# Chapter 4: Testing

This chapter presents a complete manual test plan for the Budget Management App. Each test case corresponds to a system requirement (functional or non-functional) identified in Chapter 1. Tests were conducted manually using a browser, form submissions, and data review through the user interface.

## 1. Functional Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test ID | Feature | Input | Expected Output | Actual Output | Pass/Fail |
| TC-FR1 | Add Expense | Amount=50, Category=Food, Date=Today | Expense is saved and listed in summary | Displayed in summary table | Pass |
| TC-FR2 | Edit Expense | Edit amount from 50 to 60 | Updated value saved and shown | Updated in table | Pass |
| TC-FR3 | Set Monthly Budget | Month=July, Budget=500 | Budget is stored and shown in summary | Budget shown in summary | Pass |
| TC-FR4 | Budget Notification | Expenses reach 90% of budget | Notification/alert shown | Alert displayed on summary | Pass |
| TC-FR5 | View Summary | Filter by July, Category=Food | Filtered results shown | Correct results shown | Pass |
| TC-FR6 | Export Report | Click Export CSV | Download starts | CSV file downloaded | Pass |

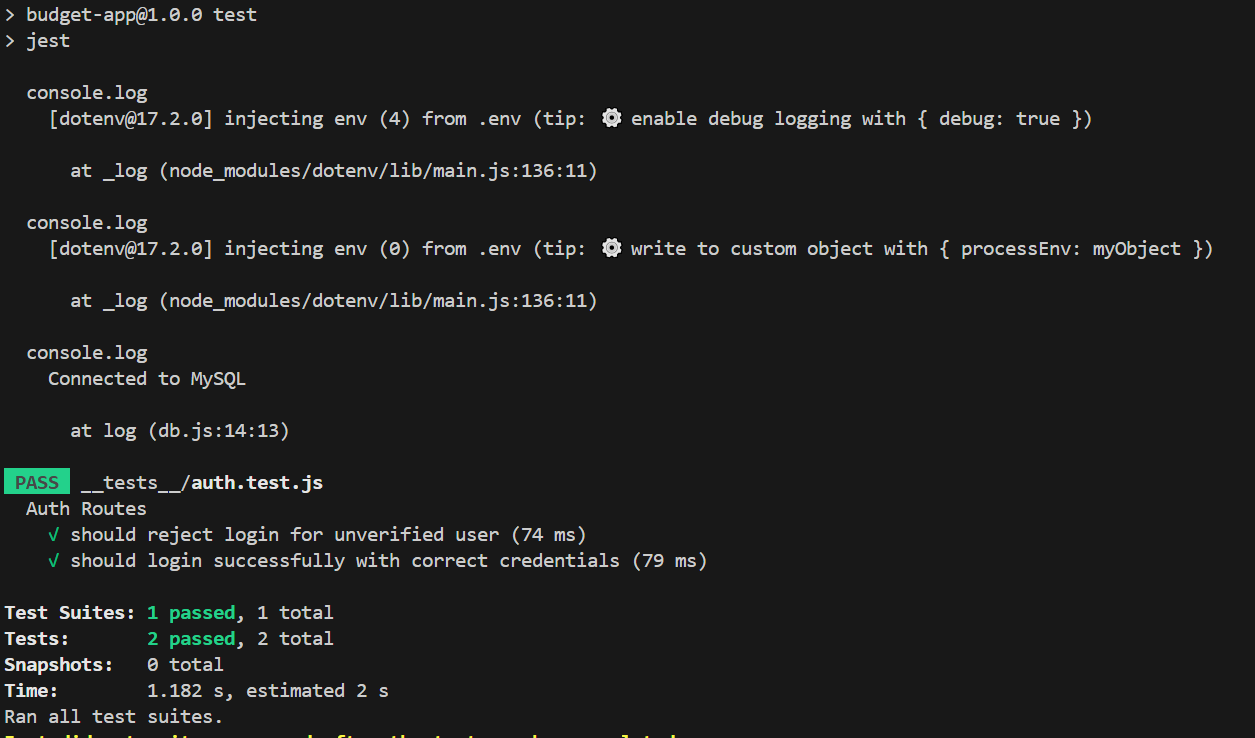
## 2. Non-Functional Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test ID | Requirement | Test Description | Expected Outcome | Actual Outcome | Pass/Fail |
| TC-NFR1 | Response Time | Submit Add Expense form | Page responds within 2 seconds | Response < 1 sec | Pass |
| TC-NFR2 | Responsive Design | Open site on phone and desktop | Layout adjusts correctly | Bootstrap responsive layout | Pass |
| TC-NFR3 | Data Security | Submit expense with SQL special chars | No SQL injection or error | Handled securely | Pass |
| TC-NFR4 | Offline Use | Disconnect internet, try adding expense | Page opens, form functions locally | Form loads and works | Pass |
| TC-NFR5 | UI Usability | Navigate through UI pages | Simple, clean design with no confusion | Intuitive and minimal UI | Pass |

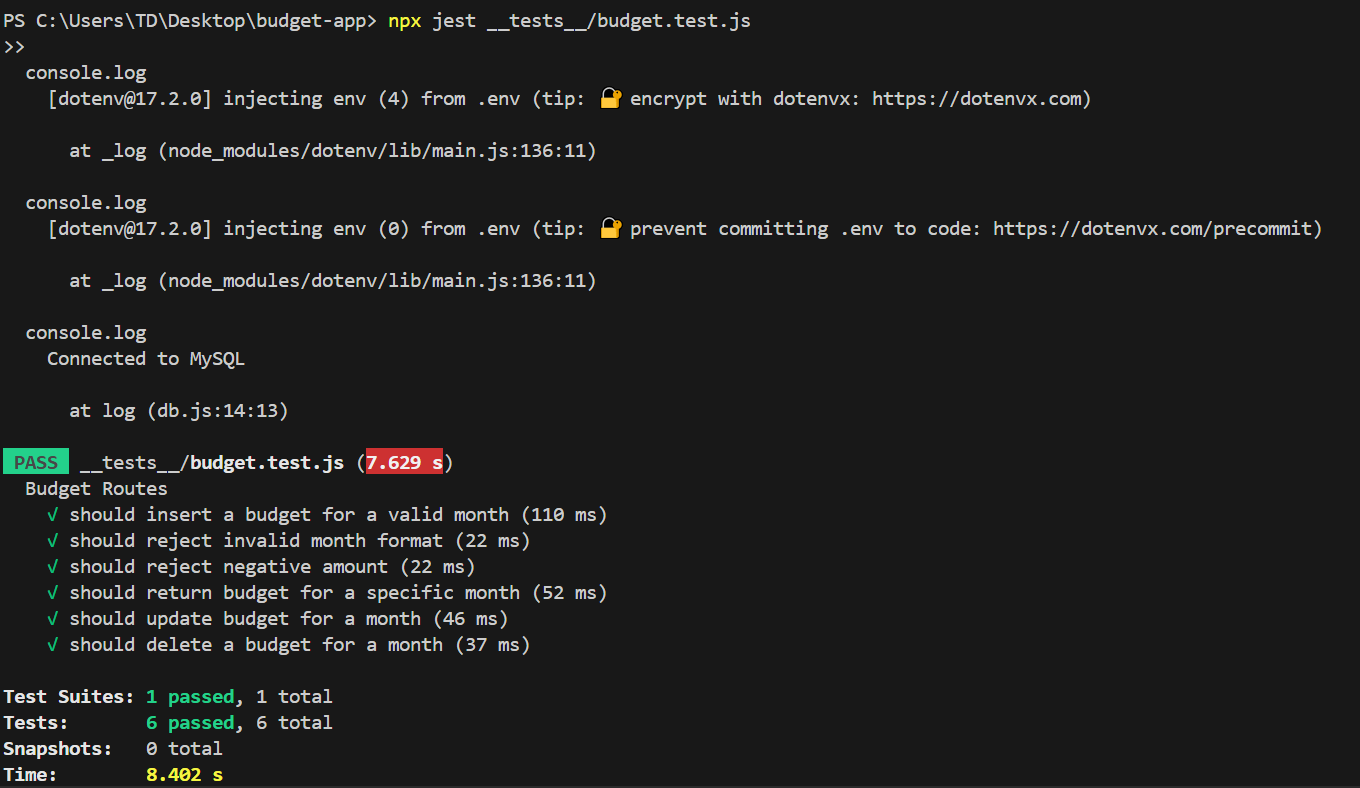
## 3. Summary

All functional and non-functional requirements were manually tested using realistic inputs. All test cases passed successfully, confirming that the implemented features match the system specifications. Screenshots of test execution and exported files were also reviewed to verify correctness.

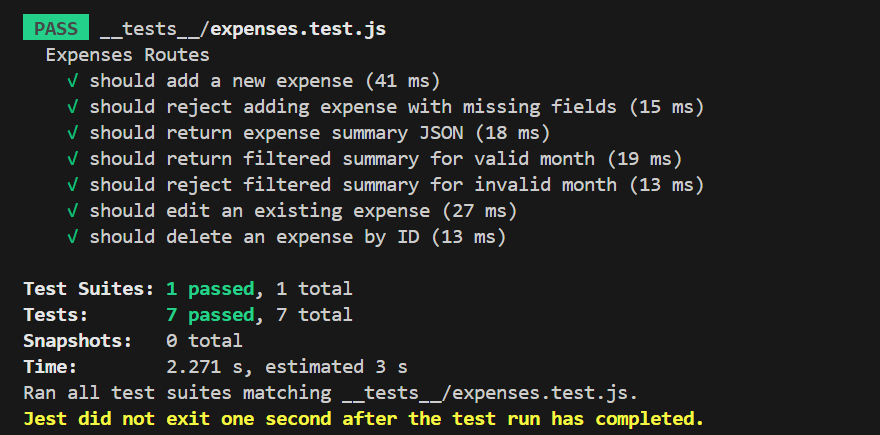
**4.Automated testing**

**- Testing verified and unverified login :**

**- Testing inserting new budget , updating & deleting budget**

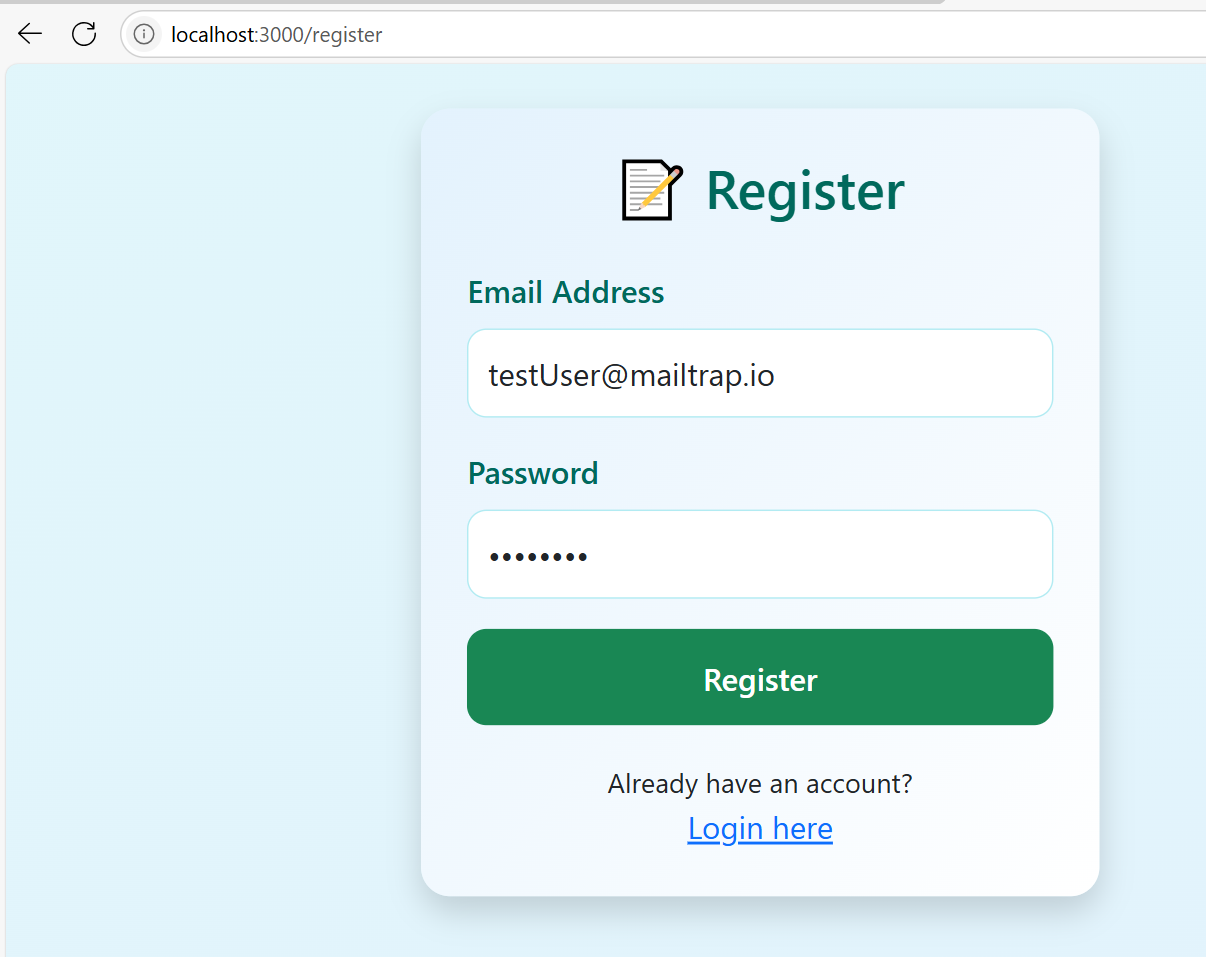


**-Testing expenses**

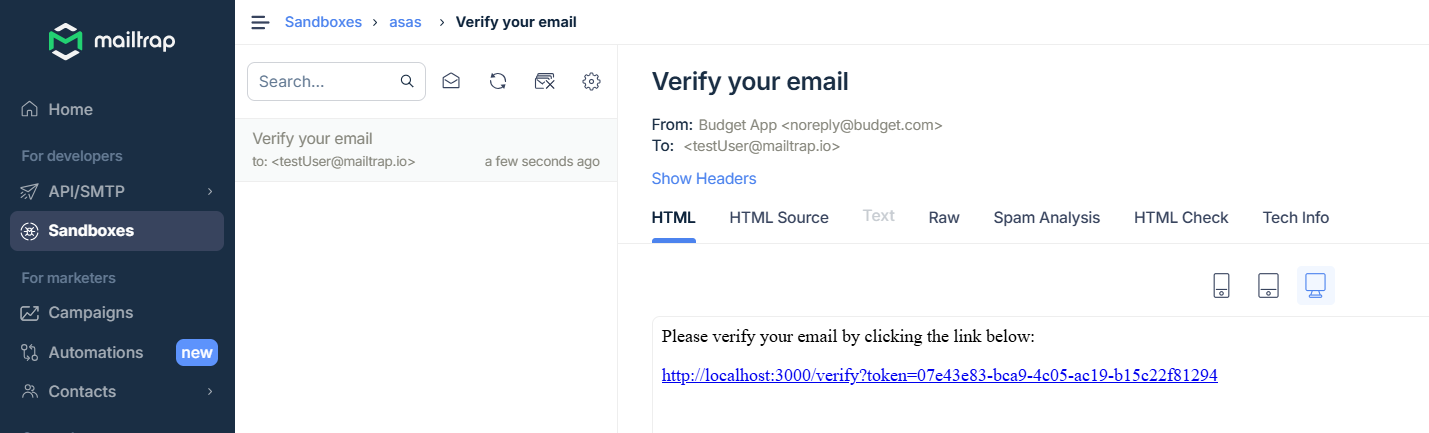


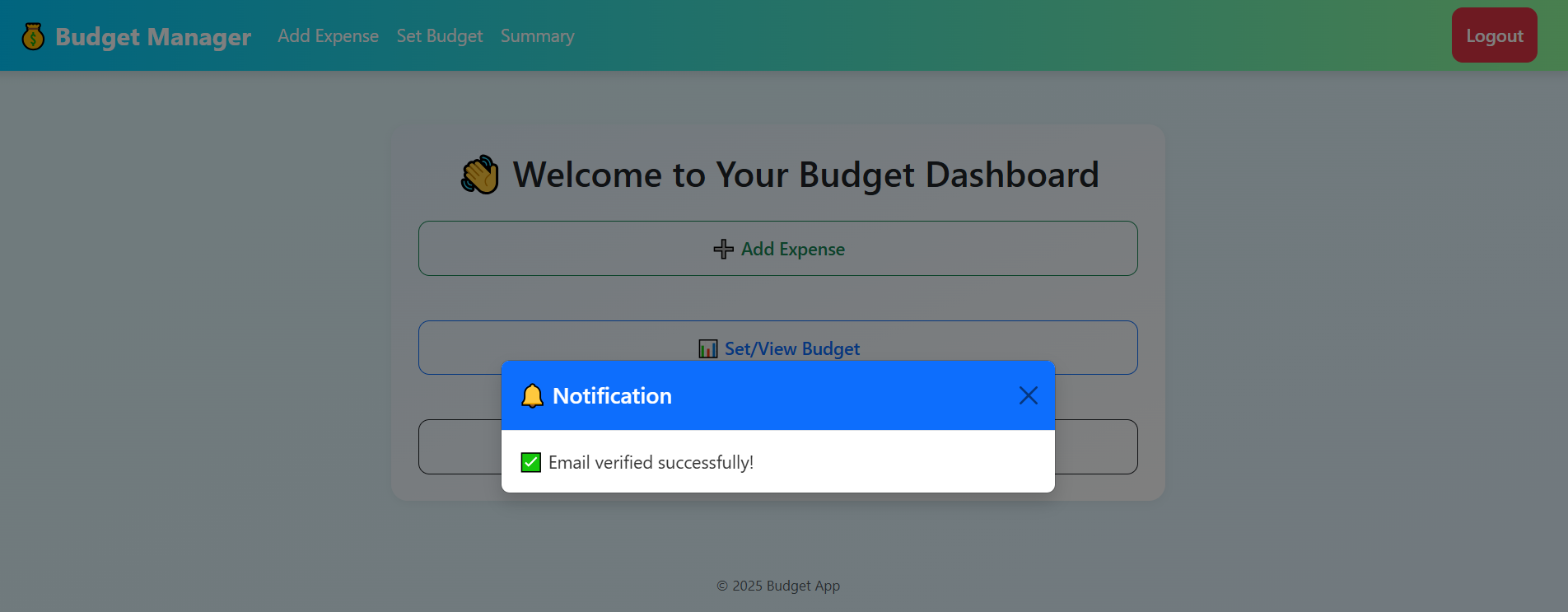
**Testing from UI :**

**Registration form**

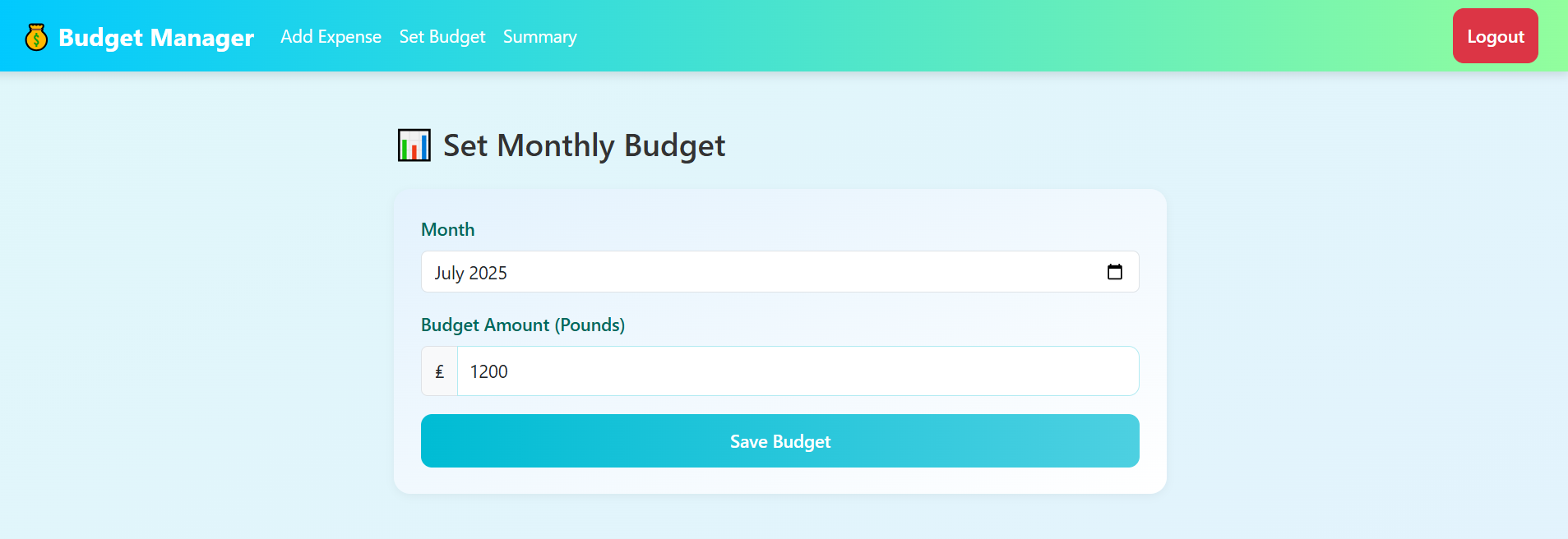


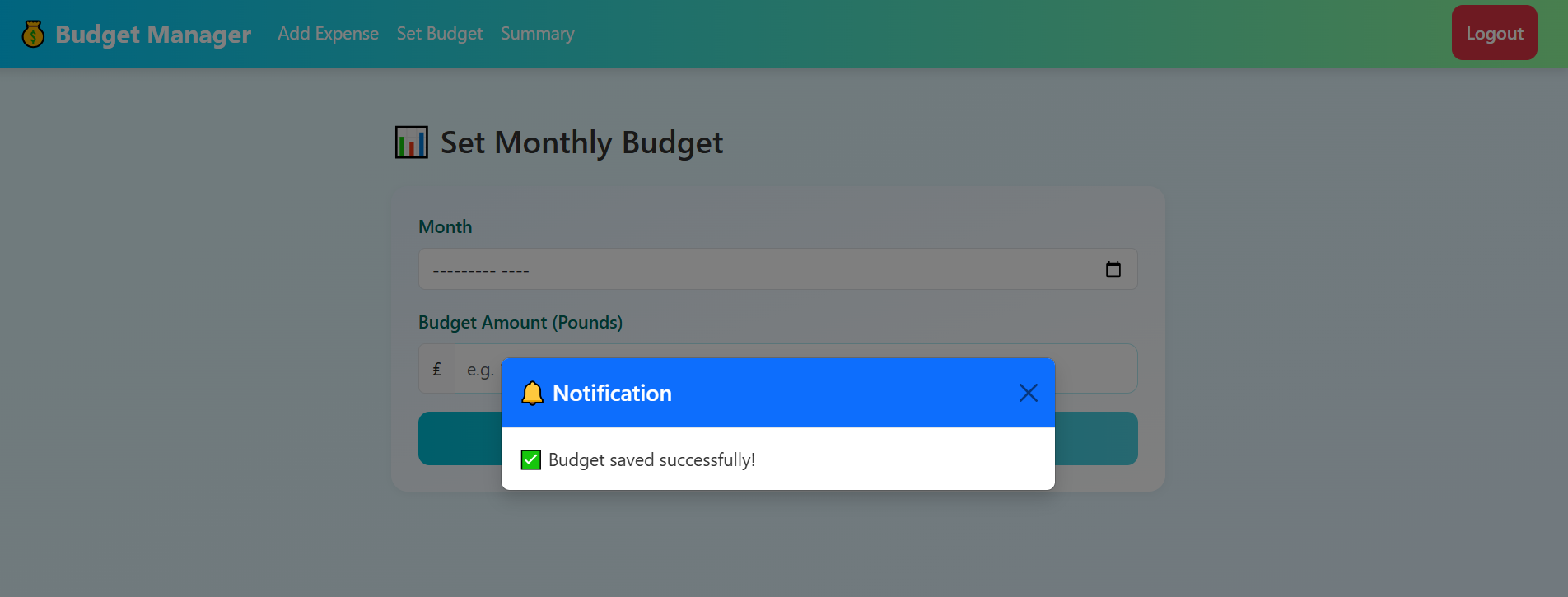
**Email verification**

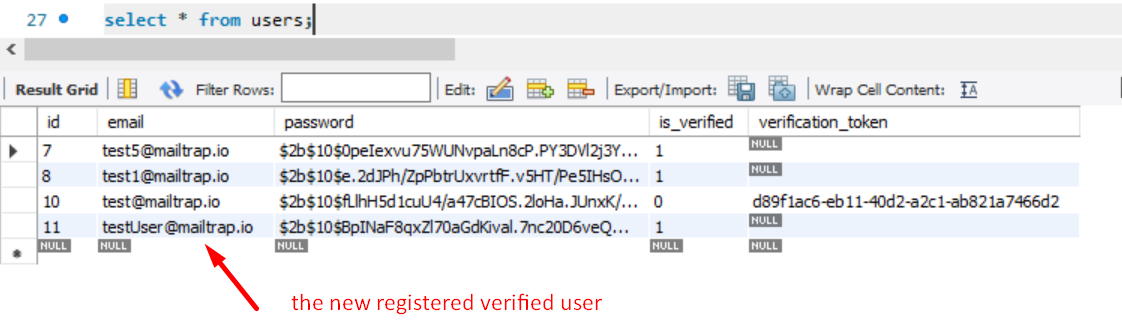




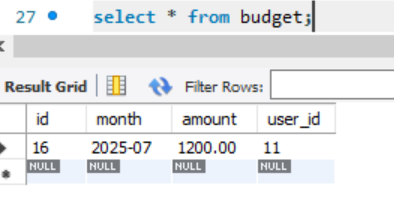
Inserting a budget



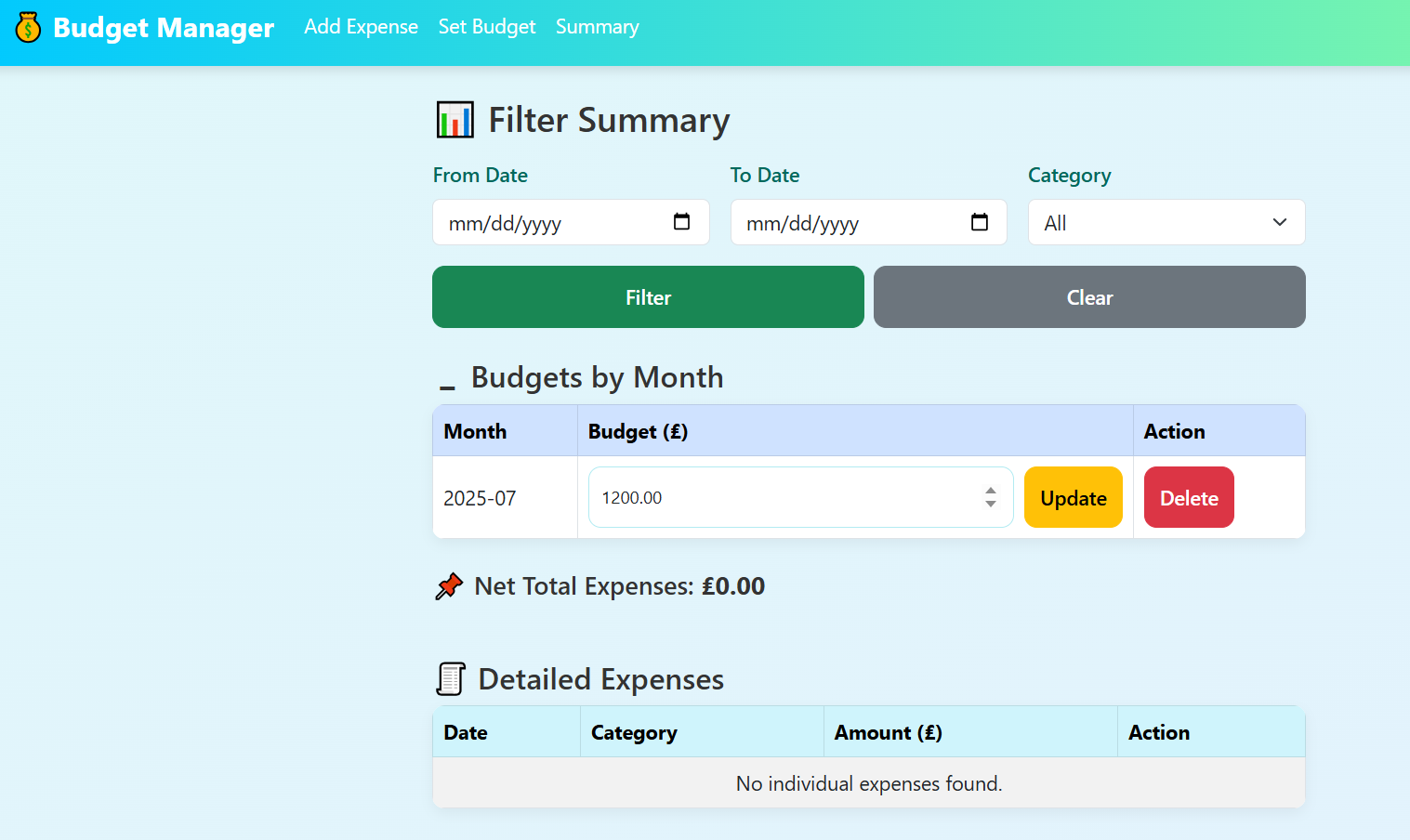




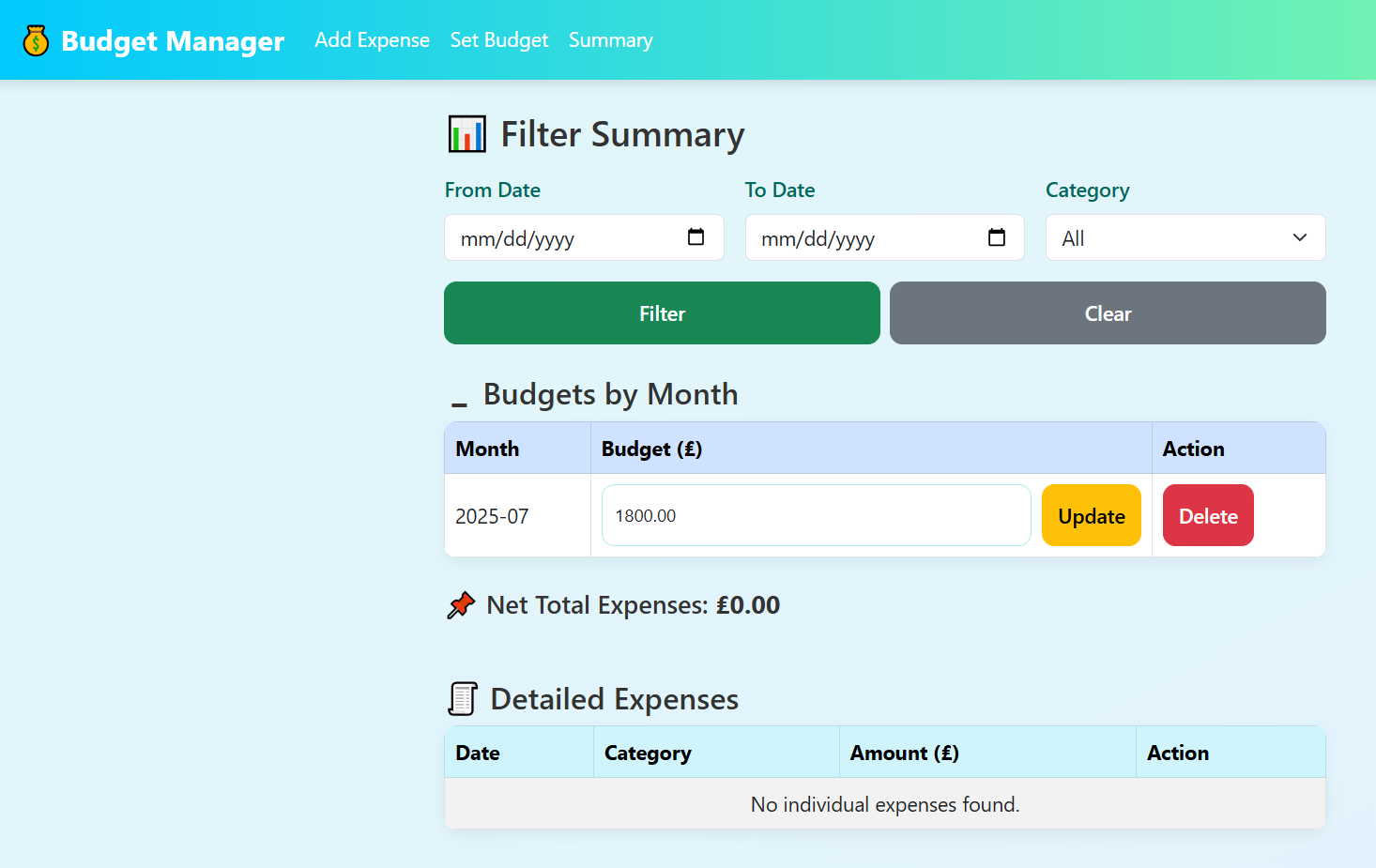
Inserted budget

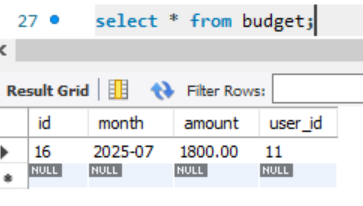


Budget showing in the summary page

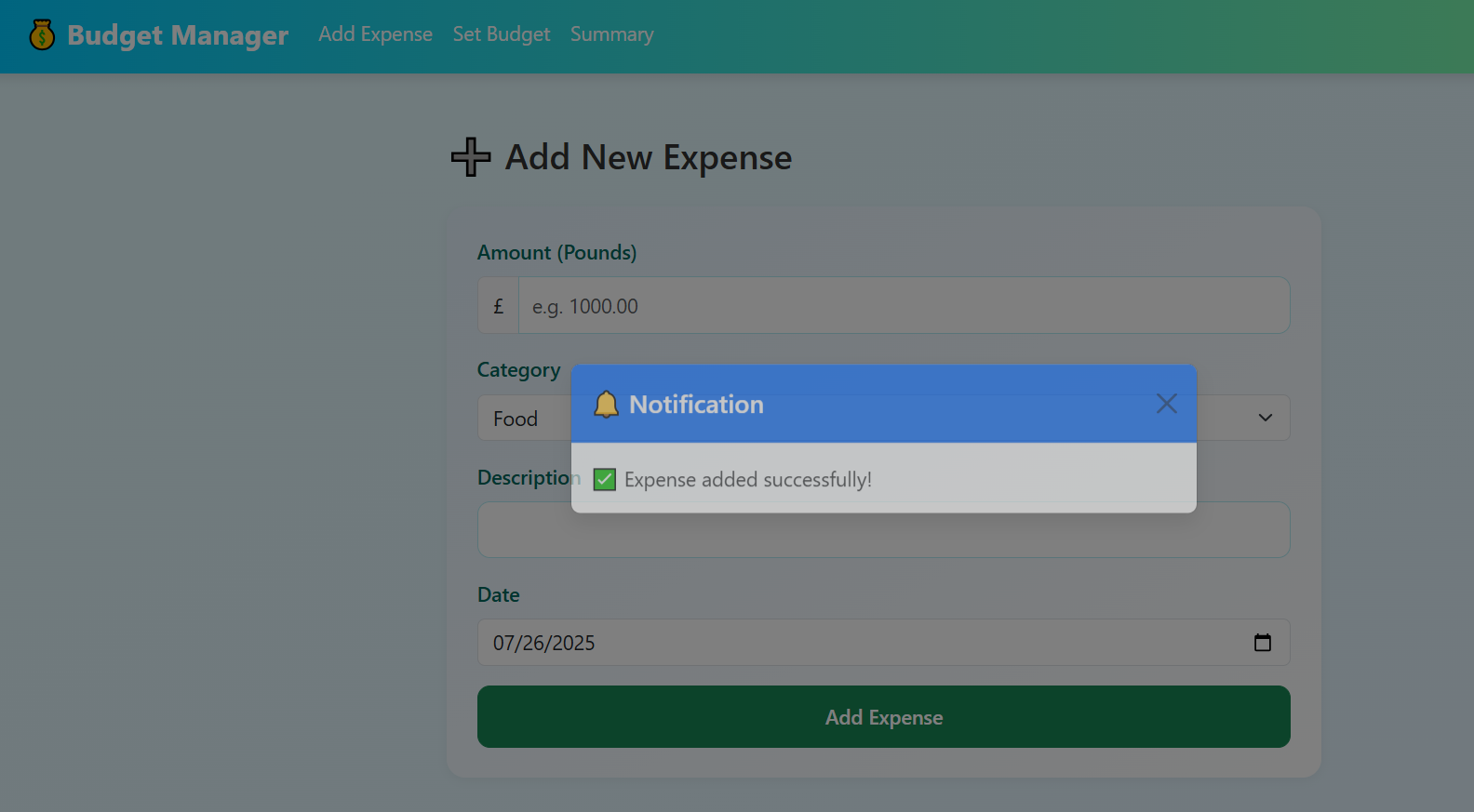


Updating budget from the summary page

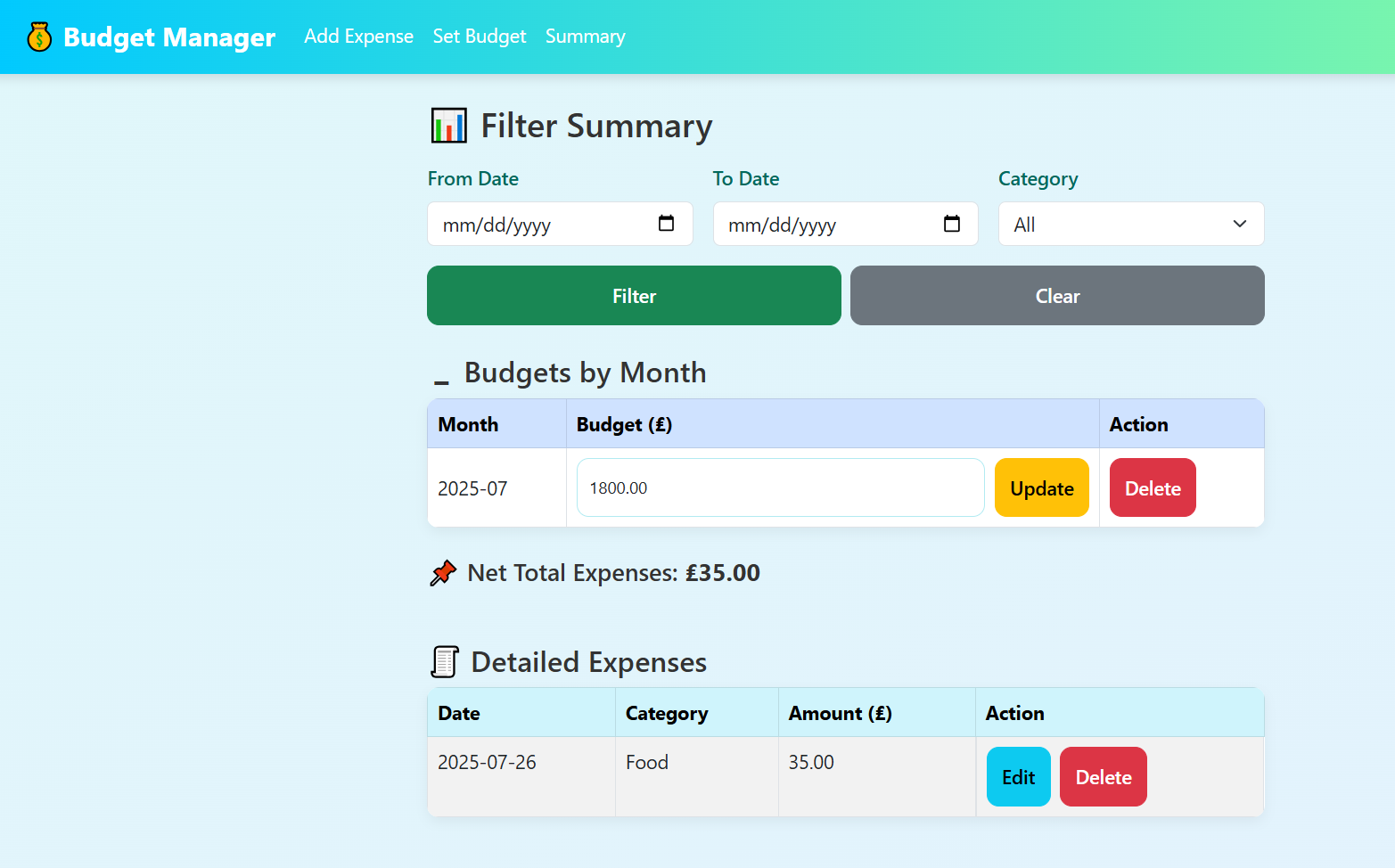




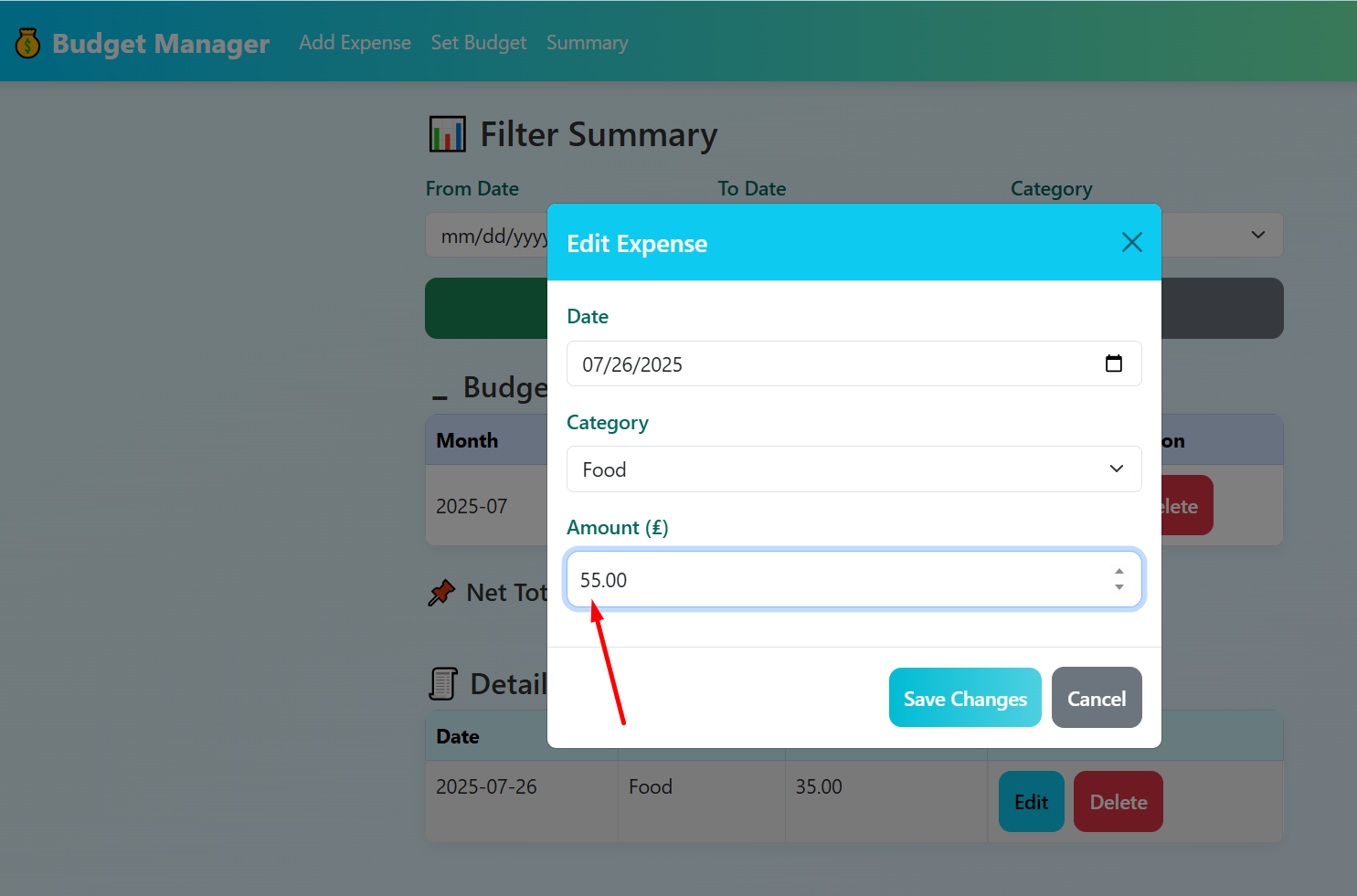
Inserting expenses

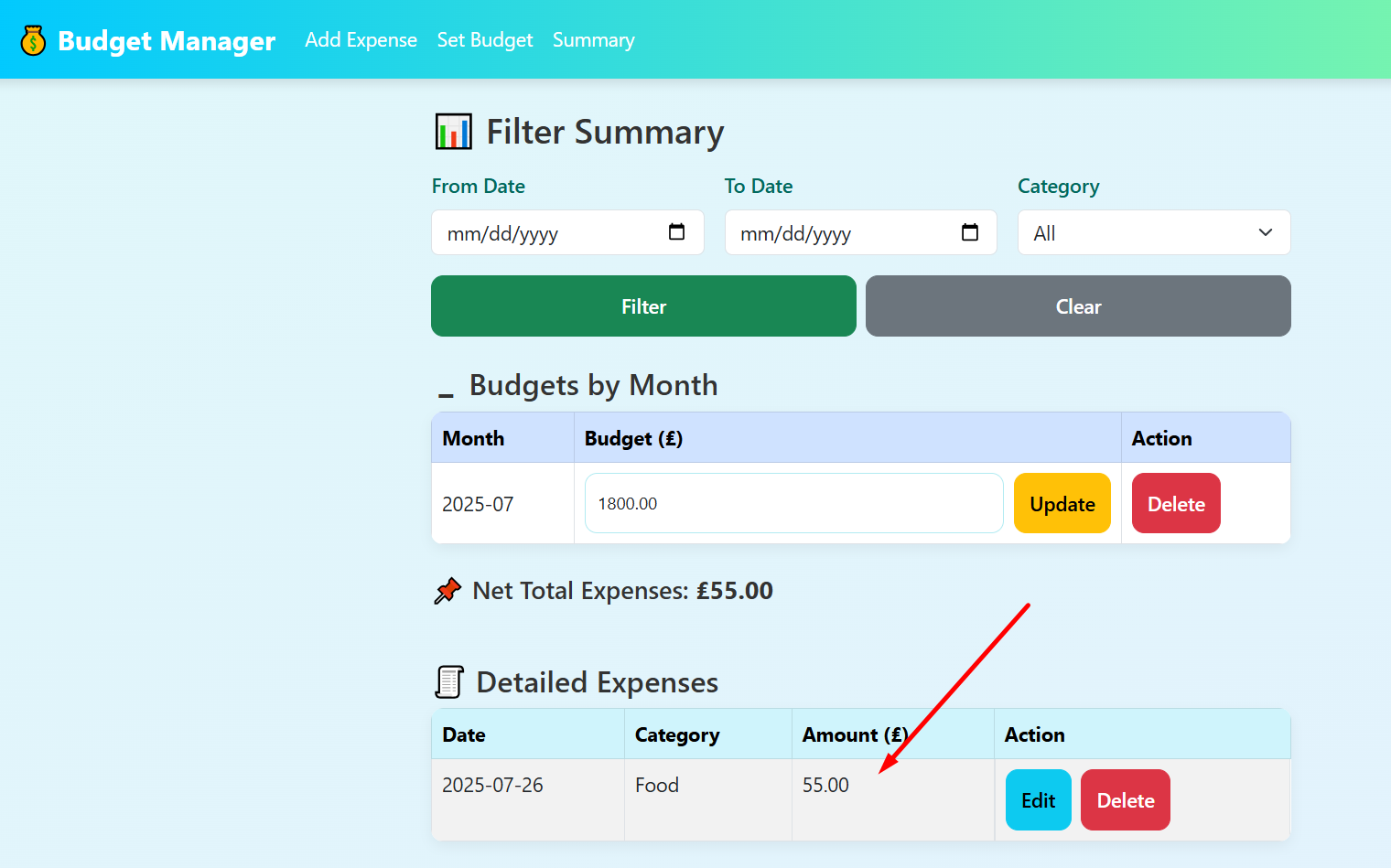


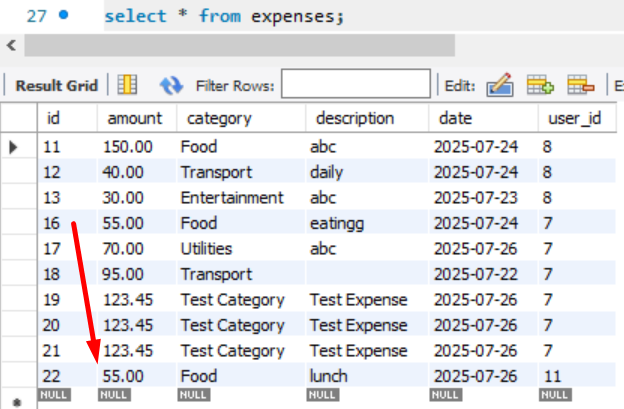
Expense showing in the summary page



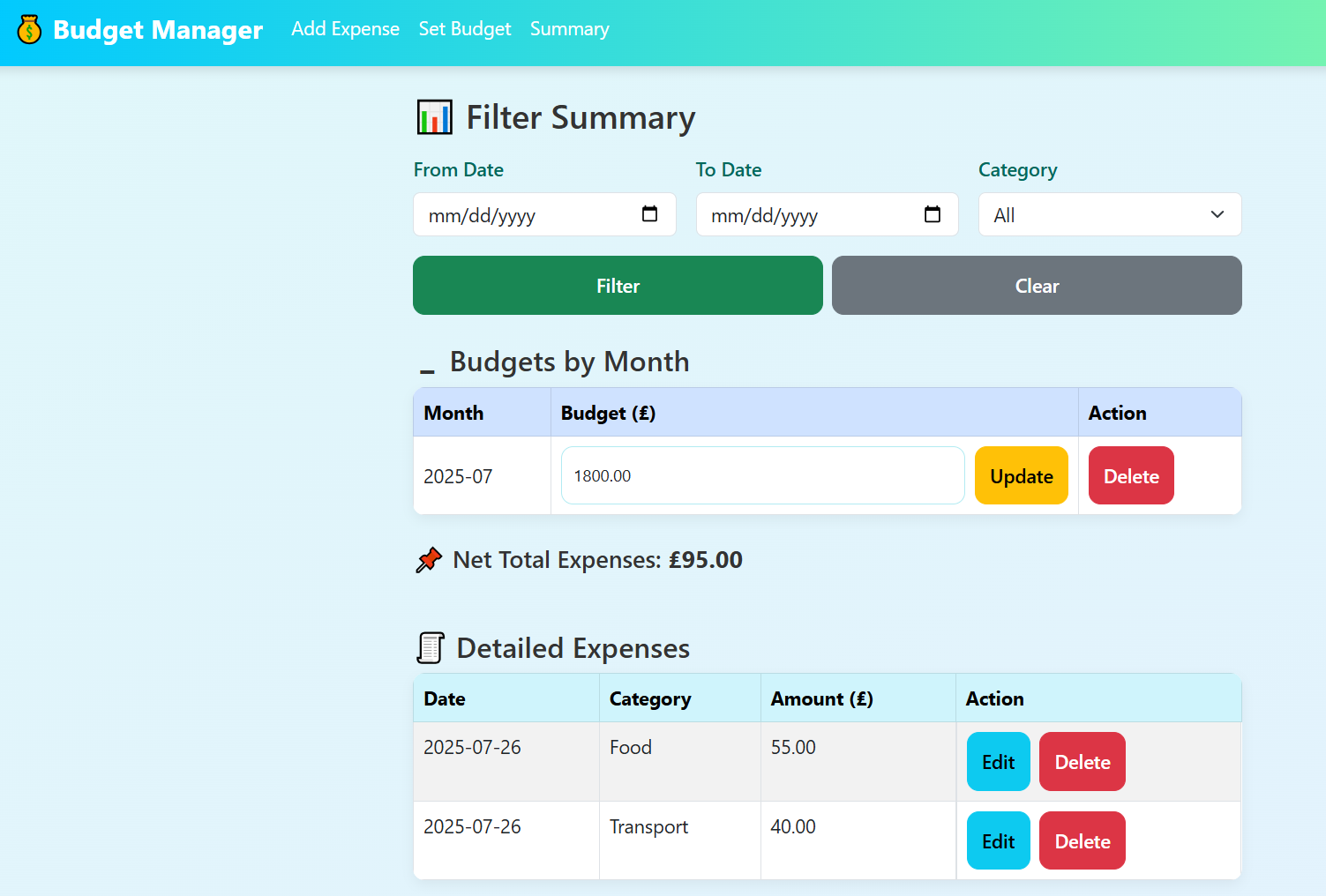
Updating expense details (changing amount from 35 to 55 pounds



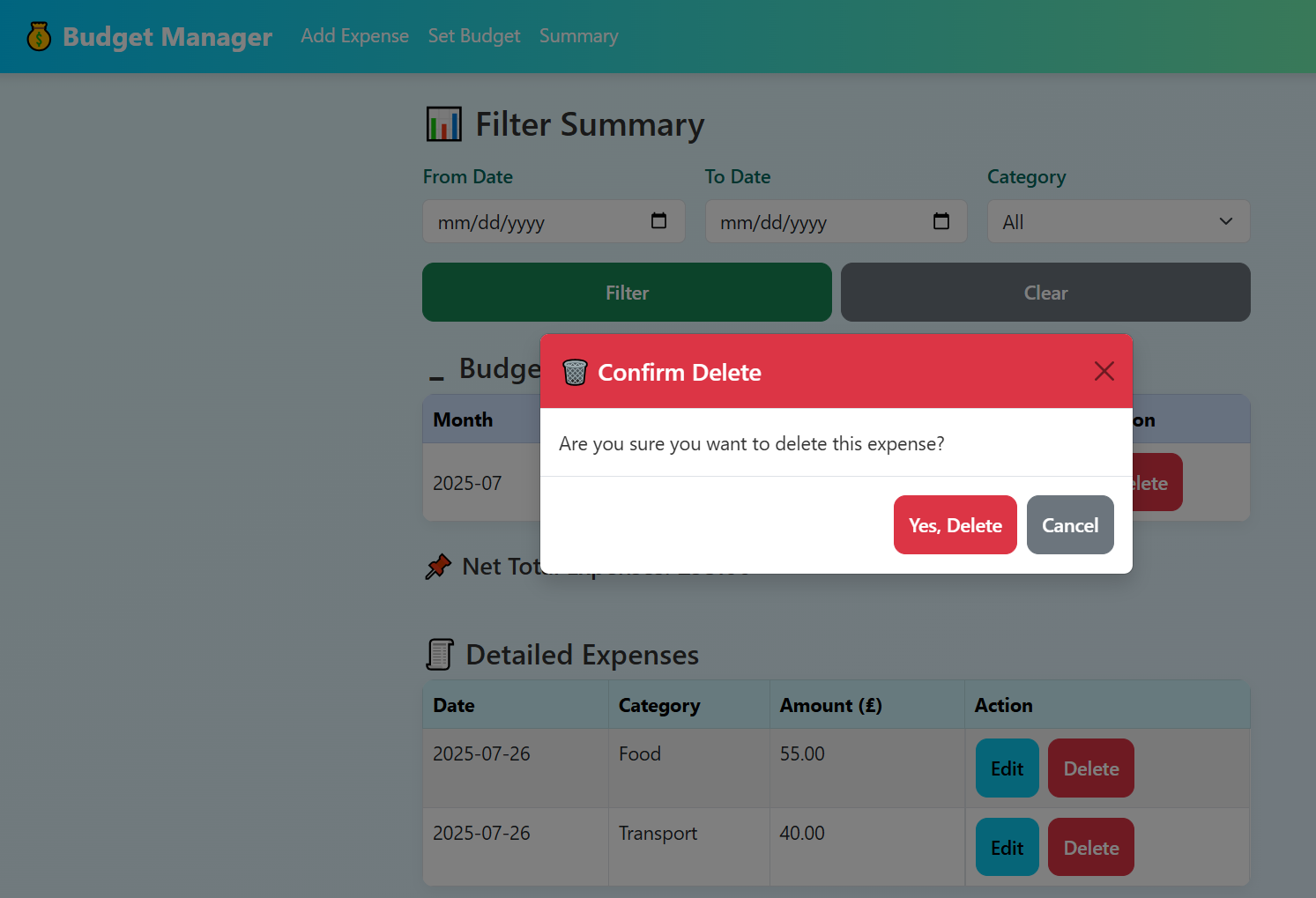


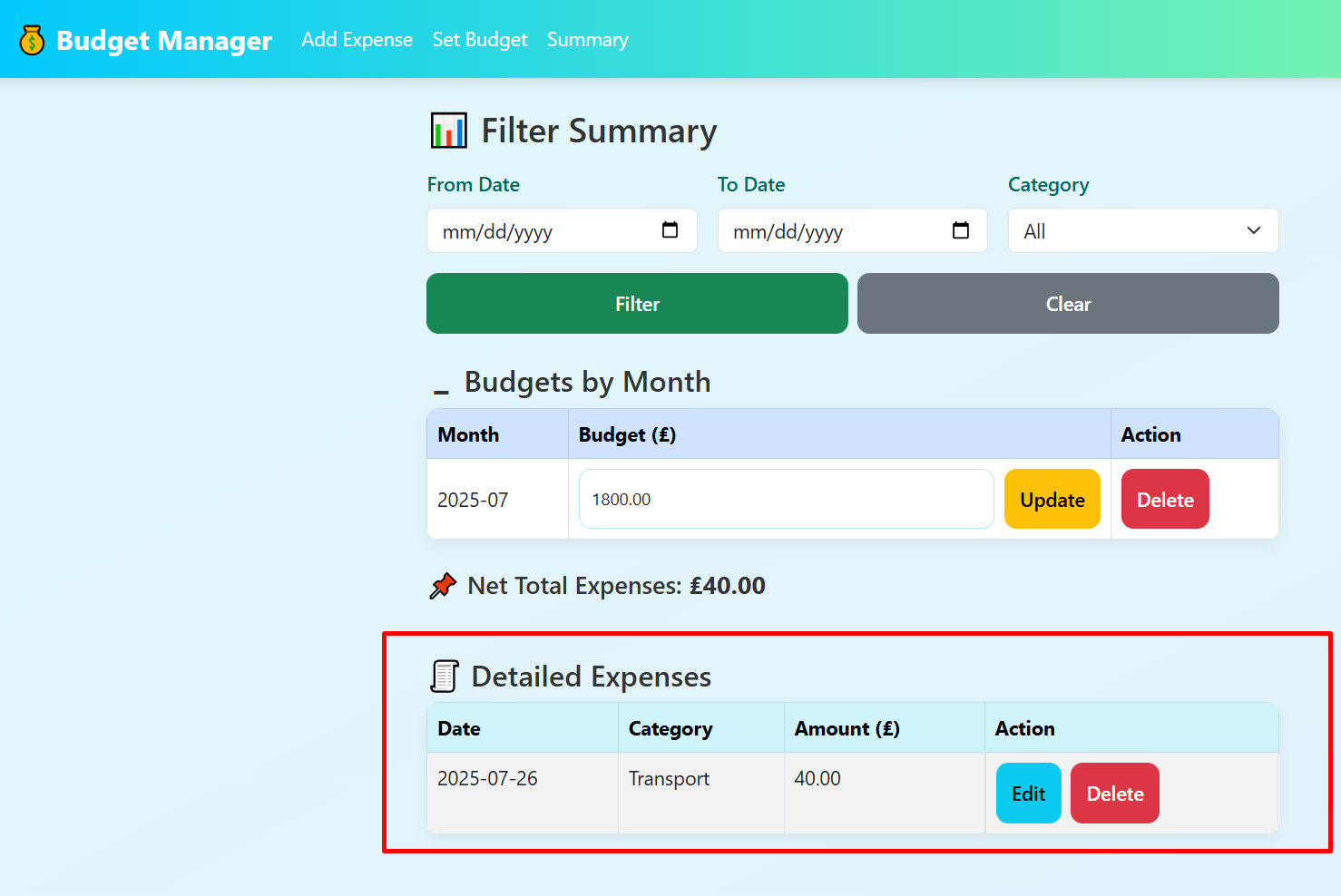


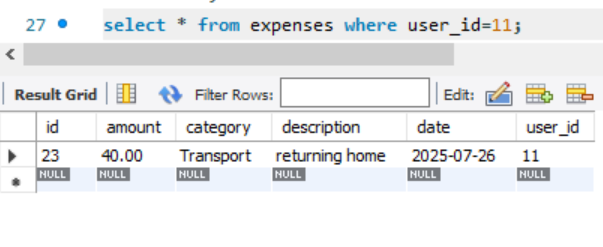
Adding new expense



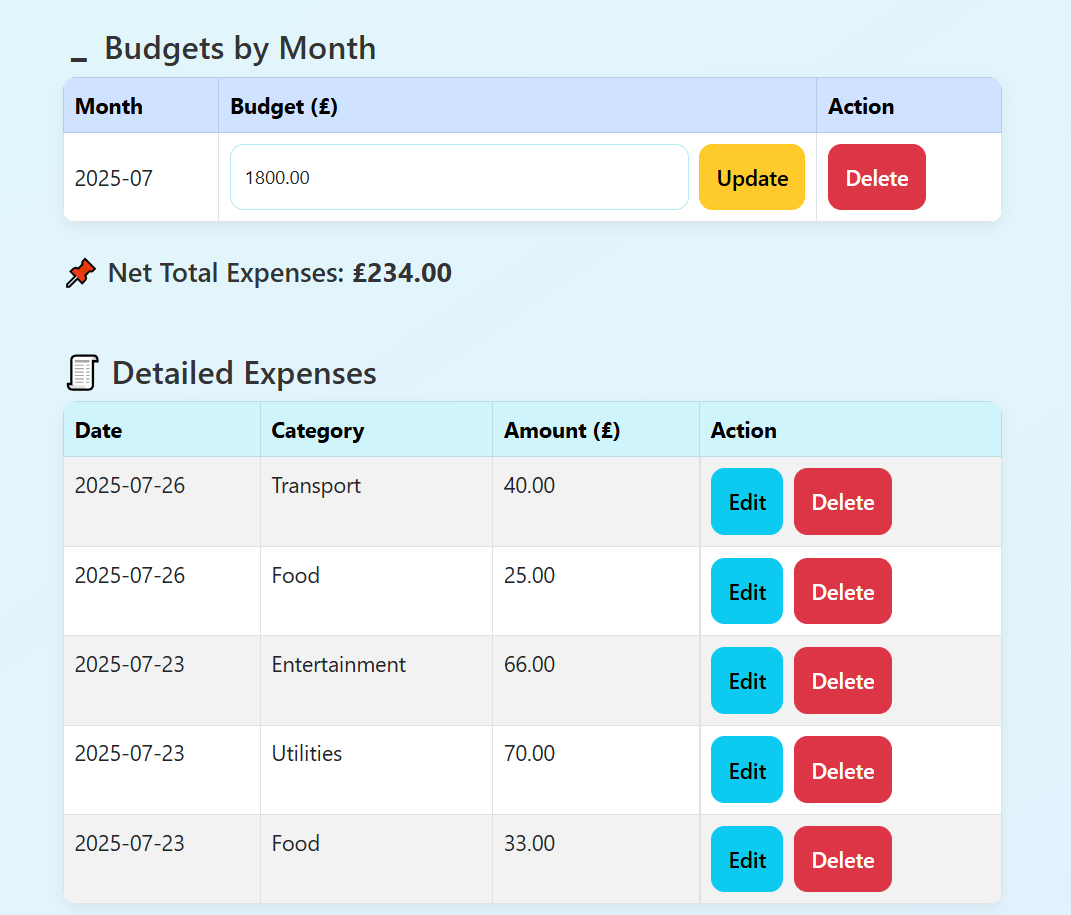
Deleting the first inserted expense



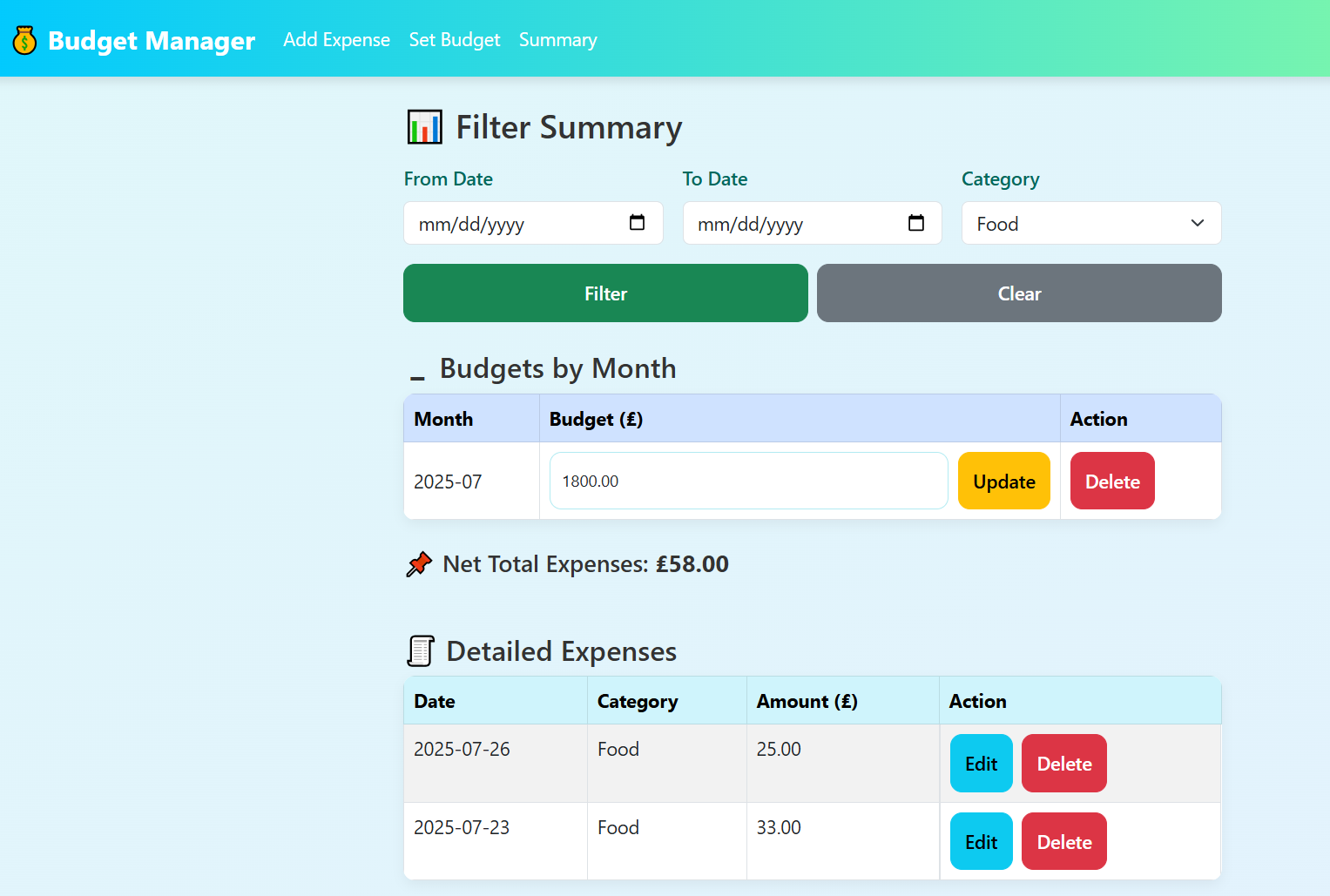


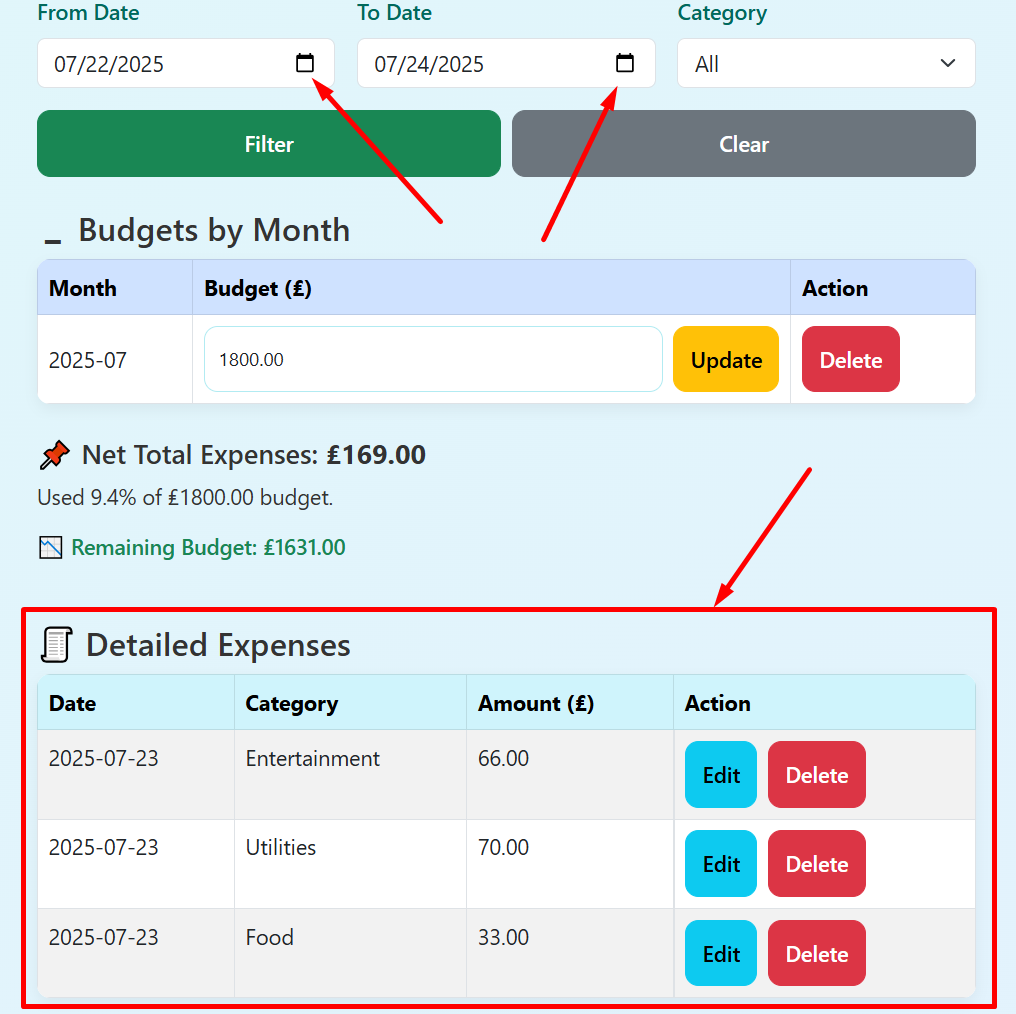


Inserting more expenses

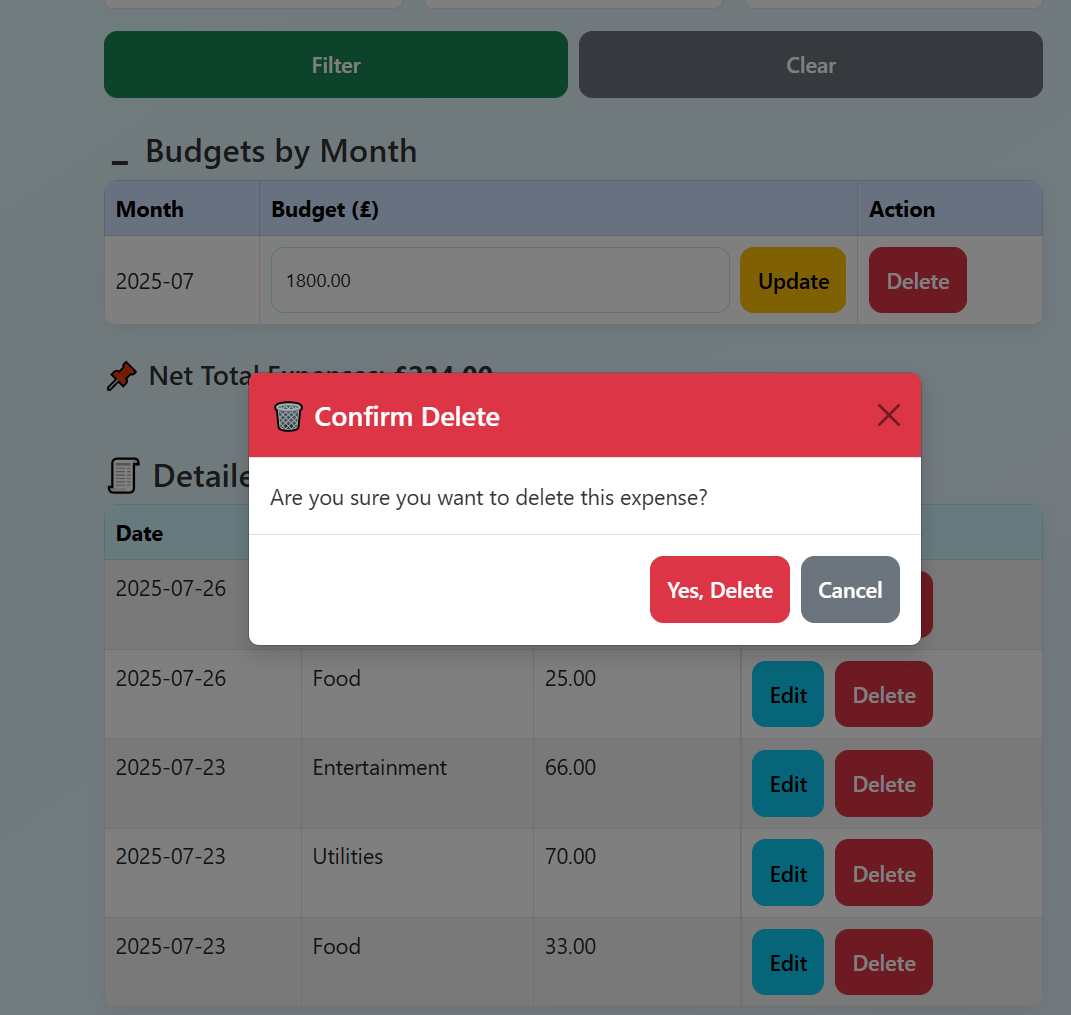


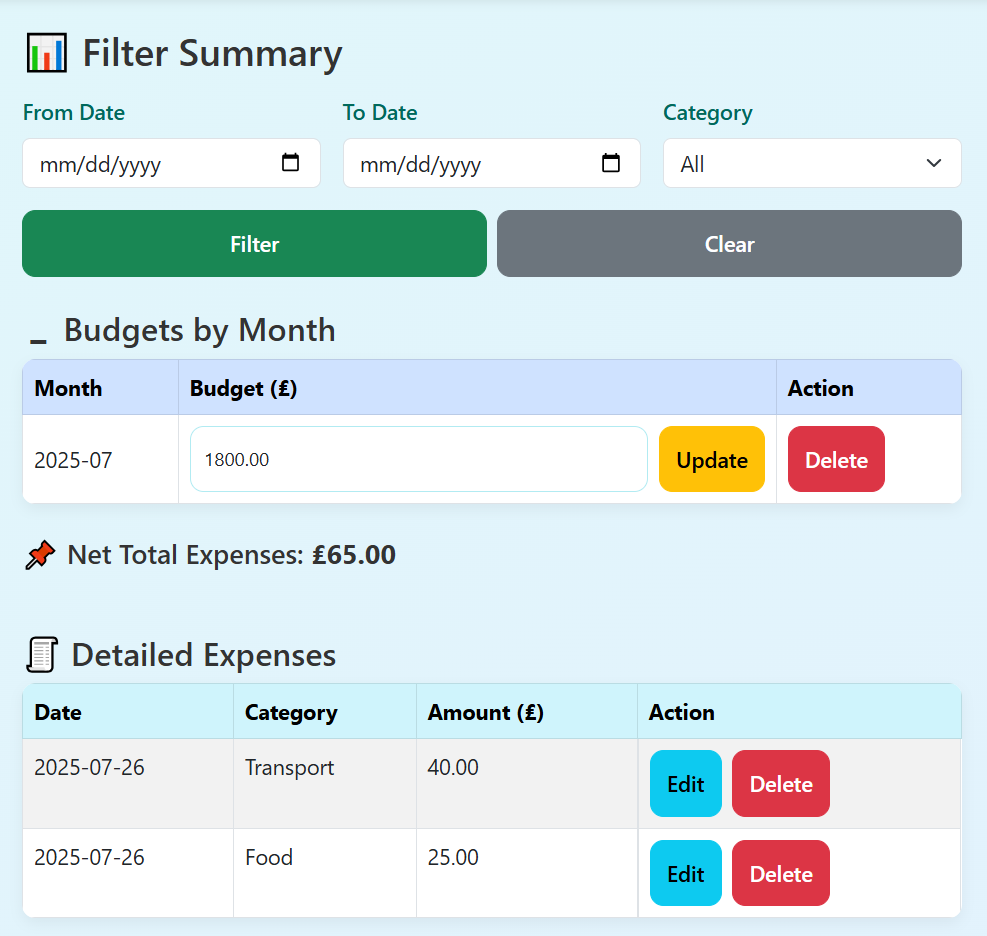
Testing filters

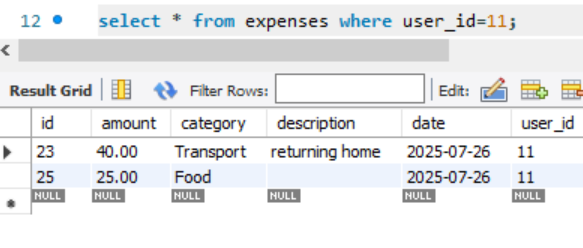




Deleting expense







Trying to insert another budget for the same month is not allowed since it is editable in the summary page

