



RAPPORT DE PROJET



REALISE PAR
Julian ALARCON
Tanguy CADIEUX
Célia GROSCH
Samira YOUNESY

SOUS LA DIRECTION DE
M. Rémi COLETTA

POUR L'OBTENTION DU DUT INFORMATIQUE DE GESTION

ANNEE UNIVERSITAIRE 2018 - 2019

Remerciements

Nous tenons à remercier toutes les personnes qui ont contribué au bon déroulement de notre projet ainsi que celles qui nous ont aidés pour la rédaction de notre rapport.

Tout d'abord, nous adressons nos remerciements à notre tuteur de projet, M. Rémi COLETTA de l'IUT de Montpellier, qui nous a montré le chemin à suivre pour la réalisation de notre projet.

Nous tenons à remercier Mme Justine DELEBARRE, professeur à l'IUT de Montpellier, qui nous a éclairés sur la façon de rédiger le rapport de projet.

M. Sébastien GAGNE, professeur à l'IUT de Montpellier mérite aussi des remerciements, car il nous a enseigné les cours de programmation web qui ont permis la réalisation de notre projet.

Enfin, nous remercions Mme Hélène BASTIDE pour sa relecture.

Table des matières

Introduction	1
1. Cahier des charges.....	2
1.1. Présentation du sujet.....	2
1.2. Analyse du contexte.....	2
1.3. Analyse des besoins fonctionnels	4
1.4. Analyse des besoins non-fonctionnels.....	5
1.4.1. Spécifications techniques.....	5
1.4.2. Contraintes ergonomiques	6
2. Rapport technique	7
2.1. Conception.....	7
2.2. Réalisation	9
3. Résultats	13
3.1. Installation	13
3.2. Test/Validation	13
3.3. Manuel d'utilisation	14
3.3.1. Connexion et déconnexion.....	17
3.3.2. Inscription au site.....	17
3.3.3. Modifier ses informations	17
3.3.4. Créer un festival	17
3.3.5. Poser une candidature pour un festival	17
3.3.6. Spécifier vos préférences et disponibilités.....	17
3.3.7. Consulter la liste des organisateurs.....	17
3.3.8. Consulter votre planning de bénévole.....	17
3.3.9. Interface des postes	17
3.3.10. Renseigner les besoins en créneaux	17
3.3.11. Accepter les candidatures des bénévoles	17
3.3.12. Accéder à la liste des bénévoles.....	17
3.3.13. Accéder à la liste et aux informations des autres organisateurs.....	17
3.3.14. Remplir, modifier et publier le planning du festival	17
4. Gestion de projet	18
4.1. Démarche personnelle	18
4.2. Planification des tâches	19
4.3. Bilan critique par rapport au cahier des charges.....	20

Conclusion.....	22
Bibliographie.....	I
Annexes techniques.....	III
Annexe 1 : Maquettes du site	IV
Annexe 2 : Analyse de contexte du site Volunteo.....	VI
Annexe 3 : Diagramme d'activités	XII
Annexe 4 : Liste des fonctionnalités à tester	XIII
Annexe 5 : Réunion 1 : Présentation du projet « Application de Gestion de Bénévoles d'un Festival ».....	XIV
Annexe 6 : Réunion 2 : Présentation des maquettes.....	XVI
Annexe 7 : Réunion 3 : Présentation de la base de données et du modèle MCV	XVII
Annexe 8 : Réunion 4 : Présentation du CRUD*	XIX

Table des figures

Figure 1. Capture d'écran du site web Volunteo, depuis la démonstration gratuite du site et avec l'interface administrateur.....	3
Figure 2. Diagramme de cas d'utilisation des bénévoles, fait avec StarUML.....	4
Figure 3. Diagramme de cas d'utilisation des organisateurs, fait avec StarUML.....	5
Figure 4. Capture d'écran du répertoire de travail	7
Figure 5. Capture d'écran de la fonction générique save.....	8
Figure 6. Diagramme de classe initial, fait avec StarUML	9
Figure 7. Diagramme de classe intermédiaire, fait avec StarUML	10
Figure 8. Capture d'écran de la page d'accueil du site	14
Figure 9. Numérisation des maquettes du site (1).....	IV
Figure 10. Numérisation des maquettes du site (2).....	V
Figure 11. Capture d'écran de la page d'accueil de Volunteo, du point de vue administrateur depuis la démonstration du site.....	X
Figure 12. Capture d'écran de la page des équipes de Volunteo, du point de vue administrateur depuis la démonstration du site.....	XI
Figure 13. Capture d'écran de la page des bénévoles de Volunteo, du point de vue administrateur depuis la démonstration du site.....	XI
Figure 14. Diagramme d'activités, fait avec Modelio.....	XII

Glossaire

Les termes définis dans ce glossaire sont identifiables dans le corps du texte au moyen d'un astérisque (*).

CRUD : Create (Créer) – Read (Lire) – Update (Mettre à jour) – Delete (Supprimer), interface de gestion d'objets sur la base de données avec les fonctionnalités de création, modification, suppression et lecture sur chaque objet.

MVC : Model (Modèle) – View (Vue) – Controller (Contrôleur), forme d'architecture de site web en trois parties interconnectées

Modèle : le modèle regroupe les fonctions d'interaction avec la base de données

Vue : la partie vue se charge de l'affichage des pages pour l'utilisateur

Contrôleur : le contrôleur est l'interface de redirection de l'utilisateur vers les différentes vues selon les actions sélectionnées, exploitant les données récupérées à partir du modèle.

Seed: graine de génération correspondant à une chaîne de caractères concaténée à avant un mot de passe pour augmenter leur sécurité

Introduction

Un festival est une manifestation musicale et un grand regroupement de personnes. De plus, parmi ces personnes se trouvent souvent des bénévoles contribuant à la gestion de différentes activités. Cependant il n'est pas toujours aisé de gérer tout le personnel. L'organisation d'un festival repose sur la présence de bénévoles (de l'ordre de 50 bénévoles pour le *Printival* ou le *festival de Mourèze*). Pour chacun d'eux, il faut collecter des informations personnelles, des disponibilités et les postes qu'ils préfèrent pour pouvoir les affecter. D'un autre côté, les organisateurs du festival renseignent les besoins en personnel pour chaque poste sous forme de créneaux. Il faut ensuite établir un planning pour les bénévoles qui couvre les besoins tout en priorisant les préférences en postes de ces derniers. Quelques sites web proposent déjà ce type de service, mais ils sont souvent payants.

L'objectif de ce projet est alors de créer une application web accessible et gratuite qui puisse servir pour plusieurs festivals, afin d'en gérer les bénévoles.

Notre projet répond donc à cette problématique : comment gérer les plannings des bénévoles en fonction de l'organisation d'un (ou des) festival(s) ?

Tout d'abord, nous allons étudier les exigences du projet dans le cahier des charges, nous présenterons ainsi le sujet et nous analyserons son contexte et les besoins associés. Ensuite, nous exposerons la conception et la réalisation du projet grâce au rapport technique. Par ailleurs, nous présenterons les résultats de notre site web. Enfin nous parlerons de la gestion du projet et de ce qu'il nous a apporté.

1. Cahier des charges

Dans cette première partie, nous allons présenter notre projet, comment il s'adapte vis-à-vis de la concurrence ainsi que les fonctionnalités et lignes générales de son implémentation.

1.1. Présentation du sujet

L'organisation de bénévoles d'un festival peut facilement devenir compliquée, ainsi notre projet comprendra quatre paramètres importants : les bénévoles, les organisateurs, les festivals et les créneaux horaires concernant les postes de ces festivals. L'intérêt sera de faire concorder ces paramètres de manière cohérente.

Les bénévoles seront des personnes volontaires qui s'occuperont des tâches liées à l'organisation d'un festival. Ces personnes auront aussi des contraintes et des préférences vis-à-vis des tâches qui les attendront lors dudit festival.

Les organisateurs encadreront les bénévoles et les affecteront aux tâches en fonction des disponibilités des bénévoles et de leurs préférences. Ils devront avoir accès aux informations des bénévoles du festival sur lequel ils participeront.

Dans un premier temps, le sujet sera présenté de sorte à accueillir un seul festival, puis nous conviendrons qu'il pourra aussi gérer une multitude de festivals. Ainsi nous devons être en mesure d'afficher les informations des bénévoles et des organisateurs en fonction du festival sur lequel ils se porteront volontaires.

Enfin, les organisateurs devront pouvoir éditer un planning après avoir affecté les bénévoles selon plusieurs critères tels que leurs préférences ou leurs disponibilités, mais aussi en fonction des créneaux et du nombre de places restantes sur ces créneaux.

Notre site offrira alors un moyen simple d'accès et gratuit d'aide à l'organisation de bénévoles d'un festival.

1.2. Analyse du contexte

Lors de nos recherches, nous avons trouvé des sites internet et des logiciels qui se rapprochent de notre objectif. Cependant, certains de ces sites présentent des fonctions moins avancées que celles que nous devons développer, ou qui sont payantes, ce qui pourrait freiner des associations. Nous allons alors présenter un site internet qui se rapproche du but de notre projet, Volunteo.

Volunteo est un programme de gestion de bénévoles pour des événements, par exemple un festival, qui est payant (les 10 premiers bénévoles offerts puis 1 euro par bénévole). Cette application web nous permet la gestion facile des volontaires en déléguant les différentes tâches aux chefs d'équipes. Ainsi la mise en place des autres équipes est indépendante des autres. Il permet aussi des prises de contacts rapides et efficaces entre l'administrateur et les utilisateurs (bénévoles et organisateurs), et entre utilisateurs. D'après ce site, un chef d'équipe peut être un simple bénévole dans une autre équipe (c'est à dire qu'il ne sera pas chef d'équipe dans cette dernière). Par ailleurs, la page d'un bénévole contient toutes les informations qui le concerne, comme ses coordonnées, ses préférences pour certaines équipes, ses disponibilités et les postes auxquels il est affecté. Cette page est, par la suite, accessible par l'administrateur, ses chefs d'équipes et par le bénévole concerné. Ce programme de gestion de bénévoles contiendra

d'autres fonctionnalités importantes, plus amplement détaillées dans l'Annexe 2 : Analyse de contexte du site Volunteo.

Le site est simple et agréable à l'œil. D'une taille adaptée à l'écran, il s'affiche facilement sur tous les navigateurs et est accessible à tous les internautes (à condition de payer) et sur toutes les plateformes techniques. Le menu est facilement identifiable et possède un moteur de recherche pour trouver un poste, un bénévole ou une équipe. Les pages sont homogènes : le nombre de couleurs est limité et celles-ci aident à la navigation. Par exemple, le rouge est utilisé quand le taux de remplissage d'une équipe ou des créneaux n'est pas atteint à 100%, et inversement, le vert sera utilisé lorsque le remplissage est de l'ordre de 100%, ainsi que pour les fonctionnalités liées à l'administrateur. Les informations du site sont compréhensibles par les utilisateurs, remplissant pratiquement tous les critères ergonomiques (cf. Annexe 2 : Analyse de contexte du site Volunteo).

Nous remarquons que ce site a des aspects à la fois positifs et négatifs. En effet, ce site offre un panel de fonctionnalités assez large pour le(s) responsable(s) de l'événement. Par ailleurs, la prise en main du site est rapide et celui-ci permet une communication efficace avec l'assistance technique et entre les utilisateurs. De plus, Volunteo suit pas à pas la vie de l'événement et informe constamment le(s) responsable(s). Néanmoins, le programme de gestion de bénévoles est payant et ne met pas à la disposition des utilisateurs un calendrier de l'événement.

Nous pouvons en conclure que le programme de gestion de bénévoles Volunteo est assez complet. Néanmoins, la démonstration gratuite ne permet pas de voir le point de vue d'un bénévole ou d'un chef d'équipe. Plusieurs événements de toutes tailles choisissent Volunteo pour simplifier leur organisation. Pour ce qui est de notre projet, nous nous appuyons sur des fonctionnalités sur lesquels Volunteo est construit car beaucoup d'entre elles se ressemblent.

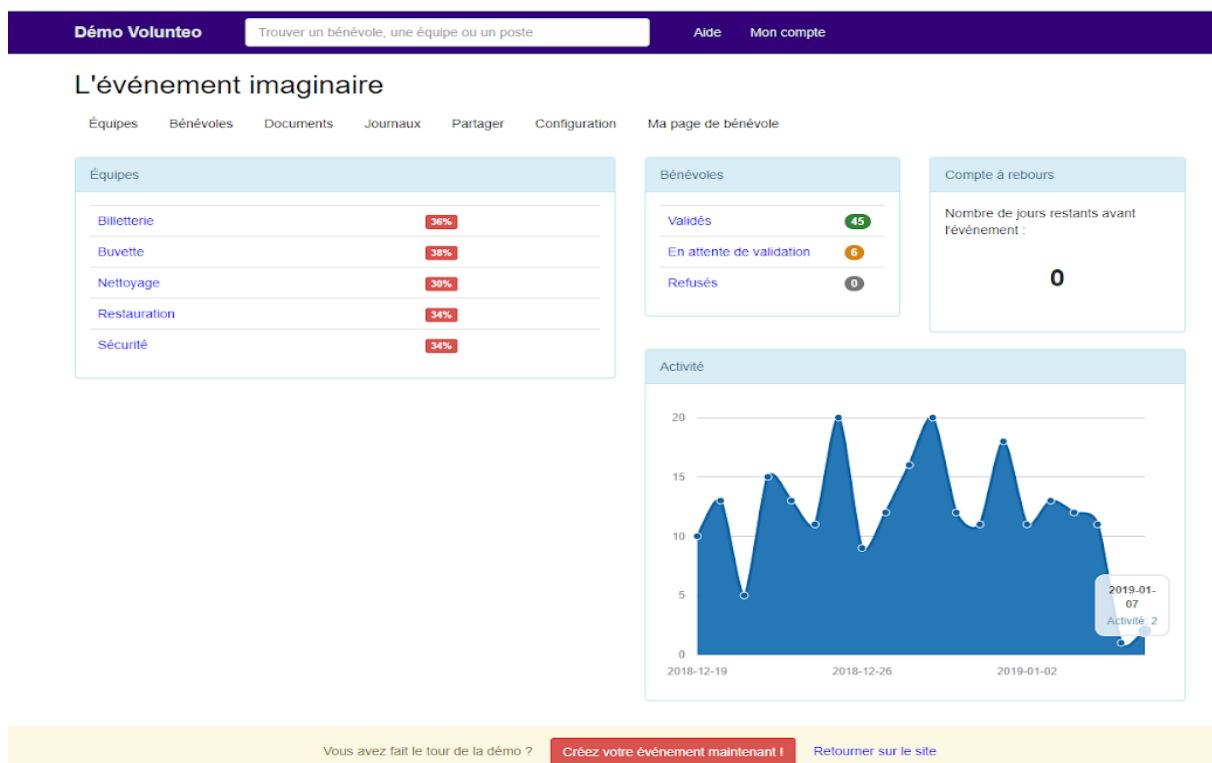


Figure 1. Capture d'écran du site web Volunteo, depuis la démonstration gratuite du site et avec l'interface administrateur

1.3. Analyse des besoins fonctionnels

Le site est accessible depuis un ordinateur, à la maison lors de la phase de préparation, ou sur un smartphone, lors d'une vérification de planning ou pour chercher le numéro de l'organisateur pour lui demander des renseignements. Notre site facilitera la gestion des bénévoles d'un festival. Nous allons alors définir les besoins des utilisateurs, qui seront regroupés en deux catégories pour chaque festival : les bénévoles et les organisateurs. Chaque groupe aura alors des besoins et fonctionnalités spécifiques.

D'une part, les bénévoles devront pouvoir renseigner leurs informations lors de leur inscription sur le site, ainsi que leurs préférences pour les postes à pourvoir et leurs disponibilités. Lorsqu'ils se connectent, ils doivent aussi pouvoir modifier leurs informations. Quand ils sont affectés à un festival, ils ont la possibilité de consulter leur planning personnel et le planning général, pour le jour mais aussi pour la période du festival. (Cf. Figure 2.)

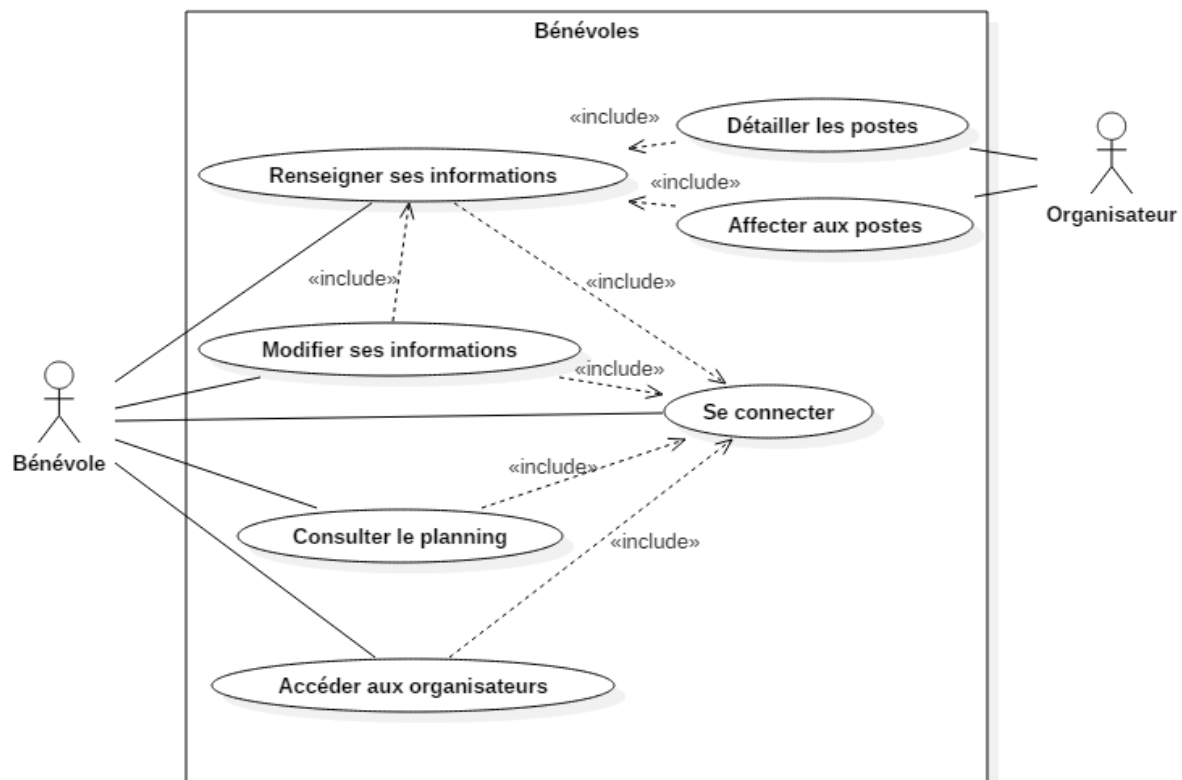


Figure 2. Diagramme de cas d'utilisation des bénévoles, fait avec StarUML

D'autre part, les organisateurs d'un festival pourront s'authentifier sur le site. Ils disposent des droits sur un festival donné, ils devront donc renseigner les besoins en postes, en créneaux et le nombre de personnes par poste et par créneaux. Pour pouvoir affecter les bénévoles, ils devront accepter leur candidature depuis le site. Après avoir affecté les bénévoles en fonction de leurs préférences et des besoins selon les postes et les créneaux, les organisateurs s'occuperont de publier le planning. Enfin, ils auront accès à la liste des bénévoles, avec leurs informations, ainsi que les plannings déjà publiés. (Cf. Figure 3.)

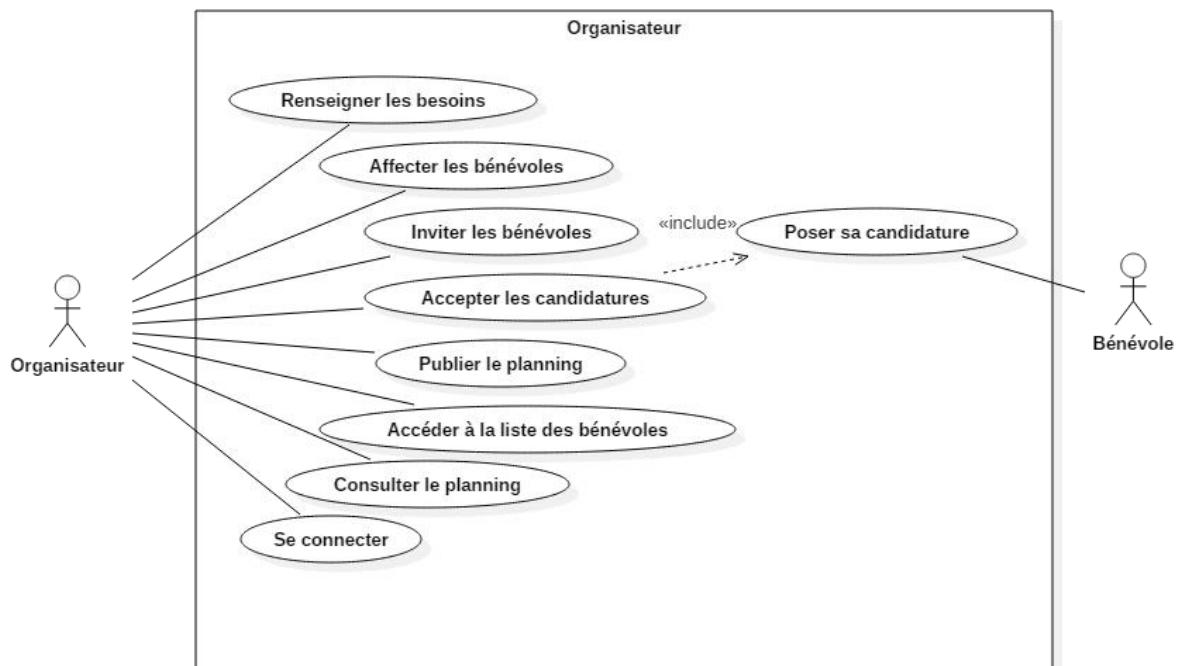


Figure 3. Diagramme de cas d'utilisation des organisateurs, fait avec StarUML

1.4. Analyse des besoins non-fonctionnels

Après avoir analysé le fond de notre projet, nous allons étudier la forme, c'est-à-dire les spécifications techniques et les contraintes ergonomiques.

1.4.1. Spécifications techniques

En vue de réaliser une application web performante, le site sera programmé en PHP selon l'architecture MVC*. Chaque élément du MVC* sera implémenté des méthodes CRUD*, nous aurons alors accès à tous les éléments et nous pourrons les manipuler. Nous devons prendre en compte plusieurs éléments, les bénévoles et les organisateurs, les festivals, les postes à pourvoir, les créneaux des différents postes et les disponibilités des bénévoles.

Il intégrera aussi une base de données comportant toutes les informations nécessaires à conserver. Le choix nous est donné quant à la base de données que nous utiliserons. Nous comprenons d'ailleurs que tous les éléments seront liés entre eux et qu'il y aura des tables qui serviront d'intermédiaires entre certains éléments.

De plus, nous penserons à générer graphiquement un planning sur le site, codé en PHP et HTML. Nous nous inspirerons alors de tutoriels sur internet pour implémenter cette fonction.

Nous prendrons aussi en compte les différences entre les organisateurs et les bénévoles afin de générer une liste de fonctionnalités différentes pour chaque festival, étant donné que chaque catégorie d'utilisateurs aura des droits différents.

1.4.2. Contraintes ergonomiques

Afin de créer un site internet simple d'utilisation et rapide à comprendre pour quelconque utilisateur, il sera important de respecter les normes de l'ergonomie, de sorte à ce que l'utilisateur soit satisfait de son expérience sur notre site et qu'il revienne pour organiser un futur festival.

Nous ferons également en sorte que le planning mis à disposition soit clair pour que l'organisation et le déroulement du festival se déroule correctement et sans encombre.

Nous entreprendrons ainsi la réalisation de maquettes (Cf. Annexe 1 : Maquettes du site) pour visualiser un premier plan d'ensemble du site. On y retrouvera le cheminement des fonctionnalités et les interfaces distinctes des bénévoles et organisateurs. Chaque panneau correspondra à une vue à implémenter et son action correspondante. Ces maquettes nous permettront de converger vers une vision commune du projet et de pouvoir communiquer cette vision plus clairement à notre tuteur. La disposition du site sera par la suite modifiée pour répondre à certaines nouvelles contraintes telles que la présence d'un compte unique permettant de postuler à tous les festivals.

Enfin pour améliorer l'expérience de l'utilisateur, nous pourrions ajouter des éléments graphiques pour le guider tels que des pourcentages pour le remplissage des postes.

2. Rapport technique

Nous avons précédemment défini le cahier des charges et les objectifs à accomplir pour la réussite de notre site. Nous allons à présent nous concentrer sur la façon dont nous avons pris en main le projet, d'abord en théorie, puis en pratique.

2.1. Conception

La phase la plus importante d'un projet consiste en sa conception, c'est-à-dire quel est le but pour l'utilisateur quant à la prise en main du logiciel, et dans notre cas, le site de gestion de bénévoles d'un festival. Nous allons alors lister les éléments importants de notre projet et présenter la manière dont nous voulons les faire interagir.

Notre site est implémenté selon la base de l'architecture MVC* (Cf. Figure 4) afin de se rapprocher de la programmation objet. Pour chaque objet dans la base de données, on utilise les quatre opérations habituelles surnommée CRUD*, avec des méthodes génériques et semi-génériques, afin que la réponse de l'application soit rapide et facile à corriger lors d'éventuels problèmes.

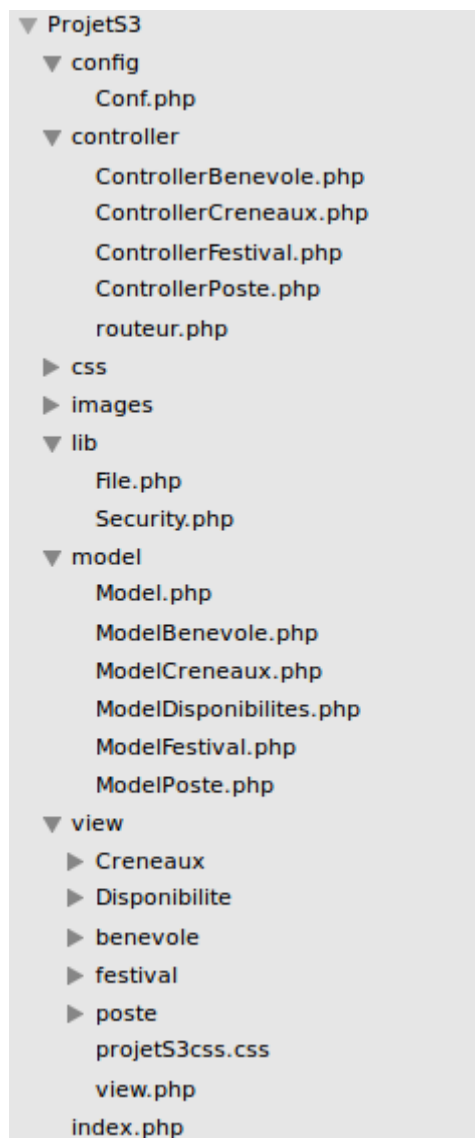


Figure 4. Capture d'écran du répertoire de travail

Afin d'optimiser la place et le temps d'exécution, nous avons décidé de faire des méthodes génériques (Cf. Figure 5 et Annexe 9 : Exemples de méthodes génériques) et semi-génériques. Pour cela nous employons un modèle générique pour une partie des fonctions. Les fonctions plus spécifiques nécessitant des jointures dans la base de données sont gérées directement depuis les modèles correspondants. Dans la figure 5, nous illustrons l'enregistrement de données dans une table quelconque.

```
public static function save($data){
    $table_name = static::$object;
    $class_name = 'Model';
    $class_name .= ucfirst($table_name);
    $attribut = "";
    $valeurs = "";
    $values = array();
    foreach ($data as $key => $value){
        $attribut .= $key . ", ";
        $valeurs .= "'$value' . ", ";
        $values_etapes = array($key => $value);
        $values = array_merge($values, $values_etapes);
    }
    $attribut = rtrim($attribut, " \t,");
    $valeurs = rtrim($valeurs, " \t,");

    $sql = "INSERT INTO $table_name($attribut) VALUES ($valeurs)";
    echo "<pre>";
    print_r($sql);
    echo "</pre>";
    $req_prep = Model::$pdo->prepare($sql);
    $req_prep->execute($values);
}
```

Figure 5. Capture d'écran de la fonction générique save

La structure est reliée à une base de données phpMyAdmin (Cf. Figure 6). Nous assistons alors à une codépendance et à des échanges importants entre les deux éléments.

Afin d'avoir une ligne de conduite vis-à-vis du design et des fonctionnalités, nous avons réalisé des maquettes en début de projet (Cf. Annexe 1 : Maquettes du site). Elles ont alors permis de regrouper tous les éléments importants de notre projet, et ainsi de nous concentrer sur ces derniers.

De plus, nous pouvons voir que l'implémentation du calendrier sur notre site sera un aspect assez important. En effet, les organisateurs auront une vue d'ensemble sur les créneaux (occupés ou pas) de leurs festivals mais aussi d'y créer des événements. De plus, les bénévoles pourront visualiser l'ensemble de leurs affectations sur les festivals auxquels ils participent. Ainsi le codage du calendrier en PHP nous permettra d'implémenter un calendrier dynamique lié à notre base de données avec des fonctionnalités telles que l'ajout ou la suppression d'un créneau ou d'un événement. Des fonctions seront mises à disposition pour accorder le calendrier du site au calendrier civil, par exemple avec les années bissextiles ou encore le nombre différent de jours de février. La partie HTML donnera vie à la forme du calendrier dans un tableau adapté. Dans cette partie, nous retrouverons du code PHP qui appellera des fonctions déjà existantes dans notre banque de procédures.

Dans le but de référencer toutes les méthodes dont nous aurons besoin et des différentes étapes de chaque action, nous avons créé un diagramme d'activité (cf. Annexe 3 : Diagramme d'activités). Il montrera alors les différentes étapes d'organisation de bénévole d'un festival, de l'émergence de l'idée du festival de l'organisateur jusqu'à la fin de l'événement. Ce diagramme

d'activités met en évidence deux événements majeurs : premièrement, la création d'un compte utilisateur et deuxièmement, la demande de participation à un festival existant, ceci nous permettra d'avoir un statut de bénévole pour celui-ci. Le système vérifie donc si le bénévole a la possibilité de postuler, c'est à dire si son compte a bien été validé et si le festival demandé ne se déroule pas lors d'un autre festival pour lequel ce bénévole aurait été accepté. Ensuite les organisateurs de ce festival ont la possibilité d'accepter ou non ce bénévole. Enfin, une fois qu'il a été accepté, les organisateurs peuvent l'affecter à un poste dans le planning et mettre en ligne le nouveau planning créé.

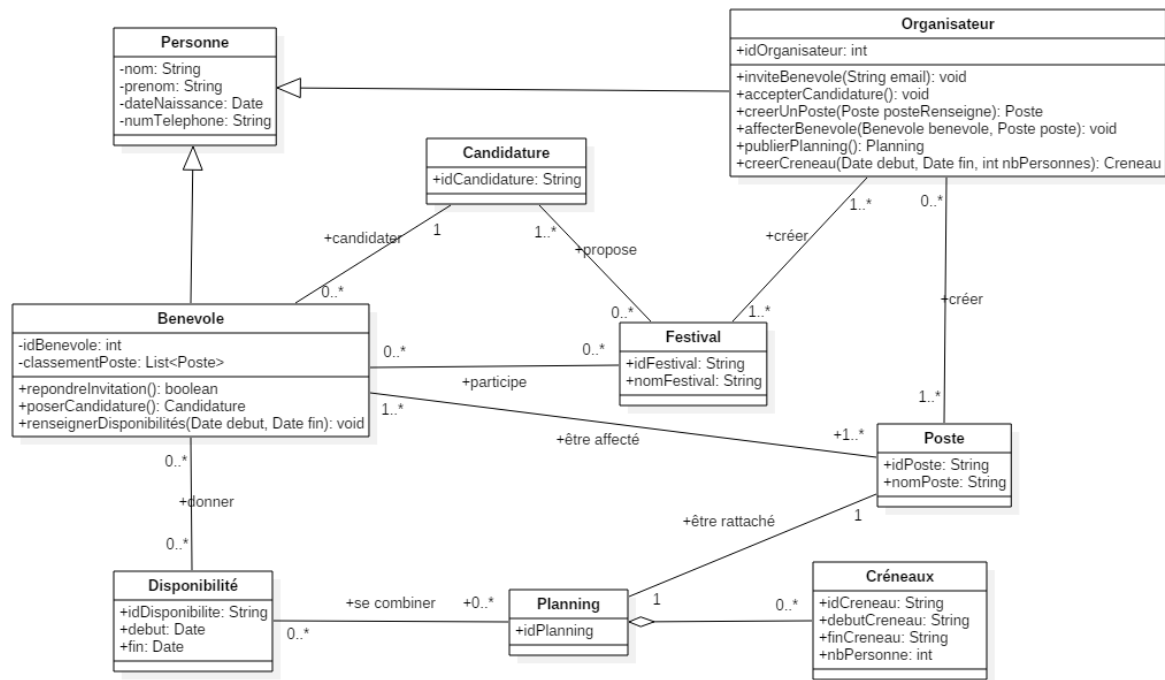


Figure 6. Diagramme de classe initial, fait avec StarUML

2.2. Réalisation

La phase de réalisation vient après la phase de conception et elle met en place tous les points abordés lors de cette dernière. Elle montre alors ses forces et ses faiblesses, que l'on va mettre en avant ou corriger. Nous montrerons donc dans cette sous-partie la réalisation de notre projet.

Nous avons développé ce projet de A à Z, nous avons donc rencontré certaines difficultés que nous avons réussi à surmonter.

Le site étant un projet dynamique et impliquant de nombreux échanges avec la base de données, nous en avons d'abord établi une complexe (Cf. Figure 6). Elle comprenait alors plus de neuf tables et on retrouve de la redondance, notamment vis-à-vis de la candidature, où il n'est pas nécessaire d'en faire car nous pourrions faire en sorte qu'un bénévole participe à un festival, mais il ne serait validé que par l'organisateur, ultérieurement.

A la suite la deuxième réunion avec M. COLETTA (Cf. Annexe 8 : Réunion 4 : Présentation du CRUD*), nous avons convenu de réduire la base de données (Cf. Figure 7), et ainsi de l'optimiser. Nous avons donc supprimé les tables planning et candidature et, à la place, nous avons fait un champ booléen 'valide' de façon que l'organisateur puisse valider lui-même

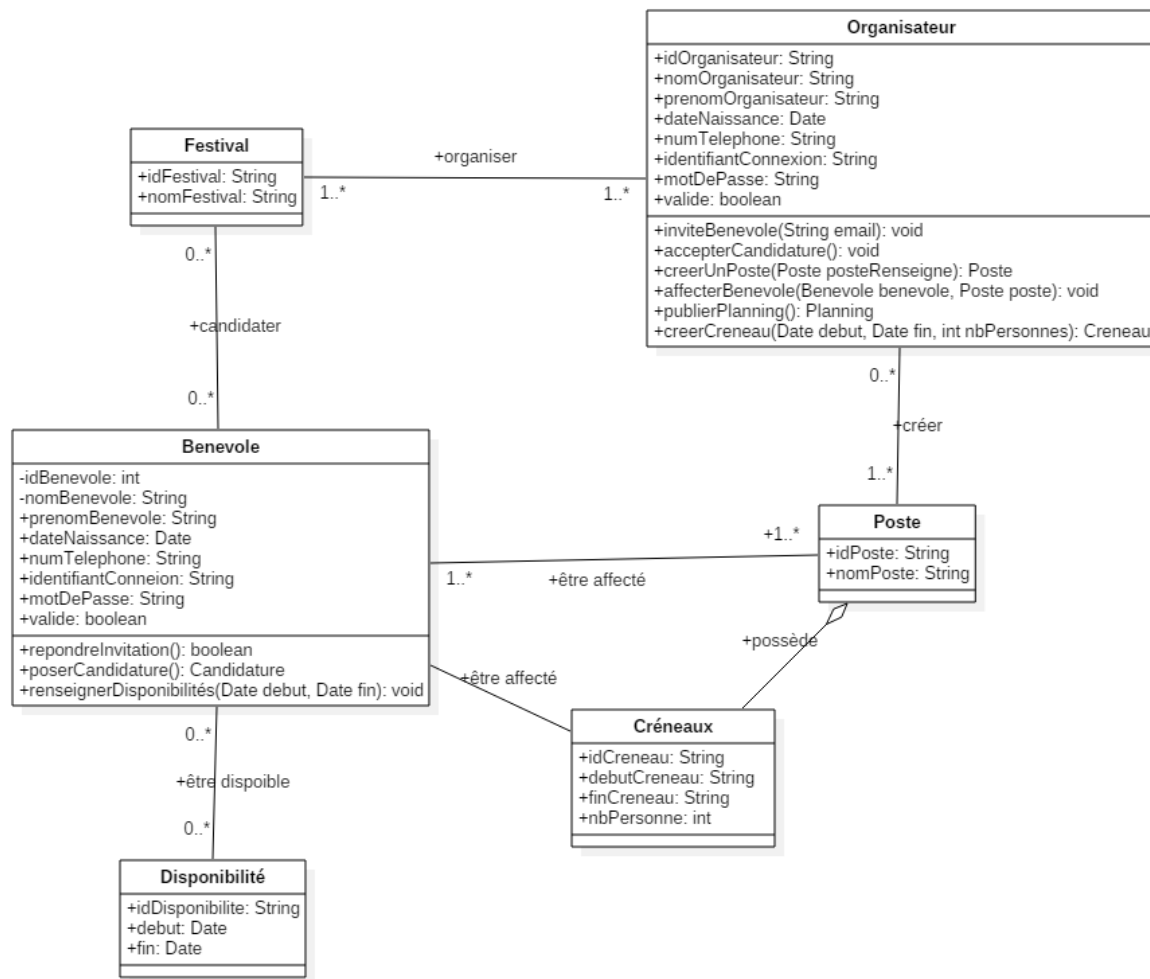


Figure 7. Diagramme de classe intermédiaire, fait avec StarUML

le bénévole pour son festival. Nous remarquons, par ailleurs, que les acteurs de notre site web pourraient être regroupés sous une seule table, que nous appellerons 'Bénévole' par la suite.

Pour la dernière et actuelle édition de notre base de données, nous avons donc uni les tables 'Bénévole' et 'Organisateur', et nous avons ajouté le champ 'estOrganisateur' et 'valide' dans la classe association 'Participe_Festival'. (Cf. Figure 8). Nous avons d'ailleurs ajouté des attributs supplémentaires pour mieux détailler les informations complémentaires, notamment au niveau de la table 'Festival', où il manquait le lieu, les dates de début et de fin ainsi qu'une brève description du festival.

Nous avons établi un portail de fonctionnalités pour chaque festival, celui-ci prend la forme d'une vue avec des liens vers les différentes fonctionnalités possibles selon les droits de l'utilisateur. (Cf.

Annexe 10 : Vue du portail d'un festival).

Après avoir validé l'agencement des tables de la base de données, une question se pose quant à la clé primaire de la table 'Bénévole' : doit-on prendre un identifiant personnel, que personne ne verra, ou doit-on utiliser l'identifiant de connexion que le bénévole aura rempli lors de son inscription au site ? Nous avons choisi d'utiliser un identifiant unique pour tous les bénévoles s'incrémentant automatiquement à chaque inscription pour avoir des comptes compatibles avec tous les festivals.

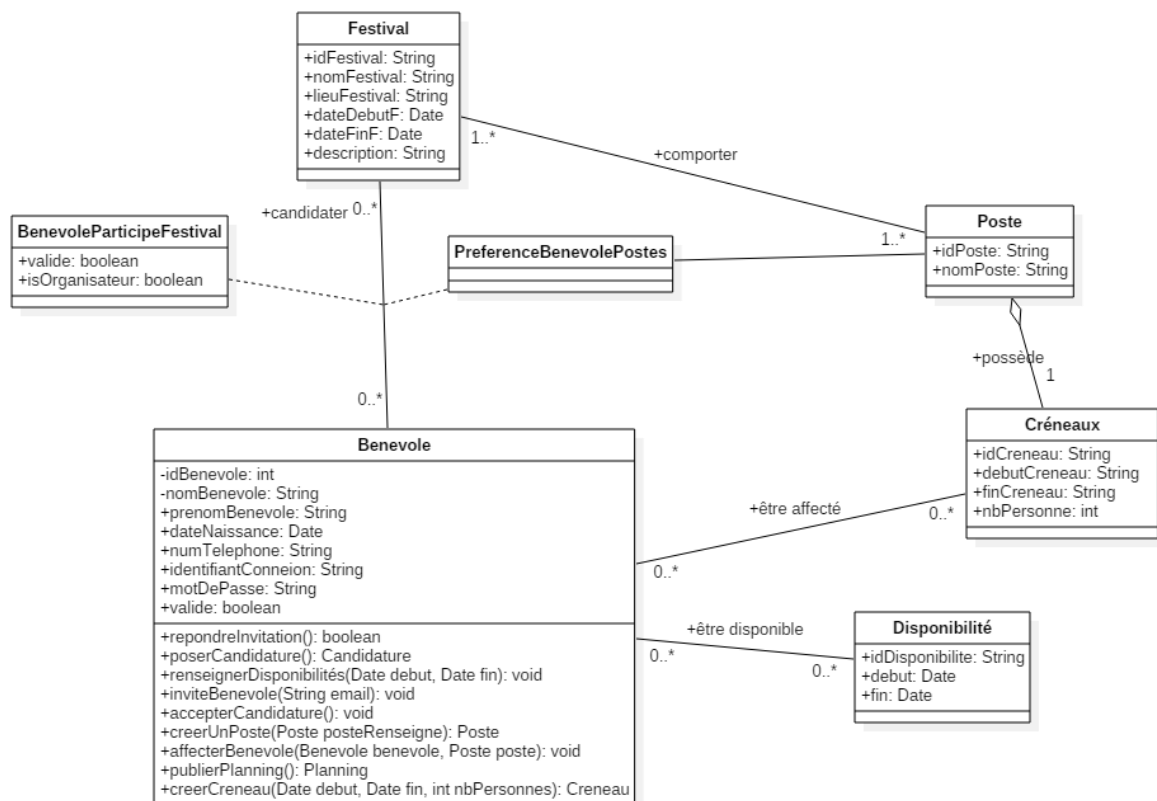


Figure 8. Diagramme de classe définitif, fait avec StarUML

La sécurité de la base de données étant importante, nous avons eu recours à plusieurs mécanismes de sécurisation, au niveau des mots de passe des utilisateurs, qui sont cryptés sur la base de données et renforcés à partir d'une *seed** pour éviter les attaques par dictionnaire.

De plus, pour garantir la fiabilité des utilisateurs, nous avons implémenté une inscription par courriel, au cours de laquelle il faut valider son compte depuis un lien envoyé par courriel.

Nous nous assurons ensuite que l'accès aux fonctionnalités des festivals ne soient disponibles qu'aux utilisateurs validés par les organisateurs, ces permissions sont gérées au niveau du contrôleur.

Concernant les difficultés rencontrées, nous avons commencé ce projet sans aucune connaissance en PHP, il a donc fallu un long temps d'adaptation pour maîtriser ce langage et le modèle MVC. Cela nous a laissé un temps conséquent pour la conception mais moins de temps pour la programmation. Une des principales difficultés techniques est de faire correspondre toutes les interfaces et permissions des comptes selon chaque festival.

Ensuite, l'étape des créneaux et des postes présentait ses propres contraintes, afin de faire coïncider les horaires d'un même bénévole pour plusieurs festivals différents et de lui générer un planning sans erreur.

Concernant le visuel du planning, nous pensions à l'implémentation d'un calendrier, mais malgré nos recherches nous ne sommes pas parvenus à exploiter les outils de génération de calendrier.

3. Résultats

Une fois que les phases de conception et de réalisation sont achevées, nous devons nous assurer que notre site de gestion de bénévoles d'un festival fonctionne correctement. Cette avant dernière-section met alors en place la validité de notre site. Nous expliquerons alors la façon dont on a testé, puis validé, notre site. Or, nous connaissons déjà notre application et nous voulons l'exporter de telle sorte qu'elle soit utilisée par de véritables organisateurs de festivals. Il faudra donc pouvoir expliquer comment notre site fonctionne.

3.1. Installation

Notre site internet ne nécessite pas d'installation spécifique. Il suffit seulement d'avoir une machine connectée, comme un ordinateur ou un smartphone, possédant une connexion internet.

3.2. Test/Validation

Afin de s'assurer que le site fonctionne bien, nous avons effectué deux sortes de séries de tests. La première se situe pendant la phase de développement, afin de s'assurer que le code écrit ne comporte pas de fautes de frappe ou de langage. Nous pouvons voir ici une erreur concernant l'enregistrement de données dans la base de données, pour pallier et corriger ce genre d'erreurs.

Ici, (Cf. Figure 9) une erreur récurrente concernant l'enregistrement de données dans la base de données.

```
Fatal error: Uncaught PDOException: SQLSTATE[42000]: Syntax error or access violation: 1064 You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'IDPoste) FROM Poste' at line 1 in /home/ann2/alarconj/public_html/ProjetS3/model/ModelPoste.php:71 Stack trace: #0 /home/ann2/alarconj/public_html/ProjetS3/model/ModelPoste.php(71): PDOStatement->execute() #1 /home/ann2/alarconj/public_html/ProjetS3/controller/ControllerPoste.php(55): ModelPoste::dernierPosteSave() #2 /home/ann2/alarconj/public_html/ProjetS3/controller/routeur.php(31): ControllerPoste::created() #3 /home/ann2/alarconj/public_html/ProjetS3/index.php(6): require_once('/home/ann2/alar...') #4 {main} thrown in /home/ann2/alarconj/public_html/ProjetS3/model/ModelPoste.php on line 71
```

Figure 9. Capture d'écran d'un exemple d'erreur courante

Pour pallier ce genre d'erreur il faut s'assurer que les données que l'on enregistre soient dans un format conforme, pour cela on utilise les attributs de la classe Poste qui doivent être les mêmes que sur la base de données.

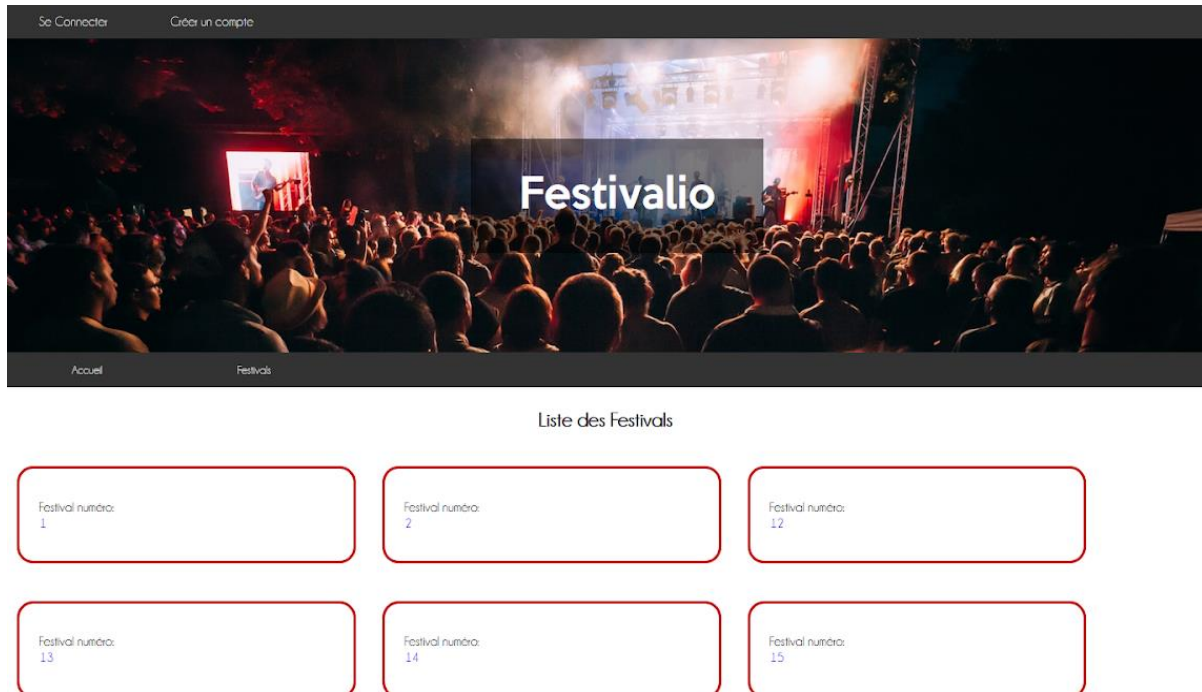
La deuxième phase se passe lorsque le site est fini. Nous avons alors demandé à plusieurs complices d'essayer différentes méthodes et leurs conséquences via une liste (Cf. Annexe 4 : Liste des fonctionnalités à tester) que nous demandons de remplir au fur et à mesure, en cochant si l'action se passe bien, ou mettre en remarque si une erreur devait se produire.

Pour réaliser cette liste de fonctionnalités, nous avons repris la plupart des éléments inscrits dans le cahier des charges, et plus particulièrement ceux des diagrammes de cas d'utilisation (Cf. Figure 2 et Figure 3), comme, par exemple, se connecter, accepter une candidature.

Si une erreur se produisait, nous allions l'analyser et la corriger au plus vite dès que le complice avait fini la série de tests. Par ailleurs, certains tests requéraient une certaine

expérience dans la programmation web (du niveau de seconde année de DUT informatique), nous n'avons alors pas hésité à demander à nos camarades de promotion de venir nous 'pirater', afin de vérifier la résistance aux attaques qui pourraient être malveillantes.

3.3. Manuel d'utilisation



3.3.1. Connexion et déconnexion

Figure 10. Capture d'écran de la page d'accueil du site

- Afin d'accéder au site de gestion de bénévoles de festival, il faut se connecter à internet via l'adresse <http://webinfo.iutmontp.univ-montp2.fr/~alarconj/ProjetS3/index.php>.
- La page d'accueil présentera alors la liste des noms festivals organisés sur le site (Cf. Figure 10).
- Le nom de chaque festival est un lien qui redirige vers une page avec les détails du festival (Cf. Figure 10), c'est-à-dire le nom du festival, le lieu, les dates ainsi qu'une petite description des événements qui auront lieu pendant le festival.
- Pour se connecter, un bouton en haut à gauche est présent "Se connecter", il faut alors rentrer son identifiant et son mot de passe et cliquer sur valider (Cf. Figure 10).
- Pour se déconnecter, il suffit d'appuyer sur le bouton "Se Déconnecter" en haut à gauche de la page (Cf. Figure 10).

3.3.2. Inscription au site

- Pour s'inscrire au site et pouvoir participer à l'organisation d'un festival, cliquer sur l'onglet "Créer un compte" (Cf. Figure 10) en haut à gauche de la page, à côté du bouton "Se Connecter", puis rentrer les informations suivantes : l'identifiant que l'on utilisera sur le site, son nom, son prénom, sa date de naissance, son courriel, son numéro de téléphone ainsi que son mot de passe. On demande deux fois le même mot de passe afin de d'assurer qu'on ne fait d'erreurs dedans. Ensuite cliquer sur le bouton envoyer, ce qui enverra un mail de confirmation de compte au courriel spécifié (Cf. Figure 10).

- D'après le mail de validation, cliquer sur le lien, qui redirigera sur la page d'accueil du site.

Toutes les manipulations suivantes demandent d'être connecté sur le site.

3.3.3. Modifier ses informations

- Sur la page d'accueil, dans l'onglet d'accueil, chaque bénévole peut cliquer sur l'onglet 'Mon Compte'. Ensuite arrivé à la page 'Mon Compte', celui-ci doit cliquer sur le lien modifier puis remplir les informations qu'il souhaite changer (mot de passe, courriel, nom, prénom, numéro de téléphone et date de naissance), puis il doit cliquer sur valider, afin que les changements soient pris en compte.

3.3.4. Créer un festival

- Chaque utilisateur ayant confirmé son compte peut créer un festival. Sur la page d'accueil du site, à la fin de la liste des festivals, cliquer sur le bouton "Créer un festival". Vous serez redirigé vers un formulaire à remplir où il faudra faire une description de votre festival, indiquer le lieu ainsi que la date de début et de fin du festival. Lorsque le festival est créé, il apparaîtra dans la liste des festivals. Vous y aurez le statut d'organisateur.

3.3.5. Poser une candidature pour un festival

- Sur la page d'accueil, cliquer sur le festival désiré, on arrive sur les détails du festival. Cliquer sur "Participer", une fois que la participation est validée par un organisateur du festival, un mail de notification vous est envoyé. Vous pouvez désormais accéder au portail du festival sur lequel vous aurez accès à vos fonctionnalités de bénévole.

Les fonctionnalités suivantes sont accessibles aux bénévoles acceptés d'un festival depuis le portail de ce festival, pour accéder à un portail il suffit de cliquer sur un festival depuis l'accueil.

3.3.6 Spécifier vos préférences et disponibilités

- Sur la page 'Choisir mes préférences et disponibilités' vous pouvez choisir les postes auxquels vous souhaitez postuler en cliquant sur "Candidater" à côté du poste désiré.
- Vous pouvez également émettre vos disponibilités sur la page 'Choisir mes préférences et disponibilités' en cliquant sur le lien "Émettre ou Modifier mes disponibilités". Après avoir cliqué sur ce lien, vous pouvez remplir un formulaire en indiquant la date de début et la date de fin de vos disponibilités.

3.3.7. Consulter la liste des organisateurs

- Depuis le portail d'un festival, vous trouverez un lien vers une liste des organisateurs de votre festival. Leur nom, prénom, adresse mail et leur numéro de téléphone y est spécifié pour les contacter.

- En tant qu'organisateur, vous aurez la possibilité de démettre un organisateur de ses fonctions en cliquant sur "Révoquer les droits".

3.3.8. Consulter votre planning de bénévole

- Depuis le portail de fonctionnalités, vous pouvez accéder à votre propre planning concernant le festival en cliquant sur le lien "Consulter votre planning", vos horaires en tant que bénévole y sont spécifiées avec le poste associé.

3.3.9. Postuler en tant qu'organisateur

- Si vous souhaitez obtenir le statut d'organisateur, vous pouvez en faire la demande depuis le portail de fonctionnalités d'un festival en cliquant sur "Postuler en tant qu'organisateur". Votre demande sera évaluée puis acceptée ou rejetée par un organisateur.

Les manipulations suivantes demandent d'être connecté, organisateur sur le festival sélectionné. Elles sont accessibles depuis le portail de fonctionnalités en tant qu'organisateur.

3.3.10. Interface des postes

- Ici sont enregistrés les différents postes d'un festival en cliquant sur le lien "Postes", vous pouvez les créer, les modifier et les supprimer grâce aux boutons correspondants.

3.3.11. Renseigner les besoins en créneaux

- Depuis l'interface d'un poste vous pouvez ajouter des besoins en créneaux en cliquant sur "Ajouter un créneau", vous devrez remplir un simple formulaire en indiquant l'heure de début et de fin, et le nombre de bénévoles nécessaires.

3.3.12. Affecter un poste à un bénévole et son créneau

- Depuis l'interface d'un poste, en cliquant sur un créneau vous obtiendrez une liste de bénévoles volontaires et disponibles pour ce poste, il vous suffira de cliquer sur "Affecter" pour l'ajouter à ce créneau de poste.
- Vous pourrez aussi le supprimer de ce poste par la suite en cliquant sur le bouton "Supprimer".

3.3.13. Accéder au planning d'un poste

- Depuis l'interface d'un poste en cliquant sur le lien "Planning", vous accéderez à une page permettant de voir l'état de tous les créneaux, en indiquant leurs horaires et si le quota de personnel est rempli.

3.3.14. Accepter les candidatures des bénévoles

- Vous pouvez cliquer sur le bouton "Liste des bénévoles à accepter". Une liste de bénévoles en attente vous sera présentée, il suffit de cliquer sur le bouton "Accepter" pour valider le bénévole.

3.3.15. Accéder à la liste des bénévoles

- Vous pouvez cliquer sur le bouton "Liste des bénévoles", une liste de bénévoles vous sera présentée.
- Pour voir les informations d'un bénévole, cliquez sur "Voir les informations" à côté du bénévole en question.

3.3.16. Accepter les candidatures des organisateurs

- En cliquant sur le lien "Liste des organisateurs à accepter", vous pouvez examiner les demandes effectuées par les bénévoles pour devenir organisateur, vous aurez le choix de l'accepter ou de les refuser.

4. Gestion de projet

Qu'est-ce un projet sans une équipe pour le réaliser ? Dans cette dernière partie, nous allons donc expliquer le côté humain du projet, c'est-à-dire la façon dont nous avons interagi entre nous afin de mener à bien notre site de gestion de bénévoles de festival.

4.1. Démarche personnelle

Nous avons travaillé en groupe où nous nous sommes réparti les tâches.

Nous avons vu notre tuteur de projet toutes les deux à trois semaines, ce qui nous laissait beaucoup d'autonomie. Il était difficile de synchroniser notre emploi du temps avec celui de M. COLETTA, et nous n'avons donc pu faire que quatre réunions pour parler de l'avancement du projet. Lors de ces réunions, nous avions pour habitude de présenter ce que nous avions réalisé pendant la période inter-réunions, puis nous évaluions les points forts et les points faibles et comment améliorer ces derniers. A la fin des réunions, M. COLETTA nous expliquait les objectifs à accomplir qu'il attendait de nous pour la prochaine réunion. Cela nous a permis d'émettre un rapport complet de chaque réunion (cf. Annexe 5 : Réunion 1 : Présentation du projet « Application de Gestion de Bénévoles d'un Festival »,

Annexe 6 : Réunion 2 : Présentation des maquettes, Annexe 7 : Réunion 3 : Présentation de la base de données et du modèle MCV et Annexe 8 : Réunion 4 : Présentation du CRUD*), dont nous avons pu nous servir pour établir les tâches à effectuer pour chaque sprint.

Pour trouver des informations, nous les avons essentiellement cherchées sur le site de M. Romain LEBRETON en collaboration avec M. Sébastien GAGNE et M. Abdelkader GOUAÏCH¹, qui regroupe tous les TD et TP correspondant au langage de programmation web (HTML/CSS/PHP). Par ailleurs, M. GAGNE nous a aidés à corriger nos erreurs et à surmonter nos difficultés. Nous n'avons d'ailleurs pas hésité à consulter des sites sur internet, comme www.php.net ou encore www.w3schools.com, pour des informations complémentaires, vis-à-vis du code en lui-même ou des informations complémentaires sur le sujet du projet.

Pour la réalisation du projet nous avons utilisé différents logiciels que nous allons énumérer.

Tout d'abord, pour la gestion de notre base de données, nous avons utilisé le site phpMyAdmin qui est une application Web gratuite et implémentée principalement en PHP pour les systèmes de gestion de base de données MySQL². C'est sur cette application que toutes les modifications de la base de données de notre site y sont faites.

Ensuite, pour éditer le code de notre projet en PHP et HTML/CSS, nous avons utilisé le logiciel Sublime Text, un éditeur de texte générique, permettant d'afficher le répertoire dans lequel nous travaillons ainsi que ses sous-répertoires.

Afin d'effectuer des recherches documentaires, des tests sur notre projet et de tester la mise en page de notre site, nous avons utilisé les navigateurs Google Chrome et Mozilla Firefox, afin d'être sûrs que la mise en page soit conforme sur plusieurs navigateurs.

De plus, étant donné que nous avons utilisé une base de données reliée à une session de l'IUT d'un de nos membres de notre groupe (Julian Alarcon), cela impliquait que le code du site, selon l'architecture MVC*, soit obligatoirement placé dans le dossier public_html. Ainsi nous avons utilisé le logiciel FileZilla afin d'accéder à nos sessions depuis nos domiciles. Nous avons aussi utilisé Google Drive qui est un service de partage et de stockage de fichiers dans le Cloud (dans lequel nous avons créé un dossier partagé aux membres du groupe) afin d'y partager différents documents tels que les maquettes, les diagrammes, les comptes-rendus de réunions, ou encore le rapport de projet et plus encore.

Par ailleurs, afin de réaliser des diagrammes UML ³(Unified Modeling Language) précis, nous avons utilisé deux logiciels de modélisation UML, StarUML et Modelio. StarUml est un logiciel qui nous a permis de concevoir des diagrammes simples, comme les diagrammes de classe (Cf Figure 6, Figure 7 et) et les diagrammes de cas d'utilisation (Cf Figure 2 et Figure 3). Nous avons également utilisé le logiciel Modelio pour concevoir un diagramme d'activités (Cf Figure 16).

¹ Voir le site [romainlebreton](http://romainlebreton.com) pour plus d'informations

² Système de gestion de bases de données relationnelles (SGBDR)

³ D'après Wikipédia : le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (UML), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.

Pour l'écriture du rapport et sa mise en page, nous avons utilisé Google Docs et Microsoft Word. Google Docs permet de faire du traitement de texte et fait partie de la suite bureautique qui est accessible via Google Drive. Word est un logiciel de traitement de texte de la suite Microsoft Office. Le document Google Docs a servi pour réaliser les premiers essais d'écritures alors que le document Microsoft Word a lui servi à la mise en page et le texte définitif.

Enfin, afin de permettre une communication constante au sein du groupe même en dehors des cours, nous avons utilisé le système de messagerie Facebook Messenger dans lequel nous avons créé un groupe de travail. Nous communiquons aussi via les documents Google Docs dans Google Drive, c'est à dire que nous pouvions communiquer via les commentaires et le chat intégré dans les documents.

4.2. Planification des tâches

La planification des tâches pour notre groupe ne comprenait ni de Trello, ni de diagramme de Gantt. En effet, nous sommes dans la même classe, nous permettant ainsi d'avoir pu réaliser d'autres projets ensemble, et nous sommes mis d'accord sur la marche à suivre sur la plupart des projets. Nous aimons aussi travailler avec des éléments des méthodes agiles, telles que les réunions journalières, où chacun annonçait le travail accompli et ce qui l'attendait pour les tâches du jour même.

Comme nous voyions M. COLETTA toutes les deux à trois semaines et à chaque réunion, il nous donnait alors les objectifs pour la fois prochaine, par exemple faire des recherches sur les calendriers électroniques pouvant être implémentés sur un site web, ou encore modifier la base de données afin de l'optimiser. (Cf. Annexes des réunions). Quand bien même nous ne nous sommes rencontrés que quatre fois, nous prenions toujours le temps de poser nos questions et M. COLETTA nous répondait avec précision. Il nous annonçait alors les objectifs du projet et ceux qu'il attendait pour la fois d'après de manière explicite. Par ailleurs, dès le début M. COLETTA avait annoncé que nous devions nous concentrer sur le fond du site et non sur la forme, ce que nous avons privilégié jusqu'à la fin du développement de l'application, puis nous nous sommes penchés sur les éléments de mise en forme du site, c'est-à-dire le développement du contenu des vues et du code du CSS.

Même si nous nous entendions bien sur la répartition des tâches, nous avons tout de même fait une liste reprenant tous les éléments qu'il manquait, afin de n'en oublier aucun, que ce soit au niveau de la programmation ou du rapport. Nous nous sommes ainsi répartis les tâches de façon équilibrée, où chacun mettait en avant ses meilleures compétences, afin d'être efficace et rapide.

Cependant, nous avons eu à faire face certaines difficultés, ce qui a perturbé notre rythme de travail.

Premièrement, nous avons eu beaucoup de projets au semestre trois, comme l'implémentation d'un système d'exploitation pour le cours d'architecture des systèmes d'exploitation, ou bien le projet de programmation web qui demandait beaucoup de travail en un temps réduit. Le seul point positif de ce projet est que nous avons pu nous en inspirer pour réaliser celui de gestion de bénévole de festival, ou du moins pour les méthodes génériques.

Deuxièmement, il était difficile de coordonner nos plannings avec M. COLETTA. En effet nous n'avons pu nous voir que quatre fois, mais à chaque fois nous prenions le temps de poser nos questions et M. COLETTA nous répondait avec précision et il nous annonçait les objectifs du projet et ceux qu'il attendait pour la fois d'après de manière explicite.

Par ailleurs, les objectifs initiaux n'incluaient pas la partie mise en page de l'application web, et nous n'avons pas pu voir M. COLETTA entre temps, donc nous avons dû faire au mieux les éléments graphiques, qui devaient alors respecter les règles d'ergonomie de site web. Ce dernier point demandait beaucoup d'efforts de la part de l'équipe, donc nous avons fait en sorte de faire une mise en page simple et efficace, afin de concentrer notre énergie sur les tâches plus importantes.

Enfin, nous avons eu difficultés de logistique. L'outil GitHub nous était inconnu au début du projet, ce qui aurait pu nous mettre en difficulté par rapport aux différentes versions que chacun possédait. Nous avons alors décidé de transférer les documents sur le serveur de l'IUT sur la session de Julian ALARCON, ainsi que la base de données qui a été transféré sur son compte phpMyAdmin. Chacun a alors pu modifier les documents à sa guise, via les ordinateurs du département informatique de l'IUT de Montpellier, ou via un ftp Filezilla, où nous avons accès à tous les documents.

4.3. Bilan critique par rapport au cahier des charges

Nous avons fait en sorte de répondre au cahier des charges, comme convenu, notre interface prend en charge différents types d'utilisateurs et leurs droits pour plusieurs festivals respectifs. Ces utilisateurs ont accès à la gestion de leur compte et peuvent tous créer leur propre festival. Nous offrons aux utilisateurs un panel de fonctionnalités nécessaires lors de l'organisation du festival conformément aux diagrammes de cas d'utilisation établis. Nous gérons aussi les candidatures des bénévoles aux festivals, pour que seuls les utilisateurs approuvés par les organisateurs puissent avoir accès aux fonctionnalités.

Il faut noter que nous ne sommes pas parvenus à modéliser les données sous forme graphique à l'aide d'un calendrier. Nous avons donc opté pour une solution plus primaire, c'est à dire que chaque bénévole a accès à sa propre liste avec ses propres créneaux et chaque organisateur a accès à une liste de tous les créneaux de son festival.

Nous avons en contrepartie pris l'initiative d'ajouter quelques fonctionnalités nécessaires à un site web dynamique au niveau de la sécurité, avec la sécurisation des mots de passe et la validation de l'inscription grâce à l'envoi d'un mail.

Notre projet est aisément modifiable pour un développeur externe, nous avons fait en sorte d'avoir une base de données sans redondances comprenant toutes les associations nécessaires. Concernant le code nous avons fait en sorte de respecter les fondements de l'architecture MVC tout en ajoutant un modèle générique. Ainsi, n'importe quel programmeur connaissant le fonctionnement MVC peut comprendre le fonctionnement de notre site. De plus le code est dûment commenté pour favoriser la compréhension.

On peut envisager de multiples évolutions possibles pour ce projet, étant donné que nous nous sommes majoritairement focalisés sur la programmation côté serveur, l'aspect graphique est améliorable, on peut aussi penser à l'ajout d'un plugin pour l'affichage des plannings.

Conclusion

Concernant les résultats obtenus nous proposons un site web de gestion de bénévoles pour des festivals gratuit. Il permet aux organisateurs de créer des festivals avec un large panel de fonctionnalités à leur disposition. De nombreux objectifs ont été atteints tel que la connexion et la gestion des informations des acteurs, la création de festival et toutes les fonctionnalités de gestion qui en découlent (gérer les deux catégories d'acteurs en fonction des festivals, accepter la candidature d'un bénévole ou encore affecter un bénévole à un créneau).

Seule l'implémentation d'un calendrier n'a pas été réalisée car celle-ci nécessitait des connaissances dans des langages alternatifs tels que l'Ajx ou le javascript, enseignements que nous n'avons pas eus jusque lors.

Ce site web présente des possibilités d'évolutions, en grande partie sur l'aspect graphique et par rapport au responsive design. C'est à dire un site qui s'adapte à toute sorte de plateforme technique.

Sur le plan technique ce projet nous a apporté une maîtrise plus profonde du langage PHP et du modèle MVC, notamment à travers la mise en place du modèle générique.

C'était également l'occasion d'exploiter nos connaissances en base de données pour fournir une base de données complexe sans redondances.

Nous avons également pu renforcer nos méthodes de travail, ce projet étant notre premier travail de groupe d'envergure, nous avons pu comprendre que la communication joue un rôle clé dans le déroulement d'un projet. Nous avons également constaté que la phase de conception a été primordiale afin de s'orienter vers une vision commune remplissant les exigences demandées. Une autre grande partie du travail d'organisation a été la tenue à jour de la documentation et des rapports de réunion, qui nous a aidé aussi bien pour notre rapport que dans la phase de réalisation.

Ainsi ce projet fut une opportunité pour exploiter nos compétences techniques et organisationnelles, ainsi qu'une expérience enrichissante pour tous les membres de l'équipe.

Bibliographie

- [1] « Volunteo - Gestion des bénévoles, création de planning, organisation du bénévolat pour un festival ou un événement ». [En ligne]. Disponible sur : <https://volunteo.com/fr/>. [Consulté le : 05-déc-2018].
- [2] « Site d'aide à la gestion des bénévoles de votre événement ». [En ligne]. Disponible sur : <https://www.vorg.fr/>. [Consulté le : 05-déc-2018].
- [3] « PHP: Hypertext Preprocessor ». [En ligne]. Disponible sur : <http://php.net/>. [Consulté le : 02-janv-2019].
- [4] « Programmation Web - Côté Serveur ». [En ligne]. Disponible sur : <http://romainlebreton.github.io/ProgWeb-CoteServeur/>. [Consulté le : 05-janv-2019].
- [5] « Les 12 règles de l'ergonomie Web [Partie 1] - Ludis Media ». [En ligne]. Disponible sur : <http://ludismedia.com/12-regles-ergonomie-web/>. [Consulté le : 07-janv-2019].
- [6] « Analyse de l'ergonomie d'un site web », WebMarketing. [En ligne]. Disponible sur : <http://alichabbouh.unblog.fr/2011/03/03/analyse-de-l-ergonomie-dun-site-web/>. [Consulté le : 07-janv-2019].
- [7] « W3Schools Online Web Tutorials ». [En ligne]. Disponible sur : <https://www.w3schools.com/>. [Consulté le : 08-janv-2019].
- [8] « phpMyAdmin — Wikipédia ». [En ligne]. Disponible sur : <https://fr.wikipedia.org/wiki/PhpMyAdmin>. [Consulté le: 08-janv-2019].
- [9] « phpMyAdmin », Wikipédia. 25-août-2018.
- [10] « MySQL », Wikipédia. 13-nov-2018.
- [11] « Sublime Text - A sophisticated text editor for code, markup and prose ». [En ligne]. Disponible sur : <http://www.sublimetext.com/>. [Consulté le : 08-janv-2019].
- [12] « Sublime Text », Wikipédia. 24-juill-2018.
- [13] « Google Drive », Wikipédia. 19-nov-2018.
- [14] « Google Docs, Sheets, Slides et Forms », Wikipédia. 09-juill-2018.
- [15] « Modelio Open Source - UML and BPMN free modeling tool », Modelio Open Source. [En ligne]. Disponible sur : <https://www.modelio.org/>. [Consulté le : 08-janv-2019].
- [16] « Modelio », Wikipédia. 03-oct-2018.
- [17] « StarUML ». [En ligne]. Disponible sur : <http://staruml.io/>. [Consulté le : 08-janv-2019].
- [18] « StarUML », Wikipédia. 17-oct-2018.
- [19] « PHP : Calendrier simple, facilement modifiable, affichage complet - CodeS SourceS ». [En ligne]. Disponible sur : <https://codes-sources.commentcamarche.net/source/42671-calendrier-simple-facilement-modifiable-affichage-complet>. [Consulté le: 08-janv-2019].
- [20] « A Calendar with PHP & MySQL | Coding with Sara | Coding with Sara ». [En ligne]. Disponible sur : <https://codingwithsara.com/a-calendar-with-php-mysql/>. [Consulté le: 08-janv-2019].

- [21] « A Calendar with PHP (starting with Monday) | Coding with Sara | Coding with Sara ». [En ligne]. Disponible sur : <https://codingwithsara.com/a-calendar-with-php-starting-with-monday/>. [Consulté le : 08-janv-2019].
- [22] jamelbaz, Fullcalendar PHP MySQL. .
- [23] « Intégration de FullCalendar avec PHP-MySQL - Jamelbaz ». [En ligne]. Disponible sur : <http://jamelbaz.com/tutos/integration-de-fullcalendar-avec-php-mysql>. [Consulté le : 08-janv-2019].
- [24] « Introduction · Bootstrap ». [En ligne]. Disponible sur : <https://getbootstrap.com/docs/4.2/getting-started/introduction/>. [Consulté le : 08-janv-2019].
- [25] « Download | FullCalendar ». [En ligne]. Disponible sur : <https://fullcalendar.io/download>. [Consulté le : 08-janv-2019].
- [26] « YouTube ». [En ligne]. Disponible sur : <https://www.youtube.com/watch?v=t0s7ycR1Ib8>. [Consulté le : 08-janv-2019].
- [27] Grafikart.fr, Tutoriel PHP/MySQL : Créer un Calendrier 2/3. .
- [28] « YouTube ». [En ligne]. Disponible sur : <https://www.youtube.com/watch?v=qzE949YCYic>. [Consulté le : 08-janv-2019].
- [29] « Tutoriel PHP/MySQL : Créer un Calendrier 3/3 - YouTube ». [En ligne]. Disponible sur : <https://www.youtube.com/watch?v=wOoyj711EnQ>. [Consulté le : 08-janv-2019].
- [30] Grafikart.fr, Tutoriel MySQL/PHP : Créer un Calendrier 4/3.
- [31] Grafikart.fr, Tutoriel PHP : Créer un calendrier.
- [32] « Liste des Festivals ». [En ligne]. Disponible sur : <http://webinfo.iutmontp.univ-montp2.fr/~alarconj/ProjetS3/index.php> . [Consulté le : 08-janv-2019].
- [33] « UML ». [En ligne]. Disponible sur : [https://fr.wikipedia.org/wiki/UML_\(informatique\)](https://fr.wikipedia.org/wiki/UML_(informatique)). [Consulté le : 02-janv-2019].

Annexes techniques

Annexe 1 : Maquettes du site

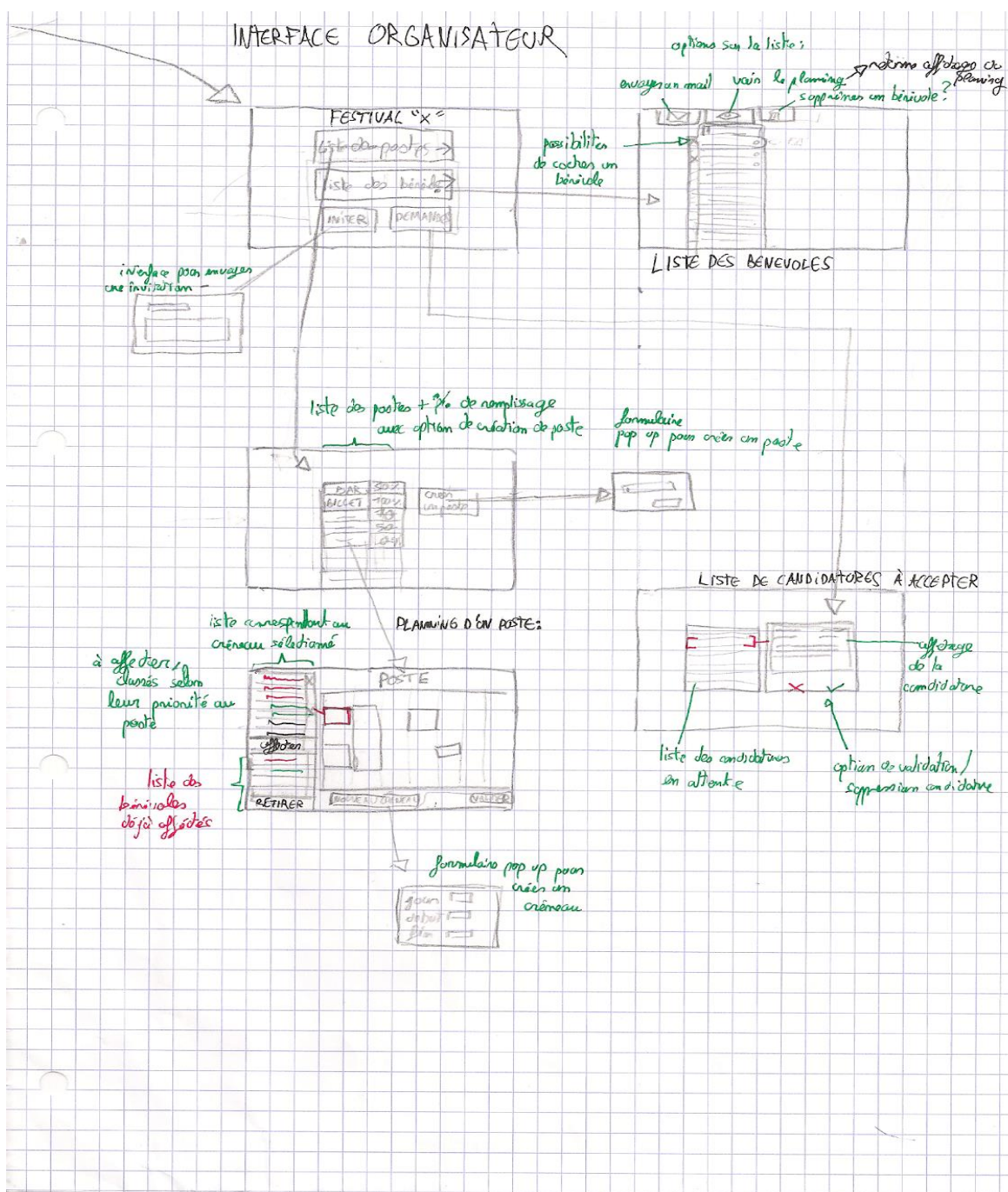


Figure 11. Numérisation des maquettes du site (1)

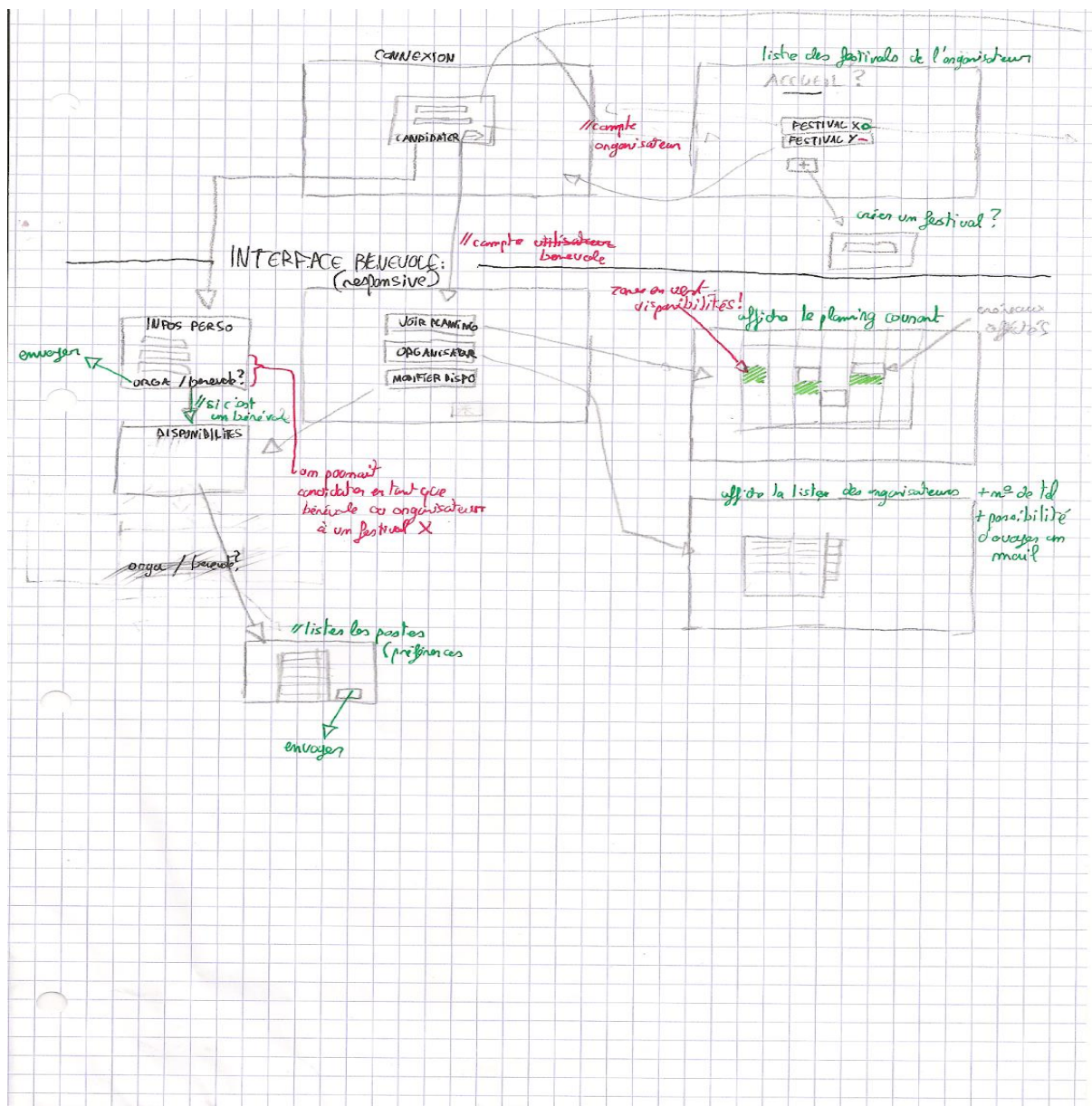


Figure 12. Numérisation des maquettes du site (2)

Annexe 2 : Analyse de contexte du site Volunteo



Nom et adresse :

Volunteo (<https://volunteo.com/fr/>)

Description du site :

Volunteo est un programme de gestion de bénévoles pour des événements, par exemple un festival, qui est payant (les dix premiers bénévoles offerts puis un euro par bénévole). Cette application web nous permet la gestion facile des volontaires en déléguant les différentes tâches aux chefs d'équipes. Ainsi la mise en place des autres équipes est indépendante des autres. Il permet aussi des prises de contacts rapides et efficaces entre l'administrateur et les utilisateurs (bénévoles et organisateurs), et entre utilisateurs. D'après ce site, un chef d'équipe peut être un simple bénévole dans une autre équipe (c'est à dire qu'il ne sera pas chef d'équipe dans cette dernière). Par ailleurs, la page d'un bénévole contient toutes les informations qui le concerne, comme ses coordonnées, ses préférences pour certaines équipes, ses disponibilités et les postes auxquels il est affecté. Cette page est, par la suite, accessible par l'administrateur, ses chefs d'équipes et par le bénévole concerné. Ce programme de gestion de bénévoles contiendra d'autres fonctionnalités importantes, que nous détaillerons par la suite.

Types d'utilisateurs :

Responsable de l'événement (administrateur), Bénévole, Chef d'équipe

Fonctions du site en lien avec la demande du client :

Les fonctions en italiques sont celles qui sont communes à notre site

Utilisateurs :	Fonctions :
Administrateur	<ul style="list-style-type: none"> - <i>Créer des bénévoles</i> - <i>Créer un poste</i> - <i>Créer une Équipe</i> - <i>Affecter un poste</i> - <i>Ajouter une sous-équipe</i> - <i>Inviter de nouveaux bénévoles</i> - Valider les bénévoles - Créer des documents - Visualiser l'ensemble des équipes de l'événement - Visualiser l'ensemble des bénévoles de l'événement - Consulter le journal (historique des affectations/de validation des bénévoles ...) - Modifier les droits sur un utilisateur (affecter la responsabilité d'une ou plusieurs équipes à un utilisateur et inversement) - Acheter des crédits bénévoles
Administrateur, Chef d'Équipe et Bénévole	<ul style="list-style-type: none"> - <i>Consulter sa page Bénévole</i> - <i>Se connecter</i> - <i>Se déconnecter</i> - <i>Consulter son profil</i> - Envoyer un courriel - Choisir les équipes qu'on souhaite rejoindre - Émettre ses disponibilités - Contacter les responsables de l'événement - Contacter la plateforme technique Volunteo
Bénévole	<ul style="list-style-type: none"> - S'inscrire

Ergonomie du Site
<p>Le site est simple et agréable à l'œil. Il a une taille adaptée à l'écran, s'affiche facilement sur tous les navigateurs et est accessible à tous les internautes (à condition de payer) et sur toutes les plateformes techniques. Le menu est facilement identifiable et possède un moteur de recherche pour trouver un poste, un bénévole ou une équipe. Les pages sont homogènes : le nombre de couleurs est limité et celles-ci aident à la navigation. Par exemple, le rouge est utilisé quand le taux de remplissage d'une équipe ou des créneaux n'est pas atteint à 100%, et inversement, le vert sera utilisé lorsque le remplissage sera de l'ordre de 100%, ainsi que pour les fonctionnalités liées à l'administrateur. Les informations du site sont compréhensibles par les utilisateurs, remplissant pratiquement tous les critères ergonomiques que nous allons détailler ci-dessous.</p>

Critères ergonomiques du site	
Architecture : Site bien rangé	Oui
Bonne organisation visuelle	Oui (la page d'accueil est un résumé de l'événement pour le(s) responsable(s) de l'événement.)
Site facile d'utilisation pour les utilisateurs	Facile d'utilisation pour le(s) responsable(s) d'après la démonstration (démonstration avec le point de vue d'un administrateur seulement) mais on ne sait pas pour les chefs d'équipes et les bénévoles.
Respect des conventions (fonctions/principes utilisés par 50 à 79% des sites) et des standards (fonctions/principes utilisés par 80% des sites)	Oui (Les fonctions s'inscrire, se connecter, se déconnecter, créer un poste, créer une équipe/sous-équipe, valider un bénévole, consulter son profil, consulter sa page bénévole, contacter les bénévoles et chef d'équipes, inviter des bénévoles et envoyer un courriel sont présentes.)
Le site informe et répond efficacement	Oui (Le site possède une rubrique "A propos de nous" tout en bas de la page avec des sous-rubriques, la page d'un bénévole contient toutes les informations qui lui sont liés, le profil, la page d'accueil pour le(s) responsable(s) permet de faire un résumé de leur événement, ...)
Le site est compréhensible par les utilisateurs	Oui

Le site aide implicitement et explicitement les utilisateurs	Oui (Implicitement : menu avec les différentes rubriques, couleurs, clics logiques, boutons explicites. Explicitement : rubrique “Aide” tout en haut à droite à côté de la rubrique “Mon Compte” avec des sous-rubriques “Contact”, “Mise en place de l’événement” et “Questions fréquentes”).
Le site gère les erreurs des utilisateurs	Oui (L’utilisateur peut toujours modifier ou supprimer l’erreur qui a été faite.)
L’utilisateur arrive à accomplir une tâche rapidement	Tout dépend de si la tâche est spécifique ou générale (la plus longue tâche est de 6 clics)
L’utilisateur a l’impression qu’il est libre lorsqu’il navigue sur le site et qu’il commande	Oui (Le(s) responsable(s) est/sont le(s) administrateur(s) de l’événement, le(s) bénévole(s) a/ont aussi beaucoup de fonctionnalités qui leur sont liées, comme émettre leurs disponibilités, faire des souhaits par rapport aux équipes, ...)
Le site est facile d’accès (accessibilité physique : les internautes ciblés et les autres internautes, accessibilité technologique : sur toutes les plateformes)	Oui (N’importe quel internaute peut créer un événement à condition qu’il paye, le site est accessible sur toutes les plateformes.)

Points positifs :	Points négatifs :
Offre d’un panel de fonctionnalités assez large pour le(s) responsable(s) de l’événement.	Pas de calendrier
Communication facile avec l’assistance technique et entre les utilisateurs.	Payant
Remplissage de presque tous les critères d’un site ergonomique.	

Rapide prise en main du site pour les utilisateurs.	
Notifications fréquentes par courriel par rapport à l'événement (envoyé par le site).	

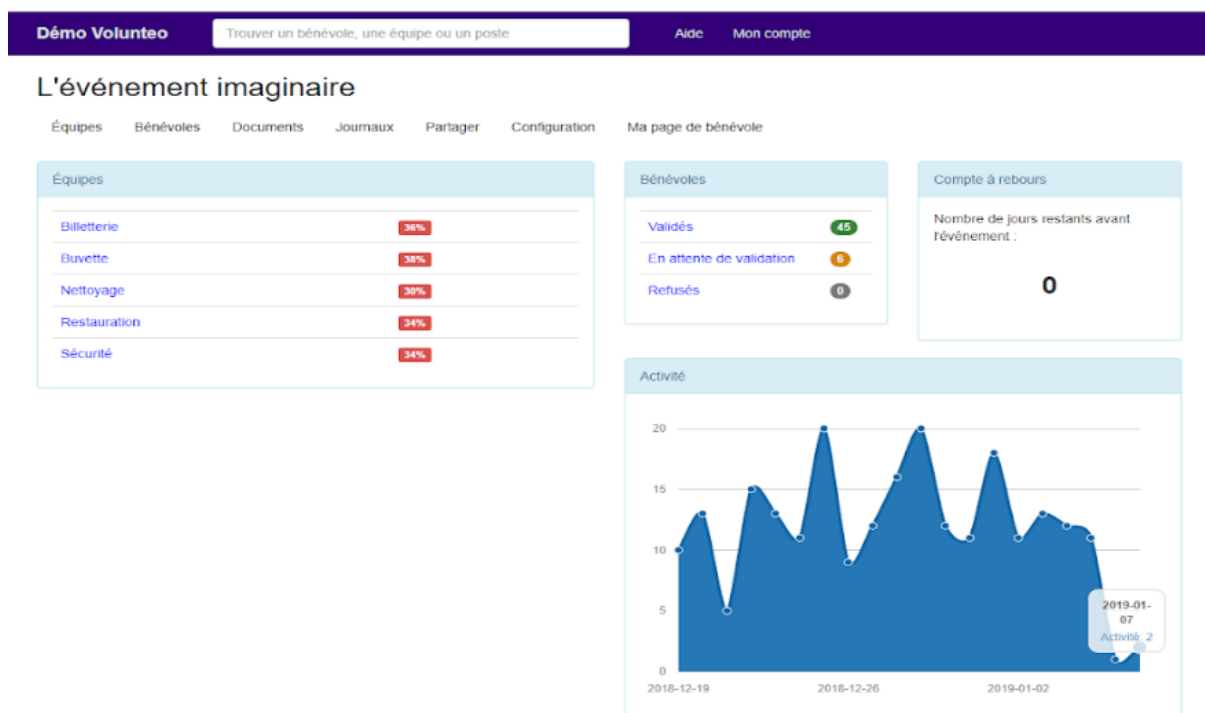


Figure 13. Capture d'écran de la page d'accueil de Volunteo, du point de vue administrateur depuis la démonstration du site

Nous pouvons retenir que ce site a des aspects à la fois positifs et négatifs. En effet, ce site offre un panel de fonctionnalités assez large pour le(s) responsable(s) de l'événement. Par ailleurs, la prise en main du site est rapide et celui-ci permet une communication efficace avec l'assistance technique et entre les utilisateurs. De plus, Volunteo suit pas à pas la vie de l'événement et informe constamment le(s) responsable(s). Néanmoins, le programme de gestion de bénévoles est payant et ne met pas à la disposition des utilisateurs un calendrier de l'événement.

Nous pouvons en conclure que le programme de gestion de bénévoles Volunteo est assez complet. Néanmoins, la démonstration ne nous permet pas de voir le point de vue d'un bénévole ou d'un chef d'équipe. Plusieurs événements de toutes tailles choisissent Volunteo pour simplifier leur organisation. Pour ce qui est de notre projet, nous nous appuyons sur des fonctionnalités sur lesquels Volunteo est construit car il y a certaines ressemblances à ne pas laisser de côté.

Démo Volunteo

Trouver un bénévole, une équipe ou un poste

Aide

Mon compte

L'événement imaginaire / Bénévoles

L'événement imaginaire

Équipes

Bénévoles

Documents

Journaux

Partager

Configuration

Ma page de bénévole

E-mail

Importer

Exporter

Colonnes

+ Créer un bénévole

Page d'inscription

Validés

Période d'affectation








Nom	Date d'inscription	Droits	Postes affectés
 Adrien Fleury	27/11/2018		6
 Adrien Rodrigues	03/12/2018		6
 Alain Marion	21/11/2018		7
 André Nicolas	17/11/2018		6
 Arthur Chartier	13/11/2018	Resp. équipes Restauration, Sécurité	7
 Audrey Mary	07/11/2018		7
 Auguste Auger	06/12/2018		6

Figure 14. Capture d'écran de la page des équipes de Volunteo, du point de vue administrateur depuis la démonstration du site

Démo Volunteo

Trouver un bénévole, une équipe ou un poste

Aide

Mon compte

L'événement imaginaire

Équipes

Bénévoles

Documents

Journaux

Partager

Configuration

Ma page de bénévole

Exporter

Ajouter une équipe

Nom	Début	Fin	Nombre de sous-équipes	Nombre de postes	Taux de remplissage	Responsables
Billetterie	mercredi 6 février 2019 08:00	dimanche 10 février 2019 02:00	2	4	38%	Élodie Mathieu
Buvette	mercredi 6 février 2019 08:00	dimanche 10 février 2019 02:00	2	4	38%	Nicole Leveque
Nettoyage	mercredi 6 février 2019 08:00	dimanche 10 février 2019 02:00	2	4	30%	Benjamin Aubry
Restauration	mercredi 6 février 2019 08:00	dimanche 10 février 2019 02:00	2	4	34%	Arthur Chartier
Sécurité	mercredi 6 février 2019 08:00	dimanche 10 février 2019 02:00	2	4	34%	Arthur Chartier

Figure 15. Capture d'écran de la page des bénévoles de Volunteo, du point de vue administrateur depuis la démonstration du site

Annexe 3 : Diagramme d'activités

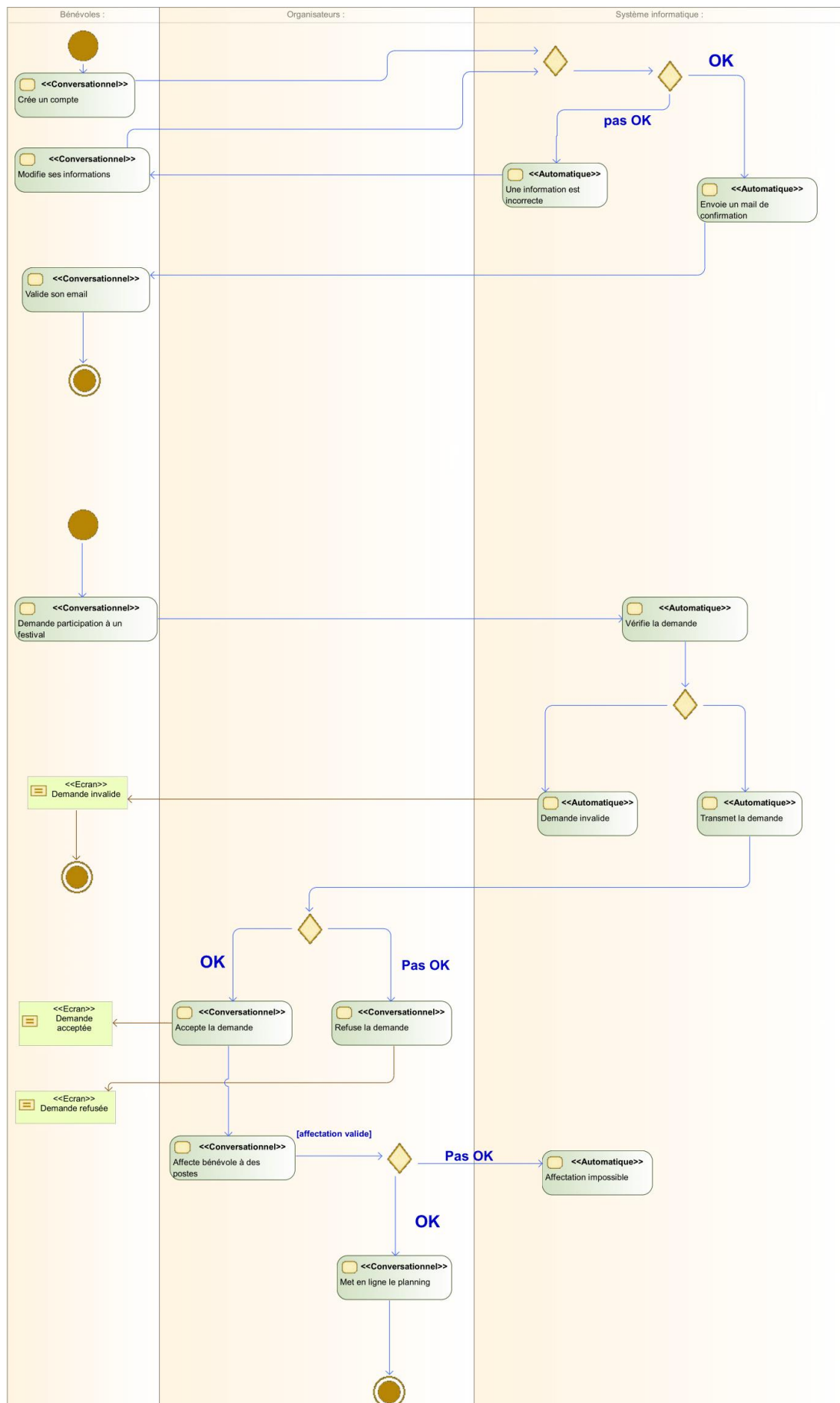


Figure 16. Diagramme d'activités, fait avec Modelio

Annexe 4 : Liste des fonctionnalités à tester

Quand on est externe au site

	Voir la liste des festivals
	S'inscrire
	Remplir ses informations pour l'inscription
	Remplir ses préférences de postes
	Recevoir un courriel de validation de compte

Quand on est bénévole

	Modifier ses informations personnelles
	Voir la liste de tous les festivals et leurs détails
	Poser une candidature pour un festival
	Remplir les disponibilités pour ce festival
	Voir les festivals auxquels le bénévole est affecté
	Voir les informations des organisateurs d'un festival auquel le bénévole est affecté
	Consulter son planning personnel pour un festival auquel le bénévole est affecté
	Consulter le planning général d'un festival auquel le bénévole est affecté
	Créer un festival et en devenir l'organisateur

Quand on est organisateur pour un festival

	Renseigner les besoins en postes
	Renseigner les besoins en créneaux et en nombre de personnes
	Inviter des bénévoles externes par courriel
	Accepter les candidatures des bénévoles sur le site
	Remplir le planning pour son festival
	Publier ce planning
	Modifier le planning déjà publié
	Accéder à la liste des bénévoles du festival qu'il organise et leurs informations
	Accéder à la liste des autres organisateurs du festival

Annexe 5 : Réunion 1 : Présentation du projet « Application de Gestion de Bénévoles d'un Festival »

Date : 18 septembre 2018

Présents : Julian ALARCON, Tanguy CADIEUX, Rémi COLETTA, Célia GROSCH, Samira YOUNESY

Présentation des collaborateurs :

M. Rémi COLETTA, professeur à l'IUT à Montpellier

M. Julian ALARCON, M. Tanguy CADIEUX, Mme Célia GROSCH et Mme Samira YOUNESY, étudiants en deuxième année de DUT informatique à l'IUT de Montpellier

Présentation du projet :

Ce projet doit aboutir à un système de gestion de bénévoles d'un festival, comportant deux parties : le côté organisateur et le côté festival. Le bénévole doit pouvoir rentrer ses informations personnelles, ainsi que ses préférences de postes et ses créneaux disponibles.

L'organisateur doit pouvoir créer des postes avec leurs créneaux respectifs et avoir la possibilité de les remplir.

Enfin, chacun d'eux doit avoir accès à son planning personnel et le planning global.

Plusieurs versions sont attendues, mais seule la première est la plus détaillée.

En effet, M. COLETTA exprime les besoins de chaque rôle, que les étudiants ont regroupé dans des diagrammes de cas d'utilisation, ci-dessous. (Cf. Figure 2 et Figure 3)

L'application n'étant pas réservée aux festivaliers mais aux bénévoles, les graphismes peuvent rester simples dans la limite où tous les éléments sont clarifiés et harmonieux.

De plus, les organisateurs doivent pouvoir ajouter des contraintes au niveau des créneaux, des postes, ... ainsi que celles que les bénévoles ont ajouté eux-mêmes lors du remplissage du formulaire.

La deuxième version du projet pousse les limites du planning plus loin. En effet cette version doit pouvoir aider l'organisateur dans son remplissage du planning, par exemple, un code couleur pour annoncer qui veut un certain poste, (vert si c'est un premier vœu, orange si c'est un deuxième, ...) si le bénévole est déjà affecté à un autre poste (couleur rouge) ou s'il n'est pas disponible ne pas le voir du tout.

De plus, des contraintes de repos pourront être appliquées, par exemple si un bénévole est présent jusqu'à deux du matin, il ne pourra pas revenir avant midi le lendemain, ...

Pour l'organisation des réunions, M. COLETTA souhaite qu'on se réunisse tous les quinze jours, voire toutes les semaines si le planning nous le permet.

Le projet final étant à rendre début janvier, M. COLETTA souhaitera que l'on rende la version 1 avant les vacances de la Toussaint, c'est-à-dire la partie de l'affichage du contenu des plannings, ainsi que les données des côtés organisateurs et bénévoles.

Pour la prochaine réunion, M. COLETTA demande deux éléments à faire, la base de données ainsi que des maquettes pour montrer l'enchaînement du site.

Annexe 6 : Réunion 2 : Présentation des maquettes

Date : 04 octobre 2018

Présents : Julian ALARCON, Tanguy CADIEUX, Rémi COLETTA, Célia GROSCH, Samira YOUNESY

Pour cette réunion, les étudiants ont montré les maquettes du site à M. COLETTA.

Quelques questions ont été posées, par exemple, arrive-t-on directement sur la page d'accueil du festival ou doit-on choisir parmi les différents festivals qui ont déjà été organisés ?

On part sur le principe que l'on doit choisir d'abord un festival sur la page d'accueil puis s'y identifier. (Cf. Annexe 1 : Maquettes du site)

Il y a eu des remarques vis-à-vis des maquettes, comme le fait que si un planning n'est pas fini d'être rempli, le bénévole ne doit pas le voir, il faudra donc **bloquer cette partie du site et ne l'activer que lorsque l'organisateur l'aura décidé**. On pourrait alors implémenter un bouton "enregistrer" ou encore "publier" pour valider cette étape.

M. COLETTA a aussi remarqué que les pages étaient plutôt complexes et donc il fallait les simplifier, comme n'avoir qu'une page pour se connecter ou qu'une page pour effectuer l'affectation.

La base de données n'étant pas totalement pas finie, nous avons préféré demander plus de renseignement sur celle-ci au professeur.

M. COLETTA a pointé du doigt l'existence de calendriers exportables sur Internet et que l'on pourrait ajouter cette fonctionnalité à notre site. (iCal)

Il y a d'ailleurs une implémentation envisageable sur la liste, ou le nombre de postes ayant besoin de tant de bénévoles quand l'organisateur les affecte au fur et à mesure (comme lors du choix des différents projets en début du semestre sur l'interface dédiée).

Pour la prochaine fois, M. COLETTA demande d'envoyer la base de données sous la forme d'un diagramme UML (implémentée depuis *PhpMyAdmin*). De plus, une recherche est attendue sur les composants d'un calendrier à exporter, ainsi que les IDE supportant le langage PHP et Git.

Enfin, il faudra mettre une partie "Documentation" sur le site où tous les documents montrés au professeur seront implémentés dans le dossier.

Annexe 7 : Réunion 3 : Présentation de la base de données et du modèle MCV

Date : 25 octobre 2018

Présents : Julian ALARCON, Tanguy CADIEUX, Rémi COLETTA, Célia GROSCH, Samira YOUNESY

La réunion s'est déroulée en trois parties.

Tout d'abord, nous avons parlé de la base de données, puis du MCV et du code PHP et enfin des problèmes et des conseils concernant le projet.

Concernant la base de données, nous n'avons toujours pas trouvé comment exporter la base de données en diagramme de classe. M. COLETTA nous alors conseillé d'utiliser IntelliJ, un logiciel puissant pour, entre autres, exporter des schémas de base de données à partir de fichiers.

Pour le MCV, M. COLETTA nous a demandé de faire un MCV classique, comme vu en cours de PHP. Un contrôleur est alors conseillé pour chaque classe, notamment pour la classe "bénévoles" et "organiseurs" car l'héritage en SQL n'est pas possible, ainsi que les model et les vues. Le model pourra reprendre une méthode générique pour chacune des classes.

Plus en détails, plusieurs remarques ont été développées.

- Premièrement, quand une personne se connecte, on observe qu'on a plusieurs moyens de reconnaître si cette personne est un organisateur ou un bénévole: on peut créer deux pages pour différencier les connexions, ou utiliser une seule page de connexion et la base de donnée devra se débrouiller pour reconnaître un bénévole ou un organisateur, via un déclencheur, comme l'a souligné Julian A. ou une vue reflétant l'union des bénévoles et des organisateurs, différencié par un attribut créé dans la vue, comme suit.

```
CREATE VIEW UTILISATEUR_DIFFERENCTION
```

```
SELECT IDBENEVOLE, PWD, "B" FROM BENEVOLE
```

```
UNION
```

```
SELECT IDORGANISATEUR, PWD, "O" FROM ORGANISATEUR ;
```

- De plus, M. COLETTA a soulevé un problème concernant la cohérence entre l'hébergeur du site et la base de données. En effet, le code du site est sur le serveur privé de Tanguy C. alors que la base de données est sur PHPmyAdmin de l'IUT de Julian A., ce qui pourrait poser des problèmes vis-à-vis de la liaison entre les deux. Une solution simple serait de tout mettre sur le même domaine.

- Ensuite, pour être sûr que le site marche, il faudrait effectuer des tests, comme en cours de conception, où l'on codait en java des tests sur des méthodes, mais cette fois-ci en PHP.
- Enfin, concernant le e-calendrier, plusieurs choix s'offre à nous, comme le logiciel ECAL, un ICS PHP (qui importera/exportera un calendrier à mettre sur son agenda personnel) ou chercher nous-même une librairie capable d'effectuer les mêmes tâches. Une seule consigne est donnée : le logiciel doit être en javascript et non un client lourd, qui ralentira notre site.

Pour la prochaine fois, il faudra montrer le diagramme de la base de données définitif, mettre tous les éléments sur un même domaine, remplir la base de données pour ensuite pouvoir faire des tests ainsi que faire une démonstration devant M. COLETTA.

Annexe 8 : Réunion 4 : Présentation du CRUD*

Date : 15 novembre 2018

Présents : Julian ALARCON, Tanguy CADIEUX, Rémi COLETTA, Célia GROSCH, Samira YOUNESY

Lors de cette réunion, nous avons d'abord parlé de la base de données.

Une interrogation s'est posée lors de la conception de la base de données : le bénévole a-t-il un compte par festival (donc il a plusieurs comptes) ou est-ce qu'on peut associer le compte à plusieurs festivals ? Cette dernière solution a été approuvée par M. COLETTA. Par ailleurs, M. COLETTA demande à quoi correspond le booléen "valide" dans la base de données et nous répondons que c'est l'organisateur qui valide un bénévole pour un festival donné.

Une erreur est mentionnée : il manque le champ courriel et n° de téléphone pour la table du bénévole.

Une remarque est alors faite sur le fait qu'on pourrait regrouper la table "Bénévole" et "Organisateur" en une seule table (comme une table "Personne") et chaque groupe est différencié via un booléen "estOrganisateur".

Enfin, un oubli est présent sur le schéma : il manque une table qui relie l'affectation d'un bénévole à un poste pour un créneau ("link_Affecter_Bénévole_Créneau_Poste"). De plus, pour chaque ajout, il faut compter le nombre de postes restés libres, pour pouvoir affecter des bénévoles pour tous les postes sur un créneau,

Pour la table des disponibilités, il faut enlever *dispoD* et *dispoF* car un bénévole est disponible à un moment donné (ce qui est unique). Sinon, il faut retirer *idDispo*, ajouter un *idBénévole* et créer un trigger qui empêche d'avoir deux créneaux qui se chevauchent.

Enfin, pour les disponibilités, il faut trouver le bon type pour avoir la date mais aussi l'heure.

Nous avons ensuite parlé du code en PHP.

Nous avons montré à M. COLETTA le code, qui est composé d'un CRUD* pour chaque objet. Nous avons incrémenté des méthodes génériques pour chaque CRUD*, sauf pour Festival où il y a un CRUD* spécifique.

Sur le code détail des bénévoles, nous remarquons un problème, qui devrait être réglé en travaillant le code. De plus, un organisateur peut voir tous les bénévoles, mais un bénévole ne peut voir tous les organisateurs, ce qui pose un problème selon M. COLETTA.

Si on venait à fusionner les tables "Bénévole" et "Organisateur" en une table "Personne", cette table aurait un rôle dynamique, car une personne pourrait être un bénévole sur un festival mais un organisateur sur un autre festival.

M. COLETTA se propose de faire tous les tests sur ce que peut faire un bénévole.

Par rapport au routeur, M. COLETTA mentionne que la sécurité du site passe par là, et qu'il faut donc sécuriser les vues ainsi que les routes du panneau administrateur.

Nous avons aussi eu d'autres remarques.

Julian A. a demandé à M. COLETTA des informations sur l'auto-incrémentation. En effet, lors de la suppression d'une ligne (par exemple avec l'identifiant 3), on passe directement à l'identifiant suivant (dans l'exemple, on passe à l'identifiant 4) lors d'une insertion. M. COLETTA répond alors que c'est tout à fait normal car il y a un compteur "caché" dans la base de données et que l'incrémentatation se fait à partir du compteur.

Pour manipuler les festivals dans les autres objets, il faut prévoir une clause "... where *idFestival* = *idFestival*". Ainsi, on pourra gérer les bénévoles par festival,

Pour la prochaine fois, il faudra retravailler la base de données, faire en sorte que les routes prennent l'identifiant d'un festival (via des jointures ou des vues) et développer les contrôleurs.

Annexe 9 : Exemples de méthodes génériques

```
public static function select($primary){
    $table_name = static::$object;
    $class_name = 'Model';
    $class_name .= ucfirst($table_name);
    $primary_key = static::$primary;

    $sql = "SELECT * from $table_name WHERE $primary_key = :nom_tag";
    // Préparation de la requête
    $req_prep = Model::$pdo->prepare($sql);

    $values = array(
        "nom_tag" => $primary,
    );
    // On donne les valeurs et on exécute la requête
    $req_prep->execute($values);

    // On récupère les résultats comme précédemment
    $req_prep->setFetchMode(PDO::FETCH_CLASS, $class_name);
    $tab_res = $req_prep->fetchAll();
    // Attention, si il n'y a pas de résultats, on renvoie false
    if (empty($tab_res))
        return false;
    return $tab_res[0];
}

public static function selectAll(){
    $table_name = static::$object;
    $class_name = 'Model';
    $class_name .= ucfirst($table_name);
    $rep = Model::$pdo->query("SELECT * FROM $table_name");
    $rep->setFetchMode(PDO::FETCH_CLASS, $class_name);

    return $rep->fetchAll();
}

public static function save($data){
    $table_name = static::$object;
    $class_name = 'Model';
    $class_name .= ucfirst($table_name);
    $attribut = "";
    $valeurs = "";
    $values = array();
    foreach ($data as $key => $value){
        $attribut .= $key . ", ";
        $valeurs .= "'$value' . ", ";
        $values_etapes = array($key => $value);
        $values = array_merge($values, $values_etapes);
    }
    $attribut = rtrim($attribut, " \t,");
    $valeurs = rtrim($valeurs, " \t,");

    $sql = "INSERT INTO $table_name($attribut) VALUES ($valeurs)";
    echo "<pre>";
    print_r($sql);
    echo "</pre>";
    $req_prep = Model::$pdo->prepare($sql);
    $req_prep->execute($values);
}
```

Figure 17. Capture d'écran du model générique

Annexe 10 : Vue du portail d'un festival

```
<?php

echo '<p>Portail du festival ' . $f->__get('nomFestival') . '</p>';
echo '<p> Lieu: ' . $f->__get('lieuFestival') . '</p>';
echo '<p>Date de début: ' . $f->__get('dateDebutF') . '</p>';
echo '<p>Date de fin: ' . $f->__get('dateFinF') . '</p>';

if(isset($_SESSION['login'])){
    //on vérifie que l'utilisateur est connecté
    $id = ModelBenevole::getIDbyLogin($_SESSION['login']); //l'id sert pour les paramètres des fonctions suivantes

    if(ModelBenevole::isParticipant($id, $_GET['IDFestival'])){
        //on regarde si l'utilisateur est enregistré pour ce festival
        if(ModelBenevole::isValid($_SESSION['login'], $_GET['IDFestival'])){
            //statut benevole
            if(ModelBenevole::isOrga($_SESSION['login'], $_GET['IDFestival'])){
                echo '<p>vous êtes organisateur</p>';
                //fonctionnalités en plus pour l'organisateur
                echo '<p><a href="index.php?controller=benevole&action=readDemandes&IDFestival=' . $f->__get('nomFestival') . '">Liste des bénévoles à accepter</a></p>';
                echo '<p><a href="index.php?controller=benevole&action=readDemandesOrga&IDFestival=' . $f->__get('nomFestival') . '">Liste des organisateurs à accepter</a></p>';
                echo '<p> <a href="index.php?action=deleted&IDFestival=' . $f->__get('IDFestival') . '"> supprimer le festival </a></p>';
                echo '<p> <a href="index.php?action=update&IDFestival=' . $f->__get('IDFestival') . '"> modifier le festival </a></p></div>';
            }else{
                //fonctionnalités uniquement pour les bénévoles
                echo '<p>vous êtes bénévole</p>';
                echo '<p> <a href="http://webinfo/~alarconj/ProjetS3/index.php?controller=benevole&action=postulateOrga&id=' . $f->__get('IDFestival') . '"> Postuler en tant qu'organisateur </a></p></div>';
            }
            //fonctionnalités communes à tout le monde validé
            echo '<p><a href="index.php?controller=benevole&action=readOrga&IDFestival=' . $f->__get('IDFestival') . '"> Liste des organisateurs</a></p>';
            echo '<p><a href="index.php?controller=poste&action=readPostesFestival&IDFestival=' . $f->__get('IDFestival') . '"> Liste des postes</a></p>';
            echo '<p><a href="index.php?controller=benevole&action=readBene&IDFestival=' . $f->__get('IDFestival') . '"> Liste des benevoles</a></p>';
        }else{
            //l'utilisateur est enregistré mais pas validé par un organisateur
            echo '<p>vous êtes en attente de validation pour ce festival</p>';
        }
    }else{
        //on ne dispose d'aucun statut pour le festival
        echo '<p> Vous pouvez postuler pour ce festival en cliquant ici: <a href="index.php?action=participe&IDFestival=' . $f->__get('IDFestival') . '"> Participer </a></p>';
    }
}
?>
```

Figure 18. Capture d'écran de la vue du portail de fonctionnalités d'un festival

Résumé :

Ce projet retrace la mise en place d'un site web de la gestion de bénévoles d'un festival. Les organisateurs peuvent alors ajouter et affecter des bénévoles aux différentes tâches en fonction des besoins et des créneaux renseignés au préalable. Ce site web répondra alors à ce besoin croissant de gestion de bénévoles de festivals, pour les deux catégories d'utilisateurs, les bénévoles et les organisateurs, afin d'en assurer le relais facilement et rapidement.

L'application web est codée en PHP selon une architecture MVC. Elle implémente une base de données MySQL via phpMyAdmin sur un serveur Apache.

Mots Clés : festival - bénévole - organisateur - PHP - organisation - site web

Summary :

This project is about setting up a website of a festival's volunteers management. The organisers can add and assign volunteers different tasks according to the needs and windows that have been previously inquired. This website will answer this increasing need of volunteers' management of festivals, for both categories of actors: volunteers and organisers in order to make relay easier and faster.

The web app is coded in PHP according to an MVC structure. It implements a MySQL database via phpMyAdmin on an Apache server.

Key Words: festival - volunteer - organiser - PHP - organisation - website