

Compte rendu MediaTek86

Compte rendu pour le devoir 2 de l'atelier de professionnalisation du BTS SIO au CNED.

Sommaire

Sommaire	1
Contexte	2
Mission	2
Réalisation.....	2
Étape 1.....	2
Objectifs	2
Démarche.....	2
Résultat	3
Étape 2.....	3
Objectifs	3
Démarche.....	3
Résultat	3
Étape 3.....	3
Objectifs	3
Démarche.....	3
Résultat	4
Étape 4.....	4
Objectifs	4
Démarche.....	4
Résultat	4
Étape 5.....	4
Objectifs	4
Démarche.....	4
Résultat	4
Étape 6.....	5
Objectifs	5
Démarche.....	5
Résultat	5
Bilan.....	5
Annexe	6
Annexe 2.1 : Schéma visuel de l'application	6

Contexte

La médiathèque MediaTek86 à besoin d'un outil facilitant la gestion de son personnel qu'utilisera le responsable.

Mission

Un logiciel devra être créé pour donner une interface graphique simple et facile d'utilisation permettant la gestion du personnel de la médiathèque et de leurs absences. Le logiciel sera écrit en C# et communiquera avec une base de données MySQL qui contiendra les diverses informations utiles.

Le logiciel devra permettre les cas d'utilisation suivants :

- Le responsable devra se connecter
- On peut ajouter un personnel
- On peut modifier un personnel
- On peut supprimer un personnel
- On peut afficher les absences d'un personnel
- On peut ajouter une absence
- On peut modifier une absence
- On peut supprimer une absence

Tous les cas devront être accessibles seulement après la connexion du responsable.

Réalisation

Étape 1

Objectifs

- Préparer l'environnement de travail
- Préparer la base de données

Démarche

- Wampserver est installé pour héberger la bdd en local, on utilisera phpmyadmin pour la configuration initiale de la bdd (création de la bdd, création de l'utilisateur du responsable)
- En utilisant phpmyadmin, la bdd et l'utilisateur du responsable est créée, le responsable possède les droits sur la bdd
- La bdd s'appelle "mediatek86" et l'utilisateur du responsable est "MediaTek86admin" avec le même mot de passe
- Le logiciel WinDesign est installé pour pouvoir ouvrir le modèle conceptuel de données qui nous est fourni

- Le script SQL de la structure de la bdd est généré par WinDesign et ensuite importé via phpmyadmin pour créer les tables dans la bdd
- Des données sont générés par le site <https://www.generatedata.com/> pour alimenter la base de données et ainsi permettre de tester l'application future

Résultat

- La bdd est créée, fonctionnelle, structurée et contient des données

Étape 2

Objectifs

- Conceptualiser et réaliser la partie visuelle de l'application
- Créer le dépôt distant de l'application sur GitHub
- Structurer le projet de l'application en MVC

Démarche

- Le logiciel "Pencil" est utilisé pour réaliser le schéma visuel de l'application (voir [Annexe 2.1](#))
- Le projet de l'application est créé sur Visual Studio
- Les dossiers "model", "view" et "control" sont ajoutés dans le projet pour la structure MVC
- Le dépôt distant est initialisé sur GitHub et synchronisé avec le projet sous Visual Studio
- L'interface graphique de l'application est réalisé
- Le dépôt distant est synchronisé

Résultat

- La base de l'application et son interface est codé
- Le dépôt distant est initialisé et synchrone

Étape 3

Objectifs

- Coder la partie modèle de l'application
- Préparer l'accès aux données de la bdd

Démarche

- La classe BddManager est importé dans le projet pour gérer l'envoi de requêtes et la réception de données de la bdd (cette classe provient d'un projet antérieur fait pour le CNED)
- Un dossier "dal" (pour data access layer) est ajouté au projet, il contiendra les classes de liaison de la bdd au contrôleur

- Les classes correspondantes aux tables de la bdd sont codés dans le paquet "model"
- Le travail est sauvegardé sur le dépôt distant

Résultat

- Les classes qui accueilleront les données des tables de la bdd sont prêtes
- La liaison avec la bdd est prête

Étape 4

Objectifs

- Coder les fonctionnalités de l'application
- Tester l'application

Démarche

- Chaque fonctionnalité demandé par les cas d'utilisations sont codés dans le respect de la structure MVC
- Des tests sont fait à chaque fonctionnalité terminé
- Le travail est sauvegardé sur le dépôt distant à chaque fonctionnalité terminé

Résultat

- L'application contient toutes les fonctionnalités demandés

Étape 5

Objectifs

- Générer la documentation technique
- Générer la documentation utilisateur

Démarche

- Des commentaires normalisés sont ajoutés dans le but de générer la documentation technique
- L'outil Sandcastle est utilisé pour générer la documentation technique au format d'un site web
- Le logiciel OBS est utilisé pour enregistrer la présentation du logiciel
- La vidéo de la présentation est ensuite montée avec Adobe After Effects

Résultat

- La documentation technique et utilisateur est faite

Étape 6

Objectifs

- Créer l'installateur de l'application
- Publier l'application et sa documentation sur mon portfolio

Démarche

- Un icône simple est ajouté à l'application
- Un projet pour l'installateur de l'application est ajouté à la solution Visual Studio
- L'installateur est généré
- Une release est publiée sur le dépôt GitHub avec l'installateur et le script SQL pour générer la bdd correspondante
- Le portfolio est créé sur GitHub Pages en utilisant Jekyll et le theme WhatATheme
- La page du projet MediaTek86 est ajoutée et y contient la vidéo de démonstration et les liens vers le dépôt de l'application, la documentation technique et ce compte rendu
- Ce compte rendu est rédigé puis téléchargé vers le portfolio

Résultat

- L'application est terminée et publiée
- L'installateur est opérationnel et publié
- La documentation est publiée
- Le portfolio est fait et publié
- La page de ce projet sur le portfolio est faite
- Ce projet est terminé

Bilan

Le logiciel et la base de données sont créés, la communication logiciel/bdd est opérationnelle et tous les cas d'utilisation sont réalisés.

Le logiciel peut être simplement installé via l'installateur disponible sur le dépôt, et la base de données peut être importée grâce au script SQL fourni avec l'installateur sur le même dépôt.

Annexe

Annexe 2.1 : Schéma visuel de l'application

The image shows a screenshot of a software application window titled "Title". The window contains a form with several input fields and buttons. On the left side of the form, there is a large, empty rectangular area with a vertical scrollbar. On the right side, there is another large, empty rectangular area with a vertical scrollbar. The form fields are arranged in the center:

- Nom
- Prenom
- Tel
- Mail
- Service (dropdown menu)
- Annuler (button)
- Enregistrer (button)
- Date debut
- Date fin
- Motif (dropdown menu)

At the bottom of the window, there are two sets of buttons. The left set includes "Ajouter" and "Supprimer" buttons. The right set includes "Ajouter" and "Supprimer" buttons. In the center of the bottom row, there are "Annuler" and "Enregistrer" buttons.