

Blood Bank Finder

Capstone 06 · HealthTech · Talodabi Academy

1 · Project overview

You are designing **fast search and contact** for blood in an emergency. You are not rebuilding full hospital logistics. Focus on clarity when **no donors** are found.

Deliverable	Full mobile MVP in Figma (not code)
Platform	Mobile app · Android-first · portrait
Screens	10 screens + 1 unhappy path (11 frames)
Features	3 core features
Bottom navigation	Bottom navigation with 3 items: Find · Donate · Profile
Prototype	4 linked flows

Build order: Auth (screens 1–4) → nav tabs (5–7) → core flow (8–10) → unhappy path → link prototypes.

2 · Problem & primary user

Hospitals often run out of blood during emergencies, and families lose critical time searching for donors. Families are asked to find compatible donors on the spot, sometimes at midnight. There is no central system where someone can quickly search for blood by type and location. If you cite demand or outcomes, name a **hospital or NGO** source.

Primary user

"Ngozi, a 31-year-old nurse in Ibadan who frequently cannot find compatible blood for emergency patients, especially for rare blood groups, and loses precious time making phone calls."

Geographic scope: State-level (Oyo)

3 · Goals

User goal: Find verified blood donors or blood bank locations near her within minutes.

Business goal: Build a national donor network and partner with hospitals to manage emergency blood requests.

Success: In a quick test, someone selects **blood group + city**, reaches a **contact action**, and understands the **empty state** fallback.

4 • Market context

QUESTION	ANSWER
Who pays	Ngozi and patients do not pay. Hospitals, NGOs, or emergency funds pay for coordination (LifeBank model). Assumption for your case study.
Why not WhatsApp	Hospitals already blast WhatsApp for "O+ needed at LUTH." That is fast but chaotic, unverified, and exposes numbers without availability status.
Design focus	Do not rebuild LifeBank end-to-end. Design a wedge: donor availability + hospital search in one city , or rare blood group empty states that fall back to blood banks (already in brief).

Competitors to check:

- **LifeBank** (blood and oxygen logistics; app, call centre, hospital delivery)
 - **Haima Health Initiative** (donor database + helpline / WhatsApp; web signup at haimahealth.org.ng)
 - **Hospital phone trees + estate WhatsApp** (default emergency search)
-

5 • Validate before you design

Interview 3–5 people who match your primary user. Update this brief if interviews contradict it. Do not start hi-fi until you complete the confirmation below.

FIELD	YOUR ANSWER
Validated user (name, age, city)	
Problem in one sentence	
One interview quote	
Success metric for your test	
Primary colour and tone	
One competitor you checked	

Interview prompts

- "Walk me through the last time this happened."
- "What did you use instead? WhatsApp, paper, or nothing?"
- "What would make you trust this on a cheap Android phone?"
- "What would stop you from using this every week?"

6 · Features & scope

v1 · three features

1. Search for blood donors by blood group and city
2. Register as a blood donor with availability status
3. Emergency request button that notifies nearby registered donors instantly

Out of scope

- Ambulance dispatch or blood storage
- Payments from patients
- Nationwide donor verification (design UI; label ops as assumption)

7 · Screens

Design every row in wireframe, high-fidelity UI, and prototype (where the flow applies).

#	LAYER	SCREEN	WHAT TO SHOW
1	Auth	Splash Screen	Bold red app logo on a white background. Tagline visible.
2	Auth	Onboarding (1 screen)	One screen: 'You could save a life by registering as a donor or helping someone find blood.' Two buttons: Find Blood and Register as Donor.
3	Auth	Sign Up Screen	Name, blood group (dropdown), phone number, state, and PIN.
4	Auth	Sign In Screen	Phone number and PIN.
5	App shell	Find (tab)	Find Blood Now button and blood group filter. Show bottom nav with Find active.
6	App shell	Donate (tab)	Register as donor form or availability toggle if already registered.
7	App shell	Profile (tab)	Name, blood group, city, and donor status badge.
8	Core flow	Search results	Donors listed by blood group and city. Show bottom nav on post-sign-in screens.
9	Core flow	Donor profile	Contact options. No home address shown. Show bottom nav on post-sign-in screens.
10	Core flow	Register · Donor saved	Donor registration confirmed on Donate tab. Show bottom nav on post-sign-in screens.

8 · Prototype flows

Link these in Figma. Examiners should click through without your voice-over.

Flow 1 · App entry

Splash → Onboarding → Sign Up or Sign In → first main tab

Flow 2 · Core task (all three features)

Sign In → Find tab → Search → Donor profile OR Donate tab → Register saved

Core flow screens: Search results · Donor profile · Register · Donor saved

Flow 3 · Navigation

Link every nav tab so examiners can tap each bottom nav item. Tabs: Bottom navigation with 3 items: Find · Donate · Profile

Flow 4 · Unhappy path

Trigger the unhappy path edge case and link to that screen.

9 · Unhappy path

Design this as its own screen state (frame 11):

No donors found for the selected blood group in the user's city. Show: 'No donors found near you. Here are the 3 closest blood banks.' List them with address and phone number.

Link in **Flow 4**.

10 · Design system

RULE	DETAIL
Colour	You choose. Keep it simple and readable.
Type	You choose. Keep body text easy to read on a phone.
Tone	Design for your user, not for a tech audience.

Reuse on every screen:

- Primary button (active and disabled states)

- Input field (empty, filled, and error states)
- Card or list item used across the app
- Navigation bar with correct labels and icons
- Empty state: what the screen looks like with no data

11 • Submit & defense

Deliverables

#	DELIVERABLE	WHAT GOOD LOOKS LIKE
1	Wireframes	Every screen in this brief, plus the unhappy path. Layout and labels clear.
2	High-fidelity UI	Same frames polished. You choose colour and type. Bottom nav on every post-sign-in screen.
3	Prototype	Four linked flows (entry, core task, navigation, unhappy path). Examiners can use the app without your voice-over.
4	AI prototype	Figma Make: at least 2 screens. Share link.
5	Case study (1 page)	Problem · User · Who pays · Competitors · Solution · One design decision
6	Presentation (6–8 slides)	Name · Problem · User · Solution · Key screens · Flow demo · One decision

Before you present

- ☐ Every screen designed in wireframe and hi-fi
- ☐ Unhappy path screen designed and linked in the prototype
- ☐ Four prototype flows linked end to end
- ☐ At least one empty state on a list tab
- ☐ Shared components reused: button, input, card or row, bottom nav
- ☐ Case study names who pays and one competitor you checked
- ☐ You can explain one design decision in critique (why, not what)
- ☐ Completed 5 user interviews or 1 usability test
- ☐ Checked at least one competitor listed under Market context