

PID – Projet Réservations

Gestion du projet

- <https://openclassrooms.com/courses/4192086-gerez-votre-projet-informatique-facilement>
- <https://openclassrooms.com/courses/4296701-gerez-un-projet-digital-avec-une-methodologie-en-cascade>
- <https://openclassrooms.com/courses/4507926-initiez-vous-a-la-gestion-de-projet-agile>

Itération 0 – Découverte du projet

Introduction

Présentation du projet

Le projet consiste à informatiser la gestion des réservations de spectacles d'une société de production. Celle-ci gère un catalogue reprenant des spectacles, leurs auteurs et leurs metteurs en scènes, les comédiens, ainsi que les lieux et les dates de représentations.

L'**internaute** pourra consulter librement le catalogue des spectacles affichant le lieu et les prochaines dates de représentation. Il pourra effectuer des recherches, des tris et des filtres à travers les pages du catalogue.

Le **membre** pourra réserver des places pour une représentation d'un spectacle, consulter la liste de ses réservations et modifier ses données de profil.

L'**administrateur** pourra gérer son catalogue à travers un back-office sécurisé. Par exemple, il pourra ajouter, modifier et supprimer un spectacle manuellement, importer/exporter des données au format CSV, mais aussi mettre à jour la liste des spectacles grâce aux nouveautés publiées par un **Web service tiers**.

À son tour, l'**application** devra produire d'une part son propre Web service (une API authentifiée avec système d'affiliation), d'autre part un flux RSS (par exemple, la liste des prochaines représentations).

Cahier de charges techniques

- Développer une application Web en respectant les contraintes suivantes
 - utiliser la **programmation orientée objet** (langages *Java*, *PHP* ou *Python*),
 - exploiter un **framework** (*Spring Boot*, *Symfony*, *Laravel*, *CakePHP*, *Django*),
 - gérer les erreurs de programmation au moyen d'outils ou de **techniques de débogage** et y apporter une solution pertinente,
 - utiliser à bon escient la **documentation** disponible,
 - **optimisation** du code, du cache et des échanges avec la base de données,

- gérer un **catalogue d'éléments** (produits, membres ou services) :
 - effectuer des recherches dans le catalogue,
 - paginer, trier et filtrer les résultats de la requête,
 - sélectionner des éléments du catalogue et effectuer des actions groupées,
 - sélectionner un élément du catalogue et en afficher le détail,
 - gérer la sécurisation et les droits d'accès aux contenus (droits d'accès par rôle administrateur, utilisateur public, utilisateur enregistré, gestionnaire, etc.),
 - afficher des contenus de manière différenciée (accessibilité, langue, sécurité, fonctionnalités, disponibilité de l'information, etc.), en fonction des profils utilisateurs,
 - optimisé par **programmation asynchrone** (AJAX, JSON, XML,...),
 - exploiter des fichiers de données structurées (XML, CSV, texte,...) en lecture/écriture (importer/exporter le catalogue),
 - intégrer des **services internes et tiers** (RESTful Web services),
- déployer un **back-office d'administration** (interface administrateur sécurisé) :
 - gérer les contenus dynamiques,
 - modifier ses données de profil,
 - configurer une interface utilisateur (couleur, affichage,...),
 - interagir avec un système de gestion de bases de données (récupérer, ajouter, modifier, supprimer des enregistrements, etc.) ;
- Veiller à la sécurité
 - utiliser des outils spécifiques de protection et d'identification,
 - protéger l'application contre les injections SQL, attaques XSS, vols de session, détournement de cookies, etc.,
 - définir la réécriture d'url,
 - configurer des paramétrages et restrictions d'accès au serveur.

Données initiales

Suite à une précédente tentative d'informatisation inaboutie, la société de production dispose déjà de l'export des données de l'ancienne application sous forme de fichier au format XML. Partant de ce fichier, elle souhaite que vous conceviez une base de données relationnelle.

Extrait du fichier XML

```
<?xml version="1.0" encoding="utf-8"?>
<reservations>
    <lieu id="delvaux">
        <designation>Espace Delvaux / La Vénerie</designation>
        <adresse>
            <rue>3 rue Gratès</rue>
            <localite>1170 Watermael-Boitsfort</localite>
        </adresse>
        <website></website>
        <tel></tel>
    </lieu>
...
    <spectacle id="ayiti" reservable="1">
        <title>Ayiti</title>
        
        <auteur>Daniel Marcellin et Philippe Laurent</auteur>
        <scenographie>Daniel Marcellin et Philippe Laurent</scenographie>
        <distribution>Daniel Marcellin</distribution>
        <lieu ref="delvaux" />
        <representations>
            <representation complet="1">
                <date>12/10/2012</date>
                <heure>13:30</heure>
                <lieu ref="delvaux" />
            </representation>
            <representation>
                <date>12/10/2012</date>
                <heure>20:30</heure>
                <lieu ref="dexia" />
            </representation>
        </representations>
    </spectacle>
...
</reservations>
```

Étude comparative

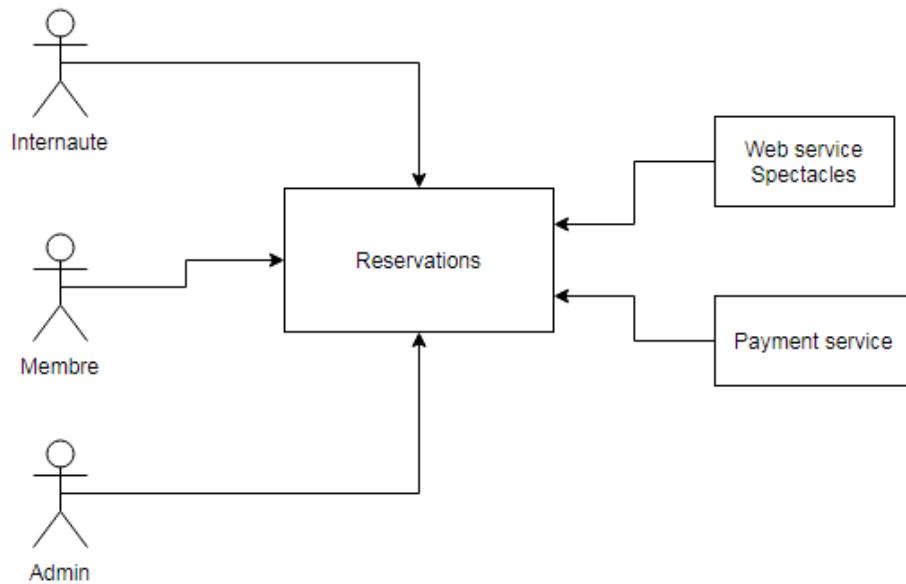
Une analyse de la concurrence ou une étude comparative des applications qui existent déjà sur le marché permet de comprendre comment ce genre d'application fonctionne. La nouvelle application pourra bénéficier de l'analyse de leurs avantages et inconvénients, et se positionner par rapport au marché, quitte à proposer des améliorations ou des innovations originales. Les critères d'analyse peuvent porter sur le processus de la fonctionnalité elle-même, mais aussi la navigation, la gestion des erreurs, l'accessibilité, l'ergonomie, les technologies utilisées, etc.

The figure displays five screenshots of theater ticketing websites side-by-side for comparison:

- Théâtre de Liège:** Shows a navigation bar with "Catalogue", "Vos sièges", "Votre panier", "Envoy et paiement", "Vos infos", and "Confirmation". Below is a show listing for "Bug" on March 22, 2018, at 20:00. A dropdown menu allows selecting the number of tickets (2). A section for "Choisissez la catégorie et le nombre d'éléments désirés" lists various ticket types with their prices and quantities.
- Théâtre de Poche:** Features a pink header with "THEATRE DE POCHE SAISON 2017/2018 BRUXELLES" and "RÉSERVATIONS". It shows a show listing for "Le printemps des barbares" on March 22, 2018, at 20:00, with a note that 22 places are available online. It includes fields for "Nom", "Prénom", "Téléphone", "Email", and "Code postal".
- Théâtre de la Toison d'Or:** Shows a navigation bar with "Code promo", "Catalogue", "Vos places", "Votre panier", "Envoy et paiement", and "Vos infos". It lists performances for "KING KONG THÉORIE" on various dates from March 22 to 31, 2018, with ticket availability status (COMPLET or RESTE 1 PLACE(S)).
- Théâtre du Parc:** Shows a "RÉCAPITULATIF DE VOTRE COMMANDE" page with two entries for "LES FEMMES... - 5,5 €" (5.50€ each, 1 ticket, Adulte) on March 22, 2018, at 20:15. It includes a "MONTANT" section showing the total command amount of 11€.
- Théâtre Royal du Parc:** Shows a "LES FEMMES... - 5,5 €" entry on March 22, 2018, at 20:15. It includes a "MONTANT" section showing the total command amount of 11€.

Analyse¹

Diagramme de contexte

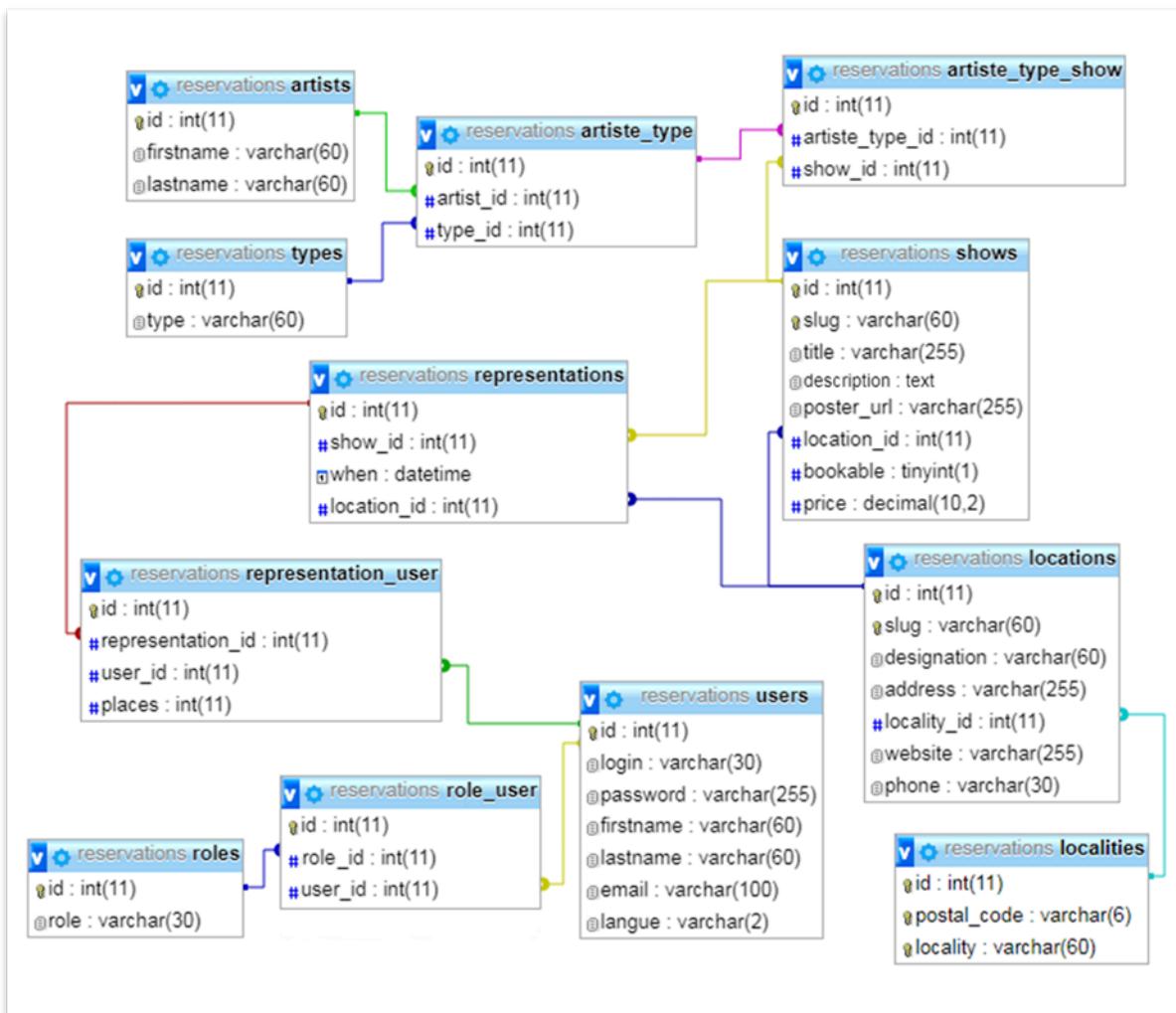


¹ Diagrammes réalisés sur <https://www.draw.io/>

Cas d'utilisation



Base de données



Contraintes et règle métier

L'étudiant devra justifier à travers la rédaction d'un **rappor t écrit** qu'il a réalisé le présent projet en veillant à respecter les contraintes suivantes :

- utiliser la **programmation orientée objet** (langages *Java*, *PHP* ou *Python*),
- exploiter un **framework backend** (*Spring Boot*, *Symfony*, *Laravel*, *CakePHP*, *Django*),
- exploiter un **framework ou une librairie frontend** (*Angular*, *VueJS* ou *jQuery*),
- intégrer des **services internes et tiers** (RESTful Web services),
- gérer les erreurs de programmation au moyen d'**outils ou de techniques de débogage** et y apporter une solution pertinente,
- utiliser à bon escient la **documentation** disponible,
- optimiser** le code, le cache et les échanges avec la base de données,
- Veiller à la **sécurité** des données et du système en général.

Modalités

Planning

L'itération 0 doit être effectuée en classe. Elle ne fait pas l'objet d'une remise.

Itération 1 – Installation du framework

Introduction

L'objectif de cette première itération consiste à effectuer les tâches suivantes :

1. Installer et configurer le *framework* ;
2. Utiliser un gestionnaire de dépendances pour organiser les librairies et composantes ;
3. Utiliser et configurer un environnement de développement intégré ;
4. Utiliser un logiciel de gestion des versions et configurer une solution d'hébergement ;
5. Utiliser un logiciel de gestion de projet et définir l'organigramme des tâches.

Modalités

Vous devez implémenter l'application en utilisant au choix un des frameworks côté serveur basé sur le langage PHP, Java ou Python : *Symfony*, *Laravel*, *CakePHP*, *Spring Boot* ou *Django*. Côté client, nous nous limiterons à JavaScript et la librairie *jQuery*. La partie statique HTML/CSS sera facilitée par le framework *Twitter Bootstrap*.



Planning

L'itération 1 doit être effectuée en classe. Elle doit faire l'objet d'une remise sur le dépôt de versioning en ligne.

Itération 2 – Prise en main du framework

Introduction

L'objectif de cette deuxième itération consiste à suivre un tutoriel simple qui permet de découvrir le workflow d'un développement avec framework. Il s'agit d'une mini-phase de formation.

Modalités

Vous devez suivre un tutoriel mettant en œuvre les activités suivantes :

1. Installation et configuration du framework,
2. Mapping relationnel simple d'une entité,
3. Implémentation d'un template simple de type liste des données,
4. Implémentation du contrôleur correspond et des actions CRUD de base (index, show, save, delete, update).

Références

Voici une liste non exhaustive de tutoriels classés par langage :

- PHP
 - Symfony
 - <https://symfony.com/blog/introducing-the-symfony-demo-application> (Demo Application)
 - Laravel
 - <https://laracasts.com/series/laravel-from-scratch-2017> (Laracasts « Laravel from scratch »)
 - Spark
 - <https://spark.laravel.com/docs/5.0/quickstart>
 - CakePhp
 - <https://book.cakephp.org/3.0/fr/tutorials-and-examples.html> (CMS)
- Java
 - SpringBoot
 - <https://spring.io/guides/gs/spring-boot/>
- Python
 - Django
 - <https://docs.djangoproject.com/fr/2.0/#index-first-steps> (prise en main)
 - <https://docs.djangoproject.com/fr/2.0/intro/> (tutoriel complet)

Planning

L'itération 2 doit être effectuée en classe. Elle doit faire l'objet d'une remise sur le dépôt de versioning en ligne.

Itération 3 – CRUD simple

Introduction

L'objectif de cette troisième itération consiste à ajouter les fonctionnalités suivantes :

1. Réaliser le mapping Objet-Relationnel d'une table sans clé étrangère de la base de données conformément au schéma de base de données et à la syntaxe de l'ORM proposé par le framework ;
 - a. Générer et personnaliser l'entité ou modèle ;
 - b. Générer et personnaliser les jeux de données (fixtures/seeder).
2. Implémenter, en utilisant la syntaxe du moteur de template, les templates de type
 - a. « *liste de produits* » (afficher tous les produits),
 - b. « *fiche produit* » (afficher un seul produit avec tous les détails),
 - c. « *formulaire nouveau produit* »,
 - d. « *formulaire modifier un produit* »,
 - e. « *formulaire supprimer un produit* » ;
3. Implémenter dans un contrôleur des méthodes chargées
 - a. d'extraire les produits de la table et les afficher dans le template de type liste ;
 - b. d'extraire un produit précis de la table et l'afficher dans le template de type fiche ;
 - c. de récupérer, valider les données de formulaire, créer un nouveau produit et l'insérer dans la table de la base de données,
 - d. de récupérer, valider les données de formulaire, modifier un produit précis et l'insérer dans la table de la base de données,
 - e. de supprimer un produit précis dans la table de la base de données ;
4. Définir les routes permettant d'exécuter les 4 opérations CRUD précédentes.

Analyse

Schéma de base de données

Diagramme de classes participantes



Routage des requêtes

Nom de la route	URL	Méthode	Description
artist.index	/artist/	GET	Affiche tous les artistes
artist.show	/artist/{id}	GET	Affiche l'artiste dont l'id est fourni dans l'URL
artist.create	/artist/create	GET	Affiche un formulaire vide
artist.store	/artist/	POST	Persiste les données du formulaire dans la BD
artist.edit	/artist/edit/{id}	GET	Affiche un formulaire rempli des données de

			l'artiste dont l'id est fourni dans l'URL
artist.update	/artist/	PUT	Persiste les données du formulaire dans la BD
artist.delete	/artist/{id}	DELETE	Supprime de la BD l'artiste dont l'id est fourni dans l'URL

Règles métier

- Interopérabilité
- Risques
- Sécurité

Dictionnaire de données

Artists

id	Entier long	Id de l'artiste	Clé primaire
firstname	Texte (60)	Prénom de l'artiste	
lastname	Texte (60)	Nom de l'artiste	

Show

id	Entier long	Id de l'artiste	Clé primaire
slug	Texte (60)	Label court d'identification	Titre slugifié automatiquement, Unique
title	Texte (255)	Titre du spectacle	

Contraintes et règle métier

Modalités

L'itération 3 doit faire l'objet d'une remise sur le dépôt de versioning en ligne.

Itération 4 –Authentification

Cette itération regroupe plusieurs problématiques liées au principe d'authentification :

- L'inscription des membres,
- La connexion et la déconnexion,
- La récupération du mot de passe,
- L'autorisation d'accès à une ressource ou une fonctionnalité en fonction du rôle d'un membre.

Contraintes et règle métier

Pour cette itération, il est demandé d'utiliser le composant adéquat du framework afin d'optimiser le développement en respectant les bonnes pratiques et les contraintes de sécurité.

Modalités

L'itération 4 doit faire l'objet d'une remise sur le dépôt de versioning en ligne.

Itération 5 – Mapping relationnel

Introduction

L'objectif de cette quatrième itération consiste à ajouter les fonctionnalités suivantes :

5. Réaliser le mapping Objet-Relationnel d'une table sans clé étrangère de la base de données conformément au schéma de base de données et à la syntaxe de l'ORM proposé par le framework ;
 - a. Générer et personnaliser l'entité ;
 - b. Générer et personnaliser les jeux de données (fixtures/seeder).
6. Implémenter au moins un template de type liste de données en utilisant la syntaxe du moteur de template proposé par le framework ;
7. Implémenter un contrôleur chargé d'extraire et d'afficher les données d'une table dans le template de type liste ;
8. Définir une route permettant l'affichage la page Web qui affiche la liste ;
9. Terminer le mapping Objet-Relationnel de la base de données en terminant par les tables possédant des clés étrangères.

Analyse

Diagramme de classes participantées

Règles métier

- Interopérabilité
- Risques
- Sécurité

Dictionnaire de données

Artists

id	Entier long	Id de l'artiste	Clé primaire
firstname	Texte (60)	Prénom de l'artiste	
lastname	Texte (60)	Nom de l'artiste	

Types

id	Entier long	Id du type	Clé primaire
type	Texte (60)	Fonction de l'artiste (métier)	

Artist_type

id	Entier long	Id de l'artiste	Clé primaire
artist_id	Entier long	Id de l'artiste	Clé étrangère
type_id	Entier long	Id du type d'artiste	Clé étrangère

Show

--

id	Entier long	Id de l'artiste	Clé primaire
location_id	Entier long	Id du lieu (locations)	Clé étrangère, Null autorisé
slug	Texte (60)	Label court d'identification	Titre slugifié automatiquement, Unique
title	Texte (255)	Titre du spectacle	
poster_url	Texte (255)	URL de l'affiche	Null autorisé
bookable	Boolean	Réserveable ou non	
price	Double	Prix du spectacle	
description	Texte long	Description du spectacle	
created_at	DATE	Date de création	Null autorisé

Localities

id	Entier long	Id de l'artiste	Clé primaire
postal_code	Texte (6)	Code postal	Unique
locality	Texte (60)	Désignation officielle de la localité	Unique

Locations

id	Entier long	Id de l'artiste	Clé primaire
locality_id	Entier long	Id de la localité (localities)	Clé étrangère, Null autorisé
slug	Texte (60)	Label court d'identification	Titre slugifié automatiquement, Unique
designation	Texte (60)	Désignation officielle du lieu	
address	Texte (255)	Adresse du lieu	Null autorisé
website	Texte (255)	Site Web du lieu	Null autorisé
phone	Texte (30)	Numéro de téléphone du lieu	Null autorisé

Collaborations (artist_type_show)

id	Entier long	Id de l'artiste	Clé primaire
artist_type_id	Entier long	Id de l'artiste associé à un type (fonction)	Clé étrangère
show_id	Entier long	Id du spectacle	Clé étrangère

TODO : Ajouter les autres dictionnaires de données

Contraintes et règle métier

Modalités

L'itération 3 doit faire l'objet d'une remise sur le dépôt de versioning en ligne.

Itération 6 – Utiliser un progiciel

Cette itération consiste à installer et utiliser un composant logiciel tierce qui permet de résoudre une fonctionnalité du cahier des charges. L'idée est de réduire le temps de développement en favorisant l'intégration et la réutilisation de code. Fonctionnalités concernées :

- Générer un flux RSS,

- Incorporer un back-office de gestion CRUD des données,
- Exporter des données en CSV ou PDF,
- Importer des données au format CSV,
- Générer une API Restful.

Contraintes et règle métier

Pour cette itération, il est demandé d'installer et utiliser un composant tierce afin d'optimiser le développement en respectant les bonnes pratiques et les contraintes de sécurité.

Modalités

L'itération 6 doit faire l'objet d'une remise sur le dépôt de versioning en ligne.

Itération 7 – Prototypage

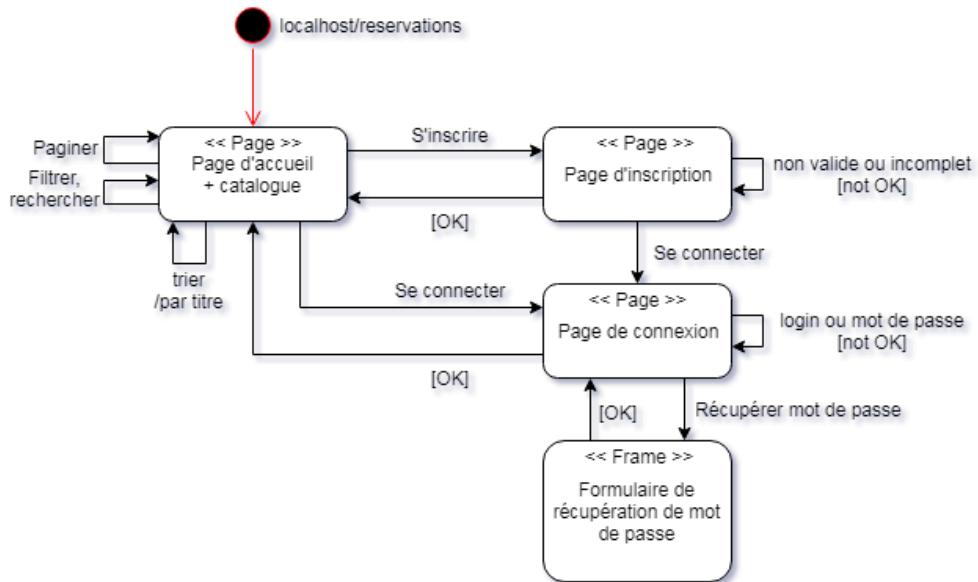
Implémenter les maquettes HTML au moyen de templates et d'un framework HTML/CSS (Bootstrap, Foundation...).

Analyse

Diagramme de navigation

Les diagrammes de navigation, issus des use cases, nous permettent d'identifier les écrans (interfaces graphiques, pages web) dont l'application a besoin.

Navigation S'inscrire + Se connecter



Mais il faut également analyser les use cases en identifiant dans les scénarios les composants visuels (titres, texte et images dynamiques, formulaires) et interactifs (boutons, hyperliens) nécessaires.

Diagrammes de cas d'utilisation

Use case S'inscrire

Scénario nominal

<ol style="list-style-type: none"> Un internaute clique sur le lien/bouton S'inscrire présent dans l'en-tête de toutes les pages. 	<ol style="list-style-type: none"> Le système affiche le formulaire d'inscription.
<ol style="list-style-type: none"> L'internaute encode : <ol style="list-style-type: none"> un login, un mot de passe (min. 6 caractères, au moins une majuscule et un caractère spécial), la confirmation du mot de passe, son prénom, 	<ol style="list-style-type: none"> De façon asynchrone, le système vérifie : <ol style="list-style-type: none"> La disponibilité du login, La validité du mot de passe, La correspondance des mots de passe, La disponibilité de l'email.

<p>e. son nom, f. son email, g. sa langue (choix dans une liste).</p>	<p>Un message de notification s'affiche pour chaque cas d'erreur et le bouton pour s'inscrire reste désactivé. Lorsque tous les champs sont validés, le bouton <i>S'inscrire</i> est activé.</p>
<p>5. L'internaute clique sur le bouton <i>S'inscrire</i> après avoir rempli tous les champs de façon adéquate.</p>	<p>6. Le système valide les données reçues côté serveur. 7. Le système enregistre les données dans la base de données (table <i>users</i>) : a. Le mot de passe est hashé, b. Le rôle <i>member</i> est assigné par défaut. 8. Le système envoie un email de confirmation à l'internaute. 9. Le système redirige l'internaute vers la page de connexion (login/mot de passe) avec un message de confirmation l'invitant à se connecter.</p>

Scénario alternatif

<p>5. L'internaute clique sur le bouton <i>S'inscrire</i> après avoir rempli tous les champs mais les données n'ont pas pu être validées côté client ou ont été altérées entretemps (piratage).</p>	<p>6. Le système invalide les données reçues côté serveur. 7. Le système redirige l'internaute vers le formulaire d'inscription avec un message d'erreur reprenant les validations qui ont échouées.</p>
---	--

Il ressort du diagramme de navigation et de ce scénario le besoin de créer les deux templates et le fragment de page suivants : un formulaire d'inscription, un formulaire de connexion et un menu d'authentification.

Vous pouvez utiliser des outils de développement rapide d'application (RAD) tels que les nombreux framework HTML/CSS/JS (*Bootstrap* ou *Foundation*) disponibles sur le Web pour personnaliser des templates et y intégrer les composants graphiques nécessaires, ainsi que des librairies afin de faciliter la programmation front-end (*jQuery*).

- <https://jquery.com/download/>
- <https://getbootstrap.com/docs/4.1/examples/>
- <https://bootsnipp.com> (page *snippets, forms ou builder*)

Use case Consulter le catalogue paginé des spectacles

Scénario nominal

1. Un internaute ou un membre clique sur le lien <i>Catalogue</i> présent dans le menu dans l'en-tête de toutes les pages.	2. Le système affiche le catalogue présentant la liste des spectacles triés par titre et paginés par 10.
3. L'internaute clique sur un des liens du menu de pagination pour changer de page.	4. Le système affiche les spectacles suivants correspondants à la page demandée.
5. L'internaute clique sur un des liens de tri (titre, lieu, réservable, prix).	6. Le système affiche la première page des spectacles triés selon le critère demandé.
7. L'internaute sélectionne un critère parmi les listes déroulantes des formulaires de filtre (lieu, réservable).	8. Le système affiche la première page des spectacles filtrés selon le critère demandé.
9. L'internaute entre un mot-clé dans le formulaire de recherche.	10. Le système affiche les spectacles correspondants au critère entré.
11. L'internaute clique sur un des liens du menu de pagination pour changer de page.	12. Le système affiche les spectacles suivants ou précédents correspondants à la page demandée en conservant les critères de tri, de filtre ou de recherche.

Il ressort de ce scénario le besoin de créer un template avec un tableau présentant la liste des spectacles.

Nous pouvons chercher ce composant graphique parmi les ceux de *Bootstrap* ou utiliser la librairie *Datatables* qui offre des fonctionnalités supplémentaires (tri, filtre, recherche et pagination) :

- <https://getbootstrap.com/docs/4.1/content/tables/>

PID – Projet Réservations

The screenshot shows the Bootstrap documentation page for version 4.1. The left sidebar has a search bar and links to 'Getting started', 'Layout', 'Content', 'Reboot', 'Typography', 'Code', 'Images', 'Tables', 'Figures', and 'Components'. The main content area is titled 'Striped rows' and shows a table with three rows, each with a different background color. The table has columns for '#', 'First', 'Last', and 'Handle'.

#	First	Last	Handle
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

- <https://datatables.net/>

The screenshot shows the DataTables.net homepage. It features a step-by-step guide on the left:

- 1 - Include these two files
- 2 - Call this single function
- 3 - You get a fully interactive table

Below the steps is a 'Full Getting Started Guide' button. On the right is a demonstration of a fully interactive table with 57 entries. The table has columns for Name, Position, Office, Age, and Start date. It includes search, sort, and pagination features.

Name	Position	Office	Age	Start date
Airi Satou	Accountant	Tokyo	33	2008/11/28
Angelica Ramos	Chief Executive Officer (CEO)	London	47	2009/10/09
Ashton Cox	Junior Technical Author	San Francisco	66	2009/01/12
Bradley Greer	Software Engineer	London	41	2012/10/13
Brenden Wagner	Software Engineer	San Francisco	28	2011/06/07
Brielle Williamson	Integration Specialist	New York	61	2012/12/02
Bruno Nash	Software Engineer	London	38	2011/05/03
Caesar Vance	Pre-Sales Support	New York	21	2011/12/12
Cara Stevens	Sales Assistant	New York	46	2011/12/06
Cedric Kelly	Senior Javascript Developer	Edinburgh	22	2012/03/29

Écrans et navigation

Mockups

Trame de page principale

The mockup illustrates the overall structure of the main page. It features a header with a back/forward button, a search bar, and user authentication links. Below the header is a large central content area containing a title, a search bar, and a placeholder for 'Prochaines représentations'. To the left is a vertical sidebar with a menu and a large placeholder for 'Liste des spectacles' which contains a table of three items. The footer includes copyright and usage conditions.

Page 1

http://localhost/reservations

Projet Réservations

Rechercher

Se connecter

S'inscrire

Menu

- Accueil
- Prochains spectacles
- Mon profil
- Mes réservations
- Administration

Prochaines représentations

Liste des spectacles

Titre	Auteurs	Lieu	Prix	Réservable
Le spectacle de la vie	John Doe	Palais des Arts	9,50 €	Oui
Du haut de mon perchoir	Smith Durand	Grand Palace	15,00 €	Non
A voir et à revoir	Bob Sull	Palais des Arts	10,50 €	Oui

<< Prev 1 2 3 4 5 6 7 8 9 10 Next >>

(c) Projet Réservations ICC / EPFC 2018

Conditions d'utilisation

Formulaire d'inscription

The screenshot shows a registration form titled "Inscription". The form includes fields for Email, Prénom, Nom, Login, Mot de passe, Confirmer, and Langue (with "Français" selected). There is also a "S'inscrire" button. The top navigation bar includes links for "Bootsnipp", "Bootstrap", "Search for snippets", "+New", "Community", "Tools", "Snippets", "Register", and "Login". The URL in the address bar is https://bootsnipp.com/snippets/featured/login-amp-signup-forms-in-panel.

Formulaire de connexion

The screenshot shows a login form titled "Connexion". It features fields for "login ou email" and "mot de passe", both preceded by icons (user and lock respectively). There is a "Retenir mon login" checkbox and a "Se connecter" button. Below the form, a link says "Vous n'avez pas de compte? Inscrivez-vous ici".

Menu d'authentification

Ce menu sera intégré dans l'en-tête de la page d'accueil.

Catalogue des spectacles

Ce catalogue sera intégré dans la page d'accueil.

← → ⌂ ⓘ localhost/reservationsSymfony/public/index.php/show

Liste des spectacles

Show 10 entries Search:

Titre	Auteur	Lieu	Représentations	Prix	Réservable
Ayiti	Bob Sull	Belfius Art Collection	2	9,50 €	Oui
Ceci N'est Pas Un Chanteur Belge		Belfius Art Collection	1	7,50 €	Non
Cible Mouvante	Marc Flynn	La Samaritaine	2	8,00 €	Oui

Showing 1 to 3 of 3 entries [Previous](#) [1](#) [Next](#)

Total: 3 spectacles.

Menu de navigation

Ce menu sera intégré dans l'en-tête ou dans la sidebar de toutes les pages (sauf administration).

← → ⌂ ⓘ localhost/reservationsSymfony/public/index.php/show?#

Projet Réservations

Rechercher  Se connecter  S'inscrire 

Menu

- Accueil 
- Prochains spectacles 
- Administration 
- Contact 

Liste des spectacles

Show 10 entries Search:

Titre	Auteur	Lieu	Représentations	Prix
 Ayiti	Bob Sull	Belfius Art Collection	2	9,50 €
 Ceci N'est Pas Un Chanteur Belge		Belfius Art Collection	1	7,50 €
 Cible Mouvante	Marc Flynn	La Samaritaine	2	8,00 €

Showing 1 to 3 of 3 entries [Previous](#) [1](#) [Next](#)

Total: 3 spectacles.

© Projet Réservations ICC / EPFC 2018
[Conditions d'utilisation](#)

Itération 8 –Formulaires et validation

Dans une application Web les formulaires permettent d'ajouter et de modifier des données.

Cependant, il faut veiller à la validité de nouvelles données fournies par l'utilisateur pour des raisons à la fois de sécurité mais aussi d'intégrité et de cohérence des données stockées.

Itération 9 – API RESTful

Un exemple d'application double (client + serveur) implémentant les principes d'une API Restful est disponible sur GitHub : <https://github.com/cedryc-ruth/caviste2018.git>

Veuillez utiliser une API publique du type OpenData. Généralement, il suffit de créer un compte pour avoir accès à l'API.

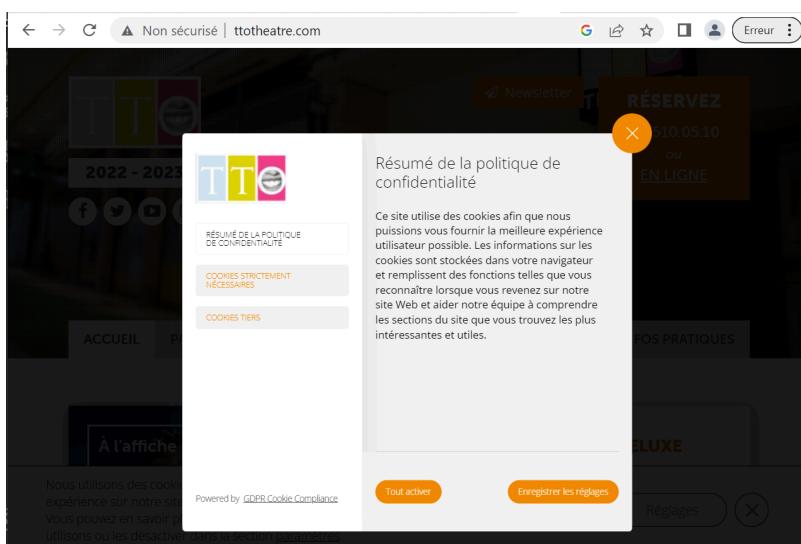
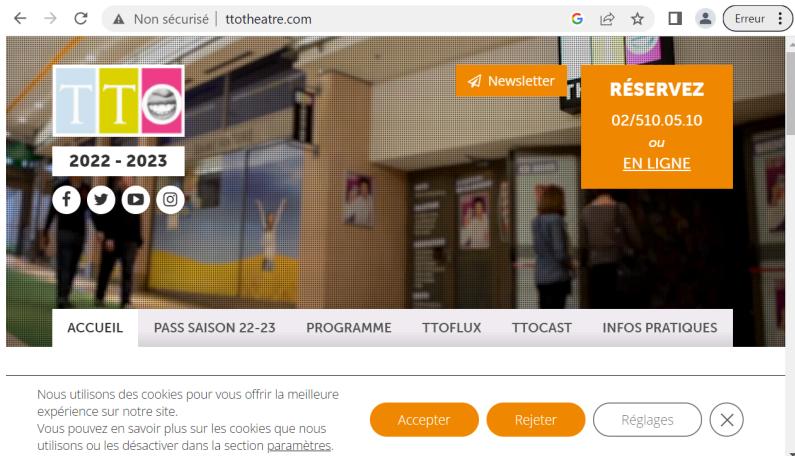
Exemples d'API :

- London Theater : <https://developer.londontheatredirect.com/Documentation>
- Théâtre de la ville de Paris: <https://api.theatredeparis.com/>
- Théâtre contemporain :
https://www.theatre-contemporain.net/images/partenaires/descriptif_opendata_api_cris-v4.pdf
- Agenda.Brussels :
<https://api.brussels/store/apis/info?name=agenda.brussels&version=0.0.1&provider=admin>

Itération 10 – Aspects légaux

Acceptation des cookies

- Bandeau d'acceptation
- Réglages / Paramétrage



PID – Projet Réservations

Cookies strictement nécessaires

Cette option doit être activée à tout moment afin que nous puissions enregistrer vos préférences pour les réglages de cookie.

Activé

Cookies tiers

Ce site utilise Google Analytics pour collecter des informations anonymes telles que le nombre de visiteurs du site et les pages les plus populaires.

Garder ce cookie activé nous aide à améliorer notre site Web.

Désactivé

Nous utilisons des cookies pour améliorer votre expérience sur notre site. Vous pouvez en savoir plus sur les types de cookies que nous utilisons ou les désactiver dans la section paramètres.

Powered by [GDPR Cookie Compliance](#)

[Tout activer](#) [Enregistrer les réglages](#) [Réglages](#) [X](#)

PID – Projet Réservations

Non sécurisé | ttotheatre.com/vie-privee/ Erreur

- droit de consentement (vous pouvez retirer à tout moment votre consentement pour le futur moyennant notification écrite à notre théâtre)

- droit d'opposition au marketing direct (envoi d'informations pour d'autres événements (newsletter,...)).

Pour exercer ces droits, il suffit d'envoyer un e-mail à l'adresse privacy@ttotheatre.be.

En cas de problème, vous pouvez également introduire une réclamation auprès de la Commission de la Protection de la Vie Privée commission@privacycommission.be.

Enfin, nous vous tiendrons informés en cas de modification substantielle de notre politique en matière de vie privée telle que décrite ci-dessus.

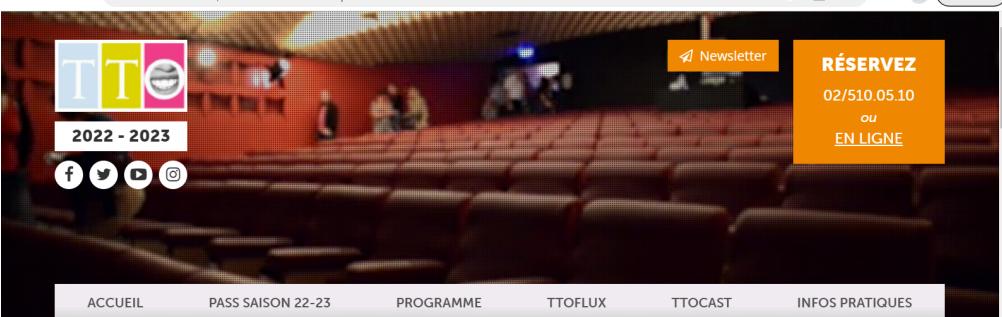
TTO THÉÂTRE
396 - 398 Galeries de la Toison d'Or
1050 Ixelles

RÉSERVATIONS
Par téléphone au 02 510 0510
ou [directement en ligne](#)

© Théâtre de la Toison d'Or 2023

Vie privée - Conditions générales made by [parfaitementweb.com](#)

Non sécurisé | ttotheatre.com/vie-privee/ Erreur



TTO
2022 - 2023

Newsletter

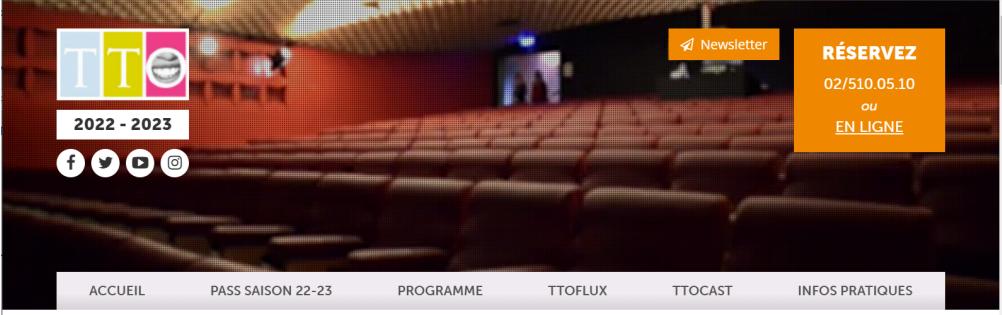
RÉSERVEZ
02/510.05.10
ou
EN LIGNE

ACCUEIL PASS SAISON 22-23 PROGRAMME TTOFLUX TTOCAST INFOS PRATIQUES

POLITIQUE DE VIE PRIVÉE

Nous mettons un point d'honneur à respecter la réglementation en matière de protection des données personnelles, à savoir principalement la loi belge du 8 décembre 1992 et le Règlement Général sur la Protection des Données 2016/679

Non sécurisé | ttotheatre.com/conditions-generales/ Erreur



TTO
2022 - 2023

Newsletter

RÉSERVEZ
02/510.05.10
ou
EN LIGNE

ACCUEIL PASS SAISON 22-23 PROGRAMME TTOFLUX TTOCAST INFOS PRATIQUES

CONDITIONS GÉNÉRALES

1. BILLETTERIE

- Les réservations se font en ligne 24/7 (via www.ttotheatre.com) ou par téléphone au 02/510.0510 du lundi au samedi de 14h à 18h ou par mail à l'adresse