CCAPDEV Term 2, AY 2021 – 2022

Mini Challenge 2: Expense Tracker

General Instructions

- Read the instructions first before proceeding!
- You may use the internet and your notes for reference, but you may not communicate with any other person for help.
- For any clarifications, ask your instructor.

Submission Details

Zip all your files and name it `MC2_LastnameFirstname.zip`. Submit the zipped folder to AnimoSpace.

SPECIFICATIONS

The webpage you will be recreating is a simple expense tracker app. For this mini challenge, you are already provided the HTML and the CSS files required. The layouts and styling are already completely defined, you only have to implement its functionality. Do not change anything in the HTML and CSS files, it is expected that you can complete the website through the JavaScript file alone!

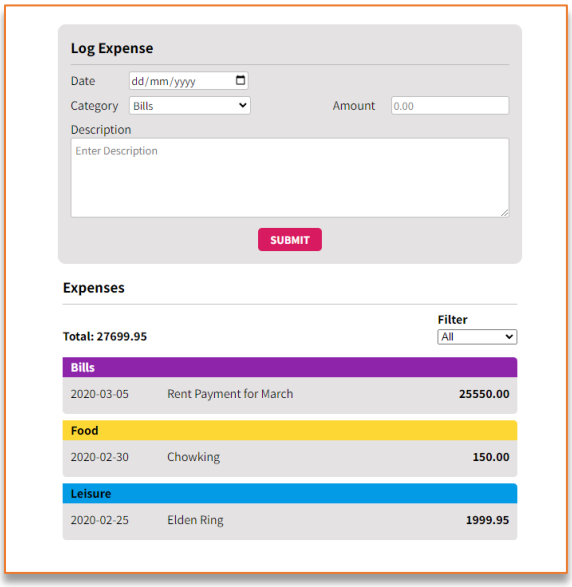

Figure 1. Expense Logger Webpage

The user is provided a form which they can log new expense entries, which would then be displayed on the expenses list below. The form requires the user to set a date, category, description, and the amount[1].

[1] **Hint:** For easy object creation, you can declare an object constructor: (https://www.w3schools.com/js/js_object_constructors.asp)

There is no limit to how many expenses can be added. After an expense is successfully added, **reset the form** back with the values set to its initial state (i.e. Bills for category, and blank for others). Additionally, the total amount (displayed in the expenses list) is also updated.

The items are of the class `financeItem`, which are then placed inside the `div` container with the class `itemsList`. You may refer to the provided html file for examples of the items and see how they are constructed.

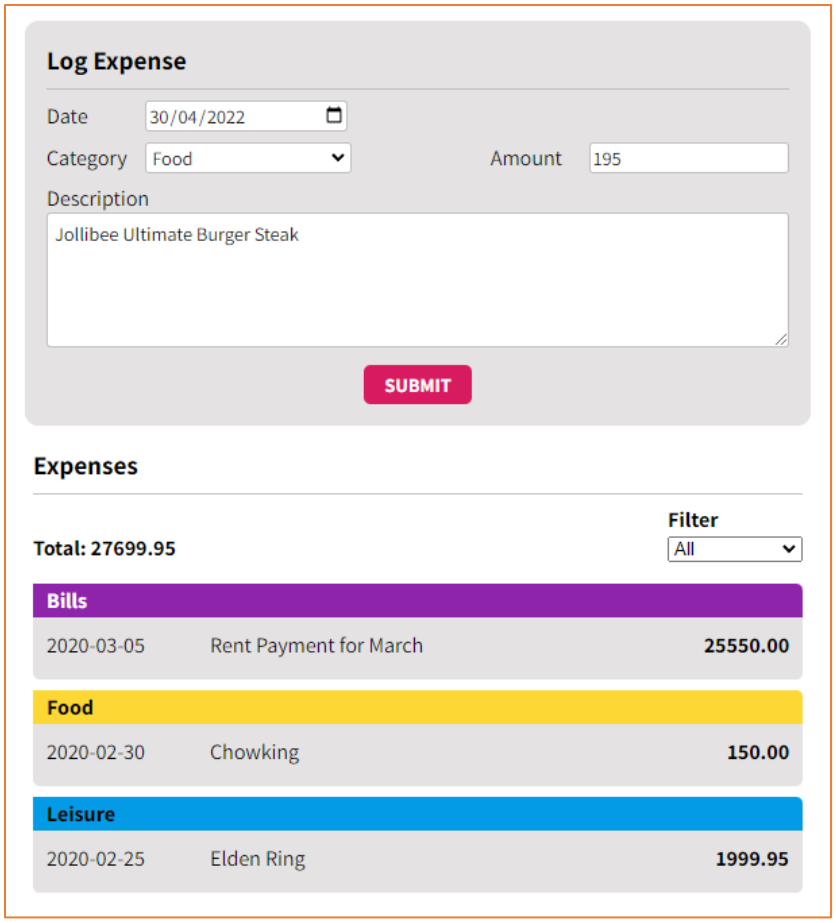

Figure 2. Filling in the Log Expense form

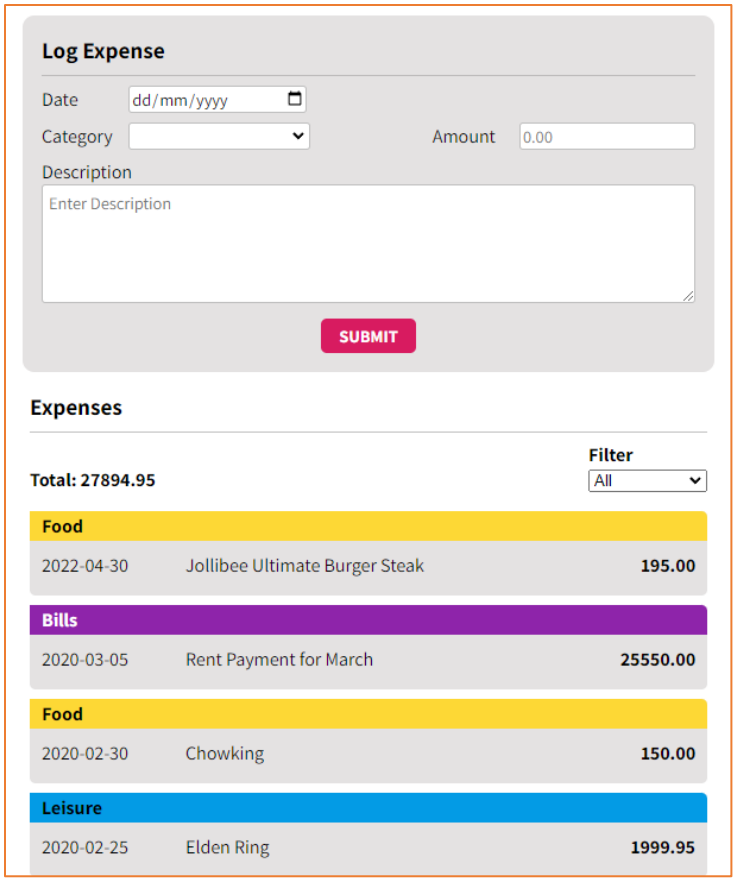

Figure 3. Logged expense submitted.

As you may have noticed from the screenshots, each expense item has its own assigned colors for its header, depending on the category. These categories are already styled in the stylesheet as the classes: `Food`, `Bills`, and `Leisure`, accordingly. You may refer to the three sample items provided in the `html` file as reference as to how these classes are assigned to the items.

Input Validation

In addition to expense logging, the form also includes a simple input validation feature. It only detects whether certain form entries are empty, and then displays the appropriate message. For reference, Table 1 below lists the error message for the given input error. Do not reset the form back if there is an error!

Table 1. ***Error messages for the given input error***

Error	Error Message
Date is unset	Date cannot be left unset.
Description is blank	Description cannot be left blank.
Amount is blank	Amount cannot be left blank.
Two or more errors occurred at the same time	Multiple input fields are blank/unset!

The error message is to be displayed inside the paragraph element with the id `errorText`.

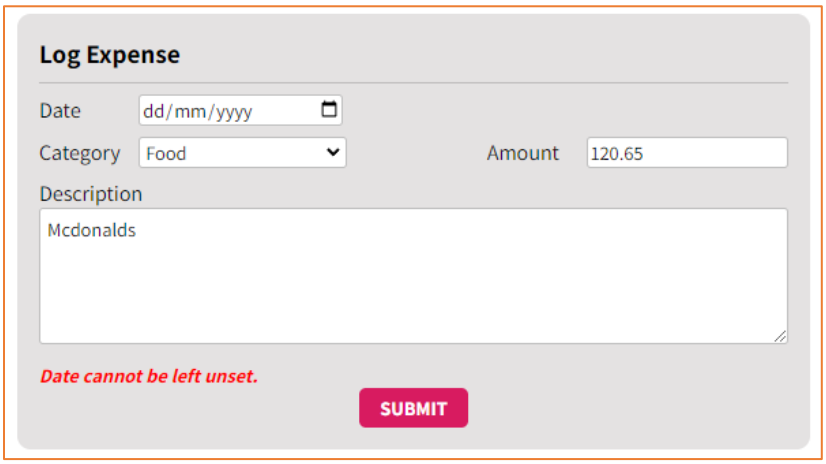

Figure 4. Unset date error

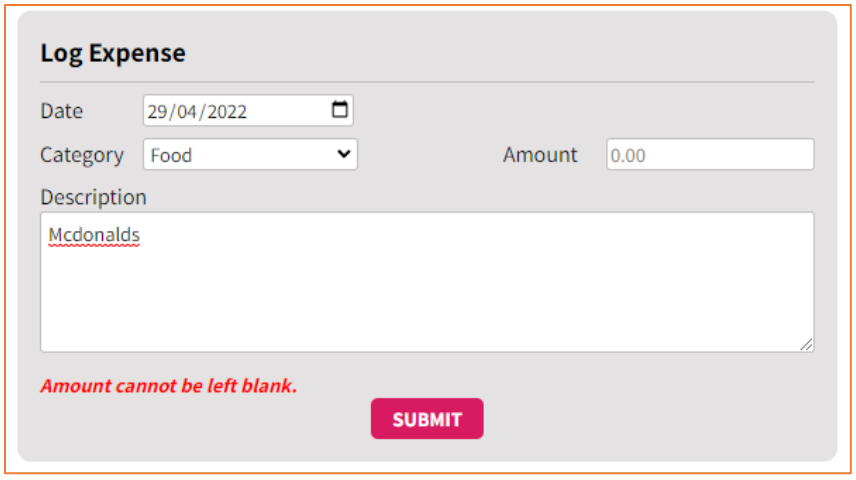

Figure 5. Amount left blank error

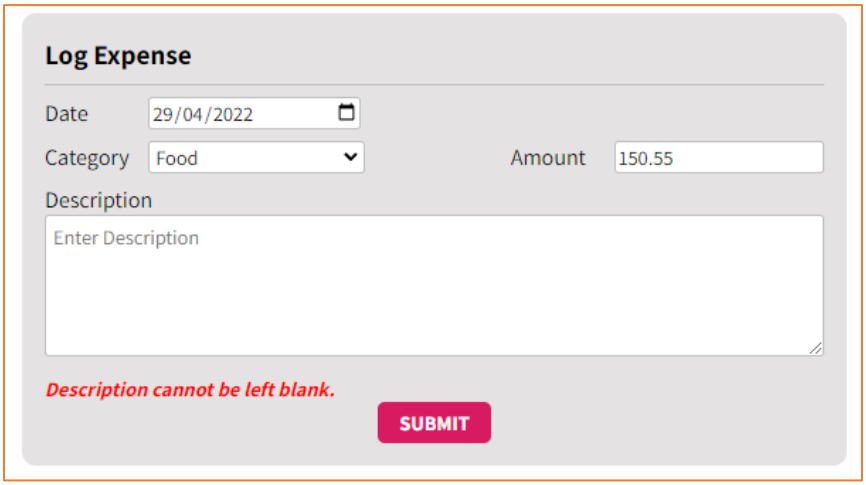

Figure 6. Description left blank error

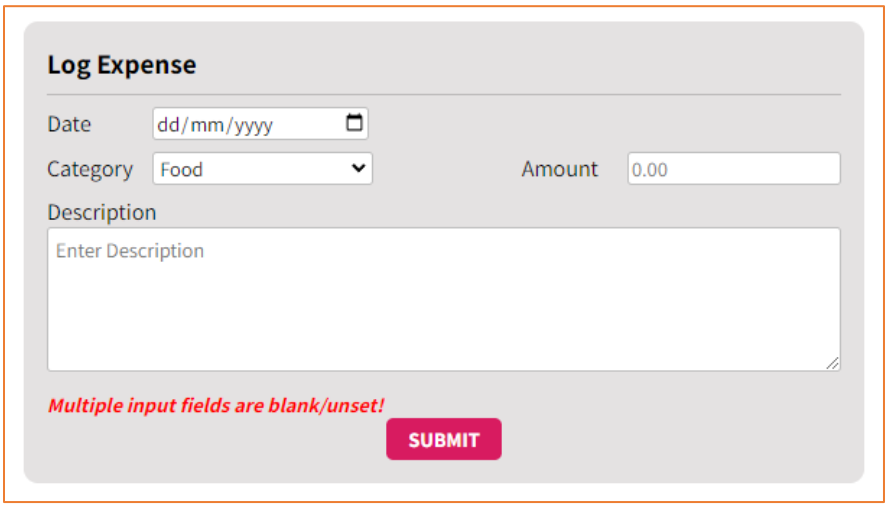

Figure 7. Multiple blank/unset input error

Bonus Feature: List Filtering

A filter dropdown is also provided. After the selection has changed, the list should only show the expenses under the selected category. The total should also reflect the changes done by the category filter.

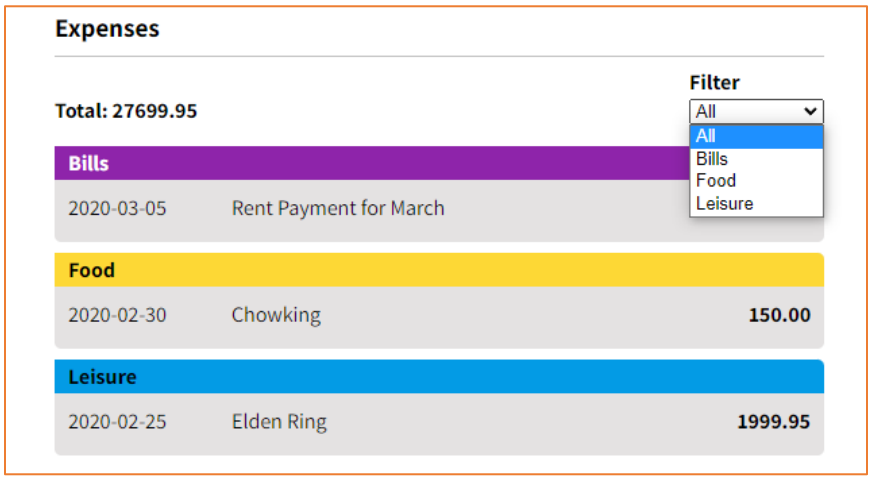

Figure 8. Filter dropdown options

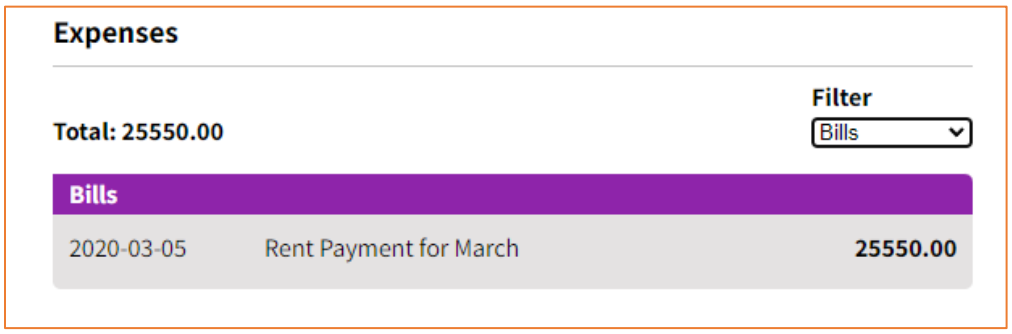

Figure 9. Filter - Bills selected

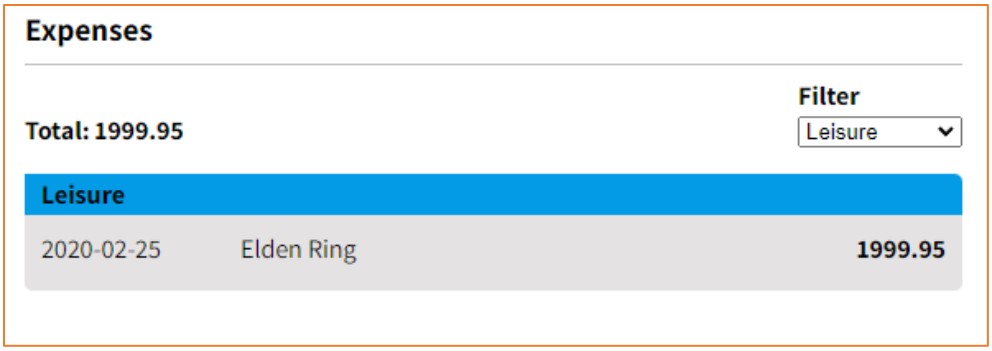

Figure 10. Filter - Leisure selected

RUBRIC FOR GRADING

Content	**Score**
Add an expense	5
Validation of expense entry	10
Reset form upon successful submit	5
Display expense item correctly in the list	15
Updating of the expense total amount	5
Total	**40**

Bonus Items:

You may only get the bonus items once you have implemented the required features.

Content	**Score**
Filter expense by category	7
Updating of the expense filtered total amount	3