ACCOUNTS REQUESTS

1. Login:

Path: /accounts/login
Method: POST
Request body:

```
{

        “username” : “string”,

        “password” : “string”

}
```

Response:

A. Invalid request method response 400

    {'error': 'Invalid request'}

B. missing field 402

    {'error': 'username and password are required’'}

C. If the credentials are invalid 401

    {'error': 'Invalid credentials'}

D. User dont exist 401

    {‘error’: ‘User does not exist’}

C. If the request succeeds. Response 200

    {“token” : “string auth-token”}

2. Logout:

URL: /accounts/logout

METHOD: DELETE

Request header: Authorization   “string auth-token”

Response:

A. Invalid request method 400

{'error': 'Invalid request'}

B. If there is no token 402

{'error': 'missing field'}

C. If the authentication token does not exist 401

{'error': 'Invalid credentials'}

D. If the request succeeds 200

{'message': 'You have been logged out'}

3. Signup:

URL:/accounts/signup

METHOD: POST

Request body:

```
{
        “username” : “string”,
        “password” : “string”
        “email” : “string”
        “phone” : “string”
        “name” : “string”
        “address” : “string”
        "hasCar": boolean,
        If hascar true
        “licence” : “string”
        “carmodel” : “string”
        “color” : “string
}
```

Responses:

A. Invalid request method 400

{'error': 'Invalid request'}

B. missing field 402

{'error': 'username, password, email, name, phone and address are required'}

C. phone exist 401

{'error': 'username already exists'}

D. username exist 401

{'error': 'phone already exists'}

E. email exist 401

{'error': 'email already exists'}

F. car info missing 402

{'error': 'licence, carmodel and color are required'}

G. email not valid email 401

{'error': "Invalid email"}

H. success 200

{'token': token}

4. Profiles

URL: /accounts/profile

METHOD: GET

Request header: Authorization “string auth-token”

Responses:

A. Invalid request method 400

```
{'error': 'Invalid request'}
```

B. invalid token response 401

```
{'error': 'Invalid token'}
```

C. success 200

```
{
'username': “string”
'password': “string”,
'email': “string”,
'name': “string”,
'phone':  “string”,
'address': “string”,
'wallet': “float”,
'carModel': “string” else None,
'carLicense': “string” else None,
'carColor': “string” else None,
}
```

METHOD: POST

Request header: Authorization “string auth-token”

Request body:

{

"name": "string",

"email": "string",

"address": "string",

"phone": "string",

"password": "string",

"hasCar": “boolean”, **(logic checks for “True” to add a car if user did not have car)**

"licence": "string" (required if hasCar is true),

"carModel": "string" (required if hasCar is true),

"carColor": "string" (required if hasCar is true)

}

Responses:

A. Invalid request method 400

{'error': 'Invalid request'}

B. email exist 401

{'error': 'email already exists'}

C. phone exist 401

{'error': 'phone already exists'}

D. car info missing 402

{'error': 'licence, carmodel and color are required'}

E. general exception 401

{'error': Exception}

F. success 200

{"response": "good"}

5. Check wallet amt

URL: /accounts/wallet

METHOD: GET

Request header: Authorization   “string auth-token”

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C return wallet amt code 200

{'wallet': “float with money amount”}

6. withdraw/topup from wallet

URL: /accounts/wallet/action

METHOD: POST

Request header: Authorization   “string auth-token”

Request body:

{

“amount” : “float”

“action”:”string”   **withdraw or topup as action word**

}

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C missing field 402

{'error': 'missing field'}

D insufficient funds 403

{'error': 'insufficient funds'}

E. invalid action 401

{'error': 'invalid action'}

F. success topup 200

{'success': 'money added'}

G. success withdraw 200

{'success': 'money withdrawn'}

7. Check transaction history
URL: /accounts/wallet/history

METHOD: GET

Request header: Authorization   “string auth-token”

Responses:

A. invalid request method response 400

```
{'error': 'Invalid request'}
```

B. invalid token response 401

```
{'error': 'Invalid token'}
```

C no history 200

```
{'success': 'No history found'}
```

D success 200. **Example below of 2 transaction history, latest time on top**

```
[
        {
        "action": "deduction",
        "amount": 10000.0,
        "date": "2024-03-29T15:46:54.514Z"
        },
        {
        "action": "deduction",
        "amount": 10000.0,
        "date": "2024-03-29T15:20:41.534Z"
        },
]
```

8. Service history
URL: /accounts/servicehistory

METHOD: GET

Request header: Authorization   “string auth-token”

Request body:

{

“type” : “string”   **requester** or **responder**

}   **pulls history of user whether a driver or rider/delivery**

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C. missing field 402

{'error': 'missing field'}

D. invalid type 401 **not responder or requester**

{'error': 'invalid type'}

E no history found for selected type 403

{'error': 'No history found'}

G success 200 **example below of 2 history, latest time on top**

[

{

"requester": "mrrequest",

"responder": "mrdriver",

"requester_rating": 4.0,

```
"responder_rating": 5.0,
"feedback_to_driver": "excellent",
"feedback_to_requester": "excellent",
"transaction_amount": 10000.0,
"service_type": "ride",
"time_started": "2024-03-29T15:45:46.129Z",
"time_accepted": "2024-03-29T15:46:35.914Z",
"time_pickup": "2024-03-29T15:46:48.941Z",
"time_completed": "2024-03-29T15:46:53.335Z",
"pickup_latitude": 1.315003487061868,
"pickup_longitude": 103.76504067446228,
"pickup_place_name": "clementimrt",
"end_latitude": 1.2838969385649643,
"end_longitude": 103.85911722552535,
"end_place_name": "mbs",
"distance": 12.345,
"duration": 35.72
},
{
"requester": "mrrequest",
"responder": "mrdriver",
"requester_rating": 4.0,
"responder_rating": 4.0,
"feedback_to_driver": "excellent",
"feedback_to_requester": "excellent",
"transaction_amount": 10000.0,
"service_type": "ride",
"time_started": "2024-03-29T15:19:25.795Z",
"time_accepted": "2024-03-29T15:20:23.115Z",
"time_pickup": "2024-03-29T15:20:35.096Z",
"time_completed": "2024-03-29T15:20:40.385Z",
"pickup_latitude": 1.315003487061868,
"pickup_longitude": 103.76504067446228,
"pickup_place_name": "clementimrt",
"end_latitude": 1.2838969385649643,
"end_longitude": 103.85911722552535,
"end_place_name": "mbs",
"distance": 12.345,
"duration": 35.72
}
]
```

SERVICES REQUESTS

9. Requesting a service

URL: /services/request

METHOD: POST

Request header: Authorization “string auth-token”

Request body: {

service_type = “string” **delivery** or **ride**

pickup_latitude = “float”

pickup_longitude = “float”

pickup_place_name = “string”

end_latitude = “float”

end_longitude = “float”

end_place_name = “string”

distance =”float”

duration=”float”

requester_text=”string”

transaction_amount =”float”

}

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C. missing field 402

{'error': 'missing field'}

D 403

{'error': 'User already has a current request'}

E 403

{'error': 'User has active ride'}

F success 200

{"response": "good"}

10. Requester pings this for updates after making request

\URL: /services/requesterupdates

METHOD: GET

Request header: Authorization “string auth-token”

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C. user no current request or active ride 403

{'Error': 'No request found for user'}

D request is success, no drivers accept the request yet 200

{'success': 'Awaiting driver'}

E driver accepted, on the way to pickup 200

{'Accepted': 'Awaiting pickup'}

F driver has picked up, on route to destination 200

{'Accepted': 'On Route'}

G delivery/passenger reach destination . dropped off, will delete trip from activeride. 200

{"ServiceHistoryID":ridehistory.id}

the id for user frontend to pass into rating api

11. Driver wants to make money, key in his location to find 3 closest current request. Closest by google maps driving time from his current location to pickup point

URL: /services/drive

METHOD: POST

Request header: Authorization “string auth-token”

Request body:

```
{
        "location_latitude": "1.315003487061868",          float
        "location_longitude": "103.76504067446228"         float
}
```

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C driver currently have active request or active ride 403

{'error': 'User already has an active ride or current request'}

D no location field in request body 402

{'error': 'location field missing'}

E user has no car 403

{'error': 'User has no car'}

F success 200 **return top 3 nearest or, if 2 or 1 if less than 3**

```
[
        [
        34,          the ID
        {
        "distanceKm": 0.0,
        "durationMinutes": 0.03,
        "transactionAmount": 10000.0,
        "serviceType": "ride",
        "requesterText": "hi",
        "pickupLatitude": 1.315003487061868,
        "pickupLongitude": 103.76504067446228,
        "pickupPlaceName": "clementimrt",
        "endLatitude": 1.2838969385649643,
        "endLongitude": 103.85911722552535,
        "endPlaceName": "mbs",
        "rideDistance": 12.345,
        "rideDuration": 35.72
        }
        ]
        [
        Other request if request list more than 3, will limit to nearest 3
        ]
]
```

12. User has a current request that no driver accepted yet, want to cancel request

URL: /services/cancel

METHOD: DELETE

Request header: Authorization “string auth-token”

Responses:

A. invalid request method response 400

```
{'error': 'Invalid request'}
```

B. invalid token response 401

```
{'error': 'Invalid token'}
```

C no current request 403

```
{'error': 'No current requests'}
```

D success 200

```
{"response": "good"}
```

13. Driver wants to accept a request
URL: /services/accept

METHOD: POST

Request header: Authorization “string auth-token”

Request body:

```
{

        "current_request_id":"13"

}
```

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C no id in body 402

{'error': 'missing field'}

D error cannot accept ride when user has a request or is in a active ride 403

{'error': 'User already has an active ride or current request'}

E ride id dont exists 401

{'error': 'Invalid current_request_id'}

F cannot accept own’s request 403

{'error': 'Cannot accept own request'}

G success, get back requester name(not username) normal name, so can verify hey are you matthew when driver is picking up. 200

{"user's name": requester_name}

14. Driver at pickup, press pickup to update

URL: /services/pickup

METHOD: POST

Request header: Authorization “string auth-token”

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C already picked up 403

{'error': 'Already picked up'}

D user do not have a active ride 403

{'error': 'No current requests'}

F success 200

{"response": "good"}

15. Driver reach destination, dropoff, update wallets, deduct one side, add other side.

URL: /services/dropoff

METHOD: POST

Request header: Authorization “string auth-token”

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C Not picked up yet 403

{'error': 'Not picked up yet'}

D Error updating wallets 500

{'error': 'Error updating wallets'}

F already dropped off 403

{'error': 'Already dropped off'}

G no current active ride 403

{'error': 'No current requests'}

H success 200 **will create entry in servicehistory, will not delete activeride, till requesterupdates gets updated on servicehistoryid , requesterupdate endpoint will delete active ride**

{"ServiceHistoryID":saved.id} **the id for user frontend to pass into rating api**

16. Rating

URL: /services/rating

METHOD: POST

Request header: Authorization “string auth-token”

Request body:

{

“rating” :”float” **users rate in stars, 0|0.5|1|1.5|2|2.5|3|3.5|4|4.5|5**

“rating_text” : “string”

“ride_id”: “servicehistory id number”

}

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C missing body 402

{'error': 'missing field'}

D ride id invalid 401

{'error': 'Invalid ride_id'}

F user not requester or responder in this ride 401

{'error': 'Invalid user for this ride'}

G user gave rating and feedback already .403

{'error': 'Rating already recorded'}

H success 200

{'success': 'Rating recorded'}

17. Let frontend ping at anytime to retrieve full current active ride info

URL: /services/current_ride_info

METHOD: GET

Request header: Authorization “string auth-token”

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C no current active ride 403

{'error': 'No current rides'}

D success 200 **show current ride info**

```
{
"requester_text": "hi",
"transaction_amount": 10000.0,
"service_type": "ride",
"pickup_latitude": 1.315003487061868,
"pickup_longitude": 103.76504067446228,
"pickup_place_name": "clementimrt",
"end_latitude": 1.2838969385649643,
"end_longitude": 103.85911722552535,
"end_place_name": "mbs",
"distance": 12.345,
"duration": 35.72,
        "responder_name": "mrdriver",

        "requster_name": "mrrequest"

}
```

18. Let frontend ping at anytime to retrieve if user has a current request or active ride or none

URL: /services/anyrides

METHOD: GET

Request header: Authorization “string auth-token”

Responses:

A. invalid request method response 400

{'error': 'Invalid request'}

B. invalid token response 401

{'error': 'Invalid token'}

C. no request or active ride

{'error': 'No rides found'}

D. got request 200

{'success': 'User has current request'}

E active ride as requester 200

{'success': 'User has active ride as requester'}

F active ride as responder 200

{'success': 'User has active ride as responder'}