# プロジェクト引き継ぎドキュメント

## プロジェクト概要

**プロジェクト名**: 案件番号逆引きシステム  
**URL**: <https://project-order-number-reverse-csharp.onrender.com>  
**技術スタック**: ASP.NET Core 8.0, PostgreSQL, Render.com

## アクセス情報

### 1. GitHubリポジトリ

* **リポジトリURL**: <https://github.com/3dv-development/project-Order-number-reverse_csharp>
* **ブランチ**: main
* **アクセス**: 3dv-development組織のメンバーは閲覧・編集可能

### 2. Render.com

* **Webサービス名**: project-order-number-reverse-csharp
* **データベース名**: project-db
* **アクセス権限の追加方法**:
  1. Render.comアカウント所有者が各メンバーをチームに招待
  2. Dashboard → Settings → Team Members → Invite

### 3. 管理者アカウント

* **社員番号**: 2024
* **パスワード**: 2024（初回ログイン後、変更推奨）
* **役割**: admin（新規ユーザー登録が可能）

## システム構成

### インフラ

* **Webサーバー**: Render.com Web Service (無料プラン)
* **データベース**: Render.com PostgreSQL (無料プラン)
* **自動デプロイ**: GitHubのmainブランチにpushすると自動デプロイ

### 環境変数 (Render.com)

Webサービスの Environment タブで設定されている環境変数:

* DATABASE\_URL: PostgreSQLの接続文字列（自動設定）

## 主な機能

1. **ログイン・認証**
   * 社員番号とパスワードでログイン
   * パスワードは社員番号と同じ（簡易認証）
   * セッションタイムアウト: 8時間
2. **プロジェクト管理**
   * プロジェクト番号の検索・登録
   * プロジェクト情報の編集
   * 編集履歴の記録
3. **ユーザー管理** (管理者のみ)
   * 新規社員アカウントの作成
   * ユーザー一覧の表示

## ローカル開発環境のセットアップ

### 前提条件

* .NET 8.0 SDK
* PostgreSQL 15以上
* Git

### セットアップ手順

1. **リポジトリのクローン**
2. git clone https://github.com/3dv-development/project-Order-number-reverse\_csharp.git

cd project-Order-number-reverse\_csharp

1. **PostgreSQLデータベースの作成**
2. CREATE DATABASE project\_numbers;
3. CREATE USER postgres WITH PASSWORD 'postgres';

GRANT ALL PRIVILEGES ON DATABASE project\_numbers TO postgres;

1. **接続文字列の確認**  
   appsettings.json の DefaultConnection を確認:
2. "ConnectionStrings": {
3. "DefaultConnection": "Host=localhost;Database=project\_numbers;Username=postgres;Password=postgres"

}

1. **マイグレーションの適用**

dotnet ef database update

1. **アプリケーションの起動**

dotnet run

ブラウザで <http://localhost:5000> にアクセス

1. **初回起動時**
   * 管理者ユーザー(2024)が自動作成されます
   * 社員番号: 2024, パスワード: 2024 でログイン

## デプロイ方法

### 自動デプロイ（推奨）

1. ローカルで変更を加える
2. Gitコミット＆プッシュ
3. git add .
4. git commit -m "変更内容"

git push origin main

1. Render.comが自動的にデプロイを開始
2. Render.com Dashboard → Logs でデプロイ状況を確認

### 手動デプロイ

1. Render.com Dashboard にアクセス
2. Webサービス project-order-number-reverse-csharp を選択
3. "Manual Deploy" → "Deploy latest commit" をクリック

## トラブルシューティング

### データベース接続エラー

**エラー**: "データベースに接続できません"

**原因と対策**:

1. **環境変数の確認**
   * Render.com → Web Service → Environment
   * DATABASE\_URL が設定されているか確認
2. **データベースの稼働確認**
   * Render.com → PostgreSQL → project-db
   * Statusが "Available" になっているか確認
3. **ログの確認**
   * Render.com → Web Service → Logs
   * [ERROR] で始まる行を確認

### Antiforgeryトークンエラー

**エラー**: "The antiforgery token could not be decrypted"

**対策**: ブラウザのCookieをクリアしてログインし直す

### アプリケーションが起動しない

**対策**:

1. Render.com Logs で起動時のエラーを確認
2. ローカルで dotnet build が成功するか確認
3. 環境変数が正しく設定されているか確認

## データベーススキーマ

### employees テーブル

| **カラム名** | **型** | **説明** |
| --- | --- | --- |
| id | integer | 主キー（自動採番） |
| employee\_id | varchar(50) | 社員番号（ユニーク） |
| name | varchar(100) | 氏名 |
| email | varchar(200) | メールアドレス |
| is\_active | boolean | アクティブフラグ |
| role | varchar(20) | 役割 (admin/user) |

### projects テーブル

| **カラム名** | **型** | **説明** |
| --- | --- | --- |
| id | integer | 主キー（自動採番） |
| project\_number | varchar(50) | プロジェクト番号（ユニーク） |
| title | varchar(200) | プロジェクト名 |
| client\_name | varchar(200) | クライアント名 |
| status | varchar(20) | ステータス |
| created\_at | timestamp | 作成日時 |
| updated\_at | timestamp | 更新日時 |
| created\_by | varchar(50) | 作成者の社員番号 |
| updated\_by | varchar(50) | 更新者の社員番号 |

### edit\_histories テーブル

| **カラム名** | **型** | **説明** |
| --- | --- | --- |
| id | integer | 主キー（自動採番） |
| project\_id | integer | プロジェクトID（外部キー） |
| edited\_by | varchar(50) | 編集者の社員番号 |
| edited\_at | timestamp | 編集日時 |
| changes | text | 変更内容（JSON） |

## 連絡先・サポート

### リポジトリ管理

* Organization: 3dv-development
* Repository: project-Order-number-reverse\_csharp

### 問題報告

GitHubのIssuesを使用してください:  
<https://github.com/3dv-development/project-Order-number-reverse_csharp/issues>

## 今後の改善案

1. **セキュリティ強化**
   * パスワードのハッシュ化（現在は社員番号=パスワード）
   * HTTPS証明書の設定
   * ログイン試行回数制限
2. **機能追加**
   * プロジェクト検索のフィルター機能
   * CSVエクスポート機能
   * メール通知機能（現在は無効化中）
3. **インフラ改善**
   * 有料プランへの移行（パフォーマンス向上）
   * バックアップの自動化
   * モニタリング・アラート設定

**作成日**: 2025年10月16日  
**作成者**: 長谷部