INSTITUTO POLITÉCNICO DE BEJA
Escola Superior de Tecnologia e Gestão
Licenciatura em Engenharia Informática
Projeto Final de Curso

Desenvolvimento de um sistema de comunicações seguras

Martinho José Novo Caeiro - 23917
Paulo António Tavares Abade - 23919
Rafael Conceição Narciso - 24473

Beja, julho de 2025

INSTITUTO POLITÉCNICO DE BEJA
Escola Superior de Tecnologia e Gestão
Licenciatura em Engenharia Informática
Projeto Final de Curso

Desenvolvimento de um sistema de comunicações seguras

Martinho José Novo Caeiro - 23917
Paulo António Tavares Abade - 23919
Rafael Conceição Narciso - 24473

Orientador: Professor Rui Silva

Beja, julho de 2025

Resumo

Neste relatório será abordado o processo de criação de uma solução de comunicações seguras, que permita a troca de mensagens de texto entre os seus utilizadores. Este relatório foi realizado no âmbito da Unidade Curricular de Estágio ou Projeto (IPBeja, 2025).

Keywords: aplicações, cibersegurança, comunicações, criptografia

Abstract

In this report, we will address the creation process of a secure communication solution that allows text message exchange between its users. This report was carried out within the scope of the Curricular Unit of Internship or Project (IPBeja, 2025).

Keywords: applications, cybersecurity, communications, cryptography

Índice

Índice de Figuras

1 Introdução

Para a realização do projeto é necessário desenvolver um sistema de comunicações seguras, que permita a troca de mensagens de texto entre os seus utilizadores. Para tal, é necessário implementar um sistema de autenticação de utilizadores, que permita a criação de contas de utilizador e a autenticação dos mesmos. O sistema deve ser capaz de garantir a confidencialidade, integridade e autenticidade das mensagens trocadas entre os utilizadores.

2 Analise de Requisitos

Para a realização deste projeto, foram necessários os seguintes requisitos:

- **Requisitos Funcionais:**
  - O sistema deve permitir a criação de contas de utilizador.
  - O sistema deve permitir a autenticação de utilizadores.
  - O sistema deve permitir o envio e receção de mensagens entre utilizadores.
  - O sistema deve garantir a confidencialidade, integridade e autenticidade das mensagens trocadas.
- **Requisitos Não Funcionais:**
  - O sistema deve ser seguro e resistente a ataques.
  - O sistema deve ser fácil de usar e intuitivo.
  - O sistema deve ser escalável e capaz de suportar um grande número de utilizadores.

3 Tecnologias Utilizadas

Para o desenvolvimento deste projeto, foram utilizadas as seguintes tecnologias:

- **Linguagens de Programação:** C#, Kotlin
- **Frameworks:** .NET, WireGuard
- **Base de Dados:** SQLite
- **Criptografia:** Rijndael (Vencedor AES), Serpent (Segundo lugar AES)
- **Ferramentas de Desenvolvimento:** Git, Visual Studio Code, Android Studio
- **Serviços de Hospedagem:** GitHub

4 Arquitetura do Sistema

O sistema é composto por uma aplicação que funcionará como um chat, onde os utilizadores que estiverem registado na VPN poderão comunicar entre si. Dentro da aplicação, os utilizadores podem saber quem está associado a um endereço IP da VPN, e assim enviar mensagens para esse utilizador. Ao enviar uma mensagem, a aplicação irá contatar o servidor da VPN, para obter o caminho até ao destinatário, e assim enviar a mensagem. Essa mensagem será encriptada antes de ser enviada, para garantir a confidencialidade e integridade da mensagem. Existe a variação da encriptação utilizada na mensagem, de acordo com uma lógica pré-definida na aplicação, sendo que os métodos de encriptação utilizados são o AES e o Sepernt.

5 Modulos do Sistema

5.1 Interface para Computador

A interface para computador será desenvolvida em C#, cuja qual é uma linguagem rápida e fácil de aprender, e que permite o desenvolvimento de aplicações desktop de forma rápida e eficiente. A aplicação será desenvolvida utilizando o framework .NET, que é um framework de desenvolvimento de aplicações desktop, web e mobile. Esta interface é composta por 3 ecrãs principais:

- **Ecrã de Login:** onde o utilizador pode autenticar-se na aplicação, utilizando as suas credenciais.
- **Ecrã de Lista de Chats:** onde o utilizador pode ver a lista de chats existentes, bem como criar novos chats.
- **Ecrã de Chat:** onde o utilizador pode enviar e receber mensagens de outros utilizadores, bem como ver quem está online.

5.2 Interface para Android

A interface para Android será desenvolvida em Kotlin, que é uma linguagem de programação moderna e concisa, que permite o desenvolvimento de aplicações Android de forma rápida e eficiente.

5.3 Módulo da VPN

A VPN estará alojada num servidor com o sistema operativo AlmaOS 8.10, e será responsável por gerir as ligações dos utilizadores à VPN, bem como a autenticação dos mesmos. O serviço escolhido foi o WireGuard, que é um serviço de VPN de código aberto, leve e de alto desempenho, que utiliza criptografia moderna para garantir a segurança das comunicações. É necessário configurar previamente o WireGuard no dispositivo do utilizador, para que este possa estabelecer uma ligação à VPN.
TODO Tentar fazer com que seja feito da maneira mais automatizada possível.

5.4 Base de Dados

A base de dados será utilizada para armazenar as informações dos utilizadores, como as suas credenciais, bem como as mensagens trocadas entre os utilizadores, esta será implementada utilizando o SQLite.

6 Desenvolvimento

6.1 Interface para Computador

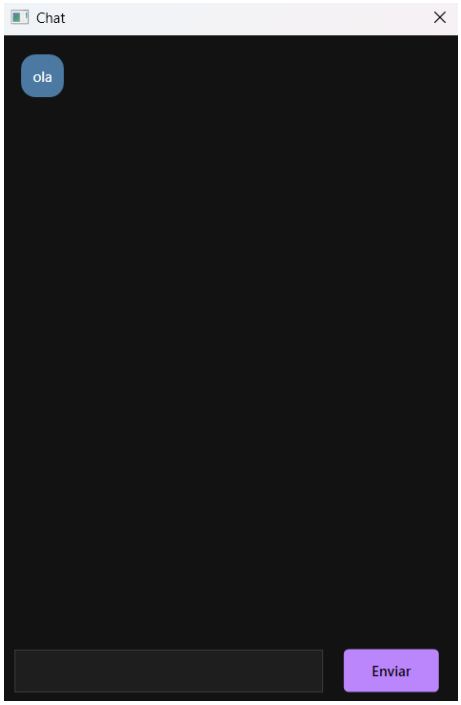
Para o desenvolvimento da interface para computador, foi utilizado o Visual Studio Code, que é um ambiente de desenvolvimento integrado (IDE) da Microsoft.

6.1.1 Ecrã de Login

O ecrã de login é composto por um formulário onde o utilizador pode inserir as suas credenciais, e um botão para autenticar-se na aplicação. Caso não tenha uma conta criada apenas tem que preencher o formulario e clicar no botão de criar conta.

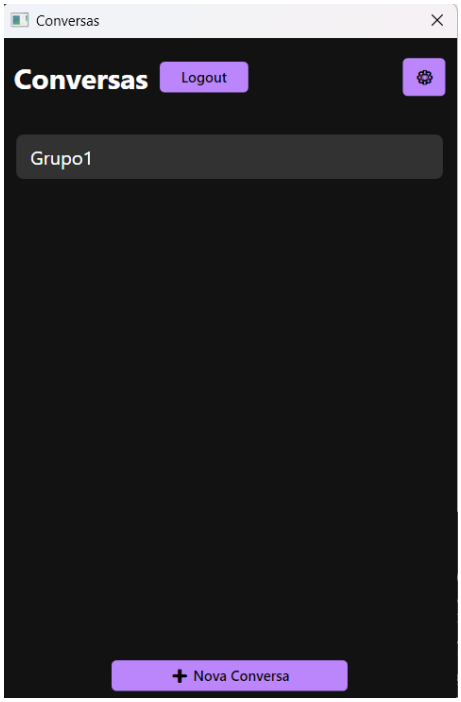
6.1.2 Ecrã de Lista de Chats

O ecrã de lista de chats é composto por uma lista de chats existentes e um botão para criar novos chats. No seu canto superior direito, existe um botão para aceder às definições da aplicação, onde o utilizador pode configurar se quer a aplicação em modo escuro ou claro, e no seu canto inferior esquerdo existe um botão para terminar sessão caso deseja mudar de utilizador.



6.1.3 Ecrã de Chat

O ecrã de chat é composto por uma lista de mensagens trocadas entre os utilizadores, mensagens recebidas ficam do lado esquerdo e mensagens enviadas no lado direito.



6.2 Interface para Android

Para o desenvolvimento da interface para Android, foi utilizado o Android Studio, que é um ambiente de desenvolvimento integrado (IDE) oficial para o sistema operativo Android.

6.3 AlmaOS - Serviço de VPN

Para configurar corretamente o serviço de VPN, é necessário instalar o WireGuard no servidor AlmaOS 8.10, para isso ser feito, foi necessário seguir os seguintes passos:

1. Adicionar o repositório EPEL - sudo dnf install epel-release
2. Adicionar o repositório ELREPO - sudo dnf install https://www.elrepo.org/elrepo-release-8.el8.elrepo.noarch.rpm

3. Ativar o CodeReady Builder - sudo /usr/bin/crb enable

4. Atualizar os metadados - sudo dnf makecache

5. sudo dnf –enablerepo=elrepo install kmod-wireguard -y

6. Instalar o WireGuardTools - sudo dnf install wireguard-tools -y

Para verificar se o WireGuard está instalado corretamente, pode-se utilizar o comando **"sudo modprobe wireguard"**, se este não devolver nenhum output, significa que o WireGuard está instalado corretamente.

Agora, para configurar o WireGuard, é necessário criar as chaves de criptografia, para isso é necessário ir para a pasta **/etc/wireguard** e executar o seguintes comando:

```
wg genkey | tee server_private.key | wg pubkey > server_public.key
chmod 600 server_private.key
```

Isto irá gerar duas chaves, uma privada e uma pública, que serão utilizadas para autenticar os utilizadores na VPN.

Após isso, é necessário criar o ficheiro de configuração do WireGuard, onde estará definida a configuração da VPN em si, como o endereço IP da VPN, a porta de escuta, as chaves de criptografia, entre outros. Para isso, é necessário criar o ficheiro **wg0.conf** na pasta /**etc**/**wireguard**, e adicionar o seguinte conteúdo:

```
[Interface]
PrivateKey = <Chave Privada do Servidor>
Address = 10.0.0.1/24
ListenPort = 51820
SaveConfig = true
[Peer]
PublicKey = <Chave Pública do Cliente>
AllowedIPs = 10.0.0.2/32
```

Para garantir o bom funcionamento da VPN, é necessário ativar o encaminhamento de endereços IP, para assim permitir o tráfego de rede. Para isso, é necessário fazer o seguinte comando:

```
# Para permitir o tráfego de rede
echo "net.ipv4.ip_forward = 1" | sudo tee -a /etc/sysctl.conf
sudo sysctl -p

# Para permitir o tráfego pela porta 51820
sudo firewall-cmd --add-masquerade --permanent
sudo firewall-cmd --add-port=51820/udp --permanent
sudo iptables -t nat -A POSTROUTING -s 10.0.0.0/24 -o enp0s3 -j MASQUERADE
sudo firewall-cmd --reload
```

Para ser possível encaminhar o tráfego de rede, é necessário adicionar uma regra ao Router para que, todos os pacotes que cheguem à porta 51820 sejam encaminhados para o servidor WireGuard. Por fim, basta iniciar o serviço do WireGuard, para isso é necessário executar o seguinte comando:

```
sudo systemctl enable --now wg-quick@wg0
```

Do lado do cada cliente, vai ser necessário configurar o WireGuard, de maneira a que apenas a informação proveniente da aplicação desenvolvida seja enviada através da VPN, onde assim serão poupados recursos do dispositivo que está a fazer o papel de servidor.

6.4 Base de Dados

A base de dados irá ter a seguinte estrutura:

- **User:**
  - UserID (chave primária)
  - Username (string)
  - Password (string)
- **Chat:**
  - ChatID (chave primária)
  - Name (string)
  - AdminID (chave estrangeira, referência à tabela User)
- **Participant:**
  - ParticipantID (chave primária)
  - ChatID (chave estrangeira, referência à tabela Chat)
  - UserID (chave estrangeira, referência à tabela User)

- **Message:**
  - MessageID (chave primária)
  - ParticipantID (chave estrangeira, referência à tabela Participant)
  - Content (string)
  - Date (data e hora da mensagem)
  - SenderUserID (chave estrangeira, referência à tabela Participant)

7 Testes

8 Conclusão

Concluimos assim que o projeto foi um sucesso, pois conseguimos desenvolver um sistema de comunicações seguras que permite a troca de mensagens de texto entre os seus utilizadores, sem que haja a possibilidade de terceiros acederem às mensagens trocadas. Em geral, foi um projeto que nos permitiu aprender bastante sobre a área de cibersegurança e comunicações seguras, e que nos possibilitou aplicar os conhecimentos adquiridos ao longo do curso de Engenharia Informática.

Bibliografia

IPBeja. (2025). *Disciplina: Estágio ou Projeto | IPBeja* [Página EP]. Obtido junho 16, 2025, de https://cms.ipbeja.pt/course/view.php?id=238