

# Testing Email Campaigns with Your Personal Email (Then Logout)

---

## Goal

Use your personal Gmail temporarily for testing, then completely remove it from the system.

---

## SAFE TESTING WORKFLOW

Step 1: Quick Gmail Setup (5 minutes)

### **1.1 Generate App Password (Safer than using real password)**

#### **1. Go to Google Account**

- Visit: <https://myaccount.google.com/security>
- Sign in with your Gmail

#### **2. Enable 2-Step Verification (if not already enabled)**

- Click "2-Step Verification"
- Follow setup wizard
- This is REQUIRED for App Passwords

#### **3. Create App Password (This is temporary and can be revoked anytime)**

- Go back to Security page
- Scroll to "App passwords" section
- Click "App passwords"
- Select App: "Mail"
- Select Device: "Other" → Type: "MakePlus Test"
- Click "Generate"
- **Copy the 16-character password** (example: `abcd efg hijk lmnop`)
- Keep this window open (you'll need it in Step 3)

---

Step 2: Configure for Testing

### **2.1 Create/Edit .env File**

Open or create `.env` file in `E:\makeplus\makeplus_backend\`:

```
# Temporary Gmail Configuration for Testing
EMAIL_HOST=smtp.gmail.com
EMAIL_PORT=587
EMAIL_USE_TLS=True
```

```
EMAIL_HOST_USER=your-email@gmail.com
EMAIL_HOST_PASSWORD=abcdefghijklmnopqrstuvwxyz
DEFAULT_FROM_EMAIL=your-email@gmail.com
SITE_URL=http://127.0.0.1:8000

# Other settings (if not already present)
DEBUG=True
SECRET_KEY=your-secret-key-here
ALLOWED_HOSTS=localhost,127.0.0.1
```

### Replace:

- `your-email@gmail.com` → Your actual Gmail address
- `abcdefghijklmnopqrstuvwxyz` → Your 16-char App Password (remove spaces)

## 2.2 Start Server

```
cd E:\makeplus\makeplus_backend\makeplus_api
..\venv\Scripts\python.exe manage.py runserver
```

## Step 3: Test Campaign System (10 minutes)

### 3.1 Quick Test with Test Script

```
cd E:\makeplus\makeplus_backend
.\venv\Scripts\python.exe test_email_setup.py
```

#### What it will do:

1. Check your email configuration
2. Ask for test email (use your own email to receive test)
3. Send test email
4. Optionally test campaign creation

### 3.2 Manual Test via Django Shell

```
cd E:\makeplus\makeplus_backend\makeplus_api
..\venv\Scripts\python.exe manage.py shell
```

Then run:

```

from django.core.mail import send_mail

# Send test email to yourself
send_mail(
    subject='Test Email from MakePlus',
    message='Testing email delivery',
    from_email='your-email@gmail.com',
    recipient_list=[ 'your-email@gmail.com' ],
    fail_silently=False,
)

```

**Expected Result:** You'll see 1 (success) and receive email in your inbox

### 3.3 Test Full Campaign

```

from dashboard.models_email import EmailCampaign, EmailRecipient
from dashboard.utils_campaign import send_campaign_email

# Create test campaign
campaign = EmailCampaign.objects.create(
    name="Test Campaign",
    subject="Testing My Campaign System",
    body_html="""
        

# Test Email



This is a test campaign.



Test Link


    """,
    from_email="your-email@gmail.com",
    trackOpens=True,
    trackClicks=True,
    status='draft'
)

# Add yourself as recipient
recipient = EmailRecipient.objects.create(
    campaign=campaign,
    email="your-email@gmail.com",
    name="Test User"
)

# Send it
send_campaign_email(recipient)

print(f"✅ Campaign sent! Check your email.")

```

```
print(f"Tracking URL:  
http://127.0.0.1:8000/track/email/open/{recipient.tracking_token}/")
```

### 3.4 Test Tracking

1. **Check your email inbox**
  2. **Open the test email** (this will track an "open")
  3. **Click the link** (this will track a "click")
  4. **View stats in admin:** <http://127.0.0.1:8000/admin/>
    - Go to "Email campaigns"
    - Click your test campaign
    - See open count and click count
  5. **View detailed stats:** <http://127.0.0.1:8000/dashboard/campaigns/1/stats/>
- 

## Step 4: Complete Testing Checklist

Test all features:

- Email sends successfully
  - Email received in inbox
  - Open tracking works (visit tracking URL)
  - Click tracking works (click links in email)
  - Stats show in admin panel
  - Detailed stats page works
  - All tabs show correct data
- 

## Step 5: LOGOUT & DISCONNECT (Important!)

### 5.1 Revoke App Password in Gmail

1. **Go to Google Account**
  - Visit: <https://myaccount.google.com/security>
2. **Click "App passwords"**
3. **Find "MakePlus Test"** in the list
4. **Click the X or Remove button** next to it
5. **Confirm removal**

**Your App Password is now revoked** - Django can no longer send emails using your Gmail

### 5.2 Remove Email Credentials from .env

Open **.env** file and **comment out or delete** the email lines:

---

```
# Email Configuration - REMOVED AFTER TESTING
# EMAIL_HOST=smtp.gmail.com
# EMAIL_PORT=587
# EMAIL_USE_TLS=True
# EMAIL_HOST_USER=your-email@gmail.com
# EMAIL_HOST_PASSWORD=abcdefghijklmnopqrstuvwxyz
# DEFAULT_FROM_EMAIL=your-email@gmail.com

# Or set to console backend for testing
EMAIL_BACKEND=django.core.mail.backends.console.EmailBackend

SITE_URL=http://127.0.0.1:8000
```

**Alternative:** Use console backend (emails print to terminal instead):

```
# Email prints to console (no real sending)
EMAIL_BACKEND=django.core.mail.backends.console.EmailBackend
SITE_URL=http://127.0.0.1:8000
```

### 5.3 Clear Campaign Test Data (Optional)

If you want to remove test campaigns:

```
# In Django shell
from dashboard.models_email import EmailCampaign

# Delete all test campaigns
EmailCampaign.objects.filter(name__icontains="test").delete()

# Or delete all campaigns
EmailCampaign.objects.all().delete()
```

### 5.4 Restart Server

```
# Stop server (Ctrl+C in terminal)
# Start again
cd E:\makeplus\makeplus_backend\makeplus_api
..\venv\Scripts\python.exe manage.py runserver
```

---

Step 6: Verify Disconnection

#### 6.1 Test That Email No Longer Sends

---

```
cd E:\makeplus\makeplus_backend\makeplus_api
..\venv\Scripts\python.exe manage.py shell
```

```
from django.core.mail import send_mail

# This should fail or print to console
send_mail(
    'Test',
    'This should not send',
    'test@example.com',
    ['test@example.com'],
    fail_silently=False,
)
```

### Expected Results:

- If using console backend: Email prints to terminal (not actually sent)
- If credentials removed: Error occurs (this is good - means it can't send)

---

## 🔒 SECURITY BEST PRACTICES

### Why This Method is Safe:

#### 1. **App Password** - Not your real Gmail password

- Can be revoked anytime
- Limited to email sending only
- Doesn't give access to your account

#### 2. **Temporary** - Easy to disconnect

- Remove from .env file
- Revoke in Google settings
- No trace left

#### 3. **Testing Only** - Not exposed

- Only on your local machine
- .env file should be in .gitignore
- Never committed to repository

### Additional Security:

- Make sure .env is in .gitignore
- Never commit .env to git
- Delete test campaign data when done

- Revoke App Password when finished
- 

## ⌚ RECOMMENDED: Switch to Professional Service Later

After testing with your personal email, switch to a professional service:

For Production:

- **Brevo (Sendinblue)**: 300 emails/day FREE forever
- **SendGrid**: 100 emails/day FREE forever
- No personal email needed
- Better deliverability
- Professional appearance

See [EMAIL\\_SETUP\\_GUIDE.md](#) for setup instructions.

---

## 📋 QUICK REFERENCE

Testing Commands:

```
# Run test script
cd E:\makeplus\makeplus_backend
.\venv\Scripts\python.exe test_email_setup.py

# Django shell
cd E:\makeplus\makeplus_backend\makeplus_api
..\venv\Scripts\python.exe manage.py shell

# Start server
cd E:\makeplus\makeplus_backend\makeplus_api
..\venv\Scripts\python.exe manage.py runserver
```

Logout Steps:

1. Revoke App Password in Gmail: <https://myaccount.google.com/security>
  2. Remove credentials from `.env` file
  3. Restart Django server
  4. Delete test data (optional)
- 

## ⚠️ TROUBLESHOOTING

"`SMTPAuthenticationError`" after revoking

**This is expected!** It means disconnection worked.

Solution: Remove email config from `.env` or use console backend.

---

## Emails still in inbox

- This is normal** - Already sent emails stay in your inbox.

Solution: Just delete them manually if you want.

## Test data still in database

- This doesn't affect security** - Just campaign records, no email credentials.

Solution: Delete via admin panel or Django shell if desired.

---

## SUMMARY

### What You'll Do:

1.  Generate temporary Gmail App Password (5 min)
2.  Add to .env file (1 min)
3.  Test campaign system (10 min)
4.  Revoke App Password in Gmail (1 min)
5.  Remove from .env (1 min)
6.  Restart server (1 min)

### What This Achieves:

- Full campaign testing
- Open/click tracking verified
- Stats system tested
- No security risk
- Easy to disconnect
- Ready for production service

### Total Time:

**~20 minutes** for complete test + logout

---

## AFTER TESTING

### When ready for real campaigns:

1. Sign up for Brevo/SendGrid (10 min)
2. Get professional SMTP credentials
3. Update .env with new credentials
4. Send real campaigns with confidence!

**Status:** Ready to test safely! 