

# Introduction to RStudio and Shiny servers



## Workshop Setup:

- ▶ Wi-Fi


Network Name:

Password:

- ▶ Requirements

An active Gmail account

# What is Google Cloud Platform?

Google Cloud Platform – known as GCP  - is a collection of cloud computing services that use the resources available at Google. GCP offers services **via the cloud** that access Google's physical hardware infrastructure such as: computers, hard disk drives, solid state drives and networking. This is a **fast** and **cost effective** alternative to having to build and maintain your own physical infrastructure.



Other popular services:




- Microsoft Azure Cloud Computing Platform & Services
- Amazon Web Services (AWS)

# Topics

## ► Workshop aim:

Learn how to setup RStudio and Shiny servers on GCP and host a shiny app online.

## ► Topics:

- Setup GCP Virtual Machine (VM) instance 
- Setup RStudio server  R Studio®
- Setup Shiny server 
- Host a shiny app online

# Setup GCP VM instance (for free)



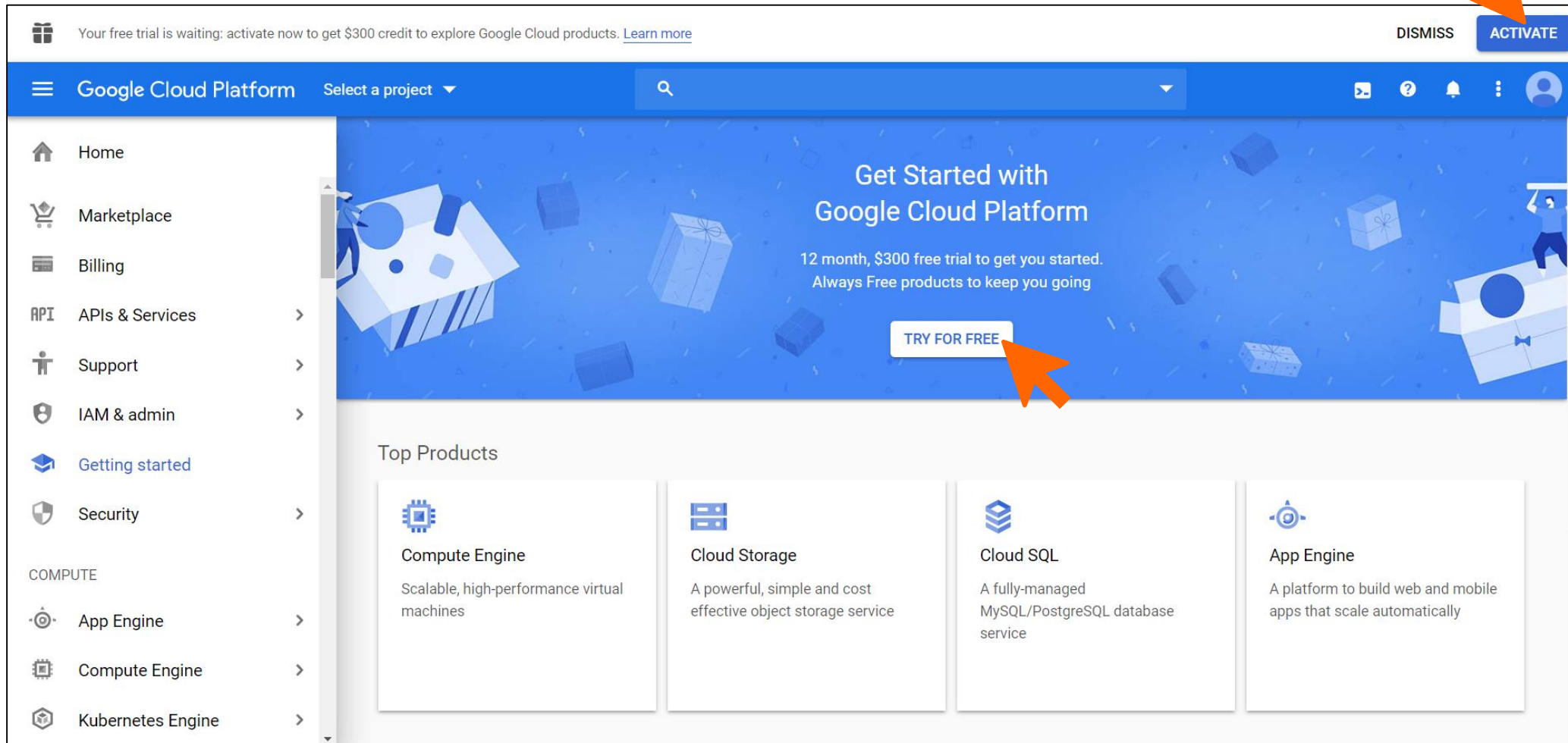
In order to access the Google Cloud Platform you will need to do the following:

1. Create a Gmail account (if you don't have one already 🤖 )
2. Visit <https://console.cloud.google.com/> (you might need to sign in)
3. Deal with the boring stuff (Terms of Service)
4. Get \$300 free trial (for 12 months) 🎉
5. Get started!



You might need to provide your card details – don't worry you are able to close your billing account if you want.

# Live Demo Part 1



The screenshot shows the Google Cloud Platform console interface. At the top, a banner reads "Your free trial is waiting: activate now to get \$300 credit to explore Google Cloud products. [Learn more](#)". In the top right corner, there are "DISMISS" and "ACTIVATE" buttons, with an orange arrow pointing to the "ACTIVATE" button. The main navigation bar includes "Google Cloud Platform", "Select a project", a search bar, and user icons. A left sidebar lists navigation options: Home, Marketplace, Billing, APIs & Services, Support, IAM & admin, Getting started (highlighted), Security, and a COMPUTE section with App Engine, Compute Engine, and Kubernetes Engine. The main content area features a large blue banner titled "Get Started with Google Cloud Platform" with the text "12 month, \$300 free trial to get you started. Always Free products to keep you going" and a "TRY FOR FREE" button, which is pointed to by an orange arrow. Below the banner, the "Top Products" section displays four cards: Compute Engine (Scalable, high-performance virtual machines), Cloud Storage (A powerful, simple and cost effective object storage service), Cloud SQL (A fully-managed MySQL/PostgreSQL database service), and App Engine (A platform to build web and mobile apps that scale automatically).

Try Google Cloud Platform for free

## Step 1 of 2

**Country**

Spain

**Terms of Service**

☐ I agree to the [Google Cloud Platform Terms of Service](#), and the terms of service of [any applicable services and APIs](#). I have also read and agree to the [Google Cloud Platform Free Trial Terms of Service](#).

Required to continue

**Email updates**

☐ I would like to receive periodic emails on news, product updates and special offers from Google Cloud and Google Cloud Partners.

CONTINUE

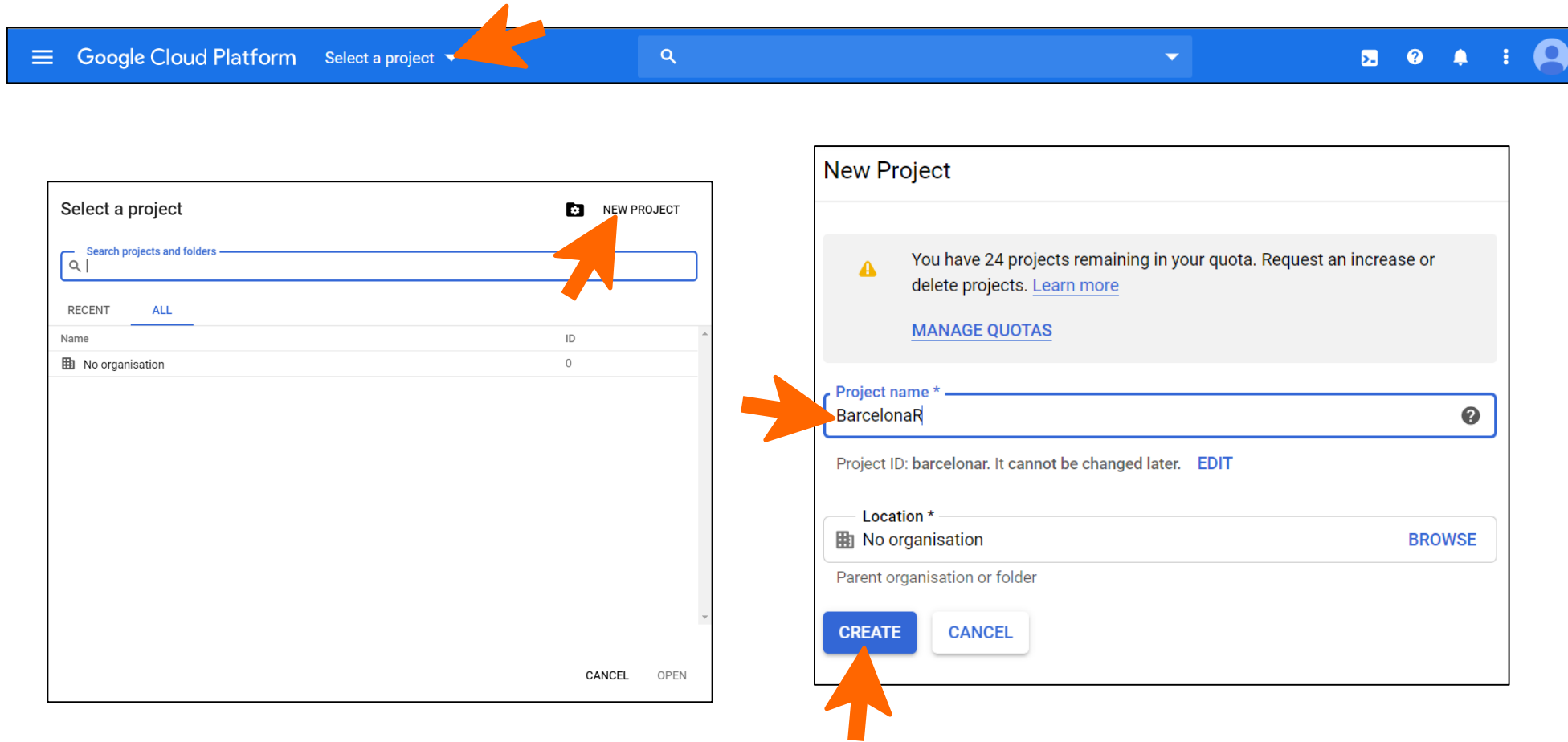
- Welcome to Google Cloud Platform**

Get everything you need to build and run your apps, websites and services, including Firebase and the Google Maps API.
- Free \$300 credit for you**

Sign up and get \$300 to spend on Google Cloud Platform over the next 12 months.
- We're always transparent**

We ask you for your credit card to make sure you are not a robot. You won't be charged unless you manually upgrade to a paid account.

# Create a project



Google Cloud Platform Select a project

Search projects and folders

RECENT ALL

Name	ID
No organisation	0

CANCEL OPEN

NEW PROJECT

### New Project

You have 24 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)

[MANAGE QUOTAS](#)

Project name \* BarcelonaR

Project ID: barcelonar. It cannot be changed later. [EDIT](#)

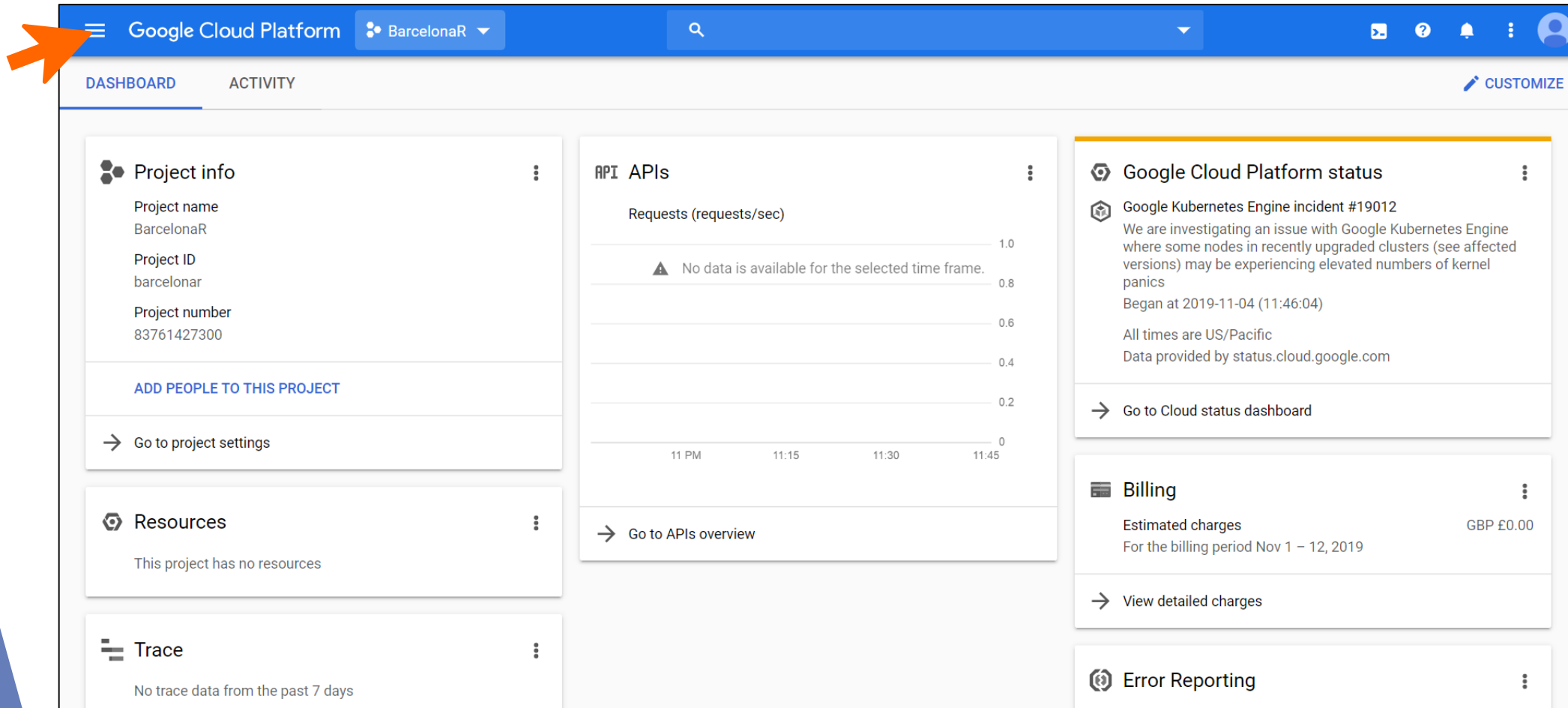
Location \* No organisation [BROWSE](#)

Parent organisation or folder

[CREATE](#) [CANCEL](#)

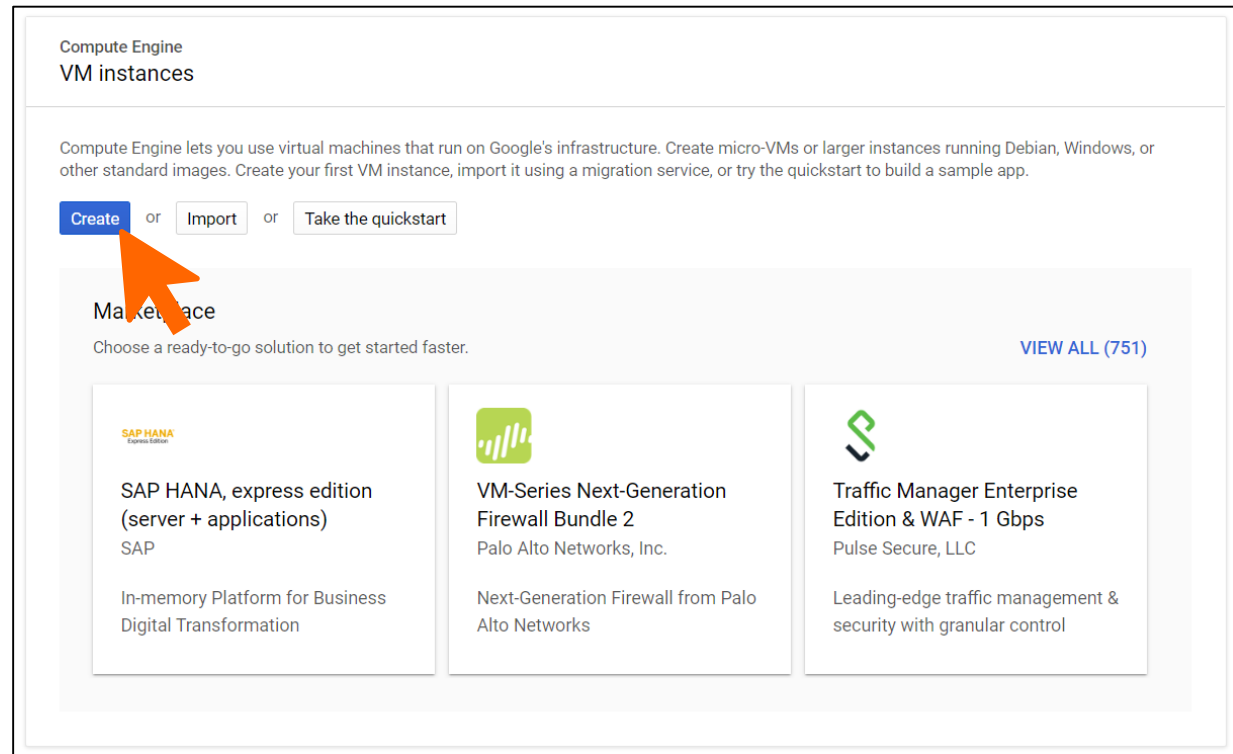
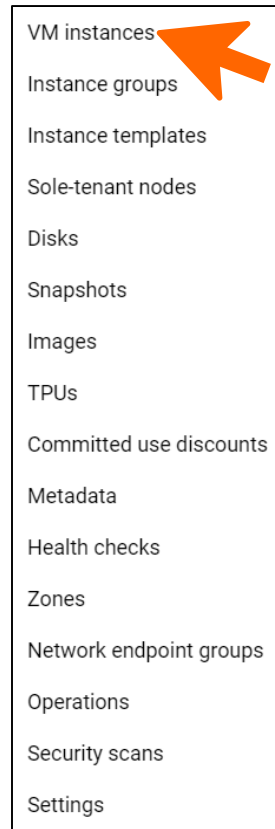
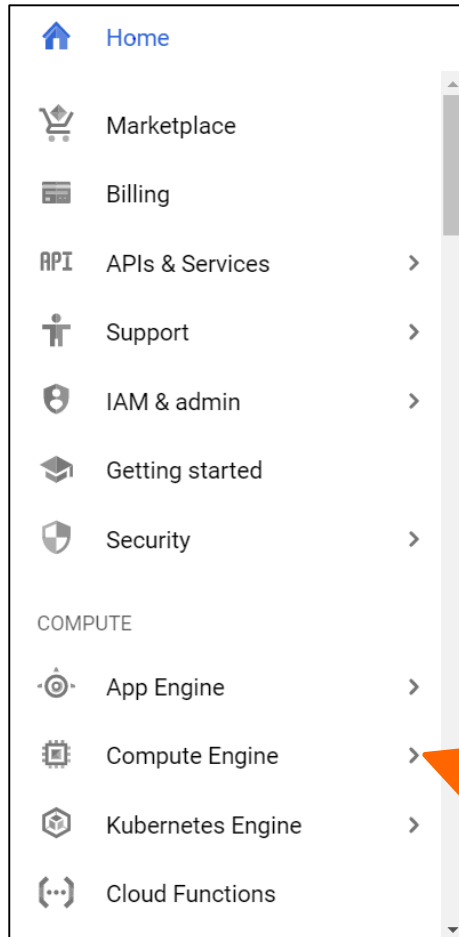


# Create a VM instance



The screenshot shows the Google Cloud Platform dashboard for the project 'BarcelonaR'. The top navigation bar includes the Google Cloud Platform logo, the project name 'BarcelonaR', a search bar, and user profile icons. The main content area is divided into several sections:

- Project info:** Displays project details such as Project name (BarcelonaR), Project ID (barcelonar), and Project number (83761427300). It includes a link to 'ADD PEOPLE TO THIS PROJECT' and a button to 'Go to project settings'.
- Resources:** Indicates that 'This project has no resources'.
- Trace:** Shows 'No trace data from the past 7 days'.
- API APIs:** A section for monitoring API requests, currently showing 'No data is available for the selected time frame.' with a graph area and a button to 'Go to APIs overview'.
- Google Cloud Platform status:** Provides information about the Google Cloud status, including a recent incident (#19012) related to Google Kubernetes Engine. It includes a link to 'Go to Cloud status dashboard'.
- Billing:** Shows 'Estimated charges' for the billing period Nov 1 – 12, 2019, with a total of GBP £0.00. It includes a link to 'View detailed charges'.
- Error Reporting:** A section for monitoring errors.



Use the pin frequently!



functionality to pin the services you use most

Name <sup>?</sup>  
instance-1


Region <sup>?</sup> us-central1 (Iowa) Zone <sup>?</sup> us-central1-a

Machine configuration <sup>?</sup>

Machine family  
General-purpose Memory-optimized  
Machine types for common workloads, optimized for cost and flexibility

Series  
N1  
Powered by Intel Skylake CPU platform or one of its predecessors


Machine type  
n1-standard-1 (1 vCPU, 3.75 GB memory)

	vCPU	Memory
	1	3.75 GB


✓ CPU platform and GPU

Container <sup>?</sup>  
☐ Deploy a container image to this VM instance. [Learn more](#)

Name you VM instance  
e.g. “rstudio-shiny-servers”

 You can select a different region/zone  
to specify the location the resource is  
used and where the data is stored.

Select the VM machine type  
e.g. “n1-standard-1” is OK

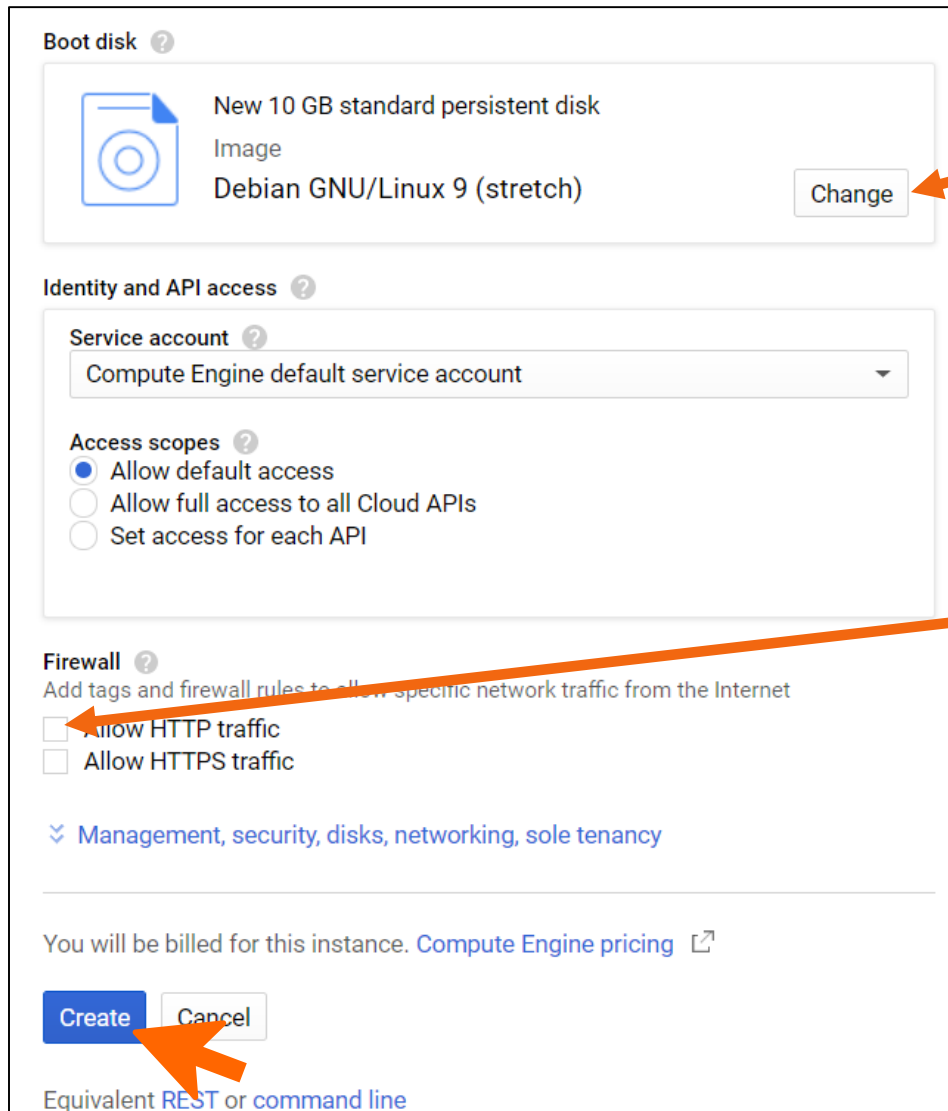
 The “stronger” the machine type (more  
CPUs and/or more memory) the more  
expensive it is to run!

\$24.67 monthly estimate

That's about \$0.034 hourly

Pay for what you use: No upfront costs and per second billing

[Details](#)



**Boot disk** ?

New 10 GB standard persistent disk  
Image  
Debian GNU/Linux 9 (stretch) Change

**Identity and API access** ?

**Service account** ?  
Compute Engine default service account

**Access scopes** ?

☒ Allow default access  
☐ Allow full access to all Cloud APIs  
☐ Set access for each API

**Firewall** ?  
Add tags and firewall rules to allow specific network traffic from the Internet

☒ Allow HTTP traffic  
☐ Allow HTTPS traffic

✓ Management, security, disks, networking, sole tenancy

You will be billed for this instance. [Compute Engine pricing](#)

Create Cancel

Equivalent [REST](#) or [command line](#)

Select the OS image.  
We will use "Ubuntu 16.04 LTS"



Other OS images may also work but  
you would need to adjust the  
installation procedure.

Check the "Allow HTTP traffic"  
box to allow incoming traffic.



We will later setup specific Firewall  
rules to allow incoming traffic to  
RStudio and Shiny servers

# VM instance is running!

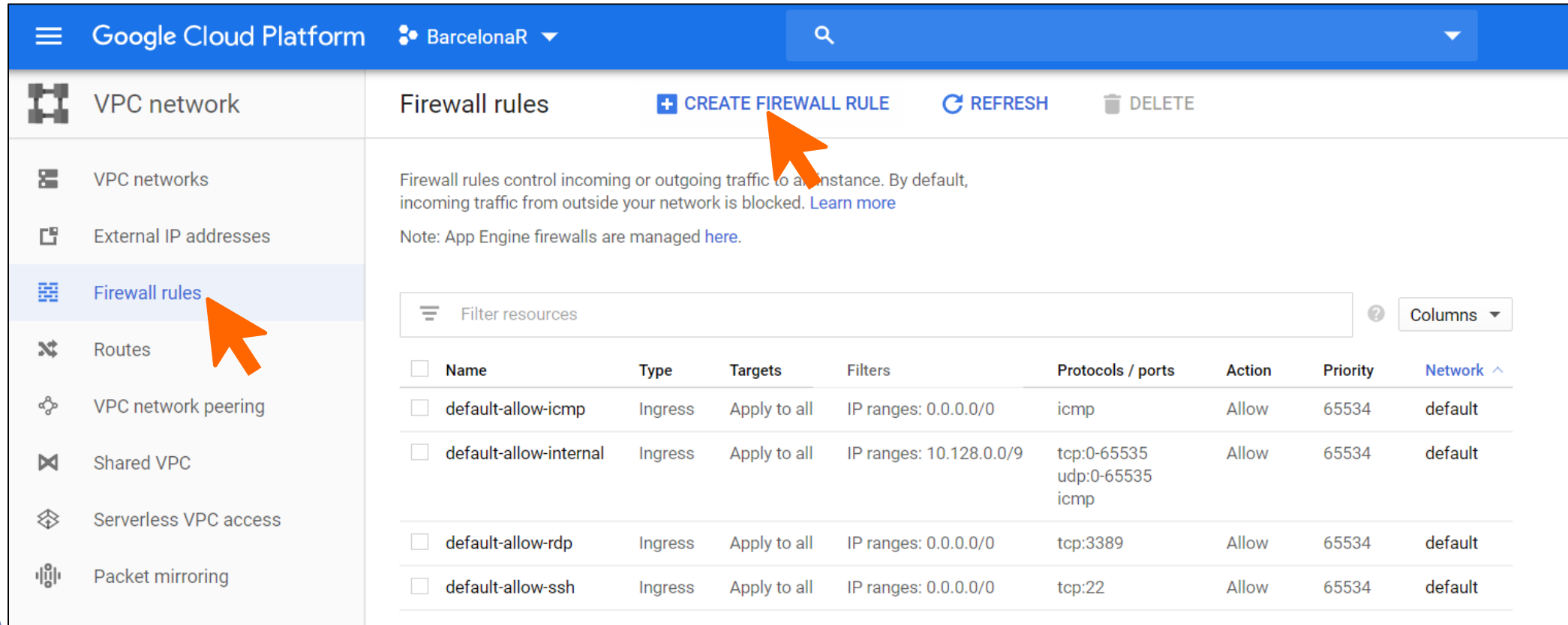


Filter VM instances							?	Columns ▾
<input type="checkbox"/> Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect		
<input checked="" type="checkbox"/>  rstudio-shiny-servers	us-central1-a			10.128.0.2 (nic0)	34.67.35.39 	SSH ▾		⋮



Make a note of this IP address...  
it will be important!

# Create a firewall rule – Go to VPC network



Google Cloud Platform BarcelonaR

VPC network

Firewall rules [+ CREATE FIREWALL RULE](#) [REFRESH](#) [DELETE](#)

Firewall rules control incoming or outgoing traffic to a VM instance. By default, incoming traffic from outside your network is blocked. [Learn more](#)

Note: App Engine firewalls are managed [here](#).

Filter resources

<input type="checkbox"/> Name	Type	Targets	Filters	Protocols / ports	Action	Priority	Network <a href="#">^</a>
<input type="checkbox"/> default-allow-icmp	Ingress	Apply to all	IP ranges: 0.0.0.0/0	icmp	Allow	65534	default
<input type="checkbox"/> default-allow-internal	Ingress	Apply to all	IP ranges: 10.128.0.0/9	tcp:0-65535 udp:0-65535 icmp	Allow	65534	default
<input type="checkbox"/> default-allow-rdp	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:3389	Allow	65534	default
<input type="checkbox"/> default-allow-ssh	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:22	Allow	65534	default



Configuring firewall rules to allow access via ports 8787 and 3838 means that you and others can access RStudio and Shiny servers from a web browser such as Chrome

"rstudio"

← Create a firewall rule

Firewall rules control incoming or outgoing traffic to an instance. By default, incoming traffic from outside your network is blocked. [Learn more](#)

Name <sup>?</sup>

Description (Optional)

Logs  
 Turning on firewall logs can generate a large number of logs which can increase costs in Stackdriver. [Learn more](#)  
☐ On  
☒ Off

Network <sup>?</sup>

Priority <sup>?</sup>  
 Priority can be 0 - 65535 [Check priority of other firewall rules](#)

Direction of traffic <sup>?</sup>  
☒ Ingress  
☐ Egress

Action on match <sup>?</sup>  
☒ Allow  
☐ Deny

Targets <sup>?</sup>

Service account scope <sup>?</sup>  
☒ In this project  
☐ In another project

Target service account

Source filter <sup>?</sup>

Source IP ranges <sup>?</sup>

Second source filter <sup>?</sup>

Protocols and ports <sup>?</sup>  
☐ Allow all  
☒ Specified protocols and ports

☒ tcp :   
☐ udp :   
☐ Other protocols

⌵ Disable rule

For RStudio server

“shiny”

← Create a firewall rule

Firewall rules control incoming or outgoing traffic to an instance. By default, incoming traffic from outside your network is blocked. [Learn more](#)

Name <sup>?</sup>

Description (Optional)

Logs  
 Turning on firewall logs can generate a large number of logs which can increase costs in Stackdriver. [Learn more](#)  
☐ On  
☒ Off

Network <sup>?</sup>

Priority <sup>?</sup>  
 Priority can be 0 - 65535 [Check priority of other firewall rules](#)

Direction of traffic <sup>?</sup>  
☒ Ingress  
☐ Egress

Action on match <sup>?</sup>  
☒ Allow  
☐ Deny

Targets <sup>?</sup>

Service account scope <sup>?</sup>  
☒ In this project  
☐ In another project

Target service account

Source filter <sup>?</sup>

Source IP ranges <sup>?</sup>

Second source filter <sup>?</sup>

Protocols and ports <sup>?</sup>  
☐ Allow all  
☒ Specified protocols and ports


☒ tcp :   
☐ udp :   
☐ Other protocols

[Disable rule](#)

[Create](#) [Cancel](#)

For Shiny server



 Filter resources ? Columns ▾



<input type="checkbox"/> Name	Type	Targets	Filters	Protocols / ports	Action	Priority	Network <sup>^</sup>
<input type="checkbox"/> default-allow-http	Ingress	http-server	IP ranges: 0.0.0.0/0	tcp:80	Allow	1000	default
<input type="checkbox"/> rstudio	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:8787	Allow	1000	default
<input type="checkbox"/> shiny	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:3838	Allow	1000	default
<input type="checkbox"/> default-allow-icmp	Ingress	Apply to all	IP ranges: 0.0.0.0/0	icmp	Allow	65534	default
<input type="checkbox"/> default-allow-internal	Ingress	Apply to all	IP ranges: 10.128.0.0/9	tcp:0-65535 udp:0-65535 icmp	Allow	65534	default
<input type="checkbox"/> default-allow-rdp	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:3389	Allow	65534	default
<input type="checkbox"/> default-allow-ssh	Ingress	Apply to all	IP ranges: 0.0.0.0/0	tcp:22	Allow	65534	default



# Setup RStudio server



In order to setup RStudio server you will need to do the following:

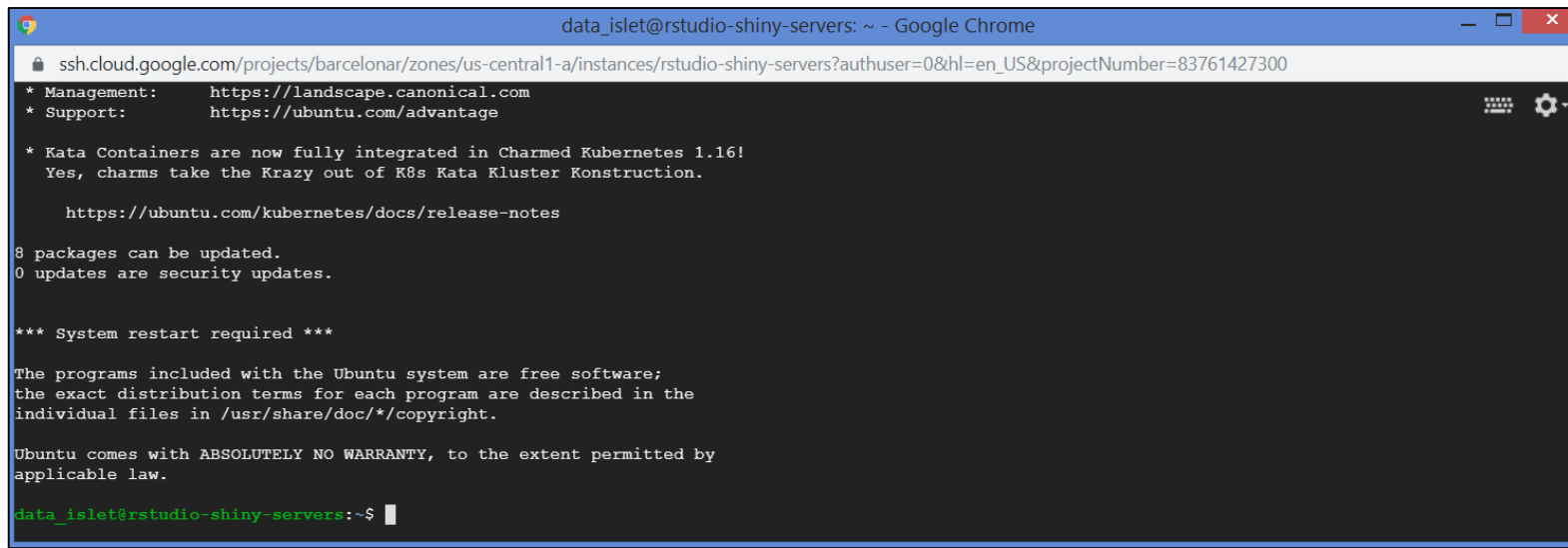
1. Connect to your VM instance (via Secure Shell - SSH)
2. Update and Upgrade stuff
3. Install 
4. Install R packages
5. Install Studio server
6. Create a user
7. Access RStudio server! 🎉



Remember you need to run commands as the “superuser” = **sudo**

## Live Demo Part 2 – Connect to VM

Filter VM instances							?	Columns ▾
<input type="checkbox"/> Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect		
<input type="checkbox"/>  rstudio-shiny-servers	us-central1-a			10.128.0.2 (nic0)	34.67.35.39 ↗	SSH ▾		⋮



```
data_islet@rstudio-shiny-servers: ~ - Google Chrome
ssh.cloud.google.com/projects/barcelonar/zones/us-central1-a/instances/rstudio-shiny-servers?authuser=0&hl=en_US&projectNumber=83761427300
* Management: https://landscape.canonical.com
* Support:    https://ubuntu.com/advantage

* Kata Containers are now fully integrated in Charmed Kubernetes 1.16!
  Yes, charms take the Krazy out of K8s Kata Kluster Konstruktion.

https://ubuntu.com/kubernetes/docs/release-notes

8 packages can be updated.
0 updates are security updates.

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

data_islet@rstudio-shiny-servers:~$
```

# Update / Upgrade

```
data_islet@rstudio-shiny-servers:~$ sudo apt-get update
...

data_islet@rstudio-shiny-servers:~$ sudo apt-get upgrade
...

data_islet@rstudio-shiny-servers:~$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys
E298A3A825C0D65DFD57CBB651716619E084DAB9
...

data_islet@rstudio-shiny-servers:~$ sudo echo "deb https://cloud.r-project.org/bin/linux/ubuntu
xenial-cran35/" | sudo tee -a /etc/apt/sources.list
...

data_islet@rstudio-shiny-servers:~$ sudo apt-get update
...
```



You might need: `sudo apt-get install dirmngr`

# Install R / Install packages


```
data_islet@rstudio-shiny-servers:~$ sudo apt-get install r-base r-base-dev
...
Do you want to continue? [Y/n] Y

data_islet@rstudio-shiny-servers:~$ sudo apt-get install libcurl4-openssl-dev libssl-dev libxml2-dev
...
Do you want to continue? [Y/n] Y
...

data_islet@rstudio-shiny-servers:~$ sudo R
...

data_islet@rstudio-shiny-servers:~$ install.packages(c('shiny', 'rmarkdown', 'dplyr'))
...

q("no")
```



Time for a break and let it run!

# Install RStudio server & add a user

```
data_islet@rstudio-shiny-servers:~$ sudo gpg --keyserver keys.gnupg.net --recv-keys
3F32EE77E331692F
...

data_islet@rstudio-shiny-servers:~$ sudo apt-get install gdebi-core
...

data_islet@rstudio-shiny-servers:~$ wget
https://download2.rstudio.org/server/trusty/amd64/rstudio-server-1.2.5019-amd64.deb
...

data_islet@rstudio-shiny-servers:~$ sudo gdebi rstudio-server-1.2.5019-amd64.deb
...
Do you want to install the software package? [y/N]: Y
...

data_islet@rstudio-shiny-servers:~$ sudo adduser <username>
Enter new UNIX password:
Retype new UNIX password:
...
Is the information correct? [Y/n] Y
```



Choose your <username>

# RStudio server is running!

In a web browser navigate to the following address: <http://<External IP>:8787>

Where <External IP> is found

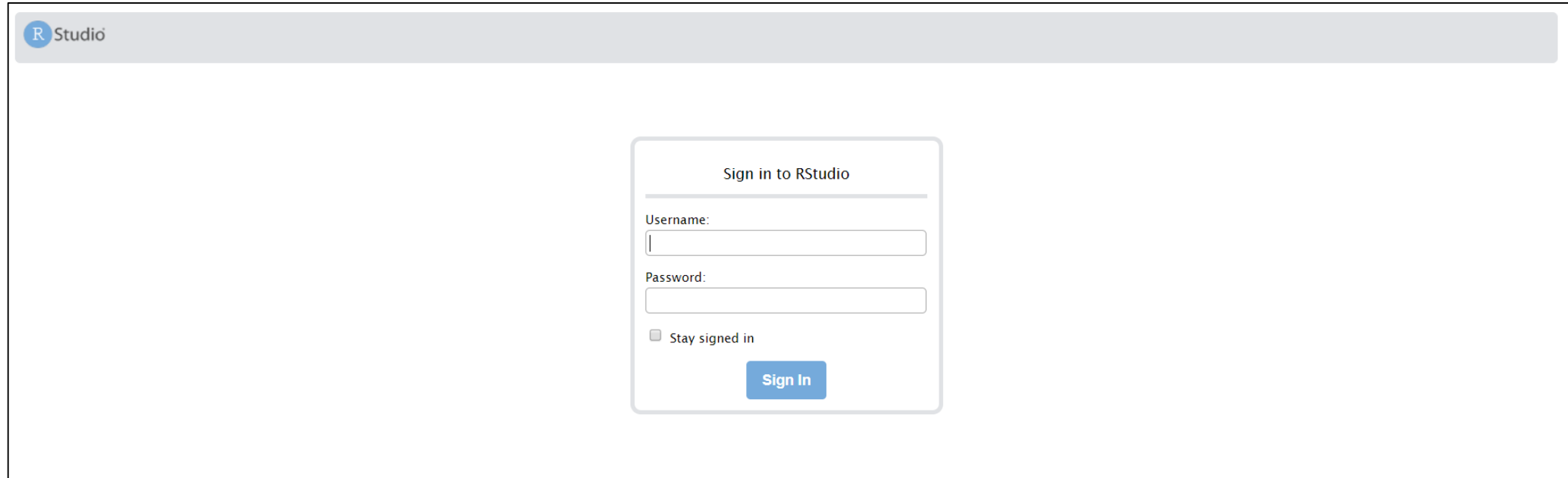
...


Filter VM instances

?

Columns

<input type="checkbox"/> Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>  rstudio-shiny-servers	us-central1-a			10.128.0.2 (nic0)	34.67.35.39 	SSH <div></div> <div></div>



 Studio

Sign in to RStudio

Username:

Password:

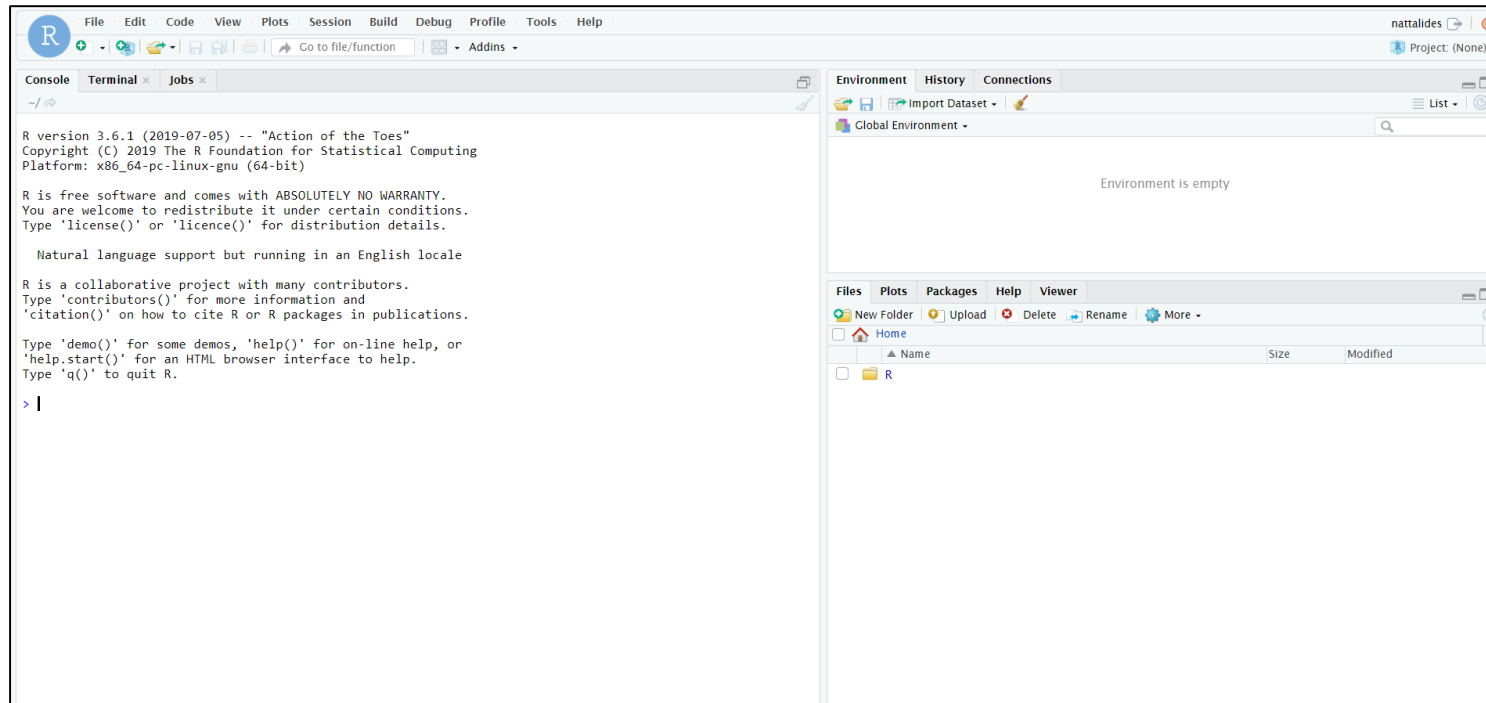
☐ Stay signed in

Sign In



Your chosen <username> and <password>

# RStudio server is running!






# Setup Shiny server



In order to setup Shiny server you will need to do the following:

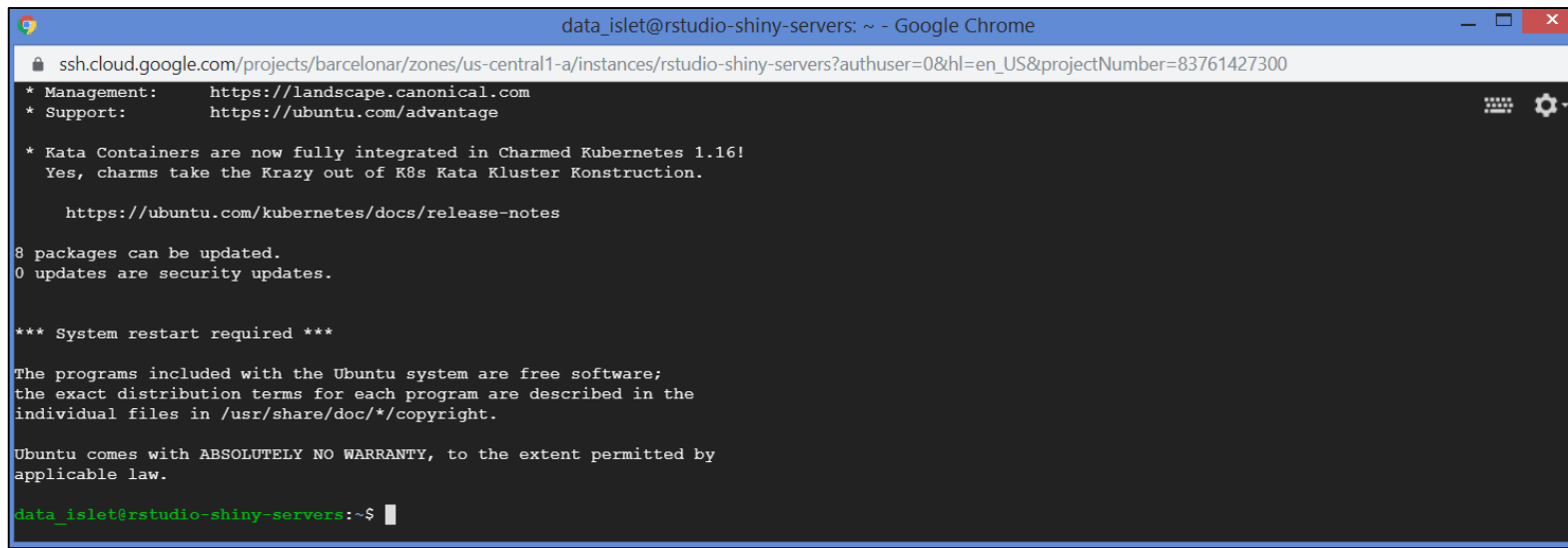
1. Connect to your VM instance (via SSH)
2. Install  server
3. Check shiny server status
4. Success! 🥳



Remember you need to run commands as the “superuser”

## Live Demo Part 3 – Connect to VM

Filter VM instances							?	Columns ▾
<input type="checkbox"/> Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect		
<input type="checkbox"/>  rstudio-shiny-servers	us-central1-a			10.128.0.2 (nic0)	34.67.35.39 ↗	SSH ▾		⋮



```
data_islet@rstudio-shiny-servers: ~ - Google Chrome
ssh.cloud.google.com/projects/barcelonar/zones/us-central1-a/instances/rstudio-shiny-servers?authuser=0&hl=en_US&projectNumber=83761427300
* Management: https://landscape.canonical.com
* Support:    https://ubuntu.com/advantage

* Kata Containers are now fully integrated in Charmed Kubernetes 1.16!
  Yes, charms take the Krazy out of K8s Kata Kluster Konstruktion.

https://ubuntu.com/kubernetes/docs/release-notes

8 packages can be updated.
0 updates are security updates.

*** System restart required ***

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

data_islet@rstudio-shiny-servers:~$
```

# Install Shiny server & check status

```
data_islet@rstudio-shiny-servers:~$ sudo apt-get install gdebi-core
...

data_islet@rstudio-shiny-servers:~$ wget https://download3.rstudio.org/ubuntu-14.04/x86_64/shiny-
server-1.5.12.933-amd64.deb
...

data_islet@rstudio-shiny-servers:~$ sudo gdebi shiny-server-1.5.12.933-amd64.deb
...
Do you want to install the software package? [y/N]: Y
...

data_islet@rstudio-shiny-servers:~$ sudo systemctl status shiny-server
...
```



To stop the shiny server: `sudo systemctl stop shiny-server`



To start the shiny server: `sudo systemctl start shiny-server`

# Shiny server is running!

In a web browser navigate to the following address: <http://<External IP>:3838>

Where <External IP> is found

...



Filter VM instances							Columns
<input type="checkbox"/>	Name ^	Zone	Recommendation	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	 rstudio-shiny-servers	us-central1-a			10.128.0.2 (nic0)	34.67.35.39 	SSH ▾ ⋮

## Welcome to Shiny Server!

If you're seeing this page, that means Shiny Server is installed and running. **Congratulations!**

### What's Next?

Now you're ready to setup Shiny — if you haven't already — and start deploying your Shiny applications.

If you see a Shiny application running on the right side of this page, then Shiny is configured properly on your server and already running an example. Bravo! You can see this application on your server at </sample-apps/hello/>.

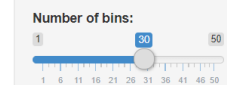
If you see a gray box or an error message, then there's a bit more work to do to get Shiny running fully. You can continue with [the installation instructions](#) or use [the Admin Guide](#) for more information. If you're seeing an error message in the panel to the right, you can use it to help diagnose what may be wrong. If you think Shiny is installed and setup properly and things still aren't working, you can look in the Shiny Server log which may have more information about what's wrong. By default, the log is stored in `/var/log/shiny-server.log`.

If you're really stuck and you've read the relevant sections in [the Admin Guide](#) then please ask for help on our [RStudio Community forum](#).

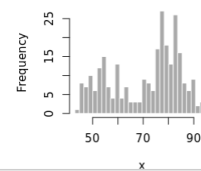
### rmarkdown

Once you have Shiny working properly (the top application on the right

### It's Alive!



### Histogram of x





When Shiny is properly configured on your server, you'll see a Shiny app above.



# Live Demo Part 4 - Host a shiny app online



In order to host a shiny app online you will need to do the following:

1. Log in to your RStudio server user account
2. Create a folder that will contain the shiny app scripts
3. Write the ui.R script
4. Write the server.R script
5. Create a symbolic link to the folder \* 
6. Test that it works
7. Share the URL! 



\*It's easier than it sounds

# Example ui.R script



```
# Define UI for application
ui <- fluidPage(

  # Application title
  titlePanel("Hello BarcelonaR!"),

  # Sidebar with an input
  sidebarLayout(
    sidebarPanel(
      textInput("text_input", "Input text here:")
    ),

    # Main with output
    mainPanel(
      textOutput("text_output")
    )
  )
)
```



# Example server.R script

```
library(shiny)

# Define server logic and R code
server <- function(input, output) {

  output$text_output <- renderText({
    # Display text input
    paste("You typed:", input$text_input)
  })
}
```

## Create symbolic link

```
data_islet@rstudio-shiny-servers:~$ sudo ln -s /home/<username>/my-app /srv/shiny-server/my-app
```



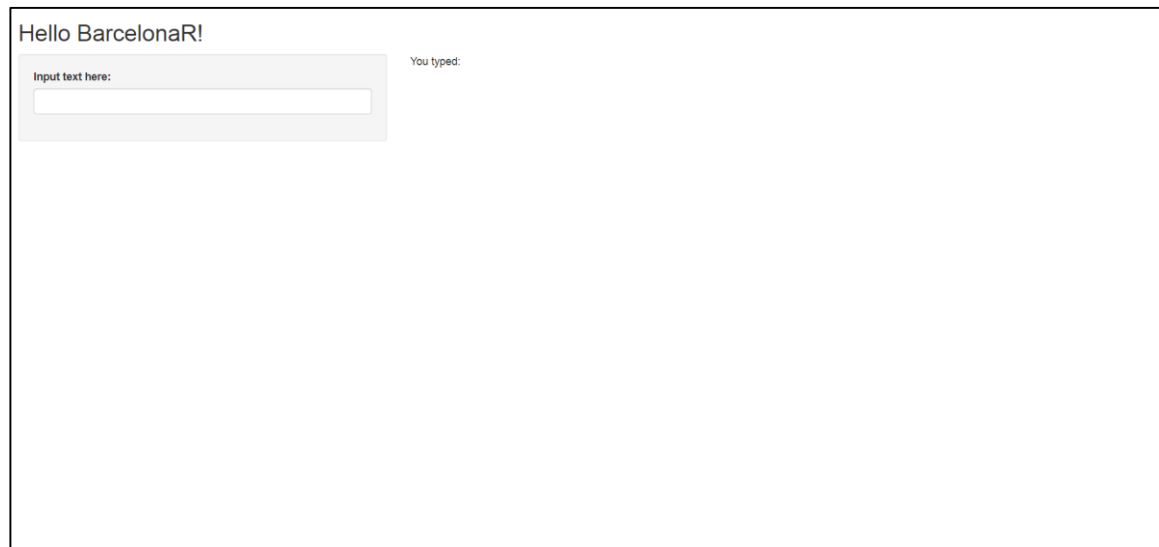
Your chosen RStudio server <username>

# Your shiny app is hosted online!

In a web browser navigate to the following address:

<http://<External IP>:3838/<app-folder>>


Where <External IP> is found just as before and <app-folder> is the name of the folder that contains the ui.R and server.R scripts



If you stop and start the GCP VM instance you will most likely get assigned a different <External IP> address!



## Other improvements

- ▶ If your shiny app code is becoming larger and more complex then why not Build a Production Grade Shiny App with **{golem}**
  - ▶ Write your shiny app in a project with code version control (such as GitHub)
  - ▶ Make <External IP> static
  - ▶ Add user authentication to Shiny Server with Nginx
  - ▶ Create an SSL certificate for Shiny server (https)
  - ▶ Control who can access your shiny apps (via GCP firewall settings)
  - ▶ Create custom domains for RStudio server, Shiny server and for your shiny apps
-  Check out: <https://docs.rstudio.com/shiny-server/> for a useful guide on how to customise other aspects of the Shiny server

# Tips for troubleshooting

- ▶ If your shiny app crashes you can use the stored shiny server logs to view what happened
  - 1) navigate to `cd /var/log/shiny-server/`
  - 2) list of available logs `ls`
  - 3) view log `cat <file-name>.log`
- ▶ In most cases the issue might be due to file and/or folder permissions – you might need to give root permissions to read/write/execute
- ▶ You might need to install necessary libraries using:
  - 1) `sudo R`
  - 2) `install.packages('magrittr')`so that they are available at root level and to all users

Thank you to our sponsors and partners!

