

Bujumbura International University

BST

BAC 2

PROJET D' UML:

Sujet: WEB SERVICE

Groupe 11:

1. MUNYESHAKA Aimable: 06-157

2. ISHIMWE Elodie: 06-140

3. IRAKOZE Liesse Delicia: 06-186

❖ Définition

A partir de la définition de service Web par le W3C un Web service « est un logiciel conçu pour l'interopérabilité machine-to-machine interaction sur un réseau ». Son interface est décrite dans un format « machine-traitable » prévu par le protocole Web Services Description Language (WSDL). SOAP est un protocole de transport indépendant, considérer comme un protocole d'échange des messages basé sur XML et utiliser pour interagir avec d'autres systèmes. Les données de service particulier et sa structure, qui est ensuite porté dans le cadre de message SOAP (enveloppe, en-tête, corps) est déclarée par le schéma XML langage, ce dernier également incorporé dans le fichier WSDL de modélisation. Dans la littérature, les services Web qui utilisent la combinaison de WSDL et SOAP sont souvent appelés des services Web basés sur SOAP.

Une alternative à ce genre de réalisation du service Web sont axés sur les RESTful Service web. Les services Web RESTful peuvent être décrits avec WSDL ou Web Application

Description Language (WADL) et les données peuvent être échangées dans un format basé sur XML.

Conception et analyse des besoins



Analyse des besoins

Les besoins fonctionnels

Il s'agit des fonctionnalités du système. Ce sont les besoins spécifiant un comportement d'entrée / sortie du Système.

Le système devra permetre :

√ La Gestion des utilisateurs Elle doit

assurer:

- Ajout, suppression, modification et consultation des utilisateurs.
- ✓ La gestion des notes Elle permet

de:

- -Creer les nouveaux notes
- -Enregistrer les notes
- Modifier les notes
- Supprimer les notes
- Consulter les notes

❖ Besoin non-fonctionnelle

Il s'agit des besoins qui caractérisent le système. Ce sont des besoins en matière de performance, de type de matériel ou le type de conception. Ces besoins peuvent concerner les contraintes d'implémentation (langage de programmation, type SGBD, de système d'Exploitation...)

Dans le cadre de ce travail, l'application devra être extensible, c'est-à-dire qu'il pourra y avoir une possibilité d'ajouter ou de modifier de nouvelles fonctionnalités.

L'application devra être capable de :

- > Tourner en réseau
- > Etre compatible avec n'importe quel système d'exploitation.
- > IL faudra aussi noter que l'application devra être hautement sécurisée car les informations ne devront pas être accessibles à tout le monde.

Identification des acteurs et les cas d'utilisation

.utilisateurs : l'utilisateur peut effectuer des taches sur le système .

- S'authentifier(se connecter)
- Editer une nouvelle note
- Modifier une note
- Supprimer une note
- Lister les notes
- Se déconnecter

nouveau utilisateur

- Créer un compte(s'inscrire)
- Editer une nouvelle note
- Modifier une note
- Supprimer une note

Lister ses notes

Description textuelle des cas d'utilisation :

4 Cas d'utilisation : s'authentifier

- **❖** Acteurs principaux :utilisateurs et nouveau utilisateurs
- **❖** Préconditions : lancement de l'application
- ❖ Post conditions : affichage d'un interface d'accueil (succès authentifié)ou d'un message d'erreur(erreur d'authentification) ❖ Scenario nominal :
 - L'utilisateur demande l'interface le formulaire d'authentification d'un compte et l'interface affiche le formulaire d'authentification
 - l'utilisateur remplit le formulaire d'authentification et clique sur le bouton pour se connecter
 - l'interface fait appel au serveur
 - le serveur vérifie dans la bdd si les informations saisit par l'utilisateur concordent avec ceux se trouvant dans la bdd
 - le serveur fait appel à l'interface pour afficher l'interface des notes de l'utilisateur (l'interface principale de l'utilisateur)

Scenarios altérnatif:

les système envoie un message « utilisateur inconnue »

Las d'utilisation : créer un compte

Acteurs principaux : nouveau utilisateur

Préconditions : lancement de l'application

Post-conditions: enregistrement du nouveau utilisateur dans la bdd et l'affichage d'un interface d'accueil (enregistrement réussi)ou d'un message d'erreurs (en cas d'erreur d'enregistrement)

Scenario nominal:

- l'interface affiche le formulaire d'authentification avec un bouton en bas créer un compte
- Le nouveau utilisateur clique sur le bouton
- L'interface affiche le formulaire pour s'inscrire
- Le nouveau utilisateur remplit le formulaire
- L'intérface fait appel au serveur
- Le serveur récupère le données remplit dans le formulaire et les stockes dans la bdd
- Le serveur fait appel a l'interface pour afficher l'interface principale d'un utilisateur (affichant les différents opérations qu'un utilisateur peut faire et ses notes personnels)

Scenario alternatif:

6-2 : les système renvoie le formulaire de création de compte»

4 Cas d'utilisation : éditer une nouvelle note

- ❖ Acteurs principale :utilisateur nouveau et utilisateur
- Préconditions :lancement de l'application ,authentification réussi et être sur l'interface de utilisateur
- ❖ Post-conditions : enregistrement de la note saisie dans la bdd et l'affichage d'un interface affichant la note saisie (enregistrement réussi)ou d'un message d'erreurs (en cas d'erreur d'enregistrement)
- **❖** Scenario nominales:
- après l'authentification
- l'interface affiche tous les notes correspondant à l'utilisateur
- l'utilisateur demande l'interface « éditer note »
- l'interface présente l'interface d'éditer une note « une zone de saisie d'un texte »
- l'utilisateur tape ces notes
- l'interface fait appel au serveur
- le serveur enregistre la nouvelle note dans les notes de l'utilisateur dans la bdd
- l'interface affiche le message de succès d'enregistrement
- le système retourne a l'interface d'accueil (de l'utilisateur affichant les différents notes de l'utilisateurs)

❖ Scenario alternatif:

4-6 le système affiche un message sur l'interface d'éditeur de note « erreur d'enregistrement »

4 Cas d'utilisation :modifier une note

- ❖ Acteurs principale :utilisateur nouveau et utilisateur
- Préconditions :lancement de l'application ,authentification réussi et étre sur l'interface de utilisateur
- ❖ Post-conditions : un update de la note modifier dans la bdd et l'affichage de l'interface principale affichant la note modifier avec les nouvelles modifications (enregistrement réussi)ou d'un message d'erreurs (en cas d'erreur d'enregistrement)

Scenario nominale :

- après authentification
- l'interface affiche tous les notes correspondant à l'utilisateur
- l'utilisateur clique sur la notes a modifier et clique sur le bouton modifier
- le bouton le dirige sur un interface d'éditeur de texte ou il peut modifier sa note
- l'utilisateur ajoute des modifications puis clique sur le bouton modifier
- l'interface fait appel au serveur

- le serveur fait un update dans la base de données
- les serveur fait appelle a l'interface pour retourner a l'interface principale de l'utilisateur avec les nouveaux modifications

4 cas d'utilisation :supprimer une note

- **❖** Acteurs principale :utilisateur nouveau et utilisateur
- Préconditions :lancement de l'application ,authentification réussi et être sur l'interface principale de utilisateur
- ❖ Post-conditions :suppression de la note choisie dans la bdd et l'affichage de l'interface principale affichant les notes restant (suppression réussi)ou d'un message d'erreurs (en cas d'erreur)

- Scenarios nominales
- après authentification
- l'interface affiche tous les notes correspondant a l'utilisateur
- l'utilisateur clique sur la notes a supprimer et clique sur le bouton supprimer
- l'interface fait appel au serveur
- Le serveur récupère l'identifient de la note choisie et la supprime dans la base de donnés
- l'interface est actualiser et la note supprimer n'est plus la

cas d'utilisation :lister les notes

- **❖** Acteurs principale :utilisateur nouveau et utilisateur
- Préconditions :lancement de l'application ,authentification réussi et être sur l'interface de utilisateur
- ❖ Post-conditions: affichage sur l'interface principale des notes suivant comment l'utilisateur les a lister.
- **❖** Scenario nominales:
- après authentification
- l'interface affiche tous les notes correspondant à l'utilisateur
- l'utilisateur clique sur le bouton lister mes notes
- le bouton le dirige sur l'interface de lister les notes (affichant le différents façons de lister ses notes par ex :par ordre alphabétiques par ordre de date ou par ordre d'importance)
- l'utilisateur liste ces notes comme il veut
- l'interface affiche les notes comme l'utilisateur les a lister

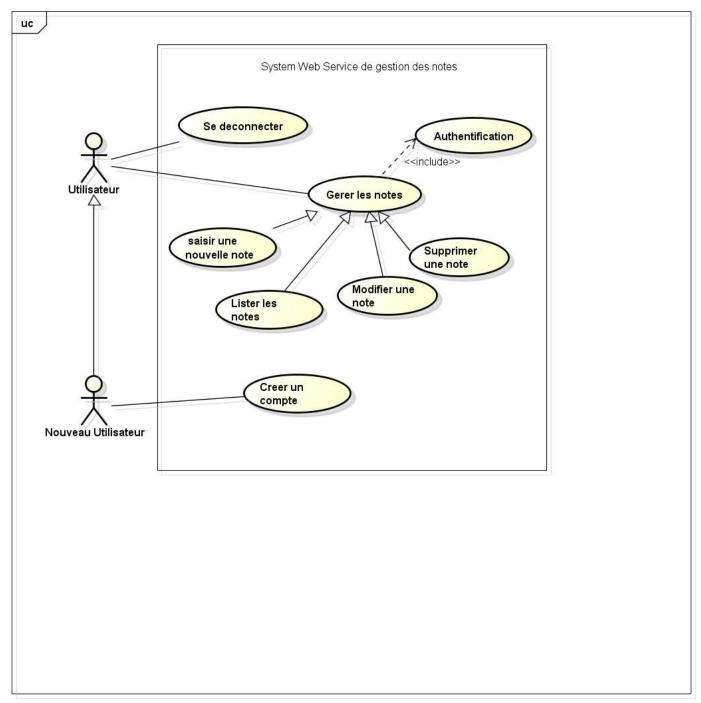
Diagrammes des cas d'utilisation



Définition

Les diagrammes de cas d'utilisation décrivent les services les plus importants rendus par un système.

Partant des acteurs, participants externes qui interagissent avec le système, ils représentent les cas les plus importants du système en cours d'utilisation. Un cas d'utilisation peut être divisé en diagrammes de séquence, qui détaillent les différentes fonctions du cas d'utilisation. Les diagrammes de cas d'utilisation sont créés dans les packages, les classes, les interfaces, les acteurs, les composants et les cas d'utilisation.



powered by Astah

Diagramme de sequences



Les diagrammes de séquence présentent la coopération entre différents objets. Les objets sont définis et leur coopération est représentée par une séquence de messages entre eux.

Les objets peuvent être connectés à des classes existantes ou bien être créés indépendamment de toute classe. Si les objets sont connectés à des classes, les messages peuvent être connectés à des opérations.

Les diagrammes de séquence sont créés dans les interactions.

| Symbole | Désignation |
|--------------------------|----------------------|
| | Objet actif |
| | |
| | Activation |
| | |
| $\leftarrow \rightarrow$ | Mesure |
| ·> | Réécouter le message |

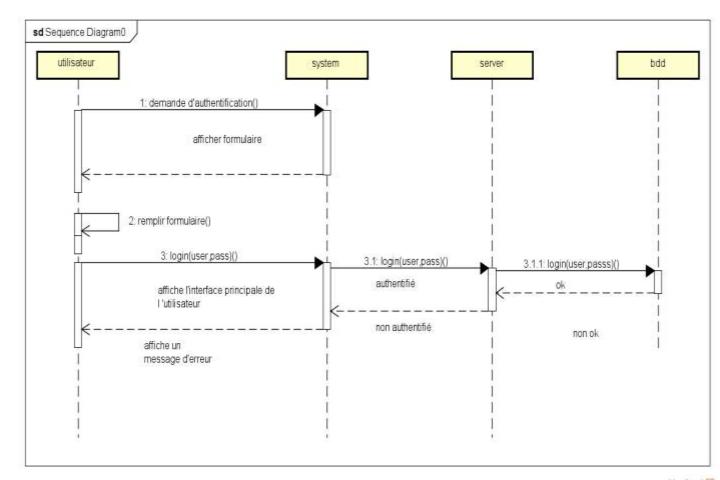
Le formalisme du diagramme de séquence

Cadre d'interaction

Pour les cas plus complexes, on peut intégrer des algorithmes dans les diagrammes de séquences. Par le biais de cadres d'interaction, on peut préciser les spécificités d'un ensemble de messages :

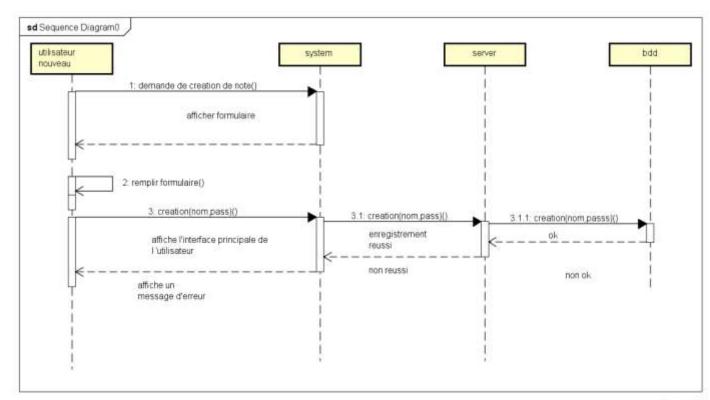
- ightharpoonup alt : fragments multiple alternatifs (si alors sinon) ightharpoonup ightharpoonup optionnel.
- > par : fragment parallèle (traitements concurrents)

* Diagramme de sequence « Authéntification »



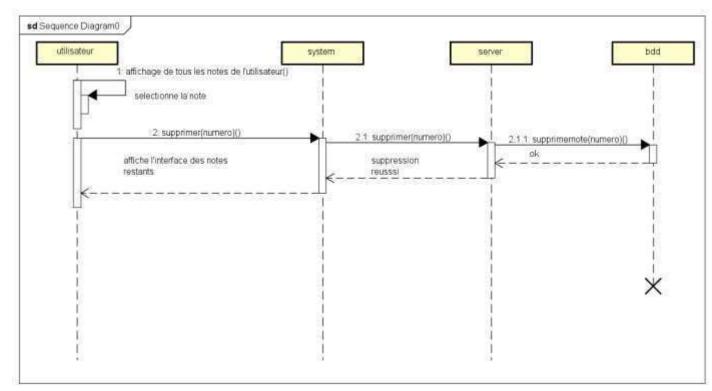
powered by Astah

* Diagramme de séquence « créer un compte »



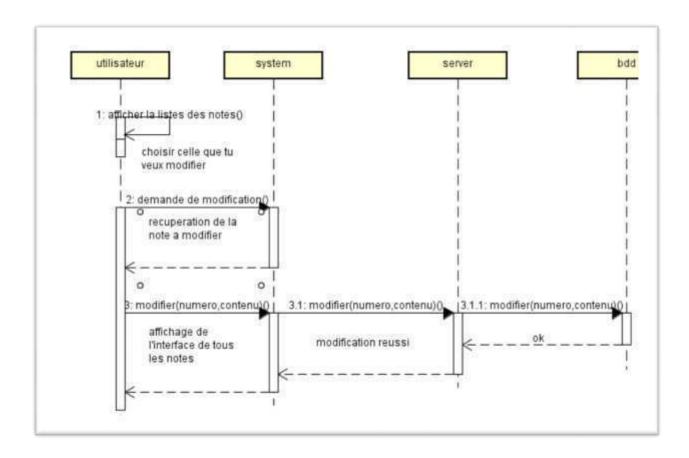
powered by Astabite

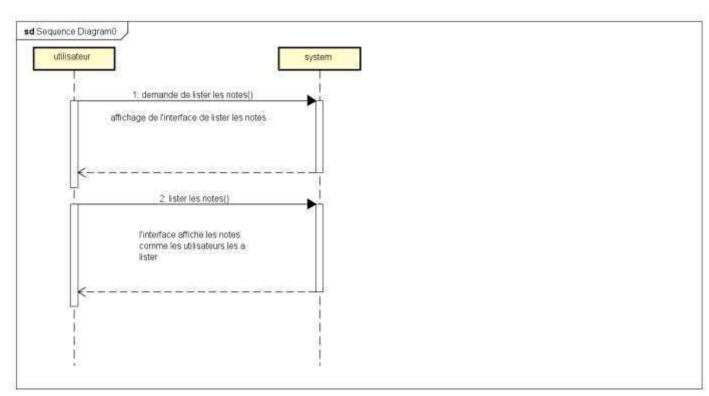
Diagramme de séquence « supprimer une note »



powered by Astability

* Diagramme de séquence « modifier une note »





powered by Astan

Conclusion

Le présent projet de fin d'études a pour but la mise en place d'une application web de gestion des notes. Ce projet nous a aidés à améliorer nos connaissances en informatique. Dans ce projet, nous avons commencé par présenter quelques généralités et notre problématique. Ainsi, se pose la difficulté de l'établissement, nous nous sommes donc occupés d'observer son fonctionnement, d'en relever les défauts, et d'envisager des solutions aux problèmes qu'il rencontre. Ensuite, nous avons cité les besoins afin d'accomplir un tel projet.

Au cours de ce travail, nous avons pu étudier l'approche orientée objet présentée par le langage de modélisation UML pour analyser et spécifier les besoins dans un premier temps.