

Opération Mise en Production — ScèneLibre

1. Analyse de risque

Risque	Origine	Ce qui peut se passer	Gravité	Probabilité	Priorité
Double vente d'une place	Règle métier n°6 + forte concurrence	Deux clients achètent la même dernière place	5	4	20
Site indisponible pendant un pic	40 000 personnes en 10 min + architecture non testée	Le site ou la base tombe pendant une grosse mise en vente	5	4	20
Secrets de paiement compromis	Mots de passe présents dans Git	Un attaquant récupère les identifiants du prestataire bancaire	5	4	20
Paiement incohérent	Service de paiement externe	Client débité sans billet, double débit ou montant incorrect	5	3	15
Défaillance d'un service externe	Paiement, e-mail, génération PDF	Achat validé mais billet non envoyé ou PDF non généré	4	4	16
Perte ou fuite de données personnelles	Base de données + justificatifs téléversés	Données clients exposées ou perdues	5	3	15
Mauvais calcul métier	TVA 5,5 %, commission 6 %, réduction 30 %	Mauvais prix, mauvaise facture ou mauvais remboursement	4	3	12
Environnement unique	Une seule machine pour dev et démo	Un test ou un bug casse la démonstration ou les données	4	4	16

Risque numéro un

Le risque principal reste **la vente incorrecte**, en particulier la double vente d'une place.

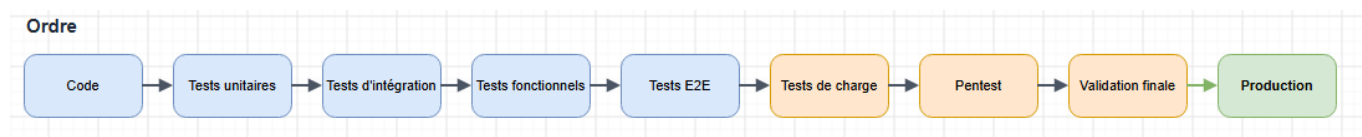
Mais le risque de **saturation de l'architecture** est au même niveau de priorité car personne n'a jamais fait de test de charge alors que 40 000 connexions sont attendues en 10 minutes.

2. Stratégie de test

Nous concentrons les tests sur les 4 risques les plus critiques.

Risque	Test	Qui	Quand	Critère de réussite
Double vente	Intégration + concurrence	Dev + qualité	Avant chaque livraison	Une seule vente possible pour une même place
Pic de charge	Charge + stress	Qualité	Avant production	Le site tient la charge prévue sans erreurs critiques
Secrets / sécurité	Pentest + revue configuration	Prestataire + dev	Avant ouverture	Aucune faille critique, aucun secret exposé
Paiement / services externes	Intégration + E2E	Dev + qualité	Avant livraison	Paiement, billet, PDF et e-mail cohérents

Ordre:



Ce qui bloque la mise en ligne

La mise en production est bloquée si:

- une place peut être vendue deux fois,
- le site ne tient pas la charge prévue,
- un paiement peut être doublé ou perdu,
- les secrets restent dans Git,
- une faille critique est présente,
- le parcours achat → paiement → billet ne fonctionne pas.

Un petit problème d'affichage ne bloque pas forcément la production.

3. Cas de test métier

Tarif réduit

Règle: moins de 26 ans avec justificatif valide.

Cas	Résultat
25 ans + justificatif valide	Tarif réduit -30 %
26 ans	Tarif normal
Moins de 26 ans sans justificatif	Tarif normal
Moins de 26 ans avec justificatif invalide	Tarif normal

Valeurs limites principales: **25 ans et 26 ans**.

Limite de 6 places

Quantité	Résultat
5	Accepté
6	Accepté
7	Refusé

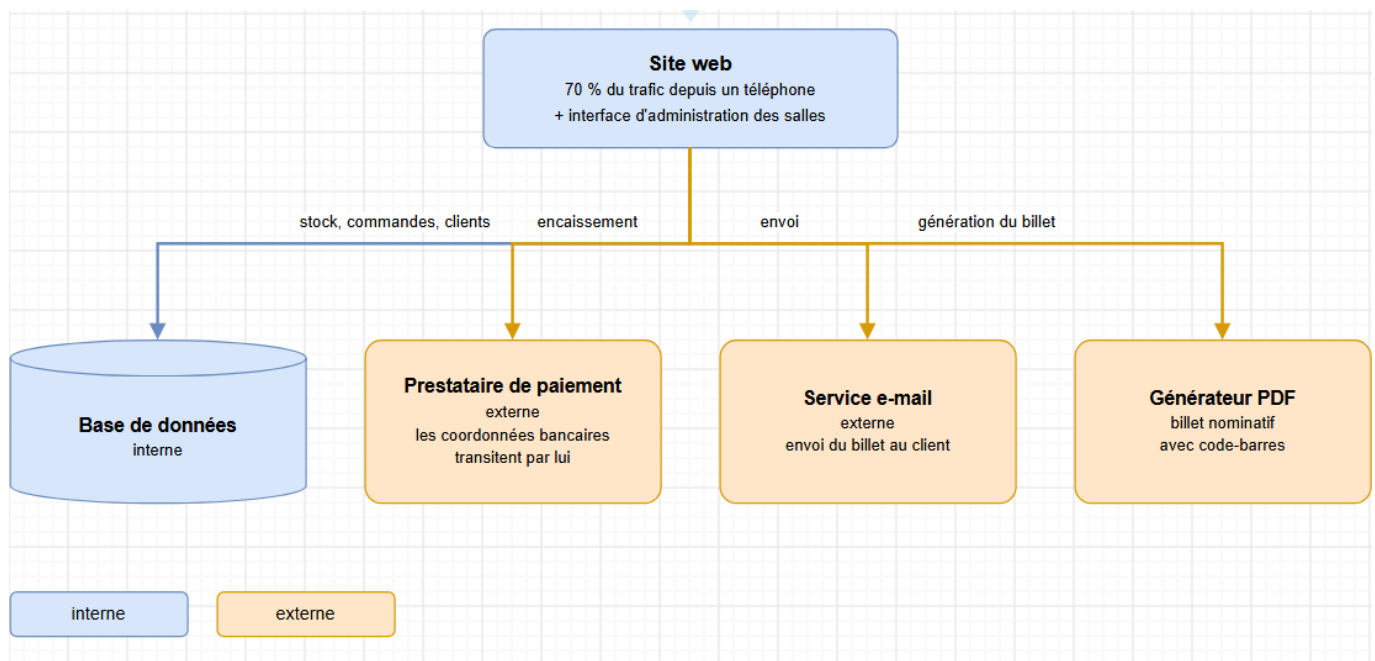
Il faut aussi vérifier qu'un client ne puisse pas contourner la règle avec plusieurs commandes.

Cas particuliers importants

- client qui fête ses 26 ans aujourd'hui,
- achat à 1 h 59 avant le spectacle,
- panier commencé avant les 2 heures mais payé après,
- deux clients sur la dernière place,
- double clic sur « Payer »,
- paiement accepté mais réponse du prestataire perdue,
- service e-mail indisponible,
- génération PDF en erreur,
- justificatif corrompu,
- client dans un autre fuseau horaire.

4. Risques liés à l'architecture

L'application dépend de plusieurs composants:



Chaque dépendance peut provoquer une panne.

Exemples:

- base indisponible → plus de recherche ni de vente,
- paiement indisponible → aucune commande possible,
- e-mail indisponible → billet non reçu,
- PDF indisponible → billet impossible à générer,
- surcharge du site → clients bloqués,
- surcharge de la base → commandes lentes ou incohérentes.

Il faut donc tester non seulement l'application elle-même, mais aussi les échanges entre les différents services.

5. Environnements

L'existant actuel est insuffisant: une seule machine sert au développement et à la démonstration.

Nous proposons:

Environnement	Rôle
Développement	Développement et corrections
Test	Tests fonctionnels et intégration
Préproduction	Validation finale et pentest
Charge	Tests temporaires de charge et stress

L'environnement de charge peut être créé seulement pendant les tests puis supprimé pour limiter le coût.

6. Jeu de données

Les 12 spectacles factices sont insuffisants.

Nous générons par exemple:

- 500 spectacles,
- 50 000 clients fictifs,
- plusieurs dizaines de milliers de commandes.

Nous ajoutons volontairement:

- spectacles complets,
- dernière place disponible,
- catégories presque pleines,
- clients de 25 et 26 ans,
- justificatifs valides et invalides,
- paiements refusés,
- spectacles annulés,
- remboursements.

Les données doivent être fictives ou anonymisées, jamais une copie directe des vrais clients.

Une graine fixe permet de recréer exactement les mêmes données et de rejouer un test.

7. Sécurité

Urgence 1: mots de passe dans Git

Un attaquant peut les récupérer dans le dépôt ou son historique.

Correction immédiate:

- révoquer les identifiants,
- en créer de nouveaux,
- supprimer les secrets du dépôt,
- utiliser un gestionnaire de secrets.

Urgence 2: environnement unique

Un test ou une erreur peut casser la démo ou les données.

Correction:

- séparer développement, test et production.

Urgence 3: aucune validation de sécurité

L'application manipule des paiements, des justificatifs et des données personnelles.

Correction:

- contrôle des accès,
 - tests de sécurité,
 - pentest avant ouverture.
-

8. Décision finale

Aujourd'hui: **NO GO**.

Dans trois semaines: **GO uniquement si**:

- aucune double vente,
- charge validée,
- paiement fiable,
- parcours achat → billet fonctionnel,
- secrets retirés de Git,
- aucune faille critique,
- règles métier principales validées.

Le but n'est pas de tout tester.

Le but est de sécuriser en priorité:

- l'argent,
- les données,
- la disponibilité du site,
- les règles métier critiques,
- les dépendances techniques indispensables.