

## ■ 이메일 발송

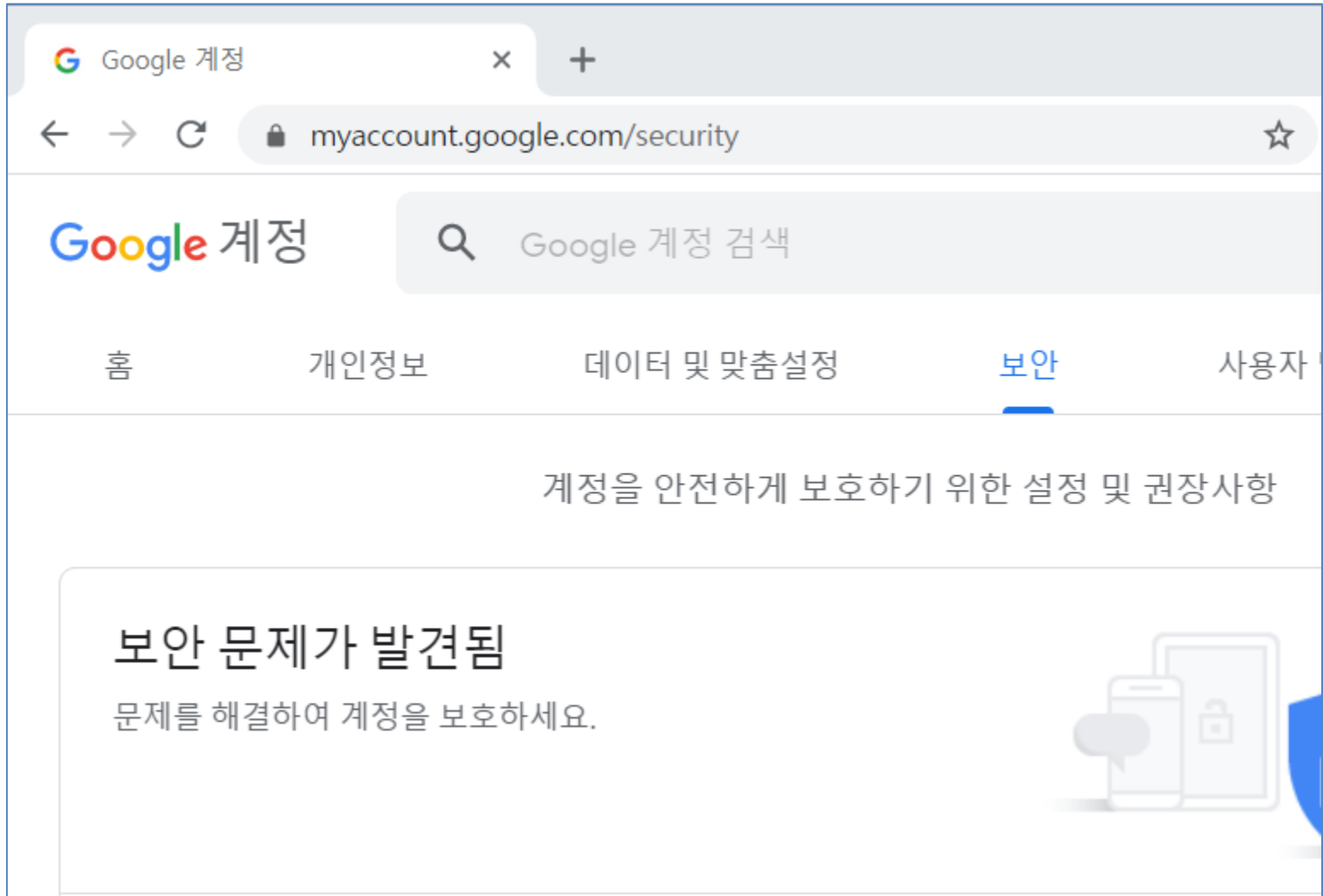
### ● 필요 기술 / 라이브러리

- HTML Design
- Database (필요 시)
- Model
- Django View
- Django Template
- Google SMTP (구글 계정, 메일앱의 비밀번호)

## ■ 이메일 발송

### ● Google 애플리케이션 비밀번호 발급

- <https://myaccount.google.com/security>




## ■ 이메일 발송

- Google 애플리케이션 비밀번호 발급  
- 2단계 인증

홈	개인정보	데이터 및 맞춤설정	보안	사용자 및 공유	결제 및 구독
---	------	------------	----	----------	---------

### Google에 로그인



비밀번호	최종 변경일: 2016. 2. 13.	>
휴대전화로 로그인	⊖ 사용 안함	>
2단계 인증	⊖ 사용 안함	>

## ■ 이메일 발송

- Google 애플리케이션 비밀번호 발급
  - 2단계 인증

### 2단계 인증으로 계정 보호

Google 계정에 로그인할 때마다 비밀번호 및 인증 코드가 필요합니다. [자세히 알아보기](#)



보안을 강화하세요.

비밀번호와 휴대전화로 전송된 고유 인증 코드를 입력하세요.



계정 도용 방지

누군가 내 비밀번호를 알게 되더라도 내 계정에 로그인할 수 없습니다.

계정 ggoreb81@gmail.com ▼

계속하려면 먼저 본인임을 인증하세요.

비밀번호 입력

.....



비밀번호를 잊으셨나요?

다음

시작하기

## ■ 이메일 발송

- Google 애플리케이션 비밀번호 발급
  - 2단계 인증

### 마지막 단계: 백업 옵션 추가

휴대전화를 분실했거나 2단계 인증 수단을 사용할 수 없는 경우, 백업 옵션이 있어야 계정에 로그인할 수 있습니다.



+82 10-8478-8182

Google은 계정 보안을 위해서만 이 번호를 사용합니다.  
Google Voice 번호는 사용하지 마세요.  
메시지 및 데이터 요금이 부과될 수 있습니다.

코드를 어떻게 받으시겠어요?



문자 메시지




전화 통화

[다른 백업 옵션 사용](#)

[보내기](#)

## ■ 이메일 발송

- Google 애플리케이션 비밀번호 발급
  - 2단계 인증



**작동 여부 확인**

Google에서 인증 코드가 포함된 SMS를 방금 **010-8478-8182**번으로 전송했습니다.

코드 입력

받지 못하셨나요? [재전송](#)

[뒤로](#) [다음](#)

## ■ 이메일 발송

- Google 애플리케이션 비밀번호 발급
  - 2단계 인증

2단계 인증을 사용 설정하시겠습니까?

2단계:           Google 메시지 (기본값)  
백업 옵션:     음성 또는 문자 메시지

다음 기기에서 **ggoreb81@gmail.com**의 로그인 상태가 유지됩니다. **Samsung Galaxy Tab4 10.1, LG Nexus 5X**

다른 기기에서 로그아웃될 수 있습니다. 다시 로그인하려면 비밀번호와 2단계가 필요합니다.

사용 설정

## ■ 이메일 발송

### ● Google 애플리케이션 비밀번호 발급

#### - 앱 비밀번호 생성

← 앱 비밀번호

앱 비밀번호를 사용하면 2단계 인증을 지원하지 않는 기기의 앱에서 Google 계정에 로그인할 수 있습니다. 비밀번호를 한 번만 입력하면 기억할 필요가 없습니다. [자세히 알아보기](#)

앱 비밀번호가 없습니다.

앱 비밀번호를 생성할 앱 및 기기를 선택하세요.

앱 선택

▼

기기 선택

▼

생성



## ■ 이메일 발송

### ● Google 애플리케이션 비밀번호 발급

#### - 앱 비밀번호 생성

← 앱 비밀번호

앱 비밀번호를 사용하면 2단계 인증을 지원하지 않는 기기의 앱에서 Google 계정에 로그인할 수 있습니다. 비밀번호를 한 번만 입력하면 기억할 필요가 없습니다. [자세히 알아보기](#)

앱 비밀번호가 없습니다.

앱 비밀번호를 생성할 앱 및 기기를 선택하세요.

메일 ▼ Windows 컴퓨터 ▼

생성

## ■ 이메일 발송

### ● Google 애플리케이션 비밀번호 발급

#### - 앱 비밀번호 생성

생성된 앱 비밀번호

Windows 컴퓨터용 앱 비밀번호

tzno ylf i kmqz phhh

사용 방법

1. '메일' 앱을 엽니다.
2. '설정' 메뉴를 엽니다.
3. '계정'을 선택한 뒤 내 Google 계정을 선택합니다.
4. 비밀번호를 위에 표시된 16자리 비밀번호로 교체합니다.

일반적인 비밀번호와 마찬가지로 이 앱 비밀번호는 Google 계정에 대한 완전한 액세스 권한을 부여합니다. 비밀번호를 기억하지 않아도 되므로 적어 놓거나 다른 사용자와 공유하지 마세요.

[자세히 알아보기](#)

[확인](#)

Add your Google account

Enter the information below to connect to your Google account.

Email address

securesally@gmail.com

Password

••••••••

☐ Include your Google contacts and calendars

## ■ 이메일 발송

### ● Google 애플리케이션 비밀번호 발급

#### - 앱 비밀번호 생성

앱 비밀번호

이름	생성됨	최종 사용일
Windows 컴퓨터의 메일	오전 11:09	-

앱 비밀번호를 생성할 앱 및 기기를 선택하세요.

앱 선택

기기 선택

생성

## ■ 이메일 발송

### ● templates/contact.html

```
{% extends 'base.html' %}
{% block content %}

<div class="jumbotron">
  <div class="container text-center">
    <form method="post" action="/contact/">
      {% csrf_token %}
      <div class="form-group">
        <label for="email">Email:</label>
        <input type="text" class="form-control" name="email">
      </div>
      <div class="form-group">
        <label for="comment">Comment:</label>
        <textarea class="form-control" rows="5" name="comment"></textarea>
      </div>
      <button type="submit" id="send" class="btn btn-primary btn-block">
        발송
      </button>
    </form>
  </div>
</div>

{% endblock %}
```

## ■ 이메일 발송

### ● templates/contact\_success.html

```
{% extends 'base.html' %}
{% block content %}

<div class="jumbotron">
  <div class="container text-center">
    <h1>메일 발송이 완료되었습니다.</h1>
  </div>
</div>

{% endblock %}
```

## ■ 이메일 발송

- board/views.py (이메일 발송 함수 추가)

```
def contact(request):
    if request.method == 'POST':
        email = request.POST.get('email')
        comment = request.POST.get('comment')
        # 발신자주소, 수신자주소, 메시지
        send_mail('구글계정이메일', email, comment)
        return render(request, 'contact_success.html')

    return render(request, 'contact.html')

import smtplib
from email.mime.text import MIMEText
def send_mail(from_email, to_email, msg):
    smtp = smtplib.SMTP_SSL('smtp.gmail.com', 465) # SMTP 설정
    smtp.login(from_email, '앱비밀번호16자리') # 인증정보 설정
    msg = MIMEText(msg)
    msg['Subject'] = '[문의사항]' + to_email # 제목
    msg['To'] = from_email # 수신 이메일
    smtp.sendmail(from_email, from_email, msg.as_string())
    smtp.quit()
```

## ■ 이메일 발송

● <http://localhost:8000/contact/>

A screenshot of a web browser showing a contact form. The browser's address bar displays 'localhost:8000/contact/'. The page has a dark header with the word 'Portfolio' and a hamburger menu icon. The form contains an 'Email:' field with the text 'seorab@naver.com', a 'Comment:' text area with the text '외주 개발 단가 문의' and '연락처 010-8478-8181', and a blue '발송' (Send) button. At the bottom, there is a 'Footer Text' section.

