


# PRODUCT PARTNER FOR FINTECH AND BANKING

# THREE DISCIPLINES THAT RUN EVERY PROJECT

This is the full breakdown of what we do. On a project, they work as one team, not three handoffs.

The point of this document is reach, so you can see exactly what sits inside each discipline, who runs it, and roughly what it takes.

Nothing here is bought à la carte. We assemble the right capabilities around the problem in front of us.

Effort ranges are indicative and depend on the system.

01

## STRATEGY

Turning a business goal into a product decision someone can defend in a board review.

02

## DESIGN

Designing products people can actually use, in markets where the details decide trust.

03

## ENGINEERING

Building systems that hold under load, pass audit, and keep moving after launch.

# STRATEGY

Turning a business goal into a product decision someone can defend in a board review.

01 Infrastructure review

02 Business process audit

03 Research and validation

04 Proof of concept

05 Solution design

06 Roadmap creation

01

## INFRASTRUCTURE REVIEW

We map how an existing system actually works and where it breaks. The output is a clear picture of the current architecture and the bottlenecks standing between you and the business goal.

TEAM

Solution architect, discipline leads, business analyst (as needed)

TYPICAL EFFORT

1–4 weeks

02

## BUSINESS PROCESS AUDIT

We analyse current processes against the goal at hand, then set out the steps and the problem areas. You get an honest read on whether the goal is reachable as things stand.

TEAM

Solution architect, business analyst (as needed)

TYPICAL EFFORT

1–4 weeks

03

## RESEARCH AND VALIDATION

Business interviews, competitor and market analysis, audience research, and in-depth user interviews. The evidence base that keeps product decisions grounded rather than assumed.

TEAM

Lead product designer, UX designer, stakeholders

TYPICAL EFFORT

Scoped per project

04

## PROOF OF CONCEPT

Before committing a budget, we test whether a specific feature can be built the way it needs to be. It settles the question early, while it's still cheap to answer.

TEAM

R&D developer

TYPICAL EFFORT

1–4 weeks

05

## SOLUTION DESIGN

We define the business processes, data flows, and system design, then validate them. This is the blueprint the build runs on, agreed before anyone writes production code.

TEAM

Solution architect, business analyst (as needed), product manager (as needed)

TYPICAL EFFORT

2–8 weeks

06

## ROADMAP CREATION

Once the direction is validated, we turn it into a sequenced plan: goals, stages, and timing you can plan budgets and resourcing against.

TEAM

Solution architect, product manager (in consultation with the team)

TYPICAL EFFORT

1–2 weeks

# DESIGN

Designing products people can actually use, in markets where the details decide trust.

01 Discovery and concept
02 Usability testing
03 UX audit
04 Accessibility audit
05 UX architecture
06 AI prototyping
07 Visual style and UI design
08 Design systems
09 Customer experience design
10 Illustrations, motion, and assets
11 Design QA, documentation, and support

01

## DISCOVERY AND CONCEPT

We turn research into a working concept of the product — the most convincing hypotheses, made tangible. It moves the conversation from words to something you can react to.

TEAM

Lead designer, interface designer, stakeholders

TYPICAL EFFORT

2–5 weeks

02

## USABILITY TESTING

We validate decisions against real users at any stage of design, before money goes into building the wrong thing. More testing, more certainty.

TEAM

Lead designer, interface designer

TYPICAL EFFORT

2–4 weeks

03

## UX AUDIT

We review an existing product against how users actually move through it: structure, flows, friction, and consistency. The output is a prioritised read on what's costing you conversion or trust. Works standalone or as the entry point to a larger engagement.

TEAM

Lead designer, interface designer

TYPICAL EFFORT

1–3 weeks

04

## ACCESSIBILITY AUDIT

We audit the product against WCAG 2.2 and the European Accessibility Act, in force for banking since June 2025. The output is a gap report with prioritised fixes: legal exposure, usability cost, and what can wait — mapped to specific screens, not abstract principles.

TEAM

Lead designer, interface designer, QA engineer (as needed)

TYPICAL EFFORT

1–3 weeks

05

## UX ARCHITECTURE

We define the high-level structure, flows, and navigation — the skeleton the whole product hangs on. This is where the scale of the work becomes clear and can still be shaped.

TEAM

Lead designer, interface designer, project team

TYPICAL EFFORT

1–5 weeks

06

## AI PROTOTYPING

We turn the agreed structure into working, clickable prototypes. AI lets us build them fast, so the full behaviour of the product is testable before a single pixel is polished, and still cheap to change while the logic is in play.

TEAM

Lead designer, interface designer, project team

TYPICAL EFFORT

2–5 weeks

07

## VISUAL STYLE AND UI DESIGN

The product gets its visual language (colour, type, components) and then its final production design, with every system state and edge case resolved. This is what users touch.

TEAM

Lead designer, interface designer(s), project team

TYPICAL EFFORT

4–20 weeks

08

## DESIGN SYSTEMS

We build the structure that keeps multiple products consistent and fast to extend — a maintainable system of components and rules, not a one-off file.

TEAM

Lead designer, interface designer(s)

TYPICAL EFFORT

1–5 months

09

## CUSTOMER EXPERIENCE DESIGN

We design the whole journey, online and off, not just the screens — so the experience holds up across every point where the customer meets the product.

TEAM

Lead designer, interface designer, behavioural scientist, stakeholders

TYPICAL EFFORT

Scoped per project

10

## ILLUSTRATIONS, MOTION, AND ASSETS

Illustration, micro-interactions, and the production graphics a release needs. The details that make a product feel finished and reduce friction in use.

TEAM

Interface designer, motion / 2D-3D artist

TYPICAL EFFORT

1–7 weeks

11

## DESIGN QA, DOCUMENTATION, AND SUPPORT

We check the build matches the design, document the logic for the teams who carry it forward, and keep iterating after launch on real data.

TEAM

Lead designer, interface designer

TYPICAL EFFORT

Ongoing

# ENGINEERING

Building systems that hold under load, pass audit, and keep moving after launch.

01 Product development

02 Deployment and monitoring

03 Cloud and infrastructure

04 Acceptance and automated testing

05 Platform and core-banking integrations

06 Security and compliance delivery

07 Maintenance and ongoing support

01

## PRODUCT DEVELOPMENT

Maintainable, documented, tested code, engineered to hold under load and meet its SLA targets, following established security and engineering practice (OWASP, SOLID, high-load testing).

TEAM

Tech lead, senior developers, QA, project manager

TYPICAL EFFORT

6+ months

02

## DEPLOYMENT AND MONITORING

Environments, infrastructure as code, continuous delivery, and monitoring set up so issues surface early and releases stay flexible.

TEAM

DevOps

TYPICAL EFFORT

1–6 weeks

03

## CLOUD AND INFRASTRUCTURE

Cloud architecture and migration, DevOps practice, and the infrastructure decisions that keep a product scalable and resilient as it grows.

TEAM

DevOps, solution architect

TYPICAL EFFORT

Scoped per project

04

## ACCEPTANCE AND AUTOMATED TESTING

Automated tests for interfaces and APIs against defined scenarios, so new functionality integrates safely and changes stay stable.

TEAM

Automation QA

TYPICAL EFFORT

1–6 weeks

05

## PLATFORM AND CORE-BANKING INTEGRATIONS

Core-banking, payment processing, card management, AML, KYC, CRM, and integration buses: the connective tissue a financial product depends on.

TEAM

Solution architect, senior developers

TYPICAL EFFORT

Scoped per project

06

## SECURITY AND COMPLIANCE DELIVERY

Security and compliance built into delivery rather than bolted on. Our work is independently certified to ISO/IEC 27001:2022, verifiable in the IAF registry. Work we've shipped has cleared audits by financial regulators, including FINMA and the Central Bank of Nigeria, and independent security firms such as Cossack Labs.

TEAM

Tech lead, DevOps, solution architect

TYPICAL EFFORT

Ongoing

07

## MAINTENANCE AND ONGOING SUPPORT

After launch: bug fixes, hypothesis validation, A/B testing, monitoring, and new features. Where the product needs it, we commit to SLAs on response and uptime. It keeps improving on real-world data.

TEAM

DevOps, developers, QA, designers (as needed)

TYPICAL EFFORT

Ongoing

# HOW TO CHOOSE A PRODUCT PARTNER

A short, vendor-neutral guide for anyone evaluating a product partner in fintech or banking — useful whether or not that partner is us.

Most engagements that go wrong go wrong for reasons you can spot before signing.

## LOOK FOR ONE TEAM THAT OWNS THE WHOLE PROBLEM

The most common failure isn't weak talent. It's strategy, design, and engineering in separate teams, handing work down a line.

Ask who stays on the product from first decision to live release. If the answer changes at each stage, context will leak at each stage.

## ASK FOR THE UNGLAMOROUS WORK, NOT JUST THE SCREENS

Anyone can show a polished interface. Ask to see architecture decisions, how they handled a migration without downtime, or how a build passed a security audit.

The boring artefacts tell you whether they can ship in a regulated environment.

## CHECK THEY CAN DEFEND DECISIONS, NOT JUST DELIVER THEM

In fintech, every product decision eventually meets a board, a regulator, or a risk team. A partner worth keeping can explain why, not just what.

Ask them to walk you through a decision they'd make differently now — the answer reveals how they think.

# HOW TO CHOOSE A PRODUCT PARTNER

### MATCH THE PROOF TO YOUR SITUATION

A case study only counts if it resembles your problem. A design-only engagement doesn't prove engineering depth; a greenfield build doesn't prove they can untangle legacy.

Ask specifically about work that matches where you actually are.

### WATCH FOR THE RED FLAGS

Promises of "end-to-end transformation" with no detail. Reluctance to name who does the work. Timelines with no ranges or assumptions. A pitch that never mentions what could go wrong.

Confidence is easy to perform; honesty about risk is harder to fake.

alty

# WE WROTE THIS BECAUSE THESE ARE THE QUESTIONS WE'D WANT TO BE ASKED.

If they're useful, the rest of this document shows how we'd answer them.