Introduction to Deployment with Render

Learn how to get set up and started with Render in order to deploy your applications!

Introduction

For most modern software engineering teams, deployment plays a key role in ensuring software can eventually be accessed by an end-user. The deployment process can be complex and, at times, involves the collaboration of multiple engineering team members. It requires careful planning and execution to ensure the software is deployed smoothly. In recent years, cloud-based deployment solutions have become a popular option for helping developers quickly deploy applications without requiring extensive infrastructure or technical expertise.

In this article, we will:

- Examine one type of cloud-based deployment solution called a PaaS
- Explore a popular PaaS provider called Render
- Break down how Render handles deployment
- Setup a Render account
- Explore the plan options that Render offers

Let's get started!

Platform as a Service (PaaS)

One popular type of cloud-based deployment solution is called a Platform as a Service (PaaS). A **PaaS** is an all-in-one platform for building, deploying, and managing applications over the internet. A PaaS often uses a set of assumptions about the things most software teams need as a way of simplifying the complex task of setting up infrastructure. This allows developers to no longer have to focus on setting up and managing resources and infrastructure on their own. Most PaaS providers offer an easy-to-use user interface that lets developers tweak the setup to meet their application's needs. They typically charge a per-usage fee to utilize their infrastructure, but some offer free, resource-limited tiers.

Other benefits of using a PaaS provider include the following:

- The PaaS provider handles the building and running of the developer's code

- Some PaaS providers offer additional resources, such as databases, for the developer to integrate and use within the project
- The PaaS provider handles the regular upgrades and maintenance of the infrastructure components
- The PaaS provider may handle some security aspects of the infrastructure
- The PaaS provider may provide options for easily scaling resources, either manually or automatically, to accommodate a growing number of users that are using the application

While there are plenty of PaaS providers, in this article, we will focus on exploring a PaaS called Render.

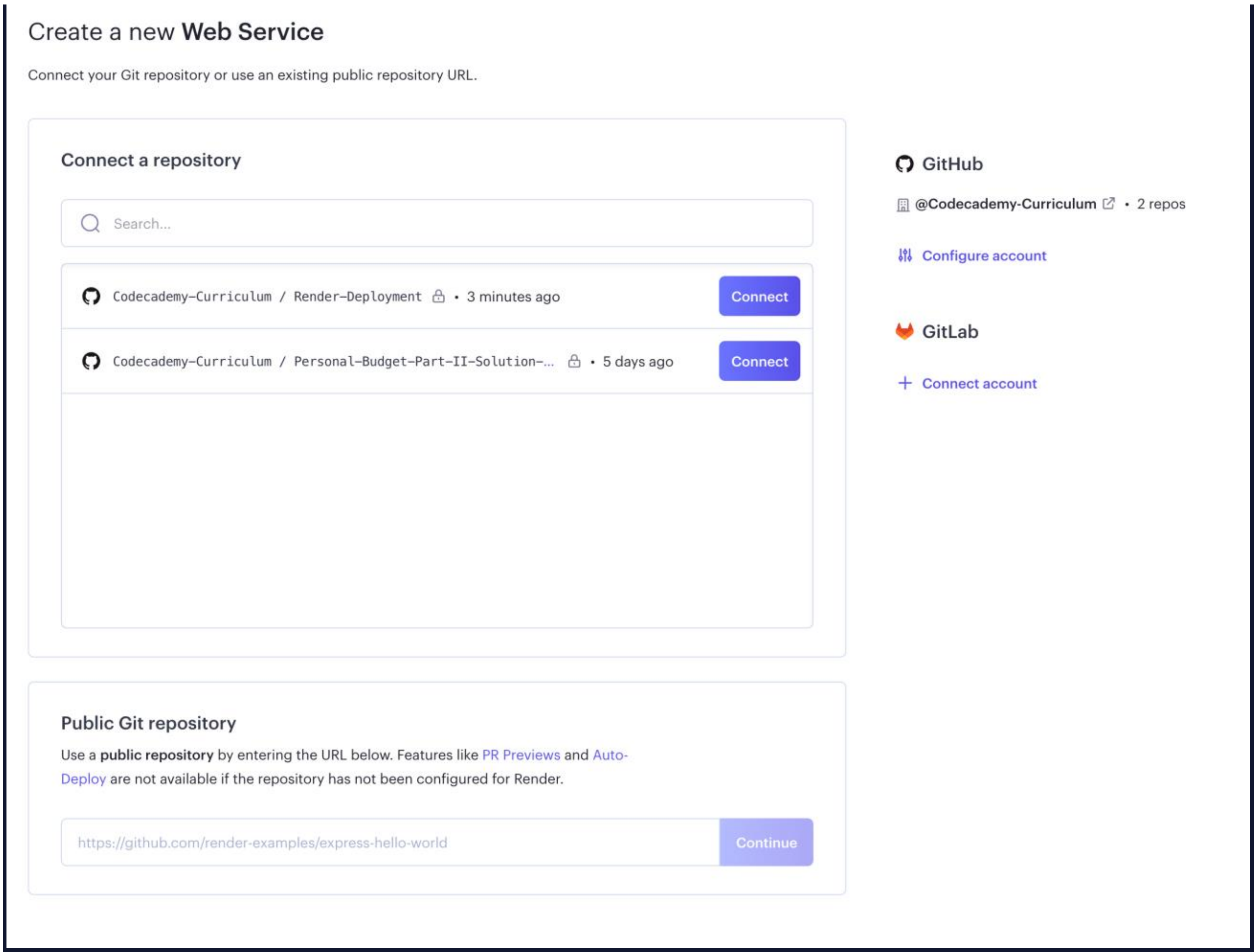
Introduction to Render

Render is a popular PaaS product that handles the building and deployment of code and provides the resources necessary to host various applications and services. By using Render for deployment, we can quickly deploy a running prototype of an application to potential users. Render supports several different programming languages, including Python, Ruby, and Javascript. Render also offers other features such as managed databases, static site hosting, and integration with popular developer tools like GitHub and Slack.

To dive into the basics of how Render handles deployment, let’s imagine we have a simple, completed e-commerce application. Let’s assume two things:

We have finished developing our application, and all of our code is hosted within a GitHub repository. Our application is a full-stack application with both a front-end and a back-end. Our back-end component uses Express.js and is powered by a PostgreSQL (PSQL) database. Our front-end component is simple HTML and CSS files, as well as some static assets like images.

For us to use Render as our deployment solution for our full-stack application, we will need to connect Render to the GitHub repository. The dashboard to connect a repository will look similar to this:



Create a new **Web Service**

Connect your Git repository or use an existing public repository URL.

Connect a repository

Search...

Codecademy-Curriculum / Render-Deployment • 3 minutes ago — Connect

Codecademy-Curriculum / Personal-Budget-Part-II-Solution-... • 5 days ago — Connect

Public Git repository

Use a **public repository** by entering the URL below. Features like PR Previews and Auto-Deploy are not available if the repository has not been configured for Render.

https://github.com/render-examples/express-hello-world — Continue

GitHub

@Codecademy-Curriculum • 2 repos

Configure account

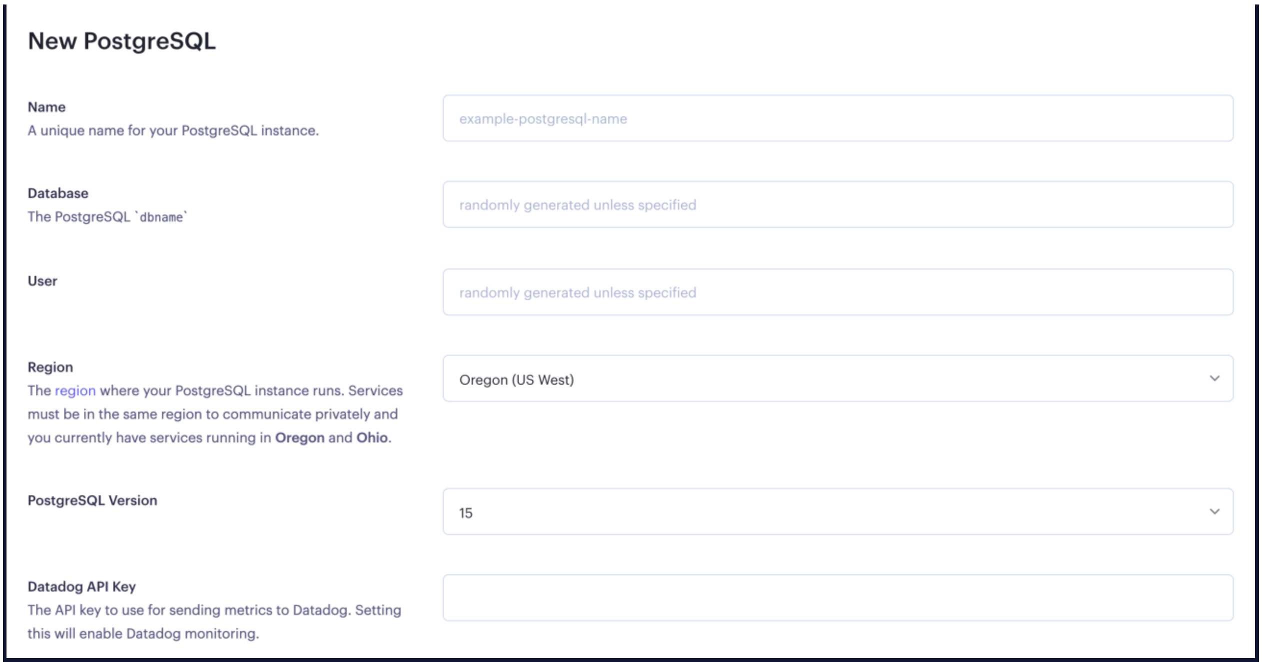
GitLab

+ Connect account

Notice in the dashboard above, a GitHub account ("Codecademy-Curriculum") is connected on the right-hand side. We can then select the specific repository we want to connect.

Connecting Render to a repository provides Render the access required to deploy our application's code but does not automatically handle hosting and connecting our PSQL database server. Fortunately, Render provides a cloud-hosted PSQL database service that can be used with our application. We can create a cloud database and have several options for accessing the database from an application's server-side code. Render also provides steps for taking a backup of existing database data and importing it into the newly created, Render-hosted database so that data can be easily transferred.

We can create a new PostgreSQL service via the Render dashboard. The setup will look similar to this:



New PostgreSQL

Name
A unique name for your PostgreSQL instance.
example-postgresql-name

Database
The PostgreSQL `dbname`
randomly generated unless specified

User
randomly generated unless specified

Region
The region where your PostgreSQL instance runs. Services must be in the same region to communicate privately and you currently have services running in **Oregon** and **Ohio**.
Oregon (US West)

PostgreSQL Version
15

Datadog API Key
The API key to use for sending metrics to Datadog. Setting this will enable Datadog monitoring.

Once the application code and database are connected, the final step is to ensure that our deployed application can be accessed by users over the Internet. Any application that is deployed via Render is provided a free publicly available URL link that resembles `<your-web-service-name>.onrender.com`. We can also customize the domain with a custom name at a later point!

Visually, the architecture of the setup resembles this:

To summarize:

- Our application code lives in the GitHub repository and is connected to Render
- Render handles setting up the application's infrastructure and deploying the code.
- The application code connects to the Render-hosted PostgreSQL database using the connection information provided by Render upon database setup.
- When the application is ready, it is made publicly available via a `.onrender.com` domain

Note that, once connected, any commits made to the repository will result in Render automatically deploying the application. This setting is called "Auto Deploy." It can be turned off if we do not want Render to deploy our application with each new commit automatically.

Now, let's take the first step of working with Render (and eventually setting up our own applications), by signing up for an account. Throughout the process, we will explore more about the features and services that Render offers!

Before moving on, let's review:

Free response

In your own words, describe some advantages of using a PaaS service like Render to deploy an application.

Your response

Using a PaaS service, allows developers to no longer have to focus on setting up and managing resources and infrastructure on their own. Most

The PaaS provider handles the building and running of the developer's code saving time for the developers.

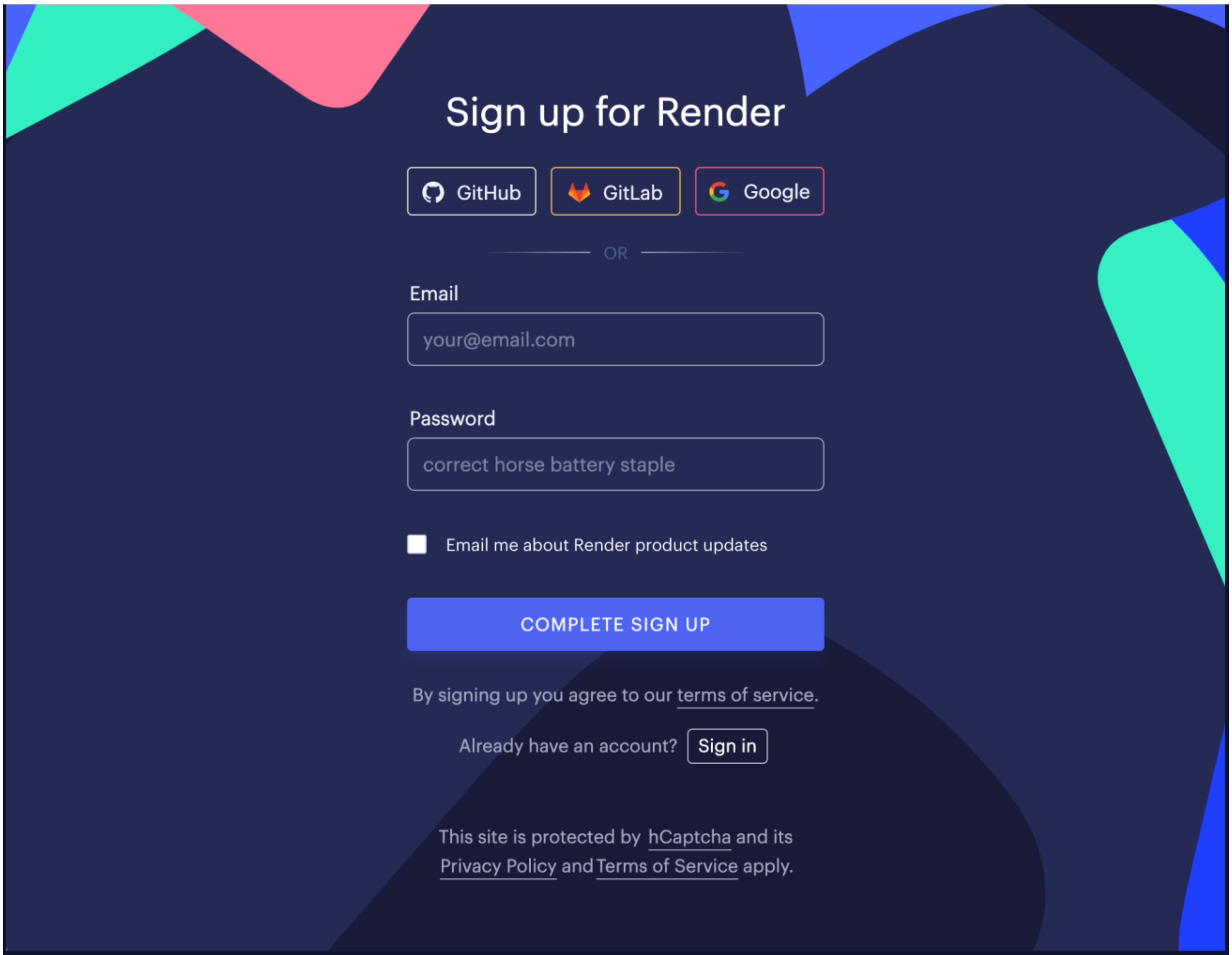
The PaaS provider handles the regular upgrades and maintenance of the infrastructure components.

Our answer

A PaaS deployment solution like Render removes the need to host and manage the infrastructure needed to deploy an application. The application will be automatically built and deployed upon any new commits to your code repository, making the deployment process more automated and quicker. This in turn will get software changes to the end users much faster. PaaS deployment solutions can also provide regular maintenance and upgrades of the infrastructure resources and a way to scale those resources when the number of users on the application increase.

Getting Started with Render

Let's create a Render account so that we can start deploying our own applications! First, navigate to the Render sign-up page. Render allows new users to sign up with a GitHub, GitLab, or Google account or by using an email and creating a password. By default, when we sign up, we start with a free Individual plan (more on plans in a bit).



Sign up for Render

Email

your@email.com

Password

correct horse battery staple

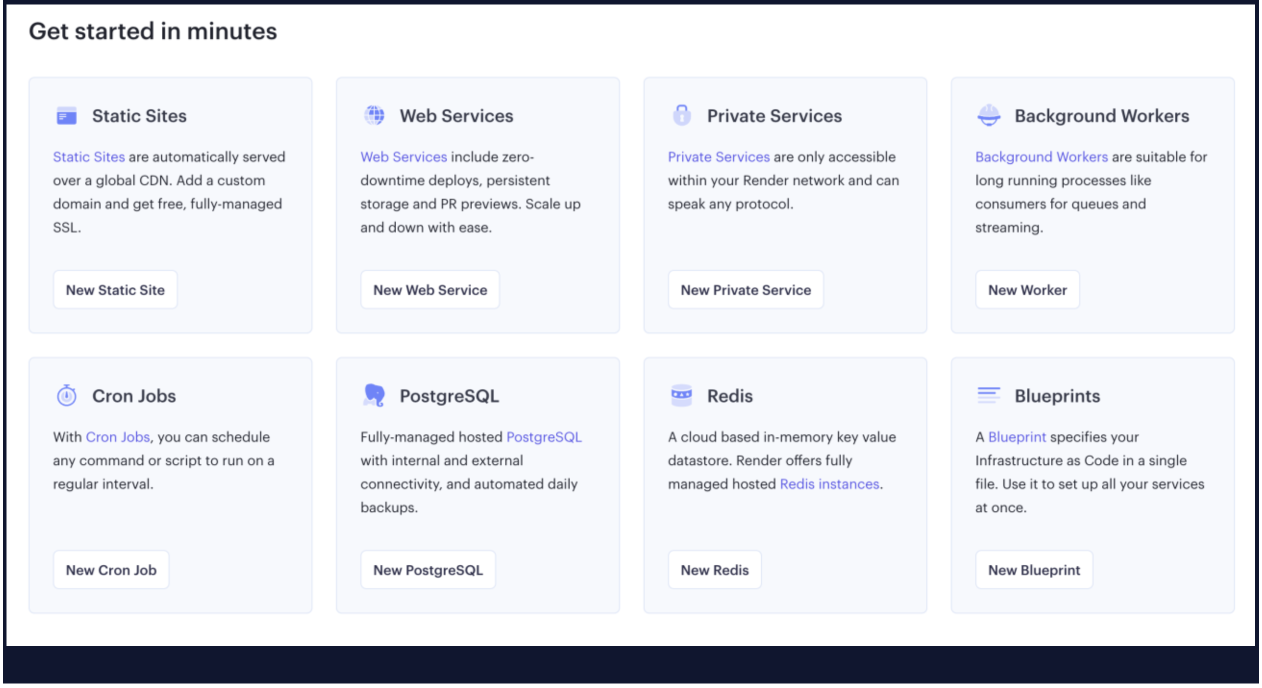
□ Email me about Render product updates

By signing up you agree to our terms of service.

Already have an account? Sign in

This site is protected by hCaptcha and its Privacy Policy and Terms of Service apply.

Once logged in, since we don’t have any services set-up already, Render will present a dashboard listing all the application and service types that can be deployed. It will look similar to this:



Get started in minutes

Static Sites

Static Sites are automatically served over a global CDN. Add a custom domain and get free, fully-managed SSL.

Web Services

Web Services include zero-downtime deploys, persistent storage and PR previews. Scale up and down with ease.

Private Services

Private Services are only accessible within your Render network and can speak any protocol.

Background Workers

Background Workers are suitable for long running processes like consumers for queues and streaming.

Cron Jobs

With Cron Jobs, you can schedule any command or script to run on a regular interval.

PostgreSQL

Fully-managed hosted PostgreSQL with internal and external connectivity, and automated daily backups.

Redis

A cloud based in-memory key value datastore. Render offers fully managed hosted Redis instances.

Blueprints

A Blueprint specifies your Infrastructure as Code in a single file. Use it to set up all your services at once.

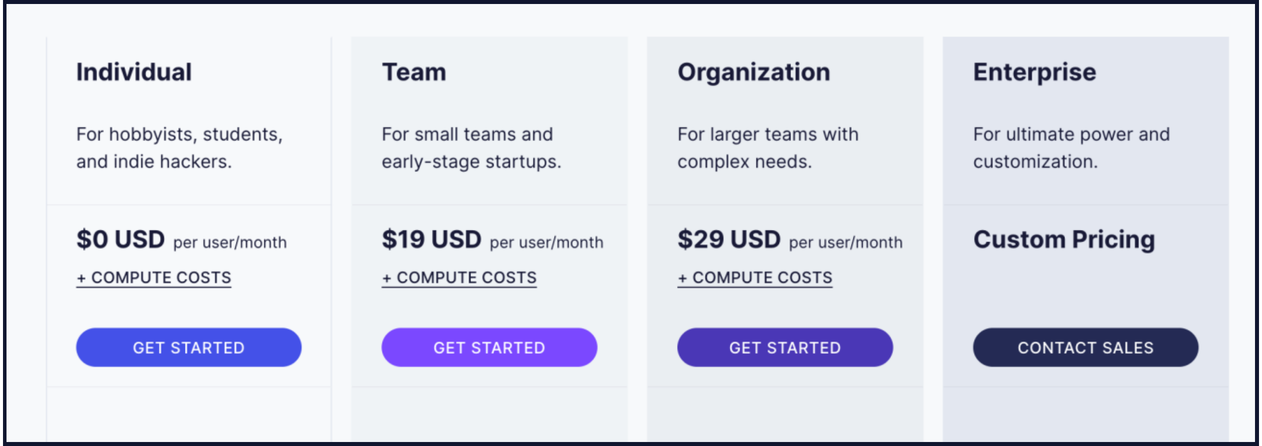
Notice two services that would come in handy for our example application from earlier. The first is called "Web Services". This service allows us to deploy web applications using multiple frameworks and languages. Render even provides quick templates for different frameworks to get an application running quickly. The second is the PostgreSQL service. This service lets us set up a cloud PSQL database that is managed by Render.

Additionally, while viewing the Render dashboard, notice that some of the services, like the Private Services option, require a paid plan subscription. A paid service will require payment information to be entered before it can be configured and created.

Let's briefly cover the plan options that Render offers.

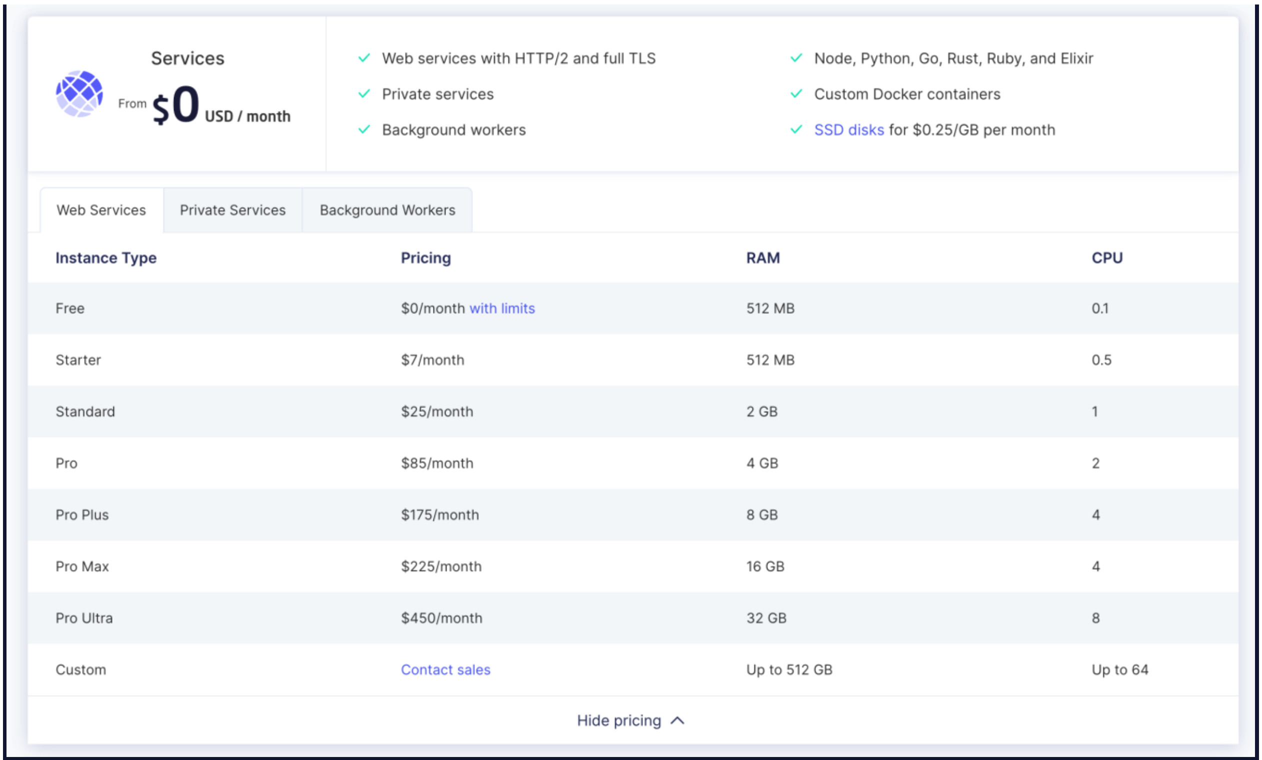
Render Plans and Services

Render offers several plans to meet the needs of an individual or team. Any plan other than "Individual" offers the ability to add multiple team members to the account and additional features such as automatic resource scaling, more application bandwidth, and even additional support from the Render team.



Individual	Team	Organization	Enterprise
For hobbyists, students, and indie hackers.	For small teams and early-stage startups.	For larger teams with complex needs.	For ultimate power and customization.
$0 USD per user/month + COMPUTE COSTS	**$19 USD** per user/month + COMPUTE COSTS	**$29 USD** per user/month + COMPUTE COSTS	**Custom Pricing**
GET STARTED	GET STARTED	GET STARTED	CONTACT SALES

It is important to note that plans are separate from the individual instances of computing power used by an account. For each service we choose to use, we can choose the "Instance Type" we think is the best fit for the service. Higher-tier instance types provide increased resource (e.g., RAM, CPU) allocation. Here is the computing cost chart for a "Web Service":



Services

From $0 USD / month

✓ Web services with HTTP/2 and full TLS
✓ Private services
✓ Background workers
✓ Node, Python, Go, Rust, Ruby, and Elixir
✓ Custom Docker containers
✓ SSD disks for $0.25/GB per month

Instance Type	Pricing	RAM	CPU
Free	$0/month with limits	512 MB	0.1
Starter	$7/month	512 MB	0.5
Standard	$25/month	2 GB	1
Pro	$85/month	4 GB	2
Pro Plus	$175/month	8 GB	4
Pro Max	$225/month	16 GB	4
Pro Ultra	$450/month	32 GB	8
Custom	Contact sales	Up to 512 GB	Up to 64

For example, if we wanted to upgrade our compute plan for an application to the “Starter” instance type tier (from “Free”), we would gain additional CPU resources, which provides benefits such as faster build times and unlimited usage hours. However, no matter the account type we use, we are always given access to a “Free” instance type which, regardless of the limitations, can be useful for experimentation.

For both the Web Service and the Postgres Service, here are the limitations to be aware of for a “Free” instance:

Render Service	Free Instance Limitations
Web Services (e.g., Express.js)	• Service is spun down after 15 minutes of ina • Cannot scale manually or automatically • Slower builds
PostgreSQL database	• Database expires 90 days after creation • Only one free database active at a time • Limit to 1GB of storage • Limit to 256MB of RAM • Not backed up

Full details of all of Render’s plan and compute tiers, as well as the features offered within each, can be found here.

Wrap Up

In this article, we familiarized ourselves with the basics of what deployment is and how PaaS products can help simplify the process. We also took a look at what Render, which is an example of a PaaS product, offers. Let’s recap what was covered:

- A Platform as a Service (PaaS) is one type of cloud-based deployment solution that handles the hardware, software, and networking resources for a project to get the project up and running for users
- Render is a type of PaaS provider that offers many advantages for deploying applications and services
- Render connects to a code repository in order to build and deploy the application code
- A Render account can be easily set up using an email and password or through linking an existing GitHub, GitLab, or Google account
- Render offers several tiers of paid subscription plans, but also has a free, Individual Plan geared towards hobbyists, students, and learners
- Several features and limitations come with the free, Individual Plan and free instance types you can use within Render

We have only covered the most basic features and capabilities available during the process of deployment via Render. To explore more Render features, take a look over the documentation, explore the Render community (a great place for help with troubleshooting), and read more about Render’s free tier.