**INTRODUCTION**

**ARCHITECTURE**

**FEATURES**

CATEGORY MANAGEMENT

[C] User should be able to define categories for expenses

[C] Users should be able to use predefined categories like groceries, utilities, entertainment

[R] User should be able to view categories available

EXPENSE MANAGEMENT

[C] User should be able to add an expense by specifying details such as the amount, date, category, and description

[U] Users should be able to modify an expense

[D] User should be able to delete an expense

[R] User should be able to view expenses as a list of all recorded expenses in a clear and organized format

[R] User should be able to filter and view expenses for specific categories

[R] User should be able to generate reports showing total expenses for specific periods (daily, weekly, monthly) and per category

FILE HANDLING

Implement file handling for saving expenses to a CSV file and loading them upon starting the application

ERROR HANDLING – FILE OPERATION HANDLING

Handle errors related to file reading and writing operations, ensuring that invalid file paths or formats don’t crash the application

ERROR HANDLING - INPUT VALIDATION

Implement robust error handling for user input errors (e.g., invalid dates, negative  
amounts and category). Ensure that invalid inputs do not crash the application and that users receive appropriate error messages

**CORE CLASSES:**

Expense: handles the create, update and delete functionality

Category:

ExpenseTracker:

MAIN FUNCTIONS

Add expenses

Delete Expenses

Modify Expenses

View Expenses

**TESTING**

**CHALLENGES**

**FUTURE ENHANCEMENTS**