

# Technologie Web

## Conception de sites Web

**Alexandre Pauchet**

INSA Rouen - Département ASI

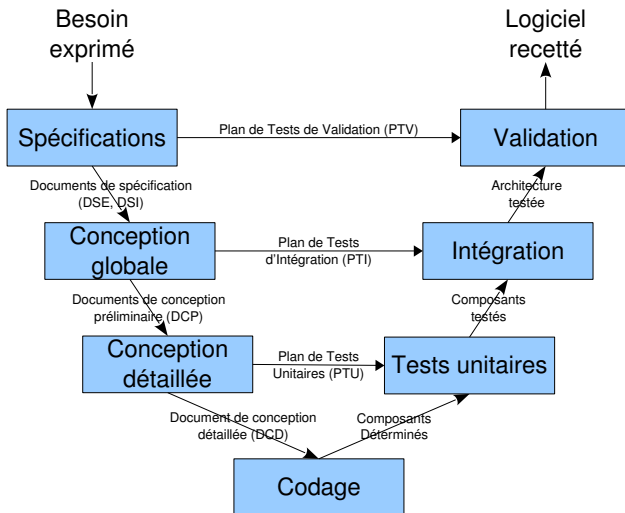
BO.B.RC.18, [pauchet@insa-rouen.fr](mailto:pauchet@insa-rouen.fr)

# Plan

- 1 Démarche
- 2 Cas d'utilisation
- 3 Navigation
- 4 Interaction
- 5 Modèle objet
- 6 Conclusion

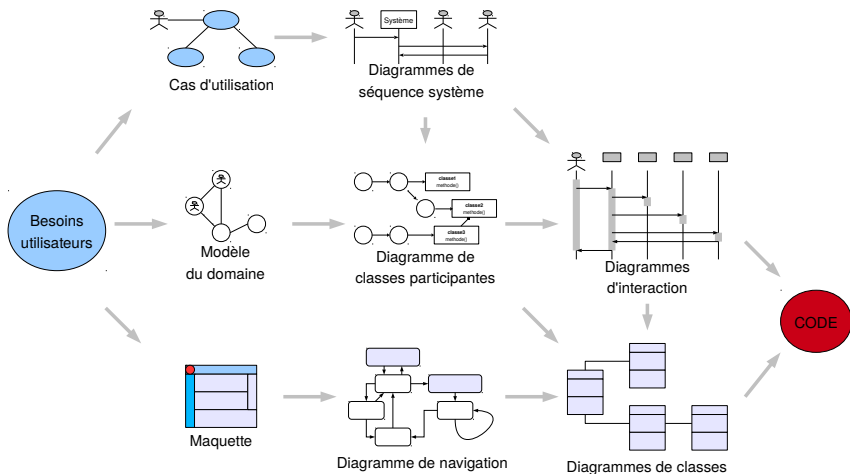
# Démarche (1/7)

## Rappel cycle en V du développement logiciel



# Démarche (2/7)

## Approche



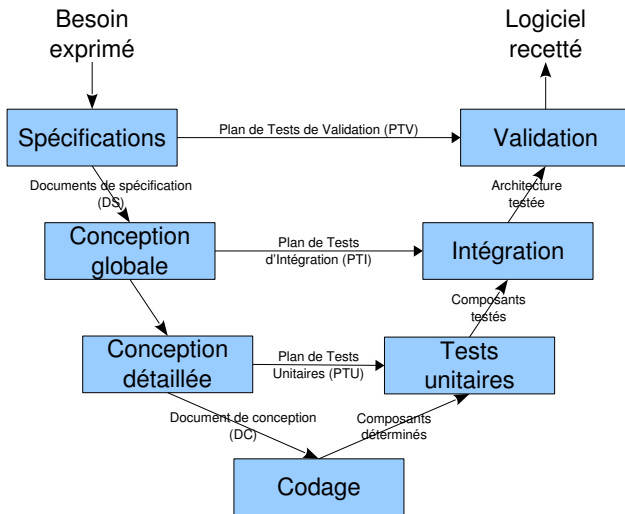
# Démarche (3/7)

## Documentation

- *Spécifications*
  - Externes : formalisation du besoin, contractuel
  - Internes : points non nécessaires au client (domaine de compétence)
  - Artefacts UML : maquette, diagramme de navigation, cas d'utilisation
- *Plan de tests de validation* et *Cahier de recette* réalisés // à la spécification
- *Conception Préliminaire* : modèle du domaine, diagramme de classes participantes, diagrammes de séquences et d'interaction, diagramme de packages
- *Plan de tests d'intégration* et *Journal des tests d'intégration* réalisés // à la conception préliminaire
- *Conception Détaillée* : diagramme de classes, pseudo-code
- *Plan de tests unitaires* et *Journal des tests unitaires* réalisés // à la conception détaillée

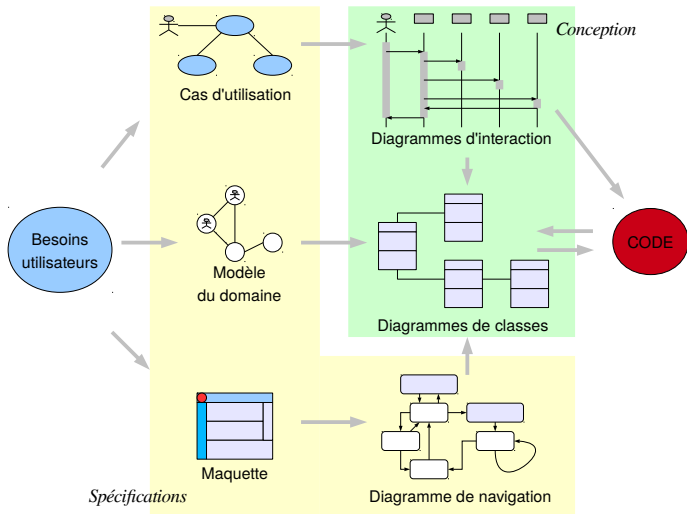
# Démarche PIC (4/7)

## Documentation simplifiée



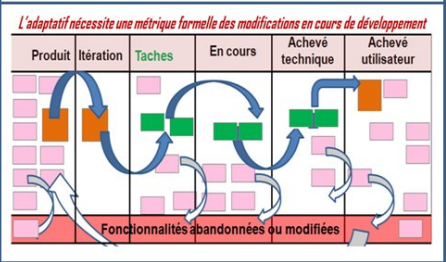
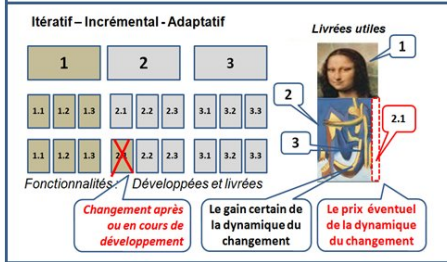
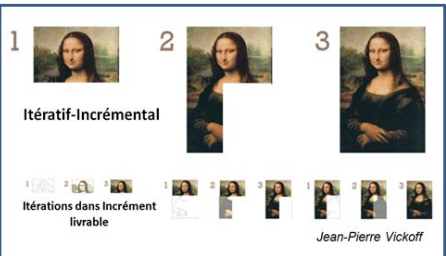
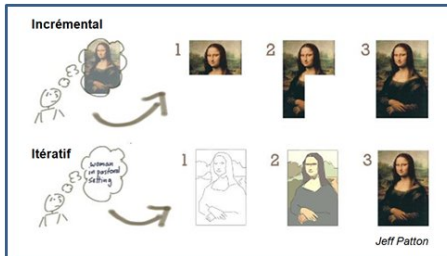
# Démarche PIC (5/7)

## Documentation simplifiée



# Démarche (6/7)

## Méthodes Agile



Tiré de : <https://fr.wikipedia.org/wiki/Fichier:IteratiIncrémentalAdaptatif.jpg>



# Démarche (7/7)

## Méthodes Agiles et PIC

- Vocabulaire différent :
  - "Cas d'utilisation"="User story",
  - "Test de validation"="Critère d'acceptation",
  - ...
  
- Mini-cycles en V
  - Chaque sprint suit un mini-cycle en V
  - Les documents sont mis à jour itérativement
  
- "Full Agile"
  - Documentation adaptée : product backlog, sprint backlog, ...
  - Les artefacts UML utilisés sont les mêmes

# Spécification des exigences (cas d'utilisation) (1/5)

## Cas d'étude

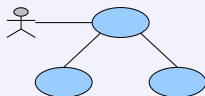
### Librairie en ligne

- Ex : `www.amazon.fr`, `www.fnac.com`, `www.eyrolles.com`, *etc.*
- Exigences fonctionnelles : recherche, présentation d'ouvrages, sélection, commande
- Exigences non fonctionnelles : ergonomie, simplicité d'utilisation, performance, *etc.*
- Contraintes de conception :
  - Mise à jour des données de référence
  - Mise à jour depuis les formulaires du site
  - Panier
  - Paiement sécurisé

# Spécification des exigences (cas d'utilisation) (2/5)

## Démarche

### Structuration en cas d'utilisation

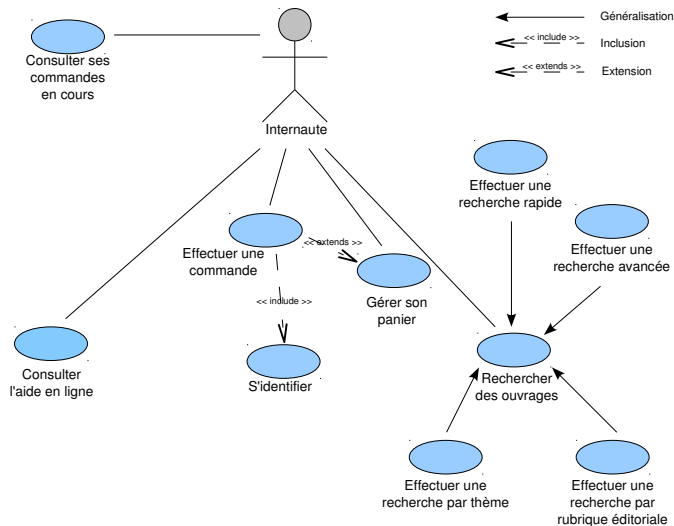


Cas d'utilisation

- Identifier les acteurs
- Identifier les cas d'utilisation
- Structurer les cas d'utilisation en package
- Étudier les relations entre cas d'utilisation
- Hiérarchiser les cas d'utilisation

# Spécification des exigences (cas d'utilisation) (3/5)

Exemple : cas d'utilisation de l'internaute



# Spécification des exigences (cas d'utilisation) (4/5)

## Scenarios

### Exemple : "Effectuer une commande"

**Acteur principal** : l'internaute

**Objectifs** : À tout moment, le client doit pouvoir accéder au formulaire du bon de commande, dans lequel il peut saisir ses coordonnées et les informations nécessaires au paiement et à la livraison.

**Préconditions** : Le panier n'est pas vide et l'Internaute a pu accéder au formulaire de commande (cf. *Gérer son panier*).

**Postconditions** : Une commande a été enregistrée et transmise au service Commandes.

**Scénario nominal** :

1. L'Internaute saisit l'ensemble des informations nécessaires au paiement et à la livraison :
  - son adresse email,
  - son adresse de facturation,
  - son adresse de livraison si elle est différente de son adresse de facturation,
  - son numéro de carte de crédit.
2. Le Système affiche un récapitulatif de la commande.
3. L'internaute valide sa commande.
4. Le Système envoie la commande validée au service Clients.
5. Le Système confirme la prise de commande à l'Internaute.

**Extensions** :

- 1a. L'Internaute est déjà client.
  1. L'internaute s'identifie avec son e-mail et son mot de passe.
  2. Le Système affiche les données concernant l'Internaute.
    - 2a. Le Système ne reconnaît pas le client  
Le Système prévient l'Internaute du problème (retour en 1a.1).
- 1-3a. L'Internaute annule sa commande.
  1. Le Système revient à l'affichage du panier (cas d'utilisation terminé).

**Exigences supplémentaires** : Utilisation de SSL.

# Spécification des exigences (cas d'utilisation) (5/5)

## Planification

### Remarque

Scenarios → diagrammes de séquence système

### Packages

- UC de l'internaute
- UC des employés
- Acteurs
- UC secondaires

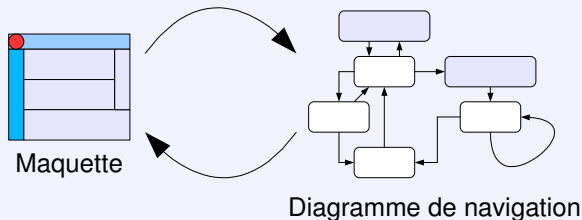
### Classement des cas d'utilisation

Cas d'utilisation	Priorité	Risque	Itération
Rechercher des ouvrages	Haute	Moyen	2
Gérer son panier	Haute	Bas	3
Effectuer une commande	Moyenne	Haut	4
Consulter ses commandes en cours	Basse	Moyen	6
Consulter l'aide en ligne	Basse	Bas	7
Maintenir le catalogue	Haute	Bas	1
Maintenir les informations éditoriales	Moyenne	Bas	5
Maintenir le site	Moyenne	Bas	5

# Navigation (1/2)

## Approche

### Approche

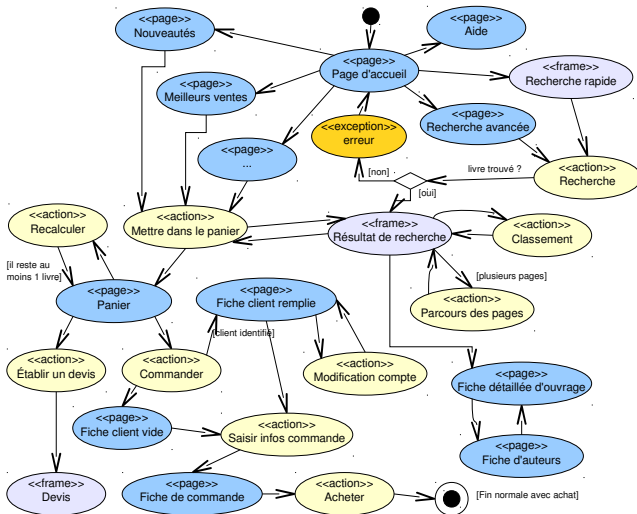


### Maquette

- Dessins/canevas pouvant intégrer des fonctionnalités de navigation
- Vue concrète mais non définitive de l'IHM
- Support de discussion avec le client

# Navigation (2/2)

## Diagramme de navigation





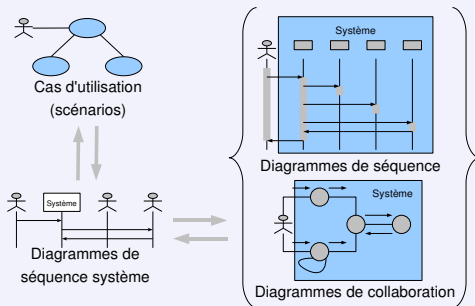
# Interaction (1/3)

## Approche

### Diagrammes d'interaction

- Diagrammes de séquence
- Diagrammes de collaboration

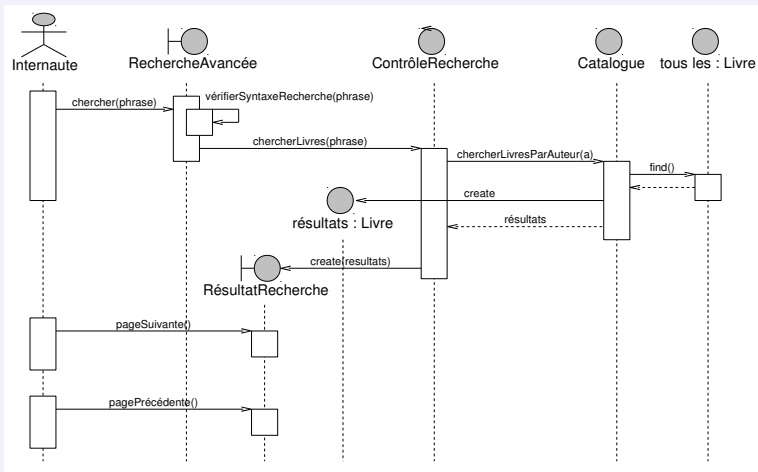
## Approche



# Interaction (2/3)

## Diagramme de séquence

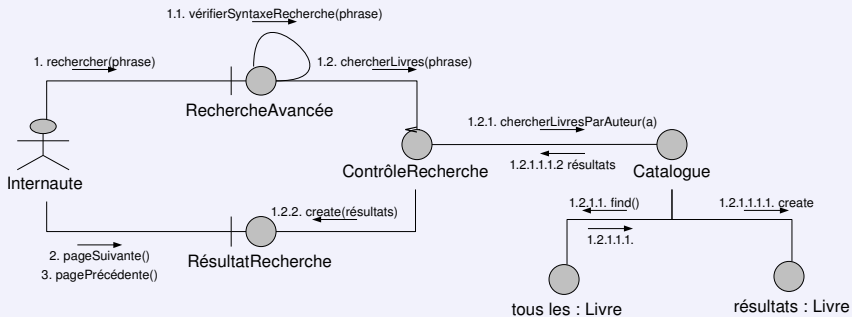
### Exemple : Recherche avancée



# Interaction (3/3)

## Diagramme de collaboration

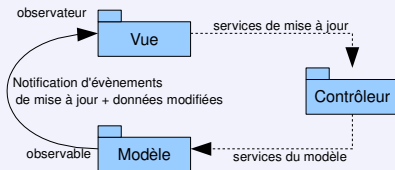
### Exemple : Recherche avancée



# Modèle objet (1/2)

Rappel : MVC

## MVC : Modèle-Vue-Contrôleur



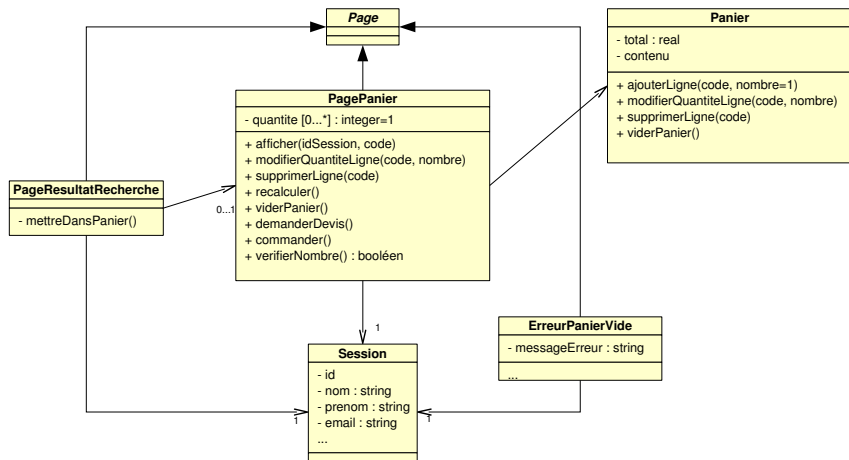
## Choix du paradigme

- Au moment de l'analyse du problème (Cas d'utilisation, modèle du domaine, diagramme de navigation)
- 3 approches :
  - 1 Fonctionnelle/Procédurale
  - 2 Objets
  - 3 Composants

# Modèle objet (2/2)

## Diagramme de classes

### Gestion du panier



# Conclusion (1/3)

## Limites du modèle UML

### Qualités

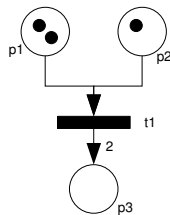
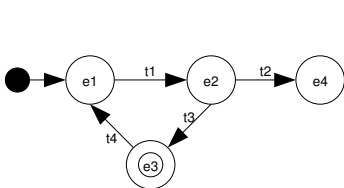
- "Simple"
- Expressif
- Méthodologique

### Limites

- Ne représente pas les états du système
- Rapidement limité (gros systèmes)
- Pas d'interconnexion de systèmes
- Pas dynamique (Agents informatiques, composants, *etc.*)
- Pas de vérification

## Conclusion (2/3)

### Alternatives



### Alternatives

- Approche SMA
- Représentation des interactions par automates/réseaux de Petri
- Vérification logique

**Contrainte : évènements discrets**

## Conclusion (3/3)

### Références

- “*UML : Modéliser un site e-commerce*”, Pascal Roques, Eyrolles.
- <http://laurent-audibert.developpez.com/Cours-UML/html/index.html>
- <http://www.coredump.fr.to/category/modelisation/uml/>
- <http://uml.free.fr/>