STEDI Human Balance Analytics – Azure Big Data Architecture Proposal

Prepared by: Robert Miller  
Date: October 23, 2025

# A. Executive Summary

We built the STEDI Human Balance Analytics project to collect and analyze human balance data using IoT devices. This proposal moves that same concept off AWS and into Microsoft Azure while keeping the design principles intact—clean ingestion, structured transformation, and curated analytics layers that serve both research and operations. The purpose is to show how the same data flow can run fully within Azure’s ecosystem while improving scalability, security, and long-term maintainability.

By the end of this project, STEDI gains a cloud architecture that can process millions of balance readings per day, filter by consent, and feed machine learning models that predict stability and risk. Azure services give us a secure, compliant platform with built‑in governance, plus native tools for real‑time streaming, machine learning, and visualization.

# B. Business Problem Recap

The business challenge behind STEDI is simple but data‑heavy. Our devices record sensor readings multiple times per second for every customer. Those records pile up fast and contain a mix of valid, duplicate, or unconsented data. We need an analytics system that can handle that scale, clean it automatically, and produce a trusted dataset for research and model training. Building that on Azure lets us meet the same goals as the AWS version without violating the Microsoft‑only contract.

# C. Needs Assessment

Big data analytics fits this problem perfectly because it gives us the elasticity to ingest and process large volumes of semi‑structured data without bottlenecks. We’re designing a pipeline that separates data into Landing, Trusted, and Curated layers in Azure Data Lake Storage Gen2. That separation keeps raw data intact, tracks every transformation, and guarantees that downstream analysis uses only validated records.

We’ll process that data using Azure Databricks for distributed Spark jobs and Delta Lake for ACID‑compliant storage. Once curated, it’s served through Azure Synapse for SQL access and Power BI dashboards. Azure Machine Learning handles training and scoring of our predictive balance models.

# D. Solution Design

We designed this around Azure’s native services, replacing each AWS component with an equivalent or better Microsoft option:

• Azure IoT Hub and Event Hubs for device telemetry ingestion.  
• Azure Data Lake Storage Gen2 as our data lake (Landing → Trusted → Curated).  
• Azure Databricks for data cleansing, joins, and transformations.  
• Azure Synapse Analytics for serving curated data and reporting.  
• Azure Machine Learning for model training and prediction.  
• Power BI for dashboards and ad‑hoc analytics.

Our data model follows a lakehouse pattern. Bronze (landing) tables hold raw JSON from devices, Silver (trusted) tables store validated data joined to consented customers, and Gold (curated) tables hold aggregated readings for analysis. The curated layer is where Synapse, Azure ML, and Power BI connect.

Security and orchestration run through Azure Data Factory pipelines with Key Vault managing credentials. Role‑based access control comes from Azure AD, and every service uses private endpoints inside a secured virtual network.

# E. Justification of Choices

We chose these services because they align one‑to‑one with the workloads we proved in the Udacity lab. IoT Hub and Event Hubs handle high‑throughput device data better than any custom endpoint could. ADLS Gen2 is cost‑effective storage with ACLs fine‑grained enough for research compliance. Databricks gives us Spark scale and Delta Lake reliability. Synapse brings the data warehouse and BI layer together. And Power BI closes the loop with accessible visualization.

The architecture keeps each concern separate—collection, transformation, curation, and visualization—which matches how we’ve built and tested our previous pipelines. Every layer can scale independently, and every decision ties back to reducing complexity for the data science and analytics teams.

# F. Future Enhancements

Once this Azure deployment stabilizes, we can extend it in a few key ways. We could push parts of the model inference closer to the devices using Azure IoT Edge. We could introduce Digital Twins to simulate user movement and predict risk scenarios in real time. And we can automate retraining through Azure ML pipelines that trigger on new curated data.

# G. Implementation Plan

1. Deploy core services: IoT Hub, Event Hubs, ADLS, Databricks, Synapse, Key Vault, and ADF.  
2. Configure identity, networking, and role‑based access for each component.  
3. Build ETL jobs in Databricks for the landing → trusted → curated flow.  
4. Set up Synapse SQL views and Power BI dashboards.  
5. Train and deploy machine learning models in Azure ML.  
6. Roll out monitoring, logging, and cost controls before production.

# Data Visualization Models

Our main dashboards break down into three audiences. Executives get a fleet‑level view of device performance, ingestion lag, and fall‑risk metrics. Product and research teams get deeper data—model drift, firmware effects, and feature importances. Operations gets alerts on late or missing data and consent coverage. All of it runs through Power BI workspaces secured by role.

# H. References

* Microsoft Learn: Azure Architecture Center – Big Data & Analytics. <https://learn.microsoft.com/azure/architecture>
* Microsoft Learn: Azure Databricks Documentation. <https://learn.microsoft.com/azure/databricks>
* Microsoft Learn: Azure Synapse Analytics Documentation. <https://learn.microsoft.com/azure/synapse-analytics>
* Udacity – STEDI Human Balance Analytics case study