Guide

Guide MVP

Spryker

MVPs and incremental deployments to drive continuous delivery and business value

How to resolve transformation pressure in a few weeks

Content

Chapter 01

MVP is more than a Buzzword

Lean Startup,

Trial-and-Error,

MVP ...

...anyone working in digital business knows the abundance of buzzwords. But by spring 2020 the latest, when the world is upside down, it should be clear to every decision-maker in commerce that an MVP is not only a well-founded technical approach but also a valuable business measure.

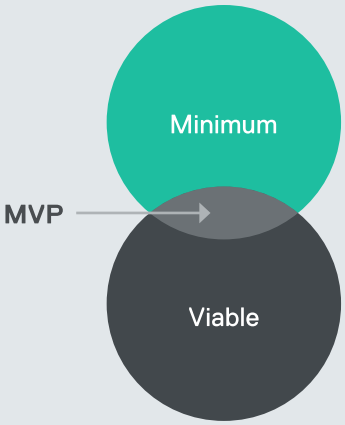
As Comprehensive as Necessary, as Simple as Possible

The "Minimum Viable Product" is the introductory, but functional version of a product. With the help of the MVP, you can collect user feedback, test demand and usability before you invest more in the development of the product or idea: Or, if the data demands, abandon it and concentrate on new, more promising ideas. This is particularly true in the case of digital products and services, such as online shops, content portals, marketplaces or mobile apps.

Every MVP must accomplish two conflicting goals:

01 On the one hand, the product should be as simple as possible so that it can be developed quickly.

02 On the other hand, it must be mature enough so that it works and generates value for the customer.

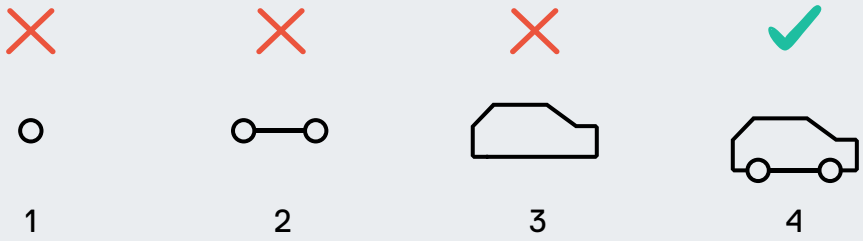


In order to realize an MVP quickly and sustainably, it is crucial to choose the right technical approach in development. **Incremental deployments can be a major contributor to the success of an MVP**. If done right, they will increase the change-throughput and foster an agile mindset on testing and automation.

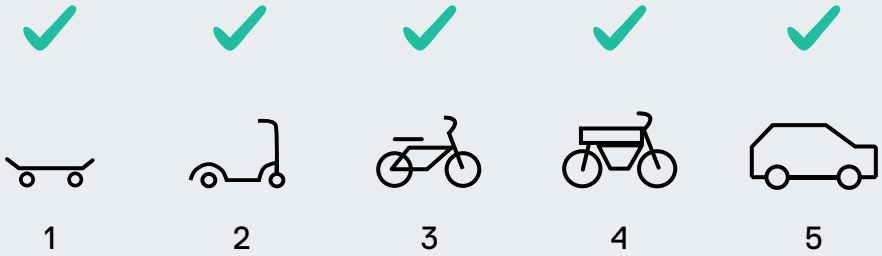
In this guide, we illustrate both the technical and business benefits that can be achieved by an MVP. We elaborate on **technical best practices to build an MVP and discuss the potential of automation in the form of continuous delivery**. Also, for any non-IT stakeholders you will find basic explanatory materials about MVPs in the appendix.

The Approach is Simple:
You are planning to build a vehicle. How do you go about it?

How **not** to build a minimum viable product



How to build **a** minimum viable product



Chapter 02

Why the MVP approach is the best solution for rapidly changing markets

In today's market, the pressure to transform is high. When the improvement involves digital projects, the smallest details are often planned ahead, every single requirement is defined prior to market entry. The intention: To have everything perfect before going live.

Beware: Today, market requirements change faster than a perfectionistically planned digital project can be implemented. Fast action and adaptability are the keys to success.

Markets and customer needs are changing faster than ever. This is not a new insight, but in the turbulent spring of 2020 it is being made clear to everyone once again. In these times, it is essential in commerce in particular to be able to adapt quickly to the changing requirements of the market and customers. Technologies are developing rapidly, new customer touch points are emerging, competitive pressure to digitize is increasing and customer expectations are becoming more and more complex. Anyone who loses too much time in coordination processes and tries to predict the unpredictable in long planning cycles runs the risk of being left behind by fast-moving market changes. With increasing project length, the probability of getting a positive ROI (Return on Investment) decreases.

The challenge of digital transformation lies in translating the digital project idea into a concept that can be tested and put into practice quickly, and then it must be implemented in a way that is fast, data-driven, and functional.

> *"More than 60% of the features in software products are never used!"*[1]

The CHAOS report by the Standish group defines success factors and investigates reasons for failure of digital projects. The results are clear: The MVP approach is not optional.[1]

Project plans with an MVP approach

16.2%

...of the examined projects were successfully completed:

- finished in time
- no cost overruns
- all originally required functionalities included

Key success factors:

+ Involvement of end users
+ Support from senior management
+ Clear requirements

Project plans without an MVP approach

52.7%

... of the projects in the study were completed with cost or time overruns.

31.1%

... of the companies surveyed had to abandon their projects.

Cause of failed projects:

– Missing input from end users
– Incomplete/unclear requirements
– Frequent requirement changes

[1] CHAOS Report: Decision Latency Theory (2018) Package, https://www.standishgroup.com/store/services/10-chaos-report-decision-latency-theory-2018-package.html

Chapter 03

The right technical way to build an MVP

There are a variety of techniques to deploy new applications to production. Two of the main criteria that should drive the decision for one deployment strategy are the impact of change on the system and on the end-users. In this regard, the sheer number of adjustments involved in an MVP process makes some form of incremental development the best choice.

Technical and business benefits of incremental deployments

Technical benefits of incremental deployments

- Quick user feedback on functional updates
- Better team learning process
- Decreased number of friction points, long-lasting code change tends to get obsolete

Increased change-throughput / IT-productivity

Business benefits of incremental deployments

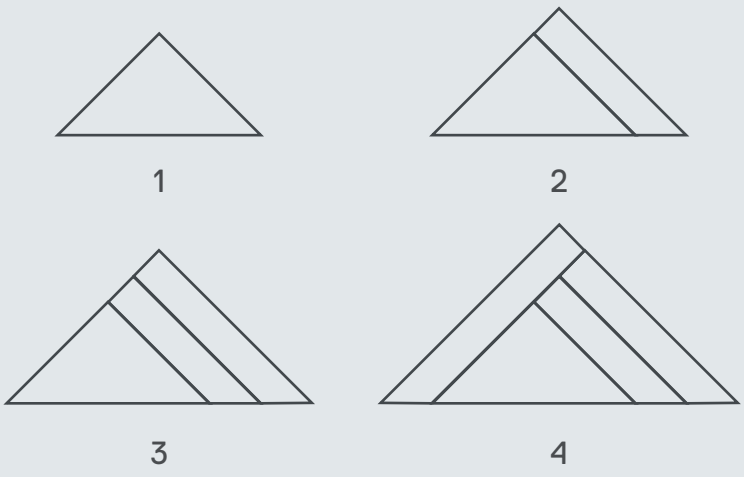
- Decreased number of tasks in progress means higher focus and predictability
- A decent level of delivery automation frees team time for delivering business value instead of operational activities
- Increased team trust in system stability and easy deployments

Agile mindset on testing and automation suits the fast-moving market conditions

MVPs and incremental deployments have a lot in common but are not identical. An increment must be usable, but may not contain those features that make the MVP viable. With every Sprint, the increment becomes more enhanced until it reaches a state that is an MVP and the Product Owner can choose to release it.

Any approach that divides the software development process in small iterations can be called "incremental." However, there is more than one way to build a product incrementally. To illustrate that, John Mayo-Smith introduced the metaphor of building pyramids[2]. Traditionally, one would start with the base and then, with each increment, add a new level on top of the previous one. This can definitely be considered an incremental process. However, the weakness of this approach is that there is no real pyramid until the last iteration of the process.

Alternatively, one can start with a small pyramid right away and then use additional iterations to increase its size. In contrast to the first approach, a real pyramid is already present in each step and only the size increases incrementally.



Of course, if an MVP with all its "build-measure-learn" feedback loops is to be built, the second approach is the right one. This allows for the MVP (small pyramid) to be launched quickly, and then have a deployable product with new features (bigger pyramid) after each iteration.

[2] Two Ways to Build a Pyramid", https://www.informationweek.com/two-ways-to-build-a-pyramid/d/d-id/1012280, accessed 22 April 2020.

Benefits and challenges of continuous delivery

For many businesses, it makes sense to introduce some form of automation to reduce the manual effort related to the high number of (incremental) deployments of an MVP. Continuous delivery (CD) requires an automated testing and release process. Then, it is possible to deploy your application at any point of time by clicking on a button. Those releases could happen daily, weekly, fortnightly, or whatever rhythm suits your business requirements.

Benefits of continuous delivery

- Operational costs reduced in mid/long-term
- Increased number of code changes
- Decreased number of in-progress tickets, therefore lower refactoring costs

Challenges of continuous delivery

- Initial setup is a bigger effort
- CD requires a reliable CI process on application and infrastructure level - all good CI processes started with a simple question: “What parts of the app are critical for my business model?”
- Consider IaC (Infrastructure as Code) for better visibility and including it to CD process
- CD also requires continuous requirement specification

Optimizing the pipeline from idea to release makes possible business value continuous delivery

Chapter 04

Business value of an MVP

Many executives and project managers do not find the MVP process to be very intuitive. Despite the technical benefits, it can be challenging from a business perspective to start small and break down a complex project into prioritized, small phases. The entire planned scope can not always be completed right away and due to the MVP's nature of pushing trial and error, initial user feedback can tremendously impact the scope or even the success of any project. But modern market conditions make MVPs more and more inevitable on the business side as well.

MVP principles meet the momentum of modern markets

A digitalized world with rapidly changing trends and ever shorter digital half-lives leaves no room for months of planning. Under the premise 'think big, build small' an MVP is becoming a valuable business measure in order to ensure quick adaptation and keep businesses on top of the latest market developments.

- **Progress** instead of perfection!
- **Test phase** instead of long development and planning phase
- **Benefits** instead of complex properties
- **Agility** instead of the waterfall model
- Focus on the **next step** instead of the last
- Success is measured not only in terms of sales, but also in terms of **cost savings**
- **Information about user behavior** through early data evaluations instead of subjective expectations from the project manager

In some retail segments, using the MVP process may seem highly unusual. However, executives need to realize that speed can determine success or failure; and you need to know for sure that you are heading in the right direction from the beginning.

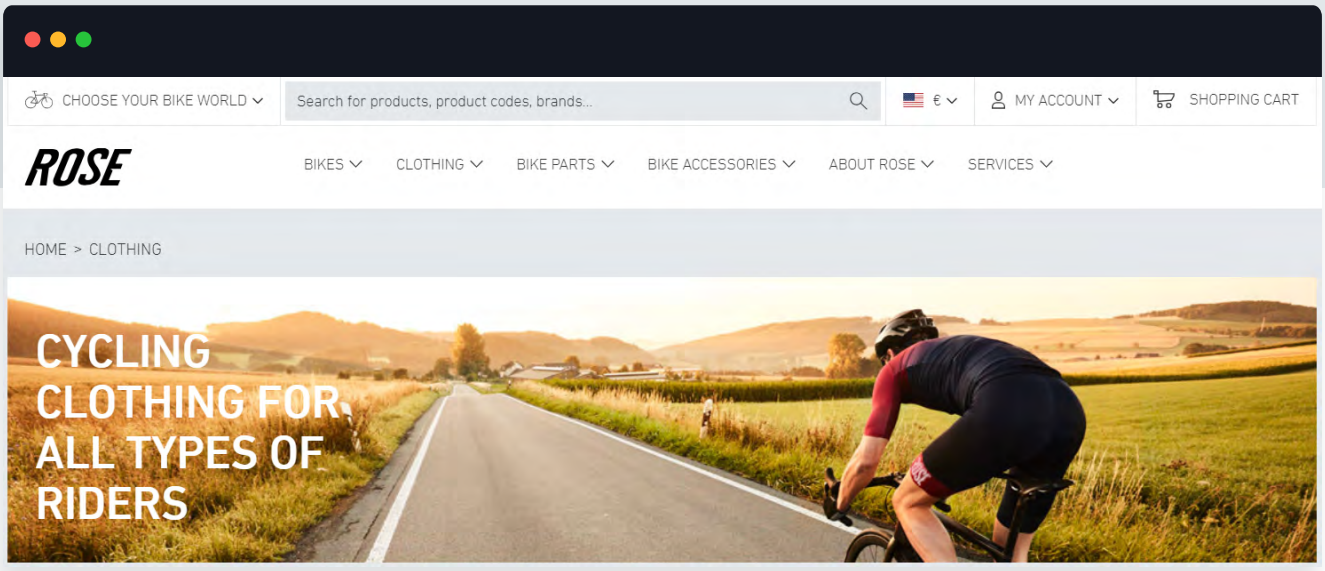
I myself made the mistake of not building an MVP. With the Document App, our company just hit a wall. An MVP would have helped me a lot."- Frank Thelen, Multiple Entrepreneur and Investor

Chapter 05

MVP Case: ROSE Bikes adapting to market changes

The direct path from target definition to implementation

In order to survive and grow in a very dynamic environment, ROSE Bikes relaunched its e-commerce shop - including the introduction of new functionalities. A successful example of an MVP process in practice.



For potential customers of this company, cycling is more than a way of getting around; it is also a way of life. For this reason, when ROSE Bikes thinks about its products and user experience, their core values are functionality and customer-oriented services brought to the extreme with a high degree of customization. They want to convey a mix of high quality technology for cycling enthusiasts. But ROSE felt their current online shop fell short of this experience. They needed more than a standard shop for e-commerce. When they relaunched, they added a new core function: the option to configure bicycles individually.

For the relaunch, the company followed a classic MVP approach. With a clear view of the most relevant features, they kept their e-commerce transformation focused and fast.

> *We're dealing with a market that is more and more competitive, the margins are falling and processes and products are more and more digitized. We have to adapt quickly and flexibly to changing conditions and try out new things."*- Thorsten Heckrath-Rose, Managing Director at ROSE Biker

Step 1: Define the Primary Goals

ROSE Bikes:

The top feature for ROSE bikes is the bike configuration. The objective is to allow customization of their ROSE bike and configure it for customers' individual needs.

Primary Goal: Bike Design

Step 2: Define the Customer Journey

ROSE Bikes:

Once the primary goal is set, the individual phases of the desired customer journey are defined. The customer can choose between different bicycle categories, so relevant products are displayed. On the product detail page, the selected bike can be adapted to the desired design - from size and weight to colour.

Primary Goal: Bike Design

Customer Journey

Step 3: Develop a Feature List for each Phase

ROSE Bikes:

Once the phases have been defined, a list of features needs to be created for each phase of the customer journey. At this point, as many features are collected as possible but not prioritized.

PIM

Feature 1: Bike Design

- Product attributes for the various parts / elements
- Filters for the configurator
- Search functions with Elasticsearch for advanced searches

Checkout

Feature 2: Checkout

- A wide range of shipping options
- Discount and rebate options
- Shopping cart with detailed product overviews

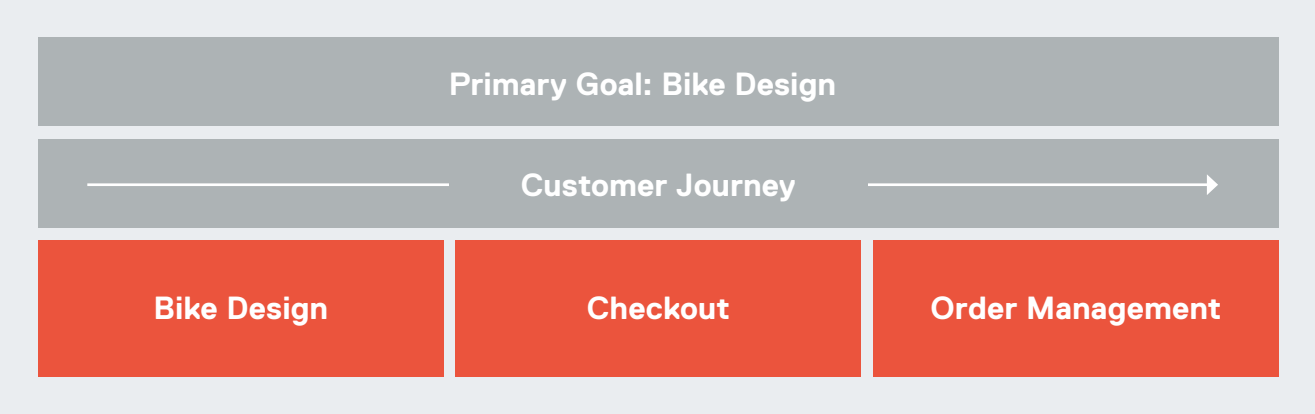
OMS

Feature 3: Order Management

- Order overview in the back-end
- Customer accounts in both front-end and back-end
- Automatic order processing

The integration of product attributes and tags allows for better categorization of the shop.

With the combination of product attributes and filters, the configurator's search function as well as the rest of the shop can show targeted and detailed results. Using Elasticsearch added many strong functions, for example the synonym search. These features make the user journey in the shop simpler.



Step 4: Prioritize Features

ROSE Bikes:

Key questions for prioritization:

- What added value does the feature bring for the user?
- How much time and money must be put into the feature?
- How can I use the ROI as a basis for further development?
- Which features form the foundation?
- Which features are not necessary?

Primary Goal: Bike Design		
Customer Journey →		
Bike Design	**Checkout**	**Order Management**
Product attributes for the various parts / elements	Shopping cart with detailed product overviews	Customer accounts in both front-end and back-end
Filters for the configurator	Discount & rebate options	Order overview in the back-end
Search functions with Elastic-search for advanced searches	A wide range of shipping options	Automatic order processing

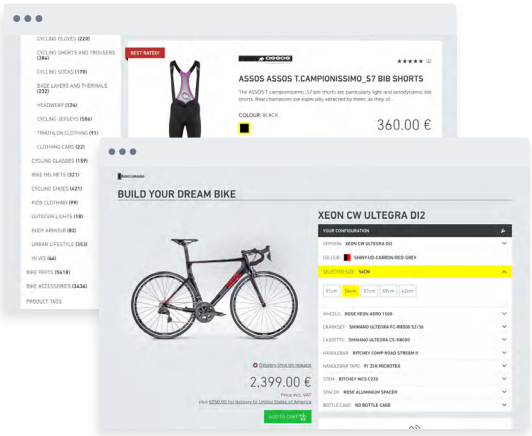
High priority

Low priority

Step 5: Definition of the MVP

The top features represent the Minimum Viable Product. The rest can be included in later versions. This type of mapping helps to plan projects more consciously. The focus is on the primary, basic needs of the user and the effective usability of the MVP shop. These form the MVP.

After an agreed-upon test period, you can evaluate the response and optimize needs, and you can expand the MVP.



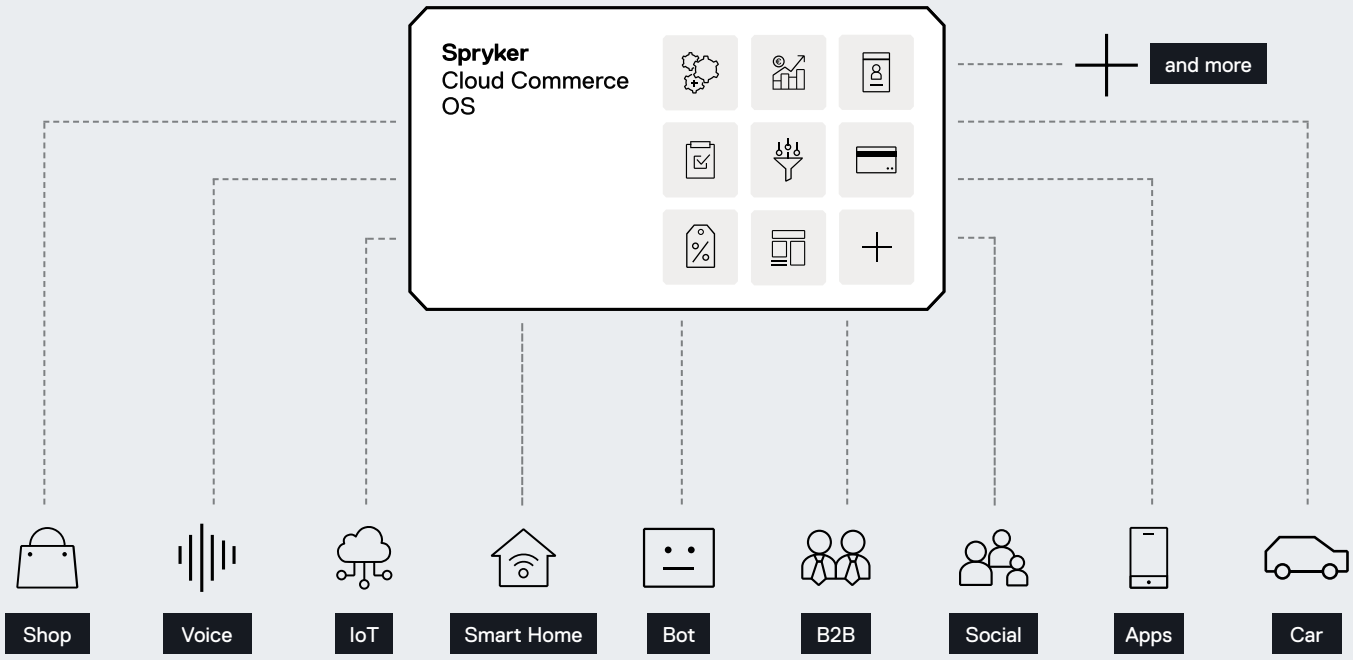
How can Spryker help?

Spryker is ideally suited for MVPs and continuous deployment processes. With Spryker's modular structure and an MVP and API-first approach, **different features or new customer interfaces can be added to the existing system very fast**. And since the system is based exclusively on standard technologies such as PHP, PostgreSQL, Redis or Elasticsearch, it can be **operated in any environment**.

Spryker is operated by our customers **on premise, on Baremetal-Hosting as well as in modern cloud environments**. Which systems are used for deployment is entirely up to the project team. While many companies like to use docker containers with or without Kubernetes, others still prefer deployment scripts based on Ant or Maven. Spryker does not limit this choice in any way.

In addition, Spryker already comes with **quality assurance and testing environments as standard**. We would always recommend to make use of the automated checking of your own artifacts within your own deployment pipeline. These tools enable our core team to release new modules and versions on a quasi-daily basis.

And in order to support you in the rapid development necessary for MVP building, Spryker also offers the **code generator Spryks**. It can quickly generate the needed code pieces, and even re-generate if needed, reducing manual work and related costs.



Appendix

Materials for non-IT stakeholders

What is a Minimum Viable Product (MVP)?

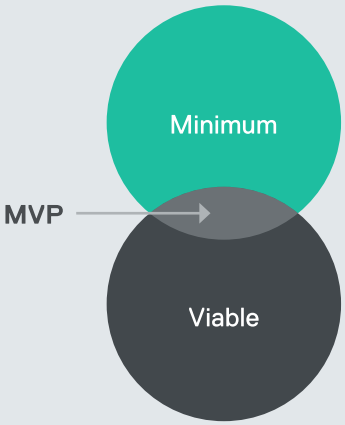
As Comprehensive as Necessary, as Simple as Possible

The “Minimum Viable Product“ is the introductory, but functional version of a product. With the help of the MVP, you can collect user feedback, test demand and usability before you invest more in the development of the product or idea: Or, if the data demands, abandon it and concentrate on new, more promising ideas. This is particularly true in the case of digital products and services, such as online shops, content portals, marketplaces or mobile apps.

Every MVP must accomplish two conflicting goals:

01 On the one hand, the product should be as simple as possible so that it can be developed quickly.

02 On the other hand, it must be mature enough so that it works and generates value for the customer.



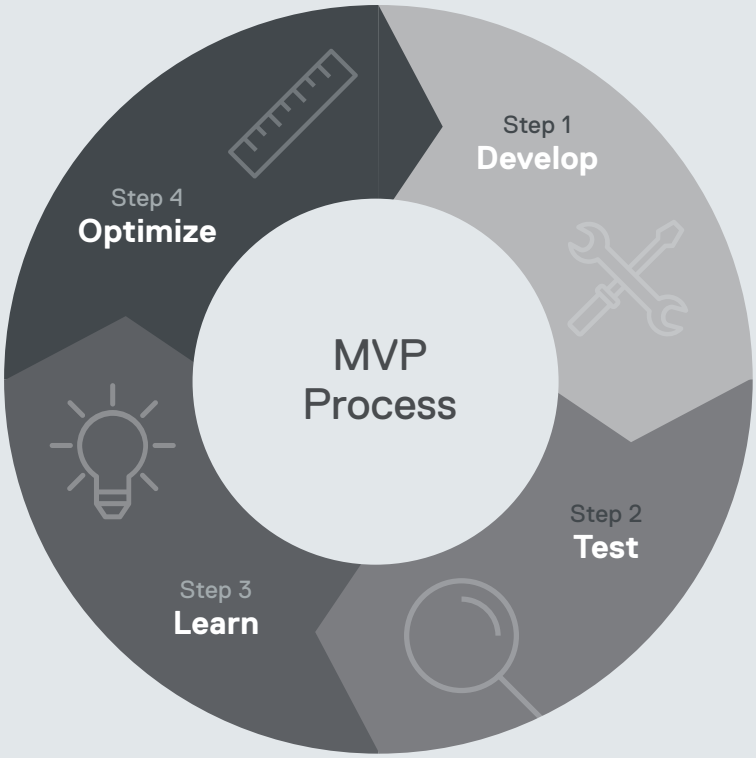
In order to realize an MVP quickly and sustainably, it is crucial to choose the right technical approach in development. Incremental deployments can be a major contributor to the success of an MVP. If done right, they will increase the change-throughput and foster an agile mindset on testing and automation.

In this guide, we illustrate both the technical and business benefits that can be achieved by an MVP. We elaborate on technical best practices to build an MVP and discuss the potential of automation in the form of continuous delivery. Also, for any non-IT stakeholders you will find basic explanatory materials about MVPs in the appendix.

How an MVP process works

The MVP process can be imagined as a cycle: Develop, test, learn, optimize. This approach increases the likelihood that the product is heading in the right direction. If the testing phases show that this is not the case, there is still a chance to course correct. **Cost and risk are reduced by testing and validating or discarding several small instances.** The process from conception to the first launch ideally takes only a few weeks.

This prevents losing sight of the market and, above all, customer needs during the process. What works well in the market today may be completely outdated and unusable tomorrow.



Spryker

Do you want to learn more about commerce best practices and innovative digital solutions?

Visit

spryker.com

Contact

hello@spryker.com

Do you have feedback for us? Please feel free to contact us!

Dorothee Thomsen
Senior Content Marketing Manager

content.marketing@spryker.com

Spryker Systems GmH
Julie-Wolfthorn-Straße 1
10115 Berlin / Deutschland

T +49 / 30 / 208 49 83 53
M hello@spryker.com
W spryker.com

Spryker