

INSTALLATION

OF

EBER

INDEX

<u>(I) Things need to be Pre-installed.</u>	6
<u>(II) Before we move further, things to consider.</u>	6
(A) Project on Server	6
(i)Things to keep handy.	6
(B) Project in Local	7
(i)Steps to Skip	7
(ii)Perform the following steps	7
(C) Important notes.	8
(i) Default admin creds	8
(ii) Important commands	8
<u>1. Purchasing Server</u>	9
<u>1.1 Purchasing Server On AWS</u>	9
<u>1.1.1 Selecting region for instance</u>	9
<u>1.1.2 Purchasing Instance</u>	10
<u>1.1.2.1: Select EC2.</u>	10
<u>1.1.2.2: Select instances.</u>	11
<u>1.1.2.3: Select Launch</u>	11
<u>1.1.2.4: Launching our instance</u>	12
<u>1.2.4(i) Choose an AMI</u>	12
<u>1.1.2.4(ii) Choose instance type</u>	13
<u>1.1.2.4(iv) Configure storage</u>	13
<u>1.1.2.4(v)Name and Tags(Optional)</u>	14
<u>1.1.2.4(vi) Configure security group</u>	15
<u>1.1.2.4(vii) Security Keys</u>	15
<u>1.1.2.4(viii) Launch Instance</u>	16
<u>1.2 Purchasing Server On Digitalocean</u>	18
<u>1.2.1) Creating a Droplet</u>	18
<u>1.2.2) Choose OS</u>	18
<u>1.2.3) Choose size for droplet.</u>	19
<u>1.2.4) Choose an authentication method.</u>	20
<u>1.2.5) Check Recommendation</u>	20
<u>1.2.6) Optional add tags</u>	21
<u>1.3 Purchasing Server On Vultr:</u>	22
<u>1.3.1 Create instance on Vultr dashboard</u>	22
<u>1.3.2 Select the Server and CPU</u>	22
<u>1.3.2.1 Choose Server</u>	22
<u>1.3.2.2 Select nearest server</u>	23
<u>1.3.2.3 Select the server Image</u>	24
<u>1.3.2.4 Keep these settings as default.</u>	25

<u>1.3.2.5 Server hostname</u>	25
<u>1.3.2.6 Server details</u>	26
<u>2. Security Groups</u>	27
<u>2.1 Security groups used in AWS?</u>	27
<u>2.1.1 In the navigation pane, choose Security Groups.</u>	27
<u>2.1.2 Choose your instance and create a security group.</u>	27
<u>2.1.3 Configure Security Inbounds</u>	28
<u>2.1.4 Add IPv4 addresses with HTTPS</u>	29
<u>2.2 Security groups used in Digital Ocean.</u>	31
<u>2.2.1 Create Firewall</u>	31
<u>2.2.2 Adding Rules to the Firewall group</u>	31
<u>2.2.3 Connecting the Firewall Group With the Droplet</u>	32
<u>2.3 Security groups used in Vultr.</u>	33
<u>2.3.1 Navigate to Firewall</u>	33
<u>2.3.2 Add Firewall group</u>	34
<u>2.3.3 Add Rules in the group</u>	34
<u>2.3.4 Connect Firewall group with the instance</u>	35
<u>3. Elastic IP / Reserved IP</u>	36
<u>3.1 Elastic IP using AWS.</u>	36
<u>3.1.1 Choose Allocate Elastic IP address.</u>	36
<u>3.1.2 Navigate to the page</u>	37
<u>3.1.3 Select IP</u>	37
<u>3.1.4 Select your project</u>	38
<u>3.2 Elastic IP using DigitalOcean</u>	39
<u>3.2.1 Navigate to Reserved IPs</u>	39
<u>3.3 Elastic IP using Vultr</u>	39
<u>3.3.1 Navigate to Reserved IPs</u>	39
<u>4. Create Storage(Optional).</u>	41
<u>(i) For AWS S3</u>	41
<u>(ii) For Digital-Ocean Spaces</u>	43
<u>4.1 Using AWS S3 Bucket</u>	43
<u>4.1.1 What is S3 Bucket and why do we use it?</u>	43
<u>4.1.2 Click on s3 from all services</u>	44
<u>4.1.3 Click on Create bucket</u>	44
<u>4.1.4 Add bucket name and select region according to country.</u>	45
<u>4.1.5 Unblock public access.</u>	46
<u>4.1.6 Now Click on Create Bucket.</u>	47
<u>4.1.7 How to get an S3 bucket image URL</u>	47
<u>(a) Update the image url value in the following listed files after the installation process has completed using bash file.</u>	47
<u>4.1.8 S3 bucket Access key and secret key download and save it in the database.</u>	48

<u>4.2 Using Digital-Ocean Spaces</u>	50
<u>4.2.1 Create Spaces Bucket</u>	50
<u>4.2.2 Configure the Spaces Bucket</u>	50
<u>5. Pointing Domain name, and SSL certificate</u>	52
<u>5.1 Domain Name(DNS)</u>	52
<u>5.2 SSL certificate</u>	55
<u>6 Establishing SSH connection</u>	56
<u>6.1 Open PuTTY or Terminal</u>	56
<u>6.2 Connect with Password.</u>	57
<u>6.3 Connect with PEM / PPK File</u>	60
<u>6.4 Perform the below steps to install all dependencies.</u>	64
<u>6.4.1 Place the bash file and SSL using FileZilla</u>	64
<u>6.4.2 Loading the Server with all the requirements.</u>	68
<u>6.5 Create MongoDB Atlas</u>	70
<u>6.6 Connection to the DB</u>	74
<u>7. Firebase Configuration</u>	75
<u>7.1 Go to the console.</u>	75
<u>7.2 Create Project</u>	75
<u>7.3 Create a web app in firebase General project settings.</u>	79
<u>7.4 Create a Service account</u>	80
<u>7.5 Create a Realtime Database.</u>	80
<u>7.6 Add Auth Sign-In Methods.</u>	83
<u>7.7 Add this firebase config</u>	84
<u>8. Basic setup on Admin panel</u>	85
<u>8.1 Set up Google api keys</u>	85
<u>8.2 Set up Notification Firebase Keys</u>	85
<u>9. Chat Push Firebase</u>	87
<u>9.1 Open putty with Backend instance of your project</u>	87
<u>10. Google API key</u>	89
<u>10.1 Open Google cloud console.</u>	89
<u>10.2 Copy that API key</u>	89
<u>10.3 Set this API Key in the given files.</u>	89
<u>10.4 Activate following Google API Keys</u>	90
<u>10.5 Configure Auth consent screen if not</u>	91
<u>10.6 Set up APIs in code</u>	91
<u>-> src/environments/environment.ts</u>	91
<u>11. Social login using Gmail</u>	93
<u>11.1 Open google console</u>	93
<u>11.2 Select Web client auto service.</u>	93
<u>11.3 Add your domain name to URI</u>	94
<u>11.4 And replace client_id in your project Database.</u>	94

<u>12. Facebook Login.</u>	95
<u>12.1 Open Facebook developer console.</u>	95
<u>12.2 Create app</u>	95
<u>12.3 Select your project app.</u>	96
<u>12.4 Click on Verification</u>	97
<u>12.5 Click on Products</u>	97
<u>12.6 Add Domains of your project</u>	98
<u>12.7 Give advance access</u>	98
<u>12.8 Complete basic settings</u>	99
<u>12.9 Data use check up</u>	99
<u>12.10 Save the changes</u>	99
<u>12.11 Replace FB App ID</u>	100
<u>13. Note For Folder Structure.</u>	101
<u>13.1 List of empty folders in clone:</u>	101
<u>13.2 File/Folder structure</u>	101
<u>13.3 The following keys/file</u>	101
<u>13.4 Overview of code.</u>	102
<u>14. Stripe Configuration.</u>	103
<u>14.1 Login Stripe Account</u>	103
<u>14.2 Use this test mode private and public key.and save this key in the Admin panel.</u>	104
<u>14.3 Add stripe publishable key in to user panel, partner panel & driver panel in code</u>	104
<u>14.4 And start the Connect Account</u>	104
<u>14.5 Activate Apple Pay</u>	105
<u>14.6 Add Webhook Url for Using Apple pay.</u>	107
<u>15. Twilio Configuration.</u>	109
<u>15.1 Login twilio</u>	109
<u>15.2 Purchase number which have sms and voice capability</u>	109
<u>15.3 In the trial account you have to verify number for test purpose.</u>	110
<u>15.4 Check your country</u>	110
<u>15.5 And save this Account SID,Auth Token & Phone number in Admin Panel Settings.</u>	111
<u>15.6 Twilio Call Masking</u>	112
<u>16. Terms & Condition Details.</u>	114
<u>16.1 Terms and Privacy Document add in to admin panel.</u>	114

Installation of EBER

(I) Things need to be Pre-installed.

- Ubuntu OS (minimum version 20.04)
- Code Editor (any)
- FileZilla (for transferring files)
- Putty (for connecting Server)
- MongoDB Compass or Robo-3t (for connecting DB)

(II) Before we move further, things to consider.

(A) Project on Server

(i) Things to keep handy.

- Purchased Domain
- Confirm repo access, as listed below.

The screenshot shows a project management interface with a sidebar on the left and a main content area on the right. The sidebar has tabs for 'Subgroups and projects', 'Shared projects', and 'Archived projects'. On the right, there is a search bar and a dropdown menu for 'Name'. The main content area displays a list of items under a 'Web' subgroup, indicated by a purple icon and the letter 'W'. The items are: Admin Panel, Backend, Corporate Panel, Dispatcher Panel, Driver Panel, Hotel Panel, Hub Panel, Partner Panel, and User Panel. The 'Hotel Panel' item is highlighted with a light blue background, indicating it is selected. A red box is drawn around the entire 'Web' subgroup section.

- Follow every step mentioned from [Section 1](#) of this document.

(B) Project in Local

(i) Steps to Skip

- Skip all the following steps from [section 1](#) of this document.

(ii) Perform the following steps

- Download the **EBER_INSTALLATION.sh** file.
 - Ask for this bash file or you can get it directly from the following location, navigate to **backend/server/documentation/installation documentation/EBER_INSTALLATION.sh**
- Navigate to the path of downloaded **sh** file and open the terminal using right-click and run the command **bash EBER_INSTALLATION.sh**
- Press 1 for full installation
- Skip the following mentioned steps in the process of running bash file.
 - Step-10
 - Step-12
 - Step-13
- After the completion of the process navigate to the folder using the file system where you want to clone the project and then open the terminal using the right click.
- Follow the steps to clone the code locally .
 - Create directory using **mkdir <ProjectName>**
 - For Frontend(Admin, Corporate, Driver, Dispatcher, Hub, Hotel, Partner and User).
 - 1. Clone using **git clone <https git url>**
 - 2. Enter Username And Password
 - 3. Enter the cloned folder **cd <FolderName>**
 - 4. Install dependencies using **npm i**
 - 5. Exit that folder **cd ..**
 - Repeat the above step(1,2,3,4,5 steps) for all the frontend git project urls.
 - For Backend(Backend).
 - 1. Clone using **git clone <https git url>**
 - 2. Enter Username And Password
 - 3. Enter the cloned folder **cd <FolderName>**
 - 4. Enter the folder **cd <FolderName>**(server, history, payments and mass notification)
 - 5. Install dependencies using **npm i**
 - 6. Exit that folder **cd ..**
 - Repeat the above step(4,5,6 steps) only for the backend git project url

- Navigate to the project and run the following commands:
 - Navigate to the **initial_data.js** file to fill all the basic data in DB.
 - backend/server/settingsdata/initial_data.js
 - **node initial_data.js**
 - Navigate to the **angular.json** file of all the panels other than the backend and replace the “**dist_new**” with “**dist**” by searching the field “**outputPath**”

(C) Important notes.

- (i) Default admin creds
 - Username “**eber**” password “**developertest123abcxyz@**”
- (ii) Important commands
 - For Running the panels in local
 - **npm start**
 - For Running the backend in local
 - **npm start**
 - **For serving all the panels and backend code on the server, the bash file will do everything but in case you may need to know the command to serve the code, below code can be used for reference.**
 - For Running the panels on Server
 - Need to upload the build and then run the server.js file after navigating to each panel folder..
 - **pm2 start server.js --name <PanelName>**
 - The above command needs to be followed for every panel.
 - For Running the backend on Server
 - Navigate to the backend folder one by one, ie. server, history-earning, payments, mas_notification and use the following command.
 - **pm2 start server.js --name <BackendName>**

Note: If you run the panels using the **npm start** command it will use the environment.js file and if we run the panels using **npm run build** command it will use the environment.prod.js file.

1. Purchasing Server

Follow the steps to purchase instance:- Below there are total 3 ways to purchase and create instance on cloud,

i.e: [AWS\(1.1\)](#) or [Digital Ocean\(1.2\)](#) or [Vultr\(1.3\)](#) you can choose any of the mentioned cloud services according to your convenience.

1.1 Purchasing Server On AWS

Create an account on AWS and complete basic registration steps that are required to get done before we can access AWS benefits.

After registration now we need to purchase an instance(server) where we can install our code. So for that, we need to purchase instances.

Register here :- [AWS account Register](#) Login here :- as root [AWS account Login](#)

1.1.1 Selecting region for instance

- We need to select the region (from the top right corner) wisely so that we get the best latency and API responses in less time. Select the region where this application is going to be used most of the time.
- If a region is not available in a particular country or state, then use the region which is nearest.

The screenshot shows the AWS EC2 Experience dashboard. At the top, it says "You are using the following Amazon EC2 resources in the US East (Ohio) Region:". Below this are four boxes: "Instances (running)" (0), "Dedicated Hosts" (0), "Elastic IPs" (0), "Instances" (0), "Key pairs" (0), "Load balancers" (0), "Placement groups" (0), "Security groups" (1), and "Volumes" (0). A callout box says "Easily size, configure, and deploy Microsoft SQL Server Always On availability groups on AWS using the AWS Launch Wizard for SQL Server. Learn more".

Launch instance

To get started, launch an Amazon EC2 Instance, which is a virtual server in the cloud.

Scheduled events

Migrate a server

Service health

Region: US East (Ohio) Status: This service is operating normally

Zones

Zone name	Zone ID
us-east-2a	use2-az1
us-east-2b	use2-az2
us-east-2c	use2-az3

Additional information

Feedback English (US) ▾

© 2006 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

1.1.2 Purchasing Instance

1.1.2.1: Select EC2.

- Navigate to the EC2.
 - Search for EC2
 - Select EC2

AWS services

Recently visited services

- EC2** (highlighted with a red box)
- S3
- VPC
- Systems Manager
- Billing
- IAM
- AWS AppConfig
- AWS Cost Explorer

All services

Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine	Build a web app	Build using virtual servers	Register a domain
With EC2 2-3 minutes	With Elastic Beanstalk 6 minutes	With Lightsail 1-2 minutes	With Route 53 3 minutes

Connect an IoT device	Start migrating to AWS	Start a development project	Deploy a serverless microservice
With AWS IoT 5 minutes	With AWS MGN 1-2 minutes	With CodeStar 5 minutes	With Lambda, API Gateway 2 minutes

Stay connected to your AWS resources on-the-go

AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device. [Learn more](#)

Explore AWS

Amazon Lookout for Metrics

Automatically detect anomalies in metrics and identify their root cause. [Learn more](#)

Free AWS Training

Complete projects faster and troubleshoot with confidence with 500+ free digital courses covering AWS products and services. [Learn more](#)

Calling All Java and Python Developers

Join the AWS BugBust challenge to bust one million bugs. [Learn more](#)

Feedback English (US) ▾

© 2008 - 2021, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

1.1.2.2: Select instances.

The screenshot shows the AWS EC2 Instances page. On the left, a sidebar menu includes 'Instances' (selected), 'Images', 'Elastic Block Store', and 'Network & Security'. The main content area has a 'Resources' section with a table of metrics like Instances (running), Dedicated Hosts, Elastic IPs, etc. Below it is a 'Launch instance' section with a large orange 'Launch instance' button. To the right are sections for 'Service health', 'Zones' (listing ap-south-1a and ap-south-1b with Zone IDs aps1-az1 and aps1-az3), and 'Account attributes' (with tabs for VPC, Default VPC, Settings, and Explore AWS). A status bar at the bottom indicates the service is operating normally.

1.1.2.3: Select Launch

Select Launch from the top right corner.

The screenshot shows the AWS EC2 Instances page with 17 running instances listed. The columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IPv4 DNS, and Public IPv4 IP. A red box highlights the 'Launch instances' button at the top right of the page.

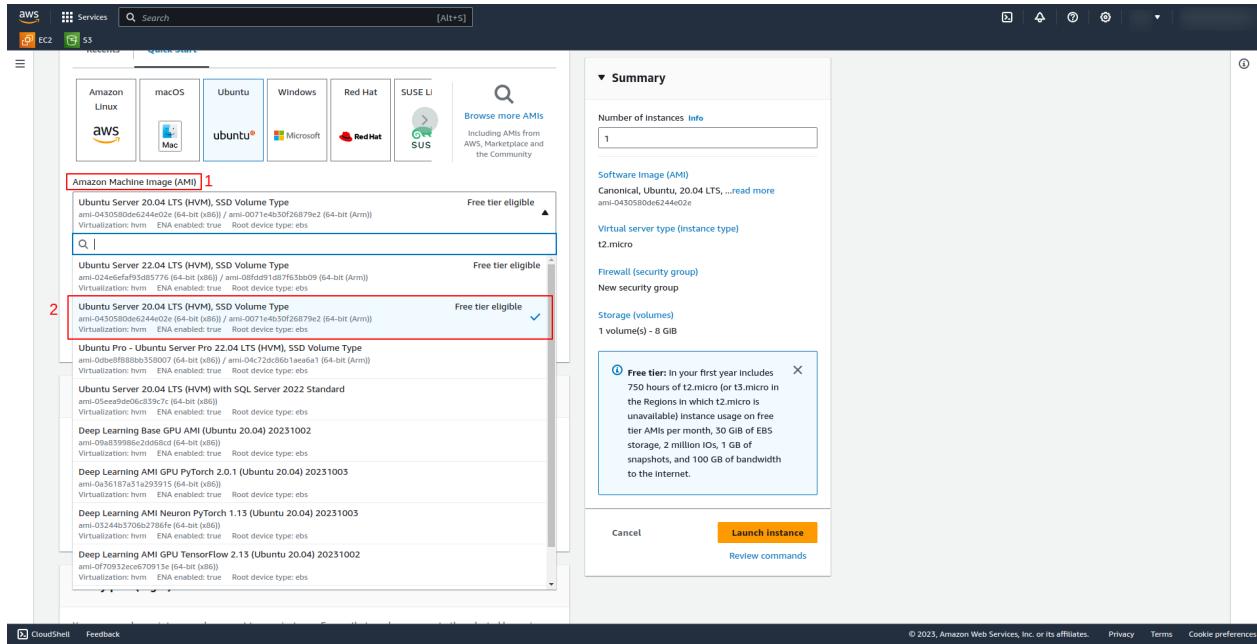
1.1.2.4: Launching our instance

1.2.4(i) Choose an AMI

- AMI is a bunch of basic preloaded software configurations like Operating Systems.

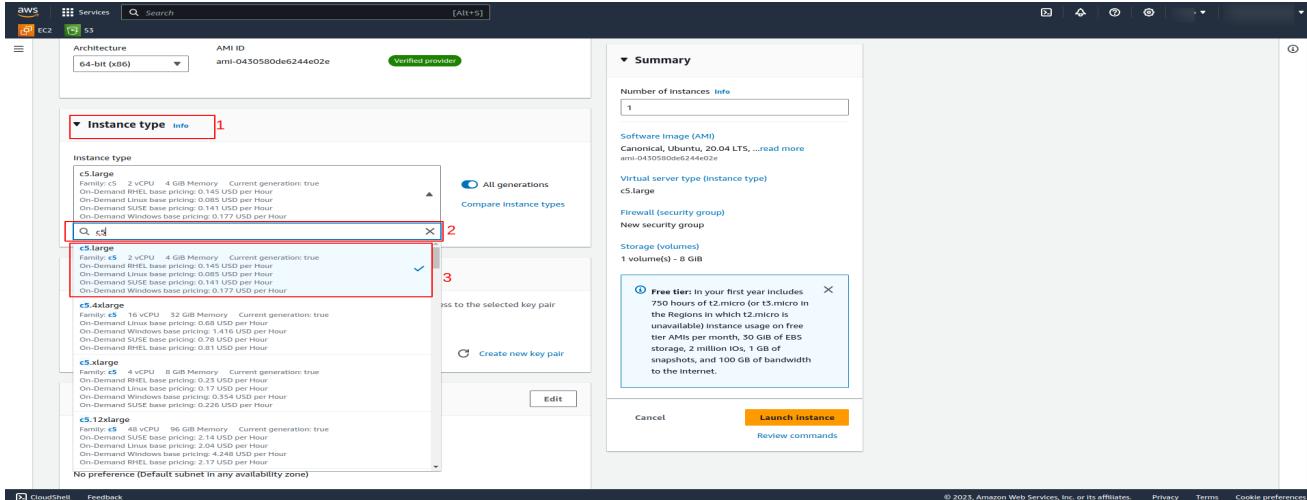
The screenshot shows the AWS EC2 Launch an instance page. Step 1 highlights the 'Application and OS Images (Amazon Machine Image)' section. Step 2 highlights the 'Ubuntu' AMI selection. A callout box provides information about the Free tier: 'Free tier: in your first year includes 750 hours of t2.micro for t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOPS, 1 GB of snapshots, and 100 GB of bandwidth to the internet.'

- Select Ubuntu which is most preferable according to our use case(minimum version 20.04).



1.1.2.4(ii) Choose instance type

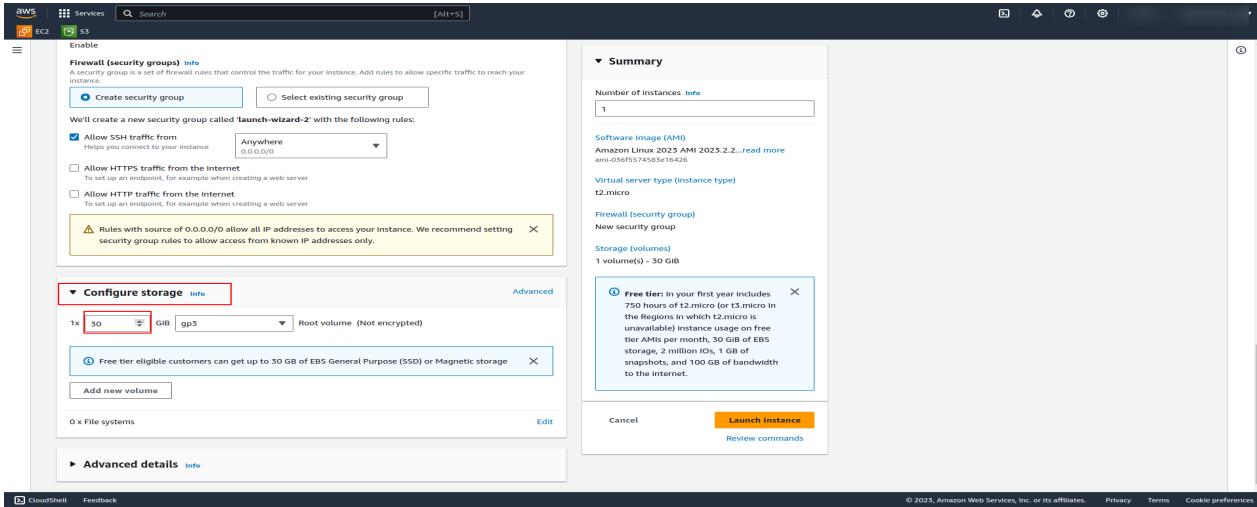
- Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications.
- Preferable selection c5.large for backend



1.1.2.4(iv) Configure storage

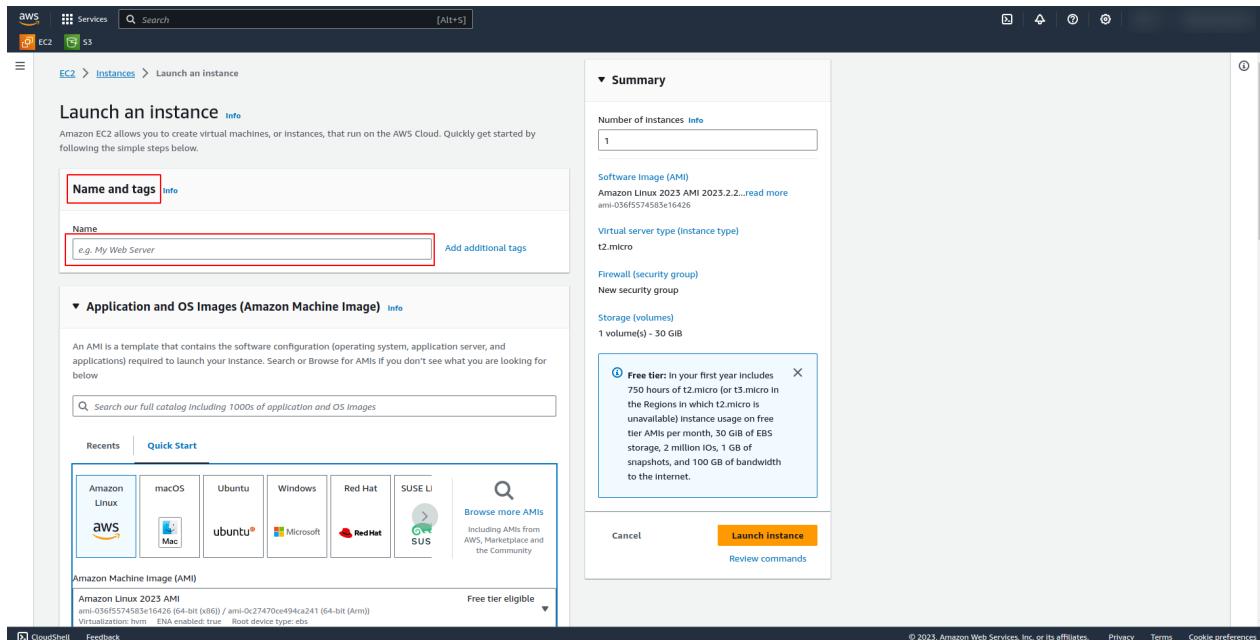
- You can use EBS volumes as primary storage for data that requires frequent updates, such as the system drive for an instance or storage for a database application. You can also use them for throughput-intensive applications that perform continuous disk scans.

- Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. Select 30 GiB.



1.1.2.4(v) Name and Tags (Optional)

- Give the name to the instance.
- A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Web Server. A copy of a tag can be applied to volumes, instances or both.
- This step is optional for us.

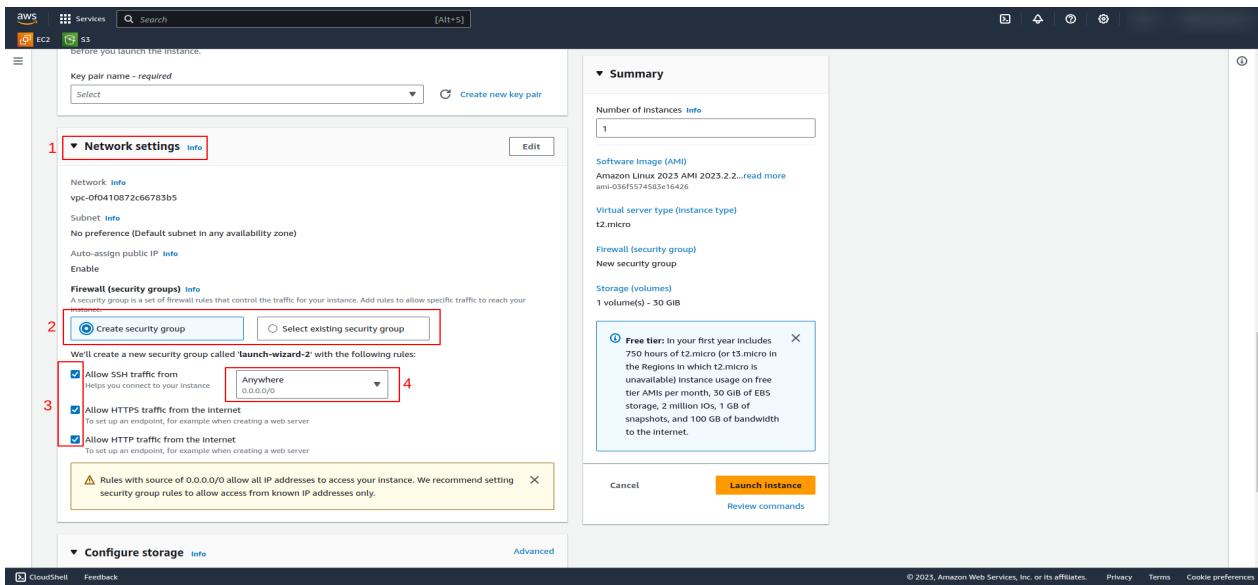


1.1.2.4(vi) Configure security group

- A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance.
- We will require SSH, HTTP, and HTTPS security groups to have ssh connection and http and https connections from web browsers.

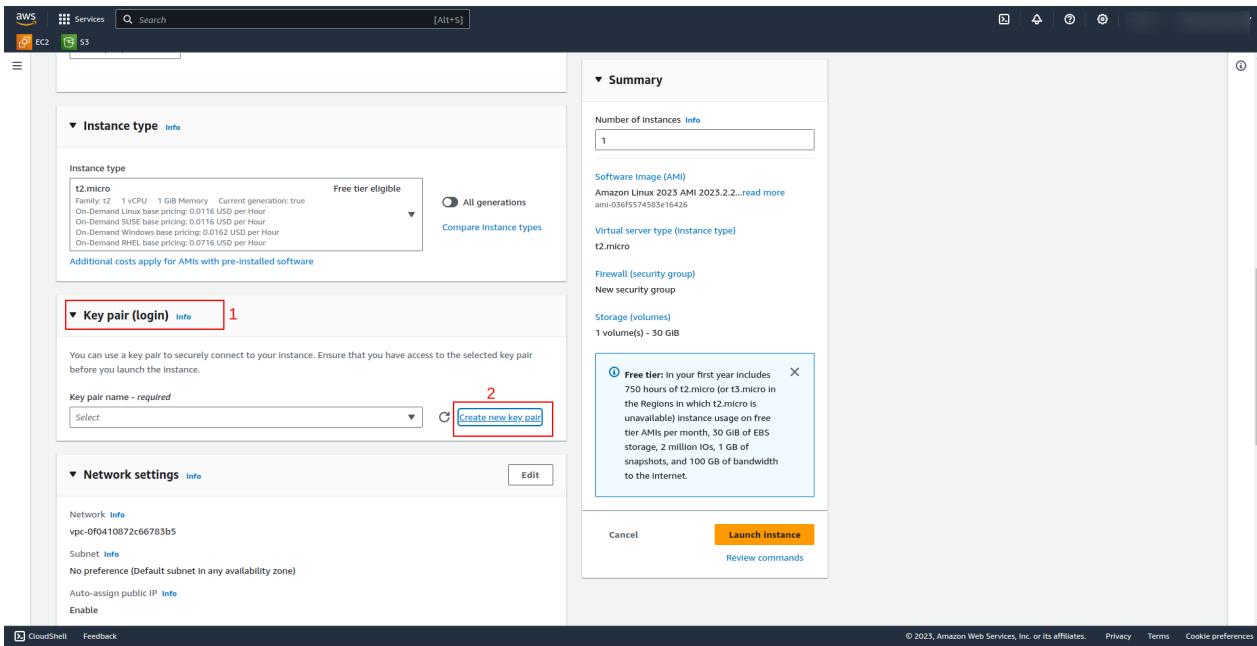
Create a new security group or select the existing one.

If creating a new security group, follow the steps mentioned below and select the option anywhere at step 4.

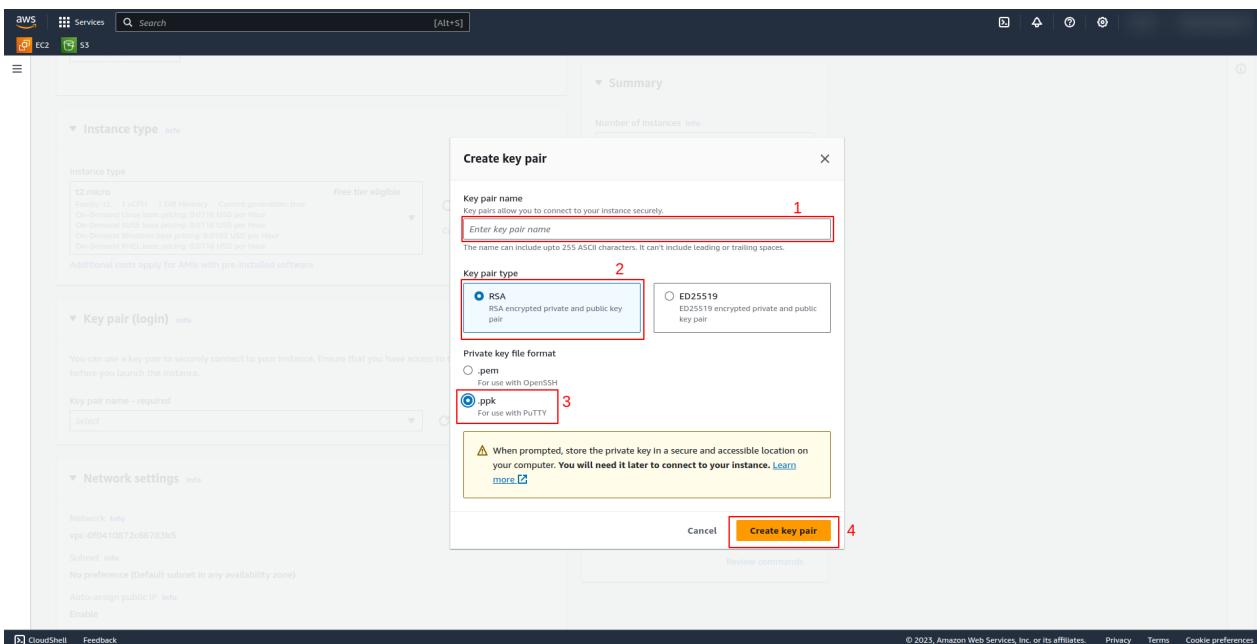


1.1.2.4(vii) Security Keys

- As shown in the image below while reviewing we can see a popup for security. If you have any security keys then select that or generate a new one for our server.
- For creating new security keys we need to select Create a new key pair and enter the appropriate name for your key and press download key pair(store it at a safe place as1 we will need it to connect the server).



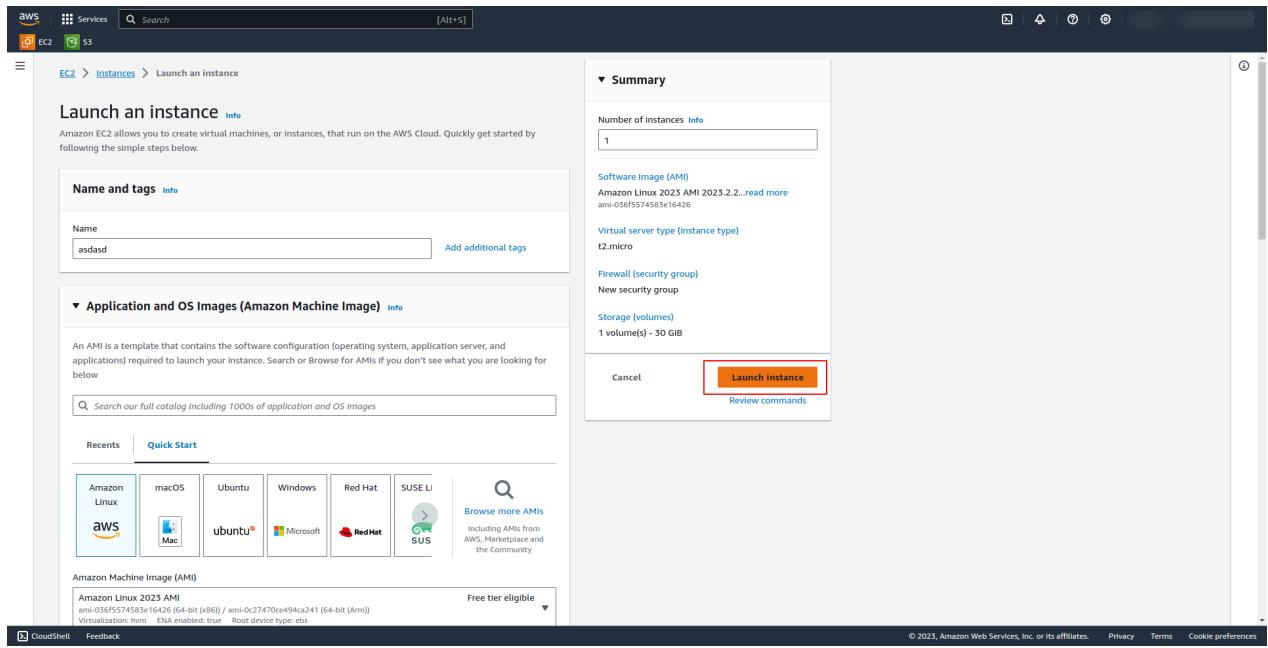
Follow the steps highlighted below.



1.1.2.4(viii) Launch Instance

After reviewing all the things properly click on the launch instance.

After a few minutes the server will be up and running.

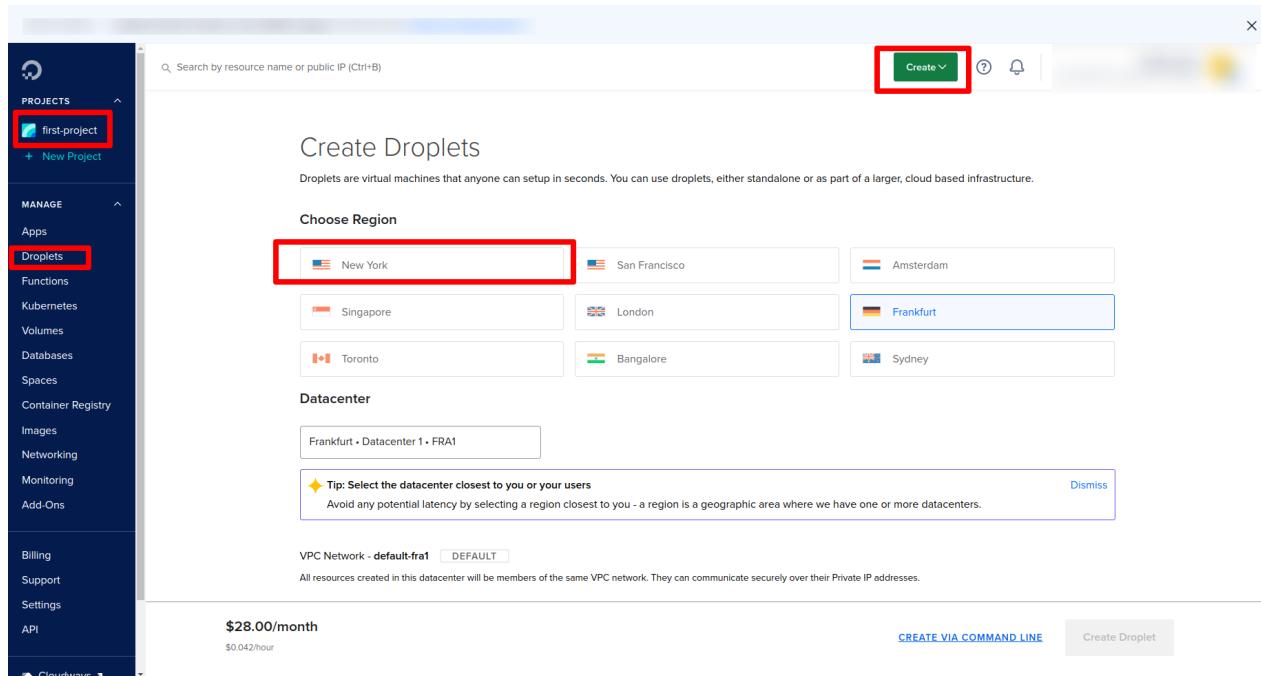


1.2 Purchasing Server On Digitalocean

Account [Register](#) or [Login](#)

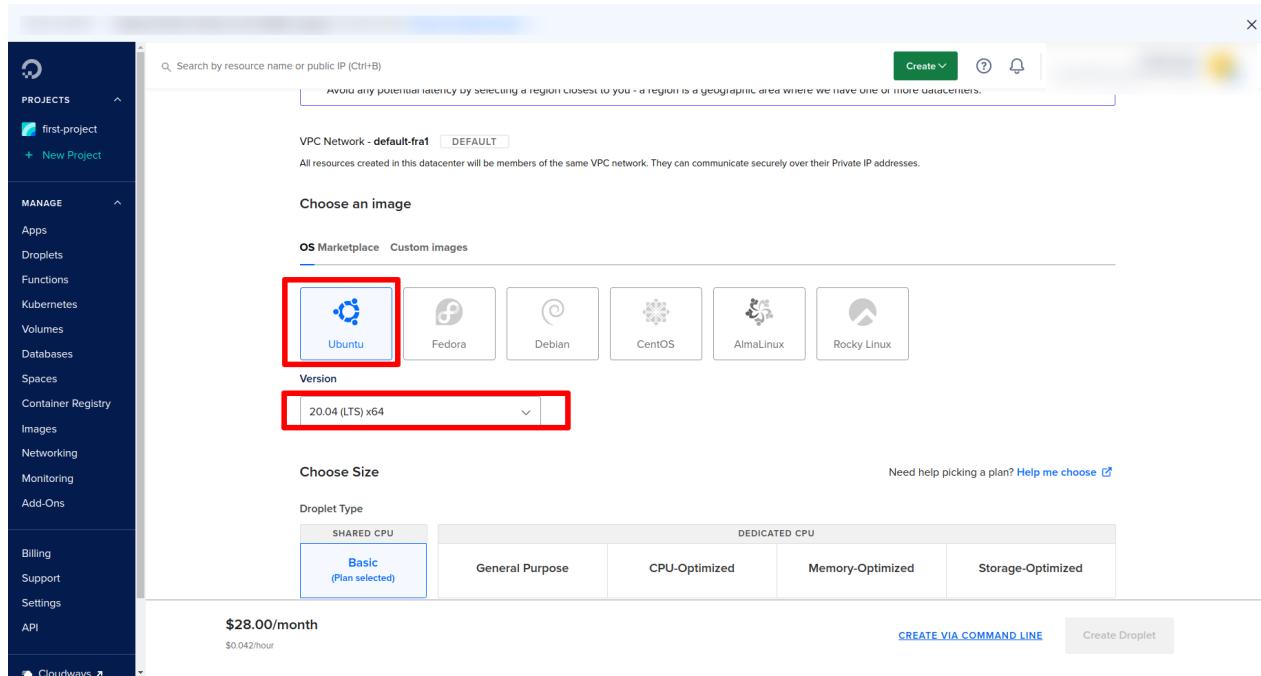
1.2.1) Creating a Droplet

choose organization, go to droplet, if there is no droplet. Create a new droplet with a create button. Choose your nearest region as per your requirements. Most of the time we go through new york region.



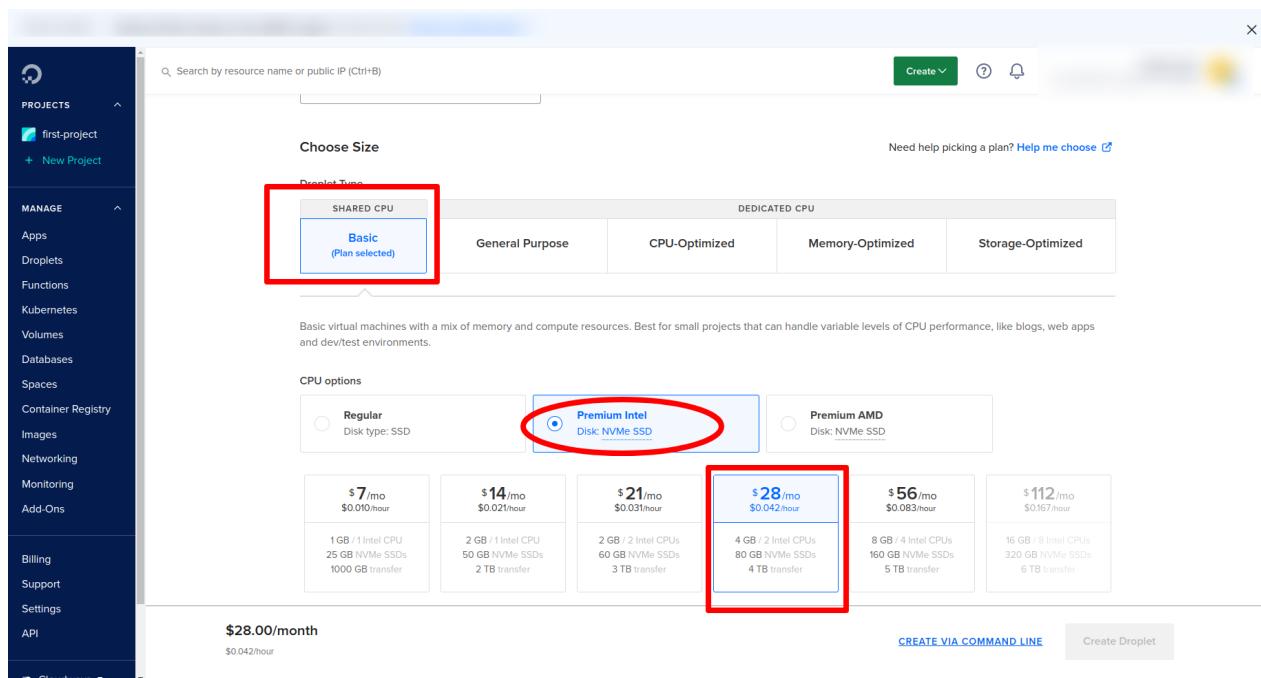
1.2.2) Choose OS

ubuntu 20.04 LTS



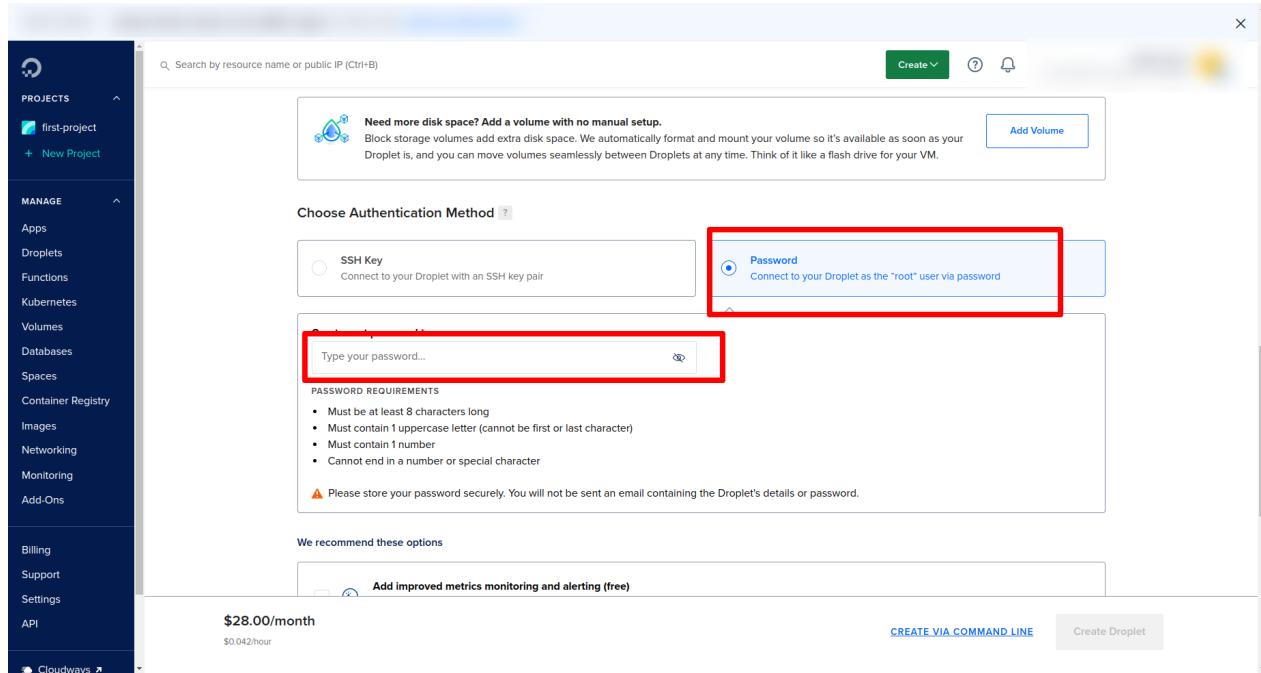
1.2.3) Choose size for droplet.

Go for a basic plan. Select premium intel CPU. choose 28\$/month plan



1.2.4) Choose an authentication method.

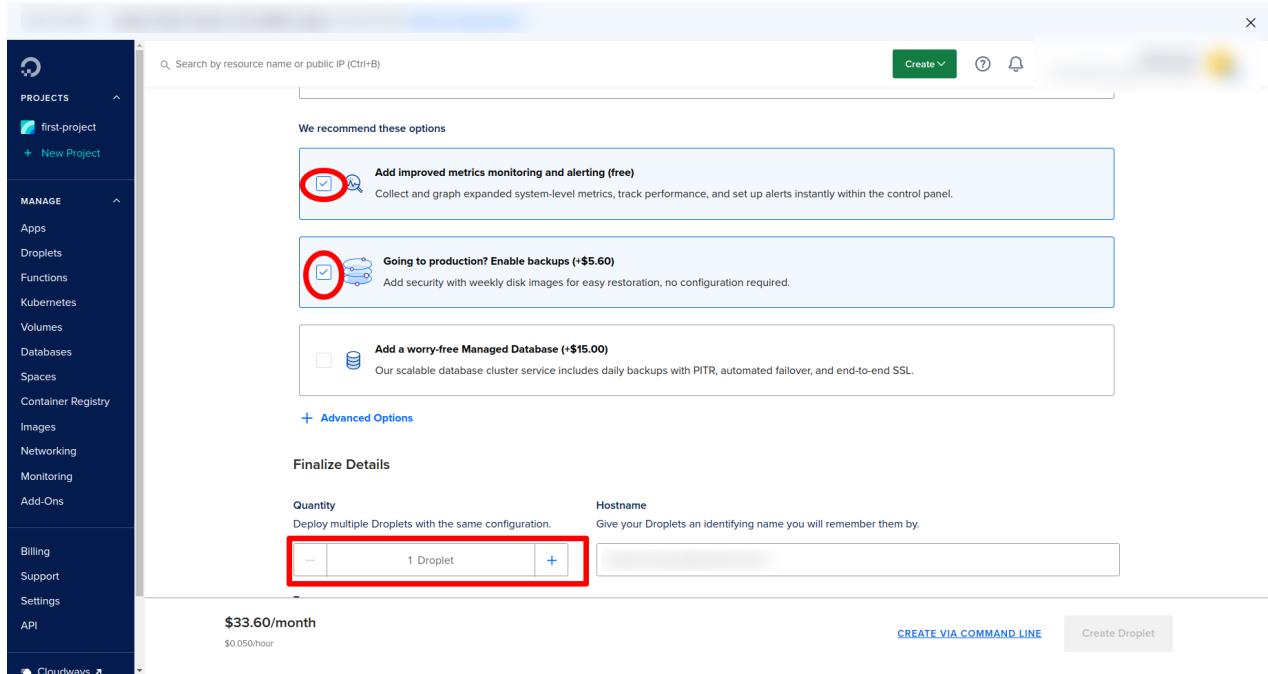
Go to the password method. Use root password. You can create your own or you can create a password with the help of password generator. It will give you a strong password (copy it and save it at safe place).



1.2.5) Check Recommendation

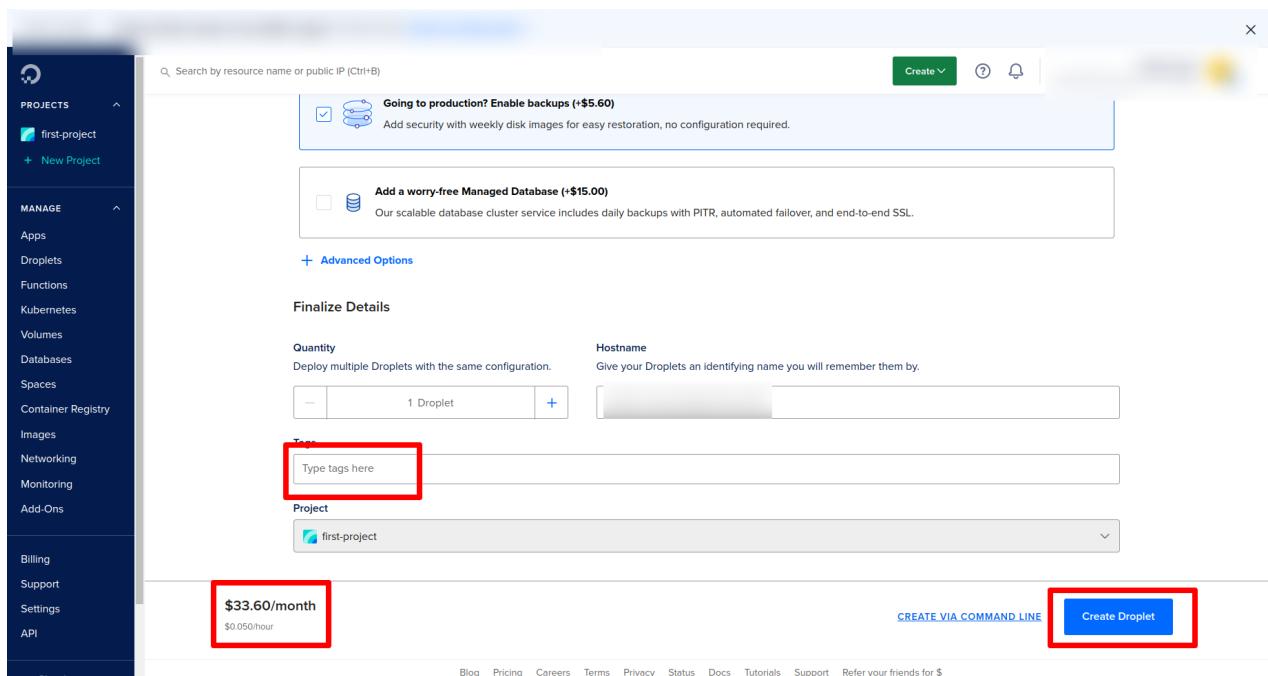
After creating a password check the first free recommendation option. The 2nd option is backup of the server. If want this will charge 5\$

Quantity 1 droplet and auto create hostname



1.2.6) Optional add tags

If you go through this it approximately charges 33\$. Hit create droplet button



On success creation of droplet you will redirect the droplet section where you get ip and password(which you created before while creating droplet). This credential will help you to connect with server.

1.3 Purchasing Server On Vultr:

Create an account on Vultr and complete basic registration steps that are required to get done before we can access Vultr benefits.

After registration now we need to purchase an instance(server) where we can install our code. So for that, we need to purchase instances.

Register here :- [Vultr account Register](#)

Login here :- as root [Vultr account Login](#)

1.3.1 Create instance on Vultr dashboard

- We need to select the region wisely so that we get the best latency and API responses in less time. Select the region where this application is going to be used most of the time.

- If a region is not available in a particular country or state, then use the region which is nearest.



1.3.2 Select the Server and CPU

1.3.2.1 Choose Server

Choose the server accordingly as mentioned in the screenshots below.

The screenshot shows the Vultr interface for deploying a new instance. On the left sidebar, there are icons for Products, Billing, Support, Referral Program, Account, and Vultr Docs. The main content area has a header "Deploy New Instance". Under "Choose Server", there are four options: Optimized Cloud Compute, Cloud Compute, Cloud GPU, and Bare Metal. The "Cloud Compute" option is highlighted with a red box. Below this, under "CPU & Storage Technology", there are four options: AMD High Performance (highlighted with a red box), intel High Performance, intel High Frequency, and intel Regular Performance. At the bottom, it shows "Servers Qty: 1" and "Summary: \$14.40/month (\$0.021/hour)". A blue "Deploy Now" button is on the right.

1.3.2.2 Select nearest server

As mentioned earlier choose the server location nearest to the user's location.

The screenshot shows the Vultr interface for selecting a server location. The left sidebar includes icons for Products, Billing, Support, Referral Program, Account, and Vultr Docs. The main section is titled "Server Location" and shows a grid of server locations categorized by continent: All Locations, America, Europe, Australia, Asia, and Africa. The "All Locations" tab is selected. The grid contains 24 entries, each with a flag icon, city name, and country. Some locations have a "New" badge. At the bottom, it shows "Servers Qty: 1" and "Summary: \$14.40/month (\$0.021/hour)". A blue "Deploy Now" button is on the right.

1.3.2.3 Select the server Image

Select Ubuntu (minimum version 20.04)

And select the server size highlighted in the image given below.

The screenshot shows the Vultr interface for selecting a server image. The left sidebar includes icons for Products, Billing, Support, Referral Program, Account, and Vultr Docs. The main area has tabs for Operating System, Marketplace Apps, Upload ISO, ISO Library, Backup, and Snapshot. Under the Operating System tab, there is a grid of server images:

Image	Name	Description
AlmaLinux	AlmaLinux	Select Version
Arch Linux	Arch Linux	Latest x64
CentOS	CentOS	Select Version
Debian	Debian	11 x64
Fedora	Fedora	Select Version
Fedor CoreOS	Fedor CoreOS	Select Version
Flatcar Container Linux	Flatcar Container Linux	Select Version
FreeBSD	FreeBSD	Select Version
OpenBSD	OpenBSD	Select Version
Rocky Linux	Rocky Linux	Select Version
Windows Standard	Windows Standard	Select Version

The "Ubuntu" row is highlighted with a red box. A dropdown menu for "Ubuntu" shows available versions:

- 23.04 x64
- 22.10 x64
- 22.04 LTS x64
- 20.04 LTS x64** (highlighted with a red box)
- 18.04 LTS x64

Below the image selection, there is a "Server Size" section:

Size	Price
25 GB NVMe	\$6/month
50 GB NVMe	\$12/month
60 GB NVMe	\$18/month
100 GB NVMe	\$24/month

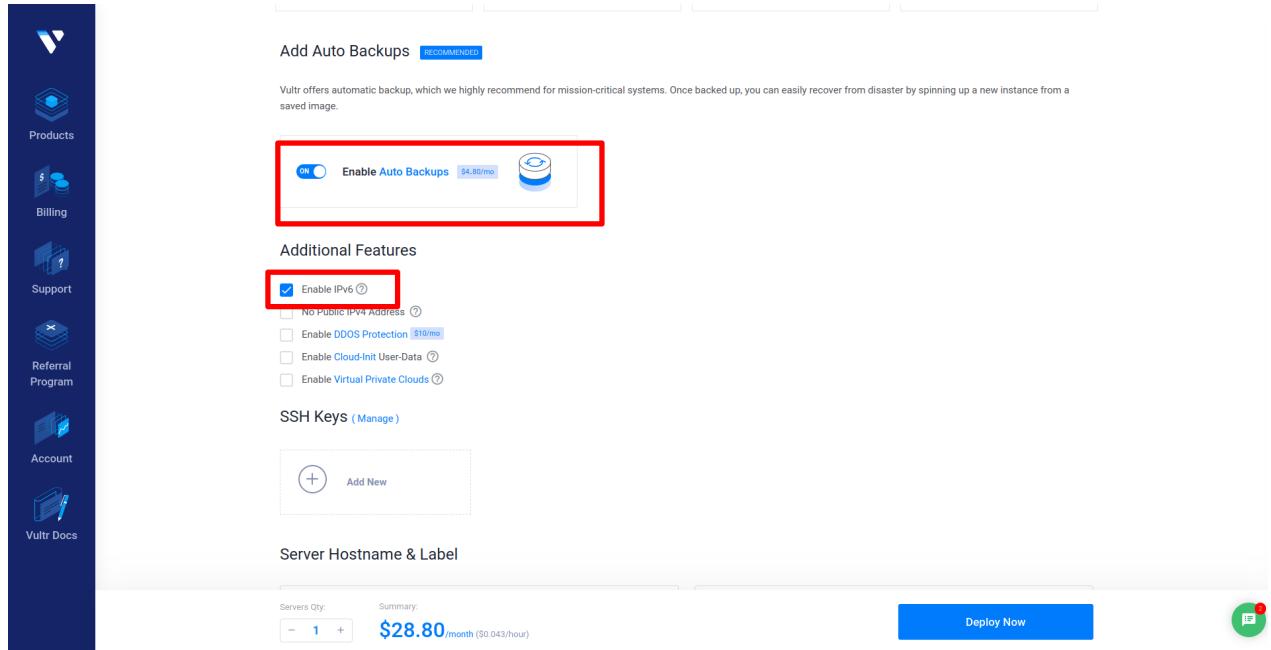
The "100 GB NVMe" row is highlighted with a red box. The summary at the bottom shows a total of \$14.40/month for 1 server. A "Deploy Now" button is visible.

The screenshot shows the Vultr interface for selecting a server size. The left sidebar includes icons for Products, Billing, Support, Referral Program, Account, and Vultr Docs. The main area has a "Server Size" section:

Size	Price	Summary
25 GB NVMe	\$6/month	\$0.009/hour
50 GB NVMe	\$12/month	\$0.018/hour
60 GB NVMe	\$18/month	\$0.027/hour
100 GB NVMe	\$24/month	\$0.036/hour
180 GB NVMe	\$48/month	\$0.071/hour
260 GB NVMe	\$72/month	\$0.107/hour
350 GB NVMe	\$96/month	\$0.143/hour
500 GB NVMe	\$144/month	\$0.214/hour

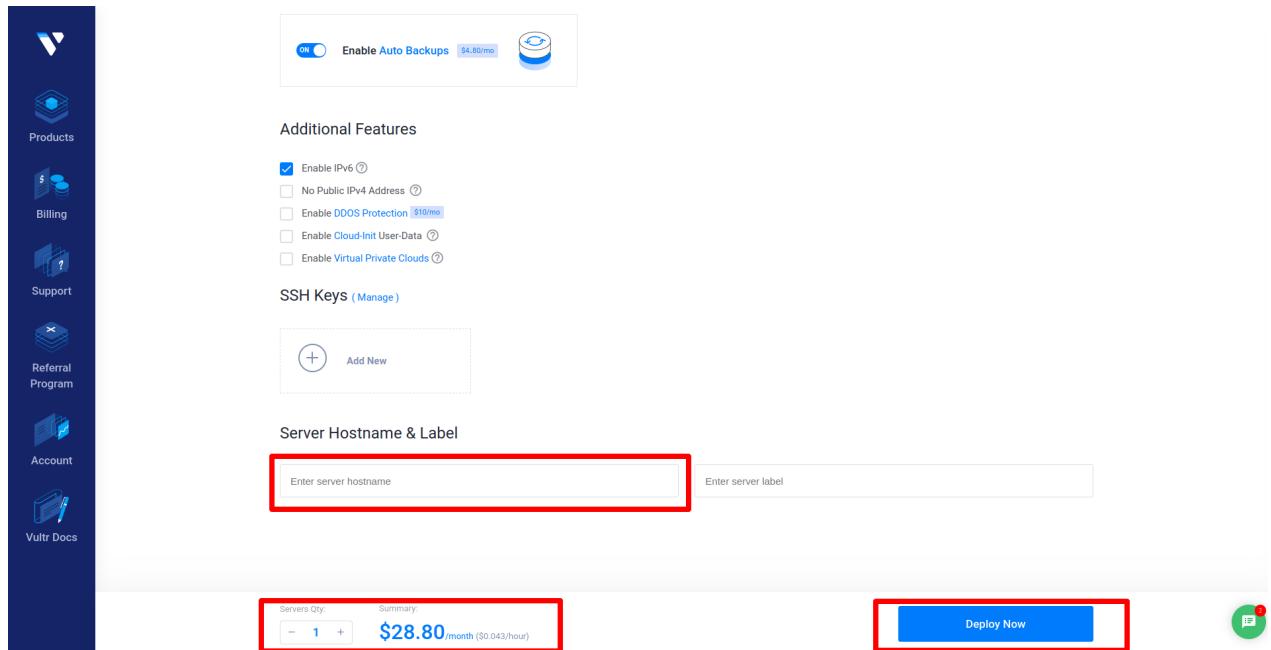
The "100 GB NVMe" row is highlighted with a red box. The summary at the bottom shows a total of \$28.80/month for 1 server. A "Deploy Now" button is visible.

1.3.2.4 Keep these settings as default.



1.3.2.5 Server hostname

Server hostname you want as tag. In the footer you can show estimation. Everything is ok then click deploy now. Check review at last before you deploy



1.3.2.6 Server details

Click on server detail to see IP, passwords and other server details

This screenshot shows the Vultr 'Products' section. On the left is a dark sidebar with icons for Instances, Kubernetes, Storage - Block / Object, Databases, Network, Load Balancers, ISOs / Snapshots / Scripts, and a '+' button. The main area is titled 'Products' and contains a search bar and a table of servers. A message at the top says 'Introducing Vultr Cloud GPUs powered by the NVIDIA A40. Provision your instance now.' A 'Dismiss' button is next to it. The table has columns for Server, OS, Location, Charges, and Status (which is 'Running'). A 'Server Details' button in the status column is highlighted with a red box. A tooltip for 'Server Details' lists options: View Console, Server Stop, Server Restart, Server Reinstall, and Server Destroy.

This screenshot shows the Vultr 'Overview' page for a specific server. The left sidebar is identical to the previous screenshot. The main area shows server details: Location (blurred), IP Address (redacted), Username (root), and Password (redacted). Above the details are buttons for monitor, power, refresh, and other server management functions. A green circular icon with an 'IP' and a red dot is in the bottom right corner.

2. Security Groups

Security groups work as firewalls for instance.

Follow accordingly:

i.e: [For AWS\(2.1\)](#) or [For Digital Ocean\(2.2\)](#) or [For Vultr\(2.3\)](#)

2.1 Security groups used in AWS?

Why are security groups used in AWS?

A security group acts as a virtual firewall for your EC2 instances to control incoming and outgoing traffic. Inbound rules control the incoming traffic to your instance, and outbound rules control the outgoing traffic from your instance. If you don't specify a security group, Amazon EC2 uses the default security group.

2.1.1 In the navigation pane, choose Security Groups.

Skip this security group section if you have already done at the time of Creating Instance ec2.

Jump to [section 3](#)

2.1.2 Choose your instance and create a security group.

The screenshot shows the AWS Management Console interface for managing security groups. The left sidebar navigation includes 'Dedicated Hosts', 'Capacity Reservations', 'Images' (with 'AMIs' and 'AMI Catalog' options), 'Elastic Block Store' (with 'Volumes' and 'Snapshots' options), 'Network & Security' (with 'Security Groups' selected, highlighted by a red box), 'Load Balancing' (with 'Load Balancers' and 'Target Groups' options), and 'Auto Scaling' (with 'Launch Configurations' and 'Auto Scaling Groups' options). The main content area displays a table titled 'Security Groups (1/3)'. The table has columns for 'Name', 'Security group ID', 'Security group name', 'VPC ID', and 'Description'. A single row is listed, with the 'Name' column containing a checked checkbox and a dropdown menu. The 'Security group name' column shows 'laun...' and 'd-1'. The 'Description' column shows 'launch-wizard-1 create...'. At the top right of the table, there is a 'Create security group' button, which is also highlighted by a red box. Below the table, a modal window titled 'INCN-WIZARD-1' is open, showing the 'Details' tab. It contains fields for 'Security group name' (with a dropdown menu), 'Security group ID' (with a dropdown menu), 'Description' (with a dropdown menu), 'VPC ID' (with a dropdown menu), 'Owner' (with a dropdown menu), 'Inbound rules count' (17 Permission entries), and 'Outbound rules count' (1 Permission entry). A message at the top of the modal says, 'You can now check network connectivity with Reachability Analyzer', with a 'Run Reachability Analyzer' button and a close 'X' button.

2.1.3 Configure Security Inbounds

First click on the add rule and then click on the custom button and select the type HTTPS.

The screenshot shows the AWS Security Groups console. In the 'Basic details' section, a security group named 'MyWebServerGroup' is configured to allow SSH access ('Allows SSH access to developers'). Under 'Inbound rules', a custom TCP rule is defined for port 0, allowing traffic from 'Custom' sources. An 'Add rule' button is visible. The 'Outbound rules' section is currently empty.

The screenshot shows the AWS Security Groups console. In the 'Basic details' section, a security group is being configured to allow HTTPS traffic. A rule is being added for protocol TCP on port 443, originating from 'Custom' sources (0.0.0.0/8). The 'Source' dropdown is highlighted with a red box. The 'Outbound rules' section is currently empty.

2.1.4 Add IPv4 addresses with HTTPS

which are given below, and Save Rules.

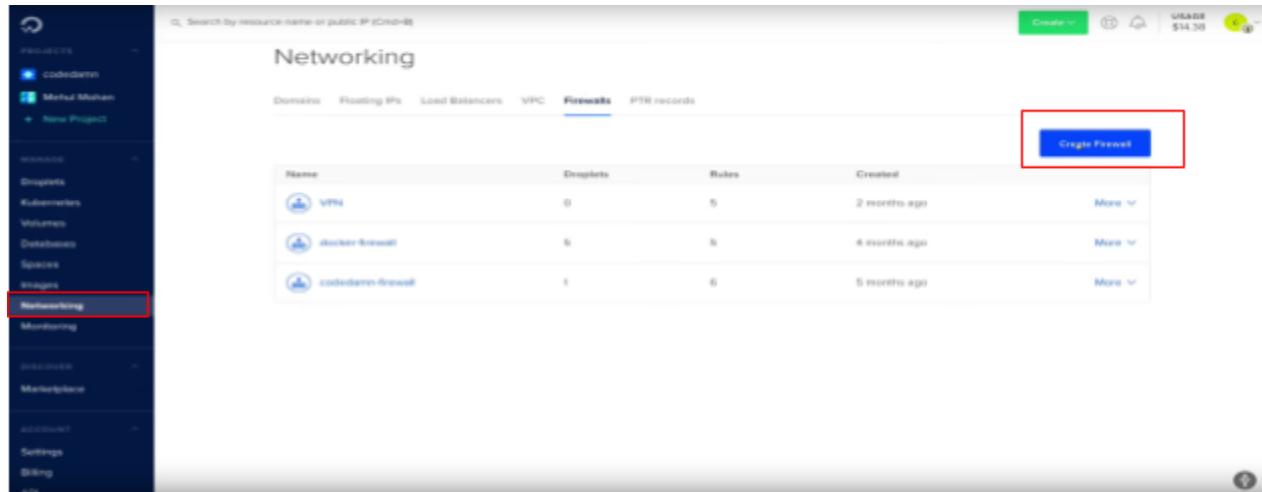
Inbound rules [Info](#)

Security group rule ID	Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
[REDACTED]	SSH	TCP	22	Custom	<input type="text"/> <input type="button" value="Delete"/>
	fill it with you ip				
[REDACTED]	HTTP	TCP	80	Custom	<input type="text"/> <input type="button" value="Delete"/>
				0.0.0.0/X	
[REDACTED]	HTTPS	TCP	443	Custom	<input type="text"/> <input type="button" value="Delete"/>
				0.0.0.0/X	
					<input type="button" value="Contact Support"/>

2.2 Security groups used in Digital Ocean.

2.2.1 Create Firewall

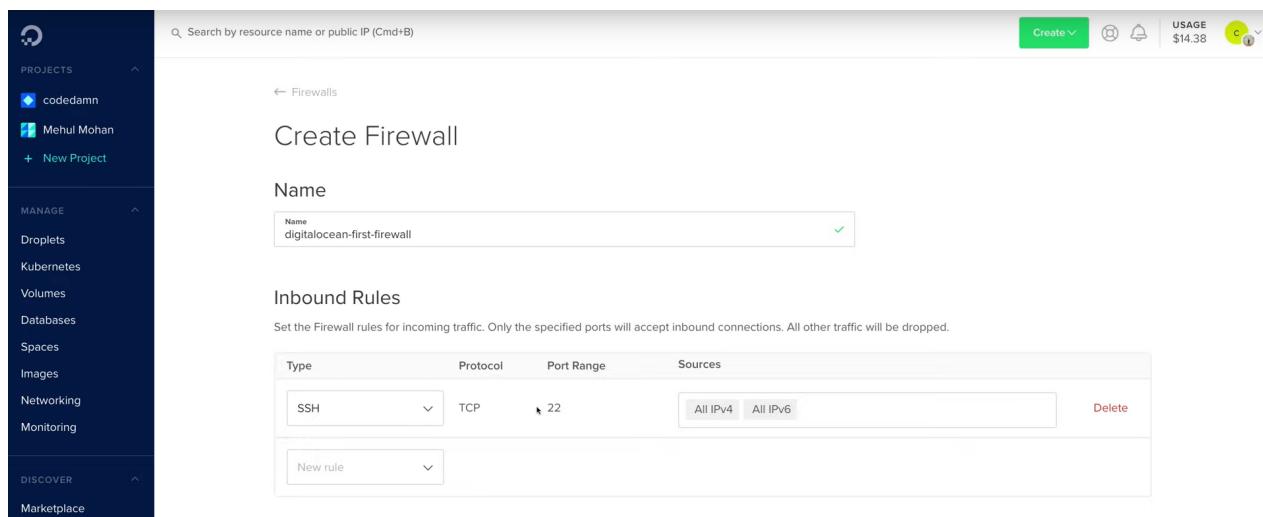
In the navigation pane select “Networking” then from tabs select “Firewalls” and then “Create Firewall”.



The screenshot shows the DigitalOcean control panel. On the left, there's a sidebar with 'PROJECTS' containing 'codedamm' and 'Mehul Mohan'. Below that are sections for 'Resources' (Droplets, Kubernetes, Volumes, Databases, Spaces, Images), 'Networking' (which is selected and highlighted with a red box), and 'Monitoring'. On the right, the main area is titled 'Networking' and has tabs for Domains, Floating IPs, Load Balancers, VPC, Firewalls (which is selected and underlined), and PTR records. A search bar at the top says 'Search by resource name or public IP (Cmd+B)'. Below the tabs is a table with columns: Name, Droplets, Rules, and Created. It lists three firewalls: 'VPN' (0 droplets, 5 rules, created 2 months ago), 'digitalocean-firewall' (5 droplets, 5 rules, created 4 months ago), and 'codedamm firewall' (1 droplet, 6 rules, created 5 months ago). At the top right of the table is a 'Create Firewall' button, which is also highlighted with a red box.

2.2.2 Adding Rules to the Firewall group

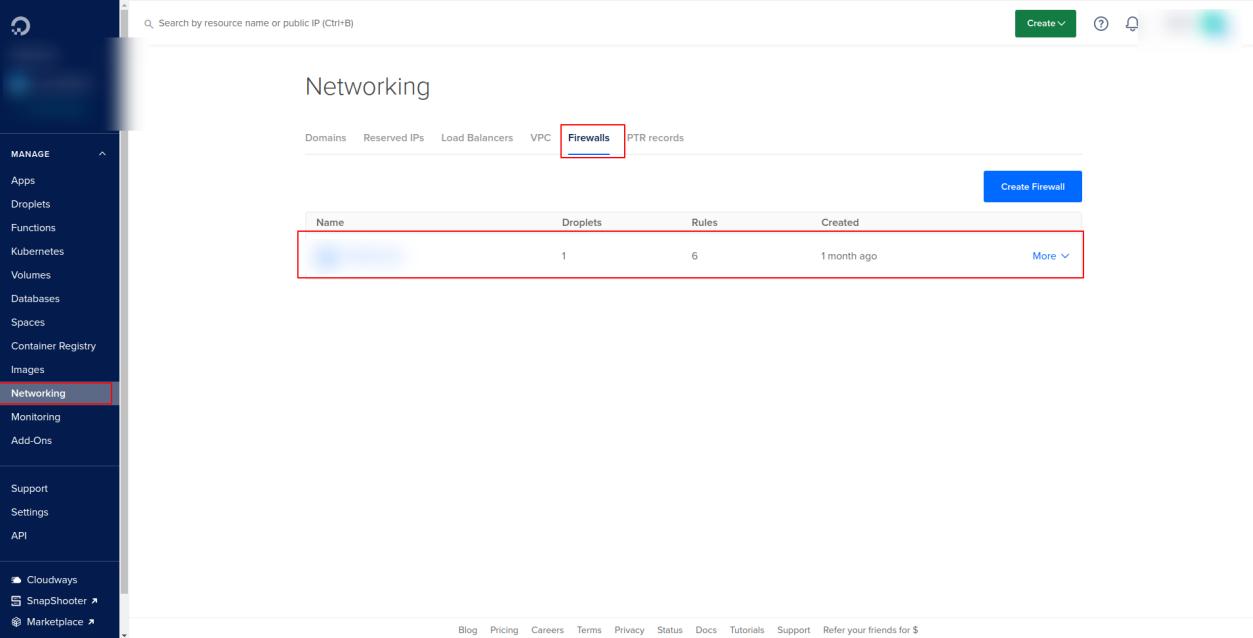
Similar to the previously mentioned groups in the AWS, use the same rules and add it here, refer ([2.1.4](#))



The screenshot shows the 'Create Firewall' page. The left sidebar is identical to the previous one. The main area has a title 'Create Firewall'. A 'Name' input field contains 'digitalocean-first-firewall'. Below it is a section titled 'Inbound Rules' with the sub-instruction 'Set the Firewall rules for incoming traffic. Only the specified ports will accept inbound connections. All other traffic will be dropped.' A table for 'Inbound Rules' is shown with columns: Type, Protocol, Port Range, and Sources. One rule is listed: Type 'SSH', Protocol 'TCP', Port Range '22', Sources 'All IPv4 All IPv6'. There's a 'Delete' link next to the rule. At the bottom of the table is a 'New rule' dropdown menu.

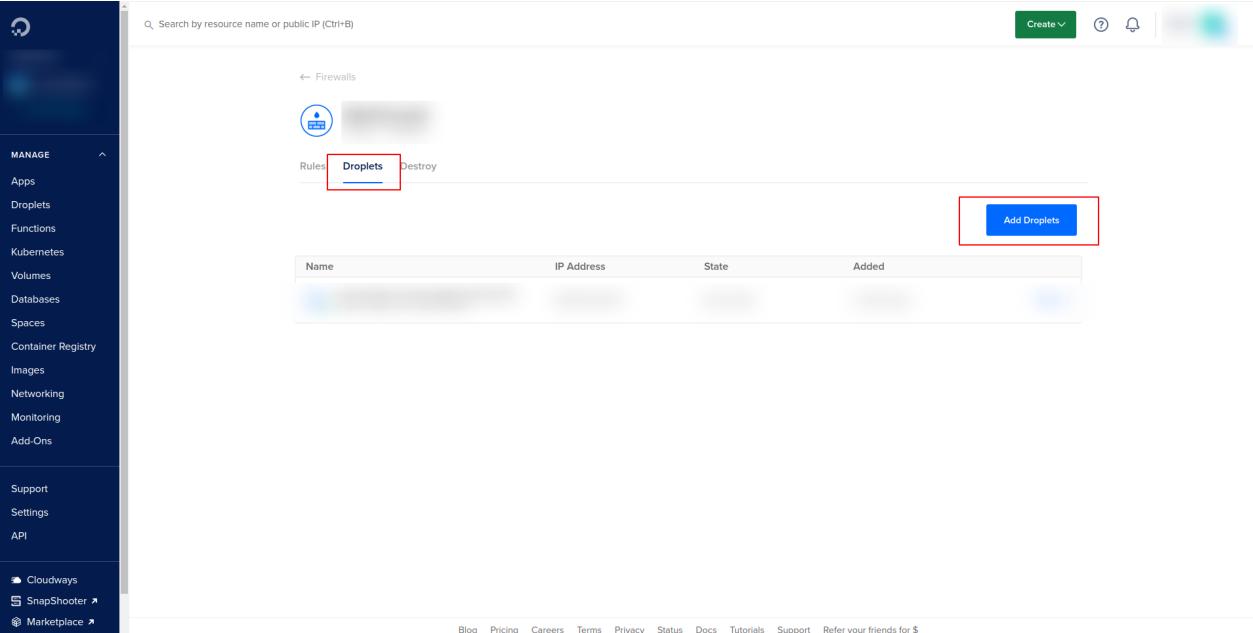
2.2.3 Connecting the Firewall Group With the Droplet

Select the firewall group you want to add to the droplet



The screenshot shows the Networking section of a cloud provider's interface. On the left, a sidebar lists various management options like Apps, Functions, Kubernetes, Volumes, Databases, Spaces, Container Registry, Images, and Networking (which is currently selected and highlighted with a red box). At the top, there are tabs for Domains, Reserved IPs, Load Balancers, VPC, Firewalls (also highlighted with a red box), and PTR records. A search bar at the top left says "Search by resource name or public IP (Ctrl+B)". On the right, a table displays a single firewall entry: Name (redacted), Droplets (1), Rules (6), and Created (1 month ago). A "Create Firewall" button is located at the top right of the table area.

After that, click on **Droplets** Tab, then after click on **Add Droplets** button and then select the droplet add it with the firewall group.



The screenshot shows the Droplets section of the same interface. The left sidebar includes Networking (selected), Apps, Functions, Kubernetes, Volumes, Databases, Spaces, Container Registry, Images, Monitoring, Add-Ons, Support, Settings, API, and Marketplace. The top navigation bar has tabs for Firewalls, Rules, and Droplets (highlighted with a red box). A "Create" button is at the top right. Below the tabs, a table lists droplets by Name, IP Address, State, and Added date. A prominent blue "Add Droplets" button is highlighted with a red box in the bottom right corner of the table area.

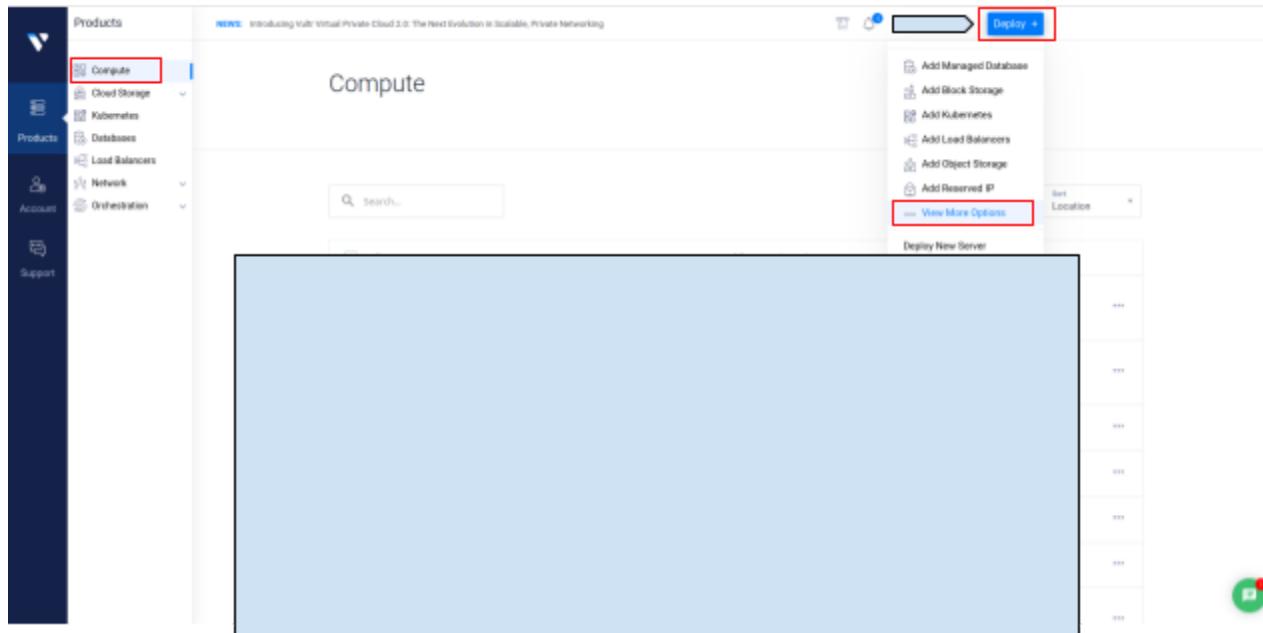
2.3 Security groups used in Vultr.

Here in Vultr there is firewall, which works same as security groups in AWS.

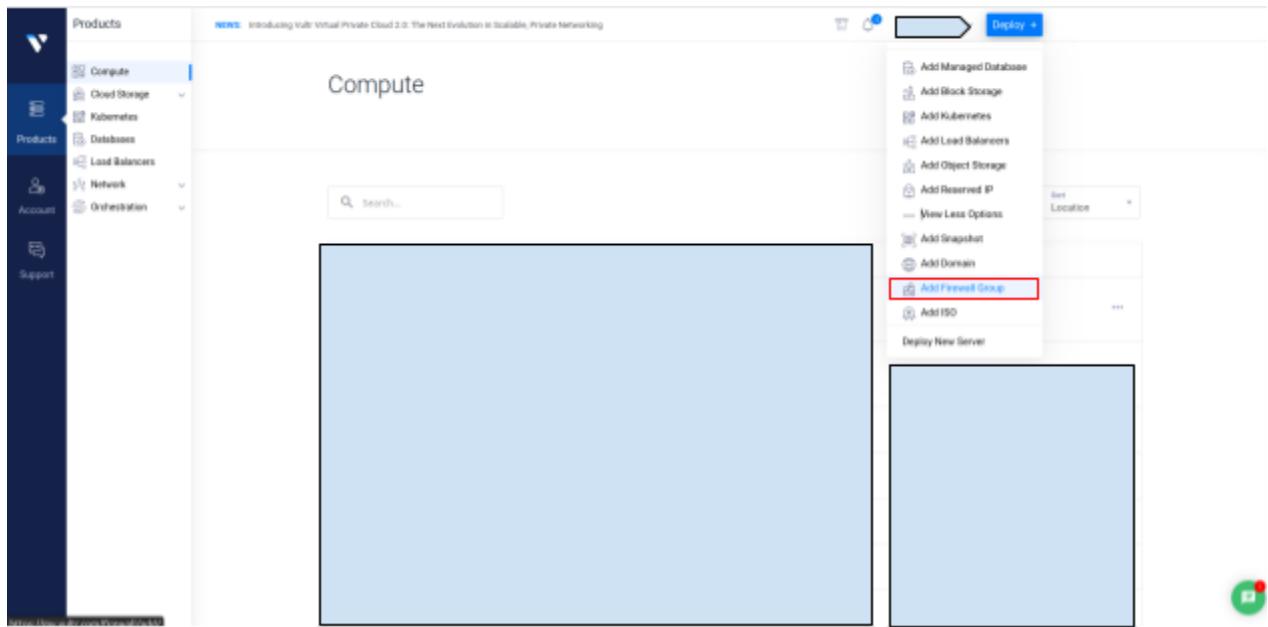
2.3.1 Navigate to Firewall

In the navigation pane, click on “**compute**” and hover on “**Deploy**”.

Then click on “**View More Options**”

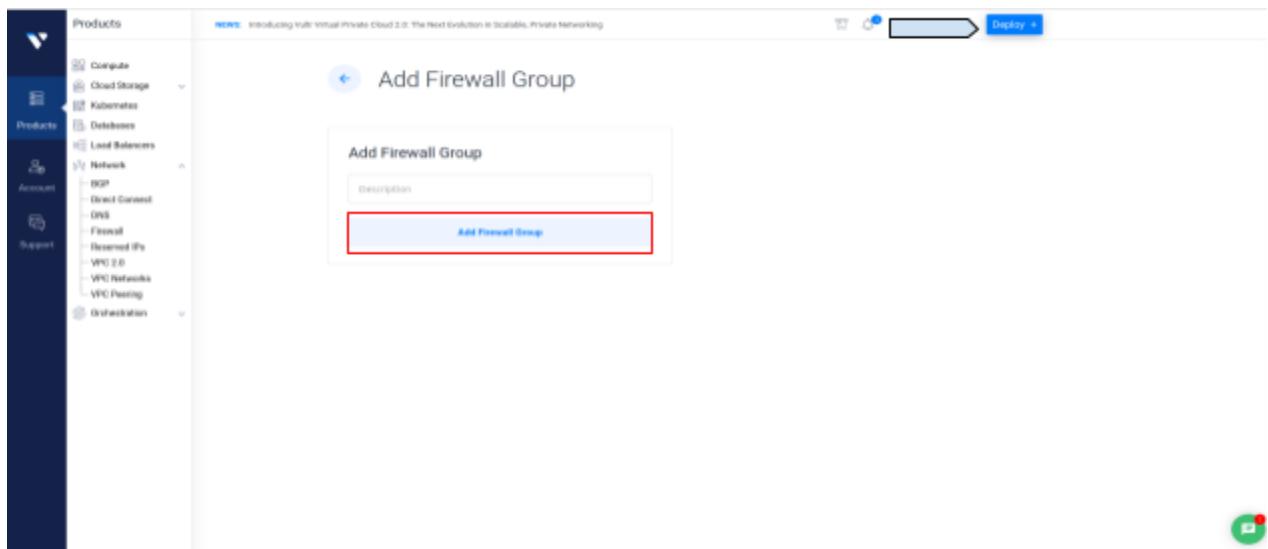


Then click on “**Add Firewall Group**”



2.3.2 Add Firewall group

Add description in the field and click “**Add Firewall Group**” button.



2.3.3 Add Rules in the group

Similar to the previously mentioned groups in the AWS, use the same rules and add it here, refer ([2.1.4](#))

The screenshot shows the 'Manage Firewall Group' page. On the left sidebar, under 'Network', 'Firewall' is selected. The main area displays a 'Description' field with 'TEST GROUP' and 'Group Rules' set to 0. Below this is a table for 'Inbound IPv4 Rules' with one row: Action (accept), Protocol (SSH), Port (or range) (22), Source (Anywhere), and Destination (0.0.0.0/0). A red box highlights the 'Add rule' button. The 'IPv6 Rules' section and 'Linked Instances' section are also visible.

2.3.4 Connect Firewall group with the instance

After the rules are properly added to the group, now we will connect the instance with the group. Click on the “**Linked Instances**” and then select the instance.

The screenshot shows the 'Manage Firewall Group' page with the 'Linked Instances' section highlighted by a red box. It lists one instance: '0'. The 'Server' column shows a placeholder 'Select instance...', the 'OS' column shows 'OS', the 'Location' column shows 'Location', and the 'Status' column shows 'Status'. A red box highlights the 'Link instances' button at the bottom right of the table.

3. Elastic IP / Reserved IP

What is Elastic Ip and why is it useful?

The Elastic IP address is a public **static IPv4 address** that is reachable from the Internet. Basically, Elastic IP addresses are used by AWS to manage its dynamic cloud computing services. Within the AWS infrastructure, customers have virtual private clouds (VPC, within the VPCs, users have instances. So when you launch an EC2 instance, you receive a Public IP address by which that instance is reachable from the internet. Once you stop that instance and restart the instance you get a new Public IP for the same instance. So it's basically a problem to connect your instance from the internet for not having a static IP. To overcome this problem, we attach an Elastic IP to an Instance that doesn't change after you stop/start the instance.

Follow accordingly:

i.e: [For AWS\(3.1\)](#) or [For Digital Ocean\(3.2\)](#) or [For Vultr\(3.3\)](#)

3.1 Elastic IP using AWS.

Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>

In the navigation pane, choose Network & Security, Elastic IPs.

3.1.1 Choose Allocate Elastic IP address.

The screenshot shows the AWS EC2 console with the 'Elastic IP addresses' page open. The left sidebar is collapsed, and the main content area displays a table of two allocated IP addresses. The table columns are Name, Allocated IPv4 add..., Type, Allocation ID, and Reverse DNS record. The first row has a Name of 'panel' and a Type of 'Public IP'. The second row has a Name of 'Backend' and a Type of 'Public IP'. At the top right of the table, there is a red box around the 'Actions' dropdown menu, and another red box highlights the 'Allocate Elastic IP address' button. The bottom of the page includes standard AWS footer links for Feedback, English (US), Copyright notice, Privacy, Terms, and Cookie preferences.

Name	Allocated IPv4 add...	Type	Allocation ID	Reverse DNS record
panel		Public IP	192.168.1.100	-
Backend	7	Public IP		-

3.1.2 Navigate to the page

For the Public IPv4 address pool, choose one of the following and then allocate

The screenshot shows the AWS Elastic IP Addresses service interface. At the top, there's a navigation bar with the AWS logo, 'Services' dropdown, a search bar containing 'Elastic IP addresses', and various icons. Below the header, the title 'Allocate Elastic IP address' is displayed with an 'Info' link. The main content area is divided into sections: 'Elastic IP address settings' (with a 'Info' link), 'Public IPv4 address pool' (radio button selected for 'Amazon's pool of IPv4 addresses'), 'Global static IP addresses' (with a note about AWS Global Accelerator), and 'Tags - optional' (with a note about tags). At the bottom right of the main content area, there are 'Cancel' and 'Allocate' buttons, with the 'Allocate' button being highlighted by a red rectangle. The footer of the page includes links for 'Feedback', 'English (US) ▾', '© 2021, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

3.1.3 Select IP

Now Select IP and click on allocate an elastic IP address.

The screenshot shows the AWS Elastic IP addresses page. On the left, there's a sidebar with 'Instances' and 'Images' sections. The main area displays a table titled 'Elastic IP addresses (1/2)' with one item: 'panel'. The 'panel' row is highlighted with a red box. In the 'Actions' column for this row, the 'Associate Elastic IP address' button is also highlighted with a red box. Below the table, the IP address '13.214.13.212' is shown, along with 'Summary' and 'Tags' tabs.

3.1.4 Select your project

Select your project Instance and allocate.

The screenshot shows the 'Associate Elastic IP address' dialog box. It starts with a message: 'Choose the instance or network interface to associate to this Elastic IP address'. Below this is a section for 'Elastic IP address:' with a dropdown menu. Under 'Resource type', the 'Instance' radio button is selected. A warning message states: 'If you associate an Elastic IP address to an instance that already has an Elastic IP address associated, this previously associated Elastic IP address will be disassociated but still allocated to your account.' Below this is a 'Instance' section with a dropdown menu containing '(panel) - running' and '(Backend) - running'. At the bottom, there's a 'Reassociation' section with a checkbox 'Allow this Elastic IP address to be reassigned'. The 'Associate' button at the bottom right is highlighted with a red box.

3.2 Elastic IP using DigitalOcean

3.2.1 Navigate to Reserved IPs

Follow Steps,

1. Networking
2. Reserved IPs
3. Search and Select the droplet
4. Assign Reserved IP

The reserved IP is now assigned with Droplet.

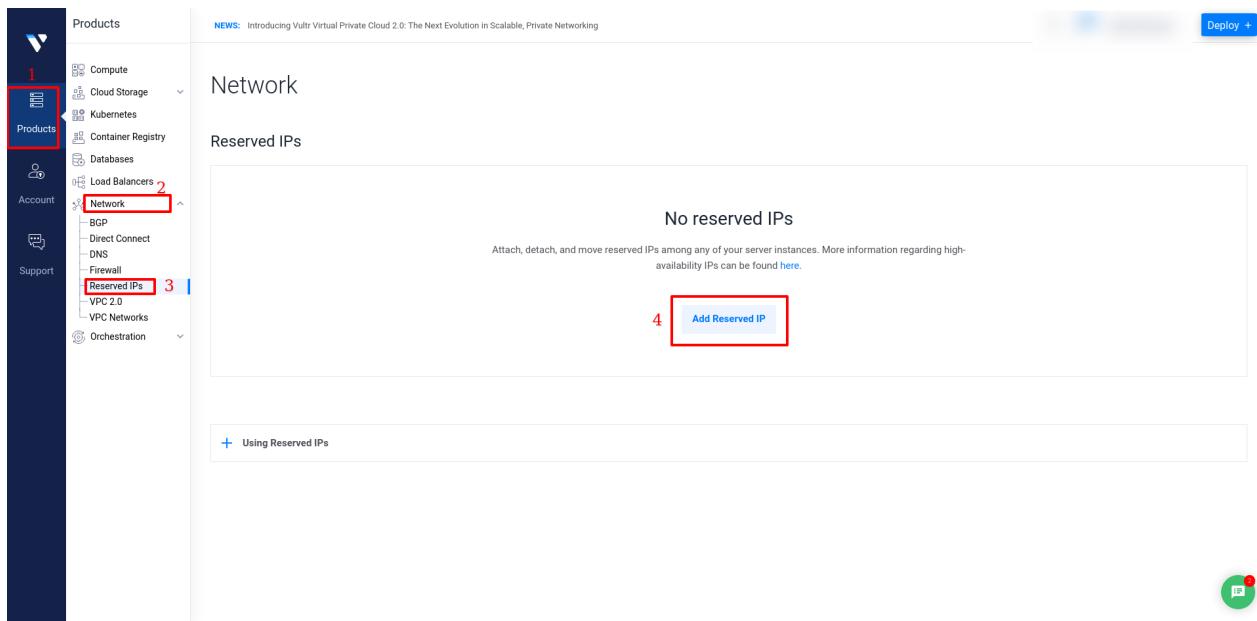
The screenshot shows the DigitalOcean control panel. On the left, a sidebar lists various services: Projects, Manage (with Apps, Functions, Kubernetes, Volumes, Databases, Spaces, Container Registry, Images, and Networking), Support, Settings, API, and Marketplace. The 'Networking' option under 'Manage' is highlighted with a red box and labeled '1'. At the top of the main content area, there's a search bar and several navigation links: Create, Help, Notifications, and a user profile. Below the search bar, the title 'Networking' is displayed, followed by tabs: Domains (highlighted with a red box and labeled '2'), Reserved IPs, Load Balancers, VPC, Firewalls, and PTR records. The 'Reserved IPs' tab is selected. A section titled 'Assign a Reserved IP' contains a note about what a Reserved IP is and a link to reserve one for a specific datacenter. It features two input fields: 'Search for a Droplet' (labeled '3') and 'Assign Reserved IP' (labeled '4'). Below this is a table titled 'Reserved IPs' with columns 'Reserved IP' and 'Assigned To'. One row in the table is highlighted with a red box. At the bottom of the page, there are links for Blog, Pricing, Careers, Terms, Privacy, Status, Docs, Tutorials, Support, and Refer your friends for \$.

3.3 Elastic IP using Vultr

3.3.1 Navigate to Reserved IPs

Follow Steps,

1. Products
2. Network
3. Reserved IPs
4. Add Reserved IP



5. Please refer [here](#) for reference.

4. Create Storage(Optional).

If we are using the bucket facility, we need to upload all the static images from the project to the bucket manually.

Note: The process of drag and drop all the files needs to be completed after the creation of the bucket and after the completion of the installation process on the server.

(i) For AWS S3

- Navigate to the bucket.
 1. Search for **S3**
 2. Select S3

The screenshot shows the AWS Management Console search interface. A search bar at the top contains the query 's3'. Below the search bar, a sidebar lists various AWS services and features. The main search results area displays a list of services under the heading 'Services'. The first item in this list, 'S3 Scalable Storage in the Cloud', is highlighted with a red box. To its right, there is a separate 'Welcome to AWS' panel containing three informational cards: 'Getting started with AWS', 'Training and certification', and 'What's new with AWS?'. The entire interface has a dark-themed background.

- Open the Bucket

The screenshot shows the Amazon S3 service dashboard. On the left, a navigation sidebar includes links for Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Storage Lens, Dashboards, AWS Organizations settings, Feature spotlight, and AWS Marketplace for S3. The main content area is titled 'Amazon S3' and features an 'Account snapshot' section with a 'View Storage Lens dashboard' button. Below this is a 'Buckets (1) Info' section. A single bucket is listed in a table with columns for Name, AWS Region, Access, and Creation date. The 'Name' column shows a blue link with a red box around it, and a red arrow points upwards from the text 'Open the Bucket' located below the table. The interface uses a light-colored background.

- Upload all the files in the **data** folder manually.
 - Navigate to the data folder
 - **backend/server/data**
 - Select all the files and now **drag and drop** it in the bucket.
 1. Objects
 2. Upload

Amazon S3 > Buckets >

Objects (10)

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Actions ▾ **Create folder** **Upload** 2

Find objects by prefix Show versions

Name	Type	Last modified	Size	Storage class
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-
Folder	Folder	-	-	-

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

3. Drop all the files here

Upload [Info](#)

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose Add files or Add folder.

Files and folders (0)

All files and folders in this table will be uploaded.

Destination

Destination details

Bucket settings that impact new objects stored in the specified destination.

Permissions

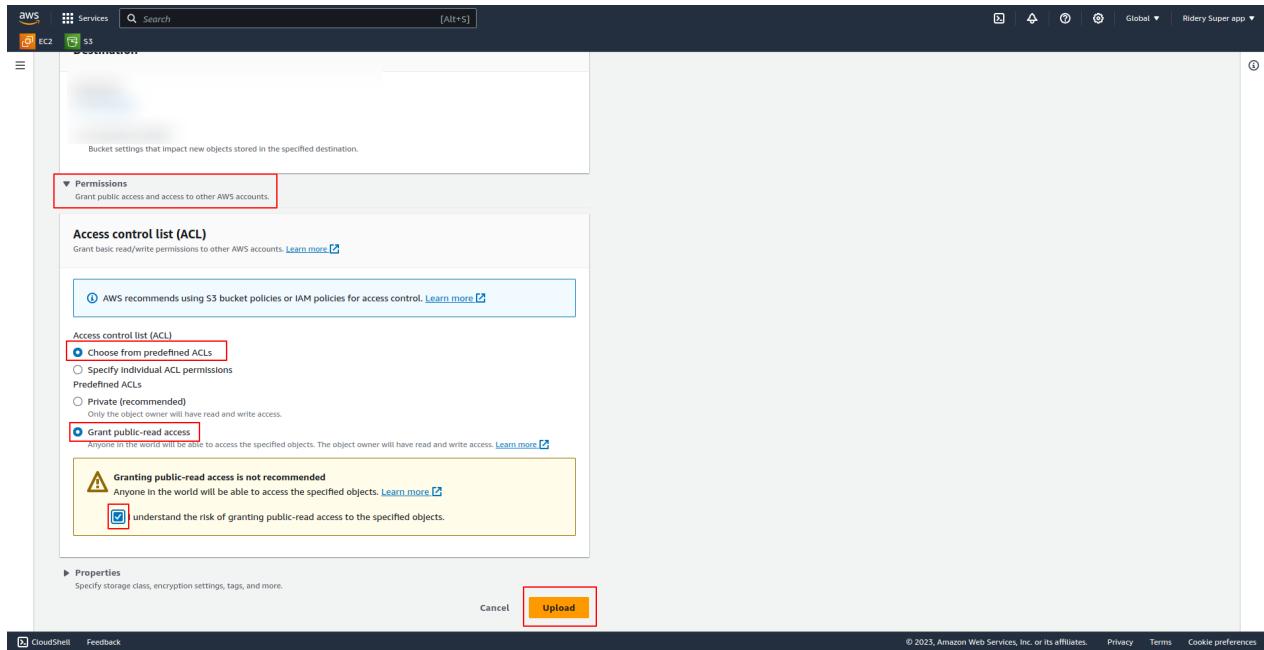
Grant public access and access to other AWS accounts.

Properties

Specify storage class, encryption settings, tags, and more.

CloudShell Feedback © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

4. Change the permission and click upload and that's it.



(ii) For Digital-Ocean Spaces

Navigate to the Spaces

And refer [here](#).

And also change the permission of the files after uploading all the files on the spaces.

[\(4.1\)For AWS S3](#) or [\(4.2\)For Digital-Ocean Spaces](#)

4.1 Using AWS S3 Bucket

4.1.1 What is S3 Bucket and why do we use it?

An Amazon S3 bucket is a public cloud storage resource available in Amazon Web Services' (AWS) Simple Storage Service (S3), an object storage offering. Amazon S3 buckets, which are similar to file folders, store objects, which consist of data and its descriptive metadata.

Amazon S3 is a program that's built to store, protect, and retrieve data from "buckets" at anytime from anywhere on any device. ... Use cases include websites, mobile apps, archiving, data backups and restorations, IoT devices, enterprise application storage, and

providing the underlying storage layer for your data lake.

In short, using the AWS S3 bucket it can help us to keep our server clean or we can say that the space will be occupied less, as all the files will be stored on S3 bucket.

4.1.2 Click on s3 from all services

The screenshot shows the AWS Management Console with the 'Services' tab selected. The 'Storage' section is expanded, and 'S3' is highlighted with a red box. Other services listed under Storage include EFS, FSx, S3 Glacier, Storage Gateway, AWS Backup, AWS Elastic Disaster Recovery, Database (RDS, DynamoDB, ElastiCache, Neptune, Amazon QLDB, Amazon DocumentDB, Amazon Keyspaces), and Media Services (Kinesis Video Streams, MediaConnect, MediaConvert). The rest of the page includes sections for Free AWS Training, AWS Cloud Training, AWS Machine Learning Training, and a feedback form.

4.1.3 Click on Create bucket

The screenshot shows the AWS S3 console. On the left, there's a sidebar with links like 'Buckets', 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens' (with 'Dashboards' and 'AWS Organizations settings'), 'Feature spotlight', and 'AWS Marketplace for S3'. The main area has a blue header bar with a message: 'We're continuing to improve the S3 console to make it faster and easier to use. If you have feedback on the updated experience, choose Provide feedback.' Below this is an 'Account snapshot' section with a 'View Storage Lens dashboard' button. The main content area is titled 'Buckets (1) info' and shows a table with one row. The table has columns for 'Name' (with an upward arrow), 'AWS Region' (with a downward arrow), 'Access' (with a downward arrow), and 'Creation date' (with a downward arrow). The first row shows a single bucket. At the top of this section are buttons for 'Create bucket' (highlighted with a red box), 'Copy ARN', 'Empty', and 'Delete'. Below the table is a search bar with 'Find buckets by name' and navigation controls (back, forward, search icon).

4.1.4 Add bucket name and select region according to country.

- In Bucket name, enter a unique name for your bucket.
- Choose a Region close to you to minimize latency and costs and address regulatory requirements. Objects stored in a Region never leave that Region unless you explicitly transfer them to another Region.

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner preferred
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

Object writer
The object writer remains the object owner.

If you want to enforce object ownership for new objects only, your bucket policy must specify that the bucket-owner-full-control canned ACL is required for object uploads. [Learn more](#)

4.1.5 Unblock public access.

- We need to disable the block off public access
- Accept the Acknowledgement because our bucket needs the public access.

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that no public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Block public access to buckets and objects granted through new access control lists (ACLs)
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

Block public access to buckets and objects granted through any access control lists (ACLs)
S3 will ignore all ACLs that grant public access to buckets and objects.

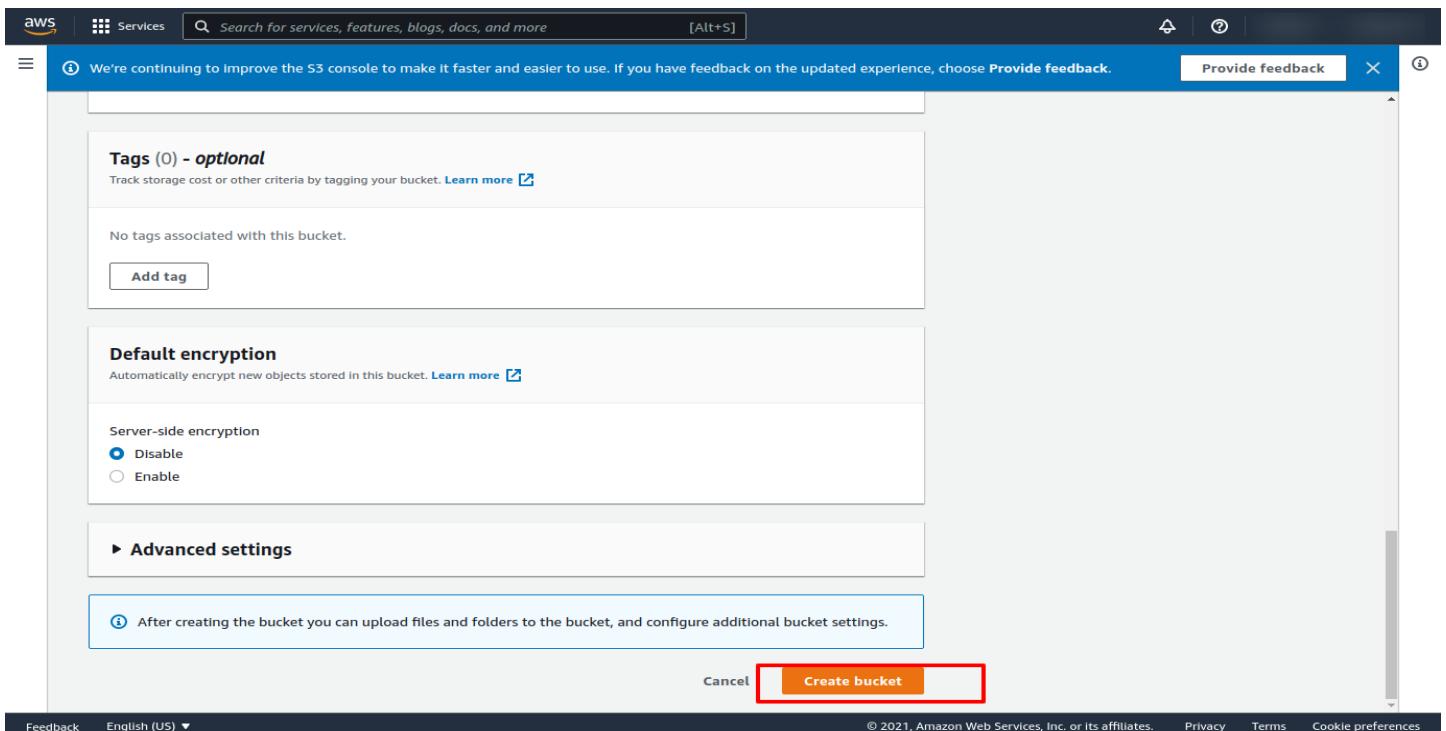
Block public access to buckets and objects granted through new public bucket or access point policies
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

Block public and cross-account access to buckets and objects through any public bucket or access point policies
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

⚠ Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

4.1.6 Now Click on Create Bucket.



4.1.7 How to get an S3 bucket image URL

- 1) Upload image in s3 bucket and select that image and open properties.
- 2) In object overview copy that Object URI.
- 3) The **Image Url** will look like - "http://your-bucket.s3-website-us-east-1.amazonaws.com"

(a) Update the **image url** value in the following listed files after the installation process has completed using bash file.

- a. admin-panel/src/environments/environment.prod.ts
- b. admin-panel/src/environments/environment.ts
- c. corporate-panel/src/environments/environment.prod.ts
- d. corporate-panel/src/environments/environment.ts
- e. dispatcher_panel/src/environments/environment.prod.ts
- f. dispatcher_panel/src/environments/environment.ts
- g. driver-panel/src/environments/environment.prod.ts
- h. driver-panel/src/environments/environment.ts

- i. hotel_panel/src/environments/environment.prod.ts
- j. hotel_panel/src/environments/environment.ts
- k. partner-panel/src/environments/environment.prod.ts
- l. partner-panel/src/environments/environment.ts
- m. user-panel/src/environments/environment.prod.ts
- n. user-panel/src/environments/environment.ts
- o. hub-panel/src/environments/environment.prod.ts
- p. hub-panel/src/environments/environment.ts

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with various navigation links like Buckets, Storage Lens, and AWS Marketplace for S3. The main area is titled 'Object overview' for an object named 'Owv.jpg'. The object details include its owner ('developers'), AWS Region, last modified date ('December 7, 2021, 16:29:33 (UTC+05:30)'), size ('9.3 KB'), type ('jpg'), and key (''). On the right, there are sections for 'S3 URI', 'Amazon Resource Name (ARN)', and 'Entity tag (Etag)'. The 'Object URL' section is highlighted with a red box and contains a link starting with 'https://...omo_l...'. At the bottom of the page, there are links for Feedback, English (US), and Copyright information.

4.1.8 S3 bucket Access key and secret key download and save it in the database.

Open the Robo-3t or MongoDb compass, then navigate to DB->setting collection->update the below highlighted keys.

- 1) Save this key & add this to '**access_key_id**' & '**secret_key_id**'.
- 2) Add bucket name '**aws_bucket_name**'.
- 3) Set the boolean of '**is_use_aws_bucket**' to **true**.
- 4) Add the '**image_base_url**'.

The screenshot shows the Robo 3T interface. On the left, a tree view displays a hierarchical list of MongoDB collections, including admin_notifications, admins, airports_to_cities, bank_details, cancellation_reasons, change_logs, cities, city_to_cities, city_types, cityzones, companies, countries, dispatchers, documents, email_details, export_histories, find_provider_logs, tokens, hotels, increments, information, messages, mass_notifications, partner_vehicle_documents, partner_weekly_earnings, partners, payment_transactions, promo_codes, provider_analytics, provider_daily_earnings, provider_documents, provider_vehicle_documents, provider_weekly_earnings, providers, redemptions, reviews, sms_details, transfer_histories, trip_histories, trip_locations, trip_services, and trips. A specific collection, 'settings', is selected. On the right, a table titled 'Welcome' shows the results of the query 'db.getCollection('settings').find({})'. The table has columns for 'Key', 'Value', and 'Type'. One row is expanded to show its sub-fields.

The screenshot shows the AWS Identity and Access Management (IAM) service. The left sidebar shows the IAM navigation menu with sections like Dashboard, Access management, Access reports, and Access analyzer. The main content area is titled 'Your Security Credentials' and lists various credential types: Password, Multi-factor authentication (MFA), and Access keys (access key ID and secret access key). A callout box highlights the 'Access keys' section with a red border and the number '2'. A red box highlights the 'Create New Access Key' button at the bottom of the access keys table. Another red box highlights the 'Security credentials' link in the top right corner of the main content area. The top navigation bar includes a search bar, a 'Global' dropdown, and a sign-out button.

Created	Access Key ID	Last Used	Last Used Region	Last Used Service	Status	Actions
Dec 2nd 2021	AJXXXXXXXXXXXXXX	2021-12-14 12:44 UTC+0530		ses	Active	Make Inactive Delete

4.2 Using Digital-Ocean Spaces

The Digital-Ocean Spaces bucket is similar to the AWS S3 bucket.

4.2.1 Create Spaces Bucket

The screenshot shows the DigitalOcean control panel. On the left, a sidebar menu is open with various options like Projects, Manage, and Spaces. The 'Spaces' option is highlighted with a red box and the number '1'. In the main content area, there's a section titled 'Spaces Object Storage' with a sub-section 'Store and deliver vast amounts of content'. A large blue button labeled 'Create a Spaces Bucket' is centered in this section and is also highlighted with a red box and the number '2'. Below this, there's a 'Learn more about Spaces Object Storage' section with links to 'PRODUCT DOCS' (Spaces Object Storage overview), 'API' (Spaces Object Storage API Docs), and 'TUTORIALS' (Spaces Object Storage community discussion).

4.2.2 Configure the Spaces Bucket

1. **Choose a datacenter region.** The datacenter region you choose will also become part of a bucket's endpoint URL. See [regional availability for Spaces](#) for more information on the available options.
2. Optionally, enable the [Spaces CDN \(Content Delivery Network\)](#). If you click **Enable CDN**, you can customize the **Edge Cache TTL**, which is the amount of time the edge servers will cache your content.
3. Choose to **restrict file listing** or **enable file listing**. The visibility of a bucket's file listing has no effect on the visibility of the files themselves. You can [change the file listing visibility](#) at any time after creation.
4. **Choose a unique name for your bucket.** The name of the bucket makes up part of its endpoint URL and cannot be changed once it is created.

For reference [click me](#)

The screenshot shows the 'Create a Spaces Bucket' page in the DigitalOcean control panel. The left sidebar has 'Spaces' selected under the 'MANAGE' section. The main area starts with a search bar and a 'Create' button. Below it, a section titled 'Choose a datacenter region' shows 'Frankfurt - Datacenter 1 - FRA1' with '2 resources'. A dropdown menu for 'Additional datacenter regions' is open. The next section, 'Content Delivery Network (CDN)', includes a note about global edge caching and a checkbox for 'Enable CDN'. The 'Finalize and create' section contains a field for 'Choose a unique Spaces Bucket name*' and a 'Select a project' dropdown set to 'e-go-platform'. To the right, there's a circular icon with a grid pattern and a list of benefits: Predictable pricing, Built-in CDN, Unlimited Spaces buckets and uploads, Use your own custom subdomain, and Easily integrate S3-compatible. A link 'Check out all Spaces has to offer' is at the bottom.

Once you get all the keys and details follow the similar steps mentioned [here](#).

5. Pointing Domain name, and SSL certificate

Now as our code is running on the server, we can't open our application from search engines or web browsers using domain names.

For that, we need to setup a few things and that are:

1. Domain Name
2. SSL certificate

5.1 Domain Name(DNS)

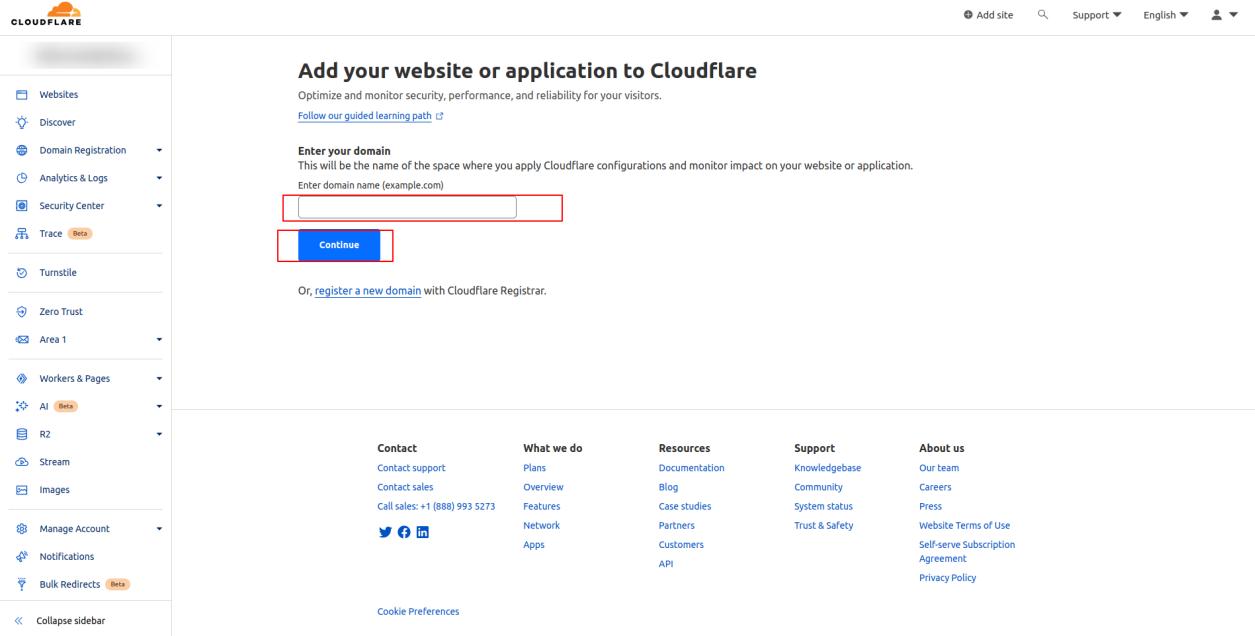
Follow the steps to add the domain to the cloudflare and configure the domain and subdomains.

For Domain configuration we are using Cloudflare (<https://www.cloudflare.com>). Register your account and go to the DNS section and click Add Record. On the given input field in the type section select A (A records) and in name enter @ (refers to main domain and also it will be the user panel) and in IPv4 enter the IP address of our server (the elastic Ip which we generated previously or the IP of the server), then click Save. For subdomains click add new record again and in the type section select A (A records) and in the name section enter admin and in IPv4 enter IP address. Similarly, add 10 more subdomains for api, history, payment, notification, driver, corporate, dispatcher, partner, hotel and hub.

And now we are done with our Domain pointing.

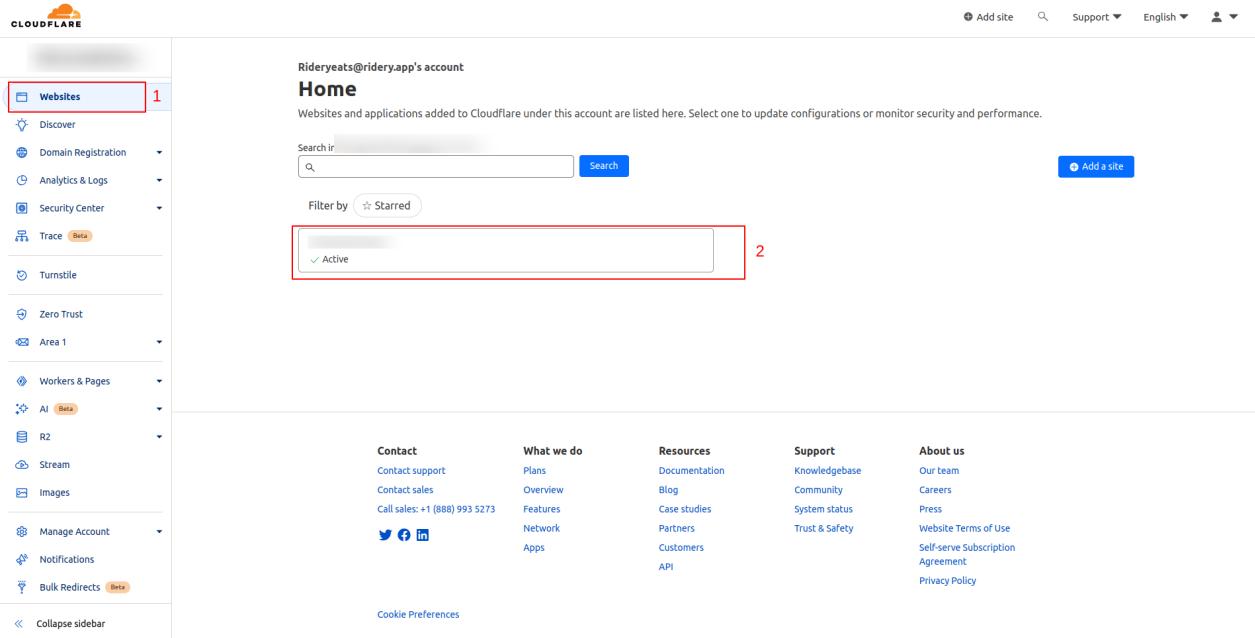
Follow the steps as below highlighted.

The screenshot shows the Cloudflare dashboard interface. On the left, there is a sidebar with various menu items: Websites (highlighted with a red box), Discover, Domain Registration, Analytics & Logs, Security Center, Trace (Beta), Turnstile, Zero Trust, Area 1, Workers & Pages, AI (Beta), R2, Stream, Images, Manage Account, Notifications, and Bulk Redirects (Beta). Below the sidebar, there is a 'Collapse sidebar' button. The main content area has a header 'Rideryeats@ridery.app's account' and 'Home'. It displays a list of websites under 'Websites and applications added to Cloudflare under this account are listed here. Select one to update configurations or monitor security and performance.' There is a 'Search in' input field and a 'Search' button. To the right of the search bar is a 'Filter by' dropdown set to 'Starred'. On the far right of the main content area is a blue 'Add a site' button, which is also highlighted with a red box. At the bottom of the page, there are links for Contact support, Contact sales, Call sales: +1 (888) 993 5273, and social media icons for Twitter, Facebook, and LinkedIn. There are also links for Plans, Overview, Features, Network, Apps, Documentation, Blog, Case studies, Partners, Customers, API, Knowledgebase, Community, System status, Press, Trust & Safety, and API. On the far right, there are links for Our team, Careers, Press, Website Terms of Use, Self-serve Subscription Agreement, and Privacy Policy. At the very bottom, there is a 'Cookie Preferences' link.



The screenshot shows the Cloudflare dashboard with the sidebar collapsed. The main heading is "Add your website or application to Cloudflare". It includes a note to optimize and monitor security, performance, and reliability. A red box highlights the "Enter your domain" input field and the "Continue" button below it. Below the input field, there's a note about registering a new domain with Cloudflare Registrar.

After adding the **domain** and once active follow the steps.



The screenshot shows the Cloudflare dashboard with the sidebar collapsed. The "Websites" menu item is selected, indicated by a red box labeled "1". The main area shows the "Home" section for "Rideryeats@ridery.app's account". A red box labeled "2" highlights the "Active" filter applied to the list of websites. The sidebar contains various other menu items like Discover, Domain Registration, Analytics & Logs, Security Center, Trace, Turnstile, Zero Trust, Area 1, Workers & Pages, AI, R2, Stream, Images, Manage Account, Notifications, and Bulk Redirects.

The screenshot shows the Cloudflare DNS management interface. On the left sidebar, under the 'DNS' section, 'Records' is selected. The main content area is titled 'Records' and contains a table of DNS records. The table has columns for Type, Name, Content, Proxy status, TTL, and Actions. There are six records listed:

Type	Name	Content	Proxy status	TTL	Actions
A	admin		Proxied	Auto	Edit
A	api		Proxied	Auto	Edit
A	merchant		Proxied	Auto	Edit
A			Proxied	Auto	Edit
CNAME			Proxied	Auto	Edit
CNAME			Proxied	Auto	Edit

Moreover we need to replace these nameservers on the domain provider platform with the nameservers provided by cloudflare.

This screenshot shows the same Cloudflare interface, but with red arrows pointing to the 'Value' field of two 'NS' records. The 'Value' field is highlighted with a green box. This indicates where the user needs to enter their own custom nameserver information.

The changes will take few minutes or even more, keep checking the overview tab.

5.2 SSL certificate

Now we have a domain pointed to our IP address, let's create an SSL certificate to secure our domain. Let's head towards Cloudflare for SSL certificates (<https://www.cloudflare.com>). Head towards the SSL/TLS section(SSL\TLS > Overview) and select your plan(Preferred full strict). For certificates go to **SSL/TLS -> Origin Server** to generate certificates. Please check screenshots for References.

The screenshot shows the Cloudflare SSL/TLS Overview page. On the left sidebar, the 'SSL/TLS' section is expanded, and the 'Overview' tab is selected. A modal window titled 'Advanced Certificate Manager' is open, displaying the message: 'Your SSL/TLS encryption mode is Full (strict)'. It includes a diagram showing traffic flow from a 'Browser' to 'Cloudflare' and then to an 'Origin Server'. To the right of the diagram, there are four options: 'Off (not secure)', 'Flexible', 'Full', and 'Full (strict)'. The 'Full (strict)' option is selected and highlighted with a red box. Below the diagram, a note states: 'Encrypts end-to-end, but requires a trusted CA or Cloudflare Origin CA certificate on the server'. At the bottom of the modal, there is a link to 'Create a Configuration Rule to customize these settings by hostname.'

Store the **private** and **origin pem** files at a safe place(it will be required in further steps) as these are SSL certificates.

The screenshot shows the Cloudflare Origin Server page. On the left sidebar, the 'SSL/TLS' section is expanded, and the 'Origin Server' tab is selected. A modal window titled 'Advanced Certificate Manager' is open, displaying the message: 'Origin Certificates'. It includes a note: 'Generate a free TLS certificate signed by Cloudflare to install on your origin server.' A large blue button labeled 'Create Certificate' is highlighted with a red box. Below the button, it says: 'Origin Certificates are only valid for encryption between Cloudflare and your origin server.' At the bottom of the modal, there are sections for 'Hosts' and 'Expires On', and a link to 'Help'.

6 Establishing SSH connection

Establishing SSH connection to connect server and installing environment for our code to run.

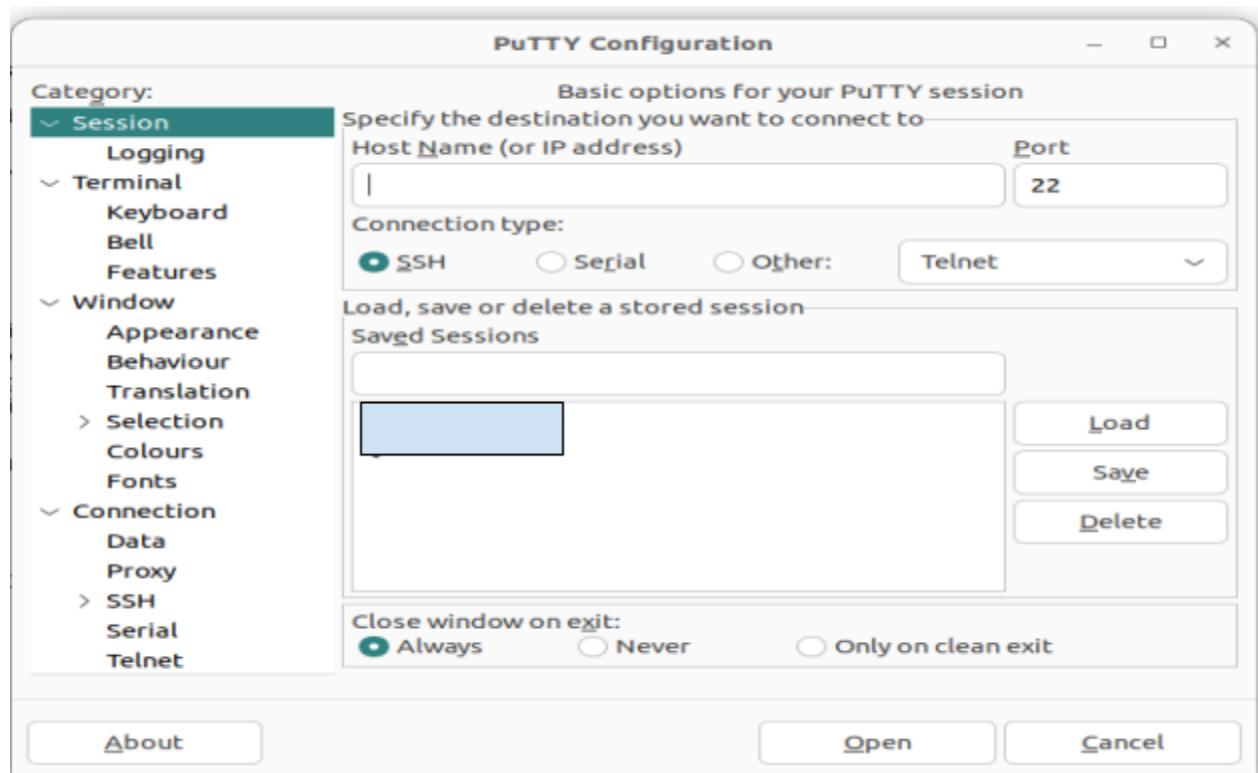
We will be Using putty for establishing SSH connection between our system and server.

What is **PuTTY** ?

PuTTY is a free and open-source terminal emulator, serial console, and network file transfer application. It supports several network protocols, including SCP, SSH, Telnet, rlogin, and raw socket connection. It can also connect to a serial port. The name "PuTTY" has no official meaning.in short, it is used for connecting the ubuntu server in any operating system for application installation on the domain server.

6.1 Open PuTTY or Terminal

Follow the steps to connect the server using Putty or with a local Terminal.

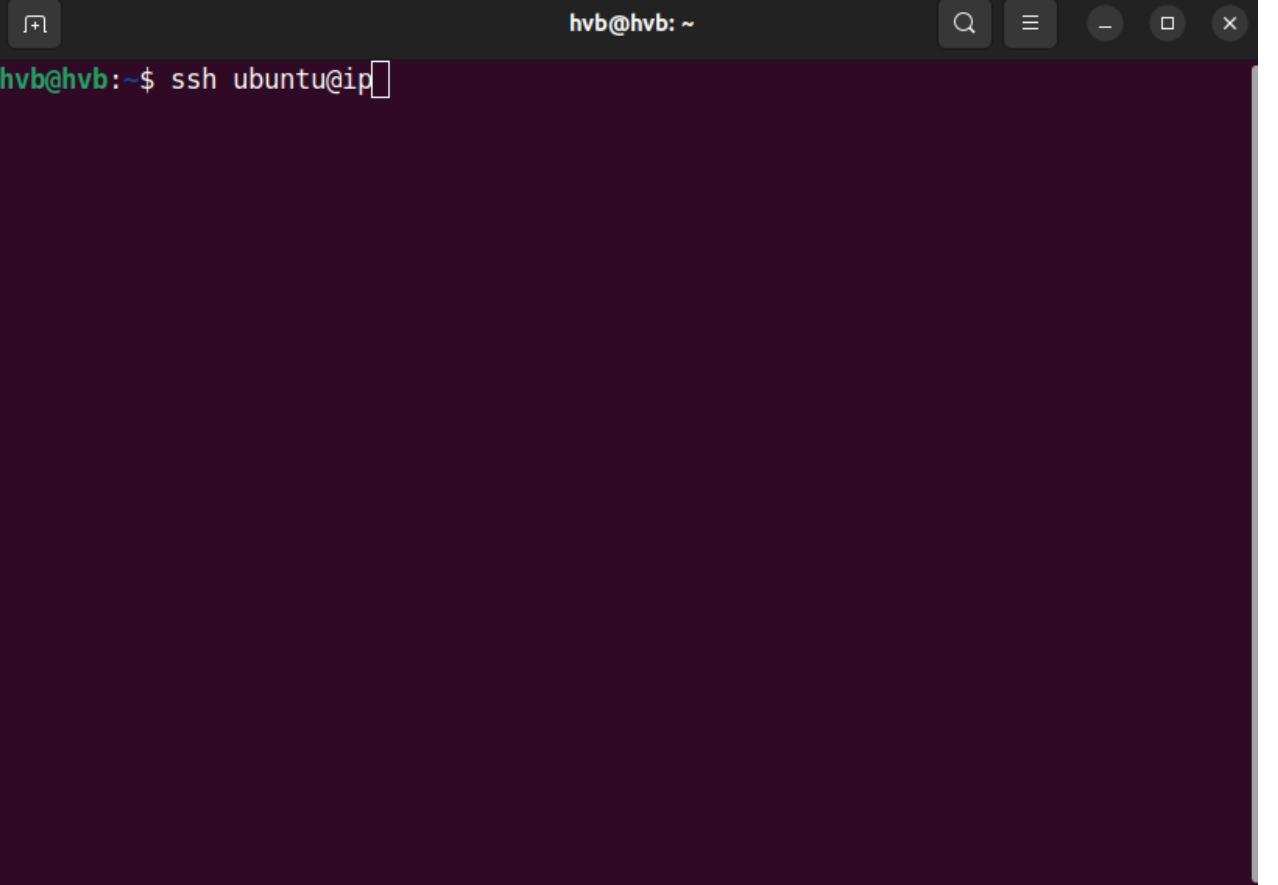


6.2 Connect with Password.

- Connect using local Terminal

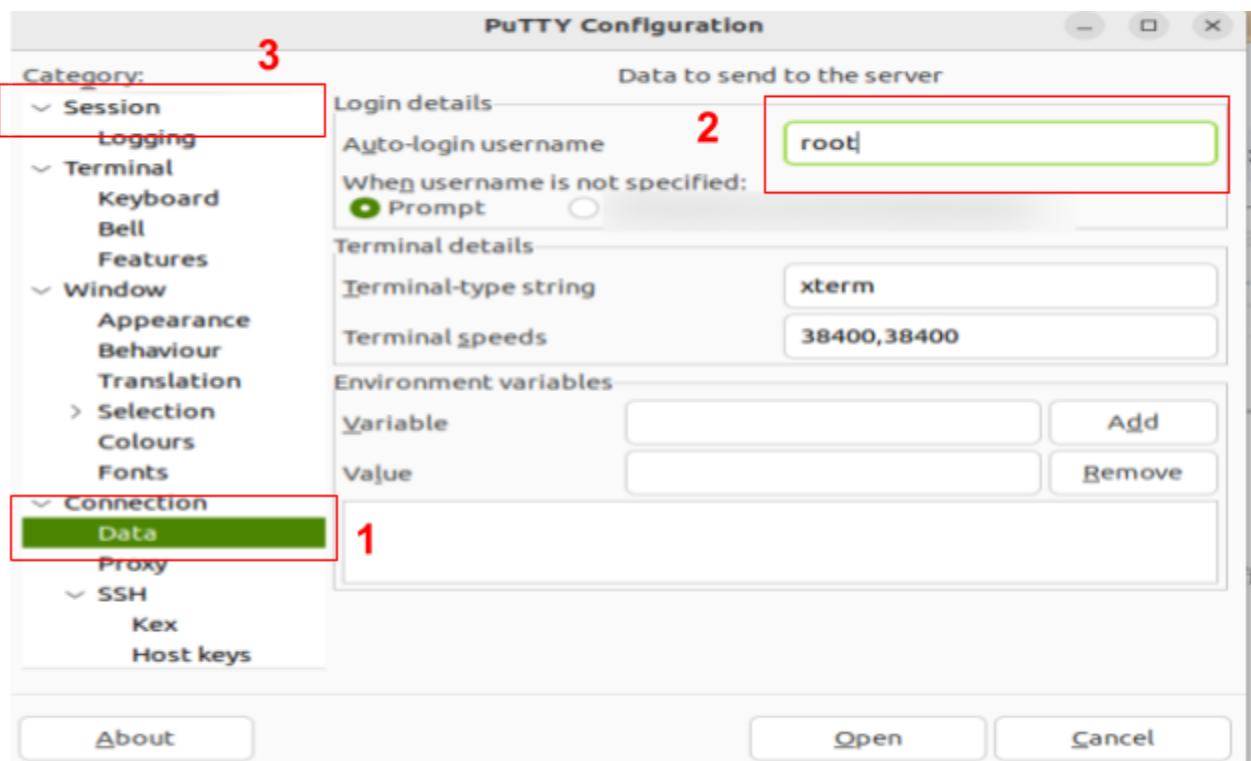
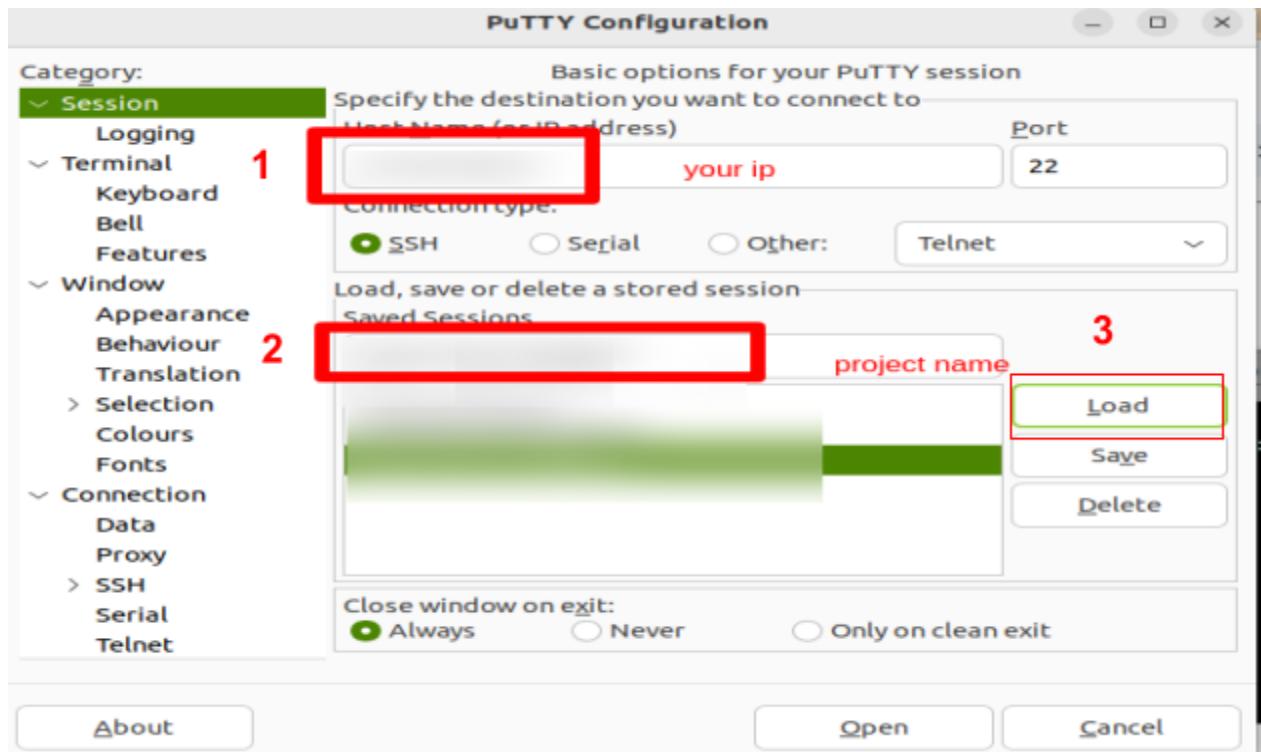
Root : ssh username@ip

Replace username with **ubuntu** or **root** accordingly.
Replace ip with actual Ip address of the server.

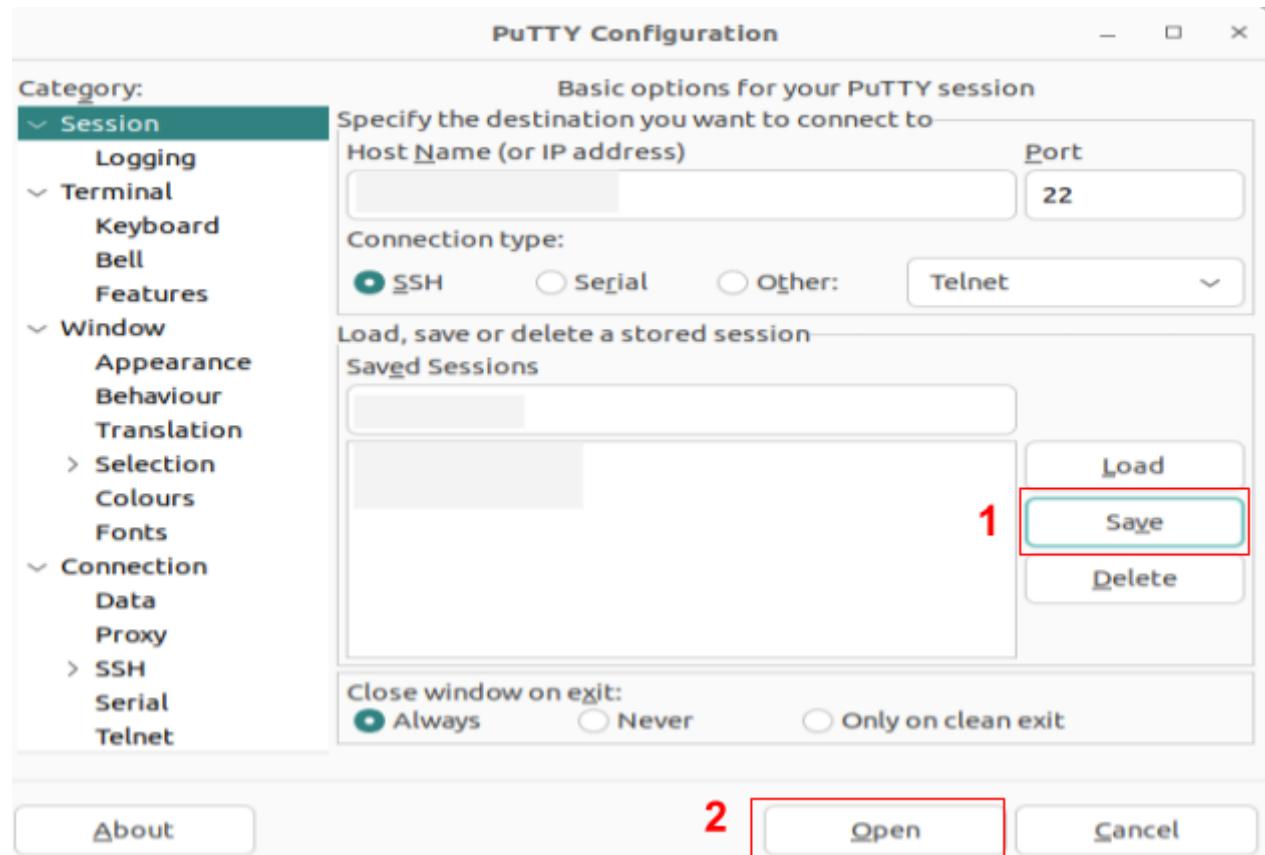


A screenshot of a terminal window titled "hvib@hvib: ~". The window has a dark background and light-colored text. In the top left corner, there is a small icon with a plus sign. On the right side, there are several window control buttons: a magnifying glass for search, a list icon, a minus sign, a square, and a close (X) button. The main area of the terminal shows the command "hvib@hvib:~\$ ssh ubuntu@ip" in green text, indicating it is ready to be executed.

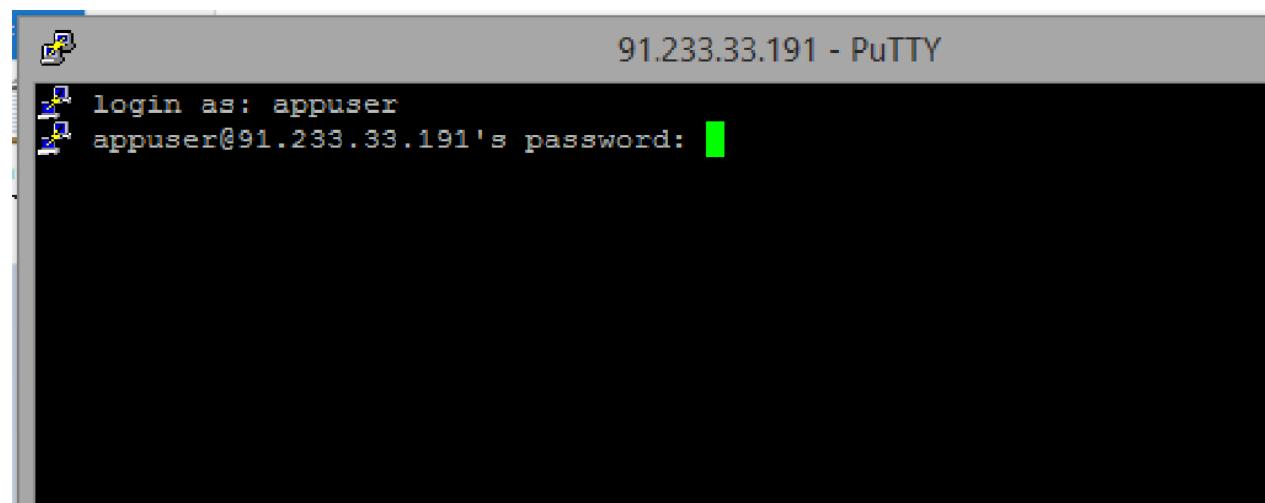
- Connect using Putty
- Follow Steps as highlighted



After this go to the session page and save the detail try to connect with open button.



Use your password of root and you will get connected.



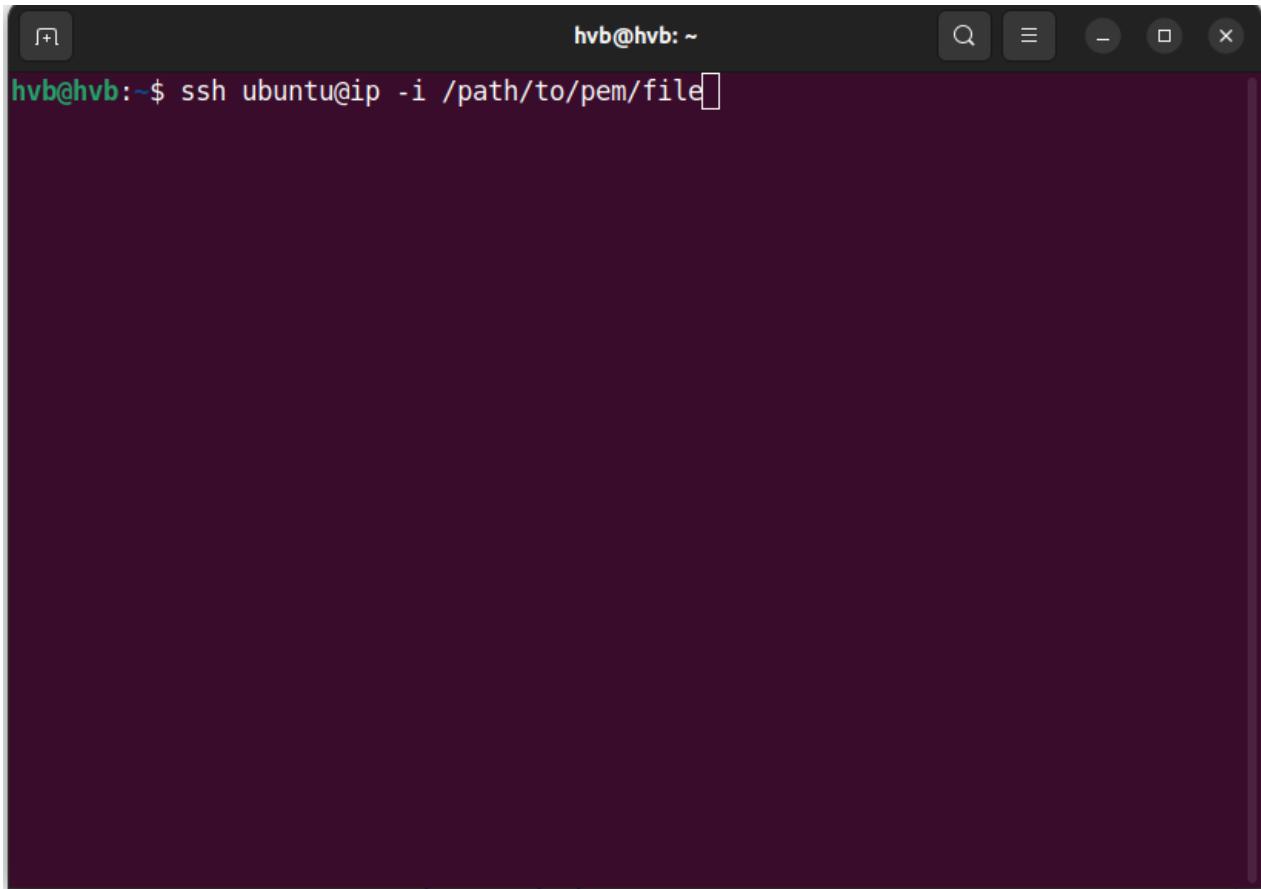
6.3 Connect with PEM / PPK File

- Connect using local terminal

ubuntu : ssh username@ip -i /path/to/pem/file

Replace username with ubuntu or root accordingly.

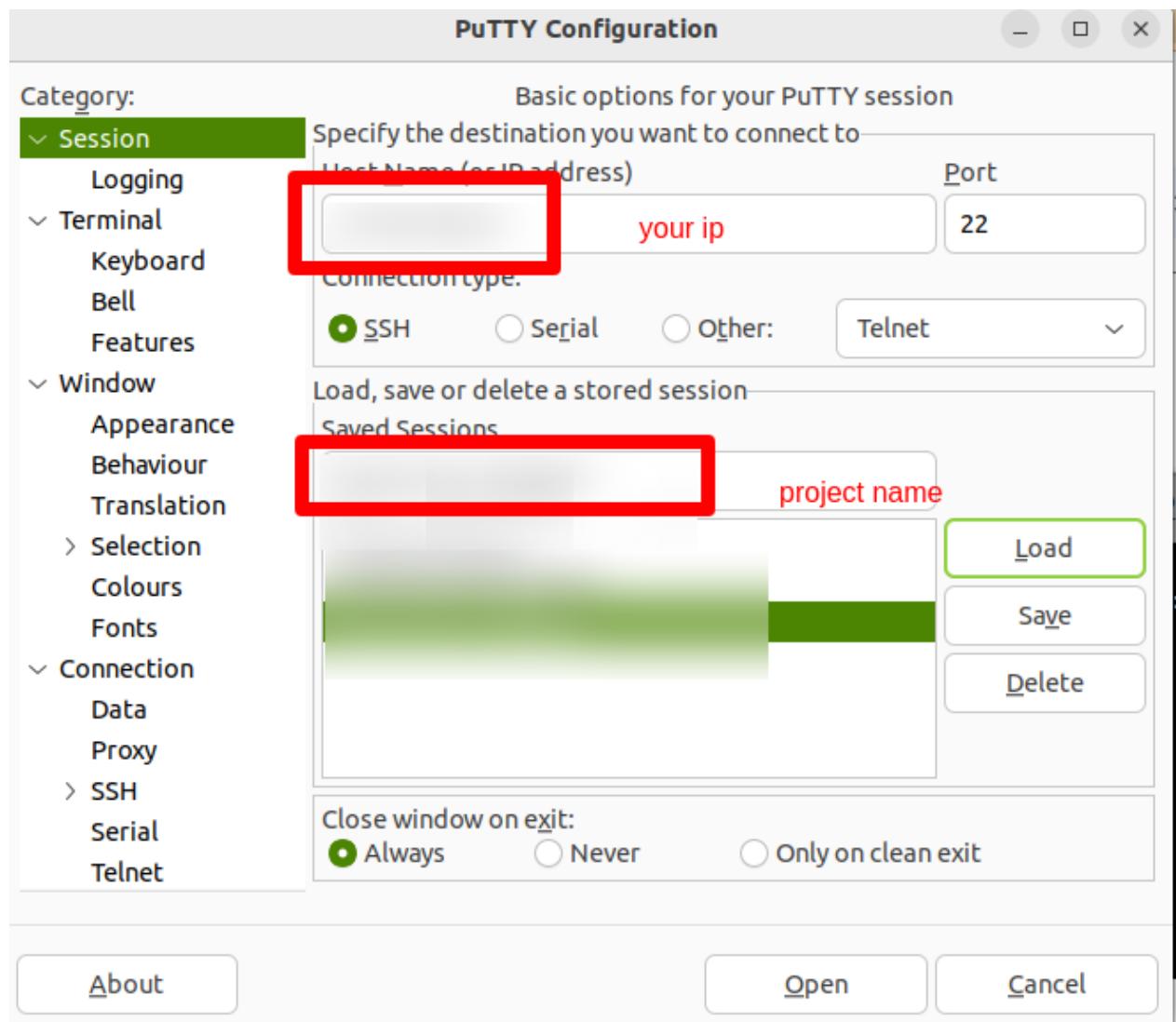
Replace ip with actual Ip address of the server.



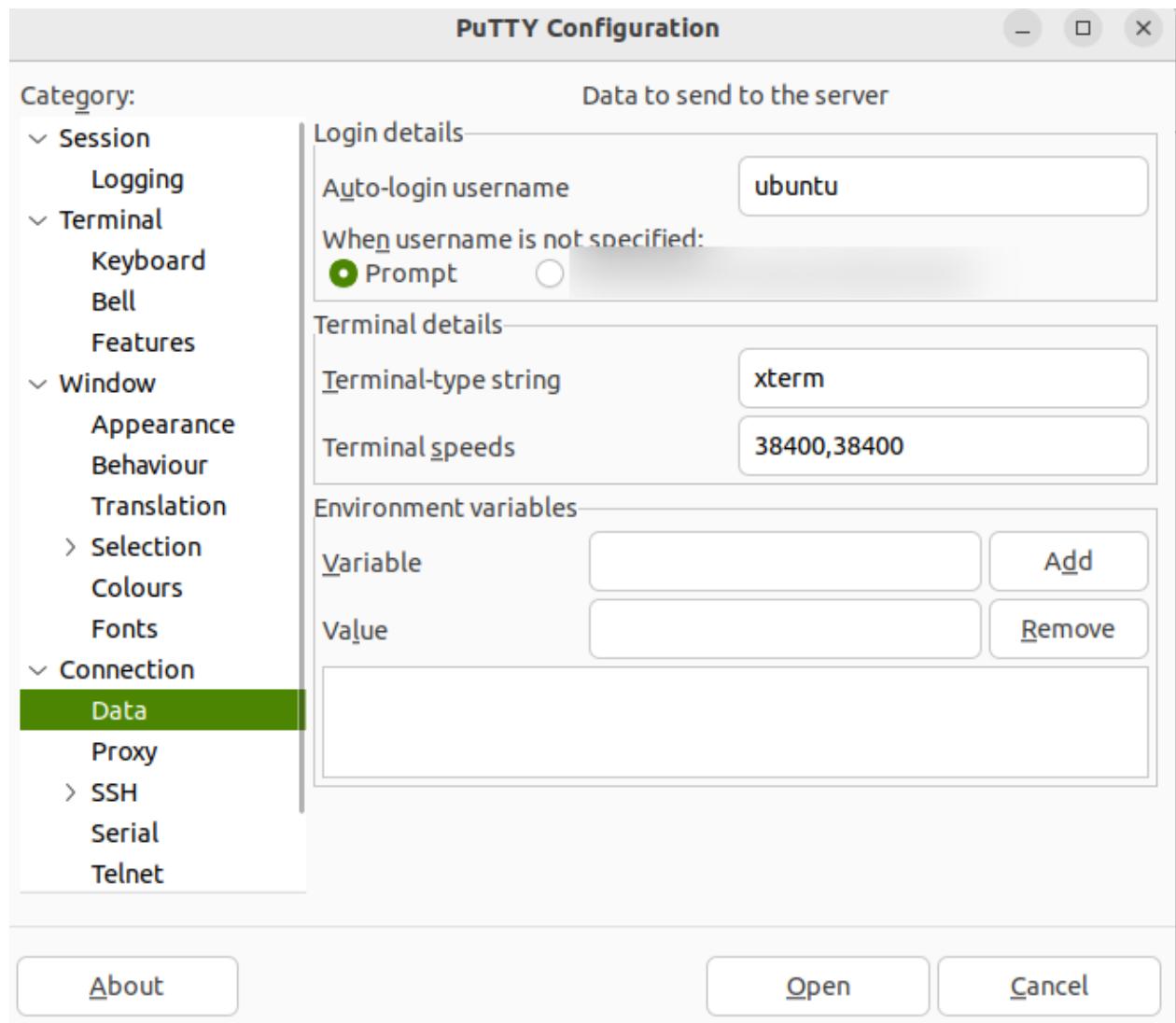
A screenshot of a terminal window titled "hvb@hvb: ~". The window has a dark background and light-colored text. In the terminal, the command "hvb@hvb:~\$ ssh ubuntu@ip -i /path/to/pem/file" is visible, with the cursor positioned at the end of the command line. The window includes standard Linux-style window controls (minimize, maximize, close) at the top right.

- Connect using Putty

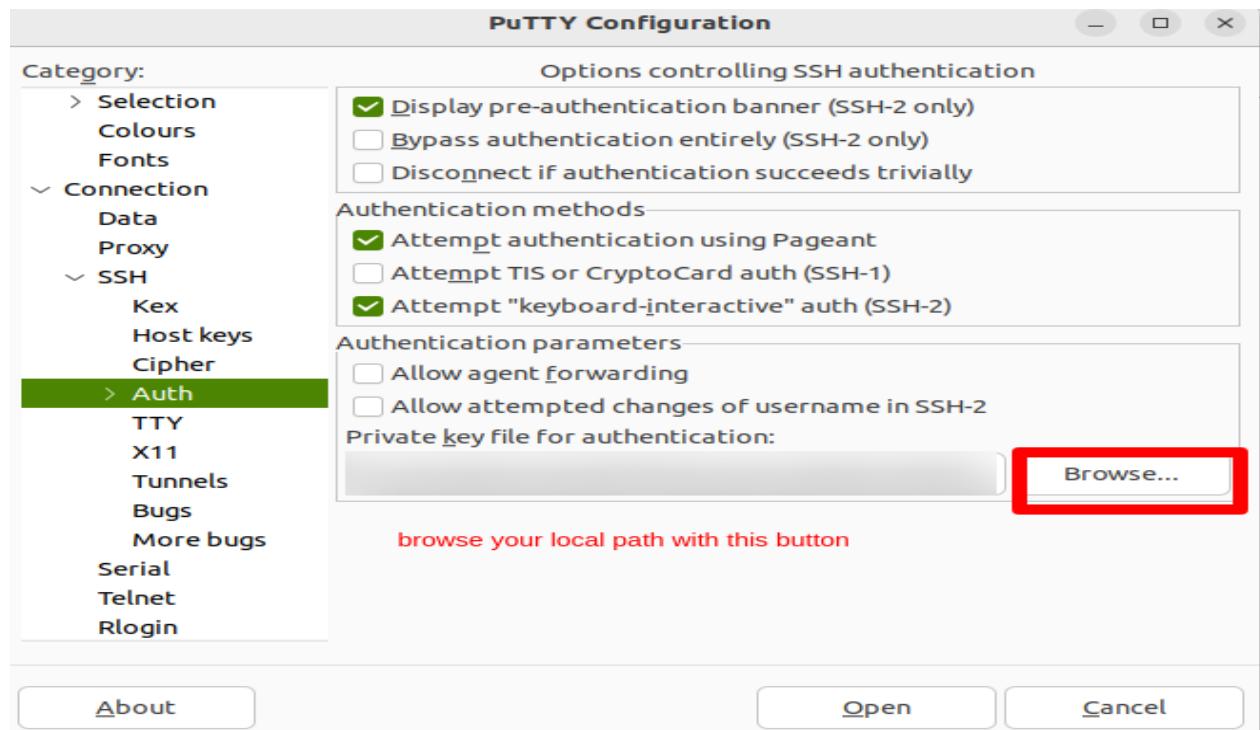
Follow steps as highlighted.



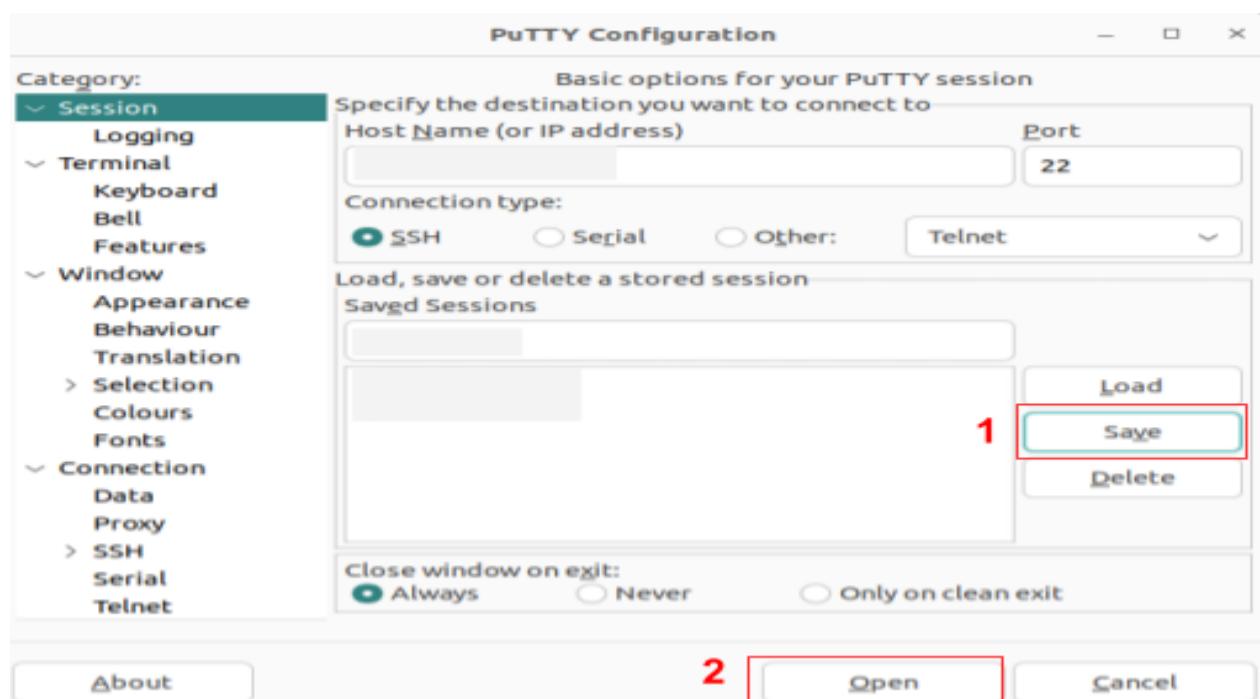
In the **Auto-login username** field replace the username with root or ubuntu accordingly.



Then navigate to the **Auth** tab and select the PEM or PPk file.



After this, go to the session page and save the details, try to connect and it will get connected to the server.



6.4 Perform the below steps to install all dependencies.

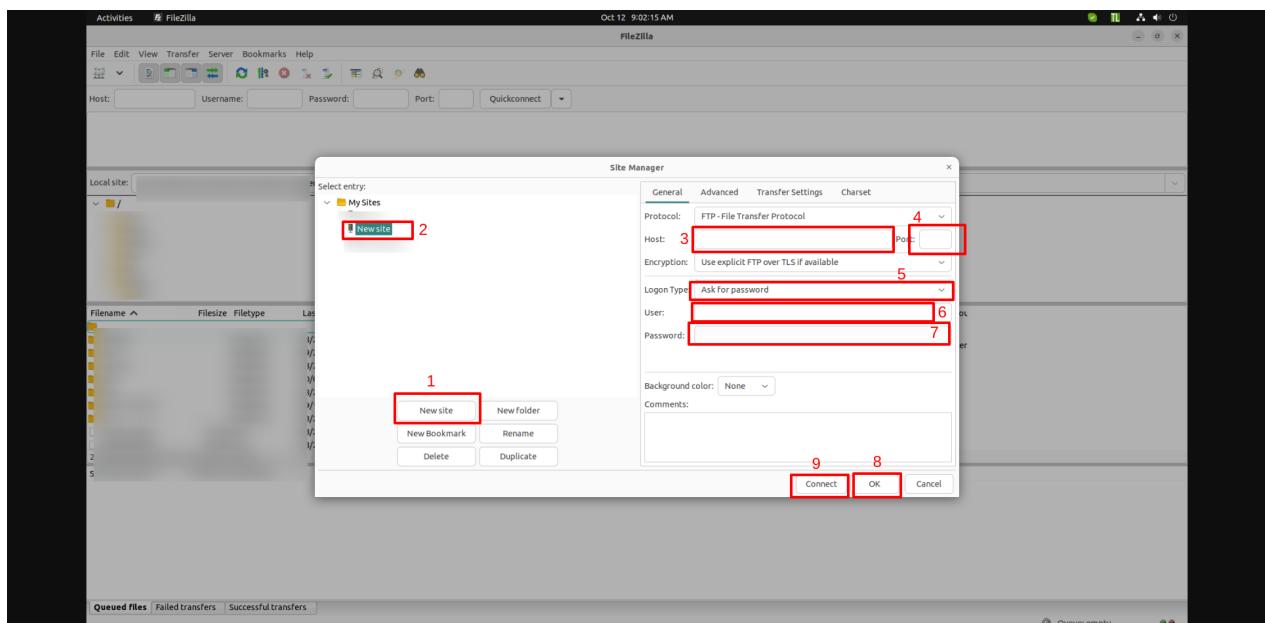
6.4.1 Place the bash file and SSL using FileZilla

- Place the **EBER_INSTALLATION.sh** file and **SSL** certificate by following the mentioned steps.

- Connect to Server using fileZilla.

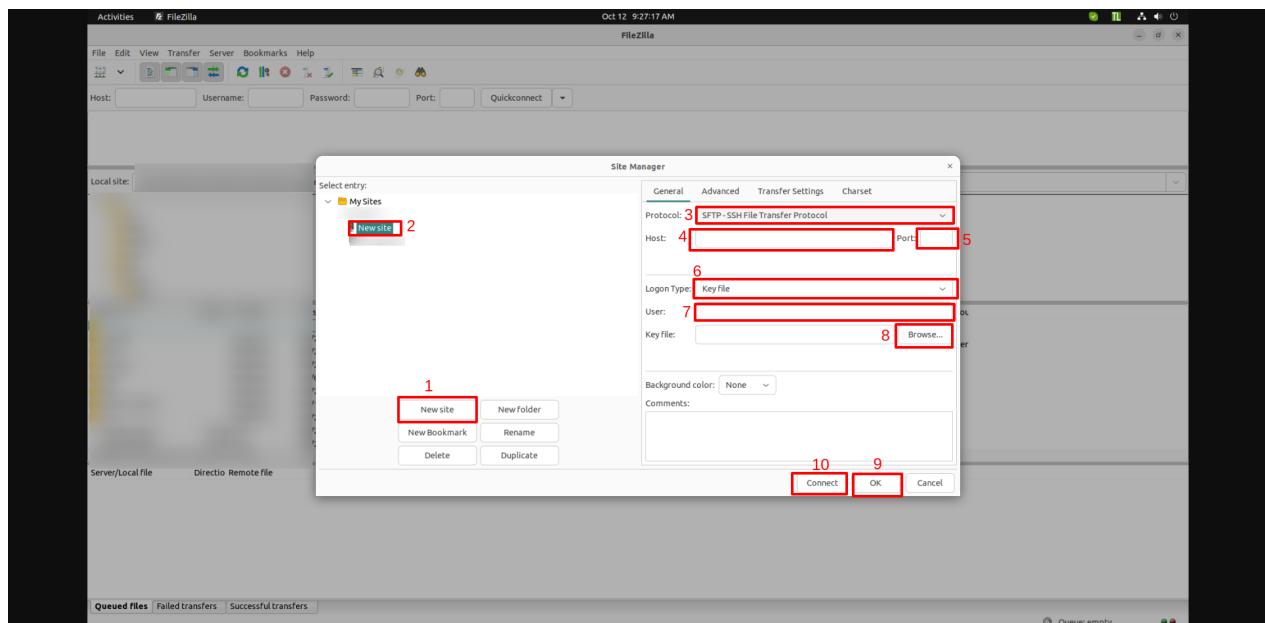
- With Password Credentials

- **New Site**
- **Rename**
- **Server IP**
- **Port 22**
- Select Logon Type **Normal** from the list
- Username **root** or **ubuntu**
- **Password**
- **OK**
- **Connect**



- With Pem/Ppk file key

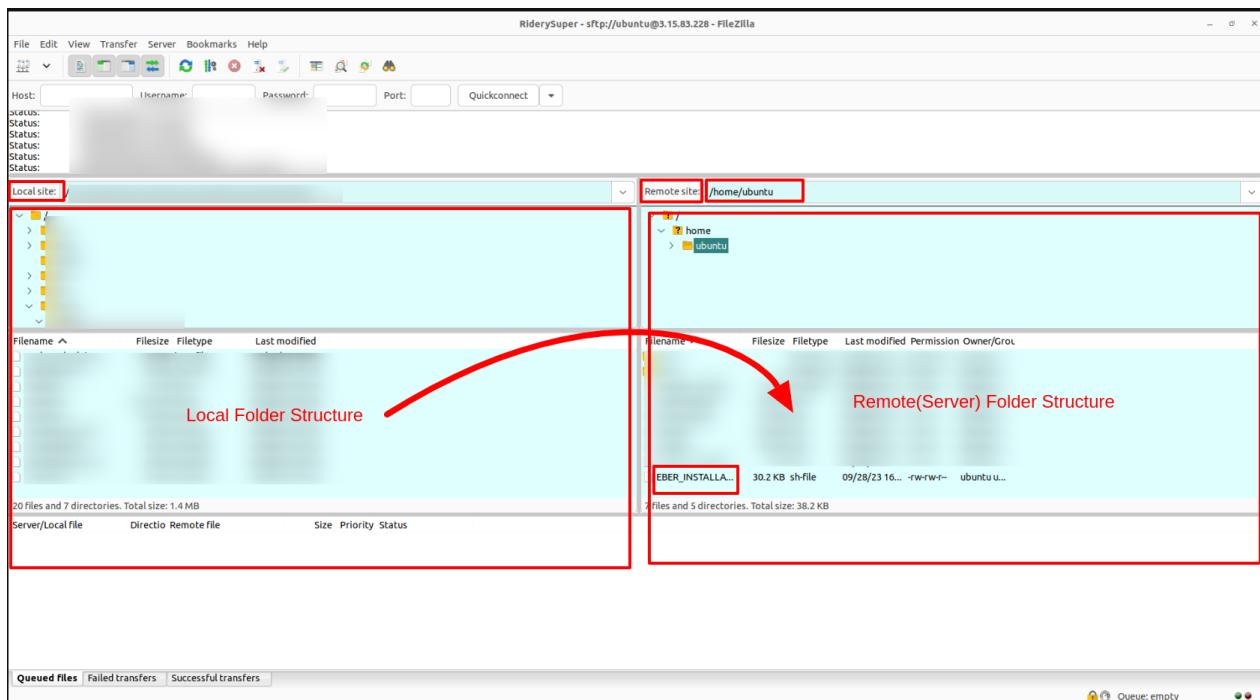
- **New Site**
- **Rename**
- **Protocol SFTP**
- **Server IP**
- **Port 22**
- Select Logon Type **Key File** from the list
- Username **root** or **ubuntu**
- **Browse** file
- **OK**
- **Connect**



- Place the files on the server manually.

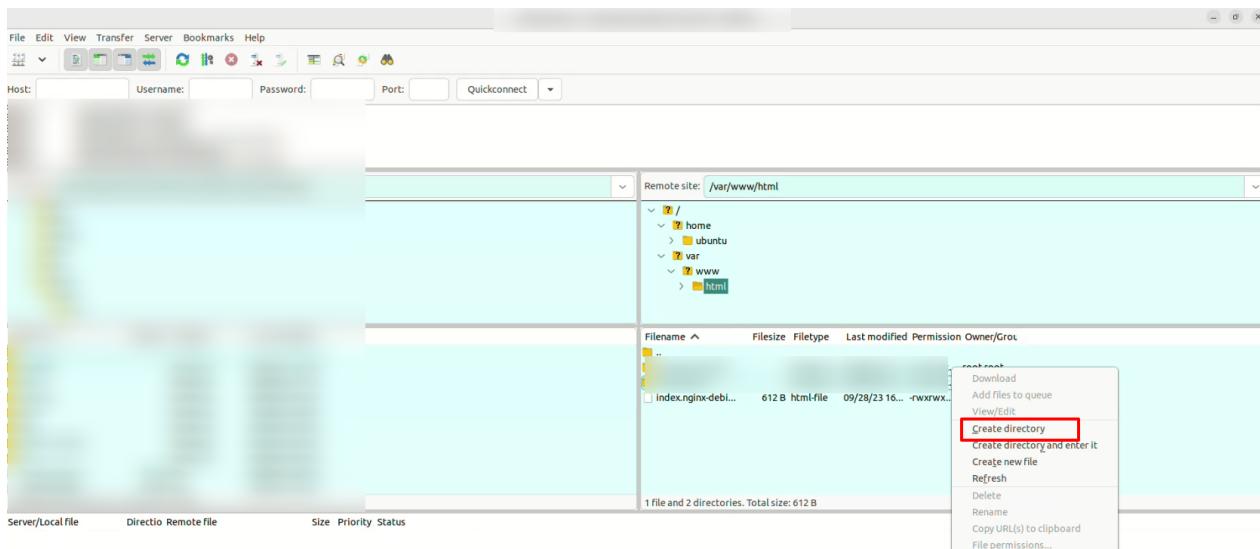
- EBER_INSTALLTATION.sh bash file

- After connecting successfully navigate to the folder path(/home/ubuntu(or)root) highlighted in the remote folder structure.
- Then after drag and drop the bash file from the local folder structure to the remote folder structure.

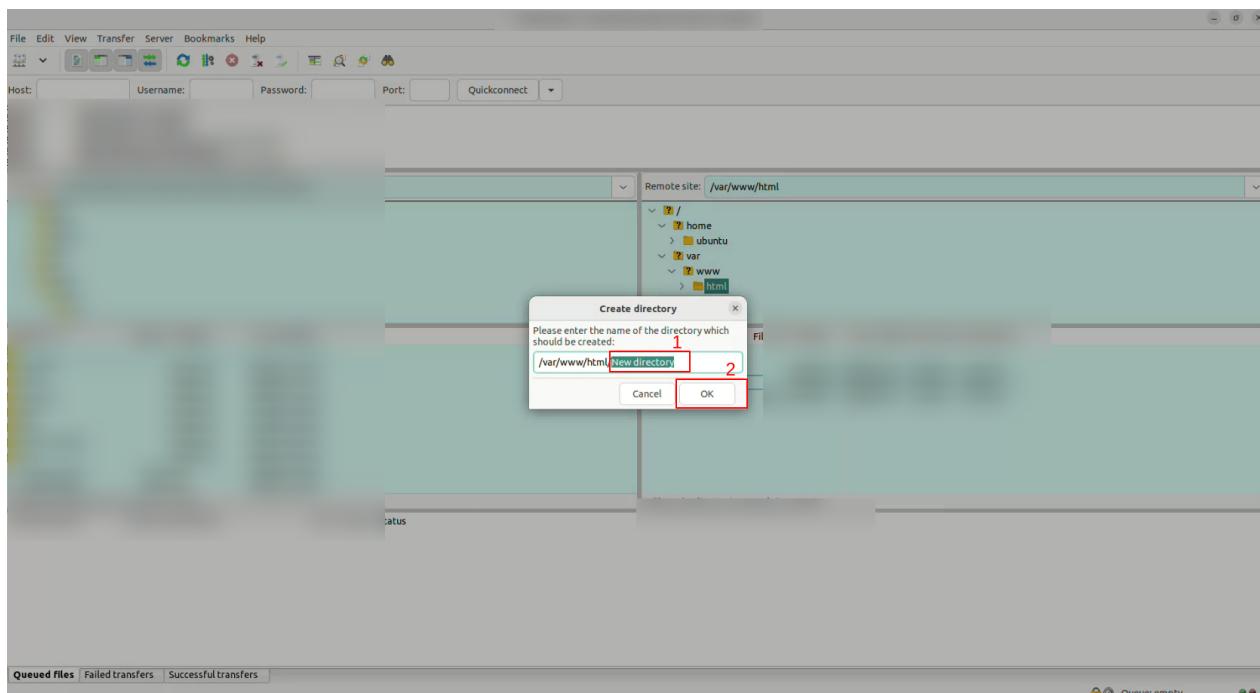


■ SSL certificates

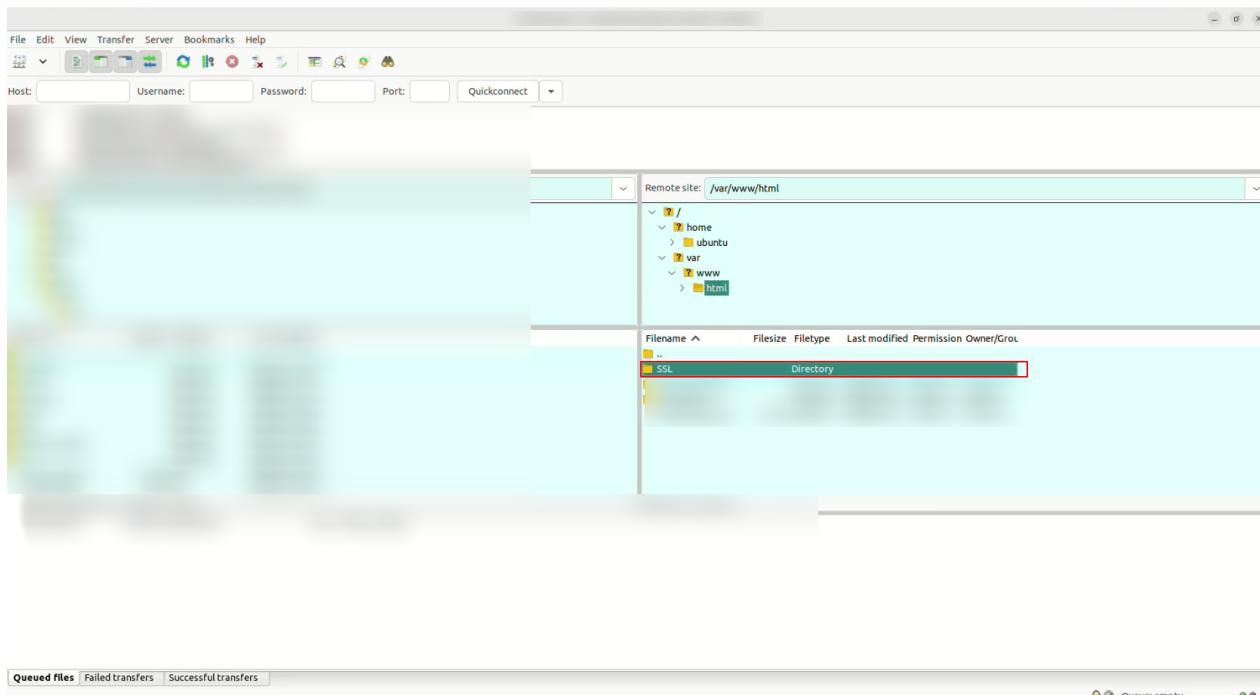
- Right click on the remote folder structure create new directory



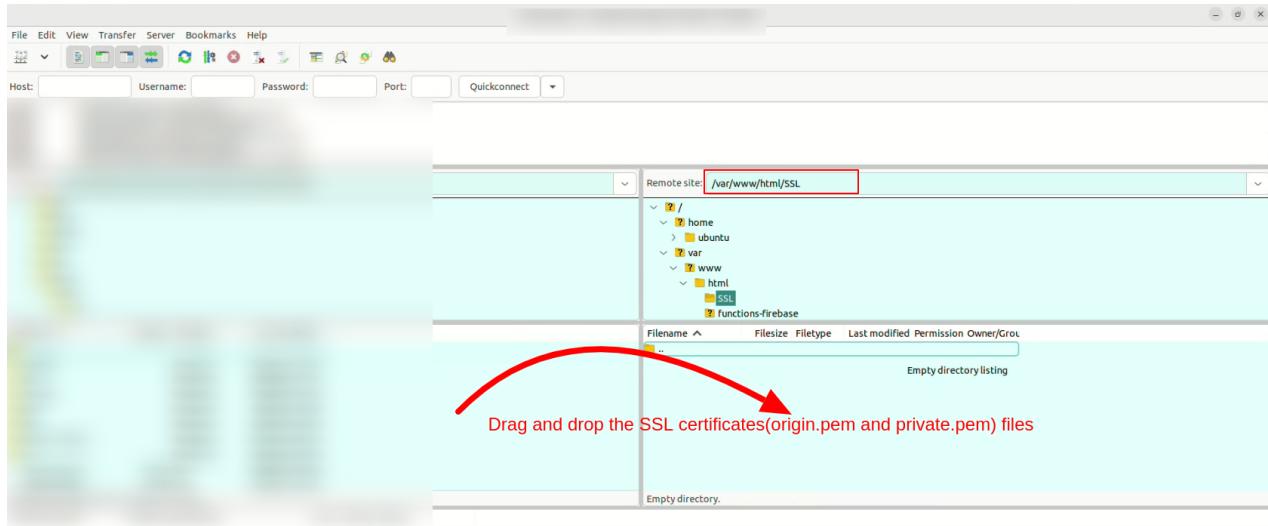
- Rename the **new directory** with **SSL**(keep the folder name in **UPPERCASE**) then press OK.



- Open the **SSL** directory



- Confirm the path as highlighted and then drag and drop the SSL certificates(origin.pem and private.pem) files.



- Now we have placed the **SSL** certificates and **Bash** file on the Server

6.4.2 Loading the Server with all the requirements.

Perform the below steps to install all dependencies or important server domain:

Command to change permission from root (Used in AWS) :

sudo chown <hostname>

hostname = root/ (for digitalocean/vultr domain)

ubuntu/ (for AWS domain)

Installing environment for our code to run

Prerequisite:

- Code Repo Urls
 - Basic Knowledge of command line.
- Place installation bash script file in the instance where you want to perform installation(we done already at step [6.4.1](#)).

- Run this file with the use of following command:

bash BASH_FILE_PATH

- It will ask you to enter choice,
Press **1** if you want full Installation.
Press **2** if you want custom Installation.

1) For Full Installation , you have to do following steps

- This process will install all required dependencies, packages and applications automatically.
- This installer file has appropriate description at each stage.
- It will ask for version selection for Nodejs(Step-2) and MongoDB(Step-5).
- Keep the git urls ready with you(Step-10). At every stage you will be asked to select option and appropriately select the option and set up the project.
- In the "default file" configuration stage it will ask to input total number of domains along with the port number on which associate server is running (Step-12).
- If you already have the SSL certificates, skip (Step-13).

2) For Custom Installation, you will get following list

- 0) Exit
- 1) Update Packages
- 2) NodeJS
- 3) Angular
- 4) Nginx
- 5) MongoDB
- 6) Redis
- 7) Nodemon
- 8) PM2
- 9) Install Git
- 10) Setup Your Project
- 11) PM2 startup
- 12) Setup Nginx Default File
- 13) Install Letsencrypt

- You can choose from the above list.

Note:

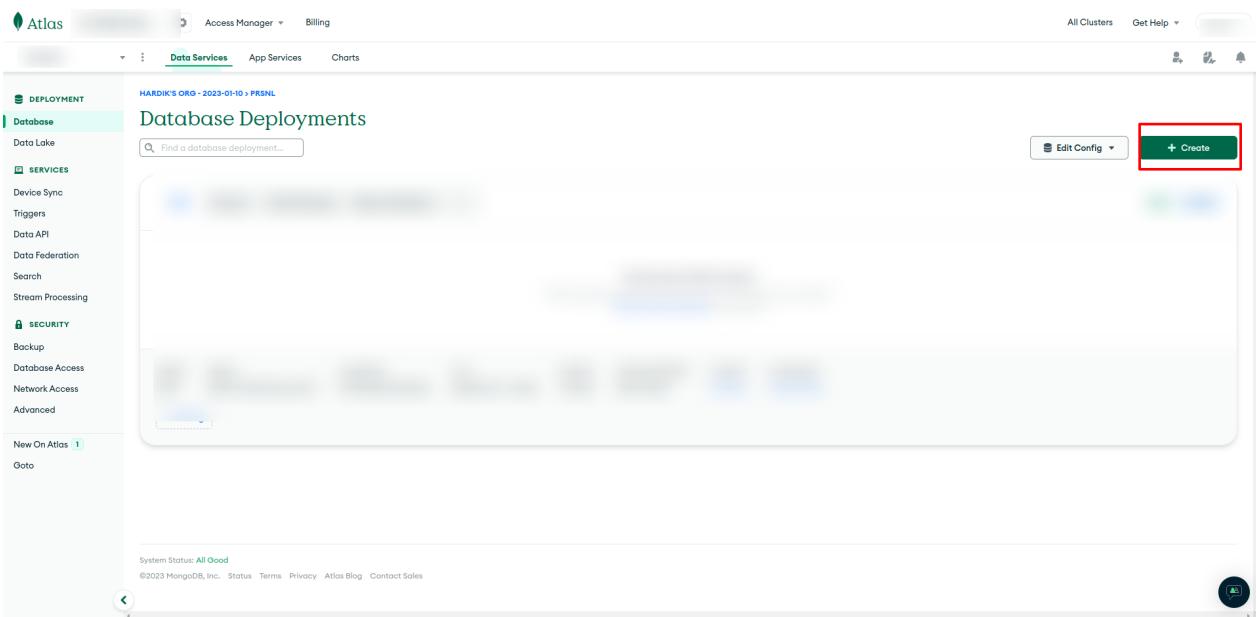
If you select the default nginx file config then you have to select 12 numbers of domains here.

```
-user panel - port - 4000 - domain - <domain name>
-backend server(main) port - 5000 - domain - api.<domain name>
-backend history port- 5001 - domain - history.<domain name>
-backend payment port - 5002 - domain - payment.<domain name>
-backend notification port - 5003 - domain - notification.<domain name>
-driver panel - port - 6000 - domain - driver.<domain name>
```

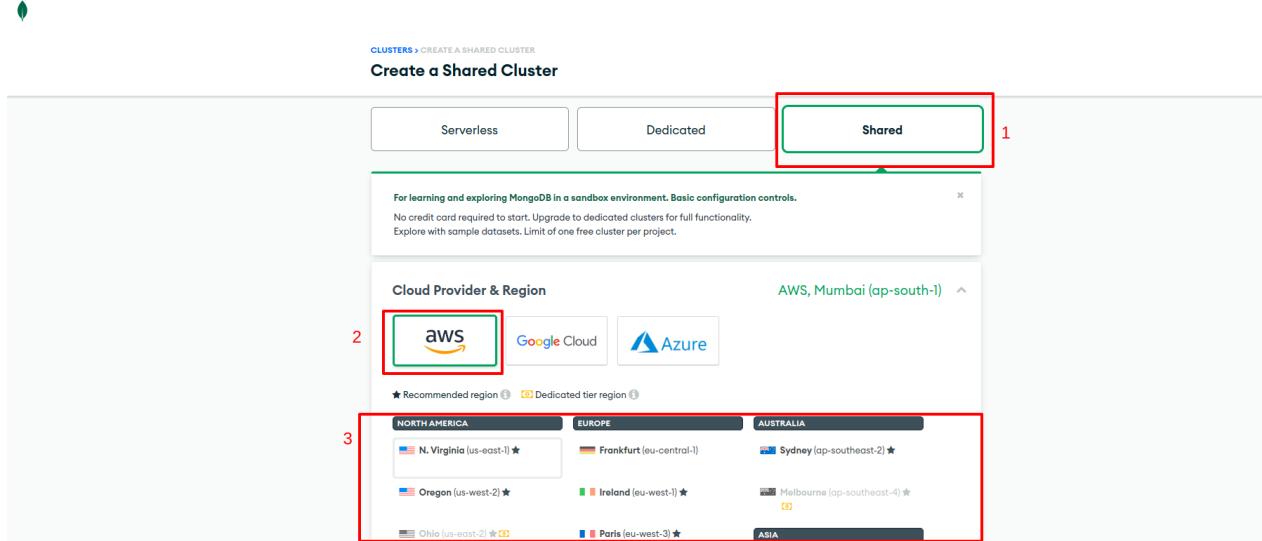
- hub panel - port - 6001 - domain - hub.<domain name>
- dispatcher panel - port - 7000 - domain - dispatcher.<domain name>
- corporate panel - port - 7500 - domain - corporate.<domain name>
- hotel panel - port - 8000 - domain - hotel.<domain name>
- partner panel - port - 8500 - domain - partner.<domain name>
- admin panel - port - 9000 - domain - admin.<domain name>

6.5 Create MongoDB Atlas

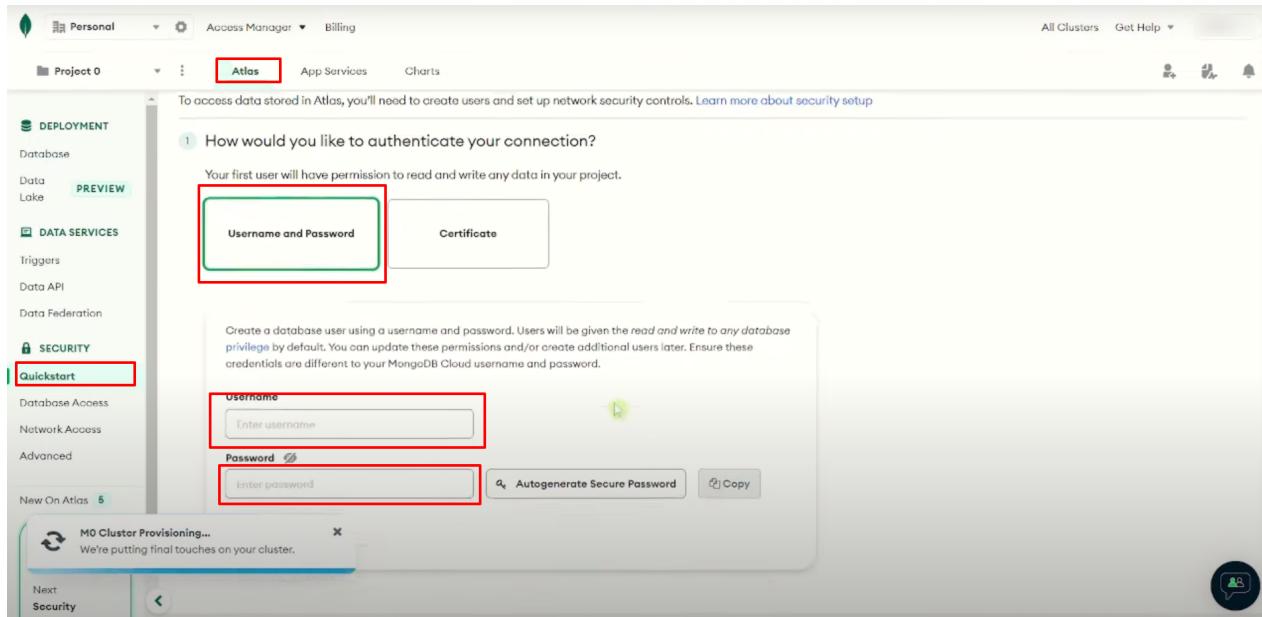
- MongodB Atlas
 - Account
 - [Login](#) or [Register](#)
 - If you registered a new account, check the mail and verify your email.
 - Create DB



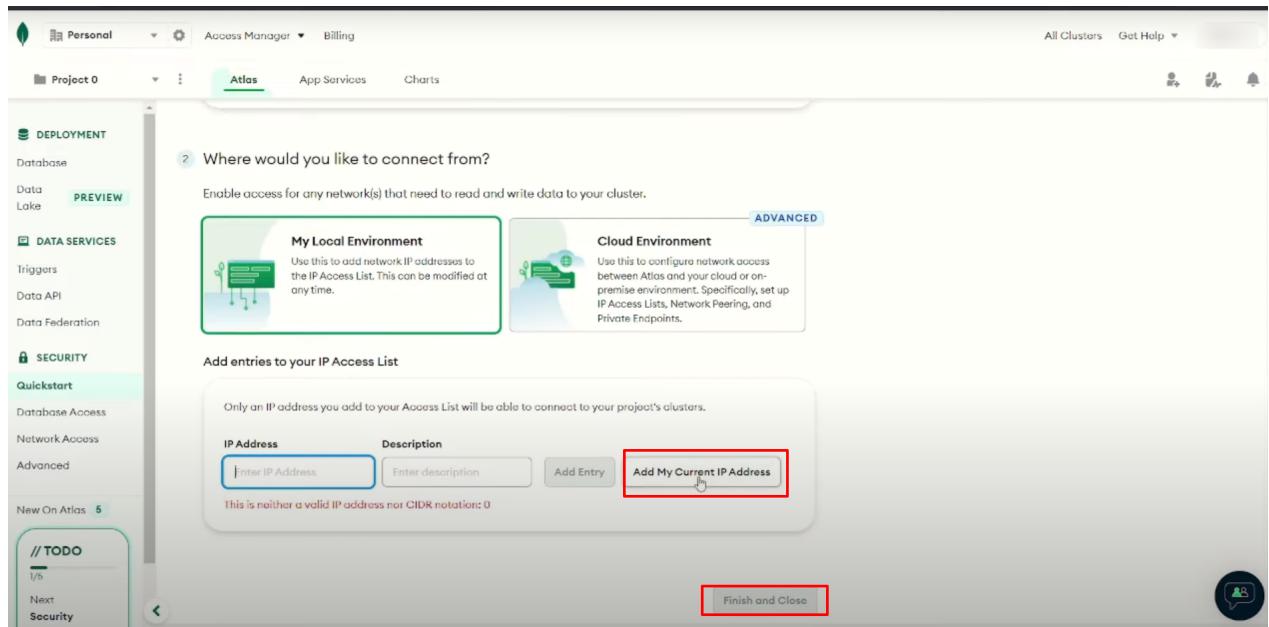
Select **Shared** as it is free and it provides us up to 512 MB.



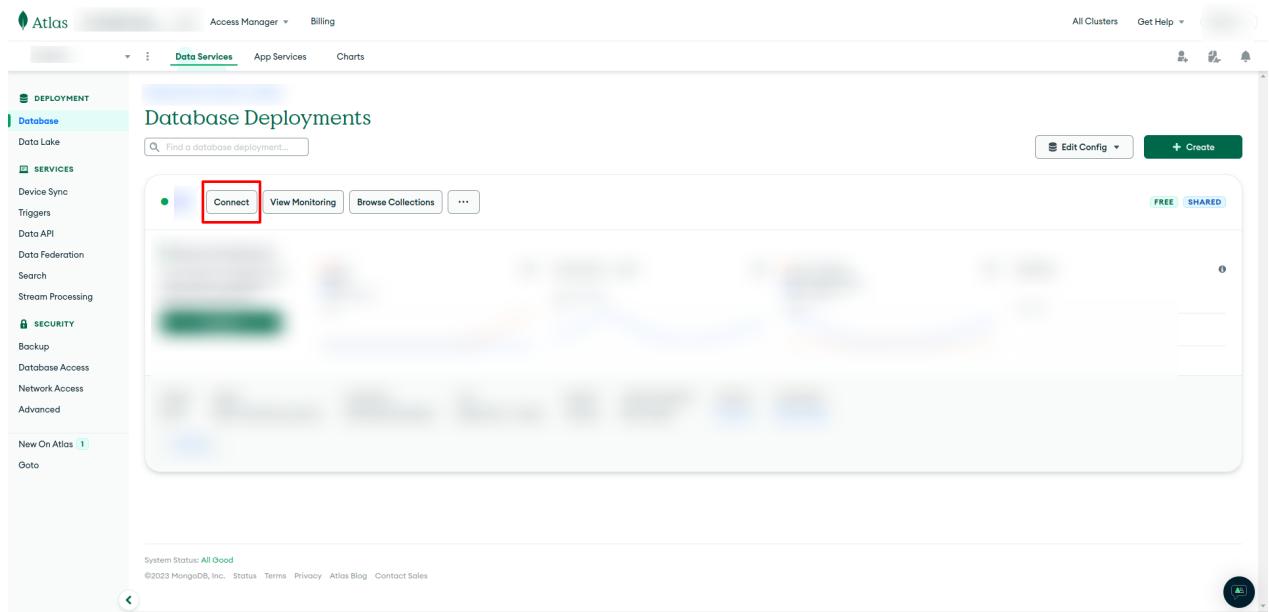
Fill the username and password(store at a safe place) to access the db.

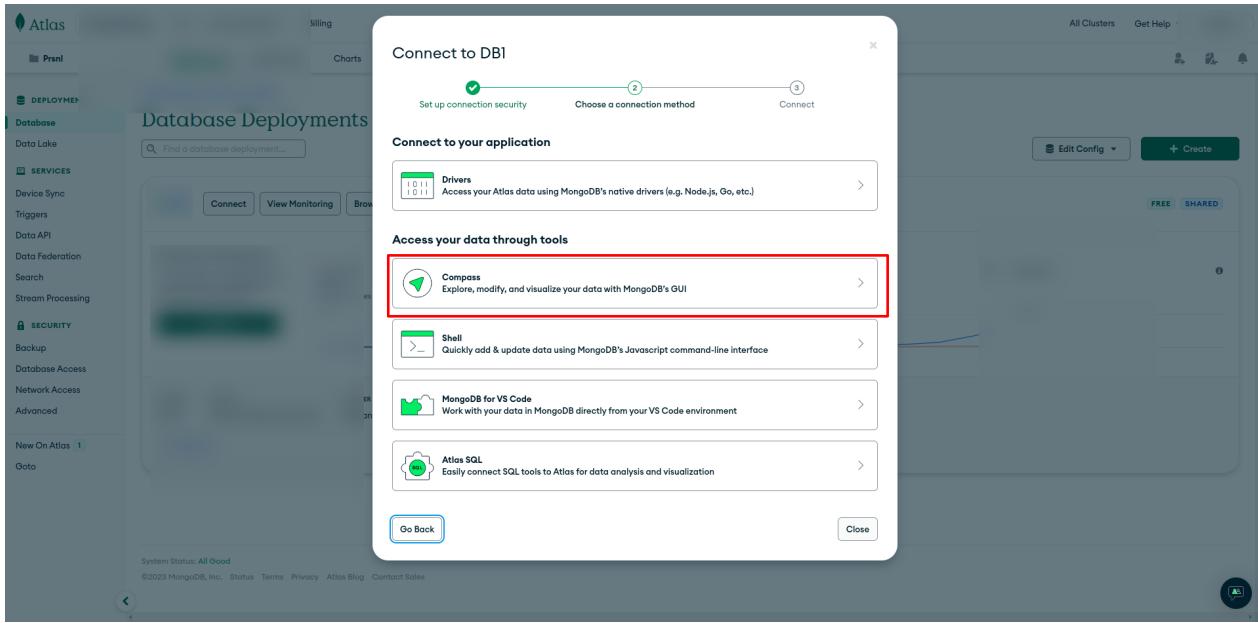


Add your current IP address so that you can access the db and also add the IP address of the server obtained while creating the server on cloud.
Then click finish and close.



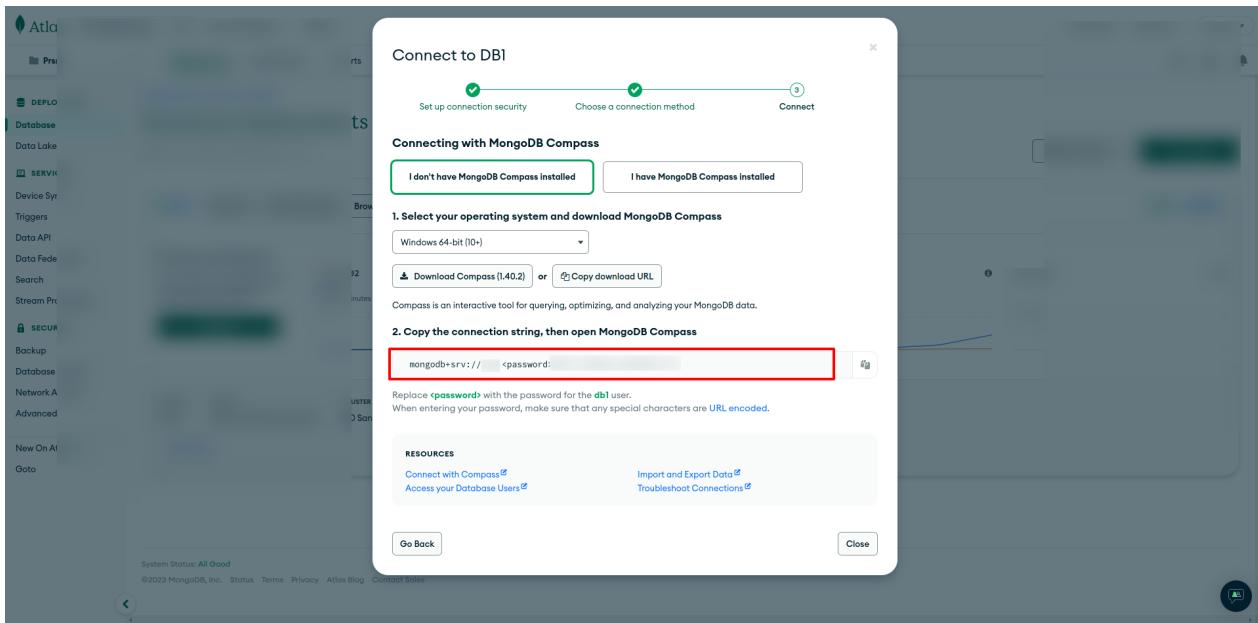
■ Get the DB url to connect it with the project.





Save this string.

ie. ***mongodb+srv://username:password@xyz.mongodb.net/***



- And Update it with the string you copied as highlighted in the above image.
- Navigate to the path of files listed below and comment all other db:

ie. db: '***mongodb+srv://username:@xyzmongodb.net/ProjectName***'
Add the project name at the end of the string as mentioned above.

Edit the following listed files using filezilla.

- backend/server/config/env/development.js
- backend/payments/config/env/development.js
- backend/mass_notification/config/env/development.js
- backend/history-earning/config/env/development.js

Once the above mentioned files are updated on the server, navigate the **initial_data.js** file as mentioned below.

- cd var/www/html/<ProjectName>/backend/server/settingsdata/
- Run this code **node initial_data.js**

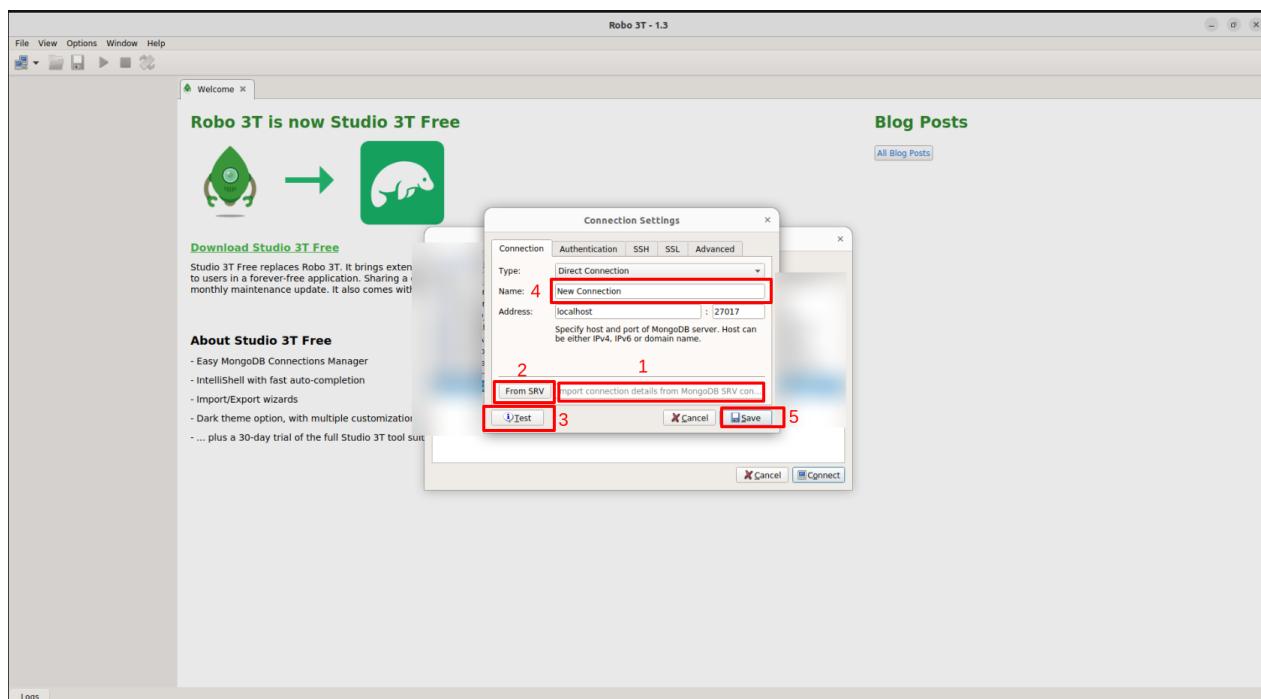
This will create all the initial files in the DB.

6.6 Connection to the DB

- Use the string which we copied from the mongodb atlas platform and follow the steps to view the DB.
- Below I have mentioned the steps to connect using Robo-3t, to connect using other platforms it is very similar to the below steps.

Steps:

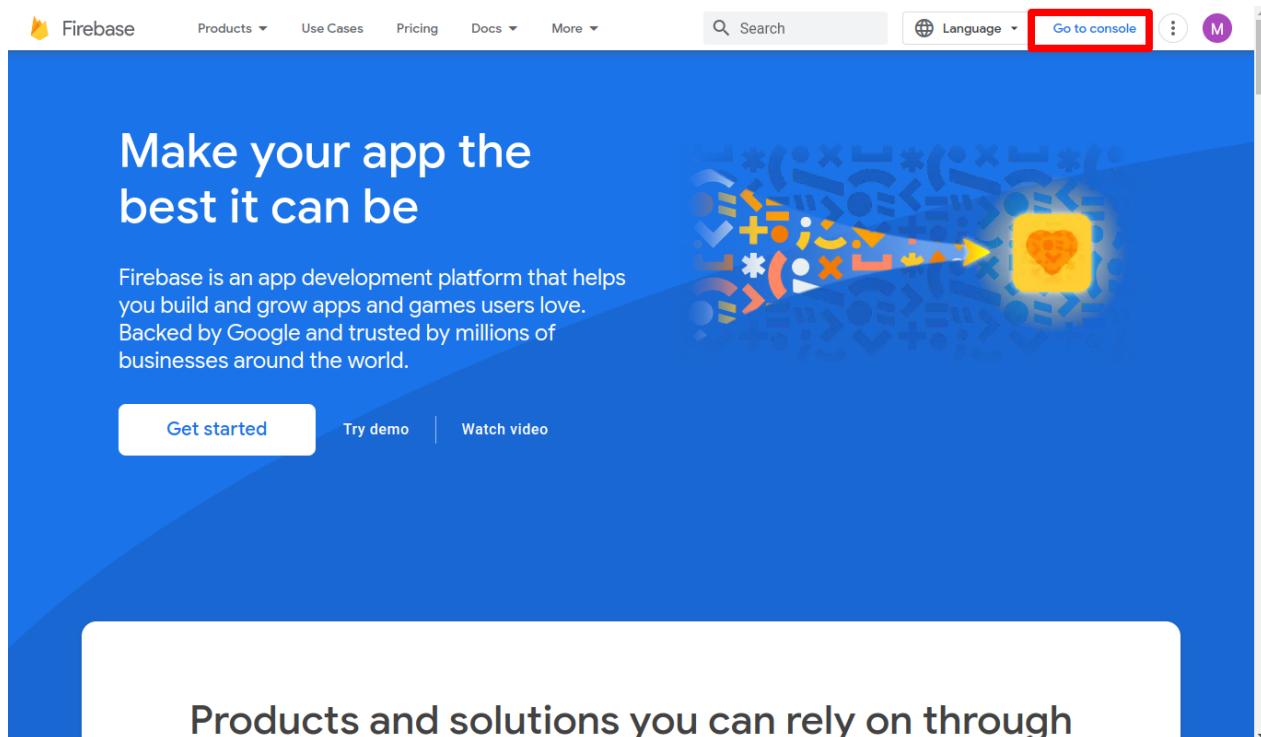
- Paste the String
- Click **From SRV**
- Click **Test**(It should get success status)
- Give the Name
- Click **Save**



7. Firebase Configuration

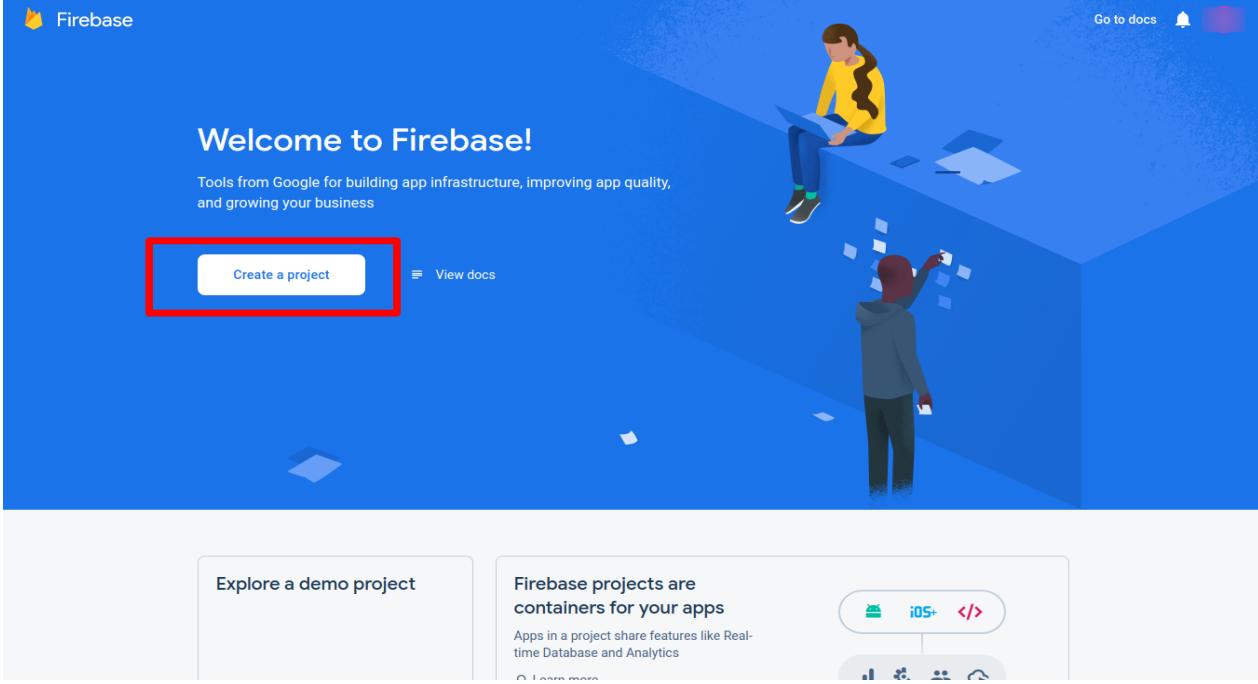
→ Open firebase console [click me](#)

7.1 Go to the console.



7.2 Create Project

Create a project, and complete that 4 steps with the given image below OR you have to use the existing one if the project is already created.



The screenshot shows the Firebase homepage. At the top left is the Firebase logo. Top right features links for "Go to docs" and a notification bell. A large blue header area contains the text "Welcome to Firebase!" and "Tools from Google for building app infrastructure, improving app quality, and growing your business". Below this are two buttons: "Create a project" (highlighted with a red box) and "View docs". The background of the header is a stylized illustration of two people working on a computer and a whiteboard.

Create a project

View docs

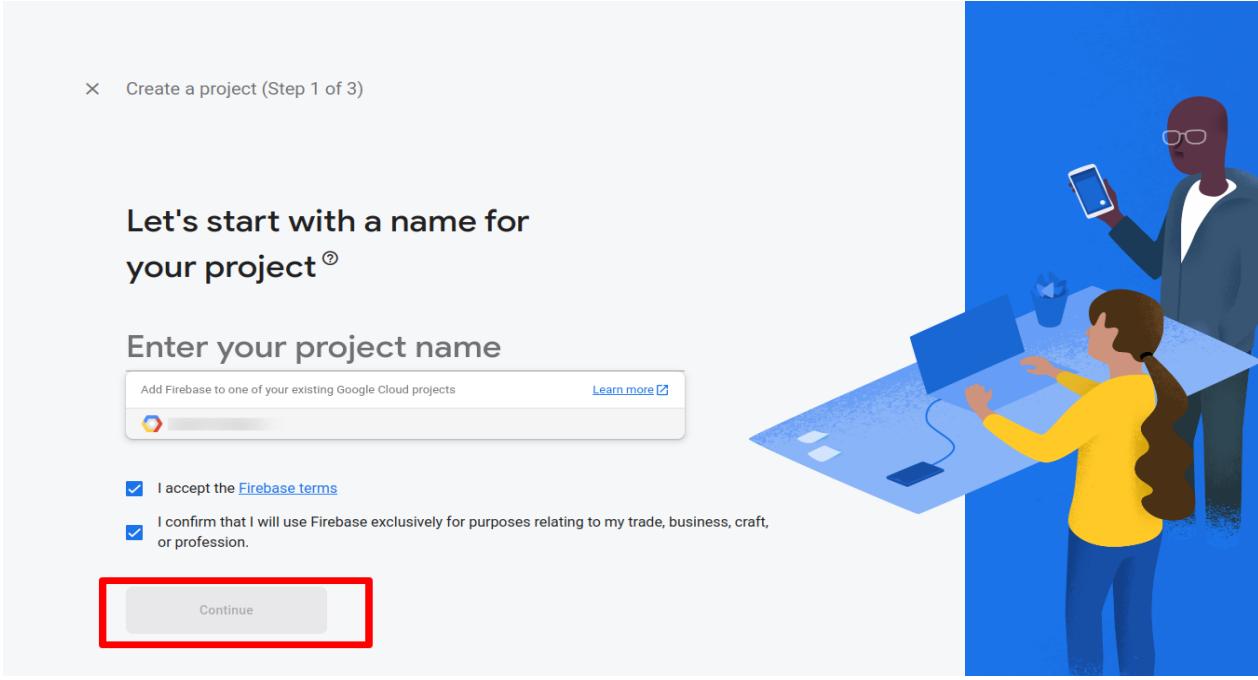
Welcome to Firebase!

Tools from Google for building app infrastructure, improving app quality, and growing your business

Explore a demo project

Firebase projects are containers for your apps

iOS+ </>



The screenshot shows the "Create a project (Step 1 of 3)" page. It includes a sub-header "Let's start with a name for your project[®]", a text input field for "Enter your project name", and a note about adding to an existing Google Cloud project. Below the input field are two checkboxes: "I accept the [Firebase terms](#)" and "I confirm that I will use Firebase exclusively for purposes relating to my trade, business, craft, or profession." At the bottom is a "Continue" button, which is highlighted with a red box.

× Create a project (Step 1 of 3)

Let's start with a name for your project[®]

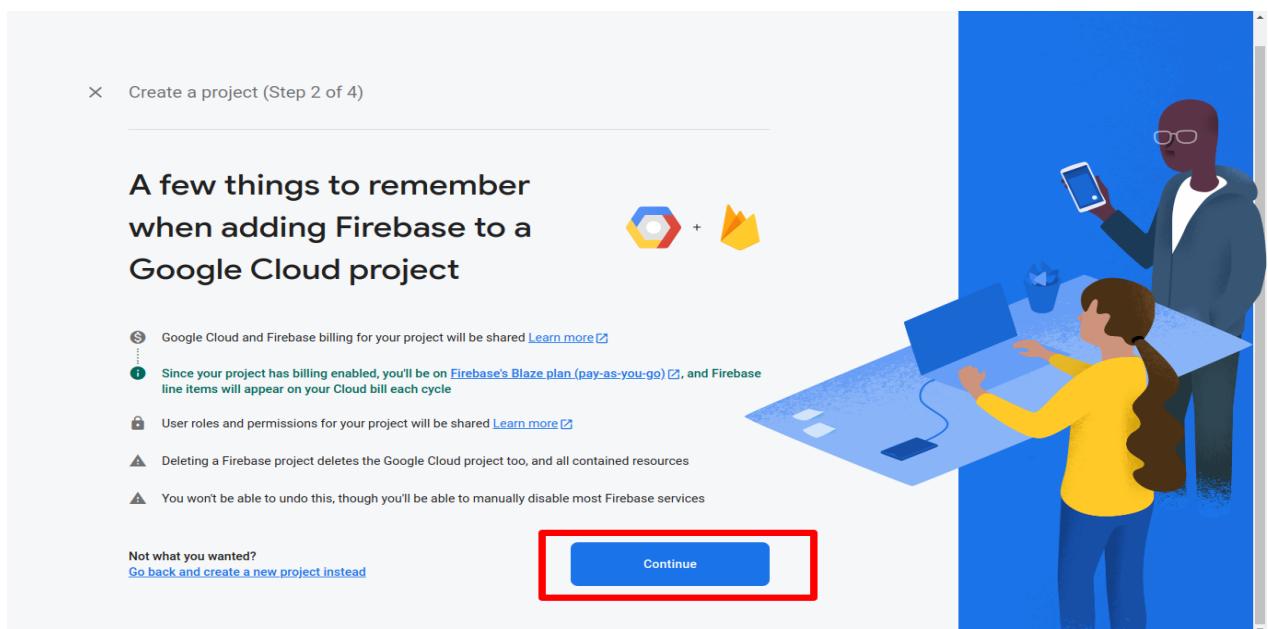
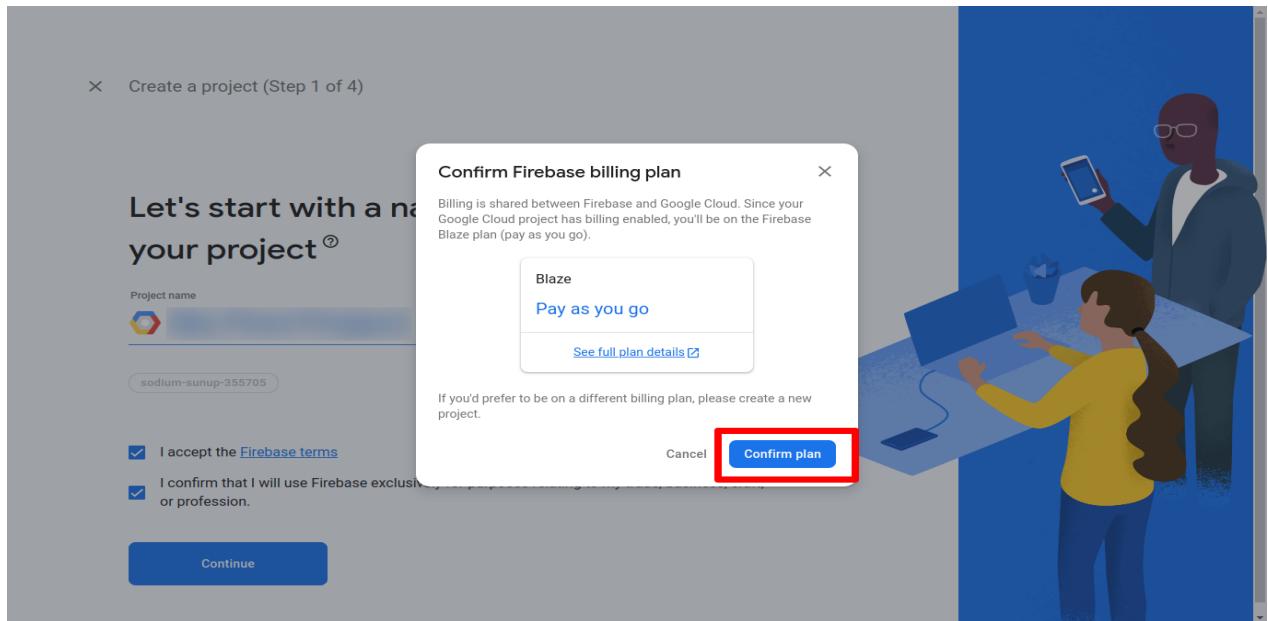
Enter your project name

Add Firebase to one of your existing Google Cloud projects [Learn more](#)

I accept the [Firebase terms](#)

I confirm that I will use Firebase exclusively for purposes relating to my trade, business, craft, or profession.

Continue



×

Create a project (Step 3 of 4)

Google Analytics for your Firebase project

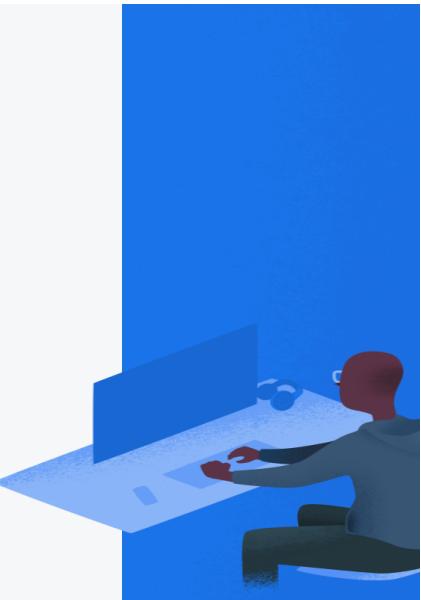
Google Analytics is a free and unlimited analytics solution that enables targeting, reporting, and more in Firebase Crashlytics, Cloud Messaging, In-App Messaging, Remote Config, A/B Testing, and Cloud Functions.

Google Analytics enables:

- A/B testing ⓘ
- User segmentation & targeting across Firebase products ⓘ
- Crash-free users ⓘ
- Event-based Cloud Functions triggers ⓘ
- Free unlimited reporting ⓘ

Enable Google Analytics for this project
Recommended

Previous Continue



×

Create a project (Step 4 of 4)

Configure Google Analytics

Analytics location ⓘ

Data sharing settings and Google Analytics terms

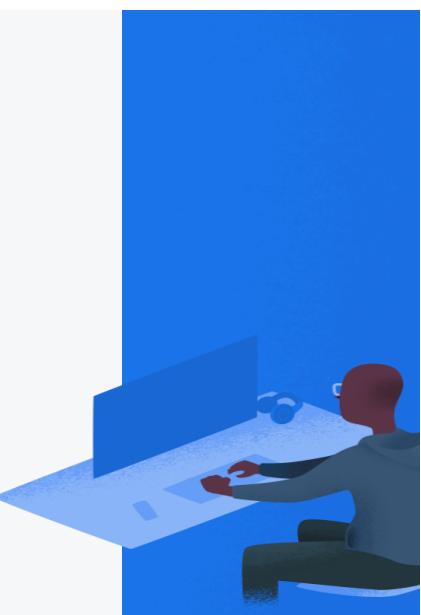
Use the default settings for sharing Google Analytics data. [Learn more](#)

- Share your Analytics data with Google to improve Google Products and Services
- Share your Analytics data with Google to enable Benchmarking
- Share your Analytics data with Google to enable Technical Support
- Share your Analytics data with Google Account Specialists

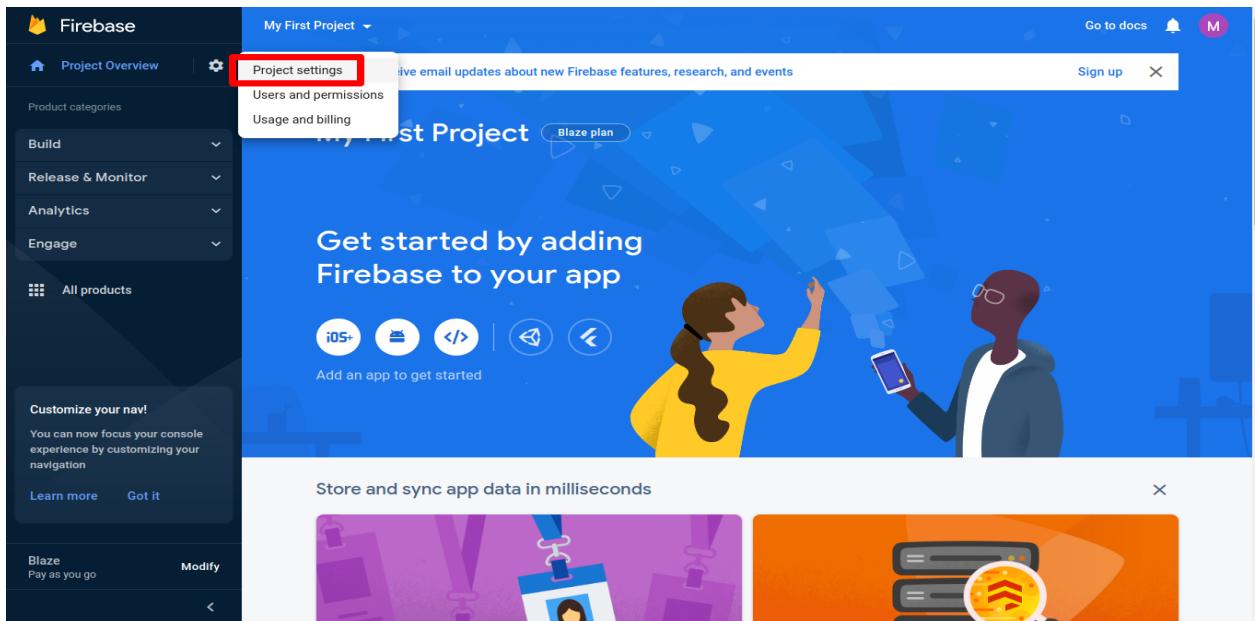
I accept the [Google Analytics terms](#)

Upon project creation, a new Google Analytics property will be created and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. [Learn more](#)

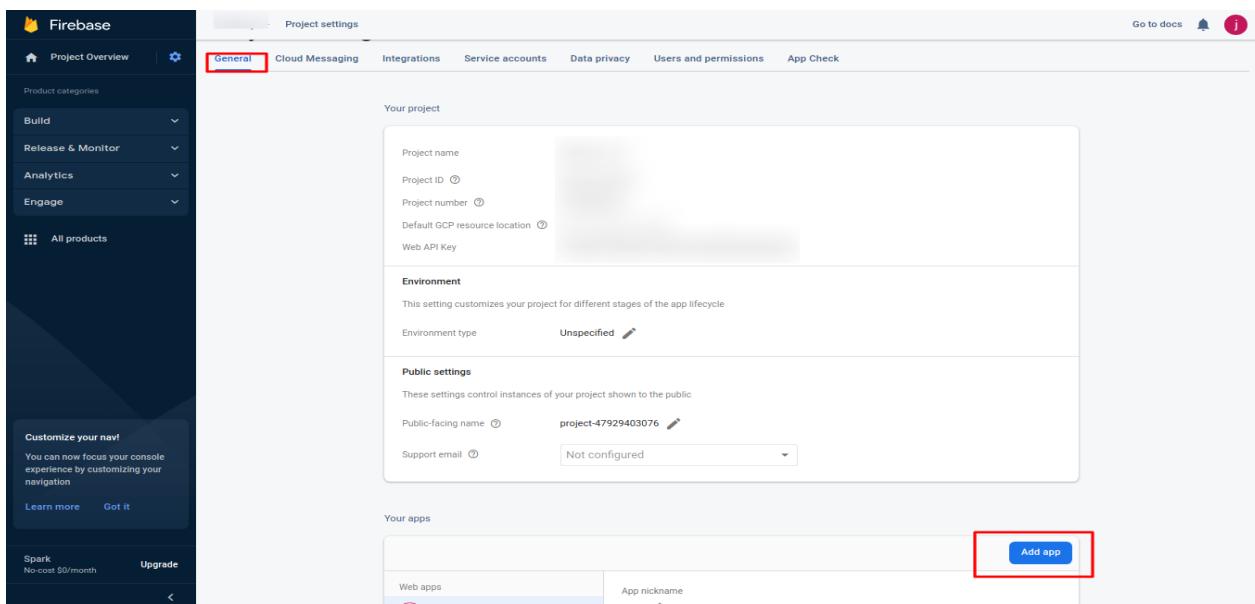
Previous Add Firebase



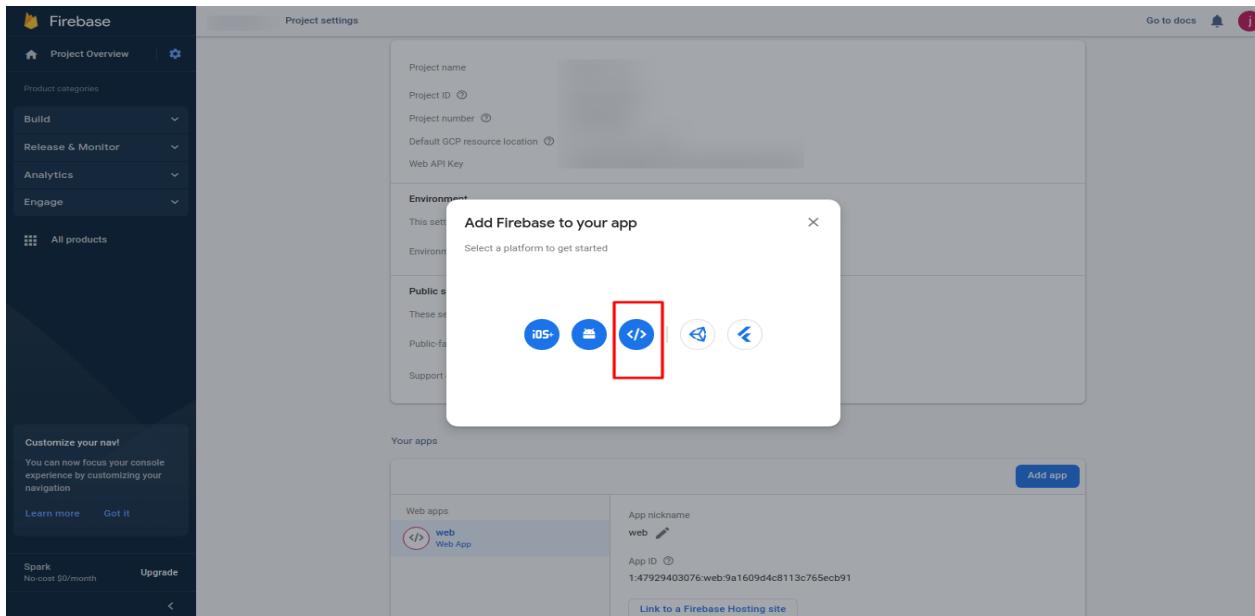
7.3 Create a web app in firebase General project settings.



The screenshot shows the Firebase Project Overview page for a project named "My First Project". On the left sidebar, there are several navigation categories: Build, Release & Monitor, Analytics, Engage, and All products. A "Customize your nav!" section allows users to focus their console experience by customizing their navigation. Below this, there are links for "Blaze" (Pay as you go) and "Modify". At the top right, there are links for "Go to docs", "Sign up", and a profile icon. A red box highlights the "Project settings" option in the top navigation bar. A tooltip for "Project settings" explains it allows users to receive email updates about new Firebase features, research, and events. Below the navigation, there's a main banner with the text "Get started by adding Firebase to your app" and illustrations of two people interacting with a smartphone. A call-to-action button says "Add an app to get started".

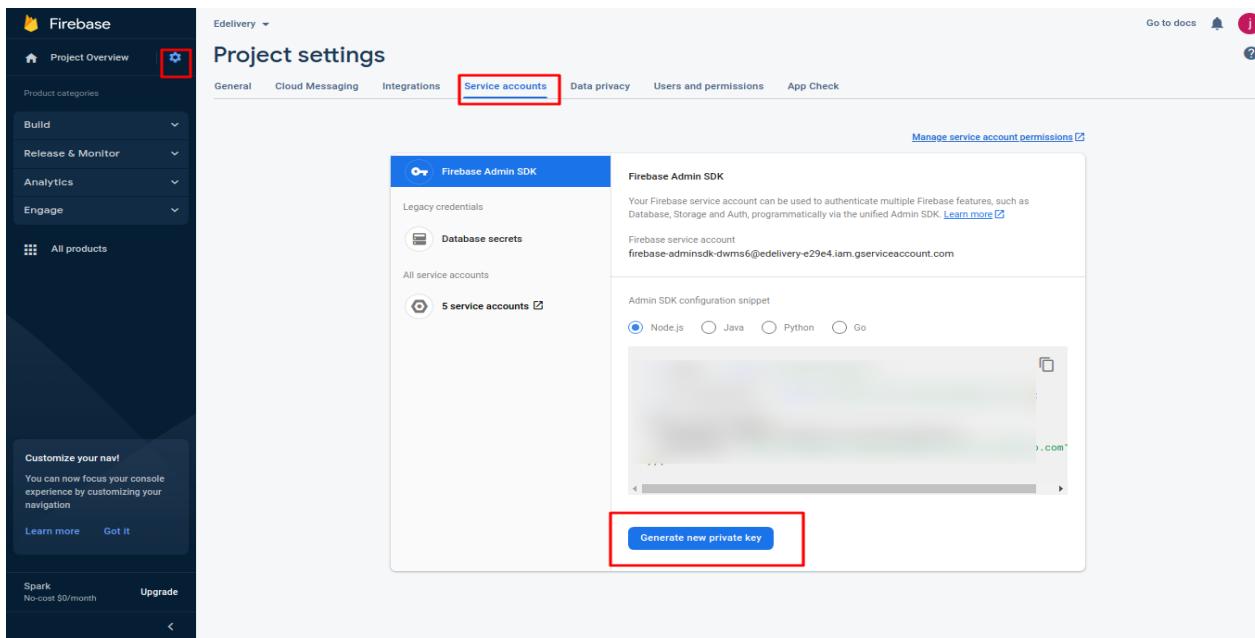


The screenshot shows the "Project settings" page for the same project. The "General" tab is selected, indicated by a red box. Other tabs include Cloud Messaging, Integrations, Service accounts, Data privacy, Users and permissions, and App Check. The "Your project" section contains fields for Project name, Project ID, Project number, Default GCP resource location, and Web API Key. The "Environment" section shows the Environment type as "Unspecified". The "Public settings" section includes fields for Public-facing name (set to "project-47929403076") and Support email (set to "Not configured"). The "Your apps" section lists a "Web apps" entry with the URL "https://web.firebaseioapp.com" and an "App nickname" of "web". A red box highlights the "Add app" button at the bottom right of the "Your apps" section.

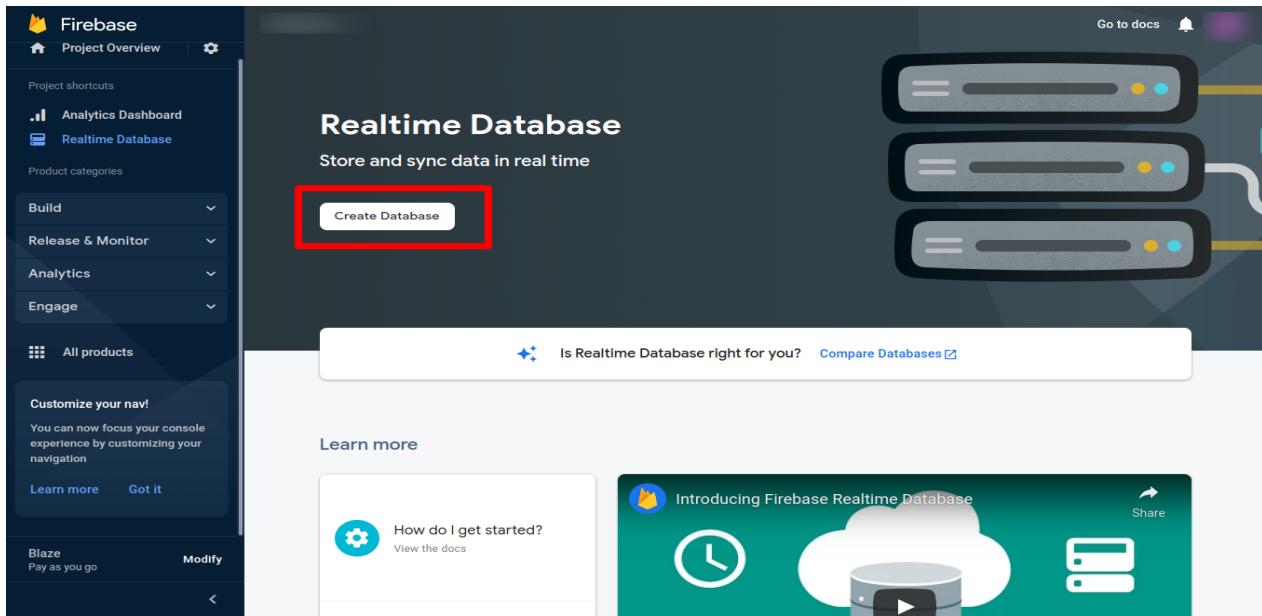


7.4 Create a Service account

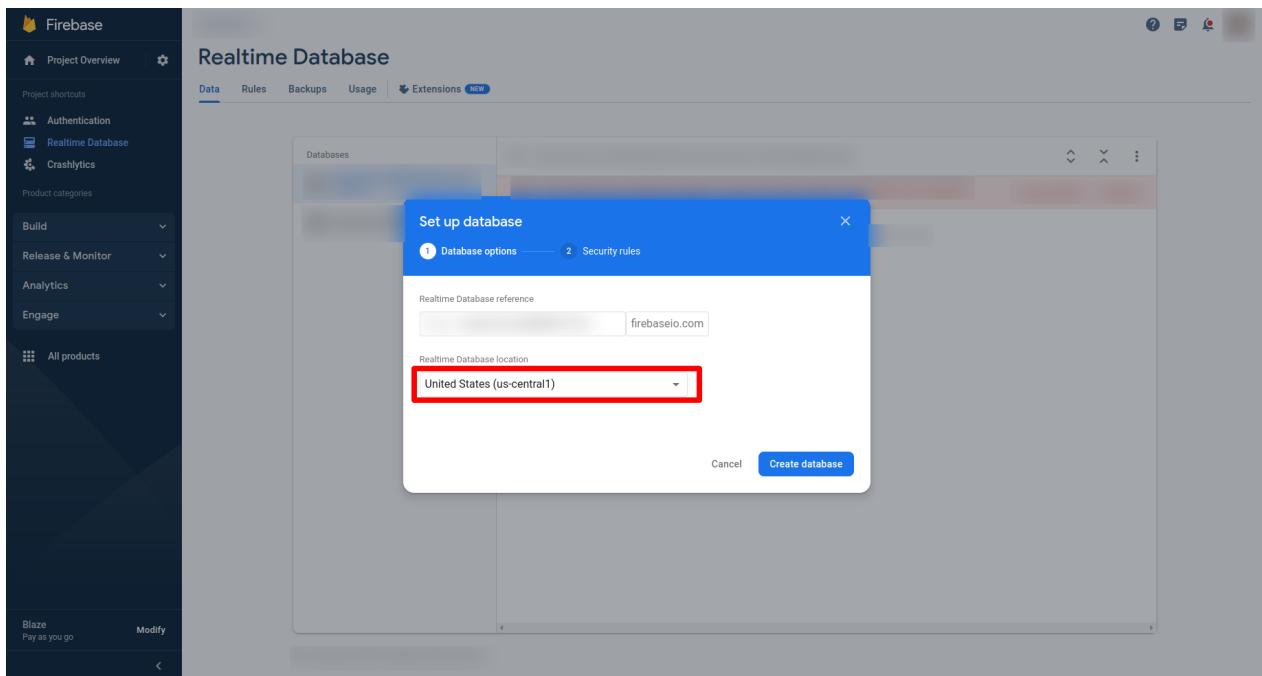
Create a Service account and download, generate a new private key and add in the setting database.

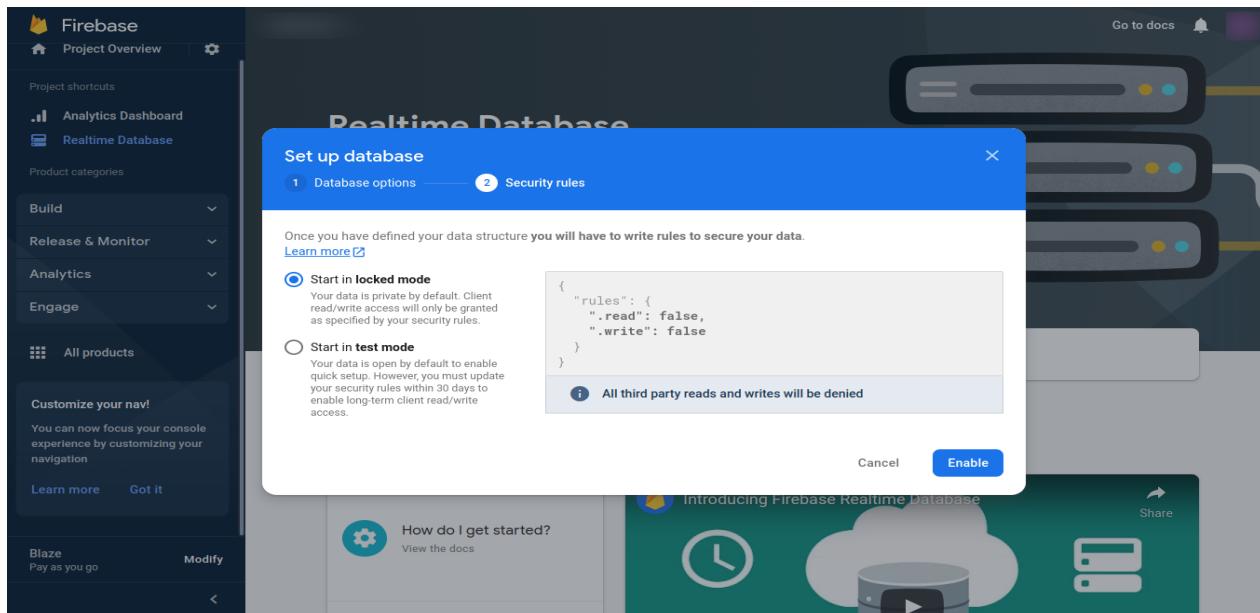
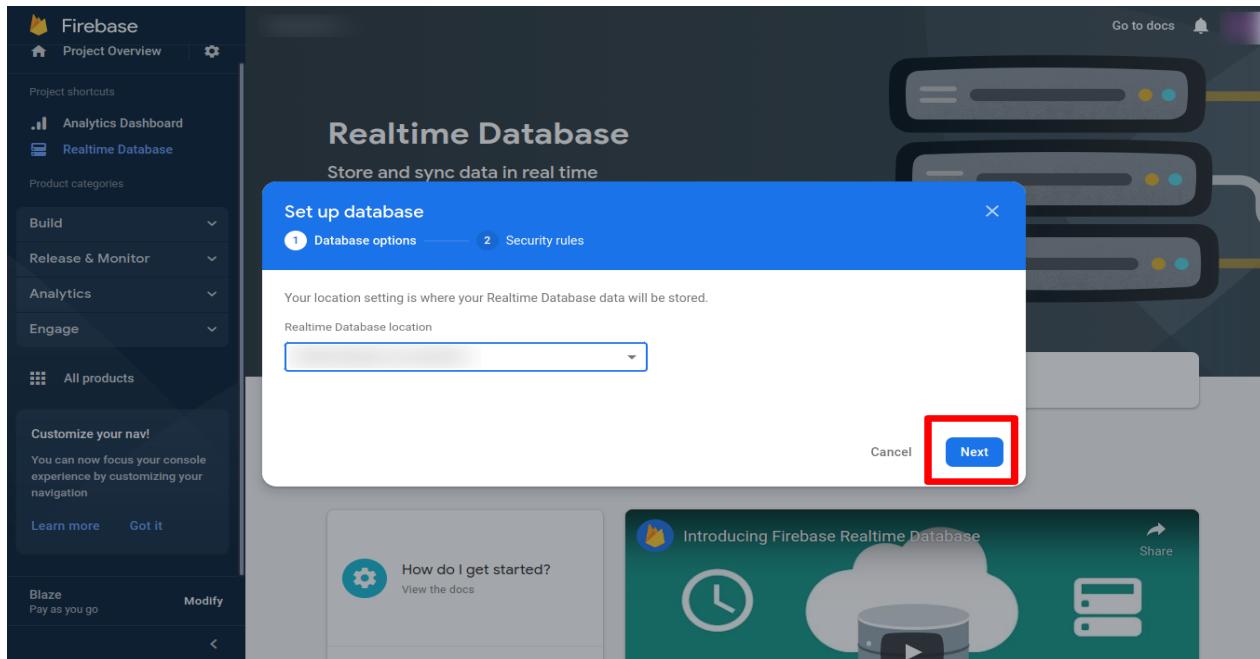


7.5 Create a Realtime Database.



NOTE: Always select the US region like this.





IMPORTANT NOTE : After selecting locked mode(false) we have to do manually rules to true and publish.

The screenshot shows the Firebase Realtime Database console. On the left, the navigation bar includes 'Analytics Dashboard' and 'Realtime Database'. The main area displays the database URL: <https://sodium-sunup-355705-default-rtdb.firebaseio.com>. A red arrow points to the URL, with the text 'save this Database URL' below it. The bottom status bar indicates the database location is 'United States (us-central1)'.

*Note -> Save this url into the database in settings collection.

7.6 Add Auth Sign-In Methods.

Enable two Sign-In methods, **Phone and Email / Password** as highlighted below.

The screenshot shows the Firebase Authentication console. The 'Sign-in method' tab is selected. A red box highlights the 'Authentication' button in the sidebar. Another red box highlights the 'Sign-in providers' section. Within this section, a red box highlights the 'Native providers' dropdown, which shows 'Email/Password' and 'Phone' selected. A red box also highlights the 'Add new provider' button. The bottom status bar indicates the project is on the 'Spark' plan.

7.7 Add this firebase config

Replace this firebase config(object) in your project in all panels listed [here](#) and setting collection in the database.

src/environments/environment.ts
src/environments/environment.prod.ts

SDK setup and configuration

npm (selected) CDN Config

If you're already using npm and a module bundler such as webpack or Rollup, you can run the following command to install the latest SDK:

```
$ npm install firebase
```

Then, initialize Firebase and begin using the SDKs for the products you'd like to use.

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
// For Firebase JS SDK v7.20.0 and later, measurementId is optional
const firebaseConfig = {
  apiKey: "...",
  authDomain: "...",
  databaseURL: "...",
  projectId: "...",
  storageBucket: "...",
  messagingSenderId: "...",
  appId: "...",
  measurementId: "..."
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
```

Update setting collection

Robo 3T - 1.3

File View Options Window Help

Welcome db.getCollection('settings').find({})

settings 3.57 sec.

Key	Type
_id	Object
provider_timeout	Object
countryname	Object
adminCurrencyCode	Object
adminCurrency	Object
adminTimezone	Object
ams_notification	Object
email_notification	Object
push_notification	Object
get_referral_profit_on_card_payment	Object
get_referral_profit_on_cash_payment	Object
userEmailVerification	Object
providerEmailVerification	Object
otp	Object
providerSMS	Object
admin_phone	Object
contact_email	Object
twilio_call_masking	Object
access_key_id	Object
secret_key_id	Object
aws_lambda_name	Object
aws_lambda_arn	Object
use_aws_bucket	Object
image_base_url	Object
payments_base_url	Object
payments_base_url	Object
app_base_url	Object
history_base_url	Object
is_ride_share	Object
is_split_payment	Object
max_order_size	Object
admin_email	Object
default_Sec_h_passive	Object
scheduled_request_prc_start_minute	Object
number_of_try_for_scheduled_request	Object
is_public_demo	Object
is_promote_update_trip	Object
stripe_publishable_key	Object
paystack_secret_key	Object
paystack_publishable_key	Object
payu_key	Object
payu_salt	Object
payment_gateway_type	Object
nautilus_cancer_key	Object

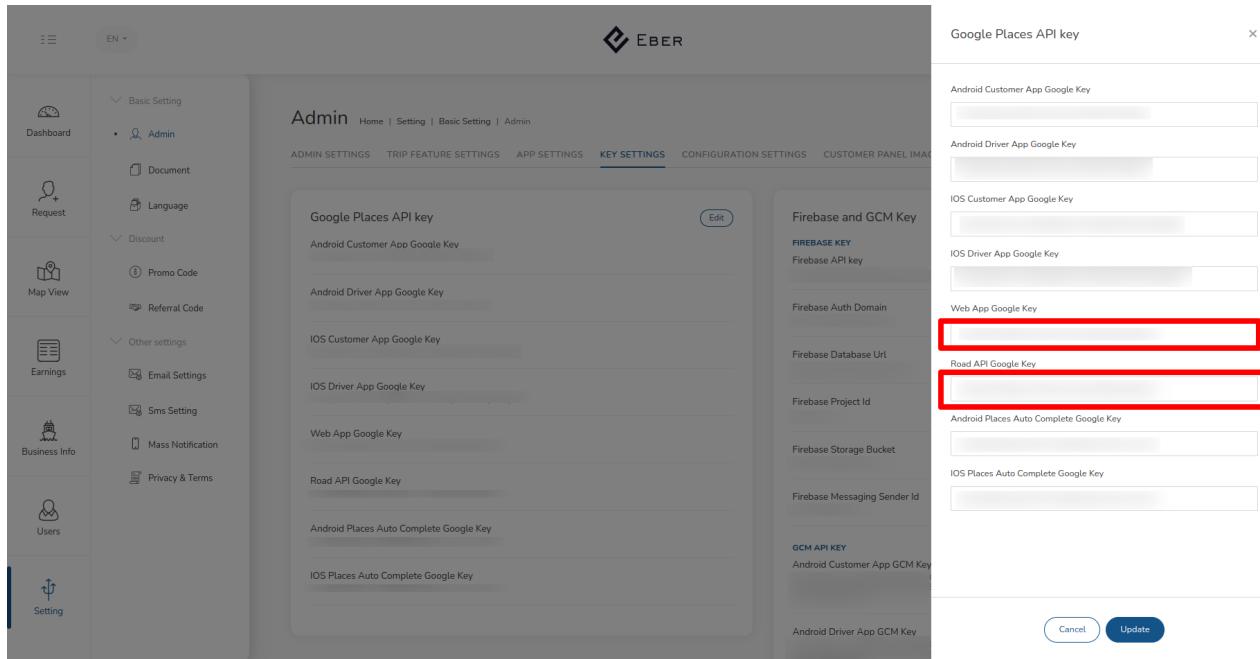
8. Basic setup on Admin panel

Basic setup on admin panel for making application fully functional.

Now our applications are live so now we need to do some configurations to make our application fully functional. We need to set Google API keys to make maps work.

As shown in the image head to app settings and press the edit button to set your to google API keys.

8.1 Set up Google api keys



- In the admin panel, navigate to settings, admin, and then key settings.
- Press the edit button located at the top-right corner in User App settings
- Place your google API keys in appropriate fields.

8.2 Set up Notification Firebase Keys

- In the admin panel, navigate to **settings->admin -> Key settings**.

The screenshot shows the EBER Admin dashboard with a sidebar containing various settings like Dashboard, Request, Map View, Earnings, Business Info, Users, and Setting. The main content area shows 'Admin' settings with tabs for ADMIN SETTINGS, TRIP FEATURE SETTINGS, APP SETTINGS, KEY SETTINGS (which is selected), CONFIGURATION SETTINGS, and CUSTOMER PANEL IMAGES. In the KEY SETTINGS tab, there's a list of API keys: Google Places API key, Android Customer App Google Key, Android Driver App Google Key, iOS Customer App Google Key, iOS Driver App Google Key, Web App Google Key, Road API Google Key, Android Places Auto Complete Google Key, and iOS Places Auto Complete Google Key. A modal window titled 'Firebase and GCM Key' is overlaid, containing sections for FIREBASE KEY (with 'Firebase API key' and 'Firebase Auth Domain'), FIREBASE AUTH (with 'Firebase Auth Domain' and 'Firebase Database Url'), FIREBASE PROJECT (with 'Firebase Project Id' and 'Firebase Storage Bucket'), FIREBASE MESSAGING (with 'Firebase Storage Bucket', 'Firebase Messaging Sender Id', and 'Firebase Messaging Sender Id'), and GCM API KEY (with 'Android Customer App GCM Key', 'Android Driver App GCM Key', and 'Android Driver App GCM Key'). Buttons for 'Cancel' and 'Update' are at the bottom right of the modal.

- Now enter your firebase keys in appropriate fields.

9. Chat Push Firebase

9.1 Open putty with Backend instance of your project

Start using the following command:

```
sudo npm install -g firebase-tools --unsafe-perm
cd /var/www/html OR cd /root
sudo mkdir functions-firebase
sudo firebase login --no-localhost
```

=>you will get a link

open that link to your browser and log in to your Gmail then you will get
Success in terminal

```
cd functions-firebase
sudo firebase init
```

select cloud function

use an existing project

select project

select javascript

Deny ESLint

install dependencies with npm Y

cd functions

then edit your index.js file (this index file you have to make as your DB structure i am giving here my demo file)

sudo nano index.js

Add(copy and paste) the following code in index.js file.

// Code Start

```
const functions = require('firebase-functions');
const admin = require('firebase-admin');
admin.initializeApp(functions.config().firebase);
exports.sendAdminNotification =
  functions.database.ref('/{Topic}/{UniqueId}/{OrderId}/').onWrite(event => {
    var topic = event.after._path.split('/');
    event.after._data.chat_type = String(event.after._data.chat_type);

    const payload = {
```

```

notification: {
    'chat_type': String(event.after._data.chat_type),
    'id': event.after._data.id,
    'is_read': String(event.after._data.is_read),
    'body': event.after._data.message,
    'sender_type': String(event.after._data.sender_type),
    'time': event.after._data.time,
    'order_id': topic[1]
}
};

if(!event.after._data.is_read)
{
    return admin.messaging().sendToTopic(event.after._data.receiver_id,
payload)
    .then(function(response) {
        return console.log('Notification sent successfully:',
response);
    })
    .catch(function(error) {
        return console.log('Notification sent failed:', error);
    });
}

});

// Code End

cd ..
sudo firebase deploy

```

10. Google API key

10.1 Open Google cloud console.

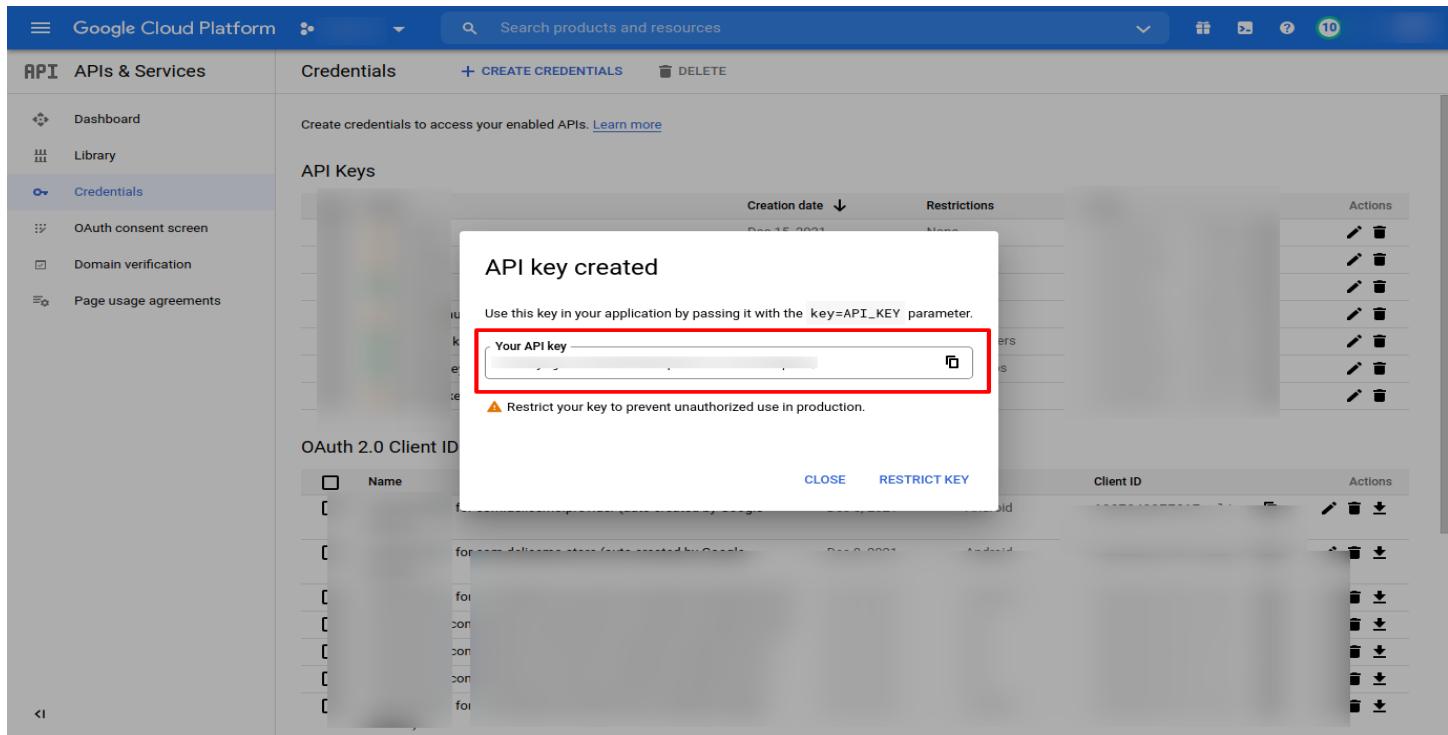
Go to console [Click me](#)

10.2 Copy that API key

Copy that API key and save it in a text file and use it in Your project

10.3 Set this API Key in the given files.

The screenshot shows the Google Cloud Platform API & Services Credentials page. The left sidebar has 'Credentials' selected. The main area shows a table for 'API Keys'. A red box highlights the '+ CREATE CREDENTIALS' button at the top right of the table header. Another red box highlights the 'API key' row in the table, which describes it as identifying your project using a simple API key to check quota and access. The table columns are 'Name', 'Restrictions', 'Key', and 'Actions'. Below this table is another section for 'OAuth 2.0 Client IDs' with a similar table structure.



10.4 Activate following Google API Keys

Directions API
Distance Matrix API
Geocoding API
Geolocation API
Maps JavaScript API
Maps SDK for Android
Maps SDK for iOS
Maps Static API
Places API
Roads API

10.5 Configure Auth consent screen if not

The screenshot shows the Google Cloud Platform interface for managing APIs and services. The left sidebar has 'APIs & Services' selected, with 'Enabled APIs & services' expanded. Under 'OAuth consent screen', the 'Edit app registration' tab is active. The main area displays the 'App information' section, which includes fields for 'App name' (highlighted with a red box) and 'User support email' (also highlighted with a red box). Below this is the 'App logo' section, followed by 'App domain' and 'Authorized domains'. At the bottom is the 'Developer contact information' section, where 'Email addresses' is highlighted with a red box. A sidebar on the right provides instructions for the configuration.

1. The logo and name of your app
A logo is recommended, but it is not required
2. The email address that users can use to contact

2nd step -> press button save and continue

3rd step -> test user save and continue

4th step -> review summary and back to dashboard

10.6 Set up APIs in code

Admin

- > src/environments/environment.ts
- > src/environments/environment.prod.ts
- > src/index.html

Corporate, Dispatcher, Driver, Partner, User, Hotel, Hub.

- > src/environments/environment.ts
- > src/environments/environment.prod.ts

For reference [click me](#).

11. Social login using Gmail

11.1 Open google console

Open google console and select your project. [Click me](#)

11.2 Select Web client auto service.

* Note -> Set google app id in the project after generating.

The screenshot shows the Google Cloud Platform API & Services Credentials page. The left sidebar has 'APIs & Services' selected. Under 'Credentials', 'API Keys' and 'OAuth 2.0 Client IDs' are listed. The 'OAuth 2.0 Client IDs' section shows a table with one row highlighted by a red box. The row details are: Name 'Web client (auto created by Google Service)', Creation date 'NOV 12, 2021', Type 'Web application', and Actions. The 'Actions' column contains edit and delete icons. The entire row for the 'Web client' is also highlighted with a red box.

11.3 Add your domain name to URI

The screenshot shows the 'Credentials' section of the Google Cloud Platform API & Services interface. It displays fields for 'Client ID' and 'Client secret' (both highlighted with a red box), and a list of 'Authorized JavaScript origins' (also highlighted with a red box). Below this is the 'Authorized redirect URIs' section, which contains a single entry ('https://[REDACTED].com/auth/handler') and a '+ ADD URI' button. At the bottom right are 'SAVE' and 'CANCEL' buttons.

11.4 And replace **client_id** in your project Database.

For hint search this string '`apps.googleusercontent.com`'

- Add client_id in driver panel (driver-panel/src/app/views/app/auth/auth.component.ts)
- User panel (user-panel/src/app/containers/pages/auth-modal/auth-modal.component.ts)

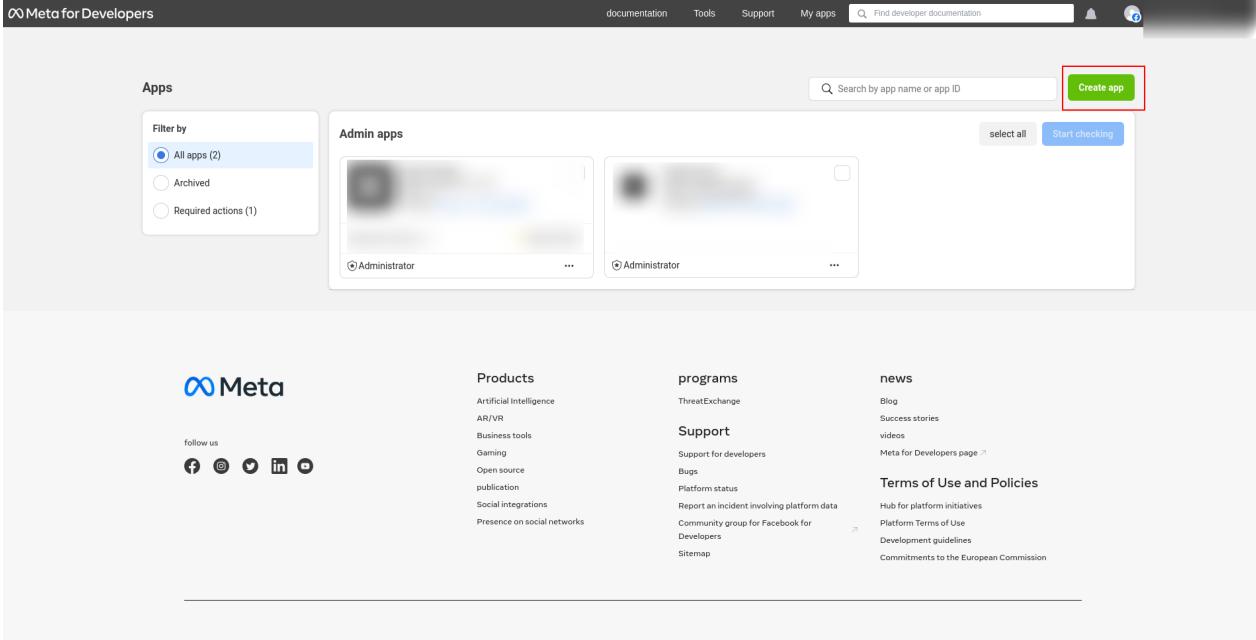
12. Facebook Login.

12.1 Open Facebook developer console.

Account [Register](#) or [Login](#)

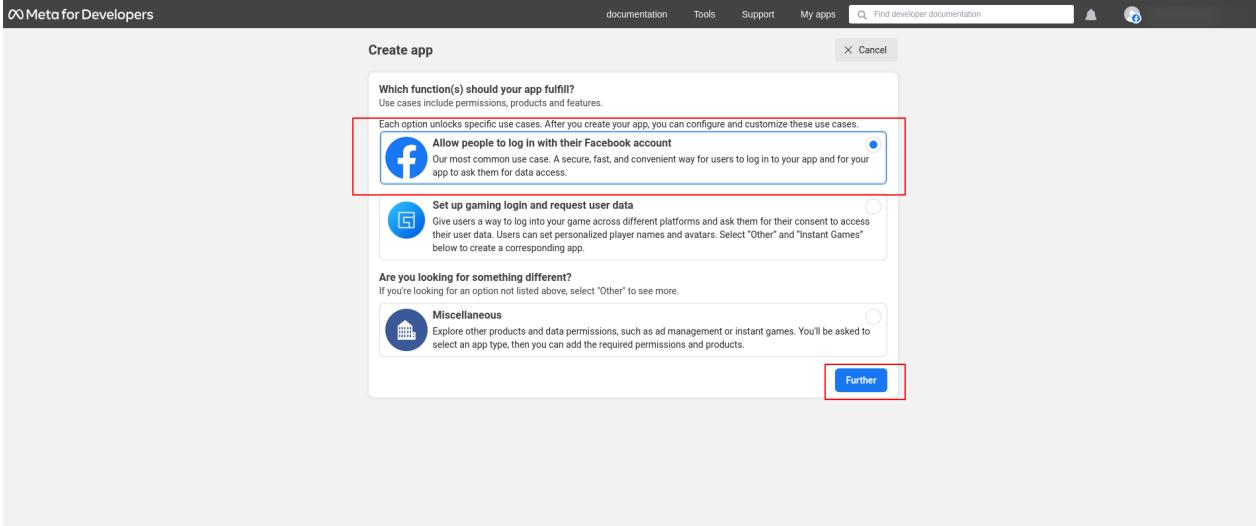
12.2 Create app

You will have to create a total 2 apps for your project.



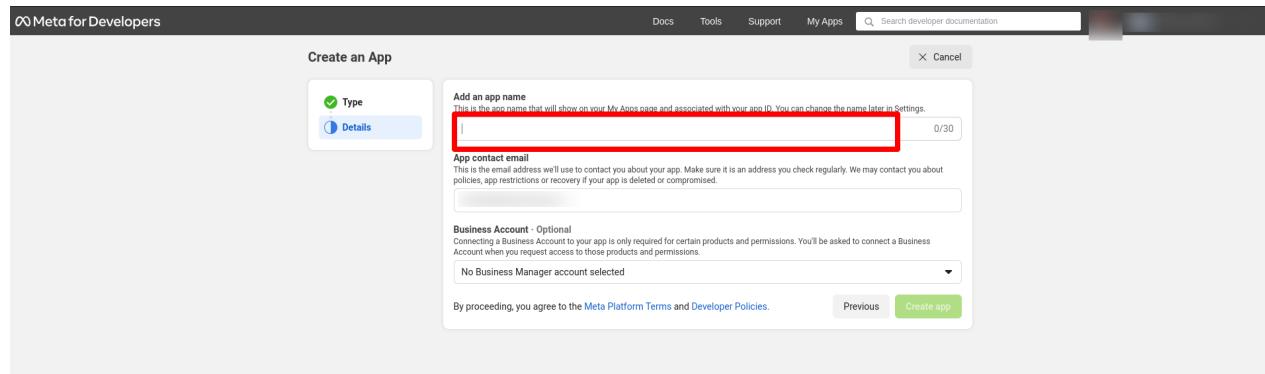
The screenshot shows the 'Apps' section of the Meta for Developers website. At the top right, there is a green button labeled 'Create app' which is highlighted with a red box. Below it, there is a search bar and some filtering options. The main area displays two 'Admin apps' cards. Each card has a blurred profile picture, the word 'Administrator' next to a radio button, and three dots at the bottom right. On the left side, there is a sidebar with a 'Filter by' dropdown set to 'All apps (2)', and other options like 'Archived' and 'Required actions (1)'. At the bottom of the page, there is a footer with links to various Meta products and services.

1 -> select accordingly



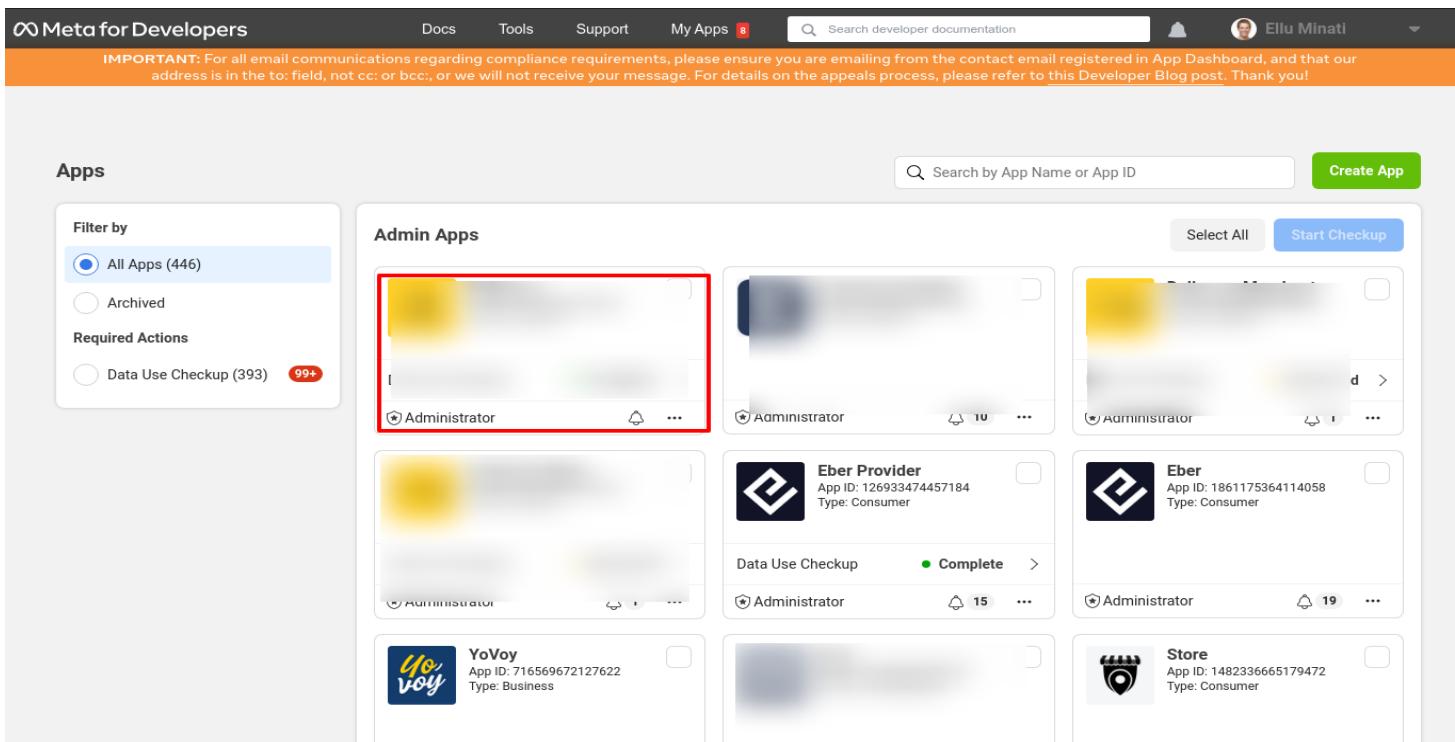
The screenshot shows the 'Create app' wizard on the Meta for Developers site. The current step is 'Which function(s) should your app fulfill?'. It asks for specific permissions and features. One option, 'Allow people to log in with their Facebook account', is selected and highlighted with a red box. Other options include 'Set up gaming login and request user data' and 'Miscellaneous'. At the bottom right of the wizard, there is a blue 'Further' button.

2 -> your app name



The screenshot shows the 'Create an App' interface on the Meta for Developers website. The 'Type' tab is selected. The 'App name' field is highlighted with a red box. Other fields include 'App contact email', 'Business Account - Optional', and a note about connecting a Business Account. At the bottom, there's a note about agreeing to terms and policies, and a 'Create app' button.

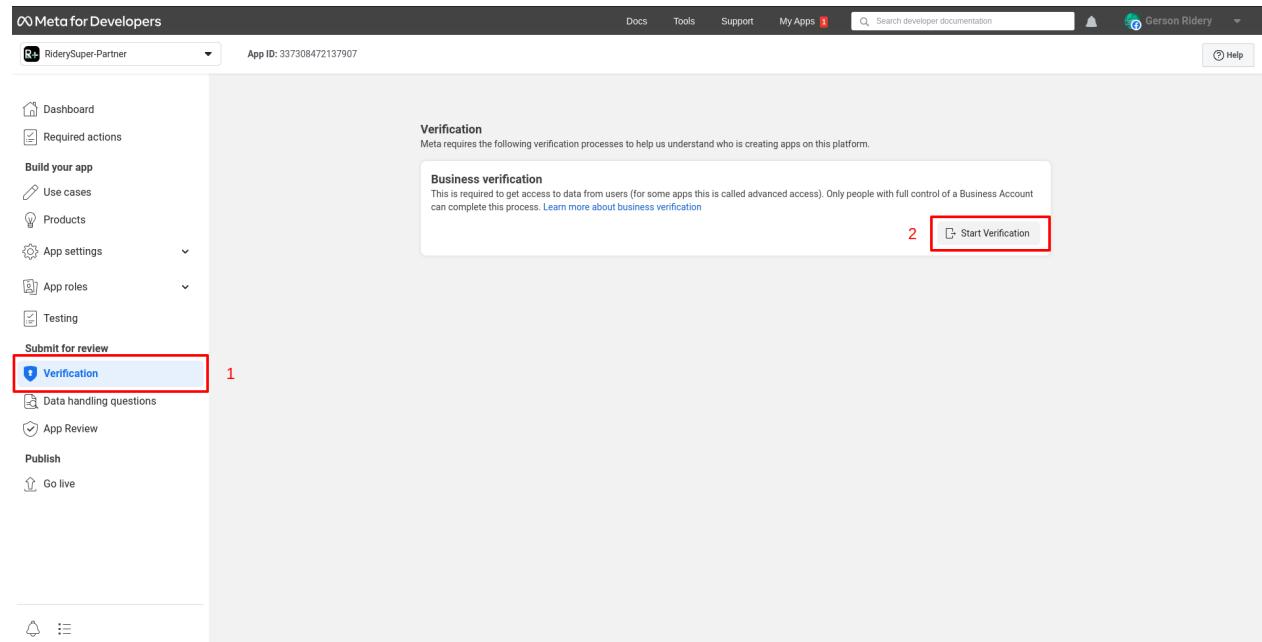
12.3 Select your project app.



The screenshot shows the 'My Apps' dashboard on the Meta for Developers website. The 'Admin Apps' section is displayed, showing a list of apps. One app card, 'YoVoy', is highlighted with a red box. Other cards include 'Eber Provider', 'Eber', and 'Store'. Each card displays the app icon, name, App ID, Type, and Data Use Checkup status. A sidebar on the left shows filter options for 'All Apps (446)', 'Archived', and 'Data Use Checkup (393)'. A top bar includes navigation links like Docs, Tools, Support, My Apps, and a search bar.

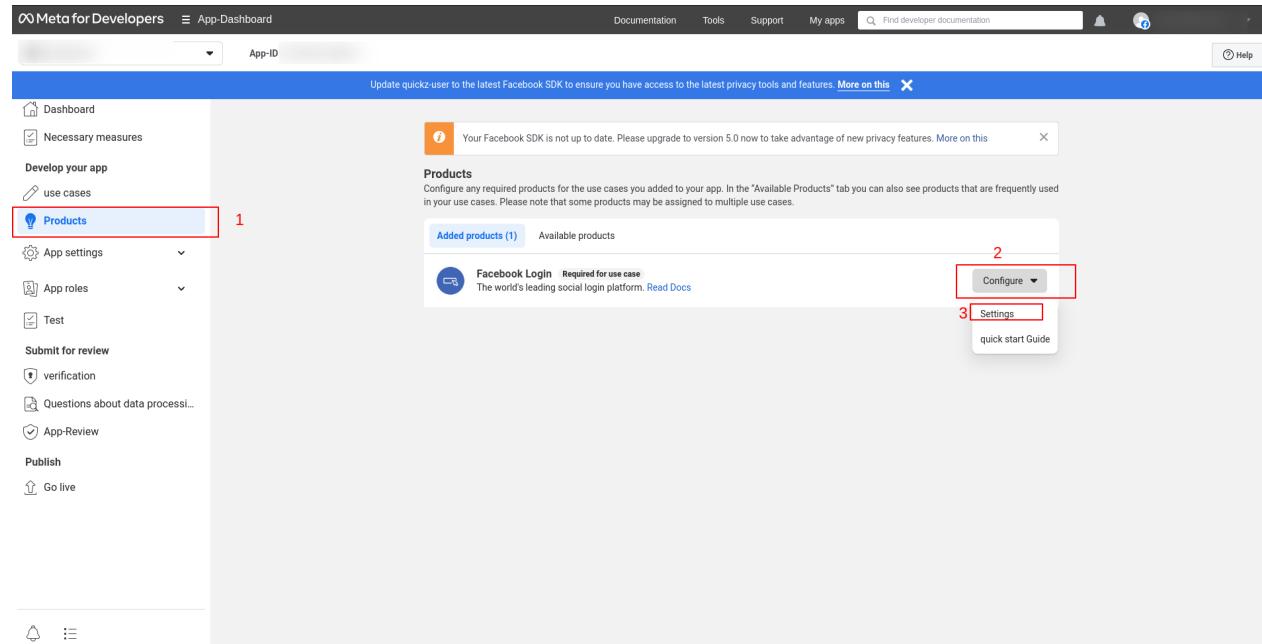
12.4 Click on Verification

Click on verification -> Start verification and fill all the basic required details and complete the process before moving ahead.



12.5 Click on Products

Click on Products -> Configure -> Settings



12.6 Add Domains of your project

1. Toggle all to yes, 2. Fill the domains, from which you will use, 3.Click and continue.

The screenshot shows the Facebook App Dashboard for 'quickz-user'. On the left, there's a sidebar with various app development and review options. The main area is titled 'Client-OAuth-Einstellungen' and contains several configuration sections with toggle switches. A red box labeled '1' highlights the 'Client-OAuth-Anmeldung' section. Another red box labeled '2' highlights the 'Gültige OAuth Redirect URIs' input field, which currently contains 'https://'. A third red box labeled '3' highlights the 'Änderungen speichern' (Save changes) button at the bottom right.

12.7 Give advance access

1. Click use cases
2. Click Edit

The screenshot shows the Facebook App Dashboard for 'quickz-user'. The sidebar has a 'use cases' link, which is highlighted with a red box labeled '1'. The main content area shows a list of use cases. One use case, 'Allow people to log in with their Facebook account', is highlighted with a red box labeled '2' over its 'Edit' button. Other use cases listed include 'Authentication and account creation' and 'Use additional Facebook usage data for personalization'.

3. Add

Use case permissions
These permissions are related to the use case you selected. Make sure to add the permissions you need for your app and check the testing and verification requirements if you want your app to go live to your users.

Most permissions require testing, verification and App Review before your app can go live to your users.

Permissions ↑	API Calls ↓ ↑	Status ↑↓	Action
email The email permission allows your app to read a person's primary email address. ① Full Description ② Requirements	—	—	3 Add
public_profile The public_profile permission allows an app to read the Default Public Profile Fields on the User node. This permission is automatically granted to all apps. ① Full Description ② Requirements	0	Ready for testing	Remove

12.8 Complete basic settings

Complete basic settings, privacy-terms condition, data deletion url, logo change if want, category business and pages, add website, app, ios.

12.9 Data use check up

Data use check up if it already completes its signal as green. If not you have to complete 2 easy setup. Only tap yes this both steps

12.10 Save the changes

App ID: [REDACTED] App Mode: Development Live App type: Consumer

Category: Business and pages Find out more information about app categories here

Verifications: Business verification Permissions and features require business verification to access certain types of data. An app admin can start the verification process at any time. Learn more about business verification Start Verification

Data Use Checkup: Facebook has introduced an annual Data Use Checkup. An app admin must certify compliance with allowed usage and all applicable terms and policies. Learn More Checkup Complete Data Use Checkup was completed by an app admin on [REDACTED]. At this time, no further action is required.

Data Protection Officer contact information: The General Data Protection Regulation (GDPR) requires certain companies doing business in the European Union to designate a Data Protection Officer who people can contact for information about how their data is being processed. This contact information will be available to people on Facebook along with other information about your app or website. Learn More Name - Optional Email [REDACTED]
Address Street Address Discard **Save changes**

IMPORTANT:

12.11 Replace FB App ID

Add **fb appId** in user panel and driver panel

Search for (**FB.init**) in the below mentioned file.

-> Project-Folder/driver-panel/src/app/views/app/auth/auth.component.ts

-> Project-Folder/user-panel/src/app/containers/pages/auth-modal/auth-modal.component.ts

13. Note For Folder Structure.

In the delivered code you might not get following files/folders because it will be generated or added after installation.

Some of the folders contain dynamic files/folders.

13.1 List of empty folders in clone:

- data/partner_document : Uploaded Images For Partner Document
- data/partner_profile : Uploaded Images For Partner Profile
- data/provider_document : Uploaded Images For Driver Document
- data/provider_profile : Uploaded Images For Driver Profile
- data/service_type_images : Uploaded Images For Vehicle Image
- data/service_type_map_pin_images : Uploaded Images For Vehicle Map Pin
- data/user_document : Uploaded Images For User Document
- data/user_profile : Uploaded Images For User Profile

13.2 File/Folder structure

The following file/folder will get auto generated after the installation

- node_modules : Auto Generate After Installation Of Node Dependencies
- package-lock.json : Auto Generate After Installation Of Node Dependencies
- log_files : Auto Generate Files For Logs
- data/xlsheet : Auto Generate xl sheets
- config/data : Auto Generate For Temporary Image Upload Cache

13.3 The following keys/file

The following keys/files are removed that need to be configured from a client account that needs to be pushed in the repository provided by the IOS developer.

- backend/server/app/ios_push/push_file.p8 - need to be added by admin from client account.

13.4 Overview of code.

Here the following instruction is for understanding the code quickly.

- Backend
 - There are a total of 4 backends in this project and all are necessary.
 - **server**
 - **history-earning**
 - **payments**
 - **mass_notification**
 - From the above list, the **server** is main. We can say, others are all very easy to understand.
 - **app** folder contains all the files like routes, controllers, models, utils etc.
 - **config** folder contains the env and basic configuration files.
 - **data** folder contains all the images, icons etc.
 - **documentation** folder contains files that explains the project.
 - The **app** and the **config** folder in the folder other than the server are similar to the server's app and config folder as explained above.
- Panels(Frontend)
 - All the panels follow Angular folder structure, therefore knowledge of Angular is must for understanding the panels code.

14. Stripe Configuration.

14.1 Login Stripe Account

Account [Login](#) or [Register](#)

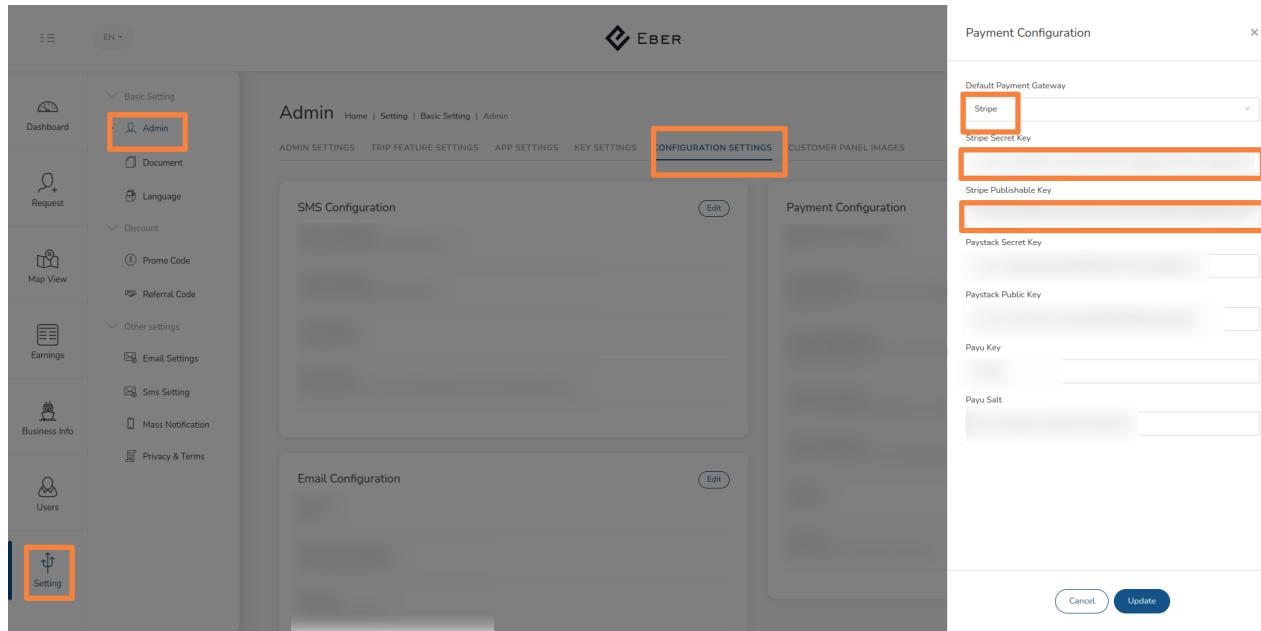
Keep Test mode toggle on as shown in the image below.

Use the following highlighted key for future reference. i.e, “**Publishable Key**” and “**Secret Key**”

The screenshot shows the Stripe dashboard interface. At the top, there's a navigation bar with links for Home, Payments, Balances, Customers, Products, Reports, Connect, More, Developers, and a prominent 'Test mode' toggle switch which is turned on. Below the navigation is a section titled 'Finish your profile to activate payments' with a 'Continue activating →' button. To the right is a decorative graphic of overlapping blue and purple rectangles. The main content area has a heading 'Get started with Stripe' and two sections: 'Create your first connected account' (with a 'Start →' button) and 'Complete your Connect platform profile' (also with a 'Start →' button). On the right side of this section, there are several links: 'Not sure where to start?', 'Explore all products', 'For developers' (which is selected), and 'Test mode'. Under 'For developers', the 'Publishable key' and 'Secret key' fields are displayed; both are highlighted with a red rectangle. Below this is a 'View docs →' link. At the bottom of the page, there's a 'Today' section showing 'Gross volume' and 'Yesterday' dropdown menus, and a 'View' button.

14.2 Use this test mode private and public key, and save this key in the Admin panel.

Navigate to, **Admin panel - > Setting -> Admin -> Configuration Settings -> Payment Configuration**



14.3 Add stripe publishable key in to user panel, partner panel & driver panel in code

-> **src/environments/environment.ts**
-> **src/environments/environment.prod.ts**

14.4 And start the Connect Account

Go to the home page and navigate to the connect tab and if it is not visible search for **connect** in the search bar.

The screenshot shows the Stripe Connect setup interface. At the top, there's a navigation bar with links for Home, Payments, Balances, Customers, Products, Reports, and a highlighted 'Connect' button. Below the navigation is a search bar and some developer settings like 'Create', 'Help', 'Test mode', and 'Developers'. The main content area is titled 'Get started with Connect' and includes several steps: 'The email kual.app.servicos@gmail.com is verified' (green checkmark), 'Learn about Connect' (with a 'Read' button), 'Add functionality to your platform', 'Complete your platform profile' (marked as required), 'Test Connect', 'Add business details to activate your account' (status: In progress), and 'Create your first live connected account' (marked as required). To the right, there's a sidebar titled 'Your platform's products' with a 'Payments' section and a 'Enable payments and payouts' button.

14.5 Activate Apple Pay

Need to activate the Apple pay method before using its feature.

1. Go to settings
2. Payment Methods

The screenshot shows the Stripe Settings page. The top navigation bar includes Home, Payments, Balances, Customers, Products, Billing, Reports, Connect, and More. The 'Connect' button in the top navigation bar is highlighted with a red box. The main content area is divided into two main sections: 'Product settings' and 'Business settings'. Under 'Product settings', there are cards for Payments, Billing, Connect, Radar, and Data Pipeline. Under 'Business settings', there are cards for Your business, Team and security, and Compliance. A sidebar on the right is titled 'Discover more products' and lists various Stripe products like Payments, Billing, Connect, Radar, etc.

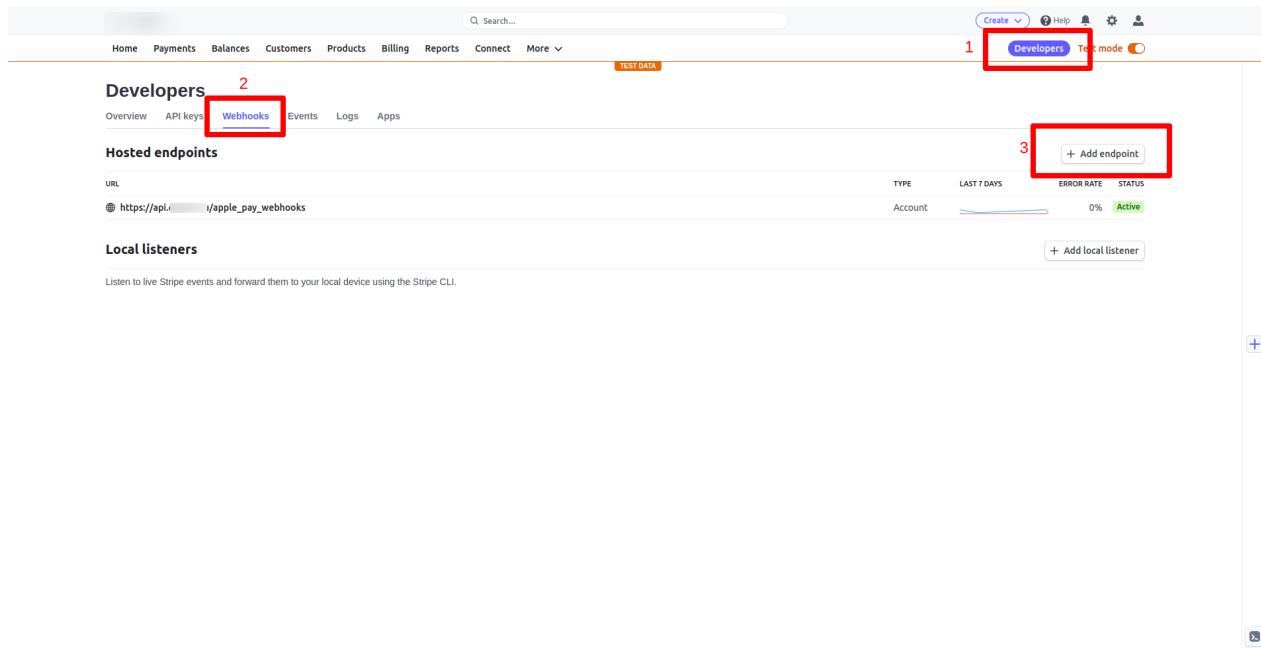
Under the you account click on **Edit Settings**

1. Default
2. Apple Pay
3. Turn on (here in the screenshot it is Active, but for you it will be Turn on button)

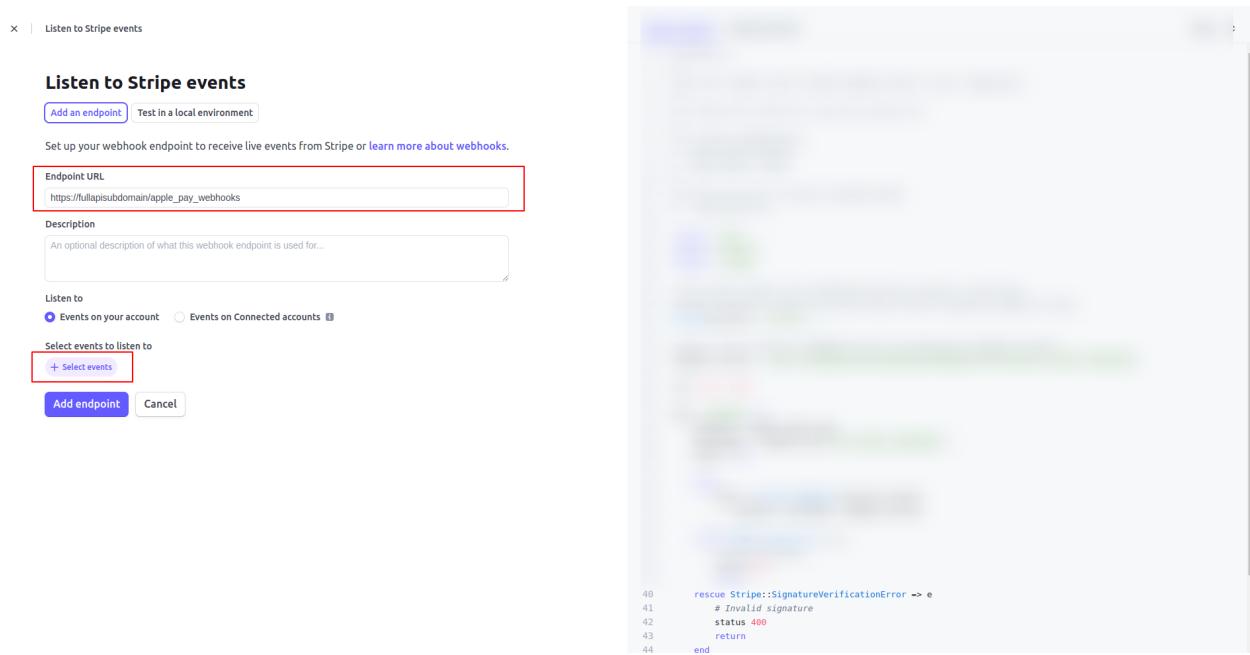
And then complete the process.

14.6 Add Webhook Url for Using Apple pay.

Navigate to, 1. Developer -> 2. Webhooks -> 3. Add endpoint



Fill the full url same as given below and replace the “fullapisubdomain” with your api url sub-domain. i.e, https://fullapisubdomain/apple_pay_webhooks
Then click on “Select events”.



Select 1. **Charge** -> 2. **charge.captured** -> 3. **charge.succeeded**, and then click
4. Add events

Select events to send

Sample endpoint Received events Ruby

17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58

Events

- Billing Portal: 3 events
- Capability: 1 event
- Cash Balance: 1 event
- Charge: 2 selected
 - charge.captured: Occurs whenever a previously uncaptured charge is captured.
 - charge.expired: Occurs whenever an uncaptured charge expires.
 - charge.failed: Occurs whenever a failed charge attempt occurs.
 - charge.pending: Occurs whenever a pending charge is created.
 - charge.refunded: Occurs whenever a charge is refunded, including partial refunds.
 - charge.succeeded: Occurs whenever a charge is successful.
 - charge.updated: Occurs whenever a charge description or metadata is updated, or upon an asynchronous capture.
 - charge.dispute.closed: Occurs when a dispute is closed and the dispute status changes to "closed", "submitted_for_review", or "won".
- Customer: 13 events

Add events Cancel

Click the **Add endpoint** button at the end and it will create an endpoint.

Listen to Stripe events

Add an endpoint Test in a local environment

Set up your webhook endpoint to receive live events from Stripe or [learn more about webhooks](#).

Endpoint URL: https://fullapisubdomain.apple_pay_webhooks

Description: An optional description of what this webhook endpoint is used for...

Listen to
 Events on your account Events on Connected accounts

Select events to listen to
 charge.captured
 charge.succeeded
 Change events

Add endpoint Cancel

56
57 status 200
58 end

15. Twilio Configuration.

15.1 Login twilio

Account [Login](#) or [Register](#)

15.2 Purchase number which have sms and voice capability

Number	Type	Capabilities	Address Requirement	Monthly fee
[REDACTED]	Voice	SMS	MMS	Fax

Suggested numbers are in the list, purchase number as per your country code.

For example, a US number starts with code (+1).

If you are choosing a US number, you will have to register for A2P 10DLC according to the new policy of Twilio.

15.3 In the trial account you have to verify number for test purpose.

The screenshot shows the Twilio Console interface. On the left, there's a sidebar with various menu items like 'Develop', 'Monitor', 'Phone Numbers', 'Verified Caller IDs' (which is highlighted with a red box), and 'Messaging'. The main area is titled 'Verified Caller IDs' and contains a sub-section for adding a new caller ID. A modal window is open, prompting for a country (+1 United States - US), a number, and an extension. It also asks for verification via SMS or call. The 'Verify number' button is highlighted with a red box.

15.4 Check your country

Check your country to have permission to enable service for the number you verified above. Go to messaging -> settings -> geo permissions

This screenshot shows the 'Messaging Geographic Permissions' page. The sidebar has 'Geo permissions' selected under the 'Messaging' section. The main content area is divided into two sections: 'North America' and 'Asia', each containing a list of countries with their respective calling codes. At the bottom left of the page, there is a 'Save geo permissions' button, which is highlighted with a red box.

15.5 And save this Account SID,Auth Token & Phone number in Admin Panel Settings.

Twilio Console - My first Twilio account

Connect to 3rd-party applications
You'll need 3 things to use Twilio with most 3rd-party applications:

- Get a trial phone number
- Read 3rd-party Integration FAQ

Account Info

Account SID: [REDACTED]
Auth Token: [REDACTED] Show

Helpful links

How does Twilio work? Understand how to use Twilio in a 2-minute video.
API documentation Learn the basics of Twilio APIs.

Docs and Support

phone number

EBER Admin

SMS Configuration

Twilio Account SID
Twilio Auth Token
Twilio Number
Twilio Call Url

Payment Configuration

Default Payment Gateway
Stripe Secret Key
Stripe Publishable Key
Paystack Secret Key
Paystack Public Key
Payu Key
Payu Salt

CUSTOMER PANEL IMAGES

SMS Configuration

Twilio Account SID
Twilio Auth Token
Twilio Number
Twilio Call Url

Configuration Settings

Domain
Domain Email Address
Password

Customer Panel Images

Cancel Update

15.6 Twilio Call Masking

-> Go to search bar and search create twiml bin

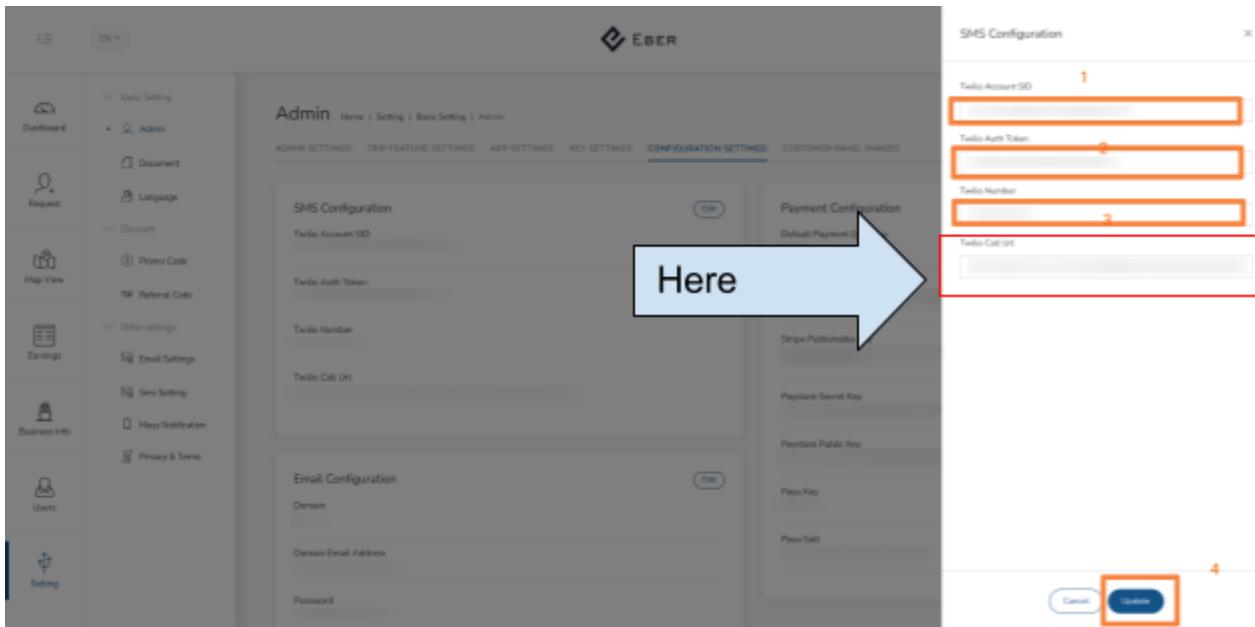
The screenshot shows the Twilio Console interface. The top navigation bar includes 'Console', 'My first Twilio account', a search bar with 'create', and 'Account' and 'Billing' dropdowns. The main menu on the left has 'Develop' selected, with options like 'Studio', 'Functions and Assets', 'Phone Numbers', 'Messaging', 'Voice', and 'TwiML Bins'. Under 'TwiML Bins', there's a sub-menu 'My TwiML bins'. The central workspace is titled 'TwiML Bin' and contains a 'Configuration' section with 'FRIENDLY NAME' and 'TWIML' fields, both of which are highlighted with red boxes. Below the configuration is a code editor window containing TwiML XML. A tooltip on the right explains TwiML and provides examples. At the bottom of the workspace are 'Create' and 'Cancel' buttons.

Any friendly name you want. Enter twiml code as below

```
<Response>
<Say>Hello From Twilio!</Say>
<Dial>
<Number>{{to}}</Number>
</Dial>
</Response>
```

Then create

> you got twiml url after this copy and paste in our admin panel settings



16. Terms & Condition Details.

16.1 Terms and Privacy Document add in to admin panel.

Navigate to, **admin panel** -> **setting** -> **Privacy & Terms** and paste the terms and policy and save it respectively.

The screenshot shows the EBER admin panel interface. On the left, there's a sidebar with various settings: Dashboard, Request, Map View, Earnings, Business Info, Users, and Setting. The 'Setting' icon is highlighted with an orange box. In the main content area, the 'Privacy & Terms' section is open. The 'CUSTOMER TERMS AND CONDITION' tab is selected and highlighted with an orange box. Below it, other tabs include CUSTOMER PRIVACY POLICY, DRIVER TERMS AND CONDITION, and DRIVER PRIVACY POLICY. The main content area displays the 'TERMS & CONDITIONS' section, which contains the text of the mobile application's terms of use. This text is also highlighted with a large orange box. At the bottom of the page, there are footer links for Home, Setting, Other settings, and Privacy & Terms.