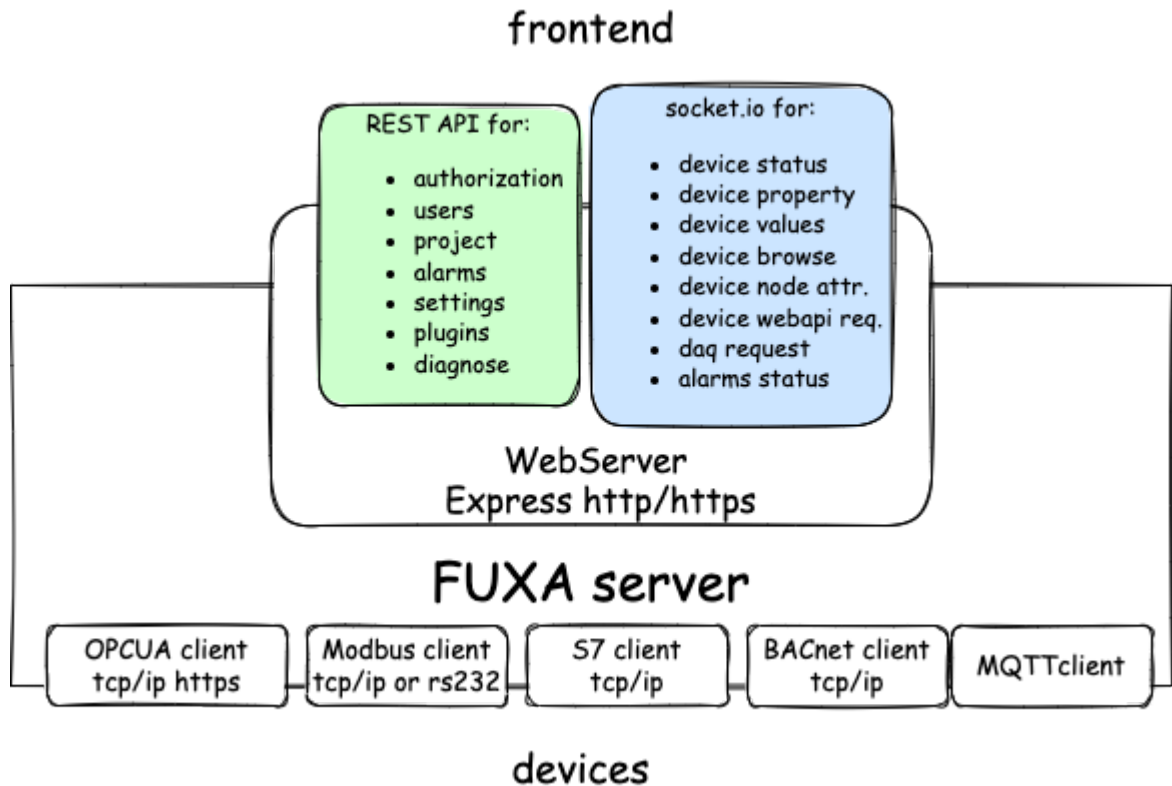


FUXA Web API



Get application settings	
GET	/api/settings
Parameters	None
Responses Example	Code: 200 <pre>{ "language": "en", "uiPort": 1881, "secureEnabled": false, "tokenExpiresIn": "1h", "smtp": {...}, }</pre>

Set application settings	
POST	/api/settings
Parameters Example	<pre>{ "language": "en", "uiPort": 1881, "secureEnabled": false, "tokenExpiresIn": "1h", }</pre>

	<pre> "smtp": {...} }</pre>
Responses	Code: 200
Note	Saved as file in 'server/_appdata/mysettings.json'

Get project property: server, devices, layout, views, charts, alarms settings, script	
GET	/api/project
Parameters	None
Responses	Code: 200 <pre>{ "version": "1.00", "server": {...}, "hmi": { "layout": {...}, "views": [...] }, "devices": {...}, "charts": [...], "graphs": [...], "alarms": [...], "notifications": [...], "scripts": [...], "texts": [...], "plugin": [...] }</pre>

Set project property	
POST	/api/project
Parameters Example	<pre>{ "version": "1.00", "server": { }, "hmi": { "layout": {...}, "views": [{"MainView": {...}] }, "devices": {}, "charts": [...], "graphs": [...], "alarms": [...], "notifications": [...], "scripts": [...], "texts": [...], "plugin": [...] }</pre>
Responses	Code: 200
Note	Used to build new project, clear all and set a new void project

Set project specific property: device, view, layout, charts, alarm, text	
POST	/api/projectData
Parameters Example	<pre>{ "cmd": "", "data": ... }</pre> <p>cmd: 'set-device', 'del-device' data: {...}</p> <p>cmd: 'set-view', 'del-view' data: {...}</p> <p>cmd: 'layout' data: {...}</p> <p>cmd: 'charts' data: [...]</p> <p>cmd: 'graphs' data: [...]</p> <p>cmd: 'set-text', 'del-text' data: {...}</p> <p>cmd: 'set-alarm', 'del-alarm' data: {...}</p> <p>cmd: 'set-notification', 'del-notification' data: {...}</p> <p>cmd: 'set-script', 'del-script' data: {...}</p>
Responses	Code: 200
Note	<p>In server are saved in tables: general, views devices, texts, alarms, notifications, scripts. The tables structure: name (string), value (string) all the same:</p> <ul style="list-style-type: none"> • general name=charts/layout/plugin/version and value=data • views name=data.id (view) and value=data • devices name=data.id (device) and value=data • alarms name=data.name (alarm) and value=data • notifications name=data.id (notification) and value=data • scripts name=data.id (script) and value=data

Upload project	
POST	/api/upload
Parameters Example	JSON file
Responses	Code: 200

Get device security property	
GET	/api/device
Parameters Example	<pre>{ "query": "security", "name": "My OPCUA" }</pre>
Responses	Code: 200 <pre>{ "mode": "", "uid": "username", "pwd": "password", "clientId": "123456" }</pre>

Set device security property	
POST	/api/device
Parameters Example	<pre>{ "query": "security", "name": "My OPCUA", "value": { "mode": "", "uid": "username", "pwd": "password", "clientId": "123456" } }</pre>
Responses	Code: 200

Get active alarms list	
GET	/api/alarms
Parameters	None
Responses	Code: 200 <pre>[{ "name": "Alarm name", "type": "", "ontime": "", "offtime": "", "acktime": "", "status": "", "text": "", "group": "", "bkcolor": "", "color": "", "toack": "" }]</pre>

	<pre> }] </pre>
--	------------------------------

Set ACK to active alarm	
POST	<code>/api/alarmack</code>
Parameters	<pre> { "params": "Alarm id" } </pre>
Responses	Code: 200
Note	'Alarm id' 'is composed from name and type

Login and take the token	
POST	<code>/api/signin</code>
Parameters	<pre> { "username": "admin", "password": "123456" } </pre>
Responses	Code: 200

Get users list	
GET	<code>/api/users</code>
Parameters	None
Responses	Code: 200 <pre> [{ "username": "username", "password": "123456", "groups": 3 }] </pre>

Set user property	
POST	<code>/api/users</code>
Parameters	<pre> { "username": "username", "fullname": "fullname", "password": "123456", "groups": 3 } </pre>

Responses	Code: 200
------------------	-----------

Remove user	
DELETE	/api/users
Parameters	{ "param": "username" }
Responses	Code: 200

Get installed plugin	
GET	/api/plugins
Parameters	None
Responses	Code: 200 [{ "name": "modbus-serial", "type": "Modbus", "version": "8.0.1", "current": "8.0.1", "status": "", "pkg": true, "dinamic": true }]

Install plugin	
POST	/api/plugins
Parameters	{ "name": "modbus-serial", "type": "Modbus", "version": "8.0.1", "current": "8.0.1", "status": "", "pkg": true, "dinamic": true }
Responses	Code: 200

Remove plugin	
DELETE	/api/plugins

Parameters	{ "param": "modbus-serial" }
Responses	Code: 200

Get DAQ data	
GET	/api/daq
Parameters	{ "query": { "gid": "gauge id", "from": "timestamp", "to": "timestamp", "event": "B", "sids": ["tag id", ""] } }
Responses	Code: 200 [{ "id": "tag id" "ts": "timestamp", "value": "12.5", }]
Note	Used to ask current or history tags value (DAQ): <ul style="list-style-type: none"> • 'gid' reference to the GUI component. • 'timestamp' milliseconds • 'event' can be 'B' or 'F' to step the range query in Back or Forward. • 'sid' list of tags id

Get list of logs files	
GET	/api/logsdir
Parameters	None
Responses	Code: 200 ["fuxa.log", "fuxa1.log", "fuxa-err.log", ...]

Get log file content	
GET	/api/logs
Parameters	{

	<pre> “query”: { “file”: “fuxa1.log” } </pre>
Responses	Download file content

Send email	
POST	/api/sendmail
Parameters	<pre> { “msg”: { “from”: “pippo@frangoteam.org”, “to”: “ciccio@frangoteam.org”, “subject”: “test”, “text”: “content”, “html”: “html-content”, }, “smtp”: { “host”: “blabla.smtp.org”, “port”: 587, “mailsender”: “pippo@frangoteram.org”, “username”: “username”, “password”: “password”, } } </pre>
Responses	Code: 200
Note	Used to send mail from frontend, ‘smtp’ is optional and used for configuring and testing.

Run script	
POST	/api/runscript
Parameters	<pre> { “script”: { “id”: “script id”, “name”: “script name”, “code”: “script code”, “parameters”: [{ “name”: “param name”, “type”: “value”, “value”: “12.5” }, {... }] } } </pre>
Responses	Code: 200
Note	Used to run scripts from frontend, ‘code’ and ‘name’ are optional and used for configuring and testing.