**Actualité**

*(micro-service)*

Documentation : Conception détaillée

Révision R0003

URBANISATIO SI – MODELISATION – DEVELOPPEMENT – WORKFLOW - DECISIONNEL

Gestion du document

|  | **Nom** | **Société / service** | **Date** | **Signature** |
| --- | --- | --- | --- | --- |
| **Rédigé par** | Alpha LAFOND | axYus | 10/12/2018 |  |
| **Vérifié par** |  |  |  |  |
| **Approuvé par** |  |  |  |  |
|  |  |  |  |  |

Historique des modifications

| **Révision** | **Date** | **Modification** | **Auteur** |
| --- | --- | --- | --- |
| R001 | 10/12/2018 | Création du document. | Alpha LAFOND |
| R002 | 14/12/2018 | Correction | Alpha LAFOND |
| R003 | 20/02/2018 | Correction | Edem ANANI-TOULASSI |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Liste de diffusion

| **Nom** | **Société** | **Service** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

Sommaire

1. Objet du document 6

2. Contexte 7

2.1. Principe des actualités 8

2.2. Cas d’utilisation : Actualité 10

2.2.1. Diagramme 10

2.2.2. Description détaillée 11

3. Modèle de données 14

3.1. Modèle conceptuel 14

3.2. Dictionnaire de données 15

3.2.1. Habilitation 15

3.2.2. Actualité 15

3.2.3. Filtre d’affichage 16

3.2.4. Les sections 17

3.2.5. Les ressources *(sous forme de lien)* 18

Image illustrative 19

3.3. Modèle physique 20

3.3.1. Autres objets du modèle 21

4. Représentation des données 22

4.1. Les entités 22

5. Services métiers 24

5.1. S001 : Création d’une actualité 24

5.1.1. Description des transactions métiers 24

5.2. S002 : Consultation d’une actualité 28

5.2.1. Description des transactions métiers 28

5.3. S003 : Modification d’une actualité 32

5.3.1. Description des transactions métiers 32

5.4. S004 : Suppression d’une actualité 33

5.4.1. Description des transactions métiers 33

5.5. S005 : Recherche d’actualités 35

5.5.1. Description des transactions métiers 35

5.6. S006 : Vérification de l’existence d’un élément d’une actualité. 36

5.6.1. Description des transactions métiers 36

5.7. S007 : Vérification des habilitations sur une actualité 38

5.7.1. Description des transactions métiers 38

5.8. S008 : Parcourir la mosaïque des actualités 41

5.8.1. Description des transactions métiers 41

5.9. S009 : Restitution d’une image d’une section 42

5.9.1. Description des transactions métiers 42

5.10. Dictionnaire des services métiers 43

6. Services REST 44

6.1. Glossaire et précision 44

6.1.1. Mode de transmission 44

6.1.2. Code erreur 44

6.2. SR001 : Créer une actualité 45

6.2.1. Adressage du service 45

6.2.2. Paramètre(s) d’entrée 45

6.2.3. Valeur de retour 47

6.2.4. Implémentation 48

6.3. SR002 : Consulter une actualité 49

6.3.1. Adressage du service 49

6.3.2. Paramètre(s) d’entrée 49

6.3.3. Valeur de retour 50

6.3.4. Implémentation 51

6.4. SR003 : Supprimer une actualité 53

6.4.1. Adressage du service 53

6.4.2. Paramètre(s) d’entrée 53

6.4.3. Valeur de retour 53

6.4.4. Implémentation 54

6.5. SR004 : Modifier une actualité 56

6.5.1. Adressage du service 56

6.6. SR005 : Rechercher des actualités 57

6.6.1. Adressage du service 57

6.7. SR006 : Parcourir la mosaïque 58

6.7.1. Adressage du service 58

6.8. SR007 : Consulter l’image d’une section 59

6.8.1. Adressage du service 59

6.9. Dictionnaire des services REST 60

# Objet du document

# Contexte

L’objectif du chantier est la réalisation d’un micro-service exposant une fonctionnalité de gestion/consultation des actualités.

Centrée sur le moteur de cette fonctionnalité ledit micro-service ne prendra pas en charge l’aspect interface utilisateur.

Il appartiendra à chaque projet intégrant le micro-service de réaliser la partie interface utilisateur en utilisant la ou les technologies de vue de son choix.

Afin de permettre à la fois une intégration simple, et un découplage quant aux technologies ou langages de programmation, le micro-service exposera une API REST permettant les interactions avec le moteur de cette fonctionnalité.

## Principe des actualités

Afin d’identifier les services que doit fournir et exposer le microservice, il est important de maitriser les enjeux fonctionnels liés à un module d’actualités.

Cette fonctionnalité permet à une population donnée de rédiger et de publier des articles, sous forme d’actualités, auprès d’un public de lecteurs.

Illustration :



Chaque lecteur a accès à une vue panoramique sur les actualités disponibles, et peu, en choisissant une d’entre elles, en consulter le détail.

Cette vue panoramique peu se présenter sous la forme d’une mosaïque affichant pour chaque actualité, son titre et un résumé introductif.

Illustration :

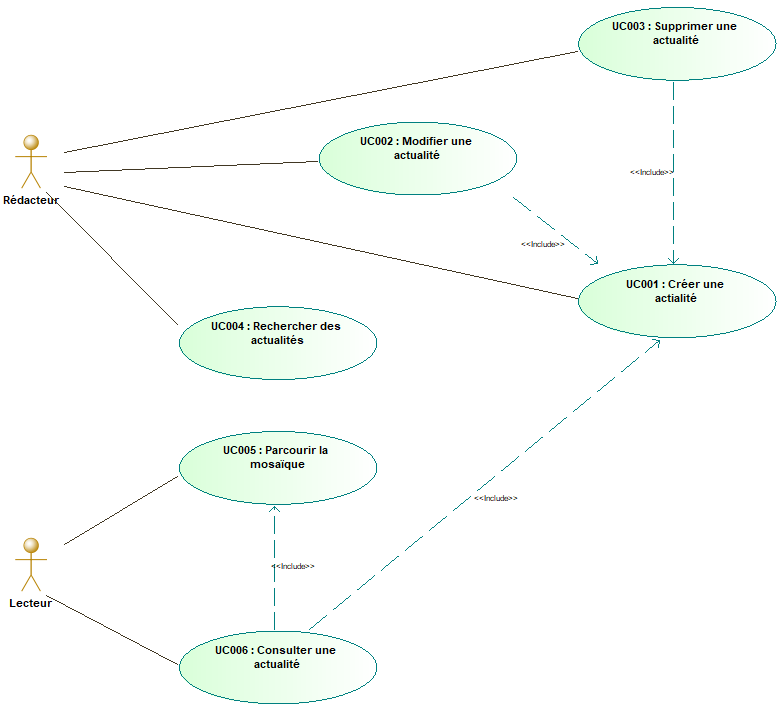




## Cas d’utilisation : Actualité

Le diagramme de cas d’utilisation ci-dessous illustre les cas d’usage les plus courants concernant la fonctionnalité « actualité ».

### Diagramme



### Description détaillée

#### Les acteurs

Rédacteur :

Cet acteur a la charge de créer et diffuser les actualités. Il en produit le contenu éditorial et en définit la structure. Par la suite, il diffuse les actualités auprès du public souhaité composé de lecteurs.

Lecteur :

Cet acteur est la cible des actualités produites et diffusées par le rédacteur. Pour prendre connaissance de l’actualité, il accède en lecture seule à l’article qui la compose.

#### UC001 : Créer une actualité

Acteur : Rédacteur

Précondition(s) : *aucune*

Un rédacteur peut créer de nouvelles actualités et en définir le contenu éditorial.

Dans le cadre de cette création, il pourra structurer l’actualité en plusieurs sections représentant chacune un paragraphe de l’article final.

Chaque section dispose des éléments suivants :

* Un titre de section *(sous-titre de l’actualité)* ;
* Une image illustrative ;
* Un titre pour l’image illustrative ;
* Un contenu texte ;
* Un ensemble de lien vers d’autres ressources *(ex : URL d’autres sites internet)*.

En plus du contenu de l’actualité, le rédacteur peut aussi définir les éléments suivants :

* Une période de diffusion pendant laquelle l’actualité sera disponible aux lecteurs *(représentée par une date de début et une date de fin)* ;
* Les habilitations donnant accès à l’actualité en modification/suppression ;
* Les filtres permettant de restreindre l’affichage de l’actualité à une certaine population ;
* Le caractère prioritaire de l’actualité afin qu’elle soit mise en avant auprès des lecteurs.

#### UC002 : Modifier une actualité

Acteur : Rédacteur

Précondition(s) :

* L’actualité doit avoir été, au préalable, créée *(cf. UC001)* ;
* L’acteur doit disposer des habilitations d’accès à l’actualité.

Un rédacteur peut modifier tout ou partie du contenu éditorial d’une actualité. Il peut aussi modifier ou spécifier une période de diffusion, ainsi que modifier le caractère prioritaire de l’actualité concernée.

#### UC003 : Supprimer une actualité

Acteur : Rédacteur

Précondition(s) :

* L’actualité doit avoir été, au préalable, créée *(cf. UC001)* ;
* L’acteur doit disposer des habilitations d’accès à l’actualité.

Un rédacteur habilité peut supprimer une actualité dans sa totalité.

#### UC004 : Rechercher des actualités

Acteur : Rédacteur

Précondition(s) : *aucune*

Afin de facilité la gestion des actualités, un rédacteur dispose d’un moteur de recherche lui permettant de sourcer les actualités qu’il souhaite modifier, supprimer ou tout simplement consulter.

Ce moteur de recherche propose différents critères comme la période de diffusion, le statut prioritaire ou le titre de l’actualité.

Le résultat d’une recherche ne pourra contenir que les actualités pour lesquelles le rédacteur à les habilitations requises.

Par souci de confort et de performance, le résultat de la recherche peut être paginé.

#### UC005 : Parcourir la mosaïque

Acteur : Lecteur

Précondition(s) : aucune

La mosaïque est un aperçu sur l’ensemble des actualités diffusées. Cette vue panoramique affiche pour chaque actualité un résumé introductif, voir même une image illustrative.

La mosaïque est le point d’entrée privilégié vers le détail d’une actualité.

Dans la mosaïque, les actualités sont ordonnées par de date de publication décroissante : de la plus récente à la moins récente.

Certaines actualités à caractères prioritaires pourront déroger à cet ordre et être positionnées en tête de fil quel que soit leur date de publication.

Certaines actualités ne pourront être présentes dans la mosaïque que pour un public ciblé, et donc affichées pour certains types d’utilisateurs et pas d’autres.

Des filtres discriminants pourront parfois être disponibles afin de restreindre l’affichage la mosaïque à certaines catégories d’actualités.

Par souci de confort et de performance, et en fonction du volume d’actualité disponible, la mosaïque pourra être paginée.

#### UC006 : Consulter une actualité

Acteur : Lecteur

Précondition(s) :

* L’actualité doit avoir été, au préalable, créée *(cf. UC001)* ;
* La consultation ne peut se faire que via la mosaïque.

Une actualité peut être consultée en détail par un lecteur. De fait, il peut avoir accès *(en lecture)* à la totalité du contenu éditorial, des images et des liens vers d’autres ressources *(ex : autre site internet)*.

# Modèle de données

## Modèle conceptuel

Ci-dessous le MCD représentant les entités du métier à persister et leurs mises en relation.

Diagramme :



## Dictionnaire de données

Voici ci-dessous, regroupé par entité, la description des données.

### Habilitation

Code : HABILITATION

Cette entité représente la liste des codes autorisés à accéder à une actualité donnée. Ladite entité participera aux différents contrôles d’accès que peut subir un rédacteur afin de savoir si ce dernier est habilité à modifier une actualité.

#### Attributs

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **Code** | **Type** | **Obl.** |
| Code d'accès | HAB\_CODE\_ACCES | Chaîne[50] | Oui |
| * **Description** : Code autorisé à accéder à une actualité. | | | |

### Actualité

Code : ACTUALITE

Cette entité représente une actualité. Les différentes parties qui peuvent constituer une actualité sont rassemblées autour de cette entité.

#### Attributs

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **Code** | **Type** | **Obl.** |
| Identifiant de l’actualité | ACT\_ID | Entier | Oui |
| * **Description** : Identifiant unique d’une actualité. | | | |
| Titre de l’actualité | ACT\_TITRE | Chaîne[100] | Oui |
| * **Description** : Titre principal d’une actualité. | | | |
| L’introduction de l’actualité | ACT\_INTRODUCTION | Chaîne[400] | Non |
| * **Description** : Introduction présentant l’actualité. | | | |
| Date de création de l’actualité | ACT\_DATE\_CREATION | Date & heure | Oui |
| * **Description** : Date de création de l’actualité. Cette date inclut l’heure exacte. | | | |
| Date de modification de l‘actualité | ACT\_DATE\_MODIFICATION | Date & heure | Oui |
| * **Description** : Date de modification de l’actualité. Cette date inclut l’heure exacte. | | | |
| Date de début de diffusion | ACT\_DATE\_DEBUT\_DIFFUSION | Date & heure | Non |
| * **Description** : Date à partir de quand l’actualité doit être disponible. Cette date inclut l’heure exacte. | | | |
| Date de fin de diffusion | ACT\_DATE\_FIN\_DIFFUSION | Date & heure | Non |
| * **Description** : Date après laquelle l’actualité ne doit plus être disponible. Cette date inclut l’heure exacte. | | | |
| Indicateur : prioritaire | ACT\_PRIORITAIRE | Booléen | Non |
| * **Description** : Indicateur informant si l’actualité a un caractère prioritaire. | | | |
| Indicateur : active | ACT\_ACTIVE | Booléen | Non |
| * **Description** : Indicateur informant si l’actualité est active. | | | |

### Filtre d’affichage

Code : FILTRE

Cette entité permet de définir des critères de filtre sur des actualités. Le principe de filtre est mis en œuvre dans tout mécanisme destiner à restreindre l’affichage de certaines actualités à une population de lecteurs donnée.

Chaque actualité peut être rattachée à plusieurs filtres.

Chaque filtre est représenté par un couple code > valeur.

#### Attributs

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **Code** | **Type** | **Obl.** |
| Identifiant du filtre | FLT\_ID | Entier | Oui |
| * **Description** : Identifiant unique du filtre. | | | |
| Code du filtre | FLT\_CODE | Chaîne[50] | Oui |
| * **Description** : Code permettant de nommer un filtre. Afin de pouvoir représenter une liste de valeur pour un filtre donné, un même code peut être associé plusieurs fois à une même actualité. | | | |
| Valeur du filtre | FLT\_VALEUR | Chaîne[50] | Oui |
| * **Description** : Valeur associée au code. | | | |

### Les sections

Code : SECTION

Une actualité est organisée en une ou plusieurs sections. Chaque section est un bloc de texte pouvant le cas échéant être illustré par une image. Lesdites sections sont représentées par cette entité.

#### Attributs

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **Code** | **Type** | **Obl.** |
| Identifiant de la section | SEC\_ID | Entier | Oui |
| * **Description** : Identifiant unique de la section. | | | |
| Titre de la section | SEC\_TITRE | Chaîne[100] | Non |
| * **Description** : Titre *(facultatif)* de la section. | | | |
| Texte de la section | SEC\_TEXTE | Texte | Oui |
| * **Description** : Bloc de texte contenu dans la section | | | |
| Ordre de la section | SEC\_ORDRE | Texte | Oui |
| * **Description** : Définit l’ordre de la section dans la liste de celle composant l’actualité. | | | |

### Les ressources *(sous forme de lien)*

Code : RESSOURCE

En plus d’un texte est d’une image illustrative, une section peut présenter des liens vers d’autres ressources *(site web, …)*. Cette entité représente ce type de ressource sous forme de lien.

Une section peut agréger plusieurs ressources.

#### Attributs

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **Code** | **Type** | **Obl.** |
| Identifiant de la ressource | RES\_ID | Entier | Oui |
| * **Description** : Identifiant unique de la ressource. | | | |
| L’URL de la ressource | RES\_URL | Chaîne[255] | Oui |
| * **Description** : Lien URL vers la ressource. | | | |
| Le libellé de la ressource | RES\_LIBELLE | Chaîne[100] | Oui |
| * **Description** : Le libellé désignant la ressource. | | | |
| Ordre de la ressource | RES\_ORDRE | Entier | OUI |
| * **Description** : Libellé du contributeur à l’origine du commentaire. | | | |

### Image illustrative

Code : IMAGE

Chaque section d’une actualité peut être illustrée par une **et une seule** image.

#### Attributs

|  |  |  |  |
| --- | --- | --- | --- |
| **Nom** | **Code** | **Type** | **Obl.** |
| Libellé de l’image | IMG\_LIBELLE | Chaîne[100] | Oui |
| * **Description** : Libellé désignant l’image. | | | |
| Type MIME du fichier image | IMG\_CONTENT\_TYPE | Chaîne[255] | Oui |
| * **Description** : Type MIME du fichier image. | | | |
| Taille du fichier de l’image | IMG\_TAILLE | Entier | Oui |
| * **Description** : Taille en octet du fichier de l’image. | | | |
| Contenu de l’image | IMG\_CONTENU | Blob | Oui |
| * **Description** : Fichier de l’image. | | | |

## Modèle physique

Le tiers de persistance choisi étant une base de données, le MPD présenté ci-dessous et orienté en ce sens. Le SGBDR choisi est PostgreSQL.

Diagramme



Les objets composants le modèle physique *(table, séquence, …)* appartiennent au schéma « **actualite** ».

AUCUN objet n’est créé dans le schéma par défaut « **public** ».

### Autres objets du modèle

Voici un focus sur les autres objets du modèle physique nécessaires pour le développement de l’application.

#### Séquence

Cette rubrique dresse la liste des séquences, et précise pour quelle table elles doivent être utilisé.

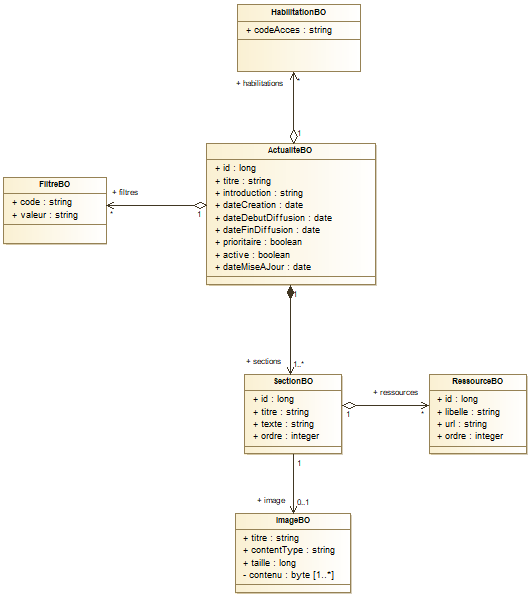
|  |  |  |
| --- | --- | --- |
| **Séquence** | **Table** | **Colonne** |
| ACTUALITE\_ACT\_ID\_SEQ | ACTUALITE | ACT\_ID |
| FILTRE\_FLT\_ID\_SEQ | FILTRE | FLT\_ID |
| SECTION\_SEC\_ID\_SEQ | SECTION | SEC\_ID |
| RESSOURCE\_RES\_ID\_SEQ | RESSOURCE | RES\_ID |

# qReprésentation des données

## Les entités

Les entités applicatives représentant les entités du modèle de données ont été modélisées dans le diagramme de classe ci-dessous.

Diagramme :



Mapping :

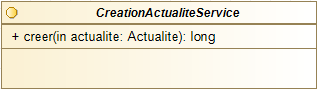
|  |  |  |
| --- | --- | --- |
| **Classe** | **Entité conceptuelle** | **Table physique** |
| HabilitationBO | Habilitation | HABILITATION |
| ActualiteBO | Actualité | ACTUALITE |
| FiltreBO | Filtre d’affichage | FILTRE |
| SectionBO | Les sections | SECTION |
| RessourceBO | Les ressources | RESSOURCE |
| ImageBO | Image illustrative | IMAGE |

# Services métiers

A des fins d’organisation et de structuration de l’application, l’ensemble des transactions métiers, identifiés dans le cadre de l’analyse des cas d’utilisation, ont été organisées en service métiers. En voici la description détaillée.

## S001 : Création d’une actualité

Nom technique : CreationActualiteService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet de créer une nouvelle actualité à partir de paramètres d’entrée.

#### TX001 : « créer »

Objectif :

Créer en base de données une nouvelle actualité à partir des informations fournies.

Prérequis :

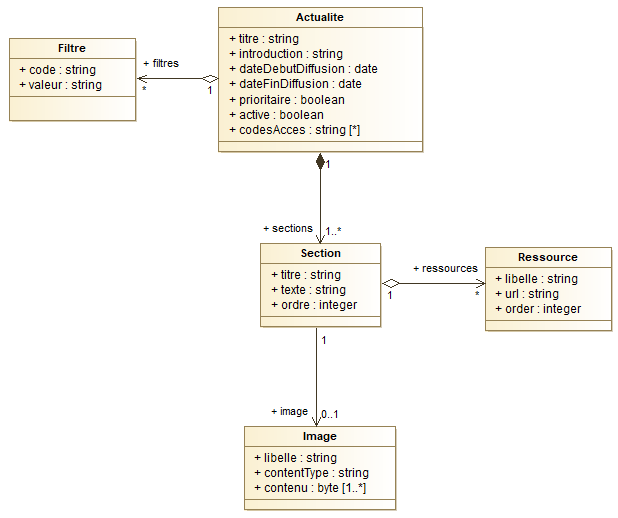
Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètre d’entrée *(1 seul)* :

* + Paramètre n°1 « actualite » :

DTO regroupant les informations nécessaires à la création d’une nouvelle actualité.

Diagramme :



Valeur de retour :

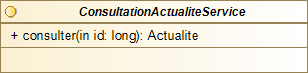
Identifiant technique de l’actualité nouvellement créée.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 - Créer l’en-tête de l’actualité**  Cette activité consiste à créer l’en-tête de l’actualité. Le cas échéant cet entête peut définir les habilitations ainsi que les filtres d’une actualité. | |
| **Code** | **Description** |
| S001-TX001-RG001 | Insérer une ligne dans la table ACTUALITE en renseignant les colonnes suivantes :   * ACT\_ID : valeur générée à partir de la séquence dédiée ; * ACT\_TITRE : « DTO > actualite.titre » ; * ACT\_INTRODUCTION : « DTO > actualite.introduction » *(si renseigné)* ; * ACT\_DATE\_CREATION : data et heure courantes ; * ACT\_DATE\_MODIFICATION : data et heure courantes ; * ACT\_DATE\_DEBUT\_MODIFICATION : « DTO > actualite.dateDebutDiffusion » *(si renseigné)* ; * ACT\_DATE\_FIN\_MODIFICATION : « DTO > actualite.dateFinDiffusion » *(si renseigné)* ; * ACT\_PRIORITAIRE : « DTO > actualite.prioritaire » *(si renseigné)* ; * ACT\_ACTIVE : « DTO > actualite.active ». |
| S001-TX001-RG002 | Si le DTO contient des codes d’accès dans le champ « actualite.codesAcces » alors, **pour chacun de ces codes**, insérer une ligne dans la table HABILITATION en renseignant les colonnes suivantes :   * ACT\_ID : identifiant de l’actualité concernée *(cf.S001-TX001-RG001)* ; * HAB\_CODE\_ACCES : « DTO > actualite.codesAcces ». |
| S001-TX001-RG003 | Si le DTO contient des filtres dans le champ « actualite.filtres » alors, **pour chacun de ces filtres**, insérer une ligne dans la table FILTRE en renseignant les colonnes suivantes :   * FLT\_ID : valeur générée à partir de la séquence dédiée ; * ACT\_ID : identifiant de l’actualité concernée *(cf.S001-TX001-RG001)* ; * FLT\_CODE : «  DTO > actualite.filtres.code » ; * FLT\_VALEUR : « DTO > actualite.filtres.valeur ». |
|  | |
| **Activité n°2 - Créer les sections**  Une actualité nouvellement créée doit avoir obligatoirement une section et peut en contenir plusieurs. Cette activité consiste à créer lesdites sections. | |
| **Code** | **Description** |
| S001-TX001-RG004 | Pour chacune des sections contenues dans le champs « actualite.sections » du DTO, insérer une ligne dans la table SECTION en renseignant les colonnes suivantes :   * SEC\_ID : valeur générée à partir de la séquence dédiée ; * ACT\_ID : identifiant de l’actualité concernée *(cf.S001-TX001-RG001)* ; * SEC\_TITRE : «  DTO > actualite.sections.titre » ; * SEC\_TEXTE : «  DTO > actualite.sections.texte » ; * SEC\_ORDRE : «  DTO > actualite.sections.ordre ». |
|  | |
| **Activité n°3 – Créer l’image associée à chaque section**  En plus des éléments textuels, une section peut *(c’est facultatif)* être complétée par une image. Cette activité consiste à traiter cette image le cas échéant. | |
| **Code** | **Description** |
| S001-TX001-RG005 | Si une section, contenue dans le champ « actualite.sections » du DTO, contient une image dans le champ « actualite.sections.image » alors, insérer une ligne dans la table IMAGE en renseignant les colonnes suivantes :   * SEC\_ID : identifiant de la section concernée *(cf.S001-TX001-RG004)* ; * IMG\_LIBELLE : «  DTO > actualite.sections.image.libelle » ; * IMG\_CONTENT\_TYPE : «  DTO > actualite.sections.image.contentType » ; * IMG\_TAILLE : «  DTO > actualite.sections.image.taille » ; * IMG\_CONTENU : «  DTO > actualite.sections.image.contenu ». |
|  | |
| **Activité n°4 – Créer le(s) ressource(s) associée à chaque section**  En plus des éléments textuels, une section peut *(c’est facultatif)* être complétée par une ou plusieurs ressource(s). Cette activité consiste à traiter ce(s) ressource(s) dans le cas échéant. | |
| **Code** | **Description** |
| S001-TX001-RG006 | Si une section, contenue dans le champ « actualite.sections » du DTO, contient une ou plusieurs ressources dans le champ « actualite.sections.ressources » alors, Pour chacune des ressouces insérer une ligne dans la table RESSOURCE en renseignant les colonnes suivantes :   * RES\_ID : valeur générée à partir de la séquence dédiée ; * SEC\_ID : identifiant de la section concernée *(cf.S001-TX001-RG004)* ; * RES \_URL : «  DTO > actualite.sections.ressources.url » ; * RES \_LIBELLE : «  DTO > actualite.sections.ressources.libelle» ; * RES \_ORDRE : «  DTO > actualite.sections. ressources .ordre ». |
|  | |
| **Activité n°5 – Retourner l’identifiant de l’actualité** | |
| S001-TX001-RG007 | Une fois l’actualité entièrement créée, la transaction doit retourner l’identifiant de ladite actualité *(cf.S001-TX001-RG001).* |

## S002 : Consultation d’une actualité

Nom technique : ConsultationActualiteService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet d’obtenir la totalité d’une actualité à partir de paramètres d’entrée.

#### TX001 : « consulter »

Objectif :

Retourner la totalité des informations d’une actualité sous la forme d’un DTO. Ledit DTO sera construit à partir des données de la base.

Prérequis :

Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètre d’entrée *(1 seul)* :

* + Paramètre n°1 « id » :

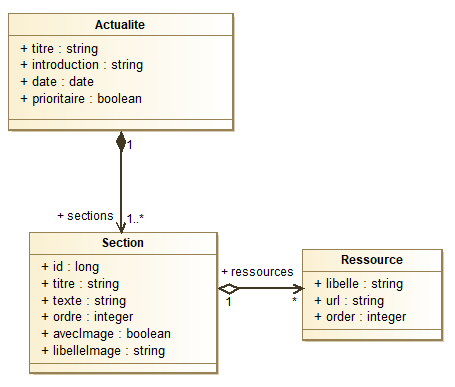
Identifiant technique de l’actualité à retourner.

Valeur de retour :

DTO contenant la totalité des informations de l’actualité.

**ATTENTION :** le contenu binaire du fichier image de chaque section n’est pas inclus.

Diagramme :

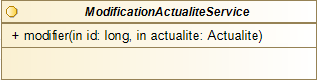


Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 - Construire le DTO actualité** | |
| **Code** | **Description** |
| S002-TX001-RG001 | A partir de l’identifiant de l’actualité fourni en paramètre d’entrée, récupérer la ligne de données de la table ACTUALITE répondant au critère suivant :   * ACT\_ID = id *(paramètre fourni en entrée)* * ACT\_ACTIVE = TRUE |
| S002-TX001-RG002 | A partir des données récupérées, construire une instance du DTO « Actualite » en renseignant les champs ci-dessous à partir des données indiquées :   * Actualite.titre : colonne ACTUALITE.ACT\_TITRE ; * Actualite.introduction : colonne ACTUALITE.ACT\_INTRODUCTION ; * Actualite.date : colonne ACTUALITE.ACT\_DATE\_DEBUT\_DIFFUSION si NOT NULL, sinon colonne ACTUALITE.ACT\_DATE\_MODIFICATION ; * Actualite.prioritaire : colonne ACTUALITE.ACT\_PRIORITAIRE.   L’alimentation des champs suivants est décrite dans le cadre d’une autre activité :   * Actualite.sections : « Voir l’activité n°2 » ; |
|  | |
| **Activité n°2 – Charger les sections associées à l’actualité** | |
| **Code** | **Description** |
| S002-TX001-RG003 | A partir de l’identifiant de l’activité fourni en paramètre d’entrée, récupérer toutes les lignes de données de la table SECTION répondant au critère suivant :   * ACT\_ID = id *(paramètre fourni en entrée)* |
| S002-TX001-RG004 | A partir des données récupérées, **composer une liste d’instances** du DTO « Section » en renseignant les champs suivants :   * Section.id : colonne SECTION.SEC\_ID ; * Section.titre : colonne SECTION.SEC\_TITRE ; * Section.texte : colonne SECTION.SEC\_TEXTE ; * Section.ordre : colonne SECTION.SEC\_ORDRE ; * Section.avecImage : « true » s’il existe une image dans la table IMAGE pour la section concernée, sinon « false » ; * Section.lebelleImage : colonne IMAGE.IMG\_LIBELLE s’il existe une image dans la table IMAGE pour la section concernée, sinon null.   L’alimentation des champs suivants est décrite dans le cadre d’une autre activité :   * Section.ressources : « Voir l’activité n°3 ».   Une fois la liste de sections composée, renseigner le champ suivant du DTO « Actualite » :  Actualite.sections : Liste de [instance de DTO « Section »]. |
|  | |
| **Activité n°3 – Charger les ressources associées à la section**  Cette activité est à réaliser pour chaque DTO « Section » créé lors de l’activité n°2. | |
| **Code** | **Description** |
| S002-TX001-RG005 | A partir de l’identifiant de la section, récupérer toutes les lignes de données de la table RESSOURCE répondant au critère suivant :   * SEC\_ID = Identifiant de la section. |
| S002-TX001-RG006 | A partir des données récupérées, **composer une liste d’instances** du DTO « Ressource » en renseignant les champs suivants :   * Ressource.libelle : colonne RESSOURCE.RES\_LIBELLE ; * Ressource.url : colonne RESSOURCE.RES\_URL ; * Ressource.ordre : colonne RESSOURCE.RES\_ORDRE ;   Puis, à l’aide de cette liste, renseigner le champ suivant du DTO « Section » :   * Section.ressources : Liste de [instance de DTO « Ressource »]. |
|  | |
| **Activité n°4 – Retourner le DTO représentant la totalité de l’actualité** | |
| S002-TX001-RG007 | Une fois l’actualité entièrement recomposée *(hors contenu binaire des images)*, la transaction doit retourner le DTO qui en résulte. |

## S003 : Modification d’une actualité

Nom technique : ModificationActualiteService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet de modifier une actualité à partir des paramètres d’entrée.

#### TX001 : « modifier »

Objectif :

Modifier en base de données les informations d’une actualité à partir des informations fournies.

Paramètre d’entrée *(2)* :

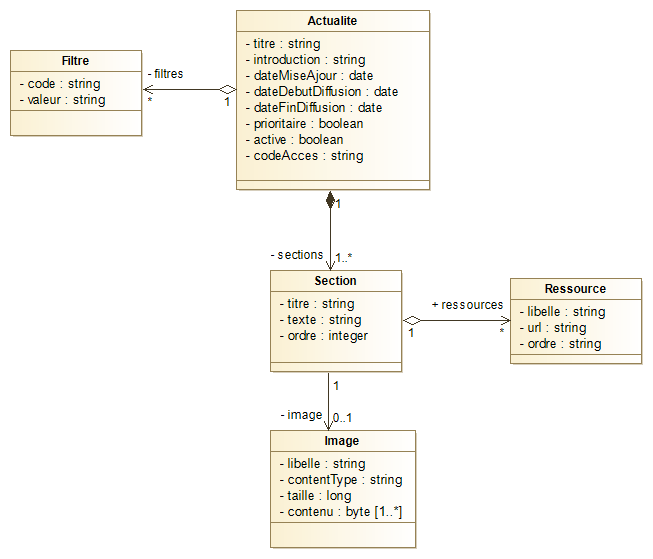
* + Paramètre n°1 « id » :

Identifiant technique de l’actualité à modifier.

* + Paramètre n°2 « actualite » :

DTO regroupant les informations nécessaires à la modification d’une actualité.

Diagramme :



Valeur de retour :

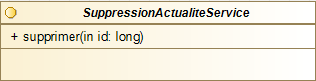
Aucune.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 - Créer l’en-tête de l’actualité à modifier**  Cette activité consiste à créer l’en-tête de l’actualité à modifier à partir du DTO fourni en paramètre. | |
| **Code** | **Description** |
| S003-TX001-RG001 | A partir de l’identifiant de l’actualité fourni en paramètre d’entrée, mettre à jour la ligne dans la table ACTUALITE en renseignant les colonnes suivantes :   * ACT\_ID : « id > identifiant reçue en paramètre » ; * ACT\_TITRE : « DTO > actualite.titre » ; * ACT\_INTRODUCTION : « DTO > actualite.introduction » *(si renseigné)* ; * ACT\_DATE\_CREATION : data et heure récupérer dans la base de données; * ACT\_DATE\_MODIFICATION : data et heure courantes ; * ACT\_DATE\_DEBUT\_MODIFICATION : « DTO > actualite.dateDebutDiffusion » *(si renseigné)* ; * ACT\_DATE\_FIN\_MODIFICATION : « DTO > actualite.dateFinDiffusion » *(si renseigné)* ; * ACT\_PRIORITAIRE : « DTO > actualite.prioritaire » *(si renseigné)* ; * ACT\_ACTIVE : « DTO > actualite.active ». |
| S003-TX001-RG002 | Supprimer l’ensemble des filtres de la table FILTRE concernés par l’actualité identifiée par paramètre d’entrée « id ». |
| S003-TX001-RG003 | Supprimer l’ensemble des habilitations de la table HABILITATION concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S003-TX001-RG004 | Si le DTO contient des codes d’accès dans le champ « actualite.codesAcces » alors, **pour chacun de ces codes**, insérer une ligne dans la table HABILITATION en renseignant les colonnes suivantes :   * ACT\_ID : identifiant de l’actualité concernée *(cf.S003-TX001-RG001)* ; * HAB\_CODE\_ACCES : « DTO > actualite.codesAcces ». |
| S003-TX001-RG005 | Si le DTO contient des filtres dans le champ « actualite.filtres » alors, **pour chacun de ces filtres**, insérer une ligne dans la table FILTRE en renseignant les colonnes suivantes :   * FLT\_ID : « valeur générée à partir de la séquence dédiée » ;; * ACT\_ID : identifiant de l’actualité concernée *(cf.S003-TX001-RG001)* ; * FLT\_CODE : «  DTO > actualite.filtres.code » ; * FLT\_VALEUR : « DTO > actualite.filtres.valeur ». |
|  | |
| **Activité n°2 - Créer les sections**  Une actualité nouvellement créée doit avoir obligatoirement une section et peut en contenir plusieurs. Cette activité consiste à créer lesdites sections. | |
| **Code** | **Description** |
| S003-TX001-RG006 | Supprimer l’ensemble des images de la table IMAGE concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S003-TX001-RG007 | Supprimer l’ensemble des ressources de la table RESSOURCE concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S003-TX001-RG008 | Supprimer l’ensemble des sections de la table SECTION concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S003-TX001-RG009 | Pour chacune des sections contenues dans le champs « actualite.sections » du DTO, insérer une ligne dans la table SECTION en renseignant les colonnes suivantes :   * SEC\_ID : «  valeur générée à partir de la séquence dédiée ;» ; * ACT\_ID : identifiant de l’actualité concernée *(cf.S003-TX001-RG001)* ; * SEC\_TITRE : «  DTO > actualite.sections.titre » ; * SEC\_TEXTE : «  DTO > actualite.sections.texte » ; * SEC\_ORDRE : «  DTO > actualite.sections.ordre ». |
|  | |
| **Activité n°3 – Créer l’image associée à chaque section**  En plus des éléments textuels, une section peut *(c’est facultatif)* être complétée par une image. Cette activité consiste à traiter cette image le cas échéant. | |
| **Code** | **Description** |
| S003-TX001-RG010 | Si une section, contenue dans le champ « actualite.sections » du DTO, contient une image dans le champ « actualite.sections.image » alors, insérer une ligne dans la table IMAGE en renseignant les colonnes suivantes :   * SEC\_ID : identifiant de la section concernée *(cf.S003-TX001-RG009)* ; * IMG\_LIBELLE : «  DTO > actualite.sections.image.libelle » ; * IMG\_CONTENT\_TYPE : «  DTO > actualite.sections.image.contentType » ; * IMG\_TAILLE : «  DTO > actualite.sections.image.taille » ; * IMG\_CONTENU : «  DTO > actualite.sections.image.contenu ». |
|  | |
| **Activité n°4 – Créer le(s) ressource(s) associée à chaque section**  En plus des éléments textuels, une section peut *(c’est facultatif)* être complétée par une ou plusieurs ressource(s). Cette activité consiste à traiter ce(s) ressource(s) dans le cas échéant. | |
| **Code** | **Description** |
| S003-TX001-RG011 | Si une section, contenue dans le champ « actualite.sections » du DTO, contient une ou plusieurs ressources dans le champ « actualite.sections.ressources » alors, Pour chacune des ressouces insérer une ligne dans la table RESSOURCE en renseignant les colonnes suivantes :   * RES\_ID : valeur générée à partir de la séquence dédiée ; * SEC\_ID : identifiant de la section concernée *(cf.S003-TX001-RG009)* ; * RES \_URL : «  DTO > actualite.sections.ressources.url » ; * RES \_LIBELLE : «  DTO > actualite.sections.ressources.libelle» ; * RES \_ORDRE : «  DTO > actualite.sections. ressources .ordre ». |

## S004 : Suppression d’une actualité

Nom technique : SuppressionActualiteService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet de supprimer la totalité d’une actualité à partir de son identifiant.

#### TX001 : « supprimer »

Objectif :

Supprimer en base de données l’ensemble des informations relatives à une actualité donnée.

Prérequis :

Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètre d’entrée *(1 seul)* :

* + Paramètre n°1 « id » :

Identifiant technique de l’actualité à supprimer.

Valeur de retour :

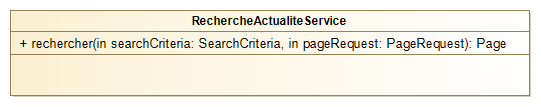
Aucune.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 - Supprimer la totalité d’une actualité** | |
| **Code** | **Description** |
| S004-TX001-RG001 | Supprimer l’ensemble des images de la table IMAGE concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S004-TX001-RG002 | Supprimer l’ensemble des ressources de la table RESSOURCE concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S004-TX001-RG003 | Supprimer l’ensemble des sections de la table SECTION concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S004-TX001-RG004 | Supprimer l’ensemble des filtres de la table FILTRE concernés par l’actualité identifiée par paramètre d’entrée « id ». |
| S004-TX001-RG005 | Supprimer l’ensemble des habilitations de la table HABILITATION concernées par l’actualité identifiée par paramètre d’entrée « id ». |
| S004-TX001-RG006 | Supprimer de la table ACTUALITE l’actualité identifiée par paramètre d’entrée « id ». |

## S005 : Recherche d’actualités

Nom technique : RechercheActualiteService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet de recherchercher des actualités à partir de paramètres d’entrée.

#### TX001 : « rechercher »

* + - 1. Objectif :
         1. Rechercher en base des actualité à partir des informations fournies.
      2. Prérequis :
         1. Accès à la base de données *(via la méthode d’accès dédiée)*.
      3. Paramètre d’entrée *(2 seul)* :
  + Paramètre n°1 « searchCritéria » :

Objet regroupant les critère de recherches, les filtres discriminant ainsi que les codes d’accès.

* + Paramètre n°1 « pageRequest » :

Objet regroupant les critère de pagination du reésultat de recherche.

1. Valeur de retour :

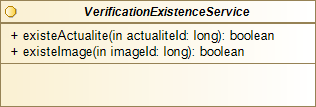
Une liste d’actualités paginée pour lesquelles le rédacteur à les habilitations requises.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 – Rechercher des actualités** | |
| **Code** | **Description** |
| S005-TX001-RG001 | Rechercher en base de données les actualité correspondants aux critères fourni en paramètre d’entrée |

## S006 : Vérification de l’existence d’un élément d’une actualité.

Nom technique : VerificationExistenceService



### Description des transactions métiers

Voici la description détaillée des transactions métiers qui composent le présent service.

#### TX001 : « vérifier l’existence de l’actualité »

Objectif :

Vérifier en base de données si une actualité existe pour l’identifiant fourni.

Prérequis :

Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètre d’entrée:

* + Paramètre n°1 « actualiteId » :

Identifiant technique de l’actualité dont l’existence doit être vérifiée.

Valeur de retour :

Booléen indiquant si l’actualité identifiée existe ou non : « true » elle existe, « false » elle n’existe pas.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 – Vérifier l’existence d’une actualité** | |
| **Code** | **Description** |
| S006-TX001-RG001 | A partir de l’identifiant « actualiteId » fourni en paramètre d’entrée, rechercher si une ligne de données présente dans la table ACTUALITE répond au critère suivant :   * ACT\_ID = actualiteId *(paramètre fourni en entrée)* ;     En fonction du résultat :   * Si une ligne existe, alors l’actualité identifiée existe. |
|  | |
| **Activité n°2 –** **Retourner le résultat** | |
| S006-TX001-RG002 | Si l’actualité existe, retourner « true » sinon retourner « false ». |

#### TX002 : « vérifier l’existence d’une image »

Objectif :

Vérifier en base de données si une image existe pour l’identifiant fourni.

Prérequis :

Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètre d’entrée:

* + Paramètre n°1 « imageId » :

Identifiant technique de l’image dont l’existence doit être vérifiée.

Valeur de retour :

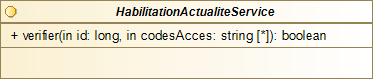
Booléen indiquant si l’image existe ou non : « true » elle existe, « false » elle n’existe pas.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 – Vérifier l’existence de l’image** | |
| **Code** | **Description** |
| S006-TX002-RG001 | A partir de l’identifiant « imageId » fourni en paramètre d’entrée, rechercher si une ligne de données présente dans la table IMAGE répond au critère suivant :   * SEC\_ID = imageId *(paramètre fourni en entrée)* ;     En fonction du résultat :   * Si une ligne existe, alors l’image identifiée existe. |
|  | |
| **Activité n°2 – Retourner le résultat** | |
| S006-TX002-RG002 | Si l’image existe, retourner « true » sinon retourner « false ». |

## S007 : Vérification des habilitations sur une actualité

Nom technique : HabilitationActualiteService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet de vérifier les habilitations d’accès sur une actualité à partir de paramètres d’entrée.

#### TX001 : « vérifier »

Objectif :

Vérifier en base de données si l’un des codes d’accès permet l’accès à l’actualité identifiée.

Prérequis :

Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètres d’entrée:

* + Paramètre n°1 « id » :

Identifiant technique de l’actualité dont l’accès doit être vérifié.

* + Paramètre n°2 « codesAcces » :

Ensemble des codes d’accès à contrôler.

Valeur de retour :

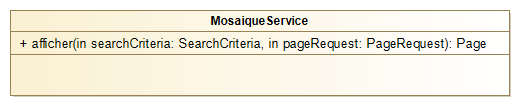
Booléen indiquant si l’actualité est accessible ou non : « true » elle est accessible, « false » elle n’est pas accessible.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 – Vérifier l’accès à l’actualité** | |
| **Code** | **Description** |
| S007-TX001-RG001 | Il y a deux cas possibles qui autorisent l’accès à l’actualité identifiée :   * Cas n°1 : Aucune habilitation n’a été définie pour l’actualité identifiée dans la table HABILITATION ; * Cas n°2 : Il y a une intersection entre la liste des codes d’accès fournie en paramètre, et les habilitations définies pour l’actualité identifiée dans la table HABILITATION. |
|  | |
| **Activité n°2 – Retourner le résultat** | |
| S007-TX001-RG002 | Si l’actualité est accessible, retourner « true » sinon retourner « false ». |

## S008 : Parcourir la mosaïque des actualités

Nom technique : MosaiqueService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet l’accès à des éléments mosaique des actualités à partir de paramètres d’entrée.

#### TX001 : « afficher »

Objectif :

Retourner des éléments mosaique des actualités paginé. Les éléments sont construit à partir des données de la base.

Prérequis :

Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètre d’entrée *(2 seul)* :

* + Paramètre n°1 « searchCriteria » :

Objet regroupant les critère de recherches, les filtres discriminant ainsi que les codes d’accès.

* + Paramètre n°2 « pageRequest » :

Objet regroupant les critère de pagination du reésultat de recherche.

Valeur de retour :

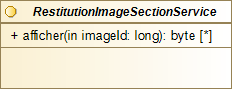
Une liste d’élément mosaique paginée pour lesquelles le lecteur à les autorisations en fonction des paramètres d’entrées.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 – Afficher éléments mosaique** | |
| **Code** | **Description** |
| S008-TX001-RG001 | Rechercher en base de données les actualité correspondants aux critères fourni en paramètre d’entrée |

## S009 : Restitution d’une image d’une section

Nom technique : RestitutionImageSectionService



### Description des transactions métiers

Le présent service n’expose qu’une seule transaction métier. Cette dernière permet de restituer la totalité du contenu d’une image à partir du paramètre d’entrée.

#### TX001 « Afficher »

Objectif :

Retourner la totalité du contenu (tableau de byte) d’une image. Ledit contenu sera construit à partir des données de la base.

Prérequis :

Accès à la base de données *(via la méthode d’accès dédiée)*.

Paramètre d’entrée *(1 seul)* :

* + Paramètre n°1 « id » :

Identifiant technique du contenu de l’image à retourner.

Valeur de retour :

Chaîne de caractères encoder en base64 représentant l’image.

Activité*(s)* et règle*(s)* de gestion :

|  |  |
| --- | --- |
| **Activité n°1 - Construire le tableau de byte de l’image** | |
| **Code** | **Description** |
| S009-TX001-RG001 | A partir de l’identifiant de l’image fourni en paramètre d’entrée, récupérer la la colonne IMG\_CONTENU de la table IMAGE répondant au critère suivant :   * SEC\_ID = id *(paramètre fourni en entrée)* |
|  | |
| **Activité n°2 – Retourner le tableau de byte représentant le contenu de l’image** | |
| S009-TX001-RG002 | Une fois le tableau de byte complètement rempli, la transaction doit retourner ce dernier comme résultat. |

## Dictionnaire des services métiers

Voici ci-dessous la liste exhaustive de l’ensemble des services métiers disponibles dans l’application.

|  |  |
| --- | --- |
| **Service** | **Rôle** |
| **S001** | Création d’une actualité. |
| **S002** | Consultation d’une actualité. |
| **S003** | Modification d’une actualité. |
| **S004** | Suppression d’une actualité. |
| **S005** | Recherche d’actualités. |
| **S006** | Vérification de l’existence d’un élément d’une actualité. |
| **S007** | Vérification des habilitations sur une actualité. |
| **S008** | Parcourir la mosaïque des actualités. |
| **S009** | Consultation de l’image d’une section. |

# Services REST

## Glossaire et précision

### Mode de transmission

Dans le cadre des échanges entre un client et un service REST, des données peuvent être envoyées du client vers le service. Ces données peuvent être transmises de différentes manières. Les modes de transmission possibles sont classés de la manière suivante :

* **BODY** :

Les données entrantes sont transmises dans le corps de la requête.

* **REQUEST** :

Les données entrantes sont transmises comme paramètre de la requête, ex : …?param1=**valeur1**&param2=**valeur2**

* **URI** :

Les données entrantes font partie de l’URI ex : /host/service/**valeur**

### Code erreur

Dans le cadre de l’implémentation de service REST, les notifications d’erreurs se font en respectant la norme HTTP sur les codes retours.

Le tableau ci-dessous dresse la liste des codes utilisés :

|  |  |
| --- | --- |
| **Code** | **Cause** |
| **400** | Requête entrante male formée, exemple erreur identifiée lors de la validation des paramètres d’entrée. |
| **403** | Accès interdit, exemple : les habilitations ne permettent pas l’accès à une ressources. |
| **404** | Ressource introuvable, exemple : aucune ressource ne correspond à un identifiant. |
| **500** | Erreur serveur, exemple : problème technique inattendu. |

## SR001 : Créer une actualité

Ce service REST permet à un client de créer une nouvelle actualité en lui fournissant les informations nécessaires. En retour, le servie renvoi l’identifiant technique de l’actualité nouvellement créée.

### Adressage du service

* Méthode HTTP supportée : **POST** ;
* Mapping d’URI : « **/actualite/creer** ».

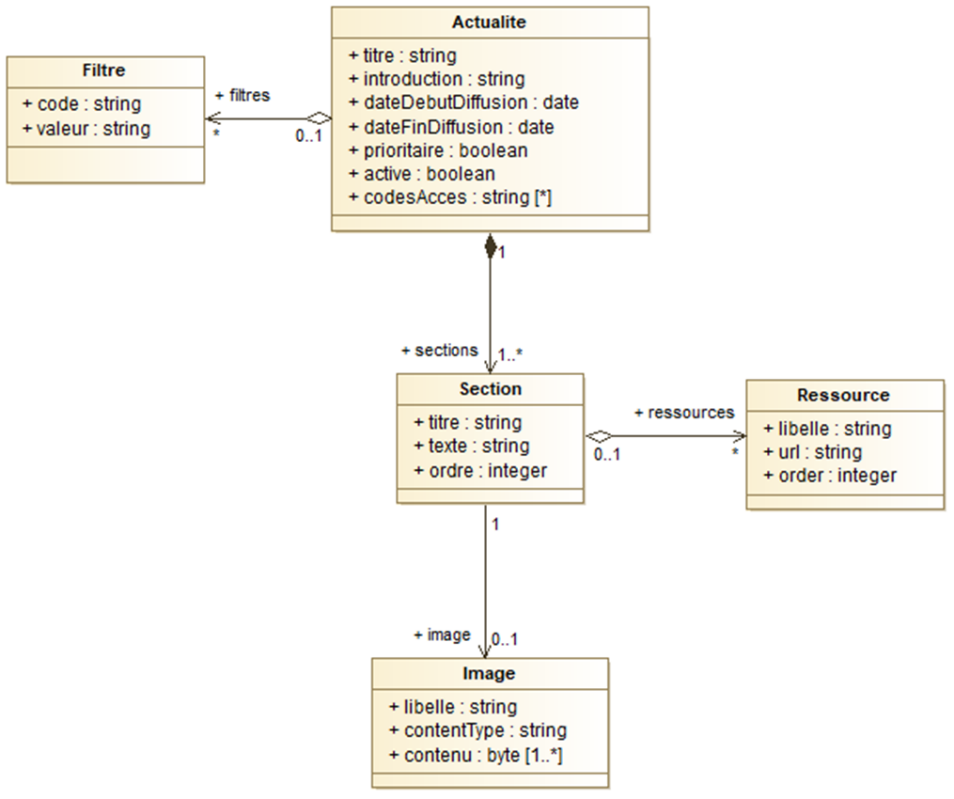
### Paramètre(s) d’entrée

#### Paramètre n°1 : « actualite »

Mode de transmission : **BODY**

Ce paramètre est une structure de données au format JSON. Le diagramme de classe ci-dessous en présente l’organisation.

Diagramme :



Contraintes :

|  |  |
| --- | --- |
| **Actualite** | |
| **Champ** | **Contraintes** |
| titre | Obligatoire.  100 caractères maximum. |
| introduction | Facultatif.  400 caractères maximum. |
| dateDebutDiffusion | Facultatif.  Date. |
| dateFinDiffusion | Facultatif.  Date. |
| prioritaire | Facultatif.  Booléen. |
| active | Obligatoire.  Booléen. |
| codesAcces | Facultatif.  50 caractères maximum pour chacun des codes. |
| filtres | Facultatif.  Tableau d’objet.  Filtre |
| sections | Obligatoire.  Tableau d’objet.  Section |

|  |  |
| --- | --- |
| **Filtre** | |
| **Champ** | **Contraintes** |
| code | Obligatoire.  50 caractères maximum. |
| valeur | Obligatoire.  50 caractères maximum. |

|  |  |
| --- | --- |
| **Section** | |
| **Champ** | **Contraintes** |
| titre | Facultatif.  100 caractères maximum. |
| texte | Obligatoire.  Texte. |
| ordre | Obligatoire.  Entier. |
| ressources | Facultatif.  Tableau d’objet.  Ressource |
| image | Facultatif.  Image |

|  |  |
| --- | --- |
| **Ressource** | |
| **Champ** | **Contraintes** |
| libelle | Obligatoire.  100 caractères maximum. |
| url | Obligatoire.  255 caractères maximum. |
| ordre | Obligatoire.  Entier. |

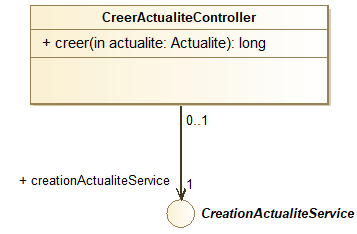
### Valeur de retour

Un entier représentant l’identifiant technique de l’actualité nouvellement créé.

### Implémentation

Le service REST est implémenté par une classe « CreerActuliteController » de type contrôleur. Ci-dessous une représentation de cette classe et de ses dépendances.

Diagramme :



#### Activité(s) et règle(s) de gestion

|  |  |
| --- | --- |
| **Activité n°1 – Analyse de la requête entrante**  Cette activité consiste à valider les données portées par la requête entrante. | |
| **Code** | **Description** |
| SR001-RG001 | Sur la base des contraintes énoncées à la rubrique « paramètre(s) d’entrée », valider le format des données entrante.   * Si les données sont non conformes, renvoyer au client un message d’erreur *(type d’erreur HTTP 400)* ; * Si les données sont conformes, poursuivre l’activité. |
|  | |
| **Activité n°2 – Créer la nouvelle actualité** | |
| **Code** | **Description** |
| SR001-RG002 | A partir des données entrantes, créer un DTO « Actualite » tel qu’attendu par me service métier « S001 ».  Puis utiliser le service métier « S001 » afin de créer, à partir du DTO, une nouvelle actualité.   * Long = creationActualiteService.creer(actualite) |
|  | |
| **Activité n°3 – Retourner l’identifiant de l’actualité** | |
| SR001-RG003 | A l’issue de la création de la nouvelle actualité, le service métier retourne l’identifiant technique de l’actualité concernée.  Le service REST doit renvoyer cette identifiant au client. |

## SR002 : Consulter une actualité

Ce service REST permet à un client de consulter la totalité d’une actualité à partir de son identifiant technique. ATTENTION : le contenu binaire des images n’est pas inclus dans cette totalité.

### Adressage du service

* Méthode HTTP supportée : **GET** ;
* Mapping d’URI : « **/actualite/consulter/{id}** ».

### Paramètre(s) d’entrée

#### Paramètre n°1 : « id »

Mode de transmission : **URI**

Ce paramètre est un entier représentant l’identifiant technique de l’actualité à consulter.

Contraintes :

Ce paramètre est obligatoire.

#### Paramètre n°2 : « codeAcces »

Mode de transmission : **REQUEST**

Ce paramètre est une chaîne de caractères représentant un code d’habilitation. Ce paramètre peut être présent plusieurs fois dans la requête.

Un code d’habilitation permet de déterminer si la consultation de l’actualité est autorisée.

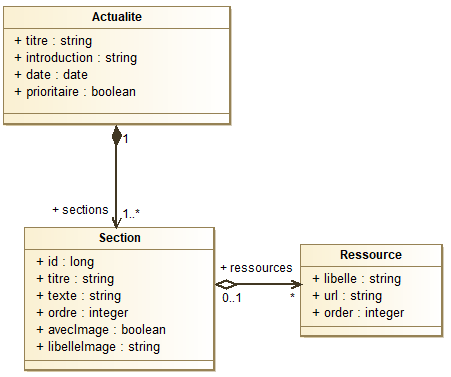
Contraintes :

Ce paramètre est facultatif.

### Valeur de retour

Une structure au format JSON représentant la totalité d’une actualité. Le diagramme de classe ci-dessous en présente l’organisation.

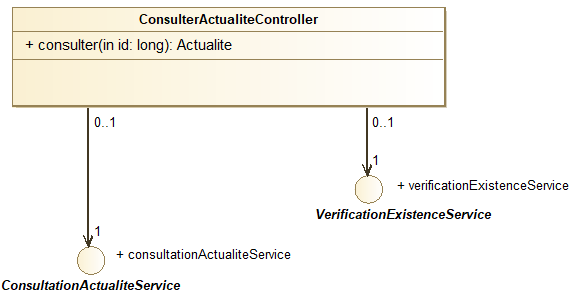
Diagramme :



### Implémentation

Le service REST est implémenté par une classe « ConsulterActualiteController » de type contrôleur. Ci-dessous une représentation de cette classe et de ses dépendances.

Diagramme :



#### Activité(s) et règle(s) de gestion

|  |  |
| --- | --- |
| **Activité n°1 – Analyse de la requête entrante**  Cette activité consiste à valider les données portées par la requête entrante. | |
| **Code** | **Description** |
| SR002-RG001 | Sur la base des contraintes énoncées à la rubrique « paramètre(s) d’entrée », valider le format des données entrante.   * Si les données sont non conformes, renvoyer au client un message d’erreur *(type d’erreur HTTP 400)* ; * Si les données sont conformes, poursuivre l’activité. |
|  | |
| **Activité n°2 – Vérifier l’existence de l’actualité** | |
| **Code** | **Description** |
| SR002-RG002 | A partir de l’identifiant « id » fourni en paramètre d’entrée, et à l’aide du service métier « S006 », vérifier qu’il existe une actualité associée :   * Booléen = verificationExistenceService.existeActualite(id)   En fonction du résultat :   * Si aucune actualité n’existe pour cet identifiant, renvoyer au client un message d’erreur *(type d’erreur HTTP 404)* ; * S’il en existe une, poursuivre l’activité. |
|  | |
| **Activité n°3 – Charger la totalité de l’actualité** | |
| SR002-RG003 | A partir de l’identifiant « id », récupérer auprès du service métier « S002 » la totalité de l’actualité.  Actualite = consultationActualiteService.consulter(id) |
|  | |
| **Activité n°4 – Retourner la totalité de l’actualité** | |
| SR002-RG004 | A partir du DTO « Actualite » composé et renvoyer au client la structure JSON attendue. |

## SR003 : Supprimer une actualité

Ce service REST permet à un client de supprimer la totalité d’une actualité à partir de son identifiant technique.

### Adressage du service

* Méthode HTTP supportée : **DELETE** ;
* Mapping d’URI : « **/actualite/supprimer/{id}** ».

### Paramètre(s) d’entrée

#### Paramètre n°1 : « id »

Mode de transmission : **URI**

Ce paramètre est un entier représentant l’identifiant technique de l’actualité à supprimer.

Contraintes :

Ce paramètre est obligatoire.

#### Paramètre n°2 : « codeAcces »

Mode de transmission : **REQUEST**

Ce paramètre est une chaîne de caractères représentant un code d’habilitation. Ce paramètre peut être présent plusieurs fois dans la requête.

Un code d’habilitation permet de déterminer si la suppression de l’actualité est autorisée.

Contraintes :

Ce paramètre est facultatif.

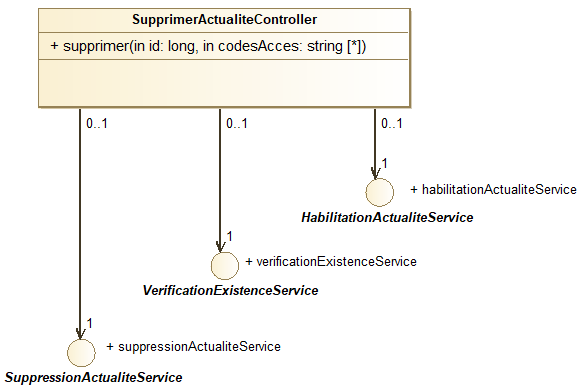
### Valeur de retour

Aucune

### Implémentation

Le service REST est implémenté par une classe « SupprimerActualiteController » de type contrôleur. Ci-dessous une représentation de cette classe et de ses dépendances.

Diagramme :



#### Activité(s) et règle(s) de gestion

|  |  |
| --- | --- |
| **Activité n°1 – Analyse de la requête entrante**  Cette activité consiste à valider les données portées par la requête entrante. | |
| **Code** | **Description** |
| SR003-RG001 | Sur la base des contraintes énoncées à la rubrique « paramètre(s) d’entrée », valider le format des données entrante.   * Si les données sont non conformes, renvoyer au client un message d’erreur *(type d’erreur HTTP 400)* ; * Si les données sont conformes, poursuivre l’activité. |
|  | |
| **Activité n°2 – Vérifier l’existence de l’actualité** | |
| **Code** | **Description** |
| SR003-RG002 | A partir de l’identifiant « id » fourni en paramètre d’entrée, et à l’aide du service métier « S012 », vérifier qu’il existe une actualité associée :   * Booléen = verificationExistenceService.existeActualite(id)   En fonction du résultat :   * Si aucune actualité n’existe pour cet identifiant, renvoyer au client un message d’erreur *(type d’erreur HTTP 404)* ; * S’il en existe une, poursuivre l’activité. |
|  | |
| **Activité n°3 – Vérifier les habilitations sur l’actualité** | |
| SR003-RG003 | A partir de l’identifiant « id » et du/des codes habilitations « codeAcces » fournis en paramètre d’entrée, vérifier auprès du service métier « S007 » si l’accès à l’actualité identifiée est autorisé à au moins un code d’habilitation.   * Booléen = habilitationActualiteService.verifier(id, codesAcces)   En fonction du résultat :   * Si l’accès n’est pas autorisé, renvoyer au client un message d’erreur *(type d’erreur HTTP 403)* ; * Si l’accès est autorisé, poursuivre l’activité. |
|  | |
| **Activité n°4 – Supprimer la totalité de l’actualité** | |
| SR003-RG004 | A partir de l’identifiant « id », supprimer à l’aide du service métier « S004 » la totalité de l’actualité identifiée.  suppressionActualiteService.supprimer(id) |

## SR004 : Modifier une actualité

Ce service REST permet à un client de modifier la totalité d’une actualité. L’ensemble du contenu de l’actualité identifiée est remplacé par les données entrantes.

### Adressage du service

* Méthode HTTP supportée : **PUT** ;
* Mapping d’URI : « **/actualite/modifier/{id}** ».

### Paramètre(s) d’entrée

#### Paramètre n°1 : « id »

Mode de transmission : **URI**

Ce paramètre est un entier représentant l’identifiant technique de l’actualité à modifier.

Contraintes :

Ce paramètre est obligatoire.

#### Paramètre n°2 : « actualite »

Mode de transmission : **BODY**

Une structure au format JSON représentant la totalité de l’actualité à modifier.

Contraintes :

Ce paramètre est obligatoire.

#### Paramètre n°3 : « codeAcces »

Mode de transmission : **REQUEST**

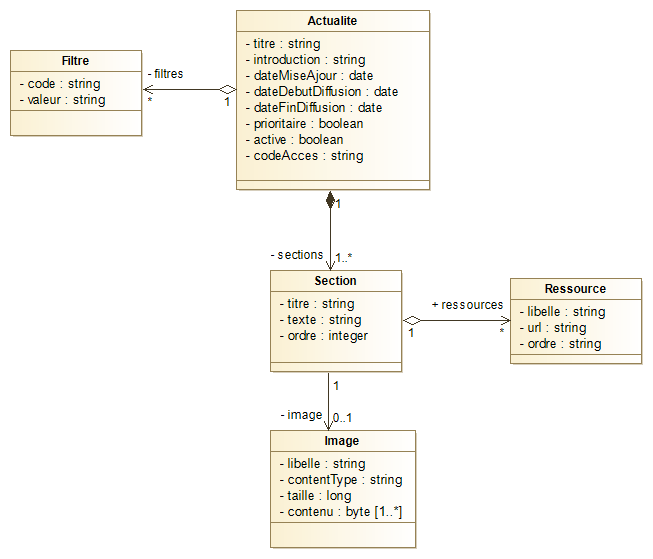
Ce paramètre est une chaîne de caractères représentant un code d’habilitation. Ce paramètre peut être présent plusieurs fois dans la requête.

Un code d’habilitation permet de déterminer si la modification de l’actualité est autorisée.

Contraintes :

Ce paramètre est facultatif.

Diagramme :



Contraintes :

|  |  |
| --- | --- |
| **Actualite** | |
| **Champ** | **Contraintes** |
| titre | Obligatoire.  100 caractères maximum. |
| introduction | Facultatif.  400 caractères maximum. |
| dateDebutDiffusion | Facultatif.  Date. |
| dateFinDiffusion | Facultatif.  Date. |
| prioritaire | Facultatif.  Booléen. |
| active | Obligatoire.  Booléen. |
| sections | Obligatoire.  Tableau d’objet.  Section |

|  |  |
| --- | --- |
| **Section** | |
| **Champ** | **Contraintes** |
| titre | Facultatif.  100 caractères maximum. |
| texte | Obligatoire.  Texte. |
| ordre | Obligatoire.  Entier. |
| ressources | Facultatif.  Tableau d’objet.  Ressource |
| image | Facultatif.  Image |

|  |  |
| --- | --- |
| **Ressource** | |
| **Champ** | **Contraintes** |
| libelle | Obligatoire.  100 caractères maximum. |
| url | Obligatoire.  255 caractères maximum. |
| ordre | Obligatoire.  Entier. |

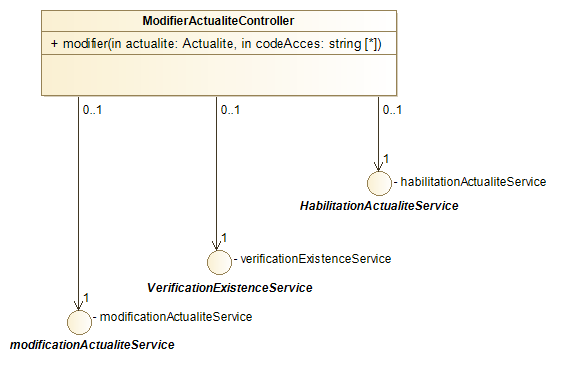
### Valeur de retour

Aucune.

### Implémentation

Le service REST est implémenté par une classe « ModifierActualiteController » de type contrôleur. Ci-dessous une représentation de cette classe et de ses dépendances.

Diagramme :



#### Activité(s) et règle(s) de gestion

|  |  |
| --- | --- |
| **Activité n°1 – Analyse de la requête entrante**  Cette activité consiste à valider les données portées par la requête entrante. | |
| **Code** | **Description** |
| SR004-RG001 | Sur la base des contraintes énoncées à la rubrique « paramètre(s) d’entrée », valider le format des données entrante.   * Si les données sont non conformes, renvoyer au client un message d’erreur *(type d’erreur HTTP 400)* ; * Si les données sont conformes, poursuivre l’activité. |
|  | |
| **Activité n°2 – Vérifier l’existence de l’actualité** | |
| **Code** | **Description** |
| SR004-RG002 | A partir de l’identifiant « id » fourni en paramètre d’entrée, et à l’aide du service métier « S012 », vérifier qu’il existe une actualité associée :   * Booléen = verificationExistenceService.existeActualite(id)   En fonction du résultat :   * Si aucune actualité n’existe pour cet identifiant, renvoyer au client un message d’erreur *(type d’erreur HTTP 404)* ; * S’il en existe une, poursuivre l’activité. |
|  | |
| **Activité n°3 – Vérifier les habilitations sur l’actualité** | |
| SR004-RG003 | A partir de l’identifiant « id » et du/des codes habilitations « codeAcces » fournis en paramètre d’entrée, vérifier auprès du service métier « S007 » si l’accès à l’actualité identifiée est autorisé à au moins un code d’habilitation.   * Booléen = habilitationActualiteService.verifier(id, codesAcces)   En fonction du résultat :   * Si l’accès n’est pas autorisé, renvoyer au client un message d’erreur *(type d’erreur HTTP 403)* ; * Si l’accès est autorisé, poursuivre l’activité. |
|  | |
| **Activité n°4 – modifier l’actualité** | |
| SR004-RG004 | A partir de l’identifiant « id », modifier à l’aide du service métier « S003 » l’actualité identifiée.  ModificationActualiteService.modifier(id, Actualite actualite) |

## SR005 : Rechercher des actualités

Ce service REST permet de sourcer un ensemble d’actualités sur la base de critères de recherche et des habilitations. Le résultat d’une recherche est présenté sous une forme paginée.

### Adressage du service

* Méthode HTTP supportée : **GET** ;
* Mapping d’URI : « **/actualite/rechercher** ».

### Paramètre(s) d’entrée

#### Paramètre n°1 : « codesAcces »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une chaîne de caractères représentant un code d’habilitation. Ce paramètre peut être présent plusieurs fois dans la requête.
      3. Un code d’habilitation permet de déterminer si la consultation ou la recherche de l’actualité est autorisée.
      4. Contraintes :
      5. Ce paramètre est facultatif.

#### Paramètre n°2 : « filtres »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une chaîne de caractères représentant un code ou une valeur de filtre. Ce paramètre peut être présent plusieurs fois dans la requête.
      3. Un code ou valeur de filtre permet de déterminer si l’actualité fait partie de la catégories recherchés.
      4. Contraintes :
      5. Ce paramètre est facultatif.

#### Paramètre n°3 : « titre »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une chaîne de caractères représentant le titre de l’actualité.
      3. Un titre d’actualité permet de faire des recherches sur les actualités ayant ce titre ou ayant cette chaine de caractères dans leur titre.
      4. Contraintes :
      5. Ce paramètre est facultatif.

#### Paramètre n°4 : « dateDeCreation »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une date représentant la date de création de l’actualité.
      3. Une date de création d’actualité permet de faire des recherches sur les actualités ayant été créée à cette date ou ultérieurement.
      4. Contraintes :
      5. Ce paramètre est facultatif.

#### Paramètre n°5 : « dateDebutDiffusion »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une date représentant la date de début de diffusion de l’actualité.
      3. Une date de début de diffusion d’une actualité permet de faire des recherches sur les actualités ayant été diffusées à partir de cette date ou ultérieurement.
      4. Contraintes :
      5. Ce paramètre est facultatif.

#### Paramètre n°6 : « dateFinDiffusion »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une date représentant la date de fin de diffusion de l’actualité.
      3. Une date de fin de diffusion d’une actualité permet de faire des recherches sur les actualités ayant été arrêté d’être diffusé à partir de cette date.
      4. Contraintes :
      5. Ce paramètre est facultatif.

#### Paramètre n°7 : « prioritaire »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est un booléen représentant le caractère prioritaire de l’actualité.
      3. Ce paramètre permet de faire des recherches sur les actualités prioritaire ou inversement.
      4. Contraintes :
      5. Ce paramètre est facultatif.

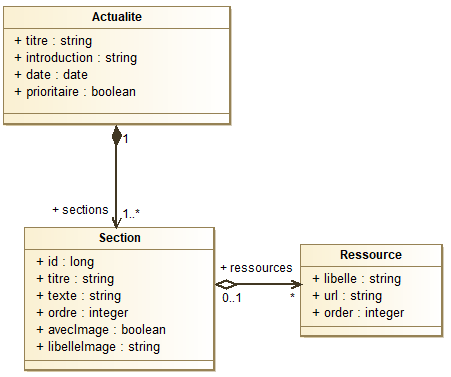
#### Paramètre n°8 : « active »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est un booléen représentant le caractère active de l’actualité.
      3. Ce paramètre permet de faire des recherches sur les actualités qui sont toujours active ou inversement.
      4. Contraintes :
      5. Ce paramètre est facultatif.

### Valeur de retour

Une structure au format JSON représentant l’ensemble des actualités répondant aux différents critères. Le diagramme de classe ci-dessous en présente l’organisation d’une des actualités.

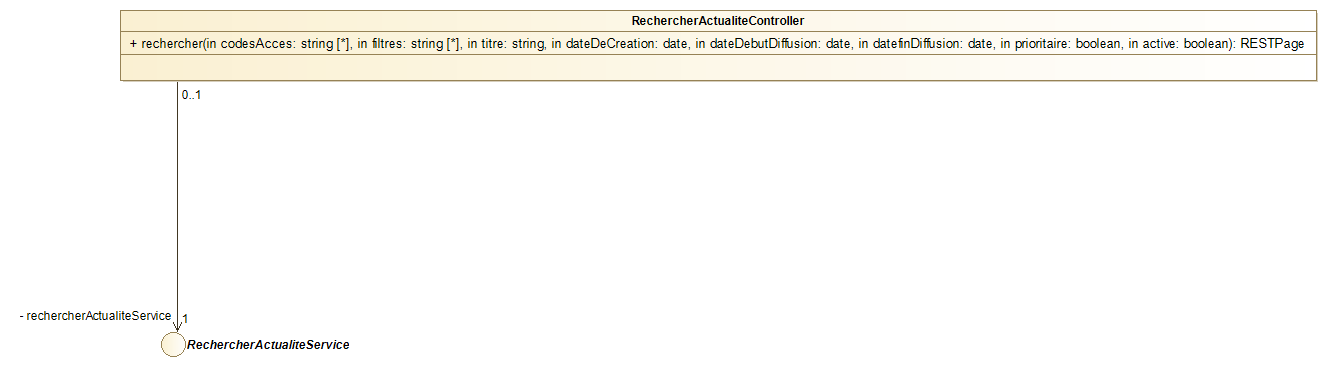
Diagramme :



### Implémentation

Le service REST est implémenté par une classe «RechercherActualiteController » de type contrôleur. Ci-dessous une représentation de cette classe et de ses dépendances.

Diagramme :



#### Activité(s) et règle(s) de gestion

|  |  |
| --- | --- |
| **Activité n°1 – Récupération des critères**  Cette activité consiste à récupérer les critères de recherches. | |
| **Code** | **Description** |
| SR005-RG001 | Récupération des paramètres de pagination dans un objet PageRequest |
| SR005-RG002 | Sur la base des contraintes énoncées à la rubrique « paramètre(s) d’entrée », Récupérer l’ensemble des critère d’entrée en les insérant dans un objet de type SearchCriteria |
|  | |
| **Activité n°2 – Rechercher les actualités** | |
| **Code** | **Description** |
| SR005-RG003 | A partir de l’objet SearchCriteria (précédement rempli) et des critères de pagination, et à l’aide du service métier « S005 », rechercher les actualités correspondant aux critères:   * Page = rechercherActualiteService.rechercher(searchCriteria, pageRequest) * Mettre l’objet Page récupérer précédement dans un autre objet RESTPage |

## SR006 : Parcourir la mosaïque

Ce service REST permet de parcourir la mosaïque présentant les actualités disponibles. En fonction de certains filtres, la mosaïque peut cibler des catégories d’actualités en particulier. La mosaïque peut être paginée.

### Adressage du service

* Méthode HTTP supportée : **GET** ;
* Mapping d’URI : « **/actualite/mosaique** ».

### Paramètre(s) d’entrée

#### Paramètre n°1 : « codesAcces »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une chaîne de caractères représentant un code d’habilitation. Ce paramètre peut être présent plusieurs fois dans la requête.
      3. Un code d’habilitation permet de déterminer si la consultation ou la recherche de l’actualité est autorisée.
      4. Contraintes :
      5. Ce paramètre est facultatif.

#### Paramètre n°2 : « filtres »

* + - 1. Mode de transmission : **REQUEST**
      2. Ce paramètre est une chaîne de caractères représentant un code ou une valeur de filtre. Ce paramètre peut être présent plusieurs fois dans la requête.
      3. Un code ou valeur de filtre permet de déterminer si l’actualité fait partie de la catégories recherchés.
      4. Contraintes :
      5. Ce paramètre est facultatif.

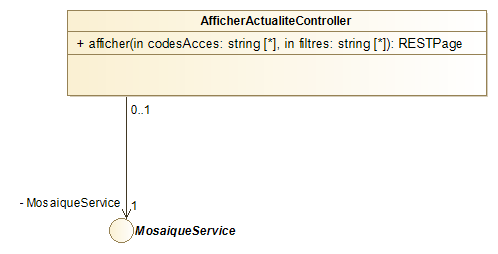
### Valeur de retour

Une structure au format JSON représentant l’ensemble éléments mosaïque répondant aux différents critères (paramètres d’entrée).

### Implémentation

Le service REST est implémenté par une classe «AfficherActualiteController » de type contrôleur. Ci-dessous une représentation de cette classe et de ses dépendances.

Diagramme :



#### Activité(s) et règle(s) de gestion

|  |  |
| --- | --- |
| **Activité n°1 – Récupération des critères**  Cette activité consiste à récupérer les critères de recherches des actualités à consulter par le lecteur. | |
| **Code** | **Description** |
| SR006-RG001 | Récupération des paramètres de pagination dans un objet PageRequest |
| SR006-RG002 | Sur la base des contraintes énoncées à la rubrique « paramètre(s) d’entrée », Récupérer l’ensemble des critère d’entrée en les insérant dans un objet de type SearchCriteria |
|  | |
| **Activité n°2 – Rechercher les actualités** | |
| **Code** | **Description** |
| SR006-RG003 | A partir de l’objet SearchCriteria (précédement rempli) et des critères de pagination, et à l’aide du service métier « S008 », rechercher les actualités correspondant aux critères:   * Page = mosaiqueService.afficher(searchCriteria, pageRequest) * Mettre l’objet Page récupérer précédement dans un autre objet RESTPage |

## SR007 : Consulter l’image d’une section

Ce service REST permet de télécharger l’image d’une section d’actualité.

### Adressage du service

* Méthode HTTP supportée : **GET** ;
* Mapping d’URI : « **/actualite/section/image/{id}** ».

### Paramètre(s) d’entrée

#### Paramètre n°1 : « id »

Mode de transmission : **URI**

Ce paramètre est un entier représentant l’identifiant technique du contenu de l’image à consulter.

Contraintes :

Ce paramètre est obligatoire.

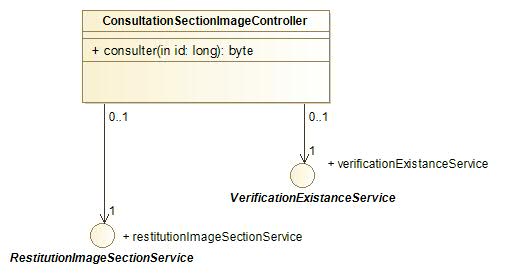
### Valeur de retour

Une structure au format JSON représentant la totalité du contenu d’une image en cha^ne de caractères base64.

### Implémentation

Le service REST est implémenté par une classe « ConsultationSectionImageController » de type contrôleur. Ci-dessous une représentation de cette classe et de ses dépendances.

Diagramme :



#### Activité(s) et règle(s) de gestion

|  |  |
| --- | --- |
| **Activité n°1 – Analyse de la requête entrante**  Cette activité consiste à valider les données portées par la requête entrante. | |
| **Code** | **Description** |
| SR007-RG001 | Sur la base des contraintes énoncées à la rubrique « paramètre(s) d’entrée », valider le format des données entrante.   * Si les données sont non conformes, renvoyer au client un message d’erreur *(type d’erreur HTTP 400)* ; * Si les données sont conformes, poursuivre l’activité. |
|  | |
| **Activité n°2 – Vérifier l’existence de l’image** | |
| **Code** | **Description** |
| SR007-RG002 | A partir de l’identifiant « id » fourni en paramètre d’entrée, et à l’aide du service métier « S006 », vérifier qu’il existe une image associée :   * Booléen = verificationExistenceService.existeImage(id)   En fonction du résultat :   * Si aucune image n’existe pour cet identifiant, renvoyer au client un message d’erreur *(type d’erreur HTTP 404)* ; * S’il en existe une, poursuivre l’activité. |
|  | |
| **Activité n°3 – Charger la totalité du contenu de l’image** | |
| SR007-RG003 | A partir de l’identifiant « id », récupérer auprès du service métier « S009 » la totalité du contenu de l’image.  Image = RestitutionImageSectionService.afficher(id) |
|  | |
| **Activité n°4 – – Retourner la totalité du contenu de l’image** | |
| SR003-RG004 | A partir du tableau de byte rempli, renvoyer au client la structure JSON attendue |

## Dictionnaire des services REST

|  |  |  |
| --- | --- | --- |
| **Code** | **Service** | **Rôle** |
| **SR001** | CreerActualiteController | Permet la création d’une nouvelle actualité. |
| **SR002** | ConsulterActualiteController | Retourne la totalité d’une actualité *(hors contenu binaire des images)*. |
| **SR003** | SupprimerActualiteController | Permet de supprimer la totalité d’une actualité si les habilitations l’autorisent. |
| **SR004** | ModifierActualiteController | Permet de modifier une actualité existante dans sa totalité. |
| **SR005** | RechercherActualiteController | Permet de rechercher un semble d’actualités répondant à certains critères et aux habilitations. |
| **SR006** | ParcourirMosaiqueController | Permet de parcourir la mosaïque des actualités disponibles. |
| **SR007** | ConsulterImageSectionController | Permet le téléchargement d’une image d’une section. |