

---

# **BeaverManager Documentation**

***Release 0.4***

**James Seden Smith**

April 11, 2016



<b>1</b>	<b>beaver_manager package</b>	<b>3</b>
1.1	Submodules . . . . .	3
1.2	beaver_manager.decorators module . . . . .	3
1.3	beaver_manager.email module . . . . .	3
1.4	beaver_manager.forms module . . . . .	3
1.5	beaver_manager.models module . . . . .	4
1.6	beaver_manager.views module . . . . .	8
1.7	beaver_manager.logic module . . . . .	11
<b>2</b>	<b>config module</b>	<b>13</b>
<b>3</b>	<b>API Requests</b>	<b>15</b>
<b>4</b>	<b>Indices and tables</b>	<b>21</b>
	<b>HTTP Routing Table</b>	<b>23</b>
	<b>Python Module Index</b>	<b>25</b>



Contents:



---

## beaver\_manager package

---

Beaver Management app using flask

`beaver_manager.app`

`Flask` – Flask application instance

`beaver_manager.db`

`SQLAlchemy` – SQLAlchemy database instance

### 1.1 Submodules

### 1.2 beaver\_manager.decorators module

`beaver_manager.decorators.async(f)`

Decorator that will make any function asynchronous

### 1.3 beaver\_manager.email module

Module to control email

`beaver_manager.email.send_email(subject, recipients, text_body, html_body)`

Sends email to recipient using the flask's email module

#### Parameters

- **subject** (*str*) – Plain text
- **recipients** (*str*) – Recipient email. Can be list of strings
- **text\_body** (*str*) – Body of message without HTML formatting
- **html\_body** (*str*) – Body of message with HTML formatting

### 1.4 beaver\_manager.forms module

```
class beaver_manager.forms.MultiCheckboxField(label=None, validators=None, coerce=<class 'str'>, choices=None, **kwargs)
```

Bases: `wtforms.fields.core.SelectMultipleField`

A multiple-select, except displays a list of checkboxes.

Iterating the field will produce subfields, allowing custom rendering of the enclosed checkbox fields.

## 1.5 beaver\_manager.models module

This module contains SQLAlchemy model definitions to create SQL tables

**class** `beaver_manager.models.Attendance` (\*\*kwargs)

Bases: `flask_sqlalchemy.Model`

A model for a meeting. Linked to a *Criterion* to allow automatic completion of badges. If beaver was present for the meeting then the criterion is completed.

**id**

*int* – Unique Primary Key.

**date**

*DateTime* – Date of meeting

**criterion\_id**

*int* – Foreign key for *Criterion*

**criterion**

*Criterion* – Provides a link to the *Criterion* that Attendance is associated with

**description**

*str* – Brief description of evening

**organiser**

*str* – Name of organiser for that meeting

**class** `beaver_manager.models.Badge` (\*\*kwargs)

Bases: `flask_sqlalchemy.Model`

Model for a Badge

**id**

*int* – Unique Primary Key.

**name**

*str* – Name of the badge

**img\_url**

*str* – URL for the badge image

**beavers\_badges**

list[*BeaverBadge*] – A list of beavers\_badges associated with the lodge

**class** `beaver_manager.models.BadgeCriterion` (*criterion\_id*, *badge\_id*, *completed*)

Bases: `flask_sqlalchemy.Model`

Model for a BadgeCriterion. Acts as an intermediary between *BeaverBadge* and *Criterion*

**Parameters**

- **criterion\_id** (*int*) – Foreign key for *Criterion*
- **badge\_id** (*int*) – Foreign key for *BeaverBadge*
- **completed** (*bool*) – Represents whether the criterion has been completed or not



**id**

*int* – Unique Primary Key.

**criterion**

*Criterion* – Provides a link to the *Criterion* that criterion is associated with

**badge**

*BeaverBadge* – Provides a link to the *BeaverBadge* that criterion is associated with

**description**

*str* – Description of criterion

**class** beaver\_manager.models.**Beaver** (\*\*kwargs)

Bases: flask\_sqlalchemy.Model

Model for a beaver

**id**

*int* – Unique Primary Key.

**first\_name**

*str* – The beaver’s first name

**surname**

*str* – The beaver’s surname

**dob**

*DateTime* – The beaver’s date of birth

**lodge\_id**

*int* – Foreign key for *Lodge* table

**lodge**

*Lodge* – Provides direct access to the *Lodge* that is that is linked to the beaver

**contacts**

list[*EmergencyContact*] – A list of *EmergencyContact* objects associated with the beaver

**badges**

list[*BeaverBadge*] – A list of *BeaverBadge* associated with the beaver

**beaver\_attendances**

list[*BeaverAttendance*] – A list of *BeaverAttendance* objects associated with the beaver

**class** beaver\_manager.models.**BeaverAttendance** (attendance\_id, beaver\_id, present)

Bases: flask\_sqlalchemy.Model

Model for a BeaverAttendance. Acts as an intermediary between *Beaver* and *Attendance*

**Parameters**

- **attendance\_id** (*int*) – Foreign key for *Attendance*
- **beaver\_id** (*int*) – Foreign key for *Beaver*
- **present** (*bool*) – Represents whether the beaver was been presesnt or not

**id**

*int* – Unique Primary Key.

**attendance**

*Attendance* – Provides a link to the *Attendance* that BeaverAttendance is associated with

**beaver**

*Beaver* – Provides a link to the *Beaver* that BeaverAttendance is associated with

```
class beaver_manager.models.BeaverBadge (beaver_id, badge_id, completed)
    Bases: flask_sqlalchemy.Model
```

Model for a BeaverBadge. Acts as an intermediary between *Beaver* and *Badge*

**Parameters**

- **beaver\_id** (*int*) – Foreign key for *Beaver*
- **badge\_id** (*int*) – Foreign key for *Badge*
- **completed** (*bool*) – Represents whether the badge has been completed or not

**id**

*int* – Unique Primary Key.

**beaver**

*Beaver* – Provides a link to the *Beaver* that badge is associated with

```
class beaver_manager.models.BeaverTrip (beaver_id, trip_id, permission, paid)
    Bases: flask_sqlalchemy.Model
```

Model for a BeaverTrip. Acts as intermediary between *Beaver* and *Trip*

**Parameters**

- **beaver\_id** (*int*) – Foreign key for *Beaver*
- **trip\_id** (*int*) – Foreign key for *Trip*
- **permission** (*bool*) – Represents whether the criterion has been completed or not
- **paid** (*bool*) – Represents whether the criterion has been completed or not

**id**

*int* – Unique Primary Key.

**beaver**

*Beaver* – Provides a link to the *Beaver* that BeaverTrip is associated with

**trip**

*Trip* – Provides a link to the *Trip* that BeaverTrip is associated with

```
class beaver_manager.models.Criterion (**kwargs)
    Bases: flask_sqlalchemy.Model
```

Model for a Criterion.

**id**

*int* – Unique Primary Key.

**badge\_id**

*int* – Foreign key for *Badge*

**badge**

*Badge* – Provides a link to the *Badge* that criterion is associated with

**description**

*str* – Description of criterion

```
class beaver_manager.models.EmergencyContact (**kwargs)
    Bases: flask_sqlalchemy.Model
```

Model for an Emergency contact

**id**

*int* – Unique Primary Key.

**beaver\_id**  
*int* – Foreign Key for the *Beaver* table

**beaver**  
*Beaver* – Provides direct access to the *Beaver* that is

**linked to the contact**

**first\_name**  
*str* – The contact’s first name

**surname**  
*str* – The contact’s surname

**email**  
*str* – The contact’s email

**phone\_number**  
*str* – The contact’s phone number

**address\_line1**  
*str* – The contact’s first address line

**address\_line2**  
*str* – The contact’s first second address line

**town**  
*str* – The town in which the Contact lives

**county**  
*str* – The county in which the Contact lives

**postcode**  
*str* – The contact’s postcode

**class** beaver\_manager.models.**Lodge** (\*\*kwargs)  
Bases: flask\_sqlalchemy.Model  
Model for a Lodge (group of beavers)

**id**  
*int* – Unique Primary Key.

**name**  
*str* – Name of the lodge

**beavers**  
list[*Beaver*] – A list of beavers associated with the lodge

**class** beaver\_manager.models.**Trip** (\*\*kwargs)  
Bases: flask\_sqlalchemy.Model  
A model for a Trip.

**id**  
*int* – Unique Primary Key.

**location**  
*str* – Place Name

**cost**  
*numeric* – Cost of trip [Max Value of 99999.99]

**date**  
*DateTime* – Date of trip

**overnight**

*bool* – Whether trip is overnight

**number\_of\_nights**

*int* – Number of nights the trip is

## 1.6 beaver\_manager.views module

This module contains the views for the app.

Using flask's `route()` function to route the request it then executes the necessary code return a rendered template at the end.

```
class beaver_manager.views.AttendanceModelView(model, session, name=None, category=None, endpoint=None, url=None, static_folder=None, menu_class_name=None, menu_icon_type=None, menu_icon_value=None)
```

Bases: `flask_admin.contrib.sqla.view.ModelView`

Custom view for Flask-Admin. Adds BeaverAttendance as an inline view

**action\_view()**

Mass-model action view.

**ajax\_update()**

Edits a single column of a record in list view.

**create\_view()**

Create model view

**delete\_view()**

Delete model view. Only POST method is allowed.

**details\_view()**

Details model view

**edit\_view()**

Edit model view

**export\_csv()**

Export a CSV of records.

**index\_view()**

List view

```
class beaver_manager.views.BadgeModelView(model, session, name=None, category=None, endpoint=None, url=None, static_folder=None, menu_class_name=None, menu_icon_type=None, menu_icon_value=None)
```

Bases: `flask_admin.contrib.sqla.view.ModelView`

Custom view for Flask-Admin. Adds criterion as inline model and modifies update behaviour

**action\_view()**

Mass-model action view.

**ajax\_update()**

Edits a single column of a record in list view.

**create\_view()**

Create model view

**delete\_view()**

Delete model view. Only POST method is allowed.

**details\_view()**

Details model view

**edit\_view()**

Edit model view

**export\_csv()**

Export a CSV of records.

**index\_view()**

List view

**on\_model\_change** (*form, model, is\_created*)

When a Badge record is created or modified ensures that it has a BeaverBadge record for all Beavers and that the BeaverBadge has the correct criteria

```
class beaver_manager.views.BeaverBadgeModelView(model, session, name=None, category=None, endpoint=None, url=None, static_folder=None, menu_class_name=None, menu_icon_type=None, menu_icon_value=None)
```

Bases: flask\_admin.contrib.sqla.view.ModelView

Only used for debugging. Custom view for Flask-Admin. Adds BadgeCriterion as an inline view

**action\_view()**

Mass-model action view.

**ajax\_update()**

Edits a single column of a record in list view.

**create\_view()**

Create model view

**delete\_view()**

Delete model view. Only POST method is allowed.

**details\_view()**

Details model view

**edit\_view()**

Edit model view

**export\_csv()**

Export a CSV of records.

**index\_view()**

List view

```
class beaver_manager.views.BeaverModelView(model, session, name=None, category=None, endpoint=None, url=None, static_folder=None, menu_class_name=None, menu_icon_type=None, menu_icon_value=None)
```

Bases: flask\_admin.contrib.sqla.view.ModelView

Custom view for Flask-Admin. Adds emergency contacts as inline model and modifies update behaviour

**action\_view()**

Mass-model action view.

**ajax\_update()**

Edits a single column of a record in list view.

**create\_view()**

Create model view

**delete\_view()**

Delete model view. Only POST method is allowed.

**details\_view()**

Details model view

**edit\_view()**

Edit model view

**export\_csv()**

Export a CSV of records.

**index\_view()**

List view

**on\_model\_change** (*form, model, is\_created*)

When a Beaver record is created or modified ensures that it has a BeaverBadge record for all Badges and that the BeaverBadge has the correct criteria. Also ensures BeaverAttendances, and BeaverTrips have been created.

**on\_model\_delete** (*model*)

When a beaver is deleted delete all :class:'BeaverBadge's, :class:'BeaverTrip's, :class:'BeaverAttendance's and :class:'EmergencyContact's associated with it.

```
class beaver_manager.views.TripModelView(model, session, name=None, category=None,
                                         endpoint=None, url=None, static_folder=None,
                                         menu_class_name=None, menu_icon_type=None,
                                         menu_icon_value=None)
```

Bases: flask\_admin.contrib.sqla.view.ModelView

Custom view for Flask-Admin modifying update behaviour

**action\_view()**

Mass-model action view.

**ajax\_update()**

Edits a single column of a record in list view.

**create\_view()**

Create model view

**delete\_view()**

Delete model view. Only POST method is allowed.

**details\_view()**

Details model view

**edit\_view()**

Edit model view

**export\_csv()**

Export a CSV of records.

**index\_view()**

List view

**on\_model\_change** (*form, model, is\_created*)

When a Trip record is created or modified ensures that it has a BeaverTrip record for all Beavers.

**on\_model\_delete** (*model*)

When a Trip is deleted, delete all :class:‘BeaverTrip’s associated with it.

`beaver_manager.views.beaver` (*beaver\_id*)

Displays Information about the given beaver

**Parameters** `beaver_id` (*int*) – The ID number of the beaver record

`beaver_manager.views.beaver_to_pdf` (*beaver*)

Takes a beaver and returns there information as a pdf

**Parameters** `beaver` (*Beaver*) – The beaver that needs exporting

`beaver_manager.views.beavers` ()

Queries the database for all beavers then displays a list of them

`beaver_manager.views.index` (*success=None, error=None*)

Displays the homepage

**Parameters**

- **success** (*str*) – Message to be displayed in toatr success pop up
- **error** (*str*) – Message to be displayed in toatr error pop up

`beaver_manager.views.register` (*attendance\_id*)

Displays a form to record beavers attendance.

**Parameters** `attendance_id` (*int*) – The ID number of the attendance record

`beaver_manager.views.registers` ()

Displays a list displaying dates for which registers can be taken

`beaver_manager.views.trip` (*trip\_id*)

Displays information about the trip and allows user to select whether or a bevaer has paid for the trip and has permission for the trip

**Parameters** `trip_id` (*int*) – The ID number of the trip record

`beaver_manager.views.trips` ()

Displays a list of all trips linking to an indiviual view showing more data

## 1.7 beaver\_manager.logic module

This module contains any logic that does not belong in any other module

`beaver_manager.logic.email_contacts_trip` (*beaver*)

Checks that a beaver has paid and given permission to go on a trip

`beaver_manager.logic.to_percent` (*value, total*)

Works out the percentage of a value from a total.

**Parameters**

- **value** (*int*) – The value to be converted
- **total** (*int*) – The total to be used in conversion

**Returns** Number between 0.0 and 100.0

**Return type** (int)

`beaver_manager.logic.update_beaver_badge(beaver_badge)`

Checks to see if all the criteria are completed for `beaver_badge` and if so sets `beaver_badge.completed` to True

**Parameters**

- **beaver\_badge** (`BeaverBadge`) – The beaver badge which need
- **checking** –

`beaver_manager.logic.update_criterion(attendance)`

Check whether a beaver was present for a given attendance and if so sets the corresponding `criterion.completed` to True

**Parameters** **attendance** (`Attendance`) – The attendance for which criterion need updating



---

## config module

---

Contains the configuration options for the flask app.

`config.basedir`

*str* – Base directory of the flask app

`config.SQLALCHEMY_DATABASE_URI`

*str* – Location of SQLAlchemy database

`config.SQLALCHEMY_MIGRATE_REPO`

*str* – Directory containing database migrations and history

`config.WTF_CSRF_ENABLED`

*bool* – Whether CSRF is enabled for forms, protecting data

`config.SECRET_KEY`

*str* – A secret key



---

## API Requests

---

**GET** /admin/beaverattendance/export/csv/  
Export a CSV of records.

**POST** /admin/beaverattendance/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/badgecriterion/export/csv/  
Export a CSV of records.

**POST** /admin/badgecriterion/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/beaverbadge/export/csv/  
Export a CSV of records.

**POST** /admin/beaverbadge/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/attendance/export/csv/  
Export a CSV of records.

**GET** /admin/beavertrip/export/csv/  
Export a CSV of records.

**POST** /admin/attendance/ajax/update/  
Edits a single column of a record in list view.

**POST** /admin/beavertrip/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/criterion/export/csv/  
Export a CSV of records.

**POST** /admin/criterion/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/beaver/export/csv/  
Export a CSV of records.

**POST** /admin/beaver/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/badge/export/csv/  
Export a CSV of records.

**GET** /admin/lodge/export/csv/  
Export a CSV of records.

**POST** /admin/badge/ajax/update/  
Edits a single column of a record in list view.

**POST** /admin/lodge/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/trip/export/csv/  
Export a CSV of records.

**POST** /admin/trip/ajax/update/  
Edits a single column of a record in list view.

**GET** /admin/beaverattendance/details/  
Details model view

**POST** /admin/beaverattendance/action/  
Mass-model action view.

**POST** /admin/beaverattendance/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/beaverattendance/edit/  
Edit model view

**GET** /admin/beaverattendance/edit/  
Edit model view

**POST** /admin/beaverattendance/new/  
Create model view

**GET** /admin/beaverattendance/new/  
Create model view

**GET** /admin/badgecriterion/details/  
Details model view

**POST** /admin/badgecriterion/action/  
Mass-model action view.

**POST** /admin/badgecriterion/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/badgecriterion/edit/  
Edit model view

**GET** /admin/badgecriterion/edit/  
Edit model view

**POST** /admin/badgecriterion/new/  
Create model view

**GET** /admin/badgecriterion/new/  
Create model view

**GET** /admin/beaverbadge/details/  
Details model view

**POST** /admin/beaverbadge/action/  
Mass-model action view.

**POST** /admin/beaverbadge/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/beaverbadge/edit/  
Edit model view

**GET** /admin/beaverbadge/edit/  
Edit model view

**POST** /admin/beaverbadge/new/  
Create model view

**GET** /admin/beaverbadge/new/  
Create model view

**GET** /admin/attendance/details/  
Details model view

**GET** /admin/beavertrip/details/  
Details model view

**POST** /admin/attendance/action/  
Mass-model action view.

**POST** /admin/attendance/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/beavertrip/action/  
Mass-model action view.

**POST** /admin/beavertrip/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/attendance/edit/  
Edit model view

**GET** /admin/attendance/edit/  
Edit model view

**POST** /admin/beavertrip/edit/  
Edit model view

**GET** /admin/beavertrip/edit/  
Edit model view

**POST** /admin/attendance/new/  
Create model view

**GET** /admin/attendance/new/  
Create model view

**POST** /admin/beavertrip/new/  
Create model view

**GET** /admin/beavertrip/new/  
Create model view

**GET** /admin/criterion/details/  
Details model view

**POST** /admin/criterion/action/  
Mass-model action view.

**POST** /admin/criterion/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/criterion/edit/  
Edit model view

**GET** /admin/criterion/edit/  
Edit model view

**POST** /admin/criterion/new/  
Create model view

**GET** /admin/criterion/new/  
Create model view

**GET** /admin/beaver/details/  
Details model view

**POST** /admin/beaver/action/  
Mass-model action view.

**POST** /admin/beaver/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/beaver/edit/  
Edit model view

**GET** /admin/beaver/edit/  
Edit model view

**POST** /admin/beaver/new/  
Create model view

**GET** /admin/beaver/new/  
Create model view

**GET** /admin/badge/details/  
Details model view

**GET** /admin/lodge/details/  
Details model view

**POST** /admin/badge/action/  
Mass-model action view.

**POST** /admin/badge/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/lodge/action/  
Mass-model action view.

**POST** /admin/lodge/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/badge/edit/  
Edit model view

**GET** /admin/badge/edit/  
Edit model view

**POST** /admin/lodge/edit/  
Edit model view

**GET** /admin/lodge/edit/  
Edit model view

**POST** /admin/badge/new/  
Create model view

**GET** /admin/badge/new/  
Create model view

**POST** /admin/lodge/new/  
Create model view

**GET** /admin/lodge/new/  
Create model view

**GET** /admin/trip/details/  
Details model view

**POST** /admin/trip/action/  
Mass-model action view.

**POST** /admin/trip/delete/  
Delete model view. Only POST method is allowed.

**POST** /admin/trip/edit/  
Edit model view

**GET** /admin/trip/edit/  
Edit model view

**POST** /admin/trip/new/  
Create model view

**GET** /admin/trip/new/  
Create model view

**GET** /admin/beaverattendance/  
List view

**GET** /admin/badgecriterion/  
List view

**GET** /admin/beaverbadge/  
List view

**GET** /admin/attendance/  
List view

**GET** /admin/beavertrip/  
List view

**GET** /admin/criterion/  
List view

**GET** /admin/beaver/  
List view

**GET** /admin/badge/  
List view

**GET** /admin/lodge/  
List view

**GET** /admin/trip/  
List view

**GET /registers**

Displays a list displaying dates for which registers can be taken

**GET /beavers**

Queries the database for all beavers then displays a list of them

**POST /index**

**POST /**

Displays the homepage

**Args:** success (str): Message to be displayed in toastr success pop up error (str): Message to be displayed in toastr error pop up

**GET /index**

**GET /**

Displays the homepage

**Args:** success (str): Message to be displayed in toastr success pop up error (str): Message to be displayed in toastr error pop up

**GET /trips/**

Displays a list of all trips linking to an individual view showing more data

**GET /admin/static/ (path: filename)**

Function used internally to send static files from the static folder to the browser.

New in version 0.5.

**POST /registers/ (attendance\_id)**

Displays a form to record beavers attendance.

**Args:** attendance\_id (int): The ID number of the attendance record

**GET /registers/ (attendance\_id)**

Displays a form to record beavers attendance.

**Args:** attendance\_id (int): The ID number of the attendance record

**GET /beavers/ (beaver\_id)**

Displays Information about the given beaver

**Args:** beaver\_id (int): The ID number of the beaver record

**POST /trips/ (trip\_id)**

Displays information about the trip and allows user to select whether or a beaver has paid for the trip and has permission for the trip

**Args:** trip\_id (int): The ID number of the trip record

**GET /trips/ (trip\_id)**

Displays information about the trip and allows user to select whether or a beaver has paid for the trip and has permission for the trip

**Args:** trip\_id (int): The ID number of the trip record



---

## Indices and tables

---

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



/	GET /admin/beavertrip/edit/, 17
GET /, 20	GET /admin/beavertrip/export/csv/, 15
POST /, 20	GET /admin/beavertrip/new/, 17
/admin	GET /admin/criterion/, 19
GET /admin/attendance/, 19	GET /admin/criterion/details/, 17
GET /admin/attendance/details/, 17	GET /admin/criterion/edit/, 18
GET /admin/attendance/edit/, 17	GET /admin/criterion/export/csv/, 15
GET /admin/attendance/export/csv/, 15	GET /admin/criterion/new/, 18
GET /admin/attendance/new/, 17	GET /admin/lodge/, 19
GET /admin/badge/, 19	GET /admin/lodge/details/, 18
GET /admin/badge/details/, 18	GET /admin/lodge/edit/, 18
GET /admin/badge/edit/, 18	GET /admin/lodge/export/csv/, 15
GET /admin/badge/export/csv/, 15	GET /admin/lodge/new/, 19
GET /admin/badge/new/, 19	GET /admin/static/ (path:filename), 20
GET /admin/badgecriterion/, 19	GET /admin/trip/, 19
GET /admin/badgecriterion/details/, 16	GET /admin/trip/details/, 19
GET /admin/badgecriterion/edit/, 16	GET /admin/trip/edit/, 19
GET /admin/badgecriterion/export/csv/, 15	GET /admin/trip/export/csv/, 16
GET /admin/badgecriterion/new/, 16	GET /admin/trip/new/, 19
GET /admin/beaver/, 19	POST /admin/attendance/action/, 17
GET /admin/beaver/details/, 18	POST /admin/attendance/ajax/update/, 15
GET /admin/beaver/edit/, 18	POST /admin/attendance/delete/, 17
GET /admin/beaver/export/csv/, 15	POST /admin/attendance/edit/, 17
GET /admin/beaver/new/, 18	POST /admin/attendance/new/, 17
GET /admin/beaverattendance/, 19	POST /admin/badge/action/, 18
GET /admin/beaverattendance/details/, 16	POST /admin/badge/ajax/update/, 15
GET /admin/beaverattendance/edit/, 16	POST /admin/badge/delete/, 18
GET /admin/beaverattendance/export/csv/, 15	POST /admin/badge/edit/, 18
GET /admin/beaverattendance/new/, 16	POST /admin/badge/new/, 18
GET /admin/beaverbadge/, 19	POST /admin/badgecriterion/action/, 16
GET /admin/beaverbadge/details/, 16	POST /admin/badgecriterion/ajax/update/, 15
GET /admin/beaverbadge/edit/, 17	POST /admin/badgecriterion/delete/, 16
GET /admin/beaverbadge/export/csv/, 15	POST /admin/badgecriterion/edit/, 16
GET /admin/beaverbadge/new/, 17	POST /admin/badgecriterion/new/, 16
GET /admin/beavertrip/, 19	POST /admin/beaver/action/, 18
GET /admin/beavertrip/details/, 17	POST /admin/beaver/ajax/update/, 15
	POST /admin/beaver/delete/, 18
	POST /admin/beaver/edit/, 18
	POST /admin/beaver/new/, 18

POST /admin/beaverattendance/action/,  
16  
POST /admin/beaverattendance/ajax/update/,  
15  
POST /admin/beaverattendance/delete/,  
16  
POST /admin/beaverattendance/edit/, 16  
POST /admin/beaverattendance/new/, 16  
POST /admin/beaverbadge/action/, 16  
POST /admin/beaverbadge/ajax/update/,  
15  
POST /admin/beaverbadge/delete/, 16  
POST /admin/beaverbadge/edit/, 16  
POST /admin/beaverbadge/new/, 17  
POST /admin/beavertrip/action/, 17  
POST /admin/beavertrip/ajax/update/, 15  
POST /admin/beavertrip/delete/, 17  
POST /admin/beavertrip/edit/, 17  
POST /admin/beavertrip/new/, 17  
POST /admin/criterion/action/, 17  
POST /admin/criterion/ajax/update/, 15  
POST /admin/criterion/delete/, 17  
POST /admin/criterion/edit/, 17  
POST /admin/criterion/new/, 18  
POST /admin/lodge/action/, 18  
POST /admin/lodge/ajax/update/, 16  
POST /admin/lodge/delete/, 18  
POST /admin/lodge/edit/, 18  
POST /admin/lodge/new/, 19  
POST /admin/trip/action/, 19  
POST /admin/trip/ajax/update/, 16  
POST /admin/trip/delete/, 19  
POST /admin/trip/edit/, 19  
POST /admin/trip/new/, 19

## **/beavers**

GET /beavers, 20  
GET /beavers/(beaver\_id), 20

## **/index**

GET /index, 20  
POST /index, 20

## **/registers**

GET /registers, 19  
GET /registers/(attendance\_id), 20  
POST /registers/(attendance\_id), 20

## **/trips**

GET /trips/, 20  
GET /trips/(trip\_id), 20  
POST /trips/(trip\_id), 20

## **b**

`beaver_manager`, [3](#)  
`beaver_manager.decorators`, [3](#)  
`beaver_manager.email`, [3](#)  
`beaver_manager.forms`, [3](#)  
`beaver_manager.logic`, [11](#)  
`beaver_manager.models`, [4](#)  
`beaver_manager.views`, [8](#)

## **c**

`config`, [13](#)



## A

action\_view() (beaver\_manager.views.AttendanceModelView method), 8  
 action\_view() (beaver\_manager.views.BadgeModelView method), 8  
 action\_view() (beaver\_manager.views.BeaverBadgeModelView method), 9  
 action\_view() (beaver\_manager.views.BeaverModelView method), 10  
 action\_view() (beaver\_manager.views.TripModelView method), 10  
 address\_line1 (beaver\_manager.models.EmergencyContact attribute), 7  
 address\_line2 (beaver\_manager.models.EmergencyContact attribute), 7  
 ajax\_update() (beaver\_manager.views.AttendanceModelView method), 8  
 ajax\_update() (beaver\_manager.views.BadgeModelView method), 8  
 ajax\_update() (beaver\_manager.views.BeaverBadgeModelView method), 9  
 ajax\_update() (beaver\_manager.views.BeaverModelView method), 10  
 ajax\_update() (beaver\_manager.views.TripModelView method), 10  
 app (in module beaver\_manager), 3  
 async() (in module beaver\_manager.decorators), 3  
 attendance (beaver\_manager.models.BeaverAttendance attribute), 5  
 Attendance (class in beaver\_manager.models), 4  
 AttendanceModelView (class in beaver\_manager.views), 8

## B

badge (beaver\_manager.models.BadgeCriterion attribute), 5  
 badge (beaver\_manager.models.Criterion attribute), 6  
 Badge (class in beaver\_manager.models), 4  
 badge\_id (beaver\_manager.models.Criterion attribute), 6  
 BadgeCriterion (class in beaver\_manager.models), 4

BadgeModelView (class in beaver\_manager.views), 8  
 badges (beaver\_manager.models.Beaver attribute), 5  
 basedir (in module config), 13  
 beaver (beaver\_manager.models.BeaverAttendance attribute), 5  
 beaver (beaver\_manager.models.BeaverBadge attribute), 6  
 beaver (beaver\_manager.models.BeaverTrip attribute), 6  
 beaver (beaver\_manager.models.EmergencyContact attribute), 7  
 Beaver (class in beaver\_manager.models), 5  
 beaver() (in module beaver\_manager.views), 11  
 beaver\_attendances (beaver\_manager.models.Beaver attribute), 5  
 beaver\_id (beaver\_manager.models.EmergencyContact attribute), 6  
 beaver\_manager (module), 3  
 beaver\_manager.decorators (module), 3  
 beaver\_manager.email (module), 3  
 beaver\_manager.forms (module), 3  
 beaver\_manager.logic (module), 11  
 beaver\_manager.models (module), 4  
 beaver\_manager.views (module), 8  
 beaver\_to\_pdf() (in module beaver\_manager.views), 11  
 BeaverAttendance (class in beaver\_manager.models), 5  
 BeaverBadge (class in beaver\_manager.models), 5  
 BeaverBadgeModelView (class in beaver\_manager.views), 9  
 BeaverModelView (class in beaver\_manager.views), 9  
 beavers (beaver\_manager.models.Lodge attribute), 7  
 beavers() (in module beaver\_manager.views), 11  
 beavers\_badges (beaver\_manager.models.Badge attribute), 4  
 BeaverTrip (class in beaver\_manager.models), 6

## C

config (module), 13  
 contacts (beaver\_manager.models.Beaver attribute), 5  
 cost (beaver\_manager.models.Trip attribute), 7  
 county (beaver\_manager.models.EmergencyContact attribute), 7

[create\\_view\(\)](#) (beaver\_manager.views.AttendanceModelView method), 8  
[create\\_view\(\)](#) (beaver\_manager.views.BadgeModelView method), 8  
[create\\_view\(\)](#) (beaver\_manager.views.BeaverBadgeModelView method), 9  
[create\\_view\(\)](#) (beaver\_manager.views.BeaverModelView method), 10  
[create\\_view\(\)](#) (beaver\_manager.views.TripModelView method), 10  
[criterion](#) (beaver\_manager.models.Attendance attribute), 4  
[criterion](#) (beaver\_manager.models.BadgeCriterion attribute), 5  
[Criterion](#) (class in beaver\_manager.models), 6  
[criterion\\_id](#) (beaver\_manager.models.Attendance attribute), 4

## D

[date](#) (beaver\_manager.models.Attendance attribute), 4  
[date](#) (beaver\_manager.models.Trip attribute), 7  
[db](#) (in module beaver\_manager), 3  
[delete\\_view\(\)](#) (beaver\_manager.views.AttendanceModelView method), 8  
[delete\\_view\(\)](#) (beaver\_manager.views.BadgeModelView method), 9  
[delete\\_view\(\)](#) (beaver\_manager.views.BeaverBadgeModelView method), 9  
[delete\\_view\(\)](#) (beaver\_manager.views.BeaverModelView method), 10  
[delete\\_view\(\)](#) (beaver\_manager.views.TripModelView method), 10  
[description](#) (beaver\_manager.models.Attendance attribute), 4  
[description](#) (beaver\_manager.models.BadgeCriterion attribute), 5  
[description](#) (beaver\_manager.models.Criterion attribute), 6  
[details\\_view\(\)](#) (beaver\_manager.views.AttendanceModelView method), 8  
[details\\_view\(\)](#) (beaver\_manager.views.BadgeModelView method), 9  
[details\\_view\(\)](#) (beaver\_manager.views.BeaverBadgeModelView method), 9  
[details\\_view\(\)](#) (beaver\_manager.views.BeaverModelView method), 10  
[details\\_view\(\)](#) (beaver\_manager.views.TripModelView method), 10  
[dob](#) (beaver\_manager.models.Beaver attribute), 5

## E

[edit\\_view\(\)](#) (beaver\_manager.views.AttendanceModelView method), 8

[edit\\_view\(\)](#) (beaver\_manager.views.BadgeModelView method), 9  
[edit\\_view\(\)](#) (beaver\_manager.views.BeaverBadgeModelView method), 9  
[edit\\_view\(\)](#) (beaver\_manager.views.BeaverModelView method), 10  
[edit\\_view\(\)](#) (beaver\_manager.views.TripModelView method), 10  
[email](#) (beaver\_manager.models.EmergencyContact attribute), 7  
[email\\_contacts\\_trip\(\)](#) (in module beaver\_manager.logic), 11  
[EmergencyContact](#) (class in beaver\_manager.models), 6  
[export\\_csv\(\)](#) (beaver\_manager.views.AttendanceModelView method), 8  
[export\\_csv\(\)](#) (beaver\_manager.views.BadgeModelView method), 9  
[export\\_csv\(\)](#) (beaver\_manager.views.BeaverBadgeModelView method), 9  
[export\\_csv\(\)](#) (beaver\_manager.views.BeaverModelView method), 10  
[export\\_csv\(\)](#) (beaver\_manager.views.TripModelView method), 10

## F

[first\\_name](#) (beaver\_manager.models.Beaver attribute), 5  
[first\\_name](#) (beaver\_manager.models.EmergencyContact attribute), 7

## I

[id](#) (beaver\_manager.models.Attendance attribute), 4  
[id](#) (beaver\_manager.models.Badge attribute), 4  
[id](#) (beaver\_manager.models.BadgeCriterion attribute), 4  
[id](#) (beaver\_manager.models.Beaver attribute), 5  
[id](#) (beaver\_manager.models.BeaverAttendance attribute), 5  
[id](#) (beaver\_manager.models.BeaverBadge attribute), 6  
[id](#) (beaver\_manager.models.BeaverTrip attribute), 6  
[id](#) (beaver\_manager.models.Criterion attribute), 6  
[id](#) (beaver\_manager.models.EmergencyContact attribute), 6  
[id](#) (beaver\_manager.models.Lodge attribute), 7  
[id](#) (beaver\_manager.models.Trip attribute), 7  
[img\\_url](#) (beaver\_manager.models.Badge attribute), 4  
[index\(\)](#) (in module beaver\_manager.views), 11  
[index\\_view\(\)](#) (beaver\_manager.views.AttendanceModelView method), 8  
[index\\_view\(\)](#) (beaver\_manager.views.BadgeModelView method), 9  
[index\\_view\(\)](#) (beaver\_manager.views.BeaverBadgeModelView method), 9  
[index\\_view\(\)](#) (beaver\_manager.views.BeaverModelView method), 10



`index_view()` (beaver\_manager.views.TripModelView method), 10

## L

`location` (beaver\_manager.models.Trip attribute), 7  
`lodge` (beaver\_manager.models.Beaver attribute), 5  
`Lodge` (class in beaver\_manager.models), 7  
`lodge_id` (beaver\_manager.models.Beaver attribute), 5

## M

`MultiCheckboxField` (class in beaver\_manager.forms), 3

## N

`name` (beaver\_manager.models.Badge attribute), 4  
`name` (beaver\_manager.models.Lodge attribute), 7  
`number_of_nights` (beaver\_manager.models.Trip attribute), 8

## O

`on_model_change()` (beaver\_manager.views.BadgeModelView method), 9  
`on_model_change()` (beaver\_manager.views.BeaverModelView method), 10  
`on_model_change()` (beaver\_manager.views.TripModelView method), 11  
`on_model_delete()` (beaver\_manager.views.BeaverModelView method), 10  
`on_model_delete()` (beaver\_manager.views.TripModelView method), 11  
`organiser` (beaver\_manager.models.Attendance attribute), 4  
`overnight` (beaver\_manager.models.Trip attribute), 7

## P

`phone_number` (beaver\_manager.models.EmergencyContact attribute), 7  
`postcode` (beaver\_manager.models.EmergencyContact attribute), 7

## R

`register()` (in module beaver\_manager.views), 11  
`registers()` (in module beaver\_manager.views), 11

## S

`SECRET_KEY` (in module config), 13  
`send_email()` (in module beaver\_manager.email), 3  
`SQLALCHEMY_DATABASE_URI` (in module config), 13  
`SQLALCHEMY_MIGRATE_REPO` (in module config), 13  
`surname` (beaver\_manager.models.Beaver attribute), 5  
`surname` (beaver\_manager.models.EmergencyContact attribute), 7

## T

`to_percent()` (in module beaver\_manager.logic), 11  
`town` (beaver\_manager.models.EmergencyContact attribute), 7  
`trip` (beaver\_manager.models.BeaverTrip attribute), 6  
`Trip` (class in beaver\_manager.models), 7  
`trip()` (in module beaver\_manager.views), 11  
`TripModelView` (class in beaver\_manager.views), 10  
`trips()` (in module beaver\_manager.views), 11

## U

`update_beaver_badge()` (in module beaver\_manager.logic), 12  
`update_criterion()` (in module beaver\_manager.logic), 12

## W

`WTF_CSRF_ENABLED` (in module config), 13