SportHawk

1. Project Overview & Introduction

Dear Teaser Software,

Following our recent discussions, this contract outlines the terms for a fast-paced, 8 day development sprint to prepare our application, **SportHawk**, for its target launch on the App Store and Google Play by **October 1st, 2025**.

SportHawk is a React Native (Expo) application with a Supabase backend, designed to simplify management for sports clubs and grow their communities. The project is in a **brownfield state**; the core architecture, navigation, and a significant portion of the screens are already implemented.

The primary goal of this contract is to complete the essential screens required for our Minimum Viable Product (MVP) to allow SportHawk to be published on the App Store and Google Play Store by **October 1st, 2025**. This involves:

- Developing the remaining 3 screens from scratch.
- Testing, debugging, and fixing the known issues within each priority (Section 2).

We have tried our best to describe the tasks and screens below, and we will provide detailed documentation for every screen. We are looking for skilled development resources to integrate into our team (Ashton and Andrew) and help us cross the finish line.

2. Scope of Work & Key Deliverables

The app's visual scope is strictly limited to the completion of the 25 screens listed in the "Prioritised Screen List" document. All development must be pixel-perfect, adhering to the provided Figma designs and technical specifications. We have recommended a development sequence below, but this can be flexible.

Priority 1: Profile/User Settings

- **Symptom:** Logged in user who refreshes the app is taken back to the Welcome screen when reopened, rather than the Home screen. The app does not recognise they were previously signed into the app.
- **Fix & Debug:**
    - Resolve the session handling issue on the Welcome screen (Node ID: 559-467).
- **Goal:** User session. 1) The user previously signed in to their account who refreshes the app, will remain signed in the next time. Only until the user chooses to Sign out, will the session end and they will be taken back to the Welcome screen. 2) Retrieve the user's device id, in order to be able to use app push notifications (this is dependent on resolving the user session issue).

Priority 2: Payments: Not Working, Missing Screen, and Corrective Tasks

Payments Not Working:

- **Symptom:** Pop-up saying "Webhook returned non-2xx code" when payment attempted.
- **Fix:** Payments. The areas involved are: 1) SupaBase Webhook & Edge functions, 2) SupaBase tables payment_*, 3) /lib/api/payments.ts ; 4) Payment details screen Figma 559-3055
- **Goal:** Payments operational: 1) completes with valid Stripe test card details, and 2) payment results in suitable pop-up error message for non-valid Stripe test card details

Fix & Debug:

- “Payments - Admin” screen Figma 559-2776 (reached via Teams / Admins / Payments icon)
- “Payments - Details Admin” screen Figma 559-2792 (reached via Teams / Admins / Payments icon). Variant of “Payment - Details” screen, for Admins to view payments responses “paid / unpaid.
- “Edit Payment Screen” screen Figma 559-2709 (reached via Teams / Payments / Details / “three dots”).
    - ○ SportHawk will provide a descriptive document “Payments Lifecycle” which outlines how these screens intend to work.

Corrective Tasks:

- Fix: Upcoming / Outstanding payments incorrectly shows completed payments (Teams / Payments)
- Fix: Teams / Payments dropdown to ('All time' or 'This Week') to look & work the same as Teams/ Events dropdown.
- Develop Send/Cancel for Send Reminder that puts rows in the SupaBase notifications table

Priority 3: Events: Related Screens

- **New "Events" Development:**

SportHawk will provide a document covering User Experience and database relations.

- ○ Reuse Event Details (559-3117) so that a Team Admin sees and can use 3-dot dropdown menu (Send Reminder, Edit Event, Select Squad), also this screen has a variant (609-10067) to show the Squad when one has been selected.
    - ■ Develop Send/Cancel for Send Reminder that puts rows in the SupaBase notifications table.
    - ■ Select Squad menu option goes to "Select Squad" if none selected, else "Edit Squad".

Missing Screens:

- Develop the "Edit Event" screen Figma 559-2833 (base it on Create Event 559-2816).
- Develop the "Select Squad" screen Figma 559-2853.
- Develop the "Edit Squad" Figma 559-2881.

- **Fix & Debug:** Address all known issues in the existing Events screens, including:
  - Fix: Teams / Events "thumbs" up/down/flat response buttons.
    - Then Fix: Teams / Event / Detail: Members Responses section totals (counts of thumbs up / flat / down) and list of Members with the response of each or "?" for unresponded.
  - Fix: Teams / Event / Detail: to show the Google Map with pin, and if tapped to jump into the default browser at a URL to show Google Maps centred & pinned on the address of the event.

Priority 4: Team Members & Admins Module

Note: We have done the development, and expect to have debugged, the following:

- Manage Members (559-2682),
- Add Members (559-2661),
- Manage Admins - (559-2613)

- **Fix & Debug:**
  - Implement the display of the team member list on the Members screen (Node ID: 559-3033).
  - Address non-functioning buttons on the Admin screen (Node ID: 559-2966) and clarify workflow. For the MVP, of the 6 buttons on Admin screen the three that must work are:
    - Payments > “Payments - Admin” screen Figma (559-2776)
    - Members > “Manage Members” screen Figma (559-2682)
    - Admins > “Manage Admins” screen Figma (559-2613)

Out of Scope: Screens not in the 25 prioritized screens are out of scope.

3. Agency Responsibilities & Technical Expectations

- **Code Quality:** Write clean, maintainable, and well-documented TypeScript code.
- **Adherence to Architecture:** Strictly adhere to the established project architecture, patterns, and principles outlined in the provided documentation (Front End Spec MVP v4, Architecture MVP v4). This includes:
  - Using the existing Sh* component library for all UI elements.
  - Using the established API layer in /lib/api for all Supabase queries.
  - Using the global config files for all styling (colors, spacing, typography) with no "magic values".
- **Version Control:** All work must be done in feature branches on GitHub. Code will be submitted via Pull Requests for review by Andrew.
- **Communication:** Provide a concise progress update every 2-3 days, highlighting completed tasks, current work, and any blockers.

4. Resources Provided by SportHawk

- Full access to the GitHub repository.
- Access to the Supabase project.
- Access to a shared Google Drive containing all essential documentation, including:
    - The Prioritised Screen List.
    - Highly detailed lifecycle documents for Payments, Events, Members, etc. ○ Front End Spec MVP v4 and Architecture MVP v4.
- Timely reviews and feedback from Ashton (styling/UX) and Andrew (code/logic).

5. Confidentiality & Intellectual Property

- **Confidentiality:** You (the Agency) agree to treat all project information, including the codebase, documentation, and business logic, as strictly confidential.
- **Intellectual Property:** Upon full payment, all code and materials ("Work Product") created under this contract will become the sole property of SportHawk.