**Technical Design Document Template**

**Name:** Alanys Suazo

**Date Created: 09/27/2025**

**Program Description:**

This program was created to calculate the cost of the user’s monthly expenses. The program will show the highest cost, the lowest cost, and the total cost of all of the monthly expenses the user inputs into the program.

**Functions used in the Program (list in order as they are called):**

1. **Function Name:** main

**Description:** calls the other functions, creates the dictionary of subscriptions and their values, and prints the results of the calculations

**Parameters:** none

**Variables:** name(the keys for the dictionary), amount(the value for the key entered), total(the total cost of all the monthly expenses), high\_name(the name of the highest expense), high\_value(the cost of the highest expense), low\_name(the name of the lowest expense), low\_value(the cost of the lowest expense), expenses(the name of the dictionary)

**Logical Steps:**

1. Creates a dictionary called expenses
2. Starts a loop for user input to append to the dictionary
3. Asks the user for the key and its value
4. Convert the cost into a float and store it in the dictionary
5. If input is invalid, show an error message
6. Once the user types ‘done’, break the loop
7. After the loop is broken, it sends the dictionary to other functions
8. Receives returned values and prints them along with the calculation of the sum of the expenses

**Returns:** none

2. **Function Name:** get\_highest

**Description:** finds the highest value in the dictionary

**Parameters:** expenses

**Variables:** highest\_value(takes the highest value that appears in the dictionary), name(the key in the dictionary), amount(the value in the dictionary)

**Logical Steps:**

1. Gets dictionary from the main function
2. Finds the number of the highest value in the dictionary and assigns it
3. Using that value, finds the key-value pair and returns it

**Returns:** name, amount

3.  **Function Name:** get\_lowest

**Description:** finds the lowest value in the dictionary

**Parameters:** expenses

**Variables:** lowest\_value(takes the lowest value that appears in the dictionary)

**Logical Steps:**

1. Gets the dictionary from the main function
2. Finds the number of the lowest value in the dictionary and assigns it
3. Using that value, it finds the key-value pair and returns it.

**Returns:** name, amount

**Logical Steps:**

1. The main function is called.
2. An empty dictionary named expenses is created
3. The user is in the loop phase and is prompted to enter the name of the subscription or monthly expense.
4. User is then prompted to enter the value of the subscription, after which the pair of key-value is appended to the dictionary as long as the value is numerical.
5. Steps 3 and 4 are repeated until the user enters ‘done’, after which does the loop break.
6. The dictionary and its data are then sent to the appropriate functions(get\_highest, get\_lowest) to carry out the needed calculations
7. The needed key-value pairs are returned
   1. The highest expense comes from the key-value pair returned from get\_highest
   2. The lowest expense comes from the key-value pair returned from get\_lowest
8. The total, which was calculated using the sum() feature of Python, is printed along with the returned key-value pairs. The user sees the item that they spend the most on, the item they spend the least on, and the total amount of their expenses together.

**Link to your repository:** [Alanys-SG/COP2373 at master](https://github.com/Alanys-SG/COP2373/tree/master)

**Output Screenshot: A screenshot of a computer

AI-generated content may be incorrect.**