



IdentityChain API ^{0.2.0}

IdentityChain Agent REST API

Schemes

HTTP

Authorize





POST

/api/login

Allow users to login, will return them a JWT valid for a specific amount of time

Allow users to login, will return them a JWT valid for a specific amount of time

Parameters

Try it out

Name	Description
data (body)	<div><div>Example Value</div><div>Model</div><div><pre>{ "username": "string", "password": "string" }</pre></div><div><div>Parameter content type</div><div>application/json</div></div></div>

Responses

Response content type

application/json

Code	Description
200	<div><div>Login successful</div><div><div>Example Value</div><div>Model</div><div><pre>{ "token": "string" }</pre></div></div></div>
400	<div><div>Bad Request</div><div><div>Example Value</div><div>Model</div></div></div>



POST

/api/login

Allow users to login, will return them a JWT valid for a specific amount of time

Allow users to login, will return them a JWT valid for a specific amount of time

Parameters

Try it out

Name	Description
data (body)	<div><div>Example Value</div><div>Model</div><div><pre>{ "username": "string", "password": "string" }</pre></div><div><div>Parameter content type</div><div>application/json</div></div></div>

Responses

Response content type

application/json

Code	Description
200	<div><div><i>Login successful</i></div><div><div>Example Value</div><div>Model</div><div><pre>{ "token": "string" }</pre></div></div></div>
400	<div><div><i>Bad Request</i></div><div><div>Example Value</div><div>Model</div><div><pre>{</pre></div></div></div>



POST **/api/login** Allow users to login, will return them a JWT valid for a specific amount of time

Allow users to login, will return them a JWT valid for a specific amount of time

Parameters

Try it out

Name

Description

data

(body)

Example Value Model

```
{
  "username": "string",
  "password": "string"
}
```

Parameter content type

application/json

Responses

Response content type

application/json

Code

Description

200

Login successful

Example Value Model

```
{
  "token": "string"
}
```

400

Bad Request

Example Value Model

```
{
```



POST

/api/login

Allow users to login, will return them a JWT valid for a specific amount of time

Allow users to login, will return them a JWT valid for a specific amount of time

Parameters

Try it out

Name	Description
data (body)	<div><div>Example ValueModel</div><div><pre>{ "username": "string", "password": "string" }</pre></div><div>Parameter content type</div><div>application/json</div></div>

Responses

Response content type

application/json

Code	Description
200	<div><div><i>Login successful</i></div><div><div>Example ValueModel</div><div><pre>{ "token": "string" }</pre></div></div></div>
400	<div><div><i>Bad Request</i></div><div><div>Example ValueModel</div><div><pre>{</pre></div></div></div>



POST **/api/login** Allow users to login, will return them a JWT valid for a specific amount of time

Allow users to login, will return them a JWT valid for a specific amount of time

Parameters

Try it out

Name	Description
data (body)	<div><div>Example Value Model</div><div><pre>{ "username": "string", "password": "string" }</pre></div><div>Parameter content type</div><div>application/json</div></div>

Responses

Response content type

application/json

Code	Description
200	<div><div><i>Login successful</i></div><div><div>Example Value Model</div><div><pre>{ "token": "string" }</pre></div></div></div>
400	<div><div><i>Bad Request</i></div><div><div>Example Value Model</div><div><pre>{</pre></div></div></div>



GET

/api/wallet

List all wallets of user

List all wallets of user

Parameters

Try it out

No parameters

Responses

Response content type

application/json

Code	Description
200	<div>Success</div> <div><div>Example Value</div><div>Model</div><pre>[{ "id": "string", "createdAt": "string", "updatedAt": "string", "owner": "string", "users": ["string"], "credentials": { "key": "string" }, "ownDid": "string" }]</pre></div>

connection Connection offer/request/response endpoints



GET

/api/connection/{myDid}

Retrieve a (pending) connection

Retrieve a (pending) connection

Parameters

Try it out

Name	Description
wallet string (header)	A unique id value identifying a wallet.
myDid * required string (path)	A unique did used in a pairwise

Responses

Response content type

application/json

Code	Description
200	<div>Example Value Model</div> <div><pre>{ "myDid": "string", "theirDid": "string", "acknowledged": true}</pre></div>
401	Unauthorized
404	Not Found

Example Value Model



GET

/api/indyschema

List low-level indy schemas

List low-level indy schemas

Parameters

Try it out

Name	Description
wallet string (header)	A unique id value identifying a wallet.

Responses

Response content type

application/json

Code	Description
200	<div>Success</div> <div><div>Example Value</div><div>Model</div><div>[{ "name": "string", "createdAt": "string", "updatedAt": "string", "attrNames": ["string"], "version": "string", "wallet": "string", "schemaId": "string", "data": { "id": "string", "attrNames": ["string"], "name": "string", "version": "string", "ver": "string" } }]</div></div>



GET /api/schema List schemas



List schemas from the logged in user

Parameters

Try it out

Name	Description
onlyActive boolean (query)	(Optional) Flag to filter all non-active schemas
wallet string (header)	A unique id value identifying a wallet.

Responses

Response content type

application/json

Code	Description
200	<div><div>Example Value</div><div>Model</div><pre>[{ "name": "Passport", "version": "1.0", "createdAt": "2018-10-21T18:27:23.759Z", "schemaId": "Th7MpTaRZVRynPiabds81Y:2:Passport:1.0" }, { "name": "ID_NL", "version": "2.0", "createdAt": "2018-09-21T18:27:23.759Z", "schemaId": "Th7MpTaRZVRynPiabds81Y:2:ID_NL:2.0" }, { "name": "Passport", "version": "2.0", "createdAt": "2018-08-21T18:27:23.759Z",</pre></div>



GET

/api/credentialdef

List credential definitions of wallet stored in DB

List credential definitions of wallet stored in DB

Parameters

Try it out

Name	Description
wallet	A unique id value identifying a wallet.
string	
(header)	

Responses

Response content type

application/json

Code	Description
200	Success

Example Value

Model

```
[
  {
    "created": "string",
    "credDefId": "string",
    "revocRegId": "string",
    "revocRegType": "string",
    "wallet": "string",
    "data": {
      "id": "string",
      "schemaId": "string",
      "type": "string",
      "tag": "string",
      "value": {
        "primary": "string",
        "revocation": "string"
      },
      "ver": "string"
    }
  }
]
```



GET

/api/proofrequesttemplate

List proof request templates

List proof request templates

Parameters

Try it out

Name	Description
wallet string (header)	A unique id value identifying a wallet.

Responses

Response content type

application/json

Code	Description
200	<div>Success</div> <div><div>Example Value</div><div>Model</div><div>[{ "wallet": "string", "createdAt": "string", "updatedAt": "string", "template": "string" }]</div></div>
401	<div>Unauthorized</div>
404	<div>Not Found</div> <div><div>Example Value</div><div>Model</div></div>

transaction Low-level indy transaction endpoints



GET

/api/transactions

Get ledger transactions from-to indexes

Parameters

Try it out

Name	Description
wallet string (header)	A unique id value identifying a wallet.
from * required number (query)	Sequence Number of first transaction to query (greater than 0)
to * required number (query)	Sequence Number of last transaction to query (greater than from index)
type string (query)	Ledger Type: pool, domain, config. By default is DOMAIN

Responses

Response content type

application/json

Code	Description
200	Returns list of transactions

Example Value

Model

```
[
  {
    "txn": {
      "data": {
```



POST **/indy/** Agent Message Endpoint

Agent Message Endpoint

Parameters

Try it out

Name	Description
data (body)	<div><div>Example Value Model</div><div><pre>{ "message": "string" }</pre></div><div>Parameter content type</div><div>application/json</div></div>

Responses

Response content type

application/json

Code	Description
202	<i>Accepted</i>
400	<i>Bad Request</i>
	<div><div>Example Value Model</div><div><pre>{ "message": "string" }</pre></div></div>
404	<i>Not Found</i>
	<div><div>Example Value Model</div></div>

walletId `string`

(Optional) A unique id value identifying a wallet

id `string`

A uniquely identifying string value

createdAt `string`

Creation Time

updatedAt `string`

Last update time

messageId `string`

Sent message id, could be a nonce depending on message format

messageType `string`

Message Type

Enum:

```
[ urn:sovrin:agent:message_type:sovrin.org/connection_offer,
urn:sovrin:agent:message_type:sovrin.org/connection_request,
urn:sovrin:agent:message_type:sovrin.org/connection_response,
urn:sovrin:agent:message_type:sovrin.org/connection_acknowledge,
urn:sovrin:agent:message_type:sovrin.org/credential_offer,
urn:sovrin:agent:message_type:sovrin.org/credential_request,
urn:sovrin:agent:message_type:sovrin.org/credential,
urn:sovrin:agent:message_type:sovrin.org/proof_request,
urn:sovrin:agent:message_type:sovrin.org/proof ]
```

senderDid `string`

Sender Did

recipientDid `string`
Recipient Did

expireAt `string`
Optional expire time

```
user {  
  id string  
    A unique string value identifying a user  
  username string  
    Username  
  wallet string  
    Id of user's current default wallet  
}
```

```
wallet {  
  id string  
    A unique string value identifying a wallet  
  createdAt string  
    wallet creation time  
  updatedAt string  
    wallet update time  
  owner string  
    Wallet Owner User Id  
  users [  
    Array of user Ids which are allowed to use this  
    wallet  
    string]  
  credentials {  
    key string  
      Wallet Key  
  }  
  ownDid string  
    Wallet primary did  
}
```



```

user_post {
    username*
    password*
    wallet

    string
    string
    wallet_post {
        name
        string
        (Optional) Wallet name, must
        be globally unique.
        Automatically generated if
        none is provided.

        credentials*
        {
            description: Wallet
            Credentials
            JSON.
            Supported
            keys vary
            by wallet
            type. A
            default
            config will
            be used if
            none is
            provided.
            key*
            string
            Passphrase
            used to
            derive
            wallet
            master key
        }
        seed
        string
        (Optional) Seed to use for
        initial did creation.
    }
}

```

```

user_put {
    username
    password
    wallet

    string
    (Optional) New username

    string
    (Optional) New password

    walletId string
    (Optional) A unique id value identifying a wallet
}

```

```
wallet_post {
  name
    string
    (Optional) Wallet name, must be globally unique.
    Automatically generated if none is provided.

  credentials*
    {
      description: Wallet Credentials JSON.
        Supported keys vary by wallet
        type. A default config will
        be used if none is provided.
      key*
        string
        Passphrase used to derive
        wallet master key
    }

  seed
    string
    (Optional) Seed to use for initial did creation.
}
```

```
credential_post {
  wallet
    walletId string
      (Optional) A unique id value identifying a wallet

  credentialRequestId* string
    Credential Request _id as stored in API

  values*
    {
      description:      Object containing attributes
                        defined in schema as key-
                        value pairs (e.g.
                        {"attrName": "attrValue",
                        "attrName1": "attrValue1"})
    }
}
```

```
credentialdef_post {
    wallet walletId string
                (Optional) A unique id value identifying a wallet

    schemaId* string
                A unique string value identifying a schema

    tag string
                default: TAG1
                A string value identifying credential definition
                uniquely within given schema

    supportRevocation boolean
                (Optional) States if revocation should be supported
                for this credential definiton
}
```



```

connectionrequest_post  {
    wallet                walletId string
                          (Optional) A unique id value identifying a wallet

    endpoint              string
                          (Optional) Response endpoint for connection response
                          - my endpoint

    theirDid              string
                          (Optional) Their (endpoint) did, required if there is
                          no connection offer

    theirVk               string
                          (Optional) Their (endpoint) did verification key,
                          required if there is no connection offer or verkey
                          not on ledger.

    theirEndpoint         string
                          (Optional) Their endpoint address, required if no
                          there is no connection offer or address not on
                          ledger.

    connectionOffer       {
                          description:      (Optional) A connection offer
                                          object
                          }
}

```

```

connectionresponse_post  {
    wallet                walletId string
                          (Optional) A unique id value identifying a wallet

    connectionRequestId* string
                          example: 5b7ddce0000000000000000000000000
                          A unique string value identifying the connection
                          request to respond to
}

```

```

proofrequest_template_post  {
    wallet                walletId string
                          (Optional) A unique id value identifying a wallet

    template*            string
                          example: { "name": "Ticket", "version": "0.1",
                          "requested_attributes": { "attr1_referent": { "name":
                          "firstname", "restrictions": [ { "cred_def_id": "{
                          cred_def_id }}" } ] }, "attr2_referent": { "name":
                          "phone" } }, "requested_predicates": {} }
                          proof request template (optionally with mustache
                          placeholders)
}

```

```
proofrequest_template_put {
  wallet
    wallet {
      id string
      A unique string value
      identifying a wallet
      createdAt string
      wallet creation time
      updatedAt string
      wallet update time
      owner string
      Wallet Owner User Id
      users [
      Array of user Ids which are
      allowed to use this wallet
      string]
      credentials {
        key string
        Wallet
        Key
      }
      ownDid string
      Wallet primary did
    }
  template* string
  example: { "name": "Ticket", "version": "0.1",
  "requested_attributes": { "attr1_referent": { "name":
  "firstname", "restrictions": [ { "cred_def_id": "{ {
  cred_def_id } }" }] }, "attr2_referent": { "name":
  "phone" } }, "requested_predicates": {} }
  proof request template (optionally with mustache
  placeholders)
}
```

```

proofrequest_post {
  wallet          walletId string
                  (Optional) A unique id value identifying a wallet

  recipientDid*   string
                  DID for whom to create a proof request

  proofRequest*   {
                    description:    proof request template _id or
                                    proof request object (see
                                    https://github.com/hyperledger/indy-sdk/blob/master/doc/getting-started/getting-started.md#apply-for-a-job)
                  }

  templateValues {
                    description:    values to use for rendering
                                    the proof request template as
                                    key-value object
                  }
}

```

```

proof_post {
  wallet          walletId string
                  (Optional) A unique id value identifying a wallet

  proofRequestId* string
                  proof request _id

  values          {
                    description:    (Optional) Object containing
                                    self-attested-attributes as
                                    key-value pairs
                  }
}

```

```

indyschema_post {
  wallet          walletId string
                  (Optional) A unique id value identifying a wallet

  name*           string
                  A unique name of the schema

  version*        string
                  version for schema

  attrNames*     [
                  list of attribute names put into the schema
                  string]
}

```

```

schema_post  {
    name*      string
                A unique name for the schema

    version*   string
                Schema version

    parentSchemaId string
                Schema Id of the parent

    createCredentialDefinition boolean
                default: true

    isRevocable boolean
                default: false
                Flag to define if credentials issued with this
                schema could be revoked. If true it will
                generate one or multiple Revocation Registries.

    attributes* [
                List of attribute names put into the schema
                {
                    name*      string
                        Name of the attribute
                        (Key)

                    type*      string
                        A valid type for
                        attributes, as requested
                        using GET /types in
                        schema compiler API
                }
            ]
}

```

```

schema_details    {
    name*           string
                    A unique name for the schema
    version*        string
                    Schema version
    parentSchemaId  string
                    Schema Id of the parent
    createCredentialDefinition boolean
                    default: true
    isRevocable      boolean
                    default: false
                    Flag to define if credentials issued with this
                    schema could be revoked. If true it will
                    generate one or multiple Revocation Registries.

    attributes*     [
                    List of attribute names put into the schema
                    {
                        name*           string
                                    Name of the attribute
                                    (Key)

                        type*           string
                                    A valid type for
                                    attributes, as requested
                                    using GET /types in
                                    schema compiler API
                    }
                ]
    isDeprecated    boolean
    lowLevelSchema  {
    }
    credentialDefinitionId string
    revocationRegistryId  string
}

```

```

message_post    {
    wallet          walletId string
                    (Optional) A unique id value identifying a wallet

    did*            string
                    Recipient DID
    message*        string
                    authcrypted, base64-encoded message to send
}

```

```

indy_post       {
    message*        string
                    Anoncrypted-message
}

```



```

login_post  {
    username*      string
    password*      string
}

nym_post    {
    wallet          walletId string
                  (Optional) A unique id value identifying a wallet

    did*           string
                  Did to write on the ledger

    verkey*        string
                  Verkey of did to write on the ledger

    alias          string
                  (Optional) Alias of did to write on the ledger

    role           string
                  (Optional) role of did to write on the ledger
                  (default = NONE)

    Enum:
        [ NONE, TRUSTEE, STEWARD, TRUST_ANCHOR ]
}

error_message {
    message      string
                  Informational Message
}

```

```

connection_offer {
  id          id string
              A uniquely identifying string value
  createdAt   createdAt string
              Creation Time
  updatedAt   updatedAt string
              Last update time
  expireAt    expireAt string
              Optional expire time
  messageId   messageId string
              Sent message id, could be a nonce depending on message format

  type        string
              urn:sovrin:agent:message_type:sovrin.org/connection_offer

  senderDid   senderDid string
              Sender Did
  recipientDid recipientDid string
              Recipient Did
  message     {
              {
                id      string
                    Offer Nonce
                type    string
                    urn:sovrin:agent:message_type:sovrin.org/connection_offer

                message {
                    {
                      did      string
                          Sender's Endpoint DID

                      verkey    string
                          (Optional) Sender's Endpoint DID
                          Verification Key, required if
                          not on ledger

                      endpoint  string
                          (Optional) Sender's Endpoint DID
                          endpoint Address, required if
                          not on ledger

                      nonce     string
                          Offer Nonce

                      data      {
                          {
                            description:  Object
                                          containing
                                          additional
                                          information
                                          about sender
                                          such as
                                          name,
                                          description,
                                          logo, etc.
                          }
                        }
                    }
                }
              }

  meta {
    {
      description:  Additional Metadata
      role         string
                    (Optional) Role which is offered

      myDid        string
    }
  }
}

```

```

connection_request {
  id          id string
              A uniquely identifying string value
  createdAt   createdAt string
              Creation Time
  updatedAt   updatedAt string
              Last update time
  expireAt    expireAt string
              Optional expire time
  messageId   messageId string
              Sent message id, could be a nonce depending on message
              format

  type        string
              urn:sovrin:agent:message_type:sovrin.org/connection_request

  senderDid   senderDid string
              Sender Did
  recipientDid recipientDid string
              Recipient Did
  message     {
  }
  meta        {
    description:    Additional Metadata
    role             string
                      (Optional) Role which was offered
                      (only on offering side)

    myDid            string
                      User's pairwise did

    theirEndpointDid string
                      (Only on sender side) Recipient's
                      endpoint did

    theirEndpointVk  string
                      (Only on sender side) Recipient's
                      endpoint did verification key

    theirEndpoint    string
                      (Only on sender side) Recipient's
                      endpoint address
  }
}

```

```

connection_response {
}

```

```

credential_offer {
  id          id string
              A uniquely identifying string value
  createdAt   createdAt string
              Creation Time
  updatedAt   updatedAt string
              Last update time
  expireAt    expireAt string
              Optional expire time
  messageId   messageId string
              Sent message id, could be a nonce depending on message
              format

  type        string
              urn:sovrin:agent:message_type:sovrin.org/credential_offer

  senderDid   senderDid string
              Sender Did
  recipientDid recipientDid string
              Recipient Did
  message     {
  }
  meta        {
    description:      Additional Metadata
    credentialLocation  string
                      (Optional) URL location of
                      credential values to
                      automatically issue credentials
                      on request
  }
}
}

```

```

indy_schema_db {
  description:      Indy low-level schema as stored in DB

  name              string
                  Schema Name

  createdAt          createdAt string
                  Creation Time

  updatedAt          updatedAt string
                  Last update time

  attrNames         [string
                  Schema attribute name
                  ]

  version            string
                  Schema Version

  wallet             string
                  Owner Wallet Id

  schemaId           string
                  Schema Id

  data              indy_schema {
                    description:      Schema as created by indy-sdk
                                      and stored on the ledger

                    id                string
                                      Schema Id

                    attrNames         [string
                                      Schema attribute name
                                      ]

                    name              string
                                      Schema Name

                    version            string
                                      Schema Version

                    ver                string
                                      Schema JSON Version
                  }
}

```

```

indy_schema {
  description:      Schema as created by indy-sdk and stored on the
                    ledger

  id                string
                  Schema Id

  attrNames         [string
                  Schema attribute name
                  ]

  name              string
                  Schema Name

  version            string
                  Schema Version

  ver                string
                  Schema JSON Version
}

```

```
credential_definition {
  created      createdAt string
               Creation Time
  credDefId    string
               Credential Definition Id
  revocRegId   string
               (Optional) Revocation Registry Id
  revocRegType string
               (Optional) Revocation Registry Type
  wallet       string
               Owner Wallet id
  data         indy_credential_definition {
               id string
               A uniquely identifying string
               value
               schemaId string
               Schema Id
               type string
               Credential Definition Type
               tag string
               Credential Definition Tag
               value {
                 primary string
                       Primary
                       Credential
                       Public Key
                 revocation string
                       (Optional)
                       Revocation
                       Credential
                       Public Key
               }
               ver string
               Credential Definition Version
             }
}
```

```

indy_credential_definition {
  id          id string
              A uniquely identifying string value
  schemaId    string
              Schema Id
  type        string
              Credential Definition Type
  tag         string
              Credential Definition Tag
  value       {
              {
                primary      string
                            Primary Credential Public Key
                revocation    string
                            (Optional) Revocation
                            Credential Public Key
              }
  ver         string
              Credential Definition Version
}

```

```

indy_credential {
  description:      Credential as created by indy-sdk, fields may vary

  referent           string
                    Credential Id as stored in wallet
  schema_id          string
                    Schema Id
  cred_def_id        string
                    Credential Definition Id
  rev_reg_def_id     string
                    (Optional) Revocation Registry Definition Id,
                    if revocable

  values             {
                    {
                      description:      Credential Values
                    }
  signature           string
                    Signature
  signature_correctness_proof string
                    Signature Correctness Proof
}

```

```

credential_request {
  id          id string
              A uniquely identifying string value
  createdAt   createdAt string
              Creation Time
  updatedAt   updatedAt string
              Last update time
  expireAt    expireAt string
              Optional expire time
  messageId   messageId string
              Sent message id, could be a nonce depending on message
              format

  type        string
              urn:sovrin:agent:message_type:sovrin.org/credential_request

  senderDid   senderDid string
              Sender Did
  recipientDid recipientDid string
              Recipient Did
  message     {
  }
  meta        {
              description:      Credential request metadata JSON
              as created by indy-sdk
  }
}

```



```
credential_message {
  id string
  A uniquely identifying string value
  createdAt string
  Creation Time
  updatedAt string
  Last update time
  expireAt string
  Optional expire time
  messageId string
  Sent message id, could be a nonce depending on
  message format
  type string
  urn:sovrin:agent:message_type:sovrin.org/credential
  senderDid string
  Sender Did
  recipientDid string
  Recipient Did
  message {
  }
  meta {
    description: Additional Metadata
    revocRegId string
    (Optional) Revocation
    Registry Id
    credRevocId string
    (Optional) Credential
    Revocation Id
    revocRegDelta string
    (Optional) Revocation
    Registry Delta
  }
}
```

```

proofrequest_message {
  id      id string
          A uniquely identifying string value
  createdAt createdAt string
          Creation Time
  updatedAt updatedAt string
          Last update time
  expireAt expireAt string
          Optional expire time
  messageId messageId string
          Sent message id, could be a nonce depending on message format

  type      string
          urn:sovrin:agent:message_type:sovrin.org/proof_request

  senderDid senderDid string
          Sender Did
  recipientDid recipientDid string
          Recipient Did
  message {
    id      string
            Request Nonce
    origin string
            Sender's pairwise did

    type      string
            urn:sovrin:agent:message_type:sovrin.org/proof_request

    message {...}
  }

  meta {
    description:      Additional Metadata
    proofId      string
                  Proof placeholder Id which can be
                  queried for proof status
  }
}

```

```

proof_message {
  id          id string
              A uniquely identifying string value
  createdAt   createdAt string
              Creation Time
  updatedAt   updatedAt string
              Last update time
  expireAt    expireAt string
              Optional expire time
  messageId   messageId string
              Sent message id, could be a nonce depending on message
              format

  type        string
              urn:sovrin:agent:message_type:sovrin.org/proof

  senderDid   senderDid string
              Sender Did
  recipientDid recipientDid string
              Recipient Did
  message     {
              id          string
                  Request Nonce
              origin      string
                  Sender's pairwise did

              type        string
                  urn:sovrin:agent:message_type:sovrin.org/proof

              message     {
                  description:    Proof as created by
                                indy-sdk
              }
            }
}

```

```
proof {
    wallet      string
                Recipient wallet Id
    did         string
                Proof Sender/Prover Pairwise Did

    proof       {
                    description:      Proof as created by indy-sdk
                    and sent by prover
                }

    status      string
                Proof status (pending or received)

    Enum:
        [ pending, received ]

    meta        {
                    description:      Additional Metadata
                proofRequest
                    {
                        description:      Proof
                        request
                        object,
                        necessary
                        for
                        further
                        processing
                    }
                }

    isValid     boolean
                Whether proof is valid or not
}
```

```
pairwise {
  myDid      string      My pairwise DID
  theirDid   string      Their pairwise DID
  metadata   {
    description:       Additional Metadata, may include other things
    theirEndpointDid    string      Their endpoint did
    theirEndpointVk      string      Their endpoint did verification key
    theirEndpoint        string      Their endpoint address
    acknowledged         boolean     Whether this connection was acknowledged
  }
}
```

```
proofrequest_template  {
    wallet              string
                        Owner wallet Id
    createdAt           createdAt string
                        Creation Time
    updatedAt           updatedAt string
                        Last update time
    template            string
                        Proof request mustachejs template string
}
```

```
indy_transaction  {
    description:        Indy Transaction as retrieved from ledger
}
```

```
indy_nym_result   {
    description:        Indy NYM Result
}
```

```
indy_nym          {
    description:        Indy NYM transaction
}
```

```

message_db {
  description: Message Object as stored in DB
  id          id string
              A uniquely identifying string value
  createdAt   createdAt string
              Creation Time
  updatedAt   updatedAt string
              Last update time
  expireAt    expireAt string
              Optional expire time
  messageId   messageId string
              Sent message id, could be a nonce depending on message format

  type        messageType string
              Message Type
              Enum:
                [ urn:sovrin:agent:message_type:sovrin.org/connection_offer,
                  urn:sovrin:agent:message_type:sovrin.org/connection_request,
                  urn:sovrin:agent:message_type:sovrin.org/connection_response,
                  urn:sovrin:agent:message_type:sovrin.org/connection_acknowledge,
                  urn:sovrin:agent:message_type:sovrin.org/credential_offer,
                  urn:sovrin:agent:message_type:sovrin.org/credential_request,
                  urn:sovrin:agent:message_type:sovrin.org/credential,
                  urn:sovrin:agent:message_type:sovrin.org/proof_request,
                  urn:sovrin:agent:message_type:sovrin.org/proof ]
  senderDid   senderDid string
              Sender Did
  recipientDid recipientDid string
              Recipient Did
  message      {
                  description: Message which is sent to recipient, see
                  message_formats

                }

  meta         {
                  description: Additional Metadata

                }
}

```