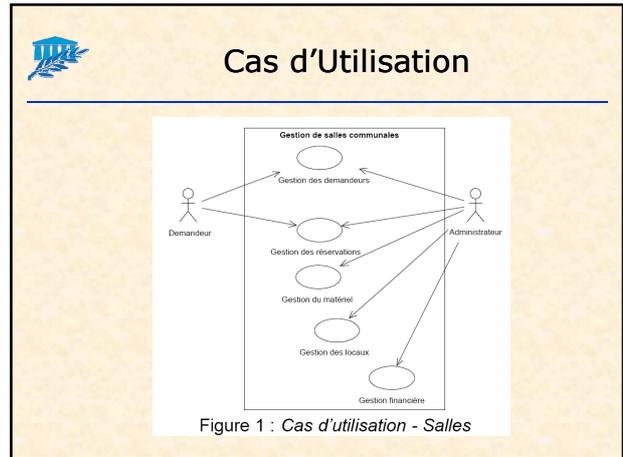


Modélisation Objet avec UML

Use Case

- A **use case** is a kind of classifier representing a coherent unit of functionality provided by a system, a subsystem, or a class as manifested by sequences of messages exchanged among the system (subsystem, class) and one or more outside interactors (called actors) together with actions performed by the system (subsystem, class).
- An **actor** defines a coherent set of roles that users of an entity can play when interacting with the entity. An actor may be considered to play a separate role with regard to each use case with which it communicates.
- An **extension point** is a reference to one location within a use case at which action sequences from other use cases may be inserted. Each extension point has a unique name within a use case, and a description of the location within the behavior of the use case.

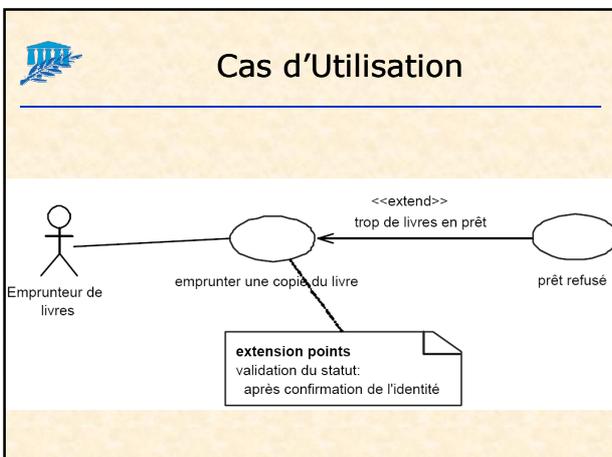
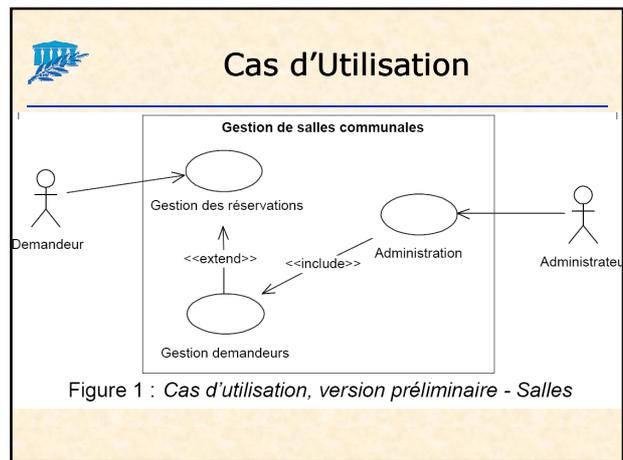
D'après le cours de Pascal André



Cas d'Utilisation

Relations

- **Association** - The participation of an actor in a use case; that is, instances of the actor and instances of the use case communicate with each other. This is the only relationship between actors and use cases.
- **Extend** - An extend relationship from use case A to use case B indicates that an instance of use case B may be augmented (subject to specific conditions specified in the extension) by the behavior specified by A. The behavior is inserted at the location defined by the extension point in B, which is referenced by the extend relationship.
- **Generalization** - A generalization from use case C to use case D indicates that C is a specialization of D.
- **Include** - An include relationship from use case E to use case F indicates that an instance of the use case E will also contain the behavior as specified by F. The behavior is included at the location which defined in E.



Cas d'Utilisation

- Points clés du diagramme des cas d'utilisation
 - Abstrait
 - Granularité : entre découpage fonctionnel et modulaire
 - Lisibilité
 - Description textuelle

Cas d'Utilisation

Cas d'utilisation : Gestion des réservations

acteurs primaires : Demandeur

invariant : Unicité de réservation
Une salle n'est pas réservée pour deux demandeurs différents au même moment.

description

La gestion des réservations comprend la réservation des salles, la consultation des réservations, l'annulation des réservations.

cas : Réservation

Les éléments de la réservations sont saisis et recherchés dans la base en fonction de critères donnés : salle, demandeur, matériel, durée, manifestation, date. A tout moment, il est possible de consulter le planning des réservations en cours. Si tous les éléments sont corrects et qu'il n'y a pas de conflit de réservation, le montant est calculé et la réservations confirmée. Le numéro de la réservation est fourni par le système au demandeur.

Cas d'Utilisation

Cas d'utilisation : Gestion des réservations (suite)

cas : Consultation des réservations
La consultation prend plusieurs formes : recherche d'une réservation par son numéro, par demandeur, par date ou par salle, consultation du planning des réservations.

cas : Annulation d'une réservation
Après recherche de la réservation, le demandeur confirme sa suppression.

exceptions

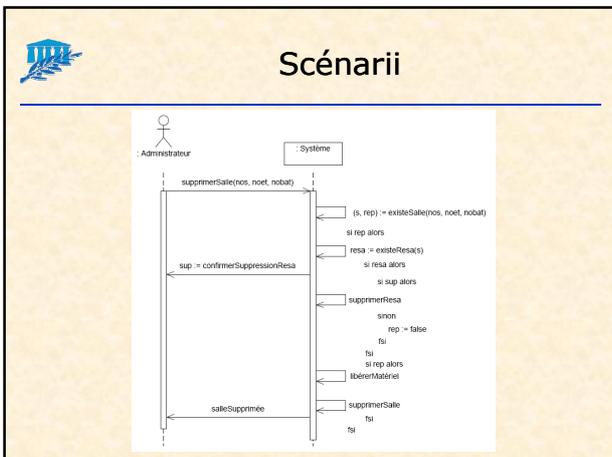
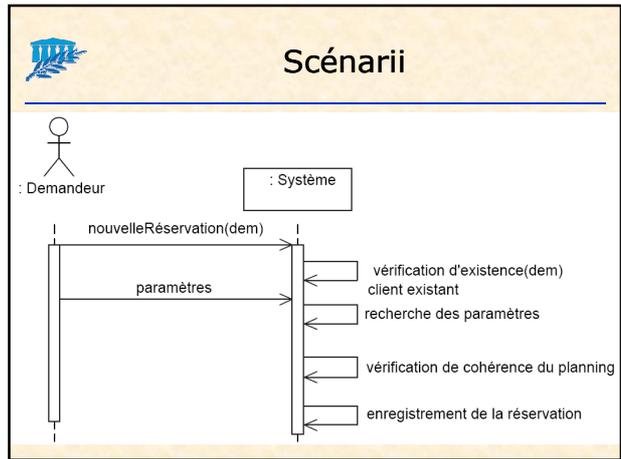
cas : Réservation avec un demandeur inexistant

précondition : Le demandeur n'est pas inscrit.

résultat : Il y a création du demandeur (voir UC Gestion des demandeurs) avant d'établir la réservation.

Scénarii

- diagramme de séquence simplifié
 - voir diagramme de séquence
 - objets (acteurs + système)
 - envoi de message (paramètres...)
 - commentaires



Activité

- diagramme d'activité étendu
 - voir diagramme d'activité
 - activités, actions
 - structures de contrôle
 - envoi/réception de message ou signaux (paramètres...)
 - flots et synchronisations
 - couloirs (noms, objets, rôles...)

