

5-Day Workflow Plan (MERN + Atlas + Predefined Model)

Day 1 – Environment Setup & Cloud Database

- Set up **MERN stack project structure** with separate folders for React frontend and Express backend.
 - Create a **MongoDB Atlas cluster** and configure connection string in the backend.
 - Define schema in Atlas (via Mongoose) to store:
 - Raw posts
 - Processed text
 - Urgency/intent classification results
 - Extracted details.
 - Document the **overall architecture**: React (UI) ↔ Express (APIs) ↔ Atlas (Database).
-

Day 2 – Data Acquisition & Storage in Atlas

- Design backend API to **collect social media posts** (using predefined dataset or simulated posts).
 - Store collected posts into **MongoDB Atlas collections**.
 - Verify Atlas entries directly in the cloud dashboard.
 - Ensure secure database access with environment variables (Atlas connection string in .env).
-

Day 3 – Preprocessing & NLP Model Integration

- Backend performs **text preprocessing** (remove links, emojis, noise).
 - Send cleaned text to a **predefined NLP model API** (e.g., HuggingFace Inference API).
 - Receive classification results such as:
 - *Help Request / Not Help Request*
 - *High / Low / Needs Review* urgency.
 - Store both processed text and classification results in **Atlas**.
-

Day 4 – Information Extraction & Structured Storage

- Use **NER (Named Entity Recognition)** through a predefined model API to extract:
 - Names, contacts, locations, type of help.

- Use **regex rules** to identify phone numbers and emails.
 - Save extracted structured details into Atlas in JSON format (nested inside each post document).
 - Prepare API endpoint to serve this structured data for frontend use.
-

Day 5 – Dashboard Development (React Frontend)

- Develop a **React dashboard** that fetches processed posts from backend APIs.
 - Display information in a clear format:
 - Raw Text
 - Processed Text
 - Urgency Classification
 - Extracted Details.
 - Add **filter options** (e.g., view only High Priority posts).
 - Showcase **end-to-end flow**: Post entry → Atlas → NLP classification → Extracted details → Dashboard visualization.
-

✓ With Atlas integrated, your project achieves **50% completion** in 5 days, covering:

- Cloud DB setup
 - API design
 - Data storage in Atlas
 - NLP integration for urgency classification
 - React dashboard for visualization.
-