API Policy

Informações API

Title	HBTrust.domain.policy
Description	API responsible for administering the policy creation and registration smart contract (Policy Factory) on the blockchain. It also performs consultation on policies and changes in the status of each policy.

Input Parameters

Below is the list of parameters expected for the API to perform its main function. This input information will be provided by the platform that will consume this API - so any error in sending the parameters will directly affect the functioning of the same.

Content type: `application/json`

Method	Summary	Description	Required?	Input parameters

POST/generatePolicy	Generate Policy	Creates and registers the policy on the blockchain.	Yes	Policy data { "privateKey": "string", "mutual": { "wallets": {} }, "coverages": { "prizeAmount": "string", "fipePercentage": "string", "app": "string", "glasses": "string", "rcfMaterials": "string", "rcfBodily": "string", "reserveCar": "string", "franchise": "string", "productCoverage": {} }, "policyInformation": { "proposal": "string", "apolice": "string", "startValidity": 0, "endValidity": 0, "apoliceStatus": "string" }, "policyHoldedrData": { "nameComplete": "string", "dateOfBirth": 0, "maritalStatus": "string", "pocket": "string", "cnpjCpf": "string", "gender": "string", "relationshipPolicyHolder": "string" }, "vehicleData": { "type": "string", "maker": "string", "model": "string", "numerSlides": "string", "yearManufacture": "string", "yearModel": "string", "licensePlate": "string", "chassis": "string", "renavam": "string", "fuel": "string", "newVehicle": "string", "vehicleFinaced": "string", "color": "string" }, "driverData": { "nameComplete": "string", "dateOfBirth": 0, "maritalStatus": "string", "pocket": "string", "cnpjCpf": "string", "gender": "string", "profession": "string", "cnh": "string", "dateFirstCnh": 0, "garage": "string", "usesWork": "string", "vehicleUse": "string" } }
PUT/Policy	Atualiza status da Apólice	Altera o valor do status da apólice na blockchain	Yes	"privateKey":"string" "addressPolicy":"string "status": "string"

POST/mutualPolicies	Insurance Policy List	Lists all insurance policies registered on the Blockchain.	Yes	"privateKey":"string" "addressMutual":"string
PUT/pause	Pause the smart contract	Sends a pause action (or unpause, depending on the current contract status)	Yes	"privateKey":"string
GET/pause	Check status	Consult if Contract is Paused		
POST/policy	Returns policy data	Returns all information from the Policy registered on the Blockchain.	Yes	"privateKey":"string" "addressPolicy":"string
POST/admin	Add admin	Adds an administrator to the policy creation and registration smart contract (Policy Factory)	Yes	"privateKey":"string" "address":"string
DELETE/admin	Remove admin	Removes an administrator in the policy creation and registration smart contract (Policy Factory)	Yes	"privateKey":"string" "address":"string

GET/admin	Check admin	Check if an address is an administrator in the policy creation and registration smart contract (Policy Factory)	Yes	`"addressWallet":"string`
POST/mutual	Add insurance company	Adds an insurance company to the policy creation and registration smart contract (Policy Factory)	Yes	`"privateKey":"string"` `"address":"string`
DELETE/mutual	Remove insurance company	Removes an insurance company in the policy creation and registration smart contract (Policy Factory)	Yes	`"privateKey":"string"` `"address":"string`
GET/mutual	Check insurance company	Check if an address is an insurance company in the policy creation and registration smart contract (Policy Factory)	Yes	`"addressWallet":"string`

List of smart contract functions called by the HBTrust.domain.policy API methods

Method	**Function**	**Description**

POST/generatePolicy	`generatePolicy`	Creates and registers the policy on the blockchain. `onlyMutual`
POST/mutualPolicies	`consultMutualPolicies`	Function that returns the list of Policies that a specific company has. `onlyMutual`
POST/lifecycle	`pause` / `unpause`	Function responsible for pausing actions that include or change the smart contract.
PUT/policy	`updateStatusPolicy`	Changes the value of the policy status on the blockchain. `onlyMutual`
POST/policy	`consultPolicyCoverage, consultDriver, consultPolicyHolder, consultPolicyInformation, consultPolicyMutual, consultVehicle, consultVehicleDocuments`	Returns all policy information registered on the blockchain.
POST/admin	`addAdmin`	Adds an administrator to the policy creation and registration smart contract (Policy Factory) `onlyAdmin`
DELETE/admin	`removeAdmin`	Removes an administrator in the policy creation and registration smart contract (Policy Factory) `onlyAdmin`
GET/admin	`isAdmin`	Consult if an address is mutual in the policy creation and registration smart contract (Policy Factory) `onlyAdmin`

POST/mutual	`addMutual`	Adds an insurance company to the policy creation and registration smart contract (Policy Factory)`onlyAdmin`
DELETE/mutual	`removeMutual`	Remove an insurance company to the policy creation and registration smart contract (Policy Factory) `onlyAdmin`
GET/mutual	`isMutual`	Consult if an address is an insurance company in the policy creation and registration smart contract (Policy Factory) `onlyAdmin`

Output parameters

Below is the list of parameters returned:

Method	Output parameter
POST/generatePolicy	transaction link blockexplorer
PUT/Policy	transaction link blockexplorer
POST/policy	
POST/mutualPolicies	Lista: data [addressPolicy, addressMutual] { "success": true, "code": "100-1000", "message": "Success!", "data": [["0xb4b0b379e6c77EB825951D40970020C2c6B36019"], ["0xc9D66B028A266FaFDFbEdD7A7ec54b0dFfEB2f61"]] }
POST/admin	`transaction link blockexplorer`
DELETE/admin	`transaction link blockexplorer`

GET/admin	true / false

Field Dictionary

Category	Variable	Description	Type	Example
privateKey		Private key of who signs the transaction(msg.sender)	`string`	2BFB60E72FB1C275CEC44446194A1620F225556686471491485936147 87C3F00
mutual				
	`wallets`	Numbers of the portfolios of each insurance company belonging to the policy.	`address[]`	[0x7105d236BdA4Db17666807726D99B70b1da19tau, 0x7105d236BdA4Db17666807726D99B70b1da19qwe, 0x7105d236BdA4Db17666807726D99B70b1da19def]
Coverage				
	`prizeAmount`	Coverage amount	`uint256`	3400
	`fipePercentage`	Fipe table index	`uint256`	2%
	`app`	Personal Passenger Accidents	`string`	App1
	`glasses`	Glass coverage	`uint256`	1
	`rcfMaterials`	Optional Liability for Material Damage	`string`	rcfM1
	`rcfBodily`	Optional Liability for bodily harm	`string`	rcfB1
	`reserveCar`	If the paid coverage has a reserve car	`uint256`	2
	`franchise`	Mandatory participation of the insured in a claim	`uint256`	1
	`productCoverage`	Insured product	`string[]`	[982001, 90201, 33221]

policyInformation	proposal	Identification of the Proposal that generated the policy	string	proposal1
	apolice	Policy identification (internal)	string	policy1
	startValidity	Effective date	uint256	1249052401
	endValidity	Effective end date	uint256	1249052401
	apoliceStatus	Policy status	uint256	2
PolicyHolderData				
	nameComplete	Insured's name	string	client
	dateOfBirth	Date of birth	uint256	1249052401
	maritalStatus	Marital status	uint256	1
	pocket		string	pocket1
	cnpjCpf	CPF / CNPJ	string	cpf11111111111
	gender	Gender	uint256	2
	relationshipPolicyHolder	Degree of kinship	uint256	1
VehicleDataModel				
	typeParam	Vehicle type	uint256	1
	maker	Maker	uint256	234
	model	Model	uint256	23451
	numerSlides		uint256	5
	yearManufactur	Year of manufacture	uint256	2019
	yearModel	Model year	uint256	2020
	licensePlate	License plate	string	ENV4211
	chassis	Chassis	string	3819920039813B23

	renavam	Renavan	string	32441HGS
	fuel	Fuel type	uint256	1
	newVehicle	Brand new car	uint256	1
	vehicleFinaced	Financed	uint256	1
	color	Color	uint256	1
DriverDataModel				
	nameComplete	Driver name	string	4CADIA Factory
	dateOfBirth	Date of birth	uint256	1984-01-12
	maritalStatus	marital status	uint256	1
	pocket		string	6140020
	cpfCnpj	CPF / CNPJ	string	99999999999
	gender	Gender	uint256	1
	profession	Profession	uint256	2341
	cnh	Driver license	string	3221333211
	dateFirstCnh	First license emission date	uint256	2013-02-01
	garage	Vehicle stays in garage	uint256	1
	usesWork	Vehicle used for work	uint256	1
	vehicleUse	Type of vehicle use	uint256	2

Request return

Method	Status	Code	Message
POST / PUT / DELETE		100-1010	`"errors": [ { "message": "apoliceData.apolice is required" }, { "message": "conductorData.date_of_birth is required" } ]`

POST / PUT / DELETE		100-1011	Gas required <gas> or always failing transaction.
POST / PUT / DELETE		100-1014	Invalid private key <privetKey>
POST / PUT / DELETE		100-3100	Transaction Failed, address does not have permission.
POST / PUT		100-1020	Policy address is invalid: <address>.
POST/policy		100-1031	Transaction subscriber has to be Mutual
POST/policy		100-1032	Don't have a policy with that address.
GET/mutualPolicy		100-1040	Mutuals wallet address is invalid.
GET/mutualPolicy		100-1012	Mutual has no policy.
POST/lifecycle		100-2110	Transaction Failed, smart contract already <pause/unpause>
GET/admin		100-7002	Wallet address is invalid.
POST/admin		100-1060	Admin's wallet address is invalid.
POST / PUT / GET / DELETE		100-7000	Internal Error.

Method	Status	Code	Message
POST/generatePolicy		100-7016	Transaction sucess to register policy!
POST/mutualPolicies		100-1000	Success! <response>
POST/lifecycle		100-1000	Success! <response>

PUT/policy		100-3031	Sucess transaction update status policy!
POST/policy		100-1000	Success! <response>
POST/admin		100-2010	Transaction sucessfull, admin added!
POST/mutual		100-2011	Transaction sucessfull, mutual added!
DELETE/admin		100-2020	Transaction sucessfull, admin removed!
DELETE/mutual		100-2021	Transaction sucessfull, mutual removed!
GET/admin		100-2030	Is Admin = true
GET/admin		100-2031	Is admin = false
GET/mutual		100-2032	Is Mutual = true
GET/mutual		100-2033	Is Mutual = false

Implementation

Instructions for use

- Clone the repository
- Align with latest version: Develop or Master
- Install the dependencies
- Configure .env file
- Run the API on localhost
- Access and test the API through the Browser using Swagger

StackEdit stores your files in your browser, which means all your files are automatically saved locally and are accessible **offline!**

Requirements

- NPM >=6.13.4

- GIT >=2.21.1
- NODE.JS >= 10.0.0

Clone repository

- git clone git@bitbucket.org:janusplatform/hbtrust.domain.claims.git

Align with latest version: Develop or Master

- git checkout develop
- git pull

Install dependencies

- cd hbtrust.domain.polucy
- npm install

Configure .env file

To be able to access smartcontracts on the Blockchain through the API it is necessary to create the .env file at the root of the directory. Clone the .env.example file with the following specifications:

```
MNENOMIC = // Your wallet's recovery words
INFURA_API_KEY = // Your Infura API Key after its registration
NETWORK_ID = // 1-Mainnet 3-Ropsten 4-Rinkeby 42-Kovan 1001-Development
TOKEN = // PDBToken Address
```

Run nodemoon on localhost

In the terminal access the root folder of the repository that was cloned, execute the command:

```
npm start
```

Run API

- After executing this command the nodemon will be initialized and the API will already be working at the address http://localhost:3000/swagger.