

# Micro-Projet Web Cahier des Charges PC3R

Mars 2021

---

# Table des matières

<b>1</b>	<b>Sujet de l'application et ses principales fonctionnalités</b>	<b>3</b>
<b>2</b>	<b>Description de l'API SNCF</b>	<b>3</b>
<b>3</b>	<b>Fonctionnalités</b>	<b>3</b>
<b>4</b>	<b>Cas d'utilisation</b>	<b>4</b>
4.1	Premier cas . . . . .	4
4.2	Deuxième cas . . . . .	4
<b>5</b>	<b>Base de données</b>	<b>4</b>
5.1	Utilisateur . . . . .	4
5.2	Réservation . . . . .	4
5.3	Session . . . . .	4
5.4	tarifLine . . . . .	4
<b>6</b>	<b>Mise a jours des données</b>	<b>5</b>
<b>7</b>	<b>Description du serveur</b>	<b>5</b>
7.1	Composant Réservation . . . . .	5
7.2	Composant Recherche . . . . .	5
7.3	Composant Authentification . . . . .	5
<b>8</b>	<b>Description du client</b>	<b>5</b>
8.1	Page d'accueil . . . . .	5
8.2	Page d'authentification . . . . .	5
8.3	Page de réservation . . . . .	6
<b>9</b>	<b>Description des requêtes</b>	<b>6</b>
<b>10</b>	<b>Schéma du système</b>	<b>6</b>

# 1 Sujet de l'application et ses principales fonctionnalités

Notre application permet à un utilisateur de trouver toutes les destinations possibles d'un voyage en entrant un point de départ, en plus, elle permet d'avoir les tarifs de premières et de deuxième classes sur toutes les destinations possibles.

Après consultation des tarifs, l'utilisateur peut réserver un billet de train directement, en créant un compte ou en se connectant à l'aide d'identifiants.

## 2 Description de l'API SNCF

Pour implémenter notre application, nous avons choisi l'API « Tarifs Intercités de jour », l'API contient plusieurs informations : départs, destinations, tarifs... Nous présenterons le format des informations de chaque attribut :

- \* "origine" : "type" : "string", "title" : "Origine", "description" : " la ville de départ"
- \* "destination" : "type" : "string", "title" : "Destination ", "description" : " la ville destinataire"
- \* "prix d'appel 2nde" : "type" : "number", "title" : "Prix d'appel 2nde", "description" : ""
- \* "plein tarif 1ere" : "type" : "number", "title" : "Plein Tarif 1ère", "description" : "Plein tarif échangeable et remboursable 1ère classe,"
- \* "plein tarif 2nde" : "type" : "number", "title" : "Plein Tarif 2nde", "description" : "Plein tarif échangeable et remboursable 2nde classe,"

cf. <https://ressources.data.sncf.com/explore/dataset/tarifs-intercites-de-jour/api>

## 3 Fonctionnalités

- \* page d'accueil.
- \* recherche un trajet.
- \* consultation des trajets possibles.
- \* consultation des tarifs.
- \* création d'un compte utilisateur / authentification.
- \* réservation d'un billet

## 4 Cas d'utilisation

### 4.1 Premier cas

David arrive sur la page d'accueil qui lui demande de saisir un point de départ, le système affiche toutes les destinations possibles à partir de ce point, il sélectionne ensuite la destination souhaitée.

Le système lui renvoie alors le tarif de 1res et 2e classes. Il choisit le tarif correspondant. Il crée un compte sur lequel il s'identifie et valide la réservation.

### 4.2 Deuxième cas

Bob est un utilisateur, qui peut consulter les tarifs et les destinations possibles afin de choisir le moins cher "sans avoir réservé".

## 5 Base de données

### 5.1 Utilisateur

```
'nom' varchar(32) NOT NULL,  
'prenom' varchar(32) NOT NULL,  
'pseudo' varchar(32) NOT NULL,  
'email' varchar(32) NOT NULL,  
'motDePasse' blob NOT NULL,  
'age' int(11) NOT NULL,  
'telephone' varchar(10) NOT NULL
```

### 5.2 Réservation

```
'pseudo' varchar(32) NOT NULL,  
'idTrain' varchar(10) NOT NULL
```

### 5.3 Session

```
'pseudo' varchar(32) NOT NULL,  
'clef' varchar(32) NOT NULL,
```

### 5.4 tarifLine

```
'origin' varchar(32) NOT NULL,  
'plein_tarif_1ere' int (3) NOT NULL,  
'destination' varchar(32) NOT NULL,  
'plein_tarif_2nde' varchar(32) NOT NULL, '  
'prix_d_appel_2nde' int(2) NOT NULL
```

## 6 Mise a jours des données

Cette partie explique le fonctionnement logique de notre architecture, en effet, on doit mettre à jour notre base, et cela, dans plusieurs cas les voici :

- Modifications des tarifs.
- Ajouter des nouvelles lignes dans notre base de données.

## 7 Description du serveur

On choisit l'approche service qui consiste à visualiser le serveur comme une bibliothèque de services. En effet, le serveur reçoit les requêtes de la part du client et les traite en se servant de la base de données à partir des paramètres données par le client.

La liste des composants fourni par serveur est la suivante :

### 7.1 Composant Réservation

Ce composant, nous permet de réserver un billet à un certain tarif. En effet, il fonctionnera de la même façon qu'un code de promotion.

### 7.2 Composant Recherche

Ce composant, nous permet d'avoir toutes les destinations à partir d'une ville origine, et bien sûr de consulter les tarifs d'un trajet spécifique que ce soit en première ou deuxième classe.

### 7.3 Composant Authentification

Ce composant, permet à un utilisateur de créer un compte ou d'ouvrir une session en entrant ces identifiants pour réserver un billet.

## 8 Description du client

Le plan du site se compose en 3 pages :

### 8.1 Page d'accueil

La page contient une barre de recherche afin de trouver un trajet et pour cela en faisant appel à la méthode "RechercherUnTrajet" du serveur avec la méthode GET.

### 8.2 Page d'authentification

Cette page contient un formulaire pour l'authentification et un bouton "authentifier" qui va faire appel à la méthode "authentifier" du serveur avec la méthode POST d'un côté, et de l'autre un formulaire d'inscription dans le cas où l'utilisateur n'a pas réussi à s'authentifier.

### 8.3 Page de réservation

La page contient un champ "le nombre de billets" qui permet de choisir le nombre de réservations, mais sans indiquer ni la date ni le train à prendre, cette fonctionnalité nous permet juste d'avoir un code de promotion pour réserver ce dernier a un certain prix. Et à la fin un bouton "submit" qui fait appel à la méthode "reserver" du serveur par la méthode POST.

## 9 Description des requêtes

GET trajet -> récupérer la description du trajet qui correspond.

GET `http://Trainline.fr/recherche_trajet?trajet="Paris-Nantes" {  
id=456, trajet= "Paris-Nantes " , tarif_first_class="20£" , , tarif_second_class="15£" ga-  
reDépart="Montparnasse", prix_appel="15£" }`

POST login mdp -> authentifier un utilisateur, récupère son identifiant

PUT login mdp nom prénom age -> créer un nouvel utilisateur, récupérer son identi-  
fiant

DELETE id -> supprimer une reservation

DELETE `http://Trainline.fr/reservation ="3" {  
résultat : OK  
}`

## 10 Schéma du système

Le schéma est disponible en pièce jointe dans le dossier : "systeme.png".