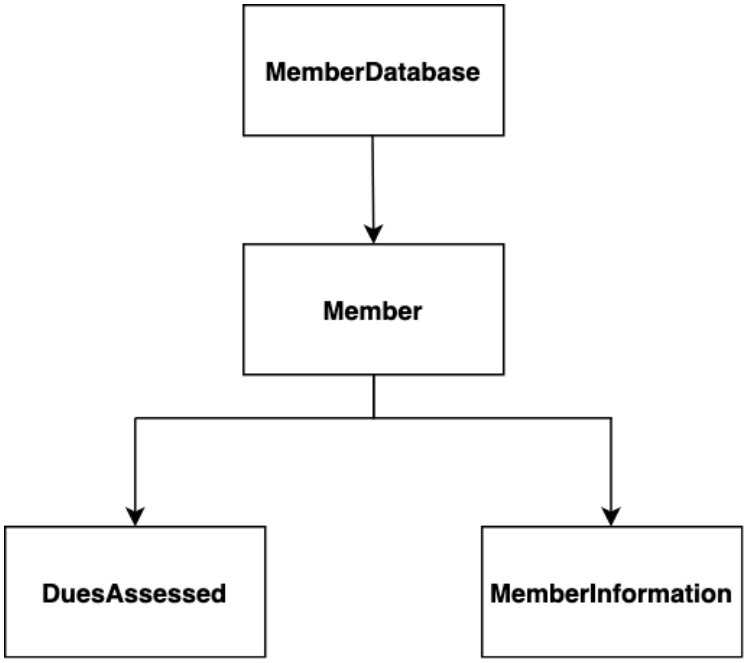
Criterion B: Design

Internal Structure



Functionality of Classes

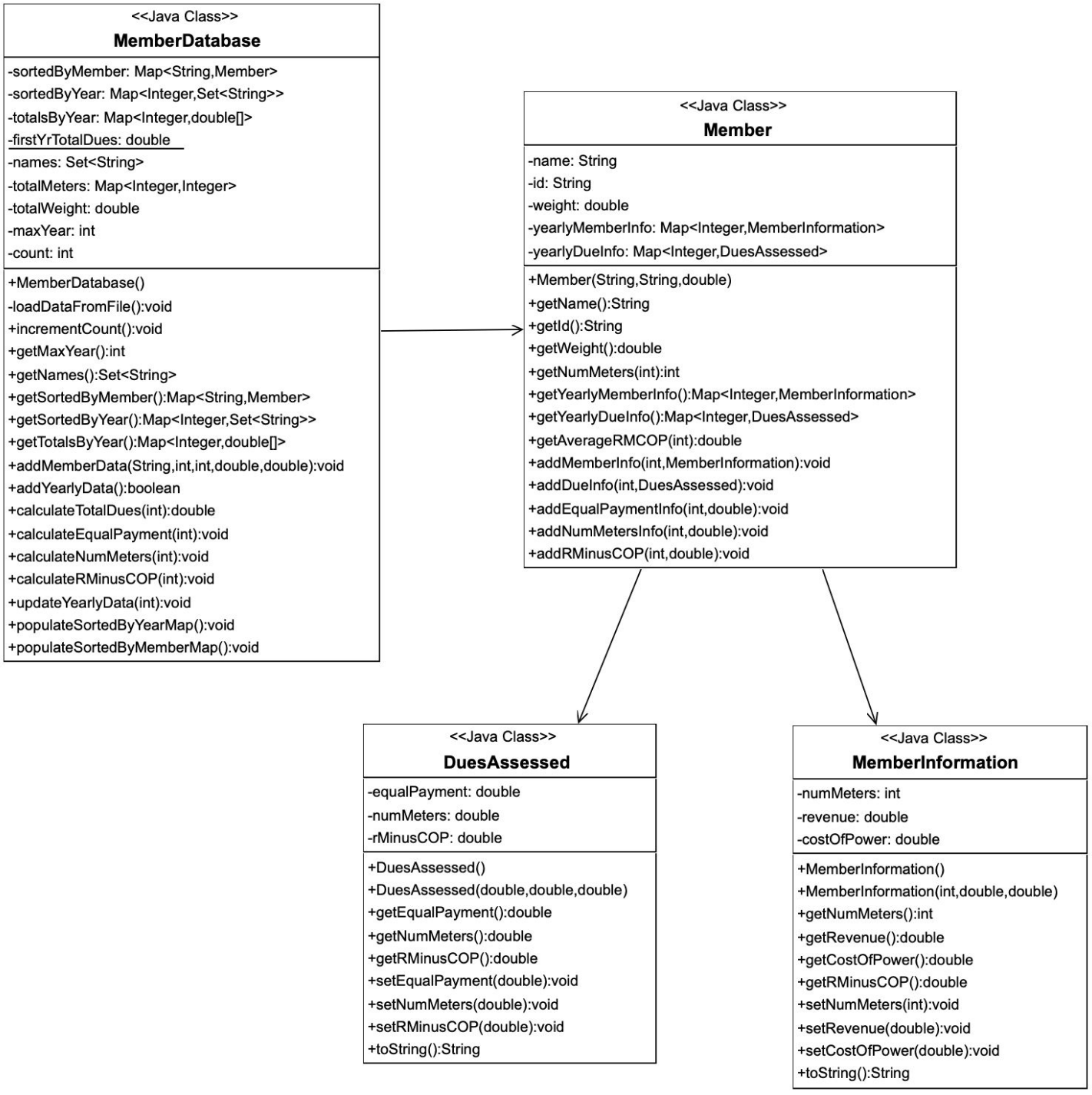
MemberDatabase - loads data provided by the client; contains membership information and dues assessed for all members; performs due calculations.

Member - contains member information and dues assessed for a member.

MemberInformation - stores the number of meters, revenue, and cost of power for a member.

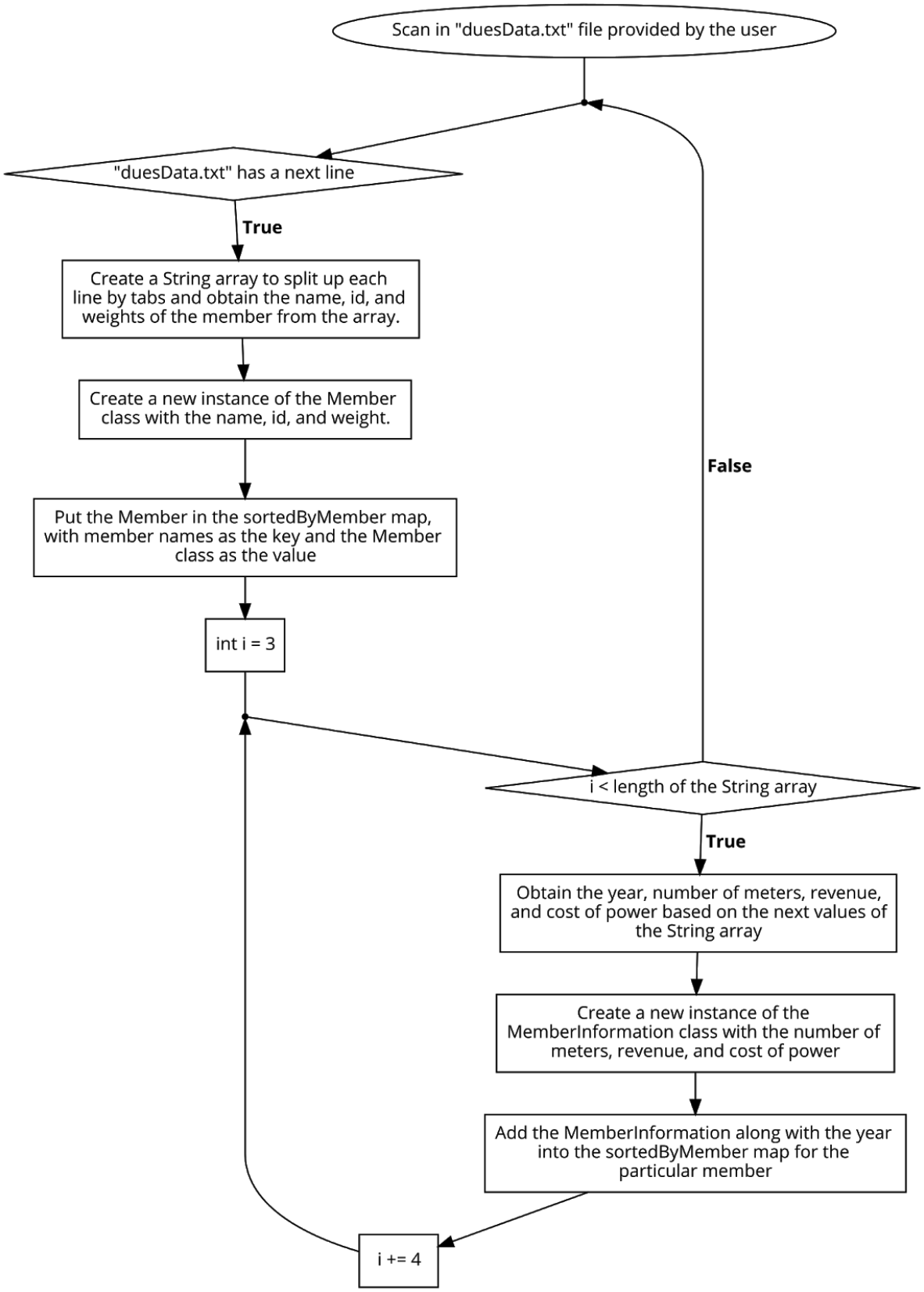
DuesAssessed - stores the due components (equal payment, number of meters, and revenue minus cost of power) assessed to a member.

Class Diagrams

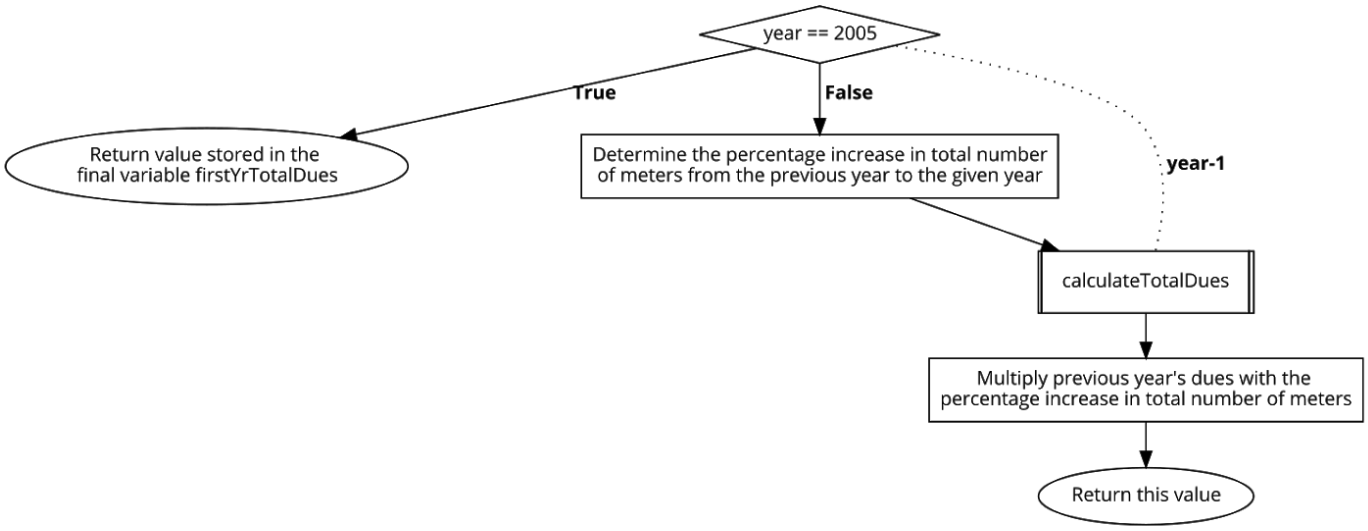


Flowcharts for Select Methods in MemberDatabase Class

loadDataFromFile()



calculateTotalDues(int)



Pseudo Code for Select Methods in MemberDatabase Class

calculateEqualPayment(int YEAR)

```
DUESASSESSED = calculateTotalDues(YEAR) / 3.0
for each N in SORTEDBYMEMBER.keySet()
       EQUALPAYMENT = (SORTEDBYMEMBER.get(N).getWeight() / TOTALWEIGHT) * DUESASSESSED
       SORTEDBYMEMBER.get(N).addEqualPaymentInfo(YEAR, EQUALPAYMENT)
```

calculateNumMeters(int YEAR)

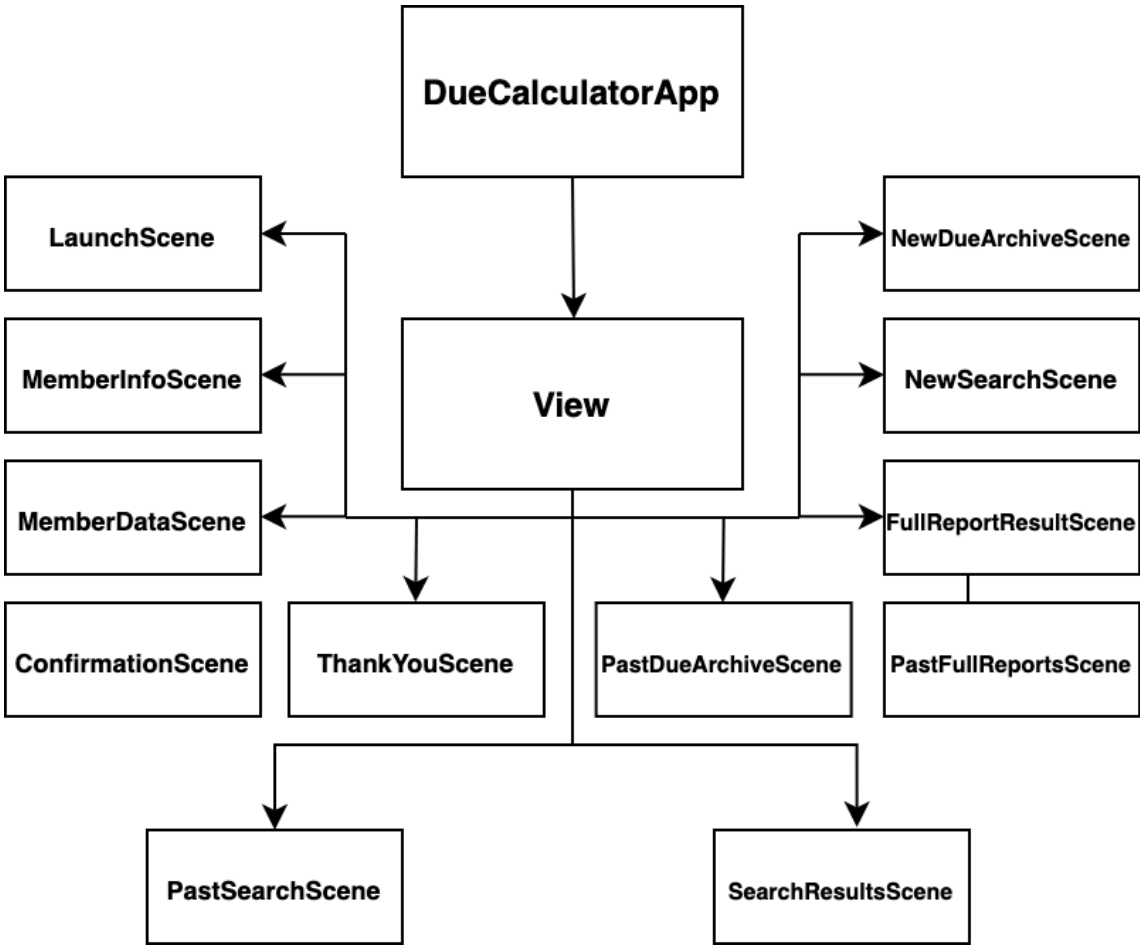
```
DUESASSESSED = calculateTotalDues(YEAR) / 3.0
TIERTOTALS = new int[6]
WEIGHT = {1.725, .675, .65, .625, .60, .575}
TIERDUES = new double[6]
for each N in SORTEDBYMEMBER.keySet()
       NUMMETERS = SORTEDBYMEMBER.get(N).getNumMeters(YEAR)
       INDEX = 0
       while NUMMETERS > 0
            if NUMMETERS <=5000 AND INDEX < 6
                 TIERTOTALS[INDEX] += NUMMETERS
            else if INDEX < 6
                 TIERTOTALS[INDEX] += 5000
            else
                 TIERTOTALS[5]+= NUMMETERS
                 NUMMETERS = 0
            NUMMETERS -= 5000
            INDEX ++

SUM = 0
loop COUNT from 0 to 5
       SUM += TIERTOTALS[COUNT] * WEIGHT[COUNT]

loop COUNT from 0 to 5
       TIERDUES[COUNT] = DUESASSESSED * ((TIERTOTALS[COUNT] * WEIGHT[COUNT])/SUM)

for each N in SORTEDBYMEMBER.keySet()
       NUMMETERS = SORTEDBYMEMBER.get(N).getNumMeters(YEAR)
       INDEX = 0
       while NUMMETERS > 0
            if NUMMETERS <=5000 AND INDEX < 6
```

```
                DUES = TIERDUES[INDEX] * (NUMMETERS/TIERTOTALS[INDEX])
                SORTEDBYMEMBER.get(N).addNumMetersInfo(YEAR, DUES)
          else if INDEX < 6
                DUES = TIERDUES[INDEX] * (5000.0/TIERTOTALS[INDEX])
                SORTEDBYMEMBER.get(N).addNumMetersInfo(YEAR, DUES)
          else
                DUES = TIERDUES[5] * (NUMMETERS/TIERTOTALS[INDEX])
                SORTEDBYMEMBER.get(N).addNumMetersInfo(YEAR, DUES)
                NUMMETERS = 0
          NUMMETERS -= 5000
          INDEX ++
```

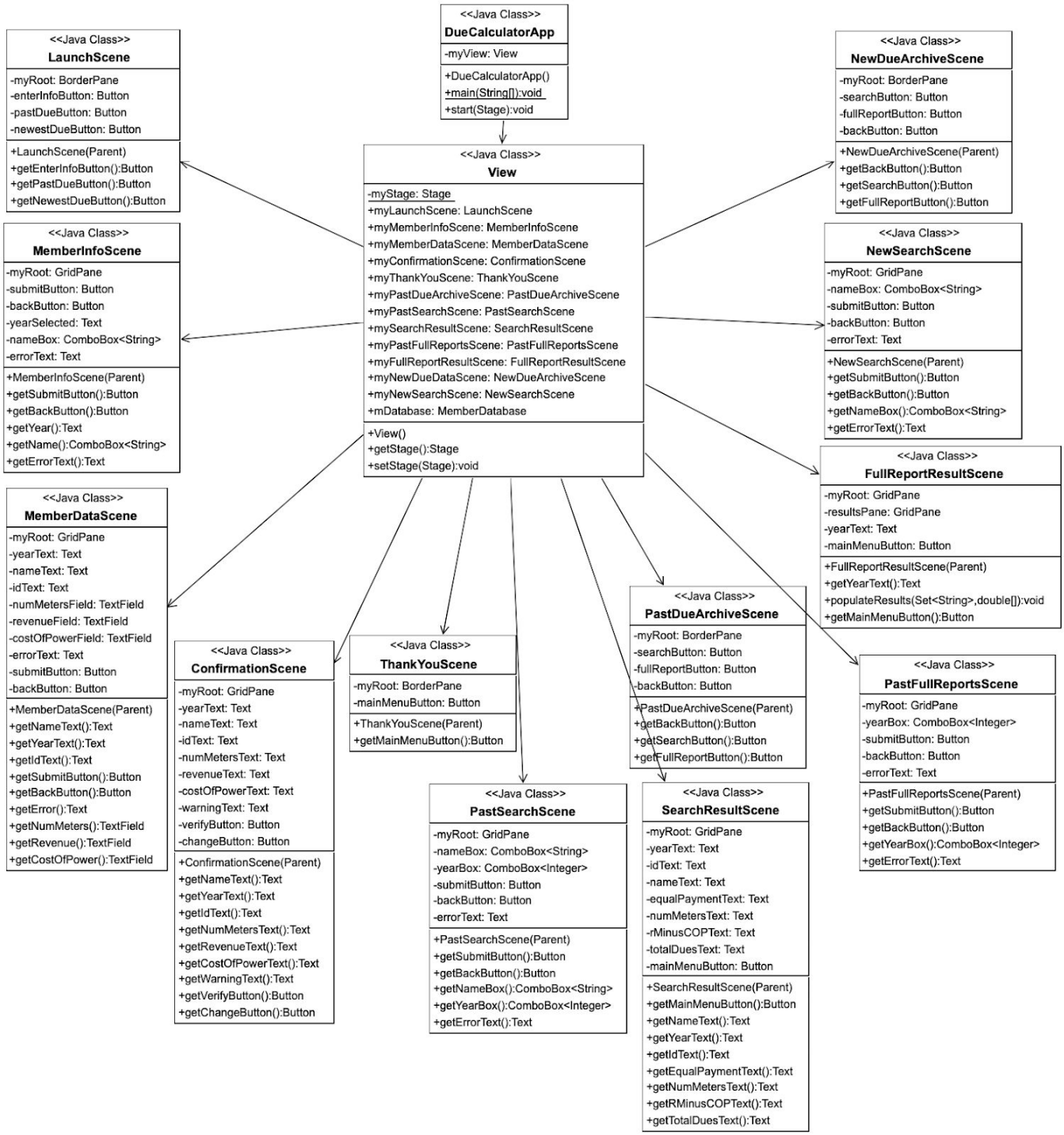
```
                                calculateRMinusCOP(int YEAR)

DUESASSESSED = calculateTotalDues(YEAR) / 3.0
REVTOTALS = new int[7]
WEIGHT = {1.625, .625, .575, .55, .525, .50, .475}
REVDUES = new double[7]
for each N in SORTEDBYMEMBER.keySet()
       RMCOP = SORTEDBYMEMBER.get(N).getAverageRMCOP(YEAR)
       INDEX = 0
       while RMCOP > 0
            if INDEX = 0 OR INDEX = 1
                 if RMCOP <= 2500000
                      REVTOTALS[INDEX] += RMCOP
                      RMCOP = 0
                 else
                      REVTOTALS[INDEX] += 2500000
                 RMCOP -= 2500000
            else if INDEX > 1 AND INDEX < 7
                 if RMCOP <= 5000000
                      REVTOTALS[INDEX] += RMCOP
                 else
                      REVTOTALS[INDEX] += 5000000
                 RMCOP -= 5000000
            else
                 REVTOTALS[6] += RMCOP
                 RMCOP = 0
            INDEX ++

SUM = 0
loop COUNT from 0 to 6
       SUM += REVTOTALS[COUNT] * WEIGHT[COUNT]

loop COUNT from 0 to 6
       REVDUES[COUNT] = DUESASSESSED * ((REVTOTALS[COUNT] * WEIGHT[COUNT]) / SUM)

for each N in SORTEDBYMEMBER.keySet()
       RMCOP = SORTEDBYMEMBER.get(N).getAverageRMCOP(YEAR)
       INDEX = 0
       while RCMOP > 0
            if INDEX = 0 OR INDEX = 1
                  if RMCOP <= 2500000
                       DUES = REVDUES[INDEX] * (RMCOP/REVTOTALS[INDEX])
                       SORTEDBYMEMBER.get(N).addRMinusCOPInfo(YEAR, DUES)
                       RMCOP = 0
                  else
                       DUES = REVDUES[INDEX] * (2500000.0/REVTOTALS[INDEX])
                       SORTEDBYMEMBER.get(N).addRMinusCOPInfo(YEAR, DUES)
                  RMCOP -= 2500000
            else if INDEX > 1 AND INDEX < 7
                  if RMCOP <= 5000000
                       DUES = REVDUES[INDEX] * (RMCOP/REVTOTALS[INDEX])
```

```
                SORTEDBYMEMBER.get(N).addRMinusCOPInfo(YEAR, DUES)
          else
                DUES = REVDUES[INDEX] * (5000000.0/REVTOTALS[INDEX])
                SORTEDBYMEMBER.get(N).addRMinusCOPInfo(YEAR, DUES)
          RMCOP -= 5000000
    else
          DUES = REVDUES[6] * (RMCOP/REVTOTALS[6])
          SORTEDBYMEMBER.get(N).addRMinusCOPInfo(YEAR, DUES)
          RMCOP = 0
    INDEX ++
```

External Structure

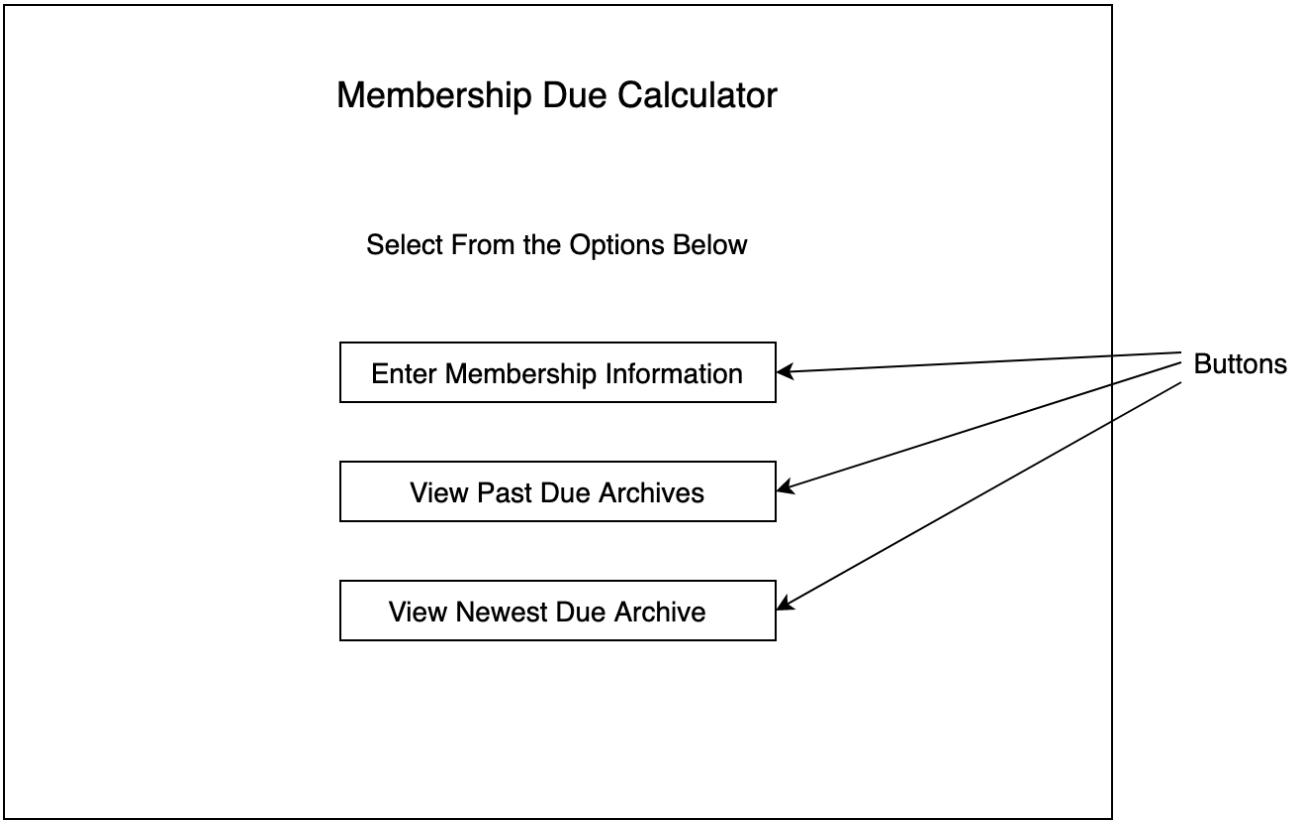


Class Diagrams

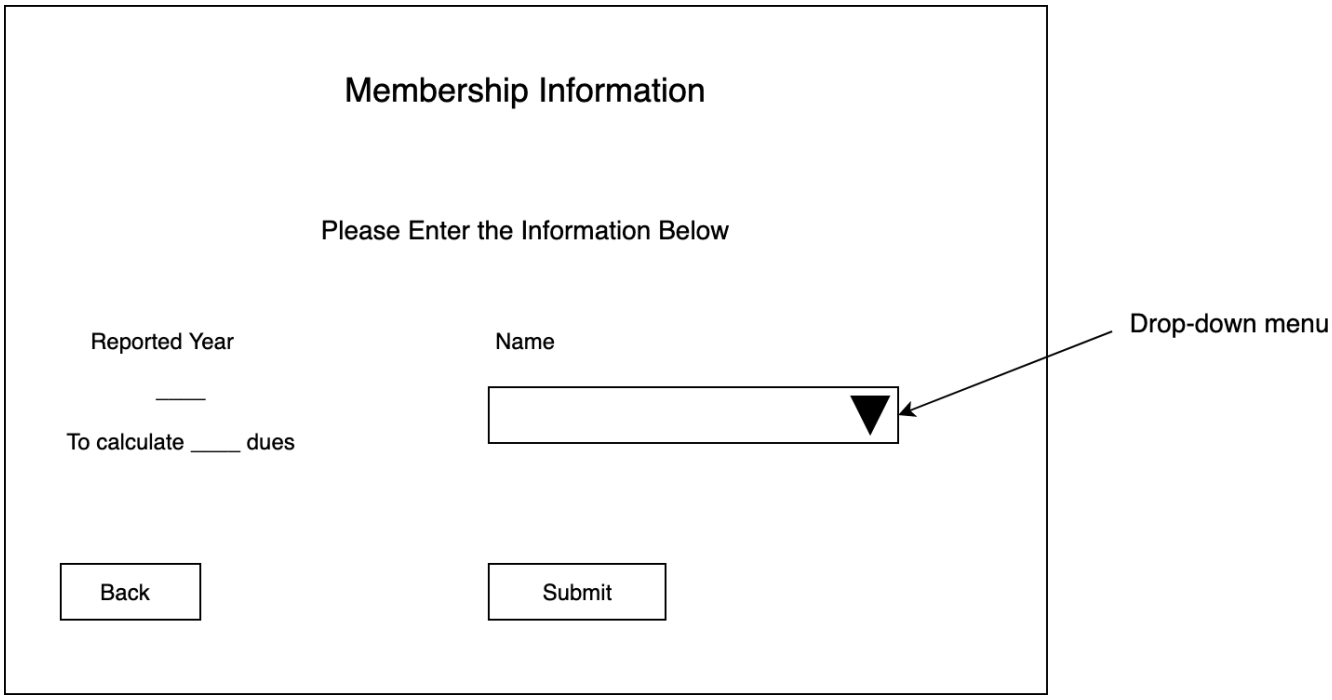


Design of User Interface

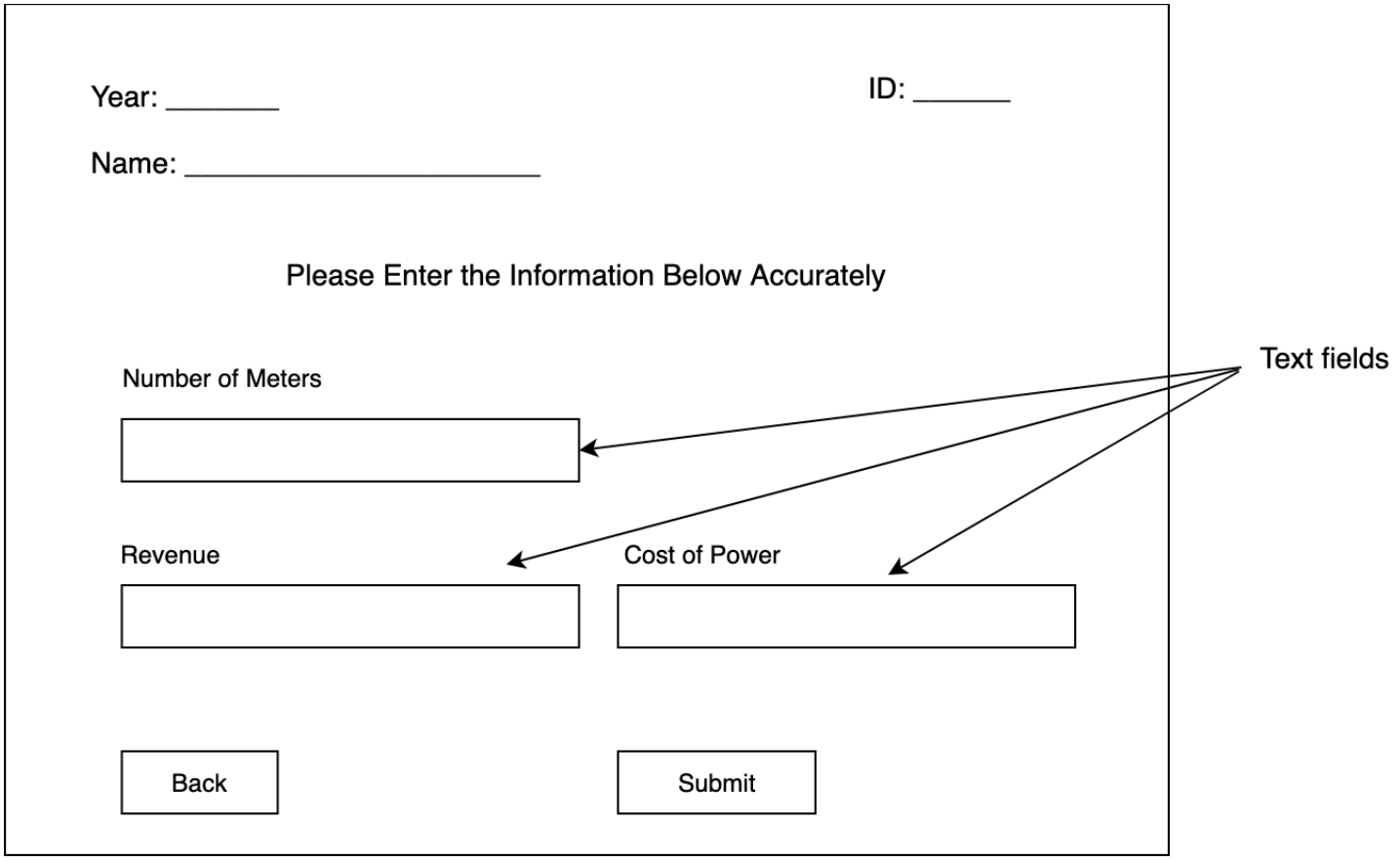
LaunchScene



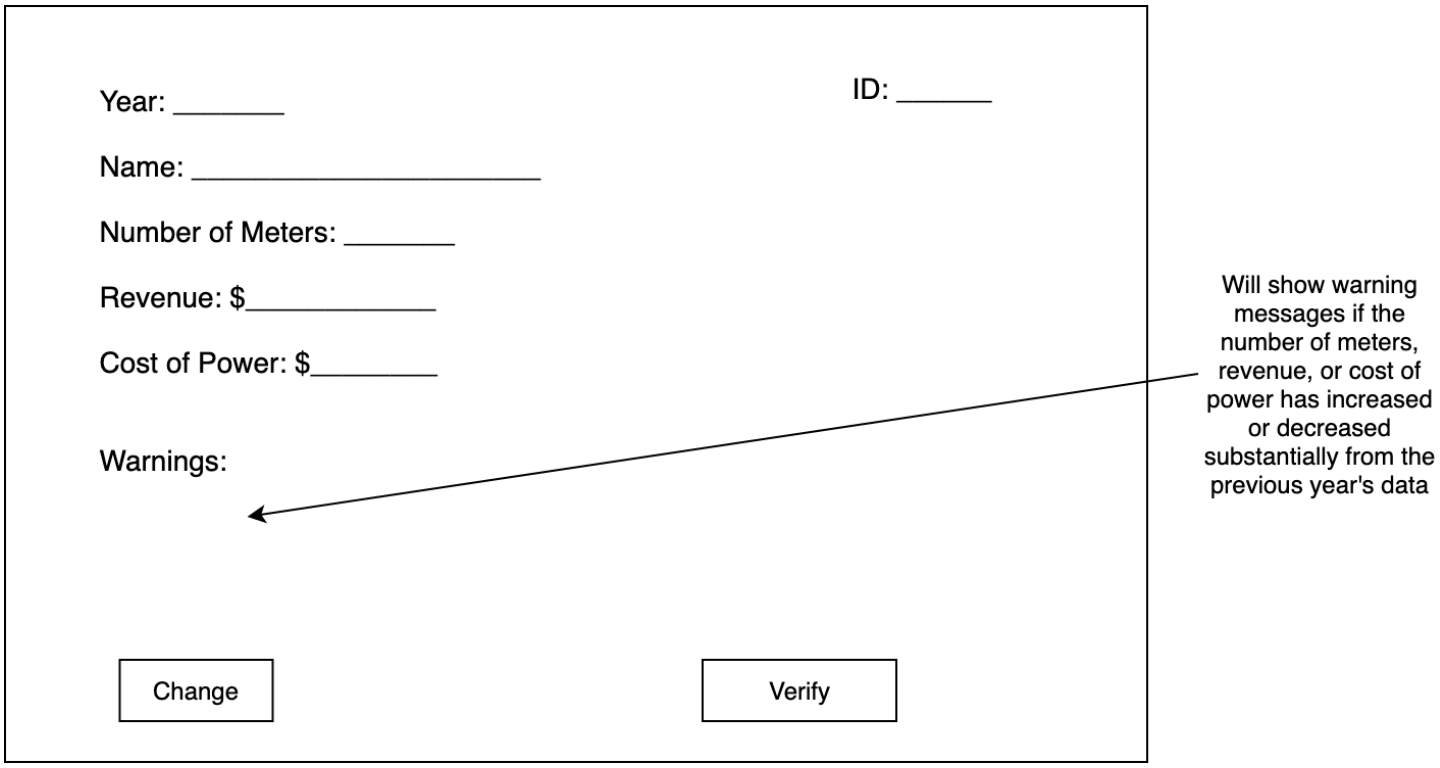
MemberInfoScene



MemberDataScene

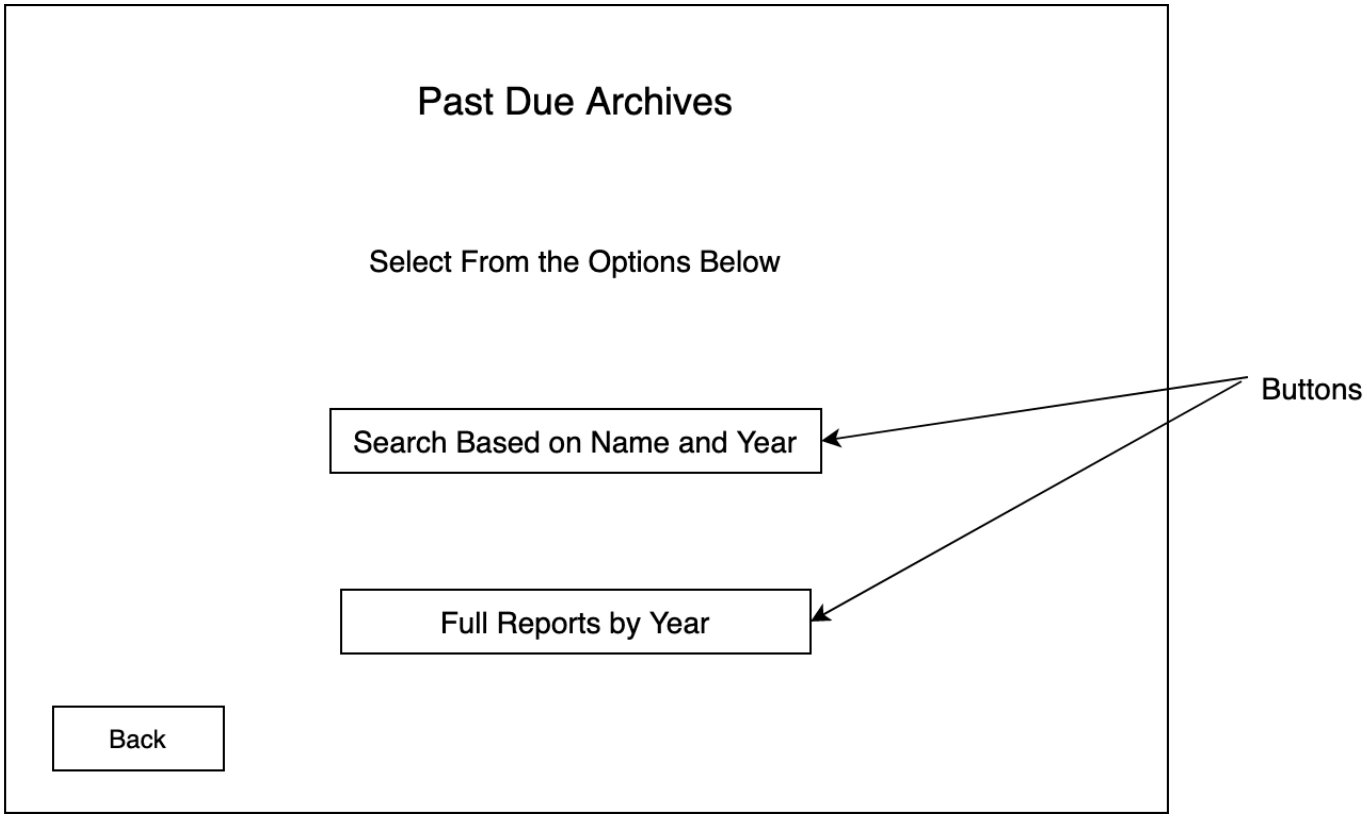


ConfirmationScene

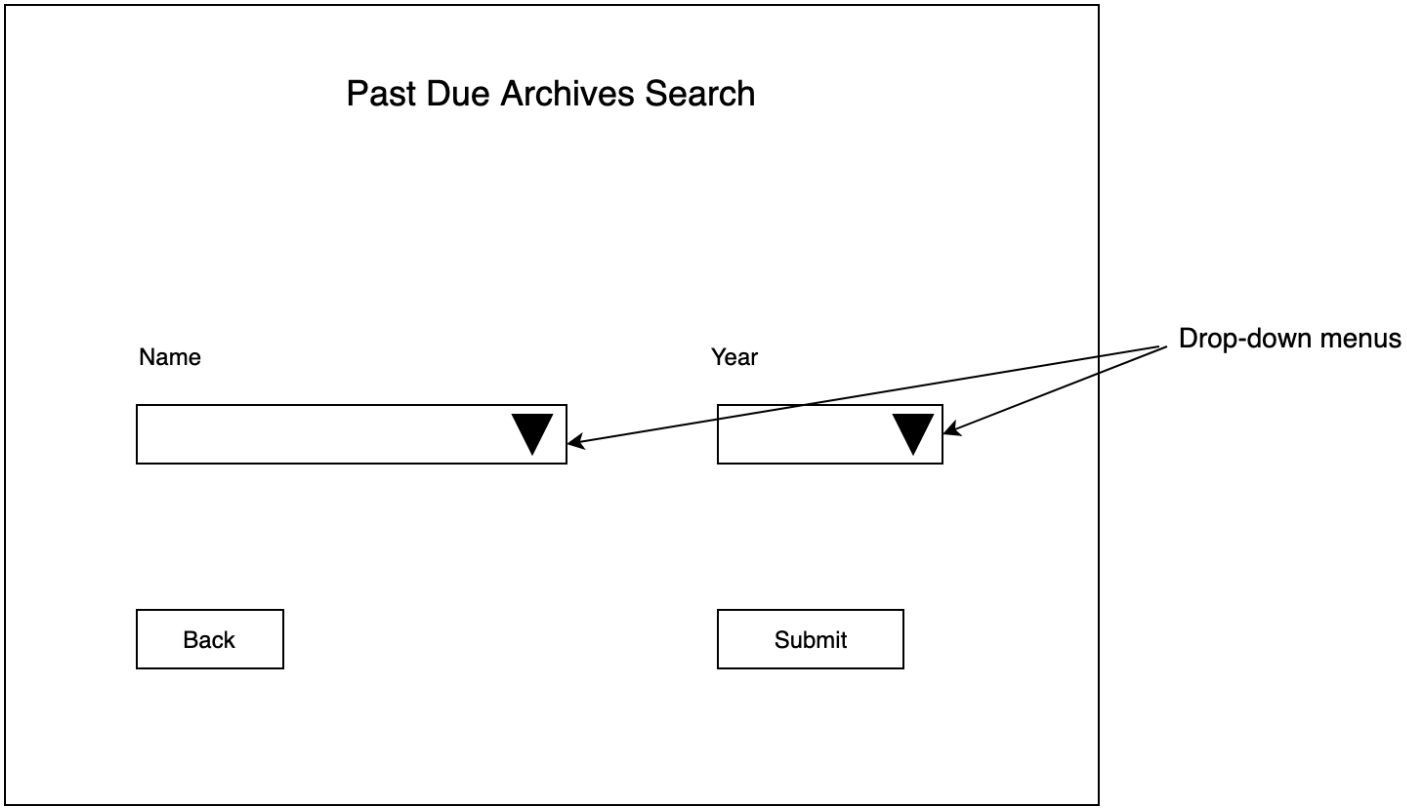


ThankYouScene

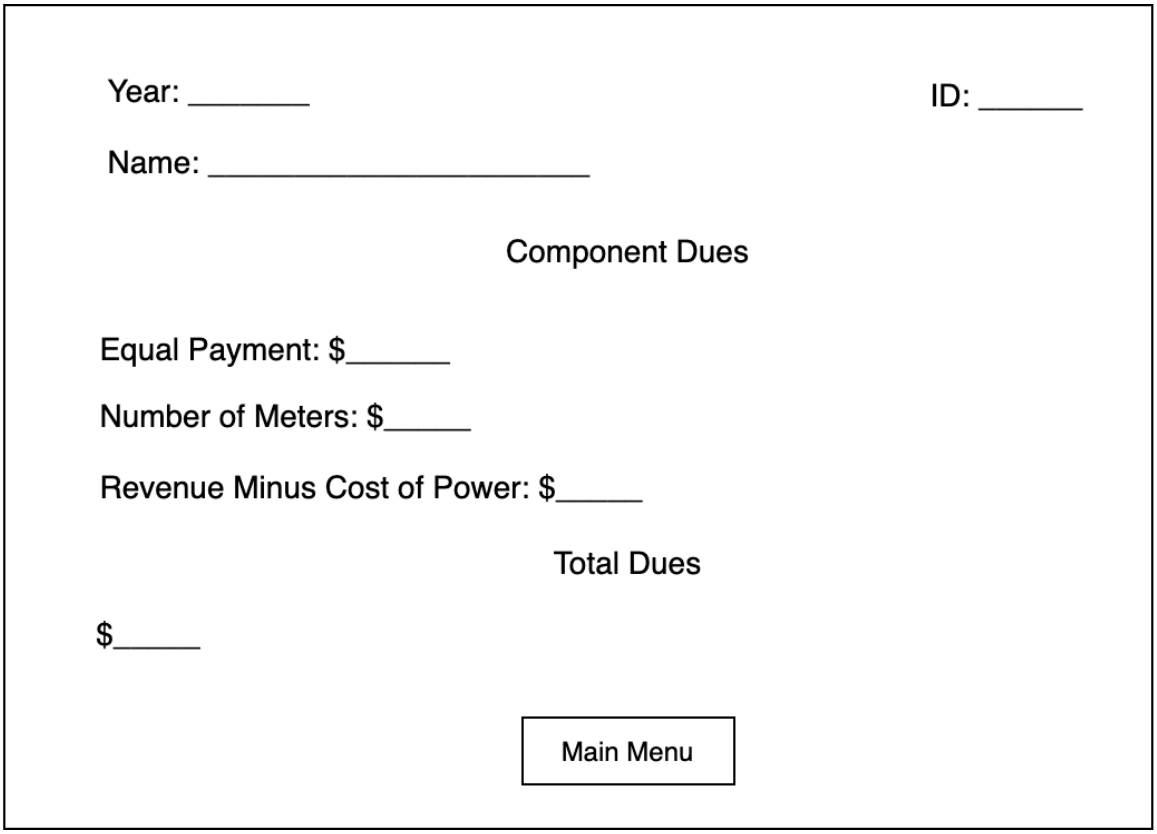
PastDueArchiveScene



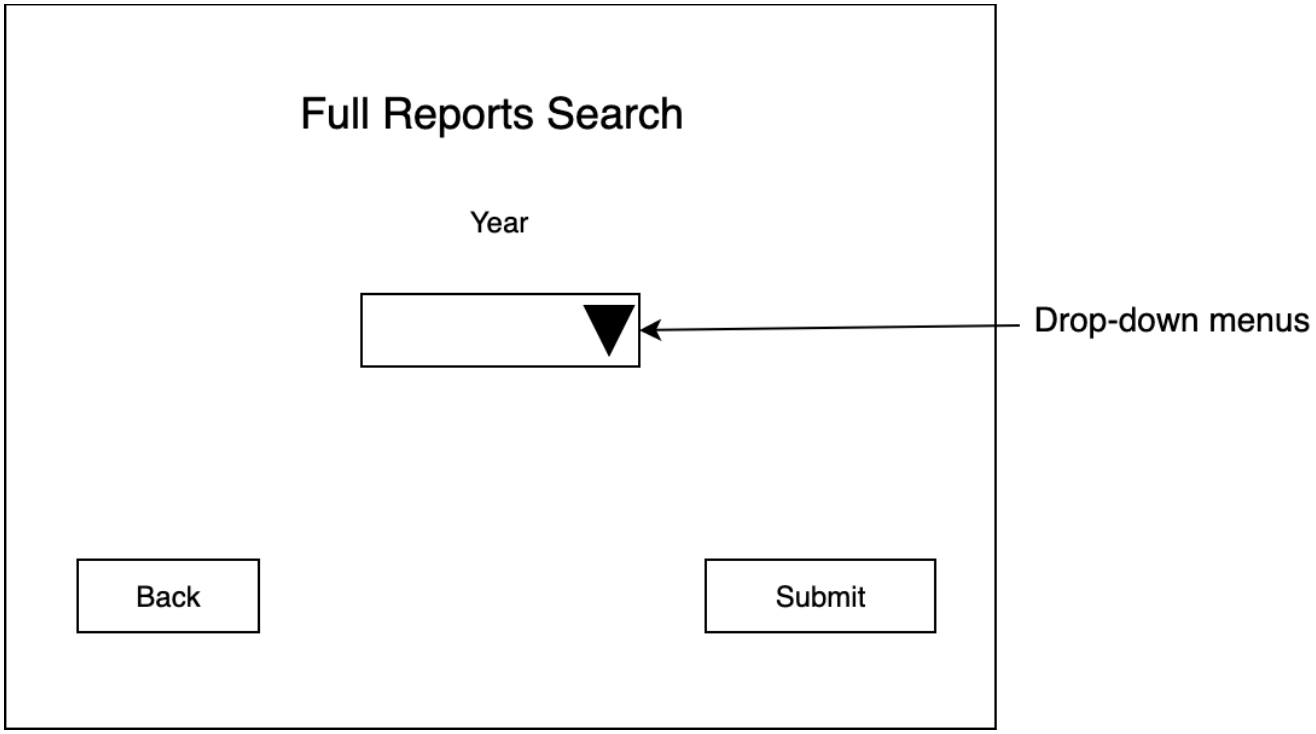
PastSearchScene



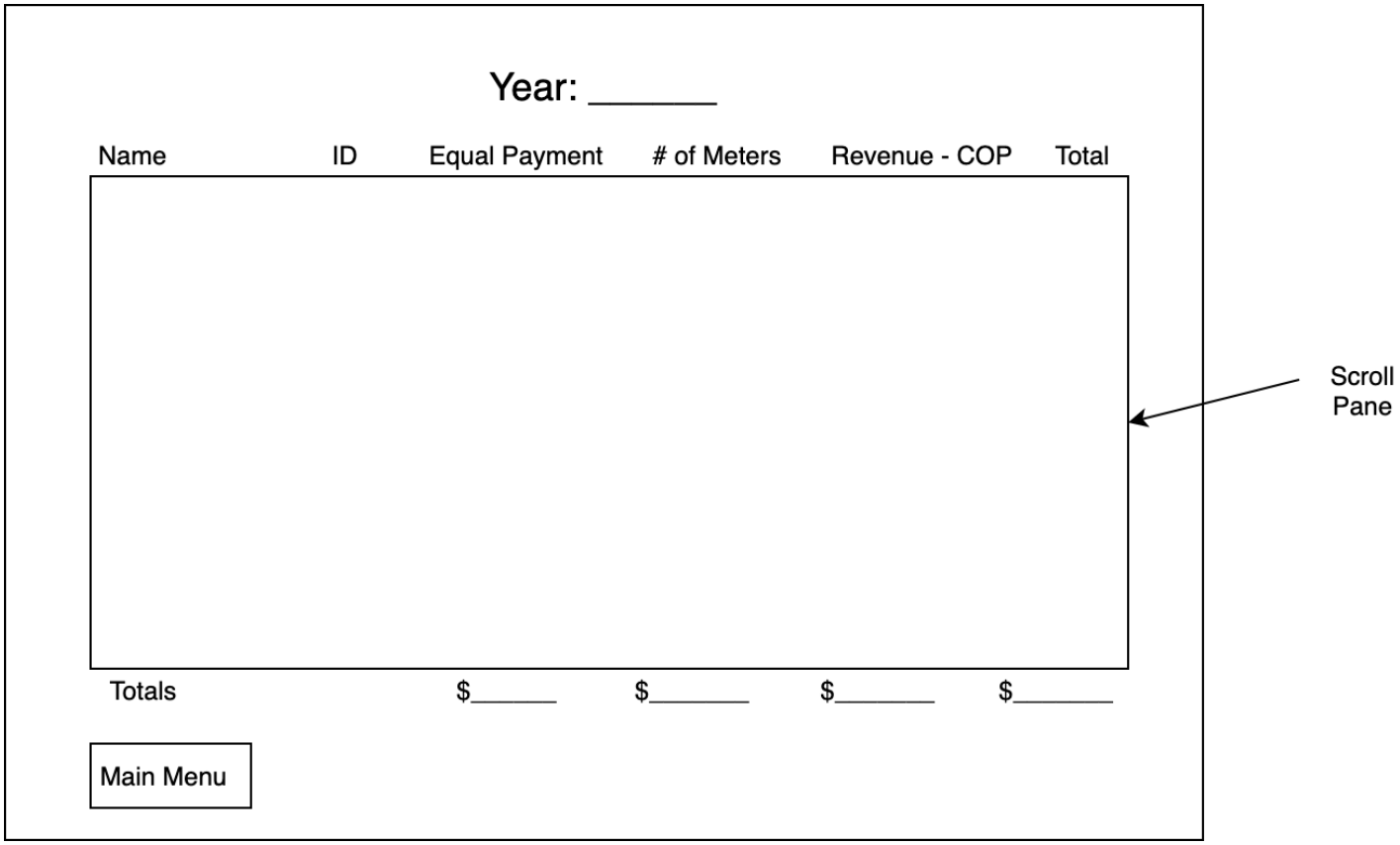
SearchResultsScene



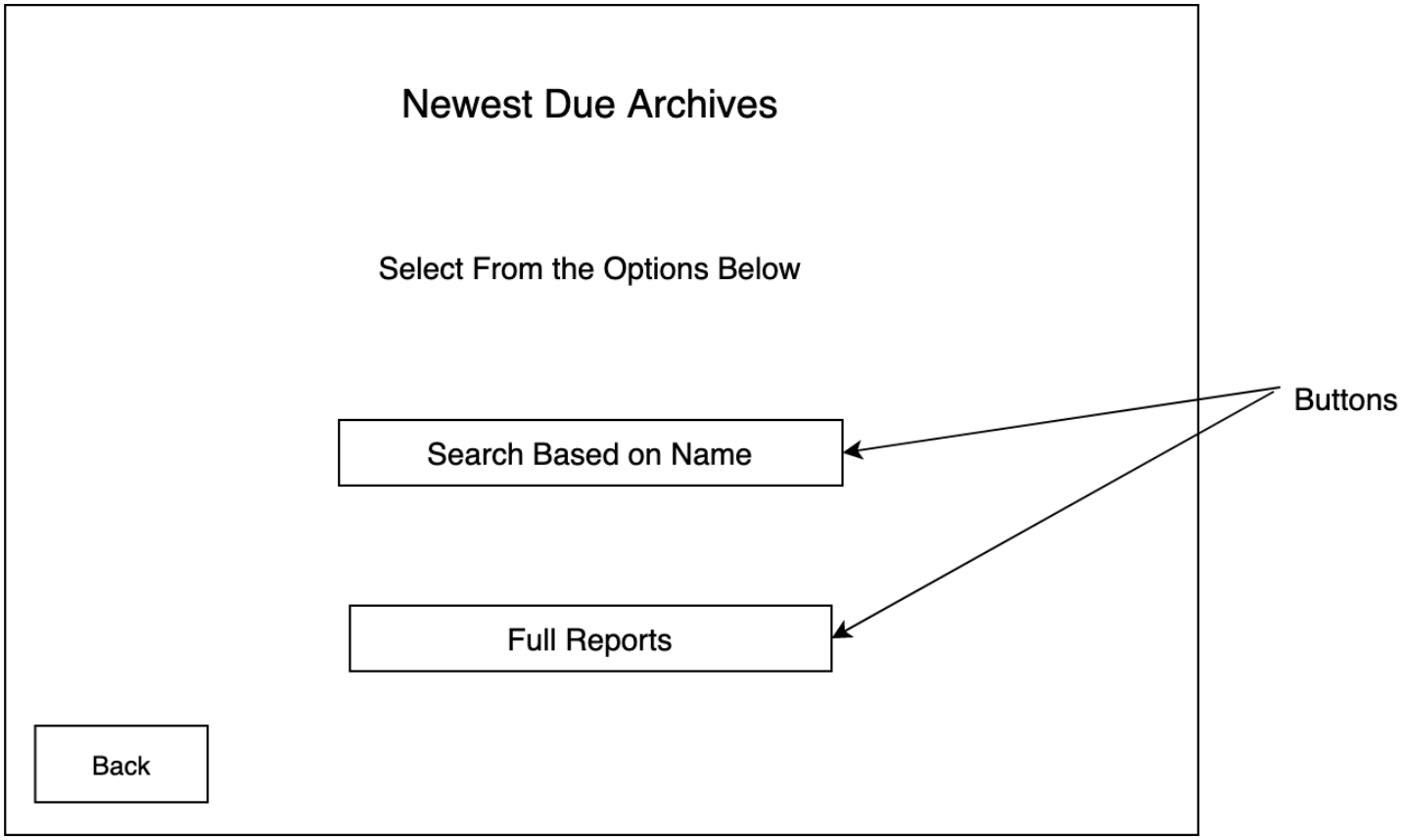
PastFullReportScene



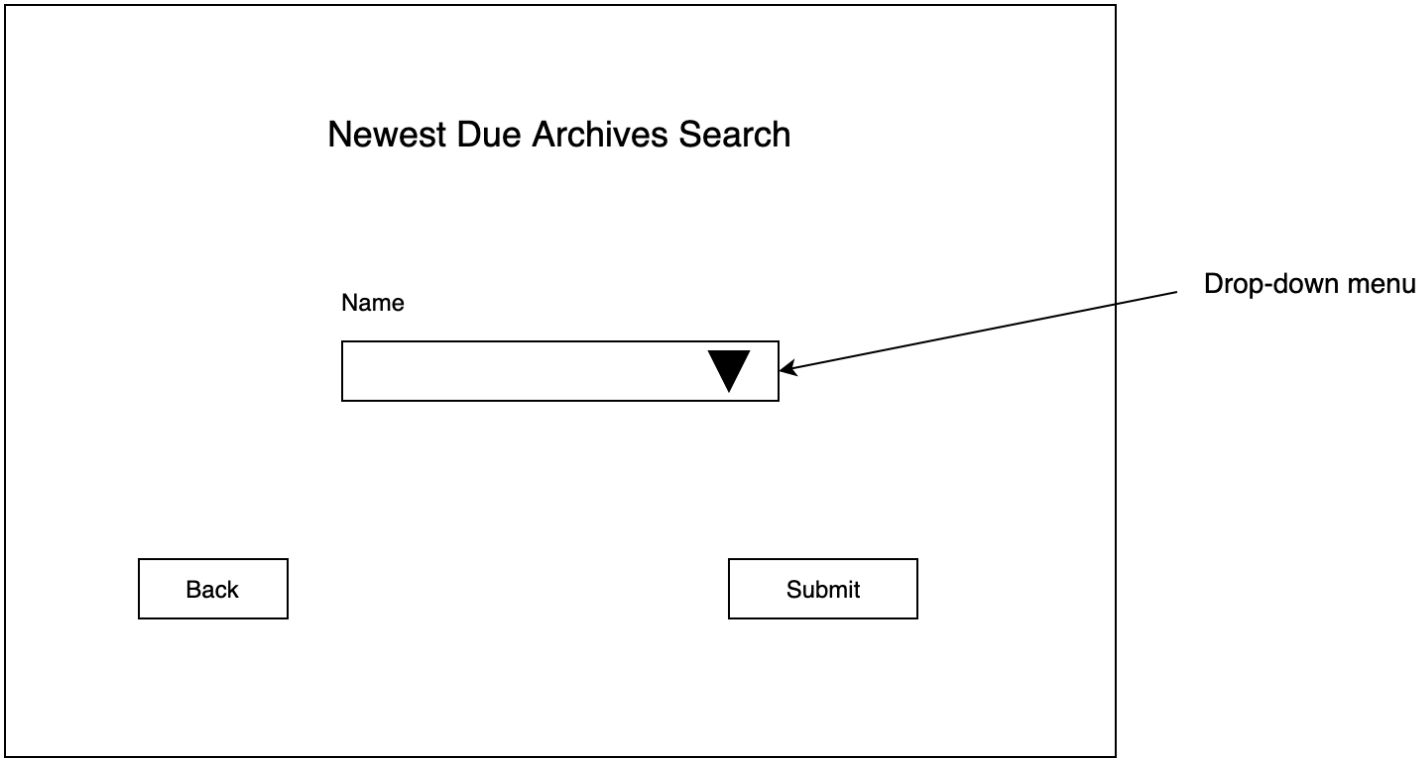
FullReportResultScene



NewDueArchiveScene



NewSearchScene



Test Plan

Sample Input / Test Case	Expected Outcome
On Launch Scene: "Enter Membership Info" button clicked	The scene transitions to display a screen with areas for the user to enter in the year and select their name from a drop-down menu. (Member Info Scene)
On Member Info Scene: "Bailey County Electric Cooperative Association" selected for the name, and "Submit" button clicked	The scene transitions to display a screen with areas for users to enter the number of meters, revenue, and cost of power for that member for a particular year. (Member Data Scene)
On Member Info Scene: Nothing selected for the name, and "Submit" button clicked	The Member Info Scene will show an error message that prompts users to select a name.
On Member Info Scene: "Back" button clicked	The scene returns to the Launch Scene.
On Member Data Scene: "a" entered for the number of meters, "a" entered for revenue, "a" entered for the cost of power, and "Submit" button clicked	The Member Data Scene will show an error message that prompts users to enter a valid number for the fields.
On Member Data Scene: "0" entered for the number of meters, "100000000000" entered for revenue, "1000000000" entered for the cost of power, and "Submit" button clicked	The scene transitions to one displaying the data that the user entered and providing warnings associated with that data. The scene will notify users that the number of meters entered is unrealistically small, and the revenue and cost of power are unrealistically large. (Confirmation Scene)
On Member Data Scene: "10000" entered for the number of meters, "32400000" entered for revenue, "20000000" entered for the cost of power, and "Submit" button clicked	The scene transitions to one displaying the data that the user entered. The scene will not show any warnings. (Confirmation Scene)
On Member Data Scene: "Back" button clicked	The scene returns to the Member Info Scene.
On Confirmation Scene: "Verify" button clicked	The scene will transition to a screen displaying a thank you message. (Thank You Scene)
On Confirmation Scene: "Change" button clicked	The scene returns to the Member Data Scene.
On Thank You Scene: "Main Menu" button clicked	The scene returns to the Launch Scene.
On Launch Scene: "View Past Due Archives" button clicked	The scene transitions to one displaying buttons for users to conduct different searches on the past due archives. (Past Due Archive Scene)
On Past Due Archive Scene: "Search based on name and year" button clicked	The scene transitions to one displaying an area for users to select a member name and the year the dues were. assessed (Past Search Scene)
On Past Due Archive Scene: "Back" button clicked	The scene returns to the Launch Scene.
On Past Search Scene: "Bartlett Electric Cooperative" and "2009" selected, "Submit" button clicked	The scene transitions to display the dues assessed for each of the due components for that particular member in the specified year. (Search Result Scene)
On Past Search Scene: Nothing selected for the year and/or name, "Submit" button clicked	The Past Search Scene will show an error message that prompts users to select a name and year.
On Past Search Scene: "Back" button clicked	The scene returns to the Past Due Archive Scene.

On Search Result Scene: "Main Menu" button clicked	The scene returns to the Launch Scene.
On Past Due Archive Scene: "Full Reports by Year" button clicked	The scene transitions to one with areas for the user to select the year of the full report they would like to view. (Past Full Reports Scene)
On Past Full Reports Scene: "2009" selected, "Submit" button clicked	The scene transitions to one displaying the dues assessed for each member in that year and the total amount of dues. (Full Report Result Scene)
On Past Full Reports Scene: Nothing selected for the year, "Submit" button clicked	The Past Full Reports Scene will show an error message that prompts users to select a year.
On Past Full Reports Scene: "Back" button clicked	The scene returns to the Past Due Archive Scene.
On Full Report Result Scene: "Main Menu" button clicked	The scene returns to the Launch Scene.
On Launch Scene: "View Newest Due Archive" button clicked	The scene transitions to one displaying buttons for users to conduct different searches on the newest due data. (New Due Archive Scene)
On New Due Archive Scene: "Search Based on Name" button clicked	The scene transitions to one displaying an area for users to select a member name. (New Search Scene)
On New Due Archive Scene: "Back" button clicked	The scene returns to the Launch Scene.
On New Search Scene: "Brazos Electric Power Cooperative" selected, "Submit" button clicked	The scene transitions to display the dues assessed for each of the due components for that particular member in the newest year. (Search Result Scene)
On New Search Scene: Nothing selected for the name, "Submit" button clicked	The New Search Scene will show an error message that prompts users to select a name.
On New Search Scene: "Back" button clicked	The scene returns to the New Due Archive Scene.
On New Due Archive Scene: "Full Report" button clicked	The scene transitions to one displaying the dues assessed for each member in the newest year and the total amount of dues. (Full Report Result Scene)

Word Count: 63