



UCSC



Manual de Instalación

Plataforma de Juez Virtual TutorBot+

VERSIÓN 1.0

Importante

La plataforma web fue probada en un servidor con sistema operativo Ubuntu Server 22.04, debido a que los desarrolladores de Judge0 recomiendan utilizar sistemas operativos Linux para la instalación, por lo que se recomienda utilizar Ubuntu 20.04 o 22.04 para su instalación.

Instalación

0. Judge0

Debes tener una instancia del juez virtual Judge0. Judge0 ofrece servicios en la nube para utilizarlo con RapidAPI, puedes ver los planes en el siguiente sitio web: <https://judge0.com/#pricing>

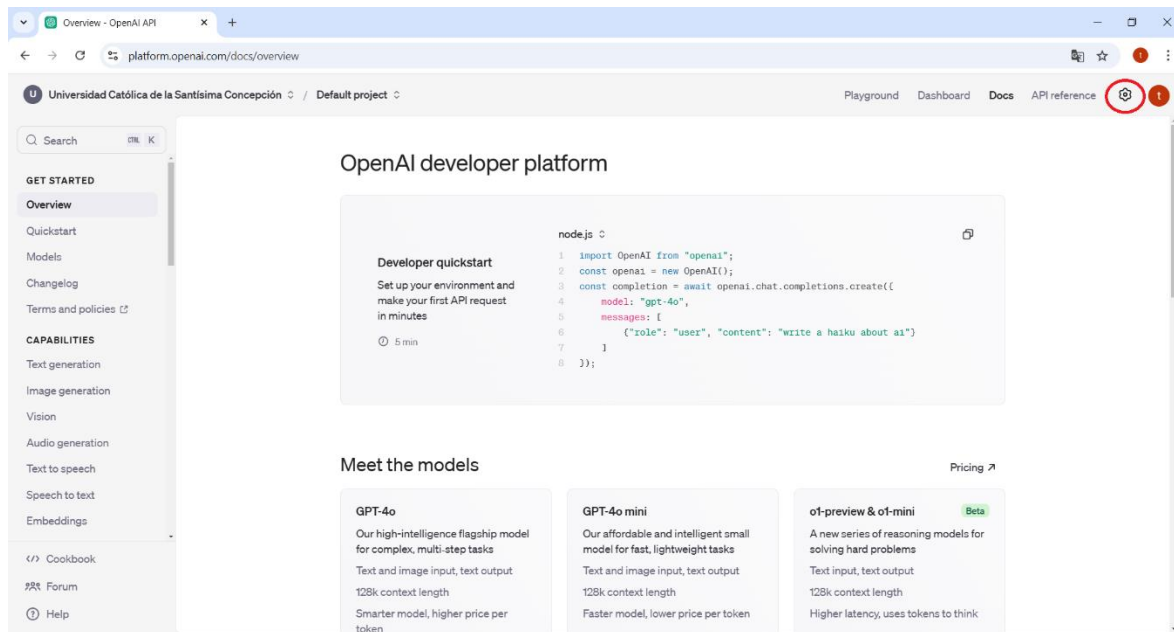
Para instalar una instancia de judge0 en un servidor local, se recomienda ver las instrucciones de instalación desde el repositorio de Judge0 para obtener la última versión disponible.

<https://github.com/judge0/judge0/blob/master/CHANGELOG.md#deployment-procedure>

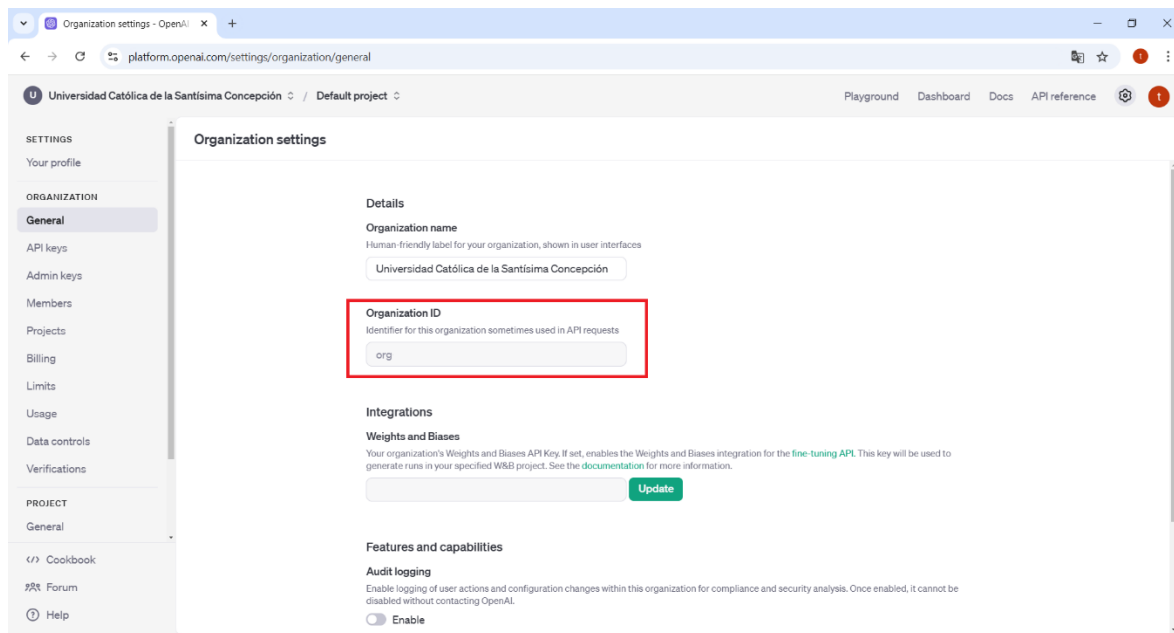
1. OpenAI GPT API

Debes tener una cuenta en OpenAI, si no has creado una, haz clic en el siguiente enlace: <https://platform.openai.com/signup>.

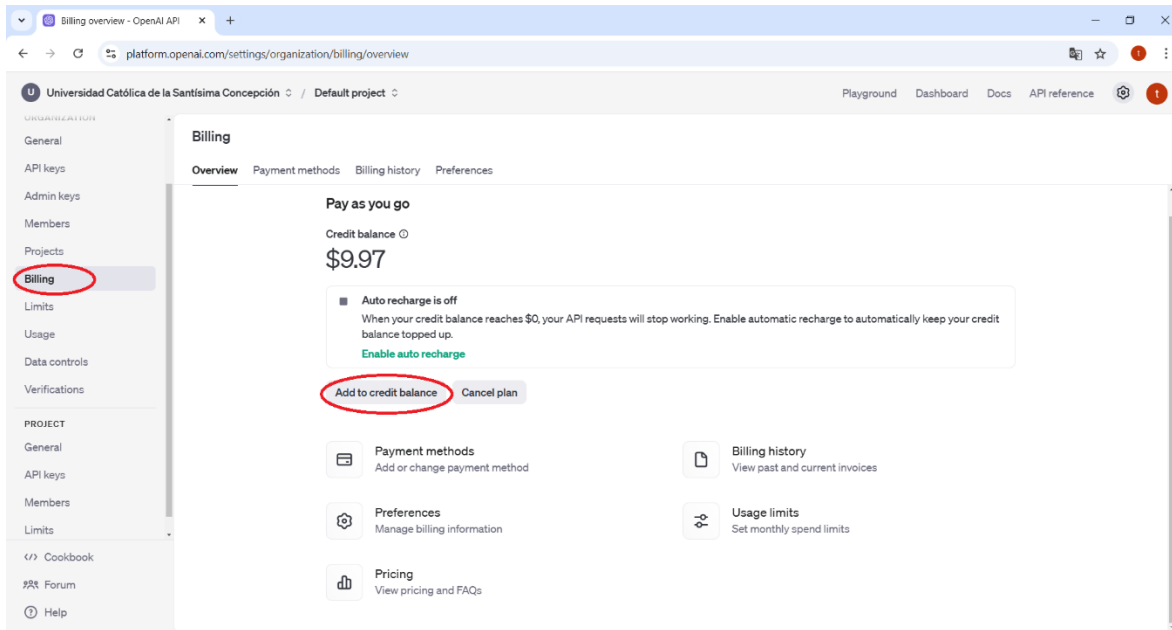
Una vez creada la cuenta e iniciado sesión, debes acceder al sitio <https://platform.openai.com> para acceder a la API de GPT, verás una página de la documentación de GPT y debes presionar el icono de engranaje.



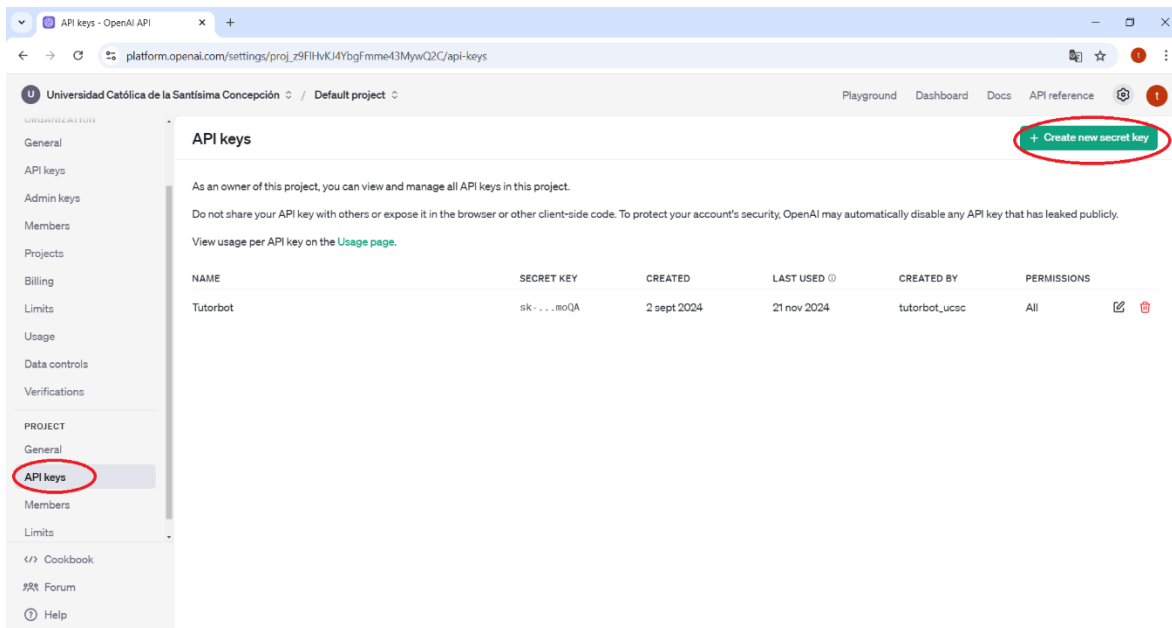
Serás redirigido a las configuraciones de tu cuenta, verás el campo “Organization ID” a primera vista, debes guardar este código generado que será utilizado en la configuración de la plataforma web.



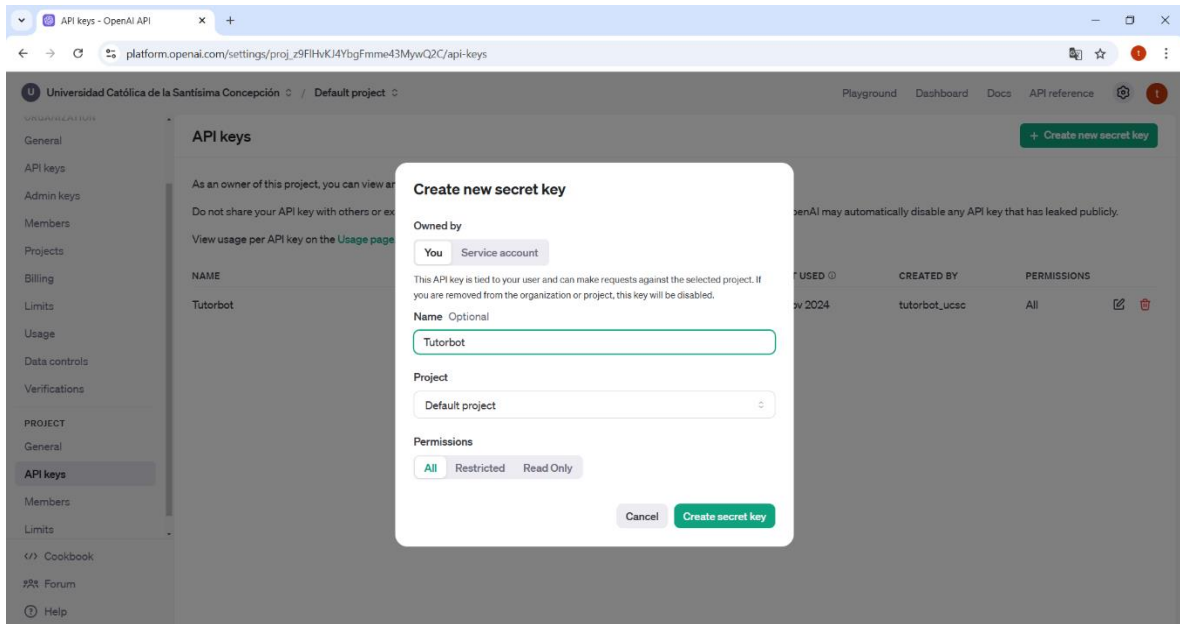
Debes tener créditos en tu cuenta de OpenAI para tener acceso a la api de GPT, debes dirigirte a “Billing”, presionándolo en la barra lateral izquierda y serás redireccionado a la página de “Billing”, aquí podrá añadir créditos al balance haciendo clic en “Add to Credit Balance”.



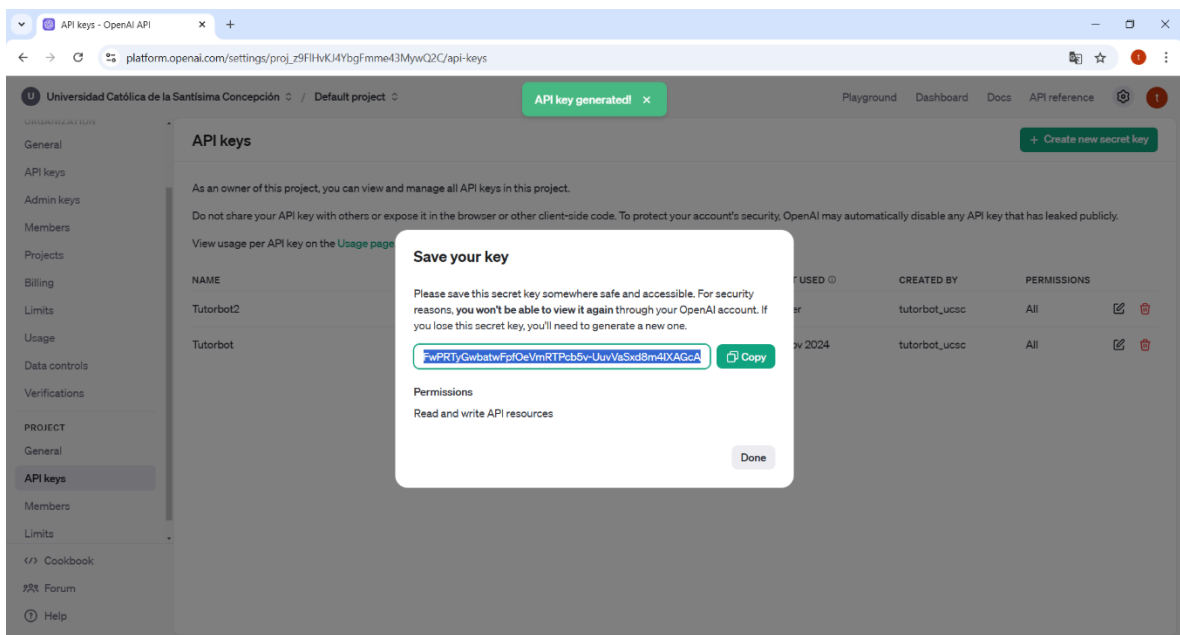
Una vez ingresados los créditos al balance, puedes ir a la sección “Api Keys”, al hacer clic, será redirigido a la página de Api Keys, debes crear una nueva secret key para utilizarlo en la plataforma web.



Al hacer clic en el botón “Create new secret key”, se abrirá una ventana en el que tienes que ingresar el nombre, el proyecto (por defecto es “Default Project”, pero puedes crear un proyecto único si lo deseas) y finalmente los permisos.



Al hacer clic en “Create secret key”, se creará una clave secreta para acceder a la API de GPT, debes guardar esta clave en algún archivo de texto, esta clave será utilizada en la configuración de la plataforma web.



Importante: asegúrate de no filtrar tu clave secreta, ya que podrán tener acceso a tus configuraciones de tu cuenta de GPT y tener acceso a la API, consumiendo tus créditos.

2. Clonar repositorio de Tutorbot+

Instalar el software Git utilizando los siguientes comandos:

```
$ sudo apt update
$ sudo apt install git
```

Una vez instalado Git, puede confirmar si se ha instalado correctamente utilizando el siguiente comando:

```
$ git --version
```

Al utilizar este comando, te entrega la versión de Git instalada. Una vez instalado Git y verificada su instalación, debes clonar el repositorio de Tutorbot+ desde Github utilizando el siguiente comando:

```
$ git clone https://github.com/jmacias-ctrl/tutorbot-laravel.git
```

Se descargará todos los archivos correspondientes a la plataforma web.

3. Instalar PHP y Composer

Para instalar PHP, debes ingresar los siguientes comandos:

```
$ sudo apt update
$ sudo add-apt-repository ppa:ondrej/php
$ sudo apt install php8.2-cli php8.2-dev php8.2-pgsql php8.2-sqlite3 php8.2-gd php8.2-imagick \
  php8.2-curl php8.2-imap php8.2-mysql php8.2-mbstring php8.2-xml php8.2-zip \
  php8.2-bcmath php8.2-soap php8.2-intl php8.2-readline php8.2-ldap php8.2-msgpack \
  php8.2-igbinary php8.2-redis php8.2-memcache php8.2-pcov php8.2-xdebug
$ sudo apt install php-cli unzip
```

Una vez instalado PHP, se debe instalar Composer, ingresa los siguientes comandos para instalar Composer:

```
$ php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
$ php -r "if (hash_file('sha384', 'composer-setup.php') === 'dac665fdc30fdd8ec78b38b9800061b4150413ff2e3b6f88543c636f7cd84f6db9189d43a81e5503cda447da73c7e5b6') { echo 'Installer verified'; } else { echo 'Installer corrupt'; unlink('composer-setup.php'); } echo PHP_EOL;"
$ php composer-setup.php
$ php -r "unlink('composer-setup.php');"
```

Una vez ejecutado los comandos, se descargara el último composer.phar en el directorio actual, debes mover este archivo al directorio PATH utilizando el siguiente comando:

```
$ sudo mv composer.phar /usr/local/bin/composer
```

Para más información sobre Composer, visite su sitio web <https://getcomposer.org/download/>

4. Configurar Base de Datos (MySQL)

Se recomienda utilizar MySQL para el sistema de base de datos, ya que la plataforma web solo ha sido probada con MySQL. A continuación, se dará las indicaciones sobre como configurar MySQL para la plataforma web.

Si no tienes instalado MySQL, puedes hacerlo utilizando los siguientes comandos:

```
$ sudo apt update  
$ sudo apt install mysql-server
```

Luego debes ingresar al entorno de mysql utilizando el comando:

```
$ mysql
```

Debes crear un usuario para la plataforma web utilizando el siguiente comando, donde debes cambiar <asignar contraseña>:

```
> CREATE USER 'tutorbotuser'@'localhost' IDENTIFIED BY '<asignar contraseña>';
```

Luego, debes crear una base de datos utilizando el siguiente comando:

```
> CREATE DATABASE tutorbot;  
CREATE USER 'tutorbotuser'@'localhost' IDENTIFIED BY '<asignar contraseña>';  
GRANT ALL PRIVILEGES ON tutorbot . * TO 'tutorbotuser'@'localhost';
```

Una vez creada la base de datos, puedes salir del entorno de MySQL escribiendo el comando "exit".

5. Instalar Node.js

Para instalar Node.js, debes tener instalado cUrl, si no ha instalado cUrl ingrese los siguientes comandos:

```
$ sudo apt update  
$ sudo apt install curl
```

Se instalará Node.js utilizando NodeSource, debes ejecutar el siguiente comando con cURL (debes reemplazar Y con la última versión disponible de node.js, debes verificar en el sitio web oficial cual es la última versión <https://nodejs.org/en/>):

```
$ curl -fsSL https://deb.nodesource.com/setup_Y.x | sudo -E bash -
```

Una vez obtenido el repositorio de NodeSource, instalas Node.js ejecutando el siguiente comando:

```
$ sudo apt-get install nodejs
```

Para verificar si se ha instalado correctamente, ejecuta los siguientes comandos, debería entregarte la última versión de node y npm respectivamente:

```
$ node -v  
$ npm -v
```

6. Instalación de la plataforma

Ir al directorio de la plataforma web clonada previamente:

```
$ cd tutorbot-laravel/tutorbot
```

Primero, se debe instalar los paquetes y dependencias de Laravel utilizando el siguiente comando:

```
$ composer install
```

Una vez ejecutado el comando, se debe establecer la configuración del entorno, debes copiar el archivo .env.example utilizando el siguiente comando:

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

Se creará un archivo llamado .env que es la configuración del entorno de la plataforma web, se debe editar el archivo, utilizando un editor como Nano, Vim u otro y modificar lo siguiente:

1. Si la plataforma va a estar en producción, debes modificar la variable `APP_ENV` de "local" a "production", y `APP_DEBUG` de "true" a "false", si no haces esto tienes el riesgo de exponer información sensible.
2. Si la plataforma va a estar en producción, `APP_URL` debe cambiarse a la dirección local. Para verificar cual es la dirección puedes utilizar el comando "ip addr" y ver la interfaz de ethernet o wifi.

3. Se debe ingresar las API Keys de Judge0: Si estás utilizando una instancia de Judge0 en la plataforma web RapidApi, solamente debes ingresar la clave de la API en la variable `JUDGE0_API_KEY_RAPID_API`. Si estás utilizando una instancia de Judge0 en un servidor local, debes ingresar la clave de la API en la variable `JUDGE0_API_KEY_PROD`, luego las clave de “autorización” y “autenticación” en las variables `JUDGE0_AUTHORIZE_KEY_PROD` y `JUDGE0_AUTHENTICATION_KEY_PROD` respectivamente, finalmente el url o dirección IP donde se encuentra la instancia de Judge0. En caso de estar en el mismo servidor donde se va a ubicar la plataforma web, utilizar 127.0.0.1:2358. Por lo general el puerto para acceder a la instancia de Judge0 es 2359, pero este puede ser cambiado según se estime conveniente durante la instalación de Judge0.
4. Se debe modificar las variables de entorno de la base de datos. Se debe reemplazar `DB_USERNAME` y `DB_PASSWORD` por el nombre de usuario y contraseña por lo que has establecido durante la instalación de MySQL u otro:

```
DB_CONNECTION=mysql #Cambiar si utilizas otro software de base de datos
DB_HOST=127.0.0.1
DB_PORT=3307
DB_DATABASE=tutorbot
DB_USERNAME=<Nombre de Usuario>
DB_PASSWORD=<Contraseña>
```

5. Configurar las variables de entorno de Mailer para enviar correos electrónicos, si se utiliza Gmail, Outlook u otro servidor SMTP, investigar la documentación de los servidores respectivos.
6. Por último, para configurar la variable de entorno de OpenAI para la generación de retroalimentación en la plataforma web, debes ingresar la clave de la API y la clave de la organización en las variables `OPENAI_API_KEY` y `OPENAI_ORGANIZATION`, estas son las claves que fueron obtenidas en la sección 1.

Guardar los cambios en el archivo `.env`.

Luego ejecuta los siguientes comandos, que generan una key única para la aplicación y el acceso al “storage” para el almacenamiento de archivos:

```
$ php artisan key:generate
$ php artisan storage:link
```

Luego debes instalar las dependencias de JavaScript, utilizando npm:

```
$ npm install
```

Se instalarán todas las librerías y dependencias de JavaScript que serán utilizados en la plataforma web, debes instalar Laravel Mix utilizando el siguiente comando:

```
$ npm install laravel-mix@latest --save-dev
```

Luego debes ejecutar el siguiente comando para generar los archivos de JavaScript y estilos SCSS:

```
$ npm run prod
```

Luego debes ejecutar el comando migration para crear las tablas en la base de datos y realizar seeding utilizando el siguiente comando:

```
$ php artisan migrate --seed
```

Para utilizar la localización en español de Laravel, debes ejecutar los siguientes comandos:

```
$ php artisan lang:publish  
$ php artisan lang:update
```

Validaciones y excepciones saldrán en español una vez ejecutado los comandos. Luego debes ejecutar los siguientes comandos para generar archivos caches para las vistas, rutas, configuraciones y eventos, esto es para uso en producción de la plataforma web:

```
$ php artisan config:cache  
$ php artisan event:cache  
$ php artisan route:cache  
$ php artisan view:cache
```

Una vez ingresado los comandos, has finalizado la instalación de la plataforma web.

7. Configurar Queues de Laravel

Para el envío de correos, se debe utilizar queues de laravel que ejecuta tareas en segundo plano. Se necesita tener instalado Supervisorctl:

```
$ sudo apt-get install supervisor
```

Por lo general, las configuraciones de supervisor se encuentran en el directorio /etc/supervisor/conf.d, en este directorio debes crear un nuevo archivo de texto y llamarlo laravel-worker.conf.d y debe contener lo siguiente, reemplazando <Directorio-al-proyecto> con el directorio a la plataforma web, <Directorio-donde-guardar-logs> con algún directorio

donde guardar los logs del evento y <ingresar-usuario> el nombre de usuario a utilizar el programa:

```
[program:laravel-worker]
process_name=%(program_name)s_%(process_num)02d
command=php <Directorio-al-proyecto>/artisan queue:work sqs --sleep=3 --tries=3 --
max-time=3600
autostart=true
autorestart=true
stopasgroup=true
killasgroup=true
user=<ingresar-usuario>
numprocs=8
redirect_stderr=true
stdout_logfile=<Directorio-donde-guardar-logs>
stopwaitsecs=3600
```

Una vez guardado el archivo con el contenido, debes ejecutar supervisor utilizando los siguientes comandos:

```
$ sudo supervisorctl reread
$ sudo supervisorctl update
$ sudo supervisorctl start "laravel-worker:*"
```

Verifica si está funcionando correctamente revisando el archivo log file.

Para ambientes de prueba, se puede utilizar el comando php artisan queue:listen para la ejecución de tareas, no es recomendable utilizarlo en producción.

8. Cron Jobs para ejecutar tareas programadas

La plataforma web requiere ejecutar algunas tareas en segundo plano cada cierto periodo de tiempo, utilizando los comandos de Task Scheduling de Laravel. Para esto se requiere tener instalado cronjobs

```
$ sudo apt-get install cron
```

Una vez instalado cron jobs, ejecutar el servicio de cron con el siguiente comando.

```
$ systemctl start cron
$ systemctl enable cron
```

Luego debes configurar un cron job, utilizando el siguiente comando (ver que usuario utilizar para la ejecución de tareas programadas utilizando -u <nombre-usuario> antes del -e):

```
$ crontab -e
```

Se abrirá un editor de texto para ingresar cron jobs, debes ingresar los siguiente:

```
* * * * * cd <Directorio-al-proyecto>/tutorbot-laravel/tutorbot && php artisan  
schedule:run >> /dev/null 2>&1
```

Guardar cambios y luego reiniciar el servicio de cron:

```
$ systemctl restart cron
```

En un ambiente de pruebas, se puede utilizar el comando `php artisan schedule:work` para ejecutar tareas programadas, este comando no debe ser utilizado en producción, solamente para hacer pruebas.

9. Configuración de nginx

Debes instalar nginx utilizando el siguiente comando:

```
$ apt install nginx
```

Debes crear un archivo `.conf` que debe contener el siguiente contenido y debe estar ubicado en el directorio `/etc/nginx/conf.d`:

```
server {  
    listen 80;  
    listen [::]:80;  
    root <directorio al proyecto>/tutorbot-laravel/tutorbot/public;  
  
    add_header X-Frame-Options "SAMEORIGIN";  
    add_header X-Content-Type-Options "nosniff";  
  
    index index.php;  
  
    charset utf-8;  
  
    location / {  
        try_files $uri $uri/ /index.php?$query_string;  
    }  
  
    location = /favicon.ico { access_log off; log_not_found off; }  
    location = /robots.txt  { access_log off; log_not_found off; }  
  
    error_page 404 /index.php;  
  
    location ~ \.php$ {  
        fastcgi_pass unix:/var/run/php/php8.2-fpm.sock;  
        fastcgi_param SCRIPT_FILENAME $realpath_root$fastcgi_script_name;  
        include fastcgi_params;  
    }  
  
    location ~ /\.(!well-known).* {  
        deny all;  
    }  
}
```



Puedes modificar el archivo como estimes conveniente. Por ejemplo, asignar un dominio al servidor, esta es una configuración base para acceder a la plataforma web. Guardar cambios en el archivo y luego ejecutar el comando:

```
$ sudo systemctl reload nginx
```

Podrás acceder a la plataforma web utilizando la dirección IP local o utilizando un dominio asignado en el servidor, debes ver la siguiente página en el navegador:



Iniciar Sesión

Ingresa su correo y contraseña

☐ Recuerdame

Iniciar Sesión

¿Olvidaste tu contraseña? [Recupera tu contraseña aquí](#)

