*Request to : /allAccounts*

Json sent to server :

* searchBy: {

firstName: '',

lastName: '',

Email: '',

userName: '',

country: '',

userType: 'string',

job: '',

hasOpenClaim: '',

accountStatus: '',

numberType: '',

numberValue: '',

},

* pageNumber : number

Desired Response :

* currentRecords: Array of Json objects in form of : {

id : user id,

profilePic: (string : encoded img),

firstName: ,

lastName: ,

Email: ,

userName: '',

}

* totalPageNum: ceil(number of all matching records / 20),

*Request to : /userDetails*

Data sent to server :

* id: user id

Desired Response : record : {

* firstName,
* lastName ,
* contractName,
* phoneNumber,
* mobileNumber,
* faxNumber,
* homepage,
* VatId,
* dateOfBirth ,
* psn ,
* discount;
* userCategory : {

value : id of the userCategory

label : name of the userCategory (string)

}

* userSubCategory: {

value : id of the userSubCategory

label : name of the userSubCategory (string)

} ,

* userActivitionStatus : {

value : id of the userActivitionStatus

label : name of the userActivitionStatus (string)

},

* currency,
* language: {

value : id of the language

label : name of the language (string)

} ,

* job : {

value : id of the job

label : name of the job (string)

},

* email,
* username,
* password nvarchar(50), (if we are allowed to show the password send it to client)
* emailConfirmed : (boolean),
* profilePic : (string : encoded img),
* bio ,
* createdAt datetime ,
* updatedAt datetime,

}

*Request to : /getOtherAccountsInfo*

Desired Response :

* infos: Json object in form of : **{**
  + - allUserCategoryies: array of jsons : [{

value : id of the userCategory

label : name of the userCategory (string)

}, … ]

* + - allUserSubCategories: array of jsons : [{

value : id of the userSubCategory

label : name of the userSubCategory (string)

}, … ]

* + - allUserActivitionStatus : array of jsons : [{

value : id of the userActivitionStatus

label : name of the userActivitionStatus (string)

}, … ]

* + - allLanguages: array of jsons : [{

value : id of the Language

label : name of the Language (string)

}, … ]

* + - allJobs : array of jsons : [{

value : id of the Job

label : name of the job (string)

}, … ]

**}**

*Request to : /allCountries*

*Desired Response : Array of Json objects in form of : {*

*value : id assigned to that country*

*label : name of the country (string)*

*}*

*====> example = [*

*{ value: 1, label: 'iran' },*

*{ value: 2, label: 'london' },*

*{ value: 3, label: 'paris' },*

*];*

*Request to : /allLanguages*

*Desired Response : Array of Json objects in form of : {*

*value : id assigned to that language*

*label : name of the language (string)*

*}*

*====> example = [*

*{ value: 1, label: 'persian' },*

*{ value: 2, label: 'english' },*

*{ value: 3, label: 'french' },*

*];*

*Request to : /allProductNames*

*Desired Response : Array of Json objects in form of : {*

*value : id assigned to that product*

*label : name of the product (string)*

*}*

*====> example = [*

*{ value: 1, label: ‘zan e rooz' },*

*{ value: 2, label: 'khanevade' },*

*{ value: 3, label: 'varzeshi' },*

*];*

*Request to : /allJobs*

*Desired Response : Array of Json objects in form of : {*

*value : id assigned to that job*

*label : name of the job (string)*

*}*

*====> example = [*

*{ value: 1, label: 'teacher' },*

*{ value: 2, label: 'student' },*

*{ value: 3, label: 'driver' },*

*];*

*Request to : /allClaims*

Json sent to server :

* searchBy :{

senderFirstName: '',

senderLastName: '',

senderEmail: '',

senderUserName: '',

recieverFirstName: '',

recieverLastName: '',

recieverEmail: '',

recieverUserName: '',

claimStatus: '',

},

* pageNumber : number

Desired Response :

* currentRecords : Array of Json objects in form of :

{

* + - id : claim id,
    - senderUserName,
    - receiverUserName,
    - msgStatus,
    - repliedMsgId,
    - status,

},

* totalPageNum: ceil(number of all matching records / 20),

*Request to : /claimDetails*

Data sent to server :

* id: claim id

Desired Response : record : {

* id ,
* messageHtml nvarchar(10000),
* senderUsername ,
* receiverUsername ,
* actionUsername,
* repliedMessageId,
* customerOrderId ,
* messageStatus,
* acceptedAdminUsername,
* isFinished : bool,
* imageAddress nvarchar(300),
* createdAt datetime,
* updatedAt datetime,

}

*Request to : /allComments*

Json sent to server :

* searchBy: {

ProductName : ‘string’

},

* pageNumber : number

Desired Response :

* currentRecords: Array of Json objects in form of : {

id : user id,

userName: username of the person who commented,

productName,

repliedCommentId,

status : comment status (should be pending :) )

text : comment text

createdAt : datetime

}

* totalPageNum: ceil(number of all matching records / 20),

*Request to : /getOtherProductsInfo*

Desired Response :

* infos : Json object in form of : **{**
  + - allPublishers: array of jsons : [{

value : id of the publisher

label : name of the publisher (string)

}, … ]

* allProductContentTypes: array of jsons : [{

value : id of the productContentType

label : name of the productContentType (string)

}, … ]

* allProductTypes: array of jsons : [{

value : id of the productType

label : name of the productType (string)

}, … ]

* allProductStatus: array of jsons : [{

value : id of the status

label : name of the status (string)

}, … ]

* allLanguages: array of jsons : [{

value : id of the language

label : name of the language (string)

}, … ]

* allAgeGroups: array of jsons : [{

value : id of the ageGroup

label : name of the ageGroup (string)

}, … ],

* allPeriods: array of jsons : [{

value : id of the period

label : name of the period (string)

}, … ],

**}**

*Request to : /allProducts*

Json sent to server :

* searchBy: {

publishers: array of jsons : [{

value : id of the publisher

label : name of the publisher (string)

}, … ]

productContentTypes: array of jsons : [{

value : id of the productContentType

label : name of the productContentType (string)

}, … ]

productTypes: array of jsons : [{

value : id of the productType

label : name of the productType (string)

}, … ]

status: array of jsons : [{

value : id of the status

label : name of the status (string)

}, … ]

languages: array of jsons : [{

value : id of the language

label : name of the language (string)

}, … ]

ageGroups: array of jsons : [{

value : id of the ageGroup

label : name of the ageGroup (string)

}, … ]

periods: array of jsons : [{

value : id of the period

label : name of the period (string)

}, … ]

}

* pageNumber : number

Desired Response :

* currentRecords: Array of Json objects in form of : {

id : product id,

title:,

description:,

coverImage: (string : encoded img),

price: ,

discount: ,

weight: ,

}

* *totalPageNum: ceil(number of all matching records / 20),*

*Request to : /productDetails*

*Data sent to server :*

*Id : product id*

*Desired Response : record : {*

*id ,*

*productType : {*

value : id of the *productType*

label : name of the *productType* (string)

*},*

*singleProductType: {*

value : id of the *singleProductType*

label : name of the *singleProductType* (string)

*},*

*ageGroup: {*

value : id of the *ageGroup*

label : name of the *ageGroup* (string)

*},*

*publishPeriod : {*

value : id of the *ageGroup*

label : name of the *ageGroup* (string)

*},*

*publisher {*

value : id of the *publisherId*

label : name of the *publisherId* (string)

*},*

*language : {*

value : id of the  *language*

label : name of the *language* (string)

*}*

*price decimal(11,4),*

*discount decimal(11,4),*

*issn nvarchar(20),*

*dewey nvarchar(20),*

*asb nvarchar(20),*

*EnglishTitle nvarchar(100),*

*EnglishDesc nvarchar(100),*

*weight decimal(11,4),*

*coverImage nvarchar(200),*

*priceUpdatedAt date,*

*tax decimal(11,4),*

*createdAt datetime,*

*updatedAt datetime,*

*}*