

SPÉCIFICATION – ADMINISTRATION PETALS

Application Web pour l'Administration de Petals ESB



Auteur :
Vincent Zurczak
Version : 1.1
Date : 22 Mars 2013

Table des matières

Contexte.....	4
Description.....	4
Utilisateurs Cible.....	4
Conditions d'Utilisation.....	4
Besoins.....	5
Cas d'Usage.....	6
Visualiser une topologie de nœuds Petals.....	6
Synchroniser les annuaires de services entre nœuds Petals.....	6
Interroger l'annuaire de services de Petals.....	7
Lister l'ensemble des artefacts déployés sur un nœud Petals.....	7
Visualiser les propriétés d'un nœud Petals.....	7
Installer un artefact Petals.....	8
Désinstaller un artefact Petals.....	8
Modifier le cycle de vie d'un artefact Petals.....	9
Modifier à chaud des paramètres de Petals.....	9
Modifier à chaud des niveaux de log.....	10
Définir ses préférences.....	10
Tâches.....	11
Concepts.....	11
Tâches liées à une topologie.....	12
Visualiser une Topologie.....	12
Rechercher un nœud dans la Topologie.....	12
Synchroniser les annuaires de services entre nœuds Petals.....	13
Tâches liées à un nœud Petals.....	13
Sélectionner un nœud Petals.....	13
Accéder au conteneur d'un nœud Petals.....	13
Accéder aux artefacts déployés sur un nœud Petals.....	14
Tâches liées au conteneur d'un nœud Petals.....	14
Visualiser les propriétés d'un nœud Petals.....	14
Modifier à chaud les niveaux de log (conteneur).....	14
Modifier à chaud des paramètres du conteneur.....	15
Lister les services.....	15
Interroger l'annuaire de services du nœud.....	16
Synchroniser l'annuaire de service du nœud.....	16
Tâches liées aux artefacts d'un nœud Petals.....	16

Installer un artéfact Petals.....	16
Lister les artéfacts déployés sur un nœud Petals.....	17
Désinstaller un artéfact Petals.....	18
Modifier le cycle de vie d'un artéfact Petals.....	18
Modifier à chaud les niveaux de log (composants).....	19
Modifier à chaud des propriétés d'un composant Petals.....	19
Tâches liées aux préférences.....	20
Découvrir une topologie au travers d'un nœud Petals.....	20
Découvrir une topologie au travers d'un fichier topology.xml.....	20
Changer le nombre d'éléments affichés par page (pagination).....	20
Tableau Associatif entre cas d'usage et tâches.....	21
IHM Abstraite.....	22
Charte Graphique.....	22
Maquettes – Topologie.....	23
Maquettes – Nœud Petals.....	24
Maquettes – Conteneur.....	25
Maquettes – Artéfacts.....	25
Maquettes – Préférences.....	27
Contraintes Techniques et Implémentation.....	28
Contraintes Techniques.....	28
Gestion des Erreurs.....	29
Gestion des Préférences.....	29

Contexte

Description

Il s'agit de décrire une application web qui permet d'administrer un ensemble de nœuds Petals (on parle de « topologie » Petals).

Cette application est une alternative à **Petals-CLI**, le client en ligne de commande qui permet d'administrer une topologie Petals. Cette application doit servir pour les démonstrations, en conférences et avant-ventes, ainsi que dans les formations et les environnements de production qui préfèrent utiliser un navigateur web pour administrer Petals.

C'est donc une application qui doit convenir à différents types d'utilisateurs.

Utilisateurs Cible

On différencie 3 types d'utilisateurs :

- Les **consultants Petals** sont des gens expérimentés sur l'ESB. Ils recherchent un outil qui leur fait gagner du temps et leur offre un confort d'utilisation supérieur à un outil en ligne de commande. Ce sont des utilisateurs réguliers, qui connaissent bien cette application. Pour eux, cette application est une boîte à outils pour exécuter une action ou préparer un diagnostic sur l'ESB.
- Les **exploitants** sont des gens qui supervisent un environnement de production dans lequel peut tourner un ensemble de nœuds Petals. Ce sont des utilisateurs occasionnels, qui ne se rappelleront pas de toutes les procédures possibles. Pour eux, cette application est un moyen d'exécuter une action d'administration au sein de l'ESB.
- Les **novices** sont des gens qui utiliseront cette application que peu de fois dans leur vie. On va les rencontrer en formation (des gens qui veulent se faire une idée sur Petals ESB ou qui vont encadrer des équipes de projet) ou bien en avant-ventes. Pour eux, cette application est un moyen d'appréhender le bus et ses concepts.

Conditions d'Utilisation

Cette application s'accède comme un site Web.

Elle est utilisée au travers d'un navigateur Web pouvant s'exécuter sur un ordinateur fonctionnant sous Windows, Linux ou Mac OS. Cette application doit permettre d'administrer un ou plusieurs nœuds Petals, que ceux-ci soient sur la même machine qu'elle ou bien qu'ils soient à distance. En d'autres termes, l'utilisateur n'a pas à savoir où sont déployés tous les nœuds de la topologie pour les administrer.

Il doit cependant savoir où se trouve l'un des nœuds de cette topologie pour pouvoir accéder aux autres.

Cette application permet d'effectuer des actions unitaires, voire groupées, sur des nœuds ou des artefacts Petals.

Besoins

Voici une liste haut-niveau des besoins à couvrir par cette application. Ils sont déclinés en cas d'usage dans la section suivante.

- ◆ Visualiser une topologie de nœuds Petals.
- ◆ Synchroniser les annuaires de services entre nœuds Petals.
- ◆ Interroger l'annuaire de services de Petals.
- ◆ Lister l'ensemble des artefacts déployés sur un nœud Petals.
- ◆ Visualiser les propriétés d'un nœud Petals (environnement et propriétés Petals).
- ◆ Installer un artefact Petals.
- ◆ Désinstaller un artefact Petals.
- ◆ Modifier le cycle de vie d'un artefact Petals.
- ◆ Modifier à chaud des paramètres de Petals (taille des pools dans les composants et les modules du conteneur).
- ◆ Modifier à chaud le niveau de log d'un composant ou d'un module du conteneur.
- ◆ Définir ses préférences.

Par artefact Petals, on entend **librairies partagées** (*shared libraries*), **composants** (*service engines* ou *binding components*), **service assemblies** et **service-units**.

Il est important de noter qu'il n'y a pas d'équivalence *un pour un* entre Petals CLI et cette application web. Notamment, cette application ne supporte pas les scripts d'administration et ce n'est pas son but.

Il faut noter aussi que le mot **besoin** peut être remplacé par **fonctionnalité** ou **cas d'usage**. Les fonctionnalités sont destinées à tous les types d'utilisateurs. Simplement, une fonctionnalité peut avoir plus d'intérêt pour certains utilisateurs que pour d'autres.

Cas d'Usage

Un diagramme UML des cas d'usage n'a pas été jugé pertinent.

Visualiser une topologie de nœuds Petals

L'application doit permettre de visualiser, au travers d'un graphe, l'ensemble de la topologie Petals. Cette représentation doit s'adapter à la taille de l'écran tout en supportant un grand nombre de nœuds.

Cette représentation n'a pas à tenir compte des notions de domaine et de sous-domaine, qui ne sont pour l'instant pas utilisées dans Petals ESB.

Pour chaque nœud, le nom du nœud devra apparaître clairement.

Les nœuds maîtres seront distingués par une icône ou un overlay, afin de les distinguer des nœuds esclaves.

Le survol de chaque nœud par la souris doit afficher certaines informations additionnelles, telles que l'adresse IP du serveur sur lequel il est déployé, ainsi que la description du nœud et son activité (démarré ou à l'arrêt).

On doit également pouvoir ordonner l'affichage des nœuds, que ce soit par nom de nœud ou par hôte. Enfin, il faut pouvoir filtrer la représentation grâce à un nom d'hôte ou un nom de conteneur, afin de ne pas obligatoirement tout afficher.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Découvrir rapidement une topologie en intervention.	Visualiser son domaine d'exploitation dans son ensemble.	S'approprier l'approche distribuée de Petals ESB.

Synchroniser les annuaires de services entre nœuds Petals

L'application doit permettre de synchroniser les annuaires de services entre nœuds Petals. Cette synchronisation peut se faire pour un nœud donné, mais n'est pas disponible pour un nœud maître.

Elle peut aussi se faire pour l'ensemble de la topologie.

Dans ce dernier cas, la progression de l'opération doit être reportée à l'utilisateur.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices

Intérêt pour les Utilisateurs		
Forcer simplement une synchronisation entre nœuds Petals sans redémarrer un ou des nœuds.	Forcer simplement une synchronisation entre nœuds Petals sans redémarrer un ou des nœuds.	Pas d'intérêt particulier.

Interroger l'annuaire de services de Petals

Avec l'application, il doit être possible de lister les services déployés dans l'infrastructure. Cet affichage peut se faire par nœud ou de manière globale. Il doit également être possible de rechercher un ou des services selon différents critères.

Pour chaque service, il faut donner son nom d'interface, nom de service, nom d'end-point et nœud de déploiement.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Lister ou rechercher des services.	Vérifier la présence d'un service donné en production.	S'approprier la notion de service et comprendre ce qui tourne dans Petals.

Lister l'ensemble des artefacts déployés sur un nœud Petals

Pour un nœud donné, l'application doit pouvoir lister les artefacts Petals déployés sur ce nœud. Pour chaque artefact, il faut donner son identifiant et son état (démarré, arrêté...).

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Découvrir rapidement ce qui tourne sur un nœud donné.	Voir ce qui est déployé sur un nœud donné.	Comprendre les artefacts Petals et les visualiser une fois déployés.

Visualiser les propriétés d'un nœud Petals

Pour un nœud donné, il faut pouvoir lister ses propriétés. Ces propriétés sont les suivantes :

- ◆ Propriétés du système (OS, JVM).

- ◆ Propriétés du nœud Petals, définies dans le **server.properties**.

Il s'agit de visualisation, donc pas de modification à chaud via l'application.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Récupérer les informations utiles sur un nœud, afin d'identifier les causes d'un problème, le tout, de manière centralisée.	Récupérer des informations sur un nœud, le tout, de manière centralisée. Intéressant pour communiquer des informations au support.	Découvrir les propriétés importantes de Petals.

Installer un artéfact Petals

L'application doit permettre d'installer un ou plusieurs artéfacts Petals sur un nœud donné. Dans le cas où plusieurs artéfacts sont déployés en bloc, c'est à l'application de déterminer l'ordre de déploiement (librairies partagées en premier, composants ensuite, service assemblies et service-units en dernier).

Le déploiement doit pouvoir se faire depuis un ou plusieurs fichiers en local. Ou bien il doit pouvoir se faire en précisant l'URL de l'artéfact à déployer (pas de déploiement en bloc).

En cas d'erreur sur une action groupée, l'utilisateur doit être interrogé pour savoir s'il faut continuer ou arrêter.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Installer en un clin d'œil des artéfacts Petals.	Idéalement, pas utilisé par l'exploitant (pas de déploiement à chaud en production). Mais peut être utilisé pour installer des artéfacts.	Pour débiter avec Petals et installer ses premiers artéfacts.

Désinstaller un artéfact Petals

La désinstallation d'artéfacts Petals doit se faire simplement, lorsque cet artéfact est à l'arrêt. Il doit également être possible d'arrêter un ensemble d'artéfacts, à condition qu'ils soient tous à l'arrêt.

En cas d'erreur sur une action groupée, l'utilisateur doit être interrogé pour savoir s'il faut continuer ou arrêter.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Désinstaller en un clin d'œil des artefacts Petals.	Désinstaller en un clin d'œil des artefacts Petals.	Désinstaller en un clin d'œil des artefacts Petals.

Modifier le cycle de vie d'un artefact Petals

L'application doit permettre de démarrer, arrêter ou passer en état « shutdown » n'importe quel artefacts Petals déployé sur un nœud. Il doit également être possible d'exécuter une même action sur un ensemble d'artefacts, à condition qu'ils soient tous du même type.

En cas d'erreur sur une action groupée, l'utilisateur doit être interrogé pour savoir s'il faut continuer ou arrêter.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Jouer sur le cycle de vie des artefacts (utile par exemple pour recharger des paramètres).	Jouer sur le cycle de vie des artefacts (utile par exemple pour recharger des paramètres).	Jouer sur le cycle de vie des artefacts.

Modifier à chaud des paramètres de Petals

Les composants Petals héritent, au travers du CDK, de certains paramètres configurables. Les paramètres suivants devraient être modifiables à chaud depuis l'application web d'administration.

- ◆ acceptor-pool-size
- ◆ processor-pool-size
- ◆ properties-file

De la même façon, il y a dans le conteneur des paramètres accessibles et utiles (transporteur, routeur). D'autres paramètres pourront être rajoutés plus tard (à terme, tous devraient être supportés par cette application).

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Utile pour <i>tuner</i> des	Adapter les taille des pools aux	Voir ce qui est paramétrable.

Intérêt pour les Utilisateurs		
composants Petals.	besoins de production. (ajustements). Modifier des propriétés du composant (nécessitera un redémarrage du composant).	

Modifier à chaud des niveaux de log

L'application doit permettre de modifier à chaud les niveaux de log des composants Petals.
Elle doit aussi permettre de modifier à chaud les niveaux de log pour des modules du conteneur.
Avant modification, il faut bien sûr pouvoir lire le niveau de log actuel.

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Extrêmement utile pour diagnostiquer des problèmes.	Diagnostiquer des problèmes, suivre les indications du support.	Vérifier la dynamique des traces dans Petals ESB.

Définir ses préférences

L'application doit permettre de définir des préférences : quelle topologie va-t-on attaquer, comment allons-nous la découvrir, quelles vont être les préférences quant à la pagination...

Intérêt pour les Utilisateurs		
Consultants	Exploitants	Novices
Confort d'utilisation pour la pagination. S'adapter à de nouvelles topologies très vite.	Passer facilement d'un environnement à un autre (topologie de tests, topologie de production)	Vérifier que Petals, ce n'est pas si compliqué. Une information à fournir et la topologie entière est administrable.

Tâches

Nous avons listé les besoins, c'est à dire, les fonctionnalités attendues par l'utilisateur mais vues sous le prisme « génie logiciel ». Nous allons maintenant traduire ces besoins en une suite de tâches, c'est à dire voir la réalisation de ces besoins sous le prisme « interface homme-machine ».

Rappels

Une tâche traduit un but utilisateur.

L'utilisation d'une fonctionnalité passe par l'exécution de tâches. Une tâche peut être composée de sous-tâches ou bien être élémentaire (pas de décomposition possible). Une composition de tâches peut se faire via une relation de dépendances ou bien via une relation temporelle.

Dans la pratique, une tâche commence par un verbe.

Ce verbe traduit une action utilisateur mais pas au niveau informatique : choisir, sélectionner, consulter, actionner...

Chaque tâche doit prévoir des scénarios : cas normal, cas d'exception, cas d'interruption.

On peut également préciser pour chaque tâche les concepts manipulés (quels objets intéressent l'utilisateur), les pré et postconditions, des contraintes (durée maximale par exemple), la fréquence de cette tâche, sa difficulté et sa criticité.

Les sections suivantes décrivent un ensemble de tâches.

Chaque tâche est rattachée à un ou plusieurs besoins. Ces associations tâches – besoins sont résumées dans un tableau à la fin de cette section.

L'identification des tâches est nécessaire pour définir une IHM abstraite.

Concepts

Les concepts sont les objets manipulés par l'utilisateur dans l'application.

Dans notre application, il y a 4 concepts essentiels :

- ◆ La **topologie** décrit un ensemble de nœuds Petals.
- ◆ Un **nœud Petals** est composé d'un conteneur et d'artéfacts.
- ◆ Le **conteneur** est le socle d'un nœud Petals.
- ◆ Les **artéfacts** sont des extensions ou des configurations pour Petals.

Les artéfacts se classent en 4 sous-concepts :

- ◆ Les **composants Petals** sont des bases de code configurables et réutilisables.
- ◆ Les **librairies partagées** sont des librairies communes au sein de l'ESB.
- ◆ Les **service assemblies** sont des artéfacts de déploiement pour Petals.
- ◆ Les **service-units** sont des configurations de composant (fournisseur ou consommateur).

Ces concepts ont différentes propriétés qui peuvent être modifiées au travers de l'application d'administration. On pourrait aussi rajouter les concepts de **service** et d'**annuaire de services**.

Il est important de remarquer qu'il y a une hiérarchie dans ces concepts.

La topologie est la racine de tout. C'est un ensemble de nœuds.

Au niveau en-dessous, on trouve le nœud Petals, qui est constitué d'un conteneur et d'artéfacts. Et ces artéfacts se décomposent, on l'a vu, en différentes catégories.

Cette hiérarchie doit être respectée dans l'application.

Elle doit fonctionner sur ce principe d'introspection, de zoom. Pour accéder à un niveau, on doit passer par celui du dessus. Cet aspect est fondamental pour la facilité d'apprentissage et d'utilisation de cette application, mais aussi pour appréhender Petals ESB dans sa globalité. En plus d'être utile, cet outil doit être pédagogique (pré-requis pour les utilisateurs de type *novices*).

C'est aussi avec cette approche à niveaux que nous allons classer les tâches.

La notation suivante est prise :

- *[tâche]* fait référence à une autre tâche.
- *Action* (sans crochets) fait référence à une action utilisateur.

Tâches liées à une topologie

Visualiser une Topologie

Identifiant : TOPO-1

Concepts associés : la topologie.

Préconditions : -

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean va sur la page d'accueil de l'application. Un ensemble de nœuds Petals apparaît sur la page.

Scénario d'erreur : Jean va sur la page d'accueil de l'application. Un message lui indique qu'aucune topologie n'a pu être trouvée. Il lui faut vérifier le point d'accès à la topologie.

Rechercher un nœud dans la Topologie

Identifiant : TOPO-2

Concepts associés : la topologie.

Préconditions : [Visualiser une Topologie]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [visualise la topologie]. Il entre ses critères de recherche et les nœuds de la topologie sont filtrés. Un sous-ensemble de nœuds Petals reste affiché. Ces nœuds correspondent aux critères entrés.

Scénario d'erreur : Jean [visualise la topologie]. Il entre ses critères de recherche et les nœuds de la topologie sont filtrés. Un message indique qu'aucun nœud correspondant à ces critères n'a

été trouvé.

Synchroniser les annuaires de services entre nœuds Petals

Identifiant : TOPO-3

Concepts associés : la topologie, annuaire de services.

Préconditions : [Visualiser une Topologie]

Postconditions : -

Contraintes : -

Complexité : -

Durée : plusieurs secondes à plusieurs minutes.

Scénario normal : Jean [visualise la topologie]. Il demande à synchroniser l'ensemble de la topologie. Jean est informé de la progression de l'opération. A la fin, un message lui indique que l'opération s'est bien passée. Le message indique également les nœuds qui n'ont pas pu être synchronisés car arrêtés.

Scénario normal alternatif : Robert [visualise la topologie]. Il s'est connecté à l'application en même temps que Jean. En arrivant sur la topologie, Robert est informé qu'une synchronisation des annuaires est en cours (lancée par Jean). A la fin, le résultat de la synchronisation lui est donné.

Scénario d'erreur : Jean [visualise la topologie]. Il demande à synchroniser l'ensemble de la topologie. Jean est informé de la progression de l'opération. Un message d'erreur lui est donné, indiquant qu'un nœud en état de marche n'a pas pu être synchronisé. Il lui est demandé s'il souhaite continuer ou bien arrêter l'opération de synchronisation.

Tâches liées à un nœud Petals

Sélectionner un nœud Petals

Identifiant : NOEUD-1

Concepts associés : topologie, nœud Petals.

Préconditions : [Visualiser une Topologie]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [visualise la topologie]. Il sélectionne le nœud qu'il souhaite administrer. Il arrive sur la page d'administration du nœud.

Scénario d'erreur : Jean [visualise la topologie]. Il sélectionne le nœud qu'il souhaite administrer. Il obtient un message d'erreur comme quoi l'application n'a pas pu établir de connexion avec le nœud.

Accéder au conteneur d'un nœud Petals

Identifiant : NOEUD-2

Concepts associés : nœud Petals.

Préconditions : [Sélectionner un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède à un nœud Petals]. Il choisit le conteneur dans la liste des éléments disponibles sur ce nœud.

Il n'y a pas de scénario d'erreur.

Accéder aux artefacts déployés sur un nœud Petals

Identifiant : NOEUD-3

Concepts associés : nœud Petals.

Préconditions : [Sélectionner un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède à un nœud Petals]. Il choisit les artefacts dans la liste des éléments disponibles sur ce nœud.

Il n'y a pas de scénario d'erreur.

Tâches liées au conteneur d'un nœud Petals

Visualiser les propriétés d'un nœud Petals

Identifiant : CONT-1

Concepts associés : conteneur.

Préconditions : [Accéder au conteneur d'un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède au conteneur d'un nœud Petals]. Il choisit de visualiser les propriétés de ce nœud. Les propriétés sont affichées sous la forme de clé-valeurs, en lecture seule.

Il n'y a pas de scénario d'erreur.

Modifier à chaud les niveaux de log (conteneur)

Identifiant : CONT-2

Concepts associés : conteneur.

Préconditions : [Accéder au conteneur d'un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède au conteneur d'un nœud Petals]. Il choisit de modifier des niveaux de log dans le conteneur. Une liste des loggers pré-identifiés lui est montrée, avec les niveaux de log qu'il peut choisir.

Il n'y a pas de scénario d'erreur pour ce cas.

Les bons noms de logger et les bons niveaux de log sont prédéfinis.

Scénario normal alternatif : Jean [accède au conteneur d'un nœud Petals]. Il choisit de modifier des niveaux de log dans le conteneur. Le logger qu'ils souhaite modifier n'apparaît pas dans la liste prédéfinie. Il entre le nom de ce logger et sélectionne le niveau de log à mettre. Un message lui indique que la modification s'est bien déroulée.

Scénario d'erreur alternatif : Jean [accède au conteneur d'un nœud Petals]. Il choisit de modifier des niveaux de log dans le conteneur. Le logger qu'ils souhaite modifier n'apparaît pas dans la liste prédéfinie. Il entre le nom de ce logger et sélectionne le niveau de log à mettre. Un message lui indique que le nom du logger entré est invalide et que l'opération a échoué.

Modifier à chaud des paramètres du conteneur

Identifiant : CONT-3

Concepts associés : conteneur.

Préconditions : [Accéder au conteneur d'un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède au conteneur d'un nœud Petals]. Il choisit de modifier des paramètres dans le conteneur. Une liste prédéfinie de ces paramètres, avec leur valeur actuelle, lui est montrée. Il modifie celui qu'il désire. Un message lui indique que la modification a bien été prise en compte.

Scénario d'erreur : Jean [accède au conteneur d'un nœud Petals]. Il choisit de modifier des paramètres dans le conteneur. Une liste prédéfinie de ces paramètres, avec leur valeur actuelle, lui est montrée. Il modifie celui qu'il désire mais met une valeur qui n'est pas valide. Jean ne peut pas enregistrer la modification, l'application ne pouvant soumettre une valeur incorrecte.

Lister les services

Identifiant : CONT-4

Concepts associés : conteneur, service, annuaire de service.

Préconditions : [Accéder au conteneur d'un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède au conteneur d'un nœud Petals]. Il choisit de lister les services visibles depuis ce nœud. Deux listes de services apparaissent, l'une pour les services déployés

sur ce nœud, et la deuxième pour les autres services de la topologie.

Il n'y a pas de scénario d'erreur.

Interroger l'annuaire de services du nœud

Identifiant : CONT-5

Concepts associés : conteneur, service, annuaire de service.

Préconditions : [Lister les services]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [liste les services]. Il entre des critères de filtrage. Seuls les services correspondant à ces critères sont affichés.

Scénario d'erreur : Jean [liste les services]. Il entre des critères de filtrage. Un message lui indique qu'aucun service ne correspond à ces critères.

Synchroniser l'annuaire de service du nœud

Identifiant : CONT-6

Concepts associés : conteneur, annuaire de service.

Préconditions : [Lister les services]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [liste les services]. Il active la synchronisation de l'annuaire de ce nœud. Un message lui indique que l'opération s'est bien déroulée.

Scénario normal alternatif : Jean [liste les services] sur un nœud maître. La synchronisation n'est pas disponible, il ne peut pas l'activer sur ce nœud.

Tâches liées aux artefacts d'un nœud Petals

Installer un artefact Petals

Identifiant : ARTI-1

Concepts associés : artefacts.

Préconditions : [Accéder aux artefacts déployés sur un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède aux artefacts d'un nœud]. Il sélectionne un ensemble d'artefacts à déployer sur son disque local. L'application installe tous ces artefacts dans le bon ordre. Un mes-

sage vient conclure l'opération, en annonçant que tout s'est bien déroulé.

Scénario d'erreur avec poursuite : Jean [accès aux artefacts d'un nœud]. Il sélectionne un ensemble d'artefacts à déployer sur son disque local. L'application commence à installer tous ces artefacts. Un message d'erreur lui indique qu'un artefact n'a pas pu être installé. Une question suit pour savoir s'il faut poursuivre l'opération avec les artefacts restants. Jean accepte, l'opération se termine. Un message résume le déroulement : tant d'artefacts déployés, et 1 échec avec poursuite.

Scénario d'erreur avec abandon : Jean [accède aux artefacts d'un nœud]. Il sélectionne un ensemble d'artefacts à déployer sur son disque local. L'application commence à installer tous ces artefacts. Un message d'erreur lui indique qu'un artefact n'a pas pu être installé. Une question suit pour savoir s'il faut poursuivre l'opération avec les artefacts restants. Jean refuse et préfère abandonner. L'opération se termine. Un message résume le déroulement : tant d'artefacts déployés, et 1 échec avec abandon, tant d'artefacts non-déployés.

Scénario d'erreur avec archive invalide : Jean [accède aux artefacts d'un nœud]. Il sélectionne un ensemble d'archives ZIP à déployer sur son disque local. L'application commence à installer tous ces artefacts. Un message d'erreur lui indique qu'une des archives n'est pas un artefact Petals. Une question suit pour savoir s'il faut poursuivre l'opération avec les artefacts restants. Jean accepte, l'opération se termine. Un message résume le déroulement : tant d'artefacts déployés, et 1 archive n'a pas été traitée.

Scénario normal alternatif : Jean [accède aux artefacts d'un nœud]. Il entre l'URL d'un artefact Petals localisé sur un serveur web. L'application installe l'artefact. Un message indique que l'opération s'est bien déroulée.

Scénario d'erreur alternatif : Jean [accède aux artefacts d'un nœud]. Il entre l'URL d'une archive ZIP localisée sur un serveur web. L'application installe l'artefact. Un message d'erreur indique que l'archive n'a pas pu être installée.

Lister les artefacts déployés sur un nœud Petals

Identifiant : ARTI-2

Concepts associés : artefacts.

Préconditions : [Accéder aux artefacts déployés sur un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède aux artefacts d'un nœud]. Il décide de gérer les artefacts déployés. Quatre listes d'artefacts apparaissent (une par type d'artefact). Jean visualise ainsi tous les artefacts déployés sur ce nœud.

Scénario normal alternatif : Jean [accède aux artefacts d'un nœud]. Il décide de gérer les artefacts déployés. Quatre listes d'artefacts apparaissent (une par type d'artefact). Un message lui indique qu'aucun artefact de ce type n'a été trouvé, et ce pour chaque liste.

Désinstaller un artefact Petals

Identifiant : ARTI-3

Concepts associés : artefacts.

Préconditions : [Lister les artefacts déployés sur un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [liste les artefacts d'un nœud]. Il décide de désinstaller l'un d'eux. Il sélectionne l'artefact à désinstaller et démarre l'opération. Un message lui indique que l'opération s'est bien déroulée.

Scénario d'erreur avec poursuite : Jean [liste les artefacts d'un nœud]. Il décide de désinstaller un service assembly. Il sélectionne l'artefact à désinstaller et démarre l'opération. Un message d'erreur lui indique que l'artefact doit d'abord être arrêté avant de pouvoir être désinstallé. Une question suit pour savoir s'il veut poursuivre la désinstallation et forcer l'arrêt. Jean accepte. Un message lui indique que la désinstallation s'est bien déroulée.

Scénario d'erreur avec abandon : Jean [liste les artefacts d'un nœud]. Il décide de désinstaller un service assembly. Il sélectionne l'artefact à désinstaller et démarre l'opération. Un message d'erreur lui indique que l'artefact doit d'abord être arrêté avant de pouvoir être désinstallé. Une question suit pour savoir s'il veut poursuivre la désinstallation et forcer l'arrêt. Jean refuse. Aucun message n'apparaît.

Scénario d'erreur avec composant : Jean [liste les artefacts d'un nœud]. Il décide de désinstaller un composant. Il sélectionne l'artefact à désinstaller et démarre l'opération. Un message d'erreur lui indique que le composant a encore des configurations de déployées et que l'opération est impossible.

Scénario normal alternatif : Jean [liste les artefacts d'un nœud]. Il décide de désinstaller plusieurs d'entre d'eux. Il sélectionne les artefacts à désinstaller et démarre l'opération. Un message lui indique que l'opération s'est bien déroulée.

Scénario d'erreur alternatif : Jean [liste les artefacts d'un nœud]. Il décide de désinstaller plusieurs d'entre d'eux. Il sélectionne les artefacts à désinstaller et démarre l'opération. Un message d'erreur lui indique que l'opération a rencontré un problème. Une question suit pour savoir s'il souhaite poursuivre l'opération ou l'annuler. Jean accepte. Un message résume le déroulement : tant d'artefacts désinstallés, et 1 en échec (avec poursuite).

Modifier le cycle de vie d'un artefact Petals

Identifiant : ARTI-4

Concepts associés : artefacts.

Préconditions : [Lister les artefacts déployés sur un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [liste les artefacts d'un nœud]. Il décide de modifier le cycle de vie de l'un d'eux. Il sélectionne l'artefact sur lequel opérer et démarre l'opération. Un message lui indique que l'opération s'est bien déroulée.

Scénario d'erreur avec composant : Jean [liste les artefacts d'un nœud]. Il décide d'arrêter l'un des composants. Il sélectionne l'artefact sur lequel opérer et démarre l'opération. Un message d'erreur lui indique que le composant a encore des configurations de déployées et que l'opération est impossible.

Scénario normal alternatif : Jean [liste les artefacts d'un nœud]. Il décide de modifier le cycle de vie de plusieurs d'entre d'eux. Il sélectionne les artefacts sur lesquels opérer et démarre l'opération. Un message lui indique que l'opération s'est bien déroulée.

Scénario d'erreur alternatif : Jean [liste les artefacts d'un nœud]. Il décide de modifier le cycle de vie de plusieurs d'entre d'eux. Il sélectionne les artefacts à désinstaller et démarre l'opération. Un message d'erreur lui indique que l'opération a rencontré un problème. Une question suit pour savoir s'il souhaite poursuivre l'opération ou l'annuler. Jean accepte. Un message résume le déroulement : tant d'artefacts modifiés, et 1 en échec (avec poursuite).

Modifier à chaud les niveaux de log (composants)

Identifiant : ARTI-5

Concepts associés : artefacts.

Préconditions : [Accéder aux artefacts déployés sur un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède aux artefacts d'un nœud]. Il décide de modifier les niveaux des loggers pour un composant précis. Il sélectionne son composant puis définit le nouveau niveau de log. Un message lui indique que l'opération s'est bien déroulée.

Il n'y a pas de scénario d'erreur.

Modifier à chaud des propriétés d'un composant Petals

Identifiant : ARTI-6

Concepts associés : artefacts.

Préconditions : [Accéder aux artefacts déployés sur un nœud Petals]

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean [accède aux artefacts d'un nœud Petals]. Il choisit de modifier la taille d'un pool dans un composant. Il sélectionne son composant. Une liste prédéfinie de ces paramètres, avec leur valeur actuelle, lui est montrée. Il modifie celui qu'il désire. Un message lui indique que la modification a bien été prise en compte.

Scénario d'erreur : Jean [accède aux artefacts d'un nœud Petals]. Il choisit de modifier la taille d'un pool dans un composant. Il sélectionne son composant. Une liste prédéfinie de ces paramètres, avec leur valeur actuelle, lui est montrée. Il modifie celui qu'il désire mais met une valeur qui n'est pas valide. Jean ne peut pas enregistrer la modification, l'application ne pouvant soumettre une valeur incorrecte.

Tâches liées aux préférences

Découvrir une topologie au travers d'un nœud Petals

Identifiant : PREF-1

Concepts associés : topologie.

Préconditions : -

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean souhaite administrer une topologie de nœuds Petals. Il entre les propriétés de connexion d'un nœud de cette topologie pour découvrir tous les autres nœuds et les administrer depuis l'application. Une fois les informations entrées, un message lui indique que la topologie a bien été récupérée.

Scénario d'erreur : Jean souhaite administrer une topologie de nœuds Petals. Il entre les propriétés de connexion d'un nœud de cette topologie pour découvrir tous les autres nœuds et les administrer depuis l'application. Une fois les informations entrées, un message d'erreur lui indique que le nœud n'a pas pu être contacté. Les paramètres de connexion sont peut-être invalides.

Découvrir une topologie au travers d'un fichier topology.xml

Identifiant : PREF-2

Concepts associés : topologie.

Préconditions : -

Postconditions : -

Contraintes : -

Complexité : -

Scénario normal : Jean souhaite administrer une topologie de nœuds Petals. Il entre l'URL d'un fichier *topology.xml*. Un message lui indique que la topologie a bien été récupérée.

Scénario d'erreur : Jean souhaite administrer une topologie de nœuds Petals. Il entre l'URL d'un fichier. Un message d'erreur lui indique que le fichier soumis n'est pas valide.

Changer le nombre d'éléments affichés par page (pagination)

Identifiant : PREF-3

Concepts associés : l'application elle-même, pas vraiment un concept.

Préconditions : -

Postconditions : -
Contraintes : -
Complexité : -

Scénario normal : Jean souhaite augmenter le nombre d'artéfacts présentés sur les pages de l'application. Il entre la nouvelle valeur pour la pagination. Un message d'erreur lui indique que la valeur a bien été prise en compte.

Scénario d'erreur : Jean souhaite augmenter le nombre d'artéfacts présentés sur les pages de l'application. Il entre la nouvelle valeur pour la pagination. Un message d'erreur lui indique que la valeur soumise est invalide.

Tableau Associatif entre cas d'usage et tâches

Cas d'Usage VS. Tâches	
Cas d'Usage	Tâches
Visualiser une topologie	TOPO-1, TOPO-2, PREF-1, PREF-2
Synchroniser les annuaires de services entre nœuds	TOPO-3, CONT-6
Interroger l'annuaire de services	CONT-4, CONT-5
Lister l'ensemble des artéfacts déployés sur un nœud	ARTI-2
Visualiser les propriétés d'un nœud	CONT-1
Installer un artéfact	ARTI-1
Désinstaller un artéfact	ARTI-3
Modifier le cycle de vie d'un artéfact	ARTI-4
Modifier à chaud des paramètres	CONT-3, ARTI-6
Modifier à chaud les niveaux de log	CONT-2, ARTI-5
Définir ses préférences	PREF-1, PREF-2, PREF-3

Les tâches NOEUD-1, NOEUD-2 et NOEUD-3 ne couvrent aucun cas d'usage.
Ce sont des tâches de transition, nécessaires pour passer d'une tâche à une autre.

Le découpage en tâches et leur description a permis de découper les cas d'usage et de préparer les actions utilisateurs nécessaires à leur réalisation. Il est maintenant possible de proposer des maquettes pour la partie graphique de l'application.

IHM Abstraite

Charte Graphique

Les éléments suivants constituent la charte graphique.

Cette charte se veut un complément à la charte graphique générale des sites et applications web de Petals Link. En cas de conflit entre la charte standard et celle ci-dessous, c'est cette dernière qui domine. En effet, elle a été définie avec les besoins des utilisateurs (Petals) en tête.

- ◆ L'application doit occuper toute la largeur de l'écran. Pas de largeur fixe.
- ◆ Le menu doit être accessible à tout instant, y compris quand la page devient très grande. Le menu sera donc fixe, sur la gauche. Le reste du site servira à présenter du contenu.
- ◆ Le menu sera surmonté d'une image marketing (donc sexy) de Petals ESB.
- ◆ La partie droite, avec le contenu du site, contiendra dans le pied de page, un copyright au nom de EBM WebSourcing, une mention de la marque commerciale Petals Link, ainsi qu'un lien vers son site. Ce pied de page ne débordera pas sur la partie gauche du site (le menu).
- ◆ Vue la complexité des structures à administrer dans Petals, tout ne peut pas apparaître dans le menu. Lorsque l'on navigue dans la page d'un nœud (artéfacts, loggers, etc), la partie supérieure principale présentera un fil d'Ariane, sous la forme

{nom du conteneur} > {page}

- ◆ Les couleurs acceptées sont le violet, le orange et le blanc.
- ◆ Les menus ont tous un arrière-plan violet, des bords arrondis. Leurs textes sont en blanc.
- ◆ Le menu actuel (localisation) a son fond en blanc.
- ◆ Le menu survolé a son fond en orange.

Plus globalement, la logique du site doit être la suivante :

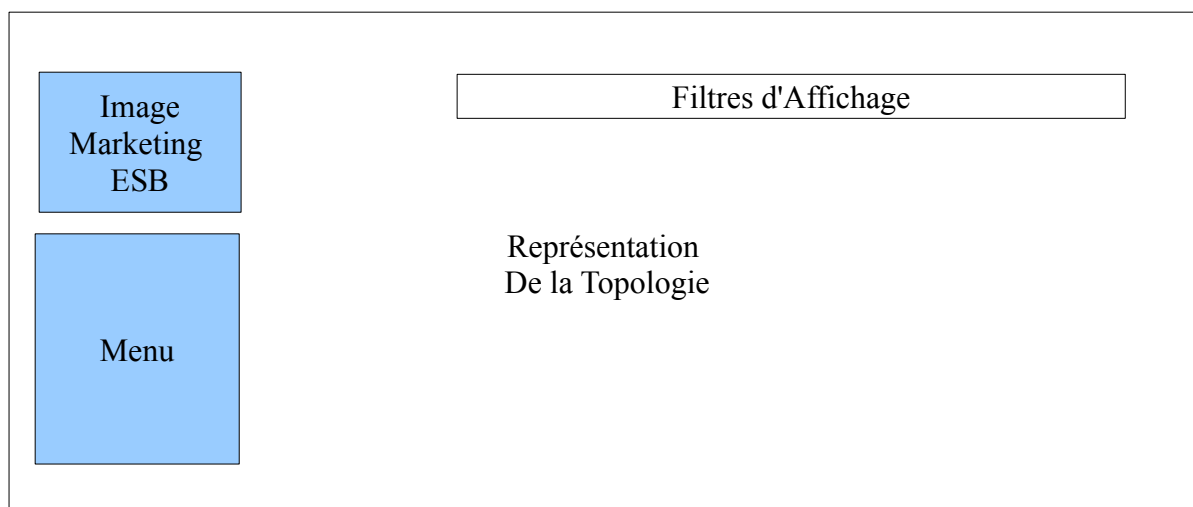
- ◆ On part de la topologie, toujours. C'est la page d'accueil du site.
- ◆ On descend d'un niveau pour arriver aux nœuds.
- ◆ On descend encore d'un niveau pour arriver au conteneur ou aux artéfacts.

Dans cette logique, le menu aide à se situer et à remonter d'un ou plusieurs niveaux.

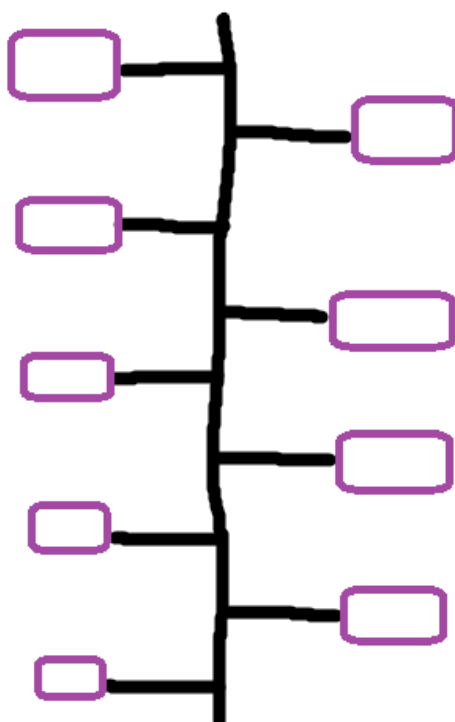
Pour redescendre, on ne passe pas par le menu mais par le contenu du site. Ainsi, le menu aide à se repérer, et en ne montrant que les actions possibles, oriente l'utilisateur vers les bonnes options. De plus, cette approche coïncide (volontairement) avec la logique de fonctionnement et de distribution de Petals.

Maquettes – Topologie

C'est la page d'accueil ou racine du site.
Elle est composée de 2 parties.



Le menu ne contient que 2 items : **Topologie** et **Préférences**.
La topologie est représentée grâce à un assemblage d'images.



Cet assemblage alterne entre un nœud à gauche, puis un nœud à droite.
Il se décompose en 6 images élémentaires :

- ◆ Les extrémités inférieures et supérieures de la topologie (pas de nœud).
- ◆ Une image pour placer un nœud et son raccord à gauche de l'axe central (nœud violet).
- ◆ Une image pour placer un nœud et son raccord à droite de l'axe central (nœud violet).
- ◆ Une variante des 2 images précédentes avec les nœuds en gris clair (pour indiquer un

nœud à l'arrêt).

Le nom des nœuds apparaît sur les images.

Le texte est écrit par l'application (et doit être blanc). Des informations additionnelles sont disponibles si l'on survole un nœud. Un popup / tool tip élaboré présente ces informations que l'on ne peut pas montrer d'un seul tenant.

Un clic sur un nœud emmène l'utilisateur vers les propriétés du nœud.

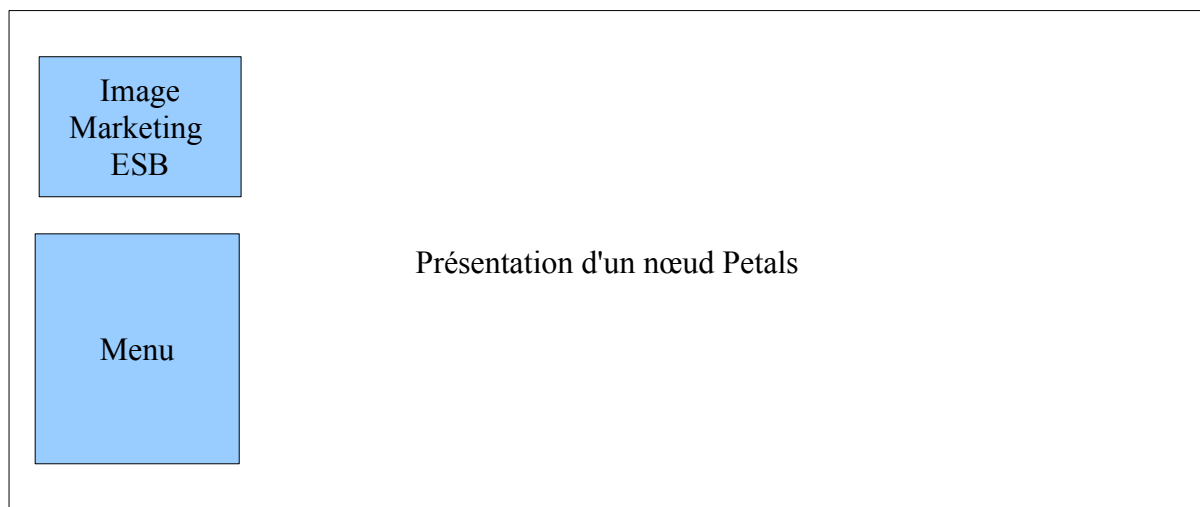
Maquettes – Nœud Petals

Un nœud Petals n'a pas de propriétés en soi.

Il n'est qu'un intermédiaire pour accéder au conteneur ou aux artéfacts.

Cependant, cette page a un intérêt pédagogique, pour expliquer qu'un nœud est constitué d'un conteneur et qu'on y déploie des artéfacts dessus.

C'est un menu élaboré.



Le menu (à gauche) est composé de la manière suivante.

- **Topologie**
- **Préférences**
- **A Propos**

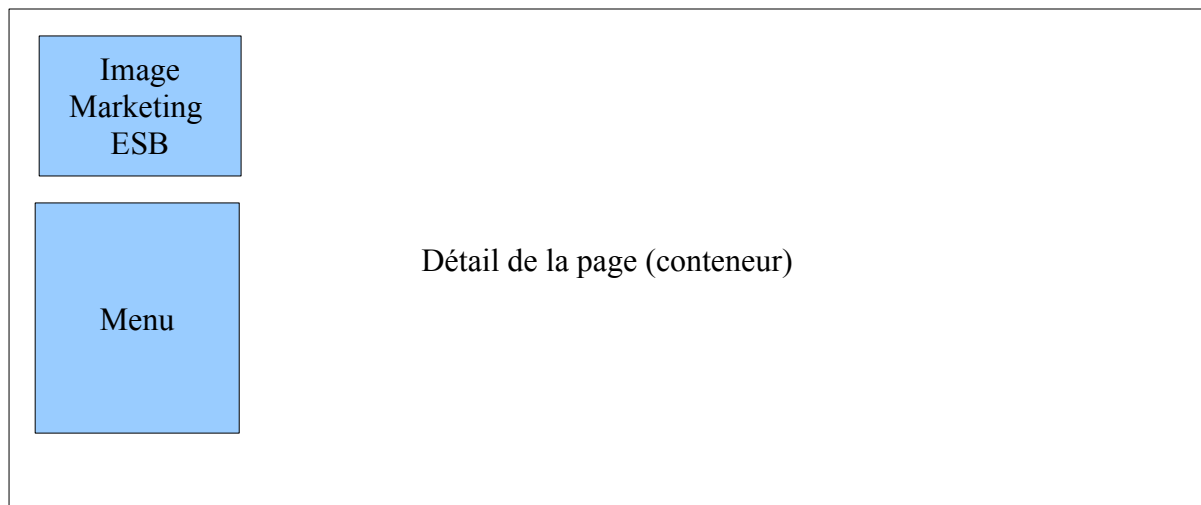
Les fils d'Ariane sont les suivants :

- *{ nom du Nœud }* > Annuaire de Services
- *{ nom du Nœud }* > Loggers
- *{ nom du Nœud }* > Paramètres

- { nom du Noeud } > Composants

Maquettes – Conteneur

Les pages liées au conteneur se présentent comme des formulaires assez classiques (paramétrage, loggers), ou comme des tableaux clé-valeurs (propriétés) ou bien encore comme une suite arborescente (annuaire : arbre interface > service > end-points).

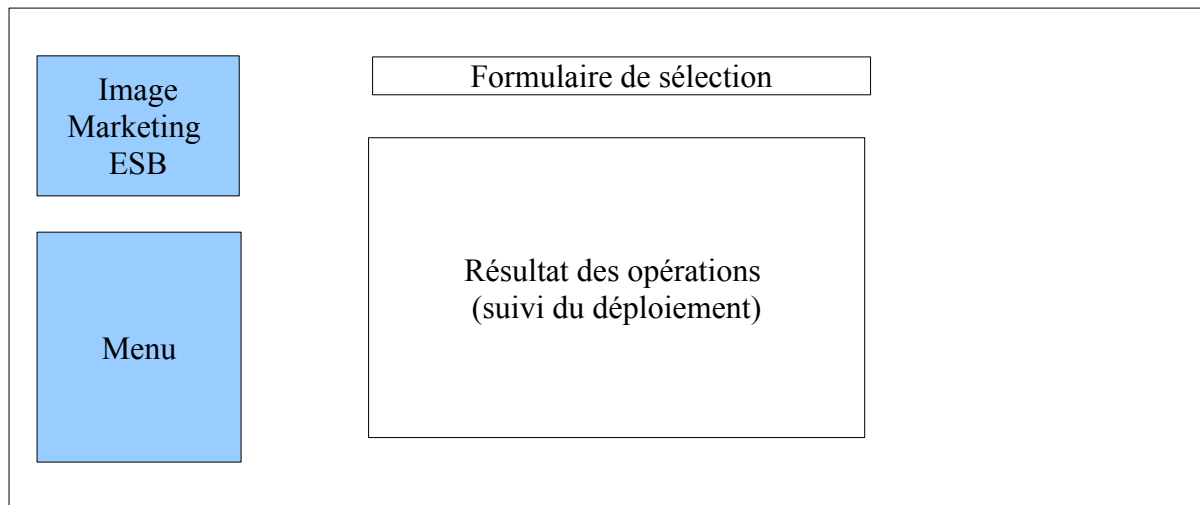


Le menu est identique à celui d'un nœud, à une exception près : le sous-menu conteneur est déplié et la page choisie est activée (fond blanc) dans le sous-menu.

Maquettes – Artéfacts

La partie **paramètres** se présente comme celle du conteneur (formulaire classique).

La partie **installation** se présente comme suit :



La partie cycle de vie reprend ce qui avait été adopté dans les anciennes applications web d'administration pour Petals ESB.

Et voici un aperçu du formulaire dans la console web 2.0.
L'idée est de faire quelque chose de similaire.



Type	Name	State		Modify Attributes	Start	Stop	Shutdown	Uninstall	Force unload
		All							
<input type="checkbox"/>	bc	petals-bc-soap	Started						
<input type="checkbox"/>	se	petals-sample-clock	Started						
<input type="checkbox"/>	se	petals-sample-helloworld	Started						

Maquettes – Préférences

La page des préférences se présente comme un formulaire classique.
Le menu affiché ne comporte que les éléments **topologie** et **préférences**.

Contraintes Techniques et Implémentation

Contraintes Techniques

- ◆ L'application doit produire des pages HTML valides (vérification W3C).
- ◆ Les feuilles de style CSS doivent être valides (vérification W3C, CSS 3 accepté, solutions propriétaires acceptées uniquement pour arrondir les bords).
- ◆ Interdiction d'utiliser des images pour faire des cadres ou des sections aux angles arrondis.
- ◆ L'application pourra utiliser du Javascript. JQuery est la librairie imposée pour Javascript.
- ◆ L'application doit être internationalisée (français et anglais).
- ◆ L'application doit tourner sous Firefox, Chrome, IE 8 et Safari. Le support de terminaux mobiles n'est pas à prévoir.
- ◆ Les interactions entre Petals et l'application se feront au travers de JMX.
- ◆ L'application doit être codée en Java.
- ◆ L'application ne nécessite pas de base de données.
- ◆ L'application doit pouvoir être ajoutée à un serveur d'intégration continue (Jenkins).
- ◆ L'application doit pouvoir être testée de manière automatique (Selenium ou équivalent).
- ◆ L'application doit être déployable sur un serveur Tomcat.
- ◆ Le packaging de l'application doit être au format WAR.
- ◆ Le poids de l'application ne doit pas excéder 15 Mo.

Sur ce dernier point, une taille supérieure justifierait des questions sur la pertinence de la technologie d'implémentation choisie et sur son fonctionnement (une solution pure JSP + Servlet ferait moins de 2 Mo, ressources incluses).

- ◆ La notion de session d'utilisation n'est pas à supporter.

On considère que les préférences sont globales et donc communes à tous les utilisateurs. Ou du moins, on considère que cette application sera utilisée par un nombre restreint de personnes sur une topologie donnée.

- ◆ Le support de la haute disponibilité n'a pas à être garanti.

Il n'est pas attendu que l'application puisse tourner sur un cluster de serveurs d'applications. En revanche, on tentera d'utiliser les solutions les plus simples et les plus courantes possibles pour réduire les développements à mettre en œuvre si ce besoin venait à venir.

Ce point est en partie lié à la tâche « synchronisation des annuaires de services de la topologie », qui est une opération longue. On ne veut pas que 2 utilisateurs puissent lancer cette même opération en parallèle. Au contraire, il faut que le 2ème utilisateur soit notifié de ce qu'a fait le premier sur ce cas d'usage précis. C'est pour le moment le seul élément identifié qui pourrait poser pro-

blème pour la haute disponibilité de l'application. Une solution pourrait être d'avoir une préférence cachée pour cet état. Il faut noter aussi que si 2 utilisateurs venaient à faire cette opération en même temps, cela ne serait pas critique pour Petals.

Gestion des Erreurs

Les scénarios mis en avant dans les tâches comportent des cas d'erreur. Ces cas correspondent à des erreurs fonctionnelles. En revanche, ces scénarios n'intègrent pas la gestion des erreurs techniques (telle qu'une coupure réseau ou la chute d'un nœud alors qu'une opération d'administration est en cours).

L'apparition d'une stack d'erreur Java dans une page web est inacceptable. Les erreurs techniques doivent toutes être récupérées par l'application.

Un message intelligible doit être adressé à l'utilisateur pour lui signifier qu'une erreur technique est survenue. On tentera, dans la mesure du possible, de fournir une explication simple sur la nature de cette erreur. Il sera proposé à l'utilisateur de rafraîchir la page d'ici quelques instants. Ce rafraîchissement peut même être effectué automatiquement par l'application ou le navigateur web.

Une trace plus avancée, pour le support, devra être laissée dans les logs de l'application.

Gestion des Préférences

Les préférences font de facto partie de l'application et améliorent son confort d'utilisation. Sont stockés dans les préférences :

- ◆ Les propriétés de connexion à un nœud de la topologie.
Ce nœud est désigné sous le nom de « point d'entrée à la topologie ».
- ◆ Une URL pointant vers un fichier de topologie (URL locale ou pointant sur un serveur web).
Ceci est une alternative à la fourniture d'un nœud comme point d'entrée.
- ◆ Le nombre d'éléments à afficher par page pour les listes d'artéfacts.
- ◆ Le nombre d'éléments à afficher par page pour les listes de services (annuaires).

Des préférences cachées peuvent aussi être ajoutées.

Les préférences citées plus haut sont celles que l'utilisateur peut modifier via l'application.

Etant donné que les préférences sont globales pour tous les utilisateurs, il est suggéré aux équipes de développement de s'appuyer sur l'API des préférences du JDK, pour stocker les préférences de l'application (ni base de données, ni fichier sauvegardé explicitement par l'application).