

ThreatIQ – Phishing Detection Chatbot

Summary Report of Accomplished Tasks:

This report provides an overview of the tasks completed for the ThreatIQ project and indicates where the corresponding work can be found in the submission.

1. Backend Development

- **Authentication System**
 - Implemented JWT-based authentication with user roles (admin, user).
 - Location: app/routers/auth.py, app/models.py.
- **Analysis Pipeline**
 - Built the /analyze/ endpoint to process user inputs and detect phishing vs. safe messages.
 - Integrated multiple heuristic checks and AI-powered results.
 - Location: app/routers/analyze.py, app/heuristics.py.
- **Data Persistence**
 - Implemented SQLAlchemy models for User and Analysis.
 - Analysis history stored with timestamps, linked to users.
 - Location: app/models.py, app/database.py.
- **Statistics API**
 - Created /stats/judgments endpoint for pie chart data (Safe vs Phishing vs Other).
 - Provides aggregated analysis counts for frontend charts.
 - Location: app/routers/stats.py.
- **Seeding Script**
 - Added seed_data.py to generate dummy analyses for testing dashboard charts.
 - Location: app/seed_data.py.

2. Frontend Development

- **Authentication & Routing**
 - Next.js pages for login, dashboard, analyze, history, settings.
 - Location: frontend/src/pages/*.
- **Analyze Page (Chat Interface)**
 - Built a conversational interface for analyzing user text.
 - Includes structured bot responses (judgment, explanation, tips).
 - Bug fixes: input clears after send, “Analyzing...” message added.
 - Location: frontend/src/app/analyze/page.tsx.
- **Dashboard**
 - Integrated charts for activity insights:
 - **Line Chart:** analyses per day.
 - **Bar Chart:** monthly analyses.
 - **Pie Chart:** judgment breakdown.
 - Location: frontend/src/app/dashboard/page.tsx,
frontend/src/components/charts/*.

3. Testing & Debugging

- Verified seeded data connects to the correct user (ID=2).
- Debugged pie chart showing only “Other” → fixed parsing logic in backend & frontend.
- Resolved frontend input persistence bug.
- Confirmed endpoints respond correctly (/analyze/, /analyze/history, /stats/judgments).

4. Documentation

- Added README.md with project overview, features, and usage instructions.
- Created detailed deliverables (Parts A–F) explaining design decisions.
- Prepared GenAI-log.docx documenting where generative AI was used (less than 30% of total work).

Conclusion

All major project requirements have been met:

- Authentication
- Analysis pipeline
- Dashboard with charts
- Seeding and statistics
- Frontend interaction

The work is fully functional and tested, with clear documentation for review.