
Mémoire de Projet de Fin d'Études

Pour l'Obtention du Titre

D'Ingénieur d'État en Informatique

Sujet

Migration de CMS Cockpit vers nouvelle génération
SmartEdit

Soutenu par :

Yasser EL MARZAK

Sous la direction de :

M. BALOUKI Youssef

M. ER RADY Adil

M. HAMID Rafik

Année Universitaire 2019-2020

Dédicace

À ma chère mère et à mon cher père qui n'ont jamais cessé de me supporter, me soutenir et m'encourager durant mes années d'études

A mes frères, le meilleur symbole de l'amitié, l'amour et la bonté. Je vous suis à jamais reconnaissant pour tout ce que vous faites pour moi. Et a tous les compagnons que j'ai eu l'honneur de rencontrer durant les années de mon existence, et qui m'ont profondément inspirés sur plusieurs facettes de la vie.

Aux professeurs qui m'ont accompagné durant mon parcours scolaire, cet exercice qui a été laborieux mais ô combien important pour fonder la persévérance et la détermination qui m'ont été inculqués, et qui je l'espère, resteront gravés à tout jamais en moi.

MERCI

Remerciements

Je tiens à présenter mes très sincères remerciements à Allah le tout puissant, qui a éclaircit ma voie par le savoir, et qui m'a armé de la bonne foi.

Je tiens à remercier tout particulièrement et à témoigner toute ma reconnaissance à M. ER RADY Adil et à M. HAMID Rafik, pour l'expérience enrichissante et pleine d'intérêt qu'ils m'ont fait vivre durant la période du stage, pour tous les conseils et les informations qu'ils m'ont prodigués et pour le temps qu'ils ont consacré à l'encadrement et le suivi de ce travail.

Je tiens à remercier également mon professeur encadrant M. BALOUKI Youssef, en dépit de ses multiples charges, son aide et les renseignements précieux qu'il m'a fourni ainsi que pour tous les conseils et les informations qu'il m'a prodigué avec un degré de patience et de professionnalisme sans égal.

Je tiens aussi à adresser mes plus sincères remerciements à l'ensemble du corps administratif et enseignant de l'Académie International de L'aviation Civile, pour avoir porté un vif intérêt à notre formation, et pour avoir accordé de l'attention et de l'énergie, et ce, dans un cadre agréable de respect.

Que les membres de jury trouvent, ici, l'expression de mes remerciements pour l'honneur qu'ils me font en prenant le temps de lire et d'évaluer ce travail.

Résumé

Le présent document synthétise notre travail effectué au sein de SQLI-Rabat au titre du projet de fin d'études, qui s'intitule La Migration de CMS Cockpit vers nouvelle génération SmartEdit, pour l'obtention du diplôme d'ingénieur d'Etat en informatique.

Ce projet vise à permettre à la société cliente, de gérer le contenu de ses sites web de manière plus flexible en plus du cockpit WCMS. Les utilisateurs professionnels peuvent désormais bénéficier d'une manière intuitive et conviviale pour créer et gérer des pages et des composants, L'idée est que SmartEdit apporte un cockpit WYSIWYG dans lequel les gestionnaires de contenu (ou tout autre type d'utilisateurs finaux) peuvent se passer du support informatique pour gérer les pages de leurs sites Web SAP Hybris Commerce.

L'étude technique a été faite conjointement entre SQLI et la Maîtrise d'Ouvrage, vu l'expertise de SQLI dans les outils utilisés qui ont prouvé une efficacité inégale sur d'autres projets similaires. Durant ce stage, nous avons opté pour la méthodologie SCRUM pour la gestion et la conduite du projet.

Abstract

This document summarizes our work carried out within SQLI-Rabat as part of the end of studies project, which is entitled The Migration of CMS Cockpit to new generation SmartEdit, for obtaining the diploma of state engineer in computer science.

This project aims to allow the client company to manage the content of its websites in a more flexible way in addition to the WCMS cockpit. Professional users can now benefit from an intuitive and user-friendly way to create and manage pages and components. The idea is that SmartEdit provides a WYSIWYG cockpit in which content managers (or any other type of end user) can do without IT support to manage the pages of their SAP Hybris Commerce websites.

The technical study was done jointly between SQLI and the Contracting Authority, given SQLI's expertise in the tools used which have proven uneven in effectiveness on other similar projects. During this internship, we opted for the SCRUM methodology for the management and conduct of the project.

Liste des abréviations

AJAX:	Asynchronous JavaScript and XML
API:	Application Programming Interface
B2B:	Business To Buisiness
B2C:	Buisiness To Constmer
cms:	Content Management Solution
CSS:	Cascading Style Sheets
HAC:	Hybris Administration Console
HMC:	Hybris Management Console
HTML:	HyperText Markup Language
ISC:	Innovation Skill Center
JEE:	Java Entreprise Edition
OOTB :	Out Of The Box
PME :	Petites et moyennes entreprises
REST :	Representational State Transfer
SVG:	Scalable Vector Graphics
WCMS:	Web Content Management Solution
WSIWYG:	What You See What You Get

Table des figures

Figure 1 : Logo SQLI	14
Figure 2 : SQLI-Partenaires	15
Figure 3 : SQLI-Organigramme	16
Figure 4 : SQLI-Missions.....	17
Figure 5 : SCRUM	24
Figure 6 : WCMS	27
Figure 7 : Diagramme des dépendances.....	28
Figure 8 : Appel à cmswebservices.....	32
Figure 9 : Gestion de medias.....	36
Figure 10 : SmartEdit	38
Figure 11 : point d’inflexion	40
Figure 12 : modifier une page	40
Figure 13 : modifier un composant	41
Figure 14 : Ajouter un composant.....	41
Figure 15 : Ajouter une nouvelle page	42
Figure 16 : Template de la page	43
Figure 17 : Gestion de la navigation	43
Figure 18 : personnalise votre site.....	44
Figure 19 : Cible la personnalisation.....	45
Figure 20 : Diagramme de dépendance de SmartEdit.....	46
Figure 21 : Structure de l’extension smarteditartil.....	53
Figure 22 : Configuration Editor.....	54
Figure 23 : Liaison de l’extension à travers Configuration Editor.....	54
Figure 24 : Liaison Via Impex	55
Figure 26 : page personnalise.....	58
Figure 25 : TrailPage.....	58
Figure 27 : toolbar personnalise	59
Figure 28 : toolbar Action	60
Figure 29 : Decorator personnalise	63
Figure 30 : bouton personnalise au menu contextuel	65
Figure 31 : bouton contextuel action.....	65
Figure 32 : items.xml	66
Figure 33 : TrainingComponent.jsp	67
Figure 34 : nouveau Composant personnalise dans SmartEdit	68
Figure 35 : Remplir le composant	68
Figure 36 : L’affichage de ce nouveau Composant.....	69
Figure 37 : Evolution du chiffre d’affaire de l’E-commerce mondial	74
Figure 38 : Vue globale sur l’architecture d’Hybris	78
Figure 39 : SmartEdit – Hybris	78
Figure 40 : HAC -Hybris.....	79

Liste des Tableaux

Tableau 1 : Plan de formation	19
Tableau 2 : Création de sites web et catalogues.....	33
Tableau 3 : Gestion des pages	34
Tableau 4 : Composant et emplacements de page.....	35
Tableau 5 : Gestion de navigation.....	36

Table des matières

Introduction générale.....	12
1 Contexte générale du projet.....	14
1.1 Présentation de l'organisme d'accueil	14
1.1.1 Présentation de SQLI	14
1.1.2 Organigramme.....	15
1.1.3 Métiers SQLI.....	17
1.2 Plan d'intégration	18
1.2.1 Compétition E-Challenge	18
1.2.2 Objectifs	18
1.2.3 Contenu de la formation	19
1.3 Contexte du projet	22
1.3.1 Présentation du client	22
1.3.2 Problématique.....	23
1.4 Conduite du projet	23
1.4.1 Introduction	23
1.4.2 Méthodologie de travail : SCRUM	24
1.5 Conclusion	24
2 Etude fonctionnel et technique.....	26
2.1 Fonctionnalités du module WCMS	26
2.1.1 Travailler avec le WCMS Cockpit	26
2.2 Architecture du module WCMS	28
2.2.1 Recipes	29
2.2.2 Extensions	29
2.3 Comparaison entre SmartEdit et CMS Cockpit.....	30
2.3.1 Technologie	31
2.3.2 Fonctionnalités	33
3 Nouvelle génération SmartEdit	38
3.1 Fonctionnalités.....	38
3.2 Smart Edit Architecture	46
3.3 Conclusion	48

4	Personnalisation de SmartEdit	50
4.1	Technologies.....	50
4.1.1	Hybris	50
4.1.2	Spring	50
4.1.3	AngularJs.....	50
4.1.4	Apache Ant.....	51
4.1.5	Git.....	51
4.2	Personnalisation.....	52
4.2.1	Création d'une extension SmartEdit personnalisée.....	52
4.2.2	Lier l'extension avec SmartEdit	53
4.2.3	Ajout d'une nouvelle page à SmartEdit	55
4.2.4	Ajout d'une barre d'outils à SmartEdit	58
4.2.5	Ajout de fonctionnalités et de modes	60
4.2.5.1	Ajout d'un mode	60
4.2.5.2	Ajout d'un décorateur	61
4.2.5.3	Ajout d'un bouton au menu contextuel	64
4.2.6	Création d'un type de composant personnalisé	65
4.3	Conclusion	69
	Conclusion générale	70
	Bibliographie	71
	Annexes	
	Le commerce électronique	73
	SAP Hybris.....	77

Introduction générale

Dans un marché de plus en plus concurrentiel, les besoins des entreprises ne cessent d'évoluer. Par conséquent, les systèmes d'information doivent également suivre le mouvement afin de s'adapter rapidement aux attentes des clients et aux nouvelles réglementations qui deviennent de plus en plus complexes et contraignantes.

Depuis de nombreuses années, nous avons été confrontés au défi de savoir comment rendre un CMS fonctionnel et confortable, comment trouver le bon équilibre entre fonctionnalité et simplicité et comment trouver une solution extrêmement flexible et adaptable aux besoins d'une entreprise. C'est un sujet complexe.

SAP Commerce SmartEdit permet aux gestionnaires de contenu de créer et de gérer facilement le contenu de leur site Web à la volée dans différents points d'inflexion et de le mettre à la disposition de leurs clients en un seul clic.

La personnalisation (basée sur SmartEdit) offre un moyen intégré et convivial de créer des expériences pertinentes pour vos clients, ce qui est essentiel pour stimuler l'engagement et la conversion. La personnalisation comprend des modules et des extensions qui facilitent l'utilisation du mode de personnalisation dans SmartEdit pour créer une expérience utilisateur entièrement personnalisée. Les capacités de personnalisation fonctionnent à la fois sur le contenu et le commerce de manière intégrée, afin que vous puissiez créer des expériences client de manière cohérente sur tous les canaux et fonctionnalités. En utilisant SmartEdit pour l'expérience client, vous pouvez voir l'expérience du client final au fur et à mesure que vous la construisez, tout en travaillant dans un ensemble d'outils.

Chapitre 1

Contexte général du projet

Ce chapitre a pour but de situer le projet dans son environnement organisationnel et contextuel. Il présente dans sa première partie l'organisme d'accueil, tandis que sa deuxième partie décrit les objectifs ainsi que les exigences pour la migration vers SmartEdit, La dernière partie est réservée à la démarche et la conduite adoptée pour la réalisation du projet.

1 Contexte générale du projet

1.1 Présentation de l'organisme d'accueil

L'objectif ici, est de présenter le groupe SQLI, de décrire ses secteurs d'activités, ainsi que ses chiffres clés.

1.1.1 Présentation de SQLI



Figure 1 : Logo SQLI

SQLI est une société française créée en 1990, pour accompagner les entreprises dans l'utilisation des nouvelles technologies.

Elle s'est spécialisée dans la réalisation des systèmes d'informations de la nouvelle génération. Elle est organisée en agences de proximité, afin de conserver le maximum de réactivité face aux besoins de ses clients. En

prenant en compte, au sein même de son organisation, les préoccupations du tissu économique régional, SQLI offre une approche sur mesure aux enjeux spécifiques des entreprises.

SQLI comporte plus de 2000 collaborateurs répartis dans les 17 agences dont 10 en France (Paris, Lyon, Toulouse, Bordeaux, Rouen, Nantes et Lille), en Suisse (Lausanne et Genève), au Luxembourg, en Belgique (Bruxelles), aux Pays-Bas et au Maroc (Rabat et Oujda).

SQLI compte plus de 1200 clients, grands comptes et PME, issus de tous les secteurs d'activité.

- L'industrie
- Les services
- Les banques et assurances
- Les administrations et services publics
- La distribution
- L'immobilier
- Les transports

- Les télécoms

Son positionnement unique au confluent du marketing et de la technologie lui permet de répondre de façon globale aux enjeux de développement des ventes et de notoriété (marketing digital et social, expérience client, commerce connecté, data intelligence. . .) ainsi qu'aux enjeux de productivité et d'efficacité interne (digitalisation des opérations, entreprise collaborative, mobilité et objets connectés, CRM. . .). S'appuyer sur les meilleures solutions. SQLI allie sa connaissance des métiers et son expertise de mise en œuvre des innovations technologiques de leurs partenaires. Ceci la permet de délivrer des solutions pertinentes et offrant des bénéfices tangibles.



Figure 2 : SQLI-Partenaires

1.1.2 Organigramme

Les différents centres de Rabat et Oujda étaient indépendants mais aujourd'hui avec la nouvelle organisation les deux sites ont été fusionnés en donnant naissance à une seule entité (Figure1.3).

Le centre de services e-commerce est composé d'un ensemble de projets pour le compte des sociétés multinationales.

Eric Chanal assure la Direction globale de l'ISC Maroc en s'appuyant sur Fouad pour les livraisons et Renaud Ferly pour la détection des futurs projets et la bonne coordination des avant-ventes. Les deux sites passent sous la responsabilité de Fouad Selmouni. Fouad aura un focus Delivery global afin de s'assurer que les produits sont de haute qualité pour les clients avec des moyens de production adéquats.

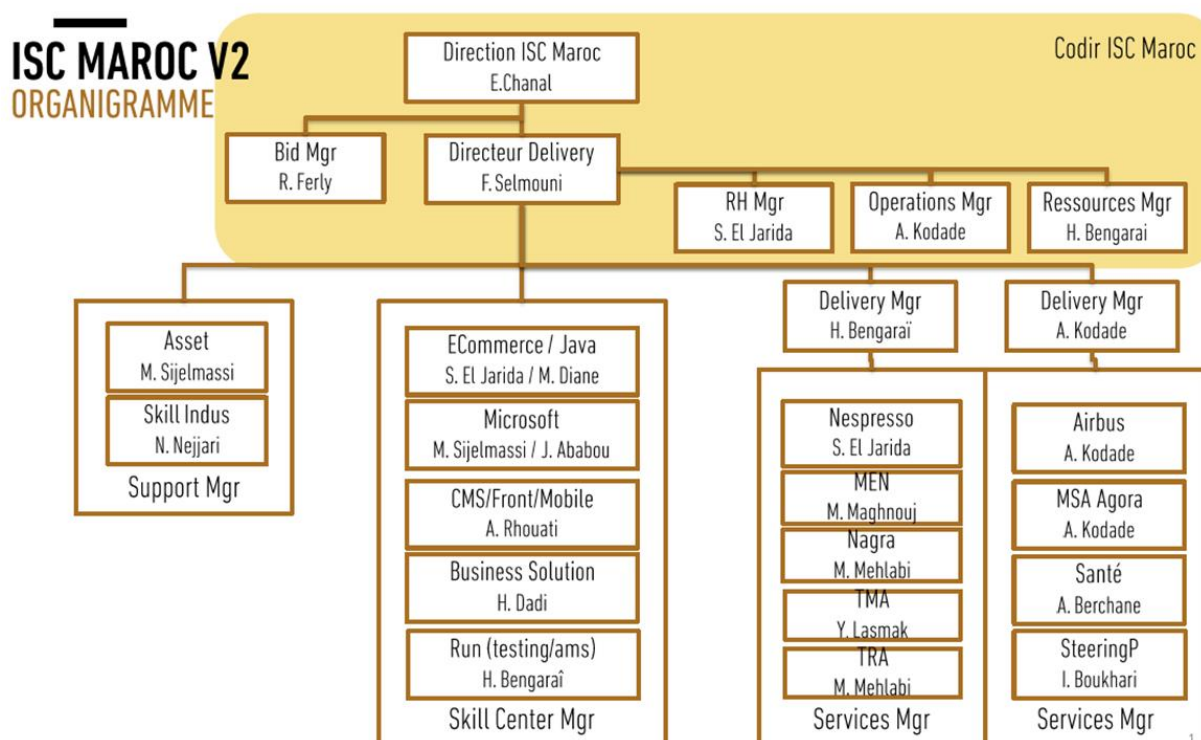


Figure 3 : SQLI-Organigramme

SQLI est composé de plusieurs Business Unit :

- **E-commerce (Java EE)** : cette Business Unit s'occupe de la mise en place des sites e-commerce et les applications Java EE. C'est dans cette business unit que j'ai passé mon stage.
- **Mobile/Front** : cette Business Unit s'occupe de la réalisation des applications mobile ainsi que le Front web, les technologies nouvelles à base de JavaScript tel que Node.js ou AngularJs...
- **Microsoft** : cette Business unit s'occupe de la réalisation des applications à base de technologie Microsoft.
- **Agency** : cette Business Unit fait un travail transverse elle assure le front web pour toutes les autres Business Unit. Elle s'occupe du découpage HTML/CSS.
- **TMA** : cette Business Unit s'occupe de la maintenance des applications clients. Elle assure une gestion des anomalies afin de garantir la continuité des applications
- **Delivery** : Ce business unit s'occupe des livraisons et recettes vis-à-vis des clients. Il est aussi de son scope de faire la gestion de configuration et la maintenance réseaux et infrastructure

1.1.3 Métiers SQLI

SQLI s'est spécialisée dans les projets e-commerce, liés aux systèmes d'informations intégrant l'utilisation des technologies internet. Pour aider les entreprises à tirer parti des technologies web, SQLI propose un accompagnement global sur tout le cycle du projet :

- Des prestations de conseil pour aider les clients à faire les bons choix.
- La mise en œuvre concrète de ces choix par la réalisation et l'intégration.
- Un accompagnement dans le déploiement des projets et le transfert de compétences.

Le groupe SQLI fédère toutes les compétences indispensables au bon déroulement des projets de ses clients, du conseil à la réalisation en passant par l'ergonomie, le design, l'interface utilisateur et la formation.

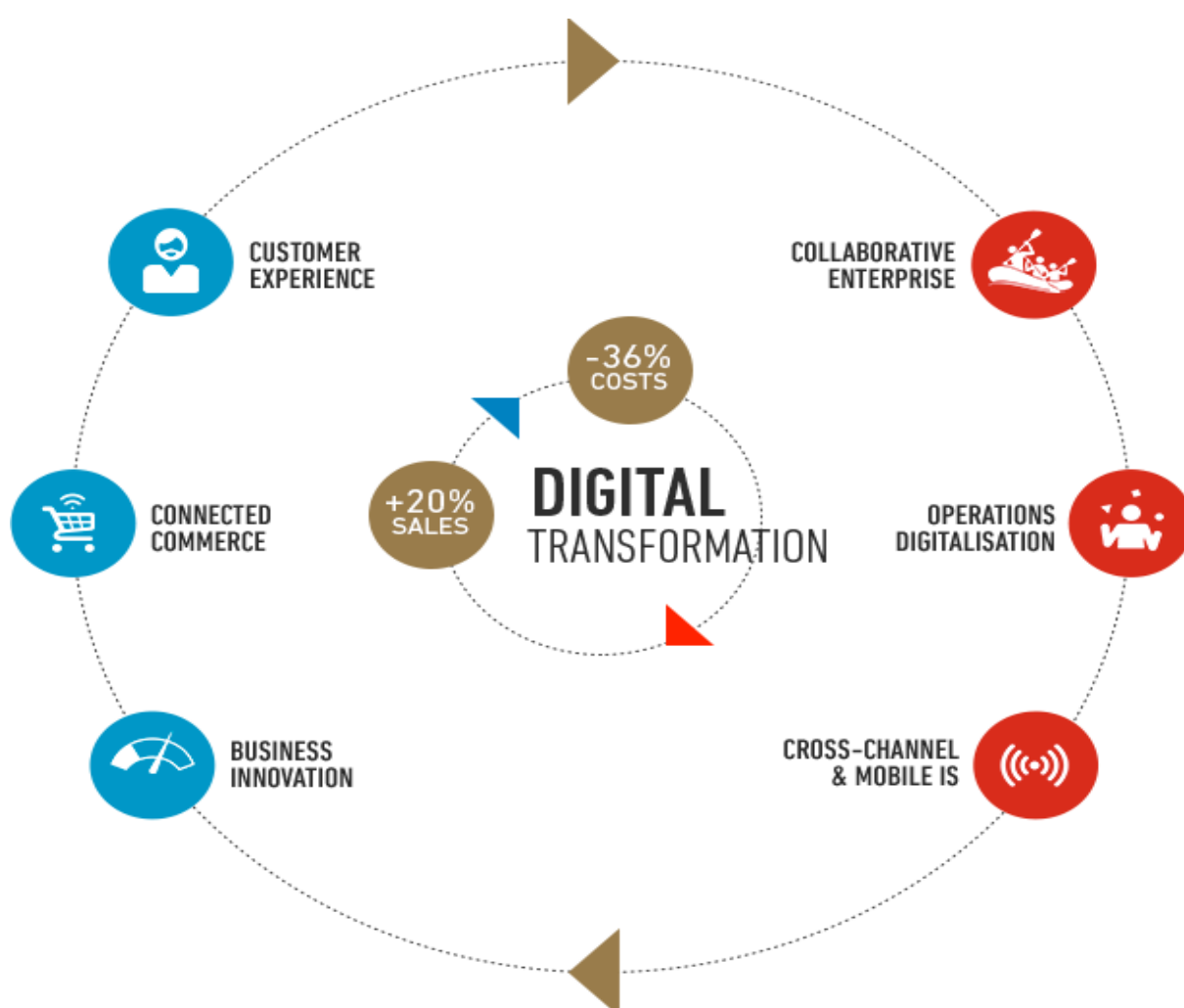


Figure 4 : SQLI-Missions

1.2 Plan d'intégration

1.2.1 Compétition E-Challenge

Depuis 2006 chaque année SQLI lance une édition de son concours national e-challenge visant à :

- Dresser une passerelle entre les écoles et SQLI
- Créer un espace d'échange mutuel
- Confronter les futurs lauréats à un contexte issu de la réalité du marché
- Proposer aux meilleurs candidats des stages de fin d'études pré-embauche
- Donner une chance de gagner des prix pour les finalistes

Le concours est une compétition de programmation qui regroupe des jeunes développeurs de toutes les écoles marocaines, le plan est divisé comme suit :

- **Présélection** : Sur la base d'un test technique à distance sur la plateforme de recrutement international Hacker Rank une présélection est faite pour choisir les candidats qui vont passer à la deuxième étape.
- **Évaluation technique** : Un autre test technique aux locaux de l'entreprise est organisé avec la participation d'une quarantaine d'étudiants de différentes écoles et facultés du Maroc. Seuls 20 candidats sont retenus pour passer à l'étape finale.
- **Entretien technique et RH** : Les candidats choisis à l'issue de l'étape 2 ont passé un entretien technique et RH individuel de 30 min max avec un jury de SQLI Rabat.

1.2.2 Objectifs

L'objectif principal de ce plan d'intégration est bien entendu de présenter le contexte du projet et permettre aux collaborateurs d'intégrer les équipes de développements.

Les formations programmées consistaient tout d'abord en quelques rappels concernant des bases du langage JAVA et des principes de développement mais aussi des technologies utilisées dans le projet. Le plan d'intégration devait surtout faire une montée en compétence des collaborateurs sur des points critiques du projet, surtout en ce qui concerne la qualité de code puisque le code de base est assez volumineux et nécessite un grand effort pour le rendre clair et lisible. Et enfin, comme objectif il fallait initier les collaborateurs à une notion vitale du projet qui est le ré-factoring puisqu'une grande partie du code existant va devoir être améliorée.

1.2.3 Contenu de la formation

Thème	Description	Durée (en j)
Séminaire d'intégration	Séminaire d'intégration	0.25
Environnement technique de Dév	InteliJ (Checkstyle, Formattage, Racourcis...) Git Maven/Ant, Jenkins, Bitbucket...etc	1.75
Java / Clean Code /Design Pattern	JAVA (principe OO, convention nommage, ...) SOLID, SMELL, Immutability Design Pattern: Singleton, factory...etc TU, TI, Mock...etc	2
	Clean Code, Loi de Demeter Java efficace	2
Spring	IOC MVC ...	1
Web service, Microservice, et API	Web service, Microservice, et API: SOA, REST API, Swagger, Architecture Microservice, authentification OAuth 2.0 ... etc	1
Hybris Core Data model, extension, cronjob, impex, service layer, update sys , init sys, Product Modeling (Classification, Product Variants ...), Smart Edit, Interceptors, Validation, Restrictions	Hybris core + Hybris 123	5
Hybris commerce Solr, Order OMS, accelertor, Promotion Engine, CronJob Scripting, Cart & Checkout and Order, Management, Warehouse Integration, Promotions & Coupons, Intrduction à SAP Commerce Cloud, Spartacus	Pres + commerce trails	5
Devops	Introduction à l'univers du Devops: Bonnes pratiques CI/CD Outils (Docker, Puppet ...etc)	0.5
Skills	Formation officielle Skills: Orga Méthodologie ...	0.5
Agile	Jira Confluence Rituels scrum: SUM, spring planning....etc. Rigueur administrative, comment écrire un commentaire Jira constructif (observation, analyse, next step)	0.5

Tableau 1 : Plan de formation

Dans ce qui suit, nous allons traiter quelques points qu'on juge important, l'objectif est de présenter les compétences, qui étaient demandés par le projet et en même temps ça va résumer le déroulement du mois de formation.

Principes SOLID : Le mot SOLID est un acronyme pour les cinq premiers principes de conception orientée objet (OOD) de Robert C. Martin, populairement connu comme l'oncle Bob.

- **S**: Single-responsibility principle
- **O**: Open-closed principle
- **L** : Liskov substitution principle
- **I** : Interface segregation principle
- **D** : Dependency Inversion Principle

Nommage : Les noms sont partout dans le logiciel. Nous nommons les variables, fonctions, les arguments, les classes, les paquets, les fichiers sources etc. Parce que nous faisons ça tellement nous devons le faire bien. Les noms doivent révéler l'intention de la classe, les fonctions et variables. Si les noms sont propres, alors il n'y a pas besoin de commentaires. Le contenu des commentaires peut être le nom de la fonction elle-même.

Les fonctions : Les fonctions doivent faire une seule chose, la faire bien, et ne faire vraiment que ça c'est la règle principale et à cela s'ajoute quelques points importants. Les fonctions doivent être brèves, si un bloc de code nécessite un commentaire alors ce bloc doit être dans une nouvelle fonction. Il faut avoir la déclaration, l'initialisation et l'appel des variables dans l'ordre. Eviter au maximum les effets de bords, il ne faut pas que la fonction ait un comportement caché. Eviter les paramètres de sortie et préférer les exceptions qu'aux codes de retour.

Les classes : Les classes sont les composants essentiels d'une application, si elles sont bien représentées alors l'application est bien structurée et sûrement vas être facilement extensible et maintenable. Pour quelqu'un qui vas lire notre code, on doit s'assurer que notre classe est facilement compréhensible et qu'on révèle très bien nos intentions, en fait ça doit être comme une histoire qu'on lit de haut en bas et c'est les noms des fonctions qui expliquent les différents scénarios. Finalement les classes doivent avoir un nombre limité de variable et utiliser au

maximum ces variables, la répartition des responsabilités, le couplage faible et la haute cohésion sont les facteurs essentiels d'un code propre et seront traité dans la suite.

Qualité de code : Compte tenu des exigences du projet, il n'est plus question de développer de manière traditionnelle et s'arrêter quand le code marche, il faut aller au delà et s'assurer que le code écrit est bien structuré et facilement compréhensible par les autres membres de l'équipe. Cela permet notamment à un autre développeur de modifier le code existant ou au même développeur de s'y retrouver lorsqu'il reprend son propre code longtemps après. Au début des points essentiels ont été traités, tel que les règles de nommage, le style d'écriture de code qui est un point important, dans l'équipe nous avons un style d'écriture commun qu'on a défini et qu'on affecte à nos éditeurs de texte et au logiciel de contrôle de la qualité de code pour qu'il soit respecté.

A vrai dire la qualité de code est le premier souci des grands projets, tant que le code s'agrandit la lisibilité diminue et l'effort de modification devient au fur et à mesure plus difficile, à un moment on peut trouver que le coût de refaire toute la solution est moins que le coût du changement.

Le code propre ou « Clean Code » est un point traité par des experts de la programmation tel que Robert C. Martin, notamment dans son fameux livre « Clean Code : A Handbook of Agile Software Craftsmanship ».

Coder proprement ne peut se faire qu'à travers un changement de la manière avec laquelle on voit le code, dorénavant on va coder pour les autres. Pour se faire plusieurs règles et principes doivent être respectées, même si ces règles ne peuvent pas être quantifiées et dépendent beaucoup du contexte mais il est important de les connaître.

Principes GRASP : GRASP pour « General Responsibility Assignment Software Patterns » sont des patrons de conception de haut niveau, à l'inverse des design patterns du GOF qui eux sont plus concrets et peuvent être directement implémentés. Le but principal de ces principes est de guider le développeur dans l'attribution des responsabilités à des objets qui collaborent. On distingue 9 patrons de conception GRASP :

- Expert en information
- Créateur
- Faible couplage

- Forte cohésion
- Contrôleur
- Polymorphisme
- Fabrication pure
- Indirection
- Protection

1.3 Contexte du projet

1.3.1 Présentation du client

SCOTT Sports SA (auparavant Scott USA) est un équipementier sportif suisse, fondé en 1958 aux États-Unis par Ed Scott.

La vente de vélos, d'accessoires et de vêtements cyclistes est la principale activité de Scott Sports, avec 65 % de l'activité totale du groupe, et est un acteur majeur du marché avec par exemple 270 000 vélos vendus par an, dont 25 000 en Suisse en 2005. Scott possède une gamme essentiellement consacrée au vélo tout terrain et au vélo de route, mais possède également des modèles urbains, des vélos électriques et des fixies. Le modèle semi-rigide destiné à la compétition, le Scott Scale est l'un des vélos les plus titré en cross-country, avec son utilisation notamment par Nino Schurter.

En juin 2015, Scott achète la marque allemande Bergamont à BMC. Scott développe également une gamme de textiles techniques pour le cyclisme.

Scott est très impliqué dans le parrainage sportif, en particulier dans la discipline du vélo tout terrain. Au niveau international, les activités de parrainage de Scott sont divisées en deux équipes : la formation de XC Scott-Swisspower, avec notamment Nino Schurter et Florian Vogel, et l'équipe de descente Scott, avec notamment Floriane Pugin. Scott est également présent en soutien à des formations nationales, par exemple Scott France, qui a soutenu des coureurs comme Julien Absalon ou Pauline Ferrand-Prévot⁸.

Pour le cyclisme sur route, Scott est sponsor en 2008 de l'équipe cycliste Saunier Duval-Scott. À la suite de contrôles antidopages positifs survenus lors du Tour de France 2008, Saunier Duval se retire. Scott assure le partenariat jusqu'à la fin de la saison, avec la société American

Beef. En 2009 et 2010, Scott sponsorise l'équipe HTC-Columbia avec le sprinteur Mark Cavendish. Le vélo Scott Addict devient le vélo détenant le plus de victoires sur les courses féminines et masculines

Pour la saison 2014, Scott est l'équipementier de l'équipe pro tour IAM. , Le 12 décembre 2016 Scott Sports est annoncé comme nouveau co-sponsor de l'équipe australienne Orica-Scott pour la saison 2017.

1.3.2 Problématique

Mi-2016, SAP a publié la première version de l'outil CMS de nouvelle génération, SmartEdit 6.0. Toutes les autres versions ont mis à jour ce module, tandis que l'ancien cockpit du CMS n'a pas été modifié depuis des années. Bien sûr, SAP prévoit de mettre l'ancien WCMS en arrière-plan et considère SmartEdit comme un "nouveau WCMS". Nous nous sommes demandés: SmartEdit est-il prêt à remplacer le cockpit CMS?

Depuis de nombreuses années, nous avons été confrontés au défi de savoir comment rendre un CMS fonctionnel et confortable, comment trouver le bon équilibre entre fonctionnalité et simplicité et comment trouver une solution extrêmement flexible et adaptable aux besoins d'une entreprise. C'est un sujet complexe.

Au final, SmartEdit devrait remplacer le bon vieux cockpit WCMS (Web Content Management System). Toutes les fonctionnalités WCMS principales sont désormais prises en charge par SmartEdit et nous voyons que toutes les nouvelles fonctionnalités concernant les WCMS de SAP Hybris Commerce sont ciblées sur ce nouveau cockpit.

1.4 Conduite du projet

1.4.1 Introduction

Afin de garantir un bon rendement de développement en ce qui concerne la qualité et la productivité, le choix d'une méthode de travail est devenu essentiel. En effet, vu la complexité caractérisant les systèmes d'aujourd'hui, le génie logiciel a essayé de remédier à cette complexité en mettant à notre disposition plusieurs démarches à suivre avec des étapes bien définies. C'est ce qu'on appelle les méthodologies de travail. C'est dans cette perspective de bien planifier et maîtriser son travail qu'SQLI a opté pour une méthode agile (SCRUM). Pour toujours assurer un produit livrable au client. La méthode Scrum met l'accent sur les efforts déployés par l'équipe pour la réalisation des fonctionnalités définies au préalable. Le point le

plus fort de cette méthode est que le client joue le rôle d'un collaborateur, il est impliqué dans toutes les phases de développement pour veiller à ce que son produit répond bien à ses besoins.

1.4.2 Méthodologie de travail : SCRUM

Scrum est un schéma d'organisation de développement de produits complexes. Il est défini par ses créateurs comme un « cadre de travail holistique itératif qui se concentre sur les buts communs en livrant de manière productive et créative des produits de la plus grande valeur possible ». Scrum est considéré comme un groupe de pratiques répondant pour la plupart aux préconisations du Manifeste Agile.

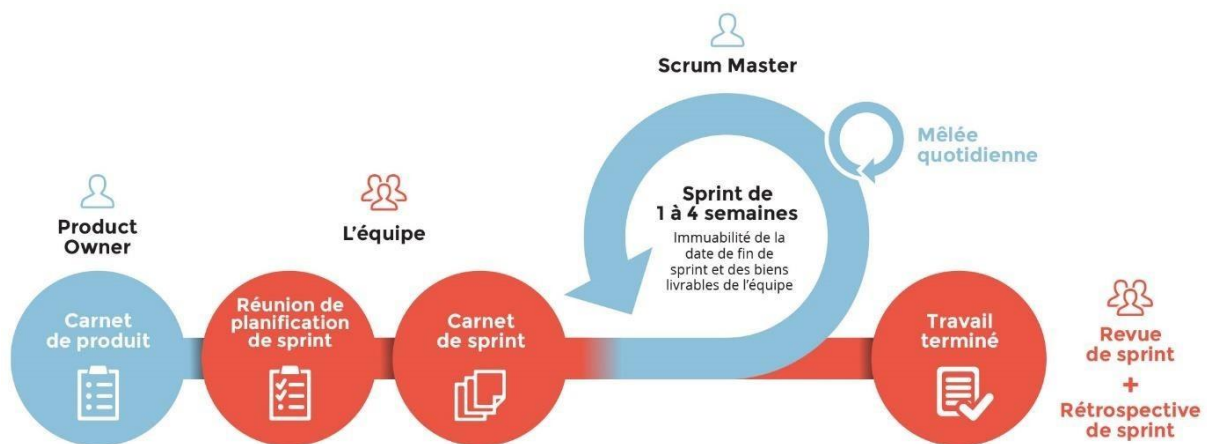


Figure 5 : SCRUM

1.5 Conclusion

Dans ce chapitre, nous avons commencé par la présentation de l'organisme d'accueil et son domaine d'activité avant de situer le projet dans son contexte général

Chapitre 2

Etude fonctionnel et technique

Le module de gestion de contenu Web SAP Hybris est utilisé pour la gestion de contenu sur plusieurs canaux comme Online, Mobile et d'autres interfaces utilisateur. Les entreprises exigent une bonne présentation de leurs produits sur tous les canaux en utilisant une solution de gestion de contenu intégrée fournie par SAP Hybris WCMS.

2 Etude fonctionnel et technique

2.1 Fonctionnalités du module WCMS

Le module WCMS fournit un système de publication multicanal qui vous permet de gérer facilement les sites Web, à la fois les parties transactionnelles et non transactionnelles des sites.

2.1.1 Travailler avec le WCMS Cockpit

SAP Commerce WCMS Cockpit vous permet de gérer vos pages Web à l'aide d'une interface utilisateur conviviale.

Le WCMS Cockpit vous permet de gérer les données du site Web. En utilisant le WCMS Cockpit, vous pouvez:

- Créer et mettre à jour des pages de site Web et des composants de contenu
- Vérifier et modifier l'état de la page du site Web
- Synchronisez les versions de catalogue de votre site Web
- Rechercher des pages de sites Web selon des critères avancés

Présentation du WCMS Cockpit

Les principales caractéristiques du WCMS Cockpit sont:

- Perspective de catalogue pour une gestion avancée des pages du site Web.
- Perspective de modification en direct pour une gestion simplifiée des pages du site Web y compris la fonctionnalité pour parcourir votre site Web.
- Modes d'affichage de liste et de grille des pages du site Web.
- Mode d'édition pour créer de nouveaux composants et modifier des composants de contenu existants.
- Mode d'aperçu pour prévisualiser les modifications.
- Mode personnalisé pour une duplication facile des pages.
- Édition de données de masse lors de l'utilisation de la vue liste.
- Personnalisation des vues lors de l'exécution, par exemple en ajoutant ou en supprimant des colonnes, des sections ou des champs de liste.
- Changez de rôle utilisateur lors de l'exécution et modifiez la configuration de la vue en conséquence.

- Conservez les listes d'utilisateurs des éléments favoris (persistantes).
- Actions de flux de travail avec des pages de site Web jointes, par exemple suivi, commentaires et changement de statut.
- Synchronisation de composants de contenu unique, ainsi que de pages de site Web entières.

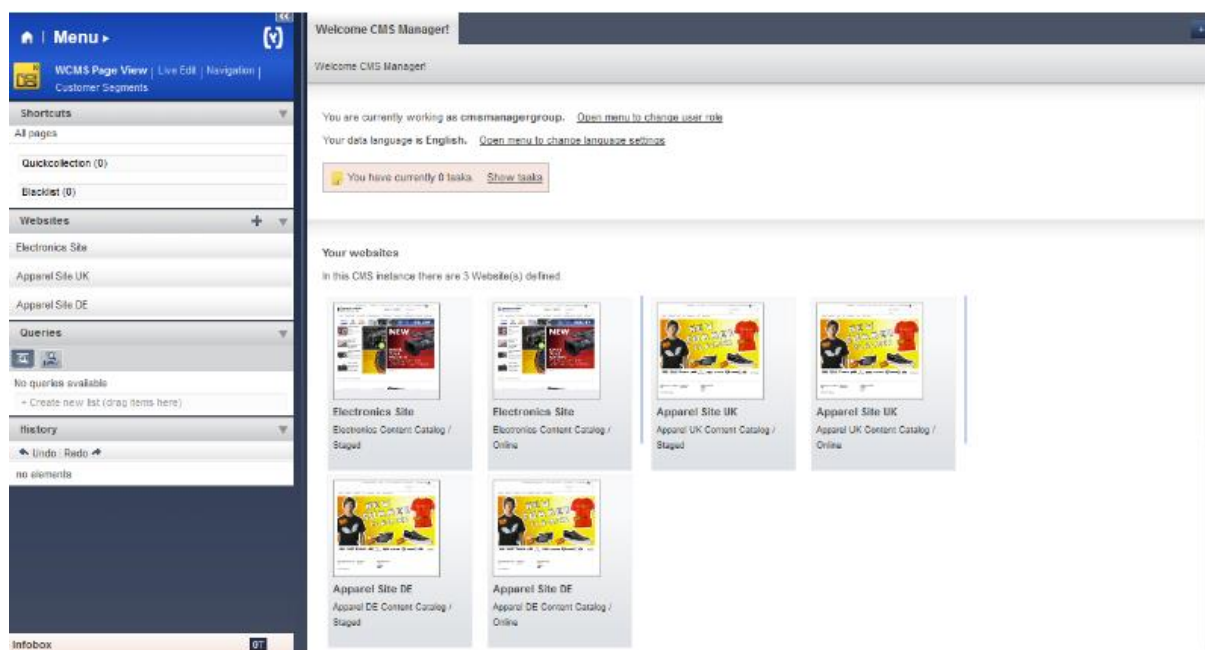


Figure 6 : WCMS

Le WCMS Cockpit contient les zones suivantes :

- Zone de navigation
- Zone du navigateur
- Espace cockpit

Voyons maintenant chacun de ces éléments en détail.

Zone de navigation

Sur le côté gauche, vous avez le volet de navigation, où vous pouvez naviguer vers les raccourcis, les requêtes, les sites Web et l'onglet Historique.

Zone du navigateur

Au centre, vous avez la zone Navigateur, qui est utilisée pour gérer le contenu Web pour tous les canaux. Vous pouvez ajouter un en-tête supérieur, un en-tête inférieur et un pied de page.

Espace cockpit

Sur le côté droit, vous avez la zone Cockpit. Les options suivantes sont disponibles sous la zone Cockpit dans la catégorie de base -

- Nom
- Titre de la page
- Modèle de page
- Version du catalogue
- Étiquette
- Page d'accueil, etc.

De même, vous avez la viabilité du contexte, la navigation, l'URL et l'onglet Administrateur dans la zone du cockpit, qui peuvent être utilisés pour gérer le contenu Web.

2.2 Architecture du module WCMS

Le module WCMS est un ensemble d'extensions fournissant des services de gestion de contenu.

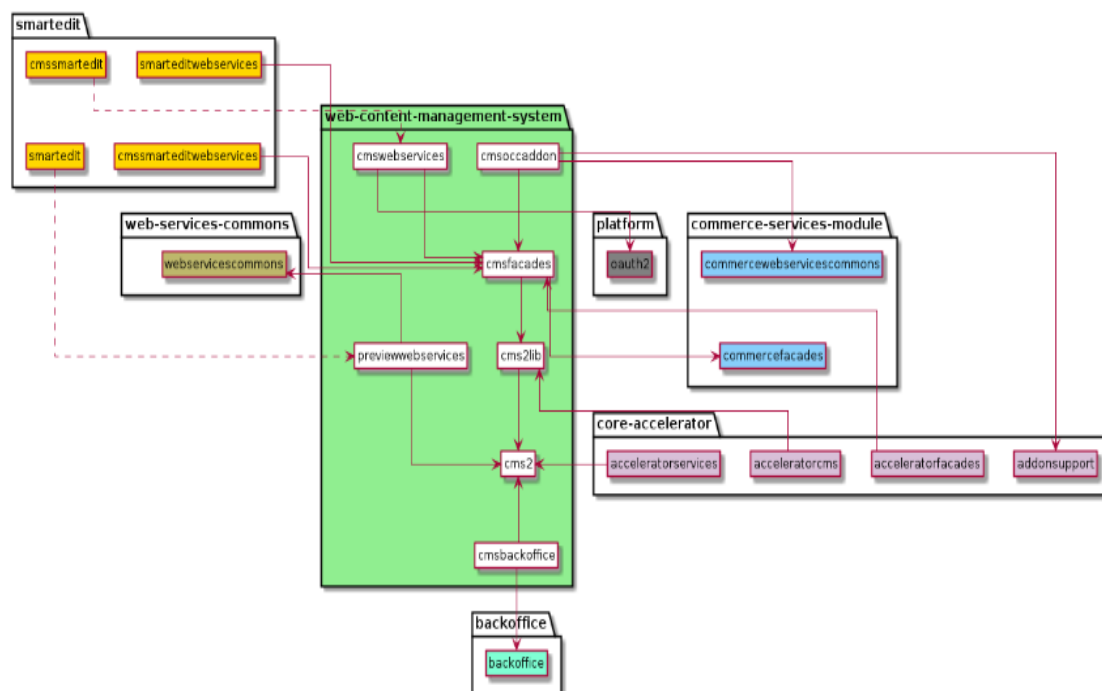


Figure 7 : Diagramme des dépendances

2.2.1 Recipes

Les recettes suivantes contiennent le module WCMS :

- b2b_acc_plus
- b2b_c4c
- b2b_china
- b2c_acc_cis
- b2c_acc_plus
- b2c_acc_ymkt
- b2c_b2b_acc_cpq
- b2c_b2b_acc_dp
- b2c_b2b_acc_oms
- b2c_china
- marketplace_acc
- marketplace_acc_china
- sap_aom_som_b2b_b2c

2.2.2 Extensions

Le module WCMS comprend les extensions suivantes:

Cms2 Extension

L'extension cms2 est le cœur du module WCMS. En utilisant l'extension cms2, vous pouvez créer un site Web que vous pouvez ensuite gérer à l'aide de SmartEdit.

Cms2lib Extension

L'extension cms2lib fournit le composant Bannière, le composant Détails du produit, le composant Liste de produits et le composant Images tournantes.

cmsbackoffice Extension

L'extension de backoffice cms définit les propriétés et les configurations requises pour créer un espace WCMS dans le cockpit d'administration du backoffice pour gérer les éléments liés au CMS.

cmsfacades Extension

L'extension cmsfacades est une extension de modèle qui contient la logique métier pour récupérer et manipuler les éléments liés au CMS, tels que les pages, les composants, les menus et la navigation.

cmsoccaddon AddOn

Le cmsoccaddon AddOn vous permet d'interroger les données de page et de composant CMS à l'aide de l'API OCC CMS REST. Alors que SmartEdit fournit une grande quantité de fonctionnalités de l'API REST CMS avec un accent sur la gestion des CMS, le cmsoccaddon AddOn est destiné uniquement à la consommation de la Storefront.

cmswebservices Extension

L'extension des services Web cms fournit une couche d'API REST pour exposer les éléments liés au CMS. Il est destiné à être indépendant du client, de sorte qu'il puisse éventuellement être consommé en tant qu'API sans tête.

previewwebservices Extension

L'API d'aperçu expose la capacité de la plate-forme à créer l'objet CMSPreviewTicketModel qui contient les paramètres de session pour l'accélérateur, tels que la version du catalogue, les données utilisateur ou les données de langue.

2.3 Comparaison entre SmartEdit et CMS Cockpit

La gestion de contenu non-produit est depuis de nombreuses années le maillon faible de la plateforme SAP Commerce. Formellement, la suite avait un module WCMS de la toute première version, mais il était évident pour tout le monde que la solution était terriblement démodée et dépassée. Déjà en 2016, SAP Hybris Commerce a été étendu avec une nouvelle solution, SmartEdit. D'une part, SAP l'a publié trop tôt.

Smartedit avait suscité de nombreuses critiques de la part des utilisateurs et des développeurs. D'un autre côté, il était attendu depuis longtemps. Il n'y a pas eu beaucoup d'eau sous le pont depuis 2018, mais la situation avec Smartedit a définitivement pris une tournure positive. Cependant, étant une composante «non obligatoire» pendant deux ans, elle a été généralement ignorée par la communauté. Après tout, pourquoi devrions-nous utiliser Smartedit s'il existe un bon vieux cockpit WCMS, éprouvé et éprouvé, bien qu'avec des problèmes connus et une mauvaise personnalisation? C'est pourquoi seul un petit pourcentage de développeurs connaît Smartedit du point de vue technique.

Il est à noter qu'à partir de la version 1811, nous n'avons actuellement plus d'options. Le cadre du cockpit, et avec lui le WCMS Cockpit et la personnalisation BTG, ont été complètement supprimés de la plateforme. Ainsi, Smartedit est la seule option pour la gestion et la personnalisation de contenu.

2.3.1 Technologie

Malgré les similitudes de l'interface utilisateur, SmartEdit n'est pas construit avec Cockpit NG Backoffice Framework. Le Cockpit NG est intensivement développé par SAP pour être utilisé pour les applications de gestion de données dans SAP hybris. Par exemple, ce cadre est utilisé comme base pour tous les nouveaux cockpits, tels que le Backoffice PCM pour les produits ou le module de recherche adaptative pour la recherche et la navigation. SmartEdit ne l'utilise pas du tout. Le cœur de SmartEdit est un framework JavaScript AngularJS. Les widgets ZK créés pour d'autres applications de backoffice ne peuvent pas être réutilisés pour SmartEdit. Cela le rend unique dans SAP Hybris Commerce et explique probablement pourquoi le développement de SmartEdit est si lent. Pour l'interface utilisateur, SmartEdit 6.7 utilise AngularJS UI Bootstrap 2.5. L'ancien cockpit WCMS a été construit avec un cadre de cockpit hérité appelé Cockpit Framework. Je dirais que chaque développeur déteste étendre les anciens cockpits parce que le cadre est incroyablement hostile aux développeurs. De nombreux composants des cockpits existants ne sont pas extensibles, et il est toujours difficile d'ajouter de nouvelles fonctionnalités aux cockpits s'ils ne sont pas dans la zone étroite des options possibles. SmartEdit interagit avec la base de données et l'API hybris via des appels AJAX. Du côté hybris, il existe des modules personnalisables, tels que cmswebservices dans ce but.

Par exemple, pour afficher la liste des pages CMS, SmartEdit fait un appel à : / cmswebservices / v1 / sites / electronics / cmsitems . Presque toutes les opérations sont gérées par des services Web.

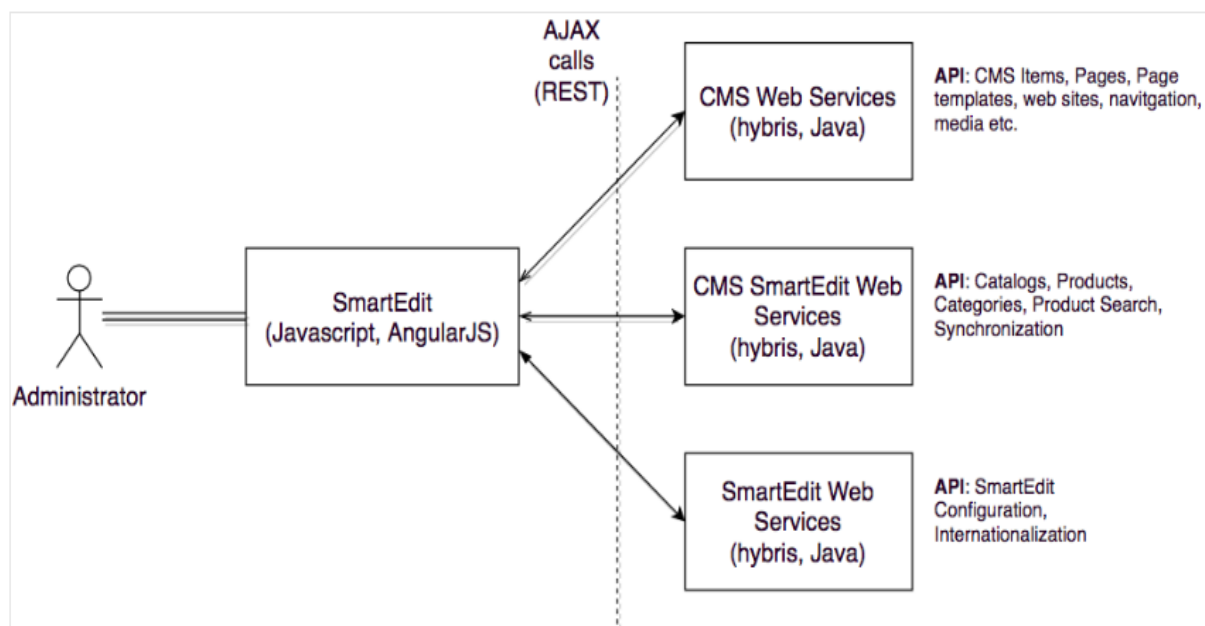


Figure 8 : Appel à cmswebservices

Si vous souhaitez prendre en charge SmartEdit et WCMS Cockpit, vous devrez probablement multiplier vos efforts de développement par deux. Il y a beaucoup de choses que vous devez faire spécifiquement pour SmartEdit en plus de vous faites généralement pour WCMS Cockpit. Par exemple, certains composants (tels que le composant de carrousel de produits) doivent avoir accès au catalogue de produits. Il est nécessaire de fournir les données de commerce électronique à SmartEdit via des services Web pour permettre aux composants Smartedit d'être configurables. Cela signifie que si vous personnalisez des composants de commerce électronique, vous devrez étendre l'ensemble de services Web par défaut uniquement pour Smartedit ainsi que définir ou modifier le code des composants où ces données sont utilisées. Les développeurs AngularJS et Java sont généralement issus de mondes différents. Donc, si vous décidez d'utiliser SmartEdit, vous devrez impliquer les deux rôles même pour les plus petits changements. L'implémentation actuelle de SmartEdit n'est pas très stable mais elle s'améliore avec chaque version. Donc, le revers d'une interface utilisateur sophistiquée pour la gestion de contenu, vous devez tenir compte du fait que les coûts de maintenance seront certainement plus élevés. Ainsi, actuellement, SAP hybris dispose de trois cadres différents pour les applications de backoffice, Cockpit Framework, Backoffice Cockpit Framework et SmartEdit.

2.3.2 Fonctionnalités

Un aperçu de la parité des fonctionnalités de SmartEdit avec WCMS Cockpit et Live Edit.

SmartEdit est l'éditeur du système de gestion de contenu SAP. Avec chaque version, SmartEdit est plus proche de la parité des fonctionnalités avec le WCMS Cockpit et Live Edit. Le tableau suivant donne un aperçu des fonctionnalités fournies par le WCMS Cockpit et Live Edit qui sont disponibles dans SmartEdit, ou quand elles seront disponibles dans SmartEdit.

Création de sites web et de catalogues

Dans WCMS Cockpit, les administrateurs peuvent créer un nouveau site Web et un catalogue de contenu directement à partir de WCMS Cockpit. Je pense que c'est une opération extrêmement dangereuse et devrait être désactivée pour tous les projets afin de garder les données en sécurité. Dans SmartEdit, ce n'est pas possible.

Fonctionnalité	WCMS Cockpit	Édition en direct	SmartEdit	Version de SmartEdit
Création de catalogues de contenu	✓			
Synchronisation des catalogues de contenu	✓	✓	✓	6.0
Versionnage de contenu			✓	1808

Tableau 2 : Création de sites web et catalogues

Gestion des pages

Comme dans WCMS Cockpit, dans SmartEdit, la liste des pages est simple, pas hiérarchique. La recherche ne fonctionne qu'avec les identifiants de page et les noms de page. Si vous devez trouver une page ayant une URL particulière, vous devrez ouvrir toutes les pages une par une ou deviner le nom de la page. La recherche est plus rapide dans SmartEdit. Il n'y a aucun moyen de filtrer les pages par type ou par d'autres attributs. La page supprimée dans WCMS n'est pas réversible. Dans SmartEdit, vous pouvez restaurer la page supprimée. Les attributs SEO, tels que la description et les mots clés, n'ont pas été ajoutés au panneau d'édition de page. Le panneau lui-même est en partie dynamique: de nombreuses modifications côté serveur nécessitent d'apporter des modifications au code de SmartEdit. Dans WCMS, vous n'avez pas besoin de toucher la base de code ou la configuration dans la plupart des cas, tout est complètement automatique. La page peut être clonée dans SmartEdit, c'est une excellente

fonctionnalité. Malheureusement, SAP ne l'a pas implémenté pour CMS Cockpit. Le modèle de page n'est pas modifiable.

Fonctionnalité	WCMS Cockpit	Édition en direct	SmartEdit	Version de SmartEdit
Définition d'une page comme page d'accueil	✓	✓	✓	1808
Création de pages de contenu	✓	✓	✓	6.1
Création de pages produit et catégorie	✓	✓	✓	6.2
Création de pages e-mail	✓	✓	✓	1905
Pages de clonage	✓		✓	6.4
Modification des attributs de page	✓		✓	6.2
Attribution de restrictions aux pages	✓	✓	✓	6.2
Modification des restrictions de page	✓		✓	6.2
Synchronisation des pages	✓	✓	✓	6.3
Suppression de pages	✓		✓	6.6
Restauration des pages supprimées			✓	1808
Suppression définitive de pages			✓	1808

Tableau 3 : Gestion des pages

Composants et emplacements de page

Le concept des composants et des emplacements de page est le même que dans l'ancien cockpit WCMS. Dans SmartEdit, vous pouvez faire glisser les composants vers un nouvel emplacement sur la page. De la même manière, vous pouvez ajouter de nouveaux composants à partir de la bibliothèque et regrouper des composants dans des emplacements.

Il s'agit d'un problème de mise en œuvre de ce glisser-déposer. Parfois, après avoir déposé le composant dans une nouvelle position, SmartEdit ne le déplace pas du tout sans erreurs ni avertissements.

Fonctionnalité	WCMS Cockpit	Édition en direct	SmartEdit	Version de SmartEdit
Conversion d'emplacements en emplacements non partagés (punchout)	✓		✓	6.6
Rétablissement des emplacements non partagés en emplacements partagés	✓		✓	6.6
Synchronisation des emplacements et des emplacements partagés		✓	✓	6.3
Synchronisation des emplacements non partagés		✓	✓	6.6
Création de composants	✓	✓	✓	6.0
Modification des composants	✓	✓	✓	6.0
Glissez-déposez (déplacez) les composants dans et entre les emplacements		✓	✓	6.0
Gestion des médias dans les composants	✓	✓	✓	6.1
Attribution de restrictions aux composants	✓	✓	✓	6.5
Modification des attributs des composants	✓	✓	✓	6.0
Composants de clonage	✓	✓	✓	6.6
Retrait de composants	✓	✓	✓	6.0
Actions de support sur les composants	✓	✓		Prévu
Prise en charge des composants imbriqués	✓		✓	6.6
Synchronisation des composants	✓	✓	✓	6.3

Tableau 4 : Composant et emplacements de page

Gestions des médias dans les composants

Tant dans SmartEdit que dans WCMS Cockpit, la gestion des médias est mise en œuvre de manière terrible. Dans le cockpit WCMS, lorsque vous souhaitez modifier une image de bannière, vous devez comprendre la différence entre changer l'image d'un objet multimédia, changer un objet multimédia d'un conteneur et changer un conteneur multimédia. WCMS Cockpit ne prend pas en charge l'aperçu SVG. Les grandes images ne s'affichent pas correctement non plus. La compréhension du concept des formats multimédias et de son mappage avec les entités et les modes de vitrine est souvent trop difficile pour les gestionnaires de contenu. Dans SmartEdit, SAP simplifiait trop les choses. Vous pouvez maintenant télécharger l'image au lieu d'une image déjà téléchargée uniquement. Il n'y a aucun moyen de choisir l'image dans la bibliothèque ou de réutiliser l'image téléchargée auparavant. Le lien REMPLACER ouvre une fenêtre contextuelle de téléchargement d'image.

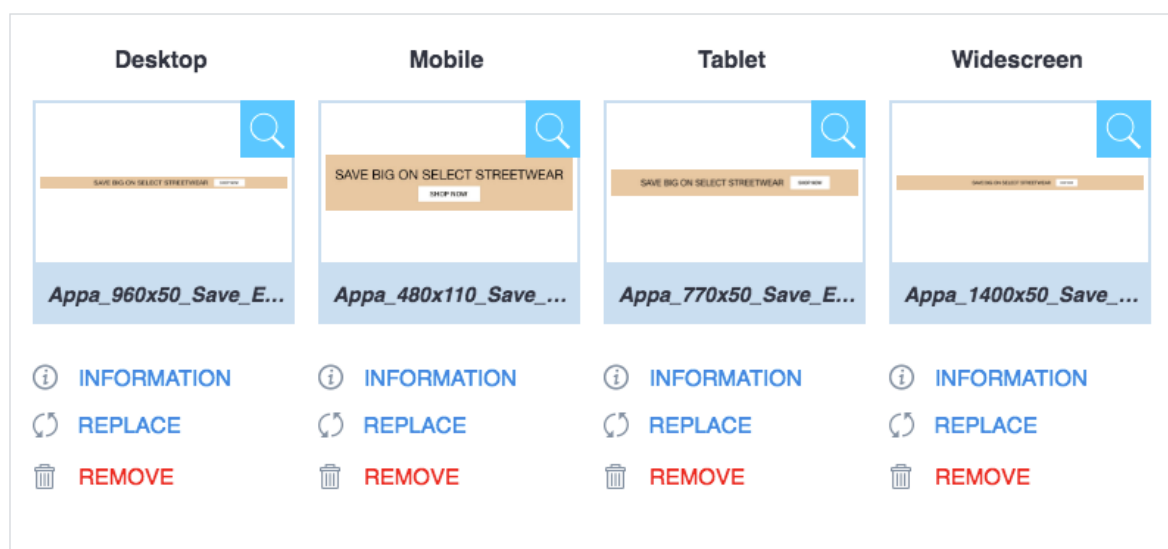


Figure 9 : Gestion de medias

Modification et autorisation

Le contrôle d'accès est dérivé de hybris Platform. Il n'y a presque rien ajouté par Smartedit en plus de ce que nous avons auparavant. Les restrictions FlexibleSearch n'aident pas beaucoup: les services Web les désactivent pour certaines opérations.

Gestion de la navigation

Fonctionnalité	WCMS Cockpit	Édition en direct	SmartEdit	Version de SmartEdit
Aperçu des modifications de contenu	✓	✓	✓	6.0
Modification des informations sur le produit		✓		
Navigation management	✓	✓	✓	6.2
Localisation d'outillage	✓	✓	✓	6.1
Workflows			✓	1905

Tableau 5 : Gestion de navigation

Chapitre 3

Nouvelle génération SmartEdit

SAP Commerce SmartEdit permet aux gestionnaires de contenu de créer et de gérer facilement le contenu de leur site Web à la volée dans différents points d'inflexion et de le mettre à la disposition de leurs clients en un seul clic.

3 Nouvelle génération SmartEdit

Déjà en 2016, SAP Hybris Commerce a été étendu avec un nouveau cockpit appelé SmartEdit. Les gestionnaires de contenu peuvent utiliser SmartEdit pour ajouter ou modifier les pages de leur site Web.

3.1 Fonctionnalités

L'interface utilisateur de SmartEdit est clairement destinée aux utilisateurs finaux. L'idée est que SmartEdit apporte un cockpit WYSIWYG dans lequel les gestionnaires de contenu (ou tout autre type d'utilisateurs finaux) peuvent se passer du support informatique pour gérer les pages de leurs sites Web SAP Hybris Commerce.

L'accès au cockpit SmartEdit avec des autorisations d'écriture et / ou de lecture peut être accordé à des utilisateurs ou groupes d'utilisateurs sélectionnés à partir du BackOffice. Une fois connecté, SmartEdit vous propose une sélection de sites disponibles et la possibilité de maintenir le catalogue de contenu en ligne ou par étapes pour le site sélectionné.

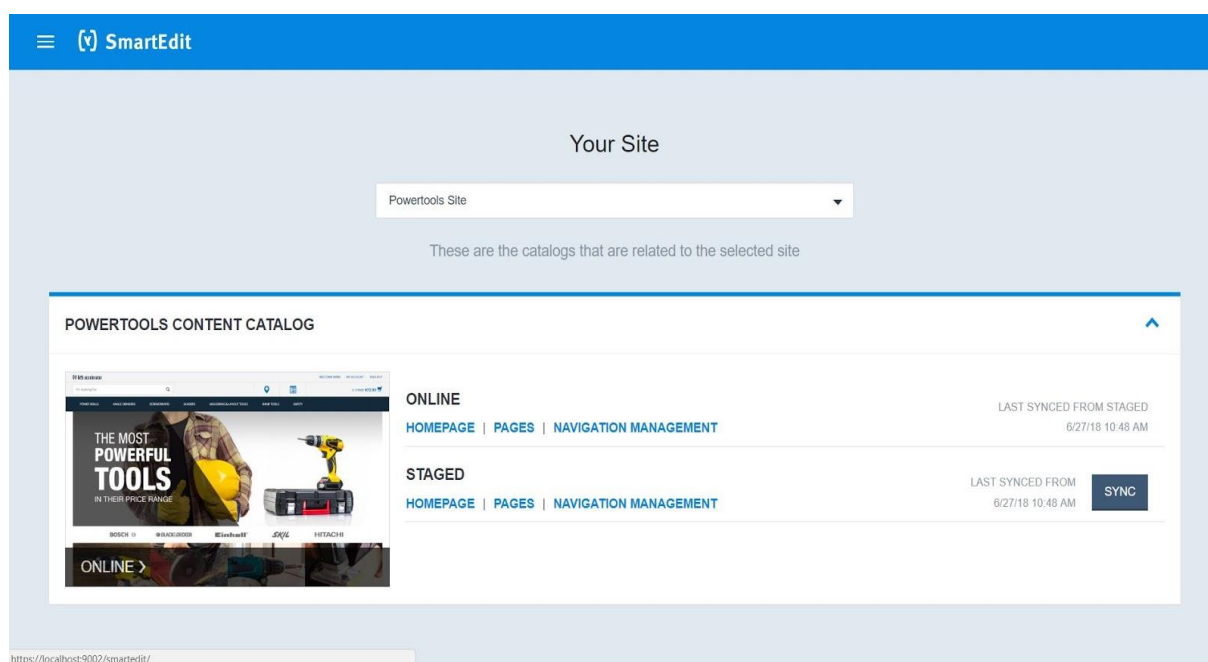


Figure 10 : SmartEdit

Normalement, on ferait toute l'édition du catalogue par étapes et, une fois approuvé, synchroniserait les modifications apportées au catalogue de contenu en ligne.

La sélection du lien «Page d'accueil» dans le catalogue intermédiaire ouvrira la page d'accueil en mode aperçu. Les entrées du menu déroulant en haut à droite représentent les principales fonctionnalités de SmartEdit:

- Aperçu: le mode aperçu affiche les pages Web comme elles le montreraient réellement aux visiteurs de votre site Web.
- Édition de base: le mode d'édition dans lequel vous pouvez uniquement ajouter ou modifier des composants sur la page Web qui ne sont pas partagés avec d'autres pages Web (par exemple, les composants dans la zone d'en-tête et de pied de page partagée ne peuvent pas être modifiés).
- Édition avancée: le mode d'édition dans lequel vous pouvez ajouter ou modifier tous les composants sur la page Web (donc tous les composants partagés également, sachez que la modification d'un composant dans l'en-tête aura un impact sur toutes les pages Web utilisant ce même en-tête).
- Personnalisation: le mode d'édition dans lequel vous pouvez créer des personnalisations d'une page ciblée sur la (combinaison de) groupes cibles dont le visiteur de votre site Web pourrait faire partie.

S'adressant à la nature réactive des pages Web de l'accélérateur, on peut prévisualiser et modifier les pages Web pour différentes tailles d'écran (points d'inflexion nommés comme téléphone mobile, tablette ou ordinateur de bureau).

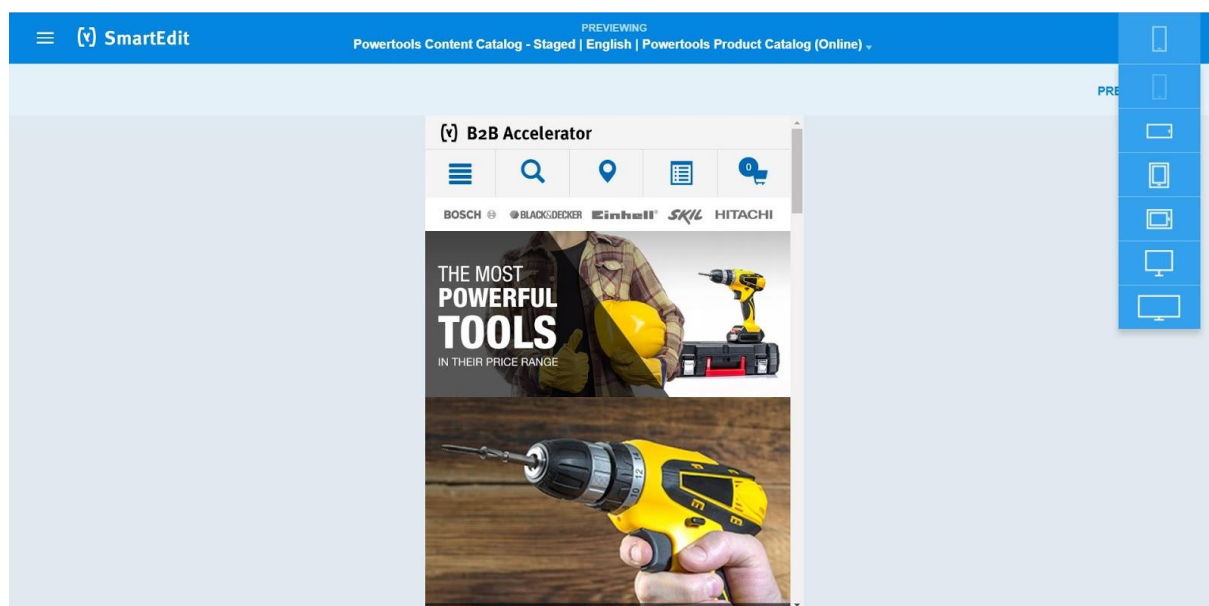


Figure 11 : point d'inflexion

Modifier une page

Comme indiqué ci-dessus, SmartEdit fournit un mode d'édition basique et avancé. Essentiellement, la modification d'une page Web revient à ajouter, déplacer ou modifier des composants sur ces pages. Une fois en mode édition, les composants existants sur une page sont mis en évidence à l'aide de cases avec des bordures en pointillés. Passer la souris sur un composant affichera une option pour modifier le composant, le supprimer ou le glisser-déposer à un autre endroit de la page.

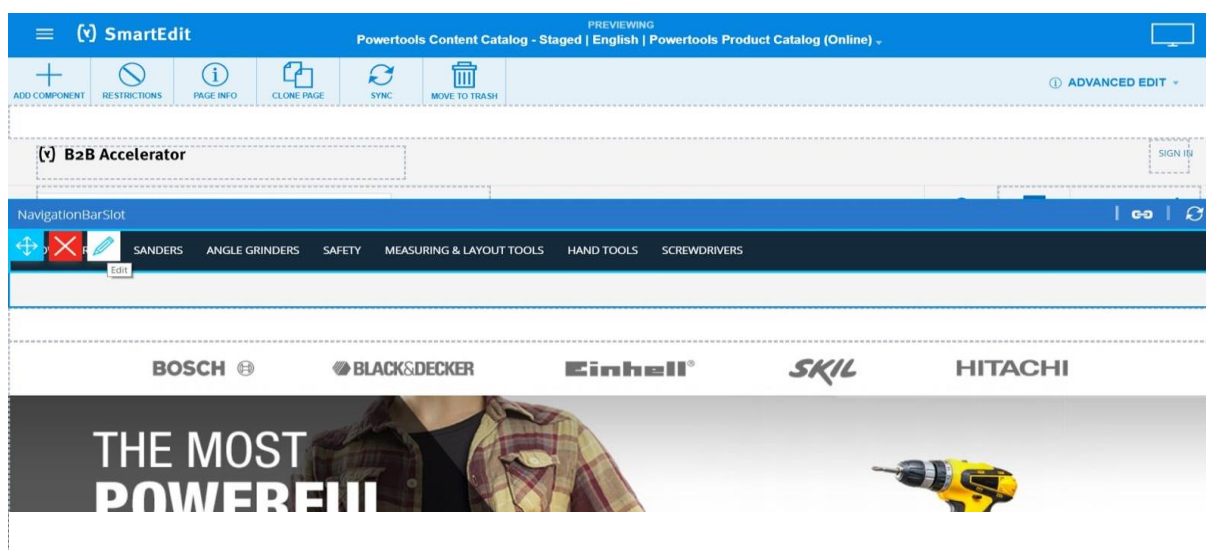


Figure 12 : modifier une page

La modification du composant ouvrira l'éditeur de fenêtres contextuelles affichant le contenu du composant correspondant. De toute évidence, ces éditeurs sont différents pour les composants individuels. Les exemples ci-dessous présentent les éditeurs du composant de menu de navigation, du composant de bannière et du composant de carrousel de produit.

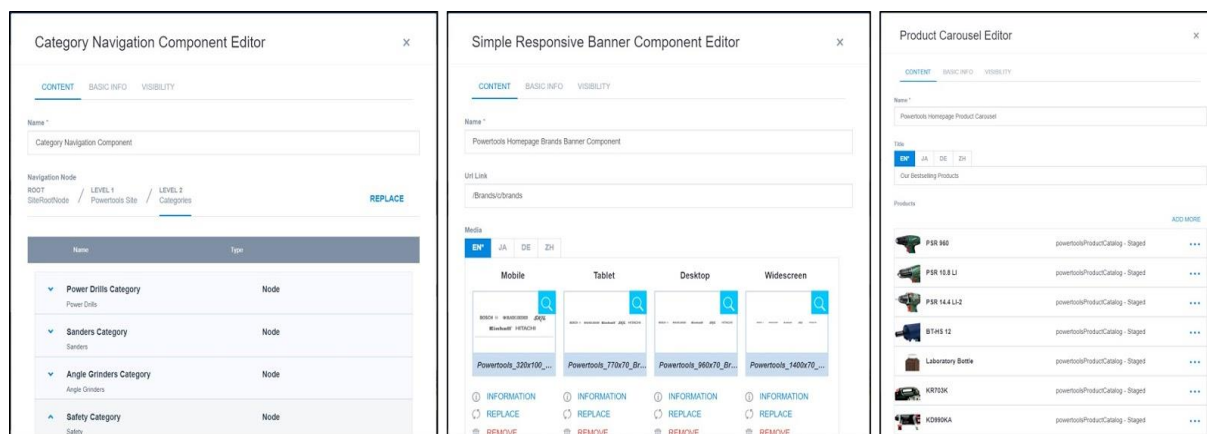


Figure 13 : modifier un composant

Outre la modification des composants existants sur une page, vous pouvez également y ajouter de nouveaux composants. Un certain nombre de composants sont livrés en standard, mais on peut également développer ses propres composants à inclure sur ces pages.

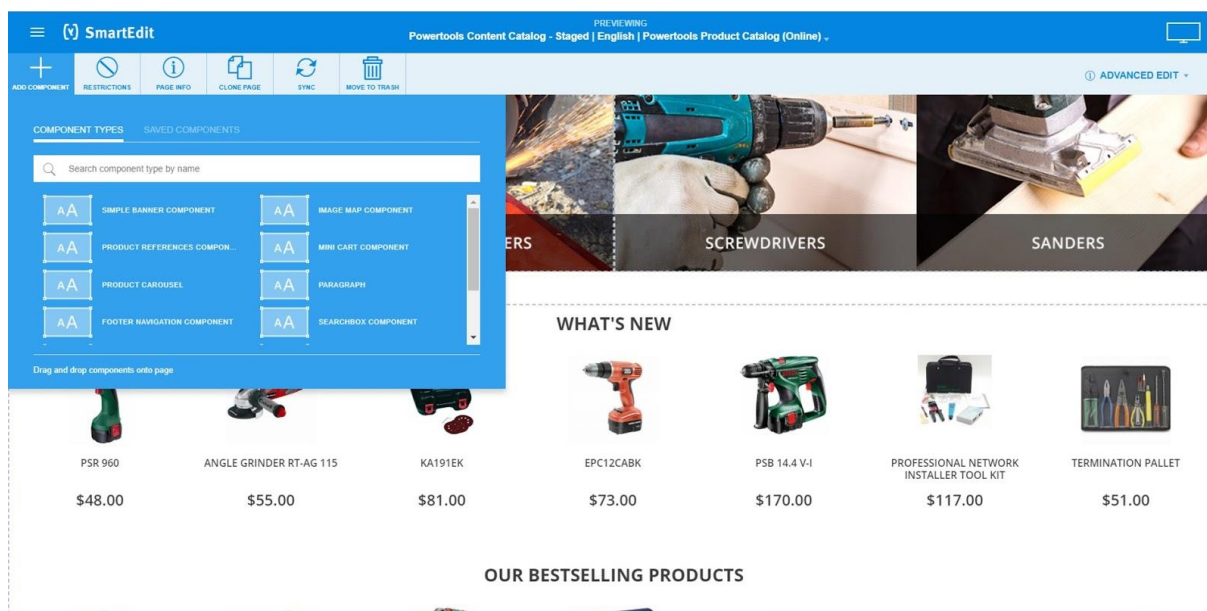


Figure 14 : Ajouter un composant

Ajouter une nouvelle page

Les captures d'écran précédentes étaient toutes basées sur la page d'accueil. De toute évidence, vous pouvez modifier n'importe quelle autre page de vos sites SAP Hybris Commerce de la même manière, y compris la possibilité d'y ajouter de nouvelles pages. La sélection du lien «Pages» dans l'aperçu du site dans SmartEdit vous mènera à une liste de toutes les pages du site sélectionné.

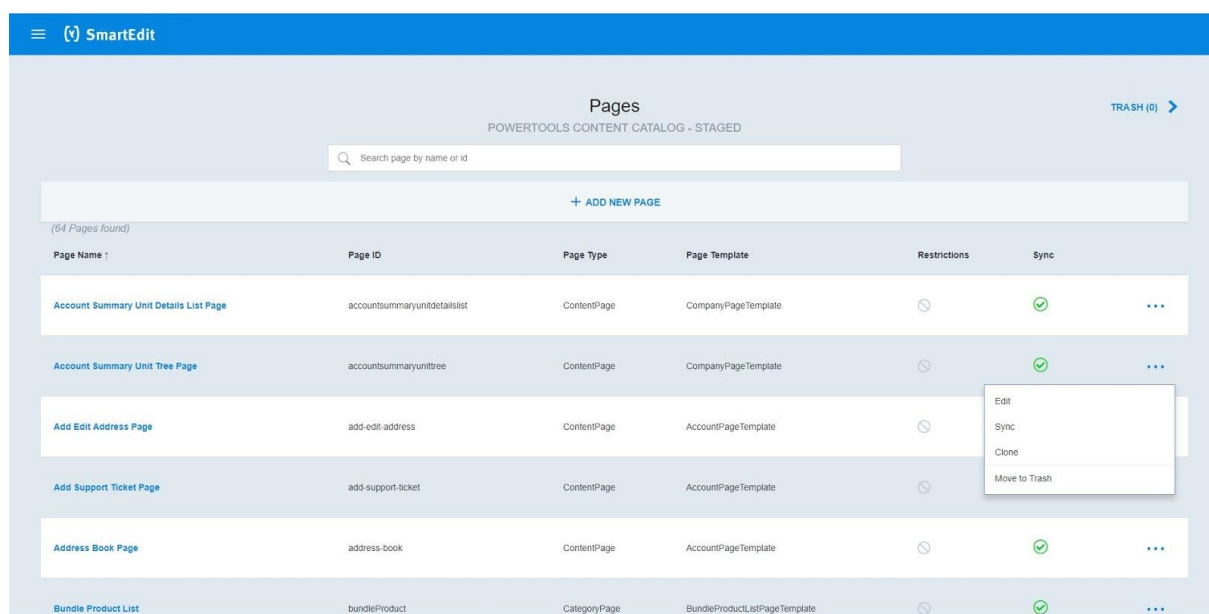


Figure 15 : Ajouter une nouvelle page

À partir de là, vous pouvez rechercher la page appropriée, puis réviser, modifier, cloner, supprimer ou synchroniser la page. L'option 'Ajouter une nouvelle page' ouvrira un assistant en 4 étapes dans lequel vous définissez le type de la nouvelle page (par exemple une page de contenu), le modèle sur lequel la nouvelle page est basée (par exemple une page de destination standard qui est en ligne avec le style de l'entreprise), la condition d'affichage (que ce soit une page principale ou une page de variation limitée par exemple à un groupe d'utilisateurs particulier) et enfin le nom, le libellé et le titre de la nouvelle page.

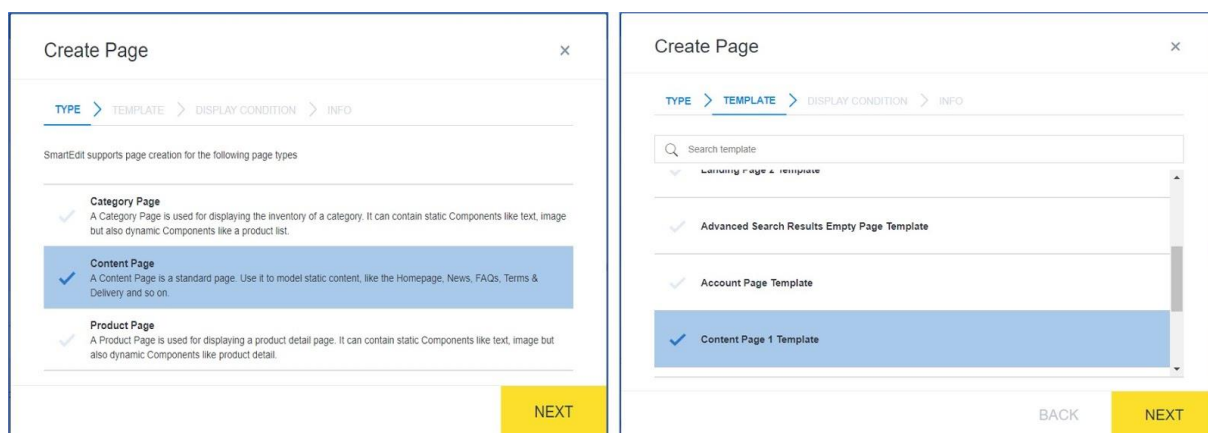


Figure 16 : Template de la page

Changer la structure de navigation

La sélection du lien «Gestion de la navigation» dans l'aperçu du site dans SmartEdit vous mènera à la structure de navigation hiérarchique de votre site Web. Les nœuds hiérarchiques répertoriés sous la racine du site peuvent être liés à des composants de liaison CMS individuels. Ces composants de lien sont définis dans le BackOffice et feront par exemple référence à l'une de vos pages. Les hiérarchies résultantes sont utilisées à divers endroits de votre site comme le menu d'en-tête (voir également la capture d'écran de l'éditeur de composant de navigation par catégorie) et le pied de page.

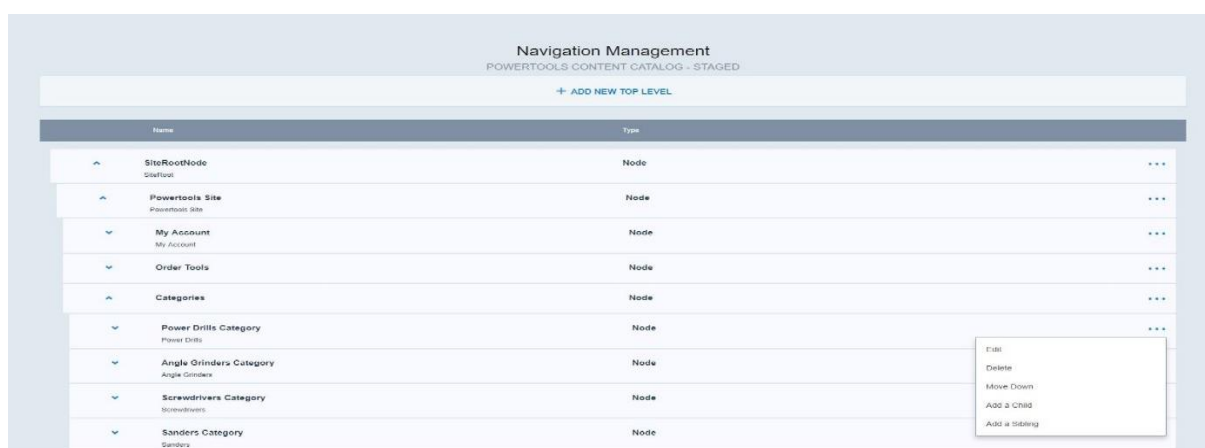


Figure 17 : Gestion de la navigation

Personnalisez votre site

La personnalisation de votre site peut être réalisée avec SmartEdit en créant des soi-disant personnalisations. En général, l'idée est que vous pouvez définir des vues uniques et

personnalisées de votre site SAP Hybris Commerce qui indiqueront quand le visiteur du site appartient au (x) groupe (s) cible (s) affecté (s).

La création d'une nouvelle personnalisation pour l'une de vos pages vous permet de remplacer le contenu du composant par défaut sur cette page par un contenu alternatif. Par exemple, vous pouvez remplacer la bannière générale de la boutique par une bannière spécifique à une personnalisation.

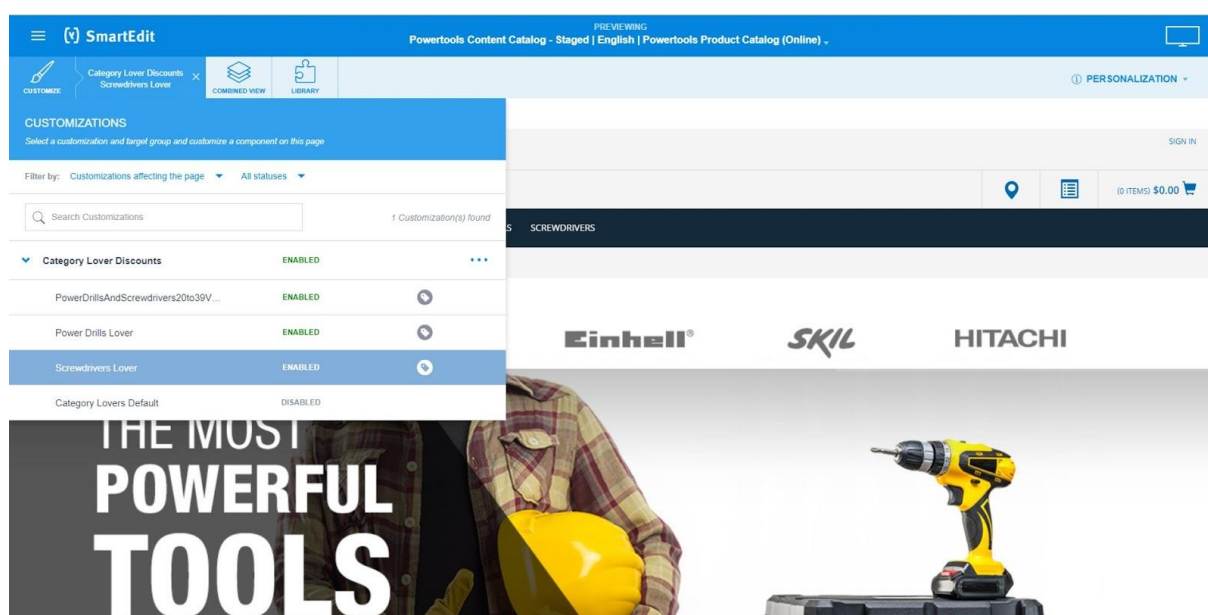


Figure 18 : personnalise votre site

En même temps, chaque personnalisation est affectée à un ou plusieurs groupes cibles. Seuls les visiteurs appartenant à un groupe cible verront la personnalisation de la page, l'autre visiteur verra la page par défaut (ou l'une des autres personnalisations).

Chaque groupe cible comprend un ou plusieurs segments de clientèle. On pourrait définir des combinaisons complexes et hiérarchiques de segments dans une expression de règle pour définir un groupe cible, mais un exemple simple est illustré ici. La personnalisation «Screwdriver Lover» à un seul groupe cible. Et ce groupe cible à un seul segment contenant tous les clients qui ont exprimé leur intérêt pour la catégorie de produits Tournevis.

Target Group ×

[VIEW FULL SCREEN](#)

Name *

Screwdrivers Lover

☐ Applies to all users

Rule expression
(CATEGORY 1593)

Segments

Select segments

OR

CREATE GROUP

CATEGORY 1593

CANCEL APPLY

Figure 19 : Cible la personnalisation

La définition des personnalisations et des groupes cibles associés se fait avec SmartEdit. La définition des segments se fait depuis le BackOffice.

3.2 Smart Edit Architecture

SmartEdit est un cadre JavaScript enfichable avec une interface utilisateur qui permet aux développeurs de gérer les pages Web.

Depandances

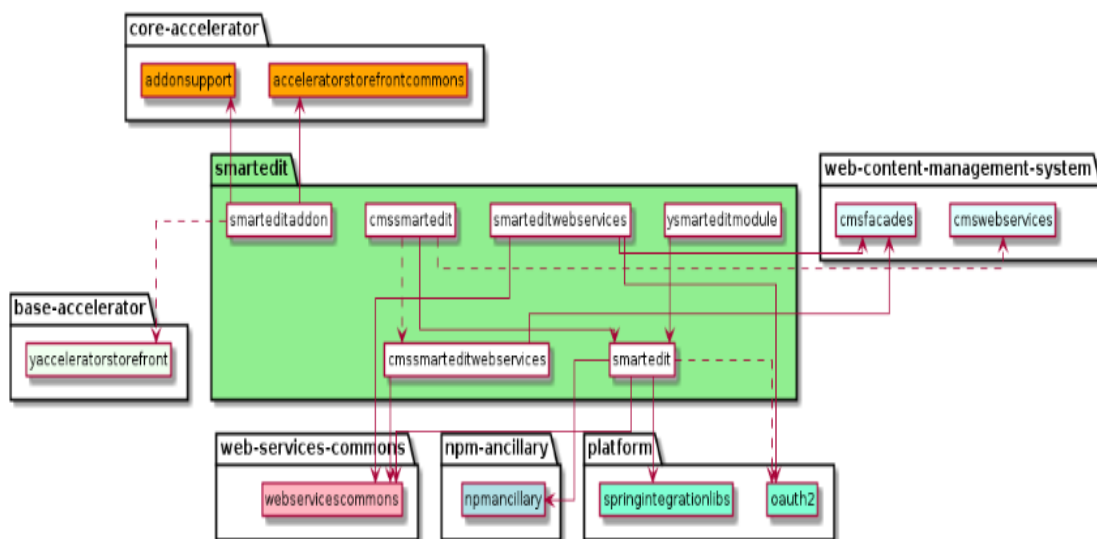


Figure 20 : Diagramme de dépendance de SmartEdit

Recipes

Les recipes suivantes contiennent le module SmartEdit :

- b2b_acc_plus
- b2b_c4c
- b2b_china
- b2c_acc_cis
- b2c_acc_plus
- b2c_acc_ymkt
- b2c_b2b_acc_cpq
- b2c_b2b_acc_dp
- b2c_b2b_acc_oms
- b2c_china

- marketplace_acc
- marketplace_acc_china
- sap_aom_som_b2b_b2c

Extensions

Le module SmartEdit comprend les extensions suivantes:

cmssmartedit Extension

L'extension cmssmartedit fournit le module cmssmartedit qui permet d'activer la gestion de contenu des vitrines.

cmssmarteditwebservices Extension

Le cmssmarteditwebservices AddOn est purement destiné à l'extension est une extension de modèle qui contient la logique métier pour récupérer et manipuler les éléments liés au CMS dans un format qui peut être facilement consommé par l'application Web SmartEdit, comme les types de structure et la synchronisation des pages. Il récupère également le contenu et les informations du catalogue de produits ainsi que leurs informations de version de catalogue et trouve les produits et les catégories dans les catalogues de produits

multicountrysampledatabaddon AddOn

L'addon multicountrysampledatabaddon est le module complémentaire chargé de créer des exemples de données pour la vitrine de l'électronique afin de démontrer les fonctionnalités multi-pays dans l'application Web Accelerator. AddOn crée des sites électroniques supplémentaires pour simuler les hiérarchies de catalogue parent-enfant.

Extension smartedit

L'extension smartedit contient tous les services offerts par le cadre SmartEdit.

smarteditaddon AddOn

Le smarteditaddon est un AddOn chargé de préparer les vitrines basées sur un coréaccélérateur pour qu'elles soient modifiées à l'aide de l'application Web SmartEdit. AddOn modifie les vitrines Accelerator afin qu'elles puissent être modifiées à l'aide de l'application Web

Accelerator. AddOn met automatiquement en œuvre les conditions du contrat de conception de l'accélérateur

smarteditwebservices Extension

L'extension smarteditwebservices est une extension de modèle qui contient des services spécifiques à SmartEdit, tels que la récupération de la configuration SmartEdit et des traductions d'outils.

ysmarteditmodule

L'extension ysmarteditmodule est un modèle de point de départ pour un projet visant à étendre les fonctionnalités fournies par l'extension smartedit.

3.3 Conclusion

Au final, SmartEdit devrait remplacer le bon vieux cockpit WCMS (Web Content Management System). Toutes les fonctionnalités principales de WCMS sont désormais prises en charge par SmartEdit et nous voyons que toutes les nouvelles fonctionnalités concernant les WCMS de SAP Hybris Commerce sont destinées à ce nouveau cockpit.

Chapitre 4

Personnalisation du SmartEdit

La personnalisation (basée sur SmartEdit) offre un moyen intégré et convivial de créer des expériences pertinentes pour vos clients, ce qui est essentiel pour stimuler l'engagement et la conversion. La personnalisation comprend des modules et des extensions qui facilitent l'utilisation du mode de personnalisation dans SmartEdit pour créer une expérience utilisateur entièrement personnalisée.

4 Personnalisation de SmartEdit

4.1 Technologies

4.1.1 Hybris

SAP Hybris est un ensemble de solutions qui stimulent les ventes, le marketing, le commerce et les revenus. Son objectif : proposer une expérience client uniforme et personnalisée en s'appuyant sur des données pertinentes pour concevoir des interactions fructueuses. Les solutions Hybris n'offrent pas seulement une sécurité et une facilité d'utilisation maximales. Elles sont également très puissantes, fournissent des informations en temps réel et simplifient et enrichissent tous les aspects de vos relations clients grâce à l'automatisation

4.1.2 Spring

Le framework Spring est un framework Java généralement utilisé avec la norme J2EE mais ce n'est pas une obligation. Ceci est la description de Spring à sa sortie car c'est de nos jours devenu bien plus qu'un simple framework : c'est maintenant une véritable plateforme pour développer. Il reste néanmoins un framework libre et open-source (avec la License Apache2.0). Il est identifié comme étant un conteneur léger, c'est à dire un conteneur n'ayant pas besoin de toute la structure d'un serveur d'application Java comme la nécessité pour les objets qu'il manipule d'hériter de certaines classes pour créer des relations. Il en reprend malgré tout le principe. L'intérêt principal de Spring est de faciliter la programmation en Java avec les POJO (Plain Old Java Object) c'est à dire les objets Java classique.

4.1.3 AngularJs

AngularJS est né en 2009 dans les locaux de Google.

Angular est construit autour de concepts clés :

- Architecture MVC (Modèle-Vue-Contrôleur) : qui consiste à avoir une stricte séparation entre les données (Modèle), la présentation des données (Vue), et les actions que l'on peut effectuer sur ces données (Contrôleur)
- Data Binding : grâce à ce concept, les liens entre votre code HTML et JavaScript ne seront que plus forts.

- Injection de dépendances : tout comme l'architecture MVC, lorsque l'on parle d'injection de dépendances, on parle d'un concept prépondérant dans tout développement. Grâce à cela, les modules que vous développerez n'auront plus à se soucier d'instancier leurs dépendances.
- La manipulation du DOM au moyen de directives : la manipulation du DOM conduit souvent à la création de code difficilement maintenable et difficilement testable.

4.1.4 Apache Ant

Ant est un projet open source de la fondation Apache écrit en Java qui vise le développement d'un logiciel d'automatisation des opérations répétitives tout au long du cycle de développement logiciel, à l'instar des logiciels Make. Le nom est un acronyme pour "Another Neat Tool" (un autre chouette outil). Ant est principalement utilisé pour automatiser la construction de projets en langage Java, mais il peut être utilisé pour tout autre type d'automatisation dans n'importe quel langage. Parmi les tâches les plus courantes, citons : la compilation, la génération de pages HTML de document (Javadoc), la génération de rapports.

4.1.5 Git

Git est de loin le système de contrôle de version le plus largement utilisé aujourd'hui. Git est un projet open source avancé, qui est activement maintenu. À l'origine, il a été développé en 2005 par Linus Torvalds, le créateur bien connu du noyau du système d'exploitation Linux. De plus en plus de projets logiciels reposent sur Git pour le contrôle de version, y compris des projets commerciaux et en open source. Les développeurs qui travaillent avec Git sont bien représentés dans le pool de talents disponible, et la solution fonctionne bien sur une vaste gamme de systèmes d'exploitation et d'environnements de développement intégrés (IDE). Par sa structure décentralisée, Git illustre parfaitement ce qu'est un système de contrôle de version décentralisé (DVCS).

4.2 Personnalisation

Vous pouvez étendre les services et les fonctionnalités de votre extension SmartEdit personnalisée. SAP Commerce vous recommande de ne pas modifier les ressources `cmssmartedit`, mais de les étendre à l'aide de l'extension de modèle `ysmarteditmodule`. Vous pouvez créer une extension personnalisée à l'aide de `ysmarteditmodule` modèle. La création d'une extension personnalisée vous permet d'ajouter des fonctionnalités à `smartedit` et `cmssmartedit` sans modifier leurs sources.

4.2.1 Création d'une extension SmartEdit personnalisée

On a créé une extension SmartEdit personnalisée nommée `smartedittrail` en utilisant l'extension `ysmarteditmodule`.

1. Démarrer Ant
 - A. Ouvrir une invite de commande
 - B. Cd `<HYBRIS_BIN_DIR>/Platform`.
 - C. Windows : `setantenv.bat`
2. Démarrer le processus de création d'extension
 - A. Saisir la commande : `ant extgen` pour générer l'extension
3. Sélectionner `ysmarteditmodule` comme modèle à utiliser pour créer l'extension.
4. Spécifier le nom et un package pour l'extension comme suit:
 - Name: `smartedittrail`
 - package: `com.sap.smartedittrail`
5. Ajouter la `smartedittrail` extension à la `<HYBRIS_CONFIG_DIR>/localextensions.xml` fichier: `<extension name = "smartedittrail" />`
6. Téléchargez et installez les bibliothèques npm tierces : `ant npminstall`
7. Mise A jour de SAP Commerce : `ant updatesystem`
8. Exécuter la commande : `ant clean build`

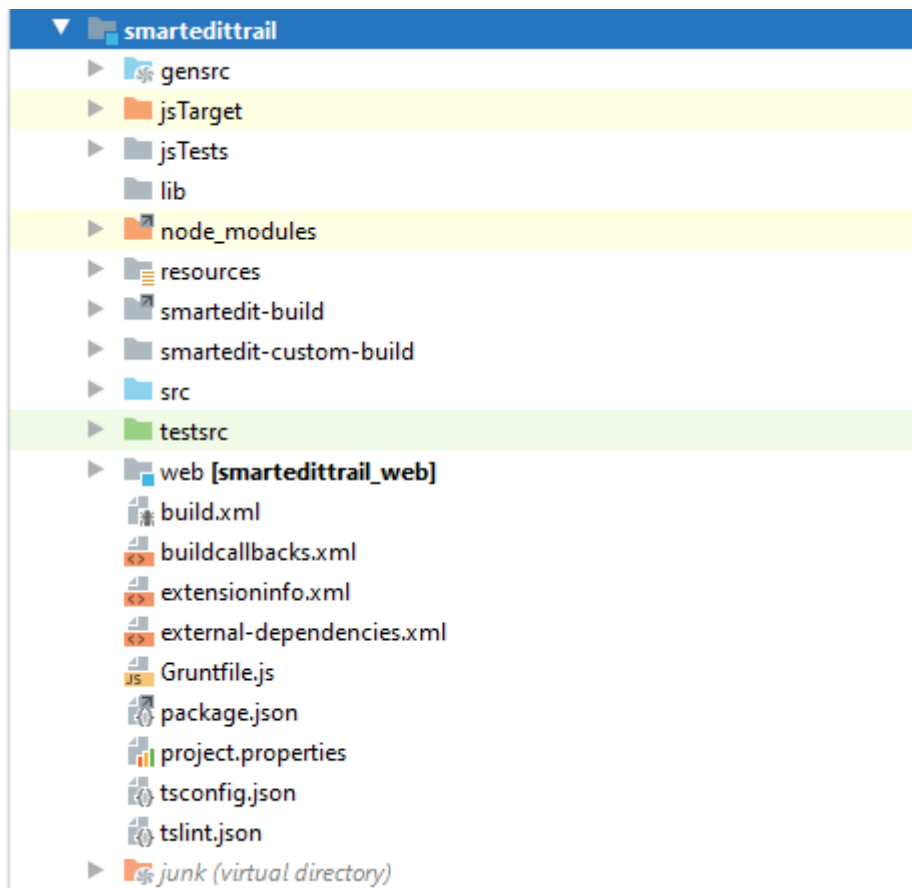


Figure 21 : Structure de l'extension smartedittrail

4.2.2 Lier l'extension avec SmartEdit

On peut lier notre extension personnalisée dans SmartEdit à l'aide de l'éditeur de configuration ou par le fichier ImpEx.

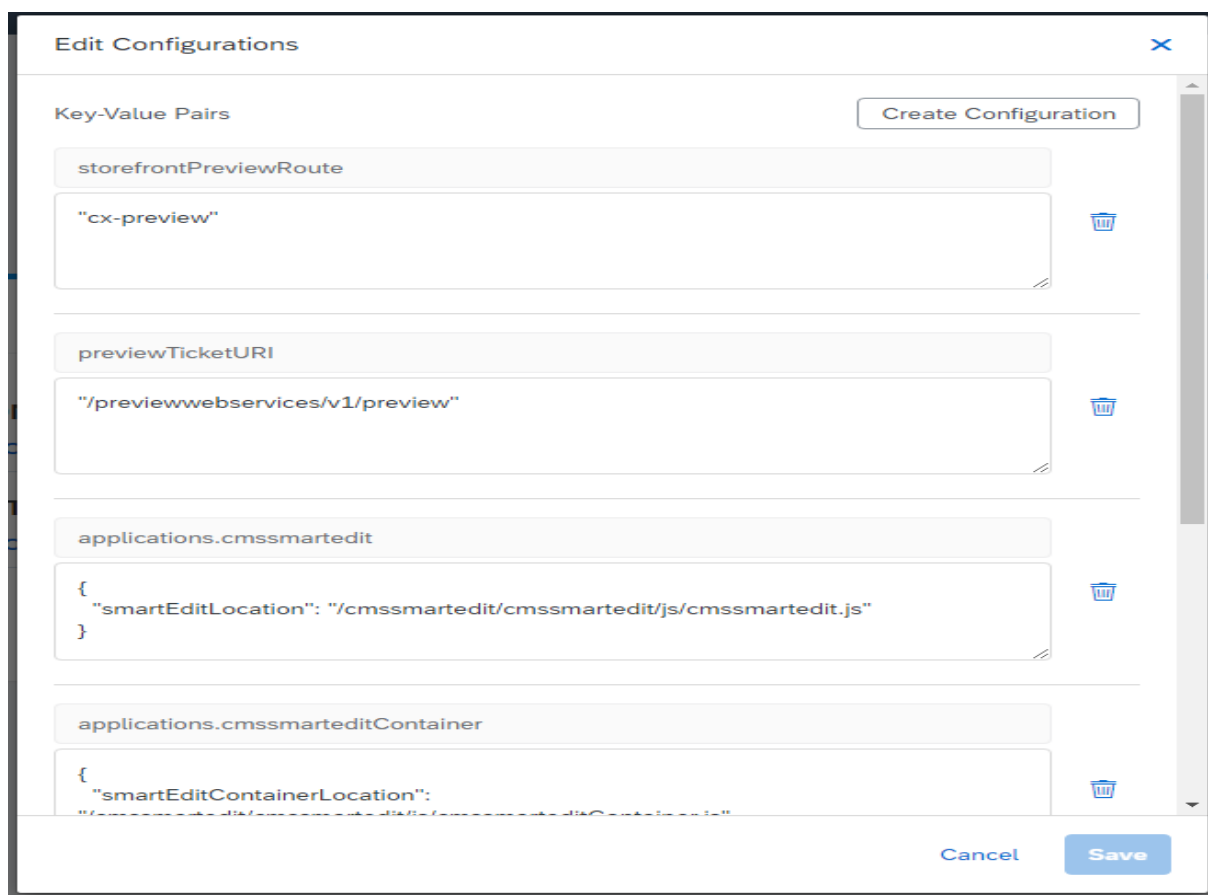
Liaison à travers Configuration Editor

Lier l'extension personnalisée à SmartEdit en ajoutant les configurations suivantes:

- Clé: applications.smartedittrail

```
{ "smartEditLocation" : "/smartedittrail/smartedittrail/js/smartedittrail.js" }
```
- Clé: applications.smartedittrailContainer

```
{ "smartEditContainerLocation" : "/smartedittrail/smartedittrail/js/smartedittrailContainer.js" }
```

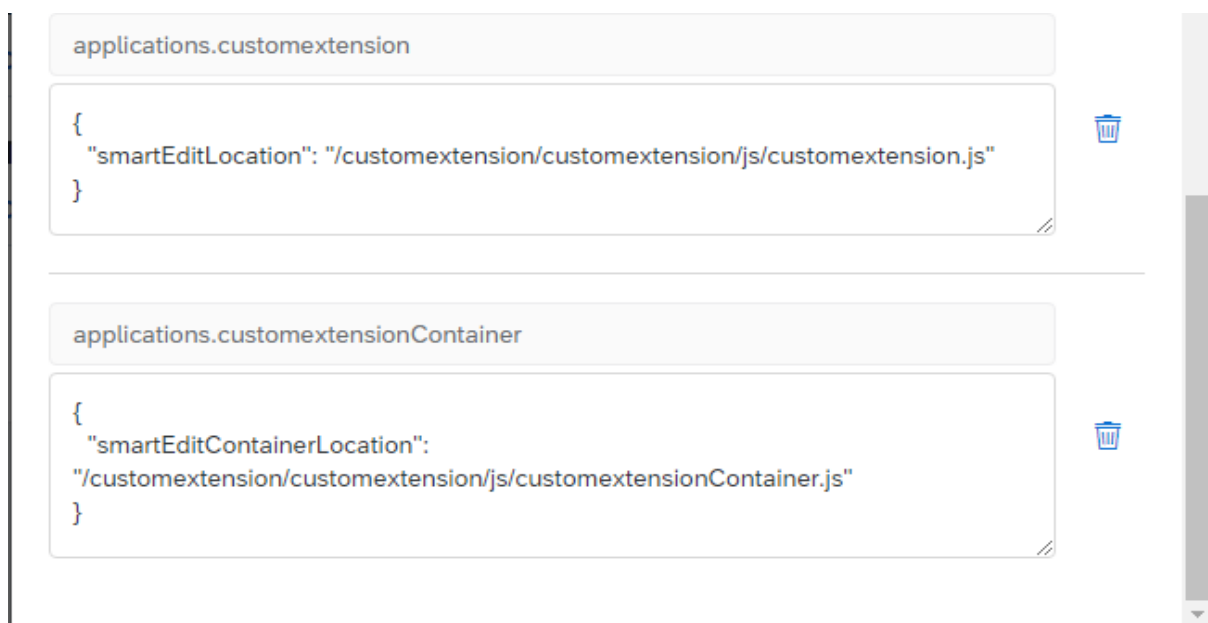


The screenshot shows the 'Edit Configurations' dialog box. It has a title bar with a close button (X). Below the title bar is a 'Key-Value Pairs' section with a 'Create Configuration' button. The main area contains four configuration entries, each with a key field, a value field, and a delete icon (trash can). The entries are:

- Key: storefrontPreviewRoute, Value: "cx-preview"
- Key: previewTicketURI, Value: "/previewwebservice/v1/preview"
- Key: applications.cmssmartedit, Value: {"smartEditLocation": "/cmssmartedit/cmssmartedit/js/cmssmartedit.js"}
- Key: applications.cmssmarteditContainer, Value: {"smartEditContainerLocation": "/cmssmartedit/cmssmartedit/js/cmssmarteditContainer.js"}

At the bottom right, there are 'Cancel' and 'Save' buttons.

Figure 22 : Configuration Editor



The screenshot shows the 'Edit Configurations' dialog box with two configuration entries. The entries are:

- Key: applications.customextension, Value: {"smartEditLocation": "/customextension/customextension/js/customextension.js"}
- Key: applications.customextensionContainer, Value: {"smartEditContainerLocation": "/customextension/customextension/js/customextensionContainer.js"}

Each entry has a delete icon (trash can) to its right.

Figure 23 : Liaison de l'extension à travers Configuration Editor

Liaison Via Impex

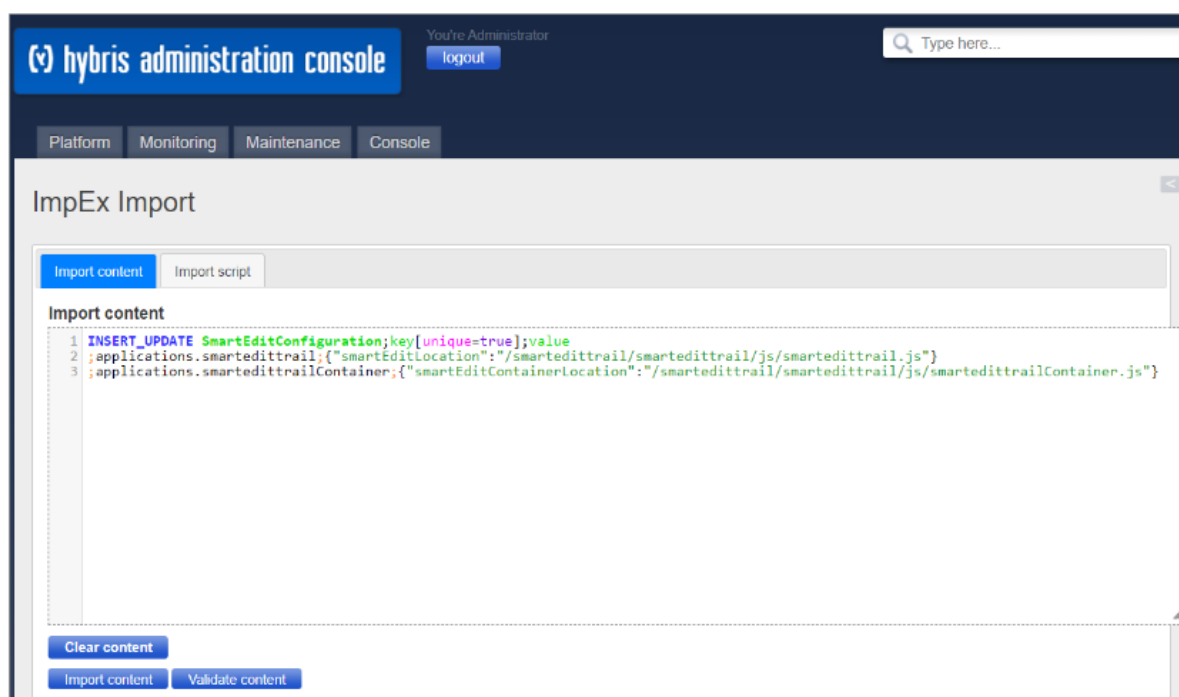


Figure 24 : Liaison Via Impex

4.2.3 Ajout d'une nouvelle page a SmartEdit

Pour ajouter une nouvelle page à SmartEdit en ajoutant un nouvel itinéraire dans ng-route. On le faire directement dans AngularJS. Cependant, dans SmartEdit, on peut également ajouter une page à l'aide de SeRouteService. L'utilisation de ce service ajoute une page avec les avantages suivants:

- Ajoute la nouvelle page au composant de raccourci
- Fonctionne avec Angular

1. Ajouter le fichier :

<HYBRIS_BIN_DIR>/custom/smartedittrail/web/features/smartedittrailContainer/trailPage
trailPage.ts dans smartedittrail par le code suivant :

```
IMPORT * AS ANGULAR FROM 'ANGULAR';

// THIS CREATES AN ANGULARJS MODULE CALLED TRAILPAGEMODULE.
EXPORT CONST TRAILPAGEMODULE = ANGULAR.MODULE('TRAILPAGEMODULE', [])

// THE ANGULARJS RECIPE SPECIFIES A NEW COMPONENT CALLED TRAILPAGE THAT WILL
DISPLAY THE NEW PAGE.
```

```

// IT CAN BE USED IN THE MARKUP OF OTHER PARTS OF THE APPLICATION USING <TRAIL-
PAGE>.

    .COMPONENT('TRAILPAGE',
{

    // THIS PROPERTY SPECIFIES THE NAME OF THE FILE THAT CONTAINS THE
MARKUP USED TO RENDER THE NEW PAGE.

    TEMPLATEURL: 'TRAILPAGETEMPLATE.HTML',

    // THIS PROPERTY SPECIFIES THE CONTROLLER OF THE NEW PAGE. IT CAN BE USED TO
ADD CUSTOM BEHAVIOR TO THE PAGE.

    CONTROLLER: () => {

        // FANCY CONTROLLER CODE.

    }

});

```

2. Ajouter le fichier trailPageTemplate.html dans smarteditrail qui sera le Template de notre page créé précédemment

```

<!-- TO CONTAIN THE TOP TOOLBAR (AND MAKE IT LOOK SMARTEDIT LIKE) -->
<DIV CLASS="SE-TOOLBAR-GROUP">

    <SE-TOOLBAR DATA-CSS-CLASS="SE-TOOLBAR-SHELL "

        DATA-IMAGE-ROOT="IMAGEROOT"

        DATA-TOOLBAR-NAME="SMARTEDITHEADERTOOLBAR"></SE-TOOLBAR>

</DIV>

<!-- THE ACTUAL CONTENT OF THE PAGE -->
<DIV CLASS="SE-LANDING-PAGE">

    TRAIL PAGE

</DIV>

```

3. Modifier le fichier legacysmarteditrailcontainer.ts

Le code legacysmarteditrailcontainer.ts crée une page appelée trailPage en créant un service d'itinéraire.

```

...
import {TRAILPAGEMODULE} from './TRAILPAGE/TRAILPAGE';
...

```



```
@SEMODULE({  
  IMPORTS: [  
    ...,  
    TRAILPAGEMODULE.NAME  
  ],  
  CONFIG: ($ROUTEPROVIDER: ANGULAR.ROUTE.IROUTEPROVIDER) => {  
    'NGINJECT';  
  
    SEROUTESERVICE.INIT($ROUTEPROVIDER);  
    SEROUTESERVICE.PROVIDELEGACYROUTE({  
      PATH: '/TRAIL',  
      ROUTE: {  
        TEMPLATE: '<TRAIL-PAGE></TRAIL-PAGE>'  
      },  
      PRIORITY: 1,  
      TITLE18NKEY: 'TRAIL PAGE'  
    });  
  
  },  
  ...  
})  
  
EXPORT CLASS SMARTEDITTRAILCONTAINER {}
```

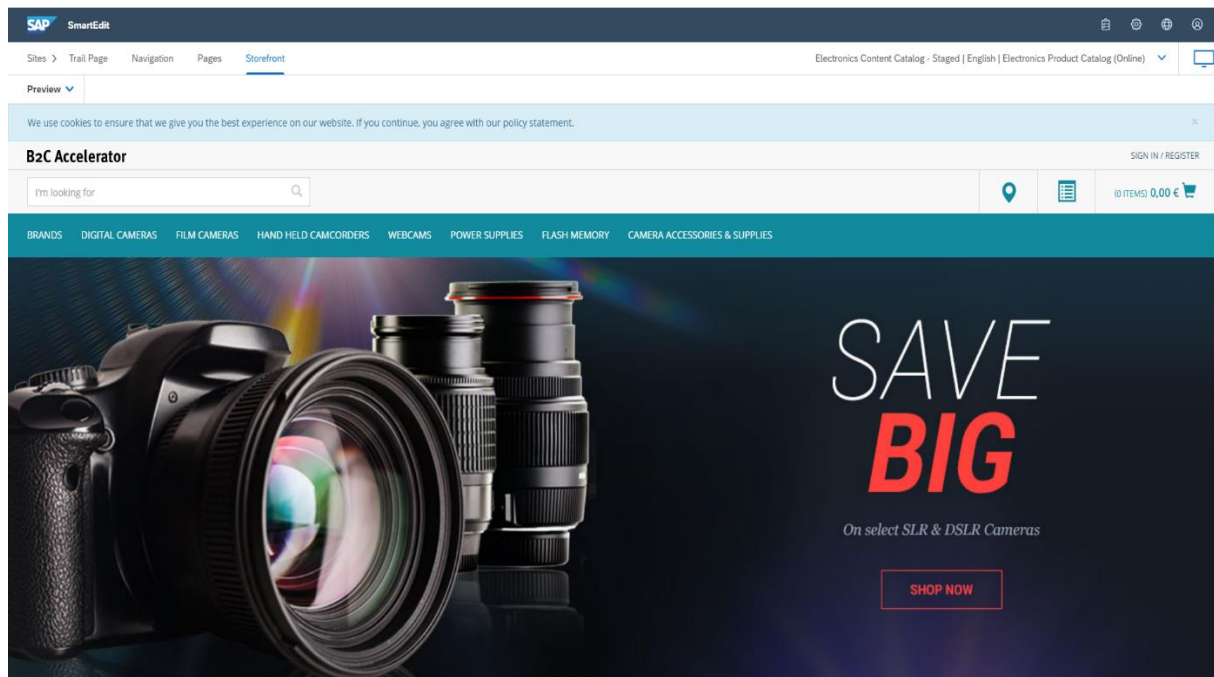


Figure 26 : page personnalisée

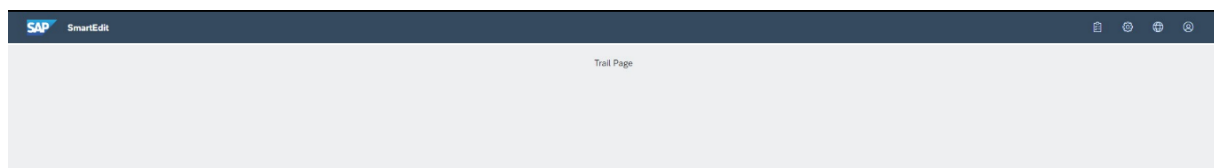


Figure 25 TrailPage

4.2.4 Ajout d'une barre d'outils à SmartEdit

SmartEdit permet de créer des barres d'outils ou d'ajouter des boutons aux barres d'outils existantes. Une barre d'outils est une directive AngularJS que nous pouvons placer n'importe où dans le code HTML. SmartEdit fournit un service de barre d'outils pour nous aider à créer des barres d'outils. Chaque barre d'outils nécessite un nom unique. On utilise le nom unique pour ajouter des boutons au service de barre d'outils. Les barres d'outils SmartEdit prennent en charge les types de boutons suivants:

- **ACTION:** comprend la clé `i18nKey` à traduire et une fonction de rappel sans arguments. On peut l'utiliser dans le service de barre d'outils interne ou externe.

- HYBRID_ACTION: comprend la clé i18nKey à traduire, une fonction de rappel sans arguments et un attribut qui pointe vers un emplacement de modèle HTML. on ne peut l'utiliser que dans le service de barre d'outils externe.
- MODÈLE: se compose d'un attribut qui pointe vers un emplacement de modèle HTML. on ne peut l'utiliser que dans le service de barre d'outils externe.

On a créer un nom de barre d'outils trailtoolbar dans la trailPage et y ajouter un bouton.

1. Modifier le fichier trailPageTemplate.html

```
<DIV CLASS="SE-TOOLBAR-GROUP">
  ...

  <!-- SAMPLE TOOLBAR -->
  <SE-TOOLBAR DATA-CSS-CLASS="SE-TOOLBAR-PERSPECTIVE"
    DATA-TOOLBAR-NAME="TRAIL TOOLBAR"></SE-TOOLBAR>
</DIV>
```

Le code crée une barre d'outils appelée trailtoolbar dans le trailPage. elle apparaît comme suit:

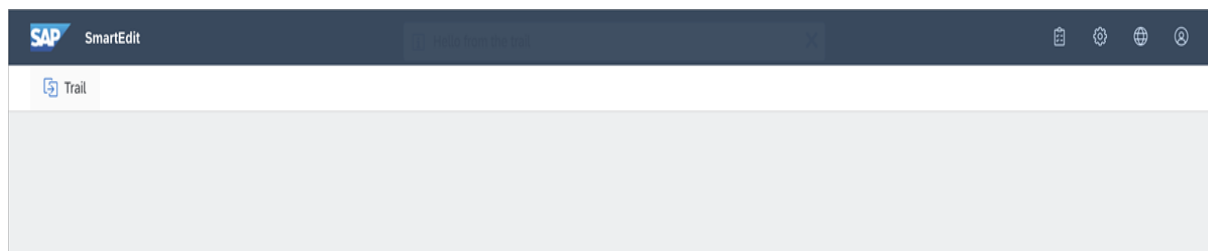


Figure 27 : toolbar personnalisée

2. Modifier le fichier trailPage.ts

```
EXPORT CONST TRAILPAGEMODULE = ANGULAR.MODULE('TRAILPAGEMODULE', [])
  .COMPONENT('TRAILPAGE', {
    ...
    CONTROLLER: (TOOLBARSERVICEFACTORY: ANY, ALERTSERVICE: ANY) =>
  {
    'NGINJECT';

    CONST TOOLBARSERVICE =
    TOOLBARSERVICEFACTORY.GETTOOLBARSERVICE('TRAIL TOOLBAR');

    // THIS PIECE OF CODE ADDS A BUTTON, CALLED TRAIL.BUTTON.
    TOOLBARSERVICE.ADDITEMS([
      TYPE: 'ACTION',
      KEY: 'TRAIL.BUTTON',
      NAME/I18NKEY: 'TRAIL',
```

```

// THIS IS THE CALLBACK FUNCTION THAT WILL BE
EXECUTED WHEN THE TOOLBAR BUTTON IS CLICKED.
CALLBACK: () => {
    ALERTSERVICE.SHOWINFO("HELLO FROM THE
TRAIL");
},
ICONCLASSNAME: 'ICON-DUPLICATE SE-TOOLBAR-MENU-
DDL-BUTTON__ICON',
});
}
});

```

Le résultat apparaît comme suit:

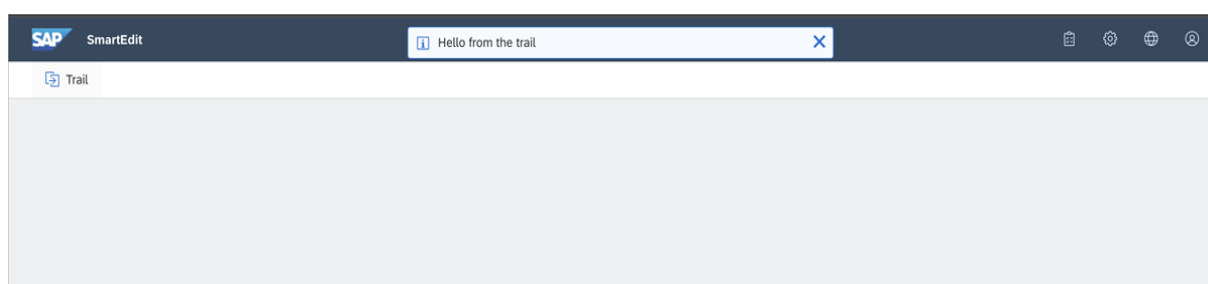


Figure 28 : toolbar Action

4.2.5 Ajout de fonctionnalités et de modes

Une fonctionnalité SmartEdit est tout élément arbitraire de fonctionnalité identifié par une clé qui peut être activée ou désactivée. Une fonction peut être aussi simple qu'un bouton de barre d'outils ou aussi complexe que la fonction glisser-déposer. La taille et la portée d'une fonctionnalité sont déterminées par le développeur qui la crée.

Si toutes les fonctionnalités SmartEdit sont activées en même temps, l'écran peut devenir encombré. Pour garder l'interface propre, SmartEdit a introduit des modes, qui utilisent le service Perspective. Un mode est un regroupement de fonctionnalités, identifié une clé. Lorsqu'un utilisateur sélectionne un mode, toutes les fonctionnalités du mode sont activées. Lorsqu'un utilisateur sélectionne un autre mode, toutes les fonctionnalités du mode précédemment sélectionné sont désactivées.

4.2.5.1 Ajout d'un mode

Ajouter le fichier legacytrail.ts

```

PERSPECTIVESERVICE . REGISTER ({
    KEY: 'TRAILPERSPECTIVE',

```

```
NAME/18NKEY: 'TRAIL MODE',
DESCRIPTION/18NKEY: 'AB.ANALYTICS.PERSPECTIVE.DESCRPTION',
FONCTIONNALITÉS: ['TRAILDECORATOR', 'TRAIL.CONTEXTUAL.BUTTON',
'SE.CONTEXTUALMENU'],
PERSPECTIVES: []});
```

Le code crée un mode appelé trailPerspective. Le mode Trail est le nom qui apparaît dans le sélecteur de menu.

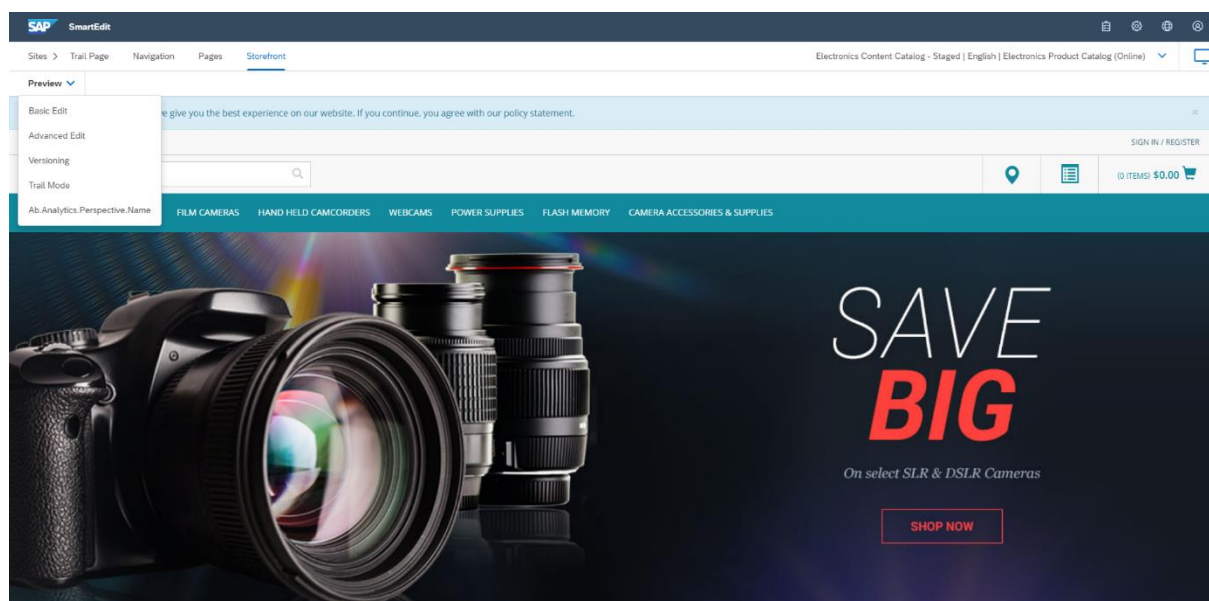


Figure 29 : Mode personnalisé

4.2.5.2 Ajout d'un decorateur

Un décorateur est une fonctionnalité qui ajoute du balisage et du comportement aux emplacements et aux composants. Pour éviter d'affecter la vitrine d'origine, on crée le décorateur dans une superposition. Toutes les modifications sont limitées à la superposition et on peut facilement supprimer la superposition.

1. Ajouter le fichier trailDecorator.ts

```
IMPORT * AS ANGULAR FROM 'ANGULAR';

CLASS TRAILDECORATORCONTROLLER {
  $ONINIT() {
    // SOME FANCY INIT CODE HERE
  }
}
```

```
// THIS CREATES AN ANGULAR MODULE CALLED TRAILDECORATORMODULE.
EXPORT CONST TRAILMODULE = ANGULAR.MODULE('TRAILDECORATORMODULE', [])

    // THIS ANGULAR RECIPE CREATES A COMPONENT CALLED TRAILDECORATOR.
    .COMPONENT('TRAILDECORATOR', {
        // THIS SPECIFIES THE FILE THAT CONTAINS THE MARKUP USED TO
        RENDER THE DECORATOR.
        TEMPLATEURL: 'TRAILDECORATORTEMPLATE.HTML',
        // THIS PROPERTY SPECIFIES THE ATTRIBUTES PROVIDED AS INPUTS TO
        THE COMPONENT.
        BINDINGS: {
            COMPONENTATTRIBUTES: '<'
        },

        // THIS PROPERTY SPECIFIES THE CONTROLLER THAT SPECIFIES THE
        BEHAVIOR OF THE DECORATOR. IT REFERENCES THE CLASS
        // CREATED ABOVE.
        CONTROLLER: TRAILDECORATORCONTROLLER
    });
```

2. Ajouter trailDecoratorTemplate

```
<DIV>
  <DIV CLASS="ROW TRAIL-DECORATOR">
    <SPAN STYLE="BACKGROUND-COLOR: #FFFF00; PADDING: 2EM">
      {{$CTRL.COMPONENTATTRIBUTES.SMARTEDITCOMPONENTID}}
    </SPAN>
  </DIV>
</DIV>
```

3 .Modifier le fichier legacysmartedittrail.ts

```
...

IMPORT {TRAILMODULE} FROM './TRAILDECORATOR/TRAILDECORATOR';

@SEMODULE({
  IMPORTS: [
    ...
    TRAILMODULE.NAME
  ],
  INITIALIZE: (DECORATORSERVICE: ANY, FEATURESERVICE: IFEATURESERVICE,
    PERSPECTIVESERVICE: IPERSPECTIVESERVICE) =>
  { // PARAMETERS ARE INJECTED FACTORY METHODS
    'NGINJECT';

    // THIS CODE SPECIFIES THE MAPPING THAT SPECIFIES THE COMPONENTS THAT
    WILL GET THIS DECORATOR. IN THIS CASE, ONLY
```

```

// SIMPLERESPONSIVEBANNERCOMPONENTS RECEIVE THE DECORATOR.
DECORATORSERVICE.ADDMAPPINGS({
  SIMPLERESPONSIVEBANNERCOMPONENT: ['TRAILDECORATOR']
});

// THIS CODE CREATES A FEATURE FOR THE DECORATOR. THAT WAY THE
DECORATOR CAN BE ENABLED OR DISABLED.
FEATURESERVICE.ADDDECORATOR({
  KEY: 'TRAILDECORATOR',
  NAME|18NKEY: 'TRAILDECORATOR.FEATURE.NAME'
});

// THE FEATURE IS ADDED TO A MODE CALLED ABANALYTICSPERSPECTIVE.
PERSPECTIVESERVICE.REGISTER({
  KEY: 'ABANALYTICSPERSPECTIVE',
  NAME|18NKEY: 'AB.ANALYTICS.PERSPECTIVE.NAME',
  DESCRIPTION|18NKEY: 'AB.ANALYTICS.PERSPECTIVE.DESCRPTION',
  FEATURES: ['TRAILDECORATOR'],
  PERSPECTIVES: []
});
}
}

EXPORT CLASS SMARTEDITTRAIL {}

```

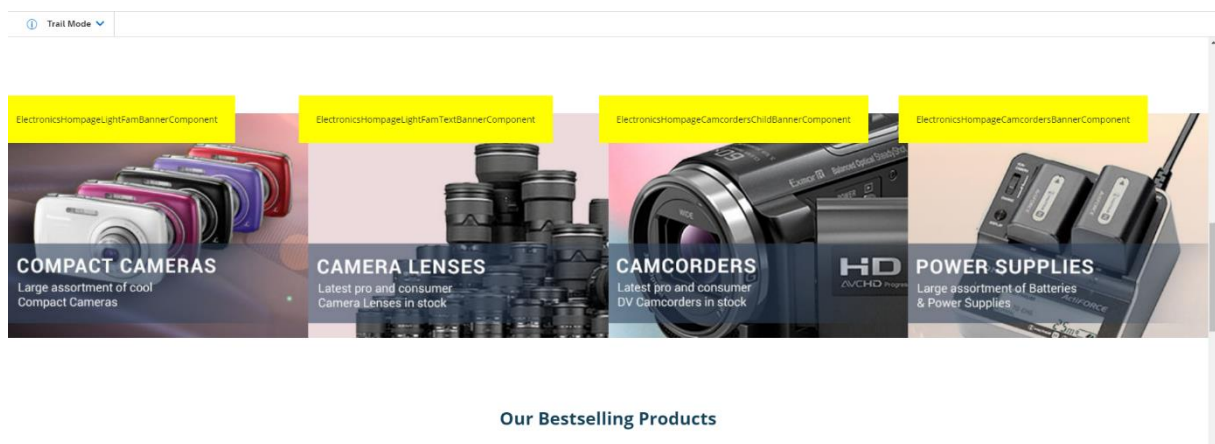


Figure 30 : Decorator personnalise

4.2.5.3 Ajout d'un bouton au menu contextuel

Le menu contextuel est un décorateur fourni dans SmartEdit prêt à l'emploi. Il fournit certains des outils de base pour l'édition de composants, tels que l'édition, le déplacement et la suppression.

Modification trailPage.ts

```
...

// THIS PIECE OF CODE CREATES A NEW FEATURE THAT ADDS A NEW BUTTON TO THE
CONTEXTUAL MENU.

FEATURESERVICE.ADDCONTEXTUALMENUBUTTON({

    KEY: 'TRAIL.CONTEXTUAL.BUTTON',

    PRIORITY: 500,

    I18NKEY: 'SE.CMS.CONTEXTMENU.TITLE.REMOVE',

    NAMEI18NKEY: 'SE.CMS.CONTEXTMENU.TITLE.REMOVE',

    // THIS SPECIFIES THE MAPPING; IT DETERMINES WHICH COMPONENTS WILL GET
    THIS BUTTON ADDED TO THEIR CONTEXTUAL MENUS.

    // IN THIS CASE, THE BUTTON WILL BE ADDED TO ALL COMPONENTS THAT ARE
    NOT SLOTS.

    REGEXPKEYS: ['^((?!SLOT).)*$'],

    ACTION: {

        // THIS SPECIFIES THE CALLBACK FUNCTION THAT WILL BE CALLED WHEN
        THE BUTTON IS CLICKED.

        // IN THIS CASE IT WILL SHOW AN ALERT WITH THE MESSAGE "HELLO
        FROM THE TRAIL".

        CALLBACK: () => {

            ALERTSERVICE.SHOWINFO("HELLO FROM THE TRAIL");

        }

    },

    DISPLAYCLASS: 'EXTERNALCOMPONENTBUTTON',

    DISPLAYICONCLASS: 'HYICON HYICON-GLOBE',

    DISPLAYSMALLICONCLASS: 'HYICON HYICON-GLOBE'

});

...

PERSPECTIVESERVICE.REGISTER({
```



```

...
FEATURES:      ['TRAILDECORATOR',      'TRAIL.CONTEXTUAL.BUTTON',
'SE.CONTEXTUALMENU'],

PERSPECTIVES: []

});

```

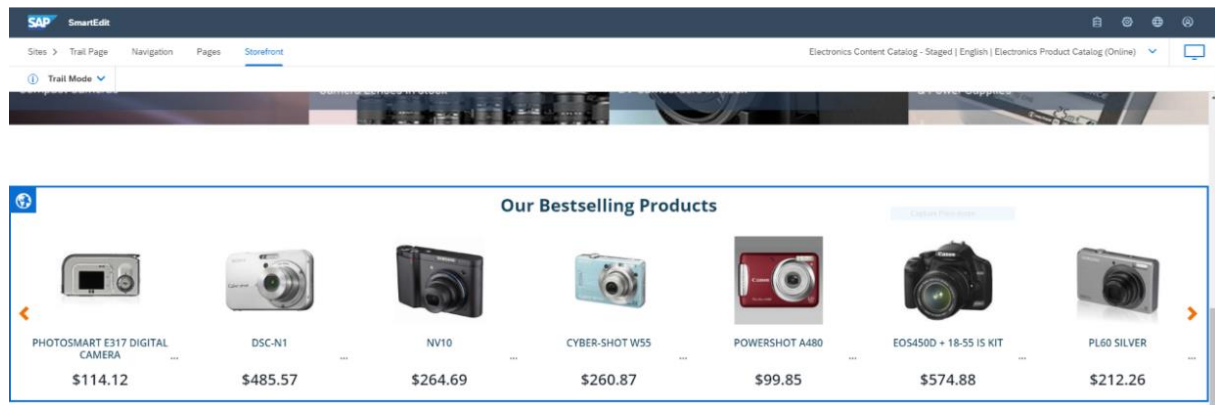


Figure 31 : bouton personnalis  au menu contextuel

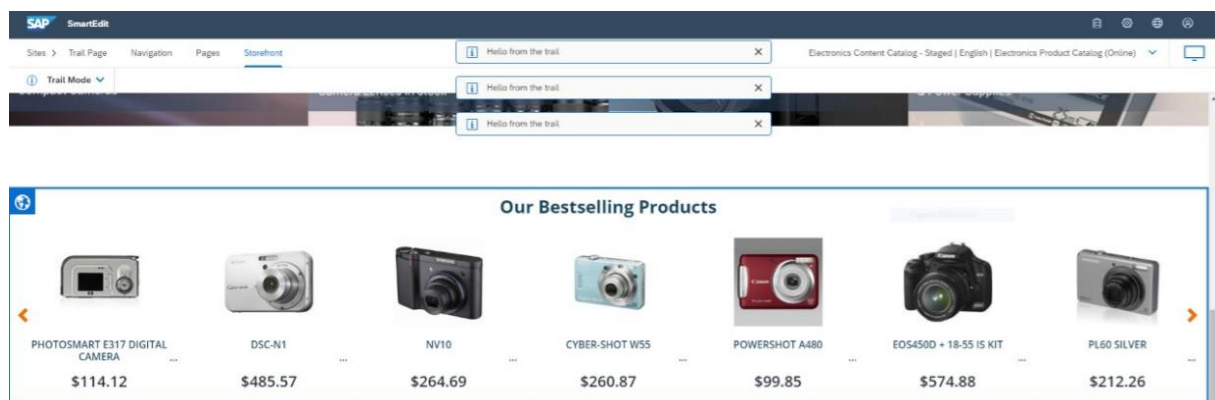


Figure 32 : bouton contextuel action

4.2.6 Cr ation d'un type de composant personnalis 

Pour cr er un composant personnalis  que nous pouvons modifier   l'aide de SmartEdit, nous devons  tendre l'API de structure CMS. Pour la plupart des types de composants que nous cr ons, nous n'aurons qu'   tendre l'API CMS Structure. L'API CMS Item fournit des convertisseurs de contenu g n riques, des populateurs et des validateurs pour prendre en charge la plupart des types de composants personnalis s. De m me, l' diteur g n rique fournit un ensemble de types de champs par d faut pour prendre en charge les types de composants personnalis s. Cependant, si vous cr ez un composant personnalis  avec des fonctionnalit s

spécialisées, vous devrez peut-être étendre l'API d'article CMS et ajouter des types de champs personnalisés à l'éditeur générique.

Extension de l'API CMS Structure

L'API CMS Structure fournit des informations sur les types de composants et leur structure. Pour ajouter un type de composant à SmartEdit, nous devons l'enregistrer dans l'API CMS Structure. On ajoute un type de composant nommé TrainingComponent à l'API.

Pour ajouter le TrainingComponent à l'API de structure CMS, on a procédé comme suit:

1. Enregistrer le nouveau type de composant, TrainingComponent, dans l'extension smarteditrail en l'ajoutant au fichier smarteditrail-items.xml.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!--
  Copyright (c) 2019 SAP SE or an SAP affiliate company. All rights reserved.
-->
<!-- ATTENTION: This is just an example file. You have to edit it according
  to your needs. -->

<items xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="items.xsd">
  <itemtypes>
    <itemtype code="TrainingComponent" generate="true" extends="SimpleCMSComponent" autocreate="true"
      javaClass="org.smarteditrail.jalo.TrainingComponent">
      <attributes>
        <attribute qualifier="customMessage" type="localized:java.lang.String">
          <persistence type="property"/>
        </attribute>
        <attribute qualifier="trainingImage" type="localized:Media" >
          <persistence type="property" />
        </attribute>
        <attribute qualifier="blacklistedMessage" type="localized:java.lang.String">
          <persistence type="property"/>
        </attribute>
        <attribute qualifier="numMessages" type="java.lang.Integer">
          <persistence type="property"/>
        </attribute>
      </attributes>
    </itemtype>
  </itemtypes>
</items>
```

Figure 33 : items.xml

Le nouveau composant est automatiquement enregistré dans l'API de structure CMS afin qu'il soit fourni dans la liste des types de composants lors de l'appel de l'API REST.

2. Pour faire le SimpleCMSComponent classe disponible pour votre type TrainingComponent, ajoutez l'extension cms2 au fichier extensioninfo.xml de l'extension smarteditrail :

```
<requires-extension name="cms2"/>
```

3. Ajouter les noms et descriptions d'attributs localisés pour le type de composant TrainingComponent en anglais au fichier smartedittrail / ressources / localization / smartedittrail- locales_en.properties. S'il existe d'autres fichiers de localisation, vous devez également ajouter les noms et descriptions d'attributs localisés pour le type de composant à ces fichiers.

```
smartedittrail.type.TrainingComponent.name=My Training Component  
smartedittrail.type.TrainingComponent.customMessage.name=Custom Message  
smartedittrail.type.TrainingComponent.blacklistedMessage.name=Blacklisted Message  
smartedittrail.type.TrainingComponent.numMessages.name=Num Messages to Show
```

4. Pour afficher le nouveau composant dans la vitrine, on a créé un fichier jsp portant le même nom que le composant (dans notre cas, c'est trainingcomponent.jsp) et enregistrer le fichier jsp dans le <HYBRIS_BIN_DIR> / modules / base-accelerator / yacceleratorstorefront / web / webroot / WEB-INF / views / responsive / cms / folder. L'extrait de code suivant provient du fichier trainingcomponent.jsp :

```
<div class="content" style="position: relative">  
    
  <div style="position: absolute; top: 50%; left: 30%">  
    ${component.customMessage}<br/>  
    ${component.numMessages}  
  </div>  
</div>
```

Figure 34 : TrainingComponent.jsp

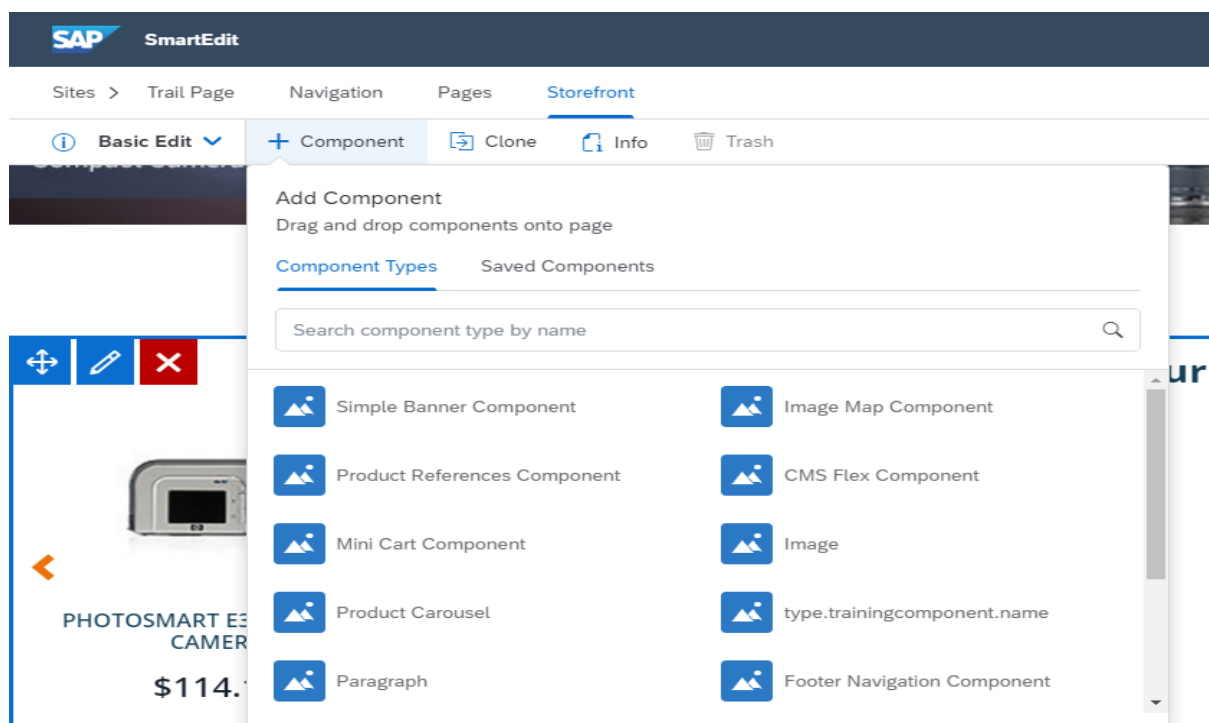


Figure 35 : nouveau Composant personnalis  dans SmartEdit

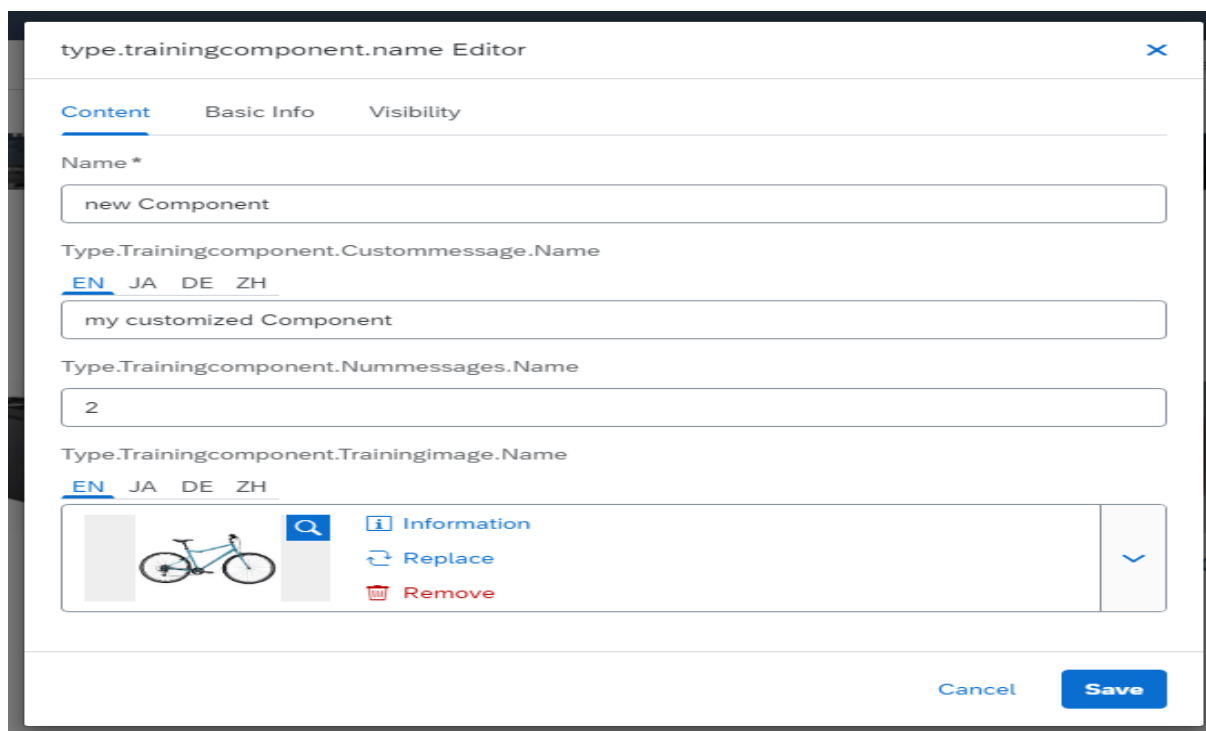


Figure 36 : Remplir le composant



Figure 37 : L'affichage de ce nouveau Composant

4.3 Conclusion

On peut personnaliser les services et fonctionnalités en ajoutant ou en remplaçant le code OOTB par le vôtre. Comme point de départ, créez votre extension à l'aide du modèle ysmarteditmodule. Il est très probable que le code exploite les services intégrés, et les capacités intégrées d'AngularJS. Ainsi, la compréhension des détails des deux est cruciale pour le succès

Conclusion générale

SmartEdit est un module SAP Commerce relativement nouveau livré avec la version 6.0. Le module fournit un outil pour gérer le contenu Web en plus du cockpit WCMS. Les utilisateurs professionnels peuvent désormais bénéficier d'une manière intuitive et conviviale pour créer et gérer des pages et des composants.

Dans la dernière version, Smartedit semble être un outil mature de gestion de contenu. le produit est suffisamment bon pour la majorité des besoins de gestion de contenu de commerce électronique. Il est formidable que tous les formulaires soient créés automatiquement en fonction des définitions de modèle de plate-forme. Contrairement au framework Cockpit, le code de Smartedit est extensible. L'architecture est claire, le code source est disponible, les composants du système sont découplés. Tout cela facilite la gestion et la personnalisation par rapport à ce que nous avons dans le cockpit WCMS hérité.

SmartEdit est le nouveau domaine d'investissement pour l'expérience client, ce qui signifie qu'aucune nouvelle fonctionnalité ne sera développée dans le cockpit WCMS. certaines des nouvelles fonctionnalités ont été considérablement améliorées dans SmartEdit, et elles ne seront pas mises à jour dans WCMS.

BIBLIOGRAPHIE

- [1] Scott. <https://www.scott-sports.com/global/en>
- [2] Ant. <https://ant.apache.org/>
- [3] SQLI Carriers. <https://www.sqli-carrieres.com/>
- [4] SQLI. <https://www.sqli.com/>
- [5] Spring Documentation. <https://docs.spring.io/spring/docs/current/spring-framework-reference/>
- [6] Git. <https://git-scm.com/>
- [7] E-Commerce guide. <https://ecommerceguide.com/guides/>
- [8] Wiki hybris. <https://cxwiki.sap.com/>
- [9] Stackextend. <https://www.stackextend.com/category/hybris/>
- [10] SAP Community. <https://community.sap.com/>
- [11] StackOverflow. <https://stackoverflow.com/>

Annexes

Annexe 1 : Le commerce électronique

• Définition du concept du e-commerce

Le e-commerce est l'abréviation de "électronique commerce" ou "commerce électronique" c'est un terme qui désigne les opérations d'achat/vente sur Internet, qui sont liés à des activités commerciales entre entreprises qu'on appelle le B2B (Business to Business), ou entre entreprises et particuliers, le B2C (Business to Customer). Une plateforme e-commerce offre un site sur lequel le marchand expose ses produits à vendre exactement comme dans le monde réel Avec une telle plateforme l'entreprise peut satisfaire les besoins du client 24 heures sur 24 et 7 jours sur 7, et avec un coût plus réduit, en plus ce site permet de présenter la société ainsi que l'ensemble des produits et services qu'elle offre. Parmi les principaux objectifs d'une plateforme e-commerce on peut citer :

- Analyser et comprendre le comportement des clients pour mieux les servir.
- Simplifier à l'utilisateur de surfer sur un site e-commerce en lui offrant le plus d'options.
- Suggérer à l'utilisateur des produits et l'orienter vers d'autres produits.
- Organiser l'information et adopter la bonne stratégie marketing.

• Les avantages du e-commerce

Pourquoi vendre sur internet ? Les prévisions concernant le chiffre d'affaires générées par le commerce sur Internet par le grand public : regroupant le B2C et le B2B, reflètent bien l'ampleur du commerce électronique. Au niveau mondial, le e-commerce représenterait presque deux trillion (1915 milliards) de dollars en 216 selon e-Marketer.

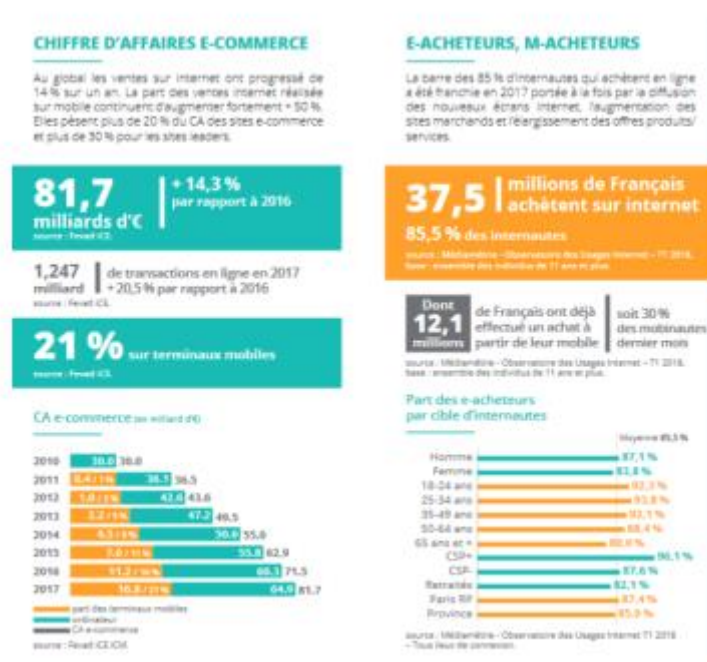


Figure 38 : Evolution du chiffre d'affaire de l'E-commerce mondial

CommerceElectronique connaît un tel succès auprès des cyberconsommateurs puisqu'il offre les avantages suivants :

- Une publicité permanente et détaillée : les moteurs et annuaires de recherche d'Internet fonctionnent comme les annuaires spécialisés ou "pages jaunes" traditionnels. Et donnent donc la possibilité, à de nombreux internautes, de découvrir votre entreprise sans même connaître à l'avance le nom de votre société.
- **Disponibilité 24h/24 et 365 jours par an** : l'Internet n'est jamais fermé, et il en va donc normalement de même pour tout site présent sur le Net. Vos ventes peuvent donc ainsi être considérablement augmentées du jour au lendemain.
- **Clientèle mondiale** : plus de 100 millions de personnes dans le Monde ont accès à Internet, et ce chiffre croît très rapidement. Il est certain que les clients que vous souhaitez toucher utilisent ou utiliseront Internet. Mais il est aussi certain que vous pouvez toucher de la même façon des clients sur toute la Planète. Vous avez donc une importante possibilité d'augmenter vos ventes.

- **Compétitivité** : internet est le moyen idéal pour trouver de nouveaux clients avec peu d'argent. Cependant il devient aussi urgent d'être présent sur le Net, car dans le cas contraire, vos clients actuels risquent de faire partie des nouveaux clients Internet de vos concurrents.

- **Investissement garantie** : en fonction du nombre potentiel de visiteurs, le coût de réalisation et de maintenance du site est quasi nul. Pour le prix d'un quart de page dans un magazine national vous pouvez avoir sur Internet de nombreuses pages avec des photos en couleurs. La gratuité du courrier électronique est aussi attractive et les possibilités sans limites.

- **Les inconvénients du e-commerce**

Il existe néanmoins trois freins importants à l'essor du e-commerce :

- **La livraison** : la livraison est un problème mineur, parce que les solutions existent et parce que la logistique s'adapte très rapidement aux nouvelles données du marché. La plupart des offres incluent désormais ou vont inclure une livraison gratuite ou à petit prix forfaitaire. C'est un premier point à respecter pour développer ses ventes sur un site de commerce électronique.

- **La sécurité de paiement** : la transaction en ligne, consiste à envoyer des informations confidentielles sur internet, ce qui pose un problème d'insécurité de paiement en ligne.

- **La peur des internautes d'acheter sur Internet** : Il y a une certaine peur quand on achète sur internet, surtout quand on achète pour la première fois. Cette peur de tomber sur des vendeurs malhonnêtes ou encore cette insécurité de paiement font du commerce électronique un mode d'achat compliqué. Il n'existe aucune garantie et aucun service après-vente pour les produits acheter. Beaucoup de consommateur sont victimes d'arnaque il achète un produit, paye mais ne reçoivent jamais leurs produits.

- **les niveaux du e-commerce**

Il est à savoir que lorsque nous parlons de l'e-commerce, on distingue 3 niveaux de complexités, nous devons choisir donc le niveau qu'on veut atteindre, sachant qu'on peut faire évoluer le commerce électronique progressivement.

Niveau 1 : Catalogue en ligne : Le premier niveau consiste tout simplement à placer sur le web une sorte de catalogue plus ou moins complet des produits, et services.

Niveau 2 : Prise de commande en ligne : On propose aux clients de passer leurs commandes directement sur le web. Ils consultent le catalogue des produits, en ajoutent à tout moment au

panier virtuel, lorsqu'ils finissent, ils envoient la commande avec les informations nécessaires à l'Administrateur de la boutique afin qu'il valide ou annule la commande.

Niveau 3 : Transaction bancaire sécurisée : A ce niveau, le client choisit les produits, les commande et peut les payer par Internet. Dans ce cas le client donne son numéro de carte bancaire et sa date d'expiration, le serveur se met alors en relation avec l'organisme qui gère les cartes bancaires et tente d'effectuer la transaction. En fonction de l'échec ou du succès de la transaction le serveur valide la commande ou informe le client du problème.

Annexe 2 : SAP Hybris

- **Architecture générale d'hybris**

La plateforme hybris est conçue pour s'adapter à toute entreprise voulant faire du e-commerce multicanal, la plateforme peut s'interfacer avec plusieurs clients frontaux mobile et desktop et supporte plusieurs technologies phare du marché, la plateforme est conçue autour d'une architecture en 6 couches la première couche rassemble les clients frontaux et les interfaces systèmes front-end et back-end (couche 1) ainsi que le Système de Gestion des Bases de Données Relationnelles (couche 6), ils sont marqués avec la couleur bleue dans le schéma d'architecture. La couche de service offre plusieurs types de services métiers, ce sont des services de haut niveau indispensable pour le métier du e-commerce (couche 2), aussi il y'a des services dite d'infrastructure (couche3) qui assure les opérations essentiels dans tout système (service de cache, sécurité, transaction, session,...) , alors que la couche du Framework (couche 4) offre les interfaces API qui sont implémentées par les services déjà mentionnés ce qui laisse place à toute sorte d'adaptation (customisation) de la plateforme par la ré-implémentation de ces interfaces, les couches 2,3,4 et 5 sont mentionnés en orange dans le schéma ci-dessous. La plateforme est compatible avec les serveurs d'applications Hybris server, Spring Server, Web Logic et tomcat, pour la gestion de persistance des données la plateforme supporte plusieurs solution standardset phare du marché, nous pouvons utiliser une base de données Oracle, SQL server comme on peut utiliser MySQL, et tous avec unegestion intégrée de cache offerte par la couche persistance d'Hybris (couche 5).

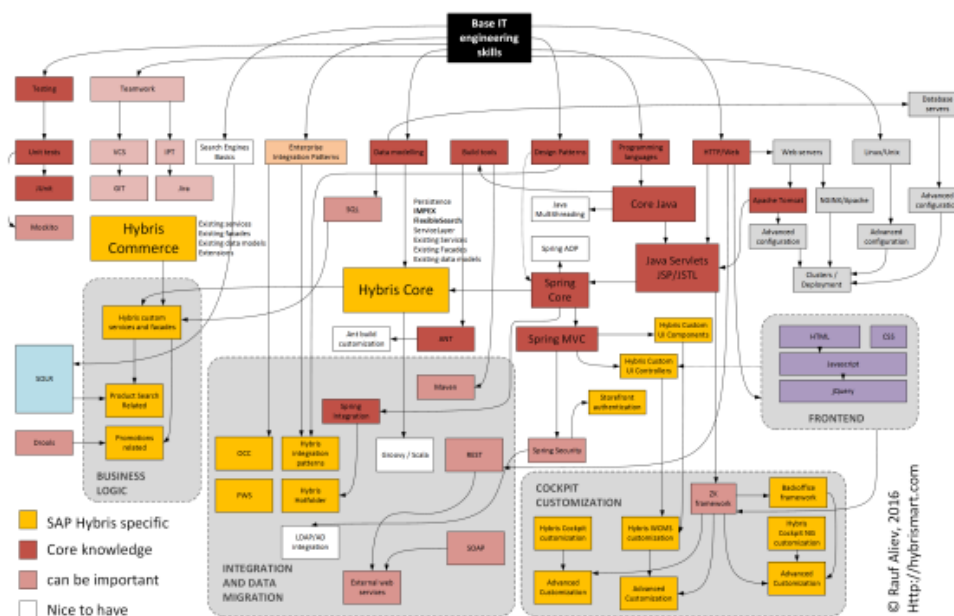


Figure 39 : Vue globale sur l'architecture d'Hybris

• Les Composant de la plateforme Hybris

D'un point de vue commercial, Hybris est divisé en modules parmi les plus importants on cite : **L'extension smartEdit** : ce composant permet de gérer d'une façon très intuitive les informations marketing d'un produit sans une dépendance aux webmasters et aux équipes IT, son interface est accessible depuis le lien suivant : <http://localhost:9002/smartedit>.

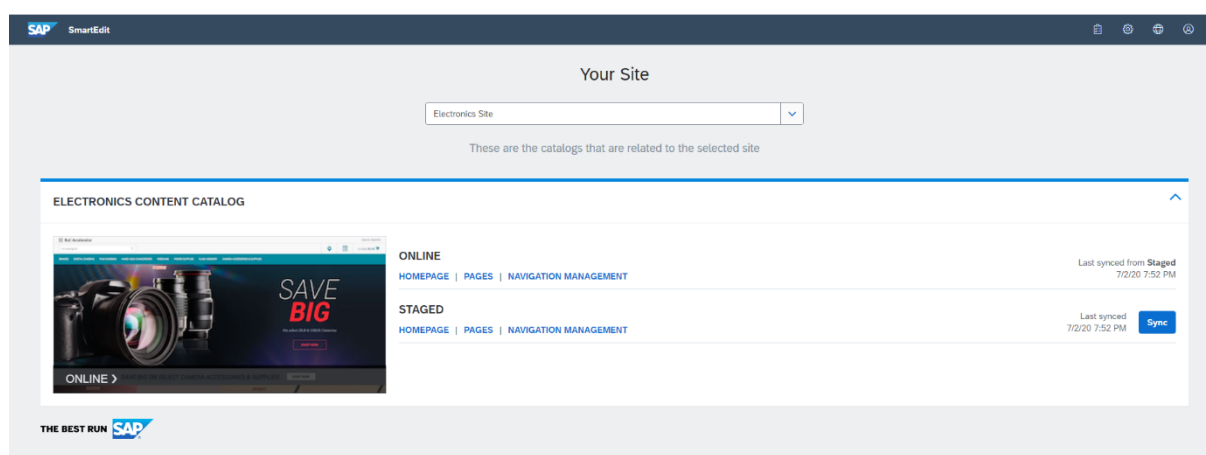


Figure 40 : SmartEdit – Hybris

HAC : c'est une interface backend du système, il permet d'effectuer des opérations d'administrations comme la supervision des performances système, la modification à chaud des paramètres généraux, l'administration du cache système, de session, d'accès aux logs etc., il est accessible depuis le lien [http : //localhost :9001/hac](http://localhost:9001/hac).

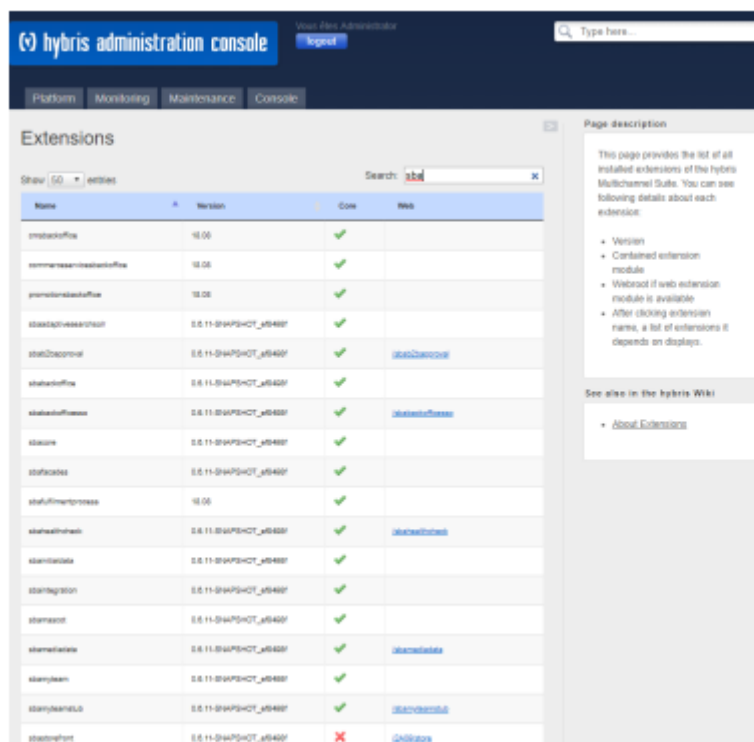


Figure 41 : HAC -Hybris

Backoffice : accessible depuis [http : // localhost : 9001 / backoffice / login.zul](http://localhost:9001/backoffice/login.zul) (sur une installation locale) est l'application de backoffice Hybris de nouvelle génération, le backoffice a remplacé le cockpit de la console HMC (ancienne version du backoffice), il offre plus de souplesse que l'ancien cockpit cadre et une interface plus agréable. Points clés de l'application de backoffice :

- C'est hautement personnalisable (xml).
- Il affiche le bon mélange d'applications en fonction de son rôle professionnel.
- Tout peut être fait à partir de l'application de backoffice (une fois connecté en tant qu'administrateur).
- Le modèle d'extension ybackoffice est utilisé pour générer une extension personnalisée dans laquelle vous implémentez vos propres composants. Ces composants sont ensuite réutilisables dans "l'Application Orchestrateur" et dans "l'application Web" de l'extension backoffice.