GitHub Webhook Event Handler Documentation

1. Project Overview

This is a Flask-based web application designed to receive, process, and store real-time events from a GitHub repository. It specifically tracks Push, Pull Request, and Merge events. The data is stored in a MongoDB Atlas database and served to a frontend UI via REST APIs.

Key Features:
- Webhook Listener: Listens for POST requests from GitHub.
- Event Processing: Distinguishes between commits, PR creations, and PR merges.
- Data Persistence: Stores structured event data in MongoDB.
- REST API: Provides endpoints for fetching event logs and statistics.

2. Technical Stack

- Language: Python 3.x
- Framework: Flask
- Database: MongoDB (using pymongo)
- Utilities: python-dateutil

3. Installation & Setup

Prerequisites:
Ensure Python is installed. Install dependencies:

```
pip install flask pymongo python-dateutil
```

Configuration:
- DB Name: WebHook
- Collection: events

Run the app:

```
python app.py
```

4. API Reference

A. Webhook Receiver (POST /webhook)

Handles logic to parse GitHub JSON payloads.
- Push: Logs committer and commit ID.
- Pull Request (Opened): Logs source/target branches.
- Merge (Closed & Merged): Logs merge timestamp.

B. Event Data API (GET /api/events)

Returns list of all stored events sorted chronologically.

```
[
  {
    'action': 'PUSH',
```

```
    'author': 'neteshdev',
    'request_id': 'a1b2c3d4...',
    'timestamp': '2023-10-27...'
  }
]
```

C. Stats API (GET /api/stats)

Returns aggregate counts of events.

```
{
  'total': 150,
  'push': 120,
  'pull_request': 20,
  'merge': 10
}
```

5. Database Schema

Documents in the 'events' collection follow this structure:

```
{
  'request_id': 'String',
  'author': 'String',
  'action': 'PUSH | PULL_REQUEST | MERGE',
  'from_branch': 'String',
  'to_branch': 'String',
  'timestamp': 'String'
}
```

6. GitHub Configuration

1. Go to Repository Settings > Webhooks.
2. Payload URL: http://<server-ip>:5000/webhook
3. Content type: application/json
4. Trigger: Push events and Pull requests.

7. Security Note

CRITICAL: The current code contains MongoDB credentials directly in the source file. Move these to an environment variable (.env) before deployment.