```python
import csv
import os
from datetime import datetime

# Configuration
DATA_FILE = "spending_history.csv"

def clear_terminal():
    os.system('cls' if os.name == 'nt' else 'clear')

def save_transaction(date, item, category, amount, remaining):
    file_exists = os.path.isfile(DATA_FILE)
    with open(DATA_FILE, mode='a', newline='') as f:
        writer = csv.writer(f)
        if not file_exists:
            writer.writerow(["Date", "Item", "Category", "Amount", "Balance_Left"])
        writer.writerow([date, item, category, amount, round(remaining, 2)])

def main():
    clear_terminal()
    print("========================================")
    print("    🛡️  DAILY SPEND GUARDIAN: GOOGLE MVP   ")
    print("========================================")

    try:
        income = float(input("💰 Enter Monthly Income (₹): "))
        emi = float(input("💳 Enter Monthly EMI (₹): "))
    except ValueError:
        print("❌ Error: Please enter numbers only.")
        return

    # Core Logic
    base_limit = (income - emi) / 30
    current_balance = base_limit
    today = datetime.now().strftime("%Y-%m-%d")

    print(f"\n✅ Calculation Complete!")
```

```
print(f"\n✅ Calculation Complete!")
print(f"Daily Spending Power: ₹{round(base_limit, 2)}")
print("-" * 42)

while True:
    print(f"\nRemaining for Today: ₹{round(current_balance, 2)}")
    item = input("What did you buy? (or 'q' to submit/quit): ")

    if item.lower() == 'q':
        break

    try:
        print("Categories: [1] Food [2] Travel [3] Shopping [4] Other")
        cat_choice = input("Select Category (1-4): ")
        categories = {"1": "Food", "2": "Travel", "3": "Shopping", "4": "Other"}
        cat_name = categories.get(cat_choice, "Misc")

        amount = float(input(f"Amount for {item}: ₹"))
        current_balance -= amount

        # Real-time Feedback Logic
        if current_balance < 0:
            print(f"🚨 ALERT: You've exceeded the limit by ₹{abs(current_balance)}!")
            recovery = abs(current_balance) / 29
            print(f"💡 TO RECOVER: Reduce spending by ₹{round(recovery, 2)} daily.")
        elif current_balance < (base_limit * 0.2):
            print("⚠️ CAUTION: Only 20% of your daily budget remains.")
        else:
            print("✨ Transaction logged successfully.")

        # Save to CSV
        save_transaction(today, item, cat_name, amount, current_balance)
```

```
        except ValueError:
            print("❌ Invalid amount. Transaction skipped.")

    print("\n==========================================")
    print(f"📊 DATA SYNCED TO: {DATA_FILE}")
    print("Ready for Google Sheets Import. Good luck!")
    print("==========================================")

if __name__ == "__main__":
    main()
```