

Oga Madam Webservice

Version One

Table of Content

Introduction	2
Token	2
User Registration	3
List Admin Users	4
List Category	6
Register Employee	7
Email & Sms	8
Email Verification	9
Employee Login	9
List Verified Employee	10
List All Employee	12
Transaction	13
List Transaction	14
Assigned Employee	15
List Salary	16
Convention	16
Status Codes	16

Introduction: Oga Madam web-service contain the application layer of the project which also handshake with third parties applications like payment gateways.

1. **Token:** User logs in to generate token that is being used in subsequent request. This token last more than 24hours and is being passed via header as authorization bearer token.

Method	URL
Post	{Base url}/token

Key Param	Value	Type
grant_type	password	string
username	admin4@yahoo.com	string
password	1234567aD#	string

Sample Response:

```
{
  "access_token": "ECODXqbAwjV5vMoa9BE-TbZUbl9dm0yqTdU-
ETQHPe5Bmu8iU6VqkJXiyXkJYN7bbU1HPQMeyI5j8hjfgqIXL_OlbwzNU0NAfMHxzpNUV8OUMpqrudV50u
7JfR__7Q9MADUTuaXtbbhLLyICF61yHLOjcCTxp79Z_jKyo-wte-
j4EIHOrQutzRzACE1Pxp6NWtODRUdP_VI40a8tircXwje28egWoEl64p_DANGvtTN-
x6OkqWJPqExlCF5yteumo1GvhNlrrtX5iPpxbrCL6KByqiBRf9vdyI7cle1yoFpUbzsyr_6sFV7EBUBaKMLJs3x2
Cmu_mY2LBAjsuh3VOTZpZl3rHAWkdXdOpVZDrMSQl4a2hYnIagUKDvlKaqwBMbPs-
THpM49sUP11vY34dZpj_3jvhwAmpShERYviz_y1D2sAZz2OWMF3DCmTX61Qs7TiHjrkzLBf_GHE-
BaH2RhTReaPeaVS3czxdsQ-eS3Z2-NBixkDbei8TUofno",
  "token_type": "bearer",
  "expires_in": 1209599,
  "userName": "admin4@yahoo.com",
  ".issued": "Fri, 28 Dec 2018 12:57:20 GMT",
  ".expires": "Fri, 11 Jan 2019 12:57:20 GMT"
}
```

Header status: 200 ok

Content-type: Application/Json

Parameter	Descripton
access_token	Is generated after user is authenticated
Token_type	Indicates the type of token returned

userName	Credential that made the request
issues	Displayed the date it was issued
Expires	Date it will expired after that the token becomes invalid

2. User Registration: User to register admin users on the platform.

Method	URL
Post	{Base Url}/ api/Account/Register

Request:

```
{
  "Password": "1234567aD#",
  "Email": "viia@yahoo.com",
  "FirstName": "Testers",
  "LastName": "Tests",
  "Address": "wuse",
  "DateOfBirth": "2018/10/05",
  "PlaceOfBirth": "Abuja",
  "MiddleName": "hitester",
  "PhoneNumber": "0803457746",
  "StateOfOrigin": "fct",
  "ConfirmPassword": "1234567aD#",
  "Sex": "Female"
}
```

Error Response:

```
{
  "Message": "The request is invalid.",
  "ModelState": {
    "": [
      "Email 'viia@yahoo.com' is already taken."
    ]
  }
}
```

```
}  
}
```

Status Code: 400 Bad Request

Content-Type: Application/Json

Right Response:

```
{  
  "ResponseCode": 201,  
  "ResponseStatus": true,if true  
  "Message": "User sucessfully created",  
  "UserId": "1e1de892-900b-489d-b320-1905705565b2"  
}
```

Status Code: 201 Created

Content-Type: Application/Json

Paramater	Description
ResponseCode	It displays the header response
ResponseStatus	Boolean param. If true it return right response
message	Gives the details of the respose
userId	Unque identification of the user that makes the request

3. List Admin Users: Returns list of registered admin users on the platform.

Method	URL
GET	{Base url}/ api/Service/ListUsers

Response:

```
{  
  "ResponseCode": 200,  
  "ResponseStatus": true,  
  "Message": "successful",  
  "Data": [  
    {
```

```

    "Id": "0af7dfc1-a50b-4fb9-bbb0-a93091b4aea3",
    "FirstName": "work2",
    "LastName": "work2",
    "Password": null,
    "Email": "admin2@yahoo.com",
    "PhoneNumber": "0803457746",
    "MiddleName": "hitester",
    "PlaceOfBirth": "Abuja",
    "DateOfBirth": "10/5/2018",
    "Address": "wuse",
    "Sex": "Female",
    "StateOfOrigin": "fct",
    "UserType": "Admin",
    "IsEmailVerified": true,
    "IsPhoneVerified": false,
    "IsUserVerified": True
  },
]
}

```

Parameter	Description	Parameter Type
Id	User Unique identification	string
UserType	Determine if user is Admin, Employer or Employee	string
isEmailVerified	Determined if user email is verified	boolean
isPhoneVerified	Determine if user phone number is verified on the platform	boolean
isUserVerified	Determine if user is being verified on the platform by Admin	boolean

4. List Category: Get list of category on the platform eg. Cook

Method	URL
GET	{Base url}/ api/Service/ListCategory

Response:

```
{
  "ResponseCode": 200,
  "ResponseStatus": true,
  "Message": "successful",
  "Data": [
    {
      "CategoryId": "catgory123456",
      "Title": "Cook",
      "Description": "People cooking for employer"
    }
  ]
}
```

Error Response:

Response:

```
{
  "ResponseCode": 404,
  "ResponseStatus": false,
  "Message": "No category found",
  "Data": null
}
```

5. Register Employee:

Method	Url
POST	{Base url}/ api/Account/RegisterEmployee

Request:

```
{  
  "Email": "admin5@yahoo.com",  
  "FirstName": "work4",  
  "LastName": "work4",  
  "Address": "wuse",  
  "DateOfBirth": "2018/10/05",  
  "PlaceOfBirth": "Abuja",  
  "MiddleName": "hitester",  
  "PhoneNumber": "0803457746",  
  "StateOfOrigin": "fct",  
  "Sex": "Female",  
  "ExtraData": {  
    "BVN": "0098776",  
    "NIMC": "988940",  
    "QualificationType": "Bsc"  
  }  
}
```

Error Response:

```
{  
  "ResponseCode": 403,  
  "ResponseStatus": false,  
  "Message": "Unable to create User"  
}
```

Response:


```
{
  "ResponseCode": 201,
  "ResponseStatus": true,
  "Message": "User sucessfully created",
  "UserId": "6b3a341e-4120-41cb-8173-b3f84cb727ec"
}
```

6. Email and Sms: API to send out email and sms on the platform.

Method	Url
POST	{Base Url}/ api/Service/SendEmailSms

Request:

```
{
  "From": "Oga Madam",
  "RecieptEmail": "Tester@yahoo.com",
  "SenderEmail": "Test@hotmail.com",
  "Subject": "developing Oga Madam",
  "Message": "working project",
  "Phone": "0803546397"
}
```

Response:

```
{
  "ResponseCode": 200,
  "ResponseStatus": true,
  "Message": "Seccessful",
  "Data": ok
}
```

N/B: Email and sms are sent if RecieptEmail and phone are passed but if one is passed then only one function is called eg if only ReceiptEmail is passed then only email will be sent out.

7. Email Verification: This is called when user tries to verify his email after registration

Method	Url
GET	{Base url}/api/Service/VerifyEmail?id={:UserId}&&hashParam={hash param}

Response:

```
{
  "ResponseCode": 200,
  "ResponseStatus": true,
  "Message": "Seccessful",
  "Data": ok
}
```

8. Employee Login: This is called when authenticating a user.

Method	Url
Post	{Base Url}/ api/Service/EmployeeLogin

Request:

```
{
  "Email": "admin5@yahoo.com",
  "Password": "1234567aD#"
}
```

Response:

```
{
  "ResponseCode": 200,
  "ResponseStatus": true,
  "Message": "successful",
  "Data": {
    "Email": "admin5@yahoo.com",
```

```

    "Password": "1234567aD#",
    "PhoneNumber": "0803457746",
    "FirstName": "work4",
    "LastName": "work4",
    "MiddleName": "hitester",
    "PlaceOfBirth": "Abuja",
    "DateOfBirth": "2018-10-05T00:00:00",
    "Address": "wuse",
    "Sex": "Female",
    "StateOfOrigin": "fct",
    "BVN": "0098776",
    "NIMC": "988940",
    "Upload": []
  }
}

```

Parameter	Description
Upload	Contents array of user upload id's

- 9. List Verified Employee:** This webservice return list of employers that has been verified on the platform by Admin.

Method	Url
GET	{Base Url}/ api/Service/ListVerifyEmployee

Response:

```

{
  "ResponseCode": 200,
  "ResponseStatus": true,
  "Message": "successful",
  "Data": [
    {
      "Id": "c7cbaa59-1bfb-484a-9e23-fd077196527e",

```

```
"FirstName": "work3",
"LastName": "work3",
"Email": "admin4@yahoo.com",
"PhoneNumber": "0803457746",
"MiddleName": "hitester",
"PlaceOfBirth": "Abuja",
"DateOfBirth": "10/5/2018 12:00:00 AM",
"Address": "wuse",
"Sex": "Female",
"StateOfOrigin": "fct",
"BVN": "123456",
"NIMC": "2342234",
"IsAttachedApproved": true,
"AttachedDate": null,
"BankName": null,
"AccountName": null,
"AccountNumber": null,
"SalaryAmount": 0,
"IsUserVerified": true,
"IsTrained": false,
"IsInterviewed": false,
"QualificationType": "Ssce"
}
]
}
```

10. List Employee: Return all employee on the platform both verified and unverified employee.

Method	Url
GET	{Base Url}/ api/Service/ListEmployee

Response:

```
{
  "ResponseCode": 200,
  "ResponseStatus": true,
  "Message": "successful",
  "Data": [
    {
      "Id": "c7cbaa59-1bfb-484a-9e23-fd077196527e",
      "FirstName": "work3",
      "LastName": "work3",
      "Email": "admin4@yahoo.com",
      "PhoneNumber": "0803457746",
      "MiddleName": "hitester",
      "PlaceOfBirth": "Abuja",
      "DateOfBirth": "10/5/2018 12:00:00 AM",
      "Address": "wuse",
      "Sex": "Female",
      "StateOfOrigin": "fct",
      "BVN": "123456",
      "NIMC": "2342234",
      "IsAttachedApproved": true,
      "AttachedDate": null,
      "BankName": null,
      "AccountName": null,
    }
  ]
}
```

```

    "AccountNumber": null,
    "SalaryAmount": 0,
    "IsUserVerified": true,
    "IsTrained": false,
    "IsInterviewed": false,
    "QualificationType": "Ssce"
  }
]
}

```

11. Post Transaction: API that is called after transaction on the platform has been carried out.

Method	Url
POST	{Base Url}/ api/Service/PayTransaction

Request:

```

{
  "TransactionId": "123456678",
  "TransactionDate": "2018/12/23",
  "Amount": "20000",
  "PaymentCategory": "Payment of employee",
  "PaymentStatus": "Successful",
  "PaymentChannel": "Web",
  "EmployeeId": "c7cbaa59-1bfb-484a-9e23-fd077196527e",
  "EmployerId": "cf1dd186-d4ad-4536-9c31-9dd0ca62d997",
  "StartDate": "2018/12/30",
  "EndDate": "2019/1/30"
}

```

Error Response:

```
{  
  "ResponseCode": 500,  
  "ResponseStatus": false,  
  "Message": "An error occurred while updating the entries. See the inner exception for details."  
}
```

Response:

```
{  
  "ResponseCode": 200,  
  "ResponseStatus": true,  
  "Message": "successful",  
  "Data": true  
}
```

12. List Transactions: It returns list of transactions made on the platform.

Method	Url
GET	{Base Url}/api/Service/ListTransaction

Response:

```
{  
  "ResponseCode": 200,  
  "ResponseStatus": true,  
  "Message": "successful",  
  "Data": [  
    {  
      "TransactionId": "1234566",  
      "TransactionDate": "12/23/2018 12:00:00 AM",  
      "Amount": "20000.00",  
      "PaymentCategory": "Payment of employee",  

```

```

        "PaymentStatus": "Failed",
        "PaymentChannel": "Web",
        "EmployeeId": "work3 work3",
        "EmployerId": "Test Test",
        "EndDate": "",
        "StartDate": ""
    }
}

```

13. Attached Employee: It returns list of employee that is attached to a specific employer.

Method	Url
GET	{Base Url}/api/Service/AttachedEmployee?EmployerId={EmployerId}

14. List of Salary: Return list of salary that has been made on the platform.

Method	Url
GET	{Base Url}/api/Service/ListSalary

Response:

```

{
  "ResponseCode": 200,
  "ResponseStatus": true,
  "Message": "successful",
  "Data": [
    {
      "SalaryId": "1234566",
      "Employee": "work3",

```



```

    "Employer": "Test",
    "Amount": "20000.00",
    "StartDate": "",
    "EndDate": ""
  }
]
}

```

Conventions:

- **Client** - Client application.
- **Status** - HTTP status code of response.
- All the possible responses are listed under 'Responses' for each method. Only one of them is issued per request server.
- All response are in JSON format.
- All request parameters are mandatory unless explicitly marked as **[optional]**

Status Codes

All status codes are standard HTTP status codes. The below ones are used in this API.

2XX - Success of some kind

4XX - Error occurred in client's part

5XX - Error occurred in server's part

Status Code	Description
200	OK
201	Created
202	Accepted (Request accepted, and queued for execution)
400	Bad request
401	Authentication failure
403	Forbidden
404	Resource not found
405	Method Not Allowed

409	Conflict
412	Precondition Failed
413	Request Entity Too Large
500	Internal Server Error
501	Not Implemented
503	Service Unavailable