Setup and Installation

Prerequisites:-

- Python 3
- Django Framework
- pip (Python package installer)

1. Clone the repository:

Clone the repository from GitHub:

```
git clone https://github.com/sajidsajad/expenses.git
cd expenses/expenses_app
```

2. Configure Django Settings

Update expenses_app/settings.py with your specific configurations such as database settings, allowed hosts, and any other necessary settings.

On my application, I have used sqlite3

3. Apply Migrations

Apply the Django migrations to set up your database schema:

```
python3 manage.py makemigrations

python3 manage.py migrate
```

4. Run the Development Server

Start the Django development server:

```
python manage.py runserver
```

5. Access the Application

Open your web browser and navigate to http://127.0.0.1:8000/ to start using the application.

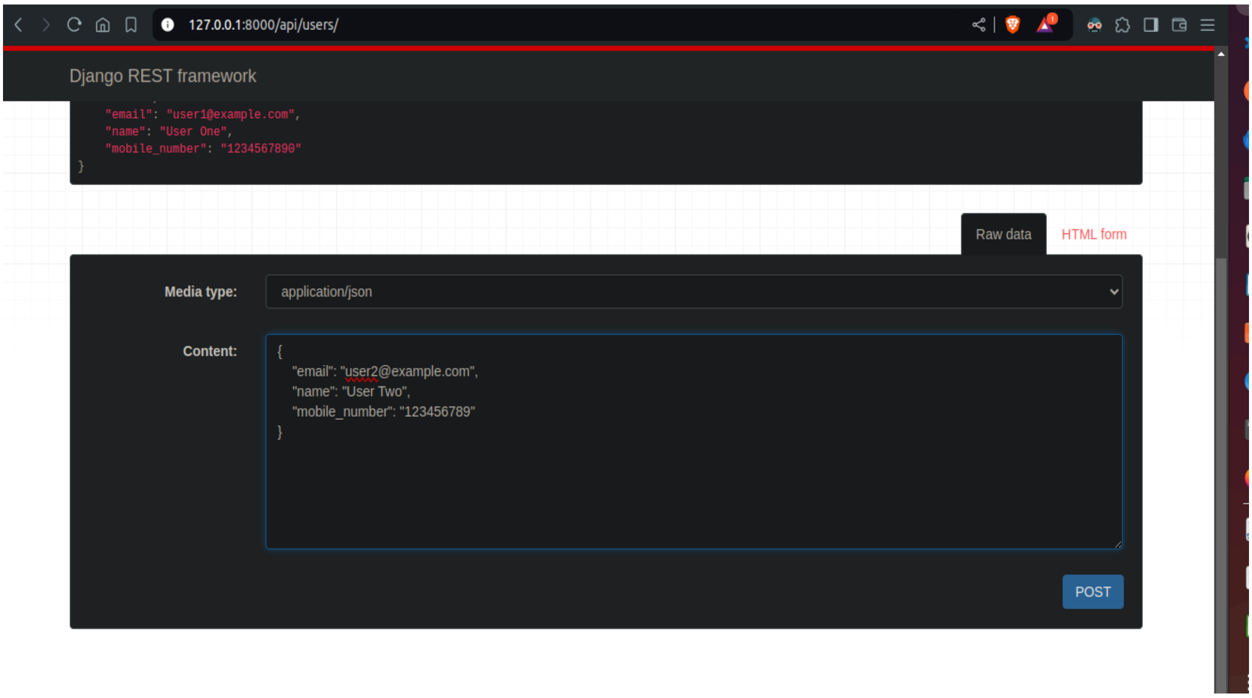
API Endpoints:-

User Endpoints:-

#Create User

Endpoint: POST /api/users/

#Request Body:



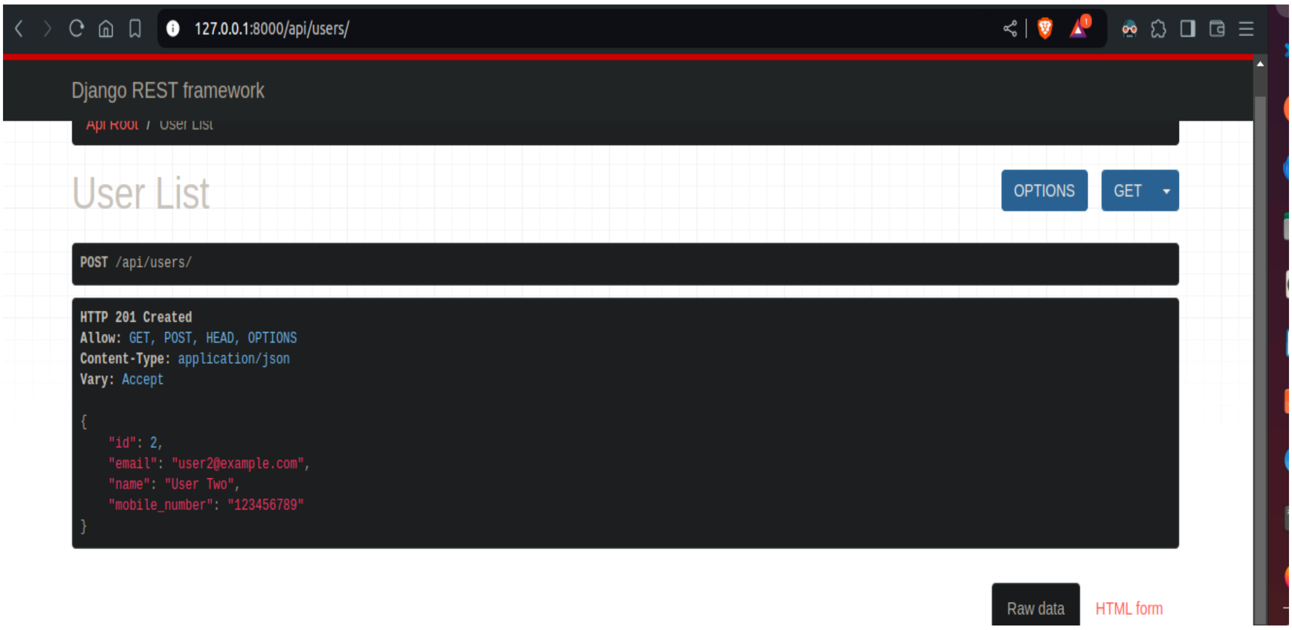
```
    "email": "user1@example.com",
    "name": "User One",
    "mobile_number": "1234567890"
}
```

Media type: application/json

Content:

```
{
    "email": "user2@example.com",
    "name": "User Two",
    "mobile_number": "123456789"
}
```

#Response Body:



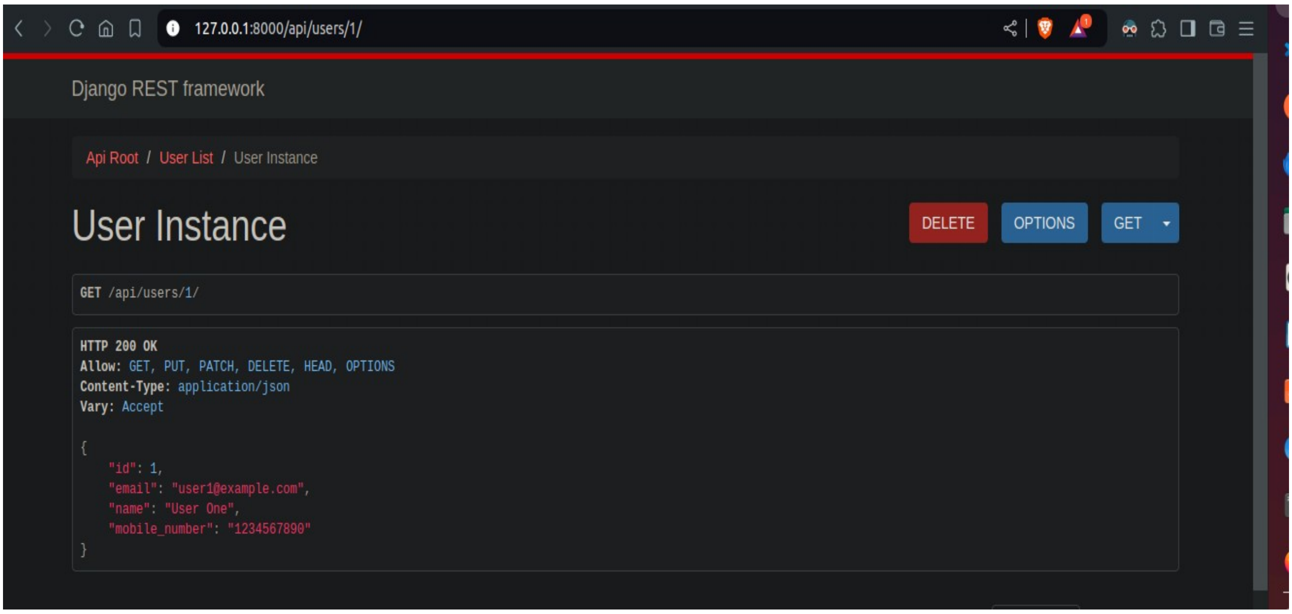
User List

```
POST /api/users/
```

```
HTTP 201 Created
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "id": 2,
    "email": "user2@example.com",
    "name": "User Two",
    "mobile_number": "123456789"
}
```

#Retrieve User Details

Endpoint: GET /api/users/{id}/

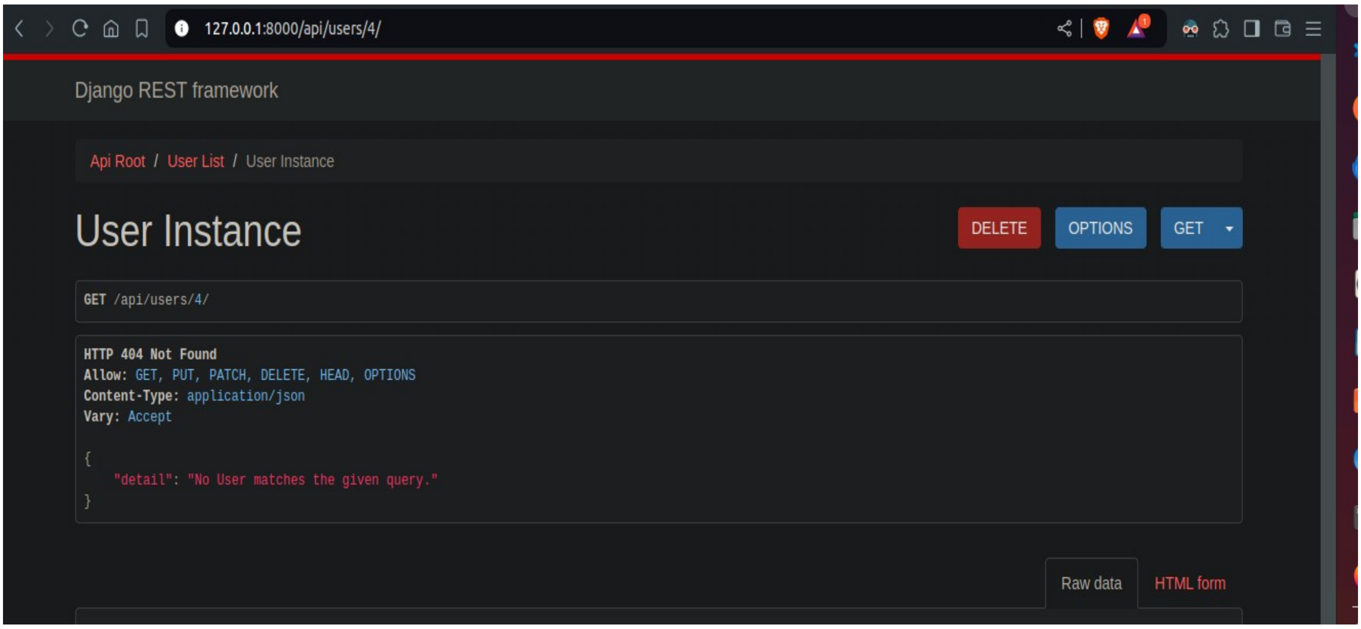


127.0.0.1:8000/api/users/1/
Django REST framework
Api Root / User List / User Instance
User Instance
DELETE
OPTIONS
GET

```
GET /api/users/1/
```

```
HTTP 200 OK
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "id": 1,
    "email": "user1@example.com",
    "name": "User One",
    "mobile_number": "1234567890"
}
```

#if no user is present:



```
GET /api/users/4/
```

```
HTTP 404 Not Found
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "detail": "No User matches the given query."
}
```

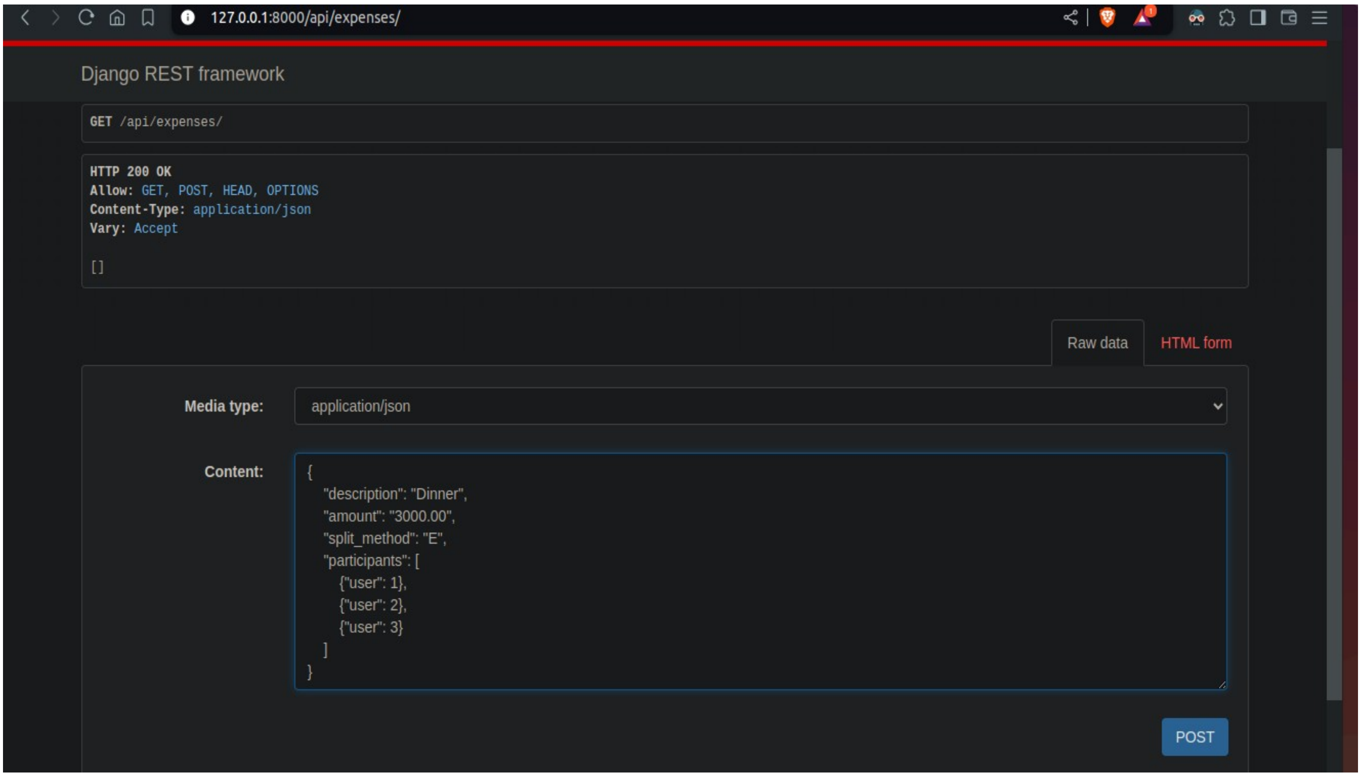
#Expense Endpoints:-

01. Add Expenses:

Endpoint: POST /api/expenses/

Scenario 01: EQUAL (E)

Request:



```
GET /api/expenses/
```

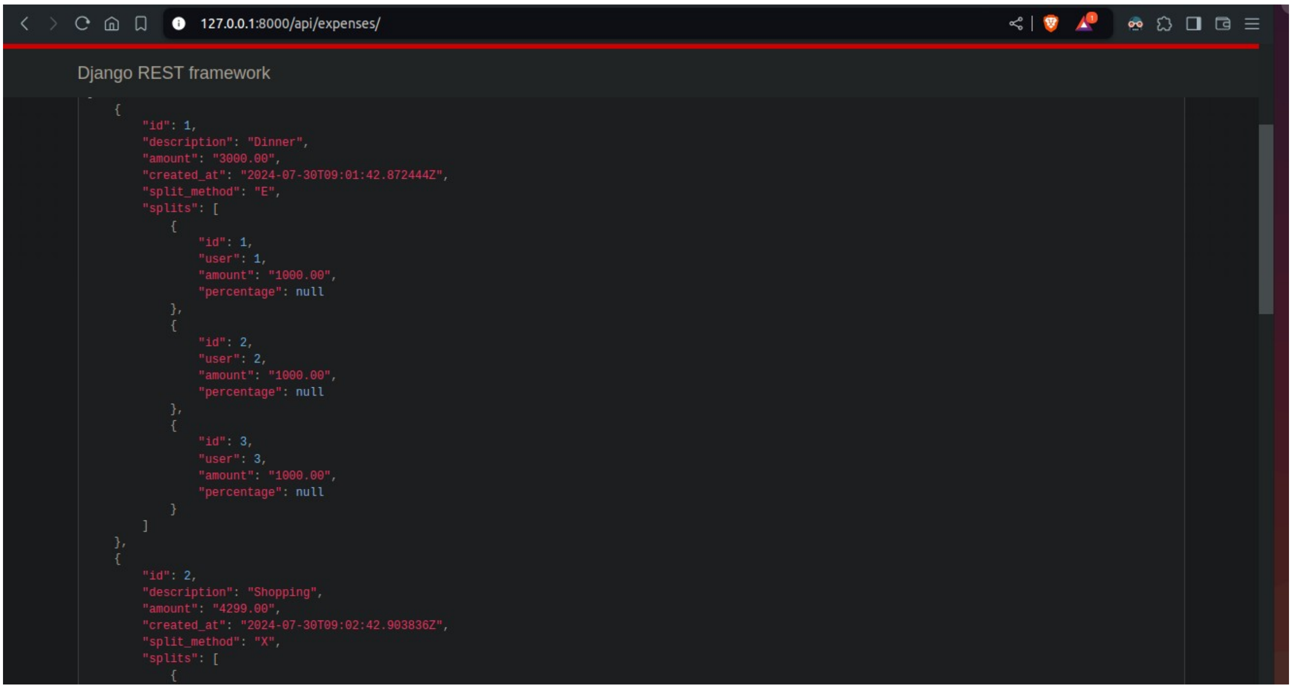
```
HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

[]
```

Media type: application/json

Content:

```
{
    "description": "Dinner",
    "amount": "3000.00",
    "split_method": "E",
    "participants": [
        {"user": 1},
        {"user": 2},
        {"user": 3}
    ]
}
```

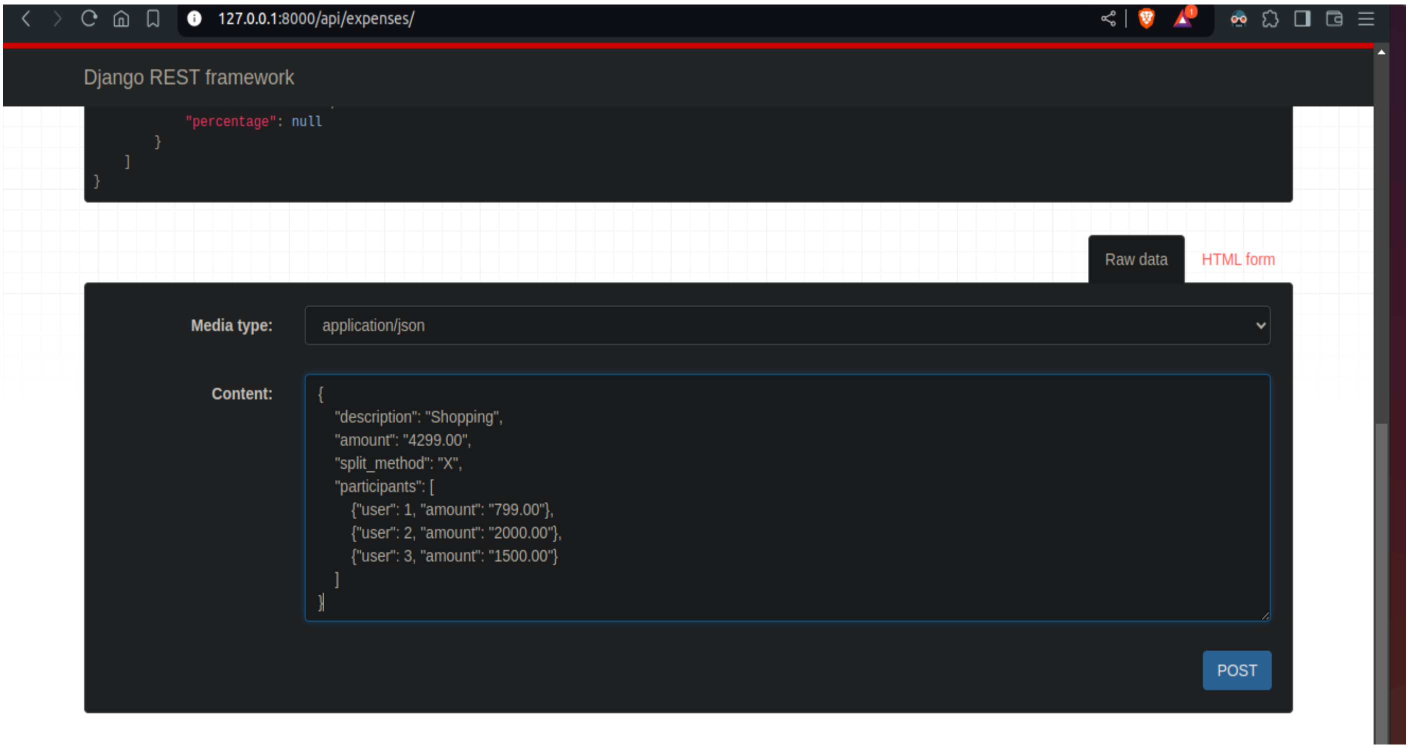
Response:



```
    {
        "id": 1,
        "description": "Dinner",
        "amount": "3000.00",
        "created_at": "2024-07-30T09:01:42.872444Z",
        "split_method": "E",
        "splits": [
            {
                "id": 1,
                "user": 1,
                "amount": "1000.00",
                "percentage": null
            },
            {
                "id": 2,
                "user": 2,
                "amount": "1000.00",
                "percentage": null
            },
            {
                "id": 3,
                "user": 3,
                "amount": "1000.00",
                "percentage": null
            }
        ]
    },
    {
        "id": 2,
        "description": "Shopping",
        "amount": "4299.00",
        "created_at": "2024-07-30T09:02:42.903836Z",
        "split_method": "X",
        "splits": [
            {
```

Scenario 02: EXACT (X)

Request:



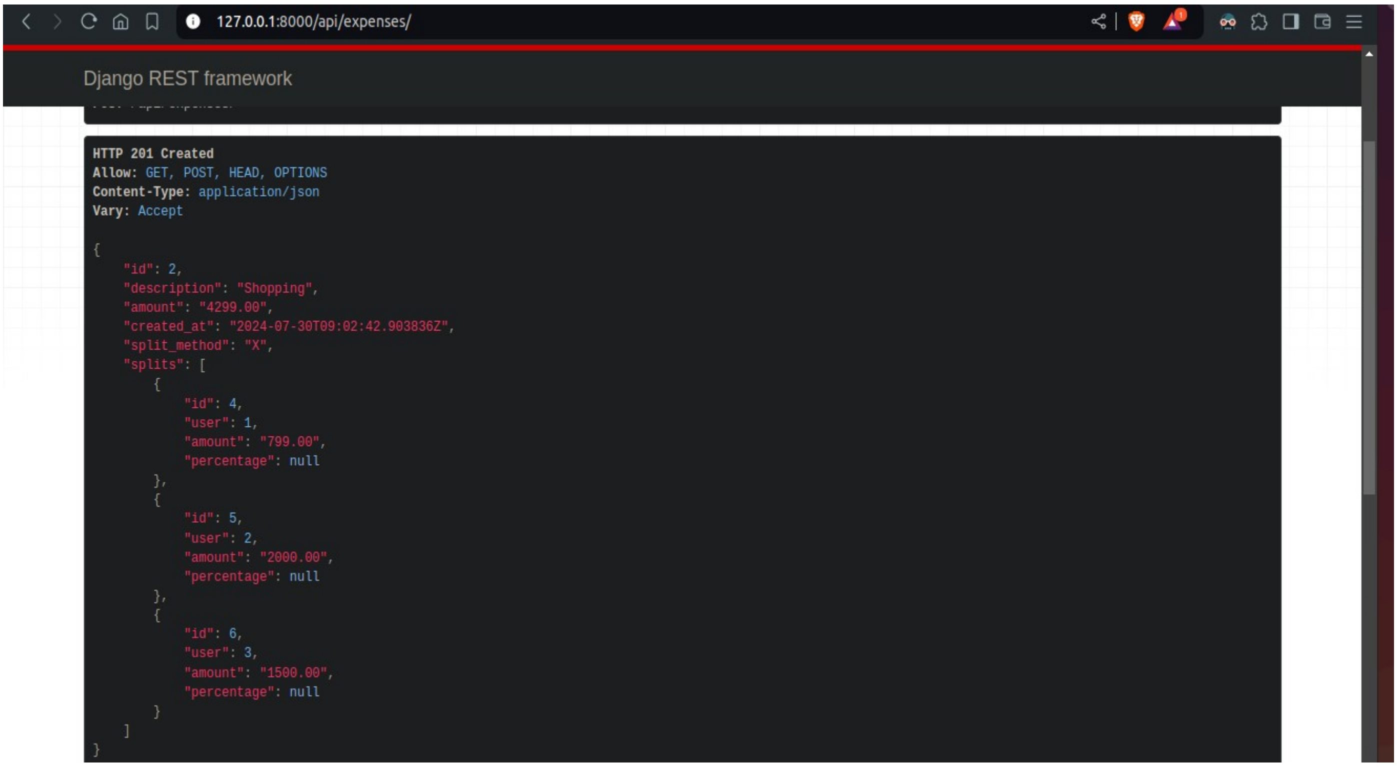
```
            "percentage": null
        }
    ]
}
```

Media type: application/json

Content:

```
{
    "description": "Shopping",
    "amount": "4299.00",
    "split_method": "X",
    "participants": [
        {"user": 1, "amount": "799.00"},
        {"user": 2, "amount": "2000.00"},
        {"user": 3, "amount": "1500.00"}
    ]
}
```

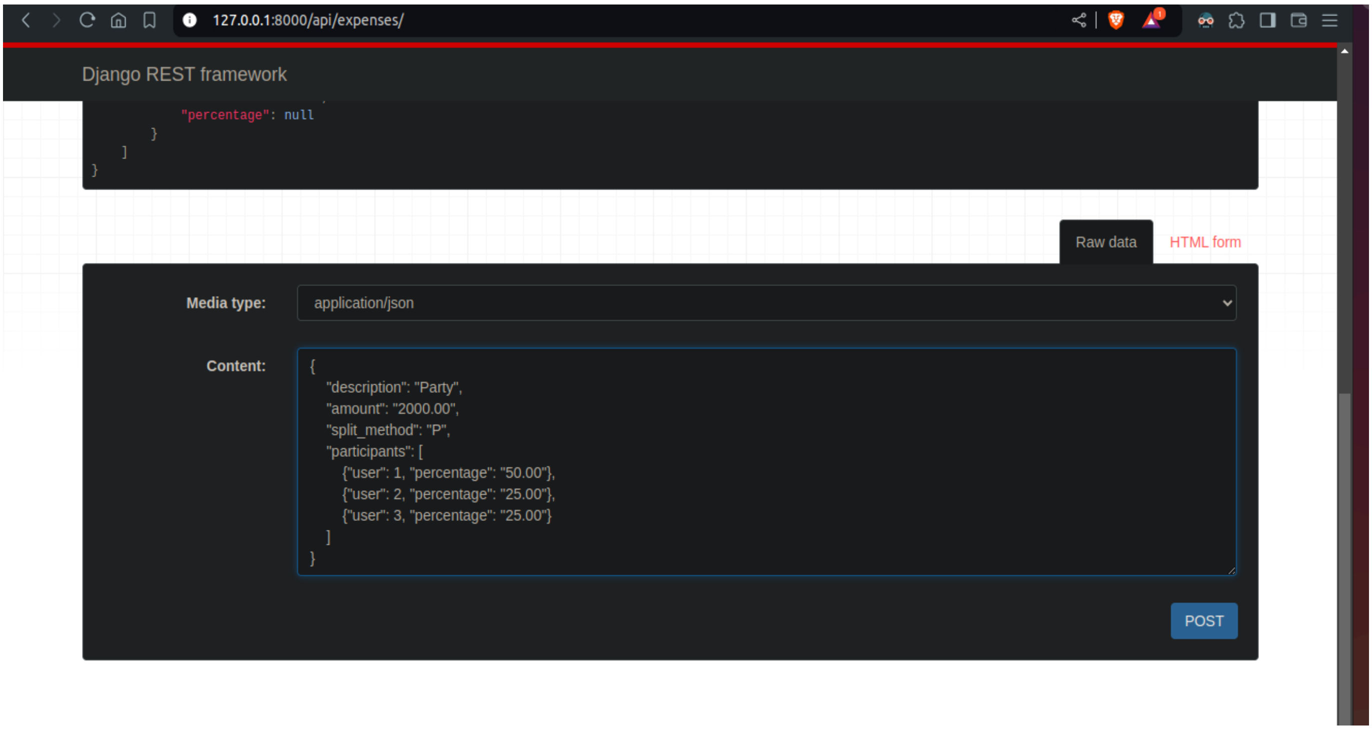
Response:



```
HTTP 201 Created
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "id": 2,
    "description": "Shopping",
    "amount": "4299.00",
    "created_at": "2024-07-30T09:02:42.903836Z",
    "split_method": "X",
    "splits": [
        {
            "id": 4,
            "user": 1,
            "amount": "799.00",
            "percentage": null
        },
        {
            "id": 5,
            "user": 2,
            "amount": "2000.00",
            "percentage": null
        },
        {
            "id": 6,
            "user": 3,
            "amount": "1500.00",
            "percentage": null
        }
    ]
}
```

Scenario 03: PERCENTAGE (P)

Request:



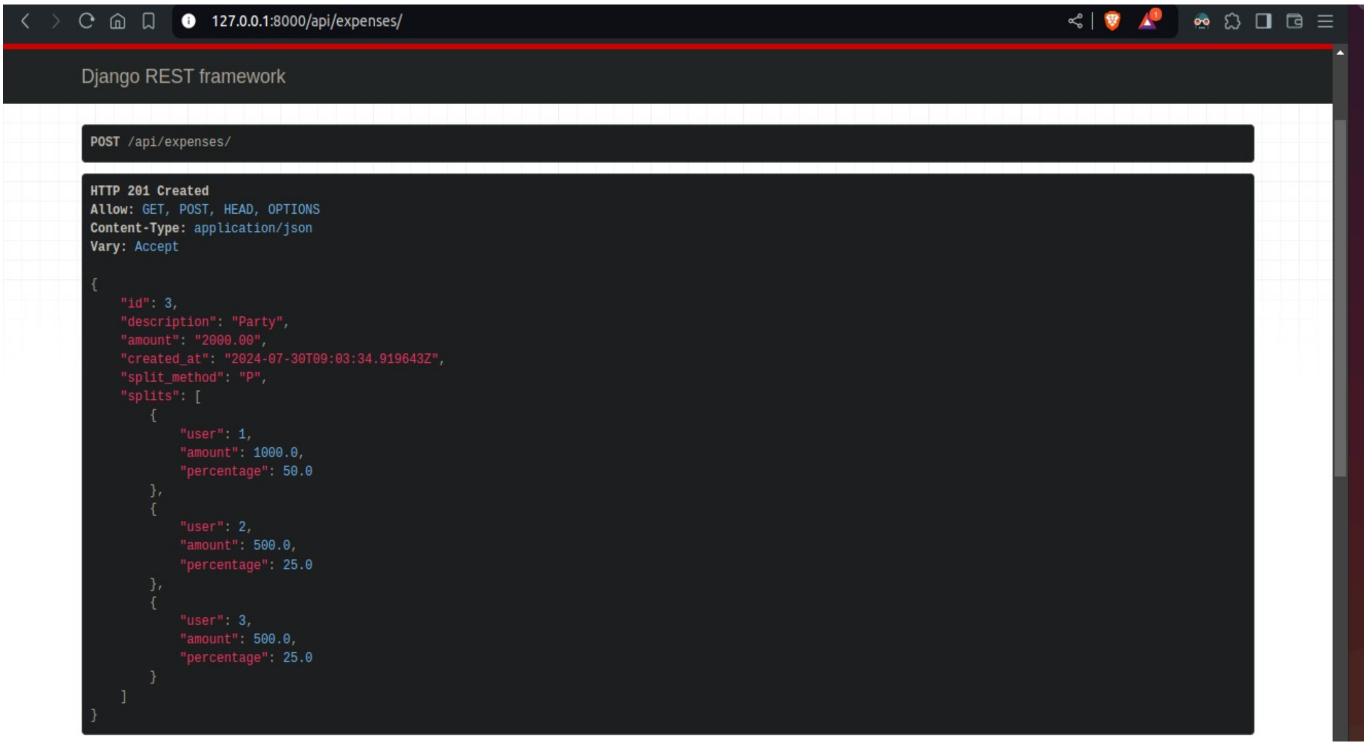
```
            "percentage": null
        }
    ]
}
```

Media type: application/json

Content:

```
{
    "description": "Party",
    "amount": "2000.00",
    "split_method": "P",
    "participants": [
        {"user": 1, "percentage": "50.00"},
        {"user": 2, "percentage": "25.00"},
        {"user": 3, "percentage": "25.00"}
    ]
}
```

Response:



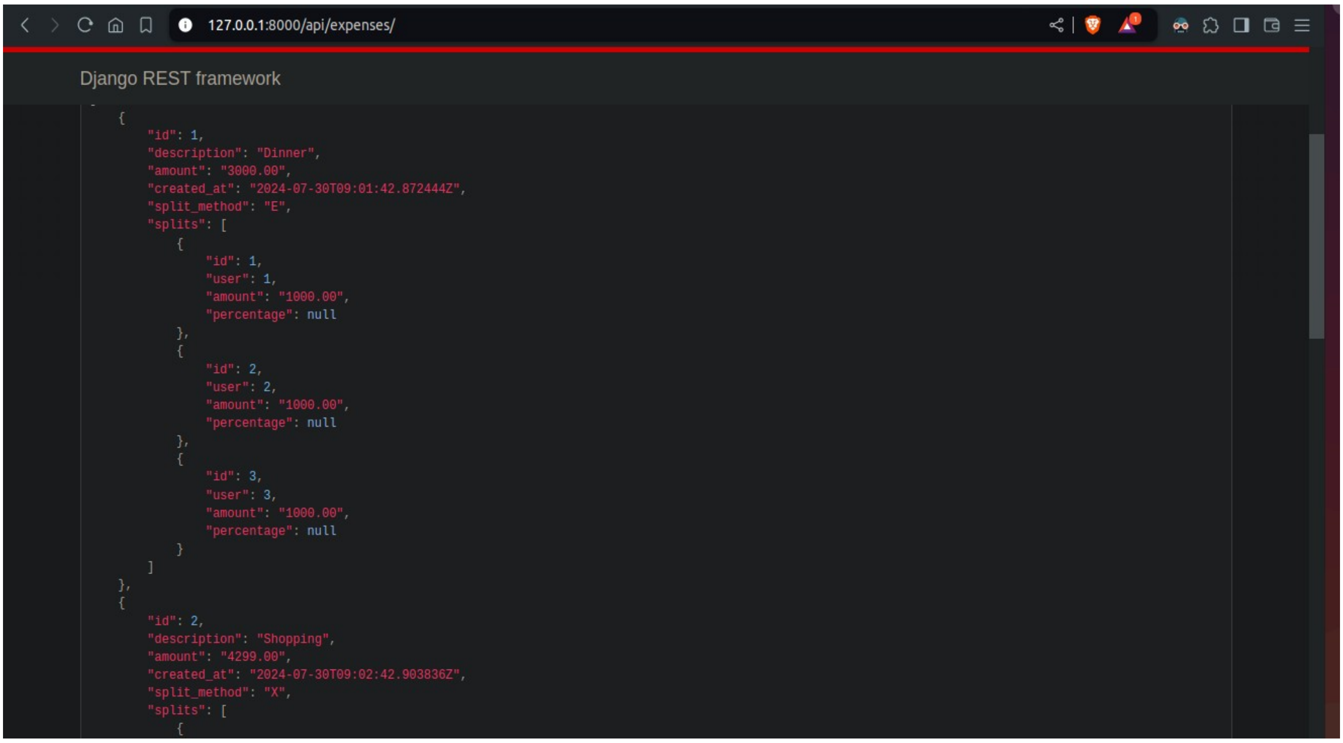
```
POST /api/expenses/
```

```
HTTP 201 Created
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "id": 3,
    "description": "Party",
    "amount": "2000.00",
    "created_at": "2024-07-30T09:03:34.919643Z",
    "split_method": "P",
    "splits": [
        {
            "user": 1,
            "amount": 1000.0,
            "percentage": 50.0
        },
        {
            "user": 2,
            "amount": 500.0,
            "percentage": 25.0
        },
        {
            "user": 3,
            "amount": 500.0,
            "percentage": 25.0
        }
    ]
}
```

02. Get Expenses:

Endpoint: GET /api/expenses/ (will give all expenses)

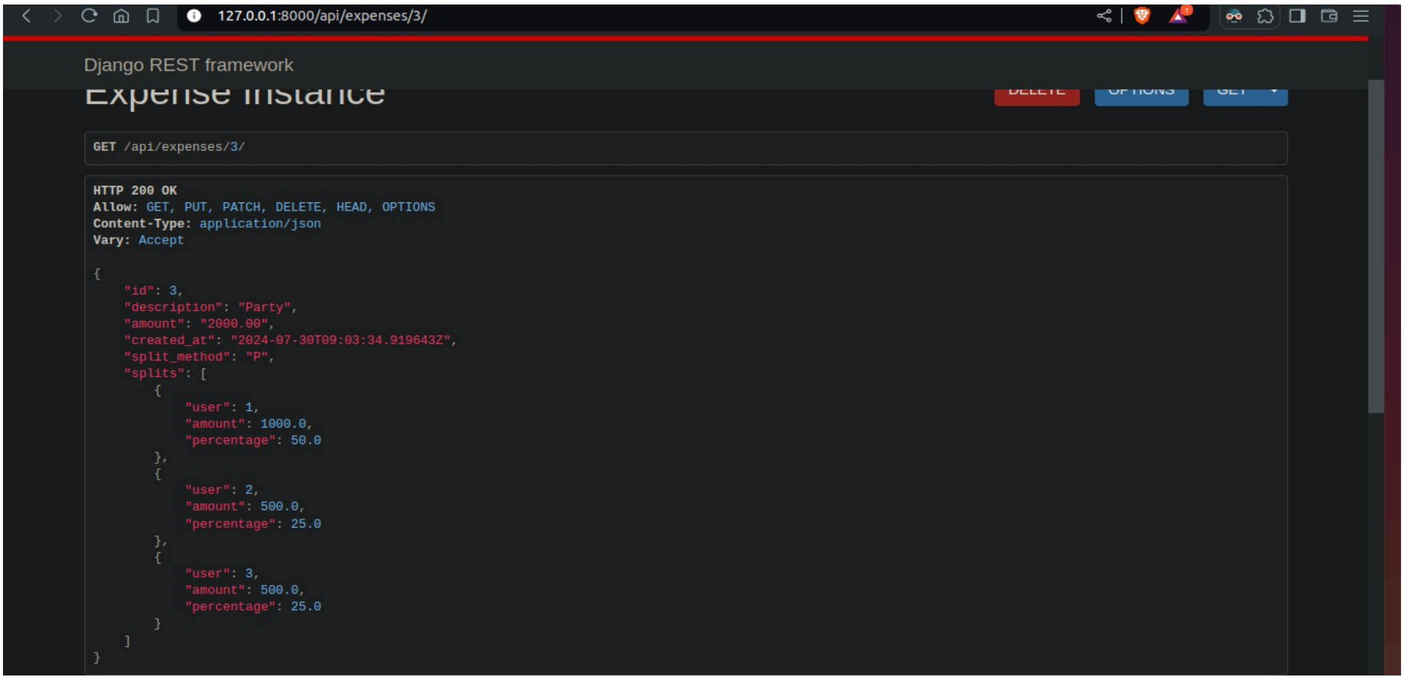
Response:



```
    {
        "id": 1,
        "description": "Dinner",
        "amount": "3000.00",
        "created_at": "2024-07-30T09:01:42.872444Z",
        "split_method": "E",
        "splits": [
            {
                "id": 1,
                "user": 1,
                "amount": "1000.00",
                "percentage": null
            },
            {
                "id": 2,
                "user": 2,
                "amount": "1000.00",
                "percentage": null
            },
            {
                "id": 3,
                "user": 3,
                "amount": "1000.00",
                "percentage": null
            }
        ]
    },
    {
        "id": 2,
        "description": "Shopping",
        "amount": "4299.00",
        "created_at": "2024-07-30T09:02:42.903836Z",
        "split_method": "X",
        "splits": [
            {
```

Endpoint: GET /api/expenses/{ex} (will give particular expense)

Response:



```
GET /api/expenses/3/
```

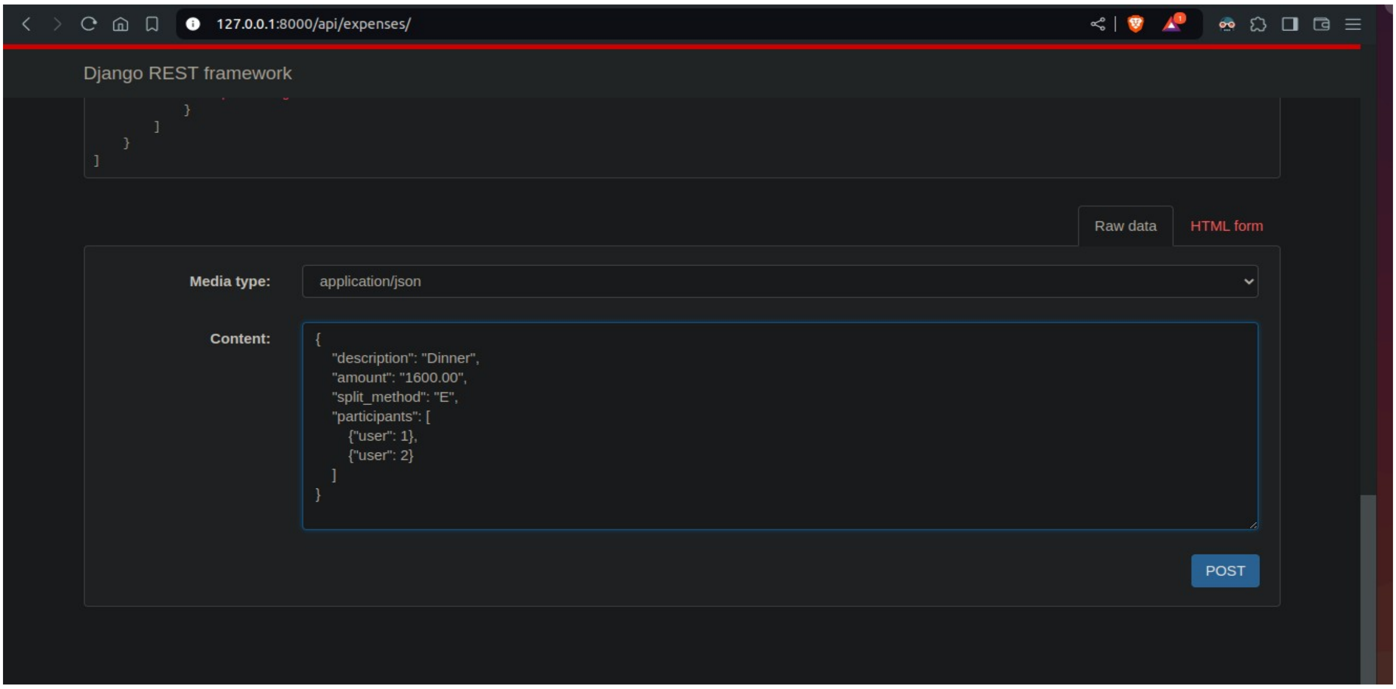
```
HTTP 200 OK
Allow: GET, PUT, PATCH, DELETE, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "id": 3,
    "description": "Party",
    "amount": "2000.00",
    "created_at": "2024-07-30T09:03:34.919643Z",
    "split_method": "P",
    "splits": [
        {
            "user": 1,
            "amount": 1000.0,
            "percentage": 50.0
        },
        {
            "user": 2,
            "amount": 500.0,
            "percentage": 25.0
        },
        {
            "user": 3,
            "amount": 500.0,
            "percentage": 25.0
        }
    ]
}
```

02. Get Expenses of Particular User:

Endpoint: GET /api/expenses/?user=user_id (using this query parameter will give all expenses of a particular user)

eg for user 3, we have not add him for a particular expenses like below:

Request:



```
            }
        ]
    }
]
```

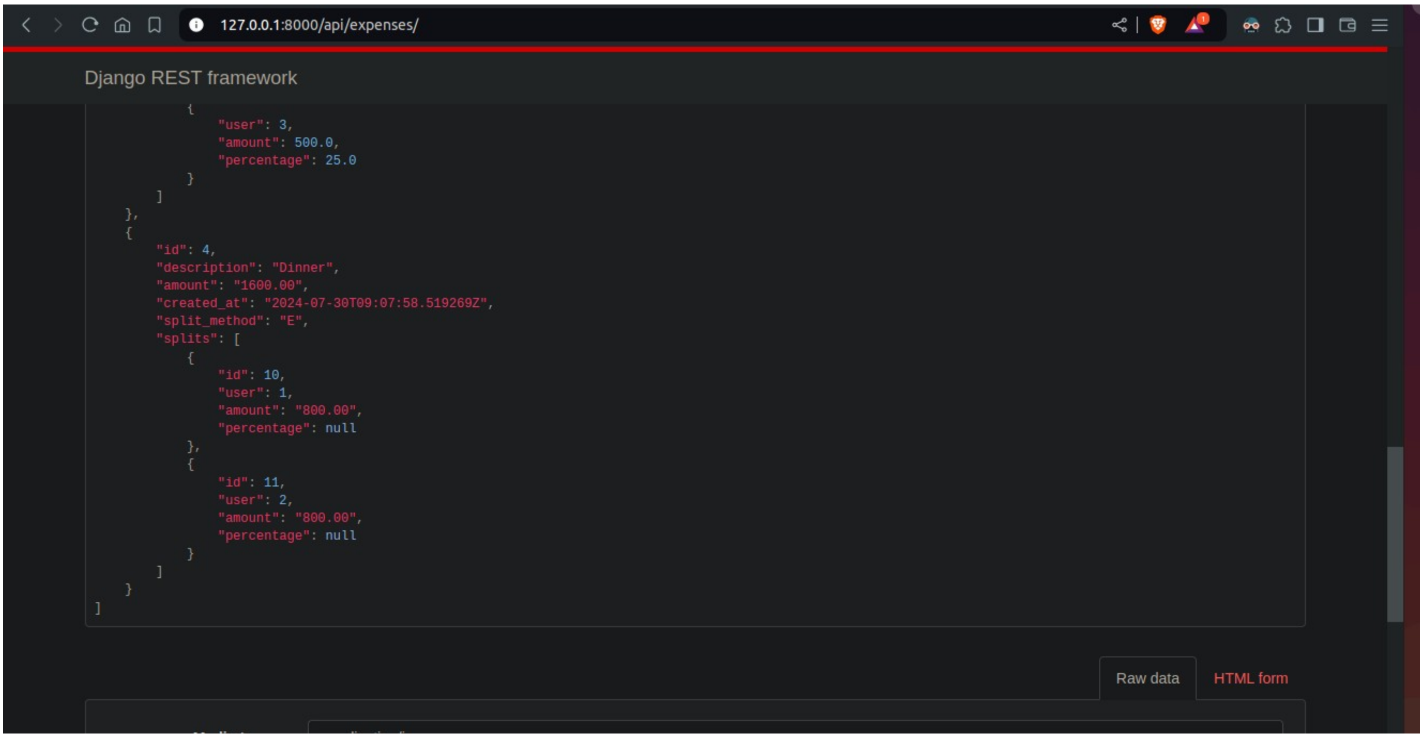
Media type: application/json

Content:

```
{
    "description": "Dinner",
    "amount": "1600.00",
    "split_method": "E",
    "participants": [
        {"user": 1},
        {"user": 2}
    ]
}
```

Now let us get the all expenses:

Response:

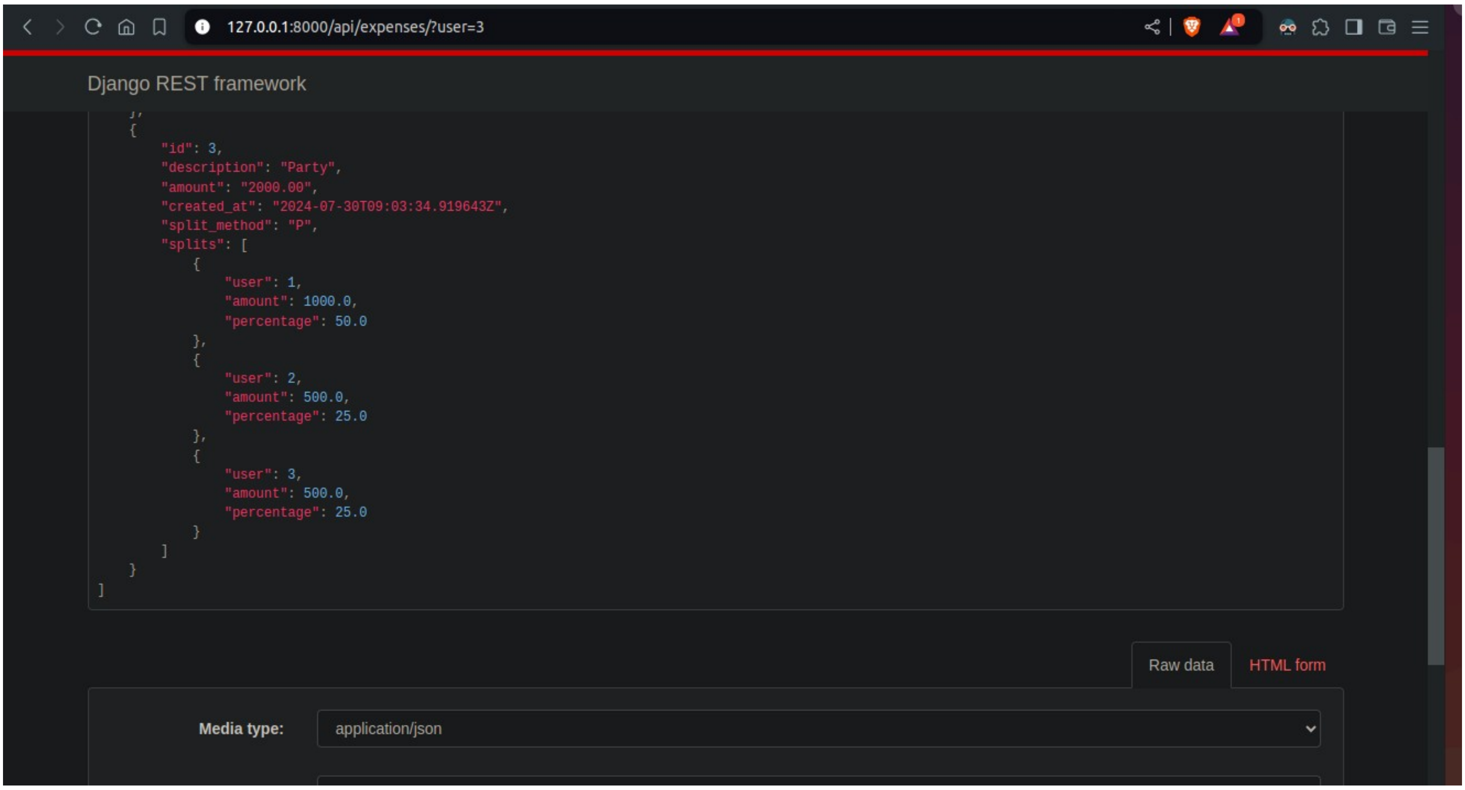


```
                {
                    "user": 3,
                    "amount": 500.0,
                    "percentage": 25.0
                }
            ]
        },
        {
            "id": 4,
            "description": "Dinner",
            "amount": "1600.00",
            "created_at": "2024-07-30T09:07:58.519269Z",
            "split_method": "E",
            "splits": [
                {
                    "id": 10,
                    "user": 1,
                    "amount": "800.00",
                    "percentage": null
                },
                {
                    "id": 11,
                    "user": 2,
                    "amount": "800.00",
                    "percentage": null
                }
            ]
        }
]
```

we can see Expense id = 4 have not user = 3 associated with it and it is the last row of the database.

Now let us get expenses of user = 3

Response:



```
    },
    {
        "id": 3,
        "description": "Party",
        "amount": "2000.00",
        "created_at": "2024-07-30T09:03:34.919643Z",
        "split_method": "P",
        "splits": [
            {
                "user": 1,
                "amount": 1000.0,
                "percentage": 50.0
            },
            {
                "user": 2,
                "amount": 500.0,
                "percentage": 25.0
            },
            {
                "user": 3,
                "amount": 500.0,
                "percentage": 25.0
            }
        ]
    }
]
```

Media type: application/json

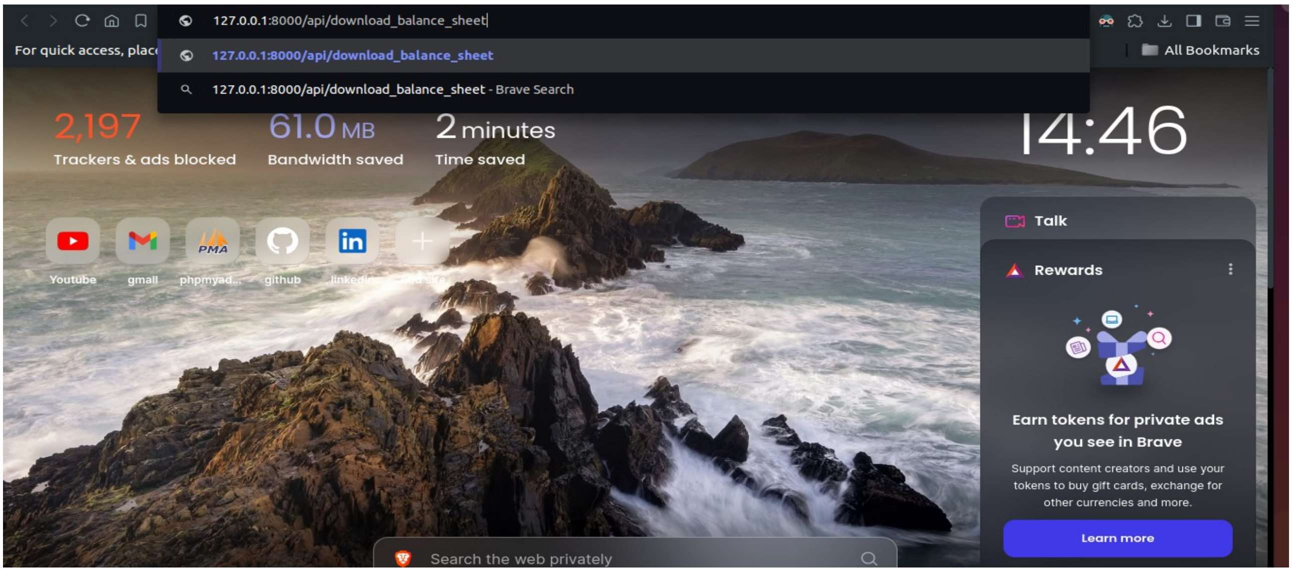
We can see that Expense id = 4 is not shown there as it was not associated with user = 3

#Download Balance Sheet:-

Endpoint: GET /api/download_balance_sheet/

Request:
http://127.0.0.1:8000/api/download_balance_sheet

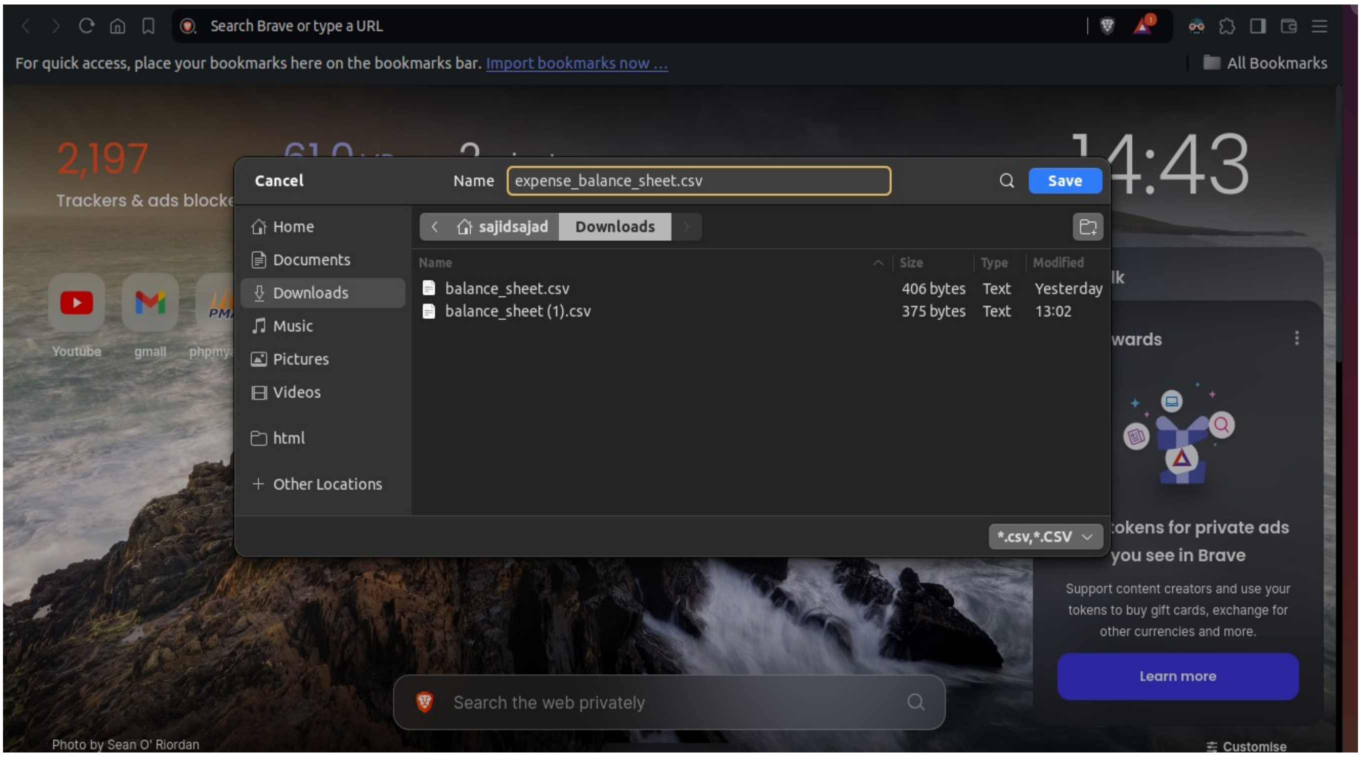
image is on next page:



127.0.0.1:8000/api/download_balance_sheet

127.0.0.1:8000/api/download_balance_sheet

127.0.0.1:8000/api/download_balance_sheet - Brave Search



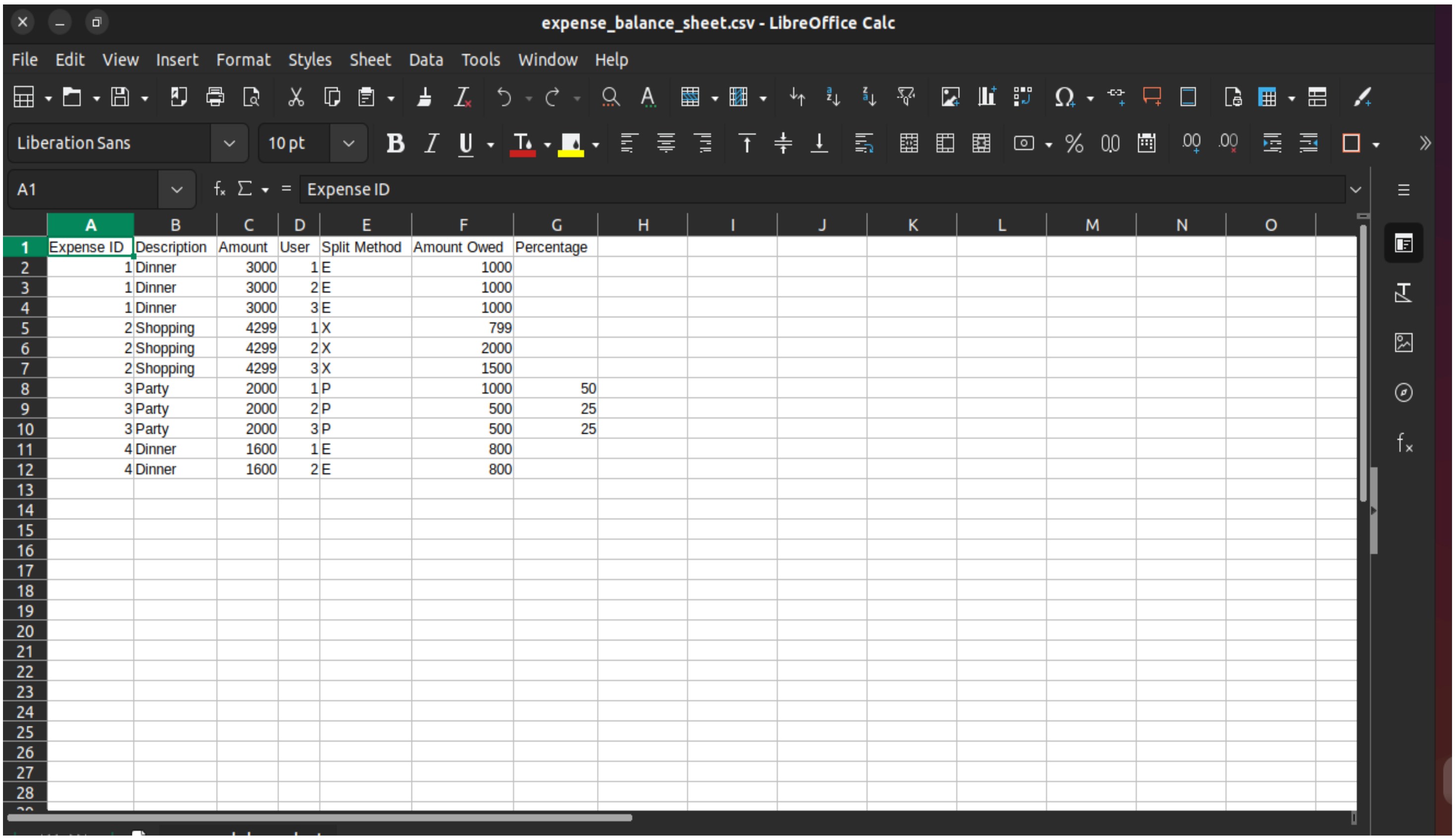
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now ...

Name: expense_balance_sheet.csv

Name	Size	Type	Modified
balance_sheet.csv	406 bytes	Text	Yesterday
balance_sheet (1).csv	375 bytes	Text	13:02

BALANCE SHEET: on next page

Balance Sheet:


expense_balance_sheet.csv - LibreOffice Calc
File Edit View Insert Format Styles Sheet Data Tools Window Help
Liberation Sans
10 pt
A1
Expense ID

Expense ID	Description	Amount	User	Split Method	Amount Owed	Percentage
1	Dinner	3000	1	E	1000	
1	Dinner	3000	2	E	1000	
1	Dinner	3000	3	E	1000	
2	Shopping	4299	1	X	799	
2	Shopping	4299	2	X	2000	
2	Shopping	4299	3	X	1500	
3	Party	2000	1	P	1000	50
3	Party	2000	2	P	500	25
3	Party	2000	3	P	500	25
4	Dinner	1600	1	E	800	
4	Dinner	1600	2	E	800	