

## BOOKSTORE

Generated by Doxygen 1.8.11



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Namespace List . . . . .	1
<b>2</b>	<b>Hierarchical Index</b>	<b>3</b>
2.1	Class Hierarchy . . . . .	3
<b>3</b>	<b>Class Index</b>	<b>5</b>
3.1	Class List . . . . .	5
<b>4</b>	<b>File Index</b>	<b>7</b>
4.1	File List . . . . .	7
<b>5</b>	<b>Namespace Documentation</b>	<b>9</b>
5.1	blog Namespace Reference . . . . .	9
5.2	blog.admin Namespace Reference . . . . .	9
5.3	blog.apps Namespace Reference . . . . .	9
5.4	blog.filters Namespace Reference . . . . .	9
5.5	blog.forms Namespace Reference . . . . .	10
5.6	blog.models Namespace Reference . . . . .	10
5.7	blog.tests Namespace Reference . . . . .	11
5.8	blog.urls Namespace Reference . . . . .	11
5.8.1	Variable Documentation . . . . .	11
5.8.1.1	document_root . . . . .	11
5.8.1.2	form1 . . . . .	11
5.8.1.3	MEDIA_URL . . . . .	11

5.8.1.4	urlpatterns	11
5.9	blog.views Namespace Reference	11
5.9.1	Function Documentation	12
5.9.1.1	accept(request, pk, name)	12
5.9.1.2	acceptread(request, pk, pk1)	13
5.9.1.3	adpage(request, pk)	13
5.9.1.4	advertise(request)	13
5.9.1.5	author(request, pk)	13
5.9.1.6	bookpage(request, pk)	13
5.9.1.7	change(request, pk)	13
5.9.1.8	community(request)	13
5.9.1.9	cpage(request, pk)	14
5.9.1.10	cprequest(request, pk)	14
5.9.1.11	crequest(request, pk)	14
5.9.1.12	delete1(request, pk)	14
5.9.1.13	delete1ad(request, pk)	14
5.9.1.14	editprofile(request)	14
5.9.1.15	follow(request, pk)	14
5.9.1.16	genre(request, pk)	15
5.9.1.17	home(request)	15
5.9.1.18	interests(request)	15
5.9.1.19	jcommunities(request)	15
5.9.1.20	leave(request, pk)	15
5.9.1.21	notes(request)	15
5.9.1.22	ocommunities(request)	15
5.9.1.23	prequest(request, pk)	16
5.9.1.24	reset(request)	16
5.9.1.25	search(request)	16
5.9.1.26	searchad(request)	16
5.9.1.27	settings(request)	16

5.9.1.28	<a href="#">signup(request)</a>	16
5.9.1.29	<a href="#">srequest(request, pk)</a>	16
5.9.1.30	<a href="#">unfollow(request, pk)</a>	17
5.9.1.31	<a href="#">upfile(request)</a>	17
5.9.1.32	<a href="#">upfileinc(request, pk)</a>	17
5.9.1.33	<a href="#">uploader(request, pk)</a>	17
5.10	<a href="#">custom_app Namespace Reference</a>	17
5.11	<a href="#">custom_app.admin Namespace Reference</a>	17
5.12	<a href="#">custom_app.apps Namespace Reference</a>	17
5.13	<a href="#">custom_app.models Namespace Reference</a>	18
5.14	<a href="#">custom_app.tests Namespace Reference</a>	18
5.15	<a href="#">custom_app.urls Namespace Reference</a>	18
5.15.1	<a href="#">Variable Documentation</a>	18
5.15.1.1	<a href="#">urlpatterns</a>	18
5.16	<a href="#">custom_app.views Namespace Reference</a>	18
5.17	<a href="#">manage Namespace Reference</a>	18
5.18	<a href="#">mysite Namespace Reference</a>	18
5.19	<a href="#">mysite.asgi Namespace Reference</a>	19
5.19.1	<a href="#">Detailed Description</a>	19
5.19.2	<a href="#">Variable Documentation</a>	19
5.19.2.1	<a href="#">channel_layer</a>	19
5.20	<a href="#">mysite.routing Namespace Reference</a>	19
5.20.1	<a href="#">Variable Documentation</a>	19
5.20.1.1	<a href="#">channel_routing</a>	19
5.21	<a href="#">mysite.settings Namespace Reference</a>	20
5.21.1	<a href="#">Detailed Description</a>	20
5.21.2	<a href="#">Variable Documentation</a>	21
5.21.2.1	<a href="#">ALLOWED_HOSTS</a>	21
5.21.2.2	<a href="#">AUTH_PASSWORD_VALIDATORS</a>	21
5.21.2.3	<a href="#">BASE_DIR</a>	21

5.21.2.4	CHAT_WS_SERVER_HOST	21
5.21.2.5	CHAT_WS_SERVER_PORT	21
5.21.2.6	CHAT_WS_SERVER_PROTOCOL	21
5.21.2.7	DATABASES	21
5.21.2.8	DEBUG	21
5.21.2.9	EMAIL_BACKEND	21
5.21.2.10	INSTALLED_APPS	21
5.21.2.11	INTERNAL_IPS	22
5.21.2.12	I	22
5.21.2.13	LANGUAGE_CODE	22
5.21.2.14	LOGIN_REDIRECT_URL	22
5.21.2.15	LOGIN_URL	22
5.21.2.16	LOGOUT_URL	22
5.21.2.17	MEDIA_ROOT	22
5.21.2.18	MEDIA_URL	22
5.21.2.19	MIDDLEWARE	22
5.21.2.20	MIDDLEWARE_CLASSES	23
5.21.2.21	redis_host	23
5.21.2.22	ROOT_URLCONF	23
5.21.2.23	SECRET_KEY	23
5.21.2.24	SESSION_COOKIE_AGE	23
5.21.2.25	STATIC_ROOT	23
5.21.2.26	STATIC_URL	23
5.21.2.27	TEMPLATES	23
5.21.2.28	TIME_ZONE	23
5.21.2.29	USE_I18N	23
5.21.2.30	USE_L10N	23
5.21.2.31	USE_TZ	23
5.21.2.32	WSGI_APPLICATION	23
5.22	mysite.urls Namespace Reference	23
5.22.1	Variable Documentation	24
5.22.1.1	urlpatterns	24
5.23	mysite.wsgi Namespace Reference	24
5.23.1	Detailed Description	24
5.23.2	Variable Documentation	24
5.23.2.1	application	24

<b>6</b>	<b>Class Documentation</b>	<b>25</b>
6.1	blog.models.Advertise Class Reference	25
6.1.1	Detailed Description	25
6.1.2	Member Data Documentation	26
6.1.2.1	author	26
6.1.2.2	description	26
6.1.2.3	genre	26
6.1.2.4	genre_choices	26
6.1.2.5	image	26
6.1.2.6	published_date	26
6.1.2.7	title	26
6.1.2.8	uploader	26
6.1.2.9	user	27
6.2	blog.filters.AdvertiseFilter Class Reference	27
6.3	blog.forms.AdvertiseForm Class Reference	27
6.3.1	Detailed Description	27
6.4	blog.apps.BlogConfig Class Reference	27
6.5	blog.models.Community Class Reference	28
6.5.1	Detailed Description	28
6.5.2	Member Data Documentation	28
6.5.2.1	admin	28
6.5.2.2	description	28
6.5.2.3	documents	28
6.5.2.4	jrequests	28
6.5.2.5	members	29
6.5.2.6	name	29
6.6	blog.forms.CommunityForm Class Reference	29
6.6.1	Detailed Description	29
6.7	custom_app.apps.CustomAppConfig Class Reference	29
6.7.1	Member Data Documentation	30

6.7.1.1	name	30
6.8	blog.models.Document Class Reference	30
6.8.1	Detailed Description	31
6.8.2	Member Data Documentation	31
6.8.2.1	author	31
6.8.2.2	description	31
6.8.2.3	document	31
6.8.2.4	genre	31
6.8.2.5	genre_choices	31
6.8.2.6	image	31
6.8.2.7	no_ratings	31
6.8.2.8	public	32
6.8.2.9	published_date	32
6.8.2.10	rating	32
6.8.2.11	rmembers	32
6.8.2.12	searchshow	32
6.8.2.13	title	32
6.8.2.14	uploader	32
6.8.2.15	user	32
6.9	blog.forms.DocumentCForm Class Reference	32
6.9.1	Detailed Description	33
6.10	blog.filters.DocumentFilter Class Reference	33
6.11	blog.forms.DocumentForm Class Reference	33
6.11.1	Detailed Description	33
6.12	blog.models.Follow Class Reference	34
6.12.1	Detailed Description	34
6.12.2	Member Data Documentation	34
6.12.2.1	flist	34
6.12.2.2	user	34
6.13	blog.models.Interest Class Reference	34



6.13.1 Detailed Description . . . . .	35
6.13.2 Member Data Documentation . . . . .	35
6.13.2.1 my_field . . . . .	35
6.13.2.2 user . . . . .	35
6.14 blog.forms.InterestForm Class Reference . . . . .	35
6.14.1 Detailed Description . . . . .	36
6.15 blog.models.Join Class Reference . . . . .	36
6.15.1 Detailed Description . . . . .	36
6.15.2 Member Data Documentation . . . . .	36
6.15.2.1 jlist . . . . .	36
6.15.2.2 user . . . . .	36
6.16 blog.forms.RatingForm.Meta Class Reference . . . . .	37
6.16.1 Member Data Documentation . . . . .	37
6.16.1.1 fields . . . . .	37
6.16.1.2 model . . . . .	37
6.17 blog.forms.CommunityForm.Meta Class Reference . . . . .	37
6.17.1 Member Data Documentation . . . . .	37
6.17.1.1 fields . . . . .	37
6.17.1.2 model . . . . .	37
6.18 blog.filters.AdvertiseFilter.Meta Class Reference . . . . .	37
6.18.1 Member Data Documentation . . . . .	38
6.18.1.1 fields . . . . .	38
6.18.1.2 model . . . . .	38
6.19 blog.forms.AdvertiseForm.Meta Class Reference . . . . .	38
6.19.1 Member Data Documentation . . . . .	38
6.19.1.1 fields . . . . .	38
6.19.1.2 model . . . . .	38
6.20 blog.forms.InterestForm.Meta Class Reference . . . . .	38
6.20.1 Member Data Documentation . . . . .	38
6.20.1.1 fields . . . . .	38

6.20.1.2	model	38
6.21	blog.forms.SignUpForm.Meta Class Reference	39
6.21.1	Member Data Documentation	39
6.21.1.1	fields	39
6.21.1.2	model	39
6.22	blog.forms.DocumentCForm.Meta Class Reference	39
6.22.1	Member Data Documentation	39
6.22.1.1	fields	39
6.22.1.2	model	39
6.23	blog.filters.DocumentFilter.Meta Class Reference	39
6.23.1	Member Data Documentation	40
6.23.1.1	fields	40
6.23.1.2	model	40
6.24	blog.forms.DocumentForm.Meta Class Reference	40
6.24.1	Member Data Documentation	40
6.24.1.1	fields	40
6.24.1.2	model	40
6.25	blog.forms.ProfileForm.Meta Class Reference	40
6.25.1	Member Data Documentation	40
6.25.1.1	fields	40
6.25.1.2	model	40
6.26	blog.models.Profile Class Reference	41
6.26.1	Detailed Description	41
6.26.2	Member Data Documentation	41
6.26.2.1	email	41
6.26.2.2	first_name	41
6.26.2.3	last_name	41
6.26.2.4	picture	41
6.26.2.5	user	42
6.27	blog.forms.ProfileForm Class Reference	42

6.27.1 Detailed Description . . . . .	42
6.28 blog.models.Rate Class Reference . . . . .	42
6.28.1 Detailed Description . . . . .	43
6.28.2 Member Data Documentation . . . . .	43
6.28.2.1 doc . . . . .	43
6.28.2.2 rating . . . . .	43
6.28.2.3 user . . . . .	43
6.29 blog.forms.RatingForm Class Reference . . . . .	43
6.29.1 Detailed Description . . . . .	43
6.30 blog.models.Readpending Class Reference . . . . .	44
6.30.1 Detailed Description . . . . .	44
6.30.2 Member Data Documentation . . . . .	44
6.30.2.1 doc . . . . .	44
6.30.2.2 rplist . . . . .	44
6.30.2.3 user . . . . .	44
6.31 blog.forms.SignUpForm Class Reference . . . . .	44
6.31.1 Detailed Description . . . . .	45
6.31.2 Constructor & Destructor Documentation . . . . .	45
6.31.2.1 __init__(self, args, kwargs) . . . . .	45
6.31.3 Member Data Documentation . . . . .	45
6.31.3.1 first_name . . . . .	45
6.31.3.2 last_name . . . . .	45
6.32 custom_app.views.UserListView Class Reference . . . . .	45
6.32.1 Detailed Description . . . . .	46
6.32.2 Member Data Documentation . . . . .	46
6.32.2.1 login_url . . . . .	46
6.32.2.2 model . . . . .	46
6.32.2.3 slug_field . . . . .	46
6.32.2.4 slug_url_kwarg . . . . .	46
6.32.2.5 template_name . . . . .	46

<b>7 File Documentation</b>	<b>47</b>
7.1 bookstore/blog/__init__.py File Reference . . . . .	47
7.2 bookstore/custom_app/__init__.py File Reference . . . . .	47
7.3 bookstore/mysite/__init__.py File Reference . . . . .	47
7.4 bookstore/blog/admin.py File Reference . . . . .	47
7.5 bookstore/custom_app/admin.py File Reference . . . . .	47
7.6 bookstore/blog/apps.py File Reference . . . . .	48
7.7 bookstore/custom_app/apps.py File Reference . . . . .	48
7.8 bookstore/blog/filters.py File Reference . . . . .	48
7.9 bookstore/blog/forms.py File Reference . . . . .	49
7.10 bookstore/blog/models.py File Reference . . . . .	49
7.11 bookstore/custom_app/models.py File Reference . . . . .	50
7.12 bookstore/blog/tests.py File Reference . . . . .	50
7.13 bookstore/custom_app/tests.py File Reference . . . . .	50
7.14 bookstore/blog/urls.py File Reference . . . . .	50
7.15 bookstore/custom_app/urls.py File Reference . . . . .	51
7.16 bookstore/mysite/urls.py File Reference . . . . .	51
7.17 bookstore/blog/views.py File Reference . . . . .	51
7.18 bookstore/custom_app/views.py File Reference . . . . .	53
7.19 bookstore/manage.py File Reference . . . . .	53
7.20 bookstore/mysite/asgi.py File Reference . . . . .	53
7.21 bookstore/mysite/routing.py File Reference . . . . .	53
7.22 bookstore/mysite/settings.py File Reference . . . . .	53
7.23 bookstore/mysite/wsgi.py File Reference . . . . .	54
7.24 bookstore/requirements.txt File Reference . . . . .	54
<b>Index</b>	<b>55</b>

# Chapter 1

## Namespace Index

### 1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">blog</a>	??
<a href="#">blog.admin</a>	??
<a href="#">blog.apps</a>	??
<a href="#">blog.filters</a>	??
<a href="#">blog.forms</a>	??
<a href="#">blog.models</a>	??
<a href="#">blog.tests</a>	??
<a href="#">blog.urls</a>	??
<a href="#">blog.views</a>	??
<a href="#">custom_app</a>	??
<a href="#">custom_app.admin</a>	??
<a href="#">custom_app.apps</a>	??
<a href="#">custom_app.models</a>	??
<a href="#">custom_app.tests</a>	??
<a href="#">custom_app.urls</a>	??
<a href="#">custom_app.views</a>	??
<a href="#">manage</a>	??
<a href="#">mysite</a>	??
<a href="#">mysite.asgi</a>	??
<a href="#">mysite.routing</a>	??
<a href="#">mysite.settings</a>	??
<a href="#">mysite.urls</a>	??
<a href="#">mysite.wsgi</a>	??



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

FilterSet	
blog.filters.AdvertiseFilter . . . . .	27
blog.filters.DocumentFilter . . . . .	33
ListView	
custom_app.views.UserListView . . . . .	45
blog.forms.RatingForm.Meta . . . . .	37
blog.forms.CommunityForm.Meta . . . . .	37
blog.filters.AdvertiseFilter.Meta . . . . .	37
blog.forms.AdvertiseForm.Meta . . . . .	38
blog.forms.InterestForm.Meta . . . . .	38
blog.forms.SignUpForm.Meta . . . . .	39
blog.forms.DocumentCForm.Meta . . . . .	39
blog.filters.DocumentFilter.Meta . . . . .	39
blog.forms.DocumentForm.Meta . . . . .	40
blog.forms.ProfileForm.Meta . . . . .	40
Model	
blog.models.Advertise . . . . .	25
blog.models.Community . . . . .	28
blog.models.Document . . . . .	30
blog.models.Follow . . . . .	34
blog.models.Interest . . . . .	34
blog.models.Join . . . . .	36
blog.models.Profile . . . . .	41
blog.models.Rate . . . . .	42
blog.models.Readpending . . . . .	44
ModelForm	
blog.forms.AdvertiseForm . . . . .	27
blog.forms.CommunityForm . . . . .	29
blog.forms.DocumentCForm . . . . .	32
blog.forms.DocumentForm . . . . .	33
blog.forms.InterestForm . . . . .	35
blog.forms.ProfileForm . . . . .	42
blog.forms.RatingForm . . . . .	43
AppConfig	
blog.apps.BlogConfig . . . . .	27

custom_app.apps.CustomAppConfig . . . . .	29
LoginRequiredMixin	
custom_app.views.UserListView . . . . .	45
UserCreationForm	
blog.forms.SignUpForm . . . . .	44



## Chapter 3

# Class Index

### 3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<a href="#">blog.models.Advertise</a>		
	This class represents the Advertisements uploaded in the website	??
<a href="#">blog.filters.AdvertiseFilter</a>		??
<a href="#">blog.forms.AdvertiseForm</a>		
	This class represents the Form used to upload advertise	??
<a href="#">blog.apps.BlogConfig</a>		??
<a href="#">blog.models.Community</a>		
	This class represents the Communities in the website	??
<a href="#">blog.forms.CommunityForm</a>		
	This class represents the Form used to create community	??
<a href="#">custom_app.apps.CustomAppConfig</a>		??
<a href="#">blog.models.Document</a>		
	This class represents the Documents uploaded in the website	??
<a href="#">blog.forms.DocumentCForm</a>		
	This class represents the Form used to upload document in community	??
<a href="#">blog.filters.DocumentFilter</a>		??
<a href="#">blog.forms.DocumentForm</a>		
	This class represents the Form used to upload document	??
<a href="#">blog.models.Follow</a>		
	This class represents the list of users followed by a user	??
<a href="#">blog.models.Interest</a>		
	This class represents the Genre Interests of Users enrolled in the website	??
<a href="#">blog.forms.InterestForm</a>		
	This class represents the Form used to edit interests of user	??
<a href="#">blog.models.Join</a>		
	This class represents the communities in which user is joined	??
<a href="#">blog.forms.RatingForm.Meta</a>		??
<a href="#">blog.forms.CommunityForm.Meta</a>		??
<a href="#">blog.filters.AdvertiseFilter.Meta</a>		??
<a href="#">blog.forms.AdvertiseForm.Meta</a>		??
<a href="#">blog.forms.InterestForm.Meta</a>		??
<a href="#">blog.forms.SignUpForm.Meta</a>		??
<a href="#">blog.forms.DocumentCForm.Meta</a>		??
<a href="#">blog.filters.DocumentFilter.Meta</a>		??
<a href="#">blog.forms.DocumentForm.Meta</a>		??

<a href="#">blog.forms.ProfileForm.Meta</a>	??
<a href="#">blog.models.Profile</a>	
This class represents the <a href="#">Profile</a> of Users enrolled in the website	??
<a href="#">blog.forms.ProfileForm</a>	
This class represents the Form used to edit profile	??
<a href="#">blog.models.Rate</a>	
This class represents the Rating of the uploaded document given by a user	??
<a href="#">blog.forms.RatingForm</a>	
This class represents the Form used to rate a book	??
<a href="#">blog.models.Readpending</a>	
This class represents the list users who want to read a document if book is private	??
<a href="#">blog.forms.SignUpForm</a>	
This class represents the Form used to sign-up	??
<a href="#">custom_app.views.UserListView</a>	
View for the chat page of the user	??

## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

bookstore/ <a href="#">manage.py</a>	??
bookstore/blog/ <a href="#">__init__.py</a>	??
bookstore/blog/ <a href="#">admin.py</a>	??
bookstore/blog/ <a href="#">apps.py</a>	??
bookstore/blog/ <a href="#">filters.py</a>	??
bookstore/blog/ <a href="#">forms.py</a>	??
bookstore/blog/ <a href="#">models.py</a>	??
bookstore/blog/ <a href="#">tests.py</a>	??
bookstore/blog/ <a href="#">urls.py</a>	??
bookstore/blog/ <a href="#">views.py</a>	??
bookstore/custom_app/ <a href="#">__init__.py</a>	??
bookstore/custom_app/ <a href="#">admin.py</a>	??
bookstore/custom_app/ <a href="#">apps.py</a>	??
bookstore/custom_app/ <a href="#">models.py</a>	??
bookstore/custom_app/ <a href="#">tests.py</a>	??
bookstore/custom_app/ <a href="#">urls.py</a>	??
bookstore/custom_app/ <a href="#">views.py</a>	??
bookstore/mysite/ <a href="#">__init__.py</a>	??
bookstore/mysite/ <a href="#">asgi.py</a>	??
bookstore/mysite/ <a href="#">routing.py</a>	??
bookstore/mysite/ <a href="#">settings.py</a>	??
bookstore/mysite/ <a href="#">urls.py</a>	??
bookstore/mysite/ <a href="#">wsgi.py</a>	??



## Chapter 5

# Namespace Documentation

### 5.1 blog Namespace Reference

#### Namespaces

- [admin](#)
- [apps](#)
- [filters](#)
- [forms](#)
- [models](#)
- [tests](#)
- [urls](#)
- [views](#)

### 5.2 blog.admin Namespace Reference

### 5.3 blog.apps Namespace Reference

#### Classes

- class [BlogConfig](#)

### 5.4 blog.filters Namespace Reference

#### Classes

- class [AdvertiseFilter](#)
- class [DocumentFilter](#)

## 5.5 blog.forms Namespace Reference

### Classes

- class [AdvertiseForm](#)  
*This class represents the Form used to upload advertise.*
- class [CommunityForm](#)  
*This class represents the Form used to create community.*
- class [DocumentCForm](#)  
*This class represents the Form used to upload document in community.*
- class [DocumentForm](#)  
*This class represents the Form used to upload document.*
- class [InterestForm](#)  
*This class represents the Form used to edit interests of user.*
- class [ProfileForm](#)  
*This class represents the Form used to edit profile.*
- class [RatingForm](#)  
*This class represents the Form used to rate a book.*
- class [SignUpForm](#)  
*This class represents the Form used to sign-up.*

## 5.6 blog.models Namespace Reference

### Classes

- class [Advertise](#)  
*This class represents the Advertisements uploaded in the website.*
- class [Community](#)  
*This class represents the Communities in the website.*
- class [Document](#)  
*This class represents the Documents uploaded in the website.*
- class [Follow](#)  
*This class represents the list of users followed by a user.*
- class [Interest](#)  
*This class represents the Genre Interests of Users enrolled in the website.*
- class [Join](#)  
*This class represents the communities in which user is joined.*
- class [Profile](#)  
*This class represents the [Profile](#) of Users enrolled in the website.*
- class [Rate](#)  
*This class represents the Rating of the uploaded document given by a user.*
- class [Readpending](#)  
*This class represents the list users who want to read a document if book is private.*

## 5.7 blog.tests Namespace Reference

## 5.8 blog.urls Namespace Reference

### Variables

- `form1` = `SignUpForm()`
- list `urlpatterns`
- `MEDIA_URL`
- `document_root`

### 5.8.1 Variable Documentation

5.8.1.1 `blog.urls.document_root`

5.8.1.2 `blog.urls.form1` = `SignUpForm()`

5.8.1.3 `blog.urls.MEDIA_URL`

5.8.1.4 list `blog.urls.urlpatterns`

## 5.9 blog.views Namespace Reference

### Functions

- def `home` (request)  
*view for the home page of the user.*
- def `signup` (request)  
*view for the signup page for a new user.*
- def `editprofile` (request)  
*view to edit profile of a user.*
- def `interests` (request)  
*view to edit interets of a user.*
- def `upfile` (request)  
*view to upload document.*
- def `advertise` (request)  
*view to advertise a book*
- def `settings` (request)  
*view for settings page*
- def `notes` (request)  
*view for notifications page*
- def `search` (request)  
*view for search page*
- def `searchad` (request)
- def `delete1` (request, pk)  
*view to delete a book*
- def `delete1ad` (request, pk)

- view to delete a advertise*
- def [bookpage](#) (request, pk)
  - view for book page*
- def [adpage](#) (request, pk)
  - view for ad page*
- def [jcommunities](#) (request)
  - view for joined communities page of a user*
- def [ocommunities](#) (request)
  - view for other communities page of a user*
- def [reset](#) (request)
  - view to edit profile of a user.*
- def [genre](#) (request, pk)
  - view for genre page*
- def [author](#) (request, pk)
  - view for author page*
- def [uploader](#) (request, pk)
  - view for uploader page*
- def [follow](#) (request, pk)
  - view to make one user follow another user*
- def [unfollow](#) (request, pk)
  - view to make one user follow another user*
- def [change](#) (request, pk)
  - view to change privacy of a document*
- def [community](#) (request)
  - view to create a community*
- def [cpage](#) (request, pk)
  - view for community page*
- def [srequest](#) (request, pk)
  - view to sendrequest to join a community by a user*
- def [crequest](#) (request, pk)
  - view to cancelrequest to join a community by a user*
- def [accept](#) (request, pk, name)
  - view to acceptrequest of a user by admin of a community*
- def [leave](#) (request, pk)
  - view for a user to leave a community*
- def [prequest](#) (request, pk)
  - view for a user to send a request to read a private book*
- def [cprequest](#) (request, pk)
  - view for a user to cancel a request sent by him to read a private book*
- def [acceptread](#) (request, pk, pk1)
  - view to acceptreadrequest of a user by uploader of book*
- def [upfileinc](#) (request, pk)
  - view to upload document in a community*

### 5.9.1 Function Documentation

#### 5.9.1.1 def blog.views.accept ( request, pk, name )

view to acceptrequest of a user by admin of a community

This view is called by accept url.  
It returns to cpage.html



#### 5.9.1.2 `def blog.views.acceptread ( request, pk, pk1 )`

view to acceptreadrequest of a user by uploader of book

This view is called by acceptread url.  
It returns to notes.html

#### 5.9.1.3 `def blog.views.adpage ( request, pk )`

view for ad page

This view is called by adpage url.  
It returns to adpage.html

#### 5.9.1.4 `def blog.views.advertise ( request )`

view to advertise a book

This view is called by advertise url.  
It returns to model\_form\_upload.html

#### 5.9.1.5 `def blog.views.author ( request, pk )`

view for author page

This view is called by author url.  
It returns to author.html

#### 5.9.1.6 `def blog.views.bookpage ( request, pk )`

view for book page

This view is called by bookpage url.  
It returns to bookpage.html

#### 5.9.1.7 `def blog.views.change ( request, pk )`

view to change privacy of a document

This view is called by change url.  
It returns to bookpage.html

#### 5.9.1.8 `def blog.views.community ( request )`

view to create a community

This view is called by community url.  
It returns to createcommunity.html

**5.9.1.9** `def blog.views.cpage ( request, pk )`

view for community page

This view is called by cpage url.  
It returns to cpage.html

**5.9.1.10** `def blog.views.cprequest ( request, pk )`

view for a user to cancel a request sent by him to read a private book

This view is called by cprequest url.  
It returns to bookpage.html

**5.9.1.11** `def blog.views.crequest ( request, pk )`

view to cancelrequest to join a community by a user

This view is called by crequest url.  
It returns to othercommunity.html

**5.9.1.12** `def blog.views.delete1 ( request, pk )`

view to delete a book

This view is called by delete url.  
It returns to home.html

**5.9.1.13** `def blog.views.delete1ad ( request, pk )`

view to delete a advertise

This view is called by delete1ad url.  
It returns to home.html

**5.9.1.14** `def blog.views.editprofile ( request )`

view to edit profile of a user.

This view is called by editprofile url.  
It returns to editprofile.html

**5.9.1.15** `def blog.views.follow ( request, pk )`

view to make one user follow another user

This view is called by follow url.  
It returns to uploader.html

**5.9.1.16** `def blog.views.genre ( request, pk )`

view for genre page

This view is called by genre url.  
It returns to genre.html

**5.9.1.17** `def blog.views.home ( request )`

view for the home page of the user.

This view is called by home url.  
It returns to home.html

**5.9.1.18** `def blog.views.interests ( request )`

view to edit interets of a user.

This view is called by interests url.  
It returns to interests.html

**5.9.1.19** `def blog.views.jcommunities ( request )`

view for joined communities page of a user

This view is called by jcommunities url.  
It returns to joinedcommunity.html

**5.9.1.20** `def blog.views.leave ( request, pk )`

view for a user to leave a community

This view is called by leave url.  
It returns to home.html

**5.9.1.21** `def blog.views.notes ( request )`

view for notifications page

This view is called by notes url.  
It returns to notes.html

**5.9.1.22** `def blog.views.ocommunities ( request )`

view for other communities page of a user

This view is called by ocommunities url.  
It returns to othercommunity.html

**5.9.1.23** `def blog.views.prequest ( request, pk )`

view for a user to send a request to read a private book

This view is called by prequest url.  
It returns to bookpage.html

**5.9.1.24** `def blog.views.reset ( request )`

view to edit profile of a user.

This view is called by reset url.  
It returns to change\_password.html

**5.9.1.25** `def blog.views.search ( request )`

view for search page

This view is called by search url.  
It returns to search.html

**5.9.1.26** `def blog.views.searchad ( request )`**5.9.1.27** `def blog.views.settings ( request )`

view for settings page

This view is called by settings url.  
It returns to settings.html

**5.9.1.28** `def blog.views.signup ( request )`

view for the signup page for a new user.

This view is called by signup url.  
It returns to signup.html

**5.9.1.29** `def blog.views.srequest ( request, pk )`

view to sendrequest to join a community by a user

This view is called by srequest url.  
It returns to othercommunity.html

5.9.1.30 `def blog.views.unfollow ( request, pk )`

view to make one user follow another user

This view is called by unfollow url.  
It returns to uploader.html

5.9.1.31 `def blog.views.upfile ( request )`

view to upload document.

This view is called by upfile url.  
It returns to model\_form\_upload.html

5.9.1.32 `def blog.views.upfileinc ( request, pk )`

view to upload document in a community

This view is called by upfilec url.  
It returns to model\_form\_upload.html

5.9.1.33 `def blog.views.uploader ( request, pk )`

view for uploader page

This view is called by uploader url.  
It returns to uploader.html

## 5.10 custom\_app Namespace Reference

### Namespaces

- [admin](#)
- [apps](#)
- [models](#)
- [tests](#)
- [urls](#)
- [views](#)

## 5.11 custom\_app.admin Namespace Reference

## 5.12 custom\_app.apps Namespace Reference

### Classes

- class [CustomAppConfig](#)

## 5.13 custom\_app.models Namespace Reference

## 5.14 custom\_app.tests Namespace Reference

## 5.15 custom\_app.urls Namespace Reference

### Variables

- list [urlpatterns](#)

### 5.15.1 Variable Documentation

#### 5.15.1.1 list custom\_app.urls.urlpatterns

##### Initial value:

```
1 = [  
2     url(  
3         regex=r'^$',  
4         view=views.UserListView.as_view(),  
5         name='user_list'  
6     ),  
7 ]
```

## 5.16 custom\_app.views Namespace Reference

### Classes

- class [UserListView](#)  
*view for the chat page of the user.*

## 5.17 manage Namespace Reference

## 5.18 mysite Namespace Reference

### Namespaces

- [asgi](#)
- [routing](#)
- [settings](#)
- [urls](#)
- [wsgi](#)

## 5.19 mysite.asgi Namespace Reference

### Variables

- `channel_layer` = `get_channel_layer()`

### 5.19.1 Detailed Description

ASGI entrypoint file for default channel layer.

Points to the channel layer configured as "default" so you can point ASGI applications at "multichat.asgi:channel\_layer" as their channel layer.

### 5.19.2 Variable Documentation

#### 5.19.2.1 mysite.asgi.channel\_layer = get\_channel\_layer()

## 5.20 mysite.routing Namespace Reference

### Variables

- list `channel_routing`

### 5.20.1 Variable Documentation

#### 5.20.1.1 list mysite.routing.channel\_routing

#### Initial value:

```
1 = [  
2     # Include sub-routing from an app.  
3     include("chat.routing.websocket_routing", path=r"^/chat/stream"),  
4  
5     # Custom handler for message sending (see Room.send_message).  
6     # Can't go in the include above as it's not got a 'path' attribute to match on.  
7     include("chat.routing.custom_routing"),  
8  
9     # A default "http.request" route is always inserted by Django at the end of the routing list  
10    # that routes all unmatched HTTP requests to the Django view system. If you want lower-level  
11    # HTTP handling - e.g. long-polling - you can do it here and route by path, and let the rest  
12    # fall through to normal views.  
13 ]
```

## 5.21 mysite.settings Namespace Reference

### Variables

- `BASE_DIR` = `os.path.dirname(os.path.dirname(os.path.abspath(__file__)))`
- string `SECRET_KEY` = `'#(wy)8s+g)n@^ ^*dri+uub86%9&5$y#(nql*!y(uer^glj6xrw'`
- bool `DEBUG` = `True`
- list `ALLOWED_HOSTS` = `[]`
- string `CHAT_WS_SERVER_HOST` = `'localhost'`
- int `CHAT_WS_SERVER_PORT` = `5002`
- string `CHAT_WS_SERVER_PROTOCOL` = `'ws'`
- list `INSTALLED_APPS`
- tuple `I`
- list `MIDDLEWARE`
- `MIDDLEWARE_CLASSES` = `I`
- string `ROOT_URLCONF` = `'mysite.urls'`
- `redis_host` = `os.environ.get('REDIS_HOST', 'localhost')`
- list `TEMPLATES`
- string `EMAIL_BACKEND` = `'django.core.mail.backends.console.EmailBackend'`
- string `WSGI_APPLICATION` = `'mysite.wsgi.application'`
- dictionary `DATABASES`
- list `AUTH_PASSWORD_VALIDATORS`
- string `LANGUAGE_CODE` = `'en-us'`
- string `TIME_ZONE` = `'Asia/Kolkata'`
- bool `USE_I18N` = `True`
- bool `USE_L10N` = `True`
- bool `USE_TZ` = `True`
- string `STATIC_URL` = `'/static/'`
- `STATIC_ROOT` = `os.path.join(BASE_DIR, 'static')`
- string `LOGIN_URL` = `'login'`
- string `LOGOUT_URL` = `'logout'`
- string `LOGIN_REDIRECT_URL` = `'home'`
- string `MEDIA_URL` = `'/media/'`
- `MEDIA_ROOT` = `os.path.join(BASE_DIR, 'media')`
- list `INTERNAL_IPS` = `['127.0.0.1', 'localhost']`
- int `SESSION_COOKIE_AGE` = `12096000`

### 5.21.1 Detailed Description

Django settings for mysite project.

Generated by 'django-admin startproject' using Django 1.11.5.

For more information on this file, see  
<https://docs.djangoproject.com/en/1.11/topics/settings/>

For the full list of settings and their values, see  
<https://docs.djangoproject.com/en/1.11/ref/settings/>



## 5.21.2 Variable Documentation

5.21.2.1 list mysite.settings.ALLOWED\_HOSTS = []

5.21.2.2 list mysite.settings.AUTH\_PASSWORD\_VALIDATORS

**Initial value:**

```
1 = [  
2     {  
3         'NAME': 'django.contrib.auth.password_validation.UserAttributeSimilarityValidator',  
4     },  
5     {  
6         'NAME': 'django.contrib.auth.password_validation.MinimumLengthValidator',  
7     },  
8     {  
9         'NAME': 'django.contrib.auth.password_validation.CommonPasswordValidator',  
10    },  
11    {  
12        'NAME': 'django.contrib.auth.password_validation.NumericPasswordValidator',  
13    },  
14 ]
```

5.21.2.3 mysite.settings.BASE\_DIR = os.path.dirname(os.path.dirname(os.path.abspath(\_\_file\_\_)))

5.21.2.4 string mysite.settings.CHAT\_WS\_SERVER\_HOST = 'localhost'

5.21.2.5 int mysite.settings.CHAT\_WS\_SERVER\_PORT = 5002

5.21.2.6 string mysite.settings.CHAT\_WS\_SERVER\_PROTOCOL = 'ws'

5.21.2.7 dictionary mysite.settings.DATABASES

**Initial value:**

```
1 = {  
2     'default': {  
3         'ENGINE': 'django.db.backends.sqlite3',  
4         'NAME': os.path.join(BASE_DIR, 'db.sqlite3'),  
5     }  
6 }
```

5.21.2.8 bool mysite.settings.DEBUG = True

5.21.2.9 string mysite.settings.EMAIL\_BACKEND = 'django.core.mail.backends.console.EmailBackend'

5.21.2.10 list mysite.settings.INSTALLED\_APPS

**Initial value:**

```
1 = [  
2     'django.contrib.admin',  
3     'django.contrib.contenttypes',  
4     'django.contrib.sessions',  
5     'django.contrib.messages',  
6     'django.contrib.staticfiles',  
7     'blog',  
8     #'django_filters'  
9     #'chat',  
10    #'channels',  
11    'django.contrib.auth',  
12    #'debug_toolbar',  
13    'django_private_chat',  
14    'custom_app',  
15 ]
```

5.21.2.11 list mysite.settings.INTERNAL\_IPS = ['127.0.0.1', 'localhost']

5.21.2.12 tuple mysite.settings.I

**Initial value:**

```
1 = (  
2     # 'debug_toolbar.middleware.DebugToolbarMiddleware',  
3     'django.middleware.security.SecurityMiddleware',  
4     'django.contrib.sessions.middleware.SessionMiddleware',  
5     'django.middleware.common.CommonMiddleware',  
6     'django.middleware.csrf.CsrfViewMiddleware',  
7     'django.contrib.auth.middleware.AuthenticationMiddleware',  
8     'django.contrib.auth.middleware.SessionAuthenticationMiddleware',  
9     'django.contrib.messages.middleware.MessageMiddleware',  
10    'django.middleware.clickjacking.XFrameOptionsMiddleware',  
11  
12 )
```

5.21.2.13 string mysite.settings.LANGUAGE\_CODE = 'en-us'

5.21.2.14 string mysite.settings.LOGIN\_REDIRECT\_URL = 'home'

5.21.2.15 string mysite.settings.LOGIN\_URL = 'login'

5.21.2.16 string mysite.settings.LOGOUT\_URL = 'logout'

5.21.2.17 mysite.settings.MEDIA\_ROOT = os.path.join(BASE\_DIR, 'media')

5.21.2.18 string mysite.settings.MEDIA\_URL = '/media/'

5.21.2.19 mysite.settings.MIDDLEWARE

**Initial value:**

```
1 = [  
2     'django.middleware.security.SecurityMiddleware',  
3     'django.contrib.sessions.middleware.SessionMiddleware',  
4     'django.middleware.common.CommonMiddleware',  
5     'django.middleware.csrf.CsrfViewMiddleware',  
6     'django.contrib.auth.middleware.AuthenticationMiddleware',  
7     'django.contrib.messages.middleware.MessageMiddleware',  
8     'django.middleware.clickjacking.XFrameOptionsMiddleware',  
9 ]
```

5.21.2.20 `mysite.settings.MIDDLEWARE_CLASSES = I`

5.21.2.21 `mysite.settings.redis_host = os.environ.get('REDIS_HOST', 'localhost')`

5.21.2.22 `string mysite.settings.ROOT_URLCONF = 'mysite.urls'`

5.21.2.23 `string mysite.settings.SECRET_KEY = '#(wy)8s+g)n@^ ^*dri+uub86%9&5$y#(nql*!y(uer^glj6xrw'`

5.21.2.24 `int mysite.settings.SESSION_COOKIE_AGE = 12096000`

5.21.2.25 `mysite.settings.STATIC_ROOT = os.path.join(BASE_DIR, 'static')`

5.21.2.26 `string mysite.settings.STATIC_URL = '/static/'`

5.21.2.27 `list mysite.settings.TEMPLATES`

**Initial value:**

```
1 = [
2     {
3         'BACKEND': 'django.template.backends.django.DjangoTemplates',
4         'DIRS': [os.path.join(BASE_DIR, 'templates')],
5         'APP_DIRS': True,
6         'OPTIONS': {
7             'context_processors': [
8                 # 'django.template.context_processors.debug',
9                 'django.template.context_processors.request',
10                'django.contrib.auth.context_processors.auth',
11                'django.contrib.messages.context_processors.messages',
12                'django.template.context_processors.media',
13            ],
14        },
15    },
16 ]
```

5.21.2.28 `string mysite.settings.TIME_ZONE = 'Asia/Kolkata'`

5.21.2.29 `bool mysite.settings.USE_I18N = True`

5.21.2.30 `bool mysite.settings.USE_L10N = True`

5.21.2.31 `bool mysite.settings.USE_TZ = True`

5.21.2.32 `string mysite.settings.WSGI_APPLICATION = 'mysite.wsgi.application'`

## 5.22 mysite.urls Namespace Reference

### Variables

- list [urlpatterns](#)

### 5.22.1 Variable Documentation

#### 5.22.1.1 list mysite.urls.urlpatterns

**Initial value:**

```
1 = [  
2     url(r'^admin/', admin.site.urls),  
3     url(r'', include('blog.urls')),  
4     url(r'^chat/', include('django_private_chat.urls')),  
5     url(r'^chat/', include('custom_app.urls'))  
6  
7 ]
```

## 5.23 mysite.wsgi Namespace Reference

### Variables

- `application` = `get_wsgi_application()`

#### 5.23.1 Detailed Description

WSGI config for example project.

It exposes the WSGI callable as a module-level variable named `application`.

For more information on this file, see  
<https://docs.djangoproject.com/en/1.9/howto/deployment/wsgi/>

#### 5.23.2 Variable Documentation

##### 5.23.2.1 mysite.wsgi.application = get\_wsgi\_application()

## Chapter 6

# Class Documentation

### 6.1 blog.models.Advertise Class Reference

This class represents the Advertisements uploaded in the website.

Inheritance diagram for blog.models.Advertise:

Collaboration diagram for blog.models.Advertise:

#### Static Public Attributes

- tuple `genre_choices` = (('Fiction','Fiction'),('LoveandRomance','LoveandRomance'),('Mystery','Mystery'),('Thriller','Thriller'),('ScienceandFiction','ScienceandFiction'),('Fantasy','Fantasy'),('Horror','Horror'),('ActionandAdventure','ActionandAdventure'),('Comedy','Comedy'),('Poetry','Poetry'),('Study','Study'),)  
*A variable which represents genre choices available.*
- `user` = models.ForeignKey('auth.User',default=1)
- `uploader` = models.CharField(max\_length=200,default="")  
*A variable which represents uploader of the document.*
- `title` = models.CharField(max\_length=200,default="")  
*A variable which represents title of the document.*
- `author` = models.CharField(max\_length=200,default="")  
*A variable which represents author of the document.*
- `genre`  
*A variable which represents genre of the document.*
- `description` = models.CharField(max\_length=255, blank=True)  
*A variable which represents description of the document.*
- `image` = models.ImageField(upload\_to='images/',default = 'pic\_folder/None/no-img.jpg')  
*A image field which contains the cover pic of uploaded document.*
- `published_date` = models.DateTimeField(auto\_now\_add=True)  
*A variable represents the published date of the book.*

#### 6.1.1 Detailed Description

This class represents the Advertisements uploaded in the website.

## 6.1.2 Member Data Documentation

6.1.2.1 `blog.models.Advertise.author = models.CharField(max_length=200,default=")` `[static]`

A variable which represents author of the document.

6.1.2.2 `blog.models.Advertise.description = models.CharField(max_length=255, blank=True)` `[static]`

A variable which represents description of the document.

6.1.2.3 `blog.models.Advertise.genre` `[static]`

**Initial value:**

```
1 = models.CharField(
2     max_length=20,
3     choices=genre_choices,
4     default='fiction',)
```

A variable which represents genre of the document.

6.1.2.4 `tuple blog.models.Advertise.genre_choices = (('Fiction','Fiction'),('LoveandRomance','LoveandRomance'),('Mystery','Mystery'),('Thriller','Thriller'),('ScienceandFiction','ScienceandFiction'),('Fantasy','Fantasy'),('Horror','Horror'),('ActionandAdventure','ActionandAdventure'),('Comedy','Comedy'),('Poetry','Poetry'),('Study','Study'),)` `[static]`

A variable which represents genre choices available.

6.1.2.5 `blog.models.Advertise.image = models.ImageField(upload_to='images/',default = 'pic_folder/None/no-img.jpg')` `[static]`

A image field which contains the cover pic of uploaded document.

6.1.2.6 `blog.models.Advertise.published_date = models.DateTimeField(auto_now_add=True)` `[static]`

A variable represents the published date of the book.

6.1.2.7 `blog.models.Advertise.title = models.CharField(max_length=200,default=")` `[static]`

A variable which represents title of the document.

6.1.2.8 `blog.models.Advertise.uploader = models.CharField(max_length=200,default=")` `[static]`

A variable which represents uploader of the document.

6.1.2.9 `blog.models.Advertise.user = models.ForeignKey('auth.User',default=1)` `[static]`

The documentation for this class was generated from the following file:

- [bookstore/blog/models.py](#)

## 6.2 `blog.filters.AdvertiseFilter` Class Reference

Inheritance diagram for `blog.filters.AdvertiseFilter`:

Collaboration diagram for `blog.filters.AdvertiseFilter`:

### Classes

- class [Meta](#)

The documentation for this class was generated from the following file:

- [bookstore/blog/filters.py](#)

## 6.3 `blog.forms.AdvertiseForm` Class Reference

This class represents the Form used to upload advertise.

Inheritance diagram for `blog.forms.AdvertiseForm`:

Collaboration diagram for `blog.forms.AdvertiseForm`:

### Classes

- class [Meta](#)

### 6.3.1 Detailed Description

This class represents the Form used to upload advertise.

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.4 `blog.apps.BlogConfig` Class Reference

Inheritance diagram for `blog.apps.BlogConfig`:

## 6.5 blog.models.Community Class Reference

This class represents the Communities in the website.

Inheritance diagram for blog.models.Community:

Collaboration diagram for blog.models.Community:

### Static Public Attributes

- `admin` = `models.ForeignKey('auth.User',default=1)`  
*Represents The user who is admin of this community.*
- `members` = `models.ManyToManyField('auth.User',related_name='members')`  
*Contains the list of users who are members of this community.*
- `name` = `models.CharField(max_length=200,default='')`  
*A variable which represents the name of `Community`.*
- `description` = `models.CharField(max_length=255, blank=True)`  
*A variable which represents description/purpose of community.*
- `documents` = `models.ManyToManyField('Document',related_name='docs')`  
*A variable which contains the list of documents uploaded/belong to this community.*
- `jrequests` = `models.ManyToManyField('auth.User',related_name='jrequests')`  
*A variable which contains the list of users who sent join request ti this community.*

### 6.5.1 Detailed Description

This class represents the Communities in the website.

### 6.5.2 Member Data Documentation

**6.5.2.1** `blog.models.Community.admin` = `models.ForeignKey('auth.User',default=1)` `[static]`

Represents The user who is admin of this community.

**6.5.2.2** `blog.models.Community.description` = `models.CharField(max_length=255, blank=True)` `[static]`

A variable which represents description/purpose of community.

**6.5.2.3** `blog.models.Community.documents` = `models.ManyToManyField('Document',related_name='docs')` `[static]`

A variable which contains the list of documents uploaded/belong to this community.

**6.5.2.4** `blog.models.Community.jrequests` = `models.ManyToManyField('auth.User',related_name='jrequests')` `[static]`

A variable which contains the list of users who sent join request ti this community.



6.5.2.5 `blog.models.Community.members = models.ManyToManyField('auth.User',related_name='members')` `[static]`

Contains the list of users who are members of this community.

6.5.2.6 `blog.models.Community.name = models.CharField(max_length=200,default='')` `[static]`

A variable which represents the name of `Community`.

The documentation for this class was generated from the following file:

- `bookstore/blog/models.py`

## 6.6 `blog.forms.CommunityForm` Class Reference

This class represents the Form used to create community.

Inheritance diagram for `blog.forms.CommunityForm`:

Collaboration diagram for `blog.forms.CommunityForm`:

### Classes

- class `Meta`

### 6.6.1 Detailed Description

This class represents the Form used to create community.

The documentation for this class was generated from the following file:

- `bookstore/blog/forms.py`

## 6.7 `custom_app.apps.CustomAppConfig` Class Reference

Inheritance diagram for `custom_app.apps.CustomAppConfig`:

Collaboration diagram for `custom_app.apps.CustomAppConfig`:

### Static Public Attributes

- string `name` = `'custom_app'`

### 6.7.1 Member Data Documentation

#### 6.7.1.1 `string custom_app.apps.CustomAppConfig.name = 'custom_app' [static]`

The documentation for this class was generated from the following file:

- [bookstore/custom\\_app/apps.py](#)

## 6.8 `blog.models.Document` Class Reference

This class represents the Documents uploaded in the website.

Inheritance diagram for `blog.models.Document`:

Collaboration diagram for `blog.models.Document`:

### Static Public Attributes

- tuple `genre_choices` = (('Fiction','Fiction'),('LoveandRomance','LoveandRomance'),('Mystery','Mystery'),('Thriller','Thriller'),('ScienceandFiction'),('Fantasy','Fantasy'),('Horror','Horror'),('ActionandAdventure','ActionandAdventure'),('Comedy','Comedy'),('Poetry','Poetry'),('Study','Study'),)  
*A variable which represents genre choices available.*
- `rating` = `models.FloatField(null=True, blank=True, default=0.0)`  
*A variable which represents rating of a document.*
- `no_ratings` = `models.IntegerField(default=0)`  
*A variable which represents no of ratings given to a documnet/book.*
- `user` = `models.ForeignKey('auth.User',default=1)`
- `uploader` = `models.CharField(max_length=200,default="")`  
*A variable which represents uploader of the document.*
- `title` = `models.CharField(max_length=200,default="")`  
*A variable which represents title of the document.*
- `author` = `models.CharField(max_length=200,default="")`  
*A variable which represents author of the document.*
- `genre`  
*A variable which represents genre of the document.*
- `description` = `models.CharField(max_length=255, blank=True)`  
*A variable which represents description of the document.*
- `document` = `models.FileField(upload_to='documents/')`  
*A file field which contains the uploaded document.*
- `image` = `models.ImageField(upload_to='images/',default = 'pic_folder/None/no-img.jpg')`  
*A image field which contains the cover pic of uploaded document.*
- `public` = `models.BooleanField(default=True)`  
*A variable which describes whether the documents is readable by all users or not.*
- `rmembers` = `models.ManyToManyField('auth.User',related_name='rmembers')`  
*A variable which represent list of users who can read the book if it is not public.*
- `published_date` = `models.DateTimeField(auto_now_add=True)`  
*A variable represents the published date of the book.*
- `searchshow` = `models.BooleanField(default=True)`  
*A boolean variable which tells whether the document should be during search or not.*

### 6.8.1 Detailed Description

This class represents the Documents uploaded in the website.

### 6.8.2 Member Data Documentation

**6.8.2.1** `blog.models.Document.author = models.CharField(max_length=200,default='')` `[static]`

A variable which represents author of the document.

**6.8.2.2** `blog.models.Document.description = models.CharField(max_length=255, blank=True)` `[static]`

A variable which represents description of the document.

**6.8.2.3** `blog.models.Document.document = models.FileField(upload_to='documents/')` `[static]`

A file field which contains the uploaded document.

**6.8.2.4** `blog.models.Document.genre` `[static]`

**Initial value:**

```
1 = models.CharField(
2     max_length=20,
3     choices=genre_choices,
4     default='fiction',)
```

A variable which represents genre of the document.

**6.8.2.5** `tuple blog.models.Document.genre_choices = (('Fiction','Fiction'),('LoveandRomance','LoveandRomance'),('Mystery','Mystery'),('Thriller','Thriller'),('ScienceandFiction','ScienceandFiction'),('Fantasy','Fantasy'),('Horror','Horror'),('ActionandAdventure','ActionandAdventure'),('Comedy','Comedy'),('Poetry','Poetry'),('Study','Study'),)` `[static]`

A variable which represents genre choices available.

**6.8.2.6** `blog.models.Document.image = models.ImageField(upload_to='images/',default = 'pic_folder/None/no-img.jpg')` `[static]`

A image field which contains the cover pic of uploaded document.

**6.8.2.7** `blog.models.Document.no_ratings = models.IntegerField(default=0)` `[static]`

A variable which represents no of ratings given to a documnet/book.

**6.8.2.8** `blog.models.Document.public = models.BooleanField(default=True)` `[static]`

A variable which describes whether the documents is readable by all users or not.

**6.8.2.9** `blog.models.Document.published_date = models.DateTimeField(auto_now_add=True)` `[static]`

A variable represents the published date of the book.

**6.8.2.10** `blog.models.Document.rating = models.FloatField(null=True, blank=True, default=0.0)` `[static]`

A variable which represents rating of a document.

**6.8.2.11** `blog.models.Document.rmembers = models.ManyToManyField('auth.User',related_name='rmembers')` `[static]`

A variable which represent list of users who can read the book if it is not public.

**6.8.2.12** `blog.models.Document.searchshow = models.BooleanField(default=True)` `[static]`

A boolean variable which tells whether the document should be during search or not.

**6.8.2.13** `blog.models.Document.title = models.CharField(max_length=200,default="")` `[static]`

A variable which represents title of the document.

**6.8.2.14** `blog.models.Document.uploader = models.CharField(max_length=200,default="")` `[static]`

A variable which represents uploader of the document.

**6.8.2.15** `blog.models.Document.user = models.ForeignKey('auth.User',default=1)` `[static]`

The documentation for this class was generated from the following file:

- [bookstore/blog/models.py](#)

## 6.9 `blog.forms.DocumentCForm` Class Reference

This class represents the Form used to upload document in community.

Inheritance diagram for `blog.forms.DocumentCForm`:

Collaboration diagram for `blog.forms.DocumentCForm`:

## Classes

- class [Meta](#)

### 6.9.1 Detailed Description

This class represents the Form used to upload document in community.

The documentation for this class was generated from the following file:

- bookstore/blog/[forms.py](#)

## 6.10 blog.filters.DocumentFilter Class Reference

Inheritance diagram for blog.filters.DocumentFilter:

Collaboration diagram for blog.filters.DocumentFilter:

## Classes

- class [Meta](#)

The documentation for this class was generated from the following file:

- bookstore/blog/[filters.py](#)

## 6.11 blog.forms.DocumentForm Class Reference

This class represents the Form used to upload document.

Inheritance diagram for blog.forms.DocumentForm:

Collaboration diagram for blog.forms.DocumentForm:

## Classes

- class [Meta](#)

### 6.11.1 Detailed Description

This class represents the Form used to upload document.

The documentation for this class was generated from the following file:

- bookstore/blog/[forms.py](#)

## 6.12 blog.models.Follow Class Reference

This class represents the list of users followed by a user.

Inheritance diagram for blog.models.Follow:

Collaboration diagram for blog.models.Follow:

### Static Public Attributes

- `user` = `models.OneToOneField('auth.User',default=1)`  
*The user associated with the [Rate](#).*
- `flist` = `models.ManyToManyField('auth.User',related_name='owner1')`  
*A variable which contains the list of users followed by the user.*

### 6.12.1 Detailed Description

This class represents the list of users followed by a user.

### 6.12.2 Member Data Documentation

6.12.2.1 `blog.models.Follow.flist = models.ManyToManyField('auth.User',related_name='owner1')` `[static]`

A variable which contains the list of users followed by the user.

6.12.2.2 `blog.models.Follow.user = models.OneToOneField('auth.User',default=1)` `[static]`

The user associated with the [Rate](#).

The documentation for this class was generated from the following file:

- `bookstore/blog/models.py`

## 6.13 blog.models.Interest Class Reference

This class represents the Genre Interests of Users enrolled in the website.

Inheritance diagram for blog.models.Interest:

Collaboration diagram for blog.models.Interest:

## Static Public Attributes

- `user` = `models.OneToOneField('auth.User',default=1)`  
The user associated with the [Interest](#).
- `my_field`  
A multiple choice field which represents the selected interests of user.

### 6.13.1 Detailed Description

This class represents the Genre Interests of Users enrolled in the website.

### 6.13.2 Member Data Documentation

#### 6.13.2.1 `blog.models.Interest.my_field` [static]

**Initial value:**

```
1 = MultiSelectField(choices=
2     (('Fiction','Fiction'),('LoveandRomance','LoveandRomance'),('Mystery',"Mystery"),("Thriller","
    Thriller"),("ScienceandFiction","ScienceandFiction"),("Fantasy","Fantasy"),("Horror","Horror"),("
    ActionandAdventure","ActionandAdventure"),("Comedy","Comedy"),("Poetry","Poetry"),("Study","Study"),))
```

A multiple choice field which represents the selected interests of user.

#### 6.13.2.2 `blog.models.Interest.user = models.OneToOneField('auth.User',default=1)` [static]

The user associated with the [Interest](#).

The documentation for this class was generated from the following file:

- [bookstore/blog/models.py](#)

## 6.14 blog.forms.InterestForm Class Reference

This class represents the Form used to edit interests of user.

Inheritance diagram for `blog.forms.InterestForm`:

Collaboration diagram for `blog.forms.InterestForm`:

## Classes

- class [Meta](#)

### 6.14.1 Detailed Description

This class represents the Form used to edit interests of user.

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.15 blog.models.Join Class Reference

This class represents the communities in which user is joined.

Inheritance diagram for `blog.models.Join`:

Collaboration diagram for `blog.models.Join`:

### Static Public Attributes

- `user = models.ForeignKey('auth.User', default=1)`  
*The user associated with the [Join](#).*
- `jlist = models.ManyToManyField('Community', related_name='jlist')`  
*A variable which contains the list of communities in which user is a member.*

### 6.15.1 Detailed Description

This class represents the communities in which user is joined.

### 6.15.2 Member Data Documentation

#### 6.15.2.1 `blog.models.Join.jlist = models.ManyToManyField('Community', related_name='jlist')` [static]

A variable which contains the list of communities in which user is a member.

#### 6.15.2.2 `blog.models.Join.user = models.ForeignKey('auth.User', default=1)` [static]

The user associated with the [Join](#).

The documentation for this class was generated from the following file:

- [bookstore/blog/models.py](#)



## 6.16 blog.forms.RatingForm.Meta Class Reference

### Static Public Attributes

- `model` = Rate
- tuple `fields` = ('rating',)

### 6.16.1 Member Data Documentation

6.16.1.1 tuple `blog.forms.RatingForm.Meta.fields` = ('rating',) `[static]`

6.16.1.2 `blog.forms.RatingForm.Meta.model` = Rate `[static]`

The documentation for this class was generated from the following file:

- bookstore/blog/[forms.py](#)

## 6.17 blog.forms.CommunityForm.Meta Class Reference

### Static Public Attributes

- `model` = Community
- tuple `fields` = ('name','description',)

### 6.17.1 Member Data Documentation

6.17.1.1 tuple `blog.forms.CommunityForm.Meta.fields` = ('name','description',) `[static]`

6.17.1.2 `blog.forms.CommunityForm.Meta.model` = Community `[static]`

The documentation for this class was generated from the following file:

- bookstore/blog/[forms.py](#)

## 6.18 blog.filters.AdvertiseFilter.Meta Class Reference

### Static Public Attributes

- `model` = Advertise
- dictionary `fields` = {'title':['icontains'], 'genre':['icontains'], 'author':['icontains'], }

### 6.18.1 Member Data Documentation

6.18.1.1 dictionary `blog.filters.AdvertiseFilter.Meta.fields` = `{'title':['icontains'], 'genre':['icontains'], 'author':['icontains'], }` `[static]`

6.18.1.2 `blog.filters.AdvertiseFilter.Meta.model` = `Advertise` `[static]`

The documentation for this class was generated from the following file:

- [bookstore/blog/filters.py](#)

## 6.19 `blog.forms.AdvertiseForm.Meta` Class Reference

### Static Public Attributes

- `model` = `Advertise`
- tuple `fields` = `('title','author','genre','description', 'image')`

### 6.19.1 Member Data Documentation

6.19.1.1 tuple `blog.forms.AdvertiseForm.Meta.fields` = `('title','author','genre','description', 'image')` `[static]`

6.19.1.2 `blog.forms.AdvertiseForm.Meta.model` = `Advertise` `[static]`

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.20 `blog.forms.InterestForm.Meta` Class Reference

### Static Public Attributes

- `model` = `Interest`
- tuple `fields` = `('my_field',)`

### 6.20.1 Member Data Documentation

6.20.1.1 tuple `blog.forms.InterestForm.Meta.fields` = `('my_field',)` `[static]`

6.20.1.2 `blog.forms.InterestForm.Meta.model` = `Interest` `[static]`

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.21 blog.forms.SignUpForm.Meta Class Reference

### Static Public Attributes

- `model` = `User`
- tuple `fields` = ('username', 'first\_name', 'last\_name', 'email', 'password1', 'password2',)

### 6.21.1 Member Data Documentation

6.21.1.1 tuple `blog.forms.SignUpForm.Meta.fields` = ('username', 'first\_name', 'last\_name', 'email', 'password1', 'password2',) `[static]`

6.21.1.2 `blog.forms.SignUpForm.Meta.model` = `User` `[static]`

The documentation for this class was generated from the following file:

- `bookstore/blog/forms.py`

## 6.22 blog.forms.DocumentCForm.Meta Class Reference

### Static Public Attributes

- `model` = `Document`
- tuple `fields` = ('title', 'author', 'genre', 'description', 'document', 'image',)

### 6.22.1 Member Data Documentation

6.22.1.1 tuple `blog.forms.DocumentCForm.Meta.fields` = ('title', 'author', 'genre', 'description', 'document', 'image',) `[static]`

6.22.1.2 `blog.forms.DocumentCForm.Meta.model` = `Document` `[static]`

The documentation for this class was generated from the following file:

- `bookstore/blog/forms.py`

## 6.23 blog.filters.DocumentFilter.Meta Class Reference

### Static Public Attributes

- `model` = `Document`
- dictionary `fields` = {'title':['icontains'], 'genre':['icontains'], 'author':['icontains'], }

### 6.23.1 Member Data Documentation

6.23.1.1 dictionary `blog.filters.DocumentFilter.Meta.fields` = `{'title':['icontains'], 'genre':['icontains'], 'author':['icontains'], }`  
[static]

6.23.1.2 `blog.filters.DocumentFilter.Meta.model` = `Document` [static]

The documentation for this class was generated from the following file:

- [bookstore/blog/filters.py](#)

## 6.24 `blog.forms.DocumentForm.Meta` Class Reference

### Static Public Attributes

- `model` = `Document`
- tuple `fields` = `('title','author','genre','description', 'document', 'image','public')`

### 6.24.1 Member Data Documentation

6.24.1.1 tuple `blog.forms.DocumentForm.Meta.fields` = `('title','author','genre','description', 'document', 'image','public')`  
[static]

6.24.1.2 `blog.forms.DocumentForm.Meta.model` = `Document` [static]

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.25 `blog.forms.ProfileForm.Meta` Class Reference

### Static Public Attributes

- `model` = `Profile`
- tuple `fields` = `('picture','first_name', 'last_name', 'email',)`

### 6.25.1 Member Data Documentation

6.25.1.1 tuple `blog.forms.ProfileForm.Meta.fields` = `('picture','first_name', 'last_name', 'email',)` [static]

6.25.1.2 `blog.forms.ProfileForm.Meta.model` = `Profile` [static]

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.26 blog.models.Profile Class Reference

This class represents the [Profile](#) of Users enrolled in the website.

Inheritance diagram for blog.models.Profile:

Collaboration diagram for blog.models.Profile:

### Static Public Attributes

- `user` = `models.OneToOneField('auth.User', on_delete=models.CASCADE)`  
*The user associated with the [Profile](#).*
- `picture` = `models.ImageField(upload_to='images/', default = 'pic_folder/None/no-img.jpg')`  
*The [Profile](#) picture of the user.*
- `first_name` = `models.CharField(max_length=30)`  
*The firstname picture of the user.*
- `last_name` = `models.CharField(max_length=30)`  
*The lastname picture of the user.*
- `email` = `models.EmailField(max_length=254)`  
*The email-id picture of the user.*

### 6.26.1 Detailed Description

This class represents the [Profile](#) of Users enrolled in the website.

### 6.26.2 Member Data Documentation

6.26.2.1 `blog.models.Profile.email = models.EmailField(max_length=254)` `[static]`

The email-id picture of the user.

6.26.2.2 `blog.models.Profile.first_name = models.CharField(max_length=30)` `[static]`

The firstname picture of the user.

6.26.2.3 `blog.models.Profile.last_name = models.CharField(max_length=30)` `[static]`

The lastname picture of the user.

6.26.2.4 `blog.models.Profile.picture = models.ImageField(upload_to='images/', default = 'pic_folder/None/no-img.jpg')`  
`[static]`

The [Profile](#) picture of the user.

#### 6.26.2.5 `blog.models.Profile.user = models.OneToOneField('auth.User', on_delete=models.CASCADE)` [static]

The user associated with the [Profile](#).

The documentation for this class was generated from the following file:

- [bookstore/blog/models.py](#)

## 6.27 `blog.forms.ProfileForm` Class Reference

This class represents the Form used to edit profile.

Inheritance diagram for `blog.forms.ProfileForm`:

Collaboration diagram for `blog.forms.ProfileForm`:

### Classes

- class [Meta](#)

#### 6.27.1 Detailed Description

This class represents the Form used to edit profile.

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.28 `blog.models.Rate` Class Reference

This class represents the Rating of the uploaded document given by a user.

Inheritance diagram for `blog.models.Rate`:

Collaboration diagram for `blog.models.Rate`:

### Static Public Attributes

- `user` = `models.ForeignKey('auth.User', default=1)`  
*The user associated with the [Rate](#).*
- `doc` = `models.ForeignKey(Document)`  
*The document associated with the [Rate](#).*
- `rating` = `models.IntegerField(choices=((1,1),(2,2),(3,3),(4,4),(5,5)), default=1)`  
*A variable which represents the rating of the document.*

### 6.28.1 Detailed Description

This class represents the Rating of the uploaded document given by a user.

### 6.28.2 Member Data Documentation

6.28.2.1 `blog.models.Rate.doc = models.ForeignKey(Document)` `[static]`

The document associated with the [Rate](#).

6.28.2.2 `blog.models.Rate.rating = models.IntegerField(choices=((1,1),(2,2),(3,3),(4,4),(5,5)),default=1)` `[static]`

A variable which represents the rating of the document.

6.28.2.3 `blog.models.Rate.user = models.ForeignKey('auth.User',default=1)` `[static]`

The user associated with the [Rate](#).

The documentation for this class was generated from the following file:

- [bookstore/blog/models.py](#)

## 6.29 `blog.forms.RatingForm` Class Reference

This class represents the Form used to rate a book.

Inheritance diagram for `blog.forms.RatingForm`:

Collaboration diagram for `blog.forms.RatingForm`:

### Classes

- class [Meta](#)

### 6.29.1 Detailed Description

This class represents the Form used to rate a book.

The documentation for this class was generated from the following file:

- [bookstore/blog/forms.py](#)

## 6.30 blog.models.Readpending Class Reference

This class represents the list users who want to read a document if book is private.

Inheritance diagram for blog.models.Readpending:

Collaboration diagram for blog.models.Readpending:

### Static Public Attributes

- `user = models.ForeignKey('auth.User',default=1)`  
*The user associated with the [Readpending](#).*
- `doc = models.ForeignKey(Document)`  
*The document associated with the [ReadPending](#).*
- `rplist = models.ManyToManyField('auth.User',related_name='rplist')`  
*A variable which contains the list of users in who want to read this private document.*

### 6.30.1 Detailed Description

This class represents the list users who want to read a document if book is private.

### 6.30.2 Member Data Documentation

#### 6.30.2.1 `blog.models.Readpending.doc = models.ForeignKey(Document)` [static]

The document associated with the [ReadPending](#).

#### 6.30.2.2 `blog.models.Readpending.rplist = models.ManyToManyField('auth.User',related_name='rplist')` [static]

A variable which contains the list of users in who want to read this private document.

#### 6.30.2.3 `blog.models.Readpending.user = models.ForeignKey('auth.User',default=1)` [static]

The user associated with the [Readpending](#).

The documentation for this class was generated from the following file:

- `bookstore/blog/models.py`

## 6.31 blog.forms.SignUpForm Class Reference

This class represents the Form used to sign-up.

Inheritance diagram for blog.forms.SignUpForm:

Collaboration diagram for blog.forms.SignUpForm:



## Classes

- class [Meta](#)

## Public Member Functions

- def [\\_\\_init\\_\\_](#) (self, args, kwargs)

## Static Public Attributes

- [first\\_name](#) = forms.CharField(max\_length=30, required=False)
- [last\\_name](#) = forms.CharField(max\_length=30, required=False)

### 6.31.1 Detailed Description

This class represents the Form used to sign-up.

### 6.31.2 Constructor & Destructor Documentation

6.31.2.1 `def blog.forms.SignUpForm.__init__( self, args, kwargs )`

### 6.31.3 Member Data Documentation

6.31.3.1 `blog.forms.SignUpForm.first_name = forms.CharField(max_length=30, required=False)` `[static]`

6.31.3.2 `blog.forms.SignUpForm.last_name = forms.CharField(max_length=30, required=False)` `[static]`

The documentation for this class was generated from the following file:

- bookstore/blog/[forms.py](#)

## 6.32 custom\_app.views.UserListView Class Reference

view for the chat page of the user.

Inheritance diagram for custom\_app.views.UserListView:

Collaboration diagram for custom\_app.views.UserListView:

## Static Public Attributes

- [model](#) = get\_user\_model()
- string [slug\\_field](#) = 'username'
- string [slug\\_url\\_kwarg](#) = 'username'
- string [template\\_name](#) = 'custom\_app/users.html'
- string [login\\_url](#) = 'admin/'

### 6.32.1 Detailed Description

view for the chat page of the user.

This view is called by messages url.  
It returns to users.html

### 6.32.2 Member Data Documentation

6.32.2.1 `string custom_app.views.UserListView.login_url = 'admin/'` `[static]`

6.32.2.2 `custom_app.views.UserListView.model = get_user_model()` `[static]`

6.32.2.3 `string custom_app.views.UserListView.slug_field = 'username'` `[static]`

6.32.2.4 `string custom_app.views.UserListView.slug_url_kwarg = 'username'` `[static]`

6.32.2.5 `string custom_app.views.UserListView.template_name = 'custom_app/users.html'` `[static]`

The documentation for this class was generated from the following file:

- bookstore/custom\_app/[views.py](#)

## Chapter 7

# File Documentation

### 7.1 bookstore/blog/\_\_init\_\_.py File Reference

#### Namespaces

- [blog](#)

### 7.2 bookstore/custom\_app/\_\_init\_\_.py File Reference

#### Namespaces

- [custom\\_app](#)

### 7.3 bookstore/mysite/\_\_init\_\_.py File Reference

#### Namespaces

- [mysite](#)

### 7.4 bookstore/blog/admin.py File Reference

#### Namespaces

- [blog.admin](#)

### 7.5 bookstore/custom\_app/admin.py File Reference

#### Namespaces

- [custom\\_app.admin](#)

## 7.6 bookstore/blog/apps.py File Reference

### Classes

- class [blog.apps.BlogConfig](#)

### Namespaces

- [blog.apps](#)

## 7.7 bookstore/custom\_app/apps.py File Reference

### Classes

- class [custom\\_app.apps.CustomAppConfig](#)

### Namespaces

- [custom\\_app.apps](#)

## 7.8 bookstore/blog/filters.py File Reference

### Classes

- class [blog.filters.DocumentFilter](#)
- class [blog.filters.DocumentFilter.Meta](#)
- class [blog.filters.AdvertiseFilter](#)
- class [blog.filters.AdvertiseFilter.Meta](#)

### Namespaces

- [blog.filters](#)

## 7.9 bookstore/blog/forms.py File Reference

### Classes

- class [blog.forms.ProfileForm](#)  
*This class represents the Form used to edit profile.*
- class [blog.forms.ProfileForm.Meta](#)
- class [blog.forms.SignUpForm](#)  
*This class represents the Form used to sign-up.*
- class [blog.forms.SignUpForm.Meta](#)
- class [blog.forms.InterestForm](#)  
*This class represents the Form used to edit interests of user.*
- class [blog.forms.InterestForm.Meta](#)
- class [blog.forms.DocumentForm](#)  
*This class represents the Form used to upload document.*
- class [blog.forms.DocumentForm.Meta](#)
- class [blog.forms.DocumentCForm](#)  
*This class represents the Form used to upload document in community.*
- class [blog.forms.DocumentCForm.Meta](#)
- class [blog.forms.RatingForm](#)  
*This class represents the Form used to rate a book.*
- class [blog.forms.RatingForm.Meta](#)
- class [blog.forms.CommunityForm](#)  
*This class represents the Form used to create community.*
- class [blog.forms.CommunityForm.Meta](#)
- class [blog.forms.AdvertiseForm](#)  
*This class represents the Form used to upload advertise.*
- class [blog.forms.AdvertiseForm.Meta](#)

### Namespaces

- [blog.forms](#)

## 7.10 bookstore/blog/models.py File Reference

### Classes

- class [blog.models.Profile](#)  
*This class represents the [Profile](#) of Users enrolled in the website.*
- class [blog.models.Interest](#)  
*This class represents the Genre Interests of Users enrolled in the website.*
- class [blog.models.Document](#)  
*This class represents the Documents uploaded in the website.*
- class [blog.models.Rate](#)  
*This class represents the Rating of the uploaded document given by a user.*
- class [blog.models.Follow](#)  
*This class represents the list of users followed by a user.*
- class [blog.models.Community](#)

*This class represents the Communities in the website.*

- class [blog.models.Join](#)

*This class represents the communities in which user is joined.*

- class [blog.models.Advertise](#)

*This class represents the Advertisements uploaded in the website.*

- class [blog.models.Readpending](#)

*This class represents the list users who want to read a document if book is private.*

## Namespaces

- [blog.models](#)

## 7.11 bookstore/custom\_app/models.py File Reference

### Namespaces

- [custom\\_app.models](#)

## 7.12 bookstore/blog/tests.py File Reference

### Namespaces

- [blog.tests](#)

## 7.13 bookstore/custom\_app/tests.py File Reference

### Namespaces

- [custom\\_app.tests](#)

## 7.14 bookstore/blog/urls.py File Reference

### Namespaces

- [blog.urls](#)

### Variables

- [blog.urls.form1](#) = SignUpForm()
- list [blog.urls.urlpatterns](#)
- [blog.urls.MEDIA\\_URL](#)
- [blog.urls.document\\_root](#)

## 7.15 bookstore/custom\_app/urls.py File Reference

### Namespaces

- [custom\\_app.urls](#)

### Variables

- list [custom\\_app.urls.urlpatterns](#)

## 7.16 bookstore/mysite/urls.py File Reference

### Namespaces

- [mysite.urls](#)

### Variables

- list [mysite.urls.urlpatterns](#)

## 7.17 bookstore/blog/views.py File Reference

### Namespaces

- [blog.views](#)

### Functions

- def [blog.views.home](#) (request)  
*view for the home page of the user.*
- def [blog.views.signup](#) (request)  
*view for the signup page for a new user.*
- def [blog.views.editprofile](#) (request)  
*view to edit profile of a user.*
- def [blog.views.interests](#) (request)  
*view to edit interets of a user.*
- def [blog.views.upfile](#) (request)  
*view to upload document.*
- def [blog.views.advertise](#) (request)  
*view to advertise a book*
- def [blog.views.settings](#) (request)  
*view for settings page*
- def [blog.views.notes](#) (request)  
*view for notifications page*
- def [blog.views.search](#) (request)

- view for search page*
- def [blog.views.searchad](#) (request)
- def [blog.views.delete1](#) (request, pk)
  - view to delete a book*
- def [blog.views.delete1ad](#) (request, pk)
  - view to delete a advertise*
- def [blog.views.bookpage](#) (request, pk)
  - view for book page*
- def [blog.views.adpage](#) (request, pk)
  - view for ad page*
- def [blog.views.jcommunities](#) (request)
  - view for joined communities page of a user*
- def [blog.views.ocommunities](#) (request)
  - view for other communities page of a user*
- def [blog.views.reset](#) (request)
  - view to edit profile of a user.*
- def [blog.views.genre](#) (request, pk)
  - view for genre page*
- def [blog.views.author](#) (request, pk)
  - view for author page*
- def [blog.views.uploader](#) (request, pk)
  - view for uploader page*
- def [blog.views.follow](#) (request, pk)
  - view to make one user follow another user*
- def [blog.views.unfollow](#) (request, pk)
  - view to make one user follow another user*
- def [blog.views.change](#) (request, pk)
  - view to change privacy of a document*
- def [blog.views.community](#) (request)
  - view to create a community*
- def [blog.views.cpage](#) (request, pk)
  - view for community page*
- def [blog.views.srequest](#) (request, pk)
  - view to sendrequest to join a community by a user*
- def [blog.views.crequest](#) (request, pk)
  - view to cancelrequest to join a community by a user*
- def [blog.views.accept](#) (request, pk, name)
  - view to acceptrequest of a user by admin of a community*
- def [blog.views.leave](#) (request, pk)
  - view for a user to leave a community*
- def [blog.views.prequest](#) (request, pk)
  - view for a user to send a request to read a private book*
- def [blog.views.cprequest](#) (request, pk)
  - view for a user to cancel a request sent by him to read a private book*
- def [blog.views.acceptread](#) (request, pk, pk1)
  - view to acceptreadrequest of a user by uploader of book*
- def [blog.views.upfileinc](#) (request, pk)
  - view to upload document in a community*



## 7.18 bookstore/custom\_app/views.py File Reference

### Classes

- class [custom\\_app.views.UserListView](#)  
*view for the chat page of the user.*

### Namespaces

- [custom\\_app.views](#)

## 7.19 bookstore/manage.py File Reference

### Namespaces

- [manage](#)

## 7.20 bookstore/mysite/asgi.py File Reference

### Namespaces

- [mysite.asgi](#)

### Variables

- [mysite.asgi.channel\\_layer](#) = `get_channel_layer()`

## 7.21 bookstore/mysite/routing.py File Reference

### Namespaces

- [mysite.routing](#)

### Variables

- list [mysite.routing.channel\\_routing](#)

## 7.22 bookstore/mysite/settings.py File Reference

### Namespaces

- [mysite.settings](#)

## Variables

- `mysite.settings.BASE_DIR` = `os.path.dirname(os.path.dirname(os.path.abspath(__file__)))`
- string `mysite.settings.SECRET_KEY` = `'#(wy)8s+g)n@^_^*dri+uub86%9&5$y#(nql*ly(uer^glj6xrw'`
- bool `mysite.settings.DEBUG` = `True`
- list `mysite.settings.ALLOWED_HOSTS` = `[]`
- string `mysite.settings.CHAT_WS_SERVER_HOST` = `'localhost'`
- int `mysite.settings.CHAT_WS_SERVER_PORT` = `5002`
- string `mysite.settings.CHAT_WS_SERVER_PROTOCOL` = `'ws'`
- list `mysite.settings.INSTALLED_APPS`
- tuple `mysite.settings.l`
- list `mysite.settings.MIDDLEWARE`
- `mysite.settings.MIDDLEWARE_CLASSES` = `I`
- string `mysite.settings.ROOT_URLCONF` = `'mysite.urls'`
- `mysite.settings.redis_host` = `os.environ.get('REDIS_HOST', 'localhost')`
- list `mysite.settings.TEMPLATES`
- string `mysite.settings.EMAIL_BACKEND` = `'django.core.mail.backends.console.EmailBackend'`
- string `mysite.settings.WSGI_APPLICATION` = `'mysite.wsgi.application'`
- dictionary `mysite.settings.DATABASES`
- list `mysite.settings.AUTH_PASSWORD_VALIDATORS`
- string `mysite.settings.LANGUAGE_CODE` = `'en-us'`
- string `mysite.settings.TIME_ZONE` = `'Asia/Kolkata'`
- bool `mysite.settings.USE_I18N` = `True`
- bool `mysite.settings.USE_L10N` = `True`
- bool `mysite.settings.USE_TZ` = `True`
- string `mysite.settings.STATIC_URL` = `'/static/'`
- `mysite.settings.STATIC_ROOT` = `os.path.join(BASE_DIR, 'static')`
- string `mysite.settings.LOGIN_URL` = `'login'`
- string `mysite.settings.LOGOUT_URL` = `'logout'`
- string `mysite.settings.LOGIN_REDIRECT_URL` = `'home'`
- string `mysite.settings.MEDIA_URL` = `'/media/'`
- `mysite.settings.MEDIA_ROOT` = `os.path.join(BASE_DIR, 'media')`
- list `mysite.settings.INTERNAL_IPS` = `['127.0.0.1', 'localhost']`
- int `mysite.settings.SESSION_COOKIE_AGE` = `12096000`

## 7.23 bookstore/mysite/wsgi.py File Reference

### Namespaces

- `mysite.wsgi`

### Variables

- `mysite.wsgi.application` = `get_wsgi_application()`

## 7.24 bookstore/requirements.txt File Reference

# Index

- `__init__`
  - `blog::forms::SignUpForm`, [45](#)
- `ALLOWED_HOSTS`
  - `mysite::settings`, [21](#)
- `AUTH_PASSWORD_VALIDATORS`
  - `mysite::settings`, [21](#)
- `accept`
  - `blog::views`, [12](#)
- `acceptread`
  - `blog::views`, [12](#)
- `admin`
  - `blog::models::Community`, [28](#)
- `adpage`
  - `blog::views`, [13](#)
- `advertise`
  - `blog::views`, [13](#)
- `application`
  - `mysite::wsgi`, [24](#)
- `author`
  - `blog::models::Advertise`, [26](#)
  - `blog::models::Document`, [31](#)
  - `blog::views`, [13](#)
- `BASE_DIR`
  - `mysite::settings`, [21](#)
- `blog`, [9](#)
- `blog.admin`, [9](#)
- `blog.apps`, [9](#)
- `blog.apps.BlogConfig`, [27](#)
- `blog.filters`, [9](#)
- `blog.filters.AdvertiseFilter`, [27](#)
- `blog.filters.AdvertiseFilter.Meta`, [37](#)
- `blog.filters.DocumentFilter`, [33](#)
- `blog.filters.DocumentFilter.Meta`, [39](#)
- `blog.forms`, [10](#)
- `blog.forms.AdvertiseForm`, [27](#)
- `blog.forms.AdvertiseForm.Meta`, [38](#)
- `blog.forms.CommunityForm`, [29](#)
- `blog.forms.CommunityForm.Meta`, [37](#)
- `blog.forms.DocumentCForm`, [32](#)
- `blog.forms.DocumentCForm.Meta`, [39](#)
- `blog.forms.DocumentForm`, [33](#)
- `blog.forms.DocumentForm.Meta`, [40](#)
- `blog.forms.InterestForm`, [35](#)
- `blog.forms.InterestForm.Meta`, [38](#)
- `blog.forms.ProfileForm`, [42](#)
- `blog.forms.ProfileForm.Meta`, [40](#)
- `blog.forms.RatingForm`, [43](#)
- `blog.forms.RatingForm.Meta`, [37](#)
- `blog.forms.SignUpForm`, [44](#)
- `blog.forms.SignUpForm.Meta`, [39](#)
- `blog.models`, [10](#)
- `blog.models.Advertise`, [25](#)
- `blog.models.Community`, [28](#)
- `blog.models.Document`, [30](#)
- `blog.models.Follow`, [34](#)
- `blog.models.Interest`, [34](#)
- `blog.models.Join`, [36](#)
- `blog.models.Profile`, [41](#)
- `blog.models.Rate`, [42](#)
- `blog.models.Readpending`, [44](#)
- `blog.tests`, [11](#)
- `blog.urls`, [11](#)
- `blog.views`, [11](#)
- `blog::filters::AdvertiseFilter::Meta`
  - `fields`, [38](#)
  - `model`, [38](#)
- `blog::filters::DocumentFilter::Meta`
  - `fields`, [40](#)
  - `model`, [40](#)
- `blog::forms::AdvertiseForm::Meta`
  - `fields`, [38](#)
  - `model`, [38](#)
- `blog::forms::CommunityForm::Meta`
  - `fields`, [37](#)
  - `model`, [37](#)
- `blog::forms::DocumentCForm::Meta`
  - `fields`, [39](#)
  - `model`, [39](#)
- `blog::forms::DocumentForm::Meta`
  - `fields`, [40](#)
  - `model`, [40](#)
- `blog::forms::InterestForm::Meta`
  - `fields`, [38](#)
  - `model`, [38](#)
- `blog::forms::ProfileForm::Meta`
  - `fields`, [40](#)
  - `model`, [40](#)
- `blog::forms::RatingForm::Meta`
  - `fields`, [37](#)
  - `model`, [37](#)
- `blog::forms::SignUpForm`
  - `__init__`, [45](#)
  - `first_name`, [45](#)
  - `last_name`, [45](#)
- `blog::forms::SignUpForm::Meta`
  - `fields`, [39](#)
  - `model`, [39](#)

- blog::models::Advertise
  - author, [26](#)
  - description, [26](#)
  - genre, [26](#)
  - genre\_choices, [26](#)
  - image, [26](#)
  - published\_date, [26](#)
  - title, [26](#)
  - uploader, [26](#)
  - user, [26](#)
- blog::models::Community
  - admin, [28](#)
  - description, [28](#)
  - documents, [28](#)
  - jrequests, [28](#)
  - members, [28](#)
  - name, [29](#)
- blog::models::Document
  - author, [31](#)
  - description, [31](#)
  - document, [31](#)
  - genre, [31](#)
  - genre\_choices, [31](#)
  - image, [31](#)
  - no\_ratings, [31](#)
  - public, [31](#)
  - published\_date, [32](#)
  - rating, [32](#)
  - rmembers, [32](#)
  - searchshow, [32](#)
  - title, [32](#)
  - uploader, [32](#)
  - user, [32](#)
- blog::models::Follow
  - flist, [34](#)
  - user, [34](#)
- blog::models::Interest
  - my\_field, [35](#)
  - user, [35](#)
- blog::models::Join
  - jlist, [36](#)
  - user, [36](#)
- blog::models::Profile
  - email, [41](#)
  - first\_name, [41](#)
  - last\_name, [41](#)
  - picture, [41](#)
  - user, [41](#)
- blog::models::Rate
  - doc, [43](#)
  - rating, [43](#)
  - user, [43](#)
- blog::models::Readpending
  - doc, [44](#)
  - rplist, [44](#)
  - user, [44](#)
- blog::urls
  - document\_root, [11](#)
  - form1, [11](#)
  - MEDIA\_URL, [11](#)
  - urlpatterns, [11](#)
- blog::views
  - accept, [12](#)
  - acceptread, [12](#)
  - adpage, [13](#)
  - advertise, [13](#)
  - author, [13](#)
  - bookpage, [13](#)
  - change, [13](#)
  - community, [13](#)
  - cpage, [13](#)
  - cprequest, [14](#)
  - crequest, [14](#)
  - delete1, [14](#)
  - delete1ad, [14](#)
  - editprofile, [14](#)
  - follow, [14](#)
  - genre, [14](#)
  - home, [15](#)
  - interests, [15](#)
  - jcommunities, [15](#)
  - leave, [15](#)
  - notes, [15](#)
  - ocommunities, [15](#)
  - prequest, [15](#)
  - reset, [16](#)
  - search, [16](#)
  - searchad, [16](#)
  - settings, [16](#)
  - signup, [16](#)
  - srequest, [16](#)
  - unfollow, [16](#)
  - upfile, [17](#)
  - upfileinc, [17](#)
  - uploader, [17](#)
- bookpage
  - blog::views, [13](#)
- bookstore/blog/\_\_init\_\_.py, [47](#)
- bookstore/blog/admin.py, [47](#)
- bookstore/blog/apps.py, [48](#)
- bookstore/blog/filters.py, [48](#)
- bookstore/blog/forms.py, [49](#)
- bookstore/blog/models.py, [49](#)
- bookstore/blog/tests.py, [50](#)
- bookstore/blog/urls.py, [50](#)
- bookstore/blog/views.py, [51](#)
- bookstore/custom\_app/\_\_init\_\_.py, [47](#)
- bookstore/custom\_app/admin.py, [47](#)
- bookstore/custom\_app/apps.py, [48](#)
- bookstore/custom\_app/models.py, [50](#)
- bookstore/custom\_app/tests.py, [50](#)
- bookstore/custom\_app/urls.py, [51](#)
- bookstore/custom\_app/views.py, [53](#)
- bookstore/manage.py, [53](#)
- bookstore/mysite/\_\_init\_\_.py, [47](#)
- bookstore/mysite/asgi.py, [53](#)

- bookstore/mysite/routing.py, [53](#)
- bookstore/mysite/settings.py, [53](#)
- bookstore/mysite/urls.py, [51](#)
- bookstore/mysite/wsgi.py, [54](#)
- bookstore/requirements.txt, [54](#)
- CHAT\_WS\_SERVER\_HOST
  - mysite::settings, [21](#)
- CHAT\_WS\_SERVER\_PORT
  - mysite::settings, [21](#)
- CHAT\_WS\_SERVER\_PROTOCOL
  - mysite::settings, [21](#)
- change
  - blog::views, [13](#)
- channel\_layer
  - mysite::asgi, [19](#)
- channel\_routing
  - mysite::routing, [19](#)
- community
  - blog::views, [13](#)
- cpage
  - blog::views, [13](#)
- cprequest
  - blog::views, [14](#)
- crequest
  - blog::views, [14](#)
- custom\_app, [17](#)
- custom\_app.admin, [17](#)
- custom\_app.apps, [17](#)
- custom\_app.apps.CustomAppConfig, [29](#)
- custom\_app.models, [18](#)
- custom\_app.tests, [18](#)
- custom\_app.urls, [18](#)
- custom\_app.views, [18](#)
- custom\_app.views.UserListView, [45](#)
- custom\_app::apps::CustomAppConfig
  - name, [30](#)
- custom\_app::urls
  - urlpatterns, [18](#)
- custom\_app::views::UserListView
  - login\_url, [46](#)
  - model, [46](#)
  - slug\_field, [46](#)
  - slug\_url\_kwarg, [46](#)
  - template\_name, [46](#)
- DATABASES
  - mysite::settings, [21](#)
- DEBUG
  - mysite::settings, [21](#)
- delete1
  - blog::views, [14](#)
- delete1ad
  - blog::views, [14](#)
- description
  - blog::models::Advertise, [26](#)
  - blog::models::Community, [28](#)
  - blog::models::Document, [31](#)
- doc
  - blog::models::Rate, [43](#)
  - blog::models::Readpending, [44](#)
- document
  - blog::models::Document, [31](#)
- document\_root
  - blog::urls, [11](#)
- documents
  - blog::models::Community, [28](#)
- EMAIL\_BACKEND
  - mysite::settings, [21](#)
- editprofile
  - blog::views, [14](#)
- email
  - blog::models::Profile, [41](#)
- fields
  - blog::filters::AdvertiseFilter::Meta, [38](#)
  - blog::filters::DocumentFilter::Meta, [40](#)
  - blog::forms::AdvertiseForm::Meta, [38](#)
  - blog::forms::CommunityForm::Meta, [37](#)
  - blog::forms::DocumentCForm::Meta, [39](#)
  - blog::forms::DocumentForm::Meta, [40](#)
  - blog::forms::InterestForm::Meta, [38](#)
  - blog::forms::ProfileForm::Meta, [40](#)
  - blog::forms::RatingForm::Meta, [37](#)
  - blog::forms::SignUpForm::Meta, [39](#)
- first\_name
  - blog::forms::SignUpForm, [45](#)
  - blog::models::Profile, [41](#)
- flist
  - blog::models::Follow, [34](#)
- follow
  - blog::views, [14](#)
- form1
  - blog::urls, [11](#)
- genre
  - blog::models::Advertise, [26](#)
  - blog::models::Document, [31](#)
  - blog::views, [14](#)
- genre\_choices
  - blog::models::Advertise, [26](#)
  - blog::models::Document, [31](#)
- home
  - blog::views, [15](#)
- INSTALLED\_APPS
  - mysite::settings, [21](#)
- INTERNAL\_IPS
  - mysite::settings, [21](#)
- image
  - blog::models::Advertise, [26](#)
  - blog::models::Document, [31](#)
- interests
  - blog::views, [15](#)
- jcommunities
  - blog::views, [15](#)

- jlist
  - blog::models::Join, 36
- jrequests
  - blog::models::Community, 28
- l
  - mysite::settings, 22
- LANGUAGE\_CODE
  - mysite::settings, 22
- LOGIN\_REDIRECT\_URL
  - mysite::settings, 22
- LOGIN\_URL
  - mysite::settings, 22
- LOGOUT\_URL
  - mysite::settings, 22
- last\_name
  - blog::forms::SignUpForm, 45
  - blog::models::Profile, 41
- leave
  - blog::views, 15
- login\_url
  - custom\_app::views::UserListView, 46
- MEDIA\_ROOT
  - mysite::settings, 22
- MEDIA\_URL
  - blog::urls, 11
  - mysite::settings, 22
- MIDDLEWARE\_CLASSES
  - mysite::settings, 22
- MIDDLEWARE
  - mysite::settings, 22
- manage, 18
- members
  - blog::models::Community, 28
- model
  - blog::filters::AdvertiseFilter::Meta, 38
  - blog::filters::DocumentFilter::Meta, 40
  - blog::forms::AdvertiseForm::Meta, 38
  - blog::forms::CommunityForm::Meta, 37
  - blog::forms::DocumentCForm::Meta, 39
  - blog::forms::DocumentForm::Meta, 40
  - blog::forms::InterestForm::Meta, 38
  - blog::forms::ProfileForm::Meta, 40
  - blog::forms::RatingForm::Meta, 37
  - blog::forms::SignUpForm::Meta, 39
  - custom\_app::views::UserListView, 46
- my\_field
  - blog::models::Interest, 35
- mysite, 18
- mysite.asgi, 19
- mysite.routing, 19
- mysite.settings, 20
- mysite.urls, 23
- mysite.wsgi, 24
- mysite::asgi
  - channel\_layer, 19
- mysite::routing
  - channel\_routing, 19
- mysite::settings
  - ALLOWED\_HOSTS, 21
  - AUTH\_PASSWORD\_VALIDATORS, 21
  - BASE\_DIR, 21
  - CHAT\_WS\_SERVER\_HOST, 21
  - CHAT\_WS\_SERVER\_PORT, 21
  - CHAT\_WS\_SERVER\_PROTOCOL, 21
  - DATABASES, 21
  - DEBUG, 21
  - EMAIL\_BACKEND, 21
  - INSTALLED\_APPS, 21
  - INTERNAL\_IPS, 21
  - l, 22
  - LANGUAGE\_CODE, 22
  - LOGIN\_REDIRECT\_URL, 22
  - LOGIN\_URL, 22
  - LOGOUT\_URL, 22
  - MEDIA\_ROOT, 22
  - MEDIA\_URL, 22
  - MIDDLEWARE\_CLASSES, 22
  - MIDDLEWARE, 22
  - ROOT\_URLCONF, 23
  - redis\_host, 23
  - SECRET\_KEY, 23
  - SESSION\_COOKIE\_AGE, 23
  - STATIC\_ROOT, 23
  - STATIC\_URL, 23
  - TEMPLATES, 23
  - TIME\_ZONE, 23
  - USE\_I18N, 23
  - USE\_L10N, 23
  - USE\_TZ, 23
  - WSGI\_APPLICATION, 23
- mysite::urls
  - urlpatterns, 24
- mysite::wsgi
  - application, 24
- name
  - blog::models::Community, 29
  - custom\_app::apps::CustomAppConfig, 30
- no\_ratings
  - blog::models::Document, 31
- notes
  - blog::views, 15
- ocommunities
  - blog::views, 15
- picture
  - blog::models::Profile, 41
- prequest
  - blog::views, 15
- public
  - blog::models::Document, 31
- published\_date
  - blog::models::Advertise, 26
  - blog::models::Document, 32

- ROOT\_URLCONF
  - mysite::settings, 23
- rating
  - blog::models::Document, 32
  - blog::models::Rate, 43
- redis\_host
  - mysite::settings, 23
- reset
  - blog::views, 16
- rmembers
  - blog::models::Document, 32
- rplists
  - blog::models::Readpending, 44
- SECRET\_KEY
  - mysite::settings, 23
- SESSION\_COOKIE\_AGE
  - mysite::settings, 23
- STATIC\_ROOT
  - mysite::settings, 23
- STATIC\_URL
  - mysite::settings, 23
- search
  - blog::views, 16
- searchad
  - blog::views, 16
- searchshow
  - blog::models::Document, 32
- settings
  - blog::views, 16
- signup
  - blog::views, 16
- slug\_field
  - custom\_app::views::UserListView, 46
- slug\_url\_kwarg
  - custom\_app::views::UserListView, 46
- srequest
  - blog::views, 16
- TEMPLATES
  - mysite::settings, 23
- TIME\_ZONE
  - mysite::settings, 23
- template\_name
  - custom\_app::views::UserListView, 46
- title
  - blog::models::Advertise, 26
  - blog::models::Document, 32
- USE\_I18N
  - mysite::settings, 23
- USE\_L10N
  - mysite::settings, 23
- USE\_TZ
  - mysite::settings, 23
- unfollow
  - blog::views, 16
- upfile
  - blog::views, 17
- upfileinc
  - blog::views, 17
- uploader
  - blog::models::Advertise, 26
  - blog::models::Document, 32
  - blog::views, 17
- urlpatterns
  - blog::urls, 11
  - custom\_app::urls, 18
  - mysite::urls, 24
- user
  - blog::models::Advertise, 26
  - blog::models::Document, 32
  - blog::models::Follow, 34
  - blog::models::Interest, 35
  - blog::models::Join, 36
  - blog::models::Profile, 41
  - blog::models::Rate, 43
  - blog::models::Readpending, 44
- WSGI\_APPLICATION
  - mysite::settings, 23