M7011E API Documentation

Oskar Sundin osksun-6@student.ltu.se
Markus Jonsson amuojo-6@student.ltu.se

December 2020

Contents

1 Introduction

When an API call is issued to the simulator server, the requester is verified by checking access token in a header called Authorization in the incoming http request. This access token is generated by the authentication server and is linked to a specific user. This means in order to make a API call yourself for a specific user you need to extract the access token related to that specific user from an ongoing session's local storage or obtain it sending a login/regis request.

1.1 Battery

1.1.1 Queries

Query name	Arguments	Description
battery	none	Returns a Battery related to the requesting user ID.

1.1.2 Object type structure

Variable name	Variable Type	Description
id	GraphQLID	ID of the battery instance
buffer	GraphQLFloat	Amount of electricity in the battery, in Ws.
capacity	GraphQLFloat	Electricity capacity in the battery in Ws.

1.2 Generator

1.2.1 Queries

Query name	Arguments	Description
generators	none	Returns a generator list belonging to a building correlated with the ID of an user.

1.2.2 Object type structure

Variable name	Variable Type	Description
id	GraphQLID	ID of the generator instance.
baseOutput	GraphQLFloat	Electricity output of the generator when working at 100% effiency in Ws.
isBroken	GraphQLBoolean	Boolean indicating whether the generator is broken or not.

1.3 GeoData

1.3.1 Queries

Query name	Arguments	Description
geodata	none	Returns a GeoData related to the requesting user ID.

1.3.2 Object type structure

Variable name	Variable Type	Description
id	GraphQLID	ID of the geo data instance.
longitude	GraphQLFloat	Geographic coordinate in degrees specifying the east-west position of the object in the world space.
latitude	GraphQLFloat	Geographic coordinate in degrees specifying the north-south position of the object in the world space.
altitude	GraphQLFloat	Geographic coordinate in degrees specifying the depth in relation to the world of the object in the world space.
windSpeed	GraphQLFloat	Wind speed in m/s at the geographic position of the geo data in the world space.
temperature	GraphQLFloat	Temperature in celsius at the geographic position of the geo data in the world space.

1.4 House

1.4.1 Queries

Query name	Arguments	Description
house	none	Returns a House related to the requesting user ID.

1.4.2 Mutations

Mutation name	Arguments	Description
setOverproductionRatio	overproductionRatio: GraphQLFloat	Sets the ratio for how much electricity should go to the house battery and how much should be sold to the power plant during overproduction. Setting the ratio to 1 will divert all electricity to the house battery if possible.
setUnderproductionRatio	underproductionRatio: GraphQLFloat	Sets the ratio for how much electricity should be taken house battery and how much electricity should be bought from the power plant during underproduction. Setting the ratio to 1 means that 100% of the electricity should be taken from the battery if possible.

1.4.3 Object type structure

Variable name	Variable Type	Description
id	GraphQLFloat	ID of the house instance
electricityConsumption	GraphQLFloat	Electricity consumption of a house in Ws.
electricityProduction	GraphQLFloat	Electricity production of a house in Ws.
overproductionRatio	GraphQLFloat	Ratio for determining how much of the produced electricity should go to the house battery and how much should go to the power plant during overproduction. Set ratio to 1 to send all produced electricity to battery if possible.
underproductionRatio	GraphQLFloat	Ratio for determining how much electricity should be bought from the market and how much should be taken from the battery during underproduction. Set ratio to 1 to take 100% from battery if possible.
hasBlackout	GraphQLBoolean	Boolean indicating whether the house is experiencing a blackout or not.
battery	Battery	The battery of the house.
geoData	GeoData	The geo data correlated with the house.
generators	GraphQLList <Generator>	List of generators in the house.
powerPlant	PowerPlant	The power plant which the house exchange electricity with.

1.5 Manager

1.5.1 Queries

Query name	Arguments	Description
manager	none	Returns a Manager related to the requesting user ID.

1.5.2 Object type structure

Variable name	Variable Type	Description
id	GraphQLID	ID of the manager.
currency	GraphQLFloat	Amount of currency the manager has.
powerPlant	PowerPlant	The power plant owned by the power plant.
prosumers	GraphQLList <Prosumer>	All prosumers connected to the manager.

1.6 Picture

1.6.1 Mutations

Mutation name	Arguments	Description
uploadProfilePicture	picture: GraphQLUpload	Uploads a picture file.

1.7 Power Plant

1.7.1 Queries

Query name	Arguments	Description
powerPlant	none	Returns a PowerPlant related to the requesting user ID.

1.7.2 Mutations

Mutation name	Arguments	Description
updateElectricityPrices	electricityBuyPrice: GraphQLFloat, electricitySellPrice: GraphQLFloat	Update the price for prosumers buying electricity from the power plant and price when selling to power plant.
updateProductionOutputRatio	productionOutputRatio: GraphQLFloat	Update the ratio for the power plant electricity production.
startProduction	id: GraphQLID	Starts production for the power plant after a delay.
stopProduction	id: GraphQLID	Stops production for the power plant after a delay.

1.7.3 Object type structure

Variable name	Variable Type	Description
id	GraphQLID	ID of the power plant instance.
electricityConsumption	GraphQLFloat	Electricity consumption of the power plant in Ws.
electricityProduction	GraphQLFloat	Electricity production of the power plant in Ws
modelledElectricitySellPrice	GraphQLFloat	Recommended amount of currency units electricity can be sold for when selling to the power plant.
modelledElectricityTopPrice	GraphQLFloat	Recommended amount of currency electricity can be bought for when buying from the power plant.
electricitySellPrice	GraphQLFloat	Amount of currency units electricity can be sold for when selling to the power plant.
electricityBuyPrice	GraphQLFloat	Amount of currency units electricity can be bought for when buying from the power plant.
hasBlackout	GraphQLBoolean	Boolean indicating if the power plant is experiencing a blackout.
totalDemand	GraphQLFloat	Total electricity demand from buildings connected to the power plant.
productionOutputRatio	GraphQLFloat	Ratio for electricity production of the power plant.
productionFlag	GraphQLBoolean	Boolean indicating if the power plant is running or stopped.
actionDelayTimeS	GraphQLFloat	Delay for actions given to the power plant in seconds.
actionDescription	GraphQLString	Description for the currently executing action in the power plant.
battery	Battery	Battery of the power plant.
geoData	GeoData	Geo data of the power plant.
generators	GraphQLList <Generator>	List of generators in the power plant.

1.8 Prosumer

1.8.1 Queries

Query name	Arguments	Description
prosumer	id: GraphQLID	Returns a Prosumer related to the requesting user ID.

1.8.2 Mutations

Mutation name	Arguments	Description
deleteProsumer	id: GraphQLID	Deletes the prosumer and the house belonging to it.
setProsumerSellTimeout	id: GraphQLID, seconds: GraphQLFloat	Blocks a prosumer from selling for seconds specified by input.

1.8.3 Object type structure

Variable name	Variable Type	Description
id	GraphQLID	ID of a prosumer instance.
currency	GraphQLFloat	Amount of currency the prosumer has.
isBlocked	GraphQLBoolean	Indication if the prosumer is blocked from selling electricity to the power plant.
isOnline	GraphQLBoolean	Indication if the prosumer is online or not. This is decided by checking if the user has fetched specific data within the last 5 seconds.
house	House	House owned by the prosumer.

2 Authentication server

The API for the authentication server is split into 2 parts, the authentication API which deals with authenticating users and the user API that handles fetching data or modifying data in the authentication data base. Below is a table of the requests that are possible to make towards the authentication server. As with the simulator server the JWT-interceptor will add an authorization header that contains the requesting user's ID and role that will be used in the authentication server API.

2.1 Authentication requests

Request type	Endpoint	Description
POST	/register	Registers a user.
POST	/login	Validates the requesting user and if successful returns a refresh token assigned to the requesting user.
POST	/refresh-access-token	Validates the requesting user's refresh token and returns a new access token if successful.
DELETE	/logout	Validates requesting user and if successful deletes refresh token related to the requesting user.