

**Github Repo link:** <https://github.com/Shubhashis-Roy/newsletter>

### **User Authentication & CRUD Operations:**

- Register API
  - Endpoint: api/auth/register
  - Method: POST
  - payload: {

```
        "email": "user@example.com",
        "password": "Password123"
    }
```
- Login API
  - Endpoint: api/auth/login
  - Method: POST
  - payload: {

```
        "email": "user@example.com",
        "password": "Password123"
    }
```
- User Get API:
  - Endpoint: api/user/profile/:id
  - Headers (Required):
  - Authorization: Bearer <your\_jwt\_token>
  - Method: GET
- User update API:
  - Endpoint: api/user/profile/:id
  - Authorization: Bearer <your\_jwt\_token>
  - Method: PATCH
  - payload: {

```
        "fullname": "user@example.com",
    }
```
- User Delete API:
  - Endpoint: api/user/profile/:id
  - Authorization: Bearer <your\_jwt\_token>
  - Method: DELETE

**# RSS Campaigns:** It should automatically generate campaigns based on new entries in specified RSS feeds, streamlining content distribution:

How it works:

1. User creates a normal campaign.
2. User attaches an RSS feed using the PATCH API.
3. A cron job runs every 5 minutes to:
  - Fetch the RSS feed
  - Detect the newest article
  - Compare with lastFetchedItem
  - Auto-create a new campaign if the article is new
  - Update lastFetchedItem

**APIs:**

- Attach RSS Feed to Campaign API:
  - Endpoint: /api/campaigns/:id/rss-feed
  - Method: PATCH
  - payload: {  
        "rssFeed": "https://techcrunch.com/feed/"  
    }

Cron Job Logic Location: src/campaigns/[campaigns.service.ts](#)

```
// Cron Job code :  
@Cron(CronExpression.EVERY_5_MINUTES)  
async autoGenerateFromRSS() { ... }
```

**# Click Statistics Tracking:** Tracks clicks and opens for every campaign link to measure subscriber engagement.

**# Features**

1. Stores link clicks in Postgres using the ClickStat entity
2. Tracks:
  - Click count

- Open tracking via a 1x1 pixel GIF
  - Click-through redirects
3. Prevents unauthorized DB updates via service-layer validation
  4. Uses IP + User-Agent + Device Info for analytics

### # Entities Involved

- 1.click\_stats table (TypeORM)
2. links table
- 3.campaign\_links table (Knex)

### APIs:

- Attach RSS Feed to Campaign API:
  - Endpoint: /api/click-stats/track/<campaignId>?link=https%3A%2F%2Fgoogle.com
  - Method: GET

### # How to Test

1. Create campaign
2. Attach RSS feed
3. Wait for cron (5 mins) or temporarily change to 1 min
4. Observe console logs

### Logs:

*Checking RSS feeds for new articles...*

*Auto-created campaign from RSS: <Article Title>*