

1. Given the scenario of a user interacting on a mobile webapp to buy casino chips using phpt crypto of paygram company serving the payment on telegram bot called @opgmbot.. There will be a form on the frontend that the customer needs to enter its username, userrole (agent or player), & amount with a submit button. Since this is exclusive for team marc members only (agents/players) then to validate the username. An Api call is needed here to check its hierarchy and to check which top manager it belongs to. I have three accounts and those 3 are all having top manager role and each of them has separate token needed in api call to the casino platform so it is vital to identify which correct top manager the user belongs to, and this is to be able to let the logic function decide which account to be used and fetched its corresponding token. As I mentioned about the 3 accounts and tokens then here is the list of the information below:
 - Bossmarc@747 = 69aa9bb1-62a1-4c3a-b044-33aaec67a8fe
 - Teammarc = 994111a6-4edc-45a6-b0b3-629ef9fe7ea4
 - Marcthepogi = e5aeeb98-d795-4410-89e8-297ad2810852

Now lets assume that the user entered this detail on the form:

Username: wakay

Userrole: player (note that userrole is boolean type where agent is true and player is false)

Amount: 100

Next on the following list below are the API needed to use and the logic functions that we need to create and how to handle it.

1. (Tmpay747 is another third party api to handle member verification) GetUser Api: this api will provide a complete details of the username input (example: userrole, top manager, immediate manager, balance, statistics and etc). On our use case here for buying chips using crypto, the importance of this api is to determine its top manager and to validate that the user belongs to my team, otherwise even if the player bought the voucher and not validated as team marc member then the casino will reject the api call as not allowed.
2. (Paygram) issueInvoice API: this serves as the api to make the payment using telegram paybot, so when this api is called then the response will return a payurl which i am thinking to embed the payurl to a pay button that when clicked will open a browser then its going to redirect to telegram and will be directly interacting with the paybot named paygram (@opgmbot) and since the amount is already entered by the user on the form so the paybot will only show a menu to pay that requested amount and when the user clicked that button then the payment will initialize immediately and in successful payment then the last Api will trigger a call which is the actual deposit of chips on the input username account with the requested amount

and with the top manager details that we will also need to create a function to handle the lookup to retrieve the correct token extracted from getuser api that is the authentication key to make the last API call to make the deposit of chips if the result of the lookup on top manager did not contain any of the 3 usernames already mentioned then the operation should stop and gave a message that the username is not under team marc and is not allowed to request any transaction.

3. (Casino) Deposit API : this is where the long journey that should end happily ever after. So upon successful payment then this api will be triggered and will deposit the given amount. And a popup should show and a message with something like succesful payment and account deposited with that amount etc and there is a button that when clicked then will redirect to the players gaming portal of 747 live. “https://747live.net”
4. I also have one API that might be needed, its a message API of casino that will send a message to the actual casino account of that username that we will capture from the form because its needed on the parameter post body. This message will then be pushed to the message inbox of the user. So maybe this is great for creating OTP.

Now that I almost explained the flow so Next is the curl commands along with the responses of the API that will serve as example to let you understand more what I mentioned above. Here are the list and further explanations:

1. Getuser hierarchy Api call (on this curl we only need to fetch the username and to give an example, a player with a username: wakay. And here is the curl command and the actual response

```
curl -X GET 'https://tmpay747.azurewebsites.net/api/bridge/get-user/wakay/false'
-H 'accept:*//*'
```

```
Response: {
  "clientId": 329777805,
  "isAgent": false,
  "userType": "Player",
  "username": "Wakay",
  "topManager": "Marcthepogi",
  "immediateManager": "platalyn@gmail.com",
  "statistic": {
    "currentBalance": 4.56,
    "statisticsForThePast7Days": {
      "totalDeposit": 37810,
      "totalBet": 159649,
      "totalWithdraw": 5000,
```

```

    "canWithdraw": true,
    "amountToBet": 0,
    "wageringFactor": 4.222401481089658
  },
  "statisticsForMostRecentDeposit": {
    "totalDeposit": 5000,
    "totalBet": 10960,
    "totalWithdraw": 0,
    "canWithdraw": true,
    "amountToBet": 0,
    "wageringFactor": 2.192
  },
  "status": 0,
  "message": "OK"
},
"turnOver": null,
"managers": [
  "platalyn@gmail.com",
  "Wakay"
]
}

```

Here we found that wakay belongs to a username: marcthepogi as “top manager” which means it belongs to my team based on the response data("top manager : Marcthepogi"). I already mentioned and provided the token of each top manager which is needed for authenticating Casino Api call for deposit.

Please note that my next api call is for payment, on this second API we are not yet going to use the token because those tokens that we need to store will be used only for casino api and this next api is for the telegram paybot to make the payments. Here we only need to fetch the amount entered by the user on the frontend, here is the curl command for Paygram issueInvoice api: curl -X GET 'https://tgin.pay-gram.org/PayGramUsers/8YVgoWU7u4gmABVwJyetBL8vLaaz3e91FevPUNW4rhm7R50IYK0MsBGyrv78SW7B4yZqmFLbOJkbbVvXJN6F63ewzdZjsKX4mnoFQ5Sbqfw62tA4D8X7tnKg7yYoYVph8iAxStvCWDXNhWNwOlZLe6MHg7yOdEex5ASByWgXzTFQZptbcyvrkWl7oUgEhkvisxeXFInTxUtXeLAzwdJX0mrXHY6m44vc4lR3JjwPPx FQEXfAWaTIWDuymCGrcq6F/IssueInvoice?amt=10&cursym=11&callbackData=https://marc747.com/api/DepositCallbackRequest/PayGramDepCallBackReqAsync',

sample response

```

{
  "payUrl": "https://telegram.me/opgmbot?"
}

```

```

start=YT1pJmM9ZjAwZGMzZWItNDQ1MS00NzJlLTgyY2EtYmU1MjI5YWY2MWE4",
  "invoiceCode": "f00dc3eb-4451-472e-82ca-be5229af61a8",
  "status": 4,
  "lastEventUtc": "2023-11-27T06:27:37.7026656Z",
  "createdUtc": "2023-11-27T06:27:37.7026654Z",
  "amount": 100,
  "expectedFee": 0,
  "callbackData": "https://marc747.com/api/DepositCallbackRequest/PayGramDepCallBackReqAsync",
  "currencyCode": 11,
  "fromUser": null,
  "toUser": "1104423387",
  "originatingUser": null,
  "isPaid": false,
  "isRedeemed": false,
  "invoiceType": 0,
  "message": "ResponseOK",
  "type": 14,
  "responseCode": 200,
  "success": true
}

```

1. I suggest that the implementation to handle the redirection of the user to telegram to make the payment should create a separate function that will extract the payurl and to embed on a button click.
1. Then on this part the function to handle the realtime update of succesful payment should come then next is the casino deposit
1. Depositchips api (747bridge): `curl -X POST 'https://bridge.747lc.com/Default/Transfer' -H 'accept:text/plain' -H 'Content-Type:application/json' -d '{
 "authToken": "994111a6-4edc-45a6-b0b3-629ef9fe7ea4",
 "platform": 1,
 "amount": 100,
 "toAgent": false,
 "currency": "php",
 "nonce": "1278",
 "username": "wakay",
 "comment": "test for callback"
}'`

```
{  
  "status": 0,  
  "message": "ok"  
}
```

1. Sendmessage api (747bridge) maybe we can use this as security features in a form of otp where we can be sure that the one who do transact will receive a message on its account inbox in the casino platform. Suggest which area on kur flow to have this. `curl -X POST 'https://bridge.747lc.com/Default/`

```
SendMessage' -H 'accept:text/plain' -H 'Content-Type:application/json' -d '{  
  "authToken": "e5aeeb98-d795-4410-89e8-297ad2810852",  
  "platform": 1,  
  "username": "platalyn@gmail.com",  
  "subject": "password",  
  "message": "use this password: j5k4g4 "  
}'
```

```
Response: {  
  "status": 0,  
  "message": null  
}
```