ARUP

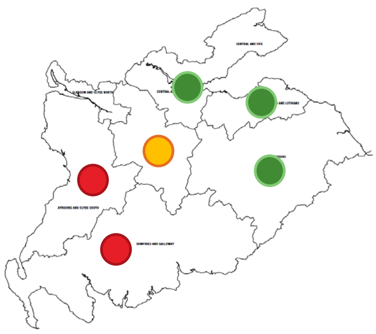
Scottish Power Energy Networks

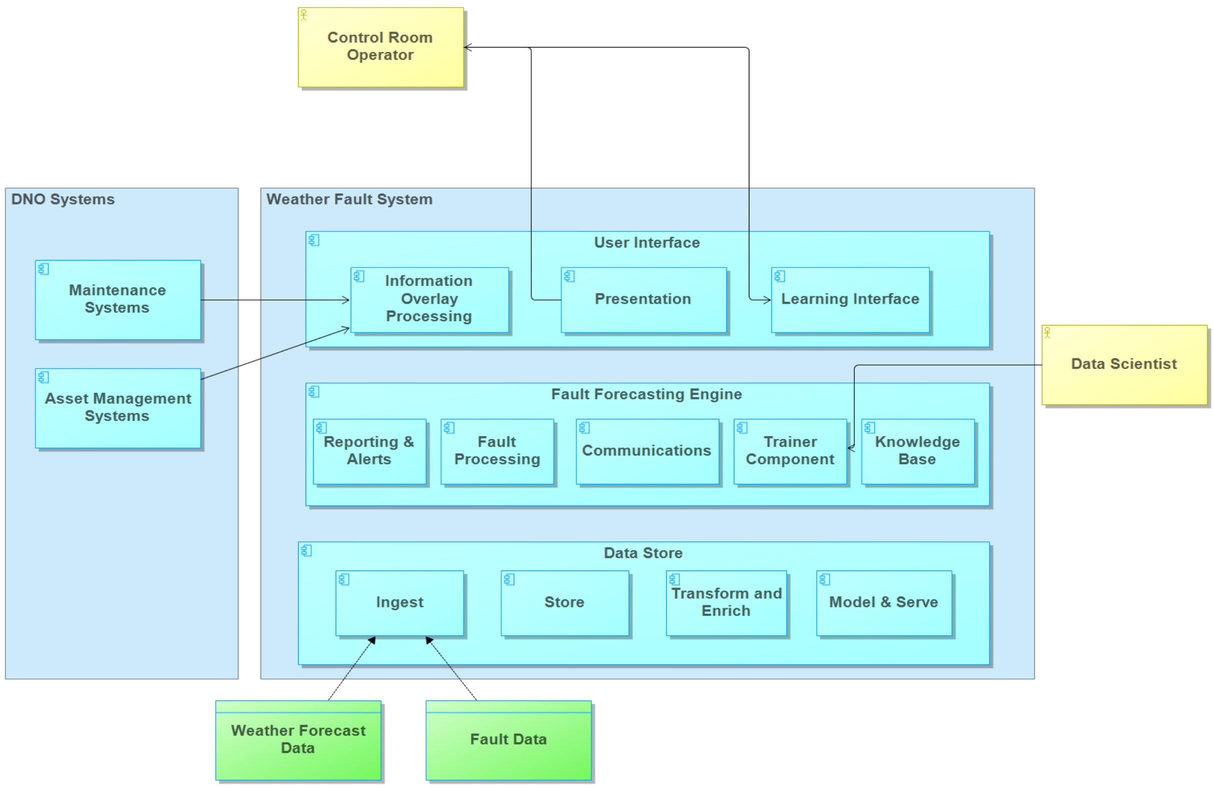
Predict4Resilience

Functional Specification

Reference:

ISSUE 1.0 | 16 May 2022





Job number

Ove Arup & Partners Ltd
8 Fitzroy Street
London
W1T 4BJ
United Kingdom
arup.com

ARUP

Document Verification

Project title	Predict4Resilience
Document title	Functional Specification
Job number	286969-00
Document ref	
File reference	

Revision	**Date**	**Filename**			
Draft V0.1	19 April 2022	**Description**	Draft for review and comment		
			Prepared by	**Checked by**	**Approved by**
		Name	Robert Morgan	Yutian Zhou	Adrian Hides
		Signature			
ISSUE 1.0	16 May 2022	**Filename**			
		Description	Final for issue		
			Prepared by	**Checked by**	**Approved by**
		Name	Robert Morgan	Yutian Zhou	
		Signature			
		Filename			
		Description			
			Prepared by	**Checked by**	**Approved by**
		Name			
		Signature			

Issue Document Verification with Document ✓

Contents

Tables

Figures

Drawings

No table of figures entries found.

Pictures

No table of figures entries found.

Photographs

No table of figures entries found.

Attachments

No table of figures entries found.

Appendices

1. Introduction

1.1 Project Background

Severe and extreme weather events have a major impact on the electricity network, resulting in widespread network outages for significant periods of time. Evidence has shown that climate change has contributed to longer and hotter heatwaves, more persistent droughts, more frequent wildfire, and more extreme rainfalls. While we cannot control the weather, we can seek to predict it more accurately with longer visibility and identify its impact in order to protect the electricity supply to consumers. This is particularly important for faults generated due to wind speed - which presents the greatest risk to the network.

The Predict4Resilience project intends to use the recent advances in supercomputing and numerical weather prediction to combine relevant datasets[1] and novel statistical processing to provide the capability to predict short-term adverse weather impacts further into the future, identify weather-related faults within a 7 to 14-day window, and predict resultant faults on the network. As a result, engineers will make informed decisions based on actionable fault and risk predictions communicated to the control room. Currently the business acts 3 days out as this is traditionally the time when weather forecasts are sufficiently accurate to make decisions that affect the operational response.

The Weather Fault System delivered by the Predict4Reilience Project will be provided as a commercial service by a design, build, operate (DBO) supplier for the benefit of all electricity distribution network operators in the UK (and potentially overseas).

1.2 Purpose of Document

The purpose of this document is to provide an initial functional specification for the Weather Fault System. The document presents user stories, business and technical requirements for the system and captures thinking to date that can be used to inform future stages of the project.

The system building blocks presented in this functional specification are conceptual and describe core capabilities of the system. They are presented to assist with the future identification and selection of the underlying technologies and software components that will make up the system. The system building blocks are not intended to reflect the final architecture of the system nor its underlying software – but to inform the future design and to provide insight and guidance for designers – describing characteristics of the system that need to be achieved to assure success. Additional building blocks may be identified in future stages of the project.

The requirements presented in this document form a baseline set of requirements on which to further develop the user needs. In future project stages, requirements will need to be prioritised as part of the agile development of the system - as part of a broader requirements management process established to provide continuous iteration and deployment of the product to users.

1.3 Project Objectives

The objective of the project is to develop and deliver a Weather Fault System for electricity Distribution Network Operators (DNOs). Initially the system shall be developed for use by SPEN and will be tested and evaluated in an operational environment during the Beta stage of the project. Following success of the pilot, the system may then be brought into full service and offered to other DNOs. In addition, the project will make the weather fault dataset openly available for others to access. This may include other organisations in the electricity industry and/or third parties, such as: local government, academics, and electricity customers.

[1] Relevant datasets include weather forecast data, satellite/topographic data (e.g: vegetation), asset and other datasets that may be used to calculate the risk of network faults occurring to the electricity distribution network due to a weather event.

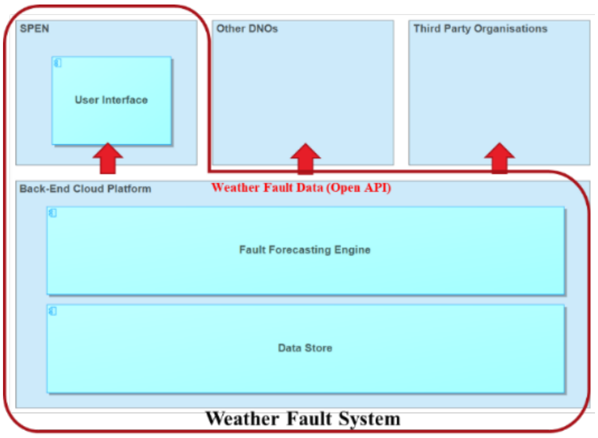


Figure 1. Vision for the Weather Fault System

The Weather Fault System is expected to provide the following benefits:

- **Improve** the ability of control room engineers and asset managers to protect network assets, and enable district emergency management teams to be better informed/prepared for forecasted weather events that may impact the network;
- **Increase** the efficiency and effectiveness of operational practices through the use of a data driven approach to fault prediction, e.g.: improve the proactive internal/external communication with staff and stakeholders who could be impacted by any potential power outage;
- **Reduce** customer minutes lost by providing early access to credible forecasted fault information. This will enable resources and equipment to be effectively distributed ahead of time - and to support a rapid response to incidents as they occur;
- **Reduce** costs associated with cancelling planned outages when necessary and if appropriate in weather impacted districts; and
- **Reduce** the carbon footprint of the business by avoiding unnecessary travel as a result of power engineers and contractors enacting response plans in preparation for fault events that do not occur.

1.4 Definitions

The following definitions apply to this document:

DBO Service Provider	The supplier responsible for the design, build and operation of the System – including the maintenance/support of the system and its on-going development.
DNO	An electricity Distribution Network Operator in the UK or overseas.
Faults	Incidents and/or events that occur on the electricity distribution network as a direct result of weather experienced. Such incidents or events that are identifiable by the SCADA system monitoring and controlling the network.
SPEN	Scottish Power Energy Networks
System	Means the Weather Fault System as described in this Functional Specification
Weather Alert Threshold(s)	A weather event capable of generating network faults greater than a predefined threshold. Such threshold relevant to each DNO, able to be specified by the user and varying depending on resource availability and probabilities of occurrence.
Weather Fault API	Weather fault data provided via open data API for use by DNOs and other organisations or individuals.

Table 1. Table of Definitions

1.5 System Modelling Notation

The architectural building blocks included in this document have been produced using the Archimate meta-model standard albeit tailored to meet the requirements of this functional specification.

The main components of the meta-model used are as follows:

Component	Description
Data Object	A Data Object is used to depict a dataset or a data entity used by the System. Data objects are also used to show how data flows between different components of the System.
Application Component	An Application Component is used to describe a logical functional component of the System that delivers core capabilities.
Application interface	An Application Interface is used to highlight critical interfaces that cross System boundaries. These interfaces will require development during Alpha/Beta stage and are identified and broadly described in this specification.
Business actor	A business actor may be a human or machine actor within the business but is typically used in this specification to show a business user of the system.
Business Object	An information artefact used by a Business Actor. This could be a report, schedule, diagram etc produced as an output of the System.

Table 2. Archimate Meta-Model Notation

2. Standards

At this (Discovery) stage of the project a limited number of standards have been identified. This is because listing standards may unnecessarily constrain the later design and development of the System by the ultimate DBO Service Provider.

2.1 General Standards / Best Practice

The System shall meet the following general standards which have been identified during Discovery stage. Additional standards may be identified during Alpha Stage during engagement with other DNOs and as a result of the further development of the technical solution:

- The System shall be designed and built to open and widely recognisable standards and shall avoid (wherever possible) lock-in to particular suppliers/vendors or proprietary protocols.
- The System shall be designed so that it is initially scalable across the 6 DNOs in the UK without major re-architecting or reconfiguration of system building blocks (notwithstanding the requirement to host DNO specific data).
- The Weather Fault Data set shall be indexed via Open Energy (https://openenergy.org.uk) and make use of the metadata standard for indexing data to Open Energy Search: (https://docs.openenergy.org.uk/1.0.0/metadata.html#data-set-metadata)
- The System shall be developed using modern application approaches based on:
    - Cloud native architectures using loosely couple microservices, managed databases, DevOps support and built-in monitoring
    - Integration using out-of-the box connectors and APIs
    - Utilisation of DevOps tools and fully managed platforms to help optimise the application
    - Built in multi-layered security and threat intelligence

2.2 Data Standards

The System shall be capable of processing Weather data using the following gridded data formats:

- netCDF
- GRIB

Additional information is contained within the following reports:

- "Predict4Resilience: Sample Data And Exploratory Analysis", 13 April 2022, V1 First Draft.
- "Predict4Resilience: Discovery Phase – Review of existing ensemble numerical weather prediction products", April 2022, Version 1.

3. Functional & Performance Requirements

3.1 Overview

The System will primarily be used by control room engineers and Emergency Action Centre management teams who will be provided with a forecast of the expected number of Faults generated by a weather event. The system will also provide additional contextual information that will enable an effective response plan to be developed with the overarching objective of minimising the customer minutes lost (CML) due to network outages. The System will improve the operational preparedness to manage events and incidents which can be predicted in advance and allow for more proactive communication in advance of a weather event with both internal and external stakeholders.

The System shall provide the following primary capabilities:

- Predict Faults that may occur due to a given weather forecast together with an associated probability of occurrence.
- Provide Fault predictions for Control Room Operators and Emergency Action Centre management teams across variable forecasting horizons from 14 days up to a day ahead.
- Provide information to assist with response planning and decision making.
- The System shall predict Faults for all districts of the SPEN electricity distribution network comprising:
    - **Scotland**: Central & Fife, Glasgow & Clyde North, Lanarkshire, Ayrshire & Clyde South, Edinburgh & Borders, Dumfries & Stranraer
    - **England**: Merseyside, Wirral and Mid-Cheshire
    - **Wales**: North Wales, Dee Valley & Mid-Wales
- The System shall predict Faults at the district level, with options for finer grained insights for network assets likely to be more vulnerable to specific hazards.

3.2 Stakeholders and Users

The System will be designed to meet the needs of users across the DNO business with a view to improving decision making and contributing to efficient and effective operations. During the project Discovery stage, potential users were interviewed within SPEN to develop an initial set of requirements to inform the design during the Alpha and Beta stages. These users are described in the following sub-sections.

3.2.1 Primary User – Control Room Engineers / Emergency Action Centre Management Teams

Context

This user is based in an office / control room. They will use a laptop/desktop on corporate LAN.

Control Room Engineers currently make use of a bespoke application called PRAE (Predictive Analytics of Energy) which is an electricity demand and generation forecasting platform. The ESRI Geographic Information System is also used by the business and is considered a core platform through which to present network related information/data.

Considerations

Line of business users such as Control Room Engineers use information to enhance how they perform their day-to-day and make decisions to predict events, inform events, and allocate resources. These users generally work to a short-term horizon which means they want to plan for one to five days out (i.e.: the decision making timeframe).

These users should be provided with a forecast of the expected number of faults, and probability of occurrence, based on a weather forecast. Associated information such as weather details, response team allocations, planned works and historical insights (e.g.: number of faults expected during similar weather conditions) that will enable them to develop an effective response plan.

These users need information that is concise and visuals that don't overwhelm. It should be easy to comprehend and specific to their purpose. If the dashboards or reports are too cumbersome, then they simply won't use them and continue to rely on intuition. For these types of users, information normally comes from a variety of sources, or a gut intuition based on experience. As a user explores this application (selects more details of a region or fault) more information should be expanded on.

A goal of the Alpha stage pilot will be to determine what is the right amount of information, presented in the right view at the right time. Giving Control Room Engineers and emergency action centre management teams access to information depending on the sequence of how they work / complete tasks will help drive retention and ongoing use of the System.

3.2.2 Secondary User – System Administrator

Context

This user is based in an office and is part of the DBO Service Provider.

Considerations

This user will need to set up new instances of the System for DNOs.

3.2.3 Secondary User – Data Scientist

Context

This user is based in an office or remote setting

Considerations

This user will need to access historical performance data and system logs in order to improve the application and modelling.

3.2.4 Secondary User – Operational Planning Engineer

Context

This user is based in an office/control room environment and works under the supervision of the Control Room Engineer.

Considerations

This user may need specific scenario based training to better learn how to use the application and/or shadow a primary user with existing knowledge.

3.2.5 Governance User – IT Team

Context

IT services procured by a DNO will be subject to DNO IT governance – this will include the Weather Fault System offered by the DBO Service Provider.

Considerations

The System will need to be designed to adhere to the policies and standards of DNO IT functions. Due to the fact that the vision for the System is that it will be provided as a service to DNO's, application policies are likely to relate to the SLA for the service, the underlying cybersecurity of the system, and the ability of data to be able to be incorporated into the datalakes/warehouses of the DNO (for broader data mining).

3.3 User Stories

Based on research with primary users, requirements have been organised into high level "epics" (or scenario level) and "user stories" (task level). The epics comprise:

- **Current Conditions** – understanding the current status of the energy network;
- **Predicting Conditions** – understand faults that may be generated due to a forecasted weather event;
- **Ongoing Learning** – supporting the training, development and learning of control room engineers;
- **Administration** – relating to the administration and management of the Weather Fault System within, and across, DNOs.

3.3.1 Epic 1: Current Conditions

The following initial list of user stories has been identified:

#	*As a Control Room Engineer I want to…*	*So that I can…*
1001	*View a list/map of the number of network faults that are predicted to occur within a district due to the latest weather forecast.*	*Understand potential resource needs*
1002	*View a list/map providing information how the number of network faults vary over time due to the latest weather forecast.*	*Understand potential resource needs*
1003	*Sort/select by region for more details on the number of network faults in that region*	*Understand region specific need*
1004	*Understand the range of the number of potential network faults that may occur within a region – including the probability associated with the forecasted number of network faults.*	*Consider the likely impact of a weather event*
1005	*View a summary of the weather forecast*	*Better predict fault impact*
1006	*View a summary of the dominating weather event*	*Better allocate resources*
1007	*Sort/select by Region for more details on local weather conditions e.g. wind speed, type of weather,* weight of snow, temperature)	*Understand details as they impact the plan*
1008	*View planned work/ ongoing customer effecting work*	*Understand workload to inform the incident response plan*
1009	*View planned customer effecting work thresholds (max, med, min numbers)*	*Understand workload*

1110	*View a list/map of all active response teams*	*Understand workload*
1111	*View a list/map of all planned response teams*	*Understand workload*
1112	*Sort/select by region for more details on response teams assigned to faults*	*Understand workload*
1113	*Edit/save details on work teams assigned to faults*	*Adjust staff allocation*
1114	*View changes to assets in regions to provide some context for why faults may have increased or decreased*	*Understand asset history*
1115	*Log / Save my assessment/actions due to the forecasted number of network faults. (e.g.: category event, response team plan)*	*Capture my decision*
1116	*Log / Vote on if the tool impacted my decision / actions*	*Capture my decision*
1117	*Share insights or planning tools with colleagues (e.g.: response managers, district managers)*	*Better inform others who may need to act*

3.3.2 Epic 2: Predicting Conditions

The following initial list of user stories has been identified:

#	*As a Control Room Engineer I want to…*	*So that I can…*
2001	*Adjust time (i.e. day to storm) and type of weather to view how the forecasted number of network faults varies.*	*Model different scenarios*
2002	*Edit/ Tag a fault by Category Event*	*Alert colleagues to potential faults*
2003	*View a list of response plans based on category event*	*Better inform my plan*
2004	*Select a specific category event response plans*	*Better inform my plan*
2005	*View a category event response plan that provides common response scenarios based on my weather category event, time of week context, etc.*	*Better inform my plan*
2006	*View if seasonal checks in a region have been completed*	*rule out errors*
2007	*Alert a Senior Manager if Five Days out and forecast faults exceed the Severe Weather Alert Threshold*	*Inform others with key tasks*
2008	*Alert a District Manager if Three Days out and forecast faults exceed the Extreme Weather Alert Threshold*	*Inform others with key tasks*
2009	*Log / Save my assessment/actions due to a fault forecast (e.g.: category event, response team plan)*	*Learn post event*
2010	*Log / Vote on if the tool impacted my decision / actions*	*Learn post event*

3.3.3 Epic 3: Ongoing Learning

The following initial list of user stories has been identified:

#	*As a Control Room Engineer I want to…*	*So that I can…*
3001	*View a list of past event events by region, type, date, and/or engineer.*	*Select a past event*

3002	*Select event to review actual results (e.g: response team plan)*	*See specific results*
3003	*View assessment/actions due to a fault. (e.g: category event, probability score, response team plan)*	*See specific results/ learn to inform future planning*
3004	*Log / Vote on actual vs predicted*	*learn to inform future predictions*
3005	*Have a simulation mode*	*Test offline model improvement and enhancement*
3006	*View the performance of fault forecast*	*Understand / learn to inform decision-making based on the forecast*
#	*As a Data Scientist I want to…*	*So that I can…*
3201	*Review historical data and usage*	*Understand trends*
3202	*Log assessment of the performance of prediction models*	*Monitor performance / accuracy and learn to inform future improvement*
3203	*Calibrate prediction models*	*Improve accuracy*
3204	*Integrate new data sets or algorithms*	*Enhance the application*
3205	*Have a simulation mode*	*Test offline model improvement and enhancement*

3.3.4 Epic 4: Administration

The following initial list of user stories has been identified:

#	*As a System Administrator I want to…*	*So that I can…*
4001	*Set up a new instance of the Weather Fault application*	*Customize for new clients*
4002	*Add relevant branding / logos*	*Customize for new clients*
4003	*Assign user and permissions*	*Customize for new clients*

3.4 Performance Requirements

The performance requirements for the System will need to be investigated during the design stage and are expected to include the following considerations:

- **Time to Respond**. The time between when a user requests fault predictions to them being displayed to the User Interface. It is envisaged that the time to respond should be of the order of 0.5 seconds.
- **Bandwidth**. The bandwidth and resilience of connectivity between the back-end (cloud platform) and the front-end (user interface).
- **Compute**. The processing required to generate fault predictions from input data (weather, vegetation, assets etc)[2]. The computation should be completed in a manner that enables users to access information in a timely manner.

[2] It is envisaged that fault predictions will be calculated on a schedule – typically when new weather forecast information becomes available, eg: daily, every 6 hours, every 1 hour depending on the product.

- **Storage**. The type and size of storage required to store System data and – including the speed of transfer of data from storage to compute. The storage requirements for the System – including the technology infrastructure – will need to be developed once the statistical model for generating fault predictions is fully understood.

3.5 Availability Requirements

The availability of the final 'in-production' version of the System must align with user needs and operational requirements. The System shall be developed through Alpha and Beta stages to meet the following availability requirements:

- The System shall have a Recovery Time Objective (RTO) of 1 business day and a Recovery Point Objective of 1 month. That is, should the system fail, it will be restored and available for service within 1 business day, and loss of data shall be limited to 1-month.
- The System shall be available for 99.99% during core business hours – defined as 08.00hrs to 18.00hrs x 365 days per year. Note that this is not to say that the System will not be available to users outside of these hours, only that SLAs shall apply during this period. If the System fails outside of core business hours, it is expected that issues would be resolved on a best endeavours basis.
- Planned downtime for the System should be considered as a non-functional requirement during the Alpha/Beta stages – and informed by the architecture and design of the System.

Note that the availability requirements above may be varied during Alpha and Beta stage and are intended as a baseline for consideration.

3.6 Security Requirements

The cybersecurity of the final 'in-production' version of the System is important and shall be developed through Alpha and Beta stages. During the Discovery stage, the following initial security requirements for confidentiality, integrity and availability have been identified for further consideration.

3.6.1 Confidentiality

Confidentiality will ensure that authorised users only are able to access the System and its associated datasets. In addition:

- Databases do not need to be encrypted unless the data stored is considered personally identifiable information (PII) under UK GDPR regulation.
- Data in transit over the internet shall utilise secure transport protocols.

3.6.2 Integrity

The System shall be developed to ensure that authorised users only are able to perform system administration activities and access underlying databases.

- Access to databases shall be restricted to system administrators
- Default System passwords (e.g.: for access to databases, administration consoles etc) shall be replaced with secure passwords using best practice approaches.

3.6.3 Availability

The System shall be designed to support the RTO/RPO objectives. In addition:

- Databases shall be backed-up shall include stand-alone copies that are not directly connected to the System (the copy is air-gapped and stored off-site).

4. System Architecture

During the Discovery Stage, a number key of building blocks has been identified for the System and these are described in this section. Building blocks represent logical groupings of features/functions that reflect the core capabilities of the System and on which underlying technologies and solutions can be identified.

4.1 Overview

At the highest level, the System is expected to comprise a front-end application that the user engages with and a back-end processing/storage platform in the cloud for forecasting faults:

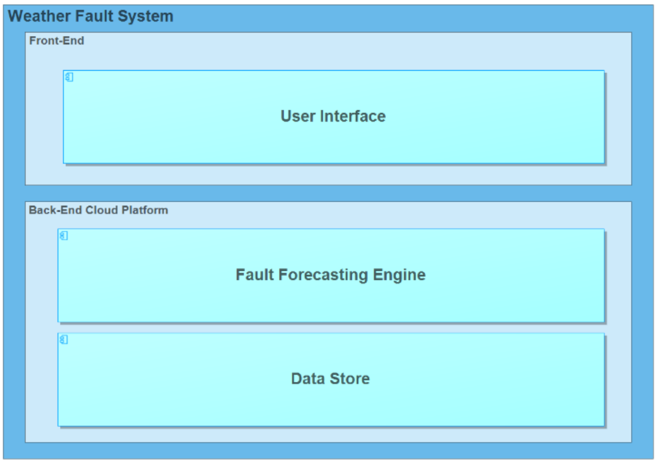


Figure 2. Weather Fault System Overview

The User Interface (UI) provides the front-end to the system. The UI presents Fault information to control room operators for each district managed by the DNO and exposes application functionality to users to enable them to manipulate information to assist with incident response planning. The Front-End is expected to communicate to the back-end cloud platform via the public internet using a secure, open standards API.

The Fault Forecasting Engine (FFE) performs the core processing functionality of the system, using various datasets to predict Faults. The Data Store provides a centralised, elastic, store of data for the System – securely segmenting each DNOs data from one another and offering it to the FFE for processing.

The components that make up each building block are described in Figure 3. Each of these provides specific capabilities which are explained in more detail in the following sub-sections:

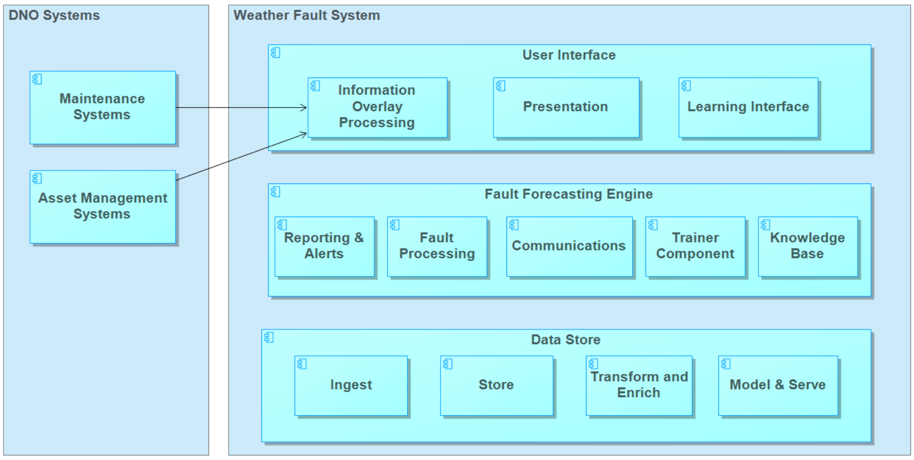


Figure 3. Weather Fault System - Core Building Blocks

4.2 User Interface

The User Interface may be a COTS application (such as ESRI GIS), further development of an existing bespoke application (such as the SPEN PRAE application); or a completely new bespoke application developed specifically for the System. This will need to be investigated further during the Alpha Stage as approaches will have different strengths and weaknesses. Whichever UI is selected, it should be able to offer as many of the following capabilities as possible (refer to Figure 4).

- **Information Overlay Processing**. This component shall receive data from third-party systems and from the fault forecasting engine and perform local processing required to send data to the Presentation component for display to the end user.
- **Presentation**. This component displays information to users and enables users to interact with the System. The user should be able to activate features and functions as expressed in the requirements catalogue (refer to Appendix) – such as varying prediction timescales, switching between prediction and simulation modes, entering information on the performance of the system, and initiating the production of various management reports.
- **Learning Interface**. This component manages/processes learning related data to the system and communicates to the Knowledge Base in the back-end of the System. Users shall be able to enter data on the actual number of faults that occurred as a result of a weather event (such that these can be used to improve the accuracy of the System over time). In addition, users will be able to enter and review historic information on the utility of the System – in terms of how accurate the System is and how the system contributed to the development of effective response plans.

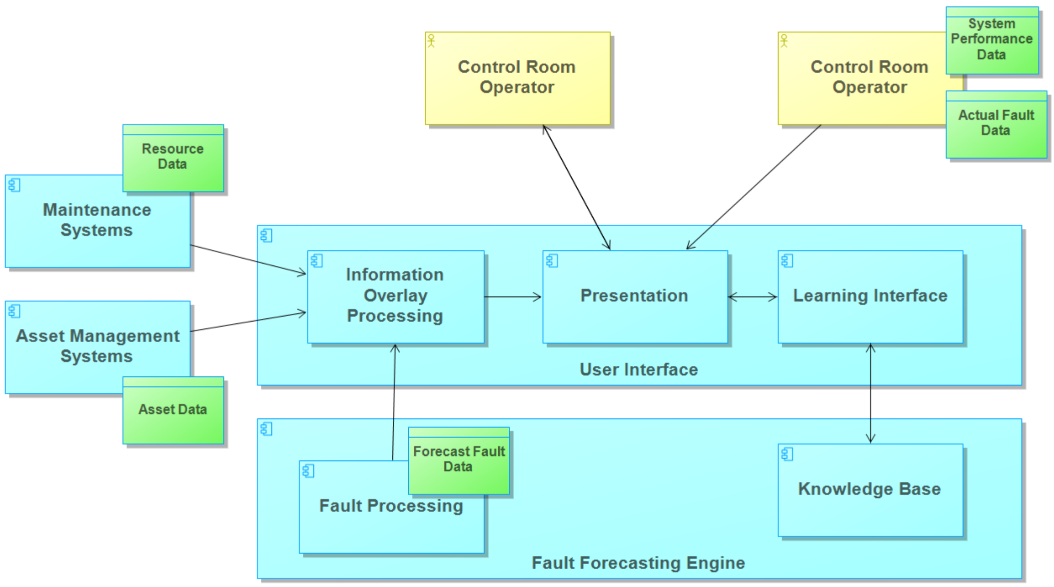


Figure 4. User Interface Building Block

The User Interface will likely vary between DNOs and, therefore, the Back-End Cloud Platform shall be able to pass data to the Front-End in an open standards format that can be integrated into a range of COTS or bespoke applications (refer to Section 4.5 System Interfaces.)

4.3 Fault Forecasting Engine

The Fault Forecasting Engine (FFE) provides the primary processing capability of the System (refer to Figure 7). There are at least two approaches to enabling the System to be used by multiple DNOs (others will be considered during the Alpha/Beta design stage):

i. The System could be architected to comprise a single FFE – with each component logically separate to share compute across DNOs; or

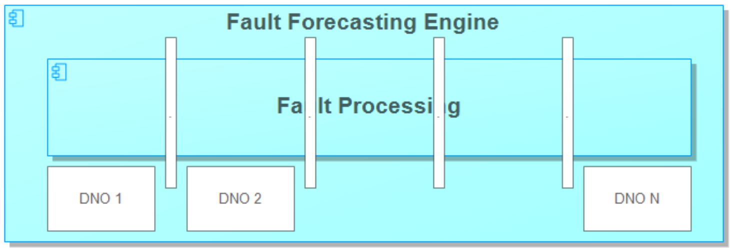


Figure 5. Logical Segmentation of the Fault Processing Component

ii. The System could be architected to comprise multiple FFEs – essentially repeating the building block for each DNO and providing dedicated compute.

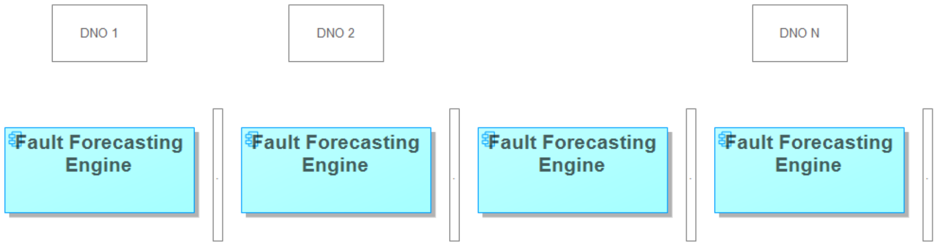


Figure 6. Logical Segmentation of the Building Block

The FFE is made up of the following major components:

- **Reporting & Alerts**. The System is required to produce management reports and automatically generate alerts if the number of faults predicted exceeds pre-set thresholds. Management reports will include assessments of the accuracy of the system and may include other performance related information such as the how the fault predications generated by the System successfully influenced the response plan.
- **Fault Processing**. The primary processing element of the System. Faults will be predicted using statistical modelling techniques given an ensemble weather forecast. Fault data will be passed to the Communications component for on-going transmission to the UI.
- **Communications**. This component is responsible for transmitting Fault predictions to the UI and for dealing with Weather Fault Data dataset requests from third parties using the open API.
- **Trainer Component**. It is expected that the Fault Processing component shall require tuning / training to enhance the accuracy of Fault predictions over time. The Trainer Component shall provide the Human-Machine Interface for Data Scientists responsible for the tuning/training activity.
- **Knowledge Base**. The component that processes learning/training data. The Knowledge Base shall output to the Reporting & Alerts component to provide reports on the utility of the System via Management Reports. In addition, the knowledge base shall be responsible for calculating the accuracy of the System from a comparison on Actual Fault Data and Forecast Fault Data (reference Section 4.4 Data Store).

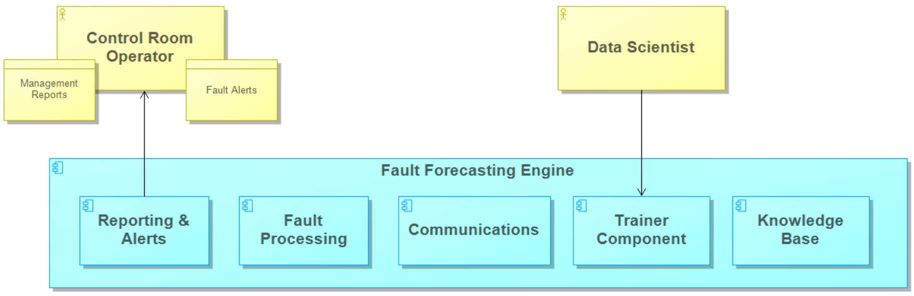


Figure 7. Fault Forecasting Engine Building Block

4.4 Data Store

The Data Store building block will ingest and store System data and perform necessary processing required to prepare and serve data to the FFE. A variety of datasets are expected to be required including those supplied by the DNO (e.g.: fault data used for training the FFE) and Weather/Environmental datasets used by the statistical model to predict faults.

The scope of datasets will be investigated further during Alpha Stage to ascertain their availability, cost and the performance improvement that can be obtained by incorporating them into the Fault prediction model.

The logical flow of data between the System components is described in Figure 8 which shows entity types for the following initial list of data entities:

Data Entity	Entity Type	Description
DNO Datasets	Training Fault Data	Historic fault data provided by a DNO and used to train the FFE.
Weather Datasets	Weather Forecast Data	Forecast weather data for an area covered by a DNO making use of the System.
	Training Weather Data	Historic weather data provided for the purposes of training the System.
Fault Datasets	Fault Forecast Data	Fault data generated by the FFE and forecasted to occur due to prevailing weather conditions.
	Actual Fault Data	Fault data entered by users as a record of the actual faults experienced due to a weather event.
Environmental Datasets	Satellite Imagery Data	Optical data that can be used as an input to the FFE and enabling risk calculations based on potential physical threats and their proximity to network assets.
	Lidar Data	High resolution optical data that can be used as an input to the FFE and enabling risk calculations based on potential physical threats and their proximity to network assets.
	Topographical Data	Data that provides information on the local terrain surrounding network assets and which can be used to enhance macro level weather forecasts. Freely available, low resolution options include SRTM.
System Utility Datasets	Accuracy Data	Calculated from a comparison of Fault Forecast Data and Actual Fault Data. This data shall contribute to the management reports produced by the Reporting & Alerts component.
	System Performance Data	Data that describes the performance of the system. This data is entered by the user and captures how well the system predicted faults and contributed to preparing response plans.

Table 3. Description of Data Entities

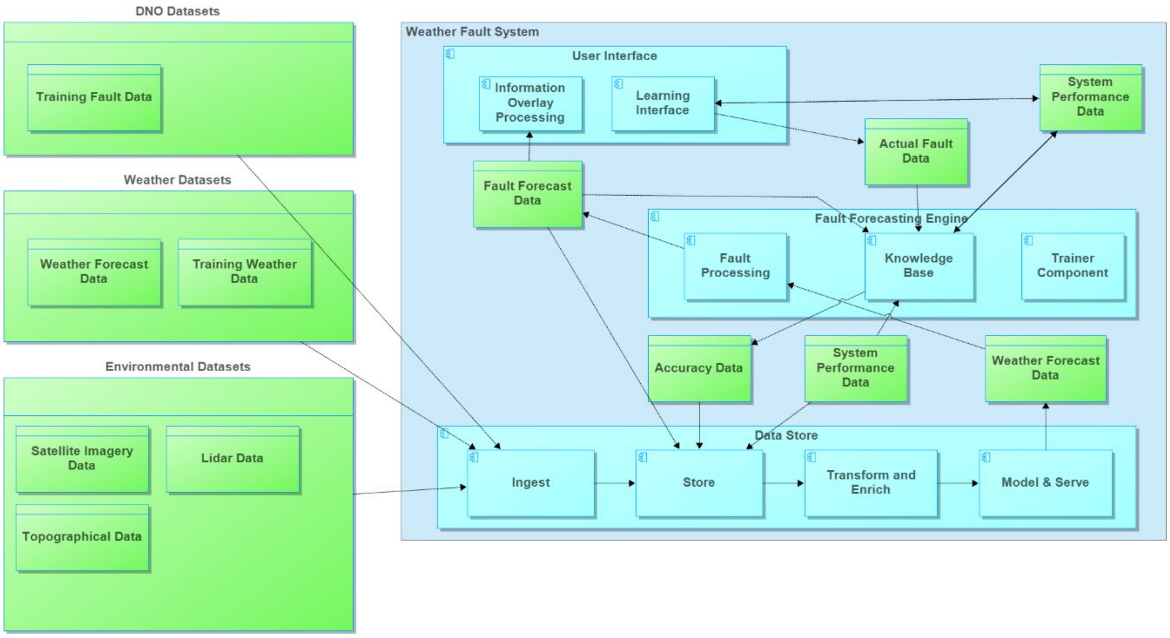


Figure 8. Datasets and Flows Between System Components

4.5 System Interfaces

System interfaces enable data to be exchanged across inter- and intra-system boundaries. System interfaces will be required to achieve the following functional outcomes:

- import data from third party DNO systems – to be determined during Alpha/Beta stage but expected to include Maintenance Systems and Asset Management Systems;
- communicate securely between the back-end and front-end of the system – most likely over the public internet and via organisational infrastructure such as LANs, firewalls etc; and
- enable third party organisation to access Fault Forecast Data via an openly accessible, yet secure, communications protocol.

The interfaces identified during the Discovery Stage are described in Figure 9 and described in more detail in Table 4:

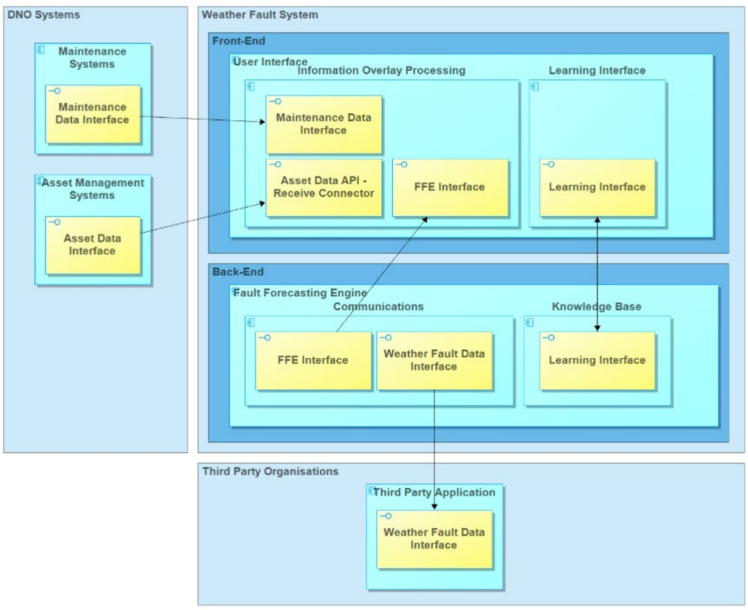


Figure 9. System Interfaces

Interface	Type	Description
Maintenance Data Interface	Inter-System Interface	Maintenance systems contain information on the availability of resources (human, equipment etc) to resolve events and incidents on the network. This information may be useful presented via the System UI such that users are able to contextualise the implications of fault predictions.
Asset Data Interface	Inter-System Interface	Asset Management systems contain information on the location and condition of critical network assets (substations, pylons etc). This information may be useful presented via the System UI such that users are able to contextualise the implications of fault predictions.
FFE Interface	Intra-System Interface	Fault predictions generated by the FFE will need to be transported to the Front-End in a secure manner over the public internet.
Learning Interface	Intra-System Interface	Information captured from users on the utility of the system will need to be communicated from the Knowledge Base in the back-end of the system to the Learning Interface in the UI.
Weather Fault Data Interface	Inter-System Interface	Weather Fault Data produced by the System should be openly accessible to Third Party Organisations via an open and standardise interface.

Table 4. Description of System Interfaces

5. System Dependencies

This section considers other aspects of the technology architecture that need to be in place and within which the System will be expected to operate. These include:

- Application Platform
- Operating System Services
- Network Services
- Communications Infrastructure

The System shall be designed and developed over the Alpha and Beta stages by the DBO Service Provider and, therefore, limited information is currently available for system dependencies, and these will need to be considered in later project stages.

5.1 Front-End Dependencies

The System front-end will be required to operate within the DNO environment and this will require further investigation with DNOs during Alpha/Beta stages.

Considerations may include:

- Windows Infrastructure Environment – Operating Systems, Servers, Active Directory Services
- Internet Browser – Chrome, Microsoft Edge, Firefox etc
- Virtualisation Infrastructure – Virtualised Applications and Servers

5.2 Back-End Dependencies

The back-end of the System shall be hosted on Amazon Web Services (AWS) or Microsoft Azure cloud services managed and maintained by the DBO Service Provider.

It is currently envisaged that cloud virtualised infrastructure shall utilise shadowed or mirrored storage for resilience. Cloud services shall be configured to support the availability requirements.

The System's use of the virtualized infrastructure shall permit the automatic backup (configurable) of data and the application components of the system.

Security components shall be configured on infrastructure components in accordance with the best practice recommendations of the cloud service provider.

Appendix A

Initial Requirements Catalogue

A.1 Requirements Catalogue

An initial requirements catalogue has been developed for the Weather Fault System. The catalogue records requirements that have been elicited to date and which may form an input to the Alpha stage. Requirements have been group into the following categories and mapped to the Epics presented in Section 3.

1. General	2. Technical	3. Functional	4. NonFunctional
Requirements that define business policies, standards and needs.	***Requirements that state the technical policies and constraints. May refer to technical infrastructure.***	***Requirements that set out the features that the application should provide.***	***Requirements that define how well the application will operate.***
1.1. Branding	2.1. Hardware	3.1. Data Entry	4.1. Archiving & Retention
1.2. Business Constraints	2.2. Interoperability	3.2. Data Maintenance	4.2. Availability
1.3. Business Policies	2.3. Software	3.3. Presentational	4.3. Back-Up / Recovery
1.4. Cultural		3.4. Procedural	4.4. Capacity
1.5. Language		3.5. Retrieval	4.5. Performance
1.6. Legal			4.6. Security
			4.7. Usability

ID	Category	Sub-Category	User Story	Requirement Description
1.1.1	General	Branding	Administration	The application branding must be reconfigurable for different DNOs, e.g., logo, colour scheme.
1.2.1	General	Business Constraints	Ongoing Learning	The application must have a simulation/ offline mode to model scenarios/ case studies.
1.2.2	General	Business Constraints	Predicting Conditions	The User needs to select specific category event response plans.
1.5.1	General	Language	Administration	The User must be able to configure the system language depending on the country of operation.
2.1.1	Technical	Hardware	Not Applicable	The Back-End Cloud Platform must provide common data processing capabilities for weather data.
2.1.2	Technical	Hardware	Not Applicable	The Back-End Cloud Platform must provide common data processing capabilities for fault data.
2.1.3	Technical	Hardware	Not Applicable	The Back-End Cloud Platform must provide common data storage capabilities for weather data.
2.1.4	Technical	Hardware	Not Applicable	The Back-End Cloud Platform must provide common data storage capabilities for fault data.
2.2.1	Technical	Interoperability	Not Applicable	The User Interface must be compatible with or make use of the ESRI GIS system.
2.2.2	Technical	Interoperability	Not Applicable	Weather fault information/ data must be output in an open standard format.
2.2.3	Technical	Interoperability	Not Applicable	The User Interface must be able to receive data from DNO systems via a defined set of APIs (or via open standard database connections).
2.3.1	Technical	Software	Not Applicable	The back end of the system must use elastic/scalable cloud technologies.
2.3.2	Technical	Software	Not Applicable	The Back-End Cloud Platform must communicate to the User Interface via the public internet.
2.3.3	Technical	Software	Not Applicable	The data store must support connectivity to structured and un-structured data sources.
3.1.1	Functional	Data Entry	Current Conditions	The user needs to be able to save data entry on response teams assigned to faults.
3.1.2	Functional	Data Entry	Administration	The User must be able to edit User permissions.
3.1.3	Functional	Data Entry	Administration	The User must be able to edit the recipient of automated notifications/alerts.
3.1.4	Functional	Data Entry	Current Conditions	The User needs to save their assessment/actions against fault predictions, e.g., category event, probability score, response team plan, etc.
3.1.5	Functional	Data Entry	Current Conditions	The User needs to log their assessment/actions against fault predictions, e.g., category event, probability score, response team plan, etc.

3.1.6	Functional	Data Entry	Current Conditions	The User must be able to input feedback on whether the Weather Fault System influenced/impacted decision making.
3.1.7	Functional	Data Entry	Current Conditions	The user needs to be able to edit information/data on response teams assigned to faults.
3.1.8	Functional	Data Entry	Administration	The User must be able to remove existing Users.
3.1.9	Functional	Data Entry	Current Conditions	The User needs to be able to capture tacit information on the number of faults that could occur as a result of a weather event.
3.1.10	Functional	Data Entry	Ongoing Learning	The User needs to be able to input data/information on actual faults that occur as a result of a weather event.
3.1.11	Functional	Data Entry	Ongoing Learning	The User must be able to input feedback on actual versus predicted faults.
3.1.12	Functional	Data Entry	Predicting Conditions	The application must allow faults to be manually edited/tagged by Category Event.
3.1.13	Functional	Data Entry	Predicting Conditions	The User needs to save their assessment/actions due to a fault forecast, e.g., category event, response team plan, etc.
3.1.14	Functional	Data Entry	Predicting Conditions	The User needs to log their assessment/actions due to a fault forecast, e.g., category event, response team plan, etc.
3.1.15	Functional	Data Entry	Predicting Conditions	The User must be able to input feedback on whether the Weather Fault System influenced/impacted decision making.
3.1.16	Functional	Data Entry	Administration	The User must be able to add new Users.
3.2.1	Functional	Data Maintenance	Ongoing Learning	The system needs to allow for ongoing calibration of the application/ artificial intelligence by a Data Scientist.
3.2.2	Functional	Data Maintenance	Ongoing Learning	The system must be able to integrate new data sets for future enhancements.
3.2.3	Functional	Data Maintenance	Ongoing Learning	The system must be able to integrate new algorithms for future enhancements.
3.2.4	Functional	Data Maintenance	Administration	The Data store must ingest data for each DNO.
3.2.5	Functional	Data Maintenance	Administration	The Data Store must model data for each DNO.
3.2.6	Functional	Data Maintenance	Administration	The Data Store must serve data for each DNO.
3.2.7	Functional	Data Maintenance	Administration	The Data Store must store data for each DNO.
3.2.8	Functional	Data Maintenance	Current Conditions	The Data Store must store actual fault data input by Users.
3.3.1	Functional	Presentational	Current Conditions	The application must show a confidence range of +/- 10% for predicted faults.
3.3.2	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by time, on both the macro and micro map view.
3.3.3	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering between a spatial view of weather faults per district and a temporal view of faults varying over time.
3.3.4	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by geographic areas to show information on local weather conditions, e.g., wind speed, type of weather, weight of snow, temperature, etc.
3.3.5	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by region for more details on response teams assigned to faults.
3.3.6	Functional	Presentational	Current Conditions	The application must show changes to asset/ asset history.
3.3.7	Functional	Presentational	Current Conditions	The application must show forecast wind speed, direction and time of year.
3.3.8	Functional	Presentational	Current Conditions	The application must allow Users to select a geographic area to see a micro view of the composition of faults within each region.
3.3.9	Functional	Presentational	Current Conditions	The application must show faults for a typical day against faults for a weather event per geographic area.
3.3.10	Functional	Presentational	Current Conditions	The application must list/map the number of planned response teams per geographic area as a range of upper, lower and mean figures.
3.3.11	Functional	Presentational	Current Conditions	The application must show weather forecasts up to five (5) days in advance.
3.3.12	Functional	Presentational	Current Conditions	The application must show fault predictions due to a weather event up to five (5) days in advance.

3.3.13	Functional	Presentational	Current Conditions	The application must show response centres per geographic area.
3.3.14	Functional	Presentational	Current Conditions	The application must have a visual User Interface.
3.3.15	Functional	Presentational	Current Conditions	The application must show the number of faults per geographic area.
3.3.16	Functional	Presentational	Current Conditions	The User Interface must show a map divided into all relevant recognised geographies.
3.3.17	Functional	Presentational	Current Conditions	The application must show the number of faults predicted within a 24 hour period.
3.3.18	Functional	Presentational	Current Conditions	The User Interface must show weather events/ faults per region as a heat map.
3.3.19	Functional	Presentational	Current Conditions	The application must establish the correlation between Overhead Lines (OHL) and Underground cables (UG).
3.3.20	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by region, on both the macro and micro map view.
3.3.21	Functional	Presentational	Current Conditions	The application must show the breakdown of predicted faults by cause.
3.3.22	Functional	Presentational	Ongoing Learning	The user needs to view assessment/actions due to a fault forecast, e.g., category event, response team plan, etc.
3.3.23	Functional	Presentational	Current Conditions	The application must show the number of planned works per geographic area as a range of upper, lower and mean figures.
3.3.24	Functional	Presentational	Current Conditions	The application must list/map the number of customer facing works per geographic area as a range of upper, lower and mean figures.
3.3.25	Functional	Presentational	Ongoing Learning	The User needs to view historical data.
3.3.26	Functional	Presentational	Ongoing Learning	The User needs to view usage data to understand trends.
3.3.27	Functional	Presentational	Ongoing Learning	The application must show a summary/list of past events.
3.3.28	Functional	Presentational	Ongoing Learning	The application must allow sorting/ filtering of past events by region.
3.3.29	Functional	Presentational	Current Conditions	The application must show the number of high voltage faults in rural areas.
3.3.30	Functional	Presentational	Ongoing Learning	The application must allow sorting/ filtering of past events by type.
3.3.31	Functional	Presentational	Ongoing Learning	The application must allow sorting/ filtering of past events by date.
3.3.32	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by weather event, on both the macro and micro map view.
3.3.33	Functional	Presentational	Ongoing Learning	The application must allow sorting/ filtering of past events by Engineer.
3.3.34	Functional	Presentational	Current Conditions	The application must list/map the number of active response teams per geographic area as a range of upper, lower and mean figures.
3.3.35	Functional	Presentational	Predicting Conditions	The application must show whether seasonal checks have been completed per region.
3.3.36	Functional	Presentational	Predicting Conditions	The User needs to view assessment/actions relating to an active weather fault. (i.e., Category event, probability score, response team plan).
3.3.37	Functional	Presentational	Current Conditions	The application must include an Alarm Page/ Summary View showing the dominating weather event.
3.3.38	Functional	Presentational	Predicting Conditions	The User needs to view a list of response plans based on category event.
3.3.39	Functional	Presentational	Predicting Conditions	The application must allow users to adjust time (i.e., day to storm) and type of weather (i.e.) to view how forecasted number of network faults varies.
3.3.40	Functional	Presentational	Predicting Conditions	The application must show the voltage number of predicted faults, e.g., HV/LV.
3.3.41	Functional	Presentational	Predicting Conditions	The application must show fault information on Customer Minutes Lost (CML).
3.3.42	Functional	Presentational	Predicting Conditions	The application must show the number of faults predicted within a window of 6 hours.
3.3.43	Functional	Presentational	Current Conditions	The application must show a total figure for predicted faults.

3.3.44	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by fault category, on both the macro and micro map view.
3.3.45	Functional	Presentational	Current Conditions	The application must show a summary of the weather forecast.
3.3.46	Functional	Presentational	Administration	The application map/ district view must be reconfigurable for different DNO networks.
3.3.47	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by fault range, on both the macro and micro map view.
3.3.48	Functional	Presentational	Current Conditions	The application must allow sorting/ filtering by prediction confidence, on both the macro and micro map view.
3.3.49	Functional	Presentational	Current Conditions	The application must show the number of fault predictions as a range of upper, lower and mean figures.
3.4.1	Functional	Procedural	Current Conditions	The application must generate a management report that can be distributed within the business.
3.4.2	Functional	Procedural	Current Conditions	The Application must send a daily automated RAG alert to the Control Room Manager when BAU.
3.4.3	Functional	Procedural	Current Conditions	The system must learn from information that is captured through data entry on the number of faults that actually occurred as a result of a weather event.
3.4.4	Functional	Procedural	Predicting Conditions	The application must send an automated alert/notification to a Senior Manager if a severe weather event is predicted in five (5) days time.
3.4.5	Functional	Procedural	Predicting Conditions	The application must send an automated alert/notification to the District Manager if a severe weather event is predicted in three (3) days time.
3.4.6	Functional	Procedural	Predicting Conditions	The Application must send an automated RAG email alert to the Control Room Manager when it predicts a specified threshold of faults will be exceeded.
3.5.1	Functional	Retrieval	Ongoing Learning	The system must store data such that it is accessible to Data Scientists for future use.
3.5.2	Functional	Retrieval	Ongoing Learning	The User must be able to access training data from the data store.
3.5.3	Functional	Retrieval	Predicting Conditions	The Application must show category event response plan that provides common scenarios based on the weather category event characteristics (e.g., time of year, time of week, etc).
4.1.1	NonFunctional	Archiving & Retention	Administration	Weather forecast data provided by DNOs must be stored for 5-years.
4.1.2	NonFunctional	Archiving & Retention	Administration	Data in the Data Store must be archived every 5 years.
4.3.1	NonFunctional	Back-Up / Recovery	Administration	Data in the Data Store must be backed up every 5 years.
4.4.1	NonFunctional	Capacity	Current Conditions	Multiple Users must be able to access the application at the same time, including: Control Room Manager, Control Engineers, Network Improvement Manager, Operational Planning Manager, District General Manager, SPD/SPM Heads of Delivery, EAC Management.
4.4.2	NonFunctional	Capacity	Ongoing Learning	The data store must have capacity to store satellite data for vegetation.
4.4.3	NonFunctional	Capacity	Predicting Conditions	The Weather Fault System must be capable of providing fault information services to the 6 energy Distribution Network Operators (DNOs) in the UK.
4.6.1	NonFunctional	Security	Not Applicable	Data transmitted between the front-end and back-end of the system must be encrypted.
4.6.2	NonFunctional	Security	Not Applicable	DNO Data must be securely segmented within the Data Store.