

FIFA MANAGER 2020

Cahier des charges



Table des matières

CONTEXTE	3
CADRE	3
PROJET	3
ANALYSE MÉTIER.....	3
DESCRIPTION DE LA SOLUTION ENVISAGÉE	3
<i>Composition du projet</i>	3
LES INTERVENANTS	3
FONCTIONS ATTENDUES	4
<i>Exigences fonctionnelles</i>	4
Fonctionnalités obligatoires dans le Backend	4
Fonctionnalités obligatoires dans MatchManagement.....	4
<i>Exigences non fonctionnelles</i>	4
CONTRAINTES BUSINESS.....	4
<i>Point de vue technique</i>	4
<i>Composition du cahier des charges</i>	4
<i>Architecture du projet, à respecter</i>	5
ANALYSE FONCTIONNELLE.....	6
DIAGRAMME USE CASES (LES PRINCIPAUX)	6
DESCRIPTIONS TEXTUELLES DES DIFFÉRENTS USE CASES	6
<i>BackEnd</i>	6
Accueil du programme BackEnd	6
EF-BE-001 — Inscrire des joueurs dans une équipe à un championnat	7
EF-BE-002 — Transférer des joueurs durant les intersaisons	8
EF-BE-003 — Visualiser un calendrier de championnat	9
EF-BE-004 — Modifier un calendrier de championnat	10
EF-BE-005 — Générer automatiquement des calendriers de matchs.....	11
EF-BE-006 — Visualiser les résultats d'un championnat	12
EF-BE-007 — Modifier les résultats d'un championnat	13
EF-BE-008 — Visualiser les cartes des joueurs, des championnats	14
EF-BE-009 — Visualiser l'historique des joueurs.....	15
EF-BE-010 — Visualiser le classement d'un championnat	16
<i>MatchManager</i>	16
Accueil du programme MatchManager	16
EF-MM-001 — Inscrire les joueurs pour un match + contrôles (vérifier si l'inscription est possible)	17
EF-MM-002 — Encoder les résultats.....	19
EF-MM-003 — Encoder les cartes	20
EF-MM-004 — Encoder les goals	21
CONTRAINTES FONCTIONNELLES.....	22
RÈGLES D'ACCÈS ET AUTORISATION.....	22
RÈGLES DE STRUCTURE.....	22
<i>Championnat</i>	22
<i>Equipes</i>	22
<i>Constitution_Championnat</i>	22
<i>Constitution_Equipes</i>	22
<i>Saisons</i>	23
<i>Joueurs</i>	23
<i>Inscription_Match</i>	23
<i>Events</i>	23
<i>Type_Events</i>	24
<i>Suspensions</i>	24
<i>Match</i>	24
<i>Types_Resultats</i>	24

RÈGLES DE VALIDATION	25
<i>Constitution_Equipes</i>	25
<i>Inscription_Match</i>	25
<i>Events</i>	26
<i>Suspensions</i>	26
<i>Match</i>	27
RÈGLES DE CALCUL	28
<i>Durée d'une saison</i>	28
<i>Calcul du classement</i>	28
<i>Calcul des types de résultats</i>	28
<i>Calcul des types Events</i>	28
DESCRIPTION DES ENTITÉS	29
SCHÉMA RELATIONNEL DE LA SOLUTION	30
SCHÉMA EA	30
SCHÉMA RELATIONNEL.....	31
IMPLÉMENTATION DES CONTRAINTES.....	32
ANALYSE TECHNIQUE	32
TECHNOLOGIES PROPOSÉES	32
ARCHITECTURE APPLICATIVE.....	32
<i>La couche UI/GUI</i>	32
<i>La couche BL (Business Layer)</i>	32
<i>La couche DAL (Data Access Layer)</i>	32
<i>Les Modèles et Erreurs</i>	32
<i>Architecture de ce projet</i>	33
ALGORITHMES SPÉCIFIQUES	33
<i>Calcul de classement</i>	33
<i>Calcul de scores</i>	33
<i>Création de matchs aller-retour</i>	34
<i>Matchs aller : première saison</i>	34
<i>Matchs retour : seconde saison</i>	34
<i>Génération automatique de calendrier</i>	34
ANNEXES	36
ENONCÉ EXAMEN (SOURCE MOODLE : PROF. V. FIEVEZ).....	36
INTERPRÉTATION PERSONNELLE DE L'ÉNONCÉ	38
<i>Règles & fonctionnement d'un championnat</i>	38
<i>Attentes relatives au programme, extraites de l'énoncé original & interprétées</i>	39
STRUCTURE DU CAHIER DES CHARGES.....	39
SECOND SCHÉMA EA.....	40

Contexte

Cadre

Ce projet est réalisé dans le cadre du cours de Projet de développement SGBD 2019-2020, à l'EPHEC. L'objectif de la réalisation de ce projet est de démontrer l'acquisition des compétences acquises au sein de ce cours : compréhension, acquisition et mise en pratique des concepts de développement SGBD.

Projet

Le projet consiste en la création d'un software, nommé « **FifaManager** », ayant pour objectif de **permettre la gestion de championnats annuel de football**, sur base des règles fournies dans l'énoncé du projet :

- 📄 Les consignes originales du projet se trouvent en fin de ce document : [Annexe « Enoncé d'examen »](#)
- 📄 L'interprétation personnelle de ces consignes, réécrites en ce sens, se trouvent dans les annexes également : [« Interprétation personnelle de l'énoncé »](#)

Analyse métier

Description de la solution envisagée

La solution devra permettre la gestion de championnats annuel de football (soccer) par 2 différents types d'intervenants, décrits plus bas dans ce document, gérés par 2 programmes distincts : *BackEnd*, et *MatchManager*. Les utilisateurs seront capables de :

1. Via le **BackEnd** : visualiser le classement, visualiser les résultats des matchs et les modifier, générer de manière automatique le calendrier d'un championnat, modifier un calendrier et/ou insérer dans le calendrier un match listé dans la liste des matchs non automatiquement insérés par le programme, inscrire/transférer un joueur dans une équipe, visualiser l'historique des matchs d'un joueur, visualiser les cartons.
2. Via le **MatchManager** : Inscrire les joueurs pour un match, encoder le résultat et/ou les cartons et/ou les goals d'un match.

Composition du projet

- ❖ 3 parties à réaliser :
 1. **Cahier des charges**
 2. **Base de données**
 3. 2 programmes :
 - a. **Backend**
 - b. **MatchManager**
- ❖ Format de remise du projet :
 - 1 solution contenant 7 projets (2 WinForms, 5 Class Libraries)
 - ou 1 solution contenant 8 projets (2 WinForms, 6 Class Libraries) — uniquement si on souhaite utiliser 2 DAL au lieu d'une

Les intervenants

1. Backend

Les utilisateurs de ce programme ont accès aux fonctions relatives à la gestion globale du championnat : gestion des calendriers, des transferts, des historiques, de l'ensemble des résultats,...

2. MatchManager

Les utilisateurs de ce programme ont uniquement accès à la gestion des feuilles de match : enregistrement des joueurs, résultat du match et des cartons liés au match.

Fonctions attendues

Exigences fonctionnelles

Fonctionnalités obligatoires dans le Backend

Code	Description
EF-BE-001	Inscrire des joueurs dans une équipe à un championnat
EF-BE-002	Transférer des joueurs durant les intersaisons
EF-BE-003	Visualiser un calendrier de championnat
EF-BE-004	Modifier un calendrier de championnat
EF-BE-005	Générer automatiquement des calendriers de matchs
EF-BE-006	Visualiser les résultats d'un championnat
EF-BE-007	Modifier les résultats d'un championnat
EF-BE-008	Visualiser les cartes des joueurs, des championnats
EF-BE-009	Visualiser l'historique des joueurs
EF-BE-010	Visualiser le classement d'un championnat

Fonctionnalités obligatoires dans MatchManagement

Code	Description
EF-MM-001	Inscrire les joueurs pour un match + contrôles (vérifier si l'inscription est possible)
EF-MM-002	Encoder les résultats
EF-MM-003	Encoder les cartes
EF-MM-004	Encoder les goals

Exigences non fonctionnelles

Code	Description
ENF-001	Réutilisable pour plusieurs championnats (différentes années)
ENF-002	Réutilisable pour des championnats de taille différente : nombre d'équipes inscrites peut varier chaque année
ENF-003	Pas d'autorisation d'accès nécessaire pour accéder aux programmes

Contraintes business

Point de vue technique

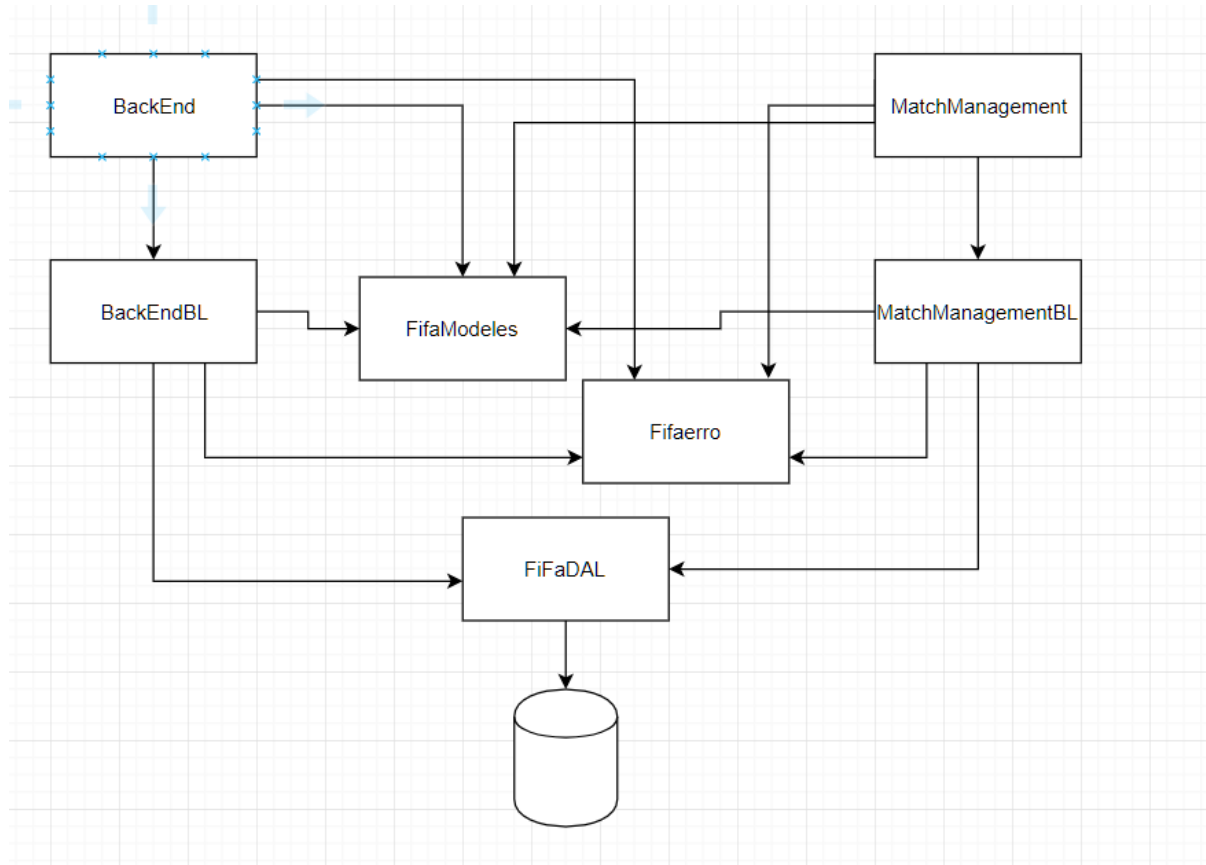
- ❖ Les tables sont accessibles uniquement par le SA (Server Admin)
- ❖ Utilisation de procédures stockées et vues avec schémas
- ❖ Sécurité au niveau des schémas
 - 📄 2 schémas : 1 Backend et 1 MatchManagement
- ❖ Découpe obligatoire : BL, DAL, MODELES, ERREURS
- ❖ Backend doit utiliser EF (Entity Framework)
- ❖ MatchManager doit utiliser ADO.NET
- ❖ 2 utilisateurs SQL :
 1. Backend
 2. MatchManagement

Composition du cahier des charges

- ❖ L'énoncé du projet, compris par nous-même
- ❖ Analyse des cas business (par exemple avec des mookups d'écrans) avec description complète de ce que ça fait
- ❖ Liste des règles et contrôles du projet :
 - R1 : un joueur ne peut être inscrit que dans une équipe par championnat

- R2 : ...
 - ➔ Ces règles sont les erreurs qui doivent être gérées par votre gestion d'erreurs
- ❖ Schéma EA
- ❖ Lister où sont implémentées les erreurs business
 - R1 est implémenté dans le trigger XXX et à tel endroit du code...
- ❖ Suivre la structure donnée lors du dernier cours (Annexe : [« Structure du cahier des charges »](#))

Architecture du projet, à respecter



Analyse fonctionnelle

Diagramme Use Cases (les principaux)



Descriptions textuelles des différents Use Cases

BackEnd

Accueil du programme BackEnd

BackEnd					
Classement	Matches	Calendrier	Gestion d'équipes	Joueurs	Cartons
Voir le classement	Voir / Modifier les résultats	Gestion de calendrier	Inscrire joueur(s) Transférer un joueur	Voir l'historique	Voir les cartons

EF-BE-001 — Inscrire des joueurs dans une équipe à un championnat

Ce cas d'utilisation permet à un utilisateur d'inscrire (et/ ou désinscrire) un ou plusieurs joueur(s) dans une équipe, pour les 2 saisons d'un championnat déterminé.

Déroulement

L'utilisateur sélectionne le menu déroulant « *Gestion d'équipes > Inscrire joueur(s)* ». Une fenêtre « *Inscription joueur(s) à une équipe pour un championnat* » s'ouvre. Il peut ensuite sélectionner un championnat dont la 1^{ère} saison n'a pas encore commencé dans le menu déroulant « *Sélectionner le championnat* ». Une fois le championnat désiré sélectionné, le menu déroulant « *Sélectionner l'équipe* » charge toutes les équipes inscrites à ce championnat.

L'utilisateur sélectionne l'équipe souhaitée dans la liste, et les listbox « *Joueurs disponibles* » et « *Joueurs inscrits dans l'équipe* » se chargent. Les joueurs inscrits dans aucune équipe pour le championnat sélectionné se retrouvent dans la liste « *joueurs disponibles* », tandis que les joueurs inscrits dans l'équipe sélectionnée pour le championnat sélectionné se retrouvent dans la liste « *joueurs inscrits dans l'équipe* ».

L'utilisateur peut sélectionner un ou plusieurs joueurs dans chaque liste et cliquer sur « *inscrire ...* » pour inscrire les joueurs sélectionnés dans la liste « *joueurs disponibles* » dans la liste des joueurs inscrits dans l'équipe ; et/ou cliquer sur « *désinscrire...* » pour désinscrire les joueurs sélectionnés dans la liste « *joueurs inscrits dans l'équipe* » et les rendre disponibles.

Une fois les modifications souhaitées effectuées, l'utilisateur peut enregistrer les modifications en cliquant sur « *Sauver modifications* », ou annuler les modifications en fermant la fenêtre (*dans ce cas, une fenêtre de confirmation apparaîtra*). Après sauvegarde, les listes de joueurs sont rafraîchies et l'utilisateur peut sélectionner un autre championnat et/ou une autre équipe, les menus déroulants chargeront les données nécessaires automatiquement.

Liens avec les règles/contraintes :

- [CF-RV-002](#)

EF-BE-002 — Transférer des joueurs durant les intersaisons

Ce cas d'utilisation permet à un utilisateur de transférer un joueur dans une équipe, durant l'intersaison d'un championnat.

Déroulement

L'utilisateur sélectionne le menu déroulant « *Gestion d'équipes > Transférer un joueur* ». Une fenêtre « *Transfert de joueur* » s'ouvre. Il peut ensuite sélectionner un championnat dont l'intersaison est en cours. *Si aucun championnat n'est en intersaison, un message préviendra l'utilisateur qu'aucun championnat n'est en intersaison, aucun transfert n'est donc possible et tous les boutons de la fenêtre sont inactifs.*

Une fois le championnat désiré sélectionné, le menu déroulant « *Sélectionner l'équipe du joueur à transférer* » charge toutes les équipes inscrites à ce championnat, **et qui comptent plus de 5 joueurs inscrits à la 2^e saison de ce championnat** ; et « *Sélectionner l'équipe vers laquelle transférer le joueur* » charge **les 3 dernières équipes du classement de ce championnat**. *En cas d'ex-aequo, la liste peut contenir plus de 3 équipes.*

L'utilisateur sélectionne l'équipe actuelle du joueur à transférer, et sélectionne ensuite le joueur à transférer dans la listbox contenant les joueurs de cette équipe.

Après, l'utilisateur sélectionne l'équipe vers laquelle transférer le joueur dans le menu déroulant « *Sélectionner l'équipe vers laquelle transférer le joueur* ». *La liste des joueurs de l'équipe receveuse s'affiche.*

Pour **sauver** le transfert, l'utilisateur clique sur « *Transférer* ».

Liens avec les règles/contraintes :

- [CF-RV-002](#)
- [CF-RV-003](#)
- [CF-RV-004](#)
- [CF-RV-005](#)
- [CF-RV-006](#)

EF-BE-003 — Visualiser un calendrier de championnat

Ce cas d'utilisation permet à un utilisateur de visualiser le calendrier d'un championnat.

Déroulement

Pour visualiser le calendrier d'un championnat, l'utilisateur sélectionne le menu déroulant « *Calendrier* » et clique sur « *Gestion de calendrier* ». Une fenêtre « *Gestion de calendrier* » s'ouvre.

L'utilisateur sélectionne le championnat désiré dans le menu déroulant, et sélectionne la ou les saison(s) qu'il souhaite visualiser. Le programme va automatiquement charger les matchs avec date dans les « *matchs programmés* » et les matchs sans date, s'il y en a, dans « *matchs non programmés* ».

BackEnd

Classement Matches Calendrier Gestion d'équipes Joueurs Cartons

Gestion de calendrier

Nom_Championnat ▼

☐ Saison 1 ☐ Saison 2 ☒ Saison 1&2

Générer un nouveau calendrier

Matches programmés :

Match_ID	Date	Jour	Equipe_Dom	Equipe_Visite
....				
...				
...				

Matches non programmés :

Match_ID	Date	Jour	Equipe_Dom	Equipe_Visite
....				
...				
...				

Liens avec les règles/contraintes :

/

EF-BE-004 — Modifier un calendrier de championnat

Ce cas d'utilisation permet à un utilisateur de modifier manuellement un calendrier de matchs pour une saison, ou un championnat complet, déterminé(e).

Déroulement

Pour accéder au menu de modification de calendrier, l'utilisateur sélectionne le menu déroulant « *Calendrier* » et clique sur « *Gestion de calendrier* ». Une fenêtre « *Gestion de calendrier* » s'ouvre.

L'utilisateur sélectionne le championnat désiré dans le menu déroulant, et sélectionne la ou les saison(s) pour laquelle/lesquelles il souhaite modifier le calendrier. Le programme va automatiquement charger les matchs avec date dans les « *matchs programmés* » et les matchs sans date dans « *matchs non programmés* ».

L'utilisateur peut ensuite sélectionner chaque match, un par un, pour lequel il souhaite changer la date en double cliquant sur la ligne du match dans son tableau.

Lorsqu'il double clique sur un match, une fenêtre « *Modification calendrier* » s'ouvre si la date du match n'est pas encore passée, sinon un message avertit l'utilisateur qu'il n'est pas possible de modifier la date de ce match. Cette fenêtre affiche les informations actuelles du match et propose un datepicker balisé par les dates de début/fin de la saison du match, et n'acceptant pas la sélection des jours de week-end. L'utilisateur peut sélectionner la date de son choix et cliquer sur « *Sauver* » pour sauver les changements ou fermer la fenêtre pour ne pas sauvegarder.

Lorsque l'utilisateur sélectionne une nouvelle date de match dans le datepicker, une vérification est faite par le programme pour s'assurer qu'une des 2 équipes du match ne joue pas ce jour-là, si oui, un message en avertit l'utilisateur pour qu'il sélectionne une autre date, et la sauvegarde n'est pas possible (bouton « *sauver* » inactif).

Liens avec les règles/contraintes :

- [CF-RV-019](#), [CF-RV-020](#), [CF-RV-021](#)

EF-BE-005 — Générer automatiquement des calendriers de matchs

Ce cas d'utilisation permet à un utilisateur de générer de manière automatique un calendrier de matchs pour une saison, ou un championnat complet. Les matchs générés sont placés le week-end, et ceux qui n'ont pas pu être placés les jours de week-end sont sans date, à modifier manuellement.

Déroulement

Pour accéder au menu de génération de calendrier, l'utilisateur sélectionne le menu déroulant « *Calendrier* » et clique sur « *Gestion de calendrier* ». Une fenêtre « *Gestion de calendrier* » s'ouvre.

L'utilisateur sélectionne le championnat désiré dans le menu déroulant, et sélectionne la ou les saison(s) pour laquelle/lesquelles il souhaite générer un calendrier. Ensuite, il clique sur « *Générer un nouveau calendrier* ». Si la/les saison(s) sélectionnée(s) ont déjà commencé, un message avertira l'utilisateur qu'il n'est plus possible de générer un calendrier pour la saison sélectionnée.

Sinon, une fenêtre « *Génération de calendrier* » s'ouvre. Si un calendrier est déjà généré, le calendrier actuel s'affiche dans les 2 tableaux « *Matchs avec date* » et « *Matchs sans date* ». L'utilisateur clique sur « *(Re)Générer un nouveau calendrier* » et le programme va (re)calculer un nouveau calendrier de matchs et les afficher dans les 2 tableaux respectifs. L'utilisateur peut ensuite : sauver les modifications en cliquant sur « *Sauver* », annuler les modifications en cliquant sur « *Annuler* », ou régénérer un nouveau calendrier si la proposition ne convient pas en cliquant à nouveau sur « *(Re)Générer un nouveau calendrier* ».

The first screenshot shows the 'Gestion de calendrier' window. It has a menu bar with 'Classement', 'Matches', 'Calendrier', 'Gestion d'équipes', 'Joueurs', and 'Cartons'. Below the menu is a 'Gestion de calendrier' sub-window. Inside, there is a dropdown menu for 'Nom_Championnat', three checkboxes for 'Saison 1', 'Saison 2', and 'Saison 1&2' (which is checked), and a button labeled 'Générer un nouveau calendrier'.

The second screenshot shows the 'Génération de calendrier' window. It also has the same menu bar. Below it is a 'Génération de calendrier' sub-window. At the top, there is a button labeled '(Re)Générer un nouveau calendrier'. Below this, there are two sections: 'Matches avec date :' and 'Matches sans date :'. Each section contains a table with 5 columns: 'Match_ID', 'Date', 'Jour', 'Equipe_Dom', and 'Equipe_Visite'. The 'Matches avec date' table has 3 rows, and the 'Matches sans date' table has 3 rows. At the bottom of each table section are buttons labeled 'Sauver' and 'Annuler' respectively.

Liens avec les règles/contraintes :

- [CF-RV-017](#) , [CF-RV-018](#), [CF-RV-019](#)

EF-BE-006 — Visualiser les résultats d'un championnat

Ce cas d'utilisation permet à un utilisateur de visualiser les résultats d'un championnat.

Déroulement :

Pour visualiser les résultats d'un championnat, l'utilisateur doit sélectionner le menu déroulant « *Matches* » et cliquer sur « *Voir / Modifier les résultats* ». Une fenêtre « *Voir/Modifier les résultats* » s'ouvre.

Il peut ensuite sélectionner le championnat désiré dans le menu déroulant, et la ou les saison(s) à afficher dans le tableau de résultats. Le tableau se charge automatiquement au fur et à mesure de la sélection.

Match_ID	Date	Equipe_Domicile	Equipe_Visiteuse	Goals_Domicile	Result_Visit
1234	13/05/2020	Barcelone	Real Madrid	Gagné	Perdu
1235	13/05/2020	Rome	Venise	Nul	Nul
1236	13/05/2020	Paris	Bordeaux	Gagné	Forfait

Liens avec les règles/contraintes :

/

EF-BE-007 — Modifier les résultats d'un championnat

Ce cas d'utilisation permet à un utilisateur de modifier les résultats d'un ou plusieurs matchs de championnat.

Déroulement :

Pour modifier les résultats d'un championnat, l'utilisateur doit sélectionner le menu déroulant « *Matches* » et cliquer sur « *Voir / Modifier les résultats* ». Une fenêtre « *Voir/Modifier les résultats* » s'ouvre.

Il peut ensuite sélectionner le championnat désiré dans le menu déroulant, et la ou les saison(s) à afficher dans le tableau de résultats. Le tableau se charge automatiquement au fur et à mesure de la sélection.

Pour modifier un résultat, l'utilisateur double clique sur la ligne du match désiré dans le tableau. Une fenêtre « *Modifier les résultats* » s'ouvre. Cette fenêtre affiche les types de résultats possibles, pour chaque équipe, avec les résultats actuels présélectionnés. L'utilisateur sélectionne les nouveaux résultats désirés et clique sur « *Sauver* » pour sauvegarder, ou « *Annuler* » pour annuler les modifications.

Voir / Modifier les résultats

Nom_Championnat ☐ Saison 1 ☐ Saison 2 ☒ Saison 1&2

Match_ID	Date	Equipe_Domicile	Equipe_Visiteuse	Goals_Domicile	Result_Visit
1234	13/05/2020	Barcelone	Real Madrid	Gagné	Perdu
1235	13/05/2020	Rome	Venise	Nul	Nul
1236	13/05/2020	Paris	Bordeaux	Gagné	Forfait

Modifier les résultats

Match #

Nom Eqp Dom: Gagné, Perdu, Nul, ...

Nom Eqp Visit: Gagné, Perdu, Nul, ...

Sauver Annuler

Liens avec les règles/contraintes :

/

EF-BE-008 — Visualiser les cartes des joueurs, des championnats

Ce cas d'utilisation permet à un utilisateur de visualiser les cartons d'un championnat.

Déroulement :

Pour visualiser les cartons d'un championnat, l'utilisateur doit sélectionner le menu déroulant « *Cartons* » et cliquer sur « *Voir les cartons* ». Une fenêtre « *Liste des cartons* » s'ouvre.

Il peut ensuite sélectionner le championnat désiré dans le menu déroulant, et le ou les type(s) de cartons à afficher dans le tableau de cartons. Il clique ensuite sur « *Filtrer* » pour appliquer les filtres.

BackEnd

Classement Matches Calendrier Gestion d'équipes Joueurs Cartons

Voir les cartons

Cartons

Nom_Championnat ▼

...

*Tous

Type_carton ▼

Jaune

Rouge

*Tous

Filtrer

Championnat	Couleur	ID_Match	ID_Joueur	Nom_Joueur	IdMatchSuspension
....					
....					
....					
....					

Liens avec les règles/contraintes :

/

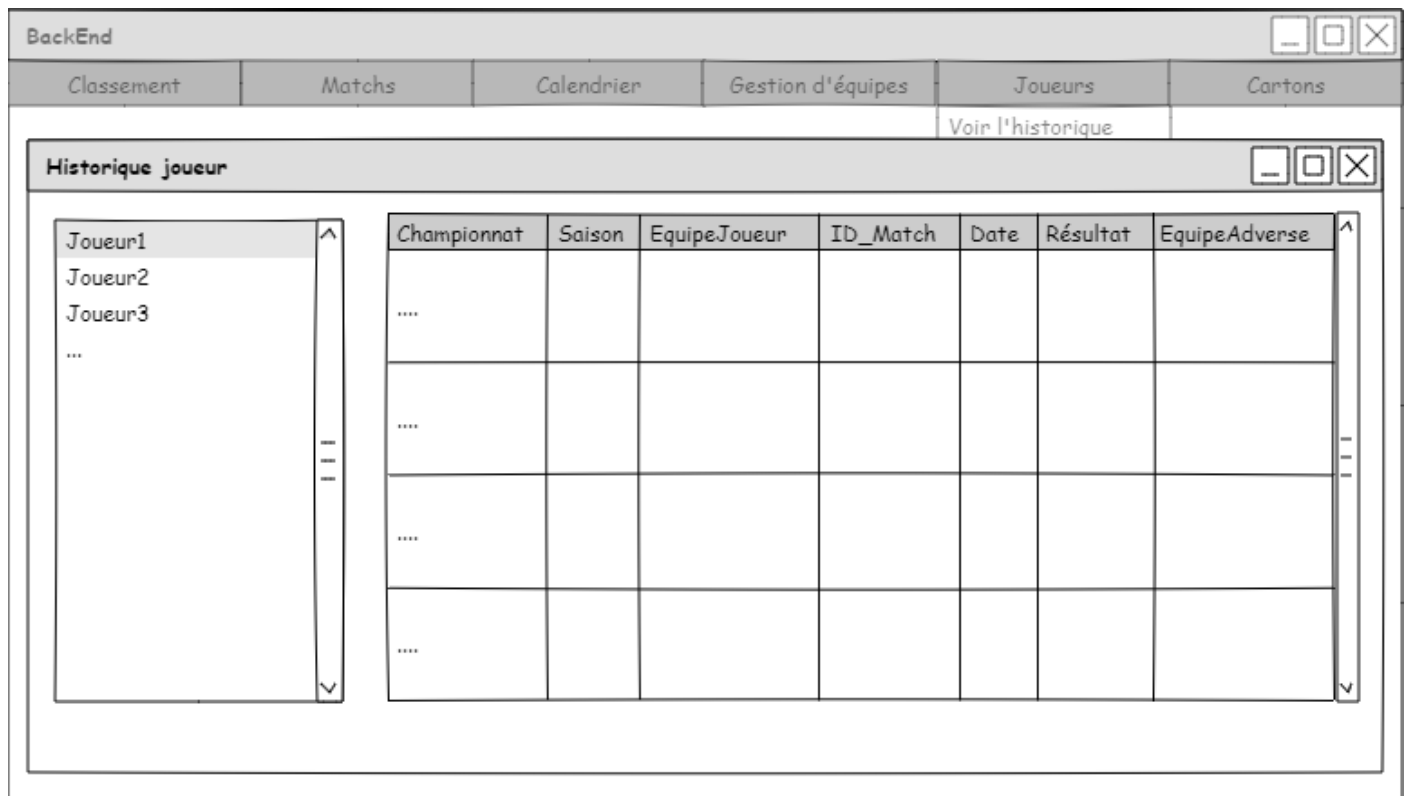
EF-BE-009 — Visualiser l'historique des joueurs

Ce cas d'utilisation permet à un utilisateur de visualiser l'historique des matchs des joueurs.

Déroulement :

Pour visualiser les historiques des joueurs, l'utilisateur doit sélectionner le menu déroulant « *Joueurs* » et cliquer sur « *Voir l'historique* ». Une fenêtre « *Historique des joueurs* » s'ouvre.

Il peut ensuite sélectionner le joueur désiré dans la liste complète des joueurs. Une fois le joueur sélectionné, le tableau se charge automatiquement avec l'historique de tous les matchs du joueur.

**Liens avec les règles/contraintes :**

/

EF-BE-010 — Visualiser le classement d'un championnat

Ce cas d'utilisation permet à un utilisateur de visualiser le classement d'un championnat.

Déroulement :

Pour visualiser le classement d'un championnat, l'utilisateur doit sélectionner le menu déroulant « *Classement* » et cliquer sur « *Voir le classement* ». Une fenêtre « *Classement* » s'ouvre.

Il peut ensuite sélectionner le championnat désiré dans le menu déroulant. Une fois le championnat sélectionné, le tableau se charge automatiquement avec le classement du championnat.

Rang	Points	Equipe	Equipe_ID
1	##	Nom_Equipe_1	#####
2	##	Nom_Equipe_2	#####
3	##	Nom_Equipe_3	#####
...

Liens avec les règles/contraintes :

- [CF-RC-002, CF-RC-003, CF-RC-004](#)

*MatchManager**Accueil du programme MatchManager*

Inscriptions	Feuilles de match

EF-MM-001 — Inscrire les joueurs pour un match + contrôles (vérifier si l'inscription est possible)

Ce cas d'utilisation permet à un utilisateur d'inscrire des joueurs pour un match.

Déroulement :

Pour inscrire des joueurs à un match, l'utilisateur doit sélectionner le menu « *Inscriptions* ». Une fenêtre « *Inscrire des joueurs à un match* » s'ouvre.

Cette fenêtre charge une liste déroulante de championnats de l'année en cours ou des années postérieure à l'année en cours. L'utilisateur sélectionne le championnat désiré, la/les saison(s) désirée(s) et la liste déroulante des matchs éditables (= dont la date est postérieure à la date du jour) se charge. L'utilisateur sélectionne le match désiré et clique sur « *Inscrire les joueurs pour ce match* ».

Une fenêtre « *Sélectionner l'équipe* » s'ouvre et l'utilisateur clique sur l'équipe pour laquelle il souhaite inscrire le(s) joueur(s).

Une fenêtre « *Inscription joueur(s) pour le match* » s'ouvre et charge la liste des joueurs de l'équipe, qui ne sont pas inscrits pour ce match, dans la liste des « *Joueurs disponibles* », et la liste des joueurs de l'équipe inscrits pour ce match dans la liste « *Joueurs inscrits pour le match* ».

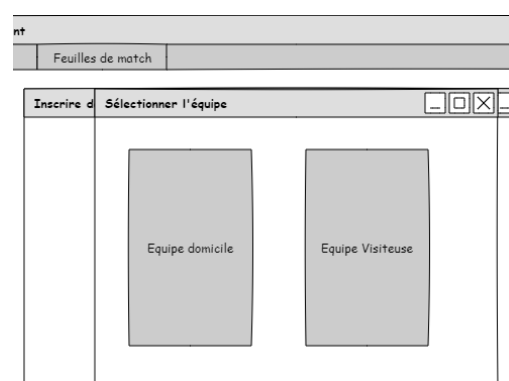
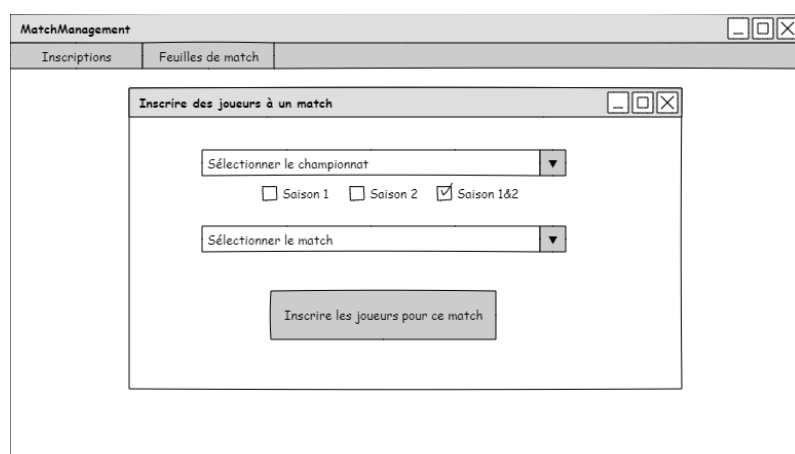
L'utilisateur peut sélectionner un ou plusieurs joueurs disponibles et cliquer sur « *Inscrire...* » pour les inscrire dans l'équipe du match. Une fois cliqué, les joueurs apparaissent dans la liste des joueurs du match. A chaque fois que l'utilisateur clique sur « *Inscrire...* » une vérification est faite pour s'assurer que le nombre de joueurs inscrits ne dépasse pas le seuil maximum pour un match, si oui, l'utilisateur est averti et les joueurs ne changent pas de liste.

A l'inverse, il peut sélectionner un ou plusieurs joueurs inscrits et cliquer sur « *Désinscrire...* » pour les désinscrire de l'équipe du match. Les joueurs apparaîtront ensuite dans la liste des joueurs disponibles. A chaque fois que l'utilisateur clique sur « *Désinscrire...* » une vérification est faite pour s'assurer que le nombre de joueurs inscrits ne dépasse pas le seuil minimum pour un match, si oui, l'utilisateur est averti et les joueurs ne changent pas de liste.

Il lui est possible de désélectionner toute sa sélection en cliquant sur le bouton « *Désélectionner tout* ».

Une fois que toutes les modifications souhaitées ont été effectuées, l'utilisateur peut cliquer sur « *Sauver modifications* » pour sauver les modifications. Il est possible de sauver les inscriptions des joueurs si la liste de joueurs n'atteint pas le seuil minimum requis pour un match, l'utilisateur est alors averti qu'il risque de perdre le match par forfait et peut décider de modifier la liste des joueurs inscrits ou de confirmer la sauvegarde.

S'il souhaite annuler, il suffit de cliquer sur « *Annuler modifications* », et la fenêtre se rafraîchit en annulant les modifications.



The screenshot shows a software window titled "MatchManagement" with a sub-window titled "Inscription joueur(s) pour le match". The sub-window contains two list boxes. The left list box, titled "Joueurs disponibles", contains the items "JoueurDispo1", "JoueurDispo2", "JoueurDispo3", "JoueurDispo4", and "...". The right list box, titled "Joueurs inscrits dans l'équipe", contains the items "JoueurEqp1", "JoueurEqp2", "JoueurEqp3", "JoueurEqp4", and "...". Between the two list boxes are two buttons: "> Inscrire les joueurs sélectionnés" and "< Désinscrire les joueurs sélectionnés". Below the left list box is a button labeled "Désélectionner tout". Below the right list box is a button labeled "Annuler modifications". To the right of the right list box is a button labeled "Sauver modifications".

Liens avec les règles/contraintes :

- [CF-RV-007, CF-RV-009, CF-RV-010](#)

EF-MM-002 — Encoder les résultats

Ce cas d'utilisation permet à un utilisateur d'encoder les résultats d'un match, le jour du match.

Déroulement :

Pour encoder les résultats d'un match, l'utilisateur doit sélectionner le menu « *Feuilles de match* ». Une fenêtre « *Sélection du match à éditer* » s'ouvre.

Il peut ensuite sélectionner le match dans le menu déroulant listant les matchs du jour, et cliquer sur « *Editer la feuille de match* ». Une fenêtre « *Feuille de match* » s'ouvre.

L'onglet « *Résultats* » de cette fenêtre affiche les scores du match, et comprend 2 listes déroulantes avec les types de résultats attribuables à chacune des 2 équipes. Le programme sélectionne par défaut les valeurs à sélectionner en fonction des résultats : « *Gagné* » pour l'équipe avec le plus de goals ou si l'autre équipe est forfait, « *Perdu* » pour l'équipe avec le moins de goals, « *Perdu/Forfait* » pour la/les équipe(s) n'ayant pas assez de joueurs inscrits au match, « *Nul* » si les 2 équipes ont le même nombre de goals ».

Il est possible, pour l'utilisateur, de sélectionner d'autres valeurs des menus déroulants.

L'utilisateur clique sur « *Enregistrer* » pour sauvegarder les résultats sélectionnés. Si les résultats sélectionnés ne sont pas en accord avec les résultats par défaut, le programme en avertit l'utilisateur et demande confirmation avant sauvegarde.

Une fois la sauvegarde effectuée, la fenêtre « *Feuille de match* » ferme. Il est toujours possible d'y accéder tant que le jour d'accès est égal à celui du match.

The screenshot displays the 'MatchManagement' application window. The 'Feuille de match' (Match Sheet) window is open, showing two main sections: 'Evenements' (Events) and 'Résultats' (Results).

Evenements Section:

- 'Choix événement' (Event Choice) dropdown menu with options: Goal, Carton rouge (Red Card), Carton jaune (Yellow Card), Goal contre son camp (Goal against own team).
- 'Choix équipe' (Team Choice) dropdown menu.
- 'Choix joueur' (Player Choice) dropdown menu.
- 'Nb événement:' (Number of event) input field with 'nb' as the placeholder.
- 'Enregistrer' (Save) button.

Résultats Section:

- Header: 'Domicile' (Home) and 'Visite' (Away).
- Sub-headers: 'Nom_Equipe_Domicile' and 'Nom_Equipe_Visite'.
- Two 'Choix type résultat' (Result type choice) dropdown menus for each team, with options: Gagné (Won), Perdu (Lost), Forfait (Forfeit), Nul (Draw).
- 'Enregistrer' (Save) button.

In the background, a smaller window titled 'Sélection du match à éditer' (Select match to edit) is visible, showing a list of matches for the day and an 'Editer la feuille de match' (Edit match sheet) button.

Liens avec les règles/contraintes :

- [CF-RV-0011](#)
- [CF-RV-0022](#)
- [CF-RC-005, CF-RC-006, CF-RC-007, CF-RC-008](#)

EF-MM-003 — Encoder les cartes

Ce cas d'utilisation permet à un utilisateur d'encoder les cartons d'un match, le jour du match.

Déroulement :

Pour encoder les cartons d'un match, l'utilisateur doit sélectionner le menu « Feuilles de match ». Une fenêtre « Sélection du match à éditer » s'ouvre.

Il peut ensuite sélectionner le match dans le menu déroulant listant les matchs du jour, et cliquer sur « Editer la feuille de match ». Une fenêtre « Feuille de match » s'ouvre.

L'onglet « Evenements » de cette fenêtre affiche un menu déroulant listant les différents types d'événements, un menu déroulant listant les équipes du match sélectionné, un menu déroulant listant les joueurs de l'équipe sélectionnée, et une case à compléter pour saisir le nombre d'occurrences de cet événement.

L'utilisateur sélectionne le type du carton dans le menu déroulant des choix d'événements, puis l'équipe et enfin le joueur et l'occurrence de l'événement, et clique sur « Enregistrer ». Une fenêtre « Choix des matchs suspendus » s'ouvre.

La fenêtre affiche le nombre de matchs de suspension à sélectionner, et la liste des matchs de l'équipe restants pour la saison, sans les matchs desquels le joueur est déjà suspendu. S'il s'agit d'un carton avec pénalité obligatoire dès le prochain match, la liste est déjà présélectionnée.

Sinon, l'utilisateur sélectionne les matchs de suspension du choix du joueur jusqu'à ce que le nombre de suspensions à sélectionner soit atteint, ou qu'il n'y ait plus de match à sélectionner.

L'utilisateur clique sur « Sauvegarder... » pour sauver les choix et enregistrer le carton.

The first screenshot shows the 'Feuilles de match' menu in the top bar. Below it, the 'Sélection du match à éditer' window is open, displaying a list of matches for the day and a 'Sélectionner le match' dropdown. An 'Editer la feuille de match' button is at the bottom.

The second screenshot shows the 'Feuille de match' window. It has two main sections: 'Evenements' and 'Résultats'. The 'Evenements' section contains three dropdowns: 'Choix événement' (with options: Goal, Carton rouge, Carton jaune, Goal contre son camp), 'Choix équipe', and 'Choix joueur'. There is also a 'Nb événement:' input field and an 'Enregistrer' button. The 'Résultats' section has two dropdowns for 'Choix type résultat' (with options: Gagné, Perdu, Forfait, Nul) for 'Domicile' and 'Visite' teams, and an 'Enregistrer' button.

The third screenshot shows the 'Choix des matchs suspendus' window. It displays '# Joueur concerné' and '# nb matchs suspension'. Below these is a list of matches: 'MatchRestant1', 'MatchRestant2', 'MatchRestant3', 'MatchRestant4', and '...'. There is a '# nb matchs à sélectionner' input field and a 'Sauvegarder choix des matchs suspendus' button.

Liens avec les règles/contraintes :

- [CF-RV-012](#)
- [CF-RV-013](#), [CF-RV-014](#), [CF-RV-015](#)
- [CF-RC-011](#), [CF-RC-012](#)

EF-MM-004 — Encoder les goals

Ce cas d'utilisation permet à un utilisateur d'encoder les goals d'un match, le jour du match.

Déroulement :

Pour encoder les goals d'un match, l'utilisateur doit sélectionner le menu « *Feuilles de match* ». Une fenêtre « *Sélection du match à éditer* » s'ouvre.

Il peut ensuite sélectionner le match dans le menu déroulant listant les matchs du jour, et cliquer sur « *Editer la feuille de match* ». Une fenêtre « *Feuille de match* » s'ouvre.

L'onglet « *Evenements* » de cette fenêtre affiche un menu déroulant listant les différents types d'événements, un menu déroulant listant les équipes du match sélectionné, un menu déroulant listant les joueurs de l'équipe sélectionnée, et une case à compléter pour saisir le nombre d'occurrences de cet événement.

L'utilisateur sélectionne le type du goal dans le menu déroulant des choix d'événements, puis l'équipe et enfin le joueur et l'occurrence de l'événement, et clique sur « *Enregistrer* ». Les scores affichés dans la fenêtre « *Feuille de match* » se mettent à jour, ainsi que les résultats par défaut.

The image displays two screenshots of the FIFA Manager 2020 software interface. The left screenshot shows a window titled 'Sélection du match à éditer' with a dropdown menu labeled 'Liste des matchs du jour' and a button 'Editer la feuille de match'. The right screenshot shows a window titled 'Feuille de match' with two tabs: 'Evenements' and 'Résultats'. The 'Evenements' tab contains three dropdown menus: 'Choix événement' (with options: Goal, Carton rouge, Carton jaune, Goal contre son camp), 'Choix équipe', and 'Choix joueur', followed by a text input field 'Nb événement' and an 'Enregistrer' button. The 'Résultats' tab contains two dropdown menus for 'Choix type résultat' under the headings 'Domicile' and 'Visite' (with options: Gagné, Perdu, Forfait, Nul), and an 'Enregistrer' button.

Liens avec les règles/contraintes :

- [CF-RV-011](#)
- [CF-RV-022](#)
- [CF-RC-005, CF-RC-006, CF-RC-007, CF-RC-008](#)
- [CF-RC-009, CF-RC-010](#)

Contraintes fonctionnelles

Règles d'accès et autorisation

Code	Description
CF-AA-001	Le programme BackEnd accède à la base de données via l'utilisateur SQL UserBackEnd
CF-AA-002	Le programme MatchManager accède à la base de données via l'utilisateur SQL UserMatchManagement

Les 2 utilisateurs SQL ayant chacun leur propre schéma SQL.

Règles de structure

Championnat

Code	Description
CF-RS-001	Un seul championnat par année civile DB > Table Championnats > Clés > UK_Championnats : clé unique sur l'année du championnat
CF-RS-002	Chaque championnat peut avoir un nombre variable d'équipes inscrites (Constitution_Equipes)
CF-RS-003	Un championnat est divisé en 2 saisons (quarters) DB > Table Saisons > Déclencheurs > TR_Saisons_CheckSsnNum => trigger qui vérifie que maximum 2 saisons sont encodées pour un championnat, et que les valeurs sont 1 et 2
CF-RS-004	Un championnat a un ID unique

Equipes

Code	Description
CF-RS-005	Une équipe a un ID unique
CF-RS-006	Une équipe a un nom unique, obligatoire DB > Table Championnats > Clés > UK_Equipes : clé unique sur le nom de l'équipe
CF-RS-007	Une équipe peut être inscrite à 0 ou plusieurs championnats différents
CF-RS-008	Une équipe peut contenir plusieurs joueurs, par championnat

Constitution_Championnat

Code	Description
CF-RS-009	L'inscription d'une équipe à un championnat est unique DB > Table Constitution_Championnat > Index > UX_Constitution_Championnat_ChampID_EqpID : l'association de l'id d'une équipe et de l'id d'un championnat est unique
CF-RS-010	Chaque constitution de championnat doit contenir une équipe enregistrée et un championnat enregistré
CF-RS-011	Une constitution de championnat possède un ID unique

Constitution_Equipes

Code	Description
CF-RS-012	Un joueur ne peut jouer que dans une équipe à la fois par saison, par championnat DB > table Constitution_Equipes > index > UX_1Jr_1Ssn : unicité d'association de l'id saison et l'id joueur (CEqp_Ssn_ID + CEqp_Jr_ID)
CF-RS-013	Une équipe est constituée de minimum 5 joueurs et maximum 10 joueurs
CF-RS-014	Une constitution d'équipe est composée d'un ID unique, et d'un ensemble unique : Constitution_Championnat_ID, Joueur_ID, et Saison_ID

Saisons

Code	Description
CF-RS-015	Un championnat est divisé en 2 saisons (quarters)
	DB > Table Saisons > Déclencheurs > TR_Saisons_CheckSsnNum => trigger qui vérifie que maximum 2 saisons sont encodées pour un championnat, et que les valeurs sont 1 et 2
CF-RS-016	Une saison dure 5 semaines
	VS > Models > MdlChampionnat : Propriété statique : SsnWeeks = 5
CF-RS-017	Une saison est unique
	DB > Table Saisons > Index > UX_Saisons_ChampID_SsnNum => index unique entre l'id d'un championnat et le numéro de la saison (1/2)
CF-RS-018	2 saisons d'un championnat doivent débuter durant la même année civile
	DB > Table Saisons > Déclencheurs > TR_Saisons_CheckYear=> trigger qui vérifie que les 2 saisons font partie de la même année que le championnat lié
CF-RS-019	Une saison est identifiée par un ID unique, et un ensemble obligatoire et unique d'id championnat et numéro de saison (1/2)
CF-RS-020	Une saison possède une date de début, obligatoire, et unique
	DB > Table Saisons > Clés > UK_Saisons_Date_Debut => clé unique sur la date de début de saison
CF-RS-021	Une saison doit et ne peut être liée qu'à un seul championnat

Joueurs

Code	Description
CF-RS-022	Un joueur est unique dans la database, il doit posséder un ensemble unique de nom et prénom, et un id unique
	DB > table Joueurs > index > Unicity_Joueur => index unique sur l'association du nom et du prénom du joueur
CF-RS-023	Un joueur peut être inscrit dans 0 ou plusieurs équipes, mais max une par saison de championnat

Inscription_Match

Code	Description
CF-RS-024	Un joueur ne peut être inscrit qu'une fois à un match
	DB > table Inscription_Matches > index > UX_Inscription_Matches_CEqpID_MchID => index unique sur l'association du joueur (CEqpID) et du match (MchID)
CF-RS-025	Maximum 7 inscriptions peuvent être créées, par équipe, par match
	DB > Inscription_Matches > déclencheurs > TR_InscriptMatches_CheckMax7JrParEqp : empêche l'insertion de plus de 7 joueurs pour une équipe pour un match
CF-RS-026	Une inscription match est composée d'un id unique et obligatoire, et d'un ensemble unique et obligatoire de : id de Constitution_Equipe et id de Match
CF-RS-027	Une inscription match peut être liée à 0 ou plusieurs Events

Events

Code	Description
CF-RS-028	Un Event doit posséder un Type_Events
CF-RS-029	Un Event possède un id unique, et est composé d'un ensemble obligatoire et unique de : id inscription match, id type_event et id match
CF-RS-030	Un Event n'est et doit être lié qu'à une seule inscription match
CF-RS-031	Un Event peut être lié à 0 ou plusieurs suspensions

Type_Events

Code	Description
CF-RS-032	Un type_event est composé d'un id unique obligatoire et d'un ensemble obligatoire de : libellé, nb de jours de suspensions (0 si pas d'application), et d'une propriété Next Match de type booléen pour signaler si la suspension est applicable obligatoirement dès le match suivant
CF-RS-033	Un type_event peut être lié à 0 ou plusieurs Events
CF-RS-034	Il existe 4 types d'événements : goal, goal contre son camp, carton rouge, carton jaune

Suspensions

Code	Description
CF-RS-035	Une suspension est composée d'un id unique et obligatoire, et d'un ensemble de : id match (non obligatoire car il est possible que la suspension ne puisse pas être effectuée) et d'un id d'événement (obligatoire)
CF-RS-036	Une suspension doit et ne peut être liée qu'à un seul Event

Match

Code	Description
CF-RS-037	Un match est unique : une équipe ne joue qu'une fois contre une autre équipe, par championnat, à domicile et qu'une fois en visite
	DB > Matches > index > UX_Matches_EqpDomID_EqpVisitID : unicité de l'association entre équipe domicile et équipe visiteuse
CF-RS-038	Un match est composé d'un id obligatoire et unique
CF-RS-039	Un match comprend 2 équipes différentes obligatoires : une équipe domicile et une équipe visiteuse
CF-RS-040	Un match doit être lié à une et une seule saison
CF-RS-041	Un match doit avoir une date, mais pouvant être de valeur nulle
CF-RS-042	Un match doit avoir un Type_Resultats, par équipe
CF-RS-043	Un match doit avoir une date de dernière mise à jour effectuée
CF-RS-044	Un match peut avoir de 0 à plusieurs Events
CF-RS-045	Un match peut être lié à plusieurs inscriptions match
CF-RS-046	Un match peut être lié à 0 ou plusieurs suspensions

Types_Resultats

Code	Description
CF-RS-047	Un Type_Resultat doit avoir un id unique, obligatoire et est composé d'un libellé unique et obligatoire, et d'un nb de points
	DB > Types_Resultats > Clés > UK_Types_Resultats_Libelle : unicité du libellé
CF-RS-048	Un Type_Resultat peut être lié à 0 ou plusieurs matchs
CF-RS-049	3 types de résultats : Gagné (3 points), Perdu (0 point), ou Nul (1 point)
	interprétation personnelle : ajout de Perdu/Fortait (0 point) et Pas joué/encodé (0 point)

Règles de validation

Constitution_Equipes

Code	Description
CF-RV-001	Vérification de la validité des données entre l'envoi des données pour le formulaire et la réception des données pour écriture : colonne LastUpdate
	DB > Constitution_Equipes > déclencheurs > CoEq_Update_SetLastUpdate : insère automatiquement la date de mise à jour lorsqu'une ligne est modifiée
CF-RV-002	Une équipe est constituée de minimum 5 joueurs et maximum 10 joueurs
	VS > BackEnd > BL > EquipesServices > SeuilMaxJoueurs_OK + SeuilMinJoueurs_OK : vérifient que les seuils max et min de joueurs soient valides
	VS > BackEnd > UI > GestionEquipe > frmInscriptionJoueurs > btnInscrire_Click + btnDesinscrire_Click + btnSave_Click : vérifient que les seuils max et min de joueurs soient valides via les services SeuilMaxJoueurs_OK + SeuilMinJoueurs_OK
	DB > Constitution_Equipes > déclencheurs > TR_CoEq_Update_CheckNbJoueurs: vérifie le nombre de joueur max et min d'une équipe après un transfert (update)
CF-RV-003	VS > BackEnd > BL > EquipesServices > NbJoueursMaxApresTransfert_OK + NbJoueursMinApresTransfert_OK : vérifient avant un transfert que les seuils max et min de joueurs soient toujours valides après l'exécution du transfert
	Un transfert ne peut se faire que durant l'intersaison
CF-RV-004	VS > BackEnd > BL > ChampionnatsServices : GetInInterSsnChampionnats() : renvoie la liste des championnats dont l'intersaison est en cours : càd le(s) championnat(s) pour le(s)quel(s) la date d'aujourd'hui se situe entre la date de fin de la 1 ^{ère} saison et la date du début de la 2 ^e saison
	Une équipe ne peut pas transférer de joueurs si elle n'a que 5 joueurs inscrits
CF-RV-005	DB > Procédures stockées > BE_SP_SelectEqPlus5Joueurs_Ssn2_byChamp : renvoie la liste des équipes contenant + de 5 joueurs inscrits pour le championnat sélectionné
	Un transfert ne peut s'effectuer que vers les 3 dernières équipes du classement
CF-RV-006	VS > BackEnd > BL > EquipesServices > GetEqPouvantRecevoirTrnsfrt : prend les 3 dernières équipes du classement. + interprétation personnelle : plus que 3 équipes sont disponibles en cas d'égalité de points avec plusieurs équipes (je ne prends pas le nombre de goals en considération)
	Lors d'un transfert, un joueur ne peut pas être transféré dans son équipe actuelle si celle-ci fait partie des équipes pouvant recevoir un transfert
CF-RV-007	VS > BackEnd > UI > GestionEquipe > frmTransfertJoueur : CheckIfSlctdEqOrgnlIsAlsoInEqTrnsfrt() + EnablingOrDisabling_BtnTransferer() : désactive l'équipe actuelle du joueur dans la liste des équipes de transfert et vérifie qu'un joueur a été sélectionné pour cliquer sur transférer

Inscription_Match

Code	Description
CF-RV-007	Inscrire des joueurs à un match peut se faire que jusqu'à la veille du jour du match (inclue)
	VS > MatchManager > BL > MatchServices > GetPlayersInscription_StillEditable_MatchList : renvoie une liste ne comprenant que les matchs dont la date est postérieure à la date du jour
CF-RV-008	Editer une feuille de match ne peut se faire que le jour du match (le but étant de s'assurer que les sanctions du match soient applicables/appliquées)
	VS > MatchManager > BL > MatchServices > GetMatchesOfTheDay : renvoie une liste ne comprenant que les matchs dont la date est égale à la date du jour
CF-RV-009	Un match est organisé avec 5 joueurs minimum et 7 joueurs maximum
	DB > Inscription_Matches > déclencheurs > TR_InscriptMatches_CheckMax7JrParEq : empêche l'insertion de plus de 7 joueurs pour une équipe pour un match
	VS > MatchManager > UI > Inscription > frmInscription > btnInscrire_Click + btnDesinscrire_Click : effectuent une vérification du seuil min/max de joueurs (via service SeuilMinJoueurs_OK / SeuilMaxJoueurs_OK) avant de valider l'opération de modification d'inscription

	<i>VS > MatchManager > UI > Inscription > frmInscription > btnSave_Click : affiche un message d'avertissement de manque de joueurs si le seuil min n'est pas atteint. La sauvegarde n'est pas empêchée si le seuil minimum n'est pas atteint car il se peut qu'il n'y ait pas de joueur disponible à cause de sanctions.</i>
CF-RV-010	<i>Lors de l'inscription d'un joueur dans la feuille de match, il doit y avoir une vérification de la possibilité d'inscrire ce joueur ou non (cartons)</i>
	<i>DB > Procédures stockées > MatchManagement_SP_SelectJoueursByEqp_AvailableForMatchInscription : la procédure ne renvoie que les joueurs d'une équipe qui sont disponibles pour un match donné en paramètre ; un joueur est disponible s'il est fait partie de l'équipe passée en paramètre, pour la saison du match et s'il n'est pas exclus pour le match passé en paramètre</i>
CF-RV-011	<i>Si une équipe n'a pas assez de joueurs pour jouer un match, à cause des exclusions dues aux cartes, elle perd le match par forfait. Si les 2 équipes présentent cette incapacité, elles perdent toutes les 2</i>
	<i>VS > MatchManager > UI > FeuilleDeMatch > frmFeuilleMatch > verifForfait_Load() : Cette fonction fait appel à MatchManager > BL > EqpMatchServices > VerifierSiEquipeForfaitMatch() qui renvoie 2 booléens pour signifier si l'équipe domicile ou visiteuse doit être déclarée forfait à cause du seuil minimum de joueurs inscrits au match non atteint. Si une seule des 2 équipes doit être déclarée forfait, la fenêtre se charge avec l'enregistrement d'événement désactivé, et les cases de résultats pré-remplies : Perdu/Forfait pour les 2 équipes si les 2 équipes n'ont pas assez de joueur Gagné pour l'autre équipe si l'une des 2 est forfait. (Le résultat reste modifiable manuellement car rien n'interdit d'enregistrer le mauvais résultat étant donné qu'ils sont modifiables ultérieurement sans modification de la feuille de match)</i>

Events

Code	Description
CF-RV-012	<i>Les cartons jaune ou rouge du tout dernier match joué du championnat n'ont aucun effet sur le quarter suivant</i>
	<i>Lors de l'enregistrement des événements de type carte : l'utilisateur enregistre un carton en enregistrant directement les suspensions parmi les matchs de la saison du match d'origine de la sanction : cf SQL : MatchManagement_SP_SelectMatchRestantsEqp() et VS : MatchManager > UI > FeuilleDeMatch > frmSuspensions > lstbxMatchs_Load(). Les matchs restants ou les matchs ne pouvant être insérés ont une valeur NULL pour le match de suspension ; et le programme ne permet que de sélectionner des matchs de suspension parmi les matchs de la saison en cours.</i>

Suspensions

Code	Description
CF-RV-013	<i>Si un joueur reçoit un carton jaune, exclusion de 1 match durant le semestre. Le joueur peut choisir de quel match il sera exclu.</i>
	<i>VS > MatchManager > FeuilleDeMatch > frmSuspensions : la form ne charge que les matchs restants de la saison, desquels le joueur n'est pas encore suspendu. Le match de suspension est au choix parmi les matchs restants disponibles</i>
	<i>DB > Procédures stockées > [sch_MatchManagement].[SP_Insert_CardAndSuspensions] : inscrit les matchs de suspensions et supprime l'inscription d'un joueur au match suspendu si le joueur y était inscrit</i>
CF-RV-014	<i>Si un joueur reçoit une carte rouge, exclusion de 3 matchs dès le match suivant.</i>
	<i>VS > MatchManager > FeuilleDeMatch > frmSuspensions > CheckIfNextMatchRule_AndAutoselect : Permet de vérifier si le carton sélectionné contient la valeur true pour TEvnt_Nxt_match. Si oui, cette fonction sélectionne automatiquement les x matchs suivants (x = le nb de matchs de suspension * le nb de cartons de ce type)</i>
	<i>DB > Procédures stockées > [sch_MatchManagement].[SP_Insert_CardAndSuspensions] : inscrit les matchs de suspensions et supprime l'inscription d'un joueur au match suspendu si le joueur y était inscrit</i>

CF-RV-015	<i>Si un joueur reçoit des cartes dans des matchs différents, les absences s'additionnent</i>
	<i>VS > MatchManager > FeuilleDeMatch > frmSuspensions > btn_SaveClick : vérification du nombre de matchs de suspension sélectionnés avant de permettre la sauvegarde de l'événement et des matchs de suspensions liés. Si les suspensions sont plus nombreuses que le nombre de matchs restants, tous les matchs restants sont automatiquement sélectionnés, et les suspensions sont enregistrées dans la DB avec une valeur NULL pour l'id du match de suspension lié à l'événement (cf MatchManager > BL > EventsServices > SaveCardEvents)</i>

Match

Code	Description
CF-RV-016	<i>Vérification de la validité des données entre l'envoi des données pour le formulaire et la réception des données pour écriture : colonne LastUpdate</i>
	<i>DB > Matchs > déclencheurs > Matchs_Update_SetLastUpdate : insère automatiquement la date de mise à jour lorsqu'une ligne est modifiée</i>
	<i>DB > Procédures stockées > [sch_BackEnd].[SP_ModifResult] : vérification que la date de dernière m à j des données entrées correspond avec celles de la db</i>
	<i>DB > Procédures stockées > [sch_BackEnd].[SP_InsertUpdateMatchClnr] : vérification que la date de dernière m à j des données entrées correspond avec celles de la db</i>
CF-RV-017	<i>Les matchs insérés automatiquement ne se jouent que les samedis ou dimanches</i>
	<i>VS > BackEnd > BL > CalendrierServices : fonctions CreateSsnDateList_Ssn et GenererDates_MatchsListe (la 1ère crée une liste de dates utilisables en fonction du jour, si c'est un samedi ou un dimanche, la 2e insère aléatoirement un des 2 jours d'un we dans un match)</i>
CF-RV-018	<i>Une équipe ne peut jouer qu'un match par week-end</i>
	<i>VS > BackEnd > BL > CalendrierServices : fonctions CreateSsnDateList_Ssn et GenererDates_MatchsListe (la 1ère crée une liste de dates utilisables en fonction du jour, si c'est un samedi ou un dimanche, la 2e insère aléatoirement un des 2 jours d'un we dans un match, si l'insertion n'est pas possible un jour de w-e car l'équipe joue déjà un des 2 jours, le match ne comprend pas de date et sera donc dans la liste des matchs à insérer manuellement).</i>
CF-RV-019	<i>Tout match non inséré automatiquement devra être inséré manuellement, un autre jour de la semaine que le week-end, faisant partie de la saison en cours</i>
	<i>VS > BackEnd > UI > frmModifCalendrier > dtTmPckrMatch_CheckValue : vérifie que la date sélectionnée n'est pas un jour de week-end</i>
	<i>VS > BackEnd > UI > frmModifCalendrier > dtTmPckrMatch_Load : balise la sélection de date entre le début et la fin de la saison</i>
CF-RV-020	<i>La date d'un match ne peut plus être modifiée si cette date est antérieure à la date d'aujourd'hui</i>
	<i>VS > BackEnd > UI > frmCalendrier > gridClnr_DoubleClick : compare la date d'aujourd'hui avec la date du match ; si la date du match est antérieure à la date du jour : pas de modification calendrier possible pour ce match</i>
CF-RV-021	<i>Une équipe ne peut jouer qu'un seul match par jour</i>
	<i>VS > BackEnd > BL > CalendrierServices > SelectedDateMatchPossible : reçoit la liste des matchs où une des deux équipes joue le jour sélectionné ; si la liste est vide (count=0) alors on peut placer le match ce jour-là, sinon (count>0) on ne peut pas placer de match ce jour-là</i>
CF-RV-022	<i>Lors de l'encodage de la feuille de match, l'utilisateur peut sélectionner d'autres types de résultats que les résultats réels des goals du match.</i>
	<i>VS > MatchManager > FeuilleDeMatch > frmFeuilleMatch > btnResultSave_Click : lorsque l'utilisateur clique pour sauver les résultats sélectionnés, le programme vérifie la justesse des résultats sélectionnés par rapport aux résultats des goals ou forfaits, s'ils ne correspondent pas, un message d'avertissement demande confirmation à l'utilisateur. Interprétation personnelle : ceci a été décidé du fait qu'il est possible de modifier ultérieurement les résultats des matchs via le BackEnd, sans vérification de la validité avec les événements enregistrés.</i>

Règles de calcul

Durée d'une saison

Code	Description
CF-RC-001	Une saison de championnat dure 5 semaines
	VS > Models > MdlChampionnat : Propriété statique : SsnWeeks = 5

Calcul du classement

Code	Description
CF-RC-002	Un match gagné vaut 3 points pour le calcul du classement
CF-RC-003	Un match nul vaut 1 point pour le calcul du classement
CF-RC-004	Un match perdu, perdu/forfait vaut 0 point pour le calcul du classement

Les points sont insérés dans la DB dans la table « Types_Resultats ».

Pour afficher le classement : DB > Procédures stockées > [sch_BackEnd].[SP_SelectClassement] : la procédure crée une table comprenant l'id d'équipe, le nom d'équipe, et la somme des points via un groupBy, et renvoie une table classée par ordre descendant sur les points.

Calcul des types de résultats

Code	Description
CF-RC-005	Un match est déclaré « Gagné » pour une équipe si elle a marqué plus de goals que l'équipe adverse, ou si l'équipe adverse a perdu par forfait
CF-RC-006	Un match est déclaré « Perdu » pour une équipe si elle a marqué moins de goals que l'équipe adverse
CF-RC-007	Un match est déclaré « Nul » pour les 2 équipes si elles ont toutes les 2 marqués le même nombre de goals
CF-RC-008	Un match est déclaré « Perdu/Forfait » pour une équipe si elle n'a pas inscrit le seuil minimum de joueurs requis pour un match (5). 2 équipes peuvent être déclarées « Perdu/Forfait » pour un même match.

- VS > MatchManager > BL > EqpMatchServices > VerifierSiEquipeForfaitMatch() : vérifie si l'équipe est forfait pour le match ;
- VS > MatchManager > UI > frmFeuilleMatch > verifForfait_Load() + boxResults_DefaultSelection() : vérifie si l'équipe est forfait pour le match, sélectionne les types de résultats en fonction des scores et du/des forfait(s)

Calcul des types Events

Code	Description
CF-RC-009	Un type événement « Goal » vaut +1 point pour l'équipe du joueur dans le calcul des scores
CF-RC-010	Un type événement « Goal contre son camp » vaut - 1 point pour l'équipe du joueur et +1 point pour l'équipe adverse dans le calcul des scores
CF-RC-011	Un type événement « Carton jaune » vaut +1 jour (match) de suspension, sans obligation d'application dès le match suivant
CF-RC-012	Un type événement « Carton rouge » vaut +3 jours (matches) de suspension, avec obligation d'application dès le match suivant

- Goals :
DB > Procédures stockées > [sch_MatchManagement].[SP_SelectMatchScores] : renvoie les scores calculés pour un match
- Cartons :
Les nombres de jours et l'obligation d'application sont enregistrées dans la table Types_Events ;
VS > MatchManager > FeuilleDeMatch > frmSuspensions : la form ne charge que les matchs restants de la saison, desquels le joueur n'est pas encore suspendu. Le match de suspension est au choix parmi les matchs restants disponibles ;
VS > MatchManager > FeuilleDeMatch > frmSuspensions > CheckIfNextMatchRule_AndAutoselect : Permet de vérifier si le carton sélectionné contient la valeur true pour TEvt_Nxt_match. Si oui, cette fonction sélectionne automatiquement les x matchs suivants (x = le nb de matchs de suspension * le nb de cartons de ce type)

Description des entités

<i>Nom</i>	<i>Description</i>
Championnat	<i>Entité regroupant les informations d'un championnat</i>
Equipes	<i>Entité regroupant les informations d'une équipe</i>
Constitution_Championnat	<i>Entité regroupant les inscriptions d'équipes aux championnats. Une entité constitution_championnat représente une inscription d'une équipe à un championnat. Cette entité lie l'entité championnat avec l'entité équipe.</i>
Saisons	<i>Entité regroupant les informations des saisons de championnat</i>
Joueurs	<i>Entité regroupant les informations des joueurs</i>
Constitution_Equipes	<i>Entité regroupant les inscriptions des joueurs aux équipes. Une entité constitution_equipe représente une inscription d'un joueur à une équipe, pour une saison déterminée. Cette entité lie l'entité Joueurs avec l'entité Equipes.</i>
Inscriptions_Matches	<i>Entité regroupant les inscriptions d'un joueur, inscrit à une équipe, à un match. Une entité inscription_match représente une inscription d'un joueur, inscrit dans une équipe, à un match. Cette entité lie l'entité constitution_equipe avec l'entité Matches.</i>
Matches	<i>Entité regroupant les informations d'un match.</i>
Events	<i>Entité regroupant les informations d'un événement avec un joueur inscrit à un match. Une entité événement représente un type d'événement réalisé par un joueur d'une équipe, inscrit à un match. Cette entité lie l'entité inscription_match avec l'entité type_event.</i>
Types_Events	<i>Entité regroupant les différents types d'événement, et donc les informations générales de ceux-ci, dont le nombre de matchs de suspensions, et l'obligation ou non de chronologie.</i>
Suspensions	<i>Entité regroupant les suspensions effectuées ou à effectuer suite à un événement de type carton. Une entité suspension représente un match de suspension suite à un événement de type carte.</i>
Types_Resultats	<i>Entité regroupant les différents types de résultats, et donc les informations générales de ceux-ci, dont les points.</i>

Schéma relationnel de la solution

Schéma EA

NB : J'ai créé 2 versions du schéma EA pour ce projet. La version ci-dessous est celle que j'ai choisi par clarté, mais je mets également l'autre version disponible en [annexe](#), afin de permettre une meilleure compréhension des relations entre entités. La différence entre ces 2 schémas est le simple choix entre une entité simple ou une entité d'association.

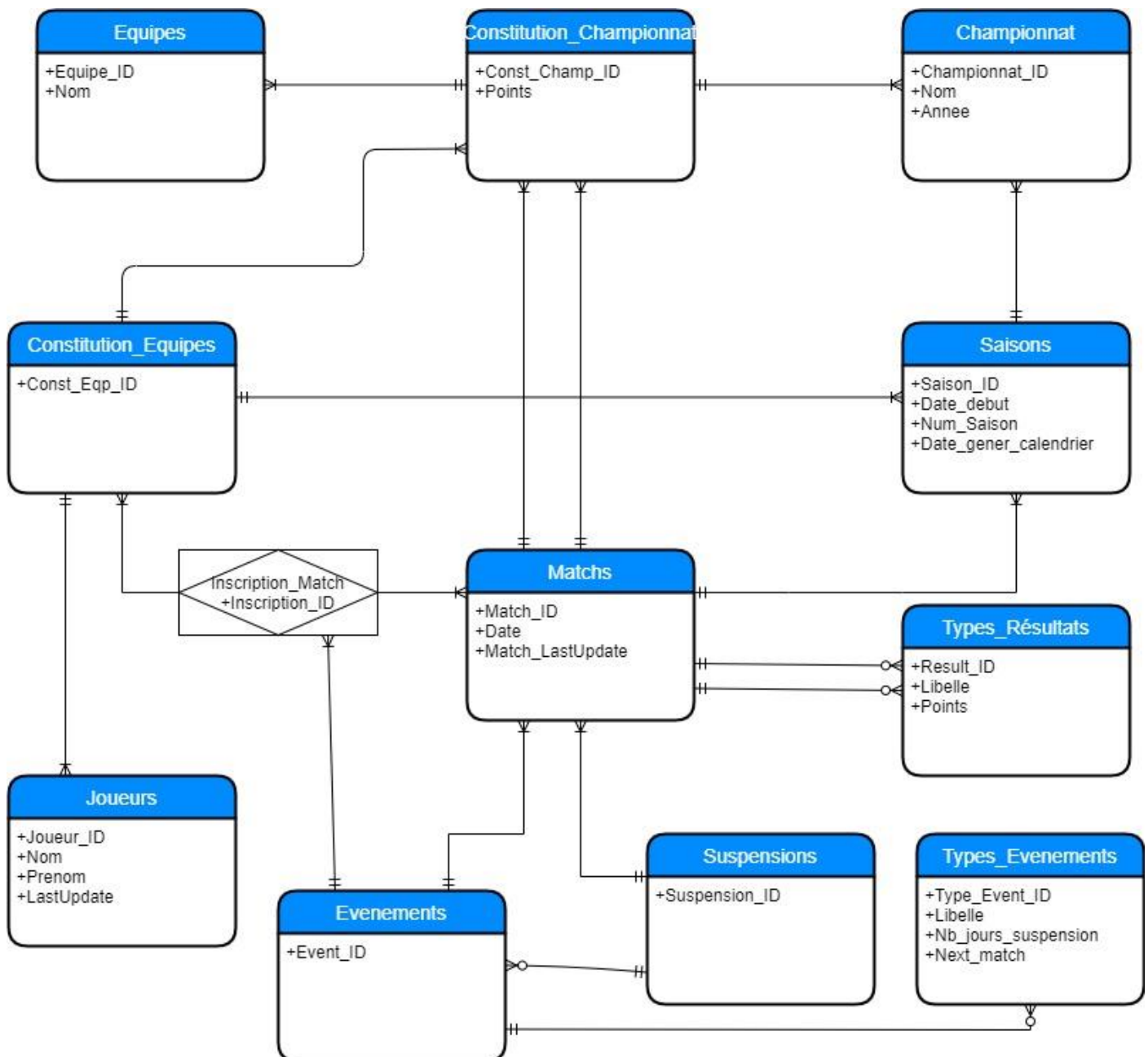
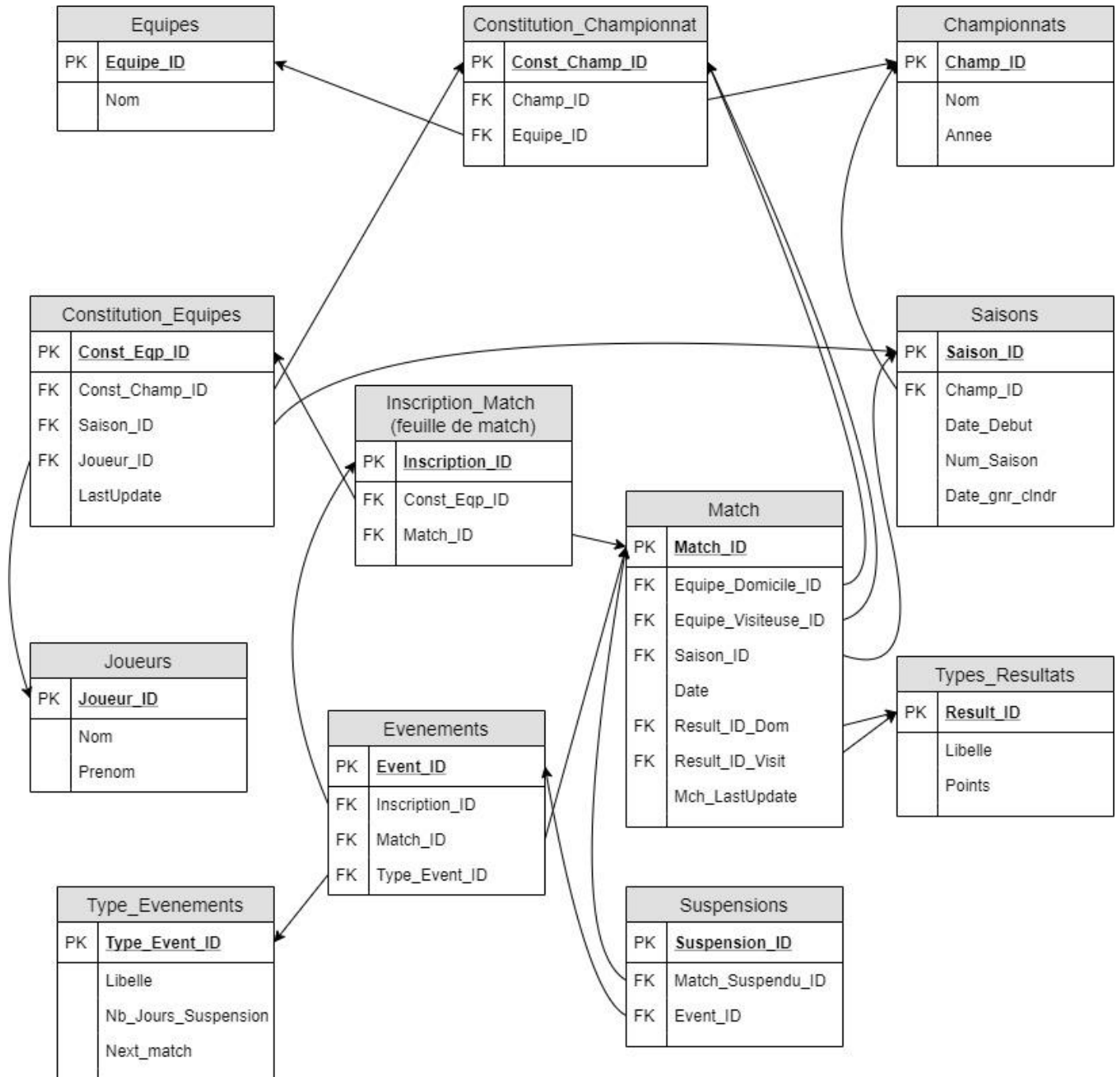


Schéma relationnel



Implémentation des contraintes

Cf [Contraintes fonctionnelles](#) (et Use Cases) : les implémentations y sont directement citées.

Analyse technique

Technologies proposées

- ❖ La solution (back-end) doit être développée en .NET, avec une architecture de découpage en couches ;
- ❖ La base de données doit être réalisée en SQL Server ;
- ❖ Le programme BackEnd doit se connecter à la base de données via Entity Framework ;
- ❖ Le programme MatchManager doit se connecter à la base de données en ADO.NET ;
- ❖ Le front-end (client) doit être réalisé en Windows Forms.

Architecture applicative

Une architecture de découpage en couches se compose de 3 couches :

1. La couche UI (ou GUI : User Interface)
2. La couche BL (ou Business Layer)
3. La couche DAL (ou Data Access Layer)

La couche UI/GUI

Il s'agit de la couche avec laquelle l'utilisateur interagit directement. Cette couche peut faire appel à tout ce qui fait partie de la couche BL, et est construite, dans ce projet, en WindowsForms.

Elle ne comprend pas de logique Business, ou de logique d'accès aux données (SQL).

La couche BL (Business Layer)

Il s'agit de la couche gérant tous les services du programme. Elle interprète les interactions de l'utilisateur, et les met en lien avec la DAL, couche d'accès aux données. Elle permet de traiter les données brutes et les utiliser ou transformer en fonction des besoins de l'utilisateur dans l'UI.

Elle permet également de faire le lien entre les Modèles de notre programme et les entités de la base de données, et de travailler avec ces modèles.

Elle ne comprend pas de logique d'accès aux données (SQL).

Dans ce projet, elle est développée en .NET.

La couche DAL (Data Access Layer)

Il s'agit de la couche qui interagit entre nos services (la couche BL) et la base de données. Elle permet de récupérer les informations brutes de la base de données et les renvoyer vers la couche de services (BL), qui va pouvoir traiter et transformer les informations nécessaires.

C'est donc la seule couche qui contient une logique d'accès aux données, mais elle ne fait que les envoyer ou recevoir, sans appliquer de traitement, ni modification.

Dans ce projet, elle est développée en Entity Framework pour le programme BackEnd, et en ADO.NET pour le programme MatchManager.

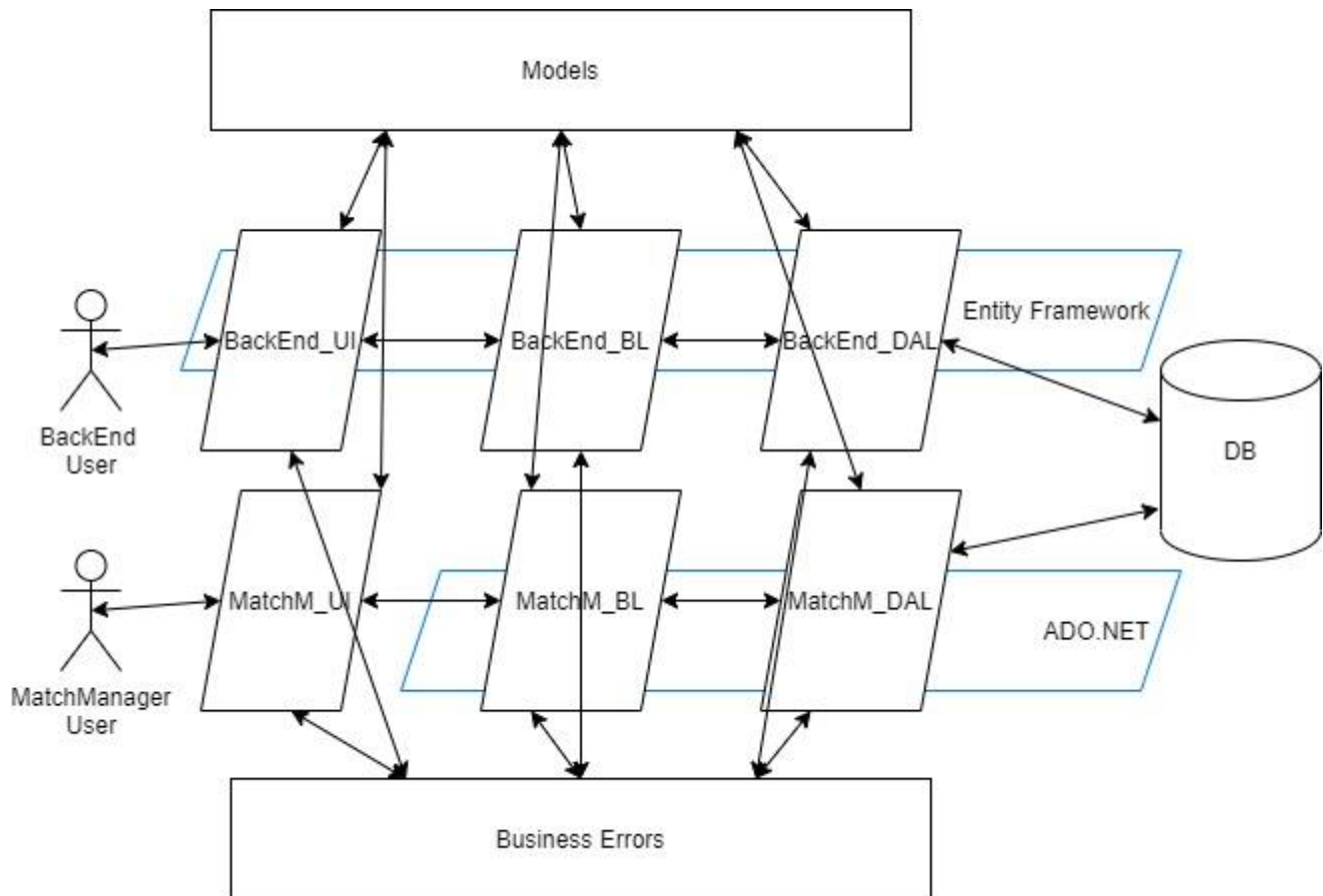
Les Modèles et Erreurs

Ce sont 2 éléments distincts des couches, accessibles par les 3 différentes couches.

Les Modèles permettent d'utiliser les entités de la base de données, adaptées aux besoins de notre application.

Les Erreurs Business, permettent de gérer de manière personnalisée les erreurs pouvant survenir durant le déroulement des opérations des programmes. On peut donc créer nos propres modèles d'erreurs en fonction des besoins/contraintes business, mais également récupérer et transformer une exception de programmation avant (et pour ne pas) qu'elle n'arrive chez l'utilisateur.

Architecture de ce projet



Algorithmes spécifiques

Calcul de classement

1. Créer une table temporaire ayant pour colonnes, au minimum : **Equipe_ID, SUM(Res_Points) AS Points** ;
2. Ajouter dans la table la sélection identique tirée de la table Matches jointe aux autres entités nécessaires (telle que Types_Resultats pour avoir les points) avec :
Mch_Eqp_Dom_ID as Eqp_ID, Eqp_Nom, SUM(Res_Points) AS Points ;
3. Puis faire la même chose mais avec :
Mch_Eqp_Visit_ID as Eqp_ID, Eqp_Nom, SUM(Res_Points) AS Points ;
4. Grouper la table temporaire par **Equipe_ID**, et, au besoin, l'ordonner dans le sens désiré sur le nombre de points.

DB > Procédures stockées > [sch_BackEnd].[SP_SelectClassement]

Calcul de scores

GE = total des goals des joueurs de l'équipe ;

GCCE = total des goals contre son camp des joueurs de l'équipe ;

GCCA = total des goals contre son camp des joueurs de l'équipe adverse ;

$$\text{Score total d'une équipe} = \text{GE} - \text{GCCE} + \text{GCCA}$$

DB > Procédures stockées [sch_MatchManagement].[SP_SelectMatchScores]

Création de matchs aller-retour

Matchs aller : première saison

1. Commencer par la **première équipe en tant qu'équipe domicile**
2. Créer un match vs **chaque équipe suivante** en paramétrant chaque équipe suivante en **équipe visiteuse**
3. Recommencer avec la 2^e équipe en tant qu'équipe domicile, recommencer les point 2 & 3 **jusqu'à l'avant-dernière équipe incluse en tant qu'équipe domicile**.
4. Une fois arrivé à l'avant dernière équipe, il ne reste que le match de cette équipe vs la dernière équipe à ajouter.

Exemple avec 4 équipes

	1 (Visit)	2 (Visit)	3 (Visit)	4 (Visit)
1 (Dom)	/	1 vs 2	1 vs 3	1 vs 4
2 (Dom)	/	/	2 vs 3	2 vs 4
3 (Dom)	/	/	/	3 vs 4
4 (Dom)	/	/	/	/

Ainsi chaque équipe affronte une première fois chaque autre équipe.

VS > BackEnd > BL > CalendrierServices > CreateMatches_FirstSsn()

Matchs retour : seconde saison

1. Commencer par la **dernière équipe en tant qu'équipe domicile**
2. Créer un match vs **chaque équipe précédente** en paramétrant chaque équipe précédente en **équipe visiteuse**
3. Recommencer avec **l'avant-dernière équipe** en tant qu'équipe domicile, recommencer les point 2 & 3 **jusqu'à la 2^e équipe incluse en tant qu'équipe domicile**.
4. Une fois arrivé à la 2^e équipe, il ne reste que le match de cette équipe vs la première équipe à ajouter.

Exemple avec 4 équipes

	1 (Visit)	2 (Visit)	3 (Visit)	4 (Visit)
1 (Dom)	/	/	/	/
2 (Dom)	2 vs 1	/	/	/
3 (Dom)	3 vs 1	3 vs 2	/	/
4 (Dom)	4 vs 1	4 vs 2	4 vs 3	/

Ainsi chaque équipe affronte une première fois chaque autre équipe.

VS > BackEnd > BL > CalendrierServices > CreateMatches_SecondSsn()

Génération automatique de calendrier

Ordre à suivre, par saison :

1. **Créer une liste de matchs** à insérer dans un calendrier, en fonction de la saison (1 ou 2) ;
 - VS > BackEnd > BL > CalendrierServices > CreateMatches_FirstSsn()
 - VS > BackEnd > BL > CalendrierServices > CreateMatches_SecondSsn()
2. **Créer une liste des dates possibles** (week-ends) d'insertion des matchs durant la saison ;
 - VS > BackEnd > BL > CalendrierServices > CreateSsnDateList_Ssn() : reçoit une date de début de saison et crée une liste de jours (samedi & dimanche) compris dans les 5 semaines suivant la date de début donnée en paramètre
3. **Insérer les dates disponibles dans les matchs**, de manière aléatoire, en respectant le fait qu'une équipe ne joue qu'une fois par week-end ;
 1. **prendre le premier week-end** de la liste de dates disponibles (si c'est un dimanche, ne prendre que ce jour-là) ;
 2. **calculer un nombre max de matchs par week-end** (nombre d'équipes / 2) ;

3. **tant que** : le nombre max de matchs par week-end n'est pas atteint, et qu'il reste des matchs à insérer dans le calendrier, et qu'il est possible d'insérer des matchs en respectant les conditions d'un match par week-end par équipe, dans ce week-end :
 - i. **choisir une des 2 dates du week-end au hasard** ;
 - ii. **choisir un match au hasard** ;
 - iii. si aucune des 2 équipes ne joue ce week-end là : **ajouter le match** à la liste des matchs avec dates et le supprimer de la liste des matchs à dater ;
sinon recommencer à l'étape 3.3.1 ;
 - iv. **recommencer à l'étape 3.3.1 jusqu'à ce qu'aucun autre match ne puisse être inséré** durant ce week-end ;
sinon **recommencer à l'étape 3.1 avec le week-end suivant** de la liste, jusqu'à ce qu'il n'y ait plus de match à insérer ou de week-end disponible ;
4. **Laisser une date nulle pour tout match n'ayant pu être inséré automatiquement** dans le calendrier des week-ends de la saison ;

 *VS > BackEnd > BL > CalendrierServices > GenererDates_MatchesListe()*

Annexes

Enoncé examen (source moodle : Prof. V. Fievez)

Examen est composé de 3 parties

- Cahier des charges
- Base de données
- Programme

Inscription à l'examen est la remise de l'ensemble dans 1 devoir.

Examen est une discussion (15-20 mn) sur ce que vous avez fait... Vous devez apporter votre analyse version papier et une machine qui tourne votre programme.

Vous pouvez travailler à 2 max sur le projet mais la défense est individuelle.

Utiliser un gestionnaire de source et me mettre actif dans ce gestionnaire.

FifaManager

Gérer un championnat de Soccer... Un championnat est basé sur une année civile... et est divisé en 2 temps (semestre)...

Chaque championnat est composé de X équipes... C'est une variable pour chaque championnat...

Une équipe est composée au minimum de 5 joueurs et maximum 10.

Je dois pouvoir retrouver l'historique des championnats et matches et résultats d'un joueur.

Dans un championnat, le joueur ne peut jouer que dans une équipe. Mais il peut changer d'équipe à l'intersaison (entre 2 Q). Quand c'est un autre championnat, pas de limite de changement mais si intersaison, il ne peut changer que dans les 3 dernières équipes.

Si une équipe n'a que 5 joueurs, pas de sortie possible.

Un match possède un résultat :

- Gagné : 3 points
- Perdu : 0
- Nulle : 1 points

Un match est organisé avec 5 joueurs au minimum et 7 au maximum. Il faut donc une feuille de match.

Si lors de ce match, un joueur reçoit un carton jaune, exclusion de 1 match durant le semestre. Si carte rouge, exclusion de 3 matches dès le match suivant.

Si carton jaune ou rouge au dernier match, on annule tout puisque on redémarre le semestre suivant...

Si on reçoit des cartes dans des matches différents, les absences s'additionnent.

On doit aussi encoder qui a marqué les goals...

Si à cause des cartes, une équipe n'a pas assez de joueur, elle perd d'office.

Lors d'une inscription d'un joueur dans la feuille de match, il doit y avoir une vérification de la possibilité d'inscrire le joueur...

Les matches ne se jouent que les samedis et dimanches. Un planning des matches doit être généré automatiquement et modifiable... Si des matches ne peuvent pas être insérés automatiquement, ils doivent être insérés dans les autres de la semaine manuellement.

C'est un quoi un Q : C'est 5 semaines obligatoires...

2 programmes à faire :

Le premier « Backend » permet de gérer les championnats, équipes,... bref tout...

Le deuxième programme « MatchManagement » permet uniquement de gérer les feuilles de match... Joueurs et qui a gagné et les cartes.

D'un point de vue technique :

- Des tables sans accès par personne sauf sa
- Des stored procedure et views avec schémas
- Sécurité au niveau du des schémas.
 - 2 schémas : Backend et l'autres MatchManagement...
- Découpe obligatoire en BL, DAL, Modèles et Erreurs...
- Le backend doit utiliser EF et MatchManagement ADO.NET
- 2 users SQL, le premier Backend et l'autres MAtchManagement...

Fonctionnalités obligatoires dans le Backend

- Inscrire des joueurs dans une équipe et les limites...transfert Q1 et Q2
- Génération des horaires des matches et la possibilité de les modifier.
- Possibilité de voir les résultats et les modifier ainsi que visualiser les cartes et les inscriptions des joueurs au matche.
- Visualiser le classement

Fonctionnalités obligatoires du MatchManagement

- Inscrire les joueurs pour le matche + contrôles
- Donner le résultat et les cartes et les goals.

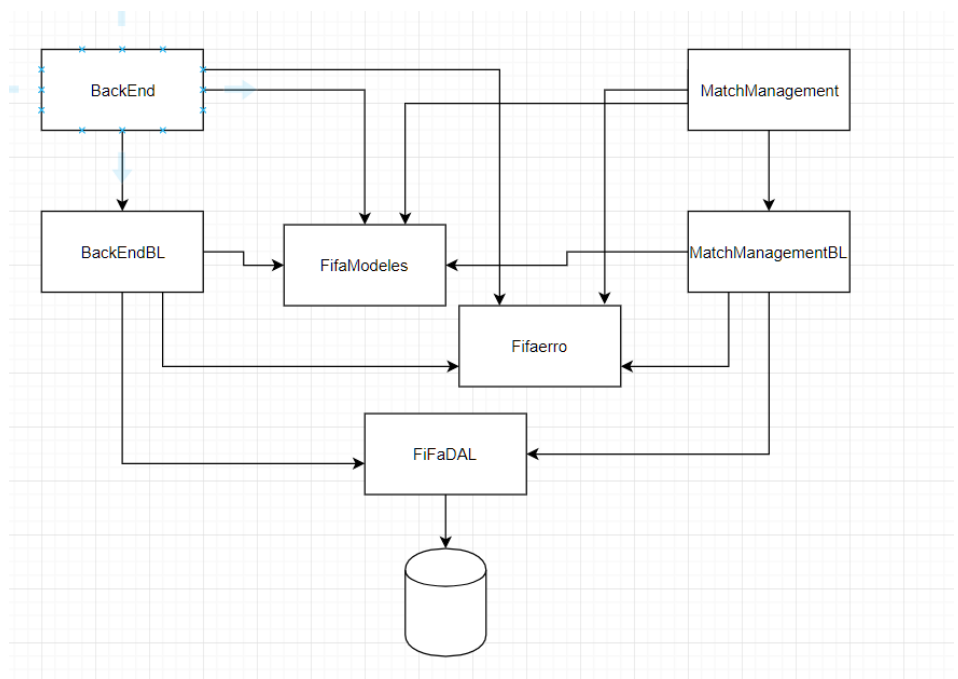
Le cahier des charges

- L'énoncé du projet compris par vous
- Analyse des cas business par exemple avec des mookup d'écrans, avec description complète de ce que cela fait.
- Liste des règles et contrôles du projet :
 - R1 : un joueur ne peut être inscrit que dans une équipe par championnat...
 - ...

Ces règles sont les erreurs qui doivent être gérées par votre gestion d'erreur.
- Schéma E-A
- Lister où sont implémentés les erreurs business
 - R1 est implémenté dans le trigger x et dans le code là...

1 solution avec 7 projets

- 2 windows forms
- 5 class libraries



Interprétation personnelle de l'énoncé

Afin de pouvoir démontrer au mieux mon interprétation personnelle de l'énoncé du projet, j'ai choisi de séparer les informations relatives aux règles et fonctionnement d'un championnat, de celles directement relatives aux fonctionnalités attendues (explicitées dans l'énoncé uniquement, non dans la partie technique).

Règles & fonctionnement d'un championnat

Un championnat est basé sur une année civile. Il est divisé en 2 saisons, nommées également « Quarters » (/Q).

Chaque quarter/saison, se déroule en 5 semaines obligatoirement, ni plus, ni moins. Le 1^{er} quarter représente les matchs aller, et le 2^e représente les matchs retour. Les 2 quarters d'un championnat se déroulent durant une même année civile.

Chaque championnat est composé d'un nombre variable d'équipes. Pour participer au championnat, chaque équipe doit s'inscrire. Pour que cette inscription soit valide, l'équipe doit être composée de 5 joueurs minimum, jusqu'à un maximum de 10 joueurs.

Un joueur ne peut être inscrit que dans une seule équipe par championnat.

Des changements (transferts) de joueurs sont possibles durant les championnats, mais en respectant 3 conditions :

1. l'équipe que le joueur va quitter doit avoir plus de 5 joueurs inscrits. Tout joueur faisant partie d'une équipe n'ayant que 5 joueurs participants n'est pas transférable ;
2. le changement n'est possible que durant l'intersaison (càd entre 2 Q) ;
3. le joueur ne peut être transféré que dans une des 3 dernières équipes du classement en cours.

Ces limitations de transfert ne sont valables que pour les championnats en cours. Tout championnat dont la 1^{ère} saison n'a pas débuté n'est pas concerné par ces limites de transfert, excepté la 1^{ère} (équipe avec seulement 5 joueurs inscrits).

Chaque saison de championnat se joue en matchs où 2 équipes s'affrontent. Chaque confrontation se passe en un match aller et un match retour. Chaque équipe est donc une fois l'équipe qui reçoit à domicile, et une fois l'équipe visiteuse.

Chaque match possède un résultat final :

- ❖ Gagné = 3 points pour le classement du championnat
- ❖ Perdu = 0 points pour le classement du championnat
- ❖ Nul = 1 point pour le classement du championnat

Pour qu'un match soit joué, il faut que 5 joueurs soient inscrits (disponibles) dans chaque équipe au minimum. Une équipe peut inscrire jusqu'à 7 joueurs au maximum pour un match. Si une des 2 équipes n'a pas assez de joueurs disponibles pour jouer le match, elle perd le match par forfait, sans que le match ne soit joué.

Les matchs ne se jouent que les samedis et dimanches. Chaque équipe ne pouvant jouer qu'une seule fois par week-end. Le planning est généré automatiquement par le programme informatique et est modifiable manuellement. Tout match n'ayant pu être inséré automatiquement durant les 5 w-e d'une saison sera listé par le programme et devra être inséré manuellement durant les autres jours de la semaine (lu-ve), faisant partie des 5 semaines de la saison. Durant un match, des joueurs peuvent être sanctionnés en commettant des fautes. En fonction de la gravité de la faute, le joueur peut être pénalisé par un carton jaune ou rouge. Ces cartons sont cumulables et non limités pendant un même match, et toute la durée du championnat.

Chaque type de carton implique une sanction en fonction de sa couleur :

- ❖ carton jaune : exclusion de 1 match, au choix, durant la saison en cours ;
- ❖ carton rouge : exclusion des 3 prochains matchs de la saison en cours.

Ces sanctions peuvent s'additionner au fil des matchs. Elles sont remises à chaque début de saison. Les sanctions du dernier match d'une saison ne sont donc jamais exécutées.

Si, à cause de ces sanctions, une équipe n'a pas assez de joueurs disponibles (5) pour jouer un match, elle perd le match par forfait. Si les 2 équipes présentent cette incapacité, elles perdent toutes les 2.

Pour chaque match, une feuille de match reprendra les informations du match : les joueurs inscrits, les cartons (jaune/rouge) donnés et les goals avec, pour chacun, le joueur lié à l'événement.

Attentes relatives au programme, extraites de l'énoncé original & interprétées

Sur base de l'énoncé uniquement, pas des fonctionnalités techniques & obligatoires détaillées plus loin.

- ❖ Il faut pouvoir retrouver l'historique des championnats, matchs et résultats d'un joueur.
- ❖ On doit pouvoir encoder qui a marqué les goals.
- ❖ Lors d'une inscription d'un joueur dans la feuille de match, une vérification doit avoir lieu pour permettre ou non cette inscription : le joueur doit remplir les conditions nécessaires à l'inscription (pas de sanction, faire partie d'une équipe inscrite...)
- ❖ Le programme devra pouvoir générer automatiquement un planning modifiable manuellement. Tout match ne pouvant être inséré automatiquement durant les 5 w-e d'une saison devra être listé et inséré manuellement durant les autres jours de la semaine de cette même saison.

Structure du cahier des charges

1. Contexte
 1. Cadre
 2. Projet
2. Analyse métier
 1. Description de la solution envisagée
 2. Intervenants
3. Fonctions attendues
 1. Exigences fonctionnelles
 - i. Backend
 - Ef-bk-001...
 - EF-bk-002...
 2. Exigences non fonctionnelles
4. Contraintes business (techniques)
5. Analyse fonctionnelle
 1. Diagrammes Use Case
 - Uniquement les plus important...
 2. Diagrammes d'activité
 3. Diagrammes d'état
6. Contraintes fonctionnelles
 1. Règles d'accès et autorisation
 2. Règles de structure
 3. Règles de validation
 4. Règles de calcul
7. Description des entités
8. Schéma relationnel de la solution
 1. Schéma E-A
 2. Implémentations des contraintes
 - Reprendre les principales contraintes et dire où ou comment elles sont implémentées
9. Analyse technique
 1. Technologies proposées
 2. Architecture applicative

