

Registre des rapports de réunion

Théo DELMAS
Lauric TEYSSEYRE
Pierre-Louis RENON
Julien WATTIER

Université Paul Sabatier
Master Informatique 1

15 avril 2023

Table des matières

1	Objectif du document	3
2	Revue de sprint du 21/02/2023	3
2.1	Présentation du travail effectué	3
2.1.1	Fallback de l'API Rest de documentation	3
2.1.2	Réécriture de domaine dans les documentations	3
2.1.3	Affichage de la documentation dans la popover	3
2.1.4	Interpolation dans les documentations	4
2.2	Présentation du travail prévu	4
2.2.1	Système de recherche	4
2.2.2	Extension de la détection de documentation invalide . . .	4
2.3	Discussion	4
2.3.1	Migration du frontend	4
3	Revue de sprint du 14/03/2023	5
3.1	Présentation du travail effectué	5
3.1.1	Synchronisation des interpolations	5
3.1.2	Révision de l'afficheur de paramètre	5
3.2	Présentation du travail prévu	5
3.2.1	Système de recherche	5
3.3	Discussion	5
3.3.1	Internationalisation	5
3.3.2	Nouvelles documentations	6
3.3.3	Système de recherche	6
3.3.4	Tests	6
4	Revue de sprint du 29/03/2023	6
4.1	Présentation du travail effectué	6
4.1.1	Affichage des tableaux	6
4.1.2	Retour à la ligne pour l'affichage par défaut tous	6
4.2	Présentation du travail prévu	7
4.2.1	Affiner le CSS	7
4.3	Discussion	7
4.3.1	Abandon de la partie recherche	7
4.3.2	Demande de revue de code	7
5	Revue de sprint du 10/04/2023	7
5.1	Présentation du travail effectué	7
5.1.1	Afficheur de paramètre	7
5.2	Présentation du travail prévu	7
5.2.1	Création d'un tutoriel de création de documentation . . .	7

1 Objectif du document

Ce document référence l'ensemble des rapports de réunion officielle qu'il y a eu durant ce projet.

2 Revue de sprint du 21/02/2023

2.1 Présentation du travail effectué

2.1.1 Fallback de l'API Rest de documentation

Le système qui parse l'URL n'est pas satisfaisant selon le client.

→ Organisation d'une réunion sur ce point nécessaire.

→ Étude d'alternative, existence d'un tel système sous Django ?

2.1.2 Réécriture de domaine dans les documentations

La syntaxe pour appeler un remplacement de nom de domaine n'est pas satisfaisante. Les collaborateurs doivent pouvoir déposer leur documentation dans le serveur et lors de la visualisation sans passer par le frontend, ils doivent pouvoir charger correctement les images. Cela implique qu'ils écrivent les liens en relatif.

→ Le système doit réécrire lorsque des chemins relatifs sont identifiés.

Le client valide le reste du système mais estime qu'il n'est pas suffisamment satisfaisant sans identifier dans l'immédiat d'alternative.

→ Organisation d'une réunion sur ce point nécessaire.

→ Étude d'alternative.

2.1.3 Affichage de la documentation dans la popover

Le client demande de centrer l'écran sur le popover lors de son ouverture.

→ Mettre en place ce système.

Il propose d'inciter les collaborateurs à écrire les documentations en format paysage.

→ Définir des conseils d'écriture des documentations.

Le client va écrire des documentations typique.

2.1.4 Interpolation dans les documentations

Le client nous a indiqué que leur système de contraintes à une convention qui est que lorsqu'un paramètre est présent mais pour lequel aucun objet n'est pointé alors cela veut dire que le paramètre pointe vers toutes les instances de ce paramètre.

Dans ce genre de cas le client demande que l'interpolation ne liste pas toutes les instances du paramètre mais affiche un mot désignant toutes les instances.

→ Mettre en place ce système

2.2 Présentation du travail prévu

2.2.1 Système de recherche

Nous avons proposé d'ajouter des tags sur les contraintes pour les utiliser lors de la recherche de contrainte.

→ Système validé

Le client nous a demandé également d'utiliser un système de recherche de correspondance qui aurait pioché dans les documentations ou dans des champs de description de contrainte qui seront éventuellement ajoutés.

→ Voir si un tel système de recherche open source existe.

2.2.2 Extension de la détection de documentation invalide

Nous avons proposé d'étendre ce système en détectant au démarrage des documentations qui demanderaient des images non hostées ou des interpolations impossibles.

Le client n'étant pas familier avec la CI il nous a indiqué de ne pas aller trop loin dans le mécanisme détection des documentations invalides.

→ Laisser tomber cette extension si elle devient trop complexe.

2.3 Discussion

2.3.1 Migration du frontend

Une rapide discussion a eu lieu sur le portage du front de l'application vers VueJS. En l'état le serveur Django le gère au travers de template. Léo Cusseau est en charge de trouver un moyen d'initier le portage en faisant cohabiter les templates Django avec une application VueJS.

Nous avons précisé que nous essayons de développer les fonctionnalités afin qu'elles fonctionnent dans le cadre de cette cohabitation et que lors du portage complet vers VueJS il soit aisé de les réadapter.

→ Prévoir une réunion avec Léo Cusseau pour comprendre comment il prévoit la cohabitation puis le portage complet afin d'adapter au mieux ce que nous développons à ce contexte.

3 Revue de sprint du 14/03/2023

3.1 Présentation du travail effectué

3.1.1 Synchronisation des interpolations

Le client a demandé pourquoi ces stores utilisaient un délai pour se charger. Nous avons précisé que ce délai venait du fait qu'on copiait un objet chargé de manière asynchrone et nous ne pouvions pas nous synchroniser sur ce chargement. De fait nous avons mis en place un délai arbitraire pour lancer la copie. D'autre part nous avons précisé qu'à terme, nous ne copierons plus un objet mais nous utiliserons directement l'API Rest existante sur laquelle nous pourrions nous synchroniser parfaitement.

→ Passer en chargement par requête à l'API.

3.1.2 Révision de l'afficheur de paramètre

Le client nous a fait remarquer que l'usage de balise <p> pour chaque instance n'était pas idéal car créant un saut de ligne à l'affichage.

→ Utiliser une autre balise qui ne crée pas de saut de ligne.

3.2 Présentation du travail prévu

3.2.1 Système de recherche

Pour le système de recherche de correspondance le client a affiné son besoin en précisant que le système de recherche devrait chercher les tags ou les mots-clefs dans une partie de la documentation et pas dans les classes de contraintes.

→ Prévoir le système pour qu'il recherche dans la documentation et donc être indépendant des objets contraintes du serveur.

Nous avons également proposé d'implémenter ce système côté serveur. Le client était inquiet que le système ne soit pas assez réactif à cause des temps de communication ou que trop de requête soit émise si à chaque fois que l'utilisateur tape un caractère une nouvelle requête soit émise.

→ Prévoir un système temporise les requêtes tant que l'utilisateur tape quelque chose.

→ Faire des mesures de performance.

3.3 Discussion

3.3.1 Internationalisation

Le client nous a précisé que l'équipe commençait à voir pour un outil d'internationalisation des interfaces mais qu'en l'état nous ne devions pas l'utiliser mais plutôt avoir des interfaces en anglais. D'autre part il nous a demandé de bien utiliser l'anglais de manière générale pour les commentaires, nom de méthode et paramètre et cetera...

→ S'assurer que ce principe est bien appliqué.

3.3.2 Nouvelles documentations

Le client nous a demandé de tirer un ensemble de documentation qu'il venait de produire. Au moment de charger la documentation nous avons remarqué que le système ne prenait pas en charge l'affichage des tableaux. Cela venant du fait que le composant tiers pour afficher les documentations nécessite une option pour afficher les tableaux.

→ Régler VueShowdown pour afficher les tableaux.

Nous avons également remarqué que les paramètres primitifs n'étaient pas gérés par l'afficheur.

→ Gérer les paramètres primitifs dans le système.

3.3.3 Système de recherche

Le client a rappelé que cette partie du projet n'était pas prioritaire

→ Prioriser le travail sur l'affichage des documentations.

3.3.4 Tests

Le client a discuté du fait de tester les développements. Il nous a demandé de tester un maximum de choses. Cependant nous n'avons pas de véritable méthode pour tester notre production.

→ Définir comment tester les composants.

4 Revue de sprint du 29/03/2023

4.1 Présentation du travail effectué

4.1.1 Affichage des tableaux

Le client nous a demandé de voir si les tableaux affichés peuvent bien séparer les cases en affichant les lignes du tableau car sinon la documentation n'est pas claire.

→ Voir si VueShowdown peut afficher les lignes d'un tableau.

4.1.2 Retour à la ligne pour l'affichage par défaut tous

Nous avons remarqué lors de la présentation que l'afficheur de paramètre faisait un retour à la ligne lorsqu'il affichait la valeur par défaut "tous".

→ Changer la balise p par span.

4.2 Présentation du travail prévu

4.2.1 Affiner le CSS

Le client nous a demandé d'affiner le CSS afin d'obtenir une interface ergonomique

- Revoir la taille de la popover sur les documentations existantes.
- Revoir le CSS sur la création de documentation afin d'enlever le blanc inutile.
- Vérifier que la taille de la popover sur la création de documentation est convenable.

4.3 Discussion

4.3.1 Abandon de la partie recherche

Nous avons convenu avec le client que faute de temps nous ne pourrions commencer le système de recherche. Nous avons décidé de nous focaliser sur l'affichage de documentation.

- Annuler les tâches relatives au système de recherche.

4.3.2 Demande de revue de code

Le client nous a demandé de prévoir une revue de code avec des mainteneurs du projet afin de présenter en détail les différentes parties du code produit, ainsi que le fonctionnement afin de faciliter la reprise de notre travail.

- Prévoir une revue de code.

5 Revue de sprint du 10/04/2023

5.1 Présentation du travail effectué

5.1.1 Afficheur de paramètre

Le client a fait remarquer que pour la création et la modification de contrainte, le système n'étant pas capable d'interpoler les paramètres il devrait en afficher le nom plutôt que des blancs. -> Ajouter un affichage par défaut lorsque le paramètre n'est pas interpolable.

5.2 Présentation du travail prévu

5.2.1 Création d'un tutoriel de création de documentation

Le client nous a rappelé d'écrire un tutoriel à ajouter au wiki du projet qui explique comment écrire une contrainte pour le système développé. -> Réaliser le besoin dédié sur le wekan.