

# CProject1 Documentation

## Statement

Create an application that allows the user to record and manage simple financial data, such as transactions, income, and expenses.

- **User Interface:** Implement a simple user interface that enables adding and viewing transactions. You can use input and output functions to interact with the user. This can be a command-line program or a graphical user interface (recommended).
- **Data Structures for Transactions:** Define data structures to store information about transactions, such as date, description, amount, and type (income or expense).
- **Basic Accounting Features:** Implement functions that allow the user to record new transactions, calculate the account balance, and display a list of previous transactions.
- **Simple Financial Reporting:** Implement a function that generates a simple financial report, such as a summary of income and expenses for a specific period.
- **Data Manipulation in Files:** Allow the user to save and load financial data in files to preserve it between sessions.
- **Input Data Validation:** Ensure that user-entered data is properly validated to prevent the input of incorrect data.
- **Documentation and Comments:** Document the code and add comments to make it easier to understand.
- **Testing:** Test the project to ensure it functions correctly and that financial calculations are accurate.
- **Error Handling:** Handle errors and exceptions properly to prevent application crashes due to incorrect input.
- **Compliance with Specific Project Requirements:** Ensure that the project meets the specific requirements set by the instructor.
- **Presentation and Explanations:** Ensure that you can explain and present the project to the instructor or peers, highlighting key features and design decisions.

## Features

Index	Feature
1	Record a transaction.
2	Display all transactions.
3	Display account balance.
4	Display the income and expenses made in a certain period
5	Save the transactions in a file.
6	Load the transactions from a file.

## Runtime scenarios

### Recording a transaction

Index	Input	Output	Explanation
1	-	What option do you want to choose?	The application asks the user to choose an option.
2	1	-	The user has chosen to record a new transaction.
3	-	Input the date on which was the expense made. It must have this format: mm/dd/yyyy	The application asks the user to input the date of the transaction in a particular format.
4	3/19/2004	-	The input is received by the application.
5	-	Input a description for the transaction.	The application asks the user for the description of the transaction.
6	This was an important transaction.	-	The input is received by the application.
7	-	What is the amount this transaction?	The application asks the user for the value of the transaction.
8	100	-	The input is received by the application.
9	-	What was the type of the transaction? (Income or Expense)	The user is asked to input the type of the expense.
10	2	-	The input is received by the application.
11	-	The transaction was recorded successfully!	The user is notified that the transactions was recorded.

### Displaying all transactions

Index	Input	Output	Explanation
1	-	What option do you want to choose?	The application asks the user to choose an option.
2	2	-	The user has chosen to display all recorded transactions.
3	-	A table.	The applications displays a table with all of the recorded transactions. It has the following format: Index, Date, Description, Value and Type.

### Displaying the account balance

Index	Input	Output	Explanation
1	-	What option do you want to choose?	The application asks the user to choose an option.
2	3	-	The user has chosen to display their account balance.
3	-	Your account balance is: <accountBalance>	The application displays the user's account balance.

### Displaying all expenses and incomes

Index	Input	Output	Explanation
1	-	What option do you want to choose?	The application asks the user to choose an option.
2	4	-	The user has chosen to display all expenses and incomes made during a certain period of time.
3	-	Input the starting date of the period in which your transactions were made. It must have the following format: mm/dd/yyyy.	The user is asked to input the starting date of the interval in which the transactions were made.
4	10/2/2000	-	The input is received by the application.
5	-	Input the ending date of the period in which your transactions were made. It must have the following format: mm/dd/yyyy.	The user is asked to input the ending date of the interval in which the transactions were made.
6	12/1/2000	-	The input is received by the application.
7	-	Expenses: A table.	The application displays a table with the expenses that were made in the input interval. The table has the following format: Index, Date, Description, Value and Type.
8	-	Incomes: A table.	The application displays a table with the incomes received in the input interval. The table has the following format: Index, Date, Description, Value and Type.

### **Saving the transactions in a file**

<b>Index</b>	<b>Input</b>	<b>Output</b>	<b>Explanation</b>
1	-	What option do you want to choose?	The application asks the user to choose an option.
2	5	-	The user has chosen to save the expenses in a file.
3	-	The transactions were saved successfully!	The transactions were save in a file called: "transactions.txt".

### **Loading the transactions from a file**

<b>Index</b>	<b>Input</b>	<b>Output</b>	<b>Explanation</b>
1	-	What option do you want to choose?	The application asks the user to choose an option.
2	6	-	The user has chosen to load the expenses from a file.
3	-	The transactions were loaded successfully!	The transactions were loaded from a file called: "transactions.txt".