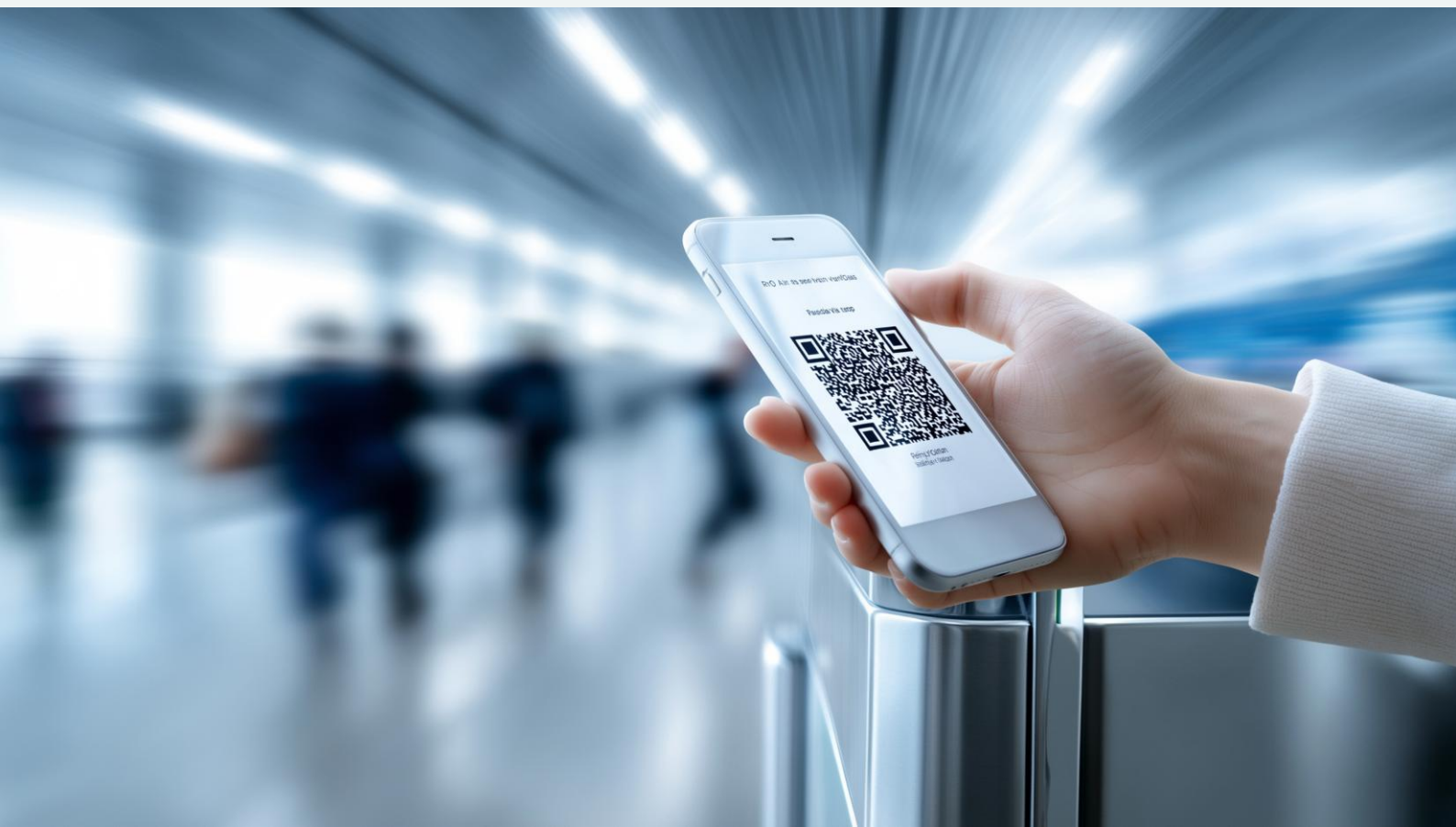


QRチケット相互運用API システム設計書



目次

1. 開発スコープ	- 1 -
1-1. 概要	- 1 -
1-2. システムを利用する業務全体像とシステム利用フロー	- 3 -
2. QR チケット相互運用 API システム：機能要件（FN/SL/AL/CO/HW/IF/UI）	- 6 -
2-1. システム機能（FN）	- 6 -
2-1-1. システムアーキテクチャ	- 6 -
2-1-2. システム機能一覧	- 10 -
2-1-3. システム機能の詳細	- 14 -
2-1-4. ソフトウェア・ライブラリ（SL）の詳細	- 43 -
2-1-5. 数理モデル・アルゴリズム（AL）の詳細	- 47 -
2-2. システムコンポーネント（CO）	- 48 -
2-2-1. システムコンポーネント図	- 48 -
2-2-2. システムコンポーネント一覧	- 49 -
2-3. ハードウェア（HW）	- 50 -
2-3-1. ハードウェアアーキテクチャ	- 50 -
2-3-2. ハードウェア一覧	- 51 -
2-3-3. ハードウェアの詳細	- 52 -
2-4. データインターフェース（IF）	- 54 -
2-4-1. データアーキテクチャ	- 54 -
2-4-2. データインターフェース一覧	- 55 -
2-4-3. データインターフェースの詳細	- 56 -
2-5. ユーザーインターフェース（UI）	- 76 -
2-5-1. 画面遷移図	- 76 -
2-5-2. ユーザーインターフェース一覧	- 77 -
2-5-3. ユーザーインターフェースの詳細	- 80 -
3. QR チケット相互運用 API システム：非機能要件（NF）	- 110 -
3-1. 非機能要件一覧	- 110 -
3-2. 非機能要件の詳細	- 111 -
4. 実証調査に利用するデータ（DT）	- 113 -
4-1. 実証調査に利用するデータ一覧	- 113 -
4-2. 実証調査に利用するデータの詳細	- 113 -
5. 用語集	- 114 -

1. 開発スコープ

1-1. 概要

本プロジェクトは、現状バラバラに存在する QR チケット認証システムの相互運用を可能とするために、各 QR チケット認証システム間で権利情報を交換するための標準的な連携インターフェースを開発することを目的としている。

具体的には、MaaS アプリ等を「出面」としたワンストップ QR チケットの運用を可能とするため、複数事業を横断する企画乗車券等を共通 QR チケットとして振り出し、これと紐づいた個別の QR チケットの発行、認証、ステータス等の情報を管理するための「仲介ハブ」システム（以下「チケット仲介 HUB ハブ」という。）のアーキテクチャと、これを利用するための API を開発する。

本書においては、「システム」とそれを運用管理する「サービサー」を明確に使い分けるため、MaaS サービサーが提供するシステム自体を「MaaS システム」又は特にバックエンドを指し示す際には「MaaS バックエンドシステム」、認証サービサーが提供するシステム自体を「認証システム」と表現する。

交通利用料金の支払いと証明及び認証の手段として、従来の紙媒体や非接触 IC カード媒体に代わり、モバイル端末等でこれを実現するデジタルチケットングが普及してきた。れい明期においては、人の目視による認証が一般的であったが、ユーザーの増加に伴い、認証精度及び量的負担の観点から限界を迎えている。

そのため、デジタル乗車券を QR 化し、鉄道の改札機やバスの車載端末で読み取らせる方式に移行しつつあるが、システムが標準化されないままサービスが乱立したことで、MaaS サービサーと認証サービサーは、各々が連携するシステムごとの個別開発が必要となっており、普及の妨げになっている。また、QR を生成・認証する認証サービサー間の連携が進んでいないため、異なる認証サービスを採用する交通事業者間をまたがる移動シーンにおいては、ユーザーは交通事業者ごとの QR を選択・表示する必要がある、ワンストップでの利用が実現できていない。

そこで、本システムでは、MaaS システムと認証システム間のチケット販売及びチケットングに係る標準化された API を提供し、サービス間の連携に要する開発工数の削減を図る。また、複数の認証システムで共通利用可能な QR を生成することで、ユーザーにワンストップな移動体験を提供することを目的とする。

システム実装に先立ち、QR チケットングにおける標準業務モデルの整理と、各認証サービサー・MaaS サービサーを相互に連携するための標準 API 仕様を策定する。標準 API 仕様及びこれを運用するための接続環境仕様（通信インフラ、セキュリティレベル）の作成に当たっては、具体的な MaaS 等のユースケースを想定したうえで、必要十分な設計とする。

チケット仲介ハブは、以下の三つの主要機能で構成される。

<チケット発行依頼振り分け機能>

MaaS システムでチケットを販売した後、ユーザーの操作等によって有効化する際、チケット認証システムでデジタルチケットングに必要なデータの生成（以下「チケット発行」という。）を行う。販売したチケット商品によってチケットを発行する認証システムが異なるため、チケット仲介ハブが MaaS システムからのチケット発行依頼を受け、商品に応じた認証システムへチケット発行依頼を振り分ける。なお、商品ごとの振り分け先については、商品発売前にチケット仲介ハブへ設定しておく。

<共通 QR 生成機能>

QR チケット相互運用 API システム設計書

MaaS システムで表示する QR をチケット仲介ハブが生成し、MaaS システムへ提供する。生成する QR は、販売済みのチケットを一意に特定するものである。QR 生成時、QR に埋め込んだ固有の ID 番号を認証システムへ連携しておくことで、認証システムが QR 読み取りデータを受け付けた際にチケット発行情報とひも付けて認証を行うことができる。

<チケット状態変化通知機能>

認証システムで QR チケットが利用され、チケットの状態が変化したことを当該チケット販売元の MaaS システム及び当該チケットを発行したほかの認証システムへ通知する。本通知を受け付けた MaaS システムと認証システムは、自システムが管理するチケットのステータスを更新する。

1-2. システムを利用する業務全体像とシステム利用フロー

1. 業務フロー

(1) チケット造成

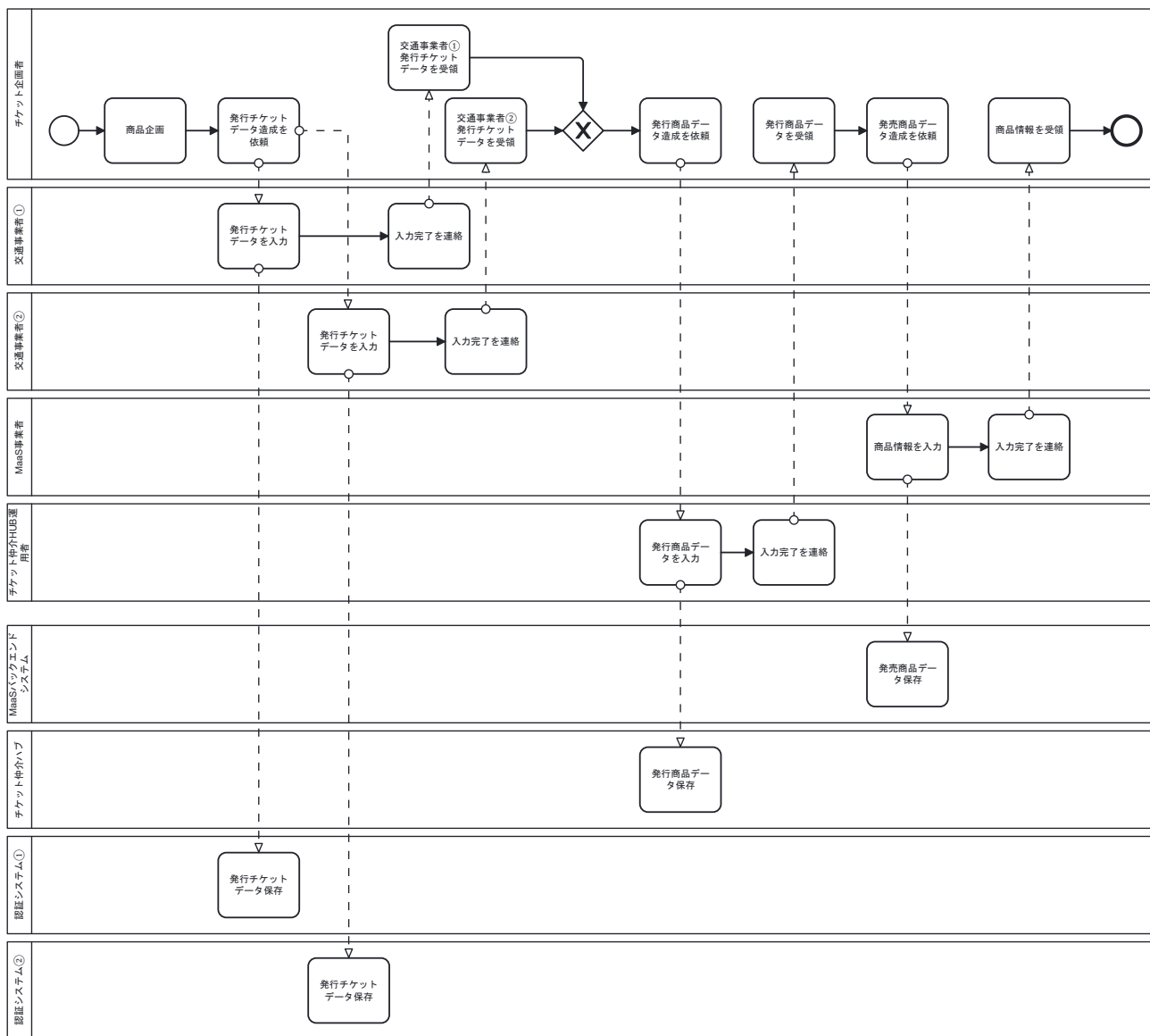


図 1-1 チケット造成業務フローの全体像

QR チケット相互運用 API システム設計書

(2) チケット発行

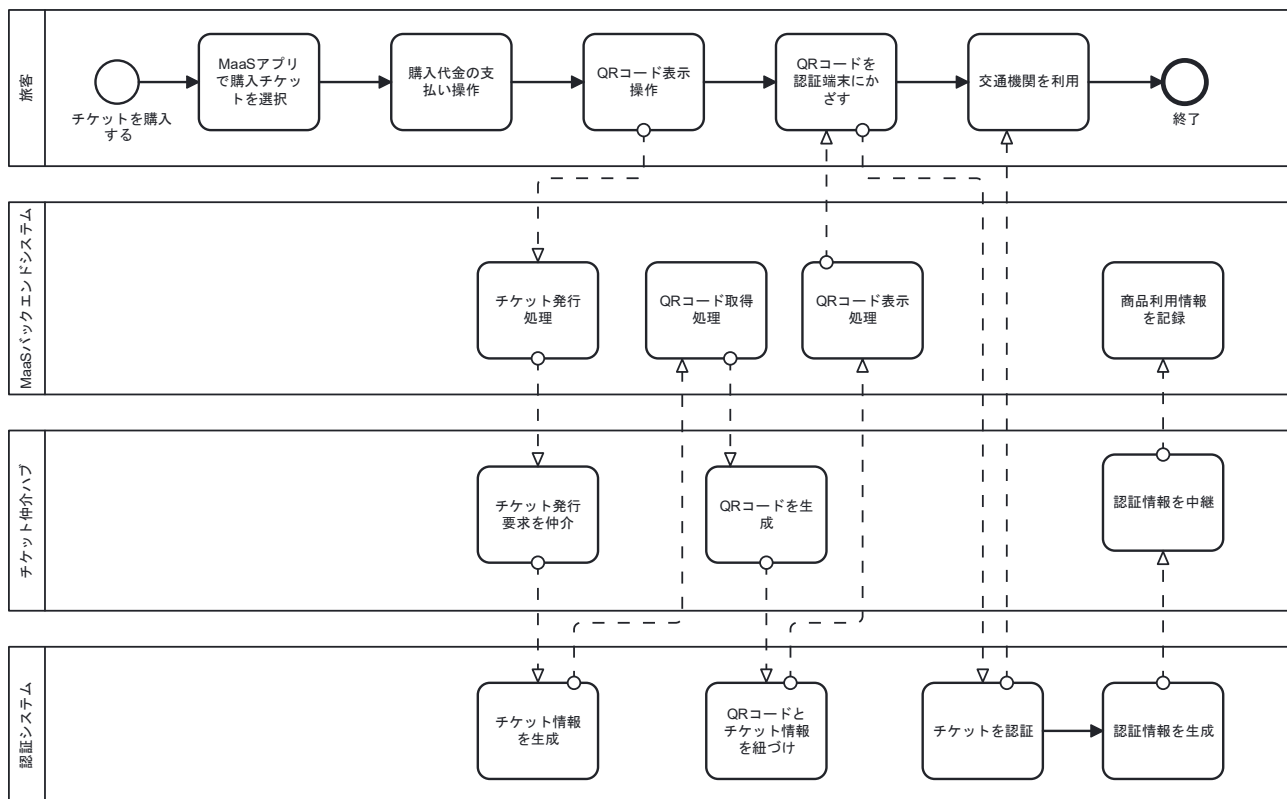


図 1-2 チケット発行業務フローの全体像

2. システムシーケンス図

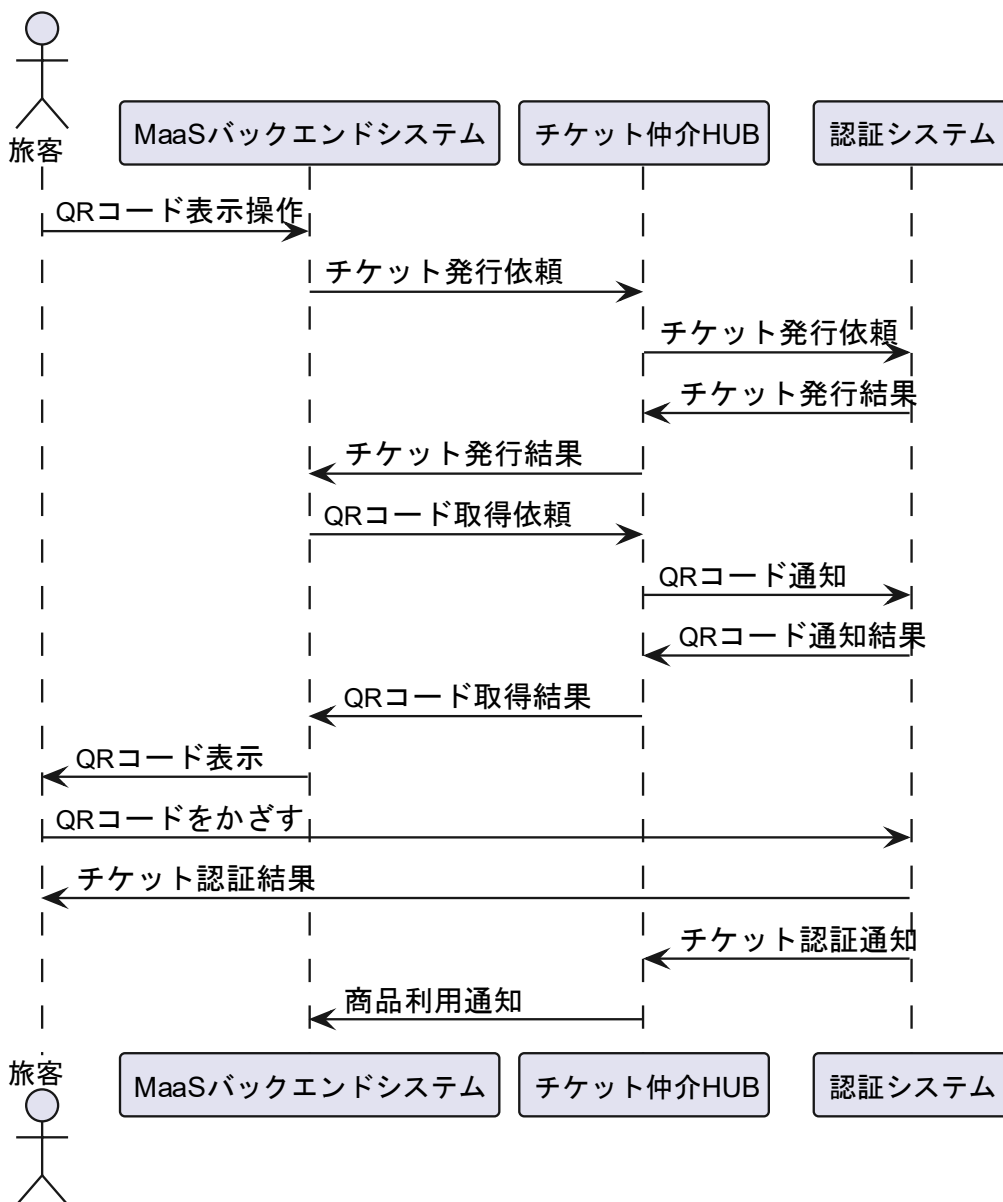


図 1-3 システムシーケンス図

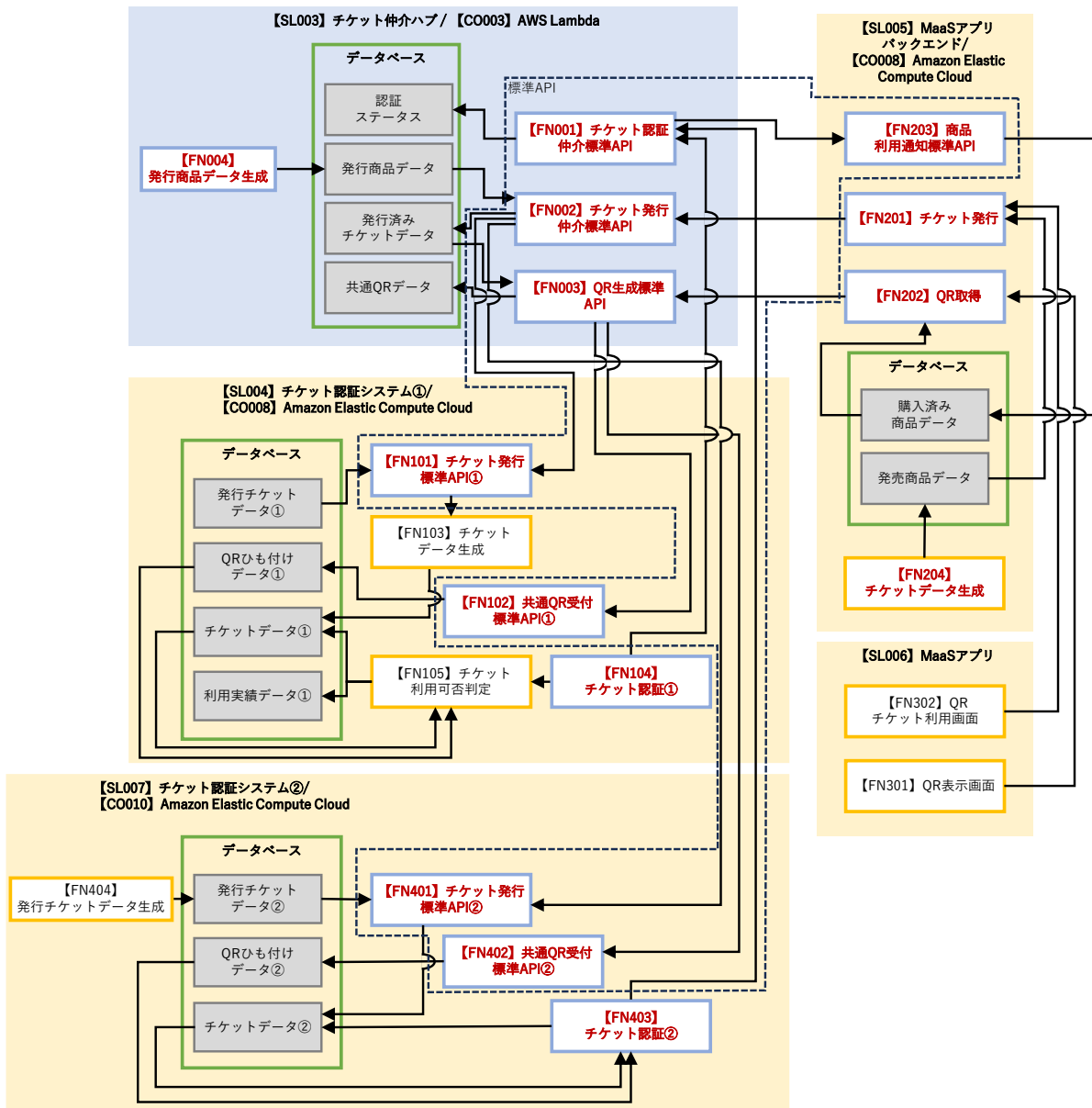
2. QR チケット相互運用 API システム：

機能要件 (FN/SL/AL/CO/HW/IF/UI)

2-1. システム機能 (FN)

2-1-1. システムアーキテクチャ

システムアーキテクチャの全体像を示す。各ブロックの関係を矢印で示す。



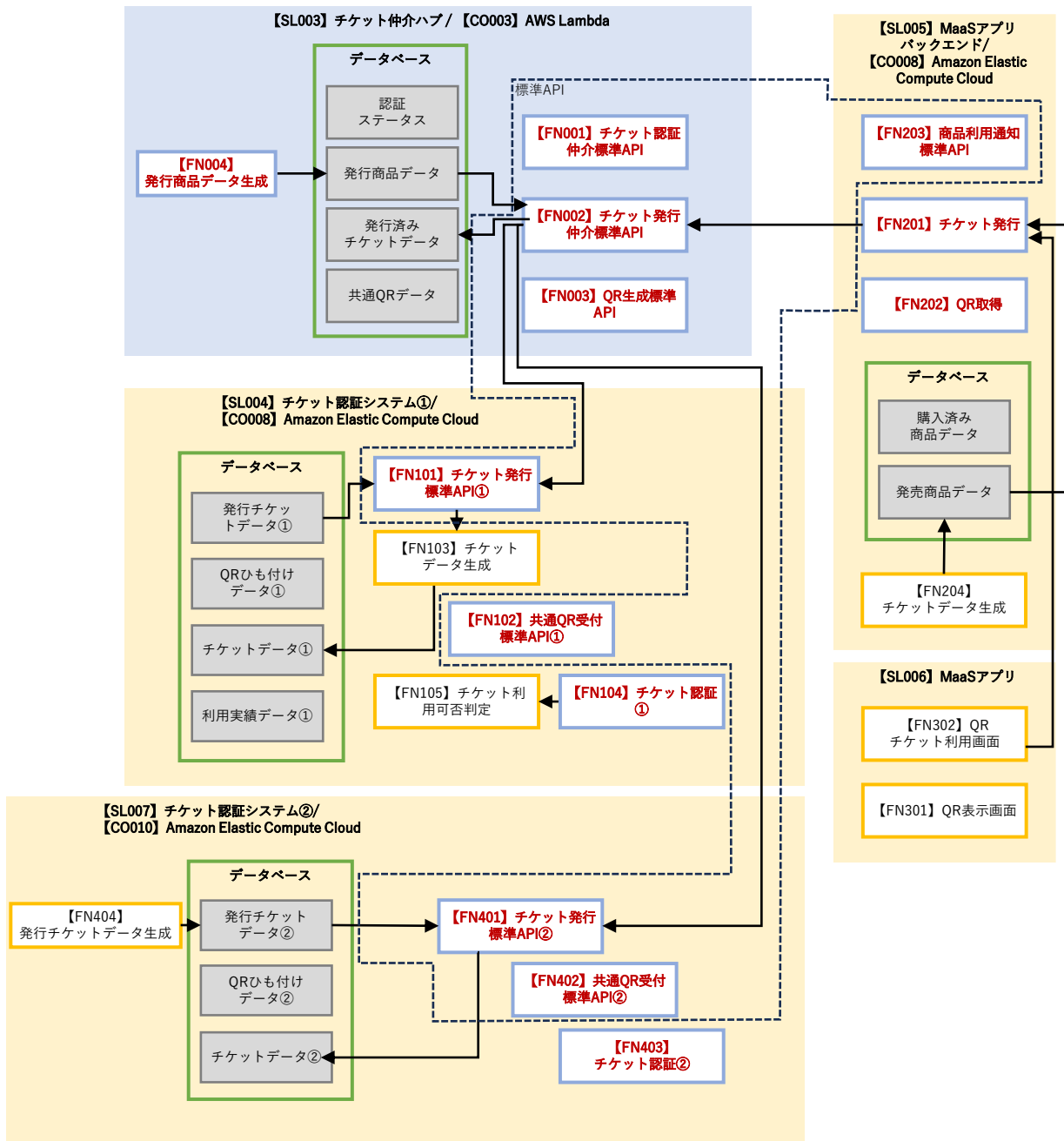
凡例



図 2-1 システムアーキテクチャ

QR チケット相互運用 API システム設計書

MaaS アプリにおいてチケット利用開始の操作が行われ、チケット発行するときの各機能の関係を示す。



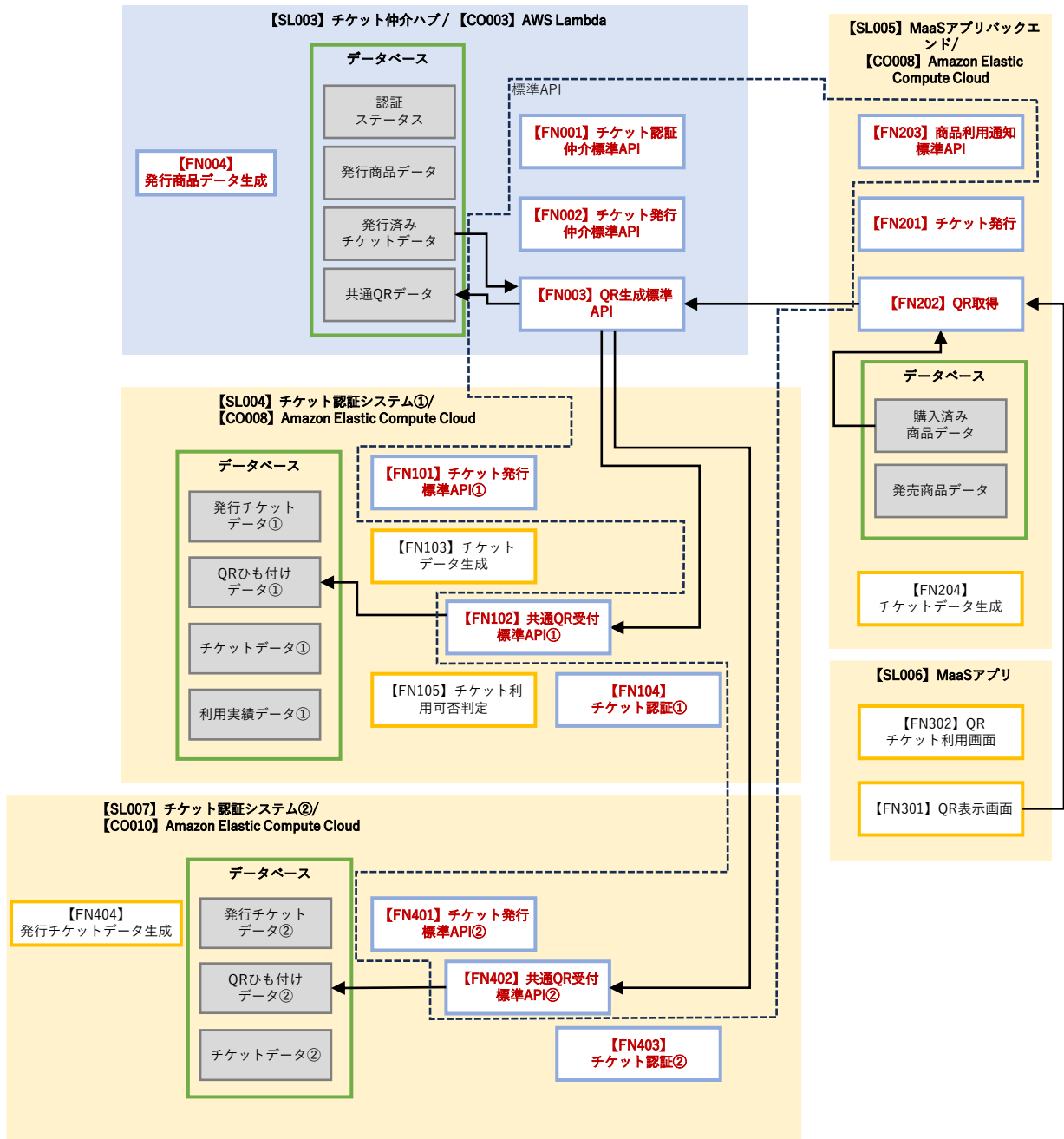
凡例



図 2-2 システムアーキテクチャ (チケット発行)

QR チケット相互運用 API システム設計書

MaaS アプリにおいて QR を表示操作が行われチケット仲介ハブにおいて QR 生成するときの各機能の関係を示す。



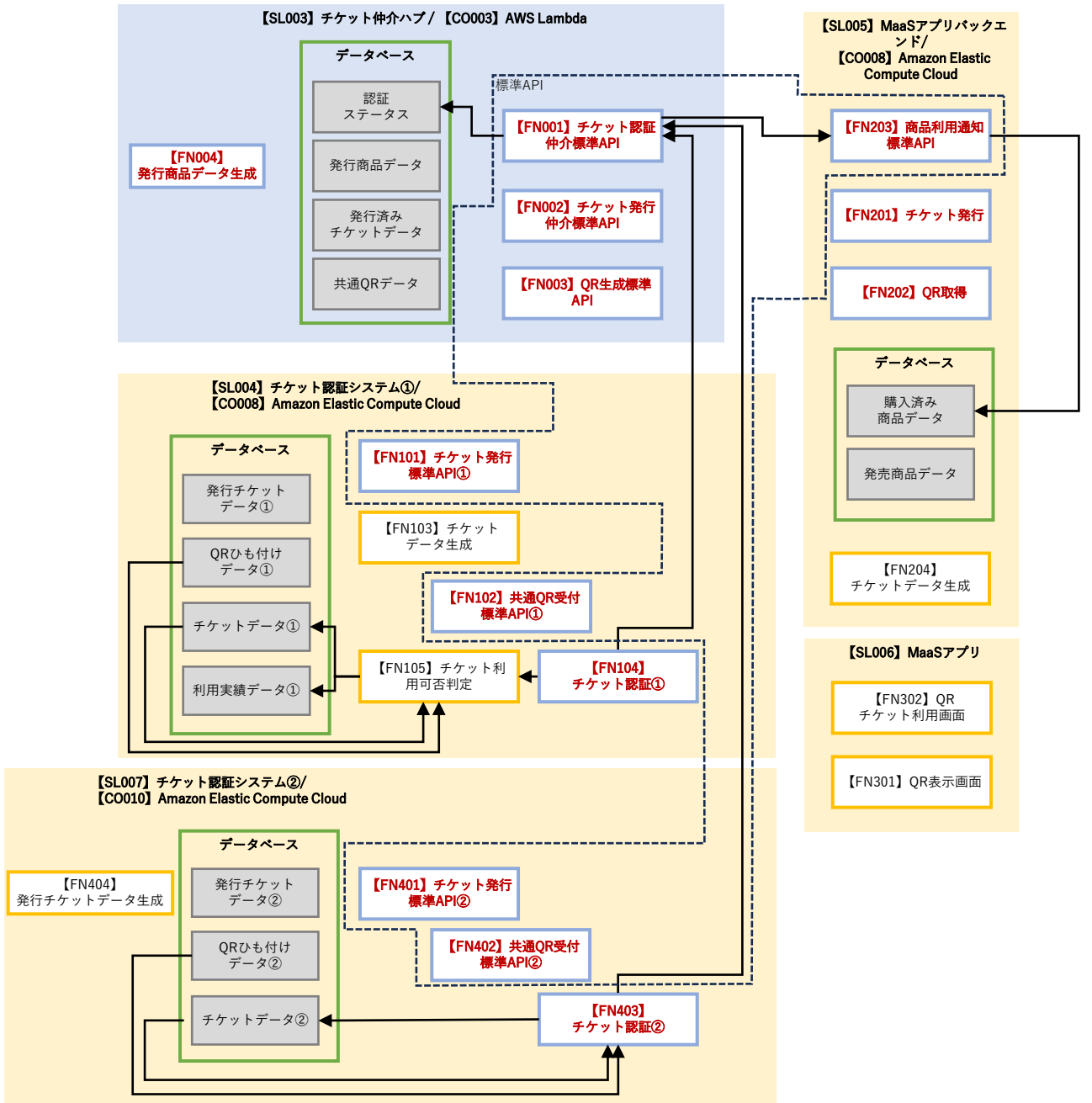
凡例



図 2-3 システムアーキテクチャ (QR 生成)

QR チケット相互運用 API システム設計書

MaaS アプリにおいて表示された QR を認証装置により認証したときの各機能の関係を示す。



凡例



図 2-4 システムアーキテクチャ (チケット認証)

2-1-2. システム機能一覧

表 2-1 チケット仲介ハブ

※朱文字：新規開発・既存改修

ID	機能名	機能説明
FN001	チケット認証仲介標準 API	<ul style="list-style-type: none"> ● QR チケットが改札機や車載決済端末で認証されたとき、チケット認証情報を MaaS システムへ通知し、MaaS システムにおけるチケットのステータス情報の更新を可能とする ● API に入力されたチケット認証情報を、チケット仲介ハブのデータベースに保存する
FN002	チケット発行仲介標準 API	<ul style="list-style-type: none"> ● MaaS アプリにおいてチケット利用開始の操作が行われたとき、MaaS アプリバックエンドから API が呼び出され、チケット認証システムでチケットが利用できるようにするためにチケット発行標準 API を呼出す ● API 入力データのチケット発行情報と、データベースから取得した発行商品データから、当該チケットが利用されるチケット認証サービスのチケット発行標準 API を呼出す
FN003	QR 生成標準 API	<ul style="list-style-type: none"> ● MaaS アプリバックエンドから API が呼び出され、MaaS アプリにおいて QR を表示するとき、QR を生成する ● API 入力データの共通 QR 生成情報から、チケットが利用される認証サービスを特定し、認証サービスへ QR データを通知する
FN004	発行商品データ生成	<ul style="list-style-type: none"> ● 発行商品データを生成し、データベースへ保存する

表 2-2 チケット認証システム①（日本信号）

ID	機能名	機能説明
FN101	チケット発行標準 API①	<ul style="list-style-type: none"> ● チケット仲介ハブにおいて生成された QR と共通チケット ID を、チケット認証システム①が受け取る ● API 入力データのチケット発行情報に含まれる商品 ID をキーに、発行チケットデータから当該チケットの効力の情報を取得し、チケットデータを生成する
FN102	共通 QR 受付標準 API①	<ul style="list-style-type: none"> ● チケット仲介ハブにおいて QR が生成されたとき、QR と共通チケット ID が通知される ● チケットの認証において QR のデータが入力されたとき、チケットを判定する際に必要となるチケットデータを取得するため、QR と共通チケット ID をひも付けデータを保存する
FN103	チケットデータ生成	<ul style="list-style-type: none"> ● ユーザーが提示した QR を改札機が読み取り、チケット認証システムでチケットの利用可否を判定するときに必要となる、チケットの効力と状態を記録したデータ（有効期間、有効区間、入出場状態など）を生成する
FN104	チケット認証①	<ul style="list-style-type: none"> ● 改札機に QR がかざされたとき、改札機が生成した QR を含むチケット認証受付データを受信し、チケット利用可否判定機能によりチケットの判定を実行する ● チケットの判定結果からチケット認証情報を生成し、チケット認証仲介標準 API を呼出すことにより、チケット仲介ハブを通して MaaS アプリバックエンドへ通知する
FN105	チケット利用可否判定	<ul style="list-style-type: none"> ● 共通 QR 受付標準 API により生成された QR ひも付けデータから、判定するチケットのチケットデータを取得し、チケットの利用可否を判定する ● 利用可否の判定結果からチケットデータのチケットステータスを更新し、データベースへ保存する ● 利用可否の判定結果から利用実績データを生成し、データベースへ保存する

表 2-3 MaaS アプリ(my route)バックエンド

※朱文字：新規開発・既存改修

ID	機能名	機能説明
FN201	チケット発行	<ul style="list-style-type: none"> ● MaaS アプリの QR チケット利用画面においてチケットの利用を開始するとき、チケット仲介ハブのチケット発行仲介標準 API を呼出し、チケット認証システムにおいてチケットが利用可能な状態にする ● 発売商品データの発行するチケットの共通商品 ID、有効期間、発行数から発行情報を生成し、チケット発行仲介標準 API 呼出し時に API 入力データとして設定する
FN202	QR 取得	<ul style="list-style-type: none"> ● MaaS アプリの QR 表示画面においてチケットの QR を表示するとき、チケット仲介ハブの QR 生成標準 API を呼出し、QR を取得する ● 購入済み商品データの共通チケット ID から QR 生成情報を生成し、QR 生成標準 API 呼出し時に API 入力データとして設定する
FN203	商品利用通知標準 API	<ul style="list-style-type: none"> ● チケット認証システムによってチケットが認証されたとき、チケット認証情報を受信し、購入済み商品データのチケットステータスの更新を行う
FN204	チケットデータ生成	<ul style="list-style-type: none"> ● 発売商品データの登録を行う。商品に含まれるチケットの共通商品 ID や有効期間、同時利用枚数などの情報を設定する。

表 2-4 MaaS アプリ(my route)

※朱文字：新規開発・既存改修

ID	機能名	機能説明
FN301	QR 表示画面	<ul style="list-style-type: none"> ● 購入したチケットを利用するために MaaS アプリをユーザーが操作することにより、QR を表示する
FN302	QR チケット利用画面	<ul style="list-style-type: none"> ● MaaS アプリにおいて購入済みのチケットを一覧表示し、ユーザーの選択操作を受け付ける ● ユーザーがチケットを選択したチケットに対して、認証装置でチケットを利用可能にするためにチケット発行呼出機能を実行する

表 2-5 チケット認証システム② (レシップ)

※朱文字：新規開発・既存改修

ID	機能名	機能説明
FN401	チケット発行標準 API②	<ul style="list-style-type: none"> ● チケット仲介ハブから呼び出され、チケットデータを生成し、データベースへ保存する ● API入力データのチケット発行情報に含まれる商品IDをキーに、発行チケットデータから当該チケットの効力の情報を取得し、チケットデータを生成する
FN402	共通 QR 受付標準 API②	<ul style="list-style-type: none"> ● チケット仲介ハブにおいて QR が生成されたとき、QR と共通チケットIDが通知される ● チケットの認証において QR のデータが入力されたとき、車載決済端末に配信するホワイトリストデータとして QR ひも付けデータを生成し、保存する
FN403	チケット認証②	<ul style="list-style-type: none"> ● 車載決済端末に QR がかざされたとき、ホワイトリストをサーチしてチケットの利用可否を判定する ● チケットの判定結果からチケット認証情報を生成し、チケット認証仲介標準 API を呼出すことにより、チケット仲介ハブを通して MaaS アプリバックエンドへ通知する
FN404	発行チケットデータ生成	<ul style="list-style-type: none"> ● 発行チケットデータを生成し、データベースへ保存する

2-1-3. システム機能の詳細

開発機能の詳細要件を記す。なお、本業務において開発や改修を行う内容については、**朱文字**で示す。

【FN001】チケット認証仲介標準 API <新規開発>

- 本システム機能の概要
 - QR チケットが改札機や車載決済端末で認証されたとき、チケット認証情報を MaaS システムへ通知し、MaaS システムにおけるチケットのステータス情報の更新を可能とする
 - API に入力されたチケット認証情報を、チケット仲介ハブのデータベースに保存する
 - チケット認証情報は、チケット認証システムによって生成されるデータであり、QR が利用された日時、場所、端末の種類、チケットを識別する ID、認証の種別（入場、出場等）が含まれる
- 本システム機能の入力・処理・出力のフローチャート

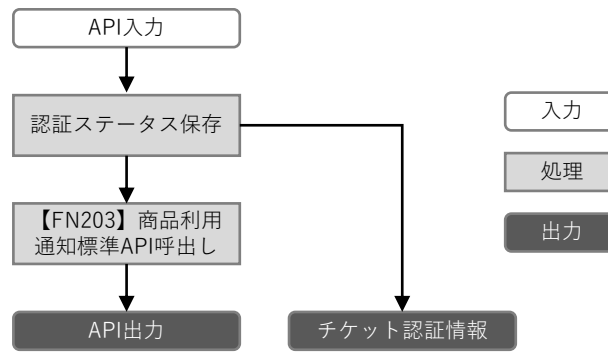


図 2-5 【FN001】のフローチャート

- 本システム機能の処理の詳細
 - 認証ステータス保存
 - ◇ 処理内容
 - API 入力されたチケット認証情報をデータベースに保存する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 【FN203】商品利用通知標準 API 呼出し
 - ◇ 処理内容
 - MaaS システムの商品利用通知標準 API を呼出し、認証情報を通知する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力

◇ API 入力

- データの内容
 - チケット認証情報
 - QR が利用された日時、場所、端末の種類、チケットを識別する ID、認証の種別（入場、出場等）
- データの形式
 - JSON
- 利用するデータインターフェース
 - 【IF001】 チケット認証仲介標準 API 呼出し用の標準 API

➢ 出力

◇ API 出力

- データの内容
 - 処理結果
- データの形式
 - JSON
- 利用するデータインターフェース
 - 【IF001】 チケット認証仲介標準 API 呼出し用の標準 API

◇ チケット認証情報

- データの内容
 - チケット認証情報
 - QR が利用された日時、場所、端末の種類、チケットを識別する ID、認証の種別（入場、出場等）
- データの形式
 - JSON
- 利用するデータインターフェース
 - 【IF006】 商品利用通知標準 API 呼出し用の標準 API
 - チケット仲介ハブ ER 図

【FN002】 チケット発行仲介標準 API <新規開発>

- 本システム機能の概要
 - MaaS アプリにおいてチケット利用開始の操作が行われたとき、MaaS アプリバックエンドから API が呼び出され、チケット認証システムでチケットが利用できるようにするためにチケット発行標準 API を実行する
 - API 入力データのチケット発行情報と、データベースから取得した発行商品データから、当該チケットが利用されるチケット認証サービスのチケット発行標準 API を呼出す
 - 発行したチケットの情報を記録するために発行済みチケットデータを保存する
- 本システム機能の入力・処理・出力のフローチャート

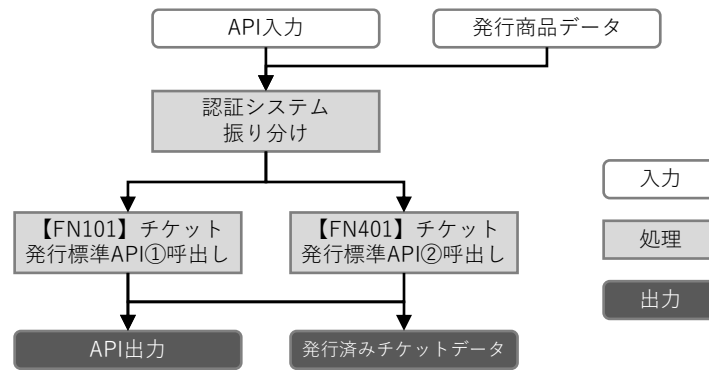


図 2-6 【FN002】のフローチャート

- 本システム機能の処理の詳細
 - 認証システム振り分け
 - ◇ 処理内容
 - API 入力データのチケット発行情報と、データベースから取得した発行商品データから、当該チケットが利用されるチケット認証サービスを判別する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 【FN101】チケット発行標準 API①呼出し
 - ◇ 処理内容
 - 【FN101】チケット発行標準 API①を呼出し、チケット認証システムにおいてチケットを発行する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 【FN401】チケット発行標準 API②呼出し
 - ◇ 処理内容
 - 【FN401】チケット発行標準 API②を呼出し、チケット認証システムにおいてチケットを発行する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ API 入力
 - データの内容

QR チケット相互運用 API システム設計書

- チケット発行情報
- 発行するチケットの共通商品 ID、有効期間、数量
- データの形式
 - JSON
- 利用するデータインターフェース
 - 【IF002】チケット発行仲介標準 API 呼出し用の標準 API
- ◇ 発行商品データ
 - データの内容
 - MaaS システムから通知される共通商品 ID に対して、認証システム ID と認証システムが商品を識別するための認証システム商品 ID をひも付けるデータ
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF026】発行商品データ取得
 - チケット仲介ハブ ER 図
- 出力
 - ◇ API 出力
 - データの内容
 - 処理結果
 - 共通チケット ID
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF002】チケット発行仲介標準 API 呼出し用の標準 API
 - ◇ 発行済みチケットデータ
 - データの内容
 - 発行したチケットを特定するための共通チケット ID と、認証システム ID、共通商品 ID、チケット発行仲介標準 API 呼出し元の MaaS システム ID をひも付けるデータ
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF027】発行済みチケットデータ保存
 - チケット仲介ハブ ER 図

【FN003】QR 生成標準 API <新規開発>

- 本システム機能の概要
 - MaaS アプリにおいて QR を表示するとき、MaaS アプリバックエンドから API が呼び出され、QR を生成する

QR チケット相互運用 API システム設計書

- API 入力データの共通 QR 生成情報から、チケットが利用される認証サービスを特定し、認証サービスへ QR データを通知する

- 本システム機能の入力・処理・出力のフローチャート

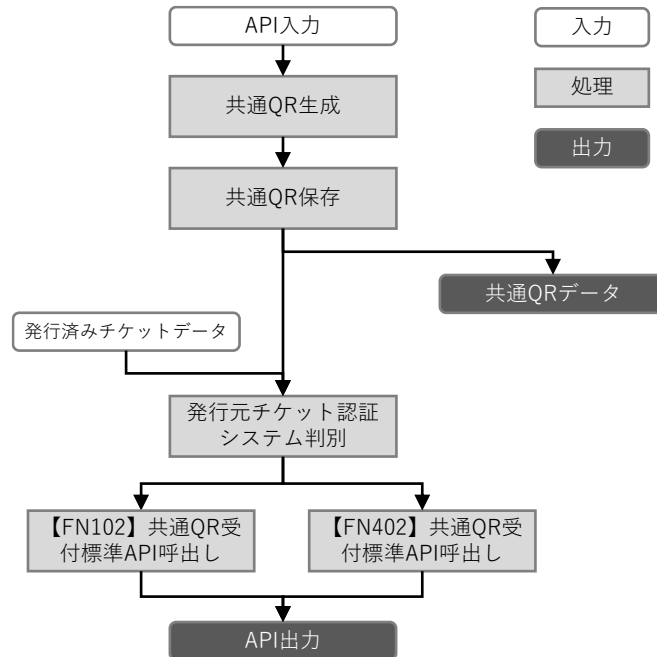


図 2-7 【FN003】のフローチャート

- 本システム機能の処理の詳細

- 共通 QR 生成

- ◇ 処理内容

- 共通 QR データを生成する
- 【AL001】TOTP ベースアルゴリズムによる QR 改ざん防止と有効期限のコード生成

- ◇ 利用するライブラリ

- -

- ◇ 利用するアルゴリズム

- 【AL001】TOTP ベースアルゴリズム

- 共通 QR 保存

- ◇ 処理内容

- 生成した共通 QR データを保存する

- ◇ 利用するライブラリ

- -

- ◇ 利用するアルゴリズム

- -

- 発行元チケット認証システム判別

- ◇ 処理内容

- QR を生成する対象チケットの発行元となるチケット認証システムを判別する

- ◇ 利用するライブラリ

QR チケット相互運用 API システム設計書

- -
- ◇ 利用するアルゴリズム
 - -
- 【FN102】 共通 QR 受付標準 API①呼出し
 - ◇ 処理内容
 - 【FN102】 共通 QR 受付標準 API①を呼出し、チケット認証システムへ QR を通知する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- 【FN402】 共通 QR 受付標準 API②呼出し
 - ◇ 処理内容
 - 【FN402】 共通 QR 受付標準 API②を呼出し、チケット認証システムへ QR を通知する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ API 入力
 - データの内容
 - 共通 QR 生成情報
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF003】 QR 生成標準 API 呼出し用の標準 API
 - ◇ 発行済みチケットデータ
 - データの内容
 - チケット発行仲介標準 API によってチケットを発行したときに保存した共通チケット ID と、認証システム ID、共通商品 ID、チケット発行仲介標準 API 呼出し元の MaaS システム ID をひも付けるデータ
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF028】 発行済みチケットデータ取得
 - チケット仲介ハブ ER 図
 - 出力
 - ◇ API 出力

QR チケット相互運用 API システム設計書

- データの内容
 - API で生成した共通 QR データ
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF003】 QR 生成標準 API 呼出し用の標準 API
- ◇ 共通 QR データ
- データの内容
 - 共通 QR データ
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF010】 共通 QR データ保存
 - チケット仲介ハブ ER 図

【FN004】 発行商品データ生成 <新規開発>

- 本システム機能の概要
 - チケット仲介ハブにおいて、発行商品データの設定用 UI を提供する
 - UI から設定された情報から発行商品データを生成する
- 本システム機能の入力・処理・出力のフローチャート

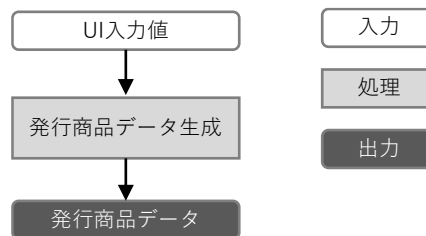


図 2-8 【FN004】 のフローチャート

- 本システム機能の処理の詳細
 - 発行商品データ生成
 - ◇ 処理内容
 - UI 入力値をもとに、発行商品データを生成する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ UI 入力値
 - データの内容

- 発行商品データ設定 UI において設定されたデータ
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - -
- 出力
- ◇ 発行商品データ
 - データの内容
 - チケット発行時に認証システムと認証システムにおける商品 ID を特定するためのデータ
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - -

【FN101】 チケット発行標準 API①<新規開発>

- 本システム機能の概要
 - チケット仲介ハブから呼び出され、チケットデータを生成し、チケット認証システム①のデータベースへ保存する
 - API 入力データのチケット発行情報に含まれる商品 ID をキーに、発行チケットデータから当該チケットの効力の情報を取得し、チケットデータを生成する
- 本システム機能の入力・処理・出力のフローチャート

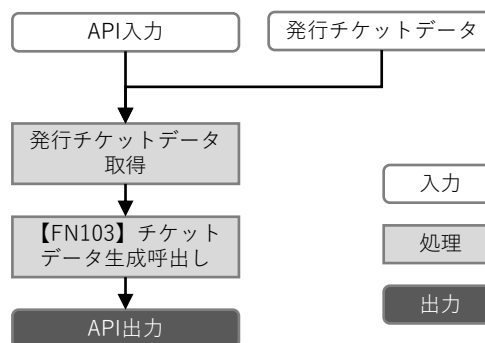


図 2-9 【FN101】 のフローチャート

- 本システム機能の処理の詳細
 - 発行チケットデータ取得
 - ◇ 処理内容
 - 発行チケットデータをデータベースから取得する
 - ◇ 利用するライブラリ
 - -

QR チケット相互運用 API システム設計書

- ◇ 利用するアルゴリズム
 - -
- 【FN103】 チケットデータ生成呼出し
 - ◇ 処理内容
 - 【FN103】 チケットデータ生成を呼出し、チケットデータを生成する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ API 入力
 - データの内容
 - チケット発行情報
 - 発行するチケットの認証システム商品 ID、有効期間、数量
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF004】 チケット発行標準 API 呼出し用の標準 API
 - ◇ 発行チケットデータ
 - データの内容
 - 発行チケットデータ
 - 認証システム商品 ID に紐づくチケットの効力データ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF011】 発行チケットデータ取得
 - 出力
 - ◇ API 出力
 - データの内容
 - 処理結果
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF004】 チケット発行標準 API 呼出し用の標準 API

【FN102】 共通 QR 受付標準 API① <新規開発>

- 本システム機能の概要
 - チケット仲介ハブにおいて生成された QR と共通チケット ID を、チケット認証システム①が受け取るための API
 - チケットの認証において QR のデータが入力されたとき、チケットを判定する際に必要となるチケットデータを取得するため、QR と共通チケット ID のひも付けデータを保存する
- 本システム機能の入力・処理・出力のフローチャート

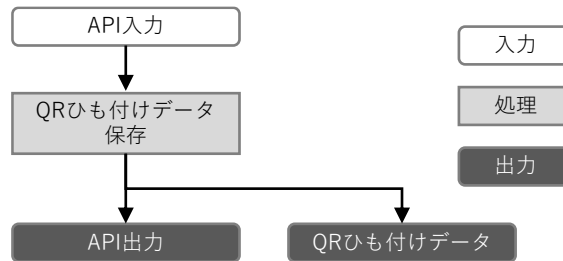


図 2-10 【FN102】のフローチャート

- 本システム機能の処理の詳細
 - QR ひも付けデータ保存
 - ◇ 処理内容
 - 受信した共通 QR データから QR ひも付けデータを保存する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ API 入力
 - データの内容
 - 共通 QR 通知
 - チケット仲介ハブで生成された QR データと、共通チケット ID、QR の有効期限
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF005】 共通 QR 受付標準 API 呼出し用の標準 API
 - 出力
 - ◇ API 出力
 - データの内容
 - 処理結果
 - データの形式
 - JSON

- 利用するデータインターフェース
 - 【IF005】 共通 QR 受付標準 API 呼出し用の標準 API
- ◇ QR ひも付けデータ
 - データの内容
 - QR ひも付けデータ
 - 共通チケット ID と QR データ、QR の有効期限
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF012】 QR ひも付けデータ保存

【FN103】 チケットデータ生成

- 本システム機能の概要
 - ユーザーが提示した QR を改札機が読み取り、チケット認証システムでチケットの利用可否を判定するときに必要となる、チケットの効力と状態を記録したデータ（有効期間、有効区間、入出場状態など）を生成する
- 本システム機能の入力・処理・出力のフローチャート

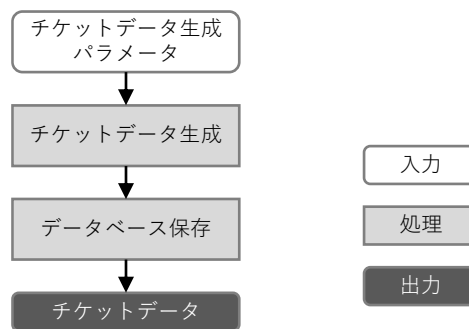


図 2-11 【FN103】 のフローチャート

- 本システム機能の処理の詳細
 - チケットデータ生成
 - ◇ 処理内容
 - チケットデータを生成する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - データベース保存
 - ◇ 処理内容
 - 生成したチケットデータをデータベースへ保存する
 - ◇ 利用するライブラリ
 - -

◇ 利用するアルゴリズム

- -

● データ仕様

➤ 入力

◇ チケットデータ生成パラメータ

- データの内容
 - チケットデータ生成パラメータ
- データの形式
 - 関数の引数
- 利用するデータインターフェース
 - 【IF007】 チケットデータ生成呼出し IF

➤ 出力

◇ チケットデータ

- データの内容
 - チケットデータ
- データの形式
 - SQL
- 利用するデータインターフェース
 - 【FN014】 チケットデータ保存

【FN104】 チケット認証①<新規開発>

- 本システム機能の概要
 - 改札機に QR がかざされたとき、改札機が生成した QR を含むチケット認証受付データを受信し、チケット利用可否判定機能によりチケットの判定を実行する
 - チケットの判定結果からチケット認証情報を生成し、チケット認証仲介標準 API を呼出すことにより、チケット仲介ハブを通して MaaS アプリバックエンドへ通知する
- 本システム機能の入力・処理・出力のフローチャート

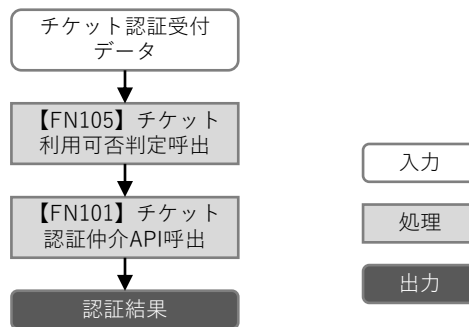


図 2-12 【FN104】 のフローチャート

- 本システム機能の処理の詳細
 - 【FN105】 チケット利用可否判定呼出し
 - ◇ 処理内容

- 【FN105】 チケット利用可否判定を呼出し、チケットの利用可否判定処理を行い認証する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 【FN001】 チケット認証仲介標準 API 呼出し
 - ◇ 処理内容
 - 【FN001】 チケット認証仲介標準 API を呼出し、他システムへ認証結果を通知する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - データ仕様
 - 入力
 - ◇ チケット認証受付データ
 - データの内容
 - チケット認証受付データ
 - データの形式
 - バイナリ
 - 利用するデータインターフェース
 - 端末 ID、業務コード、QR から構成されるデータ
 - 出力
 - ◇ 認証結果
 - データの内容
 - 認証結果
 - データの形式
 - バイナリ
 - 利用するデータインターフェース
 - チケット認証の結果データ
- 【FN105】 チケット利用可否判定
- 本システム機能の概要
 - 共通 QR 受付標準 API により生成された QR ひも付けデータから、判定するチケットのチケットデータを取得し、チケットの利用可否を判定する
 - 利用可否の判定結果からチケットデータのチケットステータスを更新し、データベースへ保存する
 - 利用可否の判定結果から利用実績データを生成し、データベースへ保存する
 - 本システム機能の入力・処理・出力のフローチャート

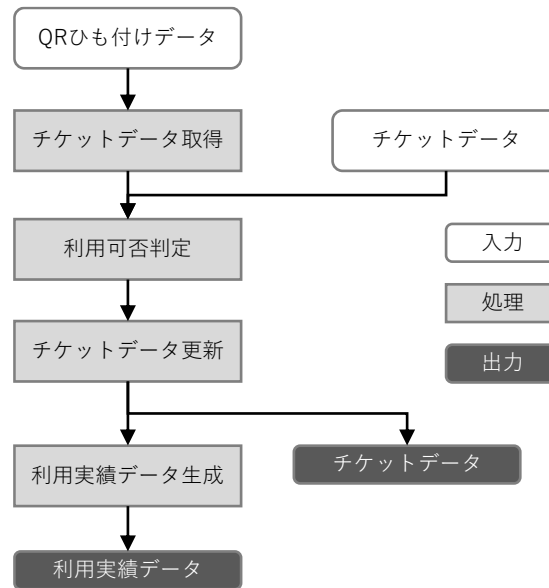


図 2-13 【FN105】のフローチャート

- 本システム機能の処理の詳細
 - チケットデータ取得
 - ◇ 処理内容
 - QR にひも付くチケット ID を特定し、チケットデータをデータベースから取得する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 利用可否判定
 - ◇ 処理内容
 - チケットデータのデータを判定し、チケットの利用可否を判定する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - チケットデータ更新
 - ◇ 処理内容
 - 利用可否判定の結果をチケットデータのステータスに反映する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 利用実績データ生成
 - ◇ 処理内容

- 利用実績データを生成する
- ◇ 利用するライブラリ
 - -
- ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ QR ひも付けデータ
 - データの内容
 - QR ひも付けデータ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF013】 QR ひも付けデータ取得
 - ◇ チケットデータ
 - データの内容
 - チケットデータ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF015】 チケットデータ取得
 - 出力
 - ◇ チケットデータ
 - データの内容
 - チケットデータ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF014】 チケットデータ保存
 - ◇ 利用実績データ
 - データの内容
 - 利用実績データ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF016】 利用実績データ保存

【FN201】 チケット発行<改修機能>

- 本システム機能の概要
 - MaaS アプリの QR チケット利用画面においてチケットの利用を開始するとき、チケット仲介ハブのチケット発行仲介標準 API を呼出し、チケット認証システムにおいてチケットが利用可能な状態にする
 - 発売商品データのチケットの共通商品 ID、有効期間、発行数から発行情報を生成し、チケット発行仲介標準 API 呼出し時に API 入力データとして設定する
 - 呼出す API を、チケット認証システムのチケット発行標準 API からチケット仲介ハブのチケット発行仲介標準 API を呼出すように改修する

既存機能の改修内容

チケット仲介ハブと接続してチケット発行するために、既存機能ではチケット認証システムのチケット発行 API を呼出していた処理を、【FN002】チケット発行仲介標準 API を呼出すように改修する

- 本システム機能の入力・処理・出力のフローチャート

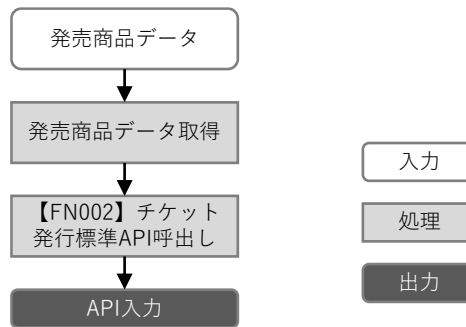


図 2-14 【FN201】のフローチャート

- 本システム機能の処理の詳細
 - 発売商品データ取得
 - ◇ 処理内容
 - 発売商品データをデータベースから取得
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 【FN002】チケット発行標準 API 呼出し
 - ◇ 処理内容
 - 【FN002】チケット発行仲介標準 API の入力データを生成し、API を呼出す
 - 【FN002】チケット発行仲介標準 API の入力データとしてチケットの共通商品 ID、有効期間、発行数から発行情報を生成する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム

- -
- データ仕様
 - 入力
 - ◇ 発売商品データ
 - データの内容
 - 発売商品データ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF019】 発売商品データ取得
 - 出力
 - ◇ API 入力
 - データの内容
 - チケット発行情報
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF002】 チケット発行仲介標準 API 呼出し用の標準 API

【FN202】 QR 取得 <改修機能>

- 本システム機能の概要
 - MaaS アプリの QR 表示画面においてチケットの QR を表示するとき、チケット仲介ハブの QR 生成標準 API を呼出し、QR を取得する
 - 購入済み商品データの共通チケット ID から QR 生成情報を生成し、QR 生成標準 API 呼出し時に API 入力データとして設定する
 - 呼出す API をチケット認証システムの QR 生成標準 API からチケット仲介ハブの QR 生成標準 API を呼出すように改修する

既存機能の改修内容

チケット仲介ハブから QR を取得するために、既存機能ではチケット認証システムの QR 生成標準 API を呼出していた処理を、【FN003】 QR 生成標準 API を呼出すように改修する

QR チケット相互運用 API システム設計書

- 本システム機能の入力・処理・出力のフローチャート

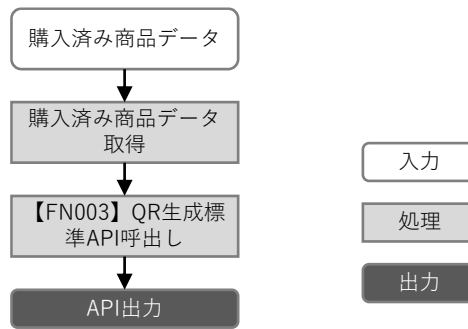


図 2-15 【FN202】のフローチャート

- 本システム機能の処理の詳細
 - 購入済み商品データ取得
 - ◇ 処理内容
 - 購入済み商品データをデータベースから取得
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - 【FN003】QR 生成標準 API 呼出し
 - ◇ 処理内容
 - 【FN003】QR 生成標準 API を呼出し、QR を取得する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ 購入済み商品データ
 - データの内容
 - 購入済み商品データ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF018】購入済み商品データ取得
 - 出力
 - ◇ API 入力
 - データの内容
 - QR 生成情報
 - データの形式

- JSON
- 利用するデータインターフェース
 - 【IF003】 QR 生成標準 API 呼出し用の標準 API

【FN203】商品利用通知標準 API <改修機能>

- 本システム機能の概要
 - チケット認証システムによってチケットが認証されたとき、チケット認証情報を受信し、購入済み商品データのチケットステータスの更新を行う
 - チケット認証情報は、チケット認証システムによって生成されるデータであり、QR が利用された日時、場所、端末の種類、チケットを識別する ID、認証の種別（入場、出場等）が含まれる

既存機能の改修内容

チケット仲介ハブからチケット認証情報を取得するために、既存機能ではチケット認証システムからの商品利用通知を受信していた処理を、チケット仲介ハブからの呼出しを受けするように改修する

- 本システム機能の入力・処理・出力のフローチャート

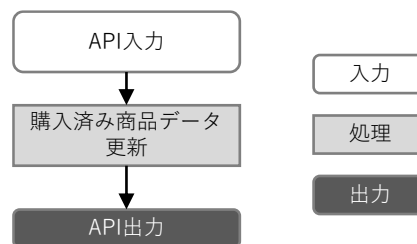


図 2-16 【FN203】のフローチャート

- 本システム機能の処理の詳細
 - 購入済み商品データ更新
 - ◇ 処理内容
 - 購入済み商品データのチケットステータスを変更し、保存する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ API 入力
 - データの内容
 - チケット認証情報
 - データの形式
 - JSON
 - 利用するデータインターフェース

- 【IF006】 商品利用通知標準 API 呼出し用の標準 API
- 出力
 - ◇ API 出力
 - データの内容
 - 処理結果
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF006】 商品利用通知標準 API 呼出し用の標準 API

【FN204】 チケットデータ生成<改修機能>

- 本システム機能の概要
 - 発売商品データの登録を行う。商品に含まれるチケットの共通商品 ID や有効期間、同時利用枚数などの情報を設定する

既存機能の改修内容

チケット仲介ハブの共通商品 ID を設定できるように改修する

- 本システム機能の入力・処理・出力のフローチャート

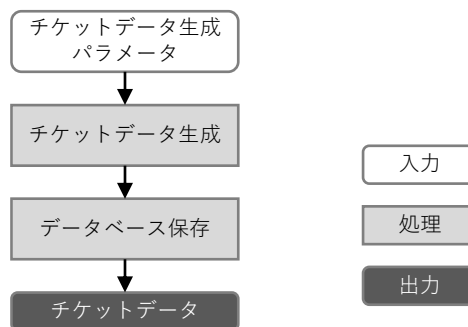


図 2-17 【FN204】 のフローチャート

- 本システム機能の処理の詳細
 - 発売商品データ登録
 - ◇ 処理内容
 - 発売する商品データの内容を登録し、保存する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ WEB 画面

- データの内容
 - 商品データ
- データの形式
 - テキスト
- 利用するデータインターフェース
 - -

【FN301】QR 表示画面

- 本システム機能の概要
 - 購入したチケットを利用するために MaaS アプリをユーザーが操作することにより、QR を表示する
- 本システム機能の入力・処理・出力のフローチャート



図 2-18 【FN301】のフローチャート

- 本システム機能の処理の詳細
 - 画面表示情報
 - ◇ 処理内容
 - 【FN202】QR 取得を呼出すことにより、QR を取得し表示する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ QR
 - データの内容
 - QR データ
 - データの形式
 - バイナリデータ
 - 利用するデータインターフェース
 - 【IF021】QR 表示 IF
 - 出力
 - ◇ 画面表示
 - データの内容

- QR 表示画面
- データの形式
 - 【UI504】 利用中チケット表示
- 利用するデータインターフェース
 - アプリケーションの GUI

【FN302】 QR チケット利用画面

- 本システム機能の概要
 - MaaS アプリにおいて購入済みのチケットを一覧表示し、ユーザーの選択操作を受け付ける
 - ユーザーがチケットを選択したチケットに対して、認証装置でチケットを利用可能にするためにチケット発行呼出機能を実行する
- 本システム機能の入力・処理・出力のフローチャート

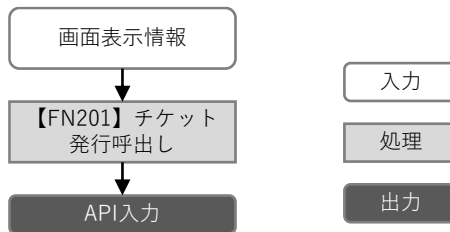


図 2-19 【FN302】 のフローチャート

- 本システム機能の処理の詳細
 - 【FN201】 チケット発行呼出し
 - ◇ 処理内容
 - 【FN201】 チケット発行を呼出し、チケットを発行する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ 画面表示情報
 - データの内容
 - チケット利用画面
 - データの形式
 - 【UI502】 チケット利用・払い戻し
 - 利用するデータインターフェース
 - アプリケーションの GUI
 - 出力
 - ◇ API 入力
 - データの内容

- 処理結果
- データの形式
 - JSON
- 利用するデータインターフェース
 - 【IF020】QR チケット利用 IF

【FN401】チケット発行標準 API②<新規開発>

- 本システム機能の概要
 - チケット仲介ハブから呼び出され、チケットデータを生成しデータベースへ保存する
 - API 入力データのチケット発行情報に含まれる商品 ID をキーに、発行チケットデータから当該チケットの効力の情報を取得し、チケットデータを生成する
- 本システム機能の入力・処理・出力のフローチャート

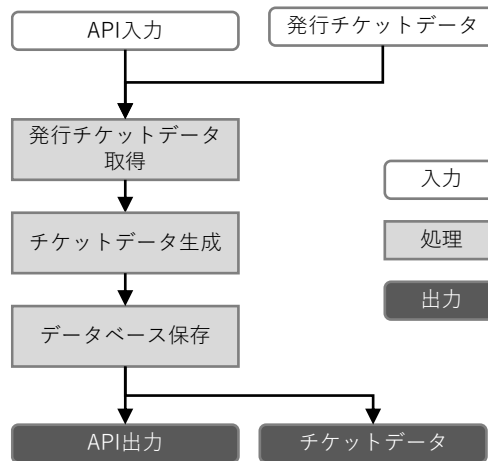


図 2-20 【FN401】のフローチャート

- 本システム機能の処理の詳細
 - 発行チケットデータ取得
 - ◇ 処理内容
 - 発行チケットデータをデータベースから取得する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
 - チケットデータ生成
 - ◇ 処理内容
 - チケットデータを生成する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -

QR チケット相互運用 API システム設計書

- データベース保存
 - ◇ 処理内容
 - 生成したチケットデータをデータベースへ保存する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ API 入力
 - データの内容
 - チケット発行情報
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF004】 チケット発行標準 API 呼出し用の標準 API
 - ◇ 発行チケットデータ
 - データの内容
 - 発行チケットデータ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF011】 発行チケットデータ取得
 - 出力
 - ◇ API 出力
 - データの内容
 - 処理結果
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF004】 チケット発行標準 API 呼出し用の標準 API
 - ◇ チケットデータ
 - データの内容
 - チケットデータ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF014】 チケットデータ保存

【FN402】 共通 QR 受付標準 API② <新規開発>

- 本システム機能の概要
 - チケット仲介ハブにおいて QR が生成されたとき、QR と共通チケット ID が通知される
 - チケットの認証において QR のデータが入力されたとき、車載決済端末に配信するホワイトリストデータとして QR ひも付けデータを生成し、保存する
- 本システム機能の入力・処理・出力のフローチャート

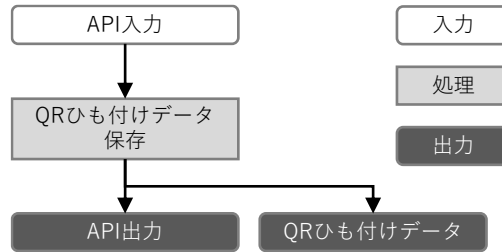


図 2-21 【FN402】のフローチャート

- 本システム機能の処理の詳細
 - QR ひも付けデータ保存
 - ◇ 処理内容
 - 受信した共通 QR データから車載決済端末に配信するホワイトリストデータとして QR ひも付けデータを保存する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ API 入力
 - データの内容
 - 共通 QR 通知
 - チケット仲介ハブで生成された QR データと、共通チケット ID、QR の有効期限
 - データの形式
 - JSON
 - 利用するデータインターフェース
 - 【IF005】 共通 QR 受付標準 API 呼出し用の標準 API
 - 出力
 - ◇ API 出力
 - データの内容
 - 処理結果
 - データの形式
 - JSON

QR チケット相互運用 API システム設計書

- 利用するデータインターフェース
 - 【IF005】 共通 QR 受付標準 API 呼出し用の標準 API
- ◇ QR ひも付けデータ
 - データの内容
 - QR ひも付けデータ
 - 共通チケット ID と QR データ、QR データ有効期限
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF012】 QR ひも付けデータ保存

【FN403】 チケット認証②<新規開発>

- 本システム機能の概要
 - 車載決済端末に QR がかざされたとき、ホワイトリストをサーチしてチケットの利用可否を判定する
 - チケットの判定結果からチケット認証情報を生成し、チケット認証仲介標準 API を呼出すことにより、チケット仲介ハブを通して MaaS アプリバックエンドへ通知する
- 本システム機能の入力・処理・出力のフローチャート

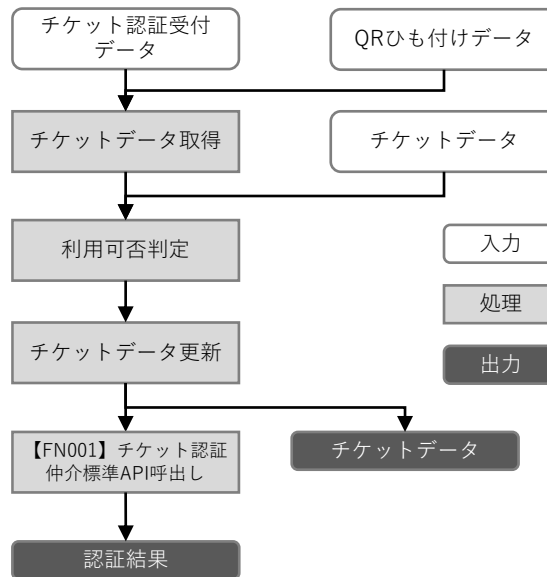


図 2-22 【FN403】 のフローチャート

- 本システム機能の処理の詳細
 - チケットデータ取得
 - ◇ 処理内容
 - QR に紐づくチケット ID を特定し、チケットデータをデータベースから取得する
 - ◇ 利用するライブラリ
 - -

QR チケット相互運用 API システム設計書

- ◇ 利用するアルゴリズム
 - -
- 利用可否判定
 - ◇ 処理内容
 - ホワイトリストをサーチしてチケットの利用可否を判定する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- チケットデータ更新
 - ◇ 処理内容
 - 利用可否判定の結果をチケットデータのステータスに反映する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- 【FN001】 チケット認証仲介標準 API 呼出し
 - ◇ 処理内容
 - 【FN001】 チケット認証仲介標準 API を呼出し、他システムへ認証結果を通知する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様
 - 入力
 - ◇ チケット認証受付データ
 - データの内容
 - チケット認証受付データ
 - データの形式
 - バイナリ
 - 利用するデータインターフェース
 - 端末 ID、業務コード、QR から構成されるデータ
 - ◇ QR ひも付けデータ
 - データの内容
 - QR ひも付けデータ
 - データの形式
 - SQL
 - 利用するデータインターフェース

- 【IF013】 QR ひも付けデータ取得
- 出力
 - ◇ チケットデータ
 - データの内容
 - チケットデータ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - 【IF014】 チケットデータ保存
 - ◇ 認証結果
 - データの内容
 - 認証結果
 - データの形式
 - バイナリ
 - 利用するデータインターフェース
 - チケット認証の結果データ

【FN404】 発行チケットデータ生成

- 本システム機能の概要
 - 入力されたパラメータシートから、発行チケットデータを生成し、データベースに保存する
- 本システム機能の入力・処理・出力のフローチャート

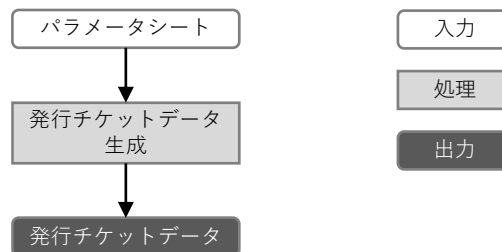


図 2-23 【FN404】のフローチャート

- 本システム機能の処理の詳細
 - 発行チケットデータ生成
 - ◇ 処理内容
 - 入力されたパラメータシートに記録されたデータから、発行商品データを生成しデータベースに保存する
 - ◇ 利用するライブラリ
 - -
 - ◇ 利用するアルゴリズム
 - -
- データ仕様

QR チケット相互運用 API システム設計書

- 入力
 - ◇ パラメータシート
 - データの内容
 - 発行チケットデータの設定値が保存されたデータ
 - データの形式
 - XLSX
 - 利用するデータインターフェース
 - -
- 出力
 - ◇ 発行チケットデータ
 - データの内容
 - 認証システム商品 ID に紐づくチケットの効力データ
 - データの形式
 - SQL
 - 利用するデータインターフェース
 - -

2-1-4. ソフトウェア・ライブラリ (SL) の詳細

表 2-6 ソフトウェア・ライブラリー一覧

※朱文字：新規開発・既存改修

ID	名称	バージョン	内容
SL001	ASP.NET	8	● アプリケーションの実行環境
SL002	PostgreSQL	16	● データを格納する RDB
SL003	チケット仲介ハブ	-	● チケット認証ソフトウェアと MaaS アプリバックエンドの間を中継する、標準インターフェース仕様の API を有するソフトウェア
SL004	チケット認証システム① (日本信号)	-	● 鉄道向けチケット発行機能とチケット認証機能を有するソフトウェア ● 個別チケットの効力を定義するチケットデータと、チケットの利用を記録する利用実績データを保存・管理する
SL005	MaaS アプリバックエンド	-	● MaaS アプリの各機能とチケット仲介ハブとの連携機能を有するソフトウェア
SL006	my route	-	● ユーザー向けのチケット購入、利用の GUI を提供するソフトウェア
SL007	チケット認証システム② (レシップ)	-	● バス向けチケット発行機能とチケット認証機能を有するソフトウェア ● 個別チケットの効力を定義するチケットデータと、チケットの利用を記録する利用実績データを保存・管理する

システムコンポーネントの詳細を記す。なお、本業務において開発（新規・改修）を行うシステムコンポーネントを朱文字で示す。

【SL001】 ASP.NET

- ベンダー
 - Microsoft
- 公式サイト
 - <https://www.asp.net/>
- 本ソフトウェア・ライブラリの概要
 - アプリケーションの実行環境
- 開発するシステムにおいて利用する機能の詳細
 - プログラム実行環境
 - HTTP サーバー機能

【SL002】 PostgreSQL

- ベンダー
 - PostgreSQL Global Development Group
- 公式サイト
 - <https://www.postgresql.org/>
- 本ソフトウェア・ライブラリの概要
 - オープンソース RDB
- 開発するシステムにおいて利用する機能の詳細
 - データベース機能

【SL003】 チケット仲介ハブ<新規開発>

- ベンダー
 - トヨタファイナンシャルサービス・日本信号
- 公式サイト
 - —
- 本ソフトウェア・ライブラリの概要
 - チケット認証ソフトウェアと MaaS アプリバックエンドの間を中継する、標準インターフェース仕様の API を有するソフトウェア
- 開発するシステムにおいて利用する機能の詳細
 - MaaS アプリバックエンドに対してチケット発行機能を提供するためのチケット発行仲介標準 API
 - MaaS アプリバックエンドに対して QR を提供するための QR 生成標準 API
 - チケット認証ソフトウェアが生成した利用実績データを、他システムへ連携するためのチケット認証仲介標準 API
 - 利用実績データを蓄積することにより、認証ステータスデータの保存・管理を実施

【SL004】 チケット認証システム①（日本信号）<既存改修>

- ベンダー
 - 日本信号
- 公式サイト
 - —
- 本ソフトウェア・ライブラリの概要
 - 鉄道向けチケット発行機能とチケット認証機能を有するソフトウェア
 - 個別チケットの効力を定義するチケットデータと、チケットの利用を記録する利用実績データの保存・管理を実施
- 開発するシステムにおいて利用する機能の詳細
 - チケット発行：個別チケットの効力とステータスを保存したチケットデータを生成し保存
 - 共通 QR 受付：チケット仲介ハブが生成した QR とチケットデータをひも付けるため、QR ひも付けデータを保存

QR チケット相互運用 API システム設計書

- チケット認証：MaaS アプリに表示された QR を受け付け、QR に紐づくチケットデータからチケットの効力を判定し、利用実績データを生成・保存

【SL005】 MaaS アプリバックエンド <既存改修>

- ベンダー
 - トヨタファイナンシャルサービス
- 公式サイト
 - —
- 本ソフトウェア・ライブラリの概要
 - MaaS アプリの機能提供と、サーバー間連携機能を提供
- 開発するシステムにおいて利用する機能の詳細
 - チケット仲介ハブのチケット発行仲介標準 API を呼出し、チケットを発行
 - チケット仲介ハブの QR 生成標準 API を呼出し、QR を取得

【SL006】 my route <既存改修>

- ベンダー
 - トヨタファイナンシャルサービス
- 公式サイト
 - —
- 本ソフトウェア・ライブラリの概要
 - ユーザーに対する MaaS アプリケーションの GUI 提供
- 開発するシステムにおいて利用する機能の詳細
 - ユーザーが購入するチケットを選択し、決済操作後に MaaS アプリバックエンドと連携しチケットを発行
 - ユーザーが利用するチケットを選択し、MaaS アプリバックエンドと連携し、QR を表示

【SL007】 チケット認証システム②（レシップ） <既存改修>

- ベンダー
 - レシップ
- 公式サイト
 - —
- 本ソフトウェア・ライブラリの概要
 - バス向けチケット発行機能とチケット認証機能を有するソフトウェア
 - 個別チケットの効力を定義するチケットデータと、チケットの利用を記録する利用実績データの保存・管理を実施
- 開発するシステムにおいて利用する機能の詳細
 - チケット発行：個々チケットの効力とステータスを保存したチケットデータを生成し、保存
 - 共通 QR 受付：チケット仲介ハブが生成した QR とチケットデータをひも付けるため、QR ひも付け

QR チケット相互運用 API システム設計書

- データを保存
- チケット認証：MaaS アプリに表示された QR を受け付け、QR に紐づくチケットデータからチケットの効力を判定

2-1-5. 数理モデル・アルゴリズム (AL) の詳細

表 2-7 数理モデル・アルゴリズム一覧

※朱文字：新規開発・既存改修

ID	名称	説明	アルゴリズムを利用した機能
AL001	TOTP ベースアルゴリズム	<ul style="list-style-type: none"> ● 共通 QR データを生成する ● 一定時間経過後に共通 QR データを使用不可状態にすることができること 	FN002

数理モデル・アルゴリズムの詳細を記す。なお、本業務において開発（新規・改修）を行う数理モデル・アルゴリズムを朱文字で示す。

【AL001】TOTP ベースアルゴリズム

- 本アルゴリズムの概要
 - QR の改ざん防止と有効期限を判定するために、秘密鍵と現在時刻を用いたハッシュ値を生成し、一致を判定する
- 本アルゴリズムを利用した機能
 - 【FN003】QR 生成標準 API
- 本アルゴリズムの計算・処理の詳細
 - 秘密鍵と現在時刻を用いて、鍵付きハッシュアルゴリズム（HMAC-SHA-2-256）でハッシュ値を生成する
 - 現在時刻は、Unix Time から算出するものとし、併せて時間ステップを 10 秒とすることにより、現在時刻の Unix Time を 10 で割った値とする
 - QR の限られたデータサイズに埋め込むために、算出したハッシュ値を一部切り捨て（トランケート）する
 - トランケート方法は、ハッシュ値の末尾 4bit をオフセット値とし、ハッシュ値の先頭からオフセットバイト数を開始位置とし、8byte のデータを抽出する

2-2. システムコンポーネント (CO)

2-2-1. システムコンポーネント図

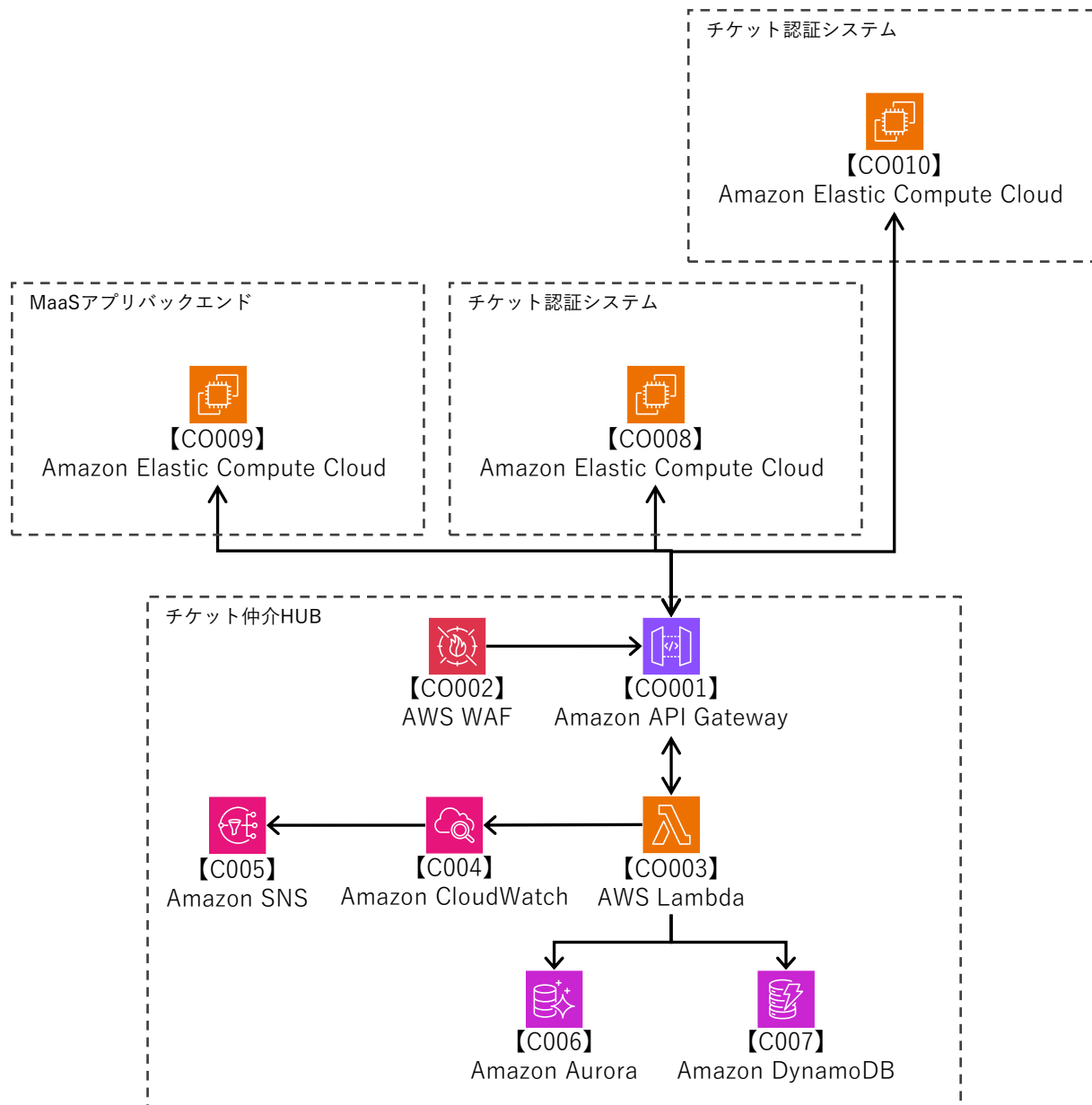


図 2-24 システムコンポーネント図

2-2-2. システムコンポーネント一覧

表 2-8 システムコンポーネント一覧

ID	種別	コンポーネント名	用途
CO001	API サーバー	Amazon API Gateway	<ul style="list-style-type: none"> ● HTTP クライアントからのリクエストを、適切なバックエンドサービスにルーティングし、変換やセキュリティ処理を行うためのシステム ● AWS Lambda で実行するプログラムを外部公開する
CO002	WAF	AWS WAF	<ul style="list-style-type: none"> ● AWS によって提供される Web Application Firewall であり、Web アプリケーションを攻撃から保護する ● Amazon API Gateway と連携して動作し、AWS Lambda で実行されるプログラムを攻撃から保護する
CO003	プログラムサーバー	AWS Lambda	<ul style="list-style-type: none"> ● AWS が提供するサーバーレスコンピューティングによるプログラム実行環境 ● Amazon API Gateway から呼び出され、チケット仲介ハブのプログラムを実行し機能を提供する
CO004	ログ管理	Amazon CloudWatch	<ul style="list-style-type: none"> ● AWS が提供する運用監視環境 ● チケット仲介ハブのログ分析と、システムインフラのパフォーマンス監視を行う
CO005	外部通知	Amazon SNS	<ul style="list-style-type: none"> ● AWS が提供するメッセージングサービス ● Amazon CloudWatch によって検出されたアラートを、メール等により発報・通知する
CO006	DB	Amazon Aurora	<ul style="list-style-type: none"> ● AWS が提供するリレーショナルデータベース ● チケット仲介ハブで発生したデータを保存する
CO007	DB	Amazon DynamoDB	<ul style="list-style-type: none"> ● key-value 型の NoSQL データベース ● チケット仲介ハブで発生したデータにおいて、低遅延での読み取りが必要なデータを保存する
CO008	アプリケーションサーバー	Amazon Elastic Compute Cloud	<ul style="list-style-type: none"> ● AWS が提供する仮想サーバーサービス ● チケット認証システムのプログラム実行
CO009	アプリケーションサーバー	Amazon Elastic Compute Cloud	<ul style="list-style-type: none"> ● AWS が提供する仮想サーバーサービス ● MaaS アプリバックエンドのプログラム実行
CO010	アプリケーションサーバー	Amazon Elastic Compute Cloud	<ul style="list-style-type: none"> ● AWS が提供する仮想サーバーサービス ● チケット認証システムのプログラム実行

2-3. ハードウェア (HW)

2-3-1. ハードウェアアーキテクチャ

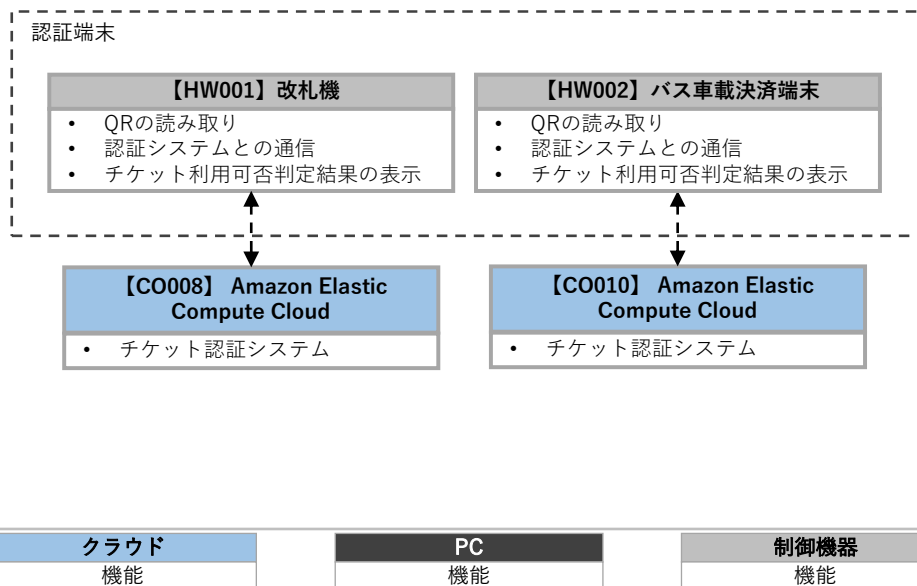


図 2-25 ハードウェアアーキテクチャ

2-3-2. ハードウェア一覧

表 2-9 ハードウェア一覧

※朱文字：新規開発・既存改修

ID	種別	ベンダー	品番	用途
HW001	改札機	日本信号	GX-8	<ul style="list-style-type: none">● QR の読み取り● 認証システムとの通信● チケット利用可否判定結果の表示
HW002	バス車載 決済端末	レシップ	LV-1	<ul style="list-style-type: none">● QR の読み取り● 認証システムとの通信● チケット利用可否判定結果の表示

2-3-3. ハードウェアの詳細

ハードウェアの詳細を記す。なお、本業務において開発（新規・改修）を行うハードウェアを朱文字で示す。

【HW001】改札機

- 本ハードウェアの概要
 - QRを読み取り、チケット認証システムへチケット利用可否の判定を依頼しチケットを認証する
 - チケット効力の判定結果の表示と通路を塞ぐドアの制御を行う
- 本ハードウェアを提供するベンダー
 - 日本信号
- 本ハードウェアの仕様・スペック
 - QRリーダーを有する
 - 認証システムとの通信機能を有する
 - チケット利用可否の判定結果を表示するディスプレイを有する
 - チケット利用可否の判定結果によりユーザーの通行を制御するドアを有する
- イメージ図



図 2-26 改札機

【HW002】 バス車載決済端末

- 本ハードウェアの概要
 - QR を読み取り、チケット認証システムへチケット利用可否の判定を依頼しチケットを認証する
- 本ハードウェアを提供するベンダー
 - レシップ
- 本ハードウェアの仕様・スペック
 - QR リーダーを有する
 - 認証システムとの通信機能を有する
 - チケット利用可否の判定結果を表示するディスプレイを有する
- イメージ図

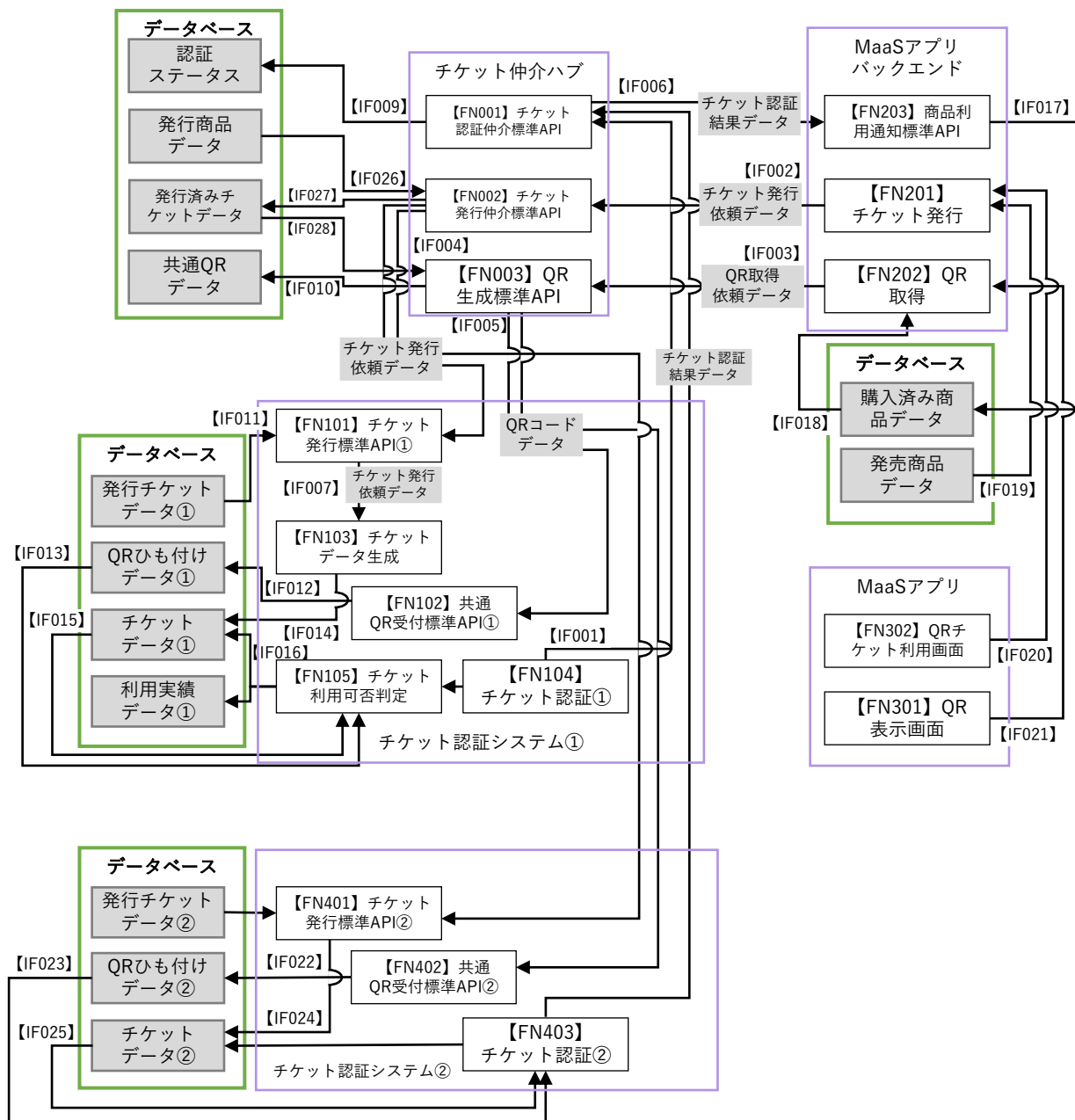
マルチ決済端末「LV-1」イメージ



図 2-27 バス車載決済端末 (引用：レシップ社 HP)

2-4. データインターフェース (IF)

2-4-1. データアーキテクチャ



凡例

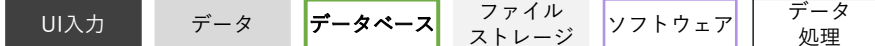


図 2-28 データアーキテクチャ

2-4-2. データインターフェース一覧

表 2-10 データインターフェース一覧

※朱文字：新規開発・既存改修

ID	名称	出力側 ID	入力側 ID
IF001	チケット認証仲介標準 API 呼出し用の標準 API	FN104	FN001
IF002	チケット発行仲介標準 API 呼出し用の標準 API	FN201	FN002
IF003	QR 生成標準 API 呼出し用の標準 API	FN202	FN003
IF004	チケット発行標準 API 呼出し用の標準 API	FN002	FN101
IF005	共通 QR 受付標準 API 呼出し用の標準 API	FN003	FN102
IF006	商品利用通知標準 API 呼出し用の標準 API	FN001	FN203
IF007	チケットデータ生成呼出し IF	FN101	FN103
IF008	チケット利用可否判定呼出し IF	FN104	FN105
IF009	認証ステータス保存	FN001	-
IF010	共通 QR データ保存	FN003	-
IF011	発行チケットデータ取得	-	FN101
IF012	QR ひも付けデータ保存	FN102	-
IF013	QR ひも付けデータ取得	-	FN105
IF014	チケットデータ保存	FN103, FN105	-
IF015	チケットデータ取得	FN105	-
IF016	利用実績データ保存	FN105	-
IF017	購入済み商品データ保存	FN203	-
IF018	購入済み商品データ取得	-	FN202
IF019	発売商品データ取得	FN201	-
IF020	QR チケット利用 IF	FN302	FN201
IF021	QR 表示 IF	FN301	FN202
IF022	QR ひも付けデータ保存	FN103	-
IF023	QR ひも付けデータ取得	-	FN403
IF024	チケットデータ保存	FN402	-
IF025	チケットデータ取得	-	FN403
IF026	発行商品データ取得	-	FN002
IF027	発行済みチケットデータ保存	FN002	-
IF028	発行済みチケットデータ取得	-	FN003

2-4-3. データインターフェースの詳細

データインターフェースの詳細を記す。なお、本業務において開発（新規・改修）を行うデータインターフェースを**朱文字**で示す。

チケット仲介ハブ ER 図

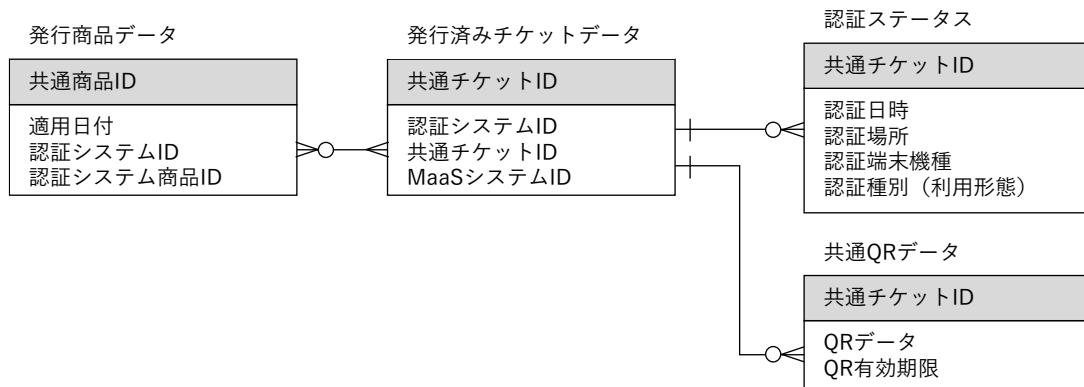


図 2-29 チケット仲介ハブ ER 図

【IF001】 チケット認証仲介標準 API 呼出し用の標準 API

- 本インターフェースの概要
 - チケット認証システムにおいて、チケットの利用可否を判定し、チケットを認証したときの認証情報をチケット仲介ハブへ通知する
- 本インターフェースを利用する機能
 - 【FN104】 チケット認証①
 - 【FN403】 チケット認証②
- 本 API の共通仕様
 - プロトコル
 - ◇ HTTPS
 - メソッド
 - ◇ POST
 - レスポンスデータ形式
 - ◇ JSON
 - 文字コード
 - ◇ UTF-8

➤ 共通リクエストパラメータ

表 2-11 チケット認証仲介標準 API 呼出し用の標準 API 共通リクエストパラメータ

項目	名称	説明	型	必須	値
認証システム ID	authentication_system_id	認証システムの ID	文字列	○	32 ビットの数値
リクエスト ID	request_id	API の実行を特定するための ID。API 実行ごとにインクリメントする	文字列	○	UUID 形式の文字列
認証日時	datetime_at	認証した日時	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00"形式の文字列
ロケーション	location	認証した端末の場所	文字列	○	鉄道の場合 rail-AA-BBBB-CCCC-DDDD rail: 「rail」 固定 AA: 地域を示す数値(HEX) BBBB: 事業者を示す数値(HEX) CCCC: 路線を示す数値(HEX) DDDD: 駅を示す数値(HEX) バスの場合 bus-AA-BBBB-CC-DDDDDDDD-EEEEEEEE bus: 「bus」 固定 AA: 地域を示す数値(HEX) BBBB: 事業者を示す数値(HEX) CC: 営業所を示す数値(HEX) DDDDDDDD: 系統を示す数値(HEX) EEEEEEEE: 停留所を示す数値(HEX)
認証端末タイプ	terminal_type	認証端末の種類	文字列	○	改札機: rial-01 運賃収受機: bus-01
共通チケット ID	common_ticket_id	認証されたチケットを特定するための ID	文字列	○	【FN002】 チケット発行仲介標準 API で生成した文字列
認証種別	use_type	認証の種別	文字列	○	以下のいずれかの文字列 "entry","exit","use"

➤ 共通レスポンスパラメータ

表 2-12 チケット認証仲介標準 API 呼出し用の標準 API 共通レスポンスパラメータ

項目	名称	説明	型	必須	値
処理結果	result	処理結果	真偽値	○	true または false

➤ レスポンスデータ

```
{
  "result": true
}
```

【IF002】 チケット発行仲介標準 API 呼出し用の標準 API

- 本インターフェースの概要
 - MaaS アプリバックエンドがチケット仲介ハブのチケット発行仲介標準 API 機能を実行するための API
 - チケットの発行に必要となるチケット発行依頼データをリクエストパラメータとする
 - チケットの発行した結果として、共通チケット ID をレスポンスパラメータとする
- 本インターフェースを利用する機能
 - **【FN201】** チケット発行
- 本 API の共通仕様
 - プロトコル
 - ◇ HTTPS
 - メソッド
 - ◇ POST
 - レスポンスデータ形式
 - ◇ JSON
 - 文字コード
 - ◇ UTF-8

➤ 共通リクエストパラメータ

表 2-13 チケット発行仲介標準 API 呼出し用の標準 API 共通リクエストパラメータ

項目	名称	説明	型	必須	値
MaaS システム ID	maas_system_id	MaaS システムの ID	文字列	○	数値
リクエスト ID	request_id	API の実行を特定するための ID。API 実行ごとにインクリメントする	文字列	○	UUID 形式の文字列
購入日時	purchase_datetime_at	チケットを購入した日時	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00"形式の文字列
共通商品 ID	common_sales_ticket_id	発行するチケットの商品 ID	数値	○	32 ビットの数値
有効期間	usable_period	有効期間	オブジェクト	○	-
有効期間開始日	start_at	有効期間開始日	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00"形式の文字列
有効期間終了日	end_at	有効期間終了日	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00"形式の文字列
終了	quantity	数量	オブジェクト	○	-
大人数量	adult	大人数量	数値	○	32 ビットの数値
子供数量	child	子供数量	数値	○	32 ビットの数値

➤ 共通レスポンスパラメータ

表 2-14 チケット発行仲介標準 API 呼出し用の標準 API 共通レスポンスパラメータ

項目	名称	説明	型	必須	値
共通チケット ID	common_ticket_id	チケット仲介ハブが発番したチケット ID	オブジェクト	○	-
大人チケット ID	adult	運賃区分が大人のチケット ID	文字列配列	○	UUID 形式の文字列
小児チケット ID	child	運賃区分が小児のチケット ID	文字列配列	○	UUID 形式の文字列

➤ レスポンスデータ

```
{
  "common_ticket_id": {
    "adult": [
      "string"
    ],
    "child": [
      "string"
    ]
  }
}
```

【IF003】 QR 生成標準 API 呼出し用の標準 API

- 本インターフェースの概要
 - 改札機とバス車載端末で利用するための QR の生成を指示する API
 - QR 生成に必要なとなる QR 取得依頼データをリクエストパラメータとする
 - 生成した QR データをレスポンスパラメータとする
- 本インターフェースを利用する機能
 - **【FN202】 QR 取得**
- 本 API の共通仕様
 - プロトコル
 - ◇ HTTPS
 - メソッド
 - ◇ POST
 - レスポンスデータ形式
 - ◇ JSON
 - 文字コード
 - ◇ UTF-8
 - 共通リクエストパラメータ

表 2-15 QR 生成標準 API 呼出し用の標準 API 共通リクエストパラメータ

項目	名称	説明	型	必須	値
MasaS システム ID	maas_system_id	MaaS システムの ID	文字列	○	数値
リクエスト ID	request_id	API の実行を特定するための ID。 API 実行ごとにインクリメントする	文字列	○	UUID 形式の文字列
共通チケット ID	common_ticket_id	チケットの ID	文字列	○	【FN002】 チケット発行仲介標準 API で生成

				した文字列
QR 有効期限	validity_period_at	QR の有効期限	文字列	○ "YYYY-MM-DDThh:mm:ss+09:00" 形式の文字列

➤ 共通レスポンスパラメータ

表 2-16 QR 生成標準 API 呼出し用の標準 API 共通レスポンスパラメータ

項目	名称	説明	型	必須	値
QR データ	qr_code	QR	文字列	○	BASE64 エンコード文字列

➤ レスポンスデータ

```
{
  "qr_code": "string"
}
```

【IF004】 チケット発行標準 API 呼出し用の標準 API

- 本インターフェースの概要
 - MaaS アプリバックエンドがチケット仲介ハブのチケット発行仲介標準 API を呼出したとき、チケット仲介 API がチケット認証システムのチケット発行標準 API を実行するための API
 - チケットの発行に必要なチケット発行依頼データをリクエストパラメータとする
 - チケットの発行した結果として、処理結果をレスポンスパラメータとする
- 本インターフェースを利用する機能
 - **【FN002】** チケット発行仲介標準 API
- 本 API の共通仕様
 - プロトコル
 - ◇ HTTPS
 - メソッド
 - ◇ POST
 - レスポンスデータ形式
 - ◇ JSON
 - 文字コード
 - ◇ UTF-8
 - 共通リクエストパラメータ

表 2-17 チケット発行標準 API 呼出し用の標準 API 共通リクエストパラメータ

項目	名称	説明	型	必須	値
リクエスト ID	request_id	API の実行を特定するための ID API 実行ごとにインクリメント	文字列	○	UUID 形式の文字列

QR チケット相互運用 API システム設計書

共通チケット ID	common_ticket_id	チケット仲介ハブが発番したチケット ID	オブジェクト	○	-
大人チケット ID	adult	運賃区分が大人のチケット ID	文字列配列	○	【FN002】 チケット発行仲介標準 API で生成した文字列
小児チケット ID	child	運賃区分が小児のチケット ID	文字列配列	○	【FN002】 チケット発行仲介標準 API で生成した文字列
購入日時	purchase_datetime_at	チケットを購入した日時	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00"形式の文字列
認証システム商品 ID	auth_sales_ticket_id	発行するチケットの認証システム商品 ID	数値	○	32 ビット数値
有効期限	usable_period	有効期間	オブジェクト	○	-
有効開始日	start_at	有効期間開始日	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00"形式の文字列
有効終了日	end_at	有効期間終了日	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00"形式の文字列
数量	quantity	数量	オブジェクト	○	-
大人数量	adult	大人数量	数値	○	32 ビット数値
子供数量	child	子供数量	数値	○	32 ビット数値

➤ 共通レスポンスパラメータ

表 2-18 チケット発行標準 API 呼出し用の標準 API 共通レスポンスパラメータ

項目	名称	説明	型	必須	値
処理結果	result	処理結果	真偽値	○	true または false

➤ レスポンスデータ

<pre>{ "result": true }</pre>

【IF005】 共通 QR 受付標準 API 呼出し用の標準 API

- 本インターフェースの概要
 - チケット仲介ハブが生成した QR と QR に紐づく共通チケット ID をチケット認証システムへ通知する
- 本インターフェースを利用する機能
 - **【FN003】 QR 生成標準 API**
- 本 API の共通仕様
 - プロトコル
 - ◇ HTTPS
 - メソッド
 - ◇ POST
 - レスポンスデータ形式
 - ◇ JSON
 - 文字コード
 - ◇ UTF-8
 - 共通リクエストパラメータ

表 2-19 共通 QR 受付標準 API 呼出し用の標準 API 共通リクエストパラメータ

項目	名称	説明	型	必須	値
リクエスト ID	request_id	API の実行を特定するための ID API 実行ごとにインクリメント	文字列	○	UUID 形式の文字列
QR データ	qr_code	QR のデータ	文字列	○	BASE64 エンコード文字列
共有チケット ID	common_ticket_id	チケット仲介ハブが発番したチケット ID	文字列	○	【FN002】 チケット発行仲介標準 API で生成した文字列
QR 有効期限	validity_period_at	QR の有効期限	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00" 形式の文字列

- 共通レスポンスパラメータ

表 2-20 共通 QR 受付標準 API 呼出し用の標準 API 共通レスポンスパラメータ

項目	名称	説明	型	必須	値
処理結果	result	処理結果	文字列	○	true または false

- レスポンスデータ

```
{
  "result": true
}
```

【IF006】 商品利用通知標準 API 呼出し用の標準 API

- 本インターフェースの概要
 - チケット認証システムにおいて、チケットの利用可否を判定し、チケットを認証したときの認証情報をチケット仲介ハブから MaaS アプリバックエンドへ通知する
- 本インターフェースを利用する機能
 - **【FN001】 チケット認証仲介標準 API**
- 本 API の共通仕様
 - プロトコル
 - ◇ HTTPS
 - メソッド
 - ◇ POST
 - レスポンスデータ形式
 - ◇ JSON
 - 文字コード
 - ◇ UTF-8
 - 共通リクエストパラメータ

表 2-21 商品利用通知標準 API 呼出し用の標準 API 共通リクエストパラメータ

項目	名称	説明	型	必須	値
リクエスト ID	request_id	API の実行を特定するための ID API 実行ごとにインクリメント	文字列	○	UUID 形式の文字列
認証日時	datetime_at	認証した日時	文字列	○	"YYYY-MM-DDThh:mm:ss+09:00" 形式の文字列
ロケーション	location	認証した端末の場所	文字列	○	鉄道の場合 rail-AA-BBBB-CCCC-DDDD rail: 「rail」 固定 AA: 地域を示す数値 (HEX) BBBB: 事業者を示す数値 (HEX) CCCC: 路線を示す数値 (HEX) DDDD: 駅を示す数値 (HEX) バスの場合

					bus-AA-BBBB-CC-DDDDDDDD-EEEEEEEE bus: 「bus」 固定 AA: 地域を示す数値 (HEX) BBBB: 事業者を示す数値 (HEX) CC: 営業所を示す数値 (HEX) DDDDDDDD: 系統を示す数値 (HEX) EEEEEEEE: 停留所を示す数値 (HEX)
認証端末タイプ	terminal_type	認証端末の種類	文字列	○	改札機: rial-01 運賃収受機: bus-01
共通チケット ID	common_ticket_id	認証されたチケットを特定するための ID	文字列	○	【FN002】 チケット発行仲介標準 API で生成した文字列
認証種別	use_type	認証の種別	文字列	○	以下のいずれかの文字列 "entry","exit","use"

➤ 共通レスポンスパラメータ

表 2-22 商品利用通知標準 API 呼出し用の標準 API 共通レスポンスパラメータ

項目	名称	説明	型	必須	値
処理結果	result	処理結果	真偽値	○	true または false

➤ レスポンスデータ

```
{
  "result": true
}
```

【IF007】 チケットデータ生成呼出し IF

- 本インターフェースの概要
 - チケット認証システム①のチケット発行標準 API①がチケットを発行するとき、生成するチケットデータの詳細パラメータを入力するインターフェース
- 本インターフェースを利用する機能
 - 【FN101】 チケット発行標準 API①

- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-23 チケットデータ生成呼出し IF データデータ詳細

項目	属性	特記事項
チケット No	BIN	チケットマスタのチケット No
開始日・終了日パターン	BIN	チケットの開始日・終了日の利用パターンを設定
乗降利用パターン	BIN	チケットの乗降利用パターンを設定
券種パターン	BIN	チケットの券種パターンを設定
利用日付補正	BIN	日付フリータイプの利用日付補正值 券種パターンが期間固定型(=0)の場合は、null を設定
フリーエリアコード	BIN	フリーエリアのコードを設定
交通モード・券種	BIN	判定に使用するためのチケットの種類を定義
時間・曜日制限	BIN	使用時間・曜日を制限
利用制限開始時刻	BCD	hhmmssxx 00000000～23595900 が設定可能範囲
利用制限終了時刻	BCD	hhmmssxx 00000000～23595900 が設定可能範囲
週次、祝祭日利用制限	BIN	週次の利用制限を曜日ごとに定義
有効開始日時	BCD	発行された後、発券可能になる日時 YYYYMMDDhhmmssxx
有効終了日時	BCD	発行された後、発券可能状態から使用不可になる日時 YYYYMMDDhhmmssxx
券種 1 員数	BIN	券種 1 の員数
券種 2 員数	BIN	券種 2 の員数
券種 3 員数	BIN	券種 3 の員数
券種 4 員数	BIN	券種 4 の員数

【IF008】 チケット利用可否判定呼出し IF

- 本インターフェースの概要
 - チケット認証システムのチケット認証①機能においてチケット利用可否を判定するとき、認証するチケットと認証条件の詳細パラメータを入力するインターフェース
- 本インターフェースを利用する機能
 - **【FN104】 チケット認証①**
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-24 チケット利用可否判定呼出し IF データ詳細

項目	属性	特記事項
チケット ID	BIN	チケット ID を設定
媒体 ID	BIN	媒体 ID を設定

判定日	BCD	0 時更新の実日付 YYYYMMDD 形式
判定時刻	BCD	hhmmss 形式
通路方向	BIN	入場：0x01 出場：0x02(改札機の場合のみ)
媒体種別	BIN	QR：0x21
機種コード	BIN	判定を依頼する機器に応じて設定
業務コード	BIN	実行する業務に応じて設定

【IF009】認証ステータス保存

- 本インターフェースの概要
 - チケット認証システムにおいてチケットが認証されチケット認証仲介標準 API に認証結果が通知されたとき、チケット認証仲介標準 API が認証結果をデータベースに保存する
- 本インターフェースを利用する機能
 - 【FN001】チケット認証仲介標準 API
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-25 認証ステータス保存 データ詳細

名称	説明	型
datetime_at	認証した日時	文字列
location	認証した端末の場所	文字列
terminal_type	認証端末の種類	文字列
common_ticket_id	認証したチケット ID	文字列
use_type	認証の種別	文字列

【IF010】共通 QR データ保存

- 本インターフェースの概要
 - チケット仲介ハブで生成した共通 QR データを保存する
- 本インターフェースを利用する機能
 - 【FN003】QR 生成標準 API
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-26 共通 QR データ保存 データ詳細

名称	説明	型
qr_code	QR のデータ	文字列
common_ticket_id	チケットの ID	文字列
validity_period_at	QR の有効期限	文字列

【IF011】発行チケットデータ取得

- 本インターフェースの概要

QR チケット相互運用 API システム設計書

- チケットを発行するためのパラメータを取得する
- 本インターフェースを利用する機能
 - 【FN101】 チケット発行標準 API①
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-27 発行チケットデータ取得 データ詳細

名称	説明	型
ticket_no	チケット番号	数値
fare_management	チケットパラメータ	文字列

チケットパラメータは、以下の JSON データとする。

表 2-28 発行チケットデータ取得 チケットパラメータデータ詳細

名称	説明	型
UsePattern	チケットの乗降利用パターンを設定	数値
TicketTypePattern	チケットの券種パターンを設定	数値
DateCollection	日付フリータイプの利用日付補正值 券種パターンが期間固定型(=0)の場合は、null を設定	数値
FreeArea	フリーエリアの LocationCode (配列の数は 4 固定とする)	数値
TransitMode	判定に使用するためのチケットの種類を定義	数値
UseLimitTime	日次の利用制限時間帯 00:00 から 23:59 まで設定可能 開始時刻と終了時刻の逆転は許容不可 配列数は 8 固定 [0]: 利用制限開始時刻 1 (hhmm) [1]: 利用制限終了時刻 1 (hhmm) [2]: 利用制限開始時刻 2 (hhmm) [3]: 利用制限終了時刻 2 (hhmm) [4]: 利用制限開始時刻 3 (hhmm) [5]: 利用制限終了時刻 3 (hhmm) [6]: 利用制限開始時刻 4 (hhmm) [7]: 利用制限終了時刻 4 (hhmm)	文字列

名称	説明	型
UseLimitWeekHoliday	週次の利用制限を曜日ごとと祝祭日に対して定義 曜日と祝祭日を定義することから、配列数は 8 固定 [0]：月曜日 (true:利用制限、false:利用制限なし) [1]：火曜日 (true:利用制限、false:利用制限なし) [2]：水曜日 (true:利用制限、false:利用制限なし) [3]：木曜日 (true:利用制限、false:利用制限なし) [4]：金曜日 (true:利用制限、false:利用制限なし) [5]：土曜日 (true:利用制限、false:利用制限なし) [6]：日曜日 (true:利用制限、false:利用制限なし) [7]：祝祭日 (true:利用制限、false:利用制限なし)	真偽値

【IF012】QR ひも付けデータ保存

- 本インターフェースの概要
 - チケット認証システム①の共通 QR 受付標準 API が QR の通知を受けたとき、QR ひも付けデータを保存する
- 本インターフェースを利用する機能
 - 【FN102】共通 QR 受付標準 API①
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-29 QR ひも付けデータ保存 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ
qr_data	QR データ	バイナリ
validity_period	QR データ有効期限	文字列

【IF013】QR ひも付けデータ取得

- 本インターフェースの概要
 - 認証装置で QR を読み取り、チケット認証システム①でチケットの利用可否を判断するとき、チケット利用可否判断機能が QR に関連付けられたチケットデータを取得するために QR ひも付けデータを取得する
- 本インターフェースを利用する機能
 - 【FN105】チケット利用可否判定
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-30 QR ひも付けデータ取得 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ
qr_data	QR データ	バイナリ
validity_period	QR データ有効期限	文字列

【IF014】 チケットデータ保存

- 本インターフェースの概要
 - MaaS アプリにおいてチケットが利用開始の操作が行われ、チケット認証システムでチケットデータが生成されたとき、チケットデータ生成機能がチケットデータを保存する
- 本インターフェースを利用する機能
 - **【FN103】** チケットデータ生成
 - **【FN105】** チケット利用可否判定
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-31 チケットデータ保存 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ
ticket_data	チケットデータ	バイナリ

【IF015】 チケットデータ取得

- 本インターフェースの概要
 - 認証装置で QR を読み取り、チケット認証システム①でチケットの利用可否を判定するとき、チケット利用可否判定機能がチケットの効力とステータスを判定するためにチケットデータを取得する
- 本インターフェースを利用する機能
 - **【FN105】** チケット利用可否判定
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-32 チケットデータ取得 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ
ticket_data	チケットデータ	バイナリ

【IF016】 利用実績データ保存

- 本インターフェースの概要
 - 認証装置で QR を読み取り、チケット認証システム①でチケットが認証されたとき、認証した結果を利用実績データとして保存する
- 本インターフェースを利用する機能
 - **【FN105】** チケット利用可否判定
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-33 利用実績データ保存 データ詳細

名称	説明	型
datetime_at	認証した日時	文字列
location	認証した端末の場所	文字列
terminal_type	認証端末の種類	文字列

common_ticket_id	認証したチケット ID	文字列
use_type	認証の種別	文字列

【IF017】 購入済み商品データ保存

- 本インターフェースの概要
 - 認証装置で QR を読み取り、チケット認証システム①でチケットが認証されたとき、MaaS アプリバックエンドにおいて商品利用通知標準 API で通知された認証情報を購入済み商品データに追加保存する
- 本インターフェースを利用する機能
 - 【FN203】 商品利用通知標準 API
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-34 購入済み商品データ保存 データ詳細

名称	説明	型
datetime_at	認証した日時	文字列
location	認証した端末の場所	文字列
terminal_type	認証端末の種類	文字列
common_ticket_id	認証したチケット ID	文字列
use_type	認証の種別	文字列

【IF018】 購入済み商品データ取得

- 本インターフェースの概要
 - MaaS アプリにおいて QR を表示するとき、MaaS アプリバックエンドがチケット仲介ハブから QR を取得するために、QR 生成標準 API の実行に必要な情報を取得する
- 本インターフェースを利用する機能
 - 【FN202】 QR 取得
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-35 購入済み商品データ取得 データ詳細

名称	説明	型
common_ticket_id	チケット ID	文字列

【IF019】 発売商品データ取得

- 本インターフェースの概要
 - 購入済み商品データから QR 発行パラメータを取得する MaaS アプリにおいてチケットが利用開始の操作が行われたとき、MaaS アプリバックエンドがチケット仲介ハブにチケット発行を指示するために、チケット発行仲介標準 API の実行に必要な情報を取得する
- 本インターフェースを利用する機能
 - 【FN201】 チケット発行
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-36 発売商品データ取得 データ詳細

名称	説明	型
maas_id	MaaS ID	文字列
display_data	表示データ	文字列

【IF020】 QR チケット利用 IF

- 本インターフェースの概要
 - MaaS アプリにおいてチケットが利用開始の操作が行われたとき、購入時にユーザーが入力した情報からチケット発行機能と呼出す
- 本インターフェースを利用する機能
 - 【FN201】 チケット発行
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-37 QR チケット利用 IF データ詳細

名称	説明	型
common_sales_ticket_id	発行するチケットの商品 ID	文字列

【IF021】 QR 表示 IF

- 本インターフェースの概要
 - 購入したチケットを利用するために MaaS アプリをユーザーが操作することにより、選択されたチケットの情報を通知し QR 取得と呼出す
- 本インターフェースを利用する機能
 - 【FN202】 QR 取得
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-38 QR 表示 IF データ詳細

名称	説明	型
common_ticket_id	チケット ID	文字列

【IF022】 QR ひも付けデータ保存

- 本インターフェースの概要
 - 認証装置で QR を読み取り、チケット認証システム②でチケットの利用可否を判断するとき、車載決済端末に配信するホワイトリストデータとして QR ひも付けデータを保存する
- 本インターフェースを利用する機能
 - 【FN402】 共通 QR 受付標準 API②
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-39 QR ひも付けデータ保存 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ
qr_data	QR データ	バイナリ

validity_period	QR データ有効期限	文字列
-----------------	------------	-----

【IF023】 QR ひも付けデータ取得

- 本インターフェースの概要
 - 認証装置で QR を読み取り、チケット認証システム②でチケットの利用可否を判断するとき、QR に関連付けられたチケットデータを取得するために QR ひも付けデータを取得する
- 本インターフェースを利用する機能
 - 【FN403】 チケット認証②
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-40 QR ひも付けデータ取得 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ
qr_data	QR データ	バイナリ
validity_period	QR データ有効期限	文字列

【IF024】 チケットデータ保存

- 本インターフェースの概要
 - MaaS アプリにおいてチケットが利用開始の操作が行われ、チケット認証システムでチケットデータが生成されたとき、チケットデータを保存する
- 本インターフェースを利用する機能
 - 【FN401】 チケット発行標準 API②
 - 【FN403】 チケット認証②
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-41 チケットデータ保存 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ
ticket_data	チケットデータ	バイナリ

【IF025】 チケットデータ取得

- 本インターフェースの概要
 - 認証装置で QR を読み取り、チケット認証システム②でチケットの利用可否を判定するとき、チケットの効力とステータスを判定するためにチケットデータを取得する
- 本インターフェースを利用する機能
 - 【FN403】 チケット認証②
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-42 チケットデータ取得 データ詳細

名称	説明	型
ticket_id	チケット ID	バイナリ

ticket_data	チケットデータ	バイナリ
-------------	---------	------

【IF026】発行商品データ取得

- 本インターフェースの概要
 - チケット発行標準 API がチケット認証システムのチケット発行標準 API を呼出すとき、呼出す認証システムと認証システムにおける商品 ID を特定するためのデータを取得する
- 本インターフェースを利用する機能
 - 【FN002】 チケット発行仲介標準 API
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-43 発行商品データ取得 データ詳細

名称	説明	型
common_sales_ticket_id	チケット仲介ハブにより仲介し発行されるチケットの商品を示す ID	文字列
apply_datetime	当該レコードの適用日を示すデータ	文字列
sales_data	認証システム ID と認証システム商品 ID の辞書型データのリスト	リスト

【IF027】発行済みチケットデータ保存

- 本インターフェースの概要
 - チケット発行標準 API が発行したチケットの情報を保存する
- 本インターフェースを利用する機能
 - **【FN002】** チケット発行仲介標準 API
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-44 発行済みチケットデータ保存 データ詳細

名称	説明	型
common_ticket_id	発行したチケットを特定するための共通チケット ID	文字列
authentication_system_ids	チケットデータを保有する認証システムを特定するための認証システムの ID リスト	リスト
common_sales_ticket_id	チケット仲介ハブにより仲介し発行されるチケットの商品を示す ID	文字列
maas_system_id	チケットを販売した MaaS システムの ID	文字列

【IF028】発行済みチケットデータ取得

- 本インターフェースの概要
 - QR 生成標準 API が生成した QR データを認証システムへ通知するとき、通知する認証システムを特定するために取得する
- 本インターフェースを利用する機能
 - **【FN003】** QR 生成標準 API
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-45 発行済みチケットデータ取得 データ詳細

名称	説明	型
common_ticket_id	発行したチケットを特定するための共通チケット ID	文字列
authentication_system_ids	チケットデータを保有する認証システムを特定するための認証システムの ID リスト	リスト
common_sales_ticket_id	チケット仲介ハブにより仲介し発行されるチケットの商品を示す ID	文字列
maas_system_id	チケットを販売した MaaS システムの ID	文字列

2-5. ユーザーインターフェース (UI)

2-5-1. 画面遷移図

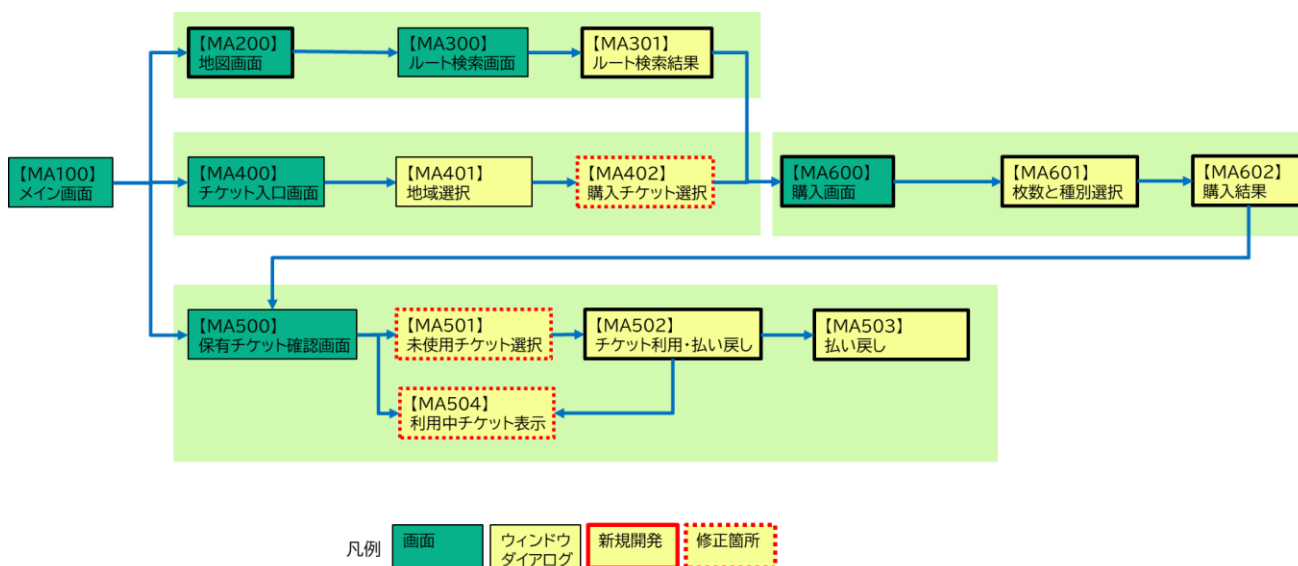


図 2-30 my route 画面遷移図

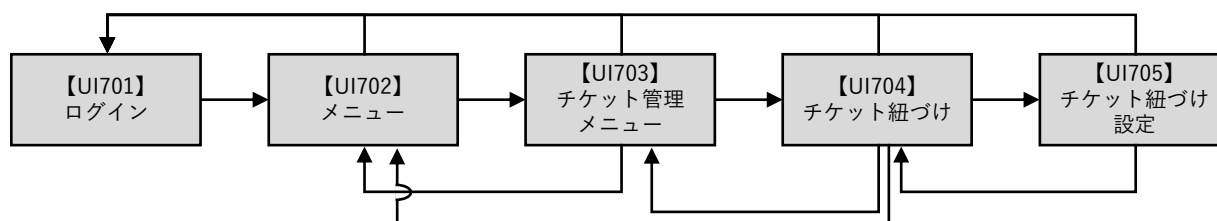


図 2-31 チケット仲介ハブ画面遷移図

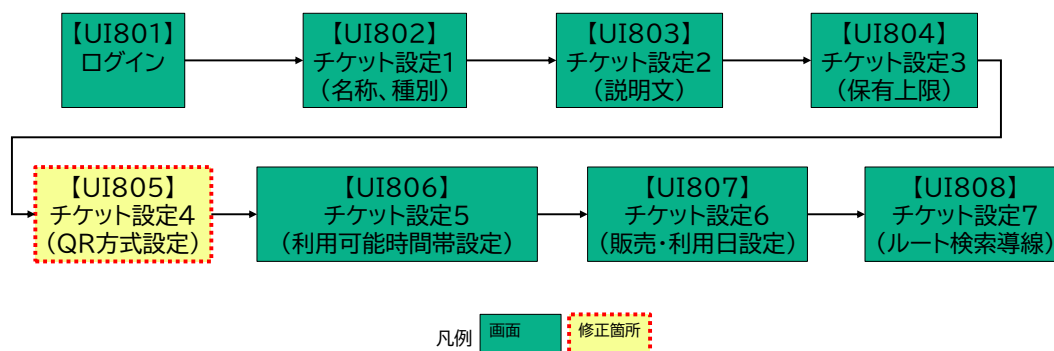


図 2-29 my route チケット造成画面遷移図

2-5-2. ユーザーインターフェース一覧

表 2-46 my route 画面一覧

※朱文字：新規開発・既存改修

ID	画面名	説明	画面を表示した機能 (ID)
UI100	メイン画面	● アプリインストール直後又は起動直後のデフォルト画面	-
UI200	地図画面	● ユーザーが地図の上で、現在地やバス停、駅、施設などを確認する画面	-
UI300	ルート検索画面	● ユーザーがルート検索の操作を行う画面	-
UI301	ルート検索結果	● ルート検索結果を表示する画面	-
UI400	チケット入口画面	● 地図画面を経由せず、直接各地域販売中のチケットを確認する画面	-
UI401	地域選択	● 地域を選択する画面	-
UI402	購入チケット選択	● 該当地域で販売中のチケットを選択する画面	-
UI500	保有チケット確認画面	● 購入済み及び利用中チケットを確認する画面	-
UI501	未使用チケット選択	● 未使用チケットを選択する画面	-
UI502	チケット利用・払い戻し	● 未使用チケットの利用、あるいは払い戻しを操作する画面	【FN201】 チケット発行 【FN203】 商品利用通知標準 API
UI503	払い戻し	● 払い戻しを操作する画面	-
UI504	利用中チケット表示	● 利用中チケットを表示する画面	【FN202】 QR 取得
UI600	購入画面	● チケットを購入する画面	-
UI601	枚数と種別選択	● 購入チケットの枚数と種別を選択する画面	-
UI602	購入結果	● 購入結果を表示する画面	-

表 2-47 チケット仲介ハブ画面一覧

ID	画面名	説明	画面を表示した機能 (ID)
UI701	ログイン	● ID とパスワードにより利用権限を持つ利用者のみ に利用を制限するための画面	-
UI702	メニュー	● 機能ごとにボタンを表示し利用者が使用する機能 を選択するための画面	-
UI703	チケット管理メニュー	● チケット管理の機能ごとにボタンを表示し利用者が 使用するチケット管理機能を選択するための画面	-
UI704	チケット紐づけ	● チケット仲介ハブにおける発行商品データ表示する ための画面 ● 利用者が共通商品 ID を選択すると当該商品を設定す るためのチケット紐づけ設定に表示遷移する	-
UI705	チケット紐づけ設定	● 発行商品データを共通商品 ID ごとに、表示・変更す るための画面	-

表 2-48 my route チケット造成画面一覧

ID	画面名	説明	画面を表示した機能 (ID)
UI801	ログイン画面	● ID とパスワードにより利用権限を持つ利用者のみ に利用を制限するための画面	-
UI802	チケット設定 1 (名称、種別) 画面	● チケットの名称と種別を設定する画面	-
UI803	チケット設定 2 (説明文) 画面	● チケットの説明文を設定する画面	-
UI804	チケット設定 3 (保有上限) 画面	● チケットの保有上限などを設定する画面	-
UI805	チケット設定 4 (QR 方式設定) 画面	● チケットの QR 方式を設定する画面	-
UI806	チケット設定 5 (利用可能時間帯設定) 画面	● チケットの利用可能時間帯などを設定する画面	-
UI807	チケット設定 6 (販売・利用日設定) 画 面	● チケットの販売・利用日を設定する画面	-
UI808	チケット設定 7	● チケットのルート検索導線を設定する画面	-

	(ルート検索導線)画面	
--	-------------	--

表 2-49 レシップチケット造成画面一覧

ID	画面名	説明	画面を表示した機能 (ID)
UI901	チケット情報設定画面 (バス認証機)	● バス認証サービサーは、チケット造成のためのシステムを利用しておらず、プログラムによりチケット情報を書き込む	-

2-5-3. ユーザーインターフェースの詳細

ユーザーインターフェース（画面）の詳細を記す。なお、本業務において開発（新規・改修）を行うユーザーインターフェース（画面）を**朱文字**で示す。

1) my route

【UI100】メイン画面



図 2-32 メイン画面

- 本画面の概要
 - アプリインストール直後 or 起動直後のデフォルト画面
 - 下のタブから「地図で探す」、あるいは「お得におでかけ」に遷移することが可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI200】地図画面



図 2-33 地図画面

- 本画面の概要
 - ユーザーが地図の上で、現在地やバス停、駅、施設などを確認する画面
 - 拡大・縮小、ポイント選択、関連情報の表示可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI300】 ルート検索画面



図 2-34 地図画面

- 本画面の概要
 - ユーザーがルート検索の操作を行う画面
 - 電車・バス、車、徒歩、自転車、マルチモーダルの検索モードを選択可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI301】 ルート検索結果



図 2-35 ルート検索結果

- 本画面の概要
 - ルート検索結果を表示する画面
 - 地図画面とルート検索の詳細を同時に確認可能
 - 「お得なチケットはこちら」から【UI600】購入画面へ遷移可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI400】 チケット入口画面



図 2-36 チケット入口画面

- 本画面の概要
 - 地図画面を経由せず、直接各地域販売中のチケットを確認する画面
 - 最近見たチケットの履歴も表示
 - 電車、バス、船・水上バスのチケットをそれぞれ表示することも可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI401】地域選択



図 2-37 地域選択

- 本画面の概要
 - 地域を選択する画面
 - 北から都道府県順に表示
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI402】 購入チケット選択

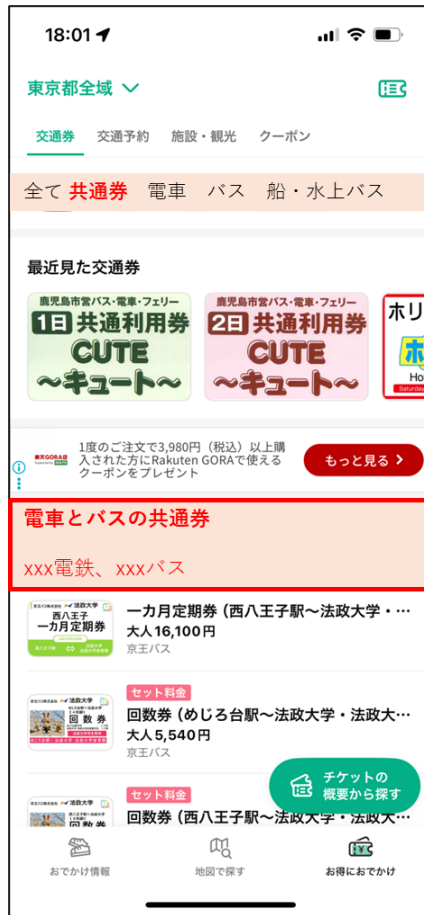


図 2-38 購入チケット選択

- 本画面の概要
 - 該当地域で販売中のチケットを選択する画面
 - 今まで電車とバスの共通で使える QR チケットが無かったため、新たな共通券という種別を追加
 - 共通券説明文書の中で複数の運行事業者の併記が可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI600】 購入画面



図 2-39 購入画面

- 本画面の概要
 - チケットを購入する画面
 - チケットの利用案内と利用可能エリアの情報を確認可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI601】枚数と種別選択



図 2-40 枚数と種別選択

- 本画面の概要
 - 購入チケットの枚数と種別を選択する画面
 - それぞれのチケットに購入枚数と種別の選択可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI602】 購入結果



図 2-41 購入結果

- 本画面の概要
 - 購入結果を表示する画面
 - 正常に購入できた場合、【UI500】 保有チケット確認画面への遷移が可能
- 本画面から利用する機能
 - 【FN201】 チケット発行

【UI500】 保有チケット確認画面



図 2-42 保有チケット確認画面

- 本画面の概要
 - 購入済み及び利用中チケットを確認する画面
 - 利用中のチケットは利用中券面を表示することが可能
 - 未使用チケットは利用開始することが可能
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI501】未使用チケット選択



図 2-43 未使用チケット選択

- 本画面の概要
 - 未使用チケットを選択する画面
 - 電車とバスを両方利用可能なチケットの場合、電車とバスの共通券であることを表示
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI502】 チケット利用・払い戻し



図 2-44 チケット利用・払い戻し

- 本画面の概要
 - 未使用チケットの利用、あるいは払い戻しを操作する画面
- 本画面から利用する機能
 - 【FN201】 チケット発行
 - 【FN203】 商品利用通知標準 API

【UI503】払い戻し



図 2-45 払い戻し

- 本画面の概要
 - 払い戻しを操作する画面
 - 払い戻しについて、所定の払い戻し手数料がかかる場合有
- 本画面から利用する機能
 - my route 独自の画面につき、参照機能 ID なし

【UI504】 利用中チケット表示



図 2-46 利用中チケット表示

- 本画面の概要
 - 利用中チケットを表示する画面
 - 電車とバスの共通券を 1 枚の QR で表示
 - もし 1 枚の QR で電車とバスの共通券を作成できない場合、電車とバスの QR を別々に表示する必要
 - その際、ユーザーが自ら利用する交通手段を選択し、出し分けをする必要
- 本画面から利用する機能
 - 【FN202】 QR 取得



図 2-47 利用中チケット表示 (従来方式の表示方法)

2) チケット仲介ハブ

【UI701】 ログイン

COMmmONS - チケット仲介HUBシステム - 管理Webサイト

ログイン

ID

パスワード

図 2-48 ログイン

- 本画面の概要
 - ID とパスワードにより利用権限を持つ利用者のみ利用を制限するための画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI702】メニュー



図 2-49 メニュー

- 本画面の概要
 - 機能ごとにボタンを表示し利用者が使用する機能を選択するための画面
- 本画面から利用する機能
 - 【FN004】発行商品データ生成

【UI703】 チケット管理メニュー

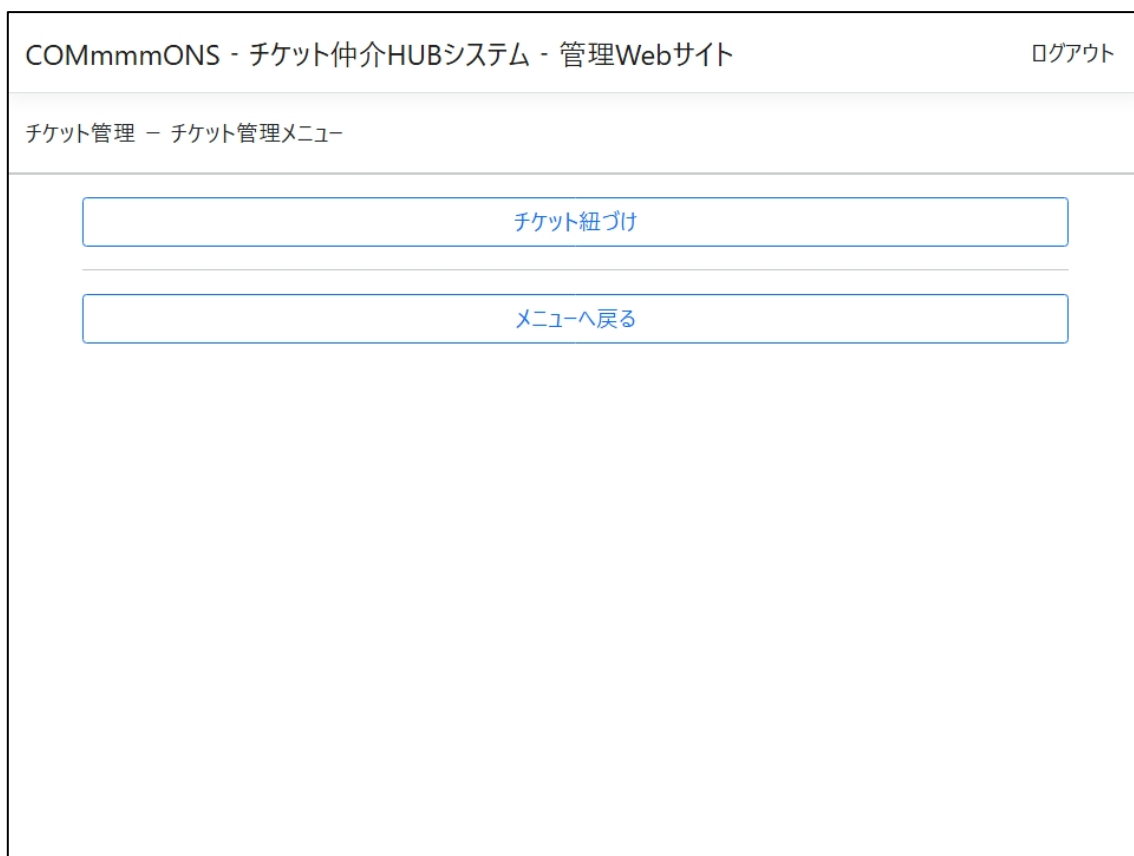


図 2-50 チケット管理メニュー

- 本画面の概要
 - チケット管理の機能ごとにボタンを表示し利用者が使用するチケット管理機能を選択するための画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI704】 チケット紐づけ

COMmmONS - チケット仲介HUBシステム - 管理Webサイト ログアウト

チケット管理 - チケット紐づけ

共通商品ID	適用日時	チケット	
		認証SysID	商品ID
0000004	2025/11/27 03:00:00	1	39
		2	1
		1	38
0000001	1970/01/01 00:00:00	2	1
		1	38
0000003	1970/01/01 00:00:00	2	1
0000002	1970/01/01 00:00:00	1	38

[新規追加](#)

[戻る](#)

[メニューへ戻る](#)

図 2-51 チケット紐づけ

- 本画面の概要
 - チケット仲介ハブにおける発行商品データ表示するための画面
 - 利用者が共通商品 ID を選択すると当該商品を設定するためのチケット紐づけ設定に表示遷移
- 本画面から利用する機能
 - 【FN004】発行商品データ生成

【UI705】 チケット紐づけ設定

COMmmONS - チケット仲介HUBシステム - 管理Webサイト ログアウト

チケット管理 - チケット紐づけ

共通商品ID： 0000001

適用日時： 1970/01/01 00:00:00

チケット1： 認証SysID , 商品ID

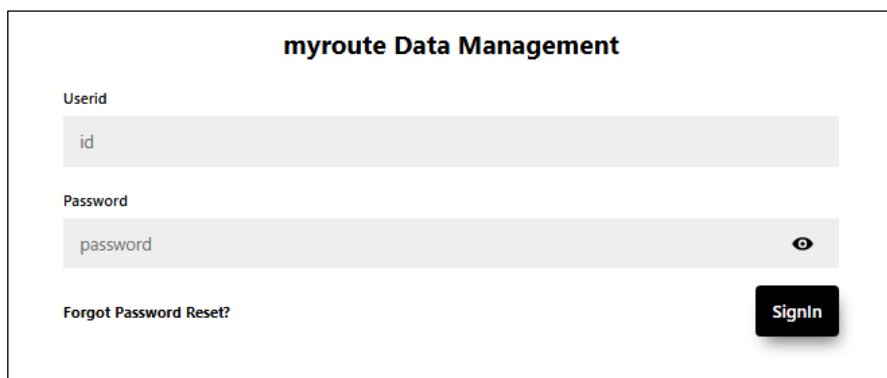
チケット2： 認証SysID , 商品ID

図 2-52 チケット紐づけ設定

- 本画面の概要
 - 発行商品データを共通商品 ID ごとに、表示・変更するための画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

3) my route チケット造成

【UI801】 ログイン画面



The screenshot shows a login interface for 'myroute Data Management'. It includes a 'Userid' input field containing 'id', a 'Password' input field containing 'password' with a visibility toggle icon, a 'Forgot Password Reset?' link, and a 'SignIn' button.

図 2-50 ログイン画面

- 本画面の概要
 - ID とパスワードにより利用権限を持つ利用者のみ利用を制限するための画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI802】 チケット設定 1（名称、種別）画面

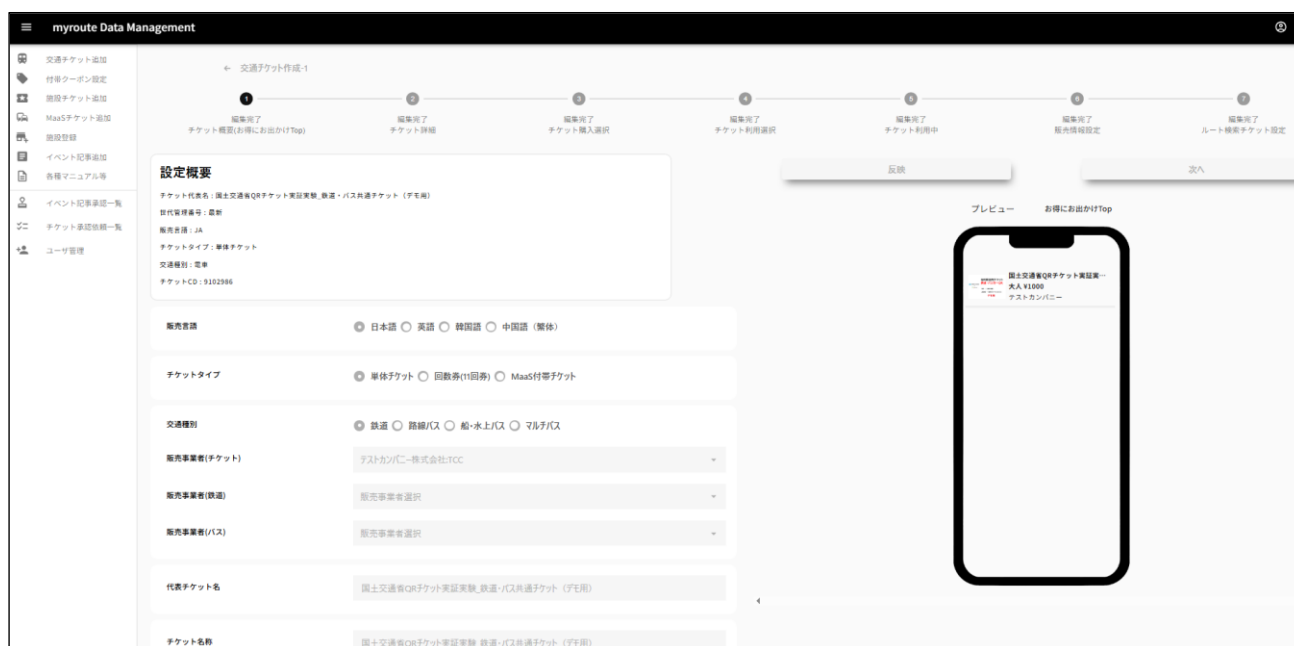


図 2-51 チケット設定 1（名称、種別）

- 本画面の概要
 - チケットの名称と種別を設定する画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI803】 チケット設定 2（説明文）画面

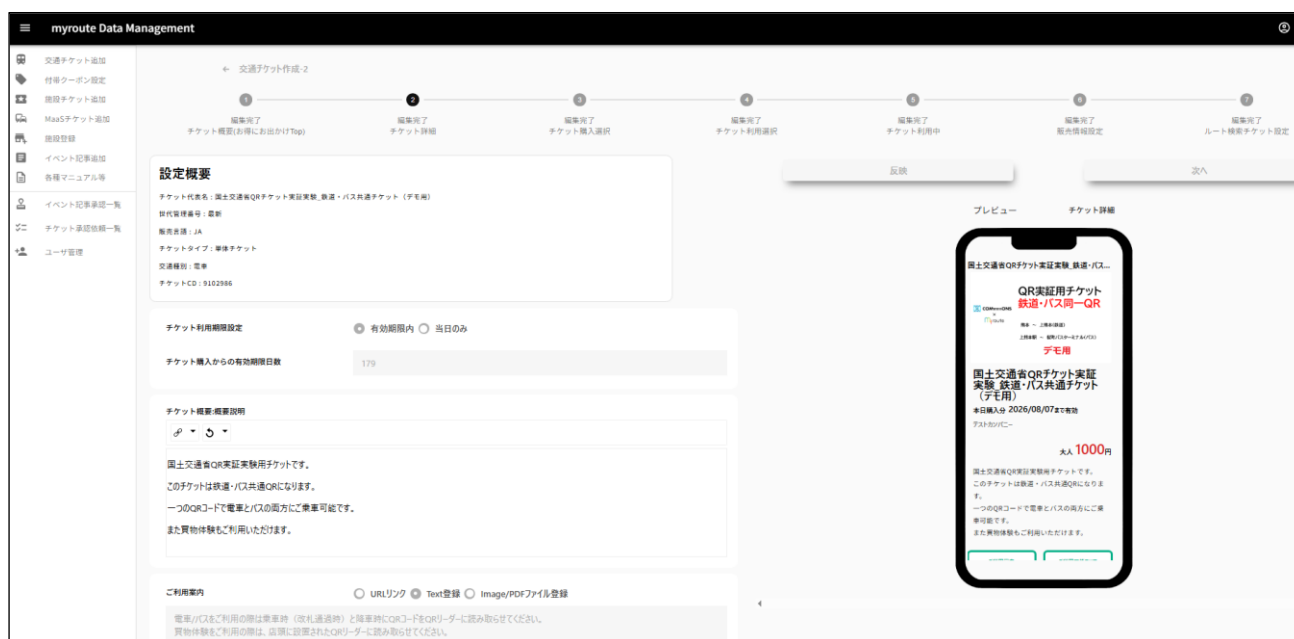


図 2-52 チケット設定 2（説明文）

- 本画面の概要
 - チケットの説明文を設定する画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI804】 チケット設定3（保有上限）画面

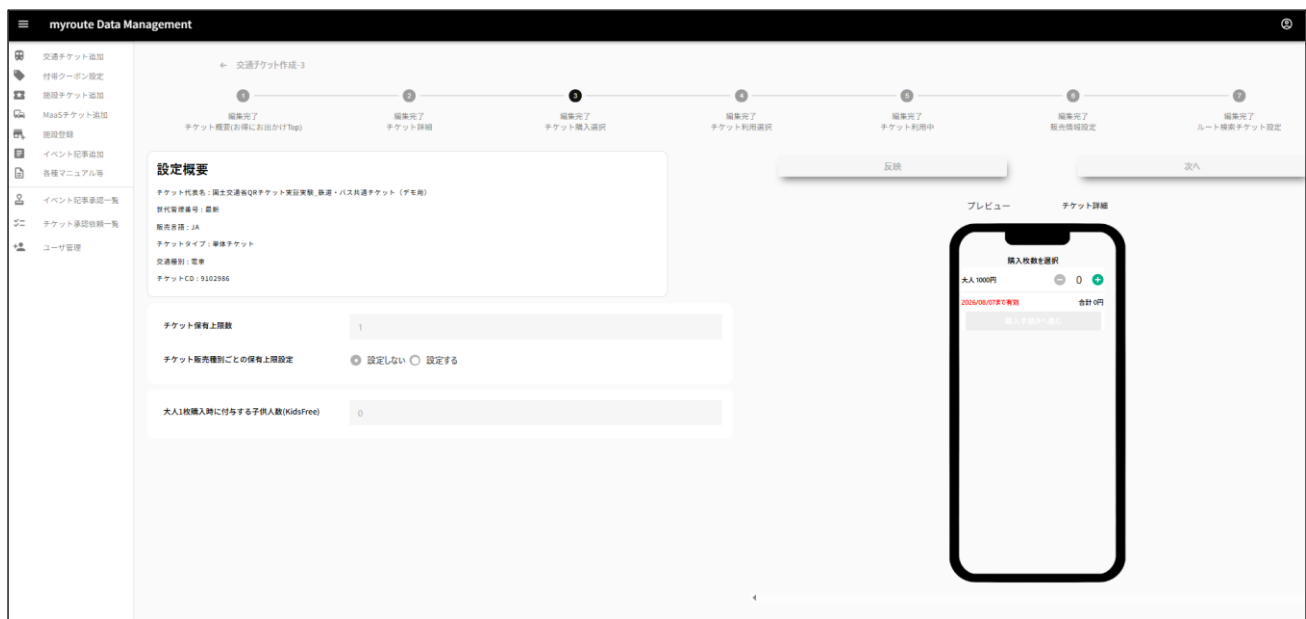


図 2-53 チケット設定3（保有上限）

- 本画面の概要
 - チケットの保有上限などを設定する画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI805】 チケット設定4 (QR方式設定) 画面



図 2-54 チケット設定4 (QR方式設定)

- 本画面の概要
 - チケットの QR 方式を設定する画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI806】 チケット設定 5（利用可能時間帯設定）画面

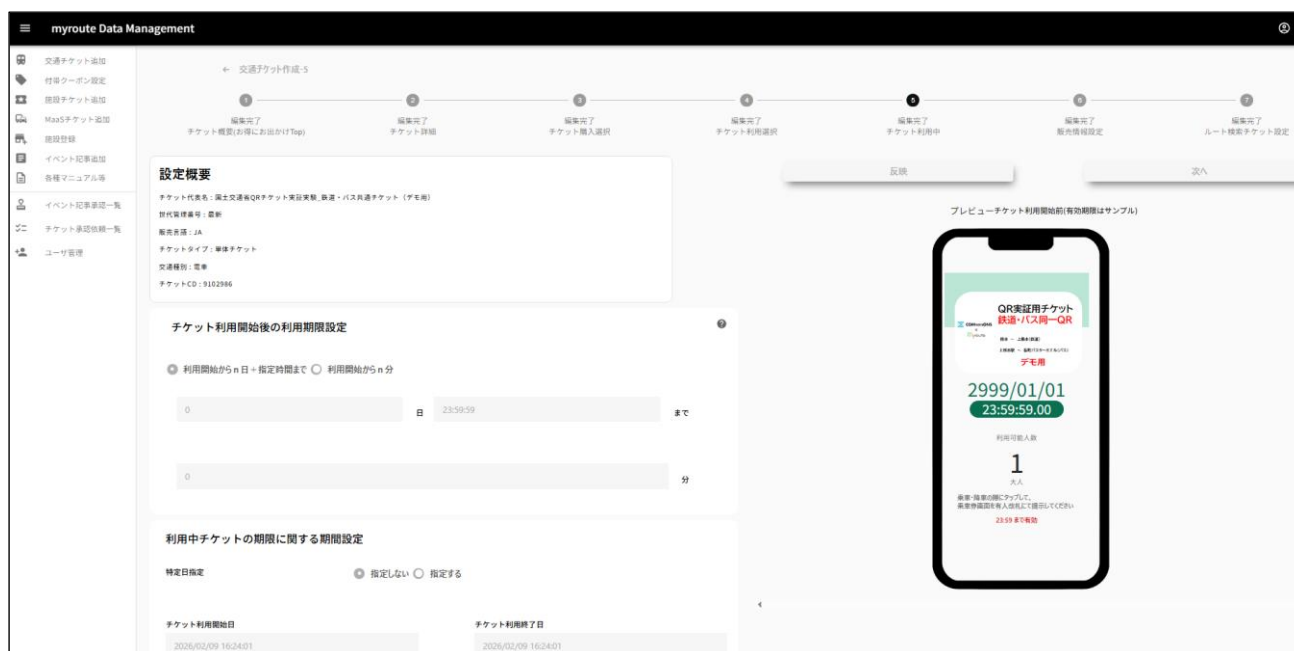


図 2-55 チケット設定 5（利用可能時間帯設定）

- 本画面の概要
 - チケットの利用可能時間帯などを設定する画面
- 本画面から利用する機能
 - 【FN004】 発行商品データ生成

【UI807】 チケット設定6（販売・利用日設定）

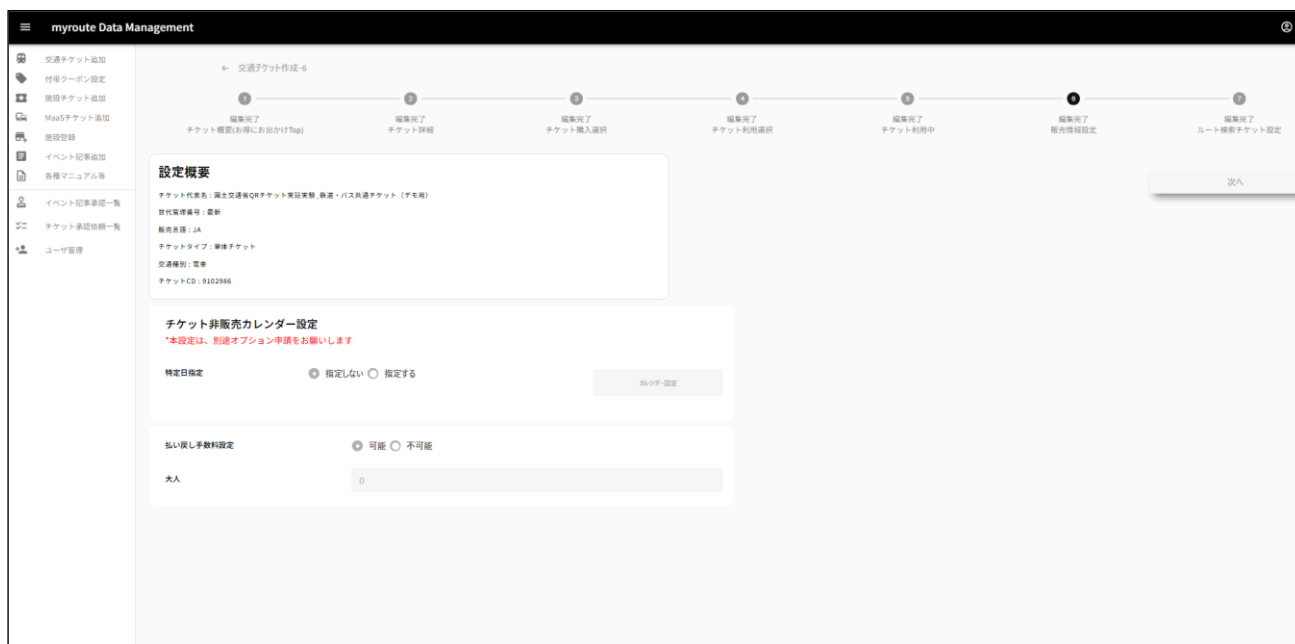


図 2-56 チケット設定6（販売・利用日設定）

- 本画面の概要
 - チケットの販売・利用日を設定する画面
- 本画面から利用する機能
 - 【FN004】発行商品データ生成

【UI808】 チケット設定7（ルート検索導線）画面

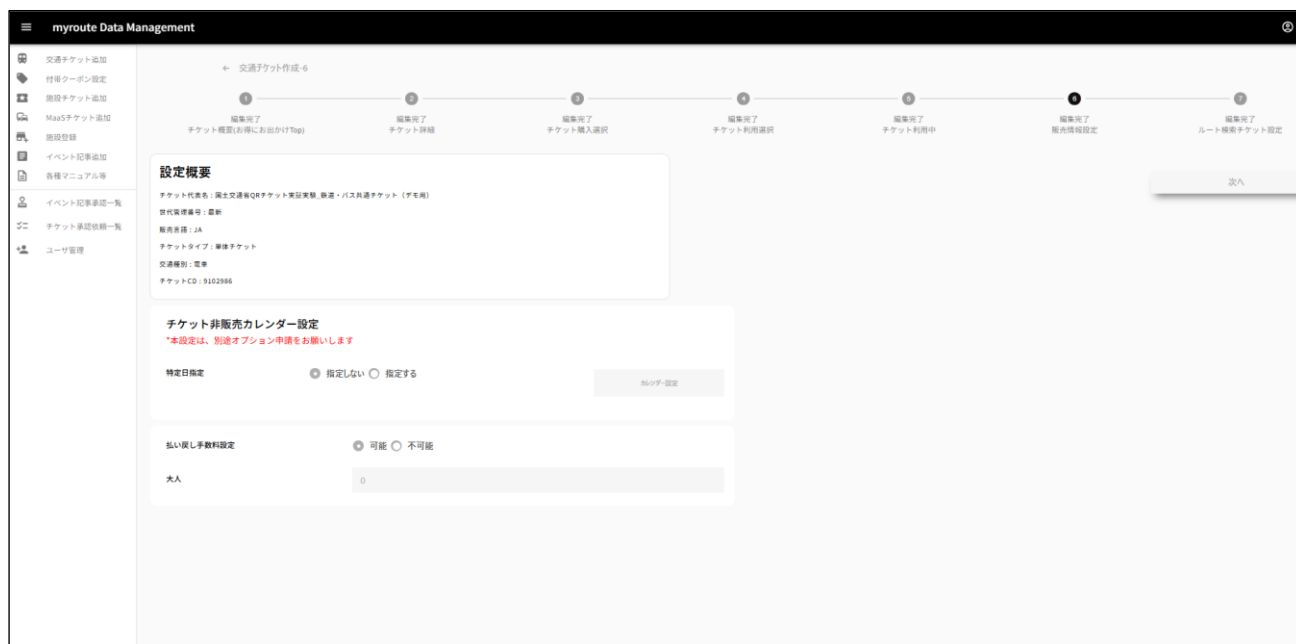


図 2-57 チケット設定7（ルート検索導線）

- 本画面の概要
 - チケットのルート検索導線を設定する画面
- 本画面から利用する機能
 - 【FN004】発行商品データ生成

【UI901】 チケット情報設定画面（バス認証機）

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	
1		協議会コード	日本サイバネフォーメーション	協議会コード毎	協議会コード毎	ランダムシート	協議会コード毎	協議会コード毎	協議会コード毎	協議会コード毎	協議会コード毎	エンコード長	協議会コード毎	協議会コード毎	発行事業者コード	共通チケットID1	障害種別	障害種別	QR有効終了年	QR有効終了月	QR有効終了日	QR有効終了時	QR有効終了分	チケット種類	チケット券種	年齢	チケット期間	大人人数	小児人数	大人人数	小児人数
2						6553	0固定	1固定	0固定	0固定	0固定	74	0固定	1固定	1固定	XXXXXXXX-XXXX-4RR-RRRR-RRRRRRRRRRRR	0固定	24-99	1-12	1-31	0-23	0-59	0固定	0固定	0固定	0固定	0固定	0固定	0固定	0固定	
3		15	0	0	0	0	12345	0	1	0	0	0	0	0	1	1	0	99	12	31	23	59	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	12345	0	1	0	0	0	0	0	1	1	0	99	12	31	23	59	0	0	0	0	0	0	0	0	0
5	2	0	0	0	0	0	12345	0	1	0	0	0	0	0	1	1	0	99	12	31	23	59	0	0	0	0	0	0	0	0	0
6	3	0	0	0	0	0	12345	0	1	0	0	0	0	0	1	1	0	99	12	31	23	59	0	0	0	0	0	0	0	0	0
7	4	0	0	0	0	0	12345	0	1	0	0	0	0	0	1	1	0	99	12	31	23	59	0	0	0	0	0	0	0	0	0
8	5	0	0	0	0	0	12345	0	1	0	0	0	0	0	1	1	0	26	12	31	23	59	0	0	0	0	0	0	0	0	0

図 2-58 チケット情報設定画面（バス認証機）

- 本画面の概要
 - バス認証サービサーは、チケット造成のためのシステムを利用せず、プログラムによりチケット情報を書き込む
 - ◇ チケット QR 情報は C#で記述したプログラムでエクセルファイルを読み込むことで生成する
 - ◇ エクセルファイルには、事前に以下の情報を格納する
 - 発行事業者コード
 - 共通チケット ID
 - QR 有効終了年/月/日/時/分
 - チケット種類
 - チケット券種
 - 年齢
 - チケット期間
 - 大人/小児人数
- 本画面から利用する機能
 - 【FN404】 発行チケットデータ生成

3. QR チケット相互運用 API システム：非機能要件（NF）

3-1. 非機能要件一覧

表 3-1 非機能要件一覧

カテゴリ	ID	非機能項目	要件詳細
可用性	NF001	システムの連続稼働時間	● 24 時間、365 日の安定稼働
	NF002	データの保管期間	● 過去 2 年分のデータ保管を保証する
性能・拡張性	NF003	システムの処理実行速度	● チケット仲介ハブの遅延時間は平均 500ms 未満とする
	NF004	トラフィック数	● 1 秒当たり 1000 アクセス
	NF005	エラー率	● API 実行プログラムのエラー発生率を 0.0028%とする
運用・保守性	NF006	セキュリティ	● WAF による外部からのサイバー攻撃に備える
	NF007	認証	● API は API キーによる認証を行う

3-2. 非機能要件の詳細

【NF001】システムの連続稼働時間

- 本非機能要件の概要
 - 24 時間、365 日の安定稼働。ただし、事前に定めたメンテナンス時間帯は停止する
- 設定理由
 - 24 時間、365 日の安定稼働に耐えうる、標準仕様を策定するため

【NF002】データの保管期間

- 本非機能要件の概要
 - 過去 2 年分のデータ保管を保証する
- 設定理由
 - 昨年分のデータのアクセスを保証するため

【NF003】システムの処理実行速度

- 本非機能要件の概要
 - チケット仲介ハブにおける以下の API の遅延時間は平均 500ms 以下とする
 - ◇ 【FN001】チケット認証仲介標準 API
 - ◇ 【FN002】チケット発行仲介標準 API
 - ◇ 【FN003】QR 生成標準 API
- 設定理由
 - MaaS アプリや認証端末の利用について、違和感なく実行できるようにするため

【NF004】トラフィック数

- 本非機能要件の概要
 - チケット仲介ハブの API ごとに 1 秒あたり 1000 アクセスを最大とする
- 設定理由
 - 全国の一日あたりの鉄道乗降人数を 6300 万人と推定し、6 時から 24 時までの 18 時間に乗降が行われるとした場合、平均で 1 秒あたり 97 アクセスが行われる。さらに、負荷集中時に平均値から 10 倍の負荷があることを想定した

【NF005】エラー率

- 本非機能要件の概要
 - チケット仲介ハブの API ごとにエラー率の平均を 0.0028%とする
- 設定理由
 - 1 秒当たりのアクセス数を 100 とし、エラー発生を 1 時間あたりに 10 回以下とする

QR チケット相互運用 API システム設計書

【NF006】セキュリティ

- 本非機能要件の概要
 - WAF により外部からのサイバー攻撃に備える
- 設定理由
 - HTTP により外部に API を公開することから、情報漏えい防止や改ざん防止のため

【NF007】認証

- 本非機能要件の概要
 - API は api-key による認証を行う
- 設定理由
 - 許可されたクライアント以外からのアクセスを排除するため。

4. 実証調査に利用するデータ (DT)

4-1. 実証調査に利用するデータ一覧

-

4-2. 実証調査に利用するデータの詳細

-

5. 用語集

用語	定義・説明
API 入力データ	API のリクエストパラメータ
購入済み商品データ	MaaS アプリにおいて、ユーザーが購入したチケット商品の情報
QR データ	QR コードにセットされたバイナリデータ
QR ひも付けデータ	QR とチケット ID をひも付けるためのデータ
共通チケット ID	システムにおいてチケットを一意に特定するための ID
チケットステータス	チケットの利用状況を示すステータス
チケットデータ	チケットの効力と状態を保存したデータ
チケット認証情報	チケットが改札機、運賃収受機等で認証されたときに生成される、認証された日時、場所、機器、チケット、種別を示すデータ
チケット発行情報	チケットを発行するために必要となるデータであり、発行するチケットの共通商品 ID、有効期間、数量を含むデータ
デジタルチケットティング	MaaS サービサーが発行した電子チケットをエンドユーザーが利用すること
認証サービサー	電子チケットの有効性を認証・判定するシステムを提供する者
発行チケットデータ	発行するチケットの利用可能日数や区間等の効力を定義するデータ
MaaS サービサー	アプリケーション又は Web サービスを通して交通に関わる電子チケットを提供する者
利用実績データ	チケットが改札機、運賃収受機等で認証されたときに生成される、チケットの利用実績のデータ



**QRチケット相互運用API システム設計書
Ver1.0**

発行日: 2026年3月

委託者: 国土交通省 総合政策局
モビリティサービス推進課

受託者: トヨタファイナンシャルサービス株式会社
日本信号株式会社