

Choosing Mobile Developers:

How to Avoid Project Failures and Wasted Budgets



[The Core Essentials Checklist]



Prepared by
5Pro App Development

How to Choose Mobile App Developers?

Not-so-obvious but Critical Points Many Entrepreneurs Overlook When Picking a Contractor.

Now, before we dive right into the checklist, let's highlight a key point often missed by many facing the decision of which development studio to pick.



See, in most cases, the entrepreneur looking to create an app (whether for their own business or for the broader market) isn't a tech specialist.

They don't know how to judge a developer's quality. So, they think this makes choosing a developer hard.

But here's the kicker: even if that entrepreneur were a super-coder themselves, it wouldn't make much of a difference.

Typically, projects don't usually fail because of weak programmers (though sometimes they do, but that's not the main reason). We've often seen cases where the code looked good, but the whole project still failed.

Why is that?

Because a successful mobile app project isn't just about well-written code. It's about a well-structured business model.

An effective mobile app is essentially a strong business model, not just a set of cool features.

Grasping this changes the lens through which you view potential developers.

In a way, it's like choosing a business partner.

Sure, a development studio is just a contractor. But in this context, it's more apt to see them as a strategic tech partner, with whom you'll need to collaboratively build a revenue-generating business model.

After all, your goal isn't just to launch the app's first version.

You want to test hypotheses, develop it further, update it in response to evolving environments (like iOS, Android), etc.

At times, a major conceptual pivot might be necessary.



And here, the role of a tech partner is crucial - their tech suggestions should primarily be based on a deep understanding of your business.



Thus, it begs the question: if your partner excels technically but lacks an understanding of the business logic behind the project, would you still build a business with them?

So, in our checklist, the main points are about whether developers see their job from this viewpoint (from the entrepreneur's perspective) and if they have the skills for this approach.



Essentials Checklist



1

Do the developers understand the core of your business?

Do you feel you're talking to fellow entrepreneurs?

Are you on the same wavelength?

Or is it all tech jargon to you?

It's important to understand that a successful app is not just a collection of features. It's primarily an effective business model.

Developers who just focus on features but don't understand your business at all are on a sure path to project failure.

Along the way to that failure, you'll additionally spend a lot of money and nerves.

2

Do they have a successful track record with similar projects, or will they use your project as their guinea pig?

If the development studio lacks experience with projects similar to yours, creating each part of your application may become a slow and difficult process.

When developers have ready-made technological solutions, they don't waste time 'reinventing the wheel.'

Instead, they put their energy into shaping those solutions to fit your specific business needs perfectly.

3

Are they willing to break down the work into manageable stages, ensuring a clear, understandable result at each step?

If developers offer you a model where you can only "get hands-on" with something many months down the line, that's not a normal situation.

There's always an opportunity, at each of the stages, to show a visible result of the work that you could discuss with the developers and adjust if necessary.

For example, for certain tasks, a prototype can be made in Figma so that you can verify that it's exactly the functionality and UI you need.

This could be a demo, showcasing a specific working function of the application.

4

When the studio you choose begins a project, is an R&D stage included at the first phase?

This stage is critically necessary to avoid wasting lots of time and money on developing something that doesn't need to be developed.

Here's a simple example: A client came to us after another studio tried to develop a platform for them, spending nine man-years of work on just one feature.

After we discussed the client's business logic and understood the business model he was building, it was clear that there was no point in developing this module from scratch.

A smarter move was to take the best ready component available at the time and integrate it into the platform being developed.

This would take weeks, not nine man-years.

You can often reduce the development budget significantly by conducting quality R&D research and finding the optimal technological "puzzle pieces" for the client's business needs.

But for some reason, most studios ignore this stage, considering it optional.

5

Forecasting monthly costs during Scaling.

Unfortunately, most developers don't discuss with clients about the financial parameters within which the system should fit.

For instance,

What will the server costs be, and how do these costs correlate with the increase in user numbers?

How will the costs of using third-party services change depending on user growth? (Payment systems, Google Maps, SMS notification services like Twilio, etc.)

And so on...

These aren't minor details!

To illustrate, in a project we're currently working on, we had a thorough discussion with the client about the list of external services the app would use.

As an example, estimating the growth in cost of one such service - Google Maps - showed that even in the initial phase (as this project is intended for mass use with active map utilization), the monthly cost of this service could reach \$10-12k.

This didn't fit within the project's financial model. This influenced the decision regarding the app's architecture.

Instead of the initial plan to use a cross-platform framework (like Flutter or React Native) that would avoid developing separate apps for iOS and Android, we and the client decided to develop two native apps.

This is because maps for Apple's native apps are free. For Android, we found a much more cost-effective alternative. (In the cross-platform case, we wouldn't be able to use Apple maps)

Yes, developing two separate native apps will cost the client more than a single cross-platform app.

But in the long run, the savings on monthly expenses will outweigh such an increase in the development budget.

You'll agree that it's crucial to work these things out in advance before development starts, right?

So that you don't find out later that your app is ready, but it doesn't fit your business model's financial scope.

This again highlights how vital it is to view the future app as a business model (including financially), not just a set of features.

This brings us to the next crucial point:

6

Technologies are chosen after defining the Business Model and overall App Logic, not before.

If developers immediately suggest specific technologies right after hearing the idea, without first clarifying all aspects of the future app's business model, that's a red flag.

To choose the best technologies, you need to see the whole picture.

Moreover, it's essential not just to know all the app's functionalities but primarily to understand:

- The business objectives of the project.
- Whether the app will be integrated into another system. Maybe it needs to be integrated into an existing business, or perhaps a new business will be built around it.
- The financial or other constraints within which the project should operate (for example, its monthly maintenance costs)

and so on.

Do developers address Data Protection from the start?

If developers don't even broach the subject, it's a red flag.

Addressing data protection is not just about ticking off regulatory requirements, it's an integral part of the system's architecture and design.

Decisions about data protection can significantly influence the application's structure and the technologies employed.

Hence, it's paramount to factor in these considerations before diving into the actual development.

On the flip side, it's a positive sign if developers suggest conducting a penetration test on the prospective app.

Delving further into this topic, here are some crucial aspects of data protection they should discuss:

Compliance with Local Regulations: Beyond GDPR, are the developers well-versed in local data protection regulations specific to your target audience's region? Different countries might have distinct requirements.

Encryption: How do developers intend to encrypt sensitive data both in transit and at rest in the future app? Effective encryption is paramount in shielding user data from potential breaches.

User Consent: Will there be mechanisms to secure and monitor user consent, especially when gathering sensitive or personal information in the app? Ensuring this isn't just best practice - it's mandated by many regulations.



Alright, so those were the key points we reckon you should be keeping an eye on when choosing mobile developers.

Of course, there might be a ton more criteria, but we've tried to focus on the really crucial ones.

And to round off, we really want to emphasize once more that the process of mobile development and (possibly even more importantly) the ongoing evolution of the app is a long-haul game.

You can't just say at some point, "Okay, the app development is fully completed," and then not make any more changes to it.

The market changes, consumer preferences shift, operating systems evolve – yeah, everything changes. And it changes super fast.

So, let's reiterate a point we made at the beginning of this document: treat choosing developers like choosing a business partner.

Ask yourself first: am I comfortable with these guys? Do they understand me and my business? Do I get them when they're explaining the technical aspects of the project to me? Would I want to work with this team in the long run?

Hopefully, the points in our checklist will help you make that choice.

As a little bonus, we've decided to include here a small article of ours about approaching mobile app development as the creation of a business model.

(We've thrown around the term "business model" so much in our checklist, it'd be unfair not to delve into this topic a bit more :))

We hope this article sparks some new ideas for your own app!



How to Develop a Winning Mobile App

Why Do Most Apps Fail? Avoiding the Common Mistakes

Mobile apps can fail for many reasons.

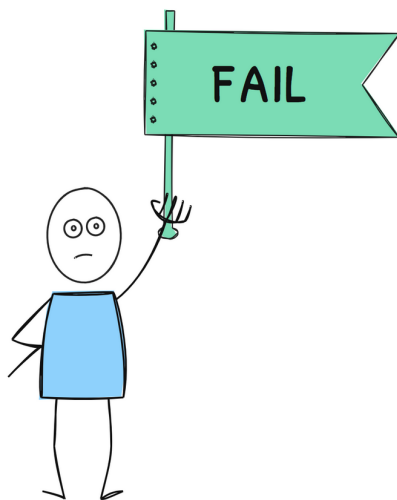
But there's one reason that stands out from the crowd - one that we see over and over again.

It's like the recurring theme of failure, the foundation that all other reasons pile on top of.

This sneaky little reason hides in the seemingly harmless thought process of average startup:

"I've got an idea! I'll just tell the guys at the development company (or my in-house tech team, if I've got one) and they will turn it into a working product."

Now, if you're building a mobile app with the mindset of "I'm making a mobile app," you're already on the road to ruin.



Hold up, you might be thinking. What's so bad about that?

How else am I supposed to do it?

Make my app like it's a chair? Or a bicycle?

Alright, let's clear up one simple, yet crucial point right here.

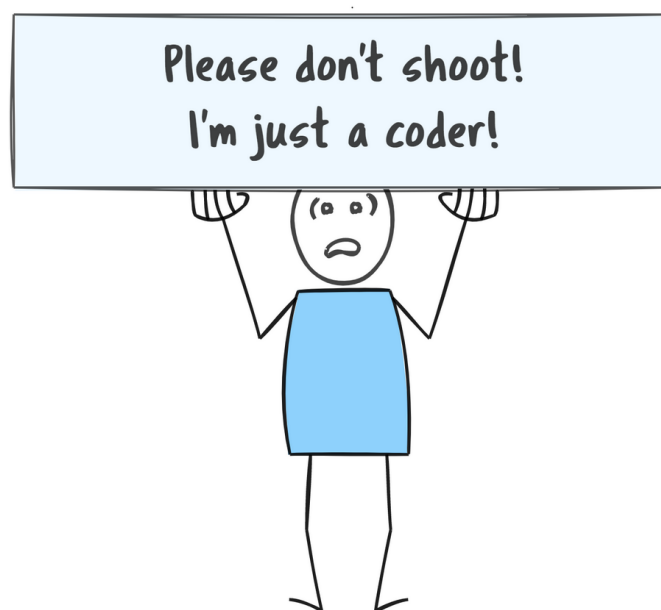
Sure, you can develop a mobile app, create an online store, or whip up some snazzy software - but these are just the technical bells and whistles of your project.

At the core, you're in the business-building game.

That means everything revolves around your business model, the foundation you're banking on (don't worry, we'll dive deeper into business models later on).

You've gotta get this: even the coolest, most genius programmers are **still just programmers**.

They ain't business-building wizards!



We often hear from entrepreneurs who need to develop a mobile application:

they stroll into a software company, lay out their project idea in detail, and then get assigned a manager who's supposed to analyze competitors, dissect the features of similar apps, and then help whip up a list of functions for their mobile app.

Seriously?





So, you've been assigned a 25-30-year-old manager who has only ever worked as an employee and has never built a single business,

and you're expecting them to help you analyze competitor businesses and advise you on how to kick their butts?

For real?

The kicker is, the problem with development studios (or your in-house tech crew) is that they're gonna crank out a mobile app for you, not a business.

But, c'mon, you don't want to end up with a slick mobile app that doesn't generate revenue, right?

Now, check this out!

If you corner even the sharpest mobile app guru and ask 'em about the core components of a mobile app, what are they gonna tell you?

Probably something like this:

The basic elements of a mobile application:

1. User Interface (UI):

- Screens: The visual components, like pages or views, that users interact with.
- Navigation: The system that allows users to move between different parts of the application.
- Controls and components: Buttons, text input fields, sliders, and other interactive elements that enable user interaction.

2. User Experience (UX):

- **Layout and design:** The arrangement of elements on the screen and the visual design that affects the overall look and feel of the application.
- **Responsiveness:** Ensuring the app performs well on different screen sizes, resolutions, and orientations.
- **Accessibility:** Designing the app to be usable by people with various disabilities, such as vision or hearing impairments.

3. Functionality and logic:

- **Application logic:** The code that determines how the app processes user inputs and performs tasks.
- **Data storage and retrieval:** Methods for storing and accessing data, such as local databases or remote servers.
- **APIs and web services:** Interfaces that allow the app to communicate with external services, like fetching data from a server or interacting with third-party services.

4. Platform-specific code and libraries:

- **Native libraries:** Platform-specific code and tools that enable access to device features and optimize performance.
- **Cross-platform frameworks:** Tools that allow developers to write code once and deploy it on multiple platforms (e.g., React Native, Flutter).

5. Security and authentication:

- **User authentication:** The process of verifying the identity of users, often through usernames and passwords or social media logins.
- **Data encryption:** Techniques for protecting sensitive data both in transit and at rest.
- **Secure coding practices:** Implementing best practices to prevent security vulnerabilities, such as input validation and proper error handling.

6. Testing and optimization:

- Unit testing: Writing and executing tests for individual components or functions to ensure they work as intended.
- Integration testing: Verifying that the different parts of the application work together correctly.
- Performance optimization: Identifying and resolving performance bottlenecks to ensure a smooth user experience.
- These are the key parts of a mobile application, and they collectively contribute to its overall functionality, user experience, and performance.

Was that answer correct?

Absolutely! Those are indeed the essential components of a mobile app.

But here's the twist. It wasn't the answer that was wrong, it's the question you're asking.

The right question is a whole different game:

[How can I build a business model for my mobile app biz that'll make it a success, rake in the profits, and keep on growing?](#)

As you might guess, it's not the programmers you should be asking this question. Instead, as the founder of the project, you should be asking yourself.



Mobile App



Business model



So if you approach creating a mobile application as a business, the key points that you need to clarify for yourself before coming to developers become clear.

Let's briefly go over the most important ones (if you're an entrepreneur, you should be well familiar with them, you just need to apply them in the context of app development).

1

Target audience

Of course, any business begins with answering the key question, "Who are we going to sell our product to?".

The same logic and questions apply to a mobile application:

- Which main user segments is our app designed for?
- What key pains and needs of the target audience will your app address?
- How is your target audience currently addressing these needs? Are there other mobile apps on the market that help them meet their needs?

And are you sure you can satisfy these needs better than the existing players? :)

If you are, then you can answer the next question:

2

What's your Unique Value Proposition?

Through what unique combination of your app's features will you be able to meet the needs of your target audience better than the current market players do?

How is the key idea of your app cooler than others?

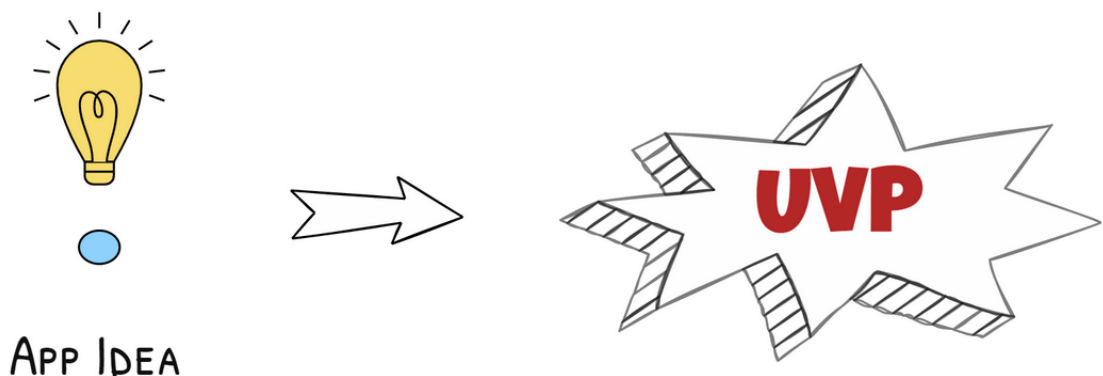
At the same time, it's important to soberly assess your competitors, especially if you plan to enter an already filled market.

For example, we often meet people who have an idea like, "We'll make a Tinder clone and capture a share of the dating app market!"

And we ask them, "Are you sure you can offer the market a more interesting Value Proposition? And do you have the resources to compete with existing serious players?"

There's no point in creating "just another mobile app". It more likely won't take off.

Because at the core of truly successful apps always lies a Unique Value Proposition.





For example, the [Robinhood](#) app has best realized the idea of "Investing for Everyone."

Essentially, Robinhood's UVP democratizes stock trading, letting anyone buy and sell stocks without commissions.



[Strava](#) has become The #1 app for runners and cyclists.

Strava's UVP is in its specific focus on these two groups of athletes, providing detailed tracking and social features for runners and cyclists.



The [DocuSign Mobile App](#) has a **rating of 4.9** in the AppStore because it perfectly represents the idea of "Prepare, sign, act on, and manage agreements on your mobile device."

So, one of the key questions of your business model is - [what Unique Value Proposition will you build it on?](#)

Have you studied your market well enough to be sure that this offer will indeed be unique? Do you have enough resources for the development of a project with such an offer?

3

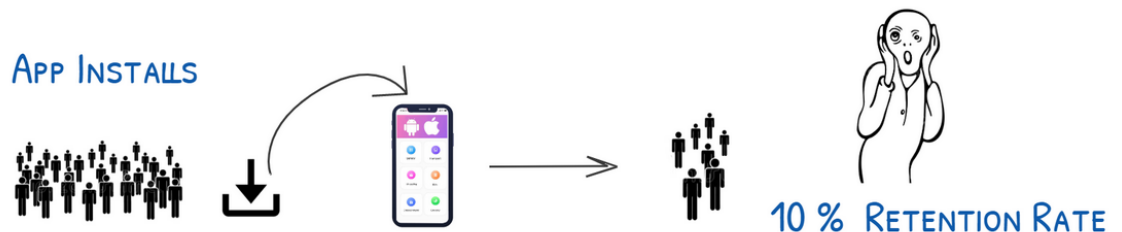
Attracting and retaining users

Do you already understand how you will attract new users, which advertising channels will you use?

What cost per acquisition indicators are you targeting?
Can your business model work with such indicators?

However, the problem for new projects often arises not so much in attracting new users (although this is also not easy), but in retaining those already attracted.

Do you know that on average about 90% of users who have installed any mobile app stop using it within a month?

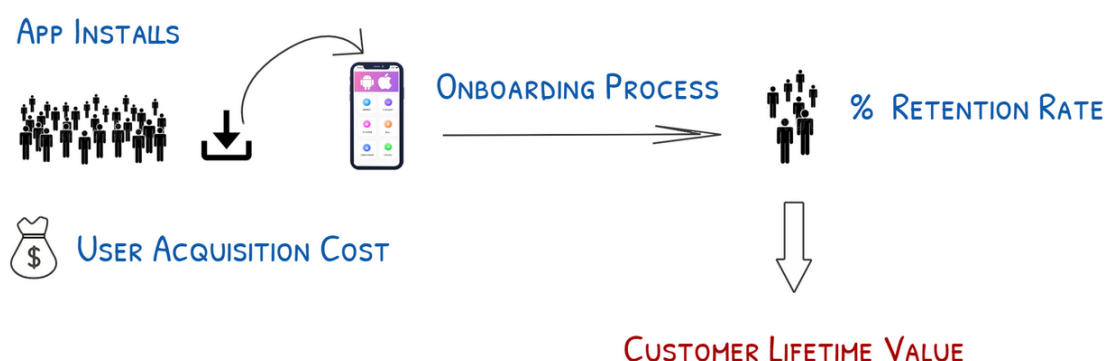


So, the key task is not even to attract new users, but to retain them. Otherwise, the money spent on attracting new users is money down the drain.

Do you understand how you will onboard new users? How will your push notification system be set up? How do you plan to push the user to re-enter your app again and again and perform some actions there?

One of the key indicators in business is LTV (Customer Lifetime Value). This indicator is about how much money one customer brings you on average for the time he remains your customer.

Accordingly, the fight here is for two components of this indicator: for the time the client is with you (we need him to be with us as long as possible) and for the revenue he brings.



It would be good if you already had at least a basic understanding of your strategy for onboarding and increasing LTV before starting to develop an app.

4

Financial model

The financial model usually consists of several components. To start, of course, you need to understand what method (methods) of monetization you plan to use.

Will you make money on a subscription and plan to charge users monthly fees? Will there be a one-time payment when installing the app?

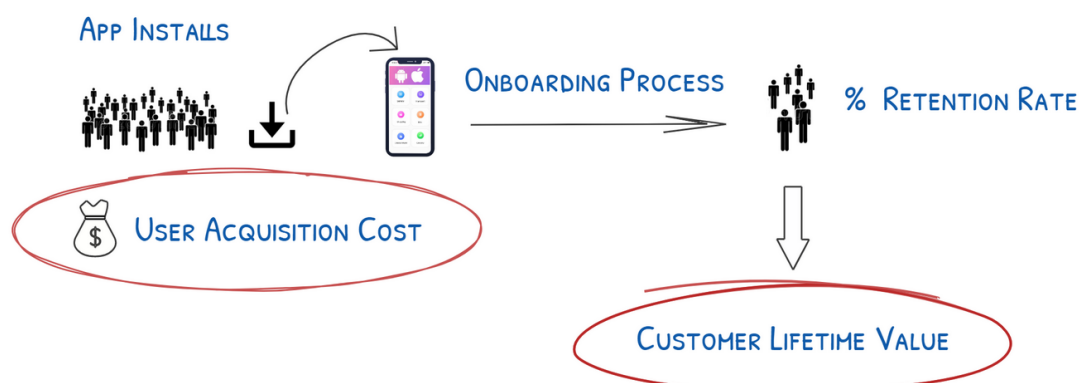
Will you make money on transactions that users make within your app?

Or maybe you will work on a White-label model, offering a customizable, ready-to-use app platform that can be rebranded and sold to other businesses?

Of course, such a choice will significantly affect the architecture and logic of your app.

Part of the financial model is already mentioned cost per acquisition and LTV.

Because the idea of monetization can be quite successful, but if you cannot achieve acceptable cost per acquisition and customer lifetime indicators, the financial model may not be effective.



It's also important to remember that you will have monthly expenses that also need to be considered in your financial model.

Such as costs for servers, SMS sending services for authorization, Google maps (yes, Google needs to be paid for using their maps :)) and other services necessary for the normal operation of your app.

As the number of users increases, these costs, of course, will also grow.

5

Key business processes

Underneath the sleek screens of any successful mobile app, there are often complex and resource-intensive business processes at play.



Take [Badoo](#), the well-known dating app.

Besides its standard dating app features, the company has developed numerous security features to ensure users are real and verified.

For example, they have a photo verification process where users upload a photo imitating a certain pose, which is then reviewed by one of Badoo's 5,000 moderators within a minute.

See, the process of managing these 5,000 moderators is a separate complex task that goes far beyond the app itself.

And in the app, it's just one feature. You can't compete with Badoo by simply copying their interface.



Another example: [Uber](#) is not just an app connecting customers and drivers.

Behind it all is a complex system managing business processes such as dynamic pricing (based on supply and demand), a rating system for drivers and passengers, payment processing, insurance, and logistics software to optimize routes.



So, let's sum it up.

If you're looking at creating a mobile app just as implementing a set of functions - the likelihood of failure is quite high.

It's better to view this project as the embodiment of a specific business model. We've tried to list the key elements in this article (there are a few more, but these are the main ones).

If you'd like to develop your mobile app with this approach, then we'll easily find common ground.

That's because our company always views the development process from a business perspective, through the eyes of an entrepreneur. Keeping in mind the business goals set by our client.

Would you like to develop a mobile app with our help?

Learn about the three simple, risk-free steps to create an app for your business:

<https://www.5pro-app-development.com/3-easy-steps>