

Ogun Cocoa Ledger

Capstone 44 · AgriTech · Talodabi Academy

1 · Project overview

Cooperative delivery log for Ogun cocoa farmers.

Deliverable	Full mobile MVP in Figma (not code)
Platform	Mobile app · Android-first · portrait
Screens	11 screens + 1 unhappy path (12 frames)
Features	3 core features
Bottom navigation	Log · Statement · Season · Profile
Prototype	4 linked flows

Build order: Auth (screens 1–4) → nav tabs (5–8) → core flow (9–11) → unhappy path → link prototypes.

2 · Problem & primary user

Ogun cocoa belt farmers sell through cooperatives without ledger clarity. Commodity apps often skip small Ogun cooperatives.

Primary user

"Mr. Fashina, a 50-year-old cocoa farmer near Ilisan who never sees a clear payout breakdown."

Geographic scope: State-level (Ogun)

3 · Goals

User goal: Log deliveries and see expected payout per season.

Business goal: Cooperatives pay for ledger seats; buyers license aggregate data.

Success: Farmer logs delivery and reads season statement.

4 · Market context

QUESTION	ANSWER
Who pays	Cooperatives pay seats; buyers license aggregate data.
Why not WhatsApp	Farmers trust cooperative verbally; no delivery statement.
Design focus	Bag log plus season statement on phone for Ogun cocoa belt.

Competitors to check:

- **Cooperative paper ledger** (default)
- **Middleman verbal price**
- **Commodity export platforms** (B2B only)

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. Log bag count and grade per delivery
2. See cooperative price and deductions
3. Season statement export for farmer

Out of scope

- Export trading
- Payments holding
- All commodities

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	App logo, name, and tagline for Ogun Cocoa Ledger.
2	Auth	Onboarding (1 screen)	Explain the app in one sentence. One Get Started button.
3	Auth	Sign Up Screen	Name, phone number, state, and PIN. No email required.
4	Auth	Sign In Screen	Phone number and PIN. Forgot PIN link.
5	App shell	Log (tab)	Add delivery button and recent deliveries list. Show bottom nav with Log active.
6	App shell	Statement (tab)	Line items: gross, fees, net payout for current season. Show bottom nav with Statement active.
7	App shell	Season (tab)	Compare this season vs last season totals. Show bottom nav with Season active.
8	App shell	Profile (tab)	Farmer name, cooperative, and farm location. Show bottom nav with Profile active.
9	Core flow	Delivery · Logged	Bags, grade, receipt photo saved. Show bottom nav on post-sign-in screens.
10	Core flow	Statement · Net payout	Gross, fees, net lines visible. Show bottom nav on post-sign-in screens.
11	Core flow	Season · Compare	This season vs last season totals. Show bottom nav on post-sign-in screens.

Layers: Auth = 1–4 · App shell = 5–8 · Core flow = 9–11

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 → Log tab → Delivery logged → Statement tab → Season tab

Core flow screens: Delivery · Logged · Statement · Net payout · Season · Compare

Flow 3 • Navigation

Link every nav tab so examiners can tap each bottom nav item. Tabs: Log · Statement · Season · 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 12):

Delivery logged without grade. Show: Select grade so payout estimate is accurate.

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

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

© Talodabi Academy · For enrolled students · Not for redistribution

