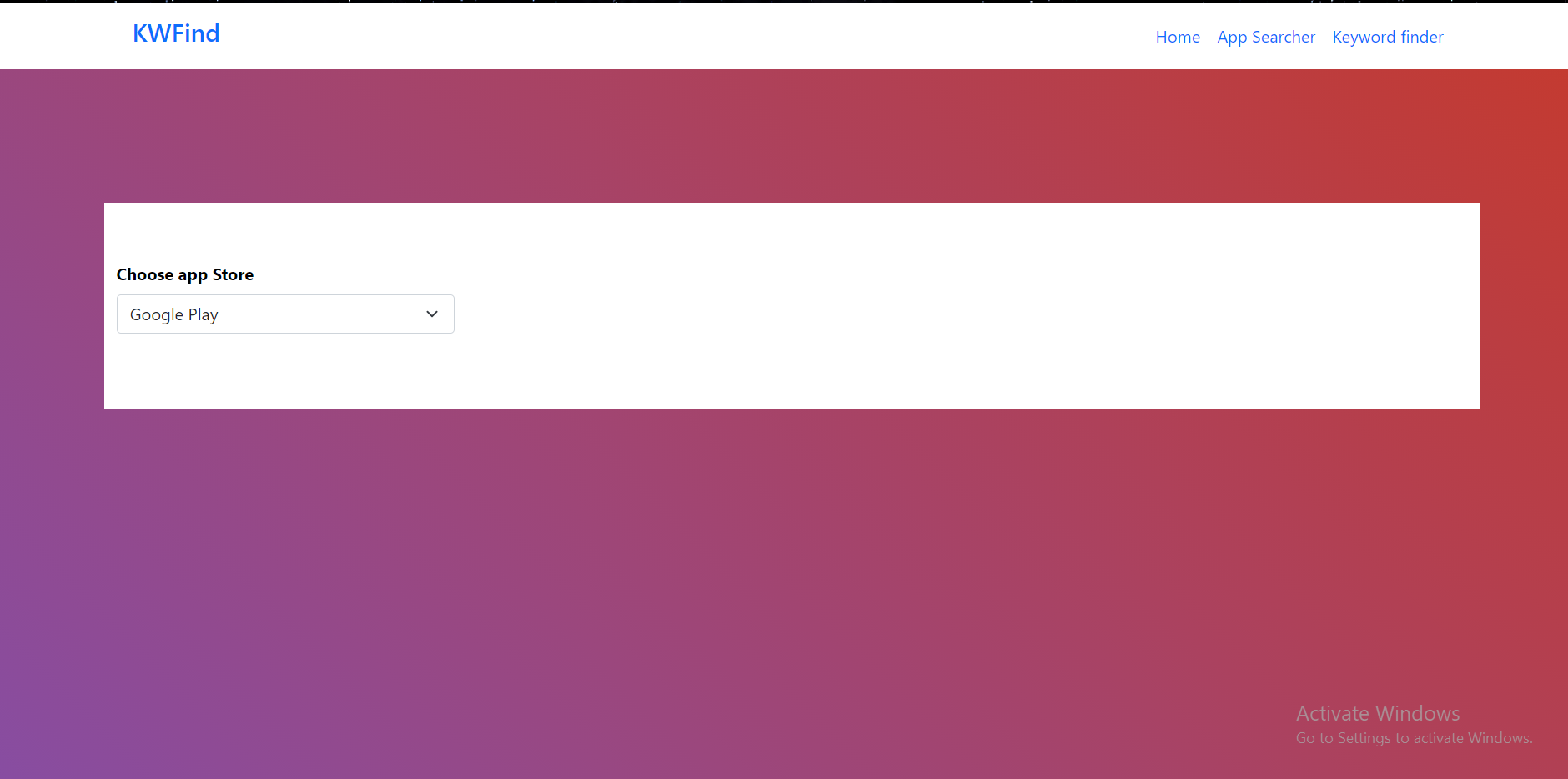
#### **Introduction:**

It is a related keyword finder and app searcher built using django framework in django.

#### **1.App searcher**

**Brief:** This page will be used to get information about apps on the Apple App store or the Google Play store.

This page is used to get the information aboutthe apps in App-store. After selecting google play or Apple store corresponding form will be visible.



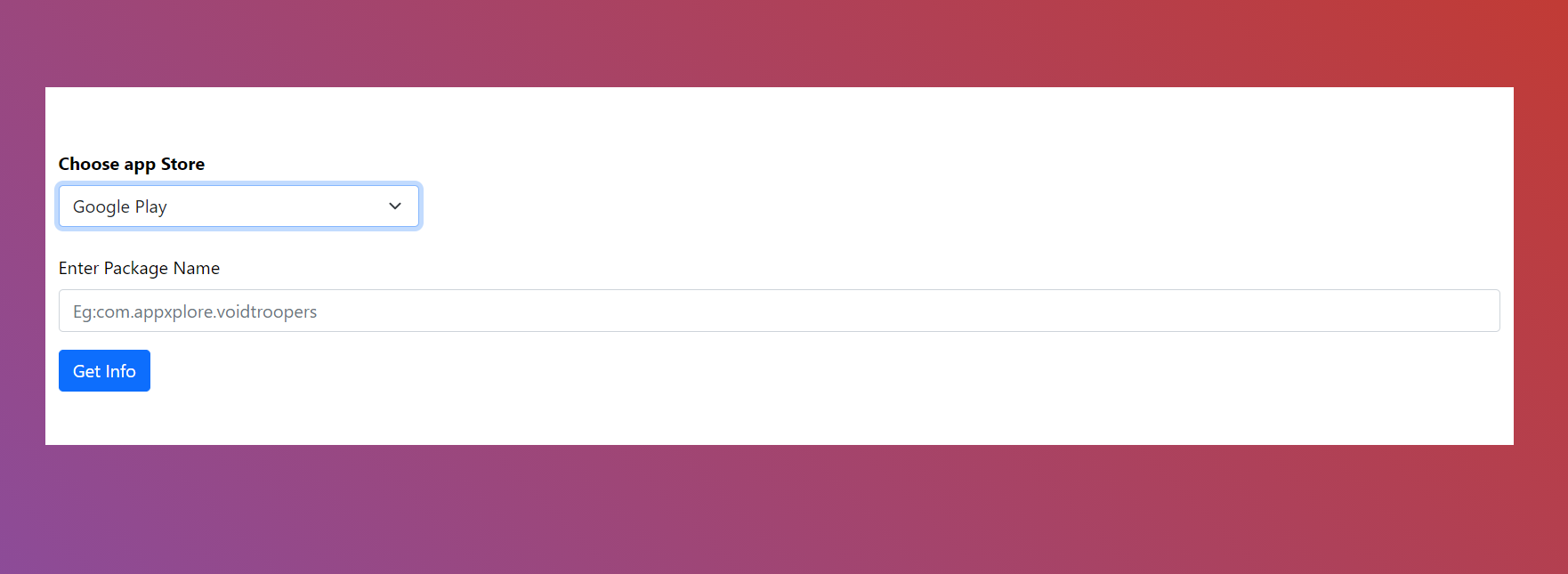
**For Google Play:**

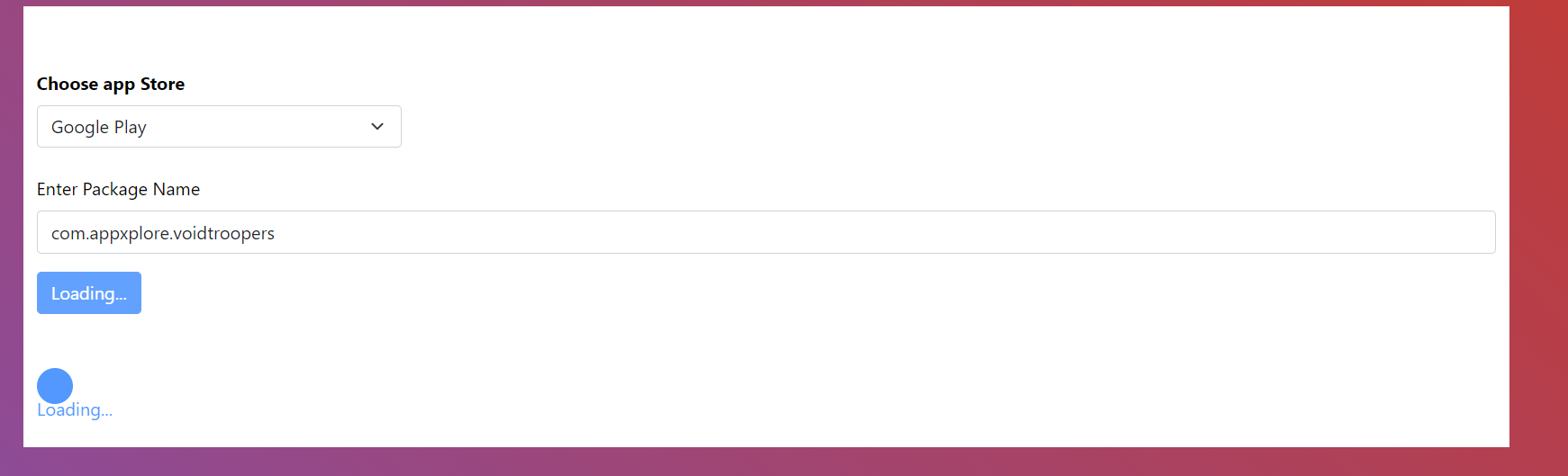
There is only one field i.e Package name and is mandatory.

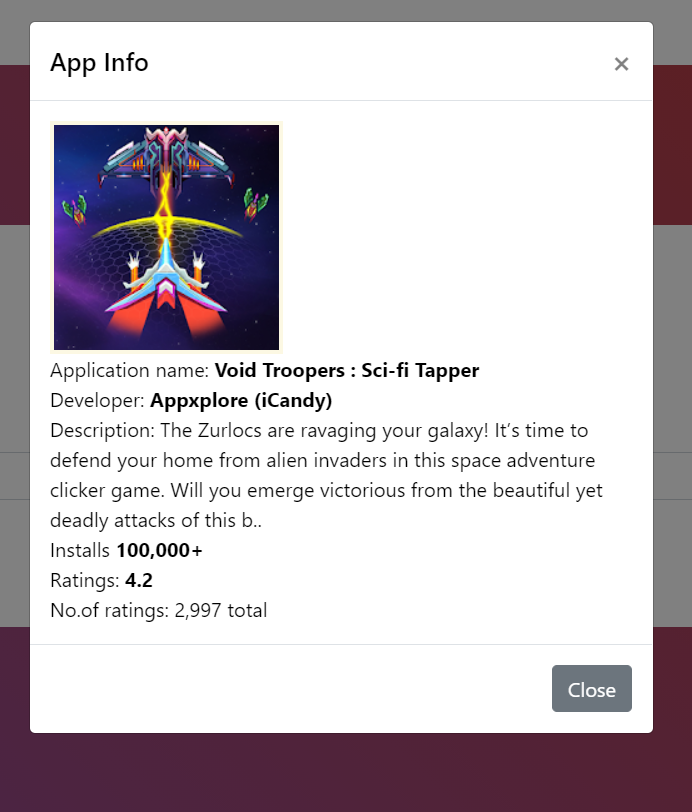
The UI is simple and responsive. The get info button will trigger an Ajax request and fetch the corresponding response.

The loader will appear once the form is submitted and button will be disabled.

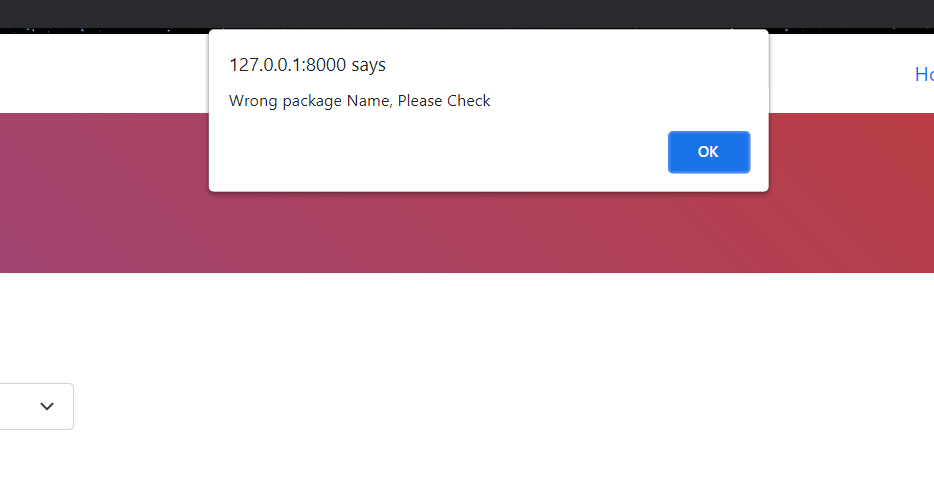
App Icon , App Name ,Developer Name , Description, No. of Downloads ,App Rating No. of ratings/reviews







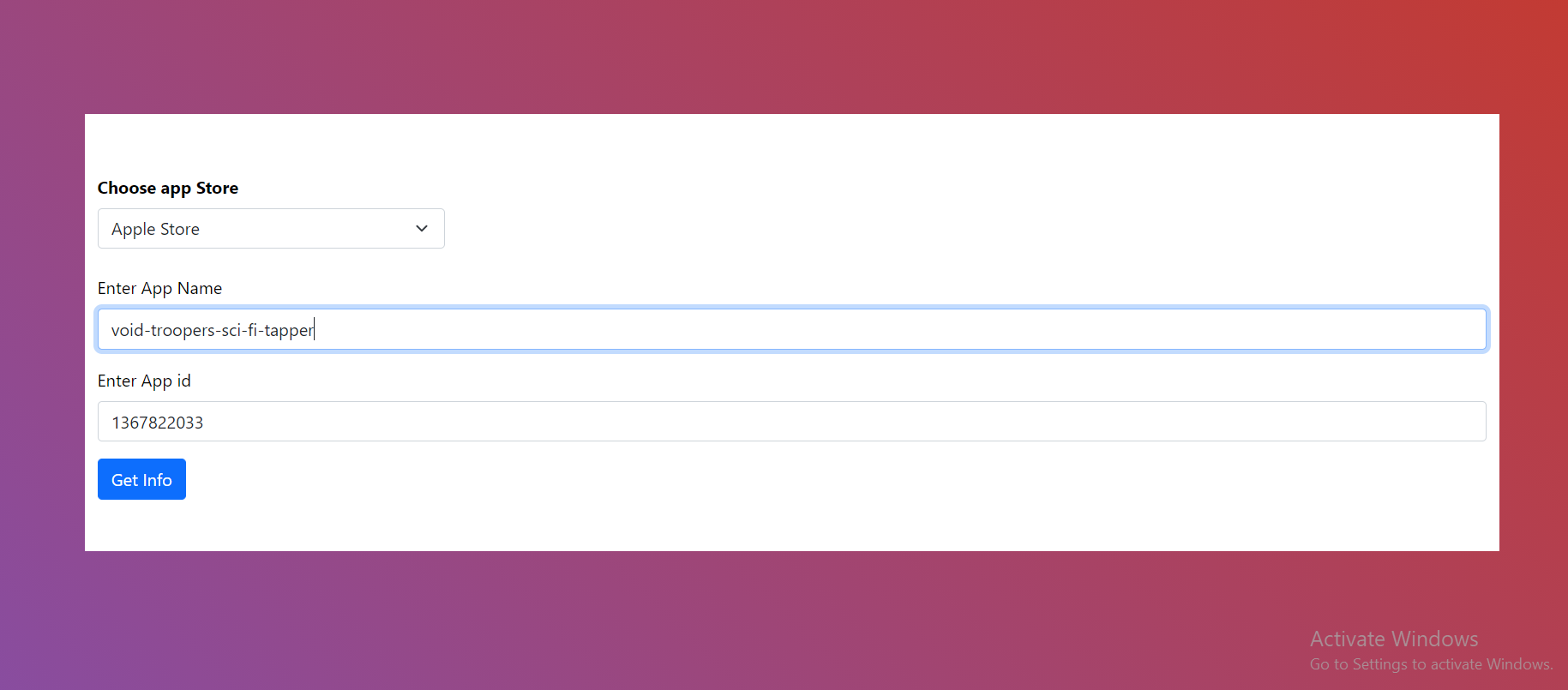
If the package name is incorrect it will create alert for the error:

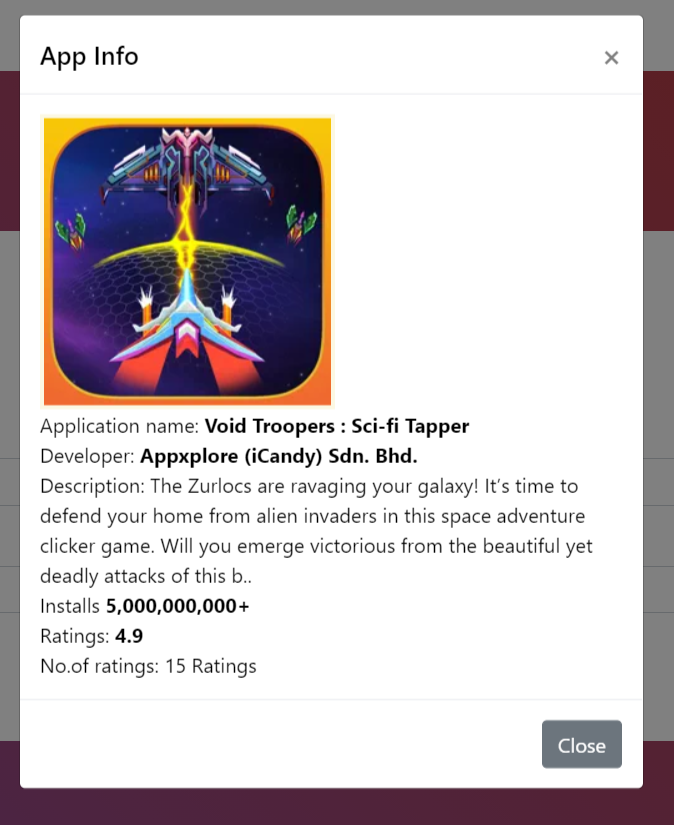


**For Apple Store:**

There are two fields app name and app id and both are mandatory.

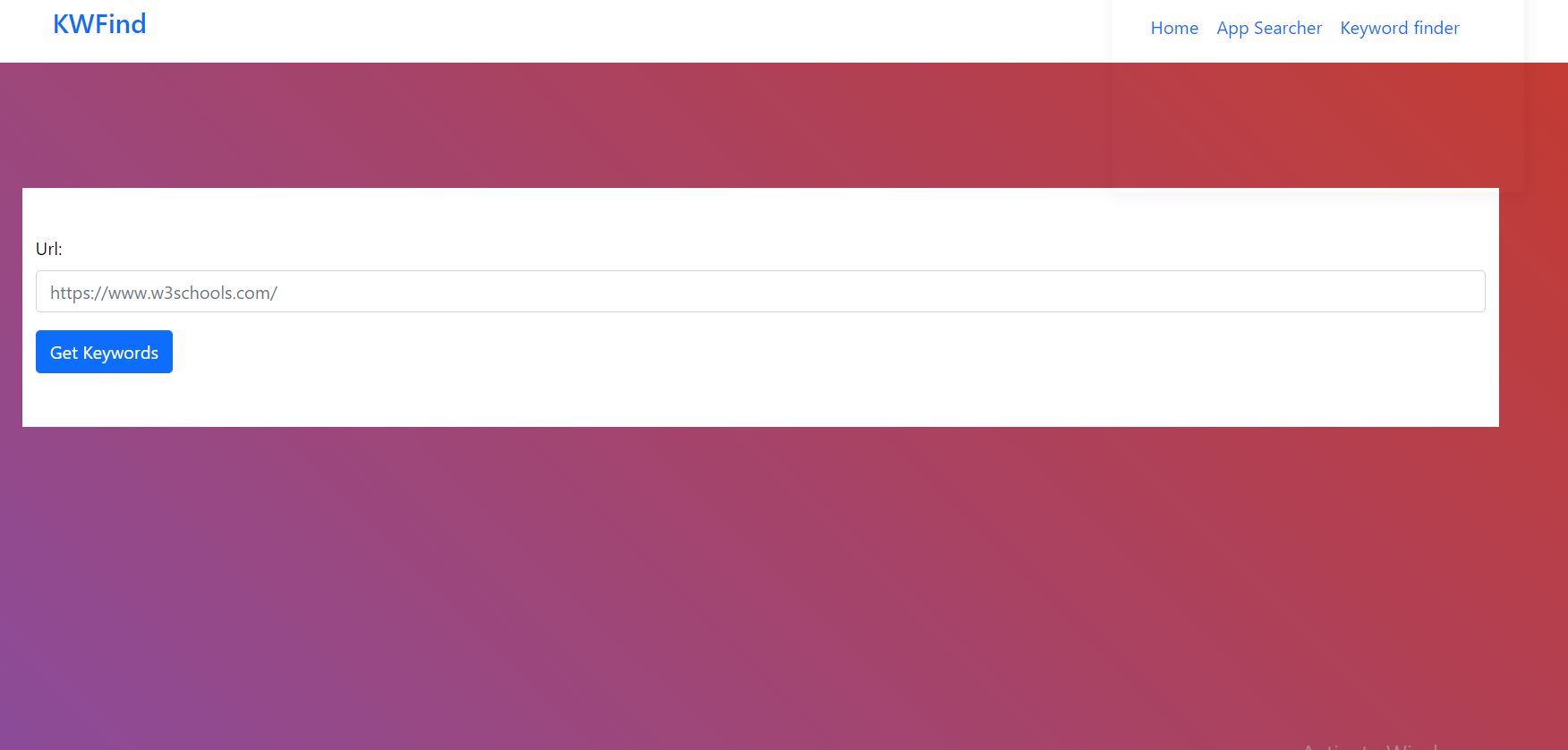
It is similar as google play.



.

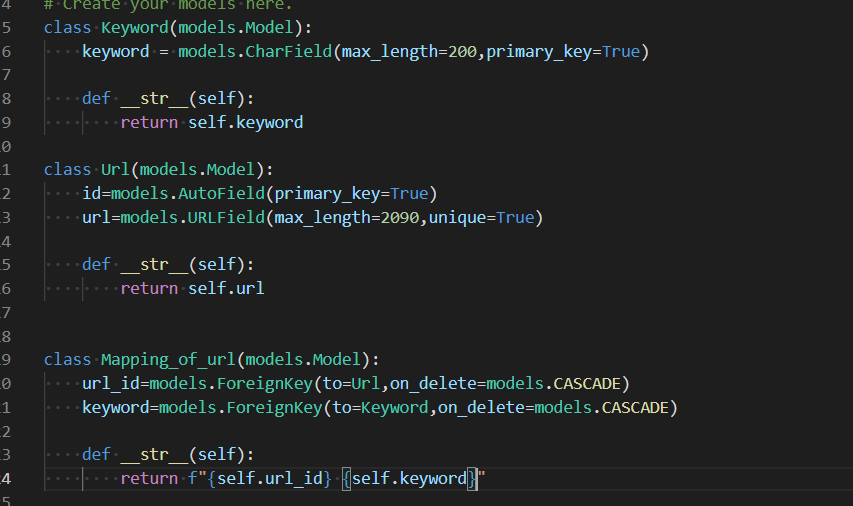
I have used beautifulsoup as a website crawling libraries as it is popular and most efficient.

**2. Keyword Finder**



It has one input box with get keywords button. It is using django forms to get the necessary information.

For storing keywords and Urls , I have used Mysqlite (to facilitate working of project on different machines without the need to connect to external database separately)



There are 3 tables in Models-

1.Keywords table- It has only one field keyword which is primary key. I have chosen this as primary key to improve searching of keywords and prevent duplication of keywords.

2.Url table- It has two fields id which is autofield and url which is urlfield and max\_length=2090 because the max\_length of url till now is 2084 characters.

3.Mapping table- The both fields in this table are Foreign keys and refer to url and keyword. It will store mapping of keyword with url and is triggered if keyword and url pair is not yet present in the table.

3 tables with this structure are the fastest way to retrieve url and keywords from db.

**Keyword finder logic:**

I am firstly extracting all the keywords from the given url and checking if the keyword already exists in table. If it does not exists then it is added in the keyword table.

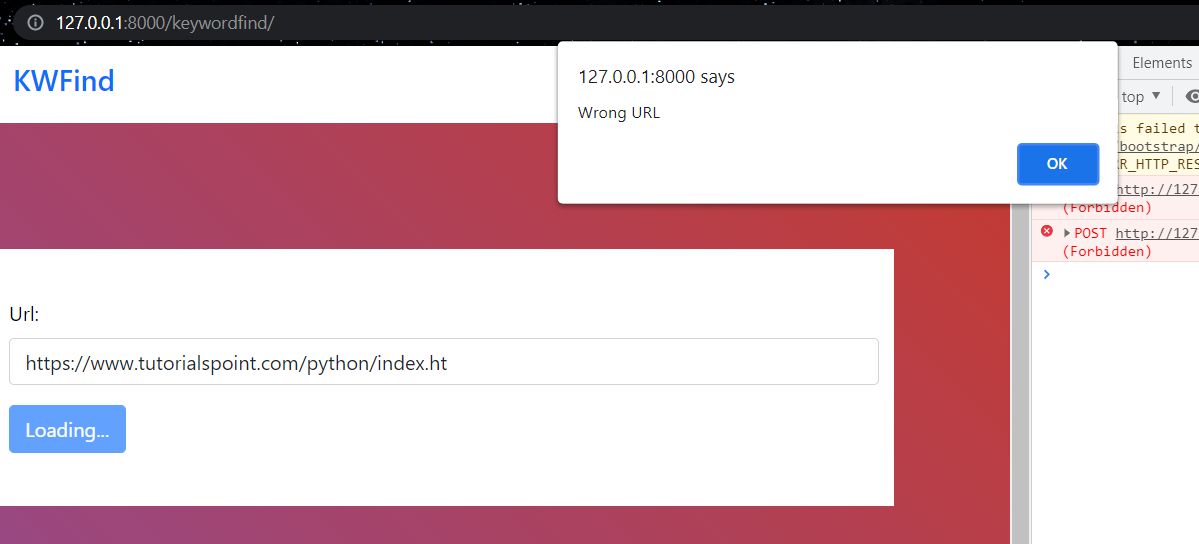
Then the url is checked if it is already present or not and added in the table if it does not exists and mapping is stored according to keyword and url.

For recommended url firstly I extract all url where keywords are same as the keywords found from our input url, and if more than 3 keywords map to the same url than that keyword is recommended.

For recommended keywords I have used recommended urls and did reverse of the above process i.e found all the keywords of recommended url and the ones which are not present in the input url’s keyboard is added to recommended keywords.

**Working:**

Url validation is done and if url is not found error is thrown to user.



Eg: I have two url’s :

[*https://www.tutorialspoint.com/python/index.htm*](https://www.tutorialspoint.com/python/index.htm)

<https://www.w3schools.com/>

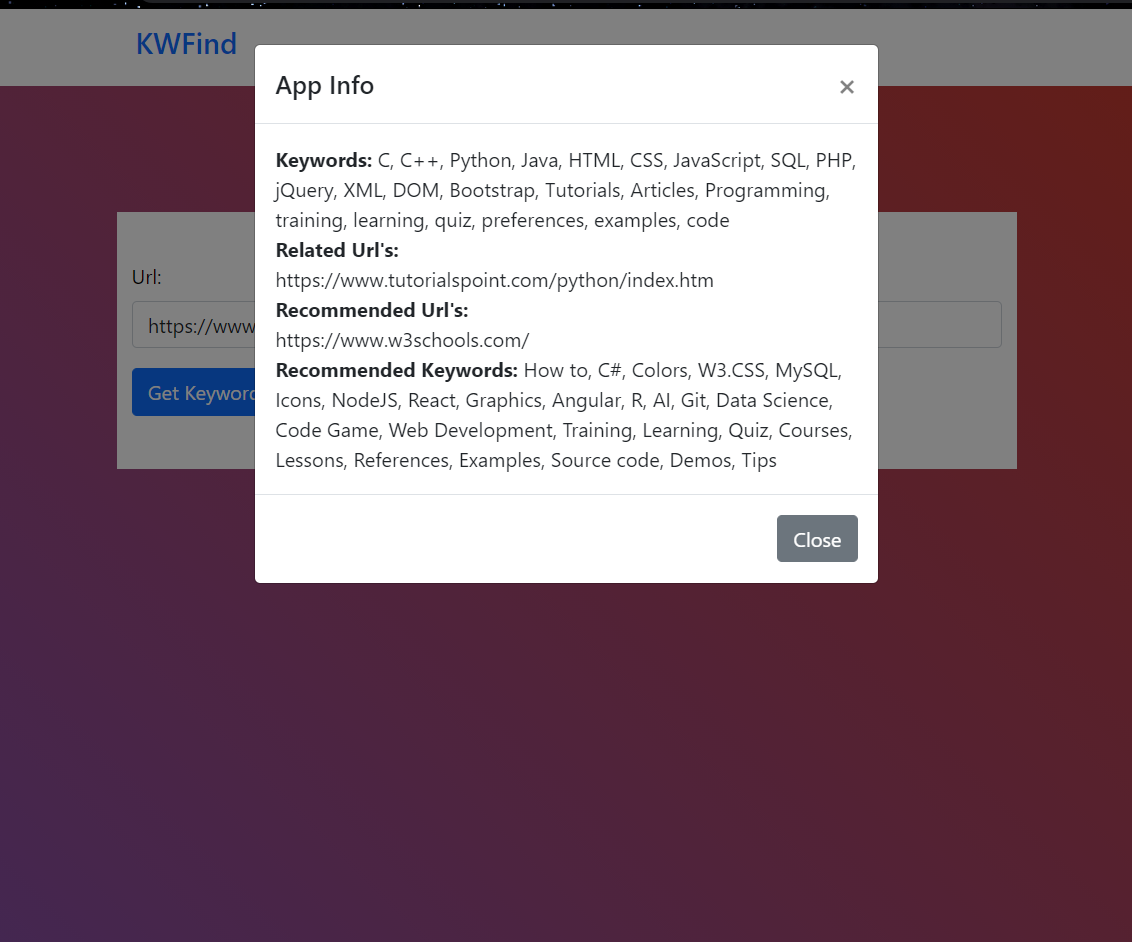
The keywords in tutorials point are: C, C++, Python, Java, HTML, CSS, JavaScript, SQL, PHP, jQuery, XML, DOM, Bootstrap, Tutorials, Articles, Programming, training, learning, quiz, preferences, examples, code.

The keywords in w3 schools are:

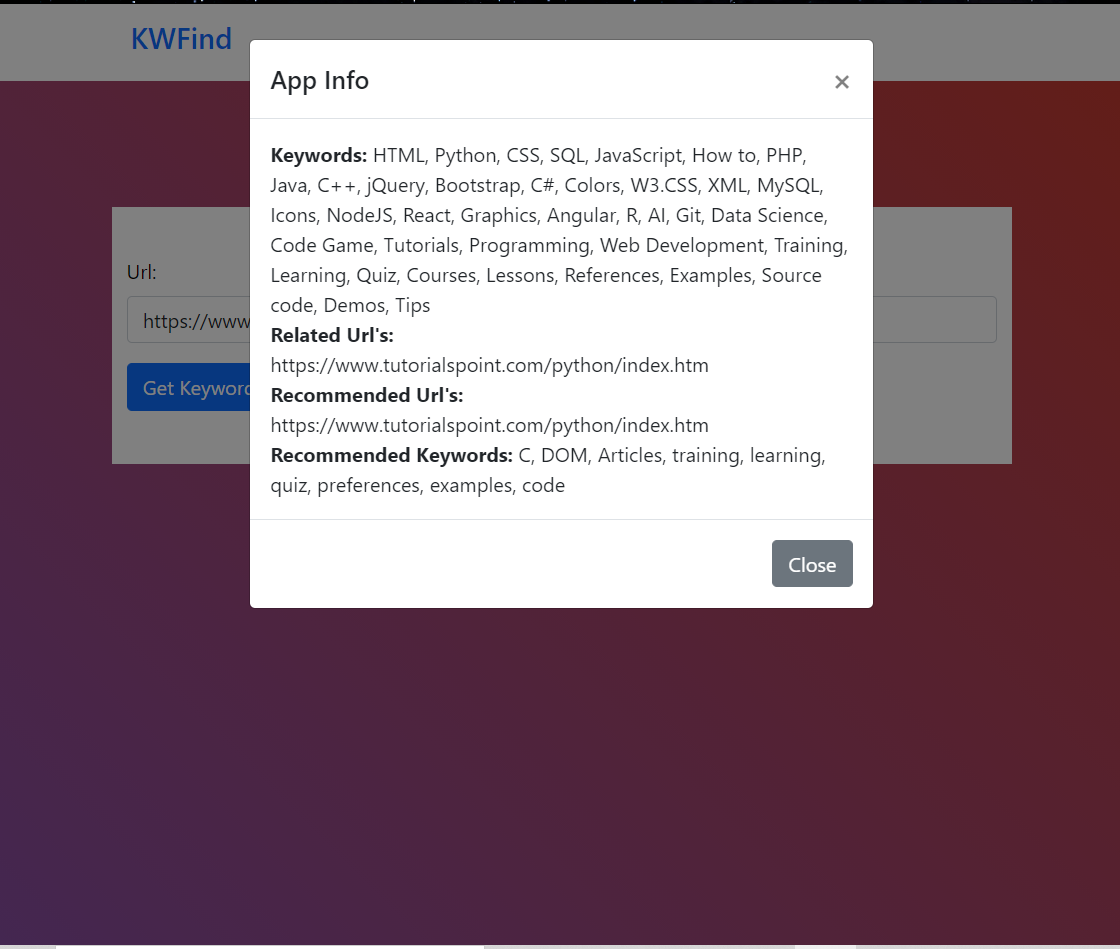
HTML, Python, CSS, SQL, JavaScript, How to, PHP, Java, C++, jQuery, Bootstrap, C#, Colors, W3.CSS, XML, MySQL, Icons, NodeJS, React, Graphics, Angular, R, AI, Git, Data Science, Code Game, Tutorials, Programming, Web Development, Training, Learning, Quiz, Courses, Lessons, References, Examples, Source code, Demos, Tips.

So both urls have more than 3 common keywords and are highly related to each other so the recommended url and keywords are shown for both examples respectively.

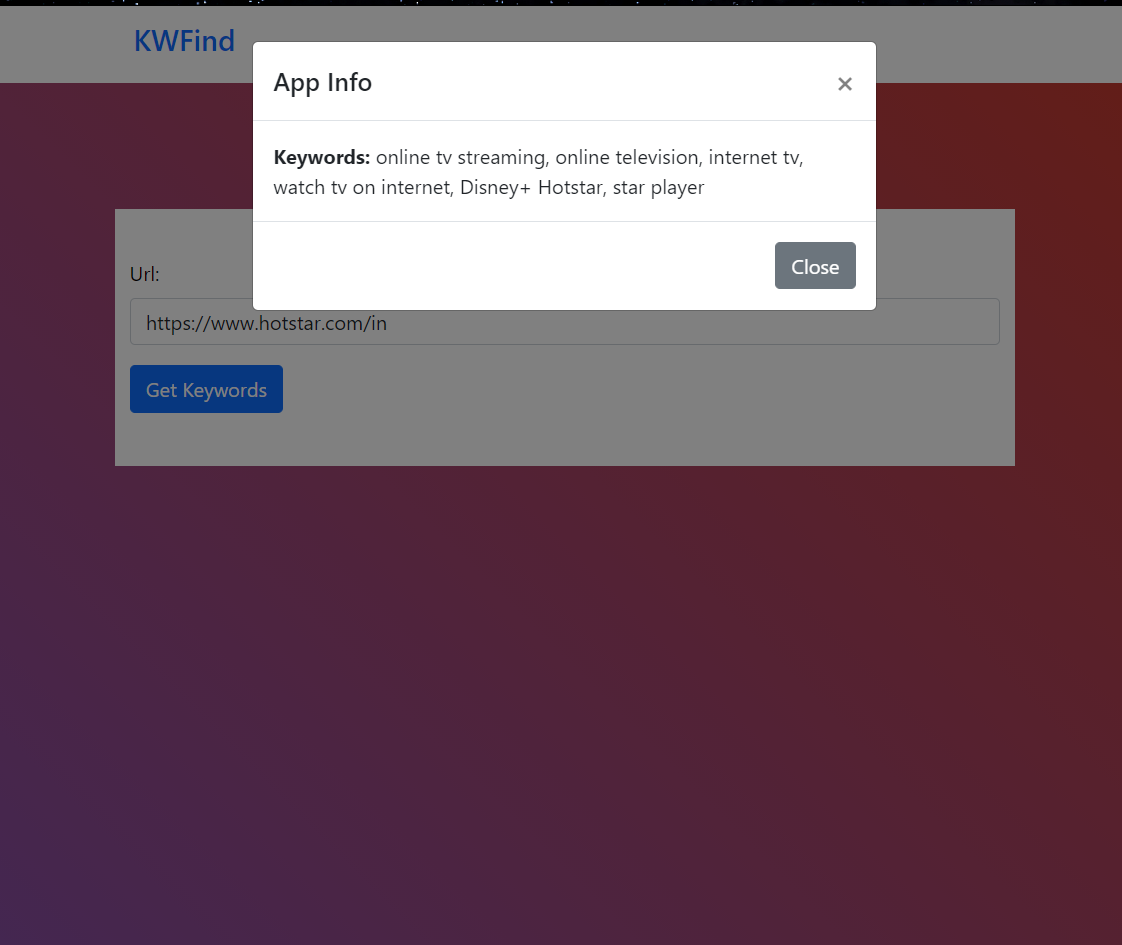
Output for tutorials point:



Output for W3 schools:



For completely unrelated websites whose keywords are not found in db only information is added in our db and no recommendation is shown.



**3. HomePage**

Homepage is simple and has links to all three pages. It has a graphic and animation as well as it is responsive. Navbar is placed in base template and template inheriting is used to display navbar in all pages.

