FeedBurner - An Overview

The "Feed Burner" is a Symfony based application to read any kind of ATOM or RSS feed and import the data to the database. Users can add feed links and import the data individually.

Terms used in the documentation

1) FeedBurner : Application name
2) Feeders : ATOM/RSS feed links
3) Feeds : Individual entries in the feed links

Database Schema

The relational database schema which has been implemented in the project has been modelled and created using PHPMyadmin. The schema is implemented in the application and models are created accordIng to the structure and different Components of atom/other feeds.

The application is based on the relational database schema modelled as explained.

Reading feeds

The application reads and imports both the atom and rss feeds. Once the import is done, the data will be stored with a unique key and if the tool is run again , the already existing feeds will be updated and new feeds will be added. Multiple feeders, both ATOM and RSS feeder can be read and imported to the database.

Automation tests

I have added various functional and isolation tests to ensure reliability and accuracy in the application. I have used PHPUnit to implement these tests with symfony.

The application has a simple user interface to perform the following,

1) Listing the feeder and feeds.
2) Forms to add and edit feeder.
3) Rating for each feed.
4) Detail view for each feed.

Database Extension

The database schema is extended in such a way that the user can add his rating for each feed. Moreover, the database and front end is designed to allow user to add his/her own feeders and import the data. The database will be extended according to the new incoming feeds.

Caching

The application has a standard symfony caching method which will ensure the faster loading of feeds in the front end.

Error handling

The application also ensures the validity of all incoming feeders to avoid any unexpected errors. Two layer error Handling is done

1) While adding the feeder links, it validates the feeder contents
2) Before importing the feeder links to the database, the feeder link is again verified.

How to Install and Configure FeedBurner

The installation guidelines are also added to projects's ReadMe file.

Setting up an Existing Symfony Project

In addition to creating a new Symfony project, you will also work on projects already created by other developers. In that case, you only need to get the project code and install the dependencies with Composer. Setup your project with the following commands

```
$ cd /feed_burner/
$ composer update
```

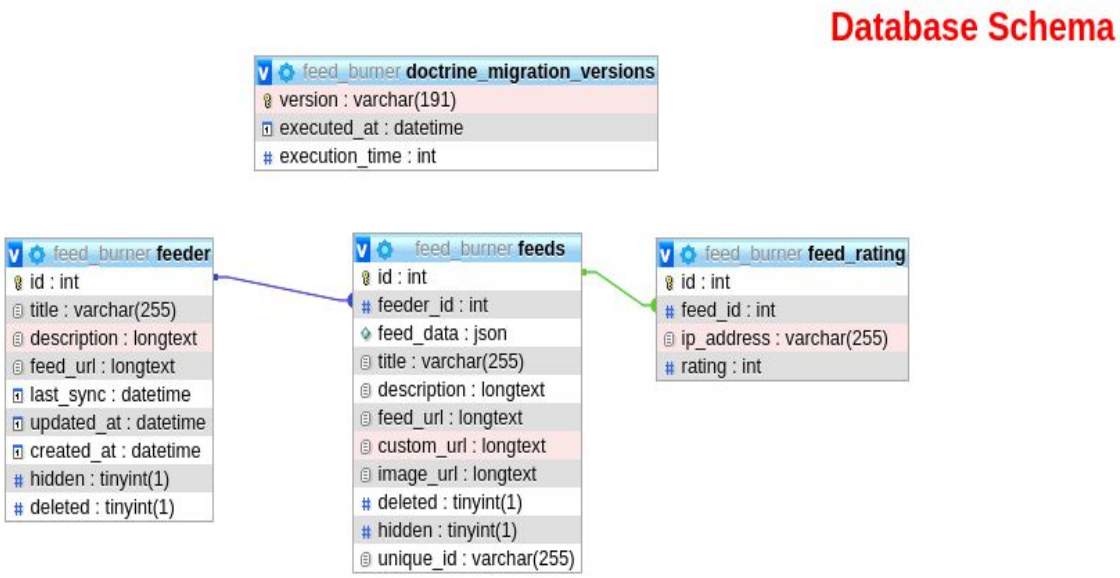
Usage

You'll probably also need to customize your .env file and do a few other project-specific tasks (e.g. creating a database). When working on a existing Symfony application for the first time, it may be useful to run this command which displays information about the project:

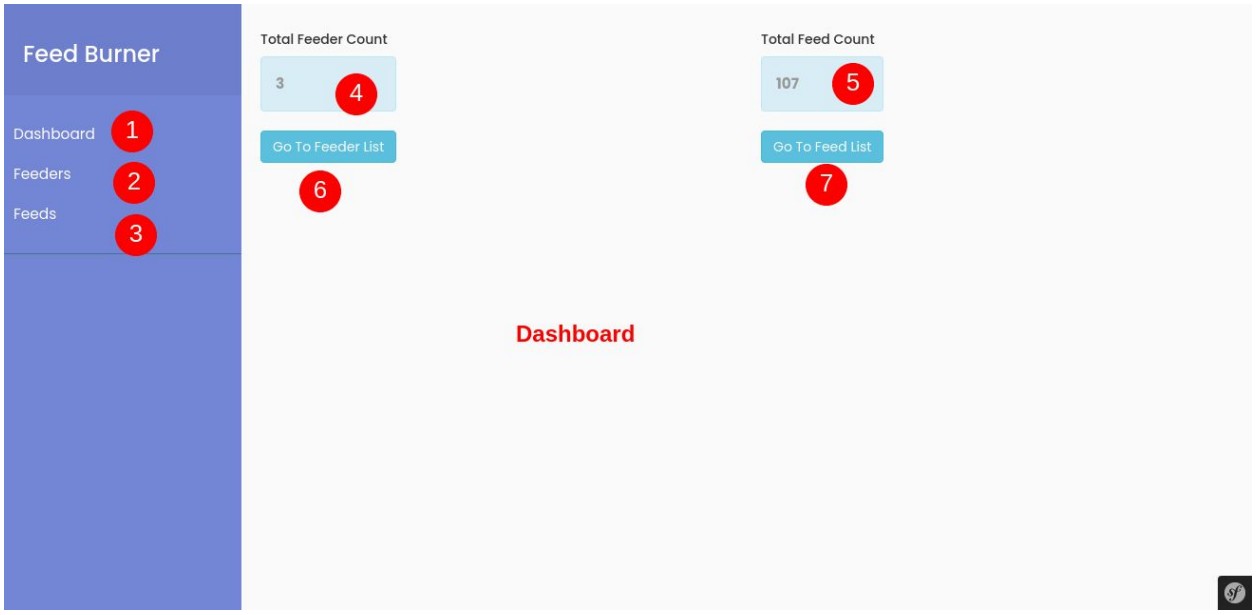
```
$ php bin/console doctrine:database:create
$ php bin/console make:migration
$ php bin/console doctrine:migrations:migrate
$ symfony server:start
```

Schema and User Interfaces

1. Database Schema



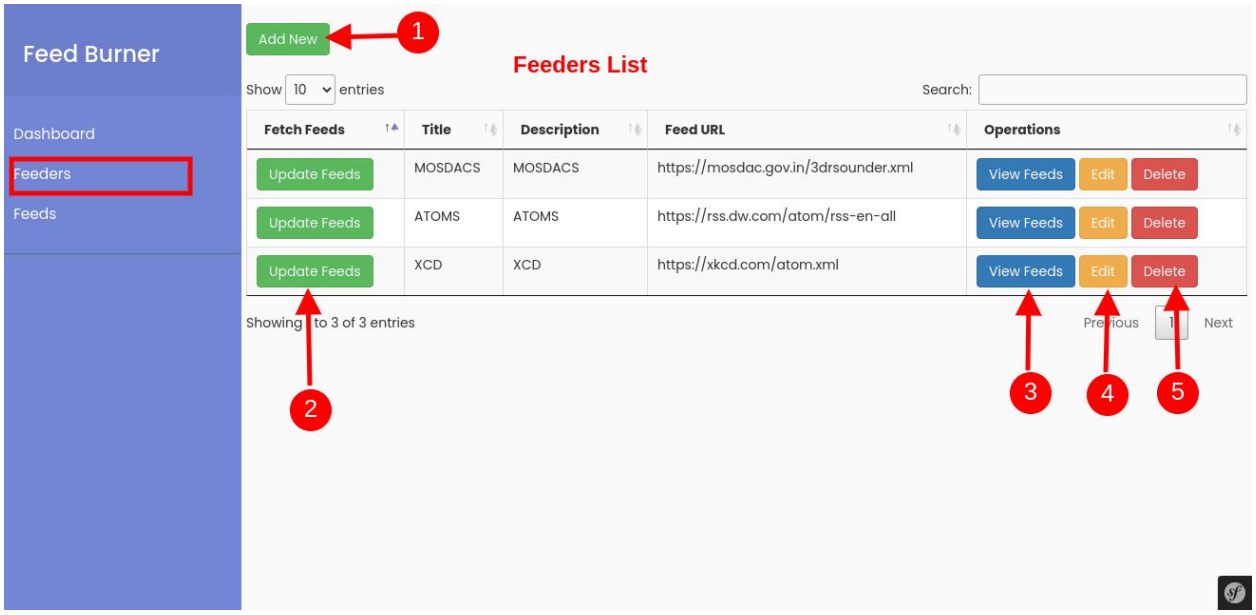
2. Dashboard



FeedBurner applications have an informative dashboard with all the details about the number of feeder links and total number of feeds available on the system.

1) Dashboard : Link which redirects to the dashboard page.
2) Feeders : Side menu link to display the list of all the feeders available on the system.
3) Feeds : Side menu link to display all the feeds imported from the whole feeders.
4) Feeder Count : Here the user will get the total number of all active feeder links.
5) Feed Count : Here the user will get the total number of all active feeds.
6) Feeder Link : Link to the list page for the feeders.
7) Feed Link : Link to the list page for the feeds.

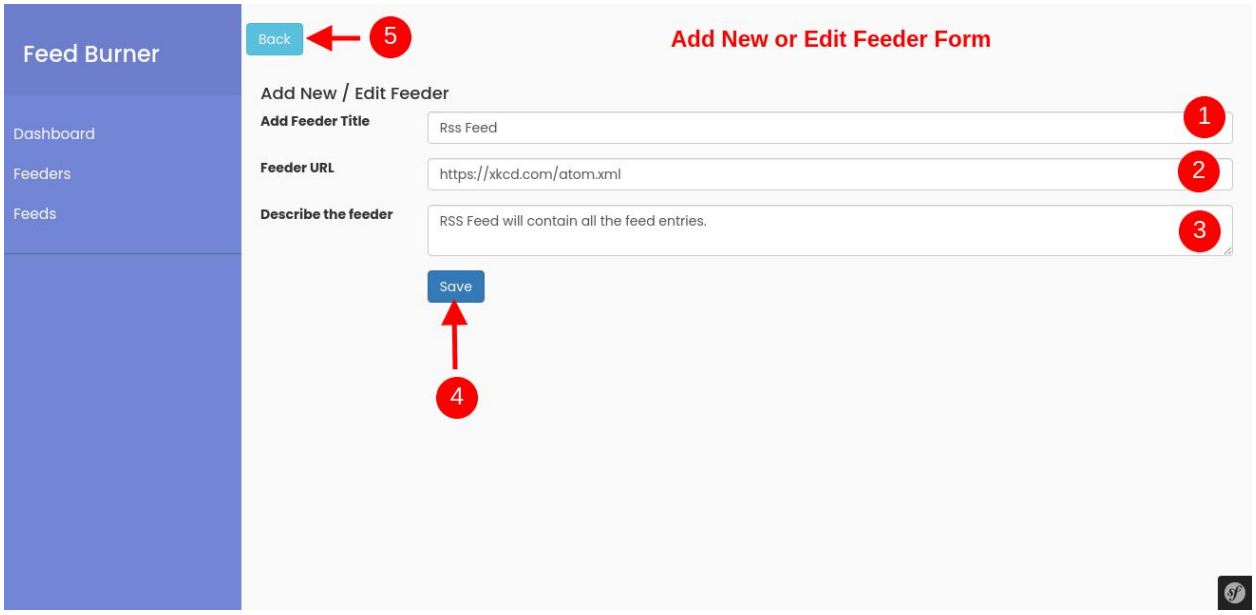
3. Feeder List



Application will provide an informative view for all the feeders available on the system. This page is self - explanatory with meaningful menu arrangements and labeling.

1) Add New: This is a link to another page where users can add new feeder links and other important details.
2) Fetch Feed / Update feeds: With this button, users can run the program to fetch the feeds from the corresponding feed URL. If the user has already executed the program once, the user can update the feeds with the same link.
3) View Feeds : Users can view the list of all the feeds associated with the Feeder with this button. Users can see all the related details of the feeds from this link.
4) Edit Feeder : Users can edit the feeder data with this link. Users will get a form to edit and update the feeder deatils.
5) Delete : Users can delete the feeder and associated feeds from the application with this button.

4. Add new Feeder / Edit Feeder data

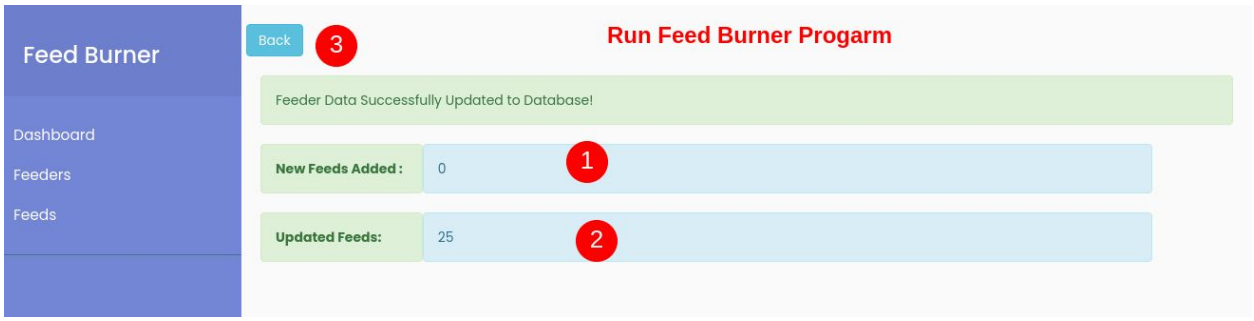


Users can add new feeders with this simple form.

1) Add feeder title : User can add custom title for this feeder.
2) Add feeder URL : User add valid feeder URL with this field. Link will be validated for correct feeds and normal URL validations.
3) Describe the feed : Users can enter the basic details about the feeder here.
4) Save button : When the user clicks on the button , the data will be validated and added to the database.
5) Back button : Users can move back to the previous page with this button.

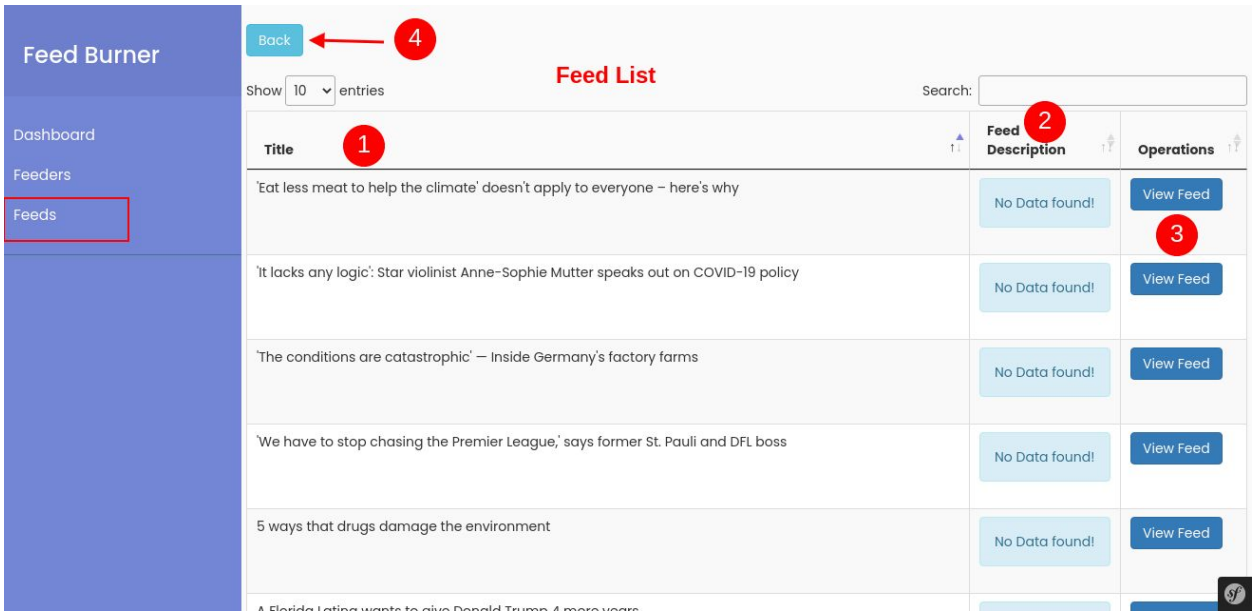
5. Run Feeder Link

Feedburner application will automatically detect the feeder type which is Atom/RSS feed and read the XML data and import the data to the database. Once the data is imported the user will get a screen below.



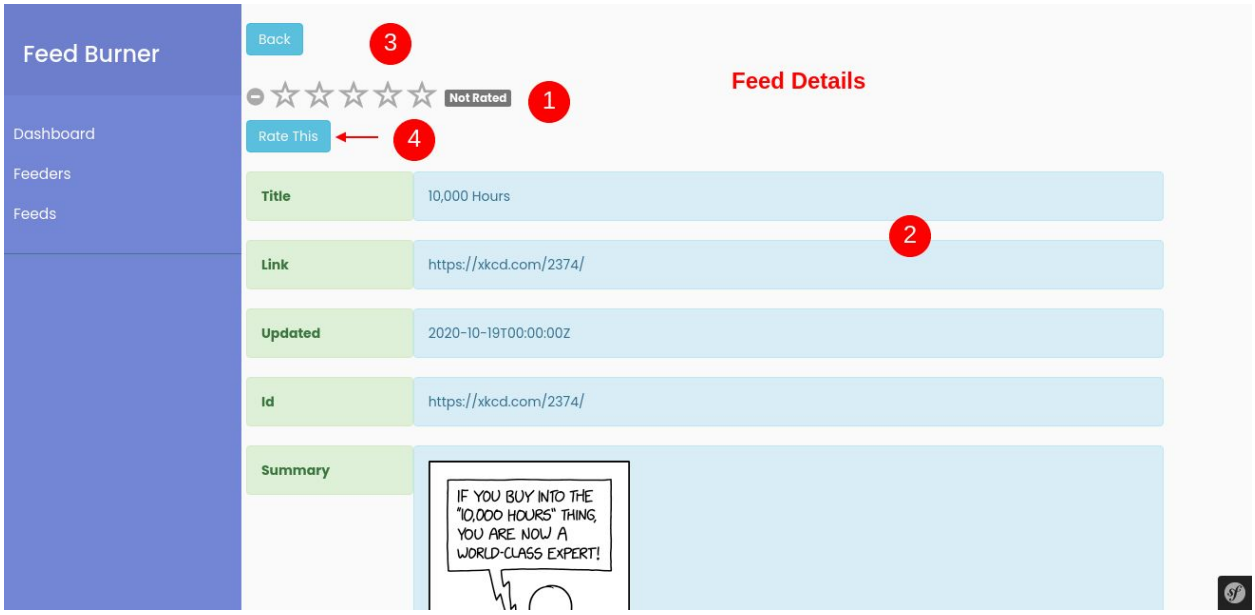
6. **Feed List - View imported feeds data**

Users can check the feeds imported with the Feeder link with the View button available on the feeder list page.



1) DataTable shows the list of all the imported feeds on the database, with the title and description.
2) Feed description will have the description from the feed.

7. View Each Feed in detail



8. Add a new rating for the feed.