

DID Phase 2 Test API (1.0)

Download OpenAPI specification:

[Download](#)

API Support: support@swagger.io | URL: <http://www.swagger.io/support> | License: Apache 2.0 | [Terms of Service](#)

This is a sample server for DID ServiceEndpoint.

Create DID

Create a DID using method and public key.

AUTHORIZATIONS:

ApiKeyAuth

REQUEST BODY SCHEMA: application/json

required

Create DID request

```
method      string  
public_key_base58  string
```

Responses

› **200 ok** example({ "did": "did:byd50:1234567890abcdef" })

› **400 bad request** example({ "code": "INVALID_PARAM", "message": "method and public_key_base58 are required" })

› **500 internal error** example({ "code": "INTERNAL_ERROR", "message": "failed to create did" })

POST /testapi/create-did



Request samples

[Payload](#)

Content type

application/json

Copy

```
{  
  "method": "byd50",  
  "public_key_base58": "3VZ6oJdR8i1qKX7kH3Yv9d7w7wzgn..."  
}
```

Response samples

200

400

500

Copy

Content type

application/json

```
{  
  "did": "did:byd50:1234567890abcdef"  
}
```

Get DID Public Key

Resolve a DID and return its public key (Base58).

AUTHORIZATIONS: >

ApiKeyAuth

PATH PARAMETERS

some_id string
required

Responses

> 200 ok" example({"publicKeyBase58":"3VZ6oJdR8i1qKX7kH3Yv9d7w7wzgn..."})

> 400 bad request" example({"code":"INVALID_PARAM","message":"did is required"})

> 404 not found" example({"code":"NOT_FOUND","message":"public key not found"})

GET /testapi/get-did-public-key/{some_id}

▼

Response samples

200

400

404

Content type

application/json

Copy

```
{  
    "publicKeyBase58": "3VZ6oJdR8i1qKX7kH3Yv9d7w7wzgn..."  
}
```

Get DID Document

Resolve a DID and return its DID Document.

AUTHORIZATIONS: >

ApiKeyAuth

PATH PARAMETERS

→ `some_id` string
required
DID

Responses

> **200 ok** example({ "document": "...did document json..." })

> **400 bad request** example({ "code": "INVALID_PARAM", "message": "did is required" })

> **404 not found** example({ "code": "NOT_FOUND", "message": "did document not found" })

GET /testapi/get-did/{some_id}



Response samples

200

400

404

Content type

application/json

Copy

```
{  
  "document": "{\\\"@context\\\": [\\\"https://www.w3.org/ns/did/v1\\\"], \\\"id\\\": \\\"  
}
```

Create VC

Create a Verifiable Credential (JWT) signed by the issuer's private key.

AUTHORIZATIONS: > *ApiKeyAuth*

REQUEST BODY SCHEMA: application/json
required

Create VC request

```
+ credential_subject > object  
+ expires_in_minutes > integer  
+ issuer > string  
+ kid > string  
+ pv_key_base58 > string  
+ subject > string  
+ type > string
```

Responses

> **200 ok** example({ "vc_jwt": "eyJhbGciOi..." })

> **400**

bad request" example({ "code": "INVALID_PARAM", "message": "kid, pv_key_base58, and credential_subject are required" })

> **500 internal error** example({ "code": "INTERNAL_ERROR", "message": "failed to create vc" })

POST /testapi/vc/create



Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{  
    "credential_subject": { },  
    "expires_in_minutes": 5,  
    "issuer": "http://demo-issuer.example",  
    "kid": "did:byd50:1234567890abcdef",  
    "pv_key_base58": "3VZ6oJdR8i1qKX7kH3Yv9d7w7wzgn...",  
    "subject": "did:byd50:holder123",  
    "type": "AlumniCredential"  
}
```

Response samples

200

400

500

Content type

application/json

Copy

```
{  
    "vc_jwt": "eyJhbGciOiJFUzI1NiIsInR5cCI6IkpXVCJ9..."  
}
```

Verify VC

Verify a VC (JWT) using DID resolver for public key lookup.

AUTHORIZATIONS: >

ApiKeyAuth

REQUEST BODY SCHEMA: application/json

required

Verify VC request

vc_jwt

string

Responses

> 200 ok" example({"valid":true})

> 400 bad request" example({"code":"INVALID_PARAM","message":"vc_jwt is required"})

POST /testapi/vc/verify



Request samples

Payload

Content type

application/json

Copy

```
{  
    "vc_jwt": "eyJhbGciOiJFUzI1NiIsInR5cCI6IkpXVCJ9..."  
}
```

Response samples

200

400

Content type

application/json

Copy

```
{  
    "error": "signature invalid",  
    "valid": true  
}
```

Create VP

Create a Verifiable Presentation (JWT) from VC JWTs.

AUTHORIZATIONS: > *ApiKeyAuth*

REQUEST BODY SCHEMA: application/json
required

Create VP request

expires_in_minutes	integer
holder_did	string
issuer	string
pv_key_base58	string
subject	string
type	string
vc_jwts	Array of strings

Responses

› **200 ok** example({ "vp_jwt": "eyJhbGciOi..." })

› **400**

bad request" example({ "code": "INVALID_PARAM", "message": "holder_did, pv_key_base58, and vc_jwts are required" })

› **500 internal error** example({ "code": "INTERNAL_ERROR", "message": "failed to create vp" })

POST /testapi/vp/create ▾

Request samples

Payload

Content type

application/json

Copy Expand all Collapse all

```
{
  "expires_in_minutes": 5,
  "holder_did": "did:byd50:holder123",
  "issuer": "client make this vp",
  "pv_key_base58": "3VZ6oJdR8i1qKX7kH3Yv9d7w7wzgn...",
  "subject": "did:byd50:holder123",
  "type": "CredentialManagerPresentation",
  - "vc_jwts": [
    "string"
  ]
}
```

Response samples

200

400

500

Content type

application/json

Copy

```
{
  "vp_jwt": "eyJhbGciOiJFUzI1NiIsInR5cCI6IkpXVCJ9..."
}
```

Verify VP

Verify a VP (JWT) using DID resolver for public key lookup.

AUTHORIZATIONS: >

ApiKeyAuth

REQUEST BODY SCHEMA: application/json
required

Verify VP request

→ vp_jwt

string

Responses

> **200 ok** example({ "valid": true })

> **400 bad request** example({ "code": "INVALID_PARAM", "message": "vp_jwt is required" })

POST /testapi/vp/verify



Request samples

Payload

Content type

application/json

Copy

```
{  
    "vp_jwt": "eyJhbGciOiJFUzI1NiIsInR5cCI6IkpXVCJ9..."  
}
```

Response samples

200

400

Content type

application/json

Copy

```
{  
    "error": "signature invalid",  
    "valid": true  
}
```