

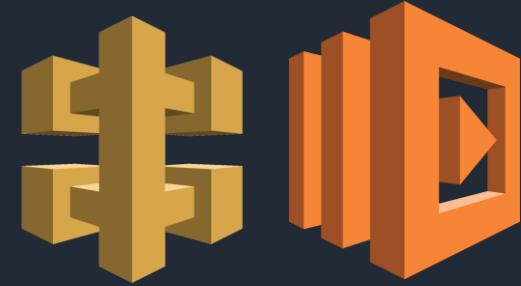
# Demo of a LIXI API for processing LIXI messages

John Matthews  
LIXI

# The LIXI API Template



GitLab



AWS API Gateway  
+ AWS Lambda

# Features of the LIXI API



Validate a messages

Download schemas

Download schema documentation

Translate messages between version of LIXI2

Check a message for deprecated Items

Customise a schema

Upload a message to storage

# Accessing the LIXI API Demo



Open API Specification available here:

<https://lix-public.s3.ap-southeast-2.amazonaws.com/LIXI.json>

View using the Swagger UI: <https://petstore.swagger.io>

Contact me for an API key: [john.matthews@lix.org.au](mailto:john.matthews@lix.org.au)

A screenshot of the Swagger UI interface. At the top left is the Swagger logo with the text 'Supported by SMARTBEAR'. In the center, a search bar contains the URL 'https://lix-public.s3.ap-southeast-2.amazonaws.com/LIXI.json'. To the right of the search bar is a green 'Explore' button. Below the search bar, the main content area displays 'LIXI-API (master)' with a '1.0.0' version tag. Underneath, it shows the base URL: '[ Base URL: a8411qcwrti.execute-api.ap-southeast-2.amazonaws.com/api ]' and the full API URL: 'https://lix-public.s3.ap-southeast-2.amazonaws.com/LIXI.json'. At the bottom left, there are three links: 'Supported LIXI Versions', 'Terms of service', 'Contact the developer', and 'Licence TBD'.

# Validate a LIXI Message



This function will validate a LIXI2 message of any supported LIXI2 version or transaction type.

The function will interpret the message to determine the LIXI schema to use for validation.

## Validate

**POST** /validate Validate the request body against the correct LIXI schema

This function will validate the request body against the correct LIXI schema by re...

### Parameters

Name	Description
<b>x-api-key</b> * required string (header)	The API key associated with this project <i>Default value</i> : IrrnpThRnN5zrdqMRB92F9WSyxWEhAklaFv
<b>body</b> * required string (body)	The LIXI message to be validated

[Example Value](#) | [Model](#)

```
<Package>
<Instructions>
<ApplicationInstructions>
<Update>
<Event Name="Comment" DateTime="2019-09-03T06:58:22.580298
</Update>
</ApplicationInstructions>
</Instructions>
<Publisher CompanyName="LIXI" LIXIcode="LIXILIXI"/>
<Recipient LIXIcode="LIXILIXI"/>
<SchemaVersion LIXITransactionType="CAL" LIXIVersion="2.6.
</Package>
```

Parameter content type

# Download LIXI Schemas



This function will return a specified LIXI transaction schema.

The schema can be returned as a single XSD file or returned as a .zip including HTML documentation.

The version can be specified, otherwise the latest schema version will be returned.

### Download

**GET** /download/{transaction} Download the latest version of a transa

**GET** /download/{transaction}/{version} Download a specific ver

#### Parameters

Name	Description
<b>x-api-key</b> * required string (header)	The API key associated with this project  Default value : IrrnpThRnN5zrdqMRB92F9WSyxV
<input type="text" value="IrrnpThRnN5zrdqMRB92F9WSyxWEhAkIaFv"/>	
<b>transaction</b> * required string (path)	LIXI transaction schema type code associated with  Default value : CAL
<input type="text" value="CAL"/>	
<b>version</b> * required string (path)	LIXI transaction schema version number associated with  Default value : 2.6.21
<input type="text" value="2.6.21"/>	
document string (query)	Documentation will be generated and returned as  Default value : true
<input type="text" value="true"/>	

# Transform between Versions



This function will transform a LIXI message from one LIXI version to another LIXI version.

The function will interpret the message to determine its LIXI version and convert the message to the specified version.

## Transform

**POST** /transform Transform the request body to the latest LIXI version for this

**POST** /transform/{version} Transform the request body to the specified

To modify this function to use your own XSL Transformation, replace the template fil

### Parameters

Name	Description
<b>x-api-key</b> * required string (header)	The API key associated with this project  Default value : IrrnpThRnN5zrdqMRB92F9WSyxWEhA <input type="text" value="IrrnpThRnN5zrdqMRB92F9WSyxWEhAKIaFv"/>
<b>version</b> * required string (path)	LIXI transaction schema version number associated with  Default value : 2.6.21 <input type="text" value="2.6.21"/>
<b>body</b> * required string (body)	The LIXI message to be transformed and returned  Example Value   Model <input type="text" value="string"/>

# Rules Engine



This function will validate a LIXI message against a Schematron rule sheet.

The rule sheet will check whether any LIXI schema items that have been deprecated are used in the message.

## Rules Engine

POST

/rulesengine Validate the request body against a schematron rule sh

Validates the uploaded LIXI message against a schematron rule sheet. The defau

### Parameters

Name	Description
------	-------------

<b>x-api-key</b> * required string (header)	The API key associated with this project  Default value : IrrmpThRnN5zrdqMRB92F9WSyxWEhA
---	--

<b>body</b> * required string (body)	The LIXI message to be validated  Example Value   Model
--	---

```
string
```

Parameter content type

text/plain



# Customise the LIXI schema



This function will customise a LIXI schema according to a set of customisation instructions provided in the request.

The customised schema will be returned as a single XSD file or returned as a .zip including HTML documentation.

## Customise

**POST** /customise/{transaction} Customise the latest version of a LIXI schema according to a set

**POST** /customise/{transaction}/{version} Customise a specified version of a LIXI schema a

### Parameters

Name	Description
<b>x-api-key</b> * required string (header)	The API key associated with this project <i>Default value</i> : IrrnpThRnN5zrdqMRB92F9WSyxWEhAkIaFwaOYUwx <input type="text" value="IrrnpThRnN5zrdqMRB92F9WSyxWEhAkIaFv"/>
<b>transaction</b> * required string (path)	LIXI transaction schema type code associated with this project <i>Default value</i> : CAL <input type="text" value="CAL"/>
<b>version</b> * required string (path)	LIXI transaction schema version number associated with this project <i>Default value</i> : 2.6.21 <input type="text" value="2.6.21"/>
document string (query)	Documentation will be generated and returned as a .zip file <i>Default value</i> : true <input type="text" value="true"/>
<b>body</b> * required string (body)	The LIXI customisation instructions <i>Example Value</i>   <i>Model</i>

# Upload a LIXI Message



This function will upload a LIXI message to storage.

An AWS S3 bucket in the same account as the API itself must be specified.

This function is intended to be used in your own implementation of the LIXI API Template.

## Upload

POST

/upload/{bucket}/{directory}/{filename} Upload the required LIXI message to storage.

### Parameters

Name	Description
<b>x-api-key</b> * required string (header)	The API key associated with this project <i>Default value</i> : IrrnpThRnN5zrdqMRB92F9WSyxWEh
<b>bucket</b> * required string (path)	<i>Default value</i> : my-lixo-bucket
<b>directory</b> * required string (path)	<i>Default value</i> : my-directory
<b>filename</b> * required string (path)	<i>Default value</i> : my-filename.txt
<b>body</b> * required string (body)	The LIXI message to be uploaded <i>Example Value</i>   Model

# How to Deploy a LIXI API



Step-by-step instruction can be found on LIXI Lab:

<https://standards.lixi.org.au/lixi2/API-Template>

1. Open an AWS account
2. Open a GitLab.com account
3. Create a new GitLab project (Import the LIXI API Template files)
4. Set the GitLab Project Variables with your AWS credentials
5. Run a GitLab pipeline to deploy the API