|  |  |
| --- | --- |
|  |  |
|  |  |



Automatic Laundry Solutions SmartOps Platform  
Business Requirement Document v1.0

|  |  |
| --- | --- |
|  | |
| 353 Oates Road | |
| Mooresville, NC 28117 | |
| **PHONE:** 704.658.1707 |
| **FAX:** 704.658.9876 |
| biztechnologysolutions.com | |

Contents

[ Document Control 4](#_Toc190347036)

[ Executive Summary 6](#_Toc190347037)

[ Scope 7](#_Toc190347038)

[ In-Scope Functional Requirements 10](#_Toc190347039)

[ Reporting Requirements 13](#_Toc190347040)

[ Out of Scope 13](#_Toc190347041)

[ Contextual Diagram 15](#_Toc190347042)

[ Conceptual Data Model 17](#_Toc190347043)

[ Non-Functional Requirements 18](#_Toc190347044)

[ Operational Requirements 19](#_Toc190347045)

[ UI and Interface Requirements 19](#_Toc190347046)

[Overview 19](#_Toc190347047)

[Key UI and Interface Requirements 19](#_Toc190347048)

[1. General UI Design 19](#_Toc190347049)

[2. Dashboard & Home Page 19](#_Toc190347050)

[3. Navigation & Layout 20](#_Toc190347051)

[4. Data Presentation & Visualization 20](#_Toc190347052)

[5. User Management & Access Control 20](#_Toc190347053)

[6. API Integration & Data Flow 20](#_Toc190347054)

[7. Alerts & Notifications 20](#_Toc190347055)

[8. Performance & Usability 20](#_Toc190347056)

[Key Features: 21](#_Toc190347057)

[ Development Strategy 25](#_Toc190347058)

[Machine Management (MM) 25](#_Toc190347059)

[Reporting & Analysis (RA) 26](#_Toc190347060)

[Real-Time Monitoring (RTM) 27](#_Toc190347061)

[Visualizations (VIS) 27](#_Toc190347062)

[ Deliverables and Exit Criteria 27](#_Toc190347063)

[ Staffing Model 27](#_Toc190347064)

[ Assumptions 27](#_Toc190347065)

[ Reference and Related Documents 28](#_Toc190347066)

* Document Control

|  |  |
| --- | --- |
| **Document Name** |  |
| Original Author(s) | Prasanna Yenugula |
| Current Revision Author(s) |  |

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Reviewers** | **Reason** |
| 1.0 | 01/06/2025 | Prasanna Yenugula |  | Initial Draft for Functional and Non – Functional Requirements |
|  |  |  |  |  |
|  |  |  |  |  |

**Revision Process**

The contents of this document are of special interest to architects and developers. Those team members should be notified when any updates are made, or related documents are created.

**Intended Audience**

This document is intended for the following audience:

|  |
| --- |
| **Audience Roles** |
| Application Solution Architect? Lead Accountant? |
|  |
|  |

* Executive Summary

The **Machine Management (MM)** and **Reporting & Analysis (RA)** systems are designed to provide a comprehensive solution for monitoring, managing, and optimizing machine performance, service alerts, and operational efficiency. These systems are crucial for ensuring that machine uptime is maximized, service issues are addressed promptly, and actionable insights are provided to various stakeholders, from system administrators to executives.

**Machine Management (MM)** focuses on tracking machine health, managing service alerts, and automating key processes such as service request creation and alert resolution. The system provides tools for technicians, lease managers, network administrators, and other stakeholders to effectively monitor machine usage, identify underutilized assets, track machine lifecycle, and measure performance metrics. By allowing users to view, filter, and analyze service data at the campus, lease, and location levels, MM enhances operational efficiency and enables proactive maintenance.

**Reporting & Analysis (RA)** serves as the backbone for strategic decision-making, providing detailed insights into machine performance, service trends, energy consumption, and sustainability efforts. With capabilities to generate customized reports and dashboards, the system empowers stakeholders to track key performance indicators (KPIs), monitor SLA compliance, and predict future maintenance needs. It supports environmental initiatives by offering detailed consumption reports and enhances team productivity by providing reports on unresolved service alerts and error types.

The system's **Functional Requirements** include real-time service alert tracking, automated maintenance scheduling, detailed error reporting, and customizable analytics. **Non-Functional Requirements** ensure the system performs optimally at scale, with fast processing times, 99.9% uptime, high data security, and the ability to integrate seamlessly with other enterprise systems.

In summary, the **MM** and **RA** systems will enhance operational oversight, improve service response times, drive proactive maintenance, and support data-driven decision-making across all levels of the organization. These systems are designed to provide value by optimizing machine performance, reducing downtime, improving user experience, and supporting long-term sustainability and operational goals.

.

* Scope

The Machine Management (MM) and Reporting & Analysis (RA) systems will be developed to streamline machine performance monitoring, service alert management, and reporting functions within the organization. These systems are designed to serve various user roles, including system administrators, technicians, managers, analysts, and executives, by providing real-time machine data, trend analysis, automated processes, and detailed reporting capabilities.

Scope of Machine Management (MM)

1. Service Alert Management:
   * The MM system will allow administrators, technicians, and managers to view, track, and manage service alerts based on various criteria such as campus, location, and alert severity.
   * Alerts will be filtered by campus, service type, and date range for efficient analysis and troubleshooting.
   * System will support automated creation of service requests for certain alert conditions (e.g., repeated errors or high-priority issues).
2. Machine Monitoring:
   * Technicians will be able to review real-time machine status, including auto-cleared and manually cleared alerts, to ensure machines are functioning optimally.
   * Machine uptime and performance metrics will be tracked, and usage trends will be analyzed at the lease level and by location.
3. Service History & Lifecycle Tracking:
   * Machines will be tracked by type, model, manufacturer, and lifecycle stage, including installation, maintenance history, and end-of-life.
   * Alerts and service logs will be linked to machine records for a comprehensive service history view.
4. Performance & Usage Analysis:
   * The system will allow users to analyze machine performance trends, including tracking error patterns, uptime, and service resolution times.
   * The system will provide reports on machine cycles, utilized machines, and recurring errors for proactive maintenance scheduling.
5. Proactive Maintenance & Alerts:
   * Users will receive proactive alerts on machines that are at risk of failure, with recommendations for maintenance based on usage or error trends.
   * Technicians and operations managers will be notified of high-cycle machines or recurring issues to prevent downtime.
6. Role-Based Access & Security:
   * The MM system will ensure role-based access control for different user types, ensuring that users only have access to relevant data.
   * All data related to machine status, service records, and usage will be encrypted and secured following industry best practices.

Scope of Reporting & Analysis (RA)

1. KPI Dashboards & Data Visualizations:
   * The RA system will provide interactive dashboards to present key performance indicators (KPIs) such as machine uptime, service alert resolution times, and average cycles.
   * Dashboards will be customizable based on user roles (e.g., executives, service managers, performance analysts).
2. Service & Machine Performance Reports:
   * The system will allow the generation of detailed reports on service alerts, error types, machine usage, and uptime.
   * Reports will be customizable to show data by location, lease, date range, and service type, helping users analyze trends and identify improvement areas.
3. Compliance & SLA Reporting:
   * Compliance officers will be able to review machine uptime reports to ensure service-level agreements (SLAs) are being met across various locations and leases.
   * Reports will be available for auditing and regulatory purposes, ensuring compliance with industry standards.
4. Environmental & Sustainability Reporting:
   * The system will provide reports on water and energy consumption by location, supporting sustainability goals.
   * These reports will help stakeholders monitor resource usage and track environmental impact for decision-making.
5. Data Export & Custom Reporting:
   * Business analysts will have the ability to export data in various formats (CSV, Excel, PDF) for further analysis using external tools.
   * Custom reporting capabilities will allow users to create tailored reports based on specific metrics, time periods, or data sets.
6. Historical Data Analysis:
   * The RA system will support the analysis of historical service data, enabling users to identify long-term trends, recurring errors, and opportunities for system optimization.
   * Users will be able to review trends related to service alerts, machine performance, and usage, helping to predict future maintenance needs.
7. Real-Time Data Processing & Reporting:
   * Both systems will provide real-time reporting on machine status, service alerts, and performance metrics, allowing for immediate action and issue resolution.
8. Role-Based Reporting Access:
   * Users will only have access to relevant reports and data based on their role in the organization, ensuring secure and appropriate data distribution.

Scope Exclusions

1. Hardware Implementation:
   * The scope does not include any hardware changes or the physical setup of machines. The system will only manage and track existing machine data.
2. Third-Party Integrations:
   * While the system will support integrations with existing CRM and ERP systems, any integration with third-party hardware or software outside of the defined platforms is outside the scope.
3. Mobile Application:
   * A dedicated mobile application for accessing machine data or service alerts is not included in the current scope. The system will be web-based for all user interactions.
4. External Data Sources:
   * The scope does not include integrating with external data sources beyond the current system's requirements for machine data, service alerts, and performance metrics.

* In-Scope Functional Requirements

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions.

Key Components:

* **Machine Management (MM):** This component focuses on tracking the health and performance of machines, handling service alerts, monitoring machine usage, and automating service requests.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req Type** | **Req. ID** | **Requirement Description** | **Priority** | **Comment** |
| WEB ADMIN | MM 1.1 | Service Alert by Campus: System administrators should be able to view service alerts filtered by campus or location. | High |  |
| WEB USER | MM 1.2 | Search by Service Type: Technicians need to search service issues by service type for effective troubleshooting and prioritization | High |  |
| WEB USER | MM 1.3 | Filter Alerts by Date Range: Lease managers must filter service alerts by date range at the lease level to analyze incidents over time. | High |  |
| WEB ADMIN | MM 1.4 | Recurring Event Trend Analysis: Analysts should see recurring service events over time to identify patterns and possible root causes | High |  |
| WEB USER | MM 1.5 | Alert to Resolution Time Measurement: Operations managers need to track the time difference from alert creation to resolution for efficiency. | High |  |
| WEB USER | MM 1.6 | Auto-Clear Report Review: Technicians should be able to review reports on fatal errors, refresh events, and reboots to confirm machine stability. | High |  |
| REPORT | MM 1.7 | Machine Uptime Metrics: Performance analysts must be able to view and track machine uptime metrics at the lease level for service reliability | High |  |
| WEB USER | MM 1.8 | Track Underutilized Machines: Service managers should be able to identify machines with no activity over a specified time period. | High |  |
| REPORT | MM 1.9 | Machine Connectivity Reporting: Network administrators must have the ability to generate reports on machine connectivity to ensure consistent availability. | High |  |
| WEB USER | MM 2.0 | Track Machine Cycles: Data analysts should monitor the average cycles of machines over different periods for performance analysis. | High |  |
| WEB USER | MM 2.1 | Identify High-Cycle Machines: Service coordinators must get alerts for machines exceeding five cycles in a day, indicating maintenance needs. | High |  |
| WEB USER | MM 2.2 | Proactive Maintenance Alerts: Operations managers should receive alerts for recurring errors to schedule preventive maintenance. | High |  |
| REPORT | MM 2.3 | Machine Error Reporting: Data analysts should generate reports on machine errors categorized by type, frequency, and resolution time | High |  |
| WEB USER | MM 2.4 | Machine Usage by Lease/Location: Lease managers should be able to view machine usage trends by lease and location. | High |  |
| WEB USER | MM 2.5 | Auto vs. Manual Alert Clearing: Technicians need the ability to distinguish between auto-cleared and manually cleared alerts | High |  |
| REPORT | MM 2.6 | Machine Inventory Tracking: Track machine types, models, and manufacturers to assess performance trends. | High |  |
| SYSTEM | MM 2.7 | Lease-Location Mapping: Support mapping of leases to specific locations to ensure proper coordination and management. | High |  |
| SYSTEM | MM 2.8 | Service Request Automation: Automate service request creation when specific service issues or errors are identified. | High |  |
| REPORT | MM 2.9 | Real-Time Machine Status: Provide real-time updates on the operational status of machines. | High |  |
| SYSTEM | MM 3.0 | Machine Lifecycle Tracking: Track the installation, maintenance, and end-of-life of each machine. | High |  |

* Reporting Requirements

**Reporting & Analysis (RA):** This component enables the generation of detailed reports and analysis on various aspects of machine performance, service alerts, energy consumption, and sustainability, helping stakeholders optimize operations and compliance.

**Reporting & Analysis (RA)**

1. **Environmental Reports**: Sustainability officers need to generate reports on water and energy consumption by location.
2. **Monthly Maintenance & Sustainability Reports**: Property managers should receive automated monthly reports regarding maintenance and sustainability.
3. **Unresolved Service Alerts by Location**: Service managers should be able to track unresolved service alerts by location.
4. **Error Type Trend Analysis**: Analysts should analyze error type trends to anticipate future maintenance needs.
5. **Machine Uptime SLA Compliance**: Compliance officers must verify machine uptime against SLA targets to ensure service compliance.
6. **Machine Availability Monitoring**: Facilities managers should track machine availability by time of day to optimize service coverage.
7. **Executive KPI Dashboards**: Executives need high-level KPIs such as uptime, cycles, and service response times presented on a dashboard.
8. **Data Export for Custom Reporting:** Business analysts need the ability to export data for more customized reporting and third-party analysis

* Out of Scope

In project management, “out of scope” means anything that is outside the parameters of an initiative. At the beginning of a project, the scope is established in documents like the scope statement. It clarifies the work and deliveries of a project, setting out the expectations for both parties.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Req Type** | **Req. ID** | **Requirement Description** | **Priority** | **Comment** |
| SYSTEM | 2.4.1 | All Business data stored in the SQL Server |  |  |
| SYSTEM | 2.4.2 | SQL Server improvements are not part of this project implementation |  |  |
| SYSTEM | 2.4.3 | Development of API is considered as out of scope requirements. |  |  |
| SYSTEM | 2.4.4 | It is assumed that more than 90% of API exists for the business data |  |  |
| SYSTEM | 2.4.5 | Any GAP found in the API; ALS team will be responsible for building the missing API for the portal Or Provide access to data sources |  |  |

* Contextual Diagram  
  A diagram of a mathematical problem

  Description automatically generated with medium confidence

This diagram illustrates the flow of real-time machine data, performance tracking, usage reports, and service alerts across various stakeholders within an organization. It is divided into two main sections:

1. Machine Management (Left - Blue Box)
2. Reporting & Analysis (Right - Green Box)

Each section highlights the data flow between system components and different user roles.

1. **Machine Management (Blue Section)**

This section focuses on real-time machine monitoring, performance tracking, and service alerts.

Key Features & Data Flow

* Real-Time Machine Status → Used by Service Managers, Technicians, and Network Administrators to monitor machine health and operational status.
* Connectivity Reporting → Shared with Executives and Network Administrators to analyze connectivity and uptime metrics.
* Machine Lifecycle Tracking → Helps Technicians and Data Analysts assess machine longevity and maintenance schedules.
* Machine Performance & Usage → Provides insights into Facilities Managers, Lease Managers, and Data Analysts for trend analysis and forecasting.
* Service Alerts & Request Automation → Automated service requests and alerts are shared with System Administrators and Operations Managers, ensuring proactive maintenance.

2. **Reporting & Analysis (Green Section)**

This section is focused on data-driven insights, reporting, and trend analysis.

Key Features & Data Flow

* KPI Dashboards → Used by Executives and Compliance Officers for high-level performance monitoring.
* Data Export & Trend Analysis → Helps Data Analysts and Facilities Managers in deriving performance insights.
* Environmental Reports → Captures machine efficiency and environmental impact for Facilities Managers.
* Report Generation & Monthly Maintenance Reports → Provides Lease Managers and Operations Managers with detailed reports for decision-making and planning.

**Key Users & Their Responsibilities**

1. Compliance Officers → Monitor regulatory compliance using real-time updates.
2. Executives → Analyze business impact via connectivity metrics & KPI dashboards.
3. Network Administrators → Manage connectivity and data flow.
4. Service Managers & Technicians → Maintain machine lifecycle and performance.
5. Data Analysts → Process performance data for reporting.
6. Facilities & Lease Managers → Evaluate machine usage trends.
7. System Administrators & Operations Managers → Oversee service alerts & automated processes.

* Conceptual Data Model

A conceptual data model is a high-level description of informational needs underlying the design of a database. It typically includes only the main concepts and the main relationships among them. Typically this is a first-cut model, with insufficient detail to build an actual database.

* Non-Functional Requirements

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. They are contrasted with functional requirements that define specific behavior or functions.

**Machine Management (MM)**

1. **Performance**: The system must ensure that service alert data, machine status, and reports are processed and available within 5 seconds.
2. **Scalability**: The system should be able to handle thousands of machines and service alerts without performance degradation.
3. **Security**: All data must be encrypted using industry-standard encryption protocols, such as AES-256, and comply with relevant data protection regulations.
4. **Availability**: The system should maintain 99.9% uptime to ensure continuous access to data and alerts.
5. **Usability**: The user interface should be intuitive, with clear navigation and filtering options for technicians and managers.
6. **Data Retention**: The system must retain service alert and machine data for at least 5 years for historical analysis.
7. **Audit Logging**: The system should maintain an audit trail of user actions (viewing, updating, or clearing alerts) for accountability and traceability.

**Reporting & Analysis (RA)**

1. **Performance**: Report generation and dashboards must be delivered within 10 seconds, even when processing large datasets.
2. **Security**: Reporting features should have role-based access control, restricting sensitive data to authorized users only.
3. **Data Accuracy**: All generated reports should reflect near real-time data with a maximum delay of 5 minutes.
4. **Scalability**: The reporting system must scale to handle increased user access and large data volumes without slowing down report generation.
5. **Customization**: The system should allow users to customize reports and dashboards to match their specific analytical needs.
6. **Compatibility**: The system must support exporting reports in formats such as CSV, Excel, and PDF for external use and analysis.
7. **Internationalization**: The system should support multiple languages and locales for international users.
8. **High Availability**: Reporting tools and dashboards must maintain 99.9% up time to ensure availability during peak usage periods.

* Operational Requirements

Requirement that specifies an operation or behavior that a system or part of a system must perform.

* UI and Interface Requirements

**Overview**

The **UI and Interface** design aims to provide a seamless, efficient, and responsive experience for users interacting with the **Machine Management and Reporting System**. The interface must support various roles, including lease managers, technicians, compliance officers, and administrators, ensuring accessibility, usability, and real-time data visualization.

**Key UI and Interface Requirements**

**1. General UI Design**

* **Modern, responsive, and intuitive design** that works across desktops, tablets, and mobile devices.
* **Consistent theme and layout**, ensuring a unified experience across different modules.
* **Standardized UI components** (buttons, dropdowns, modals) for a smooth workflow.

**2. Dashboard & Home Page**

* **Interactive dashboard** displaying key system insights:
  + Machine status and performance
  + Lease and location-based trends
  + Service alerts and maintenance updates
* **Real-time updates** from API calls to display accurate data.
* **Customizable widgets** allow users to personalize their dashboard experience.
* **Heat maps and graphs** for visual data representation.

**3. Navigation & Layout**

* **Left-side navigation panel** with expandable/collapsible menu options.
* **Top navigation bar** for quick access to profile settings, notifications, and search.
* **Breadcrumb navigation** enhances usability and maintains page context.
* **Keyboard shortcuts** for faster workflow and navigation.

**4. Data Presentation & Visualization**

* **Tabular views with filtering, sorting, and pagination** for managing large datasets efficiently.
* **Dynamic charts and graphs** for analyzing machine usage, lease trends, and performance metrics.
* **Map-based UI** to display machine locations with interactive features.
* **Color-coded indicators** for machine status, alerts, and lease activity.

**5. User Management & Access Control**

* **Role-based access control (RBAC)** ensuring restricted access to specific modules based on user roles.
* **Admin dashboard** to manage user accounts, permissions, and system settings.
* **Multi-factor authentication (MFA)** and session timeout features for security.
* **Audit log tracking** for user activities and system interactions.

**6. API Integration & Data Flow**

* **Seamless API integration** to fetch real-time machine data.
* **Asynchronous API calls** to ensure a smooth UI experience without blocking interactions.
* **Error handling and validation messages** for failed API responses.
* **Data caching mechanisms** to enhance performance and reduce redundant API calls.

**7. Alerts & Notifications**

* **Real-time notifications** for system alerts, lease updates, and maintenance reminders.
* **Email and in-app alerts** for critical events.
* **Customizable alert settings** for different user roles.

**8. Performance & Usability**

* **Optimized UI for fast loading speeds** and smooth user interactions.
* **Auto-refresh features** for dynamic data updates.
* **Dark mode and accessibility options** to support diverse user needs.
* **Multilingual support** for a global user base.

**Home Page** serves as the central hub for users to access key features, monitor system status, and navigate seamlessly through the platform. It provides a **dashboard-style interface** with real-time insights, quick actions, and essential system metrics.

**Key Features:**

**1. Dashboard Overview**

* Displays key metrics such as machine usage, active leases, and service alerts.
* Provides a **summary view** of machine performance, maintenance schedules, and recent activity.

A screenshot of a computer

Description automatically generated

**2. Quick Access Navigation**

* Links to critical sections like **Machine Management, Reports & Analytics, Lease Management, and User Administration**.

A screenshot of a computer

AI-generated content may be incorrect.

**3. Machine Locations & Map View**

* Interactive **map visualization** displaying machine locations.
* Ability to filter machines by **status, location, and lease assignment**.
* Provides **real-time updates** on machine connectivity and operational health.

A map with a location on it

AI-generated content may be incorrect.

**4. Alerts & Notifications**

* Displays **service alerts**, maintenance reminders, and system updates.
* Helps users **stay proactive** in resolving issues before they escalate.

A screenshot of a computer

AI-generated content may be incorrect.

**5. Reporting & Data Insights**

* Access to reports on **machine usage, performance trends, and lease data**.
* Exportable data for further analysis in external tools.

A screenshot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

**6. Search & Filter Options**

* Allows users to **search for specific machines, leases, or reports**.
* Filters by criteria such as **location, status, and recent activity**.

**A screenshot of a computer

AI-generated content may be incorrect.**

* Development Strategy

For implementing the above functional requirements using API calls, we will need robust architecture that enables data exchange between systems (e.g., service management, machine monitoring, and reporting tools). Here's the **Development Strategy** broken down for the functional requirements using APIs:

## Machine Management (MM)

In the realm of Machine Management (MM), various user stories have been identified that encompass the specific needs of different roles interacting with the system. Each user story outlines distinct functionalities and the corresponding API requirements necessary to fulfill these roles effectively.

**System Administrator**

User Story: As a system administrator, I want to view service alerts by campus so that I can identify and address issues specific to a location.  
Functionality: This functionality allows the administrator to access and monitor service alerts across different campuses, facilitating targeted interventions.  
API Requirements:

* GET All Machines at a Location
* GET Machine Error Details

**Technician**

User Story: As a technician, I want to search for service issues by service type to streamline troubleshooting and task prioritization.  
Functionality: This feature enables technicians to filter service issues based on their type, allowing for more efficient problem-solving.  
API Requirements:

* Currently, there is no specific API for service types, but it may be possible to retrieve issues by other parameters.

**Lease Manager**

User Story: As a lease manager, I want to filter service alerts by date range at the lease level to analyze historical incidents.  
Functionality: This enables lease managers to view and analyze service alerts over specified timeframes, enhancing their ability to manage lease performance.  
API Requirements:

* GET Machine Error Details (with date range parameters)

**Operations Manager**

User Story: As an operations manager, I want to measure the time difference from alert to resolution to improve response and service efficiency.  
Functionality: This functionality helps in assessing the efficiency of the service response mechanism by tracking the time taken to resolve issues.  
API Requirements:

* GET Machine Reports - AUDIT\_OPERATION
* GET Machine Error Details
* Combination of both API responses to generate time difference reports.

**Performance Analyst**

User Story: As a performance analyst, I want to see average machine uptime metrics at the lease level like AWS metrics for service reliability tracking.  
Functionality: This feature provides insights into machine uptime, contributing to overall service reliability analysis.  
API Requirements:

* GET Machine Reports - AUDIT\_CYCLE\_USAGE

These user stories highlight the diverse needs across various roles within the Machine Management framework. By establishing clear API requirements, developers can ensure that the system meets the functional expectations of each user, thereby enhancing overall operational efficiency.

## Reporting & Analysis (RA)

The Reporting & Analysis (RA) functionalities are central to understanding system performance and operational efficiency across various roles in the Automatic Laundry Solutions system. User stories from roles such as sustainability officers, property managers, service managers, and others outline the specific reporting needs and the APIs required to derive actionable insights from the data collected.

**Sustainability Officer**

User Story: As a sustainability officer, I want to generate reports on water and energy consumption by location to promote environmental responsibility.

Functionality: This feature allows the sustainability officer to monitor resource usage across different locations, aiding in identifying areas for improvement in sustainability practices.

API Requirements:

• GET All Machines at a Location

• Note: Water and energy consumption details may need to be defined in a future API.

**Property Manager**

User Story: As a property manager, I want to receive monthly maintenance and sustainability reports via email to stay informed about site performance.

Functionality: This functionality ensures that property managers are regularly updated on machine performance and sustainability metrics, enabling proactive management.

API Requirements:

• GET Machine Reports - AUDIT\_CYCLE\_USAGE

**Service Manager**

User Story: As a service manager, I want to track unresolved service alerts by location to manage team productivity.

Functionality: This feature helps service managers prioritize issues and allocate resources effectively by providing a clear view of outstanding service alerts.

API Requirements:

• GET Unresolved Service Alerts (specific API may need to be created)

**System Analyst**

User Story: As a system analyst, I want to analyze error type trends to predict future maintenance needs.

Functionality: This functionality allows analysts to identify patterns in machine errors over time, facilitating proactive maintenance planning.

API Requirements:

• GET Machine Error Details

**Compliance Officer**

User Story: As a compliance officer, I want to review machine uptime reports to ensure service-level agreements are met.

Functionality: This feature provides necessary documentation and insights required for compliance checks regarding machine uptime and performance.

API Requirements:

• GET Machine Uptime Reports (specific API may need to be created)

**Facilities Manager**

User Story: As a facilities manager, I want to monitor machine availability by day and time to optimize service coverage during peak usage.

Functionality: This functionality enables the facilities manager to ensure that machines are available when needed most, enhancing user satisfaction.

API Requirements:

• GET Machine Details

**Executive**

User Story: As an executive, I want summary dashboards showing key performance indicators (KPIs) such as machine uptime, average cycles, and service response times.

Functionality: This feature provides high-level insights into operational performance, supporting strategic decision-making.

API Requirements:

• GET Machine Details

• GET Machine Cycle Programs

These user stories highlight the critical role of Reporting & Analysis in enhancing operational effectiveness across the Automatic Laundry Solutions system. By specifying the required APIs, the development team can ensure that the system meets the reporting needs of various stakeholders, driving informed decision-making and operational improvements.

## Real-Time Monitoring (RTM)

Real-time monitoring is vital for maintaining operational efficiency and ensuring machine availability within the Automatic Laundry Solutions system. The following user stories illustrate the requirements for real-time insights, focusing on the need for immediate access to machine status, service alerts, and operational metrics. Each story is accompanied by the necessary API calls to facilitate live data insights.

**Service Manager**

User Story: As a service manager, I want to track unresolved service alerts by location to prioritize maintenance tasks.

Functionality: This enables service managers to immediately identify and address critical issues affecting machine availability across different locations.

API Requirements:

• GET Unresolved Service Alerts (specific API may need to be created)

**Operational Efficiency Analyst**

User Story: As an operations manager, I want to receive alerts about recurring machine errors to schedule proactive maintenance.

Functionality: By receiving timely notifications of recurring errors, operations managers can implement maintenance strategies before issues escalate.

API Requirements:

• GET Machine Error Details

• API to create a table to track error recurrence patterns.

**Data Analyst**

User Story: As a data analyst, I want to generate reports on water and energy consumption by location for sustainability tracking.

Functionality: This feature allows for the examination of resource utilization metrics in real-time, helping to promote environmentally responsible practices.

API Requirements:

• GET All Machines at a Location

• Additional APIs may be necessary for specific consumption details.

**Performance Analyst**

User Story: As a performance analyst, I want to visualize machine availability by day and time to optimize service coverage during peak usage.

Functionality: Analyzing machine availability helps in ensuring that resources are allocated efficiently, especially during high-demand periods.

API Requirements:

• GET Machine Details

**Executive**

User Story: As an executive, I want to compare machine performance across multiple locations to identify underperforming assets.

Functionality: This allows for strategic decision-making based on performance metrics, ensuring that resources are efficiently utilized across all locations.

API Requirements:

• GET Machine Performance Reports (specific API may need to be created)

**Maintenance Planner**

User Story: As a maintenance planner, I want to evaluate maintenance efficiency through time-to-resolution reporting for service requests.

Functionality: This functionality allows planners to assess how effectively service requests are handled, promoting operational improvements.

API Requirements:

• GET Service Request Details

• GET Machine Reports - AUDIT\_OPERATION

These user stories emphasize the necessity of real-time monitoring capabilities in the Automatic Laundry Solutions system. By implementing the outlined APIs, the system can deliver critical insights that enhance operational efficiency and ensure the continuous availability of machines.

## Visualizations (VIS)

Visualizations play a crucial role in enabling various user roles within the Automatic Laundry Solutions system to derive meaningful insights from data through graphical representations. The following user stories outline specific visualization requirements for different roles, emphasizing features such as interactive dashboards, heat maps, and performance graphs, along with their respective API needs.

**Data Analyst**

User Story: As a data analyst, I want to create interactive dashboards that summarize key performance metrics for different machine types, enabling easy comparison and analysis.

Functionality: The analyst can dynamically filter and drill down into specific data sets, facilitating deeper insights into machine performance and operational efficiency.

API Requirements:

• GET Machine Details

• GET Machine Reports - AUDIT\_CYCLE\_USAGE

**Performance Analyst**

User Story: As a performance analyst, I want to visualize machine availability through heat maps that indicate usage intensity across various locations.

Functionality: This feature allows the analyst to quickly identify high-demand areas, enabling better resource allocation and operational planning.

API Requirements:

• GET All Machines at a Location

• GET Machine Reports - AUDIT\_OPERATION

**Operations Manager**

User Story: As an operations manager, I want to see performance graphs that track machine uptime and error occurrences over time, helping to identify trends and potential issues.

Functionality: Visualizing performance metrics over time enhances the manager's ability to make informed decisions regarding maintenance schedules and operational improvements.

API Requirements:

• GET Machine Error Details

• GET Machine Reports - AUDIT\_CYCLE\_USAGE

**Executive**

User Story: As an executive, I want a summary dashboard showcasing key performance indicators (KPIs), including average cycles and service response times, presented in an easily digestible format.

Functionality: This dashboard allows executives to quickly assess overall operational performance and make strategic decisions based on high-level insights.

API Requirements:

• GET Machine Details

• GET Machine Cycle Programs

**Service Coordinator**

User Story: As a service coordinator, I want to create visual reports that differentiate between auto-cleared and manually cleared alerts to assess service workload effectively.

Functionality: By visualizing this data, the coordinator can better understand service demands and allocate resources appropriately.

API Requirements:

• GET Machine Error Details

• API to categorize alerts based on clearance type.

Facilities Manager

User Story: As a facilities manager, I want to visualize machine performance trends using engaging graphics to present to stakeholders during reviews.

Functionality: This feature helps facilitate discussions around operational efficiencies and areas for improvement in machine management.

API Requirements:

• GET Machine Details

• GET Machine Reports - AUDIT\_CYCLE\_USAGE

These user stories highlight the importance of effective visualizations in the Automatic Laundry Solutions system, ensuring that various user roles have access to the insights they need for informed decision-making. By implementing the specified APIs, developers can support the creation of meaningful visual representations that enhance overall system usability and analysis.

* Deliverables and Exit Criteria

Deliverables and exit criteria using both Agile and Waterfall methodologies together, you can create a diagram that visualizes how both processes integrate. Here's how you can approach it:

1. Agile Process: This focuses on iterative progress through sprints, continuous feedback, and evolving deliverables.
   * Deliverables: Each sprint should produce incremental, usable deliverables (e.g., a feature, module, or report).
   * Exit Criteria: Each sprint is completed when it meets the defined acceptance criteria (user stories, testing, and review with stakeholders).
2. Waterfall Process: This approach follows a linear, sequential flow where deliverables are achieved at each project phase, and exit criteria are met before moving on to the next phase.
   * Deliverables: These are produced at the end of each stage (e.g., documentation, designs, or a fully tested product).
   * Exit Criteria: The project can move forward only after all exit criteria of each phase (planning, design, development, etc.) are met.

Diagram Concept

You could create a hybrid diagram that places Waterfall phases on one axis (like a timeline) and Agile Sprints on another axis, showing how both interact:

1. Planning Phase (Waterfall):
   * Deliverables: Project scope, requirements, and timeline.
   * Exit Criteria: Requirements sign-off, high-level design.
2. Agile Sprints (multiple iterations):
   * Deliverables: Incremental features, functionality, or user stories delivered.
   * Exit Criteria: Sprint review, user story acceptance, regression testing.
3. Design Phase (Waterfall):
   * Deliverables: Final design documents.
   * Exit Criteria: Design approval from stakeholders.
4. Agile Sprints (continued):
   * Deliverables: Additional increments that are built on the previous deliverables.
   * Exit Criteria: Completion of features or user stories as per sprint goals.
5. Development Phase (Waterfall):
   * Deliverables: Completed modules or product version.
   * Exit Criteria: Code completion, unit testing passed, and deployment readiness.
6. Agile Sprints (final):
   * Deliverables: Final system integration and testing.
   * Exit Criteria: Product ready for delivery, including user acceptance testing (UAT).
7. Testing & Deployment (Waterfall):
   * Deliverables: Final product ready for deployment.
   * Exit Criteria: All acceptance testing passed, deployment approval.

A diagram of a methodical process

AI-generated content may be incorrect.

* Staffing Model

The approach at which we will be allocating resources to this development cycle. Who will be the Project Manager, Architect, and Developer. The roles that are crucial to the success of this development project. Including the client’s role in the development cycle.

* Assumptions

What should be expected during the duration of the development cycle? What standards need to be met to successfully meet expectations for DOU? Should the environment be available during a specific window? How quickly should the client respond to requests to keep things moving along? How long is the warranty period? What happens with out-of-scope items?

| **ID** | **Description** |
| --- | --- |
|  | Resources are available in each phase(s) as required. |
|  | Resources will provide timely responses to questions, address issues in a timely manner and assist with risk mitigation. |
|  | Estimates may vary by phase: Planning & Analysis / Design – 30%  Development – 20% Testing – 10% (Accounts for defect remediation) |
|  | Business data persisted in SQL server should be available as RESTful API for the portal data display. |
|  | Required Azure/Dev environments are created, and ready for development. |
|  | Reports or Graphs will be blank or will not show data if data is not available. |

* Reference and Related Documents

Table below depicts the referenced and related documents.

|  |  |
| --- | --- |
| Title | Link |
| User stories | [Automatic Laundry Solutions (ALS) - APIUpdated - User Stories Tracking Sheet.xlsx](https://biztechsol.sharepoint.com/:x:/s/AppDevTeam/EWjP4LwKrKBIlE-N4tYQraAB7c_i95oU3xiZ261DpjFSqQ?e=bZueqt) |
|  |  |

A picture containing graphical user interface

Description automatically generated