

MANUAL DE SISTEMA



Biaci On The Go Documentation

Versión 0.1

Miguel Méndez, Yenifer Ramírez y Manuel Zambrano

27 de junio de 2018

Contents:

1. Indices and tables	1
2. Contents:	3
2.1. Views Aplicacion de Reserva	3
2.2. Modules Aplicacion de Reserva	3
2.3. Forms Aplicacion de Reserva	4
2.4. Views Aplicacion de Consulta	4
2.5. Modules Aplicacion de Consulta	4
2.6. Views Aplicacion de Libro	4
2.7. Modules Aplicacion de Libro	5
2.8. Forms Aplicacion de Libro	6
2.9. Views Aplicacion de Prestamo	6
2.10. Modules Aplicacion de Prestamo	6
2.11. Views Aplicacion de Revista	7
2.12. Modules Aplicacion de Revista	7
2.13. Forms Aplicacion de Revista	7
2.14. Views Aplicacion de Tesis	7
2.15. Modules Aplicacion de Tesis	8
2.16. Forms Aplicacion de Tesis	8
2.17. Views Aplicacion de Usuario	8
2.18. Modules Aplicacion de Usuarios	8
2.19. Forms Aplicacion de Usuarios	9
Índice de Módulos Python	11
Índice	13

CAPÍTULO 1

Indices and tables

- `genindex`
- `modindex`
- `search`

2.1 Views Aplicacion de Reserva

class reserva.views.**ErrorReserva** (**kwargs)

Esta clase muestra un error si la reserva no se pudo realizar

class reserva.views.**ReservaLibros** (**kwargs)

Esta clase verifica las reservas realizadas por el usuario

get (request, *args, **kwargs)

Funcion para recopilar la informacion de los ejemplares reservados por el usuario logueado

A partir de los datos de la consulta, se listan los datos de la reserva

Devuelve Una plantilla llamada *lista_reserva.html*

Tipo del valor devuelto Template Response

reserva.views.**Reservar** (request, id_ejemplar)

Funcion para reservar un ejemplar.

0- Verifica que el mismo ejemplar no haya sido reservado en los ultimos dos dias por el mismo usuario 1- Reserva el ejemplar. 2. Guarda la reserva en un historial. 3. Cambia el estado del ejemplar de disponible a reservado.

Devuelve un render a *reservar.html*

Tipo del valor devuelto render

2.2 Modules Aplicacion de Reserva

class reserva.models.**HistorialReserva** (id, id_reserva, accion, fecha)

exception DoesNotExist


```
exception MultipleObjectsReturned

get_absolute_url()
    Devuelve la url para acceder a un registro de esta reserva

class reserva.models.Reserva(id, fecha_reserva, fecha_caducidad, id_ejemplar, id_usuario, estado)

exception DoesNotExist
exception MultipleObjectsReturned

get_absolute_url()
    Devuelve la url para acceder a un registro de esta reserva
```

2.3 Forms Aplicacion de Reserva

```
class reserva.forms.ReservaForm(*args, **kwargs)
```

2.4 Views Aplicacion de Consulta

```
class consulta.views.ConsultaList(**kwargs)
```

2.5 Modules Aplicacion de Consulta

```
class consulta.models.Consulta(*args, **kwargs)
    Clase que define las consultas hechas por un usuario

exception DoesNotExist
exception MultipleObjectsReturned

get_absolute_url()
    Devuelve la url para acceder a un registro de esta consulta
```

2.6 Views Aplicacion de Libro

```
class libro.views.ConsultaTodo(**kwargs)
    Esta clase lista todo los libros, revista y tesis disponibles en el OPAC.
```

Nota: La busqueda solo se realiza al usar el buscador del menu.

```
class libro.views.EjemplaresVista(**kwargs)
    Esta clase lista los ejemplares disponibles de un libro, previa llamada de la clase LibrosVista

get(request, pk, **kwargs)
    Funcion para recopilar la informacion de los ejemplares del libro seleccionado

    A partir de los datos de la consulta, se listan los ejemplares disponibles, dependiendo de lo retornado en el formulario.

    Devuelve Una render a la plantilla ejemplar.html
```

Tipo del valor devuelto Render

class libro.views.**LibrosVista** (**kwargs)

Esta clase recibe los parametros del formulario *ConsultaLibroForm*

post (request, *args, **kwargs)

La funcion proviene de la clase *LibrosVista*

Se comparan los distintos parametros recibidos por el post para realizar la consulta.

Devuelve Retorna vista *libros.html*

Tipo del valor devuelto TemplateResponse

libro.views.**buscador** (request)

Funcion para mostrar un libro en la data

A partir de los datos de la consulta, se listan los libros disponibles, dependiendo de lo retornado en el formulario.

Parámetros

- **func** (*Llamada*) – Funcion buscador
- **args** (*Request*) – Argumento a procesar

Devuelve Una render a la plantilla *buscar.html*

Tipo del valor devuelto Render

Nota: Se utilizan datos del formulario *ConsultaLibroForm*

2.7 Modules Aplicacion de Libro

class libro.models.**Autor** (*args, **kwargs)

Representacion del modelo Autor.

exception DoesNotExist

exception MultipleObjectsReturned

get_absolute_url ()

Retorna la url para ver detalles de autor.

class libro.models.**Biblioteca** (*args, **kwargs)

Modelo que representa un biblioteca de la ULA

Example Biaci, Trujillo, etc.

exception DoesNotExist

exception MultipleObjectsReturned

class libro.models.**Ejemplar** (*args, **kwargs)

Modelo que representa un ejemplar de un libro

exception DoesNotExist

exception MultipleObjectsReturned

get_absolute_url ()

Devuelve la url para acceder a un registro de este ejemplar

```
class libro.models.Idioma(*args, **kwargs)
    Modelo que representa un lenguaje

    Example ej. español, ingles, etc.

    exception DoesNotExist

    exception MultipleObjectsReturned

class libro.models.Libro(*args, **kwargs)
    Modelo que representa un libro (pero no un ejemplar para prestar).

    exception DoesNotExist

    exception MultipleObjectsReturned

    get_absolute_url()
        Devuelve la url para acceder a un registro de este libro

    save(*args, **kwargs)
        Funcion que descargar la portada del libro pasada por el url y lo guarda localmente

        Parámetros

        ■ args – Parametro Vacio

        ■ kwargs – Parametro Vacio

        Devuelve una funcion de guardar.

class libro.models.Materia(*args, **kwargs)
    Modelo que representa una materia

    exception DoesNotExist

    exception MultipleObjectsReturned
```

2.8 Forms Aplicacion de Libro

```
class libro.forms.ConsultaLibroForm(data=None, files=None, auto_id='id_%s', pre-
                                     fix=None, initial=None, error_class=<class 'djan-
                                     go.forms.utils.ErrorList'>, label_suffix=None,
                                     empty_permitted=False, instance=None,
                                     use_required_attribute=None)
```

2.9 Views Aplicacion de Prestamo

2.10 Modules Aplicacion de Prestamo

```
class prestamo.models.Prestamo(id, id_usuario, id_ejemplar, fecha_prestamo, fecha_entrega)

    exception DoesNotExist

    exception MultipleObjectsReturned
```

2.11 Views Aplicacion de Revista

```
class revista.views.EjemplaresVista (**kwargs)
```

```
class revista.views.RevistaVista (**kwargs)
```

2.12 Modules Aplicacion de Revista

```
class revista.models.EjemplarRevista (*args, **kwargs)
```

Modelo que representa un ejemplar de una revista

exception DoesNotExist

exception MultipleObjectsReturned

get_absolute_url()

Devuelve la url para acceder a un registro de este ejemplar

```
class revista.models.Revista (*args, **kwargs)
```

Modelo que representa una revista (pero no un ejemplar para prestar).

exception DoesNotExist

exception MultipleObjectsReturned

get_absolute_url()

Devuelve la url para acceder a un registro de esta revista

save (*args, **kwargs)

Funcion que descargar la portada del libro pasada por el url y lo guarda localmente

Parámetros

- **args** – Parametro Vacio
- **kwargs** – Parametro Vacio

Devuelve una funcion de guardar.

2.13 Forms Aplicacion de Revista

```
class revista.forms.ConsultaRevistaForm(data=None, files=None, auto_id='id_%s', pre-
                                         fix=None, initial=None, error_class=<class 'djan-
                                         go.forms.utils.ErrorList'>, label_suffix=None,
                                         empty_permitted=False, instance=None,
                                         use_required_attribute=None)
```

2.14 Views Aplicacion de Tesis

```
class tesis.views.EjemplaresTesis (**kwargs)
```

```
class tesis.views.TesisVista (**kwargs)
```

2.15 Modules Aplicacion de Tesis

```
class tesis.models.Tesis (*args, **kwargs)
    Modelo que representa una tesis (pero no un ejemplar para prestar).

    exception DoesNotExist

    exception MultipleObjectsReturned

    get_absolute_url()
        Devuelve la url para acceder a u registro de esta revista
```

2.16 Forms Aplicacion de Tesis

```
class tesis.forms.ConsultaTesisForm(data=None, files=None, auto_id='id_%s', pre-
                                     fix=None, initial=None, error_class=<class 'djan-
                                     go.forms.utils.ErrorList'>, label_suffix=None,
                                     empty_permitted=False, instance=None,
                                     use_required_attribute=None)
```

2.17 Views Aplicacion de Usuario

```
class users.views.ErrorSignUp (**kwargs)

users.views.PerfilView(request)
    Funcion que muestra el perfil del usuario, ademas consulta si tiene prestamos proximos a vencerse para notifi-
    carlo :param request: :return: Render

class users.views.SignUp (**kwargs)

    form_class
        alias de users.forms.CustomUserCreationForm

    form_valid(form)
        If the form is valid, save the associated model.
```

2.18 Modules Aplicacion de Usuarios

```
class users.models.CompleteUser(id, password, last_login, is_superuser, username, first_name,
                                  last_name, email, is_staff, is_active, date_joined, customu-
                                  ser_ptr, cedula, estado, sexo, cod_area, num_telefono, direc-
                                  cion)

    exception DoesNotExist

    exception MultipleObjectsReturned

class users.models.CustomUser(id, password, last_login, is_superuser, username, first_name,
                                last_name, email, is_staff, is_active, date_joined)

    exception DoesNotExist

    exception MultipleObjectsReturned
```

```
class users.models.LogCorreo (*args, **kwargs)
    Modelo para los log del envio de correo :Example:

        Envio de correo de una reserva, notificacion,...

    exception DoesNotExist

    exception MultipleObjectsReturned

    get_absolute_url()
        Devuelve la url para acceder a un log de los correos

class users.models.RegisteredUserId (*args, **kwargs)
    «docstring for RegisteredUser

    exception DoesNotExist

    exception MultipleObjectsReturned
```

2.19 Forms Aplicacion de Usuarios

```
class users.forms.CustomUserChangeForm (*args, **kwargs)

class users.forms.CustomUserCreationForm (*args, **kwargs)
```


c

`consulta.models`, 4
`consulta.views`, 4

l

`libro.forms`, 6
`libro.models`, 5
`libro.views`, 4

p

`prestamo.models`, 6
`prestamo.views`, 6

r

`reserva.forms`, 4
`reserva.models`, 3
`reserva.views`, 3
`revista.forms`, 7
`revista.models`, 7
`revista.views`, 7

t

`tesis.forms`, 8
`tesis.models`, 8
`tesis.views`, 7

u

`users.forms`, 9
`users.models`, 8
`users.views`, 8

A

Autor (clase en libro.models), 5
Autor.DoesNotExist, 5
Autor.MultipleObjectsReturned, 5

B

Biblioteca (clase en libro.models), 5
Biblioteca.DoesNotExist, 5
Biblioteca.MultipleObjectsReturned, 5
buscador() (en el módulo libro.views), 5

C

CompleteUser (clase en users.models), 8
CompleteUser.DoesNotExist, 8
CompleteUser.MultipleObjectsReturned, 8
Consulta (clase en consulta.models), 4
Consulta.DoesNotExist, 4
consulta.models (módulo), 4
Consulta.MultipleObjectsReturned, 4
consulta.views (módulo), 4
ConsultaLibroForm (clase en libro.forms), 6
ConsultaList (clase en consulta.views), 4
ConsultaRevistaForm (clase en revista.forms), 7
ConsultaTesisForm (clase en tesis.forms), 8
ConsultaTodo (clase en libro.views), 4
CustomUser (clase en users.models), 8
CustomUser.DoesNotExist, 8
CustomUser.MultipleObjectsReturned, 8
CustomUserChangeForm (clase en users.forms), 9
CustomUserCreationForm (clase en users.forms), 9

E

Ejemplar (clase en libro.models), 5
Ejemplar.DoesNotExist, 5
Ejemplar.MultipleObjectsReturned, 5
EjemplaresTesis (clase en tesis.views), 7
EjemplaresVista (clase en libro.views), 4
EjemplaresVista (clase en revista.views), 7
EjemplarRevista (clase en revista.models), 7

EjemplarRevista.DoesNotExist, 7
EjemplarRevista.MultipleObjectsReturned, 7
ErrorReserva (clase en reserva.views), 3
ErrorSignUp (clase en users.views), 8

F

form_class (atributo de users.views.SignUp), 8
form_valid() (método de users.views.SignUp), 8

G

get() (método de libro.views.EjemplaresVista), 4
get() (método de reserva.views.ReservaLibros), 3
get_absolute_url() (método de consulta.models.Consulta), 4
get_absolute_url() (método de libro.models.Autor), 5
get_absolute_url() (método de libro.models.Ejemplar), 5
get_absolute_url() (método de libro.models.Libro), 6
get_absolute_url() (método de reserva.models.HistorialReserva), 4
get_absolute_url() (método de reserva.models.Reserva), 4
get_absolute_url() (método de revista.models.EjemplarRevista), 7
get_absolute_url() (método de revista.models.Revista), 7
get_absolute_url() (método de tesis.models.Tesis), 8
get_absolute_url() (método de users.models.LogCorreo), 9

H

HistorialReserva (clase en reserva.models), 3
HistorialReserva.DoesNotExist, 3
HistorialReserva.MultipleObjectsReturned, 3

I

Idioma (clase en libro.models), 5
Idioma.DoesNotExist, 6
Idioma.MultipleObjectsReturned, 6

L

Libro (clase en libro.models), 6

Libro.DoesNotExist, 6
libro.forms (módulo), 6
libro.models (módulo), 5
Libro.MultipleObjectsReturned, 6
libro.views (módulo), 4
LibrosVista (clase en libro.views), 5
LogCorreo (clase en users.models), 8
LogCorreo.DoesNotExist, 9
LogCorreo.MultipleObjectsReturned, 9

M

Materia (clase en libro.models), 6
Materia.DoesNotExist, 6
Materia.MultipleObjectsReturned, 6

P

PerfilView() (en el módulo users.views), 8
post() (método de libro.views.LibrosVista), 5
Prestamo (clase en prestamo.models), 6
Prestamo.DoesNotExist, 6
prestamo.models (módulo), 6
Prestamo.MultipleObjectsReturned, 6
prestamo.views (módulo), 6

R

RegisteredUserId (clase en users.models), 9
RegisteredUserId.DoesNotExist, 9
RegisteredUserId.MultipleObjectsReturned, 9
Reserva (clase en reserva.models), 4
Reserva.DoesNotExist, 4
reserva.forms (módulo), 4
reserva.models (módulo), 3
Reserva.MultipleObjectsReturned, 4
reserva.views (módulo), 3
ReservaForm (clase en reserva.forms), 4
ReservaLibros (clase en reserva.views), 3
Reservar() (en el módulo reserva.views), 3
Revista (clase en revista.models), 7
Revista.DoesNotExist, 7
revista.forms (módulo), 7
revista.models (módulo), 7
Revista.MultipleObjectsReturned, 7
revista.views (módulo), 7
RevistaVista (clase en revista.views), 7

S

save() (método de libro.models.Libro), 6
save() (método de revista.models.Revista), 7
SignUp (clase en users.views), 8

T

Tesis (clase en tesis.models), 8
Tesis.DoesNotExist, 8

tesis.forms (módulo), 8
tesis.models (módulo), 8
Tesis.MultipleObjectsReturned, 8
tesis.views (módulo), 7
TesisVista (clase en tesis.views), 7

U

users.forms (módulo), 9
users.models (módulo), 8
users.views (módulo), 8