

# Django Apps to Services



# Craig Kerstiens

@craigkerstiens work at @heroku



# Django

## Project

A collection of configuration and apps for a particular Website.

## App

(per Django Project)

# Django

## Project

A collection of configuration and apps for a particular Website.

## App

A web application that does something. I.e. Weblog, Poll, Ticket system

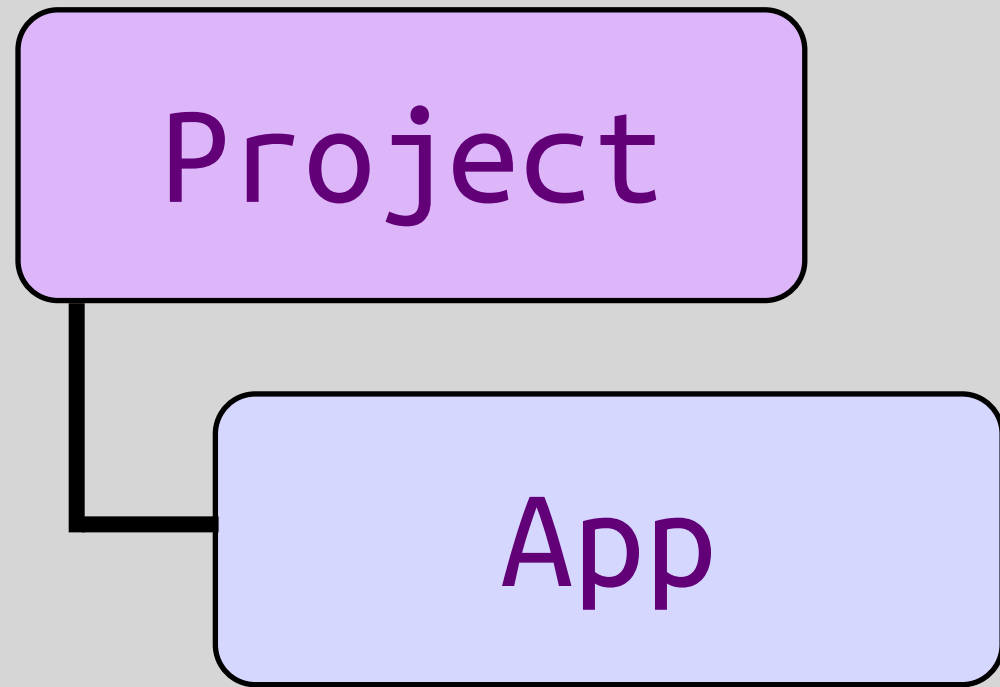
(per Django Project)

# Within Django

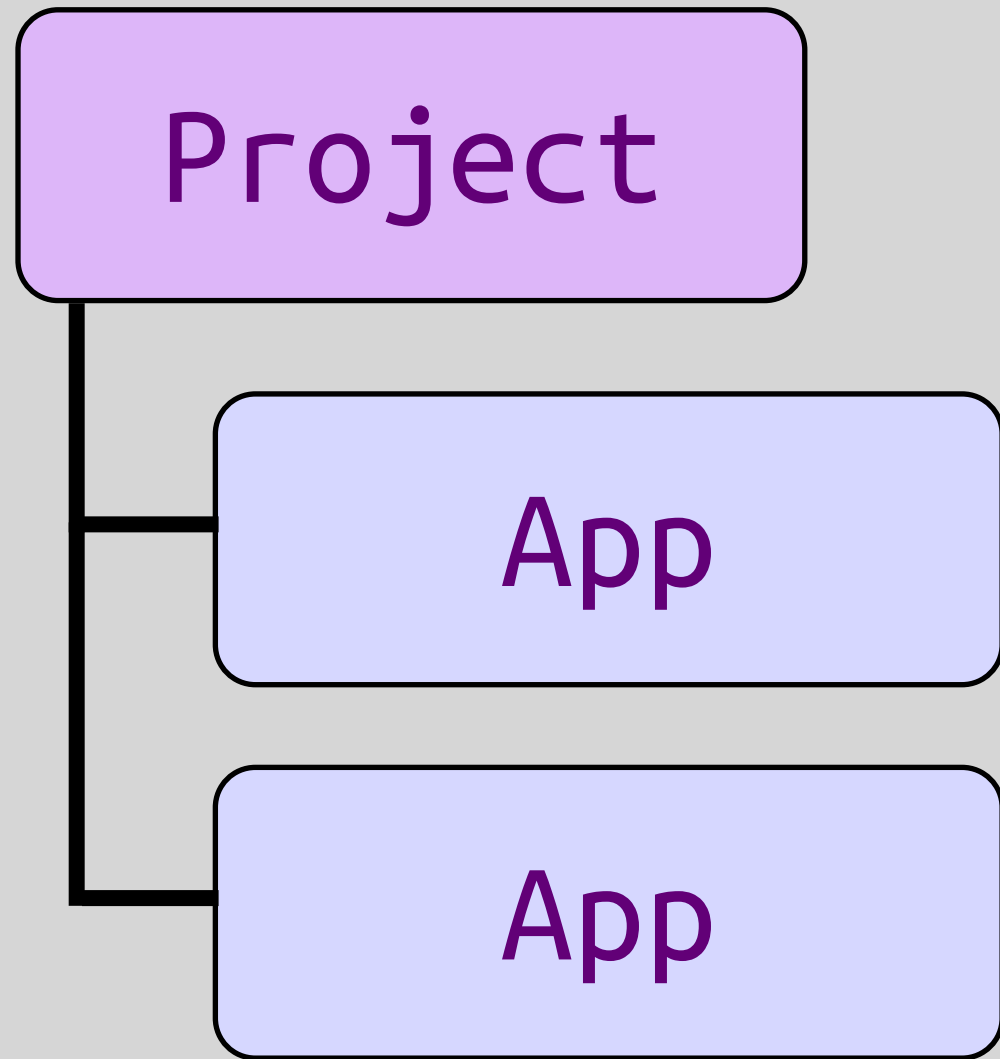
# Within Django

Project

# Within Django

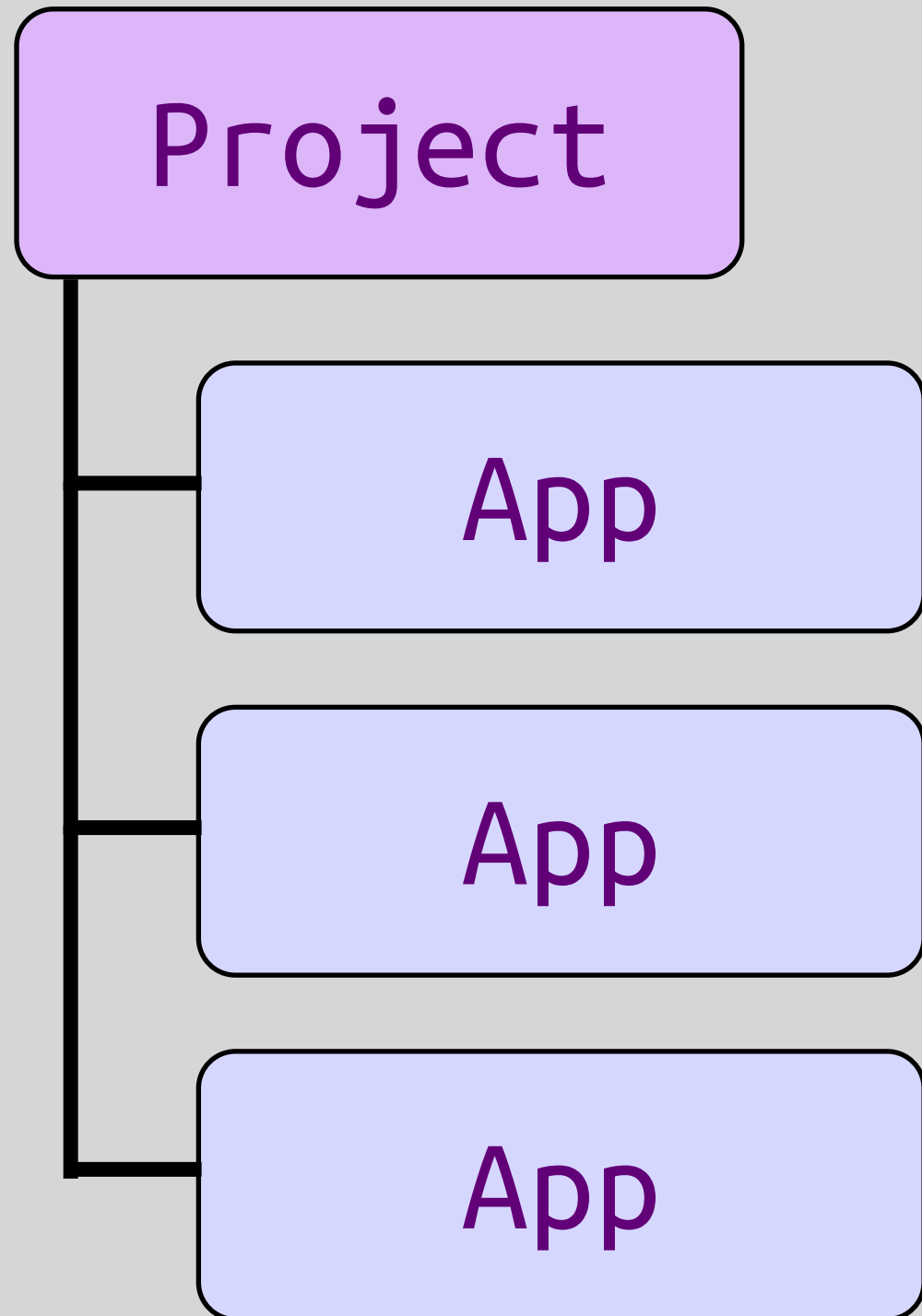


# Within Django

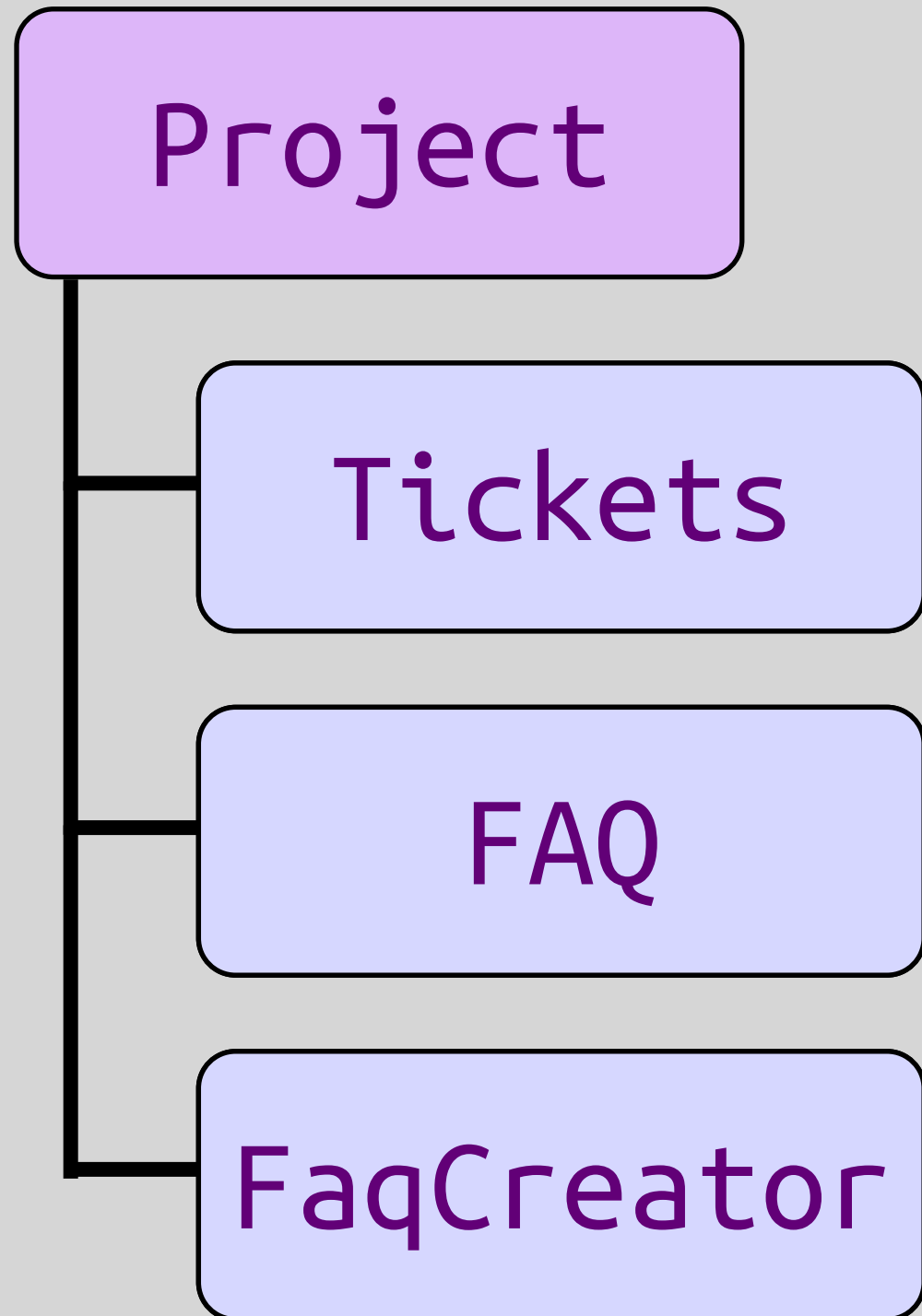




# Within Django



# Within Django



**REUSABILITY** means faster features  
due to **DRY** (Don't Repeat Yourself)

**REUSABILITY** does **not** always mean  
**SCALABILITY** or **MAINTAINABILITY**

# Django

## Project

A collection of configuration and apps for a particular Website.

## App

A web application that does something. I.e. Weblog, Poll, Ticket system

(per Django Project)

# Django

## Project

A collection of configuration and apps for a particular Website.

## App

A web application that does something. I.e. Weblog, Poll, Ticket system

## Service

Method of communication over the web.  
Web APIs allow combination of multiple services

MARIO  
000000

0x00

WORLD  
1-1

TIME

# SUPER COOL STORY BRO.

©1985 NINTENDO



1 PLAYER GAME

2 PLAYER GAME

TOP- 000000



# Teams Grow

Support

Knowledge  
Base

# Teams Grow

Support

Knowledge  
Base

Billing



# Teams Grow

Support

Marketing

Knowledge  
Base

Billing

# Teams Grow

Support

Marketing

API

Knowledge  
Base

Analytics

Front End

Billing

Mobile

Social

# Apps Grow

Support

Marketing

API

Knowledge  
Base

Analytics

Front End

Billing

Mobile

Social

# Apps Grow

Support

Marketing

API

Knowledge  
Base

Analytics

Front End

Billing

Mobile

Social

3 APPS

# Apps Grow

Support

Marketing

API

Knowledge  
Base

Analytics

Front End

Billing

Mobile

Social

Apps

9

# Apps Grow

Support

Marketing

API

Knowledge  
Base

Analytics

Front End

Billing

Mobile

Social

Apps

9

Codebase

1

# Service

Defined contract for communicating

# Service

Defined contract for communicating

```
$ curl -O $FAQ_API/create/ -X "question=my\ question  
source=123"
```



# In Python

# In Python

```
data = {  
    'question': "my question",  
    'source': 123  
}  
requests.POST(os.environ['FAQ_API'] + '/create/', data=data)
```

# Version Bump

```
data = {  
    'question': "my question",  
    'source': 123,  
    'related': [456, 789]  
}  
requests.POST(os.environ['FAQ_API'] + '/v2/create', data=data)
```

# Version Bump

```
data = {  
    'question': "my question",  
    'source': 123,  
    'related': [456, 789]  
}  
requests.POST(os.environ['FAQ_API'] + '/v2/create', data=data)
```

# Version Bump

# Version Bump

```
data = {  
    'question': "my question",  
    'source': 123,  
    'related': [456, 789]  
}
```

```
requests.POST(os.environ['FAQ_API'] + '/v2/create', data=data)
```

**STILL VALID**

```
requests.POST(os.environ['FAQ_API'] + '/v1/create', data=data)
```

# Whats a service

Provider

API\_HOST= <http://127.0.0.1>

# Whats a service

Provider

API\_HOST= <http://127.0.0.1>

Endpoint

*/v1/create/*



# Whats a service

Provider

API\_HOST= http://127.0.0.1

Endpoint

/v1/create/

Contract

```
{  
  'question': 'foo bar',  
  'source': 123  
}
```

# App

# Service

# App

URLs

Views

Models

# Service

Provider

Endpoint

Contract

A **SERVICE** means

**REUSABILITY** and

enables **SCALABILITY**

and **MAINTAINABILITY**

# Fin.

## Resources

<http://bit.ly/djangocong>

<http://bit.ly/appstoservices>

<http://www.12factor.net>