# **RTV SENIOR DATA SCIENTIST- TECHNICAL ASSESSMENT**

# **About Us**

At Raising The Village (RTV), we are dedicated to eradicating ultra-poverty in Sub-Saharan Africa. As a dynamic, rapidly growing international development organization, we’ve assembled a team of over 250 passionate individuals in Uganda, alongside 17 professionals in North America and 15 in Rwanda. Together, we are committed to elevating communities out of ultra-poverty by implementing innovative solutions and leveraging advanced data analytics to drive impact.

To date, our holistic approach has positively impacted over 1 million lives since 2012, and we’re poised to achieve even greater milestones, aiming to assist 1 million individuals annually by 2027. Our growth and success are fueled by the invaluable support of global partners who share our vision of sustainable change. Learn more about our impactful programs at [www.raisingthevillage.org](http://www.raisingthevillage.org)

The **VENN department** is the data and technology backbone of our organization, connecting advanced analytics and custom software tools with field implementation to ensure data-informed decision-making at every level.

# **Aim of the assessment**

This assessment evaluates your Technical Proficiency, Innovation, Leadership, Communication, and Contextual Relevance by challenging you to design and implement a practical, end-to-end solution using **DataScientist\_01\_Assessment** data. You will be required to develop a predictive model to identify household vulnerability levels and design a data pipeline that supports the receipt and integration of new AHS data over time. Through this exercise, we aim to understand how well you can apply advanced data science and engineering skills to real-world development challenges, while aligning with Raising The Village’s mission to deliver impactful, data-driven solutions for last-mile communities.

## **Assessment Overview**

* Time: 7 Hours
* Points:60
* For any follow-up questions, please contact – 0775648275/0705945524

# **Submission Requirements**

* GitHub repository with an organized folder structure

# **Machine Learning- Model Development, Data Engineering, and Integration**

**Scenario:**  
Design and implement a scalable, interpretable machine learning solution to predict household vulnerability levels (High, Moderate, Low) using Annual Household Survey (AHS) data. The system should support continuous learning by updating predictions as new data arrives quarterly or annually. Deliver a fully integrated pipeline that includes data processing, model training, inference API, and real-time monitoring. The final solution must provide accurate predictions alongside actionable insights to enable data-driven, timely decision-making for targeted household support.

**Dataset Description**

* The dataset comprises household demographics, program participation details, geographic data, and outcome metrics, spanning multiple villages and years.
* Data dictionary
* Create a new variable called ‘*ProgressStatus*’ based on the ‘HHIncome+Consumption+Residues/Day’ variable, categorizing the values as follows: On Track" for values >= 2.15, "At Risk" for values >= 1.77, "Struggling" for values >= 1.25, and "Severely Struggling" for values below 1.25

## **Part A: Predictive Modeling with ‘DataScientist\_01\_Assessment’ Data- 25 Points**

* Which features most significantly influence the model's prediction of a household being “At Risk” or “Struggling,” and how were these identified during model building?
* Are the model explanations consistent across sub-regions, gender groups, or household sizes?
* How does the model treat borderline cases (e.g., predicted ProgressStatus near the cutoff between “At Risk” and “Struggling”)?
* Can field officers confidently trust individual predictions made by the model?
* How should field officers combine the model's **ProgressStatus** output with interpretability insights during household visits?
* Write a 2-page summary explaining your modelling choices, key findings, and insights for RTV

## **Part B: Data Engineering for Receiving New Data- 15 Points**

* Design a simple ETL pipeline to **automatically receive and process new data** from field uploads (e.g., quarterly updates)
* Pipeline must include:
  + Ingestion
  + Cleaning and transformation logic
  + Secure storage
  + Logic for retraining ML models and updating outputs
* Provide sample code for one pipeline component and an architecture diagram
* Briefly describe your tech stack

## **Part C: Product Integration- 20 Points**

Scenario: Raising The Village (RTV) is building a **WorkMate mobile application** to support field officers in last-mile communities. The app is intended to work in **low-bandwidth or offline environments** and should provide **actionable household insights** based on predictive models.

Model Integration Demonstration Requirements

1. **Model Selection & Packaging:** Select an existing ML model that you built and demonstrate how it is packaged for deployment using an appropriate method suitable for mobile or backend inference in resource-constrained environments.
2. **Integration Architecture Design:** Provide a clear architecture diagram showing how the selected model integrates with the **WorkMate mobile app**, **backend/cloud infrastructure**, and **field devices**. The architecture should include provisions for **offline capability** and data synchronisation when connectivity is restored.
3. **Field Officer User Flow:** Illustrate a step-by-step user interaction flow, from **data input** by the field officer to **real-time prediction** and **suggested action**, including how the app handles **offline fallback** with **local storage and background syncing** once a connection is available.
4. **Code Demonstration:** Show working code that takes household-level input, generates a prediction using the packaged model, and stores the result securely for later upload or real-time use, depending on connectivity, demonstrating how the solution fits into the broader system.

\*\*\*\*\*\*\*\*\*\*\*Good luck\*\*\*\*\*\*\*\*\*\*\*