REQUIREMENT DOCUMENT

CAN 6.0

JANUARY 24, 2023

AVANSEUS TECHNOLOGIES PVT LTD

REVISION HISTORY

Version	Date	Change description	Created by	Updated by	Reviewed by
V 1.0	January, 2023	Initial Release	Raksha	Abhilash	Chiranjib

avanseus

Table of Contents

Focus

The focus of this release is to provide functionalities to improve operationalization as well as faster integration of predictive maintenance within operator networks. Performance degradation prediction as well as health index based predictive monitoring have been automated end-to-end and linked with existing alarm-based prediction. Inventory ingestion, topology stitching, topology discovery, automatically grouping predictions based on topology as well visually verifying fault clusters are made off-the-shelf. Additionally, with in-memory data preparation and real time anomaly detection, the CAN application enables more dynamic and real time prediction to match the 5G network operations and maintenance requirements. The streaming support of 3GPP using HTTP push ensures continuous input data supply at par with the requirements of such advanced networks. On the automation side, off-the-shelf support for Ansible based integration with network directly or through intermediate RPA systems has been provided. This implies auto remedial actions, based on predictions, can now be directly configured in CAN.

There has been increase in areas where custom code can be provided to further configurability. Python coding is also allowed in addition to existing Java support so that more teams can customize CAN.

On customer experience prediction, a new module has been added to predict broadband customer experience, linking customer data with network data.

CAN application cloud native deployment is also upgraded with LDAPS and better monitoring of CAN application pods using Grafana and Kiali with alert functions.

On scalability, data ingestion pipeline has been made service based and gRPC has been introduced to reduce latency further.

The list of features productised in this release, classified under functional areas, is appended below.

Cloud native ecosystem integration

1. KPI collection from Prometheus
2. Grafana and Kiali integration with LDAP
3. Alerting on threshold breach for Grafana and Kiali
4. Chaos monkey testing by randomly terminating instances
5. Resource limits for Kubernetes

Usability improvement

1. Fault trace generation configuration
2. IDE - UX review and initial viewport correction
3. Python support in IDE - Implementation based on feasibility study
4. Code snippet version control
5. Post-prediction should have IDE
6. Predicted alarm duration should be provided as part of standard product
7. Cause standardization handling during alarm ingestion (making it standardized and high performant)
8. Configurable zone selection during clustering
9. Predictive fault analysis clustering map view to handle large number of markers
10. Lat long validation
11. Lat-long mapping at office code level
12. Session expiry to handle mouse move
13. Algorithm should return error code and error message in case where gaps in superposed sequences are not same

Performance prediction

1. Allowing user to configure classifications (e.g. service impact, technology etc.) of PM KPIs
2. In Performance prediction, KPI names should be there (2G, 3G, 4G, Core, Transport, Device) with separation
3. Configuring linkage between performance KPIs and alarms

4. Configuring linkage between equipment component names as received in alarms and PM KPI records
5. In performance prediction, all or major (user configurable) KPI trends should come in single screen
6. UI changes for changing data granularity while showing KPI trend upon zoom-in
7. Rule discovery mechanism for KPI threshold breach prediction
8. Support for range and min-max type of aggregation for PM KPI
9. Limiting KPI values up to a multiple of max and min thresholds to make KPI sequence smooth
10. Format configuration for PM KPI report
11. Map based display for PM KPI threshold breach prediction
12. Map based display for health index prediction
13. Filtering KPI breach predictions based on site priority
14. Filtering health index predictions based on site priority
15. Shortlisting worst cells for KPI breach predictions

Root cause prediction

1. Allowing configuration of parent-child (P-C) groups based on user input. Similar rules are automatically identified to help user to configure new rules properly.
2. Derivation of additional P-C rules using discovery and making these available in UI for editing and confirmation by user.
3. Identifying and highlighting parent causes as well as affected nodes at cluster level
4. Displaying cluster-based fault history in UI

Standard compliance

1. Streaming support for data collection over 3GPP and HTTP REST protocols (Alarm collection and PM counter/KPI collection)

Real-time prediction

1. Real-time prediction - In-memory PM data preparation
2. Changing existing data flow to micro-services
3. Viewing Historical Anomalies in real time view

AI algorithm research

1. Solving approximate sequence comparison problem needed for synthetic failure history generation
2. New anomaly detection algorithm
3. New clustering algorithm for fault cluster discovery

5G related

1. Anomaly detection in performance KPI
2. Visual display of anomaly during stream processing
3. Selectively picking failures for master sequence generation
4. Dynamically calculating Health Index, thereby calculating offset and scaling factor
5. Calculating health index on cumulative basis

Customer Experience Prediction

1. Customer experience prediction for broadband customers including PON topology maintenance

Topology Correlation

1. Topology Discovery
2. Enhanced filter for cross domain correlation(location, office code, equipment, domain)

3. Enhanced bit pattern view for topology based clustering in topology discovery screen

Data flow improvement

1. Changing current function calling based mechanism to micro-services mechanism using gRPC

Security

1. Support for LDAPS
2. Support for secure Kafka

Additional data format

1. Support for JSON data parsing
2. Support for XML data parsing
3. Support for multiple KPI values in a single record

Automation / Integrations

1. Integration with Redhat Ansible

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 and tells us about the future state of the product and why the objective is worthwhile.
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. KPI Collection from Prometheus

Type	Functional
Priority	Critical

Introduction: This is an enhancement of Data Collection and Configuration module of CAN where an additional pre-integration of data collection through Prometheus is implemented. Prometheus is an open sourced system monitoring and alerting tool. Data is collected from Prometheus servers for the configured metrics so that the same data is used to generate predictions on the KPIs under observation.

Aim: Avanseus CAN establishes active link with Prometheus 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
REQ060001	URL definition for input data retrieval from Prometheus Servers.
REQ060002	Options to define and configure metrics for data retrieval.
REQ060003	Real-Time Prediction Switch: A toggle button to set or unset prediction at real-time.
REQ060004	Cron Pattern Implementation: Real-time prediction occurs according to this pattern.

2. Grafana and Kiali integration with LDAP

Type	Non-Functional
Priority	Important

Introduction: Grafana and Kiali are visualization tools to monitor CPU and memory usage in a particular instance of CAN application. It displays the list of applications that run on the server space along with the capacity utilization of each application. Kiali is integrated to understand the structure and health of service mesh, to monitor the traffic flow and to view the logs of the selected pod. Since there is no authentication options for Grafana and Kiali, a standard application is introduced to link Grafana and Kiali with LDAP that will enable the authentication procedure and make the entire transaction more secure.

Aim: To introduce Keycloak that provides for single sign-on with identity and access management. In keycloak user federation, existing LDAP users can be fetched and configure

Grafana and Kiali as clients. When users login to Grafana/Kiali, they are redirected to keycloak application where it authenticates the him/her with existing LDAP username and password. The user is given a token upon successful login and gets automatically redirected back to the service they were initially attempting to access.

Requirements:

Requirement ID	Requirement description
REQ060005	Addition of existing LDAP server as an identity provider in the user federation section.
REQ060006	Configuration of Grafana and Kiali as clients, so that Grafana/Kiali supports SSO.
REQ060007	Validation of user credentials and subsequent redirection.
REQ060008	Options for different user accesses i.e. Admin, Editor and Viewer.

3. Alerting on threshold breach for Grafana and Kiali

Type	Non-Functional
Priority	Important

Introduction: CAN application micro services is backed by Istio service mesh that coordinates among the micro service threads or the CAN pods. Since CAN handles a large amount of data, it is imperative that these micro-services need to be managed closely. Prometheus, Grafana and Kiali supports such data collection and visualisation where various aspects of functionality can be understood. Alerts will further augment this system where identification and resolution of issues becomes easier.

Aim: To create a pre-configured dashboard with custom panels in Grafana for visualisation and data analysis and to configure alert and send notifications. The alert will be sent to all contacts listed in the contact points.

Requirements:

Requirement ID	Requirement description
REQ060009	Addition of Contact Point: Grafana sends notifications to the list of contact point when breach occurs.
REQ060010	Creation of Grafana Managed Alerting Rule: Create alerting rules that query one or more data sources, reduce or transform the results and compare them or to fix thresholds.
REQ060011	Define Grafana Managed Alert Conditions: Select the query or expression to trigger the alert.
REQ060012	Add details for alert: Add a description and summary to customize alert messages.

4. Chaos monkey testing by randomly terminating instances

Type	Non-Functional
Priority	Important

Introduction: CAN application runs in a data intensive environment. With the adaptation to cloud native architecture, the need of a stable and reliable system had been the key consideration of development. Hence, to ensure survivability of the application on turbulent situations, chaos engineering is adopted in the application. Kube-monkey is an implementation of Netflix's Chaos Monkey for Kubernetes clusters that embraces chaos engineering for strengthening this application. It randomly deletes Kubernetes (k8s) pods in the cluster encouraging and validating the development of failure-resilient services.

Aim: To implement Kube-Monkey in CAN application for failure manipulation of Kubernetes pods of CAN application. Kube-monkey will run at a pre-configured hour (default timing set as 8 a.m.) on weekdays and builds a schedule of deployments that faces a random pod death sometime during the same day. The time range during the day when the random pod death might occur is configurable, defaults from 10 a.m. to 4 p.m. Kube-monkey works on opt-in model and only schedules termination for Kubernetes (k8s) apps that have explicitly agreed to have their pods terminated by kube-monkey.

Requirements:

Requirement ID	Requirement description
REQ060013	To enable and disable Kube-Monkey deployment in CAN pods.
REQ060014	To enable Kube Monkey configurations/opt-in settings that will control the operations of Kube-Monkey during the deployment period (like configuring kill mode, kill value, scheduling time, termination time etc.).

5. Fault trace generation configuration

Type	Functional
Priority	Important

Introduction: In the previous version, fault trace generation was a part of the product. All the parameter values were pre-defined. The backend logic used these default values. However, as the CAN application advances, the fault trace needs to evolve to accommodate the dynamic requirement of CAN users to configure the traceability of faults.

Aim: To make fault trace configurable through UI support. Users can configure parameters in the Advanced Configuration UI screen. The fault trace generation logic that runs in the backend takes values from the configured UI.

Requirements:

Requirement ID	Requirement description
REQ060015	Advance configuration has a block to configure fault trace generation parameters.
REQ060016	Backend logic uses the parameter values configured in the Advanced Configuration screen.
REQ060017	Configurable parameters through UI support - Fault Trace History Days, Fault Trace Max Day Multiplier, Max Fault Trace Count and Fault Trace Cron.

6. IDE - UX review and initial viewport correction

Type	Functional
Priority	Important

Introduction: IDE can perform a variety of functions like write code, compile code and debug code. IDE also supports additional control in the form of version management that help users to select different customization code versions easily over UI support to achieve the level of customisation that users require in their day to day operations.

Aim: Users can write the Java/Python code and compile it. It is an on the fly compilation. It displays errors and warnings. If the code is error free, developer can save that code to the DB.

Requirements:

Requirement ID	Requirement description
REQ060018	To expand and view the non-editable part of IDE.
REQ060019	Highlighting special keyword, objects, classes, strings which conventional IDE supports.
REQ060020	Displaying warnings, errors and info level information only for the editable part.
REQ060021	Auto saving of code, auto closing of brackets & quotes, providing keyboard support to select auto complete options.

7. Python support in IDE

Type	Functional
Priority	Important

Introduction: The previous version of CAN support programming codes in Java for customising the data process. For ease of use and better operationalization, we are enabling Python coding support in IDE. The simple, powerful and easy to use methodologies in Python will improve the productivity of the CAN application.

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

Requirements:

Requirement ID	Requirement description
REQ060022	Highlighting special keyword, objects, classes, strings in Python.
REQ060023	Displaying warnings, errors, and info level information only for the editable part in Python.
REQ060024	Option to download code and clear code.

8. Code snippet version control

Type	Non-Functional
Priority	Important

Introduction: The IDE environment in CAN application is designed to provide UI support towards customization of data processing and report generation. This version of CAN additionally supports Python coding along with existing Java codes. To support seamless transition between different types of customization, UI support for version management where user can select the required code snippets over click of a button is essential

Aim: To modify the IDE environment to support multiple versions of customization codes. CAN allows users to save the codes with different tag names and re-use the code as and when required. This feature allows saving the code with errors as well. Version drop-down displays all the versions of code. User can select any version of code. The screen displays only the selected version for edit and save purpose.

Requirements:

Requirement ID	Requirement description
REQ060025	Commit option to store different versions of code.
REQ060026	Request confirmation to save code changes to an existing version or to a new version.
REQ060027	In the drop-down of Version, displaying tag icon (/) to indicate the current version selected.
REQ060028	On click of Save, the selected version is saved only if the code is error free.

9. Post-prediction should have IDE

Type	Functional
Priority	Important

Introduction: Avanseus CAN currently supports IDE environment where users can use programming codes in Java & Python to customize the input data parsing. IDE can perform a variety of functions like write code, compile code, debug code, and makes a variety of compilation activities seamless. Similar IDE after the Prediction stage will enable users to deal with high volumes of predictions by enabling users to write functional codes mostly to enrich predictions with data from external sources.

Aim: To enable users to write/edit/delete customizable codes in Java and Python, executable on the prediction through Integrated Development Environment (IDE). The IDE also supports the version control, code compilation, storage and other related aspects of versatility through the user interface of CAN.

Requirements:

Requirement ID	Requirement description
REQ060029	UI based IDE support for Python and Java coding for prediction fields customization.
REQ060030	Write/ Edit options like clear, delete, etc.
REQ060031	Version control.
REQ060032	Download – Option to download Java and Python code.

10. Predicted alarm duration should be provided as part of standard product

Type	Business
Priority	Critical

Introduction: CAN application predicts the occurrence of a fault in the network. In the previous version, the alarm duplication count was provided to users along with the report to show the repeated nature of the fault indicating its criticality. As the CAN application advances, based on the alarm duration details that user provides, CAN predicts the duration of future alarms as well. This ensures that the CAN users get the criticality and impact of the predicted alarm and can proceed for calculated actions on the predicted faults.

Aim: To predict the alarm duration based on fault history and to include them in the daily report.

Requirements:

Requirement ID	Requirement description
REQ060033	Predicting alarm duration based on historical alarm durations for a particular alarm.
REQ060034	Providing predicted alarm duration as part of the daily report.

11. Cause standardization handling during alarm ingestion (Making it standardized and high performant)

Type	Functional
Priority	Important

Introduction: Alarm data that CAN processes is generally huge and diverse. Many times, the same causes are represented in different way though the underlying solution remains same. Cause standardization is converting such data into a standard format so that a better data set is created for prediction. Cause standardization provides accurate predictions.

Aim: To enable cause standardisation by converting raw causes to standard causes. UI support is provided to upload the mapping data where the raw causes, typified causes and standard causes are uploaded. Regex is used to map Raw Causes to Standard Causes. The Record parser is designed to map customer provided information with CAN. It makes customer information field compatible with CAN field

Requirements:

Requirement ID	Requirement description
REQ060035	Option to add raw cause using File Upload. Excel file creation format and file size are displayed.
REQ060036	Option to display the Raw cause, Typified cause and Standard cause.
REQ060037	Option to display Regex and its associated Standard cause.
REQ060038	Option to Search the cause.
REQ060039	Option to download Raw cause, Typified cause and Standard cause on Sheet 1 and Regex and Standard cause on Sheet 2 in a single excel file.
REQ060040	On click of “Input Mapper”, redirects the page to input mapper screen.
REQ060041	Option to Delete Raw cause and the associated Typified cause, Standard cause in Cause standardization table and Regex table.

12. Configurable location type (nation, region, zone, office code, topology) and selection during clustering

Type	Functional
Priority	Important

Introduction: Clustering is required to find a group of faults that occurred together in the past. In previous version, only zones were considered in clustering. In this function, clustering can be configured with all location types as well as domains.

Aim: Before forming clusters, input configuration is done in the Clustering algorithm configuration screen based on seven parameters. Clustering runs separately or in combined way known as grouping. Generated clusters are displayed in Cross-domain correlation tab as Bit pattern view or Map view.

Requirements:

Requirement ID	Requirement description
REQ060042	Back end support to populate all possible clustering input data groups based on the input configuration provided.
REQ060043	UI support for Input configuration with following details: Cron, Membership threshold, Slot length, No. of clusters, Enable clusterization, Similarity threshold, No. of days.
REQ060044	Option to add new clustering input configuration manually.
REQ060045	Option to Edit an input configuration.
REQ060046	Option to Delete a clustering input configuration.
REQ060047	Option to Filter clustering based on Location type, Location name or Domain.
REQ060048	Option to Search a clustering input configuration based on name or location type.

13. Session expiry to handle mouse move

Type	Functional
Priority	Important

Introduction: Session expiry or Session time out is the invalidation of user session when user does not perform any actions for certain interval. In CAN application, after session time out, user needs to log in again. In CAN application, when user logs in, only the activities that involve server action are tracked. CAN application assumes other activities like mouse

movement needing no server side call as “no activity” and the user are logged out of the application after the session expiry time. Since this leads to repeated login for users hampering seamless operations, changes are incorporated in the main application to consider any activity that indicate user presence, while managing the session timers.

Aim: To consider every movement on the user interface as a user activity. When user accesses the CAN interface and performs any activity, session expiry is extended as and when the activity is done.

Requirements:

Requirement ID	Requirement description
REQ060049	Moving the mouse on the user interface extends the session for time set by CAN application as session expiry time, from the time user has made an activity.
REQ060050	No mouse move on the user interface logs out the user from the application after session expiry time.

14. Algorithm should return error code and error message in case where gaps in superposed sequences are not same

Type	Functional
Priority	Critical

Introduction: CAN application predicts failures based on the historical failures and the due mathematical calculations derived ipso facto. Generally, CAN application requires three historical failures to generate predictions and hence the application needs to have various checks and balances to handle such KPIs where there are no 3 historical failures or very few pre-failure events.

Aim: To continue prediction generation where adequate number of failures or pre-failure events are absent by concatenating failures from similar equipment and cropping the pre-failure event history at the input sequencing stage thereby maintaining the required precision and coverage. The application must cover all the varying scenarios due to which the pre-failure event bits are fallen in short including the case of missing values of certain KPIs.

Requirements:

Requirement ID	Requirement description
REQ060051	Concatenating the failures from similar equipment components followed by the interested equipment component’s own recent history.
REQ060052	If history prior to failure event is less than the expected 30 bits, then consider the history by cropping the historical bits to 25 or 20 bits.

15. Allowing user to configure classifications (e.g. service impact, technology etc.) of PM KPIs

Type	Functional
Priority	Important

Introduction: Avanseus CAN supports performance KPI monitoring of the network where the KPI threshold breaches are identified by the system and notified to relevant stakeholders. This new feature defines the threshold boundaries along with the impact for different technologies with more user-friendly UI support.

Aim: To provide a comprehensive support in UI to configure the domain, network type, cause category, etc. The network KPIs are identified by the system from the input data and the other relevant attributes. Apart from network KPI names, other attributes are made editable in the UI to ensure that relevant inputs are fed to the system in the most accurate way for processing and decision-making. User will have options to set these values individually through edit option or to bulk upload the same as file for system to process and upload.

Requirements:

Requirement ID	Requirement description
REQ060053	Options to edit the Domain, Network type, Service impact, Threshold type, Threshold value, Cause category and unit.
REQ060054	Options to upload the bulk file for processing.
REQ060055	Options to download the data.
REQ060056	Option to edit and delete individual or multiple KPIs.
REQ060057	Options for search KPI.
REQ060058	Options to filter view.

16. In Performance prediction, KPI names should be there (2G, 3G, 4G, Core, Transport, Device) with separation

Type	Functional
Priority	Important

Introduction: In the earlier versions of CAN, there was no option to filter based on different domains and underlying technologies. In this version of CAN, UI filtering based on technology and domain is enabled for all the 3 readily available use cases such as Threshold breach, Health index degradation and Real time streaming. This new feature will also make the definition of threshold boundaries along with the impact for different technologies more user friendly with the UI support.

Aim: To provide a comprehensive support in UI to filter based on domain, network type, cause category, etc. By default, CAN assigns 'ACCESS' Domain and '4G' Network Type. However, this can be modified from advanced configuration tab.

Requirements:

Requirement ID	Requirement description
REQ060059	Option to filter by providing Domain, Network type, Service impact, Location, Site Priority, Cause category etc.
REQ060060	Ability to download Daily and Matching report based on applied filters.
REQ060061	Ability to search based on equipment component and cause name.

17. Configuring linkage between performance KPIs and alarms

Type	Functional
Priority	Important

Introduction: This is KPI to alarm correlation configuration screen. It is a function for relating KPI and alarms as KPI is related to the performance counter configuration. Alarm is triggered for a KPI for a specific reason. It is used for prediction and performance management of KPI.

Aim: In this screen, user can view the KPIs supported for a particular alarm. Due to redundancy and tracking issue, an alarm can be selected only once. Although, KPI selection for any particular alarm can be one, many or all. Reason is listed for each KPI. It is not mandatory. KPIs and reasons can be added/deleted at any point of time. When CAN runs health index prediction, user can view which alarm has an issue and focus on trends of the corresponding KPIs.

Requirements:

Requirement ID	Requirement description
REQ060062	Option to select alarm (Unique).
REQ060063	Option to select KPI/KPIs.
REQ060064	Reason may/may not be listed.
REQ060065	Edit option to add/delete KPI and reason for a particular alarm at later stage.

18. Configuring linkage between equipment component names as received in alarms and PM KPI records

Type	Functional
Priority	Important

Introduction: The equipment component names received in performance counter may differ from the data in alarms. Since both are interconnected, the application needs to link these different names together for a comprehensive prediction report.

Aim: To provide a comprehensive support in UI to configure and link the equipment names in alarm and KPI record within the input mapper screen.

Requirements:

Requirement ID	Requirement description
REQ060066	Option to write the logic to modify and match the names as received using IDE.

19. In performance prediction, all or major (user configurable) KPI trends should come in single screen

Type	Functional
Priority	Important

Introduction In the previous versions of Avanseus CAN, the KPI trends are shown discreetly for the selected equipment component. Since the telecom environment involve monitoring of larger number of KPIs that are critical for providing the adept outlook of network, it is imperative that CAN should show the trends of Multiple KPIs in single screen. This requirement will enable CAN to monitor multiple KPI trends together in a single screen that will improve the ease of operation (Analysis of KPIs).

Aim: To provide a single window to view the trends of multiple KPIs by users in graphical semantics. The UI must support the visualisation of all KPIs with option to view only the significant KPIs that are critical to network performance. The user must have easy navigation option to see the trends of all selected KPIs with option to search any KPI from the list of KPIs. The user should have an option to check the historical values of any KPI.

Requirements:

Requirement ID	Requirement description
REQ060067	On landing, the screen must display the significant KPIs.
REQ060068	Radio button for choosing between significant and all the KPIs.
REQ060069	Search option for searching a particular or multiple KPIs.
REQ060070	Multi-select option for searching a particular or multiple KPIs.

Requirement ID	Requirement description
REQ060071	Two different views for displaying the trends: List View and Thumbnail View.

20. UI changes for changing data granularity while showing KPI trend upon zoom-in

Type	Functional
Priority	Important

Introduction: Avanseus CAN displays the historical trend of KPIs in graphical semantics generally as aggregation of days or hours depending on the input data. Such trends, though provides the actual trend of monitored KPIs, the minor details of such trends is diluted due to the aggregation and smoothening of data. Users interested in such minor scale of trends will require an UI that can show the semantics at different granularity levels in accordance with the requirements of users.

Aim: To provide a zoom tool along with the user interface so that users can zoom over the KPI semantics. As the user zoom in, the granularity of KPI display deepens providing users with the details of KPI changes at seconds/minutes/hours aggregation depending on the input data. On zoom-out, the granularity again changes to provide a peripheral view based on daily aggregation.

Requirements:

Requirement ID	Requirement description
REQ060072	Zoom-in and Zoom-out for displaying the data at different levels of granularity. 2 levels of zoom-in and zoom-out functionality is provided.
REQ060073	Upon Zoom-in, UI changes for data granularity while showing the KPI trend.

21. Rule discovery mechanism for KPI threshold breach prediction

Type	Business
Priority	Critical

Introduction: The two aspects of a prediction are Precision and Coverage. When threshold breach prediction is done, predictions provided on a daily basis can be controlled. For every KPI at equipment component level, Best Fault Pair (x,y) is defined where, x indicates the coverage and y indicates the precision.

Aim: This feature provides accurate predictions with less coverage. In such scenario, few predictions are filtered out based on certain rules. CAN provides two pre-defined rules: KPIDiscoverRule and KPIEquipmentDiscoverRule. Commonly used rule is the KPIDiscoverRule.

The rules can be configured only for threshold breach prediction. Rules once used for a set of KPIs cannot be used again for another set of KPIs. There is an option to write new rule using the template provided by Avanseus and an option to edit the default rule.

Requirements:

Requirement ID	Requirement description
REQ060074	Option to select Rule from the dropdown.
REQ060075	Option to select Domain, Network Type, Key Type and Key Value.
REQ060076	Option to modify the selected KPI list (Add New).

22. Support for range and min-max type of aggregation for PM KPI

Type	Functional
Priority	Important

Introduction: Avanseus CAN supports a variety of KPIs that can be monitored on real-time basis and further their threshold breaches can be predicted. Most of these KPIs are monitored based on breaches on their minimum or maximum values. There are certain KPIs whose functionality is defined based on range of values. Hence, CAN must be enabled to understand such data when a KPI value is said to be normal when it is within a certain range or beyond a certain range.

Aim: To enable Avanseus CAN to deal with KPI values that are normal with-in certain range (Min-Max) or beyond certain range of values (range). The UI must be given option to configure these values along with other edit options from the operationalisation point of view in the KPI management screen.

Requirements:

Requirement ID	Requirement description
REQ060077	Min-Max: 2 values will be stored, Threshold 1 and Threshold 2. It considers the value between Threshold 1 and Threshold 2.
REQ060078	Range: If a value is mentioned in range, it excludes the range area and considers below or above the range.

23. Limiting KPI values up to a multiple of max and min thresholds to make KPI sequence smooth

Type	Functional
Priority	Important

Introduction: Avanseus CAN uses input data for creation of trends based on which the chances of failures and other related predictions are assessed. Since the input data is sourced from network, the actual breach variations can differ depending on the health status of equipment under monitoring. Such values, generally called as outliers can affect the overall representation of trends.

Aim: Smoothening of KPI sequences to ensure appropriate representation of the data especially in graphical semantics.

Requirements:

Requirement ID	Requirement description
REQ060079	By default, limiting the threshold value by 3 folds to that of the configured min and max value.

24. Format configuration for PM KPI report

Type	Functional
Priority	Important

Introduction: KPI report is essential for customers. It allows customer to review and analyze the operator performance. It provides notification to management for degraded operation. It helps in taking proactive action to prevent possible upcoming threshold breaches and health degradation.

Aim: To provide end user a flexibility to customize the daily/matching report template by adding/removing/rearranging the interested columns for both threshold breach as well health index related predictions. Output of prediction runs will be dumped to a specific table by name 'PerformanceCounterPrediction' (in case of threshold breach prediction) and 'HealthIndex' (in case of Health index degradation prediction). Fields of these prediction output tables are made available in the UI that allows end user to map these fields against the column name of their interest, thereby enabling them to design the prediction report of their choice.

Requirements:

Requirement ID	Requirement description
REQ060080	Page configuration tab: Enables to add selectively chosen interested fields in the report and persist the configuration.
REQ060081	Option to Delete a column in page configuration tab.
REQ060082	Option to Move a column to the desired position in page configuration tab.

Requirement ID	Requirement description
REQ060083	Ability to configure multiple page and report configuration, however at any point of time only one configuration should be set as 'ACTIVE'.
REQ060084	Default template should be made available for both threshold breach as well as Health Index predictions. However, this template can be modified to include other desired fields.
REQ060085	Option to add new configuration manually.
REQ060086	Query: Option to write a query to filter the required fields in the report.
REQ060087	Data type: If the user wants a data type other than number or string, there is an option to write query by choosing 'Complex' data type.
REQ060088	Among multiple report configuration, user should be allowed to select only one configuration as 'ACTIVE' against which report contents will be decided.

25. Identifying and highlighting parent cause at cluster level

Type	Business
Priority	Critical

Introduction: CAN application identifies Root Cause for a given network fault through two-dimensional understanding of domain and field learning. Since the application groups network elements as clusters, based on their interconnection, the root cause or the parent cause of the common fault also can be understood and displayed.

Aim: To perform root cause analysis for clustered group of faults. The parent and child causes are uploaded or discovered and reviewed by users. UI support to be provided for data upload, cluster configuration and display of data.

Requirements:

Requirement ID	Requirement description
REQ060089	Highlighting of parent causes under Topology Discovery Tab in Tabular view and Bit pattern view.
REQ060090	Marking and highlighting parent causes under Topology Discovery Tab in Map View

26. Parent-child correlation - after clustering, we should identify which are local clusters automatically

Type	Business
Priority	Critical

Introduction: Parent child correlation is helpful in root cause analysis. Clusters that have correlated causes and identified by parent-child rules are considered as local clusters. CAN application must identify such local clusters automatically and display over the UI.

Aim: UI support for selecting, filtering and searching local clusters. Map view and Bit pattern view to augment the UI support for CAN users.

Requirements:

Requirement ID	Requirement description
REQ060091	Option to select Locality type (Nation, Division, Region, Zone, Office code and Topology).
REQ060092	Option to select Cluster based on locality type.
REQ060093	Option to Filter clusters based on Domain, Equipment and Cause.
REQ060094	Option to view clusters as Map view and Bit Pattern view.

27. Allowing configuration of parent-child groups based on input

Type	Business
Priority	Critical

Introduction: Parent child correlation is an enhanced way of root cause analysis. The parent cause to group of errors are configured or discovered. This will enable seamless understanding of the network irrespective of their domains and enhance ease of root cause analysis.

Aim: To enable Parent - child grouping of network faults or errors. The parent cause will be the root cause of all child causes. Option to upload the parent-child mapped details as well as discovery option manually based on the fault logs.

Requirements:

Requirement ID	Requirement description
REQ060095	**Manual:** 1. Option to add new parent child correlation manually using Add Values. Maximum number parent causes are configurable and any number of child causes can be selected. 2. Option to add new parent child correlation using Upload File. Excel file creation format and file size are displayed. 3. Errors in the input file will be sent over email. 4. Option to Search the Parent Cause by rule id. 5. Option to download the existing parent and child causes. Generated report received via email after click on download. 6. Option to view, edit and delete Child causes, Parent cause, Root cause, Recommended NOC action and Recommended field action. 7. Option to Sort child causes by domain name. 8. Option to filter by parent causes, child cause and status. 9. Option to update status (Active/Inactive). By default, status is Active. 10. Option to sort parent and child causes by status. 11. Option to show similar rules. Similarity conditions are configurable. 12. When adding new cause, if status is active, duplicate check is carried out. Duplicate checking conditions are configurable
REQ060096	**By Discovery:** 1. Option to select parent causes among child causes. Maximum number of parent causes are configurable. 2. Option to add new, edit parent-child rules using Upload File. Excel file creation format and file size are displayed. 3. Errors in the input file will be sent over email. 4. Option to Search the Parent Cause by rule id. 5. Option to download the existing parent and child causes. Generated report received via email after click on download. 6. Option to view, edit and delete Child causes, Parent cause, Root cause, Recommended NOC action and Recommended field action. 7. Option to Sort child causes by domain name. 8. Option to filter by parent causes, child cause and status. 9. Option to update status (Active/Inactive). By default, status is Inactive. 10. Option to sort parent and child causes by status. 11. Option to show similar rules. Similarity conditions are configurable. 12. When changing status to active, duplicate check is carried out. Duplicate conditions are configurable. .

28. Streaming support for data collection over 3GPP and HTTP REST protocols (Alarm collection and PM counter/KPI collection)

Type	Functional
Priority	Important

Introduction: Avanseus CAN data collection already supports protocols like SMTP, Kafka, Prometheus, etc. for input data collection. 3GPP is such protocol, if initiated, helps in seamless data collection.

Aim: To create UI support to configure 3GPP access links with Avanseus CAN. The data collection can be enabled or disabled for automatic initiation depending on user requirement.

Requirements:

Requirement ID	Requirement description
REQ060097	UI for logging, initiating data transfer or data acceptance through 3GPP.
REQ060098	Data load occurs for those entries that have Collection Status as ACTIVE.
REQ060099	Addition of 3GPP5G configuration in data collection and configuration screen.
REQ060100	Options to select and manage the type of data transfer: **Direct Method** A NF Service Producer may return the representations of the resources directly in the response body. The response body contains an array of the resource representations. 1. **Direct Delivery with Iterations** If a large number of resource representations need to be returned, the NF Service Producer may return a representation containing a partial list of the requested resources in the response body, with link(s) containing URI(s) allowing the client to retrieve the remaining part(s) of the resources. **Indirect Method** 1. **Indirect Delivery** A NF Service Producer returns a representation containing only a list of links to the requested resources in the response body. 2. **Indirect Delivery with HTTP/2 Server Push** A NF Service Producer may use HTTP/2 Server Push, if HTTP/2 Server Push is supported in the PLMN. **HTTP Dynamic URL Method** The customer can send URL of the resource when there is any addition of data.

29. Realtime prediction - In-memory PM data preparation

Type	Functional
Priority	Critical

Introduction: During real time prediction, the biggest factor of concern is to complete the prediction activities (Inclusive of pre-processing and post processing activities) at minimum possible duration. Considering this 'time' dimension, it is highly appropriate to fetch the data from an in-memory DB like Redis instead of mongoDB. Redis is an in-memory database that is primarily used to cache most frequently used data.

Aim: To enable real-time prediction and faster data streaming by storing data related to prediction process in Redis cache.

Requirements:

Requirement ID	Requirement description
REQ060101	Redis cache configuration.
REQ060102	To store prediction related data in the Redis cache so that it can be used while running Realtime Prediction.

30. Changing existing data flow to micro-services

Type	Functional
Priority	Important

Introduction: Data loading is one of the key process before prediction generation where the remote data are extracted from source file and stored in CAN database. The existing method of batch filing while data loading is time consuming. Micro-services is a distributed system of record processing that reduces the time taken for complete data loading process. Currently, in Kubernetes based setup, the data flow process is implemented using gRPC protocol.

Aim: To reduce delay and achieve parallel processing of data while data loading by breaking down the data loading method to multiple modules called File Handler and Record Processor. File handler while accepting a file as an input and reads the entire lines one at a time and thereby submitting randomly to record processor. The Record Processor processes all the lines in parallel and at any point in time. It is a high availability distributed architecture. Hence, the delay in processing is reduced and parallel processing can be achieved reducing the time for data loading.

Requirements:

Requirement ID	Requirement description
REQ060103	Breaking down the existing batch based data flow system into multiple modules (File Handler and Record Processor).

Requirement ID	Requirement description
REQ060104	Communication between File handler and Record processor and the consumer that submits the file-to-file handler using micro services (gRPC protocol).
REQ060105	Deployment of multiple instances of File handler and Record processor to achieve parallelism to process files and records.

31. Solving approximate sequence comparison problem needed for synthetic failure history generation

Type	Business
Priority	Critical

Introduction: Prediction of faults that happen after long time gap, may not always have minimum required failure or degradation instances (typically 4 failures) for the equipment component CAN is predicting. This is also expected since faults are happening after long time and historical data is limited to typically 6 months to 1 year. For example, if a fault happens after 6 to 9 months, there may be just one or no failure within the historical data. The way to overcome this situation is to create synthetic history by combining available failure and performance data from multiple similar equipment.

Aim: The central activity is to identify similar equipment based on PM KPI trend similarity before the fault happens, in addition to choosing equipment of same type and make. The idea is to find similar trends of different PM KPIs before failure. Then the equipment is showing similar behaviour and can be used as proxy for synthetic history generation. The basic concern is how to check whether two sequences of decimal values are approximately same or not and output should be a quantitative measure.

Requirements:

Requirement ID	Requirement description
REQ060106	Possible to compare two sequences of decimal values, based on trend and get an output between 0 and 1. 0 being completely dissimilar and 1 being completely similar.
REQ060107	Sequence comparison should be an online activity and provided as HTTP API.

32. Anomaly detection in performance KPI

Type	Functional
Priority	Important

Introduction: Anomaly is understood as significant deviation from common pattern followed by the KPI sequence in the past. Anomaly can be created by single abnormal KPI value or by a set of KPI values over a period. In both cases, anomaly is a strong indicator of a breach of KPI or consequential failure of equipment or port. In case of real-time monitoring of streaming data, the key to prediction lies in understanding the performance variations at the earliest and anomaly detection is the best way to understand an underlying or upcoming issue.

Aim: To detect anomaly during real-time prediction of streaming data.

Requirements:

Requirement ID	Requirement description
REQ060108	Onset of anomaly to be detected based on historical data characteristics.

33. Visual display of anomaly during stream processing

Type	Functional
Priority	Important

Introduction: Anomaly is understood as significant deviation from common pattern followed by the KPI sequence in the past. Anomaly can be created by single abnormal KPI value or by a set of KPI values over a period. In both cases, anomaly is a strong indicator of a breach of KPI or consequential failure of equipment or port. In case of real time monitoring of streaming data, the key to prediction lies in understanding the performance variations at the earliest and anomaly detection is the best way to understand an underlying or upcoming issue.

Aim: To provide a UI for detected anomaly.

Requirements:

Requirement ID	Requirement description
REQ060109	To provide stream data display option in Real time streaming tab under the Performance Prediction screen.
REQ060110	Graphical semantics of data inflow and anomaly updated in real-time.

34. Selectively picking failures for master sequence generation

Type	Functional
Priority	Important

Introduction: This is one of the major steps in the Health Index Prediction pre-processing phase. The KPI-based health index prediction will have a huge volume of data as the details from performance counters are captured at periodic intervals ranging from 15 seconds to 1

hour. Yet, there could be cases where certain equipment might have an insufficient number of previous alarm occurrences. Since this affects the overall prediction process, this feature is enabled to overcome the shortcomings of such absence of alarm data availability.

Aim: During the pre-processing phase of Health index prediction, this function must enable the generation of input sequences for all the KPIs corresponding to all the alarm causes and for every equipment component. 'Similarity Check' algorithm to be used for those equipment components that are having an insufficient number of alarm occurrences. The algorithm provides a list of 3 similar equipment components to that of a testing equipment component for the selection of failures. A similarity check is not executed for equipment components that are having a minimum of three alarm occurrences for all the interested causes in their respective historical data.

Requirements:

Requirement ID	Requirement description
REQ060111	Master sequence extraction by considering the latest alarm date: By default, During this pre-processing step, API has to generate input sequences for all the KPIs corresponding to all the alarm causes and for every equipment component. The 'Similarity Check' algorithm will use these generated input sequences later.
REQ060112	Master sequence extraction by specifying the last date of the training period: Extraction of input sequences by accepting the end date or last date of the chosen training period ('before' parameter in HTTP API) as the input parameter.
REQ060113	Master sequence extraction for only interested equipment components: API should be capable of accepting a list of equipment components.

35. Dynamically calculating Health Index, thereby calculating offset and scaling factor

Type	Functional
Priority	Important

Introduction: Health index prediction is dependent on Offset and Scaling factors. The dynamic calculation of health index will require this offset and scaling factor to be calculated dynamically. Hence, there need to be an easy access and method to calculate the offset and scaling factor that are decisive towards calculation of health index dynamically.

Aim: Creation of API that will invoke dynamic health index reference parameter calculations. Depending on the period of calculation, there should be option to specify the end date of alarms that need to be considered while calculating the reference parameters.

Requirements:

Requirement ID	Requirement description
REQ060114	Creation of API to generate health index reference parameters upon invoking.

Requirement ID	Requirement description
REQ060115	Option to specify the end date of alarm data that need to be used towards generation of health index reference parameters.

36. Calculating health index on cumulative basis

Type	Functional
Priority	Important

Introduction: Health Index is introduced to ensure greater understanding and control of equipment and service abnormalities. Health Index is calculated continuously and any downward trends or sudden dips are directed for immediate investigation. Hence, Health Index need to be made immune to sudden fluctuations in KPI values that can act detrimental to the overall efficiency of Health Index output.

Aim: To keep Health index immune to sudden fluctuations of KPI. To identify the actual health of the equipment or service by considering the Cumulative Health Index. The Cumulative Health Index is derived from previous Health index value of equipment or service as the average of the minimum of last 3 values of Health Index. This will be initiated in cases when the Health Index value suddenly fall below the threshold value of 40 during which the CAN algorithm will reconfirm the actual status by considering the cumulative health index by looking back to the previous health index values.

Requirements:

Requirement ID	Requirement description
REQ060116	Identifying the fall of Health Index below the threshold value of 40.
REQ060117	Calculating Cumulative Health Index by checking last 4 Health Index values and taking average of minimum 3 values to create the Cumulative Health Index.
REQ060118	If previous 4 Health Index values are unavailable, consider as many data points as possible to ensure the availability of minimum 3 values for CHI calculation.

37. Customer experience prediction for broadband customers including PON topology maintenance

Type	Business
Priority	Critical

Introduction: Customer experience prediction focuses on the probability of customer to initiate call with the support centres for network related issue as well as the probability of customer reaching out for booking repair tickets. Call Probability is defined as odds for customer to initiate a call in next 7 days.

Aim: To predict the chances of customers initiating call with the support centres for next 7 days (call probability) and to predict the chances of Repair, where the customers reach out for technicians to book ticket. Such call and repair probabilities to be calculated at ONU level and based on this, service impact on the OLT port to be predicted.

Requirements:

Requirement ID	Requirement description
REQ060119	UI support to display Alarm and KPI prediction for broadband data
REQ060120	UI support to filter predictions and Customer Service Impact.
REQ060121	UI support for network view and graphical view of breaches and suggested actions.
REQ060122	Backend activities & UI support to display call and repair probabilities and service impact based on call and repair probability.
REQ060123	Modification in File Collection, Data parsing etc. to accommodate Customer Experience Module.

38. Changing current function calling based mechanism to micro-services mechanism using gRPC

Type	Functional
Priority	Important

Introduction: REST API communication was a function-based service, where the function is directly called in the code. gRPC is a more efficient and modern Remote Procedure Call that can work seamlessly in any environment to connect application services. In gRPC, functions are changed into modules. It is designed for low latency and high throughput communication. It is developed to speed up data transmission between micro-services and other systems that need to interact with each other. Hence, the transition from REST to more effective gRPC for internal micro-services will enhance the optimization of application performance.

Aim: Transition from REST to gRPC protocol for internal micro-services. Protocol buffers are defined for exchange of data over HTTP/2 protocol and enable bi-directional streaming, serving multiple service request and push messages in real time without polling.

Requirements:

Requirement ID	Requirement description
REQ060124	Modularization of functions.
REQ060125	gRPC protocol is used to call the modules.

Requirement ID	Requirement description
REQ060126	Function implementation for internal module communication.

39. Support for LDAPS

Type	Business
Priority	Important

Introduction: LDAP is an open and cross platform protocol used for directory services authentication. The benefit of using LDAP is that information for an entire organization can be consolidated into a central repository. CAN application is integrated with LDAP. LDAPS allows for the encryption of LDAP data (which includes user credentials) in transit during any communication with the LDAP server, thereby protecting against credential theft. This act as an additional layer of security while integrating with directory servers during interoperation.

Aim: To implement LDAPS integration with CAN using private key access for self and third party LDAPS.

Requirements:

Requirement ID	Requirement description
REQ060127	Integration of CAN with LDAPS protocol. 1. Private Key to access the LDAP. 2. Allowing configuration to integrate with LDAPS.
REQ060128	Integration with third party LDAPS. 1. CAN gets access only to modify the Role.

40. Support for secure Kafka

Type	Functional
Priority	Important

Introduction: Apache Kafka is a framework that allows processing of streaming data. It is an open source platform developed by Apache Software Foundation and provides unified, high throughput, low latency platform for handling real time data feeds.

Aim: Integration of CAN application with Kafka broker optimizes customer operations of sending the alarm, ticket, and performance counter data in a streaming channel. This streaming interface allows CAN to subscribe to it and digest data in real time.

Requirements:

Requirement ID	Requirement description
REQ060129	Accepting data using secured protocols from customers at the backend like SSL certificate, trust key password, etc.

41. Support for JSON data parsing

Type	Functional
Priority	Important

Introduction: Avanseus CAN input mapper is capable to parse XLSX and Delimited files. To improve the versatility, CAN is enabled to parse files of other formats that improves the operability of this application in any user environment.

Aim: To enable Avanseus CAN to parse JSON file format.

Requirements:

Requirement ID	Requirement description
REQ060130	UI support to select JSON file format in the input mapper.
REQ060131	Back end modifications to parse JSON files.

42. Support for XML data parsing

Type	Functional
Priority	Important

Introduction: Avanseus CAN input mapper is capable to parse XLSX and Delimited files. To improve the versatility, CAN is enabled to parse files of other formats that improves the operability of this application in any user environment.

Aim: To enable Avanseus CAN to support XML data parsing.

Requirements:

Requirement ID	Requirement description
REQ060132	UI support to select XML file format in the input mapper.
REQ060133	Back end modifications to parse XML files.

43. Support for multiple KPI values in a single record and vice versa

Type	Functional
Priority	Important

Introduction: Avanseus CAN parses the fault management data, performance data, etc. of telecom network where a single office code is mapped to a single fault code or performance KPI as characteristic to such data. In networks like FTTH, the fault of a single office code or equipment can be identified only in aggregation of multiple system KPIs. Hence, CAN must be able to group and split incoming data during data load based on user requirement.

Aim: To parse a single input data to multiple records under the same head and vice versa.

Requirements:

Requirement ID	Requirement description
REQ060134	Splitting of a single record into multiple records.
REQ060135	Grouping of multiple records into a single record.

44. Resource Limits for Kubernetes

Type	Functional
Priority	Important

Introduction: Avanseus applications run on Kubernetes platform. Applications run on Kubernetes platform were using the resources like CPU, RAM, etc. on real time basis without any strict bounds and applications were free to use as much resources as possible available on the host system. This sometimes led to a 100% of the resources being consumed and eventually brought down the host machine. Hence, as a standard strategy, resource limits are set for micro-services so that the overall health of the system does not compromise.

Aim: To set resource limits for micro-services application platform within the CAN application.

Requirements:

Requirement ID	Requirement description
REQ060136	Excel file provides hardware requirements for all the applications that run on Kubernetes.
REQ060137	File contains resource and limit specification for every application to run on Kubernetes platform.

45. Map based Display for PM KPI Threshold Breach Prediction

Type	Functional
Priority	Important

Introduction: CAN supports performance KPI monitoring of the network where the KPI threshold breaches are identified by the system and notified. The plotting of such breaches over a map using the GIS coordinates will enable quick understanding and hence ease of operations.

Aim: To plot the PM KPI threshold breaches over user interface using the GIS information. The users must input the latitude and longitude information to enable location identification. Such locations are plotted using drop down pins over the specific area map. On click of such drop pins will provide a quick glance of relevant information regarding such breaches.

Requirements:

Requirement ID	Requirement description
REQ060138	Display of PM KPI threshold breaches over map area, with all the relevant information and respective KPI details on click of a specific drop pin plotted
REQ060139	Zoom-out option to display the heat map of the breached KPIs.
REQ060140	On click of drop pin along with interested KPI, line chart next to map area containing actual vs predicted values
REQ060141	Clustering has to be implemented in map view to handle huge amount of drop pins.

46. Map based Display for Health Index Prediction

Type	Functional
Priority	Important

Introduction: Health Index is introduced to ensure greater understanding and control of equipment and service abnormalities. Health Index need to be made immune to sudden fluctuations in KPI values that can act detrimental to the overall efficiency of health index output. The plotting of such variations over a map using the GIS coordinates will enable quick understanding and hence ease of operations.

Aim: To plot the Health Index variations over User Interface using the GIS information. The users must input the latitude and longitude information to enable location identification. Such locations are plotted using drop down pins over the specific area map. On click of drop pins will provide a quick glance of relevant information regarding such variations.

Requirements:

Requirement ID	Requirement description
REQ060142	Display of Health Index variations over map area with all the relevant information and respective KPI details, on click of a specific drop pin plotted.
REQ060143	Zoom-out option to display the heat map of alarm occurrence.
REQ060144	On click of drop pin along with alarm, predicted health index value (ranging from 0-100) for the next 'N' slots should be displayed in the line chart.
REQ060145	Clustering has to be implemented in map view to handle huge amount of drop pins.

47. Filtering KPI Breach Predictions based on Site Priority

Type	Functional
Priority	Important

Introduction: CAN supports performance KPI monitoring of the network where the KPI threshold breaches are identified by the system and notified to relevant stakeholders. Few office codes will have high priority because of its geographical significance. This feature defines the threshold boundaries along with the impact for different technologies with the user-friendly UI support.

Aim: To provide a comprehensive support in UI to display the fault KPIs. The fault services are mapped at the office code level. Office code contains Lat-Long and site priority. By default, KPIs of all three priorities are displayed.

Requirements:

Requirement ID	Requirement description
REQ060146	Filter Site Priority like Critical/Major/Minor to display the faults associated with office codes in both map view as well as in tabular view.
REQ060147	Ability to add/remove site priority field in filtered report.

48. Filtering Health Index Predictions based on Site Priority

Type	Functional
Priority	Important

Introduction: Health Index is calculated continuously and any downward trends or sudden dips are directed for immediate investigation. Alarm is triggered for a KPI for a specific reason. It is used for prediction and performance management of KPI. Few office codes will have high priority because of its geographical significance.

Aim: To provide a comprehensive support in UI to display the alarms. The health indices are mapped at the office code level. Office code contains lat-long and site priority. By default, alarms of all three priorities are displayed.

Requirements:

Requirement ID	Requirement description
REQ060148	Filter Site Priority like Critical/Major/Minor to display the faults associated with office codes in both map view as well as in tabular view.
REQ060149	Ability to add/remove site priority field in filtered report.

49. Viewing historical anomalies in real-time

Type	Business
Priority	Critical

Introduction: Anomaly is when a KPI is following a trend and there is a sudden change in this trend with a distinct value. DB maintains all the anomalies known as Anomaly History. The previous trend of a particular equipment component is the Historical trend. Viewing of historical trend of anomalies will support users to take appropriate decisions on network operations and maintenance.

Aim: UI support to display, search and download Historical anomalies. Anomalies are displayed in the line chart with proper indication.

Requirements:

Requirement ID	Requirement description
REQ060150	UI support to filter the duration of Historical Anomaly period in steps of 1 day, 3 days, 1 week and 2 weeks.
REQ060151	UI support to Download Historical Anomaly Report: Daily and Filtered.
REQ060152	UI support to filter interested equipment component or KPI.
REQ060153	Filter support KPIs based on Location, Domain, Network type, Service impact, Criticality, Site priority and Include inappropriate lat/long.

50. Shortlisting worst cells for KPI breach predictions

Type	Business
Priority	Important

Introduction: CAN application provides a user-friendly visual solution for presentation of network elements that are predicted with upcoming faults. The core AI of the CAN application shortlist the faults that are critical for the CAN users to take immediate note and appropriate actions. Worst cell calculation is one such method that would ease the work of CAN users as the application understand the worst cells and highlight the same for user actions.

Aim: Identifying worst cells among the threshold breach predictions based on data characteristics observed for individual KPIs with regard to 3 categories: Accessibility, Retainability and Quality

Requirements:

Requirement ID	Requirement description
REQ060154	Advanced Configuration Screen 1. Configurable Standard Deviation Threshold 2. Configurable Worst Cells Limit 3. Worst cell configuration is applicable only for Access domain. 4. Configuration of KPI and Standard deviation threshold for each network type separately. Worst cell configuration is based on 3 categories: Accessibility, Retainability and Quality.
REQ060155	Threshold Breach Screen 1. Option to filter worst cells. 2. Option to view worst cells: Tabular view and Map view. 3. Option to Search a worst cell.

51. Predictive fault analysis clustering map view to handle large number of markers and correction of inappropriate Lat-long

Type	Functional
Priority	Important

Introduction: Predictive Fault Analysis screen through its map view enable easy understanding of the predicted fault distribution across the monitored region. For a particular region, when the number of predicted faults are high, the application must have options to display the predicted fault in a scaled manner in a way that can decipher the maximum details with minimum intervention. In addition, the Lat -Long details must be validated in order to avoid inappropriate mapping of network elements.

Aim: To enable zoom-in and zoom-out options in map view to accommodate the mapped network elements. Nearby network elements to be shown as clusters that will be resolved as the user zoom-in. Lat-Long values input provided by the user to be validated against the regional boundaries to identify inappropriate Lat-Long

Requirements:

Requirement ID	Requirement description
REQ060156	"Enabled" switch is to be provided to set the view of inappropriate lat-long. It is set to false by default that indicates that only appropriate markers are to be displayed
REQ060157	Option to download the Inappropriate Lat-Long excel to check which office codes have inappropriate lat-long.
REQ060158	Nearby network elements are to be shown as clusters which on zoom-in shows the markers of that area.
REQ060159	On click of a cluster, the area is expanded and the markers in that cluster will be displayed.

52. Lat-Long Validation

Type	Functional
Priority	Important

Introduction: The predicted faults are displayed in the map view using the latitude and longitude values provided at office code level. Lat-long can be updated and validated from the site management screen. Validation happens at nation level and zone level.

Aim: Lat-long is required to highlight a fault in clustering like cross-domain correlation, predictive fault analysis or performance prediction. Lat-long can be added in UI or uploaded from a file. Uploaded values cannot be removed in UI but only updated. Lat-long outside the specified region is Inappropriate Lat-Long. CAN schedules a job to validate lat-long at the configured locality and it should be set to Today/Tomorrow at a particular time 10 minutes ahead of the Update button click.

Requirements:

Requirement ID	Requirement description
REQ060160	Lat-Long values can be empty or both values are mandatory.
REQ060161	Option to download the Inappropriate Lat-Long excel which contains only the invalid Lat-Long information. It is validated and uploaded in the site management screen.

53. Lat-long mapping at office code level

Type	Functional
Priority	Important

Introduction: Latitude and Longitude details are essentially required for CAN application to display the predicted faults in the map view. Earlier versions of CAN application had lat-long mapping at equipment component level. As the CAN application advances, for better resolution and representation, the lat-long need to be mapped at office code level.

Aim: UI support to define Lat-Long at office code level.

Requirements:

Requirement ID	Requirement description
REQ060162	Lat-Long values can be empty or both values are mandatory.
REQ060163	Option to edit data manually or using File Upload. The format and maximum size of the excel file to be uploaded are displayed.
REQ060164	CAN sends success or failure mail after uploading a file.
REQ060165	Option to Download the report that includes Office Code details.
REQ060166	Option to Search a specific office code.

54. Enhanced filter for cross domain correlation (location, office code, equipment, domain)

Type	Functional
Priority	Important

Introduction: In previous versions of CAN application, cluster display was limited only to zone based filtering. As CAN 6.0 provides an enhanced clustering algorithm through topology discovery, more filters need to be included to filter out the comprehensive results as per the user requirement.

Aim: Enhanced filters in cluster view with options to filter clusters based on locality type, domain, equipment and cause along with bit pattern view and map view.

Requirements:

Requirement ID	Requirement description
REQ060167	Option to select Locality Type.
REQ060168	Option to select clusters based on the selected locality type.
REQ060169	Option to filter clusters based on Domain, Equipment and Cause.

55. Enhanced bit pattern view for topology based clustering in topology discovery screen

Type	Functional
Priority	Important

Introduction: As the clustering algorithm is advanced through topology based clustering, the bit pattern view of the CAN application, where each bit represents a time frame/window, also need to be enhanced to accommodate the additional network components that are considered by the application. Each topology clusters are identified using Topology based clustering algorithm, each topology clusters will have common bits/time frames where faults are occurred together

Aim: The backend algorithm determines clusters that occur in future. A cluster has Cluster ID and cluster components called Vector. Each vector has details about equipment and cause. To group the vectors, all the vectors that have failed together are matched and displayed. 0 indicates No failure and 1 indicates Failure occurrence.

Requirements:

Requirement ID	Requirement description
REQ060170	Topology based clustering algorithm implementation.
REQ060171	Displaying Matching bits/time frames for topology-based clusters in UI.
REQ060172	Matching Bit: On click of toggle button, filters matching bits and displays bit pattern.
REQ060173	On hover over a highlighted bit, displays the details of fault occurrence period.

56. New filter algorithm to enhance precision and coverage

Type	Business
Priority	Important

Introduction: The two aspects of a prediction are Precision and Coverage. New filter rules to enable enhancement on existing algorithm to control precision and coverage. Older rules also to be maintained in the due course of the application.

Aim: UI support for generate predictions with enhanced precision and coverage and corresponding back end changes.

Requirement ID	Requirement description
REQ060174	Allowing the users to modify the rule in IDE
REQ060175	Option to select, edit and delete rules.

57. Topology Stitching & Topology Linking with Prediction

Type	Business
Priority	Critical

Introduction: CAN application predicts the upcoming network faults and performance. Each network element that is predicted to have issues are clustered and mapped over user interface to have an easy representation and understanding of overall network that is being monitored. Topology Discovery is an enhanced method by which the end-to-end network is discovered from the inventory data provided by customer and displayed, that will further enhance the operation of CAN application.

Aim: Based on the inventory data, to perform topological connections across multiple domains (RAN, Transmission, IP and Core) and to display the same. Connection traverse from RAN → Transmission → IP → Core (If provided). There may be single or multiple active links between any two nodes. Two views to be made for all the nodes: Map view and Schematic view. When customer provides Lat-Long values for the nodes, map view is displayed. Even if Lat-Long values are not provided, schematic view will always be displayed and co-located nodes are displayed with dotted area.

Requirements:

Requirement ID	Requirement description
REQ060176	Create data mapping for inventory file upload.
REQ060177	Inventory maintenance for Radio, Transmission, IP and Core with UI support.
REQ060178	Topology Stitching Configuration - support configuration of stitching parameters (cross-domain connection, traversal configuration & termination condition), equipment component extraction and inventory type (network element types & link types) standardisation.
REQ060179	UI support to display Topology Stitching using schematic & map views and to display the same by searching source node based on domain. Highlighting source node in both the views.
REQ060180	Predictive fault analysis screen to be grouped based on clustered faults and single faults (i.e. standalone faults). UI support for clustered fault & single fault to link topology using network topology view. Support regular & extended neighbourhoods for network topology.

Requirement ID	Requirement description
	Group clustered faults based on PAT number & PC rule and group single faults based on PAT number only. For clustered faults, display of clustered fault history and supports schematic and map view for network topology. Analyse port & link faults and highlight parent cause node, nodes in clustered fault history, predicted nodes and link faults. For single faults, supports schematic and map view for network topology. Analyse port & link faults and highlight predicted nodes and link faults.
REQ060181	Adding topology sequence to the prediction report.

58. Data type restriction on record parser IDE

Type	Functional
Priority	Important

Introduction: File upload/data upload is a critical activity in the operation of CAN application during which sanity of the parsed data is essential for the efficiency of prediction reports. Hence, necessary restrictions need to be in place to avoid junk data/unwarranted fields to interfere with the actual data upload. In addition, this mechanism must ensure minimum rejections during data upload.

Aim: To enable IDE based data type restriction for seamless data upload.

Requirements:

Requirement ID	Requirement description
REQ060182	Return type is restricted in IDE based on the field being configured.
REQ060183	Object returning functionality is provided for CAN fields.

59. Data source grouping in record parser UI and back end changes

Type	Non-Functional
Priority	Desirable

Introduction: CAN processes network data from multiple sources and domains to generate prediction reports. Hence, it is imperative to provide UI support towards grouping of such input data that come from similar sources. This will enhance efficiency of operations of the CAN application.

Aim: To provide UI support for grouping similar type of sub data sources under a single data source.

Requirements:

Requirement ID	Requirement description
REQ060184	UI based grouping of similar type of sub data sources under a single data source.
REQ060185	Option to add and delete mappings for data load.

60. Master table caching for data load

Type	Non-functional
Priority	Critical

Introduction: CAN 6.0 is equipped with master table caching to improve the data upload time. In comparison to the previous versions, CAN 6.0 will have an edge for faster data upload and hence faster prediction.

Aim: To maintain a cache based search data that will complement the data upload.

Requirements:

Requirement ID	Requirement description
REQ060186	Caching of master tables according to the size and fields defined.

61. Identifying parent-child correlated predictions

Type	Functional
Priority	Critical

Introduction: CAN application is designed to deliver an enhanced understanding of the network future events by the way of predicted faults. Such predicted faults are generated on daily basis. Among these predictions, some of them will lead to critical root causes that must be prevented. Hence, identifying such critical root causes for taking appropriate actions is critical to network efficiency.

Aim: To identify critical root causes using Parent- child correlation rules. Root cause associated information like parent causes, parent node, recent correlated fault history and action that needs to be taken at parent node to be generated and shared in daily prediction reports.

Requirements:

Requirement ID	Requirement description
REQ060187	Parent child rule configuration.
REQ060188	Identifying Parent-Child correlated predictions based on historical data.
REQ060189	Identifying parent nodes and parent causes.
REQ060190	Identifying affected nodes.

62. PAT generation configuration

Type	Functional
Priority	Important

Introduction: PAT is a unique predictive ticket identifier which is required to track the predictions. Multiple predictions are clubbed to generate a single PAT. In-order to group multiple predictions there should be a common point among predicted faults. Older versions of CAN has Equipment based PAT generation and it was not configurable. In CAN 6.0, this PAT generation level/logic is made configurable.

Aim: Allowing user to configure PAT generation level/logic from the UI, based on the customer requirement. *Example:* Office code, office code & cause category, equipment, equipment & cause category, PC-correlated predictions etc. Providing interface to write custom logic for a new way of grouping.

Requirements:

Requirement ID	Requirement description
REQ060191	Allowing user to select PAT generation configuration from UI.
REQ060192	Allowing user to write complex logic if new way of PAT grouping is required.
REQ060193	Backend support to populate PAT based on PAT configuration provided in the UI.

63. Integration with Redhat Ansible

Type	Functional
Priority	Important

Introduction: Ansible is an automation tool that provisions configuration management, application deployment, orchestration, and many other processes. It is a plug and play activation system where a set of commands having a workflow is defined. It runs on various interfaces. SSH is the widely used interface. CAN uses Ansible framework to connect with various interfaces for executing high-level commands. These commands are executed remotely through SSH.

Aim: To make Avanseus CAN compatible with Ansible framework. In workflow configuration, user can create multiple jobs and each job has multiple tasks. Rule configuration contains mapping between equipment, cause and job configuration. Ansible integration allows rule configuration that are required for workflow execution, when prediction is ready. It also provides preventive measures and remedial actions configuration remotely from CAN to the equipment or network management system via a network access.

Requirements:

a) Ansible Configuration

Requirement ID	Requirement description
REQ060194	SSH is a production ready interface. To create a workflow configuration, SSH certificate key file must be uploaded to establish SSH connection during run time. Destination IP and SSH username must be specified.
REQ060195	Provision of Local destination for trial task execution.
REQ060196	Task details must be provided for task creation. Task details include Task Name, Parent Task, Command, Process Output and Roll Back fields. There is no Parent Task for the first task of the workflow. When the Process Output switch is ON, option to write code to process the output.
REQ060197	Option to Add and Delete a task in UI.
REQ060198	To create a rule configuration, the workflow created is linked with equipment component & cause combination.
REQ060199	Workflow diagrams are developed on HTML Canvas.

b) Backend Execution

Requirement ID	Requirement description
REQ060200	Jobs that are configured in rule configuration will be executed for predicted faults having equipment component & cause combination.
REQ060201	Only a subset of predicted faults having rules configured will have automatic actions.
REQ060202	Multiple tasks are executed when configured tasks are at the same level.

c) Analysis

Requirement ID	Requirement description

REQ060203	Jobs are grouped and displayed based on date.
REQ060204	On click of each Job Id, display of executed tasks workflow diagram to show the job execution flow.
REQ060205	On click of each task, display of task execution details in a popup.
REQ060206	Task configured below a failed task is not executed.
REQ060207	Option to search a job based on job id or job type.
REQ060208	Highlighting the failed task on the workflow.
REQ060209	Workflow diagrams are developed on HTML Canvas.