
AniShare Documentation

Release 1.4

Holger Dinkel, Fabian Monheim

Jul 19, 2018

Contents:

1	Introduction	1
2	Contact	2
3	Using the software	3
3.1	Main user interface	3
3.1.1	Animals	3
3.1.2	Organs	4
3.1.3	RSS Feed	5
3.2	Main animal manager tasks	6
3.2.1	Add Animals manually	6
3.2.2	Animals import	7
3.2.3	Add Organs manually	9
3.2.4	Organs import	9
3.2.5	Duplicating entries	9
3.3	Main administrator tasks	10
3.3.1	Organs used	10
3.3.2	Animals	10
3.3.3	Labs	11
3.3.4	Locations	12
3.3.5	Persons	12
3.3.6	Make a user an animal manager	13
3.3.7	Anishare change history	13
4	Installation	14
4.1	Requirements	14
4.2	First time setup	14
4.3	Importing existing data	16
4.4	In-DB Caching	16
4.5	Running Tests	16
4.5.1	Upgrading django	17
4.5.2	Upgrading python	17
5	API documentation	19
5.1	Admin	19
5.2	Models	20
5.3	Views	22
5.4	URLs	23
	Python Module Index	24

Chapter 1

Introduction

anishare is a webservice for research institutes to share animals with the goal to re-use animals and thus minimize total animal usage.

It has been developed at the [Leibniz institute for aging research](#) in Jena. This django app is meant to be used by researchers who want to share research animals with their colleagues. The basic idea is that animals are bred for experiments; however, sometimes, not all parts of the animal are used or sometimes an experiment gets cancelled for whatever reason. By sharing animals within the institute, less animals in total have to be sacrificed for research.

Anishare is a simple database of animals offered for reuse and a easy way to claim an animal with automatic generation of email messages as well as an RSS feed for updates.

AniShare

Animals

Organs

RSS

Feed

Logged in as: admin

Logout

Add Animal

AniShare Animals

Animal type: Sex: Day of birth: Line: Mutations: Location: Responsible person:

Added by:

Filter

Show

10

 entries

ID	#	Type	Sex	Date of Birth	Age (w)	Available From/To	Line	Mutations	Location	License#	ext. Info	Resp. Person	Added By	New Owner
4	2	mouse	♀	01.03.2018	10	01.05.2018 16.05.2018	ko/ko		Animal House 2	#154253	695968 LAB3584	Mike Smith (Lab A Lab)		no longer available
3	10	fish	?	06.05.2018	4	01.06.2018 30.06.2018	some line		Fish Facility	#154634	248962 LAB648	Fishy McFishface (Fish Lab)		Claim!
2	1	mouse	♀	01.03.2018	14	29.05.2018 29.06.2018	ki/ki		Animal House 2	#900238	159758 LAB3584	Roger Roger (Lab B Lab)		Claim!
1	2	mouse	♂	01.01.2018	22	29.05.2018 30.06.2018	ko/ko		Animal House 1	#900238	274628 LAB648	Mike Smith (Lab A Lab)		Claim!

Showing 1 to 4 of 4 entries

Previous

1

Next

At the moment, the software/database is geared towards handling of mice, however, it can be adjusted to handle any kind of research animal.

This documentation can also be downloaded as pdf file: [Anishare Documentation](#)

Chapter 2

Contact

Technical and application support: Fabian Monheim, fabian.monheim@leibniz-fli.de, 03641-65-6872

Content support: mouse@leibniz-fli.de, fish@leibniz-fli.de

Chapter 3

Using the software

AniShare

AnimalsAll AnimalsOrgans

Search for animals

Search

RSS

Feed

Logged in as:

admin

Logout

Add Animal

AniShare Animals

ID ↕	# ↕	Type ↕	Sex ↕	Date of Birth ↕	Age (w) ↕	Available From/To ↕	Line ↕	Mutations ↕	Location ↕	License# ↕	ext. Info ↕	Resp. Person ↕	Added By ↕	New Owner ↕
5	1	mouse	♂	01.01.2018	22	01.06.2018 01.07.2018	ko/ko		Outside	#007	ext. ID 123 LAB123	Roger Roger (Lab B Lab)	test user	Claim!
3	10	fish	?	06.05.2018	4	01.06.2018 30.06.2018	some line		Fish Facility	#154634	248962 LAB648	Fishy McFishface	test user	Claim!

The webservice is split in two parts: The **animal input** method is via the Django Admin interface (See link “Add Animal” top right) and is meant for *animal managers* only. The **claim** method is via the normal web interface and is meant for normal users (who need to be authenticated, though).

3.1 Main user interface

3.1.1 Animals

The main user-facing site is the list of animals to be shared. A user can browse this list, sort it via the headers or search for a term using the search bar.

AniShare
Animals
Organs
RSS
Feed
Logged in as: admin
Logout
Add Animal

AniShare Animals

Animal type: Sex: Day of birth: Line: Mutations: Location: Responsible person:

Added by:

Show entries

ID ↕	# ↕	Type ↕	Sex ↕	Date of Birth ↕	Age (w) ↕	Available From/To ↕	Line ↕	Mutations ↕	Location ↕	License# ↕	ext. Info ↕	Resp. Person ↕	Added By ↕	New Owner ↕
4	2	mouse	♀	01.03.2018	10	01.05.2018 16.05.2018	ko/ko		Animal House 2	#154253	695968 LAB3584	Mike Smith (Lab A Lab)		no longer available
3	10	fish	?	06.05.2018	4	01.06.2018 30.06.2018	some line		Fish Facility	#154634	248962 LAB648	Fishy McFishface (Fish Lab)		Claim!
2	1	mouse	♀	01.03.2018	14	29.05.2018 29.06.2018	ki/ki		Animal House 2	#900238	159758 LAB3584	Roger Roger (Lab B Lab)		Claim!
1	2	mouse	♂	01.01.2018	22	29.05.2018 30.06.2018	ko/ko		Animal House 1	#900238	274628 LAB648	Mike Smith (Lab A Lab)		Claim!

Showing 1 to 4 of 4 entries

If a user is interested in an animal, they should click on the button “Claim” which will bring up another page (see below) in which they can review their claim before finally submitting. When they click on “Yes, I want to claim this!”, then they will be entered as *new owner* of this animal and an email will be send to them as well as the responsible/contact person informing them about this transaction. Further steps might need to be necessary such as transferring the animal in the LIMS (eg. PyRat).

Note: If more than one animal is available, the user can adjust the number they want to claim. The remaining animals will still be available for claim. Because of uniqueness it’s only possible to offer exactly one mouse per dataset. In contrast fishes can be offer in a group.

AniShare
Animals
Organs
RSS
Feed
Logged in as: hdinkel
Logout
Add Animal

Dear **Holger Dinkel**, do you really want to claim the following animal for yourself?

ID	#	Type	Sex	Entry Date	Date of Birth	Age (w)	Available From	Available To	Line	Mutations	Location	License#	ext. Info	Resp. Person	Added By	Commer
3	10	fish	?	30.05.2018	06.05.2018	4	01.06.2018	30.06.2018	some line		Fish Facility	#154634	248962 LAB648	Fishy McFishface (Fish Lab)		multiple animals available

By clicking on the button, an email will be sent from
Holger.Dinkel@leibniz-fli.de to
Fishy McFishface (fishymcfishface@nowhere.com),
informing that you take responsibility for out of 10 animals from entry .

They will make the necessary adjustments in the database.

3.1.2 Organs

There exists an individual page for animal organ share. It is very similar to the animal page, however only individual organs are for offer. The entry at the column *Organ (used)* indicates all organs which can not be claimed. Also there is no availability period, but a day at which the animal gets sacrificed. The person responsible for sacrifice the

animal will be informed via email if anybody claims some of the available organs. The entry will remain available to others (as they might want to claim other organs).

Organ index view:

AniShare
Animals
Organs
RSS
Feed
Logged in as: hdinkel
Logout
Add Animal

AniShare Organs

Animal type: Sex: Day of birth: Day of death: Killing person: Method of killing: Line:

Mutations: Location: Licence number:

ID	#	Type	Sex	Date of Birth	Date of Death	Killing Method	Killing Person	Age (w)	Line	Mutations	Location	License#
3	2	fish	?	10.02.2018	09.06.2018	CO2	hannibal.lecter@nowhere.com	17	no line		Animal House 2	007
2	1	mouse	♂	10.03.2018	05.06.2018	CO2	hannibal.lecter@nowhere.com	12	no line	none	Animal House 2	007
1	1	mouse	♀	14.03.2018	09.06.2018	CO2	hannibal.lecter@nowhere.com	12	no line		Animal House 2	007

Showing 1 to 3 of 3 entries

Organ claim view:

AniShare
Animals
Organs
RSS
Feed
Logged in as: hdinkel
Logout
Add Animal

Dear **Holger Dinkel**, do you really want to claim an **organ** of the following animal for yourself?

ID	#	Type	Organ	Sex	Date of Birth	Date of Death	Method of Killing	Killing Person	Age (w)	Line	Mutations	Location	License#	ext. Info	Added By
1	1	mouse	brain	♀	14.03.2018	09.06.2018	CO2	hannibal.lecter@nowhere.com	12	no line		Animal House 2	007	321658 59135	

By clicking on the button, an email will be sent from
 Holger.Dinkel@leibniz-fli.de to
hannibal.lecter@nowhere.com
 informing that you are interested in the following organs/parts
 liver from the entry above.
 They will get in touch with you.

3.1.3 RSS Feed

An RSS feed containing the latest ten animals and organs is automatically generated and can be found at */animals/feed*. Users can subscribe (Most email clients allow the subscription to RSS feeds) to this feed to stay up-to-date with the animal catalogue. By clicking on a link in the feed, they are directed to the claim page of the individual animal/organ.

Anishare animal feed

Updates on animals to share.

[10 unknown fish, some line id:3 \[2018-05-06\]](#)

id:248962, lab_id:LAB648, available:2018-06-01-2018-06-30, location:Fish Facility, mutations:

[2 female mouse, ko/ko id:4 \[2018-03-01\]](#)

id:695968, lab_id:LAB3584, available:2018-05-01-2018-05-16, location:Animal House 2, mutations:

[2 male mouse, ko/ko id:1 \[2018-01-01\]](#)

id:274628, lab_id:LAB648, available:2018-05-29-2018-06-30, location:Animal House 1, mutations:

[1 female mouse, kj/ki id:2 \[2018-03-01\]](#)

id:159758, lab_id:LAB3584, available:2018-05-29-2018-06-29, location:Animal House 2, mutations:

3.2 Main animal manager tasks

An *animal manager* can add animals and organs to the database on two ways. First, it is possible to add entries manually. Secondly, it is possible to import an Excel sheet. At the FLI Jena there are two databases to manage animals. Now the databases are not connected. To transfer more than one or two datasets to anishare it's recommend to use the export function of *PyRAT* or *tick@lab* and the import function of anishare. To use the export/import process please read the topic **Animals import** or **Organs import**.

Welcome to AniShare

ANIMALS		
Animals	+ Add	✎ Change
Organs	+ Add	✎ Change
Persons		✎ Change

3.2.1 Add Animals manually

Click on Animals -> Add to add an animal.

Add animal

Amount: How many animals? (eg. fish in tank)

Animal type:

Organ type:

Day of birth: Today

Available from: Today

Available to: Today

Sex: ☐ male ☒ female ☐ unknown
Select "unknown" if multiple animals.

External id:

External lab id:

Line:

Location: Where is the animal housed?

Responsible person: Person who is responsible in the lab for dealing with the animals

Licence number:

All fields in bold **need** to be filled in, the others are optional.

After adding several animals, the main (index) view should look like this:

AniShare admin interface
WELCOME, ADMIN. VIEW SITE / DOCUMENTATION / CHANGE PASSWORD / LOG OUT

Home › Animals › Animals

Select animal to change

Search

Action: Go 0 of 4 selected

AMOUNT	ENTRY DATE	DAY OF BIRTH	AGE (W)	AVAILABLE FROM	AVAILABLE TO	LINE	SEX	LOCATION	LICENCE NUMBER	RESPONSIBLE PERSON	ADDED BY
2	30.05.2018	01.03.2018	10	01.05.2018	16.05.2018	ko/ko	female	Animal House 2	#154253	Mike Smith (Lab A Lab)	admin
10	30.05.2018	06.05.2018	3	01.06.2018	30.06.2018	some line	unknown	Fish Facility	#154634	Fishy McFishface (Fish Lab)	admin
1	29.05.2018	01.03.2018	12	29.05.2018	29.06.2018	ki/ki	female	Animal House 2	#900238	Roger Roger (Lab B Lab)	admin
2	29.05.2018	01.01.2018	21	29.05.2018	30.06.2018	ko/ko	male	Animal House 1	#900238	Mike Smith (Lab A Lab)	admin

4 animals

FILTER

By amount
All
1
2
10

By sex
All
male
female
unknown

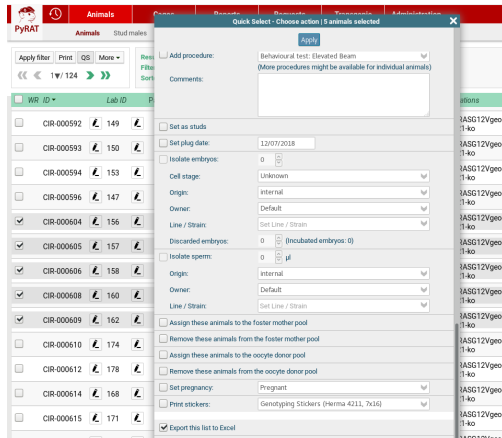
By responsible for lab
All
Lab A Lab
Lab B Lab
Fish Lab

By location
All
Animal House 1
Outdoors
Animal House 2
Fish Facility

3.2.2 Animals import

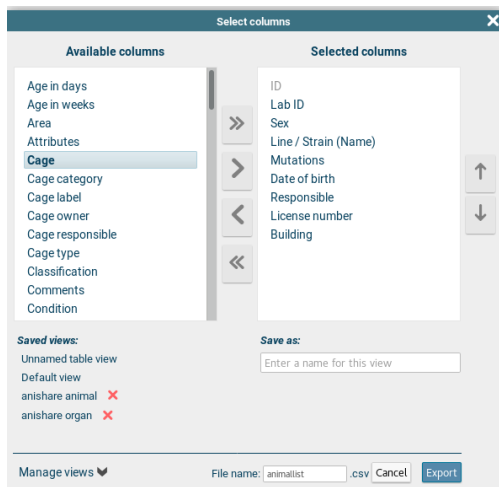
From PyRAT

First login to PyRAT and switch to the english version of PyRAT if it is no preset. Then select the animals which should be import to anishare. Click on QS (Quick Select) and activate the option `Export this list to Excel`. Push the button `Apply`.



Now it's important to select all mandatory fields: **ID, Lab ID, Sex, Line / Strine (Name), Mutations, Date of birth, Responsible, License number, Building**

It's possible to save the selected columns as a `Manage View` for reuse (fold out `Manage View` on the left side of the `File name`)



After downloading the file it's necessary to edit the file because the columns **Animal type**, **Available from**, **Available to** are missing. Please add the mentioned columns (wherever) and fill it out. As **Animal type** the two values `mouse` or `fish` are possible. Please use the same date format for the columns **Available from**, **Available to** like at the exported column **DOB** (Date of birth). To simplify this process there are macros for LibreOffice and MS Office. The macros automatically add the missing columns and add the values `mouse` (Animal type), `Current Date` (Available from), `Current Date + 14 days` (Available to). Please refer to the [macro site](#) to downloading the macros and further informations.




After adding the missing columns the file can be save, for example as `xlsx` file. Now go to the anishare admin interface to `Home > Animals > Animals` and click the button `IMPORT` (above the filter). Select the file and choose the file format. Upload the file. After submitting all datasets will show to the user if all requirements match.

From *tick@lab*

3.2.3 Add Organs manually

Click on Organs -> Add to add an organ.

Add organ

Amount:	<input type="text" value="1"/>		How many organs?
Animal type:	<input type="text" value="mouse"/>		
Sex:	<input type="radio"/> male <input type="radio"/> female <input type="radio"/> unknown Select "unknown" if multiple animals.		
Organ type:	<input type="text" value="....."/>		
Day of birth:	<input type="text"/>	Today	
Day of death:	<input type="text"/>	Today	
Method of killing:	<input type="text" value="....."/>		
Killing person:	<input type="text"/>		
	Email address of the person who is responsible for killing the animal		
Database id:	<input type="text"/>		
	ID of animal in eg. PYRAT		
Lab id:	<input type="text"/>		
	ID of lab in eg. PYRAT		
Line:	<input type="text"/>		
	genetic trait of animal		
Location:	<input type="text" value="....."/>		
	Where is the animal housed?		

All fields in bold **need** to be filled in, the others are optional.

3.2.4 Organs import

From PyRAT

From *tick@lab*

Because we expect only a small quantity of importing organs from *tick@lab* it's only possible to add entries manually.

3.2.5 Duplicating entries

For input of multiple similar entries, it is possible to duplicate an animal or organ entry. For this, select one or more entries in the list (see figure below) and select “*copy animal*” from the dropdown menu and click “*Go*”.

Select animal to change

Q Search

Action: ☒ ----- Go 1 of 100 selected
Delete selected animals
Clear claim
Copy animal

	AMN	BIRTH	AGE (W)	AVAILABLE FROM	AVAILABLE TO	LINE	SEX	LOCATION	LICENCE NUMBER	
<input checked="" type="checkbox"/>	1	06.06.2018	01.01.2018	22	01.06.2018	01.07.2018	ko/ko	male	BIZ	#007

Another option is to edit an existing animal and click on “*Save as new*”. This will save the currently edited animal as a new instance:

Change animal HISTORY VIEW ON SITE >

Delete
Save as new
Save and continue editing
SAVE

Amount: How many animals? (eg. fish in tank)

Animal type:

Day of birth: Today

Available from: Today

Available to: Today

Sex: ☒ male ☐ female ☐ unknown
Select "unknown" if multiple animals.

Database id: ID of animal in eg. PYRAT

Lab id: ID of lab in eg. PYRAT

Line: genetic trait of animal

3.3 Main administrator tasks

The administrator can edit more objects in the admin interface, namely not just animals and organs but also labs, locations and persons:

Welcome to AniShare

ANIMALS		
Animals	+ Add	✎ Change
Labs	+ Add	✎ Change
Locations	+ Add	✎ Change
Organs	+ Add	✎ Change
Persons	+ Add	✎ Change

3.3.1 Organs used

These organs are standard values for the field **Organ used**.

3.3.2 Animals

The main category to administer are animals to share. Here, several filters (such as “sex”, “location”, etc.) are available to search for any set of animals.

Select animal to change

ADD ANIMAL +

Action: ----- 0 of 5 selected

<input type="checkbox"/>	AMOUNT	ENTRY DATE	DAY OF BIRTH	AGE (W)	AVAILABLE FROM	AVAILABLE TO	LINE	SEX	LOCATION	LICENCE NUMBER	RESPONSIBLE PERSON	AD
<input type="checkbox"/>	1	31.05.2018	01.01.2018	21	01.06.2018	01.07.2018	ko/ko	male	Outside	#007	Roger Roger (Lab B Lab)	test
<input type="checkbox"/>	2	30.05.2018	01.03.2018	10	01.05.2018	16.05.2018	ko/ko	female	Animal House 2	#154253	Mike Smith (Lab A Lab)	test
<input type="checkbox"/>	10	30.05.2018	06.05.2018	3	01.06.2018	30.06.2018	some line	unknown	Fish Facility	#154634	Fishy McFishface (Fish Lab)	test
<input type="checkbox"/>	1	29.05.2018	01.03.2018	13	29.05.2018	29.06.2018	ki/ki	female	Animal House 2	#900238	Roger Roger (Lab B Lab)	test
<input type="checkbox"/>	2	29.05.2018	01.01.2018	21	29.05.2018	30.06.2018	ko/ko	male	Animal House 1	#900238	Mike Smith (Lab A Lab)	test

5 animals

FILTER

By amount

All
1
2
10

By sex

All
male
female
unknown

By responsible for lab

All
Lab A Lab
Lab B Lab
Fish Lab

By location

All
Animal House 1
Outdoors
Animal House 2
Fish Facility
Outside

Note: in order to remove a claim (thus making the animal available again), either click on an animal and remove the email address from the field “new owner”, or select one or multiple animals and select the “clear claim” *Action* and click “Go”.

Note: Once created, an animal cannot be deleted, except by the administrator.

3.3.3 Labs

Labs are research labs/research groups and need to have at least one responsible/contact person each

Select lab to change

Action: ----- 0 of 3 selected

<input type="checkbox"/>	NAME	RESPONSIBLE PERSON
<input type="checkbox"/>	Fish	Fishy McFishface
<input type="checkbox"/>	Lab B	Roger Roger
<input type="checkbox"/>	Lab A	Mike Smith

3 labs

Note: Only *administrators* are allowed to see and change Labs

3.3.4 Locations

Locations are where animals are stored. Usually something like room numbers or “animal house” or “fish facility”.

Select location to change

Action: 0 of 5 selected

<input type="checkbox"/>	NAME
<input type="checkbox"/>	Outside
<input type="checkbox"/>	Fish Facility
<input type="checkbox"/>	Animal House 2
<input type="checkbox"/>	Outdoors
<input type="checkbox"/>	Animal House 1

5 locations

Note: Only *administrators* are allowed to see and change Locations

3.3.5 Persons

Persons responsible for the animals. Could be a vet or similar. Every animal needs to have a responsible person associated to them. This person then gets an email when the animal is being claimed.

Select person to change

Action: 0 of 3 selected

<input type="checkbox"/>	NAME	EMAIL	RESPONSIBLE FOR LAB
<input type="checkbox"/>	Fishy McFishface	fishymcfishface@nowhere.com	Fish Lab
<input type="checkbox"/>	Mike Smith	mike.smith@nowhere.com	Lab A Lab
<input type="checkbox"/>	Roger Roger	roger.roger@nowhere.com	Lab B Lab

3 persons

Note: Only *administrators* are allowed to see and change Persons

3.3.6 Make a user an animal manager

The *administrator* is also responsible for user/rights management. In order to be able to add/edit animals, a user has to be in the group *animal manager* and have *staff status* in the django admin interface. For this, an *administrator* has to go to the [user management](#) in the admin interface by clicking “Home” -> “Authentication and Authorization” -> “Users”. Here, they can make a *user* an *animal manager*, by setting these values (*staff* and group *animal manager*):

Permissions

☒ **Active**
Designates whether this user should be treated as active. Unselect this instead of deleting accounts.

☒ **Staff status**
Designates whether the user can log into this admin site.

☐ **Superuser status**
Designates that this user has all permissions without explicitly assigning them.

Groups:

Available groups ⓘ

Q Filter

Chosen groups ⓘ

animal manager

Remove all

Choose all ⓘ

The groups this user belongs to. A user will get all permissions granted to each of their groups. Hold down "Control", or "Command" on a Mac, to select more than one.

3.3.7 Anishare change history

New functions and bugfix at the system should be documented. So users can be informed about changes on the system. All changes are visible to authenticated users. Please refer to the site [AniShare Change History](#) to see all changes.

Chapter 4

Installation

4.1 Requirements

We use the latest version of [django](#), which requires [python3](#). Install django and other dependencies (see file `requirements.txt`). We recommend using a virtual environment for this):

```
virtualenv -p python3 .
source bin/activate
pip install -r requirements.txt
```

4.2 First time setup

First, in the folder `anishare`, copy the file `local_settings.py.template` to `local_settings.py` and fill it in. If you want to use LDAP, comment in the respective lines. Most importantly, you should configure the following lines:

```
EMAIL_HOST = ''
SECRET_KEY = ''
ALLOWED_HOSTS = ['127.0.0.1', ]
```

Then, you can run migrations:

```
python manage.py migrate
```

Note: This will create the sqlite database `db.sqlite3` containing all the models (eg. tables) as defined in `animals.models`.

Now create a superuser:

```
python manage.py createsuperuser
```

You are now able to login to the admin interface, but first run the dev server:

```
python manage.py runserver
```

This will listen on `http://localhost:8000`, so browse to the admin page `http://localhost:8000/admin` and you should see this after login:

AniShare admin interface

Welcome to AniShare

ANIMALS

Animals

+ Add

Change

Labs

+ Add

Change

Locations

+ Add

Change

Organs

+ Add

Change

Persons

+ Add

Change

AUTHENTICATION AND AUTHORIZATION

Groups

+ Add

Change

Users

+ Add

Change

Recent actions
My actions

1 female mouse, no line id:1
[2018-03-14]
Organ

+ 1 female mouse, no line id:1
[2018-03-14]
Organ

testuser
User

admin
User

You can also import a dummy set of data using the `loaddata` command:

```
python manage.py loaddata initial_data.json
```

After loading the data, the main admin interface should look like this:

AniShare admin interface

WELCOME, ADMIN. VIEW SITE / DOCUMENTATION / CHANGE PASSWORD / LOG OUT

Home > Animals > Animals

Select animal to change

Q

Search

Action:

Go

0 of 4 selected

AMOUNT	ENTRY DATE	DAY OF BIRTH	AGE (W)	AVAILABLE FROM	AVAILABLE TO	LINE	SEX	LOCATION	LICENCE NUMBER	RESPONSIBLE PERSON	ADDED BY
2	30.05.2018	01.03.2018	10	01.05.2018	16.05.2018	ko/ko	female	Animal House 2	#154253	Mike Smith (Lab A Lab)	admin
10	30.05.2018	06.05.2018	3	01.06.2018	30.06.2018	some line	unknown	Fish Facility	#154634	Fishy McFishface (Fish Lab)	admin
1	29.05.2018	01.03.2018	12	29.05.2018	29.06.2018	ki/ki	female	Animal House 2	#900238	Roger Roger (Lab B Lab)	admin
2	29.05.2018	01.01.2018	21	29.05.2018	30.06.2018	ko/ko	male	Animal House 1	#900238	Mike Smith (Lab A Lab)	admin

4 animals

ADD ANIMAL

+

FILTER

By amount

All

1

2

10

By sex

All

male

female

unknown

By responsible for lab

All

Lab A Lab

Lab B Lab

Fish Lab

By location

All

Animal House 1

Outdoors

Animal House 2

Fish Facility

4.3 Importing existing data

For import of existing data in tabular (excel) format, a management command is available at `animals.management.commands.import_animals`

```
python manage.py import_animals
```

Note: See the file `example_import.xls` for an example...

Information exported from PYRAT:										Information to add manually			
Nr.	Entry date	ID	Lab ID	DOB	Age (w)	Line / Strain	Sex	Building	Licence num	Responsible	Available from	Available until	New Owner
1	01.05.18	ext. ID 123	LAB123	01.01.18	10	ko/ko	m	Outside	#007	Roger Roger	01.06.18	01.07.18	

4.4 In-DB Caching

By default, database caching is enabled in settings. To create the necessary tables, run this command:

```
python manage.py createcachetable
```

This will create a cache table in the SQLite database, which will speed up queries.

4.5 Running Tests

Tests reside in `animals/tests.py`. You can invoke the django tests like so:

```
python manage.py test
```

```
1  """
2  Tests for Anishare website
3  """
4  from django.test import TestCase, Client
5  from django.contrib.auth.models import User
6  from django.core.management import call_command
7
8  class GetAnimalsTest(TestCase):
9      """
10     Test module to GET Animal pages
11     """
12
13     def setUp(self):
14         """ Creating a user first and loading fixtures """
15         call_command('loaddata', 'initial_data.json', verbosity=0) # Load fixtures
16         self.user = User.objects.create_user(pk=1, username='testuser', password=
17 ↪ '12345')
18         self.client = Client()
19
20     def test_get_all_animals(self):
21         """ try to retrieve all animals """
22         response = self.client.get('/animals/')
23         self.assertEqual(response.status_code, 302)
```

(continues on next page)

(continued from previous page)

```

23     self.client.login(username='testuser', password='12345')
24     response = self.client.get('/animals/')
25     self.assertEqual(response.status_code, 200)
26
27     def test_get_one_animal(self):
28         """ try to retrieve individual animals """
29         response = self.client.get('/animals/1')
30         self.assertEqual(response.status_code, 302)
31         self.client.login(username='testuser', password='12345')
32         response = self.client.get('/animals/1')
33         self.assertEqual(response.status_code, 200)
34
35     def test_claim_one_animal(self):
36         """ try to claim individual animals """
37         response = self.client.get('/animals/claim/1')
38         self.assertEqual(response.status_code, 302)
39         self.client.login(username='testuser', password='12345')
40         response = self.client.get('/animals/claim/1')
41         self.assertEqual(response.status_code, 200)

```

4.5.1 Upgrading django

To upgrade django or any other python library for anishare, go into the anishare directory, and activate its virtualenv:

```

cd anishare
source bin/activate

```

Next, install/upgrade whatever library (here: django to the latest version):

```

pip install --upgrade django

```

Note: It's best to test the latest version in a local/development environment first!

4.5.2 Upgrading python

When upgrading the python version of the host operating system, it might be necessary to also upgrade the python in the virtualenv. Otherwise an error like the following might occur:

```

python: error while loading shared libraries: libpython3.4m.so.1.0:
cannot open shared object file: No such file or directory

```

In that case, go into the anishare directory, and delete the following directories:

- bin
- include
- lib
- lib64

Afterwards, create a new virtualenv and install the required libraries like so:

```
virtualenv -p python3 .  
source bin/activate  
pip install -r requirements.txt
```

API documentation

5.1 Admin

Admin module

```
class animals.admin.AnimalAdmin (model, admin_site)
    ModelAdmin for Animal model
```

```
    age (obj)
        Show the age in the admin as 'Age (w)' instead of 'age'
```

```
    form
        alias of AnimalForm
```

```
    resource_class
        alias of AnimalResource
```

```
    save_model (request, obj, form, change)
        Given a model instance save it to the database.
```

```
class animals.admin.AnimalForm (data=None, files=None, auto_id='id_%s', prefix=None, initial=None, error_class=<class 'django.forms.utils.ErrorList'>, label_suffix=None, empty_permitted=False, instance=None, use_required_attribute=None)
```

Form for animal editing in admin

```
    clean ()
        Hook for doing any extra form-wide cleaning after Field.clean() has been called on every field. Any ValidationError raised by this method will not be associated with a particular field; it will have a special-case association with the field named '__all__'.
```

```
class animals.admin.AnimalResource
```

```
    get_instance (instance_loader, row)
        Calls the InstanceLoader.
```

```
    import_obj (instance, row, dry_run)
        Traverses every field in this Resource and calls import_field().
```

```
class animals.admin.ChangeAdmin (model, admin_site)
    ChangeAdmin for Change model
```

```

class animals.admin.LabAdmin(model, admin_site)
    ModelAdmin for Lab model

class animals.admin.LocationAdmin(model, admin_site)
    ModelAdmin for Location model

class animals.admin.OrganAdmin(model, admin_site)
    ModelAdmin for Organ model

    age(obj)
        Show the age in the admin as 'Age (w)' instead of 'age'

    resource_class
        alias of OrganResource

    save_model(request, obj, form, change)
        Given a model instance save it to the database.

class animals.admin.OrganResource

    get_instance(instance_loader, row)
        Calls the InstanceLoader.

    import_obj(instance, row, dry_run)
        Traverses every field in this Resource and calls import_field().

class animals.admin.OrgantypeAdmin(model, admin_site)
    ModelAdmin for Organ types

class animals.admin.PersonAdmin(model, admin_site)
    ModelAdmin for Person model

animals.admin.clear_claim(modeladmin, request, queryset)
    Convenience Function to delete a claim from several selected animals

animals.admin.copy_animal(modeladmin, request, queryset)
    Copy an instance of an animal so similar entries can be easily created.

```

5.2 Models

This file describes all the models in the database.

```

class animals.models.Animal(*args, **kwargs)
    Main model containing the animals.

    exception DoesNotExist

    exception MultipleObjectsReturned

    age()
        Return the age of the animal, calculated by the difference to either the current date or the available_to date

    available()
        Returns True if the animal is still available

    clean()
        Hook for doing any extra model-wide validation after clean() has been called on every field by
        self.clean_fields. Any ValidationError raised by this method will not be associated with a particular field;
        it will have a special-case association with the field defined by NON_FIELD_ERRORS.

```

```
description ()
    Return description of this model

get_absolute_url ()
    Get absolute url for this model. Important to link from the admin.

class animals.models.Change (*args, **kwargs)
    Model for documentation all changes to anishare

    exception DoesNotExist

    exception MultipleObjectsReturned

class animals.models.Lab (*args, **kwargs)
    Labs are only defined by a name and are referenced by Person(s) which are responsible (contact) person for this
    lab

    exception DoesNotExist

    exception MultipleObjectsReturned

    responsible_person ()
        Retrieve only the person(s) which are responsible for this lab.

class animals.models.Location (*args, **kwargs)
    Location of animals. Eg. animal house, fish facilities etc.

    exception DoesNotExist

    exception MultipleObjectsReturned

class animals.models.Organ (*args, **kwargs)
    Model containing the organs

    exception DoesNotExist

    exception MultipleObjectsReturned

    age ()
        Return the age of the animal, at the time of death

    available ()
        Returns True if the animal is still available

    description ()
        Return description of this model

    get_absolute_url ()
        Get absolute url for this model. Important to link from the admin.

    get_organotypes ()
        Get all organ types which are used

class animals.models.Organtype (*args, **kwargs)
    Model containing the organ types

    exception DoesNotExist

    exception MultipleObjectsReturned

class animals.models.Person (*args, **kwargs)
    The responsible (contact) person for each lab. This person gets an email when an animal is being claimed.

    exception DoesNotExist

    exception MultipleObjectsReturned
```


5.3 Views

Django Views contains all the functions for rendering objects (HTML display). It also contains an RSS Feed generator class to create an RSS feed from newly created animals

Important: When adding new functions, use the `login_required` decorator When adding new classes, use the `LoginRequiredMixin`

```
class animals.views.AnimalDetailView (**kwargs)
    Detail view for an animal. This is rarely used, rather use the claim page.

    model
        alias of animals.models.Animal
```

```
class animals.views.LatestAnimalsFeed
    RSS Feed for new animals/organs.

    item_description (item)
        What to print as item description (use default description from model).

    item_title (item)
        What to print as item title (use default __str__ of model).

    items ()
        Get latest animals as items.
```

```
animals.views.claim (request, primary_key)
    View to claim an animal.

    Parameters primary_key – the id/pk of the animal to retrieve

    Returns rendered page with the claim form or 404 if animal not found
```

```
animals.views.claim_organ (request, primary_key)
    View to claim an organ.

    Parameters primary_key – the id/pk of the organ to retrieve

    Returns rendered page with the claim form or 404 if organ not found
```

```
animals.views.send_email_animal (request)
    Function to send an email about an animal being claimed.

    Needs these variables in the POST request: email, pk, count

    Parameters
        • email – email address of the request user / new owner
        • pk – primary_key of the animal(s) to be claimed
        • count – how many animals are being claimed
```

```
animals.views.send_email_organ (request)
    Function to send an email about an animal being claimed.

    Needs these variables in the POST request: email, pk, count

    Parameters
        • email – email address of the request user / new owner
        • pk – primary_key of the animal(s) to be claimed
        • organs_wanted – organs wanted from the given animal
```

5.4 URLs

animals URL Configuration

The *urlpatterns* list routes URLs to views. For more information please see: <https://docs.djangoproject.com/en/2.0/topics/http/urls/>

Examples:

Function views:

1. Add an import: from my_app import views
2. Add a URL to urlpatterns: `path("", views.home, name='home')`

Class-based views:

1. Add an import: from other_app.views import Home
2. Add a URL to urlpatterns: `path("", Home.as_view(), name='home')`

Including another URLconf:

1. Import the `include()` function: from `django.urls` import `include`, `path`
2. Add a URL to urlpatterns: `path('blog/', include('blog.urls'))`

Python Module Index

a

`animals.models`, [20](#)
`animals.urls`, [23](#)
`animals.views`, [22](#)

Index

A

`age()` (animals.admin.AnimalAdmin method), 19
`age()` (animals.admin.OrganAdmin method), 20
`age()` (animals.models.Animal method), 20
`age()` (animals.models.Organ method), 21
`Animal` (class in animals.models), 20
`Animal.DoesNotExist`, 20
`Animal.MultipleObjectsReturned`, 20
`AnimalAdmin` (class in animals.admin), 19
`AnimalDetailView` (class in animals.views), 22
`AnimalForm` (class in animals.admin), 19
`AnimalResource` (class in animals.admin), 19
`animals.admin` (module), 19
`animals.models` (module), 20
`animals.urls` (module), 23
`animals.views` (module), 22
`available()` (animals.models.Animal method), 20
`available()` (animals.models.Organ method), 21

C

`Change` (class in animals.models), 21
`Change.DoesNotExist`, 21
`Change.MultipleObjectsReturned`, 21
`ChangeAdmin` (class in animals.admin), 19
`claim()` (in module animals.views), 22
`claim_organ()` (in module animals.views), 22
`clean()` (animals.admin.AnimalForm method), 19
`clean()` (animals.models.Animal method), 20
`clear_claim()` (in module animals.admin), 20
`copy_animal()` (in module animals.admin), 20

D

`description()` (animals.models.Animal method), 20
`description()` (animals.models.Organ method), 21

F

`form` (animals.admin.AnimalAdmin attribute), 19

G

`get_absolute_url()` (animals.models.Animal method), 21
`get_absolute_url()` (animals.models.Organ method), 21

`get_instance()` (animals.admin.AnimalResource method), 19
`get_instance()` (animals.admin.OrganResource method), 20
`get_organypes()` (animals.models.Organ method), 21

I

`import_obj()` (animals.admin.AnimalResource method), 19
`import_obj()` (animals.admin.OrganResource method), 20
`item_description()` (animals.views.LatestAnimalsFeed method), 22
`item_title()` (animals.views.LatestAnimalsFeed method), 22
`items()` (animals.views.LatestAnimalsFeed method), 22

L

`Lab` (class in animals.models), 21
`Lab.DoesNotExist`, 21
`Lab.MultipleObjectsReturned`, 21
`LabAdmin` (class in animals.admin), 19
`LatestAnimalsFeed` (class in animals.views), 22
`Location` (class in animals.models), 21
`Location.DoesNotExist`, 21
`Location.MultipleObjectsReturned`, 21
`LocationAdmin` (class in animals.admin), 20

M

`model` (animals.views.AnimalDetailView attribute), 22

O

`Organ` (class in animals.models), 21
`Organ.DoesNotExist`, 21
`Organ.MultipleObjectsReturned`, 21
`OrganAdmin` (class in animals.admin), 20
`OrganResource` (class in animals.admin), 20
`Organtype` (class in animals.models), 21
`Organtype.DoesNotExist`, 21
`Organtype.MultipleObjectsReturned`, 21
`OrgantypeAdmin` (class in animals.admin), 20

P

Person (class in `animals.models`), [21](#)

Person.DoesNotExist, [21](#)

Person.MultipleObjectsReturned, [21](#)

PersonAdmin (class in `animals.admin`), [20](#)

R

resource_class (animals.admin.AnimalAdmin attribute),
[19](#)

resource_class (animals.admin.OrganAdmin attribute),
[20](#)

responsible_person() (animals.models.Lab method), [21](#)

S

save_model() (animals.admin.AnimalAdmin method), [19](#)

save_model() (animals.admin.OrganAdmin method), [20](#)

send_email_animal() (in module `animals.views`), [22](#)

send_email_organ() (in module `animals.views`), [22](#)