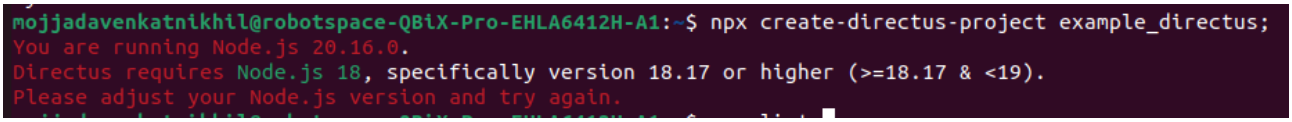
Directus Installation and Configuration with mysql

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

npx create-directus-project <project-name>



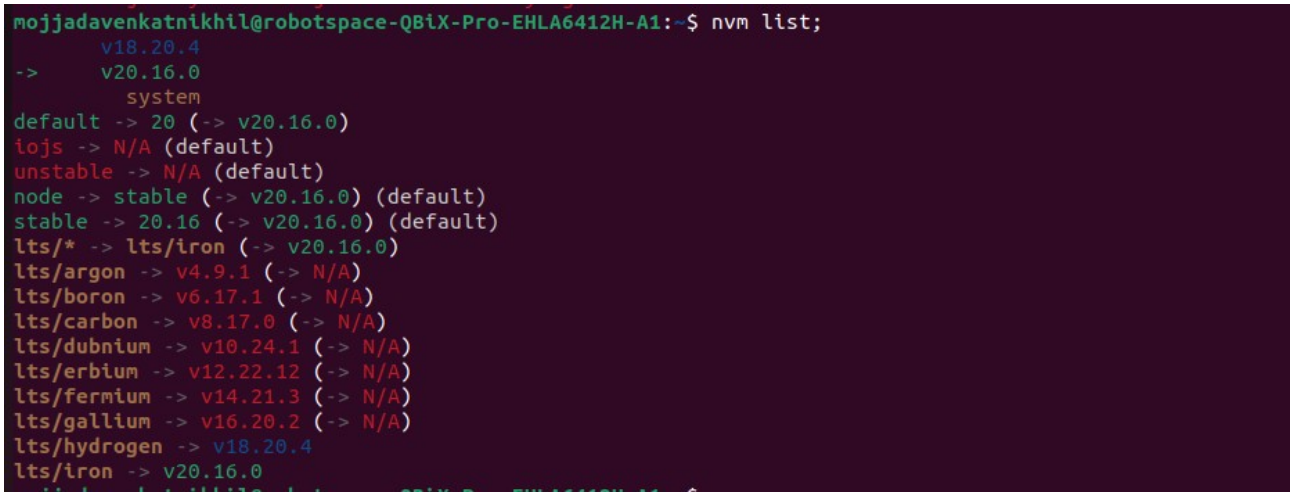
Note: Now if we are using a latest node.js version like 20.16.0 in my example then we will encounter with a problem like directus does not support node.js versions below 18.17 and above 19. So you can install the specified node.js version that is in the given range using

nvm install <version>

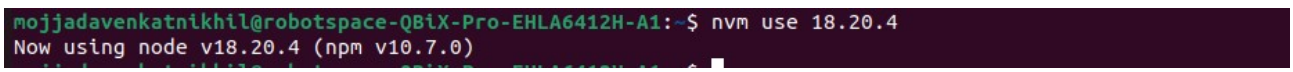
Example: I am installing node.js 18.20.4 version

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



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



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.

```
mojjadavenkatnikhil@robotspace-QBiX-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 <pr
oject-name>
```

Example:

```
mojjadavenkatnikhil@robotspace-QBiX-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_ne
w_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 installion 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:

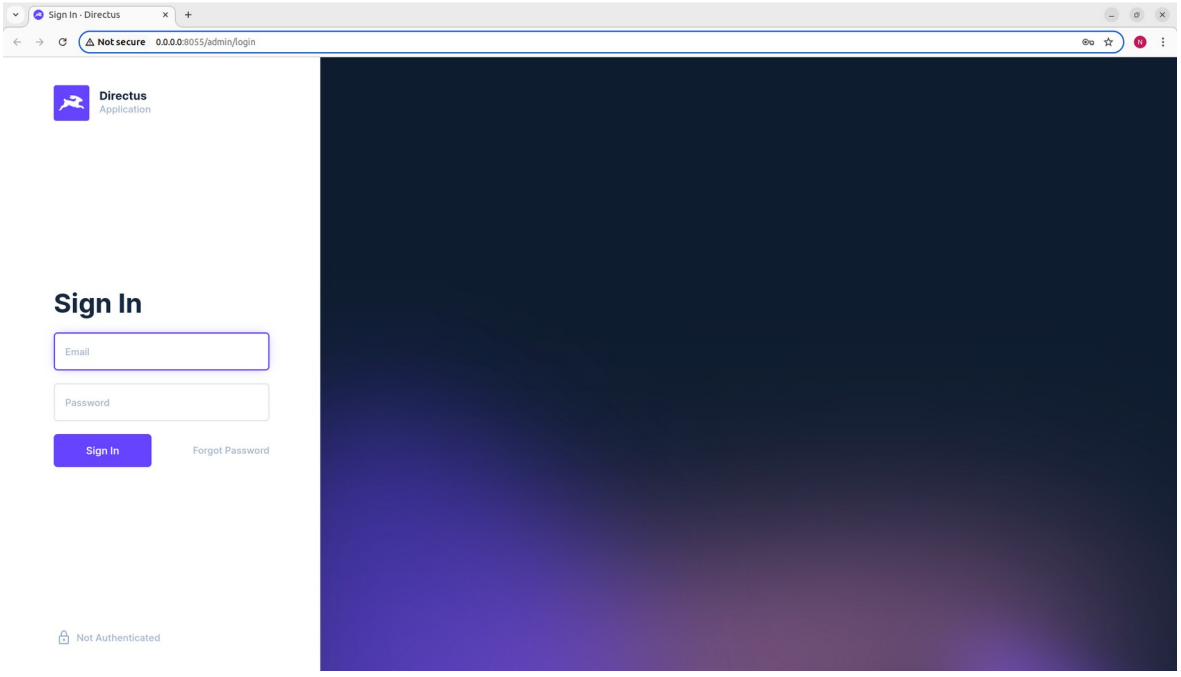
```
$ cd directus_demo
```

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 .

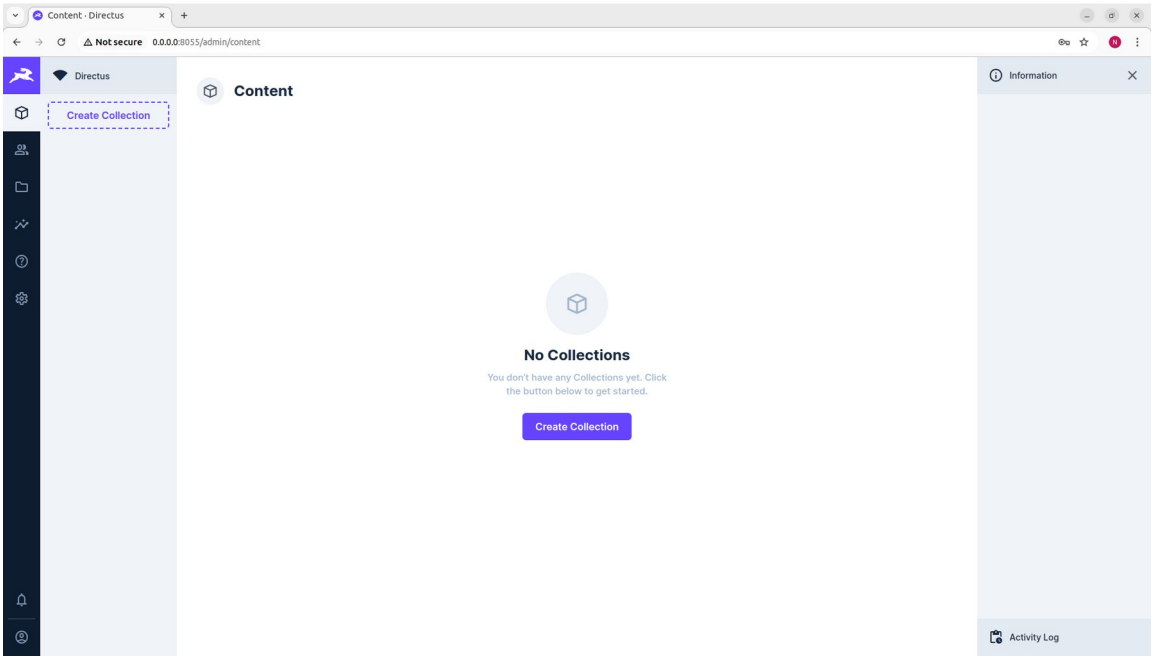


Step-9: Enter the email and password that you have given while installation

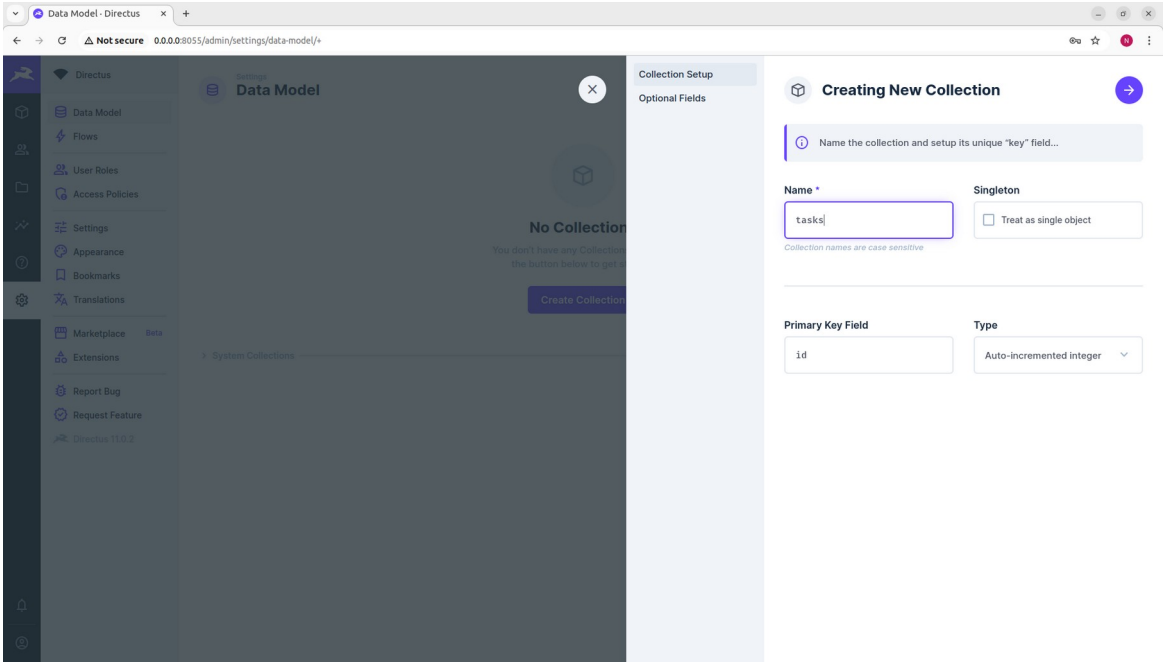
Example:

Email: admin@example.com // your email
Password: password // your password

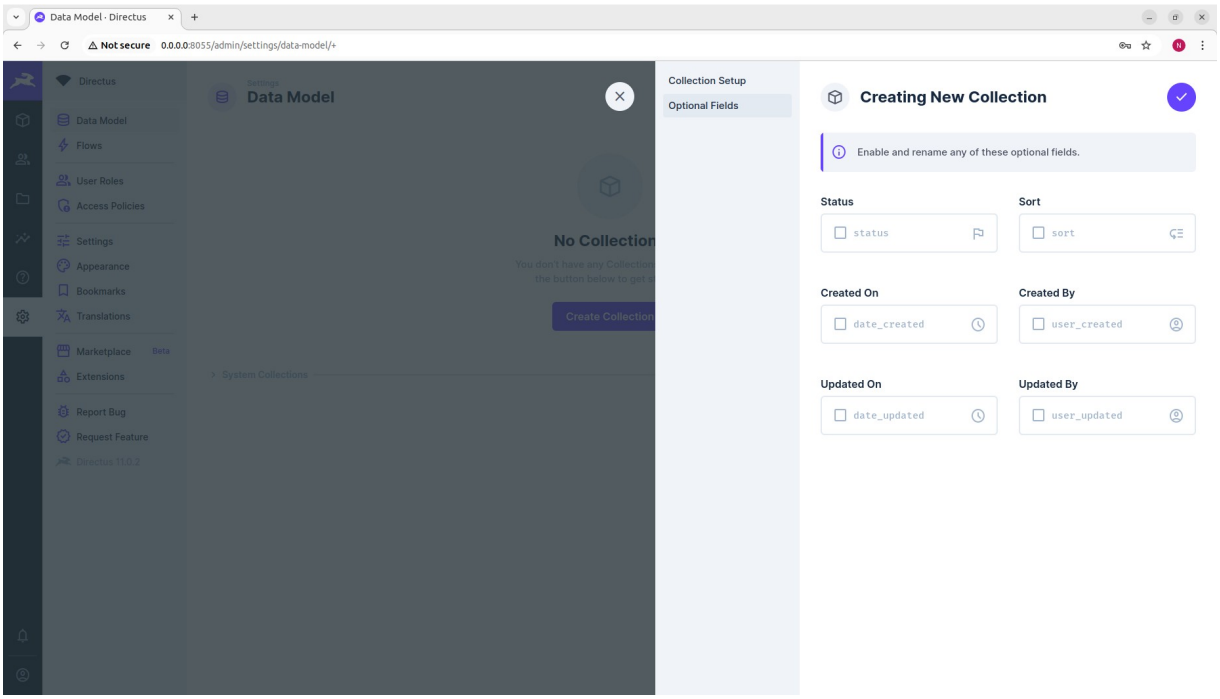
Step-10 Creating a collection: After you successfully login your screen will look like this



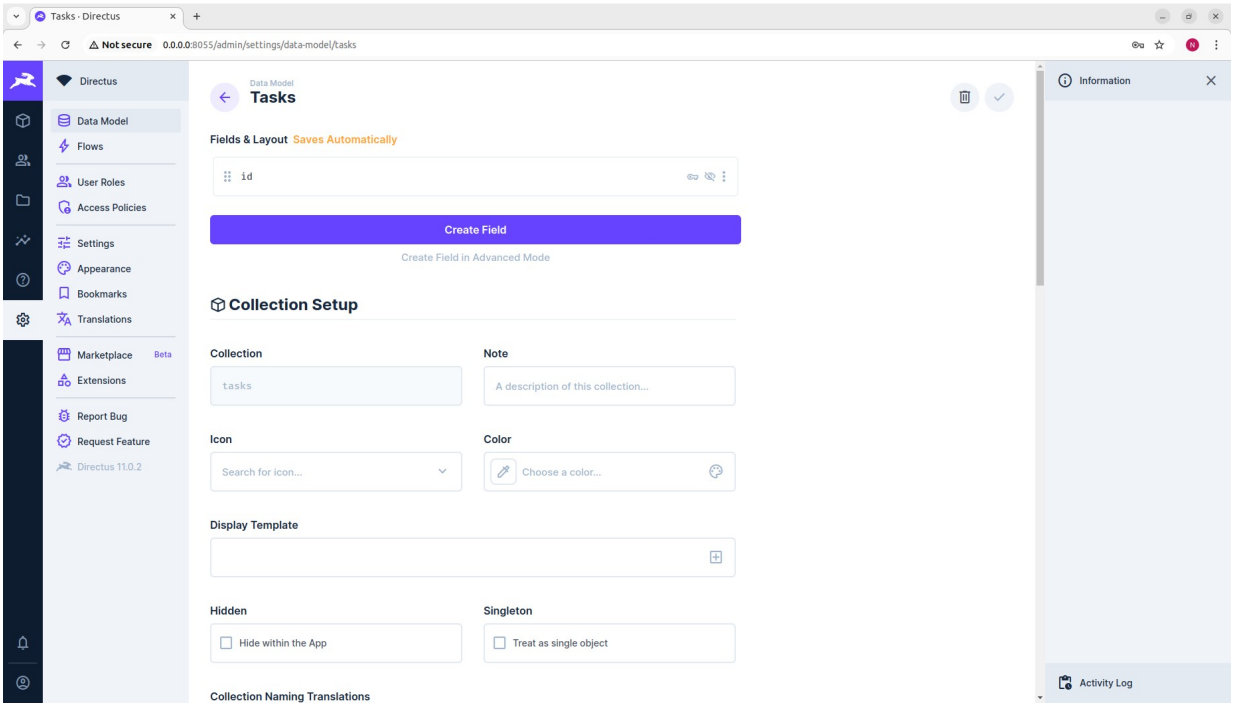
Here click on ‘create collection’ button, give the collection a name like ‘ tasks’,click the arrow to proceed further.



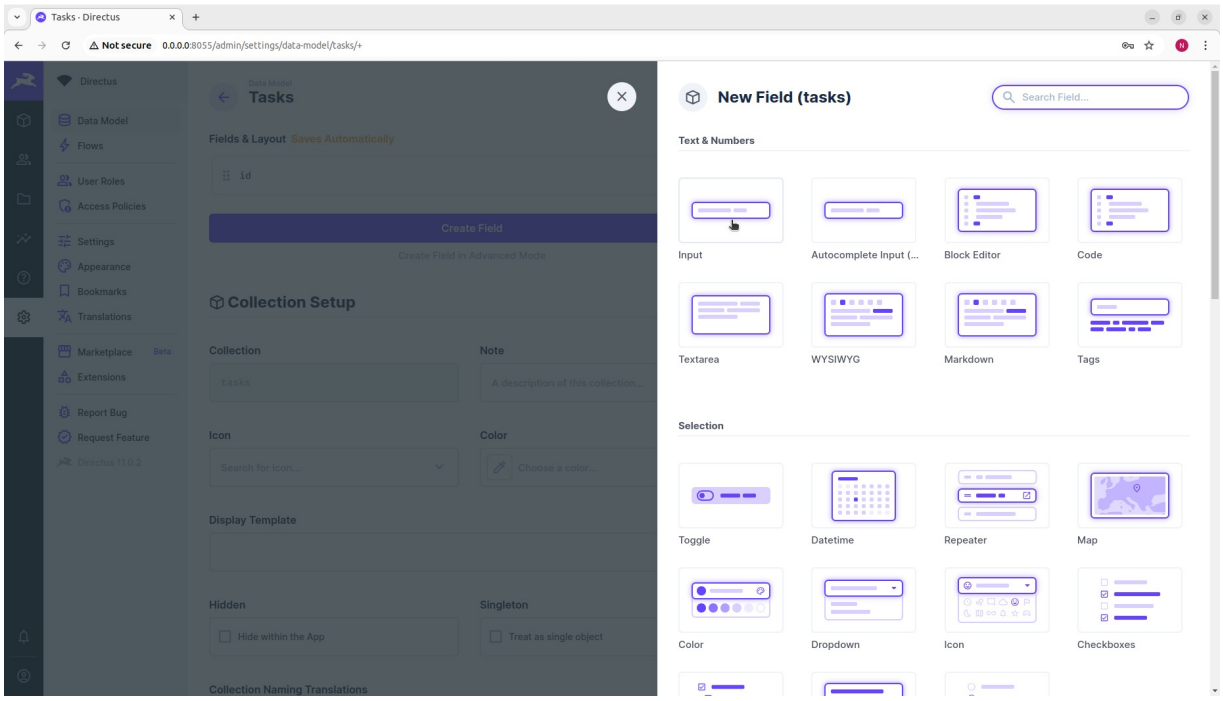
No need to select any option over here and click the tick button, you have created a collection successfully.



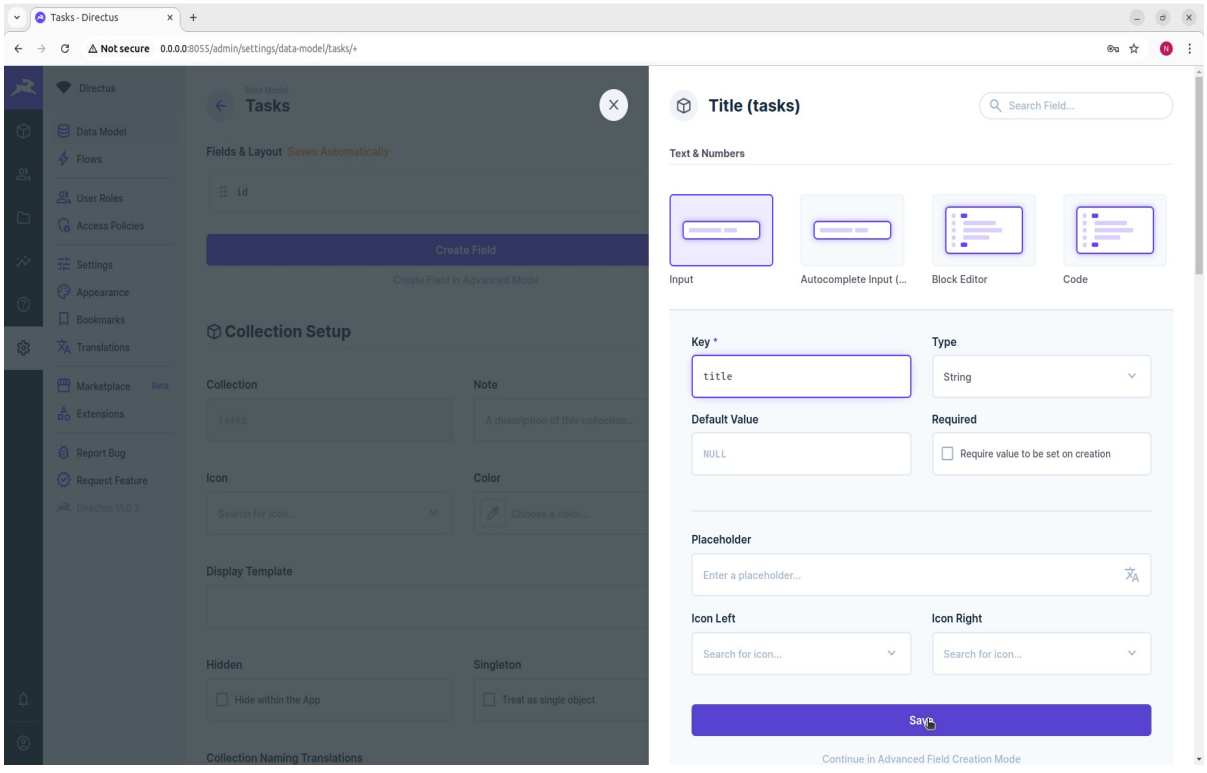
Step-11 Creating Fields: Here click on ‘Create Field’ button to create the task fields that are required for our task management app.



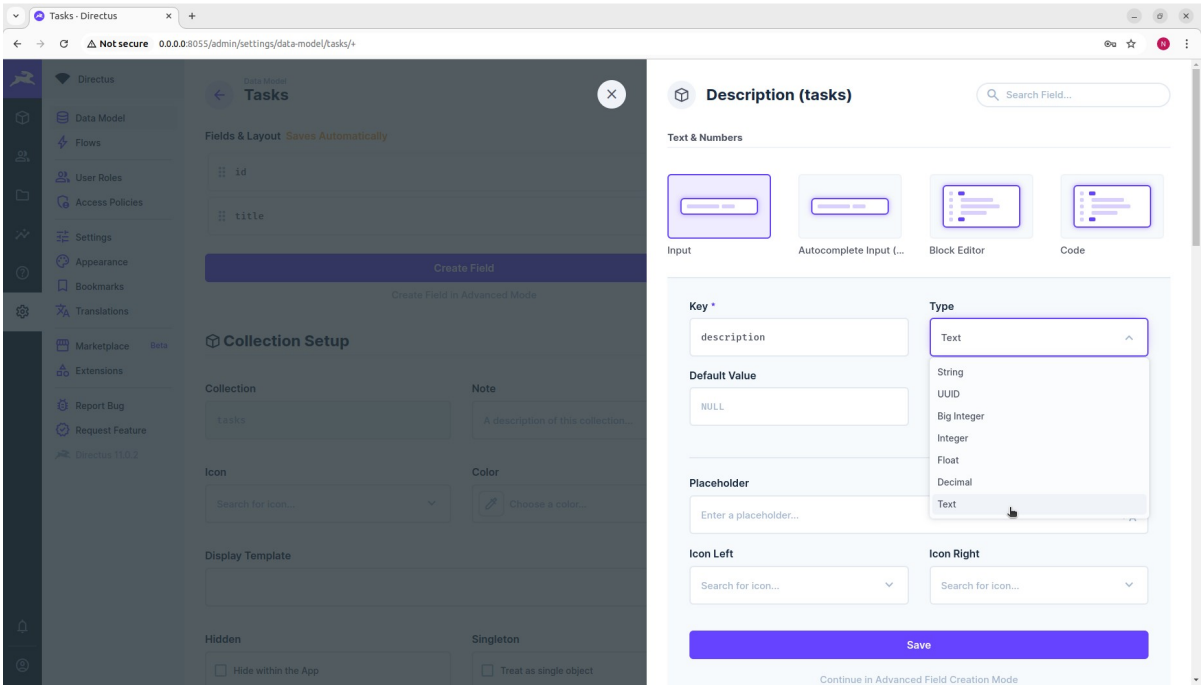
Click on the ‘input’ card.



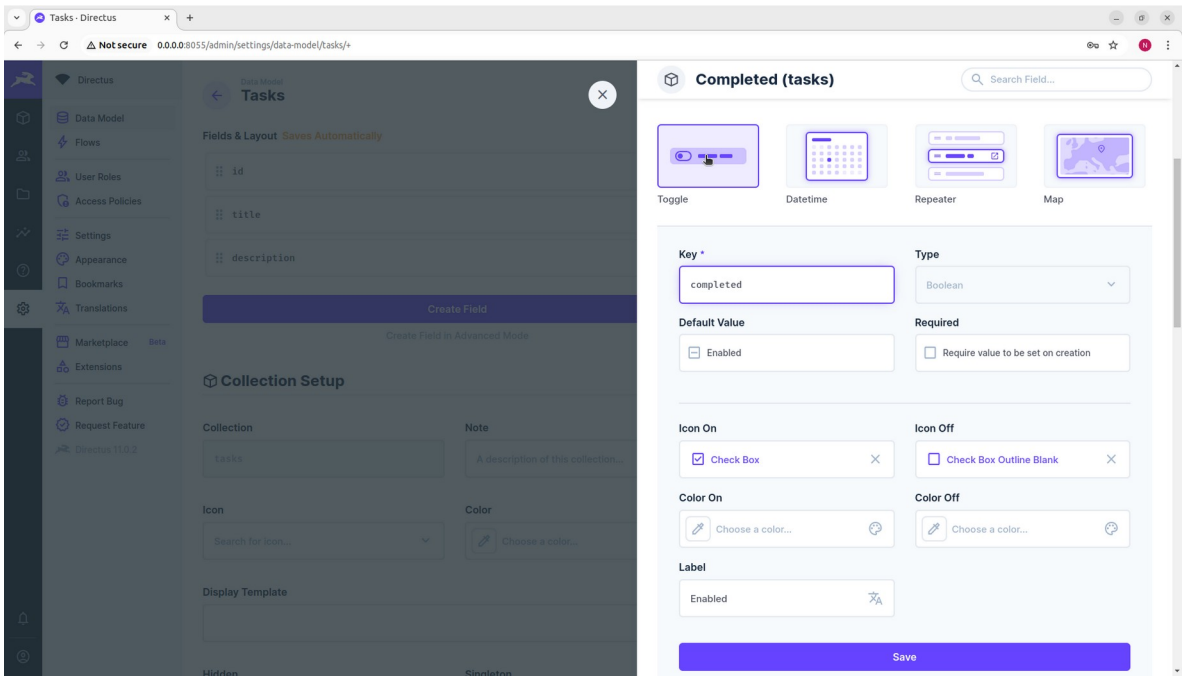
Now lets give it a name ‘title’ and click on save.



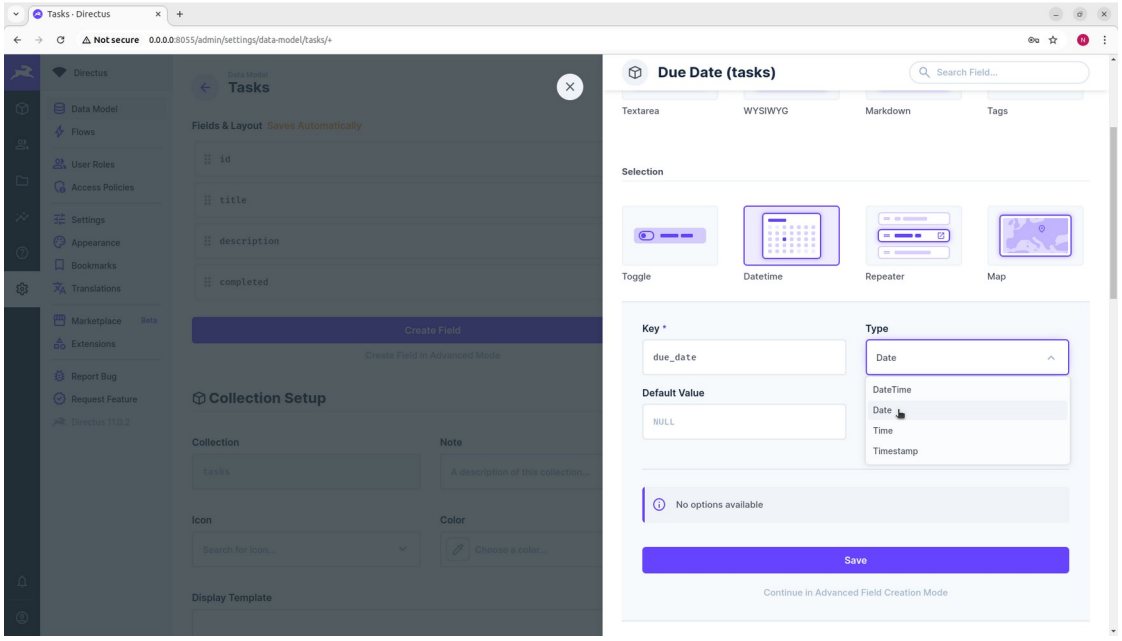
Lets create another field named ‘description’ and set the type as Text, then click save.



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.

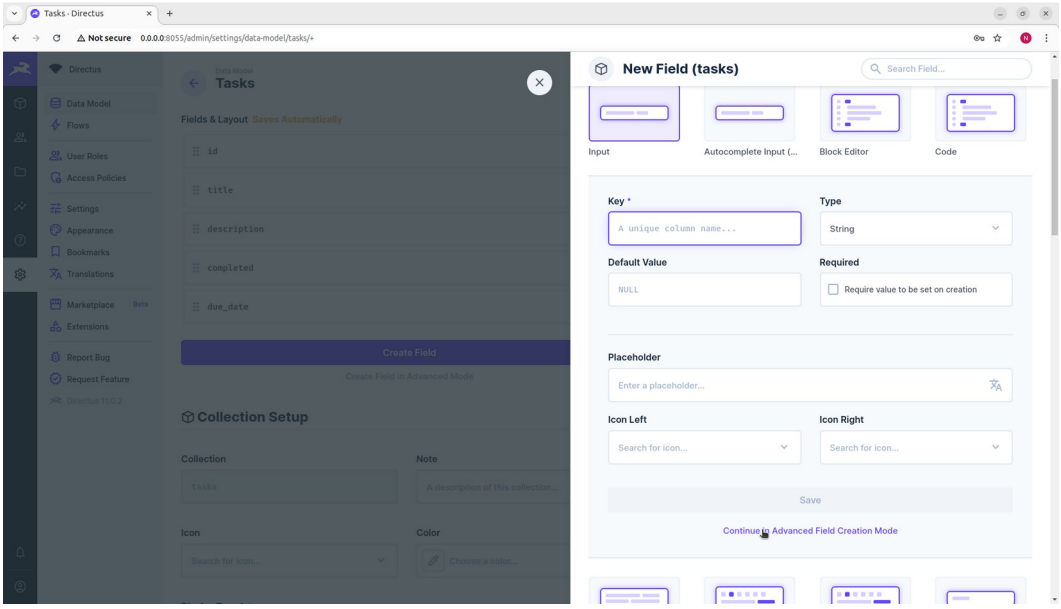


Create another field, click on ‘Datetime’ card under Selection category name it ‘due_date’ , select type as Date and then click save.

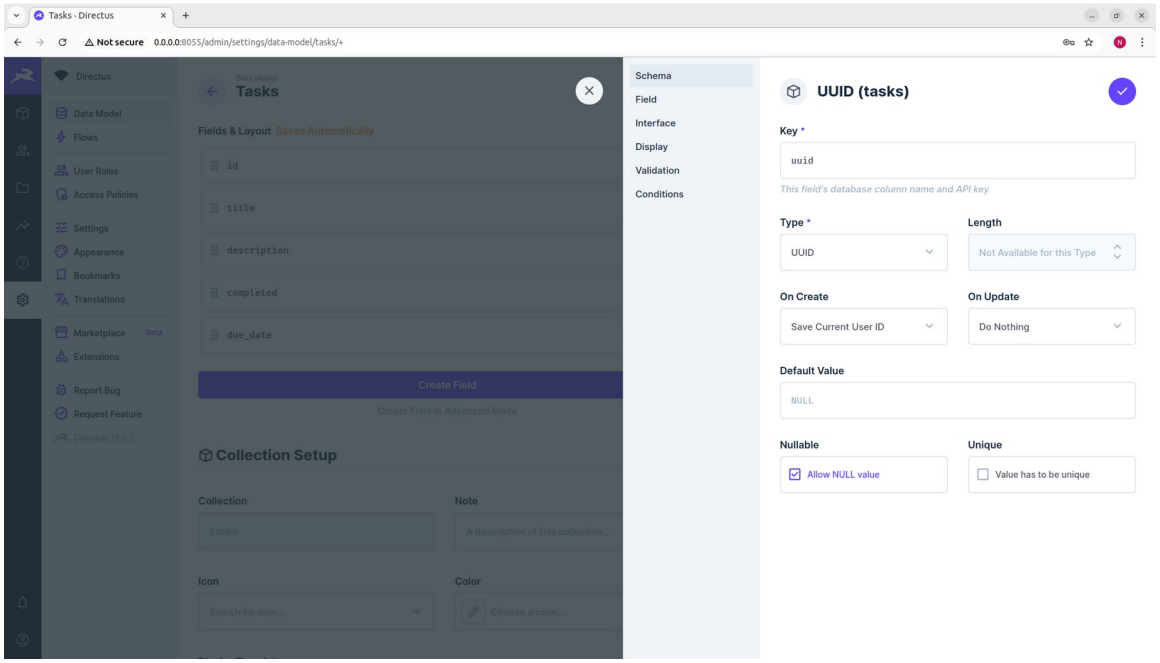


Now lets create the last field which is important in our collection, click on ‘input’ card and then click on Continue in Advanced Field Creation Mode.

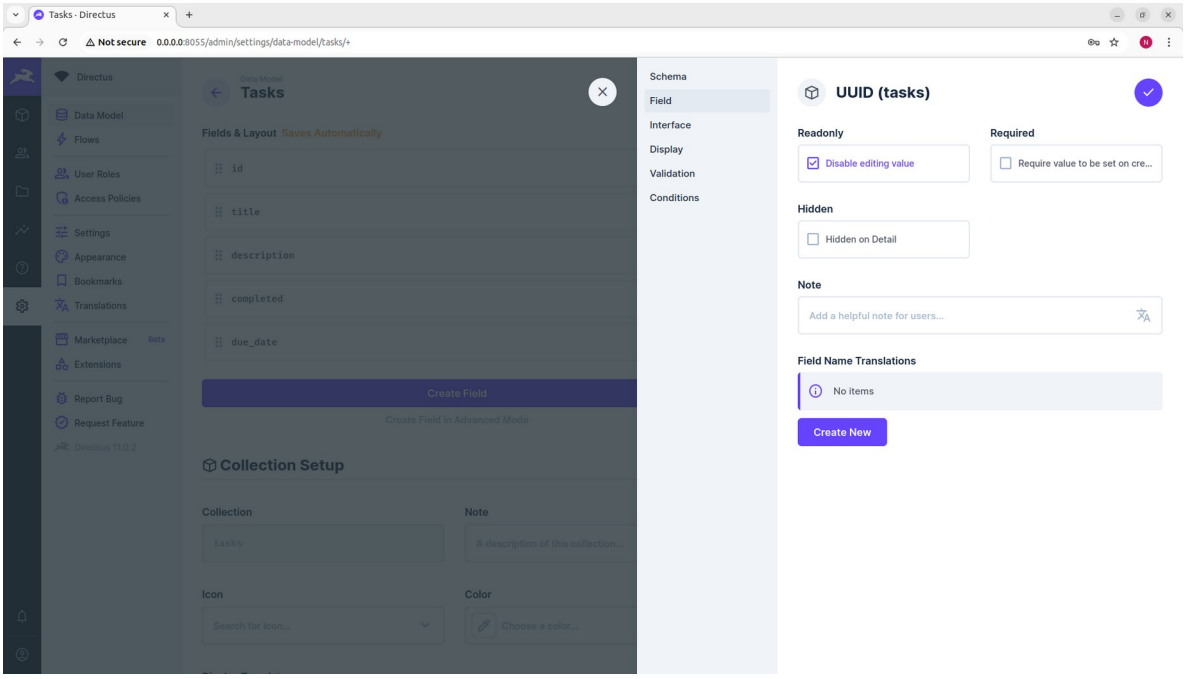
Note: The purpose for creating the uuid field is so that we can get the records of the users that have created the tasks specifically by authentication, by extracting uuid we will show the users their respective tasks instead of showing every users tasks.



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

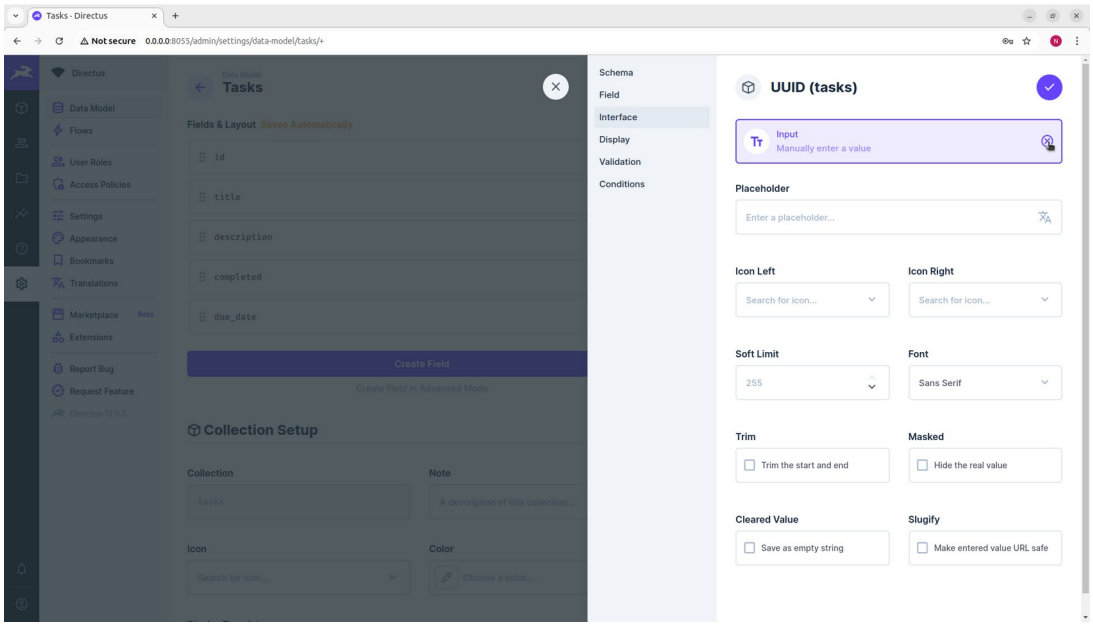


In the ‘Field’ tab check the ‘Disable editing value’ box, next click on ‘Interface’ tab.

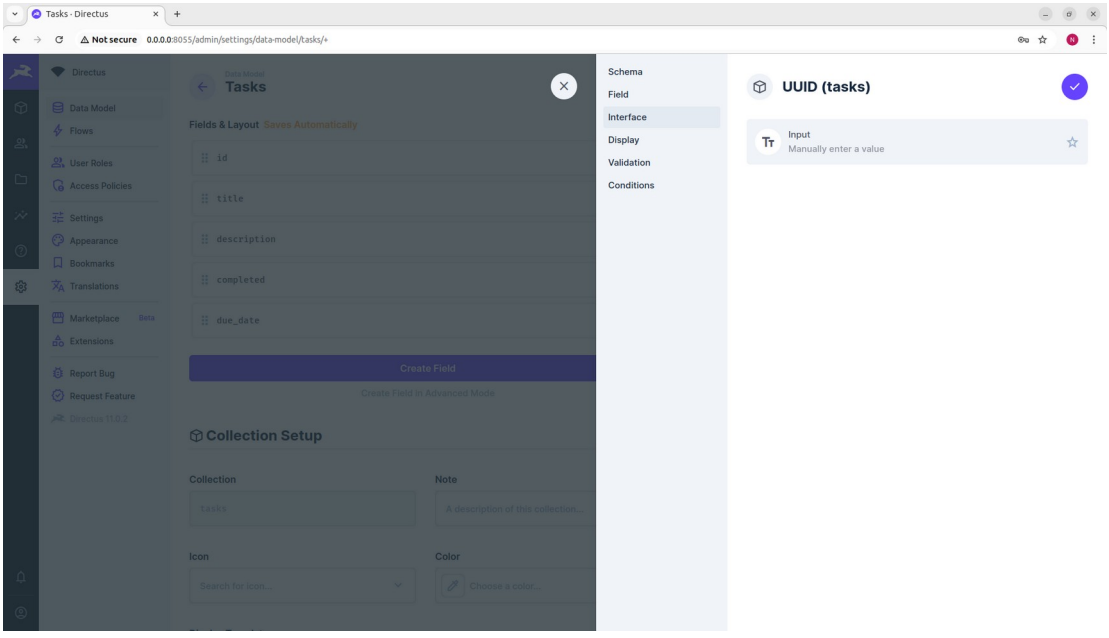


In the ‘Interface’ tab, unselect the ‘Input’ tab in the top or you can also click the ‘x’ button in the ‘Input’ tab.

Before:

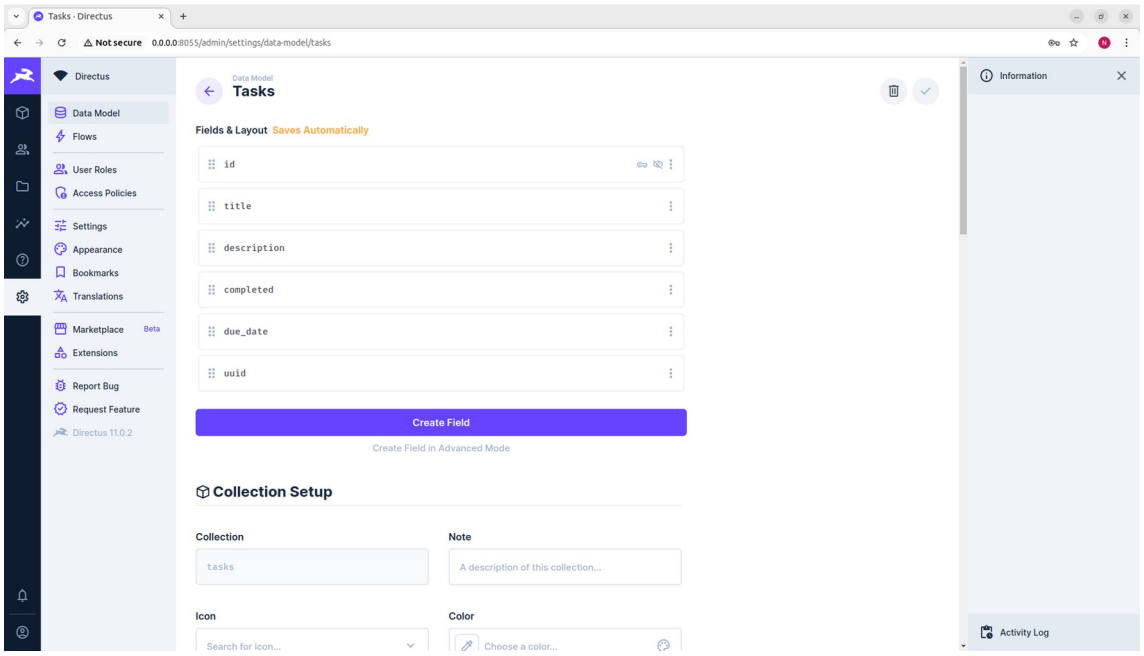


After:

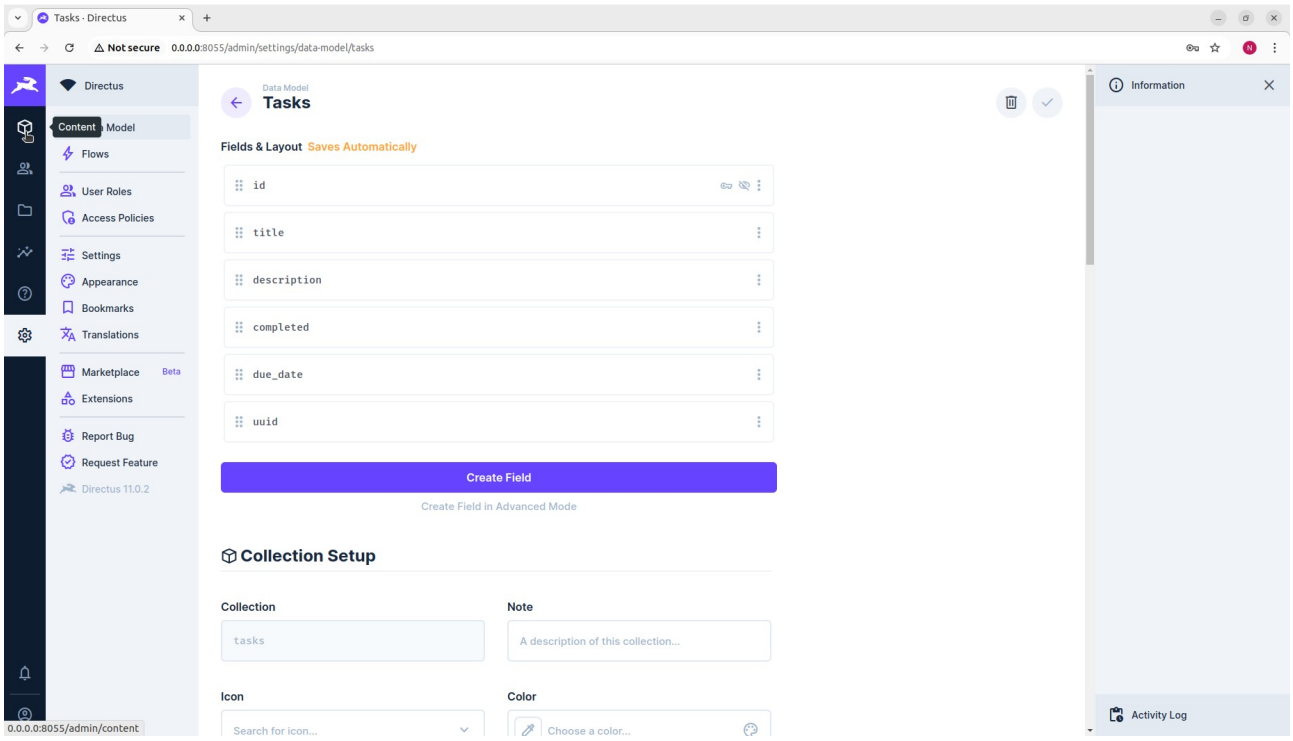


Now click on the ‘tick’ mark icon to save it.

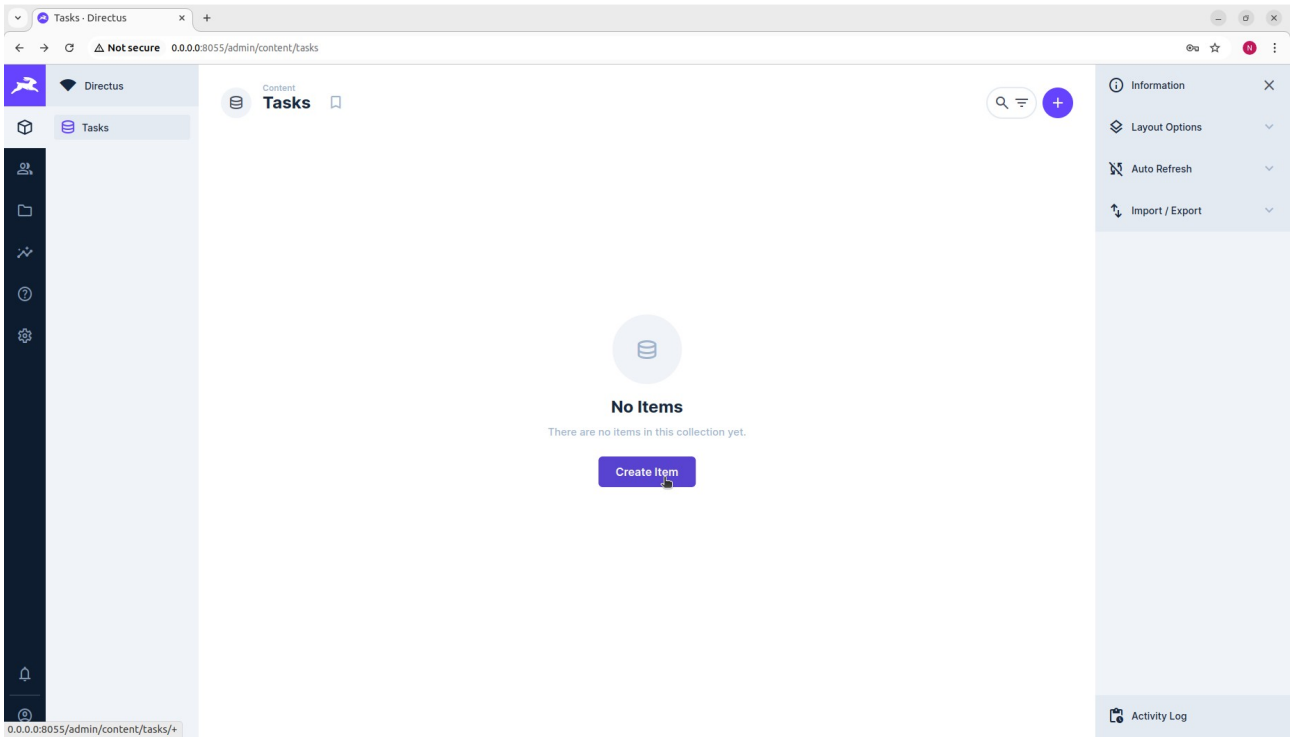
Your collection will look like this if everything is done as said.



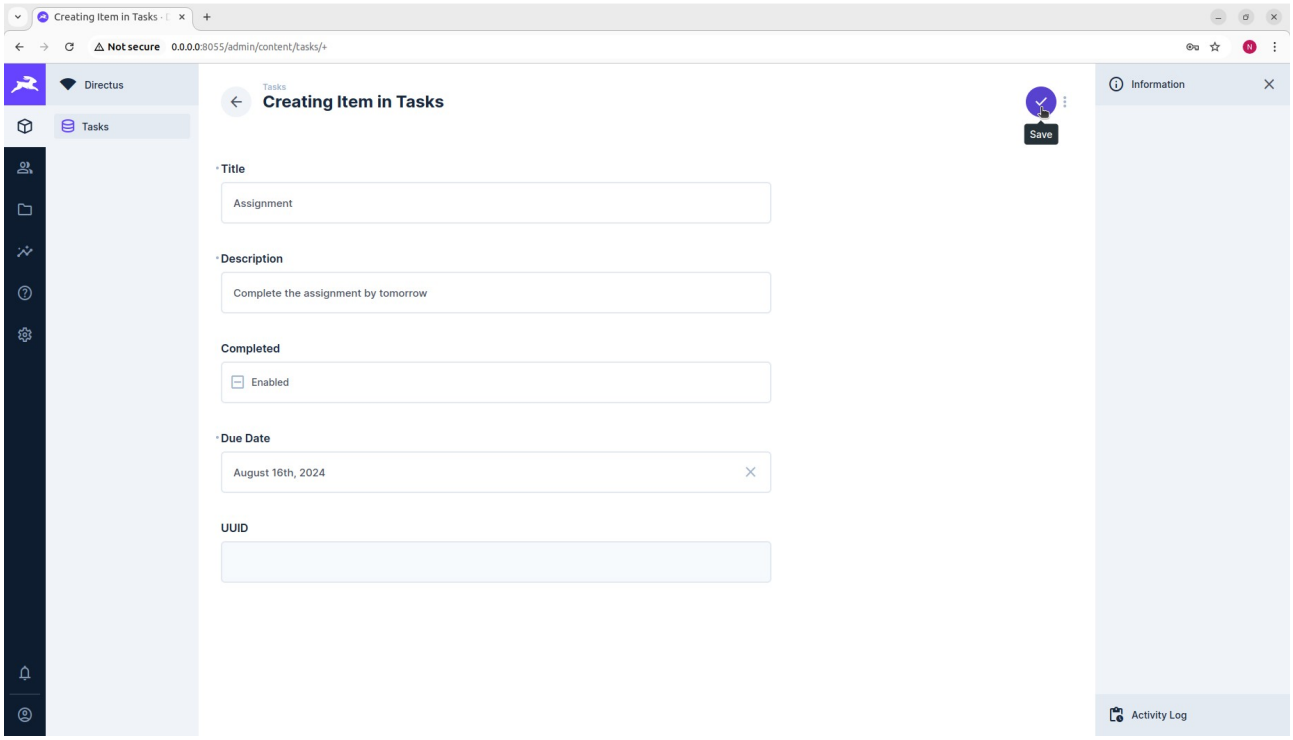
Step-12: Click on the ‘Content’ module to see the structure of the ‘tasks’ collection.



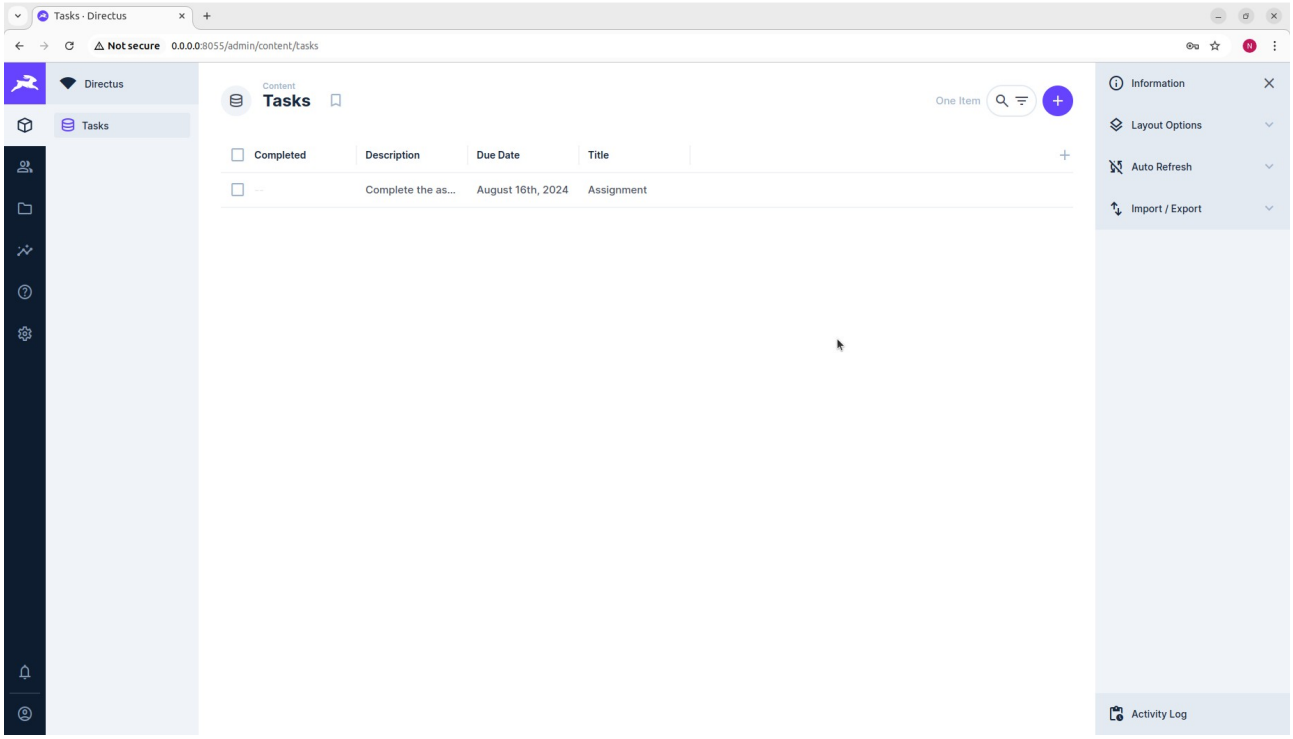
Now Click on ‘Create Item’ button.



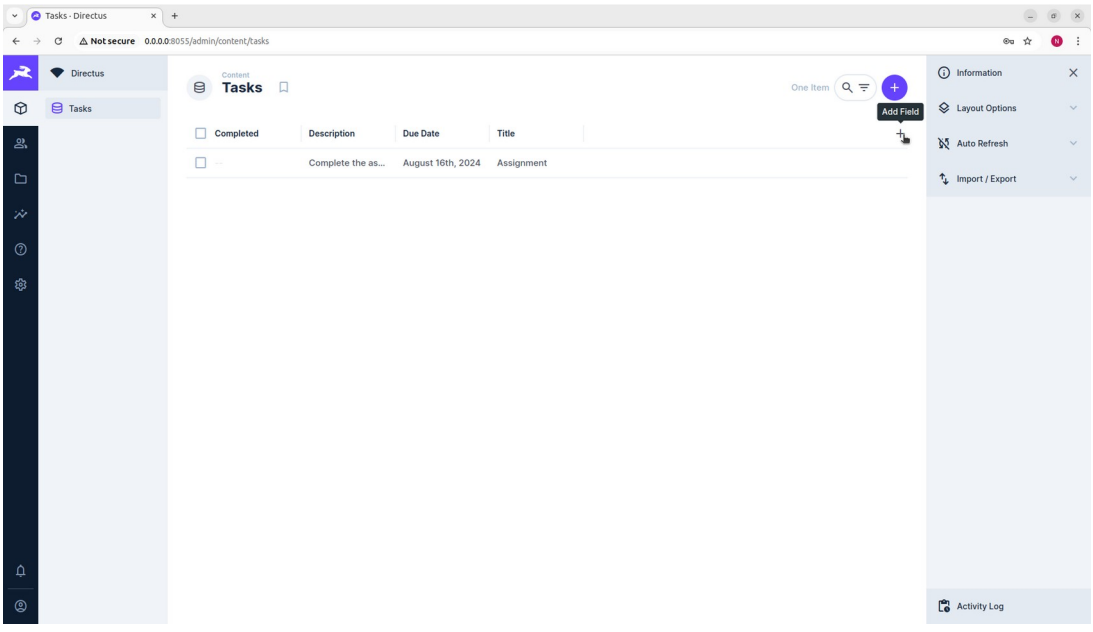
Now give some custom values in each field for testing and save it.



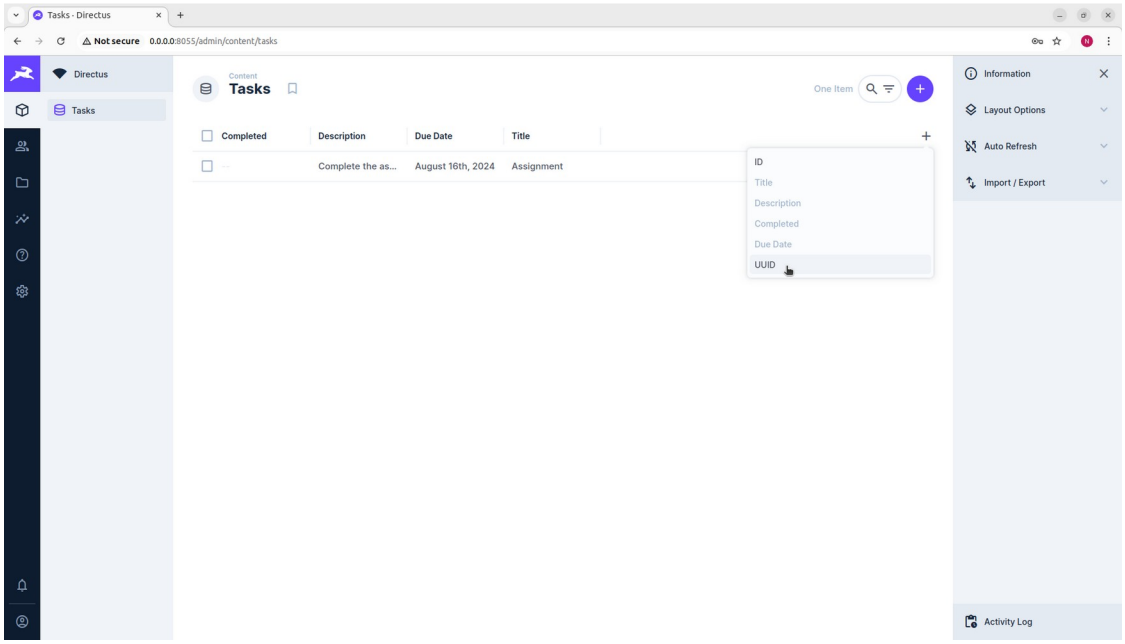
Our table should look like this.



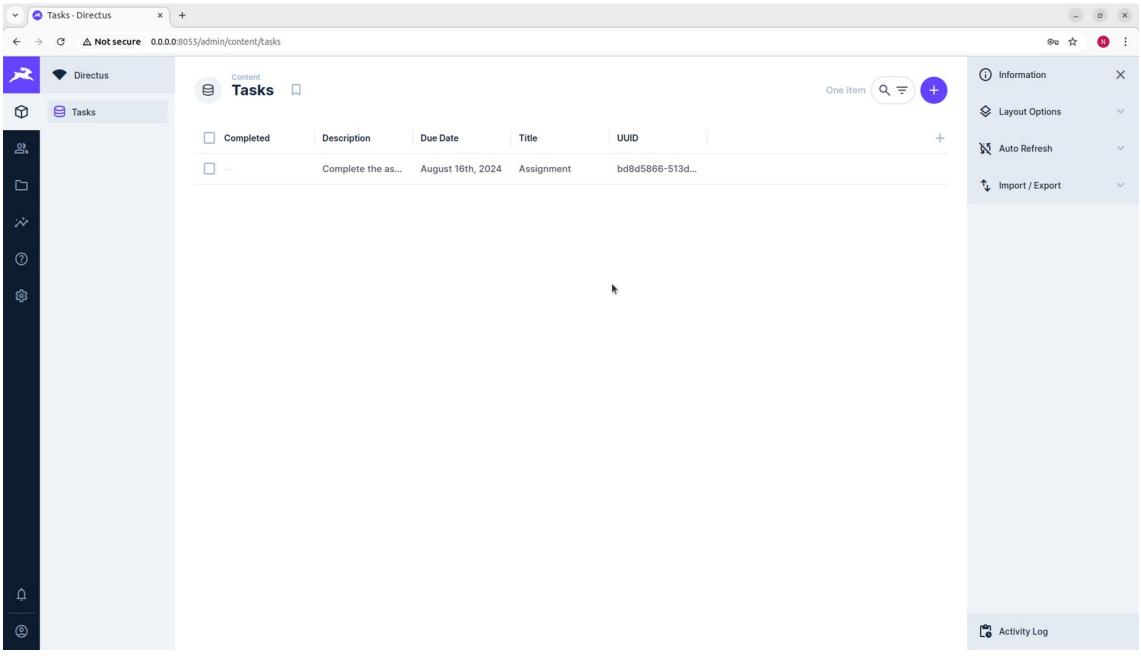
Here we cannot see the field ‘uuid’ that we have created, click on ‘+’ icon near the table Headers field



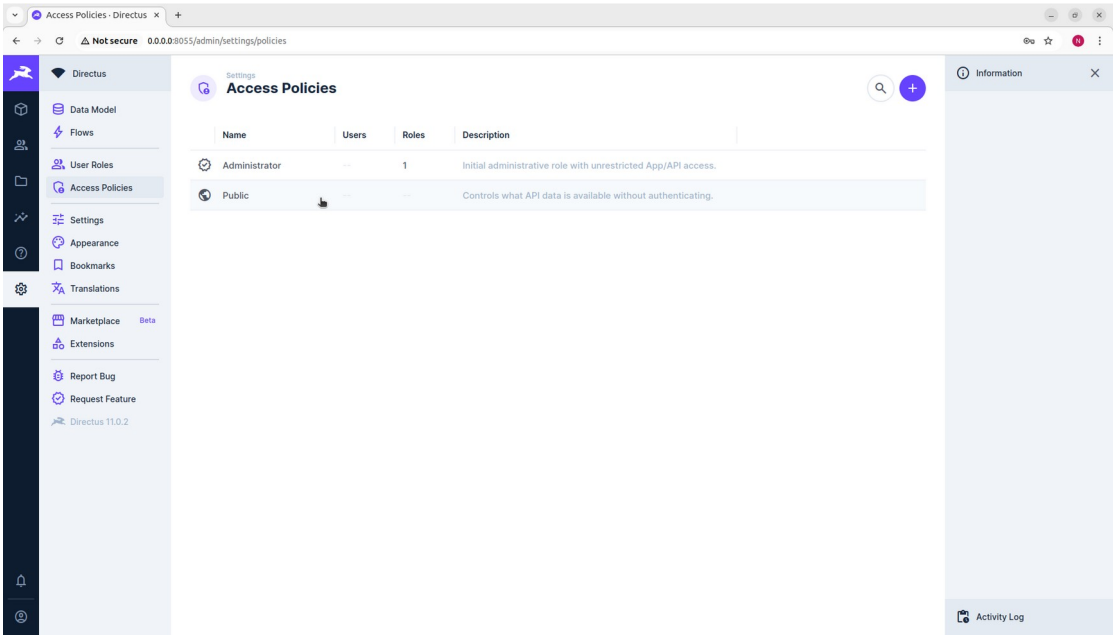
Now select the UUID option.



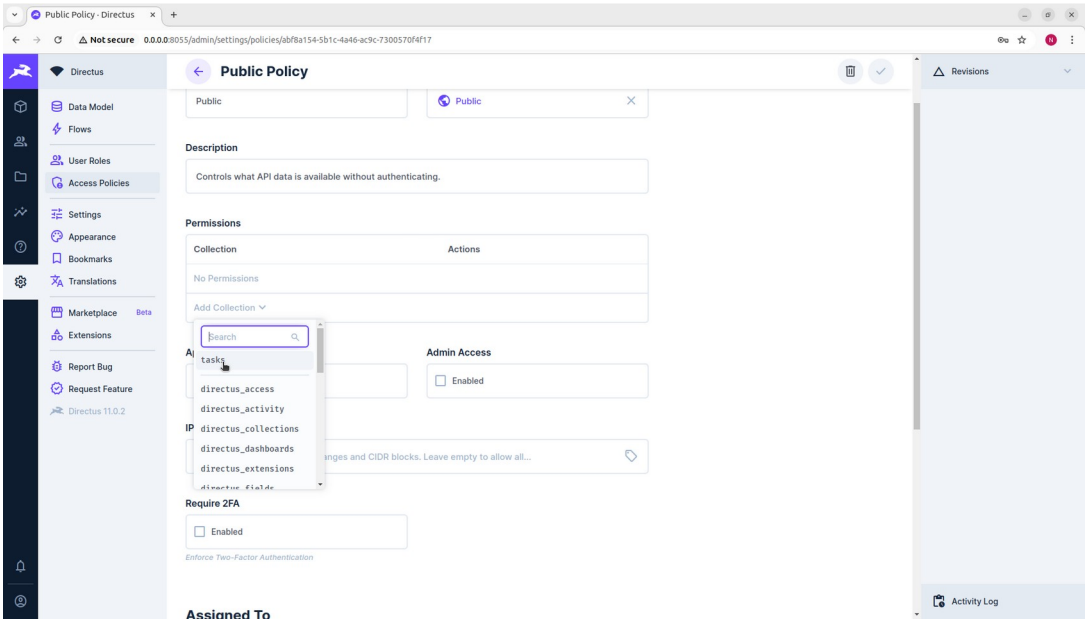
Now you can see it in your table.



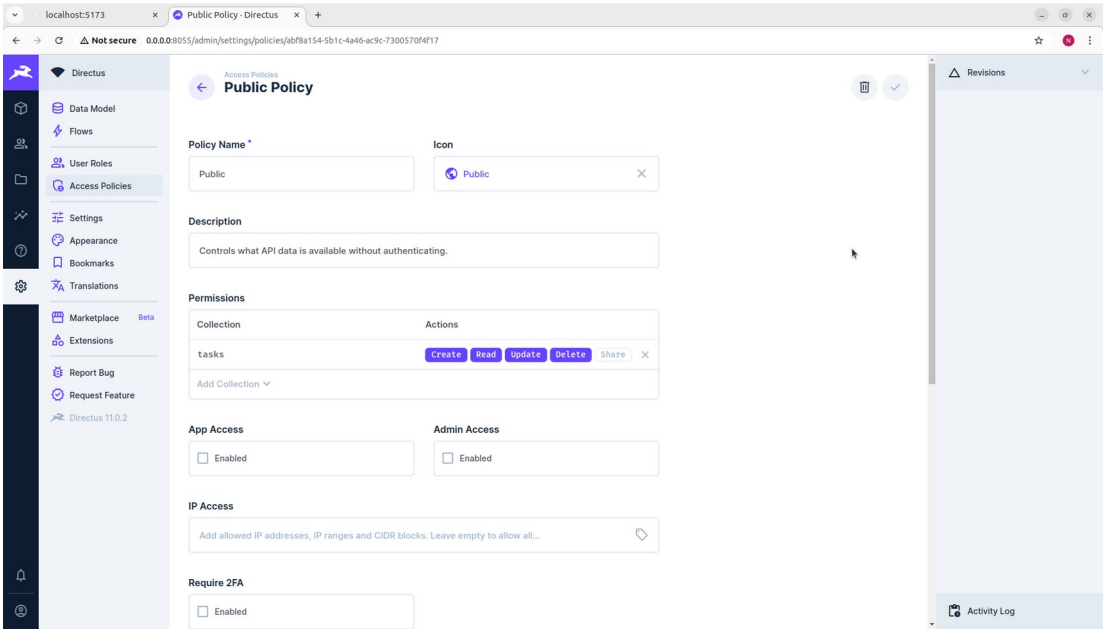
Step-13: Go to settings module click on access policies, then click on public



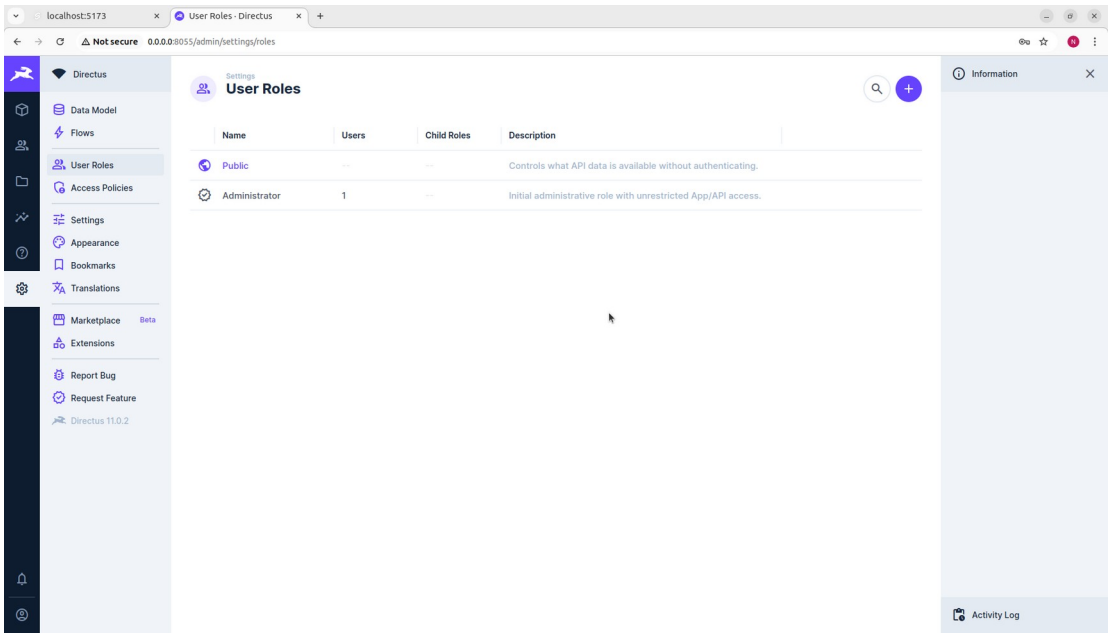
Step-14: In public policy add the tasks collection



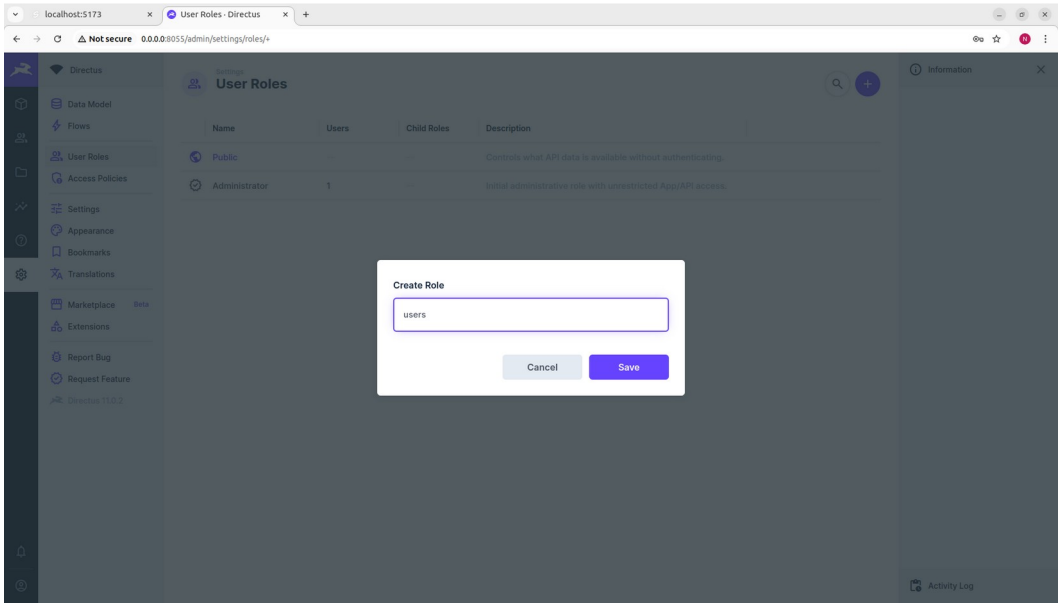
Enable the Read,Create,Delete and Update options



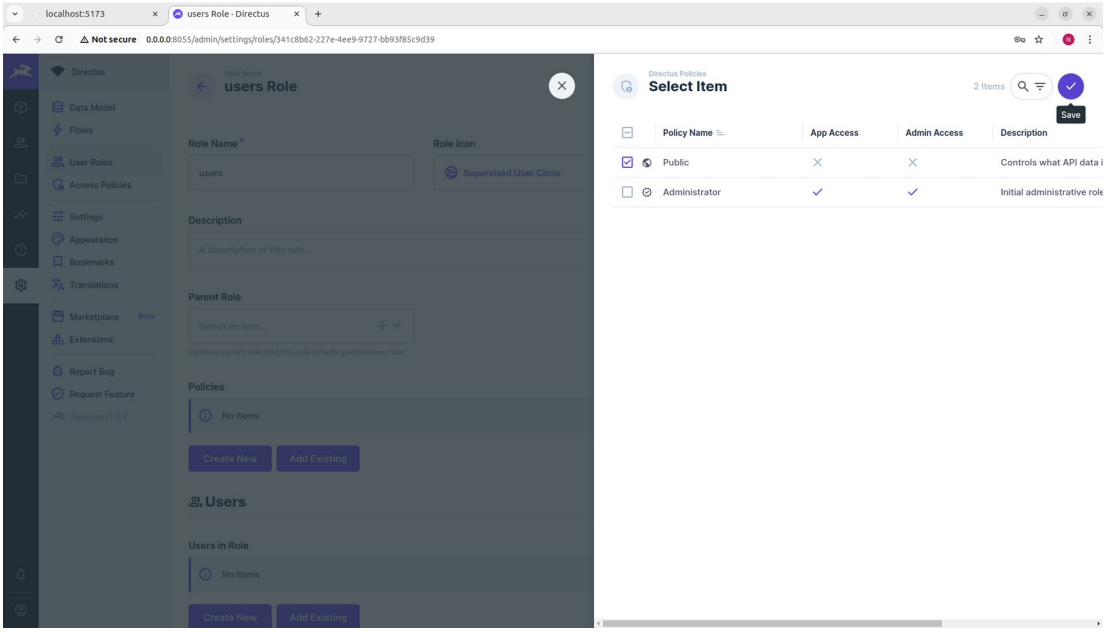
Step-15: Go to user roles above the access policies option



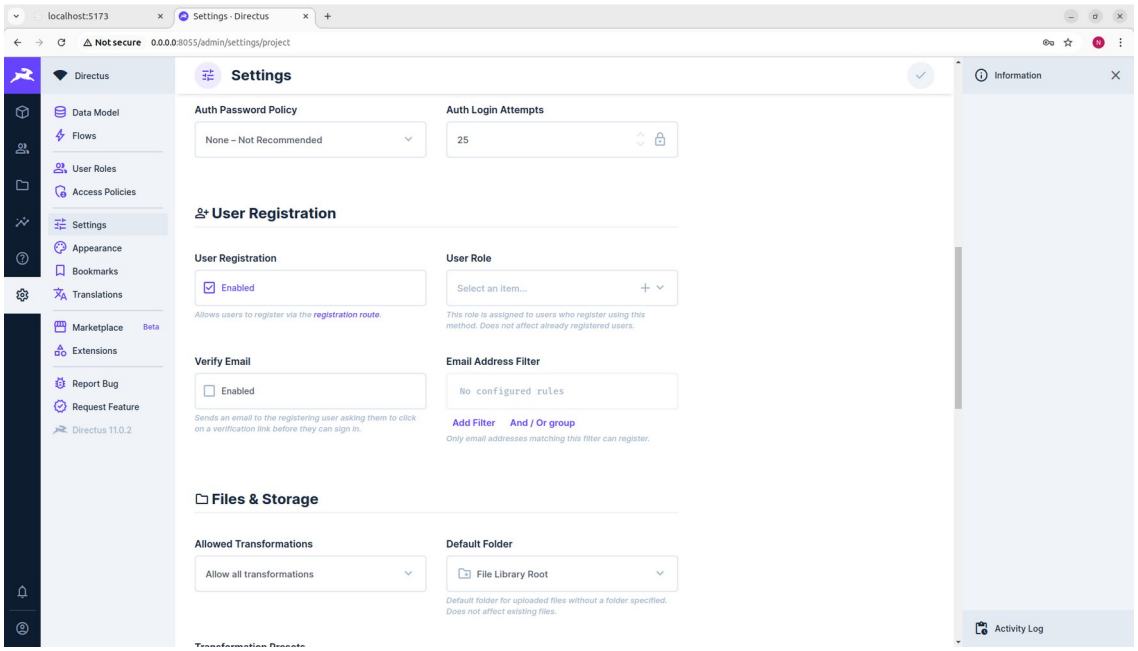
Create a new role by clicking on the ‘+’ icon ans name the role users.



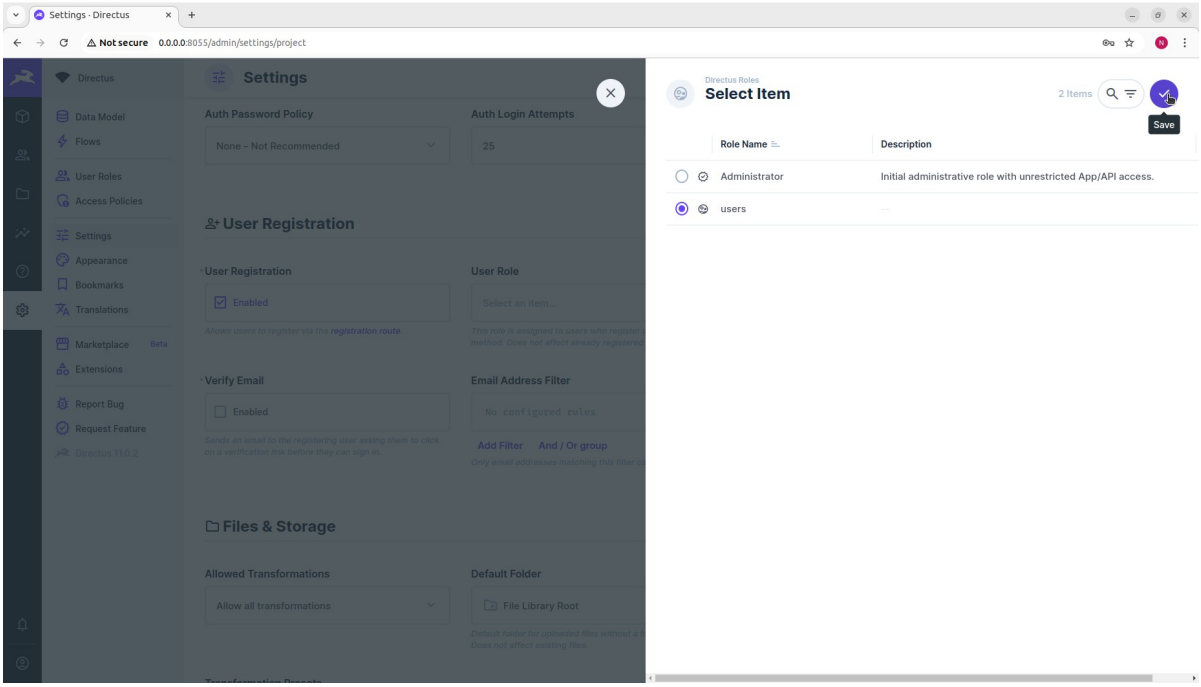
Under the Policies click on ‘Add Exisiting’ button and then select Public and save it, again save the user role that you have created.



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