

What is SaaS?

The Ultimate Guide to Software as a Service

Everything You Need to Know About Moving to a SaaS Model

Table of Contents

What is a SaaS Model?

Introduction

Why Should You Consider Moving to a SaaS Model?

Who is a SaaS Model Right For?

What is the Process to Start Building a SaaS Product?

How Much Does it Cost to Build a SaaS Product?

SaaS Examples

Conclusion

What is a SaaS Model?

A SaaS model treats software licensing and delivery as an ongoing service rather than a single purchase. Users typically purchase the software via a subscription and access it online rather than buying it and installing it on individual computers.

Introduction

SaaS stands for software as a service. It is a method of software delivery and licensing in which software is accessed online via a subscription rather than bought and installed on individual computers. This business model creates so many advantages for both developers and users that [33% of companies are using more SaaS applications than they did in 2016, and roughly three-quarters of companies will have nearly all of their apps as SaaS by 2020](#). Keep reading to better understand what is SaaS and learn more about the advantages of making the switch.

What is a SaaS Application?

SaaS applications are simply software applications that are provided online rather than as desktop applications. For example, Microsoft Outlook and Apple mail are traditional desktop email applications, while Gmail is a SaaS email application.

SaaS vs Cloud Computing

When many people try to understand what is SaaS, they equate it to cloud computing. This is partially correct because SaaS is a subset of cloud computing.

You can define cloud computing as a broad set of computer services where some aspects are handled online rather than all on the user's own computers and servers. By contrast, you can define software as a service as a piece of software accessed remotely through the cloud. In SaaS, all data related to that software is typically held on the software provider's servers rather than on the client's own computers.

A good comparison is the Microsoft Office suite. If you buy the traditional one-time purchase software and store some of your documents online for easy sharing, you're using cloud computing but not SaaS. If you've switched over to their 365 monthly billing model where you can access the software in the cloud or have your desktop version automatically update to the latest version, that's SaaS.

SaaS vs PaaS

When it comes to what is SaaS versus PaaS, the answer is what level of service is provided. SaaS-based platforms provide a total software package including the application itself and data storage. With platform as a service, you receive the infrastructure to create and run your own applications, but those applications and the accompanying data remain fully under your control.

Good examples of SaaS applications include Google Apps, Dropbox, and Salesforce. PaaS examples include AWS Elastic Beanstalk and Google App Engine.

Why Should You Consider Moving to a SaaS Model?

The SaaS business model helps both the product owner and its clients. Key benefits include the following.

Time to Launch

Each SaaS application is already installed and configured in the cloud. This minimizes common delays resulting from often lengthy traditional software deployment.

Once development is complete, the software is ready to use. There is no additional time waiting for installation or resolving unexpected issues with the client's hardware once the software is deployed on site.

Scalability

There is no need to worry about capacity as the software's usage or users grows. Moving to a SaaS model means organizations can add users as needed without worrying about procuring new hardware and other infrastructure components.

Clients can receive additional storage space or bandwidth with the simple push of a button. When additional user licenses are needed, such as for new hires, they can be obtained by creating a new user ID and password rather than having to purchase additional physical software.

The product owner must be prepared to provide additional capacity but achieves economies of scale by hosting multiple products in the same place. Therefore, these costs are much lower than they would be for the client to upgrade their own hardware. In addition, the product owner can recover incremental costs by billing based on usage.

Costs

Costs in a software as a service model are also lower in several other ways.

TOTAL COST OF OWNERSHIP

Even though clients often focus on the ongoing fees, the total cost of ownership is generally lower. This is due to a combination of lower hardware costs and not losing efficiency as traditional software becomes dated. In addition, the client doesn't need to maintain a physical space to hold servers or other hardware.

INITIAL SETUP COSTS

Initial setup cost reductions include a much lower upfront software cost if any upfront fee is charged at all. The client also avoids purchasing new hardware and buying or renovating a physical space to hold it.

SUPPORT RATES

Efficiencies in providing support provide further cost reductions that the product owner can capture as profits or pass on to the client in lower fees. These include a greatly reduced need for on-site visits, incremental development serving as preventive maintenance, and having multiple clients using software based on a common set of features rather than as truly independent products.

MAINTENANCE

Eliminating client-site hardware drastically reduces maintenance needs and downtime. The client never has to bring their system down for hardware upgrades or repairs, while the product developer can share backup systems between applications. The client's IT team can also avoid spending time and money performing preventive checks to ensure system reliability and avoid unnecessary downtime.

Additional Benefits for Software Owners

In addition to the shared benefits, SaaS companies receive the following business advantages.

INCREASED LIFETIME VALUE

Upselling and cross-selling are built into SaaS models. Instead of a single chance to pitch the customer on how many features they should take on, you can give them a menu of options. As the customer's needs change or they see that you have an easy solution to a pain point, they can upgrade their services. This allows you to capture additional revenue even when the customer only selects your most basic options on the original sale.

INCREASED CUSTOMER LOYALTY

SaaS companies gain a number of tools to increase customer loyalty. This includes the ability to add permanent or temporary benefits such as free features or storage, the ability to earn discounts, or gifts and contests.

The app can also allow for built-in communication between you and the client. The ongoing conversation can help you maintain a rapport with the client while providing a high level of service that keeps them from exploring other options.

Finally, even though SaaS can sometimes feel cookie-cutter and unpersonalized, adding design flexibility can allow the client to tailor the software to their own preferences and branding without affecting core functions.

BETTER METRICS

A SaaS model provides real-time [metrics](#) to understand how your clients use your software. This lets you make more accurate business projections and increase the lifetime value of your clients by being able to proactively adapt to their needs. You'll also have better data to help you decide when it's time to scale or pivot your product.

INCREASED VALUATIONS

Company valuations are generally based on revenues, cash flows, and/or profits. This includes both current numbers and future projections.

On-premises software providers typically see a plateau as their sales peak and previous buyers stay with their original software and don't make new purchases. On the other hand, SaaS providers typically see growing revenue as long as they control customer churn. Because of these growth projections, [SaaS companies typically trade at much higher valuation multiples than other providers](#).

Additional Considerations for Your Clients

There are also several selling points you can use to help convince your clients to transition to a software as a service model.

BUDGET FLEXIBILITY

Buyers with a consumer mentality often object to ongoing fees, but a subscription model provides the ability to switch or stop service at any time without losing a large upfront cost. Clients can quickly add features when they need them and later scale down if they need to cut costs.

EASIER BUDGET APPROVALS

One reason that many companies keep outdated and inefficient software is the difficulty of convincing managers to approve a large one-time expense. This could be because the company truly doesn't have the money or because managers don't want to take a hit on the income statement.

A low monthly or annual fee that makes up a small portion of a company's overall expenses can be much more appealing to these managers.

CONFIGURATION AND INTEGRATION

Embedded settings still give clients a high degree of ability to customize the software to their needs. Integration with other services, either your own or through third-parties, further gives clients the ability to tailor a solution to their specific needs while gaining the efficiency of having a single connected system.

TIME TO LAUNCH

Clients also gain by having a shortened sales cycle. Once a SaaS application is operating, the time to launch is almost immediate. There is little to no need for custom development, and there is no installation time. In many cases, clients may be able to create their accounts and start using the software on the same day they sign the contract.

SIMPLICITY

On-premises software that is highly customized or designed for anticipated future needs can be needlessly complex. By focusing on core features, current needs, and continuous incremental improvement, subscription software and services can be much more user-friendly.

EASY TO REVIEW USAGE AND BILLING

Built-in analytics allow customers to easily review their usage and what they're getting for their money. Customers who aren't making full use of their features can be guided to downgrade rather than drop the service entirely as a result of feeling overcharged. Those near their usage limits can plan for a potential cost increase. Finally, those who are currently paying the right amount can see exactly what value they're getting for their money so that they're less likely to explore other options.



Who is a SaaS Model Right For?

As shown by the above benefits, the answer to when to use SaaS is when you're looking to:

- Reduce your time to benefit
- Lower costs
- Make your software easier to use
- Expand to new markets
- Have a more stable and predictable business model

However, if you fall into one of the below situations, SaaS models may not be right for you.

Need for Upfront Payments

Over time, SaaS makes revenue projections and future cash flow forecasting easier, but in the short term, it can leave some gaps in cash flow.

If you're relying on upfront payments to recoup development costs or cover ongoing operational expenses, it may be too early to switch to a SaaS model.

Accelerated Revenue Recognition

One of the major revenue recognition differences between SaaS and OPS (on-premises software) is that, in many situations, OPS vendors are able to recognize license fees upon delivery and implementation service fees as the implementation services are provided. Under a SaaS model, up-front payments for licenses and implementation that don't have stand-alone value are both deferred until the go-live date and then recognized over the estimated life of the customer. Even if cash flow isn't a concern, a SaaS model could hurt your bottom line if you have an accounting need to accelerate your income.

Total Control of Data

In-house software application gives businesses a higher degree of control than hosted solutions where control resides with a third party. Typically everyone has to use the latest version of the software application and cannot defer upgrades or changes in the features. This can at times result in a loss of certain data or features that a client relies on as a core function.

Lack of Ongoing Support

Software owners must be able to provide a much higher level of ongoing customer service after the initial onboarding period. The SaaS meaning itself requires ongoing service. Otherwise, customers won't feel that they're getting anything for the ongoing fee and will seek to change providers or look for a one-time purchase option.

Similarly, software owners must be willing and able to constantly track metrics to adapt the product to their customers' needs. With increased flexibility a key benefit of using SaaS, customers will quickly be on the hunt for a new provider if their needs aren't met — at times, this will even be if you haven't heard any complaints from them.

What is the Process to Start Building a SaaS Product?

Building a SaaS product is similar to building any other product or making other business decisions. Of course, you will need to tailor the process to the business model.

Determine if It's Right For You

Review the above advantages along with the scenarios where a SaaS model may not work. Once you've convinced yourself, be sure to get total buy-in from your team. This includes not just getting them to agree with you but having them raise any concerns for discussion with the entire team so that you can move forward without any lingering doubts.

Design a Roadmap and Strategy for the Switch

There are three key questions that you should ask yourself when you begin to build a SaaS product. First, how will the new model create a better customer experience? Second, how do you measure success by changing to SaaS focused metrics that emphasize retention and longevity over the one-time sale? Third, how do your product requirements change to answer these questions and help you to gain market share?

Design and Development

Once you've established an overall strategy, you should design a product that meets these needs. Remember that it will likely need to have broad enough appeal to attract a number of clients while also having enough customizability that clients consider it the best option for their needs.

After you're satisfied with the design, begin the development process but don't be afraid to make changes if you encounter new challenges or come up with new ideas once you're putting your plan into action.

Beta Testing

No software is complete without beta testing. You simply can't anticipate every customer need and usage pattern.

You will typically start with a small group of customers at a free or greatly reduced price as compensation for their feedback and time spent dealing with any unexpected problems. Once you are satisfied with the results, you can bring in new customers at your planned price point.

Customer Onboarding

Customer onboarding is a critical moment in the SaaS business cycle. A customer's early experiences will determine if they will stick with your product or begin shopping for a new option.

Onboarding includes both initial training and ongoing support. It should last until clients are fully comfortable with your product and have had a chance to use all of the features that they plan to use.

Continue to Deliver Value and Market

Even though SaaS provides recurring revenue, it's important to remember that that revenue isn't guaranteed. You must continue to deliver value to your clients on an ongoing basis even and especially when their needs change in order to keep them as subscribers.

Since some clients will still inevitably move on to new business areas or have changing needs that you can't reasonably meet, you should also continue marketing to keep an incoming pipeline of clients.



How Much Does it Cost to Build a SaaS Product?

Building a SaaS product is a never-ending cycle of design, strategy, development, and back to design based on client feedback and new needs. The only certainty is uncertainty as to how much of the cycle you need to repeat in incremental development based on customer demand as well as your own internal testing for bugs and other problems.

Direct Development Costs

Long-term development costs typically run €10-20k per month. You can obtain a minimum viable product after just two to three months, but you should not consider this a finished product. After release, you should obtain feedback and continue to make improvements until you're satisfied with the finished product.

If you stick with the minimum viable product, it will be difficult to carve out sustainable market share in all but the least competitive markets, as competitors will find it easy to create a better alternative to your product. Keep this in mind when you're choosing how to emphasize costs, time, and scope when deciding on your budget.

Indirect Costs

Customers also need to consider secondary costs such as hosting and support. Hosting is relatively inconsequential in today's market but should still receive a line item in the project budget. Support may be handled by the customer's existing staff, but their time should be allocated to the project in order to properly track total costs and set profitable pricing.

Finally, it may not be in your or your client's best interest to truly handle every single feature. For example, you may want to integrate with existing user authentication, billing, account management, or communication services. These ongoing costs will also have to be built into your pricing model.

Marketing Costs

Client acquisition cost is one of the biggest expenses in software sales. You can reduce marketing costs to some degree by establishing yourself as the best in your niche so that customers don't leave for competitors. However, since you will likely have other competitors claiming to be the best, you will need to invest in convincing customers to choose you if you wish to expand your customer base.

Consider the [example of Monday](#), a time tracking and project management app that has raised €74.1 million in total investor contributions. It didn't raise this funding because it costs anywhere near that much to make a viable time tracking and project management tool. Instead, it spent what it needed to to create the best possible tool and then put the bulk of its funding into marketing.

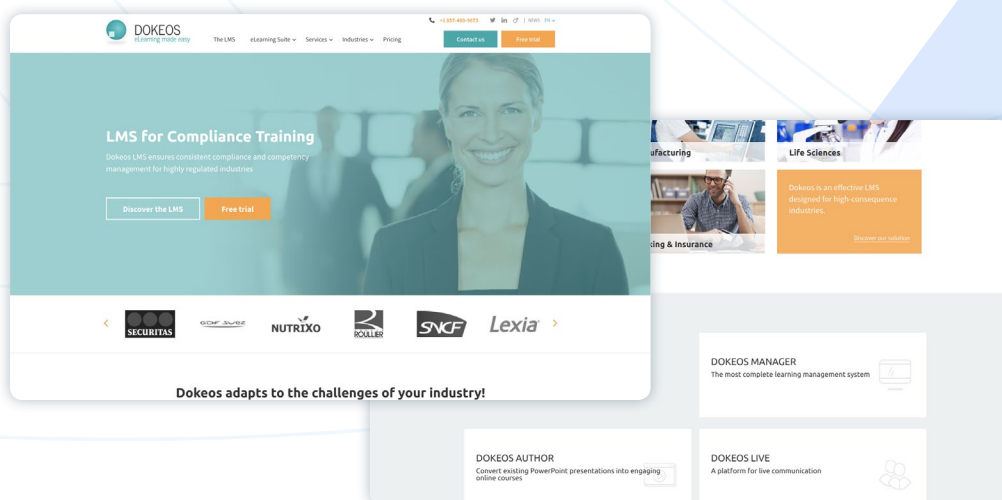
The vast majority of small to mid-sized businesses don't need to invest millions in marketing to achieve profitability, but you will need a plan to make customers aware of your product and what it can do for them.

SaaS Examples

There are many examples of successful SaaS companies that are well known even to less technically inclined people. One of our own SaaS examples is Dokeos, a leading provider of e-learning solutions tailored for businesses.

[Dokeos reduced their hosting costs by 75% after switching to a software as a service model.](#) Other key advantages they gained included the following.

- Simplifying their offering to provide for long-term scalability
- Improving the user experience
- Differentiating with smart reporting that capitalized on industry experience
- Reducing other maintenance and support costs.



Conclusion

SaaS models aren't gaining popularity — to the degree that few people ask what is SaaS — because of smart marketing or because developers are forcing them on clients.

They're gaining the majority of market share of business applications because they provide substantial benefits for both the developers and users.

While switching to a SaaS model may require an overhaul of your business strategy, there is a strong likelihood that the switch will vastly increase your odds of long-term success.

Want to see how we can help your startup, too?

Begin with a Scoping Workshop or
talk to us about your needs today.

Build Your Product Roadmap in 1 Day

Contact Us

Place de l'Universite 25
1348 Louvain-la-Neuve
Belgium

belighted