

# **Test de performance Projet Annuel**

PA EXPRESS

## **2. Préambule**

Description de l'application qui va être testée :

### **Description globale du projet :**

Notre projet annuel (PA express) consiste à réaliser un site de gestion de lignes ferroviaires avec un système de réservation.

### **Objectif de l'application :**

L'application permet aux clients de réserver un voyage et d'accéder à leurs factures ainsi que leurs billets de train. Les contrôleurs peuvent scanner les billets de train des clients et avoir les informations de la réservation.

Partie back office, elle permet aux sociétés de train de créer des lignes, des trains, des wagons ainsi que des voyages. Ils peuvent visualiser sur un planning les différents voyages qui ont été créés et d'avoir un aperçu des détails du voyage (places occupées, places disponibles).

L'administrateur du site pourra quant à lui gérer les clients, les sociétés de train, les contrôleurs ainsi que les réservations.

### **Type d'utilisateurs prévus :**

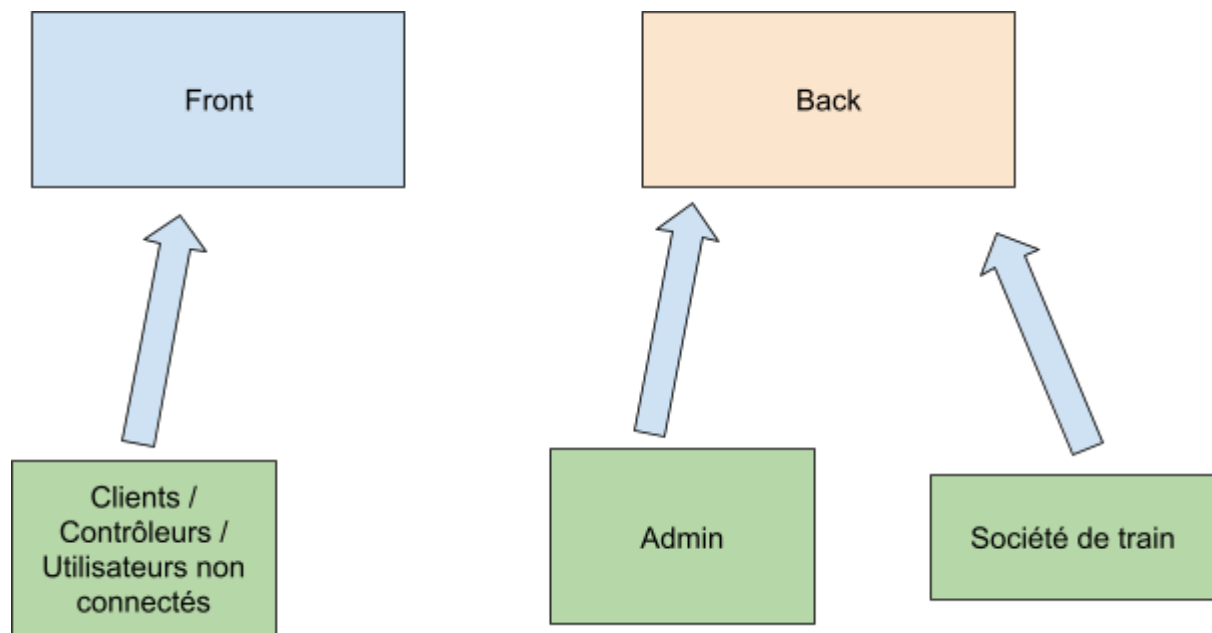
Notre application dispose de 4 acteurs :

- Clients
- Société de train
- Contrôleur
- Administrateur du site

### 3. Architecture de l'application

L'application est réalisée en Symfony, elle se connecte à une base de données postgres et tourne sur docker avec un serveur nginx.

La version en production tourne sur un docker dans un raspberry pi sous ubuntu.



## 4. Exigences du test

Des tests de performances sont nécessaires pour déterminer si l'application peut répondre correctement à une charge d'utilisateur connectés en même temps, notamment pendant la phase de réservation et de paiement.

<b>Business Transactions</b>	<b>User Load</b>	<b>Response Time</b>	<b>Transactions per hour</b>
Access login page	15	1	40
Access register page	15	1	40
Access payment page	15	1	40
Access home page	15	1	40

## 5. Environnement des tests

Test sur le CPU et la mémoire

## 6. Planification des test

## 7. Etapes des test

Step #	Business Process Name : Product Ordering
1	Home Page
2	Register
3	Login
4	Search train
5	Payment
6	Logout

## 8. Execution des tests

Test Run	Test Scenario Summary
Smoke Test	To validate the performance test scripts and monitors
Cycle 1 - Run 1	Load Test - 1 Hour test with peak load
Cycle 1 - Run 2	Repeat Load Test - 1 Hour test with peak load
Cycle 1 - Run 3	Stress Test - 1 Hour test with 150% of peak load
Cycle 2 - Run 1	Load Test - 1 Hour test with peak load
Cycle 2 - Run 2	Repeat Load Test - 1 Hour test with peak load
Cycle 2 - Run 3	Stress Test - 1 Hour test with 150% of peak load

Test Details	
<b>Purpose</b>	Peak hour transaction processing will be under examination to determine if the system can maintain response times under the highest anticipated load. This test is designed to collect performance metrics on transaction throughput, response times, and system resource utilization, in comparison to Performance requirements.
<b>No. of Tests</b>	4 (2 tests per cycle)
<b>Duration</b>	Ramp-up: X - Steady State: X - Ramp-down: X
<b>Scripts</b>	1. XXXX - 2. XXXX
<b>Scenario Name</b>	Load Test Scenario
<b>User Load / Volume</b>	500 Vusers (Threads) Load
<b>Entry Criteria</b>	<ol style="list-style-type: none"> <li>1. The code should be stable and functionally verified</li> <li>2. Test Environment should be stable and ready to use</li> <li>3. Test Data should be available</li> <li>4. All the NFRs should be agreed with the project</li> <li>5. Test scripts should be ready to use</li> </ol>
<b>Validation Criteria</b>	<ol style="list-style-type: none"> <li>1. The mean of the response time should be below 1.5 sec</li> <li>2. The error rate should be below 5%</li> </ol>