

Introduction .....	1
Save manifest.....	2
Update manifest .....	10
Delete Manifest Service (post June release).....	20
Manifest entities and fields validation for Save service .....	22
Manifest entities and fields validation for Update service .....	27
Facility Information Validation for “Image” submission type.....	31
Designated Facility (TSDF)/Generator Site Id and Site Information Validation .....	34
Generator Site Id and Site Information Validation .....	39
Transporter Site ID and Site Information Validation .....	45
Paper Signature Info Validation .....	48
Import Information Validation.....	56
New Manifest Information Validation .....	58
Rejection Information Validation.....	59
Discrepancy fields validation .....	63
Residue Fields Validation .....	64
Manifest Additional Information Validation .....	66
Waste Validation.....	68
Management Method Code Validation .....	70
DOT Information fields validation.....	71
Containers and quantity fields validation .....	72
Hazardous Waste Validation.....	73
Biennial Report Information Validation.....	76
PCB Information Validation.....	78
Waste Additional Information Validation .....	81
Attachment Validation.....	82

## Introduction

### *Description*

The following document describes the use of and validations performed by the e-Manifest services. The document focuses on the Save, Update, and Delete (post June release) services. Save is the initial create of a manifest in e-Manifest. The Save service may be used for a fully electronic manifest or to upload a paper manifest and its data. The Update service is used for

subsequent changes to a previously submitted manifest and is based on the Manifests Tracking Number provided. Delete service is used to delete the manifest (post June release).

A user implementing these services must also submit an authentication token with each service call. The authentication token is obtained using the Authentication service which is covered in [\(e-Manifest Authenticate Get and Lookup Services 03 21.pdf 03 21.pdf\)](#) document.

In order to use any service, the user must have the site manager role for at least one of the handlers listed on the manifest and have requested an API ID and API Key through the e-Manifest module of the RCRAInfo Industry Application.

## Save manifest

### Description

The Save Service creates the new Manifest. The service accepts Manifest data in JSON format compliant with e-Manifest JSON schema ([e-Manifest JSON Schema](#)). The following submission types will be supported for this release:

- "FullElectronic"
- "DataImage5Copy"
- "Image"
- "Hybrid"

For the "FullElectronic" and "Hybrid" submission types, the Manifest can be submitted at either "Pending" (post June release) or "Scheduled" status. The transition to the statuses after "Scheduled" for these submission types will be done automatically by E-Manifest.

For the "DataImage5Copy" and "Image" submission types, the Manifest will be assigned the "ReadyForSignature" status. The transition to the next status for this submission type will be done automatically by e-Manifest.

For the "DataImage5Copy" and "Image" submission types the service requires receiving the scanned compressed document attachment (Printed/Paper, Signed, Scanned Manifest form-2050). Manifest attachment shall be passed as a multipart binary content. (See sample client implementation for details at: <https://github.com/USEPA/e-manifest/tree/master/Services-Information/sample-client>)

For the "DataImage5Copy" and "Image" submission types if the Manifest contains an attachment the following metadata JSON elements shall be presented in the Manifest JSON:

```
"printedDocument": {  
  "name": "user provided document name.pdf",  
  "size": 23455, (size in bytes of the pdf document)  
  "mimeType": "APPLICATION_PDF"  
}
```

Service will validate submitted JSON (see details in Manifest entities and fields validation for Save service section) and if:

- No Errors or Warnings were found: Service will save the manifest and return generated Manifest Tracking Number
- Only Warnings were found: Service will save the manifest and return generated Manifest Tracking Number and Warning Report containing all found Warnings

- Error(s) were found: Service won't save the manifest and return Error Report containing all found errors.
- Error(s) and Warnings were found. Service won't save the manifest and return Error and Warning Report containing all found errors and warnings.

### *Service Type*

e-Manifest Data Service

### *Parameters*

- Security Token: Security Token received in Authentication. Will be passed in the Header of the request.
- Manifest JSON
- Attachment Document: If applicable, manifest attachment shall be passed as a multipart binary content.

### *Service Invocation Example*

PreProduction/ Test Environment: <Server

URL>/rcrainfo/rest/api/v1/emanifest/manifest/save/{manifest JSON}

Production: <Server URL>/rcrainfoprod/rest/api/v1/emanifest/manifest/save/{manifest JSON}

### *Request JSON Schema*

[emanifest.json](#)

### *Manifest JSON examples*

Valid Manifest JSON Example for "FullElectronic" submission type

```
{
  "submissionType" : "FullElectronic",
  "status" : "Scheduled",
  "import" : false,
  "rejection" : false,
  "generator" : {
    "epaSiteld" : "MDD981111081",
    "contact" : {
      "phone" : {
        "number" : "301-294-9727"
      }
    }
  },
  "emergencyPhone" : {
    "number" : "301-423-5455"
  },
  "transporters" : [{
    "epaSiteld" : "CAR000189282",
    "name" : "transporter 1",
    "order" : 1
  }],
  {
    "epaSiteld" : "CAD982000564",
```

```

    "name": "transporter 2",
    "order" : 2
  },
  "designatedFacility" : {
    "epaSiteId" : "AK8570028649"
  },
  "wastes" : [ {
    "dotHazardous" : true,
    "epaWaste" : true,
    "pcb" : false,
    "dotInformation" : {
      "idNumber" : {
        "code" : "UN1988"
      },
      "printedDotInformation" : "UN1988, Waste Aldehydes, flammable, toxic, n.o.s., (technical name), 3,
I, RQ(rq description)"
    },
    "quantity" : {
      "containerNumber" : 100,
      "containerType" : {
        "code" : "BA"
      },
      "quantity" : 2,
      "unitOfMeasurement" : {
        "code" : "L"
      }
    },
    "hazardousWaste" : {
      "federalWasteCodes" : [ {
        "code" : "D023"
      }, {
        "code" : "D001"
      }, {
        "code" : "D021"
      } ],
      "tsdfStateWasteCodes" : [ ],
      "generatorStateWasteCodes" : [ ]
    },
    "additionalInfo" : {
      "comments" : [ ]
    },
    "lineNumber" : 1,
    "br" : true,
    "brInfo" : {
      "densityUnitOfMeasurement" : {
        "code" : "1"
      },
      "formCode" : {
        "code" : "W107"
      },
      "sourceCode" : {
        "code" : "G07"
      },
      "wasteMinimizationCode" : {
        "code" : "X"
      }
    }
  }

```

```

    "density" : 1.0
  },
  "managementMethod" : {
    "code" : "H020"
  }
},
"additionalInfo" : {
  "comments" : []
}
}

```

### *Sequence of steps*

1. The System will validate Security Token
  - 1.1. If Web Security Token is invalid, the system stops the submission and will generate following error:
    - E\_SecurityApiTokenInvalid: Invalid Security Token
  - 1.2. If Web Security Token expired, the system stops the submission and will generate the following error:
    - E\_SecurityApiTokenExpired: Security Token is Expired
  - 1.3. If Account was inactivated after the token was issued, the system stops the submission and will generate the following error:
    - E\_SecurityApiInvalidStatus: This API Id is no longer active
2. The system will perform User Authorization
  - 2.1. The system will check if the User has Site Manager (Industry) permissions for any Site.
  - 2.2. If the User does not have permissions, the system will stop the processing and will generate the following error:
    - E\_IndustryPermissions: The user does not have industry permissions for any Site
  - 2.3. The system will check if the user has Site Manager Permissions for either the TSDF Site ID or Generator Site ID or Alternate TSDF site ID provided in the manifest. (This authorization is not yet turned on)
    - 2.3.1. If the user has Site Manager Permissions for the TSDF Site, then TSDF acts as "TSDF"
    - 2.3.2. If the user has Site Manager Permissions for the Generator Site, then TSDF acts as "Generator".
    - 2.3.3. If the user has Site Manager Permissions for the Alternate SDF Site, then TSDF acts as "Alternate TSDF".
  - 2.4. If the User does not have Site Manager Permissions for either the TSDF Site or for the Generator Site or Alternate TSDF Site, then the System will stop the processing and will generate following error:
    - E\_SitePermissions: The user does not have Industry Permissions for the either TSDF Site or Generator Site or Alternate TSDF Site specified in provided JSON
3. If the User was Authenticated and Authorized, the system will process the request
  - 3.1. The system will validate the provided Manifest JSON and Attachment Document
    - 3.1.1. The service will validate the e-Manifest according to the rules described in section Manifest entities and fields validation for Save service

- 3.2. If no errors or warnings were generated during validation process, the service will perform the following steps:
  - 3.2.1. Generate Manifest Tracking Number for the provided Manifest data
  - 3.2.2. Store provided Manifest information and an Attachment Document into e-Manifest database
  - 3.2.3. Return Manifest Tracking Number to the Requestor.
- 3.3. If a warning is generated during the validation process, the service will perform the following steps:
  - 3.3.1. Generate Manifest Tracking Number for the provided Manifest data
  - 3.3.2. Store the provided Manifest information and an Attachment Document into e-Manifest database
  - 3.3.3. Return the Manifest Tracking Number and Warning(s) Report to the Requestor
- 3.4. If an error is generated during the validation process, the service will perform following steps:
  - 3.4.1. Generate Error Report with all errors found during validation process
  - 3.4.2. Return Error Report to the Requestor.
- 3.5. If any errors and warnings were generated during validation process, the service will perform following steps:
  - 3.5.1. Generate Error/Warning Report with all errors and warnings found during validation process
  - 3.5.2. Return Error/Warning Report to the Requestor.

### *Response JSON Schemas*

[save-emanifest-success.json](#) : Schema for successful submission response

[save-emanifest-failure.json](#) : Schema for failed submission response

### *Completed Response Examples*

Response Without Warnings Example

```
{  
  "manifestTrackingNumber": "100005103ELC"}  
}
```

Response With Warnings Example

```
{  
  "manifestTrackingNumber": "100005103ELC",  
  "warningsReport": {  
    "reportId": "012345",  
    "date": "2017-07-27T19:13:28.628+0000",  
    "manifestWarnings": [  
      {  
        "message": "Invalid Format. Text not exceeding 255 characters is expected",  
        "field": "Emanifest.additionalInfo.comments.label",  
        "value": "too many characters"  
      }  
    ],  
    "generatorReport": {
```

```

    "entityId": {
      "entityIdField": "siteId",
      "entityIdValue": "MDR12345125"
    },
    "warnings": [
      {
        "message": "Invalid Format. Numbers not exceeding 6 characters is expected",
        "field": "Emanifest.generator.emergencyPhone.extension",
        "value": "N789"
      }
    ]
  },
  "tsdfReport": {
    "entityId": {
      "entityIdField": "siteId",
      "entityIdValue": "MDR12345125"
    },
    "warnings": [
      {
        "message": "Invalid Format. Alphanumeric characters not exceeding 50 characters is expected",
        "field": "Emanifest.designatedFacility.siteAddress.address2",
        "value": ":-^"
      }
    ]
  },
  "wastesReports": [
    {
      "entityId": {
        "entityIdField": "lineNumber",
        "entityIdValue": "1"
      },
      "warnings": [
        {
          "message": "Invalid Format. Numeric characters is expected",
          "field": "Emanifest.wastes.additionalInfo.handlerId",
          "value": "FG056"
        }
      ]
    },
    {
      "entityId": {
        "entityIdField": "lineNumber",
        "entityIdValue": "2"
      },
      "warnings": [
        {
          "message": "Invalid Format. Text not exceeding 4000 is expected",
          "field": "Emanifest.wastes.additionalInfo.handlingInstructions",
          "value": "Way too long..."
        }
      ]
    }
  ]
}

```

### *Failed Response Example*

```
{
  "reportId": "012345",
  "date": "2017-07-27T19:13:28.628+0000",
  "manifestErrors": [
    {
      "message": "Invalid Field Format. FullElectronic or DataImage5Copy is expected",
      "field": "Emanifest.submissionType",
      "value": "DataImage"
    },
    {
      "message": "Mandatory Field is not provided",
      "field": "Emanifest.designatedFacility"
    },
    {
      "message": "No Transporters specified. At least one Transporter is required for
Scheduled Status",
      "field": "Emanifest.transporter"
    }
  ],
  "manifestWarnings": [
    {
      "message": "Invalid Field Format. Text value not exceeding 255 characters is
expected",
      "field": "Emanifest.additionalInfo.comments.label",
      "value": "too many characters"
    }
  ],
  "generatorReport": {
    "entityId": {
      "entityIdField": "siteId",
      "entityIdValue": "MDD003473337"
    },
    "errors": [
      {
        "message": "Mandatory Field is not provided",
        "field": "Emanifest.generator.name"
      },
      {
        "message": "Lookup Value Not Found",
        "field": "Emanifest.generator.siteAddress.state.code",
        "value": "NA"
      },
      {
        "message": "Mandatory Field is not provided",
        "field": "Emanifest.generator.mailingAddress"
      }
    ],
    "warnings": [
      {
        "message": "Invalid Format",
        "field": "Emanifest.generator.emergencyPhone.extension",
        "value": "N789"
      }
    ]
  }
},
```



```

"tsdfReport": {
  "entityId": {
    "entityIdField": "siteId",
    "entityIdValue": "N/A"
  },
  "errors": [
    {
      "message": "Mandatory Field is not provided",
      "field": "Emanifest.designatedFacility.siteId"
    },
    {
      "message": "Invalid Format",
      "field": "Emanifest.designatedFacility.siteAddress.zip",
      "value": "X345"
    },
    {
      "message": "Mandatory Field is not provided",
      "field": "Emanifest.designatedFacility.emergencyPhone.number"
    }
  ],
  "warnings": [
    {
      "message": "Invalid Format",
      "field": "Emanifest.designatedFacility.siteAddress.address2",
      "value": ";^"
    }
  ]
},
"wastesReports": [
  {
    "entityId": {
      "entityIdField": "lineNumber",
      "entityIdValue": "1"
    },
    "errors": [
      {
        "message": "Lookup Value Not Found",
        "field": "Emanifest.wastes.dotInformation.idNumber",
        "value": "UN4345"
      },
      {
        "message": "Mandatory Field is not provided",
        "field": "Emanifest.wastes.quantity.quantity"
      },
      {
        "message": "Invalid Field Format.Integer number not exceeding 999 is
expected",
        "field": "Emanifest.wastes.quantity.containerNumber",
        "value": "n"
      }
    ],
    "warnings": [
      {
        "message": "Invalid Field Format.",
        "field": "Emanifest.wastes.additionalInfo.handlerId",
        "value": "FG056"
      }
    ]
  }
]

```

```

    }
  ],
  {
    "entityId": {
      "entityIdField": "lineNumber",
      "entityIdValue": "2"
    },
    "errors": [
      {
        "message": "Waste Description is not provided. Waste Description must be
provided if hazardous is false",
        "field": "Emanifest.wastes.wasteDescription"
      },
      {
        "message": "Density is not provided. Density must be provided if quantity UOM is
Gallons, Liters or Cubic meters",
        "field": "Emanifest.wastes.br.density"
      },
      {
        "message": "Lookup Value Not Found",
        "field": "Emanifest.wastes.managementMethod.code",
        "value": "X25"
      }
    ]
  }
]
}

```

#### Errors

Type	code	Description
N/A	E_SecurityApiTokenInvalid	Invalid Security Token
N/A	E_SecurityApiTokenExpired	Security Token is Expired
N/A	E_SecurityApiInvalidStatus	This API Id is no longer active
Authorization	E_IndustryPermissions	The user does not have Industry permissions for any Site
Validation	E_SitePermissions	The user does not have Industry Permissions for the either TSDF Site or Generator Site or Alternate TSDF site specified in provided JSON
System	E_SystemError	Message describing system error

## Update manifest

### Description

The Update service updates an existing Manifest. The service requires that the industry system submits a complete Manifest JSON in order to update a manifest. Individual Manifest fields cannot be updated via this service, only the entire manifest. The update service accepts Manifest data in JSON format compliant with the e-Manifest JSON schema ([e-Manifest JSON Schema](#)) and can also accept a Manifest attachment for the "DataImage5Copy" and "Image" submission types (Scanned Signed Paper Manifest Image pdf). Manifest JSON must contain the Manifest Tracking

Number generated for that manifest by the Save Manifest service. The Save manifest service must be invoked prior to the Update Manifest service for any manifest. The following submission types will be supported for this release:

- "FullElectronic"
- "DataImage5Copy"
- "Image"
- "Hybrid"

For "FullElectronic" and "Hybrid" submission types the Manifest can be updated with "Pending" (post June Release) or "Scheduled" statuses. For these submission types the transition to the next after "Scheduled" statuses will be done by e-Manifest.

For "DataImage5Copy" and "Image" submission types the Manifest status shall not be submitted. For this submission type the status transition will be done by e-Manifest.

For the "DataImage5Copy" and "Image" submission types the service supports receiving the scanned document attachment (Printed, Signed, Scanned Manifest form-2050). Manifest attachment shall be passed as a multipart binary content. (See sample client implementation for details). For the "DataImage5Copy" submission type if the Manifest contains an attachment the following metadata JSON elements shall be presented in the Manifest JSON:

```
"printedDocument": {  
  "name": "user provided document name.pdf",  
  "size": 23455, (size in bytes of the pdf document)  
  "mimeType": "APPLICATION_PDF"  
}
```

Service will validate submitted JSON (see section Manifest entities and fields validation for Update service) and if:

- No Errors or Warnings were found: The service will update the manifest and return the existing Manifest Tracking Number
- Only Warnings were found: The service will update the manifest and return the existing Manifest Tracking Number and Warning Report containing all found Warnings
- Error(s) were found: The service won't update the manifest and returns Error Report containing all found errors.
- Error(s) and Warnings were found. The service won't update the manifest and returns Error Report containing all found errors and warnings.

## Service Type

e-Manifest Data Service

## Parameters

- Security Token: Security Token received in Authentication. Will be passed in the Header of the request.
- Manifest JSON
- Attachment Document If applicable, manifest attachment shall be passed as a multipart binary content.

### *Service Invocation Example*

PreProduction/ Test Environment: <Server

URL>/rcrainfo/rest/api/v1/emanifest/manifest/update/{manifest JSON}

Production: <Server URL>/rcrainfoprod/rest/api/v1/emanifest/manifest/update/{manifest JSON}

### *Request JSON Schema*

[emanifest.json](#)

### *Manifest JSON example*

Valid Manifest JSON Example for "FullElectronic" submission type

```
{
  "manifestTrackingNumber": "100000781ELC",
  "submissionType": "FullElectronic",
  "status": "Scheduled",
  "import": false,
  "rejection": false,
  "containsPreviousRejectOrResidue": false,
  "generator": {
    "epaSiteld": "MDD981111081",
    "contact": {
      "phone": {
        "number": "301-294-9727"
      }
    }
  },
  "emergencyPhone": {
    "number": "301-423-5455"
  }
},
"transporters": [ {
  "epaSiteld": "CAR000189282",
  "name": "transporter 1",
  "order": 1
},
{
  "epaSiteld": "CAD982000564",
  "name": "transporter 2",
  "order": 2
} ],
"designatedFacility": {
  "epaSiteld": "AK8570028649"
},
"wastes": [ {
  "dotHazardous": true,
  "epaWaste": true,
  "pcb": false,
  "dotInformation": {
    "idNumber": {
      "code": "UN1988"
    }
  },
  "printedDotInformation": "UN1988, Waste Aldehydes, flammable, toxic, n.o.s., (technical name), 3, I, RQ(rq description)"
}
```

```

},
"quantity" : {
  "containerNumber" : 100,
  "containerType" : {
    "code" : "BA"
  },
  "quantity" : 2,
  "unitOfMeasurement" : {
    "code" : "L"
  }
},
"hazardousWaste" : {
  "federalWasteCodes" : [ {
    "code" : "D023"
  }, {
    "code" : "D001"
  }, {
    "code" : "D021"
  } ],
  "tsdfStateWasteCodes" : [ ],
  "generatorStateWasteCodes" : [ ]
},
"additionalInfo" : {
  "comments" : [ ]
},
"lineNumber" : 1,
"br" : true,
"brInfo" : {
  "densityUnitOfMeasurement" : {
    "code" : "1"
  },
  "formCode" : {
    "code" : "W107"
  },
  "sourceCode" : {
    "code" : "G07"
  },
  "wasteMinimizationCode" : {
    "code" : "X"
  },
  "density" : 1.0
},
"managementMethod" : {
  "code" : "H020"
}
},
"additionalInfo" : {
  "comments" : [ ]
}
}

```

### *Sequence of steps*

1. The System will validate Security Token

- 1.1. If the Web Security Token is invalid, the system stops the submission and generates the following error:
  - E\_SecurityApiTokenInvalid: Invalid Security Token
- 1.2. If the Web Security Token expired, the system stops the submission and generates the following error:
  - E\_SecurityApiTokenExpired: Security Token is Expired
- 1.3. If the Account was inactivated after the token was issued, the system stops the submission and generates the following error:
  - E\_SecurityApiInvalidStatus: This API Id is no longer active
2. The system will perform User Authorization
  - 2.1. The system will check if the User has Site Manager (Industry) permissions for any Site
  - 2.2. If the User does not have permissions, the system will stop processing the manifest and generate the following error:
    - E\_IndustryPermissions: The user does not have industry permissions for any Site
  - 2.3. The system will check if the user has the Industry Permissions for either the TSDF Site ID or Generator Site ID or Alternate TSDF Site provided on the manifest. (Authorization is not yet turned on)
  - 2.4. If the User does not have the appropriate permissions the system will stop the processing the manifest and generates the following error:
    - E\_SitePermissions: The user does not have Industry Permissions for the either TSDF Site or Generator Site or Alternate TSDF Site specified in provided JSON
3. If the User was Authenticated and Authorized, the system will process the request
  - 3.1. The system will validate the provided Manifest JSON and Attachment Document
    - 3.1.1. The system will validate the e-Manifest fields according to the rules described in section Manifest entities and fields validation for Update service
  - 3.2. If no errors or warnings are generated during the validation process, the service will perform the following steps:
    - 3.2.1. Update existing Manifest with provided Manifest information
    - 3.2.2. Create or Update Attachment Document into e-Manifest database
    - 3.2.3. Return Manifest Tracking Number to the Requestor.
  - 3.3. If only warnings are generated during the validation process, the service will perform following steps:
    - 3.3.1. Update existing Manifest with provided Manifest information
    - 3.3.2. Create or Update Attachment Document into e-Manifest database
    - 3.3.3. Return Manifest Tracking Number and Warning(s) Report to the Requestor
  - 3.4. If an error is generated during the validation process the system will perform the following steps:
    - 3.4.1. Generate Error Report with all errors found during validation process
    - 3.4.2. Return Error Report to the Requestor.
  - 3.5. If any errors and warnings were generated during validation process the system will perform following steps

- 3.5.1. Generate Error/Warning Report with all errors and warnings found during validation process
- 3.5.2. Return Error/Warning Report to the Requestor

### *Response JSON Schemas*

[save-manifest-success.json](#) : Schema for successful submission response

[save-manifest-failure.json](#) : Schema for failed submission response

### *Completed Response Examples*

Response Without Warnings Example

```
{  
  "manifestTrackingNumber": "100005103ELC"  
}
```

Response With Warnings Example

```
{  
  "manifestTrackingNumber": "100005103ELC",  
  "warningsReport": {  
    "reportId": "012345",  
    "date": "2017-07-27T19:13:28.628+0000",  
    "manifestWarnings": [  
      {  
        "message": "Invalid Format. Text not exceeding 255 characters is expected",  
        "field": "Emanifest.additionalInfo.comments.label",  
        "value": "too many characters"  
      }  
    ],  
    "generatorReport": {  
      "entityId": {  
        "entityIdField": "siteId",  
        "entityIdValue": "MDR12345125"  
      },  
      "warnings": [  
        {  
          "message": "Invalid Format. Numbers not exceeding 6 characters is expected",  
          "field": "Emanifest.generator.emergencyPhone.extension",  
          "value": "N789"  
        }  
      ]  
    },  
    "tsdfReport": {  
      "entityId": {  
        "entityIdField": "siteId",  
        "entityIdValue": "MDR12345125"  
      },  
      "warnings": [  
        {
```





```

{
  "message": "Mandatory Field is not provided",
  "field": "Emanifest.designatedFacility"
},
{
  "message": "No Transporters specified. At least one Transporter is required for
Scheduled Status",
  "field": "Emanifest.transporter"
}
],
"manifestWarnings": [
  {
    "message": "Invalid Field Format. Text value not exceeding 255 characters is
expected",
    "field": "Emanifest.additionalInfo.comments.label",
    "value": "too many characters"
  }
],
"generatorReport": {
  "entityId": {
    "entityIdField": "siteId",
    "entityIdValue": "MDD003473337"
  },
  "errors": [
    {
      "message": "Mandatory Field is not provided",
      "field": "Emanifest.generator.name"
    },
    {
      "message": "Lookup Value Not Found",
      "field": "Emanifest.generator.siteAddress.state.code",
      "value": "NA"
    },
    {
      "message": "Mandatory Field is not provided",
      "field": "Emanifest.generator.mailingAddress"
    }
  ],
  "warnings": [
    {
      "message": "Invalid Format",
      "field": "Emanifest.generator.emergencyPhone.extension",
      "value": "N789"
    }
  ]
},
"tsdfReport": {
  "entityId": {

```

```

        "entityIdField": "siteId",
        "entityIdValue": "N/A"
    },
    "errors": [
        {
            "message": "Mandatory Field is not provided",
            "field": "Emanifest.designatedFacility.siteId"
        },
        {
            "message": "Invalid Format",
            "field": "Emanifest.designatedFacility.siteAddress.zip",
            "value": "X345"
        },
        {
            "message": "Mandatory Field is not provided",
            "field": "Emanifest.designatedFacility.emergencyPhone.number"
        }
    ],
    "warnings": [
        {
            "message": "Invalid Format",
            "field": "Emanifest.designatedFacility.siteAddress.address2",
            "value": ";\u005C"
        }
    ]
},
"wastesReports": [
    {
        "entityId": {
            "entityIdField": "lineNumber",
            "entityIdValue": "1"
        },
        "errors": [
            {
                "message": "Lookup Value Not Found",
                "field": "Emanifest.wastes.dotInformation.idNumber",
                "value": "UN4345"
            },
            {
                "message": "Mandatory Field is not provided",
                "field": "Emanifest.wastes.quantity.quantity"
            },
            {
                "message": "Invalid Field Format.Integer number not exceeding
9999 is expected",
                "field": "Emanifest.wastes.quantity.containerNumber",
                "value": "n"
            }
        ]
    }
]

```

```

    ],
    "warnings": [
      {
        "message": "Invalid Field Format.",
        "field": "Emanifest.wastes.additionalInfo.handlerId",
        "value": "FG056"
      }
    ]
  },
  {
    "entityId": {
      "entityIdField": "lineNumber",
      "entityIdValue": "2"
    },
    "errors": [
      {
        "message": "Waste Description is not provided. Waste Description must be
provided if hazardous is false",
        "field": "Emanifest.wastes.wasteDescription"
      },
      {
        "message": "Density is not provided. Density must be provided if quantity UOM
is Gallons, Liters or Cubic meters",
        "field": "Emanifest.wastes.br.density"
      },
      {
        "message": "Lookup Value Not Found",
        "field": "Emanifest.wastes.managementMethod.code",
        "value": "X25"
      }
    ]
  }
]
}

```

### Errors

Type	code	Description
N/A	E_SecurityApiTokenInvalid	Invalid Security Token
N/A	E_SecurityApiTokenExpired	Security Token is Expired
N/A	E_SecurityApiInvalidStatus	This API Id is no longer active
Authorization	E_IndustryPermissions	The user does not have Industry permissions for any Site
Validation	E_SitePermissions	The user does not have Industry Permissions for the either TSDF Site or Generator Site or Alternate TSDF site specified in provided JSON
System	E_SystemError	Message describing system error

## Delete Manifest Service (post June release)

### Description

The service deletes the manifest by provided Manifest Tracking Number. Depending on the manifest submission type the manifest can be deleted in the following statuses:

- "FullElectronic": can be deleted only if in "Pending" (post June release) or "Scheduled" status
- "Hybrid": can be deleted only if in "Pending" (post June release) or "Scheduled" status
- "DataImage5Copy": can be deleted only if in "ReadyForSignature" status
- "Image": can be deleted only if in "ReadyForSignature" status

### Service Type

Emanifest Data Service.

### Parameters

- Security Token: Security Token received in Authentication. Will be passed in the Header of the request.
- Manifest Tracking Number

### Service Invocation Example

PreProduction/ Test Environment: <Server URL>/rcrainfo/rest/api/v1/emanifest/manifest/delete-manifest/100001380ELC

Production: <Server URL>/rcrainfoprod/rest/api/v1/emanifest/manifest/delete-manifest/100001380ELC

### Sequence of steps

1. The System will validate Security Token
  - 1.1. If Web Security Token is invalid the system stops the submission and will generate following error:
    - E\_SecurityApiTokenInvalid: Invalid Security Token
  - 1.2. If Web Security Token expired the system stops the submission and will generate following error:
    - E\_SecurityApiTokenExpired: Security Token is Expired
  - 1.3. If Account was Inactivated after the token was issued the system stops the submission and will generate following error:
    - E\_SecurityApiInvalidStatus: This API Id is no longer active
2. The system will perform User Authorization.
  - 2.1. System will check if the User has Site Manager (Industry) permissions for any Site.
  - 2.2. If the User does not have permission the System will stop the processing and will generate following error:
    - E\_IndustryPermissions: The user does not have industry permissions for any Site
3. The system will process the request
  - 3.1. The system will check if provided Manifest Tracking Number is valid and Manifest with provided Tracking Number is in the system.

- 3.1.1. Manifest Tracking Number shall be compliant with following rules:
- 3.1.2. Nine numeric characters + valid three character Manifest Tracking Suffix. See the list of valid Manifest Tracking Suffixes in section:
- 3.1.3. If provided Manifest Tracking Number does not have valid format the processing will be stopped and system will generate following error:
  - E\_InvalidManifestTrackingNumber: Provided Manifest Tracking Number has invalid format
- 3.1.4. If Manifest with provided Manifest Tracking Number is not in the system the processing will be stopped and system will generate following error:
  - E\_ManifestTrackingNumberNotFound: Provided Manifest Tracking Number was not found
- 3.2. The system will obtain the Manifest for the provided Manifest Tracking Number.
- 3.3. The system will check if the user has Industry Permissions for the TSDF Site ID provided on the manifest. If the User does not have permission the System will stop the processing and will generate following error:
  - E\_SitePermissions: The user does not have Industry Permissions for the Generator Site or TSDF Site
- 3.4. The system will check if the manifest can be deleted
  - 3.4.1. If Manifest submission type is equal to "FullElectronic" or "Hybrid" and Manifest status is not equal to "Pending" (post June release) or "Scheduled" will stop the processing and will generate following error:
    - E\_ManifestStatus: "FullElectronic" and "Hybrid" Manifest can be Deleted only in "Pending" (post June release) or "Scheduled" status
- 3.5. The System deletes the Manifest
- 4. On success the system returns JSON containing
  - Manifest Tracking Number
  - Status: "Deleted"
  - Date: Date/Time of the deletion
- 5. If any system errors were encountered during processing the system will return:
  - error: containing error code, error message, error id and error date

#### **Response JSON Schema for successful return**

"crud-emanifest-success.json" (not published yet)

#### **Completed Response Example**

```
{  
  "manifestTrackingNumber" : "100001380ELC",  
  "status": "Deleted",  
  "date": "2017-07-27T19:13:28.628+0000"  
}
```

#### **Errors**

Type	code	Description
N/A	E_SecurityApiTokenInvalid	Invalid Security Token
N/A	E_SecurityApiTokenExpired	Security Token is Expired

N/A	E_SecurityApiInvalidStatus	This API Id is no longer active
Authorization	E_IndustryPermissions	The user does not have Industry permissions for any Site
Validation	E_InvalidManifestTrackingNumber	Provided Manifest Tracking Number has invalid format
Validation	E_ManifestTrackingNumberNotFound	Provided Manifest Tracking Number was not found
	E_ManifestStatus	“FullElectronic” or “Hybrid” Manifest can be Deleted only in “Scheduled” status
	E_ManifestStatus	“DataImage5Copy” or “Image” Manifest can be Deleted only in “ReadyForSignature” status
System	E_SystemError	Message describing system error

#### Validation Error Return Example

```
{
  "code": "E_ManifestTrackingNumberNotFound ",
  "message": "Provided Manifest Tracking Number is Not Found"
  "errorId" : "9v7a9808-s35a5-4874-928f-12fc3def4b56",
  "date" : "2017-06-23T23:15:45.095+0000"
}
```

### Manifest entities and fields validation for Save service

#### 1. Emanifest.manifestTrackingNumber

- 1.1. If submissionType is “FullElectronic” or “Hybrid” and Emanifest.manifestTrackingNumber is provided, the service will generate a warning:

```
{
  "message": "Provided Manifest Tracking Number will be ignored",
  "field": "Emanifest.manifestTrackingNumber",
  "value": "mtn value"
}
```

- 1.2. If submissionType is “DataImage5Copy” or “Image” following applies

- 1.2.1. If Emanifest.manifestTrackingNumber is not provided, the service will generate an error:

```
{
  "message": "Mandatory Field is not Provided",
  "field": "Emanifest.manifestTrackingNumber"
}
```

- 1.2.2. If Emanifest.manifestTrackingNumber is provided, the service will check if manifestTrackingNumber has a valid format. If manifestTrackingNumber has invalid format, the service will generate an error:

```
{
  "message": "Invalid Field Format",
  "field": "Emanifest.manifestTrackingNumber",
}
```

```
    "value": "mtn value"
}
```

- 1.2.3. If manifestTrackingNumber has a valid format, the service checks if the provided manifestTrackingNumber suffix is valid. All printed suffixes except ELC are valid. If the suffix is invalid the service generates an error:

```
{
  "message": "Invalid Manifest Tracking Number Suffix is Provided",
  "field": "Emanifest.manifestTrackingNumber",
  "value": "manifestTrackingNumber value"
}
```

- 1.2.4. If manifestTrackingNumber has a valid format, the service will check if a manifest with the provided manifestTrackingNumber is already stored in the e-Manifest database. If the manifest with the provided manifestTrackingNumber is already stored in the e-Manifest database, the system will generate the following error:

```
{
  "message": "Manifest with provided Manifest Tracking Number is already stored in the e-Manifest database",
  "field": "Emanifest.manifestTrackingNumber",
  "value": "mtn value"
}
```

## 2. Emanifest.status.

- 2.1. If the submission type is "DataImage5Copy" or "Image" and provided status is not "ReadyForSignature", the service will generate the following warning:

```
{
  "message": "Provided Status will be ignored. e-Manifest will be assigned ReadyForSignature status",
  "field": "Emanifest.status",
  "value": "status value"
}
```

- 2.2. If the submission type is "FullElectronic" or "Hybrid" following applies

- 2.2.1. If status is not provided the service will generate the following error:

```
{
  "message": "Mandatory Field is not Provided",
  "field": "Emanifest.status"
}
```

- 2.2.2. If provided status is not equal to "Pending" (post June release) or "Scheduled" the service will generate the following error:

```
{
  "message": "Invalid Value is Provided. Manifest can be saved in \"Scheduled\" status If the submission type is \"FullElectronic\" or \"Hybrid\"",
  "field": "Emanifest.status",
  "value": "status value"
}
```

- }
3. `Emanifest.submissionType`.
    - 3.1. If the `submissionType` is not: `"FullElectronic"`, `"DataImage5Copy"`, `"Image"`, or `"Hybrid"` the service will generate following error:
 

```
{
    "message": "Invalid Field Format. One of the following values "FullElectronic",
    "DataImage5Copy", "Image", or "Hybrid" is expected",
    "field": "Emanifest.submissionType",
    "value": "status value"
}
```
    - 3.2. If `submissionType` is not provided, the service will generate the following error:
 

```
{
    "message": "Mandatory field is not provided",
    "field": "Emanifest.submissionType"
}
```
  4. Validation of the State Waste Codes applies only for the `"FullElectronic"`, `"DataImage5Copy"` or `"Hybrid"` submission types.
    - 4.1. It is possible that if the Waste is not hazardous no State Codes are provided (Federal Waste codes shall not be provided for non-hazardous waste. This is covered in section Hazardous Waste Validation). Manifest is not valid without a Waste Code. At least one State Waste Code shall be provided for the Manifest.
    - 4.2. If `submissionType` is `"FullElectronic"`, `"DataImage5Copy"` or `"Hybrid"` and no State Waste Codes are provided, the service generates an error:
 

```
{
    "message": "Manifest does not have any Waste Codes. Valid Manifest requires at least
    one Waste Code."
    "field": "Emanifest.Waste.hazardousWaste"
}
```
  5. Waste(s) Line Numbers Validation applies only for the `"FullElectronic"`, `"DataImage5Copy"` or `"Hybrid"` submission types
    - 5.1. If applies only for the `"FullElectronic"`, `"DataImage5Copy"` or `"Hybrid"` submission types following applies
      - 5.1.1. `Emanifest.waste.lineNumber` for each Waste must be provided. If `lineNumber(s)` is/are not provided the service generates an error:
 

```
{
    "message": "Mandatory Field is not Provided.",
    "field": "Emanifest.waste.lineNumber"
}
```
      - 5.1.2. If `lineNumber` is not a valid format, the service generates an error:
 

```
{
    "message": "Invalid Field format. Numeric value expected",
    "field": "Emanifest.waste.lineNumber",
```



- ```

    "value": "lineNumber value"
  }

```
- 5.1.3. If one waste is provided and the lineNumber is not equal to 1, the service generates a warning:
- ```

{
  "message": "Manifest expected to start with line 1",
  "field": "Emanifest.waste.lineNumber",
  "value": "order value"
}

```
- 5.1.4. If multiple wastes are provided and lineNumbers are not sequential, the service generates an error:
- ```

{
  "message": "Invalid value(s). Sequential waste line numbers are expected"
  "field": "Emanifest.wastes.lineNumber",
  "value": "lineNumber 1"
}
{
  "message": "Invalid value(s). Sequential waste line numbers are expected"
  "field": "Emanifest.wastes.lineNumber",
  "value": "lineNumber 2"
}

```
6. Transporters. Applies only for the **"FullElectronic"**, **"DataImage5Copy"** or **"Hybrid"** submission types
- 6.1. If the transporters entity is not provided, the service generates the following error:
- ```

{
  "message": "Mandatory Field is not Provided",
  "field": "Emanifest.transporters"
}

```
7. Transporter(s) order(s) validation. Applies only for the **"FullElectronic"**, **"DataImage5Copy"** or **"Hybrid"** submission types
- 7.1. If Transporter.order is not provided the service generates an error:
- ```

{
  "message": "Value is not provided"
  "field": "Emanifest.transporter.order"
}

```
- 7.2. If Transporter.order is an invalid format, the service generates an error:
- ```

{
  "message": "Invalid Field format. Numeric value expected",
  "field": "Emanifest.transporter.order",
  "value": "order value"
}

```
- 7.3. Validate consecutiveness of the Transporters order.

- 7.3.1. If one transporter is provided and the order is not equal to 1, the service generates an error:

```
{  
  "message": "Invalid Value",  
  "field": "Emanifest.transporter.order",  
  "value": "order value"  
}
```

- 7.3.2. If multiple transporters are provided and orders are not consecutive, the service generates an error:

```
{  
  "message": "Invalid value(s). Consecutive Transporter Orders are expected"  
  "field": "Emanifest.transporter.order",  
  "value": "order value 1"  
}  
  
{  
  "message": "Invalid value(s). Consecutive Transporter Orders are expected"  
  "field": "Emanifest.transporter.order",  
  "value": "order value 2"  
}
```

8. If submissionType is equal to **"Image"** Validate Facility Information. See details in section [Facility Information Validation](#) for "Image" submission type.
9. If submissionType is equal to the **"FullElectronic"** or **"DataImage5Copy"** or **"Hybrid"** Validate Import Information. See details in [Import Information Validation](#) section.
10. If submissionType is equal to the **"FullElectronic"** or **"DataImage5Copy"** or **"Hybrid"** Validate Rejection Information. See details in [Rejection Information Validation](#) section.
11. If submissionType is equal to the **"FullElectronic"** or **"DataImage5Copy"** or **"Hybrid"** Validate New Manifest Information. See details in [New Manifest Information Validation](#) section.
12. If submissionType is equal to the **"FullElectronic"** or **"DataImage5Copy"** or **"Hybrid"** Validate Discrepancy Information. See details in [Discrepancy fields validation](#) section.
13. If submissionType is equal to the **"FullElectronic"** or **"DataImage5Copy"** or **"Hybrid"** Validate Residue Information. See details in [Residue Fields Validation](#) section.
14. Validate Generator Information. See details in [Generator Site Id and Site Information Validation](#) section.
15. If submissionType is equal to the **"FullElectronic"** or **"DataImage5Copy"** or **"Hybrid"** Validate Transporters Information. See details in [Transporter Site ID and Site Information Validation](#) section.
16. Validate Designated Facility Information. See detail [in](#) Designated Facility (TSDF)/Generator Site Id and Site Information Validation section.
17. If submissionType is equal to the **"FullElectronic"** or **"DataImage5Copy"** or **"Hybrid"** Validate Wastes Information. See details in [Waste Validation](#) section.
18. If submissionType is equal to the **"DataImage5Copy"** Validate Paper Signature Info Validation. See details in [Paper Signature Info Validation](#) section.

19. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Manifest Additional Information. See details Manifest Additional Information Validation section.
20. If submissionType is equal to "DataImage5Copy" or "Image" validate Attachment Information. See details Attachment Validation section.

## Manifest entities and fields validation for Update service

### 1. Emanifest.manifestTrackingNumber

- 1.1. If Emanifest.manifestTrackingNumber is not provided, the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.manifestTrackingNumber"  
}
```

- 1.2. Emanifest.manifestTrackingNumber is provided and it has an invalid format, the service generates an error:

```
{  
  "message": "Invalid Field Format. 9 digits followed by 3 upper case letters is expected",  
  "field": "Emanifest.manifestTrackingNumber",  
  "value": "manifestTrackingNumber value"  
}
```

- 1.3. If the manifestTrackingNumber has a valid format, the service checks if the provided manifestTrackingNumber suffix is valid. If the suffix is invalid, the service generates an error:

```
{  
  "message": "Invalid Manifest Tracking Number Suffix is Provided",  
  "field": "Emanifest.manifestTrackingNumber",  
  "value": "manifestTrackingNumber value"  
}
```

- 1.4. If the manifestTrackingNumber has a valid format, the service checks if a Manifest with the provided manifestTrackingNumber is already stored in the e-Manifest database. If the Manifest was not found in the e-Manifest database, the service generates an error:

```
{  
  "message": "Manifest with provided Manifest Tracking Number was not found.  
Manifest cannot be updated",  
  "field": "Emanifest.manifestTrackingNumber",  
  "value": "manifestTrackingNumber value"  
}
```

### 2. Emanifest.status.

- 2.1. If submissionType is equal to "FullElectronic" or "Hybrid", then the status can be changed from "Pending" to "Scheduled" via update service (post June release). Statuses after "Scheduled" cannot be changed via the update service. Status transitions after "Scheduled" will be done by e-Manifest and are triggered by the following events:

- 2.3.1. If there is one transporter, the following status transition is possible:

- Scheduled->ReadyForSignature: The status transition is triggered when the Transporter signs.
  - ReadyForSignature->Signed: The status transition is triggered when the TSDF signs.
- 2.3.2. If there is more than one transporter following status transition is possible:
- Scheduled->InTransit: The status transition is triggered when the 1<sup>st</sup> Transporter signs.
  - InTransit -> ReadyForSignature: Status transition is triggered when last Transporter signs.
  - ReadyForSignature->Signed: The status transition is triggered when TSDF signs.
- 2.3.3. (post June release) If the current status is "Pending" and provided status is not "Pending" or "Scheduled" , the service generates an error:

```
{
  "message": "For the FullElectronic and Hybrid submission type Manifest status
cannot be updated to statuses after "Scheduled" via Update Manifest service",
  "field": "Emanifest.status",
  "value": "status value"
}
```

- 2.3.4. If the current status is "Scheduled" and provided status is not "Scheduled" , the service generates a warning:

```
{
  "message": "Provided Value will be ignored. For the FullElectronic and Hybrid
submission type Manifest status cannot be updated to statuses before or after
"Scheduled" via Update Manifest service",
  "field": "Emanifest.status",
  "value": "status value"
}
```

- 2.3.5. If the current status is "InTransit" and provided status is not "InTransit" , the service generates a warning:

```
{
  "message": "Provided Value will be ignored. For the FullElectronic and Hybrid
submission type Manifest status cannot be updated to statuses before or after
"Scheduled" via Update Manifest service",
  "field": "Emanifest.status",
  "value": "status value"
}
```

- 2.3.6. If the current status is "Received" and provided status is not "Received" , the service generates a warning:

```
{
  "message": "Provided Value will be ignored. For the FullElectronic and Hybrid
submission type Manifest status cannot be updated to statuses before or after
"Scheduled" via Update Manifest service",
  "field": "Emanifest.status",
  "value": "status value"
}
```

}

2.3.7. If the current status is "ReadyForSignature" and provided status is not "ReadyForSignature", the service generates a warning:

{

"message": "Provided Value will be ignored. For the FullElectronic and Hybrid submission type Manifest status cannot be updated to statuses before or after "Scheduled" via Update Manifest service",

"field": "Emanifest.status",

"value": "status value"

}

2.3.8. If the current status is "Signed" and provided status is not "Signed", the service generates a warning:

{

"message": "Provided Value will be ignored. For the FullElectronic and Hybrid submission type Manifest status cannot be updated to statuses before or after "Scheduled" via Update Manifest service",

"field": "Emanifest.status",

"value": "status value"

}

2.4. If submissionType is equal to "DataImage5Copy", or "Image" then status cannot be changed via the update service. The status transition will be done by e-Manifest and is triggered by the following event:

- ReadyForSignature -> Signed: Status transition is triggered when the TSDF electronically signs.

2.4.1. If Manifest status is not provided, no warnings or errors shall be generated.

2.4.2. The Manifest can be submitted only in the ReadyForSignature status. Any other status will be ignored with a warning:

{

"message": "Provided Manifest status will be ignored. For the DataImage5Copy submission type Manifest status is ReadyForSignature and cannot be updated to any other status via Update Manifest service",

"field": "Emanifest.status",

"value": "status value"

}

2.4.3. If the provided status is different than the current status, the service generates a warning:

{

"message": "Provided Manifest status will be ignored. For the DataImage5Copy submission type Manifest status cannot be updated to any other status via Update Manifest service",

"field": "Emanifest.status",

"value": "status value"

}

3. Manifest.submissionType.

3.3. If submissionType is equal to "DataImage5Copy" or "Image" then the submissionType cannot be updated

3.3.1. If the current submissionType == "DataImage5Copy" or "Image" and any other submissionType is provided the system will generate a warning:

```
{  
  "message": "Provided Submission Type will be ignored. Submission Type cannot be  
updated",  
  "field": "Manifest. submissionType",  
  "value": "submissionType value"  
}
```

3.4. If status is not "Pending" (post June release) or "Scheduled" and the submissionType is provided, it will be ignored with warning:

```
{  
  "message": "Provided Submission Type will be ignored. Submission Type can be updated  
only at Scheduled status",  
  "field": "Manifest. submissionType",  
  "value": "submissionType value"  
}  
}
```

4. If submissionType is equal to "Image" Validate Facility Information. See details in section Facility Information Validation for "Image" submission type.

5. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate the State Waste Codes. This section is identical to Validation of the State Waste Codes in Manifest entities and fields validation

6. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Waste(s) Line Numbers . This section is identical to Waste(s) Line Numbers Validation in Manifest entities and fields validation

7. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Transporter(s) order(s). This section is identical to Transporter(s) order(s) validation in Manifest entities and fields validation

8. Validate Generator Information. See details in Generator Site Id and Site Information Validation section

9. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Import Information. See details in Import Information Validation section.

10. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Rejection Information. See details in [Rejection Information Validation](#) section.

11. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate New Manifest Information. See details [in New Manifest Information Validation](#) section.

12. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Discrepancy Information. See details in Discrepancy fields validation section.
13. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Residue Information. See details in Residue Fields Validation section.
14. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Transporters Information. See details in Transporter Site ID and Site Information Validation section.
15. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Designated Facility Information. See detail in Designated Facility (TSDf)/Generator Site Id and Site Information Validation section.
16. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Wastes Information. See details in Waste Validation section.
17. If submissionType is equal "DataImage5Copy" validate Paper Signature Information. See details in Paper Signature Info Validation section
18. If submissionType is equal to the "FullElectronic" or "DataImage5Copy" or "Hybrid" Validate Manifest Additional Information. See details section Manifest Additional Information Validation.
19. If submissionType is equal to DataImage5Copy or "Image" validate Attachment Information. See details in Attachment Validation section.

#### Facility Information Validation for "Image" submission type

1. For Image submission type one of the following facility EPA site id must be provided: Designated Facility EPA Site Id or Generator EPA Site Id or Alternate Designated Facility Site Id
2. Only one facility EPA site id must be provided. Emanifest will determine "billable" facility by provided facility Id.
3. Following Facility EPA site ids shall be provided for the following scenarios:
  - Original Manifest, no Full Rejection: Designated Facility EPA site Id shall be provided
  - Original Manifest, Full Rejection to Alternate Designated Facility: Alternate Designated Facility EPA site Id shall be provided.
  - Original Manifest, Full Rejection to The Generator: Designated Facility EPA site Id shall be provided
  - New Manifest shipping waste back to original Generator (New Manifest is the manifest created as a "result" of the Original manifest rejection or residue): Generator Facility EPA site Id shall be provided. This EPA site id shall contain Original Designated Facility Site Id
  - New Manifest shipping waste back to another Designated Facility: Designated Facility EPA site Id shall be provided
4. If Emanifest.designatedFacility.epaSiteId and Emanifest.generator.epaSiteId and Emanifest.rejectionInfo.alternateDesignatedFacility.siteId is not provided then the service will return an authentication error
5. If provided EPA Facility site Id is not registered then the service will return an authentication error

6. If more than one of the following facility EPA site ids is provided then the service generates following errors (example is for the case if both Generator and Designated Facility EPA site IDs are provided):

```
{
  "message": "For Image submission type only one of the following facility EPA site id must be
provided: Designated Facility EPA Site Id or Generator EPA Site Id or Alternate Designated
Facility Site Id ."
  "field": "Emanifest.generator. epaSiteId"
}
{
  "message": "For Image submission type only one of the following facility EPA site id must be
provided: Designated Facility EPA Site Id or Generator EPA Site Id or Alternate Designated
Facility Site Id ."
  "field": "Emanifest.designatedFacility.epaSiteId"
}
```

7. If Emanifest.designatedFacility.epaSiteId or Emanifest.generator.epaSiteId or  
Emanifest.rejectionInfo.alternateDesignatedFacility.siteId is provided following applies

7.1. If any following Site information is provided then the service generates following warning:

- Site Name
- Site Address
- Mailing Address

```
{
  "message": "Provided Values will be Ignored. The site's registered RCRAInfo values will be
used",
  "field": "Emanifest.designatedFacility.name /siteAddress/ mailingAddress",
  "value": "name/siteAddress/ mailingAddress"
}
```

7.2. Validation of the Contact information is identical to #2 in section [Designated Facility \(TSDF\)/Generator Site Id and Site Information Validation](#)

7.3. If Emanifest.wastes are provided then the service generate following warning:

```
{
  "message": "Provided Value will be Ignored.",
  "field": "Emanifest.wastes"
}
```

7.4. If provided Emanifest.residue == true then the service generate following warning:

```
{
  "message": "Provided Value will be Ignored. Emanifest.residue = false will be assigned" ",
  "field": "Emanifest.residue",
  "value": " residue value"
}
```

7.5. If provided Emanifest.import == true then the service generate following warning:

```
{
```



```
"message": "Provided Value will be Ignored. Emanifest.rejection = false will be assigned" ",
"field": "Emanifest.import",
"value": "import value"
}
```

7.6. If provided Emanifest.containsPreviousRejectOrResidue == true then the service generate following warning:

```
{
"message": "Provided Value will be Ignored. Emanifest.containsPreviousRejectOrResidue = false will be assigned" ",
"field": "Emanifest.containsPreviousRejectOrResidue",
"value": "containsPreviousRejectOrResidue value"
}
```

7.7.

7.8. The service stores following information as a part of the manifest data:

- Emanifest.residue = false
- Emanifest.import = false
- Emanifest.containsPreviousRejectOrResidue = false

8. If Emanifest.designatedFacility.epaSiteId or Emanifest.generator.epaSiteId is provided then following applies

8.1. If provided Emanifest.rejection == true then the service generate following warning:

```
{
"message": "Provided Value will be Ignored. Emanifest.rejection = false will be assigned",
"field": "Emanifest.rejection",
"value": " rejection value"
}
```

8.2. If any of the Emanifest.rejectionInfo fields are provided then then the service generate following warning:

```
{
"message": "Provided Value will be Ignored.",
"field": "Emanifest.rejectionInfo.{field}",
"value": "field value"
}
```

9. If Emanifest.rejectionInfo.alternateDesignatedFacility.siteId is provided then following applies

9.1. If provided Emanifest.rejection == false then the service generate following warning:

```
{
"message": "Provided Value will be Ignored. Emanifest.rejection = true will be assigned",
"field": "Emanifest.rejection",
"value": " rejection value"
}
```

9.2. If provided Emanifest.rejectionInfo.transporterOnSite == false then the service generate following warning:

```
{
```

"message": "Provided Value will be Ignored. Emanifest.rejectionInfo.transporterOnSite = true will be assigned",

"field": "Emanifest.rejectionInfo.transporterOnSite",

"value": "rejectionInfo.transporterOnSite value"

}

9.3. If provided Emanifest.rejectionInfo.rejectionType == "PartiaReject" then the service generate following warning:

{

"message": "Provided Value will be Ignored. Emanifest.rejectionInfo.rejectionType = "FullReject" will be assigned",

"field": "Emanifest.rejectionInfo.rejectionType",

"value": "rejectionInfo.rejectionType value"

}

9.4. If provided Emanifest.rejectionInfo.alternateDesignatedFacilityType == "OriginalGenerator" then the service generate following warning:

{

"message": "Provided Value will be Ignored. Emanifest.rejectionInfo.alternateDesignatedFacilityType = "TsdF" will be assigned",

"field": "Emanifest.rejectionInfo.alternateDesignatedFacilityType",

"value": "rejectionInfo.alternateDesignatedFacilityType value"

}

9.5. The service stores following information as a part of the manifest data:

- Emanifest.rejection = true
- Emanifest.rejectionInfo.transporterOnSite=true
- Emanifest.rejectionInfo.rejectionType = "FullReject"
- Emanifest.rejectionInfo.alternateDesignatedFacilityType = "TsdF"

### Designated Facility (TSDF)/Generator Site Id and Site Information Validation

TSDF can act either as TSDF or as Generator. This will be determined during the authorization step. If TSDF acts as a TSDF the system requires the handler ID to be provided as the TSDF Site ID. If other information for the Site is passed, it will be ignored with warnings and only the registered information will be used. If the TSDF acts as a Generator the system requires the handler's ID to be placed in the Generator Site Id

1. If submissionType is equal to the "FullElectronic" then following applies

1.1.1.2. If Emanifest.generator.epaSiteId is not provided then the service generates following error:

{

"message": "Mandatory Field is not Provided",

"field": "Emanifest.generator.epaSiteId",

}

1.2. If Emanifest. generator.epaSiteId has wrong format then service generates following error:

```
{
  "message": "Invalid Field Format",
  "field": "Emanifest. generator.epaSiteId",
  "value": " epa site id value"
}
```

1.3. If Emanifest.generator.epaSiteId is not registered in RCRAInfo then service generates following error:

```
{
  "message": "Provided Generator Facility Id is not registered in RCRAInfo",
  "field": "Emanifest. generator.epaSiteId",
  "value": " epa site id value"
}
```

2. If submissionType is equal to the "FullElectronic", "DataImage5Copy" or "Hybrid" then following applies

2.1. If Emanifest.designatedFacility.epaSiteId is not provided then the service generates following error:

```
{
  "message": "Mandatory Field is not Provided",
  "field": "Emanifest.designatedFacility.epaSiteId",
}
```

2.2. If Emanifest.designatedFacility.epaSiteId has wrong format then service generates following error:

```
{
  "message": "Invalid Field Format",
  "field": "Emanifest.designatedFacility.epaSiteId",
  "value": " epa site id value"
}
```

2.3. If Emanifest.designatedFacility.epaSiteId is not registered in RCRAInfo then service generates following error:

```
{
  "message": "Provided Designated Facility Id is not registered in RCRAInfo",
  "field": "Emanifest.designatedFacility.epaSiteId",
  "value": " epa site id value"
}
```

2.4. If the Site Contact Phone number is not provided and registered in RCRAInfo Site does not contain the Site Contact Phone Number, the system generates the following error:

```
{
  "message": "Mandatory Field is not Provided",
  "field": "Emanifest.designatedFacility.contact.phone.number",
}
```

2.5. If the Site Contact Phone number is provided, the system will check the format.

2.5.1. If the provided phone number has a valid format the system will store site contact phone number into e-Manifest database

2.5.2. If the provided phone number has an incorrect format the system will generate the following warning:

```
{  
  "message": "Invalid Field format. Does not match phone number format of 999-999-9999",  
  "field": "Emanifest.designatedFacility.contact.phone.number ",  
  "value": "number value"  
}
```

2.6. If the Site Contact Phone Extension is provided, the system will check the format.

2.6.1. If provided phone extension has a valid format, the system will store the site contact phone extension into e-Manifest database

2.6.2. If the provided phone has an incorrect format, the system will generate the following warning:

```
{  
  "message": "Invalid Field format",  
  "field": "Emanifest.designatedFacility.contact.phone.extension ",  
  "value": "extension value"  
}
```

2.7. If any of the following TSDf Site Information entities were provided:

- Site Name
- Site Address
- Mailing Address

The system generates the following warning:

```
{  
  "message": "Provided Values will be Ignored. The site's registered RCRAInfo values will be used",  
  "field": "Emanifest.designatedFacility.name /siteAddress",  
  "value": "name/siteAddress"  
}
```

2.8. If the user is authorized to use the e-Manifest Save service for the provided Generator Site ID (It means that TSDf acts as Generator), then the provided Site ID was already validated during authorization and the system will perform following steps:

2.9. If the Site Contact Phone number is not provided in the JSON and site the is registered in RCRAInfo, the system checks if the site has a Contact Phone value registered for the site. If the Contact Phone is not registered in the system for that site, the system generates the following error:

```
{
```

```
"message": "Mandatory Field is not provided"
"field": "Emanifest.generator.contact.Phone.number"
}
```

2.10. If the Site Contact Phone number is provided and registered in RCRAInfo the system checks that the phone number is in the correct format.

2.10.1. If the provided phone number has a valid format, the system will store site contact phone number into e-Manifest database

2.10.2. If provided phone number has an incorrect format, the system will store registered in RCRAInfo Contact Phone number into e-Manifest database and generate the following warning:

```
{
"message": "Invalid Field format. Does not match phone number format of 999-999-9999",
"field": "Emanifest.generator.contact.phone.number ",
"value": "number value"
}
```

2.11. If the Site Contact Phone number is provided and registered in RCRAInfo site does not contain Contact Phone.

2.11.1. If the provided phone number has valid format, the system will store site contact phone number into e-Manifest database

2.11.2. If provided phone number has incorrect format, the system generates the following error:

```
{
"message": "Invalid Field format. Does not match phone number format of 999-999-9999",
"field": "Emanifest.generator.contact.phone.number ",
"value": "number value"
}
```

2.12. Validate emergency phone

2.12.1. If emergency phone number is not provided, the system generates the following error:

```
{
"message": "Mandatory Field is not provided"
"field": "Emanifest.generator.emergencyPhone.number"
}
```

2.12.2. If emergency phone number is provided and has an invalid format, the system generates the following error:

```
{
"message": "Provided Value is not Valid. Does not match phone number format of 999-999-9999",
"field": "Emanifest.generator.emergencyPhone.number ",
"value": "phone number value"
}
```

2.12.3. If emergency phone extension is provided and has an invalid format, the system generates the following warning:

```
{  
  "message": "Provided Value is not Valid. Does not match phone extension format of  
999999",  
  "field": "Emanifest.generator.emergencyPhone.extension ",  
  "value": "extension value"  
}
```

2.13. If Site Name, Site Location, Site Mailing Address are provided, the system generates the following warning:

```
{  
  "message": "Provided Values will be Ignored. RCRAInfo values will be used",  
  "field": "Emanifest.genrator.name/maillingAddress/siteAddress ",  
  "value": "name/maillingAddress/siteAdress value"  
}
```

2.14. If the contact email was provided the system will check the format

2.14.1. If provided contact email has a valid format the system will store site contact email into e-Manifest database

2.14.2. If the provided contact email has an incorrect format the system generates the following warning:

```
{  
  "message": "Invalid Field format. Valid email format is expected.",  
  "field": "Emanifest.designatedFacility.contact.email ",  
  "value": "email value"  
}
```

3. Update validation.

3.1. Original epaSiteId submitted via Save e-Manifest service cannot be updated via Update e-Manifest service.

3.2. If the provided designatedFacility.epaSiteId does not match current designatedFacility.epaSiteId in the Update e-Manifest service the following warning will be generated:

```
{  
  "message": "Provided Field will be ignored. Originally Submitted Designated Facility EPA  
Site Id cannot be updated",  
  "field": "Emanifest.designatedFacility.epaSiteId",  
  "value": "epaSiteId value"  
}
```

3.3. designatedFacility.Contact.phone number can be updated at any status

## Generator Site Id and Site Information Validation

1. If submissionType == "FullElectronic" then the Generator is valid if the Generator Site ID is registered in RCRAInfo and the Generator has at least one User with the e-Manifest Certifier Role and this user has ESA == Received
2. If submissionType == FullElectronic following applies
  - 2.1. If the Generator Site ID is not provided the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided. For FullElectronic submission type  
registered Generator Site Id must be provided",  
  "field": "Emanifest.designatedFacility.epaSiteId"  
}
```
  - 2.2. If the Generator Site ID has an invalid format the service generates an error:

```
{  
  "message": "Invalid Field Format. For FullElectronic submission type registered  
Generator Site Id must be provided ",  
  "field": "Emanifest.designatedFacility.epaSiteId",  
  "value": "Site ID value"  
}
```
  - 2.3. If the Generator Site ID is not registered in RCRAInfo the service generates an error:

```
{  
  "message": "For FullElectronic submission type a registered Generator Site Id must be  
provided ",  
  "field": "Emanifest.designatedFacility.epaSiteId",  
  "value": "Site ID value"  
}
```
  - 2.4. If the Generator Site ID is registered in RCRAInfo and there are no users with the e-Manifest Certifier role for the provided Generator the service generates an error:

```
{  
  "message": "No Users with Certifier Role found for the provided Generator Site Id",  
  "field": "Emanifest.designatedFacility.epaSiteId",  
  "value": "Site ID value"  
}
```
  - 2.5. If the Generator with the provided Site ID is registered and there are no users with ESA == Received the service generates an error:

```
{  
  "message": "No Users which can Electronically sign found for the provided Generator  
Site Id",  
  "field": "Emanifest.designatedFacility.epaSiteId",  
  "value": "Site ID value"  
}
```

3. If `submissionType` is "DataImage5Copy" or "Hybrid", then registered and non-registered Generators are valid and the following cases are possible for the Generator Site ID and Generator Information:
- 3.1. If the Generator is registered in RCRAInfo and service requester does not intend to provide different (than registered) Site Information, then only the `epaSiteId` shall be provided. All site information will be obtained from RCRAInfo.
- 3.2. If the Generator is registered in RCRAInfo and service requester intends to provide different (than registered) Site Information, then the following site information entities must be provided:
- Site ID
  - Site Name
  - Site Address (all fields)
  - Mailing Address (all fields)
  - Contact Phone

Provided Site Information will be validated and if valid will be stored in e-Manifest.

- 3.3. If Generator is not registered in RCRAInfo, then following site information must be provided:
- Site Name
  - Site Address (all fields)
  - Mailing Address (all fields)
  - Contact phone

Requester also can provide Site ID which is an optional field for this case. If Site ID is provided the system will check if there is a registered Generator for the provided Site ID

- 3.4. For all above cases Emergency Phone Number must be provided
- 3.5. If the manifest status is InTransit or thereafter then Generator information cannot be updated.
- 3.6. If different than currently stored Generator information is provided the service generates the following warning:

```
{  
  "message": "Provided Generator information will be ignored. Generator Information  
cannot be updated if the manifest status is InTransit or thereafter"  
  "field": "Emanifest.generator"  
}
```

- 3.7. If the Site ID is not provided AND any of the mandatory Site information entities are not provided, the service generates an error:

```
{  
  "message": "Mandatory Field is not provided"  
  "field": "Emanifest.generator.epaSiteId"  
}  
  
{  
  "message": "Mandatory Field is not provided"  
  "field": "Emanifest.generator.siteAddress"
```



```

}
{
    "message": "Mandatory Field is not provided"
    "field": "Emanifest.generator.mailingAddress"
}

```

3.8. If the Site ID is provided AND Site Address information is not provided, the system performs the following steps:

3.8.1. If Site ID is not valid the service generates an error:

```

{
    "message": "Provided Value is not Valid. Does not match format of: Two Letter Activity Location Code + Up to 10 alphanumeric characters",
    "field": "Emanifest.generator.epaSiteld ",
    "value": "EPA site Id value"
}

```

3.8.2. If Site ID is valid and found in RCRAInfo, the system will obtain the Site information from RCRAInfo and store it into the Manifest. System sets modified= false, registered = true

3.8.3. If the Site ID is not found, the service generates an error:

```

{
    "message": "Provided Value is not Found",
    "field": "Emanifest.generator.epaSiteld",
    "value": "EPA Site ID value"
}

```

3.9. If the Site ID is provided and the Site Information entities are provided, the system performs the following steps:

3.9.1. If the site name is not provided the service generates an error:

```

{
    "message": "Mandatory Field is not Provided",
    "field": "Emanifest.generator.name"
}

```

3.9.2. If the site name not valid (exceeds maximum length), the service generates an error:

```

{
    "message": "Provided Value is not Valid. Exceeded maximum length of 80",
    "field": "Emanifest.generator.name",
    "value": "name value"
}

```

3.9.3. Validate the mandatory location address fields:

- address1 (50 Character maximum)
- city (25 Character maximum)
- state.code (2 Character State abbreviation)
- zip (14 Character maximum)

3.9.3.1. If any of the mandatory location address fields are not provided or invalid, the service generates an error:

```
{
  "message": "Value is not Provided/Provided Value is not Valid",
  "field": "Emanifest.generator.siteAddress.address1/city/state/zip",
  "value": "address1/city/state/zip value"
}
```

3.9.4. Validate the mandatory mailing address fields:

- address1 (50 Character maximum)
- city (25 Character Maximum)
- state.code (2 Character State Abbreviation)
- zip (14 Character Maximum)

3.9.4.1. If any of the mandatory mailing address fields are not provided or invalid, the service generates an error:

```
{
  "message": "Value is not Provided/Provided Value is not Valid",
  "field": "Emanifest.generator.mailingAddress.address1/city/state/zip",
  "value": "address1/city/state/zip provided value"
}
```

3.10. Validate provided Site ID

3.10.1. If Site ID is not valid, the service generates a warning:

```
{
  "message": "Provided value is not valid. Two Letter Activity Location Code + Up to 10 alphanumeric characters",
  "field": "Emanifest.generator.epaSiteId",
  "value": "EPA Site ID value"
}
```

3.10.2. If the Site ID is valid the system checks if the site is registered

3.10.3. If the site is registered, the system validates if the provided site address state code matches Site ID activity location.

3.10.3.1. If the state code does not match the Site ID activity location, the service generates an error:

```
{
  "message": "Location State Code is different than Site Id Activity Location",
  "field": "Emanifest.generator.siteAddress.state.code",
  "value": "site address state code value"
}
```

3.10.4. If the site is not registered, the service generates a warning:

```
{
  "message": "Site with provided EPA Site ID is not registered",
  "field": "Emanifest.generator.epaSiteId",
}
```

"value": "Site ID value"

  }

3.10.4.1. Store provided Site Information (if no other manifest errors were found)

4. If the Site ID is not provided AND the Site Information is provided, the system performs the following steps:

4.1. Validate the provided Site Information.

4.1.1. Validate the site name

4.1.2. If the site name is not valid the service generates an error:

```
{
  "message": "Provided Value is not Valid",
  "field": "Emanifest.generator.name",
  "value": "name value"
}
```

4.1.3. Validate the mandatory site location address fields:

- address1
- city
- state.code
- zip

4.1.4. If any of the mandatory location address fields are not provided or invalid, the service generates an error:

```
{
  "message": "Value is not Provided/Provided Value is not Valid",
  "field": "Emanifest.generator.siteAddress.address1/city/state/zip",
  "value": "address1/city/state/zip value"
}
```

4.1.5. Validate the mandatory mailing address fields:

- address1
- city
- state.code
- zip

4.1.6. If any of the mandatory mailing address fields are not provided or invalid, the service generates an error:

```
{
  "message": "Value is not Provided/Provided Value is not Valid",
  "field": "Emanifest.generator.mailingAddress.address1/city/state/zip",
  "value": "address1/city/state/zip provided value"
}
```

4.2. If the provided Site Information is valid, but the Site ID is not provided the system performs the following steps:

4.2.1. The service generates a warning:

```
{
  "message": "Value is not Provided",

```

```
"field": "Emanifest.generator.epaSiteId"
```

```
}
```

4.2.2. Store the provided Site Information (if no other manifest errors were found)

4.3. If the Site Contact Phone number is not provided and site the is registered in RCRAInfo, the system checks if the site has a Contact Phone value registered for the site. If the Contact Phone is not registered in the system for that site, the system generates the following error:

```
{
```

```
"message": "Mandatory Field is not provided"
```

```
"field": "Emanifest.generator.contact.Phone.number"
```

```
}
```

4.4. If the Site Contact Phone number is provided and registered in RCRAInfo the system checks that the phone number is in the correct format.

4.4.1. If the provided phone number has valid format the, system will store site contact phone number into e-Manifest database

4.4.2. If the provided phone number has an incorrect format, the system will store registered in RCRAInfo Contact Phone number into e-Manifest database and generate following warning:

```
{
```

```
"message": "Invalid Field format. Does not match phone number format of 999-999-9999",
```

```
"field": "Emanifest.generator.contact.phone.number ",
```

```
"value": "number value"
```

```
}
```

4.5. If the Site Contact Phone number is provided and registered in RCRAInfo the system checks that the phone number is in the correct format.

4.5.1. If the provided phone number has valid format, the system will store site contact phone number into e-Manifest database

4.5.2. If provided phone number has incorrect format, the system generates following error:

```
{
```

```
"message": "Invalid Field format. Does not match phone number format of 999-999-9999",
```

```
"field": "Emanifest.generator.contact.phone.number ",
```

```
"value": "number value"
```

```
}
```

4.6. If emergency phone number is not provided, the service generates an error:

```
{
```

```
"message": "Mandatory Field is not provided"
```

```
"field": "Emanifest.generator.emergencyPhone.number"
```

```
}
```

4.7. If emergency phone number is provided and has an invalid format, the service generates an error:

```
{
```

```

    "message": "Provided Value is not Valid. Does not match phone number format of 999-999-9999",
    "field": "Emanifest.generator.emergencyPhone.number ",
    "value": "phone number value"
  }

```

- 4.8. If emergency phone extension is provided and has invalid format, the system generates a warning:

```

{
  "message": "Provided Value is not Valid. Does not match phone extension format of 9999999",
  "field": "Emanifest.generator.emergencyPhone.extension ",
  "value": "extension value"
}

```

- 4.9. If the provided Contact email has an invalid format, the service generates a warning:

```

{
  "message": "Provided Value will be ignored. Does not match email format",
  "field": "Emanifest.generator.contact.email",
  "value": "email value"
}

```

## 5. Update Validation

- 5.1. If the current status > Scheduled and provided status > Scheduled, Generator information cannot be updated

- 5.2. Service will check if any of provided following Generator fields are different than currently stored values:

- Site Id
- Site Name
- Site Address (all fields)
- Mailing Address (all fields)
- Contact (all fields)

- 5.3. If any of the Generator information fields are different than currently stored, the service will generate the following warning:

```

{
  "message": "Provided Value will be ignored. For statuses InTransit, Received, and ReadyForSignature the Generator information cannot be updated.",
  "field": "Emanifest.generator.epaSiteId/name/siteAddress/mailingAddress/contact",
  "value": "field value"
}

```

## Transporter Site ID and Site Information Validation

1. If submissionType is "FullElectronic" or "Hybrid", then the Transporter is valid if the Transporter is registered in RCRAInfo and the Transporter has at least one User with the e-Manifest Certifier Role and this user has ESA == Received

- 1.1. If the Transporter Site ID is not provided the service generates schema validation error
- 1.2. If the Transporter Site ID has an invalid format the service generates schema validation error
- 1.3. If the Transporter Site ID is not registered the service generates an error:

```
{
  "message": "For FullElectronic submission type registered Transporter Site Id must be provided ",
  "field": "Emanifest.designatedFacility.epaSiteId",
  "value": "Site ID value"
}
```

- 1.4. If the Transporter Site ID is registered and there are no users with the e-Manifest Certifier role for the provided Transporter, the service generates an error:

```
{
  "message": "No Users with Certifier Role found for the provided Transporter Site Id",
  "field": "Emanifest.designatedFacility.epaSiteId",
  "value": "Site ID value"
}
```

- 1.5. If the Transporter Site ID is registered and there are no users with ESA == Received the service generates an error:

```
{
  "message": "No Users which can Electronically sign found for the provided Transporter Site Id",
  "field": "Emanifest.transporter.epaSiteId",
  "value": "Site ID value"
}
```

2. If **submissionType == DataImage5Copy**, then registered and non-registered Transporters are valid

- 2.1. If the Transporter is registered in RCRAInfo as a Transporter, then only the epaSiteId must be provided. All the site information will be obtained from RCRAInfo. If other site information is provided it will be ignored with warnings.

- 2.2. If the Transporter is not registered in RCRAInfo, then the following site information must be provided:

- Site Address
- Site Name
- Contact phone

- 2.3. If Site ID AND Site Information are not provided, the system generates errors:

```
{
  "message": "Value is not provided"
  "field": "Emanifest.transporter.epaSiteId"
}
```

```

    "message": "Value is not provided"
    "field": "Emanifest.transporter.siteAddress"
  }
  {
    "message": "Mandatory field is not provided"
    "field": "Emanifest.transporter.name"
  }
  {
    "message": "Value is not provided"
    "field": "Emanifest.transporter.contact.phone.number"
  }
}

```

2.4. If Site ID is provided AND Site Information is not provided, the system performs the following steps:

2.4.1. Validate Site ID format

2.4.2. If Site ID is not valid, the service generates an error:

```

{
  "message": "Provided Value is not Valid. Does not match format of: Two Letter Activity
Location Code + Up to 10 alphanumeric characters",
  "field": "Emanifest.transporter.epaSiteId "
}

```

2.4.3. If the Site ID is valid, the system will search in RCRAInfo by Site ID.

2.4.4. If the site is found, the system obtains Site Information from RCRAInfo and stores it into the Manifest.

2.4.5. If site is not found, the service generates an error:

```

{
  "message": "Site with the provided EPA Site ID is not registered",
  "field": "Emanifest.transporter.epaSiteId",
}

```

2.5. If the Site ID is provided AND Site Information is provided, the site information will be ignored with warnings and the system will perform the following steps:

2.5.1. Validate provided Site ID

2.5.2. If Site ID is not valid, the service generates error:

```

{
  "message": "Provided value is not valid. Does not match format of: Two Letter
Activity Location Code + Up to 10 alphanumeric characters",
  "field": "Emanifest.transporter.epaSiteId"
  "value": "EPA Site ID value"
}

```

2.5.3. If Site ID is valid, check if site is registered

2.5.4. If the site is not registered, the service generates a warning:

```

{
  "message": "Site with provided epa site id is not registered",
}

```

```

        "field": "Emanifest.transporter.epaSiteId",
        "value": "Site ID value"
    }

```

2.6. If Site ID is not provided AND Site Information is provided, the system will perform the following steps:

2.6.1. Validate the provided Site Information

2.6.2. Validate the Site Name

2.6.3. If the Site Name is not valid, the service generates an error:

```

{
    "message": "Provided Value is not Valid",
    "field": "Emanifest.transporter.name",
    "value": "site name value"
}

```

2.6.4. Validate the mandatory site address fields:

- address1 (50 Character maximum)
- city (25 Character maximum)
- state.code (2 Character State Abbreviation)
- zip (14 Character maximum)

2.6.5. If any of the mandatory location address fields are not provided or invalid, the service generates an error:

```

{
    "message": "Value is not Provided/Provided Value is not Valid",
    "field": "Emanifest.transporter.siteAddress.address1/city/state/zip",
    "value": "address1/city/state/zip value"
}

```

### Paper Signature Info Validation

1. If submissionType is FullElectronic or Image and any (Generator, Transporter(s), TSDf) of the paperSignature Information (printedName or signatureDate) is provided the service generates a warning:

```

{
    "message": "Provided Field will be ignored. Paper Signature Information is not applicable for FullElectronic/Image submission type",
    "field": "Emanifest.generator/transporter/designatedFacility.paperSignatureInfo",
    "value": "printedName or signatureDate value"
}

```

2. If submissionType == Hybrid the following applies

2.1. If Transporter.order is not equal 1 and transporter paperSignature Information (printedName or signatureDate) is provided the service generates a warning:

```

{
    "message": "Provided Field will be ignored. For any other than 1st Transporter Paper Signature Information is not applicable for Hybrid submission type",
}

```



```
"field": "Emanifest.transporter.paperSignatureInfo",
"value": "printedName or signatureDate value"
}
```

- 2.2. If TSDF paperSignature Information (printedName or signatureDate) is provided the service generates a warning:

```
{
  "message": "Provided Field will be ignored. TSDF Paper Signature Information is not
  applicable for Hybrid submission type",
  "field": "Emanifest.designatedFacility.paperSignatureInfo",
  "value": "printedName or signatureDate value"
}
```

- 2.3. If Generator/1<sup>st</sup> Transporter Printed Name or Signature Date are not provided the service generates a warning:

```
{
  "message": "Field is not Provided",
  "field":
    "Emanifest.generator/transporter.paperSignatureInfo.printedName/signatureDate"
}
```

- 2.4. If the Generator/1<sup>st</sup> Transporter Printed Names are provided, validate the format. If an invalid format is provided, the service generates an error:

```
{
  "message": "Invalid Field format. Text not exceeding 255 characters expected",
  "field": "Emanifest.generator/transporter.paperSignatureInfo.printedName",
  "value": "value"
}
```

- 2.5. If the Generator/1<sup>st</sup> Transporter Signature Dates are not provided the service generates a warning:

```
{
  "message": "Field is not Provided",
  "field": "Emanifest.generator/transporter.paperSignatureInfo.signatureDate"
}
```

- 2.6. If the Generator/1<sup>st</sup> Transporter Signature Dates are provided the service validates the Signature Dates consecutiveness

- 2.6.1. Check if Generator signatureDate <= First Transporter signature date (Transporter.order will be used to determine consecutiveness of transporters). If not, the service generates an error:
- ```
{
  "message": "Invalid Value. Generator Signature Date must be earlier or the same
  date as the first Transporter Signature Date",
  "field": "Emanifest.generator.paperSignatureInfo.signatureDate",
  "value": "Generator Signature Date value"
}
```

```
{
  "message": "Invalid Value. Generator Signature Date must be earlier or the same
date as the first Transporter Signature Date",
  "field": "Emanifest.transporter.paperSignatureInfo.signatureDate",
  "value": "Transporter Signature Date value"
}
```

2.6.2. If any of the Signature Dates > current date the service generates an error:

```
{
  "message": "Invalid Value. Generator/Transporter Signature Date must be the same
or earlier than Current Date",
  "field": "Emanifest.generator/transporter.paperSignatureInfo.signatureDate",
  "value": "Signature Date value"
}
```

2.7. If `Emanifest.rejectionInfo.rejectionType == "FullReject"` and  
`Emanifest.rejectionInfo.transporterOnSite == true` and  
`Emanifest.rejectionInfo.alternateDesignatedFacilityType == "Generator"` following applies

2.7.1. If `Emanifest.rejectionInfo.alternateDesignatedFacilityType == "Generator"` following applies

2.7.1.1. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate` is not provided the service generates following error:

```
{
  "message": "Mandatory field is not provided",
  "field":
    "Emanifest.rejectionInfo.generator.PaperSignatureInfo.signatureDate"
}
```

2.7.1.2. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.printedName` is not provided the service generates following error:

```
{
  "message": "Mandatory field is not provided",
  "field": "Emanifest.rejectionInfo.generator.paperSignatureInfo.
printedName"
}
```

2.7.1.3. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate` is invalid the service generates following error:

```
{
  "message": "Invalid Field format",
  "field":
    "Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate",
  "value": "value"
}
```

2.7.1.4. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate < Emanifest.designatedFacility.electronicSignatureInfo.signatureDate` then the service generates following error:

```
{
  "message": "Invalid Value. Generator Signature Date must be the same or later than Designated Facility Signature Date",
  "field": "Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate",
  "value": "value"
}
{
  "message": "Invalid Value. Generator Signature Date must be the same or later than Designated Facility Signature Date",
  "field": "Emanifest.designatedFacility.electronicSignatureInfo.signatureDate",
  "value": "Designated Facility Signature Date value"
}
```

3. If `submissionType == DataImage5Copy`, the system performs the following applies:

3.1. If Generator/Transporter/TSDf Printed Names are not provided the service generates a warning:

```
{
  "message": "Field is not Provided",
  "field":
    "Emanifest.generator/transporter/designatedFacility.paperSignatureInfo.printedName"
}
```

3.2. If the Generator/Transporter/TSDf Printed Names are provided, validate the format. If an invalid format is provided, the service generates an error:

```
{
  "message": "Invalid Field format. Text not exceeding 255 characters expected",
  "field":
    "Emanifest.generator/transporter/designatedFacility.paperSignatureInfo.printedName",
  "value": "value"
}
```

3.3. If the Generator/Transporter/TSDf Signature Dates are not provided, the service generates an error (for this example: TSDf is used)

```
{
  "message": "Mandatory Field is not Provided",
  "field":
    "Emanifest.generator/transporter/designatedFacility.paperSignatureInfo.signatureDate",
}
```

3.4. If Generator/Transporter/TSDf Signature Dates are provided, validate format. If format is invalid the service generates the following error (for this example: TSDf is used)

```
{
```

```

    "message": "Invalid Field format",
    "field": "Emanifest.generator/transporter/designatedFacility.paperSignatureInfo.
signatureDate",
    "value": "value"
}

```

3.5. If there is a Generator signature and the first Transporter Signature is not provided the service generates an error:

```

{
    "message": "Mandatory Field is not Provided. First Transporter Signature must be
provided when Generator signature is provided",
    "field": "Emanifest.transporter.paperSignatureInfo.signatureDate"
}

```

4. The system will perform the following validation for consecutiveness of Signature Dates for **DataImage5Copy**:

4.1. If the signature dates for Generator, all Transporters, and TSDF are provided and have a valid format, and more than one transporter is provided, the system performs the following steps:

4.1.1. Check if Generator signatureDate <= First Transporter signature date (Transporter.order will be used to determine consecutiveness of transporters). If not, the service generates an error:

```

{
    "message": "Invalid Value. Generator Signature Date must be earlier or the same
date as the first Transporter Signature Date",
    "field": "Emanifest.generator.paperSignatureInfo.signatureDate",
    "value": "Generator Signature Date value"
}
{
    "message": "Invalid Value. Generator Signature Date must be earlier or the same
date as the first Transporter Signature Date",
    "field": "Emanifest.transporter.paperSignatureInfo.signatureDate",
    "value": "Transporter Signature Date value"
}

```

4.1.2. Check the order of the Transporters signature dates. If there is more than one Transporter provided, check if signature dates of transporters are in order or the same. For example, if there are two transporters and the signature date of transporter which has order = 2 is earlier than signature date of transporter with order = 1, the service generates an error (example):

```

{
    "message": "Invalid Value. First Transporter Signature Date must be the same or
earlier than Second Transporter Signature Date",

```

```

    "field": "Emanifest.transporter.paperSignatureInfo.signatureDate",
    "value": "First Transporter Signature Date value"
  }
  {
    "message": "Invalid Value. First Transporter Signature Date must be the same or
    earlier than Second Transporter Signature Date",
    "field": "Emanifest.transporter.paperSignatureInfo.signatureDate",
    "value": "Second Transporter Signature Date value"
  }

```

- 4.1.3. Check if the TSDF signature date >= Last Transporter signature date. If not, the service generates an error:

```

  {
    "message": "Invalid Value. Last Transporter Signature Date must be the same or
    earlier than Designated Facility Signature Date",
    "field": "Emanifest.transporter.paperSignatureInfo.signatureDate",
    "value": "Last Transporter Signature Date value"
  }
  {
    "message": "Invalid Value. Last Transporter Signature Date must be the same or
    earlier than Designated Facility Signature Date",
    "field": "Emanifest.designatedFacility.paperSignatureInfo.signatureDate",
    "value": "Designated Facility Signature Date value"
  }

```

- 4.1.4. If any of the Signature Dates > current date the service generates an error:

```

  {
    "message": "Invalid Value. Generator/Transporter/TSDF Signature Date must be the
    same or earlier than Current Date",
    "field":
    "Emanifest.generator/transporter/designatedFacility.paperSignatureInfo.signatur
    eDate",
    "value": "Signature Date value"
  }

```

- 4.2. If Emanifest.rejectionInfo.rejectionType = "FullReject" and  
Emanifest.rejectionInfo.transporterOnsite == true following applies

- 4.2.1. If Emanifest.rejectionInfo.alternateDesignatedFacilityType == "TsdF" following applies

- 4.2.1.1. If

Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo.signatur  
eDate is not provided the service generates following error:

```

  {
    "message": "Mandatory field is not provided",

```

```

        "field":
        "Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo.si
        gnatureDate"
    }
4.2.1.2. If
    Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo.printed
    Name is not provided the service generates following error:
    {
        "message": "Mandatory field is not provided",
        "field":
        "Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo.
        printedName"
    }
4.2.1.3. If
    Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo.signatur
    eDate is invalid the service generates following error:

    {
        "message": "Invalid Field format",
        "field":
        "Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo
        .signatureDate",
        "value": "value"
    }
4.2.1.4. If
    Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo.signatur
    eDate < Emanifest.designatedFacility.paperSignatureInfo.signatureDate then
    the service generates following error:

    {
        "message": "Invalid Value. Alternate Designated Facility Signature Date must be the
        same or later than Designated Facility Signature Date",
        "field": "
        Emanifest.rejectionInfo.alternateDesignatedFacility.paperSignatureInfo.signatureDa
        te",
        "value": "value"
    }
    {
        "message": "Invalid Value. Alternate Designated Facility Signature Date must be the
        same or later than Designated Facility Signature Date",
        "field": "Emanifest.designatedFacility.paperSignatureInfo.signatureDate",
        "value": "Designated Facility Signature Date value"
    }

```

```
}
```

4.2.2. If `Emanifest.rejectionInfo.alternateDesignatedFacilityType == "Generator"` following applies

4.2.2.1. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate` is not provided the service generates following error:

```
{
    "message": "Mandatory field is not provided",
    "field":
        "Emanifest.rejectionInfo.generator.PaperSignatureInfo.signatureDate"
}
```

4.2.2.2. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.printedName` is not provided the service generates following error:

```
{
    "message": "Mandatory field is not provided",
    "field": "Emanifest.rejectionInfo. generator.paperSignatureInfo.
printedName"
}
```

4.2.2.3. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate` is invalid the service generates following error:

```
{
    "message": "Invalid Field format",
    "field":
        "Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate",
    "value": "value"
}
```

4.2.2.4. If `Emanifest.rejectionInfo.generator.paperSignatureInfo.signatureDate < Emanifest.designatedFacility.paperSignatureInfo.signatureDate` then the service generates following error:

```
{
    "message": "Invalid Value. Generator Signature Date must be the same or later than
Designated Facility Signature Date",
    "field": "Emanifest.rejectionInfo. generator.paperSignatureInfo.signatureDate",
    "value": "value"
}
{
    "message": "Invalid Value. Generator Signature Date must be the same or later than
Designated Facility Signature Date",
    "field": "Emanifest.designatedFacility.paperSignatureInfo.signatureDate",
    "value": "Designated Facility Signature Date value"
}
```

## Import Information Validation

1. If Emanifest.import is not provided, the service will generate the following error:

```
{  
  "message": "Mandatory field is not provided",  
  "field": "Emanifest. import"  
}
```

2. If Emanifest.import is true following applies

- 2.1. If any of the following ImportInfo fields are not provided:

- ImportGenerator.name
- ImportGenerator.city
- ImportGenerator.address
- ImportGenerator.country.code
- ImportPortInfo.city
- ImportPortInfo.state.code

then the service generates following error:

```
{  
  "message": "Mandatory Field is not Provided.",  
  "field":  
    "Emanifest.importInfo.ImportGenerator.name/city/address/country/ImportPortInfo.city/  
    ImportPortInfo.state"  
}
```

- 2.2. If any of the following ImportInfo fields are not provided:

- ImportGenerator.postalCode
- ImportGenerator.province

then the service generates following warning:

```
{  
  "message": "Field is not Provided.",  
  "field": "Emanifest.importInfo.ImportGenerator.postalCode/province"  
}
```

- 2.3. If for any Waste the Waste.additionalInfo.consentNumber is not provided then the service generates following warning:

```
{  
  "message": "Field is not Provided.",  
  "field": "Emanifest.Waste.additionalInfo.consentNumber"  
}
```

- 2.4. If Waste.additionalInfo.consentNumber is invalid then the service generates following warning:

```
{  
  "message": "Invalid Value Provided. Expected format 12 digits. 6 digits 'I' and then 5  
  digits.",  
  "field": "Emanifest.additionalInfo.consentNumber"  
  "value": "consentNumber value"
```



```
}
```

2.5. If ImportGenerator.name is invalid then the service generates following error:

```
{  
  "message": "Invalid Value Provided. Expected length not to exceed 80 characters",  
  "field": "Emanifest.importInfo.ImportGenerator.name"  
  "value": "name value"  
}
```

2.6. If ImportGenerator.address is invalid then the service generates following error:

```
{  
  "message": "Invalid Value Provided. Expected length not to exceed 50 characters",  
  "field": "Emanifest.importInfo.ImportGenerator.address"  
  "value": "address value"  
}
```

2.7. If ImportGenerator.city is invalid then the service generates following error:

```
{  
  "message": "Invalid Value Provided. Expected length not to exceed 25 characters",  
  "field": "Emanifest.importInfo.ImportGenerator.city"  
  "value": "city value"  
}
```

2.8. If ImportGenerator.province is invalid then the service generates following warning:

```
{  
  "message": "Invalid Value Provided. Expected length not to exceed 50 characters",  
  "field": "Emanifest.importInfo.ImportGenerator.province"  
  "value": "province value"  
}
```

2.9. If ImportGenerator.postalCode is invalid then the service generates following warning:

```
{  
  "message": "Invalid Value Provided. Expected length not to exceed 50 characters",  
  "field": "Emanifest.importInfo.ImportGenerator.postalCode"  
  "value": "province value"  
}
```

2.10. If ImportGenerator.country.Code is invalid then the service generates following warning:

```
{  
  "message": "Provided Value was not Found",  
  "field": "Emanifest.importInfo.ImportGenerator.country.code"  
  "value": "code value"  
}
```

2.11. If portOfEntry.cityPort is invalid then the service generates following warning:

```
{  
  "message": "Invalid Value Provided. Expected length not to exceed 25 characters",  
  "field": "Emanifest.importInfo. portOfEntry.cityPort"  
  "value": "portOfEntry.cityPort value"
```

```
}
```

2.12. If portOfEntry.state.code is invalid then the service generates following warning:

```
{  
  "message": "Provided Value was not Found",  
  "field": "Emanifest.importInfo. portOfEntry.state.code"  
  "value": "code value"  
}
```

3. If Emanifest.import == false and any of the following fields are provided:

- ImportGenerator.name
- ImportGenerator.city
- ImportGenerator.address
- ImportGenerator.country.code
- ImportGenerator.postalCode
- ImportGenerator.province
- portOfEntry.cityPort
- portOfEntry.state.code
- Waste.additinoalInfo.consentNumber

Then service generates following warning:

```
{  
  "message": "Provided Value will be ignored. The field is non applicable if Manifest is not  
imported"  
  "field": "{provided field}",  
  "value": "field value"  
}
```

### New Manifest Information Validation

1. If the New Manifest is created for shipping the waste to another TSDF or back to the Generator as a "result" of the rejection or residue of the Original Manifest then following applies
  - 1.1. If Emanifest.status >= "ReadyForSignature" and Emanifest.containsPreviousRejectOrResidue is not provided then the service generates following error:

```
{  
  "message": "Mandatory Field is not Provided.",  
  "field": "Emanifest.containsPreviousRejectOrResidue"  
}
```

- 1.2. If Emanifest.containsPreviousRejectOrResidue == true and Emanifest.additionalInfo.newManifestDestination is not provided then the service generates following error:

```
{  
  "message": "Mandatory Field is not Provided.",  
  "field": "Emanifest.additionalInfo.newManifestDestination"  
}
```

- 1.3. If `Emanifest.containsPreviousRejectOrResidue == true` and `Emanifest.additionalInfo.originalManifestTrackingNumbers` are not valid the service generates following warning:

```
{
  "message": "Invalid value provided. Provided Field will be ignored.",
  "field": "Emanifest.additionalInfo.originalManifestTrackingNumbers",
  "value": "provided value"
}
```

- 1.4. If `Emanifest.additionalInfo.newManifestDestination == "OriginalGenerator"` and `Emanifest.rejection == true` the service generates following errors:

```
{
  "message": "New Manifest cannot be rejected if shipped back to the original Generator",
  "field": "Emanifest.additionalInfo.newManifestDestination",
  "value": "newManifestDestination value"
}
{
  "message": "New Manifest cannot be rejected if shipped back to the original Generator",
  "field": "Emanifest.rejection",
  "value": "rejection value"
}
```

### Rejection Information Validation

1. If an Original Manifest is rejected then `Emanifest.rejection` shall be specified as `"true"`
2. `Emanifest.rejection` and `RejectionInfo` shall be provided at status equal to `"ReadyForSignature"`.  
If provided for earlier statuses these fields will be ignored.

- 2.1. If `Emanifest.status < "ReadyForSignature"` and either `Emanifest.rejection` or `RejectionInfo` fields are provided then the service generates following warning:

```
{
  "message": "Provided Field will be ignored. Rejection information shall be provided at ReadyForSignature status",
  "field": "Emanifest.rejection/Emanifest.rejectionInfo",
  "value": "provided value"
}
```

3. If `submissionStatus >= "ReadyForSignature"` and `Emanifest.rejection` is not provided then the service generates following error:

```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.rejection"
}
```

4. `Emanifest.containsPreviousRejectOrResidue` is not provided then the service generates following error:
 

```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.containsPreviousRejectOrResidue"
}
```
5. If `Emanifest.rejection == true` then following applies
  - 5.1. If `RejectionInfo.rejectionComments` is not provided then the service generates following error:
 

```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.RejectionInfo. rejectionComments"
}
```
  - 5.2. If `RejectionInfo.rejectionComments` are not valid then the service generates following error:
 

```
{
  "message": "Invalid Value Provided. Expected length not to exceed 255 characters",
  "field": "Emanifest.RejectionInfo.rejectionComments"
  "vare": "rejectionComments value"
}
```
  - 5.3. If `RejectionInfo.transporterOnSite` is not provided then the service generates following error:
 

```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.RejectionInfo.transporterOnSite"
}
```
  - 5.4. If `RejectionInfo.transporterOnSite == true` following applies
    - 5.4.1. If `RejectionInfo.rejectionType == "PartialReject"` then the service set `RejectionInfo.rejectionType = "FullReject"` and generates following warning:
 

```
{
  "message": "Provided value will be Ignored. If Transporter is On Site rejection type must be "FullReject",
  "field": "Emanifest.RejectionInfo.rejectionType",
  "value": "rejectionType value"
}
```
    - 5.4.2. `Emanifest` sets `rejectionInfo.rejectionType` as `"FullReject"` in `Emanifest` database
    - 5.4.3. If `RejectionInfo.newManifestTrackingNumber` is provided then then the service generates following warning:
 

```
{
  "message": "Provided Field will be Ignored. New Manifest(s) are not created if Transporter is On Site",
  "field": "Emanifest.RejectionInfo. newManifestTrackingNumber"
}
```

- ```

    }

```
- 5.4.4. If RejectionInfo.alternateDesignatedFacilityType is not provided then the service generates following error:
- ```

{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.RejectionInfo.alternateDesignatedFacilityType"
}

```
- 5.4.5. If RejectionInfo.alternateDesignatedFacilityType == "Generator" and submissionType == "DataImage5Copy" then following applies
- 5.4.5.1. If generatorPaperSignatureInfo.printedName is not provided then the service generates following error:
- ```

{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.RejectionInfo.generatorPaperSignatureInfo.printedName "
}

```
- 5.4.5.2. If generatorPaperSignatureInfo.printedName has invalid format then the service generates following error:
- ```

{
  "message": "Invalid Field Format. Text not exceeding 255 characters expected ",
  "field": "Emanifest.RejectionInfo.generatorPaperSignatureInfo.printedName ",
  "value": "printedName value"
}

```
- 5.4.5.3. If generatorPaperSignatureInfo.signatureDate is not provided then the service generates following error:
- ```

{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.RejectionInfo.generatorPaperSignatureInfo.signatureDate "
}

```
- 5.4.5.4. If generatorPaperSignatureInfo.signatureDate < Emanifest.designatedFacility.paperSignatureInfo.signatureDate then the service generates following error:
- ```

{
  "message": "Invalid Value Provided. Generator signature date (signing as alternate Facility) shall be on or later than TSDF signature",
  "field": "Emanifest.RejectionInfo.generatorPaperSignatureInfo.signatureDate ",
  "value": "signatureDate value"
}

```
- 5.4.5.5. If generatorPaperSignatureInfo.signatureDate > current Date then the service generates following error:
- ```

{
  "message": "Invalid Value Provided. Generator signature date (signing as alternate Facility) cannot be in future",

```

```
"field": "Emanifest.RejectionInfo.generatorPaperSignatureInfo.signatureDate ",
"value": "signatureDate value"
}
```

5.4.6. If RejectionInfo.alternateDesignatedFacilityType == "TsdF" following applies

5.4.6.1. If RejectionInfo.alternateDesignatedFacility.epaSiteId is not provided then the service generates following error:

```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.RejectionInfo. alternateDesignatedFacility.epaSiteId"
}
```

5.4.6.2. If RejectionInfo.alternateDesignatedFacility.epaSiteId has invalid format then the service generates following error:

```
{
  "message": "Provided Value is not Valid. Does not match format of: Two Letter Activity Location Code + Up to 10 alphanumeric characters "
  "field": "Emanifest.RejectionInfo.alternateDesignatedFacility.epaSiteId",
  "value": "epaSiteId value"
}
```

5.4.6.3. If epaSiteId is valid, the system will search RCRAInfo by Site ID. If the Site ID is found the system will obtain the Site information from RCRAInfo and store it into the Manifest.

5.4.6.4. If the epaSiteId is not found, the service generates an error:

```
{
  "message": "Provided Value is not Found",
  "field": "Emanifest.RejectionInfo.alternateDesignatedFacility.epaSiteId",
  "value": "epaSiteId value"
}
```

5.5. If RejectionInfo.transporterOnSite == false following applies

5.5.1. If RejectionInfo.rejectionType is not provided then the service generates following error:

```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.RejectionInfo.rejectionType"
}
```

5.5.2. If RejectionInfo.newManifestTrackingNumber(s) is provided and not valid the service generates following error:

```
{
  "message": "Invalid Field Format",
  "field": "Emanifest.RejectionInfo.newManifestTrackingNumber",
  "value": "mtn value"
}
```

- 5.5.3. If manifestTrackingNumber has a valid format, the service checks if the provided manifestTrackingNumber suffix is valid. If the suffix is invalid the service generates an error:

```
{
  "message": "Invalid Manifest Tracking Number Suffix is Provided",
  "field": "Emanifest.RejectionInfo.newManifestTrackingNumber ",
  "value": "manifestTrackingNumber value"
}
```

- 5.5.4. If any of the following fields are provided:

- RejectionInfo.alternateDesignatedFacilityType
- RejectionInfo.generatorPaperSignature
- RejectionInfo.alternateDesignatedFacility

the service generates following warning:

```
{
  "message": "Provided Field will be Ignored. alternateDesignatedFacilityType/
generatorPaperSignature/ alternateDesignatedFacility not applicable if Transporter
is Off Site",
  "field": "Emanifest.RejectionInfo. alternateDesignatedFacilityType/
generatorPaperSignature/ alternateDesignatedFacility",
  "value": "alternateDesignatedFacilityType/ generatorPaperSignature/
alternateDesignatedFacility value"
}
```

6. If Emanifest.rejection == false and If any of the following RejectionInfo fields are provided:

- RejectionInfo.rejectionType
- RejectionInfo.transporterOnSite
- RejectionInfo.alternateDesignatedFacilityType
- RejectionInfo.generatorPaperSignature
- RejectionInfo.alternateDesignatedFacility

the service generates following warning:

```
{
  "message": "Provided Field will be Ignored. {Field} not applicable if Manifest is not Rejected",
  "field": "Emanifest.RejectionInfo. rejectionType /transporterOnSite/
alternateDesignatedFacilityType/ generatorPaperSignature/ alternateDesignatedFacility",
  "value": "rejectionType /transporterOnSite/alternateDesignatedFacilityType/
generatorPaperSignature/ alternateDesignatedFacility value"
}
```

### Discrepancy fields validation

1. If status >= ReadyForSignature and DiscrepancyResidueInfo.wasteQuantity is not provided then service generates following error:

```
{
  "message": "Mandatory Field is not Provided.",
}
```

- ```

    "field": "Emanifest.discrepancyResidueInfo.wasteQuantity"
  }

```
2. If status >= ReadyForSignature and DiscrepancyResidueInfo.wasteType is not provided then service generates following error:
 

```

{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.discrepancyResidueInfo.wasteType"
}

```
  3. If DiscrepancyResidueInfo.wasteQuantity == true or DiscrepancyResidueInfo.wasteType == true and DiscrepancyResidueInfo.discrepancyComments is not provided then the service generates following error:
 

```

{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.discrepancyResidueInfo.discrepancyComments"
}

```
  4. If DiscrepancyResidueInfo.discrepancyComments has invalid format then the service generates following error:
 

```

{
  "message": "Invalid Value Provided. Expected length not to exceed 255 characters",
  "field": "Emanifest.discrepancyResidueInfo.discrepancyComments"
  "value": "discrepancyComments value"
}

```
  5. If DiscrepancyResidueInfo.wasteQuantity == false and DiscrepancyResidueInfo.wasteType == false and DiscrepancyResidueInfo.discrepancyComments is provided then the service generates following warning:
 

```

{
  "message": "Provided Field will be Ignored.",
  "field": "Emanifest.discrepancyResidueInfo.discrepancyComments"
  "value": "discrepancyComments value"
}

```
  6. If DiscrepancyResidueInfo.wasteQuantity == true or DiscrepancyResidueInfo.wasteType == true then the service sets Emanifest.discrepancy = true
  7. If DiscrepancyResidueInfo.wasteQuantity == false and DiscrepancyResidueInfo.wasteType == false then the service sets Emanifest.discrepancy = false

### Residue Fields Validation

1. If status >= ReadyForSignature and DiscrepancyResidueInfo.residue is not provided then service generates following error:
 

```

{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.discrepancyResidueInfo.residue"
}

```



- ```
}
2. If DiscrepancyResidueInfo.residue == true and DiscrepancyResidueInfo.residueComments is
   provided the service validates if DiscrepancyResidueInfo.residueComments has valid format. If
   DiscrepancyResidueInfo.residueComments has invalid format the service generates following
   error:
   {
       "message": "Invalid Value Provided. Expected length not to exceed 255 characters",
       "field": "Emanifest.discrepancyResidueInfo.residueComments"
       "value": "residueComments value"
   }
3. If DiscrepancyResidueInfo.residue == false and DiscrepancyResidueInfo.residueComments is
   provided then the service generates following warning:
   {
       "message": "Provided Field will be Ignored.",
       "field": "Emanifest.discrepancyResidueInfo.residueComments"
       "value": "residueComments value"
   }
4. If DiscrepancyResidueInfo.residue == true then the service sets Emanifest.residue = true
5. If DiscrepancyResidueInfo.residue == true then the Emanifest.
   residueNewManifestTrackingNumber(s) can be provided.
5.1. If no Emanifest.residueNewManifestTrackingNumber(s) is provided the service generates
   following warning:
   {
       "message": "One of the Waste has residue. Manifest Tracking Number(s) for shipping waste
       to another Facility(ies) is not provided",
       "field": "Emanifest.discrepancyResidueInfo.discrepancyComments"
       "value": "discrepancyComments value"
   }
5.2. If Emanifest.residueNewManifestTrackingNumber(s) are provided then the service validates
   the format of residueNewManifestTrackingNumber(s). Emanifest.manifestTrackingNumber
   is provided and it has an invalid format, the service generates a warning:
   {
       "message": "Field will be ignored. Invalid Field Format. 9 digits followed by 3 upper
       case letters is expected",
       "field": "Emanifest.residueNewManifestTrackingNumbers ",
       "value": "residueNewManifestTrackingNumbers value"
   }
5.3. If the residueNewManifestTrackingNumber has a valid format, the service checks if the
   provided residueNewManifestTrackingNumber suffix is valid. If the suffix is invalid, the
   service generates an error:
   {
       "message": "Field will be ignored. Invalid Manifest Tracking Number Suffix is Provided",
```

- ```

    "field": "Emanifest.residueNewManifestTrackingNumber",
    "value": "residueNewManifestTrackingNumber value"
  }

```
6. If DiscrepancyResidueInfo.residue == false then following applies
    - 6.1. The service sets Emanifest.residue = false
    - 6.2. If at least one Emanifest.residueNewManifestTrackingNumber is provided the service generates a warning:
 

```

{
  "message": "Field will be ignored. Manifest has no Residue. New Manifest Tracking Number(s) shall not be provided ",
  "field": "Emanifest.residueNewManifestTrackingNumbers ",
  "value": "residueNewManifestTrackingNumbers value"
}

```

### Manifest Additional Information Validation

1. handlingInstructions Validation.
  - 1.1. If handlingInstructions are invalid, the service generates a warning:
 

```

{
  "message": "Invalid Value. Text can be no longer than 4000 characters",
  "field": "Emanifest.additionalInfo.handlingInstructions",
  "value": "handlingInstructions value"
}

```
2. originalManifestTrackingNumber Validation
  - 2.1. If Original Manifest Tracking Number is invalid, the service generates a warning:
 

```

{
  "message": "Invalid Format. Tracking Number must be 9 digits followed by 3 upper case letters",
  "field": "Emanifest.additionalInfo.originalManifestTrackingNumber",
  "value": "Original Manifest Tracking Number value"
}

```
3. comments Validation
  - 3.1. handlerId validation
    - 3.1.1. If the handler Id is not provided, then the comment will be ignored with a warning:
 

```

{
  "message": "Comment will be ignored. Field is not Provided",
  "field": "Emanifest.additionalInfo.comments.handlerId"
}

```
    - 3.1.2. If the handlerId has an invalid format provided, then the comment will be ignored with a warning:
 

```

{
  "message": "Comment will be ignored, due to an invalid Site ID format being submitted",
  "field": "Emanifest.additionalInfo.comments.handlerId"
}

```

```
"value": "Handler ID value"
}
```

- 3.1.3. If the provided handlerId does not match the Generator or TSDF or any of Transporter Site IDs in the manifest, the comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Provided handlerId does not match Generator or TSDF or any of Transporter Site IDs",
  "field": "Emanifest.additionalInfo.comments.handlerId"
  "value": "Handler ID value"
}
```

### 3.2. label validation

- 3.2.1. If the label element is not provided, then comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Field is not Provided",
  "field": "Emanifest.additionalInfo.comments.label"
}
```

- 3.2.2. If the label element has an invalid format provided, then comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Invalid Value. Text can be no longer than 250 characters.",
  "field": "Emanifest.additionalInfo.comments.label"
  "value": "Label value"
}
```

### 3.3. description validation

- 3.3.1. If the description element is not provided, then the comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Field is not Provided",
  "field": "Emanifest.additionalInfo.comments.description"
}
```

- 3.3.2. If the description element has an invalid format, the provided comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Invalid Value. Text can be no longer than 250 characters.",
  "field": "Emanifest.additionalInfo.comments.description"
  "value": "Description value"
}
```

## 4. newManifestDestination validation.

- 4.1. If `Emanifest.containsPreviousRejectOrResidue == true` and `newManifestDestination` is not provided then the service generates following error:
- ```
{
  "message": "Mandatory Field is not Provided. If the Manifest contains previous Rejection or Residue then newManifestDestination shall be provided",
  "field": "Emanifest.additionalInfo. newManifestDestination"
}
```
- 4.2. If `Emanifest.containsPreviousRejectOrResidue == false` and `newManifestDestination` is provided then the service generates following warning:
- ```
{
  "message": "newManifestDestination will be ignored. If the Manifest does not contain previous Rejection or Residue then newManifestDestination shall not be provided",
  "field": "Emanifest.additionalInfo. newManifestDestination",
  "value" : "newManifestDestination value"
}
```
5. If `consentNumber` is provided then the service generates following warning:
- ```
{
  "message": "consentNumber will be ignored. It shall be provided only for the Waste additionalInfo",
  "field": "Emanifest.additionalInfo.consentNumber",
  "value" : "consentNumber value"
}
```

## Waste Validation

1. If the waste element is not provided for any status  $\geq$  Scheduled, then the service generates an error:
- ```
{
  "message": "Mandatory Field is not Provided. At least one Waste must be provided for Scheduled, InTransit, Received, ReadyForSignature Status",
  "field": "Emanifest.waste",
}
```
2. If the waste element is provided, the system will perform the following steps:
- 2.1. If `waste.dotHazardous` is not provided, the service generates an error:
- ```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.waste.dotHazardous"
}
```
- 2.2. If `waste.epaWaste` is not provided, the service generates an error:
- ```
{
  "message": "Mandatory Field is not Provided.",
  "field": "Emanifest.waste.epaWaste"
}
```

}

2.3. If waste.pcb is not provided, the service generates an error:

{

“message”: “Mandatory Field is not Provided.”,

“field”: “Emanifest.waste.pcb”

}

2.4. If waste.pcb is true then , the system will validate the pcbs. See details in PCB Information Validation section

2.5. If waste.br is not provided the service generates an error:

{

“message”: “Mandatory Field is not Provided.”,

“field”: “Emanifest.waste.br”

}

2.6. If the waste.br is true, the system will validate the brInfo. See details in Biennial Report Information Validation section

2.7. Validate dotInformation/wasteDescription.

2.7.1. If the element dotHazardous is true, the waste must contain dotInformation

2.7.2. If the element dotHazardous is false the waste must contain the wasteDescription.

2.7.3. If the element dotHazardous is true and dotInformation is not provided the service generates an error:

{

“message”: “Mandatory Field is not Provided.”,

“field”: “Emanifest.waste.dotInformation”

}

2.7.4. If the element dotHazardous is true and dotInformation is provided the service will validate the dotInformation fields. See section DOT Information fields validation for details

2.7.5. If the element dotHazardous is false and dotInformation is provided the service generates a warning:

{

“message”: “For non hazardous Waste Dot Information will be ignored. ”,

“field”: “Emanifest.waste.dotInformation”

}

2.7.6. If the element dotHazardous is false and the element wasteDescription is not provided, then the service generates an error:

{

“message”: “Mandatory Field is not Provided.”,

“field”: “Emanifest.waste.wasteDescription”

}

2.7.7. If the element dotHazardous is false and the element wasteDescription is provided, then the service validates wasteDescription. If wasteDescription is not valid the service generates an error:

- ```

{
  "message": "Invalid Field format.Value of no longer than 500 characters is
expected",
  "field": "Emanifest.waste.wasteDescription",
  "value": "wasteDescription value"
}

```
- 2.7.8. If the element dotHazardous is true and the element wasteDescription is provided, the service generates a warning:
- ```

{
  "message": "For hazardous Waste, wasteDescription will be ignored.",
  "field": "Emanifest.waste.wasteDescription"
}

```
- 2.8. Validate Management Method Code. See section Management Method Code Validation for details.
- 2.9. Validate Waste.consentNumber.
- 2.9.1. If Emanifest.import is equal true Waste.consentNumber can be provided at any status. See details of Waste.consentNumber in section Import Information Validation\_step 2.3
- 2.9.2. If Emanifest.import is equal false and Waste.consentNumber shall not be provided. See details of Waste.consentNumber in section Import Information Validation\_step 3.0
- 2.10. Validate DOT Information. See section DOT Information fields validation for details.
- 2.11. Validate Containers and Quantity. See section Containers and quantity fields validation for details
- 2.12. Validate Hazardous Waste Codes. See section Hazardous Waste Validation for details
- 2.13. Validate Special Handling Instructions and Additional Information. See section Special Handling Instructions and Additional Information Validation for details
- 2.14. Validate Biennial Report Information. See details in section Biennial Report Information Validation

### Management Method Code Validation

1. If Emanifest.submissionType == "DataImage5Copy" following applies
  - 1.1. If the Waste.managementMethod.code is not provided the service generates an error:
 

```

{
  "message": "Field is Not Provided",
  "field": "Emanifest.wastes[i].managementMethod.code "
}

```
  - 1.2. If the Waste.managementMethod.code is provided validate if code in the lookup. If provided code is not found service generates an error:
 

```

{
  "message": "Provided Value not Found.",
  "field": "Emanifest.wastes[i].managementMethod.code",

```

- ```

        "value": "code value"
    }
}

```
2. If Emanifest.submissionType == "FullElectronic" or "Hybrid" following applies
    - 2.1. If Emanifest.status == "Scheduled" or "InTransit" or "ReadyForSignature" following applies
      - 2.1.1. If the Waste.managementMethod.code is not provided the service generates a warning:
 

```

{
    "message": "Field is Not Provided",
    "field": "Emanifest.wastes[i].managementMethod.code "
}

```
      - 2.1.2. If the Waste.managementMethod.code is provided validate if code in the lookup. If provided code is not found service generates a warning:
 

```

{
    "message": "Provided Value not Found.",
    "field": "Emanifest.wastes[i].managementMethod.code",
    "value": "code value"
}

```
    - 2.2. If Emanifest.status == "Signed" following applies
      - 2.2.1. If the Waste.managementMethod.code are provided then the service updates Emanifest.validationStatus = true
      - 2.2.2. If the Waste.managementMethod.code is not provided the service generates an error:
 

```

{
    "message": "Field is Not Provided",
    "field": "Emanifest.wastes[i].managementMethod.code "
}

```
      - 2.2.3. If the Waste.managementMethod.code is provided validate if code in the lookup. If provided code is not found service generates an error:
 

```

{
    "message": "Provided Value not Found.",
    "field": "Emanifest.wastes[i].managementMethod.code",
    "value": "code value"
}

```

### DOT Information fields validation

The system will perform the following steps on the DOT Information fields:

1. Validate ID Number
  - 1.1. If the dotInformation.IdNumber is not provided, the service generates an error:
 

```

{
    "message": "Mandatory Field is Not Provided",
    "field": "Emanifest.wastes.dotInformation.idNumber ",

```

```
}
```

- 1.2. If dotInformation.IdNumber is provided and not found in Id Number Lookup, the service generates an error:

```
{
```

```
  "message": "Provided Id Number is not Found in DOT's Id Number Lookup",  
  "field": "Emanifest.wastes.dotInformation.IdNumber"  
  "value": "value of Id Number"
```

```
}
```

2. Validate Printed Dot Information.

- 2.1. If printedDotInformation is not provided, the service generates following error:

```
{
```

```
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.wastes.dotInformation.printedDotInformation "
```

```
}
```

- 2.2. If printedDotInformation exceeds the 500 character limit, the service generates the following error:

```
{
```

```
  "message": "Invalid Field Format. Printed Dot Information exceeds the 500 character length",  
  "field": "Emanifest.wastes.dotInformation.printedDotInformation ",  
  "value": "value of printedDotInformation"
```

```
}
```

### Containers and quantity fields validation

1. Containers and Quantity fields are mandatory
2. If the quantity entity is not provided the system shall generate the following error:

```
{
```

```
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.waste.quantity"
```

```
}
```

3. If any of the Containers and Quantity fields are not provided the service generates the following error:

```
{
```

```
  "message": "Mandatory Field is not Provided",  
  "field":  
  "Emanifest.waste.quantity.containerNumber/containerType/quantity/quantityUnitOfMeasurem  
ent"
```

```
}
```

4. If the element containerNumber is provided, the service validates containerNumber format. If format is not valid the service generates an error:

```
{
```

```
  "message": "Invalid Field Format. Integer number not exceeding 9999 is expected",
```



```

    "field": "Emanifest.wastes.quantity.containerNumber",
    "value": "value of container number"
  }

```

5. If the element containerType code is provided, the service validates the containerType code. If containerType code is not valid the service generates an error:

```

{
  "message": "Invalid Field Format. Provided container type code not found in lookup.",
  "field": "Emanifest.wastes.quantity.containerType.code",
  "value": "value of container type code "
}

```

6. If the element quantity is provided, the service validates the quantity format. If the format is not valid the service generates an error:

```

{
  "message": "Invalid Field Format. Expect a number containing no more than 11 whole digit(s)",
  "field": "Emanifest.wastes.quantity.quantity ",
  "value": "value of quantity "
}

```

7. If the quantityUnitOfMeasurement code is provided, the service validates the quantityUnitOfMeasurement code. If the quantityUnitOfMeasurement code is not valid, the service generates an error:

```

{
  "message": "Invalid Field Format. Provided quantityUnitOfMeasurement code not found in lookup.",
  "field": "Emanifest.wastes.quantity.quantityUnitOfMeasurement.code ",
  "value": "value of container type"
}

```

### Hazardous Waste Validation

1. If for every waste in the manifest waste.epaWaste == false and for every waste in the manifest waste.pcb == true and no waste codes are provided then the service does not generate any errors or warnings

2. If generator.siteAddress.state.code == "IL" or designatedFacility.siteAddress.state.code == "IL" and no federal or state codes are provided then the service does not generate errors or warnings

3. If the waste.dotHazardous is true and waste.epaWaste is true the following applies:

- 3.1. If no federal waste codes and no TSDF/Generator State waste codes are provided, the service generates the following error:

```

{
  "message": "No Federal state waste codes or TSDF or Generator state waste codes are provided. At least one Federal or State Waste Code shall be provided for the Waste",
  "field": "Emanifest.wastes.hazardousWaste.federalWasteCodes ",
}

```

3.2. If all Federal waste codes are invalid, the service generates the following error(s):

```
{
  "message": "Invalid Federal waste code is provided. Waste codes have a 4 character length, first
character a letter F, K, P, or U, last three characters expected to be numeric. At least one valid
Federal or State waste code shall be provided for the Waste",
  "field": "Emanifest.wastes.hazardousWaste.federalWasteCodes.code ",
  "value": " federalWasteCode value"
}
```

3.3. If Generator.siteAddress.state.code is not "TX" then  
hazardousWaste.generatorStateWasteCodes shall be provided and  
hazardousWaste.txStateWasteCodes shall be ignored

3.4. If DesignatedFacility.siteAddress.state.code is not "TX"  
hazardousWaste.tsdfStateWasteCodes shall be provided and  
hazardousWaste.txStateWasteCodes shall be ignored.

3.5. If Generator.siteAddress.state.code is "TX" then hazardousWaste.txStateWasteCodes shall  
be provided and hazardousWasteCode.generatorStateWasteCodes shall be ignored

3.6. If DesignatedFacility.siteAddress.state.code is "TX" hazardousWaste.txStateWasteCodes  
shall be provided hazardousWaste.tsdfStateWasteCodes shall be ignored.

3.7. If TSDF/Generator state waste code(s) are provided and  
TSDF/Generator.siteAddress.state.code are not "TX" the following applies:

3.7.1. If TSDF/Generator state waste codes are invalid, the service generates the following  
error(s):

```
{
  "message": "Invalid TSDF/Generator waste code provided. Six-character length,
alphanumeric characters expected. At least one Federal or State waste code shall be
provided for the Waste",
  "field": "Emanifest.wastes.hazardousWaste.tsdf/generatorWasteCode.code",
  "value": "Waste Code value"
}
```

3.7.2. If TSDF/Generator state waste codes are valid then the service checks if provided  
state waste code is defined for the state. If the waste code is not defined for the  
state, the service generates following error(s):

```
{
  "message": " Invalid TSDF/Generator waste code provided. At least one Federal or
State waste code shall be provided for the Waste",
  "field": "Emanifest.wastes.hazardousWaste.tsdf/generatorWasteCodes ",
}
```

3.8. If TSDF/Generator state waste code(s) are provided and either TSDF or  
Generator.siteAddress.state.code is "TX" and state waste code(s) are invalid then the  
service generates the following error:

```
{
```

- ```

    "message": "Invalid TSDF/Generator waste code provided. Eight-character length,
    alphanumeric characters expected for TX. At least one Federal or State waste code shall be
    provided for the Waste",
    "field": "Emanifest.wastes.hazardousWaste.tsdf/generatorStateWasteCodes.code",
    "value": "Waste Code value"
  }

```
4. If waste.dotHazardous is true and waste.epaWaste is false, at least one state waste code is valid and other state waste codes are invalid or not defined for the state, then the service generates warnings for all invalid state waste codes:
 

```

      {
        "message": "Provided TSDF/Generator Waste Codes will be ignored.",
        "field":
        "Emanifest.wastes.hazardousWaste.tsdfStateWasteCodes/generatorStateWasteCodes.code",
        "value": "waste code value"
      }

```
  5. If waste.dotHazardous is false and Federal waste code(s) are provided, the service generates a warning:
 

```

      {
        "message": "Provided Federal Waste Codes will be ignored.",
        "field": "Emanifest.wastes.hazardousWaste.federalWasteCodes.code",
        "value": "federal waste code value"
      }

```
  6. If the waste.dotHazardous is false and the waste.epaWaste is true the service generates following warning:
 

```

      {
        "message": "Provided EPA Waste value will be ignored. If the Waste.dotHazardous is false the
        waste.epaWaste cannot be true ",
        "field": "Emanifest.wastes.epaWaste ",
        "value": "epaWaste value"
      }

```
  7. If the Waste.dotHazardous is true and Waste.epaWaste is false and Federal waste code(s) are provided, the service generates a warning:
 

```

      {
        "message": "Provided Federal Waste Codes will be ignored.",
        "field": "Emanifest.wastes.hazardousWaste.federalWasteCodes.code",
        "value": "federal waste code value"
      }

```
  8. If Emanifest.generator.siteAddress.state.code ==  
Emanifest.designatedFacility.siteAddress.state.code and  
waste.hazardousWaste.tsdfStateWasteCodes and  
waste.hazardousWaste.generatorStateWasteCodes are provided, the following applies

8.1. The service will store waste.hazardousWaste.tsdfStateWasteCodes and waste.hazardousWaste.generatorStateWasteCodes under waste.hazardousWaste.generatorStateWasteCodes

8.2. The service will generate following warning:

```
{  
  "message": "Provided TSDF Waste Codes will be stored with Generator Waste Codes. If  
  Generator and TSDF are located in the same state then Generator and TSDF waste codes shall  
  be provided under Generator waste codes",  
  "field": "Emanifest.wastes.hazardousWaste.tsdfWasteCodes"  
}
```

### Biennial Report Information Validation

1. If the Waste.br is true, then the Waste.brInfo fields are mandatory. If the Waste.brInfo is not provided or empty, then the system generates following error:

```
{  
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.waste.brInfo"  
}
```

2. Validate Source Code

2.1. If the Source Code is not provided, the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.waste.brInfo.sourceCode.code"  
}
```

2.2. If Source Code is not found, the service generates an error

```
{  
  "message": "Provided Value not Found",  
  "field": "Emanifest.waste.br.sourceCode.code",  
  "value": "code value"  
}
```

3. Validate Form Code

3.1. If Form Code is not provided the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.waste.brInfo.formCode.code"  
}
```

3.2. If Form Code is not found the service generates an error

```
{  
  "message": "Provided Value not Found",  
  "field": "Emanifest.waste.brInfo.formCode.code",  
  "value": "code value"
```

```
}
```

#### 4. Validate Waste Minimization Code

4.1. If Waste Minimization is not provided the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.waste.brInfo.wasteMinimization.code"  
}
```

4.2. If the Waste Minimization is not found the service generates an error

```
{  
  "message": "Provided Value not Found",  
  "field": "Emanifest.waste.brInfo.wasteMinimization.code",  
  "value": "code value"  
}
```

#### 5. Validate Density and Density Units Of Measurement

5.1. If Waste.UnitOfMeasurement.code is not G or L or N (volumes) then Density and Density Units Of Measurement will be ignored with the following warning:

```
{  
  "message": "Provided field will be ignored. Density shall be provided only if the quantity  
units of measurement is the volume (G, L, N)",  
  "field": "Emanifest.waste.brInfo.density",  
  "value": "density value"  
}
```

5.2. If Waste.UnitOfMeasurement.code is G or L or N (volumes) and density is not provided the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided",  
  "field": "Emanifest.waste.brInfo.density"  
}
```

5.3. If Waste.UnitOfMeasurement.code is G or L or N (volumes) and density is provided, the service validates the Density format. If Density format is invalid, the service generates an error

```
{  
  "message": "Invalid Field Format. Numeric value containing no more than 3 whole digit(s)  
and 2 decimal digit(s) expected",  
  "field": "Emanifest.waste.brInfo.density",  
  "value": "density value"  
}
```

5.4. If Waste.UnitOfMeasurement.code is G or L or N (volumes) and Density Units Of Measurement is not provided, the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided",
```

```
"field": "Emanifest.waste.brInfo.densityUnitOfMeasurement.code "
}
```

- 5.5. If Density Units Of Measurement is provided the service checks if the Density Units Of Measurement is in the lookup format. If the Density Units Of Measurement is not found, the service generates an error:

```
{
  "message": "Provided Value not Found",
  "field": "Emanifest.waste.brInfo.densityUnitOfMeasurement.code ",
  "value": "code value"
}
```

6. If brInfo == false and br fields are provided the following warning will be generated:

```
{
  "message": "Provided field will be ignored",
  "field": "Emanifest.waste.brInfo.",
  "value": "value"
}
```

### PCB Information Validation

1. If pcb == false and any of the pcbInfos entities or fields are provided the service generates following warning:

```
{
  "message": "Field will be ignored"
  "field": "Emanifest.waste.pcbInfos"
  "value": " pcbInfo value"
}
```

2. If pcb == true and pcbInfos are not provided the service generates following error:

```
{
  "message": "Mandatory Field is not Provided",
  "field": "Emanifest.waste.pcbInfos"
}
```

3. If at least one PcbInfos element is provided then service validates pcbInfo fields

#### 3.1. Validate PcbInfos.loadType.code

- 3.1.1. If PcbInfos.loadType is not provided the service generates an error:

```
{
  "message": "Mandatory Field is not Provided",
  "field": "Emanifest.wastes.pcbInfos.loadType.code"
}
```

- 3.1.2. If pcbInfos.loadType.code is provided the service validates it against LoadType lookup. If the field is not valid the service generates following error:

```
{
```

```
    "message": "Invalid Field Format. One of the following values is expected "Container",  
    "ArticleInContainer", "ArticleNotInContainer", "BulkWaste"",  
    "field": "Emanifest.waste.pcbInfos.loadType.code"  
    "value": "loadType.code value"  
  }
```

### 3.2. Validate pcbInfos.articleContainerId

- 3.2.1. If pcbInfos.loadType.code is "Container" or "ArticleInContainer" or "ArticleNotInContainer" and PcbInfo.articleContainerId is not provided the service generates an error:

```
{  
  "message": "Mandatory Field is not Provided. Fields must be provided if PCB load  
  type is "Container" or "ArticleInContainer" or "ArticleNotInContainer"",  
  "field": "Emanifest.waste.pcbInfos. articleContainerId"  
}
```

- 3.2.2. If pcbInfos.articleContainerId is provided the service checks the field length. If the field is invalid the service generates following error:

```
{  
  "message": "Invalid Field Format. Text no longer than 255 characters is expected",  
  "field": "Emanifest.waste.pcbInfos.articleContainerId"  
  "value": "articleContainerId value"  
}
```

- 3.2.3. If pcbInfos.loadType.code is "BulkWaste" and articleContainerId is provided then service generates following warning:

```
{  
  "message": "Field will be ignored"  
  "field": "Emanifest.waste.pcbInfos.articleContainerId"  
  "value": "articleContainerId value"  
}
```

### 3.3. Validate PcbInfo.dateOfRemoval

- 3.3.1. If pcbInfos.dateOfRemoval is not provided the service generates following error:

```
{  
  "message": "Mandatory Field is not Provided.",  
  "field": "Emanifest.waste.pcbInfos.dateOfRemoval"  
}
```

- 3.3.2. If PcbInfos.dateOfRemoval is provided the service validates field format. If the field format is invalid the service generates following error:

```
{  
  "message": "Invalid Field Format.",  
  "field": "Emanifest.waste.pcbInfos.dateOfRemoval"  
  "value": "dateOfRemoval value"  
}
```

### 3.4. Validate pcbInfos.weight

3.4.1. If pcbInfos.weight is not provided the service generates following error:

```
{  
  "message": "Mandatory Field is not Provided.",  
  "field": "Emanifest.waste.pcbInfos.weight"  
}
```

3.4.2. If PcbInfos.weight is provided the service validates field format. If the field format is invalid the service generates following error:

```
{  
  "message": "Invalid Field Format. Value containing no more than 11 whole digit(s)  
and 6 decimal digit(s) is expected",  
  "field": "Emanifest.waste.pcbInfo. weight"  
  "value": "weight value"  
}
```

### 3.5. Validate PcbInfos.wasteType

3.5.1. If pcbInfos.loadType is "Container" or "ArticleInContainer" and wasteType is not provided the service generates following error:

```
{  
  "message": "Mandatory Field is not Provided.",  
  "field": "Emanifest.waste.pcbInfos.wasteType"  
}
```

3.5.2. If PcbInfos.loadType is "BulkWaste" and wasteType is provided the service generates following warning:

```
{  
  "message": "Field will be ignored"  
  "field": "Emanifest.waste.pcbInfos.wasteType"  
  "value": "containerType value"  
}
```

3.5.3. If PcbInfos.wasteType is provided the service checks the field length. If the field is invalid the service generates following error:

```
{  
  "message": "Invalid Field Format. Text no longer than 255 characters is expected",  
  "field": "Emanifest.waste.pcbInfos.wasteType"  
  "value": "containerType value"  
}
```

### 3.6. Validate PcbInfos.bulkIdentity

3.6.1. If PcbInfos.loadType "BulkWaste" and bulkIdentity is not provided the service generates following error:

```
{  
  "message": "Mandatory Field is not Provided.",  
  "field": "Emanifest.waste.pcbInfos.bulkIdentity"  
}
```



- 3.6.2. If PcbInfos.loadType is "Container" or "ArticleInContainer" and bulkIdentity is provided the service generates following warning:
 

```
{
    "message": "Field will be ignored"
    "field": "Emanifest.waste.pcbInfos.bulkIdentity"
    "value": "bulkIdentity value"
}
```
- 3.6.3. If PcbInfos.bulkIdentity is provided the service checks the field length. If the field is invalid the service generates following error:
 

```
{
    "message": "Invalid Field Format. Text no longer than 255 characters is expected",
    "field": "Emanifest.waste.pcbInfos.bulkIdentity"
    "value": "bulkIdentity value"
}
```

### Waste Additional Information Validation

1. All Additional Information fields are optional. If any of the fields are invalid, they will be ignored with a warning.
2. handlingInstructions Validation.
  - 2.1. If handlingInstructions is invalid the service generates a warning:
 

```
{
    "message": "Invalid Value. Text can be no longer than 4000 characters",
    "field": "Emanifest.wastes[i].additionalInfo.handlingInstructions",
    "value": "handlingInstructions value"
}
```
3. comments Validation
  - 3.1. handlerId validation
    - 3.1.1. If the handler Id is not provided, then the comment will be ignored with a warning:
 

```
{
    "message": "Comment will be ignored. Field is not Provided",
    "field": "Emanifest.wastes[i].additionalInfo.comments.handlerId"
}
```
    - 3.1.2. If the handlerId has an invalid format provided, then the comment will be ignored with a warning:
 

```
{
    "message": "Comment will be ignored, due to an invalid Site ID format being submitted",
    "field": "Emanifest.wastes[i].additionalInfo.comments.handlerId"
    "value": "Handler ID value"
}
```

3.1.3. If the provided handlerId does not match the Generator or TSDf or any of Transporter Site IDs in the manifest, the comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Provided handlerId does not match Generator or
  TSDf or any of Transporter Site IDs",
  "field": "Emanifest. wastes[i].additionalInfo.comments.handlerId"
  "value": "Handler ID value"
}
```

### 3.2. label validation

3.2.1. If the label element is not provided, then comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Field is not Provided",
  "field": "Emanifest. wastes[i].additionalInfo.comments.label"
}
```

3.2.2. If the label element has an invalid format provided, then comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Invalid Value. Text can be no longer than 250
  characters.",
  "field": "Emanifest. wastes[i].additionalInfo.comments.label"
  "value": "Label value"
}
```

### 3.3. description validation

3.3.1. If the description element is not provided, then the comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Field is not Provided",
  "field": "Emanifest.wastes[i].additionalInfo.comments.description"
}
```

3.3.2. If the description element has an invalid format, the provided comment will be ignored with a warning:

```
{
  "message": "Comment will be ignored. Invalid Value. Text can be no longer than 250
  characters.",
  "field": "Emanifest. wastes[i].additionalInfo.comments.description"
  "value": "Description value"
}
```

## Attachment Validation

1. For the "DataImage5Copy" submission type an attachment contains the scanned copy of the printed manifest form 8700-22 and accompanying 8700-22a's signed by the Generator, Transporter(s), and

TSDF. This document will be obtained via Get Manifest service and will contain all Manifest data in the format of the 8700-22.

2. For the "Image" submission type an attachment contains the scanned copy of the paper manifest form 8700-22 and accompanying 8700-22a's signed by the Generator, Transporter(s), and TSDF.
3. The scanned document shall be zipped and submitted as a part of the Save and Update Manifest service. The zipped manifest attachment shall be passed as a multipart binary content. The attachment may be submitted at the ReadyForSignature status. If the manifest contains an attachment, then an attachment metadata elements shall be provided.
4. If the document is already stored as a part of the manifest and document is not provided, the service generates no errors or warnings.
5. If the document is not stored and the status is ReadyForSignature, the document is mandatory.
6. If an attachment is provided, the service checks if an attachment is a zip file (The services shall only accept in the printedDocument element one pdf file in a zip file). If a non-zipped attachment was provided the service generates the following error:

```
{
  "message": "Attached document is not compressed(zipped). Service accepts compressed (zip) attachments only",
  "field": "Emanifest.printedDocument.name",
  "value": "name of the attached document"
}
```

7. If a zipped attachment is provided, the service will unzip the document and perform the following steps:

7.1. If zip file contains more than one document the service generates an error

```
{
  "message": "Zip contains more than one document",
  "field": "Emanifest.printedDocument"
}
```

7.2. If the attachment contained in zip file is not a PDF, the service generates an error:

```
{
  "message": "Attachment Document is not pdf",
  "field": "Emanifest.printedDocument"
}
```

8. If an attachment is not provided and Emanifest.printedDocument metadata is provided, the service generates a warning:

```
{
  "message": "Attachment Document metadata is provided but no Attachment found",
  "field": "Emanifest.printedDocument.name"
}
```

9. If the following attachment metadata is not provided:

```
"printedDocument" : {
  "name" : "name provided by the user.pdf",
  "size" : {pdf document size in bytes},
```

```
"mimeType" : "APPLICATION_PDF"
}
```

The service generates the following warning:

```
{
  "message": "Attachment Document metadata is not provided",
  "field": "Emanifest.printedDocument"
}
```

10. If one or more metadata fields is not provided, the service generates the following warning:

```
{
  "message": "Attachment Document metadata is not provided",
  "field": "Emanifest.printedDocument.name/size/mimeType"
}
```

11. If metadata is provided but some of the fields values do not match the zipped content, the service generates a warning(s):

```
{
  "message": "Attachment Document name/size does not match actual file name/size ",
  "field": "Emanifest.printedDocument.name/size",
  "value": "Emanifest.printedDocument.name/size value "
}
```

12. If an incorrect mimeType is provided, the service generates following warning:

```
{
  "message": "Invalid Attachment Document Mime Type is provided. APPLICATION_PDF
mime type shall be specified",
  "field": "Emanifest.printedDocument.mimeType",
  "value": "Emanifest.printedDocument.mimeType value"
}
```