**How to use ubuxa Api**

Ubuxa Api Url : Ubuxaapi.ubuxa.net

**How to login using the ubuxa api**

**Step 1**

First user would have to provide his or her credentials , useful parameters

1. **username** as key followed by value of the username
2. **password** as key followed by users password

note username and password should be used as the form or post name

**Url data method** : POST;

**Note:** if method is not post request would not be made to the server .

**Url to post to** : Ubuxaapi.ubuxa.net/site/authorize

**Sample data structure**

{

"status": 1,

"data": {

"authorization\_code": "a86539d6e5869d464de6b6e425046a37",

"expires\_at": 1554408689

}

}

Note : authorization code expires in a space of two minute

**Step 2**

Send authorization code to the access token url to obtain the access token which should be stored on the client side, and would be used for subsequent request to the server,

Note : access token expires in a space of two month.

Access code url : Ubuxaapi.ubuxa.net/site/accesstoken.

Parameters : authorization\_code

Url data method : Post.

**Sample data structure**

{

"status": 1,

"data": {

"access\_token": "7d78e721959f2bdf07e5b372c36165dd",

"expires\_at": 1559592935

}

}

store the access\_token on the client , and send every other request with the access\_token

eg using access token

url : ubuxaapi.ubuxa.net/folder/index?access\_token=85c39f938ebcff3bd0b118c5c34bb0c9

or pass **x-access-token**  to the header of the request

**How to signup using the ubuxa api**

The signup is of two phase namely the customer signup and the user signup, every account owner, or aperson who wants to create an account on ubuxa, would have to first create a customer profile, before the create a user profile, excluding users who where invited to join a workspace.

**Step 1**

***Customer signup***

***Useful parameters :***

1. master\_email (value type: string): this should be to the user as email and should carry a valid email which is unique on the system .
2. master\_doman (value type: string) :This should be presented to the user as domain and should hold a unique domain name which a user would constantly use to access his or her workspace .
3. account\_type (value type: string, input type: select/option): This is a required input type which helps determine if the user is registering as an individual or as a company, value should be either person , or corporation, users should see account type as the field name .
4. first\_name / surname (value type: string): depending on the value of the account type, if account type value is person then the next visible input should be first\_name and surname.
5. corporation\_name (value type: string): If account type is corporation, the input for corporation\_name would be use to capture the actual corporation or company name .

**Url data method** : POST;

**Url to post to** : ubuxaapi.ubuxa.net/site/customer-signup

**Test Url to post to** : ubuxaapi.ubuxa.com/site/customer-signup

Note: An email would be sent with a token which would be used to validate the email, and finish the registration process

**Sample data structure**

{

"status": 1,

"data": {

"cid": null,

"master\_email": "ubuxang@gmail.com",

"master\_doman": "kingwhlmkh",

"plan\_id": null,

"account\_number": null,

"billing\_date": "2019-04-16",

"entity\_id": 55,

"isNewRecord": true

}

}

**Step 2**

**Validate email**

***Useful parameters :***

validation\_code (value type: int, input type: number): A user is expected to enter pin sent to their email in an input to validate email and move to next stage.

**Url to post to** : ubuxaapi.ubuxa.net/site/validate-code

**Test Url to post to** : ubuxaapi.ubuxa.com/site/validate-code

Note: retuned value should be saved to be used in the next stage,

**(account\_type) : Note that this would determine if the signup page would have first\_name and surname if account type is corporation.**

**Sample data structure**

{

"status": 1,

"data": [

{

"id": "1",

"entity\_id": "1",

"cid": "7151",

"master\_email": "kkkkkk@gmail.com",

"master\_doman": "testing it ",

"plan\_id": null,

"billing\_date": "2019-01-28",

"account\_number": "6189",

"has\_admin": "1",

"account\_type": "person",

"role": 1,

"validation\_code": 9403

}

]

}

**Step 3**

**User signup**

***Useful Parameters :***

**email (type:get):** email gotten from the previous stage would be passed to the url as a get attribute, url sample would be shown below.

**cid (type:get):** cid gotten from the previous request would be used as a url param.

**role (type:get):** role gotten from the previous request would be used as a url param for role.

**validation\_code (type:get):** validation\_code gotten from previous request would be used here.

**username (type:post):** A user is expected to provide a username which would be used for login.

**password (type:post):** A user is expected to provide a password which would also be used for login.

**password\_repeat (type:post):** A user is expected to re enter the password to be used to validate the previously entered password .

**Url to post to** : ubuxaapi.ubuxa.net/site/signups?email=kkkkkk@gmail.com&cid=7151&role=1

**Test Url to post to** : ubuxaapi.ubuxa.com/site/ signups?email=kkkkkk@gmail.com&cid=7151&role=1&validation\_code=9403

**Sample data structure**

{

"status": 1,

"data": [

{

"first\_name": "johhhn",

"surname": "smiuuth",

"username": "mikeggg46",

"password": "mikelove",

"password\_repeat": "mikelove",

"basic\_role": "1",

"address": "kkkkytykk@gmail.com",

"state\_id": null,

"country\_id": null,

"cid": "7151",

"isNewRecord": true

}

]

}

Redirect user to login page .

**How to fetch all folders using ubuxa api**

***Useful Parameters : None.***

**Url (type:get)liveserver**: ubuxaapi.ubuxa.net/folder

**Url (type:get)testserver**: ubuxaapi.ubuxa.com/folder

**How to determine a private folder**

By default private\_folder is equal to 0, as such if a private folder is greater than 0 then private folder is equal to 1.

**How to determine a subfolder**

By default parent\_id is equal to 0 as such when parent\_id is greater than 0 it is regarded as a subfolder. Note: the parent id represent the parent folder id.

**How to determine a public / shared folder**

A public and shared folder is determined by the folder role, if the role is author it’s a public folder and if the role is user then it’s a shared folder.

**Sample data structure**

{

"status": 1,

"data": [

[

{

"16": {

"id": 16,

"parent\_id": 15,

"title": "another r p",

"description": null,

"folder\_image": null,

"private\_folder": 1,

"last\_updated": "2018-12-03",

"deleted": 0,

"cid": 33,

"folderstatus": "private",

"role": "author",

"createdby": 33

},

"17": {

"id": 17,

"parent\_id": 16,

"title": "again",

"description": "TRY AGAIN LOL",

"folder\_image": "uploads/21yIDuM8t8YOMhU9go8S0D2E5E9cFV6u.png",

"private\_folder": 1,

"last\_updated": "2019-04-01",

"deleted": 0,

"cid": 33,

"folderstatus": "private",

"role": "author",

"createdby": 33

},

"18": {

"id": 18,

"parent\_id": 17,

"title": "new folder",

"description": "test",

"folder\_image": null,

"private\_folder": 1,

"last\_updated": "2018-12-03",

"deleted": 0,

"cid": 33,

"folderstatus": "private",

"role": "author",

"createdby": 33

},

"28": {

"id": 28,

"parent\_id": 14,

"title": "test here ",

"description": null,

"folder\_image": null,

"private\_folder": 0,

"last\_updated": "2019-02-26",

"deleted": 0,

"cid": 33,

"folderstatus": "public",

"role": "author",

"createdby": 33

},

}

]

]

}

**How to fetch all subfolders of a folder using ubuxa api**

***Useful Parameters :***

Id (type: GET) : id represents the parent or root folder which would be used to fetch all the sub folders.

Note: same principles of a folder applies to subfolders

**Url (type:get)liveserver**: ubuxaapi.ubuxa.net/folder/subfolder?id=16

**Url (type:get)testserver**: ubuxaapi.ubuxa.com/folder/subfolder?id=16

**Sample data structure**

{

"status": 1,

"data": [

{

"subfolders": {

"17": {

"id": 17,

"parent\_id": 16,

"title": "again",

"description": "TRY AGAIN LOL",

"folder\_image": "uploads/21yIDuM8t8YOMhU9go8S0D2E5E9cFV6u.png",

"private\_folder": 1,

"last\_updated": "2019-04-01",

"deleted": 0,

"cid": 33,

"folderstatus": "private",

"role": "author",

"createdby": 33

},

"38": {

"id": 38,

"parent\_id": 16,

"title": "TRY AGAIN LOL",

"description": null,

"folder\_image": null,

"private\_folder": 1,

"last\_updated": "2019-04-02",

"deleted": 0,

"cid": 33,

"folderstatus": "private",

"role": "author",

"createdby": 33

},

"39": {

"id": 39,

"parent\_id": 16,

"title": "TRY AGAIN LOL",

"description": null,

"folder\_image": null,

"private\_folder": 1,

"last\_updated": "2019-04-03",

"deleted": 0,

"cid": 33,

"folderstatus": "private",

"role": "author",

"createdby": 33

},

"parent\_details": {

"total\_task": 1,

"users": 4,

"completed\_task": 1

}

}

}

]

}

**How to create a new folder/ subfolder using ubuxa api**

***Useful Parameters :***

Title(type:post)required: This is the folder title and its required at the point of creating any folder.

private\_folder (type:post) optional: this is used only when you want to make a folder private, as such private folder should be equal to 1.

parent\_id (type:post) optional: this is used when you want to create a subfolder, usually when a user is in the context of another folder and creates a folder. The value of parent id should be the id of the current folder in view.

**Url (type:get)liveserver**: ubuxaapi.ubuxa.net/folder/create

**Url (type:get)testserver**: ubuxaapi.ubuxa.com/folder/create