Introduction to Data Management and Processing

DS5110 [Section-04]

Iteration 02

Aryan Rakeshbhai Shah
002337094

Atir Shakhrelia
002369722

Jwal Pranavkumar Shah
002369936

1 Project Kickoff

The following sections discuss about the specific goals, project scope, deliverables, milestones, deadlines and team capablities for this project.

1.1 Goals

Goal 1: Build a data integration pipeline to consolidate sensor data from various IoT devices into a standardized format.
Goal 2: Perform comprehensive data analysis and create a visualization dashboard for actionable insights.
Goal 3: Apply data transformation techniques to clean the dataset, making it suitable for deep learning and machine learning models, particularly for fault detection.
Goal 4: Develop a robust database system to store data extracted from online platforms on local systems for efficient retrieval and analysis.

1.2 Phase wise Deliverables

Following are the deliverables for each of the five phases of our project.

Phase 1 - Data Integration Pipeline:
Design the architecture of the pipeline.
Implement a system to collect and consolidate IoT sensor data.
Phase 2 - Data Cleaning and Transformation:
Process the data to handle missing values, inconsistent formats, and noise.
Apply transformation techniques to prepare the dataset for analysis.
Phase 3 - Data Analysis and Visualization:
Analyze the processed data for patterns and trends.
Create a visualization dashboard for stakeholders to view insights.
Phase 4 - Machine Learning Integration:
Prepare the dataset for machine learning model training.
Train a fault detection model and evaluate its accuracy.
Phase 5 - Database Creation:
Develop a database to store extracted and processed data for future use.

1.3 Milestones and Deadlines

Milestone 1: Completion of the data integration pipeline
Milestone 2: Successful data cleaning and transformation
Milestone 3: Implementation of a fully functional data analysis and visualization dashboard
Milestone 4: Fault detection model trained and evaluated

Milestone 5: Database system operational for storing and retrieving data

1.4 Team Capability and Gaps

Capabilities:
Team proficiency in Python, SQL, and data transformation techniques should support the integration and processing tasks.
Skills in visualization tools (like Tableau or Power BI) should enable effective dashboard creation.
Experience with machine learning models, particularly in fault detection, aligns with the project's ML goals.

Potential Gaps:
Lack of familiarity with specific deep learning architectures for fault detection could pose a challenge, requiring external expertise or research.

1.5 Dataset

Link: https://drive.google.com/file/d/1haIuo22-$_tFK6r0s1-TDIWBlJKE3TXVN/view?usp = drive_link$

2 Team Discussion

2.1 Core Skills:

Aryan: Experience in data visualization using tableau and various python libraries.
Atir: Training and testing of various ML/DLmodels and python expertise.
Jwal: Experience in python and data preprocessing.

2.2 Contributions:

Aryan: Responsible for implementing a dashboard to provide data visualization to gather insights on the data.
Atir: Training and testing of Deep Learning and Machine Learning models
Jwal: Responsible for implementing the data pipeline to format the data captured using IoT devices.
Overall: Responsible for data transformation process and implementing and managing the database of extracted data.

3 Skills and Tools Assessment

3.1 Experience in:

Programming language: Python, SQL
Frameworks/Libraries: Flask, sklearn, keras, tenserflow
Tools: Tableau, Git
Platform: Anaconda, PyCharm, google collab, jupyter notebook, Kaggle

3.2 Lacking skills/Need to learn:

Streamlit, MLOps and deployment of deep learning models

4 Initial Setup:

4.1 Environment Setup:

OS: Mac/Windows
Python version: 3.x
IDE: VS Code, Anaconda
Database: MySQL
Dashboard: Tableau
Version control: Git

5 Progress Review

Following sections discuss the progress so far on our project, team contributions so far and alignment with objectives.

5.1 Progress so far:

Initial Setup: We have successfully set up the GitHub repository and established the initial project structure.
Dataset Selection: The dataset for our project has been chosen, providing us with a solid foundation for our analysis.

5.2 Issues or Blockers:

Currently, there are no significant issues or blockers reported. All team members are actively contributing, and any minor questions have been addressed in discussions.

5.3 Team Contributions:

Each team member is contributing as expected. Roles are clearly defined, and everyone understands their responsibilities. Collaboration is strong, ensuring that tasks are progressing smoothly.

5.4 Timeline and Milestones:

We are on track with our timeline and milestones. All tasks are being completed as scheduled, and we're maintaining our deadlines without the need for adjustments.

5.5 Alignment with Project Objectives:

Our progress aligns well with the project's overall objectives. The initial setup and dataset selection are crucial steps toward achieving our goal of developing a reliable fault detection system for IoT sensors.

6 Plan Revision

6.1 Timeline and Milestones:

Current Assessment: Based on our progress, we do not need to adjust the project timeline or milestones at this time. We are meeting our deadlines effectively.

6.2 Task Management:

Delayed Tasks: Currently, there are no delays reported. However, we should remain vigilant for any signs of workload imbalance.
Reassignment: If we identify any tasks that are at risk due to workload or skill gaps, we should consider reassigning them to ensure balanced contributions. Regular check-ins will help identify these situations early.

6.3 Clarity on Next Steps:

Revised Plan Communication: To ensure all team members are clear on the next steps:
Regular Meetings: Schedule weekly check-ins to discuss progress and any changes to the plan.
Documentation: Maintain an updated project document or task board (e.g., Trello, Asana) where responsibilities and timelines are clearly outlined.

6.4 Communication Strategies:

Clear Channels: Establish dedicated channels for communication (e.g., Slack, Discord) for quick updates and questions.
Daily Stand-ups: Implement short daily stand-up meetings to share progress and blockers,

fostering transparency and quick resolution of issues.
Feedback Loops: Encourage open feedback on tasks and processes to identify potential misunderstandings before they escalate.

6.5 Progress Tracking:

Task Management Tools: Use project management tools like GitHub Projects or Jira to visualize progress and track individual tasks.
Milestone Reviews: Conduct milestone reviews at the end of each phase to assess progress, celebrate achievements, and recalibrate if necessary.
Progress Reports: Implement a simple reporting mechanism where team members can log their completed tasks and upcoming goals weekly.

7 Repository

Link: https://github.com/aryanshah295/Fault-Detection-in-IoT-Sensors