MICHIGAN STATE UNIVERSITY

Project Plan Presentation

Packet Forge: AI Network Protocol Engine

The Capstone Experience

Team Vectra AI

Samuel Barnhart
Nihar Bollareddy
Sean Finkel
Yeji Lee
Kaajal Shah
Aanshik Upadhyay

Department of Computer Science and Engineering
Michigan State University

Fall 2025

From Students…
…to Professionals

Project Sponsor Overview

- Cybersecurity company headquartered in San Jose, CA
- Leader in AI-driven threat detection and response solutions
- Founded in 2011, with AI/ML at the core since day one
- Serves 1,000+ organizations worldwide
- Client point of contact: Brad Woodberg and Campbell Robertson

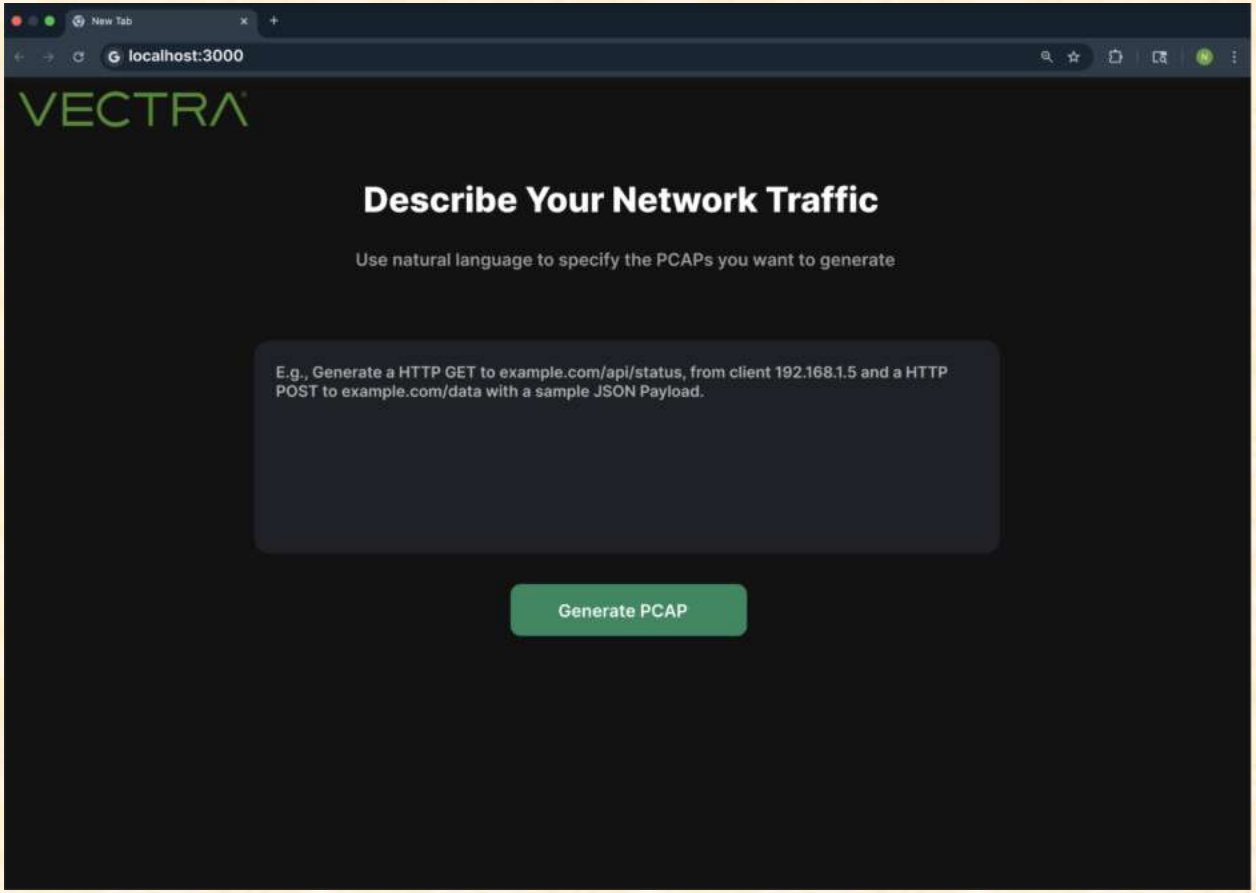
Project Functional Specifications

- Current Issue: PCAPS created manually is too time consuming
- Solution: Automated PCAP Generator from:
  - RFC Specifications
  - Natural Language Prompts
- Benefits
  - Eliminates manual authoring
  - Enables diverse, standards-compliant network traffic

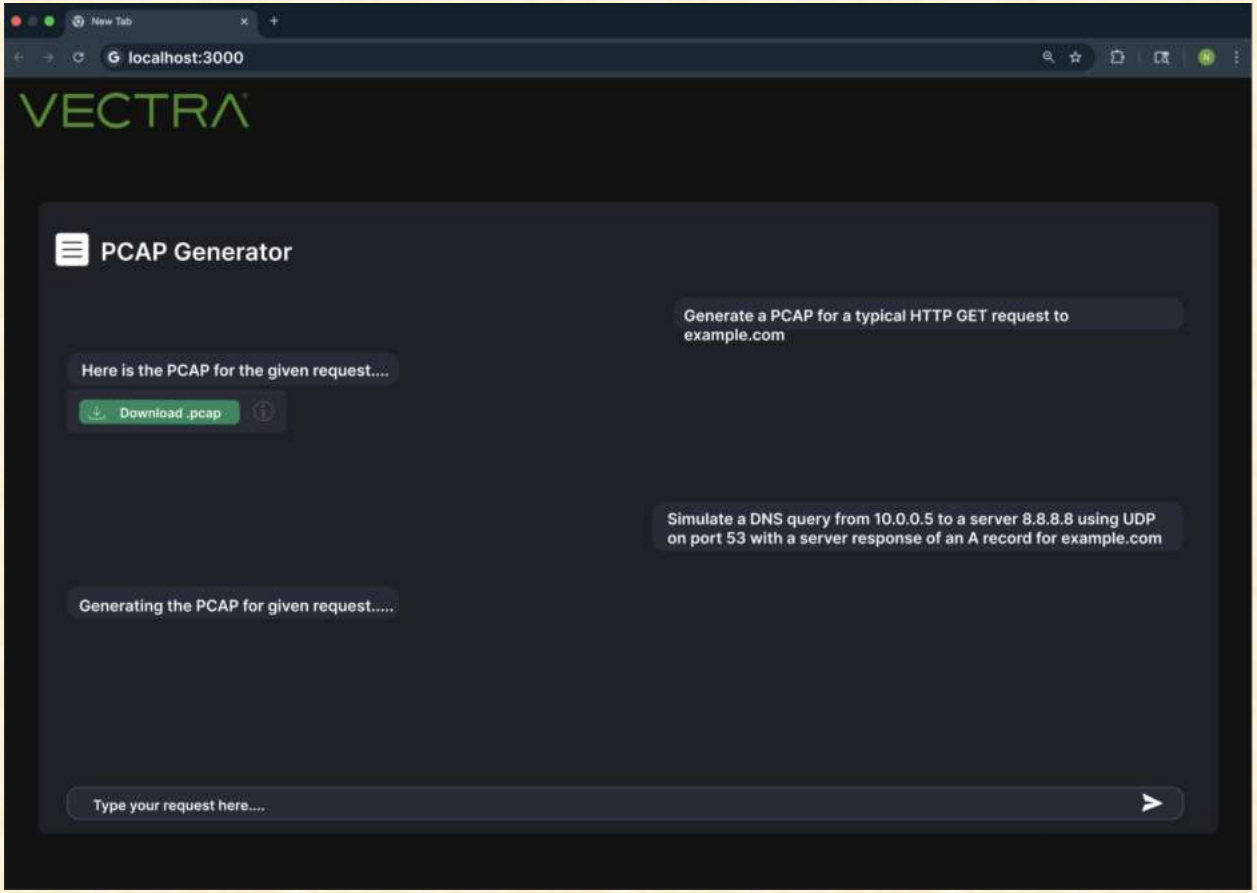
Project Design Specifications

- Web base application with simple UI
- Uses RAG (Retrieval – Augmented Generation)
  - RFCs indexed on ChromaBD
- Process:
  - Users enter natural language prompt
  - Retrieve relevant RFCs from database
  - LLM produces structured JSON
  - PCAP builder synthesizes output
  - Validated via wireshark

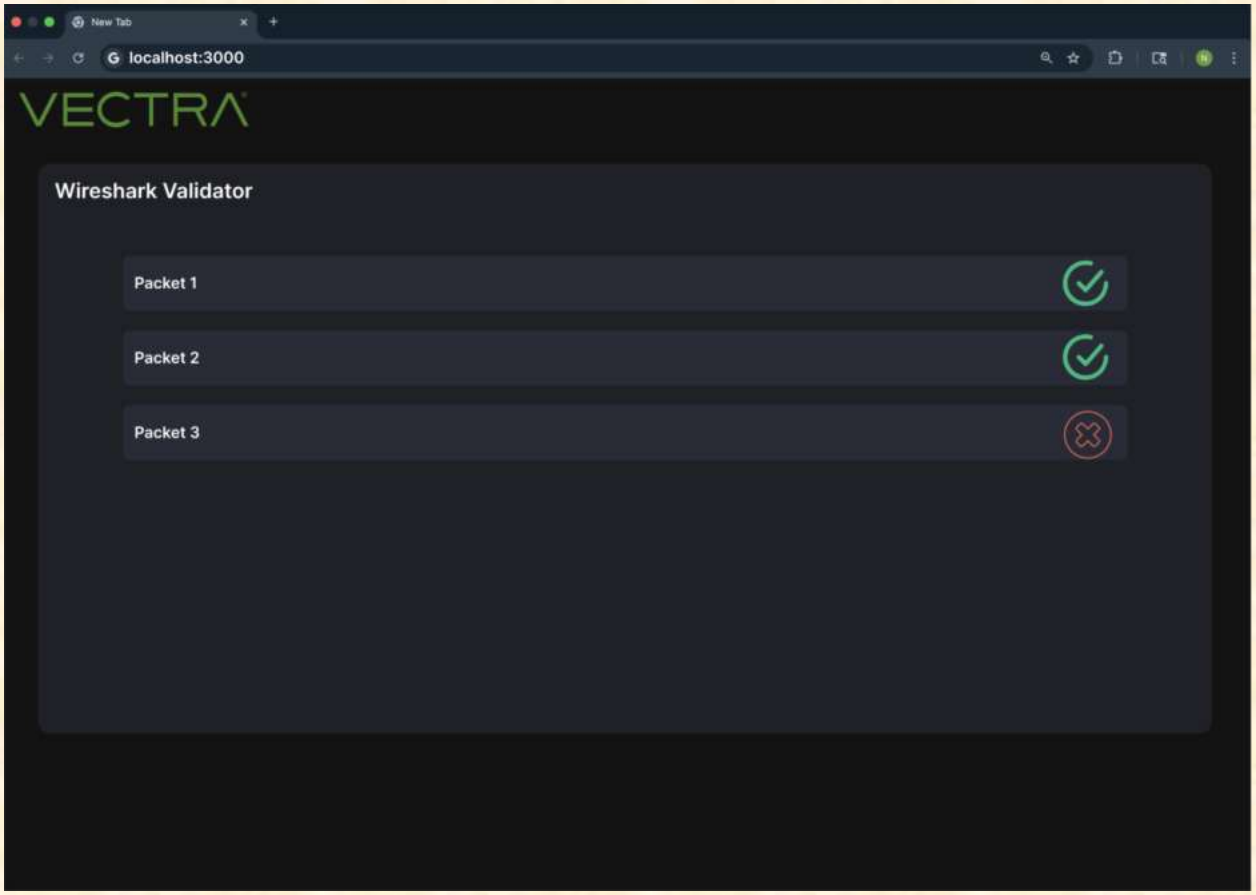
Screen Mockup: Landing Page



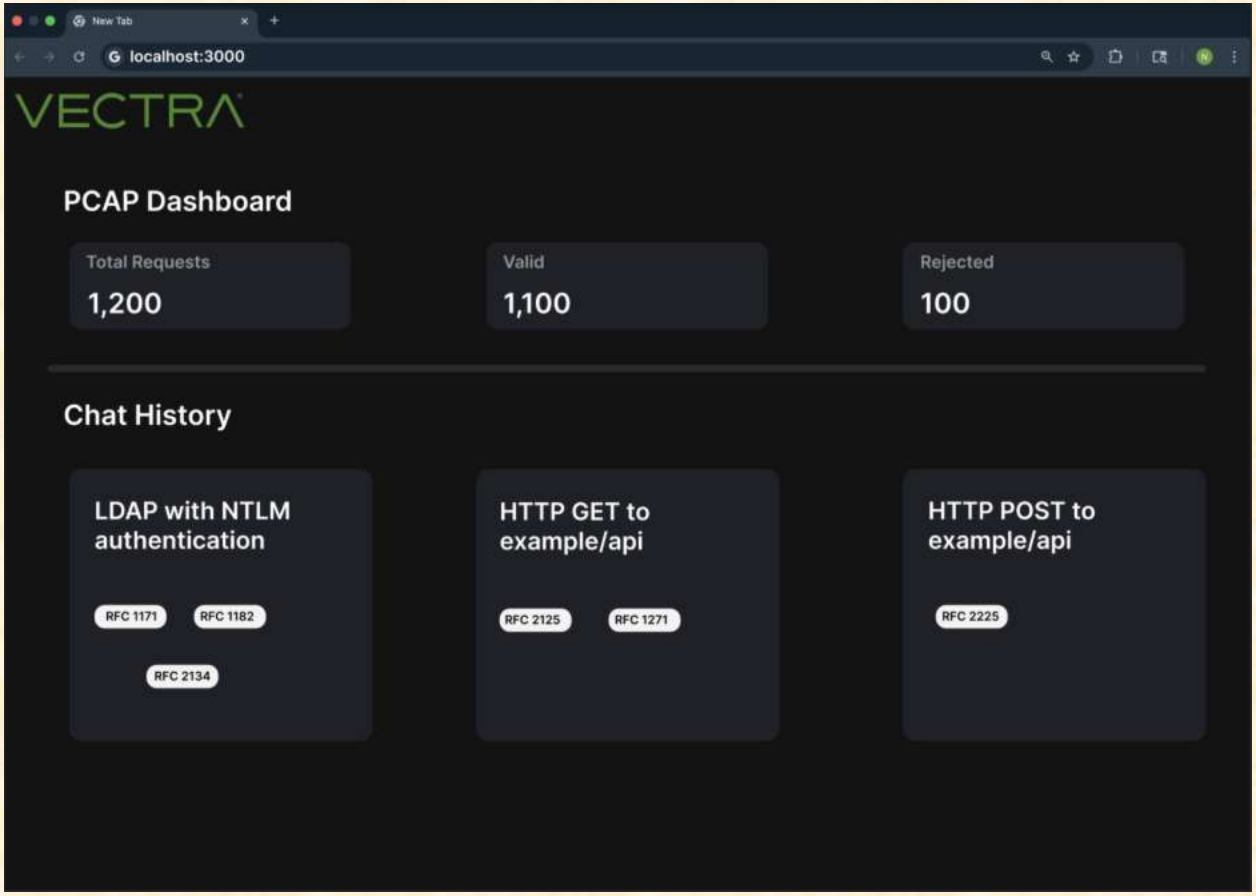
Screen Mockup: Request Chat



Screen Mockup: Wireshark Validation



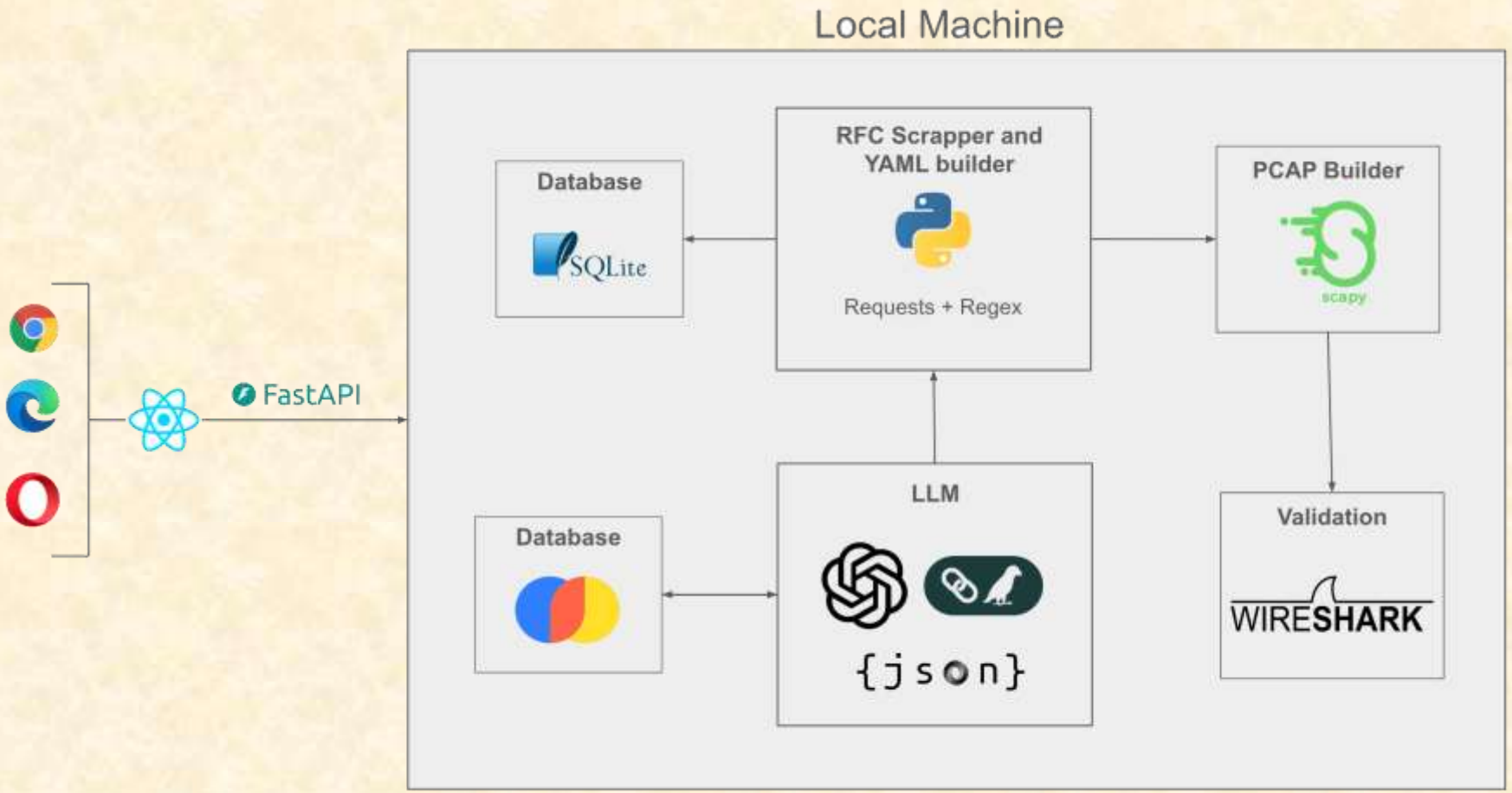
Screen Mockup: Request History



Project Technical Specifications

- Architecture Layers
  - Protocol Knowledge base (YAMAL, RFC parsing)
  - PCAP Generation Engine (Scapy, dpkt)
  - Validation and UI layer (Pyshark, Wireshark, Flask)
- Tech Stack
  - Python3
  - SpaCy(NLP for prompt parsing)
  - PyYAML (protocol rule storage)
  - Docker (containerized for portability)

Project System Architecture



Project System Components

- Hardware Platforms
  - Team Devlopment laptops
  - Dockerized Enviroments for Consistency
- Software Platforms / Technologies
  - Python, Flask, VScode
  - Scapy, dpkt, PyShark, SpaCY, PyYAML
  - Docker, Wireshark
  - ChromaDB (vector DB), SQlite (metadata DB)

Project Risks

- Filtering the Correct RFCs
  - Discovery may include stale or irrelevant RFCs, which will reduce the quality of PCAP
  - Mitigation: Follow the update / obsoletes chains, always favor newer standards, and verify with Wireshark
- Getting Valid PCAPs from LLMs
  - LLMs can output / generate unparseable or protocol breaking packets from PCAPs
  - Mitigation: Use well-formed prompts, freeze a stable version of an LLM to avoid instability, and check all outputs with tshark
- Accurate Extraction of Data from RFCs
  - The format of RFCs may lead to incomplete reads of the protocol through missed or misread data.
  - Mitigation: Build table / ABNF parsers as modules and test whole RFCs end to end
- Scraping and Storing Protocols in Database
  - Scraping large numbers of RFCs can result in rate limits, duplicates, or storage issues.
  - Mitigation: Batch scrape with retries, store in JSONL with unique IDs, and use checkpoint / resume

Questions?