

Back-End:Strapi

GUIA SOBRE EL USO DEL BAKC-END DEL PROYECTO

PRIMER EQUIPO DE DESARROLLO

Control de versiones

Modificado por:	Motivos	Fecha	Versión
Christopher Arturo Rajon Polanco	Creación del documento	12/12/23	0.1

Tabla de contenido

Introducción a Strapi: Potenciando Proyectos con un CMS de Código Abierto	3
¿Qué es Strapi?	3
Características Principales:.....	3
Casos de Uso:	3
Instalación de Strapi.....	5
Requisitos Previos:	5
Pasos de Instalación:	5
Back End del juego Conquista Matemática.....	10
Instalación del back end.....	10
Credenciales de acceso.	13
Modificar información.....	13
Content-Type Builder:	13
Content Manager:	15
Despliegue de Strapi.	22
Cambios en la base de datos.....	22

Introducción a Strapi: Potenciando Proyectos con un CMS de Código Abierto

En el dinámico mundo del desarrollo web, la gestión eficiente de contenidos es esencial para el éxito de cualquier proyecto. Strapi, un Sistema de Gestión de Contenidos (CMS) de código abierto, ha surgido como una poderosa herramienta que permite a los desarrolladores crear, administrar y distribuir contenido de manera ágil y flexible.

¿Qué es Strapi?

Strapi es un CMS headless que se ha destacado por su enfoque modular y su arquitectura flexible. A diferencia de los sistemas tradicionales, Strapi opera como una interfaz de programación de aplicaciones (API) independiente, permitiendo a los desarrolladores gestionar el contenido de sus aplicaciones web o móviles de manera eficiente.

Características Principales:

1. **API First:** Strapi adopta un enfoque "API first", lo que significa que la API es el núcleo central y la interfaz de usuario es solo una capa adicional. Esto proporciona una mayor flexibilidad y permite a los desarrolladores consumir los datos de manera más versátil.
2. **Código Abierto:** Strapi es completamente de código abierto, lo que brinda a los desarrolladores la libertad de personalizar y extender la plataforma según sus necesidades específicas.
3. **Flexibilidad y Escalabilidad:** Gracias a su arquitectura modular, Strapi permite a los equipos de desarrollo seleccionar y utilizar únicamente los módulos necesarios para su aplicación, evitando la sobrecarga y mejorando la eficiencia.
4. **Soporte para Diferentes Bases de Datos:** Strapi es compatible con una variedad de bases de datos, incluyendo PostgreSQL, MongoDB, MySQL y SQLite, ofreciendo a los desarrolladores la libertad de elegir la solución que mejor se adapte a sus requerimientos.
5. **Interfaz de Administración Intuitiva:** Strapi incluye una interfaz de administración fácil de usar que permite a los no desarrolladores gestionar el contenido sin necesidad de conocimientos técnicos avanzados.

Casos de Uso:

1. Desarrollo de Aplicaciones Web y Móviles: Strapi es ideal para la creación rápida de aplicaciones web y móviles, permitiendo a los desarrolladores concentrarse en la lógica de la aplicación mientras delegan la gestión de contenidos a la plataforma.

2. Websites Empresariales: Empresas de todos los tamaños pueden beneficiarse de Strapi al gestionar eficientemente grandes cantidades de contenido y mantener una presencia en línea dinámica.

3. Proyectos de Comercio Electrónico: Strapi facilita la gestión de catálogos de productos, contenido promocional y otros aspectos clave de las plataformas de comercio electrónico.

En resumen, Strapi emerge como una herramienta crucial para los desarrolladores que buscan una solución de gestión de contenidos flexible, escalable y personalizable. Con su enfoque API-first, Strapi empodera a los equipos de desarrollo para construir aplicaciones web y móviles modernas y dinámicas, brindando una experiencia de usuario excepcional y acelerando el proceso de desarrollo. Con su código abierto y comunidad activa, Strapi se presenta como una opción sólida para proyectos que buscan aprovechar al máximo las capacidades de un CMS moderno.

Instalación de Strapi

A continuación, se presenta una guía paso a paso para la instalación de Strapi en un entorno local. Puedes visitar la pagina oficial para más información: <https://strapi.io/>

Requisitos Previos:

Antes de comenzar la instalación, asegúrate de tener los siguientes requisitos previos en tu sistema:

1. Node.js y npm: Strapi se basa en Node.js, así que asegúrate de tenerlo instalado. Puedes descargar Node.js desde nodejs.org.
2. Base de Datos: Selecciona la base de datos que deseas utilizar con Strapi (MongoDB, PostgreSQL, MySQL o SQLite) y asegúrate de tenerla instalada y configurada.

Pasos de Instalación:

1. Instalación de Strapi:

Utiliza npm para instalar Strapi de forma global ejecutando el siguiente comando en tu terminal:

```
...  
npm install -g strapi@latest  
...
```

2. Creación de un Nuevo Proyecto:

Crea un nuevo proyecto Strapi ejecutando el siguiente comando y siguiendo las instrucciones en pantalla:

```
...  
strapi new nombre_del_proyecto  
...
```

Este comando generará la estructura básica de tu proyecto Strapi. También podemos usar el siguiente código

```
...  
npx create-strapi-app@latest my-project  
...
```

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.19045.3930]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Christopher\Desktop\Nueva carpeta>npx create-strapi-app@latest my-project_
```

3. Configuración de la Base de Datos:

Durante la creación del proyecto, Strapi te solicitará que elijas una base de datos y proporciones la información necesaria. Asegúrate de configurar correctamente la conexión a tu base de datos. Si quieres manejar las cosas de manera local puedes usar XAMPP para crear una base de datos local, ó puedes usar un servicio de base de datos en línea, siempre es importante crear primero una base de datos y después introducir la información que se pide en la terminal.

```
Microsoft Windows [Versión 10.0.19045.3930]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Christopher\Desktop\Nueva carpeta>npx create-strapi-app@latest my-project
Need to install the following packages:
create-strapi-app@4.17.1
Ok to proceed? (y) y
? Choose your installation type
  Quickstart (recommended)
> Custom (manual settings)
```

```
Microsoft Windows [Versión 10.0.19045.3930]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Christopher\Desktop\Nueva carpeta>npx create-strapi-app@latest my-project
Need to install the following packages:
create-strapi-app@4.17.1
Ok to proceed? (y) y
? Choose your installation type Custom (manual settings)
? Choose your preferred language (Use arrow keys)
? Choose your preferred language JavaScript
? Choose your default database client
? Choose your default database client mysql
? Database name: (my-project)
? Database name: my-project
? Host: (127.0.0.1)
? Host: 127.0.0.1
? Port: (3306)
? Port: 3306
? Username: user
? Username: user
? Password: ****
? Password: ****
? Enable SSL connection: (y/N) y_
```

4. Iniciar el Servidor de Desarrollo:

Cambia al directorio de tu proyecto y ejecuta el siguiente comando para iniciar el servidor de desarrollo:

```
'''
cd nombre_del_proyecto
strapi develop
ó
npm run develop
'''
```

Strapi se ejecutará en `http://localhost:1337` por defecto.


```
C:\Users\Christopher\Desktop\API-Juego\api-REST>npm run develop

> api-rest@0.1.0 develop
> strapi develop

- Building build context
\ Building build context[INFO] Including the following ENV variables as part of the JS bundle:
  - ADMIN_PATH
  - STRAPI_ADMIN_BACKEND_URL
  - STRAPI_TELEMETRY_DISABLED
✓ Building build context (391.5879000425339ms)
✓ Creating admin (49691.17480003834ms)
✓ Generating types (1748.1360000371933ms)

Project information

Time                Wed Jan 17 2024 20:45:32 GMT-0600 (hora estándar de México)
Launched in         6646 ms
Environment          development
Process PID          11796
Version              4.15.4 (node v20.10.0)
Edition              Community
Database              mysql

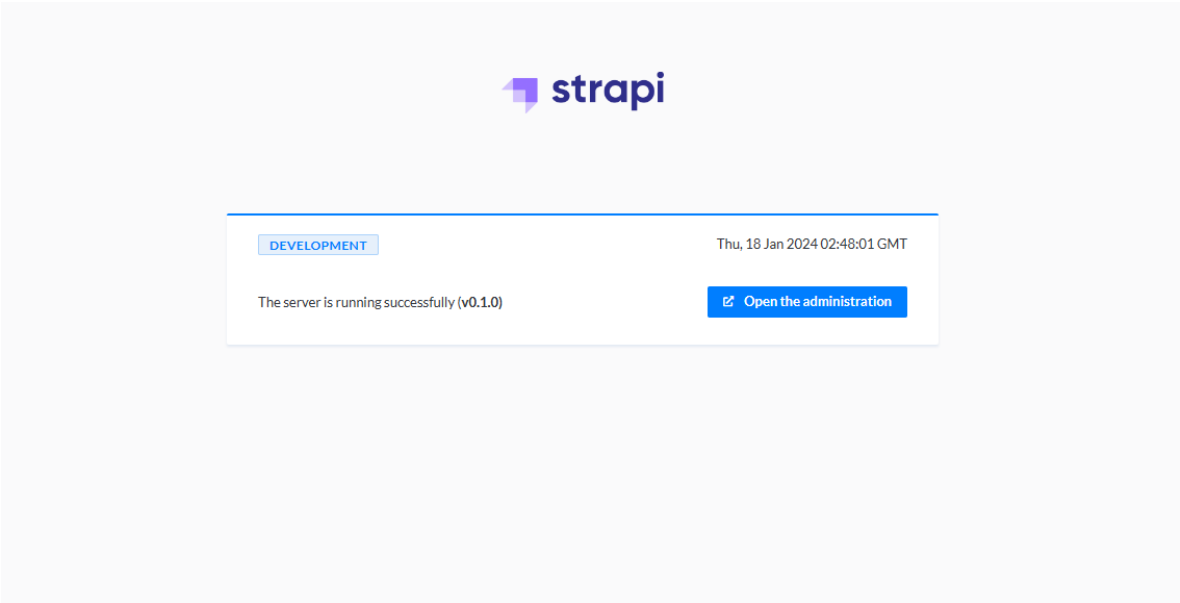
Actions available

Welcome back!
To manage your project 📁, go to the administration panel at:
http://localhost:1337/admin


To access the server 🌐, go to:
http://localhost:1337
```

5. Configuración Inicial a través del Panel de Administración:

Accede al panel de administración de Strapi en tu navegador ingresando `http://localhost:1337/admin`. Completa la configuración inicial, creando un nuevo usuario administrador y configurando cualquier ajuste adicional que tu proyecto pueda necesitar.



English ▾



Welcome to Strapi!


Log in to your Strapi account

Email*

Password*

☐ Remember me

Login

 **Strapi Dashboard**
Workplace

Content Manager

PLUGINS

Content-Type Builder

Media Library

Strapi Cloud

GENERAL

Plugins

Marketplace

Settings 1

JR

Juego RPG

<


Welcome 🙌

We hope you are making progress on your project! Feel free to read the latest news about Strapi. We are giving our best to improve the product based on your feedback.

[See more on the blog](#)


3 steps to get started

1


 Build the content structure

Go to the Content type Builder →

2

 What would you like to share with the world?


3


 See content in action


Join the community


Discuss with team members, contributors and developers on different channels.


[See our roadmap](#)


 Github


 Discord


 Reddit

 Twitter

 Forum

 Blog

 We are hiring!

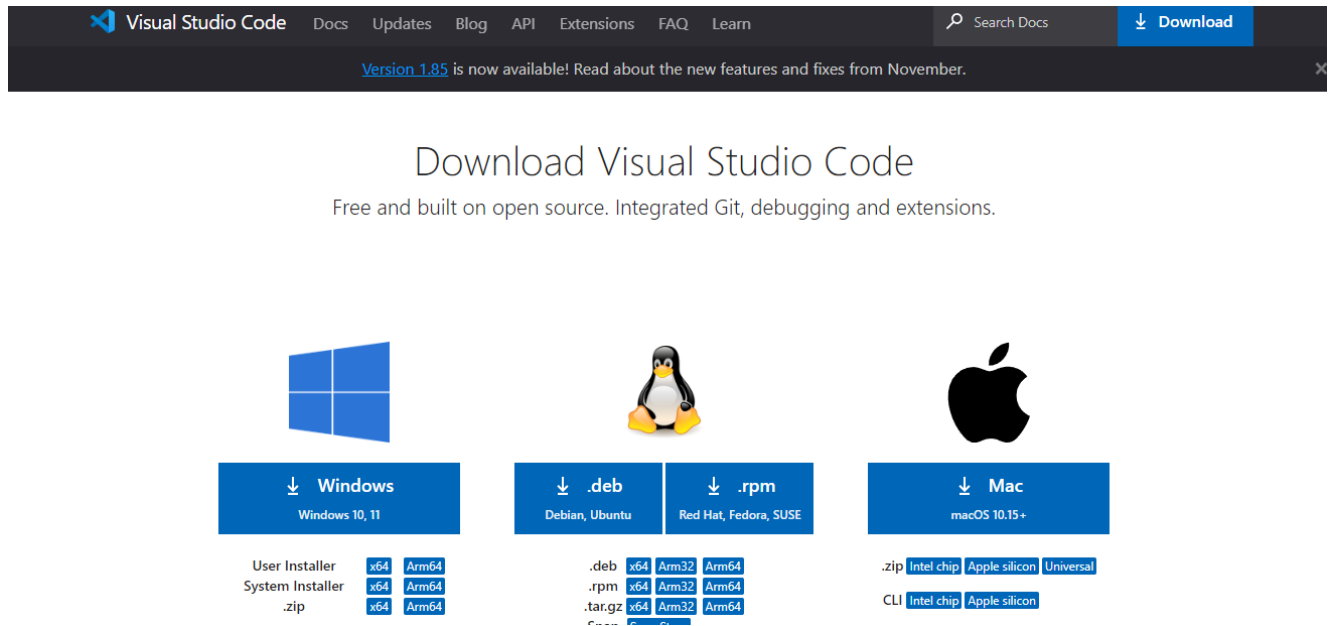
 Get help

Back End del juego Conquista Matemática

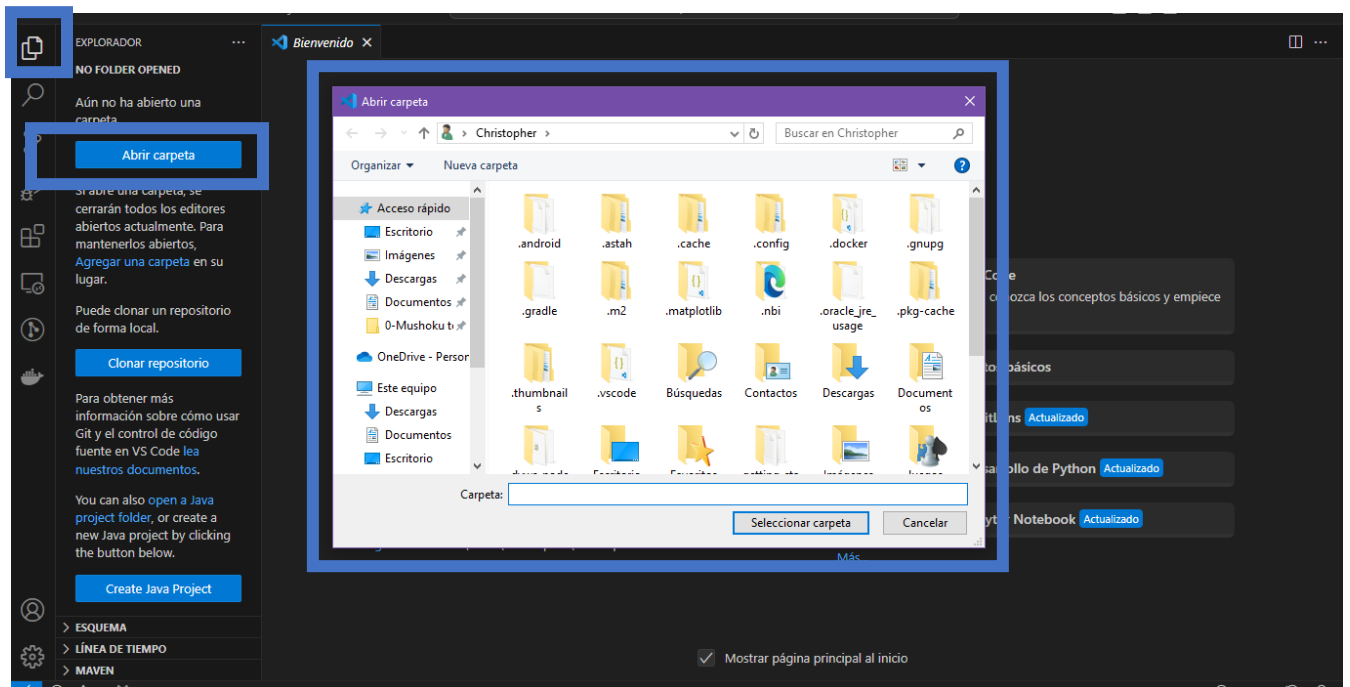
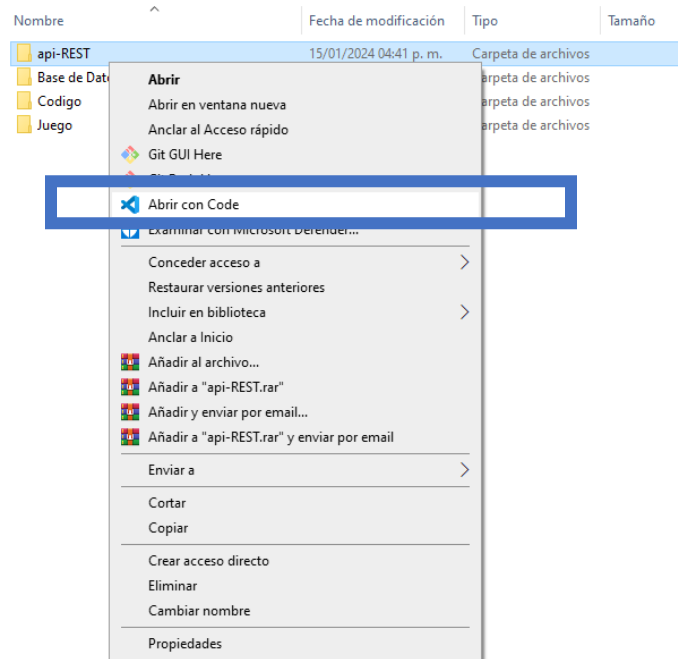
Instalación del back end

En el repositorio existe una carpeta llamada api-REST que contiene nuestro back end para lo siguiente necesitamos tener la herramienta de visual studio code.

Para ello nos vamos a la pagina oficial de visual y descargamos la versión que se adapte a nuestro equipo: <https://code.visualstudio.com/download>



Una vez descargado lo instalamos y abrimos la carpeta api-REST con visual studio code, para ello puedes usar el menú contextual o abrir visual studio y buscar la carpeta con el explorador

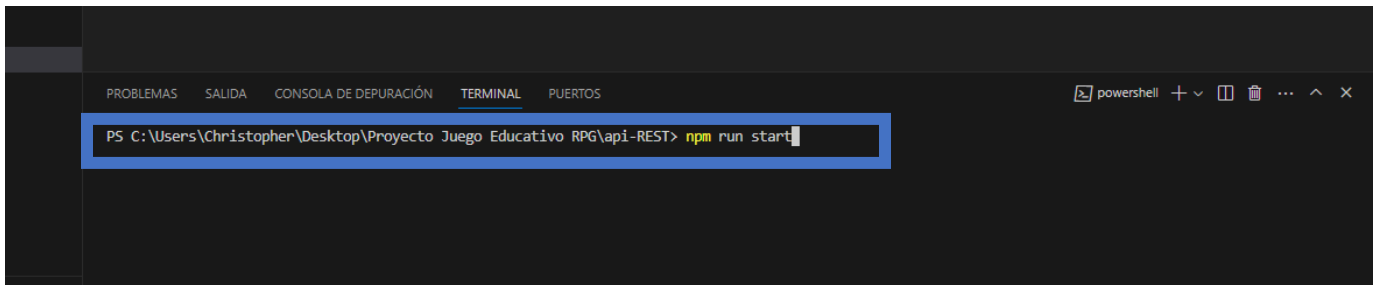


Una vez abierta nuestra carpeta abrimos un nuevo terminal con las teclas `ctrl + mayus + ñ`, una vez abierta la terminal utilizaremos los siguientes comandos de manera consecutiva. (es importante tener en cuenta que la base de datos ya se importo y nuestro servicio de apache y MySQL están activos en xampp)

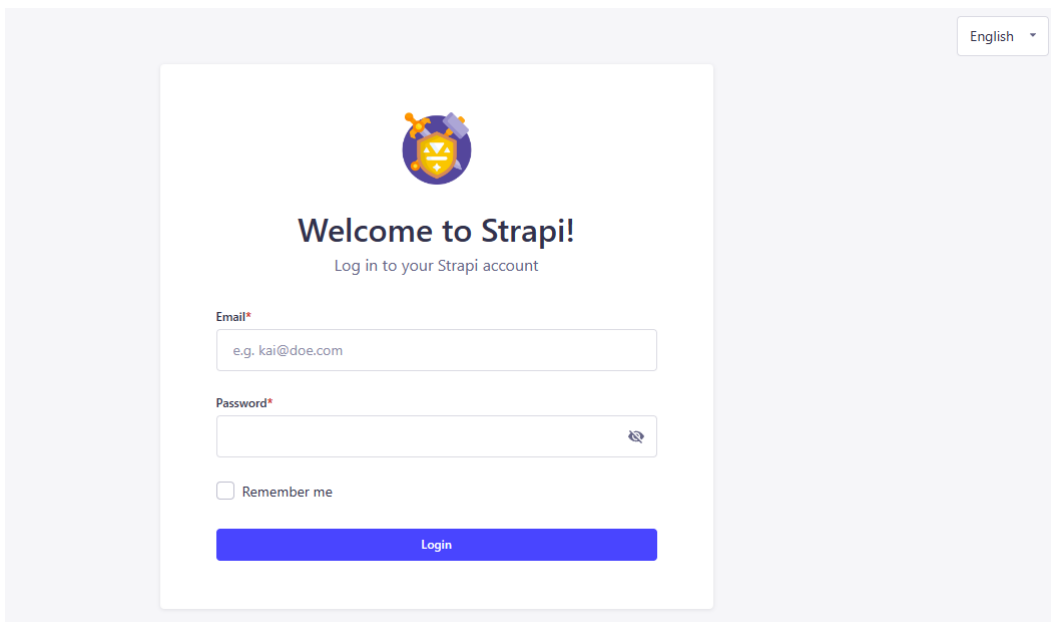
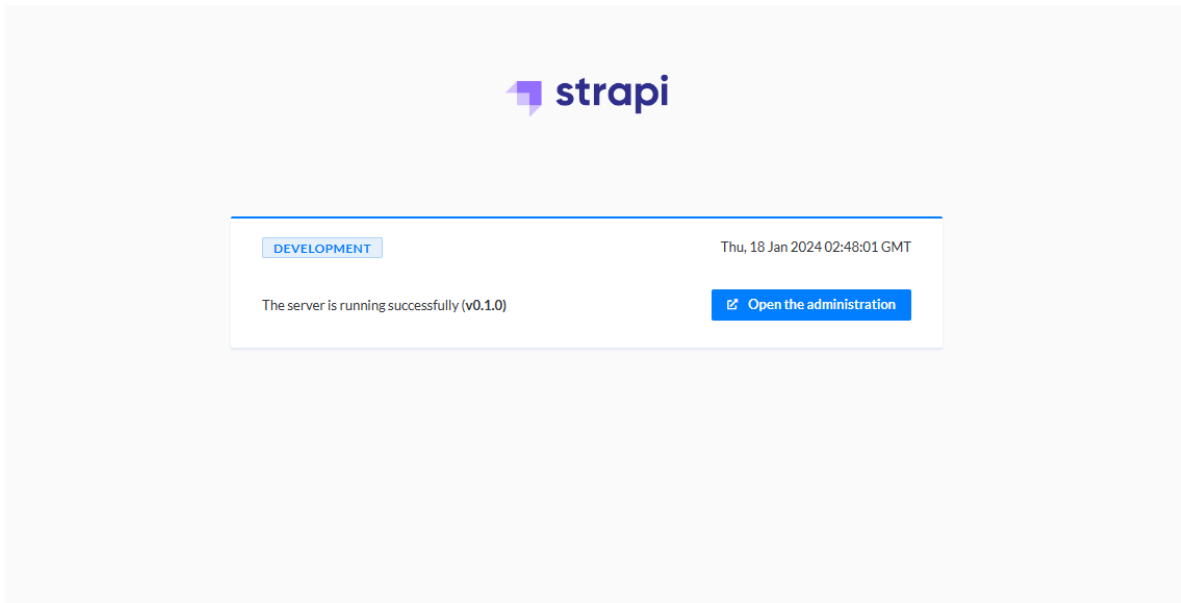
`npm install ->` Para instalar los modulos de node necesarios

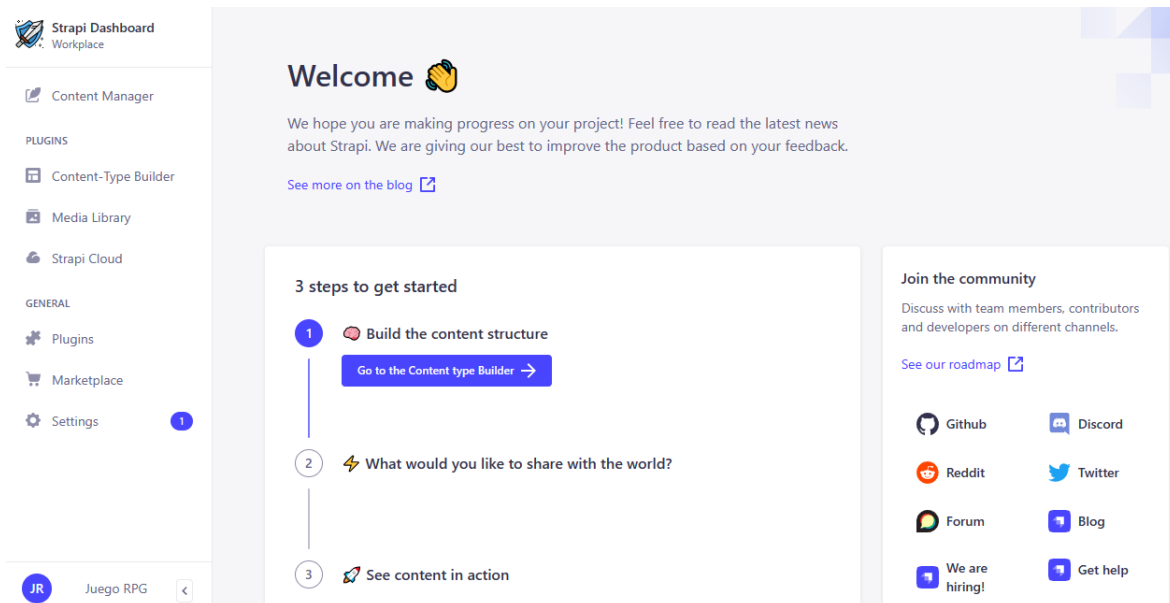
`npm run start ->` Para iniciar con nuestro servicio

ctrl + c -> Para terminar nuestro servicio



A screenshot of a Visual Studio Code terminal window. The terminal title bar shows 'powershell'. The terminal content displays the command prompt 'PS C:\Users\Christopher\Desktop\Proyecto Juego Educativo RPG\api-REST>' followed by the command 'npm run start' which has been executed. The command is highlighted with a blue selection box.





Credenciales de acceso.

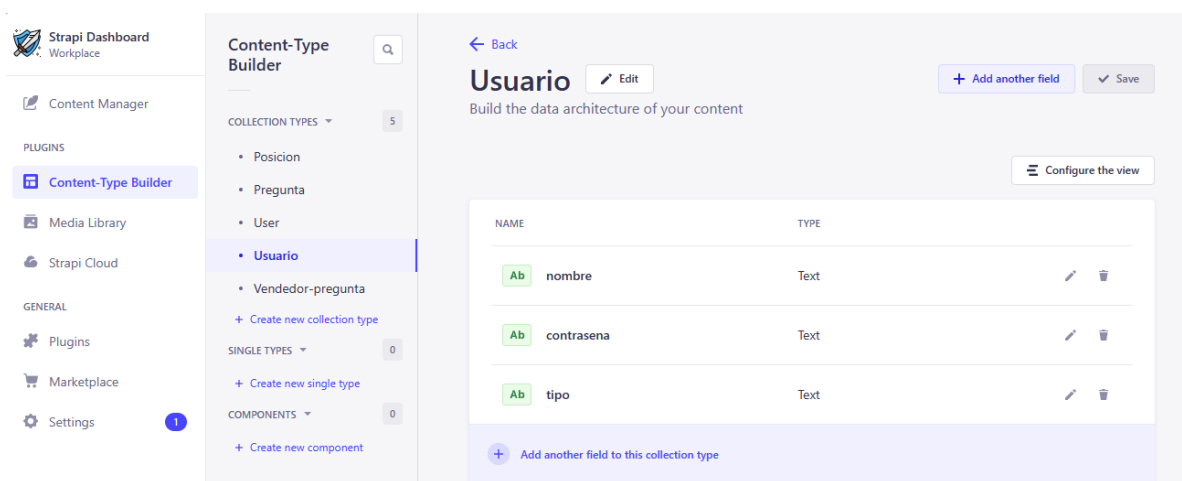
Para poder acceder como administrador y modificar la información contenida dentro de la base de datos necesitaremos introducir las siguientes credenciales.

Email	Contraseña
user@gmail.com	Strapi1234

Modificar información.

Para agregar o modificar la información vamos a usar dos apartados principales, Content Manager y Content-Type Builder.

Content-Type Builder: En esta podemos crear y modificar los valores de tablas existentes, para ser más específicos podemos asignar nombres a las tablas y agregar nuevos campos para asignarles nombres y valores a dichos campos.



Para este proyecto tenemos tres tablas principales:

Usuario: Donde se almacenarán los usuarios de nuestro juego.

Usuario

Edit

+ Add another field

✓ Save

Build the data architecture of your content

☰ Configure the view

NAME	TYPE
<div>Ab</div> nombre	Text
<div>Ab</div> contraseña	Text
<div>Ab</div> tipo	Text

+ Add another field to this collection type

Pregunta: Donde se almacenan las preguntas de los QTE.

Pregunta

Edit

+ Add another field

✓ Save

Build the data architecture of your content

☰ Configure the view

NAME	TYPE
<div>Ab</div> QTE	Text
<div>Ab</div> Pregunta	Text
<div>Ab</div> respuesta	Text
<div>Ab</div> quien	Text
<div>Ab</div> que	Text
<div>Ab</div> operaciones	Text
<div>Ab</div> contexto	Text

Vendedor-Pregunta: Donde se almacenan preguntas ingresadas por el maestro.

Vendedor-pregunta

Build the data architecture of your content

[Edit](#)[+ Add another field](#)[Save](#)

[Configure the view](#)

NAME	TYPE	
<div>Ab</div> pregunta	Text	Edit Delete
<div>Ab</div> que	Text	Edit Delete
<div>Ab</div> quien	Text	Edit Delete
<div>Ab</div> operaciones	Text	Edit Delete
<div>Ab</div> resultado	Text	Edit Delete
<div>Ab</div> contexto	Text	Edit Delete

Content Manager: En este apartado podremos empezar a llenar las tablas creadas con anterioridad con información.

Para agregar una nueva entrada es necesario que seleccionemos una tabla y en la parte superior aparece la opción de agregar una nueva entrada.

usuario

2 entries found

[+ Create new entry](#)

Es importante que cuando introduzcamos una nueva información además de guardarla también la publiquemos para que se muestren al realizar consultas.

[← Back](#)

Create an entry

API ID : usuario

[Publish](#)[Save](#)

Las tablas antes mencionadas llevan la siguiente información:

Usuarios: Aquí se puede controlar los usuarios que existen registrados en la base de datos.

usuario

2 entries found

+ Create new entry

🔍

Filters

⚙️

<input type="checkbox"/>	ID	NOMBRE ▲	CONTRASENA	TIPO	STATE	
<input type="checkbox"/>	2	Arturo	1234	profesor	Published	<div>✎</div> <div>📄</div> <div>🗑</div>
<input type="checkbox"/>	3	User	1234	estudiante	Published	<div>✎</div> <div>📄</div> <div>🗑</div>

← Back

Create an entry

API ID : usuario

✓ Publish

Save

nombre

contrasena

tipo

• Editing draft version

INFORMATION

Created

now

By

-

Last update

now

By

-

✎ Edit the model

☰ Configure the view

Preguntas: Aquí se introducen las preguntas que aparecerán en los QTE durante el desarrollo del juego. Es importante considerar que en los apartados de que, quien, operación y respuesta tener 4 opciones separadas por comas, la primera opción introducida será tomada como correcta y las demás incorrectas (op-corecta,op-incorrecata,op-incorrecata,op-incorrecata).

[← Back](#)

pregunta

6 entries found

[+ Create new entry](#)

<input type="checkbox"/>	ID	QTE	PREGUNTA	RESPUESTA	STATE
<input type="checkbox"/>	1	QTE-1	¿Cuántas explosiones de fuegos artificiales ver...	80,90,58,42	Pul
<input type="checkbox"/>	2	QTE-1	¿Cuántas explosiones de fuegos artificiales hab...	15,54,23,54	Pul
<input type="checkbox"/>	3	QTE-1	¿A qué hora verás la explosión?	8:00:06 p.m,8:50:06 p.m,8:00:10 p.m,9:00:06 p.m	Pul
<input type="checkbox"/>	4	QTE-1	¿Cuántos cohetes se han lanzado en total?	20,30,41,54	Pul
<input type="checkbox"/>	5	QTE-1	¿A qué altura explotó el segundo fuego artifici...	100,200,300,400,500	Pul
<input type="checkbox"/>	6	QTE-1	¿Cuántos cohetes hay en total?	45,50,23,02	Pul

QTE-1

API ID : pregunta

[✓ Unpublish](#)

[Save](#)

QTE

QTE-1

Pregunta

¿Cuántas explosiones de fuegos artificiales verás en total?

quien

Juan, Fuegos Artificiales,Cohete,Minuto

que

Fuegos Artificiales, Explosiones,Cohetes,Minutos

operaciones

Multiplicacion,Division,Suma,Resta

respuesta

80,90,58,42

contexto

Si un espectáculo de fuegos artificiales dura 20 minutos y hay 4 explosiones de fuegos artificiales por minuto.

▪ **Editing published version**

INFORMATION

Created last month
By Juego RPG
Last update 2 days ago
By Juego RPG

[✎ Edit the model](#)

[☰ Configure the view](#)

[🗑 Delete this entry](#)

Preguntas Vendedor: Aquí se puede controlar y ver las diversas preguntas que los maestros introduzcan para que el vendedor ambulante las muestre durante el juego.

[← Back](#)

vendedor-pregunta

2 entries found

[+ Create new entry](#)

[Filters](#) [Settings](#)

<input type="checkbox"/>	ID	PREGUNTA	QUE	CONTEXTO	STATE
<input type="checkbox"/>	2	¿Cuántos listones se necesitan quitar?	Bomba, Listones,Juan,Color	Para encender la bomba se necesitan quitar 3 l...	Published
<input type="checkbox"/>	1	¿Qué fracción de la bomba nos queda por co...	Bomba,Plomo,Material,Bolsa	Al querer Juan encender la bomba, se da cuent...	Published

10

Entries per page

[<](#) [1](#) [>](#)

Bomba, Listones,Juan,Color

API ID : vendedor-pregunta

[✓ Unpublish](#) [Save](#)

pregunta

¿Cuántos listones se necesitan quitar?

que

Bomba, Listones,Juan,Color

operaciones

Suma,Resta,Multiplicación,División

contexto

Para encender la bomba se necesitan quitar 3 listones rojos y 4 azules,

quien

Juan,Bomba,Listones,Color

resultado

7,8,9,2

• Editing published version

INFORMATION

Created

last month

By

Juego RPG

Last update

2 days ago

By

Juego RPG

[✎ Edit the model](#)

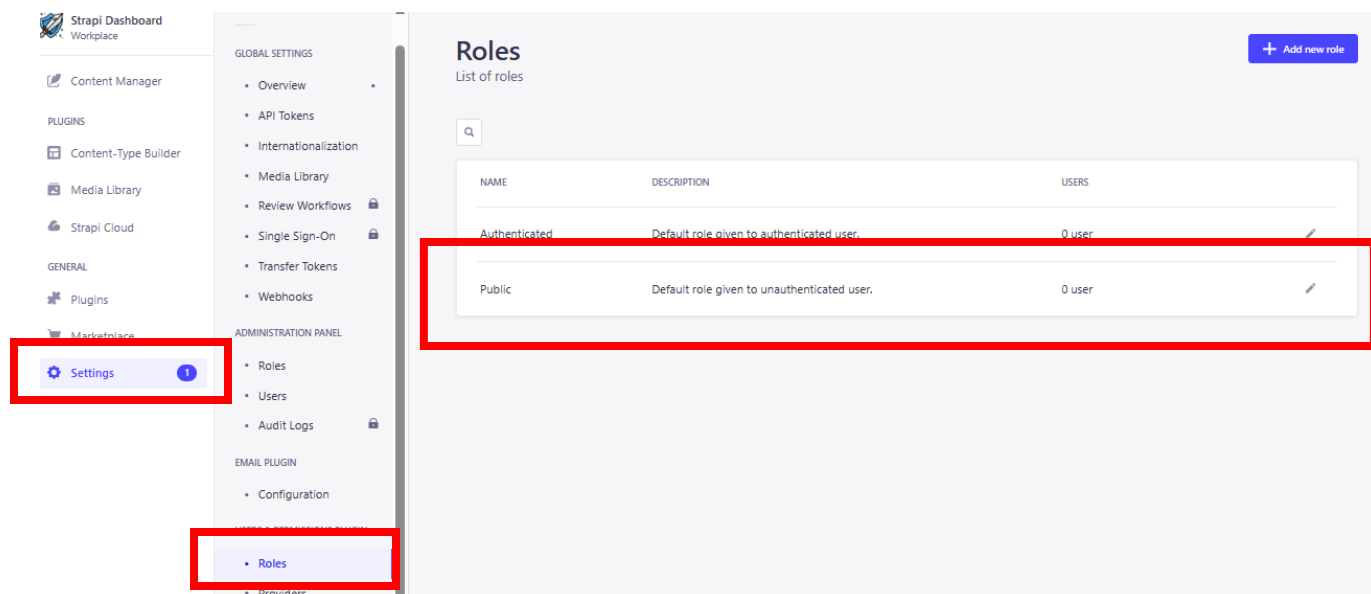
[☰ Configure the view](#)

[🗑 Delete this entry](#)

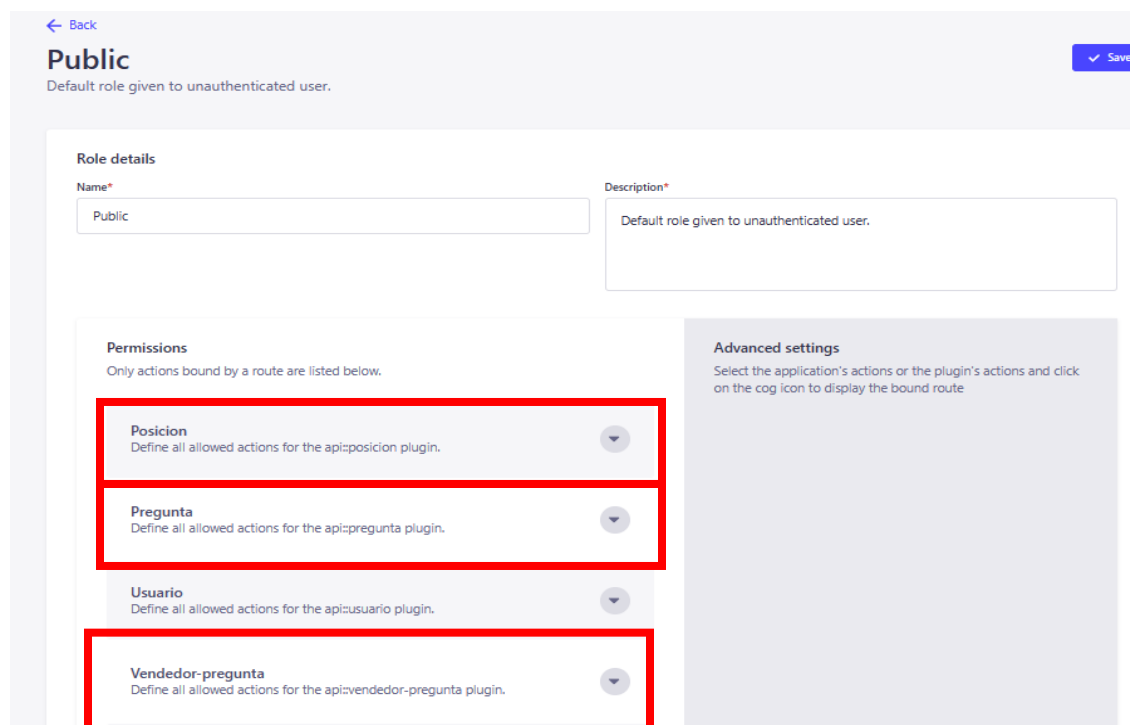
Activando los puntos de la API.

Una vez terminado el ingreso de la información tenemos que darle permisos para que al realizar las consultas se mande la información que ya registramos.

Para ello nos vamos a la parte de configuración y en **USERS & PERMISSIONS PLUGIN** seleccionamos 'Roles', aquí nos mostraran los roles disponibles, en esta parte escogeremos 'Public'.



Dentro de 'Public' nos mostraran las tablas que creamos, solamente tenemos que ir seleccionándolas para habilitar nuestras conexiones, para terminar pulsamos Save para guardar los cambios.



← Back

Public

Default role given to unauthenticated user.

✓ Save

Role details

Name*
Public

Description*
Default role given to unauthenticated user.

Permissions
Only actions bound by a route are listed below.

Posicion
Define all allowed actions for the api:posicion plugin.

POSITION ☒ Select all

- ☒ create
- ☒ delete
- ☒ find
- ☒ findOne
- ☒ update

Advanced settings
Select the application's actions or the plugin's actions and click on the cog icon to display the bound route

Al finalizar podemos comprobar que nos devuelva información al hacer las consultas, mediante <http://localhost:1337> y completándolo con las siguientes rutas:

Preguntas

POST: /api/preguntas

GET: /api/preguntas

PUT: /api/preguntas/:id

DELETE: /api/preguntas/:id

GET O FIND ONE: /api/preguntas/:id

Posiciones

POST: /api/posicions

GET: /api/posicions

PUT: /api/posicions/:id

DELETE: /api/posicions/:id

GET O FIND ONE: /api/posicions/:id

Preguntas vendedor

POST: /api/vendedor-preguntas

GET: /api/vendedor-preguntas

PUT: /api/vendedor-preguntas/:id

DELETE: /api/vendedor-preguntas/:id

GET O FIND ONE: /api/vendedor-preguntas/:id

Despliegue de Strapi.

Para finalizar necesitamos usar el siguiente código para que strapi compile de manera correcta el panel de administrador con todos los cambios introducidos:

```
npm run build
```

```
C:\Users\Christopher\Desktop\API-Juego\api-REST>npm run build

> api-rest@0.1.0 build
> strapi build

- Building build context
\ Building build context [INFO] Including the following ENV variables as part of the JS bundle:
  - ADMIN_PATH
  - STRAPI_ADMIN_BACKEND_URL
  - STRAPI_TELEMETRY_DISABLED
✓ Building build context (398.2761000394821ms)
- Building admin panel
✓ Building admin panel (49715.13810002804ms)

C:\Users\Christopher\Desktop\API-Juego\api-REST>
```

Cambios en la base de datos.

Las bases de datos que Strapi puede manejar es algo amplia, usualmente se maneja una base de datos local, sin embargo cuando la base de datos esta en la web a veces se vuelve necesario realizar modificaciones a la información sobre la base de datos, para estos casos nos vamos a los archivos de Strapi y buscamos uno que tenga de nombre .env, en este archivo se guardan las variables de entorno que Strapi usa. Dentro de este archivo nos vamos al apartador de #Database y aquí podemos modificar la información que necesitamos.

