

🔑 master ▾



[challenges](#) / [backend.md](#)



[astabile](#) Update backend.md



👤 1 contributor

Raw

Blame



354 lines (283 sloc) 8.58 KB

- [Introduction](#)
- [Requirement](#)
- [Notes](#)
- [Expectations](#)
- [Problem Statement](#)
- [Project Setup](#)
- [API Example](#)
 - [Create a new Listing](#)
 - [Show All Listings](#)
 - [Show Listing](#)
 - [Update A Listing](#)
 - [Delete A Listing](#)
 - [Add Special price to Listing](#)
 - [Delete a Special price](#)
 - [Calculate Reservation Cost](#)

Introduction

As a software engineer, you have to provide a reliable **backend** system to clients. Your task here is to implement few endpoints `list/show/create` for a small **Airbnb** like rest api.

Requirement

1. We value a **clean, simple** working solution.
2. The solution must be production ready.
3. Good understanding for how git works.
4. Good understanding of **REST API's**.

Notes

- Source code must be pushed as git branch in the provided project repository.
- Your branch name should follow this scheme `challenge/lastname-firstname` .
- (Optional) Deploy as a public api to your own host.

Expectations

- This challenge should take around 4 to 6 hours to complete.
- Your code should be modular, each module should focus on doing one thing and do it well.
- Avoid over-engineering.

Problem Statement

For reference only.



Following this diagram we want to implement a small api service that our frontend users can consume. this restfull api should give us the ability to create new listings "homes" and assign them special prices for weekends and holidays. it must also allow our clients to calculate a checkout cost for any listing.

1. Must be a **RESTful** HTTP API
2. Must implement 4 endpoints with path, method, request and response body as specified
 - One **CRUD** endpoint to list/show/create/delete a listing [see sample](#)
 - One endpoint to add a special price to a listing [see sample](#)
 - One endpoint to delete a special price [see sample](#)
 - One endpoint to get the checkout cost for any given listing [see sample](#)
 - Checkout cost can be calculated on any given listing.
 - An error response should be sent if a user tries to select a past date as checkin or checkout.
 - checkin date should be always before checkout date.
 - clients can not book a listing for more then **28** days.
3. A Database must be used (we use **MySQL**).
4. All responses should be in a **JSON** format no matter in success or failure situations.

Project Setup

You need to create a new branch to start working on, run the following command :

```
git checkout -b challenge/lastname-firstname challenge
```

You should provide the project skelton, included
routes , models , migration , controllers , tests .

API Example

Create a new Listing

- Method: POST
- URL path: /api/listings
- Request body:

```
{
  "name": "Black Raven",
  "description": "Morbi porttitor lorem id ligula. Suspendisse ornare consequat",
  "adults": 3,
  "children": 2,
  "is_pets_allowed": true,
  "base_price": 195.62,
  "cleaning_fee": 95.82,
  "image_url": "http://dummyimage.com/126x173.bmp/cc0000/ffffff",
  "weekly_discount": 0.77,
  "monthly_discount": 0.61
}
```



- Response:

Header: HTTP 200 Body:

```
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0851",
  "owner_id": "d701748f-6c54-4b01-90e6-d701748f0822",
  "name": "Black Raven",
  "slug": "black-raven",
  "description": "Morbi porttitor lorem id ligula. Suspendisse ornare consequat",
  "adults": 3,
  "children": 2,
  "is_pets_allowed": true,
  "base_price": 195.62,
```

```
"cleaning_fee": 95.82,
"image_url": "http://dummyimage.com/126x173.bmp/cc0000/ffffff",
"weekly_discount": 0.77,
"monthly_discount": 0.61,
"special_prices": []
}
```

- Tips:

If you find this **json** format strange to you, read about json api specs [link to jsonapi.org!](https://jsonapi.org/).

[↑](#) back to top

Show All Listings

- Method: GET
- URL path: /api/listings
- Response:

Header: HTTP 200 Body:

```
[
  {
    "id": "28eed9aa-c27d-4217-ab21-ad65ead3a2aa",
    "owner_id": "59f6d752-97cf-414e-a794-42794ac7511a",
    "name": "Warner",
    "slug": "revolutionize-warner",
    "description": "Maecenas ut massa quis augue luctus tincidunt.",
    "adults": 10,
    "children": 2,
    "is_pets_allowed": true,
    "base_price": 95.38,
    "cleaning_fee": 4.33,
    "image_url": "http://dummyimage.com/241x240.jpg/ff4444/ffffff",
    "weekly_discount": 0.13,
    "monthly_discount": 0.23,
    "special_prices": []
  },
  {
    "id": "b3a6e269-d0fa-4408-89b1-fe2e48963177",
    "owner_id": "e0b227da-dc6e-402c-8172-9d950ece4707",
    "name": "Burrows White",
    "slug": "burrows-white",
    "description": "Phasellus in felis. Donec semper sapien a libero. Nam du:",
    "adults": 9,
    "children": 2,
    "is_pets_allowed": false,
```

```
"base_price": 160.51,
"cleaning_fee": 31.45,
"image_url": "http://dummyimage.com/149x114.png/ff4444/ffffff",
"weekly_discount": 0.77,
"monthly_discount": 0.36,
"special_prices": [
  {
    "date": "2019-10-12",
    "base_price": 40.51
  },
  {
    "date": "2019-10-13",
    "base_price": 80
  }
]
}
```



[↑ back to top](#)

Show Listing

- Method: GET
- URL path: /api/listings/:uuid
- Response:

Header: HTTP 200 Body:

```
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0851",
  "owner_id": "d701748f-6c54-4b01-90e6-d701748f0822",
  "name": "Black Raven",
  "slug": "black-raven",
  "description": "Morbi porttitor lorem id ligula. Suspendisse ornare consequat",
  "adults": 3,
  "children": 2,
  "is_pets_allowed": true,
  "base_price": 195.62,
  "cleaning_fee": 95.82,
  "image_url": "http://dummyimage.com/126x173.bmp/cc0000/ffffff",
  "weekly_discount": 0.77,
  "monthly_discount": 0.61,
  "special_prices": []
}
```



[↑ back to top](#)

Update A Listing

- Method: PUT
- URL path: /api/listings/:uuid
- Request body:

```
{
  "name": "Black Raven",
  "description": "Morbi porttitor lorem id ligula",
  "adults": 3,
  "children": 2,
  "is_pets_allowed": true,
  "base_price": 195.62,
  "cleaning_fee": 95.82,
  "image_url": "http://dummyimage.com/126x173.bmp/cc0000/ffffff",
  "weekly_discount": 0,
  "monthly_discount": 0.61
}
```

- Response:

Header: HTTP 200 Body:

```
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0851",
  "owner_id": "d701748f-6c54-4b01-90e6-d701748f0822",
  "name": "Black Raven",
  "slug": "black-raven",
  "description": "Morbi porttitor lorem id ligula",
  "adults": 3,
  "children": 2,
  "is_pets_allowed": true,
  "base_price": 195.62,
  "cleaning_fee": 95.82,
  "image_url": "http://dummyimage.com/126x173.bmp/cc0000/ffffff",
  "weekly_discount": 0,
  "monthly_discount": 0.61,
  "special_prices": []
}
```

[↑](#) back to top

Delete A Listing

- Method: DELETE
- URL path: /api/listings/:uuid

- Response:

Header: HTTP 200 Body:

```
{
  "id": "d290f1ee-6c54-4b01-90e6-d701748f0851"
}
```

[↑](#) [back to top](#)

Add Special price to Listing

- Method: POST
- URL path: /api/listings/:uuid/special-prices
- Request body:

```
{
  "date": "2019-12-06",
  "price": 95.82
}
```

- Response: Header: HTTP 200 Body:

```
{
  "id": "d290f1ee-6c52-4b02-90e6-d701748f9854",
  "date": "2019-12-06",
  "price": 95.82
}
```

Delete a Special price

- Method: DELETE
- URL path: /api/listings/:uuid/special-prices/:uuid
- Response: Header: HTTP 200 Body:

```
{
  "id": "d290f1ee-6c52-4b02-90e6-d701748f9854"
}
```

[↑](#) [back to top](#)

Calculate Reservation Cost

- Method: GET
- URL path: /api/listings/:uuid/checkout
- Request body:

```
{
  "checkin": "2019-12-06",
  "checkout": "2019-12-10"
}
```

- Response: Header: HTTP 200 Body:

```
{
  "nights_count": 4,
  "nights_cost": 95.82,
  "discount": 13.82,
  "cleaning_fee": 3.82,
  "total": 112.95
}
```

[↑](#) back to top