

Intitulé de l'épreuve : Conception Logicielle

Nombre de copies : 1/3

Numérotez chaque page (dans le cadre en bas de la page) et placez les feuilles dans le bon sens.

Partie 1 : Etude de cas

1)

Tout d'abord, à la vue de l'expression des besoins du client (Traduct) il me semble essentiel de délimiter le périmètre du projet.

Dans les exigences du système, il n'y a pas de référence à la définition ou à la signature du contrat, au démarchage de nouveaux clients, au suivi des clients actuels par le pôle commercial

De même pour les activités de compatibilité par le pôle administratif. (par ex. facturation)

Donc, les interactions avec le système se limiteront ainsi :

* le commercial doit pouvoir

- créer le dossier : entrer toutes les données du contrat nécessaire au système dans un formulaire (en retour le système renvoie une date de rendu)

* le client doit pouvoir

- déposer le document initial (à traduire)
- récupérer le document traduit
- récupérer le document relu (si l'option a été demandée)

* le traducteur doit pouvoir

- récupérer le document initial
- déposer le document traduit
- imputer le travail effectué (pages traduites par jour)
- déclarer leur(s) spécialité(s) (typologie(s) de documents)

* l'éventuel relecteur (qui est un traducteur) doit pouvoir

- récupérer le document traduit
- déposer le document relu

* le service administratif doit pouvoir

- déclarer les disponibilités des traducteurs (jours travaillés en fonction de leur contrat plein temps / mi temps / temps partiel et de leurs congés)

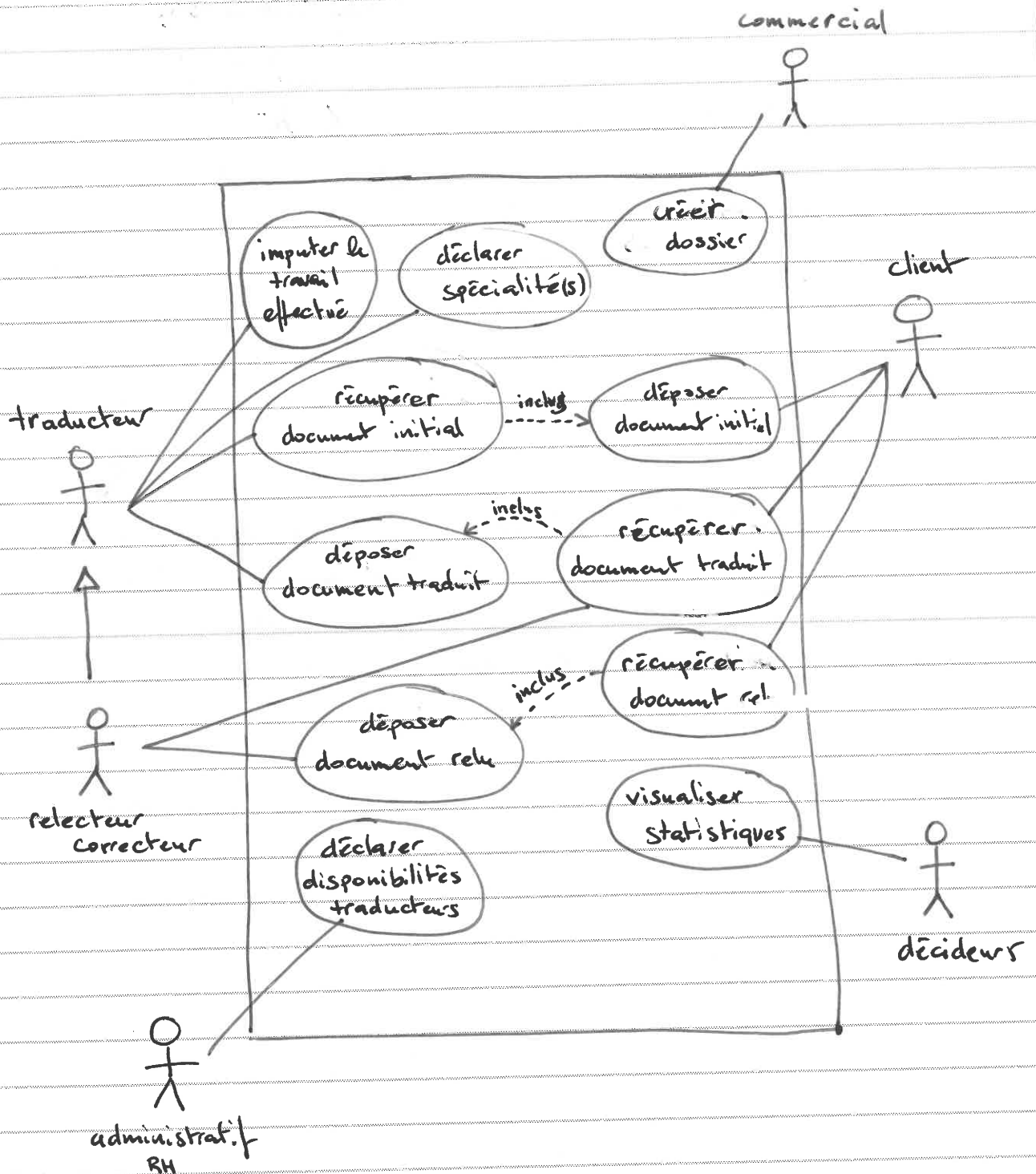
* les décideurs doivent pouvoir

- visualiser différents types de statistiques

Remarques :

- une grande partie de l'activité du service administratif va être prise en charge par le système (planification et transmission des documents)

- certains cas particuliers réels devraient imposer de nouvelles fonctionnalités



(suite des remarques)

par exemple si un traducteur tombe malade, démissionne (ou meurt) s'il est en retard le système doit pouvoir réaffecter son travail ou au moins prévenir un responsable au service administratif (pour entre autre prévenir le commercial de renégocier avec le client)

Pour l'organisation des données du système il faudrait stocker pour chaque

- * pour tous les acteurs (Client, Commercial, ...)
- des informations de connexion au système

* Contrat / Dossier :

- fourni lors du contrat
- la langue initiale du document
 - la langue de traduction
 - la taille du document (en pages)
 - date prévue de dépôt du document initial
 - option relecture ← liés
 - le client

- calculé par le système
- le traducteur et l'éventuel relecteur
 - l'adresse des documents (initial, traduit, relu)
 - date de fin prévue

- * pour le traducteur - la langue maternelle
- les langues parlées (tableau)
- son niveau d'expérience
- ses éventuelles spécialités
- ses disponibilités
- identifiant et mot de passe

Remarques

- en SQL on aura des tables de données supplémentaires pour définir
 - les langues — id de la langue — nom de la langue
 - les niveaux — id du niveau — nom du niveau
 - les spécialités id et nom de la spécialité

- pour les disponibilités on pourrait les stocker dans une table ainsi

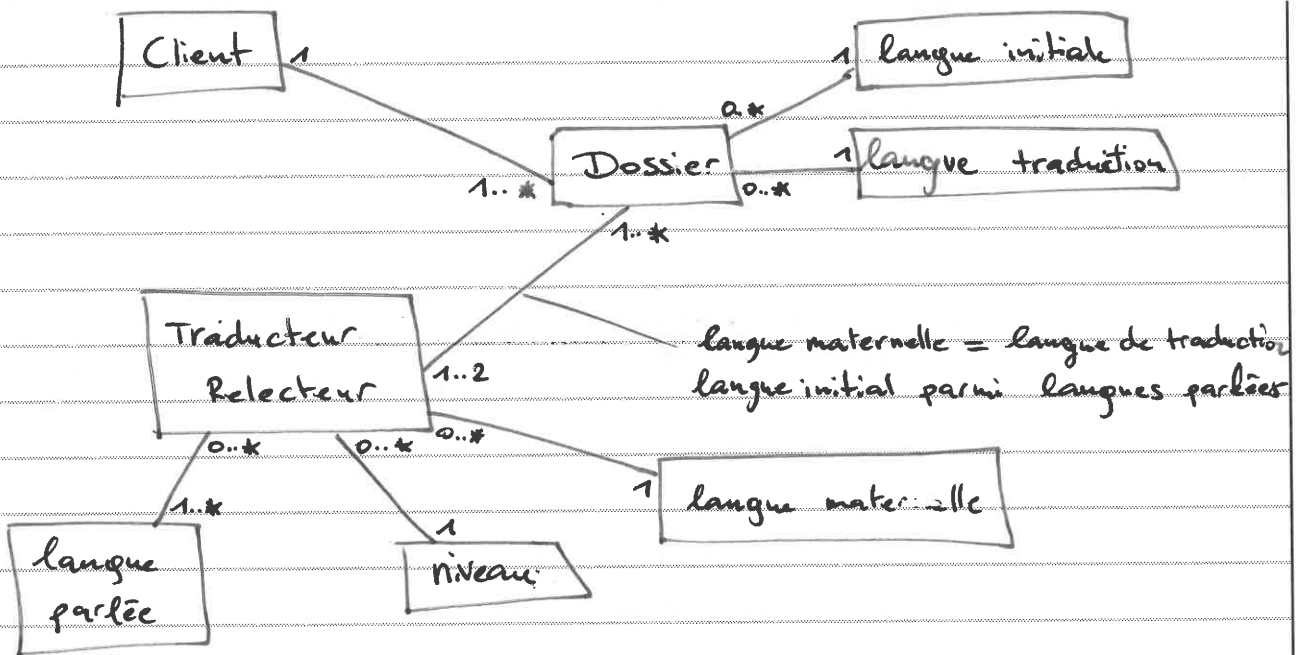
- | |
|-----------------|
| - date |
| - id traducteur |
| - état |

- 0: ne travaille pas
- 1: disponible
- 2: déjà affecté

N°
4.1.12

Intitulé de l'épreuve : Conception Logicielle
Nombre de copies : 2/3

Numérotez chaque page (dans le cadre en bas de la page) et placez les feuilles dans le bon sens.



On aura besoin aussi de la table d'équivalence entre niveau d'expertise et le quota de page traduits / relues quotidiennement (fourni dans l'énoncé)

N°

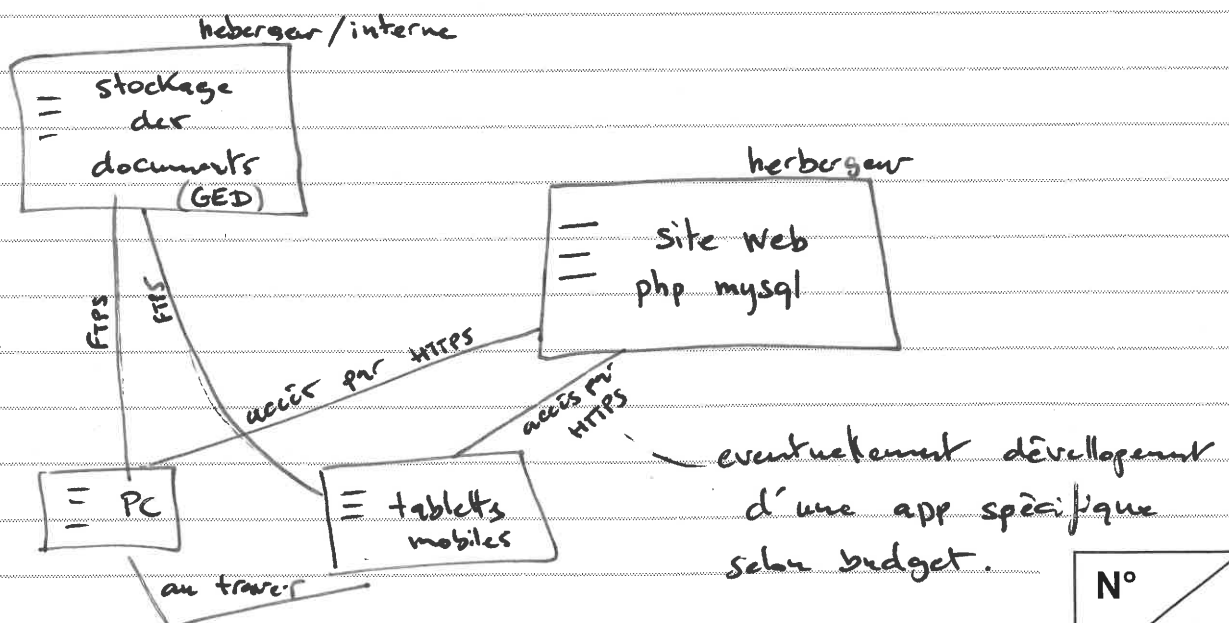
5.1/2

2) pour l'architecture technique et logicielle
tous les acteurs doivent pouvoir accéder de n'importe où
de façon sécurisée au système.

le budget étant contraint et le système
ne contenant pas de données secret défense
ni de données commerciales sensibles, je
propose d'héberger le système sur un site
web classique avec identification (mot de passe et
HTTPS)

On choisira un hébergeur en fonction de la
disponibilité voulue : défini par contrat (SLA)
ou spécifique à certains pays (Chine).

Les documents seront stockés dans un serveur
de document (GED). Selon le niveau de
sécurité voulue et le budget ils pourront
être chez l'hébergeur ou en interne.
Ils seront accessibles par FTP secure (FTPS).



tous les acteurs : commerciaux, clients, décideurs,
traducteurs, administratifs

Pour héberger le site web on pourrait utiliser un hébergeur classique comme OVH avec une configuration LAMP (Linux Apache MySQL PHP) permettant de réaliser le site web à moindre coût.

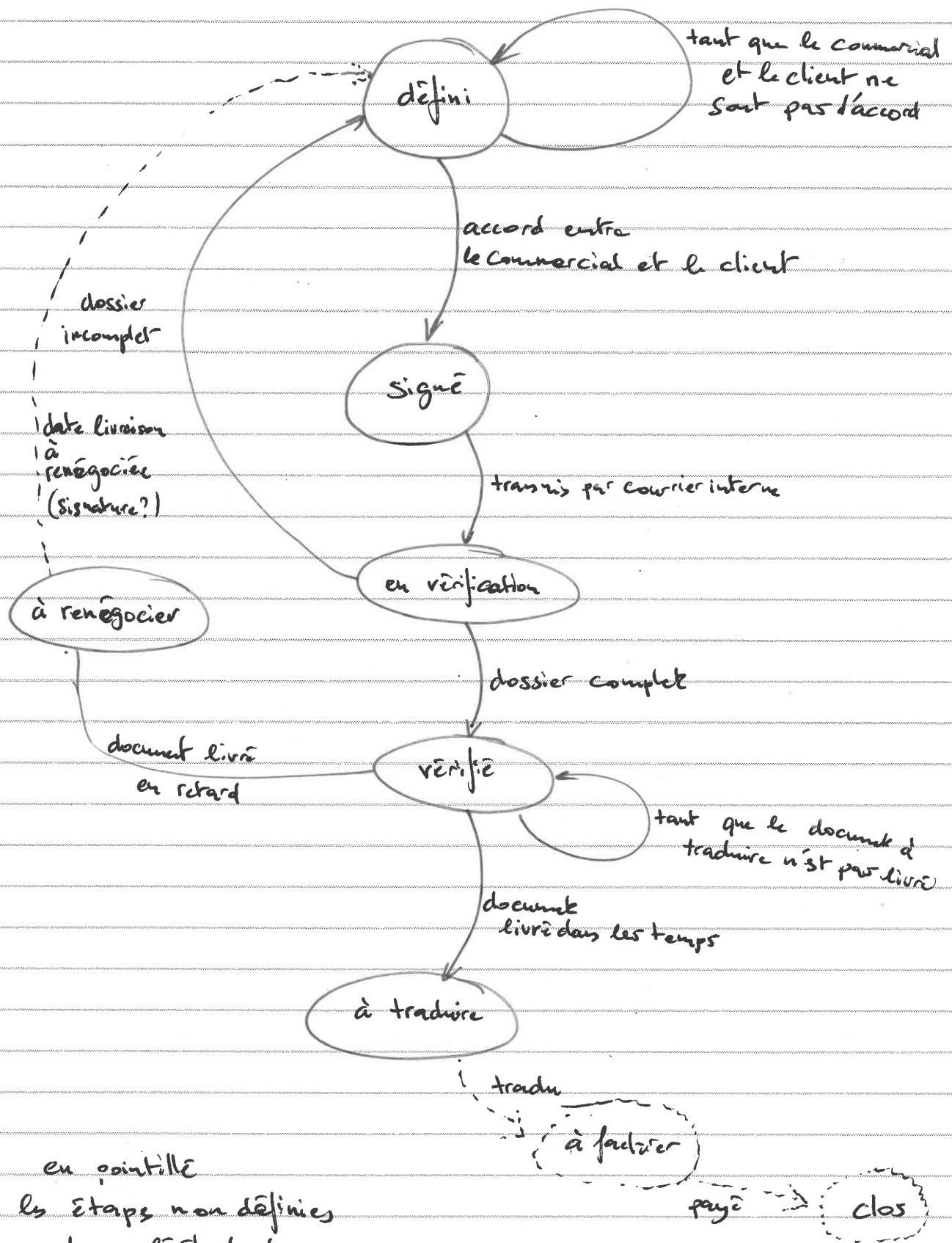
Evidemment chaque acteur n'aura accès qu'aux données le concernant. par exemple, le client ne pourra voir que l'état de sa traduction en cours et y accéder quand elle sera finie. Cela sera garanti par l'accès par identification avec mot de passe.

3) différentes étapes d'un contrat

d'après les éléments présents dans l'étude de cas

- le commercial et le client définissent le contrat
- le commercial et le client signent le contrat
- transmission au service administratif pour vérification
- si le document à traduire n'est pas fourni à temps il faut renégocier la date et donc le contrat

la renégociation n'est pas définie dans l'énoncé, ni le devenir du contrat lorsque le document est livré



en pointillé
 les étapes non définies
 dans l'étude de
 cas

Intitulé de l'épreuve : Conception Logicielle

Nombre de copies : 3/3

Numérotez chaque page (dans le cadre en bas de la page) et placez les feuilles dans le bon sens.

4) on a une traduction donnée
donc on a les entrées suivantes

- lang-init : langue initiale du document à traduire
- lang-trad : langue dans laquelle traduire le document
- date-début : date de remise du document
- nb-pages : nombre de pages à traduire

1 - On sélectionne tous les traducteurs dont la langue maternelle = lang-trad et dont une langue parlée = lang-init

* Si on obtient un tableau vide

→ on n'a pas de traducteur pour ces langues

* Si le tableau n'est pas vide le mettre dans une table = tab-trad-langue-ok

```
SELECT * FROM table-traducteur
WHERE lang-trad = lang-maternelle
lang-init IN lang-parlées
```

N°

9/17

2 - pour chaque traducteur in table-trad-or

- boucle sur les traducteurs
- récupérer le niveau = niv-trad
 - en déduire le nombre de jours nécessaire avec - la table équivalence présente dans l'énoncé
 - niv-trad
 - nb-pags \Rightarrow nbre-jour-trad
 - calculer avec la table de disponibilité du traducteur et nbre-jour-trad la date de fin de traduction estimée \Rightarrow date-fin

3 - affecter au traducteur ayant la date de fin la plus proche cette traduction

Partie I

1) Cloud Computing

C'est l'utilisation de machines, d'applications ou de données sur des serveurs présents sur le réseau. Ceux-ci sont distribués et répliqués dans plusieurs endroits : ferme de serveurs (data center) assurant que l'utilisateur ne perde rien en cas de panne. L'utilisateur a un point d'accès mais ne connaît pas la localisation de ces machines, applications ou données.

Amazon Web Service propose la virtualisation de machines, Google (Doc, Photos, Mail, Sheet, ...) propose la virtualisation d'un grand nombre d'applications ainsi que le stockage de données. Microsoft avec Office 365 a mis sur le cloud toute sa collection d'applications. Dropbox était un précurseur pour le stockage des données.

2)

- BDD relationnelles / NoSQL

BDD relationnel est un système permettant de stocker, modifier et accéder aux données à travers d'un langage normalisé (SQL) à des données stockées dans des tables.
exemples : Oracle, MySQL, PostgreSQL
elles sont plus sûres mais plus lourdes à maintenir.

NoSQL est un stockage de données dans des fichiers-textes (XML, json, ..) l'accès aux données est donc moins lourd et privilégié pour de petites bases de données : librairies javascript, sqlite, ..