

# 要件定義書

ECサイト（汎用オンラインショッピングサービス）

Requirements Definition Document

項目	内容
ドキュメントID	EC-REQ-001
バージョン	1.0
作成日	2026年2月
作成者	Wai Wai
ステータス	ドラフト
対象システム	ECサイト（一般商品販売）

## 1. プロジェクト概要

### 1.1 目的

本プロジェクトは、一般消費者向けのオンラインショッピングサービス（ECサイト）を構築することを目的とする。ユーザーが商品の閲覧・検索・購入・注文管理を行えるプラットフォームを提供し、管理者が商品・注文・ユーザーを一元管理できる管理画面も含む。

### 1.2 背景

ポートフォリオ用プロジェクトとして、エンタープライズレベルのフルスタック開発スキルを実証するために本システムを開発する。日本国内での就職活動における技術力のアピール材料とする。

### 1.3 スコープ(対象範囲)

- ・ フロントエンド：Next.js (App Router) を使用した顧客向けEC画面
- ・ バックエンド：NestJSを使用したRESTful API
- ・ データベース：AWS RDS (PostgreSQL) + Prisma ORM
- ・ ファイルストレージ：AWS S3 (商品画像) + CloudFront (CDN配信)
- ・ 決済システム：Stripe決済連携
- ・ 認証：JWTトークンによる認証・認可
- ・ 管理画面：商品・注文・ユーザー管理機能

### 1.4 対象外スコープ

- ・ 実店舗との在庫連携
- ・ 外部配送業者APIとの連携
- ・ 多言語対応（日本語のみ）
- ・ モバイルアプリ（Webのみ）

## 2. ターゲットユーザー

### 2.1 ユーザーロール一覧

ロール	説明	認証
ゲスト	会員登録なしで商品閲覧・検索が可能	不要
一般会員	会員登録後、カート・購入・注文履歴が利用可能	必要 (JWT)
管理者	商品・注文・ユーザーの管理が可能	必要 (JWT + Role Guard)

### 2.2 ペルソナ

ペルソナ A: 一般会員 (田中さん、30代、東京在住)

- ・スマートフォンよりPCを好む
- ・通販をよく利用するが、サイトが分かりにくいと離脱する
- ・セキュリティに敏感でクレジットカード情報の管理を重視する
- ・注文後は配送状況をすぐに確認したい

ペルソナ B: 管理者 (ECサイト運営担当者)

- ・Excelに慣れており、一覧表示や検索・フィルタが使いやすいことを重視
- ・商品登録・在庫管理・注文対応を効率よく行いたい
- ・操作ミスを防ぐため、削除前に確認ダイアログが必要

## 3. 機能要件

### 3.1 認証・会員管理

ID	機能名	説明	優先度
AUTH-01	ユーザー登録	メールアドレス・パスワードによる会員登録	Must
AUTH-02	ログイン	JWT発行・Cookie保存	Must
AUTH-03	ログアウト	JWTの無効化・Cookie削除	Must
AUTH-04	パスワードリセット	メール経由でのパスワード再設定	Should
AUTH-05	プロフィール編集	氏名・住所・電話番号の更新	Must

### 3.2 商品管理

ID	機能名	説明	優先度
PROD-01	商品一覧表示	ページネーション・カテゴリフィルタ・並び替え	Must
PROD-02	商品詳細表示	画像・説明・価格・在庫状況の表示	Must
PROD-03	商品検索	キーワード検索（商品名・説明文）	Must
PROD-04	商品登録（管理）	名前・説明・価格・カテゴリ・画像の登録	Must
PROD-05	商品編集（管理）	既存商品情報の更新	Must
PROD-06	商品削除（管理）	論理削除（ソフトデリート）	Must
PROD-07	画像アップロード	S3へのアップロード、CloudFront経由で表示	Must

### 3.3 カート・注文

ID	機能名	説明	優先度
CART-01	カートに追加	商品・数量をカートへ追加	Must
CART-02	カート表示	カート内商品の一覧・合計金額表示	Must
CART-03	カート更新・削除	数量変更・商品削除	Must
ORD-01	注文確認画面	注文内容・配送先・金額の最終確認	Must
ORD-02	Stripe決済	クレジットカードによる決済処理	Must
ORD-03	注文完了通知	注文確定後にメール通知（将来対応）	Could
ORD-04	注文履歴	過去の注文一覧と詳細表示	Must
ORD-05	注文管理（管理）	注文ステータスの更新（未処理→発送済）	Must

## 4. 非機能要件

カテゴリ	要件	目標値・備考
パフォーマンス	ページ読み込み速度	初回表示 3秒以内 (LCP)
パフォーマンス	API応答時間	通常時 500ms以内
セキュリティ	認証	JWT (有効期限1時間) + Refresh Token
セキュリティ	パスワード管理	bcryptによるハッシュ化
セキュリティ	HTTPS	全通信をHTTPS化 (CloudFront+ACM)
セキュリティ	SQLインジェクション対策	Prisma ORM使用によりリスク低減
可用性	稼働率	95%以上 (個人開発のため)
スケーラビリティ	同時接続数	50ユーザー程度 (ポートフォリオ用)
保守性	コーディング規約	ESLint / Prettier 適用
保守性	テスト	主要APIにユニットテスト実装 (Jest)
アクセシビリティ	対応	基本的なWCAG 2.1 Level A対応

## 5. 画面一覧

### 5.1 顧客向け画面

画面ID	画面名	URL	対象ロール
SCR-01	トップページ	/	全ユーザー
SCR-02	商品一覧	/products	全ユーザー
SCR-03	商品詳細	/products/[id]	全ユーザー
SCR-04	カート	/cart	会員
SCR-05	注文確認	/checkout	会員
SCR-06	注文完了	/checkout/complete	会員
SCR-07	注文履歴	/mypage/orders	会員
SCR-08	ログイン	/login	ゲスト
SCR-09	新規登録	/register	ゲスト
SCR-10	プロフィール	/mypage/profile	会員

### 5.2 管理者向け画面

画面ID	画面名	URL	対象ロール
ADM-01	管理ダッシュボード	/admin	管理者
ADM-02	商品管理一覧	/admin/products	管理者
ADM-03	商品登録・編集	/admin/products/[id]	管理者
ADM-04	注文管理一覧	/admin/orders	管理者
ADM-05	注文詳細	/admin/orders/[id]	管理者
ADM-06	ユーザー管理	/admin/users	管理者

## 6. 主要ユースケース

### UC-01:商品を購入する

項目	内容
アクター	一般会員
事前条件	ログイン済み・カートに1件以上の商品がある
基本フロー	1. カート画面で注文内容を確認 2. 「注文確認」ボタンをクリック 3. 配送先・支払い方法を確認 4. Stripeにてクレジットカード決済 5. 注文完了画面が表示される
代替フロー	決済失敗 → エラーメッセージを表示しカートに戻る
事後条件	注文レコードが作成され、在庫数が減少する

### UC-02:管理者が商品を登録する

項目	内容
アクター	管理者
事前条件	管理者としてログイン済み
基本フロー	1. 商品管理一覧画面を開く 2. 「新規登録」ボタンをクリック 3. 商品名・説明・価格・カテゴリ・在庫数を入力 4. 商品画像をアップロード（S3へ保存） 5. 「保存」ボタンで登録完了
代替フロー	必須項目未入力 → バリデーションエラー表示
事後条件	新商品がDBに保存され、商品一覧に表示される

## 7. 制約条件・前提条件

### 7.1 技術的制約

カテゴリ	内容
フロントエンド	Next.js 14 (App Router) + TypeScript
バックエンド	NestJS+ TypeScript
ORM	Prisma (PostgreSQL)
クラウド	AWS (EC2, RDS, S3, CloudFront)
決済	Stripe (テストモード)
認証方式	JWT (アクセストークン+リフレッシュトークン)
デプロイ	フロントエンド: Vercel バックエンド: EC2
CI/CD	GitHub Actions

### 7.2 その他の制約

- ・本システムはポートフォリオ用であり、実際の商取引には使用しない
- ・Stripeはテストモードのみで実装し、実際の課金は行わない
- ・AWSの無料枠を活用するため、高負荷テストは行わない
- ・個人情報の取り扱いはポートフォリオ用の仮データのみ使用

## 8. 改訂履歴

バージョン	日付	変更者	変更内容
1.0	2026/02	Wai Wai	初版作成