

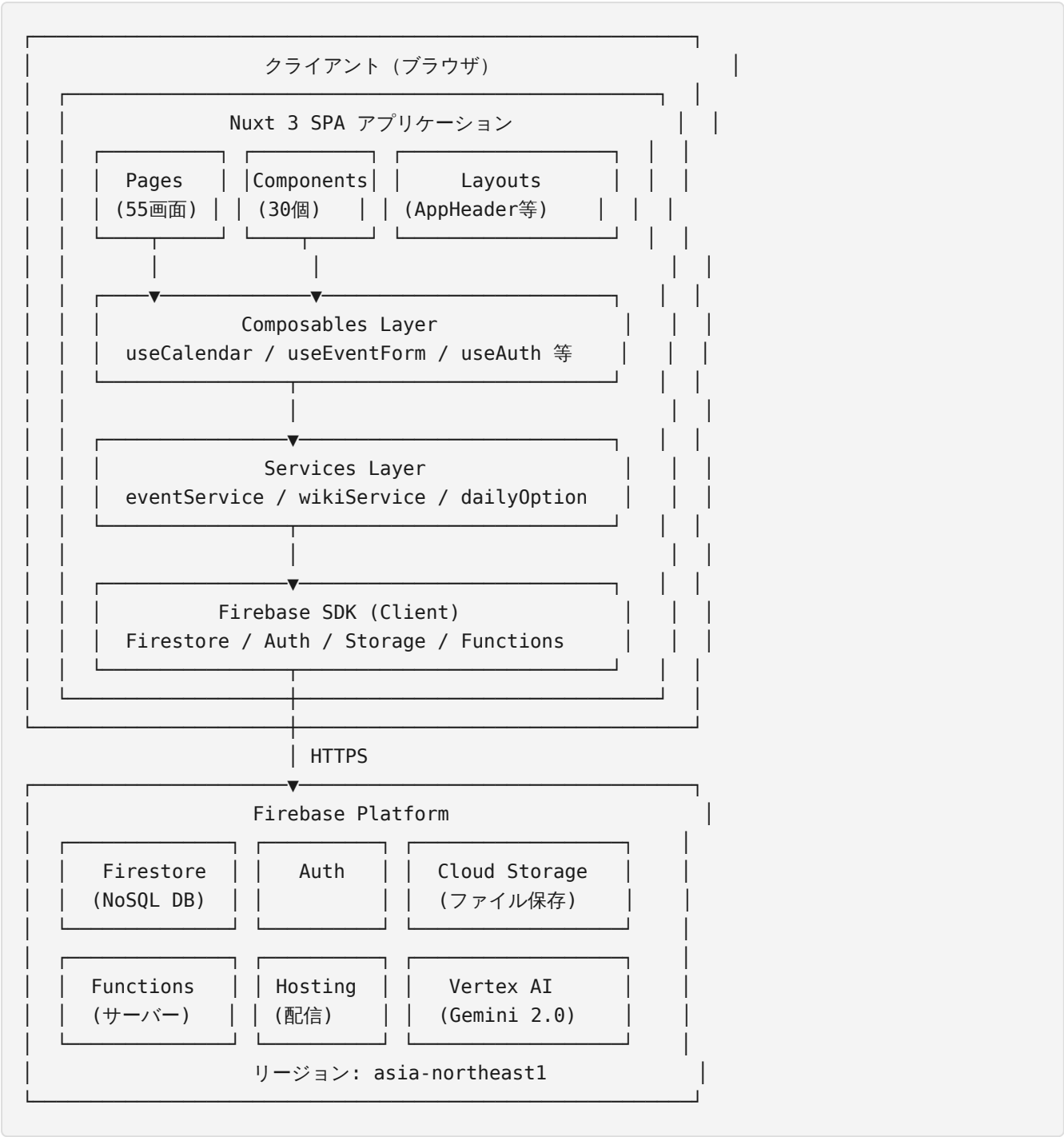
基本設計書

Tascal（タスカル） - 社員タスク管理・カレンダーアプリケーション

項目	内容
ドキュメントID	BD-001
バージョン	1.0
作成日	2026-02-26
プロジェクト名	SVN Dashboard App (Tascal)
ステータス	初版

1. システム全体構成

1.1 システム構成図



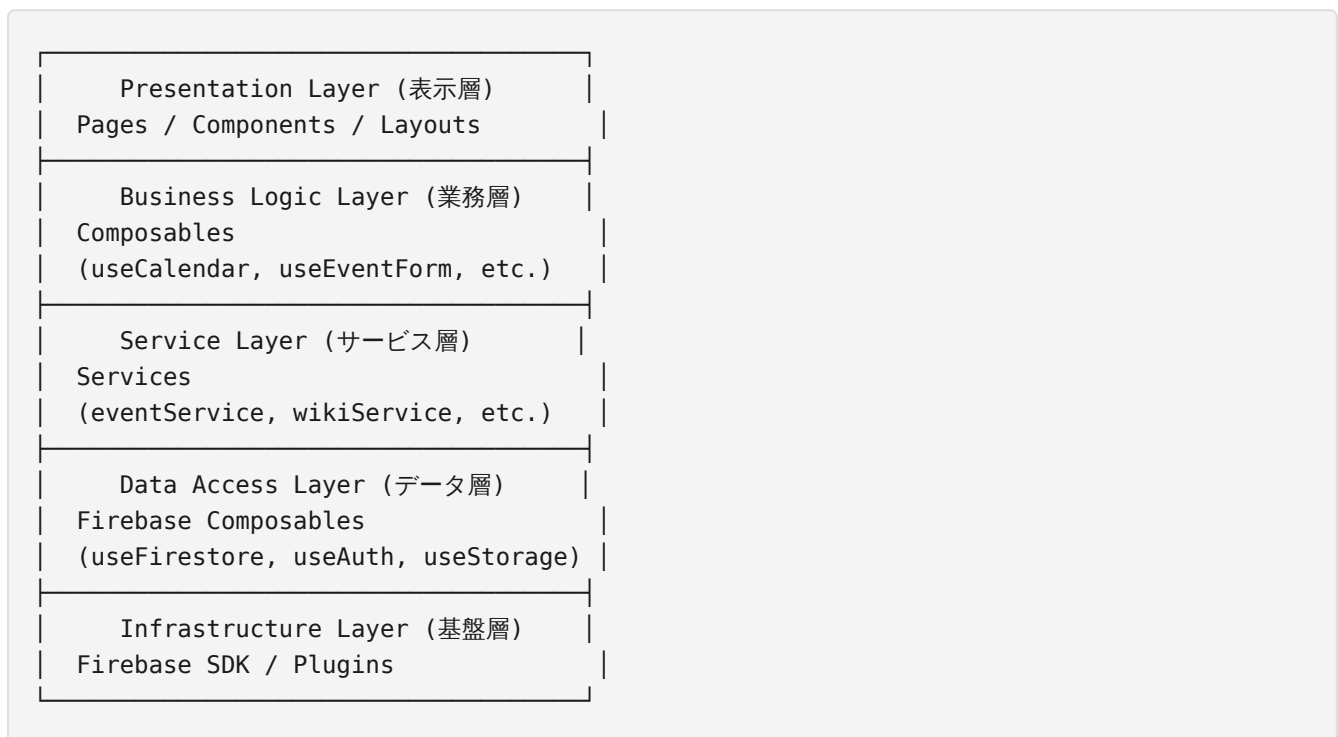
1.2 技術スタック

カテゴリ	技術	バージョン	用途
フレームワーク	Nuxt 3	3.17.3	アプリケーションフレームワーク

カテゴリ	技術	バージョン	用途
UIフレームワーク	Vue 3	3.5.13	リアクティブUIライブラリ
UIコンポーネント	Vuetify	3.8.4	Material Designコンポーネント
言語	TypeScript	-	型安全な開発言語
BaaS	Firebase	11.8.1	バックエンドサービス基盤
チャート	Chart.js	4.4.9	データ可視化
Markdown	Marked	16.1.1	Markdownパーサー
HTMLサニタイズ	DOMPurify	3.2.6	XSS対策
Markdownエディタ	EasyMDE	2.20.0	Wiki記事編集
日付処理	Moment-timezone	0.6.0	タイムゾーン対応日付処理
CSV処理	PapaParse	5.5.3	CSV入出力
ビルドツール	Vite	-	高速ビルド・HMR

2. アプリケーションアーキテクチャ

2.1 レイヤー構成



2.2 ディレクトリ構成

```
svn-dashboard-app/
├─ pages/                # 画面コンポーネント (ルーティング)
│  ├─ calendar/          # カレンダー管理
│  ├─ users/             # ユーザー管理
│  ├─ facility/          # 施設管理
│  ├─ equipment/         # 設備管理
│  ├─ team/              # チーム管理
│  ├─ section/           # 部署管理
│  ├─ wiki/              # Wikiナレッジベース
│  ├─ attendance/        # 勤怠管理
│  ├─ lunch/             # ランチシャッフル
│  ├─ sales/             # 売上管理
│  ├─ expense/           # 経費管理
│  ├─ holiday/           # 祝日管理
│  ├─ profile/           # プロフィール
│  ├─ auth/              # 認証関連
│  └─ ...                # その他
├─ components/           # 再利用コンポーネント
│  ├─ Calendar/          # カレンダー専用コンポーネント
│  │  ├─ DailyView/      # 日別ビュー
│  │  ├─ WeeklyView/     # 週別ビュー
│  │  └─ MonthlyView/    # 月別ビュー
│  ├─ AppHeader.vue      # ヘッダー
│  ├─ SideBar.vue        # サイドバー
│  └─ MarkdownRenderer.vue # Markdownレンダラー
├─ composables/          # ビジネスロジック
│  ├─ useCalendar.ts     # カレンダーロジック
│  ├─ useEventForm.ts    # イベントフォームロジック
│  ├─ useUserProfile.ts  # ユーザープロフィール
│  ├─ useCsvFirestore.ts # CSV入出力
│  ├─ useDailyOptions.ts # 日別オプション
│  ├─ useLunchShuffle.ts # ランチシャッフル
│  ├─ useWiki.ts         # Wiki
│  ├─ firebase/          # Firebase操作
│  ├─ firestoreGeneral/  # 汎用Firestore操作
│  ├─ master/            # マスターデータ
│  ├─ transaction/       # トランザクション
│  └─ common/            # 共通ユーティリティ
├─ services/             # ドメインサービス
│  ├─ eventService.ts    # イベントCRUD
│  ├─ wikiService.ts     # Wiki記事管理
│  └─ dailyOptionService.ts # 日別オプション
├─ types/                # TypeScript型定義
├─ plugins/              # Nuxtプラグイン
│  ├─ firebase.client.ts # Firebase初期化
│  ├─ vuetify.ts         # Vuetify初期化
│  ├─ chart.ts           # Chart.js初期化
│  └─ initState.ts       # 状態初期化
├─ middleware/           # ルートミドルウェア
│  └─ router.global.ts   # 認証チェック
```

```
└─ layouts/           # レイアウト
  └─ default.vue      # デフォルトレイアウト
└─ assets/            # 静的アセット
└─ public/            # パブリックファイル
└─ functions/         # Cloud Functions
└─ server/            # サーバーユーティリティ
└─ docs/              # ドキュメント
```

3. 画面一覧・画面遷移

3.1 画面一覧

No	画面ID	画面名	パス	概要
1	SCR-001	ダッシュボード	/	トップページ
2	SCR-002	サインイン	/signin	ログイン画面
3	SCR-010	カレンダー	/calendar	カレンダービュー（日/週/月）
4	SCR-011	予定登録	/calendar/register	新規予定登録
5	SCR-012	予定詳細	/calendar/[id]	予定詳細表示
6	SCR-013	予定編集	/calendar/[id]/edit	予定編集
7	SCR-020	ユーザー一覧	/users	ユーザー一覧
8	SCR-021	ユーザー新規登録	/users/new	ユーザー登録
9	SCR-022	ユーザー詳細	/users/[id]	ユーザー詳細
10	SCR-023	ユーザー編集	/users/[id]/edit	ユーザー編集
11	SCR-030	施設一覧	/facility	施設一覧
12	SCR-031	施設新規登録	/facility/new	施設登録
13	SCR-032	施設詳細	/facility/[id]	施設詳細
14	SCR-033	施設編集	/facility/[id]/edit	施設編集
15	SCR-040	設備一覧	/equipment	設備一覧
16	SCR-041	設備新規登録	/equipment/new	設備登録
17	SCR-042	設備詳細	/equipment/[id]	設備詳細
18	SCR-043	設備編集	/equipment/[id]/edit	設備編集
19	SCR-050	チーム一覧	/team	チーム一覧

No	画面ID	画面名	パス	概要
20	SCR-051	チーム新規登録	/team/new	チーム登録
21	SCR-052	チーム詳細	/team/[id]	チーム詳細
22	SCR-053	チーム編集	/team/[id]/edit	チーム編集
23	SCR-060	部署一覧	/section	部署一覧
24	SCR-061	部署新規登録	/section/new	部署登録
25	SCR-062	部署詳細	/section/[id]	部署詳細
26	SCR-063	部署編集	/section/[id]/edit	部署編集
27	SCR-070	Wikiトップ	/wiki	Wikiホーム
28	SCR-071	記事一覧	/wiki/articles	記事一覧
29	SCR-072	記事新規作成	/wiki/new	記事作成
30	SCR-073	記事詳細	/wiki/[id]	記事表示
31	SCR-074	記事編集	/wiki/[id]/edit	記事編集
32	SCR-075	カテゴリ別表示	/wiki/category	カテゴリフィルター
33	SCR-076	タグ別表示	/wiki/tag	タグフィルター
34	SCR-080	勤怠管理	/attendance	勤怠・日別オプション
35	SCR-090	ランチシャッフル	/lunch	ランチシャッフル
36	SCR-100	祝日管理	/holiday	祝日管理
37	SCR-110	経費管理	/expense	経費管理
38	SCR-120	売上管理	/sales	売上ダッシュボード
39	SCR-130	プロフィール	/profile	自分のプロフィール
40	SCR-131	プロフィール編集	/profile/edit	プロフィール編集
41	SCR-140	設定	/setting	アプリ設定
42	SCR-150	ヘルプ	/help	ヘルプ
43	SCR-160	タスク管理	/task	タスク管理
44	SCR-170	メンテナンス	/maintenance	メンテナンス
45	SCR-180	マニュアル	/manuals	マニュアル一覧
46	SCR-190	パスワードリセット	/auth/password/reset	パスワードリセット

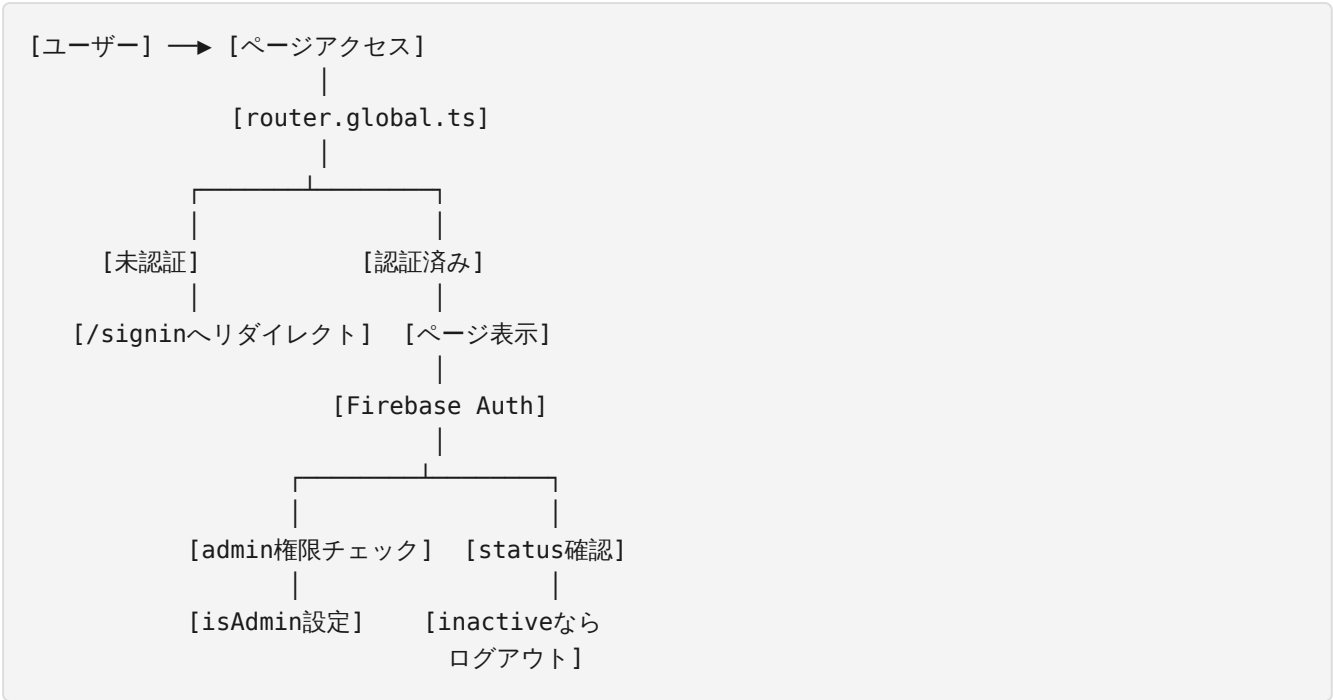
No	画面ID	画面名	パス	概要
47	SCR-191	メールリセット	/auth/email/reset	メール変更

3.2 画面遷移図



4. 共通機能設計

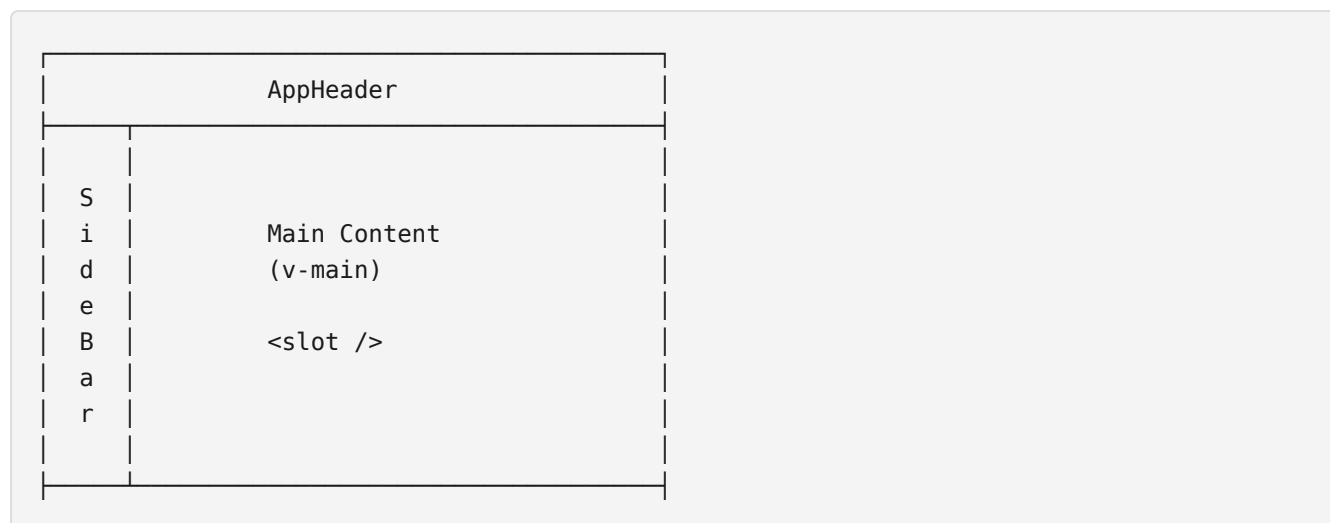
4.1 認証・認可フロー



4.2 データキャッシュ戦略

キャッシュ対象	TTL	保存先	更新契機
マスターデータ（ユーザー、祝日、施設、設備）	10時間	Nuxt useState	アプリ起動時、手動リフレッシュ
日別オプション	1時間（日付範囲別）	Nuxt useState	画面表示時
イベントデータ	都度取得	メモリ内キャッシュ	カレンダー操作時
カレンダー表示位置	永続	localStorage	ビュー変更時
カレンダービュー種別	永続	localStorage	ビュー切替時

4.3 レイアウト構成



- **AppHeader**: アプリケーションヘッダー（ナビゲーション、ユーザーメニュー）
- **SideBar**: サイドバーメニュー（折りたたみ可能）
- **v-main**: ページコンテンツ領域

5. カレンダー機能設計

5.1 ビュー別データ取得方式

ビュー	データ取得関数	取得範囲	表示対象
週別 (weekly)	<code>getEventsInRange()</code>	表示週の全イベント	全ユーザー
月別 (monthly)	<code>getEventsByParticipantInRange()</code>	表示月の自分のイベント	ログインユーザーのみ
日別 (daily)	<code>getEventsByParticipantInRange()</code>	表示日の自分のイベント	ログインユーザーのみ

5.2 イベントの種類と処理

DateType	種別	保存方式	削除オプション
single	単日予定	eventsコレクションに1件	単一削除のみ
range	範囲指定予定	masterId付きで日ごとに複数件	single / all / after / before
recurring	繰り返し予定	masterId付きで展開日ごとに複数件	single / all / after / before

5.3 コンフリクト検出

予定登録時に以下のコンフリクトを検出する。

- 参加者の時間重複
- 施設の時間重複
- 設備の時間重複

検出ロジック:

1. 登録対象の日付範囲を算出
2. 対象日のタイムスロットを生成 (繰り返し予定は3ヶ月分)
3. 既存イベントと時間帯の重複をチェック
4. `isTimeOverlapping(start1, end1, start2, end2)` で判定

6. マスター管理機能設計

6.1 CRUD共通パターン

各マスター管理（ユーザー、施設、設備、チーム、部署）は以下の共通パターンに従う。

操作	パス	メソッド	処理
一覧	<code>/resource</code>	GET	コレクション全件取得
新規登録	<code>/resource/new</code>	POST	ドキュメント追加
詳細	<code>/resource/[id]</code>	GET	ドキュメント取得
編集	<code>/resource/[id]/edit</code>	PUT	ドキュメント更新

6.2 共通フィールド

フィールド	型	説明	自動設定
<code>createdAt</code>	Timestamp	作成日時	はい
<code>updatedAt</code>	Timestamp	更新日時	はい
<code>createdBy</code>	string	作成者UID	はい
<code>updatedBy</code>	string	更新者UID	はい

7. 外部インターフェース

7.1 Firebase SDK

サービス	用途	主要メソッド
Firestore	データ永続化	getDoc, addDoc, updateDoc, deleteDoc, onSnapshot
Authentication	ユーザー認証	signInWithEmailAndPassword, createUserWithEmailAndPassword
Cloud Storage	ファイル保存	uploadBytes, getDownloadURL, deleteObject
Cloud Functions	サーバー処理	httpsCallable
Vertex AI	AI機能	Gemini 2.0 Flash モデル

7.2 キャッシュファイル (Firebase Storage)

イベントデータのキャッシュファイルをFirebase Storageに保存し、高速な読み込みを実現する。

- パス: `calendar-cache/{isoWeekYear}-{isoWeek}-cache.json`
- 形式: JSON
- 取得方法: HTTPフェッチ (`cache: 'no-store'`)

8. エラーハンドリング方針

レベル	対応方針	例
致命的	エラー画面表示	Firebase接続不可
業務エラー	トースト通知・フォームエラー表示	バリデーションエラー、コンフリクト
警告	コンソール出力・キャッシュフォールバック	ネットワーク一時エラー
情報	コンソールログ (開発時のみ)	クエリプロファイル情報