# Lab: Class-Based-Views Deep Dive

# Basic News App using CBV's

*We will create a* ***simple news app****. We will have* ***2 forms****: one for adding a new* ***Source*** *(a simple model with a* ***name*** *and an* ***url****) and one for adding an* ***Article*** *(it should have* ***title****,* ***image****,* ***description*** *and an existing* ***source instance****). All the articles will be displayed on the* ***Home page****. We will practice* ***handling forms*** *and using* ***mixins*** *with* ***CBV's****.*

## Setup a new project

Create a **new project** in a separate folder. Create a **single app** in the project folder. Make the **needed configurations**

## Models

### Source

* **name** – character field with max length of 50
* **url** – url field with max length of 200

### Article

* **title** – character field with max length of 100
* **description** – text field
* **imageUrl** – url field with max length of 200
* **source** – foreign key

## Template separation

Take the provided **resources** (**html** and **css**) and separate the html into proper **templates**

## Forms

Create **form classes** for your models. Add the needed **widgets** to apply the **styles** provided.

## Pages

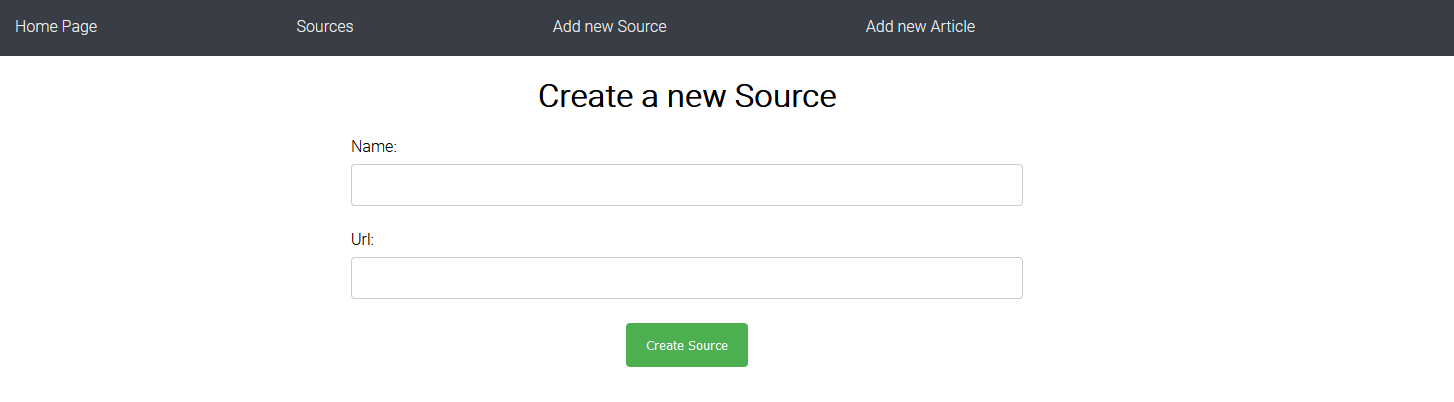
Use the needed types of **class-based-views** for each of the pages

### Home page



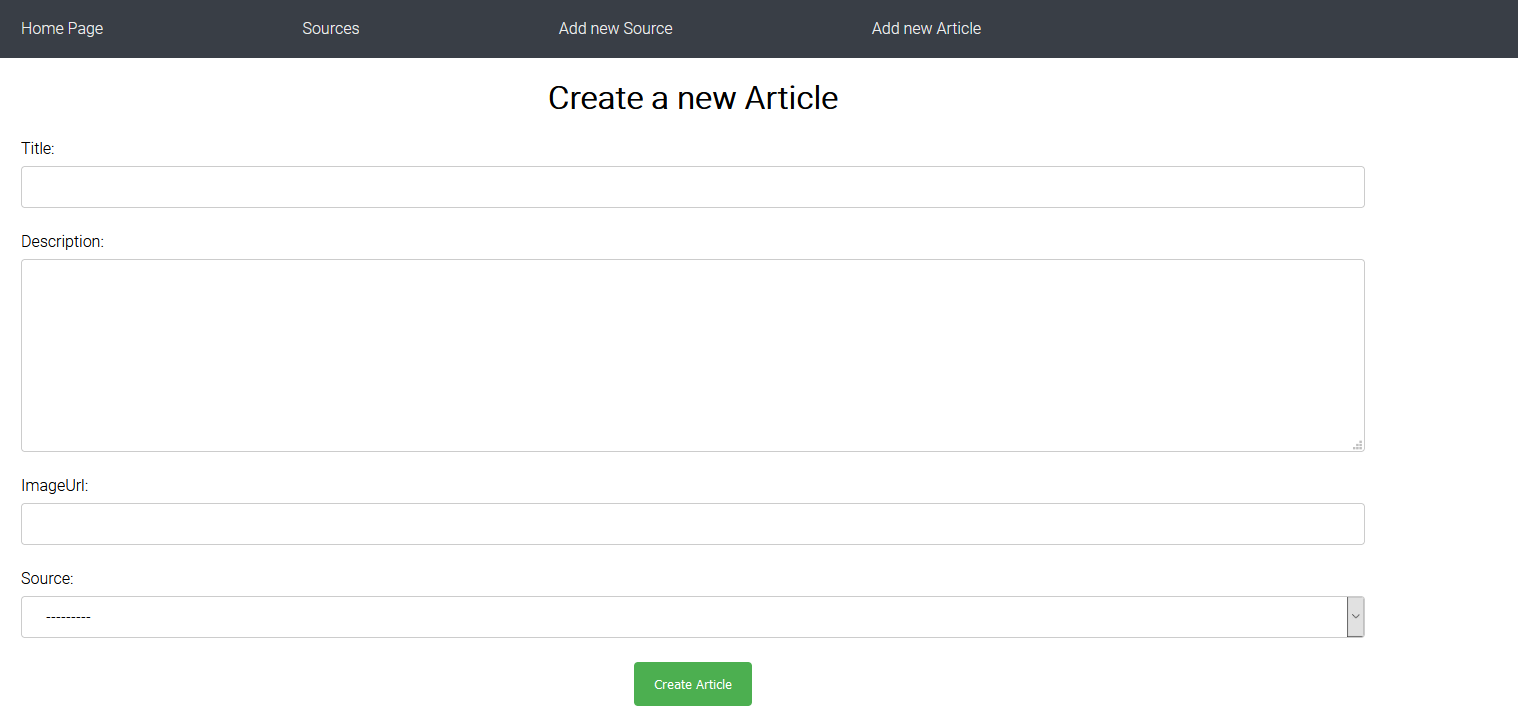
The home page should display **all articles** as shown in the picture above

### Source Create page



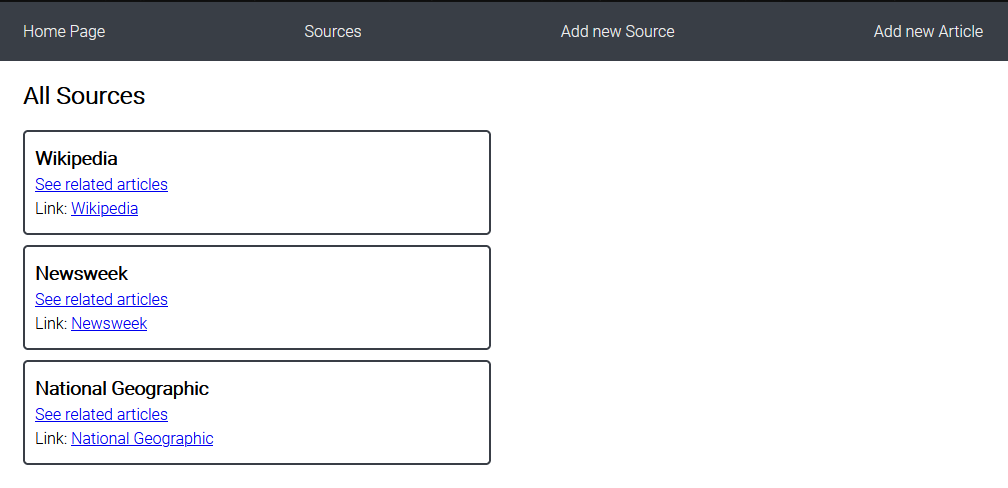
The source create page should display a **form** for **creating a source**.

### Article Create page



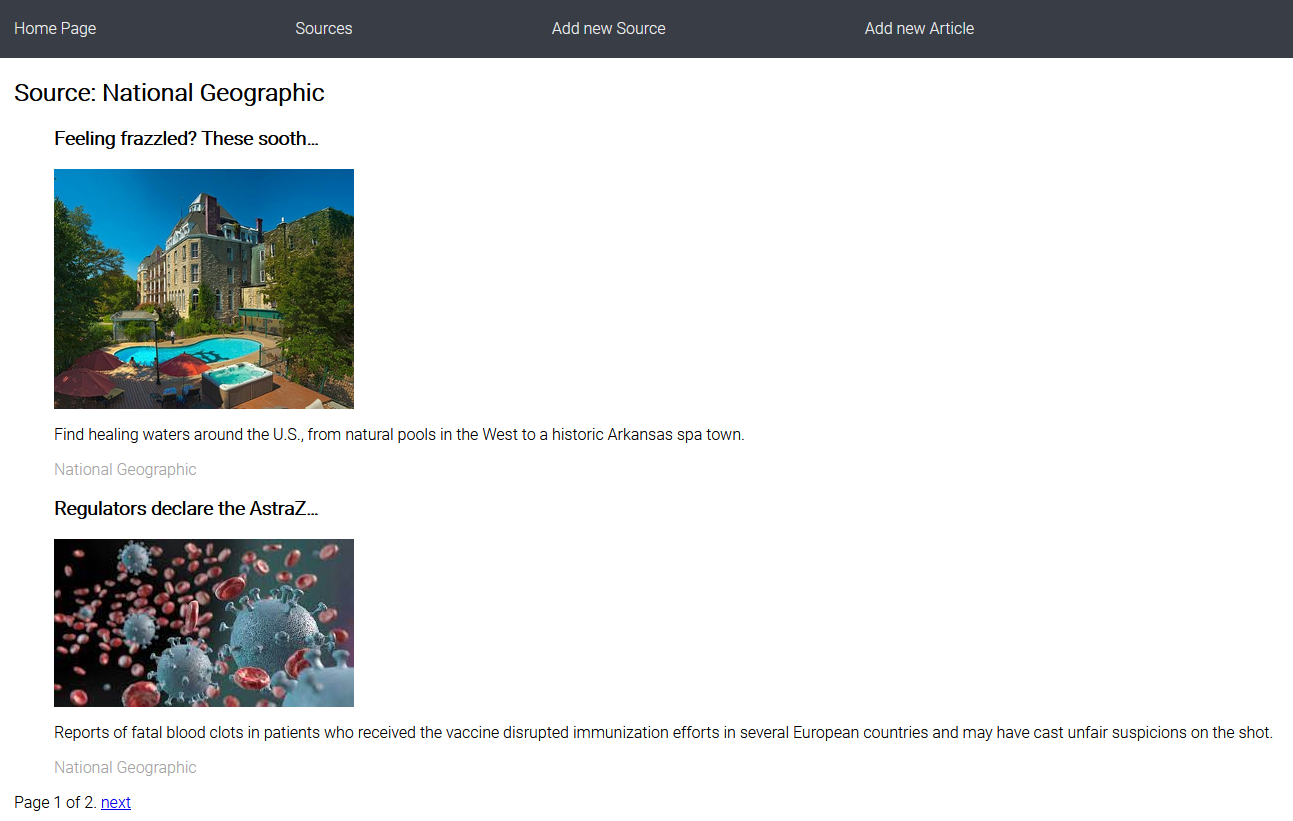
The article create page should display a **form** for **creating an article**.

### Sources page



The sources page should display **all sources** as shown in the picture above. Upon **clicking** on **"See related articles"**, a **source details page** should be displayed. When clicking the **link** of the source, open its **url** in a **new tab**

### Source Details page



The source details page should display information as shown in the picture above.

***HINT: Use SingleObjectMixin with ListView – might want to read*** [this](https://docs.djangoproject.com/en/3.1/topics/class-based-views/mixins/#using-singleobjectmixin-with-listview)