

Developer guide

Jingwei gu(N15022473)

Introduction:

This project is implemented with django framework. To make the pure django project run on google app engine, I also use djangoappengine(<http://www.allbuttonspressed.com/projects/djangoappengine>), which provides App engine backends (like database, mail).

The google app engine doesn't originally support several modules in django (like db, auth etc.), it is quite painful to migrate from a pure django project to google app project, so I decided to use third party libs to enable my project on google app engine.

Structure:

The project structure is a typical django project structure:

The project name is test2, which contains the app named surveys.

Test2:

- surveys: the survey main app
- models.py: model layer, which will generate tables in database
- views.py: views layer in django
- urls.py: url router
- admin.py: administrator

templates: all templates for django

- home.html
- base.html
- etc.

below is djangoappengine libraries and tools---

- autoload
- dbindexer
- django
- djangoappengine
- djangotoolbox

Details:

In models:

we have 4 classes, which will be generated as tables in database.

```
class Survey(models.Model):
```

```
    # a survey has many polls, each poll has many choices
```

```
class Poll(models.Model):
```

```
    # each Poll belongs to a survey
```

```
class Choice(models.Model):
    # each choice belongs to a Poll
```

```
class VoteRecord(models.Model):
    # the vote record
```

In views:

Home page view;

```
def home(request):
    # home page
    # show account information
    # show survey list
    # can be redirected to login, register, vote, create
    # @return: response to home.html
    # handle user information
```

Account administration related, which is responsible for login, logout, register:

```
def register_page(request):
    # register a new user
```

```
def register(request):
    # process register information
    # create a user
```

```
def log_me_out(request):
    # logout the system
```

Vote page related, which is responsible for vote in a survey:

```
def show_survey_list(request):
    # show all surveys in a list
    # this view is to show a general view of all surveys
```

```
def show_poll_list(request, survey_id):
    # show all polls in a survey
    # which is implemented in details
    # each poll has several choices, which is contained in p.choice_set
    # @param survey_id: input survey id
    # response:
    # survey: survey object, poll_list: a list of all polls in this survey
    # sth.html: a template show all the questions in a survey
```

```
def vote_for_survey(request, survey_id):
    # vote for a survey
```

Result page related, which is to render show result template:

```
def comment(request, survey_id):
    # show comment of a survey
```

```
def show_results(request, survey_id):
    # show results of a survey
    # @param survey_id: the unique survey id
    # display all the polls of a survey and their results.
```

```
def details(request, survey_id):
    # show details of a survey
```

Survey page related, which is to render survey admin page:

```
def create_a_survey_page(request):
    # render createSurvey.html
```

```
def create_a_survey(request):
    # create a survey
    # create a survey object
    # get input from createSurvey.html
    # return HttpResponseRedirect('create a survey ')
```

```
def delete_a_survey(request, survey_id):
    # delete a survey
```

Poll(question) page related:

```
def create_a_poll_page(request,survey_id):
    # show all polls in this survey
```

```
def create_a_poll(request,survey_id):
    # create a poll in a survey
    # get questions from input
    # redirect to craetPollPage
```

```
def delete_a_poll(request, survey_id, poll_id):
    # delete a poll in a survey
```

In templates:

*_base.html:
base html page for other templates

home.html:
home page

votePage.html:
the vote page, you can vote for a survey in this page

login.html:
login page

register.html:
register page

createSurvey.html:

managing the survey, you can create a survey here

addPolls.html

managing the survey, you can create questions for survey

results.html:

show results of a survey.