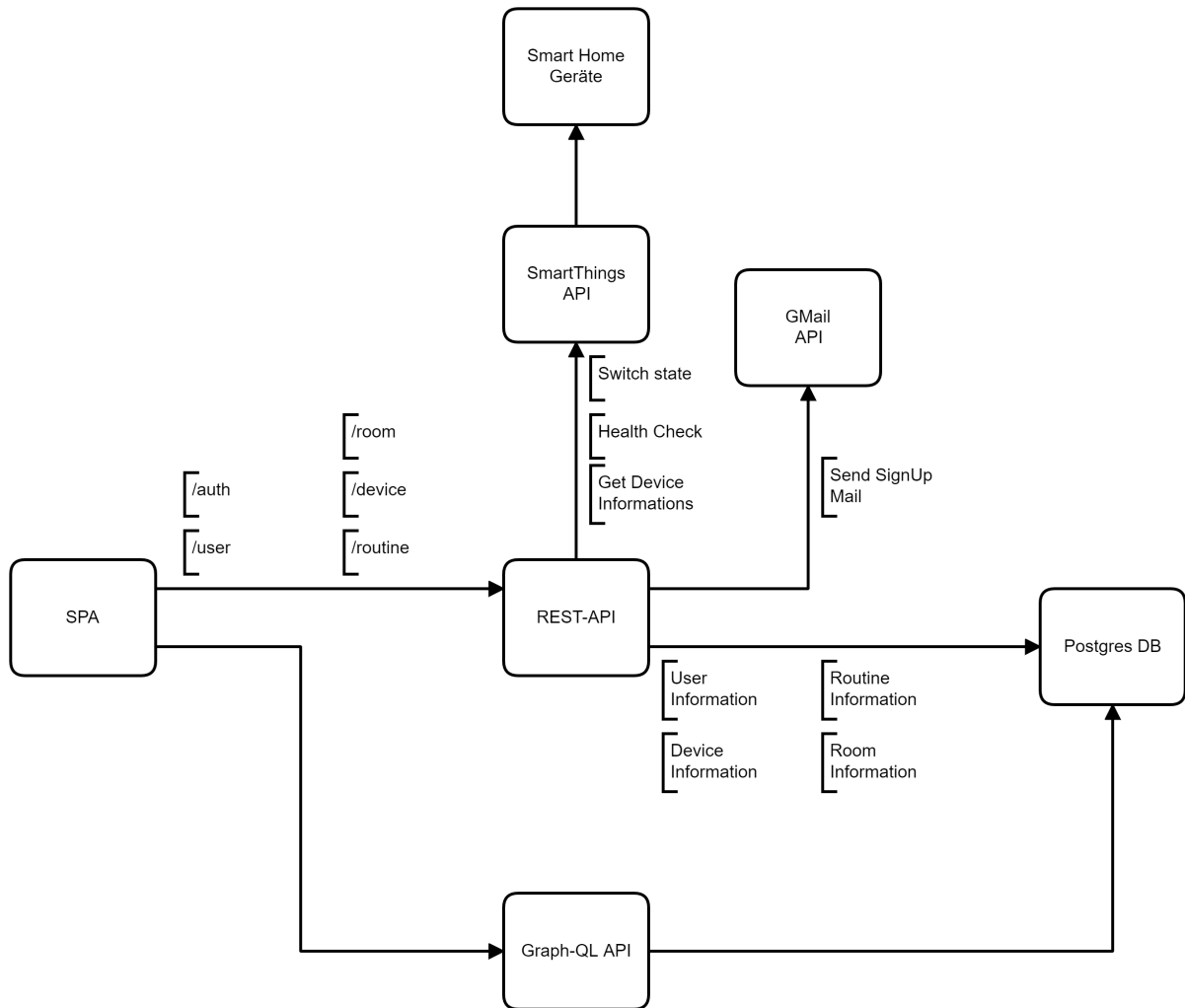


Smart Home Dashboard

Software Layout



Device Endpoint: Allows creating, deleting, modifying, and retrieving devices, as well as controlling them

Methode	Endpoint	Description	Data in	Data out
POST	/device	Gerät erstellen	Header: {"Authorization":"1234"} { "device":{ "device_id":"35df5c2a-ec4a-4283-9c9a-af7ec3d62464", "name": "Wohnzimmer Steckdose", "type":"outlet", "location": "Wohnzimmer"	200 { "message":"Device created" } 400 { "reason":"Wrong token" }

Methode	Endpoint	Description	Data in	Data out
			} }	
DELETE	/device/{device_id}	Gerät löschen	Header: {"Authorization":"1234"}	200 { "message":"Delition complete" } 400 { "reason":"Wrong token" }
PUT	/device/{device_id}	Gerät ändern	Header: {"Authorization":"1234"} { "change":{ "new-value":"Flur" "field":"location" } }	200 { "message":"Change complete" } 400 { "reason":"Passwort to short" }
GET	/device	Alle Geräte abrufen	Header: {"Authorization":"1234"}	200 { "devices":[{ "device_id":"35df5c2a-ec4a-4283-9c9a-af7ec3d62464", "name":"Wohnzimmer Steckdose", "type":"outlet", "loaction":"Wohnzimmer", "status":"online" }, { "device_id":"35df5c2a-ec4a-4283-9c9a-af7ec3d62464", "name":"Wohnzimmer Steckdose", "type":"outlet", "loaction":"Wohnzimmer", "status":"online" }] } 400
GET	/device/{device_id}	Einzelnes Gerät abrufen	Header: {"Authorization":"1234"}	200 { "device_id":"device01", "name": "Wohnzimmer Steckdose", "type":"outlet", "loaction": "Wohnzimmer", "status": "online" } 400 {

Methode	Endpunkt	Description	Data in	Data out
				"reason": "Wrong token" }
POST	/device/{device_id}/switch/{on/off}	Gerät schalten	Header: {"Authorization": "1234"}	200 { "message": "Accepted" } 400 { "reason": "Not Available for this device" } { "reason": "Connection Error" }
GET	/device/{device_id}/health-check	Gerät Status abfragen	Header: {"Authorization": "1234"}	200 { "message": "Offline / Online" } 400 { "reason": "Connection Error" }
GET	/device/smarthings	Gibt alle SmartThings Geräte zurück, welche noch nicht assigned wurden	Header: {"Authorization": "1234"}	200 { "devices": [{ "device_id": "35df5c2a- ec4a-4283-9c9a- af7ec3d62464", "name": "Wohnzimmer Steckdose" }, { "device_id": "35df5c2a- ec4a-4283-9c9a- af7ec3d62464", "name": "Wohnzimmer Steckdose" }] } 400 { "reason": "Connection Error" }

Auth Endpoint: Provides functions for login, logout, and checking server status

Methode	Endpunkt	Description	Data in	Data out
POST	/auth	anmelden	{ "email": "max@mustermann.com", "password": "pw" }	200 { "token": "123456" }

Methode	Endpunkt	Description	Data in	Data out
				400 { "reason": "Wrong credentials" }
DELETE	/auth	abmelden	Header: {"Authorization": "1234"}	200 { "message": "Sign out complete" } 400 { "reason": "Wrong token" }
GET	/auth	alive		200 { "message": "Alive" }

User Endpoint: Enables user account management, including creation, deletion, and modification

Methode	Endpunkt	Description	Data in	Data out
GET	/user	Eigenen Account abrufen	Header: {"Authorization": "1234"}	200 { "firstName": "Max", "lastName": "Mustermann", "email": "max@mustermann.com", "password": "pw" } 400 { "reason": "Wrong token" }
POST	/user	Account anlegen	{ "firstName": "Max", "lastName": "Mustermann", "email": "max@mustermann.com", "password": "pw" }	200 { "message": "Successfully created" } 400 { "reason": "Email already exists" }
DELETE	/user	Account löschen	Header: {"Authorization": "1234"}	200 { "message": "Successfully deleted" } 400 { "reason": "Wrong token" }
PUT	/user	Account ändern	Header: {"Authorization": "1234"} { "change": { "new-value": "Schulz" }	200 { "message": "Field changed" }

Methode	Endpoint	Description	Data in	Data out
			<pre>"field": "lastname" } }</pre>	<pre>400 { "reason": "Wrong token" }</pre>

Room Endpoint: Allows management of rooms, including creation, deletion, and modification

Methode	Endpoint	Description	Data in	Data out
GET	/room	Alle Räume abrufen	Header: {"Authorization": "1234"}	<pre>200 { "rooms": [{ "room_id": "room01", "name": "Wohnzimmer" }, { "room_id": "room01", "name": "Wohnzimmer" }] } 400 { "reason": "Wrong token" }</pre>
GET	/room/{room_id}	Raum abrufen	Header: {"Authorization": "1234"}	<pre>200 { "room_id": "room01", "name": "Wohnzimmer" } 400 { "reason": "Wrong token" }</pre>
POST	/room	Raum anlegen	{ "name": "Wohnzimmer", }	<pre>200 { "message": "Successfully created" } 400 { "reason": "Name already exists" }</pre>
DELETE	/room/{room_id}	Raum löschen	Header: {"Authorization": "1234"}	<pre>200 { "message": "Successfully deleted" } 400 { "reason": "Wrong token" }</pre>
PUT	/room/{room_id}	Raum ändern	Header: {"Authorization": "1234"} {"change": {"new-value": "Schlafzimmer" ,	<pre>200 { "message": "Field changed" } 400 {</pre>

			<pre> "field": "name" } </pre>	<pre> "reason": "Wrong token" } </pre>
--	--	--	--------------------------------	--

Routine Endpoint: Planned functionality for creating and managing automated sequences (not yet implemented)

Methode	Endpoint	Description	Data in	Data out
POST	/routine	Routine erstellen	<pre> Header: {"Authorization": "1234"} { "routine": { "name": "Guten Morgen", "actions": [{ "device_id": "light01", "action": "on" }, { "device_id": "outlet1", "action": "off" }], "trigger": { "type": "time", "value": "07:00" } } } </pre>	<pre> 200 { "message": "Routine created" } 400 { "reason": "Invalid device_id" } </pre>
GET	/routine	Alle Routinen abrufen	<pre> Header: {"Authorization": "1234"} </pre>	<pre> 200 [{ "id": "routine01", "name": "Guten Morgen", "actions": [...], "trigger": {...} }, ...] 400 { "reason": "Invalid token" } </pre>
GET	/routine/{routine_id}	Einzelne Routine abrufen	<pre> Header: {"Authorization": "1234"} </pre>	<pre> 200 { "id": "routine01", "name": "Guten Morgen", "actions": [...], "trigger": {...} } 400 { "reason": "Routine not found" } </pre>
PUT	/routine/{routine_id}	Routine ändern	<pre> Header: {"Authorization": "1234"} { "change": { "new-value": "Guten Abend" } "field": "name" } </pre>	<pre> 200 { "message": "Routine updated" } 400 { "reason": "Invalid action" } </pre>

DELETE	/routine/{routine_id}	Routine löschen	Header: {"Authorization":"1234"}	200 { "message":"Routine deleted" } 400 { "reason":"Unauthorized" }
--------	-----------------------	-----------------	-------------------------------------	---