COLLEGE CODE: 3105

COLLEGE NAME: Dhanalakshmi Srinivasan of college of engineering and technology

DEPARTMENT: computer science and engineering

STUDENT NM-ID: F2BC6CD21FA3002BF1F4B722A116B804

ROLL NO: 310523104045

DATE:09/05/2025

Completed the project named as,

Structural health monitoring system

SUBMITTED BY, Your Name and team member names.

R.GOKUL

A.DANIEL

N.ANBARASAN

M.DHINESH KARTHICK

S.EDISON

Phase-5: Project Demonstration & Documentation

Title: AI-Powered Structural Health Monitoring System

Abstract:

The AI-Powered Structural Health Monitoring (SHM) project aims to revolutionize infrastructure maintenance by leveraging artificial intelligence, signal processing, and IoT (Internet of Things) technologies. In its final phase, the system integrates advanced AI models to detect structural anomalies, real-time data collection from sensors (like accelerometers and strain gauges), and secure data management, while ensuring scalability and seamless integration with infrastructure management systems. This document provides a comprehensive report of the project's completion, covering the system demonstration, technical documentation, performance metrics, source code, and testing reports. The project is designed to handle large-scale monitoring with robust data security measures, providing accurate structural health insights in real time. Screenshots, sensor diagrams, and codebase snapshots will be included for a full understanding of the system's architecture and functionality.

Index (with page numbers suggested)

---

1. Project Demonstration

Overview:

The AI-Powered SHM system will be demonstrated to stakeholders, showcasing its features, performance improvements, and functionality. This demonstration highlights the system's real-time anomaly detection, IoT sensor data integration, security measures, and performance scalability.

Demonstration Details:

- **System Walkthrough:** A live walkthrough of the monitoring platform, from sensor data acquisition to output of structural integrity assessments, showcasing the system's response to structural vibrations and stress variations.
- **AI Detection Accuracy:** The demonstration will show how the AI model detects structural damage (e.g., cracks, frequency shifts) based on real-time sensor data.
- **IoT Integration:** Real-time metrics like vibration frequency, strain, and tilt angle collected from IoT-enabled sensors will be displayed and analyzed.
- **Performance Metrics:** Response time, system scalability, and load handling under multiple sensor feeds will be highlighted to show improved monitoring capacity.
- **Security & Privacy:** Encryption protocols and data protection measures will be explained and demonstrated as the system handles sensitive structural data.

Outcome:

By the end of the demonstration, the system's ability to monitor real-world infrastructure conditions, ensure data security, and deliver actionable insights through IoT integration will be effectively showcased to stakeholders.

---

2. Project Documentation

Overview:

Comprehensive documentation for the AI-Powered SHM system is provided to detail every aspect of the project. This includes system architecture, AI model details, code explanations, and usage guidelines for both engineers and administrators.

Documentation Sections:

- **System Architecture:** Diagrams illustrating the complete monitoring system, including AI algorithms, signal workflows, and IoT sensor integrations.
- **Code Documentation:** Source code and explanations for all modules, including AI damage classification, data acquisition interfaces, and anomaly detection logic.
- **User Guide:** A manual for operators explaining how to interpret monitoring data, configure sensors, and respond to damage alerts.
- **Administrator Guide:** Instructions for system maintenance, data validation, and performance testing procedures.
- **Testing Reports:** Detailed reports on detection accuracy, system responsiveness, and data security evaluations.

Outcome:

All critical components of the monitoring system will be well-documented, providing a clear guide for future expansion, deployment, or infrastructure scaling.

---

3. Feedback and Final Adjustments

Overview:

Feedback from the project demonstration will be collected from engineers, infrastructure managers, and a broader group of test users. This feedback will be used to make final refinements before project handover.

Steps:

- **Feedback Collection:** Feedback from mentors, field engineers, and test users will be gathered via surveys and observation during the system walkthrough.
- **Refinement:** Based on the feedback, any data inconsistencies, AI misclassifications, or interface usability issues will be addressed.
- **Final Testing:** After refinements, the system will undergo final tests to ensure accuracy, stability, and field-readiness.

Outcome:

Final adjustments will optimize the system for broader real-world deployment across bridges, buildings, or industrial structures.

---

4. Final Project Report Submission

Overview:

The final project report provides a comprehensive summary of all phases, key achievements, challenges faced, and outcomes of the AI-Powered SHM project. This report will include detection results, performance benchmarks, and future recommendations.

Report Sections:

- **Executive Summary:** A concise overview of the SHM project, outlining its goals, scope, and major achievements.
- **Phase Breakdown:** A detailed breakdown of each project phase, covering AI model development, sensor calibration, data integration, and security implementation.
- **Challenges & Solutions:** A section documenting the major challenges encountered (e.g., noisy data, real-time synchronization) and how they were resolved.
- **Outcomes:** A summary of the system’s effectiveness and readiness for continuous infrastructure monitoring.

Outcome:

A detailed report will be submitted, covering the full lifecycle of the SHM project — from concept to final testing and validation.

---

5. Project Handover and Future Works

Overview:

Final handover includes documentation, source code, and deployment packages. Recommendations for future improvements and scaling are also provided.

Handover Details:

- **Next Steps:** Suggestions for future developments, such as integrating more sensor types (e.g., ultrasonic, thermal), expanding AI models, and supporting multilingual dashboards.

Outcome:

The AI-Powered Structural Health Monitoring System will be officially handed over, complete with tools, guides, and future roadmaps to support infrastructure resilience and safe

Source code 1:

```
import numpy as np
from scipy.fft import fft

def get_dominant_frequency(signal, sampling_rate):
    N = len(signal)
    yf = fft(signal)
    xf = np.linspace(0.0, sampling_rate / 2.0, N // 2)
    amplitude = 2.0 / N * np.abs(yf[0:N // 2])
    return xf[np.argmax(amplitude)]

def detect_damage(dominant_freq, threshold=47):
    return dominant_freq < threshold
```

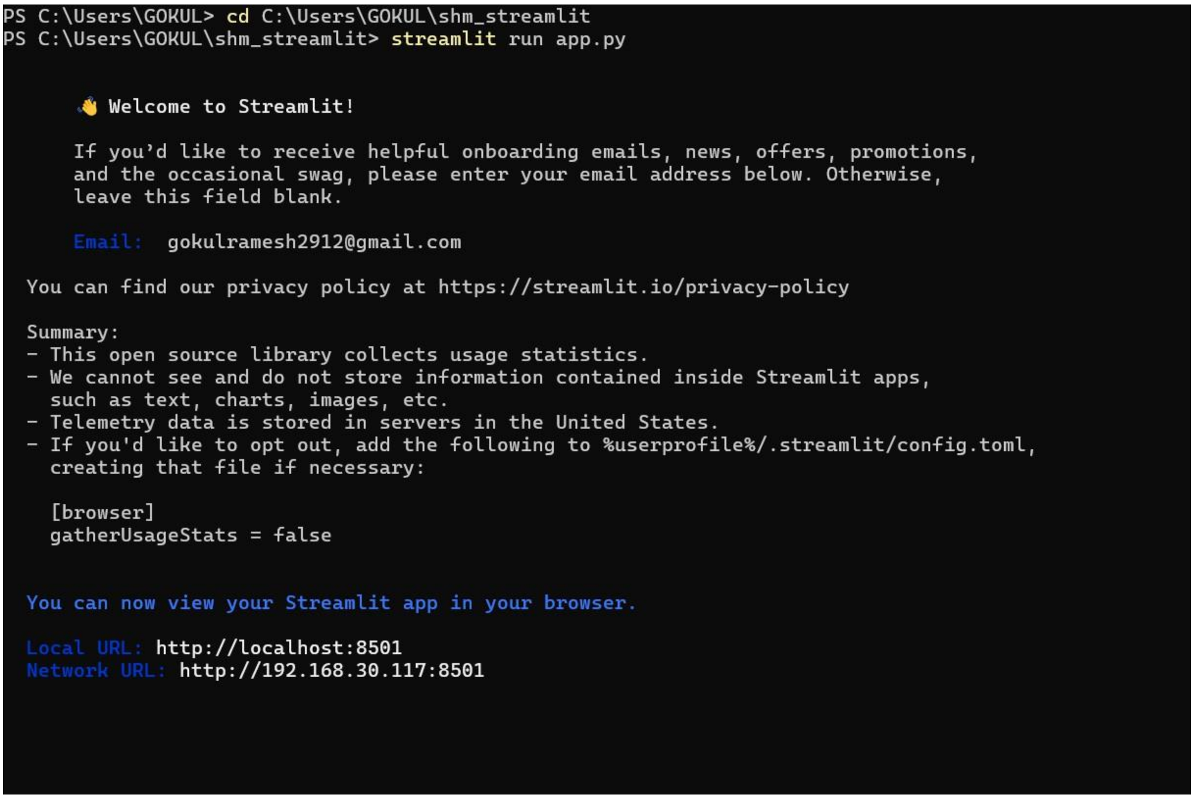
source code 2:

```
import streamlit as st
from simulate_data import simulate_signal
from analyze_signal import get_dominant_frequency, detect_damage
import matplotlib.pyplot as plt

st.set_page_config(page_title="Bridge Health Monitor", layout="centered")
st.title("🛠 Structural Health Monitoring")
st.write("Simulated bridge vibration analysis using FFT")

damaged = st.checkbox("Simulate Damaged Structure", value=False)
t, signal = simulate_signal(damaged=damaged)
freq = get_dominant_frequency(signal, 1000)
status = "⚠ Damaged" if detect_damage(freq) else "✅ Healthy"

st.metric("Dominant Frequency (Hz)", f"{freq:.2f}")
st.success(f"Status: {status}" if status == "✅ Healthy" else f"Warning: {status}")
```

```
    fig, ax = plt.subplots()
    ax.plot(t, signal, label="Vibration")
    ax.set_xlabel("Time (s)")
    ax.set_ylabel("Amplitude")
    ax.set_title("Vibration Signal")
    st.pyplot(fig)
```

source code 3:

```
import numpy as np

def simulate_signal(damaged=False, sampling_rate=1000, duration=2):
    t = np.linspace(0, duration, int(sampling_rate * duration))
    freq = 50 if not damaged else 45
    signal = np.sin(2 * np.pi * freq * t)
    noise = 0.5 * np.random.randn(len(t))
    return t, signal + noise
```



```
PS C:\Users\GOKUL> cd C:\Users\GOKUL\shm_streamlit
PS C:\Users\GOKUL\shm_streamlit> streamlit run app.py

      👋 Welcome to Streamlit!

      If you’d like to receive helpful onboarding emails, news, offers, promotions,
      and the occasional swag, please enter your email address below. Otherwise,
      leave this field blank.

      Email:  gokulramesh2912@gmail.com

  You can find our privacy policy at https://streamlit.io/privacy-policy

  Summary:
  - This open source library collects usage statistics.
  - We cannot see and do not store information contained inside Streamlit apps,
    such as text, charts, images, etc.
  - Telemetry data is stored in servers in the United States.
  - If you'd like to opt out, add the following to %userprofile%/.streamlit/config.toml,
    creating that file if necessary:

    [browser]
    gatherUsageStats = false


  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.30.117:8501
```

localhost:8501

🛠️ Structural Health Monitoring

Simulated bridge vibration analysis using FFT

☐ Simulate Damaged Structure

Dominant Frequency (Hz)

50.05

Status: ✅ Healthy