

CLAUDE.md

This file provides guidance to Claude Code (claude.ai/code) when working with code in this repository.

Repository Overview

This is the **Selby Rugby App** repository - a Progressive Web App (PWA) that replaces traditional paper match day programs for Selby RUFC. Spectators scan QR codes at ground entrance to access real-time team sheets, player profiles with sponsor integration, club updates, and an engaging mobile experience.

Full PRD: See </docs/selby-rugby-app-prd.md> for complete product requirements, user stories, and technical specifications.

Project Summary

- **Purpose:** Digital match day program for 22 teams (men's, ladies, juniors)
- **Primary Users:** Match day spectators, club members, parents/families, sponsors
- **Key Features:**
 - Team sheets with last-minute updates
 - Player profiles with sponsor integration
 - QR code access for instant mobile viewing
 - Non-technical CMS for club staff
 - Sponsor carousel and full-page advertisements

Technology Stack

Frontend

- **Framework:** Next.js 14+ (App Router)
- **Language:** TypeScript, React
- **Styling:** Tailwind CSS, Shadcn UI components
- **Icons:** Lucide React
- **Notifications:** Sonner (toast notifications)
- **PWA:** Progressive Web App capabilities (installable, offline support)

Backend/CMS

- **CMS:** Payload CMS (self-hosted, non-technical editor interface)
- **Database:** MongoDB or PostgreSQL (TBD)
- **Media Storage:** Vercel Blob or Cloudinary for sponsor images

Hosting

- **Frontend:** Vercel
- **CMS/Backend:** Vercel, Railway, or DigitalOcean (TBD)

Future (Phase 2)

- **Authentication:** Better-Auth or Clerk
- **Payments:** Stripe

Current State

- **Stage:** Planning/PRD Complete
- Repository initialized from Specify template (original template files removed)
- PRD document finalized at </docs/selby-rugby-app-prd.md>
- No code implementation yet - ready for initial setup
- Staged deletions (template files) ready to be committed

Design System

Color Palette

- **Primary Green:** #1a5f3f (header, navigation, primary elements)
- **Accent Red/Burgundy:** #9b2c2c (section headers, CTAs)
- **Gold/Yellow:** #f4a613 (logo accents, highlights, active states)
- **Cream/Beige:** #f5f1e8 (content card backgrounds)
- **Dark Gray:** Text on light backgrounds
- **White:** Text on dark backgrounds

Key UI Components

- Bottom tab navigation (5 tabs: Home | Team | Live | Sponsors | Club)
- Top header with Selby RUFC logo and menu
- Match day hero card (red background)
- Player card modals (circular photos, sponsor logos)
- Sponsor carousel (horizontal scroll)

Design Reference: </selby-rufc-design-idea-and-club-colours.jpg> and [.png](#)

Data Model (Key Collections)

- **Teams:** team_id, name, category, age_group, logo
- **Players:** player_id, team_id, name, number, position, photo_url, bio, stats, social_links, sponsor_id
- **Matches:** match_id, team_id, opponent, kick_off_time, ground_info, weather, team_sheet, status
- **Sponsors:** sponsor_id, name, logo_url, ad_jpeg_url, type (club/player), linked_player_id
- **Content:** content_id, slug, title, body (rich text), section, published_date

Development Workflow

To be established - Once project structure is initialized, this section will include:

- `npm run dev` - Start development server
- `npm run build` - Production build
- `npm run lint` - Code linting
- `npm run type-check` - TypeScript validation
- Payload CMS local development commands

Git Workflow

- Current branch: **main**
- No main/master branch distinction currently configured.
- Staged deletions (template files) ready for initial commit

Architecture Principles

1. **Mobile-First:** Primary viewport 320px-768px
2. **Low Maintenance:** Self-hosted, minimal ongoing costs
3. **Non-Technical Friendly:** CMS must be usable by club staff without technical support
4. **Performance:** Fast loading, optimized images, offline support via PWA
5. **Accessibility:** WCAG AA minimum, screen reader compatible

Key Constraints

- Low budget for ongoing maintenance/hosting
- Must work without app store installation (PWA)
- Non-technical editors must independently manage content
- Reliable mobile browser compatibility required

Next Steps (Initial Setup)

1. Initialize Next.js 14+ project with TypeScript and App Router
2. Set up Tailwind CSS and Shadcn UI
3. Configure Payload CMS with chosen database
4. Implement core data models (Teams, Players, Matches, Sponsors, Content)
5. Build MVP features: Match day display, team sheets, player profiles
6. Set up sponsor integration (carousel, full-page ads)
7. Configure PWA capabilities
8. Deploy frontend to Vercel and CMS to chosen platform

Important Context

- **5 Teams Phase 1:** 4 men's, 1 ladies
- **QR Code Entry Point:** <https://app.selbyrugby.com> (domain TBD)
- **Admin CMS:** <https://admin.selbyrugby.app> (domain TBD)
- **Sponsor Types:** Club sponsors (carousel) and player sponsors (individual cards)
- **Team Sheet Format:** 15 starters (1-15) + 8 subs (16-23)

When implementing features, always refer to the PRD for detailed requirements, user stories, and UI specifications.