

連絡帳管理システム（School Journal）アプリケーションマニュアル

目次

1. アプリケーション概要.....	1
2. アプリケーションの利用マニュアル.....	2
3. 設計時に作成した ER 図（Firestore 版）.....	3
4. アプリケーションのデプロイ先・デプロイ方法.....	4
5. テストアカウント一覧.....	5
6. 参考.....	6

1. アプリケーション概要

本アプリケーション「School Journal（連絡帳管理システム）」は、架空の私立中学校における連絡帳提出業務をデジタル化し、生徒・担任・管理者の三者間で日々の記録・確認・分析を効率化することを目的として開発した。従来の紙運用では、提出漏れや既読確認作業、記録のばらつきなどが課題であったが、本システムではそれらを解消し、クラウド上で安全かつ迅速に情報共有が行える。

本アプリケーションは Node.js + Express を基盤に、EJS による動的描画を採用し、Firebase Firestore をデータベースとして構築されたクラウド対応システムである。フロントエンドでは Chart.js によるグラフ描画、html2pdf.js による PDF 出力を行い、Render 上で自動デプロイ運用可能な構成となっている。

2. アプリケーションの利用マニュアル

■ ログイン方法

1. <https://school-journal-imwb.onrender.com/login> にアクセス
2. ID とパスワードを入力してログイン

■ 生徒ダッシュボード

- ・連絡帳を登録・提出・訂正
- ・グラフで体調・メンタルを確認
- ・PDF/CSV 出力可
- ・1 日 1 回のみ提出可能（再提出は訂正扱い）

■ 担任ダッシュボード

- ・提出一覧と未提出者を確認
- ・グラフ表示、PDF/CSV 出力、既読処理可能
- ・生徒の状態推移を時系列で確認可

■ 管理者ダッシュボード

- ・アカウント追加/削除、初期化、ログ管理
- ・グラフ表示、PDF/CSV 出力対応

■ 共通機能

- ・ログイン状態はセッションで保持され、再読み込み時も維持
- ・すべてのロールで PDF/CSV 出力が可能
- ・各操作はログとして Firestore に自動保存

■ マニュアルダウンロード機能

ログイン画面下部の「 使い方マニュアルをダウンロード」から直接入手可能。

3. 設計時に作成した ER 図 (Firestore 版)

テーブル名	カラム名	データ型	主キー	外部キー	説明
students	student_id	STRING	<input checked="" type="checkbox"/>		生徒 ID (ユニーク)
students	name	STRING			生徒名
students	grade	NUMBER			学年 (1~3)
students	class_name	STRING			クラス名 (A/B/C など)
students	password	STRING			ログイン用 パスワード
teachers	teacher_id	STRING	<input checked="" type="checkbox"/>		担任 ID (ユニーク)
teachers	name	STRING			担任氏名
teachers	class_name	STRING			担当クラス (例 : 1A, 2B など)
teachers	email	STRING			連絡用メー ルアドレス
teachers	password	STRING			ログイン用 パスワード
entries	entry_id	STRING	<input checked="" type="checkbox"/>	student_id → students.student_id	連絡帳記録 ID
entries	student_id	STRING	<input checked="" type="checkbox"/>		記録した生 徒の ID
entries	date	STRING			記録日

(YYYY-
MM-DD)

entries	condition	NUMBER			体調（数値評価）
entries	mental	NUMBER			メンタル（数値評価）
entries	reflection	STRING			1日の振り返り内容
entries	status	STRING			既読 / 未読ステータス
logs	log_id	STRING	<input checked="" type="checkbox"/>	user_id → (students / teachers).id	ログ ID
logs	user_id	STRING	<input checked="" type="checkbox"/>		操作したユーザーの ID
logs	action	STRING			操作内容 (例：ログイン、提出、既読など)
logs	timestamp	STRING			実行時刻 (JST)

【リレーション概要】

- students（生徒）と entries（連絡帳）は1対多の関係
- teachers（担任）と students（生徒）は1対多の関係
- logs（操作ログ）は全ユーザー共通の履歴を時系列で管理

4. アプリケーションのデプロイ先・デプロイ方法

【デプロイ先】Render（Node.js Web Service）

URL : <https://school-journal-heqt.onrender.com>

リージョン : Singapore

ランタイム : Node18+

ビルド : npm install

スタート : npm start

【手順】

1. GitLab に push
2. Render で新規 Web Service 作成
3. main ブランチを指定
4. 自動デプロイ ON

【環境設定】

- Node.js : v18.0.0
- パッケージ管理 : npm
- 環境変数 : FIREBASE_API_KEY, FIREBASE_PROJECT_ID (Firestore 接続に必須)

5. テストアカウント一覧

ロール	ユーザーID	パスワード	権限レベル	備考
管理者	admin	1234	全権限	全データ閲 覧・初期化可 能
担任	t3C	1234	クラス管理	1年A組担任
生徒	s3C01	1234	一般ユーザー	3年C組生徒

6. 参考

- Firebase Firestore : データベースとして使用
- Chart.js : グラフ描画
- html2pdf.js : PDF 出力機能
- Express / EJS : サーバーサイド構成
- Node.js : バックエンドランタイム
- Render : クラウドデプロイ環境