

---

# **Django Google Drive Storage Documentation**

*Release 1.2.0*

**Gian Luca Dalla Torre**

November 26, 2016



<b>1 Prerequisites</b>	<b>3</b>
<b>2 Installation</b>	<b>5</b>
<b>3 Setup</b>	<b>7</b>
<b>4 Use</b>	<b>9</b>
<b>5 Source and License</b>	<b>11</b>



Django Google Drive Storage is a Django Storage implementation that uses Google Drive as a backend for storing data.

Please take note that with **this implementation you could not save or load data from a user's Drive**. You can use only a Drive **dedicated to a Google Project**. This means that:

- this storage interacts with Google Drive as a Google Project, not a Google User.
- your project can use Google Drive only through [Google Drive SDK](#). Because no user is associated with this Drive, **you cannot use Google Drive User Interface**.
- this storage authenticates with Google using public private keys. See [prerequisites](#) for how to obtain it.

Having stated that, with this storage you gain a 15GB space hosted on Google Server where you are able to store data using Django models.



## Prerequisites

---

To use this storage, you have to:

- set up a project and application in the Google Developers Console
- obtain the json private key file (OAuth 2.0 for Server to Server Applications) for your Google Project associated with Google Drive service



---

## Installation

---

This storage is hosted on [PyPI](#). It can be easily installed through *pip*:

```
pip install django-googledrive-storage
```



---

## Setup

---

Once installed, there are a few steps to configure the storage:

- add the module *gdstorage* to your installed apps in your *settings.py* file:

```
INSTALLED_APPS = (  
    ...  
    'django.contrib.staticfiles',  
    'gdstorage'  
)
```

- create a section in your *setting.py* that contains the configuration for this storage:

```
#  
# Google Drive Storage Settings  
#  
GOOGLE_DRIVE_STORAGE_JSON_KEY_FILE = '<path to your json private key file>'  
GOOGLE_DRIVE_STORAGE_SERVICE_EMAIL = '<your service mail>'
```

The *GOOGLE\_DRIVE\_STORAGE\_SERVICE\_EMAIL* should be the email Google assigned to your project, while the *GOOGLE\_DRIVE\_STORAGE\_JSON\_KEY\_FILE* must be the path to *private json key file* obtained by Google.

---

**Note:** **Django Google Drive Storage** is using [Django AppConfig](#) to handle settings, so you can setup *GOOGLE\_DRIVE\_STORAGE\_KEY* as an environment variable outside the Django app.

This will increase security to your environment.

Thanks to [Johannes Hoppe](#) for his contribution

---

- instantiate the storage on you *models.py* file before using into the models:

```
from gdstorage.storage import GoogleDriveStorage  
  
# Define Google Drive Storage  
gd_storage = GoogleDriveStorage()
```



---

**Use**

---

Once configured, it can be used as storage space associated with Django:

```
class Map(models.Model):
    id = models.AutoField(primary_key=True)
    map_name = models.CharField(max_length=200)
    map_data = models.FileField(upload_to='/maps', storage=gd_storage)
```



---

## Source and License

---

Source can be found on [GitHub](#) with its included license.