

Back-End Development

Connecting

#14 Strapi & MongoDB





strapi

- The most advanced Node.js Content Management Framework (Headless CMS) to build powerful API with no effort.
- Before you go any further, ensure these technologies are installed on your computer:
 - Node.js* at least v9.0.0
 - NPM* at least v5.0.0
 - MongoDB* at least v3.4
- More info: <https://strapi.io>

#1 Activate MongoDB Server

■ Open terminal:

```
$ cd C:\Program Files\MongoDB\Server\3.6\bin
```

```
$ mongod -dbpath C:\data\db //jika belum//
```

```
$ mongod
```

■ If OK, it will show “Waiting for connections on port 27017”. Then open new terminal:

```
$ cd C:\Program Files\MongoDB\Server\3.6\bin
```

```
$ mongo
```

#2 Create Db, User & Collection

- Membuat db strapi_db:

```
$ use strapi_db
```

- Membuat db user:

```
$ db.createUser({user:"lintang",  
pwd:"1234", roles:["readWrite",  
"dbAdmin"]});
```

- Membuat collection “karyawan”:

```
$ db.createCollection('karyawan')
```



#3 Setup Strapi

■ Install Strapi CLI:

```
$ npm install strapi@alpha -g
```

```
$ strapi -v
```

■ Create Strapi project:

```
$ strapi new lin_strapi
```

- ✓ Choose your database: *MongoDB (recommended)*
- ✓ Database name: *strapi_db*
- ✓ Host: *127.0.0.1*
- ✓ Port: *27017*
- ✓ Username: *Lintang*
- ✓ Password: *1234*

Welcome!

To finish setup and secure your app, please create the first user (root admin) by entering the necessary information below.

Username

Password

Confirmation Password

Email

Ready to start

#4 Launch Server then Create User

- Enter project dir then launch the server:

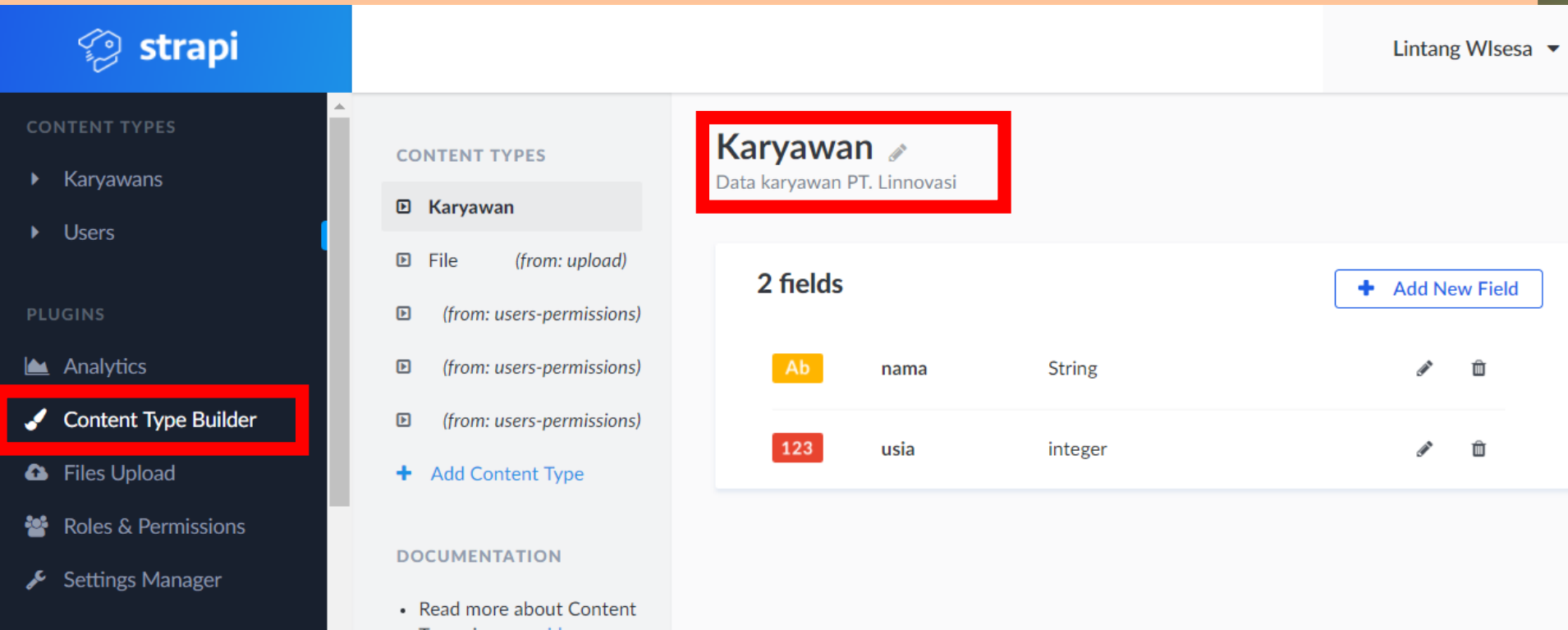
```
$ cd lin_strapi
```

```
$ strapi start
```

- Then open administration panel on browser (***localhost:1337/admin***) to ***create the first user!***

#5 Create Content Type

- From **Content Type Builder** menu, create a content type (*model*) with several fields (*props*). Eg: Content: **Karyawan** with Fields: **Nama** (string) & **Usia** (number)



The screenshot displays the Strapi Content Type Builder interface. On the left, a dark sidebar contains the 'Content Type Builder' menu item, which is highlighted with a red rectangle. The main area shows the 'Karyawan' content type, also highlighted with a red rectangle. Below the title, it says 'Data karyawan PT. Linnovasi'. The '2 fields' section lists two fields: 'nama' (String) and 'usia' (integer). The 'Add New Field' button is visible in the top right of the fields section.

CONTENT TYPES

- ▶ Karyawans
- ▶ Users

PLUGINS

- Analytics
- Content Type Builder**
- Files Upload
- Roles & Permissions
- Settings Manager

CONTENT TYPES

- ▣ Karyawan
- ▣ File (from: upload)
- ▣ (from: users-permissions)
- ▣ (from: users-permissions)
- ▣ (from: users-permissions)
- + Add Content Type

DOCUMENTATION

- Read more about Content Types in our guide

Karyawan ✎
Data karyawan PT. Linnovasi

2 fields [+ Add New Field](#)

Ab	nama	String	✎	🗑
123	usia	integer	✎	🗑

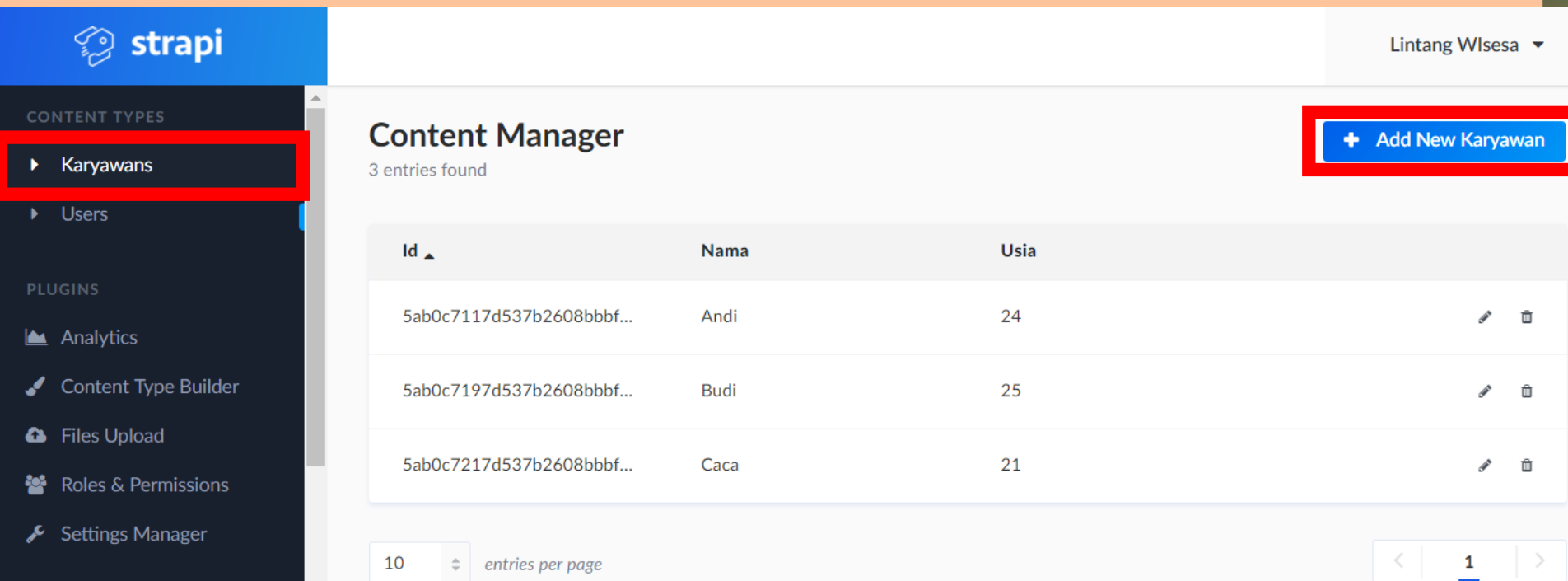
■ Add some data to Karyawan:

Nama: Andi, Usia: 24







Nama: Budi, Usia: 25

Nama: Caca, Usia: 21

#6 Add Data to Content Type



The screenshot shows the Strapi Content Manager interface. On the left, the 'CONTENT TYPES' sidebar has 'Karyawans' selected. On the right, the 'Content Manager' view for 'Karyawans' shows 3 entries found. A red box highlights the 'Add New Karyawan' button in the top right corner. The table below lists the entries:

Id	Nama	Usia	
5ab0c7117d537b2608bbbf...	Andi	24	 
5ab0c7197d537b2608bbbf...	Budi	25	 
5ab0c7217d537b2608bbbf...	Caca	21	 

At the bottom, there is a pagination control showing '10 entries per page' and a page number '1'.

#7 Allowing API Access

- From **Roles & Permissions**, click **Public** then select allowed actions.

Create - **POST**
Findone - **GET :id**
Destroy - **DELETE**
Update - **PUT**
Find - **GET**

The screenshot displays the Strapi administration interface. On the left, the 'Roles & Permissions' menu item is highlighted. The main content area shows the 'Permissions' configuration for the 'Karyawan' role. A red rectangular box highlights the permissions table, which lists actions for the 'Karyawan' role. The 'create', 'findone', and 'find' actions are checked. The 'find' action is highlighted with a blue button. To the right, the 'Advanced settings' panel is visible, showing a dropdown menu set to 'None' and a 'Bound route to Karyawan.find' section with a 'GET /karyawan' button.

Karyawan		Select All
<input checked="" type="checkbox"/> create	<input type="checkbox"/> destroy	<input checked="" type="checkbox"/> find
<input checked="" type="checkbox"/> findone	<input type="checkbox"/> update	



- A powerful GUI platform to make API development faster & easier, from building API requests through testing, documentation & sharing.
- Postman has features for every API developer: request building, tests & pre-request scripts, variables, environments & request descriptions, designed to work seamlessly together.
- Download from getpostman.com

**Try to POST & GET
with POSTMAN!**

#8 GET data

- With Postman, try to GET data from Strapi API:

GET `http://localhost:1337/karyawan`

GET `http://localhost:1337/karyawan?nama=Andi`

GET `http://localhost:1337/karyawan?usia_gte=22`

GET `http://localhost:1337/karyawan?_sort=nama:desc`

GET `http://localhost:1337/karyawan?_limit=30`

Back-End Development

Connecting

#14 Strapi & MongoDB

