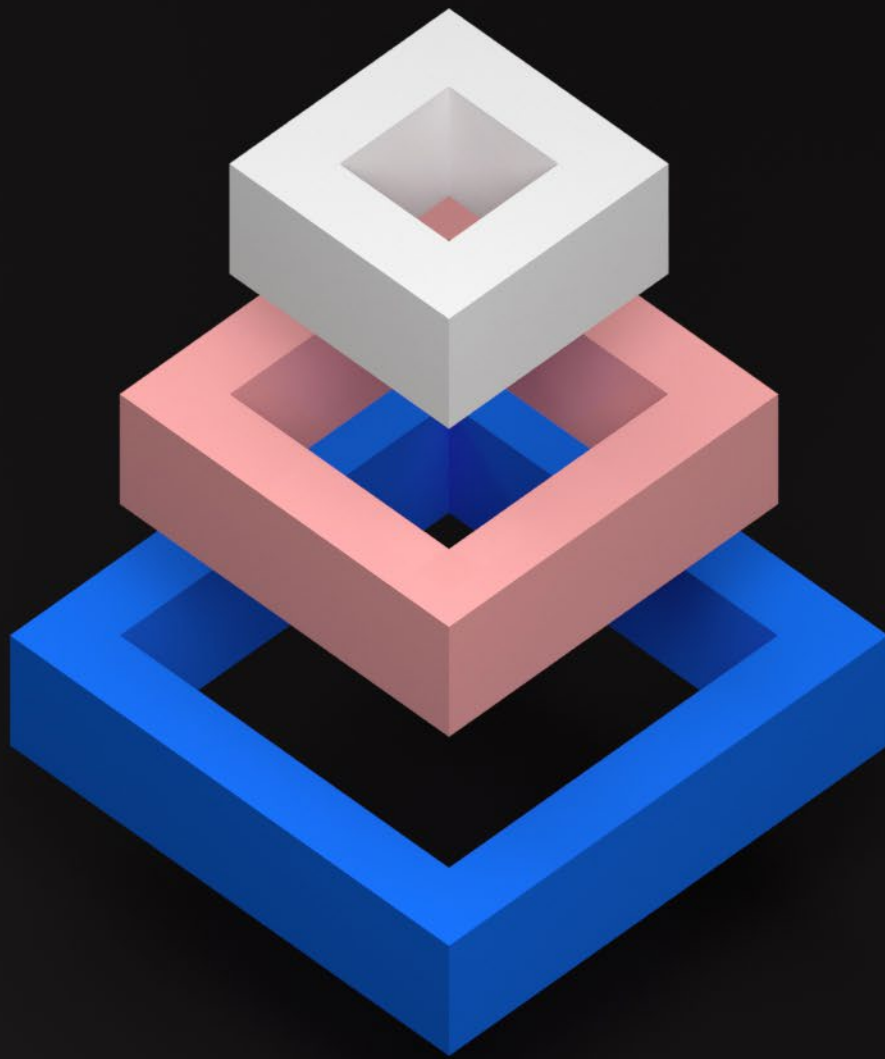


How to Build a Bank

— An Elsewhen Report



By Leon Gauhman, Matt Soczywko
and Nadav Mordechai

Copyright Elsewhen Ltd. 2021

Elsewhen

Introduction

In this report Elsewhen is sharing directly from our experience conceiving, designing and building digital banking experiences in the UK and across Europe, working both with incumbent institutions in the financial services sector, and partnering with fintechs and challenger banks.

The contributors and sources for the material within have both founded and led the strategy of successful challenger banks, and led on the digital transformation of incumbent banks.

Unlike a traditional consultancy approach to strategy where the interaction exists entirely at boardroom level, we are experienced in practically employing the strategies described here, turning them into feasible product roadmaps and leading on the tactical implementation.

By bringing strategic thinking closer to the eventual implementation, you can feasibly reduce the risk often inherent in technological opportunities. This is key to creating world-class experiences utilising modern technology, whether in banking or beyond.

We will share the insights garnered from that experience both helping to devise and build new propositions for partners, and from studying the very best players in the market. We will explore why building a bank is still a worthwhile endeavour and for who, as well as looking at the reasons so many neobanks fail before they've launched, and how this can be avoided.

We will break down the structure of a modern bank—the solution architecture, the many product options at every layer, how to evaluate the right vendors for each, and why that might change depending on your goals.

Understanding the full picture like this is the only way digital strategy can be successfully employed. To unpack the opportunities inherent in building a new banking proposition requires an ability to see clearly through the different layers of technical complexity: to understand where the risks are, and keep the vision tightly coupled to the reality of execution and feasibility.

We will run through what it really takes to build a bank, covering the different parallel and connected workstreams of product, design and technology. We will take you from day one of identifying the right 'Jobs To Be Done' to tackle, to developing a detailed strategy to follow, and planning out a 5/10/20 year product roadmap of successful product launches and constant iteration - combining our own technical knowledge and best practices.

Please note that this report is not applicable for territories beyond the UK and Europe, either in terms of the regulatory environment, or the supplier ecosystem.

How to build a bank

What is a bank?	5
Why you should read this whitepaper	8
Market context	9
Why build a bank?	12
Adopting a non-traditional approach to touchpoints	16
Why neobanks fail	20
What does it really take to build a bank?	24
Identifying your customer's needs	27
The structure of a modern digital bank	31
How to evaluate potential vendors	36
Solution architecture	40
Core banking	41
Customer onboarding and Compliance KYC/AML	41
Payments	42
Data lake	43
Cards	44
Back office	45
Treasury	46
CRM	46
Growth	47
Customer success	49
Reporting	49
Who are Elsewhen?	50
Get in touch	51

What is a bank?

Asking 'what is a bank' may appear on the surface facetious, but understanding the answer is useful in navigating both this report and in developing a strategy to launch a successful bank.

A 'bank' is a service you provide to a customer segment. If you're reading this, you're probably envisaging a digital proposition, and it's true that a bank need not have branches, but it also needn't have a mobile app, or a sleek GUI for customers at all. Your first job is to speak to customers and determine what it should have.

What constitutes a bank, or where the bank begins and ends, may be difficult to define in a given case. It may even be unhelpful to obsess over. In the case of the Apple Card, who is the bank? And why is it important? As long as the partners are happy and have a gameplan for the future, and the customer is delighted, that is what matters.

What is interesting is that with the development of Open Banking and the growth and proliferation of fintechs, the idea of what a bank constitutes is shifting towards a componentised structure. A structure that allows what was once locked up in an ivory tower to be integrated, shaped and 'remixed', challenging the de facto status quo and providing new and exciting opportunities for the benefit of the customer.

Core banking' also needs to be redefined and clarified. For incumbent retail banks, core banking can cover everything from transaction accounts,

to different lines of credit in the forms of loans and mortgages and the ability to transact, and beyond that to the CRM, Treasury, compliance, and so on.

In the age of digital, we would argue 'core banking' instead describes the minimum set of functionality required for a banking solution—an account to hold a balance, a ledger to monitor changes in the account, and a connection to a payment system to transact—and how those are fulfilled is up to the various choices you make, some of which we have outlined in this report. At the time of writing we still haven't seen an approach that at the very minimum would exclude the need for these key components.



Why you should read this whitepaper

The insights captured here are relevant to any decision maker with the remit of business growth or technology in financial services, or those in other sectors who understand that their place in their vertical value chain means there is an opportunity to pursue a move into financial services.

Whether you are working within an incumbent organisation, and meaningful change must be made, or you are representing a newcomer who is looking for strategic insight to help enter the banking market, you should find this report apposite. Or maybe you just want to compare notes.

This paper reflects Elsewhen's approach to digital strategy. By narrowing the gap between the strategic thinking and the eventual implementation, or the thinking and doing, you can reduce the overall

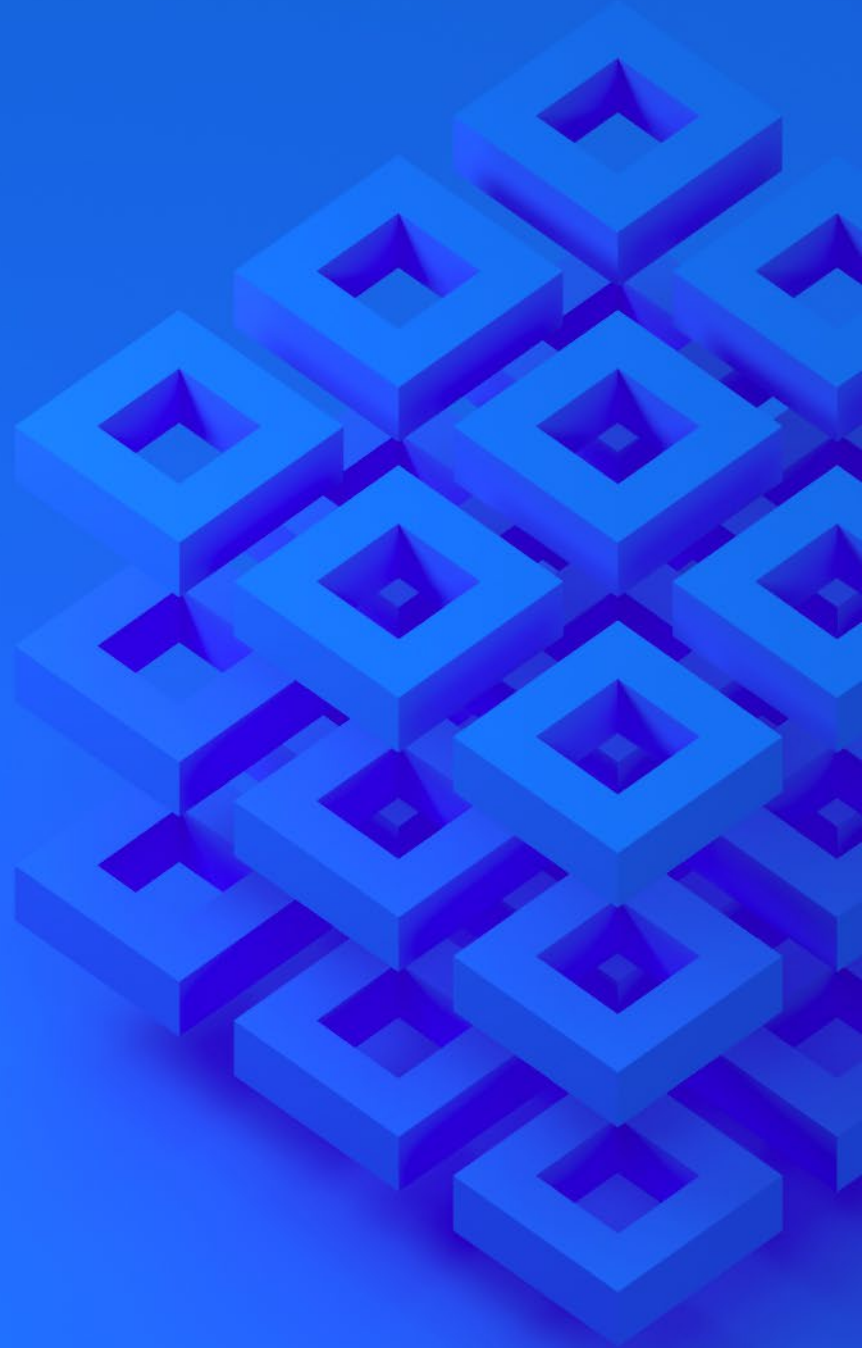
risks associated with technological opportunities.

This is key to creating the kind of world-class customer experiences that the banking customers of the future will expect.

We have not held anything back. With the right team in place, anyone could follow the guidelines found here and build their own banking solution for their own identified niche.

That is not to say that the bank they build will be successful, and with that in mind we will cover off the main reasons that neobanks fail—from limited differentiation, failing to meet customer expectations, building brands that failed to connect with customers, having a weak profile, and so on.

Market context



The increase in digital or 'direct' banking

Over the last five years, the customer base of digital-first banks has grown steadily every year. Over the same period, the number of customers using physical branches has declined at the same rate. This trend can be observed globally, and different markets can be understood to be at different stages in the move to increasingly online-only banking.

At the time of writing, the COVID-19 pandemic has seen a dramatic acceleration in the move to online services across the globe and across all industries. It has brought to the forefront were once considered predictions of future trends. The death of brick and mortar, digital-first banking, a cashless digital payment-driven economy; all of these and more are now here to stay. Banking is one of the leading industries irreversibly changed by COVID-19, and is leading the change.

A competitive environment

For years now, concern has grown among challenger banks and their potential backers around the increasingly crowded market in the UK, which has proved increasingly deadly to neobanks hoping to flourish in the retail banking sector.

Digital banks obviously have convenience on their side, and they are also cheaper for the end customer. The old barriers of trust and security, which have

previously precluded a move to new banks for most customers, are also slowly fading. While the market share for neobanks is still tiny no matter what market you consider, they are growing at a much faster rate than the incumbents.

The lack of frameworks and risk appetite

As we explored in our report on the state of Crypto Banking¹, in many emergent niches there is a low appetite from the incumbent banks and a lack of established risk frameworks. This leaves the ground open for niche players to take advantage.

¹ The banks servicing crypto in 2020, Elsewhen Ltd.
<https://reports.elsewhen.com/cryptobanking/>

There will be a stigma with being new—you must be transparent and communicative.

An interview with a potential customer for a corporate neobank

Why build a bank?

01

There are (still) underserved segments

The neobanks that grab the headlines tend to be retail banking offerings who have banking for the mass market in mind, and have their sights firmly set on the salaried current account customers. In the UK this segment saturation is high, and it is still unclear in many cases how the banks will generate enough revenue from their customers, and with what revenue-generating products.

There are myriad niches of underserved customers with specific needs and priorities from their banking. A new player tends to need an insider's understanding of their target customer segment to begin to hypothesise about a new product or service for them. A segment-tailored proposition will allow for new players to create a loyal customer base and differentiate itself from other players in the market.

Such segments examples would be servicing a certain type of SME or the peculiarities of a certain geography with many underbanked cohorts.

02

Leverage a vertical's value chain

It could also be the case that the player enjoys significant control over a vertical value chain where transactions take place, and is looking to increase its control and open up new revenue streams—or stickiness—by plugging in banking services to its customers. For example, a telecom provider teaming up with a lender to offer new finance options to its existing customers.

03

Regulatory barriers are lower than ever

Regulators in the UK and Europe seek to promote healthy competition and better outcomes for the customer (though acquiring a full banking license is still far from being a speedy process).

Barriers such as regulatory capital requirements or attitude toward cloud-based applications are more relaxed than they have traditionally been. Advanced regulators are offering different forms of licenses like the e-money license and there is a growing appetite to regulate crypto assets. That allows banks-to-be to start small, scale up with time and seek their full banking license once the business is in operation, rather than waiting months to be granted their full banking permit.

04

An abundance of off-the-shelf components

As we will explore later in this report, there is an abundance of off-the-shelf software to provide the different functions of a bank. This can dramatically reduce the time to market and also reduces the innate risk involved. The challenge instead becomes navigating the ecosystem and choosing the right vendors for your bank.

05

Progress begets progress

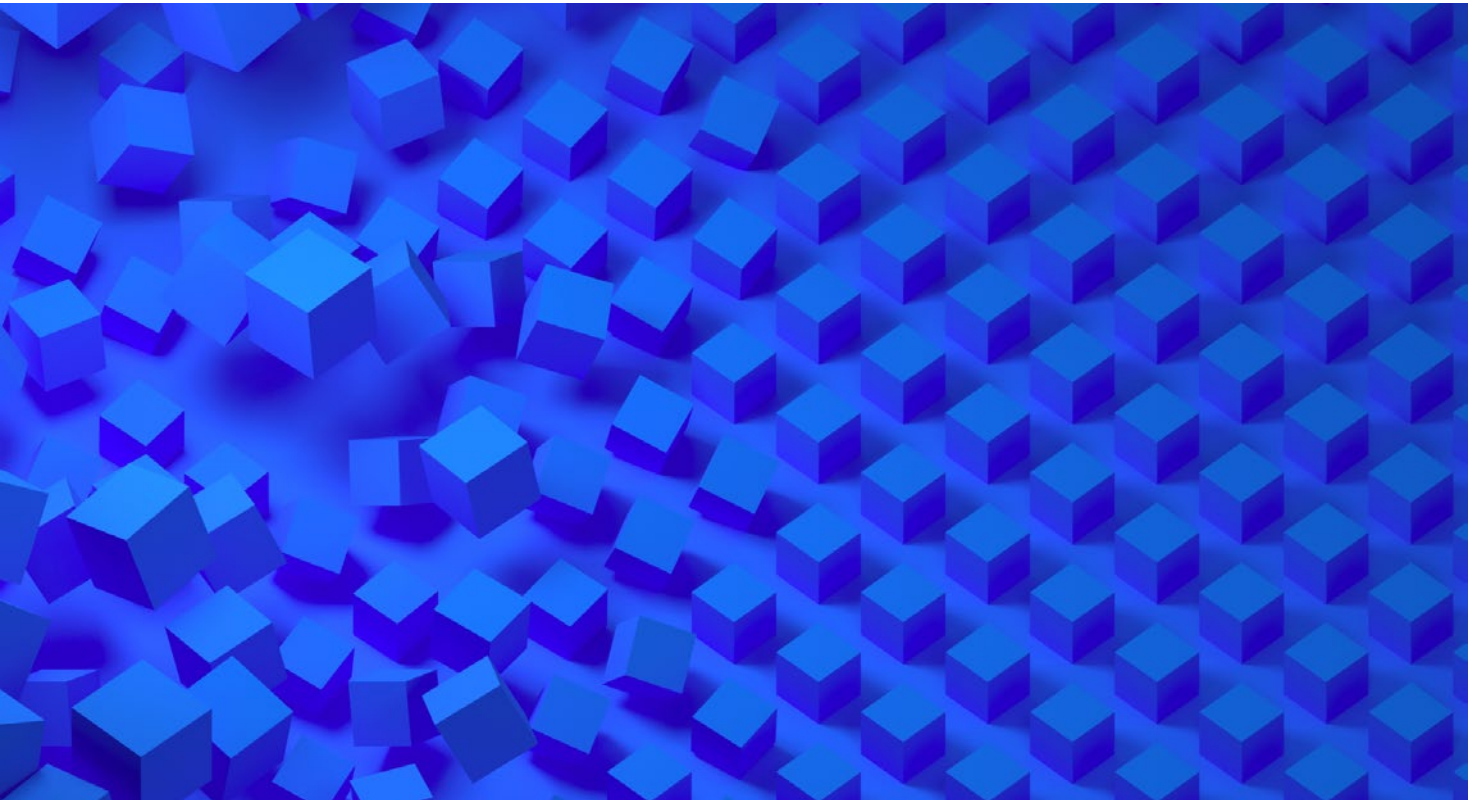
If we consider the different stages of innovation in banking, from the move from branch-based to direct banking, we are only at the start of digital-only services. In most cases, banking customers are doing online what they used to do in branch. As we will see, with standards for the underlying solution architecture vendors becoming more and more established, and potentially in the future gaining access to banking rails from incumbent providers, genuine innovation can begin. New companies will emerge to fundamentally rethink the banking products that have held sway over the last century. New vendors will emerge to handle these individual elements, and the cycle will continue, only now with the rate of progress accelerated.

In conclusion, a robust new digital bank can outmanoeuvre the legacy systems of incumbent banks easily, and pave the way to lower and lower costs, new customers, and the easy introduction and rollout of new products and services.

06

The potential of data

There is a huge opportunity in developing data-driven solutions and when you look at the landscape of the current generation of neobanks very little is actually done in this space. However this is not a chicken and the egg problem.



Adopting a non-traditional approach to touchpoints



There is a growing opportunity for banks, pre-launch right through to incumbent, to rethink the channels and touchpoints they utilise to maintain and grow their customer base.

A well executed strategy will allow you to 'own' the customer, i.e. to own the banking relationship over a competitor. This doesn't mean you need to own the actual interaction with the customer: instead you have enough data to track and serve the customer across different touch points.

Banks, or more accurately manufacturers of financial service products, will tend to inhabit certain niche markets for which they have developed real expertise. These players, or for example providers who have unique automated digital underwriting and servicing capabilities coupled with good APIs, can execute what's become known as the Invisible Bank strategy. This essentially means offering their product or service to any digital channel that can consume their API and offer it to their customers.

API-as-a-channel / The Invisible Bank

API-as-a-channel is a strategy to 'own' customers in this way, where those banks that already have the APIs and underpinning engines and can handle the capacity, can negotiate with technology companies to provide aspects of banking without the visibility of the customer. Obviously the API is just the 'pipe' so this strategy is meaningless if the API is not underpinned by some useful product capability - advanced digital onboarding, underwriting, fulfilment, decisioning engines, and so on.

A high profile example of an incumbent employing this strategy is of course Goldman Sachs and the Apple Card. This is a vertical strategy where Apple owns the customer and Goldman Sachs is providing the capability. Goldman has built the technology capabilities, API and an automated underwriting process that allows them to plug this banking product into Apple's technology, with the card technology provided by MasterCard.

Strategy in action: Cross-border payments

An example of this strategy in action would work in the following ways: an incumbent telecom provider like Vodafone could consume the API of a vendor that offers cross-border payments. As far as the end customer is concerned, they are using a service provided by Vodafone, i.e. an app on their phone. In fact the underlying service is provided by the payments vendor that provides the same service to other sectors, who could adopt the same strategy. Another potential implementation would be an aviation company who would have the same visibility and ability to deliver an app with a card to their customers.

The move from transactional, to customer-focused banking

All players, incumbent or neo, will continue the transition from a focus on the product-led (e.g. credit products - cards, overdrafts, mortgages) to a customer-focused 'customer centric' banking. There are various strategies immediately available, depending on the capabilities and specific niche or potential competitive advantage of a given player:

- Try to own more touchpoints with the customer, for example by building platforms with FS, providers can build and launch products, creating a marketplace of greater choice for the customer where more customer needs are met.

- Concentrate on specific value chains in your niche, and work to own more of the value chain, for example via acquisitions, and thereby increasing overall profitability by swallowing the margins of the various players through the chain.
- Rethink the touchpoints you have with a customer, for example your API. API-as-a-channel (Invisible Banking) where a third party technology company owns the relationship with the customer, but you are omnipresent in the customer's life as you are directly providing the banking services.

Why neobanks fail



Across Europe and in particular in the UK, the banking market is highly-competitive. Traditional players are racing to reinvent themselves and digital-first challenger brands are seeking to steal market share.

As a result, there will inevitably be more casualties before the sector settles down, but the reasons neobanks fail tend to be misstated, with history often rewritten by press releases. Below are some of the real reasons challenger banks fail to gain traction.

Limited differentiation

The UK banking sector in particular is awash with challenger brands that lack differentiation. Recent retail upstarts like Bó never offered anything customers couldn't already get from gold standard neobanks like Monzo and Starling, while German challenger N26 admitted it would be a challenge to differentiate its offering in the UK.

Savvy challenger banks must either target underbanked demographics, fulfil a specific need like Monese does, or, in the case of Revolut, offer various 'value-add' features outside of their core banking, such as crypto or forex services without the standard industry fees.

Challengers are not launched into a vacuum. Reviewers said Bó felt like "the first attempt of a fledgling start-up rather than the cutting edge of the fintech boom". By contrast, Monzo, Starling and Revolut had already spent 4-5 years working out the glitches in their offerings and building attractive new features.

Failing to meet customer expectations

In a competitive and well-established neobanking market like the UK, and to a growing extent much of western Europe, consumer expectations are extremely high. Failing to meet those expectations can lead to a quick demise for a new challenger bank. Bó, for example, ran into difficulties around accepting driving licenses and passports as proof of ID. Later it was forced to reissue 6,000 debit cards to comply with EU rules. Some observers also noted at launch how the app lacked some key standard banking functions – for example, not enabling people to get their salaries paid into the account by BACS. Another big-name challenger, N26, struggled to match the user-experience ratings of its peers in the UK.

Building brands that failed to connect with customers

Staying on our theme of high-profile retail failures, Bó invested heavily in branding; working with Accenture Interactive to build an app and brand that aimed to start with customer experience but ultimately failed to connect with consumers. The Telegraph's conclusion was that Bó's branding was clunky and confused. But the brand's inability to capture the imagination was also symptomatic of a deeper shift taking place in financial services.

For the most part, young consumers don't trust established brands and don't feel like they are being listened to. Other more successful challengers built a community of loyal customers. The reason Monzo's customers are such strong advocates is to do with radical transparency, tone of voice and commitment to its community. It's a similar story at Revolut which asked customers to help guide the brand.

Failure to connect isn't just an issue for digital spin-offs of incumbent banks. N26 spent extensively on tube (London's Underground rail network) advertising but still failed to make the top 15 fintechs in the UK by monthly active users.



Weak profile online

Banks new and old have traditionally emerged to serve specific communities, and flourished with the support of that community. Before it launched, Bó was in trouble over accusations that insiders had allegedly been faking reviews for the platform. That kind of attention is unhelpful when your core market is a younger demographic of super-savvy digital natives with whom you aim to build a community.

Bó also failed to secure the kind of authentic relationships achieved by Monzo, which organised hackathons and invited tech influencers to share its app via word of mouth and Twitter or Revolut with its RevRallys for its community. In the US, challenger bank Chime has also won fans among young audiences by using social media (e.g. #ChimeSavedMe) to prove it is delivering on core promises.

What does it really take to build a bank?

Let's start with the why

To support an Agile development process and a quick time to market—and before you can begin building a bank—you need to run a 'discovery' phase.

- Following what is sometimes called a 'Proof of Value process for your proposition
- Mapping out how you will put together the underlying architecture of your bank
- Determining the design direction you will take

This is about validating that what you are proposing is valuable.

Based on ideation workshops, strategic discussions with your stakeholders, user interviews with your target customers, and potentially with meetings with players operating in the same space, you will emerge with a clear vision for your bank and why it should exist. You will establish answers to the key questions that will help you further down the line when you are deep into your product roadmap, leaving any risky assumptions behind.

At the end of the Discovery phase, you should have a pertinent summary of your research with the key findings, the major pain points, and the financial challenges you are solving for your chosen customer cohort.

As well as a chosen direction, you will likely have several alternative solutions for building your bank, with the technical architecture and comparison of the potential vendors that will be fundamental to delivering your featureset, in a clear comparison that prioritises the on-time and on-budget delivery of the MVP.

The digital product proposition

You should emerge from the Discovery process with a solid vision of the digital product proposition.

This has to do with the entirety of the consumer proposition, rather than just the internal operational processes.

You need to validate the need in the market for the proposition, and exactly what it is. To do so, you will identify user needs and validate them; by identifying the potential customer segments, analysing the user needs for each, and ranking the opportunities. Your research should have a certain focus; concentrating on a core need, and how you will serve it.

You will need to understand the market from plan to launch while considering: what are the market characteristics, i.e. what are the market drivers for your customers:

- Are they underserved? Why or why not?
- Is your customer base growing? Why or why not?
- Does the regulator have any role to play - will they look favourably on what you're trying to achieve?
- Are there market trends you can point to as rationale for launching?
- What is the view of the incumbent banks?
Are they ignoring your niche, or are they trying to do something similar? What can you learn from them?
- What existing products or combination of products are your customers currently using instead?

You will need to understand your target market, and not just the entire servable customer base.

You will likely have more than one potential cohort within your chosen segment, with different sizes and turnover/spending power - you'll need to understand the quirks of each.

From here you will visualise a potential solution, which you can validate—both the ideas and visual concepts—with these potential customers.

You should have the following:

- The key findings from your customer research that your bank will speak to
- The main customer activities and tasks that your bank will handle
- The core feature set that will help your customers
- The primary impacts for your customers e.g. a decrease in unplanned overdrafts
- Your bank's USP
- Where your bank sits in the existing ecosystem of competitors

Identifying your customer's needs



Establishing the Jobs To Be Done (JTBD)

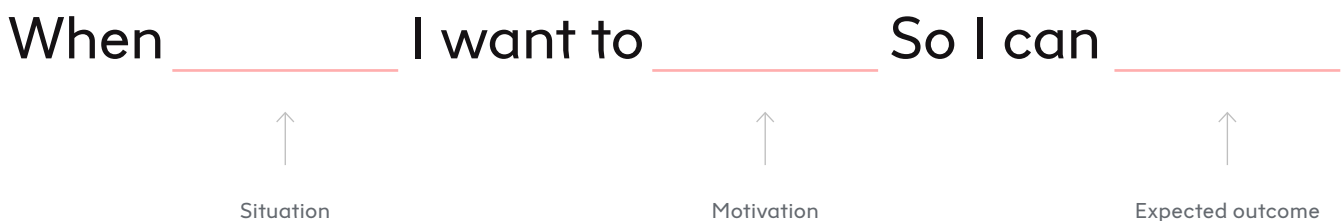
It is vitally important that you establish the needs of your customers, in order of importance that they define, not you.

Before deciding on your featureset, and determining how to develop it, you'll need to identify the most important customer needs and the key journeys to satisfy them. In other words, you need to start actually talking to your customers.

Once you've done that—conducting customer research by spending time listening to your potential customers—you will need a framework to articulate what you heard, and feed it into the follow-up product process. A straightforward way is to utilise the Jobs To Be Done framework, invented by Harvard professor Clayton Christensen, which will allow you to identify the real opportunity behind customer behaviour.

Defining what customers are trying to accomplish (their 'jobs to be done'), using JTBD helps you find early product/market fit, and prioritise product features based on their value to customers.

Once you have identified the user's 'job', the framework will help your team put that job in context and focus on what the user is trying to achieve, and how it will better their situation. The creation of jobs follows the standard and well-trodden format, which is easy to consume by stakeholders and members of the product team:



Epics and MVP

Using the JTBD you have identified as your data point, you can now define and prioritise key epic-level product features. These epic-level features should be high level - they are just descriptions of desired outcomes and solutions to help users with their jobs.

These are the starting points of a product roadmap, where the emphasis is on defining a clear MVP (Minimal Viable Product—see the next section 'What is an MVP of a bank?' for more information) of what is required for a launch product for your customers.

Designing a 5, 10, 15 and 20 years (aspirational) roadmap

You should define your product roadmap in terms of features and product launches, meaning you have the types of features e.g. channels, and for each launch the features that live underneath that. For example, under 'channels' and for the MVP, you may plan to launch with a responsive offering, but for the third iteration of the product you plan to have a native mobile app.

Your roadmap should be based on a combination of what your customers want, what's feasible, and what will allow you to get to market quickly.

Your roadmap should not be subject to meaningful change. If it is, the chances of successfully launching on time and on budget rapidly disappear. It's an obvious point, but worth remembering and prioritising, as many neobanks fail pre-launch.

Your product roadmap should prioritise the MVP. You go to market with an MVP with a clear proposition for your target niche and start acquiring customers. You should build the solution on a combination of third party vendors for both the core banking and value-add features.

Branding and cross-channel design

Adjacent to your marketing and go-to-market strategy, you will need to zero in on branding and cross-channel design that resonates with your customers and reflects your vision for the bank. You can of course work with a partner here, or undertake an internal brand sprint, bringing stakeholders and potential customers into the process.

Your branding and cross-channel design is about creating a set of values that would be the foundation for the visual territory. Based on that, you can design a full graphical language for the business—from the digital portal, to decks, merchandise and even business cards.

The structure of a modern digital bank

The technology stack

Depending on the particularities of your proposed bank, you can build on your recommendations with a broad market scan and due diligence for suppliers. You will need to identify the tech partners that can deliver the solutions—the core functionalities, the day one features, and the future value-add features.

It is recommended that you undertake a comprehensive technical due-diligence of the vendors' solutions, including their tech stack, architecture, and flexibility. We will elaborate further on how to evaluate vendors later on in this report.

Your team should all be familiar with the high level solution architecture and have a clear understanding of why this is the proposed route and the preferred vendors, even if they lack the technical understanding to challenge the detail, e.g. the data processing or microservices infrastructures.

Even if they are not being introduced on day one, you should have a map of the additional products and services:

- The service
- The vendor/partner
- The relevance to the product
- The operational and product readiness, i.e. ease of integration
- The commercial considerations for each
- Modern processes have made building easier

Modern processes have made building easier

In the consumer world, no one builds software like they used to three years ago, and certainly not how they used to 10 or 20 years ago. The radical changes in consumer software processes can and should inform enterprise software processes: you can give your customers the exact bespoke CX they need, and do it at a speed and cost that's better than implementing a giant off the shelf solution.

Modern software building—banking or otherwise—looks like this:

- You can go from whiteboard to high definition design, no wireframes are needed
- Code-based prototypes for validation of technical feasibility, not just UML diagrams
- DesignOps utilising design systems with automated handoffs to developers for robust collaboration
- A modern DevOps process, i.e. code can be deployed several times a day across environments with fast rollbacks
- Infrastructure that is 100% cloud-based, with reliance on managed services where it makes sense
- Separation of concerns via a domain-driven design to varied levels of abstraction, not microservices for microservices' sake, or passing off monoliths with microservice-like names
- Container-based infrastructure with autoscaling and provisioning e.g. Kubernetes
- Use of serverless technologies like Function or Lambda
- Polyglot backends that are selected for their fit for a particular problem
- Componentised JS framework like React
- An open and honest culture that values independent thinking and quality
- Product and business having an equal say at the table
- Modern collaboration tools across the entire process for stakeholders, designers and engineers

This combination gives you the tool box to deliver bespoke software fast and in any shape you want, from enterprise down to the leanest MVP.

As we will explore in more detail, there are now many well-established and robust FS SaaS solutions in the market such as Mambu, ComplyAdvantage and Jumio, which can be part of the solution architecture. The key here is to use the right vendor for the right job while making sure that they are abstracted and plug and play.

The goal is to make sure that the needs of a given vendor implementation don't leak into the solution architecture, so that any vendor can be removed or swapped out for another vendor easily. You should avoid anything that creates a legacy problem from day one, such as a horizontal vendor integration approach, where a particular vendor is completing multiple jobs. For example, a situation where Salesforce becomes the hammer for every nail is a disaster waiting to happen.

You should also have clear separation between what you consider the 'core' of your bank—as we defined it earlier on in What is a bank?—and the additional planned future requirements and functionality, or the entirety of the product roadmap.

The underlying structure, or solution architecture, of a modern bank can vary widely depending on the context within which the bank is launched, and the specific points of difference of the proposition, but they all follow the same fundamental structure. To help elicit understanding on the roles of the different required technologies, there is a diagram below:



What is an MVP for a bank?

Working on digital products in a regulated environment and dealing with financial transactions means that there is a lot less room for 'moving fast and breaking things', to borrow an old startup phrase. Therefore, defining an MVP in this context needs to take that into account and needs to be structured differently.

Where in general terms your minimum viable product describes the minimum feature set you can launch with to satisfy customers to be able to validate if your proposition is desirable, it is different here. In financial services, it describes the minimum feature set that will be required for launch from the perspective of being able to satisfy the conditions for being a licensed entity and validate your value proposition.

Note that as a result, increasing the set of features for launch is detrimental to your ability to launch. The faster you can get to start shipping to real customers the faster you will start to learn if what you offer them has actual value. Over the following pages we have outlined a Solution Architecture that we would consider as the minimum required to launch a banking proposition.

How to evaluate potential vendors

We have here provided an explanation of the fundamental aspects of a modern banking system. As your organisation grows, your product portfolio will need to grow too, to cater for new customers requests and market opportunities.

Evaluation criteria

- Product criteria
- Security
- Scalability
- Maintainability
- Ease of integration
- Quality of documentation
- Update frequency
- Ability to switch providers
- Tech stack

Vendor/company criteria

- Company size
- Financial stability
- Experience in the field
- Support and SLA

Internal feedback criteria

- UI
- Access control
- Compliance
- Confidence

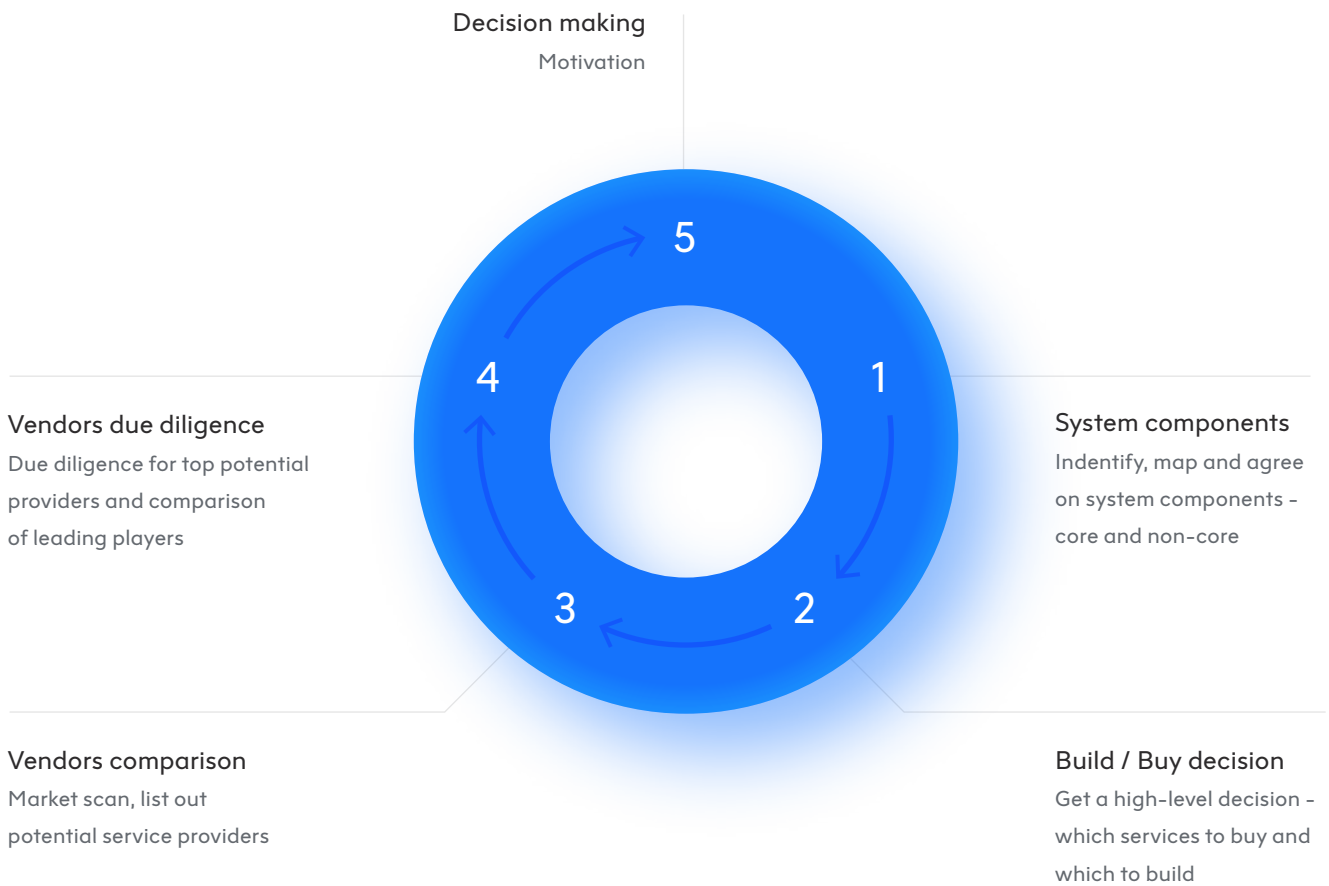
The decision on which part of the stack you will build and which to buy is key in ensuring that the new bank stays nimble, agile and yet flexible enough to pivot or add new products.

New players in the market are offering a range of services, rails and foundations for a new bank, and the right mix of those will contribute towards a shorter Time to Market of your proposition. We have not made any specific vendor recommendations here, but there is a useful set of criteria you can employ to find the right vendors for your proposition.

You should create a table of the potential vendors for each product (the products are listed below) with the suggested evaluation criteria and its application to the vendor selection process. Each evaluated vendor will get a score on each criteria and overall weighted score. These scores should highlight which are the preferred vendors.

Evaluation methodology

Now you have established the criteria and started to research the potential players, you can follow this process:



Banking as a Service

Some of the biggest retail challenger banks in Europe are built on Banking as a Service providers for their core banking, and for good reason. A BaaS provider is one of the fastest ways to get a new banking product to market, so you can concentrate on quickly bringing other features to market and providing a great customer experience.

However, this speed comes at a cost: an off-the-shelf BaaS will lack the flexibility of other configurations which creates a legacy issue from day one, where sooner or later you may need to 'grow up' and migrate.

With that said, it is possible to find a BaaS vendor where you are able to pick and choose which parts to use from their offering. Either way, if you can rely on a solution that has been used many times before it will speed up pulling together the Solution Architecture greatly.

Solution architecture



Core banking

What a given player may consider their 'core banking' stack may vary, but in the digital era there is actually a clear definition, regardless of proposition. Core banking is the functional base which you can add products to: account, customer data, ledger, transactions.

This 'core' captures the minimum set of requirements for the bank to function. Entities like people or companies, a ledger to keep track of movements, a product like an account, transactions to and from the account, and a record of parts of the payment process or documentation stored for the entities you are banking.

Ledger

Any system handling financial transactions requires a ledger. This is the part of the stack that allows the tracking of transactions across the different entities in the system. A typical set of use cases for a ledger would be:

- Product control
- Parts of the payment process
- Accounts
- Entities
- Transactions
- Rates and calculations

Depending on the simplicity of your use case, there is the option to build a ledger from scratch. Outside of simplicity, this might also be the only option for

products in which a specific offering is required.

This is an area where it really depends what features you've determined to be mission critical.

The pitfalls of building your own are around stability and reliability. A calculation error in the core of financial services bears high reputation and regulatory risks, and can draw attention to your business for all the wrong reasons. Conversely, as your product offering grows, it may take additional resources to modify something which, for daily operations, may be essential.

As with almost everything in your solution architecture, it is highly likely you can adopt a third party ledger with the option to migrate from it at a later time.

Customer onboarding and Compliance (KYC/AML)

The ideal approach to KYC/AML minimises the workload on the customer side during onboarding but provides modern best-in-class capabilities for a robust function behind the scenes. Adopting best-in-class security and taking a vigilant approach to fraud is essential to any successful bank, and this starts with your approach to KYC/AML. Happily, there are many well-established vendors in the market which you can integrate.

While this is a large area of concern, it can be broken down into certain key considerations, and KYC/AML can be componentised just like everything in your stack. The main components you need to consider are:

- ID&V - this is to provide the ability to conduct identification and verification of individuals using digital methods (including biometrics). Using a digital identification process can help to dramatically shorten the onboarding process for new customers and create an up-to-date customer experience, ultimately resulting in higher conversion rates and more prospects becoming clients. While ID&V are easy to perform on individual account holders, solving for more complex cases, such as business banking or unique holding structures, is not an easy task. You might want to define your most common use case, and evolve your identification capabilities with time.
- Address verification - this is to provide a digital verification of addresses.
- Screening (PEP, Sanctions, Adverse news) - these are the datasets that are used in screening processes. As part of your AML (Anti-Money Laundering) checks, you will need to ensure (and be able to prove) that you have performed comprehensive checks on your customer base. Screening tools are linked to large data bases, and will help you identify the risk level of a specific client.
- Transaction monitoring - provides the ability to monitor transactions in real-time as well as identify historical trends for suspicious activity with regards to money laundering or fraud, providing the capability to stop and/or flag

transactions for review based on a set of rules set by the business.

- Payments screening - this is the real-time screening of payments (before execution) to monitor higher risk payment information and prevent payments to or from potentially sanctioned customers. Again, here we would recommend Comply Advantage.

Payments

We could easily fill a white paper on the subject of payments alone, as it really depends on the complexity of your payments solution, but there are some fundamentals to bear in mind which you can follow. Our advice to a beginner would be to rely on a reputable partner if you can, and to at least stay away from launching with cross-border payments unless you absolutely must. You will need to be linked to a domestic or international payment scheme for your bank to take and make payments.

It is advisable to rely on third party rails to provide the backbone for payments. Many BaaS (Banking as a service) providers have suitably robust payments infrastructure. In cases where the BaaS architecture is designed to be decoupled, the payments component can be lifted in isolation for the rest of the solution and used in your architecture.

Payments schemes are one of the oldest technical implementations, as they were one of the first aspects

to be enabled by technology. Some payment schemes still require a physical cable to be connected to your servers. They are not following any modern principles in terms of protocols. They are essentially bespoke messaging formats themselves.

The payment processing itself is a complex process journey that requires a very robust system for checks and balances. Security-wise, it's a very common vector of attack as this is how your clients' money is becoming connected to the outside world.

Payments is also an area of high regulatory scrutiny for the same reason—this is essentially where your bank is connecting to the outside world. The money laundering mechanisms are tightly coupled with the payment processes to make sure that you are not recipients of, or passer-ons, of 'dirty' money.

These are the main payment schemes to be aware of:

UK-based

Faster Payments: The newest of the schemes, Faster Payments normally allows transfers in a matter of seconds.

BACS: At the other end of the scale, BACS is the oldest and most popular scheme. It takes three days to clear a payment, and is the only scheme that can be used for Direct Debit and Direct Credit.

CHAPS: This is generally employed where a high upper value ceiling is required, and also offers slightly more speedy payments which clear on the same day.

International

SWIFT: SWIFT allows for international payments between banks that have a direct relationship.

Corresponding banking - is the relationship between two banks that allows international money transfers.

These need to be established between banks or the payment needs to pass through an intermediary.

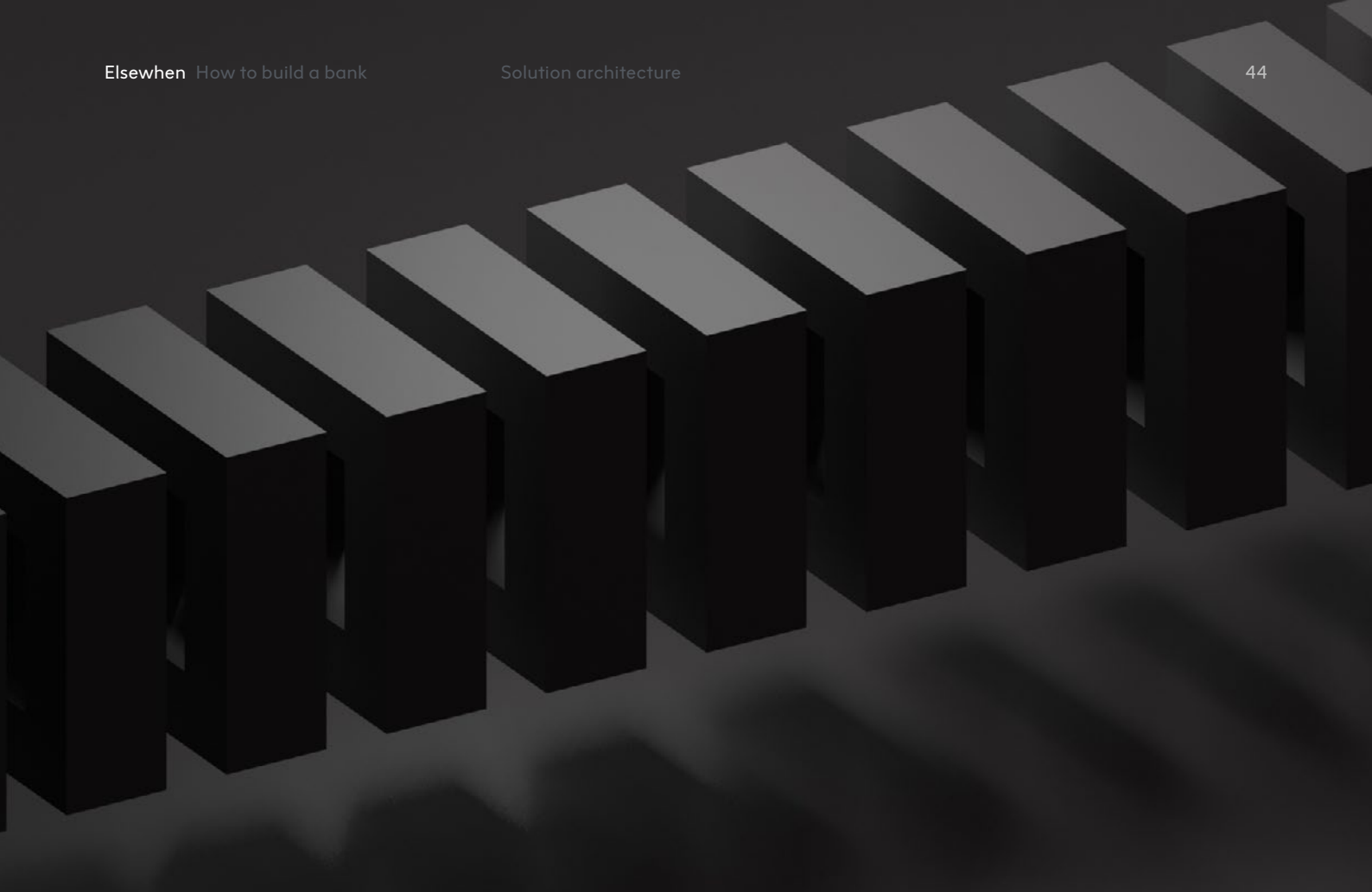
Some transfers require more than one intermediary.

SEPA: This is a new scheme for money transfers between the 34 countries in the EU 'SEPA' zone, with transfers made in Euros. In cases where a direct relationship between the banks making the transfer doesn't exist, the payment passes through the European Central Bank.

Data lake

Data in financial services can be a real burden due to the scale and sensitivity. To be clear, the issue here is not data as it is stored piecemeal across various systems in the architecture in databases, but instead the data that is produced in the lifecycle of the bank being a living system. This includes the customer lifecycle, how your customers interact with your product, and the metadata around the financial products. All of which produce enormous amounts of data.

Ultimately you will want a real strategy around data, i.e. a plan to use this data in a way that will improve the customer's life and provide innovative solutions that increase the bottom line, but first you'll need to determine how to store it in a way that is cost effective



and will facilitate future use. Initially it's about how you store the data, not what you do with it. The structure of the data doesn't need to be defined at the point that the data is captured. Your initial approach needn't consider how the data will ultimately be used, but only how it can be stored safely for future use. You need to be able to store the data and do so in a secure and scalable way from day one.

Your goal is to create an area in the architecture where data can be collected into a "lake", with the data stored with "flat" formatting, which will provide it back to other systems that will want to use it for analytics or machine learning.

IaaS providers (GCP, Amazon) have for some time provided solutions allowing you to very cheaply store

huge amounts of data with different frequencies of retrieval, for example on-demand, daily etc. There are also various "playbook" architectures that can be assembled quickly. With minimal additional code, these allow you to combine IaaS products, with the scaling and management completely offloaded to the IaaS providers. This data can then be inspected with insights packages (Grafana, Tableau etc.), enriched by third parties and provided via pipelines into machine learning engines.

Cards

Like almost everything in your bank's architecture, card provision can also be procured through a third party and even this has become a competitive space. There are multiple players and different types of players, with both BaaS providers and specialists

being potential options here. What they all share is that the third party will take your card design, print the required account codes on the cards, and provide them to your customers.

As with every other aspect of your stack, it's recommended you undertake an analysis on the potential vendors and types of vendors and evaluate which would be the best for you, as explained in our introduction to vendor evaluation. Product differentiation for cards should be carefully weighed up, as it can be a costly exercise, with metal cards notoriously difficult to produce.

Back office

Your bank's architecture will have two primary customer types that need to use it. The first is the customer in the traditional sense - anyone external that is using the bank's services. The second is the internal customer.

The internal customers are the bank's employee that require access to underlying processes for the successful day-to-day running of the bank. For example, approving a payment that the system flagged as suspicious, manually approving the final stage of a loan application, understanding a customer's balance, and gathering financial information on the bank's performance.

You will therefore need to provide an interface for these internal customers that will allow for some manipulation of the software that runs these

underlying processes in your solution architecture. To do so will mean negotiating two warring but immediate needs: the need to launch the bank as soon as possible, and the needs of these internal customers to satisfactorily do their jobs.

In modern businesses, customer experience should be exemplary regardless of the type of customer or interface.

In an ideal scenario therefore, there shouldn't be any meaningful difference in quality of customer experience for internal or external customers. While some of the SaaS providers you will have utilised in the solution architecture will likely provide an interface (at various levels of quality) this is unlikely to suffice. A typical customer journey, like making a payment or onboarding, will be split across multiple vendors, some with an interface, some without. Ultimately, you will need to create one from scratch.

To achieve this requires a bespoke modern flexible customer experience layer that helps internal customers do their jobs in the best possible way by providing them with an elegant custom solution accessible via a single sign-in. However, in the interests of a timely market launch those needs might need to be negotiated or sometimes even sacrificed.

There are three main approaches that can be adopted to different degrees in various combinations to provide a more lean approach as a solution:

1. Smart sheets (e.g. Google docs, Notion or Coda) can go a long way. These can then be accessed via API to provide data into the process
2. Workflow automation software can link the various points in the process and third party APIs and provide the back office employees with an interface to easily configure and reconfigure as needed without engineering intervention
3. Creating some quick bespoke components utilising available UI frameworks is also a way to provide an interface with a few buttons as inputs and outputs for a quick and dirty online touchpoint

The key to success and the main risk are the same here: the levels of communication, flexibility and trust between the team providing the solution and team using it. They often speak very different languages.

Treasury

The treasury provides the finance teams with visibility on various financial data points that are combined to form a picture of the bank's financial position, both internally and as a report to the regulator.

There are actually a few off-the-shelf SaaS solutions on the market that provide dedicated treasury solutions, but these have largely been built around serving the requirements of large-scale incumbent FIs rather than the needs of new players.

With the right expertise and familiarity of the toolset in your organisation, you can build bespoke dashboards in the following way. The requisite data points can be exposed by APIs provided by the core systems and storage databases in the solution architecture. These can be consumed by third party insights software like Graphana or Tableau, which can be used to provide bespoke dashboards.

CRM

Customer Relationship Management is the component that provides the necessary tooling required for the management of the information around a customer's life cycle and in particular how it pertains to sales and the deal flow. Regardless of the primary touchpoints your customers interface with for your bank, you will likely need a CRM to manage prospect data, deal flow, and then post-sale customer lifecycle stages such as renewal, upsell or referral.

The key is to resist the temptation to let the CRM do anything other than being a means to manage customer relationships.

CRM companies like Salesforce and their equivalents have been using the basic need for any company with customers to have a CRM as the entry point into an organisation. They then try to cross-sell customisations of the CRM that aim to cover implementation of internal and external customer journeys.

This can even work in certain use cases—if the customisation aligns closely with your particular proposition for example—but as it ties you to a weighty piece of legacy from day one, it is a very naive way to build a system from the ground up.

The core aspects of a CRM solution are twofold:

- the marketing/sales funnel
- the ability to store and retrieve information about individual customers and your organisation's relationship to them

The CRM should never be the central database for the customer information:

- creates unnecessary constraints imposed by the CRM's view on how the customer data is structured
- it's impossible to scale
- you're locked into a vendor from day one

The CRM is instead the ideal place for the sales team to keep the data that pertains to their role. With the CRM's API, the main customer data can be periodically augmented.

If the sales team is only composed of a handful of sales people, smart sheets are a great way to bootstrap the CRM requirement, stay flexible and keep costs down. Your CRM will grow in complexity in direct relation to the size of your sales team and the weighting of your customer acquisition towards sales versus marketing.

For example, the more sales-oriented your organisation is, the more tooling the various sales team members will require to do their job. Depending on the different products your bank will offer, you may wish to add individual products—for example, credit cards—to deals that relate to some customers but not all. If on the other hand, most of the customer acquisition is done via direct marketing and marketing automation—and not a dedicated sales person—this aspect could be potentially totally excluded.

If there is a sales team and a heavy emphasis on sales for customer acquisition, with a defined process and sales cycle to close customers and sell products to customers, there are myriad options on the market from modern dedicated sales tools with prospecting features, to more standard and familiar CRM solutions like Salesforce and hybrid sales/marketing automation tools like Hubspot.

Growth

Your marketing strategy should emerge from your initial learnings and conclusions drawn from your customer niche and your USP. That said, there is a basic set-up that any bank that is looking to grow using modern B2B or B2C strategies will need at a minimum. This starts with the way a prospect—or qualified lead—learns about your proposition and understands how they become a customer. Everything beyond that we will consider as part of the onboarding journey and should be considered as a key part of the customer facing journey and product.

Even if you don't plan for your marketing site to operate as a primary landing page(s)—for example if you plan to acquire B2B customers who consume your banking products through an API to resell to their own customers—a marketing site is still a worthwhile investment for numerous reasons. Outside of the soft power of having a strong brand, with the right analytics implementation, it can form the basis of advertising campaigns.

If you do plan to use your site as a landing page, it's recommended that it's driven by a headless CMS (content management system) with social and analytics tracking pixels (Google Analytics, Mixpanel, Hubspot, Facebook, LinkedIn etc) to allow the marketing team to run experiments, track prospects, and generate campaigns.

This should align closely with your data strategy and you should have analytics up and running.

The key is being able to provide the data from the various databases and systems in the solution to be able to attribute and describe the customer to understand where they are on the journey, what additional services they require, and which parts of the bank processes they interact with.

The important aspect is to track the customer through their lifecycle, to understand how they are interacting with your bank before and after they become a customer. This is where the retention aspect of Growth comes in, and ties in closely with Customer Success objectives. There are numerous ways to score customer satisfaction, and the likelihood of a customer to refer or upsell, thereby affecting the LTV:CAC ratio, as well as your product virality.

Your first goal is to have an automated process where you are able to provide the customer with the 'best possible next step' in a way that will feel frictionless

and effortless to them. The basis to development of that capability is rich customer data and installing the right automation tools to leverage that data. Effective growth starts with customer experience of using your bank.

Customer success

All aspects of customer support—any processes and engagement with the customer by customer service agents—should be automated. This includes messaging, the knowledge base, community and so on.

The goal here is to balance personalisation and self-service at scale. There are multiple platforms (e.g. Intercom, Zendesk) which have high quality tools and provide APIs to interact with the rest of the solution architecture if needed. They also provide additional internal customer-focused tools that can cover some of the journeys and requirements as discussed in the Back Office section such as Centralised Inbox and Case Management.

Reporting

When storing data, the primary concern should be that the storage is optimal for the structure of the data. In the case of relational data drawn from various database technologies, including third party data produced by various SaaS providers across the solution, or in the case of unstructured data, it should be stored in a data lake as we discussed in the Data Lake section. Reporting tooling should always be decoupled from the data that it represents.

The reporting tooling which provides the insights based on the data reporting should be integrated as a presentation layer only. There is a spectrum of options available, from completely pre-built insights and dashboards, to no code or pseudocode environments which are much more flexible. Your comfort in adopting the more flexible options will of course depend on the skill level of the stakeholders or the operational team responsible for creating the reports.

That said, it is important to note that in modern solution architecture there is no place for vertical or horizontal domain-specific tooling that also ingests and stores the data. This creates unnecessary duplication and in general is considered an outdated approach, mostly relevant to large-scale and/or less tech savvy organisations who have siloed organisational dynamics.



Who are Elsewhen?

Elsewhen have spent the past decade reimagining consulting for the digital age. We are a 40-person digital product consultancy established in 2011 and based in London. We combine strategy, design and engineering into one unified process.

We are independent and have none of the legacy technology, outdated culture, or procedural complexity that comes with being a thousand-person consultancy founded in the previous century.

We help companies tackle new market realities, pinpoint new business possibilities and surpass new customer expectations, and we lead the conversation—whether on technology, design, product or business—sharing our insights and best practices.

We work seamlessly with cutting edge organisations, or companies that aspire to be, to deliver best-in-class solutions into business-critical functions.

They understand that strategic thinking has to be deeply connected to execution experience; that achieving lasting change cannot come solely from the boardroom, but also from on-the-ground transformation.

Get in touch

Katy Heinemann
Growth, Elsewhen
katy@elsewhen.com

