REQUIREMENT DOCUMENT

CAN 7.0

AUGUST 4, 2023

AVANSEUS TECHNOLOGIES PVT LTD

REVISION HISTORY

Version	Date	Change description	Created by	Updated by	Reviewed by
V 1.0	August 2023	Beta Release	Raksha	Abhilash	Chiranjib

Table of Contents

Terminologies

Classification of requirements based on type and priority.

A. Requirement Types

Requirement Type	Definition
Business	Business requirement deals mainly with business goals and stakeholder expectations. It tells us about the future scope of the product and the objective's worth.
Functional	Functional requirements are much more specific and detailed compared to business requirements. They outline how a product will support business requirements and specify the steps on how the requirement will be delivered.
Non-functional	The non-functional requirement elaborates a performance characteristic of the system. These requirements fall in areas such as accessibility, documentation, efficiency, disaster recovery, security etc.,

B. Requirement Priorities

Priority	Semantics
Critical	A critical requirement without which the product is not acceptable to the stakeholders.
Important	A necessary but deferrable requirement which makes the product less usable but still functional.
Desirable	A nice feature to have if there are resources but the product functions well without it.

Requirements

1. Search predicted faults by PAT number

Type	Functional
Priority	Important

Introduction: A unique number (PAT) is generated for a group of predicted faults. In earlier version, faults along with PAT number were retrieved manually by scrolling. In CAN 7.0, search option for PAT number is introduced to ease the access.

Aim: To retrieve fault details on searching PAT number for clustered faults and single faults.

Requirements:

Requirement ID	Requirement description
REQ070001	User interface to search clustered faults by PAT number. Clustered faults to be searched as per the applied filters.
REQ070002	User interface to search single faults by PAT number. Single faults to be searched as per the applied filters.

2. Intelligent RCA - Approximation technique based on a big knowledge base of root causes

Type	Business
Priority	Critical

Introduction: Previously, root cause was determined by analysing cause and fault history as per technical knowledge or by field learning. Intelligent RCA is a technique that automatically identifies the root cause of anomalies and the resulting critical failures.

Aim: To discover the root cause(s), to take necessary actions and to minimize their effect on the end-user experience.

Requirements:

Requirement ID	Requirement description
REQ070003	Cell level mapping with predicted faults.
REQ070004	Attaching PM KPI trends as part of topology view of predicted faults.

3. Micro-front end architecture to modularize features of CAN

a) Plug & play of modules (Backend & UI) as per customer use case

b) Restructuring of menu items

Type	Functional
Priority	Important

Introduction: MicroFrontend is a type of architecture in which a single monolith web application is divided into different micro-frontend modules. Wherein, each module can use its own technology stack and later can be combined with main host module using webpack integrated with a plugin called ModuleFederationPlugin.

Aim: To integrate multiple modules with different code bases that helps in continuous deployment and testing of individual modules.

Requirements:

Requirement ID	Requirement description
REQ070005	Modularizing the front end and backend based on micro-front end architecture.
REQ070006	Plug & play to activate/deactivate menu items or features.
REQ070007	Restructuring of menu items needed to allow/deny specific sections of features based on use case.

4. Performance dashboard - Analysis items in the form of graph or charts with filters

Type	Non-Functional
Priority	Important

Introduction: Performance dashboard analyses increase or decrease in tickets resolved within SLA, the tickets resolved beyond SLA and unresolved tickets. Using the predictions of CAN, the reactive ticket percentage decreases that automatically decrease the failures.

Aim: Graphical representation of tickets raised and predictive action taken yearly, monthly and daily for various ticket type, ticket closure and ticket category. Also, to display the trend and comparison of predictive action over time.

Requirements:

Requirement ID	Requirement description
REQ070008	Graphs and charts to denote prediction analysis.

Requirement ID	Requirement description
REQ070009	Stats on predictions and priority prediction based on categories and filters such as domain, circle, comparison mode, etc.

5. Performance dashboard - Pattern identification and recommendations

Type	Non-Functional
Priority	Important

Introduction: The performance dashboard includes the history data for each equipment or equipment component that describes alarms with alarm duration along with creation date, closed date etc., ticket data duration & date of ticket creation along with closure and predicted details with date of prediction and relevant prediction information. This identifies fault history and actions can be taken on the current prediction.

Aim: To display the historical data and the pattern of a particular fault for a particular equipment or equipment component graphically.

Requirements:

Requirement ID	Requirement description
REQ070010	Graphical representation of historical data for faults on equipment with information such as the last prediction, duration and the date of ticket creation along with closure and RCA.
REQ070011	Pattern of a particular fault for an equipment to show recommendations to customer.

6. Fault prediction KPI (precision and coverage) generation based on domain code, network type and region

Type	Functional
Priority	Important

Introduction: This module shows the precision of the predictions. It specifies reactive tickets count that match with the prediction. This helps to identify the percentage of tickets that match CAN predictions and these predictive tickets can help decrease the reactive tickets.

Aim: To represent the predictions and matches of the predictions with future alarms graphically.

Requirements:

Requirement ID	Requirement description
REQ070012	Graphical representation of KPI data.

7. Domain specific report configuration and generation

Type	Non-functional
Priority	Desirable

Introduction: Prediction results are generated as an excel report. User can configure fields which they wish to see in the excel report. Earlier report was generated for all domains with a success/failure mail to the mail group. In CAN 7.0, report can be generated for a specific domain for a mail group.

Aim: To configure and generate the prediction report based on domain with distinguished mail templates.

Requirements:

Requirement ID	Requirement description
REQ070013	Allow configuration of domain specific mail groups.
REQ070014	Allow configuration of Email formats for different domains.
REQ070015	Perform report generation based on a domain.
REQ070016	Send emails with relevant reports attached to the associated groups.
REQ070017	Dropdown for Recipient Configuration to select a particular report in RoE screen.

8. Identification of Prediction uniquely - PAT number generation format need to be changed

Type	Functional
Priority	Important

Introduction: Previously, multiple predictions were clubbed to generate a single PAT. In CAN 7.0, a unique number will be generated for each predicted fault at the backend. When a predicted fault occurs, the date and time on which the prediction happened is tagged with a 6-digit prediction identifier to generate the PUID.

Aim: To generate a unique number (PUID) for every fault prediction.

Requirements:

Requirement ID	Requirement description
REQ070018	Generation of PUID in the format YYYYMMDDHHmmnnnnnn. Where, YYYYMMDD represents year, month, date; HHmm represents hour, minute and nnnnnn represents the unique prediction identifier.

9. Replacing CAS with Keycloak thereby providing 2-factor authentication

Type	Functional
Priority	Important

Introduction: Keycloak is a single sign on solution for web apps and RESTful web services. Keycloak provides customizable user interfaces for login, registration, administration, and account management. Keycloak is used as an integration platform to hook into existing LDAP server. It provides features like Two-factor Authentication, OAuth 2.0 support, OpenID Connect, CORS support, Session management, etc. The goal of Keycloak is to secure the apps and services like CAN, Grafana and Kiali.

Aim: To Replace SAML based authentication (CAS) with an OAuth 2.0 based authentication (Keycloak), that provides single sign-on with identity and access management. User login to CAN is redirected to Keycloak application, where it authenticates the user on LDAP and an OTP.

Requirements:

Requirement ID	Requirement description
REQ070019	Removal of CAS and providing SSO using Keycloak and LDAP for CAN.
REQ070020	Allow LDAP configuration in Keycloak admin console.
REQ070021	Enforce 2-factor authentication from Keycloak Admin.
REQ070022	2-Factor authentication works either with Google or Microsoft authenticator.
REQ070023	Enforce password policy from Keycloak.
REQ070024	Login, 2-Factor authentication & Forgot password features are deployed on Keycloak based on custom theme.
REQ070025	User audit shown in CAN is taken from Keycloak database.
REQ070026	Use of memcached as session storage for handling requests from multiple pods.

10. Solace queue integration for alarm collection

Type	Functional
Priority	Important

Introduction: Solace PubSub+ Platform is a complete event streaming and management platform for the real-time enterprise. The Platform helps enterprises design, deploy, and manage an event-driven system. Solace is a text-based interface for configuring and monitoring event brokers. Data is collected from Solace servers for the configured metrics so that the same data is used to generate predictions on the KPIs under observation. This is an enhancement of Data Collection and Configuration module of CAN where an additional pre-integration of data collection through Solace is implemented.

Aim: To create UI support to configure Solace access links with Avanseus CAN. It establishes active link with Solace streaming servers for data intake and retrieve the data in accordance with the metrics, configured over the CAN user interface.

Requirements:

Requirement ID	Requirement description
REQ070027	UI configurations (host, port, username etc.) to connect to Solace queue consumer.
REQ070028	To perform streaming data collection, whenever data is made available in the queue.

11. PM prediction thresholds extension - Add warning threshold to existing critical ones

Type	Functional
Priority	Important

Introduction: In previous versions, threshold alert was configured and displayed. In CAN 7.0, warning level is introduced. Customers are notified when any KPI reaches the warning level. Suitable measures can be taken to avoid equipment failures.

Aim: To configure and display the warning level in extension to the threshold alert.

Requirements:

Requirement ID	Requirement description
REQ070029	UI changes to accommodate warning threshold.
REQ070030	Back end changes to introduce warning threshold to the existing critical threshold.

12. JS interface support in IDE

Type	Functional
Priority	Important

Introduction: The previous version of CAN supported programming codes in Java and Python for customising the data process. For ease of use and better operationalization, we are enabling JavaScript coding support in IDE.

Aim: To modify the IDE environment to support JavaScript codes.

Requirements:

Requirement ID	Requirement description
REQ070031	UI support in IDE to write JavaScript code.
REQ070032	Highlighting special keywords, strings, numbers, methods, etc.
REQ070033	Displaying warnings, errors for the written JavaScript code.
REQ070034	Versioning of JavaScript code.
REQ070035	Execution of JavaScript code.

13. Process monitoring

Type	Functional
Priority	Important

Introduction: In CAN 7.0, Process monitoring screen provides a detailed information of processes, which are of importance.

Aim: To furnish the monitoring details of a process (running, in progress, completed or failed).

Requirements:

Requirement ID	Requirement description
REQ070036	Analysis/monitoring of processes running in the system (Live & Historical).
REQ070037	Detailed information on the status of task executed for each process.

14. Dynamically changing headers of the table based on the chosen domain and network type

Type	Functional
Priority	Important

Introduction: This is an enhancement of Performance Prediction module of CAN. In previous versions, all the headers were displayed in generic manner across different domains and network types. In CAN 7.0, these headers can be configured using a Java enumeration based on the selected domain and network type so that the terminology specific to that network type is used to represent the devices and fault information.

Aim: To change the header names under Real time streaming, Threshold breach and Health index for any selected domain and network type dynamically.

Requirements:

Requirement ID	Requirement description
REQ070038	Ability to configure row labels using an enumeration for different network types.

15. Addition of ComponentType and ImpactCategory during alarm dataload

Type	Functional
Priority	Important

Introduction: In the earlier version, equipment component type was introduced to identify fault type during cause management. In CAN 7.0, the component type classification is done during alarm dataload. Impact category will identify alarm type and the classification is expected during alarm dataload as well. Impact category will also help in cell level fault analysis.

Aim: To include component type and impact category during alarm dataload and to remove equipment component type from cause management module.

Requirements:

Requirement ID	Requirement description
REQ070039	Removal of EquipmentComponentType from Cause Management module.
REQ070040	Addition of ComponentType during alarm dataload. ComponentType should be categorized as NETWORK_ELEMENT, CARD, PORT, LINK or OTHERS.

Requirement ID	Requirement description
REQ070041	Addition of ImpactCategory during alarm dataload. ImpactCategory should be categorized as CELL, SITE, HARDWARE or UNSPECIFIED.

16. Full screen mode for Topology Stitching module

Type	Non-functional
Priority	Desirable

Introduction: Topology Stitching module displays cross-domain topological connections in schematic & map views for a selected source node. In earlier version, full screen mode was not supported in schematic view. Full screen mode is introduced to ease the visualisation.

Aim: CAN provides a full screen view of the topology stitching in both schematic & map view to visualise more number of nodes & their respective connections at a glance.

Requirements:

Requirement ID	Requirement description
REQ070042	User interface to provide full screen mode for Topology Stitching module.

17. System error audit

Type	Functional
Priority	Important

Introduction: The system error audit provides detailed view of errors happening in the software and allows or helps the service delivery team to correct system configuration or R&D team to debug the issue in case of any software problem.

Aim: Audit screen display errors with different level of severity and allows the system admin to acknowledge the error once reviewed or fixed.

Requirements:

Requirement ID	Requirement description
REQ070043	The system provides a user-friendly interface to display error logs in a readable format.
REQ070044	The system allows user to acknowledge error logs, by including additional information or comments related to the error.

Requirement ID	Requirement description
REQ070045	The system allows user to download the error logs as a report in Excel format, containing all relevant fields.
REQ070046	The system allows user to download the error logs as a report in Excel format, with a selected set of fields.

18. Predicted alarms enrichment by KPI trend

Type	Functional
Priority	Important

Introduction: This functionality correlates predicted alarm with its associated KPI trends. During ticket creation, "KPI_trend_report" file will be generated in pdf format and the same will be uploaded to corresponding ticketing tool. The addition of the pdf file along with the other ticketing information will boost the confidence of technicians to respond appropriately while resolving the assigned ticket. It is important that the two pre-requisites should be satisfied for a successful generation of "KPI_trend_report".

1. Equipment Component name should be same across Alarm and Performance Counter related tables.
2. Mapping should be found between Alarm cause and KPIs.

Aim: To generate and upload "KPI_trend_report" pdf file along with other relevant information during ticket creation in Remedy and ServiceNow ticketing tools.

Requirements:

Requirement ID	Requirement description
REQ070047	UI support to draw canvas based on parameters like Equipment component name, KPIs and interested date.
REQ070048	Integration with PlayWright tool to capture canvas element from headless chrome browser.
REQ070049	Backend support to upload the "KPI_trend_report" pdf file to Remedy and ServiceNow tools.
REQ070050	Ability to download and view the pdf file from ticket information present in Remedy and ServiceNow.
REQ070051	Ability to download or delete the pdf file based on the user requirement during ticket generation.

19. Support for collectD data collection interface for server telemetry data

Type	Functional
Priority	Important

Introduction: In CAN 7.0, a new data collection interface is introduced for collection of server elementary data i.e. CollectD. The CollectD interface is present under the Data Collection and configuration tab. It collects system and performance metrics of sever in which it is deployed. Further, the data is stored in the default database of CollectD i.e. Graphite. The collected data is gathered from Graphite and stored in Mongo database that is used for Realtime Streaming (Anomaly detection) and Breach Prediction.

Aim: To provide a data collection interface for system generated metrics and to support Realtime Streaming (Anomaly detection) and Breach Prediction for the collected metrics.

Requirements:

Requirement ID	Requirement description
REQ070052	Providing GUI for collectD interface in data collection and configuration tab.
REQ070053	ETL (extraction, transformation and loading) of collected system metrics from Graphite database (Default database used by CollectD) to mongo database.
REQ070054	Supporting Anomaly detection (from RTSP) and Threshold breach prediction for the above collected metrics.

20. Support for telegraf data collection interface for server telemetry data

Type	Functional
Priority	Important

Introduction: In CAN 7.0, a new data collection interface is introduced for collection of server elementary data i.e. Telegraf. The Telegraf interface is present under the Data Collection and configuration tab. It collects system and performance metrics of sever in which it is deployed. Further, the data is stored in the default database of Telegraf i.e. InfluxDB. The collected data is gathered from InfluxDB and stored in Mongo database, which is used for Realtime Streaming (Anomaly detection) and Breach Prediction.

Aim: To provide a data collection interface for system generated metrics and to support Realtime Streaming (Anomaly detection) and Breach Prediction for the collected metrics.

Requirements:

Requirement ID	Requirement description
REQ070055	Providing GUI for Telegraf interface in data collection and configuration tab.

Requirement ID	Requirement description
REQ070056	ETL of collected system metrics from Influx database (Default database used by Telegraf) to mongo database to use it for threshold breach and real time predictions.
REQ070057	Supporting Anomaly detection (from RTSP) and Threshold breach prediction for the above collected metrics.