## HTTP interface to CasperExplorer server (v2)

Black text should/will appear as is; orange text should/will be substituted with values of the specified type.

get /data

Read current state of the app from server.

| request body | response body |
| --- | --- |
|  | {  accounts: [  { // zero or more  name: string,  publicKey: string  }  ],  payment: [  string // zero or more  ],  session: [  string // zero or more  ]  } |

Errors

None

post /account

Create key pair on server, create account identified by “name” (and the public key of the newly generated key pair), and initialize its balance to “balance”.

| request body | response body |
| --- | --- |
| {  name: string,  balance: int  } | {  status: boolean,  message: string,  publicKey: string  } |

Errors

* Malformed request body
* Error emitted by node in response to account creation deploy

delete /account

Delete account identified by “name”. This only removes the account data from the persistent store of the server – it does not delete the account on the node (which cannot be done yet).

| request body | response body |
| --- | --- |
| {  name: string  } | {  status: boolean,  message: string  } |

Errors

* Malformed request body
* Account with specified name does not exist

put /contract

Store new contract in the contract library.

| request body | response body |
| --- | --- |
| {  name: string,  type: string, // “payment”,“session”  wasm: string // base64 encoded  } | {  status: boolean,  message: string  } |

Errors

* Malformed request body
* Invalid “type” value

delete /contract

Delete contract identified by “name” from the contract library.

| request body | response body |
| --- | --- |
| {  name: string,  type: string // “payment”,“session”  } | {  status: boolean,  message: string  } |

Errors

* Malformed request body
* Invalid “type” value
* Contract with specified type and name does not exist

post /contract

Deploy specified contracts to a node.

| request body | response body |
| --- | --- |
| {  account: string,  payment: string,  paymentArgs: string,  session: string,  sessionArgs: string,  propose: boolean  } | {  status: boolean,  message: string  } |

Errors

* Malformed request body
* Error emitted by node in response to deploy

get /query

Query state of the DAG.

| request body | response body |
| --- | --- |
| {  blockHash: string,  keyVariant: string, // enum  keyBytes: string,  path: string  } | {  result: string  } |

Errors

* Malformed request body
* Error emitted by node in response to query