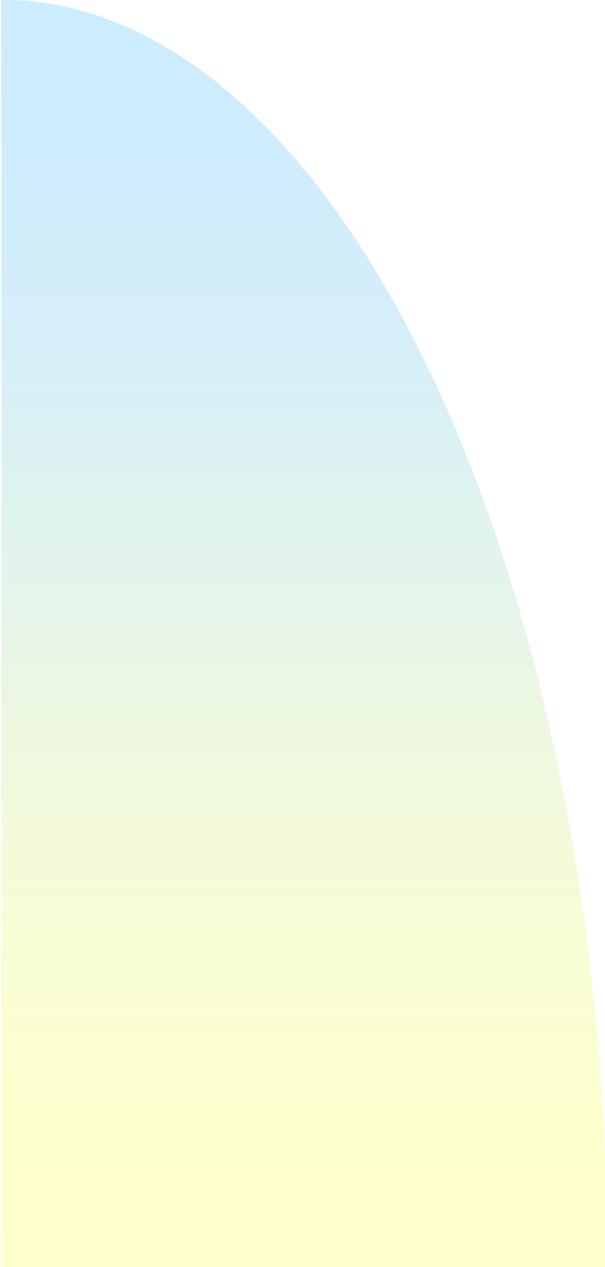


PROTÓTIPO DE SOFTWARE PARA NOTIFICAÇÃO DE RECEBIMENTO DE NOVOS E- MAIL'S, BASEADO NO PROTOCOLO IMAP

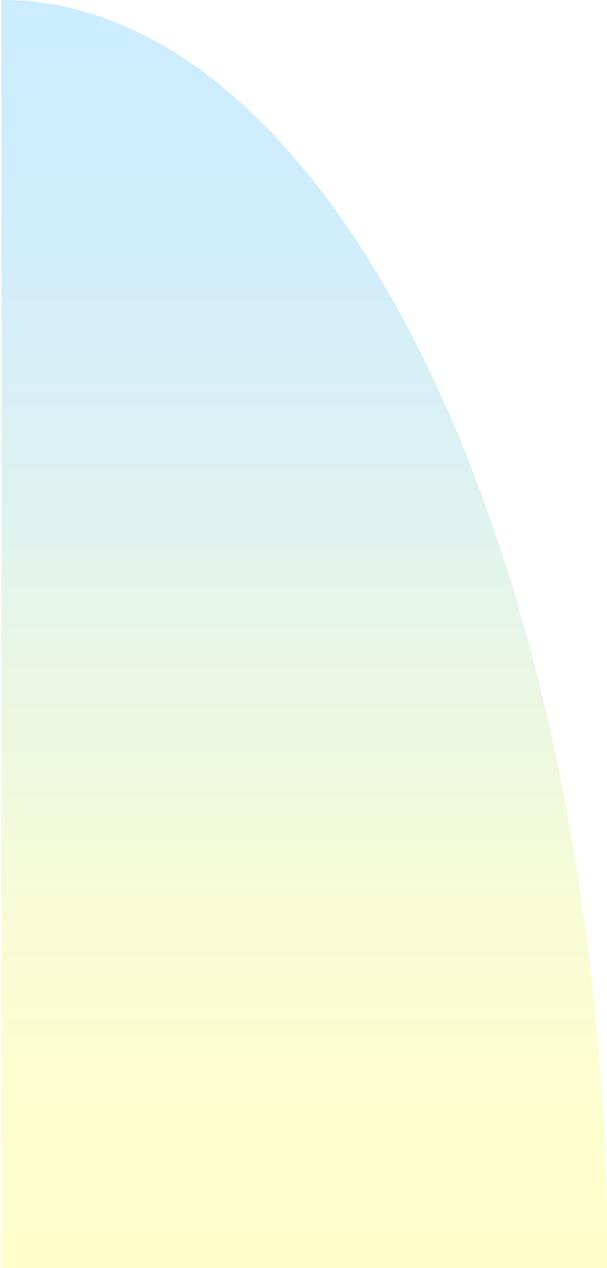
Aluno: **Alexandre Eduardo Knorst**

Orientador: **Sérgio Stringari**



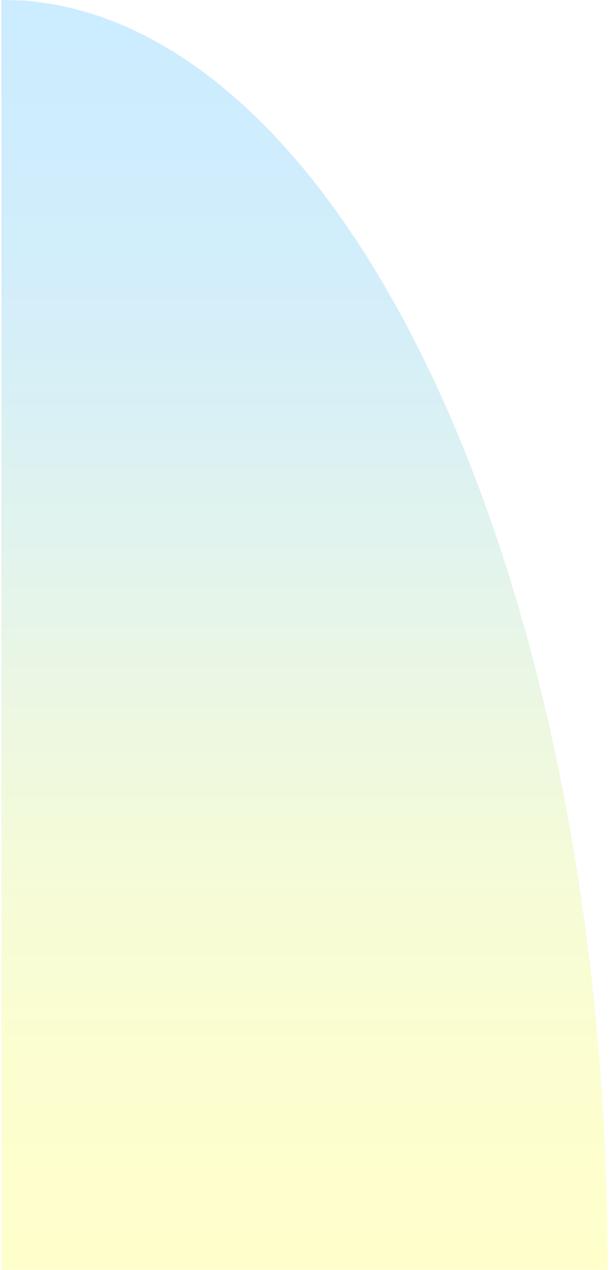
Roteiro

- Introdução
- Fundamentação Teórica
 - ◆ Protocolos de Correio Eletrônico
 - ◆ **IMAP**
- Tecnologias utilizadas
 - ◆ TCP/IP
 - ◆ Socket



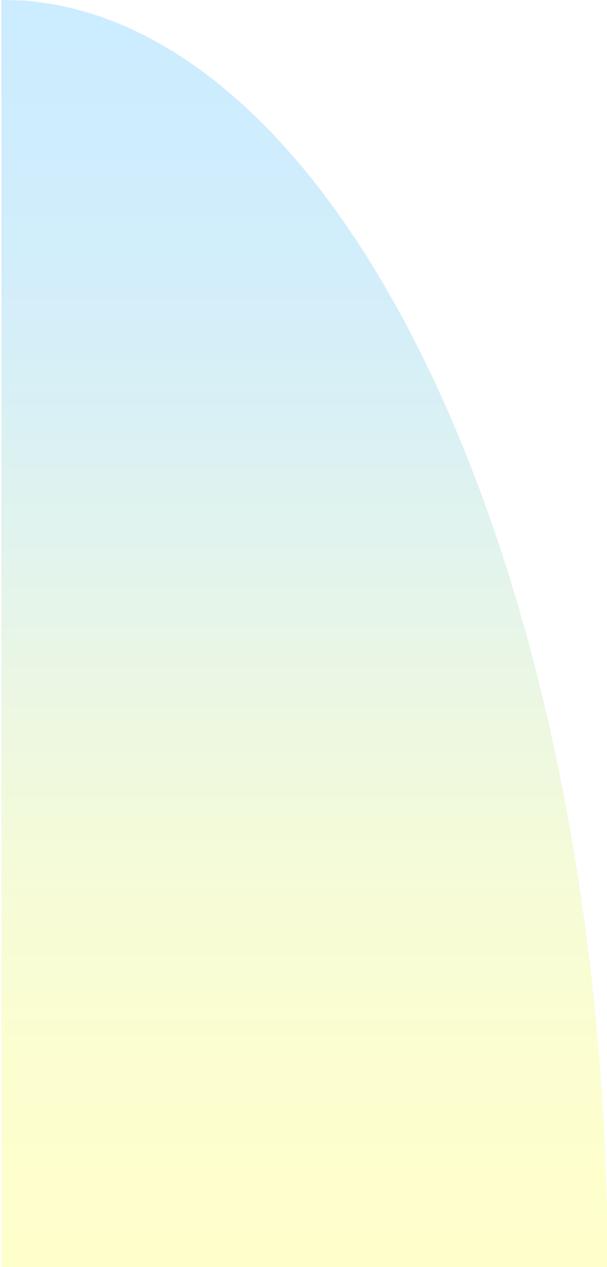
Roteiro

- Desenvolvimento do Trabalho
 - ◆ Especificação
 - ◆ Implementação
 - ◆ Funcionalidade do protótipo
- Conclusões
- Referência Bibliográfica



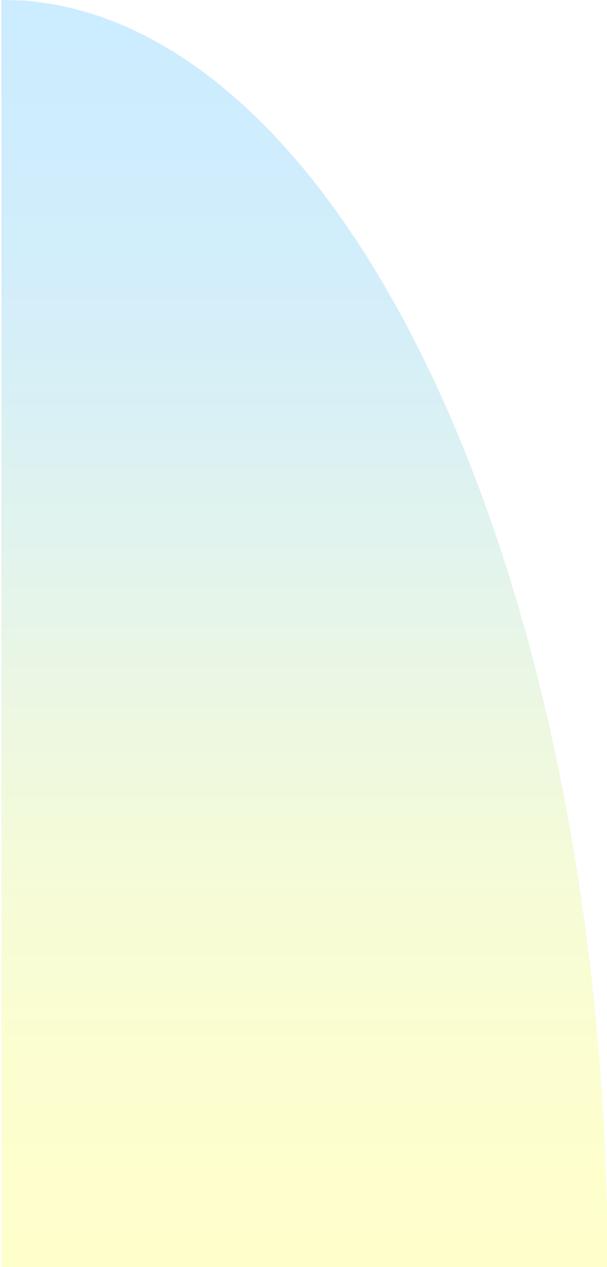
Introdução

- Origem da Internet
- Evolução da Internet
- Origem do trabalho



Fundamentação Teórica

- Protocolos de Correio Eletrônico
 - ◆ SMTP (RFC 821)
Simple Mail Transport Protocol
 - ◆ POP (RFC 1939)
Post Office Protocol
 - ◆ **IMAP (RFC 2060)**
Internet Message Access Protocol



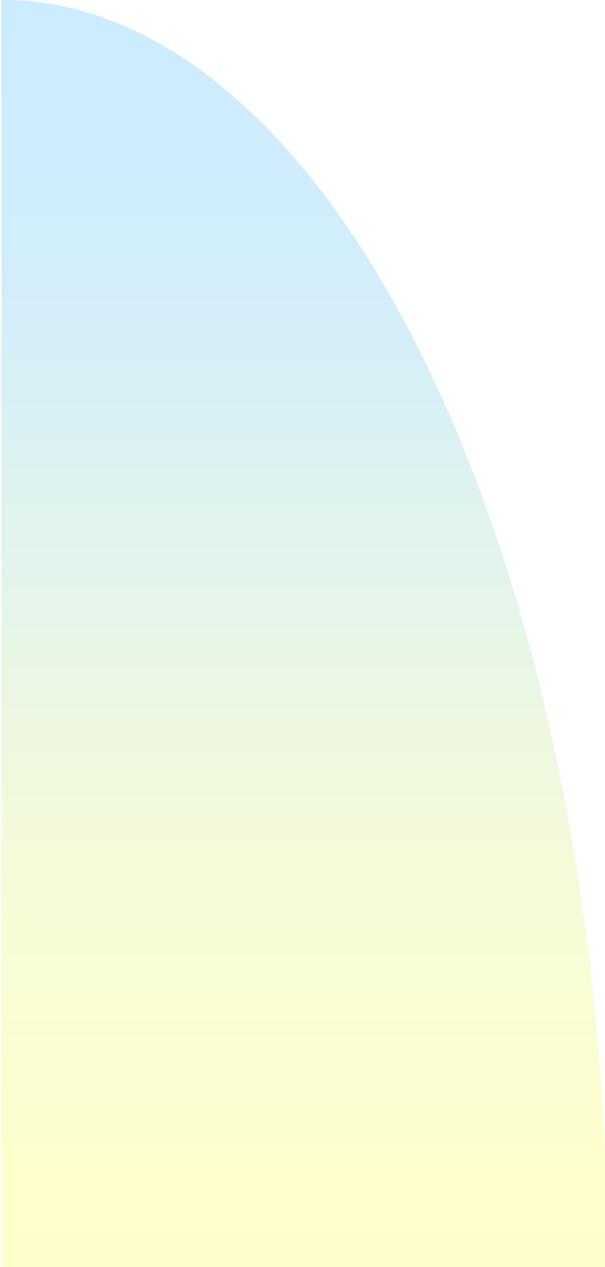
Fundamentação Teórica

- IMAP

- ◆ Características

- ◆ Estados

- ☞ Non-Authenticated
- ☞ Authenticated
- ☞ Selected
- ☞ Logout



Fundamentação Teórica

■ IMAP

◆ Comandos

- ☞ Qualquer estado: CAPABILITY, NOOP e **LOGOUT**;
- ☞ Non-Authenticated: **LOGIN** e AUTHENTICATE;
- ☞ Authenticated: SELECT, **EXAMINE**, CREATE, DELETE, RENAME, LIST, STATUS e APPEND;
- ☞ Selected: CHECK, CLOSE, EXPUNGE, **SEARCH**, **FETCH**, STORE, COPY e UID.

Fundamentação Teórica

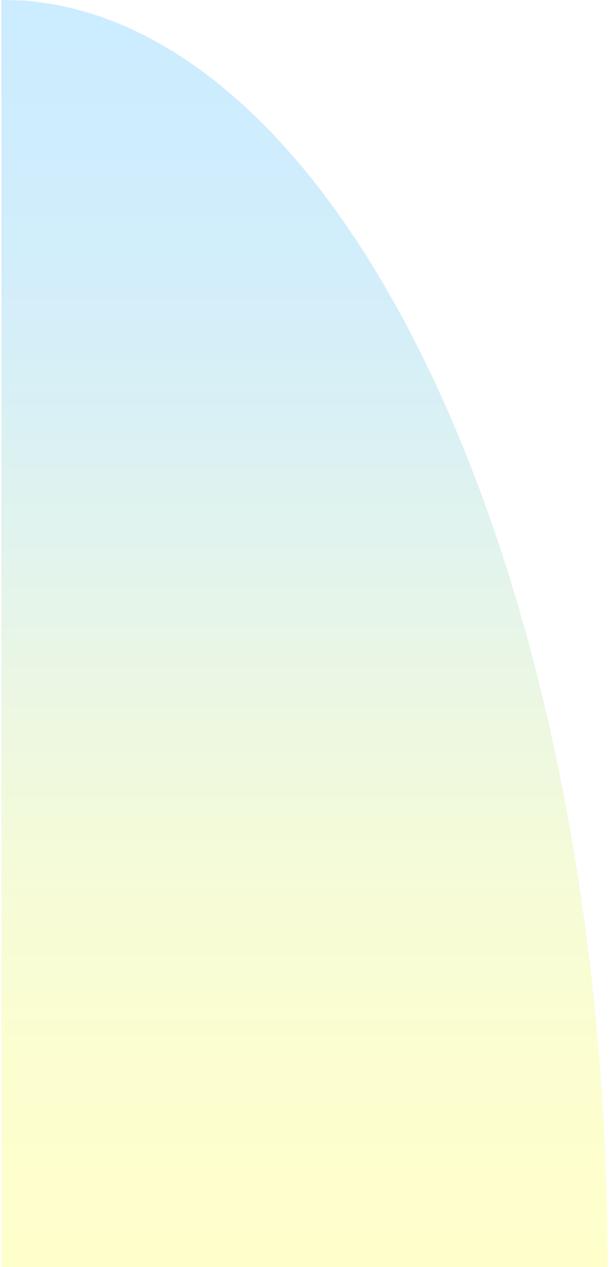
■ IMAP

◆ Comandos

☞ Itens de um comando

- **Rótulo;**
- **Comando;**
- **Parâmetros.**

Ex.: *Aek0001 LOGIN knorst minha_senha*



Fundamentação Teórica

- IMAP

- ◆ Respostas do Servidor

- 👉 OK

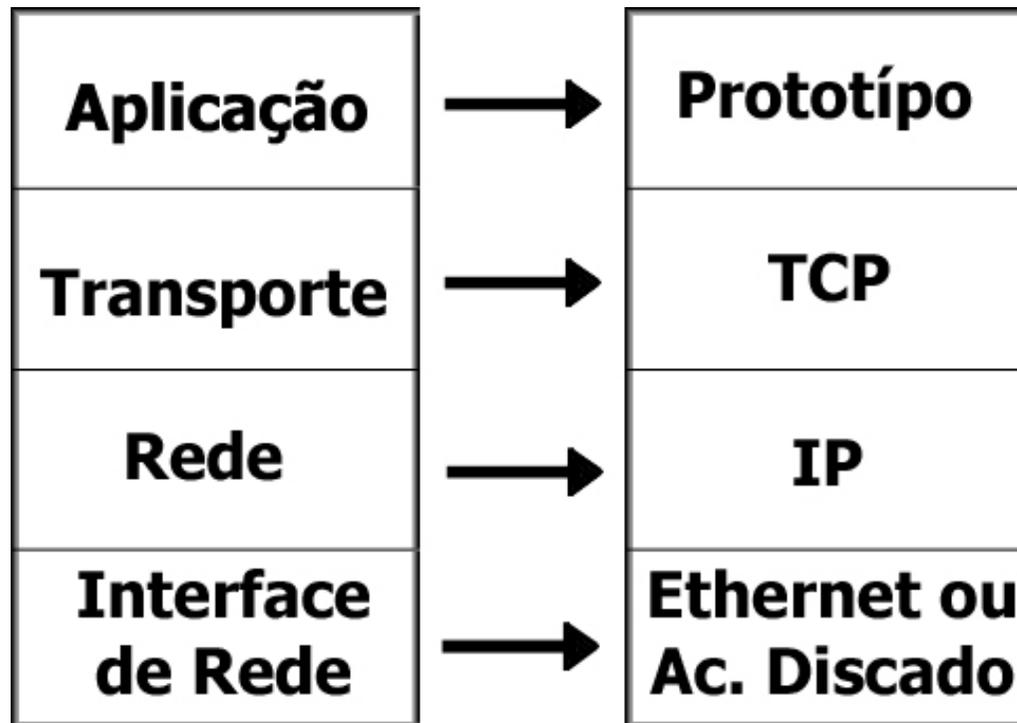
- 👉 NO

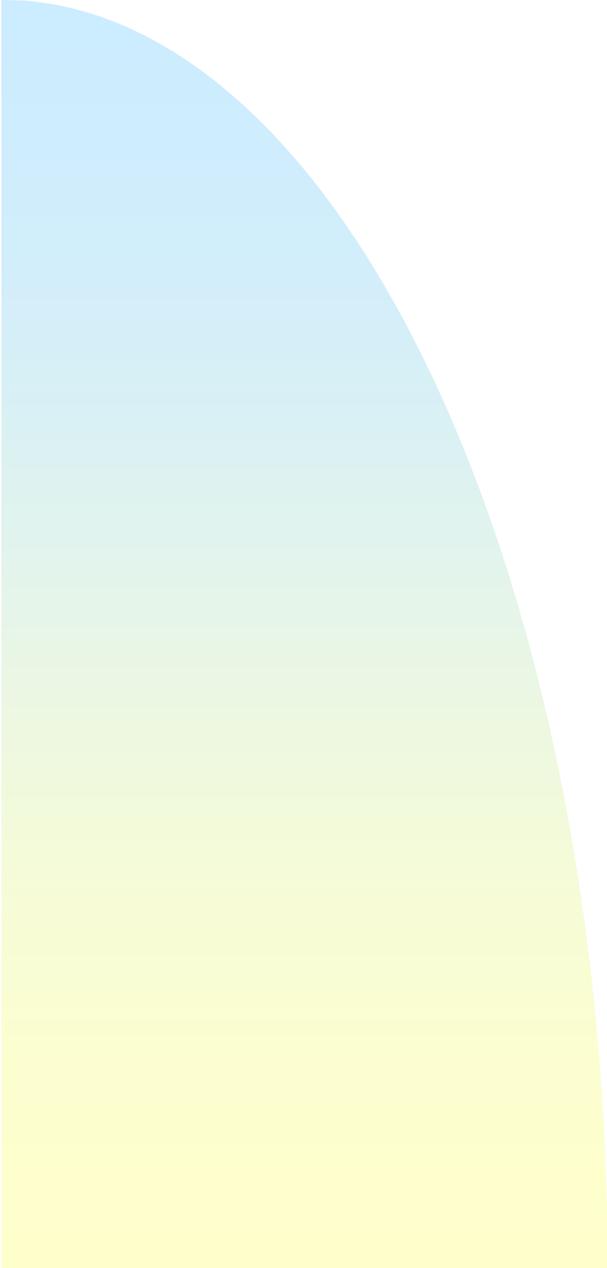
- 👉 BAD

- 👉 Sem Rótulo

Tecnologias utilizadas

- Protocolo de rede TCP/IP



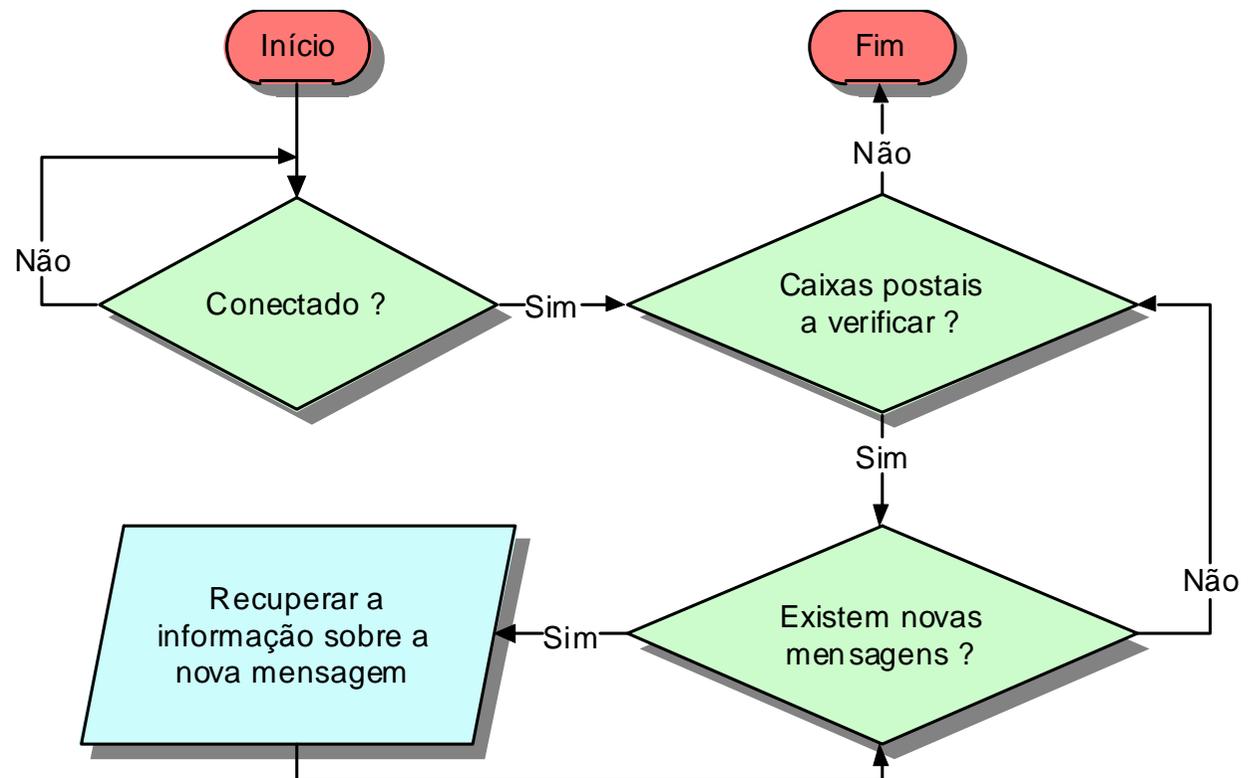


Tecnologias utilizadas

- API Socket
 - ◆ socket
 - ◆ setsockopt
 - ◆ connect
 - ◆ send
 - ◆ recv
 - ◆ close

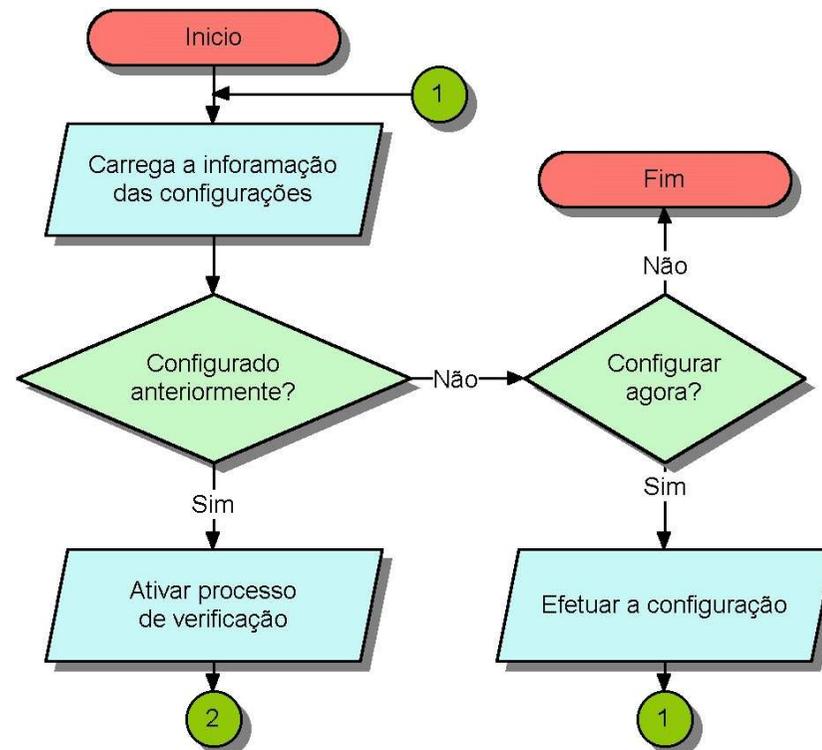
Desenvolvimento

- Especificação



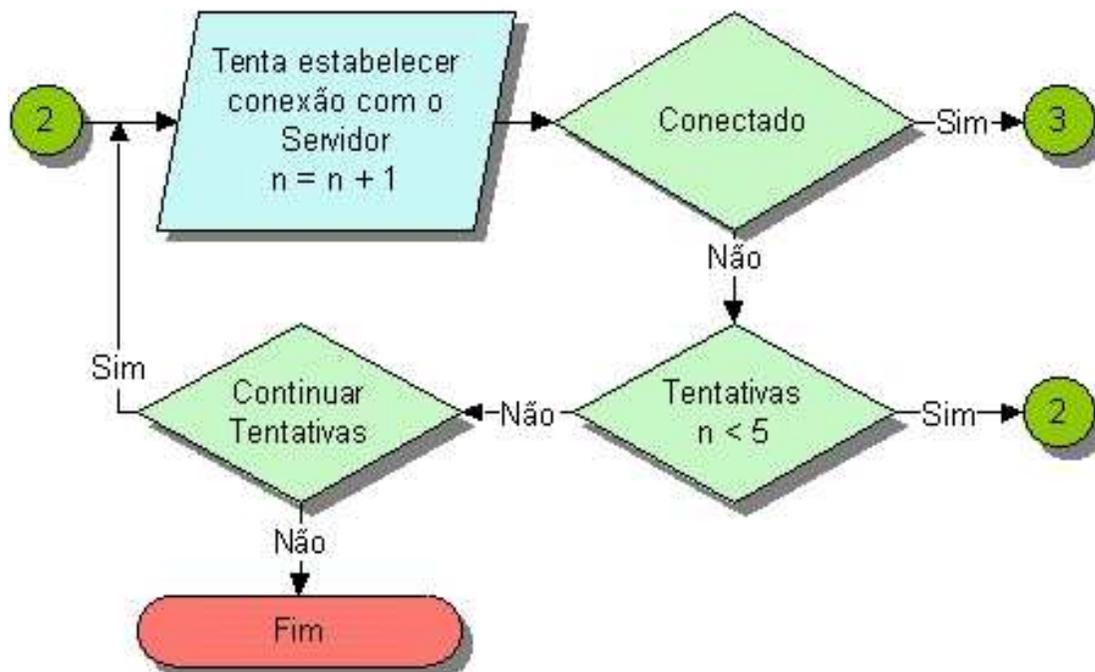
Desenvolvimento

- Especificação
 - ◆ Carga das configurações



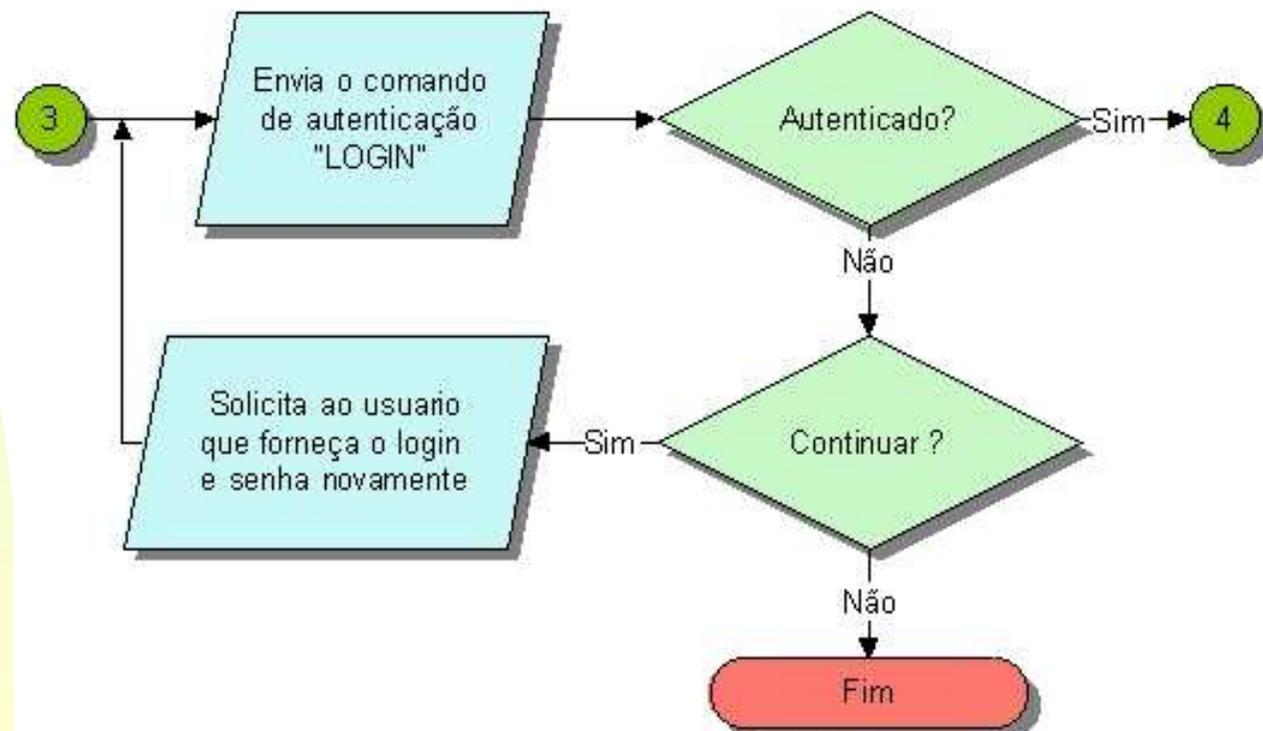
Desenvolvimento

- Especificação
 - ◆ Estabelece a conexão com o servidor



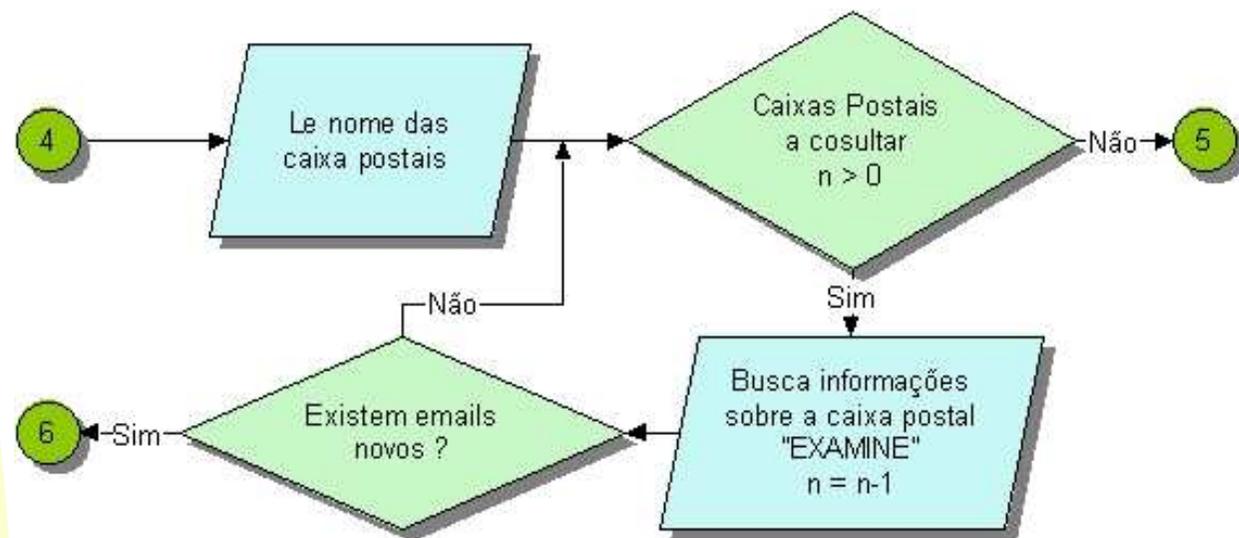
Desenvolvimento

- Especificação
 - ◆ Autenticação



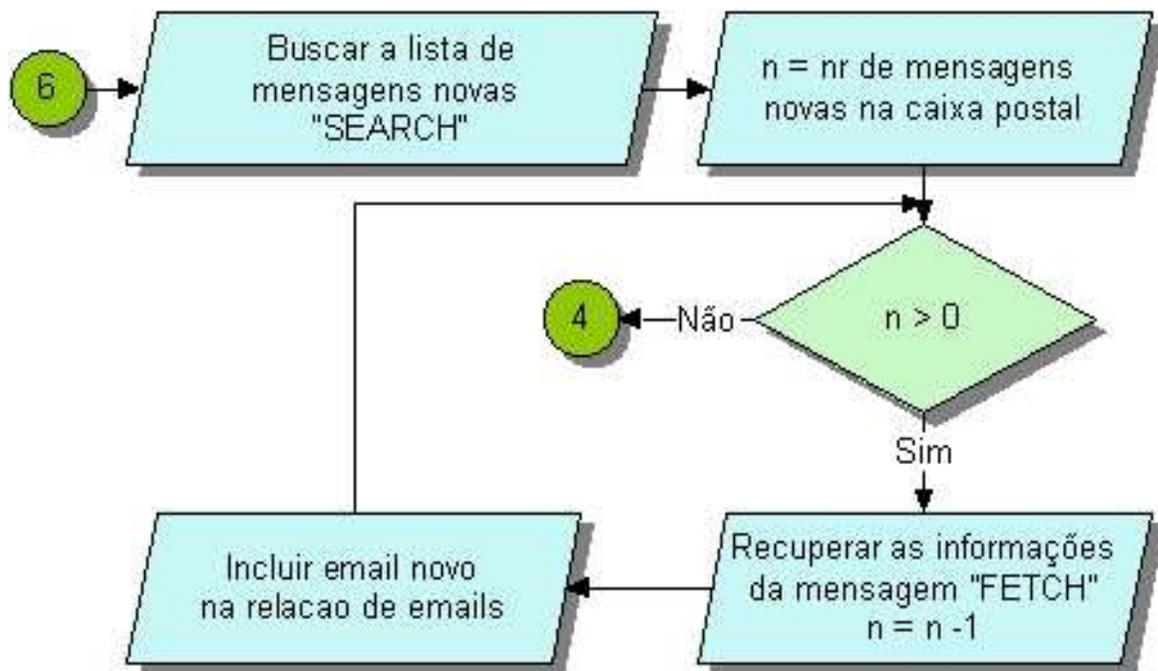
Desenvolvimento

- Especificação
 - ◆ Verificação por novas mensagens



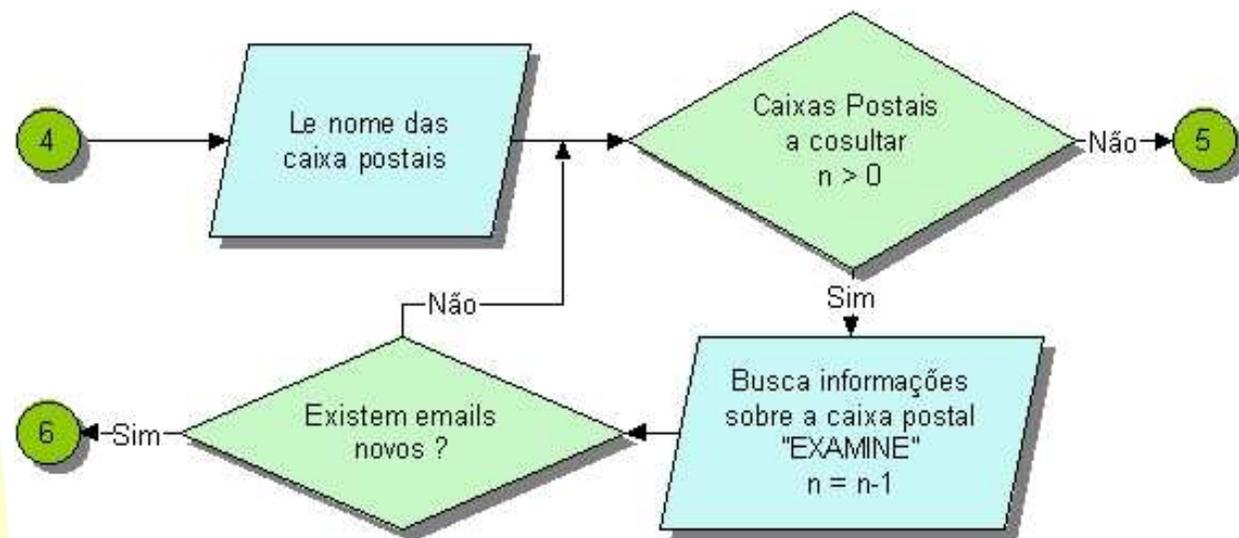
Desenvolvimento

- Especificação
 - ◆ Quando houverem novas mensagens



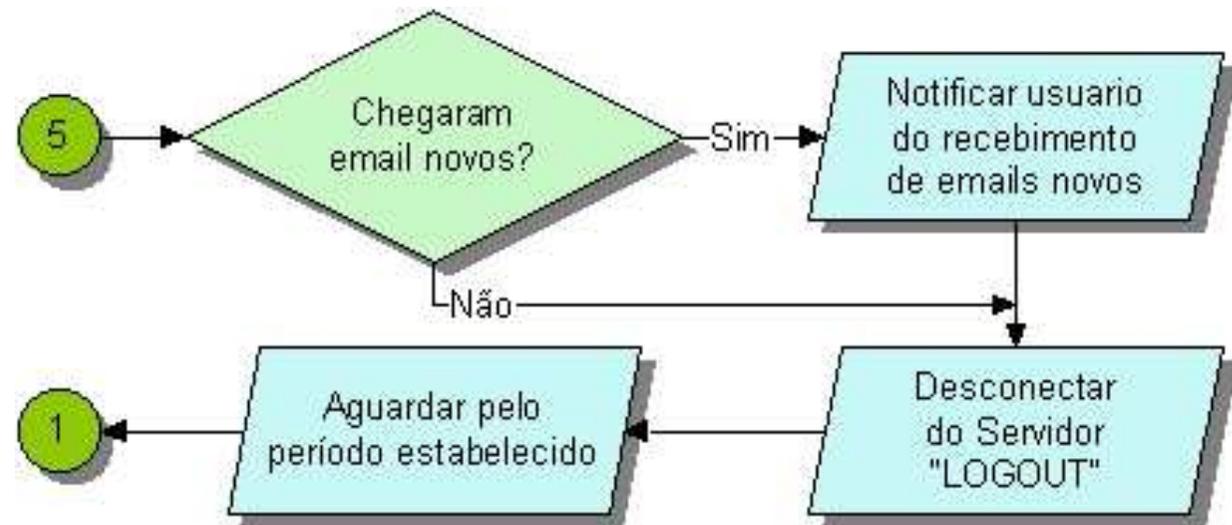
Desenvolvimento

- Especificação
 - ◆ Verificação por novas mensagens



Desenvolvimento

- Especificação
 - ◆ Processo executado após a verificação



Desenvolvimento

- Implementação

- ◆ Conecta - Estabelece a conexão;

```
...
FDSocket := socket(AF_INET, SOCK_STREAM, IPPROTO_IP);
...
SetSockOpt(FDSocket, SOL_SOCKET, SO_RCVBUF,
  PChar(IntToStr(MAXBUFF)), Length(IntToStr(MAXBUFF)));
...
if Connect(FDSocket, SockAddrIn, Sizeof(SockAddrIn)) <> 0 then
...

```

Desenvolvimento

- Implementação
 - ◆ Envia_Comando - Envia um comando ao Servidor;

```
...
Comando := QualComando;
Aux_comando := NovoRotulo+' '+QualComando+' '+Parametros;
FormDEBUG.DEBUG('C: '+Aux_comando);
Aux_comando := Aux_comando+#10+#13;
Send(FDSocket, Pchar(Aux_Comando)^, Length(Aux_comando), 0);
Result := WSAGetLastError;
...
```

Desenvolvimento

- Implementação

- ◆ NovoRotulo - Gera um novo rótulo;

```
...  
Aux := IntToStr(NSeq);  
while Length(Aux) < 4 do Aux := '0'+Aux;  
Inc(NSeq);  
Rotulo := 'aek'+Aux;  
Result := Rotulo;  
...
```

Desenvolvimento

- Implementação

- ◆ RecebeText - Lê as informações do buffer de entrada;

```
...
SetLength(Result, MAXBUFF);
Len := Recv(FDSocket, Pointer(Result)^, MAXBUFF, 0);
if Len <> SOCKET_ERROR then
    SetLength(Result, Len)
else
    Result := ERRO_INTERNO;
FormDEBUG.DEBUG('S: '+Result);
...
```

Desenvolvimento

■ Implementação

◆ TipodeResposta - Identifica o tipo de Resposta.

```
if Resposta[1]= '*' then
  Result := SEM_ROTULO
else
  begin
    Posicao := Pos(' ', Resposta);
    Delete(Resposta, 1, Posicao);
    Posicao := Pos(' ', Resposta);
    Delete(Resposta, Posicao, Length(Resposta));
    if Uppercase(Resposta) = 'OK' then
      Result := OK
    else if Uppercase(Resposta) = 'NO' then
      Result := NO
    else if Uppercase(Resposta) = 'BAD' then
      Result := BAD
    else
      Result := DESCONHECIDA;
  end;
```

Desenvolvimento

- Implementação
 - ◆ Efetua_LOGIN - Efetua o logon;

```
...
if Envia_Comando('LOGIN', Login+' '+Senha) = 0 then
begin
    Resultado := RecebeText;
    // Verifica se o LOGIN
    if TipodeResposta(Resultado) = OK then
        Logado := True
    else
        begin
...

```

Desenvolvimento

- Implementação
 - ◆ Examine - Seleciona a caixa postal;

```
...  
if Envia_Comando('EXAMINE', CaixaPostal) = 0 then  
begin  
    Aux := RecebeText;  
    if Aux <> ERRO_INTERNO then  
...  
end
```

Desenvolvimento

- Implementação (continuação)
 - ◆ Search - Busca a relação da novas mensagens;

```
...  
if Envia_Comando('SEARCH', 'RECENT') = 0 then  
  begin  
  
  Aux := RecebeText;  
    if Aux <> ERRO_INTERNO then  
      while Length(AUX) > 0 do  
  
  ...
```

Desenvolvimento

- Implementação (continuação)
 - ◆ Fetch - Recupera a informação da nova mensagem;

```
...  
if Envia_Comando('FETCH', Email+' (BODY[HEADER.FIELDS (FROM  
SUBJECT DATE MESSAGE-ID)] RFC822.SIZE)') = 0 then  
  begin  
    Aux := RecebeText;  
    if Aux <> ERRO_INTERNO then  
      ...  
  end  
end
```

Desenvolvimento

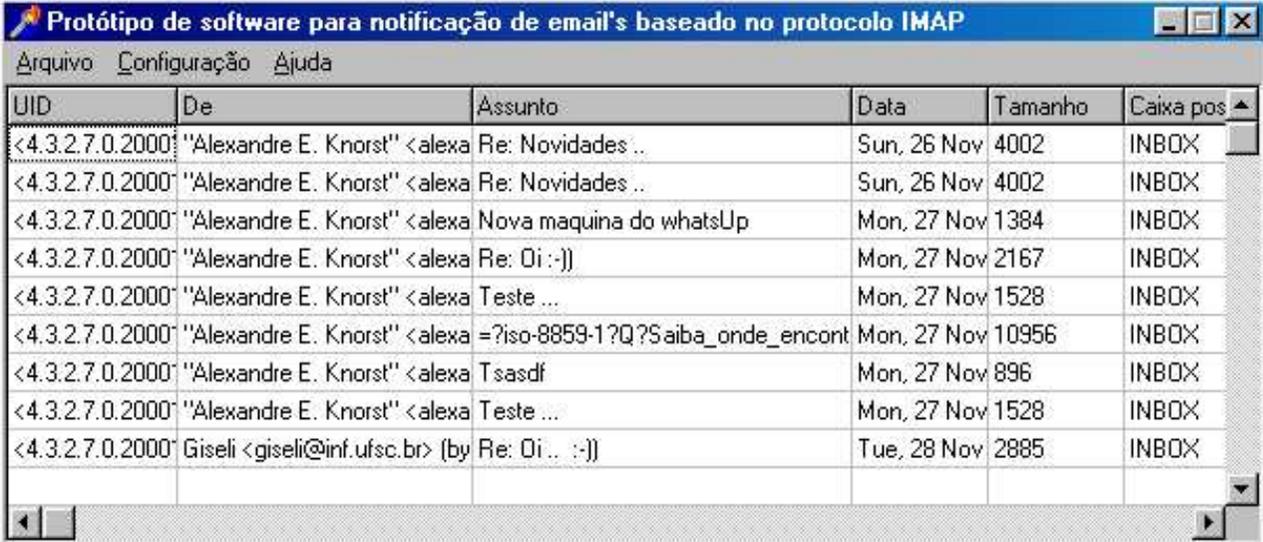
- Implementação (continuação)
 - ◆ Logout - Fecha a conexão.

```
...
if Envia_Comando('LOGOUT', '') = 0 then
begin
  Aux := RecebeText;
  if Aux <> ERRO_INTERNO then
...

```

Desenvolvimento

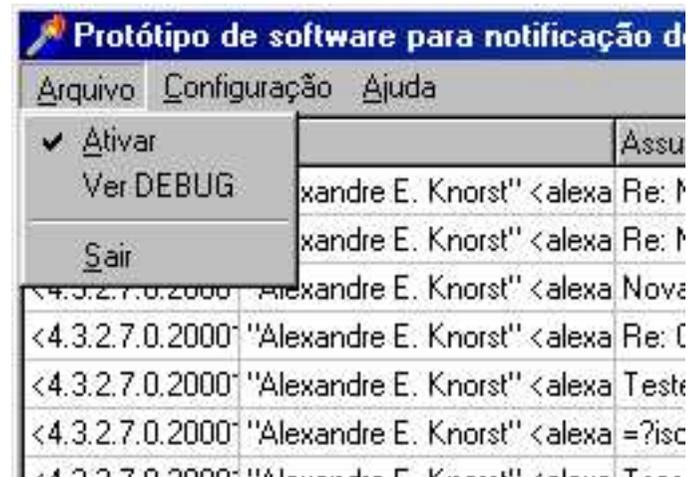
- Funcionalidade do protótipo
 - ◆ Tela principal



UID	De	Assunto	Data	Tamanho	Caixa pos
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	Re: Novidades ..	Sun, 26 Nov	4002	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	Re: Novidades ..	Sun, 26 Nov	4002	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	Nova maquina do whatsUp	Mon, 27 Nov	1384	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	Re: Oi :-))	Mon, 27 Nov	2167	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	Teste ...	Mon, 27 Nov	1528	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	=?iso-8859-1?Q?Saiba_onde_encont	Mon, 27 Nov	10956	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	Tsasdf	Mon, 27 Nov	896	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	"Alexandre E. Knorst" <alexaknorst@inf.ufsc.br>	Teste ...	Mon, 27 Nov	1528	INBOX
<4.3.2.7.0.2000@inf.ufsc.br>	Giseli <giseli@inf.ufsc.br>	[by Re: Oi .. :-))	Tue, 28 Nov	2885	INBOX

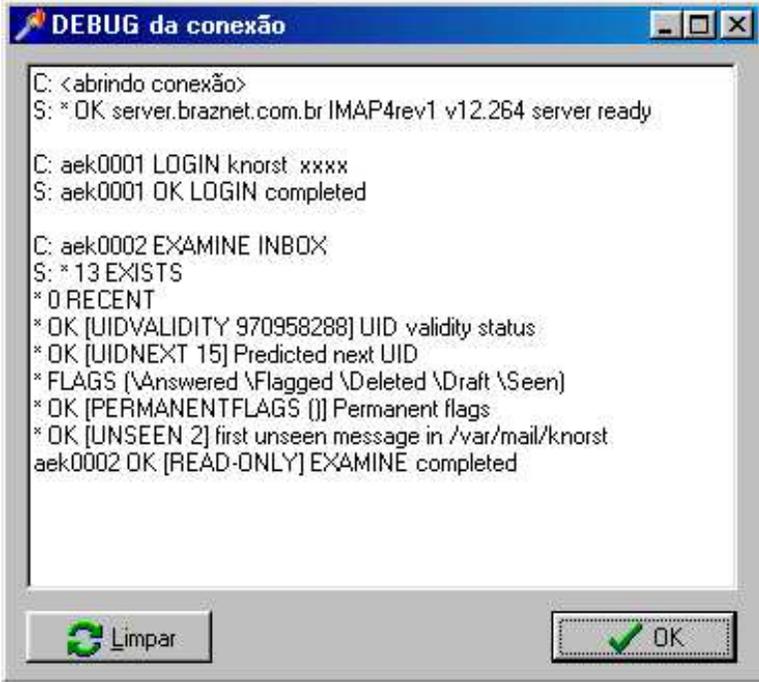
Desenvolvimento

- Funcionalidade do protótipo
 - ◆ Menu “Arquivo”



Desenvolvimento

- Funcionalidade do protótipo
 - ◆ Tela “Debug”



```
DEBUG da conexão
C: <abrindo conexão>
S: * OK server.braznet.com.br IMAP4rev1 v12.264 server ready

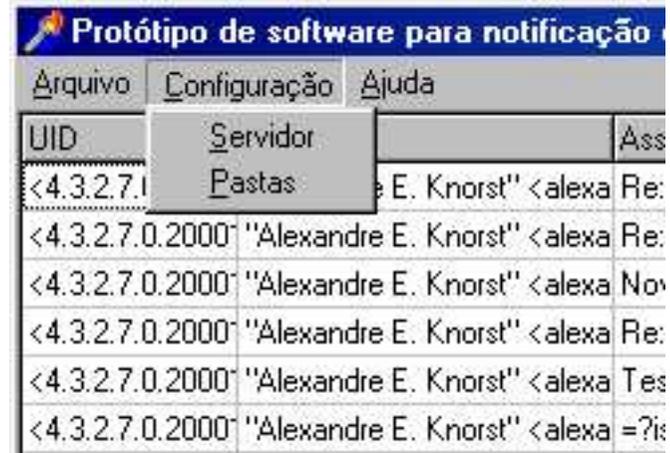
C: aek0001 LOGIN knorst xxxx
S: aek0001 OK LOGIN completed

C: aek0002 EXAMINE INBOX
S: * 13 EXISTS
* 0 RECENT
* OK [UIDVALIDITY 970958288] UID validity status
* OK [UIDNEXT 15] Predicted next UID
* FLAGS (\Answered \Flagged \Deleted \Draft \Seen)
* OK [PERMANENTFLAGS ()] Permanent flags
* OK [UNSEEN 2] first unseen message in /var/mail/knorst
aek0002 OK [READ-ONLY] EXAMINE completed
```

Limpar OK

Desenvolvimento

- Funcionalidade do protótipo
 - ◆ Menu “Configuração”



The screenshot shows a window titled "Protótipo de software para notificação" with a menu bar containing "Arquivo", "Configuração", and "Ajuda". The "Configuração" menu is open, showing "Servidor" and "Pastas". Below the menu, a list of email notifications is displayed with columns for UID, Sender, and Subject.

UID	Servidor	Pastas	Ass
<4.3.2.7.1	E. Knorst"	<alexa	Re:
<4.3.2.7.0.2000	"Alexandre E. Knorst"	<alexa	Re:
<4.3.2.7.0.2000	"Alexandre E. Knorst"	<alexa	Nov
<4.3.2.7.0.2000	"Alexandre E. Knorst"	<alexa	Re:
<4.3.2.7.0.2000	"Alexandre E. Knorst"	<alexa	Tes
<4.3.2.7.0.2000	"Alexandre E. Knorst"	<alexa	=?is

Desenvolvimento

- Funcionalidade do protótipo
 - ◆ Tela “Configuração do Servidor”

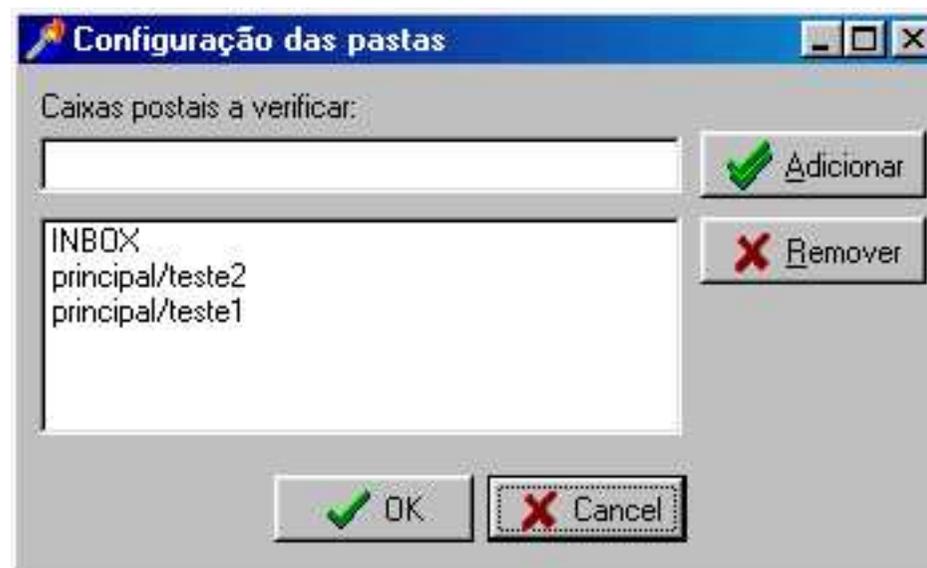


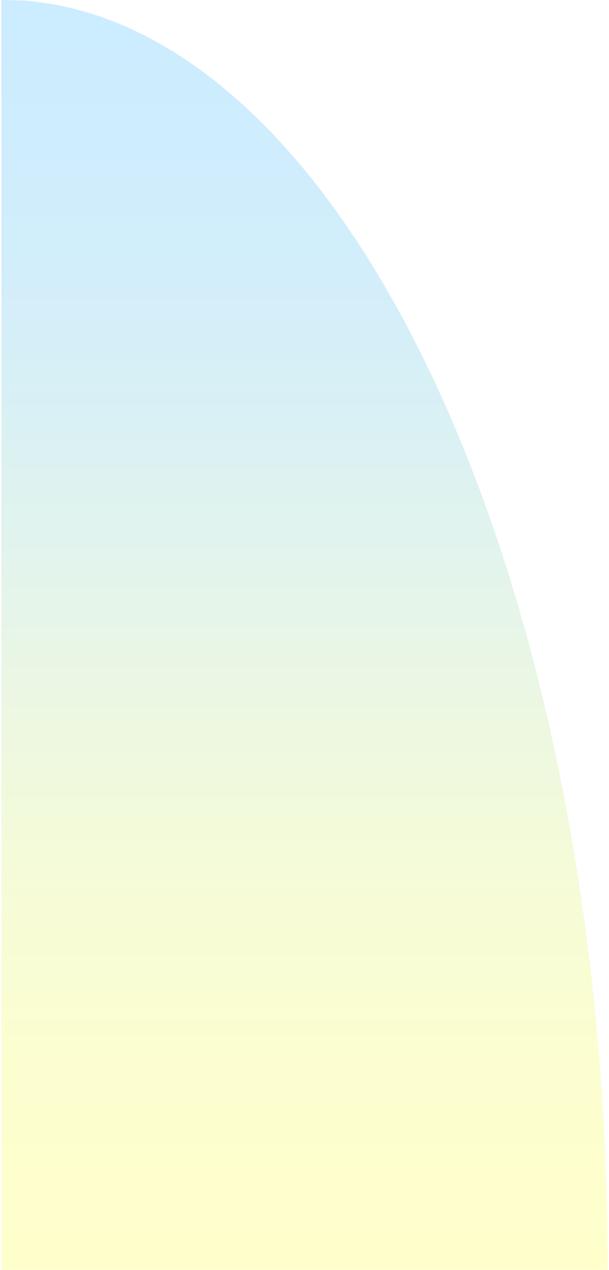
The image shows a screenshot of a Windows-style dialog box titled "Configuração do Servidor". The dialog box has a blue title bar with a small icon on the left and standard window control buttons (minimize, maximize, close) on the right. The main area is light gray and contains the following fields and controls:

- Servidor (IMAP):** A text input field containing "server.braznet.com.br".
- Porta:** A spin box containing the number "143".
- Login:** A text input field containing "knorst".
- Senha:** A text input field containing "#####".
- Verificar por novas mensagens a cada:** A label above a spin box containing the number "1".
- Minuto(s):** A label to the right of the spin box.
- Buttons:** Two buttons at the bottom: "OK" with a green checkmark icon and "Cancel" with a red X icon.

Desenvolvimento

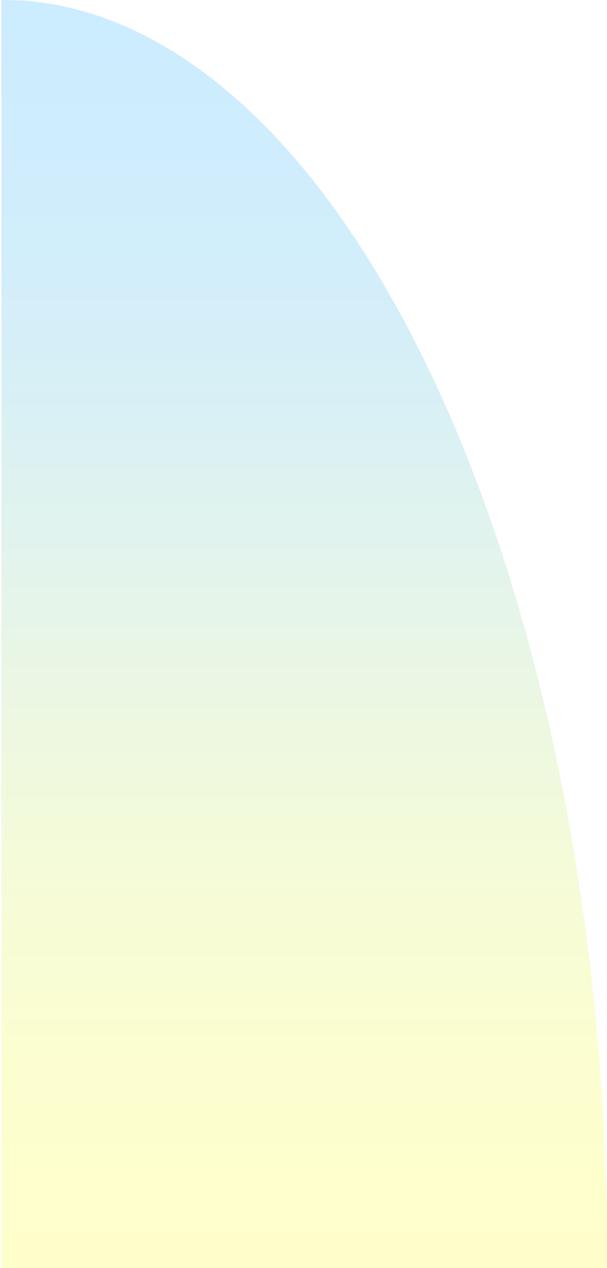
- Funcionalidade do protótipo
 - ◆ Tela “Configuração das pastas”





Conclusões

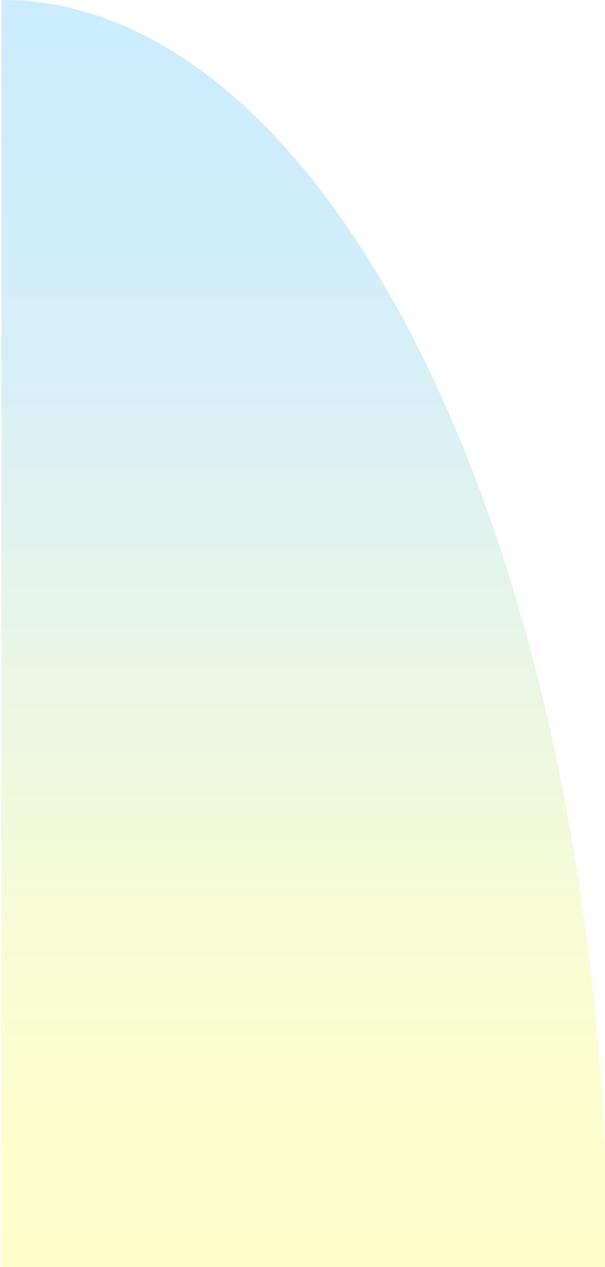
- Pouca Referência Bibliográfica;
- Protocolo pouco difundido na Internet;
- Objetivos foram alcançados;



Conclusões

- **Extensões:**

- ◆ Permitir que se exclua mensagens do servidor;
- ◆ Permitir alterar as FLAG da mensagem;
- ◆ Criar e excluir caixas postais;
- ◆ Mover mensagens de uma caixa postal para outra, utilizando o comando COPY;
- ◆ Implementar uma opção para que este mesmo protótipo possa ser utilizado com o protocolo POP.



Referência Bibliográfica

[CRI1996] CRISPIN, M; Washington. **RFC-2060 - IMAP:**
internet message access protocol 1996. Endereço
Eletrônico: <http://www.isi.edu/in-notes/rfc2060.txt>. Data da
consulta: 26/06/2000.

[MYR1996] MYRES, J.; Mellon, Carnegie; Rose, M.; Dover
Beach Consulting, Inc. **RFC-1939 – POP – post office
protocol – version 3** 1996. Endereço Eletrônico:
<http://www.isi.edu/in-notes/rfc1939.txt>. Data da consulta:
26/06/2000.

[POS1982] POSTEL, Jonathan B. **RFC821 – SMTP – simple
mail transfer protocol** 1996. Endereço Eletrônico:
<http://www.isi.edu/in-notes/rfc821.txt>. Data da consulta:
26/06/2000.