



---

# CAN REST API FOR PREDICTION DELIVERY

---

Cognitive Assistant for Networks (CAN) Release 5.5



JULY 20, 2021

AVANSEUS TECHNOLOGY PVT. LTD.

## REVISION HISTORY

| Version | Date       | Change description | Created by | Updated by    | Reviewed by |
|---------|------------|--------------------|------------|---------------|-------------|
| V 1.0   | July, 2021 | Initial Release    | Sunil      | Sandeep Singh | Chiranjib   |

## Table of Contents

|                                    |          |
|------------------------------------|----------|
| <b>1. Background .....</b>         | <b>3</b> |
| <b>2. API Specification .....</b>  | <b>3</b> |
| 2.1. getCANPredictions .....       | 3        |
| 2.2. getCANSamplePredictions.....  | 6        |
| <b>3. Output JSON Format .....</b> | <b>8</b> |

## 1. Background

This document addresses the technical details for consuming the predicted faults using REST API. The REST API provides the predictions that can be used by the registered third party applications.

## 2. API Specification

### 2.1. getCANPredictions

This API call will provide all the predictions generated for the specified date. Please note that the output structure of the json is exactly matching with the columns in the prediction report as per the agreement with the customer. Whenever the customer requests for a change, the same will be intimated 7 days in advance to all REST API consumers about the change in structure of the json output.

Swagger API specification:

```
{
  swagger: "2.0",
  info: {
    description: "Avanseus CAN REST API provides the CAN predictions output in the JSON format",
    version: "0.0.1-SNAPSHOT",
    title: "PredictedFaults REST API",
    contact: {
      name: "Avanseus Technologies Pvt. Ltd.",
      url: "https://www.avanseus.com",
      email: "support@avanseus.com"
    },
    license: {
      name: "Avanseus Software License Agreement",
      url: "https://www.avanseus.com/legal/"
    }
  },
  host: "avanseuscanapp.com",
  basePath: "/CAN",
  securityDefinitions: {
    basicAuth: {
      type: "basic"
    }
  },
  tags: [
    {
      name: "PredictedFaultsController",
      description: "REST API to fetch CAN predictions"
    }
  ],
}
```

```

paths: {
  "/CanPredictedFaultsProvider/getCanPredictions": {
    post: {
      tags: [
        "PredictedFaultsController"
      ],
      summary: "API to get CAN predictions",
      operationId: "getCanPredictionsUsingPOST",
      consumes: [
        "Authorization",
        "application/json"
      ],
      produces: [
        "application/json"
      ],
      parameters: [
        {
          in: "body",
          name: "requestParameters",
          description: "requestParameters",
          required: true,
          schema: {
            "$ref": "#/definitions/RequestParameters"
          }
        }
      ],
      responses: {
        200: {
          description: "Successful request",
          schema: {
            type: "object"
          }
        },
        400: {
          description: "Parameter value missing"
        },
        401: {
          description: "Authentication Failed! Please contact support@avanseus.com with the error code and time of API call."
        },
        403: {
          description: "Maximum number of service requests exceeded for the day!"
        }
      }
    }
  }
}

```

```

    30: {
      description: "Incorrect Date Format!. Please provide the date in the following format:
ddMMyyyy. Example: 21022019 for 21st February 2019"
    },
    31: {
      description: "Predictions not available for the given date ddMMyyyy"
    },
    32: {
      description: "Please contact support@avanseus.com with the error code and time of API
call."
    }
  }
}
},
definitions: {
  RequestParameters: {
    type: "object",
    properties: {
      applicationName: {
        type: "string"
      },
      predictionDate: {
        type: "string"
      }
    }
  }
}
}
}
}

```

Request

URL: <https://<domain>/CAN/CanPredictedFaultsProvider/getCanPredictions>

Method: POST

| Parameter       | Type                            | Requirement | Example                        |
|-----------------|---------------------------------|-------------|--------------------------------|
| Authorization   | Basic Auth HTTP header          | Mandatory   | Basic dGVzdHVzZXI6YWlzMzlxNVYw |
| applicationName | String                          | Mandatory   | CAN                            |
| predictionDate  | String(Date in ddMMyyyy format) | Mandatory   | 21062021                       |

## 2.2. getCANSamplePredictions

This API call will provide random sample predictions for the specified date. Please note that the output structure of the json is exactly matching with the columns in the prediction report as per the agreement with the customer. Whenever the customer requests for a change, the same will be intimated 7 days in advance to all REST API consumers about the change in structure of the json output.

Swagger API specification

```
{
  swagger: "2.0",
  info: {
    description: "Avanseus CAN REST API provides the CAN predictions output in the JSON format",
    version: "0.0.1-SNAPSHOT",
    title: "PredictedFaults REST API",
    contact: {
      name: "Avanseus Technologies Pvt. Ltd.",
      url: "https://www.avanseus.com",
      email: "support@avanseus.com"
    },
    license: {
      name: "Avanseus Software License Agreement",
      url: "https://www.avanseus.com/legal/"
    }
  },
  host: "avanseuscanapp.com",
  basePath: "/CAN",
  securityDefinitions: {
    basicAuth: {
      type: "basic"
    }
  },
  tags: [
    {
      name: "PredictedFaultsController",
      description: "REST API to fetch CAN predictions"
    }
  ],
  paths: {
    "/CanPredictedFaultsProvider/getCanSamplePredictions": {
      post: {
        tags: [
          "PredictedFaultsController"
        ],

```

```

summary: "API to get CAN predictions",
operationId: "getCanPredictionsUsingPOST",
consumes: [
  "Authorization",
  "application/json"
],
produces: [
  "application/json"
],
parameters: [
  {
    in: "body",
    name: "requestParameters",
    description: "requestParameters",
    required: true,
    schema: {
      "$ref": "#/definitions/RequestParameters"
    }
  }
],
responses: {
  200: {
    description: "Successful request",
    schema: {
      type: "object"
    }
  },
  400: {
    description: "Parameter value missing"
  },
  401: {
    description: "Authentication Failed! Please contact support@avanseus.com with the error code and time of API call."
  },
  403: {
    description: "Maximum number of service requests exceeded for the day!"
  },
  30: {
    description: "Incorrect Date Format!. Please provide the date in the following format: ddMMyyyy. Example: 21022019 for 21st February 2019"
  },
  31: {
    description: "Predictions not available for the given date ddMMyyyy"
  }
}

```

```

    },
    32: {
      description: "Please contact support@avanseus.com with the error code and time of API
call."
    }
  }
}
},
definitions: {
  RequestParameters: {
    type: "object",
    properties: {
      applicationName: {
        type: "string"
      },
      predictionDate: {
        type: "string"
      }
    }
  }
}
}
}
}

```

Request

URL: <https://<domain>/CAN/CanPredictedFaultsProvider/getCanSamplePredictions>

Method: POST

| Parameter       | Type                            | Requirement | Example                        |
|-----------------|---------------------------------|-------------|--------------------------------|
| Authorization   | Basic Auth HTTP header          | Mandatory   | Basic dGVzdHVzZXI6YWlxMzIxNVYw |
| applicationName | String                          | Mandatory   | CAN                            |
| predictionDate  | String(Date in ddMMyyyy format) | Mandatory   | 21062021                       |

### 3. Output JSON Format

```

{
  "statusCode": "200",
  "statusMessage": "SUCCESS",
  "predictedFaults": [
    {
      "PAT NUMBER": "7052",
      "CLUSTER": "46",

```

"SITE": "YADUBEDAZY",  
 "EQUIPMENT COMPONENT": "YADUBEDAZY-IO0026-G",  
 "CAUSE": "25888(SCTP Link Fault)",  
 "PROBABILITY": 97.06,  
 "EQUIPMENT TYPE": "Access Node",  
 "PRIORITY": "HIGH",  
 "AVG ALARM DURATION (MINS)": 5,  
 "ALARM OCCURRENCE COUNT": "27",  
 "ALARM DEDUPLICATED COUNT": "13",  
 "FAULT HISTORY": "Cause: 65085(Mains Failure Alarm), Event time: 28-04-2019 18:08:30, Cleared time: 28-04-2019 18:08:45  
 Cause: 13418(DROPRATIO\_OVER), Event time: 28-04-2019 09:13:11, Cleared time: 28-04-2019 09:13:27  
 Cause: 13418(DROPRATIO\_OVER), Event time: 27-04-2019 10:42:35, Cleared time: 27-04-2019 10:42:51  
 Cause: 65093(Generator On Load Alarm), Event time: 26-04-2019 12:03:25, Cleared time: 26-04-2019 12:07:19  
 Cause: 65085(Mains Failure Alarm), Event time: 26-04-2019 12:03:19, Cleared time: 26-04-2019 12:03:46  
 Cause: 13418(DROPRATIO\_OVER), Event time: 26-04-2019 05:17:49, Cleared time: 26-04-2019 05:18:05  
 Cause: 25888(SCTP Link Fault), Event time: 25-04-2019 19:05:41, Cleared time: 25-04-2019 19:06:22  
 Cause: 13418(DROPRATIO\_OVER), Event time: 25-04-2019 17:37:31, Cleared time: 25-04-2019 17:37:47  
 Cause: 13418(DROPRATIO\_OVER), Event time: 24-04-2019 08:32:57, Cleared time: 24-04-2019 08:33:13  
 Cause: 25888(SCTP Link Fault), Event time: 23-04-2019 23:42:10, Cleared time: 23-04-2019 23:42:51  
 Cause: 13418(DROPRATIO\_OVER), Event time: 23-04-2019 12:48:01, Cleared time: 23-04-2019 12:48:17  
 Cause: 65093(Generator On Load Alarm), Event time: 23-04-2019 07:34:47, Cleared time: 23-04-2019 07:37:32  
 Cause: 65085(Mains Failure Alarm), Event time: 23-04-2019 07:34:41, Cleared time: 23-04-2019 07:35:11  
 Cause: 13418(DROPRATIO\_OVER), Event time: 22-04-2019 07:02:47, Cleared time: 22-04-2019 07:03:02  
 Cause: 65093(Generator On Load Alarm), Event time: 22-04-2019 03:53:27, Cleared time: 22-04-2019 03:58:57  
 Cause: 65085(Mains Failure Alarm), Event time: 22-04-2019 03:53:21, Cleared time: 22-04-2019 03:53:48  
 Cause: 25888(SCTP Link Fault), Event time: 21-04-2019 12:36:27, Cleared time: 21-04-2019 12:37:05  
 Cause: 13418(DROPRATIO\_OVER), Event time: 21-04-2019 06:22:31, Cleared time: 21-04-2019 06:22:47  
 Cause: 13418(DROPRATIO\_OVER), Event time: 20-04-2019 11:42:06, Cleared time: 20-04-2019 11:44:59

```

Cause: 13418(DROPRATIO_OVER), Event time: 19-04-2019 17:44:26, Cleared time: 19-04-2019
17:44:42
Cause: 65085(Mains Failure Alarm), Event time: 19-04-2019 05:25:53, Cleared time: 19-04-2019
05:26:05
Cause: 65093(Generator On Load Alarm), Event time: 18-04-2019 09:40:15, Cleared time: 18-04-
2019 09:59:00
Cause: 65085(Mains Failure Alarm), Event time: 18-04-2019 09:40:09, Cleared time: 18-04-2019
09:40:36
Cause: 65085(Mains Failure Alarm), Event time: 17-04-2019 12:51:40, Cleared time: 17-04-2019
12:51:49
Cause: 13418(DROPRATIO_OVER), Event time: 17-04-2019 12:50:33, Cleared time: 17-04-2019
12:50:48
Cause: 13050(AM_DOWNSHIFT), Event time: 17-04-2019 10:43:40, Cleared time: 17-04-2019
10:44:06
Cause: 25888(SCTP Link Fault), Event time: 17-04-2019 10:21:53, Cleared time: 17-04-2019
10:27:09
Cause: 13418(DROPRATIO_OVER), Event time: 16-04-2019 13:24:06, Cleared time: 16-04-2019
13:24:53
Cause: 25888(SCTP Link Fault), Event time: 15-04-2019 19:22:21, Cleared time: 15-04-2019
19:23:00
Cause: 13418(DROPRATIO_OVER), Event time: 15-04-2019 15:48:30, Cleared time: 15-04-2019
15:48:46
Cause: 65085(Mains Failure Alarm), Event time: 15-04-2019 10:26:23, Cleared time: 15-04-2019
10:26:32
Cause: 13418(DROPRATIO_OVER), Event time: 14-04-2019 09:46:11, Cleared time: 14-04-2019
09:46:26
Cause: 25888(SCTP Link Fault), Event time: 13-04-2019 23:38:59, Cleared time: 13-04-2019
23:39:41
Cause: 13418(DROPRATIO_OVER), Event time: 13-04-2019 08:48:15, Cleared time: 13-04-2019
08:48:31
",
    "PREDICTIVE TICKET HISTORY": "",
    "TICKET HISTORY": "",
    "LATEST TICKET": "",
    "RECOMMENDED RCA": "Power Supply Problem",
    "FAULT TRACE 1": "65093(Generator On Load Alarm)",
    "FAULT TRACE 2": ""
  }
]
}

```