

Projet 3 : Mission 3

Site web - le formulaire et son traitement



1. Mise en situation

M. Adit, président du Club ACF2L, vous a demandé de créer un site internet de présentation des activités de l'association. M. Jacques Adit, content de votre prestation, vous demande de réfléchir à la mise en place d'une part d'un formulaire d'inscription de ses adhérents et d'autre part un formulaire de réservation pour pouvoir organiser les plannings.

2. Travail à faire en équipe

Le traitement d'un formulaire nécessite l'utilisation du langage PHP et donc d'un serveur web.

Sites pouvant vous donner des informations :

<https://openclassrooms.com/fr/courses/1603881-creez-votre-site-web-avec-html5-et-css3>

<https://bootstrap-top-design.com/bootstrap-datepicker-tutoriel/>

2.1 Etape 1 → installation du serveur web => mission 4

Avant la mise en production du site (faite dans la mission 4), vous pouvez tester votre travail en utilisant Laragon. Pour cela vous devez :


- Copier votre site dans le répertoire www de Laragon.
- Démarrer les services de Laragon pour que le PHP soit interprété.
- Un gestionnaire de base de données est installé par défaut => HeidiSQL mais vous pouvez installer phpMyAdmin si vous le souhaitez.

2.2 Etape 2 → Récupération des données des adhérents via un formulaire.

Actuellement, les adhérents remplissent une fiche d'information qui est ensuite saisie par la secrétaire dans un tableur. En ce qui concerne les activités de formation de pilote et de baptême de l'air et de réparation, les informations sont inscrites sur un grand planning accroché au mur.

Les membres du bureau de l'association ont validé la proposition d'améliorer cette gestion via des formulaires mais se posent des questions sur la protection des données à caractère personnel.

- 2.1 : enregistrement des données dans un endroit spécifique ?
- 2.2 : droit des utilisateurs ? obligation incombant au responsable du traitement
- 2.3 : proposer une durée de conservation des données
- 2.4 : indiquer dans quel document exigé par le RGPD cette durée doit être consignée

| | | | | |
|--|----------------------------------|---|------|-----|
| Auteur(s)  | Laurence Hourdeaux, Alain Dehors | AP - formulaire et base de données | | |
| 4 | Date 30/10/22 | | Page | 1/6 |

2.5 : préciser quelles données mises dans la fiche d'information ne seront pas retenues pour la création du formulaire car non pertinentes ou qui porteraient atteinte à la vie privée au regard du RGPD.

2.6: Sur une page HTML reliée à votre site (prévoir éventuellement la page dans le menu), créer le formulaire reprenant la fiche d'information. (la présentation est à votre choix, en CSS bien sur).

Fiche d'informations :

1. civilité (sous forme de bouton radio)
2. nom
3. prénom
4. date de naissance (différentes méthodes à votre choix)
5. Adresse : à décomposer en Numéro et rue, code Postal (sous forme de liste déroulante) et ville.
6. Votre situation familiale (célibataire, marié, Union Libre, PACS, divorcé) avec la date de début ou la date de fin
7. email
8. Téléphone portable
9. Nombre de personnes à charge : enfants mineurs (nombre), enfants majeurs (nombre), Parents à charge et la cause du handicap.
10. Les activités susceptibles d'être demandées (choix multiple). Ainsi, une personne peut vouloir prendre à la fois des cours de pilotage et faire un baptême de l'air par exemple.

2.3 Etape 3 → test du formulaire

Créez une nouvelle page sur le site pour l'affichage des champs saisis.

Modifiez la balise `<form>` = `<form action=traitement.php methode="post">`

Ces éléments permettent de définir le fichier de destination des données et la méthode utilisée pour transmettre les données saisies dans le formulaire.


- Récupération des données et transfert dans des variables
- Affichage des valeurs de ces valeurs (echo ou print)

2.4 Etape 4 → finalisation du formulaire

Nous allons ajouter des contrôles sur la saisie du formulaire. Ces contrôles peuvent être faits du côté client et/ou du côté serveur. (Pensez à regarder du côté de html5 ou javascript)

1. Les champs « nom, prénom, email » doivent être obligatoirement remplis. (mettre un astérisque et une indication sur la page formulaire) et tester.
2. Vérifications
 1. le champ numéro du portable
 2. vérifier la validité de l'adresse mail

1. Dans la majorité des cas, les données sont ensuite inscrites dans une base de données. Mais avant cela, elles devront passer dans des filtres, afin de limiter au maximum les soumissions indécrites

| | | | | | |
|---|------|----------------------------------|---|------|-----|
| Auteur(s) | | Laurence Hourdeaux, Alain Dehors | AP - formulaire et base de données | | |
|  | | | | | |
| 4 | Date | 30/10/22 | | Page | 2/6 |

(il est temps de vous sensibiliser aux concepts de « injection HTML/JavaScript » et « injection SQL »)

1. Définir ce que sont ces injections
2. Que faire pour les éviter ?

2.5 Etape 5 → création de la base de données

Les informations récupérées à partir du formulaire vont être stockées dans une base de données. On prend comme hypothèse que la base de données est de type Relationnel, pilotable par SQL.

(on utilisera MySQL comme système de gestion de bases de données relationnelles - SGBDR) :

1. Créer votre base de données : son nom doit correspondre aux initiales de vos noms suivi de « - ACF2L »
2. Créer la table adhérent : son nom doit correspondre aux initiales de vos noms suivi de « - adherent »
3. Testez / Mettez au point un ordre SQL d'insertion (en ligne de commande ou avec un assistant graphique)

Pour commencer et pour vous simplifier la tâche, vous commencerez par insérer les informations d'identification (nom, prénom, email) de l'adhérent dans la table xxx-ADHERENT

4. Concevoir un script cible du formulaire, filtrer les données reçues et, le cas échéant, faites exécuter l'ordre d'insertion (testé à l'étape précédente) dans la base des données (INSERT...), sinon retourner le formulaire en signalant les erreurs (si ce n'est pas encore fait, le formulaire de contact devra donc passer en php)
5. En réponse, vous répondrez par un ordre de redirection (côté client) vers une page de remerciement. En faisant apparaître les données saisies et le numéro de l'adhérent. (remplacer la page correspondant à l'étape 3).

suite sous réserve de modification

2.6 Etape 6 → La gestion des réservations des baptêmes de l'air


Pour une meilleure communication et faciliter l'activité « baptême de l'air », le bureau de l'association aimerait avoir la possibilité de gérer les réservations sur informatique.

Actuellement la secrétaire utilise un tableur pour gérer les réservations. Elle voudrait

Il faut être adhérent pour des questions d'assurance.

Un formulaire est fait par le client qui indique sur quel type d'appareil il veut faire son vol et les dates possibles. la secrétaire va rechercher son Numéro d'adhérent et va à partir d'un formulaire en partie back-office (page cachée dans le site) saisir la réservation avec quel pilote et quel vol et quelle date.

1. Établir un diagramme des différentes entités (vous pouvez utiliser DIA ou WORKBENCH ou phpmyadmin). Il va être utile de créer les tables pilotes, avions, réservations en plus de la table adhérent.
2. Concevez puis exécutez un script SQL de création de ces tables (ligne de commande ou phpMyAdmin)
3. Testez / Mettez au point un ordre SQL d'insertion (en ligne de commande ou avec un assistant graphique)

| | | | | | |
|---|------|----------------------------------|--|---|-----|
| Auteur(s) | | Laurence Hourdeaux, Alain Dehors | | AP - formulaire et base de données | |
|  | | | | | |
| 4 | Date | 30/10/22 | | Page | 3/6 |

4. En réponse, vous répondrez par un ordre de redirection (côté client) vers une page de d'information (page de test) puis à la place par un mail.
5. Pour les plus rapides vous pourrez réfléchir, sur l'insertion de l'ensemble des données dans les différentes tables sans que la secrétaire entre le numéro d'adhérent.

2.7 Etape 7 → exploitation de la base de données

Vous devez maintenant réfléchir aux utilisations qui pourront être faites de la base de données.

Dans un premier temps, vous devrez construire un tableau qui affichera les données essentielles qui ont été insérées dans la base. (Il faudra utiliser php et html pour obtenir un tableau de synthèse.)

Vous devrez également proposer au minimum 3 requêtes permettant de faire des tris ou de mettre en place des filtres permettant une meilleure exploitation de cette base, en listant et en comptant les adhérents qui se sont pré-inscrits / discipline ou activité.


Bon courage et bonne programmation !

3. Annexes

3.1 Annexe 1 : Traitement d'un formulaire

Prenons l'exemple suivant :

(source <http://www.alsacreations.com/tuto/lire/1488-jquery-mobile-par-l'exemple-formulaire-de-contact.html>)

| | | | | | |
|--|----------------------------------|---|--|------|-----|
| Auteur(s)  | Laurence Hourdeaux, Alain Dehors | AP – formulaire et base de données | | | |
| 4 | Date | 30/10/22 | | Page | 4/6 |

Home

Kiwi Party, by Alsacr ations

Nous donner un feedback

Kiwi Party

Votre nom :

Votre email :

Si vous deviez noter la Kiwi Party 2012 : (10  tant le nirvana)

Vous aviez d  j particip     une Kiwi Party ? ☐ Non

Reviendrez-vous ? ☐ Non

Langages de programmation que vous utilisez :

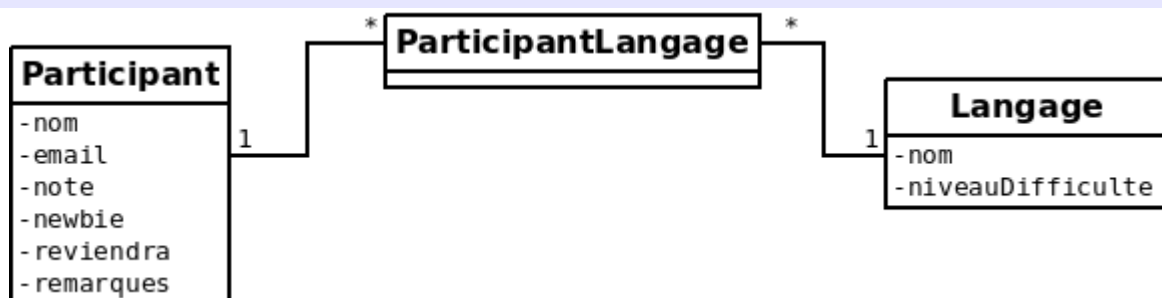
D'autres remarques pour les organisateurs ?

Cet extrait de formulaire recueille des informations qui seront  tre transmises aux organisateurs (wiki party). Le formulaire contiendra donc un bouton submit qui aura pour effet de transmettre les donn  es   un script (attribut action de la balise form).

Ensuite les donn  es seront sauvegard  es dans une base de donn  es. On prend comme hypoth  se que la base de donn  es est de type Relationnel, pilotable par SQL.

3.2 ANNEXE 2: ANALYSE DU FORMULAIRE POUR CREER LES TABLES

Diagramme UML des entit  s



Impl  mentation dans une base de donn  es MySQL. On fait correspondre une entit     une table : On obtient 3 tables, dont une table de liaison que voici :

CREATE TABLE ParticipantLangage (

| | | | | | |
|-----------|------|----------------------------------|-------------------------------------|------|-----|
| Auteur(s) | | Laurence Hourdeaux, Alain Dehors | AP – formulaire et base de donn  es | | |
| | | | | | |
| 4 | Date | 30/10/22 | | Page | 5/6 |

```

id int(11) NOT NULL AUTO_INCREMENT,
idParticipant int(11) NOT NULL,
idLangage int(11) NOT NULL,
PRIMARY KEY (id),
UNIQUE KEY unique_by_participant_lang (idParticipant,idLangage),
KEY IDX_29CECF7C43CAA294 (idParticipant),
KEY IDX_29CECF7C87785BEE (idLangage),
CONSTRAINT fk_participant FOREIGN KEY (idParticipant)
REFERENCES Participant(id),
CONSTRAINT fk_langage FOREIGN KEY (idLangage)
REFERENCES Langage (id)
) ENGINE=InnoDB;

```


Une autre solution possible (observez l'expression de la PK - clé primaire)

```

CREATE TABLE ParticipantLangage (
idParticipant int(11) NOT NULL,
idLangage int(11) NOT NULL,
PRIMARY KEY (idParticipant, idLangage),
KEY IDX_29CECF7C43CAA294 (idParticipant),
KEY IDX_29CECF7C87785BEE (idLangage),
CONSTRAINT fk_participant FOREIGN KEY (idParticipant)
REFERENCES Participant(id),
CONSTRAINT fk_langage FOREIGN KEY (idLangage)
REFERENCES Langage (id)
) ENGINE=InnoDB;

```

Ces deux types de solution ont leurs avantages et leurs faiblesses... Vous êtes libre de choisir celle qui vous convient le mieux pour commencer.

| | | | | | |
|--|----------------------------------|---|--|------|-----|
| Auteur(s)  | Laurence Hourdeaux, Alain Dehors | AP – formulaire et base de données | | | |
| 4 | Date | 30/10/22 | | Page | 6/6 |