# 📦 Release Notes – AI Newsletter Automation Project

📅 Release Date: June 26, 2025  
🔁 Version: v1.0  
✍️ Author: Devesh Ruttala

## 1. Overview

Automated newsletter system that scrapes the web for AI-related content (especially in Speech-to-Text domain), summarizes the content using OpenAI and Gemini APIs, stores it in Supabase, and emails summaries to subscribers using Mailjet.

## 2. What's New

- End-to-end automation using Python, Supabase, OpenAI/Gemini, Mailjet, and Docker

- Real-time AI news ingestion from 5+ sources using RSS, APIs, and Selenium

- On-demand trigger-based execution with n8n workflows

## 3. Improvements

- Manual ChromeDriver installation reduced Docker build size

- Environment variables structured via `.env` for secure configuration

## 4. Bug Fixes

- RSS malformed entry issue partially handled with try-except

- Email delivery fallback for recipients with invalid addresses

## 5. Breaking Changes / Known Issues

- Known Issue: RSS feeds may skip malformed entries

- Known Issue: No deduplication across scraping sources

- Known Issue: Rate limiting from Reddit and TechRadar

## 6. Upgrade Instructions

1. Pull the latest code from GitHub

2. Update `.env` with your latest keys and recipients

3. Run `docker-compose up --build` to start the updated service

## 7. Deprecations

- Deprecated use of dashboard UI temporarily until Streamlit admin panel is added

## 8. Support

Contact: deveshruttala@gmail.com or raise an issue on GitHub:  
https://github.com/deveshruttala/Remoat-GTM-Assginment

## ✅ What Shipped

• End-to-end automation for AI newsletter system

• Web scraping from 5+ sources using RSS, BeautifulSoup, and Selenium

• AI-powered summarization using Gemini (Google) and OpenAI (GPT-3.5)

• Supabase integration for storing scraped stories

• Mailjet email integration for scalable email delivery

• Dockerized setup for reproducible deployment

• n8n workflow for scheduled + trigger-based automation

• Environment variable config via .env file

## 🧱 Architecture Overview

• Language: Python 3.11

• Web scraping: feedparser, requests, BeautifulSoup, selenium

• AI summarization: Google Gemini API & OpenAI GPT-3.5

• DB: Supabase (PostgreSQL backend)

• Mail service: Mailjet (API v3.1)

• Automation: n8n + Docker Compose

• Schedule: Adjustable using INTERVAL\_MINUTES in .env

## 🛠️ Key Technical Decisions

• Chose Supabase over local PostgreSQL for easier deployment

• Used Gemini Flash for lower latency summaries

• Manual ChromeDriver installation avoided inside Docker to reduce build time

• Used .env to isolate secrets and API keys

• Separated scraping sources into categories for better parsing (rss, web, selenium)

## 📎 Appendix

### Performance Benchmarks

- 10+ newsletter editions generated and delivered

- 30–50 articles scanned per run

- 3–5 second summary generation latency via Gemini

- Mailjet open rates and engagement tracked (basic analytics)

### Future Enhancements

- Content deduplication

- Threaded story categorization

- Enhanced analytics dashboard

- Subscription interface via Mailchimp or TinyLetter