TECHNISCHE UNIVERSITÄT
CHEMNITZ

Final Term Paper

Datenbanken und Webtechniken
Summer Semester 2019

Muhammad Ramzan Mahmood
513913
Automotive Software Engineering

Introduction

The project was a development project targeting web development. The project was solely designed as an online web application so that it can display feeds of multiple News websites in one place. It is time consuming process for the user to read different news from multiple websites. He/she has to open different websites and read the news.

The objective of this task is to make the reading process less time consuming for the user, which I have done it.

The purpose of project is to facilitate the users with simplest way of accessing different feeds on one place on internet using mobile phone or computer. Which can save the user time and gives the ability to find everything on one platform.

Its user friendly and user can view multiple news at one page. User can view different news one by one or all together. It's also gives ability to user to search his/her favorite news using search bar which are given at the top. User can also have the ability to refresh the feeds. User can also view the different feeds according to interest, which is very useful approach for finding something on internet according to his/her interest.

Technology Used

HTML

HTML is the standard markup language for making Web pages.HTML code guarantees the correct designing of content and pictures so that your Internet program may show them as they are proposed to look. Without HTML, a program would not know how to show message as components or load pictures or different components. HTML additionally gives an essential structure of the page, whereupon Cascading Style Sheets are overlaid to change its appearance. One could consider HTML the bones (structure) of a site page, and CSS as its skin (appearance).

CSS

For styling we utilized essential CSS. CSS is the language for portraying the introduction of Web pages, including hues, design, and textual styles. It enables one to adjust the introduction to various sorts of gadgets, for example, enormous screens, little screens, or printers. [2]

JavaScript

For scripting we utilized JavaScript and JQuery.

To put it plainly, JavaScript is a programming language that lets web engineer’s structure intuitive locales. A large portion of the dynamic conduct you'll see on a website page is on account of JavaScript, which enlarges a program's default controls and practices. [3]

Json

JSON stands for JavaScript Object Notation.JSON is a lightweight data-interchange format.JSON is "self-describing" and easy to understand.JSON is language independent. Since the JSON format is text only, it can easily be sent to and from a server, and used as a data format by any programming language.

So, if you receive data from a server, in JSON format, you can use it like any other JavaScript object.

jQuery

It is not a language, but it is a well written JavaScript code. As quoted on official jQuery website, "it is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development."[4]

PHP

For backend, I have picked PHP Language. I have as of now involvement in PHP so I went with the PHP.

PHP represents Hypertext Pre-processor (no, the abbreviation doesn't pursue the name). It is an open source, server-side, scripting language utilized for the improvement of web applications. By scripting language, we mean a program that is content based (lines of code) composed for the computerization of errands.

PHP can be embedded in HTML and it is well suited for web development and the creation of dynamic web pages for web applications, e-commerce applications, as well as database applications. It is considered a friendly language with abilities to easily connect with MySQL.

Laravel Framework

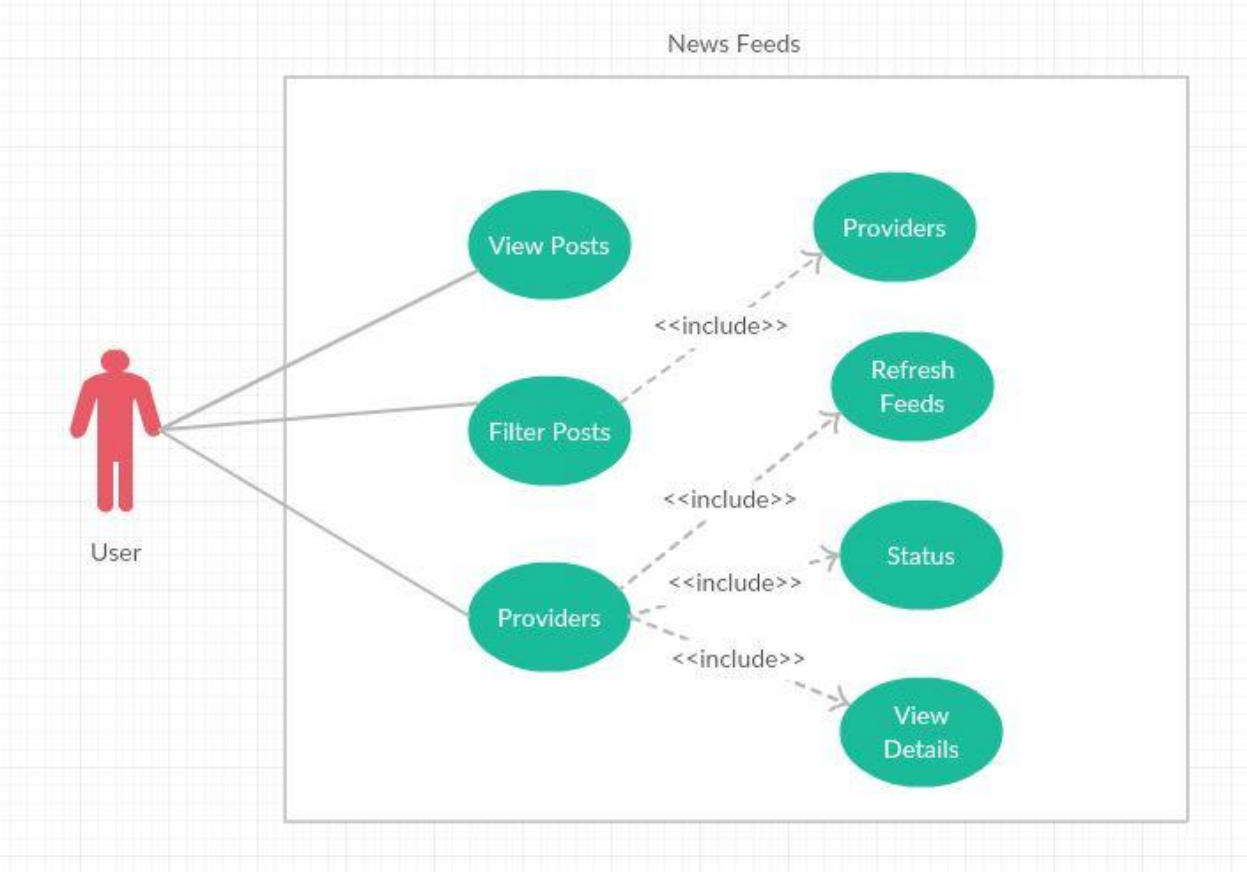
Laravel is a web application framework with expressive, elegant syntax. We believe development must be an enjoyable, creative experience to be truly fulfilling. Laravel attempts to take the pain out of development by easing common tasks used in the majority of web projects, such as authentication, routing, sessions, and caching.

Laravel is accessible, yet powerful, providing powerful tools needed for large, robust applications. A superb inversion of control container, expressive migration system, and tightly integrated unit testing support give you the tools you need to build any application with which you are tasked

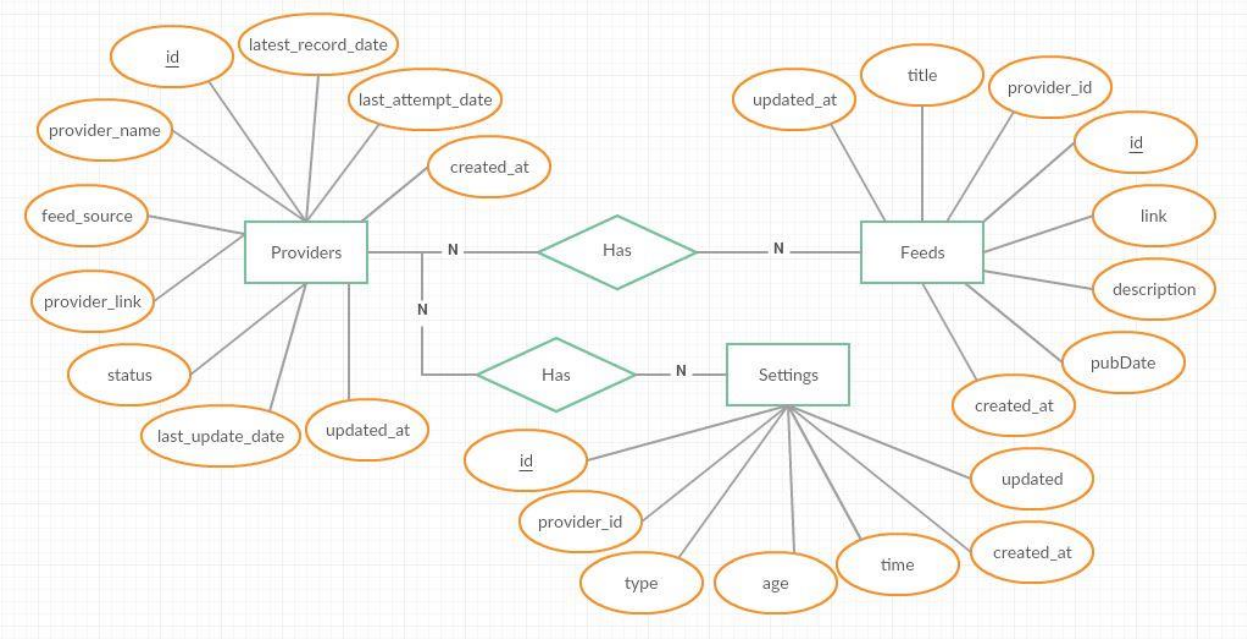
Database Selection

For Database, I have chosen MYSQL for putting away and recovering information. Again I had involvement with MYSQL, so I went with MYSQL as well. MySQL is a ground-breaking, free open-source database the executive's framework that has been around for quite a long time. It is truly steady and has a major network that keeps up, troubleshoots and redesign it.

Use Case Diagram:

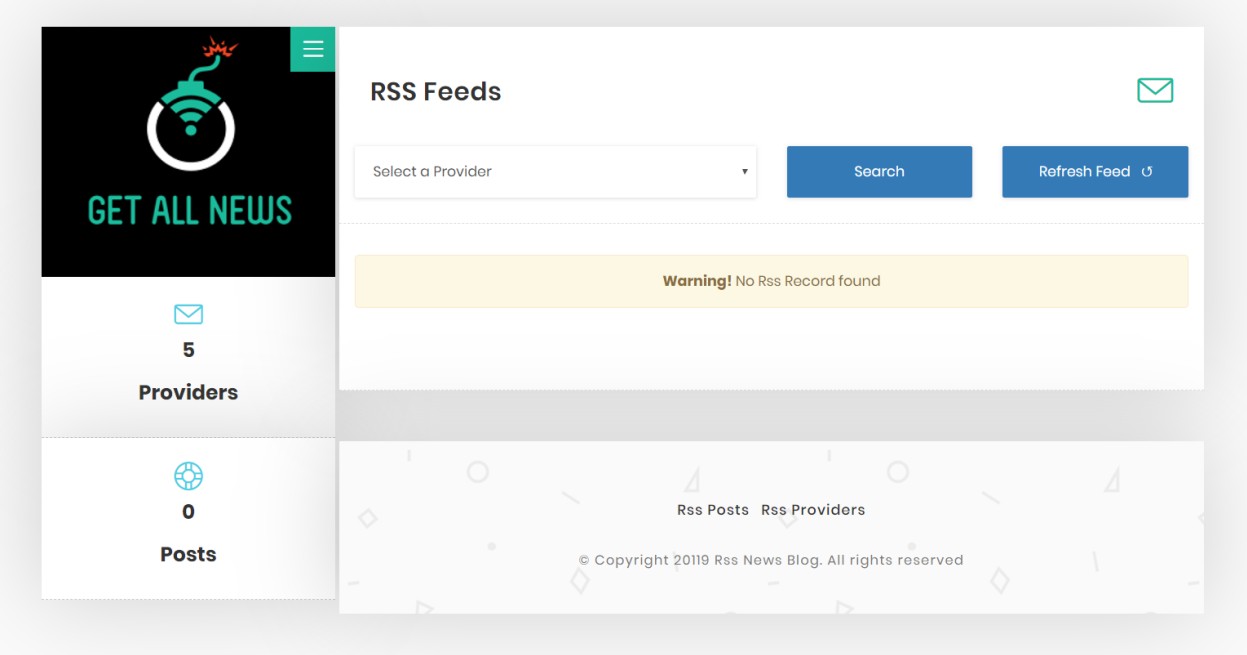


Entity Relation Diagram:

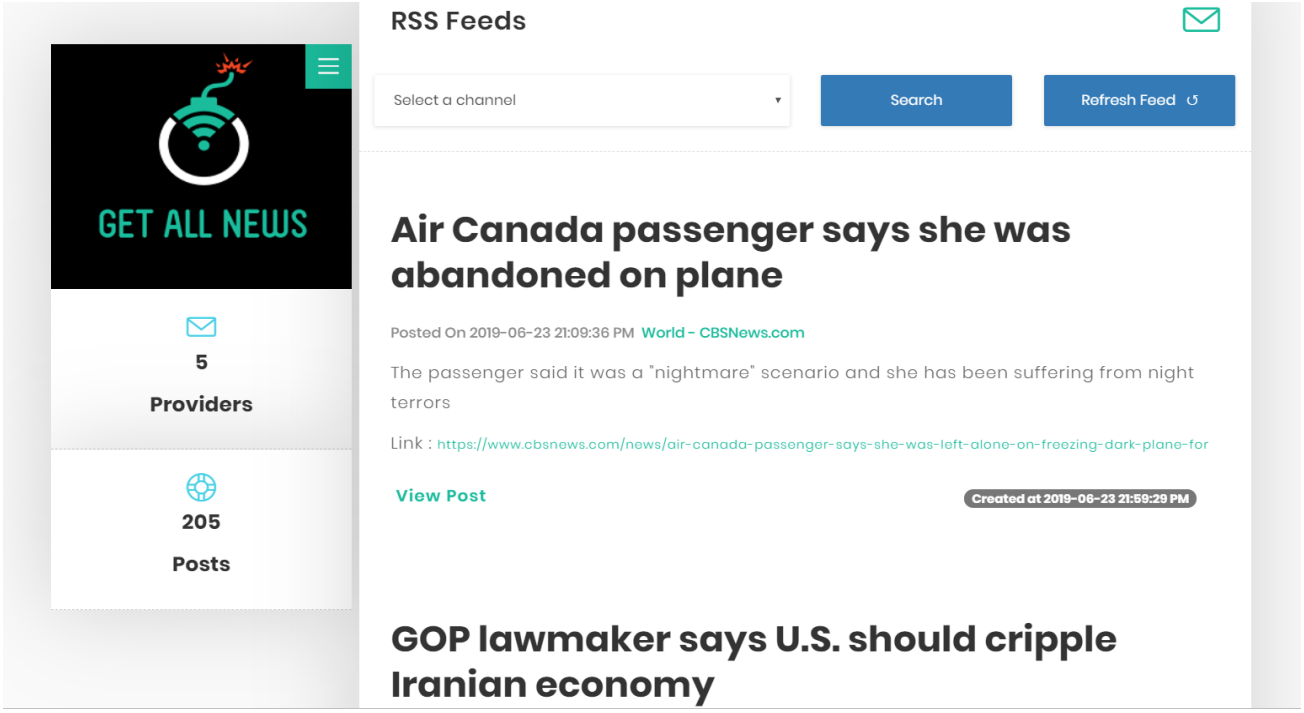


Practical Demonstration of the platform

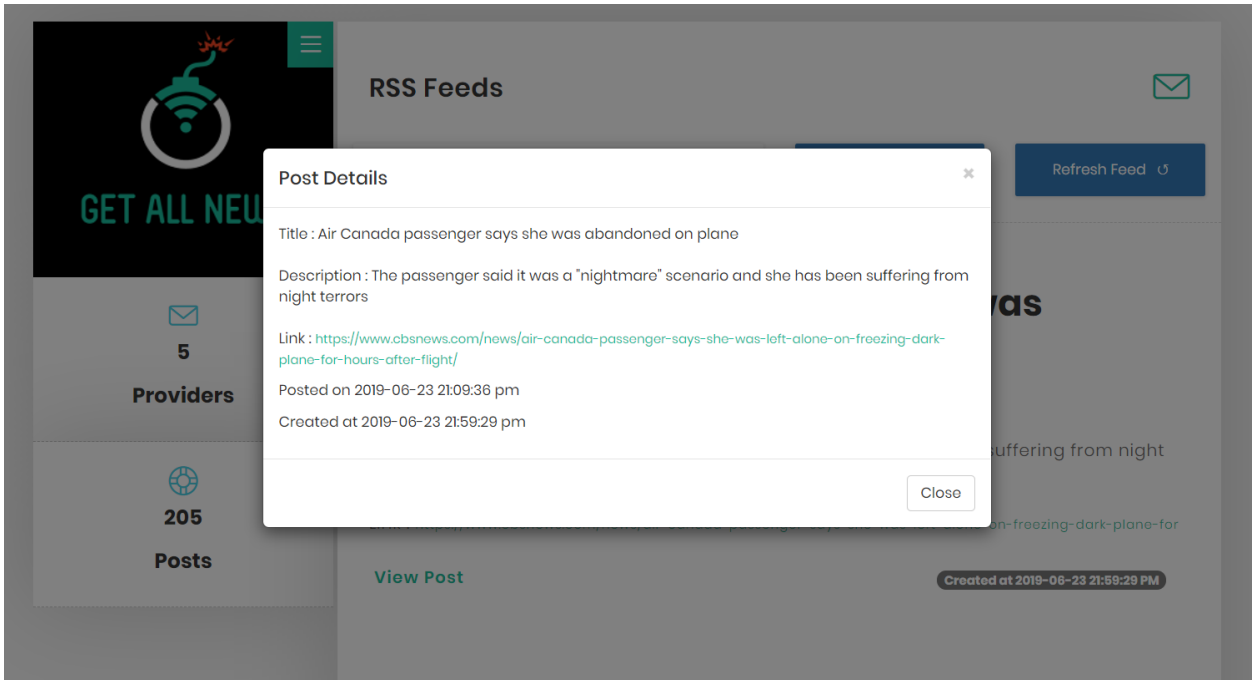
1. When it will be launched 1st time there is not any data of RSS news feeds to display.



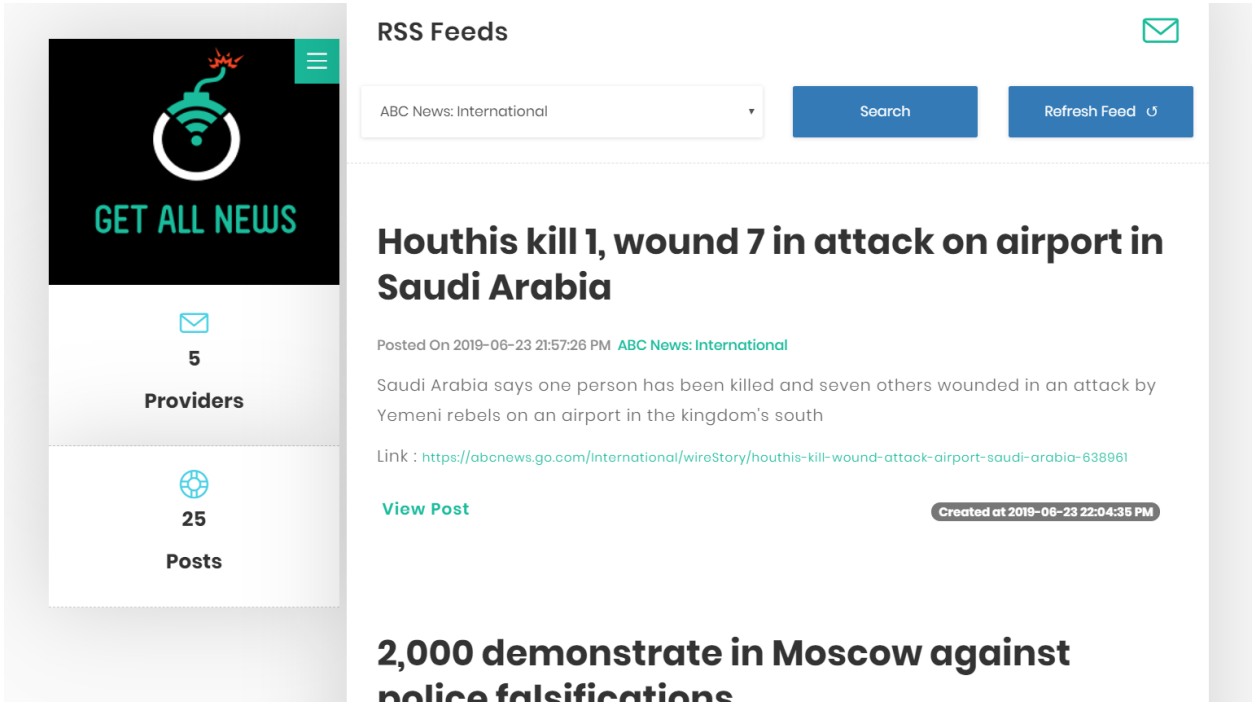
2. After clicking on refresh feed button all feeds will be fetched and will be inserted in database. Then displayed on the website and left counter also updated with number of feeds found. as shown in images below



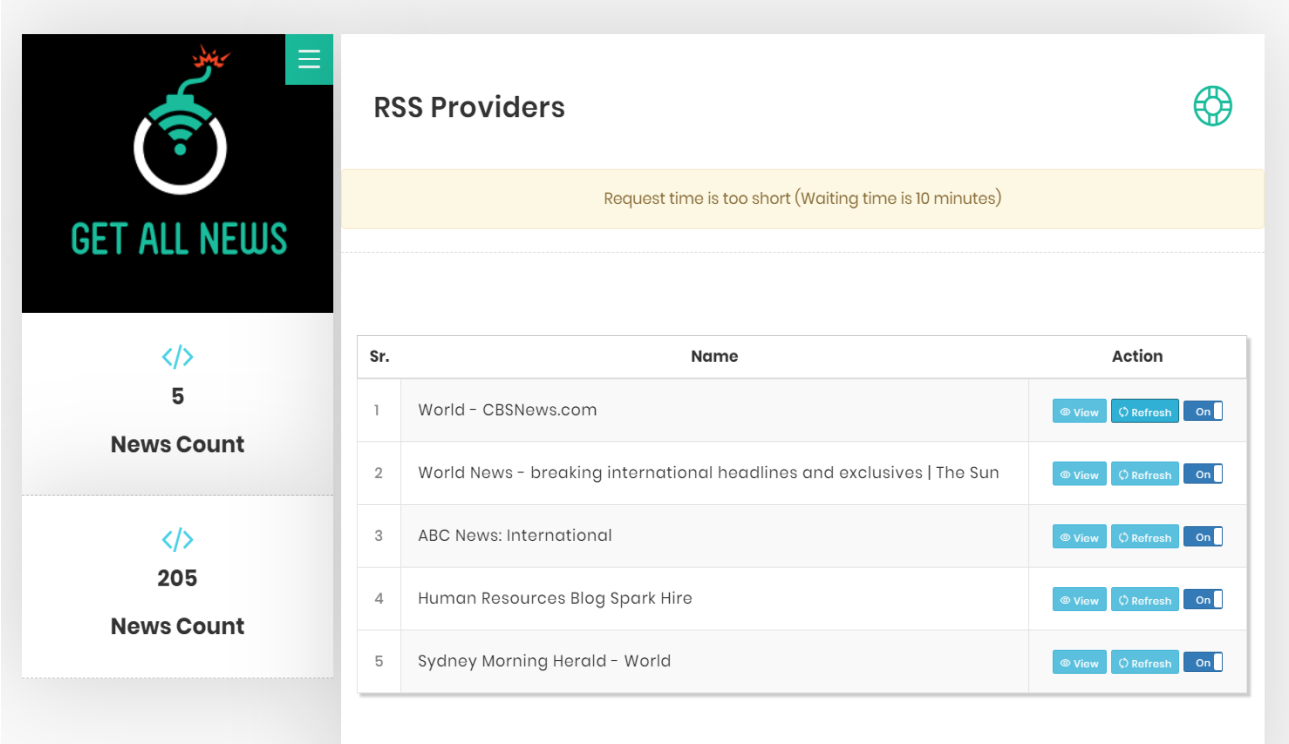
3. It is also possible to open one news feed in modal. As shown below



4. There is also possibility to filter providers to show only there feeds. From select box I am going to select one provider and after clicking refresh it will get only feeds for the specific provider and count will also update as shown in below image.

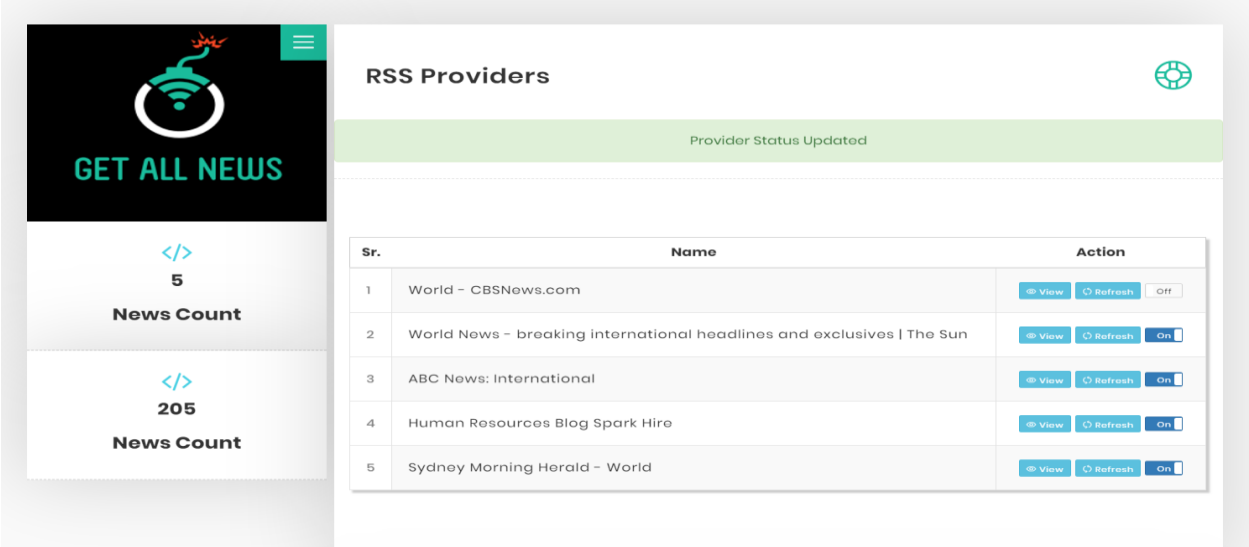


5. It prevents the updating of web feeds in a too small time interval and waits for 10 minutes between two requests for a web feed. Now I will go to provider's page by clicking left Rss Providers from left navigation menu. List off all providers is displayed. I try to refresh feed of a provider and it return's warning as shown in below image.

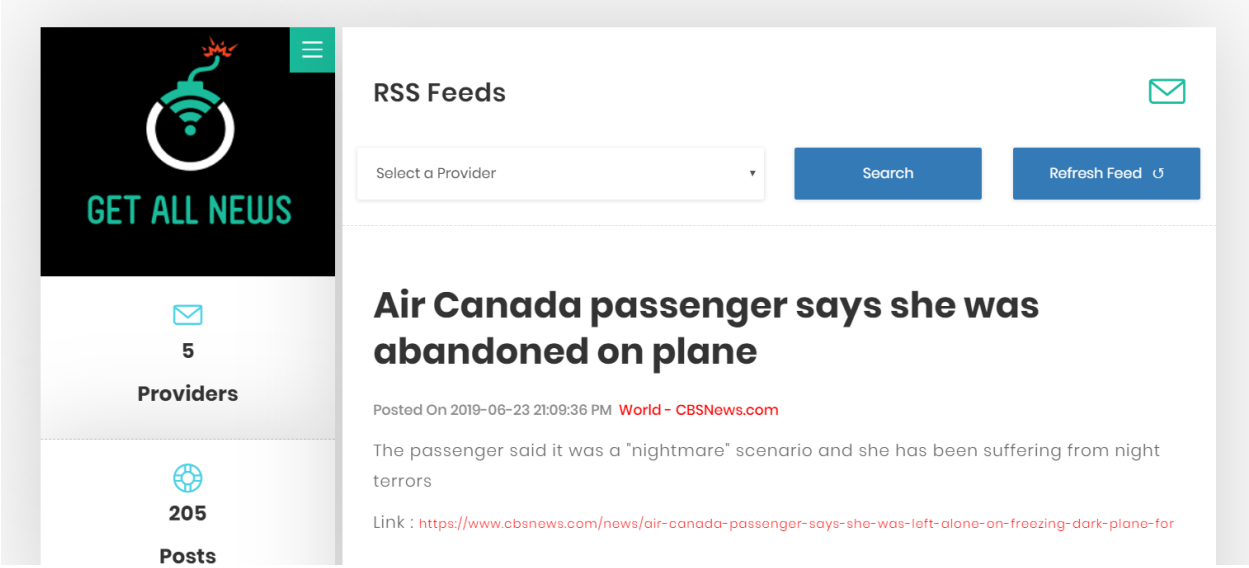


6. Website ensures consistent storage of the data in a database.
7. There is threshold to delete 30 days old rss feeds from database which is handled at the backend.
8. From the providers action tab it is possible to active or deactivate individual providers. Once a provider link is off its all posts will be disables on rss feeds page as shown below.

Step 1 : Disable Link

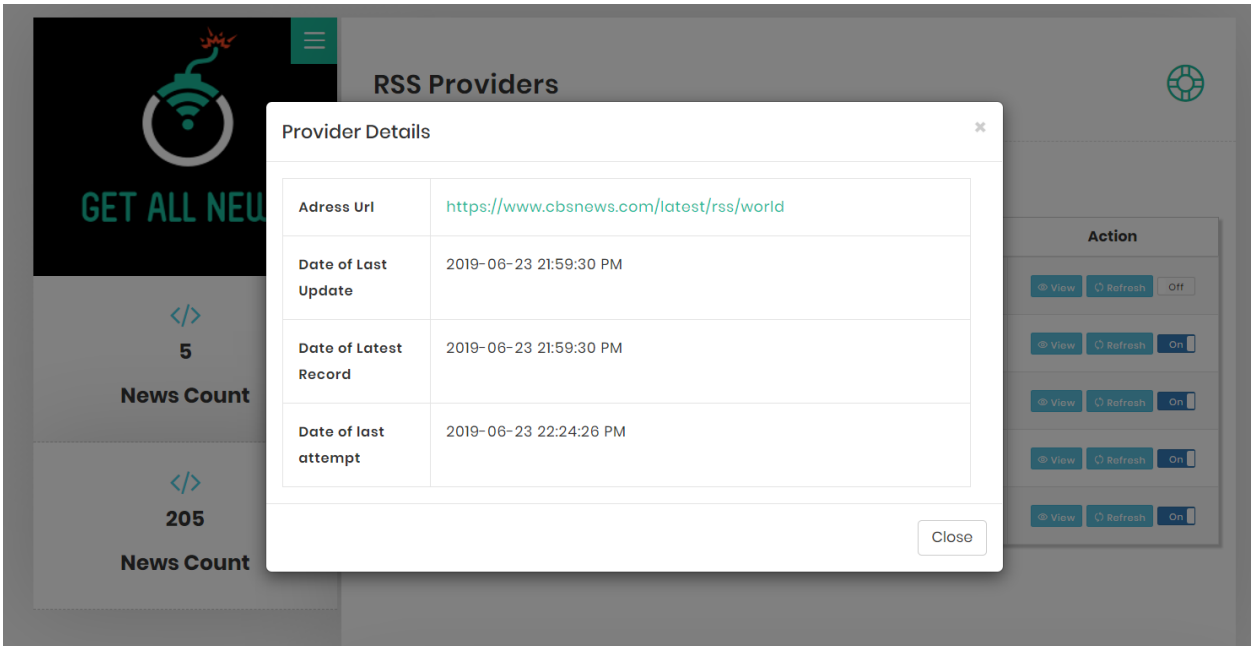


Step: 2 News Feeds of that link



9. There is also another button in the action tab for view provider it shows all details about provider.
    - Address (URL) of the web feed

- Date of the last successful update of the web feed
- Date of the latest record according to information from the web feed.
- Date of the last attempt to update
- Error count if any occurs during the update attempt.



Code Explanation and Implementation.

Controller exists inside app/http directory of the project and there RssApiController.php is responsible for api operations while ProviderController.php and RssController.php are responsible for backend operations of Rss provider and Rss feeds respectively. App/models directory contains models for Database operations. Views exits in resources/view where layout and includes folder have views master files.

RssApiController.php

```php
public function getFeeds() {

    $request_params = Input::all();

    $rss  = DB::table('feeds')->join('providers', 'providers.id', '=' , 'feeds.provider_id')
        ->select('feeds.*', 'providers.status', 'providers.provider_name', 'providers.provider_link');

    if (isset($request_params['provider_id'])) {

        $rss->where('provider_id', $request_params['provider_id']);
    } elseif(isset($request_params['post_id'])){

        $rss->where('feeds.id', $request_params['post_id']);
    }else {
        $rss;
    }

    $posts = $rss->get();

    return $this->jsonSuccessResponse('Process is processed success', $posts);
}

/**
 * channels method
 * responsible for getting channels
 * @return \Illuminate\Http\Response
 */
public function getProviders() {

    $request_params = Input::all();
    if (isset($request_params['provider_id'])) {
        $providers = \App\Models\Provider::where('id', $request_params['provider_id'])->get();
    } else {
        $providers = \App\Models\Provider::all();
    }
    return $this->jsonSuccessResponse('Process is processed success', $providers);
}
```

It render response from Traits/ApiService.php and it look likes

```php
     * @return Response
     */
    public function jsonErrorResponse($error, $code = 204) {
        $response = [];
        $response['success'] = false;
        $response['message'] = $error;
        $response['status_code'] = $code;
        return Response::json($response);
    }

    /**
     * jsonSuccessResponse method
     * @param type $msg
     * @param type $data
     * @param type $code
     * @return type
     */
    public function jsonSuccessResponse($msg, $data = [], $code = 200) {
        $response = [];
        $response['success'] = true;
        $response['data'] = $data;
        $response['message'] = $msg;
        $response['status_code'] = $code;
        return Response::json($response);
    }

    /**
     * jsonSuccessResponseWithoutData method
     * @param type $msg
     * @return type
     */
    public function jsonSuccessResponseWithoutData($msg) {
        $response = [];
        $response['success'] = true;
        $response['message'] = $msg;
        return Response::json($response);
    }
```

Appendix:

- I have used Laravel Api's all information can be found here https://laravel.com/docs/5.8/eloquent-resources
- Main api of providers and rssfeeds table are shown below.

You just need to copy the data from api.yml

- Open http://editor.swagger.io/
- Place the data from api.php

It would show you all the apis as shown below.

providers

GET	/rssproviders	List
POST	/rssproviders	Create
GET	/rssproviders/{id}	Read
PUT	/rssproviders/{id}	Update
DELETE	/rssproviders/{id}	Delete
PATCH	/rssproviders/{id}	Increment