|  |  |  |  |
| --- | --- | --- | --- |
| A picture containing icon  Description automatically generated | | Curso de TeSP em  Programação de Sistemas de Informação  Tópicos de Segurança | |
|  |  | |  |
| **Ano Letivo 2024/2025** | **Av. Periódica** | | **1º Ano, 2º Semestre** |

|  |
| --- |
| **PROJETO DE TÓPICOS DE SEGURANÇA** |

**Relatório de Especificação de Requisitos do projeto de Tópicos de Segurança**

|  |  |  |
| --- | --- | --- |
| **Turno**: PL3 | **Grupo**: A (se aplicável) | **Docente**: Nuno Simões |
| **Nº 2241567** | Alexandre Conrado | |
| **Nº 2241572** | Diogo Soares | |
| **Nº 2241570** | Guilherme Silva | |

**Índice**

[1 Introdução 5](#_Toc67476478)

[2 Especificação do Sistema 6](#_Toc67476479)

[2.1 Especificação de Requisitos 6](#_Toc67476480)

[2.1.1 Requisitos Funcionais (RF) 6](#_Toc67476481)

[2.1.2 Requisitos Não Funcionais (RNF) 7](#_Toc67476482)

[2.1.3 Wireframes UI 14](#_Toc67476483)

[3 Conclusão 15](#_Toc67476484)

**Índice de tabelas**

[Tabela 1 Requisitos Funcionais 8](#_Toc201006952)

[Tabela 2 Requisitos Não Funcionais de Usabilidade 9](#_Toc201006953)

[Tabela 3 Requisitos Não Funcionais de Fiabilidade 10](#_Toc201006954)

[Tabela 5 Requisitos Não Funcionais de Eficiência 12](#_Toc201006955)

[Tabela 6 Requisitos Não Funcionais de Disponibilidade 13](#_Toc201006956)

[Tabela 7 Requisitos Não Funcionais de Ambiente 14](#_Toc201006957)

[Tabela 8 Requisitos Não Funcionais de Desenvolvimento 15](#_Toc201006958)

# Introdução

Este relatório é realizado no âmbito da unidade curricular de Tópicos de Segurança do curso Técnico Superior Profissional em Programação de Sistemas de Informação, no ano letivo de 2024/2025. O objetivo do projeto elaborado é o desenvolvimento de um software de comunicação segura entre clientes e servidor, através de um chat desenvolvendo em C# e utilizando mecanismos de criptografia e autenticação.

Este software é constituído por dois módulos: o módulo sobre cliente, com uma interface gráfica e um módulo sobre servidor, em consola. Para garantir a segurança das comunicações, foram implementadas técnicas de criptografia assimétrica e simétrica, como assinaturas digitais para a validação da integridade das mensagens que são trocadas pelos utilizadores.

O projeto segue os princípios de confidencialidade, integridade e autenticidade, pilares fundamentais da segurança da informação, e recorre à biblioteca ProtocolSlpara o tratamento estruturado da comunicação entre cliente e servidor. O mesmo relatório apresenta as especificações de requisitos do sistema, tanto os requisitos funcionais como os não funcionais.

# Especificação do Sistema

O problema abordado neste projeto é a necessidade de um chat seguro, que permita a comunicação entre clientes e servidor com proteção avançada contra interceções e alterações de dados. Atualmente, muitas soluções não garantem uma camada forte de segurança na transmissão de mensagens, deixando os utilizadores vulneráveis a possíveis ataques, como roubo de credenciais, falsificação de identidade e manipulação de mensagens e dados.

Para resolver este problema, o sistema proposto deve implementar técnicas avançadas de criptografia simétrica e assimétrica, garantindo que as mensagens que forem trocadas estejam protegidas contra acessos não autorizados. Além disso, a autenticação dos utilizadores e a validação de mensagens com assinaturas digitais são fundamentais para garantir a integridade das comunicações.

## Especificação de Requisitos

### Requisitos Funcionais (RF)

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RF-01 | O software deve permitir uma autenticação para o utilizador no servidor, garantindo a segurança das credenciais do utilizador. | Alta | X |
| RF-02 | Os clientes devem ser capazes de enviar e receber mensagens com segurança através de um sistema de criptografia. | Alta | X |
| RF-03 | O software deve conseguir validar a autenticidade das mensagens utilizando as assinaturas digitais. | Alta | X |
| RF-04 | O servidor deve conseguir aceitar várias ligações simultâneas de diversos clientes. | Alta | X |
| RF-05 | O cliente deve criar e enviar a chave pública para o servidor enquanto faz a conexão inicial. | Média | X |
| RF-06 | O cliente deve disponibilizar uma interface gráfica intuitiva e funcional para o fácil manuseio. | Média | X |
| RF-07 | O servidor deve ser capaz de armazenar com segurança a chave pública dos clientes para as suas futuras coneções. | Média | X |
| RF-08 | O servidor deve criar um ficheiro de logs incluindo todas as interações relevantes entre servidor e cliente. | Média | X |

Tabela 1 Requisitos Funcionais

### Requisitos Não Funcionais (RNF)

#### Requisitos Não Funcionais de Usabilidade

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RNF-USA-01 | A interface gráfica do cliente deve ser uma interface intuitiva, para facilitar a interação do utilizador. | Alta | X |
| RNF-USA-02 | O software deve mostrar mensagens de erro e feedback ao utilizador assim que o mesmo faça uma ação. | Alta | X |
| RNF-USA-03 | A navegação dentro da aplicação deve ser uma navegação de fácil compreensão. | Média | X |

Tabela 2 Requisitos Não Funcionais de Usabilidade

#### Requisitos Não Funcionais de Fiabilidade

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RNF-FIA-01 | O software deve conseguir se recuperar rapidamente após algum tipo de falha inesperada. | Alta | X |
| RNF-FIA-02 | O servidor deve manter uma estabilidade entre as conexões mesmo em situações de uma carga elevada | Média | X |

Tabela 3 Requisitos Não Funcionais de Fiabilidade

#### Requisitos Não Funcionais de Segurança

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RNF-SEG-01 | O sistema deve conter uma criptografia assimétrica para a troca de chaves públicas. | Alta | X |
| RNF-SEG-02 | O sistema deve utilizar uma criptografia simétrica para proteger a confidencialidade das mensagens enviadas. | Alta | X |

Tabela 4 Requisitos Não Funcionais de Segurança

#### Requisitos Não Funcionais de Eficiência

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RNF-EFI-01 | O sistema deve garantir um alto desempenho e um tempo de resposta rápido. | Alta | X |
| RNF-EFI-02 | A aplicação deve leve em nível de requisitos computacionais. | Média | X |

Tabela 5 Requisitos Não Funcionais de Eficiência

#### Requisitos Não Funcionais de Disponibilidade

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RNF-DIS-01 | O servidor deve manter um tempo de atividade duradouro, assim evitará um período de inatividade. |  | X |
| RNF-DIS-02 | O sistema deve ser capaz de lidar com acessos simultâneos sem alterar e prejudicar a experiência do utilizador. |  | X |

Tabela 6 Requisitos Não Funcionais de Disponibilidade

#### Requisitos Não Funcionais de Ambiente

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RNF-AMB-01 | O software deve ser compatível com os sistemas operativos do Windows. | Alta | X |
| RNF-AMB-02 | O código deve ser modular para permitir futuras expansões e alterações no mesmo. | Média | X |

Tabela 7 Requisitos Não Funcionais de Ambiente

#### Requisitos Não Funcionais de Desenvolvimento

|  |  |  |  |
| --- | --- | --- | --- |
| # ID | Descrição | Prioridade | Implementado |
| RNF-DES-01 | O código deve ser bem estruturado e bem documentado para providencial alguma futura manutenção. | Alta | X |
| RNF-DES-02 | A arquitetura do software deve permitir uma escalabilidade e uma adição de novas funcionalidades. | Alta | X |

Tabela 8 Requisitos Não Funcionais de Desenvolvimento

### Wireframes UI

Uma imagem com texto, captura de ecrã, diagrama, file

Os conteúdos gerados por IA poderão estar incorretos.

# Conclusão

Ao realizar este projeto deu a possibilidade de desenvolver um software de chat seguro, com uma comunicação entre múltiplos clientes e o servidor, foi comprido os objetivos propostos na unidade curricular de Tópicos de Segurança. Ao longo do desenvolvimento do mesmo software, foram implementados mecanismos de segurança, nomeadamente criptografia assimétrica para troca de chaves, criptografia simétrica para comunicação, autenticação segura dos utilizadores e validação das mensagens com as assinaturas digitais.

Além da camada de segurança, foi realizado as boas práticas de desenvolvimento, como a modularidade do código, a utilização de logs para registo das ações no servidor e uma criação de uma interface gráfica intuitiva e funcional para os utilizadores. O sistema mostrou-se eficaz na gestão de múltiplas ligações simultâneas, permitindo um bom desempenho mesmo sob carga elevada.