```
import csv

expense_list = []  # to store all the expenses from user and csv file as a list of dictionary.

# try/except block to check whether expense.csv file exists or not.
try:
  with open(’expense.csv’, mode=’r’, newline=’’) as file:
    reader = csv.DictReader(file)
    for row in reader:
        expense_list.append(dict(row))

except FileNotFoundError:
  print("File not found as this is the first run")

# add_expense method to add user expenses
def add_expense():

  date_of_expense = input("Enter the date of expense in YYYY-MM-DD format: ")
  category = input("Enter the category of the expense: ")
  amount_spent = float(input("Enter the amount of the expense: "))
  description = input("Enter the description of the expense: ")

  expense_dict = {}

  expense_dict["date"] = date_of_expense
  expense_dict["category"] = category
  expense_dict["amount"] = amount_spent
  expense_dict["description"] = description

  expense_list.append(expense_dict)

# view_expense method shows the expense for the user for a month and discards entries where any one
field is blank
def view_expense():
  for expense in expense_list:
    if(expense["category"] == ’’ or expense["date"] == ’’
       or expense["amount"] == ’’ or expense["description"] == ’’):
      print(f"Expense entry is invalid as some fields are blank/empty: {expense[’description’]}")
    else:
      print(expense)

# track_budget method asks user to enter the monthly budget and provides insights whether user
exceeded the budget or not
def track_budget():
  monthly_budget = float(input("Enter the monthly budget: "))
  total_expense = total_expenses()
  if total_expense > monthly_budget:
    print("You have exceeded your monthly budget")
  else:
    print(f"Remaining budget: {monthly_budget - total_expense}")

# total_expenses is a helper method to calculate total expenditure done so far by the user.
def total_expenses():
  total_expense = 0
  for expense in expense_list:
    total_expense = total_expense + float(expense["amount"])
  return total_expense

# save_expense method saves the expense from the list to a csv file.
# Ensure to call it before exiting the application, otherwise newly added expenses might be lost.
def save_expense():
  if(expense_list == []):
    print("No expenses to save")
    return
  with open(’expense.csv’, mode=’w’, newline=’’) as file:
    writer = csv.DictWriter(file, fieldnames=expense_list[0].keys())
    writer.writeheader()
    writer.writerows(expense_list)

# Interactive Menu to ask for user choices.
while True:
  print("==============")
  print("1. Add Expense")
  print("2. View Expense")
  print("3. Track Budget")
  print("4. Save Expense")
  print("5. Exit")
  print("==============")

  user_choice = int(input("Enter your choice: "))
  if user_choice == 1:
    add_expense()
  elif user_choice == 2:
    view_expense()
  elif user_choice == 3:
    track_budget()
  elif user_choice == 4:
    save_expense()
  elif user_choice == 5:
    print("Exiting the application....")
    break
  else:
    print("Invalid choice")
```