

Directus Installation and Configuration with mysql

Step-1: Our first step is to install directus onto our local machine. In the terminal enter

```
npx create-directus-project <project-name>
```

```
mojjadavenkatnikhil@robotSPACE-QBiX-Pro-EHLA6412H-A1:~$ npx create-directus-project example_directus;
You are running Node.js 20.16.0.
Directus requires Node.js 18, specifically version 18.17 or higher (>=18.17 & <19).
Please adjust your Node.js version and try again.
```

Note: Now if we are using a latest node.js version like 20.16.0 in my example then we will encounter with a problem like directus does not support node.js versions below 18.17 and above 19. So you can install the specified node.js version that is in the given range using

```
nvm install <version>
```

Example: I am installing node.js 18.20.4 version

```
nvm install 18.20.4
```

To verify that we have installed it correctly use 'nvm list' command to check if the node.js version that you have installed is present

```
mojjadavenkatnikhil@robotSPACE-QBiX-Pro-EHLA6412H-A1:~$ nvm list;
v18.20.4
-> v20.16.0
    system
default -> 20 (-> v20.16.0)
iojs -> N/A (default)
unstable -> N/A (default)
node -> stable (-> v20.16.0) (default)
stable -> 20.16 (-> v20.16.0) (default)
lts/* -> lts/iron (-> v20.16.0)
lts/argon -> v4.9.1 (-> N/A)
lts/boron -> v6.17.1 (-> N/A)
lts/carbon -> v8.17.0 (-> N/A)
lts/dubnium -> v10.24.1 (-> N/A)
lts/erbium -> v12.22.12 (-> N/A)
lts/fermium -> v14.21.3 (-> N/A)
lts/gallium -> v16.20.2 (-> N/A)
lts/hydrogen -> v18.20.4
lts/iron -> v20.16.0
```

Step-2: Now use the command 'nvm use 18.20.4' to run that version of node.js

```
mojjadavenkatnikhil@robotSPACE-QBiX-Pro-EHLA6412H-A1:~$ nvm use 18.20.4
Now using node v18.20.4 (npm v10.7.0)
```

Step-3: Installing/Creating a directus project using

```
DB_CLIENT=mysql DB_PORT=3306 DB_HOST=127.0.0.1 DB_USER=<username>  
DB_PASSWORD = <password> DB_DATABASE=<database-name> npx create-directus-project <project-name>
```

IMPORTANT NOTE: Ensure that the details you are entering are same as the details of the sql user you have created and the database that you have created in the mysql.

```
npx create-directus-project  
mojjadavenkatnikhil@robotSPACE-Q8LX-Pro-EHLA6412H-A1:~$ DB_CLIENT=mysql DB_PORT=3306 DB_HOST=127.0.0.1 DB_USER=<username> DB_PASSWORD=<password> DB_DATABASE=<database-name> npx create-directus-project <project-name>
```

Example:

```
mojjadavenkatnikhil@robotSPACE-Q8LX-Pro-EHLA6412H-A1:~$ DB_CLIENT=mysql DB_PORT=3306 DB_HOST=127.0.0.1 DB_USER=directususer DB_PASSWORD=Directus@123 DB_DATABASE=new npx create-directus-project directus_new_1
```

It will take some time to install, after some time you will be prompted to select the database proceed with 'MySQL / MariaDB / Aurora' option, then enter the details in the way it is shown below

IMPORTANT NOTE: Make sure the details that you have entered in the command and in the following prompts are similar.

```
? Database Host: 127.0.0.1  
? Port: 3306  
? Database Name: <database-name>  
? Database User: <username>  
? Database Password: *****
```

Example:

```
? Choose your database client MySQL / MariaDB / Aurora  
? Database Host: 127.0.0.1  
? Port: 3306  
? Database Name: new  
? Database User: directususer  
? Database Password: *****
```

If everything is done correctly your directus configuration with mysql is done and you will be asked to create an email and password for your Directus API/server.

```
Create your first admin user:  
? Email admin@example.com  
? Password *****
```

Step-6: After the successful installation and configuration of directus you can change your current directory to the directus project/folder directory that you have given while installing directus using 'cd' command like

```
cd <directus-project-folder>
```

```
$ cd <project-name>
```

Example:

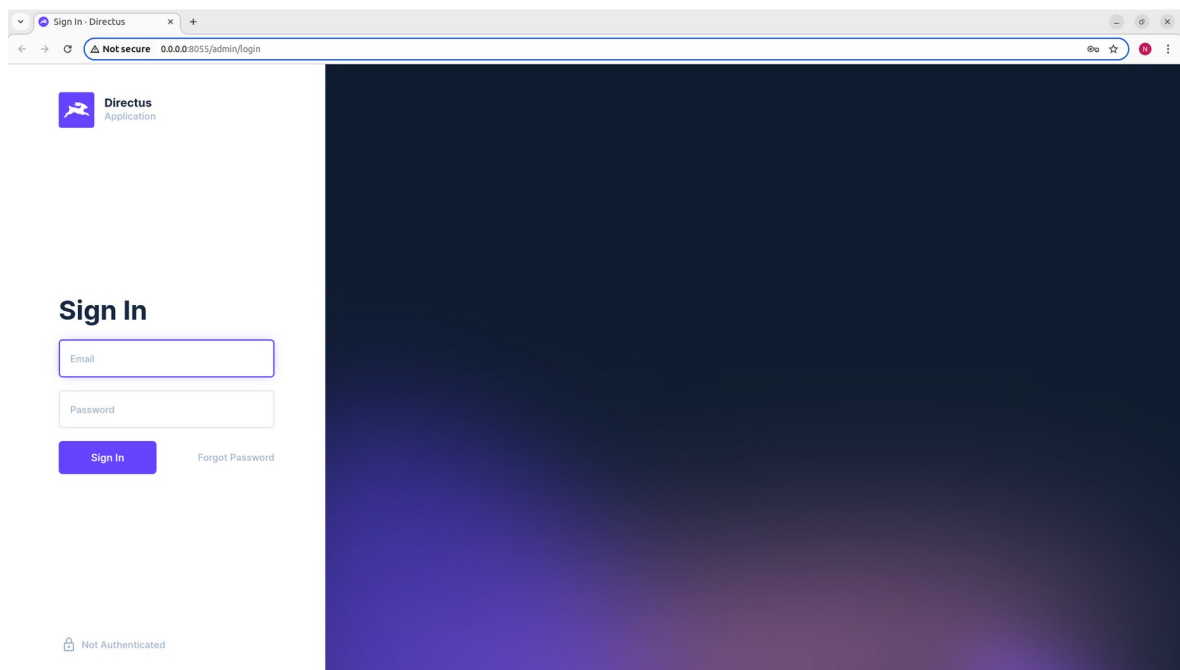
```
$ cd directus_demo
```

Step-7: Now that you have entered into the directus folder type 'npx directus start' command to start your local directus server.

```
mojjadavenkatnikhil@robotSPACE-QBiX-Pro-EHLA6412H-A1:~/directus_demo$ npx directus start
[11:40:50.941] WARN: "PUBLIC_URL" should be a full URL
[11:40:51.152] INFO: Server started at http://0.0.0.0:8055
```

Step-8: Go to the link where your server has started in your preferred browser, here i am using google chrome browser.

Note: You can just hold CTRL and click on the url that will automatically open the domain in your default browser .



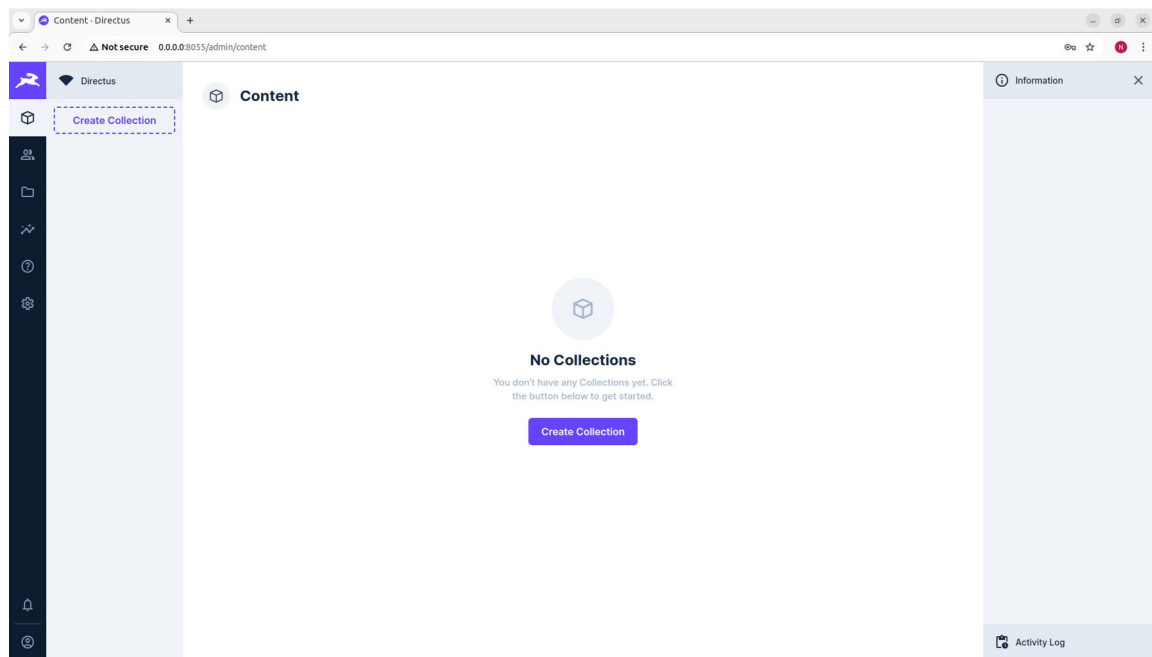
Step-9: Enter the email and password that you have given while installation

Example:

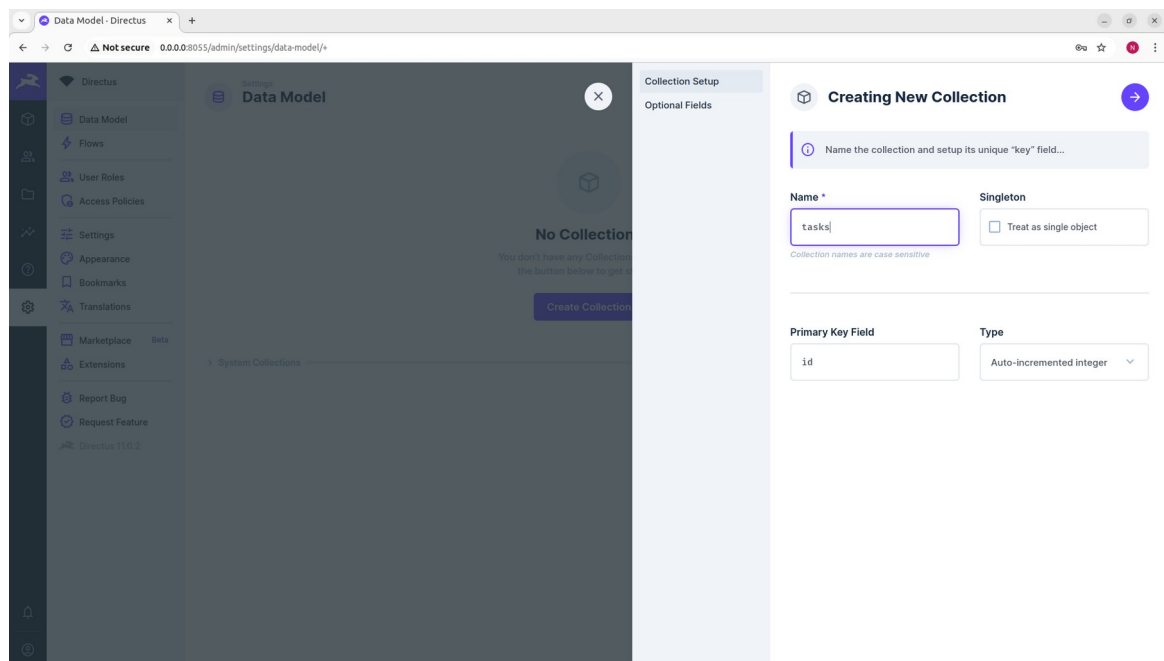
Email: admin@example.com // your email

Password: password // your password

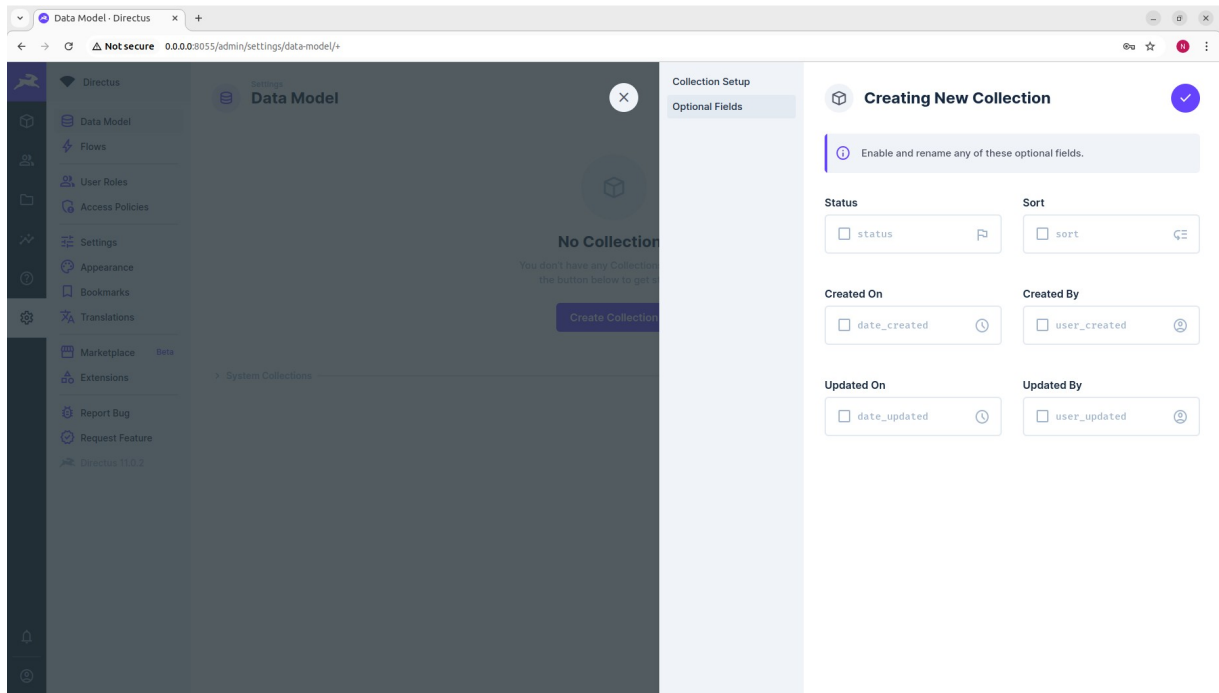
Step-10 Creating a collection: After you successfully login your screen will look like this



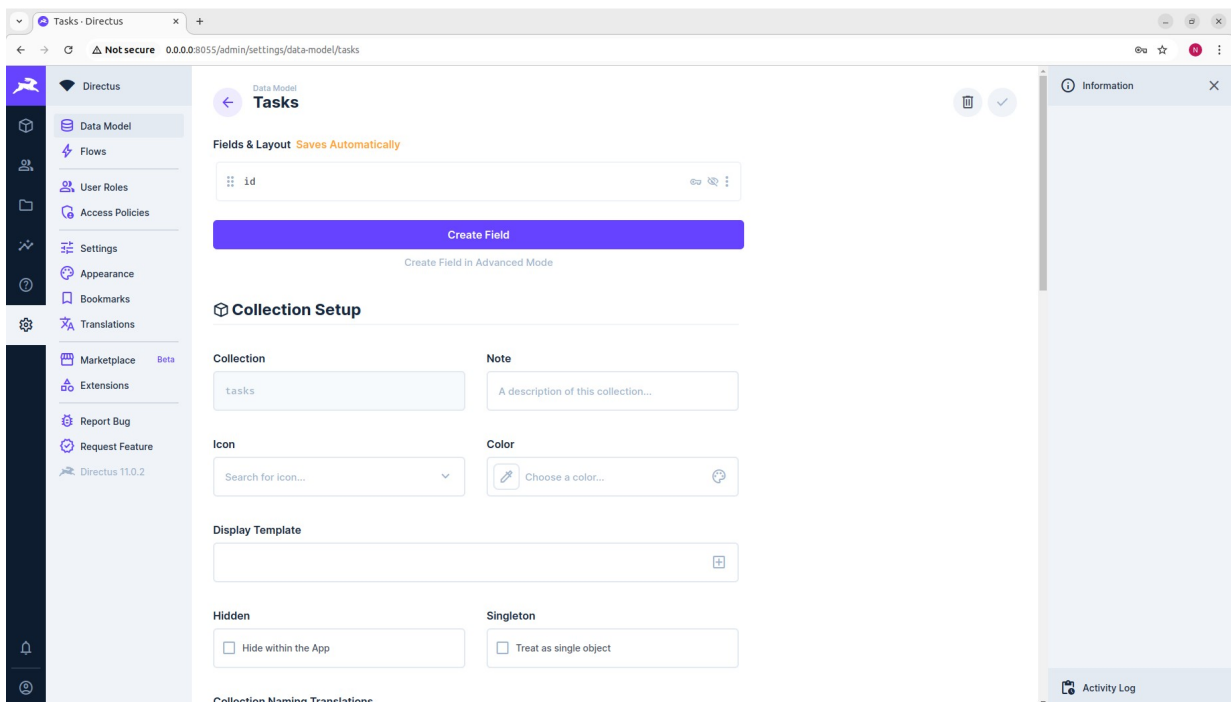
Here click on 'create collection' button, give the collection a name like 'tasks', click the arrow to proceed further.



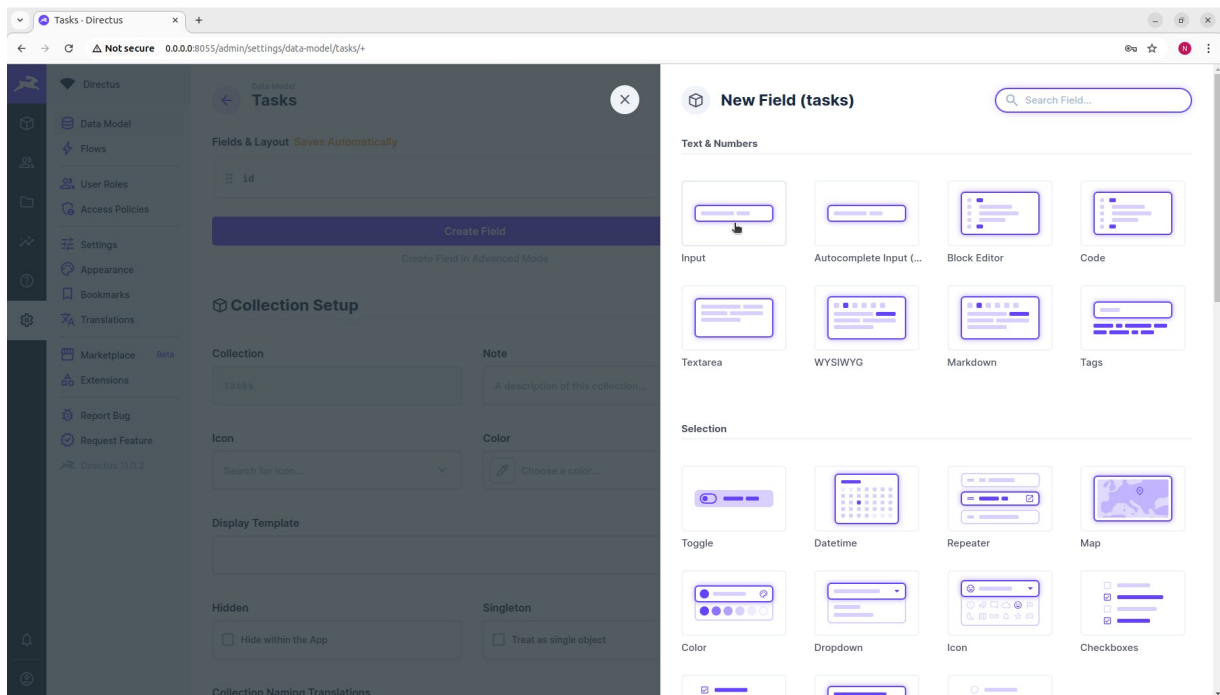
No need to select any option over here and click the tick button, you have created a collection successfully.



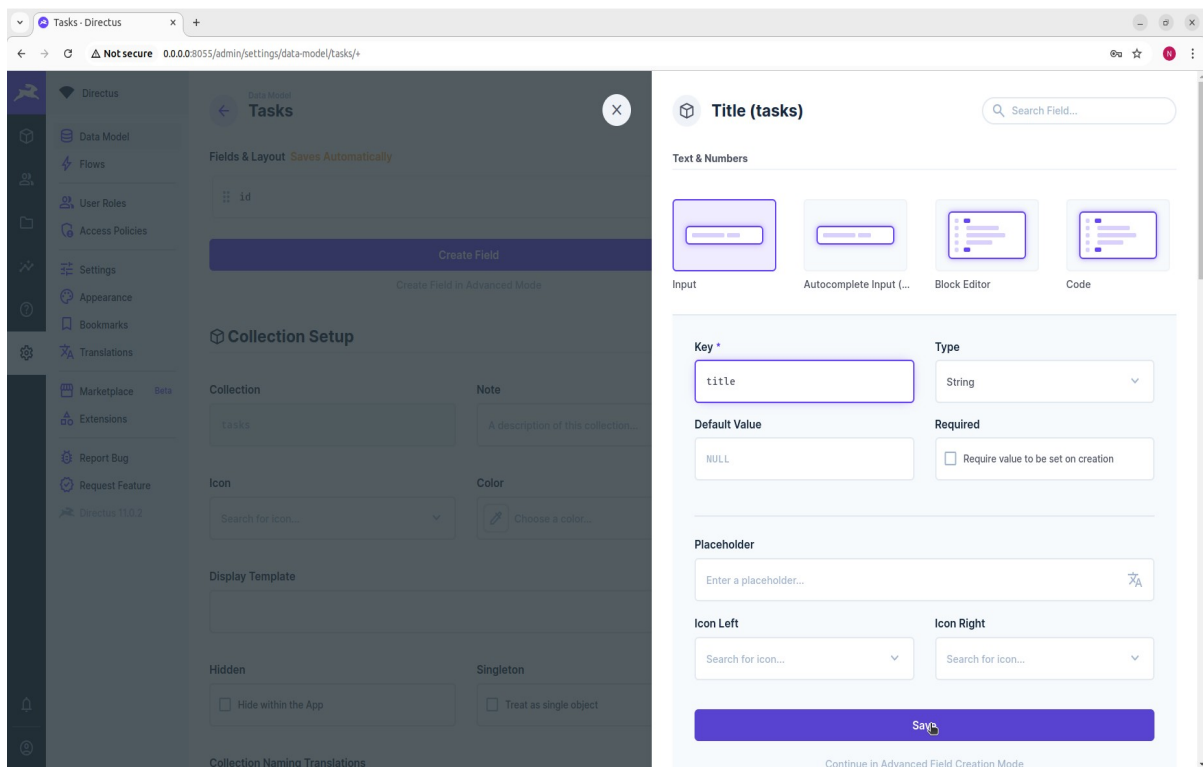
Step-11 Creating Fields: Here click on ‘Create Field’ button to create the task fields that are required for our task management app.



Click on the 'input' card.



Now lets give it a name 'title' and click on save.



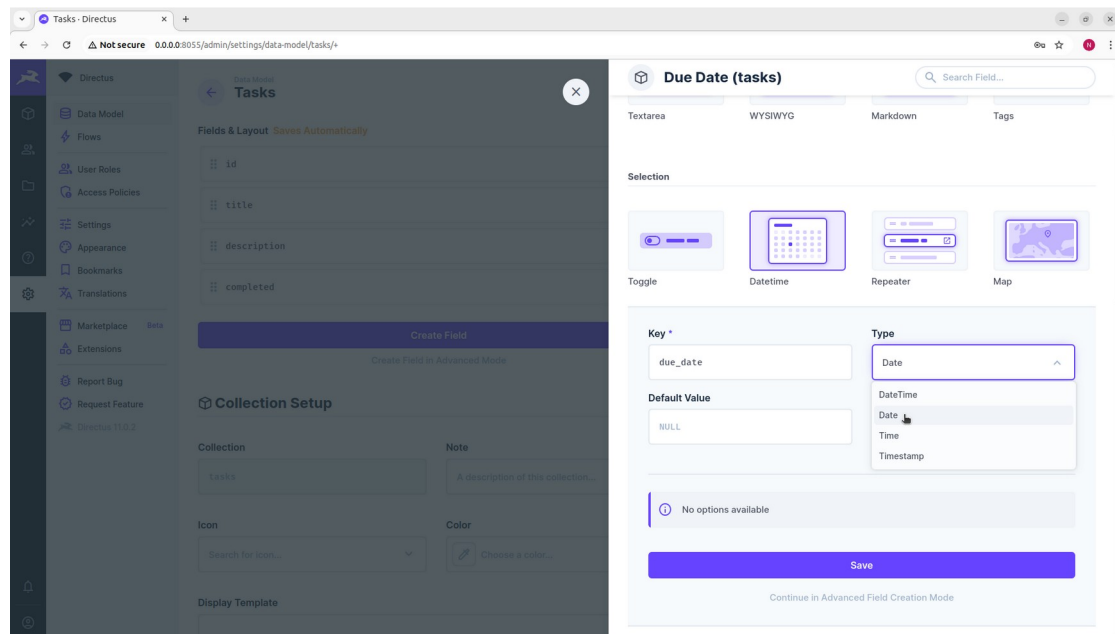
Lets create another field named 'description' and set the type as Text, then click save.

The screenshot shows the Directus admin interface with the 'Tasks' collection selected. The 'Description (tasks)' field configuration screen is open. The 'Key' is 'description' and the 'Type' is 'Text'. The 'Default Value' is 'NULL'. The 'Placeholder' is 'Enter a placeholder...'. The 'Icon Left' and 'Icon Right' are both 'Search for icon...'. The 'Save' button is at the bottom right. The 'Collection Setup' panel on the left shows the 'tasks' collection with fields 'id', 'title', and 'description'.

Create another field click on 'Toggle' card under Selection category and name it 'completed', keep the other fields as default do not change them and click save.

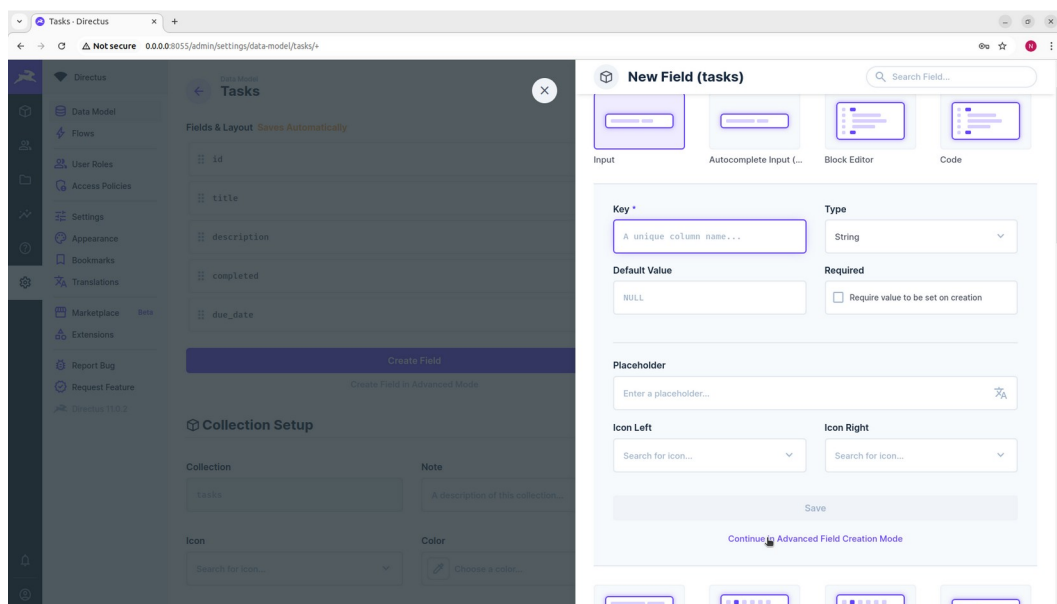
The screenshot shows the Directus admin interface with the 'Tasks' collection selected. The 'Completed (tasks)' field configuration screen is open. The 'Key' is 'completed' and the 'Type' is 'Boolean'. The 'Default Value' is 'Enabled'. The 'Required' checkbox is 'Require value to be set on creation'. The 'Icon On' is 'Check Box' and the 'Icon Off' is 'Check Box Outline Blank'. The 'Color On' and 'Color Off' are both 'Choose a color...'. The 'Label' is 'Enabled'. The 'Save' button is at the bottom right. The 'Collection Setup' panel on the left shows the 'tasks' collection with fields 'id', 'title', 'description', and 'completed'.

Create another field, click on 'Datetime' card under Selection category name it 'due_date', select type as Date and then click save.



Now lets create the last field which is important in our collection, click on 'input' card and then click on Continue in Advanced Field Creation Mode.

Note: The purpose for creating the uuid field is so that we can get the records of the users that have created the tasks specifically by authentication, by extracting uuid we will show the users their respective tasks instead of showing every users tasks.



Name the field as 'uuid', select the type as 'UUID', in the 'On Create' filed select 'Save Current user ID', keep the other fields default, after that click on 'Field' tab

The screenshot shows the Directus admin interface for configuring a field named 'uuid' for the 'tasks' collection. The 'Field' tab is selected, displaying the following configuration:

- Key:** uuid
- Type:** UUID
- Length:** Not Available for this Type
- On Create:** Save Current User ID
- On Update:** Do Nothing
- Default Value:** NULL
- Nullable:** ☒ Allow NULL value
- Unique:** ☐ Value has to be unique

The 'Interface' tab is also visible, showing the 'Schema' section.

In the 'Field' tab check the 'Disable editing value' box, next click on 'Interface' tab.

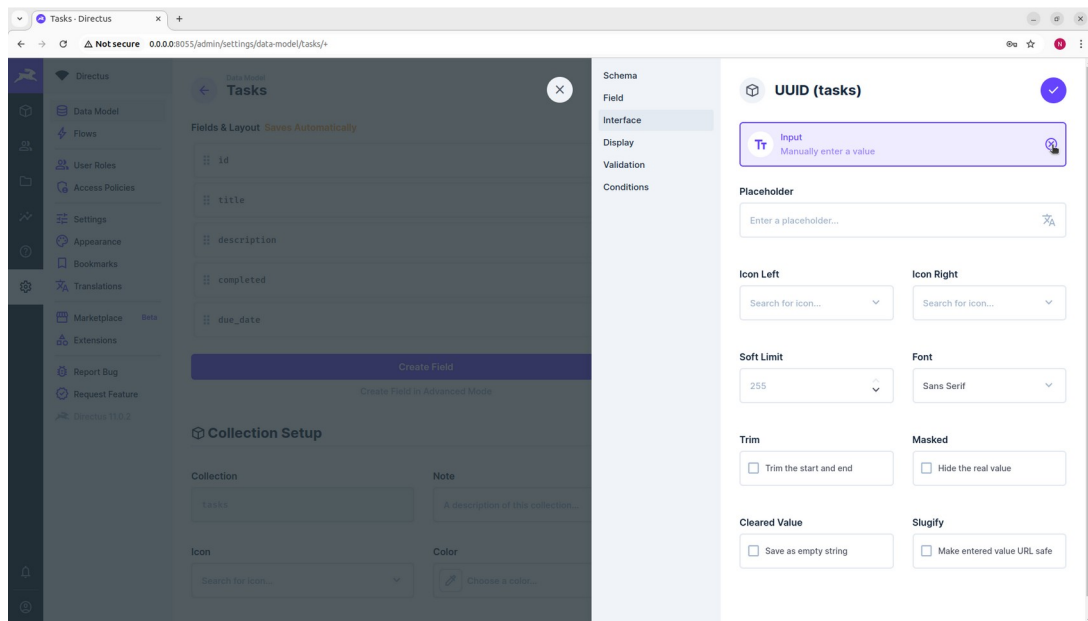
The screenshot shows the Directus admin interface for configuring a field named 'uuid' for the 'tasks' collection. The 'Field' tab is selected, displaying the following configuration:

- Key:** uuid
- Type:** UUID
- Length:** Not Available for this Type
- On Create:** Save Current User ID
- On Update:** Do Nothing
- Default Value:** NULL
- Nullable:** ☒ Allow NULL value
- Unique:** ☐ Value has to be unique

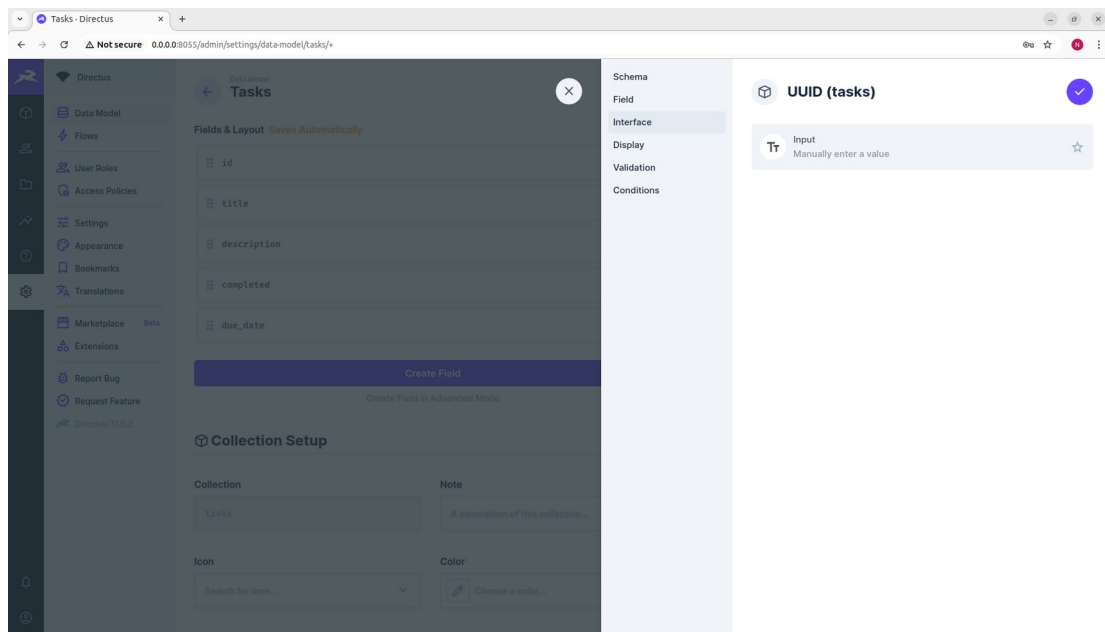
The 'Interface' tab is also visible, showing the 'Schema' section.

In the 'Interface' tab, unselect the 'Input' tab in the top or you can also click the 'x' button in the 'Input' tab.

Before:

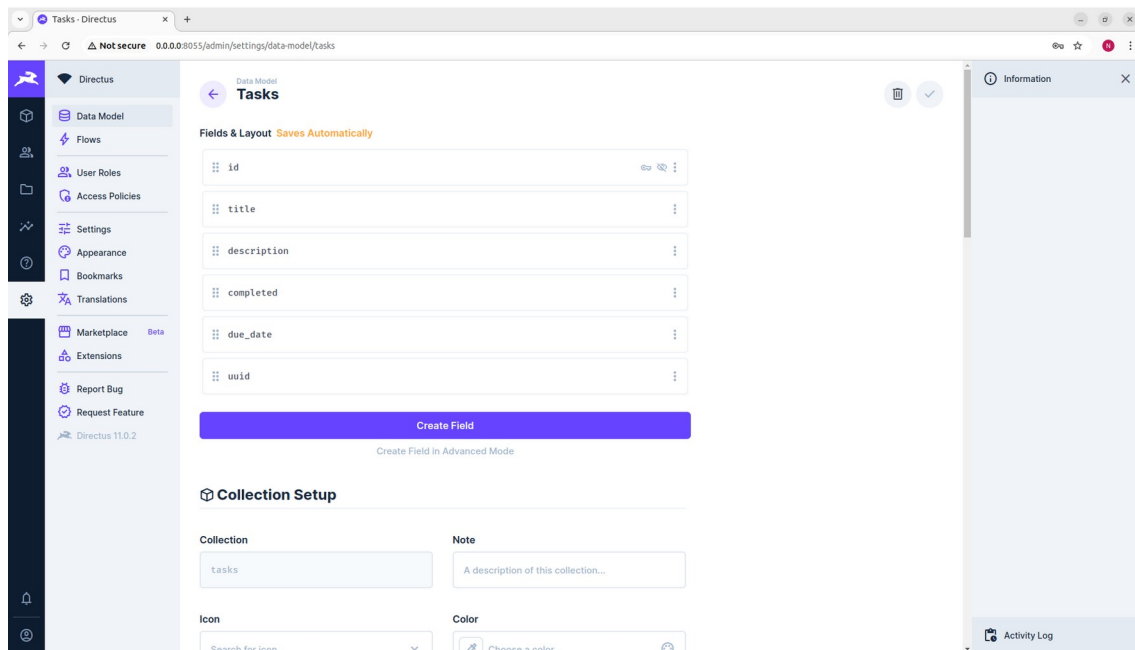


After:

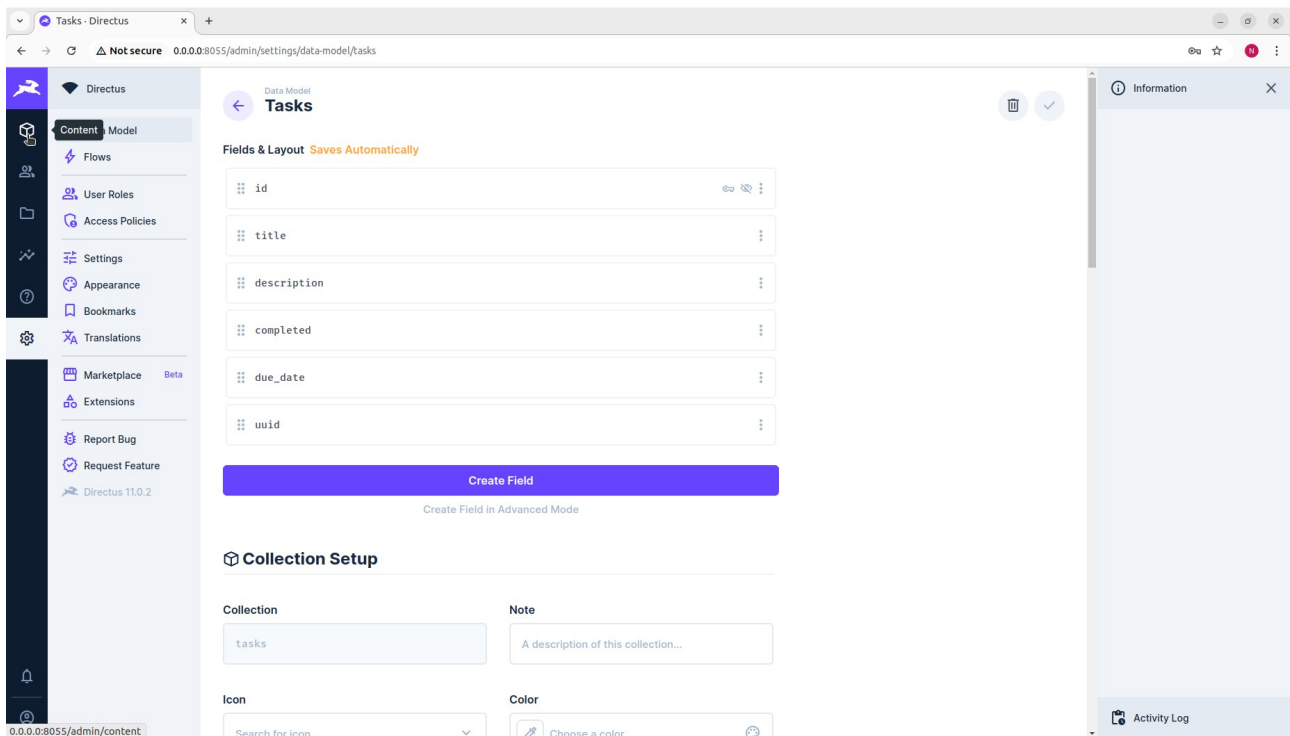


Now click on the 'tick' mark icon to save it.

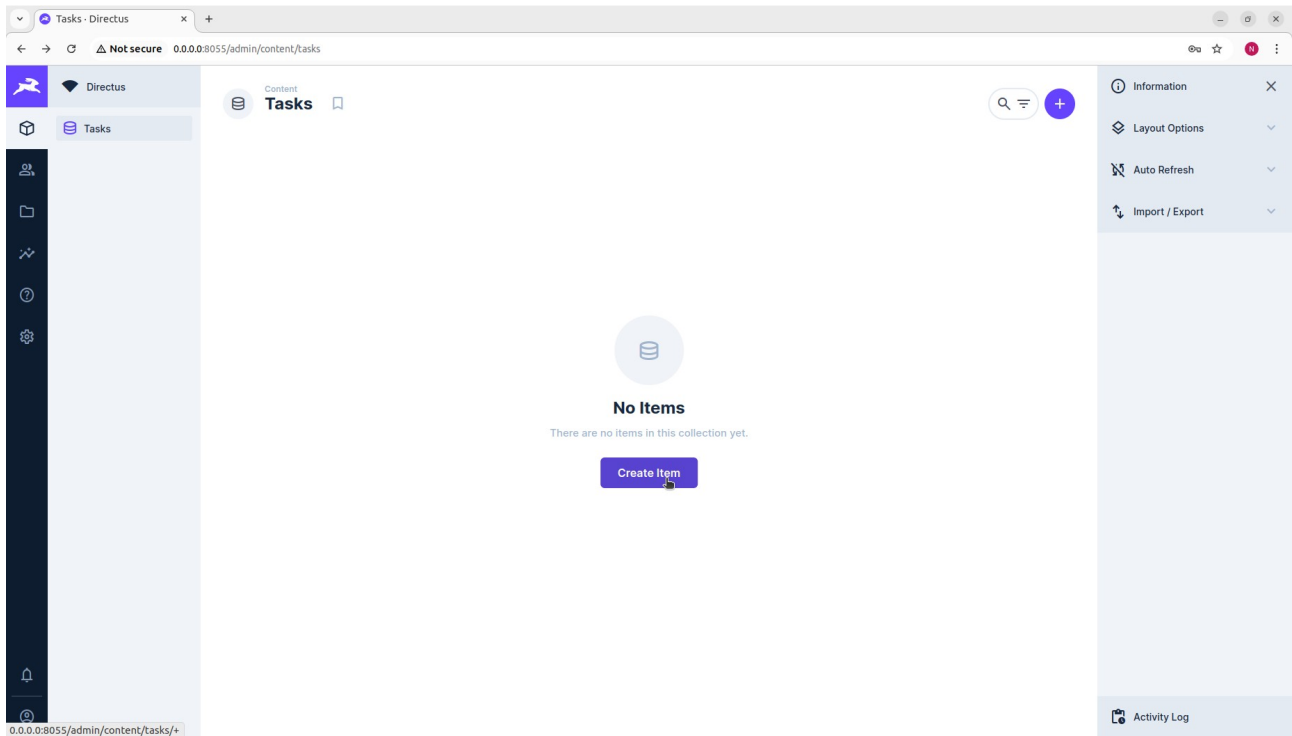
Your collection will look like this if everything is done as said.



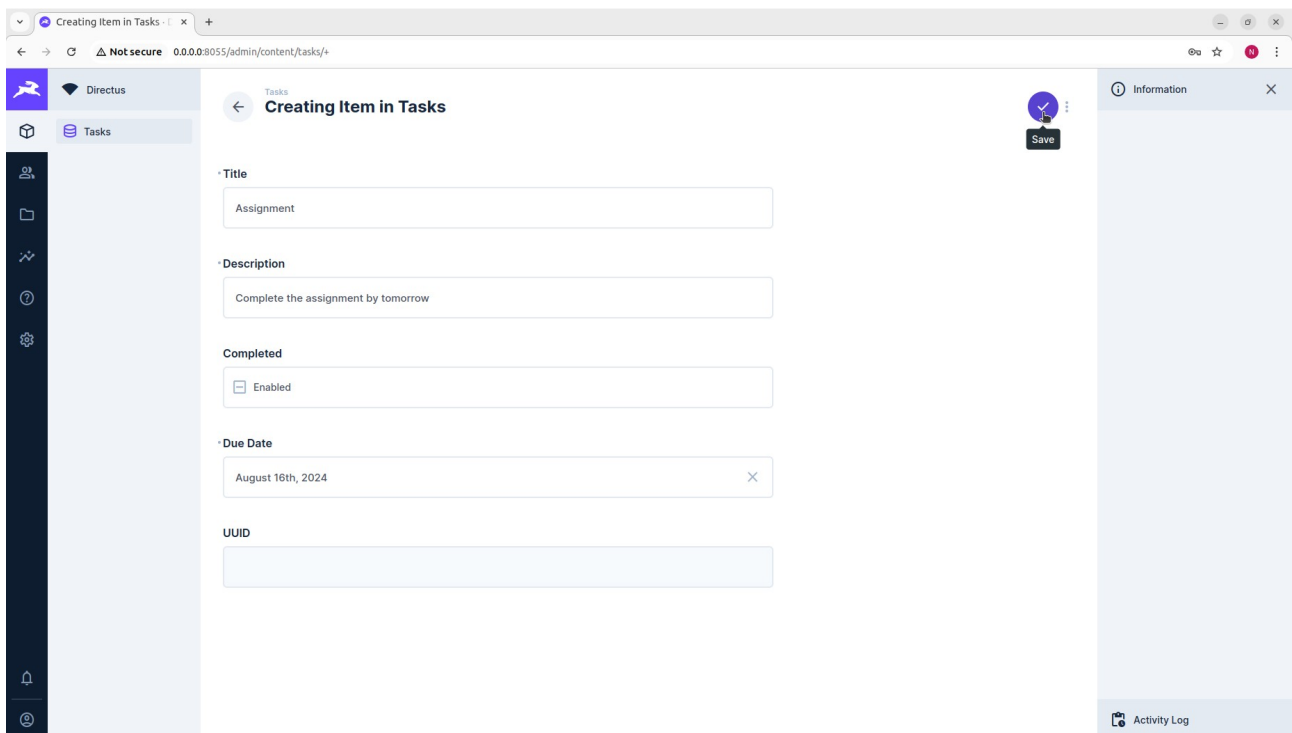
Step-12: Click on the 'Content' module to see the structure of the 'tasks' collection.



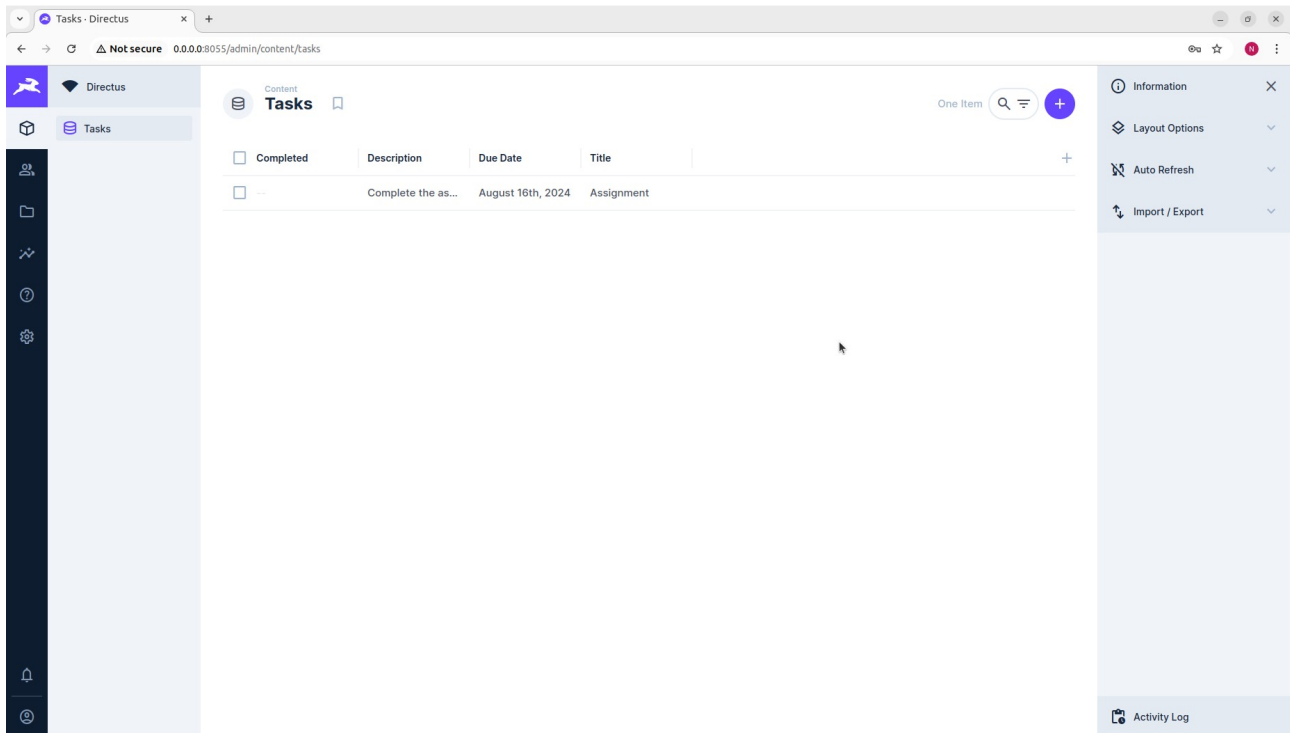
Now Click on ‘Create Item’ button.



Now give some custom values in each field for testing and save it.



Our table should look like this.



Tasks - Directus

Content Tasks

One Item

Completed	Description	Due Date	Title
<input type="checkbox"/>	Complete the as...	August 16th, 2024	Assignment

Information

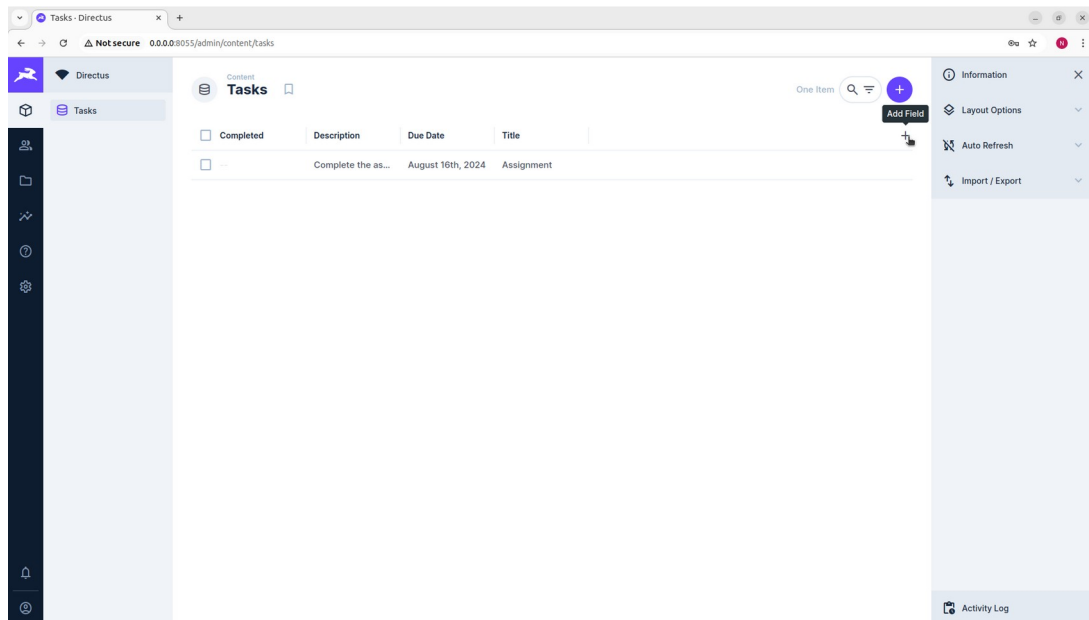
Layout Options

Auto Refresh

Import / Export

Activity Log

Here we cannot see the field 'uuid' that we have created, click on '+' icon near the table Headers field



Tasks - Directus

Content Tasks

One Item

Completed	Description	Due Date	Title
<input type="checkbox"/>	Complete the as...	August 16th, 2024	Assignment

Add Field

Information

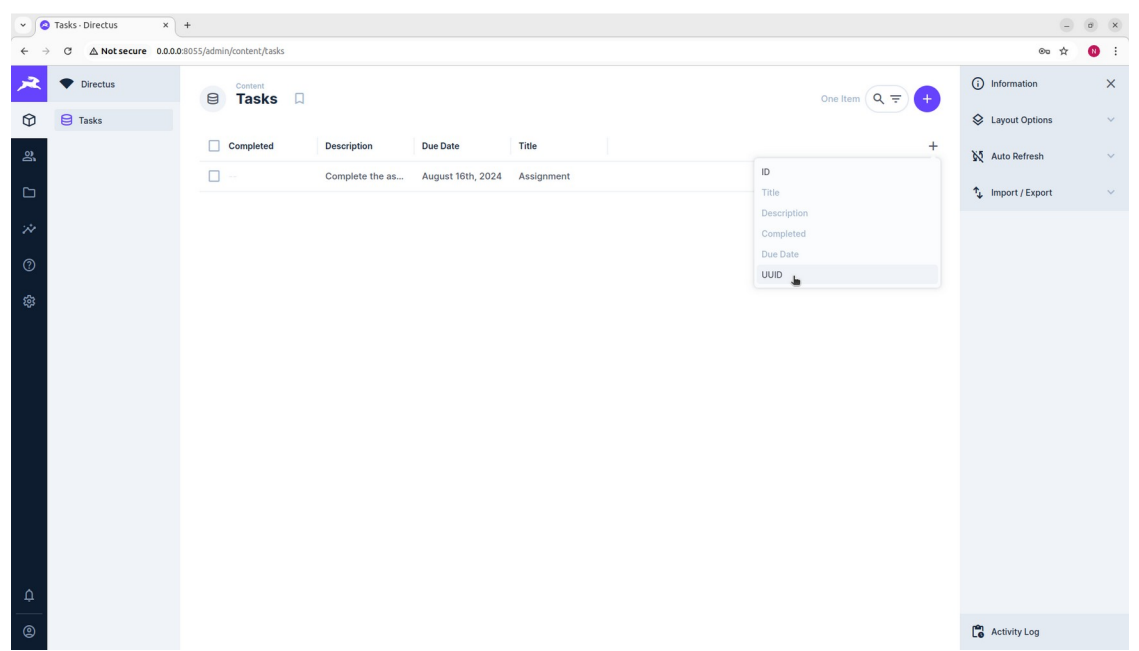
Layout Options

Auto Refresh

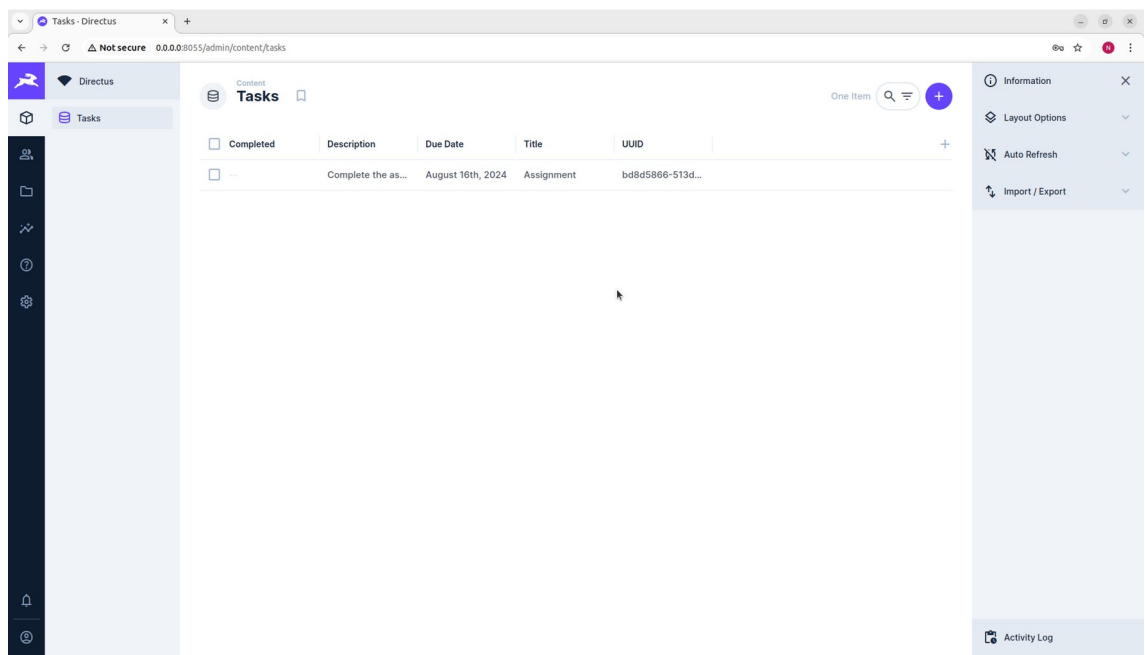
Import / Export

Activity Log

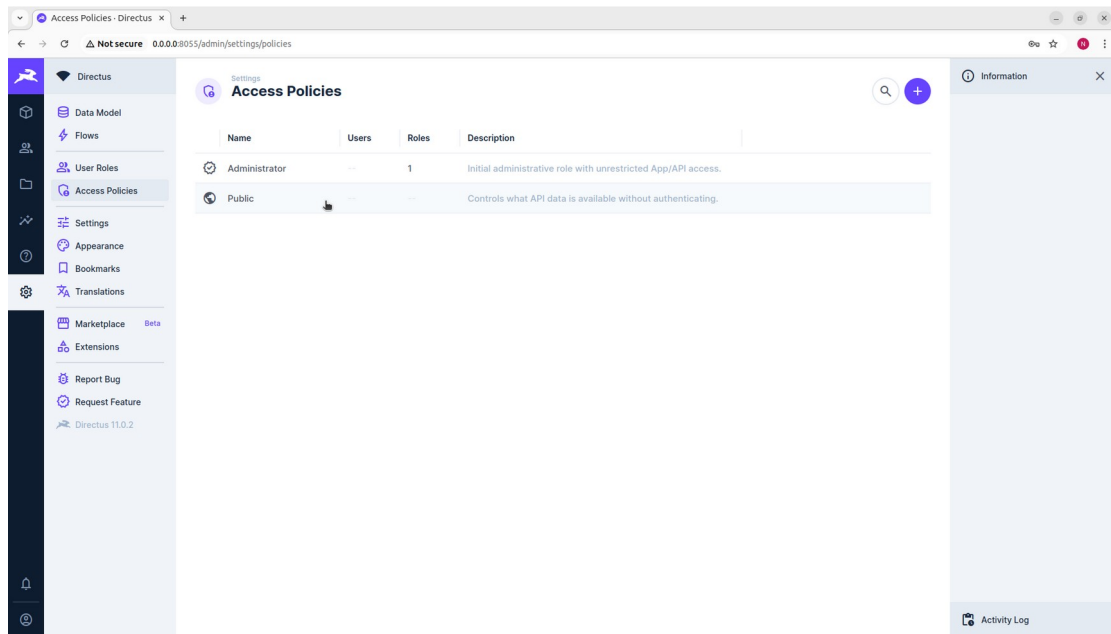
Now select the UUID option.



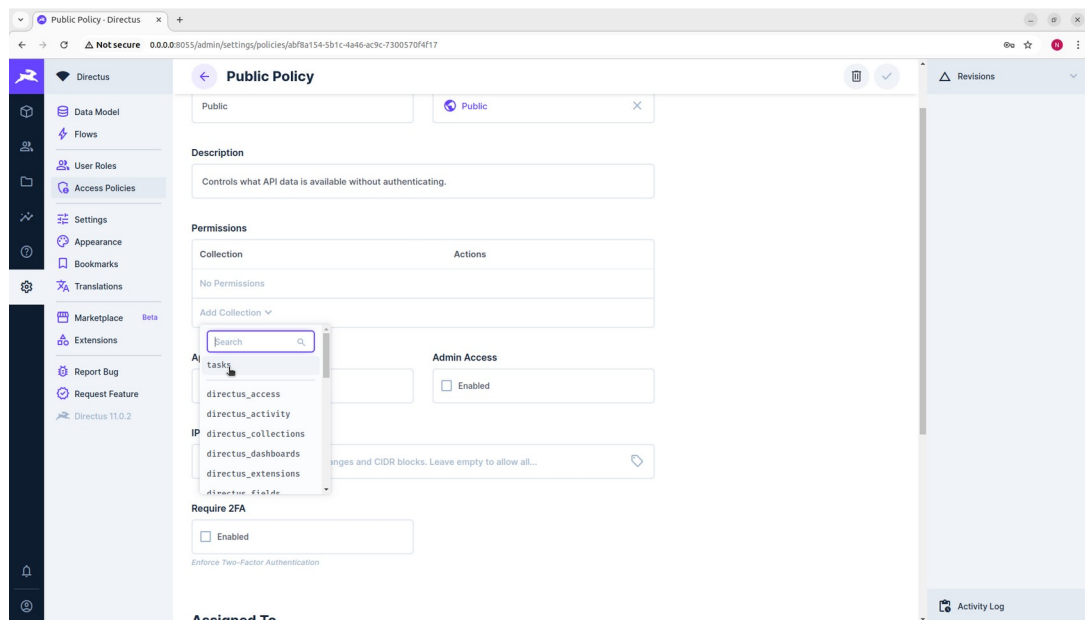
Now you can see it in your table.



Step-13: Go to settings module click on access policies, then click on public



Step-14: In public policy add the tasks collection



Enable the Read,Create,Delete and Update options

The screenshot shows the 'Public Policy' configuration page in Directus. The left sidebar contains navigation options: Directus, Data Model, Flows, User Roles, Access Policies (selected), Settings, Appearance, Bookmarks, Translations, Marketplace (Beta), Extensions, Report Bug, Request Feature, and Directus 11.0.2. The main content area is titled 'Public Policy' and includes the following sections:

- Policy Name:** A text input field containing 'Public'.
- Icon:** A dropdown menu showing 'Public' with a close button.
- Description:** A text area containing 'Controls what API data is available without authenticating.'
- Permissions:** A table with columns 'Collection' and 'Actions'. The 'tasks' collection is listed with actions 'Create', 'Read', 'Update', 'Delete', and 'Share'. Below the table is an 'Add Collection' dropdown.
- App Access:** A checkbox labeled 'Enabled'.
- Admin Access:** A checkbox labeled 'Enabled'.
- IP Access:** A text area with the placeholder 'Add allowed IP addresses, IP ranges and CIDR blocks. Leave empty to allow all...'.
- Require 2FA:** A checkbox labeled 'Enabled'.

On the right side, there is a 'Revisions' panel and an 'Activity Log' at the bottom.

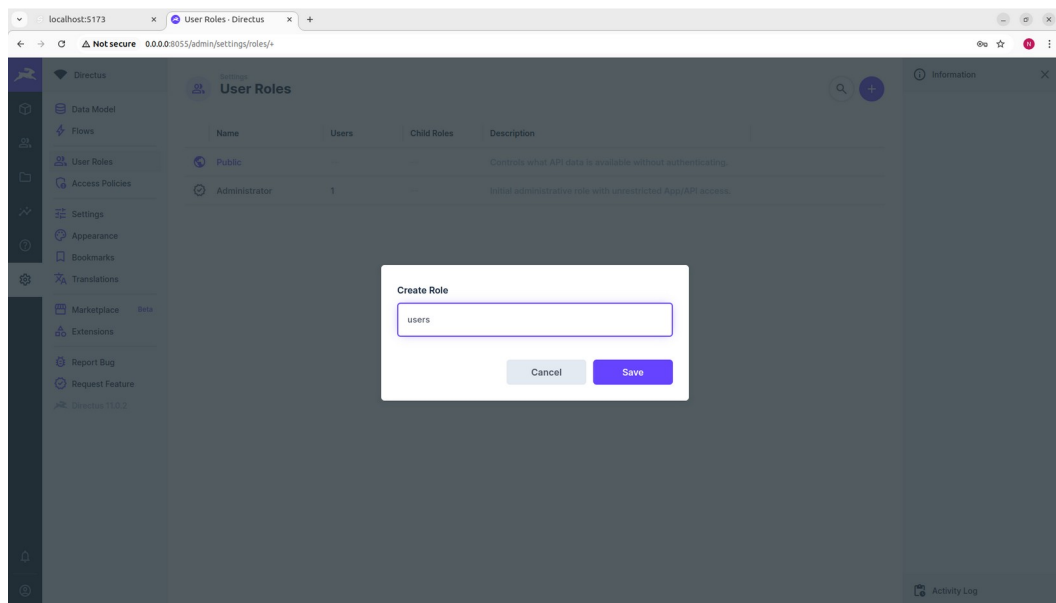
Step-15: Go to user roles above the access policies option

The screenshot shows the 'User Roles' configuration page in Directus. The left sidebar is the same as in the previous screenshot, with 'User Roles' selected. The main content area is titled 'User Roles' and includes a table with the following data:

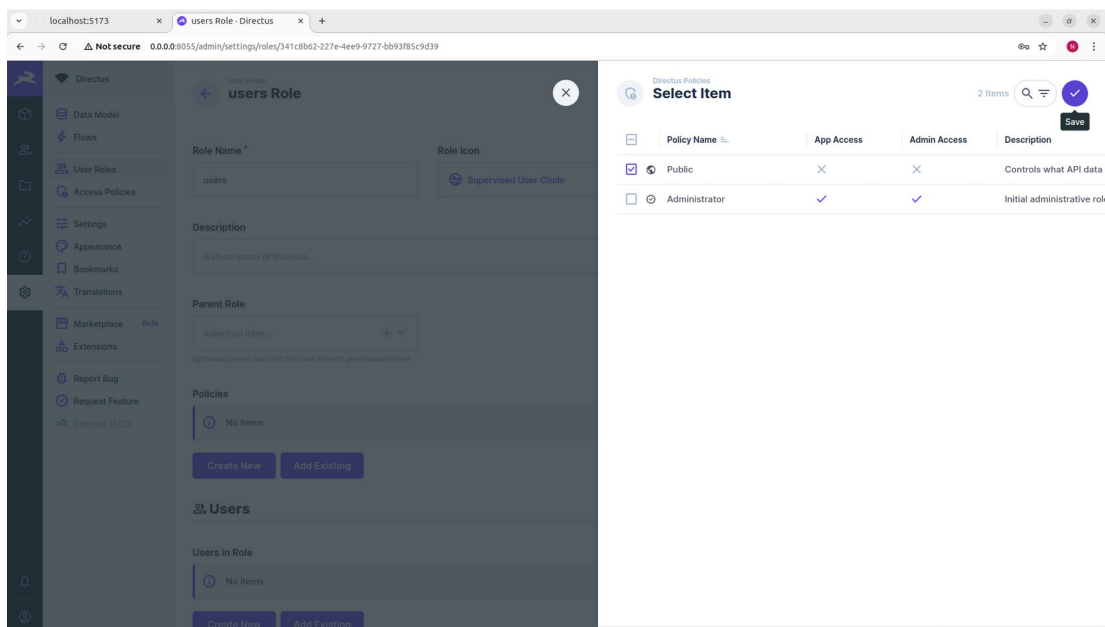
Name	Users	Child Roles	Description
Public	Controls what API data is available without authenticating.
Administrator	1	...	Initial administrative role with unrestricted App/API access.

On the right side, there is an 'Information' panel and an 'Activity Log' at the bottom.

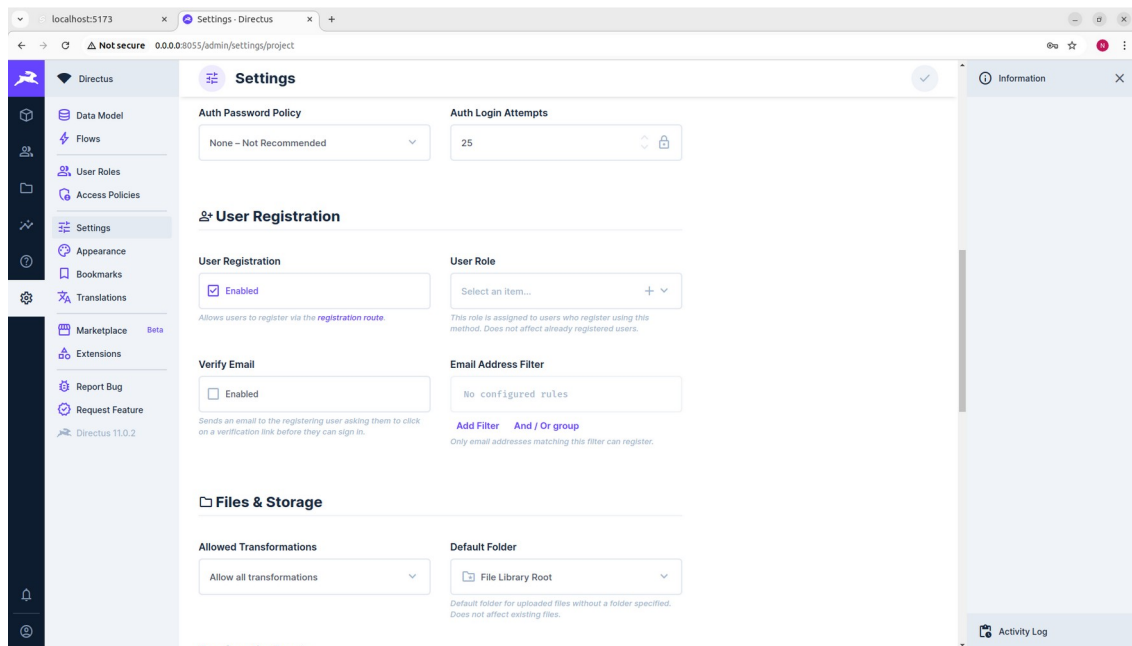
Create a new role by clicking on the ‘+’ icon and name the role users.



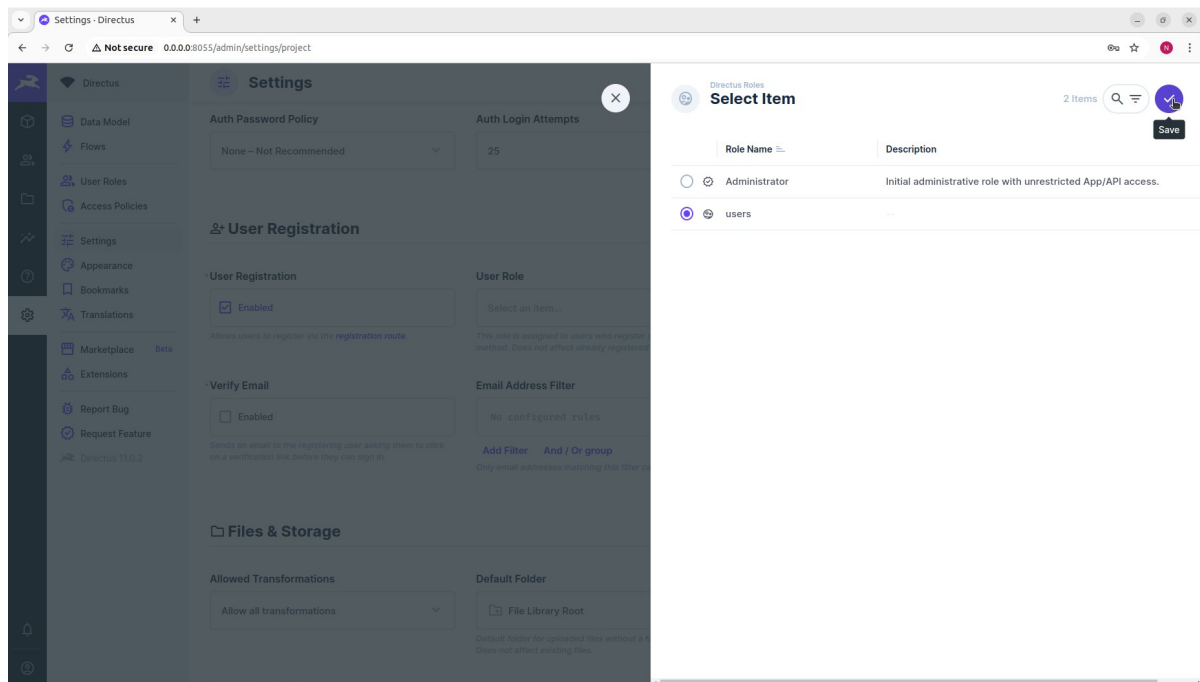
Under the Policies click on ‘Add Existing’ button and then select Public and save it, again save the user role that you have created.



Step-16: Go to settings tab in the setting module, scroll down till the ‘User registration’ section enable the user registration and disable/uncheck the Verify email checkbox



Now, in the user role field click the down arrow and select the users option, then save and then save the user regriction by clicking again on save(tick icon) option



Step-17: That’s it the directus setup and collection creation is done now we can proceed with the sveltekit implementation