

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

09/06/2018

# RAPPORT1 :

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

NGO BIBINBE ANNE

## Table de matière :

1-Différence entre reactnative et reactjs .....	2
2-Notion de client serveur .....	2
3-Notion de service .....	2
4-Difference entre la poo et la programmation procédural .....	2
5-Programmation orienté composant .....	2
6-avantage de chacune des programmations.....	3
*avantages de la programmation procédurale .....	3
*avantages de l'orienté objet.....	3
*avantages de la programmation orienté composant.....	3
7-difference entre les applications natives et hybrides .....	3
8-avantages et inconvénients de ces deux types d'applications.....	3
inconvenient.....	3
*inconvenient des applications hybrides .....	4
*avantages des applications hybrides.....	4
9-exemples de technologies pour développer les applications hybrides .....	4
10- Node.js .....	5
11-Comparaison de reactnative et du développement hybride .....	5
12-Avantage et inconvénients de react native et de reactjs.....	5
*inconvenient de reactnative.....	5
*avantage de reactnative .....	5
*inconvenient de reactjs .....	5
*avantage de reactnatjs .....	5

## 1-Différence entre reactnative et reactjs

1. ReactJs est à la base un framework JavaScript pour les navigateurs Web (PC & Mobile) or React Native est pour créer des applications natives pour mobile.
2. ReactJs manipule le DOM du navigateur web donc ses composants sont essentiellement des balises HTML et voir XML or React Native utilise des composants créés par le développeur à partir de ses propres composants de base pour créer du code natif!
3. ReactJs communique directement avec le DOM (model de documentation objet) or React Native passe par un pont ou bridge pour interpréter les actions effectuées sur les différentes vues!
4. Bien que les deux soient toujours du JavaScript qui est exécuté, l'un l'est obligatoirement sur navigateur (ReactJs) et l'autre par des Thread d'une application mobile (React Native).

## 2-Notion de client serveur

L'environnement **client-serveur** désigne un mode de communication à travers un réseau entre plusieurs programmes : l'un, qualifié de **client**, envoie des requêtes ; l'autre ou les autres, qualifiés de **serveurs**, attendent les requêtes des **clients** et y répondent. Généralement du côté serveur une base de donnée est implémenté et c'est à partir de cette base de donnée que le serveur trouve l'ensemble des réponses aux différentes requêtes du client.

## 3-Notion de service

Comme on l'entend un service permet de répondre à un besoin donc c'est une application qui en s'exécutant fournit des réponses qui répondent aux besoins de l'utilisateur. Par exemple Pour l'implémentation d'un service web on doit réaliser :

- Un fichier décrivant le service écrit en wsdl (web services description language ).(partie cliente)
- Un programme réalisant le service (partie application)
- maîtriser une API Java pour XML(api permettant d'accéder au services xml à partir du langage java)(partie bd)

## 4-Difference entre la poo et la programmation procédural

L'une des différences fondamentales est que la programmation procédural on se pose la question de : <<Qu'est-ce que fait projet >> tandis que avec la poo on se demande plutôt d'abord que doit contenir le projet. La programmation procédurale est centrée sur le traitement tandis que la poo est centré sur les données. L'approche procédural s'appuie sur le concept d'appel procédural ; tandis que la poo s'appuie sur les concepts d'encapsulation ; héritages, polymorphisme; agrégation.

## 5-Programmation orienté composant

Elle a utilisé une approche modulaire ; ce qui veut dire que notre logiciel sera construit en briques qu'on agencera à la fin pour faire le bloque final.

## 6-avantage de chacune des programmations

### \*avantages de la programmation procédurale

-contrairement par exemple à une simple programmation séquentielle on peut réutiliser le même code sans avoir à le retaper ce qui fait gagner en temps et réduit la taille du code source) ;les modifications sont plus simples à faire avec ce type de programmation

-

### \*avantages de l'orienté objet

-l'encapsulation qui nous permet de manipuler les objets ce qui nous permet de définir nos propres types de variables

-le polymorphisme qui facilite la tâche aux programmeurs parce qu'il permet à un objet de prendre plusieurs formes ;ca

- la poo utilise aussi le concept d'approche procédural donc garde tous les avantages de la programmation procédural

-

-l'orienté objet est plus utilisé ;donc dès que l'on a un problème il y a beaucoup plus de tête ouverte pour résoudre le problème

### \*avantages de la programmation orienté composant

-elle est plus lisible

-le logiciel conçu avec cette approche est plus facile à maintenir

-le fait que la conception soit faite en brique qui seront tester une à une facilite le travail permet que ces briques soit réutilisables.

## 7-différence entre les applications natives et hybrides

Les applications natives ne sont pas compatibles avec toutes les plateformes mobiles tandis que les applications hybrides si. Les applications hybrides sont nées en faisant un lien avec les applications web et les applications natives ;donc une application hybride peut parfaitement être implémenter pour les mêmes besoins auxquels répondent une natives. -

## 8-avantages et inconvénients de ces deux types d'applications

inconvénient des applications natives

- Les applications natives prennent du temps à être développées et sont relativement coûteuses.
- S'ajoutent d'éventuels problèmes de rétrocompatibilité, en raison notamment de la création de nouvelles versions du système d'exploitation.

Celles-ci peuvent aussi se voir refuser l'accès au Store sur lesquelles elles souhaitent être mises à disposition. En effet, elles doivent passer par une batterie de tests de validation de ces Stores, qui peut les accepter ou les refuser

\*avantage des applications natives

- Ces applications sont capables d'utiliser l'ensemble des fonctionnalités du mobile et peuvent être utilisées sans avoir accès à Internet.

Elles s'adaptent notamment à de nouveaux « business model » par exemple les applications de type « freemium », ou l'installation est gratuite, avec en plus, la possibilité d'avoir accès à des options supplémentaires, mais qui elles sont payantes.

\*inconvenient des applications hybrides

-les interfaces utilisateurs sont limités par exemple parce que les application hybrides utilise les vues web on ne peut pas avoir de vue 3D

-Une dépendance vis à vis de la vitesse du navigateur

-il est difficiles d'utiliser toutes les capacités des plateformes pour pouvoir y arriver il faut une certaine combinaison de plateforme

-leurs fonctionnement est plus fragiles et la maintenance est plus lourdes et complexes car le code source est mélangé et la compilation se fait en plusieurs étapes

-en utilisant une application hybride nous devenons dépendant de cordova donc compatible avec toutes les nouvelles versions d'os .

\*avantages des applications hybrides

- Ces dernières sont considérées comme un mix, entre les Web Apps et les Nativ Apps. En effet, elles sont compatibles avec toutes les plateformes mobiles. Mais ces applications sont principalement développées à l'aide d'HTML5, aujourd'hui, qui est très performant mais qui utilisent aussi d'autres langages Web comme le CSS et le JavaScript.
- Ainsi, une application dite hybride n'est pas dépendante d'une plateforme mobile en particulier. De la même manière
- contrairement aux applications Web, les applications hybrides peuvent accéder à toutes les fonctions présentes sur le mobile. Cela est rendu possible par des liens qui sont faits entre le langage natif et la technologie Web présente dans l'application hybride.
- La réutilisabilité des codes

Vous comprendrez donc qu'en combinant les points forts des applications natives et Web, celles-ci suscitent, de plus en plus, l'intérêt des développeurs d'applications mobiles

9-exemples de technologies pour développer les applications hybrides

Il en existe plusieurs :

- phoneGap
- Apache cordova ou Rho mobile

## 10- Node.js

C'est un logiciel JavaScript qui s'exécute du côté serveur ;il permet de créer des applications réseau évolutives il permet de créer un serveur de fichier abstrait au lieu d'en installer ; et celui-ci ne bug pas lorsqu'on essaye de faire des connexions .il est utilisé pour la programmation temps réel .

## 11-Comparaison de reactnative et du développement hybride

Le développement hybride mobile le plus populaire avant l'arrivée de react native c'est ionic face ;qui aujourd'hui est dépassé par react native.

-par exemple par rapport aux images ionic crée l'image et le webview se charge de l'afficher par contre react native crée l'image et appelle un composant natif de l'appareil qui se charge de l'afficher par conséquent on a une meilleure visibilité et l'affichage est plus rapide\*

En conclusion le fait que react native fasse appel aux composants natifs le rend les applications utilisant ce développement plus rapides et avec de meilleures performances

## 12-Avantage et inconvénients de react native et de reactjs

### \*inconvénient de reactnative

Avoir une certaine expérience en react

Développer le projet sous deux plateformes(ios et android)

### \*avantage de reactnative

Les codes sont réutilisables

L'utilisation des composants natifs

Elle convient à toute sorte d'applications

Ressources libres et gratuites

### \*inconvénient de reactjs

Ajouter les bibliothèques et les composants externes

Le phénomène de churn car la réutilisation connaîtra des problèmes car les éléments à réutiliser deviendront probablement obsolètes

### \*avantage de reactjs

Rapide

Il crée son propre domaine ce qui évite les opérations assez coûteuses

Beaucoup d'adhérents

L'intelligibilité car il produit des codes propres