

Membership Manager Presentation

D'où vient l'idée ?

L'idée découle d'un réel problème rencontré par un proche des membres du groupe, qui par la reprise d'une salle de sport s'est retrouvé avec une charge administrative élevée.

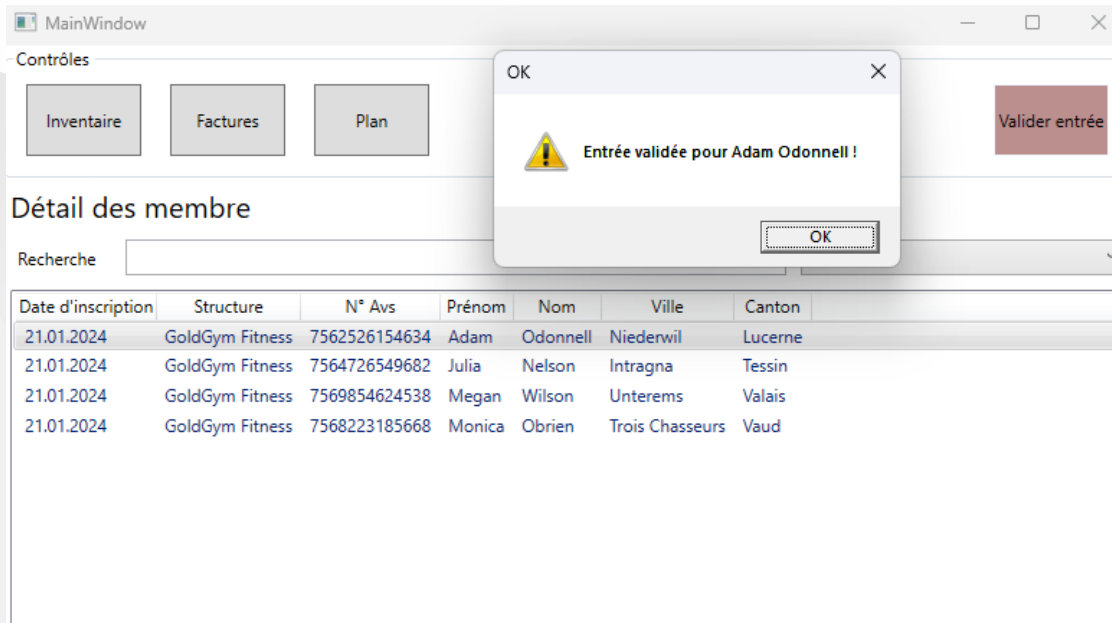
Quels étaient les problèmes à résoudre ?

- Le problème est que la gestion des membres est une tâche répétitive et chronophage.
- La gestion des entrées et abonnements prend du temps à chaque entrée de membre.
- Le contrôle de la facturation sans aide informatique est une tâche difficile et source d'erreur.

Objectifs de l'application ?

Validation des entrées

Lors de l'arrivée d'un membre, l'application permet de saisir rapidement le nom de membre et de valider son entrée.



Gestion du compte du membre

Une interface permet de gérer le compte du membre, de voir ses entrées, ses abonnements et de générer des factures.

AccountDetailWindows

Account Detail

Information du membre

Nom: Adam Odonnell

Abonné jusqu'au : 21.07.2025

N° AVS : 7562526154634

N° entrée : 21

Information du compte

Solde -868,00

En suspens 868,00

Ajouter un paiement

Ajouter une consommation

Facturation

Acheter des entrées

Date	Description	Amount
21.01.2024	Paie ment du: 21.01.2024	90,00
21.01.2024	Paie ment du: 21.01.2024	432,00
21.01.2024	Paie ment du: 21.01.2024	10,00
26.08.2023	Paie ment du: 26.08.2023	90,00
22.07.2023	Paie ment du: 22.07.2023	5,45
21.02.2023	Coca Cola Classic	-2,00
24.02.2023	Pepsi	-1,85
22.07.2023	Kit Kat	-1,60
25.08.2023	10

Gestion des abonnements et entrées

L'application permet d'ajouter un plan d'abonnement ou d'entrée au membre.

En suspens 868,00

Acheter des entrées

- 1 entrées 10 CHF
- 10 entrées 90 CHF
- 6 mois 432 CHF
- 12 mois 778 CHF
- 3 mois 240 CHF

AccountDetailWindows

Account Detail

Information du membre

Nom: Adam Odonnell Abonné jusqu'au : 21.07.2025

N° AVS : 7562526154634 N° entrée : 21

Information du compte

Solde -868,00

En suspens 868,00

Ajouter un paiement Ajouter une consommation Facturation Acheter des entrées 10 entrées 90

Date	Description	Montant
21.01.2024	Paielement du: 21.01.2024	90
21.01.2024	Paielement du: 21.01.2024	432
21.01.2024	Paielement du: 21.01.2024	10
26.08.2023	Paielement du: 26.08.2023	90
22.07.2023	Paielement du: 22.07.2023	5,00
21.02.2023	Coca Cola Classic	-2,00
24.02.2023	Pepsi	-1,85
22.07.2023	Kit Kat	-1,60
25.08.2023	10 entrées 90 CHF	

Confirmation

?

Voulez-vous acheter 10 entrées 90 CHF ?

Oui Non

Gestion des paiements

L'application permet de saisir des paiements ou de générer des factures.

The screenshot displays three overlapping windows from the 'AccountDetailWindows' application:

- PaierementDetailWindows (left):** A window for entering payment details. It includes fields for 'Membre' (Adam Odonnell), 'Date du paiement' (22.01.2024), and 'Montant' (50.00). At the bottom are 'Enregistrer' and 'Annuler' buttons.
- Account Detail (center):** The main window showing account information. It is divided into 'Information du membre' and 'Information du compte'.
 - Information du membre:** Nom: Adam Odonnell, Abonné jusqu'à: 21.01.2026, N° AVS: 7562526154634, N° entrée: 21.
 - Information du compte:** Solde: -1 300,00, En suspens: 868,00.Below this is a table of transactions and buttons for 'Ajouter un paiement', 'Ajouter une consommation', and 'Acheter des entrées'.
- Information dialog (center):** A small pop-up window with an information icon and the text 'Facture générée', with an 'OK' button.

Date	Description	Amount
21.01.2024	Paierement du: 21.01.2024	90,00
21.01.2024	Paierement du: 21.01.2024	432,00
21.01.2024	Paierement du: 21.01.2024	10,00
26.08.2023	Paierement du: 26.08.2023	90,00
22.07.2023	Paierement du: 22.07.2023	5,45
21.02.2023	Coca Cola Classic	-2,00
24.02.2023	Pepsi	-1,85
22.07.2023	Kit Kat	-1 60

Gestion des factures

Cette partie permet de suivre les factures générées selon leurs états et de la marquer comme payées si tel est le cas.

Les factures peuvent être triées par date, par membre ou par état.

BillDashBoard

Factures

Informations

Nom: -

N° AVS : -

Tri

☐ par membres

Sélectionner

☐ par status

☐ par date

Sélectionner un

15

Sélectionner un

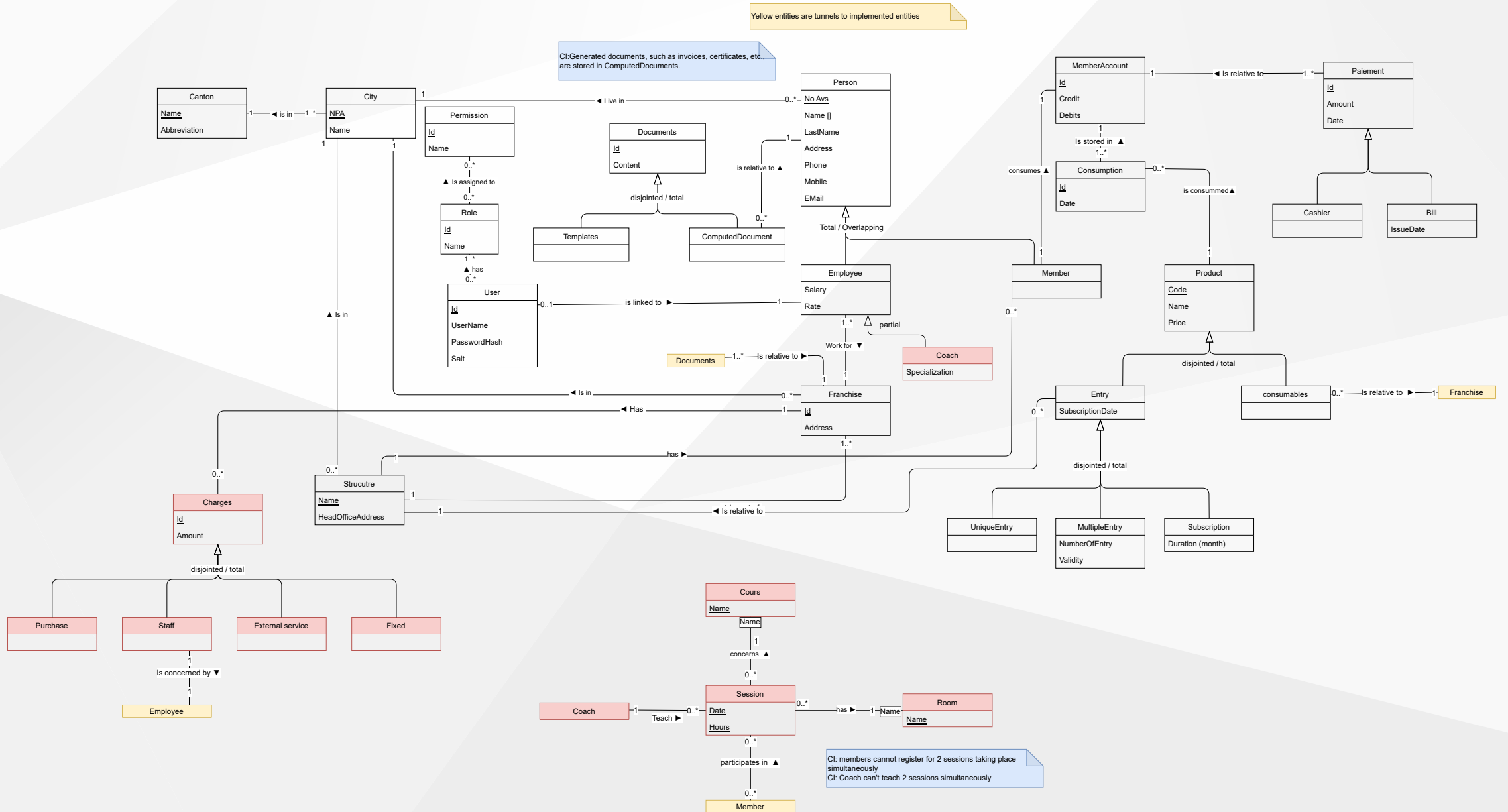
15

Contrôle

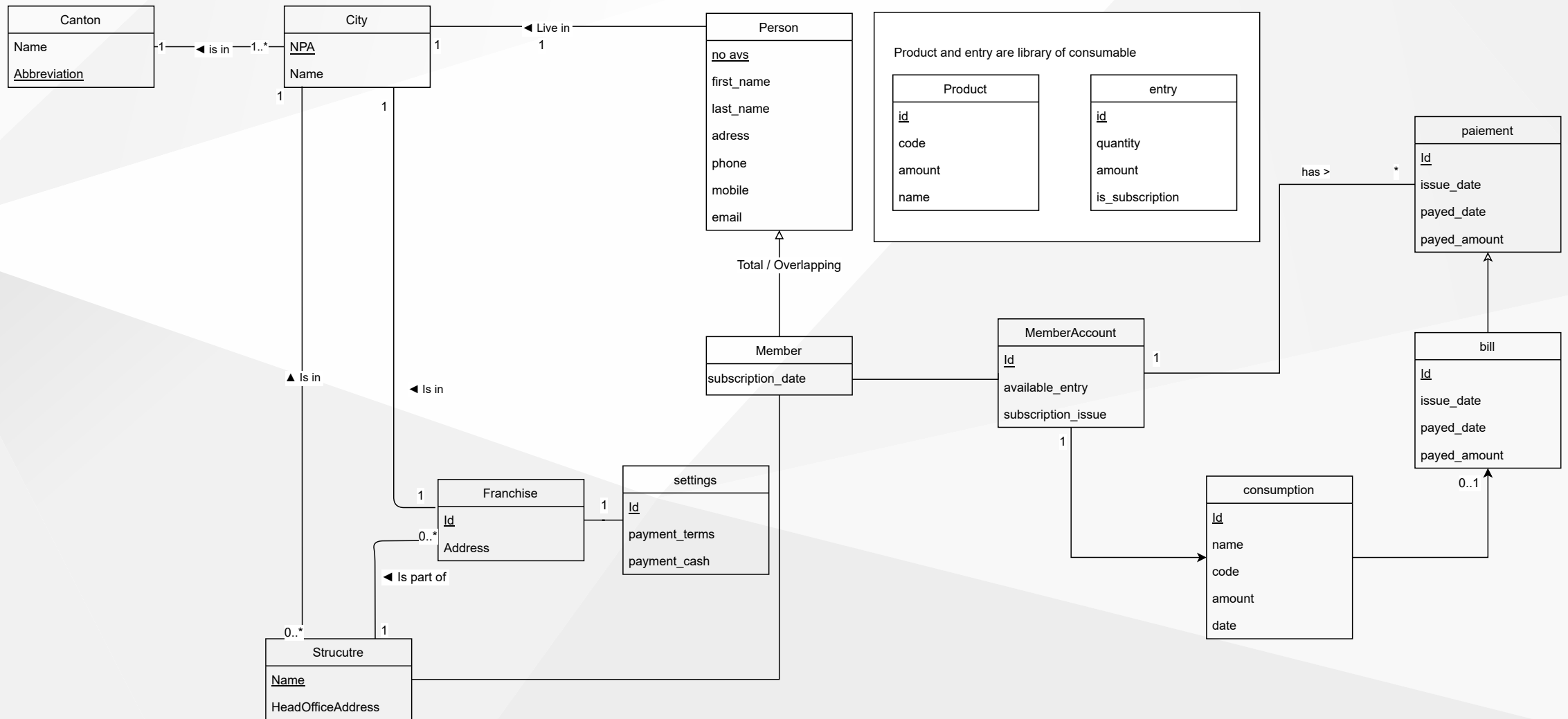
Date	Numero de facture	Destinataire	Montant	Montant Payé	Status	Date d'échéance	Date de paiement
22.01.2024	202412	Adam Odonnell	432,00	0,00	Attente	21.02.2024	
21.01.2024	20248	Adam Odonnell	432,00	432,00	Payée	20.02.2024	21.01.2024
21.01.2024	20249	Adam Odonnell	90,00	90,00	Payée	20.02.2024	21.01.2024
21.01.2024	202410	Adam Odonnell	90,00	0,00	Attente	20.02.2024	
21.01.2024	202411	Adam Odonnell	778,00	0,00	Attente	20.02.2024	
21.01.2024	Abonnement	12 mois 778 CHF	-778,00				
<div><div>Payée</div><div>Supprimer la facture</div></div>							
20.12.2023	20233	Julia Nelson	781,60	0,00	Expirée	19.01.2024	
26.08.2023	20232	Adam Odonnell	90,00	90,00	Payée	25.09.2023	19.09.2023

Implémentation

Structure du projet - Base



Structure du projet - implémenté



Technologies utilisées

- C#
- WPF - XAML

Interactions avec la base de données - Requête simple

```
public interface ISql
{
    public void Insert();
    public void Update();
    public abstract static void Delete(params object[] pk);
    public abstract static ISql? Select(params object[] pk);
    public bool Validate();
}
```

Interactions avec la base de données - Requête personnalisée

bill.cs

```
public new static List<SqlViewable>? Views(params NpgsqlParameter[] sqlParam)
{
    ...

    NpgsqlCommand cmd = new();

    StringBuilder SqlQuery = new(@"SELECT bill.id, bill.issue_date, bill.payed_date,
                                   bill.payed_amount, paiement.payed, paiement.account_id,
                                   amount, date, person.first_name, person.last_name
                                   FROM Bill
                                   LEFT JOIN paiement ON bill.id = paiement.id
                                   LEFT JOIN memberaccount ON paiement.account_id = memberaccount.id
                                   LEFT JOIN person ON memberaccount.id = person.no_avs");

    if (sqlParam.Length == 2)
    {
        SqlQuery.Append(" WHERE paiement.account_id = @id AND Payed = @payed");
        cmd.Parameters.AddRange(sqlParam);
    }
    else if (sqlParam.Length == 1 && sqlParam[0].ParameterName.Equals("@id"))
    {
        SqlQuery.Append(" WHERE paiement.account_id = @id");
        cmd.Parameters.Add(sqlParam[0]);
    }
    else if (sqlParam.Length == 1 && sqlParam[0].ParameterName.Equals("@payed"))
    {
        SqlQuery.Append(" WHERE Payed = @payed");
        cmd.Parameters.Add(sqlParam[0]);
    }
    ...
}
```