A Complete Guide to

Legacy Software Modernisation

See how digital transformation can skyrocket your business. Check our success stories, numbers and facts

miquido

How much does outdated software cost you?

Did you know that an average CIO spends between 60 and 80 percent of their time on everyday IT management, due to outdated software systems?

You want a vintage car, not vintage software. Let's take that metaphor a little further. Every car producer makes sure to redesign or enhance their available models regularly in order to maintain the target sales rates. It is impossible to conquer the 2020 market with 2010 technology, and software doesn't differ much.

If you wish to stay on the competitive edge, you need to switch your focus from day-to-day tasks to strategic planning.

$140BN

is wasted annually in the UK and US because of data trapped in legacy systems

broad-group.com

44% of CIOs

name legacy systems as main barrier on the way to digital transformation

us.logicalis.com

44% of decision makers

in companies with 1000+ employees say legacy affects every or most of their projects

computerweekly.com

60-80% of companies' IT budget

on average is spent on maintaining old software systems

itproportal.com

How do I know that it's time to freshen up my software?

Here are 7 signs that indicate your current solution requires modernisation.

- **You're spending a lot on maintenance**

Operating on outdated software gets more expensive every year. Higher costs depend on, among other things, maintaining additional hardware and staff training, not to mention lost business opportunities. To paint a picture with numbers, Microsoft has calculated that the business costs of operating a computer running on Windows XP are $780 per year, while for Windows 10 the cost is only $168.

On top of that, it's getting more difficult to find dated technology experts. According to a recent survey conducted by the National Association of Manufacturers (NAM), the economy is going to face a shortfall of almost eight million workers by 2030, and potential revenue losses are expected to be as high as $600 billion. Because of that, you soon might find yourself trapped in a vicious cycle. While your consumers are getting frustrated with using your outdated software, your employees who know how to operate it are retiring in record-breaking numbers.

- **Your application is not secure anymore**

Outdated software poses a serious threat to cybersecurity. Older security measures might no longer be sufficient for modern requirements. As the software doesn't receive security patches anymore, it's more prone to outside breaches or malware. The issue is serious especially for businesses processing a lot of customer data, such as online stores, financial institutions or medical facilities.

One of the most notable examples of recent years is Adobe Flash Player, a browser plug-in used widely for viewing multimedia content, such as videos or games. However, due to serious cybersecurity concerns discovered in Flash in 2019, the plug-in is very soon to be shut down, and many companies (including giants like IKEA) are now frantically rewriting their existing solutions based on Flash.

- **Your app's tech stack remembers the stone age**

If your application was created with an outdated or obsolete technology, modernisation is a must. A lack of appropriate skills on the market will make it difficult to find a vendor willing to work on maintaining and updating your old system. Moreover, it's probably not going to be compatible with newer software versions and newer devices, which, over time, is going to cost you the loss of potential customers using them.

- **You keep missing new tech opportunities**

Year to year, devices and browsers get new technology opportunities worth adapting to attract new customers. Biometrics, dual cameras or native mobile payments are standards nowadays when it comes to mobile devices, and yet they only appeared about 2 or 3 years ago. When it comes to browsers, we start to observe trends such as push notifications or offline modes. Continuing to use your legacy software puts you behind your competitors, as it mostly doesn't support the latest technology updates.

- **Your software is frustrating to use**

Think about all the time you are wasting because of ineffective software. At first

glance, it doesn't seem like a big deal to wait a couple of minutes for a page or report to load. But once you multiply that by the number of your employees, and then by the number of working days in a year... Paints quite a picture, doesn't it?

- **Your app rules your business and not the other way around**

Your business is constantly growing and your needs change. Your software should be able to keep up with that. Leave no space for sentiment - you shouldn't have to compromise on business opportunities or revenue because of outdated software. Moreover, adding new features or enhancements to outdated applications in order to stay on the competitive edge might turn out to be excessively expensive and time consuming, or even impossible.

- **Your app's interface has an ancient feel**

A picture paints a thousand words, as people buy with their eyes. If your business model includes offering software to your customers, or supporting your services with a mobile app, you need to remember that legacy software ages visually, and therefore loses marketing potential. Even if your app looked super fresh in 2015, as time passed, in 2020 its user interface has probably already become pretty old and puts your potential customers off.

See companies that trusted us with software refactoring

HERBALIFE NUTRITION

PLAY

BNP PARIBAS

klassik radio

Hello FRESH

Which?

TUI

What is there to gain?

Cost reduction

Save money on operational and maintenance expenses

60% to 80% of companies' IT budgets are spent on "keeping the lights on"

itproportal.com

Enhanced agility

Grow by catching new business opportunities

44% of decision makers in companies with 1000+ employees say legacy affects every or most of their projects

computerweekly.com

Greater team productivity

Access teamwork tools and share through the cloud

Over 50% of UK employees are unhappy with their workplace technology

theaccessgroup.com

Access to new solutions

Integrate your system with new business technologies

Legacy systems are one of the biggest barriers stopping businesses from digital transformation

snaplogic.com

Secure work

Make sure your business data and customer data are protected

About 87% of IT decision makers claim that legacy systems leave businesses vulnerable

realwire.com

Do I need to pause my business operations to start refactoring?

You actually can have your cake and eat it, too. We created our refactoring approach to ensure your business runs undisturbed.

Making sure that business runs smoothly and continues to bring revenue is one of the most common concerns for organizations that consider software refactoring. Fear of threatening key business operations prevents decision makers from seeing the bigger picture, and gives them the illusion that maintaining old systems equals staying on the safe side. That is especially true when it comes to existing, custom-made solutions: rewriting the entire thing from scratch might seem intimidating. All of these factors work together to create a temptation to put the modernisation off.

Our core value is to offer you a solution that won’t force you to compromise on your revenue

You won’t need to pause all of your business operations to update your software. Instead of jumping head first into modernisation big-bang style, we opt for the iterative approach: we will rebuild your system in separate modules, one by one, over time. Thanks to that, software features vital to keeping your business up and running will stay in place. We will see to it.

How is that possible?

Modernising your software is supposed to serve you, and not the other way around. That's why we prioritise your business goals over the urge to start rewriting the entire app from scratch, which is a common strategy on the market.

We believe that it's best to develop and monetise your existing solution as long as it is possible. With that in mind, we identify the core software features you need on a daily basis and prioritise refactoring tasks accordingly, setting a long-term strategy.

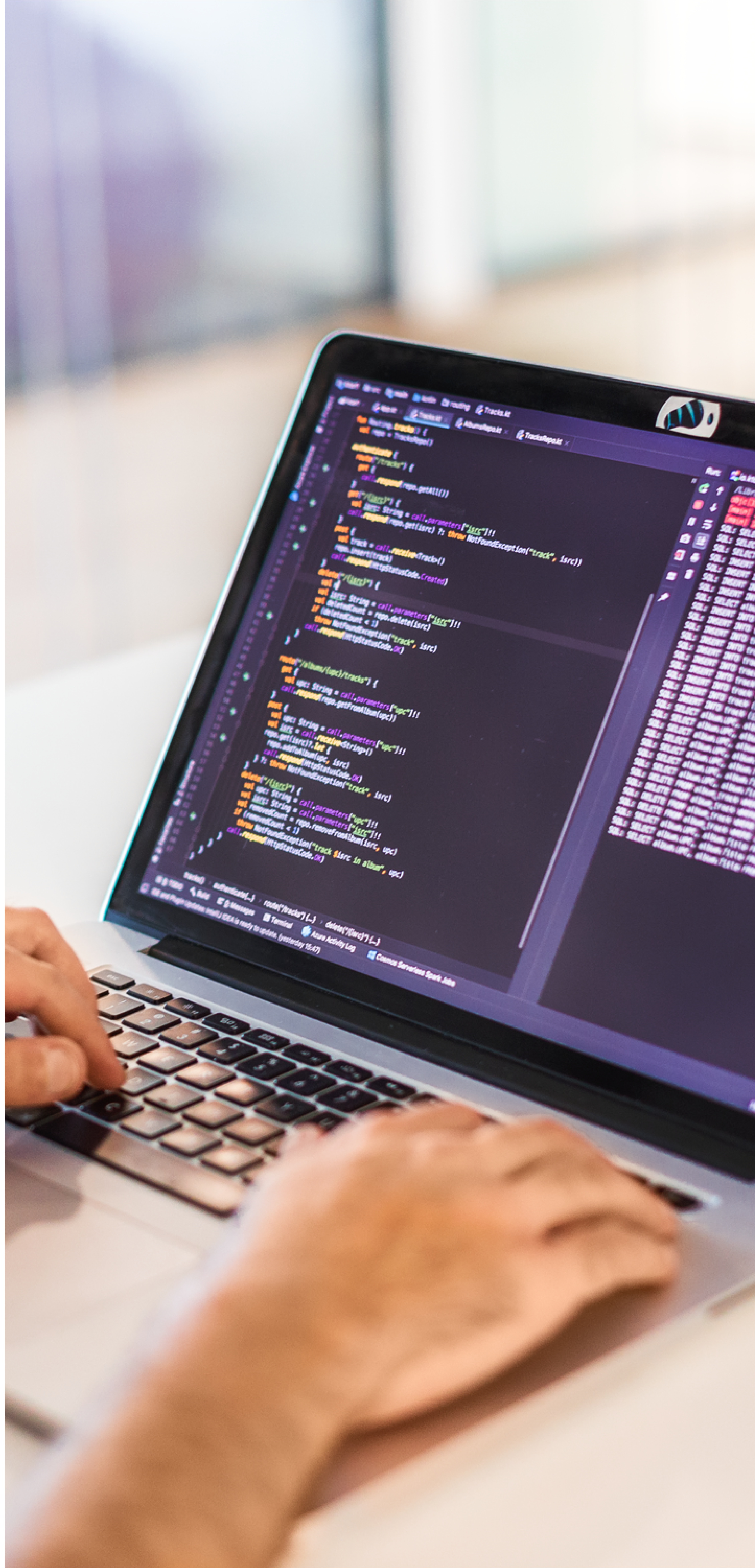
That way, two strategic timelines are created

One represents the maintenance and development of your current solution, and the other shows the refactoring process.

In most cases, we suggest splitting the development team's powers in 80-20 proportions. The majority of the team's time is spent on maintaining the existing business functionalities, while 20% of the time is dedicated to rewriting and refactoring your app – **one module at a time.**

Imagine your current application as a large tree, where the trunk symbolises the core features of your app, and each branch is a separate, less important functionality. Instead of cutting the entire tree down and planting a brand new one, we start the refactoring at the branches. From there, we slowly work our way to the trunk, where the most important features are held.

That way, our software modernisation process is **agile, based on refactoring small modules of your app one by one.** An opposite approach is called **the waterfall model** or **the big bang method.**



Thinking about giving your old system a boost?

See how we do it!

Hello Fresh – how we did it?

What was the story?

At first, Hello Fresh, the number one meal kit company in the world, reached out to Miquido with a plan to expand their existing website. That initial goal soon turned into a complex modernisation project.

What was the challenge?

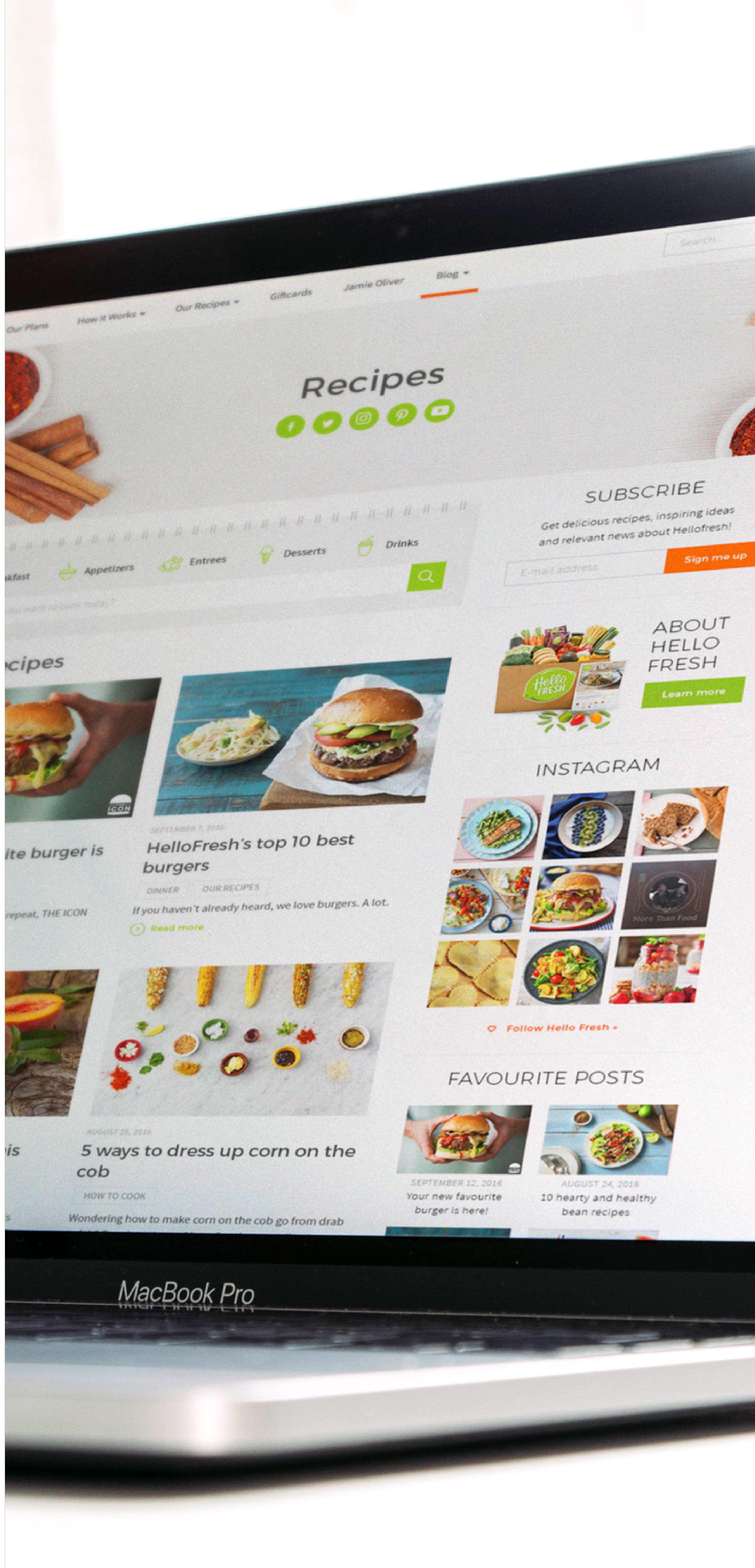
The web application that the Hello Fresh team used was initially created on Rocket Internet, a dated ecommerce platform. The app was a monolith written in PHP language, packed with dozens of microservices. The solution was based on outdated, complicated architecture and no longer supported the company's business needs and blocked the way to introducing innovation.

Results?

- **1M app users daily**
- **4.5 rating on App Store**
- **€2.6MM market value**
- **1M+ app downloads**

How did we do it?

Together with our team, Hello Fresh decided to recreate the app from scratch and write it in the Golang language instead, a much more modern and competitive programming language. The early phase of the process focused on splitting the existing app into separate modules, and then starting with rewriting the most important ones.


Recipes
SUBSCRIBE
Get delicious recipes, inspiring ideas and relevant news about Hellofresh!
Sign me up
Appetizers
Entrees
Desserts
Drinks
ABOUT HELLO FRESH
Learn more
INSTAGRAM
HelloFresh's top 10 best burgers
If you haven't already heard, we love burgers. A lot.
Read more
Follow Hello Fresh »
FAVOURITE POSTS
5 ways to dress up corn on the cob
Wondering how to make corn on the cob go from drab
Your new favourite burger is here!
10 hearty and healthy bean recipes
MacBook Pro

Need more specifics?

1. Project documentation review
2. System and code walkthrough
3. Code review
4. Transition planning
5. Checking for obstacles
6. Verifying the credentials
7. Checking the integration system
8. Infrastructure audit and cost assessment

See our step-by-step guide to a successful project transfer.

1. Project documentation review

First and foremost, our team goes through the project documentation - if there is any. This way we can find out about the goal of the project, see what the timeline looked like, review what was and was not done.

2. System and code walkthrough

After that, we conduct a system and code walkthrough with the previous team. Thanks to that, we can gain a general knowledge about the technology stack that was used, and learn about things such as 3rd party components, system architecture or deployment flow.

3. Code review

When it's done, we perform a code review of the current source code. Our development team verifies the current architecture and code quality, which allows us to decide if the application requires refactoring. Thanks to that, we can estimate the potential cost of implementing new functionalities.

4. Transition planning

An important part of our code review focuses on identifying the core business features we need to maintain at all times in order to keep your business operations running. We always take our time to plan carefully and determine what needs to be done first and what can wait. Thanks to that, we are able to prioritise our work, all the while letting you maintain business continuity.

5. Thecking for obstacles

After that, we check for obstacles that might stop us on the way to refactoring the app. We search for any private external dependencies that could belong to the previous developer. We also check if the current production version of the app is in line with the current source code version in the repository.

5. Verifying the credentials

The next step is to get and verify all required credentials, such as access to developer consoles, databases or servers. When it's done, we also make sure to get the credentials to all of the 3rd party components that are integrated with the client's system, such as payment providers or tracking systems.

5. Checking the integration system

It's also important to check if continuous integration is implemented and if it works correctly. If it's hosted on the previous developer's infrastructure, we will need a plan for moving it. We also check if the app is integrated with a tracking and log collecting system. If it's not, then such a solution needs to be implemented in the future.

5. Infrastructure audit and cost assessment

Finally, we conduct an infrastructure audit. The audit helps us assess the system performance, scalability, reliability and general cost of maintenance. This way, we give you the answer to the most important question: if the current solution could still be used without extra cost. If it can't, we determine whether it can continue to be used with some extra cost, or if it needs to be redesigned completely, from scratch.

TUI – how we did it?

What was the story?

TUI Poland reached out to us with a need for a faster, more reliable, and client-oriented mobile application. The original app was developed by a different company. Miquido took over the legacy code in Java and ObjectiveC, together with messy processes and complicated communication patterns. The initial app rating was only 3 out of 5 stars.

What was the challenge?

The path to success was bumpy. We focused on rewriting the legacy code: changing the structure and architecture, cleaning up and preparing the app for new features. We also improved the processes and communication patterns, creating an atmosphere of transparency and involvement.

The biggest challenge, however, was that TUI's application dealt with thousands of customer operations daily at that time. Deciding to go with the waterfall approach to modernisation would equal gigantic losses for the company, making the refactoring process unprofitable. At the same time, the app was developing so quickly that there was hardly any time for TUI to plan a successful refactor.

Results?

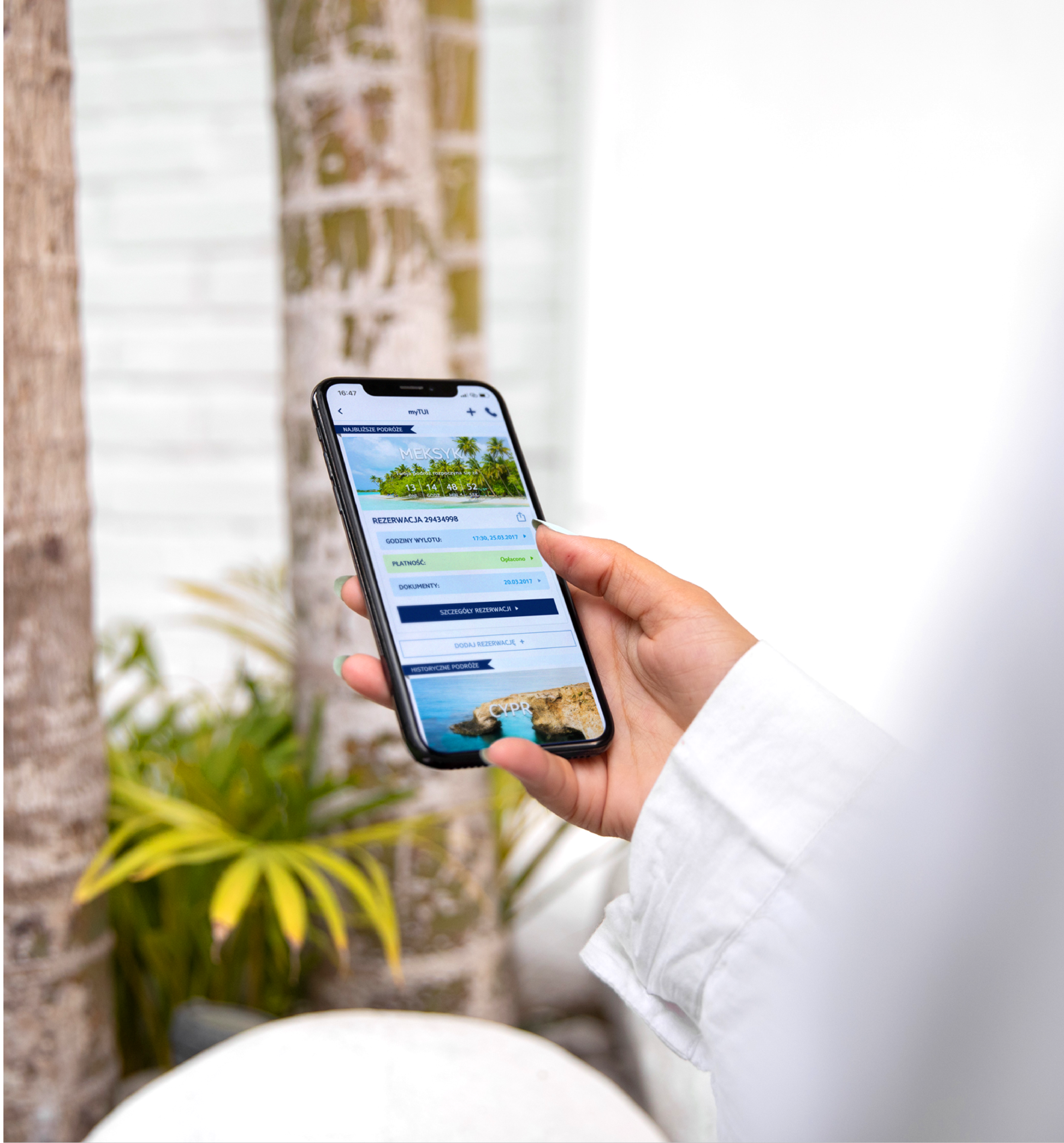
- **4.5 rating on App Store and Google Play**
- **500k+ app downloads**
- **2016 Mobile Trends Awards finalist**

How did we do it?

It took a team of 4 developers and almost a year to rewrite the legacy code. Since then, we have been implementing new features and rewriting the app into more up-to-date frameworks of Swift and Kotlin to keep up with the newest technology. After running a profound UX/UI audit, we also redesigned the app's interface to become more user-friendly.

Improved communication was the key to success in this case. We made time to research the client's needs and plan long-term accordingly. We worked our way to transparency and inclusion through regular calls and meetings, workshops, hackathons, and integration events.

TUI is now the leader of travel applications in Poland. Refactoring their mobile app pushed TUI to decide to focus more on their digital customer service channels. Thanks to that, the company was ready for the pandemic and were able to provide their customers with effective online support.



Nowe ubezpieczeni
Planujesz podróż?
Zadbaj o siebie, swoich bliskich, bagaż oraz sprzęt sportowy. Kup ubezpieczenie podróżne GOtravel w 3 prostych krokach.
KUP TERAZ

BNP Paribas – how we did it?

What was the story?

BNP Paribas started out with an out-of-the-box mobile application created by a third-party vendor. The main goal of the project was to modernise the current software, written in Objective C and Java, no longer competitive and properly supported. BNP also needed new app features that would help the company stay ahead of the curve.

Results?

- 500k+ downloads on Google Play
- 4.5 rating on the App Store
- 400k+ active users

How did we do it?

We decided to rewrite the app using more popular and better-supported programming languages. We switched from ObjectiveC and Java to Swift for iOS and Kotlin for Android. Our team changed the overall concept of the app structure, rebuilding the initial monolith construction into a set of microservices. Results? Improved performance and easier app maintenance.

Play – how we did it?

What was the story?

Play 24 is a mobile problem-solver, helping customers with their day-to-day account management. However, Play's goal was to elevate their app in a way that would set them as the technology leader of the telecom sector in Poland. Along the way, Play decided to extend the project by adding new features and functionalities to the app.

What was the challenge?

The original app was written in Java, a language popular at the time of the app's original release. At the time Play reached out to Miquido, the Java language was not only getting pushed aside and losing support, but the app itself experienced a lot of problems. As a result, the app's rating and popularity dove drastically, as users were getting frustrated with the app's ineffectiveness. The challenge was to continue developing the app and keep it running, all the while rewriting the existing, outdated modules one by one.

Results?

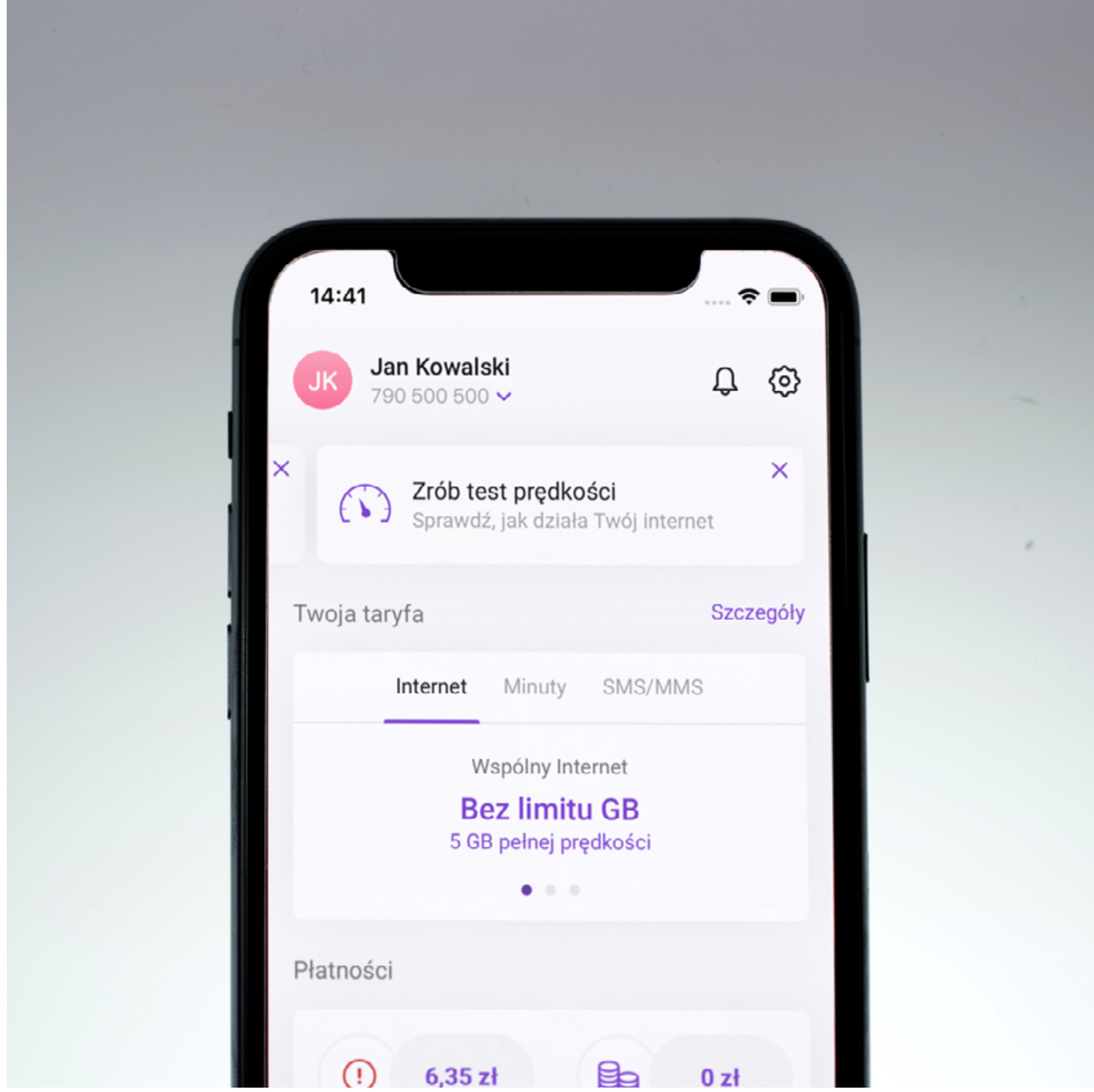
4M active app users

180k+ new users monthly

4.7 rating on Google Play

How did we do it?

Our team decided to rewrite the mobile app to Swift for iOS and Kotlin for Android, the up-and-coming languages at the time. As the app was already used by thousands, it needed to be continuously supported and the rewrite couldn't happen all at once. Because of that, we decided to separate the app into smaller modules and then rewrite each of them one by one.



As the client's goal was to introduce cutting edge solutions to their users, they also entrusted us with developing the app further and adding new features. Play 24 is now one of the most popular mobile apps available in Poland for iOS and Android, serving over 4 millions of active users.

Thinking about giving your old software a boost?

Contact our team to discuss the strategy for your legacy modernisation. We'll get back to you with an estimate in 48 hours.

Krzysztof Gudowski
HEAD OF SALES

krzysztof.gudowski@miquido.com
+48 517 453 797

miquido