



**POLYTECHNIQUE  
MONTRÉAL**

UNIVERSITÉ  
D'INGÉNIERIE

École polytechnique de Montréal

Département de génie informatique  
et de génie logiciel

# **INTRODUCTION A L'INGÉNIERIE LOGICIELLE LOG1000**

Automne 2019

**TRAVAIL PRATIQUE 5 : Restructuration de code source**

**PAR : MOUSSA FOFANA 1955968**

**LOMPO AUGUSTIN 1927045**

## **E1) Une mauvaise odeur dans les attributs**

### **1) Les odeurs qui se cachent**

La classe AppointmentData a très peu de responsabilité dans la classe Appointments et n'a aucune responsabilité prévu ailleurs dans le programme. Ceci rend difficile le

compréhension du code et peut provoquer des fuite de mémoire.

## 2) La restructuration nécessaire

La restructuration nécessaire pour enlever cette odeur du code est : **inline classe**.

## 3) Les Étapes a suivre sont dans le tableau suivant

Étape	Description
Déplacement d'attribut	On déplace l'attribut <code>_data</code> de la classe <code>AppointmentData</code> dans la classe <code>Appointments</code> on remplace ainsi l'attribut <code>AppointmentData</code>
Déplacement des méthodes	On ajoute les méthodes de la classe « <code>AppointmentData</code> » aux méthode de la classe « <code>Appointments</code> »
Remplace tous les objets de la classe « <code>AppointmentsData</code> »	On remplace tous les objets de la classe « <code>AppointmentsData</code> » par l'attribut <code>_ppointmentsData</code> de la classe « <code>Appointments</code> »
On supprime la classe « <code>AppointmentsData</code> »	On supprime le fichier « <code>AppointmentsData.hpp</code> » et « <code>AppointmentsData.cpp</code> » car n'a plus d'utilité

## 4) Les modifications effectuées dans les classes

Modifications dans « `AppointmentsData.hpp` »

```

#pragma once

#include <string>
#include <vector>
#include <algorithm>
#include <iomanip>
#include <numeric>

#include "Constants.hpp"
#include "CSVHandler.hpp"
// #include "AppointmentsData.hpp" // devenu inutile

class Appointments
{
public:
    Appointments(std::string filePath);

    // la méthode déplacée depuis la AppointmentsData
    std::vector<std::vector<std::string>>* getAppointmentsData();
    void setAppointmentsData(std::vector<std::vector<std::string>> appointmentsData);

    void display(unsigned int userCol, std::string userId, std::vector<unsigned int> displayedColumns = {});
    bool schedule(std::string patientId, std::string nurseId);

private:
    std::vector<std::string> createNewAppointment(std::string patientId, std::string nurseId);
    // l'attribut est remplacé
    std::vector<std::vector<std::string>> _appointmentsData;
    std::vector<std::string> _headers;
    CSVHandler _csvHandler;

    const unsigned int _COLUMN_WIDTH = 14;
};

```

Modifications dans « AppointmentsData.cpp »

```

57
58 bool Appointments::schedule(std::string patientId, std::string nurseId)
59 {
60     std::vector<std::string> newAppointment = this->createNewAppointment(patientId, nurseId);
61
62     for (std::vector<std::string> appointment : (this->_appointmentsData)) //plus besoin de getData
63     {
64         bool isOccupied = newAppointment[APPOINTMENT_DATE] == appointment[APPOINTMENT_DATE] &&
65         newAppointment[APPOINTMENT_TIME] == appointment[APPOINTMENT_TIME];
66         if (isOccupied)
67         {
68             return false; // Schedule did not work
69         }
70     }
71     // Update appointment list and file.
72     this->_appointmentsData.push_back(newAppointment);
73     this->_csvHandler.writeData(newAppointment);
74     return true;
75 }
76
77 //    getter de nouvel attribut _appointmentsData
78
79 std::vector<std::vector<std::string>>*> Appointments::getAppointmentsData(){
80
81     return &this->_appointmentsData;
82 }
83 //    setter de nouvel attribut _appointmentsData
84 void Appointments::setAppointmentsData(std::vector<std::vector<std::string>> appointmentsData){
85     this->_appointmentsData=appointmentsData;
86 }
87

```

## 5) Le résultats de la compilation

```

Authentification du patient
Veuillez inscrire votre ID d'utilisateur: jo411

Bonjour John Doe!

Voici la liste des infirmières:

  UserID  FirstName  LastName  Email  Telephone  StreetNumber  StreetName  City  PostalCode  HourlyRate  LicenseNumber
1.  mclouis    Louis    McCartney louis.mccartn  514-511-1111  1232  3e av.  Montreal  H1T 2V5  30  93202012
2.  manononon  Manon    Lepage manon.lepage@  514-512-1111  213  4e av.  Montreal  H3T 1P1  38  83604319

Veuillez choisir parmi une des options suivantes:
  0. Quitter.
  1. Consulter les disponibilités d'un.e infirmier.e.
  2. Prendre rendez-vous avec un.e infirmier.e
  3. Voir mes rendez-vous.

[0, 1, 2 ou 3]: 2

Veuillez sélectionner un.e infirmier.e par son index: 2

Veuillez indiquer la date du rendez-vous [jj/mm/aaaa]: 02/12/2019

Veuillez indiquer le moment du rendez-vous [AM/PM]: PM

Veuillez indiquer la raison du rendez-vous: consultation grippe

Votre rendez-vous a été correctement enregistré
Appuyez sur ENTER pour continuer...Voici la liste des infirmières:

  UserID  FirstName  LastName  Email  Telephone  StreetNumber  StreetName  City  PostalCode  HourlyRate  LicenseNumber
1.  mclouis    Louis    McCartney louis.mccartn  514-511-1111  1232  3e av.  Montreal  H1T 2V5  30  93202012
2.  manononon  Manon    Lepage manon.lepage@  514-512-1111  213  4e av.  Montreal  H3T 1P1  38  83604319

Veuillez choisir parmi une des options suivantes:
  0. Quitter.
  1. Consulter les disponibilités d'un.e infirmier.e.
  2. Prendre rendez-vous avec un.e infirmier.e
  3. Voir mes rendez-vous.

```

## 6) Commit des résultats

```

[gigl@fedora-gigl TP5]$ git add .
[gigl@fedora-gigl TP5]$ git commit -m "J'avais oublié de faire un capture pour le tout premier commit "
[master fc956a7] J'avais oublié de faire un capture pour le tout premier commit
 3 files changed, 23 insertions(+), 55 deletions(-)
[gigl@fedora-gigl TP5]$ git push
Username for 'https://github.com': fomou
Password for 'https://fomou@github.com':
Énumération des objets: 13, fait.
Décompte des objets: 100% (13/13), fait.
Compression des objets: 100% (7/7), fait.
Écriture des objets: 100% (7/7), 766 bytes | 766.00 KiB/s, fait.
Total 7 (delta 6), réutilisés 0 (delta 0)
To https://github.com:fomou/polymtl.ca
cd672ad..fc956a7 master -> master
[gigl@fedora-gigl TP5]$

```

## E2) Une mauvaise odeur dans les méthodes

### 1) Les odeurs

Les odeurs présentes dans la méthode « void Appointments::display() » est :

- **Code dupliqué**

Ceci rend la compréhension du code difficile.

2) Pour enlever cette odeur nous allons faire une **Extraction de méthode**

3) Les étapes sont dans le tableau suivant

Étapes	Description
Créer la méthode filteredAppointments	On crée cette méthode qui va renvoyer une liste filtrée des rendez-vous d'un patient qu'elle prend en paramètre
Créer la méthode fullArray	Cette méthode permet de remplir le tableau dispLEDColumn
Créer la méthode displayHeader	Cette méthode va permettre d'afficher la colonne d'entête
Créer printAppointments	Une méthode va afficher une liste de RDV qu'elle prend en paramètre

4) Les modifications apportées

Ajout des nouvelles méthodes dans le fichier « appointments.hpp »

```

1  #pragma once
2
3  #include <string>
4  #include <vector>
5  #include <algorithm>
6  #include <iomanip>
7  #include <numeric>
8
9  #include "Constants.hpp"
10 #include "CSVHandler.hpp"
11 // #include "AppointmentsData.hpp" // devenu inutile
12
13 class Appointments
14 {
15 public:
16     Appointments(std::string filePath);
17
18     // la méthode déplacée depuis la AppointmentsData
19     std::vector<std::vector<std::string>>* getAppointmentsData();
20     void setAppointmentsData(std::vector<std::vector<std::string>> appointmentsData);
21
22     // On crée cette méthode qui va renvoyer une liste filtrée des rendez-vous d'un patient qu'elle prend en paramètre
23     std::vector<std::vector<std::string>> filteredAppointments(unsigned int userCol, std::string userId);
24
25     // Cette méthode va permettre d'afficher la colonne d'entête
26     void displayHeaderColumn(std::vector<unsigned int> headerColumn={});
27
28     // Cette méthode permet de remplir le tableau displayedColumn
29     void fullArray(std::vector<unsigned int>& array );
30
31     // Une méthode va afficher une liste de RDV qu'elle prend en paramètre
32     void printAppointments(std::vector<std::vector<std::string>> appointments, std::vector<unsigned int> column={});
33
34     void display(unsigned int userCol, std::string userId, std::vector<unsigned int> displayedColumns = {});
35     bool schedule(std::string patientId, std::string nurseId);
36
37 private:
38     std::vector<std::string> createNewAppointment(std::string patientId, std::string nurseId);
39     // l'attribut est remplacé
40     std::vector<std::vector<std::string>> _appointmentsData;
41     std::vector<std::string> _headers;
42     CSVHandler _csvHandler;
43
44     const unsigned int _COLUMN_WIDTH = 14;
45 };
46

```

Implémentation des nouvelles méthodes dans  
« Appointments.cpp »



```

std::vector<std::vector<std::string>> Appointments::filteredAppointments(unsigned int userCol, std::string userId){
    std::vector<std::vector<std::string>> filteredArray;
    std::copy_if(
        this->_appointmentsData.begin(),
        this->_appointmentsData.end(),
        std::back_inserter(filteredArray),
        [userId, userCol](std::vector<std::string> appointment) {
            return appointment[userCol] == userId;
        });
    return filteredArray;
}

void Appointments::fullArray(std::vector<unsigned int> array ){
    if (array.empty())
    {
        array = std::vector<unsigned int>(this->_headers.size());
        std::iota(array.begin(), array.end(), 0);
    }
}

void Appointments::displayHeaderColumn(std::vector<unsigned int> headerColumn){
    for (unsigned int col : headerColumn)
    {
        std::cout << std::setw(this->_COLUMN_WIDTH)
            << this->_headers[col];
    }
    std::cout << std::endl
        << std::string(headerColumn.size() * this->_COLUMN_WIDTH, '_')
        << std::endl;
}

```

```

void Appointments::printAppointments(std::vector<std::vector<std::string>> appointments, std::vector<unsigned int> column){
    unsigned int appointmentIdx = 1;

    for (std::vector<std::string> appointment : appointments)
    {
        std::cout << appointmentIdx++
            << ' '
            << std::setw(this->_COLUMN_WIDTH - 2);

        for (unsigned int col : column)
        {
            std::cout << appointment[col].substr(0, this->_COLUMN_WIDTH - 1)
                << std::setw(this->_COLUMN_WIDTH);
        }

        std::cout << '\n';
    }
}

```

Le nouveau visage de la méthode Appointments ::display()

```

void Appointments::display(unsigned int userCol, std::string userId, std::vector<unsigned int> displayedColumns)
{
    // 1. On remplit le vecteur displayedColumns par tous les indexes des colonnes du fichier CSV:
    fullArray(displayedColumns);
    // 2. On affiche toutes les en-têtes des colonnes voulues du fichier CSV:
    displayHeaderColumn(displayedColumns);
    // 3. On filtre les rendez-vous pour ne conserver ceux de l'utilisateur passé en paramètre (soit un infirmier ou un patient):
    printAppointments(filteredAppointments(userCol,userId),displayedColumns);
    // 4. On affiche ensuite les rendez-vous filtrés à l'écran:
}

```

## 5) Compilation et Exécution

Exemple : voir mes rendez-vous

```

Bonjour John Doe!

Voici la liste des infirmières:

```

	UserID	FirstName	LastName	Email	Telephone	StreetNumber	StreetName	City	PostalCode	HourlyRate	LicenseNumber
1.	mclouis	Louis	Mccartney	louis.mccartn	514-511-1111	1232	3e av.	Montreal	H1T 2V5	30	93202012
2.	manononon	Manon	Lepage	manon.lepage@	514-512-1111	213	4e av.	Montreal	H3T 1P1	38	83604319

```

Veuillez choisir parmi une des options suivantes:
0. Quitter.
1. Consulter les disponibilités d'un.e infirmier.e.
2. Prendre rendez-vous avec un.e infirmier.e
3. Voir mes rendez-vous.

# [0, 1, 2 ou 3]: 3
Voici vos rendez-vous à venir:

```

	NurseUserID	PatientUserID	Date	AM/PM	Reason
1.	manononon	jo411	23/04/2019	AM	tousse
2.	mclouis	jo411	22/11/2019	PM	grippe
3.	mclouis	jo411	29/12/2019	AM	ennuie
4.	manononon	jo411	02/12/2019	PM	consultation

```

Appuyez sur ENTER pour continuer...
Voici la liste des infirmières:

```

## 6) Commit des modifications

```

[git@fedora-gigl TP5]$ git add .
[git@fedora-gigl TP5]$ git commit -m "la restructuaction de la methode Appointments::display"
[master cd672ad] la restructuaction de la methode Appointments::display
9 files changed, 0 insertions(+), 0 deletions(-)
delete mode 100755 TP5/prototype.out
delete mode 100644 TP5/src/Appointments.o
delete mode 100644 TP5/src/CSVHandler.o
delete mode 100644 TP5/src/Nurses.o
delete mode 100644 TP5/src/Utils.o
delete mode 100644 TP5/src/main.o
delete mode 100755 TP5/test.out
delete mode 100644 TP5/tests/Test.o
delete mode 100644 TP5/tests/main.o
[git@fedora-gigl TP5]$ git push
Username for 'https://github.com': fomou
Password for 'https://fomou@github.com':
Énumération des objets: 28, fait.
Décompte des objets: 100% (28/28), fait.
Compression des objets: 100% (21/21), fait.
Écriture des objets: 100% (21/21), 748.19 KiB | 2.15 MiB/s, fait.
Total 21 (delta 10), réutilisés 0 (delta 0)
To https://github.com/fomou/TP5
07a71c1..cd672ad master -> master

```

### E3) Utilisation des variables

- 1) Calcul de span , de la Portée et la durée de vie des variables choix et loggedInUser

Les deux variables ont la même portée qui correspond à la fonction main

**Span(choix)**=[(ligne50-ligne31-1)+(ligne52-ligne50-1)+(ligne105-ligne50-1)]/3=24

**Durée de vie (choix)**= 31@105

**Portée(choix)**= 30@109

**Span(loggedInUser)**=[ligne36-ligne32-1)+(ligne37-ligne36-1)+(ligne74-ligne37-1)+(ligne97-ligne74-1)+(ligne107-ligne97-1)]/3 = 14

**Durée de vie(loggedInUser)**= 32@107

**Portée(loggedInUser)** = 30@109

## 2) Interprétation des résultats précédent

Le span et la durée de vie des deux variables sont grands .Il faudrait restructurer le code afin de palier a cette odeur.

La variable **choix** gagnerait une restructuration

## 3) Les étapes de la restructuration

Étapes	Description
Réduire le span	On va rapprocher les utilisations de la variable choice pour son span
Réduire la durée de vie	On va remplacer la boucle do-while par une boucle while pour rapprocher la déclaration et la dernière utilisation de la variable

#### 4) Les modifications effectuées au main

```
int main(int argc, char **argv)
{
    Patient *loggedInUser = login();
    Nurses nurses(NURSES_CSV_PATH);
    Appointments appointments(APPOINTMENTS_CSV_PATH);

    // Les codes ANSI sont utilisés pour mettre le texte en gras.
    std::cout << "\n\n\e[1mBonjour " << loggedInUser->firstName << ' '
               << loggedInUser->lastName << "\e[0m\n\n";

    while(true)
    {
        std::cout << "Voici la liste des infirmières:\n\n";
        nurses.display();

        std::cout << "\nVeuillez choisir parmi une des options suivantes: \n"
                  << "\t0. Quitter.\n"
                  << "\t1. Consulter les disponibilités d'un.e infirmier.e.\n"
                  << "\t2. Prendre rendez-vous avec un.e infirmier.e.\n"
                  << "\t3. Voir mes rendez-vous.\n"
                  << std::endl
                  << "[0, 1, 2 ou 3]: ";

        int choice;
        std::cin >> choice;
        if(choice==0){
            break; //arrete la boucle lorsqu'on entre 0 au clavier
        }
        // Switch case pour le choix de l'utilisateur :D !
        switch (choice)
        {
            // Ceci est le cas #1. Il s'agit d'afficher
            // la liste de rendez-vous pour l'utilisateur courant!
            case 1:
            {
                // On affiche la liste de rendez-vous pour l'utilisateur courant.
                std::string nurseID = nurses.select()[NURSES_USER_ID];
                std::cout << "Il est impossible de prendre rendez-vous avec cette infirmière pour les dates suivantes:\n";
                appointments.display(APPOINTMENT_NURSE_ID, nurseID, {APPOINTMENT_DATE, APPOINTMENT_TIME});
            }
        }
    }
}
```

```

case 1:
{
    // On affiche la liste de rendez-vous pour l'utilisateur courant.
    std::string nurseID = nurses.select()[NURSES_USER_ID];
    std::cout << "Il est impossible de prendre rendez-vous avec cette infirmière pour les dates suivantes:\n";
    appointments.display(APPOINTMENT_NURSE_ID, nurseID, {APPOINTMENT_DATE, APPOINTMENT_TIME});

    // Attend que l'utilisateur appuie sur ENTER pour continuer l'exécution.
    std::cout << "\nAppuyez sur ENTER pour continuer...";
    std::cin.ignore();
    std::cin.get();

    break;
}
// Ceci est le cas #2. Il s'agit de permettre
// à l'utilisateur de prendre un rendez-vous!
case 2:
{
    // Prend un rendez-vous pour l'utilisateur.
    std::string nurseID = nurses.select()[NURSES_USER_ID];
    if (appointments.schedule(loggedInUser->profileID, nurseID))
    {
        std::cout << "\nVotre rendez-vous a été correctement enregistré";
    }
    else
    {
        std::cout << "\nCet.te infirmier.e est indisponible pour cette période.";
    }

    // Attend que l'utilisateur appuie sur ENTER pour continuer l'exécution.
    std::cout << "\nAppuyez sur ENTER pour continuer...";
    std::cin.ignore();
    std::cin.get();

    break;

    int s;
    std::cout << "DEBUG3.";
    std::cin >> s;
}

```

```

// Ceci est le cas #3. Il s'agit de permettre à
// l'utilisateur de voir les rendez-vous qu'il a pris!
case 3:
{
    // Montre la liste des rendez-vous à venir.
    std::cout << "Voici vos rendez-vous à venir:\n\n";
    appointments.display(APPOINTMENT_PATIENT_ID, loggedInUser->profileID);

    // Attend que l'utilisateur appuie sur ENTER pour continuer l'exécution.
    std::cout << "\nAppuyez sur ENTER pour continuer...";
    std::cin.ignore();
    std::cin.get();

    break;
}

// Ici nous allons désaffecter loggedInUser
delete loggedInUser;
return EXIT_SUCCESS;
}

```

## 5) Compilation et exécution du code

### resultat pour choice=1

```

Veillez choisir parmi une des options suivantes:
0. Quitter.
1. Consulter les disponibilités d'un.e infirmier.e.
2. Prendre rendez-vous avec un.e infirmier.e
3. Voir mes rendez-vous.

[0, 1, 2 ou 3]: 1

Veillez sélectionner un.e infirmier.e par son index: 1

Il est impossible de prendre rendez-vous avec cette infirmière pour les dates suivantes:
Date          AM/PM
1. 22/11/2019    PM
2. 29/12/2019    AM

Appuyez sur ENTER pour continuer...
Voici la liste des infirmières:

```

	UserID	FirstName	LastName	Email	Telephone	StreetNumber	StreetName	City	PostalCode	HourlyRate	LicenseNumber
1.	mcclouis	Louis	Mccartney	louis.mccartn	514-511-1111	1232	3e av.	Montreal	H1T 2V5	30	93202012
2.	manononon	Manon	Lepage	manon.lepage@	514-512-1111	213	4e av.	Montreal	H3T 1P1	38	83604319

```

Veillez choisir parmi une des options suivantes:
0. Quitter.
1. Consulter les disponibilités d'un.e infirmier.e.
2. Prendre rendez-vous avec un.e infirmier.e
3. Voir mes rendez-vous.

[0, 1, 2 ou 3]:
```

### Résultat choice=2

```

3. Voir mes rendez-vous.

[0, 1, 2 ou 3]: 2

Veillez sélectionner un.e infirmier.e par son index: 1

Veillez indiquer la date du rendez-vous [jj/mm/aaaa]: 14/12/2019

Veillez indiquer le moment du rendez-vous [AM/PM]: PM

Veillez indiquer la raison du rendez-vous: malaria

Votre rendez-vous a été correctement enregistré
Appuyez sur ENTER pour continuer...
Voici la liste des infirmières:

```

	UserID	FirstName	LastName	Email	Telephone	StreetNumber	StreetName	City	PostalCode	HourlyRate	LicenseNumber
1.	mcclouis	Louis	Mccartney	louis.mccartn	514-511-1111	1232	3e av.	Montreal	H1T 2V5	30	93202012
2.	manononon	Manon	Lepage	manon.lepage@	514-512-1111	213	4e av.	Montreal	H3T 1P1	38	83604319

```

Veillez choisir parmi une des options suivantes:
0. Quitter.
1. Consulter les disponibilités d'un.e infirmier.e.
2. Prendre rendez-vous avec un.e infirmier.e
3. Voir mes rendez-vous.

[0, 1, 2 ou 3]:
```

## Résultat pour choice=3

```
[0, 1, 2 ou 3]: 3
Voici vos rendez-vous à venir:
```

	NurseUserID	PatientUserID	Date	AM/PM	Reason
1.	manononon	jo411	23/04/2019	AM	tousse
2.	mclouis	jo411	22/11/2019	PM	grippe
3.	mclouis	jo411	29/12/2019	AM	ennuie
4.	manononon	jo411	02/12/2019	PM	consultation
5.	manononon	jo411	02/12/2019	AM	migraine
6.	mclouis	jo411	14/12/2019	PM	malaria

Appuyez sur ENTER pour continuer...

Voici la liste des infirmières:

	UserID	FirstName	LastName	Email	Telephone	StreetNumber	StreetName	City	PostalCode	HourlyRate	LicenseNumber
1.	mclouis	Louis	Mccartney	louis.mccartn	514-511-1111	1232	3e av.	Montreal	H1T 2V5	30	93202012
2.	manononon	Manon	Lepage	manon.lepage@	514-512-1111	213	4e av.	Montreal	H3T 1P1	38	83604319

Veillez choisir parmi une des options suivantes:

- 0. Quitter.
- 1. Consulter les disponibilités d'un.e infirmier.e.
- 2. Prendre rendez-vous avec un.e infirmier.e
- 3. Voir mes rendez-vous.

```
[0, 1, 2 ou 3]:
```

## Résultat pour choice=0

```
Appuyez sur ENTER pour continuer...
Voici la liste des infirmières:
```

	UserID	FirstName	LastName	Email	Telephone	StreetNumber	StreetName	City	PostalCode	HourlyRate	LicenseNumber
1.	mclouis	Louis	Mccartney	louis.mccartn	514-511-1111	1232	3e av.	Montreal	H1T 2V5	30	93202012
2.	manononon	Manon	Lepage	manon.lepage@	514-512-1111	213	4e av.	Montreal	H3T 1P1	38	83604319

Veillez choisir parmi une des options suivantes:

- 0. Quitter.
- 1. Consulter les disponibilités d'un.e infirmier.e.
- 2. Prendre rendez-vous avec un.e infirmier.e
- 3. Voir mes rendez-vous.

```
[0, 1, 2 ou 3]: 0
[gigl@fedora-gigl TP5_methode]$
```



## 6) Commit après la restructuration de main

```
[gigl@fedora-gigl TP5]$ git add .
[gigl@fedora-gigl TP5]$ git commit -m "la version apres la restructuration de main"
[master 7833229] la version apres la restructuration de main
 5 files changed, 69 insertions(+), 27 deletions(-)
[gigl@fedora-gigl TP5]$ git push
Username for 'https://github.com': fomou
Password for 'https://fomou@github.com':
Énumération des objets: 19, fait.
Décompte des objets: 100% (19/19), fait.
Compression des objets: 100% (10/10), fait.
Écriture des objets: 100% (10/10), 1.62 KiB | 1.62 MiB/s, fait.
Total 10 (delta 7), réutilisés 0 (delta 0)
To https://github.com/fomou/polymtl.git
   fc956a7..7833229  master -> master
[gigl@fedora-gigl TP5]$
```

## E4) Tests

un capture d'écran qui prouve que tous les 4 tests passent encore

```
fc956a7..7833229  master -> master
[gigl@fedora-gigl TP5]$ make test
g++ -Wall -std=c++11 -pedantic -g -o tests/Test.o -c tests/Test.cpp
g++ -c -o tests/main.o tests/main.cpp
g++ -Wall -std=c++11 -pedantic -g -o src/Nurses.o -c src/Nurses.cpp
g++ -Wall -std=c++11 -pedantic -g -o src/Utils.o -c src/Utils.cpp
g++ -Wall -std=c++11 -pedantic -g -o src/Appointments.o -c src/Appointments.cpp
g++ -Wall -std=c++11 -pedantic -g -o src/CSVHandler.o -c src/CSVHandler.cpp
g++ -o test.out tests/Test.o tests/main.o src/Nurses.o src/Utils.o src/Appointments.o src/CSVHandler.o -I./ -lcppunit
[gigl@fedora-gigl TP5]$ ./test.out
....

OK (4 tests)

[gigl@fedora-gigl TP5]$
```