ANSHIN ROOM

System Architecture

障害福祉サービスにおける、利用者と支援員のコミュニケーションを補助するWebアプリケーション。
AIは判断せず、利用者が自分の言葉で伝えられるように「整理」だけを行う設計方針です。

Next.js 16 · React 19 · TypeScript · Firebase · OpenAI Responses API · WebRTC · Cloudflare TURN

SYSTEM ARCHITECTURE

01 — クライアント（画面・機能）

- `app/` ログイン / 新規登録（Firebase Authentication・日本語UI）
- `dashboard/` ちぇっくルーム — カメラ・マイク確認、待機一覧
- `room/` つながルーム — ビデオ通話・チャット・AI補助パネル・交渉パネル
- `live/` ライブ確認 — スマホ映像をQRコードで支援員PCへ一方向共有
- `daily-report/` 日報 — 入力内容をFirestoreへ保存、カレンダー表示
- `admin/` 管理画面 — 利用状況・ユーザー管理（PIN認証）
- `demo/` デモモード — Firebase不要のポートフォリオ用体験版

React 19 · CSS Modules · MediaPipe Tasks Vision（人物モザイク） · qrcode

02 — API（Next.js Route Handlers）

ROUTE	役割
`turn-credentials`	要ログイン。Cloudflare TURN資格情報を発行
`ai-assist`	OpenAI Responses APIへ中継、利用上限を管理
`member-id`	会員番号（U-/S-）をトランザクション採番
`admin/verify-pin`	管理者PINを検証、署名付きセッションを発行
`admin/usage, users`	admin権限のみ。利用状況・ユーザー情報を取得
`diagnostics/unlock`	通話診断ツールのロック解除
`health`	死活監視用エンドポイント

03 — 認証・データ（Firebase）

Authentication — クライアントSDKでログイン・新規登録。エラー表示は日本語化。

IDトークン検証（独自実装） — firebase-admin/authがこの実行環境で読み込めないため、RS256のJWTをGoogleの公開鍵で自前検証。アカウント作成もIdentity Toolkit REST APIを自己署名JWTで直接呼び出す。

Firestore — クライアントは onSnapshot でリアルタイム購読、サーバーは firebase-admin/firestore でトランザクション処理（採番・通話ログ）。

Storage — プロフィール画像などのファイル保存。

04 — AI補助（OpenAI連携）

OpenAI Responses API を呼び出し、役割（利用者・支援員・管理者）とメニューごとに専用プロンプトを組み立てる。出力はJSON限定。

主なメニュー — 5W1H抽出／KPTこころ整理シート／伝わりやすい言葉にする／質問の意味を分かりやすくする

利用上限 — 1日15回・入力1,000文字まで、月間400万トークンまで。Firestore（aiUsage / aiUsageMonthly）で使用量を管理。

設計方針：AIは判断しない。AIは整理する。支援員が判断し、支援する。

REALTIME COMMUNICATION

利用者クライアント
MediaPipeで選択した人物にのみモザイク（映像は保存しない）

Offer / Answer / ICE ⇄ `calls/{callId}`

支援員クライアント
チャット・AI補助パネル・交渉パネルを同時表示

↕ ICE候補はFirestoreのリアルタイム更新（onSnapshot）で交換 → 接続確立後は `Cloudflare Realtime TURN` がNAT越えを中継（/api/turn-credentials で都度発行）

スマホのカメラ
QRコードを読み取ってライブ確認セッションに接続

一方向・5分で自動終了 → `liveSessions/{sessionId}`

支援員のPC
通話（calls）とは別の場所を使い、通話側には影響しない

通話終了後、通話時間は `callSessions/{sessionId}` に記録（料金計算は行わない）

DATA FLOW

1. Firebase Authenticationでログイン。IDトークンをサーバーが独自検証
2. ちぇっくルームでカメラ・マイクを確認し、必要ならモザイク設定
3. つながルーム入室時に /api/turn-credentials でTURN資格情報を取得
4. Firestore（calls）でOffer/Answer/ICEを交換しWebRTC接続を確立
5. AI補助パネルから /api/ai-assist を呼び出し、上限内でOpenAIが整理を提案
6. 通話終了→callSessionsへ記録、支援員が日報をFirestoreへ保存

TECH STACK

FRONTEND
Next.js 16
React 19
TypeScript
CSS Modules

BACKEND
Next.js Route Handlers
firebase-admin（Firestore）
独自JWT検証

REALTIME & AI
WebRTC
Cloudflare Realtime TURN
OpenAI Responses API
MediaPipe Tasks Vision

INFRA
Caddy（リバースプロキシ）
DuckDNS
セルフホスト
Firebase Auth / Firestore / Storage

DEPLOY
Caddy Reverse Proxy → anshin-connect.duckdns.org（セルフホスト）

DESIGN POLICY
AIは判断しない。AIは整理する。支援員が判断し、支援する。