

TP1 – WEB SERVICES - 15%

MISE EN SITUATION

Vous avez été approché par le site Internet Cinéma Montréal, pour réaliser les services qui permettront d'offrir certaines fonctionnalités pour les développeurs. Pour augmenter le nombre de site Web et d'application mobile mettant en œuvre les horaires des cinémas de la grande région de Montréal

MANDAT

En équipe de 2 ou 3 personnes, vous devez réaliser à l'aide du langage node.js et des modules nécessaires, les Web Services décrits dans la suite du document. Vous devez remettre tous les fichiers nécessaires à l'exécution de votre serveur sur un nouveau poste de travail. Vous devez remettre à la racine de votre projet un fichier README.md, contenant tous les prénoms, noms et matricules des membres de l'équipe et la documentation à suivre pour mettre en fonction le serveur.

Votre travail devra être remis dans un seul dossier de remise avant 23 h 59 le mardi 20 octobre 2015.

Vous devez aussi partager votre projet C9 avec l'utilisateur *ycharronstj*.

INTERFACE DE COMMUNICATION

Votre client vous impose certaines contraintes quant au format d'échange et à la sérialisation des objets. Les requêtes et les réponses doivent être sous la forme JSON. Voici le format demandé pour les objets du domaine de l'application

CINEMA	FILM
<pre>{ "href" : "", "nom": "", "adresse": "", "ville": "", "codePostal": "", "telephone": "", "horaires" : { "href" : "", "items": [{}, {}, {}, {}] } }</pre>	<pre>{ "href" : "", "titre": "", "pays": "", "genre": "", "classe": "", "duree": "", "realisateur": "", "imageUrl": "", "commentaires" : [{}, {}, {}] }</pre>
HORAIRE	COMMENTAIRE
<pre>{ "film" : {}, "cinema": {}, "dateHeure": "" }</pre>	<pre>{ "href" : "", "film" : {}, "texte": "", "note": "", "auteur": "", "dateHeure": "" }</pre>

Vous trouverez dans le partage le fichier Serialisation.js qui contient les exemples ci-dessus.

BASE DE DONNÉES

Vous devez créer une petite base de données qui permettra de sauvegarder les informations du système. Vous devez fournir le script de création de la base de données. Ce script crée la structure des tables et ajoute des données pour réaliser des tests. Votre base de données doit contenir 4 tables : **Cinemas**, **Films**, **Horaires** et **Commentaires**. Même s'il devrait avoir plus de tables que cela, l'objectif n'est pas de faire une base de données, vous devriez déjà savoir comment.

SERVICES WEB

CINÉMA

Les premiers services à développer permettront d'offrir les fonctionnalités importantes en lien avec les cinémas utilisant Cinéma Montréal. Voici la liste des services demandés par le client et les exigences de chacun.

L'URL de base pour cette ressource est **/cinemas**

AJOUT D'UN CINÉMA

URL	/cinemas	
Paramètres d'URL	_body : Permet de retourner ou non la représentation de l'objet dans la réponse	
Corps de requête	La représentation JSON du cinéma à ajouter	
Réponse	Réussite <ul style="list-style-type: none"> ✓ Code 201 (Created) ✓ La représentation JSON du cinéma ajouté. ✓ HEADER location contenant l'URL du cinéma ajouté. 	Échec <ul style="list-style-type: none"> ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

SÉLECTIONNER TOUS LES CINÉMAS

URL	/cinemas	
Paramètres d'URL	Aucun	
Corps de requête	Aucun	
Réponse	Réussite <ul style="list-style-type: none"> ✓ Code 200 (OK) ✓ Une collection de tous les cinémas 	Échec <ul style="list-style-type: none"> ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

SÉLECTION D'UN SEUL CINÉMA

URL	/cinemas/{uuidCinema}	
Paramètres d'URL	fields : Permet de sélectionner seulement les attributs demandés	
Corps de requête	Aucun	
Réponse	Réussite <ul style="list-style-type: none"> ✓ Code 200 (OK) ✓ La représentation JSON du cinéma ayant comme identifiant {uuidCinema} 	Échec <ul style="list-style-type: none"> ✓ Code 500 (Internal Server Error) ✓ Code 404 (Not Found) Le format de réponse d'erreur étudié en classe.

MISE À JOUR COMPLÈTE

URL	/cinemas/{uuidCinema}	
Paramètres d'URL	Aucun	
Corps de requête	La représentation JSON du cinéma à modifier	
Réponse	Réussite <ul style="list-style-type: none">✓ Code 200 (OK)✓ La représentation JSON modifiée du cinéma ayant comme identifiant {uuidCinema}.	Échec <ul style="list-style-type: none">✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

SÉLECTIONNER LES HORAIRES DU CINÉMA

URL	/cinemas/{uuidCinema}/horaires/	
Paramètres d'URL	Aucun	
Corps de requête	Aucun	
Réponse	Réussite <ul style="list-style-type: none">✓ Code 200 (OK)✓ Une collection des horaires du cinéma ayant comme identifiant {uuidCinema}.	Échec <ul style="list-style-type: none">✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

FILMS

Par la suite, vous avez à développer les services qui permettront d'offrir les fonctionnalités importantes en lien avec les films présentés dans les cinémas. Voici la liste des services demandés par le client et les exigences de chacune.

L'URL de base pour cette ressource est **/films**

AJOUT D'UN FILM

URL	/films	
Paramètres d'URL	Aucun	
Corps de requête	La représentation JSON du film à ajouter	
Réponse	Réussite ✓ Code 201 (Created) ✓ La représentation JSON du film ajouté ✓ HEADER location contenant l'URL du film ajouté.	Échec ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

SÉLECTIONNER TOUS LES FILMS

URL	/films	
Paramètres d'URL	expand : Sur la collection des commentaires. fields : Dois être rendu possible pour tous les attributs d'un film. limit et offset : Dois permettre la pagination des résultats.	
Corps de requête	Aucun	
Réponse	Réussite ✓ Code 200 (OK) ✓ Une collection de tous les films au format JSON.	Échec ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

SÉLECTION D'UN FILM

URL	/films/{uuidFilm}	
Paramètres d'URL	expand : Sur la collection des commentaires. fields : Dois être rendu possible pour tous les attributs d'un film.	
Corps de requête	Aucun	
Réponse	Réussite ✓ Code 200 (OK) ✓ La représentation JSON du film ayant comme identifiant {uuidFilm}	Échec ✓ Code 500 (Internal Server Error) ✓ Code 404 (Not Found) Le format de réponse d'erreur étudié en classe.

MISE À JOUR PARTIELLE

URL	/films/{uuidFilm}	
Paramètres d'URL	Aucun	
Corps de requête	La représentation JSON du film à modifier	
Réponse	Réussite ✓ Code 200 (OK) ✓ La représentation JSON modifiée du film ayant comme identifiant {uuidFilm}.	Échec ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

SUPPRESSION D'UN FILM

URL	/films/{uuidFilm}/	
Paramètres d'URL	Aucun	
Corps de requête	Aucun	
Réponse	Réussite ✓ Code 204 (No Content)	Échec ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

AFFICHER LES COMMENTAIRES CONCERNANT UN FILM

URL	/films/{uuidFilm}/commentaires/	
Paramètres d'URL	limit et offset	
Corps de requête	Aucun	
Réponse	Réussite ✓ Code 200 (OK)	Échec ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

AJOUTER UN COMMENTAIRE CONCERNANT UN FILM

URL	/films/{uuidFilm}/commentaires/	
Paramètres d'URL	Aucun	
Corps de requête	La représentation JSON du commentaire à ajouter	
Réponse	Réussite ✓ Code 201 (Created) ✓ La représentation JSON du commentaire ajouté ✓ HEADER location contenant URL du commentaire ajouté.	Échec ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

SUPPRIMER UN COMMENTAIRE CONCERNANT UN FILM

URL	/films/{uuidFilm}/commentaires/{uuidCommentaire}	
Paramètres d'URL	Aucun	
Corps de requête	La représentation JSON du cinéma à ajouter	
Réponse	Réussite ✓ Code 204 (No Content)	Échec ✓ Code 500 (Internal Server Error) Le format de réponse d'erreur étudié en classe.

NOTIONS AVANCÉES

PARAMÈTRES D'URL

Comme nous l'avons étudié en classe, sur les requêtes de sélection nous pouvons ajouter certains paramètres à l'URL de la requête pour demander au serveur de faire un traitement particulier. Voici les trois possibilités pour ce travail pratique. Vous devez implémenter les paramètres lorsque demandé dans la description de la fonctionnalité. N'oubliez pas que les paramètres d'URL sont situés après le ? dans l'URL. Exemple :

www.monservice.com/ressource?param1=valeur1¶m2=valeur2

EXPAND

Format	?expands={objet}
Comportement	L'objet doit être retourné avec tous ses attributs de l'objet en expansion dans la réponse.

FIELDS

Format	?fields={liste des champs} Explication : {liste des champs} : Doit être une liste d'attributs de la ressource. Chacun des attributs est séparé par une virgule (attr1, attr2, attr3)
Comportement	L'objet doit être retourné avec tous les attributs demandés dans le paramètre fields.

LIMIT ET OFFSET

Format	?offset={depart}limit={nbResultat}
Comportement	Permet d'afficher seulement une partie des résultats, {depart} et l'indice de départ et {limit} le nombre de résultat souhaités.

_BODY

Format	?_body={true false}
Comportement	Si la valeur du paramètre _body est false l'objet n'est pas retourné, dans tous les autres cas, l'objet est retourné.

LINKING

Pour ce travail, toutes les représentations retournées doivent contenir un attribut **href** qui rend possible la bonne pratique du **linking**. Un objet doit minimalement contenir cet attribut, pour permettre au code client de le retrouver au besoin. Comme vous pouvez le constater dans la section interface de communication aucun des objets ne contient de **id**, mais plutôt un **href**.

CODE D'ERREUR

Pour ce travail pratique, vous devez seulement retourner les codes d'erreur précisés dans la description du Web Service. Vous devez cependant retourner un objet erreur dans la réponse HTTP en utilisant le format suivant :

```
{
  "status" : "",
  "message": "",
  "developerMessage": ""
}
```

PONDÉRATION

Ce travail compte pour **15%** de la note final du cours, les points seront distribués de cette manière :

Élément de correction	Pourcentage
Script de création de la base de données Dois contenir des données suffisantes pour les tests	10
Web Service – Cinema	27,5
Ajout d'un cinéma	5
Sélectionner tous les cinémas	5
Sélection d'un cinéma	7,5
Mise à jour complète d'un cinéma	5
Sélectionner les horaires d'un cinéma	5
Web Service – Films	47,5
Ajout d'un film	5
Sélectionner tous les films	10
Sélection d'un film	10
Mise à jour partielle d'un film	2,5
Suppression d'un film	5
Afficher les commentaires concernant un film	7,5
Ajouter un commentaire concernant un film	5
Supprimer un commentaire concernant un film	2,5
Architecture du projet Division du code, dossiers, documentation de mise en route, ...	5
Respect des consignes, qualité du code, des commentaires et bonnes pratiques	10