

10 avril 2014

RAPPORT DE PROJET

CREATION DU SITE INTERNET POUR LA CHORALE DE LA VILLE D'EVRY

Rédacteur(s)	Responsabilité(s)
Christophe DEFAUD	Réalisation des maquettes Réalisation des diagrammes de cas d'utilisation CU pour la procédure d'affichage Elaboration des requêtes
Thomas RODRIGUES	Réalisation des maquettes Elaboration des requêtes
Pierrick KERIRZIN	Réalisation des maquettes Elaboration des requêtes
Patrice MELT	Elaboration du script de création la base de données Elaboration du script d'insertion de la base de données Elaboration des requêtes
Bastien BRESSAN	Elaboration du script de création la base de données Elaboration du script d'insertion de la base de données Elaboration des requêtes
David GONCALVES RAMOS	Elaboration des requêtes

HISTORIQUE

[illegible]

SOMMAIRE

1.	Introduction.....	5
2.	Description du projet	5
2.1	Exigences fonctionnelles.....	5
3.	Analyse et conception du projet	7
3.1	Création de la base de données	7
3.1.1	Script de création de la base de données	7
3.1.2	Les modifications apportées.....	7
3.1.3	Script de remplissage de la base de données.....	7
3.1.4	Nouvelle structure de la base de données	8
3.2	Analyse de la procédure d'affichage	9
3.2.1	Diagramme de cas d'utilisation	9
3.2.2	Les cas d'utilisation	10
3.2.3	La maquette.....	17
3.2.4	Les requêtes	17
3.3	Analyse de la procédure interactive	18
3.3.1	Diagramme de cas d'utilisation.....	18
3.3.2	Les cas d'utilisation	19
3.3.3	La maquette.....	24
3.3.4	Les requêtes	24
4.	Conception fonctionnelle de l'application	26
4.1	Structure du site.....	26
4.2	Structure des pages du site.....	26
4.3	Les variables de session.....	26
4.4	Variables PHP :.....	26
5.	Programmation et réalisation technique.....	27
5.1	Les langages utilisés.....	27
5.2	Structure de développement	27
5.3	Programmation de la procédure d'affichage.....	27
5.4	Test et validation de l'application	27
5.4.1	Les tests unitaires	27
5.4.2	Test : Affichage de la page d'accueil	27
5.4.3	Test : Authentification des choristes.....	29
5.4.4	Test de réussite	29
5.4.5	Test d'échec.....	29
5.4.6	Test : Nombre suffisant ou insuffisant de choriste pour un concert.....	30
5.4.7	Test : Déconnexion du site chorusbook.....	30
5.4.8	Test : Base de données vide.....	30
5.4.9	Test : Pagination liste choriste et liste concert.....	30

5.4.10	Les tests d'intégrations.....	30
5.5	Programmation de la procédure interactive.....	30
6.	Bilan du projet.....	31
6.1	Organisation de l'équipe	31
7.	Conclusion.....	31

1. Introduction

Un ensemble musical souhaite informatiser une partie de la gestion de son association. Cet ensemble est composé d'un chef de chœur et de choristes. Parmi ces choristes, certains sont membres du bureau de l'association (président, trésorier et secrétaire) ou ont des responsabilités autres (réservation des salles de répétition, conception des affiches, site web).

Chaque année, le chef de chœur définit un programme, contenant des œuvres sur un thème choisi. La gestion automatisée de l'ensemble doit permettre d'assurer les fonctions suivantes :

Partie site web :

- Affichage du programme de l'année en cours c'est-à-dire des œuvres étudiées dans l'année
- Affichage de la liste des choristes
- Affichage des concerts passés et à venir

Partie intranet :

- Gestion des choristes
- Gestion des événements
- Gestion des inscriptions

2. Description du projet

L'objectif du projet est de concevoir une application internet qui répond aux besoins exprimés par cet ensemble musical. Cette application devra être évolutive, paramétrable et d'une prise en main relativement simple.

D'abord une analyse de la base de données sera effectuée. Il en ressortira une liste d'évolutions et de recommandations qui permettra d'améliorer la structure de la base de données proposée.

2.1 Exigences fonctionnelles

Tableau des exigences fonctionnelles du projet		
Num	Domaine	Exigence
1	Gestion des choristes	Chaque choriste doit pouvoir s'inscrire en indiquant au minimum ses nom, prénom et un nom d'utilisateur. Puis il doit pouvoir modifier l'ensemble de ses coordonnées une fois connecté au site
2	Gestion des choristes	Un Choriste est représenté par son nom de famille, son prénom, sa ville de résidence, son ou ses numéros de téléphone, ainsi que sa voix ('s'/'a'/'t'/'b' pour respectivement les sopranos, altos, ténors et basses).
3	Gestion des événements	Pour chaque répétition ou concert, le chef de chœur souhaite pouvoir indiquer les œuvres qui vont être étudiées ou chantées. Chaque choriste doit pouvoir indiquer s'il sera présent à cet événement ou pas
4	Gestion des événements	Un Événement est un concert, une répétition ou une saison (type_evt). Un événement peut posséder une heure et une date, un lieu et doit posséder un nom;
5	Gestion des inscriptions	Le trésorier de l'association doit pouvoir entrer en début d'année les tarifs des différents types d'inscription. Puis, il doit indiquer qu'un choriste s'est bien inscrit au moment où celui-ci lui paye le montant de son inscription
6		Une Responsabilité est un rôle pris par l'un des utilisateurs, comme par exemple "Chef de chœur", "Trésorier" ou "Webmaster". Une même responsabilité ne peut être assurée que par un utilisateur au maximum, mais un même utilisateur peut prendre plusieurs responsabilités;

Tableau des exigences fonctionnelles du projet		
Num	Domaine	Exigence
7	Gestion des inscriptions	Une Inscription possède un type comme "Étudiant" et un montant pour une année donnée.
8	Gestion des oeuvres	Une Oeuvre correspond à un chant étudié, et possède un titre, un auteur, une durée (en minutes) et un style (par exemple "Renaissance" ou "Chanson à boire"). Le lien vers la partition correspondante est également stocké.
9	Gestion des statistiques	Un utilisateur authentifié doit pouvoir afficher la liste des différents choristes, avec pour chaque choriste : ses nom et prénom, le nombre de répétitions auxquelles il a assisté dans l'année, et le nombre de répétitions qu'il a manquées, ainsi que le taux de présence (nombre de répétitions où présent / nombre de répétitions où absent). Les choristes sont classés par voix. Chaque voix (avec son libellé) est clairement indiquée au début de la liste. À la fin de chaque voix, la présence moyenne des choristes de cette voix est indiquée.

3. Analyse et conception du projet

3.1 Création de la base de données

3.1.1 Script de création de la base de données

Il s'agit de vérifier la fiabilité et l'exactitude des données injectées dans la base de données de l'application

3.1.2 Les modifications apportées

Toutes ces modifications apportées à la structure de la base de données, ont pour but de produire une application évolutive et entièrement paramétrable. Nous évitons donc de figer des informations qui dans l'avenir seront susceptibles d'évoluer. L'ajout d'un nouveau type de voix ou d'un nouveau type d'évènement pourra donc se faire aisément.

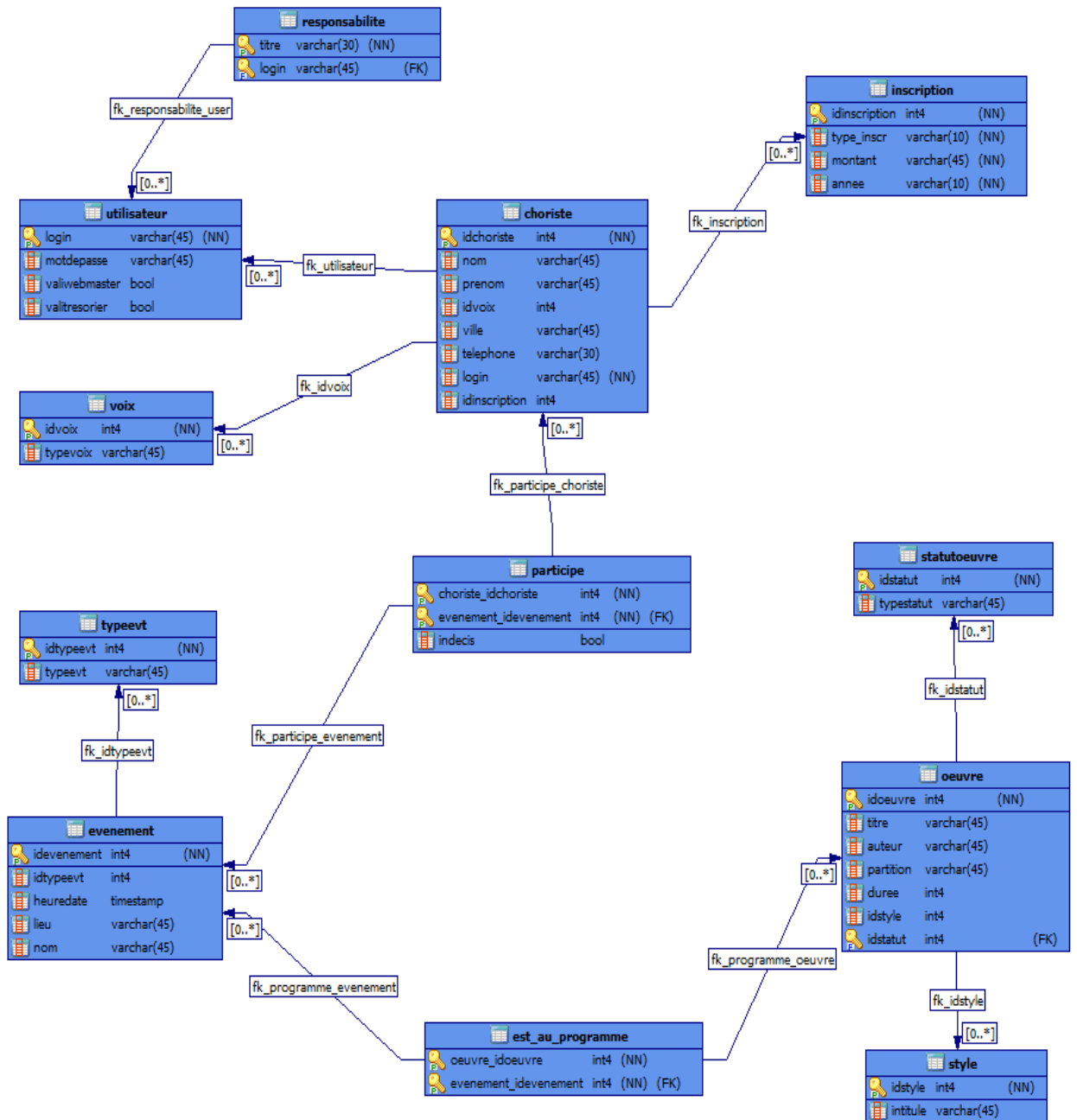
Tableau de synthèse des évolutions de la base de données		
Nom de la Table	Modifications effectuées	Commentaires
utilisateur	Ajout de 2 attributs: valideWebmaster et validateTresorerie afin de gérer le processus de validation d'inscription (RAPPEL : le trésorier et le webmaster doivent vérifier si un utilisateur a payé la cotisation)	
choriste	<ul style="list-style-type: none"> Modification de l'attribut voix en clé étrangère qui fait référence à la table voix Ajout de la table voix 	
evenement	<ul style="list-style-type: none"> Modification de l'attribut type_evt en clé étrangère qui fait référence à la table typeevt Ajout table typeevt 	
Œuvre	<ul style="list-style-type: none"> Ajout d'un attribut idStatut en clé étrangère qui fait référence à la table statutooeuvre Ajout de la table statutooeuvre Ajout de la table style 	
inscription	<ul style="list-style-type: none"> Nouvelle clé primaire idInscription 	
participe	<ul style="list-style-type: none"> Ajout d'un attribut indécis pour gérer les cas où le choriste n'est pas sûr de participer 	

3.1.3 Script de remplissage de la base de données

Afin de tester l'ensemble des fonctionnalités de site intranet « **chorusbook** », un script de remplissage contenant un jeu de données complet a été élaboré. Cela nous permettra de lancer une série de test qui nous aidera à valider le bon fonctionnement de l'application.

3.1.4 Nouvelle structure de la base de données

Les évolutions apportées à la structure de la base de données donnent le diagramme de classe suivant :

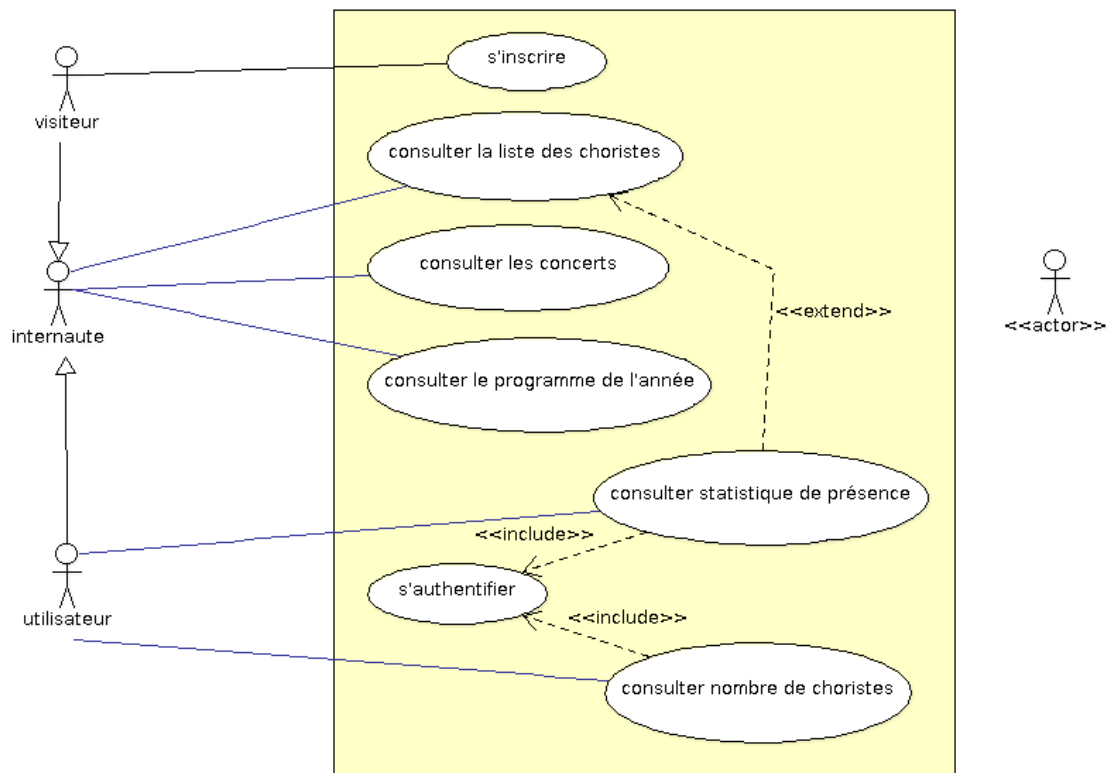


3.2 Analyse de la procédure d'affichage

L'objectif de la procédure d'affichage permet à l'utilisateur d'accéder aux fonctionnalités du site de manière intuitive afin qu'il puisse naviguer aisément.

3.2.1 Diagramme de cas d'utilisation

Le diagramme de cas d'utilisation des données s'effectue en trois phases :



3.2.2 Les cas d'utilisation

3.2.2.1 CU 1 : S'inscrire

Acteur principal :

Visiteur

Objectif :

L'utilisateur doit pouvoir s'inscrire sur l'application

Préconditions :

Néant

Postconditions :

Néant

Scénario nominal :

1. Le visiteur choisit l'action de s'inscrire sur l'application
2. L'application affiche un formulaire d'inscription
3. L'utilisateur renseigne les champs suivants :
 - nom
 - prénom
 - date de naissance
 - mail
 - ville
 - téléphone
 - voix
 - mot de passe
4. L'utilisateur valide le formulaire
5. L'application lui confirme que son inscription a été envoyée à l'administrateur
6. L'application prévient le visiteur par mail que son inscription a été validée

3.2.2.2 Consulter la liste des choristes

Acteur principal :

Internaute

Objectif :

L'internaute doit pouvoir consulter la liste des choristes.

Préconditions :

Néant

Postconditions :

Néant

Scénario nominal :

1. L'internaute choisit l'action de consulter la liste des choristes
2. L'application affiche la liste des choristes avec :
 - Le nom
 - Le prénom
 - La ville
 - La voix
 - Les concerts

2a) Pas de choriste dans la base => L'application indique à l'utilisateur qu'il n'y a aucun choriste

2b) Si le nombre de résultat est trop important => L'application affiche 20 résultats maximum et le reste est accessible à l'aide de la pagination

3.2.2.3 CU 3 : Consulter les concerts

Acteur principal :

Internaute

Objectif :

L'internaute doit pouvoir consulter la liste des concerts

Préconditions :

Néant

Postconditions :

Néant

Scénario nominal :

1. L'internaute choisit l'action de consulter la liste des concerts
2. L'application affiche la liste des concerts passés et à venir si pas de concert prévu ?
- 3a) L'utilisateur choisit de consulter les concerts passés
- 4a) L'application affiche les concerts passés avec :
 - Le nom
 - La date
 - La liste des choristes
- 3b) L'utilisateur choisit de consulter les concerts à venir
- 4b) L'application affiche les concerts à venir avec :
 - Le nom
 - La date
 - La liste des choristes
 - Places disponibles

Ergonomie :

- Si le nombre de résultat est trop important, un système de pagination est mis en place

3.2.2.4 CU 4 : Consulter le programme de l'année

Acteur principal :

Internaute

Objectif :

L'internaute doit pouvoir afficher le programme de l'année

Préconditions :

Néant

Postconditions :

Néant

Scénario nominal :

1. L'internaute choisit l'action de consulter le programme de l'année
2. l'application affiche les différentes œuvres du programme
3. L'internaute peut consulter l'ensemble des œuvres avec :
 - Le titre
 - L'auteur
 - La partition
 - La durée
 - Le style

3a) Si l'internaute est authentifié l'utilisateur peut savoir si une œuvre est en cours d'apprentissage.

Ergonomie :

- Il est possible de filtrer l'ensemble des programmes selon leurs styles et dates
- Si un nombre trop important de résultat est retourné, une pagination est mise en place pour alléger

3.2.2.5 Consulter statistique de présence

Acteur principal :

Utilisateur

Objectif :

Un utilisateur authentifié doit pouvoir consulter la liste des choristes : ses nom, prénom, le nombre de répétitions auxquelles il a assisté, le nombre de répétitions manquées, et son taux de présence aux différents événements.

Préconditions :

L'utilisateur a été authentifié

Postconditions :

Néant

Scénario nominal :

1. L'internaute authentifié (utilisateur) choisit de consulter la liste des choristes
- 2a) L'application affiche la liste des choristes classés par voix :
 - Nom
 - Prénom
 - Nombre de répétitions auxquelles il a assisté
 - Nombre de répétitions manquées
 - Taux de présence pour chaque choriste
 - Taux de présence moyenne des choristes de chaque voix
- 2b) Pas de choriste dans la base => L'application indique à l'utilisateur qu'il n'y a aucun choriste

Ergonomie :

- Si le nombre de résultats est trop important => L'application affiche 20 résultats maximum et le reste est accessible à l'aide de la pagination

3.2.2.6 Consulter le nombre de choristes

Acteur principal :

Utilisateur

Objectif :

Un utilisateur doit pouvoir consulter pour les concerts à venir s'il y a suffisamment de choristes pour le concert

Préconditions :

L'utilisateur a été authentifié

Postconditions :

Néant

Scénario nominal :

1. L'utilisateur choisit de consulter la liste des concerts à venir
- 2a) L'application affiche la liste des concerts à venir :
 - date et heure du concert
 - lieu et carte éventuellement
 - un voyant indiquant si le nombre de choristes par voix est suffisant pour le concert
- 2b) L'application indique qu'il n'y a pas de concerts à venir

Ergonomie :

- Un voyant vert indique qu'il y a suffisamment de choristes pour une voix donnée pour le concert.
- Un voyant rouge indique qu'il y a moins de 2 choristes pour une voix donnée pour le concert.

3.2.2.7 S'authentifier

Acteur principal :

Internaute

Objectif :

L'internaute doit pouvoir s'authentifier

Préconditions :

Néant

Postconditions :

Néant

Scénario nominal :

1. L'utilisateur choisit l'action de s'authentifier sur la plateforme
2. L'application propose à l'utilisateur d'entrer son identifiant ainsi que son mot de passe
3. L'utilisateur rentre ses informations
4. L'application confirme que l'authentification a pu se faire
- 4a) L'application indique à l'utilisateur une erreur lors de l'authentification
 - retour à l'étape 3

3.2.3 La maquette

La maquette du site est disponible en HTML dans le fichier tar.gz. Les liens hypertexte sont actifs au niveau des boutons. On peut donc naviguer et accéder aux différentes pages de la maquette.

3.2.4 Les requêtes

Synthèse des requêtes de la procédure d'affichage		
Num	Domaine	Requête
1	Statistique des présences (2.1.1)	<pre>(SELECT a.idChoriste, a.nom, a.prenom, (SELECT count(idEvenement) FROM choriste LEFT OUTER JOIN participe on idChoriste = choriste_idchoriste LEFT OUTER JOIN evenement on evenement_idevenement=idvenement NATURAL JOIN typeevt WHERE typeevt = 'Répétition' AND idEvenement is not null AND a.idchoriste=idchoriste AND indecis='False') as nb_presence, (SELECT count(idvenement) FROM typeevt NATURAL JOIN evenement INNER JOIN participe on idevenement=evenement_idevenement WHERE typeevt = 'Répétition') as nb_repete FROM choriste a);</pre>
Nota : Pour l'instant, le nombre d'absence sera calculé en php avec le nombre de répétitions moins le nombre de présences, idem pour le taux de présence.		
2	Affichage des concets passes et avenir (2.2.1)	<pre>(SELECT * FROM Evenement NATURAL JOIN TypeEvt WHERE typeevt like 'Concert' AND heureDate > now() ORDER BY heureDate) UNION ALL (SELECT * FROM Evenement NATURAL JOIN TypeEvt WHERE typeevt like 'Concert' AND heureDate < now() ORDER BY heureDate limit 5) ;</pre>
3	Affichage des choristes (2.3.1)	<pre>SELECT choriste.nom, choriste.prenom, voix.typevoix, array_agg(responsabilite.titre) FROM voix NATURAL JOIN choriste NATURAL JOIN utilisateur LEFT OUTER JOIN responsabilite on utilisateur.login=responsabilite.login GROUP BY choriste.nom, choriste.prenom, voix.typevoix ORDER BY typevoix,nom,prenom;</pre>
Nota : Array_agg() permet d'avoir une seule ligne pour chaque utilisateur même si celui-ci à plusieurs responsabilités		
4	Affichage du programme (2.4.1)	<pre>SELECT * FROM TypeEvt NATURAL JOIN evenement INNER JOIN est_au_programme on idEvenement=Evenement_idEvenement INNER JOIN Oeuvre on Oeuvre_idOeuvre=idOeuvre NATURAL JOIN statutOeuvre WHERE typeevt like 'Saison' AND idtypeevt=3;</pre>
Nota :		

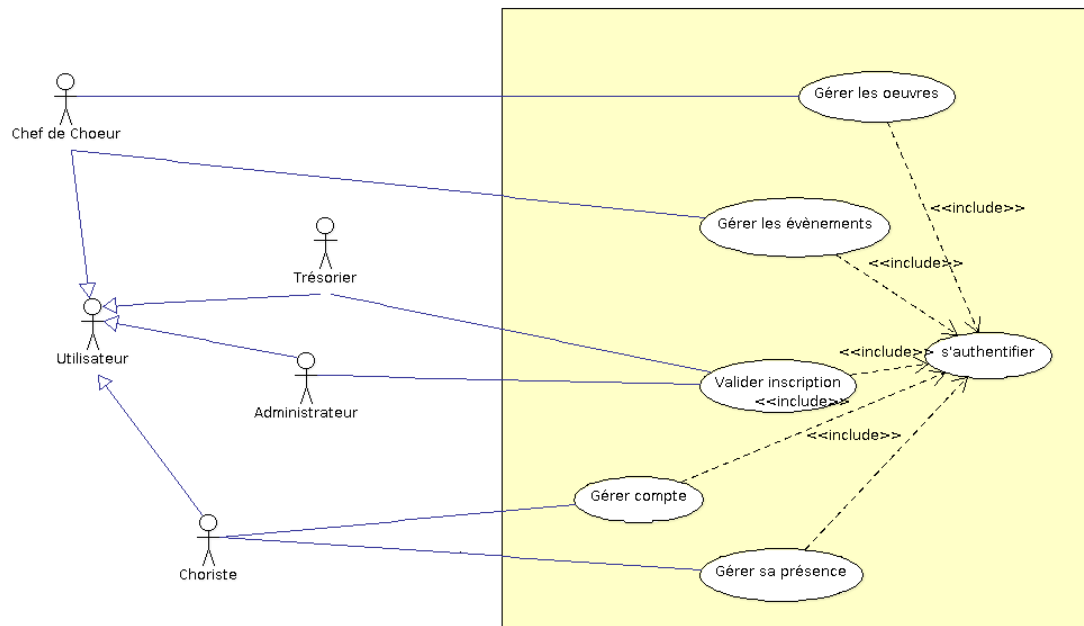
Ajout de lignes dans la base afin que la requête donne des résultats.

```
INSERT INTO est_au_programme (oeuvre_idOeuvre, evenement_idEvenement)
values (1,34);
INSERT INTO est_au_programme (oeuvre_idOeuvre, evenement_idEvenement)
values (2,34);
INSERT INTO est_au_programme (oeuvre_idOeuvre, evenement_idEvenement)
values (3,34);
```

Nota: Notre exemple de Saison est l'id 3, mais ce paramètre sera donné par le clique de l'internaute

3.3 Analyse de la procédure interactive

3.3.1 Diagramme de cas d'utilisation



3.3.2 Les cas d'utilisation

3.3.2.1 Validation Inscription par l'administrateur

Acteur principal :

Administrateur.

Objectif:

L'administrateur doit pouvoir vérifier s'il s'agit bien d'un choriste

Précondition:

L'utilisateur a envoyé le formulaire d'inscription

Postcondition:

En attente de validation du trésorier

Scénario nominal:

1. Le système affiche la liste des choristes dont l'inscription n'est pas encore validée
2. L'administrateur vérifie qu'il s'agit bien d'un choriste
3. L'administrateur valide l'inscription

Erreur:

3a L'administrateur ne valide pas l'inscription

3.3.2.2 CU 2: Validation Inscription par le trésorier

Acteur principal :

Trésorier.

Objectif:

Le trésorier doit pouvoir vérifier si le choriste a bien réglé son inscription

Précondition:

L'administrateur a validé l'inscription

Postcondition:

Le choriste est officiellement inscrit

Scénario nominal:

1. Le système affiche la liste des choristes dont l'inscription est validée par l'administrateur
2. Pour chaque choriste le trésorier vérifie si l'inscription a été réglée
3. Le trésorier valide l'inscription

Erreur:

3a Le trésorier ne valide pas l'inscription

3b L'inscription est en attente le temps que le choriste paye l'inscription

3.3.2.3 Gérer sa présence à un événement

Acteur principal :

Choriste

Objectif :

Le choriste doit pouvoir indiquer sa présence : présent, absent, indécis à un événement

Préconditions :

Le choriste doit être loggué

Postconditions :

Les choix de présence aux différents événements sont sauvegardés

Scénario nominal :

1. Le choriste se rend sur la page des événements

2a) L'application affiche la liste des événements à venir, ainsi qu'un champ pour indiquer sa présence (présent, peut-être, absent) à cet événement.

2b) L'application affiche qu'aucun événement a été prévu pour le moment.

3a) Le choriste n'a jamais effectué de choix pour un événement donné. Par défaut, il est noté « absent ». Il peut faire un autre choix ou le laisser inchangé.

3b) Le choriste a déjà effectué un choix auparavant pour un concert donné. C'est ce choix qui est sélectionné. Il peut modifier son choix ou le conserver.

Ergonomie :

- Les événements affichés sur la page sont triés par type (concert, répétition, etc...).
- Si le nombre de résultats est trop important, un système de pagination est mis en place.

3.3.2.4 Créer événement

Acteur principal :

Chef de chœur

Objectif :

Créer un événement (concert ou répétition)

Préconditions :

Doit être connecté en tant que chef de chœur

Pour créer un événement de type concert ou répétition, il faut au préalable créer une saison

Postconditions :

Néant

Scénario nominal :

1. Le chef de chœur choisit l'action de créer un nouvel événement
2. L'application affiche un formulaire de création d'événement
3. Le chef de chœur renseigne les champs suivants :
 - nom
 - type(concert ou répétition ou saison)
 - date et heure
 - œuvres du programme
 - participation
4. Le chef de chœur valide le formulaire
5. L'application lui confirme que son événement a bien été créé

Scénario alternatif :

- 3b. Le chef d'orchestre choisit l'action de créer une nouvelle œuvre
- 4b. CU 2 : créer oeuvre

Ergonomie

- Le statut d'une œuvre sera indiqué au chef d'orchestre à l'aide d'un code de couleur (vert étudié, jaune pas étudié..)
- Si le chef d'orchestre ajoute une œuvre non étudiée ou en cours d'apprentissage dans un concert l'application l'avertira par un message.

3.3.2.5 Créer oeuvre

Acteur principal :

Chef de choeur

Objectif :

Créer une oeuvre

Préconditions :

Doit être connecté en tant que chef de choeur

Postconditions :

Néant

Scénario nominal :

1. Le chef de chœur choisit l'action de créer une œuvre
2. L'application affiche un formulaire de création d'œuvre
3. Le chef de chœur renseigne les champs suivants :
 - Titre
 - Auteur
 - Partition
 - Style
 - Durée
 - statut
4. Le chef de choeur valide le formulaire
5. L'application lui confirme que son œuvre a bien été créé

Scénario d'erreur :

- 4b Si le titre d'une œuvre existe déjà
- 5b.L'application affiche un message d'erreur reprise du cas d'utilisation au 3

3.3.3 La maquette

La maquette du site est disponible dans le tar.gz sous forme d'images.

3.3.4 Les requêtes

Synthèse des requêtes de la procédure d'affichage		
Num	Domaine	Requête
1	Vérifier si l'utilisateur a déjà indiqué qu'il serait peut-être/présent à un évènement	SELECT COUNT(*) FROM participe WHERE Choriste_idChoriste=idChoriste AND Evenement_IdEvenement=idEvent;
2	Si \$resultat = 1 et que l'utilisateur a indiqué qu'il sera peut-être/présent	UPDATE participe SET Indecis=\$bouton WHERE Choriste_idChoriste=\$idChoriste AND Evenement_IdEvenement=\$idEvent;
3	Si \$resultat = 1 et que l'utilisateur a indiqué qu'il sera absent	DELETE FROM participe WHERE Choriste_idChoriste=\$idChoriste AND Evenement_IdEvenement=\$idEvent;
4	Si \$resultat = 0 et que l'utilisateur a indiqué qu'il sera peut-être/présent	INSERT INTO participe VALUES (\$idChoriste, \$idEvent, \$bouton);
5	Affiche les différents types d'évènement	SELECT typeevt FROM typeevt;
6	Création d'une oeuvre	INSERT INTO oeuvre(idoeuvre, titre, auteur, partition, duree, style, idStatut) VALUES (3, 'Cantate', 'BACH Johann Sebastian', 'BWV 150', '60', 'Religieux',3);
7	Affiche les œuvres	SELECT titre FROM oeuvre ;
8	Affiche œuvre d'une saison	SELECT titre FROM est_au_programme p, typeevt t, œuvre o, evenement e WHERE t.idTypeevt = e.type_evt AND e.idEvenement = p.idEvenement AND p.idOeuvre = O.idOeuvre AND t.typeevt = 'saison'
9	Création d'une saison	INSERT INTO evenement(idevenement, idTypeEvt, heuredat, lieu, nom) VALUES (34, 3, NULL, "", 'Saison 2013-2014');
10	Ajout œuvre à une saison/concert/répétition	INSERT INTO est_au_programme(oeuvre_idOeuvre, Evenement_IdEvenement) VALUES (3, 34);
11	Création d'un concert	INSERT INTO evenement(idevenement, idTypeEvt, heuredat, lieu, nom) VALUES (12, 1, '2014-01-29 20:00:00', 'Paris 10ème Eglise St Vincent de Paul', 'Concert 4 Requiem Mozart');
12	Création d'une répétition	INSERT INTO evenement(idevenement, idTypeEvt, heuredat, lieu, nom) VALUES (13, 2, '2014-08-29 19:00:00', 'Evry - Au conservatoire, 2 rue Bullet', 'Répétition générale');
13	Validation du formulaire d'inscription par l'utilisateur (2.3.2)	BEGIN ; INSERT INTO utilisateur(login, motdepasse, ValiWebmaster, ValiTresorier) VALUES ('pmelt', md5('pmelt'),false, false); INSERT INTO choriste(nom ,prenom ,idVoix ,ville ,telephone ,login ,idInscription) VALUES ('melt', 'patrice', '1', 'angers', '0601020304', 'pmelt', 1); COMMIT ;
Nota : ValiWebmaster et ValiTresorier sont à false, il ne peut donc pas se connecter tant que son inscription n'a pas été validé (gérer via le php)		

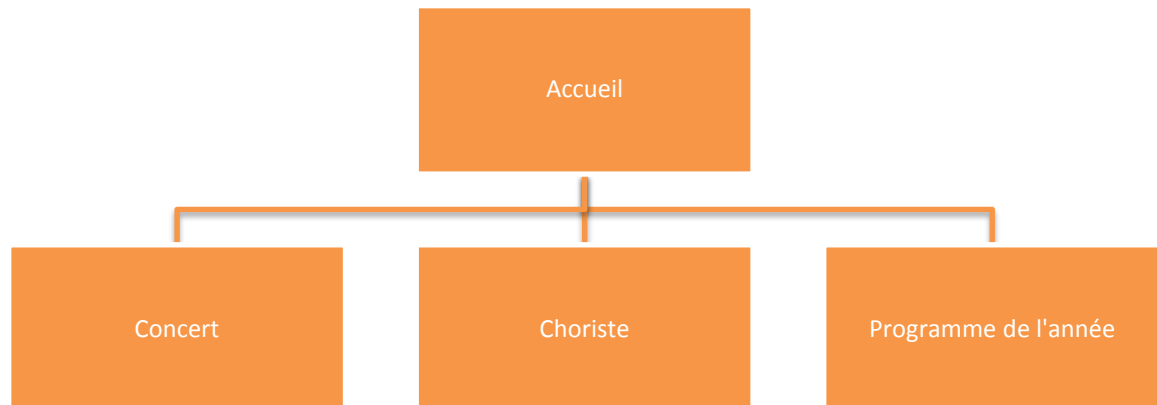
Synthèse des requêtes de la procédure d'affichage		
Num	Domaine	Requête
14	Affichage des choristes pas encore validé pour le webmaster	SELECT * FROM utilisateur NATURAL JOIN choriste WHERE ValiWebmaster='False';
15	Validation par le Webmaster	UPDATE utilisateur SET ValiWebmaster='True' WHERE login='pmelt';
16	Non validation	BEGIN; DELETE FROM choriste WHERE login='pmelt'; DELETE FROM utilisateur WHERE login='pmelt'; COMMIT;
17	Affichage des choristes pas encore validé pour le webmaster	SELECT * FROM utilisateur NATURAL JOIN choriste WHERE ValiTresorier='False';
18	Validation par le Trésorier	UPDATE utilisateur SET ValiTresorier='True' WHERE login='pmelt';
19	Modification des informations de l'utilisateur	BEGIN; UPDATE utilisateur SET motdepasse=md5('pmelt') WHERE login='pmelt' ; UPDATE choriste SET nom='melt', prenom='patrice', idVoix=1, ville='angers', telephone='0601020304' WHERE login='pmelt' ; COMMIT ;
Nota : Tous les champs sont mis à jour car sur la page de modification des informations les champs seront pré-remplis		

4. Conception fonctionnelle de l'application

Elaboration en cours...

4.1 Structure du site

Elaboration en cours...(CD)



4.2 Structure des pages du site

Toutes les pages du site auront cette présentation.



4.3 Les variables de session

Elaboration en cours...

4.4 Variables PHP :

\$bouton :

- true si l'utilisateur choisit "Peut-être"
- false si l'utilisateur choisit "Présent"
- reset si l'utilisateur choisit "Absent"

\$idChoriste :

Correspond à l'id du choriste connecté dans la table Choriste

\$idEvent :

Correspond à l'id de d'événement pour lequel le choriste effectue un changement

\$resultat :

Contient le résultat d'une requête

5. Programmation et réalisation technique

5.1 Les langages utilisés

Elaboration en cours...

5.2 Structure de développement

L'organisation retenue pour développer dans de parfaites conditions, proches d'un Framework, s'appuie sur le concept MVC et se présente comme suit :

Rapide présentation : Thomas ou Christophe ? Qui se sent capable de le faire ?

5.3 Programmation de la procédure d'affichage

L'objectif est de développer dans un langage de programmation adapté, les différents cas d'utilisation élaborés au chapitre « §6 - Analyse de la procédure d'affichage ».

Pour des raisons de simplicité et de gain de temps, toute l'équipe a décidé de renoncer à l'utilisation d'un Framework. Cependant l'organisation mise en place est proche du modèle MVC et sera présentée dans le paragraphe suivant.

5.4 Test et validation de l'application

Le but de cette étape est de lancer une série de tests en rapport avec la procédure d'affichage afin de s'assurer que le site internet « **chorusbook** » fonctionne en toutes circonstances.

5.4.1 Les tests unitaires

Ces tests permettent de valider le bon fonctionnement de chaque fonctionnalité développée indépendamment des autres. On s'assure durant cette phase de test, que les fonctions implémentées, correspondent aux exigences spécifiées en début de projet

5.4.2 Test : Affichage de la page d'accueil

A l'aide d'un navigateur (site validé sous firefox), taper l'url suivante : <http://localhost/chorusbook>. On obtient donc la fenêtre ci-dessous.

[localhost/Chorusbook2/index.php](#)

Cliquer pour avancer, maintenir pour voir l'historique

Naruto Episode Prizee IUT Site du Zero Caisse d'épargne Site officiel Distri... Android ...

Concerts

Login Mot de passe Connexion

Concerts

Choristes

Programme de l'année

S'inscrire

Nom	Date et Heure	Lieu
CONCERTS À VENIR		
Concert 2 Requiem allemand	27/05/2014 20:00	Paris - Au Palais des Congrès
Messe solennelle	03/10/2014 20:00	Paris 10ème Eglise St Vincent de Paul
Concert 4	04/10/2014 20:30	Paris 12ème Eglise La trinité"
Concert 5 profit assos ELA	05/10/2014 20:00	Paris 4ème Eglise St Louis"
Bach La passion selon St Jean	28/11/2014 20:00	Paris 19ème Le Zénith
Requiem de Mozart	29/11/2014 20:30	Evry - Au conservatoire, 2 rue Bullet
Concert anniversaire	30/11/2014 20:00	Evry - Salle Croix blanche
CONCERTS PASSÉS		
Messe de requiem	25/01/2014 11:00	Eglise de Creteil
Concert 3 a capella	28/01/2014 20:30	Paris 12ème aux "Voix sur berges"
Concert 4 Requiem Mozart	29/01/2014 20:00	Paris 10ème Eglise St Vincent de Paul"


5.4.3 Test : Authentification des choristes


Ce test permet de vérifier la connexion au site « chorusbook ». Si l'authentification est validée, le choriste sera redirigé sur une page contenant la liste des concerts. Si échec de l'authentification, un message d'erreur apparaîtra « **login et / ou mot de passe incorrects** » sur le bandeau supérieur du site chorusbook.


5.4.4 Test de réussite

Le nom du choriste authentifié apparaît sur le bandeau supérieur du site chorusbook.

Concerts
pmelt
Déconnexion


 Concerts


 Choristes


 Programme de l'année


Nom	Date et Heure	Lieu
CONCERTS À VENIR		
Concert 2 Requiem allemand	27/05/2014 20:00	Paris - Au Palais des Congrès
Messe solennelle	03/10/2014 20:00	Paris 10ème Eglise St Vincent de Paul
Concert 4	04/10/2014 20:30	Paris 12ème Eglise La trinité
Concert 5 profit assos ELA	05/10/2014 20:00	Paris 4ème Eglise St Louis
Bach La passion selon St Jean	28/11/2014 20:00	Paris 19ème Le Zénith
Requiem de Mozart	29/11/2014 20:30	Evry - Au conservatoire, 2 rue Bullet
Concert anniversaire	30/11/2014 20:00	Evry - Salle Croix blanche
CONCERTS PASSÉS		
Messe de requiem	25/01/2014 11:00	Eglise de Creteil
Concert 3 a capella	28/01/2014 20:30	Paris 12ème aux "Voix sur berges"
Concert 4 Requiem Mozart	29/01/2014 20:00	Paris 10ème Eglise St Vincent de Paul


5.4.5 Test d'échec


Un message d'erreur s'affiche en rouge « **login et / ou mot de passe incorrects** » au niveau du bandeau supérieur du site « chorusbook ».


Concerts

Login et/ou mot de passe incorrects.


 Concerts


 Choristes


 Programme de l'année


 S'inscrire

Nom	Date et Heure	Lieu
CONCERTS À VENIR		
Concert 2 Requiem allemand	27/05/2014 20:00	Paris - Au Palais des Congrès
Messe solennelle	03/10/2014 20:00	Paris 10ème Eglise St Vincent de Paul
Concert 4	04/10/2014 20:30	Paris 12ème Eglise La trinité
Concert 5 profit assos ELA	05/10/2014 20:00	Paris 4ème Eglise St Louis
Bach La passion selon St Jean	28/11/2014 20:00	Paris 19ème Le Zénith
Requiem de Mozart	29/11/2014 20:30	Evry - Au conservatoire, 2 rue Bullet
Concert anniversaire	30/11/2014 20:00	Evry - Salle Croix blanche
CONCERTS PASSÉS		
Messe de requiem	25/01/2014 11:00	Eglise de Creteil
Concert 3 a capella	28/01/2014 20:30	Paris 12ème aux "Voix sur berges"
Concert 4 Requiem Mozart	29/01/2014 20:00	Paris 10ème Eglise St Vincent de Paul

5.4.6 Test : Nombre suffisant ou insuffisant de choriste pour un concert

Lors de la consultation des concerts, un voyant, indique si le nombre de choriste est suffisant ou non pour chacun des concerts.



Concerts



Choristes



Programme de l'année

Nom	Date et Heure	Lieu
CONCERTS À VENIR		
 Concert 2 Requiem allemand	27/05/2014 20:00	Paris - Au Palais des Congrès
 Messe solennelle	03/10/2014 20:00	Paris 10ème Eglise St Vincent de Paul
 Concert 4	04/10/2014 20:30	Paris 12ème Eglise La trinité"
 Concert 5 profit assos ELA	05/10/2014 20:00	Paris 4ème Eglise St Louis"
 Bach La passion selon St Jean	28/11/2014 20:00	Paris 19ème Le Zénith
 Requiem de Mozart	29/11/2014 20:30	Evry - Au conservatoire, 2 rue Bullet
 Concert anniversaire	30/11/2014 20:00	Evry - Salle Croix blanche
CONCERTS PASSÉS		
Messe de requiem	25/01/2014 11:00	Eglise de Creteil
Concert 3 a capella	28/01/2014 20:30	Paris 12ème aux "Voix sur berges"
Concert 4 Requiem Mozart	29/01/2014 20:00	Paris 10ème Eglise St Vincent de Paul"

pmelt
Déconnexion

5.4.7 Test : Déconnexion du site chorusbook

5.4.8 Test : Base de données vide

5.4.9 Test : Pagination liste choriste et liste concert

Ce test de pagination sera proposé au prochain livrable.

5.4.10 Les tests d'intégrations

Ce test consiste à valider l'intégration des différentes fonctionnalités testées unitairement précédemment.

5.5 Programmation de la procédure interactive

Elaboration en cours...

6. Bilan du projet

Elaboration en cours...

6.1 Organisation de l'équipe

Nous avons renoncé à l'utilisation d'un Framework car certains collaborateurs n'avaient pas les compétences requises pour travailler sur ce type d'outil. D'autre part, le gestionnaire GITHUB n'a pas été utilisé pour la gestion de versions du projet.

Cependant nous avons pu gérer et échanger sur le projet à travers l'outil « Drive » de Google.

Rédacteur(s)	Responsabilité(s)
Christophe DEFAUD	Réalisation des maquettes Réalisation des diagrammes de cas d'utilisation CU pour la procédure d'affichage Elaboration des requêtes
Thomas RODRIGUES	Réalisation des maquettes Elaboration des requêtes
Pierrick KERIRZIN	Réalisation des maquettes Elaboration des requêtes
Patrice MELT	Elaboration du script de création la base de données Elaboration du script d'insertion de la base de données Elaboration des requêtes
Bastien BRESSAN	Elaboration du script de création la base de données Elaboration du script d'insertion de la base de données Elaboration des requêtes
David GONCALVES RAMOS	Elaboration des requêtes

7. Conclusion

Elaboration en cours...

