

Projet IBDW

1. Introduction : choix techniques

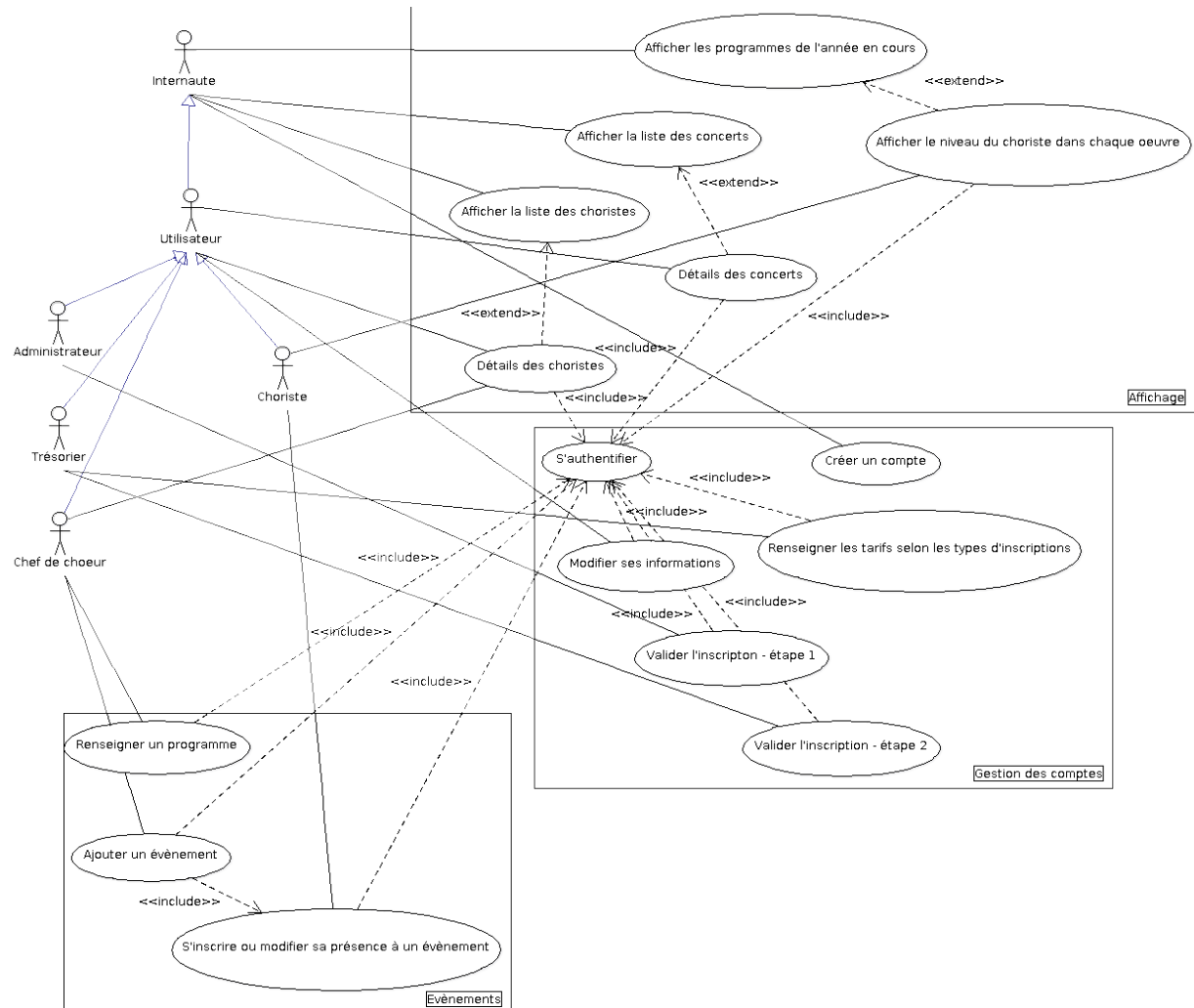
L'objectif est de développer une application web de gestion d'un ensemble musical. Pour ce premier rendu, l'objectif était de modéliser l'intégralité de la base de données et la logique d'affichage des données telle que décrite dans le sujet.

Nous avons considéré que l'utilisateur de base est l'internaute connecté au site web. Il peut effectuer les actions d'affichage de base d'une liste de choristes, d'évènements et de programmes (ensemble d'oeuvres pour chaque évènement), ainsi que créer un compte dans l'application.

Une fois son compte créé et validé par l'administrateur du site, il peut se connecter au site (affichage simplifié des listes de choristes et d'évènements...). Pour accéder à l'ensemble des fonctionnalités du site (affichage plus détaillé avec visualisation des présences, inscription à des évènements, modifications de ses coordonnées, etc...), le compte utilisateur doit également être validé par le trésorier.

Cet utilisateur peut prendre le rôle de choriste et de membre de l'administration de l'association le cas échéant.

2. Cas d'utilisation



Note :

- L'affichage du niveau pour chaque œuvre sera fait en fonction du niveau global de la chorale, et non du niveau individuel du choriste qui affiche la liste ;

3. Descriptions textuelles de la procédure d'affichage

Cette section reprend une description textuelle des traitements pour chaque cas d'utilisation de la section « Affichage » du diagramme de cas d'utilisation précédent.

3.1 Afficher la liste des événements

Acteurs principaux : Internaute, Utilisateur

Acteurs secondaires : néant

Objectif : Permettre à l'internaute de visualiser les événements passés et à venir, ordonnés par ordre chronologique.

Si l'internaute est un utilisateur authentifié, pour chaque concert, un voyant s'affiche :

- vert si suffisamment de choristes sont inscrits pour cet événement ;
- rouge sinon.

Préconditions : néant

Postconditions : néant

Scénario nominal :

1. L'internaute clique sur l'onglet «Événements»
2. La liste des 30 premiers concerts à venir et des 30 événements passés les plus récents s'affiche.

Alternatives :

- S'il n'y a pas d'événements, le texte « Aucun événement à afficher » est affiché.

Requêtes SQL :

-- Affichage des répétitions

```
SELECT *  
FROM Evenement  
WHERE idType=2;
```

-- Affichage des concerts

```
SELECT nom, heureDate, lieu  
FROM Evenement  
NATURAL JOIN Participe  
WHERE idType=2  
ORDER BY heureDate;
```

-- S'il est authentifié et que le il s'agit d'un concert à venir :
-- On doit vérifier s'il y a au moins 2 choristes par voix

```
SELECT typeVoix  
FROM Voix  
LEFT JOIN Choriste  
ON (Voix.idVoix = Choriste.idVoix)  
NATURAL JOIN participe  
NATURAL JOIN Evenement  
WHERE Evenement.idEvenement = 1  
GROUP BY typeVoix  
HAVING count(Choriste.idChoriste) <= 1;
```

-- Deuxième tentative

```
SELECT Evenement.nom, heureDate, lieu, idVoix,
```

```
count(Choriste.idChoriste) as countc
FROM Evenement
LEFT JOIN participe ON Evenement.idEvenement =
participe.idEvenement
LEFT JOIN Choriste ON participe.idChoriste = Choriste.idChoriste
GROUP BY idVoix, Evenement.nom, heureDate, lieu;
```

Maquettes :

Liste des concerts

Concerts à venir (30 résultats maximum)

date	heure	lieu	<i>présence</i>
30/03/2014	16h	<u>Evry</u>	✓
29/03/2014	16:00	<u>Evry</u>	✓
28/03/2014	16:00	<u>Evry</u>	✓

[Dates précédentes](#)[Dates suivantes](#)

Concerts passés (30 résultats maximum)

Date	Heure	Lieu	<i>Présence</i>
24/03/2014	16:00	<u>Evry</u>	✓
23/03/2014	16:00	<u>Paris</u>	✓
22/03/2014	16:00	<u>Corbeilles</u>	✓
21/03/2014	16:00	<u>Evry</u>	✓
20/03/2014	16:00	<u>Evry</u>	✓

[Dates précédentes](#)[Dates suivantes](#)

Dans le cas où il n'y a pas d'événements dans la base de données :

Liste des concerts
Concerts passés(30 résultats maximum)
Il n'y a pas de concerts passés
Concerts à venir (30 résultats maximum)
Il n'y a pas de concerts à venir

Jeux de tests :

Lorsqu'il n'y a pas d'événements, la page ci-dessous est affichée :

MyChorus	Accueil	Choristes	Evènements	Programme de l'année	Connexion
Liste des évènements					
Evènements à venir					
Nom	Date et heure	Lieu			
Evènements passés					
Nom	Date et heure	Lieu			
Une application réalisé par les étudiants FIPA 6 et FIP 19 de l'ENSIEE					

3.2 Afficher la liste des choristes

Acteurs principaux : Internaute, Utilisateur

Acteurs secondaires : néant

Objectif : Permettre à l'internaute de visualiser les choristes (nom, prénom et voix) inscrits sur le site.

Si l'internaute est un utilisateur authentifié, pour chaque choriste, les données suivantes sont affichées :

- nombre de répétitions auxquelles il a assisté et qu'il a manqué ;
- taux de présence aux répétitions.

La liste regroupe les choristes par type de voix et affiche le taux moyen de présence par type de voix.

Préconditions : néant

Postconditions : néant

Scénario nominal :

1. L'internaute clique sur l'onglet «Choristes»
2. La liste de l'ensemble des choristes s'affiche.

Alternatives :

- S'il n'y a pas de choristes, le texte « Aucun choriste à afficher » est affiché.

Requêtes SQL :

-- Affichage choristes avec nombre de participation

```
SELECT nom, prenom, typeVoix, sum(participe.confirmation) as
participations
FROM Choriste
NATURAL JOIN participe
NATURAL JOIN Voix
GROUP BY idChoriste, nom, prenom, typeVoix
ORDER BY typeVoix;
```

-- Affichage choristes avec taux de participations et responsabilité

```
SELECT nom, prenom, typeVoix, ville, telephone, titre as
responsabilite, avg(participe.confirmation) as participations
FROM Choriste
LEFT JOIN participe
ON Choriste.idChoriste = participe.idChoriste
NATURAL JOIN Voix
NATURAL JOIN Utilisateur
LEFT JOIN Responsabilite
ON Utilisateur.login = Responsabilite.login
GROUP BY Choriste.idChoriste, nom, prenom, typeVoix, ville,
telephone, responsabilite
ORDER BY typeVoix;
```

Maquettes :

Liste des choristes							
Choristes (30 résultats au maximum)							
Voix	Nom	Prenom	Responsabilités	Nombre de répétitions assistées	Nombre de répétitions manquées	Taux de présence	Coordonnées
Alto	Dupond	Henry	Chef de choeur	5	5	100	Marseille / 0752222222
Alto	Morand	Bertrand	Webmaster	3	2	175	Paris / 0633333333
Total alto						114	
Basse	Dupont	Martin	Tresorier	2	2	100	Toulouse / 0500000000
Basse	gree	Thomas	Tresorier	1	4	25	Bordeaux / 0199999999
Total Basse						50	
Soprano	row	Quentin		10	1	100	Caen / 0333322222
Soprano	green	Sebastien		4	1	40	Campiegne / 0235632145
Total Soprano						70	
Tenor	dio	Alphonse		2	3	75	Paris / 0603050409
Tenor	thi	thake		2	3	75	Paris / 0709080203
Total Tenor						75	

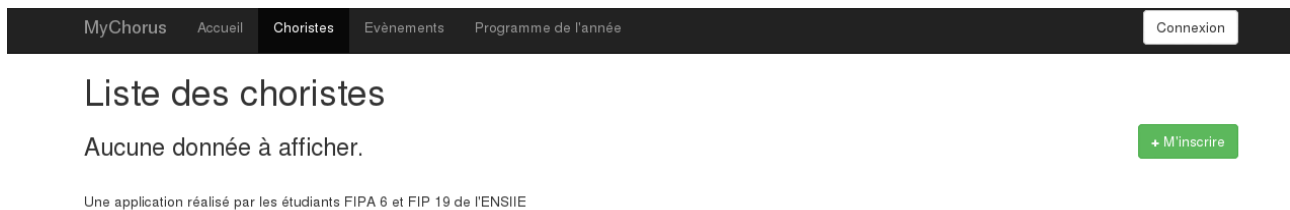
[Choristes précédents](#)[Choristes suivants](#)

Dans le cas où il n'y a pas de choristes dans la base de données :

Affichage des choristes
Choristes (30 résultats au maximum)
Il n'y a pas de choristes

Jeux de tests :

Lorsqu'il n'y a pas de choriste, le message « Aucune donnée à afficher » apparaît.



Lorsqu'un utilisateur n'était pas un choriste mais avait d'autres responsabilités, son nom et prénom n'étaient pas récupérés, voici ce qui s'affichait alors :



Ce problème a ensuite été résolu puisque l'on a ajouté tous les utilisateurs dans la table choriste.

3.2 Afficher la liste des programmes

Acteurs principaux : Internaute, Utilisateur

Acteurs secondaires : néant

Objectif : Permettre à l'internaute de visualiser les œuvres étudiées durant l'année courante. A la fin de chaque groupe d'œuvre

Si l'internaute est un utilisateur authentifié, pour chaque œuvre, un voyant s'affiche :

- vert si l'œuvre a été suffisamment étudiée pour être jouée en concert ;
- jaune si l'œuvre est en cours d'apprentissage ;
- rouge sinon.

Préconditions : néant

Postconditions : néant

Scénario nominal :

1. L'internaute clique sur l'onglet «Programme»
2. La liste de œuvres étudiées s'affiche.
3. La liste des styles étudiés avec la durée totale des œuvres de chaque style.

Alternatives :

- S'il n'y a pas d'œuvres, le texte « Aucune œuvre à afficher » est affiché.

Requêtes SQL :

```
-- Pour connaître les oeuvres au programme d'une année, le  
webmaster va ajouter un événement nommé 'Saison 2014-2015' relié à  
chacune de ces oeuvres
```

```
-- Utilisateur authentifié : indiquer si l'œuvre n'a pas encore  
été étudiée, est en cours d'apprentissage, ou peut être chantée en  
concert
```

```
SELECT Oeuvre.*  
FROM Oeuvre
```



```

NATURAL JOIN est_au_programme
NATURAL JOIN Evenement
NATURAL JOIN TypeEvt
WHERE typeEvt = 'Saison' AND nom = 'Saison 2014-2015';

```

```

SELECT idoeuvre,titre,auteur,partition,duree,style,
COUNT(idEvenement)
FROM Oeuvre
NATURAL JOIN est_au_programme
LEFT JOIN Evenement ON Evenement.idEvenement=Participe.idEvenement
NATURAL JOIN TypeEvt
NATURAL JOIN participe
WHERE typeEvt = 'Saison' AND nom = 'Saison 2014-2015' AND
idChoriste = 1 AND confirmation = 1
GROUP BY idoeuvre,titre,auteur,partition,duree,style ;

```

```

--La durée du programme par style doit pouvoir être calculée
SELECT style, SUM(duree) duree_annuelle
FROM Oeuvre
NATURAL JOIN est_au_programme
NATURAL JOIN Evenement
NATURAL JOIN TypeEvt
WHERE typeEvt = 'Saison' AND nom = 'Saison 2014-2015'
GROUP BY style;

```

Maquettes :

Programme de l'année (1/2)					
Oeuvres(30 au maximum)					
Style	Titre	Auteur	Partition	Durée	Niveau de la chorale
Baroque	Seul	Mozart	http://www.exemple.fr	02:00	✓
Classique	Toto	Titi	http://www.exemple.fr	03:00	✓
Rock	Seul	Tata	http://www.exemple.fr	01:00	✓
Baroque	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	01:00	✓
Classique	Seul	Mozart	http://www.exemple.fr	02:00	✓
Oeuvres précédentes		Oeuvres suivantes			

En dessous de cette liste, s'affiche la durée totale des œuvres par style :

Programme de l'année (2/2)	
Durée totale par style	
Style	Durée totale
Baroque	1h
Classique	2h05
Rock	30 min

S'il n'existe pas d'œuvres associées au programme de l'année dans la base de données, voici la page qui s'affiche :

Programme de l'année
Oeuvres (30 résultats au maximum)
Il n'y a pas d'oeuvres.

4. Descriptions textuelles de la procédure interactive

Cette section reprend une description textuelle des traitements pour chaque cas d'utilisation de la section « Interactive » du diagramme de cas d'utilisation précédent.

4.1 Créer un compte

Acteurs principaux : Internaute

Acteurs secondaires : néant

Objectif : Permettre à l'internaute de créer un compte pour pouvoir accéder à des informations personnalisées

Préconditions : néant

Postconditions : néant

Scénario nominal :

1. L'internaute clique sur le bouton « Création de compte »
2. L'internaute renseigne un formulaire avec toutes les informations demandées (login, nom, prénom, ville, téléphone et mot de passe[2x]) . Il sélectionne le type de voix via une liste déroulante.
3. L'internaute valide le formulaire

Alternatives :

- Si le login existe, le texte « Login déjà existant » est affiché
- Si le login est nulle, le texte « Login non renseigné » est affiché
- Si les mots de passe ne sont pas identique, le texte « Mots de passe non identiques » est affiché
- Si le prénom comporte des caractères spéciaux, le texte « Le prénom ne doit pas comporter de caractères spéciaux » est affiché
- Si le nom comporte des caractères spéciaux, le texte « Le prénom ne doit pas comporter de caractères spéciaux » est affiché
- Si le numéro de téléphone comporte des caractères spéciaux, le texte « Le numéro de téléphone ne doit pas comporter de caractères spéciaux » est affiché

Maquette :



La maquette présente une interface d'inscription intitulée "Inscription". Elle est structurée comme un formulaire avec des champs de saisie pour les informations personnelles et de connexion. Les champs sont : "Nom :", "Prénom :", "Login :", "Mot de passe :", "Confirmation de mot de passe :", "Téléphone :", "Ville :", et "Voix :". Le champ "Voix :" est une liste déroulante avec "Alto" sélectionné. Un bouton "Valider" est situé en bas à gauche du formulaire.

Inscription	
Nom :	<input type="text"/>
Prénom :	<input type="text"/>
Login :	<input type="text"/>
Mot de passe :	<input type="password"/>
Confirmation de mot de passe :	<input type="password"/>
Téléphone :	<input type="text"/>
Ville :	<input type="text"/>
Voix :	<input type="text" value="Alto"/>
<input type="button" value="Valider"/>	

Requêtes SQL :

```
-- Ajout du login / mot de passe dans la table utilisateur  
INSERT INTO utilisateur (login, motdepasse)  
VALUES (:login, :password)
```

```
-- Ajout dans la table inscription  
INSERT INTO Inscription (typeInscription, montant, annee)  
VALUES (:statut, :montant, :annee) ;
```

```
-- Ajout dans la table choristes  
INSERT INTO Choriste (nom, prenom, idVoix, ville, telephone,  
login, idInscription)  
VALUES (:nom, :prenom, :idVoix, :ville, :telephone, :login,  
:idInscription) ;
```

Jeux de tests :

Un utilisateur qui s'inscrit sur le site peut rencontrer une notification s'il valide le formulaire sans modifier certains champs déjà pré-remplis :

The screenshot shows a web application interface for 'MyChorus'. The top navigation bar includes 'MyChorus', 'Choristes', 'Evénements', and 'Programme de l'année', along with a 'Connexion' button. The main content area contains a registration form with the following fields: 'Ville' (text input with 'Evry'), 'Voix' (dropdown menu with 'alto'), and 'Statut' (dropdown menu with 'Etudiant'). A tooltip message 'Veillez compléter ce champ.' is displayed above the 'Voix' field. Below the form is a blue 'M'inscrire' button. To the right of the form, there is a sidebar with a 'Connexion' button and a block of placeholder text (Lorem ipsum).

Ainsi tous les champs doivent être renseignés afin de s'inscrire.

Un message d'erreurs apparaît si un utilisateur s'inscrit avec un login qui existe déjà dans la base de données :

[MyChorus](#)[Choristes](#)[Evènements](#)[Programme de l'année](#)

S'inscrire en tant que Choriste

Erreur

L'identifiant mpaume existe déjà dans la base de données.

Identifiant

Mot de passe

Lorsque la confirmation du mot de passe n'est pas identique au mot de passe, voici le message qui apparaît :

[MyChorus](#)[Choristes](#)[Evènements](#)[Programme de l'année](#)

S'inscrire en tant que Choriste

Erreur

Les mots de passe entrés ne sont pas identiques.

Identifiant

Mot de passe

4.2 Modifier ses informations

Acteurs principaux :Utilisateur

Acteurs secondaires : néant

Objectif : Permettre à l'utilisateur de modifier ses informations tels que sa voix, sa ville, son numéro de téléphone et son mot de passe.

Préconditions : néant

Postconditions : néant

Scénario nominal :

1. L'utilisateur clique sur l'onglet « Modifier ses informations »
2. L'utilisateur modifie les informations souhaitées et confirme en cliquant sur le bouton « Valider »

Alternatives :

- Si le numéro de téléphone comporte des caractères spéciaux, le texte « Le numéro de téléphone ne doit pas comporter de caractères spéciaux» est affiché
- Si l'ancien mot de passe n'est pas correcte, le texte « Ancien mot de passe incorrect» est affiché
- Si les nouveaux mots de passe ne sont pas identique, le texte « Nouveaux mots de passe non identiques» est affiché

Maquette :

Maquette de la page "Mon compte" pour modifier les informations. Le formulaire est intitulé "Mon compte" et contient les champs suivants :

Nom :	Dupont
Prénom :	Henry
Login :	d.henry
Ancien mot de passe :	<input type="password"/>
Nouveau mot de passe :	<input type="password"/>
Téléphone :	<input type="text" value="06 45 74 24 12"/>
Ville :	<input type="text" value="Evry"/>
Voix :	<input type="text" value="Alto"/>
<input type="button" value="Valider"/>	

4.3 Ajouter un événement

Acteurs principaux : Chef de chœur

Acteurs secondaires : néant

Objectif : Permettre au chef de chœur d'ajouter un événement (répétition ou concert ou saison) avec un programme

Postconditions : néant

Scénario nominal :

1. Le chef de chœur clique sur l'onglet ajouter un événement.
2. Il renseigne le type, le nom, la date, le lieu et le programme de l'événement qu'il veut créer.
3. Il valide.

Alternatives :

- Si aucune n'œuvre n'est sélectionnée dans le champs programme, le texte «Pas d'œuvres sélectionnées» est affiché

Maquettes :

Création d'une saison

Nom :

Date :

Lieu :

Programme :

Ajouter une oeuvre

Titre :

Auteur :

Partition :

Durée :

Style :

Création d'un concert ou d'une répétition

Type d'évènement :

Nom :

Date :

Lieu :

Programme :

4.4 Ajouter un programme

Acteurs principaux : Chef de chœur

Acteurs secondaires : néant

Objectif : Permettre au chef de chœur de définir l'ensemble des œuvres qui vont être étudiées dans l'année.

Postconditions : néant

Scénario nominal :

4. Le chef de chœur clique sur l'onglet ajouter un programme.
5. Il renseigne le type, le nom, la date, le lieu et le programme de l'événement qu'il veut créer.
6. Il valide.

Alternatives :

- Si aucune œuvre n'est sélectionnée dans le champs programme, le texte «Pas d'œuvres sélectionnées» est affiché

Maquette :

Création d'un programme

Titre:

Compositeur :

Année courante :

Style :

Type d'évènement :

Jeux de tests :

Lorsque aucune œuvre n'est associée à un programme, l'onglet « Programme de l'année » affichait la page d'erreurs suivante :

500 Internal Server Error

Undefined offset: 0 (8)

```
#0 /var/www/ENSIIE-IBDW/views/ProgrammeLayout.php(10): flight\Engine->handleError(8, 'Undefined offse...', '/var/www/ENSIIE...', 10, Array)
#1 /var/www/ENSIIE-IBDW/lib/flight/template/View.php(118): include('/var/www/ENSIIE...')
#2 /var/www/ENSIIE-IBDW/lib/flight/Engine.php(435): flight\template\View->render('ProgrammeLayout...', Array)
#3 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(199): flight\Engine->_render('ProgrammeLayout...', Array)
#4 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(142): flight\core\Dispatcher::invokeMethod(Array, Array)
#5 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(48): flight\core\Dispatcher::execute(Array, Array)
#6 /var/www/ENSIIE-IBDW/lib/flight/Engine.php(63): flight\core\Dispatcher->run('render', Array)
#7 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(199): flight\Engine->_call('render', Array)
#8 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(199): flight\Engine->render('ProgrammeLayout...', Array)
#9 /var/www/ENSIIE-IBDW/lib/flight/Flight.php(43): flight\core\Dispatcher::invokeMethod(Array, Array)
#10 /var/www/ENSIIE-IBDW/controllers/Programme.php(141): Flight::__callStatic('render', Array)
#11 /var/www/ENSIIE-IBDW/controllers/Programme.php(141): Flight::render('ProgrammeLayout...', Array)
#12 /var/www/ENSIIE-IBDW/index.php(103): Programme::get()
#13 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(162): (closure)(Object(flight\net\Route))
#14 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(143): flight\core\Dispatcher::callFunction(Object(Closure), Array)
#15 /var/www/ENSIIE-IBDW/lib/flight/Engine.php(310): flight\core\Dispatcher::execute(Object(Closure), Array)
#16 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(191): flight\Engine->_start()
#17 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(142): flight\core\Dispatcher::invokeMethod(Array, Array)
#18 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(48): flight\core\Dispatcher::execute(Array, Array)
#19 /var/www/ENSIIE-IBDW/lib/flight/Engine.php(63): flight\core\Dispatcher->run('start', Array)
#20 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(191): flight\Engine->_call('start', Array)
#21 /var/www/ENSIIE-IBDW/lib/flight/core/Dispatcher.php(191): flight\Engine->start()
#22 /var/www/ENSIIE-IBDW/lib/flight/Flight.php(43): flight\core\Dispatcher::invokeMethod(Array, Array)
#23 /var/www/ENSIIE-IBDW/index.php(179): Flight::__callStatic('start', Array)
#24 /var/www/ENSIIE-IBDW/index.php(179): Flight::start()
#25 {main}
```


Ce problème a été corrigé, désormais une liste vide est affichée.

4.5 S'inscrire ou modifier à sa présence à un événement

Acteurs principaux : Choristes

Acteurs secondaires : néant

Objectif : Permettre aux choristes d'indiquer ou de modifier leurs présences à une répétition.

Postconditions : néant

Scénario nominal :

1. Le chef de chœur sélectionne dans le champ présence : présent, absent ou indécis.
2. Il valide.

Maquette :

La maquette est identique de celle qui est dans la procédure d'affichage, seul une liste déroulante avec un bouton « Valider » permettra de modifier ou d'indiquer la présence des choristes lors d'une répétition.

Jeux de tests :

Un choriste peut uniquement indiquer ou modifier sa présence à un événement à venir.

Evènements à venir

Nom	Date et heure	Lieu	Type	Statut	Présence
Musiques diverses	28 / 08 / 2014 - 19:00:00	3 rue Victor Hugo 78000 Sartrouville	Concert	Pas assez de choristes	présent ↕
Répétition générale: musiques diverses	28 / 07 / 2014 - 19:00:00	3 rue Victor Hugo 78000 Sartrouville	Répétition	Pas assez de choristes	absent ↕
Metallica au Zénith de Strasbourg	25 / 06 / 2014 - 11:00:00	avenue de la prairie Strasbourg 67000	Concert	Pas assez de choristes	absent ↕
Fête de la musique 2014	20 / 06 / 2014 - 20:00:00	15 avenue de l'Orge 91000 Evry	Concert	Pas assez de choristes	absent ↕
Passenger en concert unique à Paris !	25 / 05 / 2014 - 11:00:00	07, rue de la Musique - 77 000 Savigny-le-Temple	Concert	Pas assez de choristes	absent ↕
Répétition générale concert 1	24 / 05 / 2014 - 10:00:00	12 rue de la pétanque 33170 Gradignan	Répétition	Pas assez de choristes	présent ↕

Mettre à jour la présence

Si l'événement est passé, il n'y a plus de bouton « Mettre à jour sa présence » ni de champs « présence » qui s'affiche. Voici ce qui apparaît :

Evènements passés

Nom	Date et heure	Lieu	Type	Statut
Eminem au Palio	25 / 04 / 2014 - 11:00:00	route du Champ-de-Manœuvre - 91000 Evry	Concert	Pas assez de choristes
Jour de l'an chinois	10 / 04 / 2014 - 19:00:00	8 allée d'Ivry 75013 Paris	Concert	Pas assez de choristes

4.6 Valider l'inscription : étape 1

Acteurs principaux : Administrateur

Acteurs secondaires : néant

Objectif : Permettre à l'administrateur de vérifier si l'inscription est bien celle d'un choriste.

Postconditions : néant

Scénario nominal :

1. Afficher la liste des nouveaux inscrits
2. Valider l'inscription

4.7 Valider l'inscription : étape 2

Acteurs principaux : Trésorier

Acteurs secondaires : néant

Objectif : Permettre au trésorier d'indiquer que l'inscription est bien réglée par le choriste.

Postconditions : néant

Scénario nominal :

1. Afficher la liste des nouveaux inscrits
2. Valider l'inscription

Maquette :

Validation des nouveaux inscrits

Nom	Prénom	Login	Validation de l'administrateur	Validation du trésorier
Dupont	Henry	d.henry	✓	✓
Morand	Bertrand	m_bertrand	✓	✓
Dupont	Martin	dmartin		

Jeux de tests :

Une fois le compte créé, il doit être validé par l'administrateur puis par le trésorier.

Dans la base de données, cela se traduit par un champs validation dans la table choriste qui peut prendre les valeurs suivantes :

- 0 si le compte n'a pas été validé
- 1 si le compte a été validé par l'administrateur
- 2 si le compte a été doublement validé par l'administrateur puis le trésorier

Ainsi pour accéder au site un utilisateur doit au moins avoir été validé par l'administrateur. Par exemple, quand l'utilisateur Leger Kevin essaiera de se connecter sur le site, voici le message qui s'affichera :

Se connecter

Erreur d'authentification

Les informations transmises n'ont pas permis de vous authentifier (identifiant ou mot de passe incorrect).

☐ Se souvenir de moi

Connexion

Pour ce même utilisateur, voici ce qui s'affiche dans la page des validations de comptes lorsque l'administrateur se connecte :

Inscriptions à valider

Nom	Prenom	Type de Voix	Type d'inscription	Validation
Leger	kevin	basse	Etudiant	Valider

Lorsque le trésorier se connecte, il n'a pas la possibilité de valider ce compte puisque cette inscription n'a pas été validée par le webmaster dans un premier temps. Le bouton « Valider » n'apparaît pas dans ce cas, voici ce qui est alors affiché :

Inscriptions en attente du Webmaster

Nom	Prenom	Type de Voix	Type d'inscription
Leger	kevin	basse	Etudiant

5. Répartition des tâches

Tâches	Participants
Conception du diagramme d'utilisation	Tout le monde
Conception du diagramme de classes	Thomas
Procédures d'affichage : <ul style="list-style-type: none">Statistique des présencesAffichage des concerts passésAffichage des choristesAffichage du programme	Quentin, Sébastien Quentin, Sébastien, Thomas, Gwenaël Quentin, Sébastien Quentin, Sébastien
Procédures interactive <ul style="list-style-type: none">Enregistrement des présences aux répétitionsCréation des évènementsAjout/Modification d'un choristeCréation du programme	Quentin, Sébastien, Dounde, Gwenaël Quentin, Sébastien Quentin, Gaëtan, Thomas Quentin, Sébastien, Dounde
Script de création de la BDD	Thomas, Gwenaël
Script de remplissage de la BDD	Mylène, Gwenaël
Réalisation des maquettes d'interface	Gaëtan, Mylène, Dounde
Intégration des technologies (Flight, Bootstrap)	Quentin
Réalisation des rapports d'avancement / de la documentation	Mylène
Jeux de test	Mylène

6. Equipe



Quentin Barrand



Gaëtan Briquet



Dounde Diallo



Gwenaël Monnier



Mylène Paume



Sébastien Sabatier



Thomas Sarboni