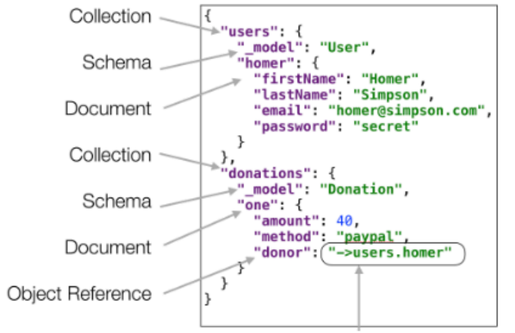
Seeding Mongo Database

Mongoose Seeding



Seeding the database can simplify exploratory development, prepopulating the database with simple test data during development.

Motivation for Database Seeding

- Pre-populating the database can enhance developer productivity
- It facilitates simple exploring of various scenarios
- Is particularly useful in establishing normalised data models with inter-document relationships
- Can also be used to pre-configure database for production

Mongoose seeders on NPM

- Variety of modules available
- Most fairly simple

mongoose-seed public

Seed data population for Mongoose

mongoose-seed lets you populate and clear MongoDB documents with all the benefits of Mongoose validation

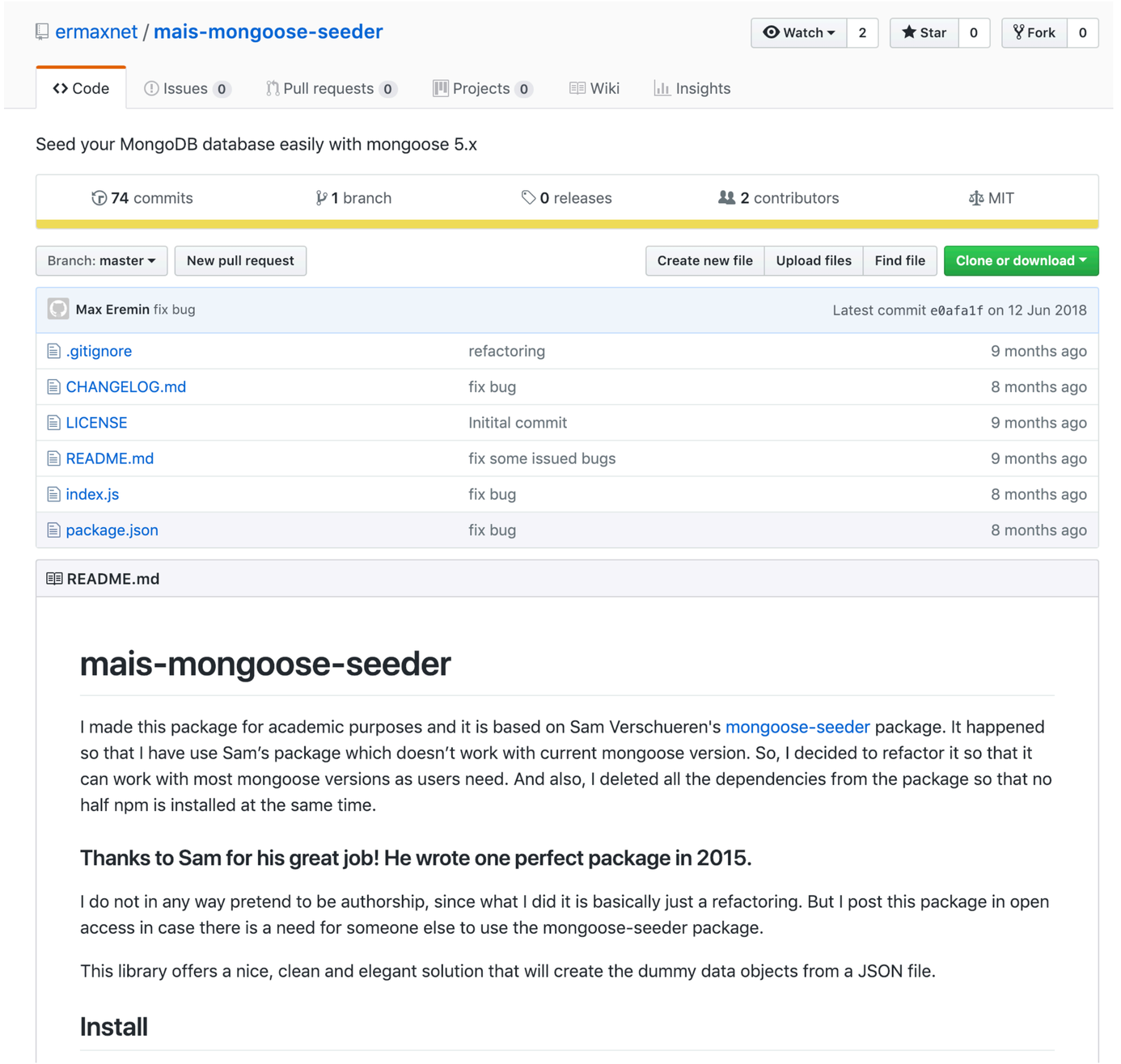
mongoose-seed-plus public

Seed data population for Mongoose

mongoose-seed-plus lets you populate and clear MongoDB documents with all the benefits of Mongoose validation

https://github.com/ermaxnet/mais-mongoose-seeder

- mongoose-seeder loads from an enhanced JSON file
- Includes special notation for loading relationships between documents



ermaxnet / mais-mongoose-seeder

Watch 2 | Star 0 | Fork 0

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Insights

Seed your MongoDB database easily with mongoose 5.x

74 commits | 1 branch | 0 releases | 2 contributors | MIT

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

Max Eremin fix bug — Latest commit e0afa1f on 12 Jun 2018

.gitignore	refactoring	9 months ago
CHANGELOG.md	fix bug	8 months ago
LICENSE	Inititial commit	9 months ago
README.md	fix some issued bugs	9 months ago
index.js	fix bug	8 months ago
package.json	fix bug	8 months ago

README.md

mais-mongoose-seeder

I made this package for academic purposes and it is based on Sam Verschueren's mongoose-seeder package. It happened so that I have use Sam's package which doesn't work with current mongoose version. So, I decided to refactor it so that it can work with most mongoose versions as users need. And also, I deleted all the dependencies from the package so that no half npm is installed at the same time.

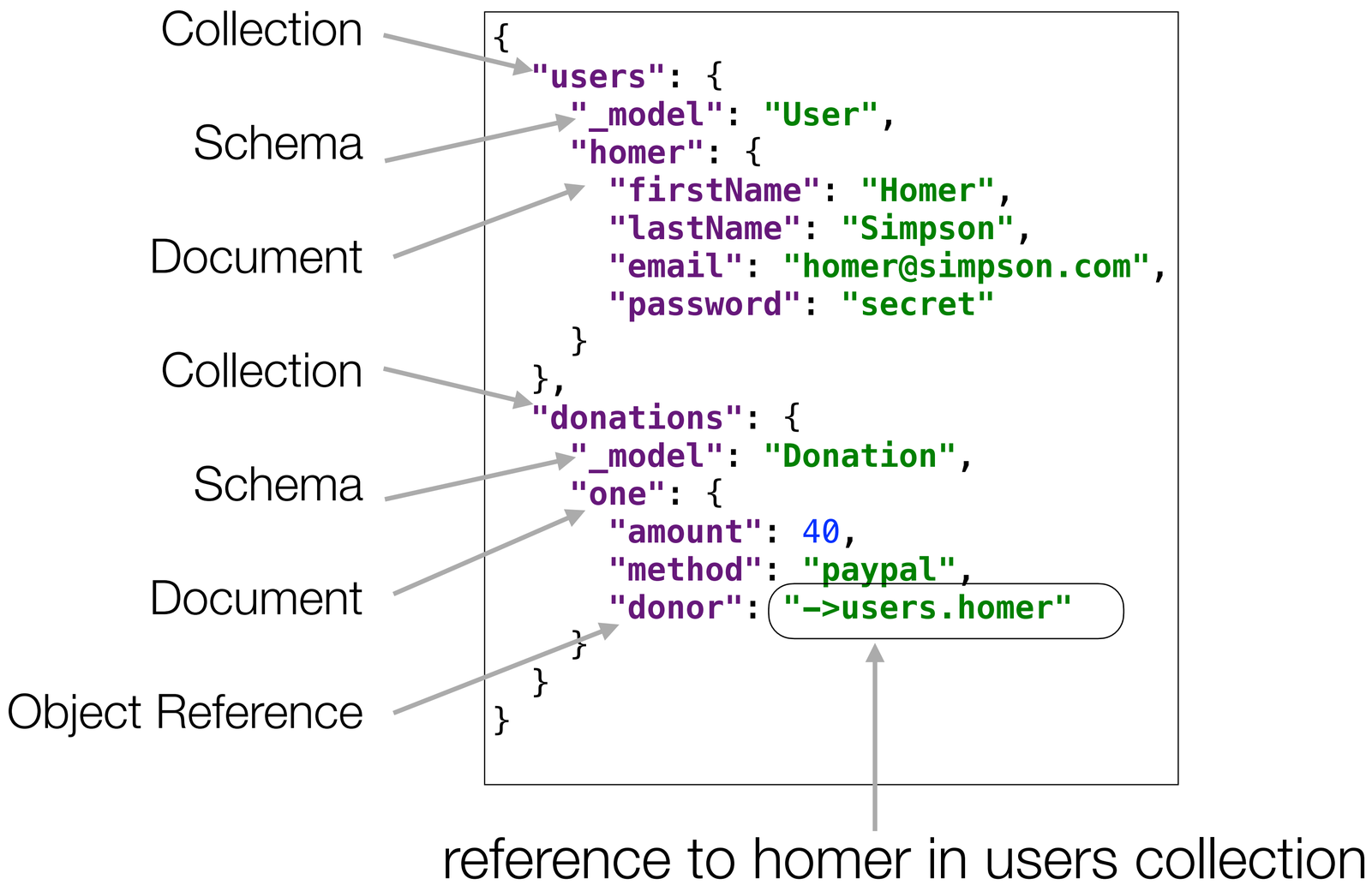
Thanks to Sam for his great job! He wrote one perfect package in 2015.

I do not in any way pretend to be authorship, since what I did it is basically just a refactoring. But I post this package in open access in case there is a need for someone else to use the mongoose-seeder package.

This library offers a nice, clean and elegant solution that will create the dummy data objects from a JSON file.

Install

initdata.json



seed function

- On application startup:
  - Delete any existing data in the collections
  - Populate the database on initial connection during startup

```
async function seed() {
  var seeder = require('mais-mongoose-seeder')(Mongoose);
  const data = require('./initdata.json');
  const Donation = require('./donation');
  const User = require('./user');
  const dbData = await seeder.seed(data, { dropDatabase: false, dropCollections: true });
  console.log(dbData);
}
```

seed function

```
var seeder = require('mais-mongoose-seeder')(Mongoose);
const data = require('./initdata.json');
const Donation = require('./donation');
const User = require('./user');
```

- import
  - the seeder library
  - the initial data json
  - the Schemas
    - Donation
    - User

seed function

- Call the seed component
  - pass ‘data’ loaded from JSON file
  - options:
    - keep the database
    - delete contents of all collections

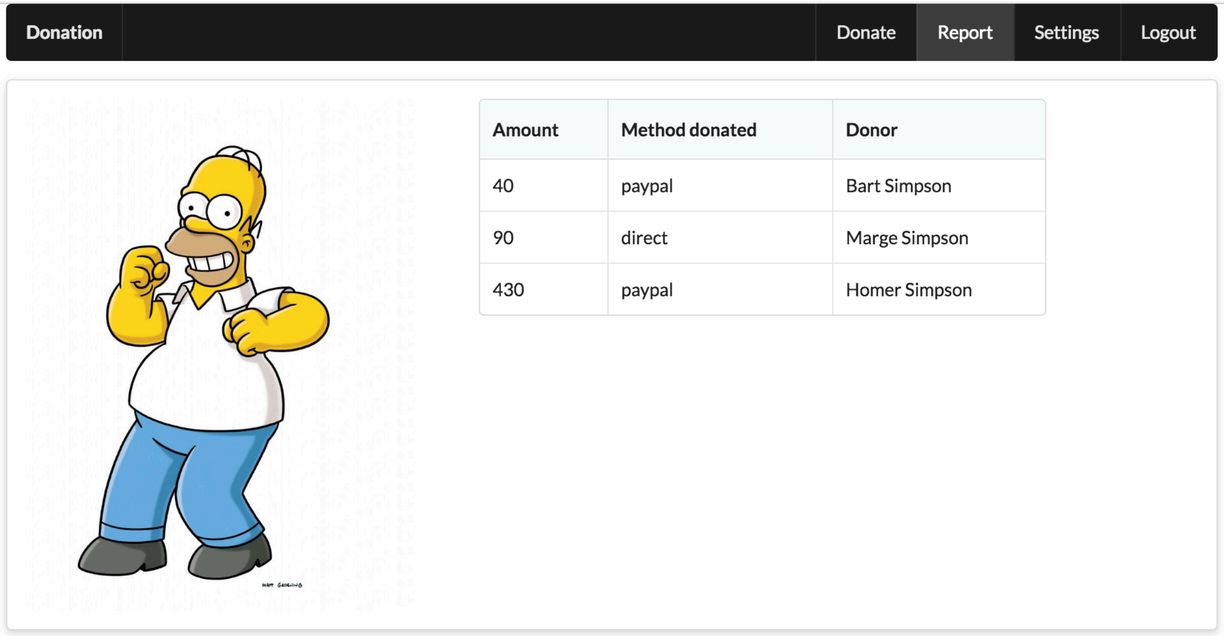
```
...
const dbData = await seeder.seed(data, { dropDatabase: false, dropCollections: true });
...
```

db.js

```
async function seed() {
  var seeder = require('mais-mongoose-seeder')(Mongoose);
  const data = require('./initdata.json');
  const Donation = require('./donation');
  const User = require('./user');
  const dbData = await seeder.seed(data, { dropDatabase: false, dropCollections: true });
  console.log(dbData);
}
```

```
db.once('open', function() {
  console.log(`database connected to ${this.name} on ${this.host}`);
  seed();
})
```

initdata.json version 2



- We can log in and view basic app functions without having to signup/ login and make donations etc…

```
{
  "users": {
    "_model": "User",
    "homer": {
      "firstName": "Homer",
      "lastName": "Simpson",
      "email": "homer@simpson.com",
      "password": "secret"
    },
    "marge": {
      "firstName": "Marge",
      "lastName": "Simpson",
      "email": "marge@simpson.com",
      "password": "secret"
    },
    "bart": {
      "firstName": "Bart",
      "lastName": "Simpson",
      "email": "bart@simpson.com",
      "password": "secret"
    }
  },
  "donations": {
    "_model": "Donation",
    "one": {
      "amount": 40,
      "method": "paypal",
      "donor": "->users.bart"
    },
    "two": {
      "amount": 90,
      "method": "direct",
      "donor": "->users.marge"
    },
    "three": {
      "amount": 430,
      "method": "paypal",
      "donor": "->users.homer"
    }
  }
}
```