|  |
| --- |
| SartrouInc. |
| App\_Name |
| Dossier de spécification |

|  |
| --- |
| Kévin REMOUÉ  Pierre TOUZÉ  09/11/2015 |

Historique des versions:

| Date | Actions | Auteur | Version |
| --- | --- | --- | --- |
| 09/11/2015 | Création | P. TOUZE | 0.1 |
| 11/11/2015 | Ajout de la portée, des acronymes, des références et du copyright | P. TOUZE | 0.2 |
| 11/11/2015 | Ajout de L’archi. logicielle et matérielle, acteurs | K. REMOUE | 0.3 |
| 14/11/2015 | Retouches suite aux commentaires de la 0.3 | P. TOUZE  K. REMOUE | 0.4 |
| 16/11/2015 | Ajout d’une ébauche des interfaces logiques | P. TOUZE | 0.5 |
| 19/11/2015 | Maj 2.2, 2.4, 2.6 | K. REMOUE | 0.6 |
| 21/11/2015 | Description étendue des interfaces logiques | P. TOUZE  K. REMOUE | 0.7 |
| 20/12/2015 | Traduction des actions en anglais. Séparation des actions spécifiqaues à la V2 dans un document à part | P. TOUZE | 0.8 |

Table des matières

[1 Introduction 3](#_Toc435476459)

[1.1 Objet 3](#_Toc435476460)

[1.2 Portée 3](#_Toc435476461)

[1.3 Définitions, acronymes et abréviations 3](#_Toc435476462)

[1.4 Références 4](#_Toc435476463)

[1.5 Vue d’ensemble 4](#_Toc435476464)

[1.6 Copyright 4](#_Toc435476465)

[2 Description Générale 4](#_Toc435476466)

[2.1 Caractéristiques des acteurs 4](#_Toc435476467)

[2.1.1 Acteur direct 4](#_Toc435476468)

[2.1.2 Acteur indirect 4](#_Toc435476469)

[2.2 Environnement 4](#_Toc435476470)

[2.2.1 Architecture matérielle et logicielle 4](#_Toc435476471)

[2.2.2 Interfaces du système 5](#_Toc435476472)

[2.3 Fonctions principales développées 7](#_Toc435476473)

[2.3.1 Cas d’usage considéré pour App\_Name 7](#_Toc435476474)

[2.3.2 Résumé des cas d’utilisation stratégiques 7](#_Toc435476475)

[2.3.3 CU 01 7](#_Toc435476476)

[2.3.4 CU 02 7](#_Toc435476477)

[2.4 Contraintes 7](#_Toc435476478)

[2.4.1 Politiques réglementaires 7](#_Toc435476479)

[2.4.2 Contraintes matérielles 7](#_Toc435476480)

[2.4.3 Exigences de fiabilité 7](#_Toc435476481)

[2.4.4 Exigences de maintenabilité 7](#_Toc435476482)

[2.4.5 Exigences de disponibilité 7](#_Toc435476483)

[2.5 Hypothèses et Dépendances 7](#_Toc435476484)

[2.6 Répartition des exigences 7](#_Toc435476485)

[3 Exigences spécifiques 8](#_Toc435476486)

[3.1 Interface Homme Machine 8](#_Toc435476487)

[3.1.1 Généralités 8](#_Toc435476488)

[3.1.2 Actions utilisateur 8](#_Toc435476489)

[3.1.3 Ecrans 8](#_Toc435476490)

[3.2 Dictionnaire du domaine 8](#_Toc435476491)

# Introduction

## Objet

Ce dossier de spécification a pour objectif de définir les fonctionnalités et exigences attendues pour le développement de l’application App\_Name et des composants nécessaires à son fonctionnement. Le client étant ici l’utilisateur final ce dossier se focalise sur l’utilisation du produit et les exigences sont définies suivant les attentes connues d’un utilisateur possédant un smartphone.

Ce document a pour but de permettre de concevoir, développer, et tester les parties logicielles de l’application App\_Name.

Ce dossier de spécification suit les recommandations de la norme IEEE 830-1998 [IEEE-830\_1998]. Il utilise des schémas et illustrations respectant la norme UML en version 2.4.1 [UML\_2.4\_2011].

## Portée

Ce document décrit les fonctionnalités et exigences du Système à l’Etude (SaE) constitué de :

* Le logiciel App\_Name gérant l’interface sur smartphone Android
* Une base de données permettant de stocker les informations de manière persistante
* Une REST API permettant d’interroger la base de données

Les trois composants constitutifs du SaE piloteront des périphériques et communiqueront via des protocoles qui seront décrits plus loin dans ce document.

Les contraintes liées au matériel sur lequel les différents composants fonctionneront seront également évoquées ultérieurement dans ce document.

## Définitions, acronymes et abréviations

Les abréviations utilisées dans le présent document sont répertoriées et expliquées dans le tableau présenté ci-dessous. Les termes utiles pour interpréter correctement ce dossier de spécifications sont définis dans le dictionnaire de domaine présent dans ce dossier dans la partie 3.3.

|  |  |
| --- | --- |
| Acronymes, abréviation | Définition |
| API (Application Program Interface) | Un ensemble de fonctions proposées par un logiciel offrant des services à un autre logiciel |
| CU | Cas d’utilisation |
| HTTP (HyperText Transfert Protocol) | Protocole de communication client-serveur développé pour le Web. |
| IEEE (Institute of Electrical  and Electronics Engineers) | Association professionnelle internationale définissant entre autres des normes dans le domaine informatique et électronique. |
| OMG (Object Management  Group) | Association professionnelle internationale définissant entre autres des normes dans le domaine informatique. |
| REST API (REpresentational State Transfer Application Program Interface) | Une API permettant d’utiliser l’architecture Web pour faire des requêtes HTTP. |
| SaE (Système à l’Etude) | Il regroupe tous les composants décrits dans le paragraphe [Portée](#_Portée). |
| A COMPLETER |  |

## Références

Voici un tableau récapitulatif des différents documents utilisés pour l’élaboration de ce dossier de spécification :

|  |  |
| --- | --- |
| Tag de la référence | Référence complète |
| [IEEE-830\_1998] | IEEE, std 830-1998 « Recommended Practice for Software Requirements Specifications »,  http://standards.ieee.org/findstds/standard/830-1998.html, 1998. |
| [UML\_2.4\_2011] | OMG, “Unified Modeling Language”, version 2.4.1,  http://www.omg.org/spec/UML/, 2011. |
| A COMPLETER |  |

## Vue d’ensemble

Ce document de spécification est structuré en 3 parties :

* La partie I présente les objectifs et la portée de ce document.
* La partie II, intitulée « description générale », a pour objectifs de présenter l’environnement et le contexte du projet Project\_Name, ainsi que les fonctionnalités principales attendues pour le SaE.
* La partie III présente les IHM attendues, les fonctionnalités détaillées du Project\_Name ainsi que le dictionnaire du domaine.

## Copyright

N/A

# Description Générale

App\_Name est un système permettant de partager entre amis des bars, restaurants et clubs. App\_Name se différencie de TripAdvisor (par exemple). En effet, les amis partagent seulement les lieux qu’ils apprécient, il n’y a pas de système de notation (qui peut être parfois faussé). De plus, l’application se veut très simple d’utilisation et de recherche d’un lieu contrairement à TripAdvisor où une recherche peut être longue car il y a beaucoup de paramètres et de lieux.

Une application Android permet d’interagir avec ses amis et d’utiliser le système. Un serveur permet de stocker les différentes informations.

## Caractéristiques des acteurs

### Acteur direct

Utilisateur :

L’utilisateur est la personne qui souhaite découvrir des lieux que ses amis ont apprécié ou partager avec ses amis les lieux qu’il a lui-même apprécié en interagissant avec eux via des groupes. Il devra s’identifier via un compte afin de pouvoir interagir avec ses amis et ne pourra rejoindre que les groupes dont il connait le nom.

### Acteur indirect

N.A

## Environnement

### Architecture matérielle et logicielle

Le diagramme ci-dessous représente l’architecture logicielle et matérielle du SaE. Il identifie les entités matérielles et logicielles avec lesquelles le SaE doit interagir et permet ainsi de déterminer les principaux échanges qu’il entretient avec son environnement.

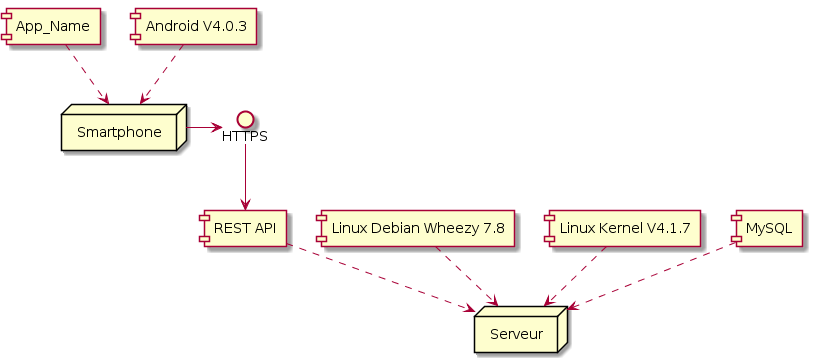


Figure 1 : Architecture matérielle et logicielle

Entités matérielles :

* **Smartphone :** Il s’agit d’un téléphone intégrant l’OS Android. Plusieurs terminaux Android peuvent posséder l’application et se connecter (via des requêtes HTTPS) sur le serveur.
* **Serveur :** il s’agit pour le moment d’une Raspberry Pi 2 connectée à internet et accessible par chaque smartphone via des requêtes HTTPS.

Entités logicielles :

* **Android V4.0.3 :** Version d’Android fonctionnant sur le smartphone. La version doit être 4.0.3 ou ultérieure afin d’assurer la couverture de 94% du parc de smartphones.
* **App\_Name :** constitue l’application Android présente sur chaque smartphone.
* **REST API :** composant logiciel présent sur le serveur permettant de répondre aux requêtes HTTPS des différents terminaux Android.
* **Linux Debian Wheezy 7.8 :** version du linux présent sur la Raspberry Pi 2. (voir aussi Kernel)
* **MySQL :** composant permettant de stocker les différentes informations nécessaires au fonctionnement de l’application Android.

### Interfaces du système

Cette section décrit les interfaces du SaE c'est à dire les entrées et les sorties du système. Ces entrées/sorties sont divisées en deux grandes catégories, les interfaces logiques et les interfaces physiques:

• Les interfaces logiques représentent les interactions, c'est à dire les évènements et les données échangées, entre les utilisateurs et les périphériques. Elles décrivent la volonté d'un acteur interagissant avec le SaE.

• Les interfaces physiques montrent les échanges entre les périphériques et le SaE. Elles représentent les données réellement échangées.

#### Interfaces avec les acteurs

Les interactions avec l’utilisateur se font directement depuis le SaE.

##### Entrées venant de l’utilisateur

Les entrées logiques venant de l’utilisateur à destination du SaE sont :

* createGroup (GroupeId) : L’utilisateur crée un nouveau groupe d’amis en lui attribuant un nom sous la forme d’une chaîne de caractères appelée GroupeId. Cela entraine la génération d’une clé, GroupeKey, unique que l’utilisateur pourra communiquer à ses amis afin que ceux-ci puissent rejoindre le groupe.
* seePlaceInfo(Place) : L’utilisateur clique sur un lieu de la carte, symbolisé par un marqueur, afin d’en obtenir les informations :
  + Nom du lieu, PlaceName
  + Adresse du lieu, PlaceAddress
  + Type de lieu, PlaceType
  + Avis de l’informateur, SpotterComment
  + Nom de l’informateur, SpotterName
  + Commentaires des amis passés par ce lieu et leur nom, FriendsComments
  + Fourchette de prix sous la forme d’un nombre de symboles «  € »
* sharePlace (Place) : L’utilisateur choisit d’ajouter un lieu qu’il a apprécié en recherchant le nom du lieu puis en ajoutant un commentaire. L’icône du lieu sera attribuée en fonction du type
* writeComment (Place, Comment) : L’utilisateur ajoute un commentaire sur un lieu déjà existant afin de partager son avis aux autres membres du groupe.
* joinGroup(GroupKey) : L’utilisateur renseigne le GroupKey d’un groupe afin de pouvoir le rejoindre et interagir avec les membres de ce groupe.
* connect(Login, MotDePasse) : L’utilisateur se connecte afin de pouvoir accéder aux groupes d’amis et de pouvoir partager des lieux.
* seePlaces () : l’utilisateur accède à la carte pour visualiser les lieux partagés par ses amis.

##### Sorties allant vers l’utilisateur

Les sorties logiques allant du SaE vers l’utilisateur sont :

* afficherEcran(Ecran) : Affiche l’écran passé en paramètre sur l’IHM.

La liste des écrans disponibles est la suivante :

* + MapScreen
  + ConnectionScreen
  + CreateGroupScreen
  + SharePlaceScreen
  + FilterSelectionScreen
  + FirstUseScreen
* displayPlaces(Filters, Groups) : affiche les types de lieux choisis des groupes sélectionnés sur la carte de l’écran MapScreen.
* displayPlacesInfo(Lieu) : Afficher les détails d’un lieu : son adresse et les commentaires ajoutés.

#### Interfaces physiques

N/A

#### Interfaces avec les logiciels

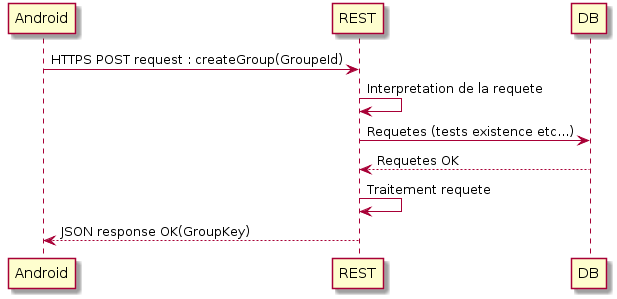


Figure : CreateGroup

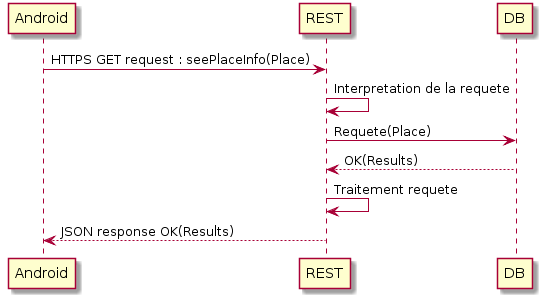


Figure : SeePlaceInfo

#### Interfaces de communication

Décrites dans l’archi matérielle

## Fonctions principales développées

### Cas d’usage considéré pour App\_Name

### Résumé des cas d’utilisation stratégiques

### CU 01

### CU 02

## Contraintes

### Politiques réglementaires

N.A

### Contraintes matérielles

L’application mobile s’exécutera sur des terminaux possédant une version d’Android 4.0.3 et plus. L’application s’affichera au format portrait sur des écrans d’une taille etc…

Le serveur, actuellement sur une Raspberry Pi 2, devra être visible publiquement et accessible via des requêtes HTTP. (Voir 2.2.1)

### Exigences de fiabilité

N.A

### Exigences de maintenabilité

N.A

### Exigences de disponibilité

Le système doit être disponible 24h/24 et 7j/7.

## Hypothèses et Dépendances

TODO

## Répartition des exigences

N.A

# Exigences spécifiques

## Interface Homme Machine

### Généralités

### Actions utilisateur

### Ecrans

#### Vue générale

#### Ecran\_1

#### Ecran\_2

## Dictionnaire du domaine