**Project Overview

We need to build an AI system to **consolidate and analyze data from multiple lab instruments** studying exosomes (extracellular vesicles). The system should identify anomalies and patterns across different data types but **NOT interpret** the results—just flag them for researchers.

---

## Input Data Sources

### 1. Flow Cytometry Data (FCS files)

- `.fcs` files containing scatter plot data
- Parameters: FSC (Forward Scatter), SSC (Side Scatter), and multiple fluorescence channels (FL1-FL6)
- Each event = one particle
- Need to parse using FlowCytometry libraries

### 2. Nanoparticle Tracking Analysis (Text files)

- The `text file discussed.txt` contains ZetaView output
- Size distribution data (particle size in nm, concentration, volume, area)
- Metadata: temperature, pH, conductivity, experimental conditions

### 3. Electron Microscope Images (TEM data)

- Image files showing exosomes
- Need computer vision to:
  - Detect scale bars
  - Measure particle sizes
  - Filter background noise
  - Identify viable exosomes

### 4. Western Blot Data (future integration - early 2025)

- Not yet provided, but needs to be architected for

---

## Architecture Components

### 1. Data Ingestion Layer

**Sub-components:**

- **FCS File Parser**: Use `fcsparser` or `FlowCytometryTools` (Python)
- **Text File Parser**: Custom parser for ZetaView format
- **Image Processor**: OpenCV/PIL for TEM images
- **Metadata Extractor**: Parse experimental conditions from all sources

### 2. Data Preprocessing Layer

**Sub-components:**

- **Data Normalization**: Standardize units across different instruments
- **Quality Control Module**:
  - Check temperature compliance
  - Validate particle drift

  - Filter invalid readings
- **Size Binning Engine**: Group particles by size ranges (40-80nm, 80-100nm, 100-120nm) based on customer-provided thresholds

### 3. Computer Vision Module (for TEM)

**Sub-components:**

- **Scale Detection**: Identify and measure scale bars
- **Particle Segmentation**: Separate exosomes from background
- **Size Measurement**: Calculate particle diameters
- **Noise Filtering**: Remove artifacts

### 4. Multi-Modal Data Fusion Layer

**Sub-components:**

- **Sample ID Matcher**: Link data from same sample across instruments
- **Feature Extraction**:
  - From FCS: scatter intensities, fluorescence profiles
  - From NTA: size distributions, concentrations
  - From TEM: morphology, size validation
- **Data Alignment**: Temporal and spatial correlation

### 5. Anomaly Detection Engine

**Sub-components:**

- **Scatter Plot Analyzer**:
  - Auto-select optimal X/Y axis combinations
  - Detect population shifts between readings
  - Identify outlier clusters
- **Statistical Comparison Module**:
  - Compare repeat measurements
  - Flag significant deviations
  - Cross-validate size data (NTA vs TEM)
- **Pattern Recognition**: Use ML (clustering, PCA) to find unusual patterns

### 6. Visualization & Reporting Layer

**Sub-components:**

- **Interactive Plot Generator**: Create scatter plots with highlighted anomalies
- **Comparison Dashboard**: Side-by-side views of multiple readings
- **Alert System**: Flag specific anomalies with timestamps
- **Export Module**: Generate reports in PDF/Excel

### 7. AI/ML Core

**Sub-components:**

- **Unsupervised Learning**:
  - K-means/DBSCAN for clustering
  - Autoencoders for anomaly detection
- **Semi-supervised Learning**: Use customer feedback to refine models
- **Feature Importance**: Identify which parameters matter most

---

## Recommended Tech Stack

### Languages & Frameworks

- **Python 3.9+** (as discussed in transcript)
- **Pandas/NumPy**: Data manipulation
- **Scikit-learn**: ML algorithms
- **PyTorch/TensorFlow**: Deep learning (if needed)

### Specialized Libraries

- **fcsparser** or **FlowKit**: FCS file handling
- **OpenCV**: Image processing
- **Matplotlib/Plotly**: Visualization
- **scikit-image**: Advanced image analysis

### Storage & Pipeline

- **Database**: PostgreSQL for structured data
- **File Storage**: S3/local for raw files
- **Pipeline**: Apache Airflow or Luigi for workflow orchestration

---

## What You Need to Deliver

### Phase 1 (Initial - for Tuesday call)

1. **System Architecture Diagram** showing all components
2. **Data Flow Diagram** from input → processing → output
3. **Technology Stack Recommendations**
4. **Timeline Estimate** (6-8 months feasibility)
5. **Resource Requirements** (1-2 developers needed?)

### Phase 2 (Implementation priorities)

1. FCS file parser + basic scatter plot generation
2. NTA text file parser + size distribution analysis
3. Anomaly detection for scatter plot shifts
4. TEM image analysis (can be later phase)

---

## Key Challenges to Address

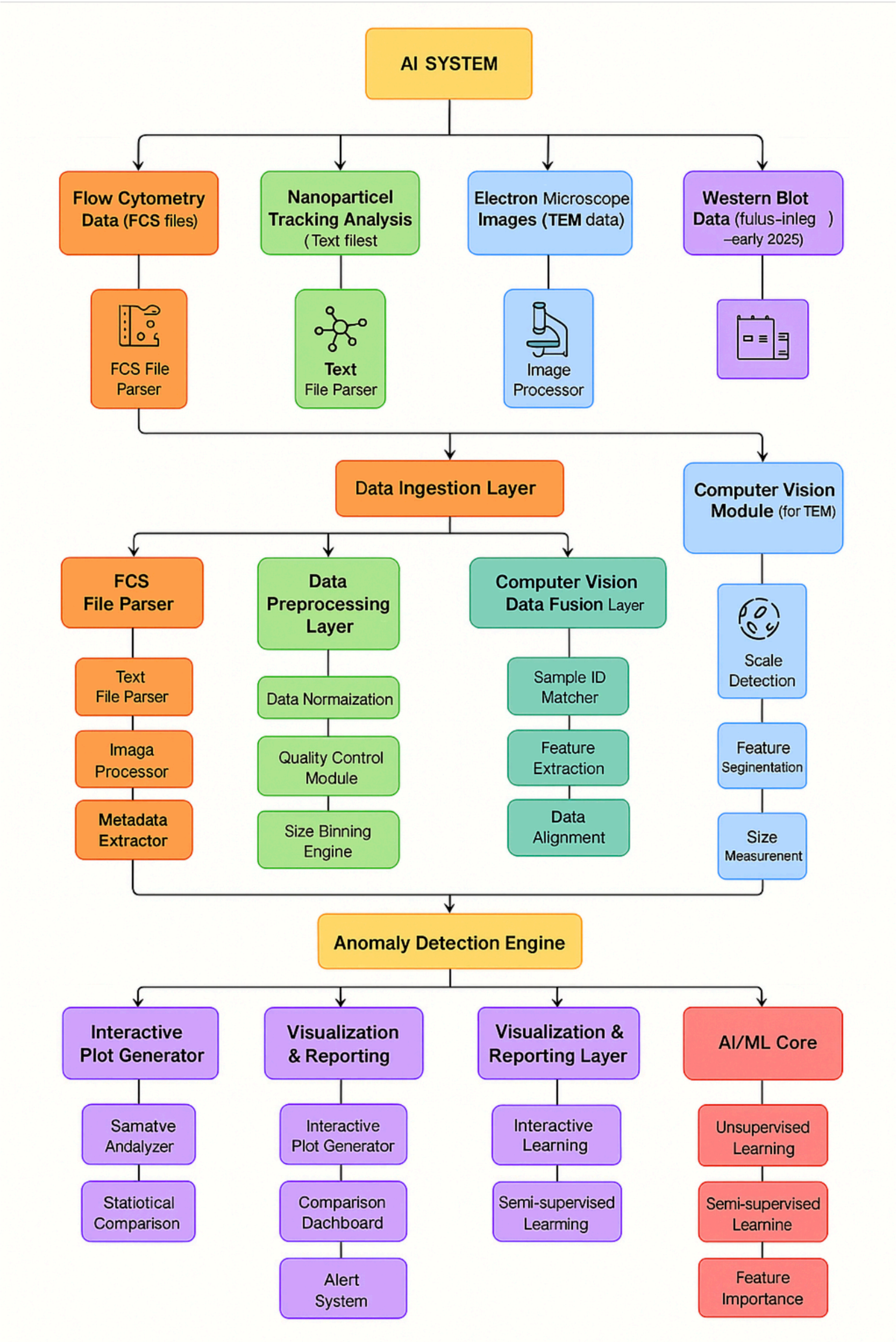
1. **How will you handle FCS files?** → Use existing Python libraries
2. **What ML approach for anomaly detection?** → Likely clustering + statistical methods
3. **How to correlate data across instruments?** → Sample ID + timestamp matching
4. **Scalability?** → Process multiple samples in batch
5. **User interface?** → Web dashboard (Flask/Django + React?)

---

## Questions to Ask the Client (Tuesday)

1. What are the "best view" combinations for scatter plots?
2. What thresholds define anomalies?
3. How many samples/week will they process?
4. Do they need real-time processing or batch?

5. What format for output reports?
6. Any existing tools they currently use?


AI SYSTEM
Flow Cytometry Data (FCS files)
Nanoparticel Tracking Analysis (Text filest
Electron Microscope Images (TEM data)
Western Blot Data (fulus-inleg ) –early 2025)
FCS File Parser
Text File Parser
Image Processor
Data Ingestion Layer
Computer Vision Module (for TEM)
FCS File Parser
Data Preprocessing Layer
Computer Vision Data Fusion Layer
Text File Parser
Imaga Processor
Metadata Extractor
Data Normaization
Quality Control Module
Size Binning Engine
Sample ID Matcher
Feature Extraction
Data Alignment
Scale Detection
Feature Segientation
Size Measurenent
Anomaly Detection Engine
Interactive Plot Generator
Visualization & Reporting
Visualization & Reporting Layer
AI/ML Core
Samatve Andalyzer
Statiotical Comparison
Interactive Plot Generator
Comparison Dachboard
Alert System
Interactive Learning
Semi-supervised Learming
Unsupervised Learning
Semi-supervised Learnine
Feature Importance

30 to 100 is one set and 100 to 200 is another set