**Endpoints**

The endpoints have been separated into different files for readability and based on the function that they fulfill. The route for the endpoints is specified in the app.py file. The donor functionalities and NGO functionalities are present in the files: donor.py and ngo.py respectively. The admin functionalities are specified in admin.py. Other functionalities that are common to donor, NGO and admin are specified in common.py.

**Donor endpoints (Donor.py):**  
  
1) Create User Account:

Endpoint to create a donor account. This function is used to create a donor account while signing up on the application.

Sample data to be passed:

{"address": "Maddur Road,2nd Cross",

"phone": "1992123445",

"email": "jack@gmail.com",

"passwordHash": "password123",

"pincode": "576136",

"name": "Jack"}

Data returned:

{“success”,”User created successfully”} or {“Error”,”Couldn’t create User)

2) donateItem:

Function to donate an item. A user, after creating an account, can decide to donate an item that can be requested by any NGO.

Sample data to be passed:

{

"name": "Tshirts",

"category": "Clothes",

"subcategory": "Men",

"details": "We need tshirts for children aged 12 to 16",

"quantity": 30,

"quality": “good”,

"date": "2021-02-22T11:49:21.690721+00:00",

"details": "Usable T-shirts",

"pincode": "110016",

"File": (image of the item),

}

Expected output:

{

“status” : “success”,

“itemId” : “V9BhyXcB0Z-vOkRVlKfH”

}

OR

{“failure”, “couldn’t create item”}

3)

getRequirements :

Function to get the requirements of NGOs. This function lets the donor see all the available requirements posted by NGOs in order to respond to either of them by donating an item.

Sample data to be passed:

{

"donorId" : "tHFSyXcBqKxLelkZi3KG"

}

Sample output:

"itemRequirements": [

{

"requirementId": "XNCSyXcB0Z-vOkRVg6cQ",

"name": "Tshirts",

"category": "Clothes",

"subcategory": "Men",

"details": "We need tshirts for children aged 12 to 16",

"quantity": 30,

"ngoId": "xXGMyXcBqKxLelkZ63KA",

"ngo": "Smile Foundation",

"date": "2021-02-22T11:49:21.690721+00:00",

"pincode": "110016"

}

]

4)

respondToRequirement:   
This function lets the donor respond to an NGO requirement by allowing the user to donate an item . The user can choose to keep this donated Item “public” , i.e that item would be visible to other NGOs as well, or the user can choose to keep it visible only to the NGO that had put the specific requirement.

Sample data to be passed:

{

"requirementId": "XNCSyXcB0Z-vOkRVg6cQ",

"name": "Tshirts",

"category": "Clothes",

"subcategory": "Men",

"details": "We need tshirts for children aged 12 to 16",

"quantity": 30,

"quality": “good”,

"ngoId": "xXGMyXcBqKxLelkZ63KA",

"date": "2021-02-22T11:49:21.690721+00:00",

"details": "Usable T-shirts",

"pincode": "110016",

"File": (image of the item),

}

Sample output:

{

“status” : “success”,

“itemId” : “V9BhyXcB0Z-vOkRVlKfH”

}

OR

{“Failure”,”Error in respond to requirement”}

5) respondToDonationRequest:

This endpoint allows the donor to respond to the request made by an NGO. An NGO can make a request to the donor for the item that the donor has put up for donation. The donor can then accept or decline that request using this endpoint.

Sample data to be passed:

{

"ngoId":"v2m8Z3cBt\_E5e8Lmk8k8",

"ngoName" : "Giving Ngo",

"itemId" : "E-Ztd3cBZKhmmo7bUeur",

"donorId" : "vmm8Z3cBt\_E5e8LmS8kT",

"requirementId" : "5mkCeHcBt\_E5e8LmP8oE",

"actionTaken" : "accept"

}

Sample Output:

{“Success”,”Responded to donation request”}

OR  
{“Error”,”Couldn’t respond to donation request”}

6) deleteItem:

This endpoint allows a donor to delete an item that the donor had put up for donation earlier.

Sample data to be passed:

{

"itemId" : "njMuXHcBNvQI55IedGtt",

"donorId" : "lXMDXHcBd3Y4wJFJIB4P"

}

Sample output:

{“success”,”deleted item successfully”}

OR

{“Error”,”Couldn’t delete item”}

7)

getUpdatesForDonor:

This endpoint returns all the updates for the items that were donated by the donor. For instance, it allows the user to view if an NGO has requested for an item, or if an NGO has accepted it’s donation request for a certain requirement.

Sample data to be passed:

{"donorId":"xHGLyXcBqKxLelkZb3KK"}

Sample output:

"updatesForDonor": [

{

"Item1": {

"itemId": "xnGRyXcBqKxLelkZdHLh",

"itemName": "Lenovo ideapad",

"itemCategory": "Electronics",

"itemSubcategory": "Computers",

"itemQuantity": 5,

"itemQuality": "Excellent",

"itemDetails": "I have 5 excellent working laptops which I want to donate for a good cause.",

"itemDate": "2021-02-22T11:48:11.663271+00:00",

"itemImageLink": "W9CRyXcB0Z-vOkRVcqe4",

"itemUpdates": [

{

"updateType": "noupdate"

}

]

},

"requestLimit": 2

}]

**NGO Endpoints:**

1) Create NGO Account:

This endpoint is used to create an NGO account. An NGO can sign up on the app using this endpoint. Once an NGO has signed up, the admin needs to verify the documents uploaded by the NGO to verify its authenticity. Once the admin verifies the documents, the NGO is granted access to the application.

Sample data to be passed:

{"ngoName": "ApproveTest",

"pan": "ABC123",

"address": "Mandya Road,1st Cross",

"website": "www.google3.com",

"phone": "1232123455",

"email": "approve@ngo.com",

"description": "You can give stuff for poorest people",

"passwordHash": "password123",

"pincode": "576203"}

Sample Output: {“success”,”NGO Account created successfully”} or {“Error”,”Couldn’t create NGO Account”}

2) getNGoInfo :

This endpoint is used to get the ngo Information such as name, address, pan ,phone ,etc.

Sample data to be passed:

{

"ngoId" : "tXFZyXcBqKxLelkZonKY"

}

Sample output:

{

"address": "917, 9th B Main 3rd C Cross, Kalyan Nagar",

"description": "We work to educate unprivileged children",

"email": "developmentfocus@gmail.com",

"ngoName": "Development Focus",

"pan": "AAATD3096K",

"phone": "9560171988",

"pincode": "560043",

"verifiedNgoFlag": "true",

"website": "https://www.devfocus.in/"

}

3)requestItem :

An NGO can request any item that is up for donation using this endpoint. This endpoint creates a request that will be visible on the corresponding donor’s update feed for that particular item. The donor can later decide to either accept or decline that request.

Sample data to be passed:

{

"category" : "phones",

"subCategory" : "Electronics",

"itemName" : "Samsung M11",

"details" : "Need phones for students",

"quantity" : "1",

"ngoId" : "ceo@ngo.com",

"pincode" : "576123",

"publicFlag" : "false",

"itemId" : "njMuXHcBNvQI55IedGtt",

}

Sample output:

{“Status”:”Success”,”requirementId”:”XNCSyXcB0Z-vOkRVg6cQ",}

OR

{“Error”,”couldn’t perorm action”}

4) getItems:

This endpoint returns all the items that are available for donation. These items will be put up for donation by donors. NGOs can view this list of items and request any of those items from the donor.

Sample data to be passed:

{

"ngoId" : "v2m8Z3cBt\_E5e8Lmk8k8"

}

Sample output:

{"donationItems": [

{

"itemId": "V9BhyXcB0Z-vOkRVlKfH",

"name": "Old Mobile Phones",

"category": "Electronics",

"subcategory": "Phones",

"details": "I have 6 old mobile phones available for donation. (Iphone 5,6,7,8) (Samsung Galaxy s8,s9)",

"quantity": 5,

"quality": "Usable",

"imgLink": "VtBhyXcB0Z-vOkRVkqej",

"donorId": "tHFSyXcBqKxLelkZi3KG",

"date": "2021-02-22T10:55:53.634264+00:00",

"pincode": "110075",

"requestLimit": 2

}]}

5) createPublicRequirement :  
  
This endpoint lets the NGO create a requirement of an item that will be visible to all the donors when they log in. The donors can choose to respond to the requirement by donating.

Sample data to be passed:

{

"category" : "phones",

"subCategory" : "Electronics",

"itemName" : "Samsung M11",

"details" : "Need phones for students",

"quantity" : "1",

"ngoId" : "ceo@ngo.com",

"pincode" : "576123",

"publicFlag" : "false",

"itemId" : "njMuXHcBNvQI55IedGtt",

}

Sample output:

{“Status”:”Success”,”requirementId”:”XNCSyXcB0Z-vOkRVg6cQ",}

OR

{“Error”,”couldn’t create requirement”}

6) deleteRequirement:

This endpoint lets the NGO delete a requirement that was posted by them earlier.

Sample data to be passed:

{

"requirementId": "m3MsXXcBd3Y4wJFJTx4k",

"ngoId" :"mTMEXHcBNvQI55Ie2Wtm"

}

Sample output:

{“Success”: “Deleted requirement successfully”}

OR

{“Error”: “Couldn’t delete requirement” }

7) getUpdatesForNGO:

This endpoint returns all the updates pertaining to each requirement posted by the NGO.

Sample data to be passed:

{"ngoId":"tXFZyXcBqKxLelkZonKY"}

Sample output:

{"updatesForNGO": [

{

"Requirement1": {

"reqId": "t3FeyXcBqKxLelkZ6XKD",

"reqName": "Grade 10 CBSE Science Books",

"reqCategory": "Books",

"reqSubcategory": "Textbooks",

"reqDetails": "We plan to distribute these books among class 10 needy children who can't afford them.",

"reqQuantity": 50,

"reqDate": "2021-02-22T10:53:00.107830+00:00",

"reqUpdates": [

{

"updateType": "noupdate"

}

]

}

}]}

8) markItem :  
  
This endpoint allows an NGO to mark an item as received after the NGO has received that particular item from the donor.

Sample data to be passed:

{

"requirementId": "m3MsXXcBd3Y4wJFJTx4k",

"donorId":"lXMDXHcBd3Y4wJFJIB4P",

"ngoId" :"mTMEXHcBNvQI55Ie2Wtm",

"itemId" : "njMuXHcBNvQI55IedGtt"

}

Sample output:

{“success”,”item marked as received”}

OR

{“Error”,”Couldn’t mark item as received”}

9)acceptDeclineDonation:

This endpoint lets the NGO to either accept or decline a donation made by a donor.

Sample data to be passed:

{

"requirementId": "m3MsXXcBd3Y4wJFJTx4k",

"donorId":"lXMDXHcBd3Y4wJFJIB4P",

"ngoId" :"mTMEXHcBNvQI55Ie2Wtm",

"itemId" : "njMuXHcBNvQI55IedGtt",

"actionToken" : "decline"

}

Sample output:

{“Success” , “accepted donation”}

OR

{“Success”, “Rejected Donation”}

OR

{“Error”, “Couldn’t respond to donation”}

**Admin enpoints (admin.py) :**

1. ApproveRejectNGO:

This endpoint allows the admin to either approve or reject the NGO after verifying the authenticity of that NGO based on the forms submitted by the NGO.

Sample data to be passed:

{"actionToken":"accept","id":"qjMQXXcBNvQI55Ie0msM"}

Sample output:

{

"\_id": "xXGMyXcBqKxLelkZ63KA",

"\_index": "accounts",

"\_primary\_term": 1,

"\_seq\_no": 12,

"\_shards": {

"failed": 0,

"successful": 0,

"total": 0

},

"\_type": "\_doc",

"\_version": 5,

"result": "noop"

}

OR

{“Error” , “Couldn’t perform action”}

1. Authenticate:

This endpoint is used to authenticate the user (Donor or NGO) when they login. The password entered is verified and if correct, the user will be logged in to the application.

Sample data to be passed:

{"email":"approve@ngo.com","password":"password123"}

Sample output:

{

"role": "donor",

"id": "xHGLyXcBqKxLelkZb3KK",

"verified": **true**,

"pass": **true**,

"name": "Sriram",

"email": "sriram@gmail.com",

"address": "B-19, Sector 7, Dwarka, New Delhi",

"phone": 9560171988,

"pincode": 110075

}

OR

{“Error”, “Something failed while authenticating”}

**Common Endpoints:**

sendMessage:

This endpoint allows the donor and NGO to send messages and hence enables two-way communication. Messages can also contain images that could update the donor regarding the usage of the donated item.

Sample data to be passed:

{

"requirementId": "m3MsXXcBd3Y4wJFJTx4k",

"donorId":"lXMDXHcBd3Y4wJFJIB4P",

"ngoId" :"mTMEXHcBNvQI55Ie2Wtm",

"itemId" : "njMuXHcBNvQI55IedGtt",

"messageFrom" : "NGO",

"details" : "Received the item"

}

Sample output:

{“Status”: “Success”}

OR

{“Error” , “Couldn’t send message”}