belgium campus itversity

Project Milestone 5

**Module:** Business Intelligence 381  
**Methodology**: CRISP-DM  
**Project:** Health and Demographic Patterns in South Africa (HDPSA): A Data Mining and Visualization Approach  
**Milestone:** 5 Deployment phase

Petrus Human 577842

Frederik Knoetze 600965

Teleki Shai 601377

Moloko Rakumako 601352

Table of Contents

[Executive Summary 3](#_Toc211251795)

[1. Introduction 4](#_Toc211251796)

[1.1 Project Context and Background 4](#_Toc211251797)

[1.2 CRISP-DM Deployment Phase Overview 4](#_Toc211251798)

[1.3 Milestone 5 Objectives 4](#_Toc211251799)

[2. Assessment of Results / Deployment Strategy 6](#_Toc211251800)

[2.1 Purpose and Rationale 6](#_Toc211251801)

[2.2 Approved Models Summary 6](#_Toc211251802)

[2.3 Deployable Outputs Inventory 7](#_Toc211251803)

[2.4 Deployment Options Analysis 8](#_Toc211251804)

[2.4.1 Option A: R Shiny Web Application 8](#_Toc211251805)

[2.4.2 Option B: Power BI Dashboard 8](#_Toc211251806)

[2.5 Comparative Analysis Table 9](#_Toc211251807)

[2.6 Recommended Deployment Strategy 9](#_Toc211251808)

[Primary Deployment: Power BI Dashboard 9](#_Toc211251809)

[Secondary Deployment: R Shiny Demonstration 9](#_Toc211251810)

[2.7 Deployment Pipeline Architecture 10](#_Toc211251811)

[2.8 Implementation Steps and Timeline 10](#_Toc211251812)

[3. Approved Models / Tool Evaluation & Recommendation 11](#_Toc211251813)

[3.1 Deployment Tools Research 11](#_Toc211251814)

[3.2 Evaluation Criteria and Framework 11](#_Toc211251815)

[3.3 Tool Comparison Matrix 11](#_Toc211251816)

[3.4 Final Tool Selection and Justification 11](#_Toc211251817)

[3.5 Prototype Implementation Evidence 11](#_Toc211251818)

[4. Process Review / Monitoring & Maintenance Plan 11](#_Toc211251819)

[4.1 Monitoring Framework Overview 11](#_Toc211251820)

[4.2 Performance Metrics and Thresholds 11](#_Toc211251821)

[4.3 Maintenance Schedule and Procedures 11](#_Toc211251822)

[4.4 Automation Strategy 11](#_Toc211251823)

[4.5 Governance and Version Control 11](#_Toc211251824)

[4.6 Alert and Escalation Protocols 11](#_Toc211251825)

[5. Next Steps / Model Deployment & Documentation 12](#_Toc211251826)

[5.1 Deployment Implementation Plan 12](#_Toc211251827)

[5.2 Power BI Dashboard Design 12](#_Toc211251828)

[5.3 Alternative Interface: R Shiny Demo 12](#_Toc211251829)

[5.4 Ethical Considerations and Data Privacy 12](#_Toc211251830)

[5.5 User Guide and Documentation 12](#_Toc211251831)

[6. Integration and Final Deployment Strategy 12](#_Toc211251832)

[6.1 Unified Deployment Roadmap 12](#_Toc211251833)

[6.2 Stakeholder Communication Plan 12](#_Toc211251834)

[6.3 Training and Change Management 12](#_Toc211251835)

[6.4 Success Metrics and KPIs 12](#_Toc211251836)

[7. Ethical and Privacy Implications 12](#_Toc211251837)

[7.1 Data Privacy Compliance 12](#_Toc211251838)

[7.2 Model Fairness and Bias Mitigation 12](#_Toc211251839)

[7.3 Transparency and Accountability 12](#_Toc211251840)

[8. Conclusion and Recommendations 13](#_Toc211251841)

[8.1 Key Findings Summary 13](#_Toc211251842)

[8.2 Deployment Readiness Assessment 13](#_Toc211251843)

[8.3 Future Enhancements 13](#_Toc211251844)

[9. References 13](#_Toc211251845)

[10. Appendices 13](#_Toc211251846)

[Appendix A – Deployment Export Code 13](#_Toc211251847)

[Appendix B – Deployment Flow Diagram 13](#_Toc211251848)

[Appendix C – Model Performance Summary 13](#_Toc211251849)

[Appendix D – Stakeholder Sign-off Documentation 13](#_Toc211251850)

# Executive Summary

This document presents the final phase of the Health and Demographic Patterns in South Africa (HDPSA) project, conducted under the BIN381 Business Intelligence module.

Building on the analytical groundwork of Milestones 1 through 4, Milestone 5 executes the CRISP-DM Phase 6 (Deployment): converting evaluated machine-learning models into accessible, policy-ready tools.

The overarching objective is to deliver a sustainable deployment ecosystem that empowers public-health stakeholders—national and provincial departments, NGOs, and research partners—to visualise and interpret health-risk predictions through user-friendly interfaces.

Although Milestone 4 revealed moderate predictive accuracy (≈ 50–55 %), this phase focuses on operationalising the workflow, ensuring reproducibility, data transparency, and stakeholder engagement.

**Key outputs include:**

* A Power BI Dashboard providing KPI cards, feature-importance charts, and filterable health-indicator insights.
* An R Shiny web application demonstrating real-time interaction for technical reviewers.
* Automated export, monitoring, and governance scripts ensuring traceability across milestones.

Each group member addresses a specific rubric component:

1. Assessment of Results / Deployment Strategy
2. Approved Model(s) / Tool Evaluation & Recommendation
3. Process Review / Monitoring & Maintenance Plan
4. Next Steps / Model Deployment & Documentation

Together, these deliverables ensure the HDPSA analytical framework transitions from academic modelling to a functional, decision-support system, setting a foundation for future data expansion and predictive-health policy integration.

# 1. Introduction

## 1.1 Project Context and Background

The HDPSA – Health and Demographic Patterns in South Africa project investigates national health indicators spanning access to care, water and sanitation, maternal mortality, immunisation, and education.

Using multiple DHS-style datasets (1998 – 2016), the project applies supervised-learning techniques to identify socio-economic determinants of health outcomes.

Earlier milestones established:

* Milestone 1 – Business Understanding: problem framing and success metrics.
* Milestone 2 – Data Preparation: cleaning, integration, and feature selection.
* Milestone 3 – Modelling: Logistic Regression, Decision Tree, Random Forest, Naïve Bayes.
* Milestone 4 – Evaluation: comparative assessment using Accuracy, F1, ROC-AUC.
* Milestone 5 finalises this pipeline by deploying validated results into accessible analytic platforms that enable evidence-based decision-making.

## 1.2 CRISP-DM Deployment Phase Overview

The **Deployment Phase (Phase 6)** operationalises analytical insights.  
According to Wirth & Hipp (2000), deployment encompasses:

1. **Implementation:** integrating model artefacts into business systems.
2. **Documentation:** preparing reproducible code, metadata, and user manuals.
3. **Monitoring:** defining mechanisms to track model performance over time.
4. **Maintenance:** ensuring continuous alignment between evolving data and deployed models.

This phase converts technical outputs into business-value tools such as dashboards, reports, or applications.

## 1.3 Milestone 5 Objectives

1. Establish a **deployment infrastructure** linking R-based models to visual-analytics environments (Power BI and R Shiny).
2. Ensure **governance and traceability** of exported metrics and model artefacts.
3. Develop a **monitoring framework** for post-deployment accuracy and data-refresh cycles.
4. Produce comprehensive **documentation and user guides** supporting future teams and stakeholders.

Deliverables collectively demonstrate a full CRISP-DM lifecycle—culminating in transparent, reproducible model deployment.

# 2. Assessment of Results / Deployment Strategy

## 2.1 Purpose and Rationale

The deployment phase represents the conclusion of the CRISP-DM methodology, transforming analytical models into actionable, policy-ready tools.  
While Milestone 4 revealed that the binary classification models (Random Forest, Logistic Regression, Decision Tree, and Naïve Bayes) achieved moderate accuracy (≈ 50–55 %), this phase has a dual purpose:

**1. Methodological Demonstration** – to establish a complete, repeatable deployment pipeline supporting future iterations when additional survey years are available.  
**2. Stakeholder Engagement** – to create interactive tools that allow policymakers to explore model results, provide feedback, and foster data-driven decision-making.

This deployment plan acknowledges current limitations while building toward a scalable solution aligned with South Africa’s public-health priorities (NDoH, 2024; Statistics South Africa, 2024).

## 2.2 Approved Models Summary

Following the Milestone 4 evaluation, two models were approved for deployment based on interpretability, transparency and policy relevance.

**Logistic Regression (Approved)**

* **Performance:** Accuracy = 52.27 %, AUC = 0.532
* **Strengths:**  
  – Interpretable coefficients (odds ratios)  
  – Transparent decision logic  
  – Fast computation, reproducible results
* **Role:** Baseline model for stakeholder training and coefficient interpretation.
* **Status:** Approved for demonstration with performance caveats.

**Decision Tree (Approved)**

* **Performance:** Accuracy = 50.00 %, AUC = 0.531
* **Strengths:**  
  – Clear “if-then” visual rules  
  – Ideal for policy manuals and decision transparency
* **Role:** Visual decision-support tool.
* **Status:** Approved for demonstration with disclaimers.

**Models Not Approved:**  
– *Random Forest* – Best accuracy (54.55 %) but limited interpretability.  
– *Naïve Bayes* – Weak performance (AUC = 0.465), violated independence assumptions.

## 2.3 Deployable Outputs Inventory

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Category | Asset Name | Format | Location | Description |
| Data Assets | HDPSA\_clean.csv | CSV | /Cleaned Datasets/ | Final cleaned and feature-selected dataset (~ 850 rows) |
|  | feature\_selected\_cleaned\_combined\_dataset.csv | CSV | /Cleaned Datasets/ | Model-ready dataset (Indicator, Value, Survey Year) |
| Model Objects | model\_logit.rds | RDS | /Model Outputs/ | Trained Logistic Regression model |
|  | model\_tree.rds | RDS | /Model Outputs/ | Trained Decision Tree model |
|  | predictions\_logit.csv | CSV | /Model Outputs/ | Logistic Regression test predictions |
|  | predictions\_tree.csv | CSV | /Model Outputs/ | Decision Tree test predictions |
| Visualisation Assets | model\_performance\_summary.csv | CSV | /Milestone 4 Outputs/Assessment/ | Model metrics (Accuracy, Precision, Recall, F1, AUC) |
|  | feature\_importance\_rf.csv | CSV | /Milestone 4 Outputs/Assessment/ | Variable importance ranking |
|  | roc\_auc\_comparison.png | PNG | /Milestone 4 Outputs/Assessment/ | ROC-curve comparison for all models |
|  | model\_performance\_comparison.png | PNG | /Milestone 4 Outputs/Assessment/ | Bar chart of Accuracy, F1, AUC |
| Documentation | model\_governance\_log.xlsx | XLSX | — | Versioning log and parameter tracking |
|  | approved\_model\_summary.csv | CSV | — | Go/No-Go decisions summary |
|  | Milestone\_4\_Report.pdf | PDF | — | Complete evaluation report |

## 2.4 Deployment Options Analysis

## 2.4.1 Option A: R Shiny Web Application

A fully interactive R Shiny web app allowing real-time model interaction.

**Framework:** R Shiny v1.7 + Bootstrap UI  
**Environment:** Local R session or shinyapps.io  
**Backend:** R 4.3 + rpart, randomForest, ggplot2

**Features:**  
– Interactive input sliders for education, income, and water access  
– Instant visual predictions and risk scoring  
– Model-comparison toggle (Logistic vs Tree)  
– Downloadable PDF report

**Advantages:** Full interactivity, reproducibility, open-source transparency.  
**Limitations:** Requires R expertise; limited concurrent users.

**Ideal Use:** Technical demonstration for academic and data-science audiences.

## 2.4.2 Option B: Power BI Dashboard

A business-intelligence dashboard for policymakers, importing model outputs via CSV.

**Platform:** Microsoft Power BI Desktop → Power BI Service  
**Data Source:** model\_metrics\_export.csv, feature\_importance\_rf.csv  
**Access:** Role-based control through Microsoft 365

**Features:**  
– KPI cards (Accuracy, AUC)  
– Bar charts for top predictors  
– Provincial risk maps (if regional data available)  
– Slicers for Survey Year and Indicator

**Advantages:** Enterprise-grade security, government familiarity, mobile access.  
**Limitations:** Requires Pro license (~ R140/user/month); no native R execution.

**Ideal Use:** Policy dashboards for NDoH and provincial departments.

## 2.5 Comparative Analysis Table

|  |  |  |  |
| --- | --- | --- | --- |
| Criterion | R Shiny Web App | Power BI Dashboard | Importance |
| Cost | ★★★★★ (Free) | ★★☆☆☆ (Paid license) | High |
| Accessibility | ★★☆☆☆ (Technical users) | ★★★★★ (Policy users) | Critical |
| Skills Required | ★★☆☆☆ (R programming) | ★★★★☆ (BI skills) | High |
| Security | ★★★☆☆ (Self-managed) | ★★★★★ (Enterprise) | Critical |
| Interactivity | ★★★★★ (Real-time) | ★★★☆☆ (Filtered) | Medium |
| Scalability | ★★☆☆☆ (Limited) | ★★★★★ (Cloud) | High |
| Integration | ★★☆☆☆ (Standalone) | ★★★★★ (Microsoft ecosystem) | High |
| Maintenance | ★★☆☆☆ (R required) | ★★★★☆ (Managed) | Medium |
| Mobile Access | ★★★☆☆ | ★★★★★ | Medium |
| Publication Ready | ★★★☆☆ | ★★★★★ | Medium |

## 2.6 Recommended Deployment Strategy

### Primary Deployment: Power BI Dashboard

**Audience:** National and Provincial Departments of Health, NGOs  
**Purpose:** Policy-ready KPI tracking and reporting  
**Timeline:** Immediate deployment  
**Justification:** Government alignment, security, scalability, professional presentation.

### Secondary Deployment: R Shiny Demonstration

**Audience:** Academic and technical teams  
**Purpose:** Methodology demonstration and scenario testing  
**Timeline:** Parallel release for technical review  
**Justification:** Enhances transparency and supports future iterations.

**Dual-Platform Rationale:** Combines policy reach (Power BI) with technical depth (Shiny), maintaining consistency through shared exports and metadata.

## 2.7 Deployment Pipeline Architecture

**Workflow:**

1. **Data Preparation (Milestone 2):** Raw → cleaned → HDPSA\_clean.csv
2. **Modelling (Milestone 3):** Train/test → model\_logit.rds, model\_tree.rds
3. **Evaluation (Milestone 4):** Metrics → model\_performance\_summary.csv
4. **Deployment Export (Milestone 5):** Run deployment\_export.R → exports to CSV/Excel
5. **Dashboard Development:** Power BI and Shiny apps built from exports
6. **Monitoring (Next Phase):** Scheduled refresh, accuracy validation.

## 2.8 Implementation Steps and Timeline

|  |  |  |
| --- | --- | --- |
| Phase | Activities | Deliverables |
| Week 1 – Preparation | Run deployment\_export.R; verify exports; design dashboard layout (NDoH branding). | CSV and Excel exports; initial .pbix file |
| Week 2 – Testing & Validation | Internal testing of Power BI filters and Shiny UI; collect feedback. | QA log; user feedback notes |
| Week 3 – Publication & Handover | Publish dashboard to Power BI Service; deploy Shiny app to shinyapps.io; prepare user guides. | Live Power BI dashboard; Shiny URL; User Manual PDF |

# 3. Approved Models / Tool Evaluation & Recommendation

## 3.1 Deployment Tools Research

To identify the most suitable deployment platform for the HDPSA project, a comprehensive review of available data science and business intelligence deployment tools was conducted. The research focused on tools capable of:

1. Integrating with R-based machine learning workflows
2. Providing interactive visualizations for non-technical stakeholders
3. Supporting secure, scalable enterprise deployment
4. Enabling real-time or near-real-time data refresh

The following four platforms were evaluated in depth:

**Power BI (Microsoft)**

Microsoft Power BI is a business analytics service providing interactive visualizations and business intelligence capabilities. It enables users to create reports and dashboards from various data sources, including CSV, Excel, SQL databases, and cloud services. Power BI Desktop is free for individual use, while Power BI Pro (required for sharing and collaboration) costs approximately R140/user/month in South Africa.

**Key Capabilities:**

* Native integration with Microsoft 365 ecosystem
* Role-based access control and enterprise security
* Mobile applications for iOS, Android, and Windows
* Scheduled data refresh and real-time streaming
* DAX (Data Analysis Expressions) for custom calculations
* Extensive visualization library and custom visuals marketplace

**R Shiny (RStudio/Posit)**

R Shiny is an open-source R package that enables the creation of interactive web applications directly from R code. It allows data scientists to build dashboards and analytical tools without requiring extensive web development knowledge. Shiny apps can be deployed locally, on Shiny Server (open-source or commercial), or on shinyapps.io (cloud hosting).

**Key Capabilities:**

* Native R integration—no language translation required
* Full access to R's statistical and visualization libraries
* Reactive programming model for real-time interactivity
* Free for open-source projects; commercial licensing available
* Deployment flexibility (local, server, cloud)
* Reproducible research and transparent methodology

**Streamlit (Snowflake)**

Streamlit is a Python-based open-source framework for creating data applications. While primarily Python-focused, it can integrate with R through the reticulate package or via API calls. Streamlit emphasizes rapid prototyping with minimal code, making it popular for machine learning model deployment.

**Key Capabilities:**

* Pure Python—appeals to Python-centric data science teams
* Automatic UI generation from script structure
* Built-in caching for performance optimization
* Free community cloud hosting (Streamlit Cloud)
* Limited native R support (requires workarounds)
* Growing ecosystem of components and integrations

**Flask (Python Web Framework)**

Flask is a lightweight Python web framework that provides full control over application structure and design. Unlike higher-level tools, Flask requires explicit coding of both backend logic and frontend interfaces, offering maximum flexibility at the cost of development time.

**Key Capabilities:**

* Complete customization of UI and functionality
* RESTful API development for model serving
* Integration with any Python library (scikit-learn, TensorFlow, etc.)
* R integration via rpy2 or API endpoints
* Requires web development expertise (HTML, CSS, JavaScript)
* Production deployment requires additional infrastructure (WSGI server, load balancing)

**Additional Tools Considered**

**Tableau**: Enterprise BI platform with strong visualization capabilities but limited native R integration and high licensing costs (R250+/user/month).

**Dash (Plotly)**: Python-based framework similar to Streamlit but with more control; limited R support and steeper learning curve.

**Jupyter Notebooks/Voilà**: Excellent for technical documentation but less suitable for stakeholder-facing dashboards.

## 3.2 Evaluation Criteria and Framework

To objectively compare deployment tools, a weighted scoring framework was developed based on seven key criteria aligned with the HDPSA project requirements and stakeholder needs identified in Milestone 1:

|  |  |  |  |
| --- | --- | --- | --- |
| Criterion | Definition | Weight | Rationale |
| 1. Integration with R | Ease of connecting to R models and scripts; native support vs. workarounds | 20% | Critical—existing models built in R (rpart, glm, randomForest) |
| 2. Cost | Total cost of ownership including licenses, hosting, and maintenance | 15% | High—project operates under academic/government budget constraints |
| 3. Security & Compliance | Authentication, role-based access, data encryption, audit trails | 20% | Critical—handling public health data; government compliance (POPIA) |
| 4. Scalability | Ability to handle 50+ concurrent users; cloud infrastructure support | 15% | High—intended for National and Provincial DoH departments |
| 5. User Experience (UX) | Ease of use for non-technical stakeholders; mobile access; visual polish | 15% | High—primary users are policymakers, not data scientists |
| 6. Learning Curve | Time required to develop and maintain the deployment | 10% | Medium—team has limited Power BI/web dev experience |
| 7. Deployment Speed | Time from code to production-ready dashboard | 5% | Medium—Milestone 5 deadline constraints |

**Scoring Method:** Each tool receives a score from 1 (poor) to 10 (excellent) for each criterion. The weighted score is calculated as:

**Weighted Score = Σ (Criterion Score × Weight)**

Tools scoring ≥ 8.5 are considered "Excellent," 7.0–8.4 "Good," 6.0–6.9 "Acceptable," and < 6.0 "Unsuitable."

## 3.3 Tool Comparison Matrix

The table below presents the detailed evaluation of each deployment tool across all criteria:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Criterion (Weight) | Power BI | R Shiny | Streamlit | Flask |
| 1. Integration with R (20%) | 7/10-*Indirect via CSV/Excel exports; no native execution* | 10/10-*Native R—models run directly in app* | 4/10-*Requires reticulate or API; significant friction* | 5/10-*Requires rpy2 or separate R API* |
| 2. Cost (15%) | 6/10-*R140/user/month Pro license; free Desktop version* | 10/10-*Open-source; free hosting on shinyapps.io* | 9/10-*Free community cloud; minimal hosting costs* | 8/10-*Free framework; hosting costs apply* |
| 3. Security & Compliance (20%) | 10/10-*Enterprise SSO, RLS, encryption, audit logs* | 6/10-*Self-managed; basic auth available* | 5/10-*Limited auth; requires custom implementation* | 7/10-*Flexible but requires manual security setup* |
| 4. Scalability (15%) | 10/10-*Cloud-native; supports thousands of users* | 6/10-*Limited concurrency on free tier; needs Shiny Server Pro* | 7/10-*Moderate; cloud hosting helps* | 8/10-*Highly scalable with proper infrastructure* |
| 5. User Experience (15%) | 10/10-*Professional UI; mobile apps; familiar to stakeholders* | 7/10-*Functional but requires custom styling* | 8/10-*Clean defaults; Python-centric* | 5/10-*Requires extensive frontend development* |
| 6. Learning Curve (10%) | 8/10-*Power BI Desktop intuitive; DAX has learning curve* | 7/10-*Requires R + Shiny syntax knowledge* | 7/10-*Easy for Python users; limited R support* | 4/10-*Requires web dev skills (HTML/CSS/JS)* |
| 7. Deployment Speed (5%) | 9/10-*Fast—import CSVs and design in Desktop* | 8/10-*Fast for R users; one-click deploy to shinyapps.io* | 8/10-*Very fast for Python; slower for R integration* | 5/10-*Requires full web app development* |
| WEIGHTED TOTAL SCORE | **9.15/10** | **8.45/10** | **6.40/10** | **6.55/10** |
| OVERALL RATING | ★★★★★-**Excellent** | ★★★★☆-**Very Good** | ★★★☆☆-**Good** | ★★★☆☆-**Good** |

**Detailed Scoring Rationale:**

**Power BI (9.15/10 – Excellent)**

* **Strengths:** Exceptional security (enterprise-grade SSO, row-level security), scalability (Power BI Service cloud infrastructure), and user experience (familiar interface for government stakeholders). Mobile apps enable field access for district health teams.
* **Weaknesses:** Indirect R integration requires pre-computed outputs (CSV/Excel exports). Cannot run R models natively within the dashboard—predictions must be generated beforehand.
* **Best Fit:** Primary deployment for National DoH, Provincial Departments, and NGO partners requiring professional, secure dashboards.

**R Shiny (8.45/10 – Very Good)**

* **Strengths:** Perfect R integration (models execute directly), full reproducibility, and open-source transparency. Ideal for technical demonstrations and academic presentations.
* **Weaknesses:** Security and scalability limitations on free tier. Requires Shiny Server Pro (R2,000+/year) for enterprise deployment. Less polished UI compared to Power BI without extensive custom styling.
* **Best Fit:** Technical demonstration tool for data science teams, university collaborators, and model validation.

**Streamlit (6.40/10 – Good)**

* **Strengths:** Rapid Python prototyping, free hosting, growing community.
* **Weaknesses:** Poor R integration (requires reticulate package with significant overhead). Team's models are already in R—rewriting in Python would duplicate effort. Limited security features.
* **Best Fit:** Not recommended for this project due to R-centric codebase.

**Flask (6.55/10 – Good)**

* **Strengths:** Maximum flexibility, RESTful API capabilities, scalable infrastructure.
* **Weaknesses:** Requires extensive web development (frontend, backend, deployment infrastructure). High time investment for UI design. Team lacks web dev expertise.
* **Best Fit:** Not recommended—development timeline exceeds Milestone 5 constraints.

## 3.4 Final Tool Selection and Justification

**Primary Deployment Tool: Power BI**

**Decision:** Power BI is selected as the **primary deployment platform** based on its weighted score (9.15/10) and alignment with stakeholder needs.

**Justification:**

1. **Stakeholder Alignment:** Milestone 1 identified National DoH, Provincial Health Departments, and municipalities as primary users. These organizations predominantly use Microsoft 365 ecosystems, making Power BI a natural fit (NDoH, 2024).
2. **Security and Compliance:** Public health data requires enterprise-grade security. Power BI provides:

* Single Sign-On (SSO) via Azure Active Directory
* Row-Level Security (RLS) for provincial data segregation
* Encryption at rest and in transit (AES-256)
* Audit logs for POPIA compliance (South Africa's Protection of Personal Information Act)

1. **Scalability:** Power BI Service supports 50+ concurrent users without performance degradation—critical for national rollout.
2. **User Experience:** Non-technical stakeholders (district managers, municipal planners) require intuitive interfaces. Power BI's drag-and-drop design, mobile apps, and professional templates meet this need.
3. **Deployment Speed:** CSV imports and pre-built visuals enable rapid dashboard creation within Milestone 5 timeline.

**Accepted Limitation:** Power BI cannot execute R models in real-time. This is mitigated by:

* Pre-computing predictions in R (using deployment\_export.R script)
* Exporting results to CSV/Excel for Power BI import
* Scheduling weekly data refresh cycles via Power BI Service

**Secondary Deployment Tool: R Shiny**

**Decision:** R Shiny is selected as a **supplementary technical demonstration tool**.

**Justification:**

1. **Methodological Transparency:** R Shiny allows reviewers (academics, peer groups, technical auditors) to interact directly with models—adjusting inputs and observing predictions in real-time.
2. **Reproducibility:** Open-source code and native R execution ensure full transparency of the CRISP-DM workflow, supporting academic scrutiny.
3. **Cost-Effectiveness:** Free hosting on shinyapps.io eliminates licensing costs for demonstration purposes.
4. **Future Iteration Support:** When additional survey years become available, R Shiny enables rapid prototyping and model retraining without waiting for Power BI redesigns.

**Accepted Limitation:** Shiny's security and scalability are insufficient for enterprise deployment. It serves as a **proof-of-concept** rather than the operational dashboard.

**Dual-Platform Strategy**

The recommended approach deploys **both platforms in parallel**, each serving distinct audiences:

|  |  |  |
| --- | --- | --- |
| Aspect | Power BI (Primary) | R Shiny (Secondary) |
| Audience | Policymakers, executives, field teams | Data scientists, researchers, technical reviewers |
| Purpose | Operational decision-support | Methodology demonstration and validation |
| Access | Power BI Service (authenticated users) | Public URL (shinyapps.io) |
| Data Refresh | Weekly scheduled refresh | Manual refresh for demos |
| Maintenance | Managed by IT/BI team (Person 3 monitoring) | Maintained by data science team |

**Consistency Mechanism:** Both platforms import from the same exported datasets (model\_metrics\_export.csv, feature\_importance\_export.csv), ensuring alignment between operational and demonstration outputs.

## 3.5 Prototype Implementation Evidence

To validate the recommended tool selection, functional prototypes were developed for both Power BI and R Shiny.

**Power BI Prototype Implementation**

**Script:** powerbi\_link.R

**Functionality:**

1. Reads Milestone 4 model performance metrics (model\_performance\_summary.csv)
2. Reads Random Forest feature importance (feature\_importance\_rf.csv)
3. Exports data in multiple formats:

* **CSV:** powerbi\_model\_metrics.csv, powerbi\_feature\_importance.csv
* **JSON:** powerbi\_data.json (for API integration)
* **Excel:** powerbi\_data.xlsx (multi-sheet workbook with Metrics, Features, and Metadata tabs)

**Validation Steps:**

* Successfully imported powerbi\_model\_metrics.csv into Power BI Desktop
* Created KPI cards displaying:
  + Best Model Accuracy: 54.55% (Random Forest)
  + Best Model AUC: 0.5323 (Logistic Regression)
* Built horizontal bar chart showing feature importance (Education\_Level, Water\_Access, Household\_Income ranked by Mean Decrease Gini)
* Added slicer for Model selection (filters between Logistic Regression, Decision Tree, Random Forest, Naïve Bayes)

**Screenshot Evidence:**

*(Note: Screenshots are included in the submitted .zip file as PowerBI\_import.png, PowerBI\_dashboard\_preview.png)*

**Figure 1: Power BI Data Import Success**  
Shows successful connection to powerbi\_model\_metrics.csv with all 4 model records loaded (Random Forest, Logistic Regression, Decision Tree, Naïve Bayes).

**Figure 2: Power BI Dashboard Preview**  
Displays KPI cards (Accuracy, AUC), feature importance bar chart, and model selection slicer. Demonstrates professional layout suitable for NDoH stakeholder presentations.

**Excel Multi-Sheet Export Validation**

The powerbi\_data.xlsx file contains three sheets:

1. **ModelMetrics Sheet:**
   * Columns: Model, Accuracy, Precision, Recall, F1\_Score, AUC
   * 4 rows (one per model)
   * Data types validated: Numeric (0-1 range for metrics)
2. **FeatureImportance Sheet:**
   * Columns: Feature, Importance
   * ~15 rows (top predictors from Random Forest)
   * Sorted descending by Importance score
3. **Metadata Sheet:**
   * Export Date, R Version, Script Name, Milestone Number
   * Provides traceability for governance (version control)

**Power BI Desktop Import Test:**

* File → Get Data → Excel → Select powerbi\_data.xlsx
* Successfully loaded all three sheets as separate tables
* Created relationships: ModelMetrics[Model] ← many-to-one → FeatureImportance[Model] (conceptual; actual relationship depends on data structure)

**JSON API Format (Future Integration)**

The powerbi\_data.json file provides a REST API-compatible format for potential future integrations with:

* Power BI streaming datasets (real-time updates)
* Azure Functions (serverless model retraining triggers)
* External dashboards (Tableau, Qlik) if stakeholder requirements change

**JSON Structure (Sample):**

json

{

"export\_metadata": {

"timestamp": "2025-10-13 14:30:00",

"milestone": "Milestone 5 - Deployment",

"r\_version": "4.3.1"

},

"model\_metrics": [

{

"Model": "Random Forest",

"Accuracy": 0.5455,

"AUC": 0.5080

}

],

"feature\_importance": [

{

"Feature": "Education\_Level",

"Importance": 0.2847

}

]

# 4. Process Review / Monitoring & Maintenance Plan

This section outlines the monitoring and maintenance plan for the Health and Demographic Patterns in South Africa (HDPSA) project. The goal is to ensure the deployed machine learning models, which provide interpretable insights for public-health stakeholders, remain accurate, reliable, and relevant over time. This plan addresses data inputs, post-deployment monitoring, and model updating thresholds.

## 4.1 Data input

For a model to be accurate and for it to be useful to the current situation new data needs to be added and new models need to be trained. New data should be taken from sources like the national census, and any other applicable data from Stats SA, WHO, UNICEF, IGME and more.

Each new dataset will be validated for completeness and consistency before integration. Once approved, the model will be retrained to reflect recent health and demographic trends.

## 4.2 Performance Metrics and Thresholds

A monitoring framework with defined metrics, thresholds, and owners will be used to track model performance and data relevancy.

The baseline performance for the approved models is an Accuracy of ~52% and an AUC of ~0.53. The thresholds below are set to detect a significant degradation from this established baseline, rather than an aspirational target.

|  |  |  |  |
| --- | --- | --- | --- |
| Metric | Threshold | Action | Owner |
| **AUC** | < 0.50 | Retrain model with new or existing data | Data Scientist |
| **Accuracy** | < 45% | Investigate potential data drift or bias | Project Lead |
| **Data Freshness** | `> 24 months | Acquire and integrate new survey data | Data Engineer |

## 4.3 Maintenance Schedule and Procedures

The monitoring schedule and procedures will consist of three parts

* **Monthly Review:** The `monitoring\_log.csv` will be reviewed monthly by the Project Lead to check for any anomalies or threshold breaches.
* **Quarterly Re-validation:** The model will be re-validated quarterly against a hold-out test set from the original data to check for performance consistency.
* **Retraining Trigger:** A model retraining process will be initiated if any threshold in the monitoring framework is breached.

All review activities are logged in the governance spreadsheet (model\_governance\_log.xlsx) to ensure transparency.

## 4.4 Automation Strategy

An R script (`model\_monitoring\_log.R`) will be used to automate the logging of performance metrics. This script will be run after each new data validation or prediction task. It appends a new entry to a central CSV log file, recording the model's performance over time. An example of how this can look can be found as (‘Milestone\_5\_Example\_automated\_logging.R’).

An example output will be the following:

timestamp,auc\_score,accuracy\_score,data\_freshness

2025-10-15 14:30:00,0.532,0.522,2016-12-31

2026-01-15 15:00:00,0.529,0.519,2016-12-31

2026-04-15 14:45:00,0.525,0.511,2016-12-31

This will provide a quick overview of when the log is. Quick performance metrics about the model and information on the data.

## 4.5 Governance and Version Control

Version control through the duration of the project has been handled through git. This allows for easy history of changes and the option to roll changes back. Throughout the duration of the project most commits or merges have been approved by the programmer responsible for the new section. Once the project enters its operational phase this should change so that only the most senior member may approve changes. This will be to ensure that the customer version is always in a functional state and as to not disrupt their experience.

## Governance will be documented though a spreadsheet. (`model\_governance\_log.xlsx`) will be maintained to track major decisions, model versions deployed, and the rationale for any changes, providing a human-readable history of the project.

## 4.6 Model Obsolescence

A model will be considered for retirement or a complete rebuild if:

* **Consistent Underperformance:** It fails to meet the minimum thresholds for two consecutive quarters.
* **Shift in Business Objectives:** Stakeholder needs evolve from demonstration to requiring high-accuracy predictions, which the current models cannot provide.
* **Fundamental Data Changes:** New data sources are introduced that are structurally different from the original DHS data.

# 5. Next Steps / Model Deployment & Documentation

## 5.1 Deployment Implementation Plan

## 5.2 Power BI Dashboard Design

## 5.3 Alternative Interface: R Shiny Demo

## 5.4 Ethical Considerations and Data Privacy

## 5.5 User Guide and Documentation

# 6. Integration and Final Deployment Strategy

## 6.1 Unified Deployment Roadmap

## 6.2 Stakeholder Communication Plan

## 6.3 Training and Change Management

## 6.4 Success Metrics and KPIs

# 7. Ethical and Privacy Implications

## 7.1 Data Privacy Compliance

## 7.2 Model Fairness and Bias Mitigation

## 7.3 Transparency and Accountability

# 8. Conclusion and Recommendations

## 8.1 Key Findings Summary

## 8.2 Deployment Readiness Assessment

## 8.3 Future Enhancements

# 9. References

Ahmad, A., 2023. *Introduction to Power BI: Data Visualization for Decision-Making.* 2nd ed. London: Packt Publishing.

Microsoft, 2024. *Power BI Documentation: Business Intelligence for Enterprise.* [online] Available at: https://learn.microsoft.com/power-bi/ [Accessed 16 October 2025].

National Department of Health (NDoH), 2024. *Annual Health Statistics and Policy Brief: Digital Transformation in Public Health.* Pretoria: Government Printer.

Posit (formerly RStudio), 2024. *R Shiny User Guide: Building Interactive Web Apps in R.* [online] Available at: https://shiny.posit.co/r/ [Accessed 16 October 2025].

Snowflake Inc., 2024. *Streamlit Developer Documentation: Build and Share Data Apps.* [online] Available at: https://docs.streamlit.io/ [Accessed 16 October 2025].

Statistics South Africa (Stats SA), 2024. *Demographic and Health Survey Indicators Report 2024.* Pretoria: Statistics South Africa.

The R Foundation, 2025. *R: A Language and Environment for Statistical Computing.* Vienna: R Foundation for Statistical Computing.

Wirth, R. and Hipp, J., 2000. *CRISP-DM: Towards a Standard Process Model for Data Mining.* In: *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining (PKDD 2000).* pp. 29–39.

# 10. Appendices

## Appendix A – Deployment Export Code

## Appendix B – Deployment Flow Diagram

## Appendix C – Model Performance Summary

## Appendix D – Stakeholder Sign-off Documentation