

PatteBlanche



Le monde s'inscrit aujourd'hui dans une période de transition et doit relever de nombreux défis environnementaux et sociétaux. Si l'avenir apparaît comme mouvementé, nous subissons déjà de grandes perturbations à travers le monde, tant sur le plan environnemental que politique.

C'est dans ce contexte que la notion de « résilience » prend son sens. C'est la capacité d'un organisme, d'une espèce ou d'un système à surmonter l'altération de son environnement. Si elle est une capacité innée des êtres vivants, c'est aussi un besoin essentiel au bien-être, et une faculté qui peut s'approfondir. Chaque humain ayant le besoin et le droit de pouvoir assurer son intégrité physique et psychologique.

Safe+ s'inscrit dans ce contexte de résilience, au sens large. Cette « application de survie » a pour vocation de permettre à ses utilisateurs de gagner en compétence et en résilience. Elle fournira des ressources permettant de faire face à des situations variées. Elle proposera un "guide de survie" hors ligne et synchronisable ainsi que des outils variés.

**Résilience - Survie - Transition - Application mobile
ReactNative**

DOSSIER DE PROJET

DÉVELOPPEUR WEB
WEB MOBILE

“La résilience, c'est l'art de naviguer dans les torrents.”

Boris Cyrulnik

2019 - 2020

CONTACT

Patte Blanche

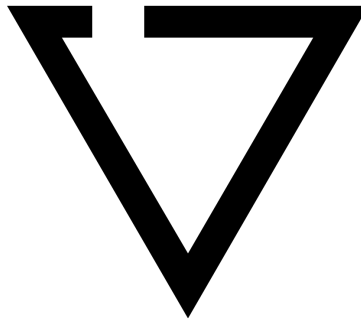
Patte Blanche

19 avenue de Toulouse
34070 Montpellier

04 99 74 08 83

contact@patte-blanche.com

www.patte-blanche.com



Valentin Besnardiere

Développeur Web - Mobile

06 23 70 17 64

valentin.besnardiere@gmail.com

www.valentin-besnardiere.com

REMERCIEMENTS

Je voudrais remercier les personnes qui m'ont entouré et qui ont contribué au succès de ce stage particulièrement riche et épanouissant, tant sur le plan professionnel que humain.

Je veux remercier Bruno Marie, mon maître de stage, développeur full-stack chez Patte Blanche, pour son accueil et sa confiance, et qui a su me laisser une grande autonomie, tout en étant à l'écoute et en me challengeant.

Je veux chaleureusement remercier toute l'équipe de Patte Blanche qui m'a réservé un accueil exceptionnel, et qui a fait de ce stage un moment de partage inoubliable. Je tiens à les remercier pour leur bienveillance et leur convivialité. Je remercie particulièrement mes voisins de bureau, Céline, Tiphaine, Vaea et Perrine pour leurs conseils et leur écoute. Je remercie également les coworkers de The Island, pour leur accueil chaleureux.

Je voudrais remercier l'équipe pédagogique de l'ADRAR, Laure Christol, Thomas Viviani, Marc Abed, Maximilien Garenne, Ludivine Crépin et Magali Tabarot pour leur accompagnement tout au long de cette année de formation.

Je veux aussi remercier les intervenants professionnels et en particulier Jean-Cristophe Dominguez pour son investissement et son aide précieuse.

Je tiens à remercier Florence Calmettes et Delphine Van Eerdewegh pour leur soutien précieux dans la dernière ligne droite vers le titre.

Enfin, et surtout, j'adresse mes remerciements à mes camarades, ma team Java. Je dois avant tout le succès de cette année à notre belle promotion, à notre entraide et à notre soutien: Vincent, Nicolas, Maurane, Eloïse, Lauréenne, Sofiane, Simon, Clément, Sylvain, Julien, Abdel, François.

RÉSUMÉ

Résilience - Survie - Transition - Application mobile - React Native

Safe+ est née de la rencontre entre Patte Blanche, un agence de communication, et moi-même développeur, chacun engagé dans la "transition sociétale".

Le monde s'inscrit aujourd'hui dans une période de transition et doit relever de nombreux défis environnementaux et sociétaux. Si l'avenir apparaît comme mouvementé, nous subissons déjà de grandes perturbations à travers le monde, tant sur le plan environnemental que politique.

C'est dans ce contexte que la notion de « résilience » prend son sens. C'est la capacité d'un organisme, d'une espèce ou d'un système à surmonter l'altération de son environnement. Si elle est une capacité innée des êtres vivants, c'est aussi un besoin essentiel au bien-être, et une faculté qui peut s'approfondir. Chaque humain ayant le besoin et le droit, de pouvoir assurer son intégrité physique et psychologique.

Safe+ s'inscrit dans ce contexte de résilience, au sens large. Cette « application de survie » a pour vocation de permettre à ses utilisateurs de gagner en compétence et en résilience. Elle fournira des ressources permettant de faire face à des situations variées. Elle proposera un "guide de survie" hors ligne et synchronisable ainsi que des outils variés.

Le format d' "application mobile" répond aux exigences d'un outil embarqué: elle fonctionne hors-réseaux, et permet d'interagir avec l'utilisateur et des capteurs.

Développée en React Native, l'application a fait l'objet d'une attention particulière sur l'interface et sur l'expérience utilisateur, primordiales pour un outil de terrain. Un back-end avec back-office WordPress a été mis en place pour fournir un système simple d'édition de contenu. Enfin des librairies créées à l'occasion du projet vont être publiées.

ABSTRACT

Resilience - Survival - Transition - Mobile App - React Native

Safe+ was born following a meeting of Patte Blanche, a communication agency and myself, web developer, each invested in the current societal transition.

The world is entering into a transitional phase and has to face many environmental and societal challenges. If the future appears to be troubled, we are going through some big changes all around the world, both political and environmental.

It is against that background that the term "resilience" becomes meaningful. It's the ability of a body, a species, or a system to survive alteration of his environment. If this ability is natural to living beings, it is also an essential point of the well being of oneself that can always be improved, as each human being needs to ensure their physical and psychological integrity, and has the right to.

Safe+ has been developed in this particular context of broader resilience. The aim of this "survival App" is to allow its users to improve their skills, concerning resilience. It will provide resources that can help facing different kinds of situations. It will offer an offline survival guide that can be synchronized, as well as various helpful tools.

The mobile app format meets the requirements of mobile tools : it's able to run offline and can interact with the user and sensors.

React Native based, this app had received a special attention regarding its interface and user experience, both significant elements to elaborate field tools. A back-end and a WordPress back-office has been settled in order to provide a simple content editing system. Finally, libraries that have been created for the purposes of the app will be published.

COMPÉTENCES MISES EN OEUVRES

Développement

Développement d'une application mobile en React Native, avec back-end
Réalisation d'un back-office avec Wordpress, et utilisation de l'API WordPress.

Utilisation d'outils de Versioning (Git)

Front-end et web design

Création de l'identité d'une application

Réalisation d'une charte graphique

Maquettage d'une application

Web design d'une application

Graphisme

Gestion de projet

Élaboration d'un cahier des charges

Utilisation d'outils de gestion de projet et de collaboration (Trello, Slack...)

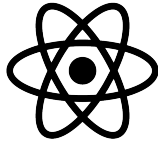
Chefferie de projet

Maquetter une application - Développer une interface utilisateur web dynamique - Réaliser une interface utilisateur avec une solution de gestion de contenu ou e-commerce - Créer une base de données - Développer la partie back-end d'une application web ou web mobile

Safe+' STACK



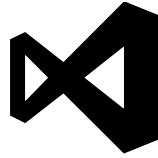
JavaScript



React-Native



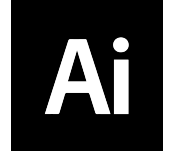
Wordpress



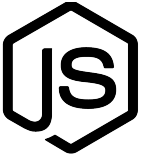
Visual Studio



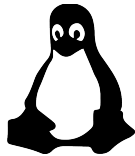
Photoshop



Illustrator



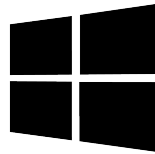
Node



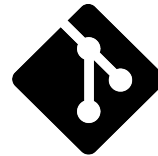
Ubuntu



MySQL



Windows



Git



Trello

SOMMAIRE

Contact.....	i
Remerciements.....	ii
Résumé.....	iii
Abstract.....	iv
Compétences mises en oeuvres	v
I. Aperçu général et cadre du projet.....	1
1. Patte blanche une entreprise à mission.....	1
2. Contextualisation : transition et résilience.....	2
II. Cahier des charges.....	3
1. Présentation du projet.....	3
2. Identité et naming.....	9
3. Charte graphique.....	10
4. Spécifications fonctionnelles.....	19
5. Spécifications techniques.....	31
6. Administration.....	38
7. Sécurité.....	40
8. Evolution et suivi de l'application.....	41
9. Organisation du projet.....	41
III. Présentation documentée de quelques fonctionnalités	43
1. Extraits sélectionnés de code commentés.....	43
2. Présentation d'un test effectué sur le « synchroniser ».....	69
3. Exemple d'une recherche anglophone effectuée.....	72
Bilan du projet.....	77
Conclusion du stage.....	78
Lexique.....	79

I. APERÇU GÉNÉRAL ET CADRE DU PROJET

1. PATTE BLANCHE UNE ENTREPRISE À MISSION

Patte Blanche est « une entreprise à mission », elle fait partie de ces nouvelles entreprises qui définissent statutairement, en plus du but lucratif, une finalité d'ordre social ou environnemental.

Depuis 2007, cette agence de communication est spécialisée en développement durable, RSE¹ et transition sociétale. Elle a pour vocation d'engager les communautés des grandes entreprises, fondations et ONG comme levier de transformation. Nouvellement entrée en gouvernance partagée³, cette agence est également certifiée B-Corp².

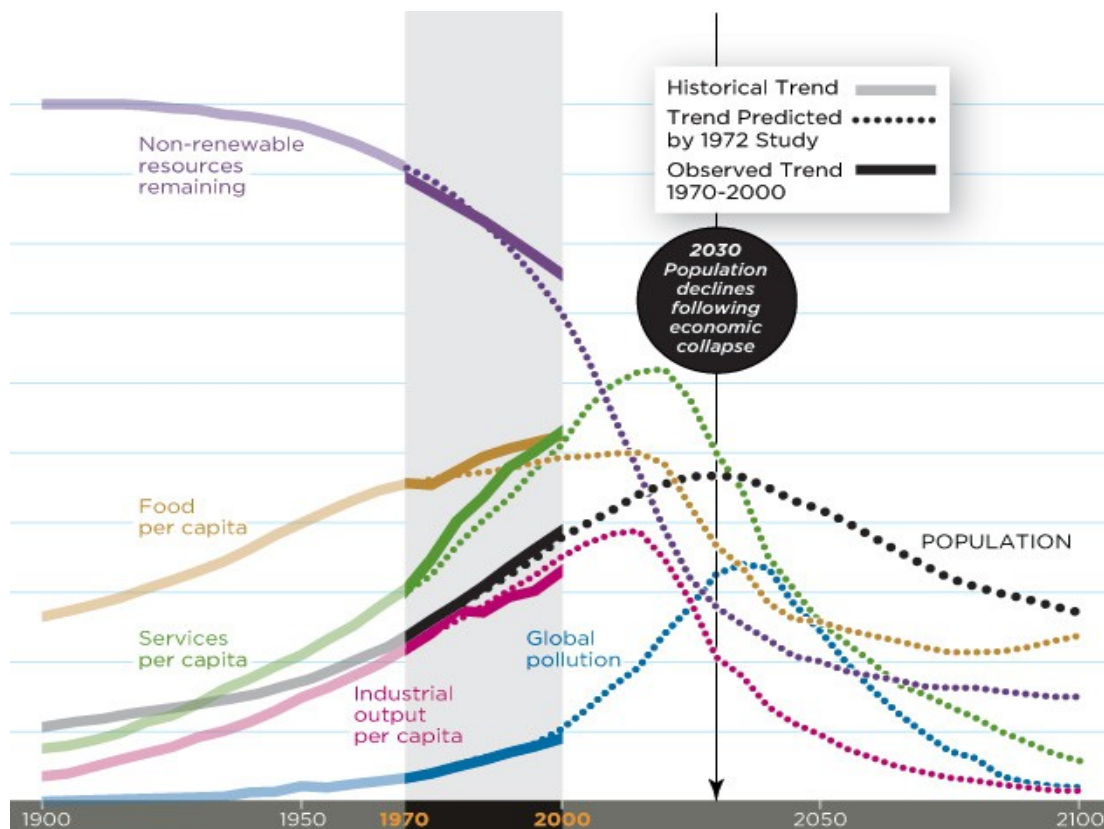
Le pôle Web est composé de Jonathan Boutonnet, chef de projet, Céline Martin, web designer, et Bruno Marie, Développeur Fullstack. L'offre web proposée par patte blanche est essentiellement la création de thèmes sur mesure sur la base du CMS Wordpress.



Organigramme de Patte Blanche

2. CONTEXTUALISATION : TRANSITION ET RÉSILIENCE

En 1970, le « Club de Rome⁴ » commande à des chercheurs du MIT un rapport portant sur l'évolution économique et démographique du monde. Pendant deux ans les chercheurs vont mettre au point « world⁵ » un modèle informatique de type « dynamique des systèmes » permettant de simuler les interactions entre population, croissance industrielle, production de nourriture et limites des écosystèmes terrestres. En 1972 est publié le « Rapport Meadows » aussi connu sous le nom « the limits of growth ». Révisé plusieurs fois, le modèle s'est avéré fiable et s'est imposé comme un modèle de référence, il prévoit un effondrement de la civilisation industrielle telle qu'on la connaît aujourd'hui.



Réactualisation des courbes de simulation extraites du « Rapport Meadows »

Le monde s'inscrit aujourd'hui dans cette période de transition et doit relever de nombreux défis environnementaux et sociétaux. Suite à ce constat, aujourd'hui indéniable, de nombreux scientifiques et collectifs s'efforcent d'imaginer et d'organiser cette transition qui s'effectue dans un contexte où l'avenir est incertain, non linéaire, et qui va subir des grandes perturbations, qu'elles soient environnementales ou politiques.

C'est dans ce contexte que la notion de « résilience » prend son sens. C'est la capacité d'un organisme, d'une espèce ou d'un système à surmonter l'altération de son environnement. Si elle est une capacité innée des êtres vivants, c'est aussi un besoin, essentiel au bien-être, et une faculté qui peut s'approfondir. Chaque humain ayant le besoin et le droit, de pouvoir assurer son intégrité physique et psychologique.

II. CAHIER DES CHARGES

1. PRÉSENTATION DU PROJET

a. OBJECTIFS DU PROJET

Safe+ s'inscrit dans ce contexte de résilience, au sens large. Cette « application de survie » a pour vocation de permettre à ses utilisateurs de gagner en compétence et en résilience. Elle leur fournira des ressources leur permettant de faire face à des situations variées. Ses principaux objectifs seront de :

- Fournir de la documentation et outils permettant de faire face à une situation dégradée, de rupture de la normalité (catastrophes naturelles, perte du domicile, menace de l'intégrité physique ...), et où les ressources sont limitées.
- Fournir des informations relatives aux premiers secours.
- Fournir des outils de sécurisation et d'organisation pour le cadre familial ou proche.
- Fournir de la documentation et des outils facilitant et sécurisant la survie en milieu naturel ou urbain.
- Fournir de la documentation sur l'art d'utiliser la nature au quotidien.
- Fournir de la documentation sur les low-tech⁶ et l'art de subsister avec des ressources limitées.

La nature intrinsèque de l'application nécessitera de réfléchir de façon cohérente les besoins d'accès au réseau internet et GSM. L'accès aux ressources pourra ainsi se faire de façon hors-ligne.

b. CIBLES

Safe+ est une application qui pourra être utile à une large part de la population, mais c'est un public déjà sensibilisé au sujet qui sera en premier lieu intéressé :

- Les personnes prévoyantes, souhaitant gagner en résilience et se sentir moins vulnérables.
- Les personnes exposées régulièrement à des situations difficiles (catastrophes naturelles, instabilité politique...)
- Les personnes désireuses de s'impliquer dans la transition de notre mode de vie.
- Les personnes s'intéressant au « survivalisme⁷ », au « bushcraft⁸ » ou à la « collapsologie⁹ ».

Au delà de ces cibles spécifiques l'application intéressera également :

- Les personnes pratiquant ou voulant pratiquer le bivouac, la randonnée, le séjour en pleine nature.
- Les personnes désireuses d'acquérir des connaissances et savoir-faire pour utiliser la nature au quotidien (cuisine, médication).
- Les personnes voulant s'instruire ou s'évader en lisant de la documentation « d'aventurier ».

c. FONCTIONNALITÉS PRINCIPALES

Safe+ s'articulera autour de deux grands axes : de la documentation proposée sous forme d'un manuel de survie, et des outils qui exploiteront la technologie de nos smartphones.

LA DOCUMENTATION

Le guide de survie sera réparti en différentes catégories thématiques. Il sera amené à s'étoffer avec le temps mais comprendra initialement les catégories suivantes :

Start-Up	La trousse de survie	Compétences	Eau et alimentation	Matériel
* Notice de l'application	* Le fonctionnement par strates	* Se former aux premiers secours	* L'eau	* Objets tranchants
* Introduction	* EDC: Every Day Cary	* S'abriter	* Les plantes comestibles	* Les vêtements
* Philosophie et réflexions	* Sac de survie	* Faire du feu	* Les champignons	* Le smartphone
	* Kit d'extraction	* Cuisiner	* La pêche	* L'éclairage
		* S'orienter	* Chasse et piégeage	* L'électronique
		* Utiliser des objets tranchants		
		* Apprendre les nœuds		
		* Reconnaître les traces		
		* Produire de l'électricité		

Des catégories supplémentaires seront par la suite ajoutées, telles que « Quitter le milieu urbain », « Communiquer »...

La documentation se verra très fonctionnelle. De nombreuses illustrations et schémas viendront appuyer le propos en montrant des détails techniques.

Cette documentation devra être accessible de façon hors-ligne. Cependant, une actualisation du contenu pour obtenir les nouveautés pourra se faire via une connexion internet. Des explications détaillées sont données aux parties II)4)c « Fonctionnalités » et II)5)a « Une application hors-ligne et synchronisable ».

En plus de la documentation, **Safe+** tirera partie de la technologie portée par les smartphones modernes en utilisant la partie matérielle et les capteurs pour fournir des outils à ses utilisateurs. Des outils seront rajoutés au fil du temps pour venir enrichir l'application. Initialement l'application embarquera :

L'ALERTE SMS GÉOLOCALISÉE

L'alerte SMS géolocalisée sera une fonctionnalité centrale de **Safe+**. Elle permettra à tout moment d'envoyer facilement un SMS Géolocalisé au service d'urgence (114) ainsi qu'à une liste de contacts prédéfinie.

L'utilisateur disposera d'une interface qui lui permettra de paramétrer ce service d'alerte. Il pourra éditer préalablement ses informations personnelles : son nom, prénom et un message personnalisé. Il pourra renseigner une liste de destinataires qui comprendra par défaut le 114.

Une fois le service paramétré, il suffira de deux clics à l'utilisateur pour envoyer un SMS groupé contenant ses informations personnelles, ses coordonnées GPS, et un lien Google maps affichant sa position.

Le positionnement pourra se faire de façon complètement « hors-ligne » en utilisant les réseaux satellitaires des GNSS¹⁰ supportés par le téléphone. La quasi totalité des smartphones supporte le système américain GPS, une grande partie peut aussi exploiter le système russe GLONASS, ou chinois BEIDOU pour augmenter la qualité du positionnement. Une proportion moindre supporte déjà le nouveau système européen GALILEO. La précision de ce positionnement peut atteindre assez facilement les 3m.

En plus de ce positionnement hors-ligne, le système pourra aussi s'appuyer sur la triangulation (via les relais gsm environnants) en cas de faible couverture satellitaire (espace couverts, bâtiments...)

L'envoi du sms ne nécessitera quant à lui qu'une connexion GSM dans la bande des 900 – 1800 mhz largement diffusée dans le monde et qui couvre 95% de la surface du territoire français.

LA BOUSSOLE ET LE GPS

L'application proposera une boussole qui s'appuiera sur le magnétomètre du téléphone ainsi qu'un GPS fonctionnant de la façon décrite plus haut.

LA LAMPE TORCHE

La boîte à outil comprendra également une lampe torche facilement accessible qui utilisera le flash LED du smartphone. Elle offrira une grande visibilité sur une portée conséquente.

LA LAMPE DISCRÈTE

La lampe discrète de **Safe+** a été spécifiquement réfléchi pour une utilisation discrète. Utilisant l'écran comme source de lumière, elle possède différents avantages :

Conserver l'acuité oculaire dans l'obscurité

Au bout de quelques minutes passées dans l'obscurité notre pupille se dilate et les bâtonnets, les cellules les plus sensibles qui tapissent le fond de notre rétine, se mettent en action à la place des cônes qui nous servent en vision diurne. Si nous perdons notre vision des couleurs (assurée par les cônes), nous gagnons énormément en visibilité car les bâtonnets sont 1000 fois plus sensibles à la lumière. L'œil s'adapte.

L'utilisation d'une lumière vient annuler cette adaptation naturelle. Les bâtonnets se désactivent et nous ne voyons plus du tout les zones non éclairées. En outre, une fois la lampe éteinte il nous faut plusieurs minutes pour récupérer la vision nocturne. Voir graphique n°1 « Seuils de sensibilité des cônes et bâtonnets dans l'obscurité chez les Humains » de la page « Étude de la lumière émise par **Safe+** » ci-après.

Cependant, le spectre de fréquence perçu par les bâtonnets est plus réduit que celui visible par les cônes (voir les graphiques n°3 et 4), ainsi au delà de 600 nm les bâtonnets sont insensibles et ne perçoivent plus la lumière.

Safe+ exploite cette « faille » en utilisant exclusivement les diodes rouges de l'écran LED dont le spectre s'étale entre 600 et 700 nanomètres. Si cela ne permet pas de voir les couleurs (seuls les cônes « rouges » sont activés) cela permet de tromper l'œil qui se croit dans l'obscurité et active ses bâtonnets (voir graphique 8). Avec cette lampe, nous voyons donc très bien ce qui est éclairé mais aussi ce qui est dans l'obscurité. A l'extinction de la lampe, nous voyons comme si nous avions toujours été dans le noir.

C'est un grand avantage pour évoluer dans l'obscurité, il est possible d'éclairer un instant son environnement proche (une carte, un outil) tout en gardant une vision périphérique et lointaine optimale, et sans devoir se réadapter ensuite.

Invisibilité aux animaux

Mis à part quelques primates, les animaux ne possèdent pas la vision trichromatique qu'ont les Humains. Ils ne possèdent que les cônes « bleus » et « verts ». Leur spectre visible est réduit et s'arrête là aussi au seuil des 600 nanomètres, au delà duquel ils sont aveugles (voir le graphique n°4 « Réponse normalisée des cônes de l'œil des Animaux »). La lumière de la lampe discrète de **Safe+** leur est donc invisible.

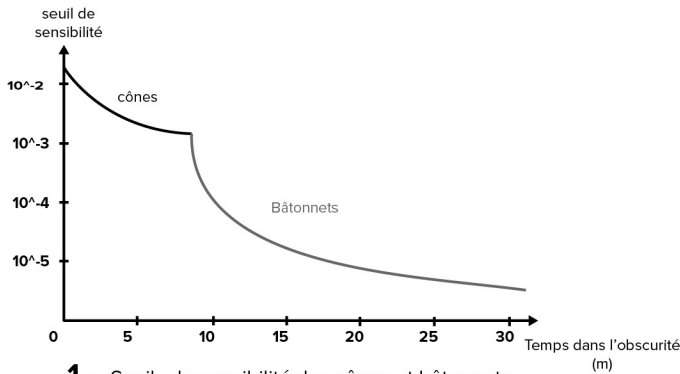
C'est là aussi un réel avantage, pour l'observation des animaux, la chasse ou pour simplement éviter d'attirer l'attention d'un chien.

Conservation de la batterie

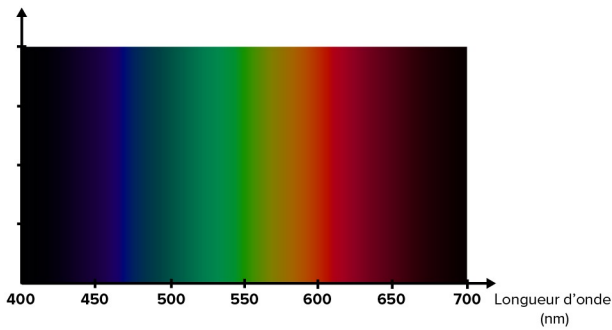
L'écran LED du téléphone est économe en batterie, sa lumière douce et diffuse peut être utilisée longtemps sans trop décharger la batterie. Un variateur permet de régler l'intensité selon le besoin.

Étude de la lumière émise par Safe+

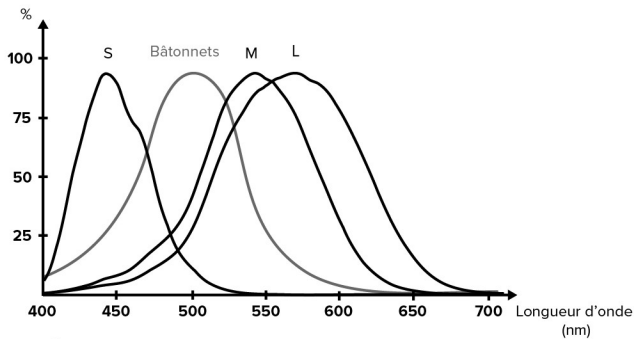
Comparaison de la vision des Humains avec les autres animaux (dichromates)



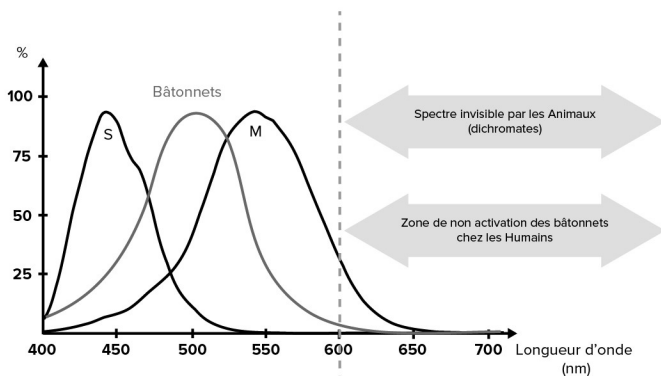
1 - Seuils de sensibilité des cônes et bâtonnets dans l'obscurité chez les Humains



2 - Spectre lumineux du visible

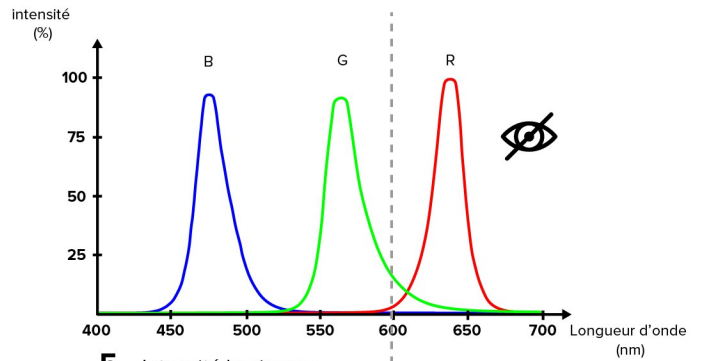


3 - Réponse normalisée des cônes et bâtonnets de l'oeil des Humains

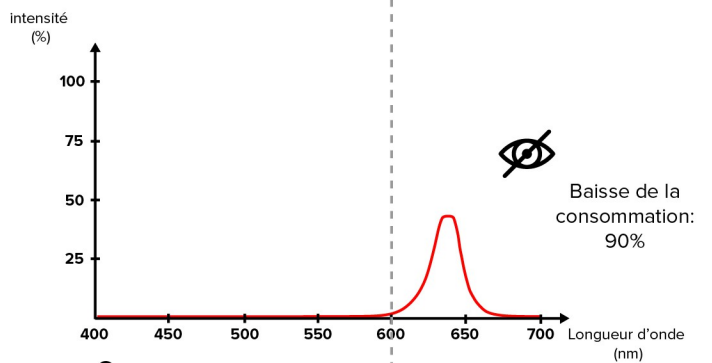


4 - Réponse normalisée des cônes de l'oeil des Animaux

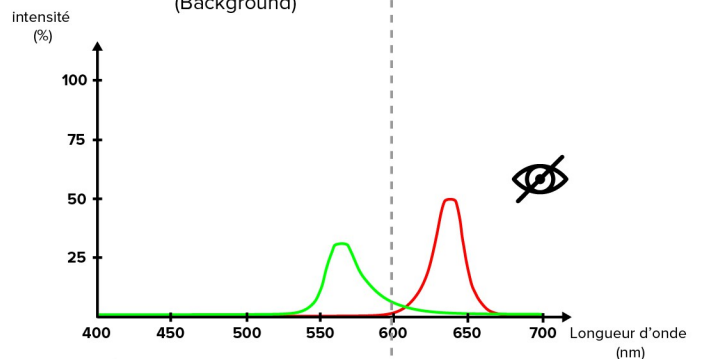
Comparaison qualitative et quantitative de l'intensité lumineuse émise par Safe+ en fonction du mode



5 - Intensité lumineuse émise par un écran led (blanc)

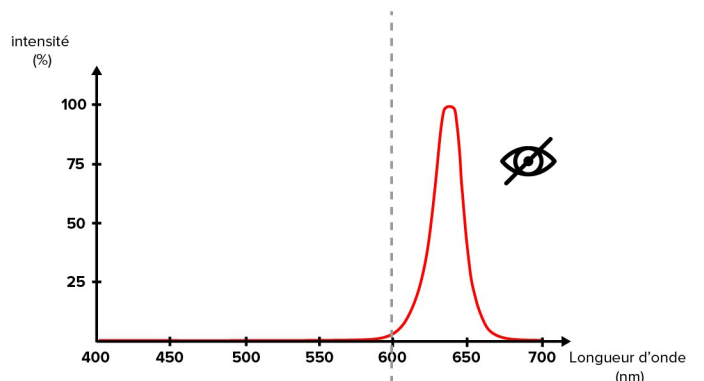


6 - Intensité lumineuse émise par Safe+ en mode nuit (Background)



7 - Intensité lumineuse émise par Safe+ en mode nuit (Images filtrées)

Intensité lumineuse émise par la lampe discrète de Safe+



8 - Intensité lumineuse maximale émise par la lampe discrète

LE MODE NUIT

L'application proposera un mode "nuit » qui s'appuiera sur la même technique que la lampe décrite plus haut. En activant le mode nuit, la luminosité sera baissée, les fonds et les textes changeront de couleur, les images seront filtrées (voir les graphiques 5,6 et 7 ci-dessus). L'utilisateur pourra ainsi consulter son application sans dégrader sa vision nocturne, en économisant de la batterie et en étant invisible aux animaux.

LA SYNCHRONISATION

De part sa destination, Il est primordial que l'application puisse être utilisée sans nécessiter une connexion au réseau. Elle doit pouvoir être fonctionnelle dans des lieux reculés, ou dans des situation où le réseau est détérioré (catastrophes naturelles, blocage politique).

Cependant, afin de pouvoir assurer un contenu de qualité, et de plus en plus riche, l'application pourra selon la volonté de l'utilisateur se connecter à l'API du back-end pour se synchroniser et télécharger les nouveaux contenus de façon persistante sur le téléphone.

d. ACCESSIBILITÉ ET CONTRAINTES D'INTÉGRATION

A terme, **Safe+** sera disponible sur les deux systèmes d'exploitation populaires Android et IOS. La première version sera disponible seulement sur Android, à partir de la version KitKat 4.4 (96 % des terminaux supportés).

L'application étant développée en React-Native, qui supporte les deux plates-formes, le développement sera orienté de façon à pouvoir facilement effectuer le passage sur IOS.

A terme, l'application sera mise à disposition sur les stores, et un site dédié de support et de promotion sera mis en place.

La première version sera disponible en français mais une version en anglais sera rapidement proposée.

e. ÉTUDE DE L'EXISTANT

S'il existe des applications sur le thème de la « survie », **Safe+** se différencie sur plusieurs plans.

Les applications existantes sont anglophones, **Safe+** sera la première application francophone sur le sujet et répondra à une réelle demande, exprimée par les utilisateurs.

Les application existantes sont plus de l'ordre du divertissement , et les contenus sensationnalistes sont souvent peu utiles. **Safe+** Veut proposer une alternative sérieuse et fonctionnelle.

3. CHARTE GRAPHIQUE

DISPOSITIONS GÉNÉRALES

La charte graphique viendra appuyer l'identité que veut se forger **Safe+**. En essayant de s'éloigner des clichés, l'application adoptera un style minimaliste et essaiera de favoriser l'expérience utilisateur en travaillant l'UX¹¹ et l'UI¹².

En effet l'application est prévue pour être utilisée en situation potentiellement difficile, en extérieur, dans l'obscurité. La facilité de lecture et l'ergonomie seront prioritaires.

UTILISATION DE PICTOGRAMMES

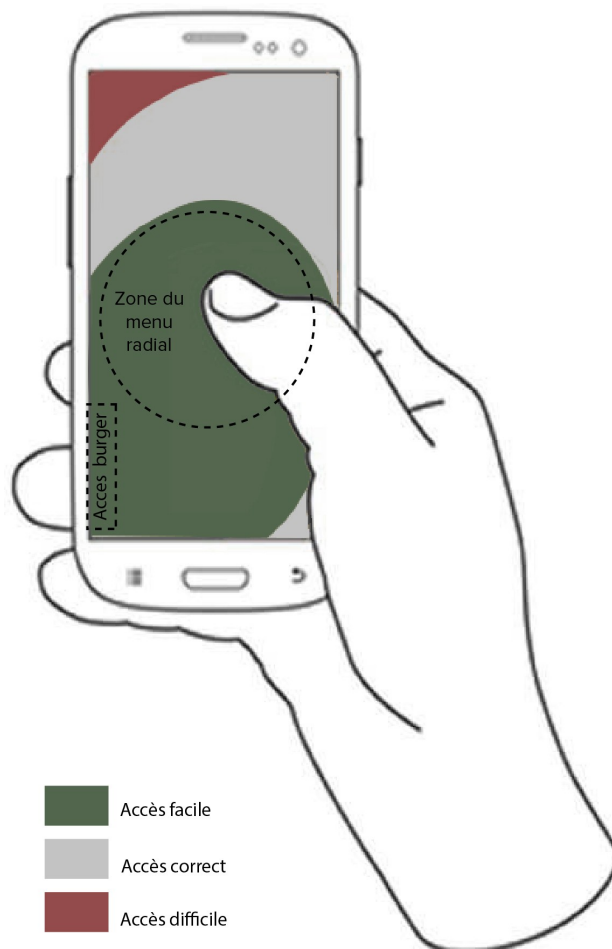
Des pictogrammes seront largement utilisés pour faciliter la lecture, la navigation et se repérer facilement.

UNE ARBORESCENCE LÉGÈRE

Afin de rendre la navigation facile, la profondeur d'arborescence n'excédera pas deux clics pour atteindre un contenu ou un outil.

ERGONOMIE

La disposition des éléments sera réfléchi pour offrir une ergonomie optimale, et favoriser l'utilisation à une main (one-handed UI).



Exemple de travail sur une ergonomie axée "one-handed"

a. LOGOTYPE

Le logo de **Safe+** est sobre. Le gris de la typographie véhicule la tempérance, le calme, le sérieux, et l'autonomie.

Le rouge du **+** est le symbole de la vie, de l'énergie et du triomphe. Il évoque aussi les notions de danger et de secourisme.

Le symbole **+** (plus) est un des symboles humains des plus antiques. Il est lié à la vie. Il évoque également le secourisme (croix rouge), la science (signe mathématique), et la plus-value. Il porte aussi l'idée d'une application «couteau suisse» .

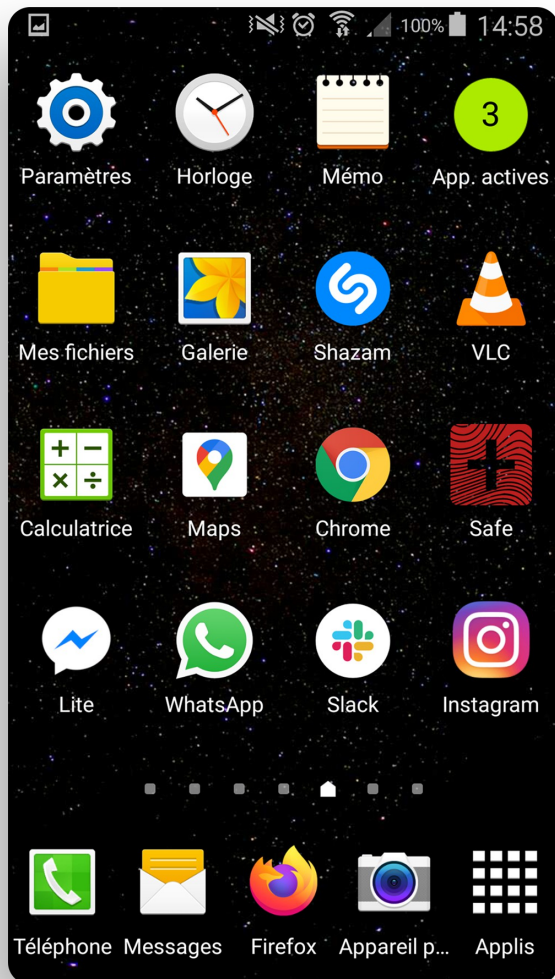
Safe+
Offline Survival App



b. ICÔNE DE L'APPLICATION

L'icône est le point d'entrée de l'application. De taille réduite, son design doit être simple et lisible facilement.

La couleur rouge permettra de retrouver l'application rapidement. Le fond est transparent. L'icône sera fournie dans toutes les résolutions demandées par les OS (512, 192, 144, 96, 72 et 48 px)



c. COULEURS

MODE « JOUR »

En Mode jour les couleurs suivent le logo. Le fond est un blanc cassé pour ne pas agresser les yeux.

Les pictogrammes, typographies et éléments vectoriels suivent des nuances de gris plus ou moins soutenues.

Seul le bouton d'alerte SMS vient trancher avec un rouge repérable facilement.



#fffef7



#303030



#64635f



#be1F1F

MODE « NUIT »

Le mode nuit applique la procédure décrite à la partie II)1)c « Lampe discrète » et à la page « Étude de la lumière émise par **Safe+** ».

Le fond est un rouge pur et sombre, les contenus sont inversement d'un rouge pur plus lumineux.

Les images sont filtrées afin d'annuler toute lumière bleu, et d'atténuer les verts.



#2b0000



#740000

d. TYPOGRAPHIE

TITRAGES

PROXIMA NOVA bold

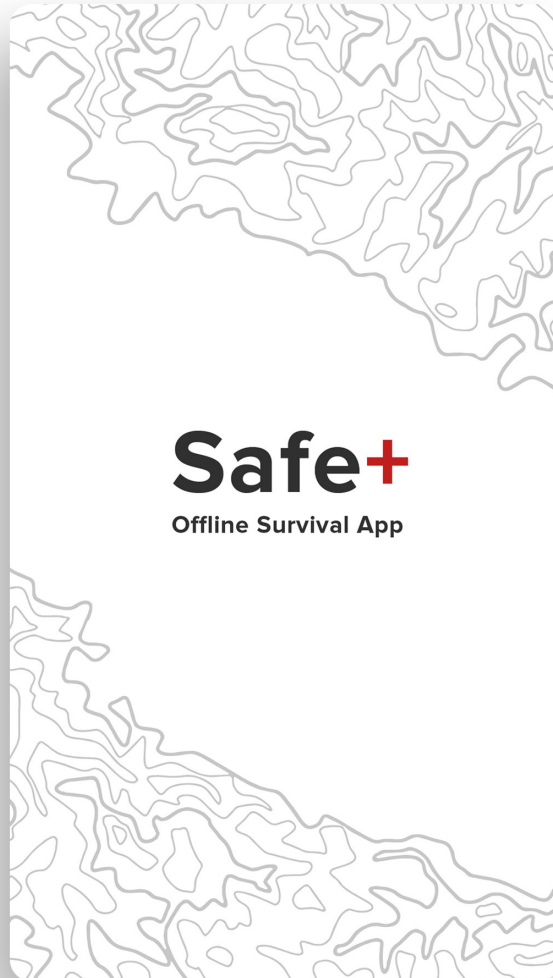
proxima nova bold

CORPS DE TEXTE

CORPORATE S PRO

Corporate S Pro

e. SPLASH SCREEN



Un splash screen a été désigné et sera affiché pendant le temps de chargement de l'application.

Il sera exporté pour toutes les résolutions d'écrans (mdpi, hdpi, xhdpi, xxhdpi, xxxhdpi)

f. ÉLÉMENTS VISUELS

CRÉATION D'UNE LIBRAIRIE D'ICÔNES : « SAFEICONS »

Une bibliothèque d'icônes a été conçue à l'occasion du projet. Elle sera proposée sous forme de librairie React-Native, de police de caractère, et d'images vectorielles. Un package npm sera publié.

Cette bibliothèque sera largement utilisée pour venir simplifier la lecture.

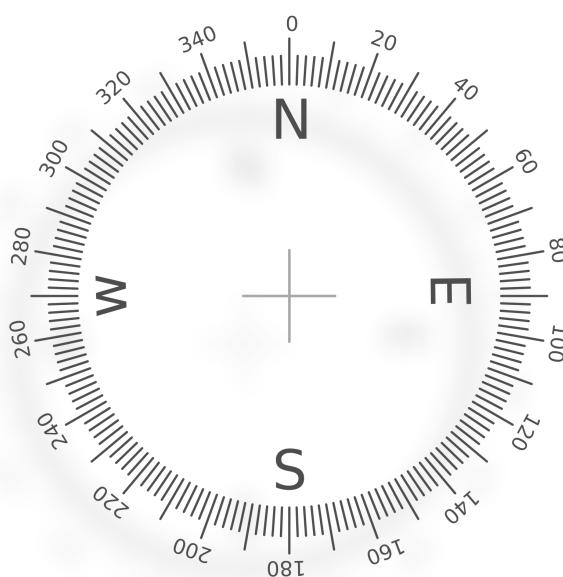
Safelcons V 1.0





ÉLÉMENTS VECTORIELS D'HABILLAGE

Des éléments vectoriels (courbes topographiques) ont été produits pour venir habiller les pages en filigrane. Une boussole a été créée pour la page dédiée.



CRÉATION DE BOUTONS STYLISÉS ET ANIMÉS

Une librairie de deux boutons animés « Switchy » et « Pushy » a été designée.

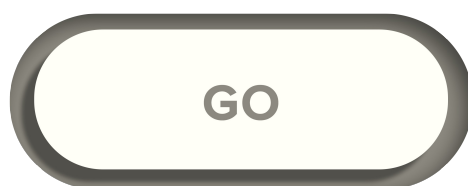
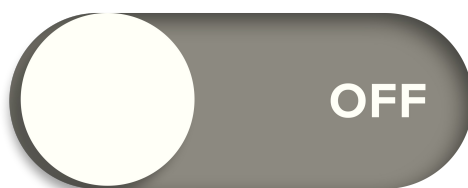
« Switchy » est un bouton switch persistant, il se synchronise automatiquement avec le booléen avec lequel il est lié et est capable de le modifier. « Pushy » est un bouton poussoir ouvert non persistant (NO) qui permet de déclencher une action.

Ils sont assez novateurs car ils ont la particularité d'être complètement vectoriels, paramétrables, mais surtout de disposer d'une ombre interne (l'ombre portée n'étant pas opérationnelle jusqu'à présent en React native pour des formes complexes, d'autant moins comme ombre interne).

Ils peuvent recevoir de nombreuses propriétés (taille, couleur du grip, couleur du fond, texte en position ON et OFF, taille et couleur du texte, décalage X et Y de l'ombre, diffusion de l'ombre...)

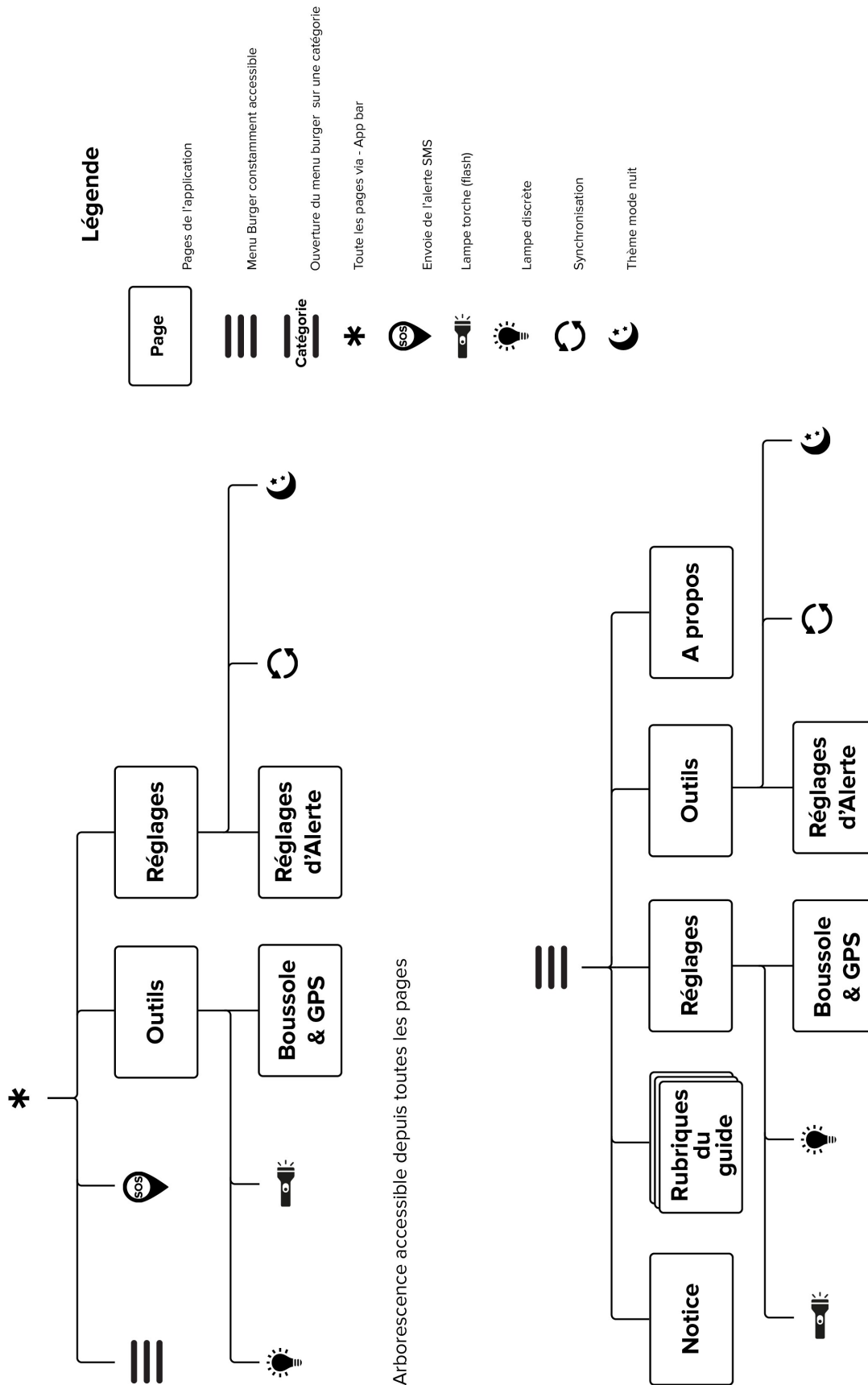
Étant 100% vectoriels, ils sont très légers et la taille ou les réglages ne viennent pas altérer le rendu.

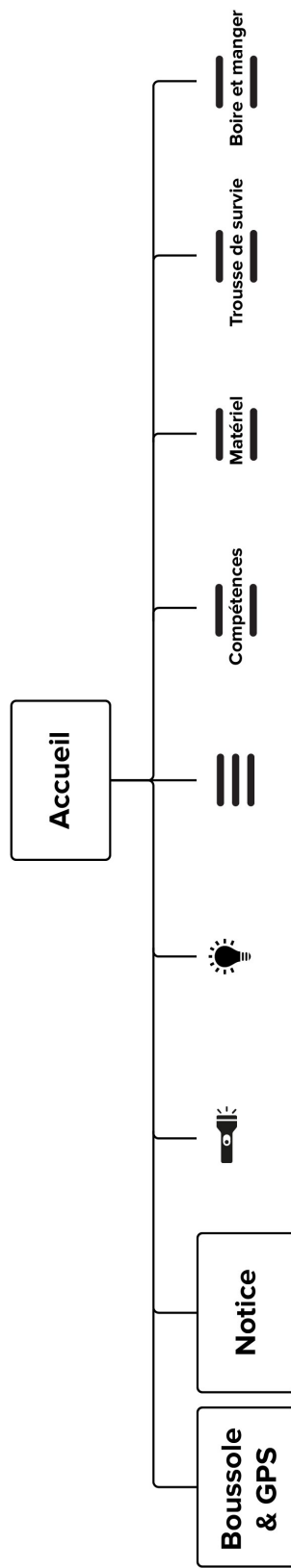
Switchy & Pushy V 1.0



4. SPÉCIFICATIONS FONCTIONNELLES

a. ARBORESCENCE DU SITE



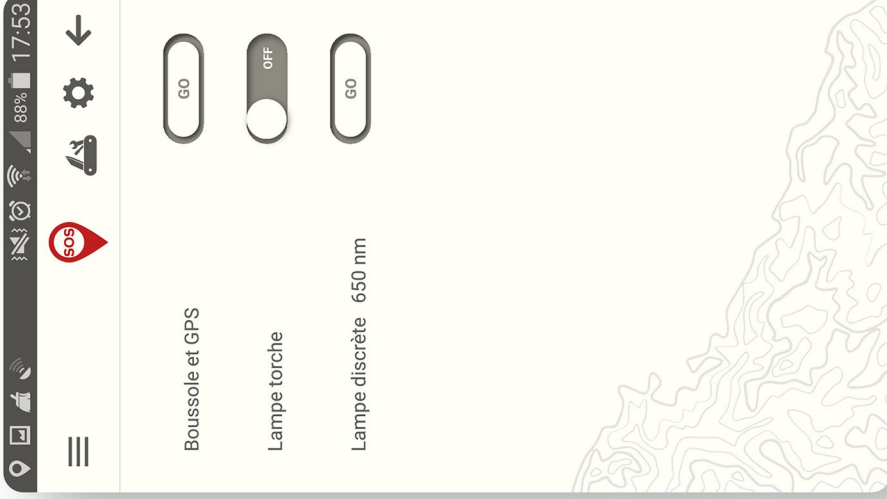


Arborescence supplémentaire accessible depuis la page Home

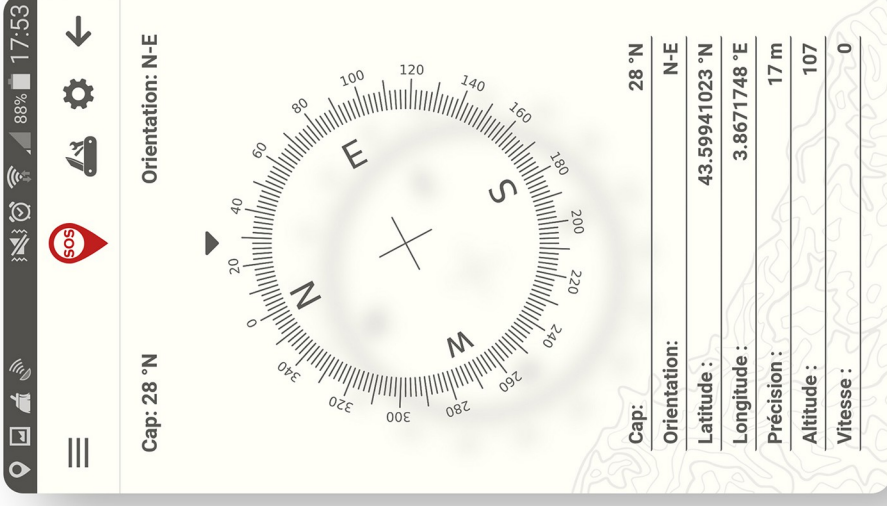
b. CONTENU DES PAGES : MAQUETTES



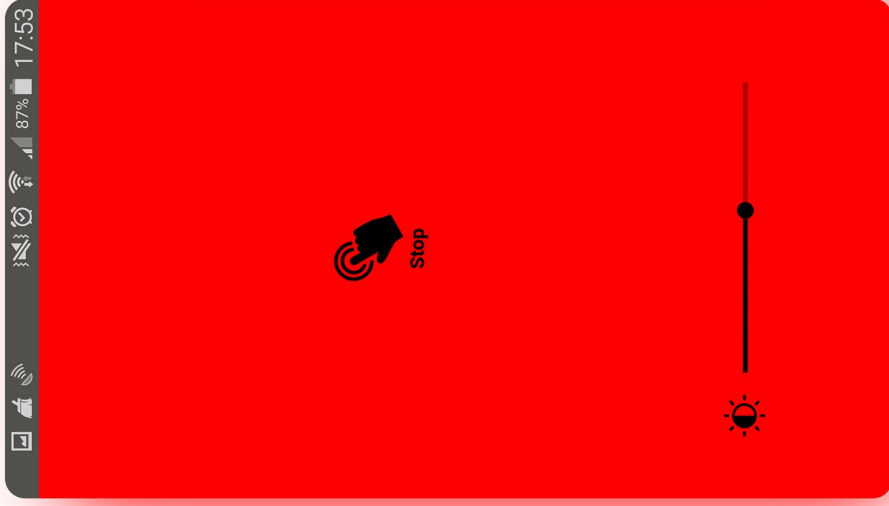
ACCUEIL



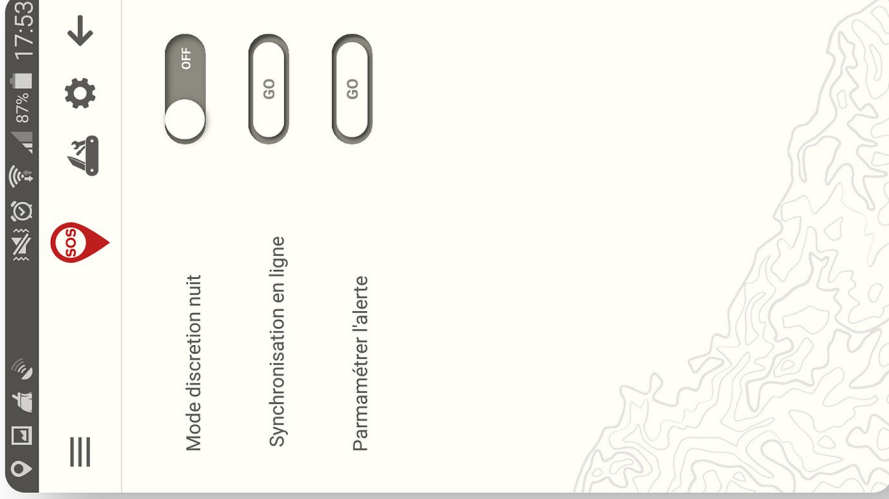
OUTILS



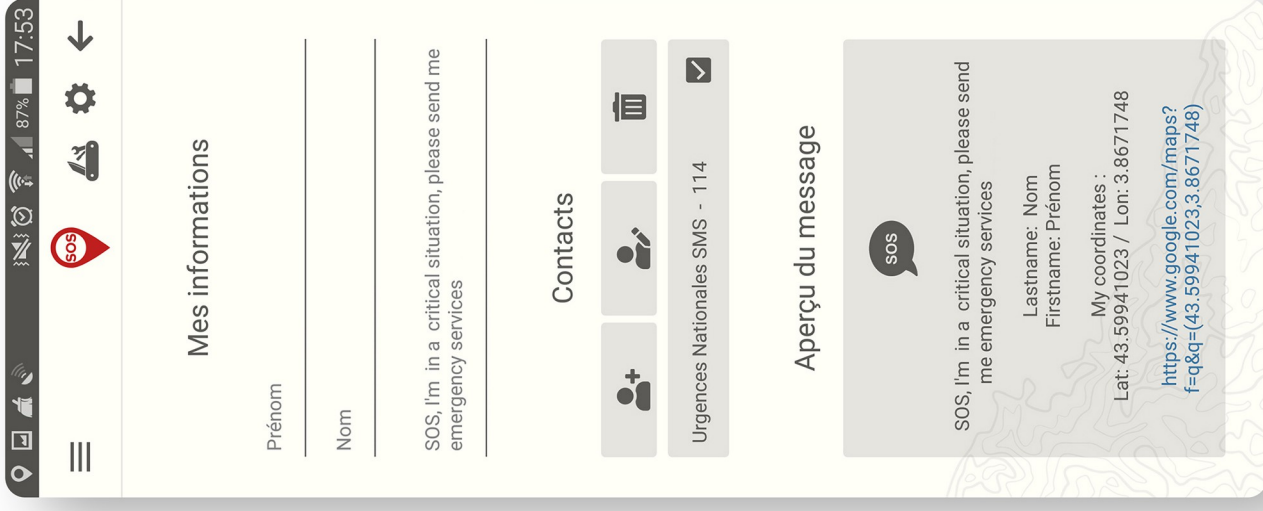
BOUSSOLE & GPS



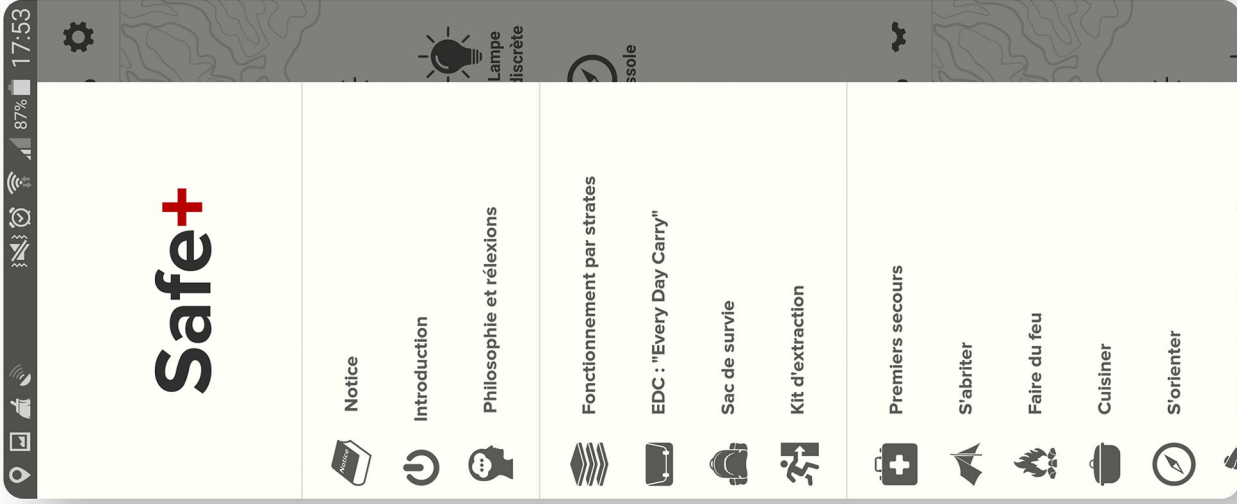
LAMPE DISCRETE



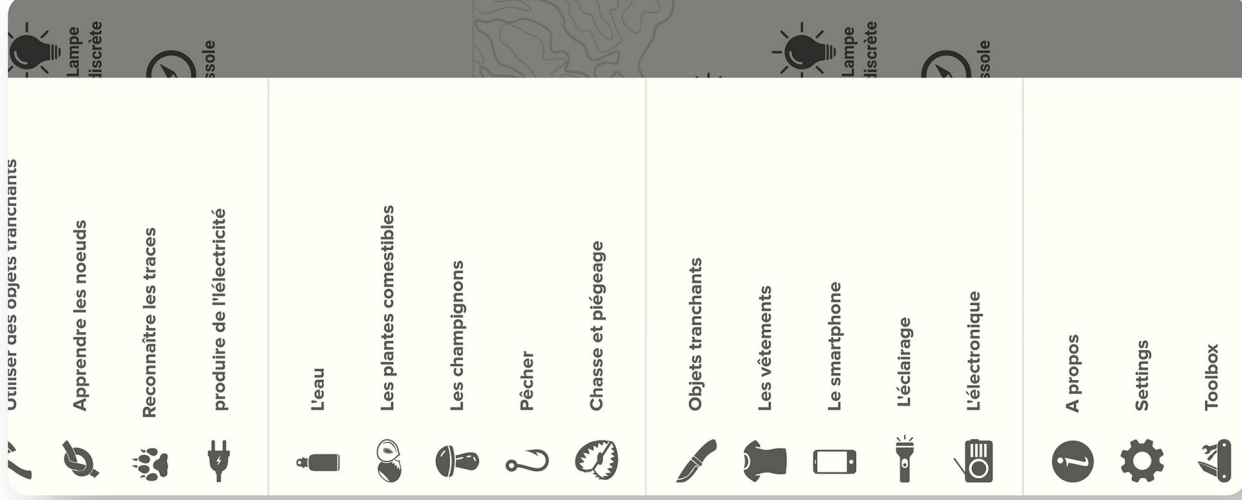
REGLAGES



REGLAGES D'ALERTE



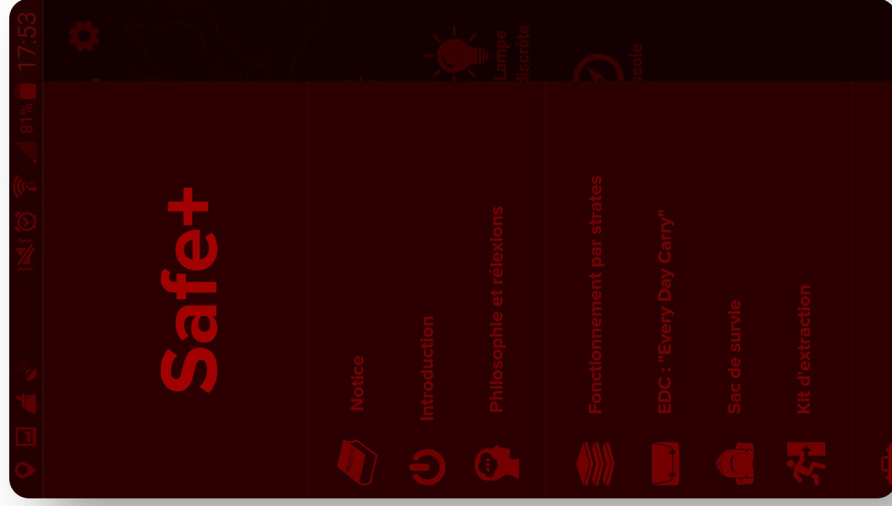
MENU BURGER



MENU BURGER



CONTENU



MENU BURGER



CONTENU



CONTENU

BY NIGHT

c. FONCTIONNALITÉS

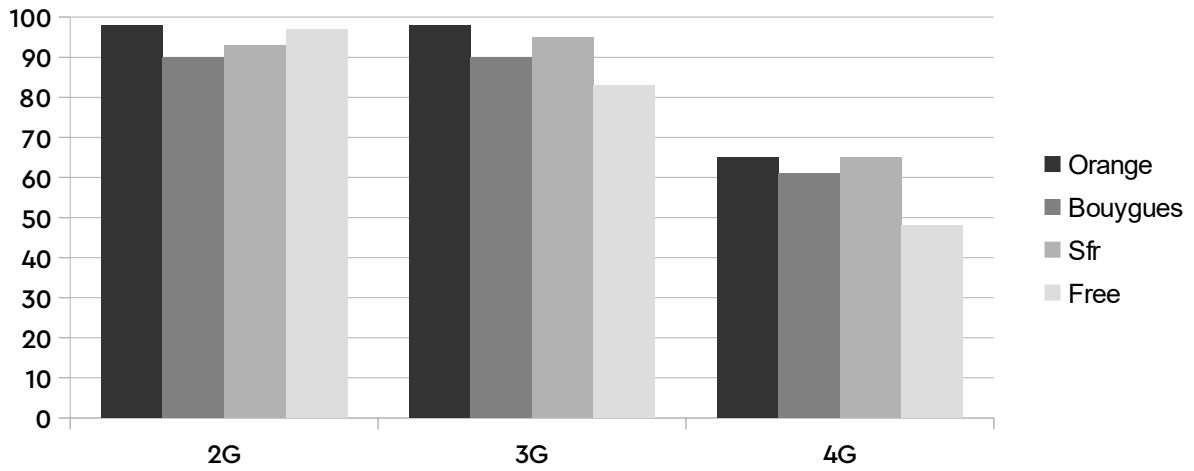
FONCTIONNALITÉS DISPONIBLES EN FONCTION DES PAGES

	Accueil	Outils	Boussole & GPS	Réglages	Réglages d'alerte	Pages du guide
Alerte SMS	✓	✓	✓	✓	✓	✓
Lampe torche	✓	✓				
Lampe discrète	✓	✓				
Synchronisation	✓			✓		
Boussole & Localisation			✓			
Lecture du guide						✓
Changer thème				✓		
Éditer profil d'alerte					✓	
Éditer destinataires					✓	

FONCTIONNALITÉS DISPONIBLES EN FONCTION DE L'ACCÈS AUX RÉSEAUX

Zone	Intérieur	Intérieur	Zone isolée Couverte	Zone Vierge	Zone isolée	Extérieur	Zone Vierge Couverte
Réseaux	WIFI uniquement	3G, 4G uniquement	GSM uniquement	GNSS uniquement	GSM GNSS	3G, 4G GNSS	AUCUN
Probabilité	Très rare	rare	rare	-5% non assuré*	95% assuré*	92% assuré*	exceptionnel
Alerte SMS précise (3-40m)					✓	✓	
Alerte SMS approx (12-50m)		✓	✓		✓	✓	
Lampe torche	✓	✓	✓	✓	✓	✓	✓
Lampe discrète	✓	✓	✓	✓	✓	✓	✓
Synchronisation	✓	✓				✓	
Boussole	✓	✓	✓	✓	✓	✓	✓
Localisation précise (3-40m)				✓	✓	✓	
Localisation approx (12-50m)		✓	✓		✓	✓	
Lecture du guide	✓	✓	✓	✓	✓	✓	✓
Changer thème	✓	✓	✓	✓	✓	✓	✓
Éditer profil d'alerte	✓	✓	✓	✓	✓	✓	✓
Éditer destinataires	✓	✓	✓	✓	✓	✓	✓

*Sur le territoire Français



Superficie relative du territoire Français couverte par les différentes technologies mobiles

USE CASE DE L'APPLICATION

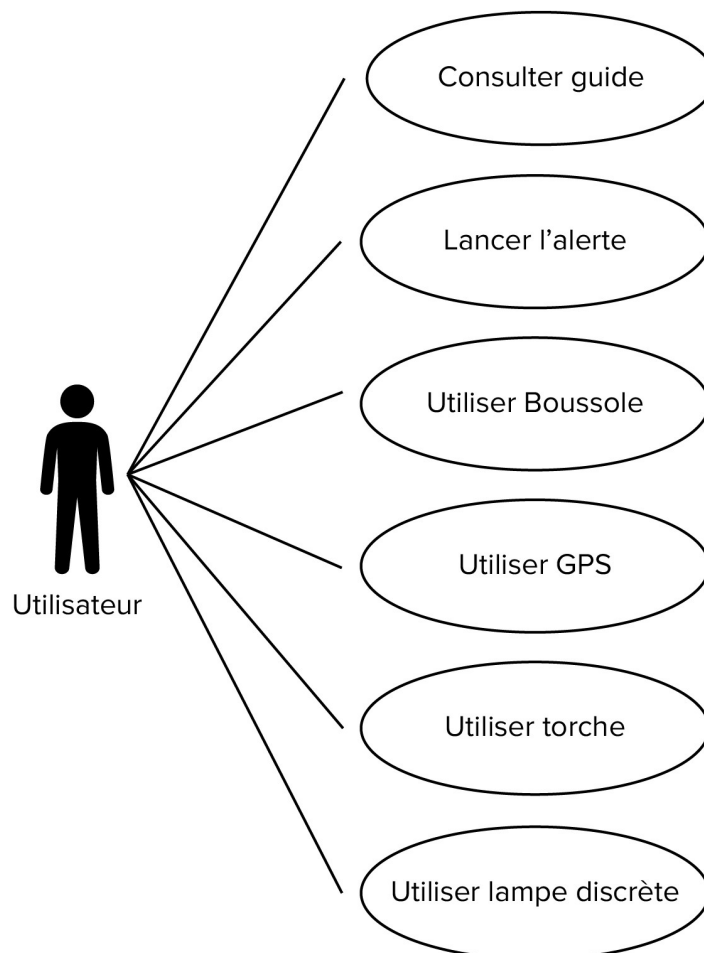


DIAGRAMME D'ACTIVITÉ DE LA FONCTIONNALITÉ DE SYNCHRONISATION

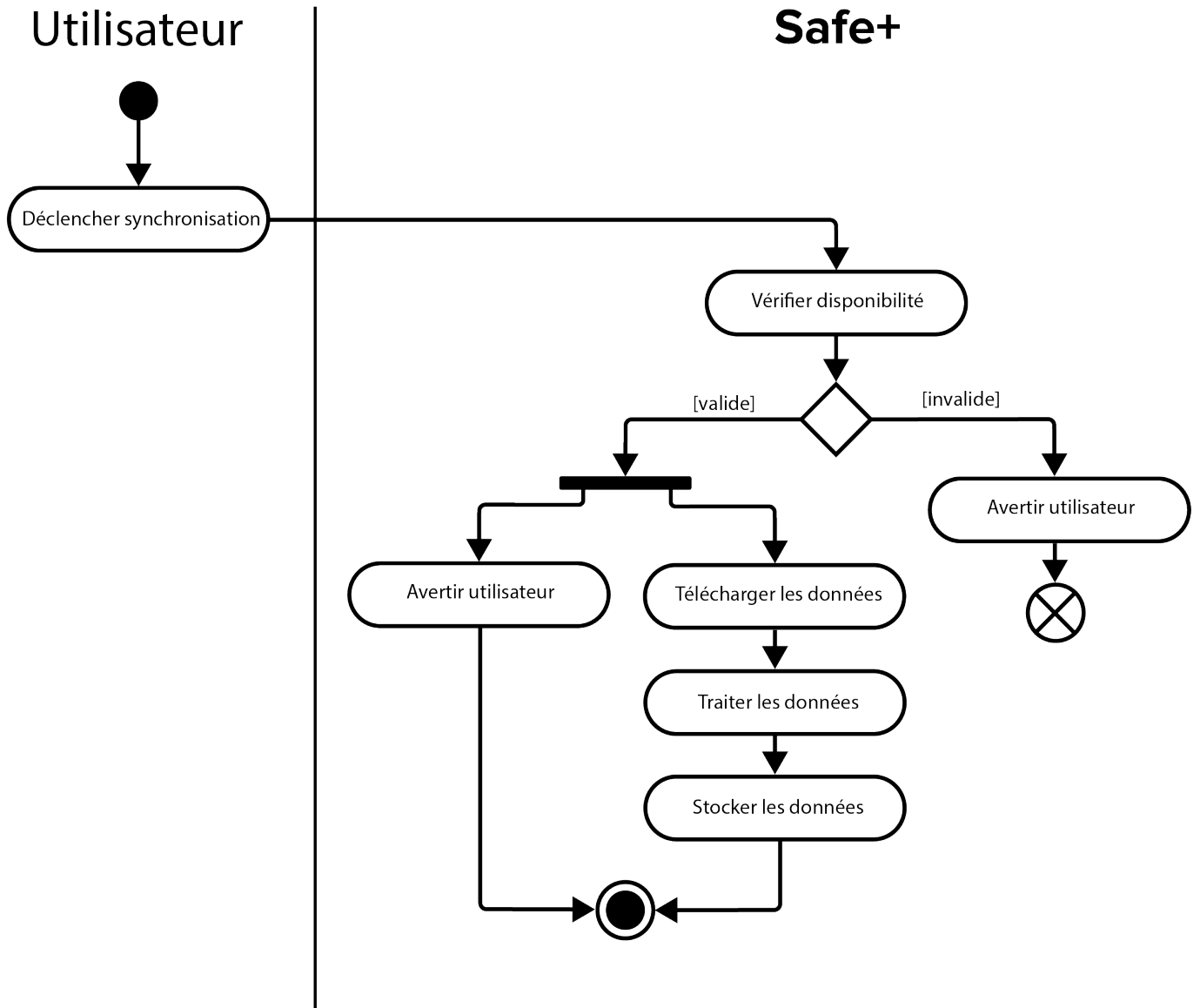


DIAGRAMME DE SÉQUENCE DE LA FONCTIONNalité DE SYNCHRONISATION, CAS VALIDE

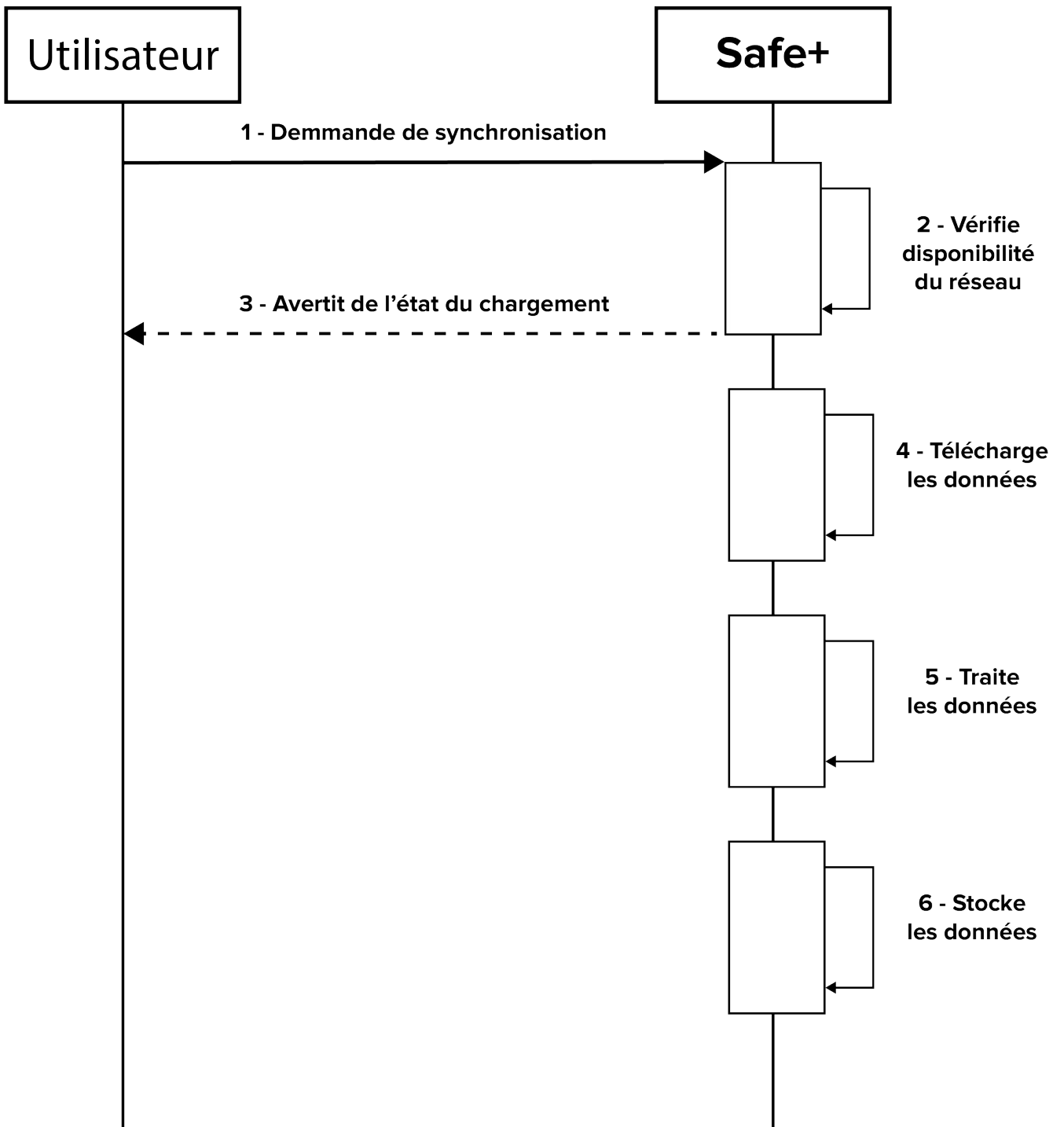
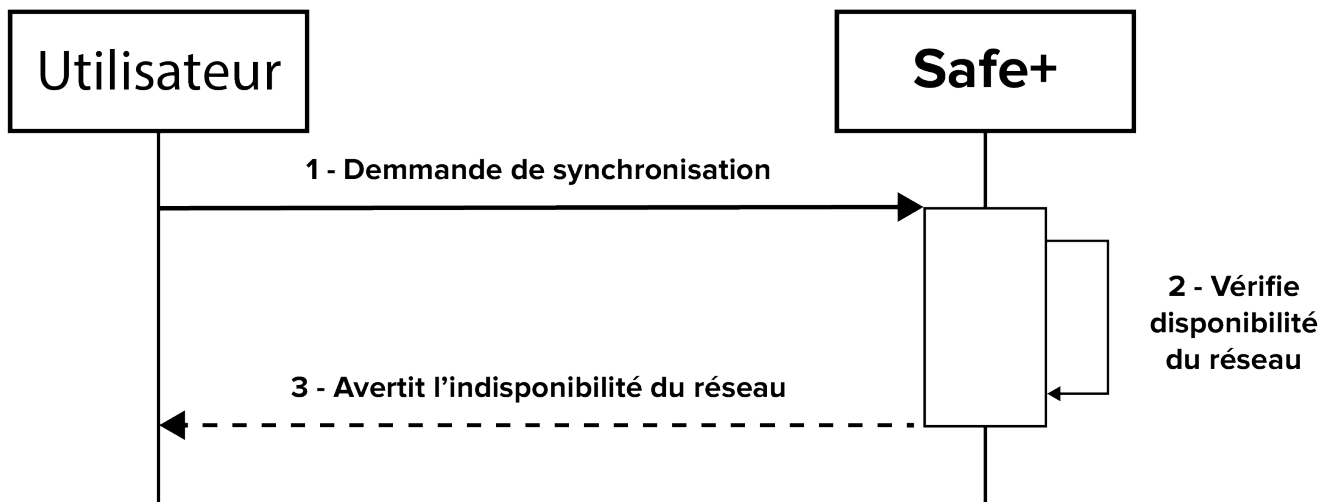


DIAGRAMME DE SÉQUENCE DE LA FONCTIONNalité DE SYNCHRONISATION, CAS INVALIDE



5. SPÉCIFICATIONS TECHNIQUES

a. UNE APPLICATION HORS-LIGNE ET SYNCHRONISABLE

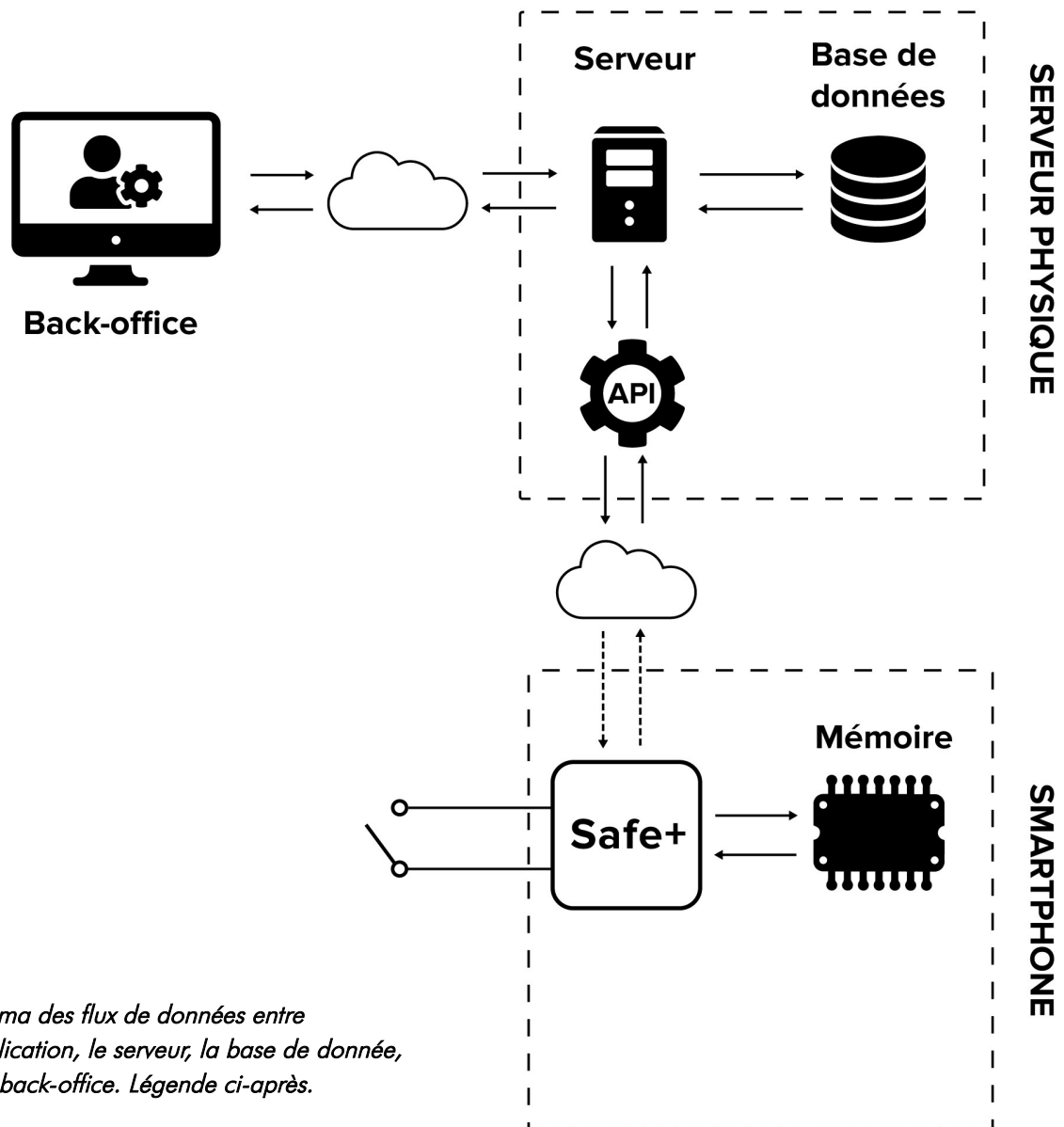


Schéma des flux de données entre l'application, le serveur, la base de donnée, et le back-office. Légende ci-après.



La création du contenu du « Guide de survie » se fait via le back-office WordPress décrit plus bas à la partie 6 « Administration ».

Le tableau de bord permet la gestion et l'organisation du contenu : les articles, les catégories, les images...

Un article créé est rattaché à une « catégorie ». Une catégorie, comme « s'orienter », contient l'ensemble des articles créés sous cette catégorie.

L'éditeur permet de créer facilement un article (ex : « utilisation d'une boussole »). Il permet d'intégrer : du texte, des titres h1,h2,h3,h4, des hyperliens, des listes à puces,des images, des images avec légende. Il a été modifié afin de prendre en compte les limitations imposées par l'application.

La création des contenus ne nécessite donc pas de connaissances techniques, et peut être réalisé par toute personne habilitée disposant des droits. L'application peut être enrichie sans nécessiter de mises à jour de l'utilisateur.



Le serveur web est un serveur « LAMP » (Linux, Apache, MySQL, PHP). C'est une pile logicielle comprenant le système d'exploitation Linux, un serveur HTTP Apache, le système de gestion de bases de données MySQL et le langage de programmation interprété PHP. Porté par le serveur virtuel OVH de Patte Blanche, il générera les pages du back-office, fera le lien avec la base de donnée, et proposera une API pour les terminaux des utilisateurs.



Le serveur HTTP sera connecté au système de gestion de base de données MySQL qui abritera le contenu créé. Voir partie c « schéma relationnel de la base de données de l'API».



Une API REST (REpresentational State Transfer) permet d'échanger via le protocole HTTP avec un serveur en utilisant le formatage JSON (JavaScript Object Notation). Elle fournit des « endpoints » spécifiques matérialisés par des urls sur lesquelles on peut effectuer des requêtes. Ces points d'entrés peuvent renvoyer des données en JSON, issues des données contenues en base de donnée, modifiées ou sans subir de traitement.

L'API REST de WordPress permet ainsi d'obtenir le contenu du site en répondant à des requêtes (Ensemble des articles, catégories, images..) de clients extérieurs.



Lors du premier accès ou à la demande de l'utilisateur, **Safe+** peut synchroniser son contenu local avec le serveur via l'API. Elle effectue une série de requêtes afin de récupérer les nouvelles images et nouveaux articles. Les images sont stockées sur la mémoire du téléphone, et leur chemin local est créé.

L'API REST renvoie les articles sous forme de contenu HTML, incompatible avec l'application développée en React Native. Un service a été développé pour l'application afin de rendre cela possible. Le contenu HTML est parcouru, cartographié et converti en JSON. Il est ensuite stocké sur la mémoire du téléphone.

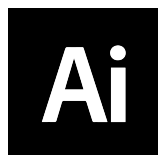
Afin d'afficher le contenu, le JSON est à son tour parcouru pour renvoyer du JSX une extension syntaxique de JavaScript d'autant plus spécifique en React-Native où les balises HTML n'existent pas.

b. TECHNOLOGIES EMPLOYÉES

MAQUETTAGE ET GRAPHISMES



Photoshop

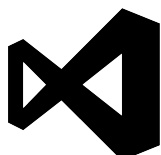


Illustrator

Photoshop et **Illustrator** sont des logiciels de la gamme d'Adobe.

Illustrator offre des outils de dessin vectoriel puissants. Photoshop est un logiciel de retouche, de traitement et de dessin assisté par ordinateur.

DÉVELOPPEMENT



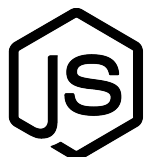
Visual Studio



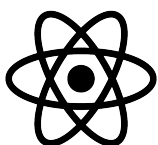
WordPress

Visual Studio Code est un éditeur de code extensible développé par Microsoft.

WordPress est un système de gestion de contenu (CMS) gratuit, libre et open-source.



Node



React-Native

Node.js est un environnement d'exécution qui utilise un moteur JavaScript (un programme logiciel qui interprète et exécute du code en langage JavaScript).

React-Native est un framework d'applications mobiles open source créé par Facebook.

VERSIONING ET GESTION DE PROJET



Git



Trello

Git est un logiciel de gestion de versions décentralisé.

Trello est un outil de gestion de projet en ligne.

SERVEUR ET BACK-OFFICE



Wordpress

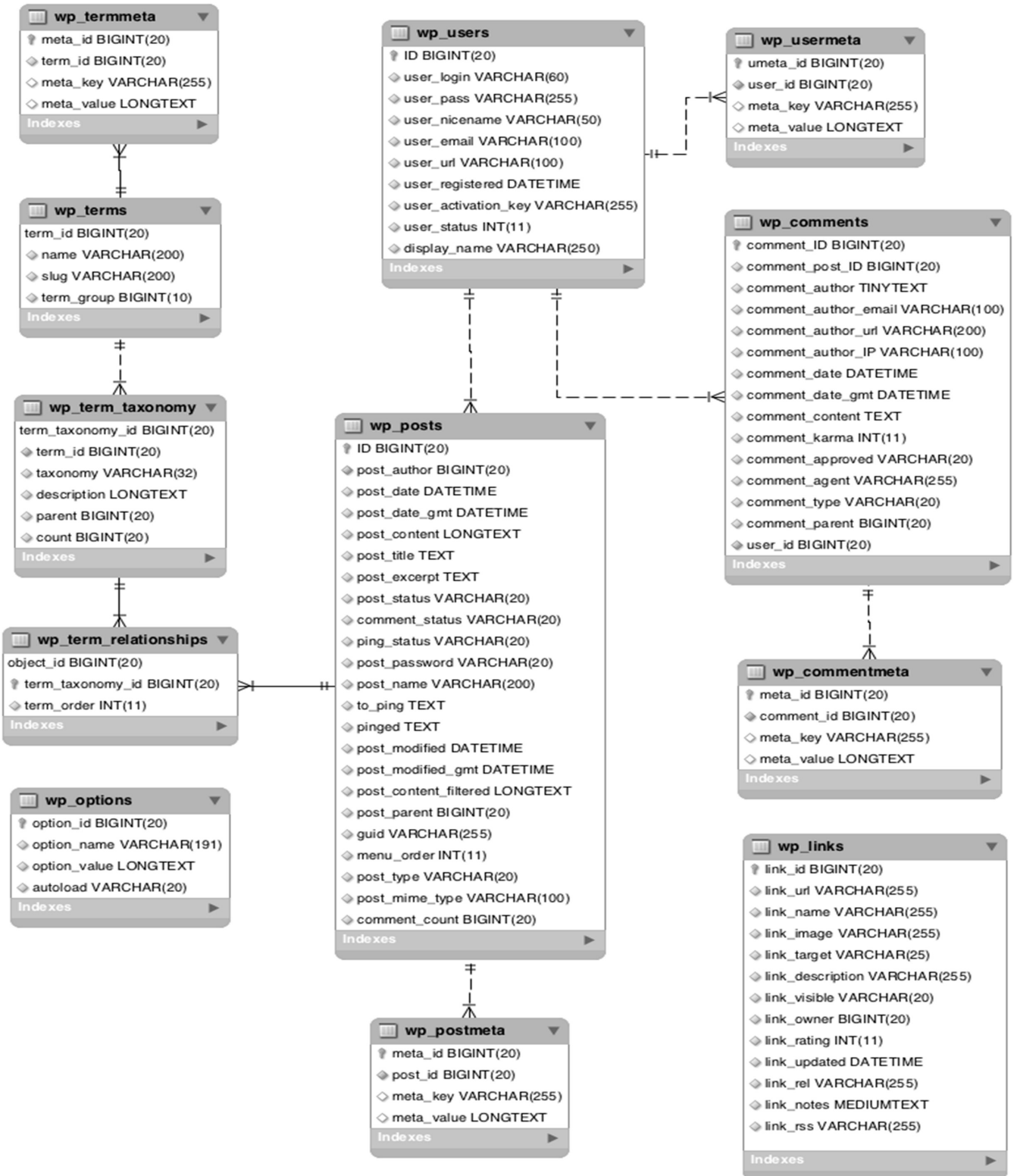


MySQL

L'**API REST Wordpress** est une API intégré au service Wordpress qui permet d'accéder à la base de données d'un site édité avec wordpress.

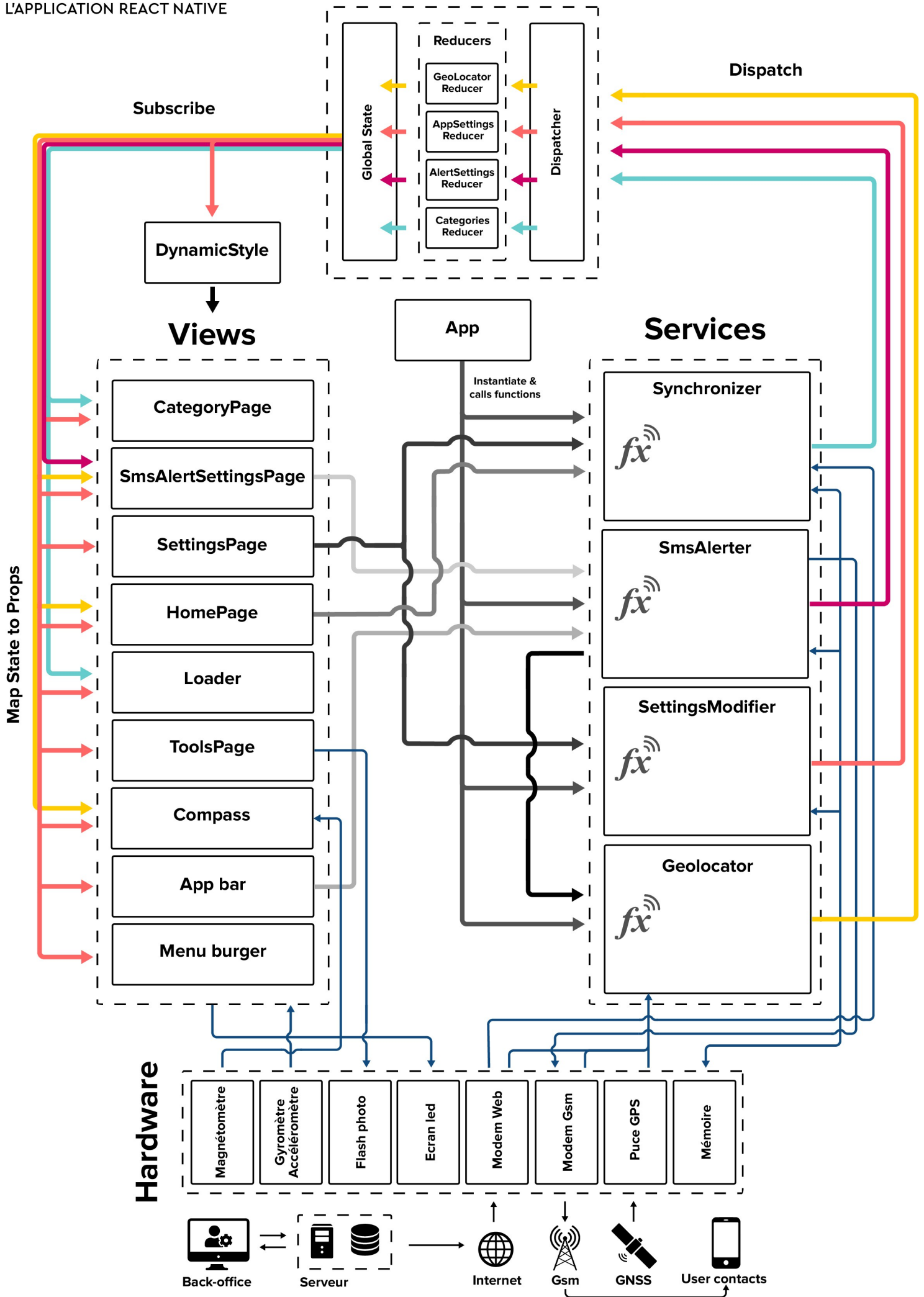
MySQL est un système de gestion de bases de données relationnelles (SGBDR). Il fait partie des logiciels de gestion de base de données les plus utilisés au monde.

c. SCHÉMA "RELATIONNEL" DE LA BASE DE DONNÉES








d. COMPOSANTS ET STRUCTURE DE L'APPLICATION REACT NATIVE

Store Redux

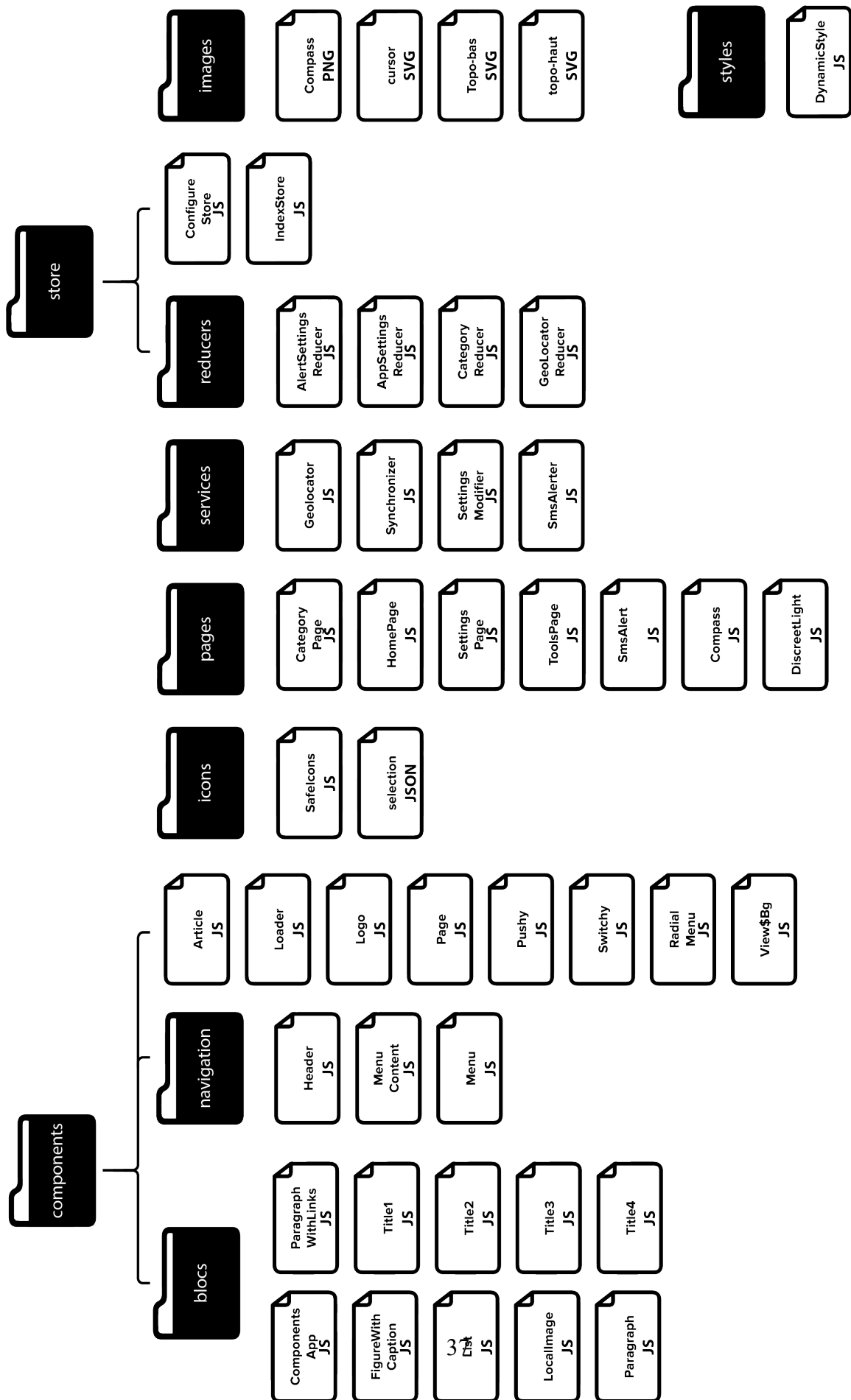


Légende

	Dispatch	«Actions» ou données dispatchées par les services au «Store» Redux
	Subscribe	Réception des données du state sous forme de «Props» par les composants abonnés
		Appels de fonctions des services
		Echanges de données avec les capteurs et le matériel du smartphone. Les échanges sont régis par l'OS.
		Transmission de données entre les dispositifs
	Views	Composants React sous forme de classes renvoyant des affichages
	App	Classe «App» instantiant la navigation contenant les différents composants, et les classes de service
	DynamicStyle	Classe de gestion d'un feuille de style dynamique s'adaptant au theme sélectionné
	Store Redux	Redux est une bibliothèque de gestion d'état pour applications. Il propose un Store gerant un état global de l'app
	Reducer	Fonction actualisant un state suite à la réception d'actions nouvelles

	Services	Classes de services, connectées au Store, et au matériel
	Synchronizer	Service permettant la synchronisation du contenu et des images avec l'API, leur stockage persistant sur le mobile, et leur mise à disposition.
	SmsAlerter	Service gerant les paramètres de l'alerte SMS, la personnalisation, les contacts (CRUD), et l'envoi via le réseau gsm
	SettingsModifier	Service gerant les paramètres généraux de l'application comme le thème nocturne
	Geolocator	Service de géocalisation actualisant la position en cas de déplacement. Il utilise la puce GPS et le fournisseur de position réseau
	Magnétomètre	Capteur mesurant l'intensité et la direction du champ magnétique. Il fourni le nord magnétique à la boussole.
	Gyromètre Accéléromètre	Ces capteurs mesurent respectivement l'accélération angulaire et linéaire. A eux deux ils permettent de déterminer l'orientation du mobile.

e. ARCHITECTURE SIMPLIFIÉE DU SYSTÈME DE FICHIERS

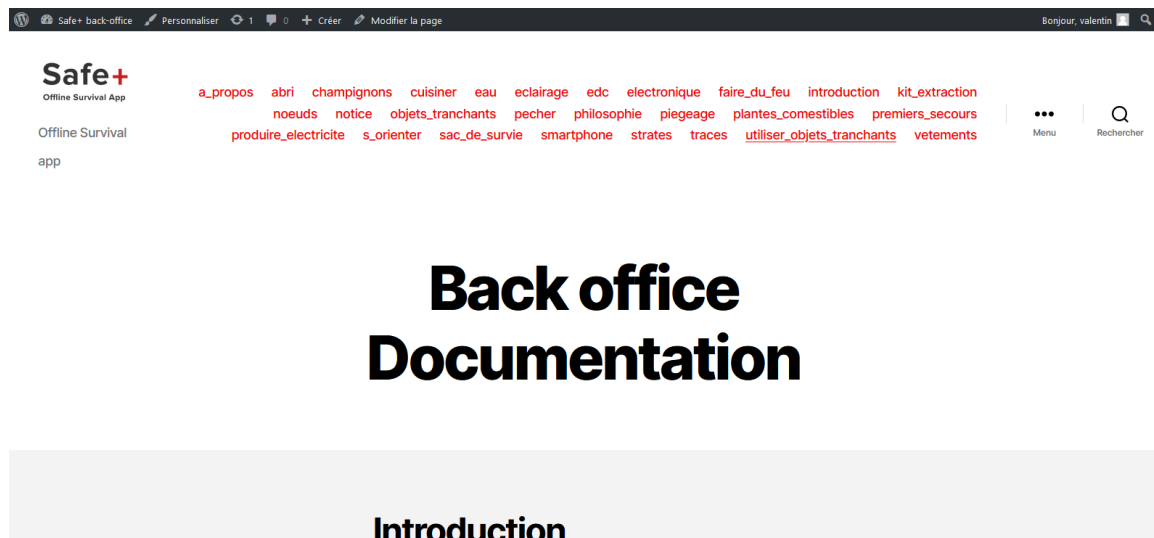


6. ADMINISTRATION

a. OUTIL D'ADMINISTRATION

L'administration de l'application se fera via le site Wordpress dédié dont le fonctionnement est expliqué à la partie ci-dessus 5)a. Le CMS wordpress enrichi de son API se présente comme un solution légère et facile d'utilisation de back-office et d'API.

La documentation du back-office et de l'application sera proposée sur la page d'accueil du back-office. Elle présentera le fonctionnement de l'API, les différents points d'entrés utilisés, le fonctionnement et les limitations de l'éditeur personnalisé.

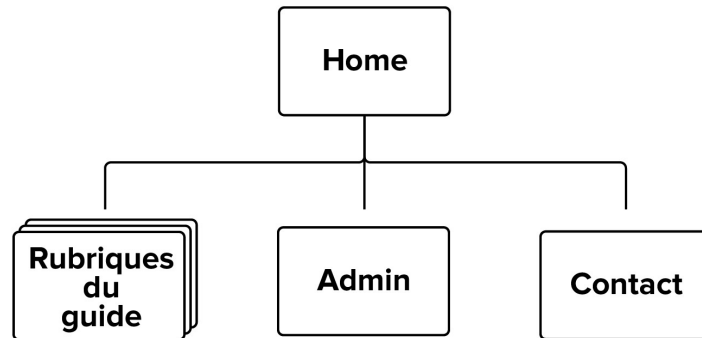


Ecran d'accueil du back-office proposant la documentation relative à son utilisation et à l'application

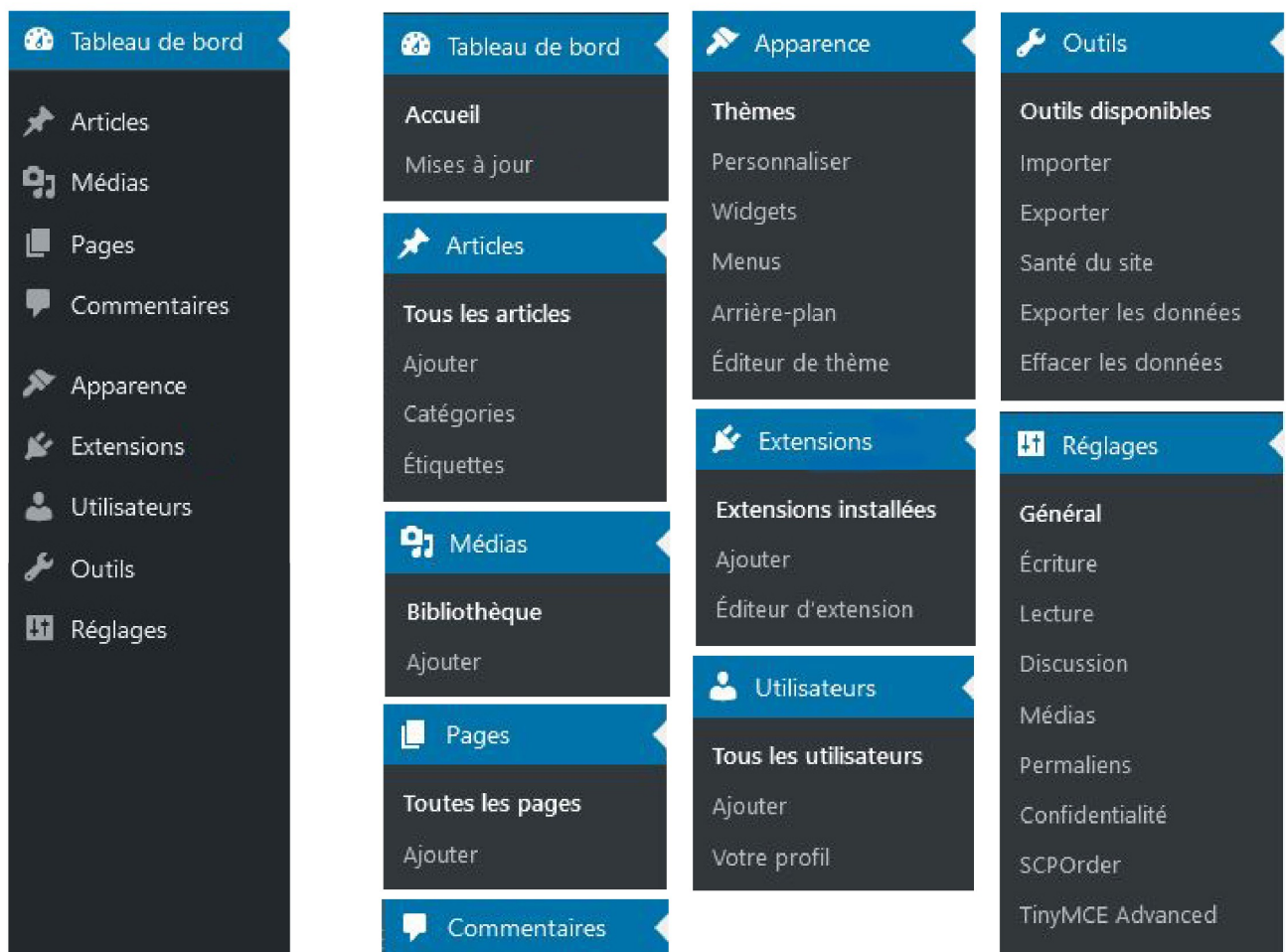


Ecran montrant l'édition d'un article à l'aide de l'éditeur

b. ARBORESCENCE DU BACK-OFFICE



Arborescence du back-office



Les différentes rubriques de la page d'administration du site WordPress

7. SECURITE

CONFIDENTIALITÉ DE L'APPLICATION

La nature de **Safe+** exige des conditions de confidentialité exceptionnelles. Prévue pour être utilisable dans des contextes d'instabilité politique, **Safe+** devra offrir à ses utilisateurs un anonymat absolu.

Suite à une enquête sur un public ciblé, des mesures strictes ont été prises :

- L'application ne proposera pas de création de compte utilisateur.
- Elle n'utilisera pas de services tiers .
- La liste de contact sera propre à l'application et ne sera pas partagée.
- Aucun envoi de données ne sera fait outre les requêtes faites à l'API de **Safe+** pour demander le nouveau contenu.

LE BACK OFFICE

Le risque majeur pesant sur le système de **Safe+** est le piratage des comptes administrateurs dans le but de modifier ou supprimer le contenu.

Différentes actions ont été mises en place afin de sécuriser le back-office :

Renommage du compte « ADMIN ».

Par défaut sur wordpress le compte administrateur est nommé « Admin ». Cet identifiant peut facilement être utilisé par les pirates. Des identifiants personnalisés ont donc été créés. Seuls les personnes sous ce rôle d'administrateur sont en mesure d'éditer le contenu.

Protection du mot de passe en base de donnée

Le stockage des mots de passe en base de donnée est protégé via un hashage MD5¹³ Outre l'insuffisance de la taille de l'empreinte (128bits) pour garantir une sécurité à long terme, le MD5 a été l'objet de diverses attaques qui ont prouvé sa faiblesse d'un point de vue cryptographique. Pour éviter cette fragilité, un Must-Use Plugin a été installé afin de remplacer le MD5 en chiffrement Bcrypt¹⁴ plus sécurisant.

Protocole sécurisé

Toutes les connexions entre l'application et l'API de WordPress se fait via le protocole HTTPS¹⁵ qui est un protocole sécurisé par chiffrement.

8. EVOLUTION ET SUIVI DE L'APPLICATION

a. DÉPLOIEMENT

L'application sera déployée sur les stores dédiés, le Play Store Android dans un premier temps et l'Apple Store dans un second temps. Un site public chargé de faire la promotion sera créé dans un futur proche et renverra vers les stores respectifs.

b. FORMATION

Une réunion sera organisée afin d'effectuer la passation et la formation au back-office.

Seront proposés :

- Une présentation de l'application V 1.0
- Des propositions d'évolutions futures
- Une présentation d'un plan de suivi de l'application
- La formation à l'utilisation du back-office

c. ÉVOLUTION

L'application qui sera fournie constituera la version Android 1.0. Des propositions d'évolution ont déjà été pensées.

- Une mise à niveau vers IOS.
- Le choix de la langue Français / Anglais
- Menu burger en Accordéon par super catégories.
- Intégration d'un module de recherche par mot clef / étiquette
- Intégration d'un dossier de fichiers ressources
- Choix entre une synchronisation manuelle et/ou automatique
- Paramétrage d'une mémoire maximale allouée à l'application
- Ajout d'une prévision météo journalière effectuée sur la base du baromètre ou de donnée d'APIs.
- Module de transcription en morse bi-directionnel.
- Optimisation du contrôleur (algorithme de traduction côté serveur)
- ...

9. ORGANISATION DU PROJET

a. PLANNING PRÉVISIONNEL

		Décembre															Janvier															Février																																					
		9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Mise en place	Activité	Élaboration du projet																																																																			
	Définition et cadrage du projet	Identification des contraintes Étude de l'existant Soumission des propositions (réunion) Objectif, cibles, fonctionnalités																																																																			
	Définition du projet	Identité de l'application, recherche du nom																																																																			
	Identité et naming	Création du logo Choix des couleurs, Typographies Création des éléments d'habillage et illustrations Dessin des pictogrammes, et création de police Design des boutons poussoirs et switches Design des screens et maquetage Design des splash screens et icône Android																																																																			
Élaboration du cahier des charges	Spécifications fonctionnelles	Absence de site Contenu des pages Fonctionnalités détaillées																																																																			
	Spécifications techniques	Sélection des technologies pour le backend Sélection des technologies pour l'application Sélection des technologies pour l'application Organisation et structure du code de l'application																																																																			
	Administration	Spécifications fonctionnelles du back-office																																																																			
	Evolution et suivi	Élaboration d'un plan de suivi de l'application																																																																			
	Mise en place du back-office	Installation des fichiers Wordpress sur le serveur Installation de la base de donnée MySQL Installation des extensions (édition, spans) Initialisation du contenu et du menu																																																																			
Mise en place de l'API		Prise en main et mise en place de l'API WP Choix techniques relative à l'utilisation de l'API																																																																			
Mise en place du Back-office et de l'API	Documentation	Production de la documentation																																																																			
	Mise en place de l'application	Installation, imports et composants vitaux (Redux...), autorisations Mise en place de la structure, navigation, menus...																																																																			
	Création des services	Service de synchronisation avec l'API : « Synchronizer » Service de géolocalisation : « Geolocator » Service d'envoi d'alerte SMS « SmsAlert » Service de gestion des réglages « SettingsModifier »																																																																			
	Création des outils	Outil boussole et géolocalisation « Compass » Outil lampe torche « TorchLight » Outil lampe discrète « DiscreetLight » Outil envoi d'alerte SMS « SosAlert »																																																																			
	Pages du guide de survie	Composants de bases (texte, image, hyperliens, listes...) Composants intermédiaires (articles...)																																																																			
	Boîtes de dialogues	Page dynamique de rendu «CategoryPage » Page «Accueil» avec menu de raccourcis Page « toolbox » proposant les outils Page «Settings» proposant les réglages (Mode Nuit...) Page « Alert Settings » de paramétrage de l'alerte Fenêtre de chargement « Loader » « Header » ou « App bar »																																																																			
	Autres screens	Librairie de deux boutons : « Switchy » et « Pushy » Librairie d'icônes « SafeIcons »																																																																			
	Création de modules	Création d'une feuille de style dynamique implémentation des éléments d'habillages Implémentation des polices externes Implémentation des « splash screens » et icônes Android Tests en situation réelle (réseau, localisation...)																																																																			
	Design	Tests et débogage																																																																			
	Finalisation	Configuration (noms, renseignements) Compilation « release », débogage de compilation																																																																			
	Préparation livraison	Préparation de la présentation																																																																			
Livraison	Présentation finale	Présentation de l'application V1.0 Propositions d'évolutions Présentation d'un plan de suivi de l'application (ios, contenu, diffusion)																																																																			
	Formation	Formation à l'utilisation du back-office																																																																			
	Passation	Transmission des fichiers																																																																			

III. PRÉSENTATION DOCUMENTÉE DE QUELQUES FONCTIONNALITÉS

1. EXTRAITS SÉLECTIONNÉS DE CODE COMMENTÉS

a. EXTRAITS DU SERVICE « SYNCHRONIZER »

PARAMÉTRAGE

L'extrait commenté ci-dessous présente l'entête du service « Synchronizer ». Il comporte les imports nécessaires à son fonctionnement ainsi que les constantes.

```
import RNFetchBlob from 'rn-fetch-blob';
//HttpRequests, téléchargement et mangement de fichiers
import AsyncStorage from '@react-native-community/async-storage'; //stockage local permanent
import Store from '../Store/configureStore';//Import du Store redux pour les dispatchs
import NetInfo from '@react-native-community/netinfo';//Analyse de l'état du réseau

//Url globale et endpoints de l'Api
//Url générale de l'API
const API_URL = 'https://bushcraft.patte-blanche.fr/wp-json/wp/v2/';
//Obtention des catégories
const GET_CATEGORIES_ENDPOINT = 'categories?per_page=100';
//Obtention des articles
const GET_POSTS_ENDPOINT = 'posts?per_page=100';
//Obtention des urls des media
const GET_MEDIA_ENDPOINT = 'media?per_page=100';

const {config, fs} = RNFetchBlob;
//chemin du stockage des images
const IMAGES_PATH = fs.dirs.DocumentDir + '/survival/';
```

```
//Modèle de l'objet servant à stocker les articles par catégorie
const CATEGORIES = {
  notice: [],
  introduction: [],
  s_orienter: [],
  abri: [],
  champignons: [],
  eau: [],
  premiers_secours: [],
  eclaireage: [],
  edc: [],
  electronique: [],
  faire_du_feu: [],
  cuisiner: [],
  kit_extraction: [],
  noeuds: [],
  traces: [],
  objets_tranchants: [],
  pecher: [],
  philosophie: [],
  piegeage: [],
  plantes_comestibles: [],
  produire_electricite: [],
  sac_de_survie: [],
  smartphone: [],
  strates: [],
  utiliser_objets_tranchants: [],
  vetements: [],
  a_propos: [],
  non_classes: [],
};
```

ORCHESTRATION DE LA SYNCHRONISATION

La fonction ci-dessous est déclenchée par l'utilisateur ou est appelée lors de la première utilisation. Elle orchestre les fonctions permettant de synchroniser le contenu stocké localement avec le contenu hébergé sur le serveur. Les fonctions appelées sont décrites plus bas.

```
//Synchronisation en ligne forcée: Fetchs sur l'API WordPress, traduction de
//l'HTML en objets locaux, téléchargement des images
hardSynchronize = async () => {
  //On vérifie l'état du réseau
  if ((await this.getNetworkState()) == true) {
    //On prévient les composants que la synchronisation est en cours (dispatch)
    this.setTextIsLoading(true);
    this.setImagesAreLoading(true);
    //On attend le téléchargement des images
    await this.downloadImages();
    //On attend la réception et conversion en objet du contenu de l'API
    cat = await this.fillCategories();
    //On stock l'objet créé
    this.storeObj('categories', cat);
    console.log(cat)
    //On dispatch l'objet'
    this.updateCategories(cat);
    //On prévient les composants que la synchronisation est terminée
    this.setTextIsLoading(false);
  }
  //On averti l'utilisateur si la synchronisation ne peut pas se faire
  else {
    alert('Pas de connexion internet');
  }
};
```


La fonction ci-dessous est appelée au démarrage de l'application. Elle récupère le contenu stocké localement, ou à défaut, sur le serveur.

```
//On check au démarrage la nécessité de synchroniser
softSynchronize = async () => {
  //On prévient les composants que la synchronisation est en cours (dispatch)
  this.setTextIsLoading(true);
  this.setImagesAreLoading(true);
  // AsyncStorage.removeItem('categories')
  //On tente de récupérer le contenu dans le stockage local
  try {
    cat = await AsyncStorage.getItem('categories');
    //Si les données existent on les dispatch
    if (cat !== null) {
      cat = JSON.parse(cat);
      this.updateCategories(cat);
      this.setTextIsLoading(false);
      this.setImagesAreLoading(false);
    } else {
      //Si les données n'existent pas on déclenche la synchronisation
      this.hardSynchronize();
    }
  } catch (error) {
    console.log(error);
  }
};
```

La fonction ci-dessous permet de dispatcher une action Redux mettant à jour l'objet « catégories » dans toute l'application.

```
//Dispatch categories
updateCategories(cat) {
  //On crée l'action redux
  const action = {type: 'UPDATE_CATEGORIES', value: cat};
  //On la dispatch au store
  Store.dispatch(action);
}
```

La fonction ci-dessous empêche la synchronisation en cas d'absence de réseau.

```
// fonction qui vérifie l'état du réseau avant de synchroniser
getNetworkState = async () => {
  let connexionState = false;
  //cette méthode fournie par NetInfo permet de récupérer l'état du réseau
  await NetInfo.fetch().then(state => {
    connexionState = state.isConnected;
  });
  return connexionState;
};
```

Cette fonction fournie par l'«AsyncStorage» permet de stocker des JSON sur la mémoire du terminal.

```
//fonction effectuant le stockage des objets en JSON
storeObj = async (objName, obj) => {
  try {
    await AsyncStorage.setItem(objName, JSON.stringify(obj));
  } catch (error) {
    console.log(error);
  }
};
```

Les articles renvoyés par l'API sont accompagnés de l'id de leur catégorie au lieu du nom. Cette méthode exécute une requête pour obtenir la correspondance.

```
// On fait un requête à l'API pour obtenir le lien entre les IDs et les noms des catégories
linkCategories = async () => {
  const response = await fetch(API_URL + GET_CATEGORIES_ENDPOINT);
  const json = await response.json();
  var CategoriesLinks = {};
  //On crée un objet de correspondance, avec les IDs comme clés, et les noms comme valeurs
  json.forEach(category => {
    CategoriesLinks[category.id] = category.name;
  });
  return CategoriesLinks;
};
```

Cette fonction orchestre différentes fonctions qui permettrons, ensemble, de retourner l'objet abritant tout le contenu du guide.

```
// Fonction mère qui appelle les sous-fonctions:
//retourne un objet hébergeant tout le contenu
fillCategories = async () => {
  // On récupère les posts
  const response = await fetch(API_URL + GET_POSTS_ENDPOINT);
  const json = await response.json();
  // On récupère le lien entre les IDs et noms des catégories
  const links = await this.linkCategories();
  //On récupère le modèle vide de l'objet
  var fetchedCategories = JSON.parse(JSON.stringify(CATEGORIES));
  // On parcourt l'objet constitué des posts renvoyé par l'API
  json.forEach(thisArticle => {
    //Pour chaque post: on récupère l'ID de la catégorie, et on en déduit le nom
    categoryId = thisArticle.categories[0];
    categoryName = links[categoryId];
    //On construit un objet correspondant à un post, l'HTML est converti en objet
    //grâce à notre méthode convertHtmlToObject
    article = {
      title: thisArticle.title.rendered,
      content: this.convertHtmlToObject(thisArticle.content.rendered),
    };
    //On le pousse dans l'objet de la bonne catégorie
    fetchedCategories[categoryName].push(article);
  });
  //on retourne l'objet final
  return fetchedCategories;
};
```

CONVERSION EN JSON

Le principe de l'algorithme de conversion est simple : une « tête de lecture » parcourt l'html balise après balise, et construit au fur et à mesure un objet mémorisant la structure et le contenu. Les extraits ci-dessous montrent le code qui permet de détecter la position et la nature de la balise qui suit.

```
//Fonction qui converti l'HTML en objet pour le stocker et servir de source au jsx
convertHtmlToObject = htmlContent => {
  //Tableau relatif à un article, qui contient l'ensemble des balises et contenus
  var arrayArticle = [];
  //On initie la "tête de lecture"
  var indexBeginToRead = 0;
  //On définit les balises que l'on veut garder (balises ouvrantes,fermantes, index)
  tagsToSearch = [
    ['<div>', '</div>', '-1'],
    ['<p>', '</p>', '-1'],
    ['<h1>', '</h1>', '-1'],
    ['<h2>', '</h2>', '-1'],
    ['<h3>', '</h3>', '-1'],
    ['<h4>', '</h4>', '-1'],
    ['<ul', '</ul>', '-1'],
    ['<figure', '</figure>', '-1'],
    ['<img', '</>', '-1'],
  ];
};
```

```
//On va mapper une à une les balises tant que toutes ne sont traitées.
while (tagsToSearch.length != 0) {
  //On exclu dès le début de la recherche les balises absentes
  tagsToSearch = tagsToSearch.filter(
    thisTag => htmlContent.indexOf(thisTag[0], indexBeginToRead) != -1,
  );
  //On cherche la première balise
  tagsToSearch.forEach((thisTag, index) => {
    //Pour chaque type de balise on mémorise le premier index de sa balise ouvrante
    var indexOfthisTag = htmlContent.indexOf(thisTag[0], indexBeginToRead);
    thisTag[2] = indexOfthisTag;

    //On cherche quelle est la première balise (index dans tagsToSearch)
    //et son index dans le contenu.
    if (index == 0 || indexOfthisTag < indexOfFirstTag) {
      indexOfFirstTag = indexOfthisTag;
      indexIntagsToSearch = index;
    }
  });
};
```

Une fois la balise ouvrante trouvée, on cherche sa balise fermante, on isole son contenu et on le traite en fonction de sa nature.

Ci-dessous, on traite les figures avec légende, stockées dans un objet avec la forme suivante :

```
{
  tag : '<figure>',
  fileName : '...',
  figureCaption : '...',
}
```

Un exemple de JSON est montré à la partie « présentation d'un test effectué sur le synchroniser »

```
// On s'occupe ici des figures avec image et légende
else if (tagsToSearch[indexIntagsToSearch][0] == '<figure') {
  //on définit les limites de la figure avec les balises
  figureBegin = tagsToSearch[indexIntagsToSearch][2];
  figureStop = htmlContent.indexOf(
    tagsToSearch[indexIntagsToSearch][1],
    tagsToSearch[indexIntagsToSearch][2],
  );
  //on isole le nom de l'image
  fileNameStop =
    htmlContent.substring(figureBegin, figureStop).indexOf('.jpg') +
    figureBegin;
  fileNameBegin =
    htmlContent.substring(figureBegin, fileNameStop).lastIndexOf('/') +
    figureBegin +
    1;
  nameOfFile =
    htmlContent.substring(fileNameBegin, fileNameStop) + '.jpg';
  //on supprime les mentions inutiles
  nameOfFile = nameOfFile.replace(/-scaled/gi, '');
  //On isole la légende de la figure
  figCaptionStop =
    htmlContent
      .substring(figureBegin, figureStop)
      .indexOf('</figcaption>') + figureBegin;
  figCaptionBegin =
    htmlContent.substring(figureBegin, figCaptionStop).lastIndexOf('>') +
    figureBegin +
    1;
  figCaption = htmlContent.substring(figCaptionBegin, figCaptionStop);
  //on traite les apostrophes
  figCaption = figCaption.replace(/&rsquo;/gi, "");
  //On pousse dans le tableau un objet correspondant à la figure
  //contenant le tag, le nom de l'image et la légende.
  arrayArticle.push({
    tag: '<figure>',
    fileName: nameOfFile,
    figureCaption: figCaption,
  });
  //on met à jour la position de la tête de lecture
  indexBeginToRead = figureStop;
}
```

Ici on traite les paragraphes avec hyperliens. (partie hyperliens)

```
//Fonction permettant de traiter les paragraphes avec hyperlien
convertTextWithHyperlink = tagContent => {
  //On crée un objet comportant un tableau correspondant au paragraphe,
  //il contient l'ensemble des blocs textes ou liens
  let pLinkArticle = {
    tag: '<pLink>',
    content: [],
  };
  //On prélève les sous blocs un à un pour les pousser dans le tableau
  while (tagContent.length != 0) {
    //Si le bloc est un lien
    if (tagContent.substring(0, 2) == '<a') {
      //on isole le bloc
      hyperlinkBegin = 0;
      hyperlinkStop = tagContent.indexOf('</a>');
      hyperlink = tagContent.substring(hyperlinkBegin, hyperlinkStop);
      //on isole le texte apparent du lien
      textBegin = hyperlink.lastIndexOf('>') + 1;
      word = tagContent.substring(textBegin, hyperlinkStop);
      //on gère les apostrophes
      word = word.replace(/&rsquo;/gi, "");
      //on isole l'url
      urlStop = hyperlink.lastIndexOf('');
      urlBegin = hyperlink.indexOf('') + 1;
      urlToGo = tagContent.substring(urlBegin, urlStop);
      //on pousse l'objet relatif au lien dans le tableau de l'article
      pLinkArticle.content.push({
        tag: 'link',
        text: word,
        url: urlToGo,
      });
      // on passe au bloc suivant
      tagContent = tagContent.substring(hyperlinkStop + 4);
    }
  }
}
```

Ici on traite les paragraphes avec hyperliens. (partie texte)

```
//Si le bloc est un texte et n'est pas suivi de liens
else if (tagContent.indexOf('<a') == -1){
  //On gère les apostrophes et sauts de lignes
  tagContent = tagContent.replace(/&rsquo;/gi, "");
  tagContent = tagContent.replace(/&nbsp;/gi, ' ');
  //on pousse l'objet relatif au texte dans le tableau de l'article
  pLinkArticle.content.push({
    tag: 'text',
    content: tagContent,
  });
  tagContent = '';
}
// si le bloc est un texte mais contient encore des liens
else {
  //On isole le texte
  textBegin = 0;
  textStop = tagContent.indexOf('<a');
  textContent = tagContent.substring(textBegin, textStop);
  textContent = textContent.replace(/&rsquo;/gi, "");
  textContent = textContent.replace(/&nbsp;/gi, ' ');
  //on pousse l'objet relatif au texte dans le tableau de l'article
  pLinkArticle.content.push({
    tag: 'text',
    content: textContent,
  });
  //on passe au bloc suivant
  tagContent = tagContent.substring(textStop);
}
}
return pLinkArticle;
};
```


b. EXTRAITS DE <CATEGORYPAGE>

Le « Synchronizer » vu plus haut permet de fournir un objet abritant tout le contenu de façon structurée.

Le code qui suit, montre le processus « inverse » qui permet de construire une page à partir de cet objet.

Une <Page> contient des <Article> qui contiennent les composants de bases <H1>, <P>, <UI> ... dérivés des balises standards JSX <Text>, <View>...

Le code ci-dessous est extrait du composant « CategoryPage » qui retourne une <Page>.

```
//fonction retournant du JSX, affiche la page si la synchronisation n'est pas en cours
displayPage = () => {
  //on vérifie si la synchronisation n'est pas en cours
  if (
    this.props.imagesAreLoading == false &&
    this.props.textIsLoading == false
  ) {
    //On retourne un component dédié créé, chargé de construire une page en fonction
    //de la catégorie qu'on lui envoie. On récupère la catégorie via la props
    // envoyée par redux selon le nom envoyé par le menu.
    return (
      <Page
        pageCategory={
          this.props.categories[
            this.props.navigation.state.params.categoryName
          ]
        }
      />
    );
  }
};
```

Le code ci-dessous montre la connexion du composant à Redux.

```
//methode mappant les props reçues du state de redux
const mapStateToProps = state => {
  return {
    imagesAreLoading: state.updateCat.imagesAreLoading,
    categories: state.updateCat.categories,
    textIsLoading: state.updateCat.textIsLoading,
  };
};

//export du composant connecté grâce à la methode connect fournie par redux
//toute modification du contenu (synchronisation) actualisera la nouvelle page
export default connect(mapStateToProps)(CategoryPage);
```

c. EXTRAITS DE <PAGE>

```
import React, {Component} from 'react';
import {ScrollView, View} from 'react-native';
import Article from './Article';
import DynamicStyle from '../styles/DynamicStyle';

export default class Page extends Component {
  constructor(props) {
    super(props);
    this.state = {};
  }

  //Fonction retournant du jsx, affiche une page scrollable contenant des articles
  pageBuilder = () => {
    //On map la catégorie fournie, et on renvoie des articles construits
    //en fonction de l'objet "content".
    return this.props.pageCategory.map((thisCategory, index) => {
      return (
        <View key={index}>
          <Article articleContent={thisCategory.content} key={index}></Article>
          <View style={DynamicStyle.theme.separator} />
        </View>
      );
    });
  };

  // La méthode render permet de renvoyer une vue du jsx
  render() {
    return <ScrollView>{this.pageBuilder()}</ScrollView>;
  }
}
```

d. EXTRAITS D' <ARTICLE>

```
import React, {Component} from 'react';
import {ScrollView} from 'react-native';
import Ul from './blocs/List';
import P from './blocs/Paragraph';
import PLink from './blocs/ParagnaphWithLinks';
import ImgLocal from './blocs/LocalImage';
import H1 from './blocs/Title1';
import H2 from './blocs/Title2';
import H3 from './blocs/Title3';
import H4 from './blocs/Title4';
import FigureLocal from './blocs/FigureWithCaption';

export default class Article extends Component {
  constructor(props) {
    super(props);
    this.state = {};
  }

  //Fonction retournant du jsx, affiche un article.
  articleBuilder = () => {
    //On map l'objet articleContent pour construire l'article
    //en renvoyant les composants un à un
    return this.props.articleContent.map((thisBloc, index) => {
      //en fonction du tag on renvoie le composant dédié créé.
      if (thisBloc.tag == '<ul>') {
        return <Ul bloc={thisBloc} key={index}></Ul>;
      } else if (thisBloc.tag == '<p>') {
        return <P bloc={thisBloc} key={index}></P>;
      } else if (thisBloc.tag == '<pLink>') {
        return <PLink bloc={thisBloc} key={index}></PLink>;
      } else if (thisBloc.tag == '<img>') {
        return <ImgLocal bloc={thisBloc} key={index}></ImgLocal>;
      } else if (thisBloc.tag == '<h1>') {
        return <H1 bloc={thisBloc} key={index}></H1>;
      } else if (thisBloc.tag == '<h2>') {
        return <H2 bloc={thisBloc} key={index}></H2>;
      } else if (thisBloc.tag == '<h3>') {
        return <H3 bloc={thisBloc} key={index}></H3>;
      } else if (thisBloc.tag == '<h4>') {
        return <H4 bloc={thisBloc} key={index}></H4>;
      } else if (thisBloc.tag == '<figure>') {
        return <FigureLocal bloc={thisBloc} key={index}></FigureLocal>;
      }
    });
  };

  // La méthode render permet de renvoyer une vue du jsx
  render() {
    return (
      <ScrollView>{this.articleBuilder()}</ScrollView>
    );
  }
}
```

e. EXTRAITS DE <FIGURELOCAL>

Le code ci-dessous montre un extrait de composant de base, ici chargé de renvoyer les figures avec légende, la suite est présenté ci-après.

```
import React, {Component} from 'react';
import ComponentsApp, {Text$, View$, SafeIcons$} from './ComponentsApp';
import {View, Image} from 'react-native';
import Synchronizer from '../../services/synchronizer';
import DynamicStyle from '../../styles/DynamicStyle';
import {RGBA} from 'react-native-color-matrix-image-filters';

export default class FigureLocal extends Component {
  constructor(props) {
    super(props);
    this.state = {
      //On construit l'url locale de l'image: chemin/nom_de_fichier
      url: 'file://' + Synchronizer.imagesPath + this.props.bloc.fileName,
    };
  }

  //En JSX la taille des images est absolue, on doit donc mesurer l'écran à
  //chaque modification et s'y adapter.
  //Cette methode est déclenché par l'évènement natif "onLayout" lié à la <View>
  updateScreenWidth = w => {
    //On mesure la source de l'image
    Image.getSize(this.state.url, (width, height) => {
      // on la redimensionne avec un ratio optimal pour prendre la largeur de l'écran
      let roundedHeight = Math.round(height * (w / width));
      this.setState({imgWidth: w, imgHeight: roundedHeight});
    });
  };
};
```

```

// Une image avec légende est traduite en Jsx par une <View> contenant
//une image et un <text$> avec des styles spécifiques
render() {
  return (
    <View
      style={{
        marginBottom: 20,
      }}>
    <View
      onLayout={event => {
        width = event.nativeEvent.layout.width;
        this.updateScreenWidth(width);
      }}>
      /* Une filtration conditionnée par le thème est appliquée aux images */
    <RGBA
      blue={DynamicStyle.theme.blueCoeff}
      green={DynamicStyle.theme.greenCoeff}
      red={DynamicStyle.theme.redCoeff}>
      <Image
        source={{uri: this.state.url}}
        style={[
          DynamicStyle.theme.imagesWithCaption,
          {
            width: this.state.imgWidth,
            height: this.state.imgHeight,
            resizeMode: 'contain',
          },
        ]}
      />
    </RGBA>
    </View>
    <Text$>{this.props.bloc.figureCaption}</Text$>
  </View>
);
}
}

```

f. EXTRAITS D'

Ici le composant se charge de renvoyer les listes à puces.

```
import React, {Component} from 'react';
import ComponentsApp, {Text$, View$, SafeIcons$} from './ComponentsApp';
import {View} from 'react-native';
import DynamicStyle from '../../styles/DynamicStyle';

export default class Ul extends Component {
  constructor(props) {
    super(props);
  }

  //Pour traduire une liste HTML en JSX on renvoie une <View>
  //contenant les lignes sous formes de <Text$>
  buildList = ul => {
    //On map le tableau contenant les lignes
    return ul.list.map((li, index) => {
      //on renvoie des lignes précédées d'une puce
      //Le composant créé <Text$> est un <Text> rendu dépendant du thème
      return <Text$ key={index}>{'• ' + li}</Text$>;
    });
  };

  render() {
    return (
      <View style={DynamicStyle.theme.list}>
        {this.buildList(this.props.bloc)}
      </View>
    );
  }
}
```

g. EXTRAITS DE < PLINK>

Le composant <Plink> gère les paragraphes avec des hyperliens.

```
import React, {Component} from 'react';
import {Linking, Text, View} from 'react-native';
import ComponentsApp, {Text$, View$, SafeIcons$} from './ComponentsApp';
import DynamicStyle from '../../styles/DynamicStyle';

export default class PLink extends Component {
  constructor(props) {
    super(props);
  }

  //Pour traduire un paragraphe avec hyperliens HTML en JSX
  //on renvoie une <View> contenant des <Text$> et des <Text> avec onPress
  buildParagraphWithLinks = pLink => {
    //on map le bloc
    return pLink.content.map((p, index) => {
      //pour le text simple on renvoie un <Text$>
      if (p.tag == 'text') {
        return <Text$ key={index}>{p.content}</Text$>;
      }
      //pour les liens on ajoute le lien via onPress
      else {
        return (
          <Text
            style={DynamicStyle.theme.link}
            onPress={() => Linking.openURL(p.url)}
            key={index}>
            {p.text}
          </Text>
        );
      }
    });
  };

  render() {
    return (
      <View style={DynamicStyle.theme.paragraphWithLinks}>
        {this.buildParagraphWithLinks(this.props.block)}
      </View>
    );
  }
}
```

h. EXTRAITS DU « GEOLOCATOR »

PARAMÉTRAGE ET ADMINISTRATION

```
import Store from '../Store/configureStore' //Import du Store pour les dispatchs;
import Geolocation from '@react-native-community/geolocation' //Géolocalisation
import GPSState from 'react-native-gps-state' //Etat de fonctionnement du GPS

//Objet contenant les informations de géolocalisation
geoLocation={
  latitude: 0,
  longitude: 0,
  altitude: 0,
  accuracy:22000,
  altitudeAccuracy:20000,
  heading:{},
  speed: 0,
}

//Booleén reflétant l'état du GPS
gpsIsTurningOff=false;
```

```
class GeoLocator{

  //getter direct permettant d'obtenir la geolocalisation
  //à tout moment sans passer par Redux
  get geoLocation(){
    return geoLocation
  }

  //Dispatch des actions Redux
  //Dispatch alertSettings
  updateLocation(){
    const action = {
      type: "UPDATE_GEOLOCATION",
      value: JSON.parse(JSON.stringify(geoLocation))
    }
    Store.dispatch(action)
  }

  //Dispatch alertSettings
  updateGpsIsTurningOff(){
    const action = { type: "UPDATE_GPS_IS_TURNED_ON", value: gpsIsTurningOff }
    Store.dispatch(action)
  }
}
```


GESTION DE L'ÉTAT DU GPS

```
//.....Gestion de l'etat du GPS.....
//Listener du changement d'état du GPS
addListenerForGpsState(){
    GPSState.addListener((status)=>{
        switch(status){
            case GPSState.NOT_DETERMINED:
                // Non utilisé ici
                break;
            case GPSState.RESTRICTED:
                // On averti les autres composants que le GPS n'est plus effectif
                gpsIsTurningOff=true;
                this.updateGpsIsTurningOff();
                break;
            case GPSState.DENIED:
                // Non utilisé ici
                break;
            case GPSState.AUTHORIZED_ALWAYS:
                // Si le GPS est activé, on lance le watcher de position GPS
                this.watchPositionId();
                break;
            case GPSState.AUTHORIZED_WHENINUSE:
                break;
        }
    })
    GPSState.requestAuthorization(GPSState.AUTHORIZED_WHENINUSE)
}
```

```
//Check l'état du gps au démarrage
checkGpsIsTurningOffOnStart(){
    GPSState.getStatus().then((status)=>{
        //Si le GPS est inactif on averti les composants
        if (status == GPSState.RESTRICTED) {
            gpsIsTurningOff=true;
            this.updateGpsIsTurningOff();
        }
        //S'il est actif on lance le watcher de position
        else if(GPSState.AUTHORIZED_ALWAYS) {
            this.watchPositionId()
        }
    })
}
```

GESTION DE LA BOÎTE DE DIALOGUE

```
//Fonction ouvrant les settings à la demande de
//la boîte de dialogue s'ouvrant en cas de GPS désactivé
openTheLocationSettings =()=>{
    GPSState.openLocationSettings()
}

//Fonction permettant et dispatchant l'état du gps
//Permet de gérer l'affichage de la boîte de dialogue
clearGpsIsTurningOff=()=>{
    gpsIsTurningOff=false;
    this.updateGpsIsTurningOff();
}
```

OBTENTION DE LA POSITION GPS

```
//.....Watching de la position GPS.....
// Retourne la position en cas de déplacement
watchPositionId =()=> Geolocation.watchPosition(
    //Fonction exécutée en cas de déplacement
    (position) => {
        geoLocation=position.coords
        //on dispatch les nouvelles coordonnées et
        //on réinitialise le watcher
        this.updateLocation()
        Geolocation.clearWatch(this.watchPositionId())
    },

    //A exécuter en cas d'erreur
    console.log("error"),

    //Configuration des options
    {
        //On autorise la localisation précise basée sur les GNSS
        enableHighAccuracy: true,
        //temps maximum accordé à l'appareil pour renvoyer une position
        timeout: 20000,
        //âge maximum pour l'utilisation d'une position mise en cache
        maximumAge: 1000,
        //distance parcourue déclenchant le renvoi de position
        distanceFilter:4
    }
);

}

export default new GeoLocator()
```

i. EXTRAITS DE « HOMEPAGE »

Ci-dessous sont présentés quelques extraits de la page d'accueil.

VÉROUILLAGE EN POSITION PORTRAIT

```
//Création de listeners vérifiant si cette page est en cours
//d'affichage afin de bloquer ici la rotation
this._willFocusSubscription = this.props.navigation.addListener(
  'willFocus',
  payload => {
    // Blockage en mode portrait
    Orientation.lockToPortrait();
  },
);
this._willBlurSubscription = this.props.navigation.addListener(
  'willBlur',
  payload => {
    // Déblockage de l'orientation
    Orientation.unlockAllOrientations();
  },
);
```

FERMETURE DE L'ÉCRAN DE DÉMARRAGE

```
componentDidMount() {
  //Peu après le montage on cache le splashScreen
  setTimeout(() => {
    SplashScreen.hide();
  }, 600);
}
```

BOÎTE DE DIALOGUE DE GESTION DU GPS

```
//Fonction affichant la boîte de dialogue en cas de désactivation du GPS
suggestTurnedOnGps() {
  return (
    <View>
      { /* C'est le service geolocator qui averti via redux en cas d'interruption */ }
      <Dialog.Container visible={this.props.gpsIsTurningOff}>
        <Dialog.Title>Voulez-vous activer le GPS?</Dialog.Title>
        <Dialog.Description>
          Pour profiter de toutes les fonctionnalités, vous devez allumer
          votre GPS
        </Dialog.Description>
        <Dialog.Button label="Cancel" onPress={this.maskSuggestGpsOn} />
        { /* Si l'utilisateur veut activer le gps, une fenêtre de réglage est ouverte */ }
        <Dialog.Button label="Yes!" onPress={this.OpenLocSettings} />
      </Dialog.Container>
    </View>
  );
}
```

PARAMÉTRAGE ET GÉNÉRATION DU MENU RADIAL

Un composant «<RadialMenu> a été créé à l'occasion l'application. Il nécessite un tableau de paramétrage listant sous forme d'objets les différents items avec leur spécificités (Référence de l'icône, taille, texte, méthode exécutée au clic, décalages x et y. Ce tableau est stocké dans le state du parent qui le fournit sous forme de props.

Le code ci-dessous montre un extrait du tableau « radialMenuContent ».

```
//Tableau abritant le contenu du menu radial, chaque objet configurant chaque bouton
radialMenuContent: [
  {
    //Référence de l'icône de la bibliothèque "SafeIcon" créée.
    iconName: 'notice',
    //Taille de l'item
    size: 43,
    //Texte à afficher sous l'icône
    text: 'Notice',
    //Méthode à exécuter au onPress
    onPress: () => {
      this.props.navigation.navigate('CategoryPage', {
        categoryName: 'notice',
      });
    },
  },
  {
    iconName: 'torch-light',
    size: 55,
    marginRight: -11,
    marginBottom: -12,
    text: 'Torche',
    onPress: () => this.toggleTorchLightState(),
  },
  {
    iconName: 'discreet-light',
    size: 46,
    marginRight: 0,
    marginBottom: 0,
    text: 'Lampe',
    text2: 'discrète',
    onPress: () => {
      this.props.navigation.navigate('DiscreetLight');
    },
  },
],
```

j. EXTRAITS DU <RADIALMENU>

A partir du tableau fourni, du nombre d'items, des coordonnées du centre, du rayon, et de l'angle de départ, une méthode calcule automatiquement la position des différents éléments et les renvoie.

Le code ci-dessous montre la première partie de la méthode, la suite est visible ci-après.

```
//Génération dynamique d'un menu radial à partir des props:
//le tableau contenant les items du menu
//le rayon, les coordonnées du centre,
//la taille des items, l'angle de départ
generateRadialMenu = (
  radialMenuContent,
  x0,
  y0,
  radius,
  startAngle,
  itemSize,
) => {
  //On map le contenu
  return radialMenuContent.map((item, i) => {
    //En fonction des paramètres et du nombre d'items fournis
    //On calcul les positions par trigonométrie
    let angle =
      (Math.PI * startAngle) / 180 -
      Math.PI / 2 +
      (2 * Math.PI * i) / radialMenuContent.length;
    let x = radius * Math.cos(angle) - itemSize / 2 + x0;
    let y = radius * Math.sin(angle) - itemSize / 2 + y0;
```

```

//Renvoi du bouton
return (
  <TouchableOpacity
    key={i}
    onPress={item.onPress}
    style={{
      position: 'absolute',
      width: itemSize,
      height: itemSize,
      alignItems: 'center',
      justifyContent: 'center',
      left: x,
      top: y,
    }}>
    <SafeIcons$
      name={item.iconName}
      size={item.size}
      style={[
        DynamicStyle.theme.iconHomeMenu,
        {
          marginRight: item.marginRight || 0,
          marginBottom: item.marginBottom || 0,
        },
      ]}
    />
    <Text style={DynamicStyle.theme.textHomeMenu}>{item.text}</Text>
    {item.text2 && (
      <Text style={DynamicStyle.theme.textHomeMenu}>{item.text2}</Text>
    )}
  </TouchableOpacity>
);
});
};

```

2. PRÉSENTATION D'UN TEST EFFECTUÉ SUR LE « SYNCHRONISER »

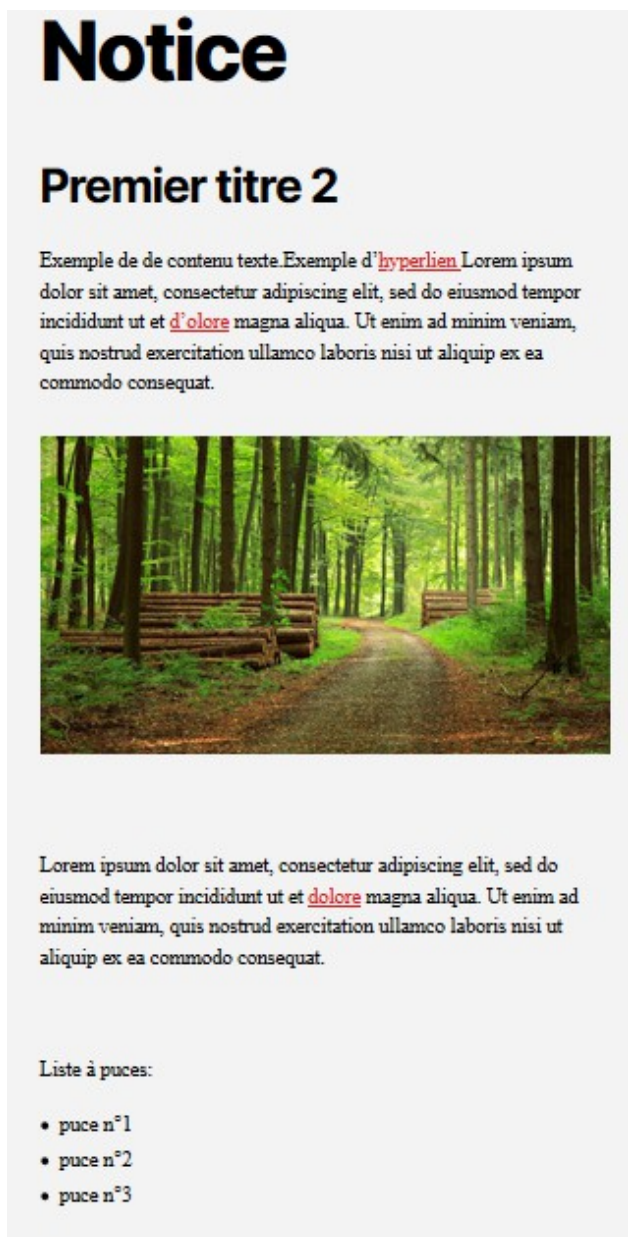
a. SCÉNARIO

Entre l'édition des articles sur le back-office et l'affichage sur l'écran d'un mobile de nombreuses étapes entrent en jeu.

Le contenu est édité dans le back-office en HTML, il est récupéré par l'application via l'API, transformé en JSON, stocké, et retourné en JSX pour être affiché. Les images sont téléchargées et stockées en local.

Un jeu de test a été effectué pour vérifier le bon déroulement du processus.


DONNÉES EN ENTRÉE



Notice

Premier titre 2

Exemple de de contenu texte. Exemple d'[hyperlien](#) Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut et [dolore](#) magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut et [dolore](#) magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Liste à puces:

- puce n°1
- puce n°2
- puce n°3

Capture d'écran de l'extrait de l'article de test

```
<h1>Notice</h1>\n<h2>Premier titre 2</h2>\n<p>Exemple de de contenu texte.Exemple d&rsquo;<a href=\"http://www.google.fr\">hyperlien </a>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod t empor incididunt ut et <a href=\"http://ww w.valentin-besnardiere.com\">d&rsquo;olore</a> magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut a liquip ex ea commodo consequat.</p>\n<p><img class=\"alignnone size-full wp-image-246\" src=\"https://bushcraft.patte-blanche.fr/wp-content/uploads/2019/12/forest.jpg\" alt=\"\" width=\"860\" height=\"480\" srcset=\" https://bushcraft.patte-blanche.fr/wp-content/uploads/2019/12/forest.jpg 860w, https://bushcraft.patte-blanche.fr/wp-content/uploads/2019/12/forest-500x279.jpg 500w, https://bushcraft.patte-blanche.fr/wp-content/uploads/2019/12/forest-768x429.jpg 768w\" sizes=\"(max-width: 860px) 100vw, 860px\" /></p>\n<p>&nbsp;</p>\n<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut et <a href=\"http://www.valentin-besnardiere.com\">dolore</a> magna aliqua. Ut enim ad minim veniam, quis nostrud ex ercitation ullamco laboris nisi ut aliquip e x ea commodo consequat.</p>\n<p>&nbsp;</p>\n<p>Liste à puces:</p>\n<ul>\n<li>puce n° 1</li>\n<li>puce n°2</li>\n<li>puce n°3</li>\n</ul>
```

Extrait HTML de l'article de test renvoyé par l'API

DONNÉES ATTENDUES

Le JSON attendu est sensé mapper l'HTML et contenir de façon hiérarchisé l'ensemble des balises recherchées et leur contenu.

A partir de ce JSON, Le JSX créé doit permettre de retourner l'article en conservant la structure et avec le style désiré.

DONNÉES OBTENUES

```
1.{notice: Array(2), introduction: Array(1), s_orienter: Array(0), abri: Array(0), champignons: Array(0), ...}
  1.notice: Array(2)
    1.0:
      1.title: "notice"
      2.content: Array(19)
        1.0:
          1.tag: "<h1>"
          2.content: "Notice"
          3.__proto__: Object
        2.1:
          1.tag: "<h2>"
          2.content: "Premier titre 2"
          3.__proto__: Object
        3.2:
          1.tag: "<pLink>"
          2.content: Array(5)
            1.0: {tag: "text", content: "Exemple de de contenu
            texte.Exemple d'"}
            2.1: {tag: "link", text: "hyperlien
            ", url: "http://www.google.fr"}
            3.2: {tag: "text", content: "Lorem ipsum dolor sit amet,
            consectetur adipiscing elit, sed do eiusmod tempor incididunt
            ut et "}
            4.3: {tag: "link", text: "d'olore", url: "http://www.valentin-
            besnardiere.com"}
            5.4: {tag: "text", content: " magna aliqua. Ut enim ad minim
            veniam, quis nostr... laboris nisi ut aliquip ex ea commodo
            consequat."}
            6.length: 5
            7.__proto__: Array(0)
          3.__proto__: Object
        4.3:
          1.tag: "<img>"
          2.fileName: "forest.jpg"
          3.__proto__: Object
        5.4:
          1.tag: "<p>"
          2.content: " "
          3.__proto__: Object
        6.5:
          1.tag: "<pLink>"
          2.content: Array(3)
            1.0: {tag: "text", content: "Lorem ipsum dolor sit amet,
            consectetur adipiscing elit, sed do eiusmod tempor incididunt
            ut et "}
            2.1: {tag: "link", text: "dolore", url: "http://www.valentin-
            besnardiere.com"}
            3.2: {tag: "text", content: " magna aliqua. Ut enim ad minim
            veniam, quis nostr... laboris nisi ut aliquip ex ea commodo
            consequat."}
            4.length: 3
            5.__proto__: Array(0)
          3.__proto__: Object
        7.6:
          1.tag: "<p>"
          2.content: " "
          3.__proto__: Object
        8.7:
          1.tag: "<p>"
          2.content: "Liste à puces:"
          3.__proto__: Object
        9.8:
          1.tag: "<ul>"
          2.list: Array(3)
            1.0: "puce n°1"
            2.1: "puce n°2"
            3.2: "puce n°3"
```

Extrait du JSON de l'article de test



Capture d'écran mobile de l'extrait de l'article de test

b. ANALYSE DES RÉSULTATS

Le test a confirmé le bon fonctionnement des algorithmes. L'ensemble du contenu a pu être transmis et affiché comme il était attendu.

3. EXEMPLE D'UNE RECHERCHE ANGLOPHONE EFFECTUÉE

a. PRÉSENTATION

Contexte :

Afin de respecter la charte graphique de l'application, il était nécessaire d'installer des polices d'écriture dans le projet react-native. Il a donc fallu trouver la procédure.

Mots clefs recherchés :

« Install local font react native »

Sites retournés :

- 1 - <https://stackoverflow.com/questions/41825276/how-to-add-custom-font-in-react-native-android>
- 2 - <https://blog.bam.tech/developer-news/add-a-custom-font-to-your-react-native-app>
- 3 - <https://medium.com/@mehran.khan/ultimate-guide-to-use-custom-fonts-in-react-native-77fcdf859cf4>
- 4 - <https://medium.com/react-native-training/react-native-custom-fonts-ccc9aacf9e5e>
- 5 - https://dev.to/edo_begagic/add-custom-fonts-to-react-native-expo-app-54d

Critères de choix des sites sélectionnés :

Je me suis concentré sur le 1er et 3ème site. Stackoverflow est un site de référence en informatique qui permet de trouver des réponses sérieuses relatives au développement web. Le système de vote du site permet de mettre en exergue les réponses de qualités.

J'ai éliminé le 5ème site qui concernait l'utilisation de « expo ».

J'avais à de multiples reprises trouvé des tutoriels de qualité sur le site medium, mais c'est surtout la date de publication qui m'a poussé à me concentrer sur ce contenu.

En effet, React-Native étant une technologie relativement jeune et « mouvante » l'âge des ressources consultées est un critère primordial pour ne pas tomber sur des informations obsolètes.

Ressource choisie : <https://medium.com/@mehran.khan/ultimate-guide-to-use-custom-fonts-in-react-native-77fcdf859cf4>

Solutions mises en œuvre :

Le tutoriel traduit ci-dessous a été suivi.

b. EXTRAIT



To make Medium work, we log user data. By using Medium, you agree to our [Privacy Policy](#), including cookie policy.

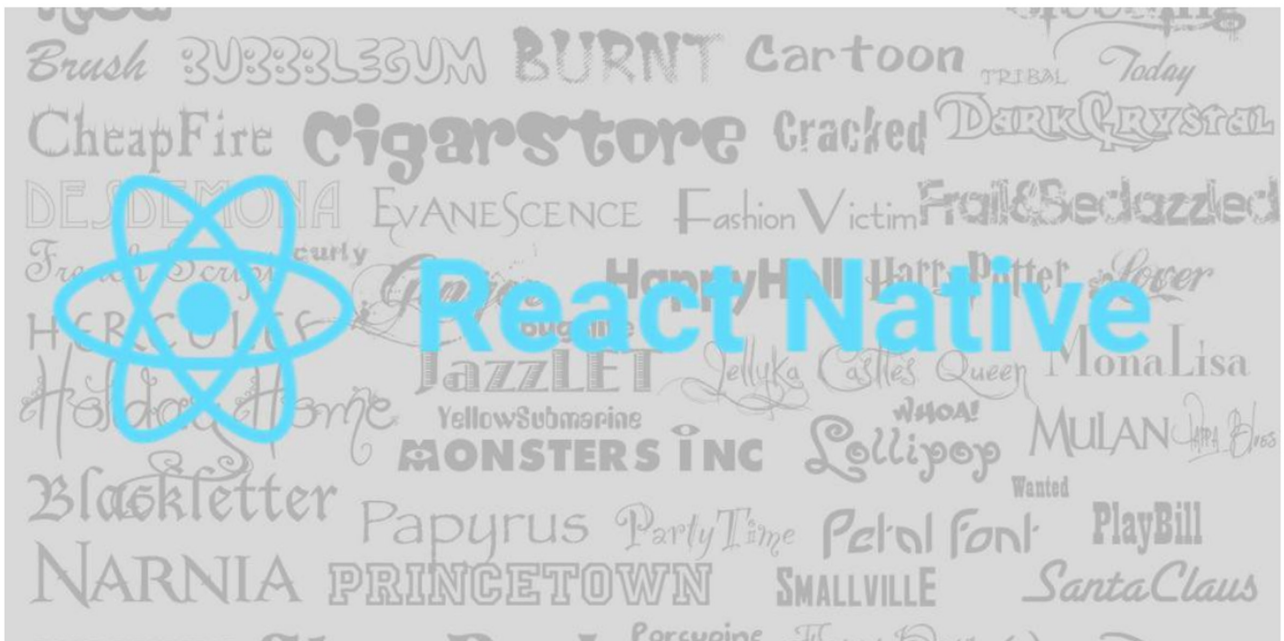
Join a member [Sign in](#)

[Get started](#)

Ultimate guide to use custom fonts in react native



Mehran Khan [Follow](#)
Jan 12, 2019 · 4 min read



Adding custom fonts in react native project is very simple task with **react-native link command** but can be tricky if you are a new in react native .

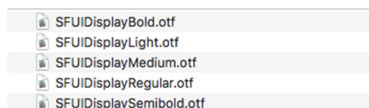
I have decided to write article on **all the possible issues** which can occur while implementing custom fonts and give you **detail information** about how it will work.

Following are steps to link fonts(Using React Native Link Command):

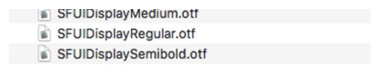
1. First step (Set Font Naming For Cross Platform)

First step is to select a font family you want to use in your react native app. I have decided to use “SF UI Display” for my sample test project.

You need to check the file name before using it . Android will use the font name but in IOS it will use the “PostScript name”. It is tricky part let me explain to you.

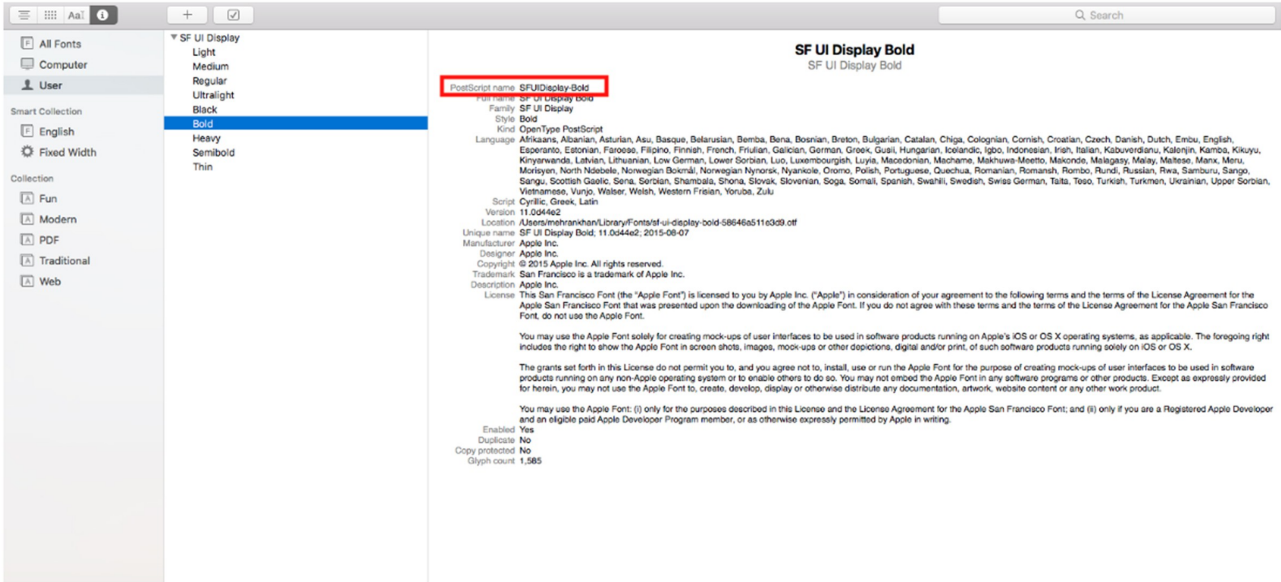


Suppose i have above font files and i am using “SFUIDisplayBold” in my

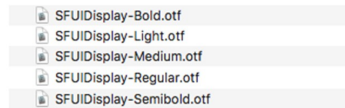


Suppose i have above font files and i am using “SFUIDisplayBold” in my react native code it will work in ANDROID but not in IOS because the “postscript name” is “SFUIDisplay-Bold”. So for IOS you have to use “SFUIDisplay-Bold” despite of the fact that you have added font file with name “SFUIDisplayBold.ott”

You can check the “postscript name” of a font file in Font book



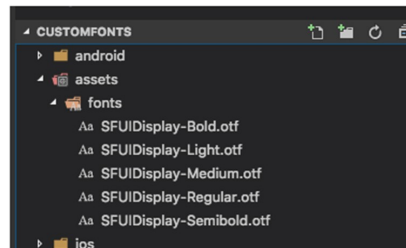
After renaming following are list of my font files .

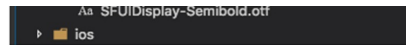


Note : Although you can add logic in your code to use “SFUIDisplayBold” for Android and “SFUIDisplay-Bold” for IOS but it is better to rename the font .

Step 2 : Add Fonts to Assets

Next add all the font files you want to us in “assets/fonts” directory (It can vary)





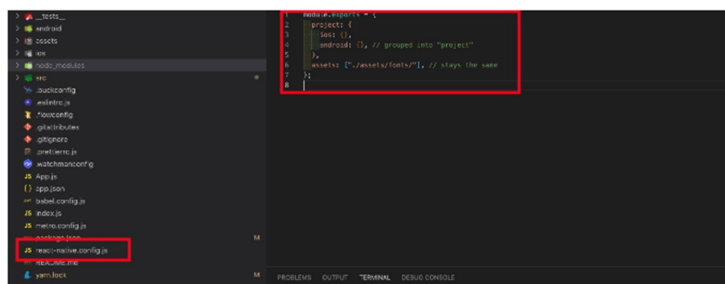
Step 3 : Define assets directory

if React Native Version \geq 0.60 [\(Detail\)](#)

Create File “react-native.config.js” and add following code

```
1 module.exports = {
2   project: {
3     ios: {},
4     android: {}, // grouped into "project"
5   },
6   assets: [".assets/fonts/"], // stays the same
7 };
```

react-native.config.js hosted with ❤️ by GitHub [view raw](#)



if React Native Version $<$ 0.60

You need to tell react native where our custom fonts are located . Adding the following lines in your package.json

```
1 "rnpm": {
2   "assets": [
3     ".assets/fonts/"
4   ]
5 }
```

package.json hosted with ❤️ by GitHub [view raw](#)

Path can vary. if you have “src” folder and you want to put assets in “src” folder then path will be “.src/assets/fonts/”

Step 4 : Link assets using react native link

Run in terminal

```
$ react-native link
```

link command will links fonts in Info.plist for IOS and creates fonts directory (android/app/src/main/assets/fonts) for Android, where copies your fonts

Step 5 : Use font in React Native Styles

Finally you can use font in styles by adding “fontFamily”

c. TRADUCTION

Guide ultime pour utiliser des typographies personnalisées dans react native

L'ajout de typographies dans un projet react native est une tâche très simple avec la commande react native link mais peut être délicat si vous êtes novice en react native.

J'ai décidé d'écrire un article sur tous les problèmes possibles qui peuvent apparaître quand on implémente une police personnalisée et de vous donner des informations détaillées sur comment ça fonctionne.

Ce qui suit décrit les étapes pour lier des polices (en utilisant la commande react native link) :

1. Première étape (définir un nom de police pour le multiplateforme)

La première étape est de choisir une typographie que vous voulez utiliser dans votre application react native. J'ai décidé d'utiliser « SF UI Display » pour mon exemple de projet test.

Vous devez vérifier le nom de fichier avant de l'utiliser. Android utilisera le nom de police mais IOS utilisera le nom « PostScript ». C'est une partie délicate, laissez-moi vous expliquer.

...

BILAN DU PROJET

Le projet a pu être mené à bien et les objectifs ont été atteints. La conception rigoureuse du cahier des charges validé par l'entreprise, un planning prévisionnel précis, et une méthode de travail réactive de type "Agile", ont permis de livrer l'application et son back-office dans les conditions escomptées.

Une attention particulière a été portée au code et à sa documentation afin de pérenniser l'exploitation du projet. Le code a été largement commenté, et une documentation fournie a été produite pour permettre une prise en main rapide du projet. Le back-office a été conçu pour pouvoir être utilisé par des personnes ne possédant pas de compétences en développement.

Enfin, une réunion de présentation est prévue avec l'équipe de Patte Blanche. Un plan de suivi sera exposé et des évolutions seront proposées. Une présentation avec formation au back-office sera également proposée.

CONCLUSION DU STAGE

La conception et le développement de **Safe+** a été une expérience professionnelle et humaine particulièrement enrichissante. La confiance et l'autonomie qui m'ont été accordées m'ont permis d'expérimenter l'étendue d'un projet.

Safe+ se présentait comme un défi singulier et stimulant : En deux mois, imaginer, concevoir, développer, et designer une application mobile, en utilisant une technologie que je découvrais : React Native.

J'ai pu ainsi mettre en œuvre une variété de compétences : la gestion de projet, la planification prévisionnelle, la conception intégrale d'un cahier des charges, la création d'une identité et d'une charte graphique, le maquettage d'une application, le web design, le graphisme, le développement avec le framework React Native dans un environnement Node, l'utilisation d'un CMS et d'une API, et le respect d'un calendrier.

D'autre part j'ai pu apprendre un nouveau logiciel, Adobe Illustrator, qui m'offre de nouvelles possibilités, comme la création de mes typographies et de mes propres bibliothèques d'icônes.

Développer avec React Native a été stimulant, la technologie fourmille, et la communauté est active. Si on peut encore ressentir une certaine instabilité, le dynamisme rend son utilisation très intéressante.

Enfin, j'ai découvert une entreprise avec laquelle je partage les valeurs et qui m'a accueilli chaleureusement. Ce cadre propice m'a permis de mener à bien ce projet, d'atteindre les objectifs visés en respectant le calendrier.

LEXIQUE

1 : RSE : La Responsabilité Sociétale des Entreprises (en anglais, *Corporate Social Responsibility* ou *CSR*) désigne la prise en compte par les entreprises, sur base volontaire, des enjeux, sociaux et éthiques dans leurs activités.

2 : B-Corp : La certification dite "B Corp" est une certification octroyée aux sociétés commerciales répondant à des exigences sociétales et environnementales, de gouvernance ainsi que de transparence envers le public. Cette certification est attribuée par B Lab, un organisme sans but lucratif établi aux États-Unis.

3 : Gouvernance partagée : Système de gouvernance non pyramidale qui réconcilie la structure décisionnelle et la participation de chacun à un projet collectif. La gouvernance est partagée par des « cercles » occupés par les salariés.

4 : Club de Rome : Groupe de réflexion réunissant des scientifiques, des économistes, des fonctionnaires nationaux et internationaux, ainsi que des industriels de 52 pays, préoccupés des problèmes complexes auxquels doivent faire face toutes les sociétés, tant industrialisées qu'en développement.

5 : World : Modèle qui permet une simulation informatique des interactions entre population, croissance industrielle, production de nourriture et limites des écosystèmes terrestres. L'approche est du type « dynamique des systèmes » qui tente de comprendre le comportement des systèmes complexes dans le temps et qui prend en compte les boucles de rétroaction internes et les effets retard qui affectent le comportement global du système.

6 : Low-tech : La low-tech ou basse technologie est un ensemble de techniques simples, pratiques, économiques et populaires. Ce sont des solutions techniques qui cherchent à être simples, bien pensées, bien dimensionnées et réparables. C'est leur design réfléchi qui fait leur efficacité.

7 : Survivalisme : Depuis quelques années, le survivalisme se démocratise et tend à devenir protéiforme. Il regroupe de nombreux domaines qui s'inscrivent dans des contextes variés mais liés: La collapsologie, le bushcraft*, la résilience aux catastrophes (naturelles ou anthropiques). Le survivalisme peut désigner les activités de certains individus ou groupes d'individus qui apprennent à faire face à toute sorte d'aléas ou catastrophes, notamment l'effondrement de la civilisation industrielle.

Plus globalement c'est aussi une approche philosophique, qui souligne l'importance de la capacité d'un individu à pouvoir assurer son intégrité physique et psychologique en étant capable de subvenir à ses besoins primaires.

8 : Bushcraft : Ce terme qualifie le réapprentissage d'une vie prospère avec la nature, ainsi que l'acquisition de compétences plus ou moins anciennes.

9 : Collapsologie : La collapsologie est un courant de pensée apparu au début du XXI^e siècle qui étudie les Risques d'effondrements environnementaux et sociétaux et ce qui pourrait succéder à la société actuelle. La collapsologie a été développée en France au sein de l'Institut Momentum co-fondé par Yves Cochet et Agnès Sinaï. Ces derniers définissent l'effondrement comme « le processus à l'issue duquel les besoins de base (eau, alimentation, logement, habillement, énergie, etc.) ne sont plus fournis (à un coût raisonnable) à une majorité de la population par des services encadrés par la loi ».

10 : GNSS : GNSS (Géolocalisation et Navigation par un Système de Satellites¹) est un système de positionnement par satellites. C'est un ensemble de composants reposant sur une constellation de satellites artificiels permettant de fournir à un utilisateur par l'intermédiaire d'un récepteur portable de petite taille sa position 3D, sa vitesse 3D et l'heure. Cette catégorie de système de géo-positionnement se caractérise par une précision métrique et une couverture mondiale.

La quasi totalité des smartphones supporte le système américain GPS, une grande partie peut aussi exploiter le système russe GLONASS, ou chinois BEIDOU pour augmenter la qualité du positionnement. Une proportion moindre supporte déjà le nouveau système européen GALILEO.

11 : UX : L'expérience utilisateur (en anglais, user expérience, abrégé UX) est la qualité du vécu de l'utilisateur dans des environnements numériques ou physiques.

12 : UI : UI est l'abréviation d'user interface ou interface utilisateur. L'UI design se rapporte donc à l'environnement graphique dans lequel évolue l'utilisateur d'un logiciel.

13 : MD5 : Le MD5, pour Message Digest 5, est une fonction de hachage cryptographique qui permet d'obtenir l'empreinte numérique d'un fichier (on parle souvent de message). Il a été inventé par Ronald Rivest en 1991. Si l'algorithme MD5 présente un intérêt historique important, il est aujourd'hui considéré comme dépassé et absolument impropre à toute utilisation en cryptographie ou en sécurité.

14 : Bcrypt : bcrypt est une fonction de hachage basée sur l'algorithme de chiffrement Blowfish. En plus de l'utilisation d'un sel pour se protéger des attaques par table arc-en-ciel (rainbow table), bcrypt est une fonction adaptative, c'est-à-dire que l'on peut augmenter le nombre d'itérations pour la rendre plus lente. Ainsi elle continue à être résistante aux attaques par force brute malgré l'augmentation de la puissance de calcul.

15 : HTTPS : L'HyperText Transfer Protocol Secure (HTTPS, littéralement « protocole de transfert hypertextuel sécurisé ») est la combinaison du HTTP avec une couche de chiffrement comme SSL ou TLS.