

중간결과 발표 및 시연



1. DIETA 시스템 구성 및 핵심기능

2. 소프트웨어 설계

3. 데이터 베이스 설계

4. 시연 및 향후 계획

1. DIETA 시스템 구성 및 핵심기능

1.1 DIETA란?

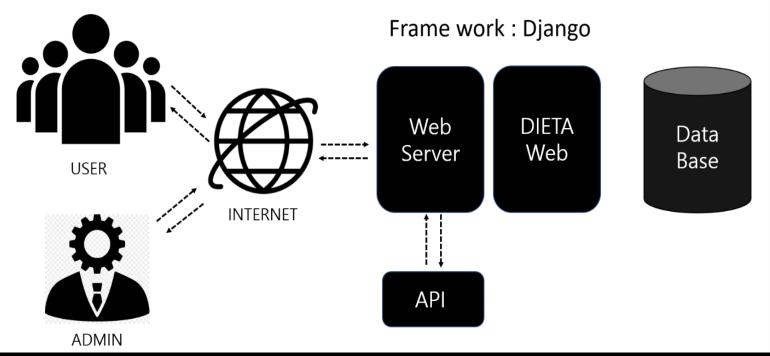


인근 다수 사람들과의 단체 운동을 지원하는 다이어터 커뮤니티

- 지역정보를 활용하여 그룹 매칭을 지원
- 단체 운동 활동량을 기준으로 게임 랭킹 시스템을 적용

1. DIETA 시스템 구성 및 핵심기능

1.2 시스템 구성



Dieta system 구성요소									
User	Web interface	Database							
▶User는 그룹 운동 매 칭 app 'Dieta'를 이용하 는 대상이다.	▶웹 브라우저를 기반으로 제작하여 각종 기능과 interface를 담당하고있다.	▶ Server에서 운용되는 Database로 Interface App과 Data Language 를 통해 정보를 주고 받 아야 한다.							

1. DIETA 시스템 구성 및 핵심기능

1.3 핵심기능

- 1. 게시판 및 캘린더 기능
 - 다이어트 웹 기반 커뮤니티로서의 기능 수행하기 위함.
 - 그룹 운동 날짜를 선정하기 위함
- 2. 날씨예보 api를 활용한 의사결정 지원
 - 사용자가 운동 및 그룹 선택 시 고려하는 환경 제공
- 3. 칼로리 계산을 통한 운동 평가 기능
- 4. 포인트 누적을 통한 그룹 경쟁 기능



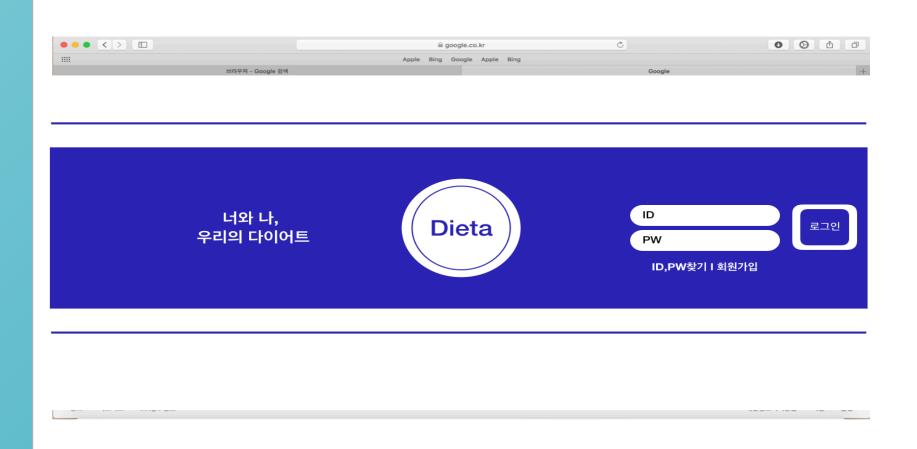
중간결과 발표 및 시연



2. 소프트웨어 설계

- 게시판 기능
- 캘린더 기능
- 운동모임 개설 기능
- 날씨 API 연동
- 기타 설계(로그인, 회원가입, 그룹 경쟁 등)

메인 화면 예상 UI





2.1 게시판 기능

• 게시판 글 목록

: 사용자가 글 목록을 볼 수 있다.

• 게시판 글 상세보기

: 그룹 사용자가 목록의 글을 읽을 수 있다.

• 게시판 글 작성

: 그룹 사용자가 카테고리를 선택해 글과 그림을 올릴 수 있다.



2. 소프트웨어 설계 2.1 게시판 기능

글 목록과 글 작성

from
django.views.generic.list

import Jist Views.generic.edit
import Form View

글 내용 보기

from django.views.generic.detail
import DetailView

이전	<u>년</u> 페이지 다음	- 페이지				글^
	글번호	카테고리	제목	작성자	작성일	Т
	1	잡담	<u>첫번째 잡담</u>	ssn99401	2019년 5월 4일 3:21 오전	
	2	잡담	<u>저녁메뉴는?</u>	ssn99401	2019년 5월 11일 2:06 오진	Car
	3	잡담	<u>운동!!</u>	정진원	2019년 5월 19일 6:04 오진	Coi
	4	잡담	<u>여행!!</u>	정진원	2019년 5월 19일 6:04 오진	
	<u>47</u>					첨부
						이

[잡담] 운동!!

정진원 / 2019년 5월 19일 6:04 오전 하기싫다!!



글쓰기 C: 오늘의식단▼ Title: 저녁 나는 오늘 다이어트를 위해 굶을 것이다! 1004 오전 1004 오전 참부파일: 파일 선택 201600621_...기소개서.docx

파일 선택 20171002_164746.png

2.1 게시판 기능

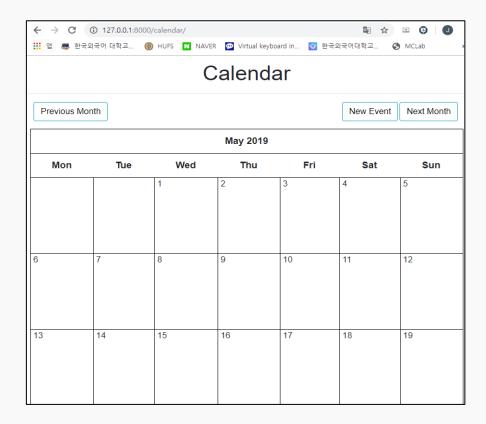




2.2 캘린더 기능

- 캘린더 화면 표시
- : 메인 화면에 게시판과 함께 노출됨

- 캘린더 전/후 달 이동
- : previous month / next month
- 캘린더 이벤트 추가
- : event에는 제목, 내용, 시간 등을 적을 수 있다.



2. 소프트웨어 설계 2.2 캘린더 기능

Calendar

```
|class||Calendar(HTMLCalendar):
     def __init__(self, year=None, month=None):
          self.vear = vear
          self.month = month
          super(Calendar, self).__init__()
|def |prev_month(d):
   first = d.replace(day=1)
   prev_month = first - timedelta(days=1)
 month = 'month=' + str(prev_month.year) + '-' + str(prev_month.month)
   return month
ldef next_month(d):
   days_in_month = calendar.monthrange(d.year, d.month)[1]
   last = d.replace(day=days_in_month)
   next_month = last + timedelta(days=1)
   month = 'month=' + str(next_month.year) + '-' + str(next_month.month)
    return month
class Event(models.Model):
     title = models.CharField(max_length=200)
     description = models.TextField()
      start_time = models.DateTimeField()
```

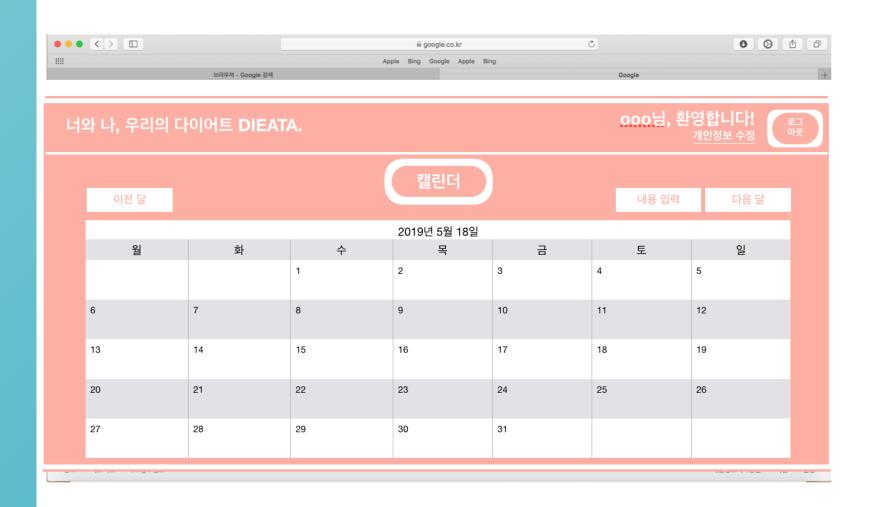
end_time = models.DateTimeField()

Calendar클래스 정의

전,후 달 이동

캘린더에 이벤트 추가

2.2 캘린더 기능





2.3 운동모임개설 기능

• 운동 모임 개설

• 운동 모임 개설 후 출석, 결석 확인가능

운동개설

- 출석 <u>수정삭제</u>
- 결석 <u>수정삭제</u>

투표

운동개설 의 투표 결과

- 1. 출석 1 표
- 2. 결석 0 표

메인으로 돌아가기

2.3 운동모임개설 기능

모임 투표 클래스 정의

```
class Question(models.Model):
    name = models.CharField(max_length=200)
    attend = models.IntegerField(default=0)
    absence = models.IntegerField(default=0)
    pub_date = models.DateField()

def __str__(self):
    return self.name
```

운동 목록 페이지

return render(request, 'vote/index.html',{'a': a})

운동 개설 페이지

return render(request,'vote/detail.html',{'q':q, 'c_list':c_list})

개설된 운동 모임 목록

• <u>운동개설</u> 생성일 : 2019년 5월 13일 <u>수정</u> <u>삭제</u>

운동개설

- 출석 <u>수정삭제</u>
- 결석 <u>수정삭제</u>

투표

운동개설 의 투표 결과

1. 출석 - 1 표

2. 결석 - 0 표

메인으로 돌아가기

2.4 날씨 api 연동 기능

```
|def get_weather_data():
    api_date, api_time = get_api_date()
    url = "http://newsky2.kma.go.kr/service/SecndSrtpdFrcstInfoService2/ForecastSpaceData?"
    key = "serviceKey=" + "DQdkxZVqaYrAOzeZOs3qc93jqj3ubKvUE3Q7hrBQVa8mhXiTgknQTNV%2FPZ2IZxvUuDZy%2FdVeNvIRgkIz%2FgEjiA%3D%3D"
    date = "&base_date=" + api_date
    time = "&base_time=" + api_time
    nx = "&nx=97"
    ny = "&ny=76"
    numOfRows = "&numOfRovs=100"
    type = "&_type=json"
    api_url = url + key + date + time + nx + ny + numOfRows + type
```

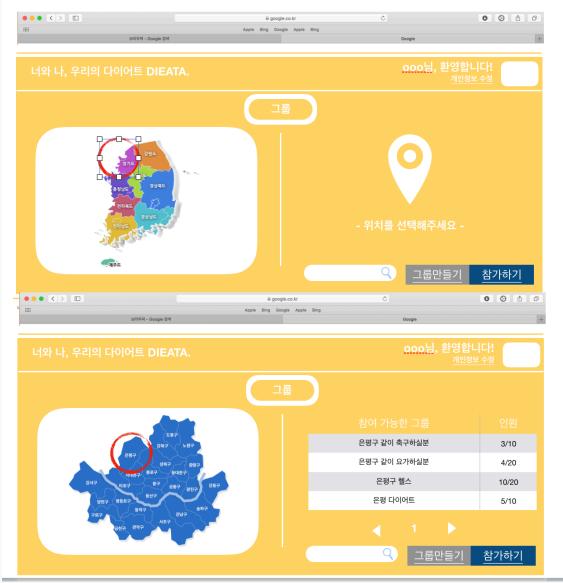
• 강수형태(PTY) 코드: 없음(0), 비(1), 비/눈(2), 눈(3) 여기서 비/눈은 비와 눈이 섞여 오는 것을 의미(진눈개비)

운동 미션 예상 *UI*





그룹 예상 *UI*





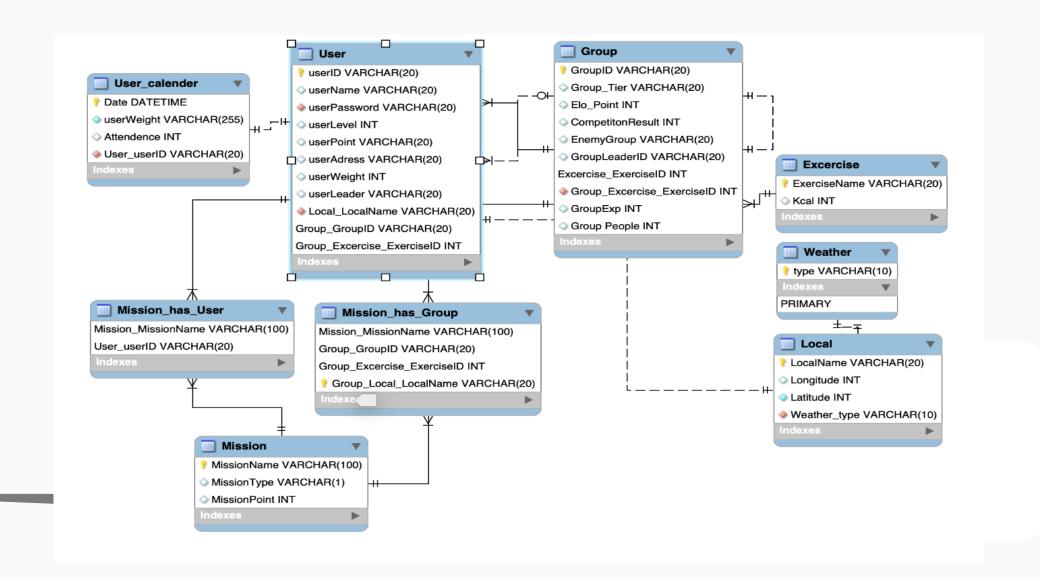
중간결과 발표 및 시연



3. 데이터베이스 설계

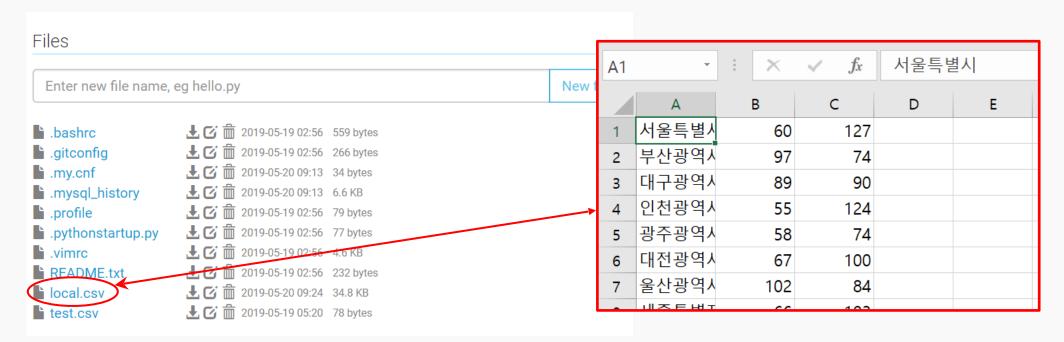
- 데이터베이스 모델링
- 데이터베이스 항목별 데이터 관리

3. 데이터베이스 모델링



3. 데이터베이스 입력

EX) 경도 위도 정보를 저장하는 LOCAL TABLE 만들기 (1)



mysql> DESCRIBE LOCAL;								
Field		Null	Key	Default	Extra			
CITY LA LO	varchar(50) int(11) int(11)	NO NO NO	PRI	NULL NULL NULL				
+++ 3 rows in set (0.00 sec)								

3. 데이터베이스 입력

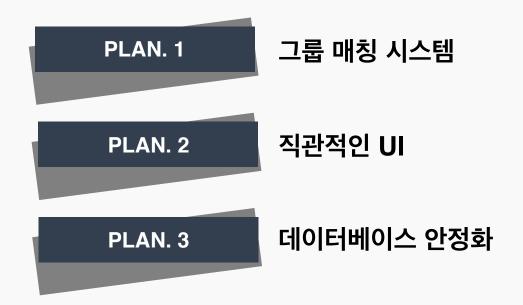
EX) 경도 위도 정보를 저장하는 LOCAL TABLE 만들기 (2)

```
mysql> LOAD DATA LOCAL INFILE '/home/JiyeongYoon/local.csv'
    -> INTO TABLE LOCAL
    -> CHARACTER SET euckr
    -> FIELDS TERMINATED BY ',';
Query OK, 156 rows affected, 14616 warnings (0.15 sec)
Records: 3771 Deleted: 0 Skipped: 3615 Warnings: 14616
```

4. 시연 및 향후계획

http://ssn.pythonanywhere.com

4. 시연 및 향후계획



감사합니다

DIETA

2팀 MAKEU