NHSx

NHS COVID App

Application and System Architecture

May 2020

Application Purpose

Stop the virus spreading

Save lives

Allow people to know when they can end isolation

When I download the app, it keeps an anonymous record of when I've been close to others (proximity events)

If I self-diagnose as a carrier in the app I can choose to upload my personal record of proximity events to a backend

The backend can work out who to message and let them know they have been in contact and provide the latest advice

Analysis of uploaded records of proximity events will allow the NHS to monitor and control the spread of the virus

Preserving the privacy of users is high priority - personal information is kept to a minimum unless entered by the user

Licensing & Usage

This document describes the Bluetooth proximity contact tracing application built by VMware Pivotal Labs for the UK Government.

Contents

Introduction

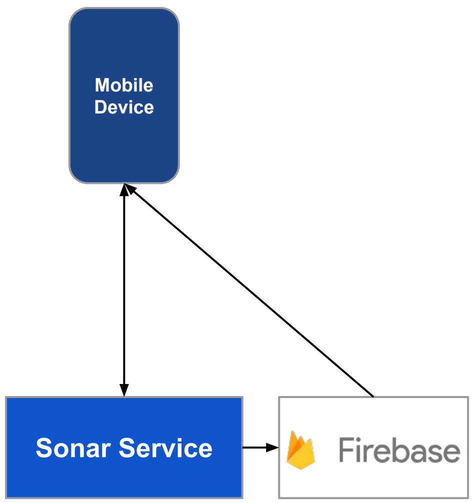
Introduction

This deck is intended to be a reference to the current state of the architecture

For details of the architectural choices made please refer to the discussion documents and associated which are linked to throughout

Overall Flow

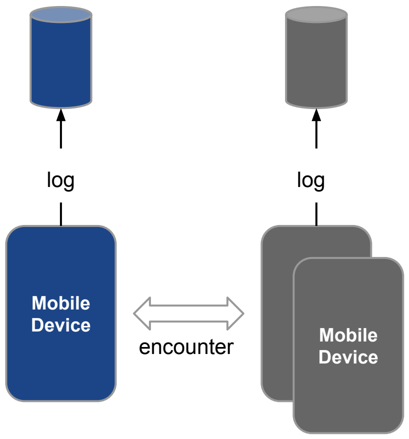
REGISTER



1. Device registers with service

IMPLEMENTED

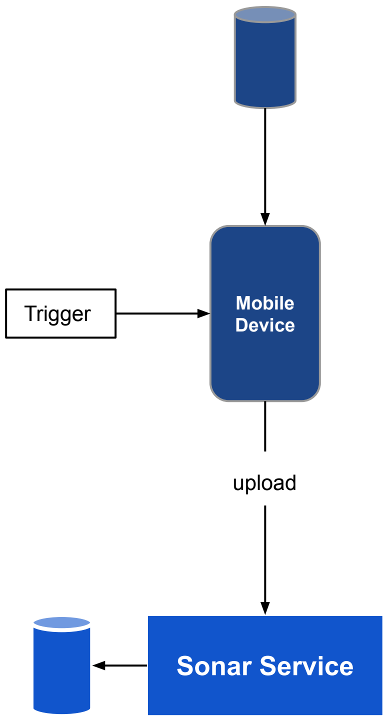
STORE



2. Device encounters other devices and stores a record of each encounter

IMPLEMENTED

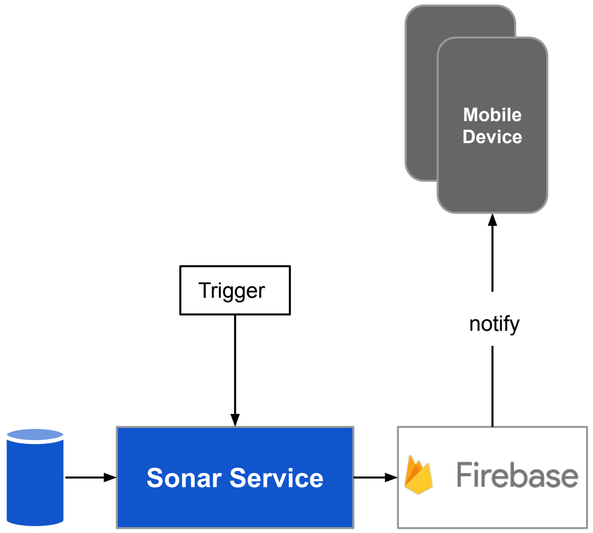
SUBMIT



3. Upload trigger causes device to ask the user to upload their stored data to the Sonar service

IMPLEMENTED

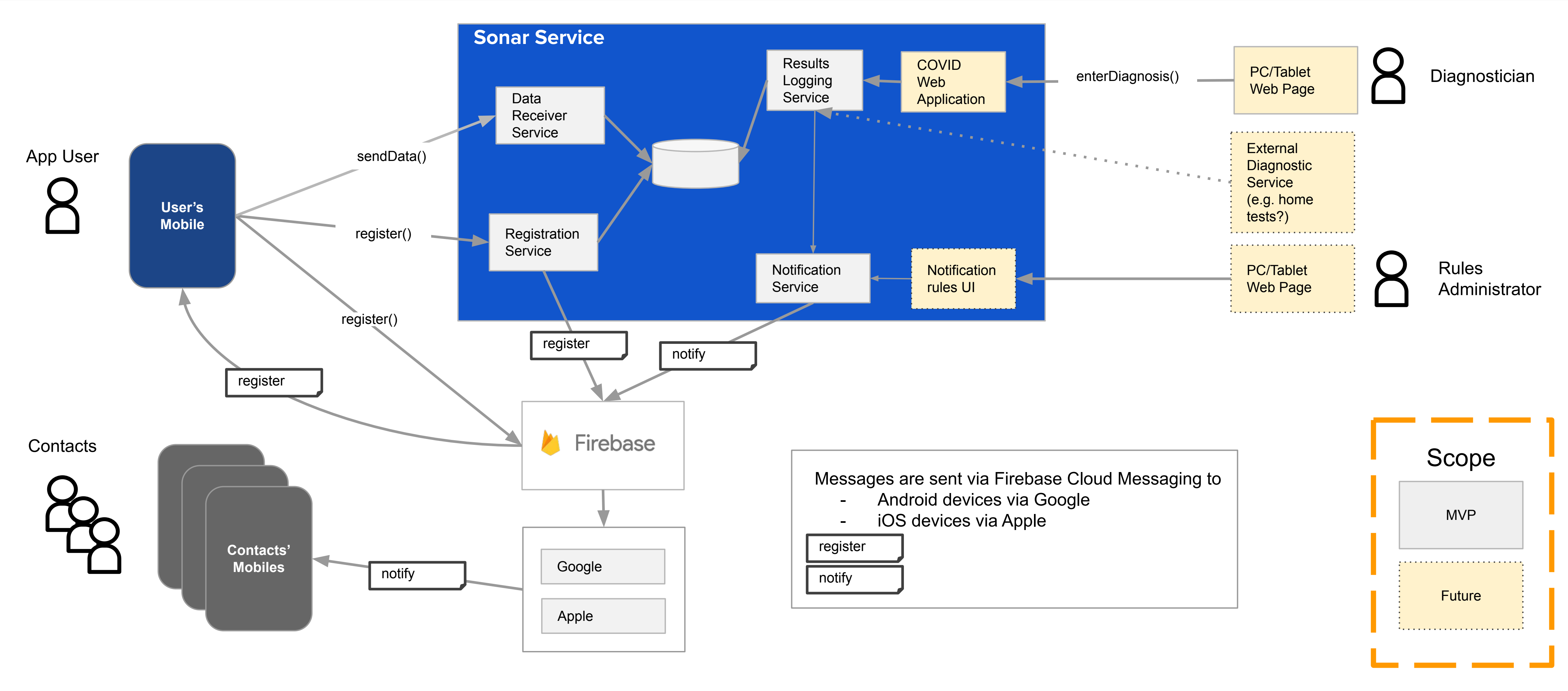
NOTIFY



4. Cascade trigger starts proximity cascade to send a notification to all encountered devices

IMPLEMENTED

Components - First full public release



Messages are sent via Firebase Cloud Messaging to

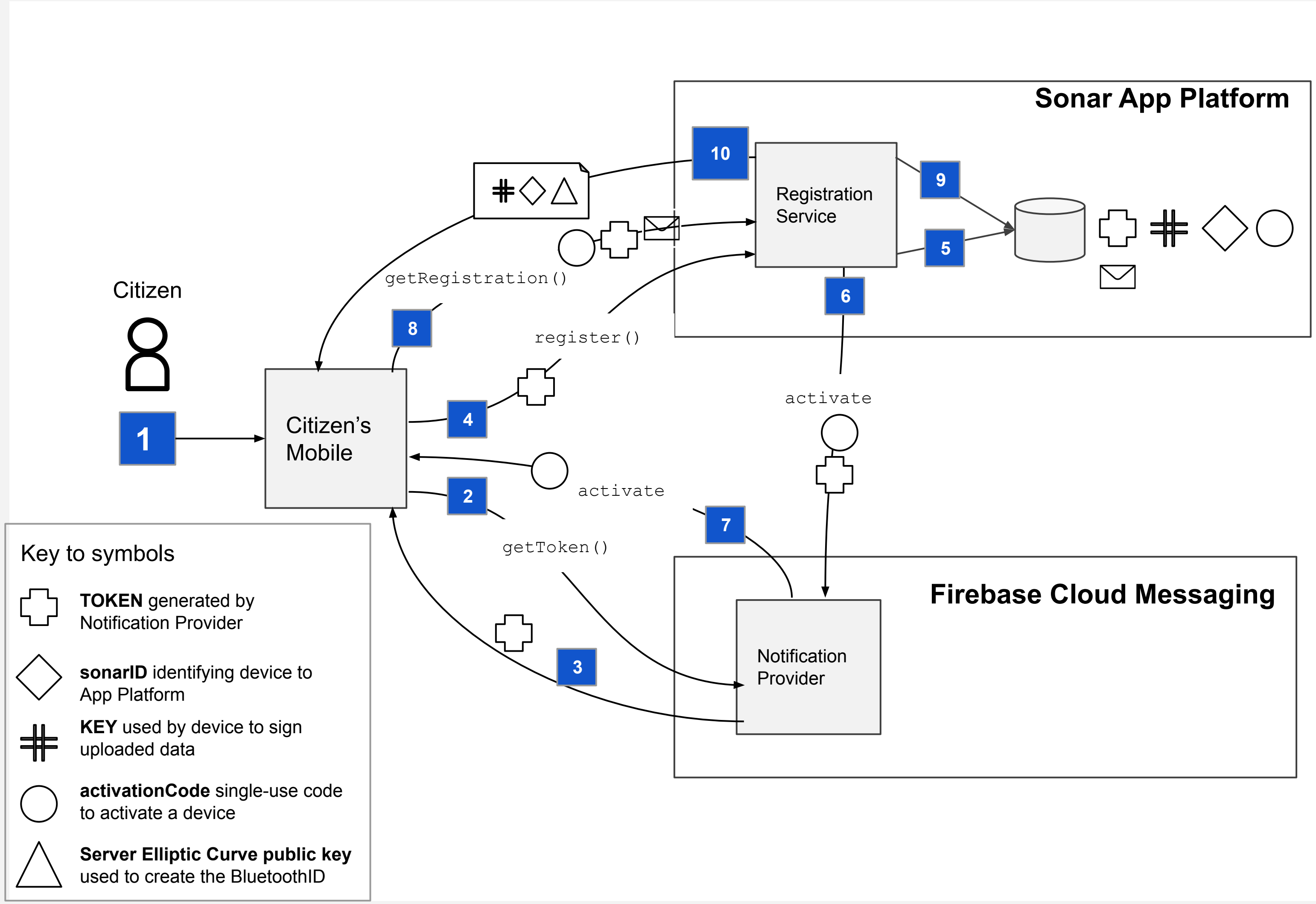
- Android devices via Google
- iOS devices via Apple

Architecture

Registration Process

How a new device enrols on the service. Enrolment does NOT take any personal or phone information. It does allow the backend to notify the app user in future. For example, if they have had a dangerous contact and need to take action.

Callback Configuration, Platform-generated anonymousID



Key to symbols

✙ **TOKEN** generated by Notification Provider

◇ **sonarID** identifying device to App Platform

KEY used by device to sign uploaded data

○ **activationCode** single-use code to activate a device

△ **Server Elliptic Curve public key** used to create the BluetoothID

Step	Data In Transit	Description
1		Citizen downloads app and initiates registration process
2		App contacts Notification Provider and requests a registration token. This generally happens at every app restart. A new token will be issued if the app has been reinstalled or the app data has been cleared.
3	✙	Notification Provider generates a token for this mobile and returns it to the app.
4	✙	App contacts the Sonar Registration service and registers its token
5		Registration Service generates anonymousID, Symmetric Key and and activationCode and stores them with the token.
6	✙ ○	Registration service send an activate message containing the activationCode to the device via the notification provider using the notification token
7	✙ ○	Notification provider forwards the activation message to the device
8	✙ ○ ✉	Device contracts the registration service, providing the token, activationCode, device type, postal district. If the activationCode has previously been used, raise a security event.
9		Registration services deletes the activationCode from the database
10	# ◇ △	Registration service returns the anonymousID and Symmetric Key to the device (one per device) and server Elliptic Curve public key.

Mobile Client Architecture

Device Proximity Detection

Various mobile devices have different support for their Bluetooth capabilities for detecting proximity. This section highlights the high-level challenges and steps being taken to overcome them.

As far as phone support, we have found:-

- Over 90% of UK phones in use support the Bluetooth Low Energy technology we require

Rotating IDs & Bluetooth network protocol

Based on the need to avoid allowing bad actors to try and track a user by the ID they are broadcasting

Decision made to rotate ID on 24 hour basis in order to provide feedback to the user on their social mixing score in future versions. The protocol does allow this period to be shortened if desired.

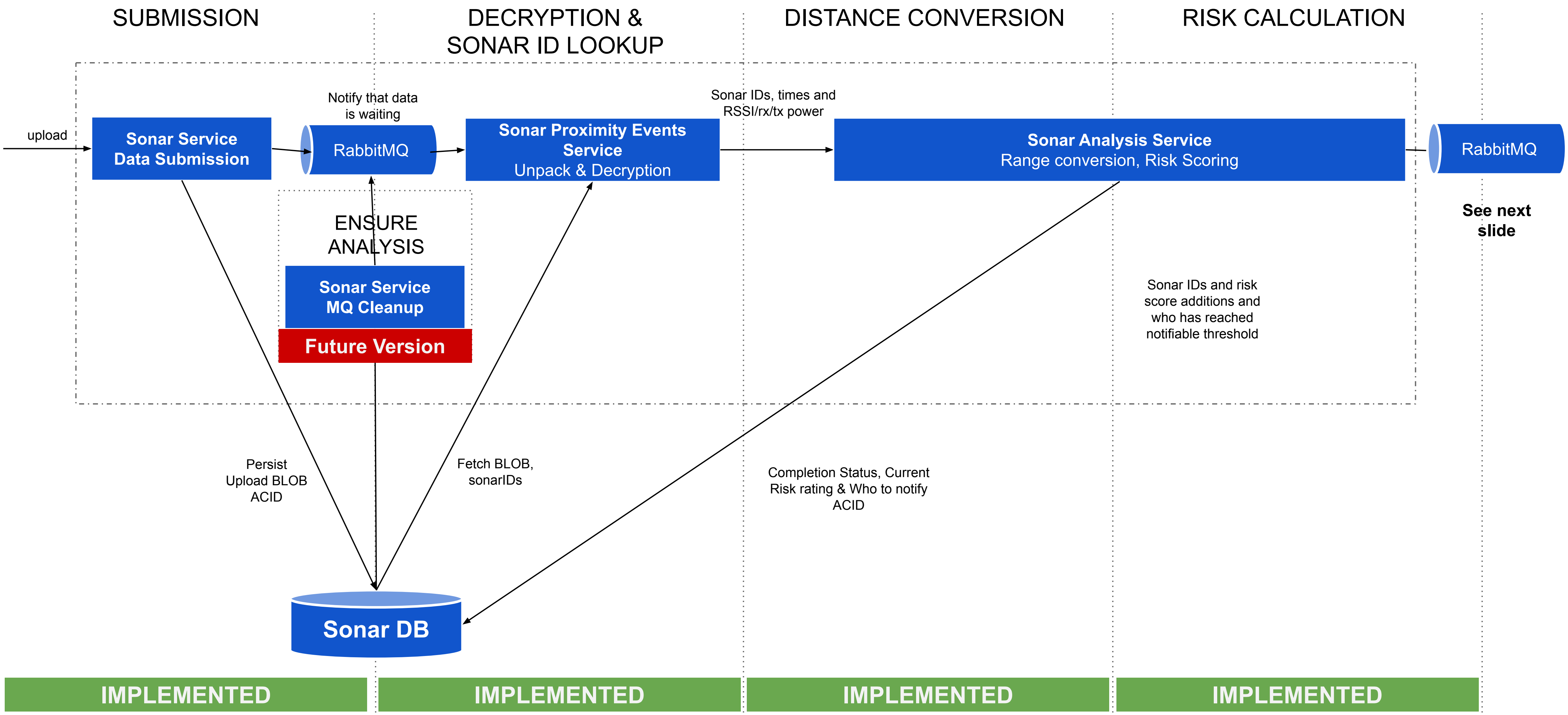
BLE Contact Tracking, Data Storage on Mobile Device

Implementation of contact capture through Bluetooth Low Energy (BLE)

See Overview - Data Logging and Submission - Mobile Device

Server Side Architecture

Proximity calculation and needs-notification flow

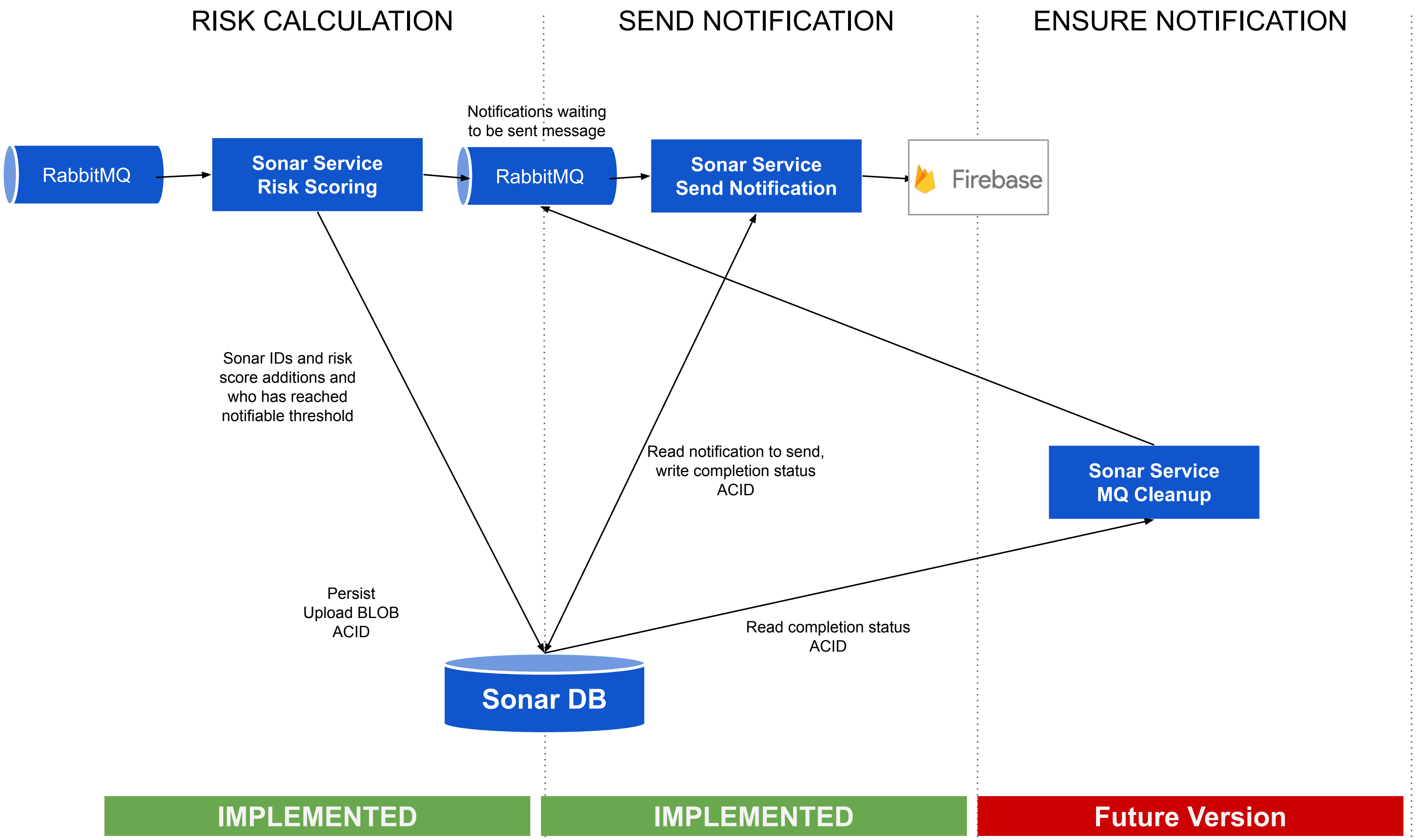

SUBMISSION
DECRYPTION & SONAR ID LOOKUP
DISTANCE CONVERSION
RISK CALCULATION
upload
Sonar Service Data Submission
Notify that data is waiting
RabbitMQ
Sonar Proximity Events Service
Unpack & Decryption
Sonar IDs, times and RSSI/rx/tx power
Sonar Analysis Service
Range conversion, Risk Scoring
RabbitMQ
See next slide
ENSURE ANALYSIS
Sonar Service MQ Cleanup
Future Version
Sonar IDs and risk score additions and who has reached notifiable threshold
Persist Upload BLOB ACID
Fetch BLOB, sonarIDs
Completion Status, Current Risk rating & Who to notify ACID
Sonar DB
IMPLEMENTED
IMPLEMENTED
IMPLEMENTED
IMPLEMENTED

See slide on Server Data Storage

See slide on Rotating IDs

See slide on Contact Risk Scoring

Send notification flow



RISK CALCULATION

SEND NOTIFICATION

ENSURE NOTIFICATION

Notifications waiting to be sent message

RabbitMQ

Sonar Service Risk Scoring

RabbitMQ

Sonar Service Send Notification

Firebase

Sonar IDs and risk score additions and who has reached notifiable threshold

Read notification to send, write completion status ACID

Sonar Service MQ Cleanup

Persist Upload BLOB ACID

Read completion status ACID

Sonar DB

IMPLEMENTED

IMPLEMENTED

Future Version

See slide on Contact Risk Scoring

See slide on Notification

Future Version

Server Data Storage

Supporting infrastructure required for capturing data on the server, processing data in the backend, and caching data for micro-services.

1. Relational database for primary storage
2. RabbitMQ for queueing requests on bursty, asynchronous endpoints
3. No caching data (Redis et al) required, but can be made available in the hosting platform if required in future

External system linking & Linking Reference Code

There is a need for the Phase 1 release - Isle of Wight and later interaction with other parts of the NHS to generate a temporary lived reference code for the user.

We cannot share the SonarID of the user as this could then be linked in other systems to PII (E.g. patient name and address for formal testing).

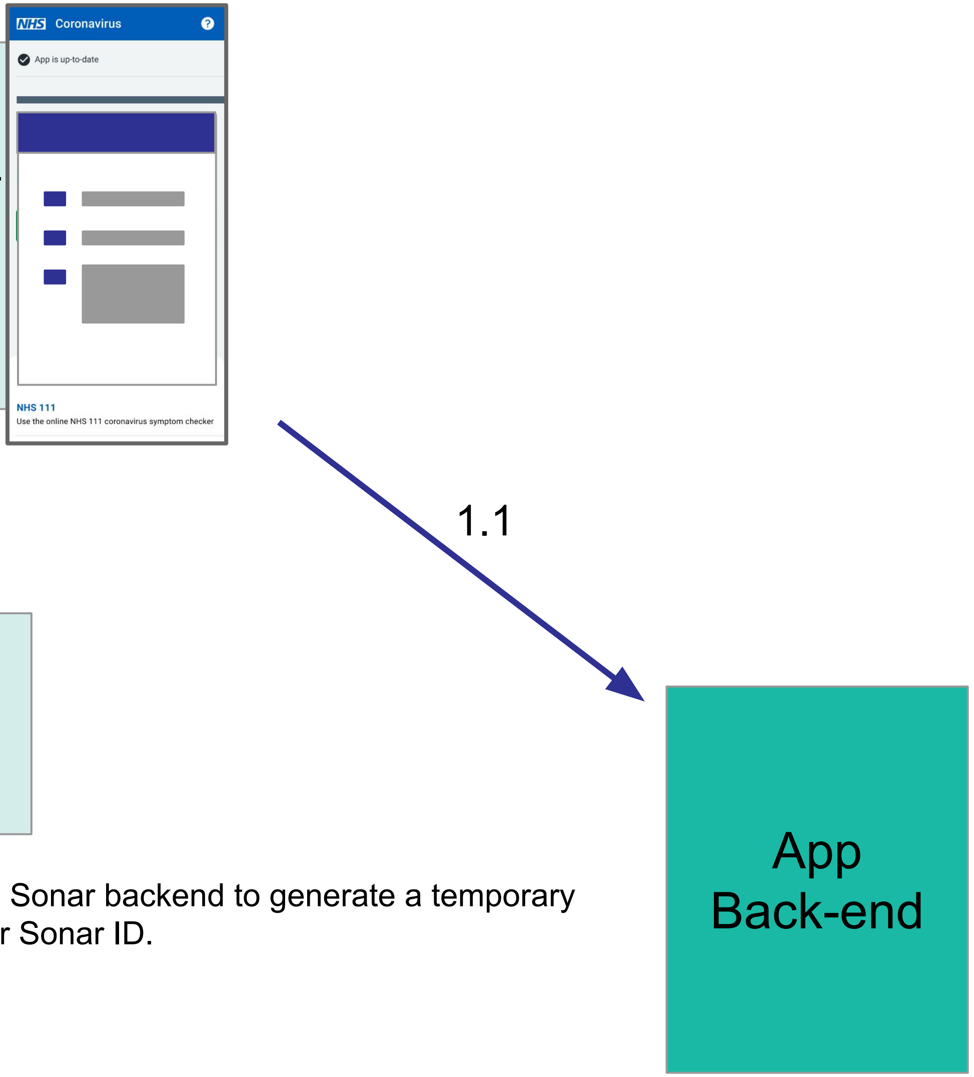
Instead the mobile app will request a one time use temporary Reference Code. (For the IoW phase this will be a single code lasting the full two weeks for all purposes. In future, it will be unique and time bound per external interaction).

A slide summarising this mechanism follows.

Approach for “v1” Privacy protecting reference code

User clicks ‘show my reference code’.

This is displayed to the user E.g. **ab5f-h38d**



Additional data never goes in to Sonar - interaction happens outside of the app

1.1: App contacts NHS Sonar backend to generate a temporary reference code for their Sonar ID.