



THE FRONTEND PLAYBOOK

Introducing an engineered approach to frontend development that empowers technology teams to deliver scalable, efficient, future-ready software.



NearForm deliver high-performing, transformational digital solutions for global companies of all sizes. Leveraging modern processes and tools alongside earned expertise in open software and development, they bring products and services to life. From product prototyping and design to application and platform development, NearForm are evolving the way businesses operate, communicate and perform. For more information, [visit nearform.com](https://nearform.com).

CONTENTS

The Frontend Playbook is our thinking on how to avoid frontend sprawl by introducing a strategic frontend architecture and approach, which we call ‘frontend engineering’.

1.	THE IMPORTANCE OF THE FRONTEND	
	• Understanding the frontend	5
2.	WHY THE FRONTEND IS BROKEN	
	• Identifying frontend sprawl	6
3.	THE CURRENT APPROACH	
	• Persistent drawbacks to the current frontend structure	8
4.	ENGINEERING THE FRONTEND	
	• Recognising the issues and planning a path forward	11
5.	A CROSS-PLATFORM APPROACH	
	• Key considerations and tools	12
	• Polaris accelerator	15
6.	THE FUTURE FRONTEND STATE	
	• Making it work	16
	• Partnering for success	17

Equipping organisations for a fluid digital reality means more than just streamlining technology. To get the frontend under control, companies must also examine their structures and processes.

The importance of the frontend

ACCORDING TO THE [2021 GARTNER CIO AGENDA SURVEY](#), 90% OF THE TOP-PERFORMING enterprises are pursuing digital channels, and almost 75% are introducing digital products faster. The need to do things online because of the Covid-19 pandemic has merely accelerated the blistering pace of the global economy's shift to digital.

With [direct spending on the technologies and services that facilitate digital transformation](#) forecast to reach a staggering \$6.8 trillion between 2020 and 2023, the most farsighted companies are looking beyond the necessity of the pandemic to find potential for future success. [Forrester predicts](#) enterprises that take advantage of these new opportunities will grow at a rate 3.5 times faster than their rivals.

Organisations everywhere are scrambling to be the ones users turn to for web and mobile offerings. They know that they can only be competitive if they refine existing products and deliver new offerings that meet the market need as quickly as possible. Technology capability is the key factor that determines how successfully organisations give the market what it wants — and time is not on their side. A natural response to the challenge is to keep investing in disparate web and mobile tools to keep pace with technological advances, but this brings its own problems.

The frontend covers any code that presents data to the user, principally websites and apps. Our approach is based on first-hand experience empowering tech teams to build apps, websites and systems that work well together and deliver a seamless frontend.

WHY THE FRONTEND IS BROKEN

Many enterprise frontend systems have evolved over the years into a jumble of legacy technologies built on older architectures.

Functionalities are often duplicated, systems don't communicate with each other, and the resulting unwieldy structures are not fertile grounds for innovation and digitisation.

Add to this clumsy mishmash of legacy systems a forced transition to digital brought on by Covid and you have a plethora of businesses scrambling to install or adapt new technologies under intense pressure. Everyone is trying to improve their digital experience, but the myriad of technologies is not always well integrated, architected or maintained.

This rapid growth and emergence of multiple technologies has put many organisations on the back foot. Instead of adopting a strategic approach, they have embraced random unconnected technologies in an effort to get their product out quickly. Hence the frontend sprawl.

The COVID-19 crisis has uncovered exciting opportunities for truly agile companies, but for others it has simply highlighted the imperative to reconfigure the frontend. To the extent that they do so, greater productivity will follow. Streamlined technology stacks will enable greater agility, clarify data insights and encourage a frictionless, customer-focused experience.

75% of top-performing enterprises are accelerating their digital products

\$6.8T estimated direct spending in support of digital transformation by 2023

3.5X faster growth predicted for companies taking advantage of digital technologies

Source: 2021 Gartner CIO Agenda Survey, Forrester Predictions 2021

THE CURRENT APPROACH TO FIXING THE FRONTEND IS NOT WORKING

THROWING RESOURCES AT THE PROBLEM, MANY COMPANIES EMPLOY MULTIPLE teams to develop, test and maintain different versions of the same app. This approach of allocating dedicated resources to each platform allows organisations to optimise app fidelity and make maximum use of the target platform because developers don't have to compromise for the sake of multiplatform functionality.

Maintaining dedicated teams for each platform may sound like a good solution for managing the frontend, but it is no magic bullet. Here are some of its drawbacks:

High cost of ownership

The biggest disadvantage of maintaining individual teams for each platform is the obvious cost of employing so many resources. The reality of [technology fragmentation](#) means that companies must support users of different operating systems (OSs), web browsers and devices — not to mention the different versions of mobile OS that a customer might be using (especially in Android, where users depend on [OS versions that are up to three years old](#)).

Silos emerge, with teams aligned to technologies rather than services. Running siloed applications on multiple platforms is expensive: Each new service or feature must go through separate design, build and test cycles for each platform and within each platform.

New launches on multiple platforms are expensive, but the initial project is a mere 10% of the life cycle for most services/products. Continuous maintenance for new features and fixes compounds the high cost of multiple design/build/test cycles for each code base, for every app, every year.

Lack of agility

Once you commit to maintaining multiple teams for each platform, you face the complexity of having to work across multiple siloed applications. This makes you less agile and delays the delivery of new features to market by up to 200%. Factors that contribute to delays include:

- Each new service or feature needs to go through a separate design, build and test cycle for each platform and for different versions within each platform.
- Maintaining feature parity and ensuring comparable user experiences across different technologies requires additional effort in design, development and testing.
- Coordinating separate teams demands extra planning and effort.
- Current application approaches for iOS/Android/Web mean separate code is required for each platform, and this does not facilitate speedy roll-out of features on a component or 'micro-frontend' basis.

Team misalignment

When teams are focused on a specific platform, they become aligned with particular technologies — not your business objectives. This means they require closer management as they follow disparate tracks in their efforts to keep each platform up to date.

If you are maintaining dedicated teams for each platform, each team will be working with its own programming language, frameworks, tools and legacy — meaning each platform will incur its own technical debt (the code you need to write today to compensate for the shortcuts you took yesterday). This makes it virtually impossible for teams to ship features at the same time.

Harmonising their work to ensure feature parity across platforms becomes more and more complex as the app matures and more functionality is added. Given the idiosyncrasies of each platform, teams are seldom at the same level. You need more people with different skill sets to work on multiple versions on different platforms, but the talent you need to form a change culture is unlikely to be attracted to a company that operates siloed apps on older technologies.

Frontend sprawl is far more than just a technological issue:

By attempting to manage a multiplicity of platforms by allocating more resources to the problem, organisations frequently end up with a monolithic structure, which is slow to change and expensive to run.

**Ultimately,
frontend sprawl
is detrimental
for productivity,
competitiveness
and organisational
culture.**

ENGINEERING AN EFFICIENT FRONTEND

As with any issue, the first step to solving frontend sprawl is to acknowledge that you have a problem in the first place.

Once you become aware of how much of a millstone frontend sprawl has become for your organisation, you can take steps to eliminate it and free technical teams to work on achieving milestones instead.

We recommend moving to a cross-platform approach to the frontend. Modern technology — specifically React Native — puts a full cross-platform (iOS/Android/web) solution within reach for most organisations. This approach resolves several issues relating to the frontend estate, including:

- Reducing TCO
- Increasing productivity
- Reducing time to market

It also improves organisational health by:

- Aligning teams with business KPIs
- Attracting the best talent with modern tech
- Reducing management overhead

MOVING TO A CROSS-PLATFORM APPROACH

TRANSITIONING TO A CROSS-PLATFORM APPROACH TO THE FRONTEND INVOLVES more than simply adopting a new technology. To maximise your potential for success, you need to have key components in place. These include selecting a modern technology that meets your organisation's needs and formulating a roadmap of where you want to go with it.

Factors you need to consider when choosing your technology should tell you whether your choice will deliver on overarching business and organisational objectives over time.

Ask yourself:

- 01 Is the technology sufficiently popular that the talent needed to implement and maintain it is available?
- 02 What do the tools and ecosystem look like?
- 03 Is the performance adequate?
- 04 What about native app features?
- 05 Is the technology component-based?

The framework

NearForm recommends the [React Native](#) framework as a logical choice when moving to a cross-platform approach. Originally developed by Facebook, React Native is used to create technologies that deliver outstanding user experiences and is also the most popular choice of developers worldwide. The framework has a committed development community, which gives us confidence that it will be supported for years to come.

React Native leverages the native resources of the mobile device and allows developers to create full, native mobile apps for both iOS and Android using JavaScript. Developers can reuse the same code across each OS, so you don't need separate teams to build and maintain apps for different platforms. Any developer with [knowledge of React](#) can use their current skillset to cater for the web, iOS and Android using React Native and React Native for Web.

Because developers don't need to be aligned with their target platform, React Native allows teams to iterate faster, reuse code across platforms and share more knowledge and resources.

Using this open source, cross-platform framework to build applications costs up to 70% less than the traditional multi-platform technology approach.

The toolkit

Once you have the appropriate framework in place, you need to ensure that you have the right tools and processes to enable on-boarding, learning and productivity for staff. The proper tools optimise your potential for successfully adopting your new approach to the frontend.

This is something we know from experience at NearForm. Having completed hundreds of projects, we have curated and refined a library of smart tools that developers love. We have selected the most effective open source tools, based on:

- Feedback from developers
- Available community support
- Suitability for enterprises

React Native and React Native Web are the core open source tools that enable us to develop cross-platform applications. We also use an open source toolchain called Expo, which makes a host of features — including over-the-air updates, certificate management, push notifications and on-device testing much easier when developing in React Native. On top of this platform, we use a range of open source libraries, depending on the application’s requirements.

We have perfected this blend of tools and processes for consistent best practice from project concept through to deployment and ongoing maintenance. The approach can then be adapted to suit the frontend needs of any organisation — whether the adoption is bottom-up or top-down.

Comparing key features and benefits of React Native, Xamarin and Flutter

	REACT NATIVE	XAMARIN	FLUTTER
Developer availability	<div></div>	<div></div>	<div>Dart language not widely used</div>
Access to native functions	<div></div>	<div></div>	<div></div>
Native UI features	<div></div>	<div></div>	<div></div>
Development ecosystem	<div></div>	<div></div>	<div></div>
Community support	<div>large and growing rapidly</div>	<div>around a long time</div>	<div>still relatively young</div>
Vendor support	<div></div>	<div></div>	<div>Google history</div>
Performance	<div></div>	<div></div>	<div></div>
Code reuse	<div></div>	<div></div>	<div></div>
Pricing	<div>open source</div>	<div>licensed</div>	<div>open source</div>
App bundle size	<div></div>	<div></div>	<div></div>

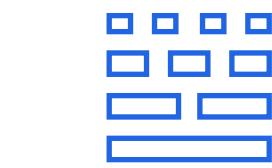


Start fast, build smart, deliver more

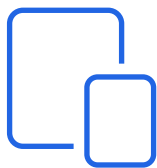
Nearform has mined its expertise in the area of architecting a new approach to the frontend to develop, iterate and test a blend of proven open source tools, technologies, best practices and opinionated workflows. We call the result [Polaris](#).

Polaris enables fast delivery to mobile and web platforms by a single team from a single codebase. It uses the best open source tools and fine tuned workflows to support collaborative development and secure, tested deployment.

Implementing multiplatform solutions involves far more than code. The decisions made on frameworks, tools and processes also impact team productivity, time-to-market and total cost.



LEADING EDGE MOBILE TECHNOLOGY STACK



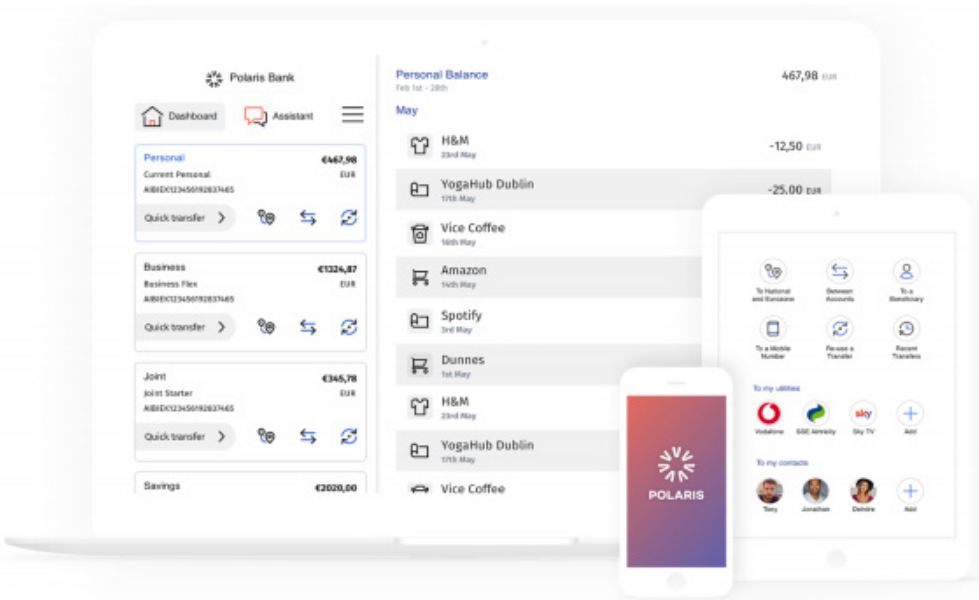
SINGLE CODEBASE FOR WEB, iOS, ANDROID



SECURITY AND RELIABILITY OUT OF THE BOX

1 codebase
1 team
3 platforms

Learn more at [nearform.com/services/accelerators/polaris](#)



THE FUTURE STATE OF THE FRONTEND

IF RACING TO MEET THE DEMANDS OF A DIGITALLY HUNGRY MARKETPLACE HAS turned your frontend into a snarl of interdependent, uncontrolled technologies, you need to shift your approach to one that prioritises the rapid rollout of products and platforms that deliver continuous change. This does not mean you need to rip out your current structure and processes and replace them with an entirely new infrastructure and set of technologies and procedures.

A tailored approach to reforming the frontend means that the approach adopted can be customised to suit your organisation's current position and requirements.

Making cross-platform work

To leverage the benefits of the cross-platform approach to the frontend, you need to understand the current state of your organisation and the context for reforming your frontend. This involves distilling the essence of your mission and the problem you are trying to solve.

Instead of getting distracted by pet projects, focus on what gets the business moving, what speeds up time to market and what will embed the idea of thinking in terms of a journey across your organisation. If you can crystallise those ideas into a message that fits onto a single page and everybody can grasp, you have something very powerful that you can start to build on.

Your mission should drive KPIs and scorecards that everybody is measured by. It should be reinforced constantly from the CEO level down. This sort of company-wide buy-in is pivotal to an effective frontend transformation.

As you embrace your mission, you should prioritise early, frequent success. Adopting a new approach to the frontend can be done incrementally, starting with small, relatively easy wins, so that the organisation can see positive results quickly before plunging into an organisation-wide transformation.

The spectrum of options includes taking a small, greenfield project as a proof of concept to serve as a low-impact, low-risk introduction to frontend reform. You could also migrate gradually to a new multiplatform frontend system by inserting React, for example, or opt for service slice replacement. Whichever route you choose, the level of risk involved is comparatively low for the potential rewards generated.

The success of your transition relies on ensuring that your people are fully supported in learning how to implement and maintain the new multiplatform frontend. Enlist expertise to build the capability your team needs to leverage its benefits, driving technological and operational improvements and encouraging a culture of collaboration.

Partnering for success

NearForm leverages strong expertise in web engineering, particularly in React Native, to tailor a modern frontend architecture to the needs of each organisation.

Our Polaris set of tools and processes makes this single codebase approach work, and we can customise an approach to help you achieve a modern frontend, including discovery workshops, proof of concept or a lighthouse project, migration from existing technology and full bottom-up transformation.

By adopting a measured approach to tackling your frontend, you will see the benefits without assuming excessive risk.



GET IN TOUCH

Tankfield, Convent Hill, Tramore, Co. Waterford X91 PV08
+353 51 330 290 | +1 916 235 6459 | nearform.com

 **NearForm**