Feature Requirements Report

Feature:

{

  "title": "Test Feature",

  "description": "The aim of this feature is to add a new two-way communication between ADT and BLMS server for\nseveral applications. One OTX to be able to get data from BLMS through a key. One OTX to be able to push data in\nthe BLMS through a key.\n\nThe user of these OTX libraries is a Renault Diagnostic User.\n\nI believe we should\nhave two epics for these features: \n1. OTX library to write to BLMS server\n2. OTX library to read from BLMS\nserver\n\nI believe we should have some user stories like this:\n-> As a Renault Diagnostic User, I want to call an\nETAS OTX library so that I can write data to BLMS server.\n-> As a Renault Diagnostic User, I want to call an ETAS\nOTX library so that I can read data from BLMS server.\n\nSome technical explanation about each library:\n1. OTX\nlibrary to write to BLMS server\n-> An OTX calls the OTX library request to write data to BLMS server. It sends the\nOTX library data to be written in the form of a key value pair MAP\n-> the OTX library calls a host service provider,\nwhich is a service in ADT runtime. The OTX library pass the data to be written. The ADT runtime reads the BIN\n(Battery Identification Number). If the BIN is missing, the host service provider returns an error Message and the\nOTX library returns an error boolean and an error message.\n-> If the BIN could be read, the ADT runtime calls a\nrest endpoint from ADT server which is PUT updateByBin/{bin} and the server performs the write to the BLMS\nserver. This endpoint already exist and there is no new development to be done on server side.\n-> the server\nsends the status to the ADT runtime and ADT runtime host service provider sends OTX the status and OTX returns\nerror false and a positive message.\n-> when the server writes in BLMS, a feedback ADT.SERVICE.BLM is sent to\nfeedback viewer\n\n2. OTX Read BLMS\n-> An OTX calls the OTX library with a list of keys. These are values that\nthe OTX wants to read from BLMS\n-> The OTX library through a host service provider sends the list of keys to the\nADT runtime.\n-> The ADT runtime reads the BIN. If BIN could not be read, an error message is returned to the\nOTX library and the process stops. The OTX library returns an error message to the OTX calling it\n-> if BIN could\nbe read, the ADT runtime sends to the ADT server the keys and BIN through a new rest endpoint that needs to be\ndevelopped\n-> In ADT server side, it needs to perform an http request with a new API. This API is unknown for\nnow. This is new development to be made\n-> ADT server then sends a

feedback ADT.SERVICE.BLMS of data that\nhas been read from BLMS\n-> The server then sends the data and status to ADT runtime\n-> ADT runtime then\nsends it to OTX library which in turn returns it to the OTX calling it\n\"",

"context": "eral OTX scenarios are needed to communicate with the BLMS server. We need to be able to read\nand write data through ‘keys’. The goal is the same as the BVM Read / Write concept.\nThe writing of data from\nADT to BLMS through this scenario should be instantly done to let NewDialogys read the data just after the\nscenario.\nExample for more context:\nFor electric vehicles, Renault rule states that in the event of an accident with\nairbag deployment, the traction battery must be replaced and sent for recycling. \n\nThis leads to a significant\nincrease in the repair cost of damaged vehicles, as in addition to bodywork expenses, the additional cost of\nreplacing the traction battery must also be covered. \n\nIn many cases, this situation leads to: \n\n- High risk of\neconomically non-repairable vehicle to be scrapped (the repair cost exceeds the car's residual value). From the\nperspective of insurers and specialized media, this is a technical and ecological nonsense. \n\n- Insurance\ncompanies compensating for vehicles that are lightly/moderately damaged, even when they are less than 2 years\nold in some cases.\n\nWith the introduction of EDR (GSR2 regulation) in July 2024, it becomes possible to\nimplement a traction battery diagnostic based on Airbag ECU data. \n\nADT reads the concerned GSR2 DIDs and\nprovide a calculated result to state if the battery is reusable or not. These DIDs and the results are sent to BLMS for\ntraceability: unlike the traction battery State of Health (SoH) that is sent automatically to BLMS via ADT, these\nGSR2 DIDs are GDPR compliant, they shall not be sent automatically to BLMS when we connect ADT to the\nvehicle as we must have the client agreement to read them and send them to BLMS.\n\"\n}\nSummary:\nThe Two-Way Communication with BLMS Server feature enables Renault Diagnostic Users to seamlessly write and\nread data to/from the Battery Lifecycle Management System (BLMS) server through ETAS OTX libraries. This\nfeature consists of two primary epics: OTX library for writing data to BLMS server, utilizing an existing endpoint, and\nOTX library for reading data from BLMS server, requiring development of a new REST endpoint and API integration.\nThe feature provides instant updates for downstream applications, enhances diagnostic capabilities, and offers\nrobust error handling and real-time operational insights through feedback mechanisms. This enhancement is crucial\nfor efficient vehicle diagnostics, particularly for electric vehicles, where it supports the implementation of traction\nbattery diagnostics based on Airbag ECU data, potentially reducing repair

costs and environmental impact.\nBenefit Hypothesis:\nWe believe that by implementing the Two-Way Communication with BLMS Server feature, Renault Diagnostic Users\nwill experience a significant reduction in diagnostic time and an increase in accuracy, leading to improved overall\nefficiency and customer satisfaction. Specifically, the instant updates to BLMS will ensure that downstream\napplications, like NewDialogys, have access to the latest data, enhancing the repair process. Furthermore, the\nfeature is expected to contribute to reduced repair costs for electric vehicles by facilitating informed decisions on\ntraction battery replacement, ultimately benefiting both Renault and its customers environmentally and economically.\nKey indicators of success will include metrics on reduced diagnostic time, increased first-time fix rates, and a\ndecrease in economically non-repairable vehicle cases due to battery replacement costs."

}

Summary:

undefined

Benefit Hypothesis:

undefined

Content:

{
  "userStories": [
    {
      "id": "US-1",
      "story": "As a Renault Diagnostic User, I want to call an ETAS OTX library to write data to the BLMS server\n through a key-value pair MAP, so that I can instantly update BLMS for downstream applications like NewDialogys to\n   ensure seamless vehicle diagnostics",
      "role": "Renault Diagnostic User",
      "feature": "OTX library to write to BLMS server",
      "benefit": "Enables instant updates for downstream applications"
    },
    {

```
      "id": "US-2",
      "story": "As a Renault Diagnostic User, I want to call an ETAS OTX library to read data from the BLMS server\n
    using a list of keys, so that I can retrieve specific diagnostic data to inform repair decisions",
      "role": "Renault Diagnostic User",
      "feature": "OTX library to read from BLMS server",
      "benefit": "Enhances diagnostic capabilities with targeted data retrieval"
    },
    {
      "id": "US-3",
      "story": "As a Renault Diagnostic User, I want the OTX library to handle errors when writing to the BLMS
server\n       (e.g., missing BIN), so that I receive informative error messages to quickly resolve issues",
      "role": "Renault Diagnostic User",
      "feature": "Error handling for OTX library (write)",
      "benefit": "Improves user experience with robust error handling"
    },
    {
      "id": "US-4",
      "story": "As a Renault Diagnostic User, I want the OTX library to handle errors when reading from the BLMS\n
  server (e.g., missing BIN, API issues), so that I can efficiently troubleshoot and resolve connection problems",
      "role": "Renault Diagnostic User",
      "feature": "Error handling for OTX library (read)",
      "benefit": "Enhances overall system reliability and user productivity"
    },
    {
      "id": "US-5",
      "story": "As a System Administrator, I want the ADT server to send feedback (ADT.SERVICE.BLM) upon\n
successful write operations to BLMS, so that I can monitor and verify the integrity of write operations in real-time",
      "role": "System Administrator",
      "feature": "Feedback mechanism for write operations",
      "benefit": "Provides real-time operational insights for improved system management"
    },
    {
      "id": "US-6",
      "story": "As a System Administrator, I want the ADT server to send feedback (ADT.SERVICE.BLMS) upon\n
successful read operations from BLMS, so that I can track and analyze data retrieval patterns for optimization",
      "role": "System Administrator",
      "feature": "Feedback mechanism for read operations",
      "benefit": "Enables data-driven decision-making for system improvements"
    }
  ],
  "acceptanceCriteria": {
    "US-1": [
      "The OTX library successfully sends a key-value pair MAP to the ADT runtime.",
      "The ADT runtime writes the data to BLMS using the existing PUT updateByBin/{bin} endpoint.",
      "The OTX library receives a success status and a positive message from the ADT runtime."
    ],
    "US-2": [
      "The OTX library successfully sends a list of keys to the ADT runtime.",
      "The ADT runtime reads the BIN and sends the keys and BIN to the ADT server via the new REST endpoint.",
      "The ADT server performs an HTTP request to the new API to read data from BLMS.",
      "The OTX library receives the read data and status from the ADT runtime."
    ],
    "US-3": [
      "The OTX library returns an error boolean and an informative error message when the BIN is missing.",
      "The OTX library handles other potential errors (e.g., network, server) and provides meaningful error
messages."
    ],
    "US-4": [
      "The OTX library returns an error boolean and an informative error message when the BIN is missing or API\n
  issues occur.",
      "The OTX library handles other potential errors (e.g., network, server) and provides meaningful error\n
messages."
    ],
    "US-5": [
      "The ADT server sends a feedback (ADT.SERVICE.BLM) to the feedback viewer upon successful write\n
operations.",
      "The feedback contains relevant information about the write operation (e.g., BIN, timestamp)."
    ],
    "US-6": [
      "The ADT server sends a feedback (ADT.SERVICE.BLMS) to the feedback viewer upon successful read\n
operations.",
      "The feedback contains relevant information about the read operation (e.g., BIN, keys, timestamp)."
    ]
  },
  "estimates": {
    "US-1": "M",
    "US-2": "L",
    "US-3": "XS",
    "US-4": "S",
    "US-5": "XS",
    "US-6": "XS"
  },
  "questions": [
    "What is the expected payload size for the key-value pair MAP in write operations, and are there any specific\n
  formatting requirements?",
    "Can you provide more information about the new API for reading data from BLMS (e.g., API endpoint,\n
authentication, rate limiting)?",
    "How will the Renault Diagnostic User authenticate and authorize with the OTX library and ADT server?",
    "Are there any specific logging or auditing requirements for the write and read operations?",
    "What is the expected frequency and volume of write and read operations, and are there any performance or\n
 scalability concerns?",
    "Can you clarify the format and content of the feedback (ADT.SERVICE.BLM and ADT.SERVICE.BLMS) and
how\n        it will be consumed by the feedback viewer?",
    "Are there any dependencies or prerequisites for the development of these user stories (e.g., other teams'
work,\n        infrastructure setup)?",
    "How will the success of these user stories be measured, and what are the key performance indicators (KPIs)?"
  ]
}
```