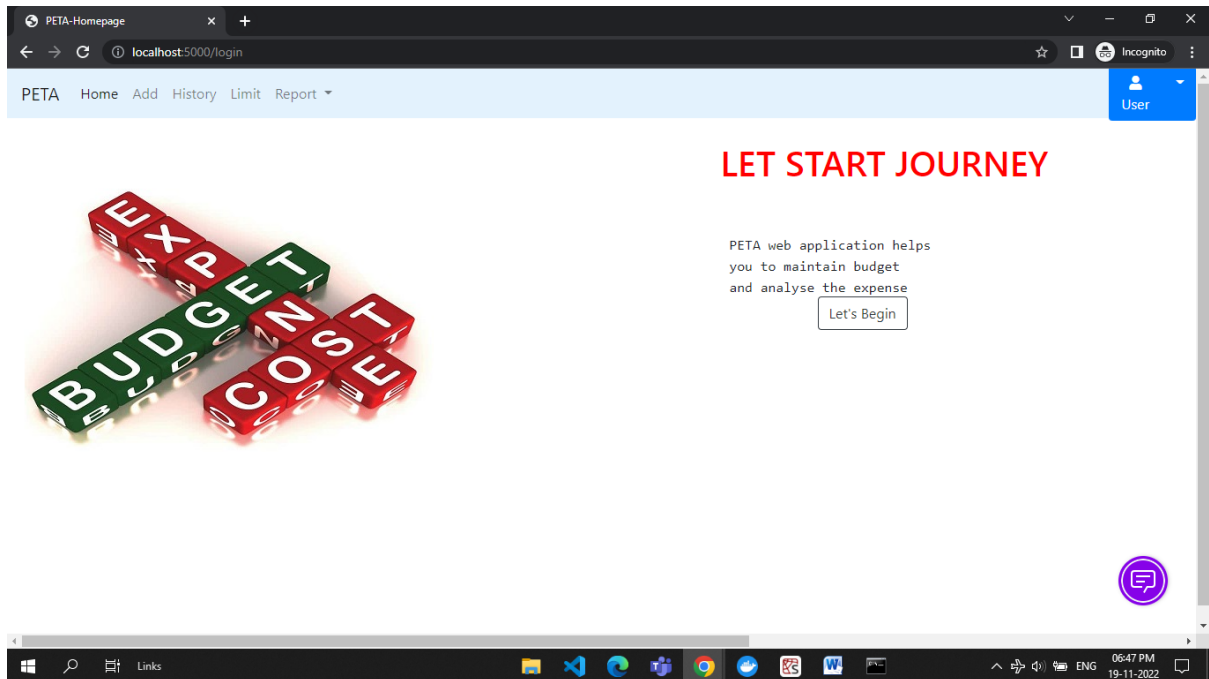


# SPRINT 2

Team ID	PNT2022TMID35573
Project Name	Personal Expense Tracker Application

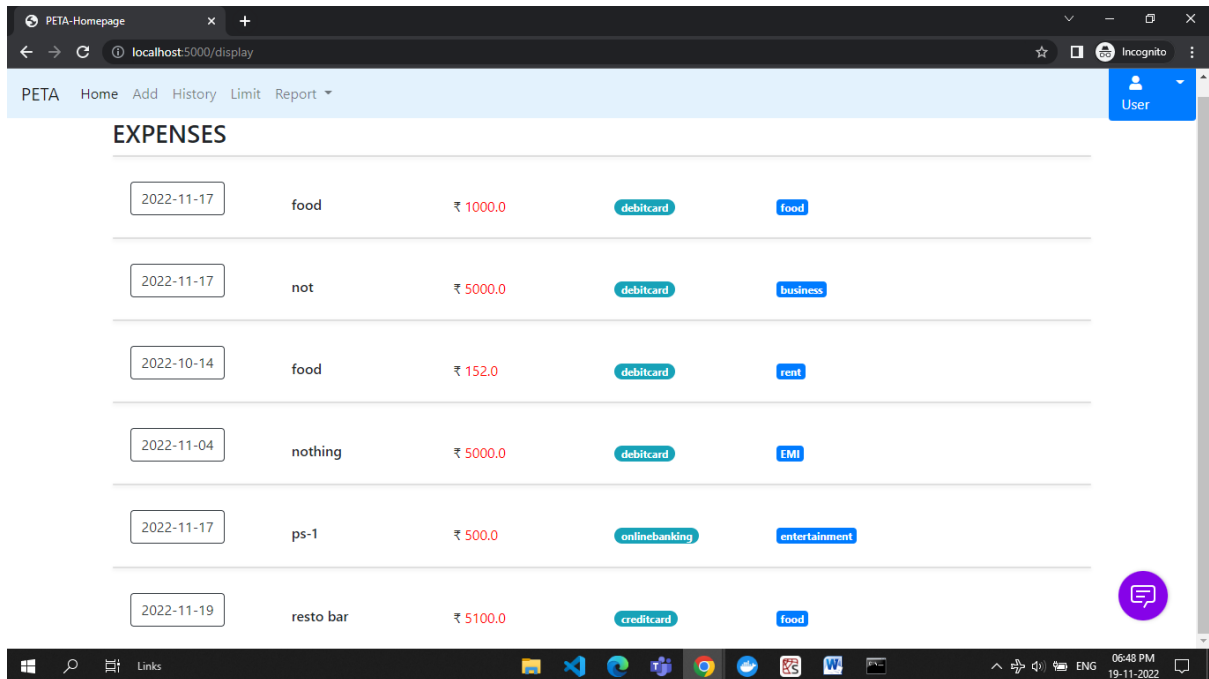
Dashboard:



Add Expense:

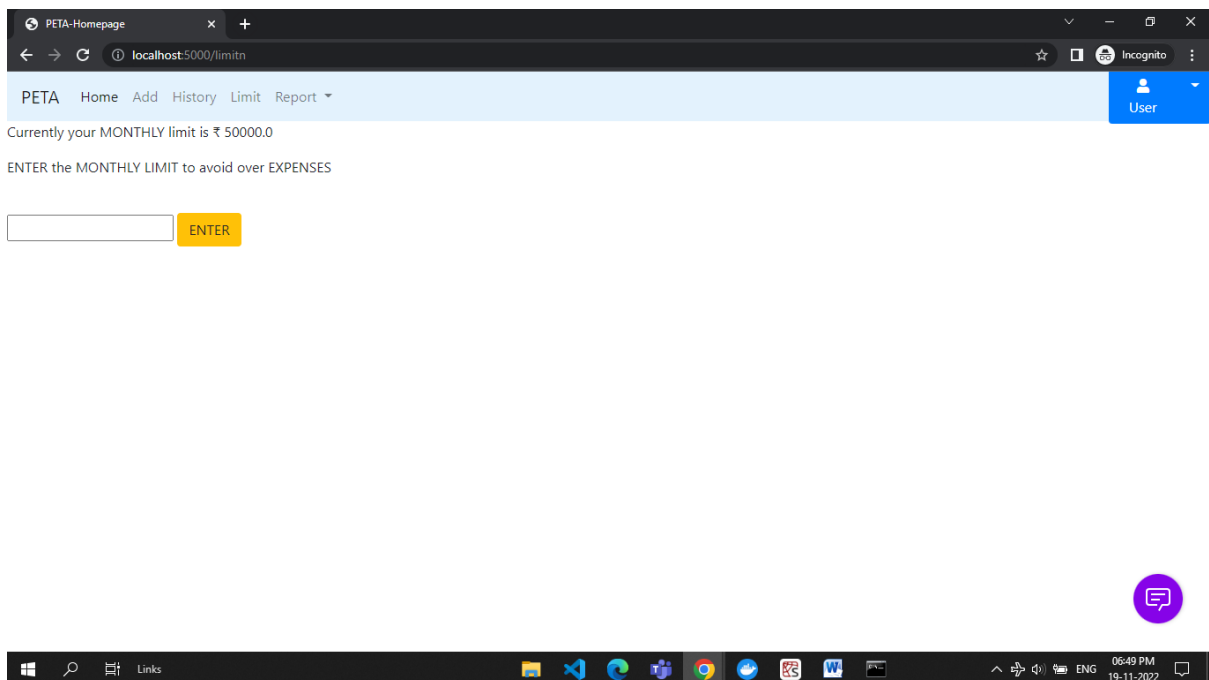
A screenshot of the 'Add Expense' form in the PETA web application. The browser's address bar shows 'localhost:5000/add'. The form includes the following fields: 'Date' with a value of '19-11-2022'; 'Expense name' with a value of 'resto bar'; 'Expense Amount' with a value of '5100'; a 'creditcard' dropdown menu; and a 'food' dropdown menu. A red 'Add' button is positioned at the bottom of the form. The navigation bar at the top is identical to the dashboard view.

## History:



Date	Category	Amount	Payment Method	Sub-category
2022-11-17	food	₹ 1000.0	debitcard	food
2022-11-17	not	₹ 5000.0	debitcard	business
2022-10-14	food	₹ 152.0	debitcard	rent
2022-11-04	nothing	₹ 5000.0	debitcard	EMI
2022-11-17	ps-1	₹ 500.0	onlinebanking	entertainment
2022-11-19	resto bar	₹ 5100.0	creditcard	food

## Limit:



Currently your MONTHLY limit is ₹ 50000.0

ENTER the MONTHLY LIMIT to avoid over EXPENSES

## IBM DB:

The screenshot shows the IBM Db2 on Cloud console interface. The 'Tables' section is active, displaying a list of tables in the 'LFX34122' schema. The 'EXPENSES' table is selected, and its definition is shown on the right. The table has 5 columns: ID, DATE, EXPENSENAME, AMOUNT, and PAYMODE. The table is approximately 5 rows in size and was last updated on 2022-11-19 12:27:40.

Name	Data type	Nullable	Length	Scale
ID	VARCHAR	Y	32	0
DATE	DATE	Y	4	0
EXPENSENAME	VARCHAR	Y	32	0
AMOUNT	DOUBLE	Y		0
PAYMODE	VARCHAR	Y	32	0

## Code:

The screenshot shows the Spyder Python IDE with a Flask application code snippet and its console output. The code defines a route for adding an expense and a route for displaying the list of expenses. The console output shows the application running on http://127.0.0.1:5000/ and receiving several GET requests for static files.

```

100 @app.route('/addexpense', methods=['GET', 'POST'])
101 def addexpense():
102     date = request.form['date']
103     expensename = request.form['expensename']
104     amount = request.form['amount']
105     paymode = request.form['paymode']
106     category = request.form['category']
107     sql = "INSERT INTO EXPENSES VALUES (?, ?, ?, ?, ?)"
108     stmt = ibm_db.prepare(conn, sql)
109     ibm_db.bind_param(stmt, 1, session['id'])
110     ibm_db.bind_param(stmt, 2, date)
111     ibm_db.bind_param(stmt, 3, expensename)
112     ibm_db.bind_param(stmt, 4, amount)
113     ibm_db.bind_param(stmt, 5, paymode)
114     ibm_db.bind_param(stmt, 6, category)
115     ibm_db.execute(stmt)
116     return redirect("/display")
117
118 #DISPLAY -- graph
119 @app.route("/display")
120 def display():
121     sql = "SELECT * FROM EXPENSES WHERE ID=??"
122     stmt = ibm_db.prepare(conn, sql)
123     ibm_db.bind_param(stmt, 1, session['id'])
124     ibm_db.execute(stmt)
125
126     acc = ibm_db.fetch_assoc(stmt)
127     expense = []
128     while acc != False:
129         expense.append(acc)
130         acc = ibm_db.fetch_assoc(stmt)
131     return render_template('display.html', expense = expense)
132
133
134

```

Console output:

```

Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - [19/Nov/2022 18:34:53] "GET / HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:34:54] "GET /static/css/home.css HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:34:54] "GET /static/js/home.js HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:34:54] "GET /static/images/img-2.png HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:34:55] "GET /static/images/hero-bg.png HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:38:24] "GET /static/css/signup.css HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:38:24] "GET /static/images/home-page.png HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:38:25] "GET /static/images/user.png HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:38:25] "GET /static/images/email.png HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:38:25] "GET /static/images/password.png HTTP/1.1" 200 -
127.0.0.1 - [19/Nov/2022 18:38:45] "GET /static/images/... HTTP/1.1" 200 -

```

