Course: HCI 584

Student: Ahmed ALmathami

Instructor: Dr. Chris Harding

Date: 09/14/2025

Title: Expense Tracker – Program Specification

- **General Description:**

The project is a basic Expense Tracker, which is a desktop Python project. The primary objective is to assist a user in capturing his or her. day to day costs, save them in a file and produce simple reports. The program will start with a command-line interface (CLI) but can eventually grow to a less complicated Streamlit GUI to interact with.

What it does: The user is free to input an expense (date, amount, category, description). All entries are saved into expenses.csv. - Data management: The application loads current information in the CSV on launch and rewrites the file with the addition of new expenses.

Reports: The user will be able to see all expenses on a table, total by category, and visualize a line graph of every spending within the past 30 days.

Packages/Tools Python standard libraries, pandas (CSV processing), matplotlib/plotly (charts), Streamlit for later.

- **Task Vignettes (User Activity Flow):**

Vignette 1: Entering a New Expense, where the user opens the program, then they are prompted to enter details about an expense (date, amount, category, description). The program saves this entry into a file named (expenses.csv) .

Technical: Input fields → Date (string), Amount (float), category (string), Description (string). Save with pandas to_csv().

Vignette 2: Viewing All Expenses. The user selects “View Expenses”. Then, the program loads the CSV file and shows all expenses in a table format.

Technical: Load data with pandas read_csv(). Display with print (CLI) in v1, Streamlit table later.

Vignette 3: Reports and charts the user selects “Reports.” The program displays totals by category and a line graph of spending trends over 30 days.

Technical: Use pandas groupby() for totals. matplotlib/plotly for line graphs. CLI totals in v1, charts in GUI later.

- **Technical Flow:**

Start: Check if expenses.csv exists or not , if it is not exist , I should create it with columns (Date, Amount, Category, Description).

Add expense: Append new row (record) to CSV immediately.

Load data: pandas read_csv() at startup.

Reports: groupby() for summaries, matplotlib/plotly for charts. Data types: Date=string (later datetime), Amount=float, Category/Description=strings.

User interaction: Version 1 --> CLI, Version 2 --> Streamlit GUI.

- **Final Self-Assessment:**

Sketch change: Reporting additions (totals and charts). - Confidence: I will be able

applying CLI version using pandas/matplotlib.

Hardest task: Management of dates (30-day reports), and adding Streamlit GUI.

Assistance required: Advice on Streamlit and datetime processing.

- **Deliverables:**

app.py (main script in project root) - expenses.csv (data file in /data folder) - docs/spec.pdf (program specification) - image/diagram.png (optional flow diagram or mockups) - README.md (summarize my program description).