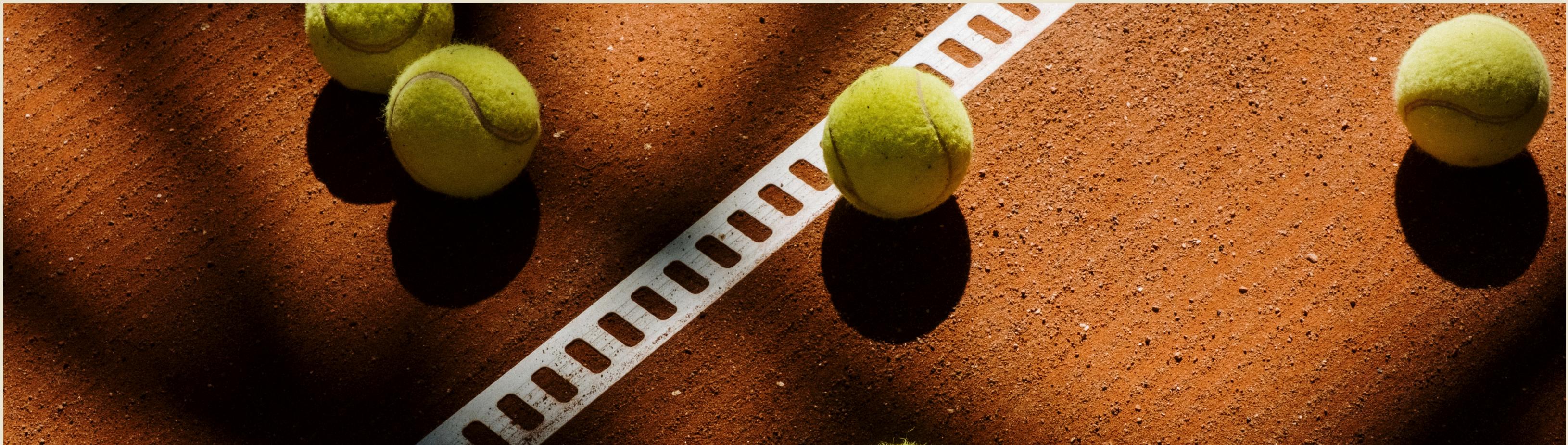


PROJET TUTORÉ >>>> >>>>

KETZINGER - MICHELI
PEDRON - RAOUF
IL-2
FIN D'ITÉRATION 4



WELL TENNIS CLUB

MICHAËL BORNE - OLIVIER PERRIN



UNIVERSITÉ
DE LORRAINE

IUT nancy Charlemagne
Informatique

SOMMAIRE



ITÉRATION 4

Présentation des objectifs atteints

Problèmes rencontrés

Objectifs pour les prochaines itérations

RETOUR SUR LE PROJET

Les éléments que nous avons respectés

Tout ce qui a changé

Nos retours sur notre organisation

Problèmes potentiels

Déroulement de l'application



OBJECTIFS ATTEINTS

ITÉRATION 4



SPRING

Modification de la base de données pour concorder avec Timefold

Création de requête personnalisée

Début de la réalisation de tests

- 100% de code coverage sur les services

Création de méthode de conversion entre les entités Spring et le domaine Timefold

TIMEFOLD

Fin de l'intégration (sauf une)
Planification des adultes > 95%
Planification globale > 85%

- retrait des cours entiers

Coverage des tests du solver à 81%

FRONT-END

Connexion et déconnexion des administrateurs.
Sécurisation des adresses.

Import de données au format Excel.

Affichage mobile.

Possibilité de s'inscrire et affichage chez l'administrateur.

PROBLÈMES RÉSOLUS

Amélioration significative du score de TimeFold.

Notamment en améliorant la valuation des pénalités.

Amélioration des messages d'erreurs du serveur.

PROBLÈMES EXISTANTS

La contrainte d'heure hebdomadaire n'est pas mise en place.
Les CORS sont configurés pour autoriser tout le monde.
Suppression des éléments à partir du front.
Problèmes d'affichage de différents éléments.



PROBLÈMES RENCONTRÉS

Configuration des CORS.

Compréhension et validation des Tokens, côté client.

Les contraintes Timefold entraient en conflit entre elles.

PLAN DES ITÉRATION

ITÉRATION 5

- Reset du mot de passe par email
- Drag & Drop
- Lancement de Timefold à partir des sessions
- Modification et sauvegarde des sessions depuis le site
- Correction des problèmes visuels
- Afficher les contraintes de séance
- Modifier les contraintes
- Joindre les modifications front-end avec Timefold
- Créer l'affichage pour les entraîneurs



ITÉRATION 6

- Outils de passage à une nouvelle année
- Envoie des plannings par email
- Faire des requêtes plus spécifiques
- Ajouter les différents filtres (données, EDT,...)
- Permettre le signalement de son absence
- Terminer le responsive du site
- Mettre en place l'affichage PDF

ITÉRATION 7

Fin du projet :

- création des dockers d'installation
- mise en place sur le serveur de Mr. BORNE
- rédaction du document de fin



LES ÉLÉMENTS QUE NOUS AVONS RESPECTÉS

Design global de l'application

La maquette Figma créée à la base a permis de gagner un temps considérable.

Utilisation de Timefold plutôt qu'une implémentation personnelle :

Domaine d'une taille de : $6,73 \times 10^{993}$

- > 248 joueurs (318 cours) et 10 entraîneurs avec 318 créneaux possibles
- 13 contraintes qui influent les unes sur les autres.
- > 5 contraintes pour les joueurs
- 3 contraintes pour les sessions
- 5 contraintes pour les entraîneurs

The screenshot displays several components of the application:

- Page d'administration:** Shows a grid of training slots for Monday and Tuesday across three courts. It includes columns for coaches and players, and rows for time slots (17:00, 18:30, etc.). A sidebar lists players and coaches with their respective age ranges (e.g., 0-17, 17-18).
- Page de connexion:** A login screen for the "Well Tennis Club" featuring a tennis court graphic and fields for "Identifiant" and "Mot de passe".
- Data entry screens:** Three separate windows for defining constraints:
 - constraint:** Shows age ranges (3-4, 3-6, 3-8, 3-10) and their combinations.
 - availability:** Shows player availability by day and time (e.g., Monday 09:00-12:30).
 - not-placed-explication:** A tooltip explaining the "not-placed" constraint, which allows modifying the entire set of data for a specific player.
- Import section:** On the right, there are sections for importing data in XLS or CSV formats, and a "Télécharger vos données" button.

Conservation des fonctionnalités initiales demandées.

Respect de la plupart des technologies
Conservation de Timefold, Spring, Vue.js et Tailwind.

Suivi global de notre étude préalable et de nos objectifs.

LES CHANGEMENTS



NOTRE BASE DE DONNÉE

Besoin de relations plus structurées entre les entités (joueurs, entraîneurs, disponibilités, terrains, sessions).

L'ABSENCE D'UN PROFESSEUR

Avec les différents retards, cette fonctionnalité bonus est mise en suspend.

FUSION DE SPRING ET TIMEFOLD

Les deux devaient être combinés.
À la place des méthodes de conversion.

INSCRIPTIONS ENTRAINEURS

Désormais, l'inscription se fait directement par un administrateur

NOS RETOURS SUR NOTRE ORGANISATION

L'étude préalable n'était pas assez poussée techniquelement.

Absence de choix préalable sur les différents URL, paramètres et réponse pour le serveur.

Manque de communication entre le backend et le frontend

Mauvaise organisation initiale :

- Actuellement : 1 Timefold, 1 Backend, 2 Frontend.
- Au lieu de : 1 Timefold, 2 Backend, 1 Frontend, puis 4 Frontend.

L'optimisation du Backend pour le connecter avec Timefold a entraîné un travail en plus du côté Frontend.



PROBLÈMES POTENTIELS NON RENCONTRÉS



SCALABILITÉ DE LA BASE DE DONNÉES

Non testée sur un grand volume de données, des optimisations pourraient être nécessaires.

OPTIMISATION DES PERFORMANCES

Pas d'utilisation de caching ou de lazy loading pour accélérer le chargement des données.

SÉCURITÉ AVANCÉE

Protection contre les attaques XSS, CSRF et validation renforcée des entrées utilisateur.
Actuellement HTTP et non HTTPS

**MERCI
POUR
VOTRE ÉCOUTE**

