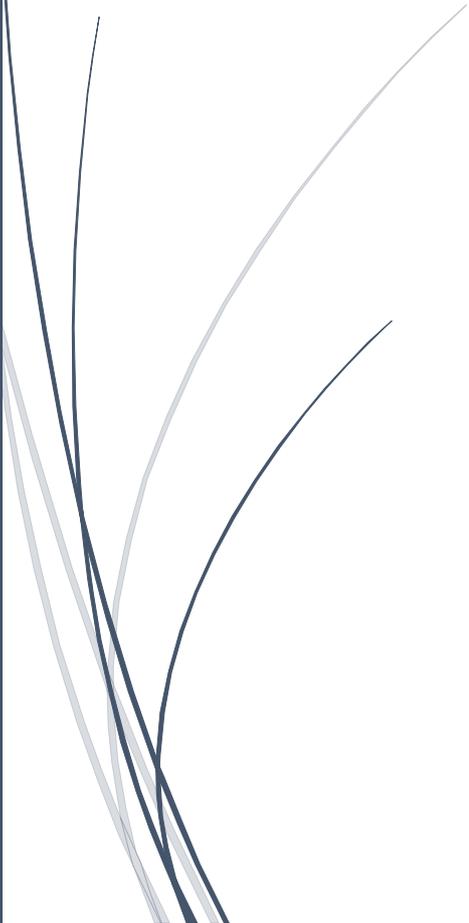


05/07/2019

Application guide du Jardin d'Essais Botanique de l'INRA

Rapport final



BASBAS Adam & BENBADA Mohamed Amine
GI-2

Contents

I-	L'INRA en bref.....	3
II-	Problématique et solution.....	4
III-	Cahier des charges.....	5
IV-	Prototypage.....	6
V-	Technologies utilisées.....	8
VI-	Partie QR Code.....	11
	a. Définition.....	11
	b. Comment ça marche ?.....	11
	c. Implémentation.....	12
VII-	Construction de la base de données.....	13
VIII-	Fonctionnement de l'application.....	14
	a. Menu principal.....	14
	b. Catalogue.....	15
	c. Plan.....	18
	d. Informations.....	20
	e. QR Code.....	21
IX-	Perspectives.....	22
X-	Webographie.....	23

I- L'INRA en bref

L'Institut National de Recherche Agronomique est un établissement public dont les origines remontent à 1914 avec la création des premiers services de recherche agricole officiel. Il s'agit d'une institution en extension perpétuelle pour une meilleure recherche de proximité, à travers :



- 10 centres régionaux de la recherche agronomique
- 23 domaines expérimentaux couvrant les différentes zones agroécologiques du Royaume
- 30 unités multidisciplinaires de recherche encadrées par 8 départements scientifiques
- 10 services recherche et développement (des structures régionales assurant l'interface entre la recherche et le développement avec des attachés de communication)

L'INRA adopte un Programme de Recherche à Moyen Terme, élaboré selon une approche inclusive, fondé sur la vision politique de développement, axé sur les filières de production agricoles et des domaines transverses, intégrant ainsi les projets de coopération nationale et internationale. L'Institut jouit aussi d'un potentiel humain qualifié, constitué de :

- 176 chercheurs
- 144 techniciens
- 49 administrateurs

II- Problématique et solution

Prenons comme exemple un visiteur du Jardin d'Essais Botanique de l'INRA. Celui-ci peut bien se servir des plaquettes présentes au niveau du JEB pour s'informer par rapport à certaines plantes, pour se situer au sein même du jardin, ou même pour avoir des informations concernant son historique ou celui de l'INRA. Mais cette approche reste néanmoins incomplète puisque le visiteur est soumis à plusieurs contraintes :

- Il doit se trouver près d'une plaquette informative soit pour consulter le plan, les informations à propos du JEB, ou des informations générales
- Toutes les espèces du jardin n'ont pas forcément une plaquette informative

Si l'INRA souhaite palier à ces manquements, plusieurs complications voient le jour :

- Le jardin renfermant plus de 660 espèces, il est inconcevable de prévoir 660 plaquettes individuelles pour chacune de ces espèces
- Le jardin s'étend sur 17 hectares, il est aussi impossible de prévoir des plaquettes informatives et redondantes sur l'ensemble de cette surface

L'application guide du Jardin d'Essais Botanique de l'INRA entre donc en jeu. Il s'agit d'une application mobile qui a pour objectif principal de guider le visiteur du jardin durant son court séjour au sein du JEB.

III- Cahier des charges

Après de multiples réunions avec les représentants de l'INRA ainsi que des déplacements au sein du JEB à Rabat, il a été convenu que l'application pilote contiendra l'ensemble des espèces du carré thématique n°1 (sur un nombre total de 22 carrés). La consultation des informations concernant une espèce devra contenir une description brève ainsi qu'une image décrivant celle-ci.

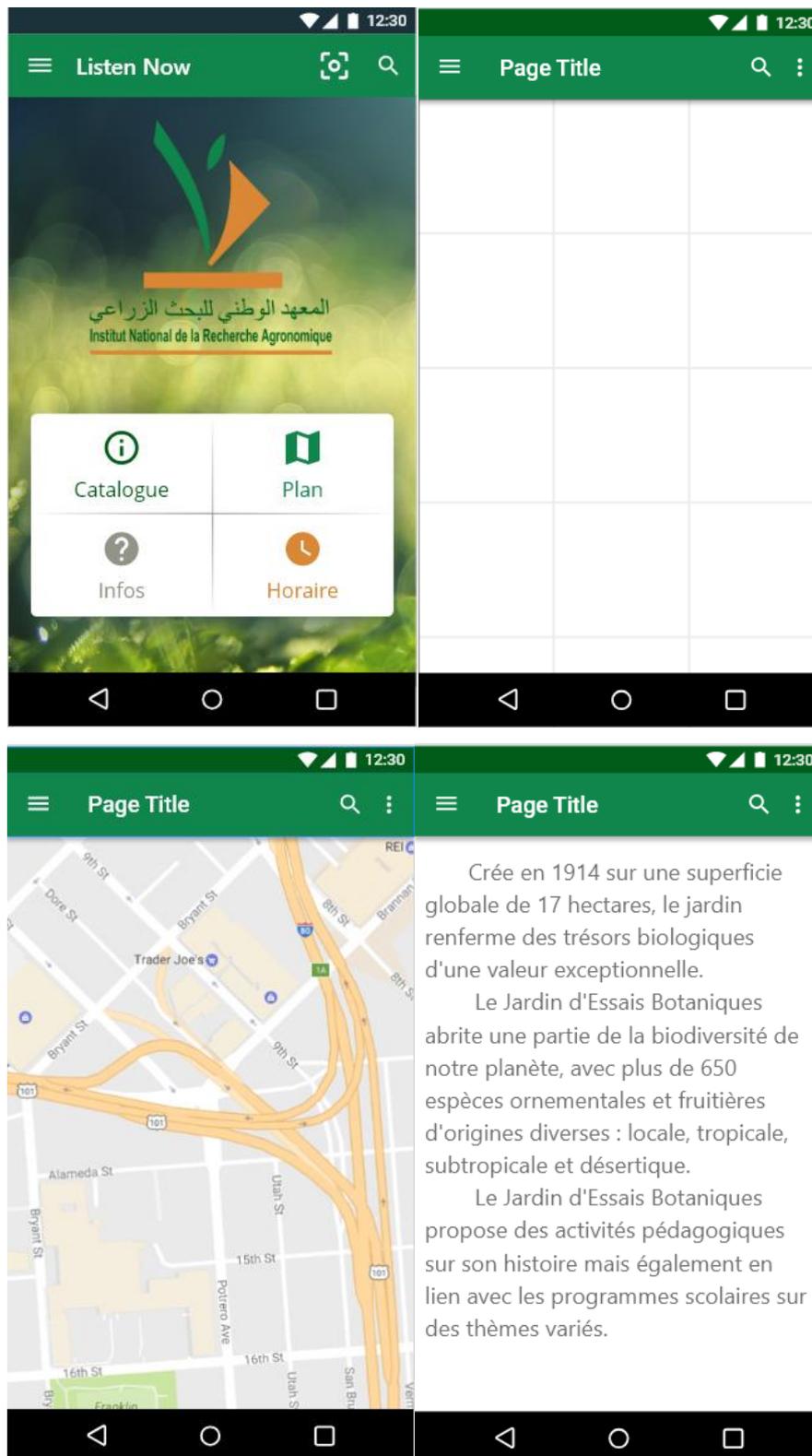
Il a aussi été convenu qu'un QR code sera fourni pour chaque plante, permettant à l'utilisateur de scanner celui-ci afin d'avoir les informations précédemment décrites directement.

Le cahier des charges prend donc la forme simplifiée suivante :

- Un catalogue regroupant l'ensemble des espèces du carré thématique n°1
- Un plan complet du JEB avec possibilité de consulter le catalogue à partir de ce plan
- Une partie regroupant historique, informations et horaires concernant l'INRA et le JEB
- Une partie QR code qui permet à l'utilisateur de scanner un QR code correspondant à chaque plante.

IV- Prototypage

Un premier prototype a été réalisé afin d'avoir une idée globale concernant l'application ainsi que l'enchaînement que celle-ci devrait suivre :



Lors du lancement de l'application, l'utilisateur peut donc choisir quelle rubrique consulter parmi 4 choix :

- Catalogue
- Plan
- Horaires
- Informations

Il lui est aussi possible de scanner un QR code à travers l'icône présente en haut à droite de l'écran.

V- Technologies utilisées

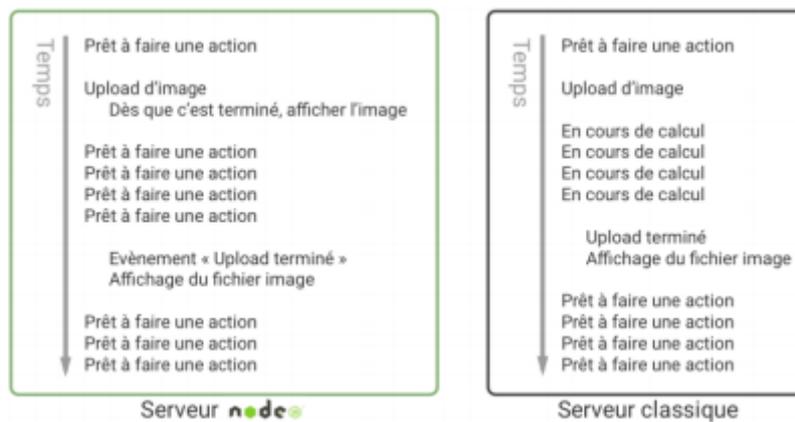
Expo : outil gratuit et open-source construit autour de React Native, permettant de faciliter l'élaboration de projet native Android et iOS, utilisant JavaScript et React.



Node.js : environnement d'exécution multiplateforme JavaScript qui exécute du code JS côté serveur, alors qu'auparavant il était nécessaire d'utiliser du JavaScript pour le développement côté client (frontend) ainsi qu'un autre langage côté serveur (backend). Node.js permet ainsi de coder les deux parties d'une application en JavaScript.



L'utilisation de Node.js en tant que serveur Web permet aussi de traiter un gros volume de requêtes simultanément de manière efficace.



Cet environnement met à disposition du codeur plus de 500.000 packages grâce au gestionnaire de packages **npm**. Celui-ci était initialement un gestionnaire de packages spécifique à Node.js, mais a réussi à convaincre le monde JavaScript.

React : React est une librairie JavaScript flexible, efficace et déclarative pour établir des interfaces d'utilisateur (UI). Elle nous permet de réaliser de complexes UI à partir de plusieurs bouts de codes isolés appelés « Components ».

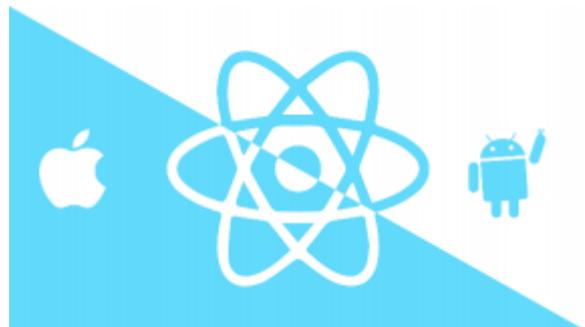


Le contenu créé sur React est référençable : grâce à l'utilisation d'un serveur Node, le code est généré côté client et côté serveur à la différence des autres frameworks JavaScript.

ReactJS est aussi extrêmement rapide, celui-ci crée son propre DOM virtuel où sont rattachés les composants créés. Cela nous apporte de la flexibilité et des performances exceptionnelles, puisque le DOM ne change que la partie qui a besoin d'être mise à jour à chaque exécution.

On pourra aussi ajouter que du code React est très simple à lire et à écrire, rendant ainsi celui-ci très facile à comprendre et manipuler.

React Native : Il s'agit de l'implémentation de React à Android et iOS. Il s'agit d'une librairie JavaScript développée par Facebook et Instagram, nous permettant de coder à la fois pour iOS et Android, entraînant ainsi un gain de productivité.



L'utilisation de React Native permet d'obtenir des applications mobiles très performantes. Elles sont fluides et responsives et offrent une très bonne « User Experience » et « User Interface ».

Firestore Realtime Database : Base de données hébergée dans le cloud. Les données sont stockées en tant que JSON et synchronisées en temps réel à chaque client connecté.



S'agissant ici d'une application cross-platform, tous les clients partagent la même base de données et reçoivent automatiquement des mises à jour contenant les données les plus récentes.

MediaWiki API: service web qui autorise l'accès à certaines fonctions du Wiki comme l'authentification, les opérations sur les pages, et la recherche. On fera appel à cette API pour accéder aux données présentes au niveau des pages Wikipédia de chaque espèce.



Les packages utilisés sont résumés dans l'entête de **package.json** qui se présente comme suit :

```
"dependencies": {
  "@expo/samples": "2.1.1",
  "@mapbox/polyline": "^1.0.0",
  "expo": "^32.0.0",
  "firebase": "^5.10.0",
  "react": "16.5.0",
  "react-native": "https://github.com/expo/react-native/archive/sdk-32.0.0.tar.gz",
  "react-native-calendars": "^1.36.0",
  "react-native-camera": "^2.9.0",
  "react-native-elements": "^1.1.0",
  "react-native-firebase": "^5.3.1",
  "react-native-grid-component": "^1.1.0",
  "react-native-maps": "^0.24.2",
  "react-native-qr-code-scanner": "^1.2.1",
  "react-native-render-html": "^4.1.2",
  "react-native-square-grid": "^1.0.5",
  "react-native-super-grid": "^3.0.4",
  "react-native-table-component": "^1.2.0",
  "react-native-webview": "^5.12.0",
  "react-navigation": "^3.0.9"
},
```

VI- Partie QR Code

a. Définition

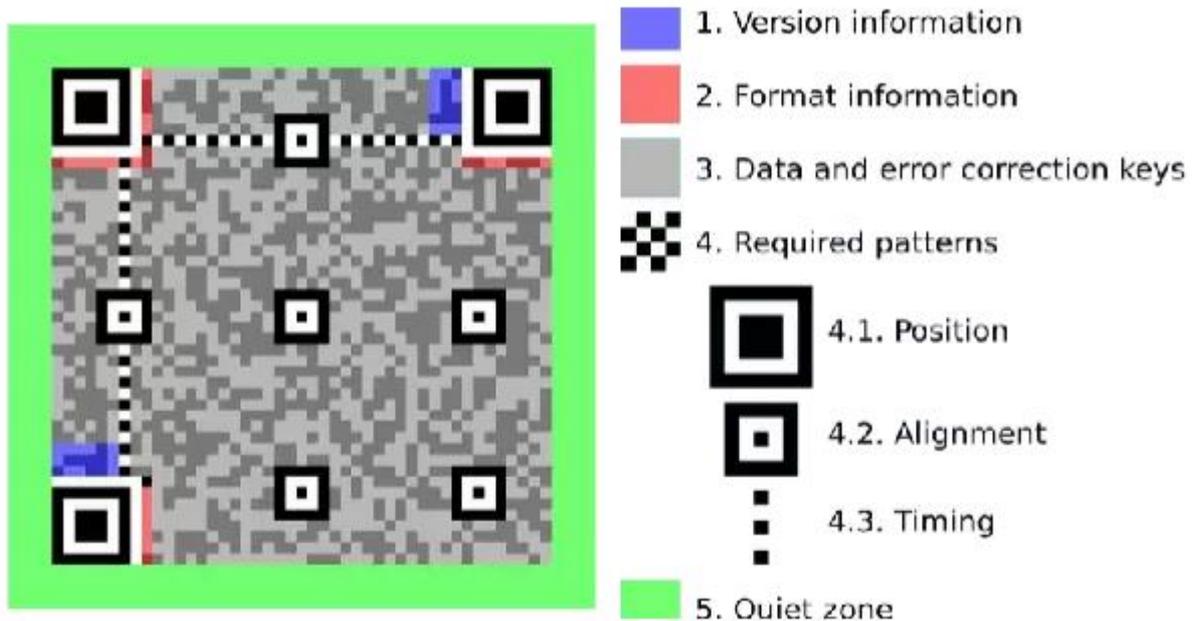
Avant d'aborder comment la partie QR Code a été implémentée, expliquons tout d'abord le principe de celui-ci.

Un QR Code est un type de code-barres en deux dimensions constitué d'un carré noir sur lequel sont disposés des motifs blancs.

Ce code a été dénommé QR parce qu'il constitue une « quick response », ce qui est nécessaire dans notre cas. Le contenu de ce code peut être décodé rapidement à travers une caméra, permettant ainsi de déclencher les actions appropriées à notre application.

b. Comment ça marche ?

L'information (string) est encodée en suivant des motifs de noir et de blanc, correspondant à des 0 et des 1. Les trois « bulleeyes » motifs sont utilisés pour déterminer le sens du code. Cela permet au décodeur de lire correctement le QR Code.



On notera aussi que chaque QR Code a une capacité de correction d'erreur : si le code scanné est endommagé, il y a toujours possibilité d'envoyer la correcte information au décodeur. Il y a quatre niveaux de correction d'erreur suivant le tableau ci-dessous :

QR Code Error	Correction capability
Level L	7%
Level M	15%
Level Q	25%
Level H	30%

Le niveau d'erreur est choisi dépendamment de l'utilisation ou l'environnement où le code sera affiché. Augmenter le niveau d'erreur revient à augmenter la taille du QR Code.

Dans la plupart des cas, le niveau M est celui sélectionné, mais on choisira le niveau H pour palier à d'éventuelles intempéries ou endommagement des plaquettes d'information au niveau du JEB.

c. Implémentation

On rappelle que le QR code qui servira à consulter les informations spécifiques par rapport à une plante est généré à partir d'un string.

Pour répondre aux nécessités de notre application, nous avons décidé que la partie consultation des détails d'une espèce suivra le cheminement suivant :

- Si la caméra est active, on effectue la lecture du QR code affiché
- On compare le « string » du QR code lu avec le « string » propre à la plante
- Si la comparaison est juste, une fenêtre d'alerte s'ouvre affichant de brèves informations concernant la plante, tandis qu'en même temps l'utilisateur est redirigé vers une page de description plus détaillée.

VII- Construction de la base de données

Afin de répondre aux spécificités de l'application, nous avons opté pour Firebase Realtime Database. On rappelle que celui-ci est un service qui permet de stocker et de synchroniser les données dans le cloud NoSQL.

Il a été nécessaire de structurer la base de données comme suit :



- Le nœud C1 : représente le carré thématique n°1
- Le nœud liste : contient tous les gens présents au niveau du 1^{er} carré thématique
- Le nœud Aberia : genre de plante, contient toutes les espèces de ce genre.
- Les champs atomiques présents au niveau de chaque espèce dont on décrira l'utilité plus tard.

Afin de connecter notre base de données à l'application, il suffit d'ajouter le snippet suivant :

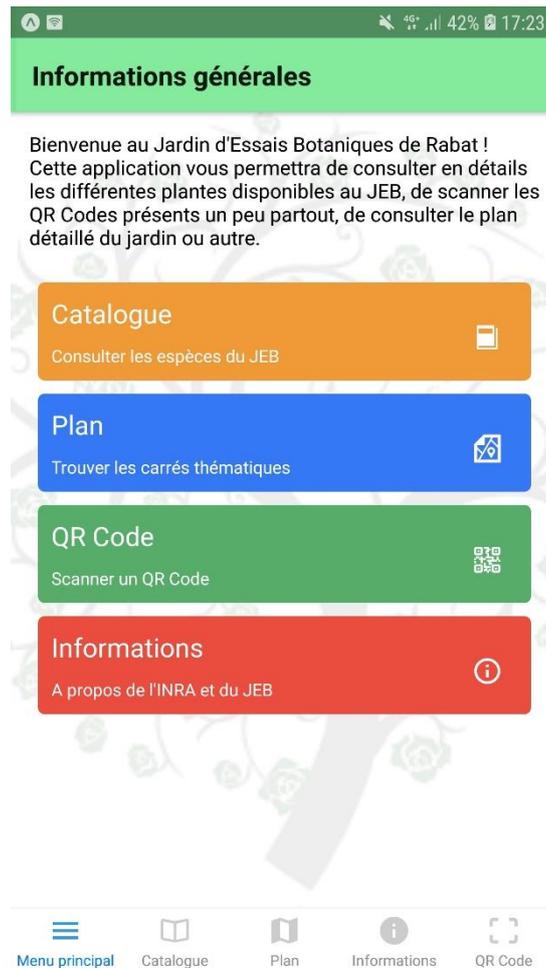
```
let config = {
  apiKey: "AIzaSyA_vpC9vC3lEcxur7VrvdCL3YgJGSr3YPQ",
  authDomain: "inra-f0866.firebaseio.com",
  databaseURL: "https://inra-f0866.firebaseio.com",
  storageBucket: "inra-f0866.appspot.com"
}; // Connection string
```

VIII-Fonctionnement de l'application

Une fois la base de données créée et liée, nous pouvons à présent établir une première expérience d'utilisation.

a. Menu principal

L'utilisateur est accueilli par un menu principal, sur lequel figure un message de bienvenue ainsi que 4 boutons l'invitant à découvrir les fonctionnalités de l'application. On notera aussi la présence d'une barre de navigation en bas de l'écran.



b. Catalogue

- 1^{er} écran : les genres du 1^{er} carré

La rubrique catalogue de l'application plonge l'utilisateur directement au sein des genres de plantes présentes au niveau du carré 1. L'affichage du titre et de l'image de fond est faite à travers les champs correspondants au niveau de la base de données.



- 2^{ème} écran : les espèces du genre sélectionné

Après sélection d'un genre au niveau du 1^{er} écran, l'affichage des espèces du genre sélectionné est effectué sous la forme suivante :



○ 3^{ème} écran : description de l'espèce

Cet écran rassemble une description brève ainsi qu'une image représentant l'espèce consultée.

Deux cas de figure peuvent se présenter :

- On utilise la combinaison « genre + espèce » afin de tenter de récupérer une description brève de la plante à travers la MediaWiki API. L'appel à l'API renvoie une table JSON qui contient du contenu HTML qu'il faudra interpréter.

```
{
  "batchcomplete": true,
  "warnings": {
    "extracts": {
      "warnings": "HTML may be malformed and/or unbalanced and may omit inline images. Use at your own risk. Known problems are listed at https://www.mediawiki.org/wiki/Extension:TextExtracts#Caveats."
    }
  },
  "query": {
    "pages": [
      {
        "pageid": 1697651,
        "ns": 0,
        "title": "Bauhinia forficata",
        "extract": "<p>La <i><b>Bauhinia forficata</b></i> ou <b><span lang='\"pt\">pata-de-vaca</span></b> (« patte de vache »), surnommée ainsi en raison de ses feuilles en forme de sabot, est une plante de la famille des fabacées. Elle se retrouve en Amérique du Sud dans des habitats du type forêt atlantique. Des études ayant prouvé qu'elle contient de l'insuline, cette plante est suivie dans la recherche sur le diabète</p>"
      }
    ]
  }
}
```

- Si cela n'est pas possible (une page en français de la plante n'existe pas, ou la plante n'est pas du tout référencée au niveau de Wikipédia), on utilise la description fournie au niveau de la base de données.

Dans les deux cas, on utilise une image stockée aussi au niveau de la base de données. Ce qui donne ainsi le résultat suivant :



The screenshot shows a mobile application interface. At the top, there is a green header bar with a back arrow and the word "Description". Below the header, there is a paragraph of text describing the plant. Underneath the text is a photograph of the plant's flowers and leaves. At the bottom of the screen, there is a navigation bar with five icons and their corresponding labels: "Menu principal", "Catalogue", "Plan", "Informations", and "QR Code".

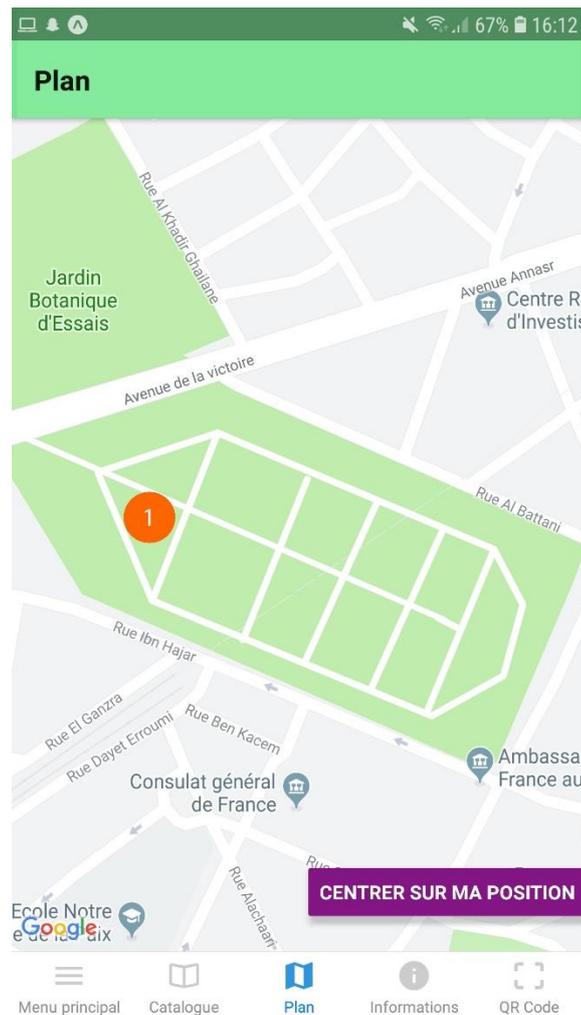
La **Bauhinia forficata** ou **pata-de-vaca** (« patte de vache »), surnommée ainsi en raison de ses feuilles en forme de sabot, est une plante de la famille des fabacées. Elle se retrouve en Amérique du Sud dans des habitats du type forêt atlantique. Des études ayant prouvé qu'elle contient de l'insuline, cette plante est suivie dans la recherche sur le diabète

c. Plan

Cette page renvoie l'utilisateur directement sur un plan personnalisé du JEB. On précise que le plan du jardin en utilisant Google Maps est comme suit :



Tandis que le plan où est plongé l'utilisateur a été conçu en utilisant le package **react-native-maps** et en utilisant les longitudes et latitudes de chaque point essentiel pour tracer les contours du jardin.



On remarque aussi la présence de markers qui indiquent à quel niveau se trouvent les carrés thématiques du jardin. Le cliquer sur un marker renvoie directement au catalogue du carré spécifié.

Il est aussi offert à l'utilisateur de localiser sa position en appuyant sur le bouton adéquat, bien qu'il doit avoir donné l'accès à sa position durant le lancement de l'application.

d. Informations

Cette page est une vue basique qui regroupe l'ensemble des informations concernant l'INRA ainsi que le Jardin d'Essais Botanique. Bien qu'il était prévu d'avoir deux vues différentes (une pour « Horaires », une autre « Informations »), il a été jugé plus utile de les regrouper au sein d'une seule et même vue.



Informations

Historique

Le Jardin d'Essais Botaniques de Rabat fut créé en 1914 au sein de la station Régionale Horticole, première station de la recherche agricole (actuellement INRA), il fut inauguré par le Souverain Feu Moulay Hafid. Son aménagement fut achevé en 1919.

Le Jardin d'Essais s'étend sur une superficie d'environ 17 hectares. Cet espace vert devrait servir de lieu d'acclimatation des espèces ornementales et les espèces fruitières exotiques, dans un but d'expérimentation scientifique au service de l'agriculture marocaine.

Les plantes ont fait l'objet d'introductions successives et proviennent aussi bien de pays à climat tempéré que de régions tropicales.

Le style des jardins de la renaissance, avec une perspective en terrasses à la française, aux lignes sobres et majestueuses, ont inspiré le concepteur et a fait la trame de fond d'aménagement du jardin.

Jardin d'essais botanique : patrimoine culturel

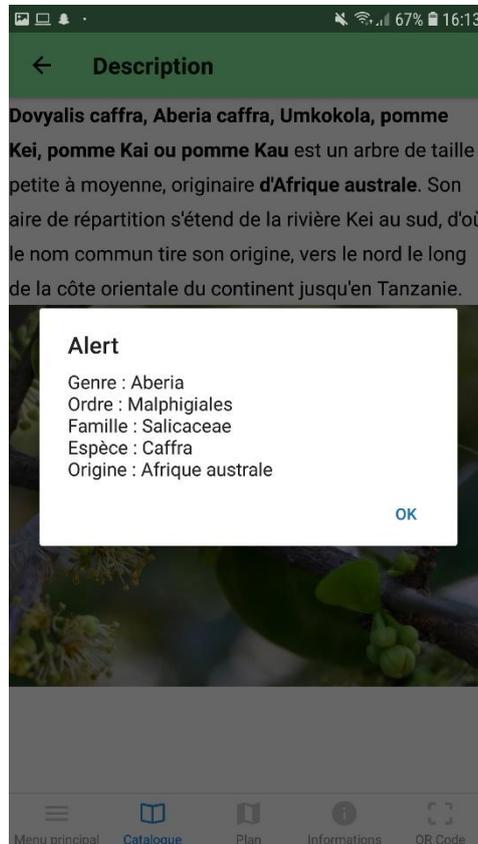
En 1992, le Jardin d'Essais Botaniques a fait l'objet d'un arrêté d'inscription comme Patrimoine National et fait désormais partie des domaines protégés de l'Etat.

Réhabilitation du jardin

Menu principal Catalogue Plan **Informations** QR Code

e. QR Code

Comme précisé auparavant, cette page a pour but de permettre à l'utilisateur de scanner un QR code spécifique à une plante. Il est demandé préalablement à l'utilisateur de donner l'accès à la caméra à l'application. Si cette condition est remplie, l'utilisateur peut donc lire le QR code qu'il souhaite, il obtiendra donc l'alerte suivante :



Le QR code est « écrit » de sorte à ce qu'il corresponde au genre et à l'ID de la plante au niveau de la base de données. Tandis qu'une alerte contenant la majorité des informations concernant la plante est affichée, l'utilisateur est aussi redirigé vers l'écran descriptif de cette plante comme vu auparavant :

IX- Perspectives

L'application dans son état actuel permet de consulter l'ensemble des espèces du carré thématique n°1, de consulter le plan interactif, de s'informer par rapport à l'INRA et au JEB ainsi que de scanner le QR code d'une plante donnée. Néanmoins, l'application reste perfectible, nous proposons ainsi les améliorations suivantes :

- Inclure l'ensemble des carrés thématiques du jardin
- Améliorer le plan en délimitant chaque carré du jardin avec une légende
- Créer une application web pour gérer l'ajout, la suppression et la modification des espèces du JEB

X- Webographie

Dabit, Nader. *React Native in Action*. Manning Publications, 2019.

“Documentation.” *Getting to Know Expo - Expo Documentation*,
docs.expo.io/versions/v28.0.0/.

“Documentation.” *Google*, Google, firebase.google.com/docs.

“Getting Started · React Native.” *React Native Blog ATOM*, facebook.github.io/react-native/docs/getting-started.

“Getting Started · React Navigation.” *React Navigation*, reactnavigation.org/docs/en/getting-started.html.