StoresMap provider application API

# API base url:

http://sm.wisyst.info/public/api/

Default method : POST

Notes: -

1. At pagination you can send (page) parameter in both methods (GET, POST)
2. The default application language is “en” unless you send it with “ar”, but there are some method that you should send it with “ar” or “en” only (will be notified)
3. access\_token is required to get the current provider with it

# API authentication: -

There are two mandatory parameters you will send each time you use this api

1. **api\_email:** api\_user@authentication.com
2. **api\_password:** M\*y@H0tSu9\_o

note I will not ask you in this documentation to send those parameters as I mentioned here that it is mandatory to send them

**if authentication failed**

return {'status' : false, 'errNum': 500, 'msg' => 'Authentication failed'}

# Prepare Sign Up API: -

**url** =>

[http://sm.wisyst.info/public/api/getProviderSignUpFirstStep]( http://sm.wisyst.info/public/api/getProviderSignUpFirstStep)

**params** => ( lang)

1- lang : (Default:en) -> represent application language

return {

'status' :true,

'errNum':errNum,

'msg' :"",

memberships: [

{

'membership\_id’: 1,

'membership\_name': " تاجر"

},

{

'membership \_id': 2,

''membership\_name': " فرد "

},

],

'countries': [

{

'country\_id': 1,

'country\_name': "المملكه السعودي",

'country\_code': 0966

},

{

'country\_id': 2,

'country\_name': "مصر",

'country\_code': 002

},

……..

],

}

# Get country cities API: -

**url** => http://sm.wisyst.info/public/api/countryCities

**params** => (country\_id, lang)

1- country\_id: (required)->represent selected country id

2- lang: (default:en)->represent app displayed language

**If fails**

return {

'status' :false,

'errNum':errNum,

'msg' :msg

}

errNum =>

1 => 'There is no data',

2 => 'country\_id is required'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Retrieved successfully' ,

'cities': [

{

"city\_id": 1,

"city\_name": "string name",

"country\_code": ex: 0966

},

{

"city\_id": 2,

"city\_name": "string name",

"country\_code": ex: 0966

},

…………..

]

}

# Provider signUp API: -

**url** => http://sm.wisyst.info/public/api/providerSignUpFirstStep “save sign data for first step”

**params** => full\_name , store\_name, phone ,country\_code, country\_id, city\_id ,password, password\_confirmation, membership\_id , lang, provider\_token)

1. - full\_name: (required) -> represent provider name(string)
2. store\_name: (required) -> represent store name (string)

3 - phone: (required |unique) -> represents provider phone (long || string)

5 - country\_code (required) -> represent provider phone country code

6 - country\_id (required|numeric) -> represents country id(integer)

7 - city\_id (required|numeric) -> represents city id(integer)

8- password: (required | MIN: 8) -> represents provider account password (string)

9- password\_confirmation: (required ) (string)

10- membership\_id (required|numeric) -> represents membership id(integer) 1- merchant 2- person

11 - lang (Default:en) -> provider application displayed language (string = "ar" || "en")

12 - provider\_token(required) -> represents provider account firebase reg id FCM

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'All fields are required',

2 => 'Password must not be less than 8 characters',

3=> ‘password not confirmed’

4 => 'Phone is already used',

5=> 'Failed to register, please try again later',

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'data': {

'id' :provider id,

'provider\_name':provider fullName,

'store\_name': store name,

'phone':provider phone,

country\_code:provider phone country code,

'country\_id': city id,

'city\_id': city id,

'membership\_id': city id,

'provider\_rate': 3.5,

Access\_token:’128 random string’,

'created\_at': account creation time in format (yyyy-mm-dd h:i:s)

},

'msg' :'Signed up successfully'

}

# Activate Account API: -

**url** =>  [http://sm.wisyst.info/public/api/](%20http://sm.wisyst.info/public/api/)[activateAccount](http://mjrb.wisyst.info/api/activateAccount)

Params :-

* access\_token :- (required) -> represent user access token random 128 string
* code :- :- (required) -> represent 4 digit code user entered as string

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تمت العملية بنجاح",

}

# Activate Account API: -

**url** =>  [http://sm.wisyst.info/public/api/](%20http://sm.wisyst.info/public/api/)resendActivationCode

Params :-

* access\_token :- (required) -> represent user access token

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

**If success**

return {

"status": true,

"errNum": 0,

"msg": " تم ارسال الكود بنجاح",

}

# Prepare Sign Up API: -

**url** =>

[http://sm.wisyst.info/public/api/getProviderSignUpSecondStep]( http://sm.wisyst.info/public/api/getProviderSignUpSecondStep)

**params** => ( lang)

1. lang : (Default:en) -> represent application language
2. access\_token (required) -> represent user token

return {

'status' :true,

'errNum':errNum,

'msg' :"",

"provider": {

"id": 96,

"provider\_name": "elamamy",

"category\_id": 2,

"store\_name": "elamamy",

"phone": "5555555333",

"country\_code": "+996",

"country\_id": 1,

"city\_id": 4,

"membership\_id": 1,

"access\_token": "F9OfTqjNSNQaIKLI3l5soKoJCtMrgABH0UYbOP9gtJi71bUn3Qof78QA9bzQzQNBlHECwk4TaATv8DyBWpxA2ksGknYXgwAf2wjWyOCOEhkrSttweYg6cOae6NCyJQX2",

"provider\_rate": 0,

"created\_at": "2019-04-21 20:41:25"

},

‘delivery\_methods’: [

{

‘method\_id’: 1,

‘delivery\_method\_name’: " مندوب",

‘ choosen‘: 0,

‘price’: 0

},

{

‘method\_id’: 1,

‘delivery\_method\_name’: " توصيل من نفس المتجر بسعر",

‘ choosen‘: 0,

‘price’: 10

},

…………………………..

],

'countries': [

{

'country\_id': 1,

'country\_name': "المملكه السعودي",

'country\_code': 0966

},

{

'country\_id': 2,

'country\_name': "مصر",

'country\_code': 002

},

……..

],

}

# Provider signUp API: -

**url** => http://sm.wisyst.info/public/api/providerSignUpSecondStep “save sign data for second step”

**params** => category\_id, commercial\_photo, delivery\_method\_id, longitude , latitude, delivery\_price, known\_phone ,access\_token)

1. - category\_id: (required|numeric) -> represent provider category id (integer)
2. commercial\_photo : (required) -> represent commercial image data

3 - delivery\_method\_id : (required |numeric) -> represents provider delivery method id (integer)

4- longitude: (required) -> represents provider location on map (double || string)

5 – delivery\_price : required only with delivery method id = 2 “ توصيل من المتجر ”

6- latitude: (required) -> represents provider location on map (double || string)

7 - lang (Default:en) -> provider application displayed language (string = "ar" || "en")

8- known\_phone (optional) -> represent phone number (string)

9 - access\_token(required) -> represents provider access\_token ’

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'All fields are required',

2 => 'Category id doesn\'t exists',

3 => 'image not valid',

4 => 'delivery method not exists',

5=> 'access\_token required',

6 => 'must be number ' ,

7 => 'delivery price required',

8 => 'Failed to register, please try again later',

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'data': {

"id": 96,

"provider\_name": "elamamy",

"store\_name": "elamamy",

"phone": "5555555333",

"country\_code": "+996",

"country\_id": 1,

"city\_id": 4,

"membership\_id": 1,

"delivery\_method\_id": 2,

"category\_id": 2,

"provider\_rate": 0,

"phoneactivated": "1",

"known\_phone": "93482738239",

"access\_token": "F9OfTqjNSNQaIKLI3l5soKoJCtMrgABH0UYbOP9gtJi71bUn3Qof78QA9bzQzQNBlHECwk4TaATv8DyBWpxA2ksGknYXgwAf2wjWyOCOEhkrSttweYg6cOae6NCyJQX2",

"longitude": "23232",

"latitude": "dssdsd",

"delivery\_price": 10,

"commercial\_photo": "",

"status": 0,

"publish": 1,

"created\_at": "2019-04-21 20:41:25"

},

'msg' :'Signed up successfully'

}

# Provider login API: -

**url** http://sm.wisyst.info/public/api/providerLogin

**params** => (phone, password, provider\_token, lang)

1- phone: (required) -> represent provider phone number

2- password: (required) -> represent provider account password

3- provider\_token: (required) -> represents provider device firebase id “FCM"

4- lang (Default:en) -> provider application displayed language (string = "ar" || "en")

**If fails**

**case errNUm = 5**

return {

'status' :false,

'errNum':errNum,

' msg' : msg,

'data': {

"id": 96,

"provider\_name": "elamamy",

"store\_name": "elamamy",

"phone": "5555555333",

"country\_code": "+996",

"country\_id": 1,

"city\_id": 4,

"membership\_id": 1,

"delivery\_method\_id": 2,

"category\_id": 2,

"provider\_rate": 0,

"phoneactivated": "1",

"known\_phone": "93482738239",

"access\_token": "F9OfTqjNSNQaIKLI3l5soKoJCtMrgABH0UYbOP9gtJi71bUn3Qof78QA9bzQzQNBlHECwk4TaATv8DyBWpxA2ksGknYXgwAf2wjWyOCOEhkrSttweYg6cOae6NCyJQX2",

"longitude": "23232",

"latitude": "dssdsd",

"delivery\_price": 10,

"commercial\_photo": "",

"status": 0,

"publish": 1,

"created\_at": "2019-04-21 20:41:25"

},

}

else

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'Phone is required',

2 => 'Wrong phone number',

3 => 'Password is required',

4 => 'Wrong data',

5 => 'You need to activate your account'

6 => 'provider\_token is required'

**Note:- if attribute (lang) is equal 'ar' these msgs will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'Logined successfully',

'data': {

"id": 96,

"provider\_name": "elamamy",

"store\_name": "elamamy",

"phone": "5555555333",

"country\_code": "+996",

"longitude": "23232",

"latitude": "dssdsd",

"commercial\_photo": "",

"membership\_id": 1,

"delivery\_method\_id": 2,

"category\_id": 2,

"phoneactivated": "1",

"known\_phone": "93482738239",

"access\_token": null,

"delivery\_price": 10,

"status": 0,

"publish": 1,

"country\_id": 1,

"city\_id": 4,

"provider\_rate": 0,

"created\_at": "2019-04-21 08:41:25"

},

}

# Forget Password Api: -

**url** => http://sm.wisyst.info/public/api/forgetPassword

Params =>

* phone( required |exists in providers)->represent provider phone

lang (Default:en)

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

If success

Return {

"status": true,

"errNum": 0,

"msg": "",

"code":

}

# Update Password Api: -

**url** =>  [http://sm.wisyst.info/public/api/](%20http://sm.wisyst.info/public/api/)updatePassword

Params =>

* password (required | minimum 6 chars),
* password\_confirmation(match password)
* access\_token(required) :- represent user access token
* lang(default:en)

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

**If success**

Return {

"status": true,

"errNum": 0,

"msg": "تمت العملية بنجاح"

}

# Prepare profile API: -

**url** => http://sm.wisyst.info/public/api/prepareEditProfile

**params** => (provider\_id, lang)

1- access\_token(required) :- represent the provider Access Token

2- lang (Default:en) -> provider application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'provider\_id is required',

2 => 'provider\_id must be a number',

3 => 'There is no data'

**Note:- if attribute (lang) is equal 'ar' these msgs will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'',

'data': {

"provider\_id": 96,

"provider\_name": "",

"phone": "",

"category\_id": 4,

"store\_name": "",

"longitude": "",

"latitude": "",

"country\_code": "+996",

"country\_id": 1,

"access\_token": "",

"city\_id": 4,

"delivery\_price": 10,

"delivery\_method\_id": 2,

"city\_name": "",

"profile\_pic": ""

},

"countries": [

{

"country\_id": 1,

"country\_name": "Saudi Arabia",

"choosen": 1,

"country\_code": 966

},

{

"country\_id": 2,

"country\_name": "Egypt",

"choosen": 0,

"country\_code": 966

}

],

"cities": [

{

**"city\_id": 1,**

**"city\_name": "Gada",**

**"choosen": 1**

**},**

**{**

**"city\_id": 2,**

**"city\_en\_name": "El Ryad",**

**"choosen":** 0

},

{

"city\_id": 3,

"city\_en\_name": "Cairo",

"choosen": 0

},

{

"city\_id": 4,

"city\_name": "Alexandria",

"choosen": 0

}

],

"cats": [

  {

"cat\_id": 2,

"cat\_ar\_name": "ملابس",

"choosen": 1

},

  {

"cat\_id": 3,

"cat\_ar\_name": "تلفونات",

"choosen": 0

},

………………..

],

"delivery\_methods": [

{

"method\_id": 1,

"method\_ar\_name": "مندوب",

"choosen": 0

},

{

"method\_id": 2,

"method\_ar\_name": "توصيل من نفس المتجر مع التكلفه",

"choosen": 1,

"price": 10

},

{

"method\_id": 5,

"method\_ar\_name": "الاستلام من المتجر",

"choosen": 0

}

],

}

**Update provider Profile API:-**

**url** => http://sm.wisyst.info/public/api/UpdateProfile

Params

full\_name(required) :- represent updated provider name

store\_name(required) :- represent updated provider store name

phone(required) :- represent the updated provider phone

country\_code(required)

longitude(required) :- represent provider location map longitude

latitude  (required) :- represent provider location map latitude

category\_id(required) :- represent the main category id

country\_id(required) :- represent the country id

city\_id(required) :- represent the city id

access\_token(required) :- represent the provider Access Token

delivery\_method\_id(optional) : represent the delivery method type id can be null

delivery\_price(required-with: delivery\_method\_id) required only with delivery\_method\_id = 2

profile\_pic (not required) :- represent the provider selected image in base64 Format as String[ The image Will be Send if the provider update His Image ]

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

If success

Return

{

"status": true,

"errNum": 0,

"msg": "",

"isPhoneChanged":,

"activation\_code":,

"provider": {

"id": 96,

"provider\_name": "ahmed emam",

"store\_name": "elamamy",

"phone": "5555555333",

"country\_code": "+996",

"longitude": "23232",

"latitude": "dssdsd",

"commercial\_photo": "",

"membership\_id": 1,

"delivery\_method\_id": 2,

"category\_id": 2,

"phoneactivated": "1",

"known\_phone": "93482738239",

"access\_token": "F9OfTqjNSNQaIKLI3l5soKoJCtMrgABH0UYbOP9gtJi71bUn3Qof78QA9bzQzQNBlHECwk4TaATv8DyBWpxA2ksGknYXgwAf2wjWyOCOEhkrSttweYg6cOae6NCyJQX2",

"delivery\_price": 10,

"status": 0,

"publish": 1,

"country\_id": 1,

"city\_id": 4,

"provider\_rate": 0,

"profile\_pic": "",

"created\_at": "2019-04-21 08:41:25"

},

"isPhoneChanged": false

}

# 

# Get provider Main category API: -

**url** : http://sm.wisyst.info/public/api/getProviderMainCat

**desc** : apis return the main category that the provider select it when first registration

**params** => (provider\_id,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

'cats' : {

"cat\_id": 2,

"cat\_name": "Clothes"

}

}

# Get provider Store categories API: -

**url** : http://sm.wisyst.info/public/api/getProviderStoreCategories

**desc** : apis return the main category that the provider select it when first registration

**params** => (access\_token,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

'cats' :  [

  {

"cat\_id": 1,

"store\_cat\_name": "ملابس رجالي"

}

],

}

# Add provider category API: -

**url** : http://sm.wisyst.info/public/api/addProviderCategory

**desc** : api to save new provider category

**params** => (access\_token,store\_cat\_ar\_name,store\_cat\_en\_name,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- store\_cat\_ar\_name : (required)-> represent Arabic name of category

4- store\_cat\_en\_name : (required)-> represent english name of category

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

'cats' : {

"cat\_id": 2,

"cat\_name": "Clothes"

}

# Edit provider category API: -

**url** : http://sm.wisyst.info/public/api/editProviderCategory

**desc** : api edit provider category by given “cat\_id”

**params** => (access\_token,cat\_id ,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- cat\_id : (required)-> represent id of category

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم تعديل التصنيف بنجاح "

}

# update provider category API: -

**url** : http://sm.wisyst.info/public/api/updateProviderCategory

**desc** : api update provider category by given “cat\_id”

**params** => (access\_token,cat\_id,store\_cat\_ar\_name,store\_cat\_en\_name,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- store\_cat\_ar\_name : (required)-> represent Arabic name of category

4- store\_cat\_en\_name : (required)-> represent english name of category

5- cat\_id : (required)-> represent id of category

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم تعديل التصنيف بنجاح "

}

# delete provider category API: -

**url** : http://sm.wisyst.info/public/api/deleteProviderCategory

**desc** : api delete provider category by given “cat\_id” will delete also all products inside it

**params** => (access\_token,cat\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- cat\_id : (required)-> represent id of category

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم حذف التصنيف بنجاح "

}

# Get provider Offers API: -

**url** : http://sm.wisyst.info/public/api/getProviderOffers

**desc** : apis return the offers belong to the provider

offers has 3 status

|  |  |
| --- | --- |
|  | 0 -> pending  1 -> approved  2 -> paid |

**params** => (access\_token,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

"offers": [

  {

"offer\_id": 1,

"offer\_title": "عرض الكبير جدا",

"offer\_photo": "",

"start\_date": "2019-04-14 15:39:04",

"end\_date": "2019-04-18 15:39:04",

"paid": "0",

"paid\_amount": 0,

"status": "0"

},

………………….

],

}

# Add provider Offer API: -

**url** : http://sm.wisyst.info/public/api/addProviderOffer

**desc** : api to save new provider offer

**params** => (access\_token,store\_cat\_ar\_name,store\_cat\_en\_name,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- offer\_title : (required)-> represent title of offer

4- start\_date : (required| date\_format:Y-m-d H:i:s) -> start date of offer

5- end\_date : (required| date\_format:Y-m-d H:i:s) -> end date of offer

6- photo (required|mimes) -> represent photo of the offer

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم اضافه العرض بنجاح ",

"offer\_id": 5

}

# pay Offer API: -

**url** : http://sm.wisyst.info/public/api/payProviderOffer

**desc** : api used to change the stauts of the specific offer to 2 (paid ) after provider make payment via application

**params** => (access\_token,offer\_id , paid\_amount ,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- offer\_id : (required)-> represent id of offer

4- paid\_amount: (required) -> represent amount paid for this offer

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم اضافه العرض بنجاح ",

"offer\_id": 5

}

# stop Offer API: -

**url** : http://sm.wisyst.info/public/api/stopProviderOffer

**desc** : provider stop the offer allow only when offer accepted from admin

**params** => (access\_token,offer\_id , paid\_amount ,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- offer\_id : (required)-> represent id of offer

**If success**

return {

"status": true,

"errNum": 0,

"msg": “تم ايقاف العرض ",

# Add Product API: -

**url** => [http://sm.wisyst.info/public/api/addProduct](http://sm.wisyst.info/public/api/addProduct%20)

**params** => (access\_token, title, category\_id, **description**, price, product\_images,options, options\_price ,sizes, sizes\_price ,colors, colors\_price, lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token: (required)-> represents provider access\_token

3- title : (required) -> represents product name

4- category\_id : (required|numeric) -> represents category id of the provider

5- **description**: (required|text) -> represents the description of the product

6- price : (required) -> represents product price

7- product\_images: (required|array) represent array of product image files must pass as 🡪

product\_images[]

8- options : (optional| array) => represent array of options of products pass as options[]

9- sizes : (optional| array) => represent array of sizes of products pass as sizes[]

10- colors : (optional| array) => represent array of colors of products pass as colors []

11 - options\_price(required-with:options) -> pass as array -> options\_price[]

12 - sizes\_price(required-with:sizes) -> pass as array -> sizes \_price[]

13 - colors\_price(required-with:colors) -> pass as array -> colors \_price[]

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'Product added successfully',

'product\_id' : 6

}

# Get Products API: -

**url** => http://sm.wisyst.info/public/api/getProducts

**params** => (access\_token,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

**If success**

return {

"status": true,

"errNum": 0,

"msg": "",

"products": {

"current\_page": 1,

"data": [

  {

"product\_id": ,

"title": " ",

"product\_image": "",

"description": "",

"provider\_id":

},

  {

"product\_id":,

"title": " ",

"product\_image": "",

"description": "",

"provider\_id": 96

},

……………….

],

"from": 1,

"last\_page": 1,

"next\_page\_url": null,

"path": "",

"per\_page": 10,

"prev\_page\_url": null,

"to": 8,

"total": 8

}

}

# Delete Product API: -

**url** : http://sm.wisyst.info/public/api/deleteProduct

**desc** : api delete provider product by given “product\_id”

**params** => (access\_token,product\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- product\_id : (required)-> represent the product id

If success

return {

"status": true,

"errNum": 0,

"msg": " product deleted successfully "

}

# Prepare update Product API: -

**url** => http://sm.wisyst.info/public/api/prepareProductUpdate

**params** => (access\_token,product\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- product\_id : (required)-> represent the product id

**If success**

Return

{

"status": true,

"errNum": 0,

"msg": "",

"product": {

"title": "sd,lsdldsadasd",

"description": "skdkskssdsdsds",

"price": 4000

},

"images": [

  {

"image\_id": 26,

"product\_image": ""

},

  {

"image\_id": 25,

"product\_image":”"

}

],

"cats": [

  {

"cat\_id": 4,

"store\_cat\_ar\_name": "ملابس رجالي",

"choosen": 0

},

  {

"cat\_id": 2,

"store\_cat\_ar\_name": "ملابس حريمي",

"choosen": 1

}

],

}

# Update Product API: -

**url** => http://sm.wisyst.info/public/api/updateProduct

**params** => (access\_token, title,product\_id, category\_id, **description**, price, product\_images,options, options\_price ,sizes, sizes\_price ,colors, colors\_price, deleted\_images,lang)

lang: (default:en) -> represent application displayed language

1- product\_id (required) -> represent product id

2- access\_token: (required)-> represents provider access\_token

3- title : (required) -> represents product name

4- category\_id : (required|numeric) -> represents category id of the provider

5- **description**: (required|text) -> represents the description of the product

6- price : (required) -> represents product price

7- product\_images: (required|array) represent array of product image files must pass as 🡪

product\_images[]

8- options : (optional| array) => represent array of options of products pass as options[]= ”sjdjsd”

9- sizes : (optional| array) => represent array of sizes of products pass as sizes[]= ”sjdjsd”

10- colors : (optional| array) => represent array of colors of products pass as colors []= ”sjdjsd”

11 - options\_price(required-with:options) -> pass as array -> options\_price[]

12 - sizes\_price(required-with:sizes) -> pass as array -> sizes \_price[]

13 - colors\_price(required-with:colors) -> pass as array -> colors \_price[]

14 - deleted\_images (optional|array ) -> represent array of images id [1,2,3,4,5,….]

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :Product updated successfully'

}

# Get Provider Job API: -

**url** : http://sm.wisyst.info/public/api/getProviderJobs

**desc** : apis return the All provider Jobs

**params** => (access\_token,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

'jobs' :  [

  {

" job\_id": 1,

"job\_title ": "",

"**job\_desc** ": "",

},

…………………….

],

}

# Add provider Job API: -

**url** : http://sm.wisyst.info/public/api/addProviderJob

**desc** : api to save new provider Job

**params** => (access\_token, job\_title , job\_desc,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- job\_title : (required|string)-> represent title of job

4- job\_desc: (required|text)-> represent job description

**If success**

return {

'status' :true,

'errNum' :0,

"msg": "تم اضافه الوظيفة بنجاح ",

"job\_id": 1

}

# Edit provider Job API: -

**url** : http://sm.wisyst.info/public/api/editProviderJob

**desc** : api edit/prepare provider Job by given “job\_id”

**params** => (access\_token,job\_id ,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- job\_id : (required)-> represent id of category

**If success**

return {

"status": true,

"errNum": 0,

"msg": "  "

"data": {

"id": 1,

"job\_title": "",

"job\_desc": ""

}

}

# update provider Job API: -

**url** : http://sm.wisyst.info/public/api/updateProviderJob

**desc** : api update provider job

**params** => (access\_token,job\_id,job\_title,job\_desc,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- job\_id: (required|integer)-> represent id of job

4- job\_title: (required|string)-> represent title of job

5- job\_desc: (required|text)-> represent job description

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم تعديل الوظيفة بنجاح "

}

# Delete provider Job API: -

**url** : http://sm.wisyst.info/public/api/deleteProviderJob

**desc** : api delete provider job by given “job\_id”

**params** => (access\_token,job\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- job\_id : (required|integer)-> represent job id

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم حذف الوظيفة بنجاح "

}

# Get Job Details API: -

**url** : http://sm.wisyst.info/public/api/getProviderJobDetails

**desc** : api return the Jobs details

**params** => (access\_token,job\_id,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- job\_id (required|integer) -> represent job id

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

"data": {

"job\_id": 2,

"job\_title": "",

"job\_desc": "",

"created\_date": "2019-04-27",

"applicants":  5

}

}

# Get Job applicants API: -

**url** : http://sm.wisyst.info/public/api/getJobApplicants

**desc** : api return the Jobs Applicants data

**params** => (access\_token,job\_id,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- job\_id (required|integer) -> represent job id

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

"applicants": {

"current\_page": 1,

"data": [

  {

"applicant\_id": 1,

"applicant\_name": "ahmed emam",

"applicant\_cv": "path to cv file ",

"created\_date": "2019-04-27"

},

…………………….

],

"from": 1,

"last\_page": 1,

"next\_page\_url": null,

"path": "",

"per\_page": 10,

"prev\_page\_url": null,

"to": 1,

"total": 1

}

}

# Get provider Excellent Requests API: -

**url** : http://sm.wisyst.info/public/api/getProviderExcellenceRequests

**desc** : apis return the ProviderExcellenceRequests

offers has 3 status

|  |  |
| --- | --- |
|  | 0 -> current  1 -> complete |

**params** => (access\_token,lang,type)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- type: (required|in:0,1) -> represents requestr types (0 = currently requests , 1 = completely request])

**If success**

**First case if order state is current**

return {

"status": true,

"errNum": 0,

"msg": "",

"requests": {

"current\_page": 1,

"data": [

  {

"request\_id": 1,

"paid": "0",

"paid\_amount": 0,

"status": "0",

"order\_date": "2019-04-27"

},

……….

],

"from": 1,

"last\_page": 1,

"next\_page\_url": null,

"path": "",

"per\_page": 10,

"prev\_page\_url": null,

"to": 1,

"total": 1

}

**}**

Second case if order state is complete

return {

"status": true,

"errNum": 0,

"msg": "",

"requests": {

"current\_page": 1,

"data": [

  {

"request\_id": 1,

"paid": "0",

"paid\_amount": 0,

"status": "0",

"expire\_date": "2019-04-30"

},

……….

],

"from": 1,

"last\_page": 1,

"next\_page\_url": null,

"path": "",

"per\_page": 10,

"prev\_page\_url": null,

"to": 1,

"total": 1

}

**}**

# Add provider Excellence Request API: -

**url** : http://sm.wisyst.info/public/api/addProviderExcellenceRequests

**desc** : api to save new provider Excellence Request

**params** => (lang,access\_token, name , country\_code, phone, start\_date, end\_date, main\_category\_id)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- name : (required)-> represent name of the provider request

4- country\_code (required) -> represent country code i.e +966

5-phone (required) -> represent phone number

6-start\_date : (required| date\_format:Y-m-d H:i:s) -> start date of request

7- end\_date : (required| date\_format:Y-m-d H:i:s) -> end date of request

8- main\_category\_id (required|integer) -> represent id of store category which the provider need not appear get with api “getProviderMainCat”

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم اضافه الطلب بنجاح ",

"request\_id ": 5

}

# Get Excellent Request Details API: -

**url** : http://sm.wisyst.info/public/api/ProviderExcellenceRequestDetails

**desc** : api return the request details

**params** => (access\_token,request\_id ,lang)

1- lang: (default:en) -> represent application displayed language

2- access\_token : (required)-> represent provider access\_token

3- request\_id (required|integer) -> represent excellent request id

**If success**

**Case 1: if the request type (status = 0 ) i.e current**

return "status": true,

"errNum": 0,

"msg": "",

"data": {

"request\_id": 3,

"status": "0",

"paid": "1",

"paid\_amount": 300, // if paid

"start\_date": "2019-04-27",

"end\_date": "2019-04-30",

"order\_date": "2019-04-27",

"daysCount": 3,

"totalCost": 30

}

**Case 1: if the request type (status = 1 ) i.e complete**

**Return {**

"status": true,

"errNum": 0,

"msg": "",

"data": {

"request\_id": 3,

"status": "1",

"paid": "1",

"paid\_amount": 300,

"start\_date": "2019-04-27",

"end\_date": "2019-04-30",

"expire\_date": "2019-04-30",

"daysCount": 3,

"totalCost": 30

}

# Get List Of Pages: -

**url** => http://sm.wisyst.info/public/api[/](http://mjrb.wisyst.info/api/activateAccount)GetPages

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

**If success**

Return {

"status": true,

"errNum": 0,

"msg": "تمت العملية بنجاح",

'page': [

{

'id': 1,

'title': ""

},{

'id': 1,

'title': ""

}

]

}

# Get Page Data: -

**url** =>  [http://sm.wisyst.info/public/api/](%20http://sm.wisyst.info/public/api/)GetPage

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

**If success**

Return {

"status": true,

"errNum": 0,

"msg": "تمت العملية بنجاح",

'page': {

'id': 1,

'title': "",

‘content’

}

}

# Get Usage Agreement Page: -

**url:** http://sm.wisyst.info/public/api/UsageAgreement

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

**If success**

Return {

"status": true,

"errNum": 0,

"msg": "تمت العملية بنجاح",

'page': {

'id': ,

'title': "",

‘content’

}

}

# Get Ticket Types: -

**url** => [://sm.wisyst.info/public/api/](http://mjrb.wisyst.info/api/activateAccount)GetTicketTypes

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg': msg

}

If success

Return

{

"status": true,

"errNum": 0,

"msg": "success",

"types": [

{

"type\_id":,

"type\_name": ""

},

{

"type\_id":,

"type\_name": ""

},

……………………………….

]

}

# Fetch meal API: -

**url** => http://zad.al-yasser.info/public/api/getMealDetails

**params** => (meal\_id, lang)

**1-** meal\_id: (required|numeric) -> represent visited meal id

**2-** lang : (in:ar,en) -> represent app language

**If fails**

return {

'status' :false,

'errNum':1,

'msg' :'Invalid meal id'

}

**If success**

return {

"status": true,

"errNum": 0,

"meal": {

"meal\_id": 1,

"meal\_name": "تجربة وجبة",

"meal\_desc": "مجرد تجربة لوجبه علشان نشوف إيه الكلام",

"main\_image": (url),

"video": (url),

"meal\_rtp": 60, //real time price

"meal\_msrp": 60, //original price

"meal\_rate": 0,

"likes\_count": 0,

"full\_name": "Provider1",

"provider\_id": 1,

"cat\_name": meal category name,

"profile\_pic": "url",

"provider\_address": "string"

},

"userRate": 0 if user didn’t rate or there is no user logged in(info for ZAD user application),

"attaches": [

{

"attach\_id": 1,

"attach\_path": "http://www.test.com/public/test/filename.jpg"

},

{

"attach\_id": 1,

"attach\_path": "http://www.test.com/public/test/filename2.jpg"

}

],

"comments": [

{

"full\_name": "mohamed",

"profile\_pic": "url",

"user\_id": 1,

"comment": "تجربة كومنت",

"id": 1

},

{

"full\_name": "mohamed",

"profile\_pic": "url",

"user\_id": 1,

"comment": "تجربة كومنت",

"id": 1

}

],

"comments\_count": 1,

"reportFlag": 1 means report before, 0 means didn't

"msg": "Retrieved successfully"

}

# Delete meal API: -

**url** => http://zad.al-yasser.info/public/api/deleteMeal

**params** => (meal\_id, lang)

**1-** meal\_id: (required|numeric) -> represent visited meal id

**2-** lang : (default:en) -> represent app language

**If fails**

return {

'status' :false,

'errNum':errNum,

'msg' : msg

}

errNum =>

1 => 'meal\_id is required',

2 => 'failed to delete, please try again later'

**If success**

return {

"status": true,

"errNum": 0

"msg": "Deleted successfully"

}

# Fetch meals API: -

**url** => http://zad.al-yasser.info/public/api/providerMeals

**params** => (provider\_id,lang)

1. lang: (default:en) -> represent application displayed language
2. provider\_id: (required|numeric)-> represents provider id

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'Retrieved successfully ',

'meals' :{

'total': 30,

'per\_page': 10,

'current\_page': 1,

'last\_page': 3,

'next\_page\_url': "https://zad.al-yasser.info/public/api/providerMeals?page=2",

'prev\_page\_url': null,

'form': 1,

'to': 10,

'data': [

{

"meal\_id": 1,

"meal\_name": "name1",

"meal\_desc": "desc1",

"meal\_rtp": "20", // real time price

"meal\_msrp": "20", // original price

"main\_image": "url",

"likes\_count": "0",

"meal\_rate": "0",

"likeFlag": "0" 0 if user is not logged in or doesn't like the meal (TO USE IN USER API),

},

{

"meal\_id": 2,

"meal\_name": "name2",

"meal\_desc": "desc2",

"meal\_rtp": "25", // real time price

"meal\_msrp": "25", //original price

"main\_image": "url",

"likes\_count": "10",

"meal\_rate": "3",

"likeFlag": "1" 1 if user likes the mea (TO USE IN USER API)l,

},

………………..

],

}

}

# SMS API: -

**url** => http://zad.al-yasser.info/public/api/sms

**params** => (lang, phone)

1. lang: (default:en) -> represent application displayed language
2. phone: (required) -> represent provider phone

**If fails**

return {

'status' :false,

'errNum':errNum,

'msg' :msg

}

errNum =>

1 => 'Failed to send the message',

2 => 'Phone is required'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Message sent successfully' ,

'code' : 'sent code'

}

# Prepare update meal API: -

**url** => http://zad.al-yasser.info/public/api/prepareMealUpdate

**params** => (provider\_id, meal\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: (required|numeric)-> represents provider id

3- meal\_id : (required) -> represents meal id

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'Both meal\_id and provider\_id are required'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

"meal": {

"meal\_name": "تجربة وجبه",

"meal\_desc": "تجربة وصف الوجبه",

"meal\_rtp": "10",

"main\_image": "<https://zad.al-yasser.info/public/mealsImages/img-2mJg.jpg>",

"video": (url),

"provider\_name": "مطعم العنود",

"qty": 20,

"cat\_id": 1

},

"attaches": [

{

"attach\_id": 1,

"attach\_path": "http://zad.al-yasser.info/public/mealsImages/attach1.jpg",

"type": "image"

},

{

"attach\_id": 2,

"attach\_path": "http://zad.al-yasser.info/public/mealsImages/attach2.jpg",

"type": "image"

},

……….

],

"cats": [

{

"cat\_id": 1,

"cat\_name": "وجبات الغداء",

"chosen": 1

},

{

"cat\_id": 2,

"cat\_name": "وجبات العشاء",

"chosen": 0

},

{

"cat\_id": 3,

"cat\_name": "وجبات الافطار",

"chosen": 0

}

]

}

# Update meal API: -

**url** => http://zad.al-yasser.info/public/api/updateMeal

**params** => (meal\_id, provider\_name, provider\_id, meal\_name, cat\_id, description, qty, price, main\_image, main\_image\_ext, meal\_images, deleted\_images, video || video\_data, lang)

1- lang: (default:en) -> represent application displayed language

2- meal\_id: ( required) -> represent meal id

3- provider\_id: (required|numeric)-> represents provider id

4- provider\_name: (required) -> represents provider name

5- meal\_name: (required) -> represents meal name

6- cat\_id: (required) -> represents selected category id

7- description: (required) -> represents meal description

8- qty: (required|numeric) -> represents allowed quantity of the meal

9- price: (required|numeric) -> represents meal price

10- main\_image: (optional) -> represents string of base64 image data of the main image of the meal

11- main\_image\_ext: (required\_with: main\_image) -> represents main image extension

12- meal\_images: (optional) -> represents array of meal images ex:- [['image'=> string of image data, 'image\_ext'=> image extension], ['image'=> string of image data, 'image\_ext'=> image extension], ….]

13- deleted\_images: (optional) -> represents array of deleted images ids [1,2,3, …]

14- video\_data: (optional) -> represents meal video string of data

15- video: (optional) -> represents meal video file

**Note:**  you can use video\_data or video but you can’t use them both

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'All fields are required',

2 => 'Both quantity and price must be a number',

3 => 'Failed to upload image, please try again later',

4 => 'Failed to update the meal, please try again later'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'Meal updated successfully'

}

# Categorized orders screen API: -

**url** => http://zad.al-yasser.info/public/api/ordersCounts

**params** => ( provider\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: (required|numeric)-> represents provider id

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'provider\_id is required'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

'new\_orders\_count': 13,

'current\_orders\_count': 0,

'future\_orders\_count': 0,

'old\_orders\_count': 0,

'complains\_count': 32

}

# Fetch provider orders API: -

**url** => http://zad.al-yasser.info/public/api/getProviderOrders

**params** => ( provider\_id, type, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: (required|numeric)-> represents provider id

3- type: (required|in:1,2,3,4) -> represents orders types (1 = new orders, 2 = current orders, 3 = futures orders, 4 = finished orders [delivered, failed to deliver or rejected])

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

"orders": {

"total": 13,

"per\_page": 10,

"current\_page": 1,

"last\_page": 2,

"next\_page\_url": "<http://zad.al-yasser.info/public/api/getProviderOrders?page=2>",

"prev\_page\_url": null,

"from": 1,

"to": 10,

"data": [

{

"order\_id": 1,

"delivery\_id": 1 will be equal zero if there is no delivery man,

"provider\_name": "مطعم العنود",

"address": "test",

"payment\_method": "الدفع عند الإستلام",

"delivery\_method": "ارامكس",

"meals\_count": 2,

"user\_name": "userName",

"total\_value": 200,

"created\_date": "2017-05-11",

"created\_time": "13:36:05"

},

{

"order\_id": 2,

"delivery\_id": 1 will be equal zero if there is no delivery man,

"provider\_name": "مطعم العنود",

"address": "test",

"payment\_method": "الدفع عند الإستلام",

"delivery\_method": "ارامكس",

"meals\_count": 2,

"user\_name": "userName",

"total\_value": 200,

"created\_date": "2017-05-11",

"created\_time": "13:36:05"

},

………………………

]

}

"time\_counter\_in\_min": 10,

"time\_counter\_in\_hours": 0.16,

"today\_date": "2017-04-16",

"now":"02:30:56",

}

# Order acceptance API: -

**url** => http://zad.al-yasser.info/public/api/orderAcceptance

**params** => ( order\_id, provider\_id, type, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: (required|numeric)-> represents provider id

3- order\_id: ( required) -> represents order id

4- type: (required|in:accept,reject) -> represents process types (accept = to accept the order, reject = to reject it)

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'order\_id is required',

2 => 'provider\_id is required',

3 => 'type is required',

4 => 'type must be (accept or reject)',

5 => 'Process failed please try again later'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'Process done successfully'

}

# Get order details API: -

**url** => http://zad.al-yasser.info/public/api/orderDetails

**params** => (order\_id, lang)

1. lang: (default:en) -> represent application displayed language
2. order\_id: (required|numeric)-> represents order id

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Retrieved successfully' ,

"header": {

"order\_code": "42345454 ",

"total": "180", //note that total value is the order price + order delivery price

"app\_value": 20,

"marketer\_delivery\_value": 20,

"marketer\_provider\_value": 20,

"user\_address": "test address",

"user\_phone": "1234567890",

"user\_email": "test1@test.com",

"delivered\_at": "0000-00-00 00:00:00",

"payment\_method": "Cash On Delivery",

"delivery\_method": "Aramex",

"delivery\_method\_id": 1,

"provider\_name": "مطعم العنود",

"user\_name": "user name"

"order\_type": " لاحقا",

"order\_status": " جارى التسليم",

"status\_id": 3,

"order\_date": "2017-05-25"

},

'details' : [

{

"meal\_name": meal name,

"meal\_price": 60,

"meal\_desc":Description of meal,

"discount": 0,

"qty": 2

},

{

"meal\_name": meal name,

"meal\_price": 60,

"meal\_desc":Description of meal,

"discount": 0,

"qty": 2

},

…….

],

"order\_status": [

{

"status\_id": 2,

"order\_status": "Processing"

},

{

"status\_id": 3,

"order\_status": "In way"

},

{

"status\_id": 4,

"order\_status": "Delivered"

},

{

"status\_id": 5,

"order\_status": "Failed to deliver"

},

{

"status\_id": 6,

"order\_status": "Refused"

}

]

"app\_percentage": 2 ,

"order\_rate": 3 // note if the order status was not delivered then this attribute will be equal ""

}

# Fetch complains API: -

**url** => http://zad.al-yasser.info/public/api/getComplains

**params** => (provider\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: (required|numeric)-> represents provider id

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'provider\_id is required'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : '' ,

"data": [

{

"user\_name": "محمد صلاح",

"order\_id": 1,

"order\_code": "OC^hTrN1",

"complain": "تجربة",

"attaches": [

{

"id": 482,

"attach\_path": "http://zad.al-yasser.info/public/mealsImages/attach4.jpg",

"type": "image"

}

]

},

{

"user\_name": "محمد صلاح",

"order\_id": 1,

"order\_code": "AB5hUrS5",

"complain": "adsfadfdf",

"attaches": []

},

………………..

]

}

# Change order status API: -

**url** => http://zad.al-yasser.info/public/api/changeOrderStatus

**params** => (order\_id, status\_id , provider\_id, lang, delivery\_id)

1- lang: (default:en) -> represent application displayed language

2- order\_id: ( required) -> represents order id

3- provider\_id: (required) -> represents provider id

3- stauts\_id: (required|in:2,3,4,5)-> represents status id (2 = processing, 3= in way, 4= Delivered, 5= failed to delivered )

5- delivery\_id: (required\_if(status\_id = 3 & order delivery method = 1)) -> represents selected delivery man

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'All fields are required',

2 => 'Process failed, please try again later',

3 => 'status\_id must be 2,3,4 or 5'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Process done successfully'

}

# Fetch provider credit data API: -

**url** => http://zad.al-yasser.info/public/api/getBalances

**params** => (provider\_id , lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: ( required) -> represents provider id

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'provider\_id is required'

**If success**

return {

"status": true,

"errNum": 0,

"msg": "",

"balance": {

"due\_balance": 0,

"current\_balance": 0,

"forbidden\_balance": 0

},

"bank\_name": “”,

"bank\_username":””,

"bank\_phone":””,

"account\_num": “”

}

# Request to withdraw credit API: -

**url** => http://zad.al-yasser.info/public/api/withdraw\_request

**params** => (actor\_id, type , current\_balance, due\_balance , forbidden\_balance , lang , bank\_name , name , **account\_num , phone**)

1- lang: (default:en) -> represent application displayed language

2- actor\_id: ( required) -> represents provider id or marketer\_id

3- current\_balance: ( required) -> represents provider current balance

4- due\_balance: ( required) -> represents provider due balance

5-bank\_name: required -> represents bank name

6-name: required -> represents name of the user

7-account\_num : required -> represents the account number

8-phone: required -> represents user phone

9- type : required -> represent provider or marketer

10 - forbidden\_balance -> required and numeric , represent actor forbidden balance

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'provider\_id is required',

2 => 'current\_balance is required',

3 => 'due\_balance is required',

4 => 'Process failed, please try again later'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Process done successfully',

}

# Prepare order properties data API: -

**url** => http://zad.al-yasser.info/public/api/order\_properties

**params** => (provider\_id, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: ( required) -> represents provider id

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'provider\_id is required'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : '',

"provider\_properties": {

"allowed\_from\_time": null,

"allowed\_to\_time": null,

"delivery\_price": 0,

"receive\_orders": 0,

"current\_orders": 0,

"future\_orders": 0,

"avail\_date": "2017-04-28",

"last\_updated": "2017-05-15"

},

"order\_times": [

{

"from\_time": "03:24:30",

"to\_time": "04:24:30"

},

{

"from\_time": "05:08:30",

"to\_time": "07:08:30",

}

],

"deliveries": [

{

"delivery\_id": 1,

"delivery\_name": "ارامكس",

"choosen": 1 (1 means that the provider already using it)

},

{

"delivery\_id": 2,

"delivery\_name": "سوق.كوم",

"choosen":0 (0 means that the provider doesn’t use it yet)

},

{

"delivery\_id": 3,

"delivery\_name": "زاجل",

"choosen": 1

},

{

"delivery\_id": 4,

"delivery\_name": "دى إتش إل",

"choosen": 0 (0 means that the provider doesn’t use it yet)

}

],

'editFlag' :1 (0 or 1 - 0 means not allowed to edit, 1 means it is ok to edit),

'max\_edit\_date' :"2017-06-28" (it will be (null) if editFlag is 0),

'today\_date' : '2017-05-29',

}

# Save order properties API: -

**url** => http://zad.al-yasser.info/public/api/save\_order\_properties

**params** => (provider\_id, allowed\_from\_time, allowed\_to\_time, delivery\_price, current\_orders, future\_orders, avail\_date, receive\_orders, delivery\_methods, exit\_order\_times, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: ( required) -> represents provider id

3- allowed\_from\_time: (required) -> represents start time at the allowed period of the day to receive orders

4- allowed\_to\_time: (required) -> represents end time at the allowed period of the day to receive orders

5- delivery\_price: (required) -> represents provider delivery price

6- current\_orders: (required|in:0,1) -> represents ability of the provider to receive current order or not (0 = no, 1= yes)

7- future\_orders: (required|in:0,1) -> represents ability of the provider to receive future order or not (0 = no, 1 = yes)

8-receive\_orders: (required|in:0,1) -> represents ability of the provider to receive orders or not (0 = no, 1 = yes)

9-avail\_date: (required|before 30 days from today only)-> represents the maximum date of the provider to receive future orders

10- delivery\_methods: (required) -> represents all delivery methods that the provider will use to deliver his order, it is an array of delivery methods ids (ex:- [1,2,3,...])

11- exit\_order\_times: (required) -> represents all times for the provider to issue his orders, it is an array of from and to arrays [[from, to], [from, to],[from,to]](ex:-[[09:00, 13:00],[ 22:00, 23:00],[14:00,16:00]])

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'provider is required',

2 => 'receive order time is required',

3 => 'receive order time is required',

4 => 'delivery price is required',

5 => 'receive current orders is required',

6 => 'receive future orders is required',

7 => 'to field is required',

8 => 'receive orders is required',

9 => 'receive current orders, receive future orders and receive orders can be only 0 or 1',

10 => 'to field must be in format (yyyy-mm-dd)',

11 => 'You can not update your to date before 30 days from last update',

12 => 'Failed to set the new settings, please try again later',

13 => 'Time to issue orders is required',

14 => 'Delivery methods is required',

15 => 'Time to issue orders can not be repeated',

16 => 'to field must not be after 30 days from today',

17 => 'Time to issue can not have a null value',

18 => 'Delivery methods can not be repeated'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'New settings has been set successfully'

}

# Activate and deactivate receiving orders: -

**url** => http://zad.al-yasser.info/public/api/receiveOrderSwitch

**params** => (provider\_id, switch, lang)

1- lang: (default:en) -> represent application displayed language

2- provider\_id: ( required) -> represents provider id

3- switch: (required|in:0,1) -> represents switch value ( 0 means deactivate, 1 means activate)

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'All fields are required',

2 => 'Switch value must be between 0 and 1',

3 => 'Process failed',

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Receiving orders activated',

}

# Add complain API:-

**url** => http://zad.al-yasser.info/public/api/complain

**params** => (subject\_id, order\_id, actor\_id,files, extentions, types,complain, complaint, app,lang)

1. subject\_id: (required|numeric) -> represent the subject (provider or delivery) id
2. order\_id: (required|numeric|exists)-> represent order number
3. user\_id: (required|numeric|exists) -> represent user number
4. files: ( optional| array) -> represent array of base64 encoded files strings [sting of image, string of image, string of video, ...]
5. extentions: ( optional|array) -> represent array of attached files extensions [jpg, png, mp4, ….]
6. types: (optional|array) -> represent type of attached file ['image', 'image', 'video',.....]
7. complain: (required) -> represent the user text complain
8. complaint: (required|in:provider,delivery) -> determine if the complain is for a provider or a delivery in this app must be equal 'delivery'
9. app: (required|in:user,provider) -> represents the used app to add complain it must be equal 'provider'
10. lang: (default:en)->represent app displayed language

**If fails**

return {

'status' :false,

'errNum':errNum,

'msg' :msg

}

errNum =>

1 => 'Invalid parameters',

2 => 'Process failed, please try again later'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Process done successfully'

}

**Get provider followers API: -**

**url** => http://zad.al-yasser.info/public/api/getProviderFollowers

**params** => (provider\_id, paginate\_flag, lang)

1- provider\_id: (required|exists)->represent selected provider id

2- lang: (default:en)->represent app displayed language

3- paginate\_flag: (optional|default:0|in:0,1) -> flag to get paginated data or all data at once

**If fails**

return {

'status' :false,

'errNum':errNum,

'msg' :msg

}

errNum =>

1 => 'provider id is required',

2 => 'Invalid provider id',

3 => 'paginate\_flag must be 0 or 1'**If success**

**If** **success**

Case paginate\_flag = 0

return {

'status' :true,

'errNum' :0,

'msg' : '' ,

followers: [

{

"user\_id": 1,

"full\_name": "string name",

"profile\_pic": "url",

},

{

"user\_id": 2,

"full\_name": "string name",

"profile\_pic": "url",

},

…………..

]

}

Case paginate\_flag = 1

return {

'status' :true,

'errNum' :0,

'msg' : '' ,

followers: {

"total": 1,

"per\_page": 10,

"current\_page": 2,

"last\_page": 3,

"next\_page\_url": http://zad.al-yasser.info/public/api/getProviderFollowers?page=3,

"prev\_page\_url":http://zad.al-yasser.info/public/api/getProviderFollowers?page=1,

"from": 1,

"to": 3,

"data": [

{

"user\_id": 1,

"full\_name": "string name",

"profile\_pic": "url",

},

{

"user\_id": 2,

"full\_name": "string name",

"profile\_pic": "url",

},

…………..

]

}

}

# Get deliveries API: -

**url** => http://zad.al-yasser.info/public/api/getDeliveries

**params** => (city\_id, lang)

1- city\_id:(required) -> represents provider city id

2- lang: (default:en) -> represents the application displayed langauge

**If fails**

return {

'status' :false,

'errNum':1,

'msg' :'Provider city is required'

}

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'',

'deliveries' : [

{

"delivery\_id": 1,

"full\_name": "string name",

},

{

"delivery\_id": 1,

"full\_name": "string name",

},

…………..

]

}

# Marketer signUp API: -

**url** => http://zad.al-yasser.info/public/api/marketerSignUp

**params** => first\_name, second\_name, third\_name, last\_name, email, phone, country\_code, password, country\_id, city\_id, profile\_pic, address, lang, marketer\_token)

1- first\_name: (required) -> represent marketer first name (string)

2- second\_name: (required) -> represent marketer second name (string)

3- third\_name: (required) -> represent marketer third name (string)

4- last\_name: (required) -> represent marketer last name (string)

5- email: (required | email format|unique) -> represents marketer email (string)

6- phone: (required |unique) -> represents marketer phone (long || string)

7- password: (required | MIN: 8) -> represents marketer account password (string)

8- profile\_pic(optional)-> marketer profile pic string of data encoded in base64 (string)

9- country\_id (required|numeric) -> represents country id(integer)

10- city\_id (required|numeric) -> represents city id(integer)

11- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

12 - country\_code (required) -> represent marketer phone country code

13- address(required) -> represent marketer address (string)

14 - marketer\_token(required) -> represents marketer account firebase reg id

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'All fields are required',

2 => 'Invalid e-mail address',

3 => 'Password must not be less than 8 characters',

4 => 'Phone is already used',

5 => 'E-mail is already used',

6 => 'Failed to register, please try again later',

7 => 'Failed to upload image, please try again later'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'data': {

"id": 2,

"marketer\_name": "Mohamed Salah Tawfik Farag",

"first\_name": "Mohamed",

"second\_name": "Salah",

"third\_name": "Tawfik",

"last\_name": "Farag",

"email": "mohamed@wisyst.com",

"phone": "01228265463",

"country\_code": "2",

"profile\_pic": "https://zad.al-yasser.info/public/marketerProfileImages/img-tneJ.jpg",

"address": "test address",

"country\_id": 1,

"city\_id": 1,

"status": 0,

"marketer\_code": ‘auto generated code’,

"created": "2017-08-01"

},

'msg' :'Signed up successfully'

}

# Marketer edit profile API: -

**url** => http://zad.al-yasser.info/public/api/marketerEditProfile

**params** => first\_name, second\_name, third\_name, last\_name, email, password, country\_id, city\_id, profile\_pic, address, lang, id)

1- first\_name: (required) -> represent marketer first name (string)

2- second\_name: (required) -> represent marketer second name (string)

3- third\_name: (required) -> represent marketer third name (string)

4- last\_name: (required) -> represent marketer last name (string)

5- email: (required | email format|unique) -> represents marketer email (string)

6- password: (optional | MIN: 8) -> represents marketer account password (string)

7- profile\_pic(optional)-> marketer profile pic string of data encoded in base64 (string)

8- country\_id (required|numeric) -> represents country id(integer)

9- city\_id (required|numeric) -> represents city id(integer)

10- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

11- address(required) -> represent marketer address (string)

12- id(required) -> represent marketer id

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'All fields are required',

2 => 'Invalid e-mail',

3 => 'Password can not be less than 8 characters',

4 => 'This e-mail is already used',

5 => 'Failed to update, please try again later',

6 => 'Failed to upload the image'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'data': {

"id": 2,

"marketer\_name": "Mohamed Salah Tawfik Farag",

"first\_name": "Mohamed",

"second\_name": "Salah",

"third\_name": "Tawfik",

"last\_name": "Farag",

"email": "mohamed@wisyst.com",

"phone": "01228265463",

"country\_code": "2",

"profile\_pic": "https://zad.al-yasser.info/public/marketerProfileImages/img-tneJ.jpg",

"address": "test address",

"country\_id": 1,

"city\_id": 1,

"status": 0,

"marketer\_code": ‘auto generated code’,

"created": "2017-08-01"

},

'msg' :'Updated successfully'

}

# Activate marketer API: -

**url** => http://zad.al-yasser.info/public/api/activateMarketer

**params** =>(id, lang)

1- id: (required) -> represent marketer id

2- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'marketer\_id is required',

2 => 'Failed to activate, please try again later'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'data': {

"id": 2,

"marketer\_name": "Mohamed Salah Tawfik Farag",

"first\_name": "Mohamed",

"second\_name": "Salah",

"third\_name": "Tawfik",

"last\_name": "Farag",

"email": "mohamed@wisyst.com",

"phone": "01228265463",

"country\_code": "2",

"profile\_pic": "https://zad.al-yasser.info/public/marketerProfileImages/img-tneJ.jpg",

"address": "test address",

"country\_id": 1,

"city\_id": 1,

"status": 0,

"marketer\_code": ‘auto generated code’,

"created": "2017-08-01"

},

'msg' :'Activated successfully'

}

# Edit marketer phone API: -

**url** => http://zad.al-yasser.info/public/api/editMarketerPhone

**params** =>(id, phone, country\_code, lang)

1- id: (required) -> represent marketer id

2- phone: (required|unique) -> represents marketer phone

3-country\_code: (required) -> represent marketer id

4- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'marketer\_id is required',

2 => 'Phone is required',

3 => 'Country code is required',

4 => 'Failed to update, please try again later',

5 => 'Phone number is used before'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'Updated successfully'

}

# Marketer Login API: -

**url** => http://zad.al-yasser.info/public/api/marketerLogin

**params** =>(phone, password, marketer\_token, lang)

1- phone: (required) -> represents marketer phone

2-password: (required) -> represent marketer password

3-marketer\_token: (required) -> represent marketer device reg id

4- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg,

}

errNum =>

1 => 'Phone is required',

2 => 'Wrong phone number',

3 => 'Password is required',

4 => 'Wrong data',

5 => 'You need to activate your account',

6 => 'marketer\_token is required'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'Logined successfully',

'data': {

"id": 2,

"marketer\_name": "Mohamed Salah Tawfik Farag",

"first\_name": "Mohamed",

"second\_name": "Salah",

"third\_name": "Tawfik",

"last\_name": "Farag",

"email": "mohamed@wisyst.com",

"phone": "01228265463",

"country\_code": "2",

"profile\_pic": "https://zad.al-yasser.info/public/marketerProfileImages/img-tneJ.jpg",

"address": "test address",

"country\_id": 1,

"city\_id": 1,

"status": 0,

"marketer\_code": ‘auto generated code’,

"created": "2017-08-01"

},

}

# Marketer get balance API: -

**url** => http://zad.al-yasser.info/public/api/getMarketerBalances

**params** =>(marketer\_id, lang)

1- marketer\_id: (required) -> represents marketer id

2- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg,

}

errNum =>

1 => 'marketer\_id is required'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

{

"status": true,

"errNum": 0,

"msg": "",

"balance": {

"current\_balance": 0,

"due\_balance": 0,

"updated\_at": "2017-11-15 13:36:11"

},

"bank\_name": null,

"bank\_username": null,

"bank\_phone": null,

"account\_num": null

}

# withdraw Reports API: -

**url** => http://zad.al-yasser.info/public/api/withdrawReport

**Note:- this one can be used for delivery application also**

**params** =>(actor\_id, type, from\_date, to\_date, lang)

1- actor\_id: (required) -> represents user id this user can be provider, delivery or user

2- type: (required|in:provider, delivery, user) -> determine who use the api (provider, delivery or user)

3- from\_date: (nullable|required\_with:to\_date, format:yyyy-mm-dd) -> represents period start date

4- to\_date: (nullable|required\_with:from\_date, format:yyyy-mm-dd) -> represents period end date

5- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg,

}

errNum =>

1 => 'actor\_id is required',

2 => 'type is required',

3 => 'type must be in provider, delivery, user',

4 => 'date format must be yyyy-mm-dd',

5 => 'from date required with to date in search with period'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'',

'requests': [

{

"current\_balance": 100,

"due\_balance": 10,

"credit": 90,

"name": "سيف الدينن ابراهيم سعد",

"status": "بإنتظار الرد",

"created": "2017-05-22"

},

{

"current\_balance": 100,

"due\_balance": 10,

"credit": 90,

"name": "سيف الدينن ابراهيم سعد",

"status": "بإنتظار الرد",

"created": "2017-05-22"

},

……………...

]

}

# Income report API: -

**url** => http://zad.al-yasser.info/public/api/income

**Note:- this one can be used for delivery application also**

**params** =>(actor\_id, type, from\_date, to\_date, lang)

1- actor\_id: (required) -> represents user id this user can be provider, delivery or user

2- type: (required|in:provider, delivery) -> determine who use the api (provider or delivery)

3- from\_date: (nullable|required\_with:to\_date, format:yyyy-mm-dd) -> represents period start date

4- to\_date: (nullable|required\_with:from\_date, format:yyyy-mm-dd) -> represents period end date

5- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg,

}

errNum =>

1 => 'actor\_id is required',

2 => 'type is required',

3 => 'type must be in provider, delivery',

4 => 'date format must be yyyy-mm-dd',

5 => 'from date required with to date in search with period'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'',

'result': [

{

"order\_code": "OC^hTrN1",

"total\_qty": 3,

"total\_value": "180",

"credit": 0,

"mealsCount": 2,

"order\_id": 1

},

{

"order\_code": "OC^hTrN1",

"total\_qty": 3,

"total\_value": "180",

"credit": 0,

"mealsCount": 2,

"order\_id": 1

},

……………...

]

}

# Total income report API: -

**url** => http://zad.al-yasser.info/public/api/totalIncome

**Note:- this one can be used for delivery application also**

**params** =>(actor\_id, type, from\_date, to\_date, lang)

1- actor\_id: (required) -> represents user id this user can be provider, delivery or user

2- type: (required|in:provider, delivery) -> determine who use the api (provider or delivery)

3- from\_date: (nullable|required\_with:to\_date, format:yyyy-mm-dd) -> represents period start date

4- to\_date: (nullable|required\_with:from\_date, format:yyyy-mm-dd) -> represents period end date

5- lang (Default:en) -> marketer application displayed language (string = "ar" || "en")

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg,

}

errNum =>

1 => 'actor\_id is required',

2 => 'type is required',

3 => 'type must be in provider, delivery',

4 => 'date format must be yyyy-mm-dd',

5 => 'from date required with to date in search with period'

**Note:- if attribute (lang) is equal 'ar' these messages will be returned in Arabic**

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'',

'result': [

{

"order\_code": "OC^hTrN1",

"invo\_no":1,

"total\_qty": 3,

"total\_value": "180",

"credit": 0,

"mealsCount": 2,

"order\_id": 1

},

{

"order\_code": "OC^hTrd1",

"invo\_no":2,

"total\_qty": 3,

"total\_value": "180",

"credit": 0,

"mealsCount": 2,

"order\_id": 2

},

……………...

]

}

# Get marketer clients API: -

**url** => http://zad.al-yasser.info/public/api/getMarketerClients

params => (marketer\_id,lang)

1. lang: (default:en) -> represent application displayed language
2. marketer\_id: (required|numeric)-> represents marketer id

**if failed**

return {

'status' :true,

'errNum' :1,

'msg' :'There is no data for the sent marketer',

}

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'Retrieved successfully ',

'code' :'klsfj734',

'result' :{

'total': 30,

'per\_page': 10,

'current\_page': 1,

'last\_page': 3,

'next\_page\_url': "https://zad.al-yasser.info/public/api/getMarketerClients?page=2",

'prev\_page\_url': null,

'form': 1,

'to': 10,

'data': [

{

"id": 1,

"full\_name": "name1",

"profile\_pic": "url",

"type": "provider",

"created": "2017-08-27 12:23:54"

},

{

"id": 1,

"full\_name": "name1",

"profile\_pic": "url",

"type": "delivery",

"created": "2017-08-27 12:23:54"

},

………………..

],

}

}

# Get marketer balance details API: -

**url** => http://zad.al-yasser.info/public/api/marketerBalanceDetails

params => (marketer\_id,lang,from,to)

1-lang: (default:en) -> represent application displayed language

2-marketer\_id: (required|numeric)-> represents marketer id

3-from: (optional|date\_formate:yyyy-mm-dd)-> represents from date

4-to: (optional|date\_formate:yyyy-mm-dd)-> represents to date

**if failed**

return {

'status' :true,

'errNum' :1,

'msg' :'There is no data for the sent marketer',

}

**If success**

return {

'status' :true,

'errNum' :0,

'msg' :'Retrieved successfully ',

'balance': [

{

"full\_name": "name1",

"total\_value":"150",

"marketer\_value":"22.5",

"marketer\_percentage": "15"

},

{

"full\_name": "name2",

"total\_value":"150",

"marketer\_value":"22.5",

"marketer\_percentage": "15"

},

],

}

# Get marketer Data: -

**url** => http://zad.al-yasser.info/public/api/prepareEditMarketerProfile

params => (marketer\_id,lang)

1-lang: (default:en) -> represent application displayed language

2-marketer\_id: (required|numeric)-> represents marketer id

**if failed**

return {

'status' :false,

'errNum' :,

'msg' :'',

}

**If success**

return {

"status": true,

"errNum": 0,

"msg": "تم استرجاع البيانات بنجاح",

"data": {

"id": 1,

"marketer\_name": "محمد محمد محمد امير",

"first\_name": "محمد",

"second\_name": "محمد",

"third\_name": "محمد",

"last\_name": "امير",

"email": "amir@yahoo.com",

"phone": "01001598248",

"country\_code": "966",

"city": "مكة",

"country": "المملكه السعودية العربية",

"profile\_pic": "http://zad.al-yasser.info/public/avatar\_ic.png",

"address": "مكة",

"country\_id": 1,

"city\_id": 87,

"status": 1,

"publish": 1,

"marketer\_code": "7M4gOJi",

"created": "2017-10-02",

"updated": "2017-12-17"

},

"cities": [

{

"city\_id": 175,

"city\_name": "Al Medayaleef",

"chosen": 0

},{

"city\_id": 175,

"city\_name": "Al Medayaleef",

"chosen": 0

}

],

"countries": [

{

"country\_id": 1,

"country\_name": "Saudi Arabia",

"chosen": 1,

"country\_code": "966"

}

]

}

# Get Actor Receive Orders :-

**url** => http://zad.al-yasser.info/public/api/isReceiveOrders

params => (actor\_id,lang , type)

1-lang: (default:en) -> represent application displayed language

2-actor\_id: (required|numeric)-> represents provider or delivery id

3- type :- (required) -> (“provider” or “delivery”)

**if failed**

return {

'status' :false,

'errNum' :,

'msg' :'',

}

**If success**

return {

"status" : true,

"errNum" : 0,

"msg" : "success",

"receive\_orders" : 1

}