

# Project Development Phase

## Sprint – 4

Team ID	PNT2022TMID13990
Project Name	Personal Expense Tracker Application

### Edit Income Page:

Personal Expense Tracker

Home Expense Analysis Income Analysis Profile

### Show Income

Income Date

11-11-2022

Income Category

Salary

Income Note

salary

Income Budget

1000

Back

Update

### Edit Expense Page:

Personal Expense Tracker

Home Expense Analysis Income Analysis Profile

### Show Expense

Expense Date

12-11-2022

Expense Category

Fuel

Expense Note

fuel sample

Expense Budget

200

Back

Update

## Code:

Base file - [HomeBase.html](#)

```
templates > HomeBase.html > ...
33
34 <!--Navbar -->
35 <nav class="navbar navbar-expand-lg navbar-light d-none d-md-block bg-light fixed-top">
36   <div class="container-md">
37     <a class="navbar-brand" href="/dashboard">
38       <span class="fw-bold">Personal Expense Tracker</span>
40     </a>
41     <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavAltMarkup"
42       aria-controls="navbarNavAltMarkup" aria-expanded="false" aria-label="Toggle navigation">
43       <span class="navbar-toggler-icon"></span>
44     </button>
45     <div class="collapse navbar-collapse" id="navbarNavAltMarkup">
46       <div class="navbar-nav ms-auto">
47         <a class="nav-link active" href="/dashboard">Home</a>
48         <a class="nav-link" href="/expenseanalysis">Expense Analysis</a>
49         <a class="nav-link" href="/incomeanalysis">Income Analysis</a>
50         <a class="nav-link" href="/profile">Profile</a>
51       </div>
52     </div>
53   </div>
54 </nav>
55 <!-- End of Navbar -->
56
57 <!-- Bottom Navbar -->
58 <nav class="navbar navbar-light bg-light border-top navbar-expand d-md-none d-lg-none d-xl-none fixed-bottom">
59   <ul class="navbar-nav nav-justified w-100">
60     <li class="nav-item">
61       <a href="/dashboard" class="nav-link active">
62         <span>
63           <i class="fas fa-home"></i>
64         </span>
65       </a>
66     </li>
67     <li class="nav-item">
68       <a href="/expenseanalysis" class="nav-link">
69         <span>
```

Show & Update Analysis – [Show\\_Income.html](#)

```
templates > Show_Income.html > ...
29
30 </div>
31 <div class="mb-3">
32   <label for="category_income" class="form-label">Income Category</label>
33   <select class="form-select" id="category_income" name="category_income">
34     <option value="Salary,salary-id" {% if data.INCOMECATEGORY=="Salary" %} selected="selected"{% endi
35     <option value="Profit,profit-id" {% if data.INCOMECATEGORY=="Profit" %} selected="selected"{% endi
36     <option value="Bonus,bonus-id" {% if data.INCOMECATEGORY=="Bonus" %} selected="selected"{% endif %
37     <option value="Other,other-id" {% if data.INCOMECATEGORY=="Other" %} selected="selected"{% endif %
38   </select>
39 </div>
40 <div class="mb-3">
41   <label for="note_income" class="form-label">Income Note</label>
42   <textarea class="form-control" id="note_income" name="note_income" rows="3"
43     required>{{data.INCOMENOTE}}</textarea>
44 </div>
45 <div class="mb-3">
46   <label for="budget_income" class="form-label">Income Budget</label>
47   <input type="number" value="{{data.INCOMEMOUNT}}" class="form-control" id="budget_income" name="budge
48     required>
49 </div>
50 <div class="row">
51   <div class="col">
52     <div class="d-grid gap-2">
53       <a href="/dashboard" class="btn btn-light">Back</a>
54     </div>
55   <div class="col">
56     <div class="d-grid gap-2">
57       <button class="btn btn-success" type="submit" name="submit">Update</button>
58     </div>
59   </div>
60 </div>
61 </form>
62 </div>
63 </div>
64 </div>
65 </div>
```

## Show & Update Expense – Show\_Expense.html

```
templates > Show_Expense.html > ...
12         <div class="col-xl-6 mt-5 mb-5">
13             <div class="justify-content-center text-center">
14                 {%if error %}
15                 <p style="color: red;">{{error}}</p>
16                 {%endif %}
17             </div>
18             <div class="card">
19                 <div class="card-body">
20                     <h3 class="card-title">Show Expense</h3>
21                     <form class="mt-3" action="" action="{{url_for('updateexpense')}}" method="post">
22                         <input type="hidden" value="{{data.EXPENSEID}}" name="expenseid">
23                         <input type="hidden" value="{{data.CATEGORYNAME}}" name="catid">
24                         <div class="mb-3">
25                             <label for="date_expense" class="form-label">Expense Date</label>
26                             <input type="date" value="{{data.EXPENSEDATE}}" class="form-control" id="date_expense" name="date_expense"
27                                 required>
28                         </div>
29                         <div class="mb-3">
30                             <label for="category_expense" class="form-label">Expense Category</label>
31                             <select class="form-select" id="category_expense" name="category_expense">
32                                 <option value="Entertainment, movie-id" {% if data.EXPENSECATEGORY=="Entertainment" %} selected="selected" {%endif %}>
33                                 <option value="Food, food-id" {% if data.EXPENSECATEGORY=="Food" %} selected="selected" {%endif %}>
34                                 <option value="Fuel, fuel-id" {% if data.EXPENSECATEGORY=="Fuel" %} selected="selected" {%endif %}>
35                                 <option value="other, other-id" {% if data.EXPENSECATEGORY=="Other" %} selected="selected" {%endif %}>
36                             </select>
37                         </div>
38                         <div class="mb-3">
39                             <label for="note_expense" class="form-label">Expense Note</label>
40                             <textarea class="form-control" id="note_expense" name="note_expense" rows="3"
41                                 required>{{data.EXPENSENOTE}}</textarea>
42                         </div>
43                         <div class="mb-3">
44                             <label for="budget_expense" class="form-label">Expense Budget</label>
45                             <input type="number" value="{{data.EXPENSEAMOUNT}}" class="form-control" id="budget_expense" name="budget_expense"
46                                 required>
47                         </div>
48                     </div>
49                 </div>
50             </div>
51         </div>
52     </div>
53 </div>
54 </div>
55 </div>
56 </div>
57 </div>
58 </div>
59 </div>
60 </div>
61 </div>
62 </div>
63 </div>
64 </div>
65 </div>
66 </div>
67 </div>
68 </div>
69 </div>
70 </div>
71 </div>
72 </div>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
78 </div>
79 </div>
80 </div>
81 </div>
82 </div>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </div>
92 </div>
93 </div>
94 </div>
95 </div>
96 </div>
97 </div>
98 </div>
99 </div>
100 </div>
```

## Backend file – connect.py

```
connection > connect.py > getexpensevalues
1 import ibm_db
2 from flask import session
3
4 def Connection():
5     try:
6         conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=21fecfd8-47b7-4937-840d-d791d0218660.bs2io90l08kqb1odd1cg.databases.appdomain.c
7         print ("Database connected Successfully !")
8         return conn
9     except:
10         print ("Unable to connect: ", ibm_db.conn_errormsg())
11
12 def getincomevalues(id,conn):
13     sql = "SELECT * FROM INCOME WHERE INCOMEID = '"+id+"'"
14     lis={}
15     try:
16         stmt = ibm_db.exec_immediate(conn, sql)
17         dictionary = ibm_db.fetch_both(stmt)
18         # print(dictionary)
19         while dictionary!=False:
20             lis=dictionary
21             dictionary = False
22         print("income value got successfully !")
23     except:
24         print("Error while get the income value")
25     return lis
26
27 def getexpensevalues(id,conn):
28     sql = "SELECT * FROM EXPENSE WHERE EXPENSEID = '"+id+"'"
29     lis={}
30     try:
31         stmt = ibm_db.exec_immediate(conn, sql)
32         dictionary = ibm_db.fetch_both(stmt)
33         while dictionary!=False:
34             lis=dictionary
35             dictionary = False
36         print("expense value got successfully !")
37     except:
```

## Docker file

```
Dockerfile > ...
1 FROM python:latest
2 WORKDIR /app
3 COPY requirements.txt ./
4 RUN pip install -r requirements.txt
5 COPY . .
6 EXPOSE 5000
7 CMD ["python", "./app.py"]
```

## Requirements.txt

```
requirements.txt
1 flask
2 ibm_db
3 sendgrid
4 flask_session
```

