BACHELOR OF SCIENCE WITH HONOURS IN COMPUTING SCIENCE / CYBER SECURITY

Final Year Project Report

DIVERSE: Zero-trust VPN Service

Report by

Heetak Yang

Supervisor

Ankit Saurabh

Date

25 February 2022

DECLARATION STATEMENT

I certify that the work submitted is my own and that any material derived or quoted from the published or unpublished work of other persons has been duly acknowledged.

Student Full Name: Yang, Heetak

Student Registration Number: 11059972

Signed: ……………………………………………………

Date: 25 February 2022

ABSTRACT

As the 4th industrial revolution, the IT field and environment are getting huge and growing rapidly, while COVID-19 creates a pandemic, and work has increased rapidly in telecommuting and online environments. As a result, the company has built a VPN, but in a fast IT growth environment and hastily built VPN can be vulnerable to security, and many black hackers aim for it or use the vulnerability. To secure this, I focused on making it easy for a variety of individuals/businesses of any size to easily and conveniently build a secure VPN. In addition, by using the next-generation VPN, WireGuard, the data transmission speed is increased to build a service suitable for a safe and fast modern Internet environment, furthermore, by combining the zero-trust model, it can protect and use clients and services more safely as a part of the zero-trust platform. I propose and study existing models, and additionally develop applications proposed through a project called Annectant.

ACKNOWLEDGEMENTS

It must be an exceedingly difficult project for a student to do alone. Nevertheless, many security-related people around helped.
First, I would like to thank Ankit Saurabh for guiding me in confirming and explaining the project. This gave me the confidence to plan and start production.
I would also like to thank Jonghun Kim, who is working as a security expert at KISA (Korea Internet and Security Agency). He gave a briefing about the Zero Trust concept and numerous examples and current situations that are currently being implemented and operated in Korea and helped in planning.
I would also like to thank Coventry University members for their diverse and varied teachings so far.

TABLE OF CONTENTS

LIST OF FIGURES

GLOSSARY

- **Zero-trust model**: The concept of preparing for hacking by distributing authority to authenticated identities through an authentication process without trusting anyone. Zero trust architecture (ZTA) is an enterprise's cybersecurity plan that utilizes zero trust concepts and encompasses component relationships, workflow planning, and access policies (NIST 2020).

- **VPN**: Abbreviation for Virtual Privacy Network, it is a private network used by a company or several groups to communicate through a public network without revealing the contents to outsiders. In a virtual private network, a message is transmitted using a standard protocol over a public network such as the Internet, or through a private network of a service provider after a service level agreement is reached between a virtual private network service provider and a customer.

- **OpenVPN**: The most representative VPN protocol provided as an open-source.

- **WireGuard:** A next-generation VPN protocol designed to increase speed and security.

- **Annecent:** Attachment is a medical term for connect with Latin, but here the project name is Annectent. It is divided by each part, and in case of main server, it is specified as annectent-root, in case of other endpoint servers, it is specified as annectent-node, and in case of client part, it is specified as annectent-door. It can be abbreviated as root, node server.

1. INTRODUCTION

The goal of the project is to have a safe and fast transfer speed by adopting a method of safely and quickly sending data through a virtual pipe, a next-generation Virtual Private Network (VPN) data stream, but by applying a zero-trust model that is difficult to apply. As a part of the platform, I propose a system that has stability and further strengthen security. In addition, based on this, I am researching and developing gates that can be easily built and used to access designated services. As an application, the project's name is Annectent, and through WireGuard, I am conducting research to create a VPN gate service that is faster and lighter than existing VPNs, but with high security and easy to use.

1.1 The need to use a VPN

Many things have developed and changed since Klaus Schwab wrote in Foreign Affairs in 2015 and mentioned the term 4th industrial revolution published by the World Economic Forum (WEF) on 20th January 2016. COVID-19, which occurred at the end of 2019, has changed many things in our daily life and work, the representative of which is the change to non-face-to-face and online services. The number of cloud-based services that can be used conveniently and simply is a big reason, but the most representative reason is the implementation of social distancing in 2020 in many countries and companies. This created the problem of changing more rapidly in rapidly changing IT. These things have the advantage of being able to work anywhere, regardless of residence and time, and tailoring the work environment to me as I use the online service. started to discover. The number of illegal hacks has increased significantly, leaving many businesses already at risk or at risk of being harmed. This is problematic even with a VPN, which is also a problem because VPNs always trust authorised users, leading to vulnerabilities and cyberattacks, as well as data or identity theft.

1.2 Difficulty in building own VPN

Since the VPN passes data through a central server, administrators can see the data they are communicating with if they wish. It is a necessary part of security, but in another sense, it can be a story that data can be viewed by a VPN provider unless it is a VPN that you build yourself. However, handling and deploying security services is a challenging area for small businesses and small businesses as it requires a lot of time, manpower, and money. Especially in the case of privately operated WordPress, even if the admin page is hidden, it is found and subjected to brute force attacks. Even if you use a VPN, security vulnerabilities can occur. In the case of the existing VPN, through tunnelling, the data stream communicates through a virtual pipe so that you can safely access the company network, but there is also the risk that anyone can easily access the company network once authenticated. If you do, there is a risk that server information such as IP may be leaked to the outside. Lastly, while the existing VPN has the advantage of being stable over a long time, it also has the disadvantage of being slow compared to the current Internet environment. To solve this problem, I want to provide a convenient and secure VPN service by grafting the zero-trust model to VPN.

1.3 The need for a fast VPN

There are many VPN services from many cloud service providers and many that make setup easy. For example, AWS VPN, Gateway on Azure, Cloudflare Access, and many other services like Perimeter 81. In the case of OpenVPN, many companies also build and use themselves VPN. However, in most cases, OpenVPN is used, and even if other IPsec/IKEv2 is used, stability has been verified since it was released before 2010, but the speed is slow. There is also a problem in terms of speed now that high-capacity is used when sending and receiving data.

That's why I'm trying to use WireGuard, which is the fastest and has the most bandwidth at this point. WireGuard is still young enough to have been officially released in 2020, but it is already a VPN protocol that is good enough to be certified for stability and security enough to be built into the Linux kernel. For this reason, there are almost no services that are properly utilised for business purposes, and personal VPN service providers have changed only the part where WireGuard stores user information on the server-side and is used. Also, traditional WireGuard is cumbersome to set up and manage with a GUI. This is fine for small-scale personal use, but when it comes to managing and operating a variety of things, it's very difficult. Therefore, Linux is lightly distributed and installed as a server, and it is planned and developed so that it can be used easily on the client. In addition, by using Sing Sign-On (SSO) and Multi-factor Authentication in the enterprise, I propose a safe but convenient method by increasing convenience, security, and integration with other services. Furthermore, I propose a model that allows users other than administrators to have separate privileges by adding a separate privilege function so that access sites or services are restricted according to the period, and when using sites other than work, block/partial permission/access with the original IP is proposed. In addition, the goal of this study is to provide a part of a zero-trust platform so that security can be easily raised in various places, and I want to create a service that works as a demonstration.

2. LITERATURE REVIEW

The literature review included in this section introduces security, speed, and contemporary issues for secure communications and VPNs:

2.1 Data breach problem.

When accessing company services from outside, people connect through the Internet established by the government or telecommunication company. The Internet is like a big plaza where anyone can access and view it, and if you use the Internet in general for things like company secrets, your security can be compromised. That's why I use VPN to hide and encrypt data through tunnelling. In addition, in the case of sites and servers, security is enhanced so that outsiders cannot access them without permission by simply allowing only designated IPs or establishing an internal intranet and using a VPN to access them. As a result of establishing and testing a VPN, if network data sniffing is performed under normal circumstances, data for login such as IP and domain as well as user ID and password may be leaked. However, if you have built OpenVPN and tunnelled through it when you try to sniff it, the output is in encrypted text, so you need a VPN. Of course, if SSL such as HTTPS is applied, it may be fine, but if you use a service port other than the web or if SSL is not applied, it can be dangerous. I think it's right to study in the direction of using a VPN and secure, but not using SSL. From what I've seen, WireGuard is the most up-to-date VPN and I think it's the best because it doesn't use SSL.

2.2 Limitations on legacy VPNs.

As computers became more popular, work environments were done on computers, and various companies started using them as security and VPNs. Originally developed by Microsoft in 1996 and open source, OpenVPN was released in 2001 and is old enough that IKEv2 was created in 2005. Therefore, it is being analysed in many ways, and many attacks and vulnerabilities are also occurring.

In general, VPNs keep you secure in terms of security, such as the value of the data you transfer, but security on the enterprise side can be problematic. Authenticated users with access to the VPN have very high trust and are granted access to resources they do not need. This can lead to things like information and confidential leaks due to data access. VPNs also have performance issues when routing traffic because they rely on a centralised method of connecting clients and resources through a VPN server. This is a very inconvenient feature, especially considering modern internet technology. Even in the 2000s, large-capacity files were not exchanged with data, but these days, large-capacity files and resources are increasing with the development of hardware and software. However, in the case of an existing VPN, it provides about ¼ of the performance in the general 1G network. 5G will become widespread in the future, which will be a very big obstacle. Here is a comparison of the most popular VPN solutions: IPsec, OpenVPN, and WireGuard.

As a result, WireGuard does not perform well in a 10/40GB network environment, but it is faster and lighter than other VPNs. It is considered to be optimal for protecting data and protecting data other than high-capacity data such as video.
On the other hand, the biggest problem with hacking is that traditional VPNs do not build a defence against security threats. For example, when a user infected with malicious code such as ransomware accesses the network, there is a risk that it will spread throughout the network. Due to the nature of VPNs that unconditionally trust their authorised users, traditional VPNs have some security risks, and I recommend using a newer security protocol like WireGuard for this. However, WireGuard is a safe and good choice for situations where the modern Internet environment has a lot of large files. But here too there is a problem. As with traditional VPNs, it gives a lot of trust to authenticated users, allows access, and shows similar aspects in terms of security. This requires additional security solutions. Here are the features and differences of each VPN, and the article that provided this graph explains why I am moving from traditional VPNs to WireGuard. Currently, few business VPNs support WireGuard except for personal VPNs.

2.3 As IP-VPN used for global policy management is used, a centralised server is used, and its limitations.

A centralised VPN approach doesn't just cause speed issues. As it is centralised, attacks such as DDOS can cause connection problems or session hijacking because the central server goes down, and sniffing can occur through packet header tampering attacks through spoofing. (Common Vulnerabilities Exposed in VPN – A Survey [1, 8])
The following vulnerabilities occurred through 2016-2021.

In particular, I can see that vulnerabilities have skyrocketed since COVID-19. For example, in April 2021, Pulse Secure VPN, one of the most popular tools for attacking and exploiting enterprise security, was published in a FireEye report. FireEye is the company that disclosed information about SolarWinds being hacked in 2020. The attackers have announced that access privileges were used to steal account credentials and other sensitive data.
Decentralisation through WireGuard improves security and enables faster data transfer rates,

and it seems like I should enforce a zero-trust model so that even if a user's computer is compromised, the user cannot directly access other users' resources.

The existing VPNs used for business use the IPsec/OpenVPN protocol and have the characteristics of being generous to authenticated clients due to the speed problem and VPN characteristics. WireGuard is light and fast but behaves like any other VPN. In particular, decentralisation seems to be necessary for security or faster latency. Also, I can use VPNs by introducing a zero-trust model for authentication and separation of privileges but aim to be the first step toward building a more secure and zero-trust platform.

3. METHODOLOGY

3.1 Research Setting

The reason I started this research was that I felt that using OpenVPN was very slow when using a VPN. From the confirmed contents, many things have gone online due to COVID-19 along with the security limitations of VPN and the 4th industrial revolution, and the speed of the existing VPN that does not match the current Internet is considered a problem. A good VPN called WireGuard has been newly released, and it is fast, light, and has been recognised for its performance, security, and stability enough to be built into the Linux kernel compared to the official release in 2020. They also say that when it comes to VPNs, using WireGuard can represent high speeds and the next generation. Still, the problem of giving the VPN's authenticators great powers didn't seem to be resolved.

As a result, many forums are talking about building enterprise services based on a zero-trust model. Beginning in mid-2020, the zero-trust concept began to take shape in full. WireGuard has also been officially released and has been very popular, but it still lacks a lot of things for business use. The zero-trust model is also good, but rebuilding all the services operated by an existing company into a zero-trust platform has the disadvantage that it can cost a lot of time and money. In addition, in the case of individuals or small businesses, even if they try to secure quickly and easily, there is a high possibility that they will find it difficult to use and develop personnel.
To recap, here is a list of additional issues.

3.1.1 OpenVPN is heavy and slow.

OpenVPN is open source, and everything is perfect, but the downside is that it is slow. In the past, there were no problems, but with modern data sizes, I often must send a lot of information, so the speed is a concern. Looking at several benchmarks, WireGuard has the upper hand in terms of latency and bandwidth, and the initial connection speed is also noticeably different.

3.1.2 It is difficult to determine who is who by looking at the access log.

When using OpenVPN, was applying NAT for packets from the VPN server to the private network. In this way, due to the nature of NAT, the IP log of the private network resources is recorded with the IP of the VPN server itself regardless of which user used it.

3.1.3 SSO linkage is inconvenient.

As a typical example, many people use a product called the Keycloak to control authentication and permissions and to link your Google account with OAuth 2. Since it is difficult to link SSO with the user account to be used when connecting to OpenVPN, the OpenVPN account must be managed separately, and there is a disadvantage that must manually do it when new subscribers come in or when there is a resignation.

3.1.4 I must borrow the zero-trust model.

Zero-trust means trusting no one. Structurally, it's best to break down the steps to keep users suspicious and have granular access rights. However, it is difficult for small businesses to do so. Many VPN products today don't take much advantage of the zero-trust model. If you successfully connect through the VPN, you will be able to read and use a lot of information in the company without a doubt.

3.1.5 WireGuard does not support Deep Packet Inspection.

No obfuscation when using WireGuard. It only cares about robust encryption and speed. Most of the articles that talk about zero-trust mention building a zero-trust platform instead of a traditional VPN. However, no research has been found to improve security by integrating VPN and the zero-trust model, and open source is also insufficient. In particular, in an era where privacy is rising in many areas these days, the goal is to create a VPN that can be used easily by individuals and can be used immediately by modifying companies that need to protect the confidentiality, server leakage, and malicious hacking. In this regard, I research and propose a model that combines VPN and zero-trust models, which are considered insufficient in research and development, as much as possible. Through this, I conduct a project to research and develop VPN services with ease of use, high speed, and high security. Analyse

WireGuard to use it as it is, but find easy and quick usage, study the zero-trust model, and discard the benefits of WireGuard by creating a secure login and lightweight management server by separating the parts to be grafted on WireGuard, user authority separation, login server separation, etc. I research products that can be easily used through GUI and automated services.

3.2 Research Design

First of all, I tried to summarize the users who will use this product. It is thought that there is an individual in the smallest unit, and it is judged that users who use blogs directly from services such as WordPress to access networks or NAS built by individuals at home will be able to use it easily and increase security. It can also be used for the same reason by those who run WordPress shopping malls, and it would be good for small businesses or companies with less development staff, especially if some of the companies in IT need to change some code and develop quickly. Overall, it is judged that the quality of security can be improved and further damage from hacking can be reduced. I conduct opinions and surveys accordingly. At this time, the questionnaire is easily created and distributed using Microsoft Forms, and the results provided by Microsoft Forms are converted into Excel for analysis and analysis as needed.

3.2.1 Development plan

Ultimately, this study should develop a GUI application that borrows WireGuard and the zero-trust model as much as possible, which follows agile development. Using the basic Software Development Life Cycle (SDLC), the development life cycle is cycled through six stages: gathering/defining requirements, design architecture, development, testing, deployment and maintenance. Organize and analyse in sequence to improve research precision and developability, and continuously revise, supplement, and update applications to make the software more complete. The reasons and things to do for each phase are as follows.

1. Gathering/defining requirements

Gather and organise needs and requirements, outline goals, and list if there are alternative solutions for research and development, and if so, why not use them. In addition, I measure the materials and costs required to develop it and create a solution for development in a better way.

Preliminary analysis

- Easy-to-use VPN construction using WireGuard

- zero-trust concept, login page through external login such as SSO rather than VPN server, and security enhancement and integration improvement through MFA
- permission setting by group Separation through VPN
- decentralisation and speed improvement through the connection between endpoints rather than a central server when using VPN
- Additionally, all records are forgery prevention and security enhancement through blockchain

Alternative solution

- Twingate: Very similar and consistent with my requirements, but open source, so I can't build it myself. You must use a cloud-based server unconditionally.
- Tailscale: It is also available as open-source and meets all requirements. However, it has become more complicated than before with many functions, and it is difficult to manage groups, except for cases such as Synology's personal use. You must use a cloud-based server.

Cost and Convenience

- It aims to install with a lightweight web server and use the built-in WireGuard in the Linux kernel.
- Login uses authenticated, convenient, and secure cloud services such as AWS Cognito and Firebase, and in this case, AWS Amplify is used for convenience.

2. Design architecture

Draw the overall operation method of VPN using WireGuard and the requirements according to the situation as Use Cases to check the necessary services for each situation, and check the overall flow of the system through the flow chart. Finally, it is used as a clean-up and development blueprint for the operation and requirements through ERD.
The development period and overall plan are drawn up through the Gantt chart, and development priorities are established by writing the Project Scope accordingly.

3. Development

It is developed based on requirements and designed results. After developing the server-side, which is the core, first, I develop the client application so that I can connect.

4. Testing

The developed application is tested for errors and bugs. When errors and bugs occur, I make them work normally by analysing and correcting their causes.

5. Deployment

Build the completed application and deploy it to the server. Take action to make it ready for use, along with usage guidelines.

6. Maintenance

When finished, deploy the service. Based on CentOS 9 version, it is distributed so that anyone can use it, and the code is also released so that anyone can build and test it.

3.2.2 Gantt chart

If I develop everything for the project I am currently planning, not only will the time be very short, but it will be very difficult to develop by myself. Therefore, through selection and concentration, I first become a starting point and research and development necessary functions. For this purpose, I draw up a Gantt chart based on the planned contents to establish the flow and plan of the project. The overall flow of the Gantt chart is as follows.

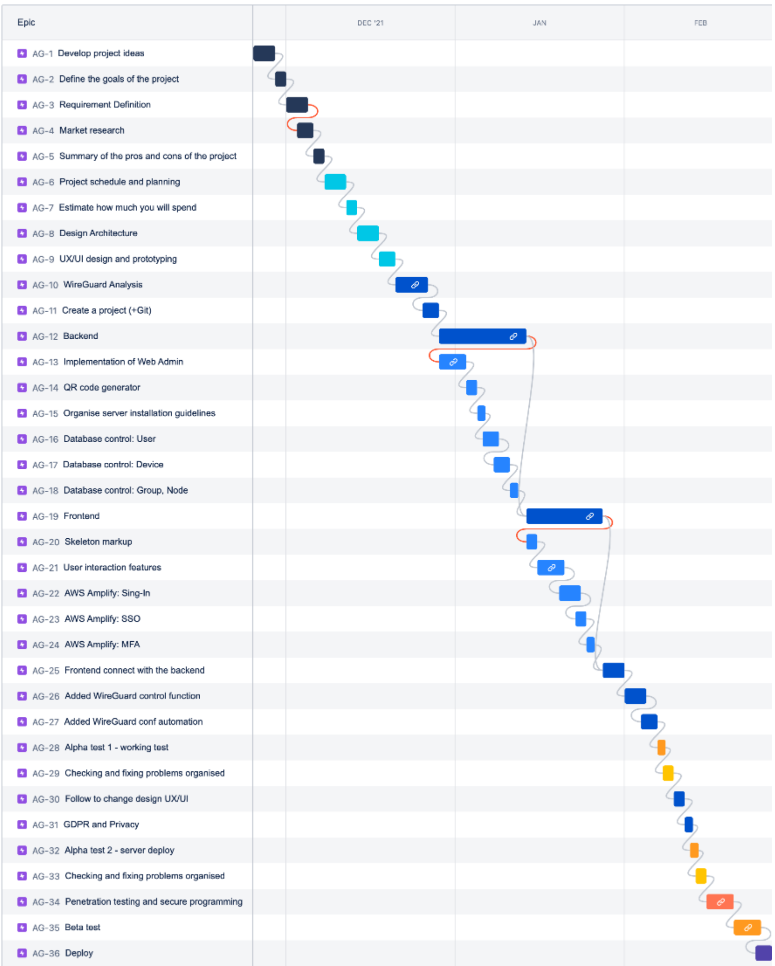


Figure 3-1 Full of Gantt chart

It can check the total parts by classifying them into three categories. The first part is for data research and design before development, the second part is for development. The third part is testing, supplementing, and deploying.

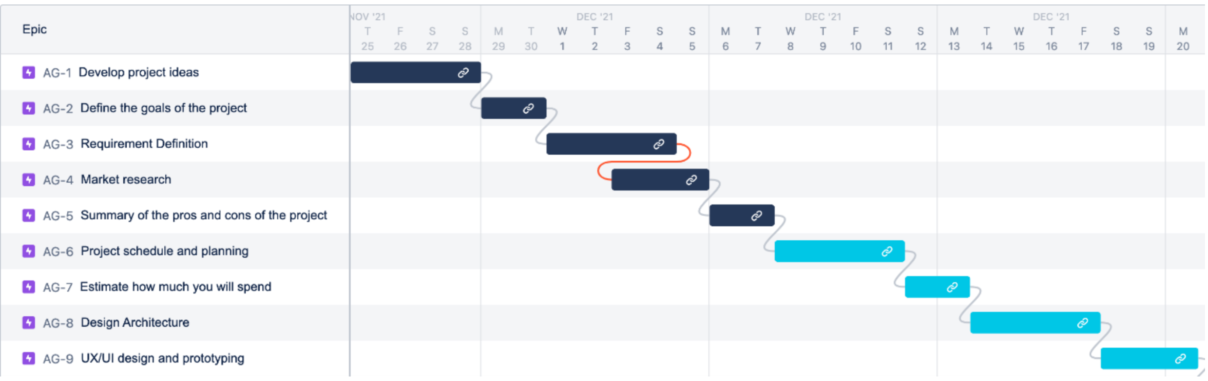


Figure 3-2 First part of Gantt chart

In the first part, research is based on data research and planning. Designation is required in advance before development is included.

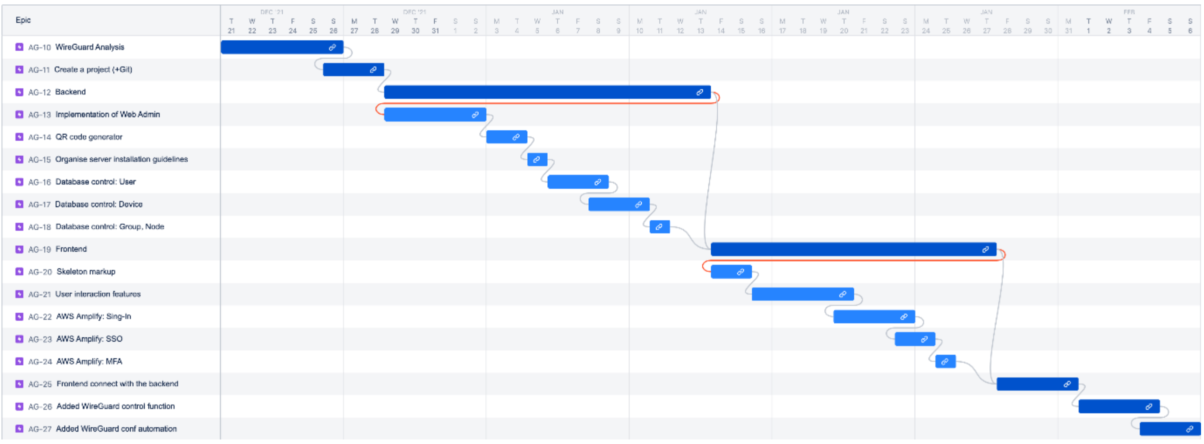
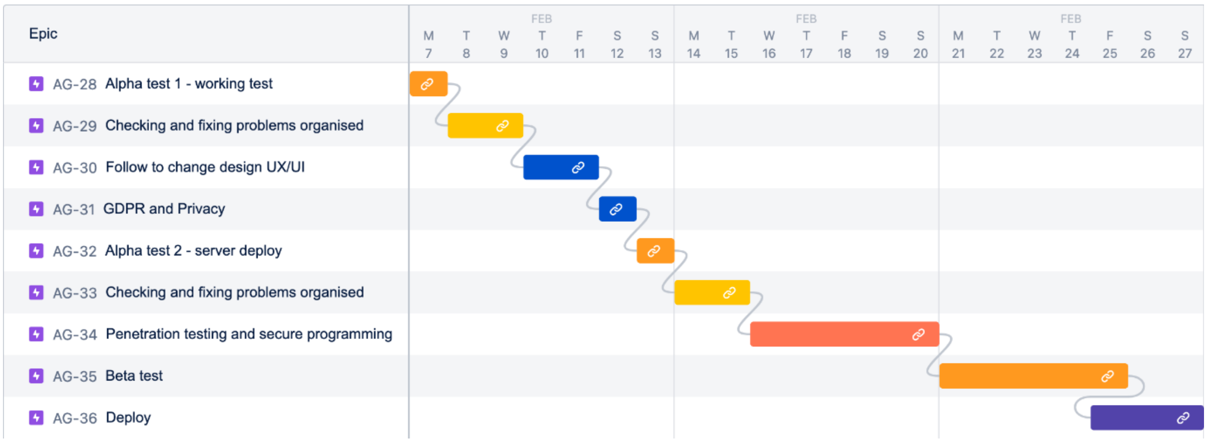


Figure 3-3 Second part of Gantt chart

The second part is the development stage. I develop the backend and the frontend sequentially and work to connect them. After that, additional necessary functions are developed sequentially.



Epic
AG-28 Alpha test 1 - working test
AG-29 Checking and fixing problems organised
AG-30 Follow to change design UX/UI
AG-31 GDPR and Privacy
AG-32 Alpha test 2 - server deploy
AG-33 Checking and fixing problems organised
AG-34 Penetration testing and secure programming
AG-35 Beta test
AG-36 Deploy

Figure 3-4 Third part of Gantt chart

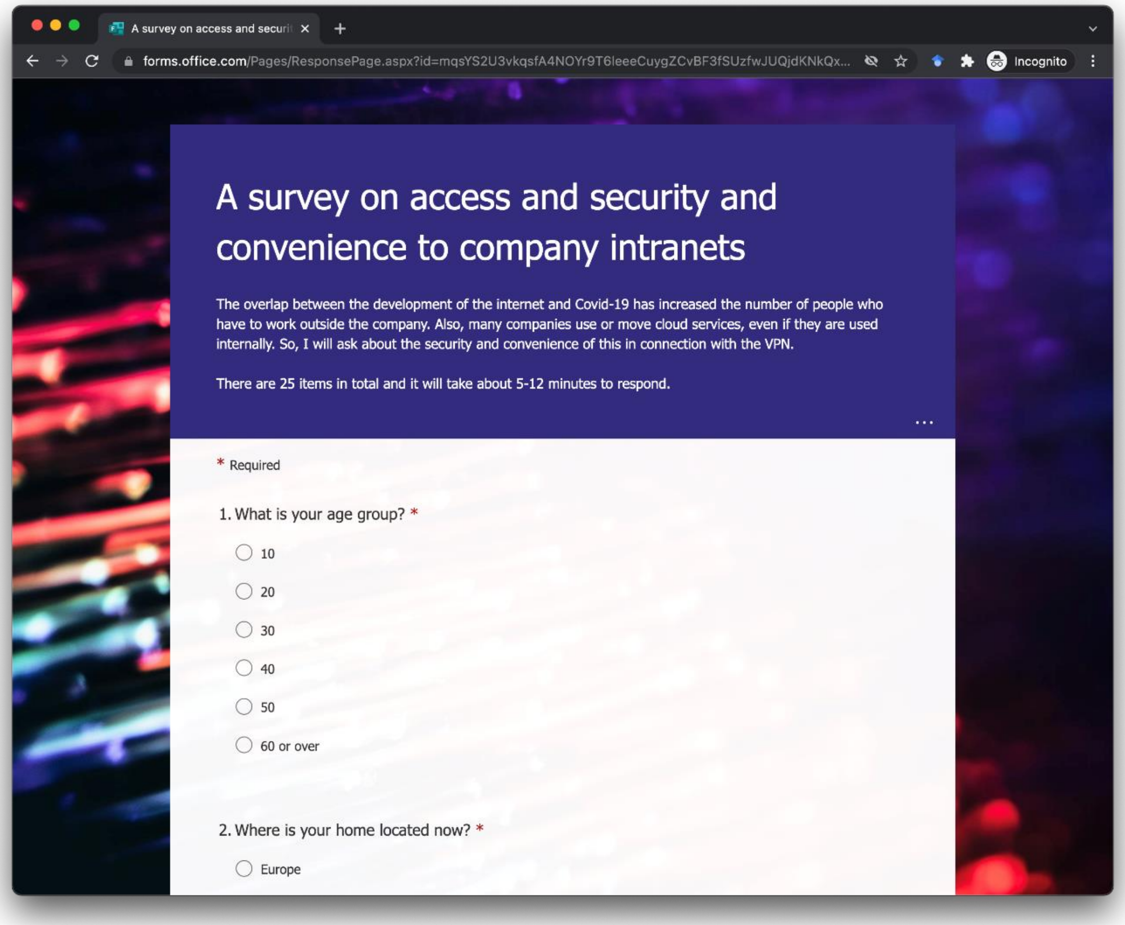
As the final part, I test what was developed and fix any errors. It supplements the missing parts and closely checks for security vulnerabilities. When it's all done, we'll deploy it with beta testing.

4. REQUIREMENT GATHERING

First, I directly build OpenVPN and WireGuard from people in the IT field who have time, including myself, and then ask for various opinions after using them. In addition, through the survey, I investigate whether the company uses VPN, know about Zero-Trust, and if so, how to apply it, as well as inconvenient points and requirements, and the status of adoption of the Zero-Trust model. Survey candidates request survey responses from IT company employees, developers, security experts, or managers.

4.1 Survey

For the first time, I surveyed a Microsoft form with a total of 25 questions, including 3 open-ended questions. A total of 27 people participated, and the average survey time is 08:26. The following are questions and answers to them.


A survey on access and securit
forms.office.com/Pages/ResponsePage.aspx?id=mqsYS2U3vkqsfA4NOYr9T6leeeCuygZCvBF3fSUzfwJUQjdKNkQx...
Incognito

A survey on access and security and convenience to company intranets

The overlap between the development of the internet and Covid-19 has increased the number of people who have to work outside the company. Also, many companies use or move cloud services, even if they are used internally. So, I will ask about the security and convenience of this in connection with the VPN.

There are 25 items in total and it will take about 5-12 minutes to respond.

* Required

1. What is your age group? *

○ 10
○ 20
○ 30
○ 40
○ 50
○ 60 or over

2. Where is your home located now? *

○ Europe

Figure 2-1 Survey form from Microsoft Form

In the case of the survey, it was divided into a total of 5 parts and investigated in detail. It is divided into two main parts: a personal question part about the person taking the survey, and a part about the current usage and perception of VPNs and zero-trust. In detail, it is divided into parts on personal characteristics, job information and size, company IT information, VPN use and perception, and perception of zero-trust.

Individual Characteristics

Questions 1 to 3 are included in this part, and these are asked to determine whether there is a significant difference according to age, current residence, and educational background.

Business information and scale

Questions 5 to 6 are included in this part, and this is the part to ask questions to get information about whether the company's solutions are different depending on the number of people or positions in the company.

Company IT information

Questions 7 to 12 are included in this part, and these are preliminary questions to obtain information about the zero-trust part along with differences in solutions for VPNs that will be asked in the future as the basis of the development team or basic ones.

VPN usage and awareness

Questions 13 and 18 are included in this part, and this part asks more detailed questions about whether a VPN is used, its platform, reason and perception.

An In-Depth Look into Zero-Trust and Security

Questions 20 and 25 are included in this part, and this part asks questions to understand and in-depth research on the current application and understanding of Zero-Trust.

4.2 Interview

Based on the survey conducted above, I answered the interview with 5 questions that were judged to be important. A total of three people responded to the interview, and all of them have been working in the IT field for more than three years. For the interview, it was conducted via an online video call via FaceTime due to COVID-19 and not all of them now live in the same country as me. The content of the interview includes the following questions:

- What challenges have you faced with the rise of telecommuting since COVID-19?
- What do you think is the popularity of WireGuard?
- What do you think of the zero-trust model?
- What do you think about contemporary computer security?
- What direction do you think cyber security should take in the future?

Each question asks the interviewer's thoughts about what was obtained from the survey, based on this, I was asked and got answered questions about what kind of perception I am going to use in this study and what direction I am pursuing. In the interview that was asked based on the contents of this survey, I was able to hear explanations and answers that supported this. First, I can collect information about the use and problems of company services through the situation where I was forced to work online through telecommuting. Second, I get the benefits of using WireGuard over a traditional VPN, and thirdly, get a zero-trust mindset and approach. The fourth and fifth ask questions about contemporary cybersecurity issues and future directions and ask and receive in-depth questions about the research direction and opinions of this project.

4.3 Ethical Statement

This survey and interview were used to obtain accurate values. Those contents are information obtained through direct request and distribution to acquaintances without manipulation. The obtained information obtains only a minimum amount of personal information for privacy and guarantees anonymity.

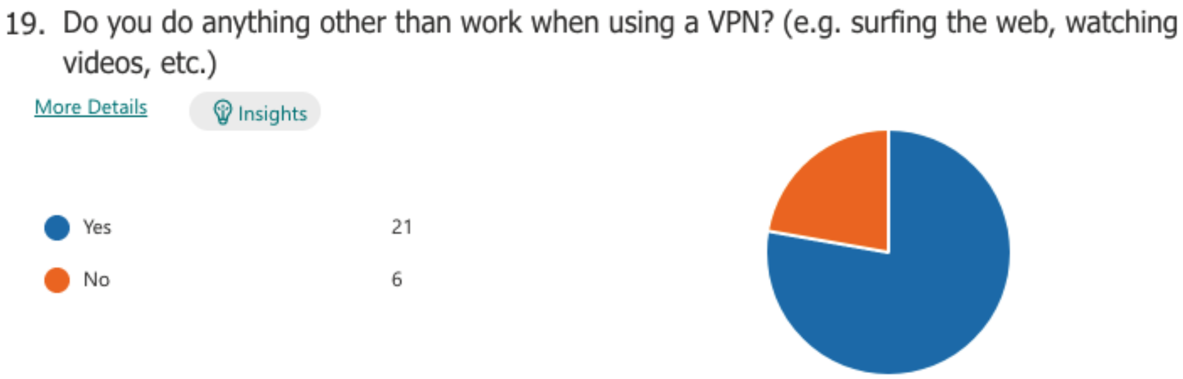
5. ANALYSIS

5.1 Survey Conclusion

The first is a summary of the overall survey. In the survey, most respondents to the survey are in their 20s and 30s. Currently, 14 people in Asia, 9 people in Europe, and 2 people each from other countries residing in Australia and North America completed the survey. The majority of those surveyed are IT and cybersecurity practitioners, all with an IT team or developer within their company. 12 places do not use SSOs, but in the remaining 15 places, one or two or more SSOs are combined and used. I used it.

The main reasons for using a VPN provider are ease of use and safety. If you develop and use a VPN service yourself, the data breach problem is the highest at 8, and in the rest, it is the reason for changing and adding functions to suit the company. As for the protocol used when using a VPN, the OpenVPN protocol was used the most, but many people answered that the VPN is slow or very slow. In addition, more than half said that the zero-trust model was applied to their company, and many people answered that in the future, zero-trust and VPN will be used interchangeably.

There's one important thing I've learned from the survey, it's that I use a work VPN for general web surfing, etc. This is not meant to be a breach of privacy, such as looking into personal information at a company. If you access a site with malicious code or a vulnerable site while surfing the web, the company server connected to the VPN may be at risk or data may be leaked. This is because the nature of VPN allows authorised users to access any type of connection. This is a problem that not only the company but also other employees may fall into danger.



19. Do you do anything other than work when using a VPN? (e.g. surfing the web, watching videos, etc.)

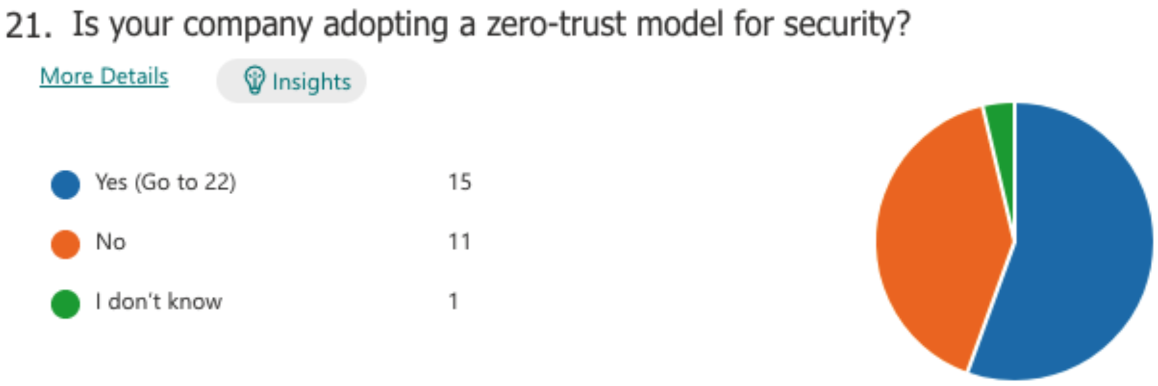
● Yes	21
● No	6

Figure 3-1 Survey result from do others using VPN

Also, as expected, many people said that they use traditional VPNs, but they are slow. It is judged that the problem according to the location on the server will also be large, and there will also be a difference between whether it feels normal or slow due to it.

Moreover, a lot of companies build and use their VPNs. OpenVPN is open-source, so anyone can verify and download it from GitHub and develop new services that use the OpenVPN protocol. The most prominent reason for making a lot of this is the company data leak, which is judged to have been developed on its own due to anxiety about the VPN server. You can also see that self-deployment is active because it has to adapt to the circumstances of the company.

In the case of Zero-Trust, I could see that many companies had adopted it, and it was surprising that it was adopted and used in more places than I expected. This is developed and provided by applying a zero-trust model when providing services on a cloud platform, and it is judged that rapid distribution was possible because it was moved from a legacy server operated within the company to a cloud platform and used flexibly. However, due to the nature of Zero-Trust, it is not only used in one service, but must be well connected and operated in several fields, so this will be the core of this study.



21. Is your company adopting a zero-trust model for security?

● Yes (Go to 22)	15
● No	11
● I don't know	1

Figure 3-2 Survey result from adopting the zero-trust

5.2 *Interview Conclusion*

In the case of the interview, he mentions the security problems caused by the rapid change caused by telecommuting and the difficulties they faced. In the case of WireGuard, they all answered with the same answer that it is fast and highly secure, and there were various viewpoints on the idea of zero-trust. Opinions about convenience were raised through not only the basic security improvement of Zero Trust, but an effective and economical option, and high security but good user experience.

As for the current computer security problem, attack technology is also developing day by day. Regarding the question about the direction of cyber security, opinions about the change of perception were mainly received, and the answer was that the idea of privacy safety and individual cyber security should change. The other answer is the training of white hackers, and the government has given the answer that they want to be able to trust the Internet environment by acting ethically through the training of white hackers.

5.3 Write-up of Finalised Requirements

Through surveys and interviews, it was decided exactly which functions should be researched and developed.

1. VPN

The focus here is to build a fast VPN. There are cases where a VPN is used for legacy system problems or work, and it is often used for security. However, basic VPNs were often criticised for being slow, and since it can be a problem with SSL weakness, it is developed with WireGuard that compensates for these problems.

2. Equipped with Zero-Trust model

It is equipped with a zero-trust model to significantly improve security but also makes it easy to use by increasing accessibility and convenience. Also, a VPN won't work if it's not authorised, like surfing the web, with the ability to only allow access to authorised sites by the group. This ensures that the VPN works and connects only to the server to which it needs to connect with authority.

It also improves login security by isolating the login server. It increases security by one level through SSO and MFA, while also gaining the advantage of user convenience in management.

3. Own operation

VPNs are often used with additional features or modifications to suit each team or company. In addition, I study microservices that can operate on their own or can be used by modifying the code as much as the VPN server runs on its own due to the data leakage problem.

6. DESIGN

6.1 Diagrams

6.1.1 Use case

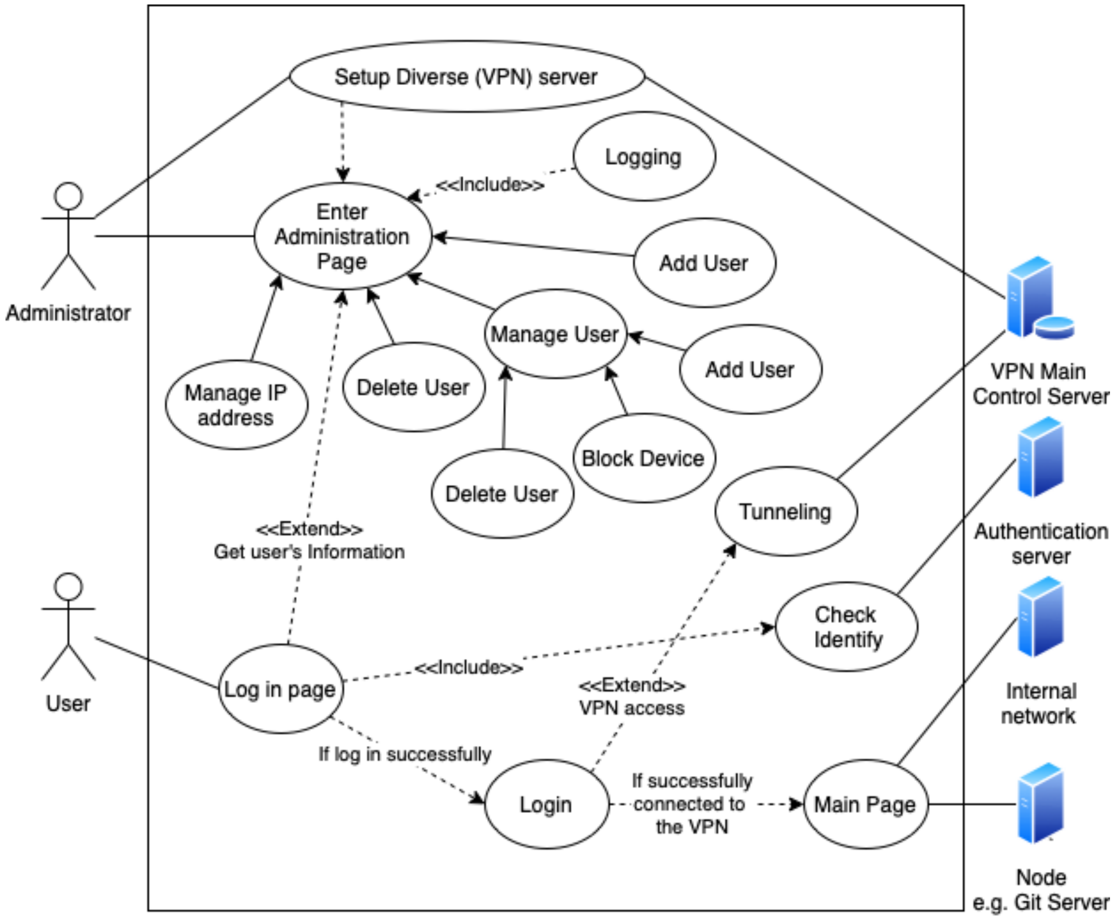


Figure 4-1 Use case

This is a use case made for the overall design before production. First, the Administrator installs the Main Control Server and then manages who can access it by adding or deleting users and gadgets on the management page. Also, depending on the user, there is a limit on the endpoint points that can be accessed (e.g., Git Remote Server). For users, if they are registered on the admin page using the client program, login. If logging is successful, the data is tunnelled using the virtual private network and can connect to the server authorised.

6.1.2 Flowchart

The flowchart was created to find out the parts to be developed, the overall flow, and the corresponding interactions. It is said that it is a general flow, but I am planning to implement the server function, which is the cornerstone, rather than designing the whole, so I will write only that part.

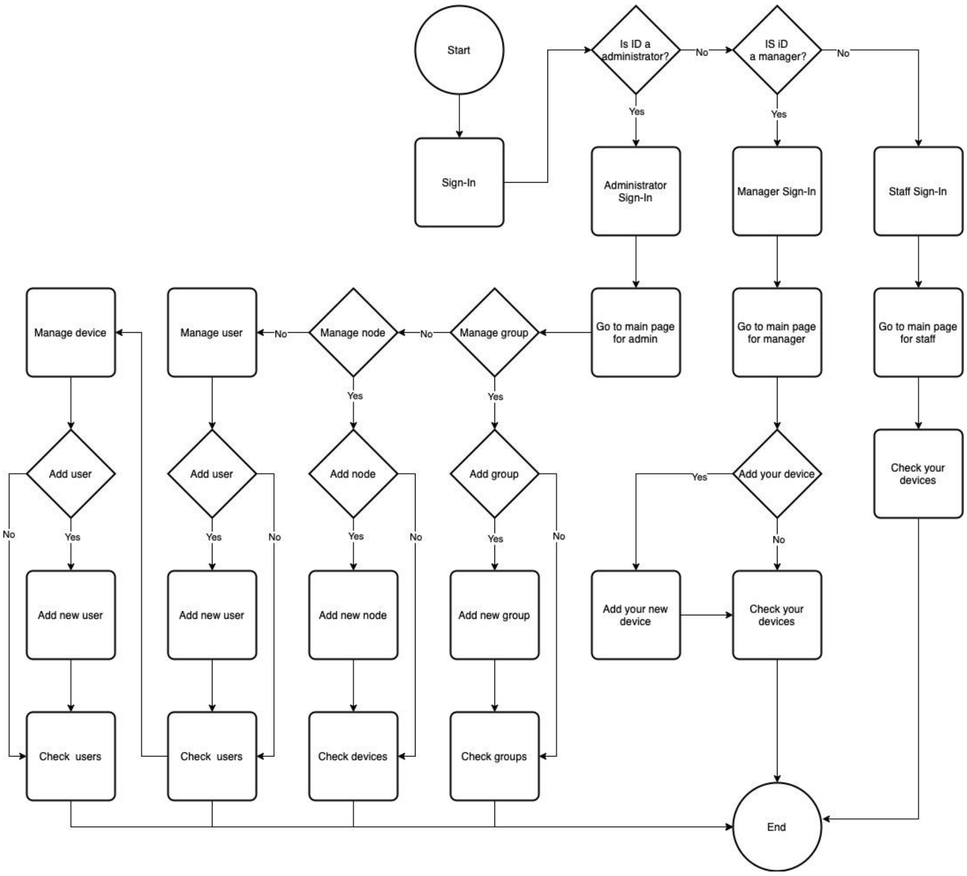


Figure 5-2 Full of flowchart

When the program is executed, the user's type is checked through login. There are three types of users: Admin, Manager, and Staff. Admins have full management rights, Managers can add/modify/delete rights to their own devices, and, finally, Staff can use only the devices designated by the Admin. To easily add to the mobile or desktop, you can access the site and check the information and QR code value.

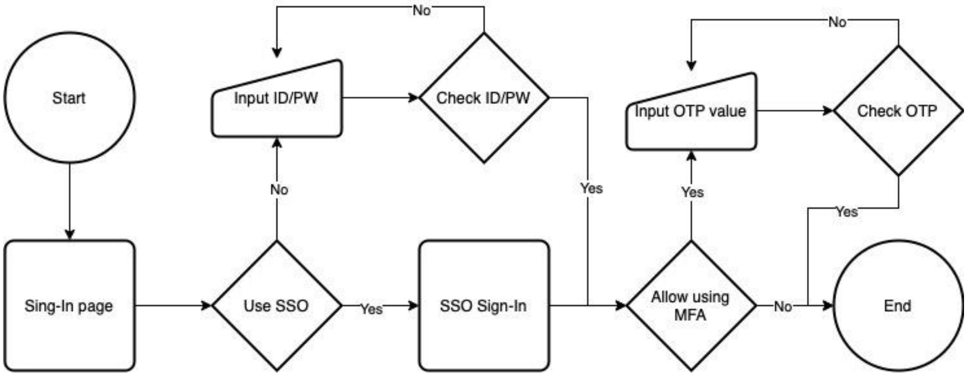


Figure 5-2 Login of flowchart

The login function is simple, but in the case of logging in via email, you can log in using your ID and password. Also, you can log in through services such as Google or Amazon through SSO, and if the MFA function is registered, you can log in by entering TOTP. All of this is done securely through AWS Amplify automation.

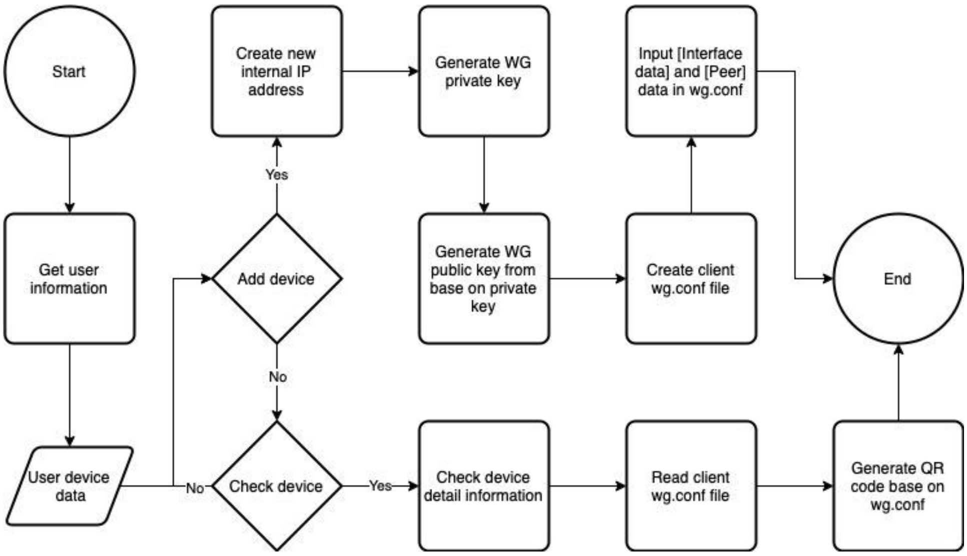

Start
Create new internal IP address
Generate WG private key
Input [Interface data] and [Peer] data in wg.conf
Yes
Get user information
Add device
Generate WG public key from base on private key
Create client wg.conf file
End
No
User device data
No
Check device
Yes
Check device detail information
Read client wg.conf file
Generate QR code base on wg.conf

Figure 5-3 Add device and view of flowchart

The method of using the user device is as in the above flowchart. Although they are created sequentially in the database, since they are divided by user, the user's information is retrieved and checked. At this time, if you press the Create button to register a new device, it is automatically created and set. The generated conf file can be checked in the device details, and it helps you to register easily by converting the generated conf file into a QR code.

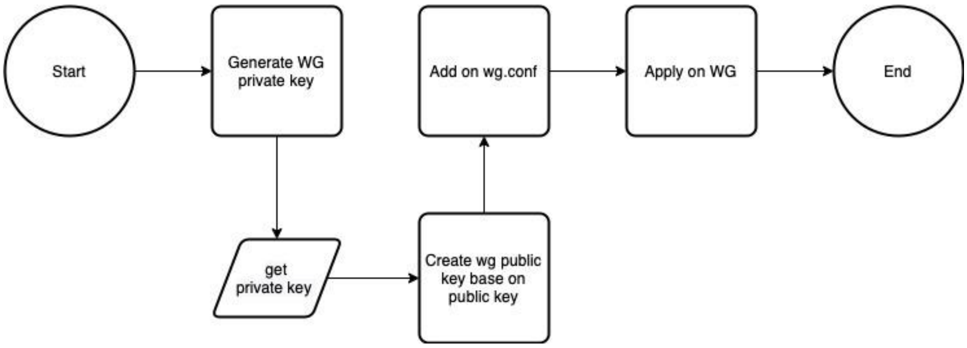


Figure 5-4 Generate WireGuard key of flowchart

To create a new user, you need a private key and a public key generated by WireGuard. For this, a private key is first generated using WireGuard, and a public key can be created based on the private key so that it is automatically registered in the server.

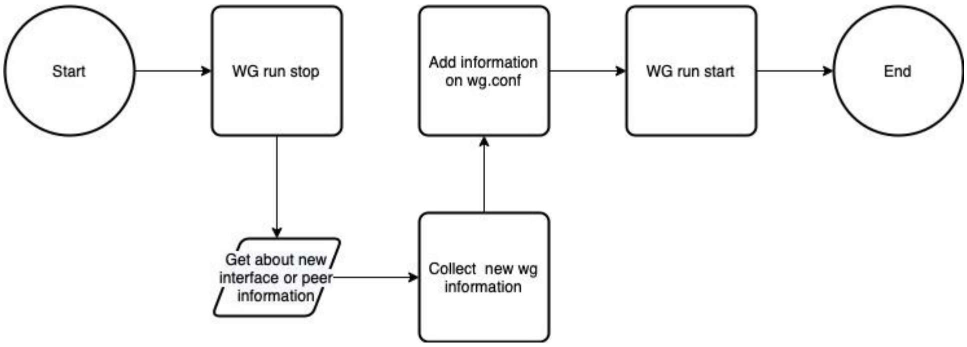

Start
WG run stop
Add information on wg.conf
WG run start
End
Get about new interface or peer information
Collect new wg information

Figure 5-5 Configure setup of flowchart

There is one problem. If you write a conf file in WireGuard, you have to restart it. The problem is that in the case of WireGuard that is already working, changes can be made to the previous conf file on restart. To avoid this, there is a method in which WireGuard is first shut down and then restarted after adding necessary values to the conf file.

6.1.3 Project Scope

Since I don't have much time for the size of the project, I focused on developing the service for the basic server first. In the development field, there is an annectent-root that manages the main server and management, there is an annectent-node, which is a server-side endpoint, and annectent-door, which is a client-side endpoint. However, in this study, only the annectent-root will be produced and tested by selection and concentration. Except for the direct connection between the Node part and the client-side Door part, I first develop a system that enables even login by installing annectent-root together with WireGuard on the server. The login authentication server will be built separately for security. In the current development, I plan to use AWS Amplify for login authentication, and in the future, I plan to use the Keycloak to further simplify the initial setup to greatly reduce the inconvenience during initial installation.

Project name	Explanation	Priority
Annectent Root	It is the most basic server and can be managed by the administrator. It is managed by accessing the web, and it is designed so that users can receive affiliation and authority by the group. It can use the SSO function as well as MFA.	1
Annectent Node	Install WireGuard based server on the server-side endpoint. The node can be added in annectent-root, and it enables the synchronisation of main server information and direct communication with clients. For this purpose, it is adjusted to be usable after login authentication through a client-only application.	2
Annectent Door for Linux	Install WireGuard based server on the server-side endpoint. The node can be added in annectent-root, and it enables the synchronisation of main server information and direct communication with clients. For this purpose, it is adjusted to be usable after login authentication through a client-only application.	2
Annectent Door for macOS and iOS/iPadOS	Develop the client part to use Annectent. It builds on the WireGuard client application. In the case of Apple products, since they are made as universal apps, they are integrated and developed from macOS to iOS/iPadOS at once.	3
Annectent Door for Windows	Develop the client part to use Annectent. It builds on the WireGuard client application. I create the most used Windows application for work.	4
Annectent Door for Android	Develop the client part to use Annectent. It builds on the WireGuard client application. Finally, I make it for Android.	5

Figure 6-6 Project scope of Annectent

The following is a large group of parts necessary for the development of annectent-root, the main of this development, and then the project scope is written.

Part	Explanation	Priority
Automatically install	A function that automatically installs the server with WireGuard as easily as possible.	1
Back-end - Manage group	This part was created to separate privileges for users and nodes through groups.	4
Back-end - Manage user	A high-level set for device management for each user. In addition, there is a function to grant the authority of Annectent through user type.	2
Back-end - Manage user’s device	Devices were available per user. The ability to connect WireGuard.	3
Front-end page	A web page for controlling the Annectent.	4
Front-end connect to back-end	The ability to connect the front-end and back-end to make them work.	5
Sign in page with Amplify	A part that adds Sign in, SSO, and MFA functions through the login function.	6
WireGuard control	The part responsible for creating and executing the Configure file is controlled by the web page by the Annectent Root using WireGuard.	7

Figure 6-2 Project scope of Annectento-Root

What needs to be done after development is to add or modify measures and processing methods that can protect GDPR and privacy, and also create and insert a document specifying this. In addition, WireGuard does not have the data obfuscation technology, so it will add a data obfuscation function, furthermore, I plan to add a data protection function through a blockchain to prevent only each node server from being hacked and deformed.

6.2 Features of the Application

6.2.1 Features

The Annectent I want to develop is a fairly large project that combines a VPN and a zero-trust model. It is equipped with various and many functions so that it can be used. If use all of the Annectent, it will look like this:

- Tunnelling through VPN service using WireGuard
- Microservices using WireGuard's fast and lightweight features
- Very low latency for the server to connect to
- Permission/blocking of server and website to be accessed through permission, improving security through this
- Improved speed and security through decentralisation
- Prevention of forgery and falsification through blockchain
- Improved security through data obfuscation
- Service that is easy to install and use

However, this is a function when I develop everything from the main server to the endpoint. The ultimate goal is to create all of them, but in this study, I cannot develop and test them all due to time constraints. Therefore, I am going to develop only the most basic and central annectent-root first. Detailed features of annectent-root are as follows:

- Install and set up Linux to install annectent-root with an automated installation script
- Building a Secure Sign-in Page Using AWS Amplify
- Add SSO and MFA functionality to AWS Amplify through AWS Cognito integration
- Ability to add and delete users by the administrator
- Configure the device to use WireGuard on a per-user basis
- Separation of privileges through groups (only presets for future development)
- WireGuard Start and Stop
- Easy to use

It would be nice if it could be developed by adding it here, but before that, I plan to faithfully research and develop the function before releasing it. This is the first step in the zero-trust model dealing with WireGuard.

6.2.2 Environment

Candidates for languages to use for development were C/C++, Rust, Go, Java, and Python. First of all, WireGuard, the main axis, was written in C language for peripheral devices and Linux kernel, so C/C++ was a candidate, but I do not need to use it when I develop it, and it also has the disadvantage of difficulty and long development time. In addition, Python and Java are excluded as I are trying to develop light and functional microservices. I wanted to develop it through Keycloak with Java, but it was judged to be heavy, so it was excluded from this study, and I plan to use it when developing for more convenient enterprise use in the future. Rust is very fast and light and is optimal, but the external API, use, and development difficulty are a little more difficult than Go. For this reason, I decided to use the Go language for wigs.

In addition, the Go language is also distributed as wireguard-go officially transferred by WireGuard, and the function library to control WireGuard through wgctrl-go is also officially provided. Above all, it was judged that the compiled language was a little more secure because it is secure software development, and although the Go language may be slightly inferior to Python, it was expected that there would be a huge benefit in security and productivity due to the nature of the language. In addition, the Go language is a language used by Google as a backend, and it was judged to be optimal for processing the backend and WEB/WAS services, and I decided to use the fast and lightweight Svelte as the frontend language. In the case of Svelte, since there is no virtual Document Object Model (DOM) that ReatJS or VueJS has, it is small and light, so the advantages of a lightweight, fast, and small capacity fit well with the project currently developing microservices. In addition, since it is possible to design easily, Svelte is used to quickly create a front-end service, and the connection with the back-end is planned to work through Restful. As a database to store it, I decided to use MySQL, which is a representative database that is commonly used. So this, it can be accessed easily from anywhere, but it increases security by separating the login server.

Type	Platform	Version	Language / Framework	
Server	Linux	CentOS Stream 9 +	Go	Svelte
Client for Linux	Linux		Go	
Client for macOS	macOS	macOS 11 +	Swift	Swift UI
Client for Windows	Windows	Windows 10 +	Go	QT 5
Client for iOS / iPadOS	iOS / iPadOS	iOS 14 +	Swift	Swift UI
Client for Android	Android	Android 10 +	Java	

Figure 7-7 Development environment

In addition, for Linux client applications, I plan to simply use the Go language for rapid development, and for Apple products, I plan to make it a universal app through the Xcode IDE and Swift language and complete it quickly.

6.2.3 Privacy

Privacy issues can arise because communications are managed and used in-between. In this regard, I plan to follow the GDPR, and I will go in the direction of processing data accordingly. I plan to store only minimal information such as user information and gadget information necessary for security, and access related to approved/ unapproved. In addition, when using a VPN, information such as access to and searches of other sites will not be recorded, and measures to protect privacy will be applied as best as possible.

6.3 *Features of the Application*

Before development, I proceed with graphic design about which parts should be made and how they should work. Otherwise, development may take longer, and certain features may be forgotten and not developed. First, simply draw the overall UX/UI through a drawable app. In my case, I draw the overall UX/UI through iPad with Procreate and understand the flow of the application I want to create through it.

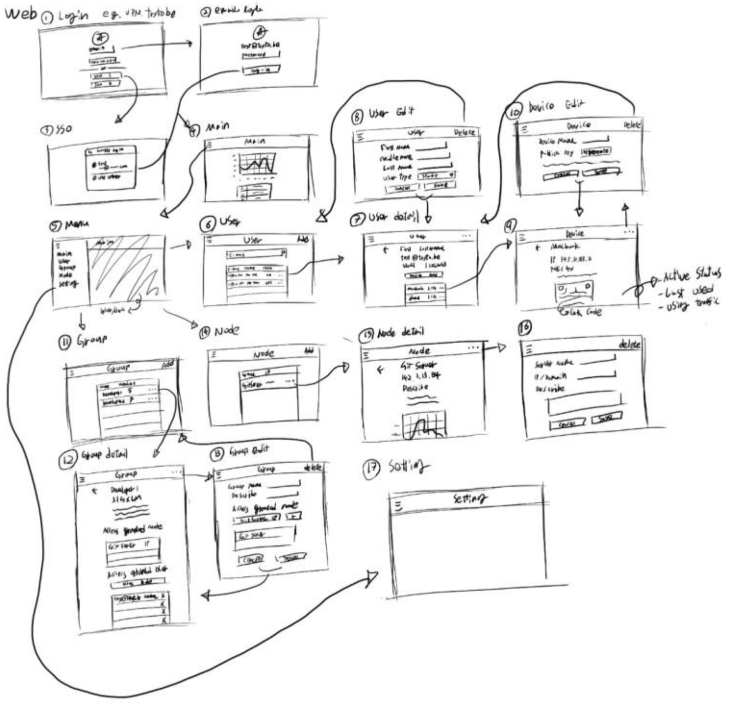


Figure 8-8 UX/UI Draw

I design things through UX/UI tools with sketches made by hand. The tool used is Sketch, and it is set up so that it can easily understand how it works overall using the prototyping tool built into Sketch as well as basic UX/UI design.

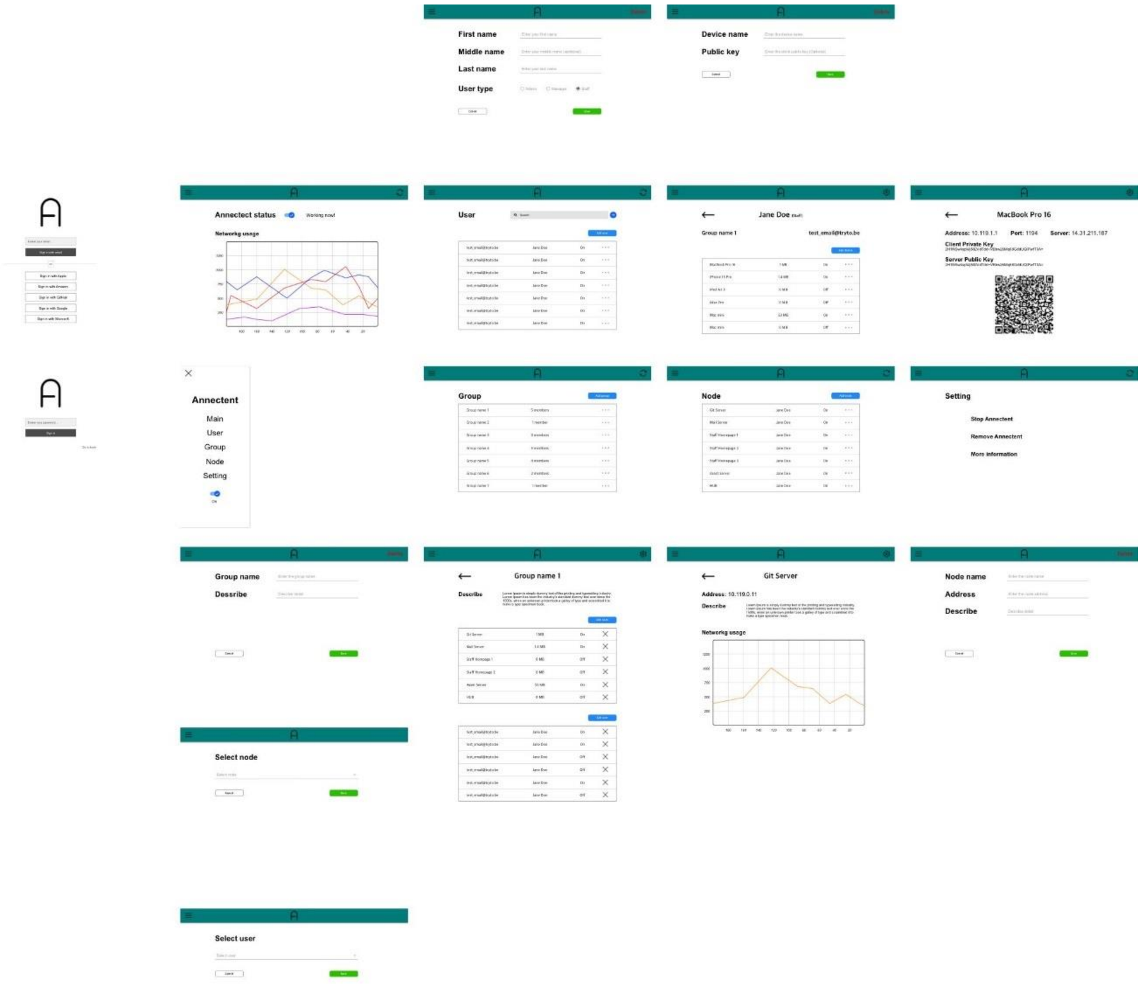


Figure 9-9 Concept of UX/UI full part

The UX/UI design I made this time was not made to apply the same thing, but simply to check the overall UI/UX and the partial list that should be shown in the front-end. Therefore, it is a design that focuses on catching a skeleton rather than a flashy or minimalistic feeling. If development continues in the future, this design will be upgraded, and a more beautiful and intuitive UI and easy-to-use UX will be applied to applications.

This manageable concept application is web-based. It can also add MFA if want along with the login page and SSO feature. After the login page, user and device management (WireGuard Peer), group and node-specific management functions were expressed.

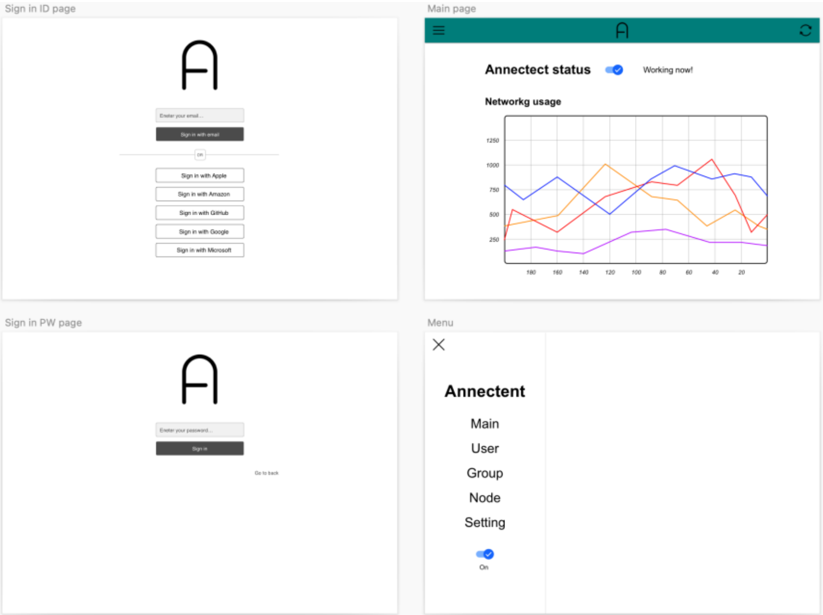


Figure 9-2 Concept of Login, Dashboard and Navigation bar

First of all, it's a login function. In general, you can log in by email, and you can log in through various other SSO logins. If you log in, the MFA function is planned to receive input through Alert.

The main page is a concept. You can monitor your network usage, there is a navigation bar that applies to all pages by default. This navigation bar allows you to navigate to the Main, Users, Groups, Nodes and Settings pages.

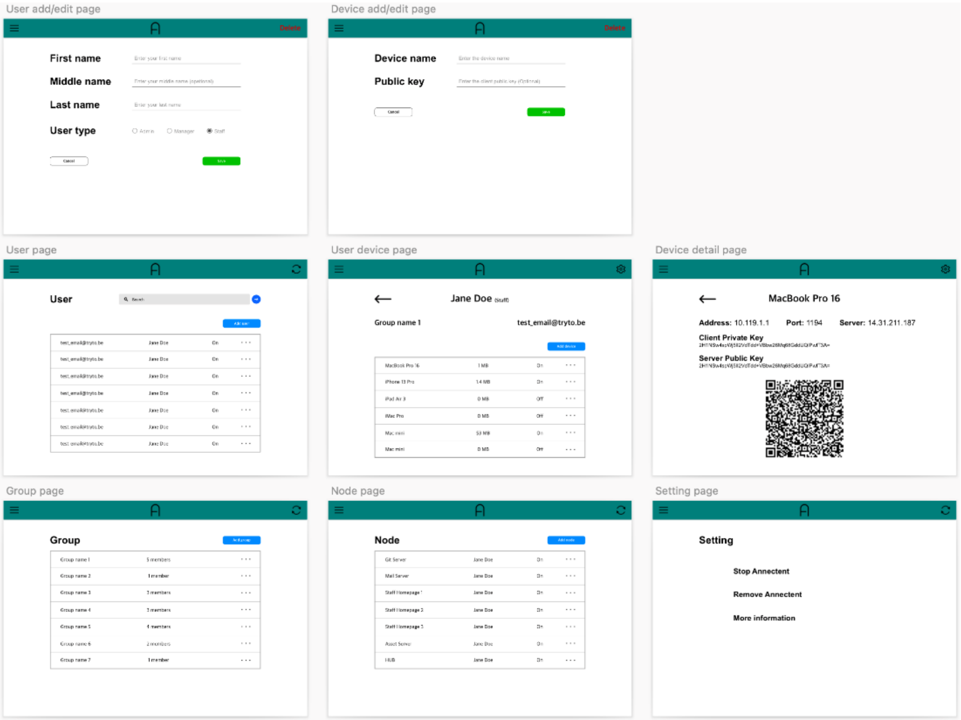
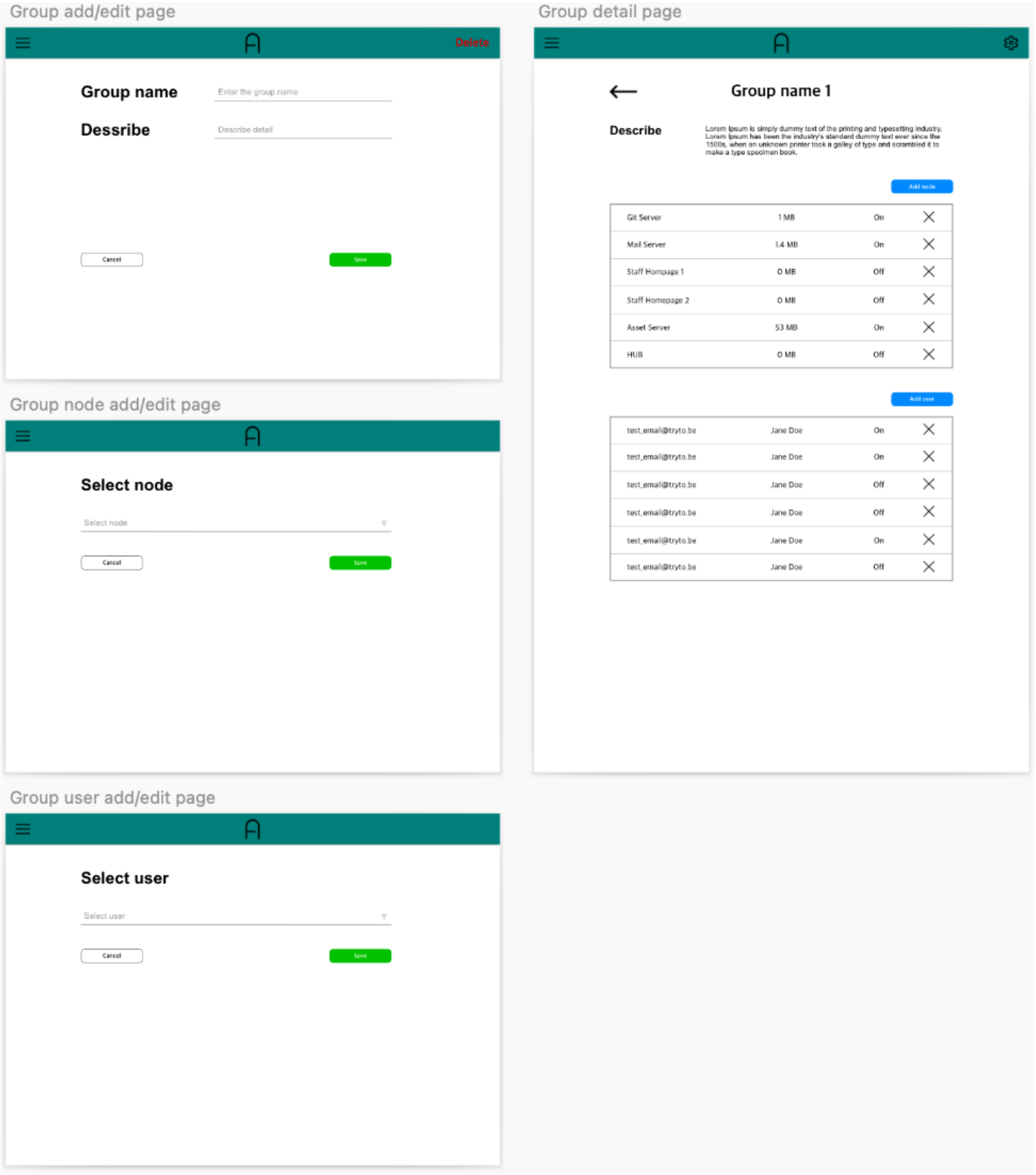


Figure 9-3 Concept of User and Device

For users, it shows a list of users who can log in, and you can manage separate devices for each user. In the case of user devices, it is a function to set the endpoint of the client part of WireGuard, so you can register and delete it through a QR code for easy use.



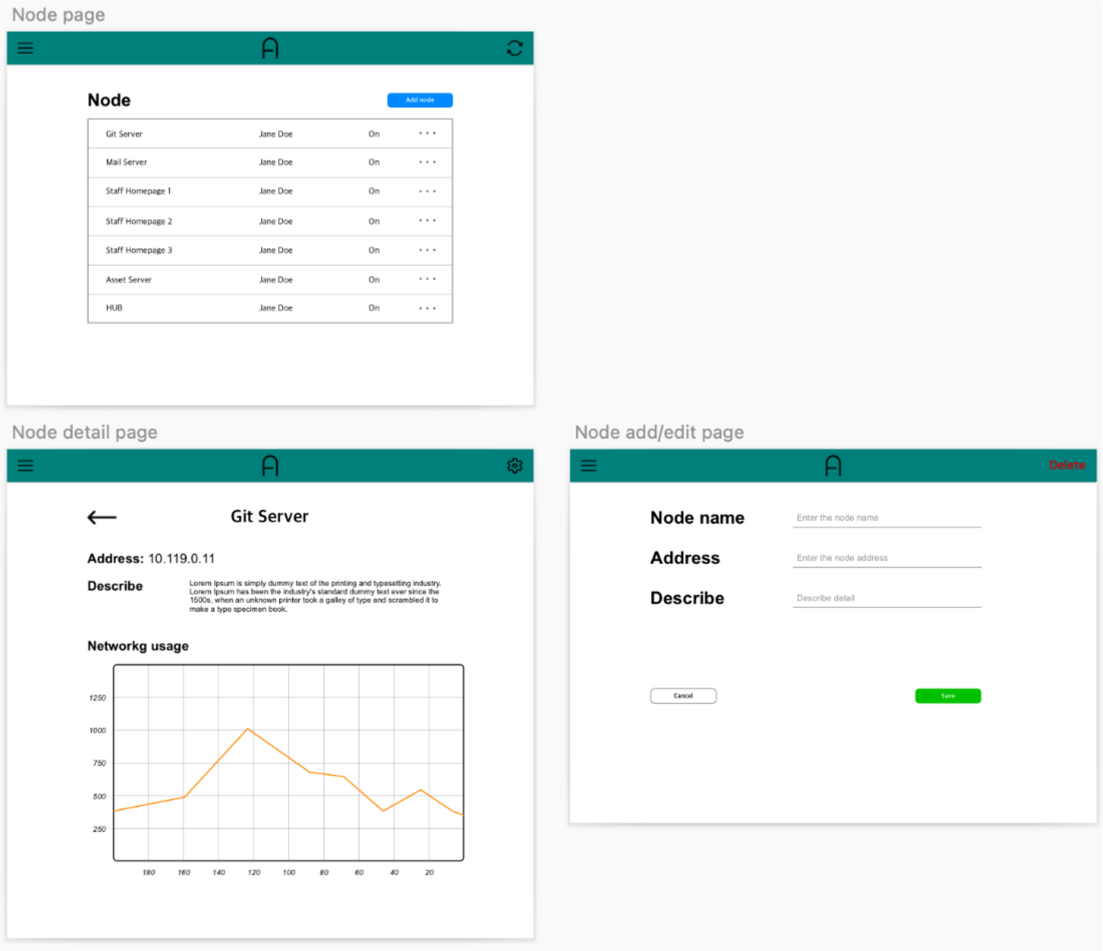
Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

Git Server	1 MB	On	×
Mail Server	1.4 MB	On	×
Staff Hompage 1	0 MB	Off	×
Staff Homepage 2	0 MB	Off	×
Asset Server	53 MB	On	×
HUB	0 MB	Off	×

test_email@tryto.be	Jane Doe	On	×
test_email@tryto.be	Jane Doe	On	×
test_email@tryto.be	Jane Doe	Off	×
test_email@tryto.be	Jane Doe	Off	×
test_email@tryto.be	Jane Doe	On	×
test_email@tryto.be	Jane Doe	Off	×

Figure 9-4 Concept of Group

The group page has a function to add nodes that users can access by adding users and nodes.



Node

Git Server	Jane Doe	On	• • •
Mail Server	Jane Doe	On	• • •
Staff Homepage 1	Jane Doe	On	• • •
Staff Homepage 2	Jane Doe	On	• • •
Staff Homepage 3	Jane Doe	On	• • •
Asset Server	Jane Doe	On	• • •
HUB	Jane Doe	On	• • •

Git Server

Address: 10.119.0.11

Describe Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

Node name Enter the node name

Address Enter the node address

Describe Describe detail

Figure 9-5 Concept of Node

The Nodes page is a satellite server and endpoint server, allowing users to add servers they need to connect to and check network usage.

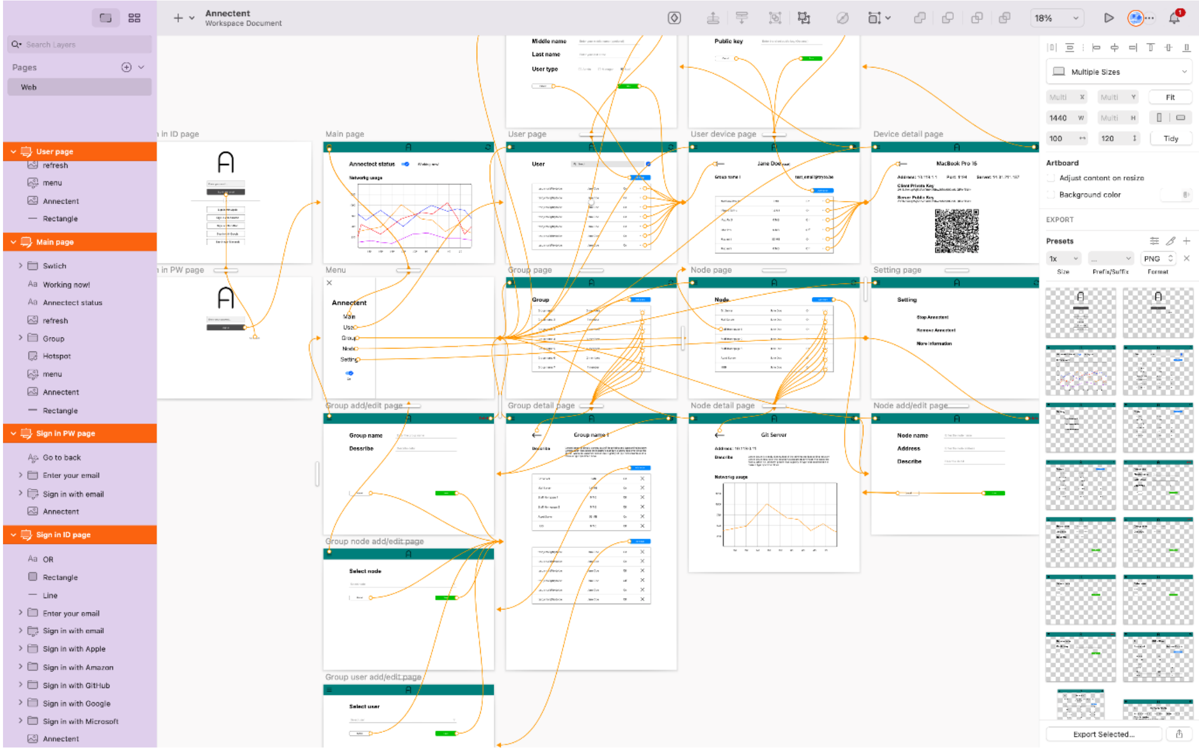


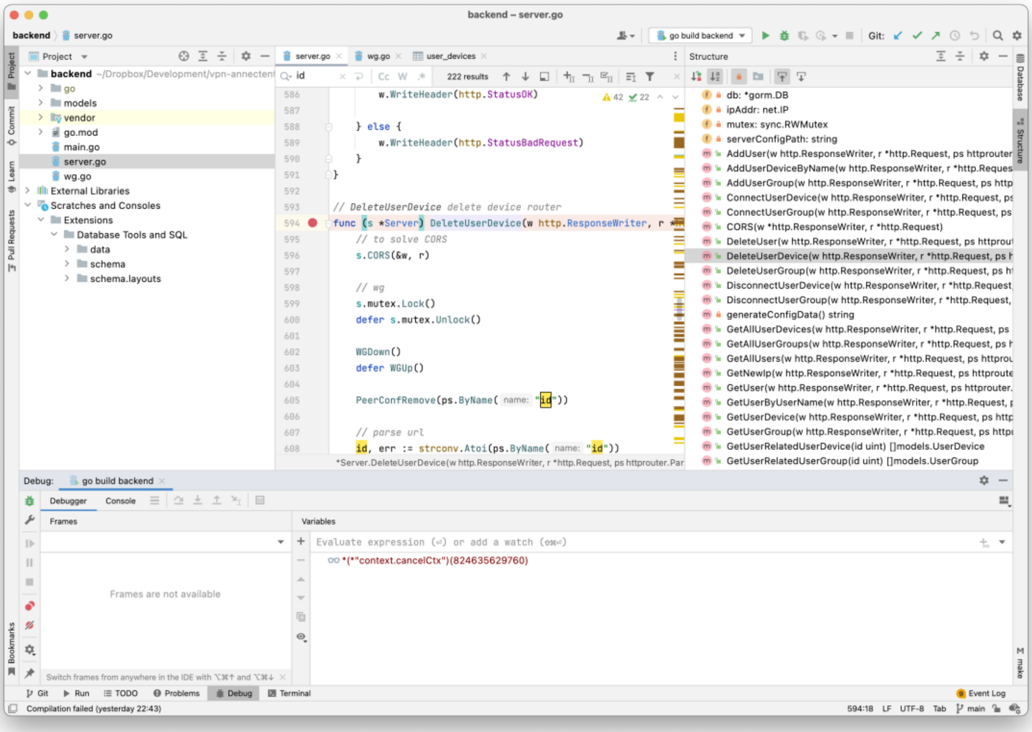
Figure 10-10 Prototyping

It's not finished neatly, but it's enough to give an idea of what parts are needed and how to make them. And based on this, I was able to test and verify the interaction through the prototyping function of Sketch.

7. IMPLEMENTATION

7.1 Building Development

The development was carried out based on what was planned in the Gantt chart, and the necessary functions were arranged and developed sequentially through the Kanban board.



```go
        w.WriteHeader(http.StatusOK)

    } else {
        w.WriteHeader(http.StatusBadRequest)
    }
}

// DeleteUserDevice delete device router
func (s *Server) DeleteUserDevice(w http.ResponseWriter, r *
    // to solve CORS
    s.CORS(&w, r)

    // wg
    s.mutex.Lock()
    defer s.mutex.Unlock()

    WGDown()
    defer WGUp()

    PeerConfRemove(ps.ByName( name: "id"))

    // parse url
    id, err := strconv.Atoi(ps.ByName( name: "id"))
```

*Server.DeleteUserDevice(w http.ResponseWriter, r *http.Request, ps httprouter.Par

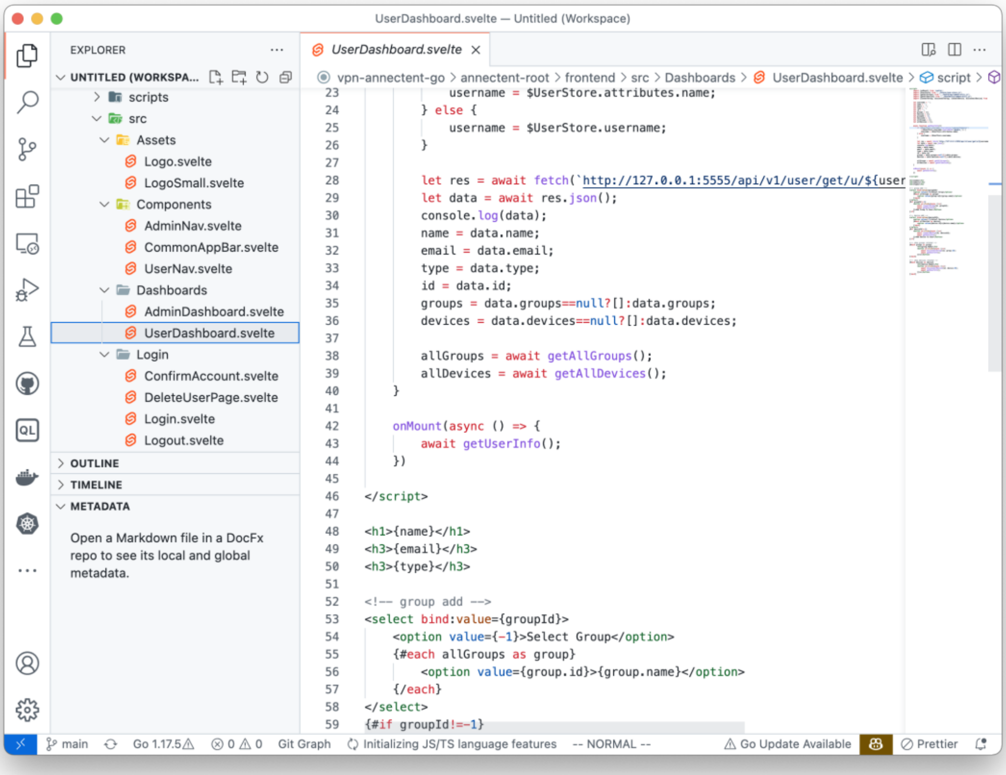
```
db: *gorm.DB
ipAddr: net.IP
mutex: sync.RWMutex
serverConfigPath: string
AddUser(w http.ResponseWriter, r *http.Request, ps httprouter
AddUserDeviceByName(w http.ResponseWriter, r *http.Reques
AddUserGroup(w http.ResponseWriter, r *http.Request, ps http
ConnectUserDevice(w http.ResponseWriter, r *http.Request, ps
ConnectUserGroup(w http.ResponseWriter, r *http.Request, ps
CORS(w *http.ResponseWriter, r *http.Request)
DeleteUser(w http.ResponseWriter, r *http.Request, ps httprout
DeleteUserDevice(w http.ResponseWriter, r *http.Request, ps h
DeleteUserGroup(w http.ResponseWriter, r *http.Request, ps ht
DisconnectUserDevice(w http.ResponseWriter, r *http.Request,
DisconnectUserGroup(w http.ResponseWriter, r *http.Request,
generateConfigData() string
GetAllUserDevices(w http.ResponseWriter, r *http.Request, ps
GetAllUserGroups(w http.ResponseWriter, r *http.Request, ps h
GetAllUsers(w http.ResponseWriter, r *http.Request, ps httprou
GetNewIp(w http.ResponseWriter, r *http.Request, ps httproute
GetUser(w http.ResponseWriter, r *http.Request, ps httprouter.
GetUserByUserName(w http.ResponseWriter, r *http.Request, p
GetUserDevice(w http.ResponseWriter, r *http.Request, ps http
GetUserGroup(w http.ResponseWriter, r *http.Request, ps http
GetUserRelatedUserDevice(id uint) []models.UserDevice
GetUserRelatedUserGroup(id uint) []models.UserGroup
```

```
Evaluate expression (⏎) or add a watch (⇧⌘⏎)
*(*"context.cancelCtx")(824635629760)
```

Compilation failed (yesterday 22:43)

Figure 11-1 Backend develop with GoLand

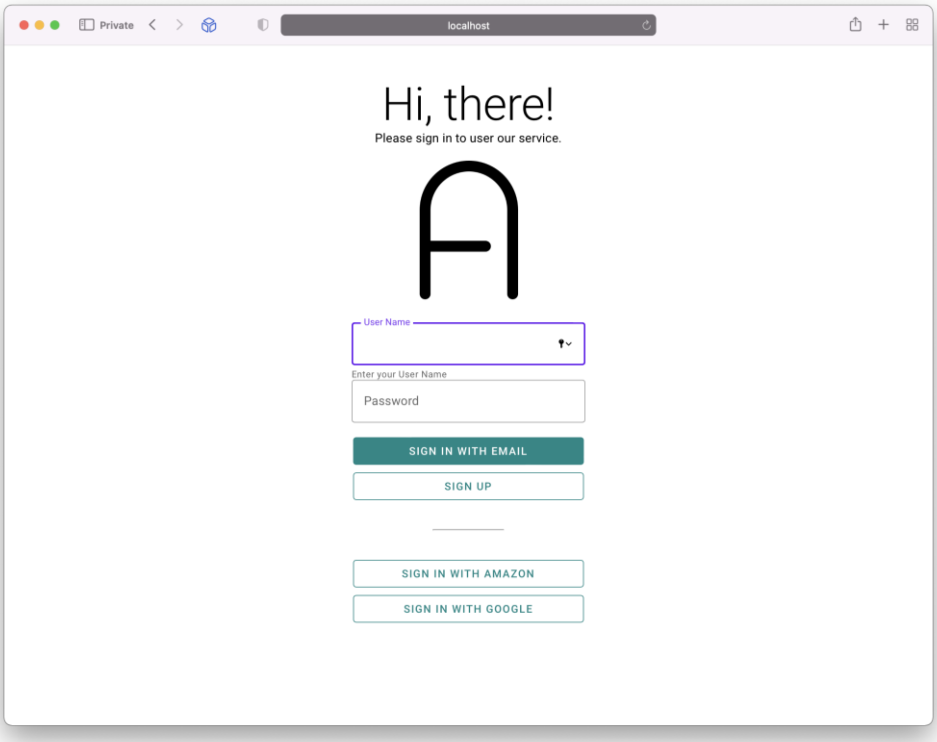
The tools used for development can be grouped into three categories: GoLand IDE for backend development and database in Go language, Visual Studio Code with a few plugins for frontend using Svelte and AWS Amplify connection, and finally Safari browser. It was developed through web testing and debugging through the built-in developer tools.



```
            username = $UserStore.attributes.name;
        } else {
            username = $UserStore.username;
        }

        let res = await fetch(`http://127.0.0.1:5555/api/v1/user/get/u/${user
        let data = await res.json();
        console.log(data);
        name = data.name;
        email = data.email;
        type = data.type;
        id = data.id;
        groups = data.groups==null?[]:data.groups;
        devices = data.devices==null?[]:data.devices;

        allGroups = await getAllGroups();
        allDevices = await getAllDevices();
    }

    onMount(async () => {
        await getUserInfo();
    })

</script>

<h1>{name}</h1>
<h3>{email}</h3>
<h3>{type}</h3>

<!-- group add -->
<select bind:value={groupId}>
    <option value={-1}>Select Group</option>
    {#each allGroups as group}
        <option value={group.id}>{group.name}</option>
    {/each}
</select>
{#if groupId!=-1}
```

Figure 11-2 Frontend develop with Visual Studio Code

Of course, I implemented the user management and login functions through Git and GitHub, which are configuration management systems, NodeJS and NPM for deployment preparation, server testing, and Svelte development through them, and AWS Amplify.



localhost

Hi, there!

Please sign in to user our service.

User Name

Enter your User Name

Password

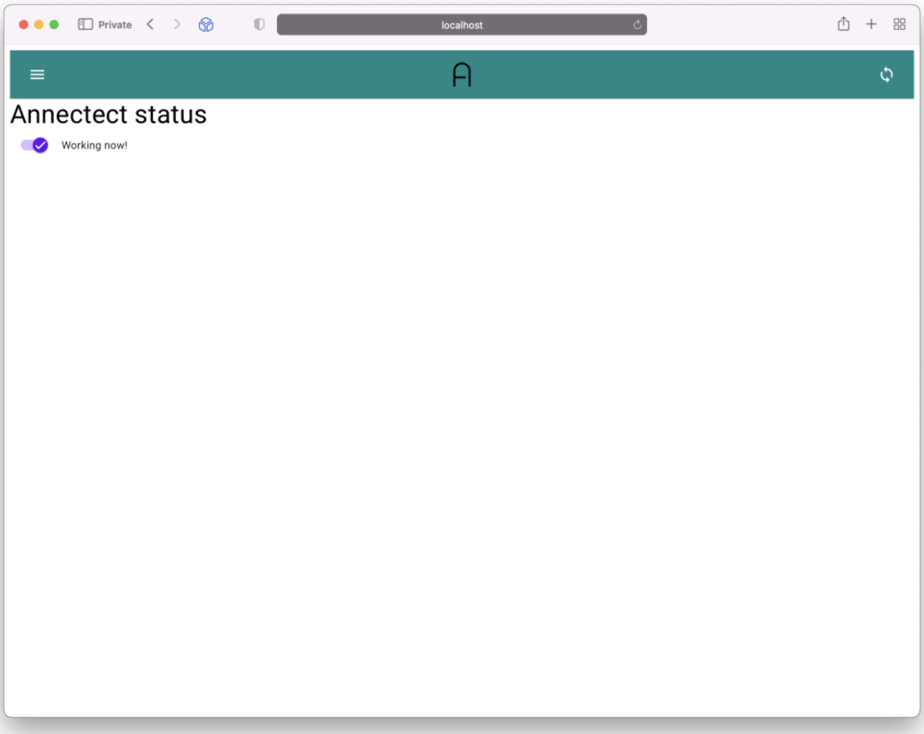
SIGN IN WITH EMAIL

SIGN UP

SIGN IN WITH AMAZON

SIGN IN WITH GOOGLE

Figure 12-3 Working of Sign-In page

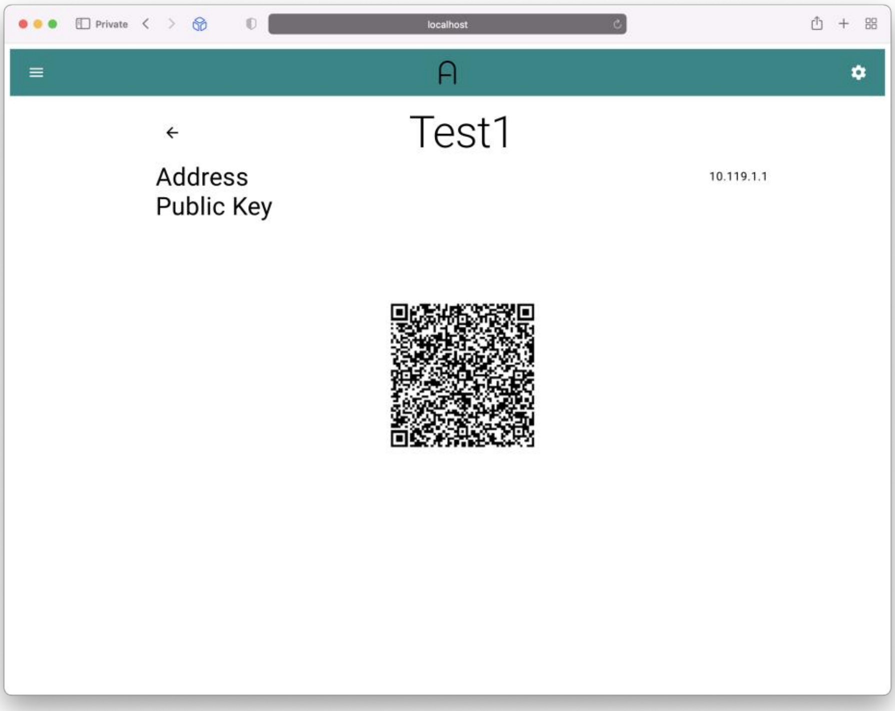


localhost

Annectect status

Working now!

Figure 12-2 Working of Dashboard page

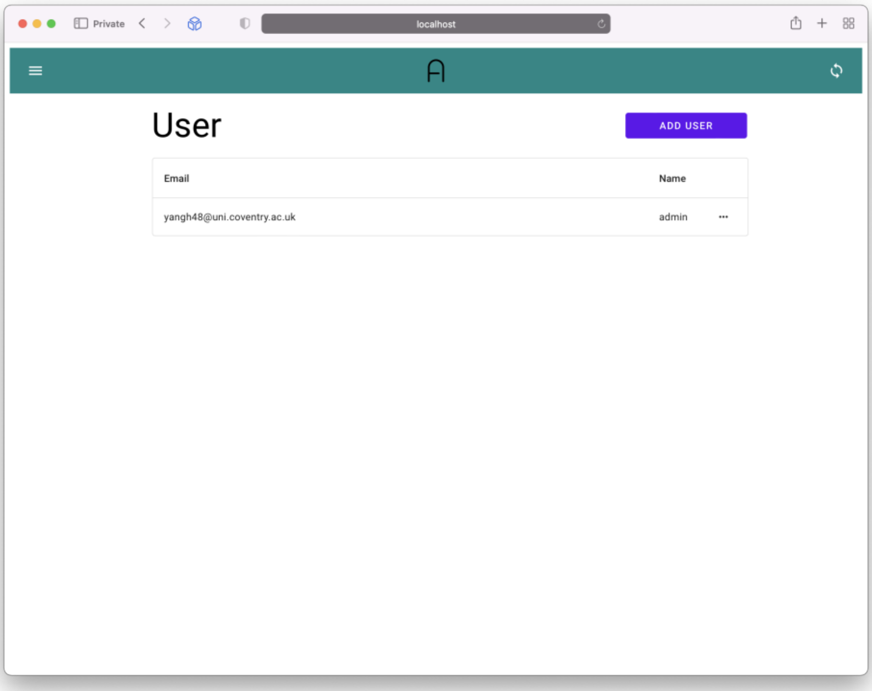


Test1

Address 10.119.1.1

Public Key

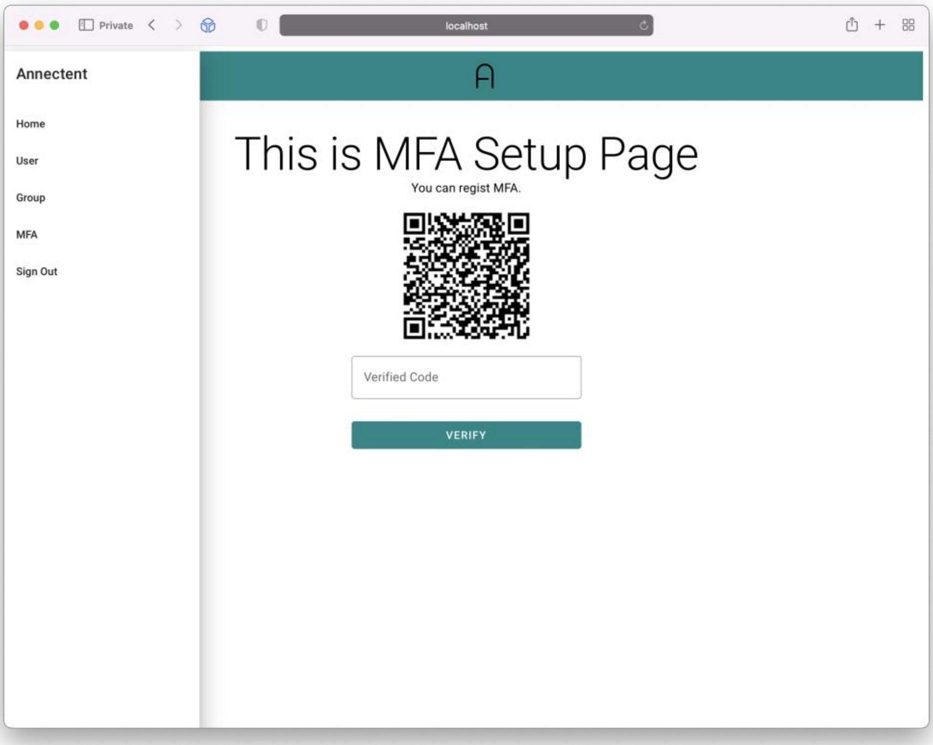
Figure 12-3 Working of Device view page



User

Email	Name
yangh48@uni.coventry.ac.uk	admin

Figure 12-4 Working of User list page



This is MFA Setup Page

You can regist MFA.

Verified Code

Figure 12-5 Working of setup MFA page

7.2 Building Features

The test is divided into two parts. There is an alpha/beta test, which tests the operation in real time at the same time as development, and an overall test of the application afterwards. The first test I do during development is simply to verify that the code is working properly. While developing this part, the method of checking and developing each function was repeated. In the second test stage, a list was prepared, and based on this, tests and errors were corrected. The test list and result values for the final code are as follows.

No.	Title	Check process	Result	Note
1	Backend running normally	go run.		-
2	Automatically create database tables	When running, if there is only a database, the table is automatically created	Pass	-
3	Save input data	The entered values are properly stored in the database table.	Pass	-
4	WireGuard Start	Check whether WireGuard is started as a server function with the wg command in the terminal	Pass	-
5	WireGuard Stop	Check whether WireGuard is stopped as a server function with the wg command in the terminal	Pass	-
6	wireguard directory	Check if wireguard is created	Pass	Error when creating duplicates
7	peer directory	Check if wireguard/peer is created	Pass	Error when creating duplicates
8	Server part key generation	Check if the Private/Public key is created in the wireguard directory as a server function	Pass	-
9	Generate client part key	Check if the Private/Public key is created in wireguard/peer directory as a server function	Pass	-
10	Server part conf	As a server function,	Pass	-

		conf creates and checks data values in the wireguard directory		
11	Client part conf	As a server function, conf creates and checks data values in the wireguard/peer directory	Pass	-
12	Client Peer Matching	Check if the database value and the client conf value match	Pass	

Figure 13-1 Test of backend

No.	Title	Check process	Result	Note
1	Frontend running normally	npm run dev	Pass	-
2	Site access	Connect to port 8080	Pass	-
3	Sign-Up - Email	Try Sign-Up	Pass	-
4	Sign-Up - SSO	Tryp SSO Register MFA after sign-in, log out and sign-in again	Pass	-
5	Sign-In – Email	Try to log in with a registered account	Pass	-
6	Sign-In – SSO		Pass	-
7	Sign-Out	Check logout operation	Pass	-
8	MFA – Email	Register MFA after sign-in, logout and sign-in again	Pass	You need to refresh when you go to the dashboard.
9	MFA – SSO		Fail	-
10	Navigator bar	Make sure all pages are working properly	Pass	-
11	Dashboard WireGuard Button Operation	Check the on/off button operation of the wire guard	Pass	-
12	Group add-on	Check group page access and add/modify/delete the new group	Pass	-
13	Group edit function		Pass	-
14	Group delete function		Pass	-
15	User add-on	Check user page access and	Pass	-
16	User edit function		Pass	-

17	User delete function	add/modify/delete the new group	Pass	-
18	Device add-on	Check device page access and add/modify/delete the new group	Pass	-
19	Device edit function		Pass	-
20	Device deletion function		Pass	-
21	Device check by the user	Make sure the device is different from each other	Pass	-

Figure 13-2 Test of frontend

No.	Title	Check process	Result	Note
1	Server running	Check if it runs on CentOS Stream 9	Pass	Amplify not working for AArch64 architects
2	Client - Mobile Connection	Registration and connection with the server based on the contents of the Conf file output as a QR code from the client device part	Pass	DNS must be entered
3	Client - Linux Connection		Pass	DNS must be removed for normal operation
4	Client - macOS Connection		Yet	-

Figure 13-3 Test of Deploy

7.2.1 Deployment Plan

The most important thing about this project is deployment. However, it has not been fully developed due to time, and additional functions and various bugs need to be resolved. You can get the current code from GitHub, build and run it, but since there is no integrated build and run function, if you want to use it, you have to build the frontend and the backend yourself and run them separately. Currently, it is necessary to temporarily run the frontend through "npm run dev" and the backend through "go run .". It is expected that these issues will be resolved by March 5, when the first distribution is expected.

8. CONCLUSION

8.1 Review each chapter

Chapter 1. INTRODUCTION

In this chapter, I've described the whole project, what it's trying to do, and why I'm making it. The current situation and the need for a secure and fast VPN, along with the overall reasons for wanting to do this, have allowed us to go from research and design to development for it.

Chapter 2. LITERATURE REVIEW

In this chapter, I've described the whole project, what it's trying to do, and why I'm making it. The current situation and the need for a secure and fast VPN, along with the overall reasons for wanting to do this, have allowed us to go from research and design to development for it.

Chapter 3. METHODOLOGY

In this chapter, I've described the whole project, what it's trying to do, and why I'm making it. The current situation and the need for a secure and fast VPN, along with the overall reasons for wanting to do this, have allowed us to go from research and design to development for it.

Chapter 4. REQUIREMENT GATHERING

In the fourth chapter, before the precise design, I checked how it is being used in current companies and work, and whether what I am trying to develop needs. And it is possible to secure the justification for development and further collect additional requirements.

Chapter 5. ANALYSIS

This is the stage to describe which functions should be developed by analysing the data collected through surveys and interviews. It allowed me to organize the necessary functions and lists for design and development.

Chapter 6. DESIGN

In the design, it was possible to understand the overall situation through use cases and determine what operation should be performed for each part with a flowchart. Furthermore, it played an important role in figuring out how to operate in advance with UX/UI design and prototyping and applying it.

Chapter 7. IMPLEMENTATION

I developed a backend in Go language, a frontend in JavaScript and Svelte, and created an application to store in MySQL. A function to handle WireGuard was added to make the VPN function easy to use, and it was made safe from attacks by implementing a secure but easy login function through AWS Amplify and separating it from the main server. We've even prepared it for deployment via GitHub.

8.2 End of the research and project

In this study, I designed a new VPN service with a zero-trust model concept using WireGuard. In terms of the overall design, it was judged that it was very effective and highly secure, and it was convenient to use. In addition, it was determined that the server's network usage would be reduced and security auditing would be an easy service.

Due to the lack of time in the development part through this, only the main server, annectent-root, which is the core, was produced. Accordingly, the achievements that can be used and completed this time are as follows.

Easy to install and use WireGuard

Separating Login Servers for Zero Trust Model Enforcement

Enforce SSO and MFA with AWS Cognito Connections via AWS Amplify

A lightweight service is written in the Go language

Light and clean UI made with JavaScript and Svelte It was judged to be very effective, high security, and convenient to use. In addition, it was determined that the server's network usage would be reduced and security auditing would be an easy service.

Due to the lack of time in the development part through this, only the main server, annectent-root, which is the core, was produced. Accordingly, the achievements that can be used and completed this time are as follows.

- Easy to install and use WireGuard
- Separating Login Servers for Zero Trust Model Enforcement
- Enforce SSO and MFA with AWS Cognito Connections via AWS Amplify
- A lightweight service is written in the Go language
- Lightweight and clean UI made with JavaScript and Svelte

And this project is big challenge for me. I've built and used WireGuard before, but it's all done in a CLI environment. It was a challenge for me to use this conveniently in a GUI environment. Developing a backend with the Go language, especially for the first time, was a huge challenge for me. In addition, to provide a more microservice, the front end was also challenged with Svelte. It was very difficult and difficult for me to study new languages and data materials. Especially since it was a full stack development, there were a lot of things that were confusing and difficult. In addition, when fetching the front-end value, various interworking problems occurred, such as a problem that the device could not be deleted because the value was not properly received from the back-end. There was no problem that I was stuck on for several days because one thing didn't work, but there were so many errors and bugs that I couldn't even count. I think it would have been very difficult to complete development without GoLand's automatic recommendation function and debugging function. I can't handle it perfectly yet, but thanks to this, I've finally gotten used to the Go language and Svelte a lot. It is judged that it will be of great help in developing the Annectent project continuously in the future, and it is judged that it will be of great help when doing other projects.

In this study, it went well according to the Gantt chart, but there was a delay due to a laptop failure during the development process. Fortunately, there were no problems with the development, but the time is right and the function works, but I think it is insufficient to call it a proper function. In addition, it was not completely completed, so distribution was not made, and I plan to distribute it through additional development and modification in the future. Although it has been distributed to the current server, it is my subjective judgment that it is only for testing and it is difficult to see that it has been officially distributed.
Also, I think the problem was that the ER graph was not charted. There is no problem in operation, but because of this, data storage and other parts are a bit unnatural. The part where values are entered into the database is not yet perfect, and in particular, functions to store and manage private or public keys will need to be further developed. Also, it seems that the frontend and backend need to be modified to work more organically.

REFERENCES

Abdulazeez, A. 'Comparison of VPN Protocols at Network Layer Focusing on Wire Guard Protocol'. JIM – Vol. 14, No. 18, 157-177

Bansode, R. (2021) ‘Common Vulnerabilities Exposed in VPN – A Survey’. CONSILIO 2020 Journal of Physics: Conference Series 1714(2021) 012045, 1-8

Donenfeld J. (2020) ‘WireGuard: Next Generation Kernel Network Tunnel’. www.wireguard.com, 1-20

Iqbal, M. 'Analysis of Security Virtual Private Network (VPN) Using OpenVPN’. International Journal of Cyber-Security and Digital Forensics 8(1), 58-65

Lipp B. (2018) ‘A Mechanised Cryptographic Proof of the WireGuard Virtual Private Network Protocol’. 2018 IEEE International Congress on Internet of Things, 88-95

NIST (2020) Special Publication 800-207 Zero Trust Architecture [online] <https://csrc.nist.gov/publications/detail/sp/800-207/final> [August 2020]

Papakonstantinou (2021) 'A Zero Trust Hybrid Security and Safety Risk Analysis Method'. Journal of Computing and Information Science in Engineering 2021 ASME, 1-26

Pudelko, M. 'Performance Analysis of VPN Gateways’. 2020 IFIP Networking Conference, 325-333

Samaniego M. (2016) ‘Zero-Trust Hierarchical Management in IoT’. 2016 IEEE, 1-6

APPENDICES

Appendix A. Servey

No.	Survey question	Answer
1	What is your age group?	• 10: 0 • 20: 12 • 30: 12 • 40: 3 • 50: 0 • 60 or over: 0
2	Where is your home located now?	• Europe: 9 • Africa: 0 • Asia: 14 • Australia: 2 • North America: 2 • South America: 0 • Pacific Ocean: 0
3	What is the level of education or highest degree to you have completed?	• High School: 0 • Diploma's Degree: 2 • Bachelor's Degree: 20 • Master's Degree: 5 • Ph.D. or higher: 0
4	What is your job field?	• IT & Computer: 14 • Cybersecurity: 10 • Engineering: 1 • Business Management: 1 • Others: 1
5	What is your job title?	• CEO: 2 • Vice President: 0 • Director: 3 • Manager: 7 • Individual Contributor: 12 • Entry-Level: 3 • Freelancer: 0
6	How many workers in your company?	• 1-9: 0 • 10-49: 4 • 50-99: 10 • 100-499: 12 • 500+: 1
7	What is your working desktop or laptop OS for work?	• Windows: 17 • macOS: 9 • Linux or BSD: 1
8	What is your mobile phone or tablet OS for work?	• Android: 7 • IOS: 20
9	Does your team have an IT team or developers?	• Yes: 27 • No: 0
10	10. Do your company use SSO?	• Google: 9

		• Microsoft: 1 • Amazon (AWS): 3 • GitHub: 1 • Okta: 5 • OneLogin: 0 • Auth0: 0 • Not use: 12
11	Does your company use an email service?	• Gmail: 7 • Outlook: 1 • Exchange: 14 • Amazon WorkMail: 1 • Built by the company itself: 4 • Only use instance type (e.g. Slack, Teams): 0 • Not using email: 0
12	Does your team use a security solution?	• Yes: 27 • No: 0
13	Does your company use a VPN? What VPN are you using?	• AWS VPN (Go to 14): 4 • Azure VPN (Go to 14): 1 • GCP VPN (Go to 14): 1 • OpenVPN (Go to 14): 3 • Perimeter 81 (Go to 14): 0 • Twingate (Go to 14): 0 • NordLayer (Go to 14): 0 • Built by the company itself (Go to 15): 10 • Not using VPN (Go to 16): 8
14	If you're not using a VPN you built yourself, why not?	• No developer: 0 • Difficult to build: 2 • Use to easy: 8 • It's secure and safe: 6 • Cheaper than build myself: 1
15	If your company has developed its own VPN, why?	• Beware of data breach: 8 • Expensive: 3 • Lack of desired features: 5 • Difficult to customise: 6
16	If you don't use a VPN, why?	• Implement a zero-trust model: 1 • Slow transfer speed: 2 • No need for security: 2 • Cost issue: 3 • No reason: 2
17	What VPN protocol are you using?	• PPTP: 0 • SSTP: 1 • L2TP/IPSec: 2 • IKEv2: 2 • OpenVPN: 14 • Not using VPN: 8
18	If you using a VPN, do you have any complaints about speed?	• Very fast: 0 • Fast: 1 • Moderate speed: 10 • Slow: 11 • So slow: 5
19	Do you do anything other than work when using a VPN? (e.g. surfing the web, watching	• Yes: 21 • No: 6

	videos, etc.)	
20	Does the company have its own servers?	• Yes, using BareMetal, IaaS or PaaS: 24 • No, using SaaS (e.g. Google workspace): 3
21	Is your company adopting a zero-trust model for security?	• Yes (Go to 22): 15 • No: 11 • I don't know: 1
22	If the zero-trust model is being applied, in what form is it applied? (Only if known)	▪ 10 answers, and 3 respondents (30%) answered 'service' for this question.
23	It is predicted that by 2023, 60% of all enterprises will move away from VPNs and shift their security concept to a zero-trust basis. What do you think?	• Use Zero Trust Only: 9 • Still used a lot as a VPN alone: 1 • Combining VPN and Zero Trust: 17 • Others (Go to 25): 0
24	If you have any other comments about 23, feel free to write them.	▪ 0 answer.
25	Thanks! Finally, curious about your opinion. What does zero-trust mean to you?	▪ 9 answers, and 2 respondents (30%) answered 'access' for this question.

Appendix B. Interview

Interview question	Answer 1	Answer 2	Answer 3
What challenges have you faced with the rise of telecommuting since COVID-19?	It is thought that it has become difficult to control individuals as they switch to working from home. Our current office is our own house or cafe. This makes the endpoint susceptible to exposure to the outside and poses a big problem in security. Security solutions such as networks and endpoint protectors are being used, but vulnerability attacks are also	In some cases, companies and schools have to have meetings or classes using group video calls, resulting in security vulnerabilities. In particular, security risks have increased significantly when accessing call content eavesdropping, leakage, and company data, and handling them is consuming a lot of manpower and money.	In our company, developers simply log in through VPN and ordinary employees through SSO. However, when working from home began, business disruptions occurred as internal homepages that were accessible within the company could not be used from outside. So, we urgently built a VPN for general employees, but we have no choice but to be anxious.

	increasing, making it very difficult.		
What do you think is the popularity of WireGuard?	Isn't it because of its high security and fast speed? There is a disadvantage that only UDP is possible, but VPNs created in the past in the Internet environment are very slow these days. Especially when you have to exchange high-capacity data, it's really hard. I think it's a good direction to solve this problem, and I think building a new security algorithm and this well also plays a part.	I think not using SSL plays a big role. In 2014, a very big event occurred as an OpenSSL vulnerability, and in 2015, an attack called FREAK found a variety of vulnerabilities, including vulnerabilities that force downgrade to vulnerable RSA. Wouldn't it be hard to be unpopular if you didn't use it and were also fast?	The new encryption algorithm and its fast transmission speed are very alarming. It was also produced because of its high security and was just released, but the fact that it was installed in the Linux kernel shows that so many people want WireGuard. WireGuard, which caught both rabbits, is difficult to be unpopular with, security and speed.
What do you think of the zero-trust model?	This is a good concept. It is introduced in many places, and it is considered to be the basis for a good design because it requires only a little effort compared to a number of systems that operate organically. Many places recommend introducing and using zero-trust, and it seems to be a really good choice from the perspective of	Once a large castle is built and not just a castle gate, but a lot of castle gates are built in layers to increase accessibility, but it is thought to be a work that continues to be checked through many watchtowers. Now that various services are moving to cloud platforms in the modern Internet environment and servers are often located not only inside but also outside the	Except for existing inevitable services, all are being replaced. As mentioned earlier, ordinary employees did not use VPN because they often access through SSO and work, but now they are using it interchangeably because it is inevitable. However, VPN is a disadvantage of being slow, and security can be weakened because it is too trusted.

	employee user experience.	company, it is considered a very effective and economical security solution.	Therefore, zero-trust is a good concept.
What do you think about the problem of contemporary computer security?	There is no perfect defense. However, there are cases where perfection is sought in many places. Security vulnerabilities can arise as it is developed by humans. Of course, I'm not saying to develop vulnerably. I would like to say that it is important to develop it as safely as possible and then cope with it in the future. I think it is important to immediately correct vulnerabilities when they are found, and to immediately seek and apply countermeasures and security methods when hacking attacks come in.	If there is defense, there is attack. Developers and security experts continue to improve the level of security, but hackers who commit intense attacks as powerful security vulnerabilities appear in proportion to this. There is thorough security, but there is no complete security. Just as the shield develops, the windows become stronger, and this phenomenon is natural for cybersecurity. Therefore, I hope that many developers pay a lot of attention not only to development but also to security.If there is defense, there is attack. Developers and security experts continue to improve the level of security, but hackers who commit intense attacks as powerful security vulnerabilities appear	Compared to the early 2000s, it has matured in many ways, but there are still many people who think lightly of hacking. With the development of the Internet and many things connected to the Internet, it is very difficult to live without it. However, more and more people leak data and spy on privacy through hacking. I hope people will become more mature about these crimes.

		in proportion to this. There is thorough security, but there is no complete security. Just as the shield develops, the windows become stronger, and this phenomenon is natural for cybersecurity. Therefore, I hope that many developers pay a lot of attention not only to development but also to security.	
What direction do you think cybersecurity should take in the future?	Many things are connected to the Internet and will make more progress in the future. Accordingly, many security experts are needed, but in reality, there are not many. I hope the government will help train white hackers along with ethical education. It is important to have practical skills and application skills, not just to study security, because this is often insufficient. It is thought that this will not only benefit companies, but also contribute greatly to becoming a beautiful	There is too much gap in what the general public thinks about cybersecurity. Typically, some people are indifferent to security due to cyber safety insensitivity, while others think it is too easy, such as movies and TV shows. I think it is very important to correct this perception and increase individual security. Just as you lock the door well when you leave the house and carry your wallet safely, I hope cybersecurity will increase security.	Machine learning and deep learning, which will greatly help us in the future, are considered to be the technologies that will lead the world. That means a lot of data, which is a material for analysis. This is directly related to the privacy problem, and many people do not know that privacy is violated or leaked in more areas than expected. Of course, there are convenient or useful cases, but it is necessary to supplement the public's perception and importance of privacy.

	Internet world.		

Appendix C. Code

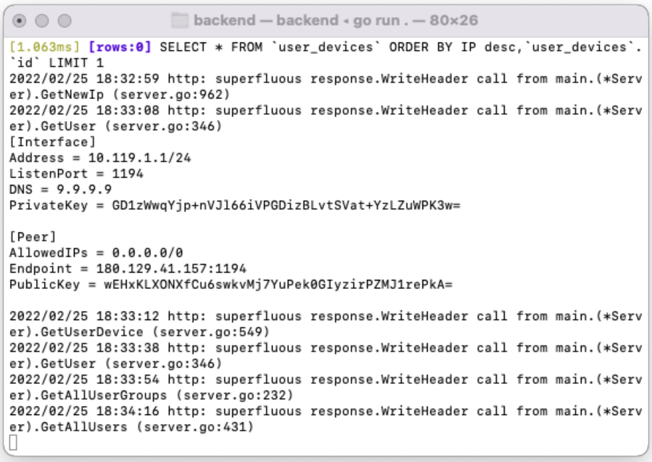
This code is an annectent-root project in vpn-annectent-go, and it is divided into two parts: frontend and backend.

The important source code files for the backend are: server.go, wg.go.

The frontend is auto generated by Svelte except for the amplify folder, and the code is all in the src directory.

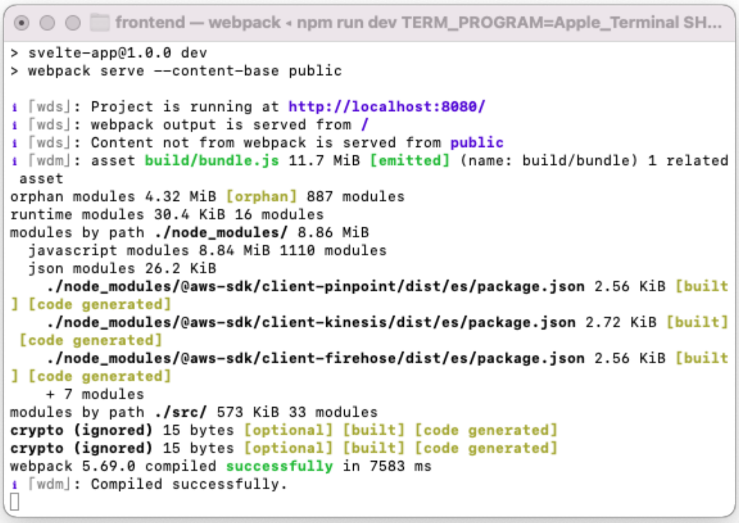
More detailed code can be found on github. https://github.com/leselsey/vpn-annectent-go

Appendix D. Run code



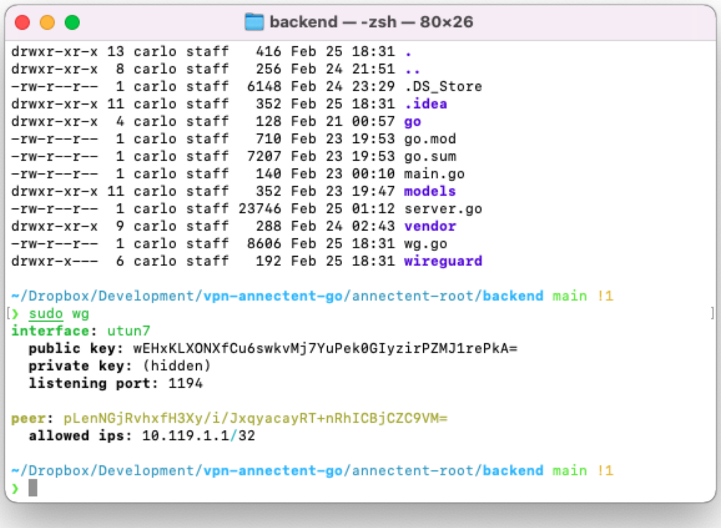
```
[1.063ms] [rows:0] SELECT * FROM `user_devices` ORDER BY IP desc,`user_devices`.
`id` LIMIT 1
2022/02/25 18:32:59 http: superfluous response.WriteHeader call from main.(*Serv
er).GetNewIp (server.go:962)
2022/02/25 18:33:08 http: superfluous response.WriteHeader call from main.(*Serv
er).GetUser (server.go:346)
[Interface]
Address = 10.119.1.1/24
ListenPort = 1194
DNS = 9.9.9.9
PrivateKey = GD1zWwqYjp+nVJl66iVPGDizBLvtSVat+YzLZuWPK3w=

[Peer]
AllowedIPs = 0.0.0.0/0
Endpoint = 180.129.41.157:1194
PublicKey = wEHxKLXONXfCu6swkvMj7YuPek0GIyzirPZMJ1rePkA=

2022/02/25 18:33:12 http: superfluous response.WriteHeader call from main.(*Serv
er).GetUserDevice (server.go:549)
2022/02/25 18:33:38 http: superfluous response.WriteHeader call from main.(*Serv
er).GetUser (server.go:346)
2022/02/25 18:33:54 http: superfluous response.WriteHeader call from main.(*Serv
er).GetAllUserGroups (server.go:232)
2022/02/25 18:34:16 http: superfluous response.WriteHeader call from main.(*Serv
er).GetAllUsers (server.go:431)
```

Go run . – Run backend


frontend — webpack ◂ npm run dev TERM_PROGRAM=Apple_Terminal SH...

```
> svelte-app@1.0.0 dev
> webpack serve --content-base public

ℹ ｢wds｣: Project is running at http://localhost:8080/
ℹ ｢wds｣: webpack output is served from /
ℹ ｢wds｣: Content not from webpack is served from public
ℹ ｢wdm｣: asset build/bundle.js 11.7 MiB [emitted] (name: build/bundle) 1 related
 asset
orphan modules 4.32 MiB [orphan] 887 modules
runtime modules 30.4 KiB 16 modules
modules by path ./node_modules/ 8.86 MiB
  javascript modules 8.84 MiB 1110 modules
  json modules 26.2 KiB
    ./node_modules/@aws-sdk/client-pinpoint/dist/es/package.json 2.56 KiB [built
] [code generated]
    ./node_modules/@aws-sdk/client-kinesis/dist/es/package.json 2.72 KiB [built]
 [code generated]
    ./node_modules/@aws-sdk/client-firehose/dist/es/package.json 2.56 KiB [built
] [code generated]
    + 7 modules
modules by path ./src/ 573 KiB 33 modules
crypto (ignored) 15 bytes [optional] [built] [code generated]
crypto (ignored) 15 bytes [optional] [built] [code generated]
webpack 5.69.0 compiled successfully in 7583 ms
ℹ ｢wdm｣: Compiled successfully.
```

npm run dev – Run frontend



```
drwxr-xr-x 13 carlo staff   416 Feb 25 18:31 .
drwxr-xr-x  8 carlo staff   256 Feb 24 21:51 ..
-rw-r--r--  1 carlo staff  6148 Feb 24 23:29 .DS_Store
drwxr-xr-x 11 carlo staff   352 Feb 25 18:31 .idea
drwxr-xr-x  4 carlo staff   128 Feb 21 00:57 go
-rw-r--r--  1 carlo staff   710 Feb 23 19:53 go.mod
-rw-r--r--  1 carlo staff  7207 Feb 23 19:53 go.sum
-rw-r--r--  1 carlo staff   140 Feb 23 00:10 main.go
drwxr-xr-x 11 carlo staff   352 Feb 23 19:47 models
-rw-r--r--  1 carlo staff 23746 Feb 25 01:12 server.go
drwxr-xr-x  9 carlo staff   288 Feb 24 02:43 vendor
-rw-r--r--  1 carlo staff  8606 Feb 25 18:31 wg.go
drwxr-x---  6 carlo staff   192 Feb 25 18:31 wireguard

~/Dropbox/Development/vpn-annectent-go/annectent-root/backend main !1
> sudo wg
interface: utun7
  public key: wEHxKLXONXfCu6swkvMj7YuPek0GIyzirPZMJ1rePkA=
  private key: (hidden)
  listening port: 1194

peer: pLenNGjRvhxfH3Xy/i/JxqyacayRT+nRhICBjCZC9VM=
  allowed ips: 10.119.1.1/32

~/Dropbox/Development/vpn-annectent-go/annectent-root/backend main !1
>
```

wg – Check WireGuard working