

Organisation de l'entreprise TD 2

OCRISPÉE

O = listing des problèmes, en choisir un

C = caractériser

R = rustiner

I = identifier

s = solution élémentaire (= détaillée)

S = solution globale combinaison de solutions élémentaires

P = planifier la solution

é = évaluer et améliorer la solution

E = expérimenter la solution

Indicateurs = est-ce que ça marche ? Est-ce mis en place ? Permet le pilotage

- Montréal, entreprise d'ingénierie
- Mission : pages sur un site intranet (web)
- Langage ASP utilisé sur un serveur pour mettre en ligne des sites
- Documentation bancaire, peu de possibilités
- Travail seul

Étape 0

Problèmes :

- Seul
- Documentation inexistante
- Technologie obsolète : ASP + serveur => nécessité d'innover
- Pas d'instructions précises
- Pas accès au serveur, pas de droits root, limitations aux technologies obsolètes
- Rien de prêt à l'arrivée : 4 jours perdus
- Compréhension avec les québécois
- Blocage à l'accès de certaines ressources

	Gravité	Apparence	Fréquence
Pas d'accès au serveur	5	3	5
Seul	1	1	5
Documentation inexistante	2	2	4
Technologie obsolète	3	1	3
Pas d'instructions précises	4	2	4
Rien de prêt	2	1	1
Compréhension avec les québécois	1	1	1

Problème choisi : pas d'accès au serveur en raison de son caractère répétitif

Caractérisation

Faits observables

1. Documentation sur l'ASP faible sur site, au début. Principes de bases : syntaxe, 50 pages avec un principe par page. Manque de cas complexes, par exemple développement d'un forum = site dynamique, données générées à la volée pour chaque utilisateur.
2. Tentative de modification du code impossible en ligne, trois tentatives se sont soldées par un échec.
3. Perte des droits cinq fois.
4. Panne du serveur deux fois.
5. Panne du réseau une fois.

Symptômes

Vieilles technologies : ASP (+15 ans) nécessairement à utiliser, logiciel serveur obsolète
Manque de renseignements sur ce qui est attendu (« tu fais le plus possible... »)

Conséquences

Demande des spécifications

Travail ralenti

- Perte estimée à 1 mois

Découvertes au fur et à mesure

Le problème sera résolu si le stagiaire obtient les droits pour effectuer les modifications.

Rustiner

Demander l'autorisation, de l'aide. Recherches personnelles sur internet, tests.

Recyclage du site et non refonte => améliorations

Identifier*5 P*

- Demande d'autorisation : pourquoi ?
- Droits d'utilisateur propre à chaque groupe d'employés : pourquoi ?
- Pour éviter un accès malintentionné : pourquoi ?
- Confidentialité : pourquoi ?
- Concurrence ou terrorisme par exemple : pourquoi ?
- Données sensibles : actionnariat d'aéroports français, travail dans des exploitations offshores

5 P bis

- Absence de documentation : pourquoi ?
- Vieille technologie peu documentée : pourquoi ?
- À l'époque, peu de mise en ligne de l'information : pourquoi ?
- Web pas aussi répandu qu'aujourd'hui : pourquoi ?
- + de solutions logicielles que web

5 P ter

- Perte de temps : pourquoi ?
- Besoin de se documenter : pourquoi ?
- Technologie n'est plus enseignée : pourquoi ?
- Technologie obsolète : pourquoi ?
- Remplacée par d'autres langages plus performants : ASP.NET, PHP...

5 M

- Méthode : logiciel IIS5
- Main d'œuvre : moi
- Milieu : secteur informatique pétrole et gaz
- Matériel : serveur obsolète
- Matière : non cohérent ici

Solutions élémentairesSolutions cause 1 : documentation ASP faible

1. Contacter les experts du langage dans l'entreprise pour récupérer de la documentation, éventuellement non mise à disposition
2. Changement du langage vers un autre qui a une meilleure documentation : ASP.NET par exemple
3. Prendre exemple sur un langage proche ayant une documentation correcte : vBscript par exemple
4. Prendre exemple sur les codes sources déjà écrits
5. Écrire une documentation au fur et à mesure de la réalisation avec les découvertes. Utilisation possible de Word pour un partage ultérieur.

Critères des choix	Coût	Acceptabilité	Faisabilité	Efficacité	Total
Solution 1	2	5	5	3	15
Solution 2	0	1	0	3	4
Solution 3	2	4	4	3	13
Solution 4	3	4	5	3	15
Solution 5	2	4	4	5	15

Solutions cause 2 : tentative de modification du code impossible en ligne

1. Développer en local en reproduisant le serveur à l'aide de technologies de virtualisation (IIS local par exemple)
2. Demander les droits d'accès au responsable réseau pour accéder effectivement à la modification
3. Changer la politique des droits d'accès : autoriser l'accès au serveur pour les stagiaires
4. Outrepasser la sécurité imposée par le responsable réseau

Critères des choix	Coût	Acceptabilité	Faisabilité	Efficacité	Total
Solution 1	5	4	5	5	19
Solution 2	2	4	4	5	15
Solution 3	2	2	1	5	10
Solution 4	5	0	2	5	12

Solutions cause 3 : perte des droits

1. Sauvegarde des droits sécurisée : duplication des données la nuit
2. Mettre en place une liste pour savoir qui a accès à quelle ressource précisément afin d'éviter des nettoyages imprévisibles sur les autorisations
3. Désactiver le système de droits d'accès : tout le monde a le droit d'accéder à l'intégralité des ressources
4. Créer un formulaire de demande de droits destiné au gestionnaire réseau

Critères des choix	Coût	Acceptabilité	Faisabilité	Efficacité	Total
Solution 1	0	5	5	5	15
Solution 2	2	4	4	5	15
Solution 3	5	0	5	5	15
Solution 4	2	4	5	5	16

Solutions cause 4 : panne du serveur

1. Mettre en place un serveur de secours avec sauvegarde quotidienne
2. Avoir un serveur local pour continuer à développer sans être affecté par les modifications de la machine en ligne
3. Mettre en place une équipe de technicien propre au datacenter où se trouve le serveur pour une réparation rapide
4. Auditer les pannes du serveur pour éviter qu'elles ne se reproduisent

Critères des choix	Coût	Acceptabilité	Faisabilité	Efficacité	Total
Solution 1	0	5	5	5	15
Solution 2	2	3	3	5	10
Solution 3	0	3	4	2	9
Solution 4	2	5	4	4	15

Solutions cause 5 : panne du réseau

1. Mettre en place un réseau de secours
2. Avoir un réseau local pour continuer à développer sans être affecté par les modifications de la machine en ligne
3. Mettre en place une équipe de technicien propre au réseau pour une réparation rapide
4. Auditer les pannes du réseau pour éviter qu'elles ne se reproduisent

Critères des choix	Coût	Acceptabilité	Faisabilité	Efficacité	Total
Solution 1	0	5	5	5	15
Solution 2	2	3	3	5	10
Solution 3	0	3	4	2	9
Solution 4	2	5	4	4	15

On ne garde que les solutions strictement en dessous de 12 pour éliminer les solutions les moins efficaces.

Regroupement des solutions élémentaires

	S1.1	S1.3	S1.4	S1.5	S2.1	S2.2	S3.1	S3.2	S3.3	S3.4	S4.1	S4.4	S5.1	S5.4
S1.1		+	+	0	0	0	0	0	0	0	0	0	0	0
S1.3			+	+	0	0	0	0	0	0	0	0	0	0
S1.4				+	0	0	0	0	0	0	0	0	0	0
S1.5					0	0	0	0	0	0	0	0	0	0
S2.1						0	0	0	-	0	-	0	-	0
S2.2							+	0	--	+	0	0	0	0
S3.1								+	--	+	+	+	+	+
S3.2									--	0	0	0	0	0
S3.3										--	0	0	0	0
S3.4											0	0	0	0
S4.1												+	+	+
S4.4													+	+
S5.1														+
S5.4														

Planifier la solution

Au vu du tableau ci-dessus et de la méthode CAFE mise en œuvre précédemment, on choisit de :

- Le stagiaire commence par contacter, des collègues pour plus de renseignements sur l'ASP en l'absence de documentation claire : spécificités, conventions de code au sein de l'entreprise
 - Quand : début du stage
 - Comment : par mail, ou physiquement de préférence
 - Durée : une semaine, deux maximums
- Si les renseignements ne s'avèrent pas suffisants ou que l'on a besoin d'éléments plus spécifiques, le stagiaire peut ensuite s'inspirer de codes sources déjà écrits, que ce soit en ASP ou dans des langages proches comme l'ASP.NET
 - Quand : tout au long du stage
 - Comment : par internet
 - Durée : tout au long du stage
- Le stagiaire peut enfin rédiger une documentation propre grâce aux renseignements obtenus plus tôt
 - Quand : deux approches possibles
 - Au fur et à mesure
 - À la fin du stage
 - Comment : document Word
 - Durée : une semaine
 - Indicateur de conformité : s'assurer que la documentation écrite suit les conventions de l'entreprise
 - Indicateur de satisfaction : demander aux développeurs ce qu'ils pensent de la nouvelle documentation, s'ils ont des réclamations

- Le stagiaire demande les droits d'accès au responsable réseau pour être en mesure de modifier le contenu distant
 - Quand : au début du stage
 - Comment : un mail ou, mieux, mettre en place un formulaire dédié
 - Quand : au début du stage
 - Durée : deux jours
 - Indicateurs de fonctionnement : temps d'attente, durée d'exécution, retards éventuels
 - Durée : une demi-journée
 - Indicateurs de fonctionnement : temps d'attente, durée d'exécution, retards éventuels
- Pour éviter que les droits du stagiaire soient révoqués, une liste doit être créée par le responsable réseau, dupliquée sur un serveur de secours chaque nuit, indiquant qui a accès à quoi
 - Quand : le plus tôt possible
 - Comment : liste sous forme de base de données, duplication via tâche (cron) lancée automatiquement une fois par jour, la nuit
 - Durée : une semaine de mise en place
 - Indicateur de satisfaction : demander aux utilisateurs si leurs droits restent bien en place
 - Indicateurs de fonctionnement : temps d'attente, durée d'exécution, retards éventuels
- Sur le même principe que précédemment, le stagiaire peut créer un serveur de pré-production pour avoir une plateforme de développement fonctionnelle sans risque de casser le serveur principal. Ce serveur assurera également la redondance en cas de panne du serveur de production, grâce à une sauvegarde nocturne journalière.
 - Quand : au milieu du stage
 - Comment
 - Nouveau serveur physique mis à disposition du stagiaire
 - Duplication via tâche cron lancée automatiquement une fois par jour, la nuit
 - Durée : deux semaines
 - Indicateurs de fonctionnement : temps de réponse, suivi des pannes

Expérimenter la solution et suivre sa mise en œuvre

Au cours de son travail, le stagiaire a contacté ses collègues afin de se renseigner sur la syntaxe du langage ASP, mais aussi regardé des programmes préalablement codés afin de s'en inspirer.

Évaluer et améliorer la solution puis la généraliser

On se sert des indicateurs pour vérifier la bonne mise en œuvre des solutions proposées.

Commentaire du stagiaire, Timothée Barbot

Après de longues semaines de solitude dans mon combat avec l'environnement de développement, voici ce que je conseillerais à mon supérieur pour rendre le stage plus intéressant et efficace.

Je commencerais d'abord par changer le serveur hébergeant le site par une machine plus récente. Elle sera ainsi plus stable et proposera des outils et des langages récents avec une communauté active (documentation, exemples, etc.).

Il y a aussi un autre avantage à utiliser des langages récents et actifs : il y a plus de chance que le stagiaire, moi dans le cas présent, est déjà eu un contact avec ce langage, soit durant mon parcours scolaire, soit dans mon temps libre.

Quant au sujet des droits et de la modification du code source, la solution d'une liste de droits tenus à jour permettra d'éviter les pertes temporaires d'accès.

Dans l'ensemble, mon stage m'a demandé beaucoup de réflexions pour parvenir à contourner ces problèmes et un milieu de développement plus favorable m'aurait laissé plus de temps pour réaliser un produit plus avancé et complet.