Dr Raj Kumar, 2026, 14:3
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology

An Open Access Journal

Health Monitoring Dashboard: Data Analytics & Visualization - Architecture, Implementation, and Future Directions

Dr Raj Kumar[1], Ayush Kumar Anand[2], Harivansh Sharma[3], Devansh Rohila[4], Aman Jaiswal[5]

[1]Assistant Professor Quantum University Roorkee

[2-5]BCA Scholar Quantum University Roorkee, Uttarakhand, India

Abstract- The rapid rise in digital health records has sparked a strong need for smart tools that convert unprocessed body data into practical medical decisions. This study introduces the Health Monitoring Dashboard (HMD), a unified web application combining data analytics, interactive visualization, and rule-based health insights. Developed using Microsoft Power BI, the platform leverages its built-in data connectors, DAX (Data Analysis Expressions) for metric calculations, Power Query for data transformation, and AI-powered visuals for predictive analytics. The interface features six components: filtering user conditions, aggregating key performance metrics, displaying time-based trends and cross-data relationships, applying rule-driven health thresholds, and offering an interactive machine learning testing environment via Power BI's integration with Azure Machine Learning. Tests confirm the system accurately handles inputs such as heart rate, blood pressure, sleep habits, steps taken, and physical activity with near-real-time dashboard refresh. Future improvements include direct integration with wearable device APIs (Fitbit, Garmin), secure cloud deployment via Power BI Service and Azure, row-level security for multi-user access, and advanced ML model integration through Azure ML pipelines. The results demonstrate that accessible BI tools can build reliable healthcare analytics systems and offer a replicable blueprint for others.

Keywords: Health monitoring dashboard; data analytics; Power BI; DAX; Power Query; Azure Machine Learning; KPI; healthcare visualization; predictive analytics; interactive dashboards.

I. INTRODUCTION

Healthcare's shift to digital formats over the last ten years has led to massive amounts of patient data being created. Devices like wearables, electronic health records, and health apps collect vast quantities of bodily signals every day. However, much of this information still lacks practical use because few tools exist that turn raw readings into useful health insights [1].

Traditional methods like spreadsheets, fixed reports, and manually reviewed notes cannot handle today's fast-moving, large-scale data volumes. Both clinicians and individuals tracking their own health need platforms that consolidate multiple measurements, surface meaningful patterns, and alert users instantly to potential concerns. Modern business intelligence tools such as Microsoft Power BI have made creating such systems significantly more accessible, enabling analysts to build dynamic, interactive dashboards without requiring deep programming expertise.

i.

ii. This study outlines the Health Monitoring Dashboard (HMD), developed at Quantum University, Roorkee, to address these challenges. The HMD uses Power BI's end-to-end data platform, encompassing data ingestion through Power Query, transformation and modelling, rich interactive visuals, DAX-powered clinical logic, and Azure Machine Learning integration for predictive analytics. Earlier approaches typically address one concern in isolation. In contrast, the HMD delivers a fully integrated solution — from raw CSV or database inputs to interpretable predictions — through a single Power BI report interface.

Research Contributions

This study offers four distinct advances:

- A modular, replicable HMD framework built entirely in Microsoft Power BI with Power Query, DAX, and Azure ML integration.
- Classification of health visualization types — time-series, cross-variable, distributional — aligned with targeted clinical analysis goals.
- A rule-based DAX engine that delivers immediate, transparent health status insights without external ML dependencies.
- A structured roadmap for future development including wearable API connectivity, Power BI Service cloud hosting, and advanced Azure ML pipelines.

II. RELATED WORK

Health informatics dashboards have attracted strong interest across clinical and consumer health contexts. Systems like Microsoft HealthVault and Apple HealthKit established early unified health record repositories but offered limited advanced analysis. Plaisant et al. [3] developed LifeLines, a visual tool rendering medical history over time, demonstrating that temporal layouts improve clinical pattern detection.

Modern deep learning approaches, including convolutional and recurrent networks, are applied to ECG irregularity detection [4] and sleep stage classification [5]. However, these methods typically demand specialized hardware and significant technical expertise, limiting adoption in routine clinical settings. This system instead leverages Power BI's accessible interface with transparent DAX measures and optional Azure ML models, prioritizing ease of use while maintaining solid analytical capability.

Multiple teams have developed BI-powered dashboards for health tracking. Bhatt et al. [6] created a real-time COVID-19 analytics application. Gupta and Sharma [7] implemented an interactive dashboard for post-surgical rehabilitation monitoring. This study builds on those efforts by covering broader health metrics, combining rule-based and machine learning analysis within a single Power BI environment, and providing a replicable design framework.

III. SYSTEM ARCHITECTURE

The HMD uses a five-layer pipeline designed for flexibility and clear separation of concerns. Each layer is described below.

Data Input Layer

Health data enters the system through Power Query's Get Data connectors. Currently, a structured CSV file with timestamped entries for heart rate, blood pressure, body temperature, blood oxygen saturation (SpO_2), calorie intake, sleep duration, steps taken, and physical activity type serves as the primary input. Future expansion will include direct REST API connections to wearable platforms. Power Query's data type detection and datetime parsing ensure consistent schema enforcement across all data sources.

Data Processing Layer

All transformations are performed in Power Query (M language). The workflow runs four stages: schema validation and null-value filling using Table.FillDown(); outlier flagging via conditional columns based on IQR thresholds; feature engineering including seven-day rolling averages (using DAX AVERAGEX with DATESINPERIOD) and binary risk flags anchored to clinical limits; and KPI aggregation via DAX CALCULATE and SUMMARIZE functions. Power Query's modular step design allows replacing individual transformation steps without affecting downstream logic.

Visualization Layer

The visualization layer uses Power BI's native and custom visuals because of their interactive capabilities including cross-filtering, drill-through, tooltips, and slicers. Four chart families are used based on their analytical role:

Chart Family	Analytical Objective xv.
Line / Bar (time-series)	Temporal trend detection in step count, heart rate, and sleep xvi. xvii. xviii.
Scatter Plot	Cross-variable correlation (e.g., sleep vs. heart rate)
Histogram/ Distribution	Parameter distribution profiling and normality assessment
Pie / Donut	Categorical proportion analysis (activity type breakdown)

xix.

All visuals are placed on report pages formatted in xx.
Power BI Desktop's responsive canvas. Slicers xxi.
propagate filter context across all visuals on a page, ensuring analytical coherence.

Prediction Layer

The system uses two linked forecasting methods. The first is a DAX-powered rule engine that evaluates input values against WHO-recommended thresholds — for example, heart rate above 100 bpm indicating tachycardia, or sleep below 360 minutes indicating inadequacy. This produces GREEN, AMBER, or RED xxii.
status labels with no latency or compute cost, xxiii.
functioning reliably independent of any ML service. xxiv.
The second method integrates a trained Azure Machine Learning model via Power BI's Azure ML integration panel. It accepts six health metrics and returns a wellness classification: Healthy, At Risk, or Requires Attention. The model is invoked through Power Query's Invoke Function step and its output is surfaced as a calculated column within the data model.

xxv.

Output Layer xxvi.

Processed results are presented through Power BI's xxvii.
interactive report canvas. The design uses a three-column KPI card row at the top of each page, followed by full-width chart panels. Conditional formatting on card visuals renders status colours (green, amber, red) based on DAX measures. A dedicated Prediction page allows users to enter hypothetical health values via What-If parameters and immediately see classification outputs and feature importance visuals.

IV. MODULE DESCRIPTIONS

1. User-State Filtering Module

The report sidebar contains core interaction controls: a dropdown slicer for activity level, a date range picker, and a multi-select slicer for health parameters. Each slicer propagates filter context to all visuals on the page through Power BI's cross-filter engine. This ensures KPIs, charts, rule outputs, and predictions all reflect the same filtered dataset, avoiding inconsistencies that arise when components source data independently.

2. KPI Aggregation Module

Six key performance indicators are computed via DAX measures applied to the filtered model: average heart rate, average SpO_2, total daily steps (SUM), average sleep duration, average sedentary time, and an Activity Score derived from weighted combinations of very-active, fairly-active, and lightly-active minutes (weights of 3, 2, 1 scaled to 0–100). Card visuals display each KPI alongside a trend indicator derived from the seven-day moving average, highlighting directional change.

3. Rule-Based Health Heuristics Module

The DAX heuristics engine evaluates five health domains — cardiovascular, respiratory, metabolic, sleep, and physical activity — against thresholds set by WHO and AHA guidelines. Each domain receives a status of GREEN, AMBER, or RED rendered through conditional formatting on card and table visuals. Built entirely as DAX measures without external dependencies, the engine executes at render time regardless of dataset size.

4. ML Predictive Playground Module

The predictive playground uses Power BI What-If parameters to expose six numeric input sliders matching the Azure ML model's feature set. When the report refreshes or the user adjusts sliders, values are passed through a Power Query Invoke Function call to the Azure ML scoring endpoint. The most likely classification and its confidence scores are displayed in a bar chart visual. A precomputed SHAP-style feature importance table, imported as a static dataset, is shown alongside predictions to support transparency required in healthcare AI applications.

V. IMPLEMENTATION DETAILS xxviii.

xxix.

1. Technology Stack xxx.

Component	Technology / Version
BI Platform	Microsoft Power BI Desktop (June 2024)
Data Transformation	Power Query (M Language)
Metric Calculations	DAX (Data Analysis Expressions)
Visualisation	Power BI Native + Custom Visuals
ML Integration	Azure Machine Learning (Scoring Pipeline) xxxi.
Model Training	Azure ML with Scikit-learn 1.4 backend xxxii.
Cloud Deployment	Power BI Service / Azure xxxiii. xxxiv.
Dataset Format	CSV (primary) / Azure SQL (optional)

2. Data Processing Pipeline

Power Query handles all ingestion and transformation steps. CSV files are loaded via the Text/CSV connector with explicit data-type assignments. Null values are forward-filled using Table.FillDown(). IQR-based outlier flags are added as conditional columns. Rolling averages and risk markers are then computed as DAX calculated columns in the data model. The modular Power xxxv. Query step list allows any single transformation to be xxxvi. replaced or reordered without breaking downstream xxxvii. measures.

3. Model Training Protocol

A Random Forest Classifier with 200 trees, maximum depth of 8, and balanced class weights was trained in Azure Machine Learning using a synthetic dataset of 10,000 labelled records generated through Monte Carlo simulations based on physiological constraints. Three classes were produced: Healthy (60%), At Risk (30%), Requires Attention (10%). Cross-validation yielded 91.4% accuracy and a macro F1 score of 0.89. The trained model was registered in Azure ML Model Registry and deployed as a real-time scoring endpoint, invoked from Power BI via the Azure ML integration connector in Power Query.

4. Environment Setup

The complete environment can be reproduced with the following steps: (1) Install Power BI Desktop from the Microsoft Store or powerbi.microsoft.com. (2) Open the provided .pbix file. (3) Update the CSV data source path or Azure SQL connection string in Power Query settings. (4) For ML predictions, connect to an active Azure ML scoring endpoint via the Azure Machine Learning connector in Power Query's Get Data dialog. The dashboard runs on any Windows machine with at least 4 GB RAM; 8 GB RAM with an Intel Core i5 or equivalent is recommended for smooth performance on 10,000+ record datasets.

VI. FUTURE DIRECTIONS

Advanced ML Model Integration

The existing Random Forest baseline deployed in Azure ML will be extended to evaluate gradient boosting methods (XGBoost, LightGBM) and deep learning approaches (TabNet) using larger, demographically diverse datasets. Power BI's AutoML capabilities within Power BI Premium will be explored for automated model selection across KPI forecasting tasks. All models will be versioned in Azure ML Model Registry and consumed through standardised Power Query Invoke Function calls, ensuring consistent integration regardless of algorithm.

Wearable API Integration (Fitbit / Garmin)

Real-time wearable data will be ingested via OAuth 2.0-secured REST connections to Fitbit Web API (v1.2) and Garmin Health API, configured as Web data sources in Power Query. A Power Automate flow will poll these APIs at five-minute intervals and write new records to an Azure SQL database. The Power BI dataset will use DirectQuery or scheduled incremental refresh against this database, replacing the static CSV dependency. A pilot with ten volunteers will measure latency, API rate limits, and token refresh timing.

Secure Cloud Deployment

The HMD will be published to Power BI Service with xxxviii row-level security (RLS) enforced through DAX role xxxix filters, restricting each user to their own health records. An Azure Active Directory (Entra ID) integration will govern authentication. Data at rest in Azure SQL will be encrypted with AES-256 and in transit via TLS 1.3. Power BI Premium capacity will be used to support paginated reports, larger dataset sizes, and XMLA endpoint access for enterprise-grade governance.

xl.

Multi-User Authentication and Profile Controls

Power BI Service workspaces will be configured with xli. role-based access — Viewer, Contributor, and Admin xlii. — managed through Microsoft 365 groups. Each user's profile history, personal targets, and notification preferences will be stored in an Azure SQL user-profile table linked to the data model. Admin reports will provide usage analytics, group trend comparisons, and audit logs, transforming the HMD from a single-user tool into a scalable, multi-tenant health analytics platform.

VII. DISCUSSION

xliii.

xliv.

The Health Monitoring Dashboard demonstrates that Power BI's reactive cross-filter engine, Plotly-equivalent interactive visuals, and Power Query's vectorised transformation capabilities create a robust foundation for healthcare analytics on standard hardware. Dashboard interactions triggering visual updates complete in approximately 300 to 600 milliseconds on a 10,000-record dataset — well within the sub-second threshold perceived as responsive by users [8].

The combination of DAX rule-based heuristics and xlv. Azure ML predictions is deliberate. Rules provide xlvi. immediate, deterministic responses grounded in validated clinical guidelines, ensuring consistent safety signals. The ML component contributes personalised forecasts that improve with accumulated data and surfaces prediction confidence through probability outputs. This approach aligns with augmented intelligence principles [9], positioning the system as clinical decision support rather than a replacement for clinicians.

The primary constraint remains dependence on static CSV files and an ML model trained on synthetic data. Validation against real patient cohorts with verified health outcomes is essential before any clinical deployment. The wearable API and cloud deployment roadmap outlined in Section VI directly addresses the data currency limitation. Prospective clinical trials using live datasets are the top-priority next phase.

VIII. CONCLUSION

This paper presents the Health Monitoring Dashboard, an open-source Power BI solution that unifies data analytics, interactive visualization, DAX-powered clinical logic, and Azure Machine Learning predictions within a single browser-accessible report environment. Its five-layer design — covering data ingestion through to live user interaction — separates concerns clearly, enabling each component to be tested, updated, or replaced independently.

The dashboard addresses a gap in accessible public health tooling by requiring no custom code to run. It operates from Power BI Desktop using a single .pbix file. Features include dynamic KPI cards, multi-format interactive charts, visible health status alerts, and ML-backed forecasts with explainability visuals — delivering functionality comparable to commercial health analytics platforms without additional licensing costs beyond the Power BI subscription. Future development plans extend it into a real-time, multi-user, cloud-based platform for health analysis.

The methodology described here offers a practical foundation for researchers and engineers developing health monitoring systems in clinical, wellness, or workplace environments.

REFERENCES

1. World Health Organization. (2021). Global strategy on digital health 2020–2025. WHO Press.
2. Microsoft Corporation. (2024). Power BI documentation: Build and share data reports. https: //learn.microsoft.com/power-bi
3. Plaisant, C., Milash, B., Rose, A., Widoff, S., & Shneiderman, B. (1996). LifeLines: Visualizing personal histories. CHI '96 Proceedings, ACM, 221–227.
4. Hannun, A. Y., et al. (2019). Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. Nature Medicine, 25(1), 65–69.
5. Supratak, A., Dong, H., Wu, C., & Guo, Y. (2017). DeepSleepNet: A model for automatic sleep stage scoring. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 25(11), 1998–2008.
6. Bhatt, P., Patel, D., & Shah, M. (2021). A real-time COVID-19 analytics web application. Journal of Medical Internet Research, 23(4), e24252.
7. Gupta, R., & Sharma, A. (2022). Interactive dashboard for post-surgical rehabilitation monitoring using Plotly Dash. International Journal of Medical Informatics, 162, 104762.
8. Nielsen, J. (1993). Response times: The three important limits. In Usability Engineering. Academic Press.
9. Topol, E. J. (2019). High-performance medicine: The convergence of human and artificial intelligence. Nature Medicine, 25(1), 44–56.
10. Microsoft Corporation. (2023). DAX reference documentation. https: //learn.microsoft.com/dax
11. Géron, A. (2022). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow (3rd ed.). O'Reilly Media.
12. Han, J., Kamber, M., & Pei, J. (2011). Data Mining: Concepts and Techniques (3rd ed.). Morgan Kaufmann.