

Save manifest.....	1
Update manifest	10
Manifest entities and fields validation for Save service	17
Manifest entities and fields validation for Update service	21
Designated Facility (TSDF)/Generator Site Id and Site Information Validation	24
Generator Site Id and Site Information Validation	28
Transporter Site id and Site Information Validation.....	35
Paper Signature Info Validation	37
Special Handling Instructions and Additional Information Validation.....	41
Waste Validation.....	43
DOT Information fields validation.....	45
Containers and quantity fields validation	49
Hazardous Waste Validation.....	50
Biennial Report Information Validation	51
Waste Special Handling Instructions and Additional Information Validation	53
Management Method Code Validation	54
Attachment Validation.....	55

Save manifest

Description

Service creates the new Manifest. Service accepts Manifest data in JSON format compliant with e-Manifest JSON schema. Following submission types will be supported for this release:

- "DataImage1Copy" (Won't be supported for June 2018 release)
- "Data"
- "DataImage5Copy"

For "DataImage1Copy" submission type Manifest will be assigned "Scheduled" status. The transition to the following statuses can be done only via Update Manifest service:

- "InTransit"
- "Received"
- "ReadyForSignature"

For "Data" submission type Manifest will be assigned "Scheduled" status. The transition to the next statuses for this submission type will be done automatically by E-Manifest.

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

For the "DataImage5Copy" submission type 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). 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,  
  "mimeType": "APPLICATION_PDF"  
}
```

Service will validate submitted JSON (see details in section Manifest entities and fields validation) 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 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

Service Invocation Example

<server url>/rcrainfo/rest/api/v1/emanifest/manifest/save/{manifest JSON}

Request JSON Schema

"emanifest.json"

Manifest JSON examples

Valid Manifest JSON Example for "DataImage1Copy" submission type

```
{  
  "submissionType": "DataImage1Copy",  
  "generator": {  
    "epaSiteId": "MDD981111081",  
    "contact": {  
      "phone": {  
        "number": "301-294-9727"  
      }  
    }  
  },  
  "emergencyPhone": {  
    "number": "301-423-5455"  
  }  
}
```

```

    },
    "transporters" : [ {
      "epaSiteId" : "CAR000189282",
      "name" : "transporter 1",
      "order" : 1
    },
    {
      "epaSiteId" : "CAD982000564",
      "name" : "transporter 2",
      "order" : 2
    }
  ],
  "designatedFacility" : {
    "epaSiteId" : "AK8570028649"
  },
  "wastes" : [ {
    "hazardous" : true,
    "dotInformation" : {
      "idNumber" : {
        "code" : "UN1988"
      },
      "hazardClass" : {
        "code" : "3"
      },
      "properShippingName" : {
        "code" : "Aldehydes, flammable, toxic, n.o.s."
      },
      "technicalName" : "technical name",
      "packingGroup" : {
        "code" : "I"
      },
      "rqDescription" : "rq description",
      "printedDotInformation" : "UN1988, Waste Aldehydes, flammable, toxic, n.o.s., (technical name), 3, I, RQ(rq description)",
      "emergencyGuideNumber" : {
        "code" : "131"
      }
    },
    "quantity" : {
      "containerNumber" : 100,
      "containerType" : {
        "code" : "BA"
      },
      "quantity" : 1.67,
      "unitOfMeasurement" : {
        "code" : "L"
      }
    },
    "hazardousWaste" : {
      "federalWasteCodes" : [ {
        "code" : "D023"
      }, {
        "code" : "D001"
      }, {
        "code" : "D021"
      }
    ],
    "tsdfStateWasteCodes" : [ ],

```

```

    "generatorStateWasteCodes" : [ ]
  },
  "additionalInfo" : {
    "comments" : [ ]
  },
  "lineNumber" : 1,
  "brInfo" : true,
  "br" : {
    "densityUnitOfMeasurement" : {
      "code" : "1"
    },
    "formCode" : {
      "code" : "W107"
    },
    "sourceCode" : {
      "code" : "G07"
    },
    "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 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
 - 2.3. The system will check if the user has Site Manager Permissions for the TSDF Site ID or Generator 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”.
- 2.4. If the User does not have Site Manager Permissions for the TSDF Site and for Generator 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 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
 - 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 error and warning 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. DataImage1Copy 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 specified in provided JSON

System	E_SystemError	Message describing system error
--------	---------------	---------------------------------

Update manifest

Description

The 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 and can also accept Manifest attachment (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.

Following submission types will be supported for this release:

- "DataImage1Copy" (Won't be supported for June 2018 release)
- "Data"
- "DataImage5Copy"

For "DataImage1Copy" submission type Manifest can be submitted with one of the following status:

- "Scheduled"
- "InTransit"
- "Received"
- "ReadyForSignature"

Submission type can be updated only at "Scheduled" status.

For "Data" submission type the Manifest status shall not be submitted. For this submission type the status transition will be done by e-Manifest.

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

For the "DataImage1Copy" and "DataImage5Copy" submission types (in "Received" and "ReadyForSignature" statuses) 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). 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,
  "mimeType": "APPLICATION_PDF"
}
```

Service will validate submitted JSON (see section [Manifest entities and fields validation](#)) and if:

- No Errors or Warnings were found: The service will update the manifest and return existing Manifest Tracking Number
- Only Warnings were found: The service will update the manifest and return existing Manifest Tracking Number and Warning Report containing all found Warnings

- Error(s) were found: The service won't update the manifest and return Error Report containing all found errors.
- Error(s) and Warnings were found. The service won't update the manifest and return 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. Manifest attachment shall be passed as a multipart binary content

Service Invocation Example

<server url>/rcrainfo/rest/api/v1/emanifest/manifest/update/{manifest JSON}

Request JSON Schema

"emanifest.json"

Manifest JSON examples

Valid Manifest JSON Example for "DataImage1Copy" submission type

```
{
  "manifestTrackingNumber": "100000781ELC",
  "status": "Scheduled",
  "submissionType": "DataImage1Copy",
  "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": [ {
```

```

    "hazardous" : true,
    "dotInformation" : {
      "idNumber" : {
        "code" : "UN1988"
      },
      "hazardClass" : {
        "code" : "3"
      },
      "properShippingName" : {
        "code" : "Aldehydes, flammable, toxic, n.o.s."
      },
      "technicalName" : "technical name",
      "packingGroup" : {
        "code" : "I"
      },
      "rqDescription" : "rq description",
      "printedDotInformation" : "UN1988, Waste Aldehydes, flammable, toxic, n.o.s., (technical name), 3,
I, RQ(rq description)",
      "emergencyGuideNumber" : {
        "code" : "131"
      }
    },
    "quantity" : {
      "containerNumber" : 100,
      "containerType" : {
        "code" : "BA"
      },
      "quantity" : 1.67,
      "unitOfMeasurement" : {
        "code" : "L"
      }
    },
    "hazardousWaste" : {
      "federalWasteCodes" : [ {
        "code" : "D023"
      }, {
        "code" : "D001"
      }, {
        "code" : "D021"
      } ],
      "tsdfStateWasteCodes" : [ ],
      "generatorStateWasteCodes" : [ ]
    },
    "additionalInfo" : {
      "comments" : [ ]
    },
    "lineNumber" : 1,
    "brInfo" : true,
    "br" : {
      "densityUnitOfMeasurement" : {
        "code" : "1"
      },
      "formCode" : {
        "code" : "W107"
      },
      "sourceCode" : {

```

```

    "code" : "G07"
  },
  "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. 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 processing the manifest and generates 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 the TSDF Site ID or Generator Site ID 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 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
 - 3.2. If no errors or warnings are generated during the validation process the service will perform 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 the 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 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 error and warning 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-emanifest-success.json” : Response for successful submission

“save-emanifest-failure.json”: Response for failed submission

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. DataImage1Copy 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": "siteld",
            "entityIdValue": "N/A"
        },
        "errors": [
            {
                "message": "Mandatory Field is not provided",
                "field": "Emanifest.designatedFacility.siteld"
            },
            {
                "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"
                }
            ]
        }
    ]
}

```


$$\left\{ \begin{array}{l} \text{[} \\ \text{] } \end{array} \right\}$$

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 specified in provided JSON
System	E_SystemError	Message describing system error

1. `Emanifest.manifestTrackingNumber`

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

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

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

- 1.4. If submissionType is "DataImage5Copy" and 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 TrackingNumber is already stored in the e-
Manifest database",
 "field": "Emanifest.manifestTrackingNumber",
 "value": "mtn value"
}
```

2. Emanifest.status.

2.1. If the status is provided and the submission type is "DataImage5Copy" 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"
}
```

3. Emanifest.submissionType.

3.1. One of the following submissionType must be provided: "Data", "DataImage1Copy", "DataImage5Copy".

3.2. If the submissionType is not "Data", "DataImage1Copy", or "DataImage5Copy" the service will generate following error:

```
{
 "message": "Invalid Field Format. One of the following values
'Data', 'DataImage1Copy', 'DataImage5Copy' is expected",
 "field": "Emanifest. submissionType",
 "value": " status value"
}
```

3.3. If submissionType is not provided the service will generate following error:

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

4. Validation of the State Waste Codes.

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 any Waste Code. At least one State Waste Code shall be provided for the Manifest.

4.2. If 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

### 5.1. Validate lineNumber(s)

- 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 provided validate format.

{  
“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”: “Provided Value is Not Valid”,  
“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.

- 6.1. If transporters entity is not provided the service generates following error:

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

```
}
```

7. Transporter(s) order(s) validation.

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 provided validate order format. If Transporter.order is invalid 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 orders.

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. Validate Generator Information. See details in [Generator Site Id and Site Information Validation](#) section

9. Validate Transporters Information. See details in [Transporter Site id and Site Information Validation](#) section.

10. Validate Designated Facility Information. See detail in [Designated Facility \(TSDF\)/Generator Site Id and Site Information Validation](#) section.

11. Validate Wastes Information. See details in [Waste Validation](#) section.

12. Validate Handling Instructions Information. See details in [Special Handling Instructions and Additional Information Validation](#) section.
13. If `submissionType == DataImage5Copy` validate Attachment Information. See details in 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 `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 provided `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 == DataImage1Copy` following validations will be done

- 2.1.1. If status is not provided the service generates an error:

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

2.1.2. If status is provided but the status is not "Scheduled", "InTransit", "Received", "ReadyForSignature" the service generates an error:

```
{
 "message": "Invalid Field Format. One of the following values: "Scheduled",
 "InTransit", "Received", "ReadyForSignature" is expected",
 "field": "Emanifest.status",
 "value": " status value"
}
```

2.1.3. If a valid status is provided check, if the update service attempt to perform valid status transition. Following rules will apply:

2.1.3.1. Manifest can transition from Scheduled to any of the following statuses:

- InTransit
- Received
- ReadyForSignature

2.1.3.2. Manifest can transition from InTransit to any of the following statuses:

- Received
- ReadyForSignature

2.1.3.3. Manifest can transition from Received to any of the following statuses:

- InTransit
- Received
- ReadyForSignature

2.1.3.4. Manifest cannot transition to any other status from ReadyForSignature

2.1.4. If current Manifest status is "InTransit" and provided status is "Scheduled" the service generates an error:

```
{
 "message": "Current Manifest status is InTransit. It cannot be updated to
Scheduled",
 "field": "Emanifest.status",
 "value": " status value"
}
```

2.1.5. If current Manifest status is "ReadyForSignature" and provided status is not "ReadyForSignature" the service generates an error:

```
{
 "message": "Current Manifest status is ReadyForSignature. It cannot be
updated to any other status",
 "field": "Emanifest.status",
 "value": " status value"
}
```

2.2. If submissionType == Data then the status cannot be changed via the update service. Status transitions will be done by e-Manifest and are triggered by the following events:

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

- Scheduled->ReadyForSignature: The status transition is triggered when Transporter signs.
  - ReadyForSignature->Signed: The status transition is triggered when 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 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. If the provided status is different than the current status the service generates a warning:
- ```

{
    "message": "Provided Manifest status will be ignored. For the Data submission
type Manifest status cannot be updated to any other status via Update Manifest
service",
    "field": "Emanifest.status",
    "value": " status value"
}

```
- 2.4. If submissionType == DataImage5Copy 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 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. Emanifest.submissionType.
- 3.3. If submissionType == "DataImage5Copy" then the submissionType cannot be updated
- 3.3.1. If the current submissionType == "DataImage5Copy" 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": "Emanifest. submissionType",
    "value": "submissionType value"
}

```
- 3.4. If current submissionType is "Data" it can be updated to "DataImage1Copy" only at Scheduled status.

3.5. If current submissionType is "DataImage1Copy" it can be updated to "Data" only at Scheduled status.

3.6. If status is not 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": "Emanifest. submissionType",
  "value": "submissionType value"
}
```

3.7. If current Manifest status is Scheduled and the submissionType is not "Data","DataImage1Copy" the service generates following error:

```
{
  "message": "Invalid Field Format. One of the following values "Data","DataImage1Copy"
expected",
  "field": "Emanifest. submissionType",
  "value": "submissionType value"
}
```

4. Validation of the State Waste Codes. This section is identical to Validation of the State Waste Codes in [Manifest entities and fields](#) validation
5. Waste(s) Line Numbers Validation. This section is identical to Waste(s) Line Numbers Validation in Manifest entities and fields validation
6. Transporter(s) order(s) validation. This section is identical to Transporter(s) order(s) validation in Manifest entities and fields validation
7. Validate Generator Information. See details in Generator Site Id and Site Information Validation section
8. Validate Transporters Information. See details in Transporter Site id and Site Information Validation section.
9. Validate Designated Facility Information. See detail in Designated Facility (TSDF)/Generator Site Id and Site Information Validation section.
10. Validate Wastes Information. See details in Waste Validation section.
11. If submissionType == DataImage1Copy or submissionType == DataImage5Copy validate Paper Signature Information. See details in Paper Signature Info Validation section
12. Validate Handling Instructions Information. See details in Special Handling Instructions and Additional Information Validation section.
13. Validate Management Method Code. See details in Management Method Code Validation section.
14. If submissionType == DataImage1Copy or submissionType == DataImage5Copy validate Attachment Information. See details in Attachment Validation section.

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 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 be placed in the Generator Site Id and Emergency phone number must be provided. If other Site Information is provided the information will be ignored with warnings.

1. If the user is authorized to use the e-Manifest Save service for the provided TSDF Site Id then provided Site Id was already validated during authorization and the system will check if the TSDF Site Contact Phone number is provided.

- 1.1. If the Site Contact Phone number is not provided the system generates following error:

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

- 1.2. If the Site Contact Phone number is provided the system will check the format.

- 1.2.1. If the provided phone number has a valid format the system will store site contact phone number into emanifest database

- 1.2.2. If the provided phone number has an incorrect format the system will generate 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"  
}
```

- 1.3. If the Site Contact Phone Extension is provided the system will check the format.

- 1.3.1. If provided phone extension has valid format the system will store the site contact phone extension into emanifest database

- 1.3.2. If the provided phone has incorrect format the system will generate following warning:

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

2. If any of the following TSDF Site Information entities were provided:

- Site Name
- Site Location
- Mailing Address

The system generates 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"  
}
```

3. 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

- 3.1. If the Site Contact Phone number is not provided and registered in RCRAInfo site does not contain Contact Phone the system generates following error:

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

- 3.2. If the Site Contact Phone number is provided and registered in RCRAInfo site contains Contact Phone.

- 3.2.1. If the provided phone number has a valid format the system will store site contact phone number into emanifest database

- 3.2.2. If provided phone number has an incorrect format the system will store registered in RCRAInfo Contact Phone number into e-Manifest database and generate 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"  
}
```

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

- 3.3.1. If the provided phone number has valid format the system will store site contact phone number into emanifest database

- 3.3.2. 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"  
}
```

- 3.4. Validate emergency phone

- 3.4.1. If emergency phone number is not provided the system generates following error:

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

3.4.2. If emergency phone number is provided and has an invalid format the system generates 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"
}
```

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

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

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

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

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

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

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

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

4. Update validation.

4.1. Original epaSiteId submitted via Save emanifest service cannot be updated via Update emanifest service.

4.2. If provided designatedFacility.epaSiteId does not match current designatedFacility.epaSiteId at Update emanifest service 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"
  }
```

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

Generator Site Id and Site Information Validation

1. If submissionType == Data then the Generator is valid, if the Generator is registered and the Generator has at least one User with Certifier Role and this user has ESA == Received
 - 1.1. If submissionType == Data and Generator site id is not provided the service generates an error:

```
{
  "message": "Mandatory Field is not Provided. For Data submission type registered
Generator Site Id must be provided",
  "field": "Emanifest.designatedFacility.epaSiteId"
}
```
 - 1.2. If submissionType == Data and the Generator site id has invalid format the service generates an error:

```
{
  "message": "Invalid Field Format. For Data submission type registered Generator Site Id
must be provided ",
  "field": "Emanifest.designatedFacility.epaSiteId",
  "value": "site id value"
}
```
 - 1.3. If submissionType == Data and the Generator with the provided site id is not registered the service generates an error:

```
{
  "message": "For Data submission type registered Generator Site Id must be provided ",
  "field": "Emanifest.designatedFacility.epaSiteId",
  "value": "site id value"
}
```
 - 1.4. If submissionType == Data and the Generator with the provided site id is registered and there are no users with 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"
}
```
 - 1.5. If submissionType == Data and 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"
```

```
}
```

2. If submissionType == DataImage1Copy or submissionType == DataImage5Copy then registered and non-registered Generators are valid and following cases are possible for the Generator site Id and Generator Information

- 2.1. 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.

- 2.2. If the Generator is registered in RCRAInfo and service requester intend to provide different (than registered) Site Information then 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 emanifest.

- 2.3. 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 optional field for this case. If Site Id is provided the system will check if there is registered Generator with provided Site Id

- 2.4. For all above cases Emergency Phone Number must be provided

3. If the manifest status is InTransit or thereafter then Generator information cannot be updated.

- 3.1. If different than currently stored Generator information is provided the service generates 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"
```

```
}
```

4. Validate Site Id and Site Information

- 4.1. 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"
 }

```
5. If the Site id is provided AND Site address information is not provided.
    - 5.1. Validate site id
    - 5.2. 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.generator.epaSideld ",
 "value": "epa side Id value"
 }

```
    - 5.3. If site id is valid the system will search RCRAInfo by site id.
    - 5.4. If the site id is found
      - 5.4.1. The system will obtain the Site information from RCRAInfo and store it into the Manifest.
      - 5.4.2. set modified= false, registered = true
    - 5.5. If the site id is not found the service generates an error:
 

```

 {
 "message": "Provided Value is not Found",
 "field": "Emanifest.generator.epaSideld",
 "value": "epa site id value"
 }

```
  6. If the Site id is provided AND the Site Information entities are provided.
    - 6.1. If the site name is not provided generate error or not valid (exceeds maximum length) the service generates an error:
 

```

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

```
    - 6.2. 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"
 }

```
    - 6.3. Validate the mandatory location address fields:
      - address1

- city
- state.code
- zip

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

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

6.5. Validate the mandatory mailing address fields:

- address1
- city
- state.code
- zip

6.6. 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"
}
```

6.7. Validate provided site id

6.7.1. 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"
}
```

6.7.2. If the site id is valid check if site is registered

6.7.3. If the site is registered validate if provided site address state code match site id activity location.

6.7.3.1. If the state code does not match 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"
}
```

6.7.4. 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"
}

```

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

7. If the Site id is not provided AND the Site Information is provided.

7.1. Validate the provided Site Information.

7.1.1. Validate the site name

7.1.2. 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"
}

```

7.1.3. Validate the mandatory site location address fields:

- address1
- city
- state.code
- zip

7.1.4. 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"
}

```

7.1.5. Validate the mandatory mailing address fields:

- address1
- city
- state.code
- zip

7.1.6. 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"
}

```

7.2. If provided the Site Information is valid

7.2.1. The service generates a warning:

```

{

```



```
"message": "Value is not Provided",
"field": "Emanifest.generator.epaSiteId"
}
```

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

8. Validate Site Contact Phone Number

8.1. If the Site Contact Phone number is not provided and registered in RCRAInfo does not contain Contact Phone the service generates an error:

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

8.2. If the Site Contact Phone number is provided and registered in RCRAInfo site contains Contact Phone.

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

8.2.2. If provided phone number has 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"
}
```

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

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

8.3.2. 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"
}
```

9. Validate emergency phone

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

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

9.2. If emergency phone number is provided and has 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"
}
```

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

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

## 10. Validate Contact email

10.1. If the provided Contact email has invalid format generate warning:

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

## 11. Update Validation.

11.1. If current status > Scheduled and provided status > Scheduled Generator information cannot be updated.

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

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

11.3. If any of the Generator information fields are different then currently stored the service will generate following warning:

```
{
 "message": "Provided Value will be ignored. For statuses InTransit, Received, ReadyForSignature 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 == Data then Transporter is valid if the Transporter is registered and the Transporter has at least one User with the Certifier Role and this user has ESA == Received
  - 1.1. If submissionType == Data and Transporter site id is not provided the service generates an error:

```
{
 "message": "Mandatory Field is not Provided. For Data submission type registered
Transporter Site Id must be provided",
 "field": "Emanifest.designatedFacility.epaSiteld"
}
```
  - 1.2. If submissionType == Data and Transporter site id has an invalid format the service generates an error:

```
{
 "message": "Invalid Field Format. For Data submission type registered Transporter Site
Id must be provided ",
 "field": "Emanifest.designatedFacility.epaSiteld",
 "value": " site id value"
}
```
  - 1.3. If submissionType == Data and Transporter with the provided site id is not registered the service generates an error:

```
{
 "message": "For Data submission type registered Transporter Site Id must be provided ",
 "field": "Emanifest.designatedFacility.epaSiteld",
 "value": " site id value"
}
```
  - 1.4. If submissionType == Data and Transporter with the provided site id is registered and there are no users with the 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.epaSiteld",
 "value": " site id value"
}
```
  - 1.5. If submissionType == Data and the Transporter 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 Transporter
Site Id",
 "field": "Emanifest.transporter.epaSiteld",
 "value": "site id value"
}
```

2. If submissionType == DataImage1Copy or 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 following site information must be provided:

- Mailing 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.mailingAddress"
}
{
 "message": "Value 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.

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 obtain Site Information from RCRAInfo and store 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
 - 2.5.1. Validate provided site id.
 - 2.5.2. 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",
            "value": "epa site id value"
          }
          
```
 - 2.5.3. If site id is valid check if site is registered
 - 2.5.4. If site is not registered generate 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.
 - 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 mailing address fields:
 - address1
 - city
 - state.code
 - zip
 - 2.6.5. If any of the mandatory location address fields are not provided or invalid generate an error:


```

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

Paper Signature Info Validation

1. Paper signature information shall not be provided for Data submission type.

1.1. If submissionType == Data and any (Generator, Transporter(s), TSDF) of 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 Data submission type",  
  "field": "Emanifest.generator/transporter/designatedFacility.paperSignatureInfo",  
  "value": "printedName or signatureDate value"  
}
```

2. signatureDate must be provided for DataImage5Copy submission type. printedName is optional field for DataImage5Copy submission type

3. All paperSignatureInfo fields must be provided for DataImage1Copy submission type.

4. The Paper signature fields are optional in the Received status.

5. If submissionType == DataImage1Copy or submissionType == DataImage5Copy Validate the Generator/Transporter/TSDF Printed Names and Dates

5.1. If submissionType == DataImage1Copy and Generator/Transporter/TSDF Printed Names 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.printedName"  
}
```

5.2. If submissionType == DataImage5Copy and Generator/Transporter/TSDF Printed Names are not provided the service generates a warning {

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

5.3. If submissionType == DataImage1Copy and the Generator/Transporter/TSDF Printed Names are provided validate format. If invalid format provided the service generates an error (for this example: TSDF is used)

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

5.4. If submissionType == DataImage5Copy and 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"
}

```

5.5. 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",
}

```

5.6. If Generator/Transporter/TSDF Signature Dates are provided validate format. If format is invalid the service generates an error (for this example: TSDF is used)

```

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

```

5.7. If submissionType == DataImage1Copy and the Generator Signature Date < Manifest Created Date the service generates an error:

```

{
    "message": "Generator Signed Date must be the same or after Created Date",
    "field": "Emanifest.generator.paperSignatureInfo. signatureDate",
    "value": " value"
}

```

5.8. 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"
}

```

5.9. If there is first Transporter signature and the Generator 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"
}

```

6. Validate Consecutiveness of Signature Dates

6.1. If the signature dates for Generator, all Transporters, and TSDF are provided and have valid format and more than one transporter is provided do following steps

6.1.1. Check if Generator signature Date \leq 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"
}
```

6.1.2. 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"
}
```

6.1.3. Check if the TSDF signature date \geq 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"
    }
  
```

- 6.1.4. 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.signaturesD
ate",
      "value": "Signature Date value"
    }
  
```

- 6.1.5. If submissionType == DataImage1Copy and any of the Signature Dates <= created date the service generates an error:

```

    {
      "message": "Invalid Value. Generator/Transporter/TSDf Signature Date must be the
same or later than Created Date",
      "field":
"Emanifest.generator/transporter/designatedFacility.paperSignatureInfo.signaturesD
ate",
      "value": "Signature Date value"
    }
  
```

Special Handling Instructions and Additional Information Validation

1. All Special Handling Instructions and Additional Information are optional. If any of the fields are invalid they will be ignored with warning

2. Special Handling Instructions Validation.

- 2.1. If Special Handling Instructions 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"
    }
  
```

3. Original Manifest Tracking Number Validation

- 3.1. 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"
  }

```

4. References Validation

4.1. Handler Id validation.

4.1.1. If the Handler id is not provided then comment will be ignored with a warning:

```

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

```

4.1.2. If the handler id has invalid format provided then comment will be ignored with a warning:

```

{
  "message": "Comment will be ignored.Invalid format. Valid site id format is expected",
  "field": "Emanifest.additionalInfo.comments.handlerId"
  "value": "handler id value"
}

```

4.1.3. If provided handlerId does not match neither Generator or TSDF or any of Transporter site ids comment will be ignored with a warning:

```

{
  "message": "Comment will be ignored. Provided handlerId does not match neither
  Generator or TSDF or any of Transporter site ids",
  "field": "Emanifest.additionalInfo.comments.handlerId"
  "value": "handler id value"
}

```

4.2. Label validation

4.2.1. 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"
}

```

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

```

{
  "message": "Comment will be ignored. Invalid format.",
  "field": "Emanifest.additionalInfo.comments.label"
  "value": "Label value"
}

```

4.3. Description validation

4.3.1. If the description 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.description"  
}
```

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

```
{  
  "message": "Comment will be ignored.Invalid format.",  
  "field": "Emanifest.additionalInfo.comments.description"  
  "value": "Description 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

2.1. Validate the hazardous element.

- 2.1.1. The hazardous element must be provided. If hazardous is not provided the service generates an error:

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

- 2.1.2. If the hazardous element is provided the system will validate the hazardous element's format. If hazardous is not true or false the service generates an error:

```
{  
  "message": "Invalid Field format. true or false is expected",  
  "field": "Emanifest.waste.hazardous",  
  "value": "hazardous value"  
}
```

2.2. Validate brInfo

- 2.2.1. brInfo must be provided. If brInfo is not provided the service generates an error:

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

“field”: “Emanifest.waste.brInfo”

}

- 2.2.2. If brInfo is provided the system will validate the brInfo element’s format. If brInfo is not true or false the service generates an error:

{

“message”: “Invalid Field format. true or false is expected”,

“field”: “Emanifest.waste.brInfo”,

“value”: “brInfo value”

}

- 2.2.3. If the element brInfo is true the system will validate the BR Information fields. See details in [Biennial Report Information Validation](#) section

2.3. Validate dotInformation/wasteDescription.

- 2.3.1. If the element hazard is true, the waste object must contain the element dotInformation or must contain the wasteDescription element.

- 2.3.2. If the element hazard is true and dotInformation is not provided the service generates an error:

{

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

“field”: “Emanifest.waste.dotInformation”

}

- 2.3.3. If the element hazard is true and dotInformation is provided the service will validate the dotInformation fields. See section [DOT Information fields validation](#) for details

- 2.3.4. If the element hazard is false and dot Information is provided the service generates a warning:

{

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

“field”: “Emanifest.waste.dotInformation”

}

- 2.3.5. If the element hazard 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”

}

- 2.3.6. If the element hazardous is false and the element wasteDescription is provided, then the service validate’s 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”

```
}
```

- 2.3.7. If the element hazardous 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"
```

```
}
```

2.4. Validate DOT Information. See section DOT Information fields validation for details.

2.5. Validate Containers and Quantity. See section Containers and quantity fields validation for details

2.6. Validate Hazardous Waste Codes. See section Hazardous Waste Validation for details

2.7. Validate Special Handling Instructions and Additional Information. See [section](#) Waste Special Handling Instructions and Additional Information Validation for details

2.8. Validate Biennial Report Information. See details in section Biennial Report Information Validation

DOT Information fields validation

1. Validate Proper Shipping Name (PSN)

- 1.1. If the PSN is not provided the service generates an error:

```
{
```

```
  "message": "Mandatory Field is Not Provided",
```

```
  "field": "Emanifest.wastes.dotInformation.properShippingName"
```

```
}
```

- 1.2. If the provided PSN is not found in PSN Lookup the service generates an error:

```
{
```

```
  "message": "Provided Proper Shipping Name is not Found in Proper Shipping Name Lookup",
```

```
  "field": "Emanifest.wastes.dotInformation.properShippingName",
```

```
  "value": "value of PSN"
```

```
}
```

2. Validate RQ Indicator and RQ Description

- 2.1. If the element rqIndicator is false and the rqDescription is provided then rqDescription is ignored and the service generates a warning:

```
{
```

```
  "message": "Provided RQ Description will be Ignored. RQ Indicator is false",
```

```
  "field": "Emanifest.wastes.dotInformation.rqDescription",
```

```
  "value": "value of rqDescription"
```

```
}
```

- 2.2. If the size of the provided RQ Description exceeds 105 characters the service generates an error:

```
{
```

```
  "message": "Invalid Field Format. RQ Description exceeds 105 characters length",
```

```
"field": "Emanifest.wastes.dotInformation.rqDescription",  
"value": "value of rqDescription"  
}
```

3. Validate Technical Name

3.1. If the size of the provided Technical Name exceeds 75 characters the service generates an error:

```
{  
  "message": "Invalid Field Format. Technical Name exceeds 75 characters length",  
  "field": "Emanifest.wastes.dotInformation.technicalName",  
  "value": "value of technicalName"  
}
```

4. Validate Id Number

4.1. If the IdNumber element is not provided the service generates an error:

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

4.2. If IdNumber is provided and not found in Id Number Lookup the service generates an error:

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

5. Validate Hazard Class.

5.1. With the exception of Id Number, UN0190 and PSN "Samples, explosive, other than initiating explosives" if the element hazardClass is not provided the service generates the following error:

```
{  
  "message": "Mandatory Field is Not Provided",  
  "field": "Emanifest.wastes.dotInformation.hazardClass"  
}
```

5.2. If provided the Hazard Class is not found in the Hazard Class Lookup the service generates the following error:

```
{  
  "message": "Provided Hazard Class is not Found in Hazard Class Lookup",  
  "field": "Emanifest.wastes.dotInformation.hazardClass ",  
  "value": "value of hazardClass "  
}
```

6. Validate Packing Group

6.1. If the combination of Id Number and PSN requires a Packing Group and the Packing Group is not provided the service generates an error:

```
{
  "message": "Mandatory Field is Not Provided",
  "field": "Emanifest.wastes.dotInformation.packingGroup"
}
```

6.2. If provided Packing Group is not found in Packing Group Lookup the service generates the following error:

```
{
  "message": "Provided Packing Group is not Found in Packing Group Lookup",
  "field": "Emanifest.wastes.dotInformation.packingGroup",
  "value": "value of packingGroup"
}
```

7. Validate Emergency Guide Number

7.1. If Emergency Guide Number for provided ID Number does not correspond to provided ID Number generate following warning:

```
{
  "message": "Provided Emergency Guide Number does not correspond to provided ID Number",
  "field": "Emanifest.wastes.dotInformation.emergencyGuideNumber",
  "value": "value of emergencyGuideNumber "
}
```

Based on provided ID Number service finds Emergency Guide Number in the link table and populates the e-Manifest database.

7.2. If provided Emergency Guide Number is not found in Emergency Guide Number Lookup the service generate following warning:

```
{
  "message": "Provided Emergency Guide Number is not Found in Emergency Guide Number Lookup",
  "field": "Emanifest.wastes.dotInformation.emergencyGuideNumber",
  "value": "value of emergencyGuideNumber "
}
```

Based on provided ID Number service finds Emergency Guide Number in the link table and populates the e-Manifest database.

7.3. If Emergency Guide Number is not provided the service generate following warning:

```
{
  "message": "Field is not Provided",
  "field": "Emanifest.wastes.dotInformation.emergencyGuideNumber"
}
```

Based on provided ID Number service finds Emergency Guide Number in the link table and populates the e-Manifest database.

8. If the PSN and ID Number are found in the lookups, the service validates if the provided combination is in the link table.

8.1. If the provided combination of PSN and ID Number are not found in the link table the service generates the following errors:

```
{
  "message": "Provided Proper Shipping Name does not correspond to Provided ID Number",
  "field": "Emanifest.wastes.dotInformation.properShippingName",
  "value": "value of PSN"
}
{
  "message": "Provided ID Number does not correspond to Provided Proper Shipping Name ",
  "field": "Emanifest.wastes.dotInformation.idNumber",
  "value": "value of ID number"
}
```

9. If the provided combination of PSN and ID Number are found in the linking table and Packing Group (PG) is found in lookup the service validates if the provided PG is in the link table for the combination of PSN and ID Number.

9.1. If provided the PG is not found the service generates the following error:

```
{
  "message": "Provided Packing Group does not correspond to Provided Proper Shipping Name and Id Number ",
  "field": "Emanifest.wastes.dotInformation.packingGroup",
  "value": "value of PG"
}
```

10. If the provided combination of PSN and ID Number are found in the linking table and Hazard Class (HC) is found in lookup then the service validates, if the provided HC is in the linking table for the combination of PSN and ID Number.

10.1. If provided HC is not found the service generates the following error:

```
{
  "message": "Provided Hazard Class does not correspond to Provided Proper Shipping Name and Id Number ",
  "field": "Emanifest.wastes.dotInformation.hazardClass",
  "value": "value of HC"
}
```

11. Validate Printed Dot Information.

11.1. If printedDotInformation is not provided the service generate following error:

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

11.2. 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 quantity entity is not provided the system shall generate 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 999 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 format is not valid the service generates an error:

```
{
"message": "Invalid Field Format. Expect a number containing no more than 11 whole digit(s)
and no more than 6 decimal 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 containerType 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 the element hazardous is true and (number of Federal waste codes == 0 or all Federal waste codes are invalid) and (number of TSDf state waste codes== 0 or all TSDf state waste codes are invalid or not defined for the state) and (number of Generator state waste codes == 0 or all Generator state waste codes are invalid or not defined for the state) the service generates following errors:

- 1.1. If No federal waste codes are provided the service generates following error:

```
{  
  "message": "No Federal waste codes are provided. At least one Federal Waste Code shall be  
  provided for the Waste",  
  "field": "Emanifest.wastes.hazardousWaste.federalWasteCode ",  
}
```

- 1.2. If all Federal waste codes are invalid the service generates 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.federalWasteCode.code ",  
  "value": " federalWasteCode value"  
}
```

- 1.3. No TSDf/Generator State waste codes provided the service generates following error(s):

```
{  
  "message": "No TSDf/Generator State waste codes provided. No state waste codes are defined  
  for the state. At least one Federal or State waste codes shall be provided for the Waste",  
  "field": "Emanifest.wastes.hazardousWaste.tsdf/generatorWasteCode ",  
}
```

- 1.4. If TSDf/Generator state waste code is invalid the service generates following error(s):

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

}

1.5. If TSDF/Generator state waste code is provided but not defined for the state the service generates following error(s):

{

"message": " Invalid TSDF/Generator waste code provided.No state waste codes are defined for the state. At least one Federal or State waste codes shall be provided for the Waste",

"field": "Emanifest.wastes.hazardousWaste.tsdf/generatorWasteCode ",

}

2. If the element hazardous is true and at least one federal or state waste code is valid and other federal and state waste codes are invalid or not defined for the state then service generates warnings for all invalid federal and state waste codes.

{

"message": "Provided Federal/TSDF/Generator Waste Codes will be ignored.",

"field":

"Emanifest.wastes.hazardousWaste.federalWasteCodes/tsdfStateWasteCode/generatorStateWasteCode.code",

"value": "waste code value"

}

3. If the element hazardous 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"

}

Biennial Report Information Validation

1. If the element brInfo is true the br fields are mandatory. If br element is not provided or empty the system generate following error:

{

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

"field": "Emanifest.waste.br"

}

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.br.sourceCode.code"

}

2.2. If the Source Code is provided check if Source Code is in lookup. 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.br.formCode.code"
}
```

3.2. If the Form Code is provided the service checks if the Form Code is in the lookup. If Form Code is not found the service generates an error

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

4. Validate Density and Density Units Of Measurement

4.1. 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.br.density",
  "value": "density value"
}
```

4.2. 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.br.density"
}
```

4.3. 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.br.density",
  "value": "density value"
}
```

- ```

 }

```
- 4.4. 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.br.densityUnitOfMeasurement.code "
}

```
- 4.5. 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.br.densityUnitOfMeasurement.code ",
 "value": "code value"
}

```
5. If brInfo == false and br fields are provided following warning shall be generated:
- ```

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

```

Waste Special Handling Instructions and Additional Information Validation

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


```

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

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


```

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

```
 - 3.1.2. If the handler id has invalid format then comment will be ignored with a warning:


```

{

```

```
"message": "Comment will be ignored. Valid handler id format is expected",  
"field": "Emanifest.wastes.additionalInfo.comments.handlerId"  
"value": "handler id value"  
}
```

3.1.3. If provided handlerId does not match neither Generator or TSDF or any of
Transporter site ids comment will be ignored with a warning:

```
{  
"message": "Comment will be ignored. Provided handlerId does not match neither  
Generator or TSDF or any of Transporter site ids",  
"field": "Emanifest.wastes.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.additionalInfo.comments.label"  
}
```

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

```
{  
"message": "Comment will be ignored. Invalid field format.",  
"field": "Emanifest.wastes.additionalInfo.comments.label"  
"value": "Label value"  
}
```

3.3. Description validation

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

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

3.3.2. If Description has invalid format then comments will be ignored with a warning:

```
{  
"message": "Comment will be ignored. Invalid format.",  
"field": "Emanifest.wastes.additionalInfo.comments.Description"  
"value": "Description value"  
}
```

Management Method Code Validation

1. If the manifest status is ReadyForSignature the Management Method Code can be submitted. If
Manifest will be submitted without Management Method Code the Manifest it will be accepted

with warnings. If Manifest will be submitted with invalid Management Method Code this Management Method Code will be ignored, the Manifest will be accepted with warning. TSDF Certifier will need to add Management Method Code when signing the Manifest.

1.1. If the Management Method Code is not provided the service generates a warning:

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

1.2. If the Management Method 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.waste.managementMethod.code",
    "value": "code value"
}
```

2. If submission type == DataImage1Copy and the manifest status is ReadyForSignature the Management Method Code is mandatory, it is optional in the previous statuses.

1.1. If the Management Method Code is not provided the service generates an error:

```
{
    "message": "Mandatory Field is Not Provided",
    "field": "Emanifest.wastes.managementMethod.code "
}
```

1.2. If the Management Method Code is provided validate if code in the lookup. If provided code is not found the service generates an error (If status == ReadyForSignature) or warning:

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

Attachment Validation

1. For the DataImage1Copy and DataImage5Copy submission types an attachment contains printed, scanned, signed by Generator, Transporter(s), and TSDF, a manifest form 8700-22 and accompanying 8700-22a's. This document will be obtained via Get Manifest service and will contain all Manifest data in the format of the 8700-22. The document will be zipped and submitted as a part of the Update Manifest service. The zipped manifest attachment shall be passed as a multipart binary content. The attachment may be submitted either at the Received or ReadyForSignature statuses. If the manifest contains an attachment, metadata elements shall be in the manifest JSON

2. If the document is already stored as a part of the manifest and document is not provided service generates no errors or warnings.

3. If the document is not stored and status is ReadyForSignature the document is mandatory.

3.1. An attachment is not provided but json contains printedDocument metadata generate warning:

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

3.2. If an attachment is provided the service checks if an attachment is zip (For DataImage1Copy service 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(zippped). Service accepts compressed (zip) attachments only",
  "field": "Emanifest.printedDocument.name",
  "value": "name of the attached document"
}
```

3.3. If zipped attachment is provided the Service will unzip the document and perform the following steps

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

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

3.3.2. If attachment contained in zip file is not a pdf the service generates an error:

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

3.3.3. Check if json contains following attachment metadata:

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

3.3.4. If metadata is not provided the service generates the following warning:

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


3.3.5. 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"  
}
```

3.3.6. If metadata is provided but some of the fields values does 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 "  
}
```

3.3.7. If 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 "  
}
```