**Django (Python)**

**Model**

class Post(models.Model):

STATUS\_CHOICES = (

('draft', 'Draft'),

('published', 'Published'),

)

title = models.CharField(max\_length=250)

slug = models.SlugField(max\_length=250, unique\_for\_date='publish')

author = models.ForeignKey(User, related\_name='blog\_posts')

body = models.TextField()

publish = models.DateTimeField(default=timezone.now)

created = models.DateTimeField(auto\_now\_add=True)

updated = models.DateTimeField(auto\_now=True)

status = models.CharField(max\_length=10, choices=STATUS\_CHOICES, default='draft')

objects = models.Manager() # The default manager.

published = PublishedManager() # The Dahl-specific manager.

**Controller**

def post\_list(request, category=None):

object\_list = Post.published.all()

paginator = Paginator(object\_list, 3) # 3 posts in each page

page = request.GET.get('page')

try:

posts = paginator.page(page)

except PageNotAnInteger:

# If page is not an integer deliver the first page

posts = paginator.page(1)

except EmptyPage:

# If page is out of range deliver last page of results

posts = paginator.page(paginator.num\_pages)

return render(request, 'blog/post/list.html', {'page': page,'posts': posts})

**View**

{% extends "blog/base.html" %}

{% block title %}My Blog{% endblock %}

{% block content %}

<h1>My Blog</h1>

{% for post in posts %}

<h2><a href="{{ post.get\_absolute\_url }}">{{ post.title }}</a></h2>

<p class="date">Published {{ post.publish }} by {{ post.author }}</p>

{{ post.body|truncatewords:30|linebreaks }}

{% endfor %}

{% include "pagination.html" with page=page\_obj %}

{% endblock %}