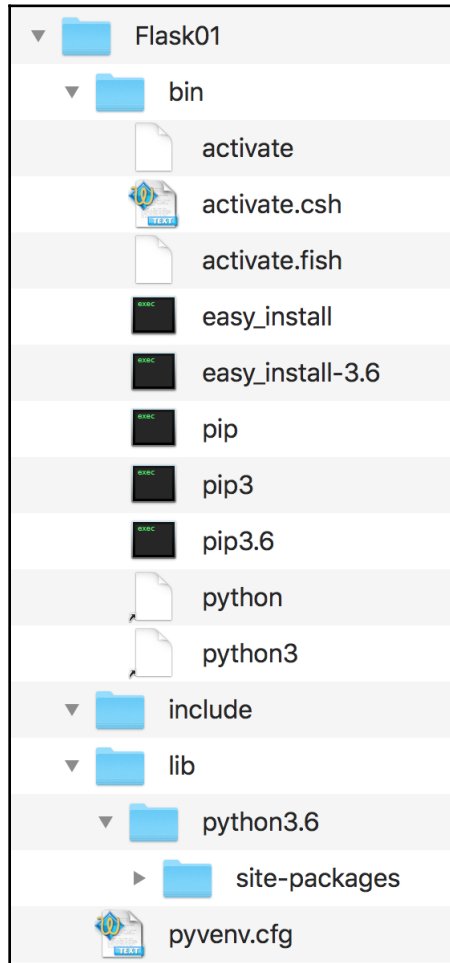
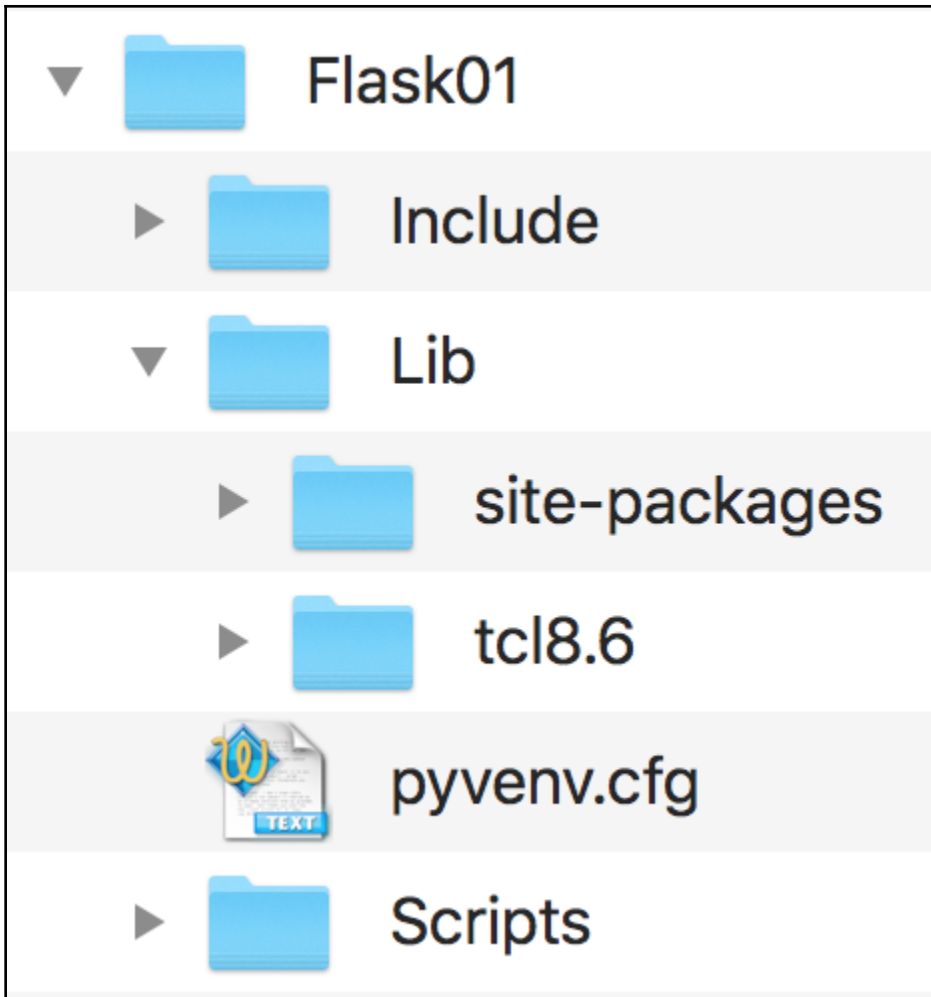


Chapter 1: Developing RESTful APIs and Microservices with Flask 1.0.2





```
gaston — -bash — 114x35
Gastons-MacBook-Pro:~ gaston$ python3 -m venv ~/HillarPythonREST2/Flask01
Gastons-MacBook-Pro:~ gaston$ echo $SHELL
/bin/bash
Gastons-MacBook-Pro:~ gaston$ source ~/HillarPythonREST2/Flask01/bin/activate
(Flask01) Gastons-MacBook-Pro:~ gaston$ █
```

```
Command Prompt
Microsoft Windows [Version 10.0.17134.345]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\gaston>python -m venv %USERPROFILE%\HillarPythonREST2\Flask01

C:\Users\gaston>%USERPROFILE%\HillarPythonREST2\Flask01\Scripts\activate.bat
(Flask01) C:\Users\gaston>
```

```
Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

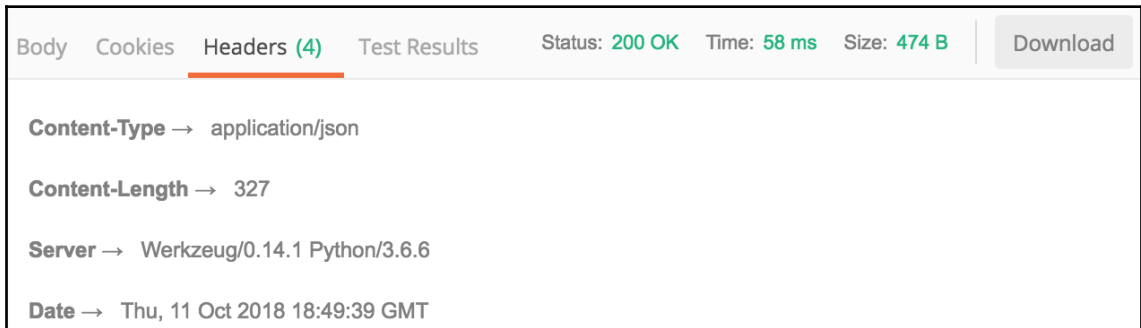
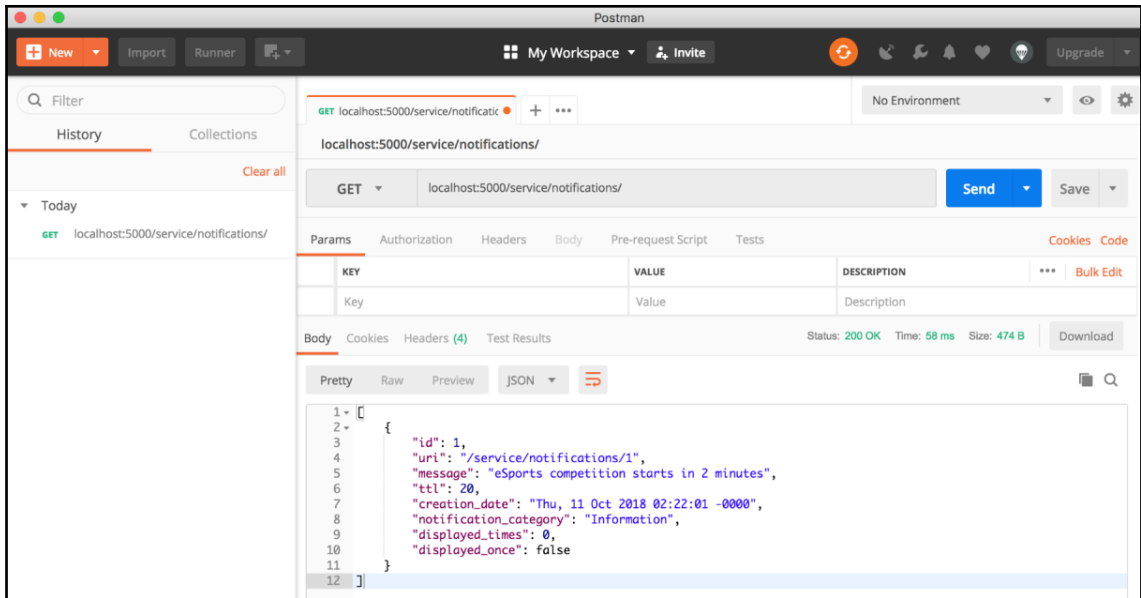
PS C:\Users\gaston> cd $env:USERPROFILE
PS C:\Users\gaston> HillarPythonREST2\Flask01\Scripts\Activate.ps1
(Flask01) PS C:\Users\gaston>
```

```
Flask01 -- Python - Python service/service.py -- 51x42
(Flask01) Gastons-MacBook-Pro:Flask01 gaston$ python service/service.py
* Serving Flask app "service" (lazy loading)
* Environment: production
  WARNING: Do not use the development server in a production environment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 122-606-712
127.0.0.1 - - [11/Oct/2018 19:08:12] "POST /service/notifications/ HTTP/1.1" 201 -
127.0.0.1 - - [11/Oct/2018 19:08:23] "POST /service/notifications/ HTTP/1.1" 201 -
127.0.0.1 - - [11/Oct/2018 19:08:39] "GET /service/notifications/ HTTP/1.1" 200 -

Date: Thu, 11 Oct 2018 22:08:23 GMT
Server: Werkzeug/0.14.1 Python/3.6.2
{
  "creation_date": "Thu, 11 Oct 2018 22:08:23 -0000",
  "displayed_once": false,
  "displayed_times": 0,
  "id": 2,
  "message": "Ambient temperature is above the valid range",
  "notification_category": "Warning",
  "ttl": 15,
  "uri": "/service/notifications/2"
}

(Flask01) Gastons-MacBook-Pro:Flask01 gaston$ http ":5000/service/notifications/"
HTTP/1.0 200 OK
Content-Length: 652
Content-Type: application/json
Date: Thu, 11 Oct 2018 22:08:39 GMT
Server: Werkzeug/0.14.1 Python/3.6.2
[
  {
    "creation_date": "Thu, 11 Oct 2018 22:08:12 -0000",
    "displayed_once": false,
    "displayed_times": 0,
    "id": 1,
    "message": "eSports competition starts in 2 minutes",
    "notification_category": "Information",
    "ttl": 20,
    "uri": "/service/notifications/1"
  },
  {
    "creation_date": "Thu, 11 Oct 2018 22:08:23 -0000",
    "displayed_once": false,
    "displayed_times": 0,
    "id": 2,
    "message": "Ambient temperature is above the valid range",
    "notification_category": "Warning",
    "ttl": 15,
    "uri": "/service/notifications/2"
  }
]

(Flask01) Gastons-MacBook-Pro:Flask01 gaston$
```



GET localhost:5000/service/notificatic • POST localhost:5000/service/notificat • + ... No Environment

localhost:5000/service/notifications/

POST localhost:5000/service/notifications/ Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code

form-data x-www-form-urlencoded raw binary JSON (application/json)

```

1 {
2   "message": "Calculating the most appropriate ambient temperature",
3   "ttl": 20,
4   "notification_category": "Warning"
5 }
6

```

Body Cookies Headers (4) Test Results Status: 201 CREATED Time: 36 ms Size: 444 B Download

Pretty Raw Preview JSON

```

1 {
2   "id": 3,
3   "uri": "/service/notifications/3",
4   "message": "Calculating the most appropriate ambient temperature",
5   "ttl": 20,
6   "creation_date": "Thu, 11 Oct 2018 19:18:35 -0000",
7   "notification_category": "Warning",
8   "displayed_times": 0,
9   "displayed_once": false
10 }

```

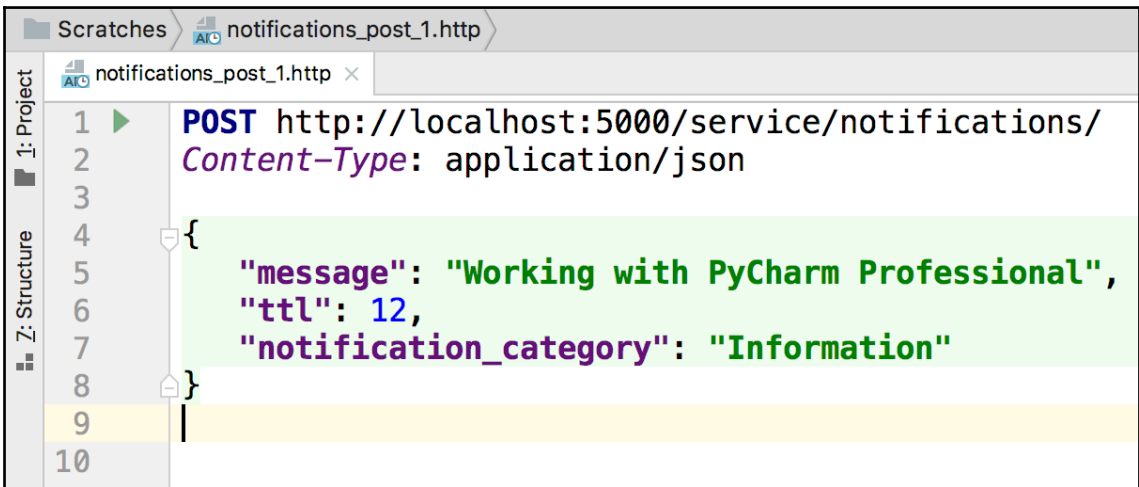
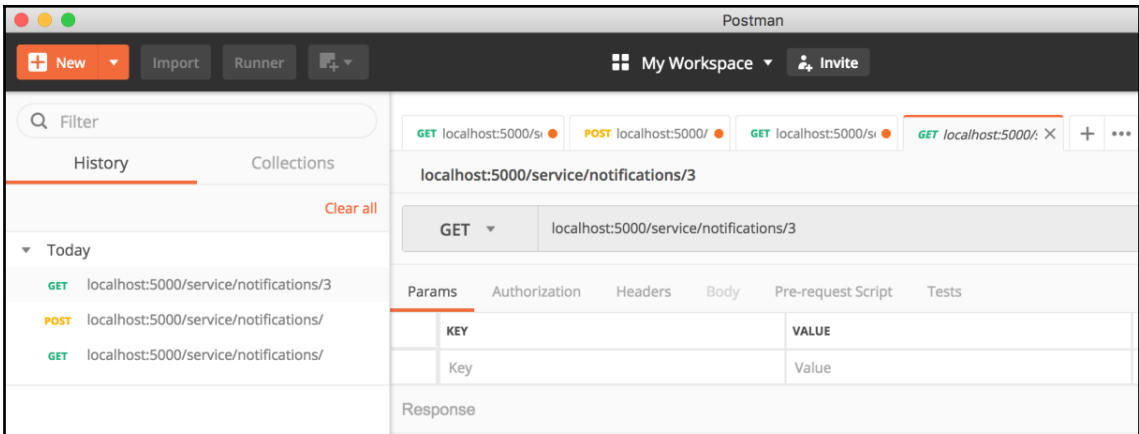
GET localhost:5000/service/notif • POST localhost:5000/service/not • GET localhost:5000/service/notif • + ... No Environment

localhost:5000/service/notifications/3

GET localhost:5000/service/notifications/3 Send Save

Params Authorization Headers Body Pre-request Script Tests Cookies Code

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		



The screenshot displays the PyCharm IDE interface. The top editor pane shows an HTTP POST request to `http://localhost:5000/service/notifications/` with a `Content-Type: application/json`. The request body is a JSON object: `{ "message": "Working with PyCharm Professional", "ttl": 12, "notification_category": "Information" }`. Below the request, the filename `<> 2018-10-11T054308.201.json` and the separator `###` are visible.

The bottom editor pane shows the response from the server. The status is `HTTP/1.0 201 CREATED`. The response headers are `Content-Type: application/json`, `Content-Length: 277`, `Server: Werkzeug/0.14.1 Python/3.6.2`, and `Date: Thu, 11 Oct 2018 20:43:08 GMT`. The response body is a JSON object: `{ "id": 4, "uri": "/service/notifications/4", "message": "Working with PyCharm Professional", "ttl": 12, "creation_date": "Thu, 11 Oct 2018 20:43:08 -0000", "notification_category": "Information", "displayed_times": 0, "displayed_once": false }`.

At the bottom of the response pane, it states: `Response code: 201 (CREATED); Time: 37ms; Content length: 277 bytes`.

The IDE interface includes a sidebar on the left with 'Project', 'Structure', and 'Favorites' views. The bottom status bar shows 'Database Changes', 'Run', 'TODO', 'REST Client', 'Python Console', and 'Terminal' tabs.

18:06 Thu 11 Oct 192.168.1.106:5000/service/notifications/ Go

GET HEAD POST PUT DELETE TRACE OPTIONS Basic Verbose

Content-Length: 978
Server: Werkzeug/0.14.1 Python/3.6.2
Date: Thu, 11 Oct 2018 21:04:52 GMT

```
[
  {
    "id": 1,
    "uri": "/service/notifications/1",
    "message": "eSports competition starts in 2 minutes",
    "ttl": 20,
    "creation_date": "Thu, 11 Oct 2018 02:22:01 -0000",
    "notification_category": "Information",
    "displayed_times": 0,
    "displayed_once": false
  },
  {
    "id": 3,
    "uri": "/service/notifications/3",
    "message": "Calculating the most appropriate ambient temperature",
    "ttl": 20,
    "creation_date": "Thu, 11 Oct 2018 19:18:35 -0000",
    "notification_category": "Warning",
    "displayed_times": 0,
    "displayed_once": false
  },
  {
    "id": 4,
    "uri": "/service/notifications/4",
    "message": "Working with PyCharm Professional",
    "ttl": 12,
    "creation_date": "Thu, 11 Oct 2018 20:43:08 -0000",
    "notification_category": "Information",
    "displayed_times": 0,
    "displayed_once": false
  }
]
```

NS: 0.0s TCP: 0.1s SSL: ---- FB: 0.1s Total: 0.8s H/1.0

User iPhone iPad Safari (Mac) IE (PC) Chrome (Mac) Firefox (PC)

Chapter 2: Working with Models, SQLAlchemy, and Hyperlinked APIs in Flask

```
(Flask01) Gastons-MacBook-Pro:bin gaston$ psql --username=your_user_name --dbname=flask_notifications --command="SELECT * FROM notification_category;"
id | name
---+---
 1 | Information
 2 | Warning
(2 rows)

(Flask01) Gastons-MacBook-Pro:bin gaston$ psql --username=your_user_name --dbname=flask_notifications --command="SELECT * FROM notification;"
id | message | ttl | creation_date | notification_category_id | displayed_times | displayed_once
---+---+---+---+---+---+---
 1 | eSports competition finishes in 10 minutes | 30 | 2018-10-16 21:57:15.867853 | 1 | 0 | f
 2 | No winners yet | 15 | 2018-10-16 21:58:43.737812 | 2 | 0 | f
(2 rows)

(Flask01) Gastons-MacBook-Pro:bin gaston$
```

```
(Flask01) Gastons-MacBook-Pro:bin gaston$ psql --username=your_user_name --dbname=flask_notifications --command="SELECT * FROM notification_category;"
id | name
---+---
 1 | Information
 2 | Warning
 3 | Error
(3 rows)

(Flask01) Gastons-MacBook-Pro:bin gaston$ psql --username=your_user_name --dbname=flask_notifications --command="SELECT * FROM notification;"
id | message | ttl | creation_date | notification_category_id | displayed_times | displayed_once
---+---+---+---+---+---+---
 1 | eSports competition finishes in 10 minutes | 30 | 2018-10-16 21:57:15.867853 | 1 | 0 | f
 2 | No winners yet | 15 | 2018-10-16 21:58:43.737812 | 2 | 0 | f
 3 | Score calculation error | 30 | 2018-10-16 23:07:27.372778 | 3 | 0 | f
(3 rows)

(Flask01) Gastons-MacBook-Pro:bin gaston$
```

Chapter 3: Improving Our API and Adding Authentication to it with Flask

```
(Flask01) Gastons-MacBook-Pro:service gaston$ psql --username=your_user_name --dbname=flask_notifications --command="SELECT * FROM notification WHERE id = 2;"
```

id	message	ttl	creation_date	notification_category_id	displayed_times	displayed_once
2	No winners yet	15	2018-10-16 21:58:43.737812	2	1	t

```
(1 row)

(Flask01) Gastons-MacBook-Pro:service gaston$
```

```
(Flask01) Gastons-MacBook-Pro:service gaston$ psql --username=your_user_name --dbname=flask_notifications --command="SELECT * FROM public.user;"
```

id	name	creation_date	password_hash
1	gaston-hillar	2018-10-20 12:58:14.118624	\$6\$rounds=656000\$6aZxI3Zy/M9RvJxe\$Js3HpvGY.hrzPHX5wk92Dglhq/z/unpvHXBfiiMI5ZDfavP2m19KTHi2NKkrJ4LIHoiRRh0AsSsaCDET03Bn.

```
(1 row)

[END]
```

Chapter 4: Testing and Deploying an API in a Microservice with Flask

Coverage report: 51%

Module ↓	statements	missing	excluded	branches	partial	coverage
models.py	101	27	0	24	7	66%
views.py	208	112	0	46	10	43%
Total	309	139	0	70	17	51%

coverage.py v4.5.1, created at 2018-10-22 19:13

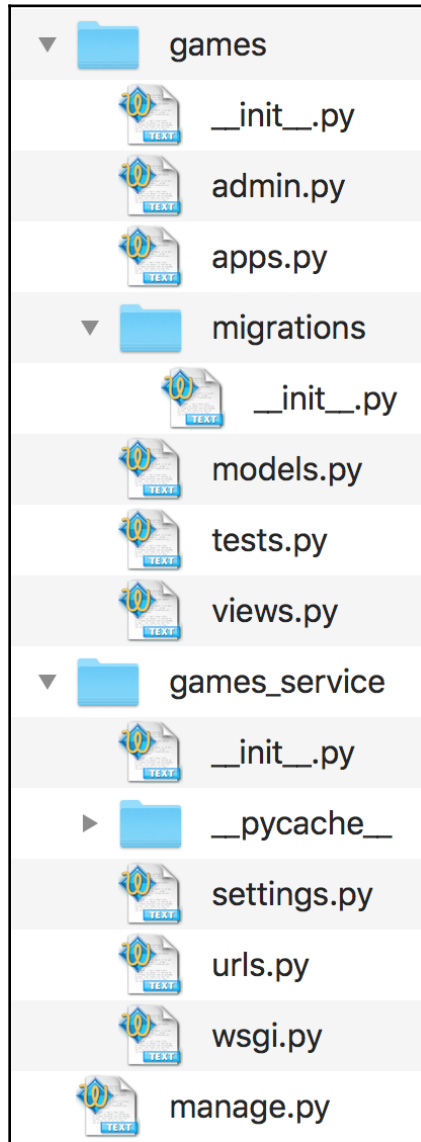
Coverage for **views.py** : 43%

208 statements 96 run 112 missing 0 excluded 10 partial

```
1 from flask import Blueprint, request, jsonify, make_response
2 from flask_restful import Api, Resource
3 from http_status import HttpStatus
4 from models import orm, NotificationCategory, NotificationCategorySchema, Notification, NotificationSchema
5 from sqlalchemy.exc import SQLAlchemyError
6 from helpers import PaginationHelper
7 from flask_httpauth import HTTPBasicAuth
8 from flask import g
9 from models import User, UserSchema
10
11
12 auth = HTTPBasicAuth()
13
14
```

```
35 class UserResource(AuthenticationRequiredResource):
36     def get(self, id):
37         user = User.query.get_or_404(id)
38         result = user_schema.dump(user).data
39         return result
40
41
42 class UserListResource(Resource):
43     @auth.login_required
44     def get(self):
45         pagination_helper = PaginationHelper(
46             request,
47             query=User.query,
48             resource_for_url='service.userlistresource',
49             key_name='results',
50             schema=user_schema)
51         result = pagination_helper.paginate_query()
52         return result
53
54     def post(self):
55         user_dict = request.get_json()
56         if not user_dict:
57             response = {'user': 'No input data provided'}
58             return response, HttpStatus.bad_request_400.value
59         errors = user_schema.validate(user_dict)
60         if errors:
61             return errors, HttpStatus.bad_request_400.value
62         user_name = user_dict['name']
63         existing_user = User.query.filter_by(name=user_name).first()
```

Chapter 5: Developing RESTful APIs with Django 2.1



```
games_service -- bash -- 114x35
(Django01) Gastons-MacBook-Pro:games_service gaston$ sqlite3 db.sqlite3 "SELECT * FROM games_game ORDER BY name;"
1|2018-10-24 17:47:44.482989|PAW Patrol: On A Roll!|2018-10-24 17:47:30.177610|E (Everyone)|0|0
2|2018-10-24 17:48:13.018438|Spider-Man|2018-10-24 17:47:30.177610|T (Teen)|0|0
(Django01) Gastons-MacBook-Pro:games_service gaston$
```

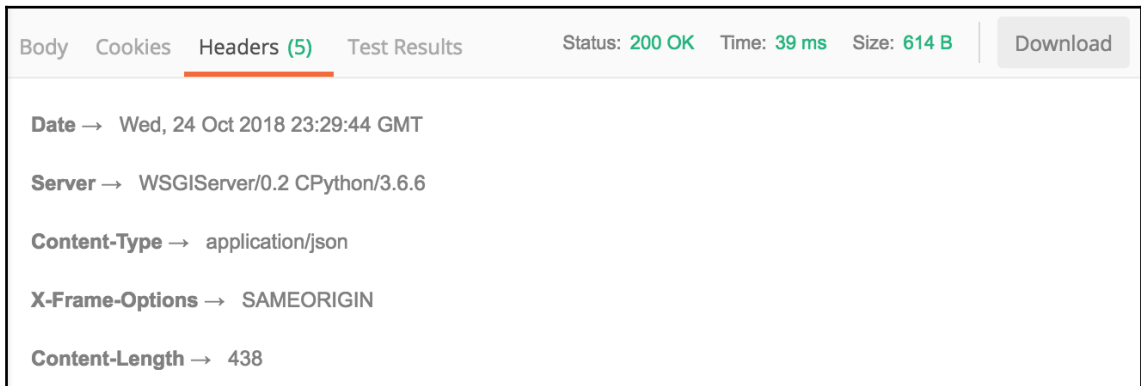
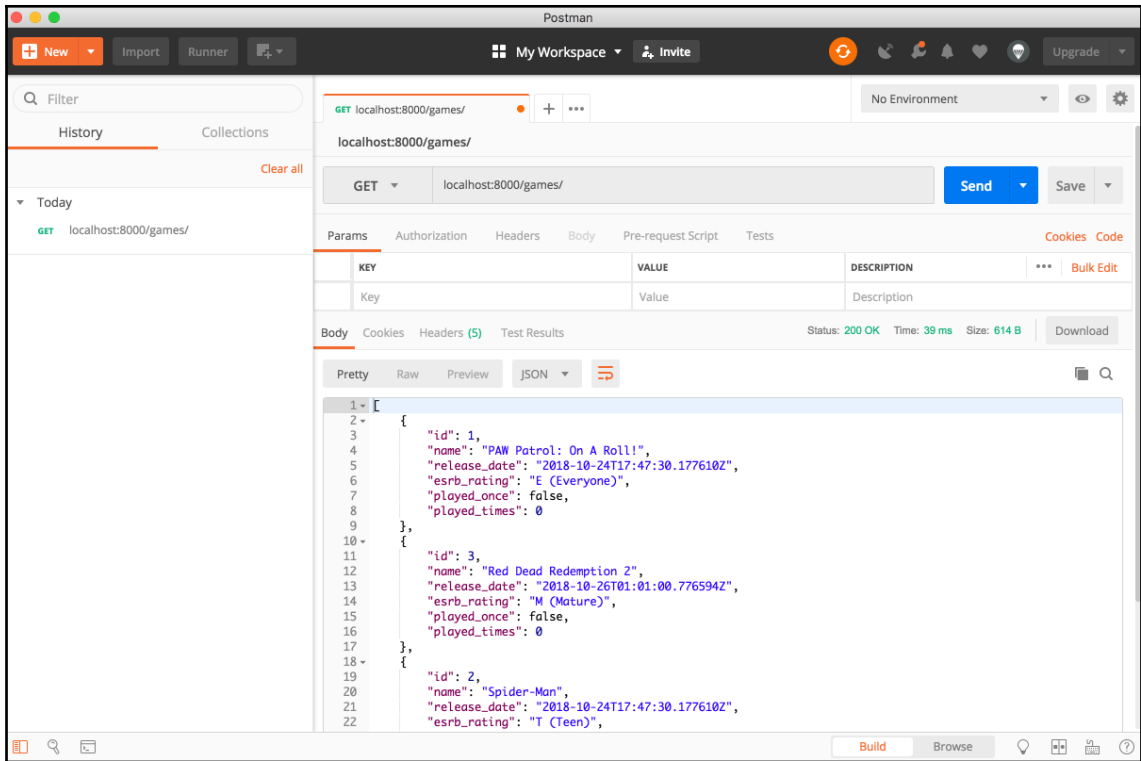
```
games_service -- Python - Python manage.py runserver -- 52x42
Performing system checks...

System check identified no issues (0 silenced).
October 24, 2018 - 20:31:02
Django version 2.1.2, using settings 'games_service.settings'
Starting development server at http://127.0.0.1:8000 /
Quit the server with CONTROL-C.
[24/Oct/2018 20:31:21] "GET /games/ HTTP/1.1" 200 438
[]

games_service -- bash -- 76x40
(Django01) Gastons-MacBook-Pro:games_service gaston$ http ":8000/games/"
HTTP/1.1 200 OK
Content-Length: 438
Content-Type: application/json
Date: Wed, 24 Oct 2018 20:31:21 GMT
Server: WSGIServer/0.2 CPython/3.6.2
X-Frame-Options: SAMEORIGIN

[
  {
    "esrb_rating": "E (Everyone)",
    "id": 1,
    "name": "PAW Patrol: On A Roll!",
    "played_once": false,
    "played_times": 0,
    "release_date": "2018-10-24T17:47:30.177610Z"
  },
  {
    "esrb_rating": "M (Mature)",
    "id": 3,
    "name": "Red Dead Redemption 2",
    "played_once": false,
    "played_times": 0,
    "release_date": "2018-10-26T01:01:00.776594Z"
  },
  {
    "esrb_rating": "T (Teen)",
    "id": 2,
    "name": "Spider-Man",
    "played_once": false,
    "played_times": 0,
    "release_date": "2018-10-24T17:47:30.177610Z"
  }
]

(Django01) Gastons-MacBook-Pro:games_service gaston$
```



GET localhost:8000/games/ POST localhost:8000/games/ No Environment

localhost:8000/games/

POST localhost:8000/games/ Send Save

Params Authorization Headers (1) Body Pre-request Script Tests Cookies Code

form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {
2   "name": "Crazy Kong 2019",
3   "esrb_rating": "E (Everyone)",
4   "release_date": "2019-02-01T03:02:00.776594Z"
5 }
6
```

Body Cookies Headers (5) Test Results Status: 201 Created Time: 35 ms Size: 325 B Download

Pretty Raw Preview JSON

```
1 {
2   "id": 5,
3   "name": "Crazy Kong 2019",
4   "release_date": "2019-02-01T03:02:00.776594Z",
5   "esrb_rating": "E (Everyone)",
6   "played_once": false,
7   "played_times": 0
8 }
```

Chapter 6: Working with Class-Based Views and Hyperlinked APIs in Django 2.1

```
games_service — Python • Python manage.py runserver — 135x42
(Django01) Gastons-MacBook-Pro:games_service gaston$ python manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).
October 25, 2018 - 04:19:12
Django version 2.1.2, using settings 'games_service.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
Internal Server Error: /games/
Traceback (most recent call last):
  File "/Users/gaston/HillarPythonREST2/Django01/lib/python3.6/site-packages/django/core/handlers/exception.py", line 34, in inner
    response = get_response(request)
  File "/Users/gaston/HillarPythonREST2/Django01/lib/python3.6/site-packages/django/core/handlers/base.py", line 137, in _get_response
    "returned None instead." % (callback._module_, view_name)
ValueError: The view games.views.game_collection didn't return an HttpResponse object. It returned None instead.
[25/Oct/2018 04:19:17] "OPTIONS /games/ HTTP/1.1" 500 51566
```

```
localhost
Django REST framework

Game Collection

Game Collection [OPTIONS] [GET]

GET /games/

HTTP 200 OK
Allow: GET, POST, OPTIONS
Content-Type: application/json
Vary: Accept

[
  {
    "id": 5,
    "name": "Crazy Kong 2019",
    "release_date": "2019-02-01T03:02:00.776594Z",
    "esrb_rating": "E (Everyone)",
    "played_once": false,
    "played_times": 0
  },
  {
    "id": 6,
    "name": "Kingdom Hearts III",
    "release_date": "2019-01-25T03:02:00.776594Z",
    "esrb_rating": "E10+ (Everyone 10+)",
    "played_once": false,
    "played_times": 0
  }
]
```

Game Collection

OPTIONS

GET

POST /games/

HTTP 201 Created

Allow: GET, POST, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "id": 7,
  "name": "Assassin's Creed Origins",
  "release_date": "2018-01-10T03:02:00.776594Z",
  "esrb_rating": "M (Mature)",
  "played_once": false,
  "played_times": 0
}
```

Media type:

application/json

Content:

POST

Game Collection / Game Detail

Game Detail

OPTIONS

GET

GET /games/7/

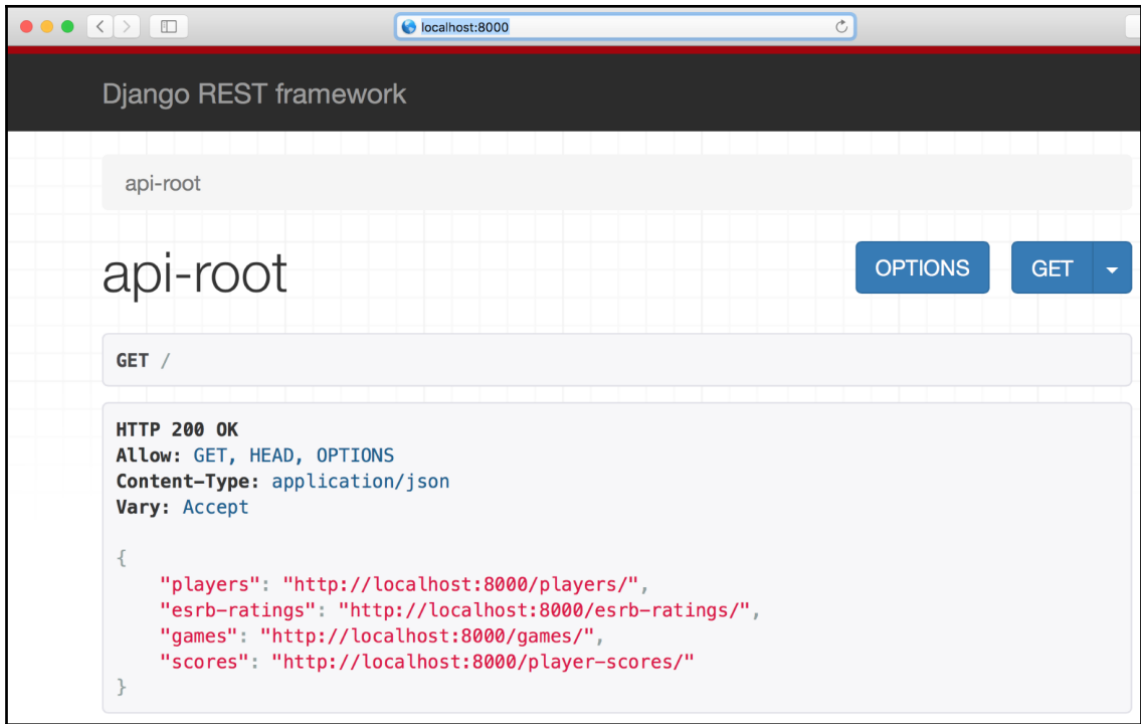
HTTP 200 OK

Allow: GET, PUT, POST, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "id": 7,
  "name": "Assassin's Creed Origins",
  "release_date": "2018-01-10T03:02:00.776594Z",
  "esrb_rating": "M (Mature)",
  "played_once": false,
  "played_times": 0
}
```



api-root / esrbrating-list

esrbrating-list

OPTIONS

GET



GET /esrb-ratings/

HTTP 200 OK

Allow: GET, POST, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

[]

Raw data

HTML form

Description

AO (Adults Only)

POST

esrbrating-list

OPTIONS

GET

POST /esrb-ratings/

HTTP 201 Created

Allow: GET, POST, HEAD, OPTIONS

Content-Type: application/json

Location: http://localhost:8000/esrb-ratings/1/

Vary: Accept

```
{
  "url": "http://localhost:8000/esrb-ratings/1/",
  "id": 1,
  "description": "AO (Adults Only)",
  "games": []
}
```

Raw data

HTML form

Description

POST

```
(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_games_user_name --dbname=django_games --command="SELECT * FROM games_esrbrating;"
 id | description
----+-----
  1 | AO (Adults Only)
  2 | T (Teen)
(2 rows)

(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_games_user_name --dbname=django_games --command="SELECT * FROM games_game;"
 id | created | name | release_date | played_once | played_times | esrb_rating_id
----+-----+-----+-----+-----+-----+-----
  1 | 2018-10-25 22:54:55.066995-03 | Battlefield V | 2017-04-30 22:02:00.776594-03 | f | 0 | 1
  2 | 2018-10-25 22:55:04.228801-03 | Mutant Football League: Dynasty Edition | 2018-10-20 00:02:00.776594-03 | f | 0 | 1
(2 rows)

(Django01) Gastons-MacBook-Pro:games_service gaston$
```

```
games_service -- -bash -- 156x40
(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_games_user_name --dbname=django_games --command="SELECT * FROM games_esrbrating;"
id | description
----+-----
 1 | A0 (Adults Only)
 2 | T (Teen)
(2 rows)

(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_games_user_name --dbname=django_games --command="SELECT * FROM games_game;"
id | created | name | release_date | played_once | played_times | esrb_rating_id
----+-----+-----+-----+-----+-----+-----
 1 | 2018-10-25 22:54:55.066995-03 | Battlefield V | 2017-04-30 22:02:00.776594-03 | f | 0 | 1
 2 | 2018-10-25 22:55:04.228801-03 | Mutant Football League: Dynasty Edition | 2018-10-20 00:02:00.776594-03 | f | 0 | 1
(2 rows)

(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_games_user_name --dbname=django_games --command="SELECT * FROM games_player;"
id | created | name | gender
----+-----+-----+-----
 1 | 2018-10-25 23:49:00.486847-03 | Gaston Hillar | M
 2 | 2018-10-25 23:49:06.908684-03 | Enzo Scocco | M
(2 rows)

(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_games_user_name --dbname=django_games --command="SELECT * FROM games_playerscore;"
id | score | score_date | game_id | player_id
----+-----+-----+-----+-----
 1 | 17500 | 2019-01-01 00:02:00.776594-03 | 1 | 1
 2 | 3225 | 2018-12-31 22:02:00.776594-03 | 2 | 1
 3 | 43200 | 2019-01-01 00:02:00.776594-03 | 1 | 2
 4 | 17420 | 2019-01-01 02:02:00.776594-03 | 2 | 2
(4 rows)

(Django01) Gastons-MacBook-Pro:games_service gaston$ █
```

Chapter 7: Improving Our API and Adding Authentication to it with Django

```
(Django01) Gastons-MacBook-Pro:bin gaston$ psql --username=your_games_user_name --dbname=django_games --command
="\d games_esrbrating"
      Table "public.games_esrbrating"
  Column |          Type          | Collation | Nullable |          Default
-----|-----|-----|-----|-----
 id      | integer                |           | not null | nextval('games_esrbrating_id_seq'::regclass)
 description | character varying(200) |           | not null |
Indexes:
 "games_esrbrating_pkey" PRIMARY KEY, btree (id)
 "games_esrbrating_description_360ae479_uniq" UNIQUE CONSTRAINT, btree (description)
 "games_esrbrating_description_360ae479_like" btree (description varchar_pattern_ops)
Referenced by:
 TABLE "games_game" CONSTRAINT "games_game_esrb_rating_id_59e5b261_fk_games_esrbrating_id" FOREIGN KEY (esrb
_rating_id) REFERENCES games_esrbrating(id) DEFERRABLE INITIALLY DEFERRED

(Django01) Gastons-MacBook-Pro:bin gaston$ psql --username=your_games_user_name --dbname=django_games --command
="\d games_game"
      Table "public.games_game"
  Column |          Type          | Collation | Nullable |          Default
-----|-----|-----|-----|-----
 id      | integer                |           | not null | nextval('games_game_id_seq'::regclass)
 created | timestamp with time zone |           | not null |
 name    | character varying(200) |           | not null |
 release_date | timestamp with time zone |           | not null |
 played_once | boolean                |           | not null |
 played_times | integer                |           | not null |
 esrb_rating_id | integer                |           | not null |
 owner_id | integer                |           | not null |
Indexes:
 "games_game_pkey" PRIMARY KEY, btree (id)
 "games_game_name_4e86b7e1_uniq" UNIQUE CONSTRAINT, btree (name)
 "games_game_esrb_rating_id_59e5b261" btree (esrb_rating_id)
 "games_game_name_4e86b7e1_like" btree (name varchar_pattern_ops)
 "games_game_owner_id_c554a59d" btree (owner_id)
Foreign-key constraints:
 "games_game_esrb_rating_id_59e5b261_fk_games_esrbrating_id" FOREIGN KEY (esrb_rating_id) REFERENCES games_e
srb_rating(id) DEFERRABLE INITIALLY DEFERRED
 "games_game_owner_id_c554a59d_fk_auth_user_id" FOREIGN KEY (owner_id) REFERENCES auth_user(id) DEFERRABLE I
NITIALLY DEFERRED
Referenced by:
 TABLE "games_playerscore" CONSTRAINT "games_playerscore_game_id_9e96c1f4_fk_games_game_id" FOREIGN KEY (gam
e_id) REFERENCES games_game(id) DEFERRABLE INITIALLY DEFERRED

(Django01) Gastons-MacBook-Pro:bin gaston$ psql --username=your_games_user_name --dbname=django_games --command
="\d games_player"
      Table "public.games_player"
  Column |          Type          | Collation | Nullable |          Default
-----|-----|-----|-----|-----
 id      | integer                |           | not null | nextval('games_player_id_seq'::regclass)
 created | timestamp with time zone |           | not null |
 name    | character varying(50)  |           | not null |
 gender  | character varying(2)   |           | not null |
Indexes:
 "games_player_pkey" PRIMARY KEY, btree (id)
 "games_player_name_16dd04a0_uniq" UNIQUE CONSTRAINT, btree (name)
 "games_player_name_16dd04a0_like" btree (name varchar_pattern_ops)
Referenced by:
 TABLE "games_playerscore" CONSTRAINT "games_playerscore_player_id_fc0a094c_fk_games_player_id" FOREIGN KEY
(player_id) REFERENCES games_player(id) DEFERRABLE INITIALLY DEFERRED

(Django01) Gastons-MacBook-Pro:bin gaston$
```


api-root / game-list

game-list

OPTIONS GET

<< 1 2 3 >>

GET /games/

HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

```
{
  "count": 12,
  "next": "http://localhost:8000/games/?limit=4&offset=4",
  "previous": null,
  "results": [
    {
      "url": "http://localhost:8000/games/4/",
      "esrb_rating": "T (Teen)",
      "name": "ARK: Survival Evolved",
      "release_date": "2018-06-21T03:02:00.776594Z",
      "played_once": false,
      "played_times": 0
    },
    {
```

```
games_service — -bash — 74x40
(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_
games_user_name --dbname=django_games --command="SELECT id FROM auth_user
WHERE username = 'your_games_super_user';"
 id
----
  1
(1 row)

(Django01) Gastons-MacBook-Pro:games_service gaston$
```

You are trying to add a non-nullable field 'owner' to game without a default; we can't do that (the database needs something to populate existing rows).

Please select a fix:

- 1) Provide a one-off default now (will be set on all existing rows with a null value for this column)
- 2) Quit, and let me add a default in models.py

Select an option: █

Select an option: 1

Please enter the default value now, as valid Python

The datetime and django.utils.timezone modules are available, so you can do e.g. timezone.now
Type 'exit' to exit this prompt

>>> █

```
(Django01) Gastons-MacBook-Pro:games_service gaston$ psql --username=your_games_user_name --dbname=django_games --command="SELECT * FROM games_game;"
```

id	created	name	release_date	played_once	played_times	esrb_rating_id	owner_id
1	2018-10-25 22:54:55.066995-03	Battlefield V	2017-04-30 22:02:00.776594-03	f	0	1	1
2	2018-10-25 22:55:04.228801-03	Mutant Football League: Dynasty Edition	2018-10-20 00:02:00.776594-03	t	10	1	1
3	2018-10-26 16:26:18.112533-03	Heavy Fire: Red Shadow	2018-06-21 00:02:00.776594-03	f	0	2	1
4	2018-10-26 16:26:23.811192-03	ARK: Survival Evolved	2018-06-21 00:02:00.776594-03	f	0	2	1
5	2018-10-26 16:26:29.171342-03	The Escapists 2	2018-06-21 00:02:00.776594-03	f	0	2	1
6	2018-10-26 16:26:34.34064-03	Honor and Duty: D-Day	2018-06-21 00:02:00.776594-03	f	0	2	1
7	2018-10-26 16:26:40.171755-03	Speed Brawl	2018-06-21 00:02:00.776594-03	f	0	2	1
8	2018-10-26 16:26:45.58564-03	Unearthing Mars 2	2018-06-21 00:02:00.776594-03	f	0	2	1
9	2018-10-26 16:26:49.765978-03	Super Street: The Game	2019-01-21 00:02:00.776594-03	f	0	2	1
10	2018-10-26 16:26:53.972909-03	Valkyria Chronicles 4	2019-01-21 00:02:00.776594-03	f	0	2	1
11	2018-10-26 16:26:58.767648-03	Tales of Vesperia: Definitive Edition	2019-01-21 00:02:00.776594-03	f	0	2	1
12	2018-10-26 16:27:03.600984-03	Moonfall Ultimate	2019-01-21 00:02:00.776594-03	f	0	2	1

(12 rows)

api-root

api-root

OPTIONS

GET ▾

GET /

HTTP 200 OK**Allow:** GET, HEAD, OPTIONS**Content-Type:** application/json**Vary:** Accept

```
{
  "users": "http://localhost:8000/users/",
  "players": "http://localhost:8000/players/",
  "esrb-ratings": "http://localhost:8000/esrb-ratings/",
  "games": "http://localhost:8000/games/",
  "scores": "http://localhost:8000/player-scores/"
}
```

api-root / game-list / game-detail

game-detail

[DELETE](#)[OPTIONS](#)[GET](#) ▾

DELETE /games/4/

HTTP 403 Forbidden

Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "detail": "You do not have permission to perform this action."
}
```

Chapter 8: Throttling, Filtering, Testing, and Deploying an API with Django 2.1

Filters ×

Field filters

Game name:

Player name:

Score:

Score date is greater than or equal to:

Score date is less than or equal to:

Score is greater than or equal to:

Score is less than or equal to:

Ordering

playerscore-list

Filters

OPTIONS

GET

GET /player-scores/?from_score_date=&game_name=&max_score=58000&min_score=1000&ordering=-score&player_name=Enzo+Scocco&score=&to_score_date=

HTTP 200 OK

Allow: GET, POST, HEAD, OPTIONS

Content-Type: application/json

Vary: Accept

```
{
  "count": 2,
  "next": null,
  "previous": null,
  "results": [
    {
      "url": "http://localhost:8000/player-scores/3/",
      "id": 3,
      "score": 43200,
      "score_date": "2019-01-01T03:02:00.776594Z",
      "player": "Enzo Scocco",
      "game": "Battlefield V"
    },
    {
      "url": "http://localhost:8000/player-scores/4/",
      "id": 4,
      "score": 17420,
      "score_date": "2019-01-01T05:02:00.776594Z",
      "player": "Enzo Scocco",
      "game": "Mutant Football League: Dynasty Edition"
    }
  ]
}
```

Chapter 9: Developing RESTful APIs with Pyramid 1.10

```
(Pyramid02) Gastons-MacBook-Pro:metrics gaston$ pserve development.ini
Starting server in PID 9908.
Serving on http://localhost:6543
Serving on http://localhost:6543
2018-11-08 23:56:09,506 DEBUG [metrics:104][waitress] route matched for
url http://localhost:6543/metrics/; route_name: 'metrics', path_info:
'/metrics/', pattern: '/metrics/', matchdict: {}, predicates: ''
4
2018-11-08 23:56:13,699 DEBUG [metrics:104][waitress] route matched for
url http://localhost:6543/metrics/; route_name: 'metrics', path_info:
'/metrics/', pattern: '/metrics/', matchdict: {}, predicates: ''
4
2018-11-08 23:56:18,877 DEBUG [metrics:104][waitress] route matched for
url http://localhost:6543/metrics/; route_name: 'metrics', path_info:
'/metrics/', pattern: '/metrics/', matchdict: {}, predicates: ''
[]

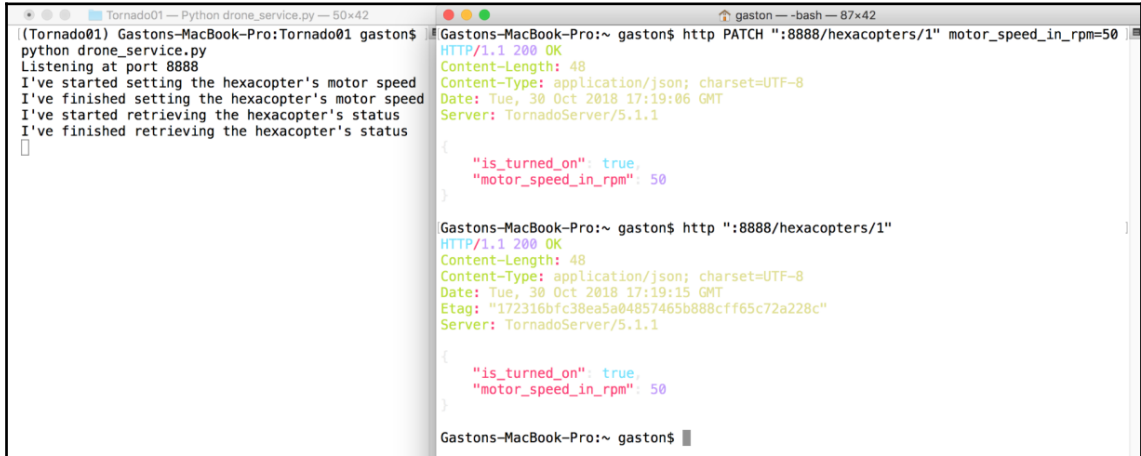
(Pyramid02) Gastons-MacBook-Pro:Pyramid02 gaston$ http POST ":6543/metrics/" status
='PADDLING' speed_in_mph=3 altitude_in_feet=3 water_temperature_in_f=59
HTTP/1.1 201 Created
Content-Length: 94
Content-Type: application/json; charset=UTF-8
Date: Fri, 09 Nov 2018 02:56:13 GMT
Server: waitress

{
  "altitude_in_feet": 3,
  "id": 2,
  "speed_in_mph": 3,
  "status": "PADDLING",
  "water_temperature_in_f": 59
}

(Pyramid02) Gastons-MacBook-Pro:Pyramid02 gaston$ http ":6543/metrics/"
HTTP/1.1 200 OK
Content-Length: 206
Content-Type: application/json
Date: Fri, 09 Nov 2018 02:56:18 GMT
Server: waitress

[
  {
    "altitude_in_feet": 2,
    "id": 1,
    "speed_in_mph": 1,
    "status": "IDLE",
    "water_temperature_in_f": 58
  },
  {
    "altitude_in_feet": 3,
    "id": 2,
    "speed_in_mph": 3,
    "status": "PADDLING",
    "water_temperature_in_f": 59
  }
]
```

Chapter 10: Developing RESTful APIs with Tornado 5.1.1

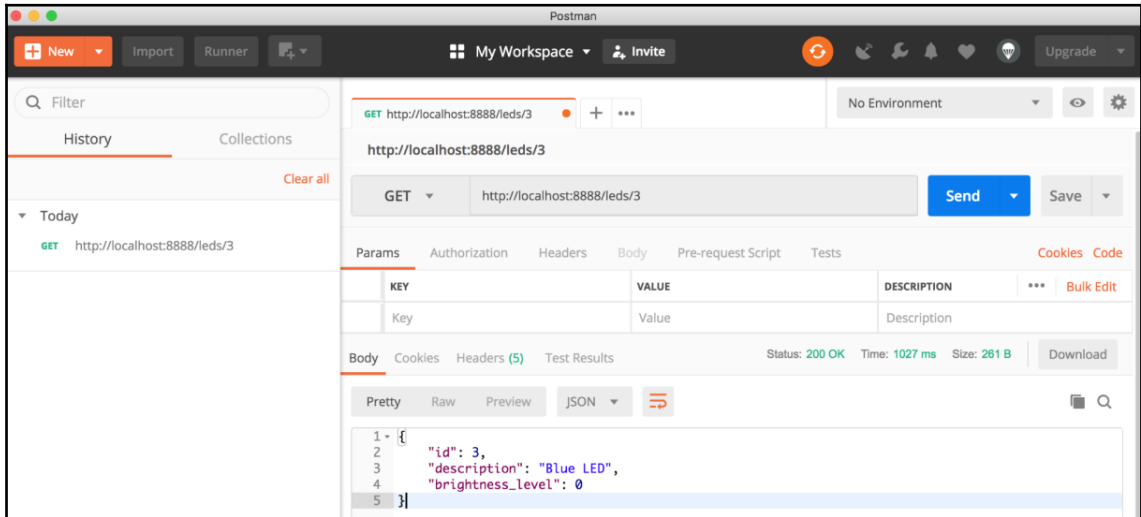


```
(Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ python drone_service.py
Listening at port 8888
I've started setting the hexacopter's motor speed
I've finished setting the hexacopter's motor speed
I've started retrieving the hexacopter's status
I've finished retrieving the hexacopter's status
[]

Gastons-MacBook-Pro:~ gaston$ http PATCH ":8888/hexacopters/1" motor_speed_in_rpm=50
HTTP/1.1 200 OK
Content-Length: 48
Content-Type: application/json; charset=UTF-8
Date: Tue, 30 Oct 2018 17:19:06 GMT
Server: TornadoServer/5.1.1
{
  "is_turned_on": true,
  "motor_speed_in_rpm": 50
}

Gastons-MacBook-Pro:~ gaston$ http ":8888/hexacopters/1"
HTTP/1.1 200 OK
Content-Length: 48
Content-Type: application/json; charset=UTF-8
Date: Tue, 30 Oct 2018 17:19:15 GMT
Etag: "172316bfc38ea5a04857465b888cff65c72a228c"
Server: TornadoServer/5.1.1
{
  "is_turned_on": true,
  "motor_speed_in_rpm": 50
}

Gastons-MacBook-Pro:~ gaston$
```



Postman interface showing a GET request to `http://localhost:8888/leds/3`. The response is a JSON object:

```
1 - {
2   "id": 3,
3   "description": "Blue LED",
4   "brightness_level": 0
5 }
```


Body Cookies **Headers (5)** Test Results Status: 200 OK Time: 1027 ms Size: 261 B Download

Server → TomadoServer/5.1.1

Content-Type → application/json; charset=UTF-8

Date → Tue, 30 Oct 2018 18:07:56 GMT

Etag → "de6fdc3029f06d52f59656c1cc5537a01302fbe9"

Content-Length → 59

http://localhost:8888/leds/3

PATCH http://localhost:8888/leds/3 Send Save

Params Authorization Headers (1) **Body** Pre-request Script Tests Cookies Code

form-data x-www-form-urlencoded raw binary JSON (application/json)

```
1 {
2   "brightness_level": 64
3 }
4
```

Body Cookies Headers (4) Test Results Status: 200 OK Time: 2007 ms Size: 212 B Download

Pretty Raw Preview JSON

```
1 {
2   "id": 3,
3   "description": "Blue LED",
4   "brightness_level": 64
5 }
```

15:23 Tue 30 Oct 25%

192.168.1.101:8888/leds/2 Go

GET HEAD POST PUT DELETE TRACE OPTIONS Basic Verbose

```
HTTP/1.1 200 OK
Server: TornadoServer/5.1.1
Content-Type: application/json; charset=UTF-8
Date: Tue, 30 Oct 2018 18:23:12 GMT
Etag: "c4d7a68c8bdf7ae8309acb3fdcea17f3328bfdc4"
Content-Length: 60

{"id": 2, "description": "Green LED", "brightness_level": 0}
```

Chapter 11: Working with Asynchronous Code, Testing, and Deploying an API with Tornado

```
(Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ python drone_service.py
Listening at port 8888
I've started setting the Red LED's brightness level
I've finished setting the Red LED's brightness level
I've started setting the Green LED's brightness level
I've finished setting the Green LED's brightness level
I've started setting the Blue LED's brightness level
I've finished setting the Blue LED's brightness level
[]

(Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ http PATCH ":8888/leds/1"
brightness_level=255
HTTP/1.1 200 OK
Content-Length: 60
Content-Type: application/json; charset=UTF-8
Date: Tue, 30 Oct 2018 21:51:05 GMT
Server: TornadoServer/5.1.1

{
  "brightness_level": 255,
  "description": "Red LED",
  "id": 1
}

(Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ http PATCH ":8888/leds/2"
brightness_level=128
HTTP/1.1 200 OK
Content-Length: 62
Content-Type: application/json; charset=UTF-8
Date: Tue, 30 Oct 2018 21:51:07 GMT
Server: TornadoServer/5.1.1

{
  "brightness_level": 128,
  "description": "Green LED",
  "id": 2
}

(Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ http PATCH ":8888/leds/3"
brightness_level=64
HTTP/1.1 200 OK
Content-Length: 60
Content-Type: application/json; charset=UTF-8
Date: Tue, 30 Oct 2018 21:51:09 GMT
Server: TornadoServer/5.1.1

{
  "brightness_level": 64,
  "description": "Blue LED",
  "id": 3
}

(Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$
```

```
Tornado01 — Python async_drone_service.py — 57x42
((Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ python
async_drone_service.py
Listening at port 8888
I've started setting the Blue LED's brightness level
I've started setting the Red LED's brightness level
I've started setting the Green LED's brightness level
I've finished setting the Blue LED's brightness level
I've finished setting the Red LED's brightness level
I've finished setting the Green LED's brightness level
[]

Tornado01 — -bash — 81x16
((Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ http PATCH ":8888/leds/1"
brightness_level=255
HTTP/1.1 200 OK
Content-Length: 60
Content-Type: application/json; charset=UTF-8
Date: Wed, 31 Oct 2018 03:33:02 GMT
Server: TornadoServer/5.1.1

{
  "brightness_level": 255,
  "description": "Red LED",
  "id": 1
}

Tornado01 — -bash — 81x15
((Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$ http PATCH ":8888/leds/2"
brightness_level=128
HTTP/1.1 200 OK
Content-Length: 62
Content-Type: application/json; charset=UTF-8
Date: Wed, 31 Oct 2018 03:33:02 GMT
Server: TornadoServer/5.1.1

{
  "brightness_level": 128,
  "description": "Green LED",
  "id": 2
}

Tornado01 — -bash — 81x13
HTTP/1.1 200 OK
Content-Length: 60
Content-Type: application/json; charset=UTF-8
Date: Wed, 31 Oct 2018 03:33:02 GMT
Server: TornadoServer/5.1.1

{
  "brightness_level": 64,
  "description": "Blue LED",
  "id": 3
}

(Tornado01) Gastons-MacBook-Pro:Tornado01 gaston$
```

Coverage report: 46%

filter...



<i>Module ↓</i>	<i>statements</i>	<i>missing</i>	<i>excluded</i>	<i>branches</i>	<i>partial</i>	<i>coverage</i>
async_drone_service.py	141	81	0	20	4	40%
drone.py	63	23	0	10	3	59%
Total	204	104	0	30	7	46%

coverage.py v4.5.1, created at 2018-10-31 15:06

Coverage for `async_drone_service.py` : 40%

141 statements

60 run

81 missing

0 excluded

4 partial

```
1 from http import HTTPStatus
2 from concurrent.futures import ThreadPoolExecutor
3 from datetime import date
4 from tornado import web, escape, ioloop, httpclient, gen
5 from tornado.concurrent import run_on_executor
6 from drone import Altimeter, Hexacopter, LightEmittingDiode, Drone
7
8
9 thread_pool_executor = ThreadPoolExecutor()
10 drone = Drone()
11
12
13 class AsyncHexacopterHandler(web.RequestHandler):
14     SUPPORTED_METHODS = ("GET", "PATCH")
15     HEXACOPTER_ID = 1
16     _thread_pool_executor = thread_pool_executor
17
18     @gen.coroutine
19     def get(self, id):
20         if int(id) is not self.__class__.HEXACOPTER_ID:
21             self.set_status(HTTPStatus.NOT_FOUND)
22             self.finish()
23             return
24         print("I've started retrieving the hexacopter's status")
25         hexacopter_status = yield self.retrieve_hexacopter_status()
26         print("I've finished retrieving the hexacopter's status")
27         response = {
28             'motor_speed_in_rpm': hexacopter_status.motor_speed,
29             'is_turned_on': hexacopter_status.is_turned_on,}
30         self.set_status(HTTPStatus.OK)
31         self.write(response)
32         self.finish()
33
34     @run_on_executor(executor="_thread_pool_executor")
35     def retrieve_hexacopter_status(self):
36         return drone.hexacopter.status
37
38     @gen.coroutine
39     def patch(self, id):
40         if int(id) is not self.__class__.HEXACOPTER_ID:
41             self.set_status(HTTPStatus.NOT_FOUND)
```

```

103 | @gen.coroutine
104 | def patch(self, id):
105 |     int_id = int(id)
106 |     if int_id not in drone.leds.keys(): 106→107
107 |         self.set_status(HTTPStatus.NOT_FOUND)
108 |         self.finish()
109 |         return
110 |     led = drone.leds[int_id]
111 |     request_data = escape.json_decode(self.request.body)
112 |     if ('brightness_level' not in request_data.keys()) or \ 112→114
113 |         (request_data['brightness_level'] is None):
114 |         self.set_status(HTTPStatus.BAD_REQUEST)
115 |         self.finish()
116 |         return
117 |     try:
118 |         brightness_level = int(request_data['brightness_level'])
119 |         print("I've started setting the {0}'s brightness level".format(led.description))
120 |         yield self.set_led_brightness_level(led, brightness_level)
121 |         print("I've finished setting the {0}'s brightness level".format(led.description))
122 |         response = {
123 |             'id': led.id,
124 |             'description': led.description,
125 |             'brightness_level': brightness_level}
126 |         self.set_status(HTTPStatus.OK)
127 |         self.write(response)
128 |         self.finish()
129 |     except ValueError as e:
130 |         print("I've failed setting the {0}'s brightness level".format(led.description))
131 |         self.set_status(HTTPStatus.BAD_REQUEST)
132 |         response = {
133 |             'error': e.args[0]}
134 |         self.write(response)
135 |         self.finish()
136 |

```