**©2012 ID-INFORMATIK**

2012

**Webframework ®**

**Module UTILISATEUR**

**Thomas AUGUEY**

Table des matières

[Introduction 4](#_Toc334797600)

[Cahier des charges 5](#_Toc334797601)

[Compte utilisateur 5](#_Toc334797602)

[Données utilisateur 5](#_Toc334797603)

[Session 5](#_Toc334797604)

[Session automatique 5](#_Toc334797605)

[Connexion 5](#_Toc334797606)

[Expiration de la connexion 6](#_Toc334797607)

[Temps de vie des sessions 6](#_Toc334797608)

[Identité 6](#_Toc334797609)

[Adresse 7](#_Toc334797610)

[Contraintes 7](#_Toc334797611)

[Analyse 8](#_Toc334797612)

[Entités 8](#_Toc334797613)

[USER 8](#_Toc334797614)

[SESSION 8](#_Toc334797615)

[CONNECTION 8](#_Toc334797616)

[IDENTITY 8](#_Toc334797617)

[ADDRESS 9](#_Toc334797618)

[Domaines 10](#_Toc334797619)

[STREET\_PREFIX 10](#_Toc334797620)

[SEX 10](#_Toc334797621)

[Relations 11](#_Toc334797622)

[Fonctions 13](#_Toc334797623)

[Acteurs 14](#_Toc334797624)

[Visiteur 14](#_Toc334797625)

[Utilisateur 14](#_Toc334797626)

[Administrateur 14](#_Toc334797627)

[Système 14](#_Toc334797628)

[Cas d’utilisation 15](#_Toc334797629)

[Créer compte 15](#_Toc334797630)

[Connexion client 17](#_Toc334797631)

[Check Connexion 19](#_Toc334797632)

[Déconnexion d’un utilisateur 20](#_Toc334797633)

[Déconnexion de tous les utilisateurs 21](#_Toc334797634)

[Activer un compte utilisateur 22](#_Toc334797635)

[Contrats d’opération 23](#_Toc334797636)

[Connect 23](#_Toc334797637)

[Create 24](#_Toc334797638)

[Réalisation 25](#_Toc334797639)

[Applications Web (Front Office) 25](#_Toc334797640)

[Liste des Pages HTML 25](#_Toc334797641)

[Liste des Requêtes HTTP 25](#_Toc334797642)

[Applications Web (Back-Office) 26](#_Toc334797643)

[Liste des Pages HTML 26](#_Toc334797644)

[Liste des Requêtes HTTP 26](#_Toc334797645)

[Liste des Fichiers/Dossiers 26](#_Toc334797646)

[Base de données SQL 27](#_Toc334797647)

[Liste des tables 27](#_Toc334797648)

[Liste des Fonctions 27](#_Toc334797649)

[Liste des Scripts 27](#_Toc334797650)

# Introduction

*WebFrameWork* a besoin de gérer des comptes utilisateurs pour privilégier l’accès aux données privé en ligne.

## Dépendances

* Webframework
* Webframework Module Client

# Cahier des charges

Le gestionnaire sera implanter comme module de *WebFrameWork* et sera nommé **USER**.

Les utilisateurs pourront se connecter via une interface HTML avec un nom d’utilisateur et un mot de passe. Après connection l’utilisateur pourra accéder et modifier ses données personnelles de façon sécurisée.

## Compte utilisateur

Un compte utilisateur est identifié par un nom d’utilisateur unique. Un mot-de-passe personnalisable permet la connexion du compte à une session.

Un compte utilisateur possède aussi une adresse électronique unique permettant de recevoir des messages du système.

## Données utilisateur

Les données sont conservées dans une entrée du module Client. L’Entrée conserve l’ensemble des fichiers et données variables d’un utilisateur.

Les règles d’usage sont définit dans le document technique : **module Client***.*

## Session

Une session localise une ressource sur le serveur. Une ressource peut être un dossier ou un fichier.

Une session renseigne les données suivants :

1. Chemin relatif aux données (chemin d’accès privé sur le serveur)

## Session automatique

Identifie un accès vers le dossier client d’un utilisateur (**module client**).

Dans ce cas, la session est gérée automatiquement lors de la connexion d’un utilisateur.

L’identifiant d’une session automatique est précédé du caractère UNDERSCORE ‘\_’.

## Connexion

Créé un lien entre une session et un utilisateur.

Une connexion est identifiée par l’adresse IP du client et son nom d’utilisateur.

Une connexion enregistre la date en cours à chaque fois qu’un utilisateur l’actualise (check).

Si un certain temps d’inactivité (temps de vie) est dépassé, la connexion est automatiquement supprimée.

Une connexion renseigne les données suivants :

1. Adresse IP
2. Date du dernier accès
3. Durée de vie (relatif à la date d’accès)
4. Lien symbolique vers les données (chemin d’accès public sur le serveur)

## Expiration de la connexion

La déconnexion d’un utilisateur intervient après un certain temps d’inactivité, c’est une tâche système qui s’occupe d’exécuter le code de fermeture de la connexion.

Pour maintenir une activité, l’utilisateur ‘*check’* la connexion par le biais d’une requête HTTP. La date d’expiration s’en voit reculée et la connexion maintenu.

## Temps de vie des sessions

Tout comme les connexions, une session possède une date d’expiration.

La suppression d’une session engendre automatiquement la déconnexion de tous les utilisateurs qui en dépendent.

Une tache système permettra l’exécution du code.

## Identité

Définit une personne physique.

Une identité renseigne les champs suivants :

1. Nom
2. Prénom
3. Date d’anniversaire
4. Sexe (Domaine : Sexe)

## Adresse

Définit une adresse postale.

# Modèle de données

Le modèle des données persistantes est conçu sur le modèle Merise.

Le document « **mcd/user.mcd** » propose une version schématisé du modèle conceptuel de données, ce modèle est utilisé pour générer le script d’installation SQL. [Vous pouvez ouvrir ce fichier avec le logiciel **PowerAMC** de **Sybase**].

## Entités

Liste les tables ajoutées à la base de données.

Le tableau des entités est définit ainsi :

* Le **type** de donnée est écrit de façon générique (VARCHAR, CHAR, INTEGER, REAL, …)
* La **contrainte** est appliquée sur les champs par la base de données (Domaines, Clés, Check, …)
* Le **format**, définit plus bas, définit la syntaxe pour les chaines de caractères

### USER

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribut | Description | Type | Contrainte | Format |
| USER\_ID | Identifiant | VARCHAR(64) | PRIMARY KEY | Identifier |
| USER\_PWD | Mot-de-passe | VARCHAR(64) | NOT NULL | Password |
| CLIENT\_ID | Identifiant du dossier client | VARCHAR(64) | NOT NULL | Name |
| USER\_MAIL | Adresse électronique de contact | VARCHAR(80) | NOT NULL | Mail |

### SESSION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribut | Description | Type | Contrainte | Format |
| SESSION\_ID | Identifiant | CHAR(64) | PRIMARY KEY | Name |
| LOCAL\_PATH | Emplacement de la ressource dans le système de fichier (PRIVE) | CHAR(260) | NOT NULL | UnixPath |

### CONNECTION

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribut | Description | Type | Contrainte | Format |
| CLIENT\_IP | Adresse IP du client | VARCHAR(15) | PRIMARY KEY | IPv4 |
| LAST\_ACCESS | Date de dernier accès à la session | DATE | NOT NULL |  |
| LIFE\_TIME | Temps de vie avant déconnexion | INTEGER | NOT NULL |  |
| LINK\_PATH | Emplacement du lien symbolique dans le système de fichier (PUBLIC) | CHAR(260) | NOT NULL |  |

### IDENTITY

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribut | Description | Type | Contrainte | Format |
| IDENTITY\_ID | Identifiant | INTEGER | PRIMARY KEY |  |
| FIRST\_NAME | Prénom | CHAR(80) | NOT NULL |  |
| LAST\_NAME | Nom | CHAR(80) | NOT NULL |  |
| BIRTH\_DAY | Date de naissance | DATE | NOT NULL |  |
| SEX | Sexe | DOMAIN(SEX) | NOT NULL |  |

### ADDRESS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribut | Description | Type | Contrainte | Format |
| ADDRESS\_ID | Identifiant | INTERGER | PRIMARY KEY |  |
| STREET\_NAME | Nom de rue | VARCHAR(128) | NOT NULL |  |
| STREET\_PREFIX | Préfix de rue | DOMAIN (STREET\_PREFIX) |  |  |
| STREET\_NUMBER | Numéro de rue | NUMBER(4,0) | NOT NULL |  |
| ZIP\_CODE | Code postal | VARCHAR (10) | NOT NULL |  |
| CITY\_NAME | Ville | VARCHAR (80) | NOT NULL |  |
| COUNTRY\_NAME | Pays | VARCHAR (80) | NOT NULL |  |
| BUILD\_NUMBER | Numéro de bâtiment | NUMBER(4,0) |  |  |
| APT\_NUMBER | Numéro appartement | NUMBER(4,0) |  |  |

## Domaines

### STREET\_PREFIX

Préfix d’une rue.

|  |  |
| --- | --- |
| Valeur : CHAR(3) | Description |
| BIS | Bis |
| TER | Ter |

### SEX

Sexe de la personne.

|  |  |
| --- | --- |
| Valeur : CHAR(1) | Description |
| M | Homme |
| F | Femme |

## Formats

Tous les formats utilisé hérites du projet **Webframework**, reportez-vous au document technique lié au projet pour plus d’informations.

## Relations

1. USER : CONNECTION
   1. Un **utilisateur** peut ouvrir plusieurs **connexions**
   2. Une **connexion** peut être ouverte par un **utilisateur**
2. CONNECTION : SESSION
   1. Une **session** peut être attaché à plusieurs **connexions**
   2. Une **connexion** peut être attachée à une seule **session**
3. USER : IDENTITY
   1. Un **utilisateur** peut renseigner une **identité**
   2. Une **identité** peut être renseignée par plusieurs **utilisateurs**
4. IDENTITY: ADDRESS
   1. Une **identité** peut posséder une **adress**e
   2. Une **adresse** peut être posséder par plusieurs **identités**

## Contraintes

1. Une **connexion** expire après un certain temps d’inactivité, dans ce cas, la connexion est automatiquement supprimée.
2. Pour être utilisé, un **compte utilisateur** doit d’abord être validé avec un code confidentiel envoyé par mail lors de sa création ou par demande du client.
3. Un utilisateur ne peut pas se connecter sous plusieurs adresses IP à la fois, cependant plusieurs utilisateurs peuvent utiliser la même adresse IP. (*Unicité de l’attribut* ***USER\_ID*** *dans la table* ***CONNECTION***).
4. Un lien symbolique est propre à une seule connexion (Unicité de l’attribut **LINK\_PATH** dans la table **CONNECTION).**
   1. Si un lien était partagé entre plusieurs connexions, une connexion fermée (supprimée) pourrait continuer à utiliser ce lien.

# Operations système

Les opérations systèmes sont conçues sur le modèle UML.

Le document « **mcd/operations.moo** » propose une version schématisé des cas d’opération. [Vous pouvez ouvrir ce fichier avec le logiciel **PowerAMC** de **Sybase**].

## Acteurs

### Visiteur

Utilisateur non enregistré.

### Utilisateur

Utilisateur enregistré, ayant créé un compte utilisateur.

### Administrateur

Utilisateur le l’interface de gestion.

### Système

Définition du système avec ses différentes couches de dépendances.

**Système Langage**

1. Application HTML/Javascript
2. Interface Serveur PHP
   1. Système embarqué (système de fichier) Console UNIX
3. Interface de Base de données PG/SQL PL/pgSQL
   1. Base de données SQL SQL

## 

## Résultats d’opération

Les résultats d’opération sont basés sur le modèle du projet **WebframeWork**. Pour plus de détails à ce sujet reportez-vous au document technique concerné.

Noté que ses codes d’erreurs viennent s’ajouter aux codes déjà existant dans le projet **Webframework**. Certains d’entre eux peuvent être utilisé par les opérations présentes ci-dessous.

### Détail des codes

**Code Sous-Code Description**

ERR\_OK USER\_CREATED L’Utilisateur est créé

ERR\_OK USER\_DELETED L’Utilisateur été Supprimé

ERR\_FAILED USER\_EXISTS L’Utilisateur existe déjà

## Cas d’utilisation

### Créer compte

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC1 | | | Créer un compte utilisateur | |  |
| Intentions et objectifs | | | Ouvrir une connexion sur une session client (automatique) | | |
| Conditions | | | Les champs suivants sont fournis :   * Adresse eMail * Nom d’utilisateur * Mot-de-passe | | |
| Résultat | | | Une entrée USER est crée dans la base de données.  Un dossier client est créé dans le système de fichier. | | |
| Déclenchement | | | Le visiteur demande la création d’un compte utilisateur | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **CO** | **Action** | | |
|  | Visiteur | **S1** | Saisie des informations | | |
|  | Système (SYS) | **S2** | Crée le dossier de données dans le module client (E1) | | |
|  | Système (SGDB) | **S3** | Crée le compte utilisateur (E2) | | |
|  | Système (SYS) | **S4** | Crée les **liens d’accès** dans le système de fichier (E3) | | |
|  | Système (SYS) | **S5** | Crée **le lien de vérrou** dans le système de fichier (E3) | | |
|  | Système (SGBD) | **S6** | Envoie un mail d’activation (A1) | | |
|  | Système (SYS) | **S7** | Affiche un message de confirmation (A2) | | |
| Scénario alternatif | | |  | | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Système | **A1** | Le mail d’activation ne peut pas être envoyé.  [Appel du cas d’utilisation : Activer un compte utilisateur] | | |
|  | Système | **A2** | L’Utilisateur confirme l’activation du compte.  [Appel du cas d’utilisation : Activer un compte utilisateur] | | |
| Exceptions | | |  | | |
|  | **Code d’erreur** | **Etape** | **Action** | | |
|  | Client : [UC.Create] | **E1** | Le compte client ne peut pas être créé. | | |
|  | User : [CO.Create] | **E2** | Le compte utilisateur ne peut pas être créé. | | |
|  | SYS\_LINK\_FILE | **E3** | Le lien symbolique ne peut pas être créé. | | |

### Connexion client

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC1 | | | Connexion à une session utilisateur | |  |
| Intentions et objectifs | | | Ouvrir une connexion sur une session client (automatique) | | |
| Conditions | | | L’Adresse IP du client est connue.  Le temps de vie de la connexion est connu.  Le login et le mot-de-passe utilisateur est connue | | |
| Résultat | | | Une connexion est créée dans la base de données.  Une session ‘automatique’ est créée dans la base de données. | | |
| Déclenchement | | | Le visiteur demande une connexion à sa session utilisateur | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Visiteur | **S1** | Saisie du mot-de-passe et du nom d’utilisateur | | |
|  | Système (REQ) | **S2** | Vérifie le nom d’utilisateur et le mot-de-passe (A1, E1) | | |
|  | Système (REQ) | **S3** | Vérifie l’existence d’une **connexion** pour cet utilisateur (E2) | | |
|  | Système (REQ) | **S4** | Crée les **liens d’accès** dans le système de fichier (E3) | | |
|  | Système (REQ) | **S5** | Crée la **tache de fermeture** automatique de connexion à la **date d’expiration** (E4) | | |
|  | Système (SGBD) | **S6** | Vérifie l’existence de la **session** automatique liée à cet utilisateur | | |
|  | Système (SGBD) | **S7** | Crée une **connexion** avec l’adresse IP du client, l’identifiant de session, l’identifiant d’utilisateur et la date d’accès. | | |
|  | Système (REQ) | **S8** | Retourne l’identifiant de connexion à l’utilisateur | | |
| Scénario alternatif | | |  | | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Système | **A1** | Si le compte n’existe pas, informe l’utilisateur que le mot de passe et/ou le nom d’utilisateur est invalide.  [Retour à l’étape S1] | | |
|  | Système | **A2** | Si la session n’existe pas, la session est créée avec le nom de dossier client associé au compte utilisateur et le chemin d’accès. | | |
| Exceptions | | |  | | |
|  | **Code d’erreur** | **Etape** | **Action** | | |
|  | USER\_UNACTIVATED | **E1** | Le compte n’est pas actif. | | |
|  | USER\_ALERADY\_CONNECTED | **E2** | Une connexion existe déjà avec cette adresse IP. | | |
|  | SYS\_CREATE\_FILE | **E3** | Le lien symbolique ne peut pas être créé. | | |
|  | SYS\_CREATE\_TASK | **E4** | La tâche ne peut pas être créée. | | |

### Check Connexion

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC2 | | | Actualise l’état de la connexion | |  |
| Intentions et objectifs | | | Vérifier l’état de la connexion et renvoie le chemin d’accès aux fichiers. | | |
| Conditions | | | L’identifiant de connexion est connu  ou  Le login, le mot-de-passe et l’IP de l’utilisateur est connue | | |
| Résultat | | | La date d’expiration est actualisée (temps de vie + date en cours).  Le chemin d’accès au système de fichier est retourné. | | |
| Déclenchement | | | L’utilisateur actualise l’état de sa connexion | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Utilisateur | **P1** | Saisie l’identifiant de connexion | | |
|  | Système | **P2** | Vérifie l’existence de la connexion (E1, E2) | | |
|  | Système | **P3** | Actualise la date d’expiration (date system + durée de vie) | | |
|  | Système | **P4** | Actualise la tâche fermeture automatique (E3) | | |
|  |  |  |  | | |
|  |  |  |  | | |
|  |  |  |  | | |
| Scénario alternatif | | |  | | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Système | **A1** | Si la connexion n’existe pas, informe l’utilisateur que le mot de passe et/ou le nom d’utilisateur est invalide.  [Retour à l’étape P1] | | |
|  | Système | **A2** | Si la session n’existe pas, la session est créée avec le nom de dossier client associé au compte utilisateur et le chemin d’accès. | | |
| Exceptions | | |  | | |
|  | **Code d’erreur** | **Etape** | **Action** | | |
|  | USER\_CONNECTION\_NOT\_FOUND | **E1** | La connexion n’existe pas. | | |
|  | USER\_CONNECTION\_IP\_REFUSED | **E2** | L’Adresse IP ne correspond pas. | | |
|  | SYS\_UPDATE\_TASK | **E3** | La tâche ne peut pas être mise à jour. | | |

### Déconnexion d’un utilisateur

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC3 | | | Déconnexion d’un utilisateur | |  |
| Intentions et objectifs | | | Terminer une connexion entre un compte et une session | | |
| Conditions | | |  | | |
| Résultat | | | La connexion et supprimée de la base de données. | | |
| Déclenchement | | | L’Utilisateur demande une déconnection | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Utilisateur | **P1** | Saisie son mot de passe et son nom d’utilisateur | | |
|  | Système | **P2** | Vérifie l’exactitude des informations de connexion (A1) | | |
|  | Système | **P3** | Vérifie si la session est automatique (A2). | | |
|  | Système | **P4** | Supprime la session. | | |
|  | Système | **P5** | Retourne le code de résultat ERR\_OK. | | |
| Scénario alternatif | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  |  |  |  | | |
|  | Système | **A2** | [Retour à l’étape P5] | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  |  |  |  | | |
| Exceptions | | |  |  | |
|  |  | **Code** | **Action** | | |
|  |  |  |  | | |

### Déconnexion de tous les utilisateurs

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC4 | | | Déconnexion de tous les utilisateur | |  |
| Intentions et objectifs | | | Terminer toutes les connexions utilisateur | | |
| Conditions | | |  | | |
| Résultat | | | Toutes **les connexions** sont supprimées de la base de données.  Toutes **les sessions** **automatiques** sont supprimées de la base de données.  Tous les liens de sessions automatiques sont supprimés du **système de fichiers**. | | |
| Déclenchement | | | L’Administrateur demande la fermeture de toutes les connexions | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Administrateur | **P1** | Confirme la fermeture des connexions | | |
|  | Système | **P2** | Supprime toutes les connexions | | |
|  | Système | **P3** | Supprime toutes les sessions automatiques | | |
|  | Système | **P5** | Retourne le code de résultat ERR\_OK. | | |
| Scénario alternatif | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  |  |  |  | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  |  |  |  | | |
| Exceptions | | |  |  | |
|  |  | **Code** | **Action** | | |
|  |  |  |  | | |

### Activer un compte utilisateur

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| UC5 | | | Activer un compte utilisateur | |  |
| Intentions et objectifs | | | Permet d’activer l’utilisation du compte client | | |
| Conditions | | |  | | |
| Résultat | | | Le compte est actif. | | |
| Déclenchement | | | L’Utilisateur demande activation de compte | | |
| Scénario principal | | |  |  | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Utilisateur | **S1** | Saisie du code d’activation et du nom d’utilisateur (A1) | | |
|  | Système | **S2** | Vérifie l’exactitude des informations de connexion (A2) | | |
|  | Système | **S3** | Supprime le lien d’inactivation pour cet utilisateur dans le système de fichier. | | |
|  | Système | **S4** | Averti l’utilisateur que le compte est actif et prêt à être connecté. | | |
| Scénario alternatif | | |  | | |
|  | **Acteur** | **Etape** | **Action** | | |
|  | Système | **A1** | Si l’utilisateur ne connaît pas son code d’activation, le système lui renvoi par mail.  [Retour à l’étape S1] | | |
|  | Système | **A2** | Si l’utilisateur entre un code d’activation incorrecte, le système demande à nouveau la saisie des informations.  [Retour à l’étape S1] | | |
| Exceptions | | |  | | |
|  |  | **Code** | **Action** | | |
|  |  |  |  | | |

## Contrats d’opération

### Create

|  |  |  |  |
| --- | --- | --- | --- |
| Create | | Créé un utilisateur |  |
| Intentions et objectifs | | Crée une entrée utilisateur | |
| Système | | SGBD | |
| Résultat | | Une entrée utilisateur est créée dans la base de données. | |
| Algorithme | |  | |
|  | | Insertion dans la table USER | |
| Implémentation | |  | |
| PL/PGSQL | | Fichier : /sql/func.sql  Fonction : wfw\_user.create() | |
| Exceptions | |  | |
|  | USER\_EXISTS | L’Utilisateur existe déjà | |  |
| Entrées | |  | |
|  | USER\_ID | Identificateur (Nom d’utilisateur) | |
|  | USER\_PWD | Mot-de-Passe | |
|  | CLIENT\_ID | Identificateur (Dossier client) | |
|  | USER\_MAIL | Adresse Mail | |
| Retour | |  | |
|  | USER\_CREATED | L’Utilisateur est créé | |
|  | USER\_EXISTS | L’Utilisateur existe déjà. | |
|  | NOT\_SPECIFIED | Exception système non gérée. | |

### Connect

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| UC3 | | Déconnexion d’une session | |  |
| Intentions et objectifs | | Terminer une connexion entre un compte et une session | | |
| Conditions | |  | | |
| Résultat | | La connexion et supprimée de la base de données. | | |
| Déclenchement | | L’Utilisateur demande une déconnection | | |
| Algorithme | |  |  | |
|  | SI ( L’Utilisateur existe )  Mes a jour   * l’Id de session * le temps de vie de la connexion   SINON  Crée une connexion avec   * l’Id utilisateur * l’IP du client * l’Id de session * le temps de vie de la connexion   FINSI | | | |
| NOM OPERATION | | Connect |  | |
| CODES : | |  |  | |
| PL/PGSQL | | <mod\_user>/sql/func.sql |  | |

# Maquettage

## Vues

* Connexion
* Demande de mot de passe

## Plan de navigation

# Réalisation

## Applications Web (Front Office)

### Liste des Pages HTML

|  |  |  |
| --- | --- | --- |
| Fichier | Emplacement | Description |
| user\_login.html | private/template/pages | Ouvre une session utilisateur |
| user\_logout.html | private/template/pages | Ferme une session utilisateur |
| user\_check.html | private/template/pages | Crée un compte utilisateur |
| … | private/template/pages | Supprime un compte utilisateur |

### Liste des Requêtes HTTP

|  |  |  |
| --- | --- | --- |
| Fichier | Emplacement | Description |
| open.php | req/user | Ouvre une session utilisateur |
| close.php | req/user | Ferme une session utilisateur |
| create.php | req/user | Crée un compte utilisateur |
| delete.php | req/user | Supprime un compte utilisateur |

## Applications Web (Back-Office)

### Liste des Pages HTML

|  |  |  |
| --- | --- | --- |
| Fichier | Emplacement | Description |
| user.html | private/template/admin | Gestion des sessions en cours |

#### Liste des Popup HTML

|  |  |  |
| --- | --- | --- |
| Fichier | Emplacement | Description |
| user.html | private/template/admin | Gestion des sessions en cours |

### Liste des Requêtes HTTP

|  |  |  |
| --- | --- | --- |
| Fichier | Emplacement | Description |
| open.php | private/req/user | Ouvre une session utilisateur |
| close.php | private/req/user | Ferme une session utilisateur |
| create.php | private/req/user | Crée un compte utilisateur |
| delete.php | private/req/user | Supprime un compte utilisateur |

### Liste des Fichiers/Dossiers

|  |  |  |
| --- | --- | --- |
| Emplacement | Fichier | Description |
| private/user/validate | \*.\* | Liens symboliques sur **les fichiers clients(1)** en attente de validation |
| private/user/bymail | \*.\* | Liens symboliques sur **les fichiers clients(1)** nommés par mail |
| private/user/byid | \*.\* | Liens symboliques sur **les fichiers clients(1)** nommés par identifiant |
| data/module | user.png | Icone graphique du module |

1. Fichiers de données gérés par le module CLIENT.

## Base de données SQL

### Liste des tables

|  |  |
| --- | --- |
| Nom | Usage |
| wfw\_user.user | Utilisateurs |
| wfw\_user.connection | Connexions |
| wfw\_user.session | Sessions |
| wfw\_user.identity | Identités (nom, prénom, etc...) |
| wfw\_user.address | Adresses postal |

### Liste des Fonctions

|  |  |
| --- | --- |
| Fonction | Description |
| wfw\_user.create\_user | Crée un utilisateur |
| wfw\_user.create\_session | Crée une session |
| wfw\_user.delete\_user | Supprime un compte utilisateur |
| wfw\_user.connect | Connecte un compte à une session |
| wfw\_user.connect\_client | Connecte un compte à une session client |
| wfw\_user.disconnect\_all | Termine toutes les connections |
| wfw\_user.disconnect\_all\_client | Termine toutes les connections créées avec une session client |
| wfw\_user.disconnect | Termine une connexion |
| wfw\_user.close\_unused\_session | Ferme les sessions non utilisées |

### Liste des Scripts

|  |  |  |  |
| --- | --- | --- | --- |
| Exécution | Fichier | Description | Rôle |
| 1 | create.sql | Crée le schéma | Intégration |
| 2 | tables.sql | Crée les tables et les contraintes | Intégration |
| 3 | init.sql | Initialise les contraintes additionnels, données globales etc... | Intégration |
| 4 | func.sql | Crée les objets et fonctions | Intégration |
| 5 | populate.sql | Initialise les données de la base | Jeu de test |