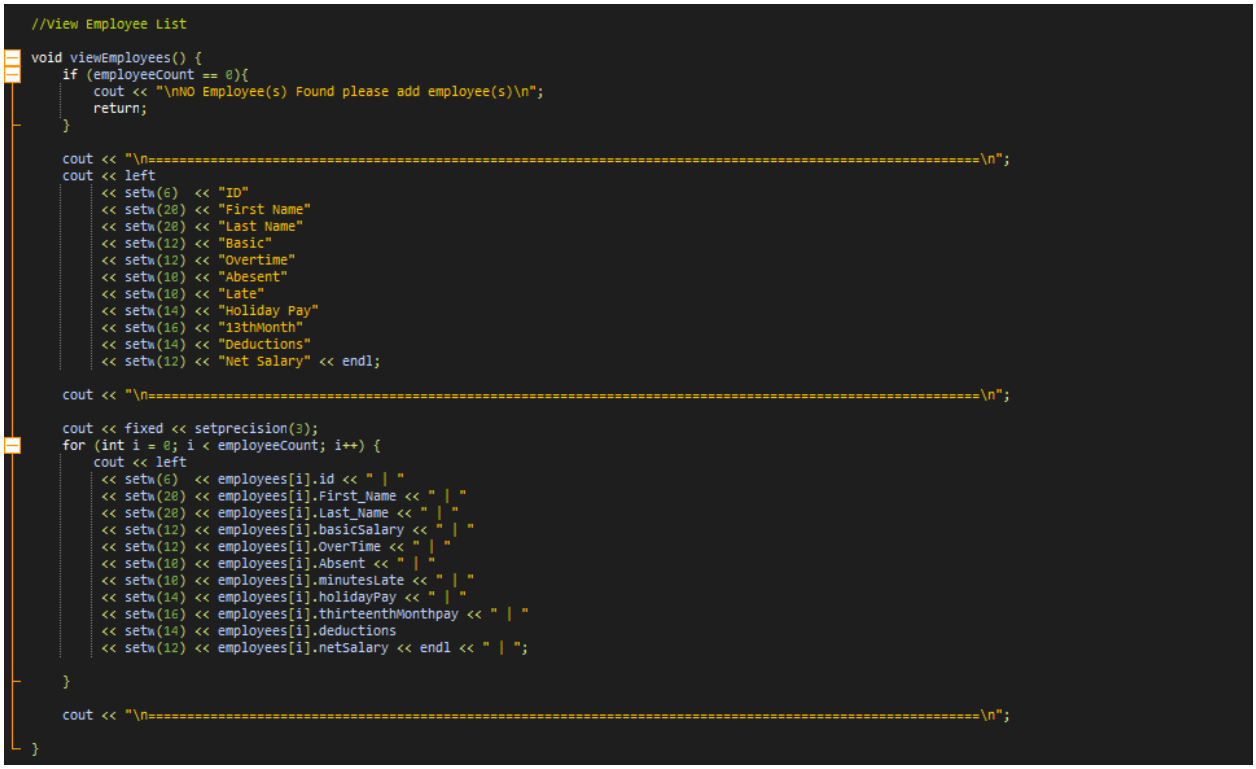
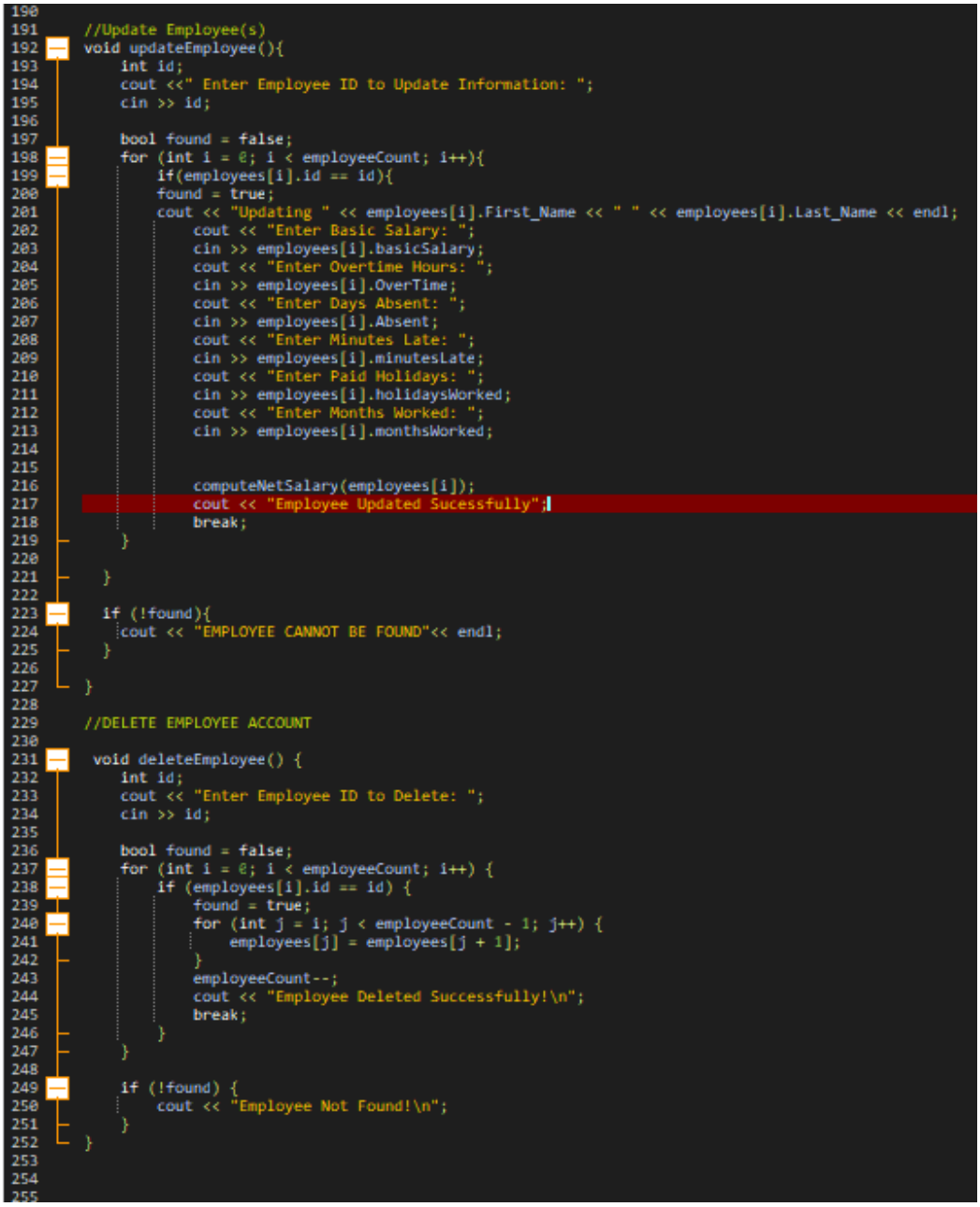
PROGRESS REPORT 2

1. FINISH the First Version of the system by Benj Ariel M. Ratcho



```
//View Employee List

void viewEmployees() {
    if (employeeCount == 0){
        cout << "\nNO Employee(s) Found please add employee(s)\n";
        return;
    }

    cout << "\n=============================================================================================================\n";
    cout << left
         << setw(6)  << "ID"
         << setw(20) << "First Name"
         << setw(20) << "Last Name"
         << setw(12) << "Basic"
         << setw(12) << "Overtime"
         << setw(10) << "Abesent"
         << setw(10) << "Late"
         << setw(14) << "Holiday Pay"
         << setw(16) << "13thMonth"
         << setw(14) << "Deductions"
         << setw(12) << "Net Salary" << endl;

    cout << "\n=============================================================================================================\n";

    cout << fixed << setprecision(3);
    for (int i = 0; i < employeeCount; i++) {
        cout << left
         << setw(6)  << employees[i].id << " | "
         << setw(20) << employees[i].First_Name << " | "
         << setw(20) << employees[i].Last_Name << " | "
         << setw(12) << employees[i].basicSalary << " | "
         << setw(12) << employees[i].OverTime << " | "
         << setw(10) << employees[i].Absent << " | "
         << setw(10) << employees[i].minutesLate << " | "
         << setw(14) << employees[i].holidayPay << " | "
         << setw(16) << employees[i].thirteenthMonthpay << " | "
         << setw(14) << employees[i].deductions
         << setw(12) << employees[i].netSalary << endl << " | ";

    }

    cout << "\n=============================================================================================================\n";

}
```



```
190
191     //Update Employee(s)
192     void updateEmployee(){
193         int id;
194         cout <<" Enter Employee ID to Update Information: ";
195         cin >> id;
196
197         bool found = false;
198         for (int i = 0; i < employeeCount; i++){
199             if(employees[i].id == id){
200             found = true;
201             cout << "Updating " << employees[i].First_Name << " " << employees[i].Last_Name << endl;
202                 cout << "Enter Basic Salary: ";
203                 cin >> employees[i].basicSalary;
204                 cout << "Enter Overtime Hours: ";
205                 cin >> employees[i].OverTime;
206                 cout << "Enter Days Absent: ";
207                 cin >> employees[i].Absent;
208                 cout << "Enter Minutes Late: ";
209                 cin >> employees[i].minutesLate;
210                 cout << "Enter Paid Holidays: ";
211                 cin >> employees[i].holidaysWorked;
212                 cout << "Enter Months Worked: ";
213                 cin >> employees[i].monthsWorked;
214
215
216                 computeNetSalary(employees[i]);
217                 cout << "Employee Updated Sucessfully";
218                 break;
219         }
220
221       }
222
223       if (!found){
224         cout << "EMPLOYEE CANNOT BE FOUND"<< endl;
225       }
226
227     }
228
229     //DELETE EMPLOYEE ACCOUNT
230
231      void deleteEmployee() {
232         int id;
233         cout << "Enter Employee ID to Delete: ";
234         cin >> id;
235
236         bool found = false;
237         for (int i = 0; i < employeeCount; i++) {
238             if (employees[i].id == id) {
239                 found = true;
240                 for (int j = i; j < employeeCount - 1; j++) {
241                     employees[j] = employees[j + 1];
242                 }
243                 employeeCount--;
244                 cout << "Employee Deleted Successfully!\n";
245                 break;
246             }
247         }
248
249         if (!found) {
250             cout << "Employee Not Found!\n";
251         }
252     }
253
254
255
```

Draft Of The Flow Chart
Created by: Will Stuart and Kherwin Milan