accenture siapartners sopra steria

LISA

Liberté Sécurité Anonymat

3 May 2020

V0.5

General technical specifications of the protocol used by the StopC19 system

Version History		
Version	Date	Description
0.1	04/04/2020	Creation of the document
0.2	12/04/2020	Protocol Bluetooth added
0.3	16/04/2020	Document review 1
0.4	20/04/2020	Covid test result management added
0.5	03/05/2020	Translated from French using a AI based translator.

Table Of Content

1 Foreword

1.1 Introduction

More than 100 years ago, the "Spanish flu" was one of the worst pandemics the world has ever seen. At that time, all countries were helpless against the spread of the H1N1 epidemic. At the height of the contamination, more than 500 million people were infected, representing more than a third of the world's population. It is estimated that between 50 and 70 million people died from H1N1 in 1918.

Today, the world is faced with a new deadly pandemic of SARS-COV-2, a coronavirus-like virus otherwise known as COVID19. Unlike the "Spanish flu", the world today has a technological maturity that allows nations to combat the spread of epidemics and thus prevent the resurgence of waves of contamination.

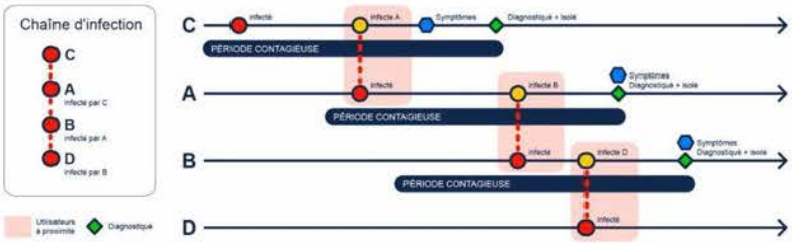
1.2 Why do "proximity tracing"?

Smartphones all have Internet connectivity and most have Bluetooth Low Energy (BLE) enabling the implementation of "proximity tracing" functionality to prevent and control the spread of epidemics such as COVID19.

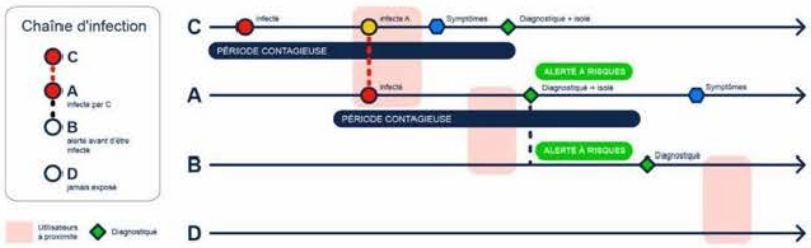
Proximity tracing is a digital system for alerting a user to an interaction with a person who has tested positive for COVID-19 using Bluetooth technology in smartphones. When appropriate, proximity tracing allows users to take the necessary steps to avoid contaminating others.

Diagram illustration:

Without proximity tracing, diagnostics and screening are performed as soon as symptoms appear.



With a proximity tracing system, it is possible to warn a person who has crossed paths with a sick person and thus to anticipate the diagnosis / screening even before the arrival of possible symptoms. The person who would have been contaminated after interaction with the sick person can thus be isolated more quickly. The system thus participates in slowing down or even interrupting the chain of diffusion of the virus.



1.3 Design Principles

The LISA protocol was designed and implemented as part of the development of the global StopC19 system as part of a socially responsible approach by the group of companies Sia Partners, Sopra Steria, Accenture, Orange, Dassault Systèmes, and Cap Gemini.

The aim of this document is to present the LISA protocol in itself and does not constitute a specification of all the components of the StopC19 system.

LISA is based on the following design principles:

- Voluntary use: each user decides in the application of its level of involvement: non activation of proximity tracing, activation of proximity tracing and feedback of proximity interactions only in case of positive screening, activation of proximity tracing and regular feedback of proximity interactions.
- Transparency towards users: the user is informed about the data used and shared by the application.
- Ability to delete all personal data: the user has a simple way in the application to delete all personal data he has entered.
- Limitation of the technological scope linked to the Bluetooth protocol: the service uses exclusively information linked to the Bluetooth protocol, no geolocation or telecom operator data is used.
- Ensuring anonymity and protection of private data: anonymous, decentralised, temporary and rotating identifiers are exchanged ensuring anonymity and strict compliance with personal data protection.
- Provide the greatest reliability in the information communicated: only test results recorded by accredited professionals are taken into account in the identification of high-risk interactions, through a secure process accessible only to healthcare professionals through a strong authentication system. No "self-declaration" is possible.

1.4 Anonymity and protection of personal data

The protection of personal data and user anonymity are at the heart of the founding principles of the LISA protocol.

LISA draws on best practices to ensure these elements with in particular:

- Anonymity and decentralised system: periodic identifiers are generated by the user's smartphone and are exchanged during Bluetooth interactions (see section 4.1) with other terminals. If the user consents (see section 5.1) and these periodic identifiers have expired, they can be sent to the system.
- Anonymity between users: the characterization of proximity interaction information is carried out at regular time intervals without it being possible for a user to identify another COVID-positive user19.

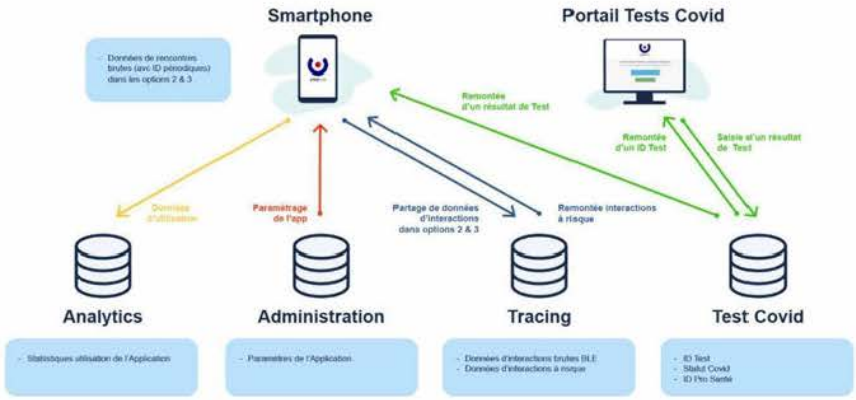
2 General description of the protocol

2.1 Introduction

The global StopC19 system includes:

- A smartphone application dedicated to (10)(2a) citizens
- A web-based application for healthcare professionals authorized to record serological and/or Coronavirus test results
- An administration portal
- Four dedicated and independent Back Ends that allow the management of disjointed databases and guarantee the security and anonymity of data
- Datascience tools

Overview diagram of the overall system:



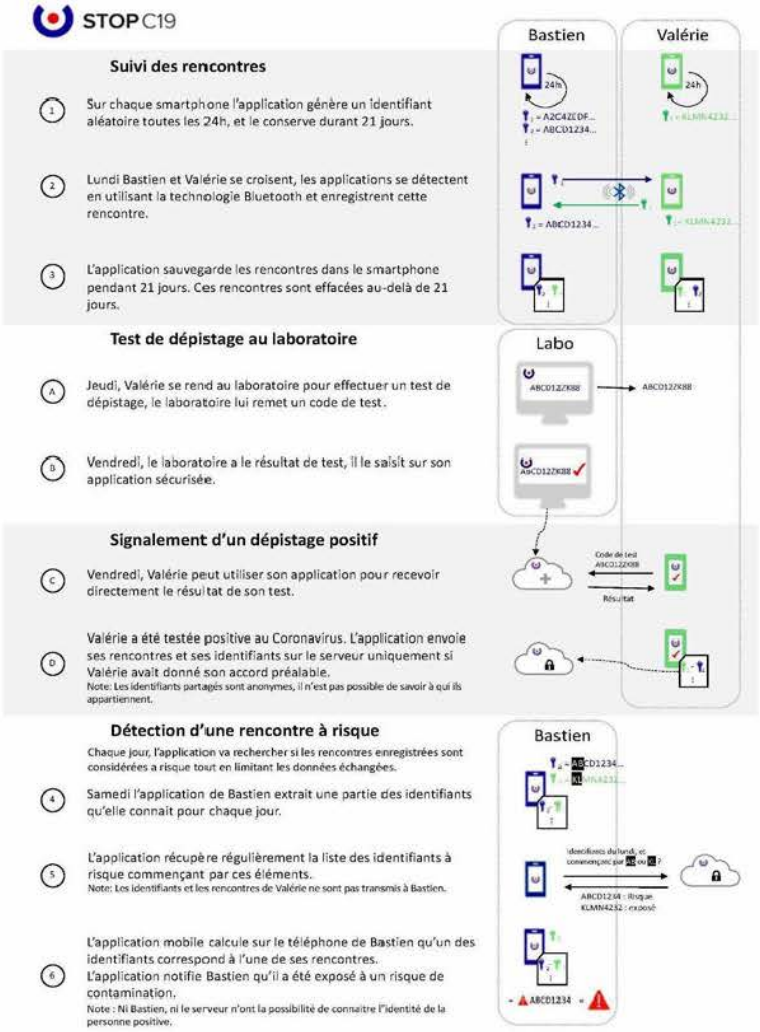
The smartphone is the only receptacle for reconciling data and informing the user according to his own context.

2.2 Convention of terms used in the document

Terms	Definition
Back End Tests Covid	Service platform hosting the capacities of data reception, processing and restitution through REST type services. Platform dedicated to anonymised Covid test data.
Back End Tracing	Service platform hosting the data reception, processing and restitution capacities through REST type services. Platform dedicated to anonymized interaction data.
Back End Administration	Service platform hosting the data reception, processing and restitution capacities through REST type services. Platform dedicated to system administration data (mobile application and exchange settings, usage data).
Personal data	Data entered by the user relating to his identity or that of one of his relatives (title, surname, first name, date and place of birth, addresses), only stored in the phone.
Risk interaction	Recording a close interaction with a person who has tested positive for Covid-19 in the past 14 days.
Proximity Interaction	Recording of a pair of temporary identifiers between 2 smartphones with LISA protocol, when detection via Bluetooth is estimated at less than 4 meters.
Periodic Identifier	Temporary random identifier generated on the user's smartphone, formed by a 128-bit string in hexadecimal.
Covid Test Portal	The Covid Test Portal is the web application for healthcare professionals to create a Coronavirus test identifier and associate a result to it.
Near	Relatives are the various people who are confined to the application user and who do not have a mobile phone.
Covid Status	Serology or screening test results are associated with a field called Covid Status. These test results are retrieved from licensed health care professionals. By default, the status is "undetermined".

Terms	Definition
Low Energy Bluetooth Technology	A radio communication standard (2.4 Ghz) allowing bidirectional data exchange at very short distances (< 10 m). Bluetooth Low Energy (or BLE) uses less energy than conventional Bluetooth.
Serological test	Test carried out by blood sampling to measure the patient's IgM and IgG antibody levels to determine contamination or immunization.
Screening Test	PCR swab screening test (to determine whether or not Covid-19 is present in the patient's airway)

2.3 Global kinematics



2.4 Processes handled

The following business processes are described in the LISA protocol:

- First-time installation
- Proximity Interaction
- Sharing interaction data
- Characterization of risk interactions
- Reconciliation of information
- Coronavirus testing process

For each business process, a description of the business rules and detailed technical specifications are given below.

3 First-Time Installation

3.1 Functional Description

At the first launch of the application, no personal data is required to create an account. The user must simply accept the general terms of use and validate via a "Captcha" mechanism that he is not a robot.

An access token is then created and will be used in the user's next actions from the application in order to secure the exchanges between the smartphone and the different Back Ends.

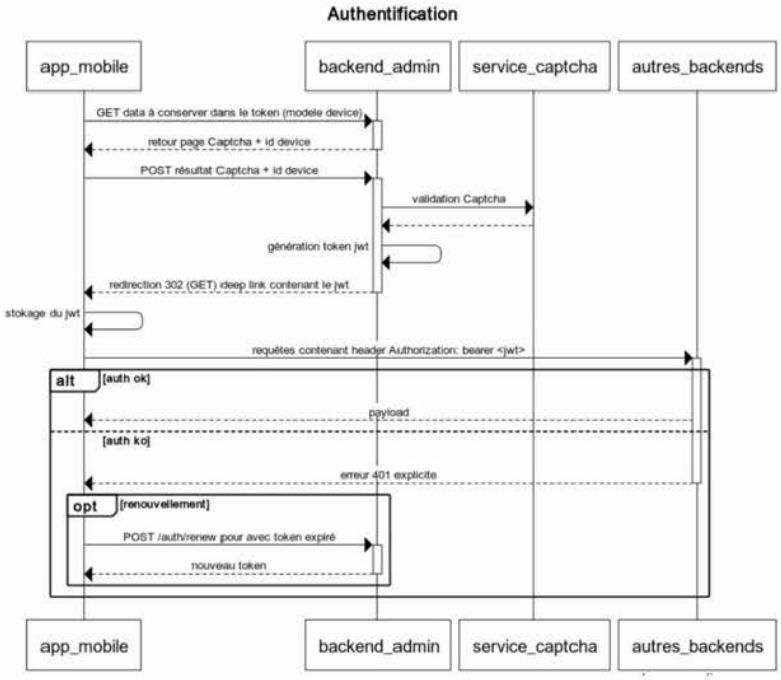
From the first launch of the application, the user is invited to choose his level of involvement in order to:

- Authorize or not to authorize the collection and storage of his proximity interaction data on his smartphone (interaction tracking functionality) ;
- Choose when he wants to share this interaction data: all the time or only if he is tested positive.
- If he consents to the collection of his interaction data, the application will ask him:
    - To activate Bluetooth in order to allow the detection of nearby terminals with the LISA protocol.
- Activate notifications in order to be notified by the application:
    - In the event that he has come across a person testing positive for Coronavirus in the last 21 days,
    - To invite him to activate his interaction tracking.

The user can at any time choose to change his level of involvement by going to the dedicated menu "Interactions / Review my choice of involvement".

3.2 Technical Specification

Diagram of the kinematics:



An access token is retrieved by the application, provided it has successfully validated a "Captcha" mechanism. The purpose of this mechanism is to confirm that the user requesting the access token is indeed a human being, and not a robot. This is implemented for the purpose of mitigating cyber-terrorist attack scenarios. The solution currently used is ReCaptcha V2 from Google, it can be replaced by another solution providing this service on demand.

The recovered access token must be communicated in the headers of all requests to web services in the backend, otherwise the requests will be rejected.

The access token has a finite validity period. Once the expiration date is reached, this token must be renewed via the FT_AUTH_RENEW technical stream. Again, enforcing this renewal is a security mechanism. Indeed, an access token cannot be revoked, so it will always be accepted over its entire range of validity. Limiting its lifetime, and forcing the renewal of the token, allows the backend to refuse the renewal of an access token considered to have questionable behavior.

In order to counter an attack on FT_AUTH_RENEW to generate multiple tokens from a single initial token, each token can only be renewed once, once its expiration date has been reached.

Description of the FT_AUTH_RENEW stream:

Nom du flux :	FT_AUTH_RENEW		
Description	Ce flux permet de renouveler un jeton d'accès.		
Entrées	Valeur		
Jeton d'accès	*JWT*		
Sorties	Valeur	Persistance	Description
Jeton d'accès	*JWT*	En base de données chiffrée	Nouveau jeton d'accès

–

4 Proximity Interactions

4.1 Functional Description

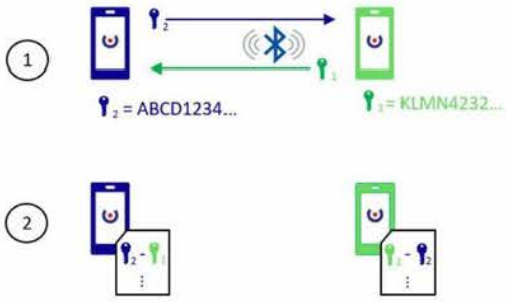
Bluetooth Low Energy technology is used to detect users of the application who have agreed to enable tracking functionality for their interactions.

This feature allows the user's interactions to be recorded on his smartphone, i.e. to detect other users of the application that he has come across while on the move.

As soon as Bluetooth is activated on the smartphone on the one hand, and the tracking functionality in the application on the other hand, the smartphone locally generates a random identifier at a regular and configurable frequency. This periodic identifier guarantees user anonymity.

When the user moves around, if tracking is manually activated by the user in the application, the application locally records each interaction with another user of the application as well as the parameters of the interaction that will identify if the interaction is at risk.

Illustration of detection between two smartphones:



In the case of a meeting between two people A and B, both of whom have activated the Bluetooth functionalities and tracking interactions :

- A's smartphone generates a random periodic identifier
- B's smartphone generates a random periodic identifier
- The smartphones are detected via the StopC19 application and each one records the pair of random identifiers corresponding to the encounter.

Only so-called proximity interactions are recorded in the smartphone, i.e. interactions where the distance is estimated to be less than 4 meters. The data in the Bluetooth signal is filtered in this way. This detection can be asymmetrical (A having detected B, the reciprocal not being true), this is not a problem because the asymmetry of a detection can be compensated for later on in the protocol.

Each smartphone stores the history of interactions over a period of 21 days (configurable period).

No personal data (name, number, identifier...) is associated to the smartphone or recorded during an interaction.

4.2 Technical Specification

At each meeting, the application stores locally a tuple with the following form:

interaction=(〖id〗 _local , 〖id〗 _meet ,time stamp ,txPower ,RSSI ,distance ,duration)

$$interaction = (\ id_{local}\ , id_{rencontré}\ , horodatage\ , txPower\ , RSSI\ , distance\ , durée\ )$$

With:

- 〖id〗 _local: periodic identifier used by the application for broadcasting BLE
- 〖id〗 _(encountered): periodic identifier detected when interacting with another smartphone
- timestamp: date and time of the meeting
- txPower: transmission power of the device detected, in dBm
- RSSI: measurement of the received power of the detected device signal, in dBm
- distance: estimated distance calculated using txPower and RSSI, in metres
- duration: exposure time of the meeting, if available, in seconds

Periodic identifiers are UUIDs generated according to RFC4122.

This detection is first performed via a BLE broadcasting mechanism: each smartphone will regularly transmit BLE frames containing a manufacturer identifier, a service number specific to the application. These two identifiers are used to discriminate and only consider the Bluetooth signals emitted by our application.

The rest of the process depends on the platform:

- On Android, BLE frames also contain the periodic identifier
- On iOS, when an Android smartphone is detected, it records the interaction, and the iOS smartphone connects to the Android to write its periodic identifier into a feature.
- On iOS, when an iOS smartphone is detected, a point-to-point connection is made to read the ID of the opposite smartphone.

The distance d is calculated based on the following formula:

$$EffectiveTxPower = \begin{cases} si\ txPower = null\ et\ hintAndroid = true, 11 \\ si\ txPower = null\ et\ hintAndroid = false, 12 \\ si\ txPower < 0, \quad txPower + 20 \\ sinon, txPower \end{cases}$$

avec $hintAndroid = true$ si le téléphone distant est un Android.

$$P_{Android} = \begin{cases} si\ EffectiveTxPower \in [9,12[, -71 \\ si\ EffectiveTxPower \in [12,20[, -67 \\ sinon, -86 \end{cases}$$

$$P_{iOS} = \begin{cases} si\ EffectiveTxPower \in [9,12[, -71 \\ si\ EffectiveTxPower \in [12,20[, -57 \\ sinon, -86 \end{cases}$$

$$d = \begin{cases} si\ RSSI \geq 20.0, -1 \\ sinon, \quad 10^{\frac{(P-RSSI)}{10n}}\ avec\ n = 2 \end{cases}$$

The different parameters are the result of experiments on the same subject and of a calibration campaign and are subject to change (and can easily be changed)

The application will also characterize the meetings by "session": the interaction data are aggregated by the application up to a maximum session duration of 5 minutes (configurable duration). In this context of aggregation, the RSSI value used is the median of all the RSSIs scanned. It is this median that is used to calculate the distance and thus exclude short encounters from the stored data. For example, an individual at a distance of 10 meters during a public transit trip passing close to 1 meter for 2 seconds to get off the bus or railcar will not be retained.

4.3 Current Technical Limitations

The current solution has limitations induced by the Android and iOS platforms, as do all detection solutions based on BLE technology.

Thus, on iOS terminals, when the application is not in the foreground (in the case of the terminal going into sleep mode or using another application in the foreground), the smartphone's operating system limits the reactivity of BLE detection:

- The detection is not a priority for the system, the detection delay may be longer (from a few seconds to a few minutes).
- The system limits the detection of a device that will only be detected once at most. In this framework, it is not possible to estimate an encounter time, so we return the value 0 for this encounter time to let data science algorithms interpret this result.
- The ability of the protocol to process mutual detections by different devices (asymmetric encounter) allows us to limit the impact of this iOS processing. It will be necessary to evaluate the use and mastery of the solution currently being developed by Apple, subject to compliance with the structuring principles of the protocol (cf. paragraph 1.3).

On Android terminals, and particularly on "entry-level" devices, the detection reactivity and therefore the estimation of the encounter time may be less accurate. This impact is limited by the choice of the minimum version of Android supported (API 21, i.e. Android 5.0), which allows the platform's BLE APIs to be used without restriction, and by our ability to support asymmetric encounters.

It should also be noted that the Android system requires geolocation permission in order to enable Bluetooth background scanning on many devices. This permission is only required to be able to run in the background, at no time is the position of the smartphone used.

5 Sharing interaction data

5.1 Functional Description

When installing the application, then at any time in a dedicated tab "Interaction / Review my choice of involvement", the user can choose his level of involvement.

3 levels of involvement are proposed:

5.1.1 Option 1: "If I am detected"

When the user chooses this option, their proximity interaction data is collected as they go and stored, only in their smartphone, for a configurable period of 21 days.

When the user performs a screening test, he or she can view his or her test result on his or her smartphone (using an anonymous test ID provided by a healthcare professional - see Chapter 8). In case the user has tested positive for Covid-19, he is then informed, at the time of consulting his test result, that his interaction data of the last 21 days (which were previously stored only in his smartphone) will be transmitted to a database dedicated to anonymized interaction data - Back End Tracing.

From the day the user is tested positive, his smartphone will continue to transmit interaction data to Back End Tracing on a daily basis for a configurable period of 14 days after the date of his positive test (this period corresponds to the known contagion period to date).

5.1.2 Option 2: "All the time

When the user chooses this option, their proximity interaction data is collected as they go and stored in their smartphone for 21 days (duration can be configured).

Once a day, when the phone has changed its periodic identifier, the interaction data is transmitted to Back End Tracing. As long as the user has not tested positive to Covid-19, these interaction data are used only to train the algorithms and improve the risk characterization of an encounter.

5.1.3 Option 3: "Not at this time".

When the user chooses this option, no interaction data is collected by the application and the application does not broadcast its periodic identifier via Bluetooth. The user is therefore excluded from the protocol.

For the user choosing options 1 or 2, the local collection of his interaction data allows to offer him an additional feature: the monitoring of his "social distancing index".

5.1.4 Social distancing index

This index presents the number of interactions captured by the Bluetooth of your smartphone over the last 24 hours. This social distancing index presents 2 levels of the real-time evolution of its activity:

- Less than 5 interactions recorded in the last 24 hours.
- More than 5 interactions recorded in the last 24 hours

This index also reports at least one high-risk interaction in the last 21 days with a person who tests positive for Covid-19 (regardless of the number of interactions in the last 24 hours) and for the next 14 days. This alert is not issued in real time.

5.2 Technical Specification

5.2.1 Data manipulated

The interaction data transmitted via the FT_TESTED_ENCOUNTERS stream in case of a positive Covid-19 test ("Option 1" or "Option 2") are the following:

〖interactions〗_1=(〖interaction〗_(j-1...j-21), 〖modèleAppareil〗_local, 〖tempsAppareil〗_local ,secretSanté)

The interaction data transmitted via the FT_UNTESTED_ENCOUNTERS stream in the case of "Option 2" is as follows:

〖interactions〗_2=(〖interaction〗_(j-1...j-21), 〖modèleAppareil〗_local, 〖tempsAppareil〗_local)

With:

- 〖interaction〗_(j-1...j-21): set of interaction tuples (defined in paragraph 4.2) stored over the last 21 days, except those containing the periodic id in use and those already transmitted.
- 〖modèleAppareil〗_local: technical identifier of the Android or iOS device type, given by the manufacturer (e.g. SM-G920F for a Samsung Galaxy S6, iPhone10,6 for an Apple iPhone X)
- 〖tempsAppareil〗_local: time of the device at the time of transmission of information
- secretHealth: security key produced by the Back End Tests Covid and certifying a status "tested positive for Covid-19", present only if the user has chosen to transmit his data only if he is screened

5.2.2 Description of the flows

Back End Tracing is a database containing only interaction data information from users' smartphones. It is composed of 2 distinct tables:

- "Untested Encounters'': in this table are stored the interaction data reported daily by users who have opted for choice 2 "All the time".
- "Tested Encounters": in this table are stored the interaction data reported by users who tested positive to Covid-19, on a one-off basis (then daily for 14 days).

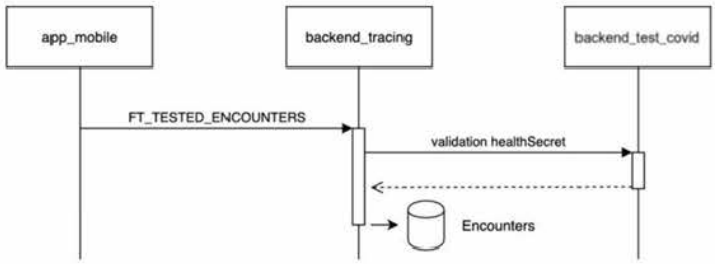
The interaction data traced back to the Back End Tracing are completely anonymized (not pseudonymized) because:

- These data do not contain any location data.
- The periodic identifier of the communicated telephone is no longer used by the telephone to transmit at that time: the interception of the data transmission flow to Back End Tracing cannot therefore make it possible to find the telephone at the origin of the data transmission.
- The different temporary identifiers of the same phone are transmitted in separate files so that it is impossible to link 2 different temporary identifiers as corresponding to the same smartphone.

5.2.3 Flow FT_TESTED_ENCOUNTERS

Nom du flux :	FT_TESTED_ENCOUNTERS		
Description	This stream allows, if the user is diagnosed positive, to transmit all interactions stored over the last 21 days except those with the periodic id currently in use and those already transmitted via this stream.		
Entrées	Valeur		
Jeton d'accès	JWT		
encounters	$interaction_{j-1\ldots j-21}$		
deviceModel	$modèleAppareil_{local}$		
currentDeviceTime	$tempsAppareil_{local}$		
healthSecret	$secretSanté$		
Sorties	Valeur	Persistance	Description
Aucune			

Diagram:

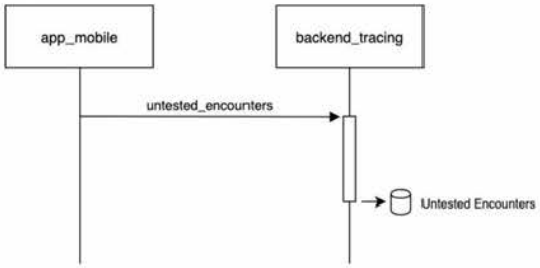


5.2.3.1 Flow FT_UNTESTED_ENCOUNTERS

Nom du flux :	FT_UNTESTED_ENCOUNTERS

Description	This flow allows the application to transmit all the interactions stored over the last 21 days except those with the periodic Id currently in use and those already uploaded via this flow.		
Entrées	Valeur		
Jeton d'accès	JWT		
encounters	$interaction_{j-1\_j-21}$		
deviceModel	$modèleAppareil_{local}$		
currentDeviceTime	$tempsAppareil_{local}$		
Sorties	Valeur	Persistance	Description
Aucune			

Diagram:



5.2.4 Call logic

In the case of "Option 1" (sharing if tested positive for Covid-19),

- The FT_TESTED_ENCOUNTERS flow is called after a positive test result. The HealthSecret is then checked to validate that the person tested positive, and all the encounters are stored in the Back End Tracing "Encounters" table.
- The FT_TESTED_ENCOUNTERS flow is recalled every day for the 14 days following the test.

In the case of "Option 2" (sharing interaction data as it happens),

- The FT_UNTESTED_ENCOUNTERS stream is called every day before a positive test result. All encounters are stored in the Untested Encounters table of the Back End Tracing.
- The FT_TESTED_ENCOUNTERS stream is called if a "detected positive" test result is retrieved from the smartphone (according to the process described in Chapter 8). It is then checked to validate that the person tested positive, and the set of encounters is stored in the Back End Tracing "Encounters" table.
- The FT_TESTED_ENCOUNTERS flow is recalled every day for the 14 days following the test.

6 Characterization of risky interactions

6.1 Functional Description

The characterization of interactions at risk is a processing performed on the server from raw interaction data transmitted by the smartphone and stored on the Encounters table. An encounter between two smartphones results in the feedback of multiple short and successive interactions.

The objective of the processing is to aggregate the raw interactions concerning a pair of periodic IDs in a limited time into a single interaction (= aggregated interaction).

Once aggregated, the risk level of the interaction will be characterized.

An aggregated interaction is considered "at risk" if it meets the following two cumulative conditions:

- The estimated distance between the two smartphones is less than 2 meters.
- The estimated duration* of the interaction is greater than 15 minutes (This condition does not apply to iOS terminals potentially going back a duration of 0 due to current platform limitations)

*The estimated interaction time between two smartphones A and B is the cumulative duration of the raw interactions before aggregation.

In addition to the characterization on distance and duration, a risk factor is associated with the interaction. In particular, if the user is screened positive according to the process provided in the application (see Chapter 8), interactions associated with the status "screened positive" will be considered at risk. They are then stored on the server's separate table named "Risky Encounters".

Following the acquisition of data on the server, statistical analyses will allow to refine the characterization thresholds of the interactions defined above from the interaction data of the "Encounters" and "Untested_encounters" tables.

This processing is triggered 24 hours after the interaction.

6.2 Technical Specification

From the "Encounters" table and the following examples of interactions:

{ ⟦idPériodique⟧ _A, ⟦IdPériodique⟧ _B,t_1,distance,duration,RSSI,txPower, ⟦device⟧ _A, ⟦ statutCovid ⟧ _A},

{ ⟦idPériodique⟧ _A, ⟦IdPériodique⟧ _B,t_2,distance,duration,RSSI,txPower, ⟦device⟧ _A, ⟦ statutCovid ⟧ _A},

{ ⟦idPériodique⟧ _A, ⟦IdPériodique⟧ _B,t_3,distance,duration,RSSI,txPower, ⟦device⟧ _A, ⟦ statutCovid ⟧ _A },

{ ⟦idPériodique⟧ _A, ⟦IdPériodique⟧ _B,t_4,distance,duration,RSSI,txPower, ⟦device⟧ _A, ⟦ statutCovid ⟧ _A }

For t_actual-48 hours < t_i>t_actual-24 hours:

Meet caractérisée→

{ ⟦IdPériodique⟧ _(A,) ⟦IdPériodique⟧ _B,t_1,min(distance),∑_?(distance@<2 metres)▒durée,median(RSSI),txPower, ⟦statutCovid ⟧ _A}

For each meeting caractérisée:

if ∑_?(distance@<2 meters)▒durée>time threshold

meets caractérisée→rencontre at risk { ⟦idPériodique⟧ _A, ⟦IdPériodique⟧ _B, aRiskFactor= indexRiskCovid,bRiskFactor= indexRiskCovidPair,updatedAt=t_actual}

6.2.1 Original French Text:

$\{ idPériodique_A, IdPériodique_B, t_1, distance, durée, RSSI, txPower, device_A, statutCovid_A \}$,

$\{ idPériodique_A, IdPériodique_B, t_2, distance, durée, RSSI, txPower, device_A, statutCovid_A \}$,

$\{ idPériodique_A, IdPériodique_B, t_3, distance, durée, RSSI, txPower, device_A, statutCovid_A \}$,

$\{ idPériodique_A, IdPériodique_B, t_4, distance, durée, RSSI, txPower, device_A, statutCovid_A \}$

$Pour\ t_{actuel} - 48\ heures\ <\ t_i > t_{actuel} - 24\ heures:$

$Rencontre\ caractérisée \rightarrow$

$\{ IdPériodique_A, IdPériodique_B, t_1, min(distance), \sum_{\substack{distance \\ <2\ mètres}} durée, median(RSSI), txPower, statutCovid_A \}$

Pour chaque rencontre caractérisée :

$si \sum_{\substack{distance \\ <2\ mètres}} durée > seuil\ durée$

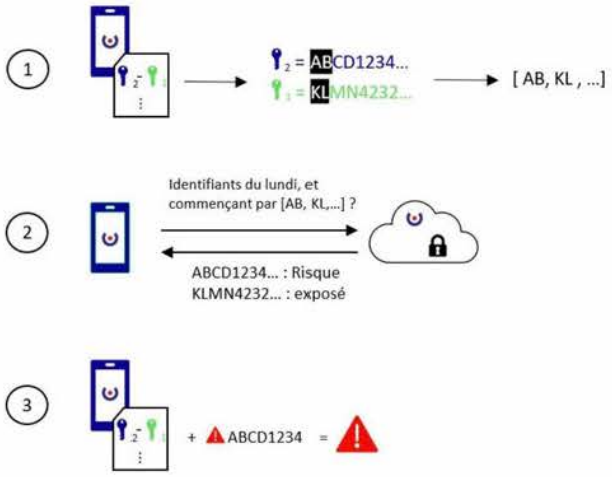
$rencontre\ caractérisée \rightarrow rencontre\ à\ risque\ \{idPériodique_A, IdPériodique_B, aRiskFactor = indiceRisqueCovid, bRiskFactor = indiceRisqueCovidPaire, updatedAt = t_{actuel}\}$

7 Reconciliation of risk interactions

7.1 Functional Description

The interactions that are risky for the user are determined by the application locally on his smartphone. To do this, the application queries the server (Back End Tracing) to retrieve a subset of periodic identifiers characterized as risky. This query contains a list of identifier prefixes, which is the union of the identifier prefixes used by the application for the BLE broadcasting of the smartphone (local identifiers) and the identifier prefixes encountered (generated by other smartphones and stored during proximity interactions). Mixing both local identifiers and encountered identifiers, and using only identifier prefixes, ensures the anonymity of the request while limiting the volume of data processed.

Illustration of reconciliation by the smartphone:



The server returns a list of couples (idPeriodic, status), the status making it possible to discriminate between directly contaminated identifiers and those that have been in contact with a contaminated person.

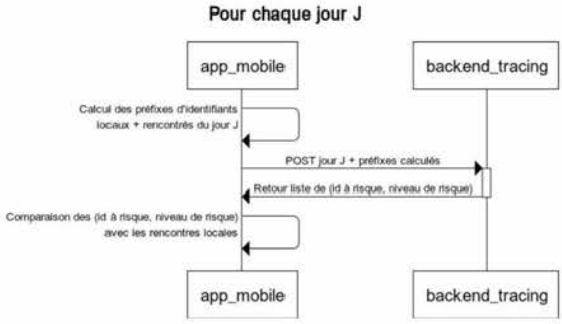
The application will then compare the identifiers it has locally (both its own identifiers, and those it may have encountered) with those returned by the server, and will be able to detect if the user has been exposed to a risk.

Three cases are possible:

- No local identifiers match those returned by the server (from the "Untested Encounters" table without risky interaction).
- At least one identifier encountered is returned with a "at risk" status by the server (from the "Risky Encounters" table of interactions).
- At least one identifier generated by the application is returned with a "at risk" status by the server (from the "Risky Encounters" table of interactions).

7.2 Technical specification

Diagram of the kinematics:



In the context of the reconciliation of interactions at risk for the user, the tuple interaction data (defined in paragraph 4.2) used are 〖id〗_local, 〖id〗_encountered, and timestamp.

Consider the following:

- D: today's date
- H: the number of days of interaction data historization
- C : the number of prefix characters (in hexadecimal representation, 2 characters are therefore equivalent to 8 bits)
- 〖IdLocaux〗_J: set of local identifiers used on D-Day
- 〖IdRencontrés〗_J: set of identifiers encountered on D-Day
- 〖PréfixesIdLocaux〗_J: set of prefixes of 〖IdLocaux〗_J, a prefix being the first C characters of a local identifier
- 〖PréfixesIdRencontrés〗_J: set of prefixes for 〖IdRencontrés〗_J

For each day J∈{ D-i,i ∈{0,1,2,…,H-1} }, the application will perform the following process:

Generate the following sets:

- ⟦IdLocaux⟧ _J
- ⟦IdRencontrés⟧ _J
- ⟦PréfixesIdLocaux⟧ _J
- ⟦PréfixesIdRencontrés⟧ _J

Call up the technical flow FT_RISKY_ENCOUNTERS described below

Reconcile the identifiers received with those of ⟦IdLocaux⟧ _J and ⟦IdRencontrés⟧ _J

Generation of sets

The sets:

⟦IdLocaux⟧ _J

⟦IdRencontrés⟧ _J

⟦PréfixesIdLocaux⟧ _J

⟦PréfixesIdRencontrés⟧ _J

Are generated from the captured interaction data (see section 4.2).

7.2.1 Calling the FT_RISKY_ENCOUNTERS stream

Nom du flux:	FT_RISKY_ENCOUNTERS
Description	This feed returns the set of D-day identifiers whose prefix is included in the requested identifier prefixes, and whose characterization as an at-risk identifier occurred after the bookmark.
Entrées	Valeur
Jeton d'accès	JWT
Jour	J
Préfixes d'identifiants	$PréfixesIdLocaux_J \cup PréfixesIdRencontrés_J$
Bookmark	Bookmark, ou rien si non existant

Sorties	Valeur	Persistance	Description
Identifiants à risque	$\{(id, niveauRisque)\}$	Aucune	Set of tuples with: - id: periodic identifier - Risk level: a level of risk (5 or 9) 9 corresponds to an identifier at risk. 5 corresponds to an identifier that has been in contact with a risky identifier.

The use of prefixes and paging per day allows the generation of immutable pages of server-side characterized identifiers, which are thus simple to cache and ensure that the system can handle the load given the sizing assumptions.

The bookmark mechanism is used to minimize the volume of data transmitted to smartphones, as well as the amount of comparisons to be made during each reconciliation pass.

7.2.2 Reconciliation of received identifiers with stored identifiers

The application will apply the following algorithm:

Pour chaque tuple $t \in \{(id, niveauRisque)\}$

Si $niveauRisque_t = 9\ ET\ id_t \in IdRencontrés_j$ alors
Interaction à risque
Fin Si
Si $niveauRisque_t = 5\ ET\ id_t \in IdLocaux_j$ alors
Interaction à risque
Fin Si

Fin Pour

This search on both ⟦IdLocaux⟧ _J and ⟦IdRencontrés⟧ _J allows to manage cases of asymmetric detection between smartphones (i.e. when only one of the 2 smartphones managed to capture an interaction).

8 Coronavirus testing process

8.1 Functional Description

Illustration of the process:

8.1.1 Provision of a secure Portal for Covid tests

The certification of a person contaminated with Covid-19 is obtained through test results recorded by a licensed health care professional.

Thus, a "Covid Tests" Portal is provided for health professionals designated to carry out serological and screening tests (PCR type) of Covid-19 (in particular biological analysis laboratories). The portal is accessible via a computer or mobile phone from a web browser.

Connection to the Covid Testing Portal is secured by the ProSanteConnect authentication mechanism. Thus any holder of a CPS, CPE or CPA card can connect to the Covid Test Portal using the reader box of his card (or the digital version of his e-CPS card). This connection via ProSanteConnect also allows the identification of the connected person.

This portal allows you to:

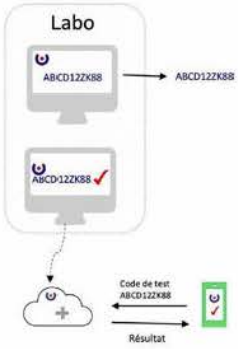
- Generate random and anonymized identification keys;
- Enter the results of tests performed in a simple interface.

A dedicated database, the Covid Back End Tests, stores only the results of each test:

- the generated test ID
- the date of generation of this ID
- the identifier of the health professional or health care facility that generated the ID

- the result associated with that Test ID (4 possible results: Screened Positive / Screened Negative / Immune / Not Immune)
- the date of entry of the result in the Covid Test Portal
- the identifier of the health professional or health facility that entered this result
- the date on which the result was sent back to the user's smartphone after being queried by the smartphone (see below)

Test Result Recovery Process Illustration:



The patient goes to the facility where he or she is to be tested. The staff at the facility will provide a test ID generated via the Covid Test Portal. This key is unique and anonymous.

Once the test is performed, the health professional who obtains the test result enters the result into the Covid Test Portal.

The use of this result is subject to the consent and actions of the user as follows:

- The user goes to the application in the menu "My Coronavirus Tests".
- To add a test, he is asked to enter a personal secret code that he chooses and that will be asked again at each subsequent consultation of his results in order to encrypt his results.
- The user then enters his or her identification key (Test ID) transmitted by the health care institution in which he or she performed the test.
- It indicates whether the test was done for him or for someone close to him...
- His smartphone will then query the Covid Back End Tests to verify that such a Test ID has been generated and if a result has been assigned to it:
    - If no result has been recorded yet, a message is displayed and no data is shared.
    - If a result is saved, the result is displayed.

- If the result indicates a positive screening at the Covid-19 of the user, and according to the level of implication chosen by the user, the consultation of this result can trigger a sharing of the interaction data of the last 21 days (duration can be configured).
    - If the user has chosen the consent option "If I am screened", his consent is reminded and his interaction data of the last 21 days are transmitted to Back End Tracing with his status "screened positive". His data will continue to be transmitted for 14 days (see Paragraph 5. Sharing of interaction data).
    - If he has chosen the "All the time" consent option, he is reminded of his consent and his interaction data from the last 21 days is transmitted to Back End Tracing with his "screened positive" status (see Paragraph 5. Sharing of interaction data) via the same test ID.
    - If he or she has chosen the "Not immediately" consent option, viewing the test result has no effect.

Once the user has viewed the test result once, he or she can return to view the result in his or her "Test History". The personal PIN code he or she has chosen will be asked again on this occasion (instead of the test ID). The test result is stored in the user's smartphone and can be viewed by the user for 14 days from the test date.

Only by entering the identification key can the test result be retrieved, guaranteeing both anonymity and the user's freedom of choice.

Once the key is saved, each time the application is opened the smartphone queries the Covid Back End Tests to retrieve the test result, until the result is saved in the database. The encoded result is then stored on the smartphone for 14 days.

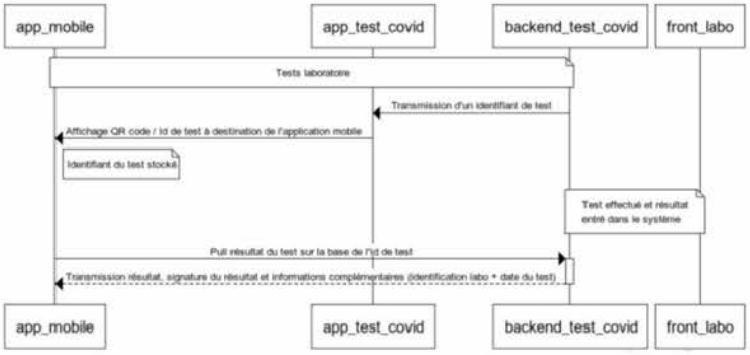
8.2 Technical specification

8.2.1 Authentication of the professional on the Covid Tests portal

Authentication of the healthcare professional who wishes to connect to the Covid Test Portal is done via ProSanteConnect, the authentication system developed by ASIP.

8.2.2 Kinematics of exchanges concerning the test result

Diagram of the kinematics:



8.2.3 Data exchanged

Data	Description	Treatment	Format	Storage
ID Test	Unique identification key associated with each test	Generated by the health professional in the portal	10 alphanumeric characters	Stored in the Covid Back End Tests
Date of test	Date of generation of	Stored automatically at the time of ID generation	Date	Stored in the Covid Back End Tests

	the test identifier			
Id Pro Health ID Test	Identifier of the health professional generating the test identifier	Automatically registered at the time of authentication of the health professional	Text	Stored in the Covid Back End Tests
Status COVID	Field associated with the test result	Entered by the health professional in the portal in the list of the 4 possible values (configurable): Immunized, Non-immunized, Screened positive, Screened negative	List of values	Stored in the Covid Back End Tests Stored in the smartphone for 14 days from test date
Date entered COVID status	Date of test result entry	Automatically recorded when the test result is validated by the health professional	Date	Stored in the Covid Back End Tests
Id Pro Health Status COVID	Identifier of the health professional entering the test result	Automatically registered at the time of authentication of the health professional	Text	Stored in the Covid Back End Tests
Description status COVID	Explanatory wording associated with the test result	Set up in the back office	Text	Stored in the Covid Back End Tests Stored in the smartphone for 14 days from test date
Result Updated Date	Date of retrieval of the test result by the smartphone	Recorded automatically when the smartphone comes to query the Back End	Date	Stored in the Covid Back End Tests

The mobile application will regularly query the backend_test_covid (polling) in order to retrieve a test result. Two cases are implemented in the application:

- I consult the test result of a relative: in this case and in case of a positive test, consulting the test result will not trigger the creation of a health secret, and therefore will not trigger the sharing of interaction data.
- I consult the result of one of my tests: in this case and in case of a positive test, a health secret will be generated.

The backend_test_covid is responsible for generating the health secret. In order to reinforce the overall security of the system, the generation of a HealthSecret implements the following constraints:

- A healthsecret is generated only once for a given test. Once the results are visualized, the flow will not return any more HealthSecret.
- The generated HealthSecret contains the "sub" field from the access token used by the smartphone (see 3.2) during the first installation of the application.

This second constraint is used to validate on the backend_tracing side, when sharing interaction data, that the smartphone that emits the data is the same smartphone that retrieved the Covid test result.

As health secrecy is linked to a smartphone, and the access token has very tightly controlled renewal and lifetime constraints, information sharing linked to a health secrecy and a given access token allows to detect potentially questionable behaviours.