**VPBank Technology Hackathon 2025**

General Brief

|  |  |
| --- | --- |
| **Challenge Statement** | 24\_EDA Challenge\_Data Accuracy Test Rule Engine |
| **Team Name** | Nhóm 70 |

**Team Members**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Full Name** | **Role** | **Email Address** | **School Name**  **(if applicable)** | **Faculty / Area of Study** | **LinkedIn Profile URL** |
| Nguyễn Hoài Nam | Trưởng nhóm | namnguyen7230@gmail.com |  | Công nghệ thông tin và truyền thông | https://www.linkedin.com/in/nam-nguyen-32b6b5201/ |
| Nguyễn Huy Hoàng |  |  |  |  |  |
| Lê Nguyên Phương Linh |  |  |  |  |  |
| Member 4 |  |  |  |  |  |
| Member 5 |  |  |  |  |  |

**Content Outline**

|  |  |
| --- | --- |
| Page No. | |
| **Solutions Introduction** | 3 |
| **Impact of Solution** | 3 |
| **Deep Dive into Solution** | 4 |
| **Architecture of Solution** | 6 |

**Solutions Introduction**

**Tổng quan:** Một cloud-native, lightweight Data Accuracy Test Rule Engine tự động hóa việc validation business logic trên nhiều database tables sử dụng hệ thống định nghĩa rules dựa trên JSON đơn giản với khả năng real-time execution.

* **Main Features:**
  + **JSON Rule Builder**: Web interface đơn giản để tạo và quản lý validation rules mà không cần coding
  + **Multi-Database Connector**: Unified interface hỗ trợ PostgreSQL, MySQL, và AWS RDS với khả năng mở rộng dễ dàng cho các databases khác
  + **Real-time Validation Engine**: Thực thi rules nhanh chóng với results có sẵn trong vài giây, hỗ trợ cả batch và streaming data validation
* **Workflow hoạt động:**

1. **Rule Definition**: Users định nghĩa validation rules thông qua web interface sử dụng JSON templates
2. **Rule Compilation**: Engine chuyển đổi JSON rules thành optimized SQL queries cho target database
3. **Execution**: Rules được execute trên live data với results được cache để tăng performance
4. **Reporting**: Real-time dashboard hiển thị validation results với detailed pass/fail reports và trend analysis

**Impact of Solution**

**Benefits cho Society/Target Audience**

* **Giảm thiểu Data Quality Issues**: Ngăn chặn bad data lan truyền qua business systems, cải thiện accuracy trong decision-making
* **Cost Savings**: Loại bỏ manual testing efforts và giảm data-related incidents lên đến 80%
* **Faster Time-to-Market**: Data teams có thể deploy validation rules trong vài phút thay vì weeks phát triển custom scripts

**Competitive Advantages**

* **Tại sao solution này superior?**
  + **Zero-Code Approach**: Business users có thể tạo complex validation rules mà không cần SQL knowledge
  + **Cloud-Native Performance**: Tận dụng AWS serverless architecture để automatic scaling và cost optimization
* **Unique Selling Points:**
  + **Template-Based Rule Creation**: Pre-built templates cho common validation scenarios (email formats, date ranges, statistical comparisons)
  + **Visual Rule Builder**: Drag-and-drop interface để tạo complex Boolean logic combinations

**Market Differentiation**

* **Comparison với existing solutions:**
  + Khác với traditional ETL tools đòi hỏi expensive licenses và complex setup, solution của chúng tôi là serverless và pay-per-use
  + So với custom-built validation scripts, solution cung cấp standardization, reusability và maintenance-free operation
  + Superior hơn database-only solutions bằng cách cung cấp cross-database validation và business-friendly interfaces

**Deep Dive into Solution**

**Detailed Solution Architecture**

Solution bao gồm ba main layers:

* **Presentation Layer**: React-based web application cho rule management và result visualization
* **Processing Layer**: Python-based rule engine chạy trên AWS Lambda cho serverless execution
* **Data Layer**: Multi-database connectivity với intelligent query optimization và result caching

**Technical Implementation**

* **Core Components:**
  + **Rule Parser**: Chuyển đổi JSON rule definitions thành executable SQL với parameter binding và optimization
  + **Database Abstraction Layer**: Unified interface hỗ trợ multiple database engines với connection pooling và failover
  + **Execution Engine**: Quản lý rule scheduling, parallel execution và result aggregation với built-in retry mechanisms

**3Supporting Features**

* **Rule Versioning**: Track changes của rules với rollback capabilities
* **Scheduled Execution**: Automated rule execution với configurable frequency (hourly, daily, weekly)
* **Alert System**: Email/Slack notifications khi validation rules fail với customizable thresholds
* **API Integration**: RESTful APIs để integration với existing data pipelines và CI/CD systems

**Architecture of Solution**

**AWS Services Integration**

**Primary AWS Services được sử dụng:**

| **Service** | **Purpose** | **Implementation Details** |
| --- | --- | --- |
| AWS Lambda | Rule Execution Engine | Serverless functions để executing validation rules với automatic scaling |
| Amazon RDS/Aurora | Database Connectivity | Managed database services để connecting với various database engines |
| DynamoDB | Rule Repository | NoSQL storage cho rule definitions, metadata và execution history |
| S3 | Data Storage | Storage cho large result sets, logs và backup data |
| CloudWatch | Monitoring & Logging | Real-time monitoring của rule execution performance và error tracking |
| API Gateway | REST API Interface | Secure API endpoints cho external integrations và mobile applications |
| Amplify | Web Application Hosting | Hosting cho React-based rule management interface |

**Service Integration Strategy**

* **Cách services work together:**
  + **API Gateway + Lambda**: Cung cấp scalable REST API cho rule management và execution triggers
  + **Lambda + RDS**: Secure database connections với connection pooling cho optimal performance
  + **DynamoDB + S3**: Hot storage cho active rules trong DynamoDB, cold storage cho historical data trong S3
  + **CloudWatch Integration**: Comprehensive monitoring across tất cả services với custom metrics và alarms

**Architecture Diagram**

**A diagram of a cloud computing process

AI-generated content may be incorrect.**

**Scalability và Performance**

* **Scalability considerations:**
  + **Serverless Architecture**: Automatic scaling từ 0 đến thousands concurrent executions
  + **Database Connection Pooling**: Efficient database resource utilization với RDS Proxy
  + **Caching Strategy**: Multi-level caching với DynamoDB DAX và Lambda memory caching
* **Performance optimizations:**
  + **Query Optimization**: Intelligent SQL generation với index recommendations
  + **Parallel Execution**: Concurrent rule execution cho large rule sets
  + **Result Streaming**: Real-time result updates sử dụng WebSocket connections through API Gateway
  + **Edge Caching**: CloudFront distribution cho global low-latency access