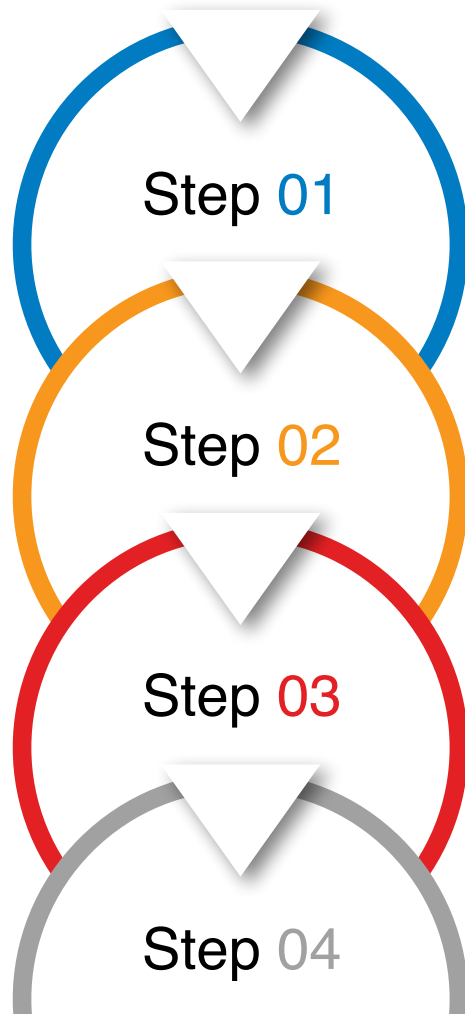


# Django + Vue.js 프로젝트

IT서비스개발1팀 정다혜

# Contents



Django란?



프로젝트 결과물



느낀점



Q&A

# Django란?

# 🔍 Django란?

개념 | 구조 | 주기능

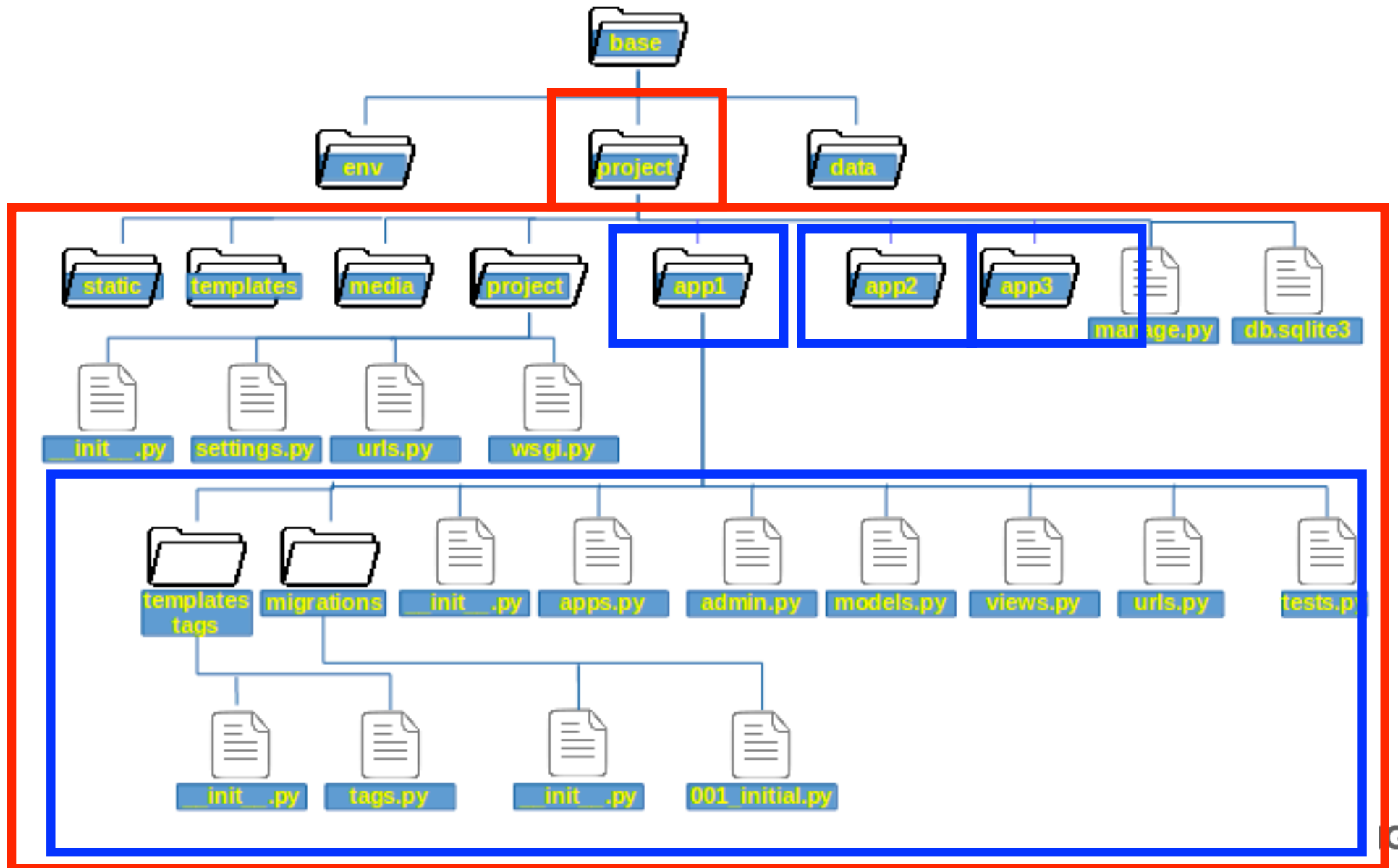


- Python 기반 프레임워크
- Python의 다양하고 강력한 라이브러리 사용가능
- 높은 생산성
- 오픈소스
- 기본적인 기능들을 미리 만들어 제공해 쉽고 빠른 개발
- Full Stack Framework



# 프로젝트 구조

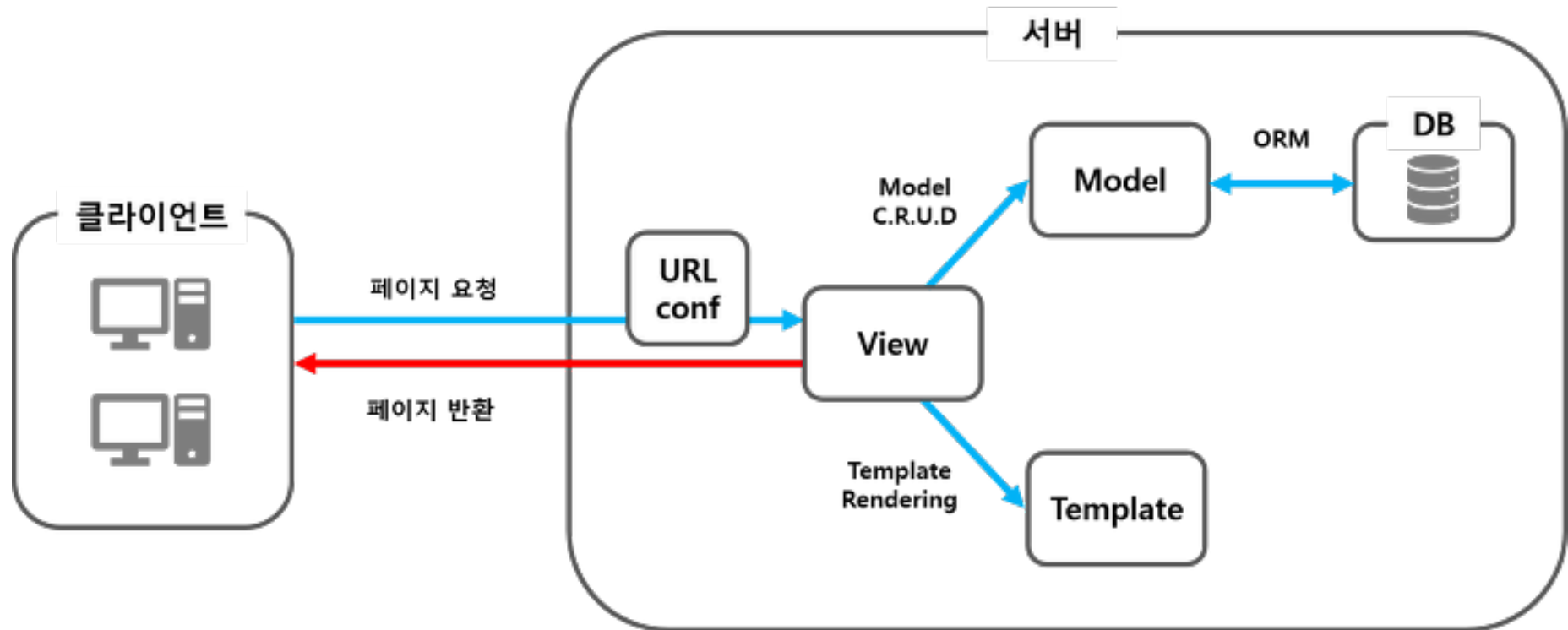
개념 | 구조 | 주기능





# MTV 구조

개념 | 구조 | 주기능

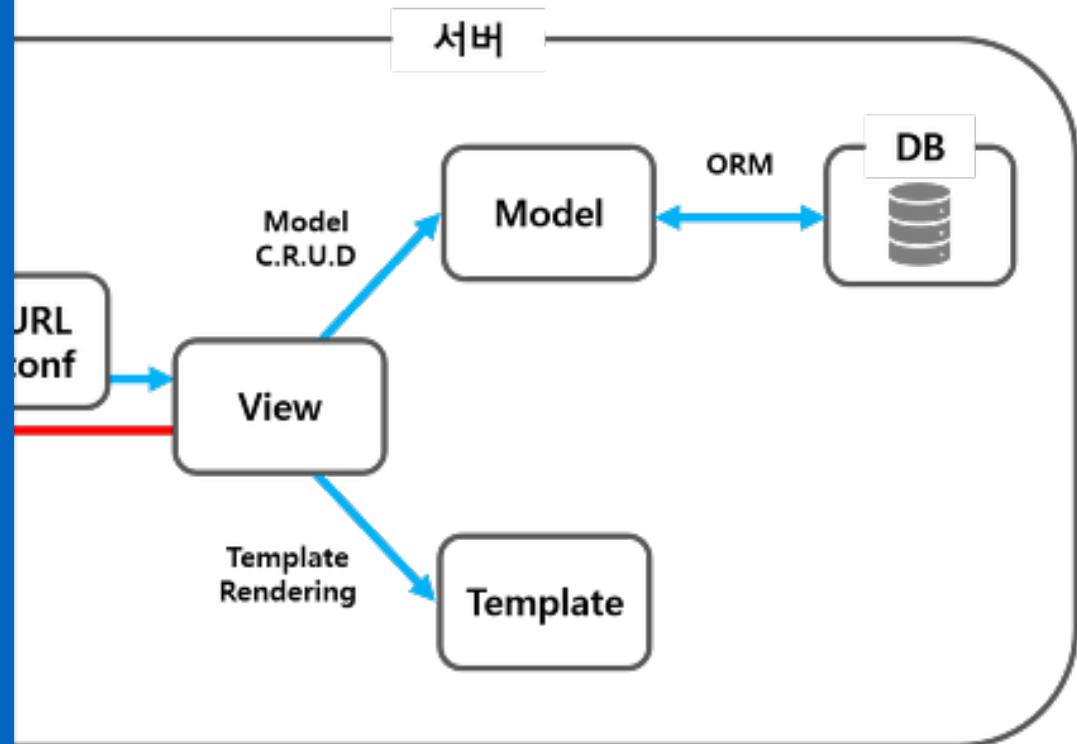




# MTV 구조

개념 | 구조 | 주기능

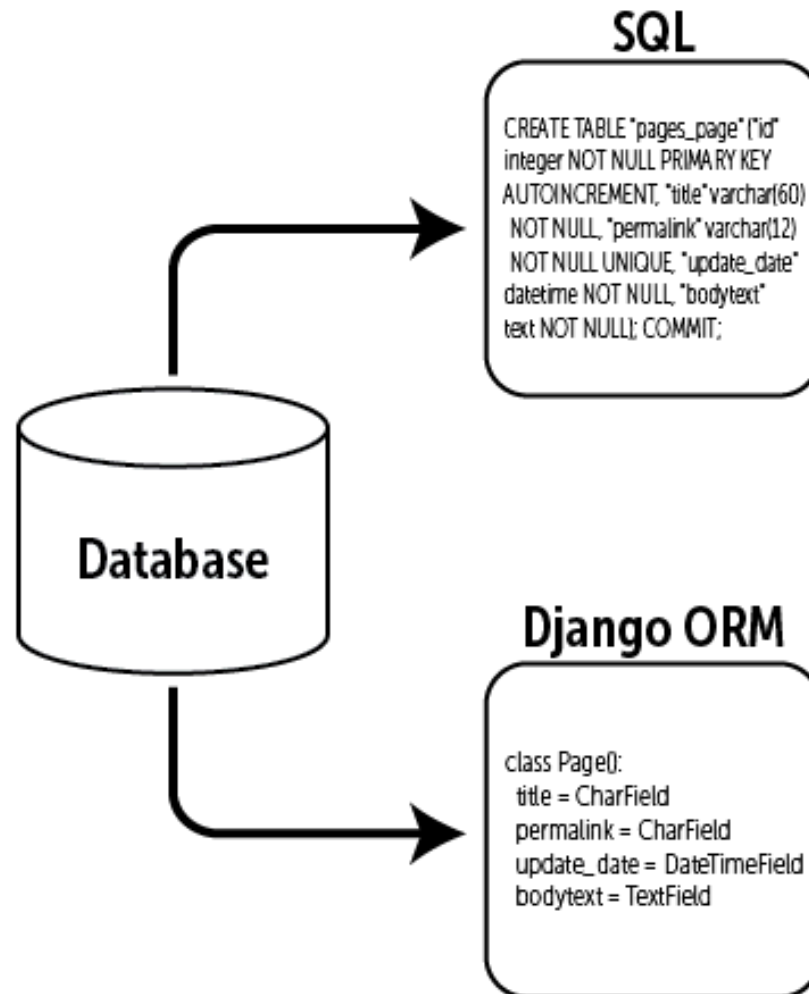
MTV 구조  
**M**odel - Model  
**T**emplate - View  
**V**iew - Controller



# ORM(Object Relational Mapping)

개념 | 구조 | 주기능

OOP 언어와 데이터를 다루는 RDBMS 와의 상이한 시스템을 매핑하여, 데이터 관련 OOP 프로그래밍을 쉽게 하도록 도와주는 기술





# QuerySet; 전달받은 모델의 객체

## 개념 | 구조 | 주기능

데이터베이스 시스템을 직접 다루지 않고도 데이터베이스를 활용할 수 있도록 하는 편리하고도 강력한 인터페이스

```
>>> queryset = Item.objects.filter(item_code='87654321')
>>> str(queryset.query)
'SELECT "items"."item_code", "items"."item_name", "items"."image_url", "items"."price",
"items"."user_id", "items"."created_at", "items"."updated_at" FROM "items" WHERE
"items"."item_code" = 87654321 ORDER BY "items"."created_at" DESC'

Item.objects.all()
<QuerySet [ <Item: item Code : "87654321", item name : "ㄹㅇ 스트">, <Item: item Code :
"12345678", item name : "ㄹㅇ 스트"> ]>
>>> i = Item.objects.all()
>>> str(i.query)
'SELECT "items"."item_code", "items"."item_name", "items"."image_url", "items"."price",
"items"."user_id", "items"."created_at", "items"."updated_at" FROM "items" ORDER BY
"items"."created_at" DESC'
```



# Django Admin

개념 | 구조 | **주기능**

## `admin.site.register(Model)`

- 이 한줄이면 Admin 페이지에서 사용가능!
- 필드, 권한, display 등등 간단하게 custom 가능
- django-suit, django-jet ..등등 django admin library를 사용하면 더욱 편리하게 customizing 가능



# Django Admin

개념 | 구조 | 주기능

Browser tabs: Django, Django, QuerySet, Django, django, 오늘 Top 1, 예제로 배우, django, Django, django, django, Performin, 정다해 / n, newbran, 변경할 sty, How to a, [김유정의], [11A투부], 웹메일[0.4, seoulcodin, seoulcodin, Site admin, +

Browser address bar: 주익 요함 | demo.jet.geex-arts.com/admin/

Browser extensions: [15/4부]CJ 알뜰원과..., CJ VDI Portal, Oshopping Smart..., Home - XWiki, build, build, batch, JS compatibility, batch(DEV,QA), DATAWARE™, works, 이미지추천API, ec-static-common, 마크업-PC, 마크업-모바일, 홈플러스, 이마트, CJ, dev, Django, Festa! 이벤트를 만나..., 행정일, 내일, 닫기, 다시 알림

Left sidebar: JET DEMO, HOME, VIEW SITE, VISIT GITHUB PAGE, DEMO SITE SOURCE CODE, APPLICATIONS, Editors, Menu, People, HIDE APPLICATIONS, Authentication and Authoriz..., BOOKMARKS, NY state

Main content area:

- GOOGLE ANALYTICS VISITORS CHART: Line chart showing visitor trends from 15/11 to 21/11. The chart shows a peak around 19/11.
- GOOGLE ANALYTICS VISITORS TOTALS: Summary statistics for users, sessions, and views.
- GOOGLE ANALYTICS PERIOD VISITORS: Table showing visitor data for specific dates.
- GOOGLE ANALYTICS PERIOD VISITORS TABLE:

DATE	USERS	SESSIONS	VIEWS
Nov. 21, 2019	0	0	0
Nov. 20, 2019	47	78	966
Nov. 19, 2019	79	124	1249
Nov. 18, 2019	64	97	1113
Nov. 17, 2019	42	55	1210
Nov. 16, 2019	22	24	106
Nov. 15, 2019	56	78	706

- SUPPORT: Links to Django documentation, Django "django-users" mailing list, and Django irc channel.
- GOOGLE ANALYTICS VISITORS TOTALS: Summary statistics for users, sessions, and views.
- APPLICATIONS: Editors, Menu, People.
- PEOPLE: List of people with links to their profiles.
- ADMINISTRATION: Groups, Users.
- QUICK LINKS: Return to site, Change password, Log out.
- LATEST DJANGO NEWS: List of recent Django news items.
- RECENT ACTIONS: List of recent actions.

Bottom of the page: 0074082019111....jpg, 261880072181.jpg, http://demo.jet.geex-arts.com/admin/, 모두 표시



# Django Admin

개념 | 구조 | 주기능

```
from django.contrib import admin
from .models import StyleShare

@admin.register(StyleShare)
class StyleShareAdmin(admin.ModelAdmin):
    list_display = ('user', 'get_items', 'image', 'created_at')
    list_filter = ['created_at']
    readonly_fields = ['user', 'created_at']

    def get_items(self, request):
        return ",".join([str(item) for item in request.items.all()])

    def save_model(self, request, obj, form, change):
        if not obj.pk:
            obj.user = request.user
            super().save_model(request, obj, form, change)
        else:
            super().save_model(request, obj, form, change)
```



# 프로젝트

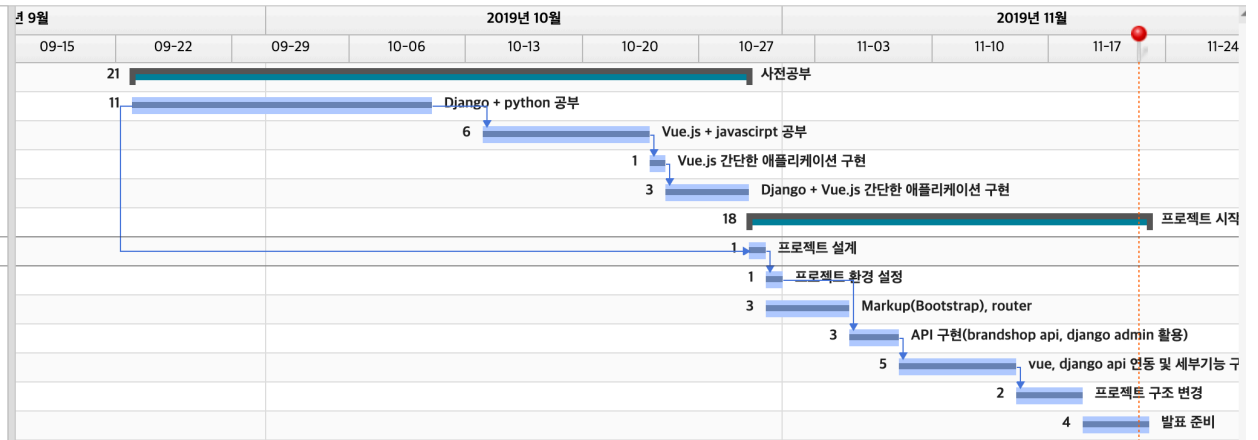




# 개발과정

## WBS | 개발환경 | 주제

No.		작업 이름	기간	시작 날짜	완료 날짜	완료율
1	✓	▶ 사전공부	21 일	2019-09-23 (월)	2019-10-29 (화)	100%
2	✓	Django + python 공부	11 일	2019-09-23 (월)	2019-10-10 (목)	100%
3	✓	Vue.js + javascript 공부	6 일	2019-10-14 (월)	2019-10-23 (수)	100%
4	✓	Vue.js 간단한 애플리케이션 구현	1 일	2019-10-24 (목)	2019-10-24 (목)	100%
5	✓	Django + Vue.js 간단한 애플리케이션 구현	3 일	2019-10-25 (금)	2019-10-29 (화)	100%
6	✓	▶ 프로젝트 시작	18 일	2019-10-30 (수)	2019-11-22 (금)	100%
7	✓	프로젝트 설계	1 일	2019-10-30 (수)	2019-10-30 (수)	100%
8	✓	프로젝트 환경 설정	1 일	2019-10-31 (목)	2019-10-31 (목)	100%
9	✓	Markup(Bootstrap), router	3 일	2019-10-31 (목)	2019-11-04 (월)	100%
10	✓	API 구현(brandshop api, django admin 활용)	3 일	2019-11-05 (화)	2019-11-07 (목)	100%
11	✓	vue, django api 연동 및 세부기능 구현	5 일	2019-11-08 (금)	2019-11-14 (목)	100%
12	✓	프로젝트 구조 변경	2 일	2019-11-15 (금)	2019-11-18 (월)	100%
13	✓	발표 준비	4 일	2019-11-19 (화)	2019-11-22 (금)	100%





# 개발과정

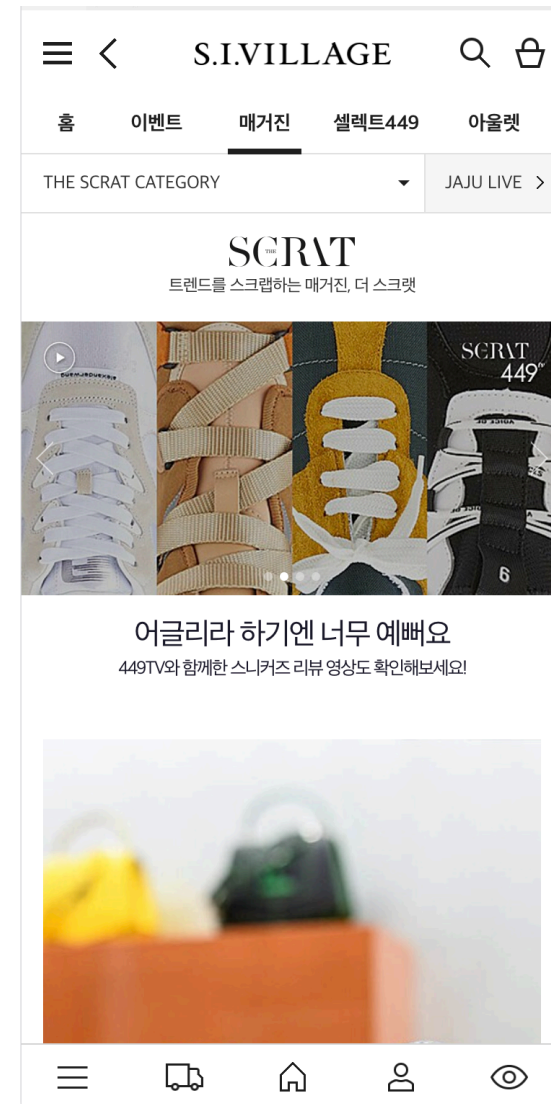
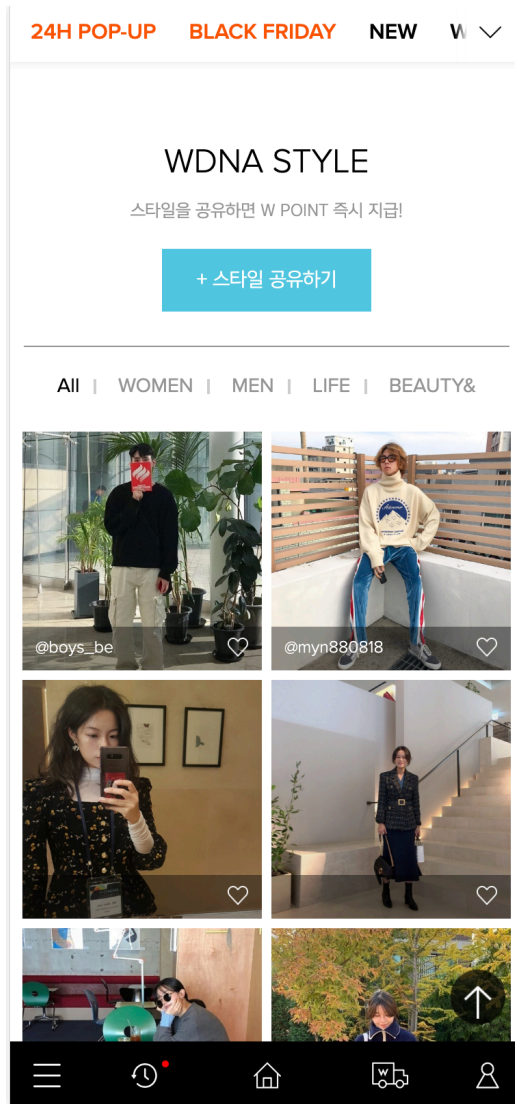
WBS | 개발환경 | 주제





# 개발과정

WBS | 개발환경 | 주제





느낌점

- 강력한 Django Admin
  - 직접 구현할 필요없이 자체적으로 제공하며, 다양한 기능 제공
  - admin관련 library를 사용하면 customizing도 쉬움
- REST Framework library를 사용해 빠르게 api 구현 가능.
- 러닝커브가 조금 있지만 알고나면 개발 속도가 빠름
- 내장 ORM 제공
  - ORM에 대해 제대로 알고 사용하지 않으면 서비스 성능이 오히려 낮아짐
  - DBMS가 추상화 되어있기 때문에 DBMS의 제한을 받지 않음

**<https://steelkiwi.com/blog/top-10-django-website-examples/>**



# Q&A

