|  |
| --- |
| Projet Modélisation |
| Festival d’humour |
|  |

13/05/2016

Contenu[[1]](#footnote-1)

[Introduction 3](#_Toc450917385)

[Analyse globale 3](#_Toc450917386)

[Contexte 3](#_Toc450917387)

[Conclusion 5](#_Toc450917388)

[Documentation de classe 6](#_Toc450917389)

[Documentation des associations 16](#_Toc450917390)

[Dictionnaire de données 17](#_Toc450917391)

[Acteurs et rôles 24](#_Toc450917392)

[Acteurs et événements 24](#_Toc450917393)

[Description narrative 25](#_Toc450917394)

[Diagramme de contexte 44](#_Toc450917395)

[Diagramme de package 45](#_Toc450917396)

[Diagramme de cas d’utilisation Gestion de contrat 46](#_Toc450917397)

[Gestion d’artiste 47](#_Toc450917398)

[Gestion des catégories d’artiste 48](#_Toc450917399)

[Gestion d’agence 49](#_Toc450917400)

[Gestion de personnel 50](#_Toc450917401)

[Gestion d’évènement 51](#_Toc450917402)

[Prototype d’IHM Sommaire des contrats 52](#_Toc450917403)

[Détail d’un contrat 53](#_Toc450917404)

[Confirmation d’enregistrement Confirmation de modification Confirmation de suppression 54](#_Toc450917405)

[Erreur de format Erreur de champ vide 54](#_Toc450917406)

[Détails d’un statut de contrat 55](#_Toc450917407)

[Confirmation d’enregistrement Confirmation de modification Confirmation de suppression 55](#_Toc450917408)

[Erreur de champ vide 55](#_Toc450917409)

[Sommaire des catégories d’artiste 56](#_Toc450917410)

[Détails d’une catégorie d’artiste 57](#_Toc450917411)

[Confirmation d’enregistrement Confirmation de modification Confirmation de suppression 57](#_Toc450917412)

[Erreur de champ vide 57](#_Toc450917413)

[Détails d’un artiste 58](#_Toc450917414)

[Confirmation d’enregistrement Confirmation de modification Confirmation de suppression 59](#_Toc450917415)

[Erreur de champ vide 59](#_Toc450917416)

[Détails des engagements 60](#_Toc450917417)

[Confirmation d’enregistrement Confirmation de modification Confirmation de suppression 61](#_Toc450917418)

[Erreur de format Erreur de champ vide 61](#_Toc450917419)

[Détails des exigences 62](#_Toc450917420)

[Confirmation d’enregistrement Confirmation de modification Confirmation de suppression 63](#_Toc450917421)

[Erreur de format Erreur de champ vide 63](#_Toc450917422)

[Diagramme de classe 64](#_Toc450917423)

# Introduction

Suite à la commandite que nous avons reçue, nous avons réalisé le mandat qui nous avait été confié. Ce mandat, comme entendu comportait les analyses globales et fonctionnelles ainsi que la réalisation des parties suivantes :

* Gestion des contrats
* Gestion des artistes
* Gestion des catégories d’artiste

Pour réaliser tout cela, plusieurs documents ont étés produits. Tout d’abord un diagramme de classe général comportant également une documentation précise des différents éléments qui le composent. Deuxièmement, des diagrammes de cas d’utilisations ont étés mis en forme afin de de spécifier le fonctionnement particulier de chaque parties du mandat. Des descriptions narratives de chaque fonctionnalité ont également était réalisées dans le but de formaliser par le texte les différentes manières dont les différentes parties du projet pourraient être utilisées une fois celui-ci terminé. Enfin une série d’interfaces correspondantes aux descriptions narratives citées plus haut ont aussi étés réalisées dans le but de donner une idée de la manière dont sera présenté le projet une fois finis.

# Analyse globale

Contexte :

L’application permettra entre autres de :

* Gérer le personnel au travail
  + Leurs tâches lors d’un évènement;
  + L’état des tâches au cours du développement;
  + Émettre des rapports d’efficacité;
  + Avoir des compte-rendus globaux des tâches;
* Gérer les contrats
  + Gérer les statuts de contrats;
  + Gérer les artistes;
    - Ainsi que les catégories d’artistes
    - Les photos liées à un artiste
    - Les photos officielles de l’artiste lié à un contrat
  + Déterminer un responsable d’agence;
  + Déterminer un responsable pour le diffuseur (vous);
  + Enregistrer les contrats avec des signataires uniques;
  + Noter chaque exigence des agences;
    - Les dates d’échéance et
    - Les montants en jeu
  + Noter les engagements;
    - Leur durée,
    - La nature,
    - La date ainsi que
    - Le lieu
* Gérer l’embauche
  + Embaucher du personnel local;
  + Embaucher des artistes pour prestations;
* Gérer les évènements
  + Créer un évènement global;
    - Festi-rire
  + Associer des évènements subséquent;
    - Un spectacle,
    - Un diner,
    - Un gala,
    - Un concours ou
    - Autre, à déterminer par le gestionnaire.
  + Lier des artistes et leurs contrats;

La majorité de ces informations auront des cases de description et de commentaires permettant aux gestionnaires de mieux se repérer lors de l’organisation des évènements. Nous croyons que le volume d’employés qui utiliseront cette application sera relativement restreint, car pour le moment, nous n’avons pas d’information sur qui utilisera réellement le système mis en place. Pour l’instant, nous savons qu’une secrétaire fera office d’utilisateur principal, administrant la majorité des données à la place du gestionnaire. Nous savons aussi que l’application comportera une fonction pour gérer les tâches des employés, mais pas si les employés eux-mêmes y auront accès pour y inscrire des mises à jour ou non, ce qui changerait drastiquement le volume d’employés utilisant le système en création.

# Conclusion

Au terme de notre projet, il était question de faire l’analyse globale et l’analyse fonctionnelle du package contrat l’application de gestion du festival d’humour qui se tient à Jonquière. Nous avons pu faire:

* Le diagramme de classe du package contrat et sa documentation
* Diagramme de contexte
* Diagramme de package
* Dictionnaire des données
* Les descriptions narratives d’ajout et sommaire contrat, ajout et sommaire des catégories d’artistes, ajout d’un statut d’un contrat
* Les cas d’utilisation de gestion des contrats, d’artistes, gestion d’une catégorie d’artistes, de l’agence
* Les interfaces des messages des erreurs
* Les interfaces d’ajout et sommaire des contrats, d’ajout et sommaire d’une catégorie d’artistes, d’ajout d’un engagement, d’un statut

Au niveau de l’avancement, l’objectif qu’on s’était fixé a été atteint. Mais cependant, une amélioration s’impose au niveau des cas d’utilisation.

# Documentation de classe

|  |  |
| --- | --- |
| Diffuseur | Le diffuseur est l'organisme organisateur de l'événement. Une occurrence unique est pour le moment en place. Cependant il serait éventuellement possible d'en créer, modifier ou supprimer un si le besoin s'en faisait sentir. |
| noDiffuseur | Le champ noDiffuseur est un entier (int) auto-incrémenté permettant d'identifier de manière unique un diffuseur dans la base de données. Sa présence est obligatoire. |
| nom | Le champ nom est une chaine de caractères de type Varchar(20) permettant d'identifier facilement un diffuseur dans la base de données. Sa présence est obligatoire. |
| courriel | Le champ courriel est une chaine de caractères de type varchar(30) permettant d'identifier l'adresse email associée à l'agence. Il se compose ainsi : machin@truck.quelquechose. Sa présence est obligatoire |

|  |  |
| --- | --- |
| Photo | La classe Photo représente une occurrence de la photo utilisée pour un contrat donné avec un artiste donné. Une occurrence de cette classe peut être créer lorsqu’un contrat relié à un artiste a déjà été enregistré. |
| noPhoto | Le champ noPhoto est un integer auto-incrémenté permettant d'identifier une photo officielle de manière unique à l'intérieur de la base de données. Sa présence est obligatoire |
| taille | le champ taille est une chaine de caractères de type varchar(10) permettant d'identifier la taille d'un fichier (informatiquement parlant). Il se compose ainsi : 20Mo, 1Ko etc. |
| description Text | Le champ description permet de stocker tout ce qui est caractéristique du contrat: exigences, engagement. Il est de type  text et sa présence n'est pas obligatoire. |

|  |  |
| --- | --- |
| Statut | Cette classe permet de déterminer l'avancée des négociations pour un contrat. Elle est créée lors de la création d'un contrat. Elle est supprimée si le contrat avec lequel elle est associée est supprimé. Le statut peut changer à tout moment |
| noStatut | NoStatut est un entier (int) auto-incrémenté permettant d'identifier uniquement un statut dans la base de données. Sa présence est obligatoire. |
| nomStatut | Le champ nom est une chaine de caractères permettant de retrouver plus facilement un statut dans la base de données. Il est de type varchar(20) mais n'a pas de format précis. Sa présence est obligatoire. |
| description | Le champ description est une chaine de caractères de type Text permettant a l'utilisateur d'entrer toutes les informations voulues dans la base de données. Sa présence n'est pas obligatoire |

|  |  |
| --- | --- |
| CV | Un fichier personnel est une sorte de classeur contenant les données se rapportant à un artiste (comme les photos par exemple), il est possible de créer une occurrence de cette classe l'lorsque l'artiste associée est déjà présent à l'intérieur de la base de donnée. Il est cependant possible de la supprimer ou de la modifier à n'importe quel moment. |
| noCV | Le champ noCV est un integer auto-incrémenté permettant d'identifier de manière unique un fichier dans la base de données. Sa présence est obligatoire |
| commentaire | Le champ commentaire est une chaine de caractères de type Text permettant à l'utilisateur d'enregistrer dans la base de données toutes les informations qu'il désire. Sa présence n'est pas obligatoire et il n'a pas de format précis |
| description | La description d'un cv est une chaine de caractères permettant de décrire en détail une occurrence de la classe. Il est de type text et ne possède pas de format particulier, sa présence est obligatoire. |
| cheminFichier | Le champ chemin de fichier est une chaine de caractères de type Text permettant d'identifier la localisation le fichier de l'occurrence de la classe Cv. Sa présence est obligatoire. Il se compose ainsi : Document/dossier/fichier |

|  |  |
| --- | --- |
| ResponsableAgence | Un responsable d'agence est un intermédiaire quant à la signature d'un contrat entre un diffuseur et une agence. Une occurrence de la classe peut être crée lorsque l'agence qu'il représente a déjà été créée. Cette occurrence peut être modifiée ou supprimée à tout moment |
| noResponsableAgence | Le champ NoresponsableAgence est un entier auto-incrémenté (int) utilisé pour identifier le représentant de manière unique dans la base de données. Sa présence est obligatoire |
| nom | Le champ nom est une chaine de caractères de type varchar(20) utilisée pour retrouver plus facilement un responsable d'agence dans la base de données. Sa présence est obligatoire. |
| prenom | Le champ prénom est une chaine de caractères de type varchar(20) il sert aussi, comme le nom, à identifier plus facilement le responsable dans la base de données. Sa présence est obligatoire |
| courriel | Le champ courriel est une chaine de caractères de type varchar(30) permettant d'identifier l'adresse email de contact du responsable. Il se compose ainsi : machin@truck.quelquechose. Sa présence est obligatoire |
| signataire | Signataire est une chaine de caractères permettant de spécifier si le responsable agence est aussi le signataire du contrat. Il est de type varchar(20). Sa présence est obligatoire si le champ idem est égal à false. |
| dateSignature | dateSignature permet de stocker la date à laquelle le contrat sera signé ou approuvé. Elle peut être nulle si le contrat n'est pas approuvé. Elle est type date |
| telBureau | Le champ telBureau est une chaine de caractères permettant d'identifier un numéro de téléphone résidentiel de l'agence. Il est de format varchar(15) et se compose ainsi : 123-456-7890. Sa présence est obligatoire. |
| telCellulaire | Le champ telCellulaire est une chaine de caractères permettant d'identifier un numéro de téléphone mobile appartenant à l'agence (ou à un représentant), il est de type varchar(15) est ce compose ainsi : 123-456-7890. Sa présence est obligatoire. |
| Extension | L'extension est une chaine de caractères de type varchar(5) et se forme ainsi : 85 (par exemple). Sa  présence est obligatoire |
| Idem | Le chant idem est un booléen null a sa création. Il permet de déterminer si le responsable est aussi le signataire du contrat. |

|  |  |
| --- | --- |
| CategorieArtiste | Une catégorie d'artiste est une classe à laquelle appartient un artiste servant a déterminer son domaine avec précision. Tout artiste possède au moins une catégorie. Une occurrence de cette classe peut être créée, modifiée ou supprimée n'importe quand par l'utilisateur de l'application. |
| noCategorie | Le noCategorie est un nombre unique auto-incrémenté permettant d'identifier une catégorie d'artiste dans la base de données. Il est de type integer et n'a pas de format particulier. Sa présence est obligatoire. |
| nom | Le nom de catégorie est une chaine de caractères permettant de retrouver une catégorie d'artiste facilement à l'intérieur de la base de données. Il est de varchar(20) et ne possède pas de format particulier. Sa présence est obligatoire lors de la création d'une occurrence de la classe. |
| description | La description d'une catégorie d'artiste est une chaine de caractères permettant de décrire en détail une occurrence de la classe. Il est de type text et ne possède pas de format particulier, sa présence est obligatoire. |

|  |  |
| --- | --- |
| Engagement | Un engagement est une entente prise entre un diffuseur et une agence. (Par exemple sur l'heure d'un évènement).Une occurrence de la classe engagement peut être créée, modifiée ou supprimée en tout temps. |
| noEngagement | NoEngagement est un entier (int) auto-incrémenté permettant d'identifier de manière unique un engagement dans la base de données. Sa présence est obligatoire |
| description | Le champ description est une chaine de caractères permettant de stocker toutes informations ou détails concernant un engagement. Il  est de type Text et sa présence n'est pas obligatoire. |
| date | Le champ date est une entrée de type Date permettant d'identifier la date à laquelle un engagement devra être respecté. Il peut être null si aucune date n'a encore été convenue. |
| heure | Le champ heure est une entrée de type Time permettant de déterminer l'heure à laquelle un engagement devra être respecté. Il peut être null si aucune heure n'a été spécifié ou si l'engagement en question ne prend pas en charge cela. |
| commentaire | Le champ commentaire est une chaine de caractères de type Text permettant à l'utilisateur de stocker toutes informations supplémentaires dans la base de données. Sa présence n'est pas obligatoire. |
| nature | Le champ nature est une chaine de caractères de type varchar(50) permettant de définir le type exact de l'engagement. Exemple : Levée de fonds. Sa présence est obligatoire |
| durée | Le champ durée est une chaine de caractères de type varchar(6) permettant de déterminer le temps de validité d'un engagement. Il se compose ainsi : 3H40. Sa présence est obligatoire |
| lieu | Le champ lieu est une chaine de caractères de type Text permettant d'identifier l'endroit de validité d'un engagement. Il n'a pas de format précis et sa présence est obligatoire. |

|  |  |
| --- | --- |
| Contrat | Un contrat est la représentation d'une entente entre un diffuseur et une agence concernant un évènement. On peut créer une occurrence de la classe contrat s'il y a au moins une agence qui fait affaire avec un diffuseur. On peut le supprimer ou le modifier à tout moment. |
| noContrat | noContrat permet d'identifier uniquement un contrat et il est obligatoire. c'est un Varchar de longueur 50 et son format est un standard de l'entreprise: XX-999. |
| lieu | Ce champ définit l'environnement dans lequel l'évènement aura lieu. il peut être null si le contrat n'est pas encore signé. Par contre lorsque le contrat est signé ce champ devient obligatoire. Il est de type Text. |
| nom | Ce champ permet de trouver facilement le contrat dans la base de données. Il est de type varchar de longueur 50. Sa présence n'est pas obligatoire |
| description | La description d'un contrat est une chaine de caractères permettant de décrire en détail une occurrence de la classe. Il est de type text et ne possède pas de format particulier, sa présence est obligatoire. |
| commentaire | Le champ commentaire est une chaine de caractères de type Text permettant à l'utilisateur d'enregistrer dans la base de données toutes les informations qu'il désire. Sa présence n'est pas obligatoire et il n'a pas de format précis |

|  |  |
| --- | --- |
| Exigence | Une Exigence est une contrainte définie lors de la création d'un contrat. L'agence définie une exigence et le responsable du diffuseur décide s'il l'accepte ou non. Elle peut être supprimée ou modifiée sur demande de l'agence ou du diffuseur. |
| noExigence | Le champ noExigence est un entier (int) auto-incrémenté permettant d'identifier de manière unique une exigence dans la base de données. Sa présence est obligatoire. |
| nom | Le champ nom est une chaine de caractères de type varchar(20) permettant de retrouver plus facilement une exigence dans la base de données et de la rendre interprétable par un humain. Elle peut être par exemple : repas, transport etc. Sa présence est obligatoire. |
| description | Le champ description est une chaine de caractères de type Text permettant de stocker toutes informations ou détails au sujet de l'exigence dans la base de données. Sa présence n'est pas obligatoire. |
| statut | Le champ statut est une chaine de caractères de type varchar(50) permettant de déterminer l'état de l'exigence. Sa présence est obligatoire. |
| date | date permet de stocker la date à laquelle l'exigence a été approuvée. Elle est de type date et peut être nulle si l'exigence n'est pas encore approuvée. |
| montant | Le champ montant est un décimal permettant de déterminer le montant d'argent relié à une exigence. Il se compose ainsi : 2.34. Sa présence est obligatoire. |
| commentaire | Le champ commentaire permet de stocker toutes les informations voulues par l'utilisateur dans la base de données. Il est de type Text et sa présence n'est pas obligatoire. |

|  |  |
| --- | --- |
| Responsable | Un responsable est en quelques sortes le représentant humain du diffuseur. Il aura notamment la tache de signer les contrats. Une occurrence de la classe Responsable peut être créée, modifiée ou supprimée en tout temps. |
| noResponsable | Le champ noResponsable est un entier (int) auto-incrémenté permettant d'identifier de manière unique un responsable dans la base de données. Sa présence est obligatoire. |
| nom | Le champ nom est une chaine de caractères de type varchar(20) permettant d'identifier facilement un responsable dans la base de données. Sa présence est obligatoire. |
| prenom | Le champ prénom est une chaine de caractères de type varchar(20) permettant de compléter le nom lors d'une recherche dans la base de données. Sa présence est obligatoire. |
| courriel | Le champ courriel est une chaine de caractères de type varchar(30) permettant de stocker l'adresse e-mail d'un responsable. Il se compose ainsi : unemail@unfai.undomaine. Sa présence est obligatoire. |
| signataire | Le champ signataire est une chaine de caractères de type varchar(20) permettant d'identifier la personne qui signera le contrat. Il peut être null uniquement si le champ idem est égal a true. |
| dateSignature | dateSignature permet de stocker la date à laquelle le contrat sera signé ou approuvé. Elle peut être nulle si le contrat n'est pas approuvé. Elle est type date |
| telBureau | Le champ telBureau est une chaine de caractères permettant d'identifier un numéro de téléphone résidentiel du diffuseur. Il est de format varchar(15) et se compose ainsi : 123-456-7890. Sa présence est obligatoire. |
| telCellulaire | Le champ telCellulaire est une chaine de caractères permettant d'identifier un numéro de téléphone mobile appartenant au diffuseur (ou à un responsable), il est de type varchar(15) est ce compose ainsi : 123-456-7890. Sa présence est obligatoire |
| Extension | L'extension est une chaine de caractères de type varchar(5) et se forme ainsi : 85 (par exemple). Sa  présence est obligatoire |
| Idem | Le chant idem est un booléen null à sa création. Il permet de déterminer si le responsable est aussi le signataire du contrat. |

|  |  |
| --- | --- |
| Adresse | Une adresse permet d'identifier autrement que par un nom une agence avec qui l'utilisateur fait affaire. Pour créer une occurrence de cette classe et l'associé a une agence, l'agence en question doit déjà être enregistré dans la base de données. Il est possible de la modifier à n'importe quel moment. |
| noAdresse | NoAdresse est un champ permettant d'identifier uniquement une adresse à l'intérieur de la base de données. Il est de type integer(auto-incrémenté) est n'a pas de format particulier. Sa présence est obligatoire. |
| ville | Le champ ville est une chaine de caractères permettant d'identifier la commune ou se trouve l'agence. Il est de type varchar(20) et n'a pas de format particulier. Sa présence est obligatoire. |
| codePostal | Le code postal est une chaine de caractères de type varchar(6) permettant de compléter l'adresse courrier d'une agence. Sa présence est obligatoire. |
| Province | La province est un champ de type varchar(2) qui permet d'identifier par des initiale la partie du pays où se trouve l'agence. Sa présence est obligatoire. Exemple d'entrée : QC, ON,NB etc. |
| pays | LE pays est une chaine de caractères permettant d'identifier le pays où se trouve l'agence afin encore une fois de compléter l'adresse postale de celle-ci. Il est de type varchar(20) et n'a pas de format particulier. Sa présence est obligatoire. |
| telBureau | Le champ telMobile est une chaine de caractères permettant d'identifier un numéro de téléphone résidentiel de l'agence. Il est de format varchar(15) et se compose ainsi : 123-456-7890. Sa présence est obligatoire. |
| telCellulaire | Le champ telCellulaire est une chaine de caractères permettant d'identifier un numéro de téléphone mobile appartenant à l'agence (ou à un représentant), il est de type varchar(15) est ce compose ainsi : 123-456-7890. Sa présence est obligatoire. |
| Extension | L'extension est une chaine de caractères de type varchar(5) et se forme ainsi : 85 (par exemple). Sa  présence est obligatoire |

|  |  |
| --- | --- |
| Agence | Une agence est une entité avec laquelle l'utilisateur signe un contrat. Une occurrence de cette classe peut être créée, modifiée, supprimée n'importe quand. |
| noAgence | Le champ noAgence est une chaine de caractères permettant d'identifier uniquement une agence dans la base de données. Il est de type varchar(15) et se compose de la manière suivante : XXXXXX90. Sa présence est obligatoire. |
| nom | Le nom d'une agence est une chaine de caractères permettant de trouver facilement une agence dans la base de données. Il est de type varchar(50) et n'a pas de format particulier. Sa présence est obligatoire. |
| courriel | Un courriel est une chaine de caractères dédiée à identifier le premier moyen de contact d'une agence : L'email. Il est de type varchar(30) et se compose ainsi : agencemachin@truck.bidule. Sa présence n'est pas obligatoire. |

|  |  |
| --- | --- |
| Artiste | La classe artiste est une entité représentant les personnes retenue à la suite d'un contrat pour participer à un évènement. Il est possible de créer une occurrence de cette classe, de la modifier ou de la supprimer à n'importe quel moment. |
| noArtiste | le noArtiste est un nombre auto incrémenté mis en place afin d'identifier uniquement un artiste a l'intérieur de la base de donnée. Il est de type integer et n'a pas de format particulier. Sa présence est obligatoire. |
| nom | Le nom est une chaine de caractères permettant de retrouver simplement l'artiste a travers la base de données. Il est de type varchar(20) et ne possède pas de format particulier. Sa présence est elle aussi obligatoire. |
| Commentaire | Un commentaire est une chaine de caractères permettant à l'utilisateur d'entrer toutes les informations qui lui semblent nécessaires à l'intérieur de la base données. Il est de type text et n'a pas de format particulier. Sa présence n'est pas obligatoire. |

# Documentation des associations

|  |
| --- |
| Une agence est associée à un contrat puisqu'elle fournit les artistes relatifs à ce contrat. Une agence peut être reliée à plusieurs ou aucuns contrats. |
| Un artiste est associé à un contrat et en fait partie intégrante. Un artiste peut être relié à plusieurs ou aucun contrat et inversement. |
| Un contrat est associé à un duo Artiste/Photo lorsqu'il comporte une photo officielle pour un artiste compris à l'intérieur des négociations. Il peut être associé à plusieurs ou aucuns duos Artiste/Photo |
| Une photo peut être associée à un duo Artiste/Contrat lorsque la négociation du contrat avec un artiste le spécifie. Elle peut être associée à plusieurs ou aucuns duos Artiste/contrat |
| Un artiste est associé à un duo contrat/Photo lorsqu’une photo officielle est transmise lors de la négociation d'un contrat. Il peut être associé à plusieurs ou aucuns duos Contrat/Photo |
| Un responsable est associé à un diffuseur qu'il représente lors de la négociation d'un contrat. Un responsable peut être lié à plusieurs diffuseurs |
| Un diffuseur est associé à une adresse qui servira à connaitre les moyens de contacts de ce diffuseur. Plusieurs diffuseurs peuvent être associés à une adresse. |
| Un contrat est associé en tout temps à un statut qui représente l'avancement des négociations. Un contrat est associé à un unique statut alors qu'un statu peut être associé à plusieurs ou aucuns contrats. |
| Un artiste est associé à un cv lorsqu’il décide d'en transmettre un. U:n artiste peut être associé à plusieurs ou aucun Cv. |
| Un représentant est associé à une agence qu'il représente lors de la négociation d'un contrat. Un représentant peut être lié à plusieurs agences. |
| Un responsable est associé à un contrat qu'il négociera jusqu'à sa signature. Un responsable peut être associé à plusieurs ou aucuns contrats. |
| Un artiste est associé à une catégorie d'artiste en fonction de son type d'art. L'artiste peut être associé à une ou plusieurs catégories alors qu'une catégorie peut être associée à plusieurs ou aucuns artistes. |
| Un engagement est associé à un contrat lorsqu'il est compris dans celui-ci. Un engagement est associé à un unique contrat. En revanche un contrat peut être associé à plusieurs ou aucun engagements. |
| Une exigence est associée à un contrat lorsque celle-ci est spécifiée à l'intérieur de ce dernier. Une exigence est associée à un unique contrat alors qu'un contrat peut être associé à plusieurs ou aucunes exigences. |
| Un responsable est associé à un contrat qu'il négociera jusqu'à sa signature. Un responsable peut être associé à plusieurs ou aucuns contrats. |
| Une agence est associée à une adresse qui servira à connaitre les moyens de contacts de cette agence. Plusieurs agences peuvent être associées à une adresse. |
| Un artiste est associé avec une photo lorsque l'agence qui le représente décide d'en transmettre une. Un artiste peut être associé à aucunes ou plusieurs photos. Une photo en revanche est obligatoirement associée à un artiste. |

# Dictionnaire de données

Classe Agence :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noAgence | Varchar(15) |
| nom | Varchar(50) |
| Courriel | Varchar(30) |

Classe Artiste :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noArtiste | Integer |
| Nom | Varchar(20) |

Classe Adresse :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noAdresse | Integer |
| Ville | Varchar(20) |
| codePostal | Varchar(6) |
| Province | Varchar(2) |
| Pays | Varchar(20) |
| telBureau | Varchar(15) |
| TelCellulaire | Varchar(15) |
| Extension | varchar(5) |

Classe Categorie Artiste :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noCategorie | Integer |
| Nom | Varchar(20) |
| description | Text |

Classe responsable :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noResponsable | Integer |
| Nom | Varchar(20) |
| Prenom | Varchar(20) |
| courriel | Varchar(30) |
| signataire | Varchar(20) |
| dataSignature | Date |
| telBureau | varchar(15) |
| telCellulaire | varchar(15) |
| Extension | varchar(5) |
| Idem | Boolean |

Classe Contrat :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noContrat | varchar(50) |
| Lieu | Text |
| description | Text |
| Nom | Varchar(50) |
| Commentaire | Text |

Classe Engagement :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noEngagement | Integer |
| description | Text |
| Date | Date |
| Heure | time |
| commentaire | Text |
| nature | Varchar(50) |
| durée | Varchar(6) |
| lieu | Text |

Classe Exigence :

|  |  |
| --- | --- |
| Champ : | Type (longueur) |
| noExigence | Integer |
| nom | Varchar(20) |
| description | Text |
| statut | varchar(50) |
| date | Date |
| montant | Decimal |
| commentaire | Text |

Classe Statut:

|  |  |
| --- | --- |
| Champ : | Type (Longueur) |
| noStatut | Integer |
| nomStatut | varchar(20) |
| description | Text |

Classe Diffuseur :

|  |  |
| --- | --- |
| Champ | Type (Longueur) |
| noDiffuseur | Integer |
| nom | Varchar(50) |
| courriel | Varchar(30) |

Classe Photo:

|  |  |
| --- | --- |
| Champ | Type (Longueur) |
| noPhoto | Integer |
| taille | varchar(10) |
| commentaire | Text |

Classe CategorieArtiste:

|  |  |
| --- | --- |
| Champ | Type (Longueur) |
| noCategorie | Integer |
| nom | varchar(20) |

Classe CV:

|  |  |
| --- | --- |
| Champ | Type (Longueur) |
| noCV | Integer |
| commentaire | Text |
| description | Text |
| cheminFichier | Text |

# Acteurs et rôles

|  |  |  |
| --- | --- | --- |
| Acteur | Primaire/Secondaire | Rôle |
| Représentant | Primaire | Peut :  proposer un contrat  réserver des logements  réserver des repas  engager des techniciens  réserver une salle  fixer le prix des billets  faire publicité  annuler un évènement  Gérer personnel/étudiant  Créer des taches |
| Agent | Secondaire /Primaire | Peut :  Fournir renseignement concernant un évènement  Approuver une publicité  Présenter artiste  annuler un évènement |
| Personnel | secondaire | Peut :  Accomplir des taches |

# Acteurs et événements

|  |  |
| --- | --- |
| Acteur | Évènement |
| Diffuseur | Paye Agent  Reserve logement/repas/salle  Met en place publicité  Fixe/consulte le prix des billets et les statistiques  Annule événement  Créer tache |
| Agent | Présente un artiste (avec renseignement)  Approuve publicité  Annule événement |
| Personnel | Accompli une tache |

# Description narrative

|  |  |  |
| --- | --- | --- |
|  | | |
| ***Titre du cas d’utilisation :*** Consulter le sommaire des contrats | | ***Numéro:*** |
|  | | |
| ***Description: Permet de consulter tous les contrats disponibles*** | | |
| ***Acteur primaire: Gestionnaire*** | | |
| ***Acteur secondaire : Réceptionniste*** | | |
| ***Date de création : 05/04/2016*** | ***Date de modification :*** | |
| ***Version :1.0*** | ***Responsable : Kevin Bradford*** | |
|  | | |
| ***PréConditions :*** Être authentifié avec les droits de lecture des contrats. | | |
| ***Déclencheur :*** L’utilisateur désire consulter les contrats dans le système à partir du menu | | |
| ***Processus nominal :***     1. Le système affiche la fenêtre du sommaire des contrats 2. La réceptionniste consulte le sommaire 3. La réceptionniste veut fermer la fenêtre 4. Le système ferme la fenêtre | | |
| ***Arrêt :***  Le cas se termine lorsque la fenêtre du sommaire des contrats est fermée | | |
| ***PostConditions :***  *Aucune* | | |
|  | | |
| ***Extensions:***    E1 : Ajouter un contrat  Début: 1  1. L’utilisateur veut ajouter un contrat en appuyant sur le bouton ajouter  Appel -> Description narrative: Ajouter un contrat  Retour: 1    E2 : Voir les détails  Début: 1  1. La réceptionniste veut voir les détails d’un contrat en appuyant sur le bouton détails  Appel -> Description narrative: Ajouter un contrat -> A1  Retour: 1    E3 : Modifier un contrat  Début: 1  1. La réceptionniste veut modifier un contrat en appuyant sur le bouton détails  Appel -> Description narrative: Ajouter un contrat -> A3  Retour: 1    E4 : Supprimer un contrat  Début: 1  1. La réceptionniste veut supprimer un contrat en appuyant sur le bouton détails  Appel -> Description narrative: Ajouter un contrat -> A2  Retour: 1 | | |
| ***Scénarios Alternatifs:***    **A1 : Rechercher un contrat.**  Début:1  1. La réceptionniste entre une chaîne de caractères correspondant aux termes qu’elle souhaite rechercher dans l’ensemble des champs du contrat  2. La réceptionniste envoie la requête au système  3. Le système affiche les résultats de la requête au même endroit que la liste originale  Retour: 1 | | |
|  | | |
| ***Contraintes non-fonctionnelles:***  *Aucune* | | |
| ***Contraintes liées à l'IHM:***  Tri par nom, date, etc. Tri par défaut: date d’évènement plus récente à plus ancienne (croissant)  Champ recherche dans les détails (fenêtre).  Les boutons pour ajouter, modifier et supprimer sont désactivés si l’utilisateur n’a pas le droit d’ajouter, de modifier et de supprimer.  La fenêtre devra être pleine largeur et la hauteur ajustée au contenu | | |
| ***Divers:***  Filtre à voir dépendamment du temps, budget, énergie, etc.  Réviser la colonne liée à l’engagement. (après les tests vérifier avec le client) | | |

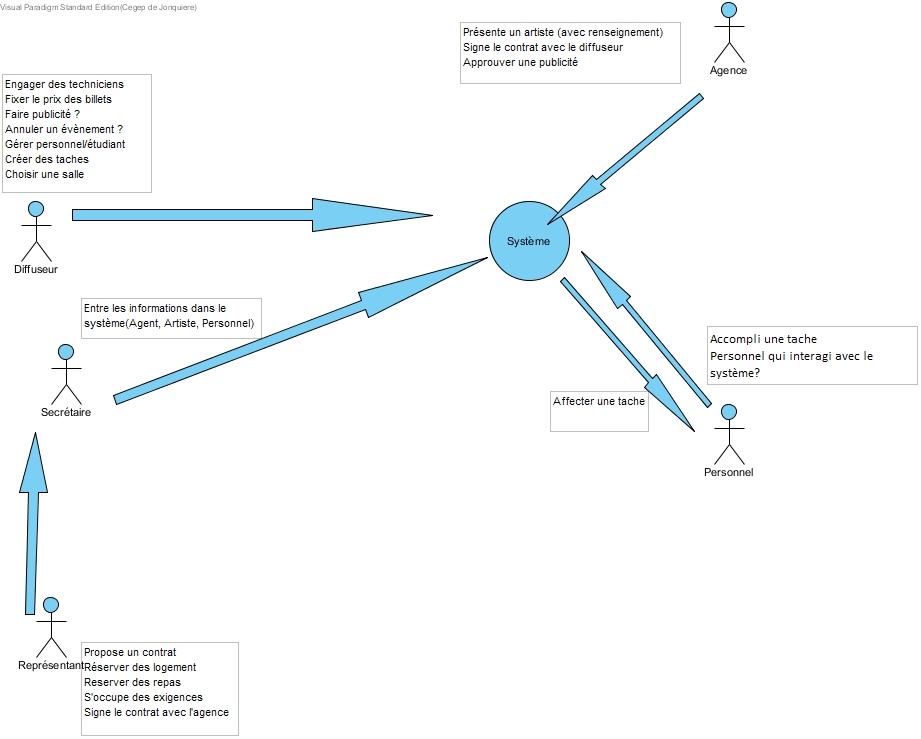
|  |  |  |
| --- | --- | --- |
|  | | |
| ***Titre du cas d’utilisation :*** Ajouter un contrat | | ***Numéro:*** |
|  | | |
| ***Description: Permet de créer un contrat avec une agence*** | | |
| ***Acteur primaire: Gestionnaire*** | | |
| ***Acteur secondaire : Réceptionniste*** | | |
| ***Date de création : 05/04/2016*** | ***Date de modification :*** | |
| ***Version :1.0*** | ***Responsable : Kevin Bradford*** | |
|  | | |
| ***PréConditions :*** Être authentifié avec le droit de création pour contrat | | |
| ***Déclencheur :*** L’utilisateur désire ajouter un contrat à partir de l’écran «Sommaire des contrats» | | |
| ***Processus nominal :***     1. Le système affiche la fenêtre de détails du contrat.   2. La réceptionniste ajoute les informations du nouveau contrat.  3. La réceptionniste enregistre le contrat.  4. Le système affiche un message de confirmation.  5. La réceptionniste confirme l’enregistrement.  6. Le système enregistre les données.  7. Le système ferme la fenêtre. | | |
| ***Arrêt :***   * *Le cas se termine quand toutes les informations ont été ajoutées.* * *Aucune erreur de format ni d’entrées vides n’est présente.* * *Le cas peut être arrêté à tout moment à la suite de l’annulation.* | | |
| ***PostConditions :***  Le contrat et toutes les informations sont enregistrés dans la base de données.  Le nouveau contrat est affiché dans la fenêtre sommaire et est automatiquement sélectionné.  Si le cas se termine par une annulation on devra être positionné sur le premier élément de la fenêtre sommaire « Gérer les contrats ». | | |
|  | | |
| ***Exceptions/Extensions:***    **E1 : Une donnée ne correspond pas au format demandé par un champ**  Début: 3  1. Le système affiche le modèle d’erreur de format.  2. La réceptionniste s’informe de l’erreur.  3. La réceptionniste ferme le message.  4. Le système retourne le curseur sur le champ erroné.  Retour: 1    **E2 : Une donnée d’un champ obligatoire est vide**  Début: 3  1. Le système affiche le modèle d’erreur null.  2. La réceptionniste s’informe de l’erreur.  3. La réceptionniste ferme le message.  4. Le système retourne le curseur sur le champ vide.  Retour: 1  **E3: Ajouter un statut de contrat**  Début: 2  1. L’utilisateur veut ajouter un statut de contrat  Appel -> Description narrative: Ajouter un statut | | |
| ***Scénarios Alternatifs:***    **A1 : Consulter un contrat**  **Début:1**   1. **Le système affiche la fenêtre avec les données du contrat sélectionné sur l’écran sommaire « Sommaire des contrats »** 2. **La réceptionniste peut consulter le contrat** 3. **La réceptionniste choisit d’annuler**   **Retour: 7**  **A2 : Supprimer un contrat**  **Début:1**   1. **Le système affiche la fenêtre avec les données du contrat sélectionné sur l’écran sommaire « Sommaire des contrats »** 2. **La réceptionniste peut supprimer le contrat** 3. **La réceptionniste choisit de supprimer le contrat** 4. **Le système affiche un message de confirmation de suppression.** 5. **La secrétaire confirme la suppression.** 6. **Le système supprime le contrat.**   **Retour: 7**  **A3 : Modifier un contrat**  **Début:1**   1. **Le système affiche la fenêtre avec les données du contrat sélectionné sur l’écran sommaire « Sommaire des contrats »** 2. **La réceptionniste peut modifier toutes les informations du contrat.** 3. **La réceptionniste choisit d’enregistrer.** 4. **Le système affiche le message de confirmation de modification.** 5. **La secrétaire confirme la modification.** 6. **Le système met à jour les données.**   **Retour: 7** | | |
|  | | |
| ***Contraintes non-fonctionnelles:***  *Peu d'utilisateurs auront le droit d'ajouter des contrats.* | | |
| ***Contraintes liées à l'IHM***  Commentaire: Gras, italique, souligné, couleur.  Validation du signataire. (aucun par défaut)  Code de couleur pour le statut du contrat.  Hypertexte du site web de l’agence.  Lors d'un ajout, le bouton supprimer devra être désactivé.  La fenêtre devra être positionnée aux mêmes coordonnées (x,y) que toutes les fenêtres de détail. Elle devra être de demi-largeur d'écran et la hauteur ajustée au contenu.  Vérifier si on doit sélectionner le contrat lorsqu'on retourne à l'écran sommaire ou on doit seulement s'y positionner. | | |
| ***Divers:***  Le lieu dans un contrat peut changer  Réviser la colonne liée à l’engagement. (après les tests vérifier avec le client)  *Vérifier l’authentification, qui a le droit de modifier, créer et supprimer.* | | |

|  |  |  |
| --- | --- | --- |
|  | | |
| ***Titre du cas d’utilisation :*** Ajouter un statut | | ***Numéro:*** |
|  | | |
| ***Description: Permet de créer un statut*** | | |
| ***Acteur primaire: Gestionnaire*** | | |
| ***Acteur secondaire : Réceptionniste*** | | |
| ***Date de création :*** | ***Date de modification :*** | |
| ***Version :*** | ***Responsable : Kevin Bradford*** | |
|  | | |
| ***PréConditions :***  *Être authentifié avec le droit de création pour statut d’un contrat* | | |
| ***Déclencheur :*** L’utilisateur désire ajouter un statut en passant par la fenêtre détails de contrat | | |
| ***Processus nominal :***   1. Le système affiche la fenêtre d’ajout de statut 2. La réceptionniste ajoute le titre du statut et son code de couleur 3. La réceptionniste enregistre le statut 4. Le système affiche un message de confirmation 5. La réceptionniste confirme l’enregistrement 6. Le système enregistre les données 7. Le système ferme la fenêtre | | |
| ***Arrêt :***   * *Le cas se termine quand toutes les informations ont été ajoutées.* * *Le cas peut être arrêté à tout moment à la suite de l’annulation.* | | |
| ***PostConditions :***  *Le statut est ajouté dans la base de données*  *Le statut est ajouté et est automatiquement sélectionné dans la liste déroulante*  *Si le cas se termine par une annulation on devra être positionné sur le premier élément de la liste déroulante* | | |
|  | | |
| ***Exceptions/Extensions:***  Aucune | | |
| ***Scénarios Alternatifs:***  **Aucun** | | |
|  | | |
| ***Contraintes non-fonctionnelles:***   * *A déterminer avec le client si il y aura une possibilité de modifier et supprimer un statut, et si oui, comment?* * *Peu d’utilisateurs auront le droit d’ajouter des statuts* | | |
| ***Contraintes liées à l'IHM:***   * *Aucune* | | |
| ***Divers:***   * *Vérifier l’authentification, qui a le droit de créer* | | |

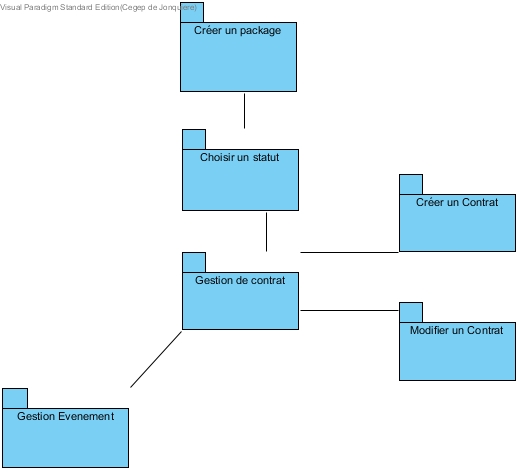
|  |  |  |
| --- | --- | --- |
|  | | |
| ***Titre du cas d’utilisation :*** Gérer les catégories d’artiste | | ***Numéro:*** |
|  | | |
| ***Description : Permet de gérer toutes les catégories d’artiste disponible : Consulter les détails et ajouter une catégorie d’artiste*** | | |
| ***Acteur primaire : Gestionnaire*** | | |
| ***Acteur secondaire : Réceptionniste*** | | |
| ***Date de création : 05/04/2016*** | ***Date de modification :*** | |
| ***Version : 1.0*** | ***Responsable : Kevin Bradford*** | |
|  | | |
| ***PréConditions :***   * *Être authentifié avec le droit de modification, de lecture et d’écriture sur les catégories d’artistes* | | |
| ***Déclencheur :*** *L'utilisateur veut gérer les catégories d’artistes :*   * *Consulter les détails d’une catégorie d’artiste* * *Ajouter une catégorie d’artiste* | | |
| ***Processus nominal :***   1. Le système affiche la fenêtre de sommaire des catégories d’artistes en chargeant toutes catégories d’artiste existantes 2. La réceptionniste visualise les catégories existantes 3. La réceptionniste ferme la fenêtre 4. Le système ferme la fenêtre | | |
| ***Arrêt :***   * *Le cas se termine lorsque la fenêtre sommaire est fermée* | | |
| ***PostConditions :***   * *Aucune* | | |
|  | | |
| ***Extensions :***  E1 : Ajouter une catégorie d’artiste  Début: 1  1. L’utilisateur veut ajouter une catégorie d’artiste en appuyant sur le bouton ajouter  Appel -> Description narrative: Ajouter une catégorie d’artiste  Retour: 1  E2 : Voir les détails  Début: 1  1. La réceptionniste veut voir les détails d’une catégorie d’artiste en appuyant sur le bouton détails  Appel -> Description narrative: Ajouter une catégorie d’artiste -> A1  Retour: 1  E3 : Modifier une catégorie d’artiste  Début: 1  1. La réceptionniste veut modifier une catégorie d’artiste en appuyant sur le bouton détails  Appel -> Description narrative: Ajouter une catégorie d’artiste -> A3  Retour: 1    E4 : Supprimer une catégorie d’artiste  Début: 1  1. La réceptionniste veut supprimer une catégorie d’artiste en appuyant sur le bouton détails  Appel -> Description narrative: Ajouter un contrat -> A2  Retour: 1 | | |
| ***Scénarios Alternatifs :***  **1** : **Rechercher.**  Début :1   1. La réceptionniste saisit une chaîne de caractères 2. La réceptionniste lance la recherche 3. Le système fait une recherche syntaxique 4. Le système souligne la ou les catégories d’artiste correspondant à la recherche dans la grille ou bien il se positionne sur la première occurrence correspondant à la recherche   Retour : 1 | | |
|  | | |
| ***Contraintes non-fonctionnelles :***   * *Les catégories seront créées en bloc au début puis peu modifiées par la suite.* * *Peu d'utilisateurs auront le droit de gérer les catégories d’artiste.* * *En conséquence, peu de besoins d'aide et de documentation et ne pas complexifier cette fenêtre inutilement.* | | |
| ***Contraintes liées à l’IHM :***   * *La fenêtre devra être pleine largeur et la hauteur ajustée au contenu* * *Les boutons pour ajouter, modifier et supprimer sont désactivés si l’utilisateur n’a pas le droit d’ajouter, de modifier et de supprimer.* * *Tri par nom des catégories d’artistes par défaut, sinon possibilité de modifier le tri en cliquant sur la colonne choisi.* | | |
| ***Divers :***   * *Vérifier l’authentification, qui a le droit de modification, de lecture et d’écriture sur les catégories d’artiste* * *Vérifier l’état du bouton Ajouter (désactiver ou activer) par rapport à l’authentification de l’utilisateur* * *Vérifier la différence entre Rechercher et Filtre. Est-ce que la recherche souligne ou réactualise la grille par rapport aux catégories d’artiste recherchées* * *En ce qui concerne le filtre il sera plus analyser l’automne prochain* * *Filtre à voir dépendamment du temps, budget, énergie, etc.* | | |

|  |  |  |
| --- | --- | --- |
|  | | |
| ***Titre du cas d’utilisation :*** Ajouter une catégorie d’artiste | | ***Numéro:*** |
|  | | |
| ***Description : Permet de créer une catégorie d’artiste*** | | |
| ***Acteur primaire: Gestionnaire*** | | |
| ***Acteur secondaire : Réceptionniste*** | | |
| ***Date de création : 05/042016*** | ***Date de modification :*** | |
| ***Version : 1.0*** | ***Responsable : Kevin Bradford*** | |
|  | | |
| ***Préconditions :***   * *Être authentifié avec le droit de création pour catégorie d’artiste* | | |
| ***Déclencheur :*** *L'utilisateur désire ajouter une catégorie à partir de "Gérer les catégories" ou à partir de la fenêtre de détail "Artiste"* | | |
| ***Processus nominal :***   1. Le système affiche la fenêtre de détail de catégorie d’artiste. 2. La réceptionniste ajoute le nom et la description de la catégorie d’artiste 3. La réceptionniste enregistre la catégorie d’artiste 4. Le système affiche un message de confirmation d'enregistrement 5. La réceptionniste confirme l’enregistrement 6. Le système enregistre les données 7. Le système ferme la fenêtre | | |
| ***Arrêt :***   * *Le cas se termine quand toutes les informations ont été ajoutées.* * *Le cas peut être arrêté à tout moment à la suite de l’annulation* | | |
| ***PostConditions :***   * *La catégorie est ajoutée dans la base de données* * *La nouvelle catégorie est affichée dans la fenêtre sommaire et est automatiquement sélectionnée* * *Si le cas se termine par une annulation, on devra être positionné sur le premier élément de la fenêtre sommaire "Gérer les catégories d'artiste"* | | |
|  | | |
| ***Exceptions/Extensions:***  **Aucune** | | |
| ***Scénarios Alternatifs:***  **A1** : **Consulter une catégorie d’artiste.**  Début:1   1. Le système affiche la fenêtre avec les données de la catégorie sélectionnée sur l'écran sommaire *"Gérer les catégories d'artiste"*. 2. La réceptionniste peut consulter la catégorie d’artiste. 3. La réceptionniste choisit d’annuler.   Retour : 7  **A2** : **Supprimer une catégorie d'artiste.**  Début:1   1. Le système affiche la fenêtre avec les données de la catégorie sélectionnée sur l'écran sommaire *"Gérer les catégories d'artiste"*. 2. La réceptionniste choisit de supprimer la catégorie. 3. Le système affiche un message de confirmation de suppression. 4. La secrétaire confirme la suppression. 5. Le système supprime la catégorie.   Retour : 7  **A3** : **Modifier une catégorie d’artiste.**  Début:1   1. Le système affiche la fenêtre avec les données de la catégorie sélectionnée sur l'écran sommaire *"Gérer les catégories d'artiste"*. 2. La réceptionniste peut modifier le nom ou la description. 3. La réceptionniste choisit d’enregistrer. 4. Le système affiche le message de confirmation de modification. 5. La secrétaire confirme la modification. 6. Le système met à jour les données.   Retour : 7 | | |
|  | | |
| ***Contraintes non-fonctionnelles:***   * *Les catégories seront créées en bloc au début puis peu modifiées par la suite.* * *Peu d'utilisateurs auront le droit d'ajouter des catégories.* * *En conséquence, peu de besoins d'aide et de documentation et ne pas complexifier cette fenêtre inutilement.* | | |
| ***Contraintes liées à l'IHM:***   * *Lors d'un ajout, le bouton supprimer devra être désactivé.* * *La fenêtre devra être positionnée aux mêmes coordonnées (x,y) que toutes les fenêtres de détail. Elle devra être de demi-largeur d'écran et la hauteur ajustée au contenu.* * *Vérifier où positionner la sélection après annulation, modification, ou suppression de catégorie.* * *Vérifier si on doit sélectionner la catégorie lorsqu'on retourne à l'écran sommaire ou on doit seulement s'y positionner.* | | |
| ***Divers:***   * *Vérifier l’authentification, qui a le droit de modifier, créer et supprimer.* * *Vérifier ce qu'on souhaite faire lorsque des artistes sont liés à une catégorie qu'on souhaite supprimer* | | |

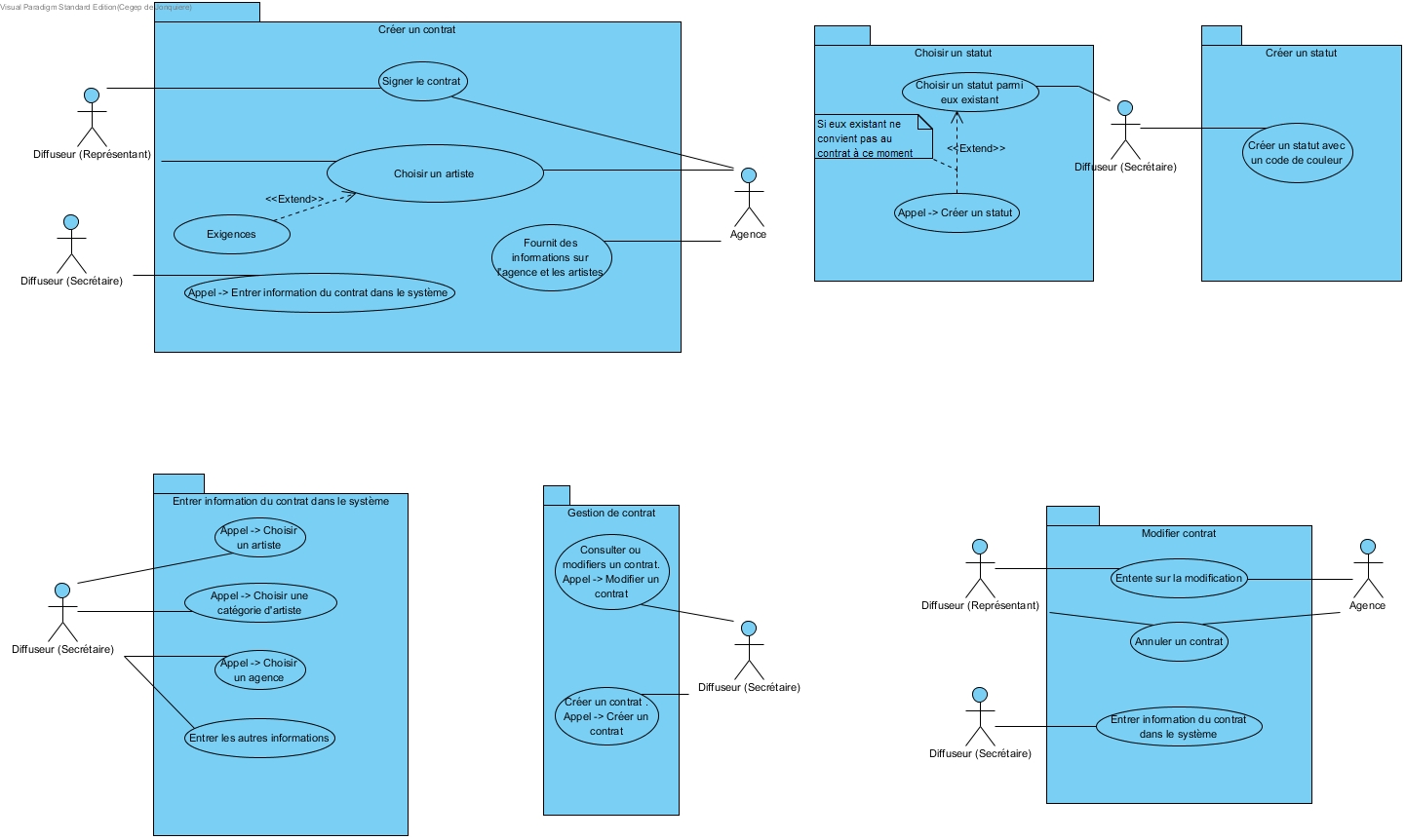
# Diagramme de contexte



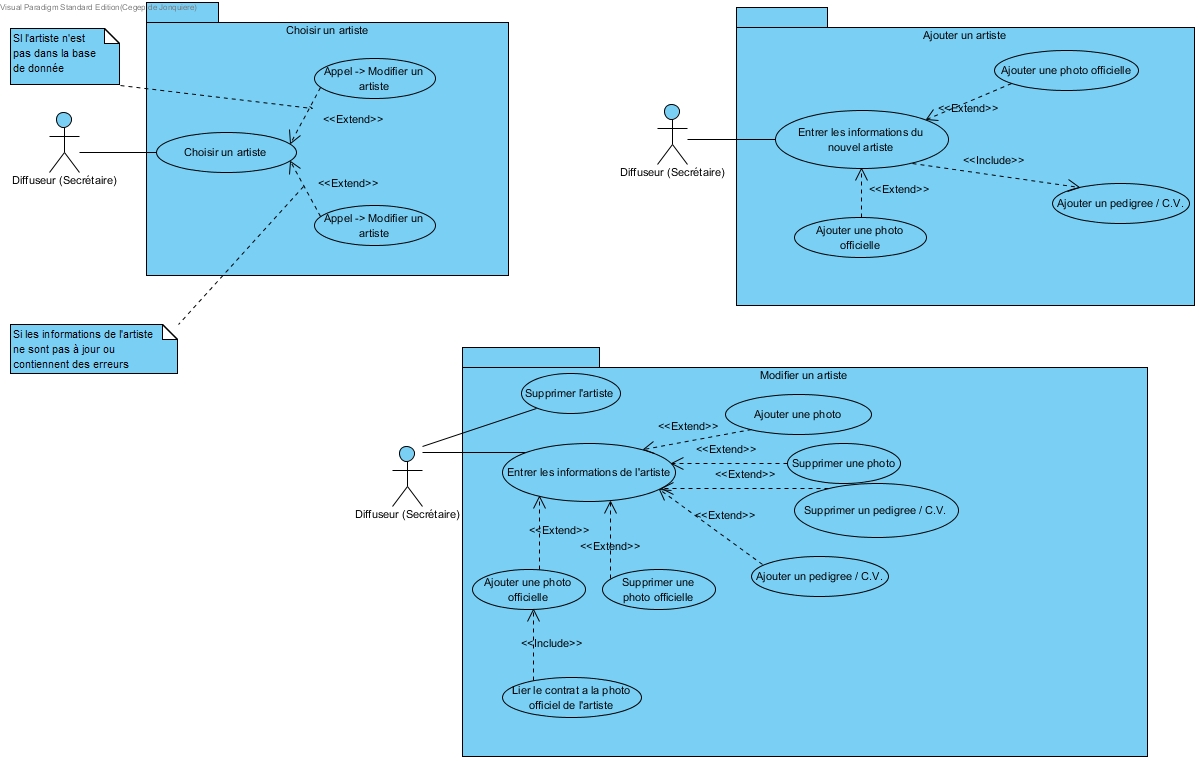
# Diagramme de package



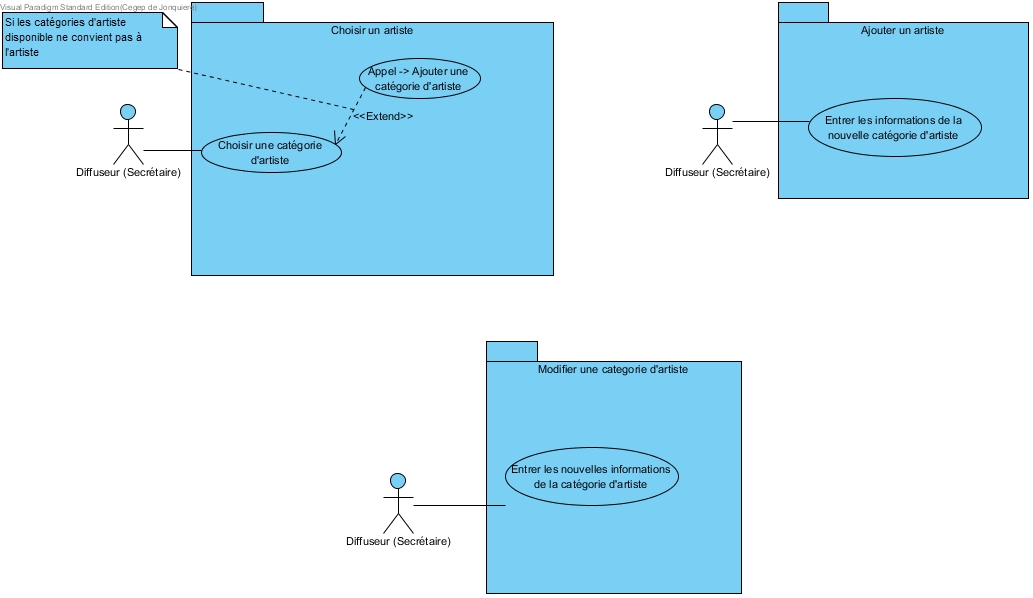
# Diagramme de cas d’utilisation Gestion de contrat



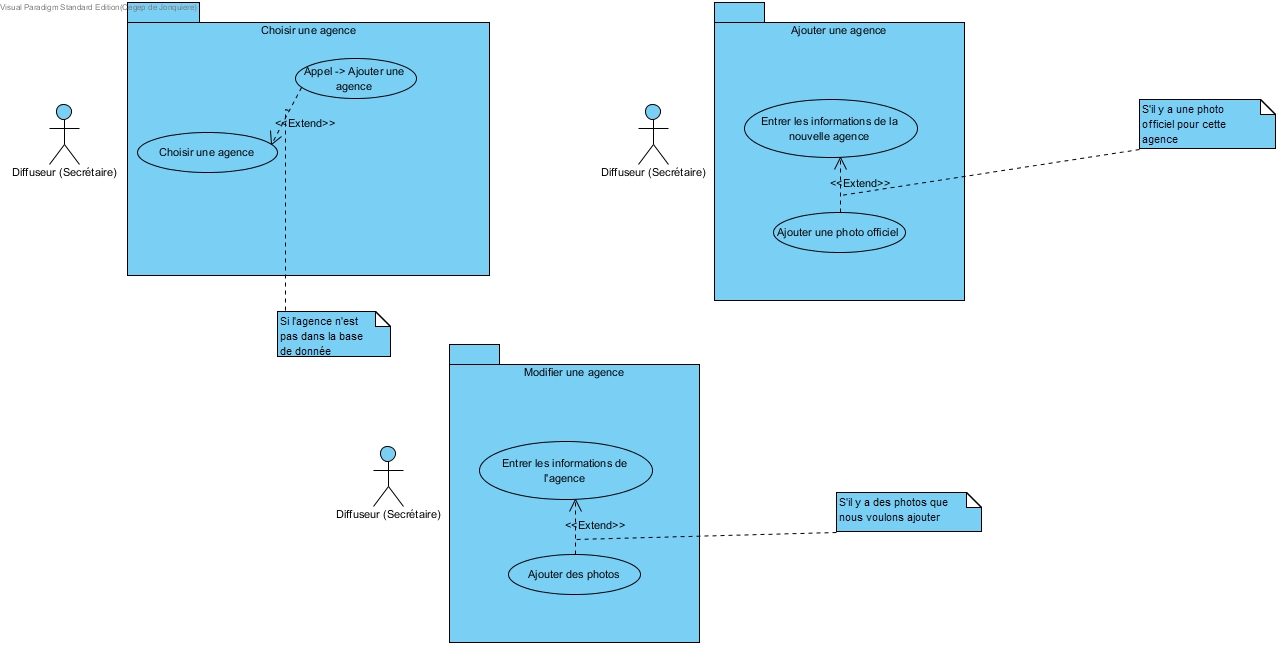
## Gestion d’artiste



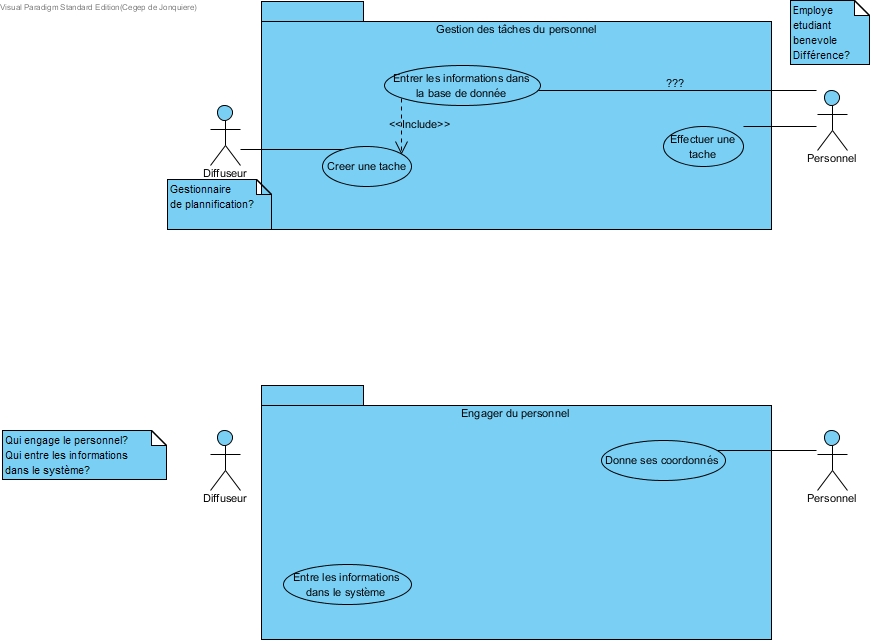
## Gestion des catégories d’artiste



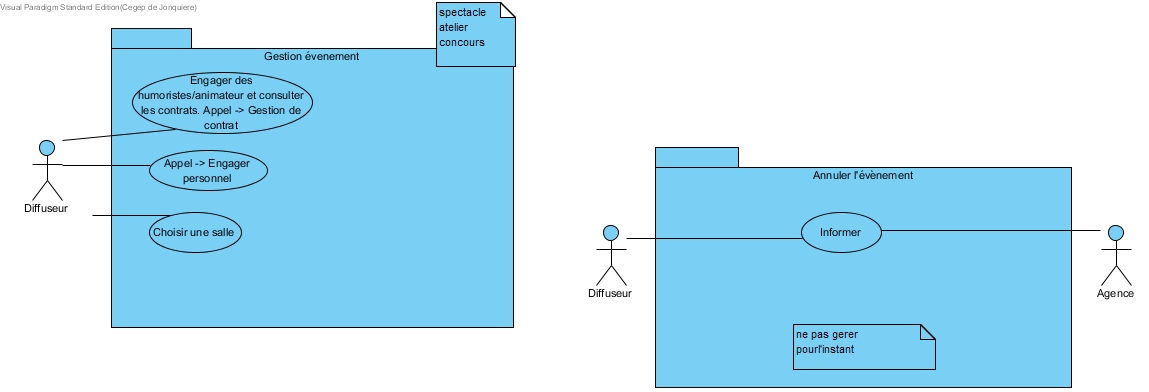
## Gestion d’agence



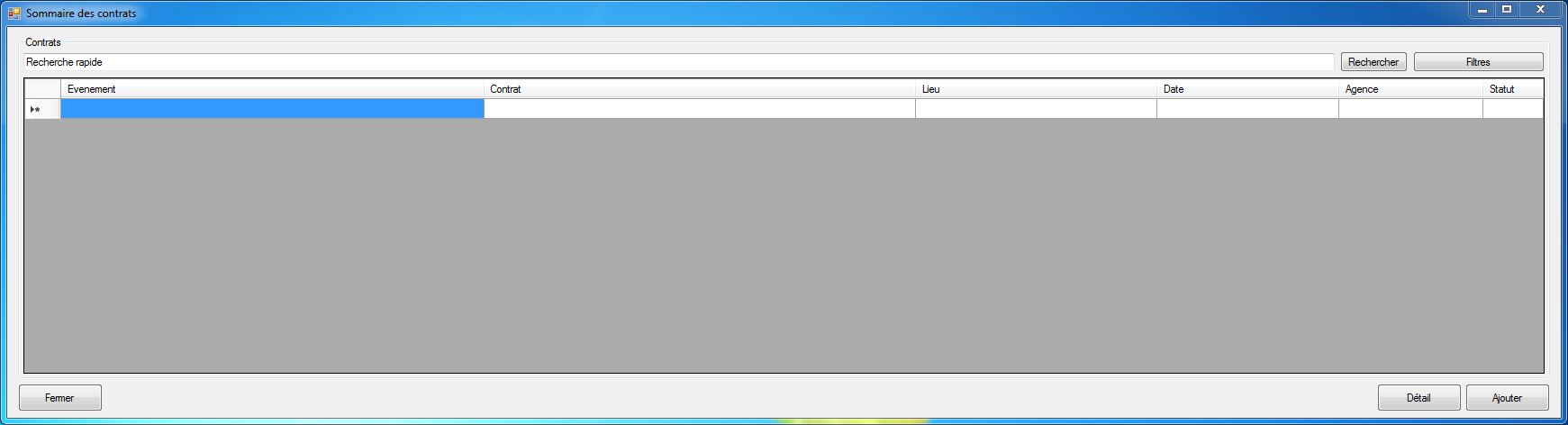
## Gestion de personnel



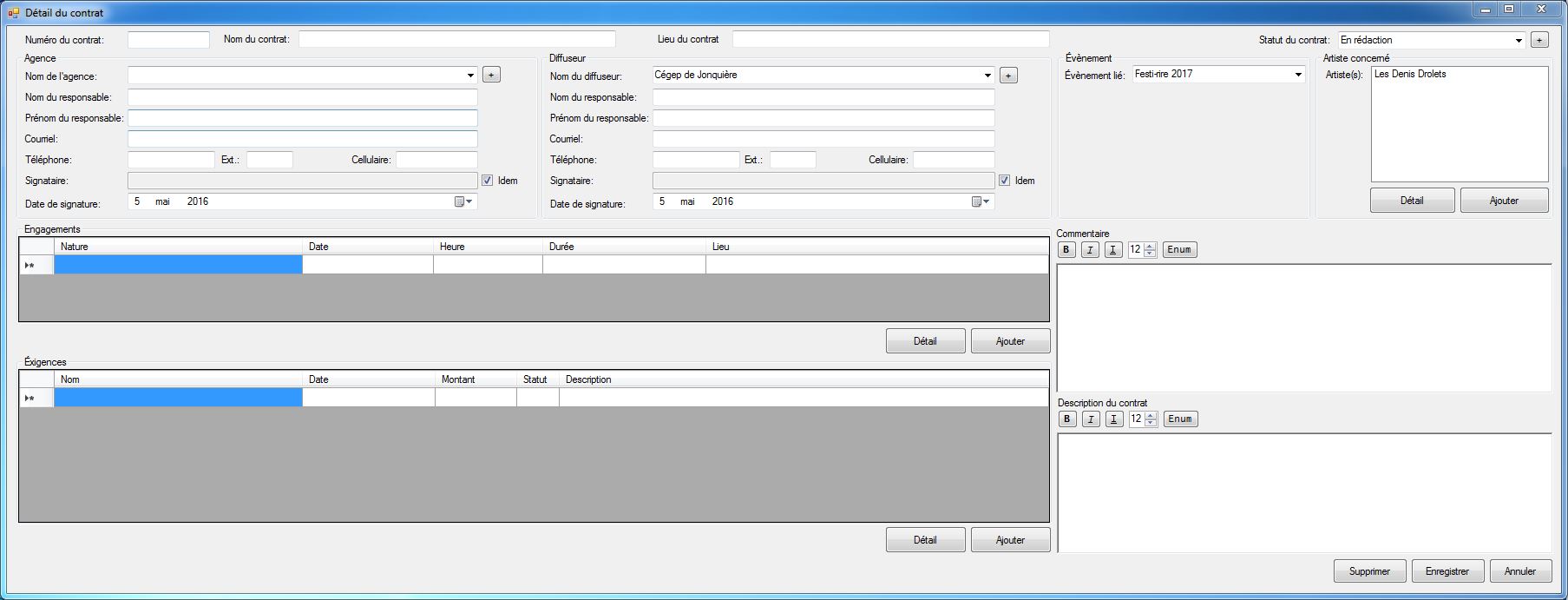
## Gestion d’évènement



# Prototype d’IHM Sommaire des contrats



## Détail d’un contrat



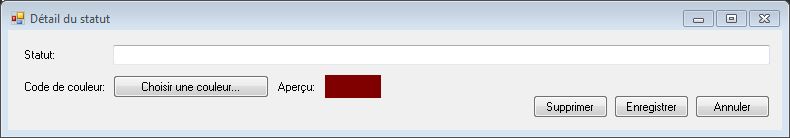
### Confirmation d’enregistrement Confirmation de modification Confirmation de suppression



### Erreur de format Erreur de champ vide



## Détails d’un statut de contrat



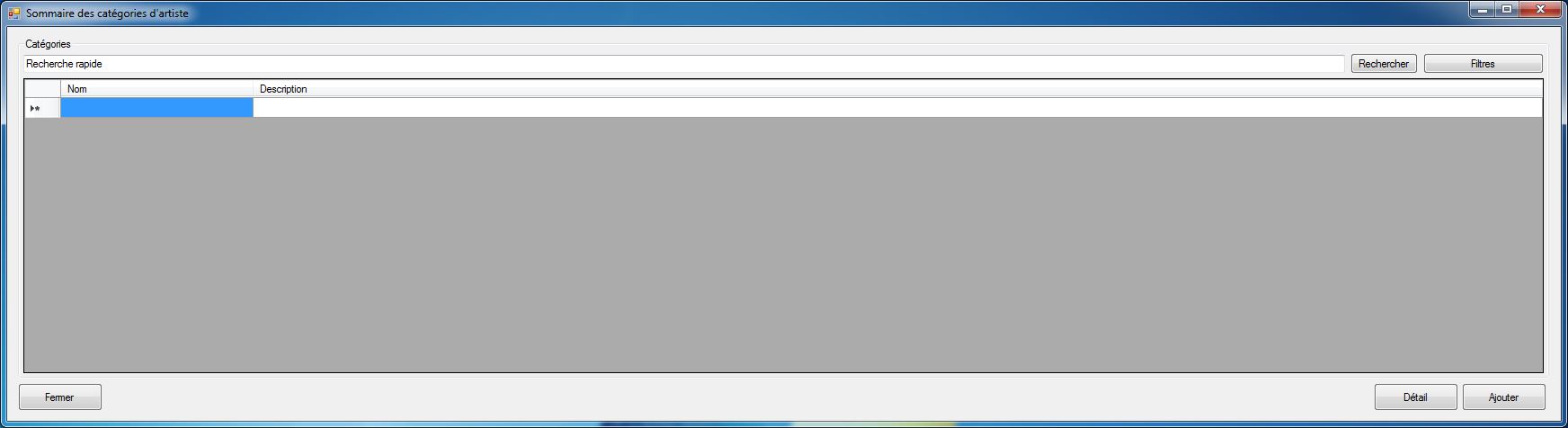
### Confirmation d’enregistrement Confirmation de modification Confirmation de suppression



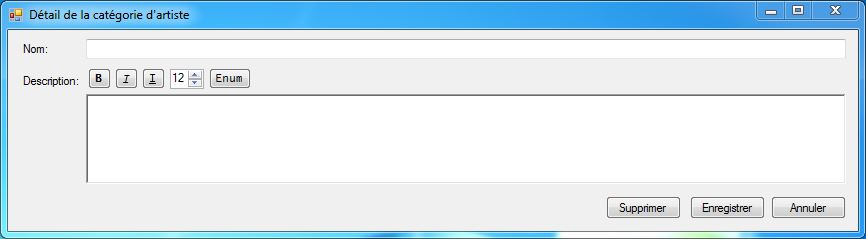
### Erreur de champ vide



## Sommaire des catégories d’artiste



## Détails d’une catégorie d’artiste



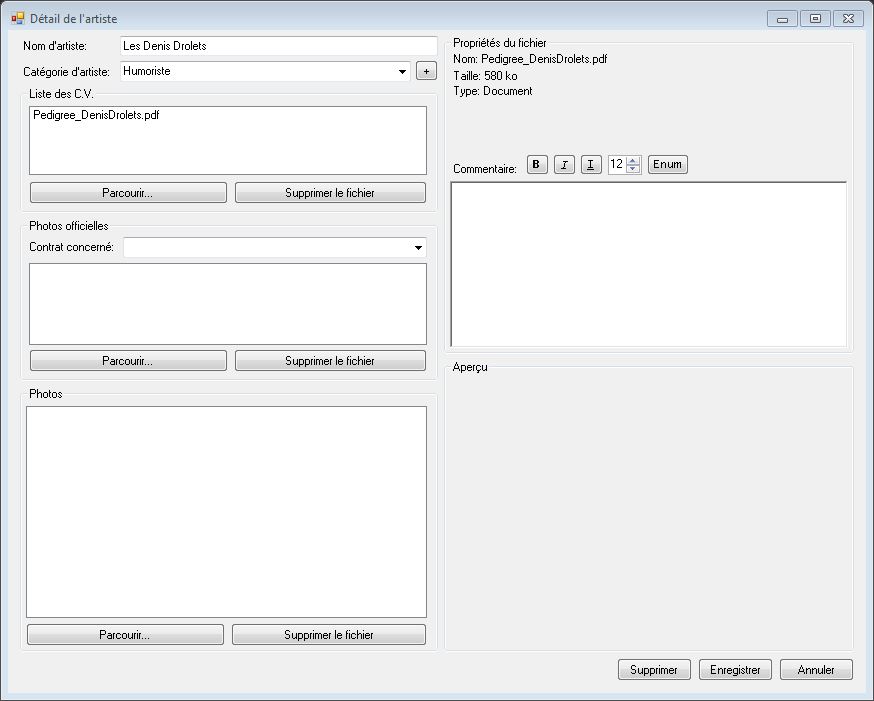
### Confirmation d’enregistrement Confirmation de modification Confirmation de suppression



### Erreur de champ vide



## Détails d’un artiste



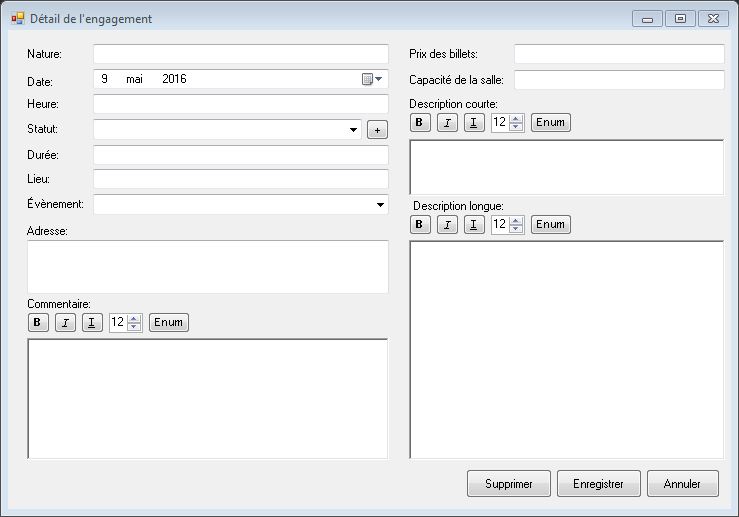
### Confirmation d’enregistrement Confirmation de modification Confirmation de suppression



### Erreur de champ vide



## Détails des engagements



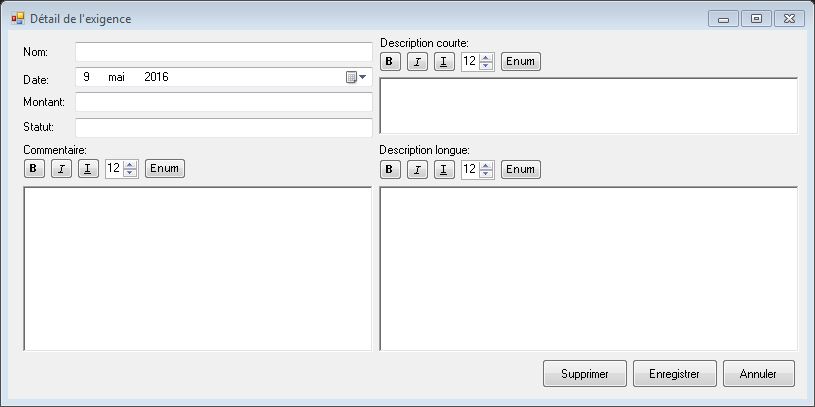
### Confirmation d’enregistrement Confirmation de modification Confirmation de suppression



### Erreur de format Erreur de champ vide



## Détails des exigences



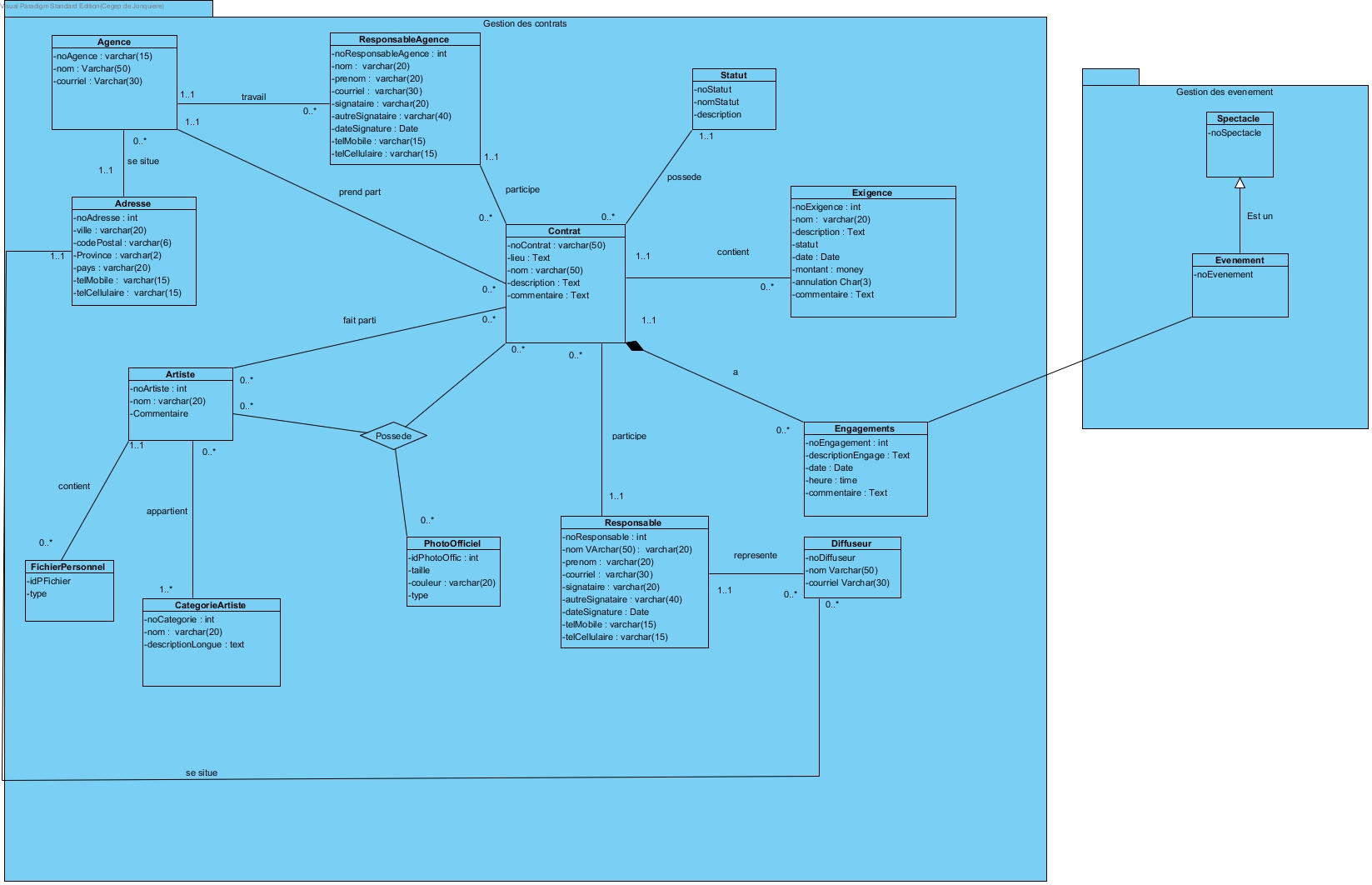
### Confirmation d’enregistrement Confirmation de modification Confirmation de suppression



### Erreur de format Erreur de champ vide



# Diagramme de classe



1. Chaque partie de gestion comportera des interfaces, des diagrammes de cas d’utilisation ainsi que des descriptions narratives. [↑](#footnote-ref-1)