

TITRE CONCEPTEUR INTEGRATEUR DEVOPS

BLOC n° 1 – Développement d'applications informatiques

Cahier des Charges de la MSPR « Développement d'une solution applicative en langage Java à partir d'un cahier des charges »

Activité type A1 : Développer, tester et intégrer de façon continue des solutions applicatives adaptées à l'infrastructure

COMPETENCES EVALUEES :

Etre capable de :

- . Concevoir une solution algorithmique
 - Construire un algorithme en langage des algorithmiques (LDA) afin de répondre à des besoins de programmation.
 - Adapter un algorithme aux contraintes d'un langage de programmation
 - Valider une solution algorithmique par exécution dans un langage de programmation
- . Concevoir et développer une solution applicative objet
 - Maquetter une application objet
 - Programmer à l'aide des concepts de programmation orientée Objet
 - Développer une solution applicative à l'aide des langages Objet : Java – J2EE
- . Créer du code avec l'intégration et livraison continues
 - Assurer la qualité d'une application
 - Utiliser un outil d'intégration tel que : Jenkins ou Maven vision Java

PHASE 1 : PREPARATION DE CETTE MISE EN SITUATION PROFESSIONNELLE RECONSTITUEE

Durée de préparation : 20 heures

Mise en œuvre : Travail d'équipe constituée de 4 apprenants-candidats (5 maximum si groupe impair)

Résultat attendu :

Deux applications, correspondant aux sujets

- Compétences langage objet
- Compétence langage web et nomade

PHASE 2 : PRESENTATION ORALE COLLECTIVE + ENTRETIEN COLLECTIF

Durée totale par groupe : 60 mn se décomposant comme suit :

- 20 mn de soutenance orale par l'équipe ;
- 40 mn d'entretien collectif avec le jury (questionnement complémentaire).

Objectif : mettre en avant et démontrer que les compétences visées par ce bloc sont bien acquises

Jury d'évaluation : 2 personnes (binôme d'évaluateurs) par jury – Ces évaluateurs ne sont pas intervenus durant la période de formation et ne connaissent pas les apprenants à évaluer.

PLAN DU DOCUMENT

SUJET 1.1 : COMPETENCES LANGAGES OBJET	4
I - PRESENTATION DE L'ENTREPRISE/CLIENT A L'ORIGINE DU BESOIN	4
II – DESCRIPTION DU SYSTEME D'INFORMATION DU CLIENT ET DE SES SPECIFICATIONS.....	5
III – CONTEXTE DU BESOIN (CAHIER DES CHARGES).....	5
3.1 – LES OBJECTIFS.....	5
3.2 – LES CIBLES	6
3.3 – LE TYPE DE SOLUTION.....	6
3.4 – LES BESOINS FONCTIONNELS	6
3.5 – LES BESOINS NON FONCTIONNELS	6
3.6 – GRAPHISME ET ERGONOMIE	6
3.7 – AUTRES RESSOURCES	9
IV- EXPRESSION DE LA DEMANDE – REALISATION ATTENDUE DE L'APPRENANT .	10
V - SPECIFICATION(S) TECHNIQUE(S), DOCUMENTATION, ANNEXES, SCHEMA, BASES... OU TOUT ELEMENT NECESSAIRE A LA BONNE REALISATION DU SUJET	10
5.1 – DIAGRAMME DE SEQUENCE	11
VI – BESOINS SPECIFIQUES	11
SUJET 1.2 : COMPETENCES LANGAGE WEB ET NOMADE	12
I – PRESENTATION DE L'ENTREPRISE/CLIENT A L'ORIGINE DU BESOIN	12
II – DESCRIPTION DU SYSTEME D'INFORMATION DU CLIENT ET DE SES SPECIFICATIONS.....	13
III– CONTEXTE DU BESOIN (CAHIER DES CHARGES).....	13
3.1 – CARACTERISTIQUES TECHNIQUES.....	13
3.2 – LES OBJECTIFS.....	17
3.3– LES CIBLES	17
IV – EXPRESSION DE LA DEMANDE – REALISATION ATTENDUE DE L'APPRENANT	17
V – SPECIFICATION(S) TECHNIQUE(S), DOCUMENTATION, ANNEXES, SCHEMA, BASES... OU TOUT ELEMENT NECESSAIRE A LA BONNE REALISATION DU SUJET	18
5.1 – DIAGRAMMES DE CAS D'UTILISATION	18
5.2 – DIAGRAMMES DE SEQUENCES	20
VI – BESOINS SPECIFIQUES	23

SUJET 1.1 : COMPETENCES LANGAGES OBJET

I - PRESENTATION DE L'ENTREPRISE/CLIENT A L'ORIGINE DU BESOIN

Depuis 2005, la société « Go Securi » intervient dans le cadre de la sécurité, du gardiennage, de la protection et de la surveillance. Leur mission consiste à assurer la sécurité des activités, déplacements et manifestations de leur client.



La société intervient dans le gardiennage, la surveillance et la télésurveillance de sites ainsi que dans la sécurité d'évènements.

« Go Securi » met à disposition des agents professionnels, hautement qualifiés et formés à toute situation relevant de la sécurité, de la surveillance, du gardiennage et de la protection.

La société compte 64 salariés dont 53 agents de sécurité.

Les agents partagent un local où sont stockés le matériel. Le matériel est parfois sensible et doit être stocké dans un local sécurisé. Aujourd'hui, il n'existe pas de contrôle d'accès au local automatisé, les agents remplissent une fiche d'affectation de leur matériel tous les jours avant leur service. Cette pratique est obsolète, et avec le nombre grandissant de collaborateurs, de plus en plus compliquée à gérer. Aujourd'hui la société souhaite se tourner vers une solution sécurisée et informatique de l'accès au local, mais aussi de la gestion du matériel.

Les équipements mis à disposition des agents de sécurité sont variés :

- Mousqueton
- Gants d'intervention
- Brassard de sécurité
- Porte menottes
- Bandeau agent de sécurité cynophile

- Talkies walkies
- Lampe Torche
- Kit oreillette
- Tasers
- Bombes lacrymogènes

II – DESCRIPTION DU SYSTEME D’INFORMATION DU CLIENT ET DE SES SPECIFICATIONS

La société n’a actuellement pas de système d’information. En l’absence de SI, le choix se portera sur Google Firebase pour traiter les informations de la solution.

III – CONTEXTE DU BESOIN (CAHIER DES CHARGES)

Le but de cette demande est de concevoir une solution permettant de vérifier l’accès au local sécurisé en prenant une photo de l’utilisateur via une webcam. L’accès au local sera donc autorisé aux seuls agents dont la photo sera présente en base de données.

Ainsi, la réalisation de ce projet nécessite de concevoir et de développer une solution intelligente permettant l’extraction et la comparaison des photos prises par la webcam et celles présentes sur un serveur.

La photo prise et le résultat de l’accès positif ou négatif s’affichera sur une interface utilisateur. En cas d’accès positif, un écran s’affichera permettant à l’utilisateur de cocher le matériel qu’il utilisera durant sa journée d’intervention.

Le choix de la solution se portera sur le développement d’une application développée en Java.

Quant au choix du système d’exploitation sur lequel fonctionnera l’application, il reste libre (Windows, Linux, MacOS), ainsi que le matériel (Ordinateur portable, Raspberry, PC Fixe ...).

3.1 – Les objectifs

Le travail demandé sur ce projet consiste à :

- La prise de photo d’un agent de sécurité via un webcam
- La comparaison de cette photo avec un ensemble de photos d’identité déjà présentes sur un serveur distant

- Configuration d'une base de données Google Firebase intégrant notamment la liste des agents ainsi que la liste du matériel
- Développer une application Java qui réalisera l'acquisition d'image, la comparaison de deux images, ainsi que l'affichage d'une interface utilisateur qui servira à la validation de l'accès au local ainsi qu'à l'affichage d'un formulaire de saisie du matériel
- Communication entre la base de données Firebase et l'application Java

3.2 – Les cibles

L'application sera utilisée par l'ensemble des agents de sécurité de la société.

3.3 – Le type de solution

Ce projet a pour objectif de développer un système sur-mesure de vérification de l'accès à un local, ainsi qu'à l'enregistrement de l'affectation du matériel présent aux agents précédemment identifiés.

Les solutions sur-mesure permettent de proposer un système parfaitement adapté aux besoins de l'utilisateur. C'est l'un des espoirs sur lesquels se repose cette demande.

3.4 – Les besoins fonctionnels

La solution proposée doit satisfaire d'une façon informelle les fonctionnalités suivantes :

- L'acquisition d'information : au cours de cette première étape, la caméra doit prendre une image d'une personne
- La vérification : c'est l'étape de vérification proprement dite. Les contrôles de l'identité de la personne sont effectués par le système. Le système doit réaliser une comparaison entre les informations de l'image prise par la webcam et un ensemble d'images stockées sur un serveur
- La réponse : un rapport de vérification est produit, il est envoyé à l'application.
- L'enregistrement d'information : Après un accès validé, un écran d'information récapitulant l'ensemble du matériel disponible doit s'afficher. L'utilisateur peut alors enregistrer le matériel qu'il utilisera durant sa journée d'intervention. L'enregistrement se fera en base de données.

3.5 – Les besoins non fonctionnels

- La disponibilité en permanence et la capacité : à tout moment le système permet de vérifier l'identité de l'utilisateur en temps réel
- La performance : l'application doit respecter ce critère pour assurer le fonctionnement des fonctionnalités répondant aux besoins des utilisateurs avec le minimum de temps de réponse.
- L'intégrité : en cas de détection d'erreur ou de non conformités des images de l'utilisateur, l'appareil doit être capable de traiter ces cas.
- L'ergonomie : les interfaces de l'application doivent être claires, intuitives, simples et conviviales et on doit essayer au maximum d'éviter l'encombrement.

3.6 – Graphisme et ergonomie

3.6.1 - La charte graphique

La charte graphique de l'interface utilisateur devra reprendre des couleurs principales du logo de la société, soit le noir (#000000), le vert (#659224) et le bleu (#379EC1).

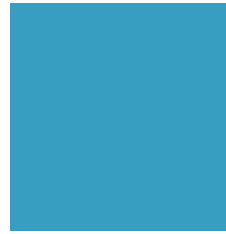


Deux couleurs prises à partir du logo de la société seront à utiliser :



#000000

Pour le texte



#379EC1

Titres et boutons

Titres / Boutons :

Ce format est utilisé pour indiquer le nom de la fonctionnalité sur laquelle l'utilisateur se trouve.

Style : Medium, majuscule à la 1ère lettre uniquement

Texte :

Ce format est utilisé pour tous les textes hors format spécifique.

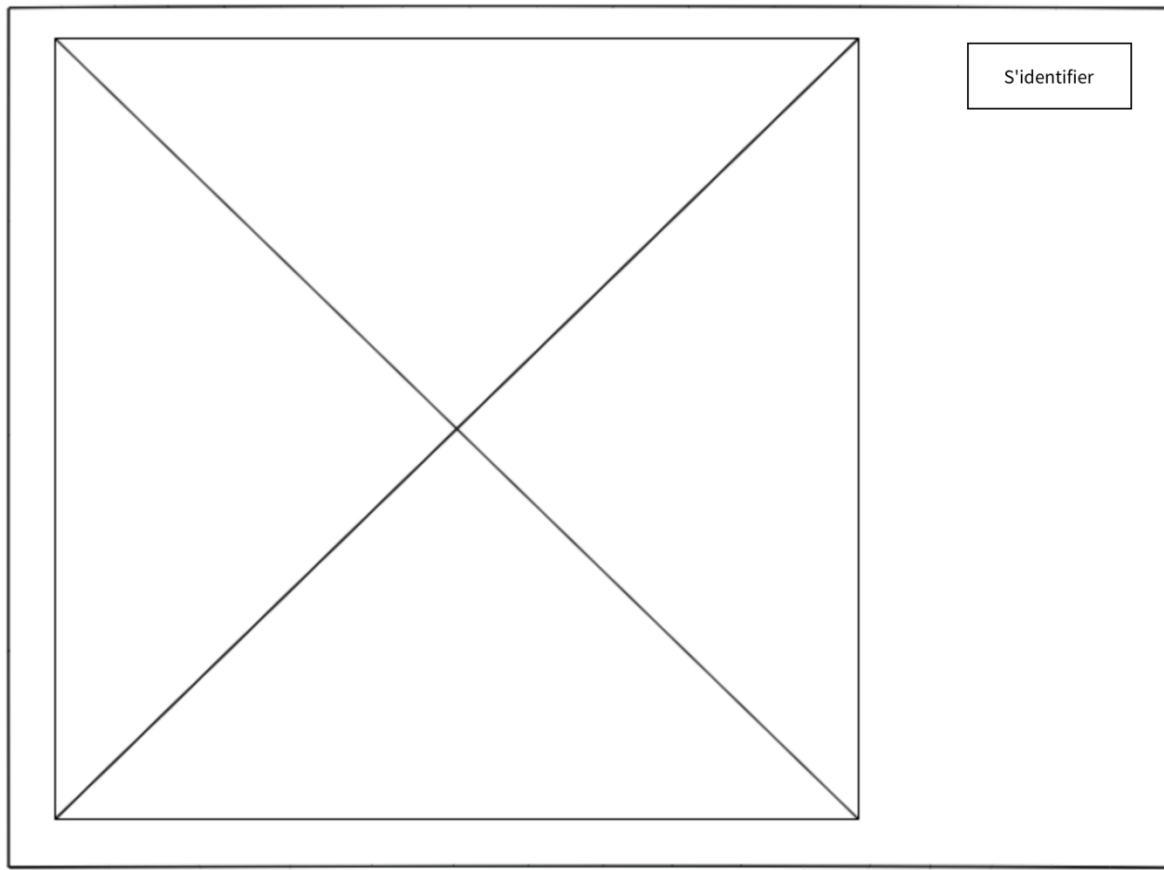
Style : Light

La police à utiliser sur l'application sera la police Roboto.

<https://www.dafont.com/fr/roboto.font>

3.6.2- Wireframes et maquettings

Écran principal

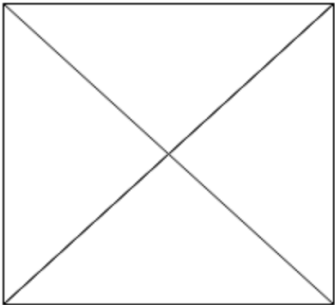


Sur cet écran s’affiche la vue caméra ainsi qu’un bouton qui permet de prendre la photo. Un fois la photo prise, en cas de résultat négatif, une fenêtre popup s’affiche informant l’utilisateur du résultat.

Dans le cas contraire, l’écran suivant s’affiche.

Ecran de saisie du matériel

Identification



Mousqueton

☒

Gant d'intervention

☒

Ceinture de sécurité tactique

☒

Détecteur de métaux

☒

Brassard de sécurité

☒

Lampe torche

☒

Gilet pare balle

☒

L'écran de saisie du matériel s'affiche lors d'un résultat positif de l'identification de l'agent de sécurité. Les informations affichées sont l'image de l'agent (celle issue du serveur) ainsi que la liste du matériel restant dans le local.

Le bouton « Identification » fait revenir l'utilisateur à la page principale.

Ex : Si 15 agents ont déjà pris des mousquetons précédemment, la ligne « Mousqueton » sera grisée. Elle sera dégrisée une fois qu'un autre agent aura décoché la case mousqueton.

Lors d'un deuxième scan d'un agent, la liste du matériel s'affiche. Une fois son matériel remis dans le local, il pourra alors décocher le matériel sur le formulaire indiquant qu'il l'a rendu.

3.7 – Autres ressources

3.7.1 – Liste du matériel présent dans le local

- 15 mousquetons
- 10 gants d'intervention
- 20 ceintures de sécurité tactique
- 25 détecteurs de métaux
- 30 brassards de sécurité
- 5 lampes torches

- 5 bandeaux « Agents cynophiles »
- 12 gilets pare-balles
- 30 chemises manches courtes
- 30 blousons
- 30 coupe-vents
- 20 talkies walkies
- 10 kits oreillettes
- 5 tasers

IV- EXPRESSION DE LA DEMANDE – REALISATION ATTENDUE DE L'APPRENANT

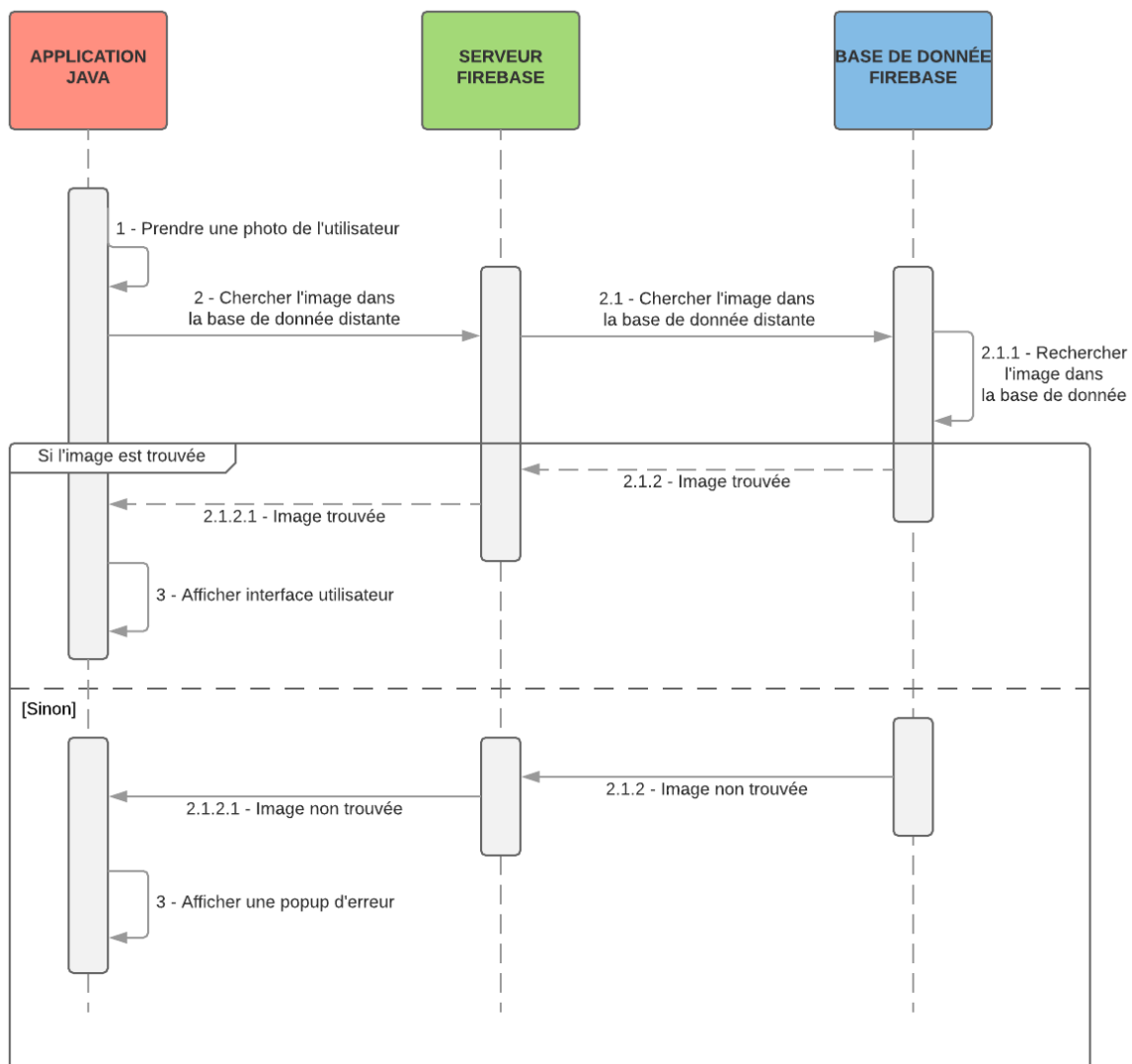
La réalisation attendue de l'apprenant est de se mettre à la place du prestataire réalisant l'application.

Il devra être décisionnaire tout au long du projet et proposer une solution en cohérence avec le cahier des charges du client. Cela nécessite qu'il montre son savoir-faire sur les sujets suivants :

- Construire des algorithmes afin de répondre aux besoins fonctionnels de l'application, **un algorithme est attendu de la part de l'étudiant** :
 - Un algorithme de comparaison de deux images : l'algorithme devra prendre en entrée deux images données et fournir un indice de comparaison qui déterminera si les deux visages présents sont la même personne. Cet indice sera un pourcentage donné.
- Présenter cet algorithme en LDA (Langage Descriptif des Algorithmes)
- Développer une solution applicative en langage Java
- Développer en langage Objet
- Mettre en place une architecture logicielle en phase avec la cible (MVP, MVC, MVVM ...)
- Mettre en place un outil d'intégration continue : Jenkins ou Maven
- Configurer le déploiement de l'application via Jenkins ou Maven (Les livrables attendus devront être générés uniquement à partir de la plateforme d'intégration continue !)
- Assurer la qualité des livrables via la plateforme d'intégration continue
- Respecter un cahier des charges suivant des points précis : charte graphique, besoins fonctionnels, diagrammes de séquences, diagrammes d'états.

V - SPECIFICATION(S) TECHNIQUE(S), DOCUMENTATION, ANNEXES, SCHEMA, BASES...

Diagramme de séquence



Ce diagramme de séquence représente l'enchainement des actions et des interactions entre l'utilisateur, l'application et le serveur.

VI – BESOINS SPECIFIQUES

L'apprenant devra avoir à disposition :

- Un ordinateur Fixe sous Windows / Linux / MacOS équipé d'une webcam + Écran
- Ou

- Un ordinateur portable sous Windows / Linux / MacOS équipé d'une webcam
- Ou
- Un Raspberry PI sous Linux équipé d'une caméra + webcam

SUJET 1.2 : COMPETENCES LANGAGE WEB ET NOMADE

I – PRESENTATION DE L'ENTREPRISE/CLIENT A L'ORIGINE DU BESOIN

Depuis 2005, la société « Go Securi » intervient dans le cadre de la sécurité, du gardiennage, de la protection et de la surveillance. Leur mission consiste à assurer la sécurité des activités, déplacements et manifestations de leur client.



La société intervient dans le gardiennage, la surveillance et la télésurveillance de sites ainsi que dans la sécurité d'évènements.

« Go Securi » met à disposition des agents professionnels, hautement qualifiés et formés à toute situation relevant de la sécurité, de la surveillance, du gardiennage et de la protection.

La société compte 64 salariés dont 53 agents de sécurité.

Il arrive très souvent que ces agents doivent vérifier des documents d'identité lors de différentes manifestations ou de leur mission de surveillance. Ils n'ont aujourd'hui pas à disposition d'outils digitaux / informatiques qui leur permettent d'effectuer ces vérifications de façon sûre et rapide.

II – DESCRIPTION DU SYSTEME D'INFORMATION DU CLIENT ET DE SES SPECIFICATIONS

La société n'a actuellement pas de système d'information. En l'absence de SI, le choix se portera sur Google Firebase pour traiter les informations de la solution.

III– CONTEXTE DU BESOIN (CAHIER DES CHARGES)

Le but de cette demande est de concevoir une solution qui permet de vérifier un document officiel afin de donner la permission d'accès à un bâtiment pour un visiteur.

En effet, le contrôle d'identité a un but préventif. Nous avons choisi pour ce projet de vérifier la carte d'identité nationale aussi appelée pièce d'identité, puisque c'est le document le plus utilisé. La carte d'identité est sécurisée, elle possède de nombreuses informations qui permettent de la prémunir contre la falsification.

Ainsi, la réalisation de ce projet nécessite de concevoir et de développer une solution intelligente permettant l'extraction et la vérification de données personnelles du visiteur, l'acquisition et le développement d'un système d'information pour assurer la gestion des visiteurs.

Le choix de la solution se portera sur le développement d'une application mobile Android développée en Java.

L'ensemble des agents de sécurité sur le terrain seront équipés d'un téléphone Android Samsung Galaxy S7 Active.

3.1 – Caractéristiques techniques



Le choix s'est porté sur ce smartphone pour ses nombreux atouts de solidité : la résistance aux chutes, à l'eau et à la poussière mais aussi sa grande autonomie. On retrouve les mêmes performances, l'appareil photo et le stockage extensible du S7 original.

Marque :	Samsung
Autres noms :	SM-G891A, G891A
Version Android :	Android 6.0.1 Marshmallow
Taille :	74.9 mm x 148.8 mm x 9.9 mm
Ratio d'aspect :	16 :9
Poids :	185 g
Surface utilisée :	64 %
Matériaux :	Polycarbonate
Certificats de résistance (poussière, eau) :	IP68, MIL-STD-810G

Écran

Diagonale :	5.1"
Type :	Super AMOLED
Résolution :	1440 x 2560 px (QHD)
Densité :	Très haute densité, 576 Pixels par pouce différenciables à moins de 5 cm. de distance
Autres : résistant	Capacitive, Corning Gorilla Glass 4, Multi-touch, boutons physiques, Scratch

Processeur

Modèle :	Qualcomm Snapdragon 820 MSM8996 2.2GHz
CPU :	Dual cluster Kryo 2x2.2GHz + 2x1.6GHz
Type :	Quad-Core
Fréquence de l'horloge :	2.2 GHz
64 Bits :	Oui

Graphismes

GPU :	Adreno 530 624MHz
-------	--------------------------

RAM

RAM :	4 GB
-------	-------------

Antutu

Score :	140.955 (Antutu v6) Meilleur rendement que 92% des appareils
---------	---

Stockage

Capacité : de ROM) : 57-63 GB)	64 GB (Capacité avec le système d'exploitation (dépend du fabricant et de la version
-----------------------------------	---

Équivaut environ à : **11636 Chansons, 25600 Photos et 914 Vidéos**

Extensible ? (Slot SD) : **Oui, SD en slot indépendant**

Sécurité

Lecteur d'empreinte : **Oui, sur le devant**

Capteurs

Fingerprint : **Oui**

Accéléromètre : **Oui**

Baromètre : **Oui**

Boussole : **Oui**

Gyroscope : **Oui**

Heart Rate : **Oui**

Lumière ambiante : **Oui**

Proximité : **Oui**

SpO2 : **Oui**

Autres

Notifications LED : **Non**

Appareil photo

Arrière

Résolution : **12.2 Mpx**

Capteur : **Sony IMX260 Exmor RS**

Type : **CMOS**

Ouverture : **f/1.7**

Flash : **Dual LED**

Stabilisation optique : **Oui**

Vidéo Slow Motion : **Non, 30.0 fps**

Caractéristiques : **Autofocus, Déclenchement continu ou rafale, Stabilisateur d'image numérique, Zoom Digital, Exposure compensation, Détection de visage, Géo-étiquetage, HDR, ISO**

settings, Macro mode, Optical image stabilization (OIS), Photos panoramiques, RAW, Scene mode, Self-timer, Mise au point tactile, White balance settings

Selfie

Résolution : **5 Mpx**

3.2 – Les objectifs

Le travail demandé sur ce projet consiste à :

- L'extraction des informations de la carte d'identité nationale tel que : nom, prénom, numéro de la carte d'identité et photo
- Configuration d'une base de données Google Firebase
- Développer une application Android qui sera la HDMI (Human Machine Interface)
- Communication entre la base de données Firebase et l'application Android
- Récupérer les informations de la carte d'identité dans l'interface Android
- L'enregistrement des informations du visiteur dans la base de données Firebase

3.3– Les cibles

L'application sera utilisée par l'ensemble des agents de sécurité de la société. Chaque agent sera équipé d'un smartphone Android disposant de l'application mobile.

IV – EXPRESSION DE LA DEMANDE – REALISATION ATTENDUE DE L'APPRENANT

La réalisation attendue de l'apprenant est de se mettre à la place du prestataire réalisant l'application mobile.

Il devra être décisionnaire tout au long du projet et proposer une solution en cohérence avec le cahier des charges du client. Cela nécessite qu'il montre son savoir-faire sur les sujets suivants :

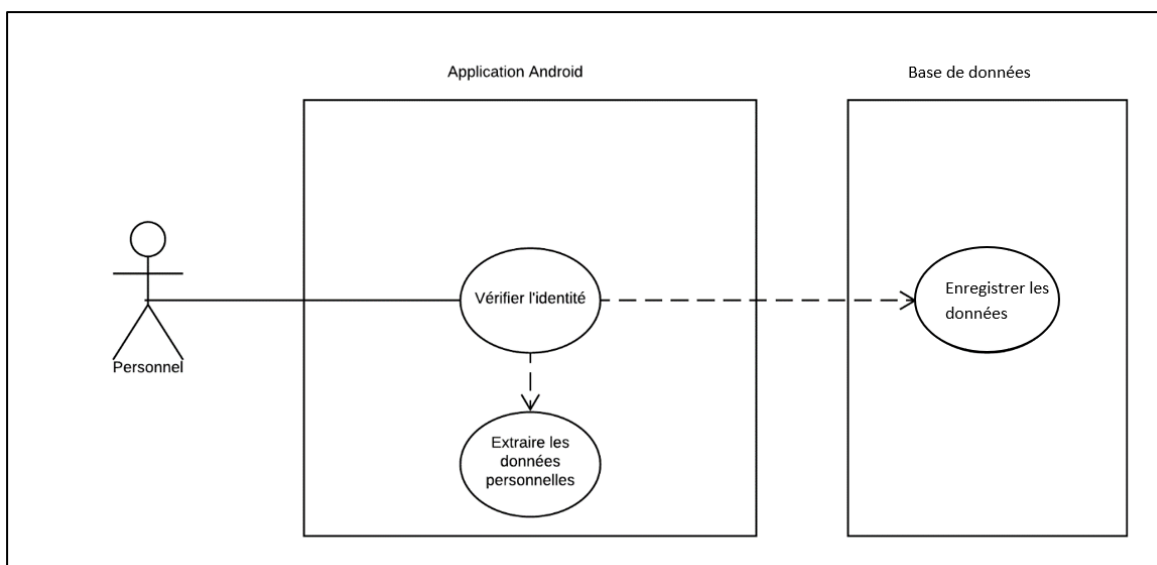
- Construire un algorithme afin de répondre aux besoins fonctionnels de l'application (traitement et comparaison de deux images, acquisition d'information à partir d'image ...)
- Développer une solution applicative en langage Java
- Développer en langage Objet
- Mettre en place une architecture logicielle en phase avec la cible (MVP, MVC, MVVM ...)

- Mettre en place un outil d'intégration continue : Jenkins ou Maven
- Configurer le déploiement des applications via Jenkins ou Maven (les livrables attendus devront être générés uniquement à partir de la plateforme d'intégration continue !)
- Assurer la qualité des livrables via la plateforme d'intégration continue
- Respecter un cahier des charges suivant des points précis : charte graphique, besoins fonctionnels, wireframes, diagrammes de séquences, diagrammes d'états.

V – SPECIFICATION(S) TECHNIQUE(S), DOCUMENTATION, ANNEXES, SCHEMA, BASES... OU TOUT ELEMENT NECESSAIRE A LA BONNE REALISATION DU SUJET

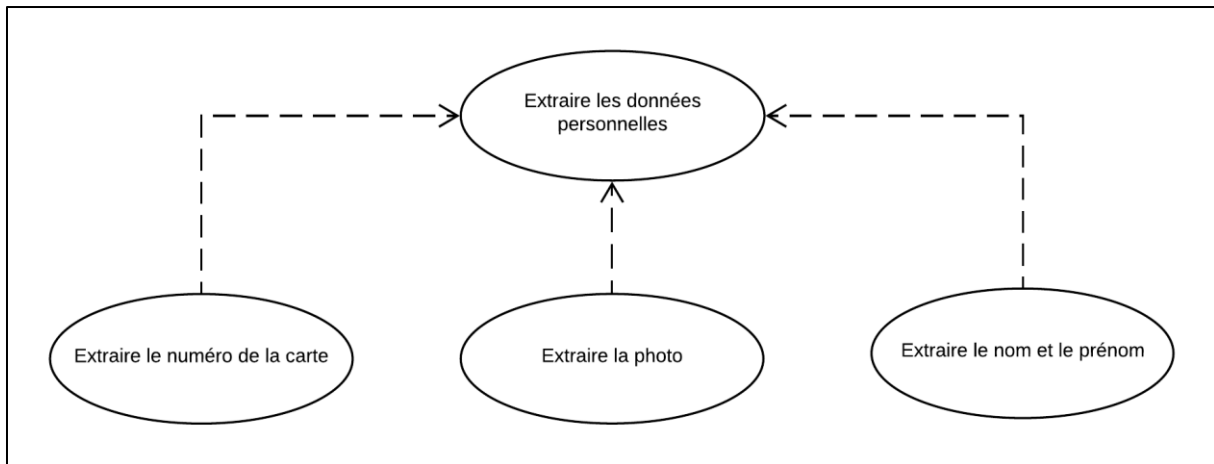
5.1 – Diagrammes de cas d'utilisation

Diagramme de cas d'utilisation général



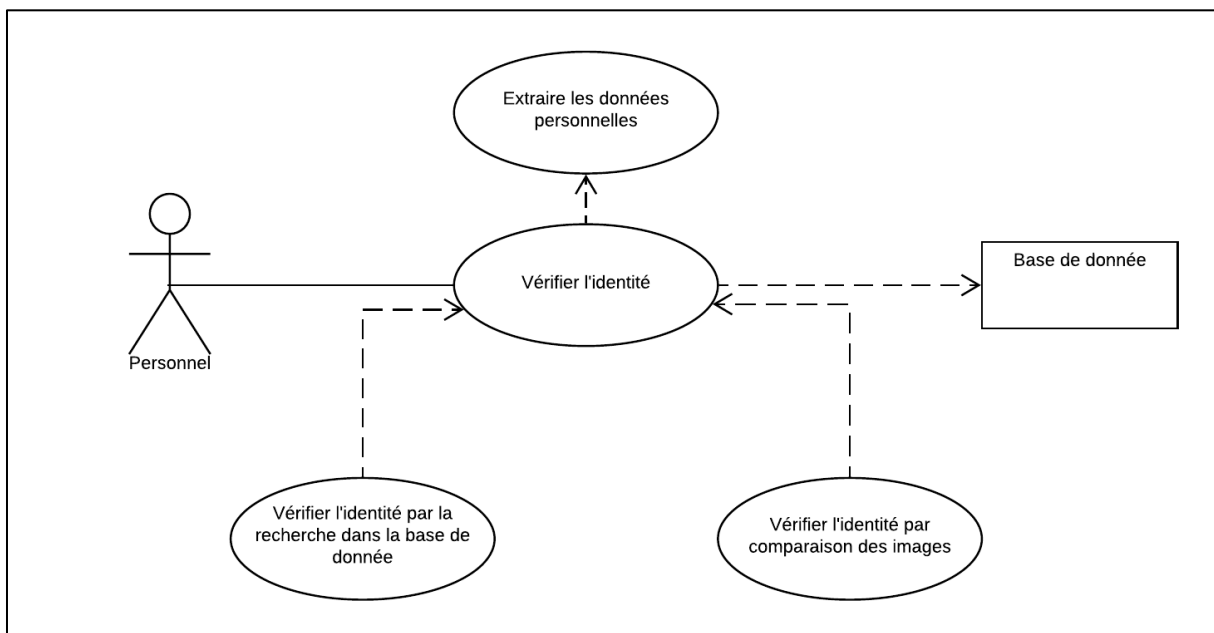
Les acteurs principaux dans le système sont les personnels et les visiteurs. Les personnels peuvent réaliser des vérifications d'identité. Le système extrait les données personnelles et vérifie l'identité de chaque visiteur puis enregistre ces informations dans une base de données.

Diagramme de cas d'utilisation « Extraire les données personnelles »



Afin de donner l'accès à un visiteur, le système doit en premier lieu, extraire les données personnelles du visiteur détectées à partir d'un papier officiel. L'application doit extraire le numéro de la carte d'identité, le nom, le prénom et la photo pour permettre au système de vérifier l'identité du visiteur.

Diagramme de cas d'utilisation « Vérifier l'identité »



La phase de vérification de l'identité des visiteurs est une fonctionnalité nécessaire dans le système. Elle débute par réaliser une recherche de l'identité de l'utilisateur dans la base de données pour savoir s'il est un ancien visiteur ou non. Puis, dans le cas où l'identité n'est pas connue, le système réalise une comparaison entre l'image de la personne extraite de la carte d'identité et l'image réelle de l'utilisateur.

5.2 – Diagrammes de séquences

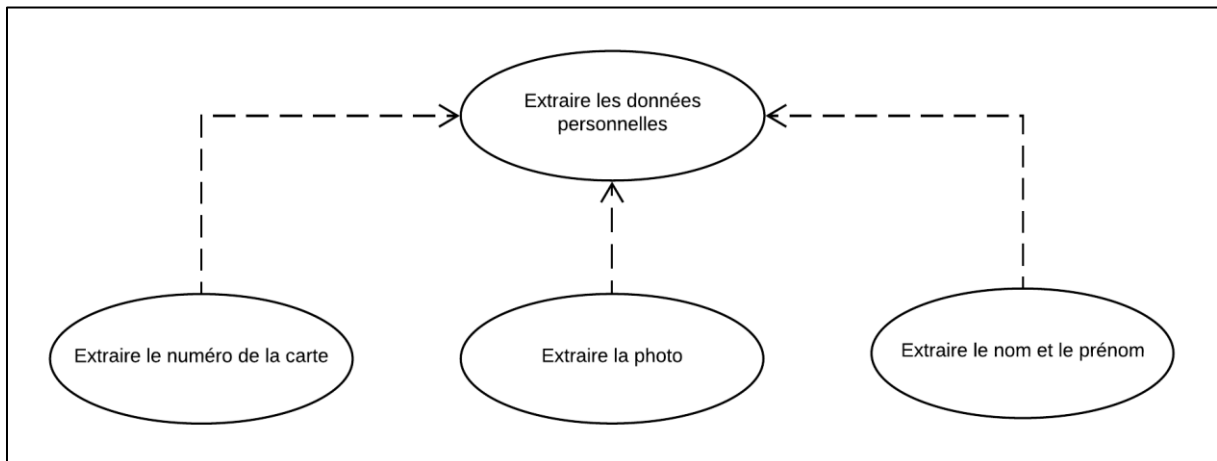
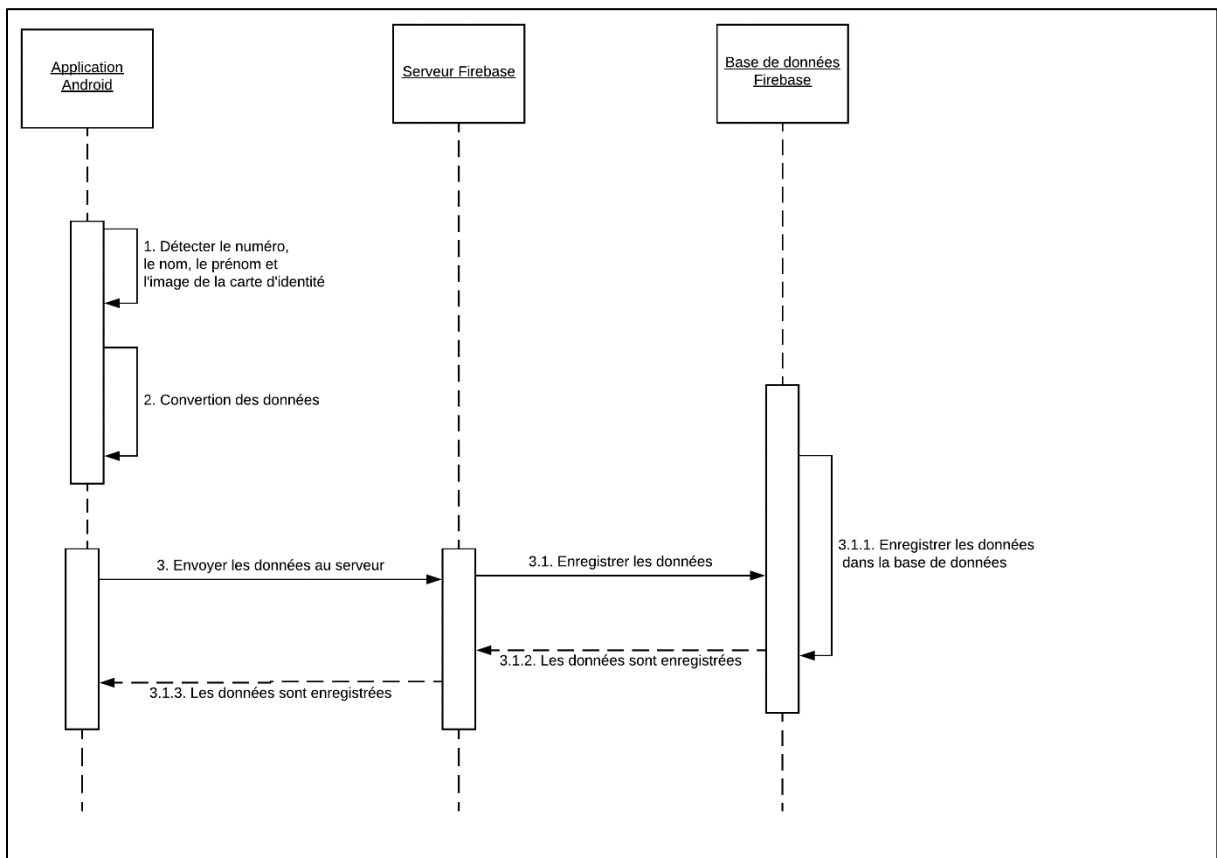


Diagramme de séquence du cas d'utilisation « Extraire les données personnelles »

Ce diagramme de séquence représente l'enchaînement des actions et des interactions entre l'utilisateur de l'application et le système embarqué (Android) pour extraire les données de la carte d'identité.



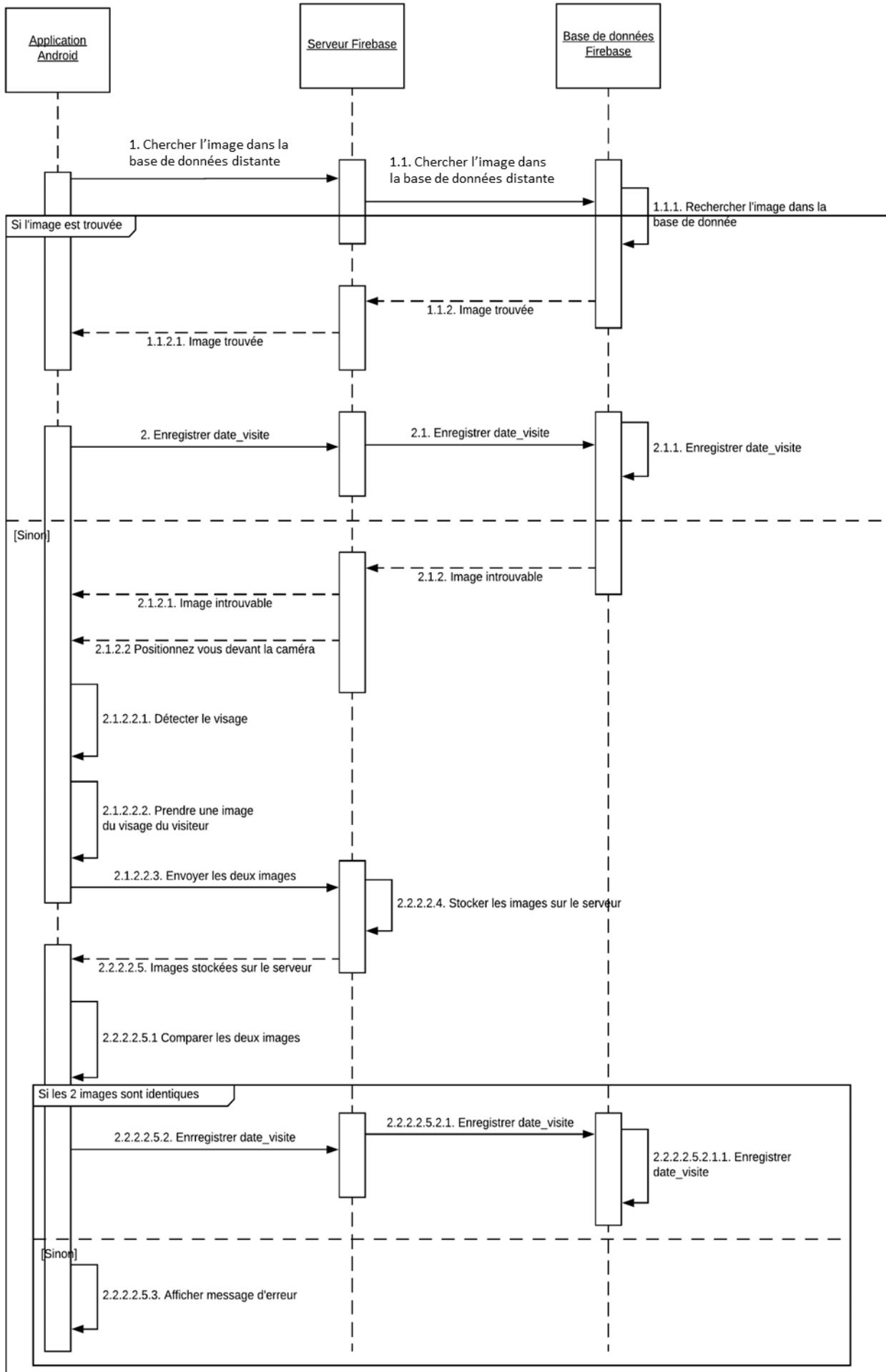
La phase d'extraction des informations personnelles est une fonction qui permet au système de décomposer l'image de la carte d'identité et d'en extraire le numéro, le nom, le prénom et la photo de l'utilisateur.

Ces informations représentent les attributs de chaque utilisateur. Elles construisent la base de données et permettent de gérer les accès des utilisateurs et rendent le système sécurisé.

Diagramme de séquence du cas d'utilisation « Vérifier l'identité »

Le diagramme de séquence « Vérifier l'identité » représente les scénarios possibles en phase de vérification de l'identité de l'utilisateur. L'application Android doit vérifier l'identité à chaque visite. Au début, le système doit effectuer une recherche dans la base de données existante pour savoir s'il est un nouveau ou un ancien visiteur.

Ensuite, si l'identité n'est pas connue, le système doit comparer entre l'image du visage de la personne extraite de la carte d'identité et l'image réel de l'utilisateur. En cas de conformité entre les deux images, le système donne l'accès à l'utilisateur mais, dans l'autre cas, il affiche un message d'erreur.



VI – Besoins spécifiques

L'apprenant devra avoir à disposition :

- Un smartphone Android devant supporter au minimum Android 6.0.1