**Author Cafe**

**RESTFUL API DOCUMENTATION**

**Base Url :** [http://kutung.com/demo/authors-cafe-php/api](http://kutung.com/demo/author-cafe-new/api/signup)

**1. SIGNUP**

**Url:** /signup

**Method :** POST

**Request Parameters :**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| userName | String | Valid User Name |
| email | String | Valid Email Address |
| password | String | Password |
| fileInfo | String | Key of the bio-data file |

**Responses**

1. **Email Already Exist**

{“error”: true,”message”: “Sorry, Username or Email already exist.”}

1. **Success Registration**

{“error”: false,”message”: “Registered Successfully”,”userName”:”karthick”,”userId”: 1,”paperId”:1}

**2. SIGNUP BIODATA UPLOAD**

**Url:** /upload/biodata

**Method :** POST

**Request Parameters :**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| biodata | File | Bio data File |

**Responses**

1. **If File not Selected**

{“error”: true,”message”: “Biodata file not exist in request”}

1. **If Upload Failed**

{“error”: true,”message”: “Upload Failed”}

1. **Upload Success**

{“error”: true,”message”: “Upload Success”,”fileInfo”:”biodata.pdf”}

**3. SIGNUP VALIDATIONS**

**Url: /validator/signup**

**Method :** POST

**Request Parameters :**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| field | String | Fields can be userName, email |
| value | String | Appropiate value fo the field to validate |

**Example :**

**field :** userName

**value :** karthick

**Responses**

1. **Username Already Exist**

{“error”: true,”message”: “Someone already has that username. Try another?”}

1. **Success Registration**

{“error”: false,”message”: “Username avaliable to use”}

**field :** email

**value :** karthick@kutung.com

**Responses**

1. **Email Already Exist**

{“error”: true,”message”: “Someone already has that Email. Try another?”}

1. **Success**

{“error”: false,”message”: “Email avaliable to use”}

**4.LOGIN**

**Url:** /login

**Method :** POST

**Request Parameters :**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| userName | String | User Name of the user |
| password | String | Password to login |

**Example :**

**userName:** karthick

**password : \*\*\*\*\*\*\*\***

**Responses**

1. **Incorrect Credentials**

{“error”: true,”message”: “Incorrect credentials”}

1. **Login Success**

{“error”: false,”userName”: “karthick”,”email”: “[karthick@kutung.com](mailto:karthick@kutung.com)”,”userId”:1,”paperId”:1}

{“error”: false,”userName”: “karthick”,”email”: “[karthick@kutung.com](mailto:karthick@kutung.com)”,”userId”:1,”paperId”:NULL}

.

**5. FETCH ASSETS**

**Url:** /assets4

**Method :** POST

**Request Parameters :**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| paperId | Integer | Id of the paper |

**Example :**

paperId : 1

**Responses:**

1. **If paperId is not given**

Status Code : 400

{“error”: true,”message”: “Bad Request”}

1. **If paperId is correct**

Status Code : 200

{“error”: false,”assets”: [{"fileKey":"Chrysanthemum\_1415876578.jpg","fileSize":"879394","lastModified":"Tue, 25 Nov 2014 06:25:54 GMT","fileUrl":"<https://prototypeupload.s3.amazonaws.com/prototype/assets/Chrysanthemum_1415876578_1416896753.jpg>"}]}

{“error”: false,”assets”: []}

**6. UPLOAD ASSET**

**Url:** /upload/assets4

**Method :** POST

**Content Type :** application/json

**Request Parameters :**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| userId | String | User Id of the user |
| paperId | Integer | Paper Id of the paper in which the asset need to be uploaded |
| assetFile | File | Asset File to Upload |

**Example 1 :**

userId: 1

paperId: 1

assetFile : desert.jpg

**Responses:**

1. **If Asset file is not in request**

{“error”: true,”message”: “File is Empty”}

1. **Upload Success**

Status Code : 201

{“error”: false,”message”: “Upload Success”,”fileInfo”:”http://....”, “fileType”:”jpg”}

**Example 2 :**

userId: 1

paperId: 1

assetFile : docxfilename.docx

**Responses:**

1. **If Asset file is not in request**

Status Code : 200

{“error”: true,”message”: “File is Empty”}

1. **If UserId is not valid**

Status Code : 401

{“error”: true,”message”: “Unauthorized Access. User Not Found”}

1. **Upload Success**

Status Code : 201

{“error”: false,”message”: “Upload Success”,”fileInfo”:”http://....”, “fileType”:”jpg” ,”assetId”,”1”,”fileType”: “docx”}

**7. CONVERT DOCX AND STORE HTML**

**Url:** /upload/htmlstore4

**Method :** POST

**Content Type :** application/json

**Request Parameters :**

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Value** | **Description** |
| assetId | Integer | Asset Id which is send at the time of docx upload in 6th Request |
| docHtml | String | Converted Html String to store |

**Example 1 :**

assetId : 1

keyPrefix : xx/xxx/xxx

docHtml : <html>…….</html>

**Responses:**

1. **If Asset id is Invalid**

Status Code : 200

{“error”: true,”message”: “Invalid Asset Id”}

1. **Upload Success**

Status Code : 200

{“error”: false,”message”: “Upload Success”,”fileInfo”:”http://....”, “fileType”:”jpg”,”fileType”:”docx”}

{

projectName : “Kutung - Authors Cafe”,

projectDesc : “   “,

baseUrl : “http:// ../api”,

data:[

url : “/“,

smallDesc: “”,

detailedDesc:””,

method: “post/get/“,

requestParam : {

userName : desc,

},

response: [

success: {

},

failure:{

}

}