

Project.SAMUGWAN Technical Manual

최병재

2018년 3월 17일

1 들어가며

안녕하세요! 수습사무관들을 위한 모의 부처지원 플랫폼, Project.SAMUGWAN(프로젝트 사무관)¹을 만든 제작자 최병재라고 합니다.

이 문서를 보시는 분이 누가 될지 잘 모르겠네요. 어쩌면 이 프로젝트는 내년에는 잊혀져서 아무도 이 문서를 찾지 않을지도 모르겠습니다. 하지만 개인적으로는 다음 기수, 다다음 기수 분들이 꾸준히 프로젝트를 업데이트하고 써 주셔서, 나중에 누군가가 부처배치 하는데 도움이 됐다고 말씀해 주시면 정말 고마울 것 같습니다.

각설하고, 이 문서는 다음의 목적을 위하여 만들어졌습니다:

- Project.SAMUGWAN을 구동하기 위한 절차 및 필요한 패키지
- Project.SAMUGWAN의 프로그램 구조 설명
- Project.SAMUGWAN의 한계점 및 개선점
- 기타 잡설

그러므로, 한마디로 말해서 이 문서는 매우 공돌이적인 이야기들로 대부분 채워져 있습니다! 이 프로젝트를 먼지 속에서 꺼내서 다시 돌리실 분이나, 아니면 이 프로젝트를 개선하고 싶어하는 의욕충만하신 분이 아니시면 별로 읽는 걸 추천드리지 않습니다. 그러나, 대체 이 프로젝트가 왜 이렇게 못생겼는지 궁금하시거나, 코드를 왜 이따위로밖에 짜지 못했는지 제작자의 머릿속을 살펴보고 싶으신 분이라면 대환영입니다. 아니면 혹시 제가 풀어놓는 잡설이 궁금하신 분이 있으실지도요... 자, 그러면 시작해 볼까요! (혹시 바로 기술적인 이야기를 보시고 싶으신 분은 섹션 2를 건너뛰어 바로 섹션 3부터 시작해 주세요)

2 제작 목적

행정학의 역사적? 신제도주의를 공부하신 분이라면, 정책이란 것이 갑자기 하늘에서 떨어지는 것이 아니라 기존 정책들의 역사적인 맥락 상에서 만들어진다는 것을 알고 계실 겁니다.² 프로젝트 사무관 또한 같습니다. 작년-그러니까 2017년-의 모의 부처배치는 다음과 같은 엑셀을 활용한 방식으로 이루어졌습니다.

1. Google Form을 이용하여 각 수습사무관들에게 점수 및 지원부처 정보를 얻는다.
2. 엑셀을 이용하여 2차성적-연수원성적 1:1 합산 기준으로 줄을 세운다.
3. 1등부터 노가다로 어떤 부처에 배치될지 계산한다.³

¹ 해당 프로젝트의 소스코드는, 당연히하지만, https://github.com/Salmorunge/project_samugwan에 모두 공개되어 있습니다.

² 제작자는 시험을 친지 2년 반이 넘어가고, 행정학 답지에 부진정-헛소리(나는 헛소리가 아니라고 주장하지만 실제로는 헛소리)를 써냈던 사람입니다. 설명이 틀렸더라도 이해해 주세요.

³ 이건 확실하진 않은데... 엑셀 파일상 어떠한 함수나 매크로를 보지 못한 것으로 미루어보건대 거의 이랬을 겁니다.

	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH
1	일	월	수	목	금	토	일	월	수	목	금	토	일	월	수	목	금	토	일	월	수	목	금	토	일	월	수	목	금	토	일
2	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
4	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
5	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
6	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
7	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
8	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
9	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
10	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
11	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
12	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
13	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
14	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
15	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
16	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
17	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
18	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
19	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
20	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
21	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
22	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
23	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
24	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
25	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
26	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
27	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
28	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
29	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
32	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
33	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
34	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
35	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
36	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
37	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
38	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
39	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
40	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
	부처 정보		개인 성적		TO		배지		관산점수		부처별 지명		Raw Data																		

Figure 1: 2017년 모의 부처배치 방식

4. 일행,재경에 대하여 각각 앞의 과정을 4 5차에 걸쳐 반복한다. 참 쉽죠?

오해하지 말아 주세요. 전 이걸 해낸 앞 기수 분들을 정말로 존경하고 있습니다. 철저한 공익목적동기 (PSM)에 기반하지 않고서는 불가능한 일이에요. 그러나, 필연적으로 해당 방식은 다음과 같은 문제점이 생기게 됩니다:

1. 부처별로 다른 반영비율을 고려하지 못하게 됨
2. 실시간으로 지원부처에 대한 선호 변경을 반영하지 못함
3. 관리자의 엄청난 노가다.
4. 자료의 단절. 2년 전 부처배치 자료는 어딘가로 사라져버려 참조할 수 없게 되었습니다.

Project.SAMUGWAN(Simulator of Application to Ministries Under Government, Web-based Anticipation Navigator⁴)은 이러한 단점들을 보완하기 위해 만들어졌습니다.

Project.SAMUGWAN의 목표는 다음과 같았습니다:

- 부처별 다양한 반영비율 및 가점요인 고려
- 수습사무관들이 시간적, 공간적 제약 없이 실시간으로 지원부처를 변경할 수 있게 함
- 관리자의 노가다 축소 및 일행-재경 통합 데이터베이스 관리
- 해당 년도에만 쓸 것이 아니라 이후 기수들도 쓸 수 있는 플랫폼 구축

이러한 웹사이트를 만드는 것은 저에게도 큰 모험이었습니다. 그 이유는...

- 제가 대학교때 배웠던 것은 계산과학(수치선형대수 *Numerical Linear Algebra* 및 수치해석 *Numerical Analysis*, 그나마도 고시한다고 배우는등 마는등 함)이고 웹 프로그래밍은 해본적도 없으며 마지막 코딩한 게 언젠지 까마득함
- 따라서 코딩을 하려면 많은 시간이 필요할 텐데 그럴 시간이 없음
- 만들어봤자 자치회가 안쓴다거나, 더 근사한 엑셀이 나오면 웹방식을 쓸 하등의 이유가 없음

⁴제가 생각해도 참 이상한 이름입니다. 사실 사무관으로 두문자 만들면 재밌겠다 해서 만들어봤는데... 처음 5자는 어떻게 됐는데 나머지 3자가 너무 어려웠어요...

그런데 정말, 그것이 실제로 일어나 버렸던 것이었습니다. 쓰레기통 모형 아시죠? 프로젝트 사무관을 만들 수 있는 때와 자원과 상황이 우연하게 모여버렸던 것입니다.

- 평창올림픽조직위원회에서 휴일도 없이 일하느라 강제로 코딩할 시간이 생겨버림
- 자치회와 모든 수습사무관이 바빠서 엑셀 자료수집 및 모의배치를 진행할 여력이 없었음. 따라서 자치회가 강제로 프로젝트 사무관을 사용하여야 하는 끔찍한 상황 발생
- Python 기반 web framework인 Django의 존재를 우연히 서핑하다 발견. Python은 그나마 기억이 좀 나는 언어이므로 개발할수 있겠다 생각이 들 (물론 근거없는 자신감이었던 것을 얼마 안되어 뼈저리게 느낌)

어쨌든, 그리하여 전 2018년 2월 8일? 평창동계올림픽이 시작하기 하루 전부터 개발을 시작하여 3월 3일 토요일에 베타 버전을 런칭하고, 3월 5일 월요일에 정식 버전을 런칭하는데 가까스로 성공했습니다. 와, 한달 동안 개발한 것 치고는 정말 완성도 낮군요. 사실 올림픽 하느라 별로 코딩할 시간이 없었다고 변명해 보렵니다. 흑흑.

이제 이 프로젝트를 만든 동기는 대강 썼으니, 본격적인 프로그램 이야기로 넘어가 보죠!

3 프로그램 구동

https://github.com/Salmorunge/project_samugwan의 Project.SAMUGWAN 소스코드를 구동하기 위해서는, 다음과 같은 패키지가 필요합니다:

- Python ver.3.6
- Django ver.2.0.2
- django-registration-redux ver.2.2
- django-tables2 ver. 1.17.1
- Git⁵
- IDE⁶

얼마나 자세하게 이 매뉴얼을 쓸 지 고민을 좀 해 봤는데, 저의 귀찮음과 시간없음이 발동해서 모든 것을 다 쓸 수는 없을 것 같습니다. Django와 Python을 사용할 줄 아신다는 전제 하에서 이 매뉴얼을 기술할게요! 혹시나 Django를 모르시지만 한번 배워보고 싶으신 분은 섹션 3.1을 참조해 주세요. 멍청한 저도 한달만에 어찌어찌 배웠으니 똑똑한 분들은 더 빨리 배우실 수 있으실 거예요!

3.1 Django를 처음 쓰시는 분들께

만약 Django를 예전에 써보신 적이 없고, 어떻게 설치하고 쓰는지 궁금하다 하시는 분은 **Tango with Django** <http://www.tangowithdjango.com/>를 추천해 드립니다. 제가 이 책을 참고하며 대부분의 개발을 진행했고요, Free sample에도 굉장히 많은 내용이 담겨 있습니다!

Django Girls Tutorial <https://tutorial.djangogirls.org/ko/> 또한 추천드립니다. 위의 책보다 더 쉽고 한글 번역본도 있어요!(게다가 계속 업데이트 되고 있습니다. 정말 좋은 프로젝트예요.)

⁵라이브 서버에 Deploy하기 위해서는 Github가 필수입니다.

⁶전 PyCharm Community Edition(무료)를 썼어요. 단 한가지 단점 - field.objects method를 인식 못한다는 것만 빼면 완벽했습니다.

3.2 프로젝트 배치 (Deployment)

웹 프로젝트를 실제 서버 상에 올린다는 것은 굉장히 어려운 일입니다. 전 1시간이면 끝나겠지 했는데 거의 3 4시간 잡아먹더라고요. 흑흑. 저는 Pythonanywhere <https://www.pythonanywhere.com/>을 이용했는데요, 현재 돌아가고 있는 사이트 - <http://salmorunge.pythonanywhere.com/> -를 계속 유지할지, 매년마다 새로이 deploy를 할지는 생각해 보아야 할 점입니다. 어쨌든, 배포에 관한 도움말은 <https://tutorial.djangogirls.org/ko/deploy/>을 참고하시면 좋을 듯 합니다.

어떤 방식을 쓰든, 다음 해의 모의지원을 새로이 시작하기 위해서 데이터베이스는 리셋되어야 합니다. <https://simpleisbetterthancomplex.com/tutorial/2016/07/26/how-to-reset-migrations.html>를 참조하시고, 데이터베이스 리셋 후엔 `runmigration manage.py` 및 `migrate manage.py`를 실행하신 후, `populate_ministry.py`를 실행해 주세요.

4 프로그램 구조

Github 등에서 Project.SAMUGWAN repository를 열어봅시다. Django 프로젝트에서 공동으로 쓰이는 것들은 빼고 특이한 것만 설명하자면 :

- **applying** : Project.SAMUGWAN은 하나의 app - applying만을 사용합니다. 프로그램의 핵심인 model 및 view를 정의하며, 코드 간소화를 위해 빼낸 기타 method들이 존재합니다.
- **populate_ministry.py** : Ministry model 하의 각 ministry object를 일괄적으로 생성합니다. 부처별 반영비율, 일행-재경 쿼터가 이 파일을 통해 입력됩니다. 자세한 것은 코드 주석을 참조해 주세요.

4.1 Model-Template-View 구조

Project.SAMUGWAN중 applying app의 Model-Template-View 구조는 다음과 같습니다.

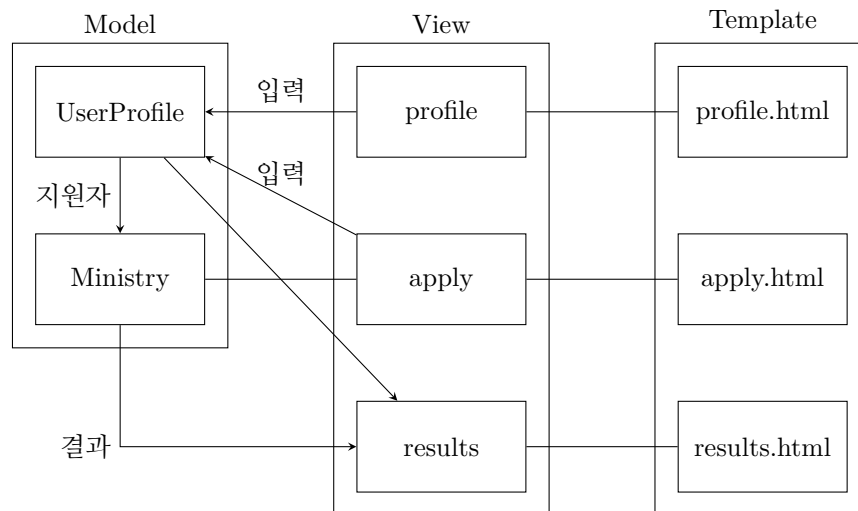


Figure 2: Project.SAMUGWAN Model-View-Template Diagram

각각을 설명하자면 다음과 같습니다. 해당 설명들은 코드의 주석과 상호보완적인 관계입니다. 이 설명을 읽고 주석을 보시면 코드를 더 잘 이해하실 수 있을 겁니다.

- Model

- UserProfile : 회원가입한 유저들의 정보(직렬, 표준점수, 선호부처 등)를 object로 가지는 Model입니다. Django 자체 User model과 1:1 관계(one-to-one relationship)을 가집니다.⁷
- Ministry : 각 부처의 정보(부처 이름, 쿼터, 점수반영비율, 지망한 유저 정보)를 object로 가지는 Model입니다. 직렬(series_of_class)에 따라 다른 object로 취급됩니다. 지망한 유저 정보는, 형식이 엄밀히 정해져있지 않고 JSONField로 슬쩍 남겨둔 다음 View의 apply method에서 형식을 만들어 입력받게 됩니다.

- View

- profile : 각 유저가 자신의 UserProfile 중 성별, 직렬, 표준점수를 변경할 수 있게 하는 view입니다. forms.py에서 UserProfileForm를 이용하여 변경할 정보를 입력받습니다. 그와 동시에 직렬 내 단순합산점수에 기반한 순위를 재계산합니다.
- apply : 각 유저가 자신의 UserProfile 중 선호부처를 변경할 수 있게 합니다. forms.py의 UserApplyingForm을 이용합니다. 선호부처에 따라 전공점수 등의 (2차, 연수원 표준점수를 제외한) 기타점수가 달라지므로 기타점수 또한 새로 입력받습니다. 이를 통해 부처별 반영비율을 이용한 부처별 점수를 산출합니다. UserProfile을 변경한 다음에는 Ministry의 선호부처에 해당하는 object의 applied_samugwan(지망유저 정보)에 이를 반영합니다. 반영되는 정보에는 유저이름, 단순총합점수, 단순총합등수, 지망순위 등이 있습니다.
- results : UserProfile 및 Ministry의 정보를 가공하여 보기 좋게 제공하는 method들의 집합입니다. results라는 하나의 method가 있는 게 아니라고요... 잘 아시듯 유저 목록 일람, 부처별 결과 일람, 모의배정결과 일람 등을 보여주는 method들이 포함됩니다.(result_by_ministry 등.) 그러나 result에서 데이터베이스에 직접 영향을 미치는 항목도 몇 있습니다. 부처점수별 순위 계산은 result에서 이루어집니다. 표로 보여주는 기능은 대개 django-tables2를 이용했습니다.⁸ 사용된 table들의 class는 tables.py에 존재합니다.

- Template : 템플릿입니다. View의 각 method와 1:1 대응이 되며 우리가 볼 수 있는 웹페이지를 생성하는 역할을 수행하기 때문에 별로 설명드릴 것은 없습니다. 스타일링을 위하여 bootstrap <http://bootstrapk.com/>을 이용했습니다.

4.2 기타 기능

applying app이 프로그램의 대부분을 차지하긴 하지만, 그 외 기능들이 있습니다:

- 인증 관련 - Django가 기본적으로 제공하는 로그인 및 인증 기능을 일일이 View 및 Template을 만들어 구현하는 귀찮은 일을 피하기 위해 django-registration-redux를 이용하였습니다. templates 폴더 중 registration은 이에 관련한 템플릿들입니다. password reset 기능도 구현하려고 했으나 잘 안됐어요.

5 한계점 및 개선점

Project.SAMUGWAN에는 여러 한계점들이 존재합니다. 몇 가지는 제가 귀찮아서 해결하지 않은 것도 있고, 몇 가지는 해결방안이 떠오르지 않았던 것도 있습니다. 각각을 써보겠습니다.

1. 유지보수 관련

- (a) 코드가 더러움 : 같은 내용을 중복해서 쓴 코드가 좀 됩니다. 특히 일행-재경에 대해 각각 result를 구해야 할때 그런 부분이 많요. 한 method에 밀어넣고 input parameter가 일행이나 재경이냐에 따라 결과를 다르게 했으면 될걸... 정신차리고 후반엔 좀 바꾸긴 했는데 아직 코드가 더러운 부분이 있어요. 그리고 더미 코드도 좀 있습니다.

⁷다만, 이는 inherit과는 다른 개념이라 UserProfile이 User model의 여러가지 field - username, user_id 등 - 을 가지는 것은 아닙니다. 따라서 UserProfile.objects.get(username='someone')을 해봤자 오류가 납니다. 일단 user를 찾은 다음에 UserProfile.objects.get(user=selected_user)를 하셔야 합니다.

⁸부처별결과관 빼고요. 왜 이런 일이 벌어졌는가는 나중에 한계점에서 써드릴게요.

- (b) 변수명이 더러움 : 부처별 지원자를 나타내는 변수가 user였다가 selected_user였다가 applied_samugwan이었다가 막 왔다갔다 합니다. 보시기에 문제가 없는 수준이면 좋으련만.

2. 구조적 한계 관련

- (a) Ministry Model 구조의 한계 : 코드를 보면 아시겠지만, Ministry Model에서 각 부처에 지원한 지원자들은 list of dictionarys (그나마도 dictionary도 key 하나에 대응되는 list로 이루어져있는 기형적인 구조입니다.)로 applied_samugwan field에 들어갑니다. 통상적인 생각으로는 각 지원자들을 하나의 object로 하는 class를 만들어야겠죠? 그러지 못했습니다. 흑흑.

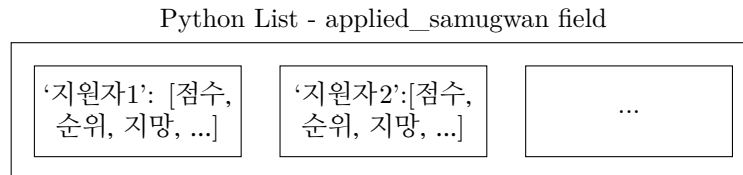


Figure 3: 현재 Ministry 구조

그 이유를 말씀드리자면, 저는 프로그래밍 초보라 데이터를 다룰 때 List나 Dictionary에 iterator를 이용해서 접근하는 게 편했거든요.... 흑흑. 아마 이렇게 바꾸면 좋을 것 같습니다.

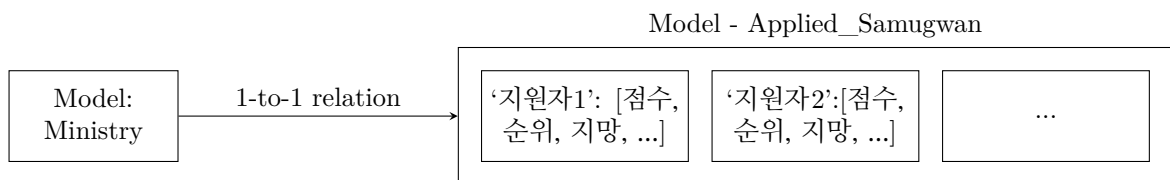


Figure 4: Model을 이용한 개선된 구조

- (b) 유산(Legacy)로써의 이용에 한계 : 제가 앞서 이 프로젝트의 목적 중 하나가, 앞으로도 계속 쓰일 수 있는 시스템을 만드는 것이라고 했는데 현 상태에서는 반쪽의 달성인 것 같습니다. 현재 데이터베이스는 지원 연도별 구분이 되어있지 않아 내년에도 다시 이 프로젝트를 쓰려면 데이터베이스 리셋이 필요합니다. 당연히 올해(2017년도) 데이터에 접근을 할 수 없게 되겠죠. 이를 막으려면 자료구조를 바꾸거나, 현재 사이트를 아카이브해야 할 것 같습니다. 둘 다 일장일단이 있습니다.
- 자료구조 변경 : 같은 사이트에서 편하게 옛날 자료를 검색 가능하지만, 사이트 유지보수 및 최적화에 더 많은 노력이 듭니다.
 - 아카이브 : 사이트 유지보수 및 최적화는 필요없지만 검색 편리성이 떨어집니다. 어쩌면 아카이브 비용이 들수도...
- (c) 모의부처배치의 한계 : 현재 모의부처배치는 단순합산점수 기준으로 이루어집니다. 어떤 부처에 어느 사무관이 갈지 배치하는 구조는 다음과 같습니다 :

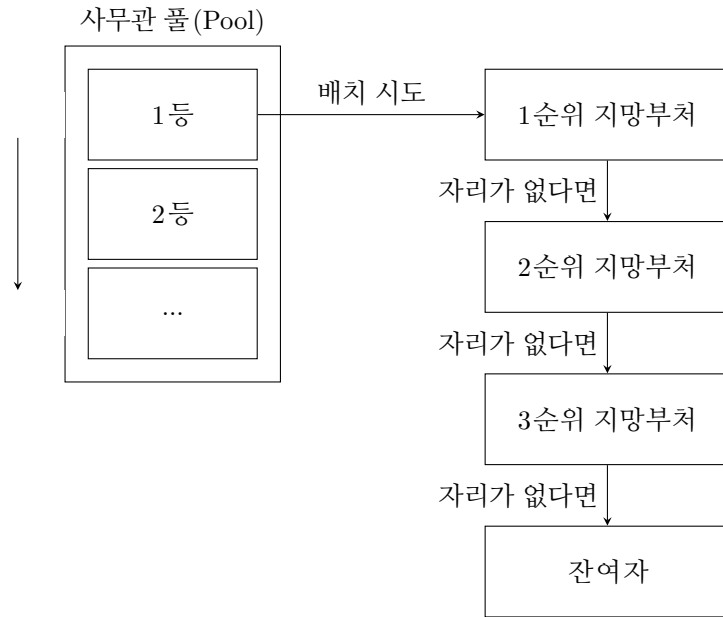


Figure 5: 모의부처배치 구조

모의부처배치를 단순합산점수가 아닌 부처별 점수로 하여 정확한 결과를 내 달라는 요청은 꾸준히 제기되어 왔고, 저도 정말 구현하고 싶었습니다. 그러나 이를 구현하기는 매우 어려웠습니다.

- 위 그림에서 보았듯이, 이 간단한 알고리즘을 계속 쓰려면 사무관 Pool을 Sorting할 하나의 기준이 필요합니다. 그러나 부처별 점수는 각 부처마다 다르고, 각 유저가 지원한 부처도 모두 다르기 때문에 특정 부처점수를 가지고 Sorting할 수가 없습니다. 어떤 사람은 공정위 지원, 어떤 사람은 문체부를 지원했다고 하면 어떤 부처별 점수를 이용해야 할까요?
- 다음으로 생각한 것은 부처별로 쿼터 내의 사람만 뽑는 것이었으나, 지망이 문제가 되었습니다. 해당 부처를 2 3지망으로 쓴 사람이 1지망에 붙을지, 아닐지 어떻게 확인할 수 있을까요?

이 문제를 짧은 시간 안에 해결할 수 없다는 생각이 들자 저는 시간 내에 프로젝트가 돌아가게 하는 것이 더 중요하겠다고 판단하여 해결을 포기하였고, 단순합산점수 기반으로 모의배치가 되었습니다. 이를 해결하는 것은 미래의 후배님들의 몫으로 남겨둘게요.

3. 기능 미구현

- 비밀번호 찾기 미구현 : 비밀번호 e-mail로 리셋 기능을 넣고 싶었습니다만, 테스트 없이 급하게 넣다 보니 작동이 안 된다고 하네요. Django 자체적으로 지원을 한다고 해서 template 만 만들었는데 아마 settings.py에서 설정을 건드려야 할 문제 같습니다.
- 부처별 결과 표 sorting기준 바꿀 수 없음 : django-tables2를 쓰지 못해서 그러합니다. django-tables2에서 sorting 기준을 바꾸기 위해서는 페이지를 새로고침하는데 그 과정에서 지금 어떤 부처를 보고 있는지가 초기화되어서 예러가 뜨더라고요. 이를 막으려면 부처별 결과마다 새로운 페이지를 할당(지금은 result3인가 하나의 페이지에서 form을 이용해서 볼 수 있도록 만들어져 있죠) 하면 될 텐데... 귀찮은 작업이라 못 만들었습니다.

6 마치며

이제 이 문서를 마칠 때가 온 것 같습니다. 좀 더 자세히 써야 할 것 같은 기분을 지울 수가 없으나, 저도 부처면접 준비를 해야 해서 더 이상 시간을 쓸 수가 없군요. 지금 보니 이 프로그램은 너무나 초보적이지만,

능력이 부족한 저로써 이걸 만들기 위해 수많은 밤을 지새웠네요. 이걸 돈 받고 하는 것도 아닌데 그냥 때려치우고 싶은 날들도 많았습니다. 그러나, 다른 동기들이 이 프로젝트 덕분에 그나마 편하게 지원할 부처를 정했다는 말을 들을 때마다 그간의 노력이 보상받는 기분이었습니다. 이 프로젝트를 개량할 분들도 같은 기분을 느끼셨으면 좋겠습니다. 아니면 아예 새로운 프로젝트를 만들어도 좋겠지요. 꼭 제 방식을 계속 따를 필요는 없으니까요. 프로그래밍 계에 오래된 격언이 있죠. There is more than one way to do it!

그러면 저는 수습사무관 여러분을 돕기 위한 하나의 방법을 제시했다는 것에 만족하며 이만 물러가도록 하겠습니다.

2018년 3월 17일

평창조직위 강릉사무소 옆의 카페 프랑에서