📄 API設計書（sfr.tokyo）

# エンドポイント一覧

## 1. 評議員関連エンドポイント

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| GET | /councils | 必須 | 評議員一覧取得 |
| POST | /councils | 必須 | 評議員登録（管理者） |
| POST | /councils/{id}/vote | 必須 | 投票処理 |

## 2. キャラクター関連エンドポイント

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| GET | /characters | 必須 | 自分のキャラクター一覧取得 |
| POST | /characters | 必須 | キャラクター新規作成 |
| PUT | /characters/{id} | 必須 | キャラクター情報更新 |
| PATCH | /characters/{id}/status | 必須 | ステータス変更（死亡など） |
| GET | /characters/{id} | 必須 | キャラクター詳細取得 |

## 3. チーム関連エンドポイント

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| GET | /teams | 必須 | 所属チーム一覧取得 |
| POST | /teams | 必須 | チーム作成 |
| POST | /teams/{id}/members | 必須 | メンバー追加申請 |
| PATCH | /teams/{id}/transfer | 必須 | チーム運営移譲 |
| GET | /teams/{id}/posts | 任意 | チーム作品一覧取得 |

## 4. 出納帳関連エンドポイント

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| GET | /wallets/{id} | 必須 | 出納帳詳細取得 |
| POST | /wallets/{id}/entries | 必須 | 出納記録追加 |
| GET | /wallets/{id}/summary | 必須 | 月次収支サマリー取得 |

## 5. 経済活動・ショップ関連エンドポイント

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| POST | /shops | 必須 | オンラインショップ作成 |
| POST | /shops/{id}/items | 必須 | 商品登録（新品／中古） |
| PATCH | /shops/{id}/items/{itemId}/status | 必須 | 商品ステータス変更 |
| GET | /shops/{id}/items | 任意 | 商品一覧取得 |
| POST | /auctions | 必須 | オークション作成 |
| POST | /auctions/{id}/bid | 必須 | 入札処理 |
| GET | /auctions/{id} | 任意 | オークション詳細取得 |

## 6. 学習機能関連エンドポイント（学校／オンラインサロン／公式ファンクラブ共通）

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| POST | /learning-spaces | 必須 | 学習空間作成 |
| POST | /learning-spaces/{id}/contents | 必須 | コンテンツ登録 |
| GET | /learning-spaces/{id}/progress | 必須 | 進捗取得 |
| POST | /learning-spaces/{id}/feedback | 必須 | AIフィードバック送信 |
| GET | /learning-spaces/{id}/community | 任意 | ディスカッション取得 |
| POST | /learning-spaces/{id}/points | 必須 | ポイント／バッジ付与 |

# リクエスト/レスポンス形式（DTOとの対応）

## 1. POST /councils（評議員登録）

**Request Body（CouncilMemberDto）**

{

"userId": "uuid",

"termStart": "2025-09-01",

"termEnd": "2028-08-31",

"role": "MEDIATION",

"notes": "初期評議員"

}

**Response**

{

"id": "uuid",

"userId": "uuid",

"termStart": "2025-09-01",

"termEnd": "2028-08-31",

"role": "MEDIATION",

"notes": "初期評議員"

}

## 2. POST /characters（キャラクター作成）

**Request Body（CharacterDto）**

{

"name": "ミュージシャン用キャラ",

"description": "音楽活動用の人格",

"status": "ACTIVE"

}

**Response**

{

"characterId": "uuid",

"userId": "uuid",

"name": "ミュージシャン用キャラ",

"description": "音楽活動用の人格",

"status": "ACTIVE",

"lifespanPoints": 365,

"createdAt": "2025-08-14T12:00:00Z"

}

## 3. POST /teams（チーム作成）

**Request Body（TeamDto）**

{

"name": "文学サークル",

"type": "LITERATURE",

"leaderCharacterId": "uuid"

}

**Response**

{

"teamId": "uuid",

"name": "文学サークル",

"type": "LITERATURE",

"leaderCharacterId": "uuid",

"walletId": "uuid"

}

## 4. POST /wallets/{id}/entries（出納記録追加）

**Request Body（WalletEntryDto）**

{

"amount": 1200,

"type": "INCOME",

"source": "オンラインショップ",

"description": "Tシャツ販売",

"isTaxIncluded": true,

"notes": "初回販売分"

}

**Response**

{

"entryId": "uuid",

"walletId": "uuid",

"amount": 1200,

"type": "INCOME",

"source": "オンラインショップ",

"description": "Tシャツ販売",

"isTaxIncluded": true,

"notes": "初回販売分",

"timestamp": "2025-08-14T12:30:00Z"

}

## 5. POST /shops/{id}/items（商品登録）

**Request Body（ProductDto）**

{

"name": "限定ポスター",

"price": 2500,

"type": "PHYSICAL",

"creatorId": "uuid",

"status": "PUBLIC",

"description": "2025年夏イベント限定"

}

**Response**

{

"id": "uuid",

"name": "限定ポスター",

"price": 2500,

"type": "PHYSICAL",

"creatorId": "uuid",

"status": "PUBLIC",

"description": "2025年夏イベント限定"

}

## 6. POST /learning-spaces（学習空間作成）

**Request Body（LearningSpaceDto）**

{

"type": "SCHOOL",

"ownerUserId": "uuid",

"title": "創作基礎講座",

"description": "初心者向けの創作入門コース"

}

**Response**

{

"id": "uuid",

"type": "SCHOOL",

"ownerUserId": "uuid",

"title": "創作基礎講座",

"description": "初心者向けの創作入門コース",

"createdAt": "2025-08-14T13:00:00Z"

# } 認証・認可要件

* **JWT（JSON Web Token）**
  + **各リクエストにおいて、AuthorizationヘッダーにBearerトークンを付与する必要があります。**
  + **トークンの有効期限は24時間とし、リフレッシュトークンを用いて再発行可能とします。**
* **ロールベースのアクセス制御**
  + **各エンドポイントに対して、以下のロールを前提としたアクセス制御を実施します。**
    - **ADMIN: 管理者権限を持つユーザー**
    - **USER: 一般ユーザー**
    - **GUEST: 認証不要のエンドポイントにアクセス可能**
  + **例: @PreAuthorize("hasRole('ADMIN')")**

# バリデーションルール

* **共通ルール**
  + **必須項目: DTO内の@NotNullアノテーションで定義**
  + **文字数制限: @Size(min=1, max=255)などで制約**
  + **フォーマット: @Patternを用いて正規表現で検証（例: UUID形式、メールアドレス形式）**
* **エンドポイント別ルール**
  + **POST /councils**
    - **termStartとtermEndの期間整合性をチェック**
    - **roleは定義済みの列挙型（MEDIATION, ARBITRATIONなど）である必要あり**
  + **POST /characters**
    - **nameはユニークである必要あり（DB側で一意性制約を設定）**
    - **statusはACTIVE, INACTIVEのいずれか**
  + **POST /wallets/{id}/entries**
    - **amountは正の整数**
    - **isTaxIncludedがtrueの場合、amountに消費税が含まれていることを確認**

# エラーコードとメッセージ

* **400 Bad Request**
  + **入力値が不正な場合**
  + **メッセージ例: "Invalid input: 'termStart' must be before 'termEnd'"**
* **403 Forbidden**
  + **認可されていない操作を試みた場合**
  + **メッセージ例: "Access denied: Admin role required"**
* **422 Unprocessable Entity**
  + **バリデーションエラーが発生した場合**
  + **メッセージ例: "Validation failed: 'name' must be unique"**

**🔐 暗号資産関連エンドポイント**

**エンドポイント一覧**

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| GET | /crypto-rewards | 必須 | 自分の暗号資産付与履歴取得 |
| POST | /crypto-rewards | 必須 | 暗号資産付与処理（運営または自動） |
| GET | /wallets | 必須 | 自分のウォレット一覧取得 |
| POST | /wallets | 必須 | ウォレット新規作成 |
| PATCH | /wallets/{id}/status | 必須 | ウォレットの有効／無効切り替え |

**DTO定義例**

**POST /crypto-rewards（CryptoRewardDto）**

json

{

"characterId": "uuid",

"userId": "uuid",

"pointsUsed": 30,

"tokenAmount": 0.005,

"mirroredToAdmin": true

}

**Response**

json

{

"id": "uuid",

"characterId": "uuid",

"userId": "uuid",

"pointsUsed": 30,

"tokenAmount": 0.005,

"issuedAt": "2025-08-14T14:00:00Z",

"mirroredToAdmin": true

}

**POST /wallets（WalletDto）**

json

{

"userId": "uuid",

"address": "0xabc123def456"

}

**Response**

json

{

"id": "uuid",

"userId": "uuid",

"address": "0xabc123def456",

"createdAt": "2025-08-14T14:05:00Z",

"isActive": true

}

**バリデーションルール**

* pointsUsed: 正の整数、lifespan\_points を超えないこと
* tokenAmount: 0以上の小数
* address: Ethereum形式のウォレットアドレス（例: 0x で始まる40文字）

**🎓 学習関連エンドポイント**

**エンドポイント一覧**

| **メソッド** | **パス** | **認証** | **機能** |
| --- | --- | --- | --- |
| POST | /learning-history | 必須 | 学習履歴登録 |
| PATCH | /learning-history/{id}/progress | 必須 | 進捗更新 |
| POST | /learning-history/{id}/feedback | 必須 | AIフィードバック登録 |
| GET | /learning-progress | 必須 | 自分の学習進捗一覧取得 |
| PATCH | /learning-progress/{id}/badges | 必須 | バッジ追加処理 |

**DTO定義例**

**POST /learning-history（LearningHistoryDto）**

json

{

"characterId": "uuid",

"lessonId": "uuid",

"progress": 0

}

**Response**

json

{

"id": "uuid",

"characterId": "uuid",

"lessonId": "uuid",

"progress": 0,

"completionStatus": null,

"score": null,

"aiFeedbackId": null

}

**PATCH /learning-progress/{id}/badges（BadgeUpdateDto）**

json

{

"badges": ["初級修了", "創作力向上"]

}

**Response**

json

{

"id": "uuid",

"userId": "uuid",

"learningSpaceId": "uuid",

"completedPercent": 80,

"badges": ["初級修了", "創作力向上"],

"lastAccessed": "2025-08-14T14:30:00Z"

}

**バリデーションルール**

* progress: 0〜100の整数
* badges: JSON配列、各要素は255文字以内
* completionStatus: ENUM（例: COMPLETED, IN\_PROGRESS, FAILED）

# ENUM定義一覧（sfr.tokyo）

このドキュメントは、DB・API・フロントエンドで共通利用されるENUM定義を一元管理するためのものです。

---

## 🧑‍💼 ユーザー関連

### role

- USER

- ADMIN

---

## 🧙 キャラクター関連

### status

- ACTIVE

- INACTIVE

- DECEASED

---

## 👥 チーム関連

### role（team\_member）

- ARTIST

- STAFF

### economic\_role（team\_member）

- SHOP\_OWNER

- TEACHER

- MENTOR

- NONE

### type（team\_entity）

- MUSIC

- LITERATURE

- VISUAL\_ART

- PERFORMANCE

- OTHER

---

## 💰 出納帳関連

### type（wallet\_entry）

- INCOME

- EXPENSE

---

## 🛍️ 商品関連

### type（product）

- DIGITAL

- PHYSICAL

### status（product）

- PUBLIC

- PRIVATE

- ARCHIVED

---

## 📦 注文関連

### status（order）

- PENDING

- COMPLETED

- CANCELLED

---

## 💳 決済関連

### payment\_method（transaction）

- CREDIT\_CARD

- BANK\_TRANSFER

- CRYPTO

- PAYPAL

---

## 🧑‍⚖️ 評議員関連

### role（council\_member）

- MEDIATION

- PROPOSAL

- ARBITRATION

---

## 🪙 暗号資産関連

### mirrored\_to\_admin（crypto\_reward）

- true

- false

---

## 🎓 学習関連

### type（learning\_space）

- SCHOOL

- ONLINE\_SALON

- OFFICIAL\_FANCLUB

### completion\_status（learning\_history）

- COMPLETED

- IN\_PROGRESS

- FAILED

---

## 🏅 バッジ関連（learning\_progress）

形式：JSON配列

例：

```json

["初級修了", "創作力向上", "ファン交流達成"]

**📌 管理ルール**

* ENUM定義はDB・API・フロントで共通利用されるため、変更時は全体に通知すること
* 新規追加時はこのファイルに追記し、各層のバリデーションに反映すること
* ENUMの表示名と内部値が異なる場合は、別途 enum\_labels.md を作成すること