
USER MANUAL

For

Expense Tracker

Prepared By: -

*Abhishek Dutta
Department of Software Engineering
Arizona State University*

Academic Year: 2016 - 2017

Version: 1.0

Preface

This document is intended for users who wants to build and use this application for their own use. The goal of this document is to give step by step instructions for the user to build this application from the scratch and how to use it, highlighting the capabilities and functionalities of this application.

Revision History

Date	Revision	Description	Author
11/26/16	1.0	Initial Version	Abhishek Dutta

Table of Contents

Preface	xi
Revision History	xii
Chapter 1 Introduction	1
1.1 Purpose	1
1.2 Scope	1
1.3 Software Requirements	1
1.4 Hardware Requirements	1
1.5 Definitions, Acronyms, and Abbreviations	2
1.6 References	2
Chapter 2 Software Installation	2
2.1 Steps to build the application from build.xml.	2
2.2 Steps to launch application	3
Chapter 3 Process Workflow	3
3.1 Add Expense	3
3.2 Delete Expense	4
3.3 Update Expense	5
3.4 Display Expense	5
3.5 Add a new member	5
3.6 Delete member	6
3.7 Display Members	6

1. Introduction

1.1 Purpose

The purpose of this document is to guide the users of this application on how to use this application effectively and to the best of their use. Assuming that the user has little knowledge on setting paths and class paths in a computer, user should be able to easily set up the application in their personal computers and install it.

1.2 Scope

Expense Tracker basically is maintaining the user expense information with details as such category, date of entry and the amount spent on the category. A user can go back and update an existing entry or delete an existing entry. Users can also add/delete different members or check other user's expenses. The application takes input using command line console and accepts only a valid entry for an existing user.

The project is mainly designed for the use of students, who want to track their monthly expenses or a family person who wants to keep track of the expenses made by the family members. It can be also useful to a small group of people or a small organization/association who wants to track their expenses made from a common fund. The user interface is a simple command line interface and all the options are clear to understand for a person. The programming language used for building this application is Java and data is stored using Json.

1.3 Software Requirements

The below mentioned software are needed to be installed in the system for running this application:

- Java Development Toolkit JDK 1.8+
- Apache Ant 1.8
- JSON 1.1

1.4 Hardware Requirements

A standard computer system should be able run this application easily. Any system with below minimum configurations are sufficient to host it.

- 2 GB RAM
- X86 or x64 bit Windows/Linux Operating system
- 50 GB Hard drive (Depends on usage to store the data)
- 1.6 GHZ Intel CPU

1.5 Definitions, Acronyms, and Abbreviations

ANT	Java library.
IDE	Integrated Development Environment
JAVA	Platform Independent Programming Language
JSON	JavaScript Object Notation
JRE	Java Runtime Environment
SRS	Software Requirements Specification

Table 1. Acronyms and Terminology

1.6 References

➤ Books

- Software Engineering: Ninth Edition by Ian Sommerville
- The Personal Software Process(PSPSM), November 2000 by Watts S. Humphrey

2. Software Installation

In this section, steps for building the application is described. Please follow the instructions to build the application from scratch.

2.1 Steps to build the application from build.xml script using Apache Ant.

Assuming your JAVA_HOME and ANT_HOME, and JSON library in the lib folder.

1. Open your command prompt and navigate to the Project Folder directory “Expense Tracker”.
2. Run “ant”.

```
D:\Workspace_Eclipse\Expense Tracker>ant
Buildfile: D:\Workspace_Eclipse\Expense Tracker\build.xml

targets:

clean:
  [delete] Deleting directory D:\Workspace_Eclipse\Expense Tracker\output

prepare:
  [mkdir] Created dir: D:\Workspace_Eclipse\Expense Tracker\output

build:
  [javac] Compiling 6 source files to D:\Workspace_Eclipse\Expense Tracker\output
  [javac] Note: Some input files use unchecked or unsafe operations.
  [javac] Note: Recompile with -Xlint:unchecked for details.

execute:
  [jar] Building jar: D:\Workspace_Eclipse\Expense Tracker\ExecutableJar\ExpenseTrackerAbhishekDutta.jar

BUILD SUCCESSFUL
Total time: 4 seconds
```

The executable jar file should be created under:

“Expense Tracker/ExecutableJar/ExpenseTrackerAbhishekDutta.jar”.

2.2 Steps to launch the application.

Assuming your JAVA_HOME is set,

1. Navigate to “Expense Tracker”/ExecutableJar/ExpenseTrackerAbhishekDutta.jar”.
2. Run “java -jar ExpenseTrackerAbhishekDutta.jar”. You should see a window like below:

```
D:\Workspace_Eclipse\Expense Tracker\ExecutableJar>java -jar ExpenseTrackerAbhishekDutta.jar

=====> EXPENSE TRACKER <=====

a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
```

3. Process Workflow

Please follow the step by step instructions one by one to understand how the application works and fulfills user requirements. This section is concentrated towards helping the user on how to effectively use this application.

3.1 Add an Expense for a user.

1. Enter “1” in the choice.
2. Enter “Name” of the member for whom the expense needs to be added.
3. Enter the category.
4. Enter the expense amount.

You should see the output as below.

```
=====> EXPENSE TRACKER <=====

a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
1
Name of the Member to add expense:
Tom
Category:
misc
Amount:
500

Expense for Tom in misc added Successfully!!

All transactions for member Tom

Date      | Category | Amount
-----|-----|-----
20161129  | food     | 700.0
20161130  | medicine | 400.0
20161202  | misc     | 500.0
```

3.2 Delete an Expense for a user.

1. Enter “2” in the choice.
2. Enter “Name” of the member for whom the expense needs to be deleted.
3. Enter the category.
4. Enter the date of the entry in “yyyymmdd”.

You should see the output as below.

```
===== > EXPENSE TRACKER <=====
a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
2

Name of the Member whose expense to be deleted:
Tom

Category to be deleted:
misc

Enter the date of entry in the format yyyymmdd:
20161202

Expense misc for Tom, deleted successfully!!

All transactions for member Tom

Date      | Category | Amount
-----
20161129 | food    | 700.0
20161130 | medicine | 400.0
```

3.3 Update an Expense for a user.

1. Enter “3” in the choice.
2. Enter “Name” of the member for whom the expense needs to be updated.
3. Enter the category of the expense.
4. Enter the date of the entry in “yyyymmdd”.
5. Enter the new amount.

You should see the output as below.

```
===== > EXPENSE TRACKER <=====
a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
3

Name of the Member whose expense needs to be updated:
Tom

Category to be Updated:
food

Enter the date of entry in the format yyyymmdd:
20161129

Enter the new Amount:
300

Expensefood Updated Successfully!!

All transactions for member Tom

Date      | Category | Amount
-----
20161129 | food    | 300.0
20161130 | medicine | 400.0
```

3.4 Display expenses for a user.

1. Enter "4" in the choice.
2. Enter "Name" of the member for whom the expense needs to be displayed.

You should see the output as below.

```

===== > EXPENSE TRACKER <=====
a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
4

Name of the Member whose balance to be shown:
Tom

All transactions for member Tom

Date      | Category | Amount
-----
20161129 | food    | 300.0
20161130 | medicine| 400.0

```

3.5 Add a new member

1. Enter "5" in the choice.
2. Enter "Name" of the member to be added.

You should see the output as below.

```

===== > EXPENSE TRACKER <=====
a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
5

Name of the Member to be Added:
Rich

Rich added Successfully!!

All Members :-
=====
1)abhi
2)Abhishek
3)Apoorv
4)Rich
5)Tom

```

3.6 Delete a member

1. Enter "6" in the choice.
2. Enter "Name" of the member to be added.

You should see the output as below

```
===== > EXPENSE TRACKER <=====
a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
6
Enter the name of the Member to remove:
Rich

User Rich successfully deleted!!

All Members :-
=====
1)abhi
2)Abhishek
3)Apoorv
4)Tom
```

3.7 Display members

1. Enter "7" in the choice.

You should see the output as below.

```
===== > EXPENSE TRACKER <=====
a) Add Expense -- 1
b) Delete Expense -- 2
c) Update Expenses -- 3
d) Display Expenses -- 4
e) Add a Member -- 5
f) Remove a Member -- 6
g) Display the list of Members -- 7

Please enter your Choice:
7

All Members :-
=====
1)abhi
2)Abhishek
3)Apoorv
4)Tom
```