

TIC-GPE5 - StatFive

Fiche Projet

L'ÉCOLE DES TECHNOLOGIES NUMÉRIQUES
AVANCÉES



Type de projet : Analyse Vidéo et
Récupération de Stats



Sommaire

Sommaire	2
Contextualisation	3
Description du projet	4
Description de l'équipe	7
Amélioration du projet	9
Disponibilité du produit	10

Contextualisation

Il est aujourd'hui impossible, dans le monde du football à 5, de fournir des statistiques détaillées du match rapidement sans mettre une énorme somme d'argent et sans équipement supplémentaires contraignant pour le joueur. C'est pour cela que nous proposons la technologie **StatFive**, un programme qui analyse les vidéos directement enregistrées par les caméras des complexes sportifs de foot à 5, déjà existantes, pour en récolter les statistiques individuelles et d'équipe d'un match.

Les **adhérents** de Foot Five débattent des matchs sur leur ressenti au lieu de se baser sur des statistiques. Il y a besoin de gamification pour pouvoir continuer le match en dehors du terrain.

Pour les **propriétaires** de Foot Five, l'enjeu est de fidéliser la clientèle et de se démarquer de la concurrence en offrant à ses clients une expérience inédite.

StatFive cible avant tout les gérants des complexes de futsal mais aussi les adhérents de manière générale. C'est donc du B2B2C (*Business to business to Consumer*)

Pour la concurrence, il n'y a pas vraiment de concurrents directs puisque nous ajoutons de l'intelligence artificielle à notre génération de statistiques via l'analyse vidéo. Contrairement à **Urbansoccer** et **LE FIVE**, qui eux donnent seulement le nombre de but à la fin du match. Dans certains complexes, il peut y avoir la retransmission du match sur Youtube au bon vouloir des personnes. Nous avons par ailleurs, les **applications de montres connectées** qui peuvent sortir des statistiques personnelles mais rien ne touchant au match joué mise à part, les km parcourus et les calories dépensées.

StatFive permet aux gérants de complexe de futsal de pouvoir fidéliser leurs adhérents en y ajoutant un système d'analyse vidéo sur le match afin d'en sortir des statistiques individuelles mais aussi d'équipe grâce à l'intelligence artificielle permettant de gamifier les adhérents. Cette analyse vidéo, donnera l'envie aux joueurs de progresser et de s'améliorer et donc revenir au complexe. Le joueur aura la possibilité de consulter ses statistiques mais aussi les comparer avec les différentes statistiques proposées sur les profils des utilisateurs. Un système de classement sera présent afin de mettre en avant les joueurs les plus performants et ainsi leur donner de la visibilité pour les grosses structures.

Description du projet

StatFive est une solution permettant d'analyser les vidéos directement enregistrés par les caméras des complexes sportifs de foot à 5, pour en récolter les statistiques individuelles et d'équipes d'un match et les rendre disponible sur nos applications web et mobile.

L'analyse de vidéo sera faite avec du **Python** car il est plus compliqué pour nous de structurer notre analyse avec C++, nous allons utiliser une librairie nommée **OpenCV** pour traiter l'image. Cette librairie est open source, ce qui nous permet de l'utiliser gratuitement. Elle est aussi très utilisée dans le milieu car celle-ci est incontournable et efficace

L'utilisation de l'api **Tensor Flow**, permet de faire de la détection d'objets afin de différencier dans notre exemple, le ballon d'un joueur. Gain de temps incroyable sur les différents modèles qu'on pourrait utiliser ailleurs.

Structuriser via des services et des usages de notre application mobiles et applications web. Pour l'application mobile nous utilisons **React Native** car cette technologie s'appuie sur le framework **ReactJS** (qu'on pourra voir juste ci-dessous) qui allie simplicité et modularité pour les développeurs. En pratique, cela signifie que les développeurs travaillent sur une seule base de code, cette technologie est *cross platform*.

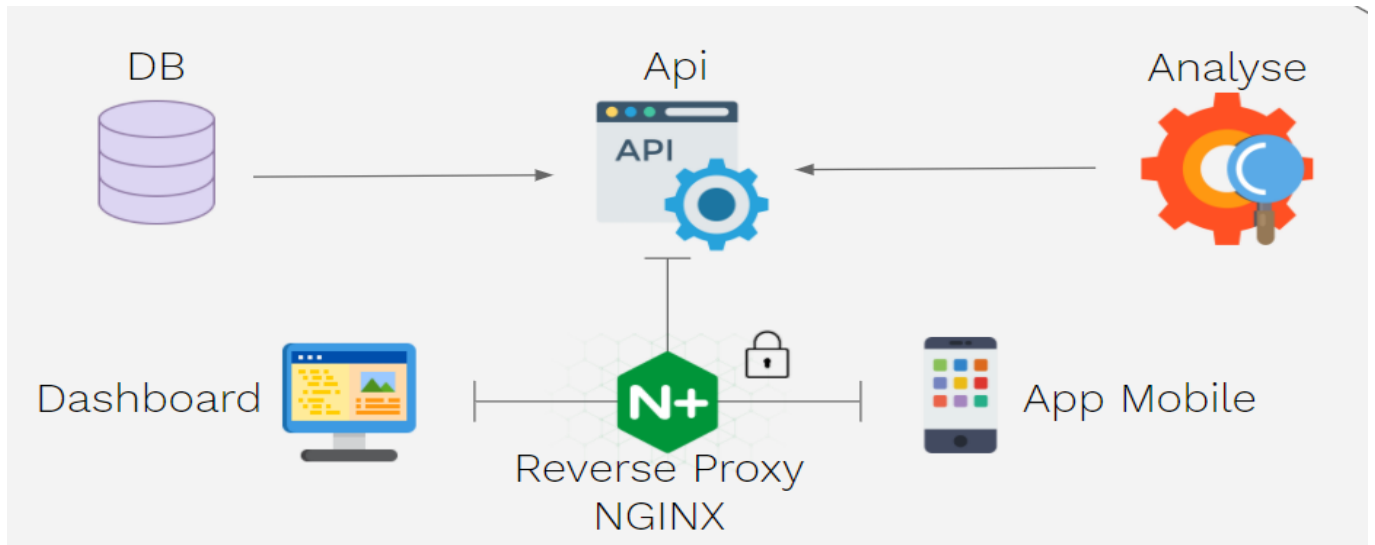
L'application web qui elle utilise le framework **ReactJS** qui va nous permettre de créer une interface utilisateur efficace car la communauté est présente en masse et bien plus grande que celle d'AngularJS. C'est pratique car on peut récupérer des composants existants pour gagner du temps de développement ou bien créer notre propre composant. Le fait qu'en mobile nous utilisons **ReactNative** est aussi un plus pour le choix de **ReactJS** car on pourra se partager les composants et avoir une interface web/mobile cohérente.

Entre nos deux applications, nous utilisons un reverse proxy avec la technologie d'**NGINX** qui est open-source. **NGINX** est un serveur web léger et performant qui joue le rôle d'un intermédiaire avec les applications placées en backend (sans contact direct avec les clients). Le reverse proxy permet d'avoir de la sécurité et des performances.

Nous utiliserons ensuite **mysql** ainsi que **phpmyadmin** car nous cela ne nécessite pas une base de données complexe. De plus, nous maîtrisons ces deux outils que nous utilisons quotidiennement.

Pour notre api, nous utiliserons du **Python** car c'est l'un des langages de programmation les plus utilisés. Il est également connu pour sa syntaxe moins complexe et sa programmation de haut niveau, orientée objet, robuste et polyvalente.

Enfin, nous utiliserons **Docker** allié à **Kubernetes** car cela nous permettra d'avoir tous le même environnement de travail afin que cela fonctionne sur toutes les machines. Cette technologie est très rapide à démarrer et nous permet d'avoir tous les avantages d'une machine virtuelle.



Nous caractérisons notre projet avec 6 mots clés principaux : Service, Gamification, Statistiques, Analyse Vidéo, Automatisation, Cross Platform

Description de l'équipe

L'équipe est composée de 6 membres dont 3 ISR et 3IDV qui sont Acal Habi, Brandon Kernin, Nouredine Nassiri, Jaouad El Hormi, Sofiane Mokhtari, Franck Courtaux.

Chaque membre de notre équipe a choisi une ingénierie métier qu'il veut développer :

Infrastructure Du Cloud	Acal	Brandon	Franck
Ingénierie Logicielle Du Cloud & Des Microservices	Jaouad		
Ingénierie Du Développement D'Application Mobile/Web/UX	Sofiane		
Big Data	Nouredine		

Chacun à une part de responsabilité et nous allons vous le montrer via un tableau :

	App Mobile	App Web	Analyse	API	Infrastructure	Scrum Master
Açal					✓	
Franck	✓				✓	✓
Brandon	✓					
Jaouad			✓	✓		
Sofiane		✓				
Nouredine			✓			

Pour la méthodologie de travail que nous allons exploiter est la **méthodologie Scrum** qui est la plus utilisée parmi les **méthodes Agiles**. Elle consiste à planifier notre projet en différents **Sprints** (comme nous avons pu le voir avec l'ETNA) nous allons, dire qu'un Sprint dure **3 semaines** et durant cette période il va y avoir **3 réunions**, une en fin de semaine de préférence afin de voir l'avancement de chacun et potentiellement aider la personne ayant du retard. À la fin de chaque sprint, nous allons nous réunir afin de faire un compte rendu et prévoir les prochaines tâches à accomplir tout en suivant la **RoadMap** et la **StoryMap**.

À ne pas oublier que le Scrum Master, a un rôle de **COACH** et non pas un **dirigeant** de projet.

Amélioration du projet

Plusieurs améliorations sont à venir, en commençant par l'essentiel, une légère amélioration de l'**optimisation** de l'analyse vidéo, afin de gagner du temps sur le rendu final. Il a aussi été prévu de faire **une librairie avec les différentes routes** de notre api afin de les utiliser plus facilement via l'application mobile et l'application web et par la même occasion, éviter une répétition du code entre les deux plateformes et donc gagner du temps.

Parlons des améliorations à apporter sur l'**application mobile**. Dans un premier temps, il nous faut ajouter la **vérification des comptes** via l'e-mail avec une validation de celui-ci avec un code généré et envoyé. Puis, nous devons refaire une **refonte du code** afin d'ajouter un **système de cache**, prévu lorsqu'on perd la connexion à internet afin d'éviter de charger une page "vide". Par la suite, nous allons faire en sorte d'utiliser la librairie nous permettant de lier l'api avec l'application mobile plus facilement et en rapport avec l'application web. Nous devons aussi y ajouter les **statistiques d'équipes et un classement des meilleures équipes**.

Parlons maintenant des améliorations à apporter sur l'**application web**. Comme pour l'application mobile, il faut ajouter la librairie de liaison api afin de ne pas doubler notre code inutilement. Dans la même lignée, nous devons y ajouter des **filtres de recherche** des "joueurs" / "équipes" pour la création des matchs. Nous devons **retravailler l'ux** qui pour le moment est trop "basique" pour attirer l'œil du client.

Du côté de l'**infrastructure**, nous devons trouver une solution afin de dépenser moins dans les machines azure car ceci revient assez chère pour le peu d'utilisation que nous effectuons. Nous avons donc pensé à effectuer une **technique d'économie**, celle-ci consiste à travailler en local tout le long, lorsque nous devons présenter / essayer, nous allumons les machines et à la fin, nous les éteignons.

Enfin, nous devons faire des **benchmarking** ainsi que des **tests unitaires / fonctionnels**. Sans compter le nombre de **documents** à effectuer dans le futur.

Disponibilité du produit

Nous avons pour but final de le rendre disponible sur :

- <https://statfive.fr> (Déjà disponible).
- Compatible Android / IOS, ajout dans le store respectif de l'os.
- Actuellement dispo sur expo via :

<https://expo.dev/@fr4nck/StatFiveV3>