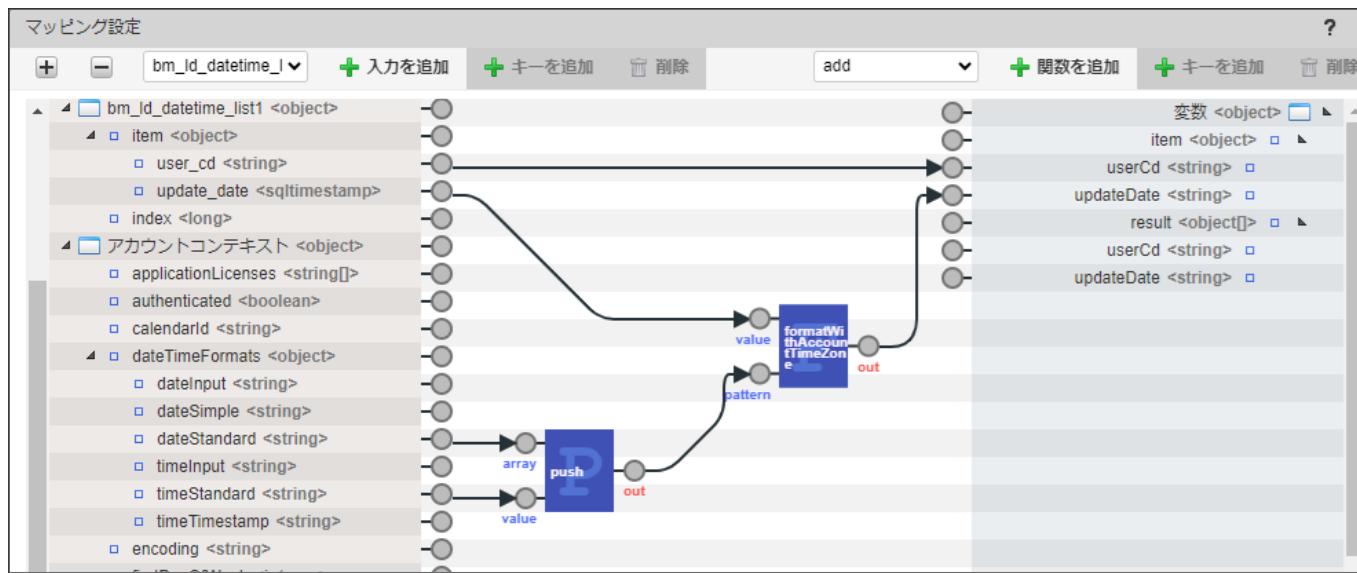
Database Fetchユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|-------|--|
| クエリ種別 | SELECT |
| 入力値 | なし |
| 返却値 | <code>user_cd <string></code> <code>update_date <sqltimestamp></code> |
| クエリ | <code>SELECT user_cd, update_date FROM example_table ORDER BY user_cd</code> |

用意したDatabase Fetchユーザタスクから取得した日時をマッピング設定でユーザーの日時の時刻の表示形式で文字列に変換します。

マッピング関数「formatWithAccountTimeZone」を利用してユーザーの日時の時刻の表示形式で文字列に変換します。

マッピング関数「formatWithAccountTimeZone」の入力の「value」にDatabase Fetchユーザタスクから取得した日時を設定します。日時形式文字列のフォーマットとして、ユーザーが設定している「日付（標準表示）」形式と「時刻（標準表示）」を組み合わせた形式にするためにマッピング関数「formatWithAccountTimeZone」の「pattern」に「アカウントコンテキスト - dateTimeFormats - dateStandard」と「アカウントコンテキスト - dateTimeFormats - timeStandard」の2つを配列操作マッピング関数「push」を利用して設定します。



作成したロジックフローにおける日時の流れ

データベースに保存されている日時が以下の日時であるとします。

2012-09-18T18:46:00+00:00

マッピング関数「formatWithAccountTimeZone」を利用して日時をユーザーの日付の時刻の表示形式で文字列に変換しています。

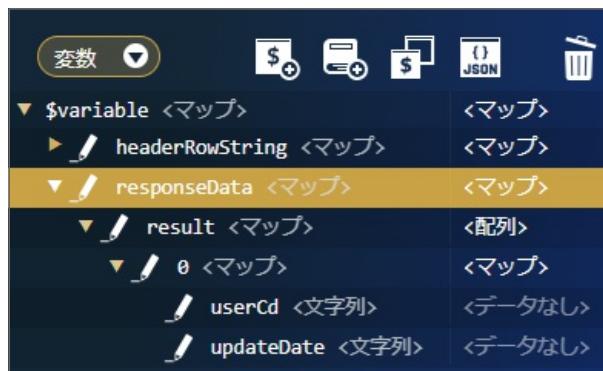
マッピング関数「formatWithAccountTimeZone」のフォーマットとしてユーザーの「日付（標準表示）」と「時刻（標準表示）」形式を指定しているため日時は以下の文字列に変換されます。

Sep 19, 2012 3:46 AM

IM-BloomMaker で日時一覧を表示する

テーブル（繰り返し）エレメントを利用して日時を一覧表示するために変数を設定します。

日時である「updateDate」のデータ型を「文字列」に設定します。



日時を一覧表示するためにテーブル（繰り返し）エレメントを設定します。

| |
|------------------|
| ス (繰り返し) |
| テーブル (繰り返し) |
| リスト (繰り返し) |
| 連番付きリスト (繰り返し) |
| ④ 繰り返し (Bulma) |
| ④ フォーム部品 |
| ④ フォーム部品 (Bulma) |
| ④ 共通マスタ |
| ④ 次用 |

表形式でエレメントを繰り返し配置するエレメントです。
listプロパティで指定した配列の要素数分エレメントが配置されます。
他のエレメントと比較して読み込みが遅いため注意してください。
オンラインフレックス (繰り返し) エレメントを代替エレメントとして使用することを推奨します。

| | | |
|---------|---------|---------|
| Element | Element | Element |
| Element | Element | Element |
| Element | Element | Element |

#Table (repeat) #table

テーブル (繰り返し) エレメントの list プロパティには日時文字列を含んだマップ型の配列を設定します。

テーブル (繰り返し)

④ エレメント固有

showBorder 固定値 変数値
②

showHeader 固定値 変数値
②

showFooter 固定値 変数値
②

columnCount 2
②

list ② \$variable.responseData.result

アクション「IM-LogicDesigner フローラーティング○にリクエストを送信する」のレスポンスデータとして日時を含んだマップ型の配列変数を設定します。

アクションアイテム説明

IM-LogicDesigner フローラーティング tutorial/sample/bm_ld_datetime/datetime_list にリクエストを送信する

| | | | | |
|---------------|-----|---------------------------------|----|----|
| リクエストパラメータ | IN | \${} \$variable.responseData | 検索 | 削除 |
| リクエストヘッダ | IN | \${} \$variable.responseData | 検索 | 削除 |
| ステータスコード | OUT | \${} \$variable.responseData | 検索 | 削除 |
| レスポンスデータ | OUT | \${} \$variable.responseData | 検索 | 削除 |
| セキュアトークンを送信する | ■ | | 検索 | 削除 |
| エポックミリ秒を送信しない | ■ | | 検索 | 削除 |

常に実行

保存されている日時一覧をリッチテーブルに一覧表示する

IM-LogicDesigner で日時は date 型で扱います。

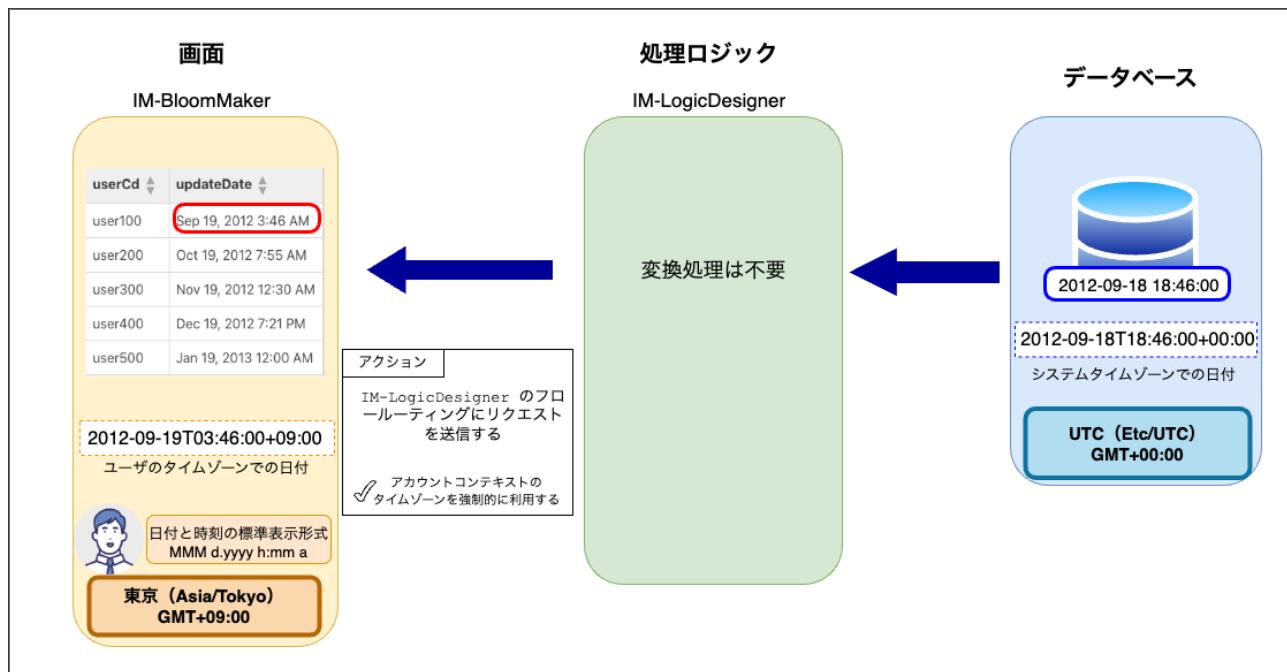
データベースから取得した日時をロジックフローの出力値として設定します。 (マッピング)

また、リッチテーブルで日時を一覧表示する際にはユーザタイムゾーンの日時でIM-BloomMakerに返却する必要があります。 (アクション)

IM-BloomMaker では IM-LogicDesigner から取得した情報を「日付・時刻」型の変数に設定します。

リッチテーブルエレメントで日時を扱う場合は、columnSettingsプロパティで「type=date」を設定します。

リッチテーブルで日時を扱う場合、allowEditingプロパティをTrueにしても、日時を編集することはできません。（時刻を編集できないため、必ず日付（時刻が00:00:00）が保存されます。）



IM-LogicDesigner で保存されている日時を返却する

ロジックフローの「入出力設定」でデータベースから取得した情報を返却する出力を設定します。

日時を扱う「updateDate」のデータ型「date」に設定します。



データベースから取得した日時を返却するDatabase Fetchユーザ定義タスクを用意します。

Database Fetchユーザ定義タスクは以下が設定されているものとします。

| 設定項目 | 値 |
|-------|---|
| クエリ種別 | SELECT |
| 入力値 | なし |
| 返却値 | user_cd <string> update_date <sqltimestamp> |
| クエリ | SELECT user_cd, update_date FROM example_table ORDER BY user_cd |

用意したDatabase Fetchユーザタスクから取得した日時をマッピングします。



IM-BloomMaker で日時一覧をリッチテーブルに表示する

リッチテーブルエレメントを利用して日時を一覧表示するために変数を設定します。

日時である「updateDate」のデータ型を「日付・時刻」型に設定します。



日時を一覧表示するためにリッチテーブルエレメントを設定します。



リッチテーブルエレメントの datasource プロパティには日時を含んだマップ型の配列を設定します。

リッチテーブルエレメントのheaderRowStringプロパティ、columnSettingsプロパティはそれぞれ以下のように設定します。

columnSettingsプロパティの日付を表示するcolumn（列）には、「type=date」を設定します。



アクション「IM-LogicDesigner フロールーティング○にリクエストを送信する」のレスポンスデータとして日付を含んだマップ型の配列変数を設定します。

リッチテーブルエレメントで日時を正しく表示するためには、「アカウントコンテキストのタイムゾーンを強制的に使用する」プロパティに必ずチェックに入れてください。

IM-LogicDesigner フロールーティング `date_richtable` にリク
エストを送信する

リクエストパラメータ `IN` `$()`

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` `$()` `$variable.responseData`

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

アカウントコンテキストのタイムゾーンを強制的に使用する

常に実行

ロジックフローからBloomMakerに返却される日時の流れ

データベースに保存されている日時が以下の日時であるとします。

2012-09-18T18:46:00+00:00

IM-BloomMaker に日時を返却する際に、ユーザタイムゾーンの日時に変換されます。

2012-09-19T03:46:00+09:00

リッチテーブルエレメントのその他の設定

リッチテーブルには、日付・時刻に関連するtypeを4つ用意しています。以下に使い分けについて示します。

| type | ユースケース | BloomMakerの変数型 | 備考 |
|----------------|----------------------------------|----------------|---|
| date | 日時を扱いたい場合に使用します。 | 日付・時刻型 | |
| dateOnlyDate | 日付を扱いたい場合に使用します。 | 日付・時刻型 | 日付入力エレメントと互換性があります。 互換したい場合は、preserveTimeプロパティをfalseに、appliedTimeZoneプロパティをaccount-contextに設定してください。 |
| dateOnlyNumber | javaScriptのDate型を扱いたい場合を想定しています。 | 整数型 | |
| dateOnlyString | 文字列型で日付を扱いたい場合に使用します。 | 文字列型 | |

さらに詳細な仕様は以下のページでご確認ください。

https://document.intra-mart.jp/library/bloommaker/public/im_bloommaker_user_guide/texts/appendix/elements/regular.html#imrichtable

個別の機能要件

以下に記載されている通り、intra-mart Accel Platform における日時は、絶対的に扱われます。

- 「intra-mart Accel Platform タイムゾーン仕様書」 - 「日時は絶対的」

この仕様を実現するために、入力された日時をシステムタイムゾーンに変換して保存し、ログインユーザのタイムゾーンに変換して画面に表示する必要があります、具体的な実装方法については「[日時](#)」に記載しています。

しかし、個別の機能要件として、日時をそのまま保存し、どのタイムゾーンのログインユーザが確認した場合でも、同じ日時を表示するようにしたいといった場合があります。

本章では、どのログインユーザが確認しても同じ日時を表示するための方法を説明します。

日時を保存する

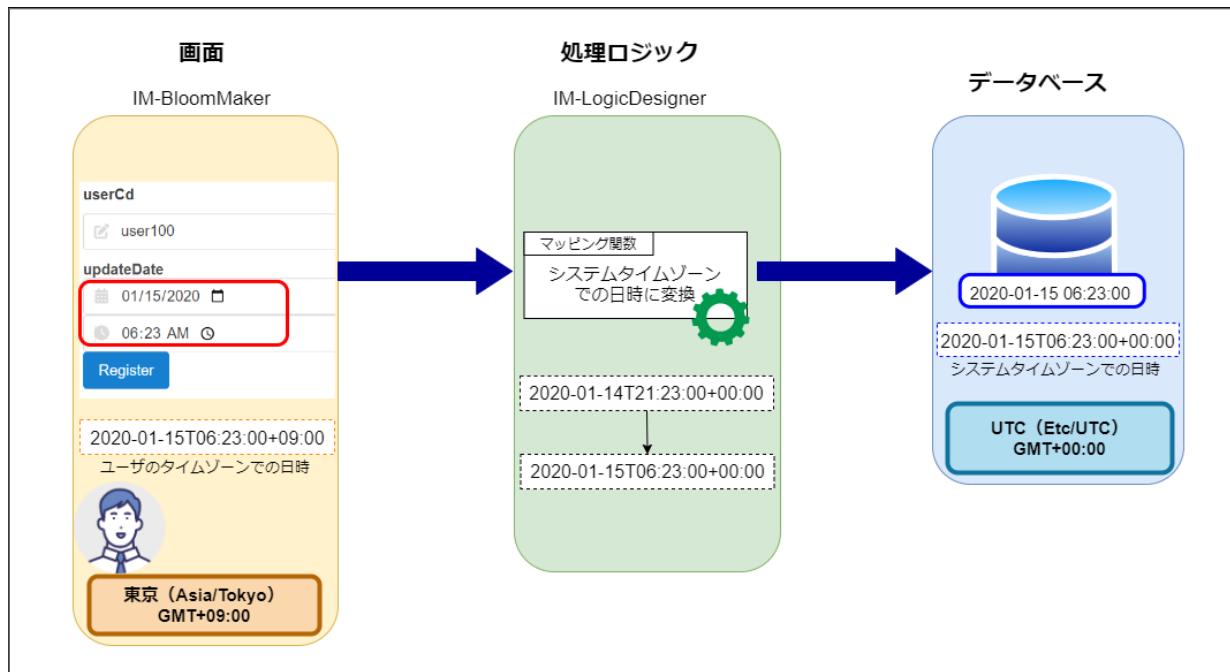
IM-BloomMaker で日時を編集する画面を作成する場合、日付入力エレメントと時刻入力エレメントを利用します。

日時入力エレメント・時刻入力エレメントには、日付・時刻型の変数を設定します。

IM-BloomMaker で選択した日時はユーザのタイムゾーンでの日時です。ユーザのタイムゾーンでの日時を IM-LogicDesigner に送信します。

IM-LogicDesigner で日時は *date* 型で扱います。

どのログインユーザが確認しても同じ日時を表示するためには、ユーザのタイムゾーンでの日時をシステムタイムゾーンでの日時に変換する必要があります。



IM-BloomMaker から日時の送信する

「[日時](#)」の「[日時を保存する](#)」に記載されている、「[IM-BloomMaker から日時の送信する](#)」と同様に設定してください。

IM-LogicDesigner での日時の受信とデータベースへの保存

日付を保存する場合と同じように、変換後に保存する必要があります。

「[日付](#)」の「[日付を保存する](#)」に記載されている、「[IM-LogicDesigner での日付の受信とデータベースへの保存](#)」と同様に設定してください。

保存されている日時を画面に表示する

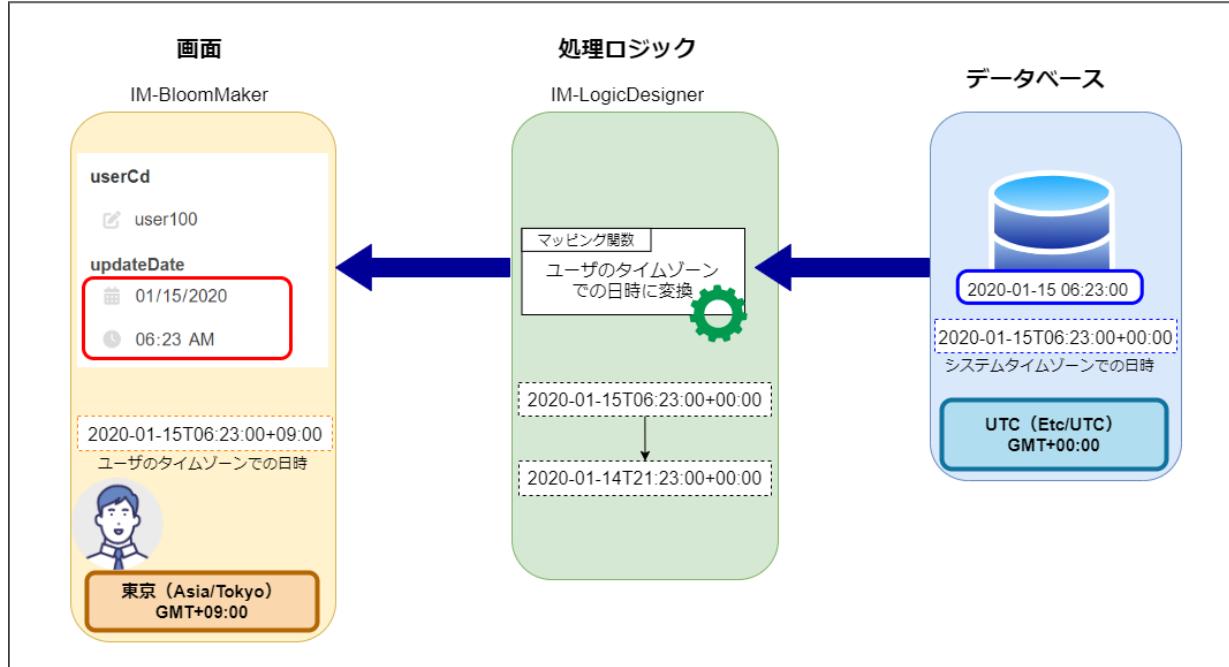
IM-LogicDesigner で日時は *date* 型で扱います。

IM-BloomMaker でどのログインユーザが確認しても同じ日時を表示するには、システムタイムゾーンでデータベースに保存されている日時をユーザのタイムゾーンでの日時に変換する必要があります。

IM-BloomMaker では日付入力エレメントと時刻入力エレメントを利用して日時を表示します。

日付入力エレメントには、日付・時刻型の変数を設定します。

IM-LogicDesigner から取得した情報を「日付・時刻」型の変数に設定することで日時を画面に表示します。



IM-LogicDesigner で保存されている日時を返却する

日付を返却する場合と同じように、変換後に返却する必要があります。

「[日付](#)」の「[保存されている日付を画面に表示する](#)」に記載されている、「[IM-LogicDesigner で保存されている日付を返却する](#)」と同様に設定してください。

IM-BloomMaker で日時を表示する

「[日時](#)」の「[保存されている日時を画面に表示する](#)」に記載されている、「[IM-BloomMaker で日時を表示する](#)」と同様に設定してください。

保存されている日時一覧を画面に一覧表示する

IM-LogicDesigner で日時は `date` 型で扱います。

IM-BloomMaker でどのログインユーザが確認しても同じ日時を表示するには、システムタイムゾーンでデータベースに保存されている日時をユーザのタイムゾーンでの日時に変換する必要があります。

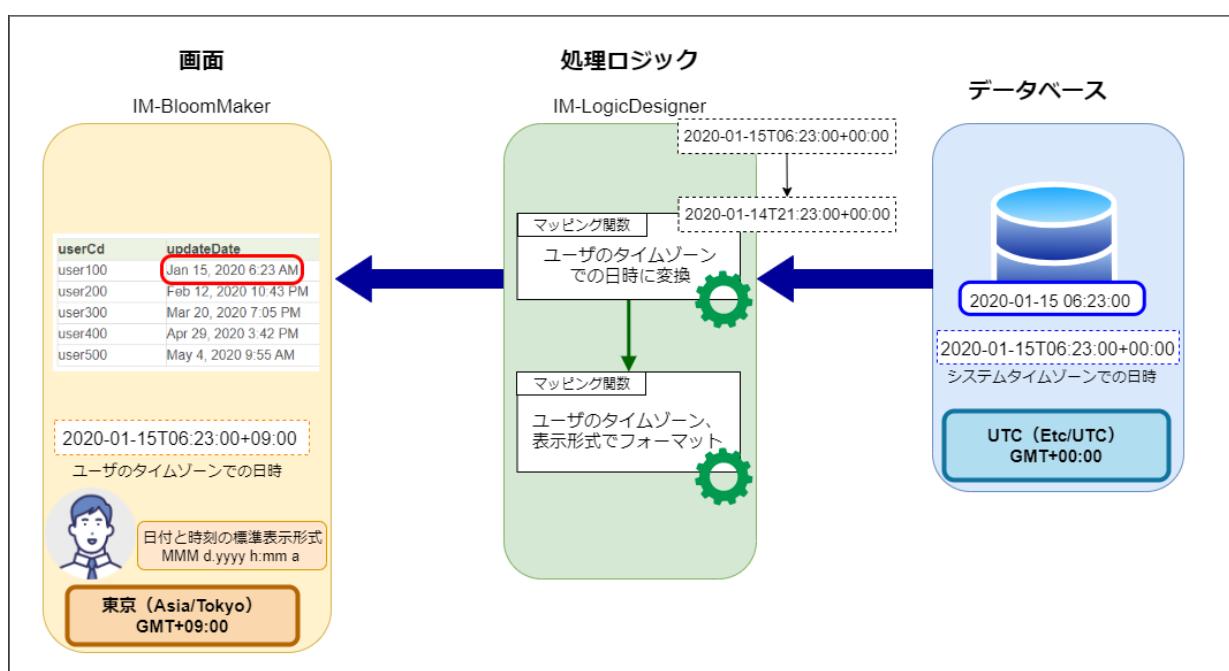
また、日時を一覧する際には日時をユーザの日時の時刻の表示形式で文字列に変換する必要があります。

データベースから取得した日時をユーザのタイムゾーンでの日時に変換した後に日時文字列に変換し、ロジックフローの出力値として設定します。

IM-BloomMaker で日時を一覧表示する画面を作成する場合、テーブル（繰り返し）エレメントを利用します。

テーブル（繰り返し）エレメントで日時を表示する列には、文字列型の変数を設定します。

IM-LogicDesigner から取得した情報を「文字列」型の変数に設定することで日時を画面に一覧表示します。



日付を返却する場合と同じように、変換後に文字列化して返却する必要があります。

「[日付](#)」の「[保存されている日付一覧を画面に一覧表示する](#)」に記載されている、「[IM-LogicDesigner で保存されている日付文字列を返却する](#)」と同様に設定してください。

ただし、日時形式文字列のフォーマットに関しては、「[日時](#)」の「[保存されている日時一覧を画面に一覧表示する](#)」に記載されている、

「[IM-LogicDesigner で保存されている日時文字列を返却する](#)」を参照してください。

IM-BloomMaker で日時一覧を表示する

「[日時](#)」の「[保存されている日時一覧を画面に一覧表示する](#)」に記載されている、「[IM-BloomMaker で日時一覧を表示する](#)」と同様に設定してください。

保存されている日時一覧をリッチテーブルに一覧表示する

IM-LogicDesigner で日時は date 型で扱います。

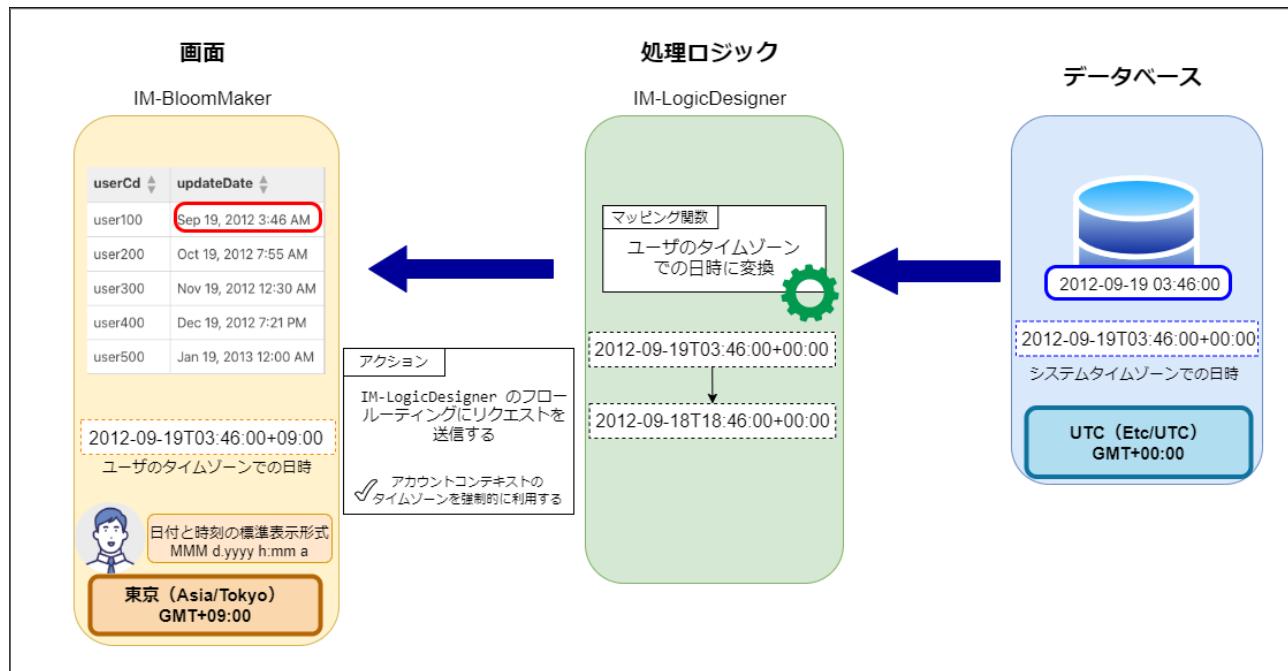
IM-BloomMaker でどのログインユーザが確認しても同じ日時を表示するには、システムタイムゾーンでデータベースに保存されている日付をユーザのタイムゾーンでの日付に変換する必要があります。（マッピング関数）

また、リッチテーブルで日時を一覧表示する際にはユーザタイムゾーンの日時で IM-BloomMaker に返却する必要があります。（アクション）

IM-BloomMaker では IM-LogicDesigner から取得した情報を「日付・時刻」型の変数に設定します。

リッチテーブルエレメントで日時を扱う場合は、columnSettingsプロパティで「type=date」を設定します。

リッチテーブルで日時を扱う場合、allowEditingプロパティをTrueにしても、日時を編集することはできません。（時刻を編集できないため、必ず日付（時刻が00:00:00）が保存されます。）



IM-LogicDesigner で保存されている日時を返却する

日付を返却する場合と同じように、変換後に返却する必要があります。

「[日付](#)」の「[保存されている日付一覧をリッチテーブルに一覧表示する](#)」に記載されている、「[IM-LogicDesigner で保存されている日付を返却する](#)」と同様に設定してください。

IM-BloomMaker で日時一覧をリッチテーブルに表示する

「[日時](#)」の「[保存されている日時一覧をリッチテーブルに一覧表示する](#)」に記載されている、「[IM-BloomMaker で日時一覧をリッチテーブルに表示する](#)」と同様に設定してください。

チュートリアル

本章では、ユーザが設定したタイムゾーンや日付と時刻の形式に従って対応するための手順を紹介します。

IM-BPM のタスクの期限日を動的に決定する

IM-BPM のタスクの期限日をユーザのタイムゾーンを考慮して動的に決定する方法については、以下を参照ください。

- 「IM-BPM チュートリアルガイド」 - 「タスククリスナを利用して、ユーザタスクの期限日時を直前ユーザタスクの期限日時から3日後に設定する」

画面の入力値を IM-BPM のタスクの期限日に設定する

作成した画面の入力値を IM-BPM の期限日に設定する方法については、以下を参照ください。

- 「IM-BPM チュートリアルガイド」 - 「IM-BloomMakerで作成した画面の入力値をもとにタスクの期限日時を設定する」

ViewCreator で日時情報を含む集計データを表示する

ViewCreator 日時情報を含む集計データ結果をユーザのタイムゾーンを考慮して表示する方法については、以下を参照ください。

- 「ViewCreator 管理者操作ガイド」 - 「IM-LogicDesigner を利用して取得したタイムスタンプ型のレコードを ViewCreator のデータ参照のリスト集計で表示する。」