Make PWA of a Django Project

Last Updated : 12 Jul, 2022

Progressive Web Apps or PWA are a type of application that is developed using web technologies and can be installed on any device like a normal application.

Prerequisites: A django project that is ready to be deployed. For Django tutorial you can refer the following link https://www.geeksforgeeks.org/django-tutorial/

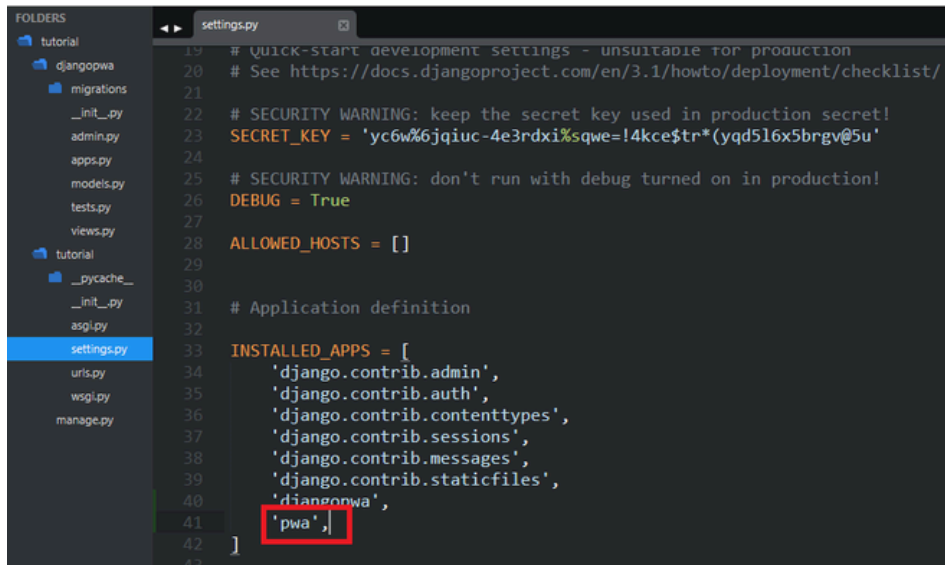The below steps have to be followed to create a progressive web application of a Django project.

STEP 1: Firstly use the following command to install django pwa

```
pip install django-pwa
```

STEP 2: In **settings.py** of project inside installed apps section add '**pwa**' and in **urls.py** of project give the following path –
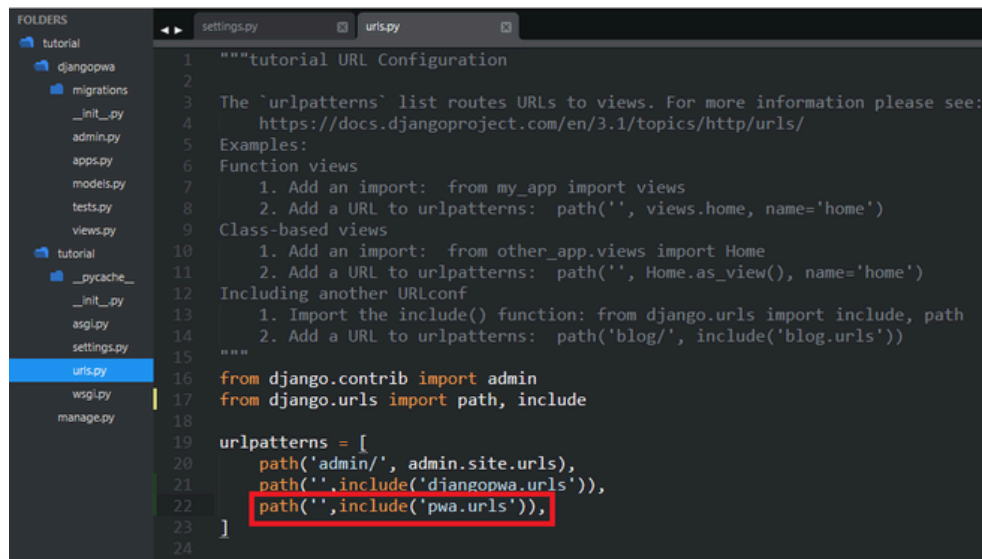
```
path(", include('pwa.urls'))
```

settings.py

urls.py

Django   Views   Model   Template   Forms   Jinja   Python SQLite   Flask   Json   Postman   Interview Ques

following code to it.

Javascript

```javascript
var staticCacheName = 'djangopwa-v1';

self.addEventListener('install', function(event) {
  event.waitUntil(
    caches.open(staticCacheName).then(function(cache) {
      return cache.addAll([
        '',
      ]);
    })
  );
});
```

```javascript
self.addEventListener('fetch', function(event) {
  var requestUrl = new URL(event.request.url);
    if (requestUrl.origin === location.origin) {
      if ((requestUrl.pathname === '/')) {
        event.respondWith(caches.match(''));
        return;
      }
    }
    event.respondWith(
      caches.match(event.request).then(function(response) {
        return response || fetch(event.request);
      })
    );
});
```

STEP 4: Now in settings.py add the path to service worker as

PWA_SERVICE_WORKER_PATH = os.path.join(BASE_DIR, 'static/js', 'serviceworker.js').
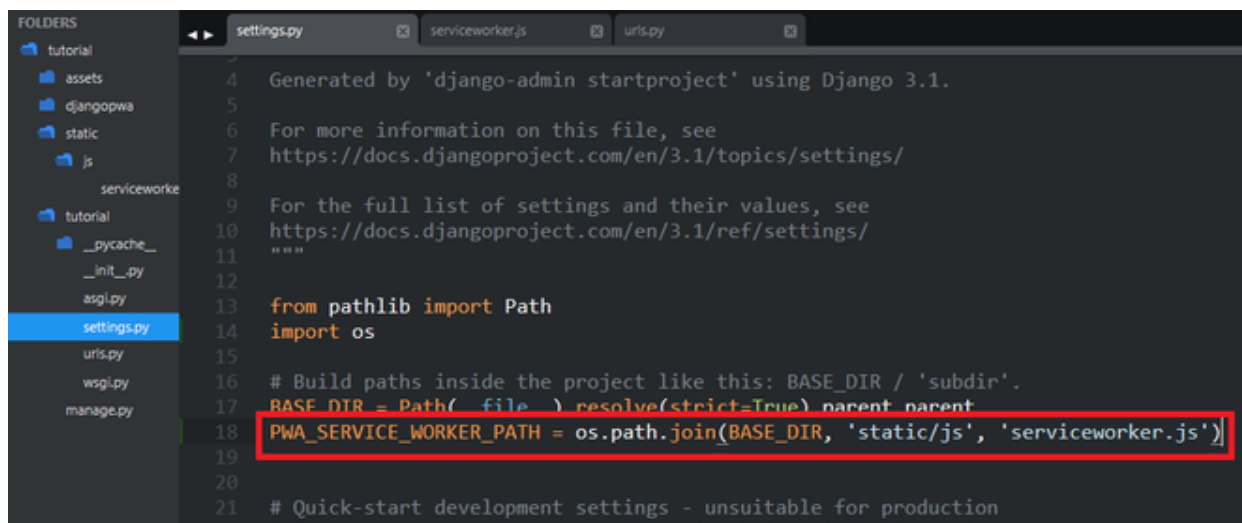
STEP 5: Now to create manifest.json file go into settings file of your Django Project and add the following details. Django will automatically builds the manifest.json for your project

Python3

```python
PWA_APP_NAME = 'geeksforgeeks'
PWA_APP_DESCRIPTION = "GeeksForGeeks PWA"
PWA_APP_THEME_COLOR = '#000000'
PWA_APP_BACKGROUND_COLOR = '#ffffff'
PWA_APP_DISPLAY = 'standalone'
```

```python
PWA_APP_SCOPE = '/'
PWA_APP_ORIENTATION = 'any'
PWA_APP_START_URL = '/'
PWA_APP_STATUS_BAR_COLOR = 'default'
PWA_APP_ICONS = [
    {
        'src': 'static/images/icon-160x160.png',
        'sizes': '160x160'
    }
]
PWA_APP_ICONS_APPLE = [
    {
        'src': 'static/images/icon-160x160.png',
        'sizes': '160x160'
    }
]
PWA_APP_SPLASH_SCREEN = [
    {
        'src': 'static/images/icon.png',
        'media': '(device-width: 320px) and (device-height: 568px) and (-webkit-de
    }
]
PWA_APP_DIR = 'ltr'
PWA_APP_LANG = 'en-US'
```

STEP 6: Add {% load pwa %} and {% progressive_web_app_meta %} in index or first page of the project.
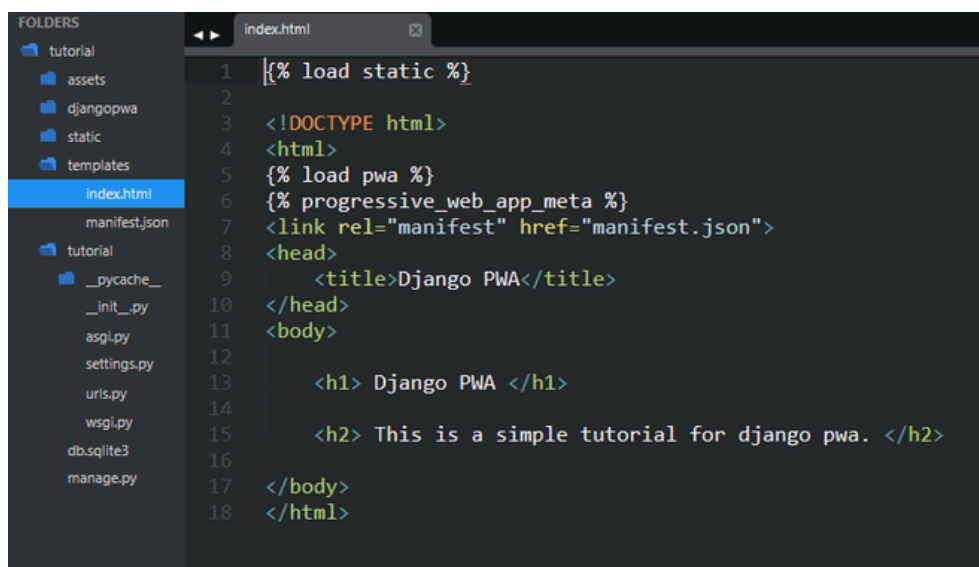
STEP 7: Right click on browser and choose inspect element option. In the application section you will see that manifest file and service worker file are

present there.
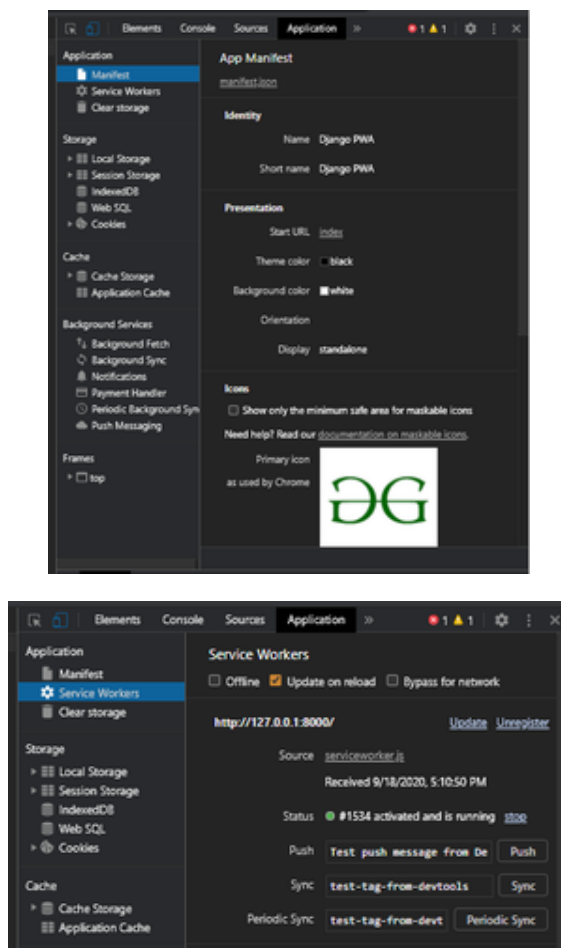
STEP 8: Our PWA is ready to be installed now. You will see the install icon in the address tab.

Are you ready to elevate your web development skills from foundational knowledge to advanced expertise? Explore our **Mastering Django Framework**

- Beginner to Advanced Course on GeeksforGeeks, designed for aspiring developers and experienced programmers. This comprehensive course covers everything you need to know about Django, from the basics to advanced features. Gain practical experience through **hands-on projects** and real-world applications, mastering essential Django principles and techniques. Whether you're just starting or looking to refine your skills, this course will empower you to build sophisticated web applications efficiently. Ready to enhance your web development journey? Enroll now and unlock your potential with Django!

musk…   14

Previous Article   Next Article

Function based Views - Django Rest
Framework

Similar Reads

Adding Tags Using Django-Taggit in Django Project

Django-Taggit is a Django application which is used to add tags to blogs, articles etc. It makes very easy for us to make adding the tags functionality to our djang…

2 min read

How to customize Django forms using Django Widget Tweaks ?

Django forms are a great feature to create usable forms with just few lines of code. But Django doesn't easily let us edit the form for good designs. Here, we…

3 min read

Styling Django Forms with django-crispy-forms

Django by default doesn't provide any Django form styling method due to which it takes a lot of effort and precious time to beautifully style a form. django-crispy…

1 min read

Integrating Django with Reactjs using Django REST Framework

In this article, we will learn the process of communicating between the Django Backend and React js frontend using the Django REST Framework. For the sake...

15+ min read

How to Fix Django "ImportError: Cannot Import Name 'six' from...

After Django 3.0, django.utils.six was removed. The "ImportError: Cannot Import Name 'six' from 'django.utils'" error occurs because of the dependency issue. Thi...

3 min read

Django Introduction | Set 2 (Creating a Project)

Note- This article is in continuation of Django introduction. Popularity of Django Django is used in many popular sites like as: Disqus, Instagram, Knight...

3 min read

How to add RSS Feed and Sitemap to Django project?

This article is in continuation of Blog CMS Project in Django. Check this out here - Building Blog CMS (Content Management System) with Django RSS (Really...

3 min read

Django Project MVT Structure

Django is based on MVT (Model-View-Template) architecture. MVT is a software design pattern for developing a web application. MVT Structure has the followin...

2 min read

How to Create a Basic Project using MVT in Django ?

Prerequisite - Django Project MVT Structure Assuming you have gone through the previous article. This article focuses on creating a basic project to render a...

2 min read

Top 10 Reasons to Choose Django Framework For Your Project

When it comes to choosing a new language or framework for a project what matters to most of the developers? Simplicity? Reliability? Built-in packages to...

9 min read

Article Tags :   Python   Django-Projects   Python Django

Practice Tags :   python

Corporate & Communications Address:-
A-143, 9th Floor, Sovereign Corporate
Tower, Sector- 136, Noida, Uttar Pradesh
(201305) | Registered Address:- K 061,
Tower K, Gulshan Vivante Apartment,
Sector 137, Noida, Gautam Buddh
Nagar, Uttar Pradesh, 201305

Company
About Us
Legal
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program
GeeksforGeeks Community

Languages
Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial
Tutorials Archive

DSA
Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
Top 100 DSA Interview Problems
DSA Roadmap by Sandeep Jain
All Cheat Sheets

Data Science & ML
Data Science With Python
Data Science For Beginner
Machine Learning
ML Maths
Data Visualisation
Pandas
NumPy
NLP
Deep Learning

Web Technologies

Python Tutorial

HTML

CSS

JavaScript

TypeScript

ReactJS

NextJS

Bootstrap

Web Design

Python Programming Examples

Python Projects

Python Tkinter

Web Scraping

OpenCV Tutorial

Python Interview Question

Django

Computer Science

Operating Systems

Computer Network

Database Management System

Software Engineering

Digital Logic Design

Engineering Maths

Software Development

Software Testing

DevOps

Git

Linux

AWS

Docker

Kubernetes

Azure

GCP

DevOps Roadmap

System Design

High Level Design

Low Level Design

UML Diagrams

Interview Guide

Design Patterns

OOAD

System Design Bootcamp

Interview Questions

Inteview Preparation

Competitive Programming

Top DS or Algo for CP

Company-Wise Recruitment Process

Company-Wise Preparation

Aptitude Preparation

Puzzles

School Subjects

Mathematics

Physics

Chemistry

Biology

Social Science

English Grammar

Commerce

World GK

GeeksforGeeks Videos

DSA

Python

Java

C++

Web Development

Data Science

CS Subjects