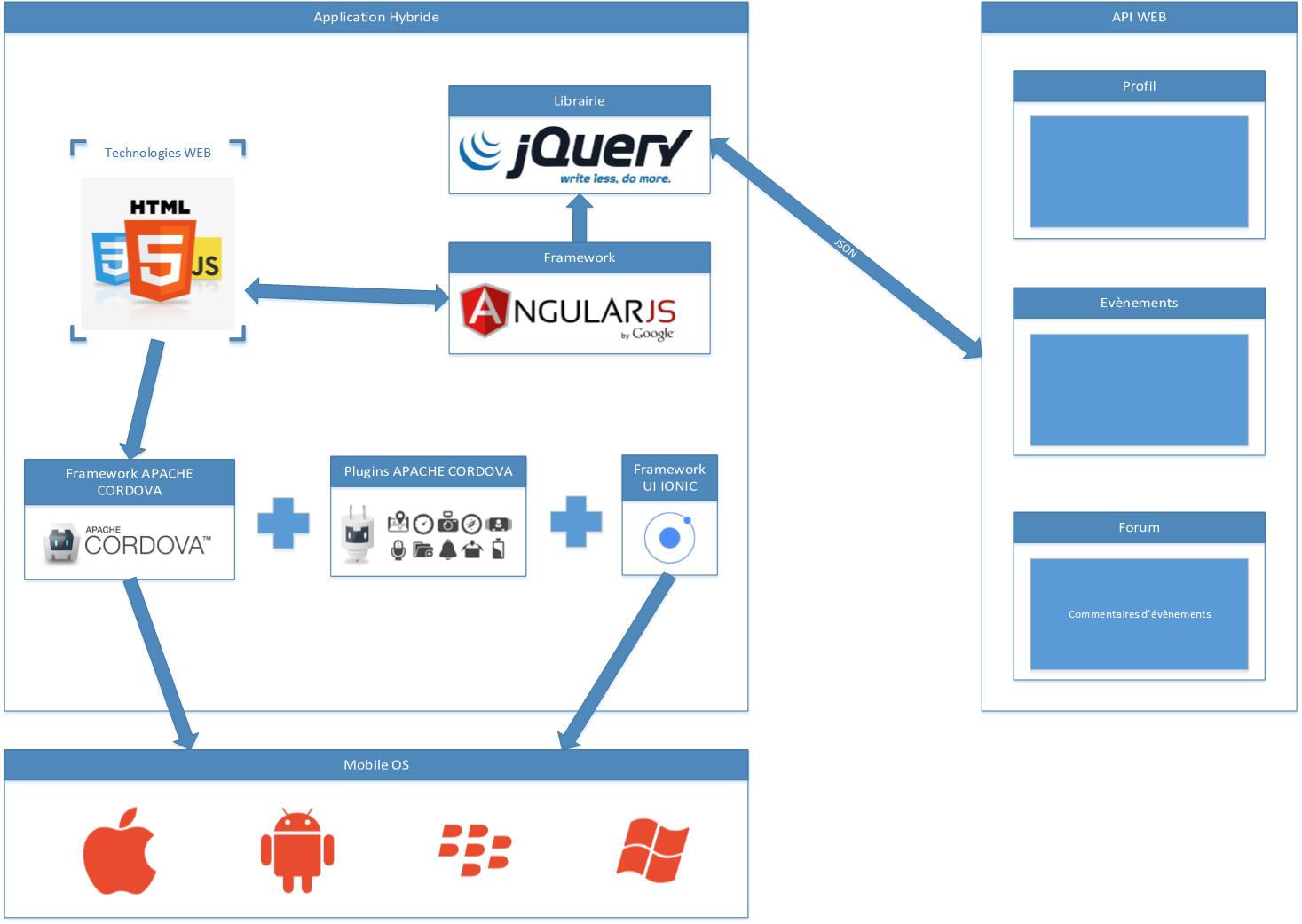
DOCUMENT D’ARCHITECTURE

Ce document vise à exposer les choix effectués par l’équipe en charge du développement de l’application mobile cross-platforms sous cordova quant aux différentes technologies utilisées pour le développement de l’application.

L’équipe désignée ci-après par le pronom « nous » est constituée de Matthias DEVLAMYNCK, Adrien AUDEMAR et de Yoann SOLACROUP.



**Cordova** :

Il s’agit d’un framework open-source de développement d’applications mobiles cross-patforms qui se base sur les technologies de développement web à savoir le html, le CSS et le Javascript.

Il fournit également une couche d’abstraction sous la forme de plugins qui permettent d’interagir directement avec les fonctionnalités natives essentielles des smartphones comme par exemple l’accéléromètre, la géolocalisation via la puce GPS, la caméra et l’appareil photo ou encore les notifications, etc…

Développé sous le nom de PhoneGap par Nitobi Software puis racheté par la suite par Adobe, une partie de son code source a été cédée à la fondation Apache : c’est de là qu’est né le framework Apache Cordova.

Nous avons choisi d’utiliser ce framework pour de multiples raisons dont la principale n’est autre que le fait de pouvoir réaliser une application cross-platforms à la fois disponible sous les dispositifs iPhone, Android et Windows Phone. De plus, ce choix nous permettait de gagner du temps dans notre processus de développement puisque le code fourni avant compilation est unique et ce quel que soit le dispositif final ciblé. En rajoutant le fait que d’utiliser des technologies web, pour nous plus familières, nous évitait un certain nombre d’heures de recherches qu’il nous aurait fallu faire en utilisant les langages natifs propres aux différentes plateformes et le fait que l’utilisation de ce framework open-source gratuit ne nous obligeait à aucune dépense, il apparaissait clair que l’utilisation de ce framework répondant aux besoins du projet nous offrait de multiples avantages.

**Ionic** :

Ionic est un framework open-source, développé par la startup Drifty et basé sur le framework Apache Cordova, qui permet de donner un « look and feel » natif aux applications développées grâce au framework Apache Cordova.

L’équipe s’est tournée vers l’utilisation du framework Ionic car il permet un rendu cohérent et un « look and feel » natif de l’application suivant le dispositif pour lequel l’application est compilée.

Ce framework, lui aussi open-source, ne nous génère aucun frais supplémentaires et nous permet de gagner là aussi du temps grâce à une feuille de style déjà préconfigurée. Le rendu graphique de l’application est déjà géré par le framework lui-même.

Basé sur le framework Cordova, il s’intègre parfaitement à ce dernier et ne nécessite aucun paramétrage supplémentaire pour son utilisation.

La communauté autour de ce framework étant très importante, il sera alors facile de trouver des solutions et de la documentation en cas de problème lors du développement de l’application.

**AngularJS** :

AngularJS est un framework javascript open-source développé par Google. Il permet de construire des applications web et plus précisément de type monopage en suivant le concept MVC (Modèle – Vue – Contrôleur).

Le but de ce framework est de simplifier le développement, de réduire le couplage entre la présentation, les données et la logique métier en insérant du code directement dans les balises du code html et de faciliter l’écriture de tests pour les différentes parties.

Une application mobile est exactement de type monopage puisqu’elle comprend généralement une partie fixe dans laquelle viennent se charger des informations de façon dynamique ce qui rend ce framework très utile dans notre processus de développement.

De plus le framework Ionic dont j’ai parlé un peu plus haut utilise déjà le framework javascript AngularJS et donc l’utilisation d’AngularJS est pleinement supportée par les différents outils déjà utilisés.

Encore une fois open-source, aucun frais n’est à prévoir pour l’utilisation de ce framework.

**JQuery** :

Enfin pour terminer nous utiliserons la librairie JQuery qui nous permettra d’écrire du code javascript de façon plus rapide et plus simple.

Encore une fois ce choix ne nécessite aucun frais supplémentaire et s’adapte parfaitement aux technologies déjà utilisées puisque la librairie JQuery est aussi utilisée dans par le framework AngularJS.

**Résumé** :

Notre idée principale était de pouvoir créer une application cross-platform qui générerait à la fois un minimum de coût et un minimum de temps.

Chaque technologie citée un peu plus haut apporte une plus-value au projet tout en nous permettant de gagner en productivité, en temps et en facilité de développement.

Mais nous n’avons pas utilisé ces technologies uniquement pour les avantages qu’elles nous fournissaient puisque nous avons apporté une réflexion afin que chacune des technologies utilisées vienne parfaitement s’intégrer avec les autres.

Ainsi nous pouvons dire que toutes les technologies utilisées sont utiles au projet mais surtout complémentaires.

**API Web** :

Afin de communiquer avec les différentes API Web utilisées par l’application mobile cordova, à savoir Profil, Evènement et Forum (Commentaires d’évènements), nous avons convenu avec les différents modules de communiquer via des requêtes de type « POST », « GET », « UPDATE » et « DELETE ».

**Trello** :

Les tickets dans Trello portent le nom des éléments du DSF auxquels ils font référence. Ils portent aussi un code à deux chiffres permettant de les référencer dans les messages des commits sur git ou encore dans les commentaires d’un autre ticket Trello.

Par exemple, dans le DSF utilisateur, l’écran « Inscription » est référencé dans le Trello par le ticket « 1 Utilisateur – 2 Inscription ». Dans les commits, il sera référencé par un message commençant par « 1-2 » et dans le ticket « 1 Utilisateur – 1 Login », il sera aussi référencé par la ligne « Lié à 1-1 ».