RESTful APIs (Representation State Transfer)

> **❶ Warning**
>
> This page is work-in-progress. This is a proposed api which most likely is going to change.

REST

REST is an architectural style for designing loosely coupled applications over HTTP which was coined by Rob Fielding in 2000. We will define services according to REST.

All REST APIs should be versioned.

```
/api/v1/resource
```

Resources

A resources is a thing that the service knows about. A resource in a system should have only one logical URI and that should provide a way to fetch related or additional data.

Platform service

Data service to store data for apps, metadata for production services

Resources: Instance, Service, ServiceOwner, Schema, Model

Instance (Service instance)

An service instance is created when a instance onwer starts a workflow in an Altinn application. An instance replaces Altinn2 Message. An instanceOwner is a person/company that reports information via Altinn. An applicationId refers to the application information which defines the metadata about the service.

```
{
    "id": "762011d1-d341-4c0a-8641-d8a104e83d30",
    "applicationId": "KNS/sailor",
    "applicationOwnerId": "KNS",
    "instanceOwnerId": "666",
    "createdDateTime": "2019-03-06T13:46:48.6882148+01:00",
    "createdBy": "XXX",
    "lastChangedDateTime": "2019-03-07T23:59:49+01:00",
    "lastChangedBy": "XXX",
    "dueDateTime": null,
    "visibleDateTime": null,
    "presentationField": "Færder påmelding 2019",
    "externalSystemReference": null,
    "currentWorkflowStep": "started",
    "isCompleted": true,
    "isDeleted": [{
        "deletionDateTime": "2017-12-22",
        "deletedBy": "KNS"
    }],
    "applicationOwnerFeedback": {
        "receivedDate": "2019-05-11T03:00:23+01:00",
        "status": "OK"
    },
    "data": {
        "boatdata": {
            "762011d1-d341-4c0a-8641-d8a104e83d30": {
                "contentType": "application/json",
                "storageUrl": "KNS/sailor/762011d1-d341-4c0a-8641-d8a104e83d30/data/boatdata/992011d1-d341-4c0a-8641-d8a104e83d30",
                "fileName": "davidsyacht.json",
                "createdDateTime": "2019-03-06T15:00:23+01:00",
                "createdBy": "XXX",
                "signature": "oajviojoi2j3l23889yv8js909u293840zz092u3",
                "fileSize": 2003,
                "isLocked": true
            },
            "crewlist": {
                "999911d1-d341-4c0a-8641-d8a104e83d30": {
```

```
                    "contentType": "text/xml",
                    "storageUrl": "KNS/sailor/762011d1-d341-4c0a-8641-d8a104e83d30/data/crewlist/999911d1-d341-4c0a-8641-
    d8a104e83d30",
                    "fileName": "crewLIst.xml",
                    "createdDateTime": "2019-03-07T23:59:49+01:00",
                    "createdBy": "XXX",
                    "lastChangedDateTime": "2019-03-10T23:59:49+01:00",
                    "lastChangedBy": "XXX"
                }
            }
        }
    }
```

Create a new instance. Post with params that identifies the service. ServiceOwnerId is transmitted as a part of the token.

```
/instances?serviceId=KNS/sailor
```

Get information about an instance.

```
/instances/{instanceId}
```

Get all instances that a instance owner has completed

```
/instances[?services={serviceId}][&reporter={instanceOwnerId}][&completed=true][&since=2017-01-01]
```

Get a specific data element

```
/instances/{instanceId}/data/{dataId}
```

Post/put/delete a specific data element

```
/instances/{instanceId}/data/{dataId}
```

Delete a specific instance (also deletes its data).

```
/instances/{instanceId}
```

ServiceOwner

Get metadata about a service owner

```
/serviceowners/{serviceOwnerId}
```

Get all services of a spesific service owner

```
/serviceowners/{serviceOwnerId}/services
```

Service (metadata)

Resource: http://platform.altinn.no/service/sailor

```
{
    "id": "KNS/sailor",
    "createdDateTime": "2019-03-06T13:46:48.6882148+01:00",
    "createdBy": "XXX",
    "title": "Færder påmelding",
    "type": "innsending",
    "serviceOwnerId": "KNS",
    "workflowId": "standard",
    "isDeleted": false,
    "isArchived": false,
    "validFrom": null,
    "validTo": null,
    "forms": {
        "boatdata": {
            "contentType": "application/schema+json",
            "storageUrl": "sailor/schema/boatdata",
            "createdDateTime": "2019-03-04T12:01:00+01:00",
            "createdBy": "M2",
            "signatureRequired": true,
            "shouldEncryptData": true
        },
        "crewlist": {
            "contentType": "application/xsd+xml",
            "storageUrl": "sailor/schema/crewlist",
            "createdDateTime": "2019-03-04T12:01:00+01:00",
            "createdBy": "M2",
            "lastChangedDateTime": "2019-03-10T23:59:49+01:00",
            "lastChangedBy": "M42"
        }
    }
}
```

Get a list of all services

```
/services
```

Get metadata about a specific service

```
/services/{serviceId}
```

Get the schema of a specific form element in a service

```
/services/{serviceId}/forms/{dataId}?format=jsonSchema
```

Altinn-studio

...

tbd

App services (runtime)

tb changed

One cluster per service owner. A service owner can have many apps.

```
https://{serviceownerNick}.apps.altinn.no
```

Resources: App, Service, Instance

Get list of available services.

`/services`

Post to create an instance for a specific instanceOwner. Returns a new instanceId.

`/services/{serviceId}/instances`

Filter instances for an instance owner

`/services/{serviceId}/instances?instanceOwner={instanceOwnerId}&[since="2018-07-01"]`

Put to update a specific instance. Get to get instance metadata

`/services/{serviceId}/instances/{instanceId}`

Post/put to upload/update form data

`/services/{serviceId}/instances/{instanceId}/forms/{formId}`

Put/post to change workflow step

```
/services/{serviceId}/instances/{instanceId}/workflow/{stepId}
```

Get receipt

```
/services/{serviceId}/instances/{instanceId}/receipt
```

Get validate model

```
/services/{serviceId}/instances/{instanceId}/forms/{formId}/validate
```

Get metadata about a specific service

```
/services/{serviceId}
```

Get schema for a specific form in the service

```
/services/{serviceId}/schemas/{schemaId}
```

Get texts for a specific form in the service

`/services/{serviceId}/texts`