

## 2021 Database Documentation

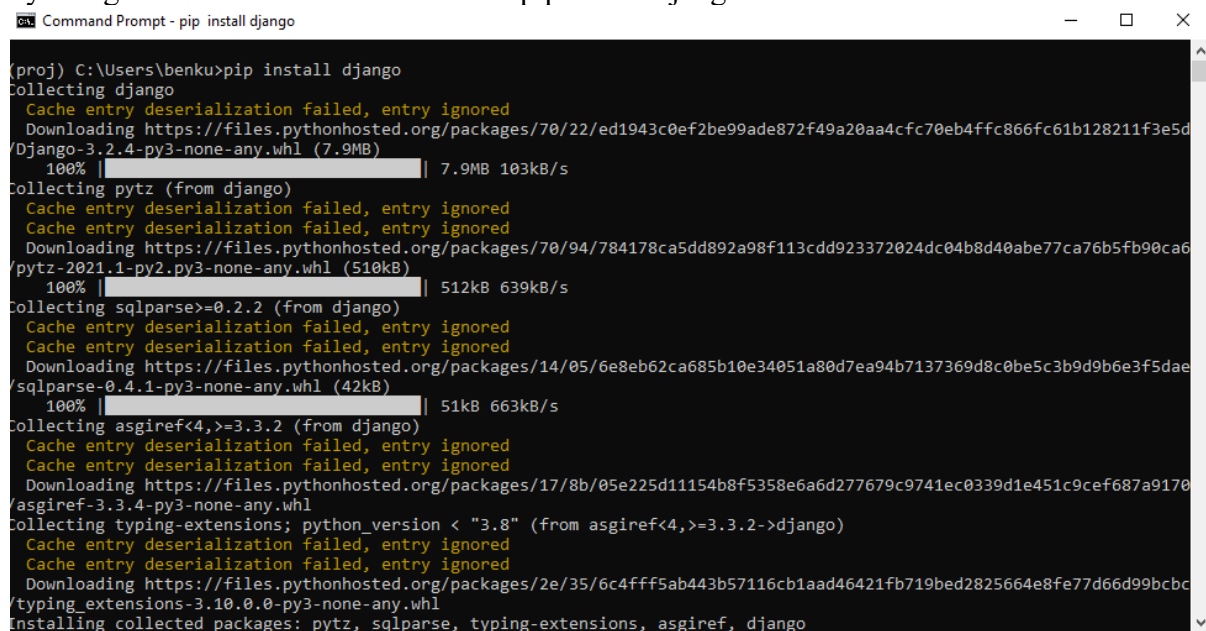
<b>Project Title</b>	<b>English</b>	<b>Library-blog website</b>
--------------------------	----------------	-----------------------------

# Overview

The implementation of my program divided into two categories. First categories, where the back-end I made in Django by using ORM and second I translates my ORM implementation in SQL tables and insertion I used for a person who is aware of SQL, but have not idea about Django. During this course I understood a lot of materials and understood that using knowledge from database and sql, I can try other database and that is why I used sqlite3, because it is simple and not for big projects. I made a blog-library where members can create, read, update, delete the discussion about a book and other can comment, like and view a book, but they need to login first. Most of my logic is made in python such as creating tables, making queries. Inserting new records in database is done by using Django framework which has an ORM, which interacts with database by using python script. Currently the website works from my PC with the address 127.0.0.1 is the standard address for IPv4. I have two sql files, they are for the viewer who is trying to understand what kinds of tables I added to my database and insertion.sql is for values that I added to my database.

## Run the program

First of all, to run the website, we need to install the interpreter of python 3.6 in the official website <https://python.org/>, then we need to install the framework that can be easily installed by using CMD. And run the command 'pip install django'



```
Command Prompt - pip install django

(proj) C:\Users\benku>pip install django
Collecting django
  Cache entry deserialization failed, entry ignored
  Downloading https://files.pythonhosted.org/packages/70/22/ed1943c0ef2be99ade872f49a20aa4cfc70eb4ffc866fc61b128211f3e5d/Django-3.2.4-py3-none-any.whl (7.9MB)
    100% |#####| 7.9MB 103kB/s
Collecting pytz (from django)
  Cache entry deserialization failed, entry ignored
  Cache entry deserialization failed, entry ignored
  Downloading https://files.pythonhosted.org/packages/70/94/784178ca5dd892a98f113cdd923372024dc04b8d40abe77ca76b5fb90ca6/pytz-2021.1-py2.py3-none-any.whl (510kB)
    100% |#####| 512kB 639kB/s
Collecting sqlparse>=0.2.2 (from django)
  Cache entry deserialization failed, entry ignored
  Cache entry deserialization failed, entry ignored
  Downloading https://files.pythonhosted.org/packages/14/05/6e8eb62ca685b10e34051a80d7ea94b7137369d8c0be5c3b9d9b6e3f5dae/sqlparse-0.4.1-py3-none-any.whl (42kB)
    100% |#####| 51kB 663kB/s
Collecting asgiref<4,>=3.3.2 (from django)
  Cache entry deserialization failed, entry ignored
  Cache entry deserialization failed, entry ignored
  Downloading https://files.pythonhosted.org/packages/17/8b/05e225d11154b8f5358e6a6d277679c9741ec0339d1e451c9cef687a9170/asgiref-3.3.4-py3-none-any.whl
Collecting typing-extensions; python_version < "3.8" (from asgiref<4,>=3.3.2->django)
  Cache entry deserialization failed, entry ignored
  Cache entry deserialization failed, entry ignored
  Downloading https://files.pythonhosted.org/packages/2e/35/6c4fff5ab443b57116cb1aad46421fb719bed2825664e8fe77d66d99bcbcd/typing_extensions-3.10.0-py3-none-any.whl
Installing collected packages: pytz, sqlparse, typing-extensions, asgiref, django
```

next, we need to get into folder of our project and run inside this project following commands

```
Command Prompt - python manage.py runserver

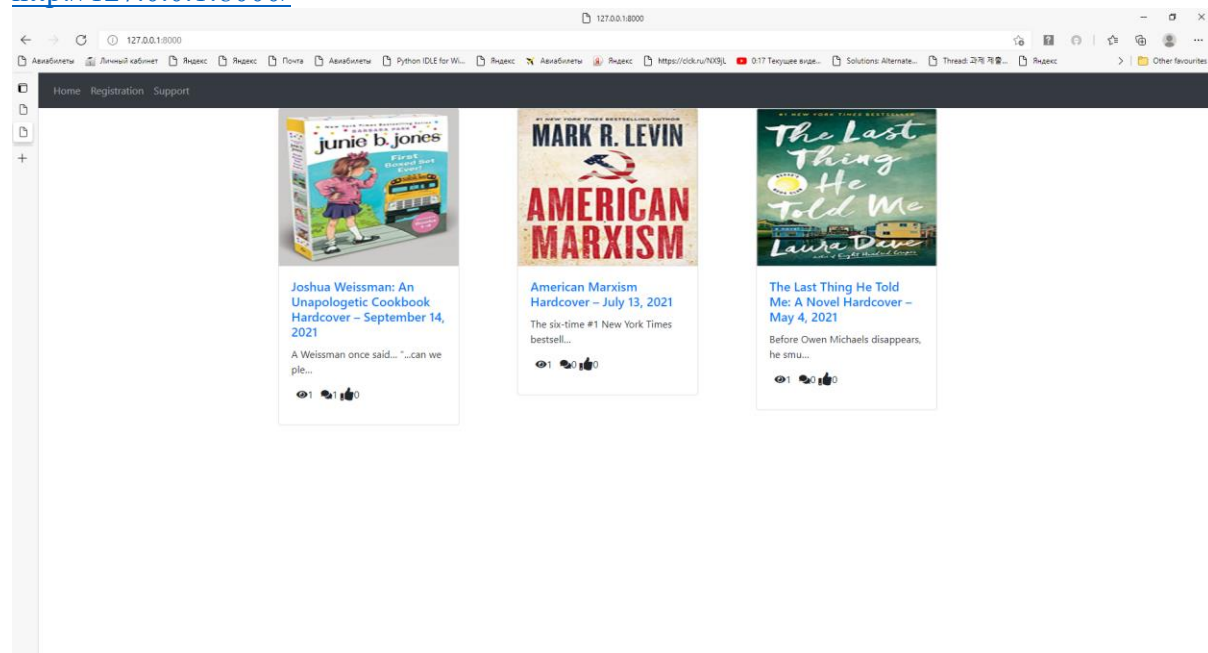
C:\Users\benku\PycharmProjects\shop1>env\Scripts\activate.bat

(shop1) C:\Users\benku\PycharmProjects\shop1>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
June 11, 2021 - 20:39:11
Django version 3.2.3, using settings 'shop1.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Now we can type in browser and see the website.

<http://127.0.0.1:8000/>



For the sake of simplicity, I will add a video demonstration how my program works.

# Implementation

How it was written earlier the concept of creating tables was derived from SQL

```
class author(models.Model):  
    familyName = models.CharField(max_length=20)  
    FirstName = models.CharField(max_length=20)  
    email = models.CharField(max_length=50)
```

```
create table author(  
    id int primary key identity not null,  
    familyName char(20) not null,  
    firstName char(20) not null,  
    Email varchar(50),  
);
```

One code is written in Python and another code in sql. As you can see the syntax is the same and the logic also.

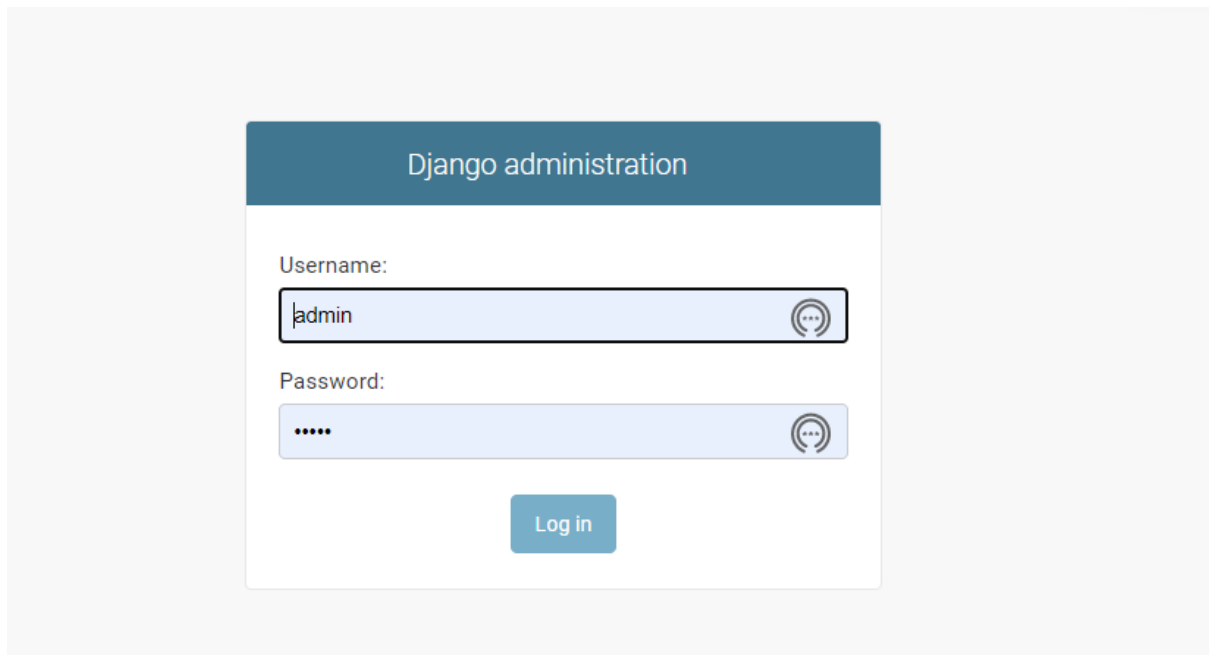
## Admin Panel

We can either add values with insertion.sql in SQL SERVER or with using sqlite3 using database and webpage. However, the main difference between both of them is simplicity in viewing tables as well as insertion new values because of the Html, css, javascripts. I created a super user or admin panel, and if we login as a super user we will have an access to the all tables and values and we have all rights to do what we want to do.

To open an admin panel, we need to write following numbers in search bar

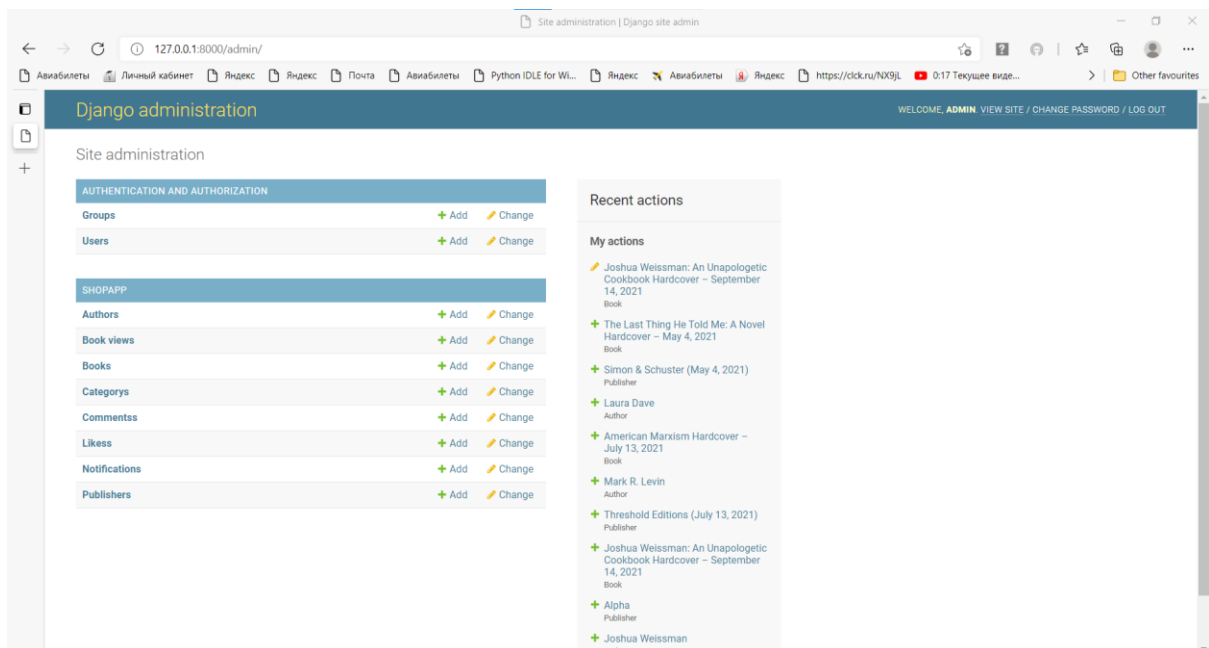
**127.0.0.1:8000/admin**

and login as an admin that I created.



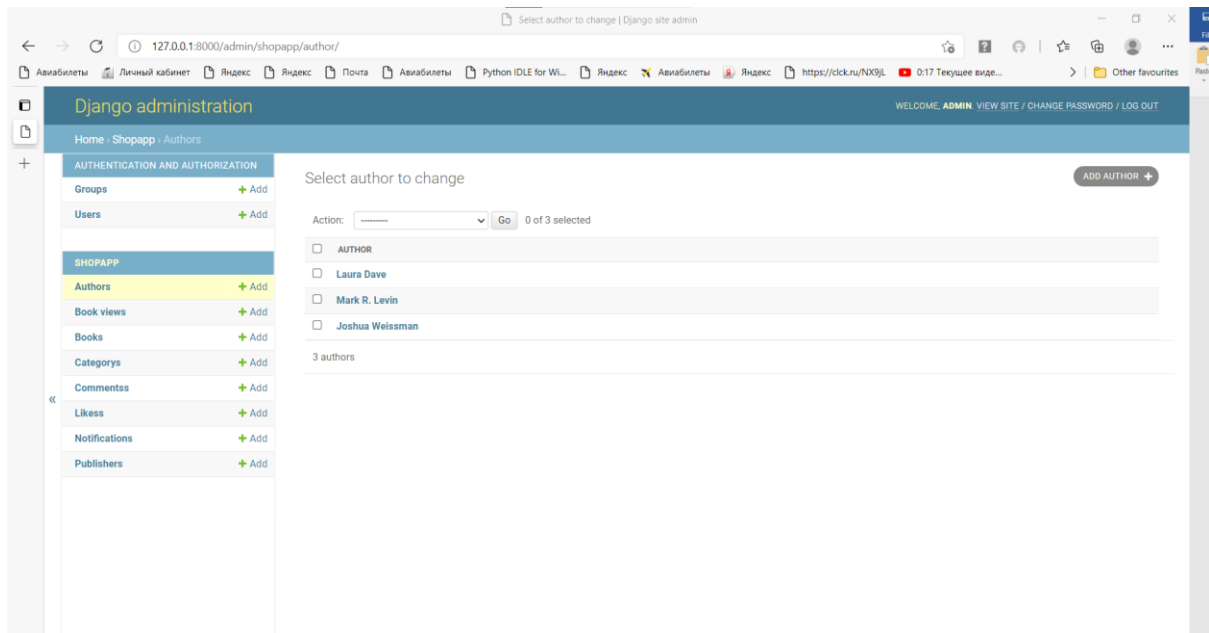
The image shows the Django administration login page. It has a dark blue header with the text "Django administration". Below the header, there are two input fields: "Username:" and "Password:". The "Username:" field contains the text "admin". The "Password:" field contains six dots. To the right of each input field is a circular icon with three dots. Below the input fields is a blue button with the text "Log in".

**Username:admin**  
**Password:admin**



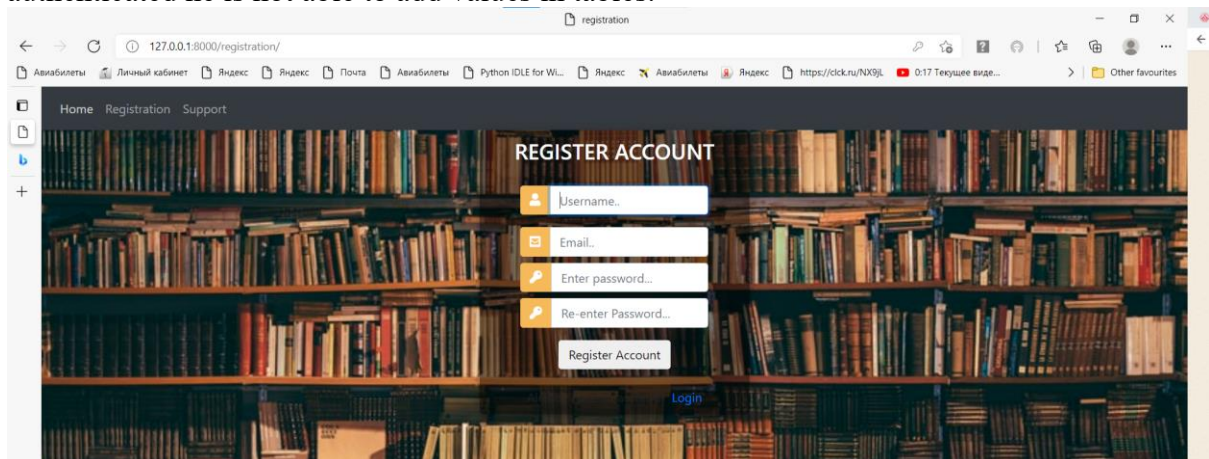
The image shows the Django administration dashboard after logging in as admin. The browser address bar shows "127.0.0.1:8000/admin/". The page has a dark blue header with the text "Django administration" and a welcome message "WELCOME, ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT". The main content area is divided into two columns. The left column is titled "Site administration" and contains two sections: "AUTHENTICATION AND AUTHORIZATION" and "SHOPAPP". The "AUTHENTICATION AND AUTHORIZATION" section has two rows: "Groups" and "Users", each with a green plus icon and a yellow pencil icon. The "SHOPAPP" section has seven rows: "Authors", "Book views", "Books", "Categories", "Comments", "Likess", and "Publishers", each with a green plus icon and a yellow pencil icon. The right column is titled "Recent actions" and contains a section "My actions" with a list of recent actions, including "Joshua Weissman: An Unapologetic Cookbook Hardcover - September 14, 2021" and "The Last Thing He Told Me: A Novel Hardcover - May 4, 2021".

After login as admin, we will have an access to all database and records inside tables. As you can see, we have 10 tables, but we should take into consideration 'Group' table, because it is provided by default. If we want to access Author table, we just click and we can see all the values that are inserted.



## Registration

Registrations is needed to create a discussion forms, leave comments and more if a user is not authenticated he is not able to add values in tables.

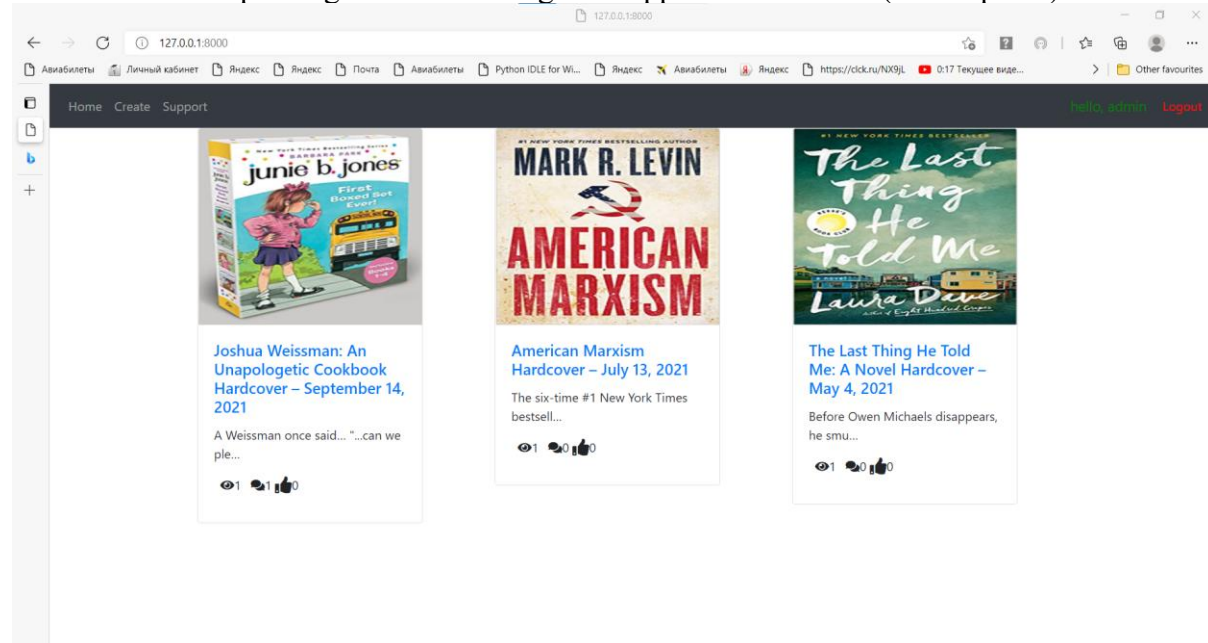


After filling the form, 'user' will be redirected to a login page, where 'user' can sign in.



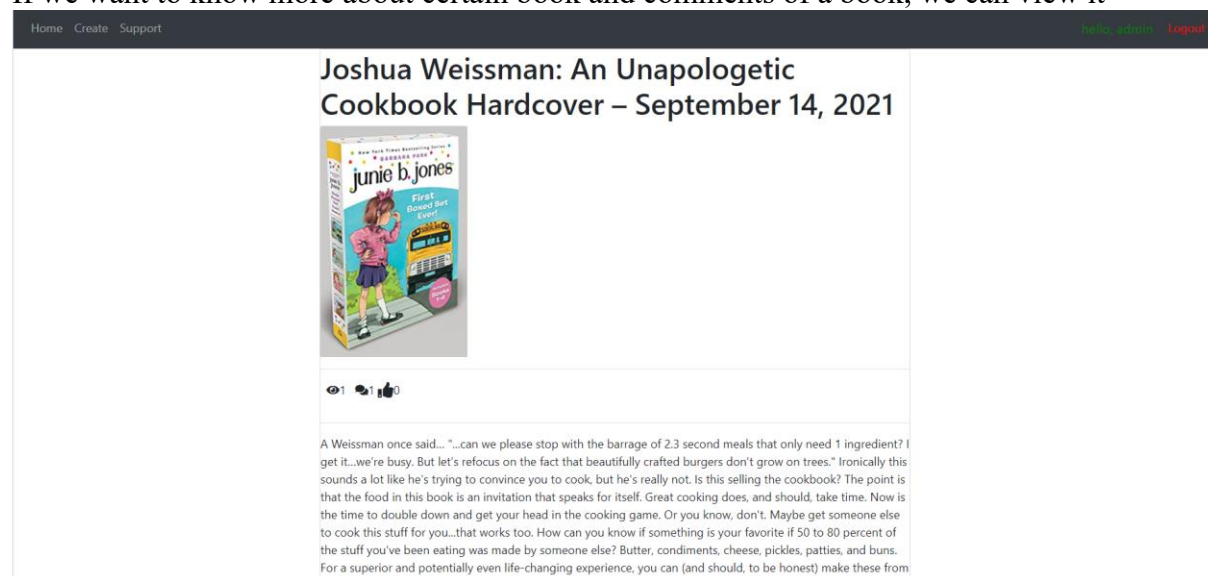
# Home page

In the main page or home page we can see a list of books that are stored in our database. We can see in the left top corner ('Create') it is a link to the new page, where we can add a book description in 'book' table and ('Support') is for connecting with administrators or staff members that are privileged to see message in Support tables in the (admin panel)




# Book Detail

If we want to know more about certain book and comments of a book, we can view it



Joshua Weissman—your own—twist. As Joshua would say, "If you don't like blue cheese, then don't use blue cheese." From simple staples to gourmet to deep-fried, you are the master of your own kitchen, and you'll make it all, on your terms. With no regrets, excuses, or apologies, Joshua Weissman will instruct you how with his irreverent humor, a little bit of light razzing, and over 100 perfectly delectable recipes. If you love to host and entertain; if you like a good project; if you crave control of your food; if fast food or the frozen aisle or the super-fast-super-easy cookbook keeps letting your tastebuds down; then Joshua

Enjoyed this book?

 0

Delete

Update

Leave a comment below

Content\*

Comment

Comments

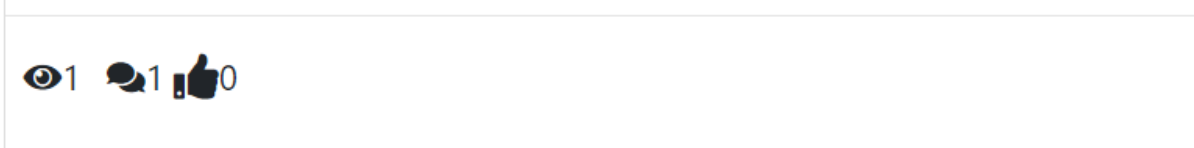
i really like this book

Some time ago

As we already login, we can update, delete a record from a table, but if we log out we do not have rights.

## Icons

Overall we have three icons in each book and every icon has their own functionality.



If we click in the same book every time, the 'eye icon' increases by one, but only with every new user. If we clicked sever times in book with the same user, 'eye icon' increases by one and do not go further, but if login as an another 'user' and click the book, the 'eye icon' will increase by one. 'Comment icon' increases by one every times we add a new comment and 'like' also adds every time when we like a post with the same user. In brief, the functionality is pretty much the same as we can see in other websites.



# Create/Update Form

When we want to add new values in book table, we can do it with special pages.

## update discussion about book

Title\*

Joshua Weissman: An Unapologetic Cookbook Hardcover – September

Img\*

Currently: 51zODjklJoL\_SX316\_BO1204203200.jpg

Change:

Choose File No file chosen

Content\*

cook, but he's really not. Is this selling the cookbook? The point is that the food in this book is an invitation that speaks for itself. Great cooking does, and should, take time. Now is the time to double down and get your head in the cooking game. Or you know, don't. Maybe get someone else to cook this stuff for you...that works too.

How can you know if something is your favorite if 50 to 80 percent of the stuff you've been eating was made by someone else? Butter, condiments, cheese, pickles, patties, and buns. For a

Page num\*

232

Author\*

Joshua Weissman

Category\*

Cookbooks  
Essays  
History  
Memoir

Publisher\*

Alpha

Slug\*

Cookbook1

update

## create discussion about book

Title\*

Img\*

Choose File No file chosen

Content\*

Page num\*

Author\*

.....

Category\*

Action and Adventure  
Classics  
Comic Book or Graphic Novel  
Detective and Mystery

Publisher\*

.....

Slug\*

create

