

Sommaire

- Introduction : Rappel contexte, revue des sprints précédents et objectifs du dernier sprint
- Organisation
- Présentation de quelques tâches de travail
- Demonstration MyTheatre
- Perspectives d'évolutions si sprint 4

Introduction : présentation des membres du groupe

Contributeurs	Rôles	
ABL (Antoine)	Developpeur, Intégration	
MBN (Marjorie)	Developpeuse, Responsable qualité	
NKE (Nourredine)	Developpeur, Responsable qualité	
YBI (Youssef)	Communiquant Developpeur	

Contexte : description du problème

- Le client, directeur de théâtre, souhaite informatiser son système de réservation de places pour les spectacles.
- L'application doit pouvoir présenter une liste de représentations, la réservation de places doit pouvoir se faire en ligne.
- Un système de récupération de billets doit être mis en place.
- Contexte de travail se base principalement sur deux scénarios d'utilisations :
 - **S1 et S1bis: Consultation et pré-réservation**
 - **S2 : Achat de tickets**

Introduction : Rappel objectifs sprints précédents

Sprint 1

- Conception de la consultation des représentations (scenario 1) et couverture partielle du scenario 2 (achat de places)
- Développement sommaire de la partie “consultation”

Sprint 2 :

- Finalisation conception, couverture totale scénario 2
- Conception IHM abstraite relative au scénario 2 (à faire valider)
- Développement consultation, ajout de filtres
- Mise en place d'une charte graphique pour permettre la suite du développement

Introduction : Objectifs Sprint 3

Répondre au S2 côté développement \Rightarrow Gérer les achats :

- Faire valider l'IHM abstrait
- Choix spectacle
- Plan de salle
- Gestion panier
- Valider réservation
- Générer billet

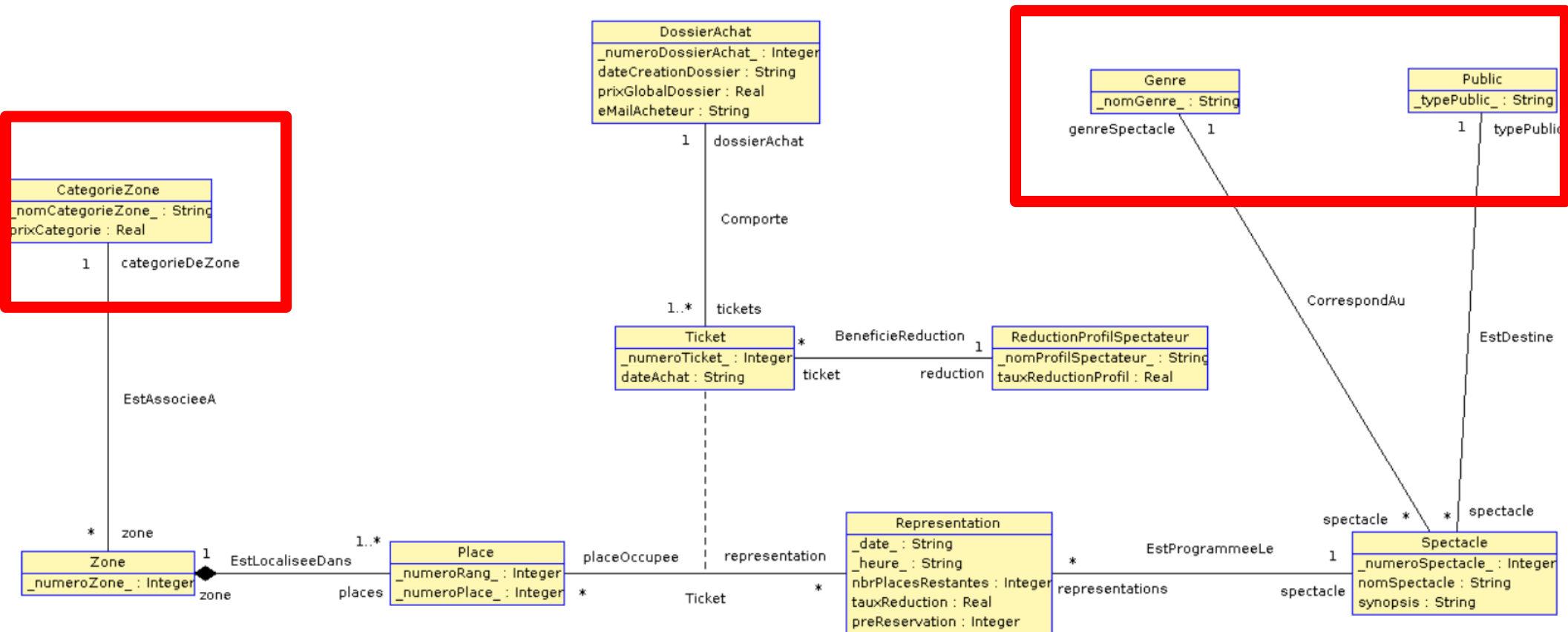
Extension : **Option pré-réservation**

- Gérer l'option sur les représentations
- Gérer le timer entre la date/heure de création de dossier et la date/heure de génération du ticket
- Gérer le panier Sauvegardé

Organisation

- Répartition des tâches de GL et Java
- Travail sur différentes branches. Une **branche** = une fonctionnalité. Puis intégration dans master.
- Découverte vision **agile** pour le suivi du projet :
Définition des tâches sous **github**. Assignment quotidienne lors des **stand up**.
- Système de **priorisation** des tâches revues en cours de projet suites aux remarques.
- Vision agile complétée par une vision plus « classique ».
=> Définition d'un planning sous **Gantt**

Diagramme de classes



Définition Dossier Achats au cours de son cycle de vie. Prise en compte du S1bis et S2.

3 états pour un dossier achat :

- Etat metastable :

Dossier existant en base, mais pas encore pré-réservé ni acheté.

Le dossier n'a pas de date.

- Etat pré-réservé :

pré-réservation effectuée par l'utilisateur.

Le dossier possède une date (date de pré-réservation)

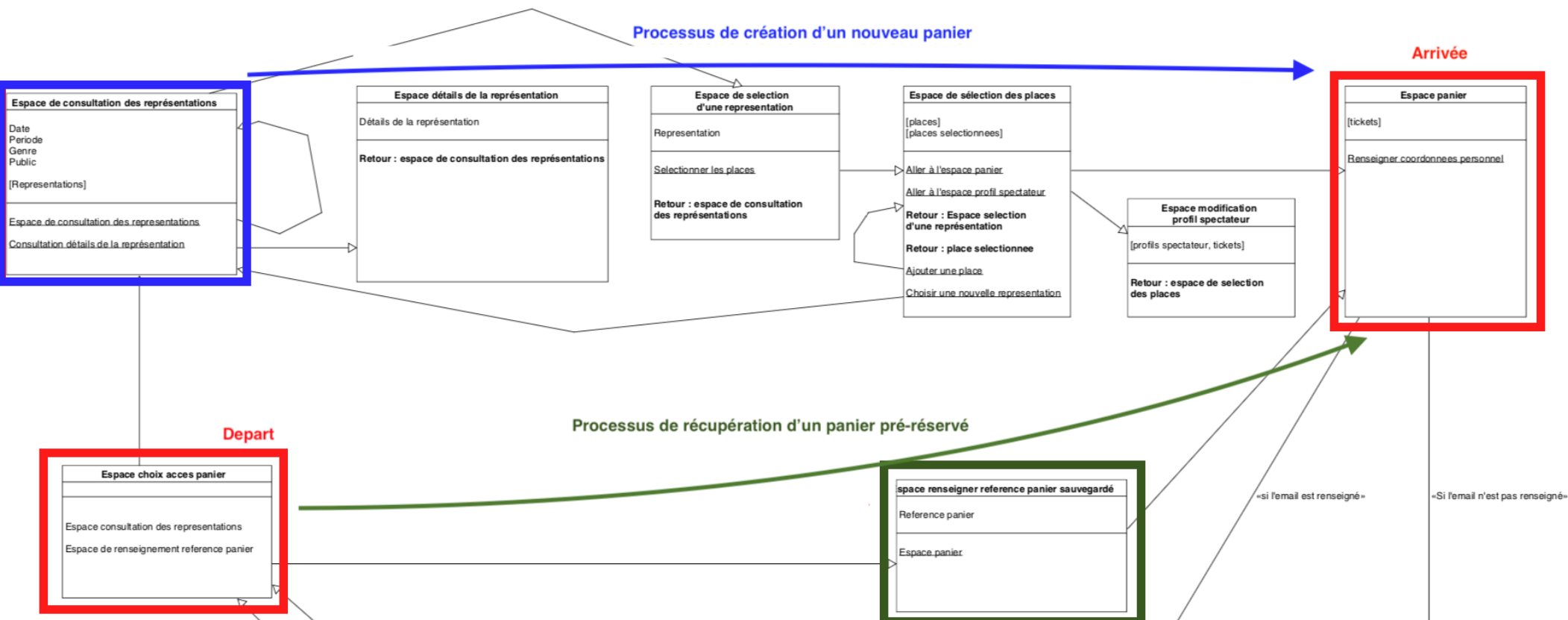
- Etat acheté:

L'utilisateur est allé au bout du processus d'achat. Les tickets associés sont envoyés.

Le dossier possède une date (date d'achat).

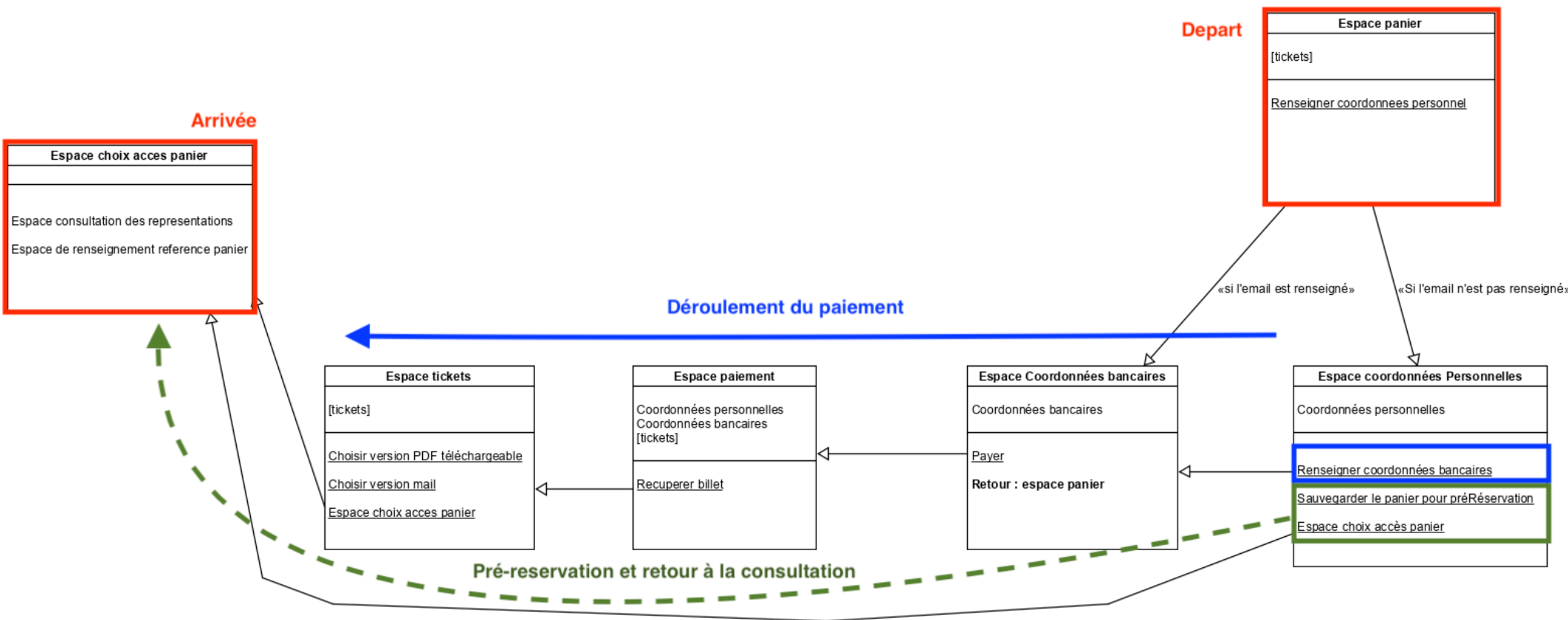
IHM abstrait : Les fondations de l'application

Comment accéder à son panier?



IHM abstrait : Les fondations de l'application

Pré-réserve ou achat



Developpement : Notions et librairies utilisées

Front End: Integration de bibliothèques

- Bootstrap
- JQuery
- JQuery-seat-chart

Back End:

- Modele Vue Controleur
- Ajax
- Gestion de session utilisateur

Gestion d'une session utilisateur

Informations **en session** :

- En permanence : Numéro de dossier
- Une fois le choix de places terminé : prix du dossier, liste de tickets, Email utilisateur

=> Accès plus rapide aux données de l'utilisateur.

Une session est supprimée lorsque l'utilisateur termine un processus de pré-réservation ou un processus d'achat, ou encore procède à l'annulation d'un panier.

Qualité de Production

- Ecriture de **test unitaires** pour confirmer le bon fonctionnement des méthodes codées (par manque de temps les tests unitaires ne couvrent pas toutes les méthodes).
- Ecriture de **tests fonctionnels** pour vérifier le bon fonctionnement de l'application

N°	Action testée	Résultat attendu	Résultat obtenu	Status test (pass or fail)	#defect
1	Voir le tableau des représentations après l'ouverture de <u>MyTheatre</u>	Voir le tableau complet des représentations programmées	On voit toutes les représentations programmées	Pass	
2	Consultation détails représentation	Affichage à droite du tableau des présentation de l'intégralité du synopsis	Visualisation du synopsis entier	Pass	

- Application proposée à l'essai à des personnes externes à notre groupe de projet.
==> Retours sur **l'utilisabilité** de l'application.

Demo Vfinale

“Jeu” de 3 scénarii :

1) Consultation et pré-réservation :

Utilisateur “curieux” qui cherche une idée de pièce et qui procède à une pré-réservation.

2) Achat direct de places pour une représentation:

Utilisateur conscient de ses besoins qui réalise directement son achat.

3) Récupération d'un dossier d'achat pré-réservé:

Que se passe-t-il si le dossier est déjà payé ?

Que se passe-t-il si le dossier est inexistant ?

Bilan projet : sprints 1, 2 et 3

- La conception et le développement permettent de couvrir les **scenarii 1, 1bis et 2**.
- **Anticipation** dans la conception sur une future possibilité dans la gestion par un administrateur (ex : genres et type public ==> sélecteurs dans la consultation contruits dynamiquement)

Retrospective :

- Sprint 1 (et partie du 2) focalisés sur la conception, laisse le champ libre au sprint 3 sans revenir sur nos décisions => **conception et développement aligné**
- Stand-Up quotidien + Audit hebdomadaire permet de **garder un cap cohérent et maintenir une synergie de groupe**
- Amélioration notable dans l'organisation entre Sprint 1 et 3

Perspectives sprint 4

1. Prise en compte des retours de l'audit N°3

2. Amélioration/finalisation concernant des fonctions existantes :

- Gérer la **contrainte des 70 places** pour l'achat en ligne
- Ajouter une **sécurité** pour la récupération d'un dossier achat
- Ajouter une fonction de “**nettoyage**” dans la base :
 - dossier “métastable”
 - Gérer la **contrainte des 24h** pour un dossier pré-réservé
- **Gérer la concurrence** lors d'achats simultanés
- Uniformiser le CSS, adaptation écran mobiles (**responsive**)
- Vérifier l'alignement avec la **RGPD** (stockage adresse mail)

3. Ajout d'un système d'administration (**S3**) : conception > Développement > Evaluation