

Site de billetterie

Ticket to the Moon

Date de mise à jour : 27/06/2022

Version : 2.1

Nom	FENGAROL	Numéro d'auditeur	XX
------------	----------	--------------------------	----

Table des matières

1.	Objectif du document	3
2.	Cas d'utilisation.....	4
2.1.	Cas d'utilisation d'un gestionnaire.....	4
2.1.1	Créer un évènement.....	5
2.1.1	Créer un tarif	8
3.	Regroupement des classes	12
3.1.	Groupe domaine	12
3.2.	Groupe domaine et cycle de vie	13
3.3.	Groupe Service.....	14
3.4.	Groupe interface utilisateur et système	14
4.	Annexes	15
4.1.	Terminologie	15

1.Objectif du document

Le but de ce document est de définir une analyse du projet d'un site de Billetterie en ligne pour des spectacles appelé ***Ticket to the Moon***. Ce document a pour but d'analyser les cas d'utilisations identifiés dans l'expression des besoins, et d'en extraire les objets ou domaines d'objets qui interviendront dans leur réalisation. L'interaction entre ces objets sera décrite par le biais de diagramme de séquence.

Les domaines d'objet seront regroupés de la façon suivante :

- <<**Entity**>> : Les objets métiers
- <<**Lifecycle**>> : Les objets d'interfaces avec la base de données qui manipulent les entités (DAO)
- <<**Control**>> : Les services (ou façades) qui auront pour but de réaliser le cas d'utilisation
- <<**Boundary**>> : Les objets d'interfaces avec l'utilisateur

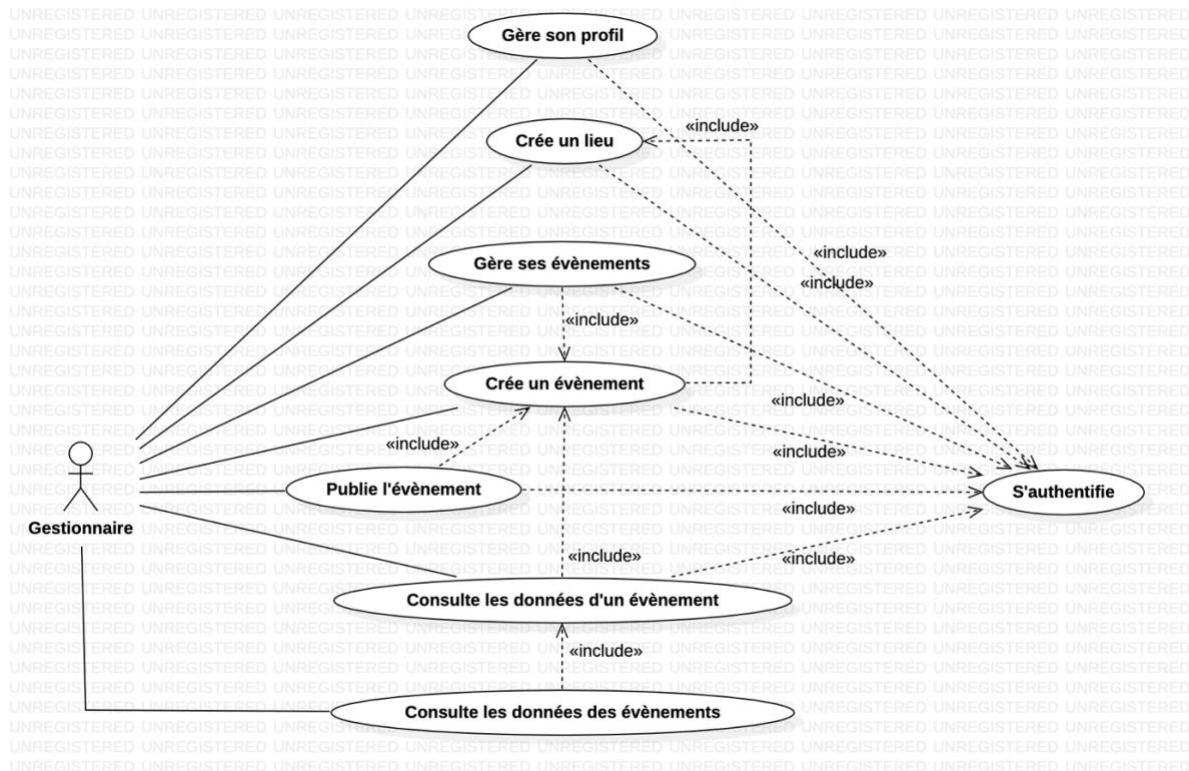
Il ne s'agit pas ici d'identifier les classes précises qui composeront l'application mais uniquement des domaines de classes. Seule la liste des classes entités sera exhaustive pour les cas d'utilisation étudiés. Les noms de méthodes utilisés dans les diagrammes de séquences ont uniquement pour but d'aider à leur compréhension, mais ne seront peut-être pas repris lors de la réalisation. Tous les cas d'utilisations ne seront pas analysés dans ce document. Seuls ceux nécessaires à l'identification d'une majorité pertinente des entités seront étudiés.

2. Cas d'utilisation

2.1. Cas d'utilisation d'un gestionnaire

Vue macro

Un administrateur doit créer les comptes des gestionnaires. Le gestionnaire doit s'authentifier car toutes ses actions nécessitent une authentification. Il peut gérer son profil, créer et gérer ses événements, consulter les données d'un événement ou des événements dans leur ensemble. Il peut diffuser un événement ce qui nécessite sa création.



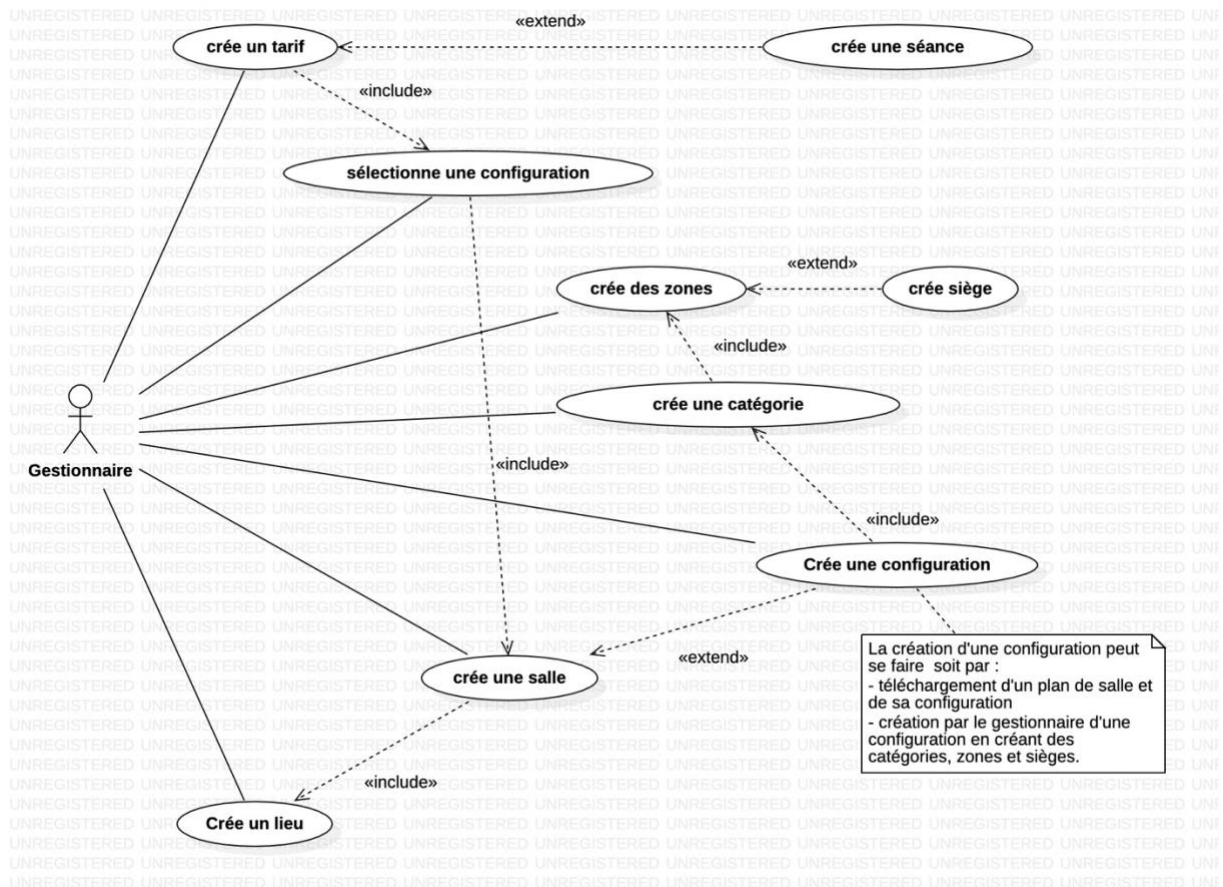
Vue micro pour créer un événement

C'est le cas d'utilisation le plus important pour le gestionnaire.

Un événement nécessite la création d'un lieu composé d'une ou de plusieurs salles et d'une configuration de salle comprenant une catégorie (balcon, orchestre...), une zone (zone A, B, C...), la liste des places numérotées si c'est le cas et un quota de places.

La création d'une configuration n'est pas obligatoire puisque le site *Ticket to the Moon* offre la possibilité de télécharger des plans de salles et des configurations toutes faites.

La création d'un tarif nécessite une configuration de salle (téléchargée ou pas) avec au moins une zone (et donc une catégorie). Une catégorie peut avoir 1 ou plusieurs zones et une zone être composée de 1 ou plusieurs catégories. Une zone peut avoir plusieurs prix dans le cas des séances par exemple. Une zone a 0 prix dans le cas des places indisponibles car utilisées par le personnel technique ou car le siège est cassé ou derrière un poteau... Ces places ne seront donc ainsi pas comptabilisées.



2.1.1 Créer un évènement

Liste des objets candidats

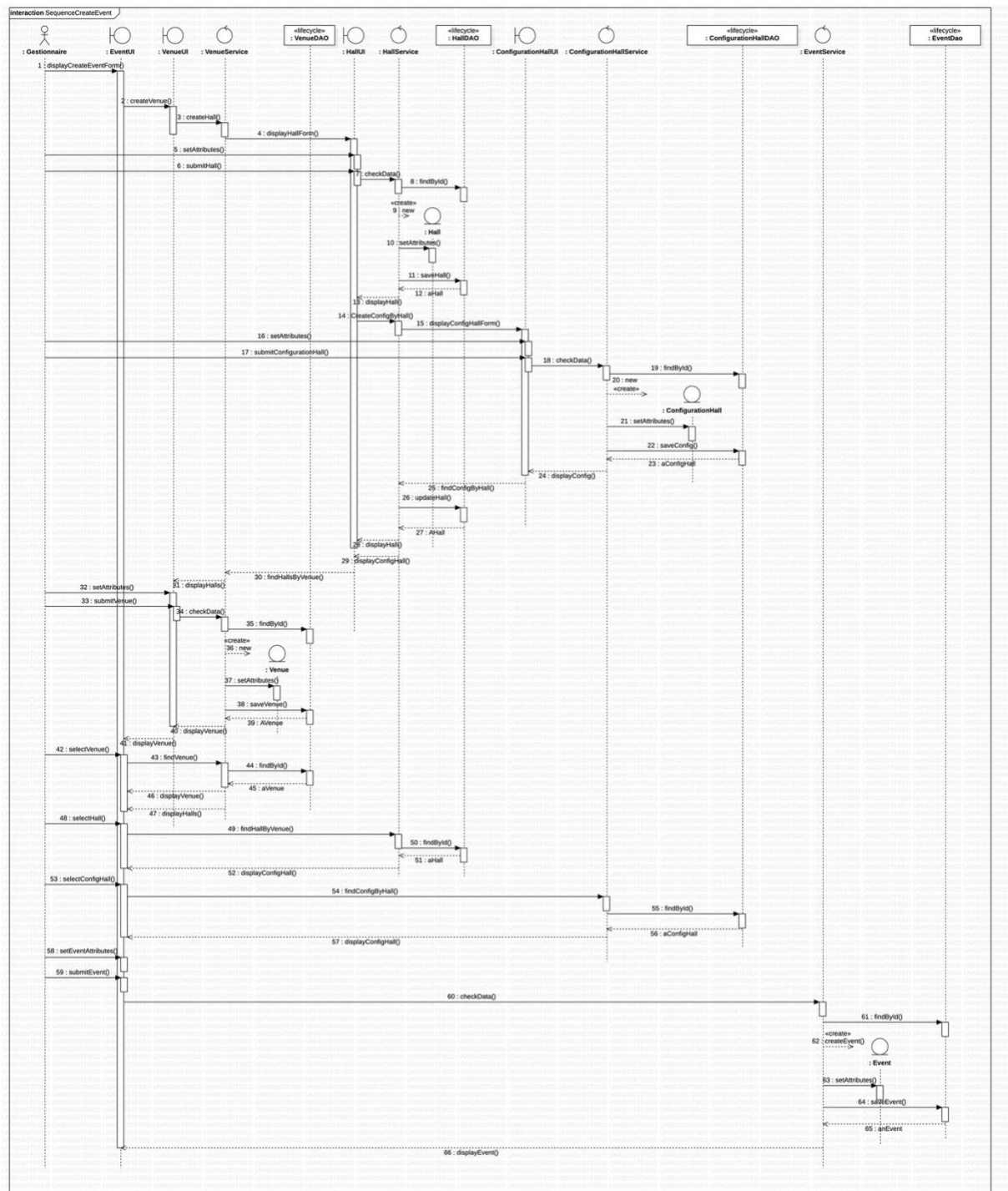
- <<Entity>> :
 - Event : informations de l'évènement
 - Venue : Informations du lieu de l'évènement
 - Hall : Salle où se passe l'évènement
 - ConfigurationHall : Configuration de la salle avec les entités qui en découlent suivantes :
 - Category : une catégorie de place (balcon, orchestre, fosse...)
 - Area : une zone de prix (zone A, zone B, zone C...) dans une catégorie et concernant ou pas des sièges.
 - Seat : la liste des places qui se rapportent à une zone de prix
- <<lifecycle>> : EventDAO, VenueDAO, HallDAO, ConfigurationHallDAO
- <<control>> : EventService, VenueService, HallService, ConfigurationHallService
- <<boundary>> : EventUI, VenueUI, HallUI, ConfigurationHallUI

Description des interactions entre objets

Il est à noter pour simplifier et permettre une meilleure lisibilité que :

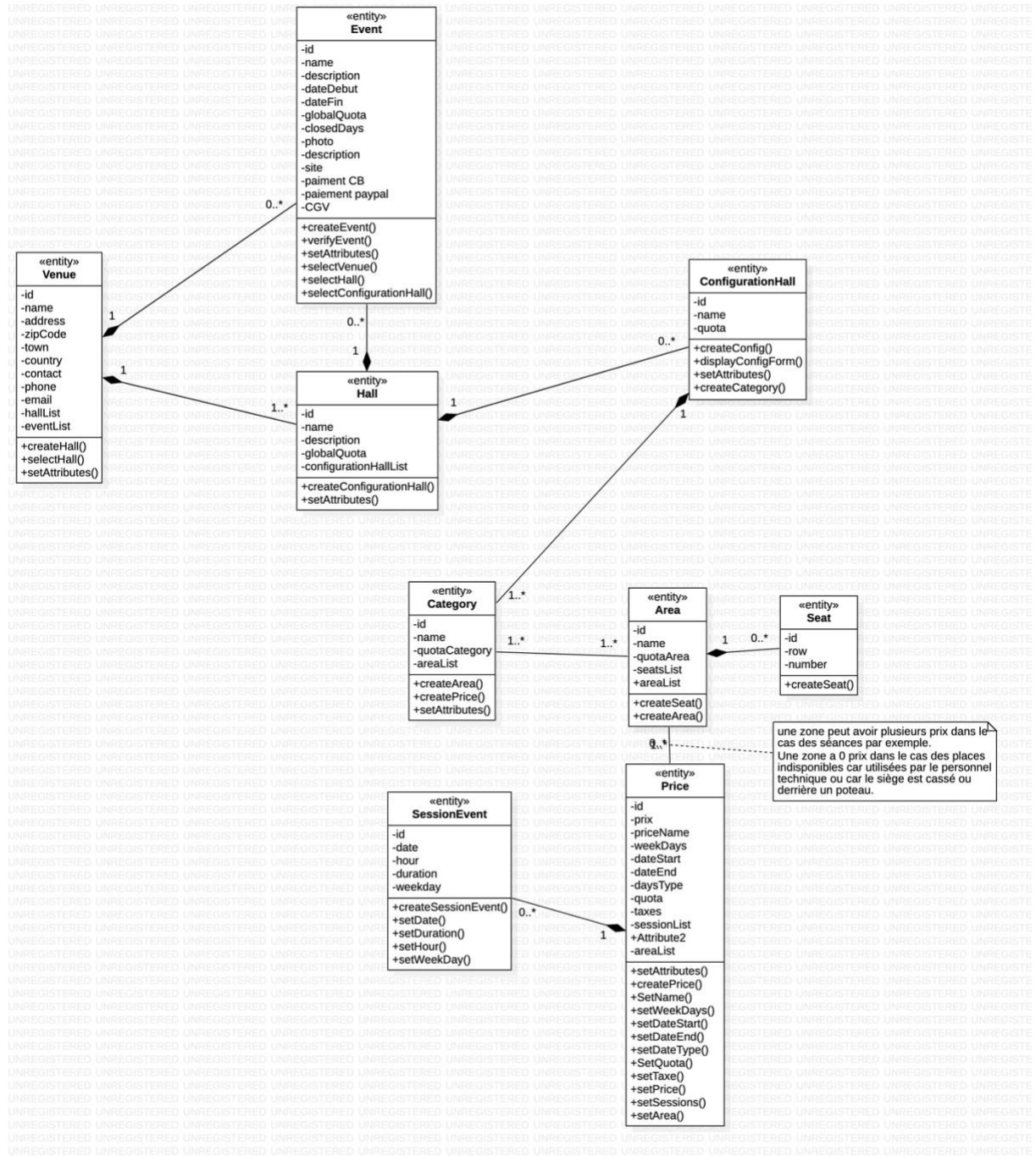
La création d'un évènement inclut les use case créer une salle et créer une configuration. Le niveau de détail de créer une configuration n'est pas développé ici. Celui-ci nécessite la création de catégories, zones et sièges si nécessaire (Category, Area, Seat). Ces entités sont toutefois disponibles dans la description des classes. La création d'un prix fait aussi l'objet d'un cas d'utilisation à part développé dans ce document.

Sont détaillés ici uniquement les attributs qui créent des entités. La méthode appelée `setAttributes()` reprends tous les attributs qui sont détaillés dans le diagrammes des classes. Cela toujours dans un souci de lisibilité.

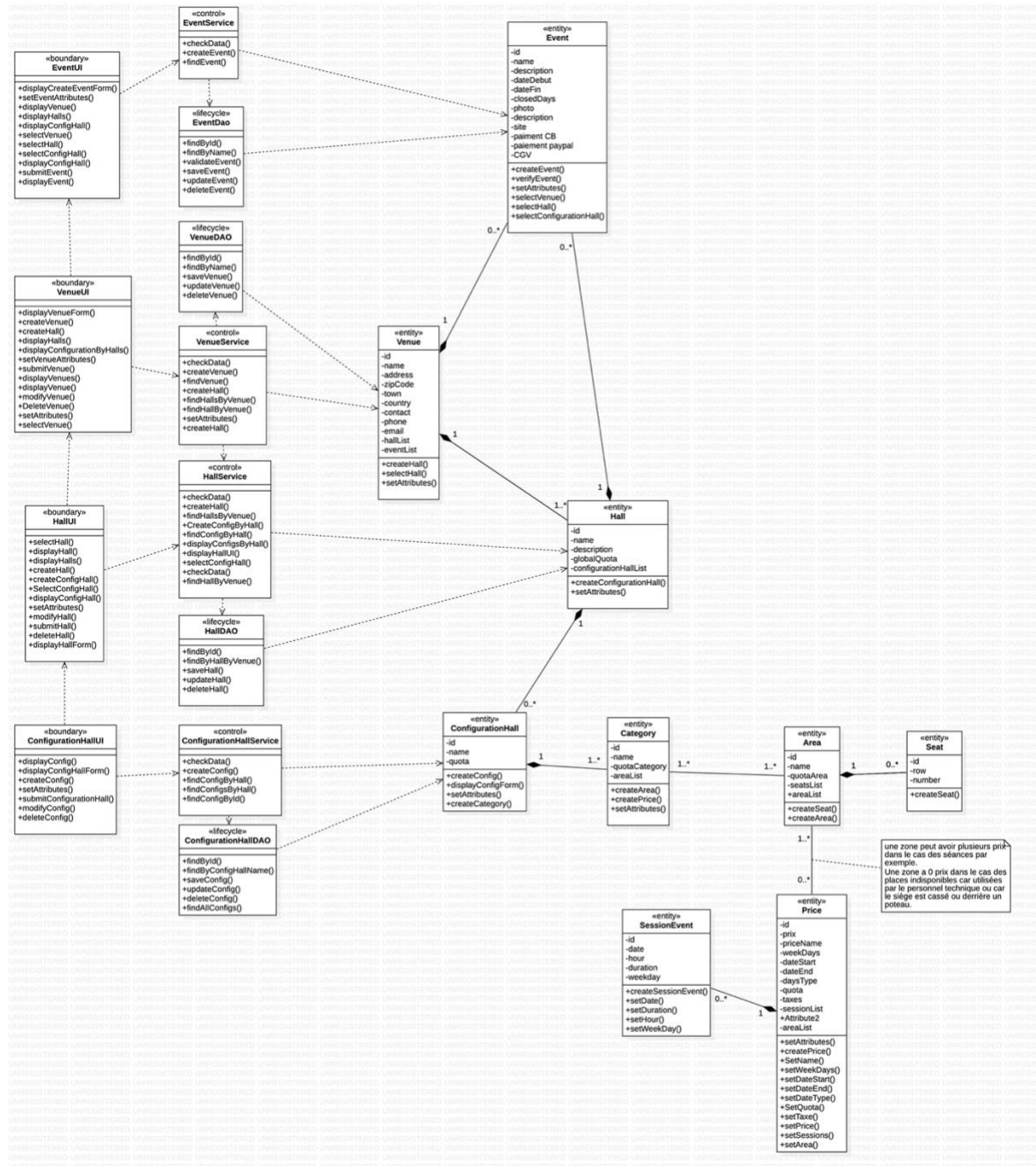


Description des classes

Détail des relations entre entités



Ensemble des classes du cas d'utilisation



2.1.1 Créer un tarif

Liste des objets candidats

- <<Entity>> :
 - Price : informations d'un prix
 - ConfigurationHall : Configuration de la salle avec les entités qui en découlent suivantes :
 - Category : une catégorie de place (balcon, orchestre, fosse...)
 - Area : une zone de prix (zone A, zone B, zone C...) dans une catégorie de place et concernant ou pas des sièges.
 - Seat : la liste des places qui se rapportent à une zone de prix

- Session : séance d'un spectacle avec une spécificité de dates, jours, heures.
- <<lifecycle>> : PriceDAO, ConfigurationHallDAO, AreaDAO, SessionDAO
- <<control>> : PriceService, ConfigurationHallService, AreaService, SessionService
- <<boundary>> : PriceUI, SessionUI

Description des interactions entre objets

La création d'un tarif nécessite une configuration de salle (téléchargée ou pas) avec au moins une zone (et donc une catégorie). Une catégorie peut avoir 1 ou plusieurs zones et une zone être composée de 1 ou plusieurs catégories. Une zone peut avoir plusieurs prix dans le cas des séances par exemple. Une zone a 0 prix dans le cas des places indisponibles car utilisées par le personnel technique ou car le siège est cassé ou derrière un poteau... Ces places ne seront donc ainsi pas comptabilisées dans le chiffre d'affaires.

Étapes de création sans séances :

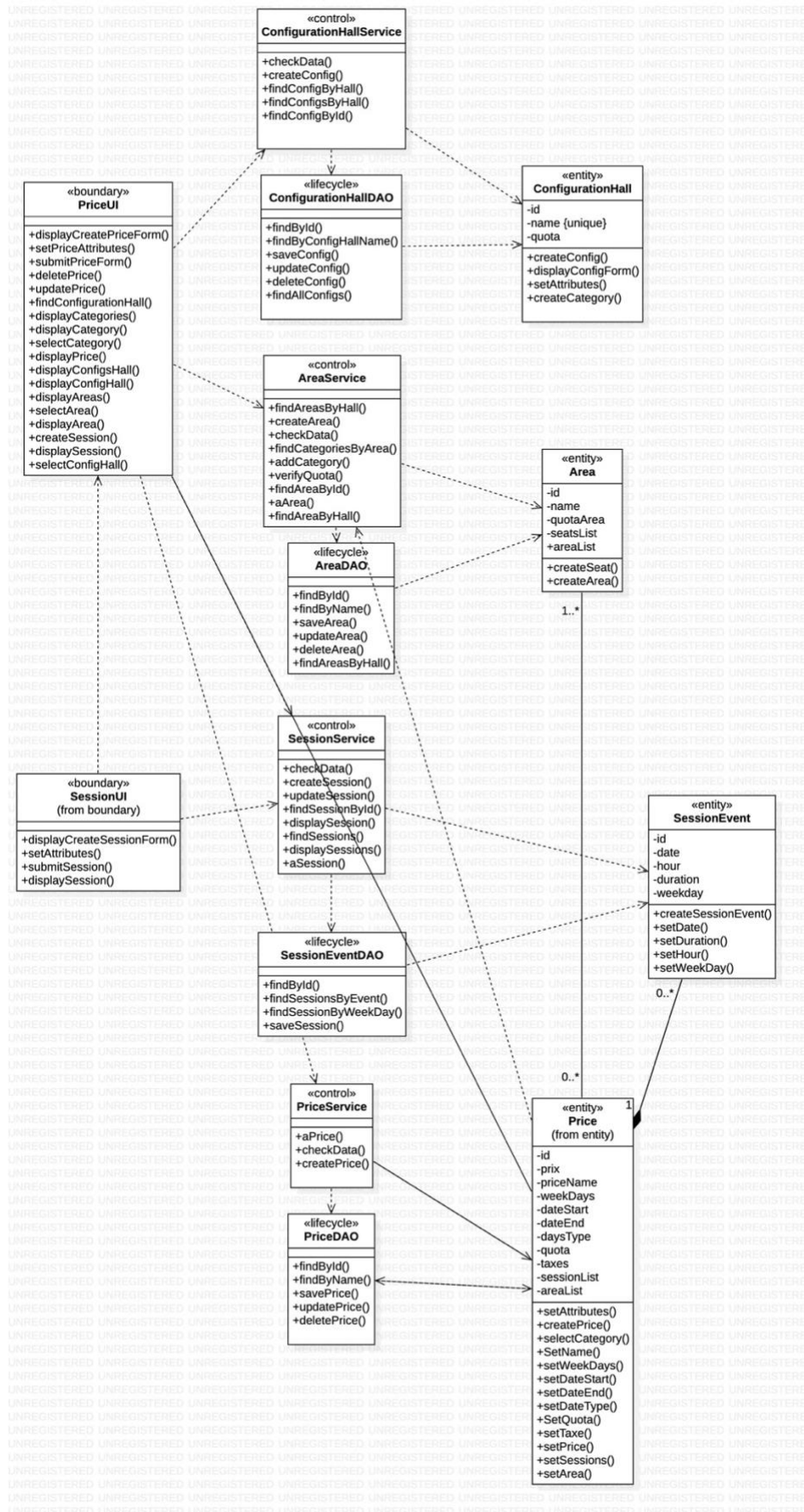
- Sélection d'une configuration pour l'évènement
- Sélection d'une zone : zone A
- Définir prix, nom

Étapes de création avec séances :

- Sélection d'une configuration pour l'évènement
- Sélection d'une zone : zone A
- Création d'une ou de plusieurs séances
- Définir prix, nom

C'est ce dernier cas que nous développerons ici.

Description des classes



3.Regroupement des classes

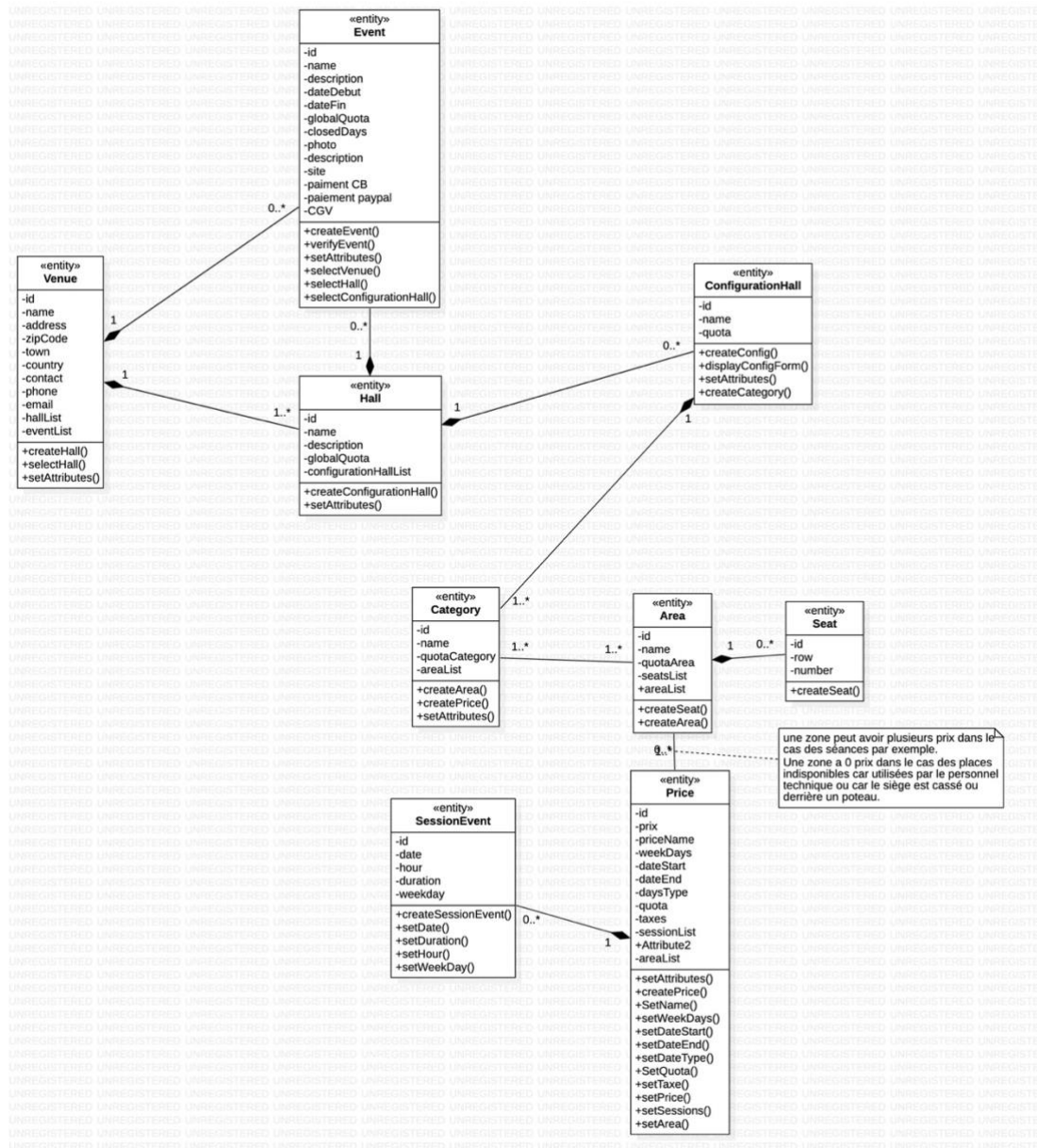
Les attributs et opérations ne sont pas exhaustifs et peuvent être sujet à changement d'appellation lors de la phase développement.

3.1. *Groupe domaine*

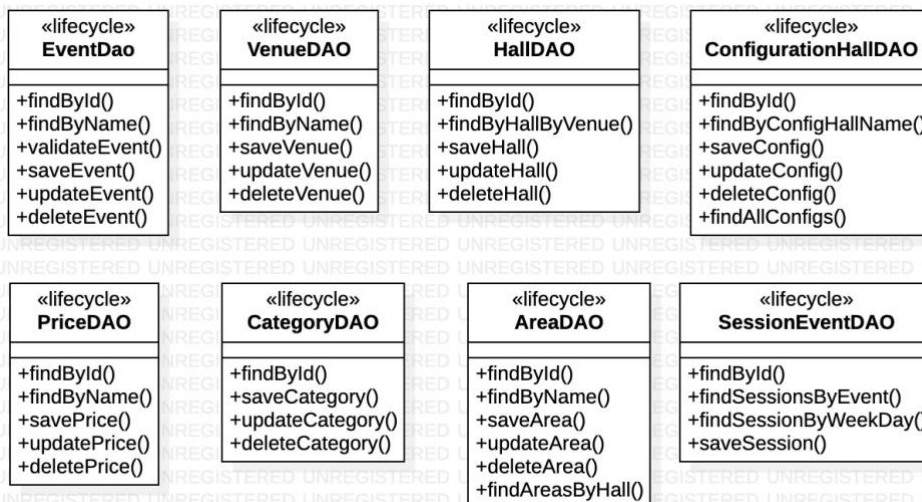
- Event : informations de l'évènement
- Venue : Informations du lieu de l'évènement
- Hall : Salle où se passe l'évènement
- ConfigurationHall : Configuration de la salle avec les entités qui en découlent suivantes :
 - Category : une catégorie de place (balcon, orchestre, fosse...)
 - Area : une zone de prix (zone A, zone B, zone C...) dans une catégorie et concernant ou pas des sièges.
 - Seat : la liste des places qui se rapportent à une zone de prix
- Price : informations d'un prix
- SessionEvent : heure et date de la représentation pour ce prix. La séance d'un spectacle peut avoir une spécificité de dates, jours, heures.

Notons que :

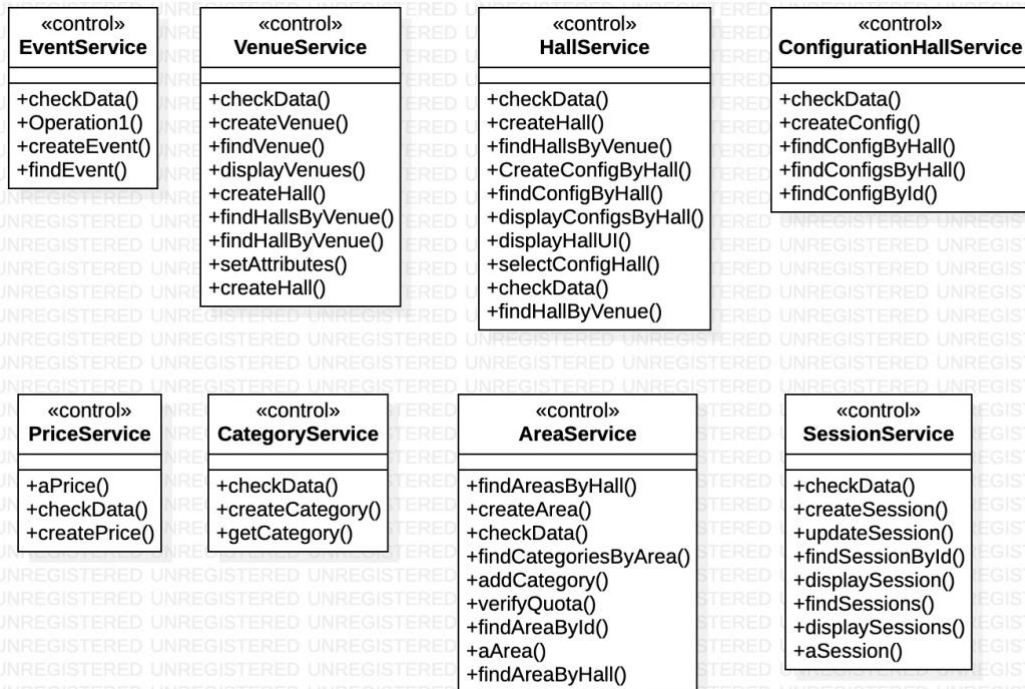
- Un évènement peut avoir plusieurs configurations. Par exemple : une configuration du soir en place debout et assises et une configuration dans la journée avec que des places assises. Une configuration pour les vacances etc...
- Une zone peut avoir plusieurs prix : un pour la semaine, les jours fériés et les week-ends en soirée ou pas.



3.2. Groupe domaine et cycle de vie

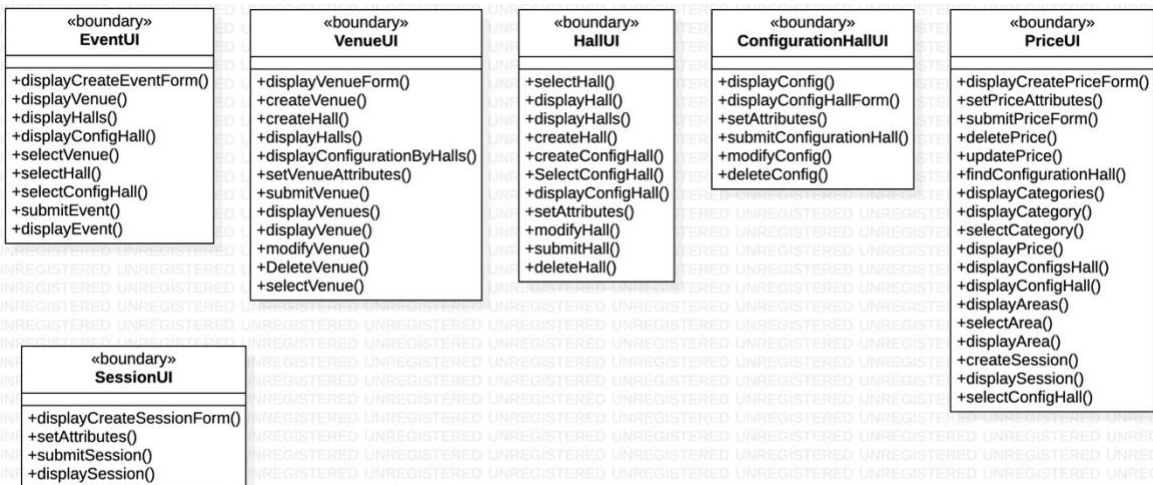


3.3. Groupe Service



3.4. Groupe interface utilisateur et système

L'interface utilisateur HallUI pourrait être inclus dans VenueUI afin de simplifier les démarches utilisateur.



4. Annexes

4.1. Terminologie

ENTITY	Désigne les objets métier de notre application, des informations durables et persistantes.
BOUNDARY	Objets qui servent à modéliser les interactions entre le système et ses acteurs.
CONTROL	Objets utilisés pour représenter la coordination, l'enchaînement, la gestion et le contrôle d'autres objets. Façade entre les objets Boundary et Entity.
LIFECYCLE	Objets responsables de trouver les objets Entity et permettant la liaison à un SGBDR