

POST-MORTEM AREA

BENISTANT David | DENNI Robin | FRANK Oscar | LIEBENGUTH Alessio | SCHMELTZ Théotime

1. Mise en contexte

Le projet AREA, intégré dans le programme de troisième année du Programme Grande École à EPITECH, représente notre deuxième projet technique de l'année, mais surtout notre première expérience dans le domaine du développement web et mobile avec des attentes aussi élevées.

L'objectif de ce projet est de concevoir une application web, une application mobile, et une API. Cette application permettra aux utilisateurs de configurer des actions-réactions, appelées AREA, à travers une interface utilisateur graphique, en se connectant à divers services que nous avons choisi d'implémenter dans notre projet tels que Discord, Trello, Github, Asana, Google et Jira.

Le concept AREA se base sur l'automatisation de tâches inter-applications. Par exemple, une AREA pourrait être configurée de la manière suivante : à la création d'une carte dans Trello, un message privé est automatiquement envoyé sur Discord à l'utilisateur concerné, et simultanément, une carte est générée dans un tableau Jira.

L'objectif ultime est de réaliser ces opérations de manière fluide et automatisée, offrant une optimisation significative de la productivité personnelle et professionnelle de l'utilisateur.

Par ailleurs, étant donné que nous sommes une équipe de 5 étudiants, notre solution finale doit inclure au moins 6 services et comporter 15 actions/réactions pour répondre aux exigences du projet.

2. Choix des technologies utilisées

Dans le cadre du développement du projet AREA, nous avons procédé à une évaluation comparative des technologies potentielles afin de déterminer les langages les plus adaptés pour chaque composant de notre projet. Cette démarche stratégique visait à garantir la pertinence de nos choix technologiques.

Pour l'application web, nous avons mis en concurrence React.js, Angular, Vue.js et SvelteKit. Les critères de sélection comprenaient la compétence collective du groupe dans chaque technologie, la performance, l'ampleur de la communauté d'utilisateurs (notamment sur des plateformes comme Stack Overflow), la courbe d'apprentissage et l'espace occupé par les fichiers compilés. À travers une évaluation par tableau Excel, React.js s'est démarqué comme le choix optimal, bénéficiant de notre maîtrise collective, de performances comparables à ses concurrents et d'une vaste communauté.

Concernant le développement de l'application mobile, SwiftUI, Kotlin, React Native et Flutter ont été soumis à une analyse comparative. Nous avons pris en compte la maîtrise de l'équipe, les performances, la compatibilité multiplateforme, la taille de la communauté et la facilité d'apprentissage. React Native s'est avéré être le choix optimal en raison de notre familiarité avec la technologie et de ses capacités multiplateformes.

Pour l'API, une analyse a été effectuée sur ExpressJs, ASP.NET Core, Python Flask et Phoenix (Elixir). Les critères de sélection ont inclus la maîtrise par l'équipe, les performances, l'aisance d'utilisation, la présence de la communauté et l'adéquation de la technologie avec la taille prévue du projet. ExpressJs s'est distingué comme étant le plus conforme à nos exigences, offrant simplicité d'utilisation et adéquation avec les petits projets/microservices.

Enfin, pour la base de données, l'évaluation entre MySQL, PostgreSQL, MongoDB, DynamoDB et Redis s'est basée sur la maîtrise de l'équipe, les performances, la scalabilité, l'approche technologique (SQL ou NoSQL) et la communauté. DynamoDB a été retenu comme le candidat idéal, bénéficiant de nombreuses connaissances au sein de l'équipe, de la technologie NoSQL et d'une intégration simplifiée avec AWS.

Le choix des technologies constitue une étape cruciale dans la réalisation d'un tel projet. Nous avons navigué avec succès à travers cette phase grâce aux connaissances individuelles des membres du groupe et à la détermination de chacun. Bien que nous n'ayons pas rencontré de difficultés majeures, des recherches approfondies ont été nécessaires pour nous baser sur des statistiques réelles plutôt qu'approximatives.

3. Répartition des tâches

Afin de favoriser une ambiance motivante et collaborative, nous avons adopté une approche laissant à chaque membre la liberté de choisir sa mission principale, en fonction de ses compétences et de ses préférences. Cette stratégie s'est révélée fructueuse, conduisant naturellement à une distribution équitable des rôles au sein de l'équipe.

Dans le domaine du développement web, nous retrouvons Oscar FRANK et Alessio LIEBENGUTH, tandis que Robin DENNI et Théotime SCHMELTZ se consacrent au développement mobile. David BENISTANT quant à lui, a pris en charge l'API et la base de données.

Cette étape revêt une importance particulière, car elle assigne à chaque membre du groupe des tâches spécifiques. Chacun s'engage ainsi à donner le meilleur de lui-même pour accomplir sa mission avec succès, évitant ainsi toute source potentielle d'insatisfaction au sein de l'équipe. Nous sommes particulièrement satisfaits d'avoir complété cette étape rapidement, préservant ainsi un temps précieux et contribuant à l'épanouissement global de l'équipe. Cette répartition équilibrée des responsabilités a non seulement favorisé la collaboration fluide au sein du groupe, mais a également jeté les bases pour aborder avec succès les nouvelles tâches liées à l'intégration des services dans notre solution.

4. Choix des services

Comme mentionné précédemment, l'objectif principal de notre projet est de connecter diverses applications en utilisant des actions et des réactions. Pour sélectionner les applications à intégrer, nous avons organisé une réunion afin de discuter des différentes propositions. Il était essentiel que chaque membre du groupe contribue en intégrant au moins un service. Ce processus de sélection individuelle des services a contribué à la qualité globale de notre solution finale.

Le choix des services s'est déroulé de manière naturelle, chacun d'entre nous ayant une idée spécifique de service à intégrer dans notre solution. De plus, l'utilité évidente de ces services à être interconnectés a ajouté une couche de satisfaction supplémentaire au sein du groupe. La collaboration

et l'engagement de chaque membre dans le choix des services ont non seulement renforcé notre solution, mais également renforcé la cohésion de l'équipe.

5. Maquettage – UI/UX

Dans le souci de respecter les normes de design et d'accessibilité, nous avons pris l'initiative de concevoir des maquettes à la fois pour la version web et mobile de notre application. Cette démarche visait à garantir une cohérence de design entre les deux plateformes tout en observant attentivement les règles d'accessibilité, notamment en ce qui concerne la taille des boutons, la navigation sans souris pour la version web, et le respect de certains contrastes de couleur.

La création de maquettes a également fourni aux équipes de développement web et mobile une vision claire de la direction à suivre. Cela a évité le risque de devoir réfléchir au design pendant le processus de développement, réduisant ainsi le besoin de modifications majeures ultérieures. Les maquettes ont agi comme un guide visuel, permettant une meilleure compréhension des intentions de design et assurant une cohérence visuelle à travers toutes les étapes du développement.

En résumé, l'élaboration de maquettes a été une étape cruciale pour garantir la conformité aux normes de design et d'accessibilité, tout en facilitant le processus de développement en fournissant une direction claire et en minimisant les risques de modifications majeures en cours de route.

6. Développement de notre solution

Affectés chacun à notre mission principale, notre premier objectif était d'élaborer un Proof of Concept (POC). Motivés par nos tâches respectives, nous avons rapidement obtenu un premier échantillon avec une authentification fonctionnelle. Par la suite, nous avons poursuivi le développement de nos composants individuels pour finalement aborder la phase des actions et des réactions.

Initialement, chaque membre a développé son propre module de manière indépendante. Par la suite, nous avons fusionné l'ensemble pour lier le "front-end" avec le "back-end". Une fois cette intégration réalisée, nous avons résolu les derniers problèmes techniques pour parvenir à un produit fonctionnel.

Le développement de notre solution représente la première étape technique à laquelle nous avons été confrontés. Dans l'ensemble, cette phase s'est bien déroulée. Cependant, nous avons parfois rencontré des lacunes de connaissances sur les technologies sélectionnées, entraînant un ralentissement dans le processus de développement. De plus, nous avons accordé une attention particulière à rendre notre solution accessible, en respectant des normes telles que la taille des boutons, la possibilité d'utilisation sans souris et en attribuant des descriptions alternatives aux images (attribut "alt").

En résumé, bien que nous ayons rencontré quelques défis liés à la familiarité avec les technologies choisies, notre collaboration et notre engagement ont permis de surmonter ces obstacles pour aboutir à une solution fonctionnelle et accessible.

7. Mise en place des services

Avec notre application prête à intégrer les services que chacun de nous a choisi, chaque membre du groupe s'est attelé à l'intégration de son service assigné. Chaque service devait être doté de son propre mécanisme d'authentification, ainsi que d'actions et/ou de réactions spécifiques.

Les services que nous avons décidé d'inclure à notre solution sont :

- **Discord** -> David BENISTANT,
- **Trello** -> Oscar FRANK,
- **Github** -> Alessio LIEBENGUTH,
- **Asana** -> Robin DENNI,
- **Google** -> Théotime SCHMELTZ,
- **Jira** -> Oscar FRANK & Robin DENNI.

Durant la mise en place de ces services, nous avons fait face à quelques défis. Discord a été le premier service à être implémenté sans difficultés majeures. Cependant, lors de la mise en place du service Trello, nous avons rencontré un obstacle. Le webhook de Trello nécessitait une URL sur laquelle il pouvait renvoyer les informations, ce qui devenait problématique en environnement de développement local. La solution a résidé dans la mise en place d'un processus de déploiement, confirmé par notre benchmark qui a validé la facilité de déploiement de nos technologies sélectionnées. Ainsi, une configuration GitHub Actions a été instaurée pour automatiser le déploiement à chaque "git push" effectué sur le dossier du projet.

Un autre défi a surgi en raison de la complexité de certaines APIs rencontrée par quelques membres du groupe lors de la connexion ou de la gestion des actions et réactions. Malgré ces complexités, l'intégration des services s'est déroulée de manière globalement fluide et rapide.

Un exemple concret de cette complexité s'est manifesté lors de la mise en place de l'authentification GitHub. Notre projet AREA était développé simultanément sur deux branches distinctes, à savoir "prod" (main) et "dev", ce qui impliquait l'existence de deux instances distinctes du projet AREA avec des URLs différentes. L'authentification GitHub reposait sur l'utilisation d'une application (OAuthApp) liée directement à un site, donc à une URL spécifique. Pour résoudre ce problème, nous avons été contraints de créer une seconde GitHub OAuthApp spécifiquement dédiée à la branche "prod". Cette démarche a introduit des complexités, notamment en termes de compréhension, nécessitant une attention particulière dès qu'une modification spécifique à l'authentification était requise. Les deux URLs distinctes, les deux clientID différents, et les deux secrets client différents ont ajouté une couche de complexité supplémentaire à la gestion de l'authentification.

En résumé, bien que des obstacles aient émergé au cours de l'intégration des services, notre capacité à les surmonter a permis une intégration réussie de l'ensemble des services dans notre solution AREA.

8. Documentation

Pour notre projet, nous avons mis en place une documentation dite « utilisateur » ainsi qu'une documentation dite « technique ».

La première documentation permet aux futurs utilisateurs d'être entièrement guidés, depuis la création de leur compte, jusqu'à la configuration et l'utilisation de leurs AREA. Elle a été élaborée dans le but d'accompagner nos utilisateurs tout au long de leur expérience avec notre solution. En leur donnant accès à des explications claires et détaillées, cette documentation vise à rendre l'utilisation d'AREA aussi intuitive et accessible que possible, garantissant ainsi une expérience la plus positive possible.

En parallèle, la documentation technique détaille de manière approfondie chaque composant clé de notre projet, à savoir le développement front (web et mobile) et le fonctionnement de l'API. Ces ressources détaillées ont été créées dans le but d'aider les développeurs et membres du groupe à comprendre en profondeur l'architecture du produit, les technologies utilisées ainsi que les interfaces du projet. Cette documentation facilite également la maintenance du système et la résolution de problèmes éventuels. Elle joue en plus de cela un rôle essentiel dans le transfert de connaissances au sein de notre groupe de développeurs.

Ces deux types de documentations servent donc à guider les utilisateurs dans leur expérience avec le produit AREA, mais offrent également une ressource complète pour les membres de l'équipe, garantissant ainsi la compréhension et la pérennité du système dans le temps.

9. Ce que l'on retient

À la suite d'une discussion de groupe, nous avons constaté plusieurs aspects cruciaux de notre expérience collective.

Tout d'abord, nous avons réaffirmé l'importance d'une bonne entente au sein de l'équipe pour favoriser un échange fluide et des remises en question sans problème.

Par ailleurs, nous avons clairement observé les avantages significatifs de l'utilisation d'outils de gestion tels que Jira, Trello ou encore Asana. Ces outils nous ont permis un suivi efficace du projet, générant ainsi une meilleure productivité au sein du groupe.

Une répartition la plus équitable possible des tâches s'est également avérée essentielle. Cette approche garantit que chaque membre contribue de manière équivalente, évitant ainsi à certains de s'ennuyer, tout en empêchant d'autres de se débattre avec des tâches trop lourdes.

Nous avons également noté qu'une répartition bien orchestrée des responsabilités, une exécution ponctuelle des tâches par chaque membre, et le maintien à jour des documents partagés éliminent tout risque de tension au sein du groupe. Un tel environnement conduit à une ambiance de travail agréable où la productivité se retrouve grandement augmentée.

10. Conclusion

Le projet AREA a représenté un défi passionnant et formateur pour notre équipe. En mettant en œuvre une approche méthodique depuis la sélection des technologies jusqu'à sa finalisation, nous avons réussi à développer une solution complète et fonctionnelle répondant à nos objectifs initiaux.

Les enseignements tirés de ce projet comprennent l'importance de la collaboration au sein de l'équipe, une répartition équilibrée des tâches, ainsi que la documentation approfondie pour permettre une bonne compréhension du projet.

Le succès de notre produit AREA repose sur la détermination et l'engagement de chaque membre de l'équipe, démontrant ainsi notre capacité à relever des défis complexes et à livrer un produit de qualité. Nous sommes tous fiers du résultat obtenu et prêts à appliquer ces précieux enseignements dans nos projets futurs.