Data: Name: Macarie Cristian

# Problem statement:

5. APARTMENT BUILDING ADMINISTRATOR

Jane is the administrator of an apartment building and she wants to manage the monthly expenses for each apartment. Jane needs an application to store, for a given month, the expenses for each apartment. Each expense is stored in the application using the following elements: apartment (number of apartment, positive integer), amount (positive integer), type (from one of the predefined categories: water, heating, electricity, gas, other). The applications shall provide the functionalities to help Jane:

# Apartment Admin - Feature list

|  |
| --- |
| F1. Add a new transaction to the list. |
| F2. Modify expenses from the list. |
| F3. Write the expenses having different properties |
| F4. Obtain different characteristics of the expenses. |
| F5. Filter expenses. |
| F6. Undo the last operation that modified program data. |

# Iteration plan

|  |  |
| --- | --- |
| **Iteration** | **Planned features** |
| I1 | 1. Add a new transaction to the list. |
| I2 | 2. Modify expenses from the list. |
|  | 2a.Remove expenses from apartment. |
|  | 2b.Remove expenses between two apartments. |
|  | 2c.Remove all expenses for a type. |
|  | 2d.Replace the value of an apartment expense. |
| I3 | 3. Write the expenses having different properties. |
|  | 3a.Write the entire list of expenses. |
|  | 3b.Write all expenses for an apartment. |
|  | 3c.Write all the apartments whose total expenses are </=/> then a given value. |
| I4 | 4. Obtain different characteristics of the expenses. |
|  | 4a. Write the total amount for the expenses having the given type. |
|  | 4b. Write the maximum amount per each expense type for the given apartment. |
|  | 4c. Write the list of apartments sorted ascending by total amount of expenses. |
|  | 4d.Write the total amount of expenses for each type, sorted ascending by amount of money. |
| I5 | 5. Filter expenses. |
|  | 5a. Keep only expenses for the type given. |
|  | 5b. Keep only expenses having an amount of money smaller than a given value. |
| I6 | 6. Undo the last operation that modified program data. |

# Iteration 1

## Running scenario

|  |  |  |
| --- | --- | --- |
| User | Program | Description |
|  | Print the list. | The current list |
|  | Print the Main Menu | The menu with the commands for F1,F2,F3,F4,F5,F6 |
|  | Ask for the command: | Prepare to get a command |
| 1 |  | Give the command for F1 |
|  | Ask for the ap. num | Prepare to get the ap num |
| 1 |  | Give the ap num |
|  | Ask for the expense type | Prepare to get the expense type |
| gas |  | Give the expense type |
|  | Ask for the expense value | Prepare to get the expense value |
| 200 |  | Give the expense value |
|  | [[1,”gas”,200]] | The current list |

# Iteration 2

## Running scenario

|  |  |  |
| --- | --- | --- |
| User | Program | Description |
|  | Ask for the command: | Prepare to get a command |
| 2 |  | Give the command for F2 |
|  | Print the menu for F2 | The menu commands for 2a,2b,2c,2d |
|  | Ask for the command: | Prepare to get a command |
| 2a |  | Give the command for 2a |
|  | Ask for the ap. num | Prepare to get the ap num |
| 1 |  | Give the ap num |
|  | [[1,”gas”,20],  [1,”water”,10]]  🡪[ ] | Print the modyfied list |
|  | Print the menu for F2 | The menu commands for 2a,2b,2c,2d |
|  | Ask for the command: | Prepare to get a command |
| 2b |  | Give the command for 2b |
|  | Ask for the first ap. num | Prepare to get the ap num |
| 1 |  | Give the first ap num |
|  | Ask for the secound ap. num | Prepare to get the ap num |
| 3 |  | Give the secound ap num |
|  | [[1,”gas”,20],  [1,”water”,10],  [3,”heating”,11],[4,”electricity”,21]]🡪[[4,”electricity”,21]] | Print the modyfied list |
|  | Print the menu for F2 | The menu commands for 2a,2b,2c,2d |
|  | Ask for the command: | Prepare to get a command |
| 2c |  | Give the command for 2c |
|  | Ask for the expense type | Prepare to get the expense type |
| water |  | Give the expense type |
|  | [[1,”water”,10],[2,”water”12],  [3,”heating”,11],[4,”electricity”,21]]  🡪[[3,”heating”,11],[4,”electricity”,21]] | Print the modyfied list |
|  | Print the menu for F2 | The menu commands for 2a,2b,2c,2d |
|  | Ask for the command: | Prepare to get a command |
| 2d |  | Give the command for 2d |
|  | Ask for the ap. num | Prepare to get the ap num |
| 1 |  | Give the ap num |
|  | Ask for the expense type | Prepare to get the expense type |
| gas |  | Give the expense type |
|  | Ask for the expense value | Prepare to get the expense value |
| 200 |  | Give the expense value |
|  | [[1,”gas”,10]]🡪[[1,”gas”,200]] | Print the modyfied list |

# Iteration 3

## Running scenario

|  |  |  |
| --- | --- | --- |
| User | Program | Description |
|  | Ask for the command: | Prepare to get a command |
| 3 |  | Give the command for F3 |
|  | Print the menu for F3 | The menu commands for 3a,3b,3c |
|  | Ask for the command: | Prepare to get a command |
| 3a |  | Give the command for 3a |
|  | Print the entire list of expenses | The current list |
|  | Ask for the command: | Prepare to get a command |
| 3b |  | Give the command for 3b |
|  | Ask for the ap. num | Prepare to get the ap num |
| 1 |  | Give the ap num |
|  | [[1,”gas”,20]] | Print the expenses for ap 1 |
|  | Ask for the command: | Prepare to get a command |
| 3c |  | Give the command for 3c |
|  | Ask for the criteria | </=/> |
| > |  | Give the criteria |
|  | Ask for the amount | Prepare to get a value |
| 30 |  | Give the amount |
|  | [2,3,4] | Print the list of apartments with total expenses >30 |

# Iteration 4

## Running scenario

|  |  |  |
| --- | --- | --- |
| User | Program | Description |
|  | Ask for the command: | Prepare to get a command |
| 4 |  | Give the command for F4 |
|  | Print the menu for F4 | The menu commands for 4a,4b,4c,4d |
|  | Ask for the command: | Prepare to get a command |
| 4a |  | Give the command for 4a |
|  | Ask for the expense type | Prepare to get the expense type |
| water |  | Give the expense type |
|  | [[1,”water”,10],[2,”water”12],  [3,”heating”,11],[4,”electricity”,21]]  🡪22 | Print the value of all water expenses |
|  | Ask for the command: | Prepare to get a command |
| 4b |  | Give the command for 4b |
|  | Ask for the ap. num | Prepare to get the ap num |
| 1 |  | Give the ap num |
|  | [[1,”water”,10],[1,”water”12],  [1,”heating”,11],[1,”electricity”,21]]  🡪[[1,”water”,12],  [3,”heating”,11],[4,”electricity”,21]] | Print the maximum amount for each expense for ap 1 |
|  | Ask for the command: | Prepare to get a command |
| 4c |  | Give the command for 4c |
|  | Print the list of apartments sorted ascending by total amount of expenses | - |
|  | Ask for the command: | Prepare to get a command |
| 4d |  | Give the command for 4d |
|  | Print the total amount of expenses for each type, sorted ascending by amount of money | - |

# Iteration 5

## Running scenario

|  |  |  |
| --- | --- | --- |
| User | Program | Description |
|  | Ask for the command: | Prepare to get a command |
| 5 |  | Give the command for F5 |
|  | Print the menu for F5 | The menu commands for 5a,5b |
|  | Ask for the command: | Prepare to get a command |
| 5a |  | Give the command for 5a |
|  | Ask for the expense type | Prepare to get the expense type |
| water |  | Give the expense type |
|  | [[1,”water”,10],[1,”water”12]  [1,”heating”,11],[1,”electricity”,21]]🡪 [[1,”water”,10],[1,”water”12]] |  |
|  | Ask for the command: | Prepare to get a command |
| 5b |  | Give the command for 5b |
|  | Ask for the amount | Prepare to get a value |
| 30 |  | Give the amount |
|  | [[1,”water”,50],[1,”water”12]  [1,”heating”,11],[1,”electricity”,30]]🡪  [[1,”water”12],  [1,”heating”,11]] | Keep only the expenses having the value smaller than 30 |

# Iteration 6

## Running scenario

|  |  |  |
| --- | --- | --- |
| User | Program | Description |
|  | Ask for the command: | Prepare to get a command |
| 6 |  | Give the command for F6 |
|  | Reverse the last operation on the list | The undo command |

***Work items/tasks***

|  |  |
| --- | --- |
| T1 | Implement domain functions: sum type, sum expenses for an apartment |
| T2 | Implement iterations 1 and 2: add, remove, replace |
| T3 | Implement iterations 3 and 4: list, sum, max, sort |
| T4 | Implement iterations 5 and 6: filter, undo |
| T5 | Implement user interface |