Mise en place d'une base de données avec Oracle

1. Présentation générale du problème

1.1 Projet

Créer une application permettant de formuler des vœux pour la suite des études des étudiants.

Cette application utilisera une base de donnée (pas encore définie) dans laquelle les vœux seront stockés en fonction des étudiants. L'application sera sous Java et la base de donnée sous Oracle, car Java peut interagir avec Oracle. C'est un projet universitaire, il n'y aura donc pas d'investissements monétaires.

1.2 Contexte

Un DUT est un diplôme qui s'obtient en 2 ans, il prépare à pouvoir travailler mais aussi à la poursuite d'études. À la fin de ces 2 années, des choix doivent être fait. C'est-à-dire que l'étudiant doit choisir s'il continue ses études ou s'il décide de les arrêter.

À l'IUT de Nantes, paradoxalement, la gestion de ces vœux est manuscrite. Et notre secrétaire se retrouve donc avec plusieurs dizaines de dossiers à gérer et à envoyer aux écoles en un temps limité à cause de la date de rendu des étudiants et les dates maximales d'envoi des dossiers dans certaines écoles. C'est pour ça qu'il nous est demandé, dans le cadre de ce projet, de proposer une application en Java, communiquant avec une base de donnée Oracle, qui permettrait de gérer ces vœux tout en facilitant la gestion des vœux par la secrétaire.

Ainsi étudiants, professeurs et secrétaire pourront avoir accès à un certain nombre d'informations qui permettront en fonction du statut de faire certaines choses. Cependant, tout le monde ne doit pas avoir tous les droits : chaque étudiant ne pourra voir, modifier et supprimer que ses informations (vœux, noms, prénoms, etc…), les professeurs ne pourront qu'observer les informations des élèves et y apporter des modifications et la secrétaire observer, imprimer et/ou envoyer par mail les informations des élèves.

1.2.1. Situation du projet par rapport aux autres projets de l'entreprise

C’est un projet qui a un but plus que ludique, il sera utile aussi. Il pourra être réutilisé pour l’émission de nos voeux futurs. De plus, en deuxième année, un projet similaire est en cours de développement pour le département GEII, cependant il est développé en PHP avec l’aide du célèbre framework Symphony. Leur projet est donc plus orienté web.

1.2.2. Études déjà réalisées

M. Nachouki nous a présenté une première étude lors de l’attribution du projet. Une rencontre avec un étudiant de 2ème année travaillant sur un projet similaire nous a permis d’obtenir de nombreuses informations et réponses.

1.2.3. Études menées sur des sujets voisins

Nous avons eu des cours lors du premier semestre sur le langage SQL (des TP, des TD et des CM) et nous continuons aujourd'hui. De plus, nous avons des cours de programmation objet avec le langage Java.

1.2.4. Suites prévues

L’application sera utilisée par l’Université pour gérer la poursuite d’étude. Des modifications peuvent également être faites par la suite.

1.2.5. Nature des prestations demandées

Élaboration d’une application logicielle d'accès et de modification de base de données.

1.2.6. Parties concernées par le déroulement du projet des résultats (demandeurs, utilisateurs)

Les demandeurs sont les enseignants et chargés de voeux des deuxième année. Les utilisateurs sont les élèves de S4 ainsi que certains professeurs ou personnels administratifs.

1.2.7. Caractère confidentiel s'il y a lieu

Confidentialité importante des identifiants des étudiants ainsi que de leurs voeux de poursuite d’étude.

1.3 Énoncé du besoin (finalité du produit pour le futur utilisateur tel que prévu par le demandeur)

Le programme sera une interface en Java, sur laquelle l'étudiant se connectera avec ses identifiants de connexion de l’IUT, et pourra ensuite, pour sa poursuite d’étude, créer ses voeux dans l’ordre de son choix, les modifier ou bien les supprimer. Il aura accès à une base de données des différentes écoles, qu’il pourra demander dans ses différents choix de voeux et aussi demander une alternance ou non pour une école qui la proposerait. Cette interface sera également accessible à un administrateur (optionnel : Professeurs) qui pourra voir les voeux de tous les étudiants, les modifier, les supprimer et naturellement les imprimer. Cet administrateur aura également accès à la base de données des écoles et des élèves de l’IUT et pourra la ou les modifier en conséquence. Il pourra aussi contacter les étudiants par mail afin de voir directement avec eux si un problème se pose ou si l'échéance pour une école approche.

1.4 Environnement du produit recherché

Nous sommes actuellement 6 à travailler sur ce projet, tous en 1re année du DUT Informatique dont 2 redoublants. Nous aurons donc un accès sur les ordinateurs de l'IUT avec Java ainsi que divers outils de développement installés dessus, ainsi qu’un accès sur un serveur de base de donnée (SGBD Oracle). Les données de l’utilisateur seront donc stockés sur ce serveur et l’identification pourra être géré par le système d’authentification “CAS” de l'université. Nous n'aurons que quelques mois pour faire ce projet, aucune rémunération n’est prévue et nous aurons des délais à respecter pour certaines tâches. À prendre en compte aussi la contrainte intellectuelle, nous devrons faire une interface graphique, une recherche assez importante est donc à prévoir sur ce sujet. (Aucun cours à l’heure actuelle)

1.4.1. Liste exhaustive des éléments (personnes, équipement, matières…) et contraintes (environnement)

6 personnes, tous en première année de DUT informatique, travaillent sur ce projet : Erwan Azzoug, Arnaud Chesnay, Ludovic Chevraux, Florian Guisnel, Corentin Jézéquiel et David Morvan.

Nous avons à notre disposition des ordinateurs afin de mener à bien notre projet. Des logiciels sont déjà installé pour pouvoir créer la base de données et faire des requêtes dessus.

1.4.2. Caractéristiques pour chaque élément de l'environnement

Cette interface sera disponible sur le serveur de l’IUT, avec la base de données des étudiants inscrits en deuxième année et la base de donnée des écoles. L’interface sera disponible sur tous les ordinateurs de l’IUT afin de faciliter son utilisation par les étudiants de deuxième année et par l’administrateur.

2. Expression fonctionnelle du besoin

2.1 Fonctions de service et de contrainte

2.1.1 Fonctions de service principales (qui sont la raison d’être du produit)

|  |  |
| --- | --- |
| FP1 | Le programme doit permettre de formuler des voeux |
| FP2 | Le programme doit permettre de voir les voeux |

2.1.2 Fonctions de service complémentaires (qui améliorent, facilitent ou complètent le service rendu)

|  |  |
| --- | --- |
| FSC1 | Le programme permet à l'administrateur d'imprimer la liste des voeux |
| FSC2 | Le programme permet de voir la date limite et d'être alerté si l'on y approche |
| FSC3 | Le programme peut se déconnecter et se quitter |
| FSC4 | Le programme permet à l'administrateur d'envoyer des messages aux étudiants |
| FSC5 | Le programme sera doté d'un bouton "enregistrer", avec une alerte si on ne l'a pas fait |

2.1.3 Contraintes (limitations à la liberté du concepteur-réalisateur)

|  |  |
| --- | --- |
| FC1 | Le programme doit être sécurisé |
| FC2 | Le programme doit être fiable |
| FC3 | Le programme doit être rapide |
| FC4 | Le programme doit être cohérent avec les possibilités de poursuite d’étude |
| FC5 | Le programme doit être multi-plateforme |

2.2 Critères d’appréciation (en soulignant ceux qui sont déterminants pour l’évaluation des réponses)

**Absence d’erreur** dans la retranscription des voeux dans la BDD

Temps de connexion acceptable ( - de 6 secondes )

2.3 Niveaux des critères d’appréciation et ce qui les caractérise

2.3.1 Niveaux dont l’obtention est imposée

* Le programme permet à l'administrateur d'imprimer la liste des voeux
* Le programme peut se déconnecter et se quitter
* Le programme sera doté d'un bouton "enregistrer", avec une alerte si on ne l'a pas fait

2.3.2 Niveaux souhaités mais révisables

* Le programme permet de voir la date limite et d'être alerté si l'on y approche
* Le programme permet à l'administrateur d'envoyer des messages aux étudiants

3. Cadre de réponse

3.1 Pour chaque fonction

3.1.1 Solution proposée

- Faire un WBS : lister les tâches

- Diagramme de séquence : basé sur le diagramme de classe métier et sur les scénarios des cas d’utilisation.

- Jalon : Date à laquelle la fonction doit être livrée. (18 mai ? → fin de la 1re semaine après les vacances)

3.1.2 Niveau atteint pour chaque critère d'appréciation de cette fonction et modalités de contrôle

3.1.3. Part du prix attribué à chaque fonction

À chaque fonction est attribué la part de 0,00 €. C’est un projet universitaire avec un but plus éducatif que lucratif.

3.2 Pour l'ensemble du produit

Le produit sera gratuit, étant donné qu'il ne sera pas commercialisé. Il sera utilisé uniquement dans le cadre de l'Université de Nantes.

3.2.1 Prix de la réalisation à la base

Ce projet est réalisé dans un but éducatif et non-lucratif.

3.2.2 Options et variantes proposées non-retenues au cahier des charges

- Connexion automatisée à partir des sessions de l'IUT

- Système de cryptage des données afin de mieux sécuriser l'application et les transmissions

3.2.3 Mesures prises pour respecter les contraintes et leurs conséquences économiques

Il n’y aura aucune conséquences économiques car on travaille bénévolement

dans le cadre d’un projet universitaire.

3.2.4 Outils d’installation, de maintenance à prévoir

Il faudra avoir Java installé sur son ordinateur et un ordinateur connecté au réseau de l’université. L’installation se fera par un administrateur réseau sur les ordinateurs ou installation du programme dans un dossier commun à tous les utilisateurs.

3.2.5 Décomposition en modules, sous-ensembles

Le programme sera composé de deux parties :

* Un programme client que l’utilisateur exécutera sur son ordinateur
* Une base de donnée pour stocker les données clients

Le programme client communiquera donc avec cette base de donnée

3.2.6 Prévisions de fiabilité

Le programme dépendra très fortement de la base de donnée mise à disposition par l’IUT, une instabilité de cette base rendrait donc le programme non fiable.

3.2.7 Perspectives d’évolution technologique

Amélioration de la sécurité, proposition de compatibilité smartphone,

amélioration de l’ergonomie du programme.

Questions posée au client :

- Utiliserons-nous la base de données de l’université ? Si oui, quand ?

Non, il n'y en a pas. Création de notre propre base

- Faut-il que l’application soit sécurisée ?

Pas le temps de crypter, etc..

- Un thème en particulier pour l’application (couleur, boutons, etc…) ?

Détail à voir après

- Utilisateurs : Etudiant-Administrateur / Etudiant-Administrateur-Professeur ?

Enseignant = Administrateur

Annexes : Diagramme de Gantt et diagramme de PERT.

RDV :

Pouvoir rechercher les écoles sur l’application

-> Tri par école - Tri par Licence - Par école d’ingénieur - MIAGE - Formation par

alternance/continu

-> Modification des écoles par l’administrateur

Prendre en compte la date limite

Connexion manuelle

A faire :

Diagramme de séquence : Professeur fait,

Diagramme de classe : En cours...

Diagramme de cas d’utilisation : Fait.

WBS : A faire.

Cahier des charges : En cours… (A compléter)

Diagramme de Gantt : En cours…

Ajouter une date limite à la formation et un attribut alternance