

WEB2.0 TP

ADMINISTRATION D'UN RÉSEAU AU SEIN D'UNE ENTREPRISE « NET-WORK »

Table des Matières

1 Description du projet.....	2
2 Équipements.....	2
Types d' équipements.....	2
Propriétés des équipements.....	2
3 Interface administrateur.....	3
4 Stockage et application serveur.....	4
5 Validation.....	4
6 Livraison.....	4

1 Description du projet

Le client final souhaite administrer son parc d'équipements réseau de manière moderne, gérer les divers équipements, leurs adresses et autres propriétés, l'ajout, la modification et la suppression.

L'administration devra se présenter sous la forme d'une application « Web2 » (voir détails) donc sous la forme d'une page Web.

Le développement sera réalisé en trois phases :

- Phase 1 : Proposition des choix technologiques et des estimations de délais
- Phase 2 : Architecture de la base de données et concept de l'interface d'administration
- Phase 3 : Réalisation

Il n'est pas demandé de conseils sur le matériel ni l'infrastructure réseau, l'application serveur sera hébergée en interne chez le client sur son serveur d'application (Windows ou Linux), le site Web sera géré par Apache ou IIS (ou équivalent gratuit) au choix du prestataire.

2 Équipements

Types d'équipements

Le terme « équipement » regroupe (à ce jour) les matériels suivants :

- Ordinateurs fixes type PC/MAC
- Ordinateurs portables
- Imprimantes
- Photocopieuses
- Téléphones
- Routeurs
- Serveurs

- Il n'est pas demandé de gérer d'autres types pour le moment -

Propriétés des équipements

- Identifiant unique (R/W)
- Type (R/W)
- Nom (R/W)
- Fabricant (R/W)
- Adresse physique (R/W)
- Adresse IP (si fixe) (R/W)
- Propriétaire ou utilisateur principal (R/W)
- Localisation (si fixe) (R/W)
- Numéro support (réparation) (R/W)
- État technique (Fonctionnel, En panne mineure, en panne majeure, Inconnu) (R)
 - Note : L'état technique est un état en lecture seule qui est déduit d'une lecture hardware hors du projet.
- État fonctionnel (En marche, éteint, en veille, en arrêt de maintenance) (R pour les 3 premiers, W pour le dernier)
- Message de maintenance (R/W)

3 Interface administrateur

Page web unique sans rechargements il est important de se sentir « comme dans une application de bureau ».

- Il n'est pas demandé d'authentification pour ce projet -

L'application doit fonctionner sur les navigateurs Firefox et Chrome (dernières versions) obligatoirement
Aucun plugin ne doit être nécessaire (Flash ou autre)

L'application doit proposer au minimum :

- La liste des équipements (rafraîchissement automatique) et la consultation de leurs propriétés (Les équipements en panne seront colorés en rouge, les équipements éteints grisés, etc.)
- La modification de leurs propriétés autorisées en écriture
- L'ajout et la suppression d'équipements
- Un journal de bord visible en permanence indiquant les divers événements et actions réalisées depuis l'ensemble des postes clients (apparition des pannes, ajout d'équipements, mise en panne forcée, etc.)

Et optionnellement :

- Une vue d'ensemble du parc sur synoptique
- La possibilité de lier les équipements entre eux de manière hiérarchique (Exemple : si un routeur tombe en panne les équipements « enfants » qui lui sont rattachés seront marqués comme « indirectement inutilisables ») et graphique (depuis le synoptique)

La modification de la propriété « État fonctionnel » doit impliquer une justification de la part de l'utilisateur, celle ci doit être conservée sur le serveur et indiquée dans la liste des propriétés de l'équipement.

Les équipements en panne doivent se distinguer clairement des autres dans la liste des équipements, une alarme sonore (lors de l'apparition d'une panne) serait un plus non négligeable.

Optionnel :

Placement sur synoptique (en mode graphique) des équipements, le fond image doit être paramétrable mais il n'est pas nécessaire de le prévoir au niveau de l'interface, un paramètre serveur suffira.

Aucun framework (ou librairie) n'est imposé ou refusé mais l'application doit être conçue le plus simplement possible afin de faciliter sa maintenance et son évolution, le choix du frameWork doit être justifié dans la phase 1.

- Exemples: HTML5+Javascript « pur », jQuery(ui), Bootstrap, Prototype.js, Angular, etc.

4 Stockage et application serveur

L'application serveur peut être réalisée dans n'importe quel langage , les langages suivants sont recommandés :

Php, Jsp, Servlet Java

L'application serveur doit être simple et doit essentiellement consister à recevoir les demandes des postes clients et y répondre, **il n'est pas demandé de gestion de chiffrement ni même d'authentification ni même de gestion des conflits**

Le serveur aura également en charge le stockage (en base de données) du parc d'équipements.

Le serveur de base devra être de type « MySQL » ou équivalent.

- Il n'est pas demandé de gérer de sauvegardes/récupération de la base -

5 Validation

Afin de procéder à la validation de l'application il est demandé de pouvoir actionner sur le serveur un mode « Simulation » où le parc des équipements sera « en vie », exemples :

- Les équipements peuvent s'allumer ou s'éteindre (sauf les routeurs) → simuler le changement spontané de l'état fonctionnel.
- Les équipements peuvent tomber en panne mineure ou majeure → simuler le changement spontané de l'état technique.

Il n'est pas demandé de simuler le protocole physique de détection des anomalies, la simulation pourra donc inscrire ces informations dans la base de donnée.

6 Livraison

La livraison de l'application doit suivre la procédure suivante :

- Phase 1
 - Document PDF indiquant les choix technologiques et leurs justifications
 - Un résumé (un simple tableau ou liste) des temps estimés pour chaque partie de la solution
- Phase 2
 - Document PDF indiquant la structure de la base de données (tables et champs) et comportant également la proposition d'interface graphique
- Phase 3
 - Document de livraison (liste des livrables et procédure d'installation)
 - Fichier de livraison (zip ou tgz)

– FIN DU DOCUMENT –