

RAPID PROTOTYPING

How to Build MVPs



CLEARCODE

Table of contents

1. How to Build An MVP Using Rapid Prototyping	p.3
MVPs are the first pillar of business success.	
<hr/>	
2. What is an MVP?	p.4
The definition, aims and myths of an MVP.	
<hr/>	
3. Why start with an MVP?	p.6
Generate ideas, test assumptions and minimize risk.	
<hr/>	
4. What do I need to build an MVP?	p.9
Planning, market research, a full-service technology partner and financial resources.	
<hr/>	
5. Which features should be included in the MVP?	p.15
Selecting the features to include and the features to exclude.	
<hr/>	
6. Takeaways	p.17
The key points to take away.	
<hr/>	
7. Our Rapid Prototyping Offer	p.18
Build your prototype in just 30 days.	
<hr/>	
8. About Clearcode	p.19
A Full-service Technology Partner to Startups.	

1. HOW TO BUILD AN MVP USING RAPID PROTOTYPING

The First Pillar of Business Success

The rapid rise of the software-development industry over the past decade has brought with it a number of new techniques, methodologies, and processes that have all changed the way startups and software-development companies approach building web and mobile applications.

The combination of high startup-failure rates and the extremely challenging startup market means that maximizing funds, optimizing time, reducing risk, and gathering vital customer feedback has never been so important.

Traditional IT and software-development approaches focus on solving problems for a defined audience. In the startup industry, knowing if your product will solve a problem or if it will captivate your target audience is hard to gauge without a working application. Startups that follow traditional methods of developing software spend months planning, designing, and building an application without truly knowing if there is a market for it. This significantly increases the risk of instant project failure. A large part of a startup's success is based on its ability to test its target market, validate ideas, and showcase the business's potential.

As a result, a number of new methods have emerged to help software developers plan, build, and release early working applications.

One of the most common forms of a working application is a minimum viable product, or MVP, which has become a pillar of business success and an important way to discover unknown customer and market data.

2. WHAT IS AN MVP?

Defining an MVP

An MVP contains the bare minimum set of essential features and functions required to be deployed and released to a group of early adopters – customers who will use the product and provide honest and valuable feedback.

The term MVP traditionally lies within product design, but was made popular in web-application development by Eric Ries and Steve Blank, pioneers of the lean startup movement.

Even though the traditional definition of an MVP relates to a working product, some consider it to be something that just validates initial ideas, such as a landing page, a customer survey, or even a social-media campaign. While these pre-development-phase activities certainly are a vital part of MVP development as a whole, many within the industry argue that an MVP should both validate ideas and offer the highest customer value as possible from the most minimum set of features.

**You're selling the vision and delivering the
minimum feature set to visionaries,
not everyone.**

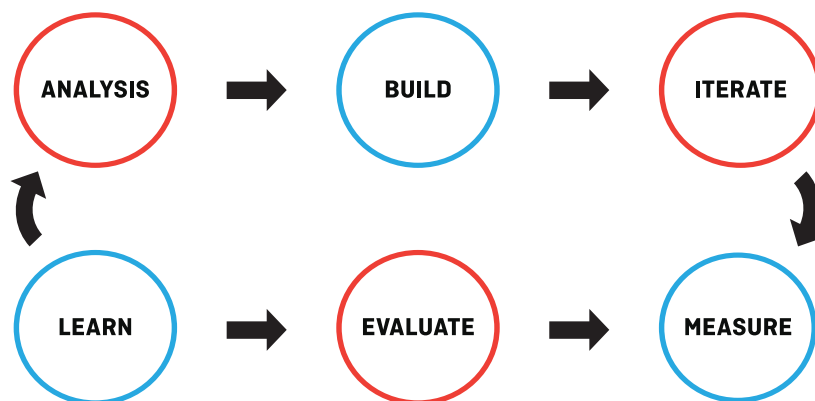
Steve Blank

What is the Aim of an MVP?

The main aim of an MVP in web-application development is to plan, build, and deploy a working product to fulfill its role, which is to gain customer feedback and showcase the business's potential to investors.

Starting with an MVP will also answer a number of very important questions and provide a rich learning experience, allowing you to discover more about your true customers and the market you wish to enter. The initial outcomes of the MVP will pave the way for future development and confirm the subsequent steps to take with the project – whether that be to pivot and change direction, or continue ahead.

The process of developing an MVP is made up of six main stages:



What isn't an MVP?

A MINIMUM PRODUCT

Contrary to its name, an MVP is not a minimum product. In order to successfully build an MVP, you must not look at it as the application's first complete release, but rather as a way of validating ideas, minimizing risk, gaining insightful feedback from potential customers, and selling the potential to future customers and investors.

THE FINAL SOLUTION

An MVP is very much just the starting point. The application's direction will be shaped based on customer feedback and initial market response. All future application development will be determined by these preliminary results and therefore it is not uncommon for the MVP to look completely different than the final product.

A FEATURE-PACKED APPLICATION

The features need to align with the MVP's main goal – to gain initial feedback and test the market. If a feature isn't essential to the application's goals and requirements, then it will need to be left out and included at a later development stage.

3. WHY START WITH AN MVP?

Investors Have Evolved

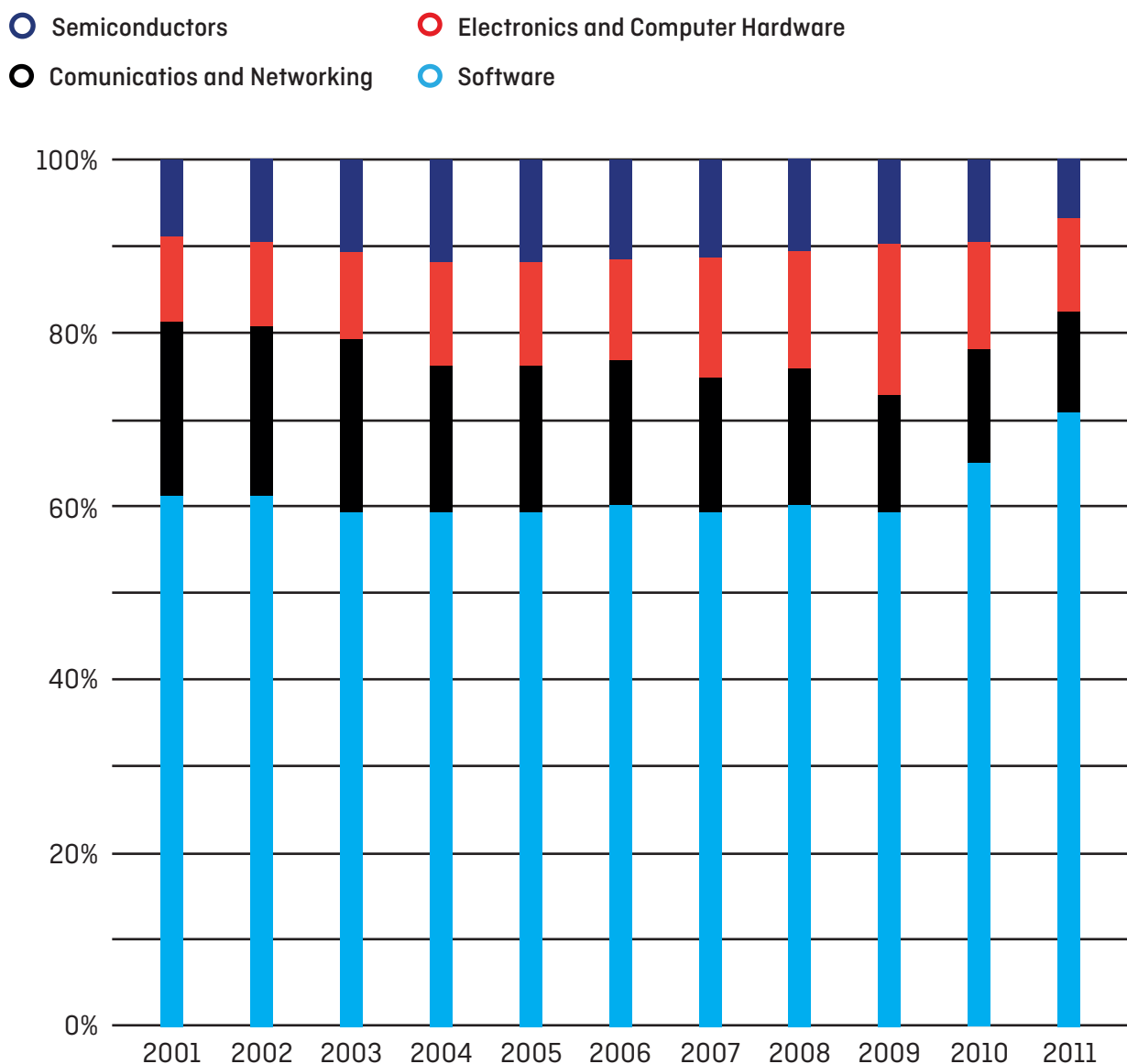
In the beginning of the tech-startup boom in the late '90s, investors would inject their money into promising young startups as quickly as possible. Back then, opportunities were a dime a dozen and the need to fund promising new ideas and technologies was vital to early success and avoiding missing the boat.

After the dot-com bubble burst, investors are now suffering from a case of once bitten, twice shy. As a result, they rarely invest money into invalidated ideas. Startups need to bring more than just an original plan to the investing table; they certainly need to have an established business and an operational application that represents potential before investors start writing out even the most modest check.

While the number of venture-capital investments in software companies rose 10% between 2001 and 2011, the highly concentrated and extremely competitive startup industry means that investors are becoming more conservative with their money and are analyzing the startups' business acumen just as much as their application's potential. They are now more focused on using their money to scale companies, rather than for initial market and customer testing.

By developing an MVP, startups demonstrate their ability to plan, create, and execute – instilling confidence in their potential investors.

Industry Segments of VC investments



Source: Software Eating The World: Facts & Figures, Forbes, 2011.

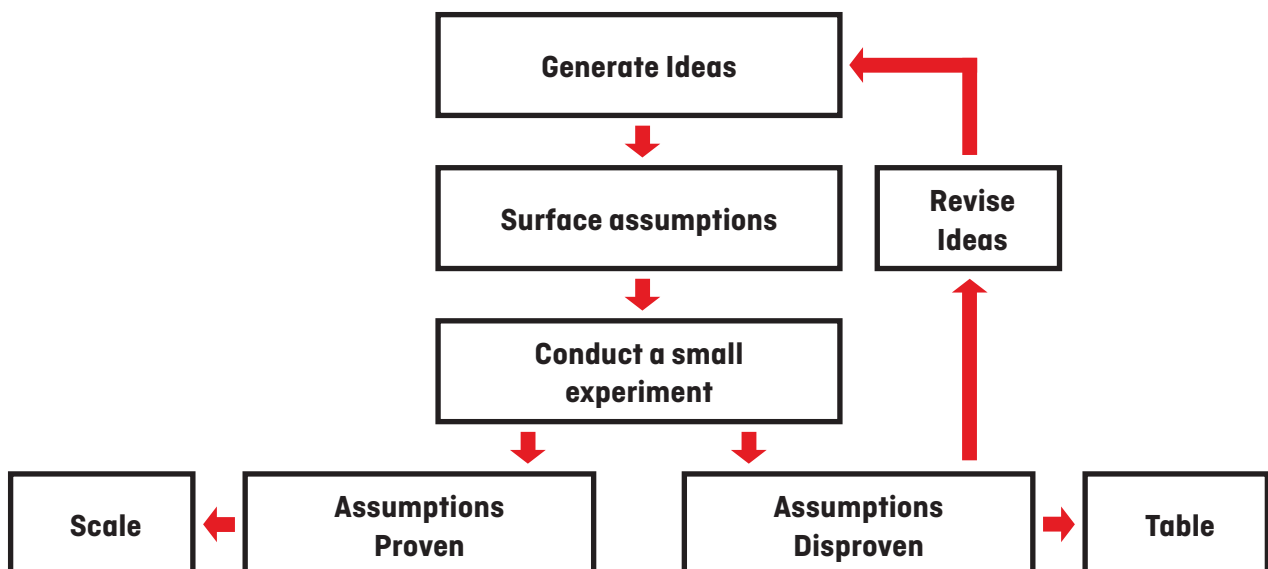
To Generate Ideas, Test Assumptions, and Minimize Risk

The core aims of an MVP are to:

Generate ideas and test assumptions. Every startup idea comes loaded with questions, such as

What problem am I trying to solve?	How can I avoid failure?
Is there a demand for my product?	Will people really want to use this?
Is my product engaging enough to capture an audience?	Will I be able to monetize the application the way I've planned?

By producing an operational MVP, you can easily and quickly validate these initial assumptions and generate new ideas that will help shape the next version of your application.



Minimize Risk

Risk is a common element in all new business ventures. Within the software and web-application industry, however, risk can be a bit easier to manage, as startups can invest a small amount of time and funds to test the waters and react quickly to changing situations based on the conducted customer and market research.

One of the biggest mistakes many startups make is falsely assuming and misjudging their customers' true desire for their product. Even if you think your application is the next big thing, it might not receive the same response from your potential customers.

Another problem is that customer research can produce false conclusions, as there is a world of difference between what potential customers say they will do and what they actually do. Producing a working application essentially asks the potential customers to put their money where their mouth is. If one has expressed interest in using your application by registering via a landing page, then by developing a working application, you are providing them with a way to actually engage with your idea, thus generating accurate research.

Developing an MVP that allows potential customers to use the application helps combat a lot of the primary risks and saves both time and money.

4. WHAT DO I NEED TO BUILD MVP?

Foolproof Planning

Planning is a critical part of an MVP, and it must be foolproof in order to identify potential risks, achieve goals, and set a clear path of development. Planning will help gain a better understanding of the business and assist you and the development team in selecting the most necessary features required for the initial launch.

Customer and Market Research

**We must learn what customers really want,
not what they say they want or what we
think they should want.**

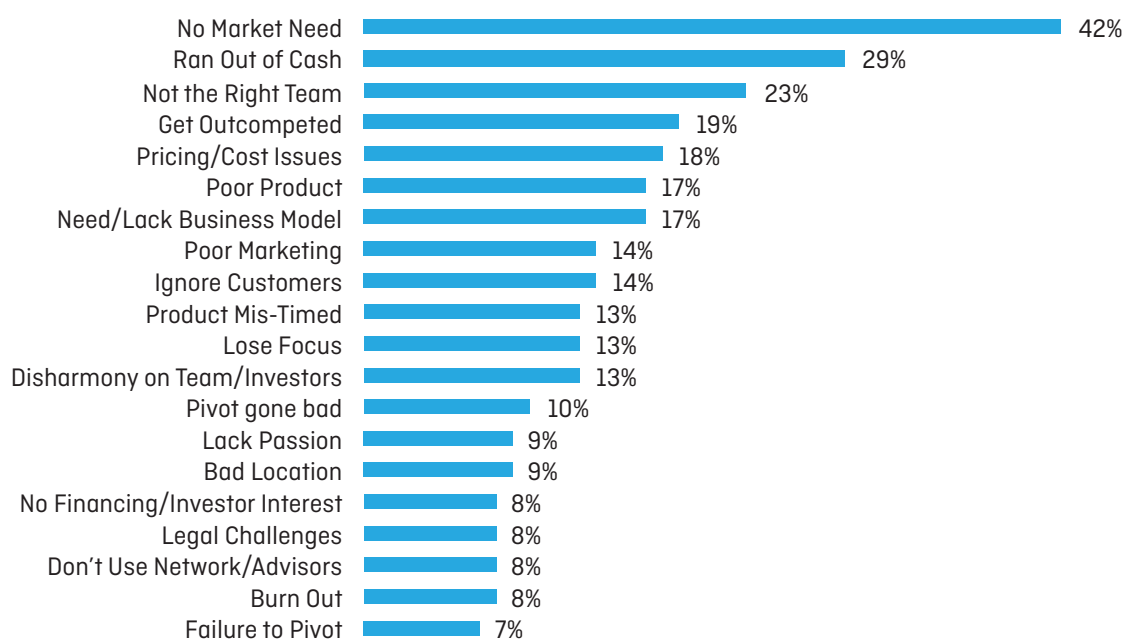
Eric Ries

Conducting thorough market research and fully understanding your prospective customers are the first steps to building a successful project.

According to a recent report released by CB Insights, startups stated that the number one reason why their project failed was because there was no market need for their product. The report highlights the importance of performing comprehensive customer and market research in order to thrive.

Top 20 Reasons Startups Fail

Based on an Analysis of 101 Startup Post-Mortems



Source: The Top 20 Reasons Startups Fail, CB Insights, 2014.

Many startups avoid this crucial step, as they consider the exercise to be an added expense they can do without, and they also don't want to bear the negative feedback. However, in order for an MVP to succeed, the startup must learn and understand the market they are entering and the needs, wants, and behaviors of their future customers. Without thorough analysis, the effectiveness and overall success of the MVP will be severely compromised.

Customer Research

Before you conduct research, you need to clearly define your target customers. A great way to start is by answering these questions:

- Are you solving a problem (if yes, then which problem?) or are you providing people with something they will desire?
- Why will they choose you over your competitors?
- What would they be willing to pay for your application?

Once you have the answers to these questions, it's then time to go out and confirm them.

There are a number of ways you can conduct customer research. Some of the most common methods include focus groups, one-on-one interviews with people who fit your target audience, behavioral research, and surveys.

Market Research

While a large part of market research is spent identifying and learning more about your true customers, it is also vital to thoroughly research the market itself, which will help you uncover:

- The size of the market.
- How many companies operate within the market.
- Who your competitors are.
- The challenges this market faces.
- The overall uniqueness of your application.
- The maturity of the market; is it an established market or is it new?

The results from this research will provide you with a greater understanding of your application's position in the market and will help define the key features and functions for the MVP.

Avenues that you can use for market research include survey results, national- and local-business census bureaus, and interviews with business owners and managers who operate in your target market.

Technology Partner

If you lack the experience and the necessary programming and design skills, or even just the manpower, needed to develop a working product, then finding a quality and dedicated technology partner is essential to your MVP's success.

Finding and hiring a partner to take on all the technical responsibilities and challenges will not only ensure a high MVP-success rate, but it will also allow you to focus on the business side of the project - such as marketing, raising finance, etc.

Traditionally, technical co-founders are software engineers or developers who accept a certain percentage of equity, and in return, provide the technical services needed to bring the MVP to life. The problem with this is that for a large majority of programmers, finding a full-time, very well-paid job isn't difficult, so why would they take a risk on an unvalidated idea that isn't even theirs? In addition, the amount of non-tech entrepreneurs looking for a technical co-founder massively outnumbers the amount of programmers looking for a non-technical co-founder.

The best way to ensure your project contains the right ingredients needed to succeed is to find a software-development partner who can handle all of the technical responsibilities and who also acts as a Technology Partner – without taking equity.

Advantages of Hiring a Technology Partner

- Prevent yourself from giving away your equity early on.
- Receive a dedicated and motivated technical team: A successful MVP means further development and a continued working relationship with you.
- Access to more developers: This will greatly speed up the development process.
- More skills and experience with developing MVPs.

Full-Service Software-Development Company

While looking for a software-development company to become your Technology Partner is important, finding one that offers full-service software development is paramount. In order to plan, design, develop, launch, and maintain your application, it is essential that you find a team that can deliver in all those key areas.

Developing MVPs is a full-team process and requires seamless cooperation from all involved – developers, project managers, designers, and system administrators. It takes a full-service team to deliver a high-quality MVP, not a sole developer.

Financial Resources

One of the first questions startups ask their technical partner is about cost. The complex nature of software development makes it very hard to determine an exact price; however, most MVPs will cost between US\$25,000 and US\$75,000 to build.

Raising Funds

Getting the right amount of funds to cover MVP development is often one of the first challenges startups face. Receiving funding from investors at this early stage based only on an idea is extremely hard. A highly effective way to increase your chances of securing initial funds is to create a high-level prototype.

By working with UX/UI designers to create a prototype of your application, you are providing investors with a visual concept. This enables them to get a better understanding of your idea and vision for the application.

The money used for funding an MVP is often referred to as seed funding – early investments used to support the development of the MVP and the initial stages of the business. There are a number of ways startups can raise seed funding:

- Personal savings
- Friends and family
- Loans
- Angel investors
- Crowdfunding

When deciding on the right option, it's important to carefully analyze and weigh the potential rewards and risks.

Hourly Rate vs. Fixed Price

Another important financial aspect is the payment method to the software-development company. The two options most commonly offered are hourly-rate and fixed-bid models.

Fixed bid

This model is popular among some software-development companies, but is very often the wrong choice for startups. While the idea of paying a fixed price for developing your MVP may seem attractive, as it allows you to better control your budget, it often causes irreparable problems during the development stages:

- **Developers and client work against each other.** The developers want to finish the project as early as possible to maximize profit, while the client wants to squeeze in as many features as they can to maximize value. These competing goals make it extremely hard for both parties to be completely happy and receive an equal and fair deal, meaning that one party will always be disadvantaged, which is not a factor in a successful project.
- **Developing software is a complex process that requires agility:** The fixed-bid model forces the development process to be very rigid and therefore lacks the ability to change direction when required.
- **Spending more time on planning than development:** This is used by developers to reduce the risk associated with developing an MVP; however, it takes time to carefully plan an MVP and quite often doesn't go to plan anyway. This time eats into development phases and causes unnecessary stress and lost money for both parties.

Hourly rate

More and more software-development companies are moving away from the fixed-bid model towards hourly rates, as they greatly benefit both the developers and the client. This model aligns with the agile methodologies and provides the following benefits:

- **More transparency and a better developer-client relationship:** The hourly-rate model provides the client with a better understanding of the progress of their project, as they can see how much time has been spent on different parts and what the results are.
- **Ability to change direction:** As the hourly-rate model coincides with the agile methodologies, the developers are able to develop software progressively. Change is a common part of software development and the freedom and ability to change direction when needed is critical to the success of all projects.

- **Faster development speed:** Because developers don't have to spend a lot of time planning out the project to avoid risk, unlike the fixed-bid model, they can start working on the project earlier, provide working features quicker, and gather feedback from the client that will assist in shaping the remainder of the MVP.

While some startups fear that the hourly-rate model doesn't provide developers with enough motivation to succeed, the opposite is actually true. Software-development companies rely on the success of client projects to fund further development. If an MVP is successful, then the startup will need to develop the application further, resulting in continued cooperation with the development team. In addition, the hourly-rate model requires a close working relationship with the startup, which essentially makes the developers part of your team.

5. WHICH FEATURES SHOULD BE INCLUDED IN THE MVP?

As you consider building your own minimum viable product, let this simple rule suffice: remove any feature, process, or effort that does not contribute directly to the learning you seek.

Eric Ries

Deciding which features to include in the prototype is a difficult process. Applications are made up of a series of features, which makes choosing the right ones hard. The main thing to remember is that the features included in your MVP should ONLY be the ones that connect with its goal.

In the end, the process of deciding which features to include will be a combined task between the client and the software-development partner (their Technology Partner). By analyzing the results from your customer and market research, understanding your goals, your core business, and vision for the company will enable the Technology Partner to plan out features that will be included in the MVP.

Once a list of the main features has been determined, the developers will often look at ways to combine certain smaller features to make one main feature. This is a worthwhile activity, as it will save precious development time.

While the features are different for each MVP, there are some that have no place in any MVP and should be left out completely.

Features to Leave Out

Cool features

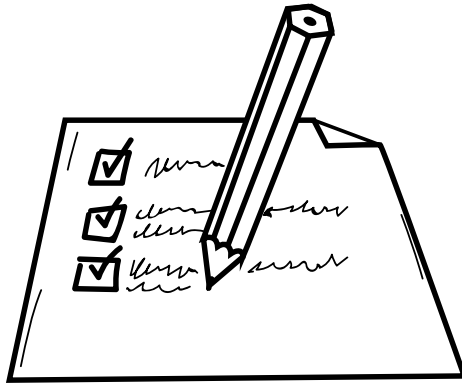
These can be defined as features that have no real purpose except to look good and add very little value to the MVP. An example of this would be social-media integration, which rarely adds value to a majority of MVPs.

Copypat features

Just because your competitors have a certain feature, doesn't necessarily mean you have to. Implementing features on your MVP that match those found on competitors' applications will add more time to the development phases. When deciding which features to include, don't focus on the ones your competitors have - the aim is to develop your MVP based on your application's goals.

Features requested by early adopters

Even though users help shape your application, you should be cautious when adding features requested by users – especially the freemium ones – as the features may not be suitable for others and won't necessarily result in a better user experience. In addition, it is often hard for users to truly know if they want certain features, especially without having actually engaged with them. Implementing user-requested features should be based upon thorough research and analysis and will generally occur after the initial MVP phase.



TAKEAWAYS

- ✓ Beginning with an MVP is the first pillar of startup success.
- ✓ Thoroughly plan your MVP.
- ✓ Conduct detailed customer and market research.
- ✓ Find a full-service Technology Partner.



BUILD YOUR PROTOTYPE IN 30 DAYS!

A working prototype is the bridge between ideas and an MVP.

Are you looking for a way to quickly bring your app idea to life and begin testing it?

Let us help you have a working interactive prototype up and running within just a couple of weeks! Click the link below to learn more about our rapid prototyping plans.

FREE CONSULTATION



CLEARCODE

We specialize in programmatic advertising, big-data analytics, and creating enterprise-grade SaaS platforms that experience exponential growth, attract investment dollars, and undergo exits.

We assume the role of Technology Partner to startups, working closely with them on every aspect of MVP development, from planning and feature identification, to UX/UI design and prototyping, development, and application support and monitoring.

E-mail

info@clearcode.cc

Phone

+1-917-300-1772