

Réunion sur les besoins techniques

27 Octobre 2023

Participants

Nom	Prénom	Groupe
Macadré	Bruno	Support Technique
Binginot	Etienne	1
Mazuy	Axelle	1
Sapin	Robin	2

Ordres du jour

- Présentation des besoins
- Conseils techniques et discussion des besoins
- Architecture possible

Table des matières

1	Présentation des besoins	3
1.1	Rappel de l'organisation du projet	3
1.2	Besoins exprimés par chaque groupe	3
2	Conseils techniques et discussion des besoins	4
2.1	Réponses aux besoins exprimés	4
2.1.1	GitLab	4
2.1.2	Intégration et déploiement continu	4
2.2	Besoins à étudier	4
3	Architecture possible	5
3.1	Technologies conseillées	5
3.2	Organisation des serveurs	5

1 Présentation des besoins

1.1 Rappel de l'organisation du projet

Le projet AGORA-1 V3 a pour objectif la refonte de la plateforme AGORA qui est présentée lors de la fête de la science. Pour ce faire, deux groupes ont été affectés à ce projet, chacun ayant une partie spécifique du projet à réaliser.

Il est toutefois à noter que les deux groupes doivent pouvoir échanger afin de créer une interface d'échange entre ces deux parties et mener à bien ce projet.

On peut donc résumer l'organisation du projet comme suit :

- Le groupe 1 s'occupe de la refonte de la partie jeux de la plateforme. Le but est de créer une bibliothèque de jeux modulable et facile à compléter.
- Le groupe 2 s'occupe quant à lui de la refonte de la partie plateforme application web afin de la rendre plus actuelle dans son code et ergonomique.
- Les deux groupes travaillent conjointement sur la refonte de l'interface reliant la partie jeux et la partie principale de l'application web, ce qui nécessite une communication des deux parties.

1.2 Besoins exprimés par chaque groupe

Lors de l'échange, les représentants de chaque groupe et leur accompagnant ont présenté ce qui serait nécessaire pour le travail sur les tâches présentées précédemment.

Le besoin le plus essentiel était l'accès au dépôt GitLab afin de pouvoir utiliser Git dans les échanges de code, mais aussi les autres fonctionnalités présentes qu'offre la plateforme DevOps.

Un autre point évoqué a été la mise à disposition de serveurs de tests lors de la phase de développement afin de mettre à l'épreuve le code, toutefois ceux-ci ne seront fournis que plus tard.

Enfin un dernier besoin évoqué a été un système de déploiement et intégration continus, par exemple avec l'utilisation de SonarQube.

2 Conseils techniques et discussion des besoins

2.1 Réponses aux besoins exprimés

2.1.1 GitLab

Un premier problème discuté a donc été les accès GitLab que l'on devait se voir confier. Il fallait tout d'abord déterminer de quelle façon les deux groupes souhaitaient travailler afin de voir si l'on allait travailler sur un ou deux GitLab séparés.

Chaque solution présentait ses avantages et inconvénients : travailler sur un même dépôt avec des branches distinctes, ou travailler sur deux dépôts indépendants.

La première solution présentait comme avantage la plus grande facilité à mettre en commun le travail de chaque groupe afin de créer l'interface les reliant. En revanche, cela impose une certaine rigueur pour l'affectation et le découpage des tâches (la partie DevOps du projet) afin de retrouver efficacement pour chaque groupe les tâches qui l'incombent. De même, il faut faire davantage attention aux actions sur le Git afin de respecter le découpage en branches distinctes.

La seconde solution présentait comme avantage la distinction efficace des tâches attribuées à chaque groupe. En revanche, il est plus difficile de mettre en commun le code, notamment pour la partie interface entre les deux parties du projet.

Pour ces raisons, il a été mis en avant que dans le cadre de ce projet, la première solution était la plus adaptée. Toutefois, si besoin il sera possible de changer d'organisation plus tard.

2.1.2 Intégration et déploiement continus

Nous avons discuté de l'utilisation d'un serveur SonarQube par groupe, afin de permettre un code cohérent entre les deux groupes, ainsi qu'un code de qualité.

Il nous reste à définir entre nous des différents outils d'intégration et déploiement continus que l'on utilisera dans le projet. Cela nous imposera une rigueur sur leur configuration et leur utilisation.

2.2 Besoins à étudier

Nous avons discuté de la possibilité de conteneuriser notre application, notamment avec l'utilisation de Docker. Toutefois, il n'est pas encore possible de déterminer s'il sera possible de faire cela. Nous étudierons cela lors de la phase

de modélisation de l'application. La conteneurisation favorisait toutefois l'utilisation d'un dépôt commun avec multi-branches.

Il nous faudra également faire attention au développement des sockets réseaux. En effet, il s'agit d'une difficulté récurrente à ce projet, les serveurs de l'application web ne pouvant pas tenir une charge importante. Il sera donc ce problème afin de trouver des solutions au problème et étudier les risques qui y sont liés.

3 Architecture possible

3.1 Technologies conseillées

Comme discuté lors de la réunion de lancement avec le client, la technologie Symfony est préconisée. De même un support technique est disponible plus largement sur ce framework particulièrement.

Nous avons également discuté des technologies front comme Angular et React. Il nous a été recommandé de nous baser sur les compétences intra-équipe pour choisir la technologie front qui nous conviendrait le mieux. Lors de cette discussion, nous avons pu relever que de manière générale sur ce projet, certains membres semblaient avoir plus de connaissances sur React que sur d'autres technologies front. Le choix reste toutefois à déterminer.

3.2 Organisation des serveurs

Lors de discussions nous avons pu réaliser un schéma d'organisation possible pour les serveurs :

- Un serveur de production commun aux deux groupes.
- Un serveur de pré-production par groupe.
- Un serveur de développement par groupe.

Il reste à déterminer si des serveurs doivent être ajoutés ou retirés après avoir choisi le mode de fonctionnement pour le déploiement et l'intégration continus.