Mathaq delivery application API

# API base url:

http://zad.al-yasser.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)

# 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://zad.al-yasser.info/public/api/[deliveryPrepareSignUp](https://dev.al-yasser.info/zadapi/public/api/sendMail)

**params** => ( lang)

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

return {

'status' :true,

'errNum':0,

'msg' :"",

'countries': [

{

'country\_id': 1,

'country\_name': "المملكه السعودي",

'country\_code': 966

},

{

'country\_id': 2,

'country\_name': "مصر",

'country\_code': 002

},

…….

]

}

# Get country cities API: -

**url** => http://zad.al-yasser.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': 002

},

{

"city\_id": 2,

"city\_name": "string name",

'country\_code': 002

},

…………..

]

}

# delivery signUp API: -

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

**params** => (firstname, secondname, thirdname, lastname, country\_id, city\_id, email, phone, country\_code, password, car\_type, longitude, latitude, profile\_pic, image\_ext,car\_img1, car\_img2, car\_img3, insurance\_img, license\_img, license\_date, id\_img,id\_date , authorization\_img, lang, token, marketer\_code, cats)

1- firstname: (required) -> represent delivery first name (string)

2- secondname: (required) -> represent delivery second name (string)

3- thirdname: (required) -> represent delivery third name (string)

4- lastname: (required) -> represent delivery last name (string)

5- images: (required)-> represents delivery papers [[image:’string of data’,ext:’jpd’],......]

6- email: (required | email format|unique) -> represents delivery email (string)

7- phone: (required |unique) -> represents delivery phone (long || string)

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

9- longitude: (required) -> represents delivery location on map (double || string)

10 - latitude: (required) -> represents delivery location on map (double || string)

11- profile\_pic(optional)-> delivery profile pic string of data encoded in base64 (string)

12- image\_ext(required\_with:profile\_pic) -> represents delivery profile pic extension (string)

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

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

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

16 - country\_code (required) -> represent delivery phone country code

17- car\_type (required) -> represents delivery car type

18- car\_img1: (required|mimes: jpeg, png) -> represents string of data of car image 1

19- car\_img2: (required|mimes: jpeg, png) -> represents string of data of car image 2

20- car\_img3: (required|mimes: jpeg, png) -> represents string of data of car image 3

21- id\_img: (required|mimes: jpeg, png) -> represents string of data of delivery id image

22- authorization\_img: (required|mimes: jpeg, png) -> represents string of data of authorization image

23- license\_img: (required|mimes: jpeg, png) -> represents string of data of license image

24- insurance\_img: (required|mimes: jpeg, png) -> represents string of data of delivery insurance image

25- token: (required) -> represent account firebase reg id

26- marketer\_code: (optional) -> represents marketer code

27 - license\_date : (required) ->represent his license exp date

28 - id\_date : (required) -> represent his id exp date

29 - cats: (required) -> represents array of chosen categories [1,2,3,...]

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'All fields are required',

2 => 'country and city must be numeric',

3 => 'E-mail must be in email format',

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

5 => 'Failed to upload images',

6 => 'Failed to sign up, please try again later',

7 => 'Repeated email',

8 => 'Repeated phone'

10 => 'categories required',

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

**If success**

return {

'status' :true,

'errNum':0,

'delivery': {

'delivery\_id' :delivery id,

'delivery\_name':delivery fullName,

'firstname':delivery firstname,

'secondname':delivery secondname,

'thirdname':delivery thirdname,

'lastname':delivery lastname,

'phone':delivery phone,

country\_code:delivery phone country code,

'email':delivery email,

'profile\_pic' :delivery image(url),

car\_img1 :delivery car image 1(url),

'car\_img2' :delivery car image 2(url),

'car\_img3' :delivery car image 3(url),

'insurance\_img' :delivery insurance image(url),

'license\_img' :delivery license image(url),

'id\_img' :delivery id image(url),

'authorization\_img' :delivery authorization image(url),

'longitude' :delivery longitude,

'latitude':delivery latitude,

car\_type:delivery car type,

'Device\_reg\_id':string represents account firebase reg id,

'country\_id': country id,

'delivery\_rate': delivery rate in decimal,

'city\_id': city id,

'city\_name': city name,

'created': account creation time in format (yyyy-mm-dd)

},

'msg' :'Signed up successfully'

}

# 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 user 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'

}

# Delivery activate API: -

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

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

1- delivery\_id: (required) -> represent delivery id

2- type: (default:register|in:register, edit) -> determine if the activation for registration or editing the phone number

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

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'delivery\_id is required',

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

3 => 'type must be in (register, edit)'

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

**If success**

return {

'status' :true,

'errNum':0,

'msg' :'Activated successfully, and waitting for management approval',

'delivery': {

'delivery\_id' :delivery id,

'delivery\_name':delivery fullName,

'firstname':delivery firstname,

'secondname':delivery secondname,

'thirdname':delivery thirdname,

'lastname':delivery lastname,

'phone':delivery phone,

country\_code:delivery phone country code,

'email':delivery email,

'profile\_pic' :delivery image(url),

car\_img1 :delivery car image 1(url),

'car\_img2' :delivery car image 2(url),

'car\_img3' :delivery car image 3(url),

'insurance\_img' :delivery insurance image(url),

'license\_img' :delivery license image(url),

'id\_img' :delivery id image(url),

'authorization\_img' :delivery authorization image(url),

'longitude' :delivery longitude,

'latitude':delivery latitude,

car\_type:delivery car type,

'Device\_reg\_id':string represents account firebase reg id,

'country\_id': country id,

'delivery\_rate': delivery rate in decimal,

'city\_id': city id,

'city\_name': city name,

'created': account creation time in format (yyyy-mm-dd)

},

}

# Delivery login API: -

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

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

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

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

3- token: (required) -> represents delivery device firebase id

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

**If fails**

**case errNUm = 5 or 6**

return {

'status' :false,

'errNum':errNum,

' msg' : msg,

'delivery': {

'delivery\_id' :delivery id,

'delivery\_name':delivery fullName,

'firstname':delivery firstname,

'secondname':delivery secondname,

'thirdname':delivery thirdname,

'lastname':delivery lastname,

'phone':delivery phone,

country\_code:delivery phone country code,

'email':delivery email,

'profile\_pic' :delivery image(url),

car\_img1 :delivery car image 1(url),

'car\_img2' :delivery car image 2(url),

'car\_img3' :delivery car image 3(url),

'insurance\_img' :delivery insurance image(url),

'license\_img' :delivery license image(url),

'id\_img' :delivery id image(url),

'authorization\_img' :delivery authorization image(url),

'longitude' :delivery longitude,

'latitude':delivery latitude,

car\_type:delivery car type,

'Device\_reg\_id':string represents account firebase reg id,

'country\_id': country id,

'delivery\_rate': delivery rate in decimal,

'city\_id': city id,

'city\_name': city name,

'created': account creation time in format (yyyy-mm-dd)

},

}

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 => 'Waitting for management activation'

7 => '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',

'delivery': {

'delivery\_id' :delivery id,

'delivery\_name':delivery fullName,

'firstname':delivery firstname,

'secondname':delivery secondname,

'thirdname':delivery thirdname,

'lastname':delivery lastname,

'phone':delivery phone,

country\_code:delivery phone country code,

'email':delivery email,

'profile\_pic' :delivery image(url),

car\_img1 :delivery car image 1(url),

'car\_img2' :delivery car image 2(url),

'car\_img3' :delivery car image 3(url),

'insurance\_img' :delivery insurance image(url),

'license\_img' :delivery license image(url),

'id\_img' :delivery id image(url),

'authorization\_img' :delivery authorization image(url),

'longitude' :delivery longitude,

'latitude':delivery latitude,

car\_type:delivery car type,

'Device\_reg\_id':string represents account firebase reg id,

'country\_id': country id,

'delivery\_rate': delivery rate in decimal,

'city\_id': city id,

'city\_name': city name,

'created': account creation time in format (yyyy-mm-dd)

},

}

# Send mail API: -

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

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

1- email: (required) -> represent delivery email

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

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'Delivery email is required',

2 => 'Delivery email must be in email format',

3 => 'This email is not exist',

4 => 'Failed to send, 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,

'msg' :'Please check your mail to reset your password'

}

# Forget password page: -

Make forget password? link redirect to this url =>

https://zad.al-yasser.info/public/api/dForgetPass/{deliveryIid}

# Orders count screen API: -

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

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

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

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

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'delivery\_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 delivery complains API: -

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

**params** => (delivery\_id, complain\_from,lang)

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

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

3- complain\_from: (required|in(provider, user, both))-> represents who is the owner of the complain

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'delivery\_id is required',

2 => 'complain\_from is required',

3 => 'complain\_from must be in (provider, user, both)'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : '' ,

"data": [

{

"user\_name": "محمد صلاح",

"provider\_name": "",

"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": "",

"provider\_name": "مطعم العنود",

"order\_id": 1,

"order\_code": "AB5hUrS5",

"complain": "adsfadfdf",

"attaches": []

},

………………..

]

}

# Prepare edit profile API: -

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

**params** => (delivery\_Id, lang)

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

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

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'delivery\_id is required'

**If success**

return {

'status' :true,

'errNum':0,

'msg' :"",

'countries': [

{

'country\_id': 1,

'country\_name': "المملكه السعودي",

'country\_code': 966

},

{

'country\_id': 2,

'country\_name': "مصر",

'country\_code': 002

},

…….

],

'cities': [

{

"city\_id": 1,

"city\_name": "string name",

'country\_code': 002

},

{

"city\_id": 2,

"city\_name": "string name",

'country\_code': 002

},

…………..

],

'delivery': {

'delivery\_id' :delivery id,

'delivery\_name':delivery fullName,

'firstname':delivery firstname,

'secondname':delivery secondname,

'thirdname':delivery thirdname,

'lastname':delivery lastname,

'phone':delivery phone,

'country\_code:delivery phone country code,

'email':delivery email,

'profile\_pic' :delivery image(url),

'longitude' :delivery longitude,

'latitude':delivery latitude,

'attach\_id':integer,

' car\_type:delivery car type,

'country\_id': country id,

'delivery\_rate': delivery rate in decimal,

'city\_id': city id,

'city\_name': city name,

'created': account creation time in format (yyyy-mm-dd)

"admin\_activation\_date": "0000-00-00",

"admin\_activation\_time": "00:00:00"

},

'attachies': [

{

'id': 1,

'attach\_path': "link to attached image"

},

{

'id': 20,

'attach\_path': "link to attached image"

},

…….

],

"cats": [

{

"cat\_id": 1,

"cat\_name": "Launch",

"chosen": 1

},

{

"cat\_id": 2,

"cat\_name": "Dinner",

"chosen": 1

},

{

"cat\_id": 3,

"cat\_en\_name": "Breakfast",

"chosen": 1

},

……………..

]

}

# delivery edit profile API: -

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

**params** => (delivery\_Id, firstname, secondname, thirdname, lastname, country\_id, city\_id, email, phone, country\_code, password, car\_type, longitude, latitude, profile\_pic, image\_ext, lang, car\_img1, car\_img2, car\_img3, insurance\_img, license\_img, license\_date, id\_img,id\_date , authorization\_img)

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

2- firstname: (required) -> represent delivery first name (string)

3- secondname: (required) -> represent delivery second name (string)

4- thirdname: (required) -> represent delivery third name (string)

5- lastname: (required) -> represent delivery last name (string)

6- email: (required | email format|unique) -> represents delivery email (string)

7- password: (optional | MIN: 8) -> represents delivery account password (string)

8- longitude: (required) -> represents delivery location on map (double || string)

9 - latitude: (required) -> represents delivery location on map (double || string)

10- profile\_pic(optional)-> delivery profile pic string of data encoded in base64 (string)

11- image\_ext(required\_with:profile\_pic) -> represents delivery profile pic extension (string)

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

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

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

15- car\_type (required) -> represents delivery car type

16- car\_img1: (optional|mimes: jpeg, png) -> represents string of data of car image 1

17- car\_img2: (optional|mimes: jpeg, png) -> represents string of data of car image 2

20- car\_img3: (optional|mimes: jpeg, png) -> represents string of data of car image 3

21- id\_img: (optional|mimes: jpeg, png) -> represents string of data of delivery id image

22- authorization\_img: (optional|mimes: jpeg, png) -> represents string of data of authorization image

23- license\_img: (optional|mimes: jpeg, png) -> represents string of data of license image

24- insurance\_img: (optional|mimes: jpeg, png) -> represents string of data of delivery insurance image

25 - license\_date: represent date of exp in formate “2011-10-17”

26- id\_date: represent date of exp in formate “2011-10-17”

**If fails**

return {

'status' :false,

'errNum':errNum,

' msg' : msg

}

errNum =>

1 => 'All fields but password and profile pic are required',

2 => 'country and city must be numeric',

3 => 'E-mail must be in email format',

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

5 => 'Failed to upload images',

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

7 => 'Repeated email',

8 => 'Repeated phone',

9 => 'You can not update your name'

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

**If success**

return {

'status' :true,

'errNum':0,

'delivery': {

'delivery\_id' :delivery id,

'delivery\_name':delivery fullName,

'firstname':delivery firstname,

'secondname':delivery secondname,

'thirdname':delivery thirdname,

'lastname':delivery lastname,

'phone':delivery phone,

"license\_date": "2011-10-17",

"id\_date": "2011-10-17",

country\_code:delivery phone country code,

'email':delivery email,

'profile\_pic' :delivery image(url),

'longitude' :delivery longitude,

'latitude':delivery latitude,

car\_type:delivery car type,

'Device\_reg\_id':string represents account firebase reg id,

'country\_id': country id,

'delivery\_rate': delivery rate in decimal,

'city\_id': city id,

'city\_name': city name,

'created': account creation time in format (yyyy-mm-dd)

},

'msg' :'Signed up successfully'

}

# Delete attachment API:-

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

**params** => (file\_id, attach\_no, lang)

1. file\_id: (required|numeric) -> represent file number
2. attach\_no: (required|numeric)->represent attachments group number
3. lang: (default:en)->represent app displayed language

**If fails**

return {

'status' :false,

'errNum':errNum,

'msg' :msg

}

errNum =>

1 => 'file\_id is required',

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

3 => 'attach\_no is required',

4 => 'attach\_no must be a number',

5 => 'Failed to remove the file, please try again later'

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Removed successfully' ,

'attach\_no' :5 //represent the attached group number(will be 0 if there is no any files in that number)

}

# Get delivery orders API: -

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

**params** => ( delivery\_id, type, lang, allPages)

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

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

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

4- allPages: (default:0|in:0,1)->represents paginate flag if 1 then get all pages, if 0 then paginate

**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/getDeliveryOrders?page=2",

"prev\_page\_url": null,

"from": 1,

"to": 10,

"data": [

{

"user\_longitude": 13.3423,

"user\_latitude": 14.9834,

"order\_id": 1,

"provider\_name": "مطعم العنود",

"address": "test",

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

"delivery\_method": "ارامكس",

"meals\_count": 2,

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

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

},

{

"user\_longitude": 12.84783,

"user\_latitude": 11.34324,

"order\_id": 2,

"provider\_name": "مطعم العنود",

"address": "test",

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

"delivery\_method": "ارامكس",

"meals\_count": 2,

"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",

}

# Get order details API: -

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

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

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

"delivery\_price": 20,

"app\_value": 20,

"marketer\_delivery\_value": 20,

"marketer\_provider\_value": 20,

"user\_address": "test address",

"user\_longitude": 33.43342,

"user\_latitude": 33.33345,

"user\_phone": "1234567890",

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

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

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

"delivery\_method": "Aramex",

"provider\_name": "مطعم العنود",

"profile\_pic": "url" //represent provider profile pic,

"provider\_address": "عنوان تجريبى",

"profile\_longitude": 34.333,

"profile\_latitude": 34.658,

"user\_name": "user name" //To use in provider application

"order\_type": " مستقبلى",

"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

},

…….

],

"app\_percentage": 2 // to use in provider application,

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

}

# Order acceptance API: -

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

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

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

2- delivery\_id: (required|numeric)-> represents delivery 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 => 'delivery\_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'

}

# Fetch delivery credit data API: -

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

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

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

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

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'delivery\_id is required'

**If success**

return {

"status": true,

"errNum": 0,

"msg": "",

"balance": {

"current\_balance": 0,

"due\_balance": 0,

"forbidden\_balance": 0,

“Updated\_at”: “2018/01/04 17:12:17”

},

"bank\_name":””,

"bank\_phone":””,

"account\_num":””,

"bank\_username": “”

}

# Request to withdraw credit API: -

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

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

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

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

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

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

5 - name : (required) represent username

6 - phone : (required) represent user phone

7 - bank\_name : (required)

8- account\_num : (required)

9- forbidden\_balance : (required , numeric)

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'delivery\_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',

}

# Activate and deactivate receiving orders: -

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

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

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

2- delivery\_id: ( required) -> represents delivery 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',

}

# Deliver or cancel orders: -

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

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

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

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

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

4- type: (required|in:0,1) -> represents process type value ( 0 means cancel order, 1 order delivered)

**if fails**

return {

'status' :false,

'errNum' :errNum,

'msg' :msg

}

errNum =>

1 => 'All fields are required',

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

3 => 'Process failed',

**If success**

return {

'status' :true,

'errNum' :0,

'msg' : 'Order delivered successfully',

}