

# Email Reminder System

## Enhancements & Deployment (Deadline – Week 9)

**Additional Features:** Add recurring reminders and custom notification settings.

**UI/UX Improvements:** Enhance layout, add icons, and make responsive for all devices.

**API Enhancements:** Improve data validation, add better error messages, and faster response.

**Performance & Security Checks:** Optimize speed, perform vulnerability scans, and use secure APIs.

**Testing of Enhancements:** Conduct functional and integration testing.

**Deployment:** Host app on Netlify, Vercel, or Cloud platforms for access and scaling.

### ***Sample Python Program: Email Reminder Simulation***

```
# Email Reminder System Simulation (Enhancement Example)
import datetime
import time

# Function to simulate sending an email reminder
def send_reminder(email, message, reminder_time):
    print(f"Reminder set for {email} at {reminder_time.strftime('%H:%M:%S')}")
    while datetime.datetime.now() < reminder_time:
        time.sleep(1)
    print(f"\n■ Email sent to {email}: {message}")

# Input details
email = "user@example.com"
message = "Meeting with team at 3 PM."
reminder_time = datetime.datetime.now() + datetime.timedelta(seconds=5)

# Call function
send_reminder(email, message, reminder_time)
```

### ***Expected Output:***

```
Reminder set for user@example.com at 14:25:30
```

```
■ Email sent to user@example.com: Meeting with team at 3 PM.
```