Design Pattern

Template Method & Duplicate Code

Elisa Gougerot & Taj Singh & Victor Deyanovitch

Mode opératoire

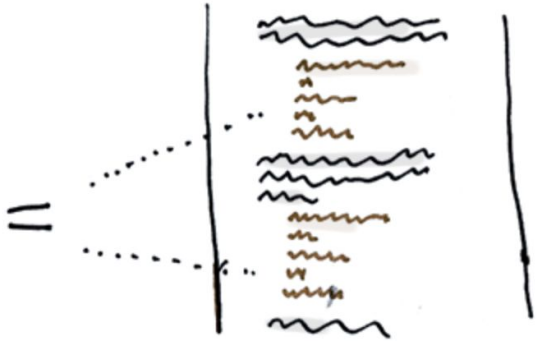
Nous avons commencé la création de notre exposé le week-end précédant le rendu via un Google Slide afin d'y déposer en vrac nos idées.

Nous avons ensuite construit le plan en s'appuyant sur celui proposé par notre enseignant sur Discord et fait la synthèse de nos recherches.

Enfin nous avons finalisé notre la présentation, afin de la rendre la plus visuelle possible.

Le plus difficile était de préparer un exposé qui est présenté à distance (synchronisation de l'oral entre les membres du groupe)

Duplicate Code



Répétition d’une ligne ou d’un bloc de code dans un même fichier/projet

Comment arrive t’on à ce code smell ?

- Copier-coller et s’il y a des contraintes de délai

- Quand plusieurs développeurs travaillent sur différentes tâches d’un projet en même temps

- Lorsque des parties de code semble différentes mais qu’elles exécutent la même tâche

Une approche pour y remédier

Befor Java C# PHP Python TypeScript

```java
void printOwing() {
  printBanner();

  // Print details.
  System.out.println("name: " + name);
  System.out.println("amount: " + getOutstanding
}
```

After

```java
void printOwing() {
  printBanner();
  printDetails(getOutstanding());
}

void printDetails(double outstanding) {
  System.out.println("name: " + name);
  System.out.println("amount: " + outstanding);
}
```

Extract Method

```java
double disabilityAmount() {
  if (seniority < 2) {
    return 0;
  }
  if (monthsDisabled > 12) {
    return 0;
  }
  if (isPartTime) {
    return 0;
  }
  // Compute the disability amount.
  // ...
}
```

```java
double disabilityAmount() {
  if (isNotEligibleForDisability()) {
    return 0;
  }
  // Compute the disability amount.
  // ...
}
```

Consolidate conditional expression

```java
if (isSpecialDeal()) {
  total = price * 0.95;
  send();
}
else {
  total = price * 0.98;
  send();
}
```

```java
if (isSpecialDeal()) {
  total = price * 0.95;
}
else {
  total = price * 0.98;
}
send();
```

Consolidate duplicate conditional code

Befor Java C# PHP Python TypeScript

```java
class Manager extends Employee {
  public Manager(String name, String id, int gra
    this.name = name;
    this.id = id;
    this.grade = grade;
  }
  // ...
}
```

After

```java
class Manager extends Employee {
  public Manager(String name, String id, int gra
    super(name, id);
    this.grade = grade;
  }
  // ...
}
```

Pull Up
(Field,Constructor, Method)

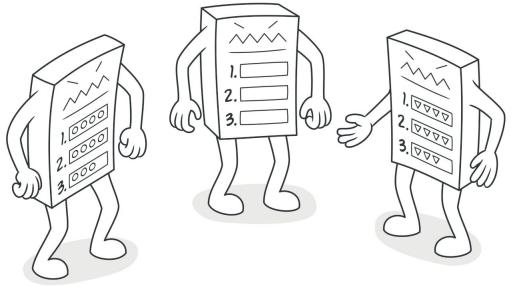
Template Method (patron de méthode)

- Modèle de conception

- Définit le squelette d’un algorithme dans la classe mère

- Permet aux classes filles de remplacer des étapes de l'algorithme général sans changer sa structure

Son intention



Faire varier le comportement de chaque opération de façon indépendante dans les sous-classes

Deux classes peuvent être amenés à utiliser une méthode similaire mais avec quelques petits changements

Les conséquences de l’utilisation d’un Template Method

Évite la duplication du code :

Mutualiser des comportements dans une classe de base tout en offrant la possibilité de redéfinir certaines parties de l’algorithme pour les spécifiés

Maintien du code plus complexe :

Plus un patron de méthode comporte d’étape, plus il est difficile à maintenir

Exemple de code

```
public class Cafe {
    public void PreparerCafe() {
        FaireBouillirEau();
        FairePasserLeCafeDansLeFiltre();
        AjouterLeSucre();
        Remuer();
    }

    private void FaireBouillirEau() {
        Console.WriteLine("Faire bouillir de l'eau");
    }

    private void FairePasserLeCafeDansLeFiltre(){
        Console.WriteLine("Faire passer le café dans le filtre");
    }

    private void AjouterLeSucre(){
        Console.WriteLine("Ajouter le sucre");
    }

    private void Remuer(){
        Console.WriteLine("Remuer");
    }
}
```

```
public class The {
    public void PreparerThe(){
        FaireBouillirEau();
        FaireInfuser();
        AjouterLeSucre();
        AjouterDuCitron();
        Remuer();
    }

    private void FaireBouillirEau(){
        Console.WriteLine("Faire bouillir de l'eau");
    }

    private void FaireInfuser(){
        Console.WriteLine("Faire infuser le thé");
    }

    private void AjouterLeSucre(){
        Console.WriteLine("Ajouter le sucre");
    }

    private void AjouterDuCitron(){
        Console.WriteLine("Ajouter du citron");
    }

    private void Remuer(){
        Console.WriteLine("Remuer");
    }
}
```

Exemple de code

```
public abstract class Boisson {
    public void PreparerBoisson(){
        FaireBouillirEau();
        Preparer();
        AjouterLeSucre();
        AjouterLesSupplements();
        Remuer();
    }

    private void FaireBouillirEau(){
        Console.WriteLine("Faire bouillir de l'eau");
    }

    protected abstract void Preparer();

    private void AjouterLeSucre(){
        Console.WriteLine("Ajouter le sucre");
    }

    protected abstract void AjouterLesSupplements();

    private void Remuer(){
        Console.WriteLine("Remuer");
    }
}
```

😊 Solution

```
public class Cafe : Boisson {

    protected override void Preparer() {
        FairePasserLeCafeDansLeFiltre();
    }

    protected override void AjouterLesSupplements() {
        // aucun supplément
    }

    private void FairePasserLeCafeDansLeFiltre() {
        Console.WriteLine("Faire passer le café dans le filtre");
    }
}

public class The : Boisson {
    protected override void Preparer() {
        FaireInfuser();
    }

    protected override void AjouterLesSupplements() {
        AjouterDuCitron();
    }

    private void FaireInfuser() {
        Console.WriteLine("Faire infuser le thé");
    }

    private void AjouterDuCitron() {
        Console.WriteLine("Ajouter du citron");
    }
}
```

Sources

- https://sourcemaking.com/design_patterns/template_method
- https://sourcemaking.com/refactoring/smells/duplicate-code
- http://www.neotech-solutions.fr/blogs/nicolas-hilaire/dumd-1224-le-design-pattern-template-method/
- https://refactoring.guru/design-patterns/template-method