



Une entreprise ordinaire à l'agilité extraordinaire

FORMATION JAVA SPRING ANGULAR

UML

mohamed.el-babili@fms-ea.com

33+ 628 111 476

Version : 2.0

DMAJ : 14/02/23

Sommaire

2

- Uml c'est quoi ?
- Uml, ça sert à quoi
- Concrètement
- Etude de cas
- Les cas d'utilisations / use cases (1)(2)
- Spécifications fonctionnelles
- Diagramme de classe : class diagram
- Diagramme de séquence : sequence diagram
- Architecture de l'appli : Diagramme de package
- Diagramme d'activité
- Les autres diagrammes
- Tools
- Ressources

Uml, c'est quoi ?



- Le langage UML (Unified Modeling Language, ou langage de modélisation unifié) a été pensé pour être un **langage de modélisation visuelle commun**
- Il est **destiné à l'architecture, la conception** et la mise en œuvre de **systèmes logiciels complexes** par leur structure aussi bien que leur comportement
- Il **ressemble aux plans (d'architectes par ex)** utilisés dans d'autres domaines et se compose de différents types de diagrammes.
- Les **diagrammes UML décrivent dans un système d'information :**
 - limite
 - structure
 - comportement
 - objets
- **Uml n'est pas un langage de programmation** mais il existe des outils pour générer du code à partir de diagramme de classe par ex.
- **Uml a une relation directe avec l'analyse et la Poo** qui domine le monde de la programmation.
- On **doit la naissance d'Uml aux Three Amigos : Grady, Booch et Rumbaugh** qui ont mutualisé leurs travaux ayant abouti à la version **0.9 en 1996**, puis de nombreuses entreprises y ont vu un intérêt majeur (Microsoft, Oracle et IBM...) qui favorisa le développement d'un **langage de modélisation complet publié en 1999**. Suivi d'une mise à jour comportant des informations sur **l'UML 2.0 en 2005**.

A ce jour, **la version 2.5 est la plus récente (2017)**

Uml, ça sert à quoi ?

- Selon son site web, l' [Object Management Group](#) (OMG) est un consortium de normes international, qui sont appliquées par les entreprises, les utilisateurs... Les normes de modélisation de l'OMG, dont l'UML et le Model Driven Architecture (MDA), permettent la conception, l'exécution et la maintenance de logiciels et d'autres processus d'une façon visuellement efficace.
- L'OMG définit les objectifs d'Uml comme suit :
 - **Fournir aux concepteurs** de systèmes, ingénieurs logiciels et développeurs de logiciels **des outils pour l'analyse, la conception** et la mise en œuvre de systèmes logiciels, ainsi que pour la modélisation de processus métier et d'autres processus similaires.
 - **Faire progresser l'industrie en permettant l'interopérabilité des outils de modélisation visuelle orientés objet.** Toutefois, pour permettre un échange significatif d'informations de modèles entre outils, il est nécessaire de trouver un accord sur la sémantique et la notation.



Concrètement

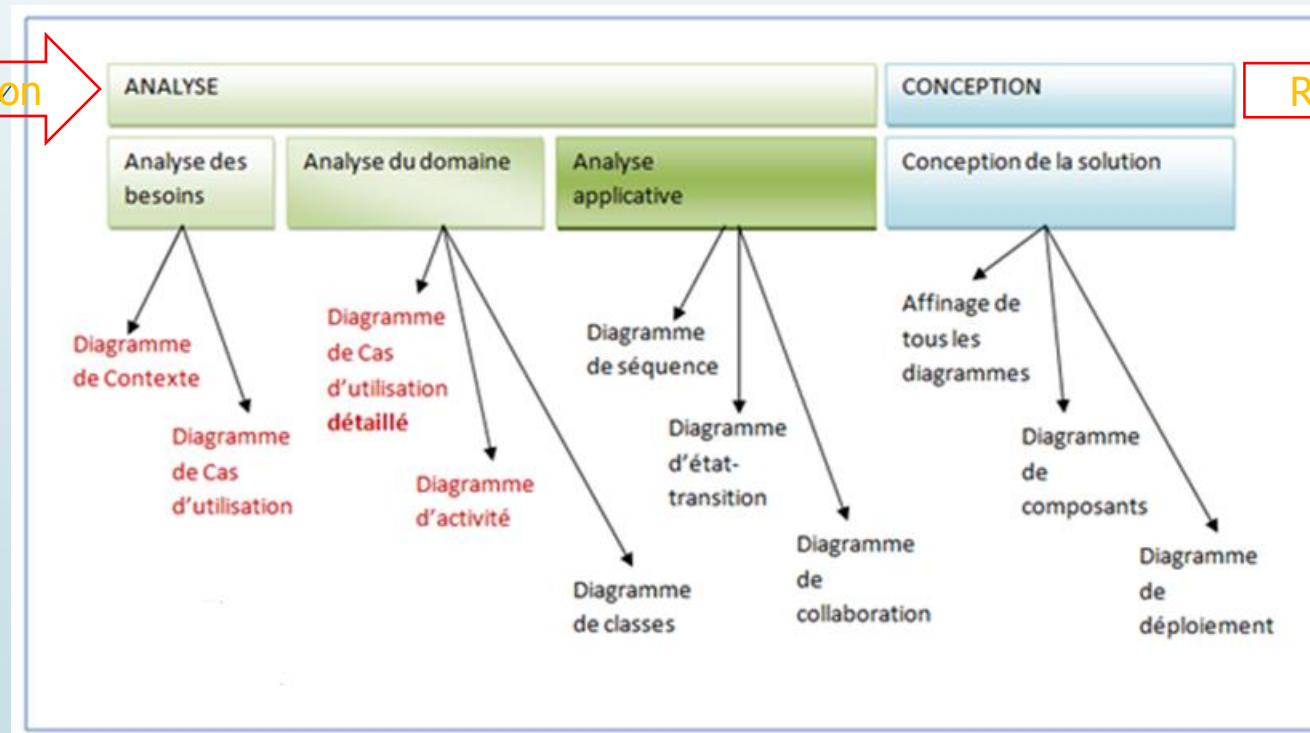
5

- Les clients émettent des besoins, on peut y répondre sans Uml, le résultat dépendra de l'expérience de chacun.



- On peut, pour chaque phase du cycle de vie d'un logiciel, utiliser un ou plusieurs diagrammes

Modélisation



Réalisation

Etude de cas

- Le meilleur moyen de comprendre l'intérêt d'Uml reste de l'utiliser dans un exemple concret.
 - Dans l'exemple qui suit, nous choisissons, à juste titre ou pas, d'utiliser que 3 diagrammes qui nous paraissent incontournables : Use case, Class, Sequence.
 - En effet, faute de temps, rares sont les équipes qui mettent en œuvre tous les diagrammes, **vous en pensez quoi ?**
- *Expressions des besoins : Application E-commerce ou boutique en ligne de vente de matériel informatique par ex, offrant la possibilité d'inter agir avec l'utilisateur lui permettant de voir une liste de produits, disponibles ou pas, classés ou pas par catégories, avec plus ou moins d'informations détaillées, dont il pourra constituer son caddy à tout instant puis passer commande s'il le souhaite. L'appli doit être ergonomique et facilite l'expérience utilisateur. L'appli doit enfin être accessible depuis un navigateur ou un smartphone.*

Les cas d'utilisations : use cases (1)

7

Un cas d'utilisation est une liste d'étapes qui définissent les interactions entre un acteur (un humain qui interagit avec le système ou un système externe) et le système lui-même.

Les diagrammes de cas d'utilisation décrivent les spécifications d'un cas d'utilisation et modélisent les unités fonctionnelles d'un système.

Ces diagrammes aident les équipes de développeurs à comprendre les besoins de leur système, notamment le rôle des interactions humaines et les différences entre plusieurs cas d'utilisation.

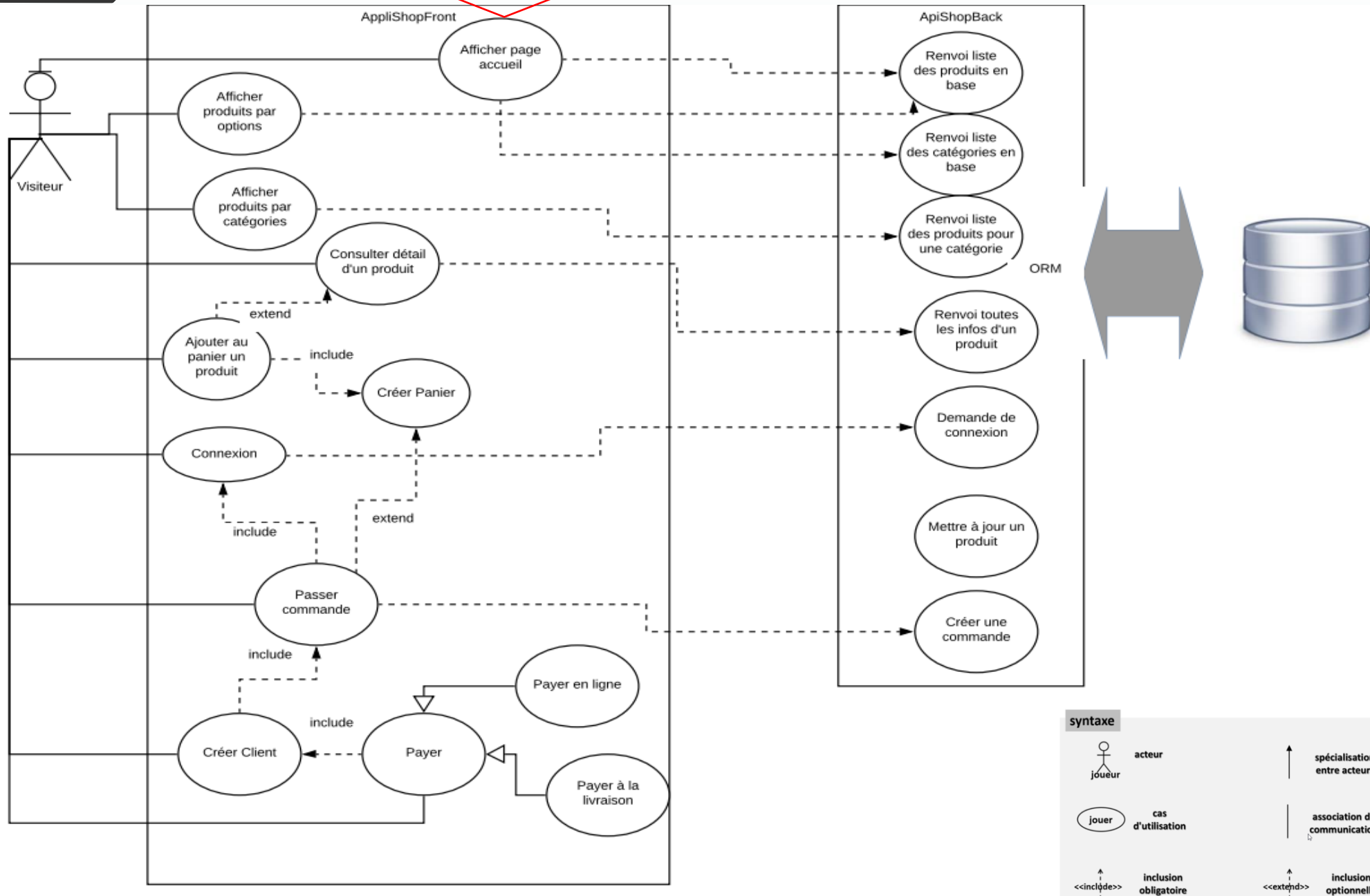
Un diagramme d'utilisation peut illustrer tous les cas d'utilisation du système ou seulement un groupe de cas d'utilisation ayant des fonctionnalités similaires.

Finalités des cas d'utilisations : traduire les besoins exprimés par l'utilisateur en spécifications fonctionnelles

Les cas d'utilisations : use cases (2)

8

1^{er} cas
d'utilisation



Spécifications fonctionnelles

9

• POUR LES VISITEURS DU SITE, l'appli permet :

- Afficher par défaut la liste des **produits** dispo au niveau du back (page d'accueil ou en cliquant sur home). Un produit est caractérisé par un flag dispo ou pas (boolean), afficher les photos de chaque produit (prévoir une photo de substitution par défaut)
- Afficher par défaut la liste des **catégories** contenue côté back (api ou pas)
- Dans la barre de navigation avec nom du site... au clic sur home, nous aurons donc la liste des produits dispos + contenu du **Caddy** en cours + login/out (renvoi vers un formulaire d'authentification)
- Afficher la liste des produits correspondant à une catégorie par le simple clic sur celle-ci (elle apparaîtra en bleu par exemple pour informer l'utilisateur)
- Afficher le pointer vers la catégorie ou produit survolée avec la souris (pour clic + Les catégories non sélectionnés doivent apparaître en grisées)
- Afficher sous chaque produit des icônes signifiant s'ils sont dispo ou pas
- Lorsqu'on clic sur une photo d'un produit par exemple, afficher les informations détaillées sur le produit correspondant : nom, description, prix, dispo...
- Permettre d'ajouter des produits au caddy, de supprimer ou visualiser le contenu du caddy.
- Créer un **compte client** pour **passer commande** notamment

• POUR LES UTILISATEURS CONNECTES, l'appli permet en plus :

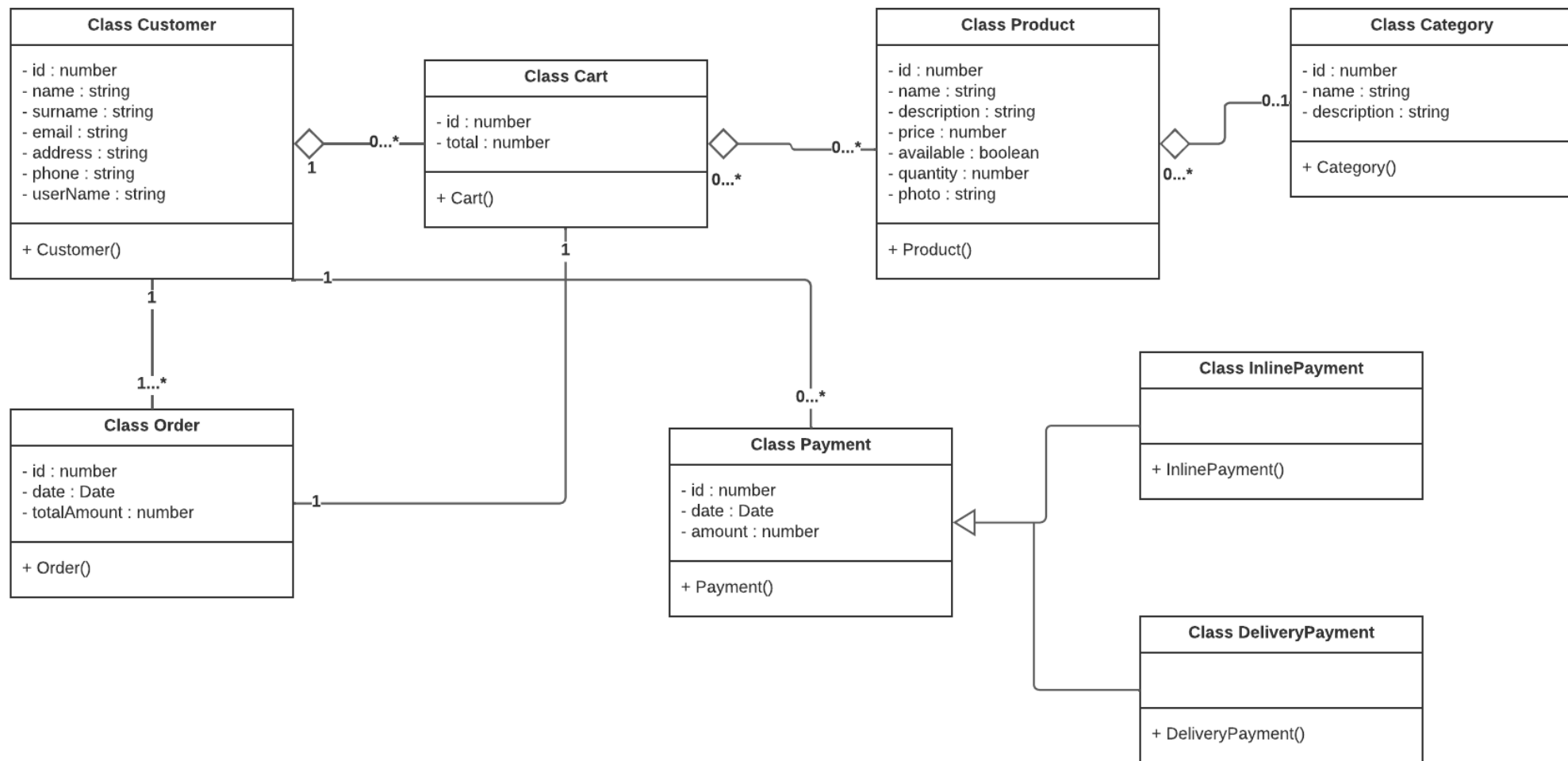
- Passer commande et **payer**

• POUR LES ADMINISTRATEURS DU SITE, l'appli permet en plus :

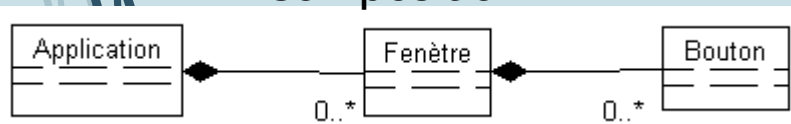
- Ajouter, supprimer un produit ou modifier les infos ...

Diagramme de classe : class diagram

10



Composition



Agrégation

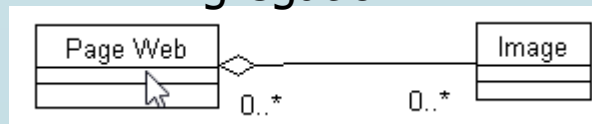
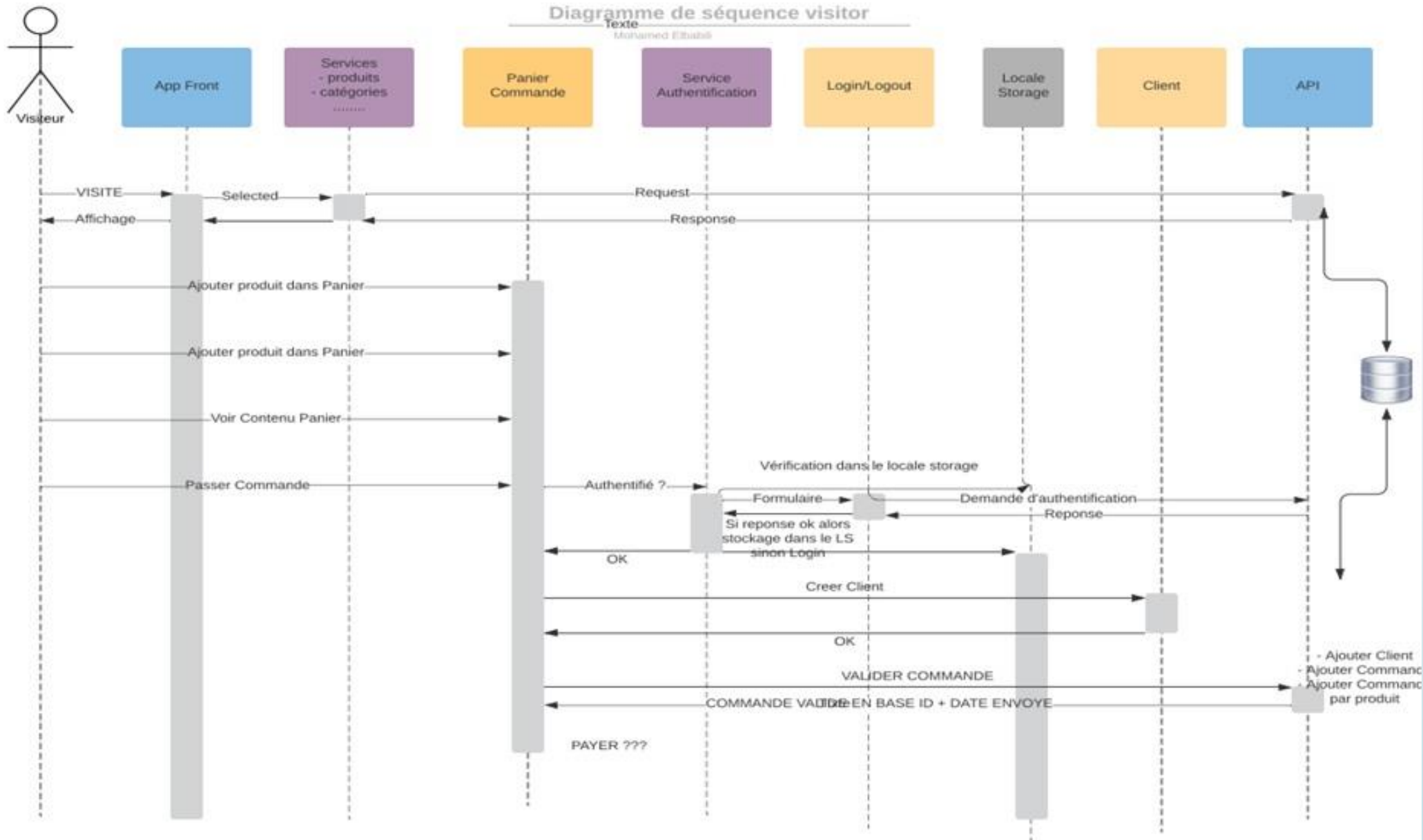


Diagramme de séquence : sequence diagram

Montre comment les objets interagissent entre eux et dans quel ordre dans un scénario précis



Architecture de l'appli : Diagramme de package

- Pour représenter l'architecture de l'application, on peut utiliser les diagrammes de packages, ici incomplets mais cela nous donne une idée globale entre les différentes couches de l'application

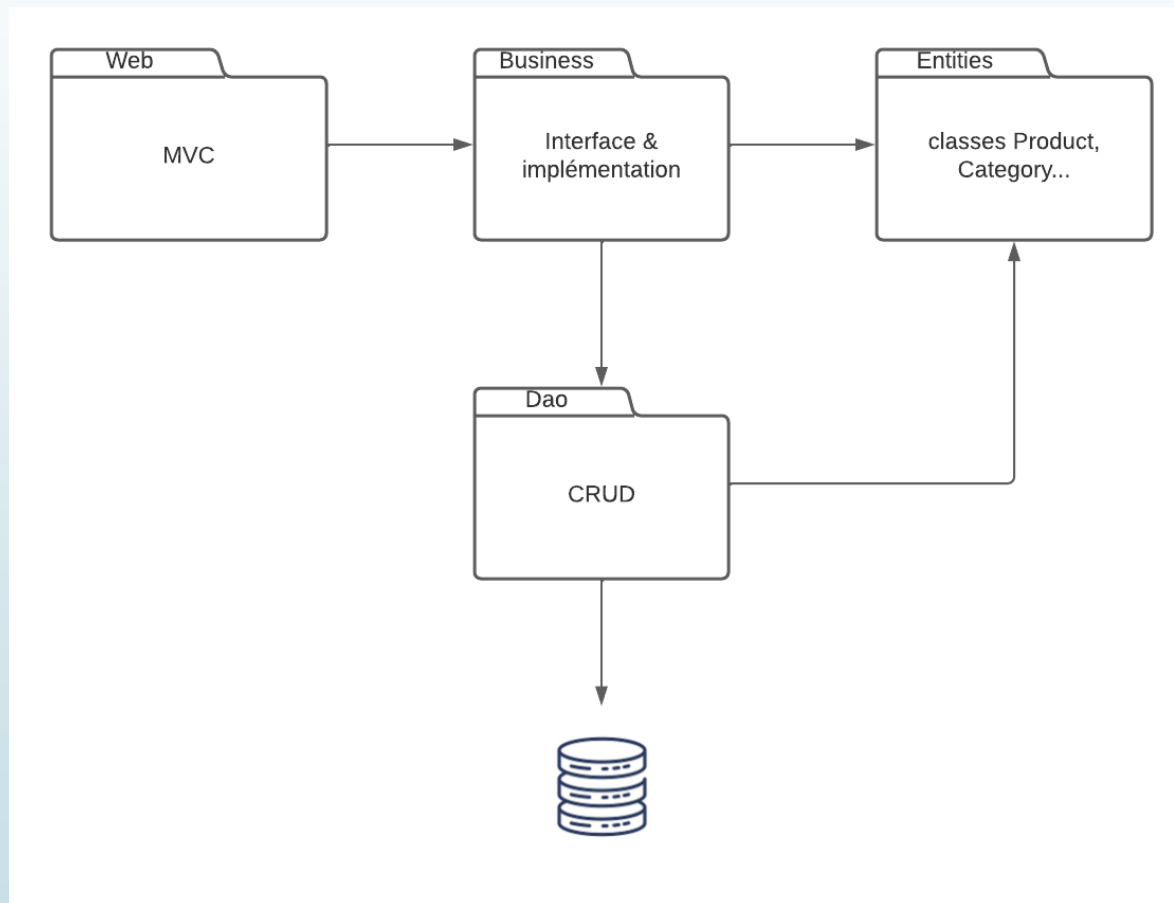
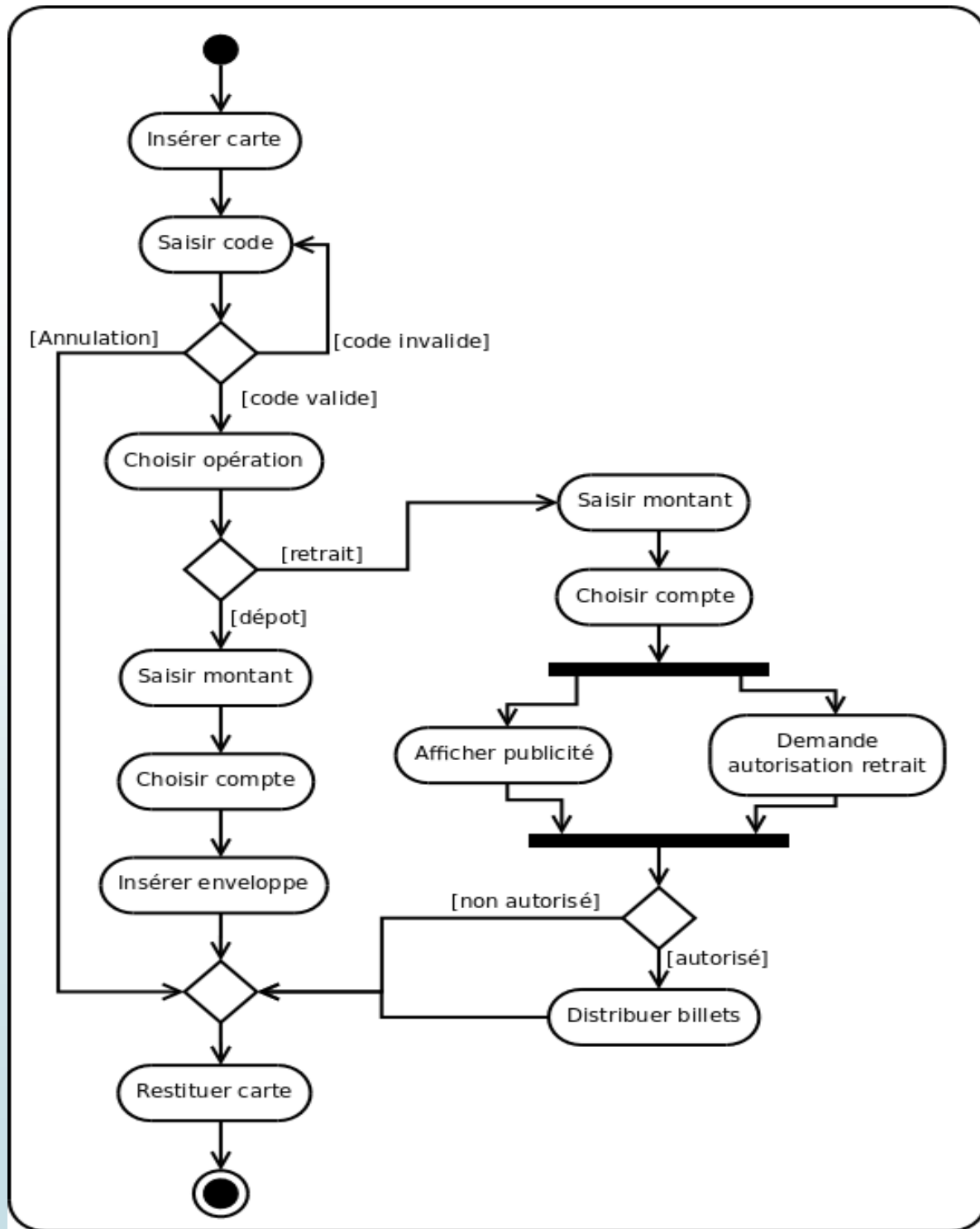
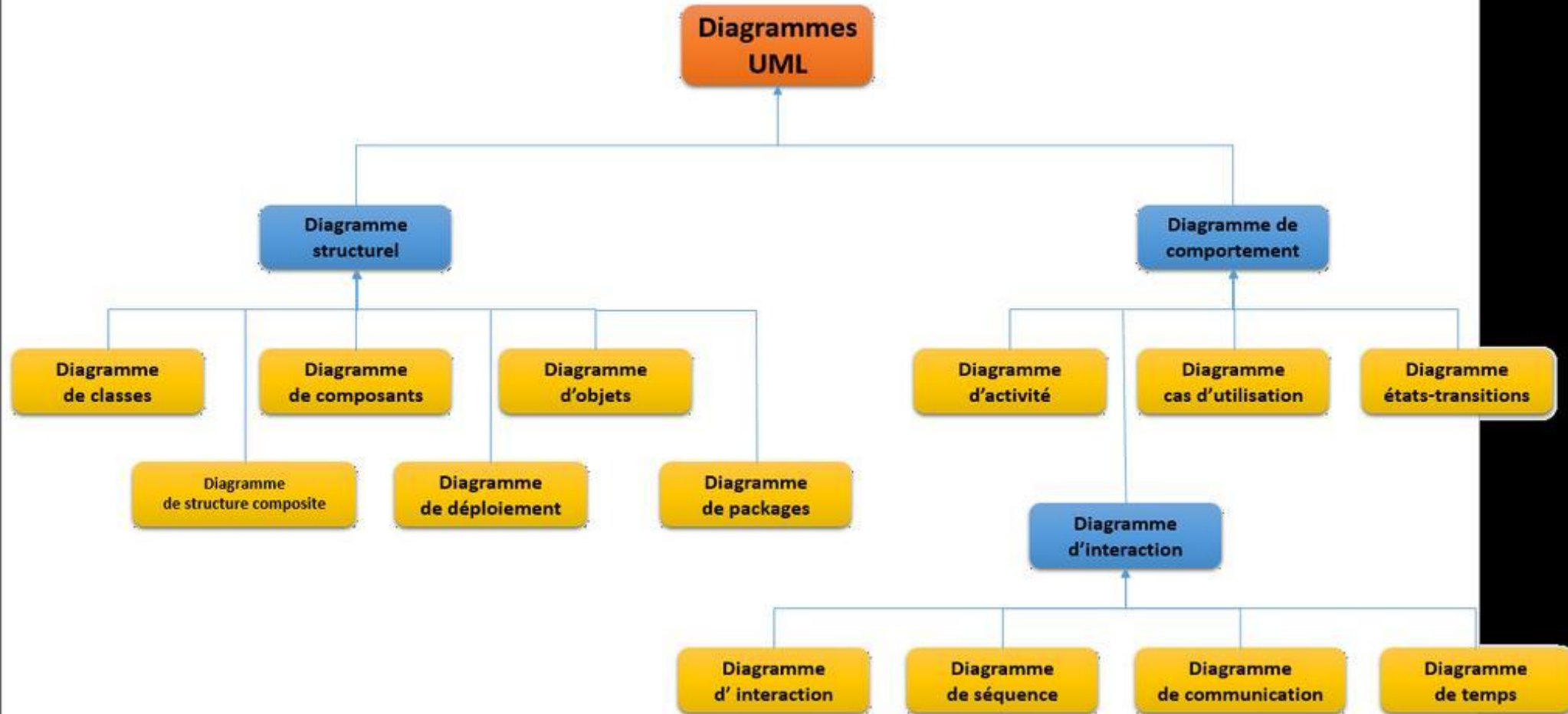


Diagramme d'activité



Liste des diagrammes UML 2.5

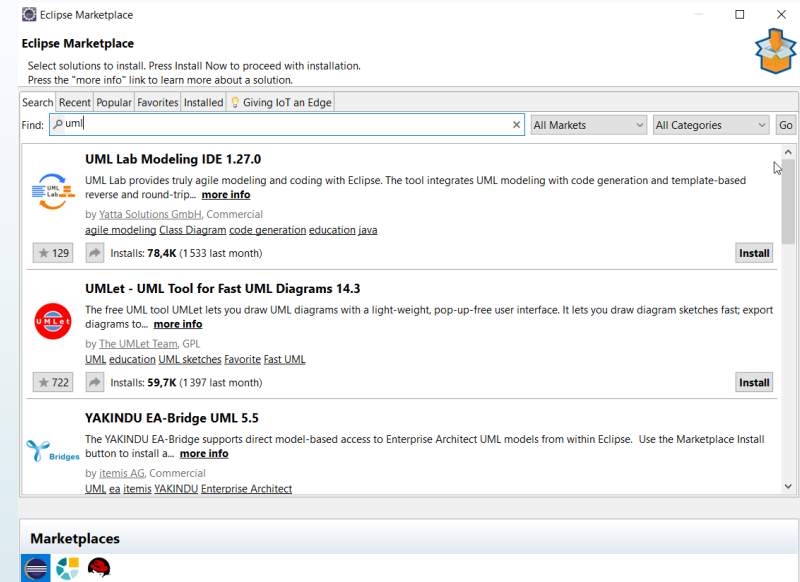


Tools

15

- Sur le marketplace d'Eclipse, vous avez l'embarras du choix :

➔ [25 BEST UML Tools | FREE UML Diagram Software in 2022](#)



Ressources

- <https://www.lucidchart.com/pages/fr/langage-uml>
- Livre « uml par la pratique » Pascal Roques
- <https://creatly.com/blog/fr/uncategorized-fr/tutoriel-sur-les-diagrammes-de-classe/>
- <https://youtu.be/fpxjv5rwXc4>