

15-Day Roadmap to Build AI-Powered Fake News & Misinformation Filter

Total Available Time: 75 Hours

Primary Focus: WhatsApp Integration





 **Tech Stack:** FastAPI (Backend), Twilio API (WhatsApp), Mistral/Hugging Face (AI), MongoDB

Phase 1: Setup & Core AI Model (Day 1-5) 25Hours25 Hours

- ♦ **Goal:** Build the AI model & API to detect fake news.

Day 1: Setup & WhatsApp Integration (5 Hours)




- ♦ **Tasks:**


-  Set up FastAPI in Python.
-  Register for Twilio WhatsApp API.
-  Create a simple WhatsApp bot that **receives messages & responds**.
-  Deploy a test bot using Ngrok for local testing.

 **Checkpoint:** Your bot should **reply on WhatsApp** with a test message.

Day 2-3: AI Model - Fake News Detection (10 Hours)




- ♦ **Tasks:**

-  Use **Mistral AI API** or **Hugging Face pre-trained model** to analyze messages.
-  If using **Hugging Face**, test models like **Fake-News-BERT**.
-  Write a FastAPI endpoint `/analyze_message` that:
 - Receives text from WhatsApp
 - Calls AI model
 - Returns "Fake" or "Real"

 **Checkpoint:** Your API should classify text correctly.

Day 4-5: Action System & MongoDB Storage (10 Hours)

- ♦ **Tasks:**

-  Create actions:
 - If **Fake News** → **Send warning to user**
 - If **Real News** → **Do nothing**
 -  Store analyzed messages in **MongoDB** (for logging).
 -  Add a `/flag_message` API to flag fake messages manually.

 **Checkpoint:** Your bot **detects fake news & warns users in WhatsApp**.



Phase 2: Enhancements & Dashboard (Day 6-10)

25Hours25 Hours

- ♦ **Goal:** Improve AI accuracy & build a simple dashboard.

✓ Day 6-7: AI Accuracy & Explainability (10 Hours)

- ♦ **Tasks:**

- ✓ Add **fact-checking sources** (Google Fact Check API).
- ✓ Make AI **explain why** a message is fake (clickbait score, source check).
- ✓ Improve **accuracy** by fine-tuning the model (if needed).

📌 **Checkpoint:** AI **gives reasons** when flagging fake news.

✓ Day 8-10: Basic Dashboard for Monitoring (15 Hours)

- ♦ **Tasks:**

- ✓ Build a **basic web dashboard** using **HTML, CSS, and FastAPI templates**.
- ✓ Show flagged messages & allow admin **to delete them**.
- ✓ Add a **search bar** to find past messages.

📌 **Checkpoint:** You have a **basic web UI** to monitor flagged messages.



Phase 3: Testing & Deployment (Day 11-15)

25Hours25 Hours

- ♦ **Goal:** Ensure stability, deploy, & prepare for the hackathon.

✓ Day 11-12: Testing & Fixing Bugs (10 Hours)

- ♦ **Tasks:**

- ✓ Test AI with **real WhatsApp messages**.
- ✓ Fix **false positives & negatives**.
- ✓ Ensure **MongoDB correctly stores flagged messages**.

📌 **Checkpoint:** Your bot **accurately detects fake news with minimal errors**.

✓ Day 13-14: Deployment & Final Tweaks (10 Hours)

- ♦ **Tasks:**

- ✓ Deploy FastAPI backend to **Render / Railway / AWS**.
- ✓ Set up **Twilio API webhook for production**.
- ✓ Optimize response time.

📌 **Checkpoint:** **Live bot running 24/7 on WhatsApp**.

✓ Day 15: Hackathon Preparation (5 Hours)

◆ Tasks:

- ✓ Create a **short pitch** & prepare a demo.
- ✓ Make a simple **presentation** for judges.
- ✓ Ensure all **APIs are stable & working**.

🔒 **Final Checkpoint:** You are **ready to present!** 🚀

🎯 Summary

Phase	Focus	Tasks	Duration
Phase 1	Core AI & WhatsApp	AI Model, WhatsApp Bot, MongoDB	25 Hours
Phase 2	Accuracy & Dashboard	AI Improvements, Simple Web UI	25 Hours
Phase 3	Testing & Deployment	Bug Fixes, Hosting, Presentation	25 Hours

🔥 Next Steps

- 1 Start with **FastAPI & WhatsApp integration today**.
- 2 Use **Mistral or Hugging Face API** for AI.
- 3 Let me know **when you finish each phase** so I can guide you further.

This plan ensures you finish **before the hackathon** and have a working **WhatsApp bot & dashboard!** 🚀