I'mmune

COVID-19 Immunity Identification

A. Yüksel
K. Kostadinov
L. Franschman
R. Santana Trejo
S. Cirikka

TU Delft

l'mmune

COVID-19 Immunity Identification

by

A. Yüksel
K. Kostadinov
L. Franschman
R. Santana Trejo
S. Cirikka

at the Delft University of Technology,
to be presented on Friday June 26, 2020 at 11:00 AM.

Project duration:	April 20, 2020 – June 26, 2020	
Guiding committee:	M. de Vos,	Delft Blockchain Lab, Client
	T. Aerts,	TU Delft, Coach
	W. J. Baartman,	TU Delft, Teaching Assistant

TU Delft

Preface

The document you are reading right now is a report on the project titled: “Building a critical infrastructure for the nation-wide identification of recovered COVID-19 (Corona) patients”, provided to us on the project forum of TU Delft. We participated in this project for the course “Software Project” (CSE2000), which is a second-year course that takes place in quarter 4 of the academic year 2019-2020. We are a group of 5 students who have taken an interest in the pandemic, which is still going as of writing this report (June 6, 2020). We wanted to chip in to the battle against the pandemic somehow, and we figured that participating in this project was an unmissable opportunity for doing just that.

Creating an app for providing Immunity Passports in the context of COVID-19 would not only be a way to help resolve the current situation, but it would also serve as a basis for similar technologies solving similar problems that can perhaps build upon the framework we have set up with this project. Especially by having used SSI, a technology which is gaining traction rapidly thanks to its privacy-preserving nature, the project we have participated in will almost certainly remain relevant in the coming years. For these reasons, we are proud to have participated in a project that matters.

Making the project in line with the desires of both our client Martijn and ourselves was a fun and educational journey. We would like to thank him for shaping the project together with us by conveying his desires clearly and hearing us out on our ideas as well. We would also like to thank our teaching assistant Wesley and our coach Taico for helping us throughout the project. Finally, we would like to thank you, dear reader, for spending some of your time reading our project report.

Without further ado, please enjoy.

A. Yüksel
K. Kostadinov
L. Franschman
R. Santana Trejo
S. Cirikka
Delft, June 2020

Summary

Our project's client is the Delft Blockchain lab. For this project, we were tasked with creating a COVID-19 immunity check to integrate in their identity solution. The main goal of our project is to protect people that are still not resilient to the virus, whilst allowing for the reopening businesses and providing support for everyone in need. Immune people will be extremely valuable in the months after the pandemic is over, because there is always a possibility of a second wave of the same pandemic, which might have worse consequences than the first. "Immunity passport" holders may be used as a resource. This would mean that these people could be sent to infection hotspots in order to provide critical support.

In the context of what we were tasked with, we made an Android app which allows users to prove their immunity for COVID-19. The application that we developed consists of three main parts: The front-end, the back-end, and The Android application. To ensure a robust app in terms of the front-end, we created it using React Native and we used Typescript as the main scripting language. The back-end is written in Python and is responsible for handling threads. The Android Application is written in Java and is responsible for gluing together the back-end and the front-end.

The two roles involved in the creation of a proof are the attester of the proof and the holder of the proof. The attester is the professional in charge of issuing a proof, whereas the holder is the individual to whom the proof is issued. The process of acquiring proofs follows the procedure of the IPv8 attestation service. Basically, the attester (a health expert) will issue an immunity proof for COVID-19 to a patient through the use of our app. Afterwards, the patient will attest to this by again using our app. Finally, after validation, the patient will possess a proof for the COVID-19, as issued by the health expert, and thereby become a holder of a proof. By using this pipeline and with it the IPv8 library, we achieve preservation of privacy. This is because the Identity Solution of the Delft Blockchain Lab is based on the ten principles of Self Sovereign Identity, a technology which is privacy-preserving in nature. Bundled with this, we also provide the holder of the data with absolute control over it. The user can delete any proof and even their entire account if they so much as feel like doing it.

However, this project can also serve as inspiration as a building block for more elaborate projects which can expand upon what we made. We ensured great expandability for such a scenario, both in terms of front-end and back-end. The front-end and back-end are also decoupled, so each of them can be replaced by something better without anything falling apart.

Contents

1 Introduction

Over the course of history, humankind has faced many epidemics and pandemics. Some of these, such as the multiple Influenza epidemics, have killed millions. Obviously, this was not without consequence for the economy during those times. The illness caused by SARS-CoV-2, COVID-19, has the potential to deal a hard blow to the economy as well. With millions infected and hundreds of thousands deceased at the moment of writing of this report, the pandemic is still going strong. The unemployment rate in the US has seemingly risen above 15%, which is already more than the peak unemployment rate during the Great Recession.

The aim of the project we are writing a report of is to produce an app which allows people who are immune to the disease to, for example, be able to work again without any risk of getting themselves or others infected. This way, we can further reduce the impact of the pandemic caused by SARS-CoV-2 on the economy. We have developed this app such that it is privacy preserving and easy to use. The way we have achieved all this is by extending the SSI framework of TUDelft with a health app, which functions as an app to acquire and use immunity passports. These immunity passports will serve as proof that the holder is allowed to participate in society as they did before the pandemic. By using an SSI framework, we have ensured preservation of privacy. This report document our choices in the design and implementation of our project.

The report will look as follows: Chapter 2 gives insight on the problem at hand and various problems that are tied to it. For details on the solution we implemented, please refer to chapter 3. An in-depth description of the technical aspects of our implementation can be found in chapter 4. Chapter 5 contains our take on the project as a whole and what we recommend are the next steps for this project. Next, in chapter 6, we will discuss the ethical implications our app might have. Finally, the conclusions we arrived at are located in chapter 7.

2

Problem Analysis

In this chapter we will define our problem as introduced in chapter 1 and describe the different facets of this problem. Firstly, we present a bit of background on COVID-19 as an introduction to our problem definition. Afterwards, we discuss immunity and immunity passports in the context of COVID-19 and some pain points in regards with these immunity passports. Then, we will discuss the idea of a Self-Sovereign Identity as the basis for these immunity passports. Lastly, we talk about the Delft Blockchain Lab being our client.

2.1. COVID-19

COVID-19, the disease caused by SARS-CoV-2, has affected the entire globe [1]. The negative impact is not only felt by everyone that is currently infected and the healthcare personnel, but the entire world economy suffers the consequences of the disease [2]. Countries around the world have tried several strategies, from informing the public on the virus, enforcing curfews, intensive contact tracing of infected individuals and many more [3]. The decisions taken by governments, which might turn out to be even more harmful than the pandemic itself, are one of the reasons people are witnessing the forming of a global economic crisis. The imposed lockdowns and measures to protect the public from the virus are changing the way our society behaves, which forces many owners to cease their business activities temporarily and in some cases permanently. As a result, the way cash flows throughout the economy shifts and many people end up struggling with their monthly bills or even losing their jobs. In order to recover from this inevitable crisis, governments have to take into account all trade-offs between rescuing the economy and public safety. This would lead to the imposition of new regulations and rules that people and businesses will be forced to follow.

2.2. Immunity

One of the main ideas to combat the current pandemic is to make use of immunity as a resource by introducing “immunity passports” and issuing them to people that are considered immune to the virus [4]. People who possess them will be allowed to again live their life normally and engage in many sorts of activities, such as working or studying. The main goal is to protect people that are still not resilient to the virus, whilst reopening businesses and providing support for everyone in need.

2.2.1. Building Immunity

Usually, immunity to a disease is proven through a test, which checks for the presence of antibodies associated with this disease inside a person’s organism. However, since this is a novel virus, one of the main problems that needs to be taken into consideration is the fact that scientists have not yet proven that the presence of antibodies in someone’s organism is solid evidence that this person is immune to SARS-CoV-2 [5]. Although repeated infections might turn out to be possible, scientists think that this is highly unlikely and it is generally believed that former patients of COVID-19, that test positive when checked for antibodies, have built immunity [6]. Considering the importance of the project and the fact that the virus is still spreading widely, the possibility of reinfections will not be regarded as a possible

hurdle. When the duration of the immunity is later defined, necessary corrections will be applied.

2.2.2. Immunity Passports

The problem with lifting the imposed emergency regulations might be efficiently solved through the use of immunity passports. They effectively provide people with a “permit” for their usual economic and social activities. Immune people will be extremely valuable in the months after the pandemic is over, because there is always a possibility of a second pandemic, which might have worse consequences than the first one. “Immunity passport” holders may be used as a resource. This would mean that these people could be sent to infection hotspots in order to provide critical support [6]. There are several ways in which one could grant such an “immunity passport”. A mere passport on paper could suffice, however this brings with itself several issues of authentication and authenticity [7]. Furthermore, immunity passports pose a privacy issue because by using them, people are forced to share their health condition, which might end up in misuse of sensitive information. The “passports” also might make use of mobile technology, which will further exclude parts of the population, currently not having access to such technologies. Nevertheless, there are already more than 60 companies working on defining immunity certification. Most of them are using blockchain technology, as it provides some of the required privacy and security properties [8].

2.3. Privacy and Security

As discussed in section 2.2.2, there is a risk of personal information misuse cases, which poses an enormous privacy threat. This might mean that people are less likely to trust the technology and they might choose not to use it, which will make it obsolete. Most of the companies that are working towards a solution, use blockchain technologies to address these caveats. Historically, Bitcoin was created to allow people to have full control over their information, ultimately trying to solve the privacy and security issues. But, since 2009 there have been many cases in which personal privacy was breached [9]. Recently, the World Wide Web Consortium released a new standard called “Verifiable Credentials Data Model”, which claims to provide security and anonymity to its users. This model allows users to prove each other claims about themselves without revealing sensitive information [10]. The problem that remains is that such a system is not widely deployed yet.

2.4. Self-Sovereign Identity

One approach which has gained a lot of traction recently is the usage of a Self-Sovereign Identity (SSI). This is an implementation of the “Verifiable Credentials Data Model”, as discussed in section 2.3. The idea behind SSI is that citizens have full control over their own identity. The individual has the complete right to their own data and with whom they want to share what piece of their own data. It’s a decentralized identity which is solely of the individual him- or herself. Instead of the government issuing identification documents, every person could claim information about himself voluntarily and organizations could then attest these claims if they are correct. For example, holders of SSI could claim that they are above a certain age and request a responsible organization to attest this claim after which the holders could use this proof by showing the appropriate attestation without revealing any other sensitive information about themselves (e.g. exact age or date of birth) [11]. The ten principles of Self-Sovereign Identity is a guidance for many SSI solutions [12].

2.5. Delft Blockchain Lab

The task set is to combine these two topics, namely the granting of an immunity passport through the use of a Self-Sovereign Identity. The client, Delft Blockchain Lab, is TU Delft’s initiative for research, education, and training in blockchain technology and trust in the internet. In cooperation with the Ministry of the Interior and Kingdom Relations it has developed a library for SSI that is set to provide passport-grade identities in the future [13]. The outcome of this project is adding the COVID-19 “immunity passport” feature to the library as it could prove very useful in the coming months. Furthermore, if this project is successful and ends up becoming a nation-wide deployed system, it could possibly be the first step towards mass adoption of Self-Sovereign Identity.

3

Product Design

3.1. Requirements

The first key step in designing our product was to use our problem analysis to decide what features it would include. We used the MoSCoW method to write down and prioritize those features. In our case we decided to create an android app that can be used as a means of communication between certificate attesters, certificate holders and verifiers. This communication would happen on an SSI oriented platform so that the holder could have full control over their data. Our MoSCoW list reflects that overall product idea, in the “must haves” we include the core features that enable this app to function, while the rest of the list contains ease-of-use or commodity features ranked based on how useful we and the client considered them to be. Our MoSCoW list is as follows:

Functional Requirements

Must Haves

- Attesters can create certificates
- Attesters can send certificates to corresponding holders via their key identifier
- Holder can view their certificates
- Holder can accept or decline certificates
- Holders can present their certificates to verifiers
- Verifier can review certificate contents
- Verifier can verify the certificate signature

Should Haves

- Application configuration and authentication
- Notifications about certificates

Could Haves

- Certificate time-out
- Receiving an old certificate
- Reviewing a certificate for verification automatically
- Creating new types of certificates for other purposes (not just COVID-19)

Won’t Haves

- Questionnaire for determining whether a person has COVID-19
- iOS support, since the process of deployment does not suit the project timeframe
- Method for requesting certificates

Non-Functional Requirements
Must Haves

- Holder can at all moments control what data is added to their chain.
- No user can access a Holder's data without the Holder's explicit consent.
- The backend uses the IPv8 library
- The whole pipeline can be performed without the need for anyone to reveal any personal data other than their online ID.

Should Haves

- The backend is flexible and allows diverse frontends and certificates.
- The app is ready for small-scale deployment.
- The app can be expanded through further development to the point of large-scale deployment.

Won't Haves

- The app is sufficiently tested for immediate nation-wide deployment

With this requirement prioritization we ensure that the entire attestation and verification processes can be performed within the app while keeping expandability and usability as a high priority. We also define the scope of the project by limiting the production level of the app to small trials.

3.2. "Immunity Passports" Using SSI

Our Solution to the problem that'll occur when lifting the imposed emergency regulations is handing out immunity certificates. They would serve as "permit" for their usual economic and social activities. Immune people will be extremely valuable in the months after the pandemic is over, because there is always a possibility of a second pandemic, which might have worse consequences than the first one. "Immunity passport" holders may be used as a resource. This would mean that these people could be sent to infection hotspots in order to provide critical support. In order to get such a health certificate, people must share their medical information or health status. this means that people need to give out a lot of personal information that might have a chance of ending up being misused. Therefore we needed to find a way to collect this kind of sensitive information while making sure people can maintain their privacy and security. In order to do this we use SSI (Self Sovereign Identity) principles. Those principles are designed to protect user data by giving them ownership of their information. By placing the sensitive data in the hands of the owner, most privacy concerns which relate to a centralized data point owned by a third party are circumvented.

We realize these SSI principles as follows: A holder requests a certificate. An issuer must be able to pick the correct type of certificate from a list. He must then be able to fill in the contents of the certificate. After the issuer learns the key identifier of a holder and creates a certificate, he must be able to sign it, save it to his blockchain and send it to the holder using his key. When a holder receives a certificate for review, he must be able to see it in his inbox. After a holder receives a certificate, he must be able to access it through his inbox, manually review its contents and choose whether to accept or reject.

After all that, a holder must be able to verify his obtained certificate. A holder does this by choosing the certificate he needs from a list of all his certificates. After picking it, the certificate is shown in appropriate format for verification (e.g QR code). Now it is verifiable and can be verified by a verifier. After reviewing the certificate either automatically or manually, both parties get notified by either showing a pass or a fail notification.

3.3. System Description and Main Features

Our final product allows for users to sign attributes and verify signatures in a secure and privacy focused way. These attributes can be of different types, in our demonstration we show the use case of proving an immunity for the COVID-19 disease.

Roles and Terminology:

Roles:

- **Attester:** professional in charge of issuing an attribute.
- **Holder:** individual described by attributes.
- **Verifier:** professional / company who needs the proof of the Holder's attribute.

Terminology:

- **Chain:** the data storage for an individual in the network,it is composed by attributes that define an individual's identity in the system.
- **Attribute:** information that describes a fact about a Holder, this information has been signed by an attester, is controlled by the Holder and can be checked by a Verifier.
- **Certificate:** a request sent by the attester to the holder for the creation of an attribute in the Holder's chain.

The Process:
The system starts when an Attester knows the Holder's online ID and is willing to sign an attribute in the name of a Holder. The online ID can be shared by any means (phone call, email, QR Code...). Once the professional has the online ID the attestation and verification processes can be seen in figure 3.1 :

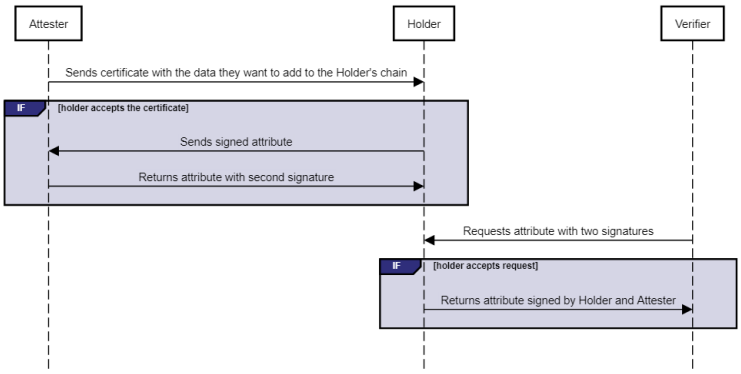


Figure 3.1: Sequence Diagram for Attestation and Verification processes

Once this process is complete the Verifier can be confident that the attribute at hand was attested by a specific user of the network. It is up to the Verifier to decide whether the given Attester has sufficient authority to attest the claim.

The conditional "IF holder accepts the certificate" step is a key feature that makes this system follow SSI rules, the Attester cannot impose on you what data is available or what identifies you, you have to actively choose what forms part of your identity. Another key step in keeping holder control is the "IF holder accepts request" step, this tackles the issue of companies accessing an individual's data without their permission and allows a Holder to actively choose exactly what data is being shared with the Verifier. It is not the Holder's entire medical history or curriculum being shared, it is only the data that the company officially requested and that the Holder authorizes.

Main Components The system is divided into two main components, the React Native Frontend and

the Python Backend, both of which are imported and started by an Android application.

The React Native Frontend makes up the part of the application that the user interacts with. It is divided into several screens based on the different steps of the attestation and verification processes. The frontend communicates with the Python Backend using HTTP requests, which are performed as needed in the app. For a better description of the Frontend design refer to section 3.4.

The Python Backend is in charge of managing the storage, transmission and signing of attributes, most of it is the IPv8 library but we did add some extra endpoints and functionality to satisfy our specific needs. The backend can be controlled exclusively through its API which allows flexibility to change the frontend, as long as the API calls are respected, the Frontend and Backend can be changed or upgraded if desired.

More details on how these technologies are used in our app and how they are pieced together can be found in Chapter 4.

Main Features

We believe our product has various interesting and unique features that contribute to its usability and expandability. We have listed them below for ease-of-read:

- **Full attestation and verification pipeline:** the product allows for users to build up an identity formed of attributes which have been attested by other users. If a user is given the right credentials, they can verify the attribute of another user and check who signed it.
- **Absolute Holder control:** the Holder of the data can always control what data is added to their identity and when it is accessed, this allows our system to follow the SSI principles and solves many of the modern user privacy issues.
- **Decoupled Frontend and Backend:** The Frontend and Backend use a very slim API to communicate with each other, this allows for either system to be replaced or updated very easily.
- **Expandable:** the system currently handles one type of attribute (COVID-19 Immunity) but both the Frontend and Backend have been built to be able to handle multiple types of attributes, this will require minor changes to the source code to define the new attributes.
- **Security and Integrity:** we used the IPv8 library to handle all storage, communication and verification steps, this means that the main attack surfaces of our application are handled by the IPv8 library, which has a lot more development time, testing and experienced academics as developers. Also the IPv8 system goes to great extent to guarantee attribute integrity, which is a key aspect for using the app as means of verification of data.

3.4. Frontend design

A good User Interface should not be about making an application just usable for people. it should be easy for the users to see what the product is and it should be designed in a way to display the services that you offer without ambiguity. In other words: user friendliness was our main concern when designing the GUI.

Our frontend is going to be built as a mobile application which consists of multiple screens each with its own purpose. The first screen you'll see is the login screen. This screen provides extra security and when you enter your password, you'll see the dashboard screen. the dashboard screen will contain all your obtained certificates and a part of your completed Blockchain. This is also where the proofs can be verified with a QR code that'll show after clicking on one of the obtained certificates. You can access other screens from here by clicking on the menu button in the top left corner. This button will be available on every screen at the same location.

In order to get a certificate you'll need to have an attested claim that proves a certain something. If you want to get an immunity proof for example, you'll first need to request it at an issuer. this issuer then goes to the New Certificate Screen where the issuer can choose the type of certificate to send. but before the issuer can hit send, the issuer needs to type in the reference key of the holder so that the

certificate will be sent to the right person. The issuer should now have this certificate in the Outstanding Screen. It will stay there until the holder has either accepted or rejected the certificate. The holder on the other hand now receives the certificate in the inbox. Here the holder can reject and accept certificates. After all this is done, the holder will be able to see his immunity certificate in the dashboard.

We made our screens more user friendly by Giving the the different options in our application clear names and by adding some guiding text like choose here or password here. But in case the user is still not so sure what to do, he can visit the Help Screen. This screen is accessible by clicking the help button in the top right corner of any screen. You can find information here about the entire application.

As mentioned before, we use SSI because we want the user to have total control over their data. That also means that they should be able to completely delete their account. They can do this in the setting screen. Here you also have the option to delete a certificate and customize the application colors. Before deleting anything, there will be a pop up screen asking for confirmation so nothing can be deleted accidentally.

4

Implementation

After introducing the main problem this project is solving and describing the general design of our solution, it is time to dive into the details of the actual implementation. The application that we developed consists of three main parts: The front-end, written through the use of React Native, which is responsible for the whole graphical user interface and the calls to the API, provided by IPv8; The back-end, written in Python, which is responsible for handling threads and contains an extension to the IPv8's API; and The Android application, written in Java, which is responsible for gluing the front-end and the back-end together.

In order to understand the decisions that we have made about the front-end and the back-end, we need to know what restrictions Android poses upon developing such a system. That is why this chapter is structured in the following manner. First, we start by giving an in-depth view of the Android application's development. Then we are going to continue our journey with the back-end and we are finishing the chapter with argumentation about how our front-end is made suitable for any type of user.

4.1. Android and IPv8

The main challenge during the entire project was the design of the application. First, the IPv8 library is written in Python which is not supported by Android. Second, the IPv8 library is meant to run on a static machine and not on a mobile device. Essentially, Android's architecture does not allow for such services to run indefinitely in the background [14].

4.1.1. Packaging IPv8 Within Android

There were two main possibilities for tackling the former issue. We could have rewritten the entire library in Java or we could have used a bridge between Python and Android in order to package the already existing code and make it available in Android through a bridge-like API. We decided to go with the latter, since we were constrained in time, the IPv8 library was well developed and tested and someone else was already rewriting IPv8 in Kotlin, which is a language derived from Java.

By deciding to use packaging for Python, we were faced with the decision between three main tools that were going to provide the bridge. We splitted the back-end development into three branches and our plan was to just try every single tool and the first one to work was going to be merged with the main development flow. Between Chaquopy [15], Beeware [16] and Kivy [17], Chaquopy looked most promising. It was easy to set up, by just adding it as a dependency to the Android application, and we were able to call Python modules with only one line of code. Beeware was easy to use as well, but was in an early stage of development and was not thoroughly tested. Kivy on the other hand was around for quite some time, but it was really difficult to use, since we needed to explicitly add the whole library to the project and use a custom tool to build the application. That is why at the end we went with Chaquopy.

However, there was one disadvantage of choosing Chaquopy. The library is licensed. We got into contact with the maintainer of the tool and were able to get a license key. The key is going to be

available free of charge and includes lifetime support as long as this project stays open-source, which is the plan. As a result we had to add a basic LGPL-3.0 license to the project.

Furthermore, after getting the bridge up and running, we encountered another major issue. IPv8 uses a library called Libsodium [18] for its cryptographic needs, like key creation and management, encryption, etc. The problem was that Libsodium was not available for Android. After some research, we were able to find a project on GitHub [19] which had some custom build scripts that were building the library for the various processor architectures that Android devices use. Since they were making the library available for use in Java code, but we needed it to be available to the IPv8 service, we were forced to modify the build scripts, build the whole library locally and extract the base .so files for the architectures which Chaquopy supports, namely ”x86”, ”x86_64”, ”armeabi-v7a” and ”arm64-v8a”. Indeed, those architectures are the most frequently used by Android devices [20]. Libsodium is as of this moment a dependency of our project which is not made trivial to update. Probably in future versions this problem can be addressed, but for now this solution suffices.

4.1.2. Background Services in Android

After dealing with IPv8 and successfully packaging it within the main Android application, we stumbled upon yet another problem: Compatibility. Every single change to the design that we made was working for just a subset of all available Android versions. Since this app targets the whole population, we needed to ensure that it will be widely available for as most Android users as possible. Our goal was to provide support for Android APIs 16 through 29 [21].

Android applications generally consist of three parts: Activities, Services and Receivers [22]. Activities are the screens with which the user interacts. Services are responsible for all the tasks the user needs, but is not actively interacting with. And lastly, Receivers are components that provide means of communication between different components within an application or between separate applications. The issue is that different components have different life cycles on the various Android versions and the choice of the correct component for running the IPv8 service was not a trivial task.

Our initial prototype was relying on the packaged library running in the background. We found a feature in Android which allowed for keeping the Service alive without a need for the application to even run. But later it turned out that this feature is only available to APIs below and including 25 [23]. This led to a total redesign of the application.

Since every single application that is installed on an Android device receives its own Linux process [24] we decided to split the main application into two separate ones. The first one was going to take care of the graphical user interface while the second one was going to just run the IPv8 service in the background. The idea behind this refined design was to exploit the abilities of Android by tricking it into thinking that there was always an invisible Activity in the foreground and binding the Service to that Activity in order to prevent the system from killing it. The reason was because the operating system is allowed to kill any process for which it is sure that is not needed by the user [14].

However, it turned out that this was not the best design choice, since again the application was not robust enough and it behaved differently across our target Android versions. Also for user convenience we wanted to force install the second application through the first one, but Android considers every Android Package that does not originate from Google Play Store as malware and requires special permissions from the user [25]. At the end we needed to redesign the application again, since instead of making it more user-friendly it had become less.

Our third and final design however turned out even better than we expected. We were able to introduce a new feature to the user, mainly making the service optional in the sense that it was always required whenever the application was open, but keeping the service alive after the application closes was left as an option to the user. The way we achieved that was through the use of a Widget [26]. A Widget is just a special type of receiver which gets bound to the user’s home screen and the user has total control over it. It was used as a binding point for the service, in order to prevent the system from killing our process. We also brought the service to the front, meaning that we changed it from being a background one to being a foreground one. The latter was only done because, since API 28, only foreground services are guaranteed to not be interrupted and killed by the system throughout their lifecycle [23].

Alongside improving user experience, our final design also improved battery life and network usage. The reason for that was because the service was made optional and when it is not running, it does not use any battery nor any network packets. The application now allows normal users to run the service on demand and professional users to run it indefinitely when needed.

4.2. Trustchain as Reliable Certificate Storage

While we have mentioned the IPv8 library quite a bit now, we haven't really dived deep into its inner workings. Therefore,In this section we will give a somewhat deeper explanation of what the IPv8 library actually is. After that, we will go over TrustChain and its role in reliably creating these immunity passports (or any other attestation, for that matter). Lastly, we will discuss how we have extended the IPv8's API to suit our needs.

4.2.1. IPv8 workings

IPv8 is a network layer which works on top of The Internet that we know. It has been developed over the last 13 years by students and employees of the TU Delft [27]. It is aimed on offering identities to their users with which these users can communicate with other users. Authentication of the users on the network is guaranteed through the usage of cryptography. This means that a user can be sure that he/she is talking to whom he/she thinks he/she is talking to. It furthermore offers global connectivity and end-to-end encryption in a decentralized manner. Lastly, it tries to achieve distributed trust in this network by making use of TrustChain, which we will discuss in 4.2.2. To sum up, IPv8 is much more than just a way to create immunity passports. We will however look a bit more closely into how IPv8 can provide this service for us. The IPv8 library has an attestation service which can provide attestations for their users. The workings of which are best illustrated by a flow example:

(probably a graph here or something)

Assume we have two peers, Peer 1 and Peer 2.

- Peer 1 sends a request to Peer 2 with the message: {please attest: attribute_name}. In which attribute_name would be whatever attribute Peer 1 wants to have attested by Peer 2.
- Peer 2 sees this request and can reply to it with a value: {ok I attest: attribute_name, value}
- Peer 1 will now have this attested request in his/her personal record, which he/she can show to whoever he/she wants.

This attestation service is the basis on which we have build our app. We would like to note that this is quite a high level overview of IPv8 and the attestation service. For a more extensive overview please refer to the documentation [28] .

4.2.2. TrustChain

This personal record in which Peer 1 saved his/her attribute, is handled by TrustChain. TrustChain is a tamper-proof, scalable and blockchain-based data structure [29]. It differs from the blockchains used by cryptocurrency in that every peer has it's own personal ledger compared to a global ledger. The attestation service provided by IPv8 leans heavily upon TrustChain. The way that IPv8 uses TrustChain can also be best illustrated by a flow example: Assume once again that we have two peers, Peer 1 and Peer 2.

- Peer 1 wants to do a transaction with Peer 2. To do so, Peer 1 creates a block which has included in it the transaction and a signature of Peer 1. Peer 1 saves this block to his/her own personal ledger and sends it to Peer 2.
- Peer 2 sees this sent block and, if agreeable, creates a complementary block on which it puts his/her own signature. Then it sends this complementary block to Peer 1 while also saving this block to his/her own personal ledger.
- Peer 1 and Peer 2 now both have immutable proof of the transaction as they both have both blocks in their personal ledger. Figure 4.1 shows an example of this combined block which has the signature of both parties in both of their ledgers.

There are certain properties guaranteed by TrustChain. It is tamper-proof, as both parties have proof of this transaction. This long chain of blocks is immutable, as blockchain technology can guarantee. Lastly, with every transaction the web of blocks grows and grows, as can be seen in Figure 4.2. However, because every user only has to keep track of their own ledger, it is very scalable.

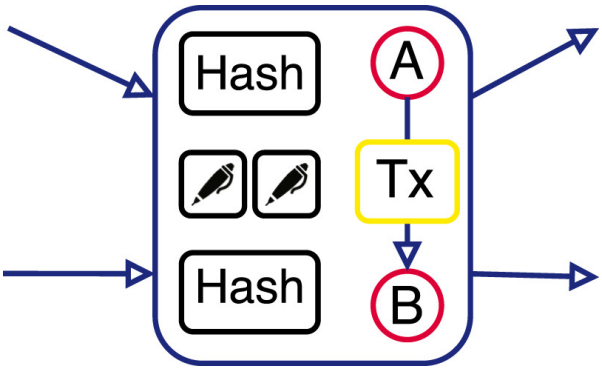


Figure 4.1: An example block.

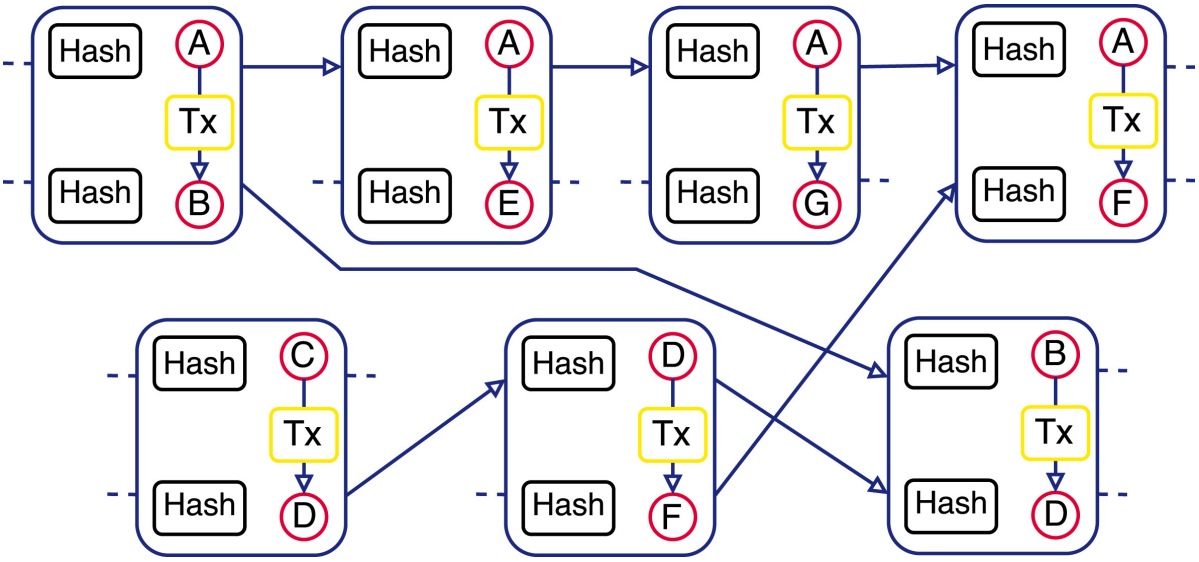


Figure 4.2: Web of blocks.

4.2.3. IPv8 API extension

One can clearly see the similarities between the two flows. Our implementation's task was to make use of the IPv8 attestation service to suit our needs. The attestation service is provided to us through an API shown in Figure 4.3. As the service runs on the local machine, one can call these and several other API through the methods in these EndPoints. From a high overview, every API request goes through the RootEndpoint which will delegate the call to the appropriate endpoint. And then write some other stuffffffffff todo.

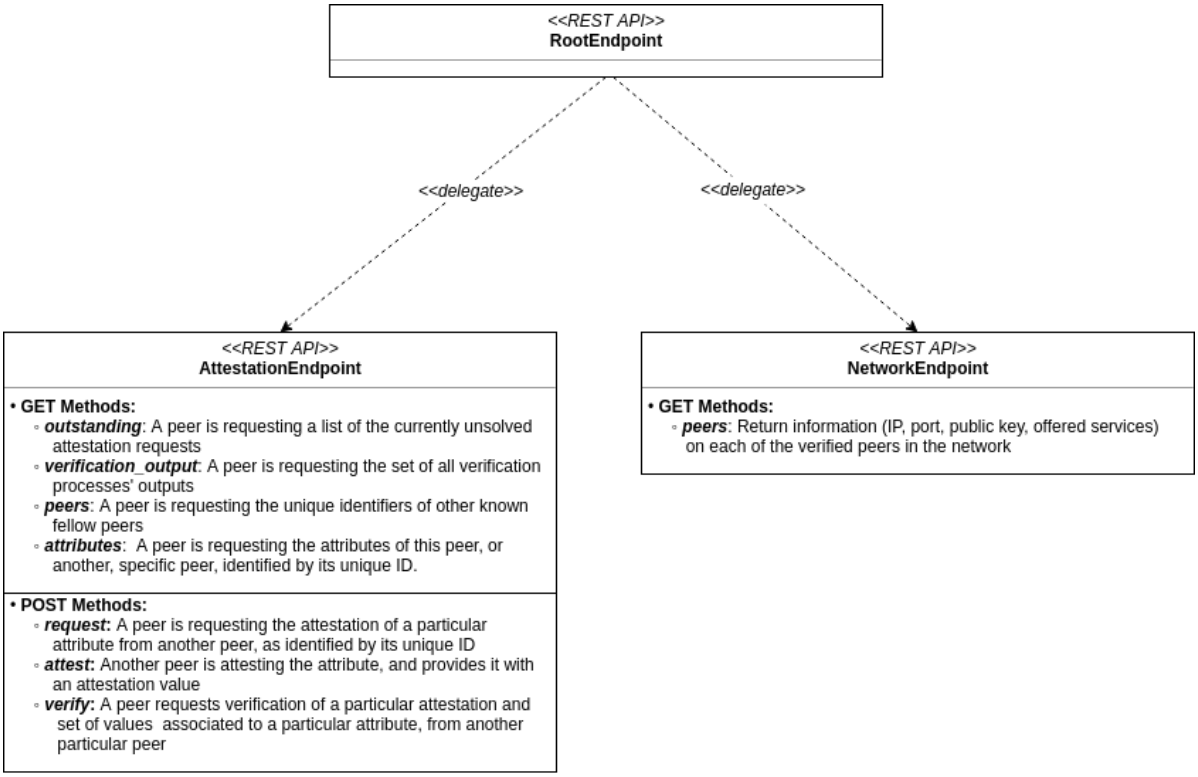


Figure 4.3: Available API for an IPv8 instance.

4.3. Mobile Application

In this section, we will discuss the technical details of what we have implemented in terms of the frontend and we will shed a little bit of light on the process. At first we used Expo as the platform for our app. However, we dropped it after finding out it was incompatible with the IPv8 library. The code on the frontend side of this product was typed in React Native, which is a relatively new tool that is widely used in the development of apps. We used TypeScript rather than JavaScript as our main scripting language. We used Visual Studio Code to edit the various screens we created.

4.3.1. React Native

We used React Native because it is an easy and flexible tool to develop apps in. We were planning on releasing a web app alongside an android app at first, and we figured React Native was the perfect tool for that. At the end, we only made the Android app and dropped the web app entirely. However, thanks to the flexibility of React Native, not much of our code had to be changed to not lose any of our progress on the web app and transfer it all to the Android app. All in all, our experience with React Native was a pleasant one.

4.3.2. TypeScript

We used TypeScript rather than JavaScript because it has more features and we thought we could make use of those features. It is also a more robust language, so we figured this was a valid reason to use TypeScript as well. Furthermore, since all of us already have some experience with JavaScript, the stiff learning curve would not be an issue. At the end of the project, we conclude that this was indeed the right choice to make. We ended up using a more robust and featureful language without sacrificing much of our time to get to know the language better. After all, it has a lot of similarities with JavaScript.

4.3.3. Files

To conclude this section, we will show the various screens we made and explain the details behind them. We will also discuss the files “App.tsx” and “Store.tsx”.

- **AttestationScreen.tsx:** The attestation screen serves as a screen where a user can find their attested proofs.
- **DashboardScreen.tsx:** The dashboard screen is the main screen of the app. Here, a user can see an overview of the proof(s) they have acquired.
- **HelpScreen.tsx:** The help screen functions as a screen which explains the various terms we use. It also explains how to use the app. Once the app has been deployed and we get questions we can add a FAQ section/page to it as well.
- **InboxScreen.tsx:** The Inbox screen contains all certificates received by this attestee, the attestee can choose whether to keep or discard (accept / decline) this data. When accepted, the data gets added to their chain.
- **LoginScreen.tsx:** In the login screen, a user can login as a health expert or as a patient. Depending on the role, the user will get access to more features.
- **NewCertificateScreen.tsx:** In the new certificate screen, a health expert can issue a new immunity proof to a patient. This is where the extra privilege depending on user role comes to play.
- **RegisterScreen.tsx:** The register screen will only be prompted once at the first startup of the app. Here, the user will create an account, either as a patient or as a health expert. Once created, the account stays until the user deletes the app.
- **SettingsScreen.tsx:** In the settings screen, the user can configure the app to their liking. For example, there are buttons to allow for push notifications and to enable/disable dark mode. The user can also delete a proof. Logging out is also an option here.
- **App.tsx:** This file serves as the main entry point for the app runtime.
- **Store.tsx:** The store contains all data types and functions related to the global state of the React app, it defines what the global state contains and what the initial values will be.

5

Ethical Implications

In this chapter we will describe the ethical facets of our project. Firstly, we will talk about the ownership of the data being created by our application. Secondly, liability and stuff. Lastly social issues and stuff

5.1. Sensitive Data

There are several clear ethical issues which we have to be wary of in the design of our product. Firstly, we are explicitly dealing with data which is very personal in its nature. Namely, health data. We have to therefore be very careful in our design with regards to the technological side of things, such as the transmission, representation and storage of this data. Secondly, the only one who should have the rights to make decisions on the data of a user should be the user themselves [30]. We should be wary that this is actually the case. Especially in this day and age, in which the collection of personal data is quite prevalent. And while there is a (perhaps justifiable) rise in collection of COVID-19 data in the interest of the public health [31, 32, 33, 34, 35], the collection of this personal COVID-19 immunity data is, in our eyes, unwarranted.

5.2. Data Ownership as a Privacy Issue

We will now discuss the identity solution of the Delft Blockchain lab a bit more and how we could use it in our design. The identity solution of the Delft Blockchain lab is based on the ten principles of a self-sovereign identity [12, 11]. Such an identity is decentralized and the ownership of this identity lies solely at the user. The user can decide for themselves what parts of their identity they want to share and what parts they want to keep hidden for the counterparty with the use of zero-knowledge proofs. This identity solution should preserve the institutional and constitutional privacy of the individual [30]. As for handling the data itself, the ideal situation would be that your identity is decentralized in the spirit of a self-sovereign identity. In this case no information of the user is stored centrally but merely by the user themselves. The transmission of this information should also be conducted securely through the channels to avoid third parties listening in. Another issue that might arise with the transmission is not knowing who is on the other line. Therefore authentication of one's identity in these transactions is very important. One could use public-private key cryptography to make sure that a message is sent by who you think it's sent by. The design of our application leans a lot on this identity library. Our application will be running the identity solution in the background and will leave the management and transmission of the data solely to the identity library. By doing so we inherit the traits which we summed up above, namely a decentralized self-sovereign identity on the phone of the user. This library furthermore uses public-private key cryptography in its communication between users and saves the data in a hashed format. Through use of the library we feel that we can alleviate the mentioned issues. While this does place a lot of trust in a library which we have not written, we feel that this trust is not misplaced. Secure transmission and saving of data is not to be taken lightly. We therefore feel like the library should do this for us instead.

5.3. Anonymous Data Transmission and Legal Liability

TODO Value sensitive design and legal liability.

5.4. Other Social Issues

We would also like to discuss ethical implications our app might have on a societal level in the long run. As a result of possible widespread use of our app or a similar one, there is a nonzero probability that society will be divided into two groups: those who do have an immunity passport and those who do not. If this happens, antagonization of those with and/or those without passports may occur [36]. This will likely result in great social stigma on people with immunity passports from those without and vice versa. While this scenario might sound exaggerated, at least a mild form of it should not be out of the question. Sadly, there is not much else we as the developers of the app can do to prevent this. Thus, we will leave it to the government to decide what to do in such a situation.

6

Product Discussion and Recommendations

In this chapter we'll talk about our process and about possible next steps like Large-Scale Deployment and extensions of the certificates.

6.1. Retrospect

Before we started to work on this project we first wanted to make a plan of what we were going to make and how we were exactly going to do that. You can find this initial project plan in Appendix C. We managed to implement all the main functionalities we wanted the application to have (all our must haves and could haves) and we even got some of our optional functionalities to work like reviewing a certificate for verification automatically. You can find our prioritization of the functionalities also in the project plan.

Even though having a good plan of what exactly you want to make is important, having a decent work division and good team work is essential to realise that plan. The first thing we did was dividing our team into a backend team, a frontend team and someone who helps out with both frontend and backend and works on the connection between them. This division instantly organized our work plan since we could now just focus on one thing only.

Besides dividing the work, communication is key. We had daily communication to make sure the frontend and backend would be compatible, to make sure we all had a working version at all times and to see if everything was running smoothly in general. We would also have close contact on a weekly basis with our client to see if we were on the right track or to see if he would want us to make some changes to the original plan. All of this was our recipe for a successful application.

6.2. Large-Scale Deployment

If we were to deploy our application on a large scale there are a few aspects we need to take into consideration and test for. We need to make sure that our application can handle a large group of users simultaneously, that the GUI is the same on every device and that people find the application easy to use.

Another important aspect to take into consideration is ethics and the effects on a societal level in the long run. As a result of widely spread use of our app or a similar one, there is a nonzero probability that society will be divided into two groups: those who do have an immunity passport and those who do not. If this happens, antagonization of those with and/or those without passports may occur. This will likely result in great social stigma on people with immunity passports from those without and vice versa. While this scenario might sound exaggerated, at least a mild form of it should not be out of the question.

6.3. More Certificates

As you'll probably know by now, our application makes it possible for people to get certificates that prove their immunity. But it has so much more potential then that. With the way we determine if someone is eligible to receive an immunity certificate we can determine so much more.

Besides immunity we can determine for example how long someone was sick or what their symptoms were. This would be very beneficial when it comes to better understanding the virus. This can, of course, also be used for other and future viruses and diseases.

The option to use the application for things other than healthcare is also a possibility. You could for example use it to determine someone's identity. That means that in theory this application could even be used as a passport.

Because of all this potential it would be a better solution to create a framework which will allow for the issuance and verification of many kinds of certificates using the attestation model implemented by the Delft Blockchain Lab.

In order to have a specific implementation of a certification service, interested stakeholders would create their own applications. This process should be a part of the application which implements the framework. The idea behind this is that for some certificates, holders will personally go to the issuer and ask for a certificate. In other cases, holders might approach issuers remotely, all possibly part of the application.

The only visible changes in the GUI when using different certificate types is in the "Drop Down Menu" in the screen where you make new certificates. You'll just find different certificates to choose from.

7

Conclusions

A

Individual Reflections

Akif

Over the course of the past 2 months, my team and I have concluded the project we set out to do and we can safely say we are proud of our results. We have developed an Android app that matters for society. At the start of the project, we had no idea what measures would be taken against the problems that were unfolding as a result of the pandemic. While businesses are starting to open and our daily lives are slowly turning back to normal, the lingering fear of a second wave is not something to be ignored. We believe our app can serve as a resource for such a situation. This is why I believe we achieved the goal of the project.

At first, the project details were very vague and it appeared as though the client himself didn't really know exactly what we were supposed to do. However, after a few weeks of communicating the details and such, we managed to finally start coding. Everything was still not clear at that moment, but we were slowly progressing towards a meaningful end product.

Throughout the project, I worked mainly on the frontend. Normally, I would work on the backend or the logic side of the product. This was the first time I was mainly working on the frontend. I have to say, it was an educational experience. I believe learning about React Native and about TypeScript is definitely useful for my potential career as a software engineer. However, I realized I would much rather take the role of a Jack of all trades: working on both the frontend and the backend and making sure they are compatible at all times. This is why I believe my strong suites have not really showed throughout the project, that is not to say I didn't do my best for my part of the project though.

As a side note, because of the troubling times we have endured in the past few months, motivating myself to keep up was very hard. At times I would really have to force myself to dish out some work I was responsible for. This led to feelings of incompetence even though I was doing my part for the project. However, I managed to endure through those setbacks and find motivation for finishing the job each time.

As for our team, I am very part to have been a part of it. Everyone was truly hardworking and it was pleasant to settle on matters related to the project with each other. We were (and still are) respectful towards each other, even if there were to be miscommunications between us. I picked up no hard feelings throughout our time working together, which made it a pleasant experience all around. Any problems that occurred were solved through civil communication and listening to what each person had to say. I'd say the star of the team was definitely Kalin, he would put in the extra hours to make sure the project would progress smoothly.

All in all, I am satisfied with how the project went. If I were to have the chance of working on another project with the same team, I would take that chance in a heartbeat. I truly believe there was good chemistry between us. It was good to get to know everyone, including our supervisor TA Wesley, our coach Taico and our client Martijn.

Kalin

For the last two months, my team and I have worked on the project ”Building a Critical Infrastructure for the Nation-Wide Identification of Recovered COVID-19 Patients”. Our client is the Delft Blockchain Lab and we were tasked with the creation of an Android application, which implements IPv8’s service, developed by the Lab. IPv8 is a library providing a Self Sovereign Identity solution.

The goal of the project is to extend the number of already developed IPv8 applications with one which provides the ability for issuance of immunity certificates. These immunity certificates are going to be used as a proof that their holders have already recovered from the COVID-19 disease. The reason behind the development of such a system is the fact that the pandemic has caused a worldwide economic crisis and in order to get back on track as soon as possible, such certificates could be used to allow immune people restart their usual daily activities.

The app is targeted at three types of users. These are Attesters, Holders and Verifiers. In particular, Attesters are the healthcare employees that are providing the wide public with tests and are essentially going to issue the immunity certificates. Holders are people that get tested and later receive an immunity certificate, if the tests prove that they have developed immunity. Lastly, the Verifiers are all users that need to know whether someone else has immunity or not. Since this system uses Self Sovereign Identities, the privacy of all Holders is preserved and they have total control over their data.

Task Related Conflicts

At the beginning of the project, we as a group decided to develop a working agreement. The idea behind its creation was to prevent any types of conflicts or at least try to prevent any conflicts. In it, we have specified rules which needed to be followed strictly by everyone. They include:

- It’s OK to disagree with each other.
- Everyone has an equal voice and valuable contribution.
- No personal attacks; we debate the merit of ideas, not people.

and many more.

I think that this agreement really helped with circumventing task related conflicts. It is true that conflicts are inevitable and we experienced some, but in my opinion the consequences from those were almost always positive. That is the case because at the beginning of the project we also discussed the personal goals of each member. In that way, we were able to distribute work more evenly and everyone worked on tasks for which they had the appropriate knowledge, skills and abilities.

Most of the times when we experienced some kind of a task related conflict, we either scheduled a brainstorming session, through which everyone’s ideas were getting heard and after some discussion we were ending up with an agreement or we scheduled a meeting with our Client, since not always our ideas matched the expectations of the Client.

Intragroup Conflicts

Again, our working agreement played a big role in preventing intragroup conflicts as well. In my opinion we did not really experience any social loafing, since we knew from the beginning how much effort every group member is willing to put into the project and we kept that in mind, every time we needed to plan any tasks or redistribute work.

Furthermore, whenever we sensed that some argument may turn into an interpersonal conflict, we agreed on leaving the discussion and picking it up again during our next meeting with either our Client or our Teaching Assistant. This generally saved us from losing time in disputes which were not necessarily focused on the project and we were able to better work as one.

Even though we tried to guarantee the smooth proceeding of the project, this is the first time we work as a group and we are not familiar with each other, meaning that there were a lot of compromises that needed to be made and not always every group member was satisfied by the outcome. But, given the fact, that we agreed to do what is best for the product and not what each of us desired, we were able to keep the dissatisfaction levels at minimum.

Intervention Techniques

Since most of the conflicts we encountered could be regarded as ”healthy”, we did not really need to use any of the intervention techniques. The work we did upfront guaranteed no major conflicts. Of course, everyone made sure to motivate and help his peers and every milestone achieved was celebrated as much as possible.

We usually had more than one way of tackling a certain issue and we always picked the one that is within reach of everyone’s capabilities whilst still keeping ourselves challenged and motivated.

Lesley

In the past couple of weeks my team and I have been working on an application that could provide health certificates to people while providing privacy and security by giving the users who request certificates total control over their data. We do this by using SSI principles which could be realized by using the Ipv8 library that was developed by our client, the Delft Blockchain Lab.

The main purpose of this project is to get the economie back on track. The pandemic that is affecting so many lives at the moment also did a number on the economy. Our hope is that the economic decline could be countered by the issuing of immunity certificates that proof that you already had the disease. Because if people can proof they are immune, they can go back to the way their life was before the pandemic which would benefit the economy greatly.

I have learned a lot while doing this project. I learned how to program in react and typescript, I learned how it is to work for a client and one of the most interesting things I learned was SSI. More specifically, the fact there exist a way to truly have an identity “online”. It seems to me that this would have a lot of potential to be used in a great deal of future technological endeavors that would drastically mitigate the misuse of personal data. this is something I am definitely going to look into further.

During the project I worked mostly on the frontend. I was content with how the frontend ended up but I feel like it didn’t reach its maximum potential. I have been in quarantaine for almost 3 months now due to the fact I’m taking care of 2 sick relatives and it was when I started to work on the frontend that I first noticed my lack of focus. It was hard for me to concentrate which was a weird experience for me since focussing on something and working on that something for a long period of time is usually my strong suite. This is why I wasn’t able to work as hard as I’m used to from myself and why I feel like that I could have done better if the circumstance were different. Nonetheless I believe that the result of my work is not below the standards of what we have planned and I believe it’s not below the standards of my team which, I am happy to say, were quite high.

before we started working on our application, we made a plan of what we wanted the application to contain exactly and we prioritized using the MoSCoW method. Out the the 4 divisions: must have, should have, could have, won’t have, we managed to get all of the must have and should have functionalities and even a few of the could haves. I’m very happy with that result but with a team like this I wouldn’t expect anything else.

My personal strengths revealed itself whenever someone needed help. I would always be available to help someone out or to help them find a fix for a certain problem. Even if it is in the middle of the night. My personal weaknesses revealed itself when I wasn’t able to focus on my work as mentioned above. It made me realize that I need to work on my adaptability to surroundings and work environments.

The next time I would do a project like this I would prefer to work on the backend. Even though I had fun designing the GUI, I feel like I can achieve a lot more by working on the backend. Not just because I find it more interesting but also because I have way more experience in working on the backend than on the frontend.

Raul
Selim
Introduction
These past couple of weeks we have worked as a project team on our project called “Building a critical infrastructure for the nation-wide identification of recovered COVID-19 (Corona) patients”. As the name suggests, we were tasked with creating an application which could create and verify immunity passports for COVID-19 in this ongoing pandemic. Our client is the Delft Blockchain Lab. The Delft Blockchain Lab has an Self-Sovereign Identity solution which they, in collaboration with the Ministry of Internal Affairs, wish to use to provide passport-grade identities to the citizens of the Netherlands. This project is a small corona check which makes uses of the underlying library of this solution, namely a library called IPv8. The underlying thought is that people who have caught the COVID-19 virus, would be immune for a second infection. Being able to identify these people and give them an immunity passport, could become an important aspect in alleviating the consequences of the pandemic. These people could, for example, be very valuable in critical industries like health care.

Task-related conflicts
I personally did not experience task-related conflicts in this project team during the past couple of weeks. The reason for this would be that in the beginning of the project, we had a couple of key meetings in which we as a team sat together to discuss a couple of points. We are all not that familiar with each other, which makes the forming stage a bit vague perhaps. In these initial meetings we however put on paper concrete points on which we agreed upon as a team. Not necessarily requirements for the project but more in the sense of how to treat each other and how to approach this project in a working agreement document. In the beginning I occasionally did take a peek at it, as having something concrete is an assuring thing. One other very important point, which I’ve noticed is done many times in group projects, is that we had a round in which we all expressed our expectations and goals for this project. 5 random students in one project group is bound to have a variation in what one would want to get out of a project. Making sure that we play open cards with these goals is a proper way to create familiarity and trust in a group I’ d say. The requirements for the project did initially create some friction I’ d say, but not necessarily conflicts. I would call it healthy friction. We were given a lot of freedom by our client to implement the application as we saw fit. While freedom is a nice thing, we eventually need to pin down our requirements. Figuring out what we as a group

Intragroup conflict
TODO

Intervention techniques
TODO Task circumplex model

B

Original Project Description

The research of the Delft Blockchain Lab orients around digital identity, cryptographic key management, verifiable claims, self-organisation, trust, and tamper-proof datastructures (e.g. blockchain). In collaboration with the Dutch National Institute for Public Health and the Environment (RIVM) and the Ministry of Internal Affairs, the Delft Blockchain Lab offers a unique project on building critical infrastructure during the Corona crisis.

The ongoing outbreak of the novel COVID-19 (Corona) virus is an unprecedented threats to humanity. The developments around the Corona virus across the world are demonstrating the fragility of current organisational models, usually involving a political model where decisions flow from top (a government) down to municipalities and individuals. One of the objectives of our lab is to explore and deploy bottom-up organizational paradigms that put citizens in power, instead of authorities. The main goal of this project is to build open-source, critical infrastructure for the identification of recovered Corona patients. The project aligns with digital companies and health initiatives around the world, like VODAN (Virus Outbreak Data Network) and the medical centers of Rotterdam and Leiden.

Problem Description

As a first step towards this goal, we seek to expand our Self-Sovereign Identity (SSI) library with health features. Delft University of Technology, together with the ministry of internal affairs developed a library for SSI that enables the creation of attestations and privacy-preserving proving of claims. Our mobile Android application has the potential to provide passport-grade identities to every citizen of the Netherlands. Specifically, this project would entail the integration of a Corona blood test where you can prove to other users (and businesses) whether you recovered from the Corona virus and therefore has built up resistance. During the coming months, quickly proving such resistance might become a key requirement by certain sectors, e.g., health care, and to better quantify the progress of herd immunity. Integration of a ”Corona check” is a small, yet critical step towards a full health record database later on, based on a Self-Sovereign Identity solution.

Self-Sovereign Identity

SSI puts people in charge of their own digital identities. The key idea is that companies can provide attestations to the digital identities of users. Users can then use these attestations in interactions with others to prove various claims, e.g., whether they have the legal age to buy alcoholic drinks from stores, without revealing sensitive information. Since 2016, our research line on SSI has been covered by several media outlets and currently, the third generation of prototypes is being defined together with the RvIG.

C

Initial Project Plan

COVID-19, the disease caused by SARS-CoV-2, has affected the entire globe [1]. The negative impact is not only felt by everyone that is currently infected and the healthcare personnel, but the entire world economy suffers the consequences of the disease [2]. Countries around the world have tried several strategies, from informing the public on the virus, enforcing curfews, intensive contact tracing of infected individuals and many more [3].

The decisions taken by governments, which might turn out to be even more harmful than the pandemic itself, are one of the reasons people are witnessing the forming of a global economic crisis. The imposed lockdowns and measures to protect the public from the virus are changing the way our society behaves, which forces many owners to cease their business activities temporarily and in some cases permanently.

As a result, the way cash flows throughout the economy shifts and many people end up struggling with their monthly bills or even losing their jobs. In order to recover from this inevitable crisis, governments have to take into account all trade-offs between rescuing the economy and public safety. This would lead to the imposition of new regulations and rules that people and businesses will be forced to follow.

One of the main ideas is to introduce “immunity passports” and issue them to people that are considered immune to the virus [4]. People who possess them will be allowed to again live their life normally and engage in many sorts of activities, such as working or studying. The main goal is to protect people that are still not resilient to the virus, whilst reopening businesses and providing support for everyone in need. Through the use of immunity certificates, the recovery of the economy is expected to be easier and faster. But, in order for this idea to succeed there has to be a nation-wide testing campaign, which is still impossible, considering the insufficient number of available tests.

The aim of this project is to materialize such an immunity certificate. Delft Blockchain Lab has made this project available, because they wish to extend their current Self-Sovereign Identity library with support for “immunity passports”.

The project plan is built up in the following structure. Chapter 1 will provide some deeper background information. Chapter 2 will go into the technical, legal and operational feasibility of the project, given the duration of this project. Chapter 3 will explain any risks which might be faced during or after the project. Chapter 4 lists the requirements that need to be realized during the project. Chapter 5 describes the team’s communication and workflow strategies. And lastly, chapter 6 contains a Roadmap for the coming weeks.

C.1. Problem Analysis

The project plan begins with background information about the problem. Through the use of reduction techniques, the problem was reduced to deploying a nation-wide system which implements Self-Sovereign Identification.

Building Immunity

Usually, immunity to a disease is proven through a test, which checks for the presence of antibodies associated with this disease inside a person's organism. However, since this is a novel virus, one of the main problems that needs to be taken into consideration is the fact that scientists have not yet proven that the presence of antibodies in someone's organism is solid evidence that this person is immune to SARS-CoV-2 [5].

Although repeated infections might turn out to be possible, scientists think that this is highly unlikely and it is generally believed that former patients of COVID-19, that test positive when checked for antibodies, have built immunity [6]. Considering the importance of the project and the fact that the virus is still spreading widely, the possibility of reinfections will not be regarded as a possible hurdle. When the duration of the immunity is later defined, necessary corrections will be applied.

Immunity Passports

The problem with lifting the imposed emergency regulations might be efficiently solved through the use of immunity certificates. They effectively provide people with a "permit" for their usual economic and social activities. Immune people will be extremely valuable in the months after the pandemic is over, because there is always a possibility of a second pandemic, which might have worse consequences than the first one. "Immunity passport" holders may be used as a resource. This would mean that these people could be sent to infection hotspots in order to provide critical support [6].

Unfortunately, immunity certificates need a lot of consideration before deploying them. Since only people that have previously been infected with the virus will become immune and thus have the right to work and ultimately earn money, there will be a majority of people being left behind, possibly without access to food and other supplies. This could be considered as a new kind of discrimination, where the population is split in immune and vulnerable, with the latter having restricted rights. At some point they might try to catch the virus on purpose in order to become immune and get access to their full rights again. Another problem is the duration of the immunity. If everyone has to be regularly tested, there will not be enough tests for everyone [36].

There are several ways in which one could grant such an "immunity certificate". A mere certificate on paper could suffice, however this brings with itself several issues of authentication and authenticity [7]. Furthermore, immunity certificates pose a privacy issue because by using them, people are forced to share their health condition, which might end up in misuse of sensitive information. The "passports" also might make use of mobile technology, which will further exclude parts of the population, currently not having access to such technologies.

Nevertheless, there are already more than 60 companies working on defining immunity certification. Most of them are using blockchain technology, as it provides some of the required privacy and security properties [8].

Privacy and Security

As discussed above, there is a risk of personal information misuse cases, which poses an enormous privacy threat. This might mean that people are less likely to trust the technology and they might choose not to use it, which will make it obsolete. Most of the companies that are working towards a solution, use blockchain technologies to address these caveats. Historically, Bitcoin was created to allow people to have full control over their information, ultimately trying to solve the privacy and security issues. But, since 2009 there have been many cases in which personal privacy was breached [9].

Recently, the World Wide Web Consortium released a new standard called "Verifiable Credentials Data Model", which claims to provide security and anonymity to its users. This model allows users to prove each other claims about themselves without revealing sensitive information [10]. The problem that remains is that such a system is not widely deployed yet.

Self-Sovereign Identity

One approach which has gained a lot of traction recently is the usage of a Self-Sovereign Identity (SSI). This is an implementation of the “Verifiable Credentials Data Model”, discussed above. The idea behind SSI is that citizens have full control over their own identity. Instead of the government issuing identification documents, every person could claim information about himself voluntarily and organizations could then attest these claims if they are correct. For example, holders of SSI could claim that they are above a certain age and request a responsible organization to attest this claim after which the holders could use this proof by showing the appropriate attestation without revealing any other sensitive information about themselves (e.g. exact age or date of birth) [11].

Client

The task set is to combine these two topics, namely the granting of an immunity certificate through the use of a Self-Sovereign Identity. The client, Delft Blockchain Lab, is TU Delft’s initiative for research, education, and training in blockchain technology and trust in the internet. In cooperation with the Ministry of the Interior and Kingdom Relations it has developed a library for SSI that is set to provide passport-grade identities in the future [13]. The outcome of this project is adding the COVID-19 “immunity passport” feature to the library as it could prove very useful in the coming months. Furthermore, if this project is successful and ends up becoming a nation-wide deployed system, it could possibly be the first step towards mass adoption of Self-Sovereign Identity.

C.2. Feasibility Study

The issue of user privacy and data sovereignty is one of the most hotly debated topics of the 21st century. In a world where big tech companies such as Google and Facebook are having to pay multi-billion euro settlements, research shows that the world is looking for change [37].

The project aims to further popularise a solution to this pressing issue by implementing a user friendly application that abides by the well tested and researched SSI principles, which provide user privacy by placing individuals in control of their data. All projects using innovative technologies give place to several doubts. Can this work in the real world? Will uncertainty challenge on-time completion? Is such a product even possible? This section aims to answer these questions.

Legal Feasibility

When dealing with user data, privacy protection laws are a major concern but there is one advantage to this project, it is all about user privacy and data protection.

The purpose of this project is to showcase the usability of the SSI principles, which are designed to protect user data by giving them ownership of their information. By placing the sensitive data in the hands of the owner most privacy concerns, which relate to a centralized data point owned by a third party, are circumvented.

By using the IPv8 [13] framework and TrustChain [29] as the backbone of the data transfer and validation processes, the hard work of dozens of researchers and specialists is going to be put into use to guarantee safe data transfer and integrity. Safe storage combined with safe data transfer allows the application to be secure and legally safe.

Operational Feasibility

Another important point is whether or not the product would be attractive and practical for the common user. It will take the form of a mobile application, which is a tried and tested platform to get users engaged with a product, it is also a platform that many people are familiar with which improves ease of use.

In terms of practicality the numbers do not lie. So many court cases, debates, legal settlements and outright protests have been conducted on the base of data ownership that there is little room to deny that individuals are looking for more control over their data. People do not trust big corporations anymore, they want control and knowledge of where their data is and who can access it, and such is their right based on article 12 of the UN charter of human rights [38].

SSI provides what people want in an intuitive way, they own their data and only they can give access to it. The application is expected to become the first SSI experience for many users, and allow citizens to acquaint themselves with the concept of owning their internet identity. The concept of proving COVID-19 immunity is an isolated and privacy-sensitive issue, its sensitivity supports the choice of SSI implementation, while its isolation allows us to create a useful application within the allocated time.

Scheduling and Technology

New technologies lead to new problems, and new problems lead to unpredictable solution times, so how can project completion within ten weeks be guaranteed? The project will involve some relatively new technologies including blockchain and decentralized data ownership. Building all of these systems would greatly extend the project lifetime, but the work is simplified by making use of many systems already in place.

The IPv8 framework will be used to handle most of the complications concerning decentralization, SSI and data integrity. This project is also heavily inspired by other health-related SSI systems [39] as a guide in what components the project will involve, reducing design time. The front-end is going to be built as a mobile application, which has many tools to quickly prototype and deploy the product. While most of the complications reside in the back-end, they are handled by external frameworks with several times the development time that this project will have, allowing for achieving more in less time.

By focusing solely on the COVID-19 immunity verification pipeline, using the SSI principles to handle user privacy, and using well established frameworks and technologies wherever possible, ten weeks of development time look achievable, this is also backed by the Roadmap which sets out exactly how those weeks are going to be spent.

C.3. Risk Analysis

The integration of an “immunity passport” feature on top of a Self-Sovereign Identity solution does not come without difficulties. A discussion follows about the risk factors involved in the large-scale deployment of “immunity passports” and their use, concerning both the individual and the public.

Privacy and Security Concerns

First of all, the privacy and security aspects of the product need to be considered. The immunity certificate support is going to be built as an extension to a privacy-preserving and secure SSI framework. However, failure to convince the potential user of the safety with regard to privacy may cause the product to not be used. As these technologies are rather new, scepticism by the general public is not out of the question. Thus, the final product should be as user-friendly as possible.

Nonexistent Immunity

At this point in the pandemic, there is not enough evidence about the effectiveness of antibody-mediated immunity to guarantee the accuracy of an “immunity passport” or “risk-free certificate” for COVID-19 [5]. From this, it can be concluded that the immunity passport may thus induce a false sense of security to the relatively carefree holder. People possessing such passports, now wrongfully thinking they are immune, attend social activities, such as going to work or to meet-ups, while being ignorant to the fact that they might be exposing themselves and those around them to great danger. Therefore, the immunity passport may have the unintended side effect of increasing the number of active cases in the areas it is introduced to.

Expiration of the Passport

It is unknown how long a person can keep the antibodies in their body after recovering from COVID-19 [40]. So, even if a person is declared immune and gets the immunity passport they are officially eligible for, it is not certain until when those passports should be valid. This might cause a second wave of infections with novel SARS-CoV-2, as people who might no longer be immune will use their passports to gain access to social activities.

However, the coronavirus causing COVID□19 has not rapidly mutated so far, which could indicate that immunity confers long□term protection [6]. This means that immunity passports may actually last for as long as is necessary. Besides, an arbitrary expiration date to the passport could be set, which will require the user to again undergo the procedure of acquiring a valid “immunity passport”. As a result, the risk of a susceptible individual to be allowed back in society will significantly decrease.

Purposeful Infection

The privileges the immunity passport grants to its holders may prove attractive to those who have not been infected by SARS-CoV-2. As a result, people may want to purposefully contract this coronavirus for a chance to gain immunity and with that eligibility for an immunity passport, which in turn will allow them to be able to return to their normal daily lives. Thus, immunity passports could again increase the number of infections wherever they get introduced.

To combat this issue, people who have never contracted the virus must be convinced that they are not gaining anything by purposely getting infected. Specifically, they only need to realize that their lives severely outweigh the few privileges the immunity passport can grant them, and that immunity passports are only useful for a specific group of people.

Library Incompatibility

The library that is going to be used for implementing the certification service is still in development. Thus, always using the most recent version of it might cause severe incompatibility issues which would break the application. To combat the risk of an unusable application as a result of incompatibility with the IPv8 library, a snapshot of the library in its current state could be used. However, if there are updates to this library, this snapshot would need to be updated.

Feasibility of Design

If for some reason (either time constraint, or technological constraints), delivering all requirements is impossible, there might either be a reduction of the requirements or the product might be left in such a state that it can still be improved upon in the future, while making sure that the application is still presentable in the current state. In any case, the client needs to be updated on the overall progress as regularly as possible, in order to reduce probability of failure.

Deviations During the Project

The application was designed with the conducted research and the development time frame available in mind. Nonetheless, considering that all information collected from the research is quite new, during the project it might turn out that parts of this information were not adequate and that the product's design and/or requirements might therefore need to change. At that point, additional research and a talk with the client should be done. Based on that, the project should be updated accordingly and all supervisors should be notified.

C.4. Requirements and Solution Proposal

During the process of requirements engineering, the team made use of two scientific papers, which describe a possible implementation of an “immunity passport” [41] and some properties that such an application should hold [42]. There were also two meetings with the client, which tremendously helped with narrowing down the list of specific requirements that need to be implemented before the end of the project. This chapter begins with a summary of all requirements. Detailed information about each requirement follows in the section “Elicitation and Analysis of the Requirements”. The chapter concludes with some application restrictions and the solution proposal.

Functional Requirements

Must Haves

- Attesters can create certificates
- Attesters can send certificates to corresponding holders via their key identifier
- Holder can view their certificates
- Holder can accept or decline certificates
- Holders can present their certificates to verifiers
- Verifier can review certificate contents
- Verifier can verify the certificate signature

Should Haves

- Application configuration and authentication
- Notifications about certificates

Could Haves

- Certificate time-out
- Receiving an old certificate
- Reviewing a certificate for verification automatically
- Creating new types of certificates for other purposes (not just COVID-19)

Won’t Haves

- Questionnaire for determining whether a person has COVID-19
- iOS support, since the process of deployment does not suit the project time frame
- Method for requesting certificates

Non-Functional Requirements

- IPv8 will be used as the attestation service. Included in it is TrustChain, which is going to be used as the blockchain instance, taking care of storing all holder’s certificates.
- Since IPv8 is written in Python, a packaging tool for Python applications on Android will be used. The available tools are: BeeWare [16], Chaquopy [15] and Kivy [17].
- The GUI will be implemented using React [43] and Expo[44].
- The application is going to run on Android OS versions 7+.

Elicitation and Analysis of the Requirements

The following section depicts the requirements analysis process of the team. This analysis was conducted through the use of user stories.

Three main roles were introduced:

- Holder - The role of an entity that possesses a Self-Sovereign Identity.
- Issuer - The role of an entity that issues certificates.
- Verifier - The role of an entity that verifies certificates.

Throughout this section, the term certificate regards to any kind of certificate (e.g. immunity certificate). In every user story, a certificate is going to be implemented by either a claim or an attestation in terms of IPv8. A claim is some signed information by a user who claimed it about himself or someone else. All claims need to be sent to an attester for validation. An attester is the entity which validates the truthfulness of the claim. And an attestation is a verified and signed claim by an attester. After a holder receives an attestation, he is free to use it for verification when requested by a verifier.

User Story 1: “Application startup before configuration”

- When any of the three users wants to start using the application, they must be able to open the application and if this is their first usage, they must be prompted to complete a setup process. It must contain a password configuration and local blockchain instance initialization (which happens automatically). Then the main menu must appear. There is a need for a local blockchain instance, since all users need a storage for their certificates.

User Story 2: “Application startup after configuration”

- When any of the three users wants to start using the application, they must be able to open the application and if this is not their first usage, they must be prompted to enter their password. Then the main menu must appear.

User Story 3: “Certificate creation”

- After a holder has requested a certificate, an issuer must be able to pick the correct type of certificate from a list. He must then be able to fill in the contents of the certificate. All certificates must be in user-readable format (e.g. json could be used to allow screen renders of the certificate for manual reviews).
- Since the holder does not need to know the format of the certificate, but he needs to be able to review the information contained in it manually, a simple data model that will allow this is going to be used. Also, the application does not solve the problem of certificate request. For more information look at “Solution Proposal”.

User Story 4: “Sending a certificate”

- After an issuer learns the key identifier of a holder and creates a certificate, he must be able to sign it, save it to his blockchain and send it to the holder using his key.
- The key identifier of a holder is just his address, where his other certificates reside, as defined by IPv8. They will be shared through the use of QR codes or some network. All blocks of information that get transferred between entities must be signed. As implemented in TrustChain, valid blocks are the ones who contain at least two signatures. One from the holder and one from the issuer are required.

User Story 5: “Receiving a certificate”

- When a holder receives a certificate for review, he must be able to see it in his inbox.
- Holders could review certificates since this is one of the properties provided by SSI [11].. They are allowed to either accept or decline a certificate.

- The inbox is where all requests will reside for all users. It is going to be implemented as a separate list that could be accessed through the main menu.
- When an issuer receives a signed certificate that is on his blockchain, he must be able to verify the certificate.

User Story 6: “Certificate review”

- After a holder receives a certificate, he must be able to access it through his inbox and manually review its contents. Then he must be able to decide whether to accept the certificate or not.

User Story 7: “Certificate acceptation”

- After a holder has reviewed a certificate and has decided to accept it, he must be able to sign it, save it to his blockchain and send it to the issuer who originally created the certificate.

User Story 8: “Certificate decline”

- After a holder has reviewed a certificate and has decided to decline it, he must be able to remove it from his inbox. He does not need to notify anyone about his decision.

User Story 9: “Certificate time out”

- After a certain amount of time, an issuer should be able to forget an outstanding certificate if it is not signed and returned by its holder.

User Story 10: “Receiving an old certificate”

- When an issuer receives a certificate that is not on his blockchain anymore, he should notify the holder that this certificate is no longer valid.

User Story 11: “Notify issuer and holder on certificate validity”

- When a certificate is received both the issuer and the holder get a notification.
- The content of the notification depends on the certificate validity.

User Story 12: “Showing certificates for verification”

- A holder must be able to choose from a list of all his certificates, the one he needs for verification. After picking it, the certificate is shown in appropriate format for verification (e.g QR code). Certificates may also be sent using Bluetooth or other networks.
- A holder must have a list of all his certificates. He must be able to click on any of them and the user-friendly render of the certificate must show up. Then the holder must be able to choose how he is going to use his certificate for verification. Thus, the holder must be at least able to show the certificate as a QR code. Other methods of verification could also be possible (e.g. sharing through Bluetooth or another network).

User Story 13: “Reviewing a certificate for verification”

- A verifier must be able to enter into a state for verification There may be different states, depending on the medium that is going to be used for the verification. For example, if the holder presents the certificate using a QR code, the verifier must be able to use his camera for verification. After scanning the certificate, the verifier must be able to manually review it.

User Story 14: “Reviewing a certificate for verification automatically”

- The verifier is able to pick the type of certificate they are going to verify in advance. Once selected, verifying any certificate of that type will be done automatically, without showing the certificate in user readable format.

User Story 15: “Verifying a certificate”

- After reviewing the certificate either automatically or manually, both parties must get notified by either showing a pass or fail notification.

Application Restrictions

Because there are some restrictions posed by the bill [42], some additional features are being considered. If the project duration is not sufficient to implement them, they are going to be listed as recommendations in the final report of the project.

First, there is a portion of the citizens, which are considered technologically excluded. They either do not have access to devices with internet connection or they do not possess sufficient knowledge to handle such devices. In this case, there might be a feature which allows for the issuance and usage of paper certificates.

Second, holders which are not carrying their device at all times or which device is not working, should not be penalised. For this case, a web application is being considered, which would allow for holders to remotely use their certificates. Furthermore, another possible solution is the introduction of "smart bracelets" which are going to contain their owner's certificates. This will allow for verification on demand.

Solution Proposal

The problem analysis has shown that "immunity passports" might turn out to be obsolete if implemented now, since there is no proof that people build immunity against COVID-19. The specific regulations about those certificates will also become available only if immunity exists. That is why the solution should not solely focus on implementing "immunity passports".

A better solution is to create a framework which will allow for the issuance and verification of many kinds of certificates. The idea is to provide a framework that makes use of the attestation model implemented by the Delft Blockchain Lab. The main feature of this framework is the support for any kind of certificate.

In order to have a specific implementation of a certification service, interested stakeholders would create their own applications. That is why earlier the process of establishing communication between a holder and an issuer or a verifier was vaguely described. This process should be a part of the application which implements the framework. The idea behind this is that for some certificates, holders will personally go to the issuer and ask for a certificate. In other cases, holders might approach issuers over the phone, Internet or through a video calling service, all possibly part of the application.

As a proof of concept, an Android application is going to be delivered which will allow for the creation, attestation and verification of immunity certificates. This fulfills the needs of the client. It will be assumed that patients will have to personally go to a healthcare expert in order to get tested and receive a certificate. That is why the process of requesting an immunity certificate will not be implemented. If later the government approves home testing, the application might be extended with a video calling service in order to provide supervision, which is just an implementation of the certificate requesting process.

C.5. Project Approach

In the first week of the project there was a long meeting with regards to the approach to this project. In this meeting a working agreement was set up, in which several guidelines that need to be followed were outlined. As most of the group members are not familiar with each other or with how well they can work in a team, a working agreement appeared to be appropriate.

This project has several supervisors. These are the client, the coach, the TA and, if necessary, the Software Project coordinators.

The contact between the group and the client happens on a weekly basis. The client is expecting questions and also posing ones, in order to gauge the progress. The medium used for this meeting is the Discord voice chat. For small inquiries the client may be asked questions by sending messages through Discord.

The communication with the coach happens at certain points in the project. Those meetings are done through Jitsi. The coach will give feedback during these different points throughout the project. For general inquiries the TA is the first point of contact. He should be asked first instead of the coach.

The contact with the TA happens on a weekly basis. This meeting also goes through Jitsi. Furthermore there is a shared Mattermost channel with the group, TA and coach, where small questions at any point in time may be asked. The TA is the first point of contact for day-to-day guidance.

The team itself has two means of communication. There is a WhatsApp group and a Mattermost channel. The Mattermost channel is the primary means of communication. The WhatsApp group is only used in case of heads up if there is an absentee in the Mattermost channel. The Mattermost channel is used to discuss anything in regards to the project.

Lastly, if required, there is a public Mattermost channel for all students in the Software Project to ask their questions regarding the Software Project to the Software Project coordinators.

C.6. Roadmap

This Roadmap is a general guideline of the topics the group will focus on during each week. It is both a planning tool and a feasibility check to see how work could be divided into appropriate time slots. It is not a strict project schedule and is subject to change based on unforeseen problems or a change in project direction or scope under the client's decision. This can be accommodated thanks to the use of the agile SCRUM process.

The Roadmap predicts two main lines of work:

- Front-end: focused on designing the interface and making the correct calls to the back-end.
- Back-end: focused on implementing immunity passports through the use of the Delft Blockchain Lab's own SSI framework.

The prediction is that the back-end section will take significantly more effort for implementing most features, it will also receive more developers if required.

The front-end will start with a functionality focused approach, so the application's visual appeal will only be a focus later in development. This allows for prioritization of a functional product, keeping the front-end ahead of the back-end, and guaranteeing that time is only invested in styling UI components that are actually necessary in the final product.

Week 3: 04/05 - 10/05

- Front-end: UI Design.
- Back-end: Start work on packaging the Python framework for Android. Project setup: repository, pipeline, dependencies, continuous integration. Basic mobile application (no SSI functionality).

Week 4: 11/05 - 17/05

- Front-end: Implement all UI components in design for joining the network.
- Back-end: Finish pipeline for joining the network. Incorporate SSI into mobile application (through some form of interaction with the back-end).

Week 5: 18/05 - 24/05

- Front-end: Implement UI components for adding and sending immunity proofs.
- Back-end: Start work on proof creation by professionals and transmission to user.

Week 6: 25/05 - 31/05

- Front-end: Implement UI components for sending proofs to other users (e.g., QR code generator + reader).
- Back-end: Finish implementing immunity proof creation and transmission.

Week 7: 01/06 - 07/06

- Front-end: Overall styling, debugging and quality, validation and system checks.
- Back-end: Finish establishing, debugging and testing proof pipeline.

Week 8: 08/06 - 14/06

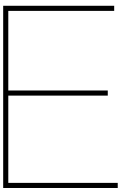
- Buffer week for unpredicted delays, requirements, problems. Overall quality, validation and system checks, debugging and report writing.

Week 9: 15/06 - 21/06

- Overall quality, validation and system checks, debugging and report writing.

D

Info Sheet



General Division of Labor

Bibliography

[1] David Heymann and Nahoko Shindo. “COVID-19: what is next for public health?” In: *Lancet* 395.10224 (Feb. 2020). Accessed: 2020-4-25, pp. 542–545. ISSN: 0140-6736. DOI: 10.1016/S0140-6736(20)30374-3.

[2] Netherlands Bureau for Economic Policy Analysis. *Corona crisis scenarios*. Accessed: 2020-4-29. Mar. 2020. URL: https://www.cpb.nl/en/corona-crisis-scenarios#.

[3] Janice Tanne et al. “Covid-19: how doctors and healthcare systems are tackling coronavirus worldwide”. en. In: *BMJ* 368 (Mar. 2020). Accessed: 2020-4-25. ISSN: 0959-8138. DOI: https://doi.org/10.1136/bmj.m1090.

[4] Jason Horowitz. “In Italy, Going Back to Work May Depend on Having the Right Antibodies”. In: (Apr. 2020). Accessed: 2020-4-25. URL: https://www.nytimes.com/2020/04/04/world/europe/italy-coronavirus-antibodies.html.

[5] *”Immunity passports” in the context of COVID-19*. Accessed: 2020-4-26. Apr. 2020. URL: https://www.who.int/news-room/commentaries/detail/immunity-passports-in-the-context-of-covid-19.

[6] R Eichenberger et al. *Certified Coronavirus Immunity as a Resource and Strategy to Cope with Pandemic Costs*. Accessed: 2020-4-25. Apr. 2020. DOI: 10.1111/kykl.12227.

[7] Laura Smith-Spark. *Is this how to get out of lockdown?* Accessed: 2020-4-29. Apr. 2020. URL: https://www.cnn.com/2020/04/03/health/immunity-passport-coronavirus-lockdown-intl/index.html.

[8] Ian Allison. *COVID-19 ‘Immunity Passport’ Unites 60 Firms on Self-Sovereign ID Project - CoinDesk*. Accessed: 2020-4-26. Apr. 2020. URL: https://www.coindesk.com/covid-19-immunity-passport-unites-60-firms-on-self-sovereign-id-project.

[9] Arvind Narayanan et al. *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*. en. Accessed: 2020-5-2. Princeton University Press, July 2016.

[10] Many Sporny, Dave Longley, and David Chadwick. *Verifiable Credentials Data Model 1.0*. Accessed: 2020-4-25. Nov. 2019. URL: https://www.w3.org/TR/vc-data-model/.

[11] Quinten Stokkink and Johan Pouwelse. “Deployment of a Blockchain-Based Self-Sovereign Identity”. In: Accessed: 2020-4-25. IEEE, July 2018, pp. 1336–1342. DOI: 10.1109/Cybermatics_2018.2018.00230.

[12] Christopher Allen. *Life With Alacrity*. Apr. 2016. URL: http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html.

[13] Johan Pouwelse. *Blockchain-based identity with government support*. Accessed: 2020-4-26. URL: https://www.blockchain-lab.org/trust/.

[14] *Services overview | Android Developers*. Accessed: 2020-6-9. URL: https://developer.android.com/guide/components/services.

[15] *Chaquopy*. Accessed: 2020-5-5. URL: https://chaquo.com/chaquopy.

[16] *BeeWare*. Accessed: 2020-5-5. URL: https://beeware.org.

[17] *Kivy: Cross-platform Python Framework for NUI*. Accessed: 2020-5-5. URL: http://kivy.org/.

[18] *Sodium*. Accessed: 2020-6-9. URL: https://doc.libsodium.org.

[19] *libsodium-jni*. Accessed: 2020-6-9. URL: https://github.com/joshjdevl/libsodium-jni.

[20] *Android ABIs | Android NDK | Android Developers*. Accessed: 2020-6-9. URL: https://developer.android.com/ndk/guides/abis.

[21] Mishaal Rahman. *How to find the Android Version Distribution statistics in Android Studio*. Accessed: 2020-6-9. Apr. 2020. URL: https://www.xda-developers.com/android-version-distribution-statistics-android-studio/.

[22] *Application Fundamentals | Android Developers*. Accessed: 2020-6-9. URL: https://developer.android.com/guide/components/fundamentals.

[23] *Background Execution Limits | Android Developers*. Accessed: 2020-6-9. URL: https://developer.android.com/about/versions/oreo/background.

[24] *Processes and Application Lifecycle | Android Developers*. Accessed: 2020-6-9. URL: https://developer.android.com/guide/components/activities/process-lifecycle.

[25] *Alternative distribution options | Google Play | Android Developers*. Accessed: 2020-6-9. URL: https://developer.android.com/distribute/marketing-tools/alternative-distribution.

[26] *App Widgets Overview | Android Developers*. Accessed: 2020-6-9. URL: https://developer.android.com/guide/topics/appwidgets/overview.

[27] Tribler. *Tribler/py-ipv8*. Accessed: 2020-5-5. URL: https://github.com/Tribler/py-ipv8.

[28] *IPv8 Documentation*. 2019. URL: https://py-ipv8.readthedocs.io/en/latest/.

[29] Pim Otte, Martijn de Vos, and Johan Pouwelse. "TrustChain: A Sybil-resistant scalable blockchain". In: *Future Gener. Comput. Syst.* 107 (June 2020). Accessed: 2020-4-25, pp. 770–780. ISSN: 0167-739X. DOI: 10.1016/j.future.2017.08.048.

[30] Jeroen van den Hoven et al. *Privacy and Information Technology*. Ed. by Edward N. Zalta. 2020. URL: https://plato-stanford-edu.tudelft.idm.oclc.org/archives/sum2020/entries/it-privacy/ (visited on 05/16/2020).

[31] Jeremy Cliffe. *The rise of the bio-surveillance state*. Mar. 2020. URL: https://www.newstatesman.com/science-tech/2020/03/rise-bio-surveillance-state.

[32] Alfred Ng. *The coronavirus pandemic is changing how your privacy is protected*. URL: https://www.cnet.com/news/coronavirus-pandemic-changes-how-your-privacy-is-protected.

[33] Uptin Saiidi. *Hong Kong is putting electronic wristbands on arriving passengers to enforce coronavirus quarantine*. Mar. 2020. URL: https://www.cnbc.com/2020/03/18/hong-kong-uses-electronic-wristbands-to-enforce-coronavirus-quarantine.html.

[34] Natasha Singer and Choe Sang-hun. *As Coronavirus Surveillance Escalates, Personal Privacy Plummets*. Mar. 2020. URL: https://www.nytimes.com/2020/03/23/technology/coronavirus-surveillance-tracking-privacy.html.

[35] Aaron Holmes. *The CDC will set up a coronavirus surveillance and data collection system as part of the $2 trillion stimulus bill, which President Trump just signed into law*. Mar. 2020. URL: https://www.businessinsider.com/cdc-coronavirus-surveillance-and-data-collection-stimulus-package-2020-3.

[36] Stephanie Baker and Erik Larson. *The Problem With Immunity Certificates*. Accessed: 2020-4-26. Apr. 2020. URL: https://www.bloomberg.com/news/articles/2020-04-09/there-s-a-big-problem-with-coronavirus-immunity-certificates.

[37] Brooke Auxier et al. *Americans and Privacy: Concerned, Confused and Feeling Lack of Control Over Their Personal Information*. Accessed: 2020-5-2. Nov. 2019. URL: https://www.pewresearch.org/internet/2019/11/15/americans-and-privacy-concerned-confused-and-feeling-lack-of-control-over-their-personal-information/.

[38] *Universal Declaration of Human Rights*. Accessed: 2020-4-30. Oct. 2015. URL: https://www.un.org/en/universal-declaration-human-rights/.

[39] Xueping Liang et al. *Integrating blockchain for data sharing and collaboration in mobile healthcare applications*. Accessed: 2020-4-25. Oct. 2017. DOI: 10.1109/PIMRC.2017.8292361.

[40] Hillary Leung. *What to Know About Coronavirus Immunity and Chances of Reinfection*. Accessed: 2020-5-2. Apr. 2020. URL: https://time.com/5810454/coronavirus-immunity-reinfection/.

[41] Marc Eisenstadt et al. *COVID-19 Antibody Test Certification: There's an app for that*. Accessed: 2020-4-27. Apr. 2020. URL: https://arxiv.org/abs/2004.07376.

[42] Lilian Edwards et al. *The Coronavirus (Safeguards) Bill 2020: Proposed protections for digital interventions and in relation to immunity certificates*. Accessed: 2020-4-25. Apr. 2020. DOI: 10.31228/osf.io/yc6xu.

[43] *React*. Accessed: 2020-5-5. URL: https://reactjs.org/.

[44] *Expo*. Accessed: 2020-5-5. URL: https://expo.io.