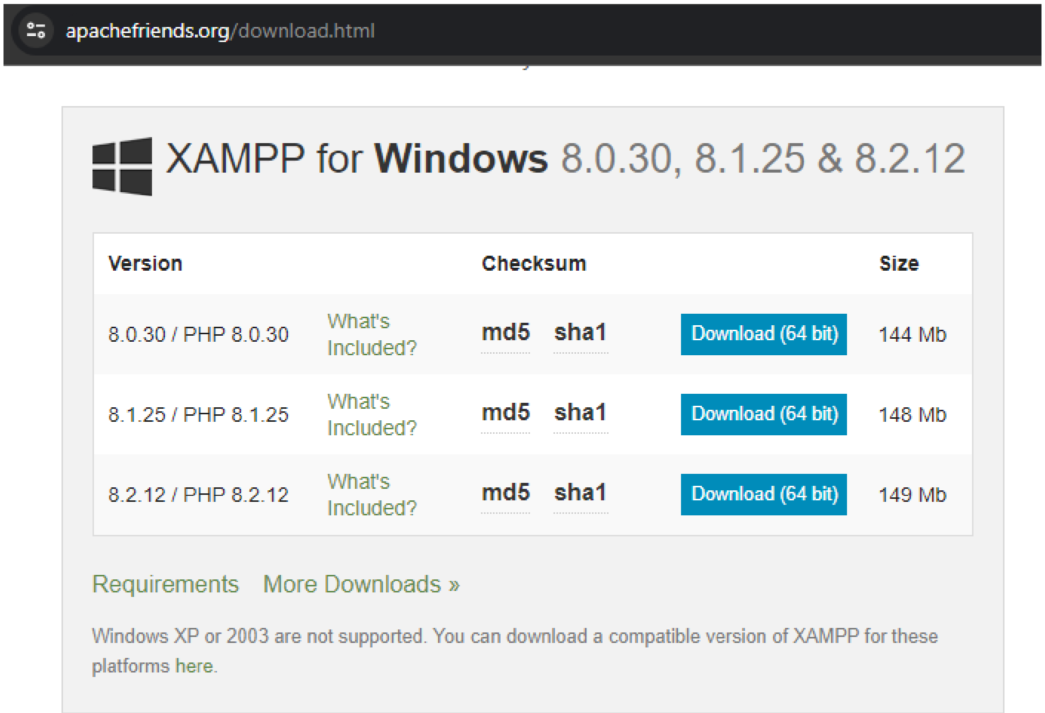
# Technical Guide to Run platform on your local system

Running a 360-degree Video Platform project (360 Vision Pro) from start to finish involves several steps, from setting up your local environment to downloading the project from GitHub and running it.

## Here’s a step-by-step guide:

### 1. Install Xampp

- Install Xampp from https://www.apachefriends.org/index.html



apachefriends.org/download.html

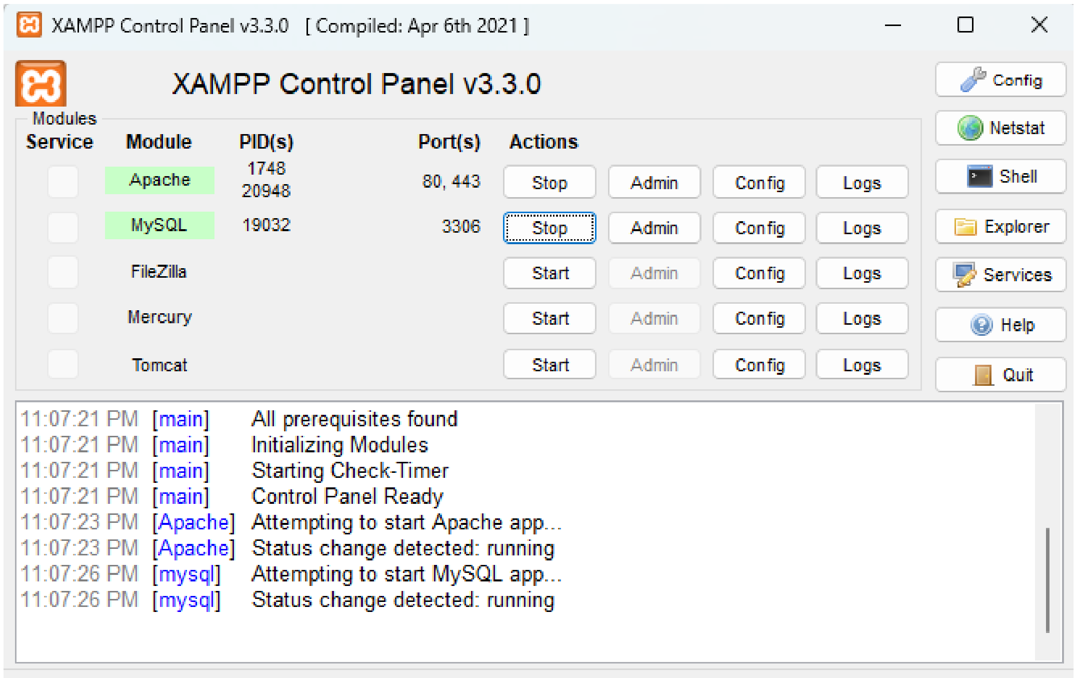
XAMPP for **Windows** 8.0.30, 8.1.25 & 8.2.12

| Version | | Checksum | | | Size |
|---|---|---|---|---|---|
| 8.0.30 / PHP 8.0.30 | What's Included? | md5 | sha1 | Download (64 bit) | 144 Mb |
| 8.1.25 / PHP 8.1.25 | What's Included? | md5 | sha1 | Download (64 bit) | 148 Mb |
| 8.2.12 / PHP 8.2.12 | What's Included? | md5 | sha1 | Download (64 bit) | 149 Mb |

Requirements   More Downloads »

Windows XP or 2003 are not supported. You can download a compatible version of XAMPP for these platforms here.

- Run the Xampp installer and open the Xampp control panel
- Make sure that you enable the Apache and MySQL services



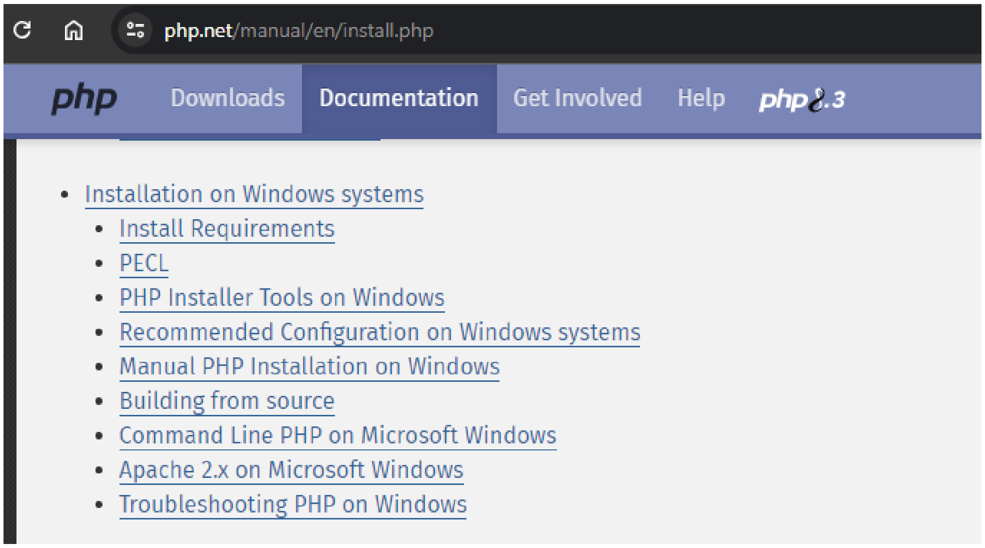
XAMPP Control Panel v3.3.0 [ Compiled: Apr 6th 2021 ]
XAMPP Control Panel v3.3.0
Modules
Config
Netstat
Shell
Explorer
Services
Help
Quit

| Service | Module | PID(s) | Port(s) | Actions | | | |
|---|---|---|---|---|---|---|---|
| | Apache | 1748<br>20948 | 80, 443 | Stop | Admin | Config | Logs |
| | MySQL | 19032 | 3306 | Stop | Admin | Config | Logs |
| | FileZilla | | | Start | Admin | Config | Logs |
| | Mercury | | | Start | Admin | Config | Logs |
| | Tomcat | | | Start | Admin | Config | Logs |

```
11:07:21 PM  [main]     All prerequisites found
11:07:21 PM  [main]     Initializing Modules
11:07:21 PM  [main]     Starting Check-Timer
11:07:21 PM  [main]     Control Panel Ready
11:07:23 PM  [Apache]   Attempting to start Apache app...
11:07:23 PM  [Apache]   Status change detected: running
11:07:26 PM  [mysql]    Attempting to start MySQL app...
11:07:26 PM  [mysql]    Status change detected: running
```

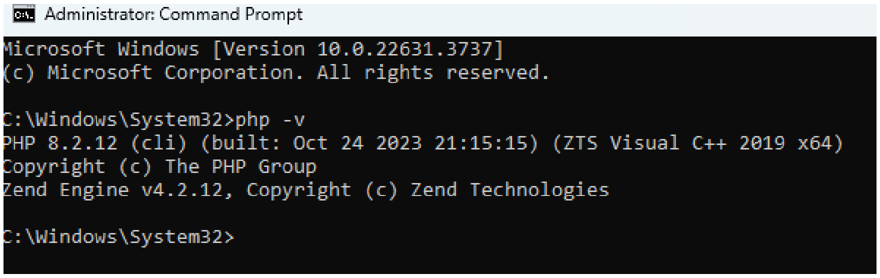
2. **Install PHP:**

- Make sure PHP is installed on your system.
- For Windows: Download and install PHP from php.net.



- Installation on Windows systems
  - Install Requirements
  - PECL
  - PHP Installer Tools on Windows
  - Recommended Configuration on Windows systems
  - Manual PHP Installation on Windows
  - Building from source
  - Command Line PHP on Microsoft Windows
  - Apache 2.x on Microsoft Windows
  - Troubleshooting PHP on Windows

- Verify installation:

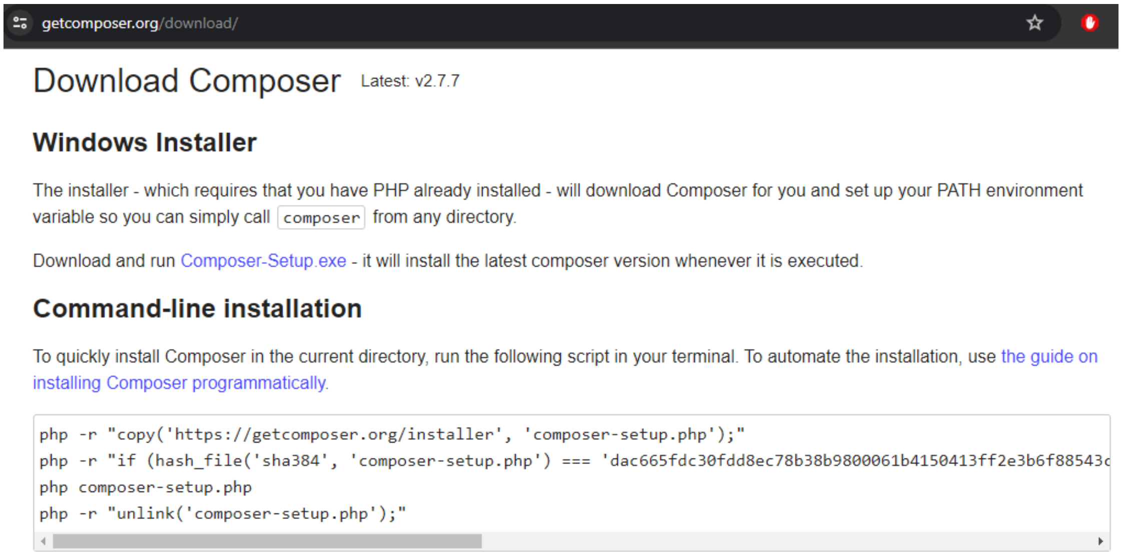


```
Microsoft Windows [Version 10.0.22631.3737]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\System32>php -v
PHP 8.2.12 (cli) (built: Oct 24 2023 21:15:15) (ZTS Visual C++ 2019 x64)
Copyright (c) The PHP Group
Zend Engine v4.2.12, Copyright (c) Zend Technologies

C:\Windows\System32>
```

- For macOS: Use Homebrew: brew install php.
- For Linux: Use your package manager, e.g., sudo apt install php for Ubuntu.

3. **Install Composer**

- Go to [https://getcomposer.org/download](https://getcomposer.org/download)



getcomposer.org/download/

Download Composer Latest: v2.7.7

**Windows Installer**

The installer - which requires that you have PHP already installed - will download Composer for you and set up your PATH environment variable so you can simply call `composer` from any directory.

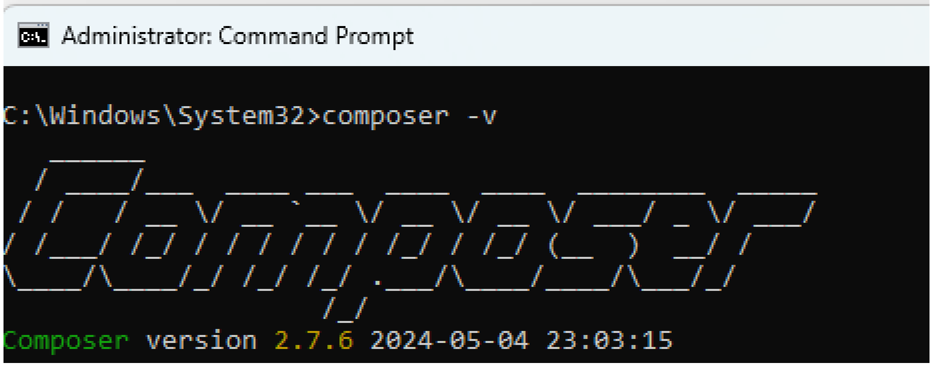
Download and run Composer-Setup.exe - it will install the latest composer version whenever it is executed.

**Command-line installation**

To quickly install Composer in the current directory, run the following script in your terminal. To automate the installation, use the guide on installing Composer programmatically.

```
php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
php -r "if (hash_file('sha384', 'composer-setup.php') === 'dac665fdc30fdd8ec78b38b9800061b4150413ff2e3b6f88543c
php composer-setup.php
php -r "unlink('composer-setup.php');"
```

- Verify installation:



Administrator: Command Prompt

```
C:\Windows\System32>composer -v
   ______
  / ____/___  ____ ___  ____  ____  ________  _____
 / /   / __ \/ __ `__ \/ __ \/ __ \/ ___/ _ \/ ___/
/ /___/ /_/ / / / / / / /_/ / /_/ (__  )  __/ /
\____/\____/_/ /_/ /_/ .___/\____/____/\___/_/
                    /_/
Composer version 2.7.6 2024-05-04 23:03:15
```

- On Windows, download and run the installer
- On Mac, copy the php commands and run in the terminal. Then copy the mv command and run in terminal. You can also install composer with Homebrew

4. **Install Laravel:**

- Install the Laravel installer globally via Composer.  
  bash
  - *composer global require laravel/installer*
- Ensure composer and laravel commands are available: Add Composer's global bin directory to your system's PATH.
  - For Windows: Add

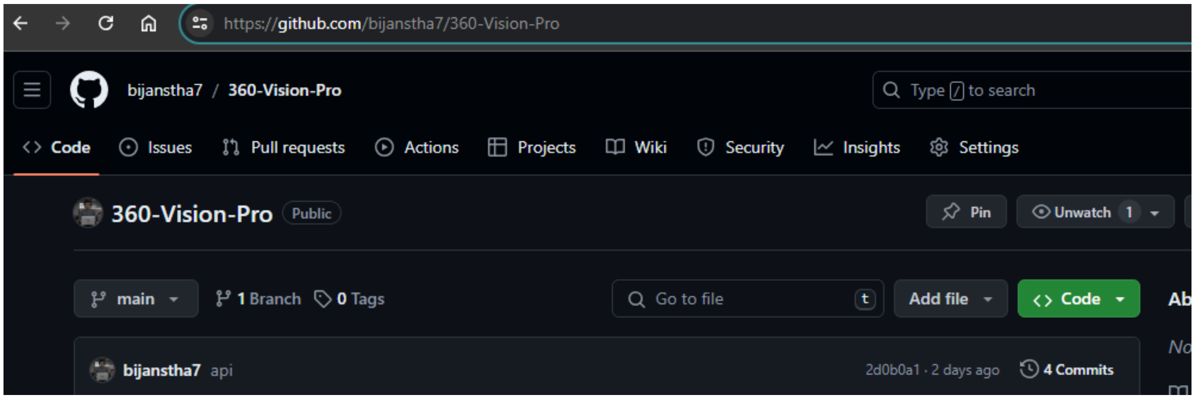
C:\Users\<Your-User>\AppData\Roaming\Composer\vendor\bin to your PATH.

- For macOS and Linux:
  Add export PATH="$PATH:$HOME/.composer/vendor/bin" to your ~/.bash_profile or ~/.zshrc.

## Steps to Run a Laravel Project from GitHub

### 1. Clone the Project from GitHub:

- Navigate to the directory where you want to store your project.
- Clone the repository using Git.



- bash

```
git clone https://github.com/bijanstha7/360-Vision-Pro.git
cd repository
```

### 2. Install Dependencies:

- Navigate to the project directory and install dependencies via Composer.
- bash

```
cd path/to/your/laravel/project
composer install
```

### 3. Set Up Environment Configuration:

- Copy the .env.example file to .env.
- bash

```
cp .env.example .env
```

- Generate an application key.
- bash

```
php artisan key:generate
```

- Configure your .env file with the necessary environment variables (e.g., database credentials).

4. **Set Up the Database:**
   - Ensure your database server is running and create a new database for your project.
   - Update your .env file with your database details.
   - plaintext
     ```
     DB_CONNECTION=mysql
     DB_HOST=127.0.0.1
     DB_PORT=3306
     DB_DATABASE=your_database_name
     DB_USERNAME=your_database_user
     DB_PASSWORD=your_database_password
     ```
   - Run database migrations.
   - bash
     ```
     php artisan migrate
     ```

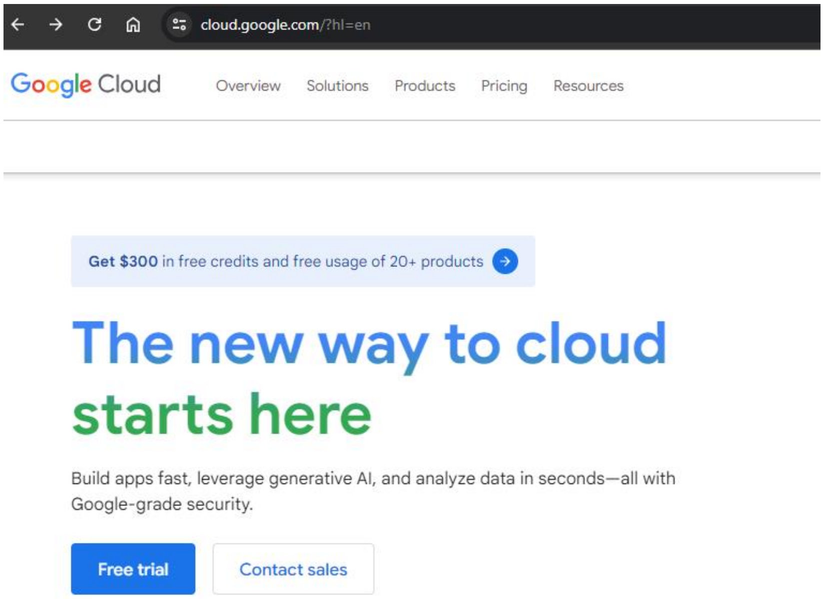
5. **Run the Development Server:**
   - Start the Laravel development server.
   - bash
     ```
     php artisan serve
     ```
   - Open your browser and go to http://localhost:8000 to see your Laravel application running.

# Technical Guide to Deploy platform on Google Cloud Platform

Deploying 360-degree Video Platform project (360 Vision Pro) from start to finish with a MariaDB database on Google Cloud Platform (GCP) involves several steps. Here’s a comprehensive guide to walk you through the entire process:

## Step 1: Set Up Google Cloud Platform

1. **Create a GCP Account:**
   - If you don’t have a GCP account, create one at [https://cloud.google.com/](https://cloud.google.com/).

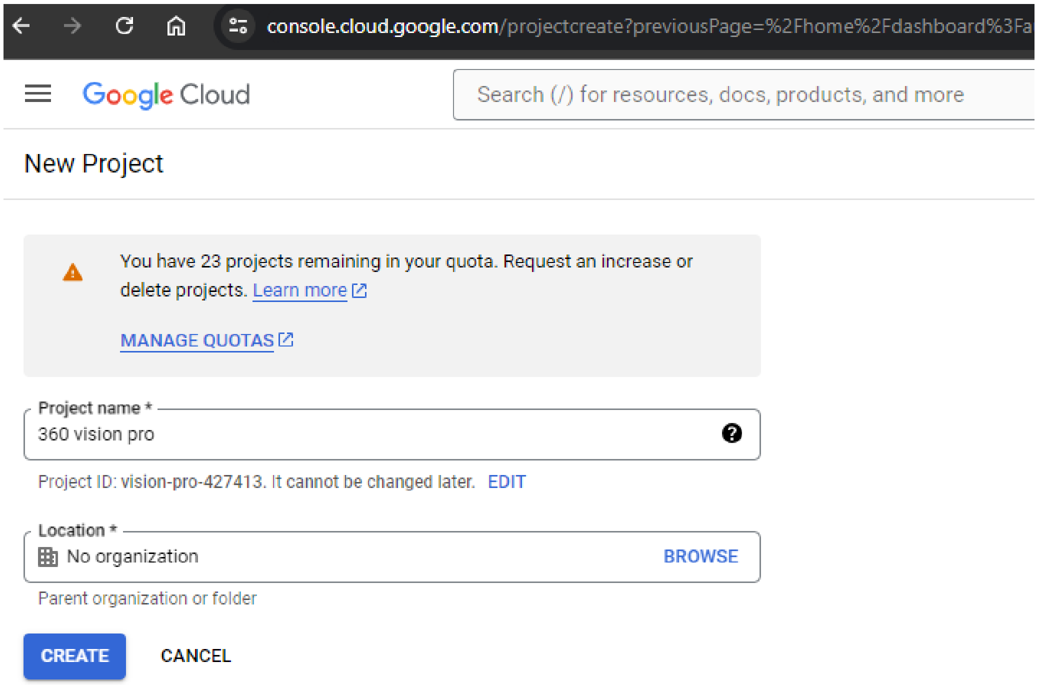


**Get $300** in free credits and free usage of 20+ products

The new way to cloud starts here

Build apps fast, leverage generative AI, and analyze data in seconds—all with Google-grade security.

2. **Create a New Project:**
   - Go to the GCP Console, select the project dropdown and click on “New Project”.
   - Enter a project name and create the project.



console.cloud.google.com/projectcreate?previousPage=%2Fhome%2Fdashboard%3Fa

New Project

You have 23 projects remaining in your quota. Request an increase or delete projects. Learn more

Project name *: 360 vision pro

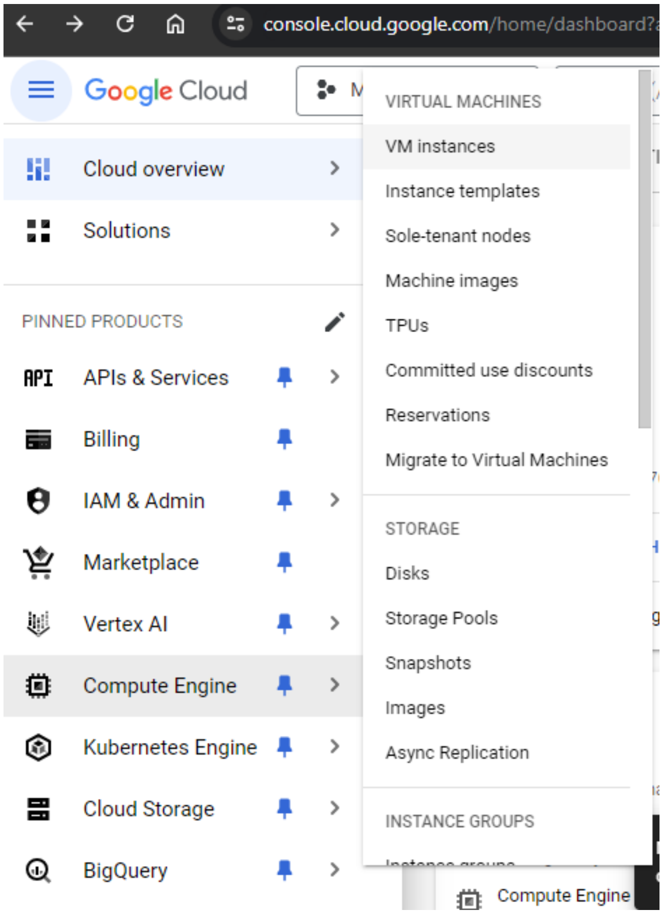
Project ID: vision-pro-427413. It cannot be changed later. EDIT

Location *: No organization

Parent organization or folder

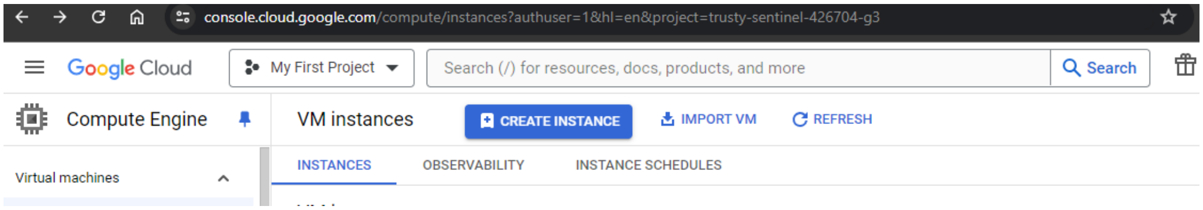
## Step 2: Set Up a Virtual Machine (VM)

1. **Navigate to Compute Engine:**
   - In the GCP Console, navigate to Compute Engine > VM instances.

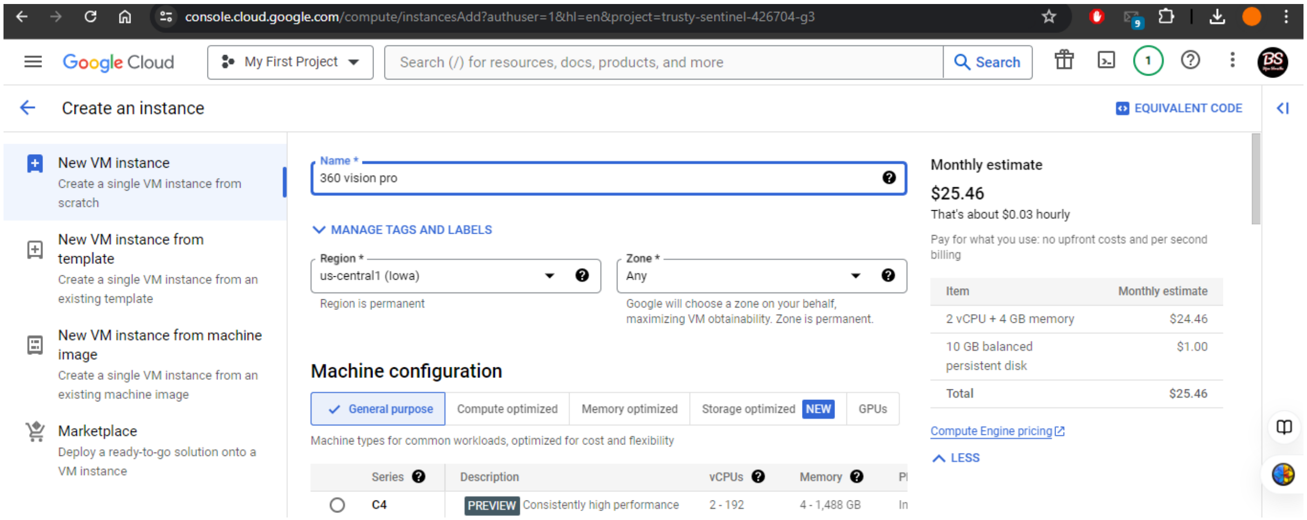


console.cloud.google.com/home/dashboard?

- Create a New VM Instance:
- Click on “Create Instance”.



2. **Configure the instance:**
   - Name: vision 360
   - Region and Zone: Select your preferred region and zone.
   - Machine type: Choose a machine type according to your requirements (e.g., e2-medium).
   - Boot disk: Click on “Change” to select an OS. Choose Debian or Ubuntu.
   - Allow HTTP and HTTPS traffic under "Firewall".
   - Click on “Create”.


console.cloud.google.com/compute/instancesAdd?authuser=1&hl=en&project=trusty-sentinel-426704-g3
Google Cloud
My First Project
Search (/) for resources, docs, products, and more
Search
Create an instance
EQUIVALENT CODE
New VM instance
Create a single VM instance from scratch
New VM instance from template
Create a single VM instance from an existing template
New VM instance from machine image
Create a single VM instance from an existing machine image
Marketplace
Deploy a ready-to-go solution onto a VM instance
MANAGE TAGS AND LABELS
Machine configuration
General purpose
Compute optimized
Memory optimized
Storage optimized
NEW
GPUs
LESS

Name * 360 vision pro

Region * us-central1 (Iowa)

Region is permanent

Zone * Any

Google will choose a zone on your behalf, maximizing VM obtainability. Zone is permanent.

Machine types for common workloads, optimized for cost and flexibility

| Series | Description | vCPUs | Memory | Pl |
|---|---|---|---|---|
| ○ C4 | PREVIEW Consistently high performance | 2 - 192 | 4 - 1,488 GB | In |

Monthly estimate

$25.46

That's about $0.03 hourly

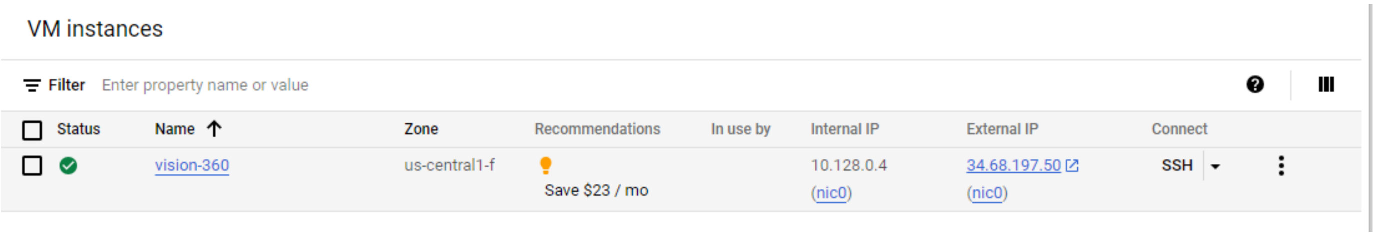
Pay for what you use: no upfront costs and per second billing

| Item | Monthly estimate |
|---|---|
| 2 vCPU + 4 GB memory | $24.46 |
| 10 GB balanced persistent disk | $1.00 |
| Total | $25.46 |

Compute Engine pricing

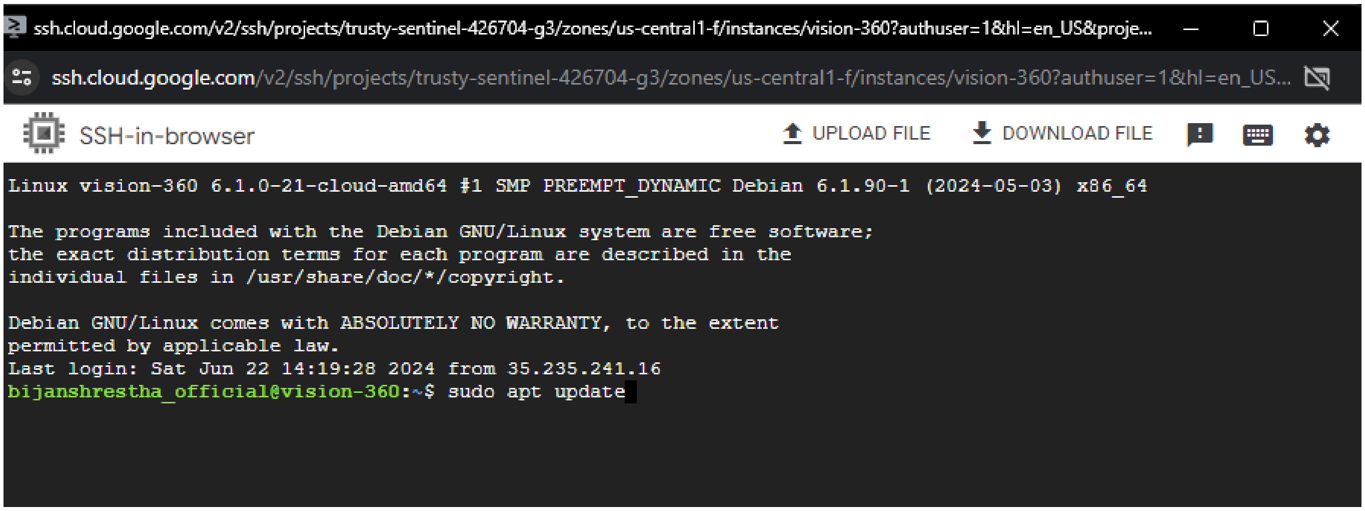
## Step 3: Install Necessary Software on the VM

1. **SSH into the VM:**
   - Click on the SSH button next to your VM instance to open a terminal.



| □ | Status | Name ↑ | Zone | Recommendations | In use by | Internal IP | External IP | Connect |
|---|---|---|---|---|---|---|---|---|
| □ | ✓ | vision-360 | us-central1-f | Save $23 / mo | | 10.128.0.4 (nic0) | 34.68.197.50 (nic0) | SSH |

- Update the Package List:
  ***sudo apt update***


ssh.cloud.google.com/v2/ssh/projects/trusty-sentinel-426704-g3/zones/us-central1-f/instances/vision-360?authuser=1&hl=en_US&proje...
ssh.cloud.google.com/v2/ssh/projects/trusty-sentinel-426704-g3/zones/us-central1-f/instances/vision-360?authuser=1&hl=en_US...
SSH-in-browser
UPLOAD FILE
DOWNLOAD FILE

```
Linux vision-360 6.1.0-21-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.90-1 (2024-05-03) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Jun 22 14:19:28 2024 from 35.235.241.16
bijanshrestha_official@vision-360:~$ sudo apt update
```

2. **Install Apache, PHP, and Required PHP Extensions:**
   - ***sudo apt install apache2***
   - ***sudo apt install php php-cli libapache2-mod-php php-mysql php-xml php-mbstring php-zip unzip***

3. **Install Composer:**
   - ***php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"***
   - ***php composer-setup.php --install-dir=/usr/local/bin --filename=composer***
   - ***rm composer-setup.php***

**Step 4: Set Up MariaDB**

1. Install MariaDB Server:
   - ***sudo apt install mariadb-server***

2. Secure MariaDB Installation:
   - ***sudo mysql_secure_installation***

3. Log into MariaDB and Create Database:
   - ***sudo mysql -u root -p***

4. sql
    - ***CREATE DATABASE 360_video_project;***
    - ***CREATE USER 'laravel_user'@'localhost' IDENTIFIED BY 'your_password';***
    - ***GRANT ALL PRIVILEGES ON laravel_db.* TO 'laravel_user'@'localhost';***
    - ***FLUSH PRIVILEGES;***
    - ***EXIT;***

## Step 5: Deploy Laravel Application

1. Navigate to Apache Root Directory:
    - ***cd /var/www/html***

2. Clone Your Laravel Application from GitHub:
    - **sudo git** ***clone https://github.com/bijanstha7/360-Vision-Pro.git***
    - **sudo mv your-repository laravel-app**
    - **cd laravel-app**

3. Install Laravel Dependencies:
    - ***sudo composer install***

4. Set Up Environment File:
    - ***sudo cp .env.example .env***
    - ***sudo nano .env***

5. Update the .env file with your database credentials:
    - makefile
    - Copy code
    - ***DB_CONNECTION=mysql***
    - ***DB_HOST=127.0.0.1***
    - ***DB_PORT=3306***
    - ***DB_DATABASE=laravel_db***
    - ***DB_USERNAME=laravel_user***
    - ***DB_PASSWORD=your_password***

6. Generate Application Key:
   - ***sudo php artisan key:generate***

7. Run Database Migrations:
   - ***sudo php artisan migrate***

## Step 6: Configure Apache for Laravel

1. Create a New Virtual Host Configuration File:
   - ***sudo nano /etc/apache2/sites-available/laravel-app.conf***

2. Add the Following Configuration:

apache

Copy code

```
<VirtualHost *:80>
   ServerAdmin webmaster@localhost
   DocumentRoot /var/www/html/laravel-app/public

   <Directory /var/www/html/laravel-app>
      Options Indexes FollowSymLinks
      AllowOverride All
      Require all granted
   </Directory>

   <Directory /var/www/html/laravel-app/public>
      Options Indexes FollowSymLinks
      AllowOverride All
      Require all granted
   </Directory>

   ErrorLog ${APACHE_LOG_DIR}/error.log
   CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

3. Enable the Site and Rewrite Module:
    - ***sudo a2ensite laravel-app.conf***
    - ***sudo a2enmod rewrite***
    - ***sudo systemctl restart apache2***

## Step 7: Test Your Laravel Application

1. Access Your Application:
    - Open a web browser and navigate to the **external IP** of your VM instance (found in the VM instance details).

2. Verify Everything is Working:
    - Ensure the Platform homepage is displayed.