

Book de créations

Florian Michel

Avant-propos

Ce livret a été créé entièrement par Florian Michel résidant à Lyon. J'atteste de la totale véracité des propos tenus dans ce livret et atteste également être l'auteur de tous les projets présentés et disposant de tous les droits de propriété sur ceux-ci (sauf projets de groupe et projets dont la propriété relève de l'agence «Paprika Annecy»). Les projets exposés le sont par ordre de réalisation chronologique. Ces créations et ce livret sont présentés dans le cadre d'une reprise d'activité. Le contenu a été sciemment limité afin de ne présenter que les projets aboutis ou ayant un intérêt pour l'entité (entreprise ou client). D'autres projets peuvent être vus dans la section «réalisation» du site : <http://florianmichel-online.com>

D'autre part, les démonstrations techniques se feront principalement autour du langage Javascript auquel j'attache un certain intérêt. D'autres extraits de code, tel que le PHP, le HTML, le CSS ou encore le SASS, peuvent être vus directement via mon espace Git Hub : <https://github.com/FM-O>

Sommaire

Avant-propos.....	Page 3
Site web «florianmichel-online.com».....	Page 6-7
Application web «Lave & Ris».....	Page 8-9
Application mobile «Edible».....	Page 10-11
Site web «Lagrange».....	Page 12-13
Site web «Deep Nature».....	Page 14-15
Annexe 1.....	Page 16
Annexe 2.....	Page 17

Quel projet ?

Ce projet a été initialement développé et prévu pour la mise en place de mon site personnel afin de me créer une présence en ligne. Je suis d'abord parti d'une conception maquette (voir illustration page ci-contre), pour ensuite l'intégrer à l'aide des langages descriptifs HTML et CSS 3 dans un premiers temps. J'y ai ensuite ajouté du dynamisme avec des scripts Javascript que j'ai moi-même écrits. Enfin, j'ai fait évoluer mon formulaire de contact en Ajax (Javascript natif) afin d'éviter le rechargement de la page, indispensable pour la conception d'un site «One page» comme le mien.



Retrouvez un extrait de code du projet en annexe 1

Ressources & technologies

Environnement de développement

IDE : IntelliJ V13 (jetBrains)

Langages : HTML5 / CSS3 / Javascript / PHP

Plugins : Magnific Popup / FullPage JQuery

Compatible tablette / mobile : Oui

Logiciels de conception graphique : Photoshop / illustrator (CS6)

Liens utiles

Repository Git Hub du projet :

https://github.com/FM-O/site_v2

Site Actuellement en ligne :

<http://florianmichel-online.com>

Chiffres clés



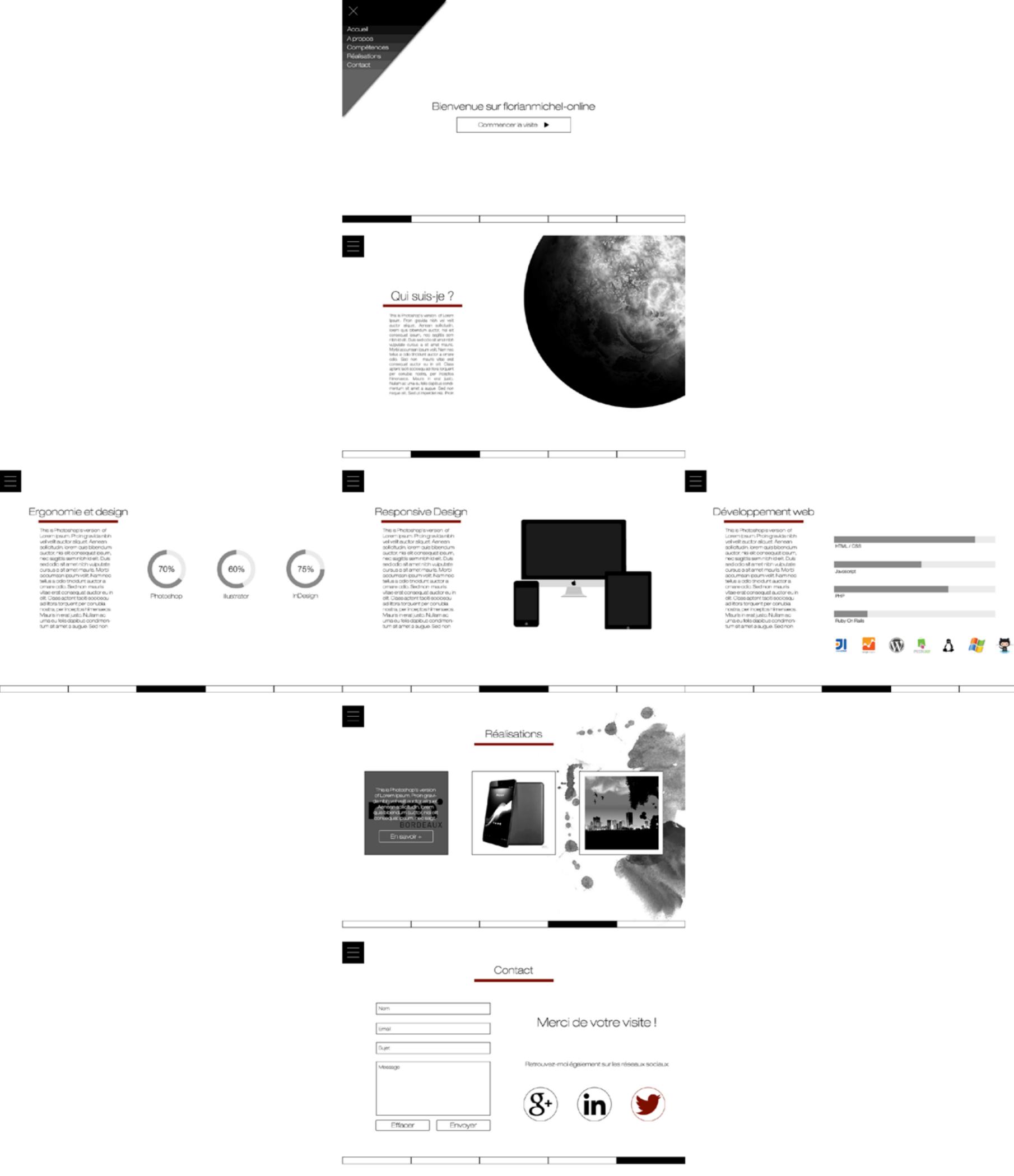
500H
de travail

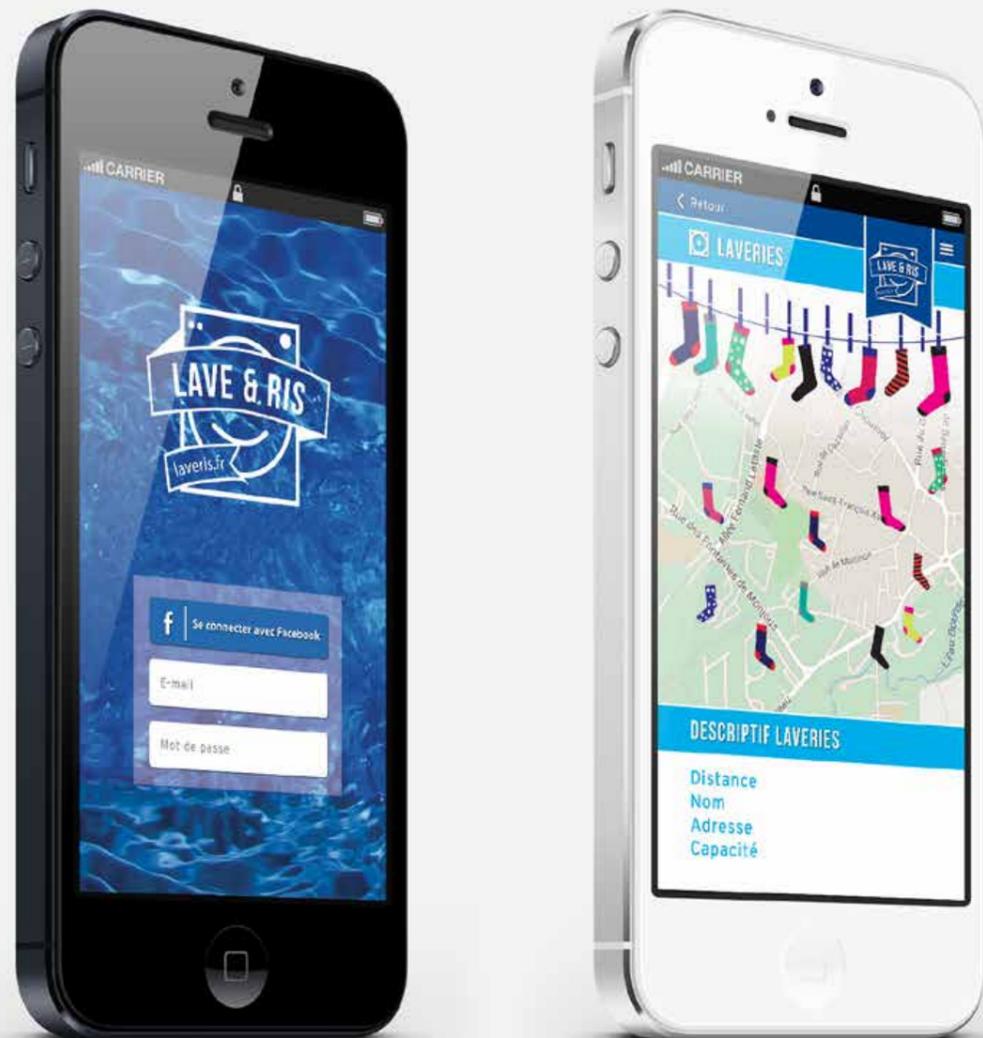
JS

953
lignes de Javascript



4065
lignes de CSS





Quel projet ?

L'application web Lave & Ris a été réalisée dans le cadre de la formation MMI en coopération avec un professionnel intervenant. L'exercice consistait à créer une application native ou web en partant d'un concept innovant. Une contrainte technologique nous était imposée pour les applications web : l'utilisation du langage Ruby On Rails.

Notre idée

Àu départ notre projet était de créer une application qui permette d'organiser des événements (jeux de société, apéritifs conviviaux, etc.) dans les laveries automatiques de Bordeaux. L'objectif était de créer un petit réseau social autour d'une activité qui peut paraître ennuyeuse dans le quotidien des clients de laveries.

Fonctionnement

La connexion à l'application est gérée via Facebook. L'utilisateur a ensuite accès à une carte de Bordeaux recensant les différentes laveries automatiques de la ville. Il peut également ajouter une laverie en renseignant ses coordonnées. Sur la page événements, il peut ajouter ses propres événements, mais également participer à ceux des autres.

Mon rôle

Nous étions deux développeurs sur ce projet. Mon rôle consistait en l'intégration des maquettes sur le prototype de la maquette en Rails. J'ai également développé une partie des maquettes en statique (HTML / CSS).

Ressources & technologies

Environnement de développement

IDE : IntelliJ V13 (jetBrains)

Langages : HTML5 / CSS3 / Ruby On Rails

Plateforme de déploiement : Heroku

Application native : Non

Etat actuel : déployée

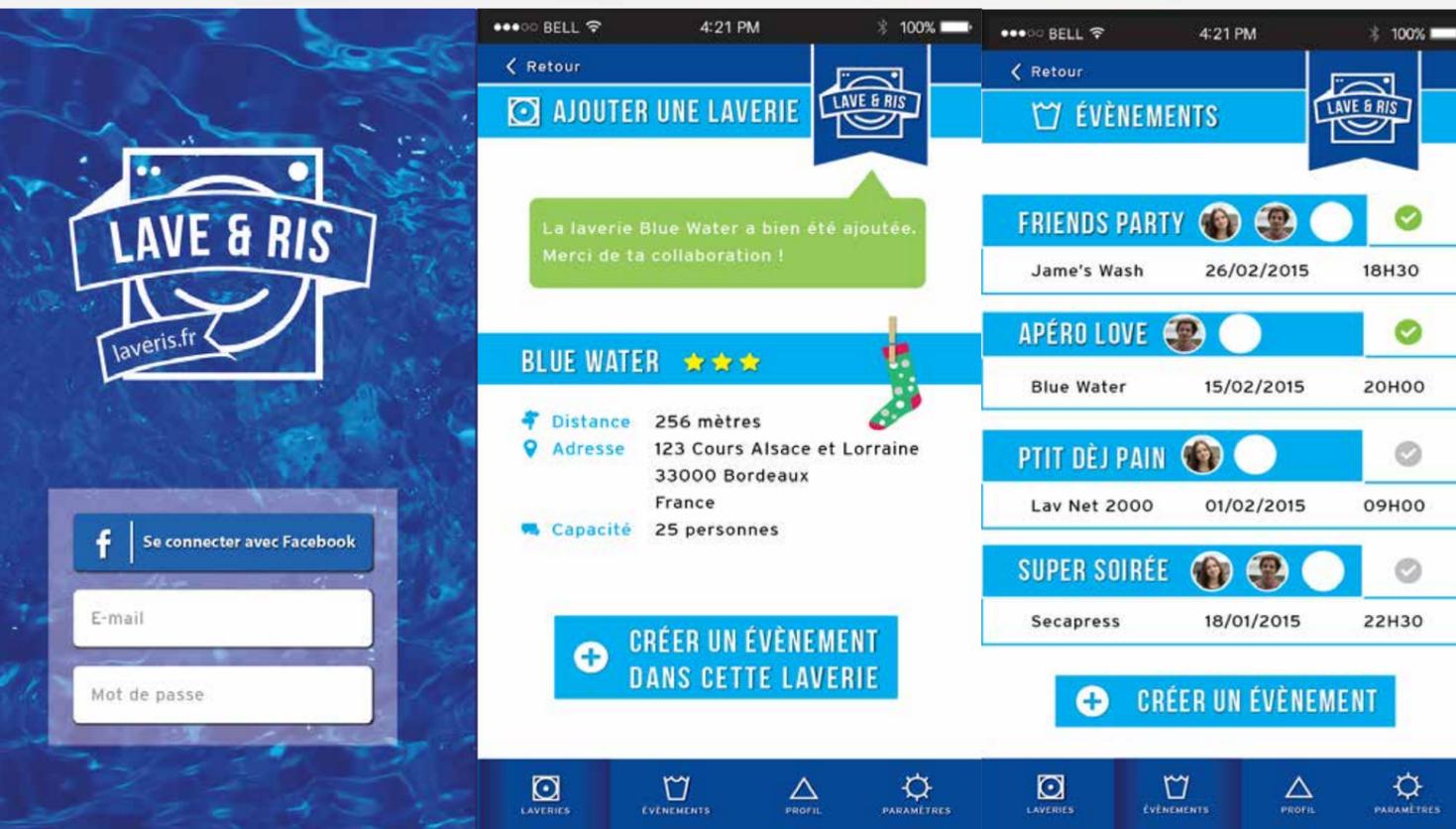
Liens utiles

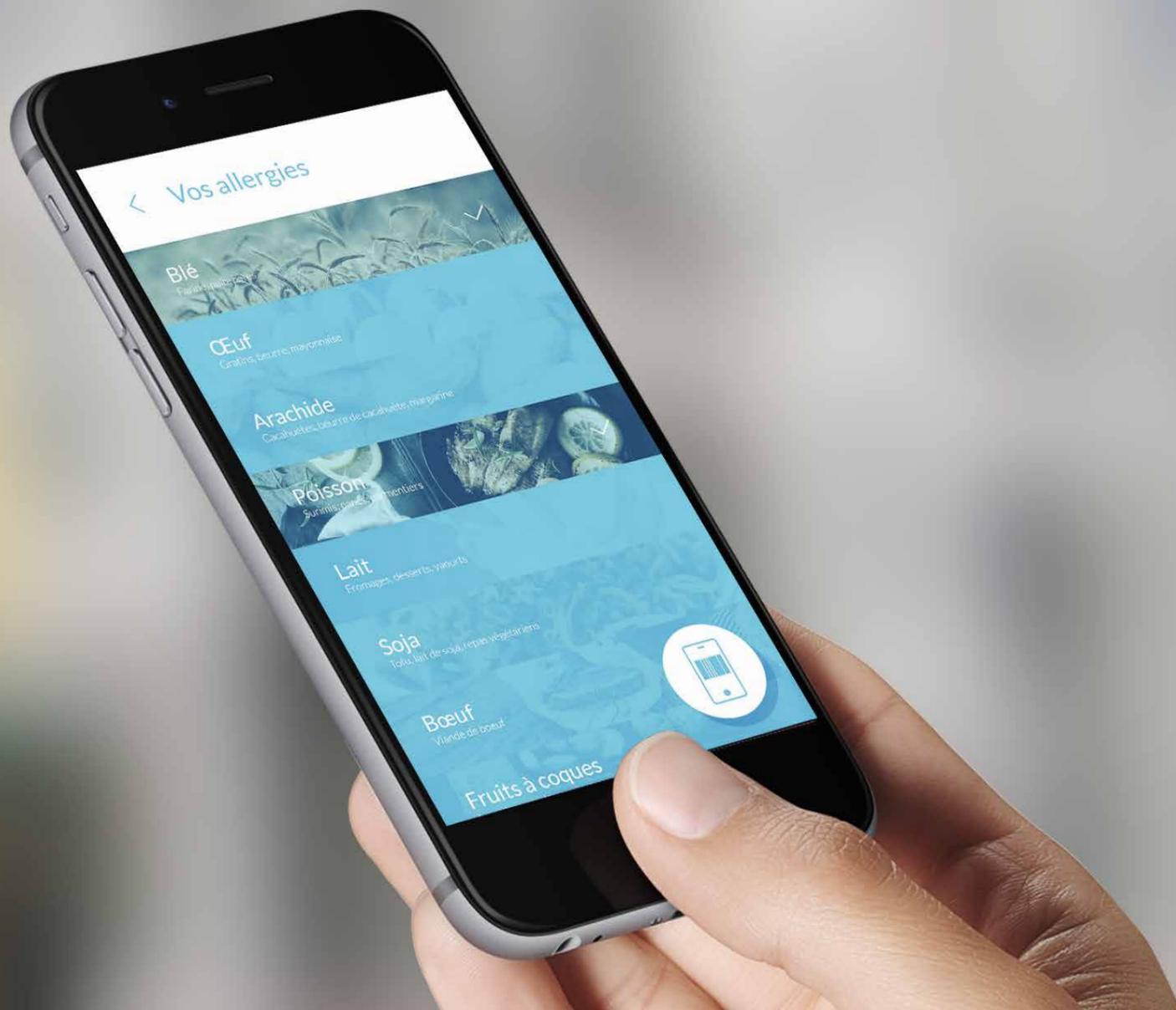
Repository Git Hub du

projet : [https://github.com/](https://github.com/YannBergonzat/Laveris)

[YannBergonzat/Laveris](https://github.com/YannBergonzat/Laveris)

Version live : <http://app.laveris.fr>





Quel projet ?

Ce travail a été réalisé dans le cadre de la semaine «mimi workshop» organisée au sein de l'IUT Michel de Montaigne. L'objectif de cette semaine était de constituer des groupes et de trouver des idées novatrices en matière de technologies et d'en créer un prototype fonctionnel.

Notre idée

L'idée que nous avons voulu développer s'appuie sur le monde de l'hygiène alimentaire et touche directement le monde de la santé. En effet, nous connaissons déjà l'application Open Food Facts qui permet de scanner un aliment et de savoir ce qu'il contient. En s'appuyant sur ce concept nous avons voulu développer une application qui permette, en scannant un aliment, de savoir s'il est dangereux pour un consommateur ayant des allergies alimentaires qu'il aura sélectionné au préalable.

Fonctionnement

En arrivant sur l'application vous sélectionnez les allergies qui vous correspondent (photo 1) et appuyez sur la touche «scan», vous scannez et l'application renvoie la réponse s'il y a danger ou non (photo 2). Le matching est réalisé à l'aide de l'API d'Open Food Facts

Mon rôle

Au sein de ce projet, j'ai été responsable du développement de l'application. Je me suis concentré sur le développement «front» de l'application, tandis qu'un développeur «back end» s'occupait de la partie fonctionnelle et du matching allergènes /aliments.

Ressources & technologies

Environnement de développement

IDE : IntelliJ V13 (jetBrains)

Langages : HTML5 / CSS3 / Javascript / PHP

Technologie de compilation mobile : PhoneGap

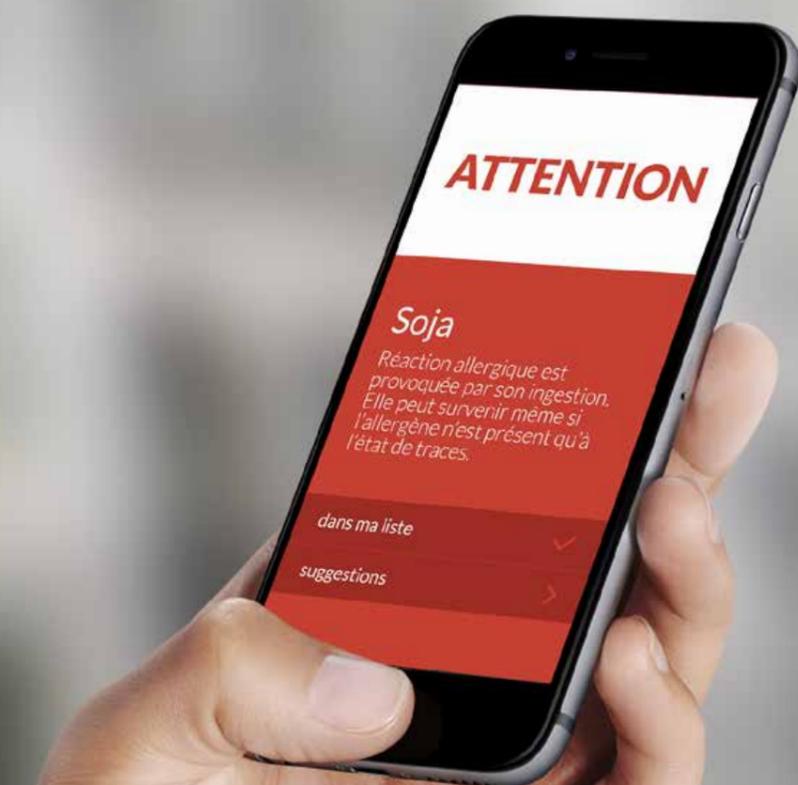
Application native : Oui

Etat actuel : prototype

Liens utiles

Repository Git Hub du projet :

<https://github.com/FM-O/Edible>



ATTENTION

Soja

Réaction allergique est provoquée par son ingestion. Elle peut survenir même si l'allergène n'est présent qu'à l'état de traces.

dans ma liste ✓

suggestions >



L'application n'est actuellement pas disponible sur le play Store en raison de la suspension de son développement

Extrait de code en annexe 2



Les produits Lagrange LES GAMMES



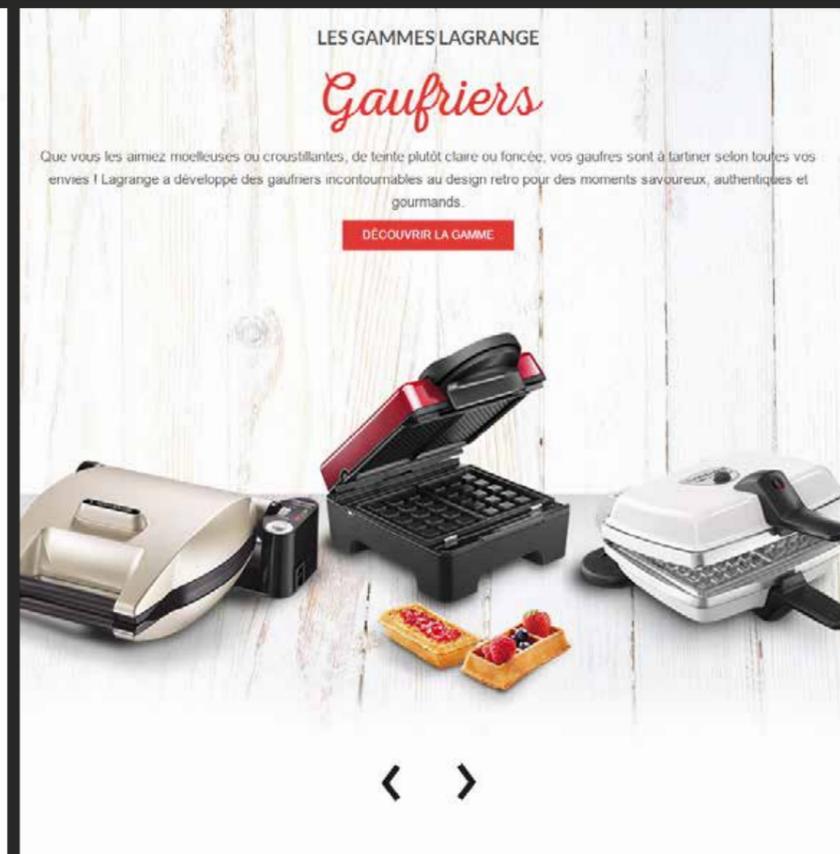
Quel projet ?

Ce projet a été initié par l'agence Paprika pour la refonte du site web de Lagrange à la demande du client. Suite à cela une phase de réflexion a été engagée avec toute le pôle web de l'agence afin de mener à bien cette nouvelle mission.

Organisation

L'organisation du projet s'est fait autour de l'existant. Ainsi il a fallu faire un état des lieux de ce qui existait déjà sur le site actuel. S'en est suivi la classique -mais néanmoins indispensable- période de brainstorming autour de l'UX / UI du site et du webdesign qui en découlerait. L'équipe a ensuite fait la proposition de plusieurs pistes au client afin de prendre une décision finale.

Site vitrine



Mon rôle

Mon rôle au sein du projet a été d'entamer la phase de réflexion avec l'ensemble de l'équipe au départ. Lors du lancement de la phase de développement j'ai réalisé plusieurs «modules» Javascript (notamment un carousel de produits), mais plus largement l'ensemble des animations du site.

Spécificités

La plupart des caractéristiques spécifiques du site Lagrange se trouvent en BackOffice avec le développement de plusieurs modules spécifiques afin que le client puisse mieux gérer l'ensemble du site. J'ai été responsable du développement de quelques fonctionnalités côté interface comme le carousel de produits.



Pour chaque paragraphe concernant les spécificités d'un projet, il s'agit de fonctionnalités dont le développement était entièrement sous ma responsabilité.

Ressources & technologies

Environnement de développement

- IDE : IntelliJ V13 (jetBrains)
- Langages : HTML5 / CSS3 / Javascript
- CMS : Prestashop
- Etat actuel : Site en ligne

Quel projet ?

Deep Nature était déjà un projet réalisé par l'agence Paprika à mon arrivée. Suite au renouvellement de l'équipe web, il a été décidé en accord avec le client une refonte complète du site. Le site a bénéficié de très nombreuses améliorations tant sur le plan technique que graphique.

Organisation

Ce projet s'est déroulé en plusieurs temps. Une phase de réflexion a été engagée par la directrice de création de l'agence. Puis suite à la livraison des maquettes par le webdesigner et la validation côté client, le projet a débuté. Lors du développement du projet les phases de retour client et le bon suivi ont permis de mener à bien celui-ci.

Mon rôle

À l'instar de nombreux projets, ma responsabilité pour le développement de Deep Nature était concentrée autour de l'interface utilisateur (que ce soit pour l'intégration ou le développement de fonctionnalités). J'ai également pu intervenir au niveau back office en coopération avec le chef de projet pour ajouter des fonctionnalités. Enfin j'ai été chargé du développement de plusieurs pages statiques.

Spécificités

Les spécificités de Deep Nature sont principalement concentrées sur le fonctionnement du menu (full page). Celui-ci utilise un déploiement en 3 volets maximum, le visuel est au cœur de l'UX. Une autre fonctionnalité spécifique du site est le «localisateur de spa» qui fonctionne grâce à un appel Ajax à l'API de Google. J'ai été responsable du développement de l'intégralité du menu ainsi que du localisateur.

Ressources & technologies

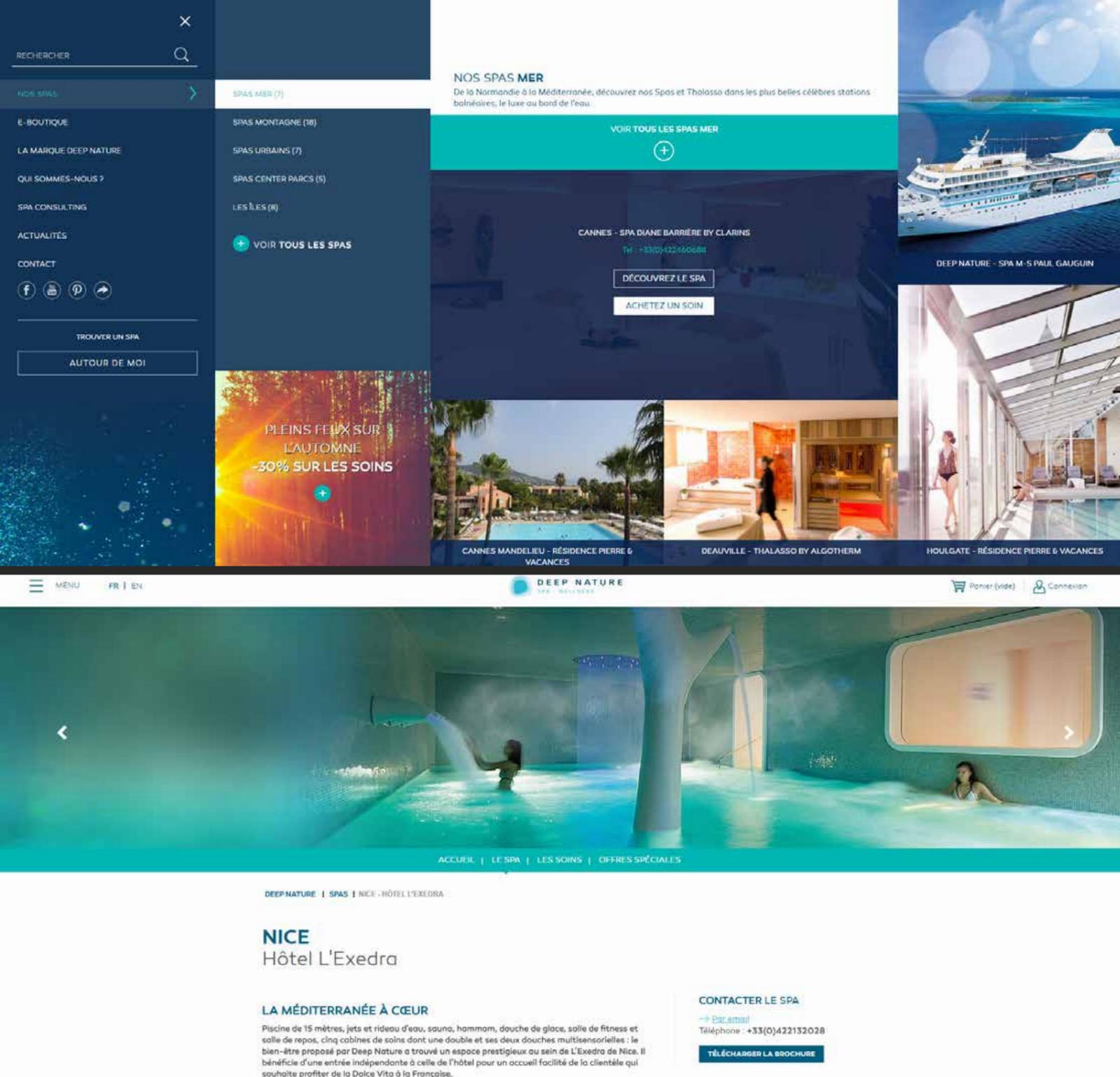
Environnement de développement

IDE : IntelliJ V13 (jetBrains)

Langages : HTML5 / CSS3 / Javascript

CMS : Prestashop

Etat actuel : Site en ligne



“Script de gestion du formulaire de contact en Ajax”

```

var xhr = new XMLHttpRequest();

xhr.open('POST', 'mail.php');

xhr.setRequestHeader("Content-Type", "application/x-www-form-
urlencoded");

xhr.onreadystatechange = function() {
  if (xhr.readyState == 4 && xhr.status == 200) {

    //remove loader
    loader.style.display = "none";

    var response = xhr.responseText,
    resp = JSON.parse(response);

    if (resp.error == 'true') {

      actual.style.color = "red";
      actual.replaceChild(errorMess, actual.firstChild);
    }
    overlay.style.display = "block";
    actual.lastElementChild.focus();
    downOverlay();
  } else if (xhr.readyState == 4 && xhr.status != 200) {

    //remove loader
    loader.style.display = "none";

    var err = document.getElementById('xhrErr');
    if (xhr.statusText != "") {
      var txtErr = document.createTextNode('erreur : ' + xhr.statusText);
      err.appendChild(txtErr);
      err.style.display = "block";
      setTimeout(function(){
        err.removeChild(txtErr);
      }, 7000);
    } else {
      err.innerHTML = 'Vérifiez votre connexion Internet';
      err.style.display = "block";
    }
    actual.style.color = "red";
    actual.replaceChild(errorMess, actual.firstChild);
    overlay.style.display = "block";
    actual.lastElementChild.focus();
    downOverlay();
  }
};

xhr.send('param1=' + value_name + '&param2=' + value_mail + '&param3=' +
value_subject + '&param4=' + value_message);
}

```

```

var app = {
  // Application Constructor
  initialize: function() {
    this.bindEvents();
  },
  // Bind Event Listeners
  bindEvents: function() {
    document.addEventListener('deviceready', this.onDeviceReady, false);
  },
  // deviceready Event Handler
  onDeviceReady: function() {
    app.receivedEvent('deviceready');
  },
  // Update DOM on a Received Event
  receivedEvent: function() {
    var test = document.getElementById('goscan');

    test.addEventListener('click', function(e) {
      e.preventDefault();
      console.log('scanning');
      cordova.plugins.barcodeScanner.scan(
        function (result) {
          alert("We got a barcode\n" +
            "Resultat: " + result.text + "\n" +
            "Format: " + result.format + "\n" +
            "Cancelled: " + result.cancelled);

          $.ajax({

            url: "http://api.edibleapp.fr/match/"+result.text+"/1", // l'url
            type: "POST", // la méthode
            data: "", // sérialisation de données : username=test&password=test
            dataType:'json', //type de données, permet de parser le JSON
            success: function(msg) {

              var main = document.getElementById('main'),
              main_skanok = document.getElementById('main_skanok'),
              main_scanok = document.getElementById('main_scanok');

              if (msg.result.matching.traces == "" && msg.result.matching.allergens == "") {

                var allergen = document.getElementById('allergen'),
                allergenName = document.createTextNode("Nutella");

                allergen.appendChild(allergenName);

                main.style.display = "none";
                main_skanok.style.display = 'block';

                alert("Nom du produit : "+msg.result.product.name);
              } else {

                main.style.display = "none";
                main_scanok.style.display = "block";

                alert("Nom du product : "+msg.result.product.name);
              }
            },
            error: function( jqXHR, textStatus, errorThrown ){
              alert("error:"+textStatus+errorThrown );
            }
          });
        },
        function (error) {
          alert("Scanning failed: " + error);
        }
      ), false);
    });
  }
};

app.initialize();

```

“Script principal de l'application, appelant les différentes fonctionnalités et le plugin de scan”