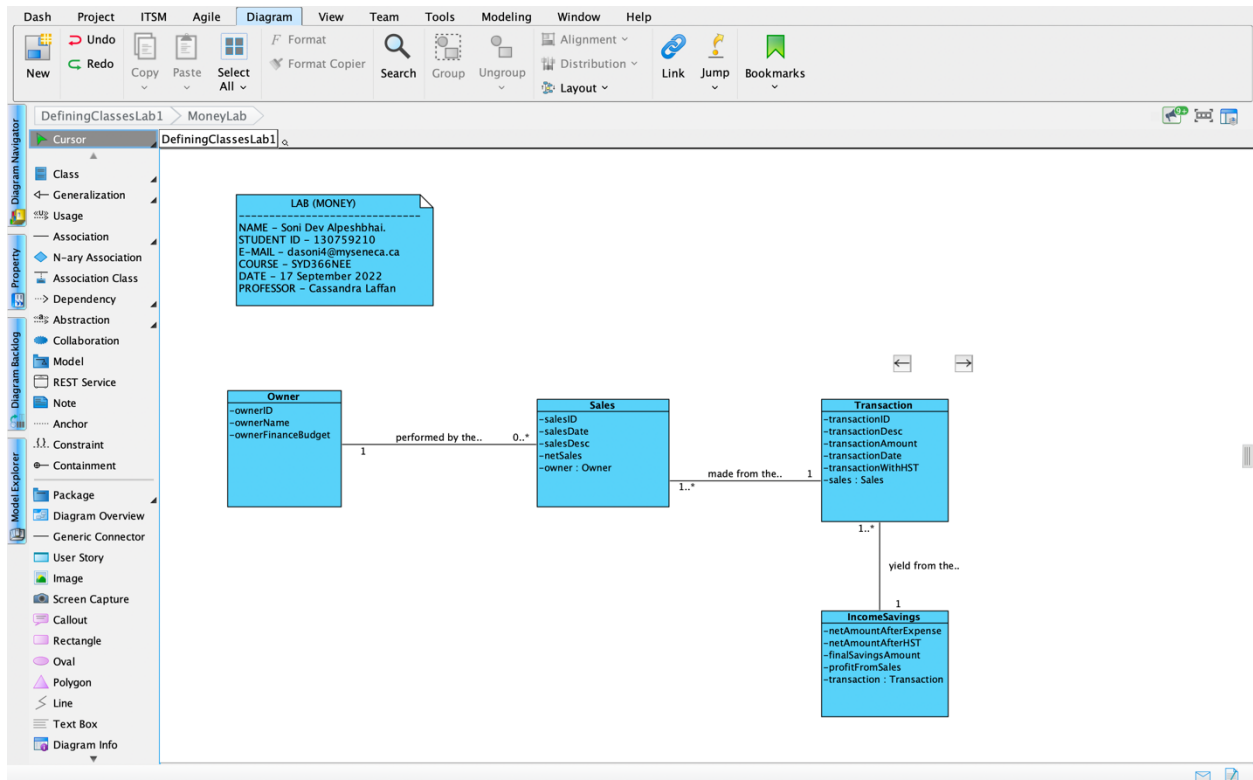
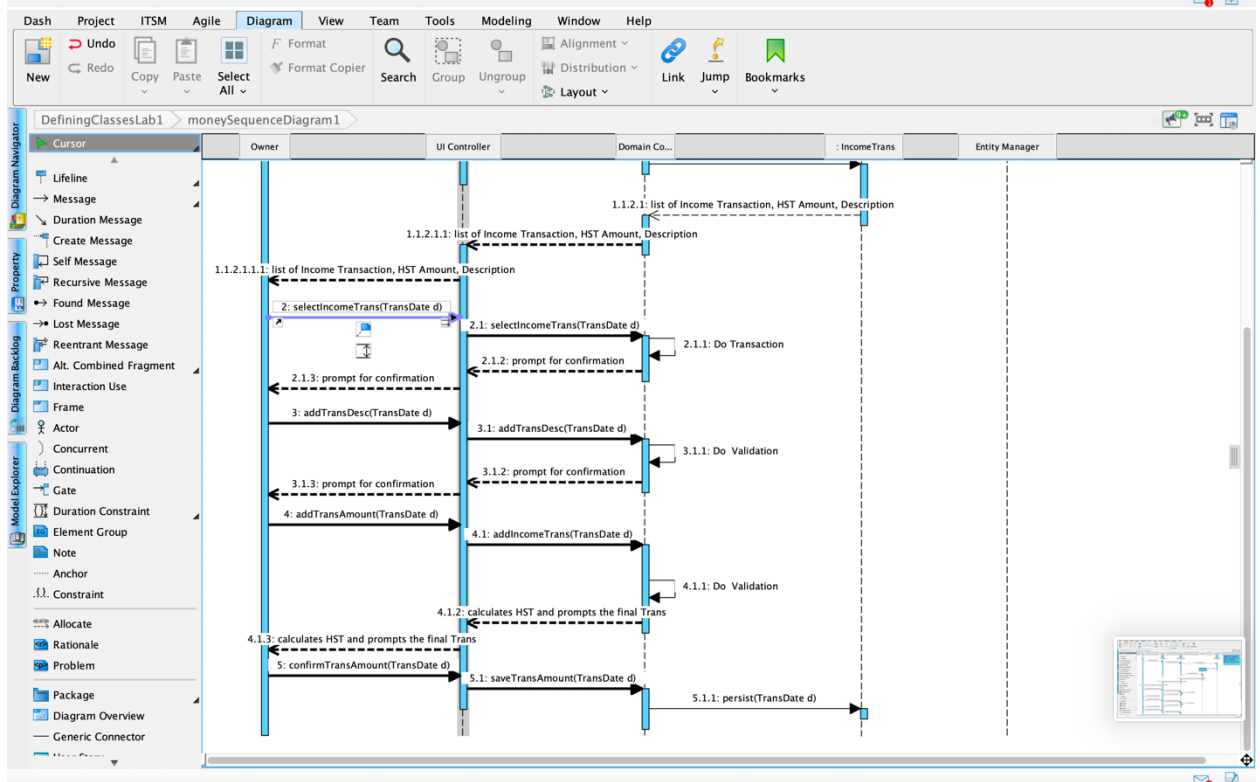
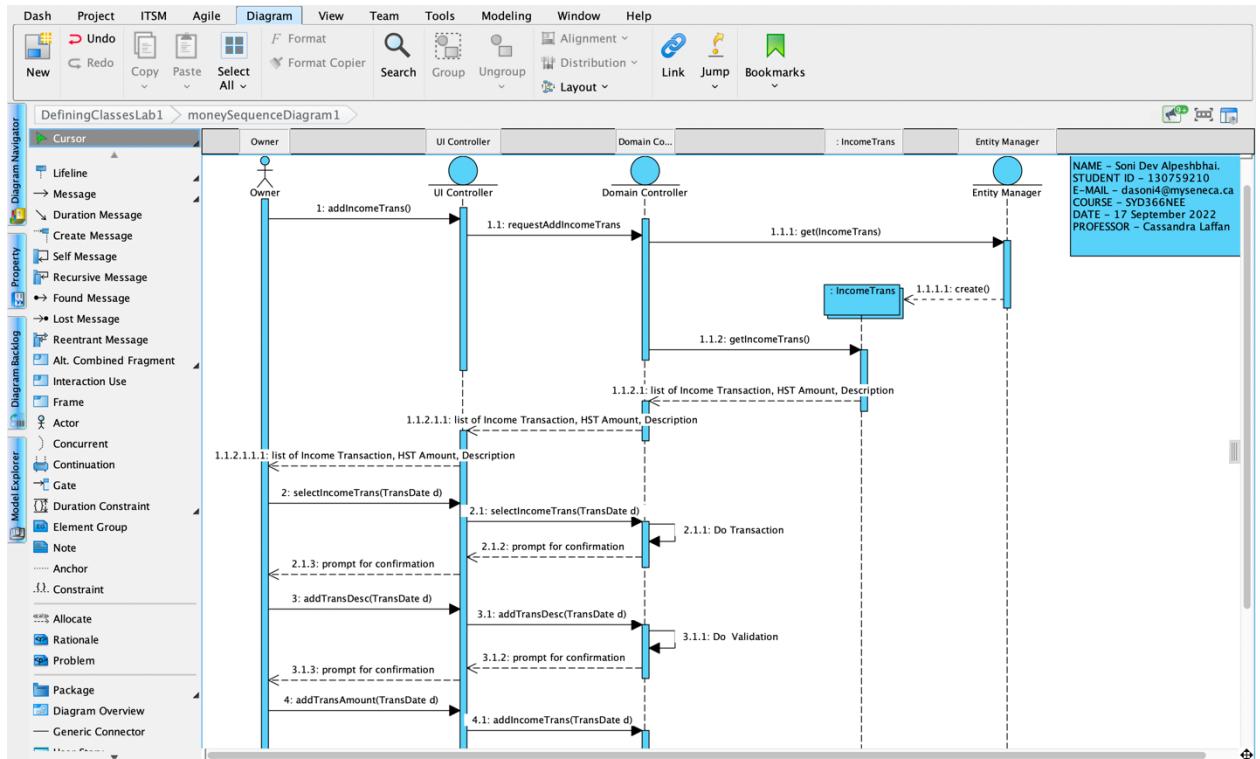


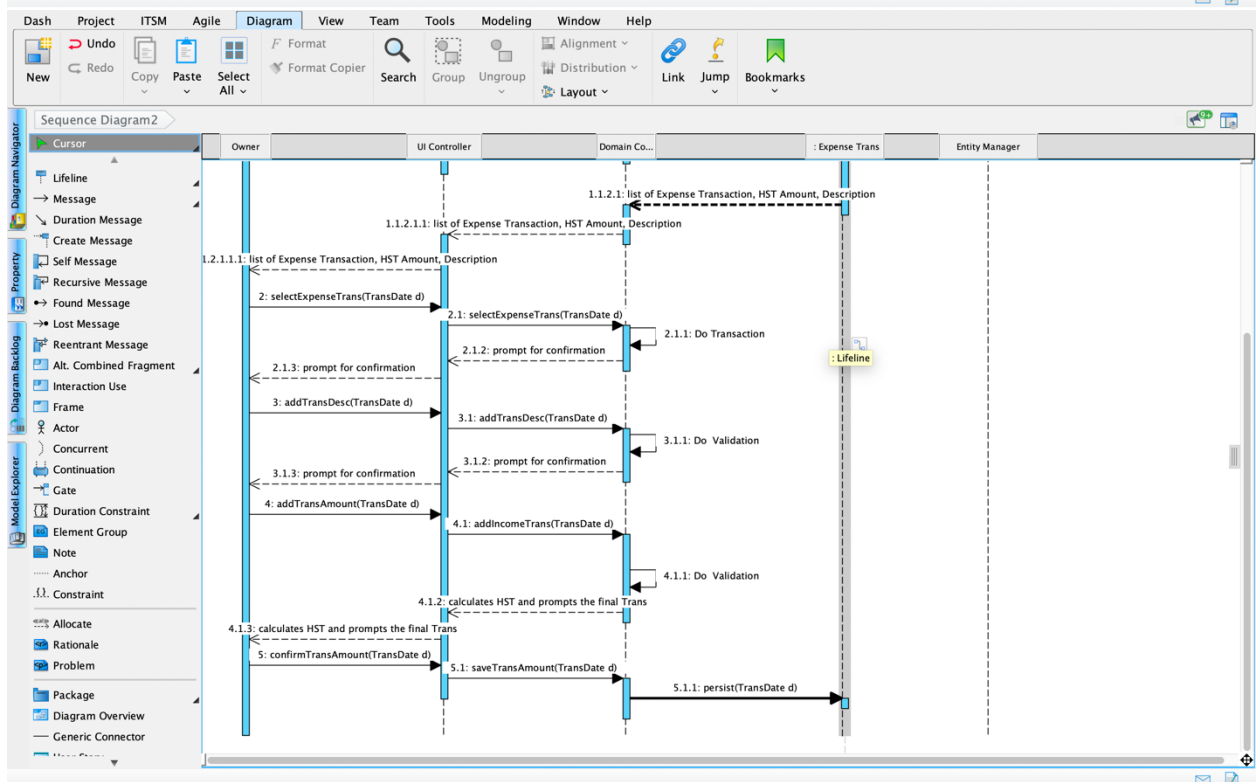
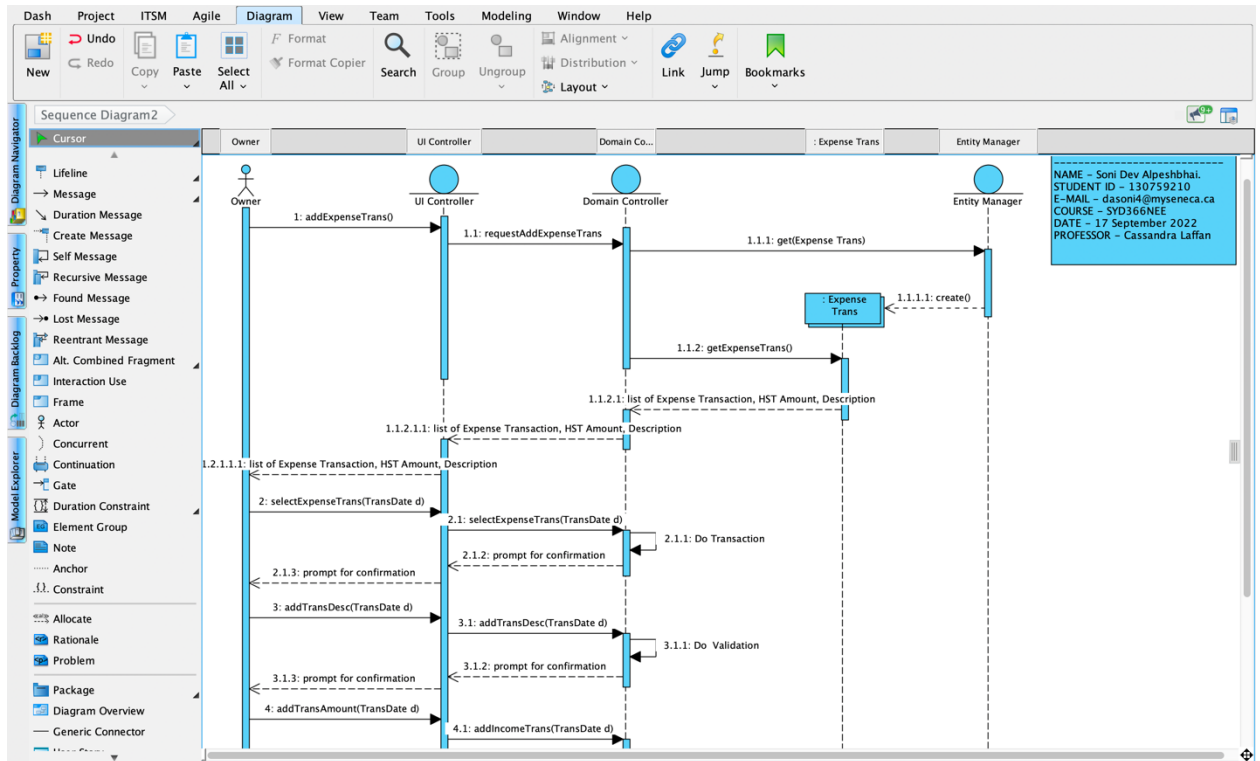
1. CLASS DIAGRAMS



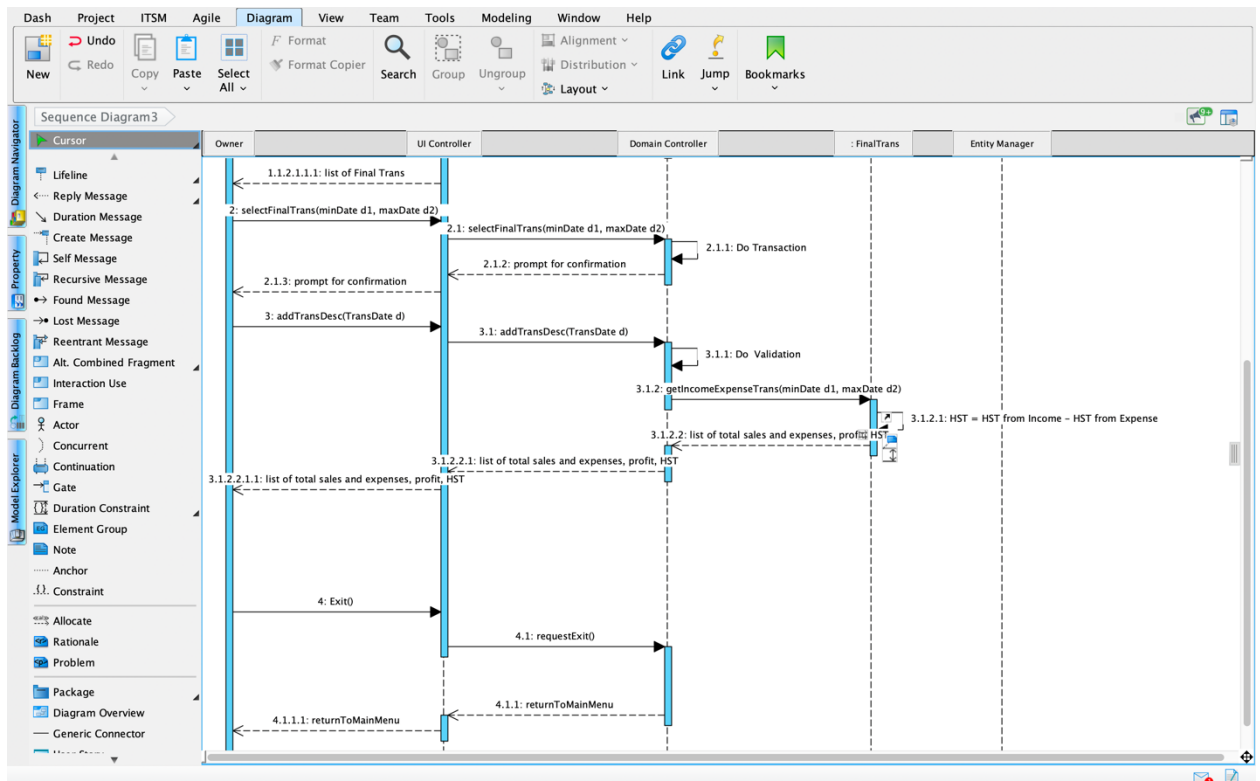
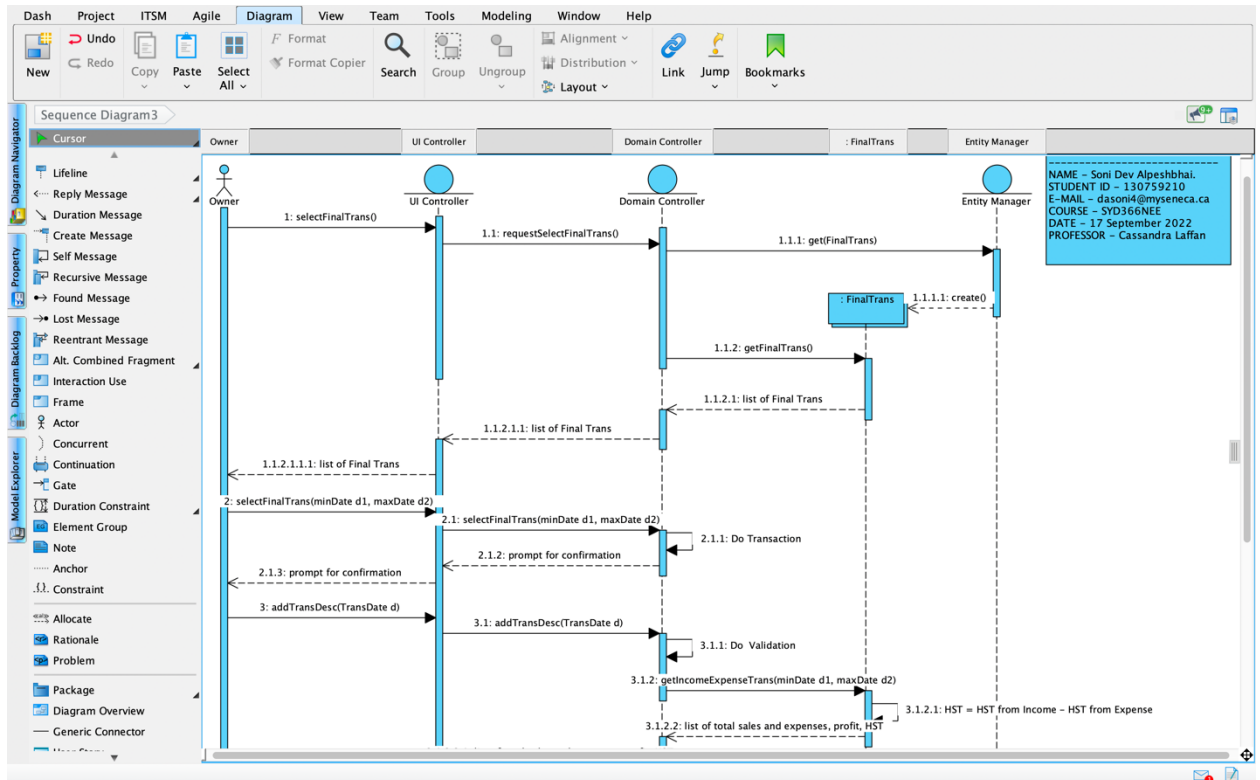
2. HEADER FILE (USE CASE 1)



USE CASE – 2 :



USE CASE – 3 :



Header files

```
C MoneyLab.h
11
12 #ifndef MONEYTRANS_H_ // Conditional Header Commands to the console.
13 #define MONEYTRANS_H_
14
15 // Standard or Other Libraries for the program.
16 #include <stdio.h>
17
18 // All the Class Declarations for the program.
19 class Owner{
20
21     const int OwnerID;
22     char ownerName[100];
23     double ownerFinanceBudget;
24
25 public:
26
27     // Functionalities
28
29     void requestAddIncomeTrans(); // to add Income Transaction
30
31     void selectIncomeTrans(int transDate) const; // select Income Transaction, between specific dates.
32
33     void addTransDesc(int transDate); // add Transaction description.
34
35     void requestAddExpenseTrans(); // to add Expense Transaction
36
37     void selectExpenseTrans(int transDate) const; //select expense Transaction, between specific dates.
38
39     void requestFinalTrans(); // request final transaction
40
41     void selectFinalTrans(int transDate) const; // select final Transaction, for specific dates.
42
43     void selectFinalTrans(int minDate, int maxDate); // select final Transaction, between specific dates.
44
45     void exit(); // exit to main menu
46
47 };
48 class Sales{
49
50     const int SalesID;
51
52     int salesDate;
53
54     char salesDesc[500];
55
56     double netSales;
57
58 }
```

```
C MoneyLab.h
58
59 Owner* owner; // reference attribute of the class Owner.
60 };
61
62 class Transaction{
63
64     int transactionID;
65
66     int transactionDate;
67
68     double transactionWithHST;
69
70     char transactionDesc[500];
71
72     double transactionAmount;
73
74     Sales* sales; // reference attribute of the class Sales.
75
76 public :
77     int addIncomeTrans(); // add Transaction description.
78     int addExpenseTrans(); // to add Expense Transaction
79     int HSTcalculator(double finalTrans); // to convert transaction to HST.
80 };
81
82 class IncomeSavings{
83
84     double netAmountAfterExpense;
85
86     double netAmountAfterHST;
87
88     double finalSavingsAmount;
89
90     double profitFromSales;
91
92     Transaction* transaction; // reference attribute of the Transaction.
93
94 public:
95
96     void confirmTransAmount(int transDate) const; // This functionality will confirm the transaction amount from the List, chose by the user.
97
98     void saveTransAmount(int transDate); // This functionality will save the transaction amount, chose by the user onto the Domain Controller (System).
99
100 };
101
102 #endif
```