**UNITOLEDO**

**ÍGOR HENRIQUE SCUCULHA SOARES**

**JOÃO VITOR PEDON BONTEMPO**

**VICTOR DE AZEVEDO SANTATERRA**

**PROJETO: SISTEMAS ESTRUTURADOS**

**CRIPTOGRAFIA DE TEXTOS E ARQUIVOS**

**ARAÇATUBA**

**2020**

**Ígor Henrique Scuculha Soares**

**João Vitor Pedon Bontempo**

**Victor de Azevedo Santaterra**

**PROJETO: SISTEMAS ESTRUTURADOS**

**CRIPTOGRAFIA DE TEXTOS E ARQUIVOS**

Trabalho solicitado pelo professor Evandro Estevam, da disciplina Projeto: Sistemas Estruturados, do curso Análise e desenvolvimento de sistemas, turno noturno da Instituição UniToledo.

**ARAÇATUBA**

**2020**

SUMÁRIO

[1. Levantamento de requisitos 4](#_Toc42707222)

[1.1 Introdução 4](#_Toc42707223)

[1.2 Oportunidade de negócio 4](#_Toc42707224)

[1.3 Visão geral do produto 4](#_Toc42707225)

[1.4 Problemas 4](#_Toc42707226)

[2. Análise 5](#_Toc42707227)

[2.1 Modelo Ambiental 5](#_Toc42707228)

[2.1.1 Objetivos do sistema (Requisitos) 5](#_Toc42707229)

[2.1.2 Lista de eventos 8](#_Toc42707230)

[2.1.3 Diagrama de contexto 11](#_Toc42707231)

[2.2 Modelo comportamental 11](#_Toc42707232)

[2.2.1 DER – Diagrama entidade-relacionamento 11](#_Toc42707233)

[2.2.2 Modelo relacional 11](#_Toc42707234)

[2.2.3 Diagrama de fluxo de dados 11](#_Toc42707235)

[2.2.4 Especificação de processos 25](#_Toc42707236)

[3. Engenharia de Software 30](#_Toc42707237)

[4. Projeto 31](#_Toc42707238)

[4.1 Projeto de interface 31](#_Toc42707239)

[4.2 Ambiente do sistema 40](#_Toc42707240)

[4.2.1 Ambiente de desenvolvimento 40](#_Toc42707241)

[4.2.2 Ambiente de execução 40](#_Toc42707242)

[5. Referências 41](#_Toc42707238)

**1. Levantamento de requisitos**

**1.1 Introdução**

Este documento tem como objetivo detalhar, analisar e definir as funcionalidades e necessidades do sistema iCrypto. De maneira geral, este sistema concentra-se na utilização de cinco tipos de cifras de criptografia diferentes que podem ser utilizadas e exploradas pelo usuário. O objetivo da criação deste software é proporcionar uma maior segurança da informação de forma gratuita para qualquer usuário.

**1.2 Oportunidade de negócio**

O software iCrypto, tendo como base dar o suporte a criptografar textos e arquivos, contém uso gratuito nos primeiros dois meses com todas as funcionalidades disponíveis, após isso será cobrado um único pagamento obrigatoriamente por trinta reais e noventa centavos, tendo acesso a todas opções que o software fornece.

**1.3 Visão geral do produto**

O software iCrypto tem como principal funcionalidade a execução e manipulação da criptografia dos dados do usuário, seja em formato de texto ou qualquer formato de arquivo, em diversos modelos de criptografia, assim como a verificação das propriedades dos arquivos gerados nesses determinados modelos

O Sistema possui recursos de criação de conta, que deve ser efetuada obrigatoriamente antes de seu uso geral, além da manipulação dos dados registrados e armazenados na mesma, sendo estes, salvos em um registro de histórico de atividades.

**1.4 Problemas**

|  |  |
| --- | --- |
| **Problema:** | O usuário precisa enviar uma mensagem ou arquivo na qual seu conteúdo possui um alto valor e este conteúdo não pode ser visto por pessoas não autorizadas. |
| **Afeta:** | Empresas/usuários em geral |
| **Cujo impacto é:** | A interceptação das informações valiosas de uma empresa/usuário podem causar impactos incalculáveis dependendo da manipulação dos dados que será feita pelo interceptador. |
| **Benefícios de uma solução seriam:** | Garantir que os dados enviados, mesmo que interceptados, estejam ilegíveis. Dessa forma, não há a possibilidade de manipular a informação. |

# 

# Análise

# Modelo Ambiental

# 2.1.1 Objetivos do sistema (Requisitos)

a) Requisitos funcionais

|  |  |
| --- | --- |
| **Identificador** | **Descrição** |
| **RF001** | O sistema deve cadastrar o nome, e-mail, nome de usuário e senha dos usuários. |
| **RF002** | Deve-se permitir que o usuário cadastre um servidor SMTP personalizado para o envio de e-mails. |
| **RF003** | O sistema apresentará uma tela de login e só deve ser iniciado caso o usuário esteja logado com sua conta. |
| **RF004** | Caso o usuário esqueça sua senha, deve ser enviado um e-mail (que será informado pelo usuário) ao usuário com as informações de sua conta e uma nova senha gerada pelo sistema. Em caso de e-mail inválido cadastrado, deve-se permitir a utilização de um e-mail alternativo na qual o usuário tenha acesso. |
| **RF005** | O sistema deve criptografar textos digitados pelo usuário com as seguintes cifras: Cifra de César, RSA, AES e Código Morse. |
| **RF006** | O sistema deve gerar automaticamente chaves criptográficas do RSA no tamanho especificado pelo usuário. |
| **RF007** | O sistema deve criptografar arquivos selecionados pelo usuário com a cifra AES. |
| **RF008** | O sistema deve executar a técnica de esteganografia com arquivos selecionados pelo usuário e gerar um novo arquivo esteganografado, que será salvo em uma pasta especificada pelo usuário. |
| **RF009** | O sistema deve decifrar mensagens de texto das seguintes cifras: Cifra de César, RSA, AES e Código Morse. |
| **RF010** | O sistema deve descriptografar arquivos com a cifra AES. |
| **RF011** | O usuário pode enviar suas mensagens ou arquivos criptografados por e-mail diretamente do sistema. |
| **RF012** | Deve-se permitir a cópia das mensagens criptografadas pelo usuário por meio de um botão. |
| **RF013** | No RSA, o usuário pode salvar suas chaves, tanto pública quanto privada, de quatro métodos diferentes: impressão, salvar em arquivos .txt, copiar e enviar por e-mail. |
| **RF014** | As chaves do algoritmo RSA podem ser enviadas por e-mail como texto ou arquivo .txt. |
| **RF015** | Um botão para limpar mensagens deve ser colocado em todas as criptografias de texto, tanto para saída quanto para entrada de dados. |
| **RF016** | Um botão para colar mensagens deve ser colocado em todas as criptografias de texto, apenas para entrada de dados. |
| **RF017** | No Menu Principal, haverá uma tela para que o usuário possa alterar e/ou excluir as informações do seu cadastro. |
| **RF018** | O sistema deve apresentar uma tela pra que o usuário acesse o histórico de atividades no sistema, cifra por cifra. |
| **RF019** | Para cifras de texto, o histórico deve armazenar o modo (criptografar/decifrar), a mensagem de entrada, a mensagem de saída e a data em que foi feita a atividade pelo usuário. |
| **RF020** | Para a esteganografia, o histórico deve armazenar as seguintes informações: Nome do arquivo original, do arquivo a ser ocultado, do arquivo esteganografado e a data da atividade. |
| **RF021** | O e-mail só deve ser enviado se o usuário inserir os dados do seu e-mail, caso contrário, o sistema deve emitir uma mensagem de que não será possível enviar o e-mail. |
| **RF022** | Caso seja um e-mail da Google, o sistema deve perguntar ao usuário se a opção “Acesso a Apps Menos Seguros” está ativada em seu gmail. Em caso afirmativo, o sistema deve enviar o e-mail normalmente. Caso contrário, o usuário deve ser redirecionado para o link na qual é possível ativar esta opção. |
| **RF023** | Quando o usuário retornar para o sistema após o link de ativação “Acesso a Apps Menos Seguros” da Google ser aberto, o sistema deve perguntar novamente se a opção foi ativada. Se sim, o e-mail deve ser enviado normalmente. Se não, deve ser emitida uma mensagem de que não será possível enviar o e-mail. |
| **RF024** | Antes de enviar um arquivo de texto no RSA, deve-se perguntar ao usuário se deseja salvar o arquivo novamente. Se o arquivo não existir, o sistema deve construí-lo e salvá-lo em um diretório selecionado pelo usuário. |
| **RF025** | Nas cifras Código Morse e Cifra de Cesar, as mensagens devem ser cifradas em tempo real. |
| **RF026** | A exclusão de mensagens do histórico só devem ocorrer após confirmação pelo usuário. |
| **RF027** | Os sistemas de envio de arquivos anexados por e-mail deverão realizar a verificação dos tamanhos dos arquivos a serem enviados por e-mail. Caso ultrapassem 25MB o total, o sistema perguntará ao usuário se deseja compactar todos eles em um .rar, com a finalidade de diminuir o tamanho. |
| **RF028** | O sistema de envio de arquivos do AES não deverá permitir que o usuário envie arquivos executáveis por e-mail. |
| **RF029** | O sistema de criptografia de arquivos do AES deve permitir ao usuário selecionar um arquivo de cada vez para criptografar/descriptografar, ou escolher algum diretório, assim o sistema pegará todos os arquivos dentro desse diretório. |
| **RF030** | Os sistemas de enviar e-mail do AES e Esteganografia deve conter botões de anexar e remover arquivos no datagrid. |
| **RF031** | Os forms das cifras de AES e RSA devem armazenar no banco as informações de cifragem, decifragem logo após de realizar a operação, com exceção das cifras Código Morse e Cifra de Cesar, em que o processo será feito por meio de um botão. |
| **RF032** | Para as todas cifras de Mensagem de Texto, o campo de entrada deve conter as opções de "Copiar", "Colar" e "Limpar", enquanto o campo de saída deve conter "Copiar" e "Limpar".Exceto para Cifra de Cesar e Código Morse quanto ao "Limpar" no campo saída. |
| **RF033** | O sistema do AES deverá disponibilizar a opção de trocar de senha secreta. |
| **RF034** | O sistema só deve ser finalizado na tela de login. Caso o usuário feche alguma outra janela, o sistema deve retornar para a tela anterior. |
| **RF035** | Para Cifra de Cesar o histórico deve mostrar a chave utilizada. |
| **RF036** | Para Cifra de Cesar e Código Morse, o sistema deve permitir o usuário salvar automaticamente no histórico de atividades. |
| **RF037** | Para Cifra de Cesar e Código Morse, o sistema deve permitir o usuário resetar os campos. |
| **RF038** | Para Cifra de Cesar, o sistema deve permitir o usuário um botão de soma e subtração de valores unitários da chave. |

b) Requisitos não funcionais

|  |  |
| --- | --- |
| **Identificador** | **Descrição** |
| **RNF001** | As senhas devem ser armazenadas no banco de dados por hash md5. |
| **RNF002** | As senhas devem ser validadas por um algoritmo matemático. |
| **RNF003** | Os nomes de usuário devem contem pelo menos 8 caracteres. |
| **RNF004** | O sistema deve detectar automaticamente os seguintes servidores de e-mail SMTP e suas respectivas portas: gmail.com, hotmail.com, live.com e outlook.com. |
| **RNF005** | O banco de dados deve ser gerado automaticamente na pasta Arquivos de Programas(x86) do usuário. |
| **RNF006** | O sistema deve ser compatível apenas com o SO Windows. |
| **RNF007** | A senha secreta do AES para 128 bits deve ser criado a partir da hash md5. |
| **RNF008** | A senha secreta do AES para 256 bits deve ser criado a partir da hash SHA256. |
| **RNF009** | A cifragem e decifragem de textos do AES deve ter retorno de menos de 3 segundos. |
| **RNF010** | Na cifra AES, uma barra de progressão deve indicar ao usuário como está o andamento dos processos de criação e cadastro de chaves e de criptografia de arquivos. |
| **RNF011** | O arquivo a ser descriptografado no AES necessita estar com a última extensão ".aes", caso contrário não será possível realizar a operação. |
| **RNF012** | O sistema deve suportar as chaves criptográficas do RSA de 512, 1024, 2048, 4096 e 8192 bytes. |
| **RNF013** | O software deve conter como fonte padrão "Verdana". |
| **RNF015** | A senha secreta do AES não deve ser vazia. |
| **RNF016** | O campo de texto de entrada de dados do AES não deve ser vazio, caso contrário a criptografia não irá acontecer. |
| **RNF017** | O botão de "Enviar cifragem por e-mail" do AES só deve ser funcional caso o "Modo" seja "Criptografar" |
| **RNF018** | O botão de "Salvar em" do AES só deve ser funcional caso algum texto seja criptografado. |
| **RNF019** | O botão "Criptografar" de arquivo do AES só será funcional caso tenha ao menos um arquivo no datagrid. |
| **RNF020** | O botão "Descriptografar" de arquivo do AES só será funcional caso tenha ao menos um arquivo no datagrid. |
| **RNF021** | O campo de texto de saída dos textos deverá ser limpo assim que o usuário selecionar o modo "Descriptografar" |
| **RNF022** | O label de status da operação de criptografar e descriptografar arquivos do AES deverá mostrar assim que for realizado uma criptografia, uma descriptografia ou quando houver algum erro. |
| **RNF023** | O sistema AES deverá trocar o nome do arquivo selecionado do datagrid assim que for clicado em cima, jogando-o no textbox de seleção de arquivos. |
| **RNF024** | As janelas devem ser fixas para os forms de AES. |
| **RNF025** | As janelas de AES não devem conter "Minimizar" e "Maximizar". |
| **RNF026** | A técnica de esteganografia só deve ser aplicada caso o nome do arquivo escolhido pelo usuário seja validado. |
| **RNF027** | Todas as telas devem ser iniciadas no centro. |
| **RNF028** | A tela de "Salvar em" do AES deve disponibilizar opções de salvar : ".txt", ".doc", ".docx" , ou "Todos". |
| **RNF029** | O Sistema deverá salvar no histórico de atividades automaticamente a cada 5 segundos nas cifras em Cifra de Cesar e Código Morse o que foi alterado em tempo real caso o CheckBox esteja ativado. |
| **RNF030** | Para Código Morse e Cifra de Cesar, os botões "Salvar" e "Enviar Email", devem ser bloqueados pelo sistema caso não haja texto no campo de saída. |
| **RNF031** | Para Cifra de Cesar, o sistema deve reiniciar o processo em tempo real selecionado, caso o valor da chave seja mudado |
| **RNF032** | Para Cifra de Cesar, os botões de soma e subtração dos valores da chave devem estar limitados em um valor entre 0 e 25 |

# 2.1.2 Lista de eventos

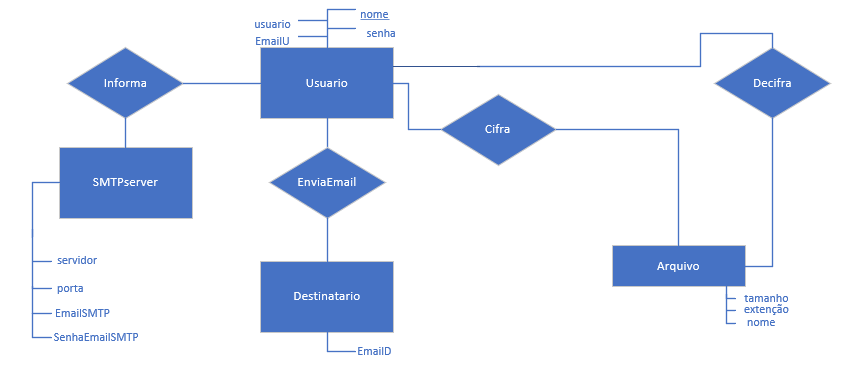
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **N°** | **Evento** | **Estímulo** | **Resposta** | **Tipo de Evento** |
| **1** | Usuário gera chave AES | Senha secreta, Tamanho da chave | Armazena chave secreta | FD |
| **2** | Usuário criptografa texto AES | Texto puro, texto cifrado | Texto cifrado | FD |
| **3** | Usuário descriptografa texto AES | Texto cifrado | Texto original | FD |
| **4** | Usuário criptografa arquivo AES | Arquivo | Arquivo cifrado | FD |
| **5** | Usuário descriptografa arquivo AES | Arquivo cifrado | Arquivo decifrado | FD |
| **6** | Usuário gera chave RSA | Tamanho da chave | Retorna chave | FD |
| **7** | Usuário exporta chaves RSA por cópia | Chave RSA | Retorna chave | FD |
| **8** | Usuário importa chaves RSA | Chave RSA | Armazena chave secreta | FD |
| **9** | Usuário criptografa com RSA | Mensagem normal, chave atual | Mensagem cifrada com RSA | FD |
| **10** | Usuário aplica técnica de esteganografia | arquivo original, arquivo para esconder, nome do arquivo gerado, diretório do arquivo gerado | Retorna arquivo esteganografado | FD |
| **11** | Usuário envia mensagem cifrada por e-mail | Mensagem saída, dados e informações do e-mail | Enviar e-mail | FD |
| **12** | Usuário cifra em Código Morse | Texto Puro | Retorna Texto Cifrado | FD |
| **13** | Usuário decifra em Código Morse | Texto Cifrado | Retorna Texto Decifrado | FD |
| **14** | Usuário realiza cadastro | Dados do usuário | Armazenar informações do usuário | FD |
| **15** | Usuário solicita recuperação de senha | E-mail destinatário | Envia e-mail com nova senha | FD |
| **16** | Usuário envia e-mail com anexos (Esteganografia) | Dados, informações e anexos do e-mail | Envia e-mail | FD |
| **17** | Usuário decifra com RSA | Mensagem cifrada, chave atual | Mensagem decifrada | FD |
| **18** | Usuário realiza login | Dados usuário | Menu principal | FD |
| **19** | Usuário cifra em Cifra de Cesar | Texto Puro, chave | Retorna texto cifrado | FD |
| **20** | Usuário decifra em Cifra de Cesar | Texto Cifrado | Retorna texto decifrado | FD |
| **21** | Sistema Salva automaticamente em Cifra de Cesar | Texto entada, Texto Saída, Chave, modo Cifragem | Armazena Informações no Histórico | FT |
| **22** | Sistema Salva automaticamente em Código Morse | Texto entada, Texto Saída, modo Cifragem | Armazena Informações no Histórico | FT |
| **23** | Usuário Salva Informações em Cifra de Cesar | Texto entada, Texto Saída, Chave, modo Cifragem | Armazena Informações no Histórico | FD |
| **24** | Usuário Salva Informações em Código Morse | Texto entada, Texto Saída, modo Cifragem | Armazena Informações no Histórico | FD |
| **25** | Usuário Copia Texto | Texto Selecionado | Armazena na Área de Transferência | FD |
| **26** | Usuário exporta chaves RSA por impressão | tipo de chave, chave, impressora | Retorna chave | FD |
| **27** | Usuário exporta chaves RSA por e-mail | tipo de chave, chave, e-mail destinatário, dados e informações do e-mail | Retorna chave | FD |
| **28** | Usuário exporta chaves RSA por arquivo.txt | tipo de chave, chave, caminho do arquivo | Retorna chave | FD |
| **29** | Usuário Cola Texto | Campo de Texto Selecionado | Retorna Texto da Area de Transferência | FD |

### 2.1.3 Diagrama de contexto

# 

### 2.2 Modelo comportamental

### 2.2.1 DER – Diagrama entidade-relacionamento

****

### 2.2.2 Modelo relacional

Usuario {nome, usuário, senha, EmailU};

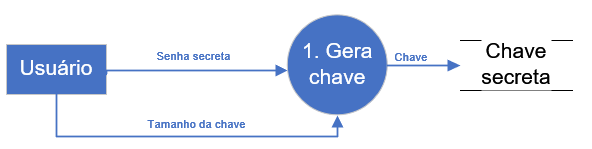
SMTPserver {servidor,porta,EmailSMTP,SenhaEmailSMTP};

Destinatario {EmailD};

Arquivo {tamanho, extenção, nome}

### 2.2.3 Diagrama de fluxo de dados

**Evento 1: Usuário gera chave AES**

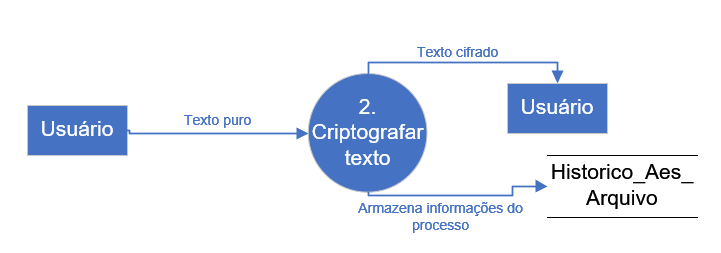


**Senha secreta:** Uma senha digitada pelo usuário podendo conter todo e qualquer caractere.

**Tamanho da chave:** [128,256].

**Chave:** Chave é guardada no sistema.

**Evento 2: Usuário criptografa texto AES**

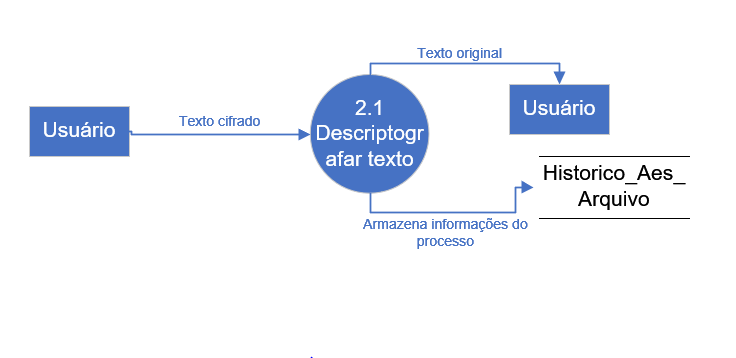


**Texto puro:** Texto original digitado ou copiado pelo usuário.

**Texto cifrado:** Texto cifrado retornado do sistema para o usuário.

**Armazena informações do processo:** Informações que serão armazenadas no banco de dados, sendo ela: Modo, Texto entrada, Texto saída e data.

**Evento 3: Usuário descriptografa texto AES**

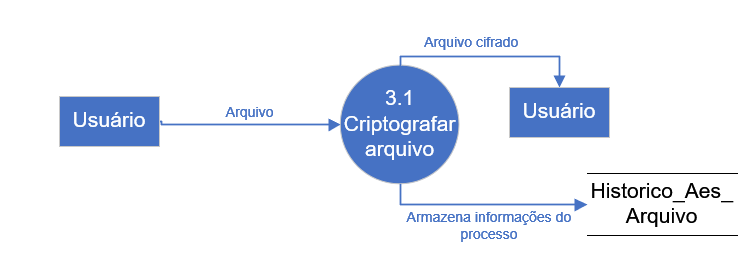


**Texto cifrado:** Texto criptografado inserido pelo usuário.

**Texto original:** Texto de retorno do sistema para o usuário, contendo o texto original.

**Armazena informações do processo:** Informações que serão armazenadas no banco de dados, sendo ela: Modo, Texto entrada, Texto saída e data.

**Evento 4: Usuário criptografa arquivo AES**

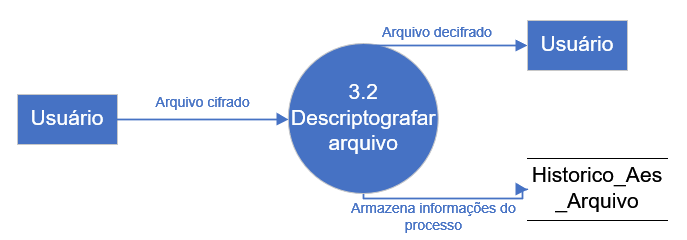


**Arquivo:** Arquivo inserido pelo usuário (Arquivo original)

**Arquivo cifrado:** Arquivo retornado do sistema para o usuário, sendo o arquivo criptografado.

**Armazena informações do processo:** Informações que serão armazenadas no banco de dados, sendo ela: Modo, Texto entrada, Texto saída e data.

**Evento 5: Usuário descriptografa arquivo AES**



**Arquivo cifrado:** Arquivo cifrado inserido pelo usuário.

**Arquivo decifrado:** Arquivo descriptografado retornado do sistema para o usuário.

**Armazena informações do processo:** Informações que serão armazenadas no banco de dados, sendo ela: Modo, Texto entrada, Texto saída e data.

**Evento 6: Usuário gera chave RSA**



Dicionário de fluxo de dados:

**Tamanho da chave:** [512, 1024, 2048, 4096, 8192]

**Chaves:** chave criptográfica do algoritmo RSA gerada automaticamente pelo sistema

**Evento 7: Usuário exporta chaves RSA por cópia**



Dicionário de fluxo de dados:

**Tipo de chave:** [Chave Pública, Chave Privada]

**Chave\_Pública:** Chave criptográfica pública RSA

**Chave\_Privada:** Chave criptográfica privada RSA

**Chave:** Chave criptográfica selecionada pelo usuário

**Evento 8: Usuário importa chaves RSA**

Dicionário de fluxo de dados:

**Chave:** Chave criptográfica do algoritmo RSA

**Evento 9: Usuário criptografa com RSA**

Dicionário de fluxo de dados:

**Chave Atual:** Chave importada pelo usuário

**Mensagem normal:** Mensagem normal digitada pelo usuário

**Mensagem criptografada:** Mensagem criptografada pelo sistema

**Informações do processo:** Modo + Mensagem de entrada + Mensagem de entrada + Data

**Evento 10: Usuário aplica técnica de esteganografia**

Dicionário de fluxo de dados:

**Arquivos**: arquivo original + arquivo para esconder + diretório para salvar aquivo + nome do novo arquivo

**Arquivo esteganografado:** Novo arquivo esteganografado gerado pelo sistema

**Informações do processo:** Nome do arquivo original + Nome do arquivo oculto + Nome do arquivo esteganografado + data

**Evento 11: Enviar Mensagem Cifrada por Email**



Dicionário de fluxo de dados:

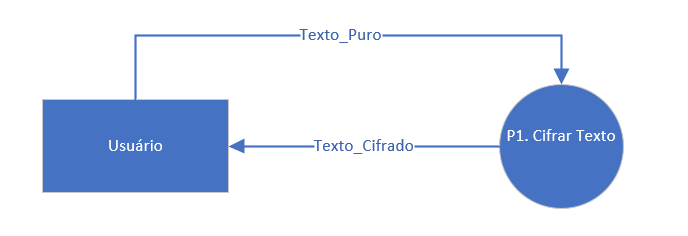
**Mensagem cifrada:** Mensagem de texto cifrada por um dos algoritmos do sistema

**Informações do e-mail:** endereço de e-mail + senha + assunto

**E-mail:** Mensagem de e-mail contendo a mensagem cifrada

**E-mail destinatário:** Endereço de e-mail que receberá a mensagem (e-mail)

**Evento 12: Usuário Cifra em Código Morse**

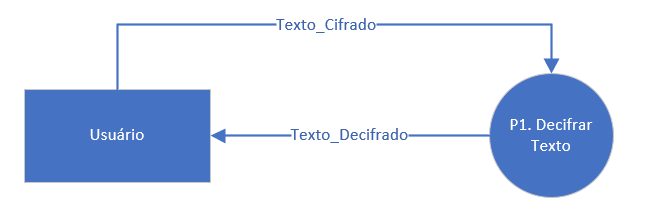


Dicionário de dados:

**Texto Puro:** Texto sem cifragem digitado ou colado pelo usuário. Sistema retorna texto cifrado ao usuário.

**Texto Cifrado:** Texto cifrado pelo usuário.

**Evento 13: Usuário Decifra em Código Morse**



Dicionário de dados:

**Texto Cifrado:** Texto cifrado digitado ou colado pelo usuário. Sistema retorna texto decifrado ao usuário.

**Texto Decifrado:** Texto decifrado pelo usuário.

**Evento 14: Usuário realiza cadastro**



Dicionário de fluxo de dados:

**Informações do cadastro:** Nome + e-mail + nome de usuário + senha + (e-mail\_SMTP, senha\_SMTP, servidor\_SMTP, porta\_SMTP)

**Evento 15: Usuário solicita recuperação de senha**



Dicionário de fluxo de dados:

**E-mail destinatário:** Endereço de e-mail que receberá a mensagem (e-mail)

**E-mail de recuperação:** E-mail contendo as informações do cadastro e a nova senha gerada pelo sistema

**Evento 16: Usuário envia e-mail com anexos (Esteganografia)**



Dicionário de fluxo de dados:

**Anexos do e-mail:** arquivos anexados pelo usuário

**Informações do e-mail:** endereço de e-mail + senha + assunto

**E-mail:** Mensagem de e-mail contendo as informações do e-mail e os anexos

**E-mail destinatário:** Endereço de e-mail que receberá a mensagem (e-mail)

**Evento 17: Usuário decifra com RSA**

Dicionário de fluxo de dados:

**Chave Atual:** Chave importada pelo usuário

**Mensagem cifrada:** Mensagem criptografada digitada pelo usuário

**Mensagem criptografada:** Mensagem decifrada pelo sistema

**Informações do processo:** Modo + Mensagem de entrada + Mensagem de entrada + Data

**Evento 18: Usuário realiza login**



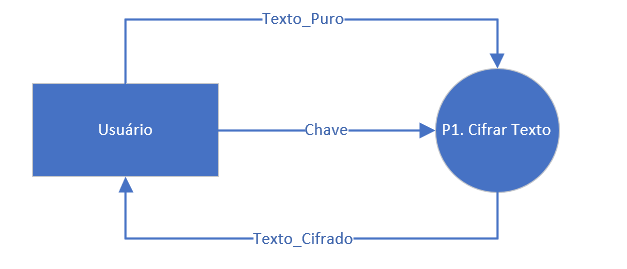
Dicionário de fluxo de dados:

**Dados usuário:** Nome de usuário + senha

**Usuários:** Depósito de dados com todos os usuários

**Menu principal:** Tela do menu principal

**Evento 19: Usuário Cifra em Cifra de Cesar**



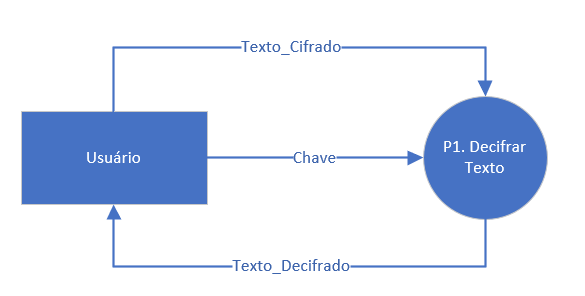
Dicionário de dados:

**Texto Puro:** Texto sem cifragem digitado ou colado pelo usuário. Sistema retorna texto cifrado ao usuário.

**Texto Cifrado:** Texto cifrado pelo usuário.

**Chave:** Texto numérico digitado pelo usuário. Sistema usa para cifrar o texto de entrada.

**Evento 20: Usuário Decifra em Cifra de Cesar**



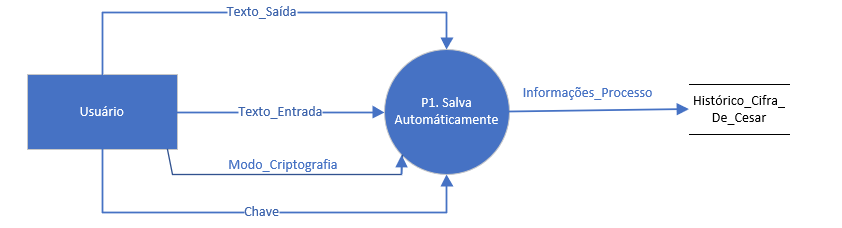
Dicionário de dados:

**Texto Cifrado:** Texto cifrado digitado ou colado pelo usuário. Sistema retorna texto decifrado ao usuário.

**Texto Decifrado:** Texto decifrado pelo usuário.

**Chave:** Texto numérico digitado pelo usuário. Sistema usa para cifrar o texto de entrada.

**Evento 21: Usuário Salva Automaticamente em Cifra de Cesar**



Dicionário de dados:

**Chave:** Texto numérico digitado pelo usuário. Sistema usa para cifrar ou decifrar o texto de entrada.

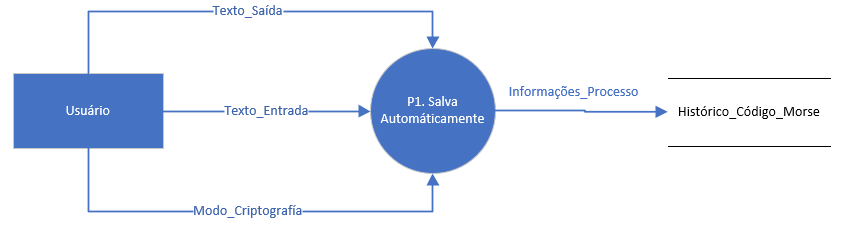
**Texto Entrada:** Texto digitado ou colado pelo usuário.

**Texto Saída:** Texto cifrado ou decifrado pelo usuário.

**Modo Criptografia:** Modo de criptografia selecionado pelo usuário.

**Informações Processo:** Informações que serão armazenadas no banco de dados. Sendo elas: Chave, Texto Entrada, Texto Saída e Modo Criptografia.

**Evento 22: Usuário Salva Automaticamente em Código Morse**



Dicionário de dados:

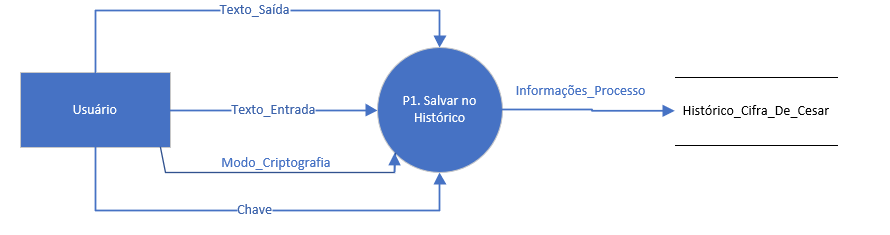
**Texto Entrada:** Texto digitado ou colado pelo usuário.

**Texto Saída:** Texto cifrado ou decifrado pelo usuário.

**Modo Criptografia:** Modo de criptografia selecionado pelo usuário.

**Informações Processo:** Informações que serão armazenadas no banco de dados. Sendo elas: Texto Entrada, Texto Saída e Modo Criptografia.

**Evento 23: Usuário Salva Informações em Cifra de Cesar**



Dicionário de dados:

**Chave:** Texto numérico digitado pelo usuário. Sistema usa para cifrar ou decifrar o texto de entrada.

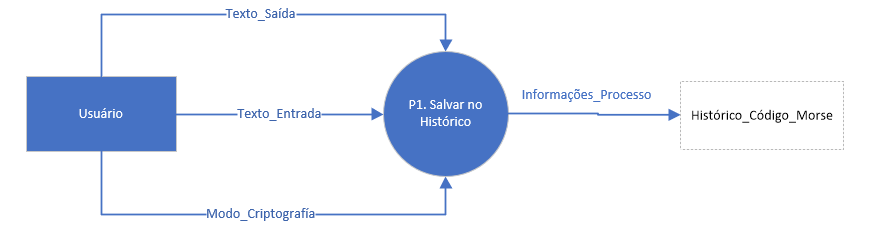
**Texto Entrada:** Texto digitado ou colado pelo usuário.

**Texto Saída:** Texto cifrado ou decifrado pelo usuário.

**Modo Criptografia:** Modo de criptografia selecionado pelo usuário.

**Informações Processo:** Informações que serão armazenadas no banco de dados. Sendo elas: Chave, Texto Entrada, Texto Saída e Modo Criptografia.

**Evento 24: Usuário Salva Informações em Código Morse**



Dicionário de dados:

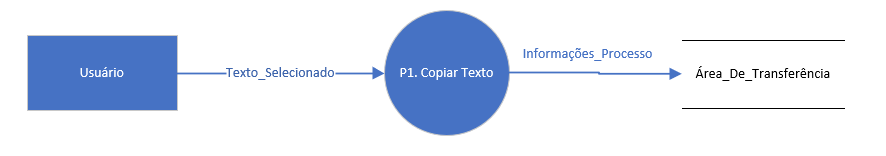
**Texto Entrada:** Texto digitado ou colado pelo usuário.

**Texto Saída:** Texto cifrado ou decifrado pelo usuário.

**Modo Criptografia:** Modo de criptografia selecionado pelo usuário.

**Informações Processo:** Informações que serão armazenadas no banco de dados. Sendo elas: Texto Entrada, Texto Saída e Modo Criptografia.

**Evento 25: Usuário Copia Texto**



Dicionário de dados:

**Texto Selecionado:** Texto cifrado ou decifrado pelo usuário

**Informações Processo:** Informações que serão armazenadas no banco de dados. Sendo ela(s): Texto Selecionado

**Evento 26: Usuário exporta chaves RSA por impressão**



Dicionário de fluxo de dados:

**Tipo de chave:** [Chave Pública, Chave Privada]

**Impressora:** Impressora selecionada pelo usuário

**Chave\_Pública:** Chave criptográfica pública RSA

**Chave\_Privada:** Chave criptográfica privada RSA

**Chave:** Chave criptográfica selecionada pelo usuário

**Evento 27: Usuário exporta chaves RSA por e-mail**



Dicionário de fluxo de dados:

**Tipo de chave:** [Chave Pública, Chave Privada]

**E-mail destinatário:** E-mail de destino

**Chave\_Pública:** Chave criptográfica pública RSA

**Chave\_Privada:** Chave criptográfica privada RSA

**Chave:** Chave criptográfica selecionada pelo usuário

**Evento 28: Usuário exporta chaves RSA por arquivo .txt**



Dicionário de fluxo de dados:

**Tipo de chave:** [Chave Pública, Chave Privada]

**Caminho do arquivo:** Caminho de arquivo selecionado pelo usuário

**Chave\_Pública:** Chave criptográfica pública RSA

**Chave\_Privada:** Chave criptográfica privada RSA

**Chave:** Chave criptográfica selecionada pelo usuário

**Evento 29: Usuário Cola Texto**



Dicionário de fluxo de dados:

**Campo\_Texto\_Selecionado:** Caixa de texto selecionada pelo usuário

**Texto\_Área\_Tranferência:** Texto da área de transferência do usuário

**2.2.4 Especificação de processos**

**Evento 1: Usuário gera chave AES**

Inicio

Obtenha Senha\_Secreta

Obtenha tamanho\_chave

Gera chave

Fim

**Evento 2: Usuário criptografa texto AES**

Inicio

Obtenha Texto\_puro

Criptografar\_texto

Receba Texto\_cifrado

Fim

**Evento 3: Usuário descriptografa texto AES**

Inicio

Obtenha Texto\_cifrado

Descriptografar\_texto

Receba Texto\_original

Fim

**Evento 4: Usuário criptografa arquivo AES**

Inicio

Obtenha Arquivo

Criptografar\_arquivo

Receba Arquivo\_cifrado

Fim

**Evento 5: Usuário descriptografa arquivo AES**

Inicio

Obtenha Arquivo\_cifrado

Decriptografar\_arquivo

Receba Arquivo\_decifrado

Fim

**Evento 6: Usuário gera chave RSA**

Inicio

Obter tamanho da chave

Gerar chave pública e privada

Mostrar chaves para o usuário

Fim

**Evento 7: Usuário exporta chaves RSA por cópia**

Inicio

Obter tipo de chave

Obter chave

Copiar chave para área de transferência

Fim

**Evento 8: Usuário importa chaves RSA**

Inicio

Obter chave

Obter tipo de chave

Guardar chave no depósito chaveAtual

Fim

**Evento 9: Usuário criptografa com RSA**

Inicio

Obter chaveAtual

Obter mensagem normal

Aplicar algoritmo de criptografia RSA

Retornar texto cifrado

Guardar informações do processo no depósito Historico\_RSA

Fim

**Evento 10: Usuário aplica técnica de esteganografia**

Inicio

Obter arquivo original

Obter arquivo a ser escondido

Obter diretório do novo arquivo

Obter nome do novo arquivo

Validar nome do arquivo

Se validar entao

Aplicar técnica de esteganografia

Retornar arquivo esteganografado para o usuário

Senao

Mostrar mensagem de erro

Fimse

Fim

**Evento 11: Enviar Mensagem Cifrada por Email**

Inicio

Obter Texto Saída

Obter Email Destinatário

Obter Infromações Email

Se Infomações Email existem no banco então

Enviar Email para Email Destinatário

Senão

Mensagem de erro

Fimse

Fim

**Evento 12: Usuário Cifra em Código Morse**

Inicio

Obter texto entrada

Cifrar texto entrada

Mostrar texto saída

Fim

**Evento 13: Usuário Decifra em Código Morse**

Inicio

Obter texto entrada

Decifrar texto entrada

Mostrar texto saída

Fim

**Evento 14: Usuário realiza cadastro**

Inicio

Obter dados do usuário

Gravar dados do usuário no banco de dados

Fim

**Evento 15: Usuário solicita recuperação de senha**

Inicio

Obter e-mail destinatário

Gerar senha aleatória

Gravar nova senha no banco de dados com hash

Enviar e-mail de recuperação para e-mail destinatário

Fim

**Evento 16: Usuário envia e-mail com anexos (Esteganografia)**

Inicio

Obter anexos

Obter informações do e-mail

Obter e-mail destinatário

Enviar e-mail com anexos para e-mail destinatário

Fimse

**Evento 17: Usuário decifra com RSA**

Inicio

Obter chaveAtual

Obter mensagem cifrada

Decifrar mensagem com RSA

Retornar texto decifrado

Fim

**Evento 18: Usuário realiza login**

Inicio

Obter dados do usuário

Verificar se dados correspondem

Se dados correspondem

Ir para o Menu principal

Senão

Retornar mensagem de erro

Fimse

Fim

**Evento 19: Usuário Cifra em Cifra de Cesar**

Inicio

Obter texto entrada

Obter Chave

Cifrar texto entrada

Mostrar texto saída

Fim

**Evento 20: Usuário Decifra em Cifra de Cesar**

Inicio

Obter texto entrada

Obter Chave

Decifrar texto entrada

Mostrar texto saída

Fim

**Evento 21: Usuário Salva Automaticamente em Cifra de Cesar**

Inicio

Obter texto entrada

Obter Chave

Obter texto saída

Obter modo criptografia

Armazenar informações processo no histórico Cifra de Cesar

Fim

**Evento 22: Usuário Salva Automaticamente em Código Morse**

Inicio

Obter texto entrada

Obter texto saída

Obter modo criptografia

Armazenar informações processo no histórico Código Morse

Fim

**Evento 23: Usuário Salva Informações em Cifra de Cesar**

Inicio

Obter texto entrada

Obter chave

Obter texto saída

Obter modo criptografia

Armazenar informações processo no histórico Cifra de Cesar

Fim

**Evento 24: Usuário Salva Informações em Código Morse**

Inicio

Obter texto entrada

Obter texto saída

Obter modo criptografia

Armazenar informações processo no histórico Código Morse

Fim

**Evento 25: Usuário Copia Texto**

Inicio

Obter texto selecionado

Armazenar texto selecionado na Área de transferência do usuário

Fim

**Evento 26: Usuário exporta chaves RSA por impressão**

Inicio

Obter tipo de chave

Obter chave

Obter impressora

Construir folha de impressão com a chave

Imprimir folha com a chave

Fim

**Evento 27: Usuário exporta chaves RSA por e-mail**

Inicio

Obter tipo de chave

Obter chave

Construir corpo do e-mail com a chave

Obter dados e informações do e-mail

Enviar e-mail

Fim

**Evento 28: Usuário exporta chaves RSA por arquivo .txt**

Inicio

Obter tipo de chave

Obter chave

Obter caminho do arquivo

Construir arquivo com a chave

Salvar arquivo no caminho

Fim

**Evento 29: Usuário Cola Texto**

Inicio

Obter texto da área de transferência do usuário

Colar texto no campo de entrada

Fim

# 3. Engenharia de Software

a) A engenharia de software é uma área de estudo voltada para a manutenção, desenvolvimento e a criação de alta qualidade de um artefato. “A engenharia é a área em que os conhecimentos científicos e técnicos e a experiência prática são aplicados para exploração dos recursos naturais, para o projeto, construção, exploração e operação de objetos úteis para a humanidade.” (FRANCO, 1).

b) Através do modelo genérico de cinco passos (Comunicação, Planejamento, Modelagem, Construção e Entrega) o projeto foi realizado focando principalmente nas partes de Construção, Comunicação e Planejamento, onde toda a equipe realizou certas partes funcionais e não funcionais, discutindo ideias já propostas e algumas funcionalidades a mais ao decorrer do desenvolvimento, de maneira simultânea. Todos os outros processos desse modelo genérico exceto a “Construção” foram feitos em conjunto, em que todo o grupo realizava uma discussão sobre de que forma seria feito, como estava o andamento do projeto, como seria implementado, entre outros aspectos.

c) O modelo de processo utilizado para a construção do projeto foi semelhante modelo RAD, possuindo certas características do modelo, como a divisão do grupo para cada funcionalidade do projeto e a construção serem em curto prazo, uma vez que é para ser entregue entre trinta a noventa dias. O modelo não é totalmente RAD pelas seguintes características: O grupo seguiu certa ideia de “ciclo”, uma vez que enquanto a construção do software estava em andamento novos debates e novas funcionalidades (funcionais e não funcionais) foram surgindo e sendo implementados constantemente. As vantagens em utilizar essa “mistura” de modelos foi a alta produtividade das funcionalidades, a possibilidade de todos os integrantes participarem e colocarem as habilidades de programação em ação e a possibilidade de todos integrantes do grupo ficarem sempre ligados às mudanças, adição e remoção dos requisitos. A única desvantagem de se trabalhar no “modelo” que o grupo seguiu, é a falta de um padrão conhecido, assim deixando o projeto com um modelo “bagunçado”, não padronizado e pouco utilizado por empresas.

# 4. Projeto

## 4.1 Projeto de interface



Figura 1 – Tela inicial de Login

Esta é a primeira tela do sistema, em que o usuário deve utilizar suas credenciais para fazer login. Também é possível acessar a tela de cadastro ou a tela de recuperação de senha.

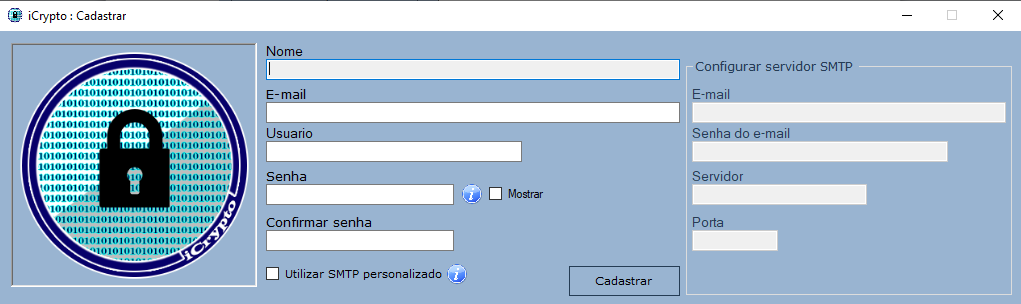


Figura 2 – Cadastro de usuário

Nesta tela o usuário deve informar suas informações para que o sistema cadastre-o. Também é possível cadastrar um servidor SMTP personalizado para que seja utilizado para envio de e-mails.

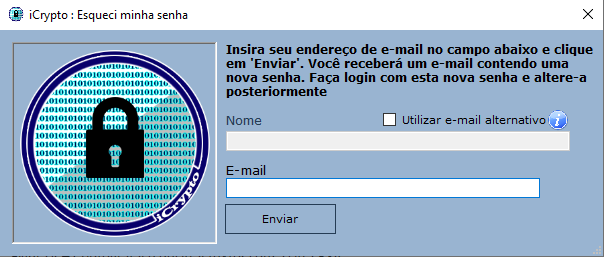


Figura 3 – Função “Esqueci minha senha”

Quando o usuário não se recorda de sua senha, o sistema, por meio desta tela, envia ao usuário uma nova senha gerada automaticamente para que seja possível fazer para o usuário login novamente e utilizar o software.

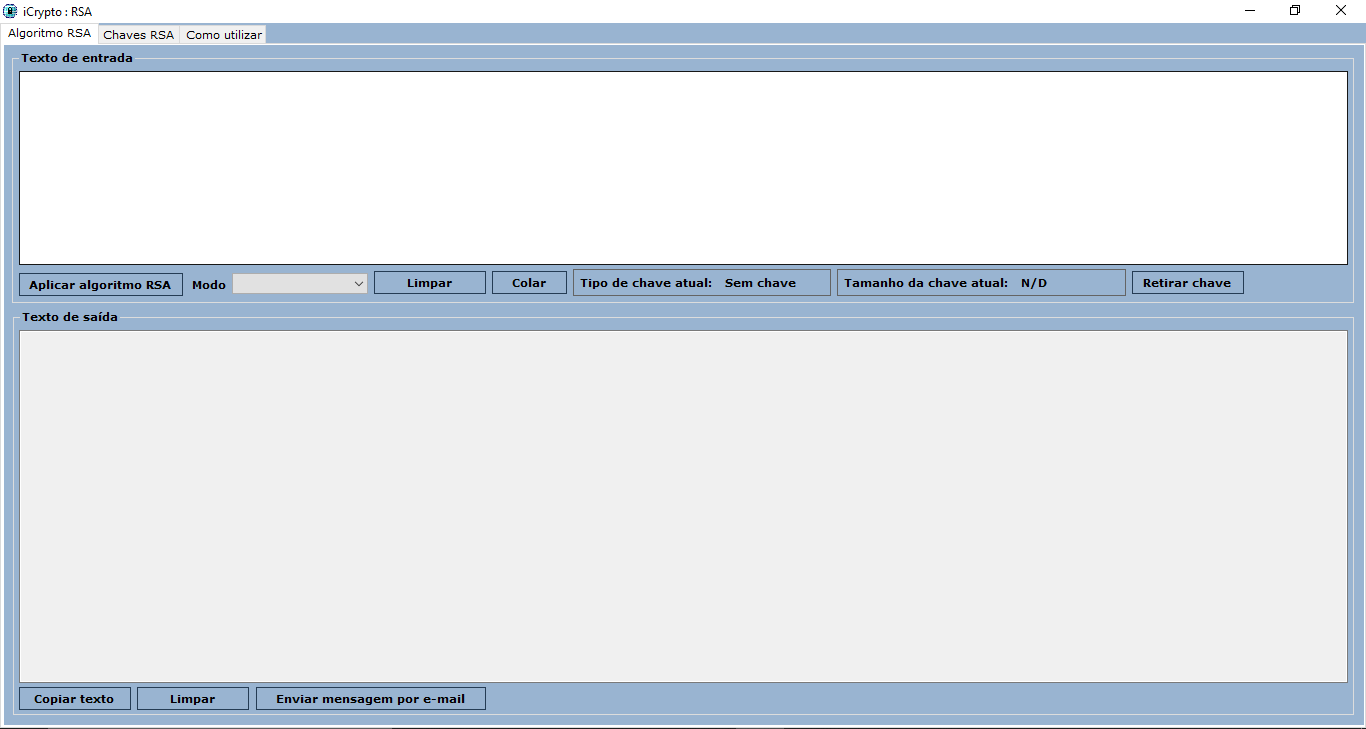


Figura 4 – Criptografia com o algoritmo RSA

O usuário deve informar um texto de entrada (que pode já estar cifrado ou não), definir o modo (Criptografar ou descriptografar) e aplicar o algoritmo RSA na mensagem, que retorna ou o texto decifrado ou cifrado, dependendo do modo selecionado pelo usuário.

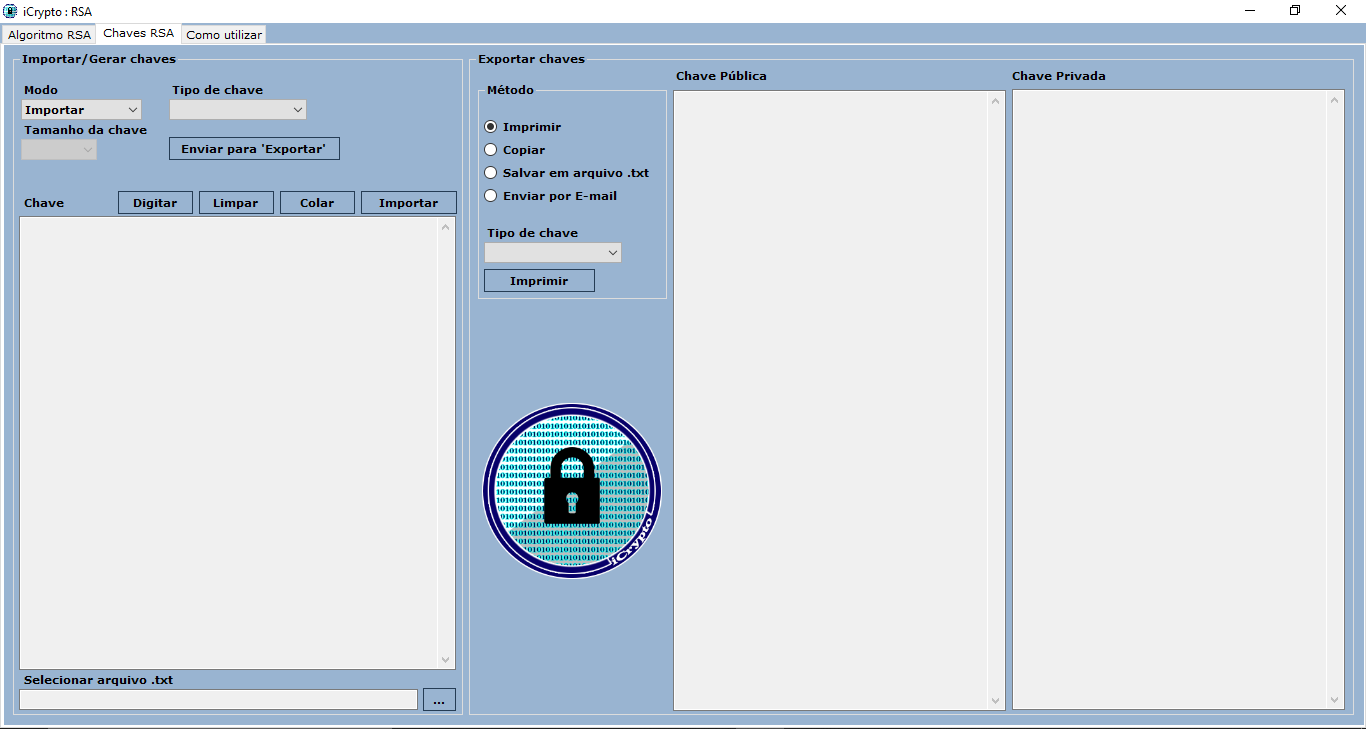


Figura 5 – Gerar, importar e exportar chaves criptográficas do RSA

Nesta tela o usuário pode gerar chaves da criptografia RSA de 5 tamanhos diferentes (512, 1024, 2048, 4096 e 8192 bytes) e salvá-las de 4 métodos diferentes (copiar, imprimir, enviar por e-mail ou salvar como arquivo .txt). Também é possível importar chaves que já estão salvas para que sejam utilizadas na tela de criptografia

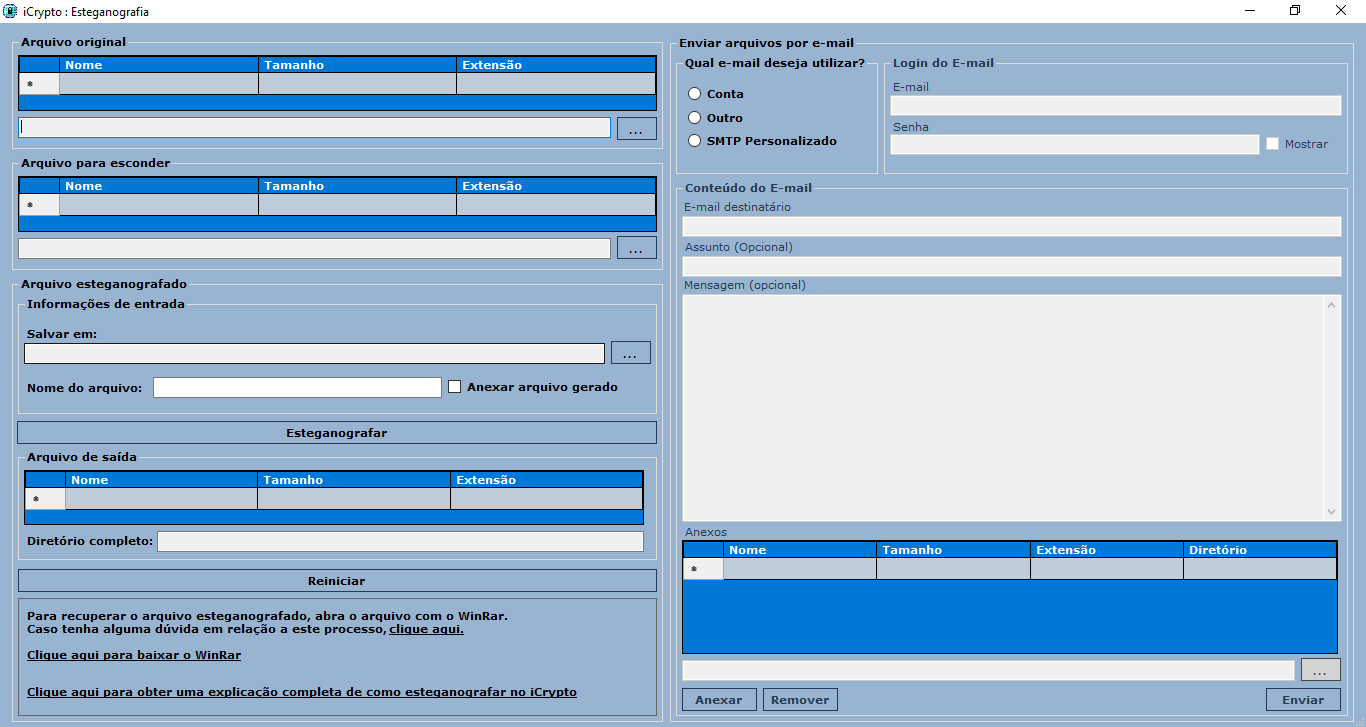


Figura 6 – Ocultando arquivos com a esteganografia

O usuário deve informar um arquivo original, um arquivo para ser ocultado dentro do arquivo original, um nome para o arquivo final gerado e um diretório para este arquivo final. O sistema retorna para o usuário um novo arquivo esteganografado. Também é possível enviar arquivos anexados por e-mail diretamente pelo sistema nesta tela.



Figura 7 – Menu principal

Esta é a tela principal do sistema. A partir dela, o usuário pode selecionar qual tipo de criptografia deseja utilizar, além de ter a possibilidade de alterar as informações do seu cadastro, bem como seu servidor SMTP. Por fim, também é possível limpar os dados do histórico.

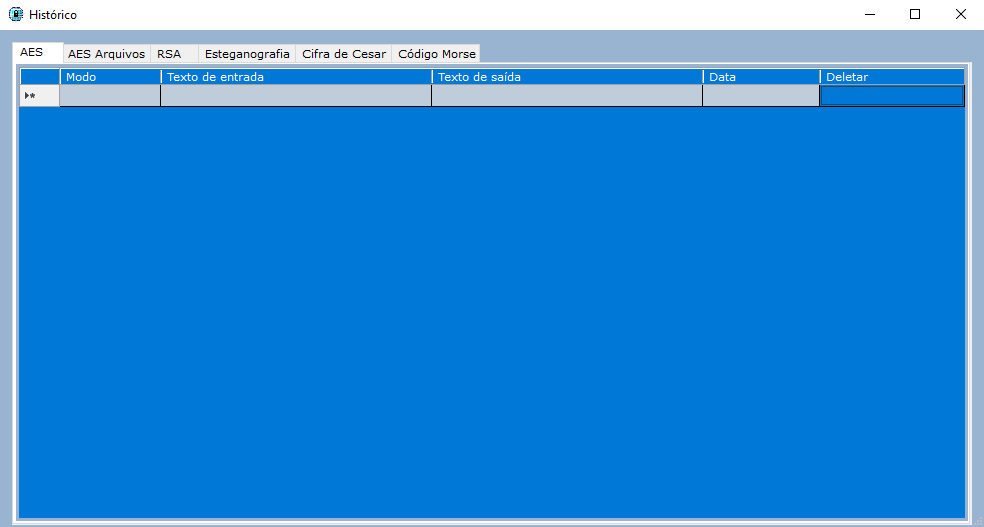


Figura 8 – Histórico

Aqui são apresentadas todas as atividades do usuário, cifra por cifra. O usuário também pode apagar uma linha específica do histórico.

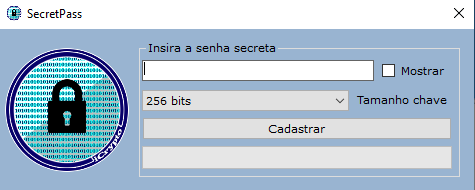


Figura 9 – Senha secreta do AES

O usuário deve digitar uma senha no campo “Insira a senha secreta” para que possa ser utilizada como chave na criptografia AES. Também há a possibilidade de escolher o tamanho da chave que será utilizada, que pode ser de 256 ou 128 bits.

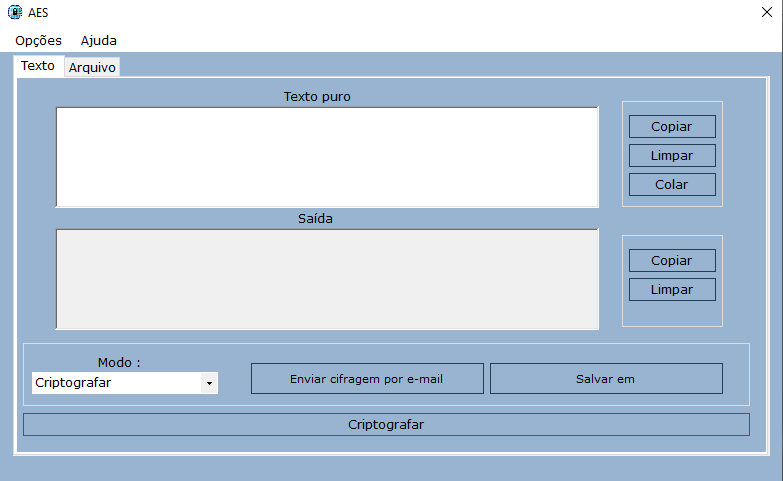


Figura 10 – Criptografia de texto com algoritmo AES

Nesta tela, o usuário utilizará a criptografia AES para cifrar e decifrar seus textos inseridos, além de enviá-los por e-mail.

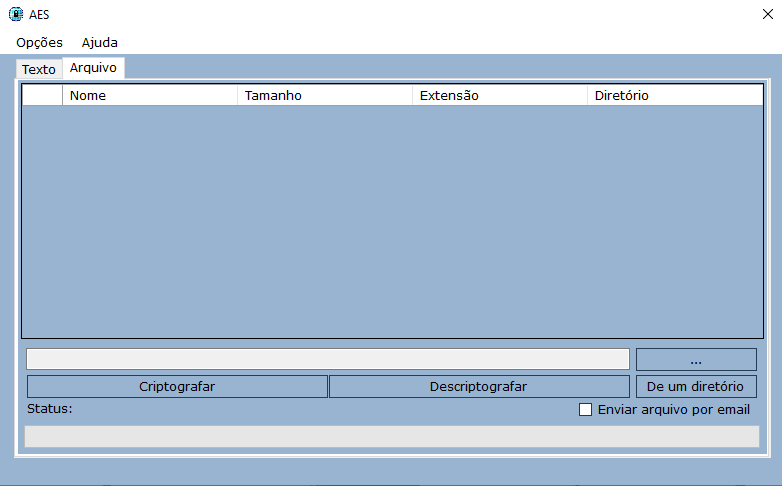


Figura 11 – Criptografia de arquivos com algoritmo AES

Nesta tela, o usuário utilizará a criptografia AES para cifrar e decifrar seus arquivos, além de enviá-los por e-mail.

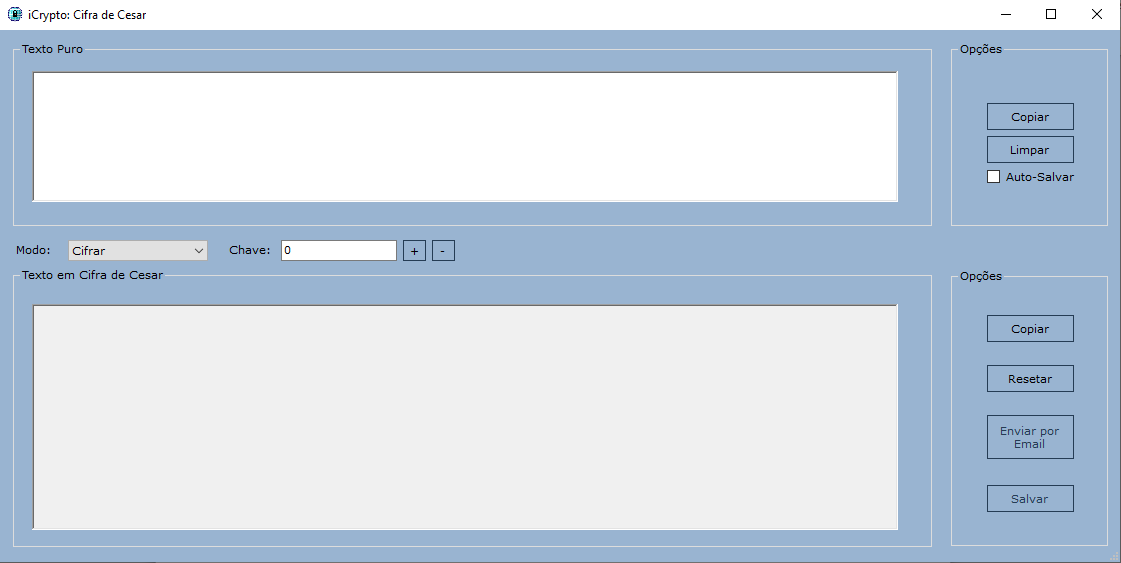


Figura 12 – Criptografia de textos com Cifra de Cesar

Nesta tela, o usuário utilizará a Cifra de Cesar para cifrar e decifrar seus textos inseridos, além de enviá-los por e-mail.

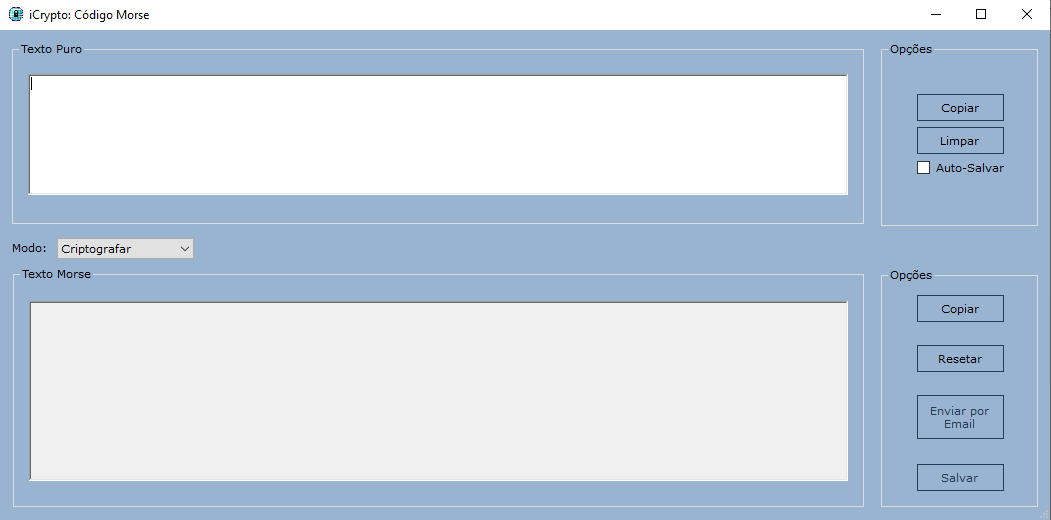


Figura 13 – Criptografia de textos com Código Morse

Nesta tela, o usuário utilizará o Código Morse para cifrar e decifrar seus textos inseridos, além de enviá-los por e-mail.

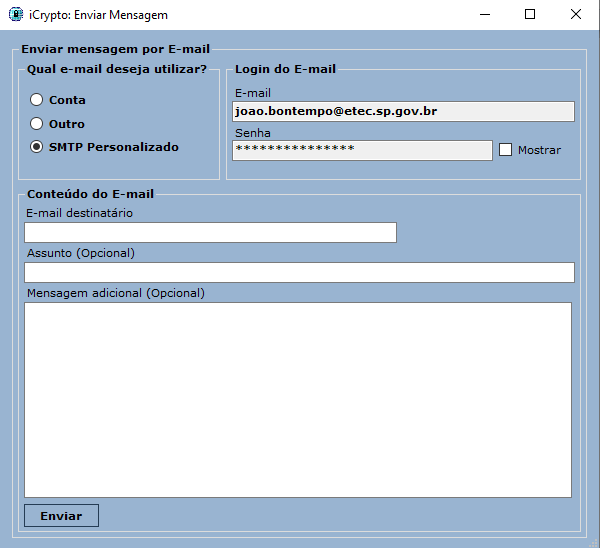


Figura 14 – Enviar mensagens cifradas por e-mail

Esta tela é utilizada para enviar as mensagens de saída das criptografias de texto por E-mail. O usuário deve especificar qual o e-mail e senha que serão utilizados, bem como o e-mail destinatário.

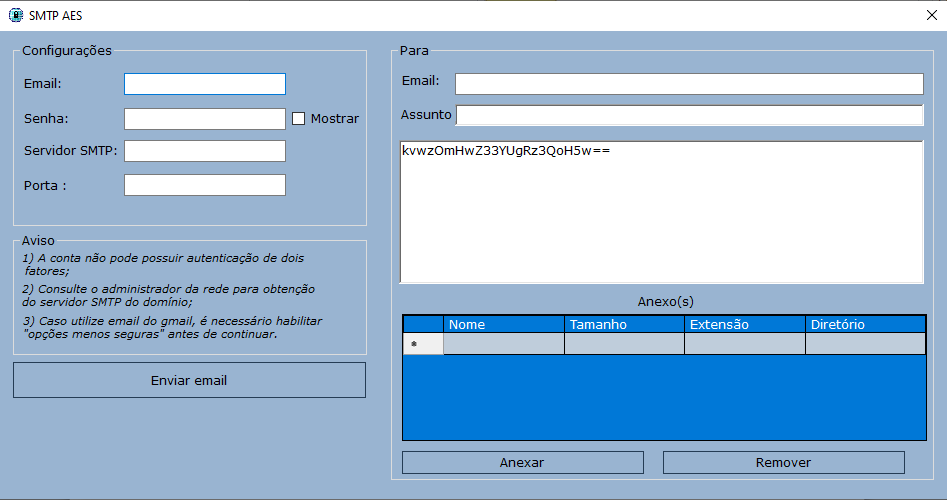


Figura 15 – Enviar mensagem e arquivo cifrados por e-mail com AES

Esta tela é utilizada para enviar as mensagens de saída das criptografias de texto da criptografia AES por E-mail. O usuário deve especificar qual o e-mail e senha que serão utilizados, bem como o e-mail destinatário como campos obrigatórios. Também é possível enviar arquivos em anexo.

## 4.2 Ambiente do sistema

### 4.2.1 Ambiente de desenvolvimento

O sistema foi desenvolvido em Windows 10 Pro na arquitetura 64 bits, através do Microsoft Visual Studio Community 2019 16.4.5 contendo Microsoft .NET Framework Versão 4.8.03761.

## 4.2.2 Ambiente de execução

O sistema deverá ser executado em máquinas que possuem o framework .NET instalado, bem como o sistema operacional Windows em suas variadas versões. De preferência, o computador deve estar conectado à internet, para que seja possível enviar e-mails diretamente pelo sistema. Entretanto, as principais funcionalidades não dependem de uma conexão com a internet.

**REFERÊNCIAS**

MICHEL, Franco. O que é Engenharia de Software?. Acessado em 20 mai 2020. Disponível em:<<http://www.forumdaconstrucao.com.br/conteudo.php?a=0&>Cod=1573>.