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': “**<string>** are required”} string=**username**/**password**

C. If the credentials are invalid 401

    {'error': 'Invalid credentials'}

D. User don’t exist 401

    {‘error’: ‘User does not exist’}

E. If the request succeeds. Response 200

    {“token” : string auth-token}

2. Logout, deletes auth token and signs out

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, new user sign up

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': “**<string>** are required’”} string=**username/ password/ email/ name/phone/address/hasCar**

C. username exist 401

{'error': 'username already exists'}

D. phone 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, get method for retrieving profile details, post method for editing

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",

"hasCar": “boolean”, **(logic checks for “True” to add a car if user did not have car or edit car details if user already has a 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. success 200

{"response": "good"}

5. Check wallet amount

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”}

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': “<**string>** are required”} string=**amount/action**

D. insufficient funds 403

{'error': 'insufficient funds'}

E. have current request or ride as requester, cannot withdraw 403

{'error': 'cannot withdraw while on a ride'}

F. invalid action 401

{'error': 'invalid action'}

G. success topup 200

{'success': 'money added'}

H. 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

{[ ]} **empty**

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. Get service history, for ride,delivery and as a driver
URL: /accounts/servicehistory

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 found found 200

{[ ]} **empty**

D. 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

}

]
```

<u>SERVICES REQUESTS</u>

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': “<**string>** are required’'} string=**any of the key in request body**

D. 403

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

E. 403

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

F. 403

{'error': 'Insufficient funds'}

G. success 200

{"response": "good"}

10. Requester keeps pinging 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 200

```
{'status': 'none'}
```

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

```
{'status': 'Awaiting driver'}
```

E. driver accepted, on the way to pickup 200

```
{'status': 'Awaiting pickup',
 'driver_latitude': ”float”,
 'driver_longitude': ”float”}
```

F. driver has picked up, on route to destination 200

```
{'status': 'On Route'
  'driver_latitude': ”float”,
  'driver_longitude': ”float”}
```

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

```
{"ServiceHistoryID": “int‘}
```

the id for 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': “<**string>** are required”} string=**any of the key in request body**

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 truncate 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" **int**

“driver_latitude”:”103.3123131”, **float**

“driver_longitude”:”1.231222” **float**

}

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': “<**string>** are required’’} string=**any of the key in request body**

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 the requester’s name, so can verify when 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'}

E. success 200

{"response": "good"}

15. Driver reaches destination, click drop off, update wallets, deduct requester $$, add driver $$.

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, database error 500

{'error': 'Error updating wallets'}

E. already dropped off 403

{'error': 'Already dropped off'}

F. no current active ride 403

{'error': 'No current requests'}

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

{"ServiceHistoryID":”int”} **the id integer for frontend to pass into rating api**

16. Rating, call when end trip to update ratings

URL: /services/rating

METHOD: POST

Request header: Authorization “string auth-token”

Request body:

{

“rating” :”float” **users rate in int, 0|1|2|3|4|5**

“rating_text” : “string”

“ride_id”: “int” **service history 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': “<**string>** are required”} string=**any of the key in request body**

D. ride id invalid 401

{'error': 'Invalid ride_id'}

E. user not requester or responder in this ride 401

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

F. user gave rating and feedback already .403

{'error': 'Rating already recorded'}

G. success 200

{'success': 'Rating recorded'}

17. Let frontend ping at anytime to retrieve full current active ride info and let driver update his location

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",

"driver_latitude": 1.315003487061868,

"pickup_longitude": 103.76504067446228,

"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"

}
```

METHOD: POST

Request header: Authorization “string auth-token”

Request body:

{

“driver_latitude”:”1.231323”, **float**

“driver_longitutde”:”123.3123” **float**

}

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. missing field 402

{'error': “**<string>** are required”} string=**any of the key in request body**

E. success 200 **show current ride info**

{'success': 'Location updated'}

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 200

{'status': 'none'}

D. got request 200

{'status': 'request'}

E. active ride as requester 200

{'status': 'active_requester'}

F. active ride as responder 200

{'status': 'active_responder'}

19. Get rating

URL: /services/get_rating

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. database error 500

{'error': 'Database error'}

D. success 200

```
{
'ratingAsRequester': “float”,
'ratingAsResponder': “float”
}
```

20. Get current request details

URL: /services/current_request_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 request 403

{'error': 'No current requests'}

D. success 200

{

'requester_text': ”string”,

'transaction_amount': “float”,

'service_type': ”string”,

'pickup_latitude': “float”,

'pickup_longitude': “float”,

'pickup_place_name': ”string”,

'end_latitude': “float”,

'end_longitude': “float”,

'end_place_name': ”string”,

'distance': “float”,

'duration': “float”

}

21. Get current request details and driver location coordinates

URL: /services/responderstatus

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 request 200

```
{'status': 'none'}
```

D. success awaiting pickup 200

```
{

'status': 'Awaiting pickup',

 'driver_latitude':”float”,

 'driver_longitude':”float”

}
```

E. success on Route 200

```
{

'status': 'Awaiting pickup',

 'driver_latitude':”float”,

 'driver_longitude':”float”

}
```

22. Get the service history id searching by the time started

URL: /services/getServiceIdByTimeStarted

METHOD: POST

Request body:

```
{
        'time_started':”string”
}
```

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 service history id 200

{'ServiceHistoryID': “int”}

E. error no such service history found 403

{'error': 'No service history with that time started'}