

Elevyn App: Concept Document



2025

Table of contents

1. Project idea and Description
2. Sources of Inspiration
3. Competitor research
4. App features
5. Technical stack
6. Design & Styleguide
7. Updated Project timeline & Hours
8. Updated Project goals

Project idea & description

App name: *Elevyn*

Elevyn is a mobile-first app designed to support student-athletes in managing both their academic and athletic responsibilities within a single, streamlined platform. The app bridges the gap between existing learning management systems (LMS) and sports tracking tools by offering an integrated, user-friendly experience that reflects the dual life of a student-athlete.

Why is it important to create this :

- As a student athlete myself, we are overwhelmed juggling performance, school, and recovery.
- No major app balances both academic and athletic needs in a unified platform.
- Helps athletes build discipline and balance — not just performance.

- Ideal for personal development, sports organisations, and school athletic programs.

What the App Represents:

- Student–Athlete dual life (balance, time management)
- Performance + wellness + academics in one space
- Built for growth, clarity, and control
- Designed with empathy and intention

Sources of Inspiration



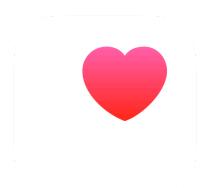
Nike Training Club



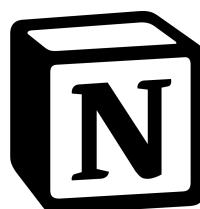
Canvas LMS (uLwazi)



Nike Running Club



Apple Health App



Notion



Headspace Wellness App



Basketball Stats Assistant

Competitor research

In an increasingly saturated app ecosystem, student-athletes remain underserved. While there are powerful tools for fitness, wellness, or academics, few — if any — speak to the interconnected lifestyle of student-athletes who must simultaneously manage academic deadlines, athletic performance, and personal wellbeing. This is where Elevyn emerges as a unique solution: a holistic, mobile-first companion built specifically for student-athletes.

A few established apps were reviewed to better understand the competitive landscape:

- Nike Run Club / Nike Training Club

These apps offer excellent guidance on physical fitness, personalised workouts, and running metrics. However, they focus solely on athletic performance, without addressing academic obligations or mental health. They cater to general users and professional athletes — not the dual responsibilities of students balancing training with coursework.

- Headspace

Headspace provides a strong foundation for mindfulness, meditation, and mental health. It is well-designed, easy to use, and supportive of emotional wellbeing. Yet, it does not offer integration with academic or athletic tracking, and lacks contextual relevance to the student-athlete experience.

- Notion / Canvas LMS

Both Notion and Canvas are widely used in academic settings. Notion excels in task management and flexibility; Canvas supports learning management and course tracking. Still, these tools lack intuitive support for athletic planning, and do not incorporate any features related to physical performance or wellness.

- Basketball Stats Assistant & Other Sport-Specific Tools

These tools specialise in game-day statistics, team management, and season performance analysis. While useful for coaches and athletes, they tend to focus narrowly on in-game metrics, and exclude off-court lifestyle factors like sleep, nutrition, or academic scheduling.

- Health Apps (e.g., Apple Health, Samsung Health)

Generic health apps track daily activities like sleep, water intake, and heart rate. However, they operate in silos, do not personalise data for the student-athlete context, and offer little to no planning or academic support. The experience is often fragmented and passive.

Why Elevyn is Different

Elevyn offers an integrated, seamless experience where student-athletes can manage their:

- Academic workload (class schedules, assignments, study sessions)
- Athletic training (preloaded and custom logs, weekly planner, calendar)
- Personal wellness (mood logs, hydration, sleep tracking)
- Mindset and reflection (journaling, performance summaries)

Unlike other apps, Elevyn does not require the user to jump between platforms. Everything is in one place — balanced, intuitive, and tailored for the demands of dual life as a student and athlete. It supports both preloaded mock data and lightweight user input to demonstrate functionality during early phases, with a clear roadmap toward scalable features.

App features

Mock Authentication (Intro Screen)

- **What:** Simple login screen with a hardcoded user ("Jamie").
- **Purpose:** Simulates user-specific experience (even though no real auth).
- **How:**
 - Input fields for email + password
 - Button: "Login"
 - On success → redirect to dashboard with personalised data ("Welcome, Jamie!")

Dashboard (Home Screen)

Summary tiles with a blend of academic + athletic info:

- Upcoming event: "Assignment due tomorrow"
- Next game: "Vs Wits on Thursday"
- Training: "3 sessions this week"
- Academic task: "You have 2 tasks outstanding"
- Wellness highlight: "Your energy was low on game day"

Integrated Calendar View

Academic + Athletic Events Combined:

- "Research Methods Assignment – Due Tuesday"
- "Basketball Game – Thursday"
- "Recovery Day – Sunday"

Preloaded data only

Weekly Planner / Task Manager (Academic + Sport)

- User can add tasks
- Tasks have categories/tags: Athletic or Academic
- Tasks show status: "To do" or "Done"

Examples:

- "Submit Lit Review"
- "Watch game film"
- "Meet tutor at 3pm"

Wellness Log (Academic + Athletic Impacts)

- Track energy, stress, sleep — can relate to school, sport, or both
- User inputs daily data
- Visual insights: "Stress levels highest during game + deadline week"

Progress Dashboard

A holistic view:

- Academic Tasks Done: 5/7 this week
- Athletic Events Attended: 3 trainings, 1 game
- Wellness Score: 72% energy average
- Balance Insight: "You've logged more hours on sport than study this week — try adjusting."

Settings/Profile

- Just visual — change avatar, sport, major
- Potential future settings: notification preferences, reminders

Performance Tracker (Athletic)

- Preloaded game stats (3 basketball games)
- No user input for now
- Graphs + summary (FG%, points over time)

Technical Stack

Technical Stack:

Frontend:

- React Native (Core framework)
- CSS/SCSS or Tailwind CSS (for styling)
- React Router (for navigation)
- Framer Motion (for subtle animations)

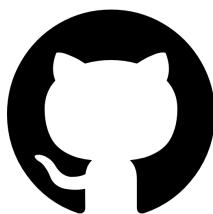
Backend:

- Firebase

Other Tools:

- Figma (for UI/UX design and prototyping)
- Notion (to manage tasks)
- Git + GitHub (for version control)

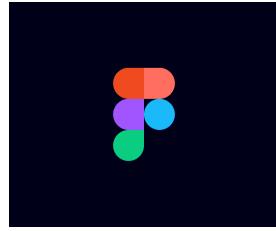
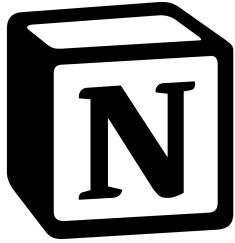
Project safety and version control:



Github will be used for version control of the entire project- it'll help me check technical changes made to the project and recover code that was removed which worked better than new code.

VS Code will be used to write the code for the project, I will be writing most of the front-end code here using React (JavaScript) followed by the necessary backend code that may be needed.

Expo/ ExpoGo will be used to host the app that I'm building so we can see what the app will look like and how it will function on a mobile device.



Notion is a great way to keep track of the overall project. I get to see which parts of the project are complete and which parts are in progress or haven't started. It's a great software for a project timeline overview.

Figma will be used for the user interface of the project, it will help flesh out what final design of the app will look like and help build and mould the functionality of the app

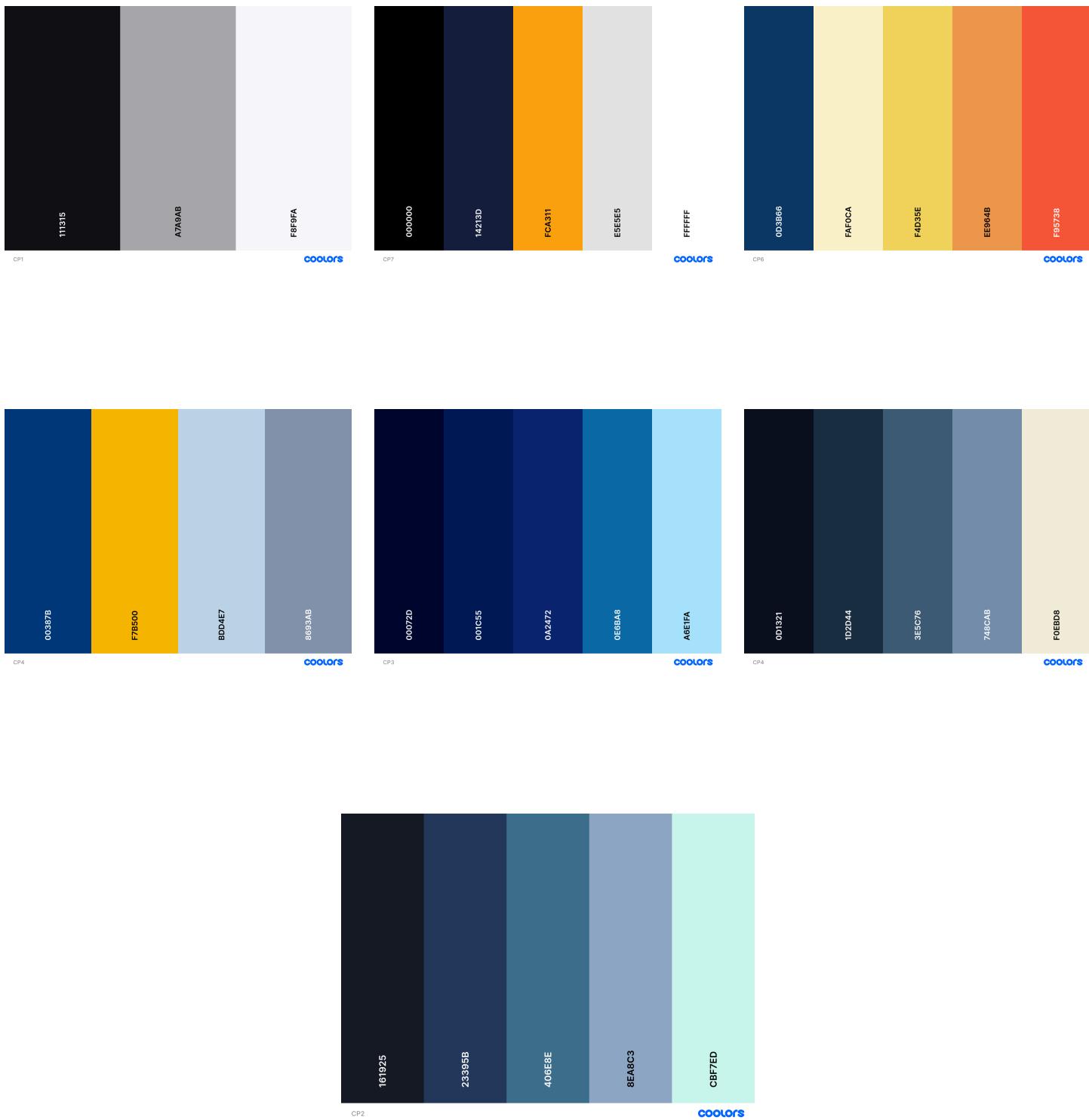
Design & Styleguide

This guide outlines the core visual elements — including colour palettes, typography, logo usage, iconography, and interface patterns — that ensure consistency across all touch points. Whether designing a new screen, marketing material, or pitch deck.

Logo Design ideas:



Colour Palettes:



Typography

Primary Font:

- Font Family: Poppins (*Google Font — clean, modern, highly legible*)
- Weights: 400 (Regular), 600 (Semi-Bold), 700 (Bold)

Usage Guidelines:

- **Headings:** Poppins Bold, all caps, slight letter-spacing
- **Body Text:** Poppins Regular or Medium, 16px for default size
- **Captions / Notes:** Poppins Light, 12–14px, subtle contrast color

Text Type	Font	Size	Weight	Use Case
H1	Poppins	32px	Bold	Main screens, titles
H2	Poppins	24px	Semi-Bold	Subheadings, section titles
Body Text	Poppins	16px	Regular	Paragraphs, logs
Caption/Text	Poppins	12px	Light	Notes, timestamps

2. Iconography

Style:

- Linear / outline icons (HeroIcons, Feather Icons, or custom SVGs)
- Stroke width: 1.5px–2px
- Rounded corners for a friendly but modern look

Primary Icons Used:

- Calendar
- Stats chart
- Mental wellness (brain or heart icon)
- Dumbbell or running figure
- Academic cap
- Home
- Checklist
- Add / Edit

3. Component Styling

Component	Style Details
Buttons	Rounded edges (8px radius), bold text, elevation (shadow), dark/light modes
Inputs	Underline or box styles depending on theme, placeholder text in gray tone
Cards	Shadowed container with slight padding, title & small icon for quick access
Navigation Bar	4-Tab bottom nav (Home, Dashboard, Log, Profile), active tab highlighted
Chart Widgets	Circular progress bar, bar chart with color-coded data from Recharts
Calendar View	Expandable with colour-coded events and smooth animations

Authentication user flow:

Welcome Screen →

→ Login (email/password) → Success → Home Dashboard

→ Register (email, name, password) → Set Preferences → Home

→ Forgot Password → Reset Email Sent → Login

Main App Flow (Post-Login):

Home Dashboard

|—— Performance Tracker (Recharts + Input Logs)

| |—— Wellness (Mood, Sleep, Water, Mental State)

| |—— Academics (Deadlines, Study Logs)

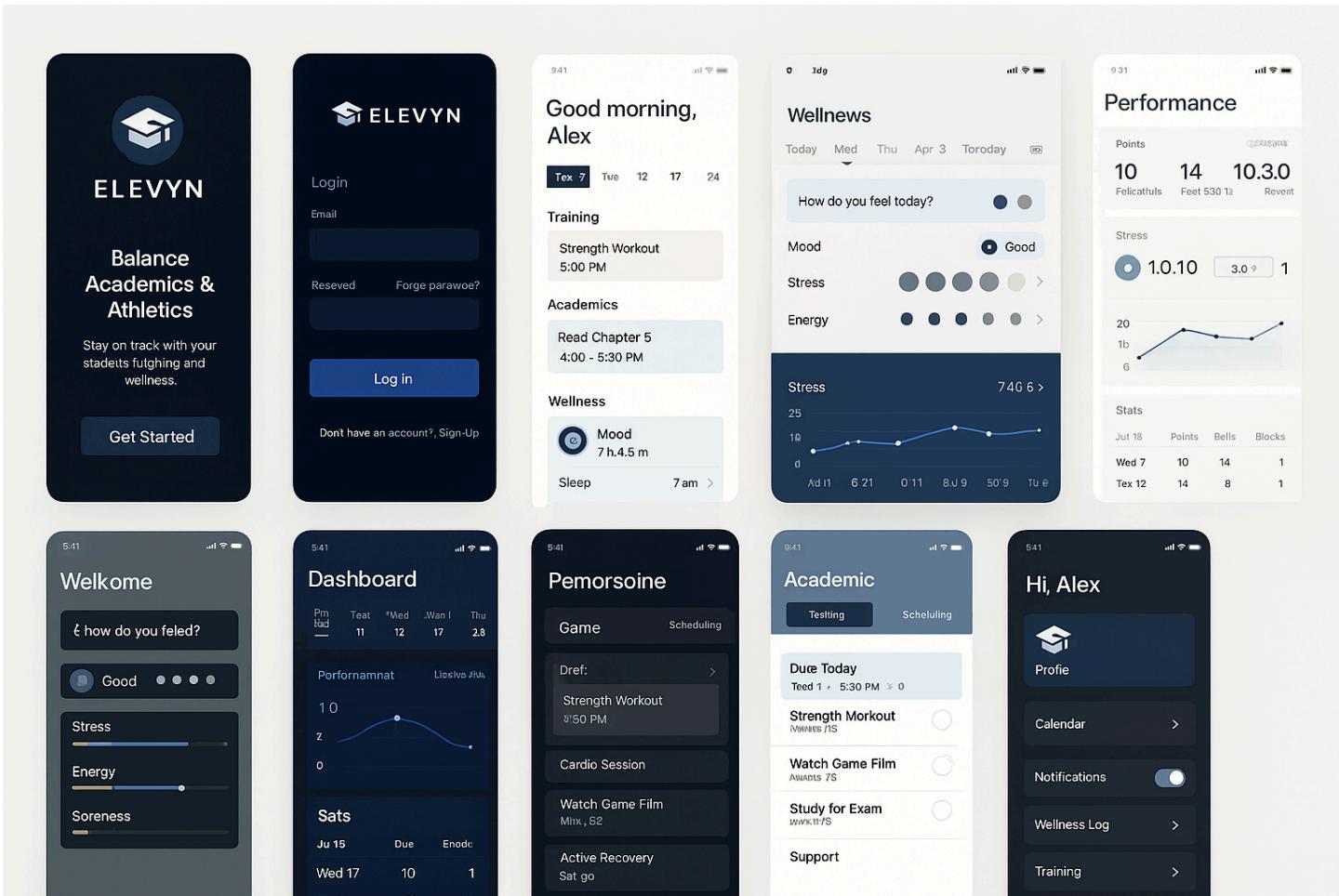
|—— Calendar (View events, workouts, academic tasks)

|—— Progress (Graphs, Weekly Summaries, Notes)

|—— Add New Log → Choose: [Wellness] / [Academic] / [Workout]

|—— Profile → Preferences / Edit Info / Logout

Example of wireframe:



Project Timeline & Hours

Phase 1: Concept

- Finalize branding, logo, colours
- Create wireframes and user flow
- Build basic screens in Expo (login, dashboard, 1-2 key features)
- Load sample data

Phase 2: Feature Development

- Finish all key screens (calendar, planner, performance, wellness, dashboard)
- Add more interactivity (task completion, wellness check-ins)
- Use local data for persistence
- Polish navigation + animations

Phase 3: Polish + Backend

- Refine UI and transitions
- Optional: Add Firebase back-end
- Conduct user testing (2–3 people)
- Collect feedback and improve UX

Phase 4: Final Touches + Submission Prep

- Final bug fixes and testing
- Prepare documentation, README, and project summary
- Record complete demo walkthrough video

Project Hours:

Planning, Research, & Wireframes

Task	Hours
Competitor research + feature scoping	8
User personas + journey mapping	6
Initial sketches & low-fidelity wireframes	10
UI moodboarding + design system setup	6
Mid–high fidelity wireframes	6
Finalizing feature list + roadmap	4

Front-End (React Native)

Task	Hours
Project setup (Expo, Git, folder structure)	8
Navigation (React Navigation setup)	10
Authentication flow (mock login/signup)	10
Dashboard UI (calendar, performance tiles)	20
Calendar feature (preloaded + editable)	18
Performance tracking UI (mock stat display)	18
Planner feature (add/edit tasks)	16
Wellness log UI (mood, sleep, input form)	16
Academic module (class schedule, tasks)	18
Settings page + mock profile	8
Reusable components & styling cleanup	8

Back-End (Firebase)

Task	Hours
Backend setup + environment config	6
Design mock database schema	10
Build mock user authentication API	10
Endpoints for calendar, planner, stats	14
Wellness log & academic data APIs	14
Local data storage & retrieval logic	12
API testing	6
Basic error handling + response formatting	8

Feature Integration + Data Storage

Task	Hours
Connect front-end to mock data	10
Integrate calendar data into UI	8
Populate preloaded performance data	6
Planner data add/edit/delete workflow	8
Wellness log input and chart/summary display	8
Academic planner integration (classes, tasks)	6
Settings toggles (theme mode, profile data mock)	4

UI Polish + Accessibility

Task	Hours
Design system cleanup (colors, fonts, spacing)	8
Add icons, visuals, empty states	6
Responsive layouts for different screen sizes	6
Accessibility (alt text, keyboard nav, contrast)	6
UI animations (subtle transitions)	4

Testing + Bug Fixing

Task	Hours
Manual testing (feature-by-feature)	12
Cross-device & OS testing	6
Fixing common UI and logic bugs	8
Code cleanup & commenting	4

Documentation + Demo Prep

Task	Hours
Write README (features, install, usage)	6
Record walkthrough demo (video or live)	8
Prepare slides for project presentation	6
Write developer notes for future dev	6
Backup and project folder organisation	8
Polish demo	6

Project Goals

Personal goals include:

- Improved confidence in full-stack mobile development (including back-end development to my technical stack)
- Better understanding of how to scope and manage a solo product build from scratch.
- Refined skills in UI/UX communication, visual hierarchy, and cross-disciplinary app thinking.

Professional goals:

- Improve UI/UX design thinking, including colour theory, typography, wireframing, and prototyping in Figma.
- Create a polished, real-world portfolio piece that demonstrates end-to-end product thinking: from ideation to deployment.
- Explore ways to ethically and mindfully represent underrepresented groups (e.g., African student-athletes) in app design and data models.

Project specific goals:

- Design and build a cross-platform mobile app using React Native for the front-end and Firebase for the back-end.
- Provide integrated wellness tracking, including sleep, hydration, mood, and mental wellness.
- Incorporate academic planning tools, such as a calendar and task log, to support scheduling around training and game days.
- Offer visual performance insights using charts and logs to show patterns in both academic and physical performance.
- Ensure the app is accessible, user-friendly, and visually aligned with the needs of student-athletes (e.g., bold visuals, dark mode, intuitive navigation).
- Maintain data integrity and scalability with future database integration.