

SyncEasy – Aplicativo para sincronização de arquivos entre dispositivos móveis e computadores utilizando metadados

Acadêmico: Bernardo Marquardt Müller

Orientador: Prof. Dr. Mauro Marcelo Mattos

Roteiro

- ▶ Introdução
- ▶ Objetivos do trabalho
- ▶ Fundamentação teórica
- ▶ Desenvolvimento do trabalho
- ▶ Resultados
- ▶ Conclusão



Introdução

- ▶ Produção de conteúdo
- ▶ Vida digital
- ▶ Duplicação de arquivos
- ▶ Sincronização



Objetivos do trabalho

- ▶ Disponibilizar um algoritmo de sincronização
- ▶ Sincronizar arquivos entre celular e computador



Fundamentação teórica

- ▶ Java Micro Edition (J2ME)
- ▶ Sistema de arquivos Symbian
- ▶ Metadados
- ▶ Algoritmos de sincronização



J2ME

- ▶ Criada para dispositivos com pouco poder de processamento
- ▶ Dividida em configurações e perfis
 - ▶ Configurações:
 - ▶ CDC – Connected Device Configuration
 - Perfil – Foundation Profile
 - ▶ CLDC – Connected Limited Device Configuration
 - Perfil – MIDP (Mobile Information Device Profile)



Sistema de arquivos Symbian

- ▶ Suporta até 26 unidades
- ▶ Unidades no formato conhecido do DOS
- ▶ Firmware reside na unidade Z:
- ▶ Unidade C: é o padrão para os usuários
- ▶ A partir da D: é para dispositivos removíveis



Metadados

- ▶ Dados que descrevem dados
- ▶ Armazenam apenas informações relevantes dos arquivos
- ▶ Facilitam a busca



Algoritmos de sincronização

- ▶ Dificuldade na sincronização
 - ▶ Definição dos parâmetros de sincronização
 - ▶ Velocidade do meio de comunicação
 - ▶ Quantidade de arquivos a serem sincronizados
- ▶ Estratégia de replicação otimista
- ▶ Algoritmo *rsync*
- ▶ Algoritmo baseado em metadados



RSYNC

- ▶ Muito utilizado em sistemas Unix
- ▶ Útil em sincronizações onde ambas as partes possuem alguma versão do arquivo
- ▶ Quebra o arquivo em partes para a transmissão
- ▶ Transmite somente o necessário



Baseado em metadados

- ▶ Cria um dicionário de dados
- ▶ Trata arquivos inteiros
- ▶ Facilita o gerenciamento de muitos arquivos



Trabalhos correlatos

- ▶ Dropbox (DROPBOX, 2010)
- ▶ ActiveSync (MICROSOFT, 2007)



Desenvolvimento do trabalho

- ▶ **Principais requisitos**

- ▶ Funcionais
- ▶ Não-Funcionais

- ▶ **Especificação**

- ▶ Técnicas e ferramentas utilizadas
- ▶ Diagrama de Casos de Uso
- ▶ Diagrama de Classes
- ▶ Diagrama de Sequência

- ▶ **Implementação**

- ▶ Técnicas e ferramentas utilizadas
- ▶ Operacionalidade da implementação



Requisitos funcionais

- ▶ Permitir o espelhamento completo de um diretório, juntamente com os seus subdiretórios
- ▶ Sincronizar bidirecionalmente um dispositivo móvel e um computador
- ▶ Identificar arquivos duplicados



Requisitos não funcionais

- ▶ Utilizar a linguagem JEE na implementação do servidor
- ▶ Utilizar a linguagem J2ME para a implementação do cliente
- ▶ Utilizar a plataforma móvel *Symbian* para a validação do protótipo.



Técnicas e ferramentas utilizadas

- ▶ Enterprise Architect – Diagramas de Casos de Uso, Classes e Sequência



Diagrama de caso de uso

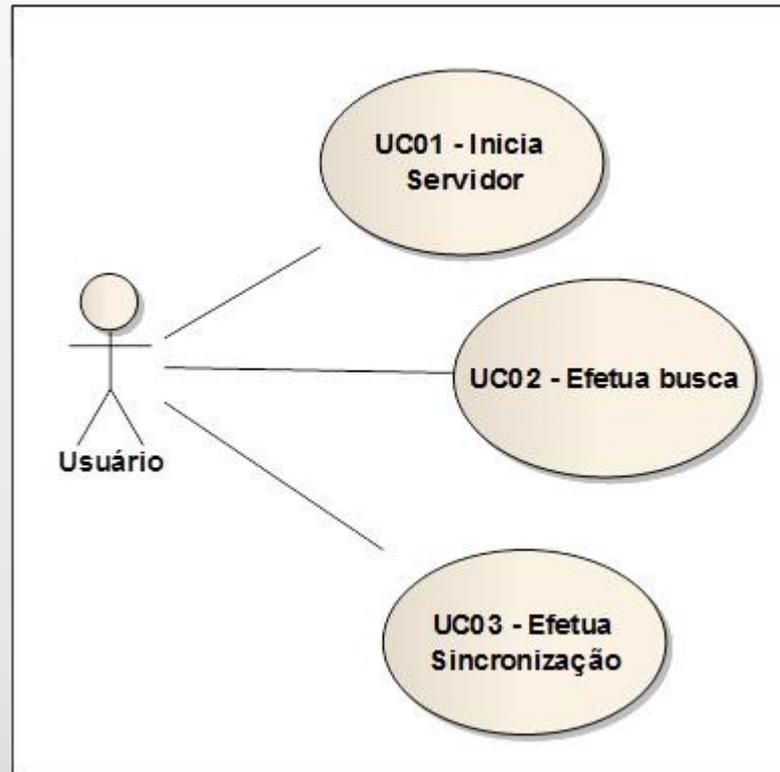


Diagrama de classes - cliente

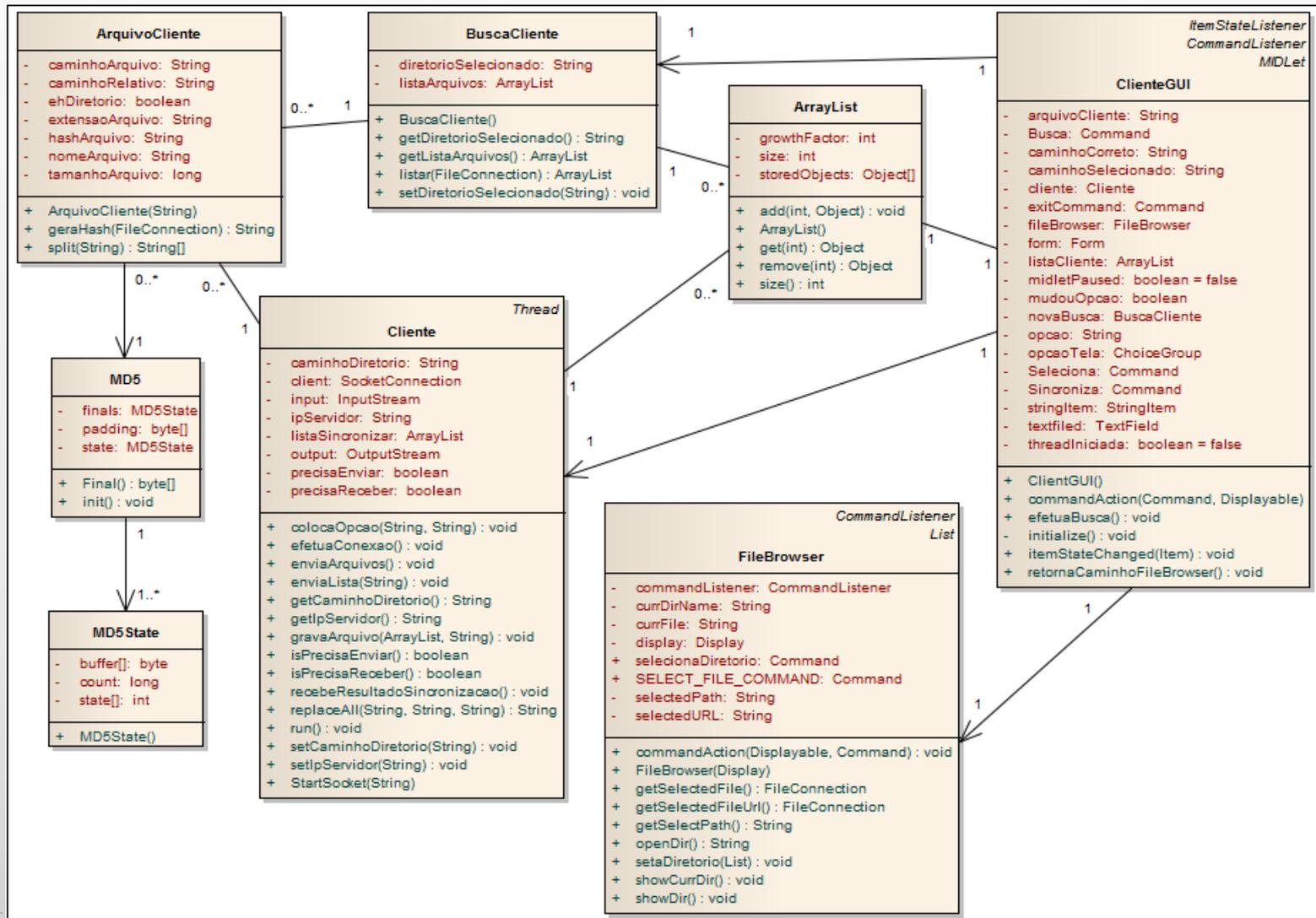


Diagrama de classes - servidor

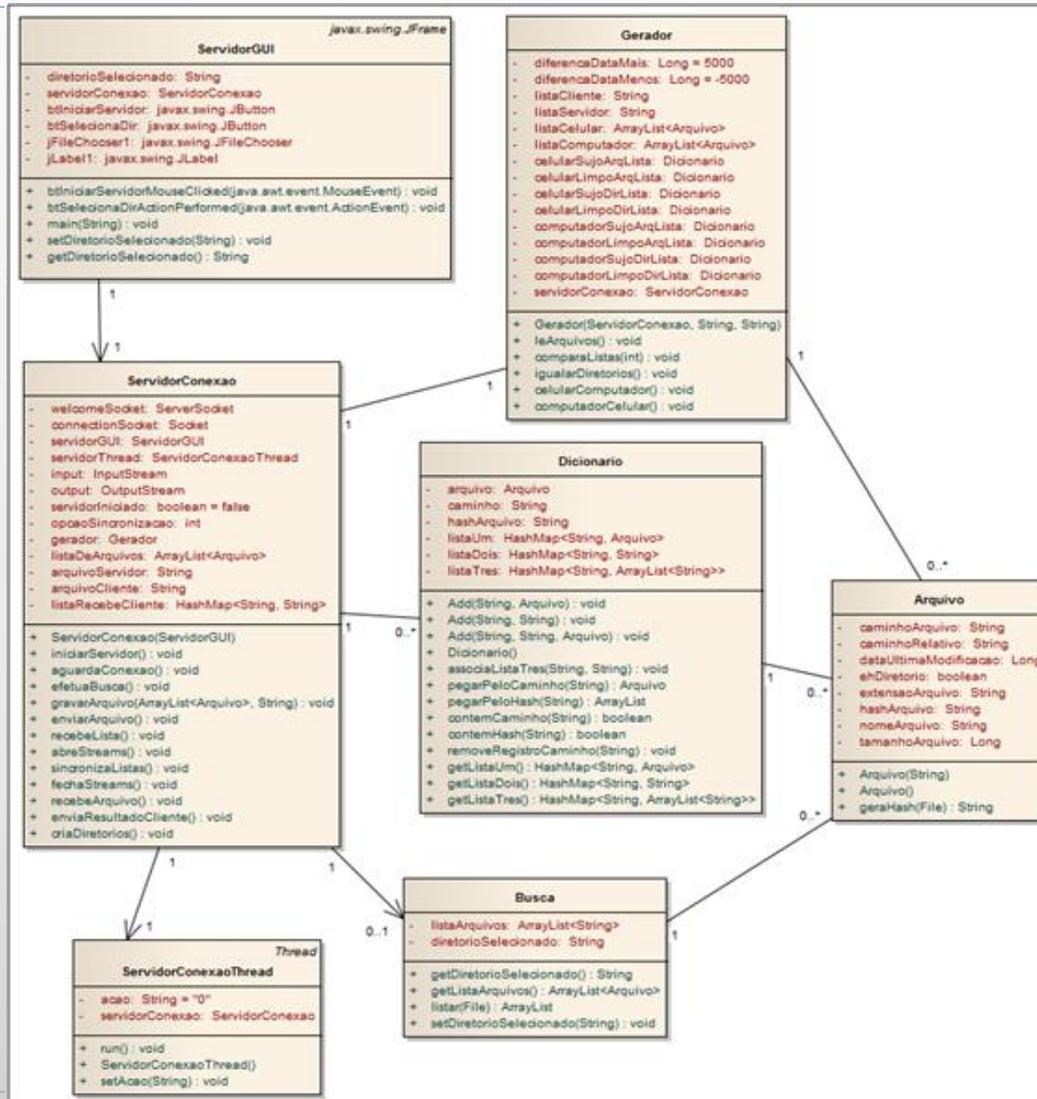


Diagrama de sequência

▶ Caso de uso UC01 – Inicia Servidor

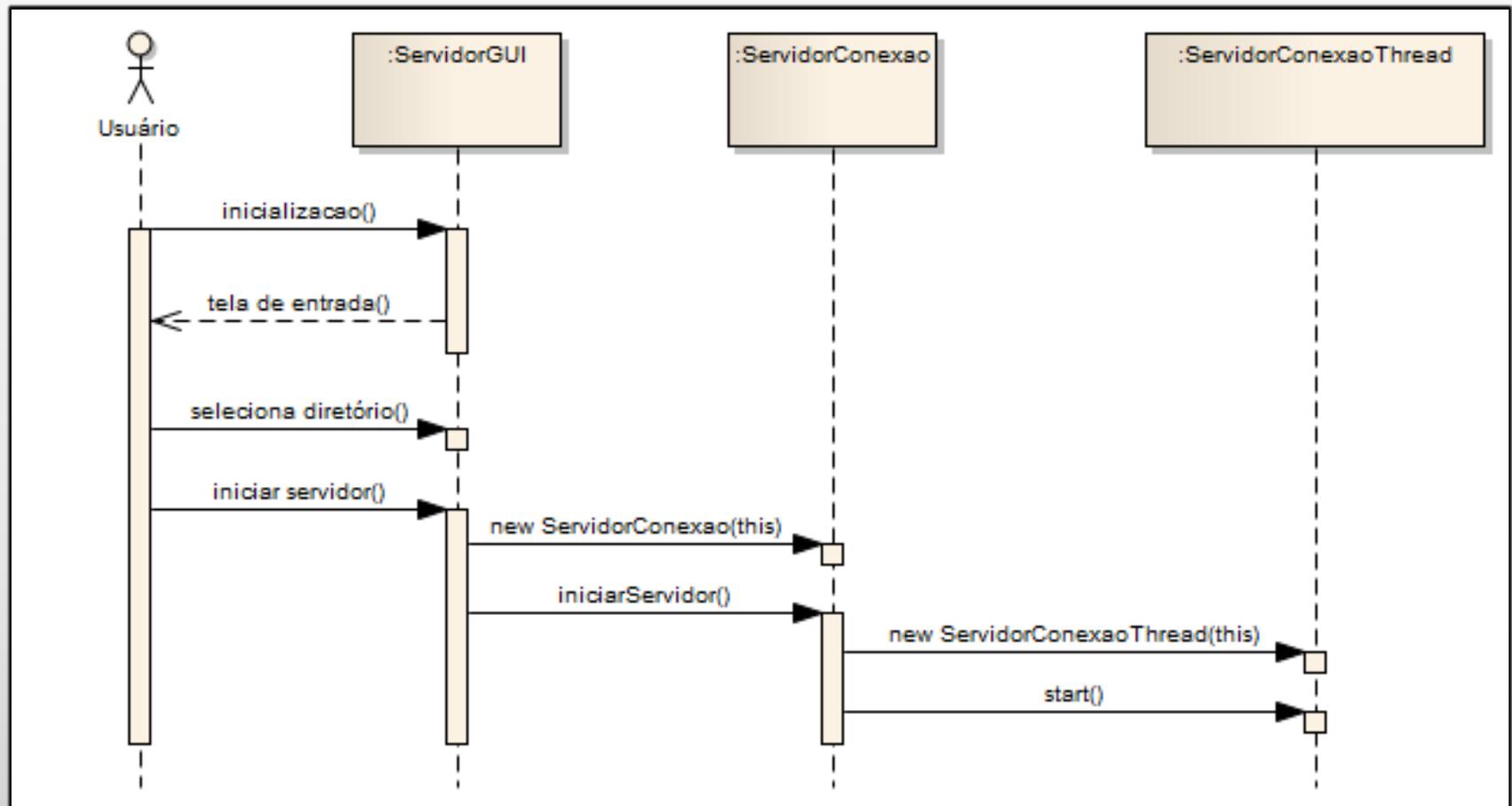


Diagrama de sequência

▶ Caso de uso UC 02 – Efetua Busca

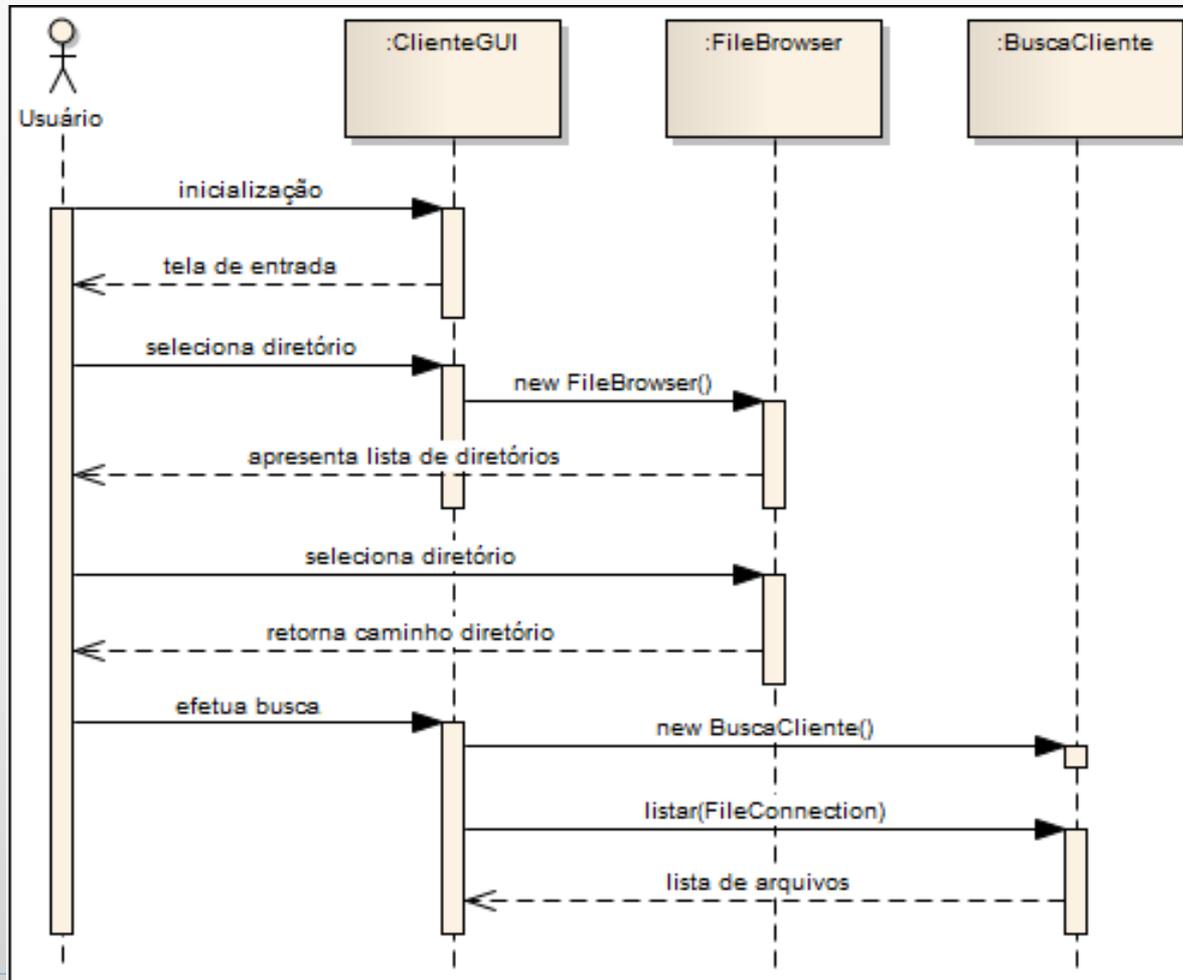
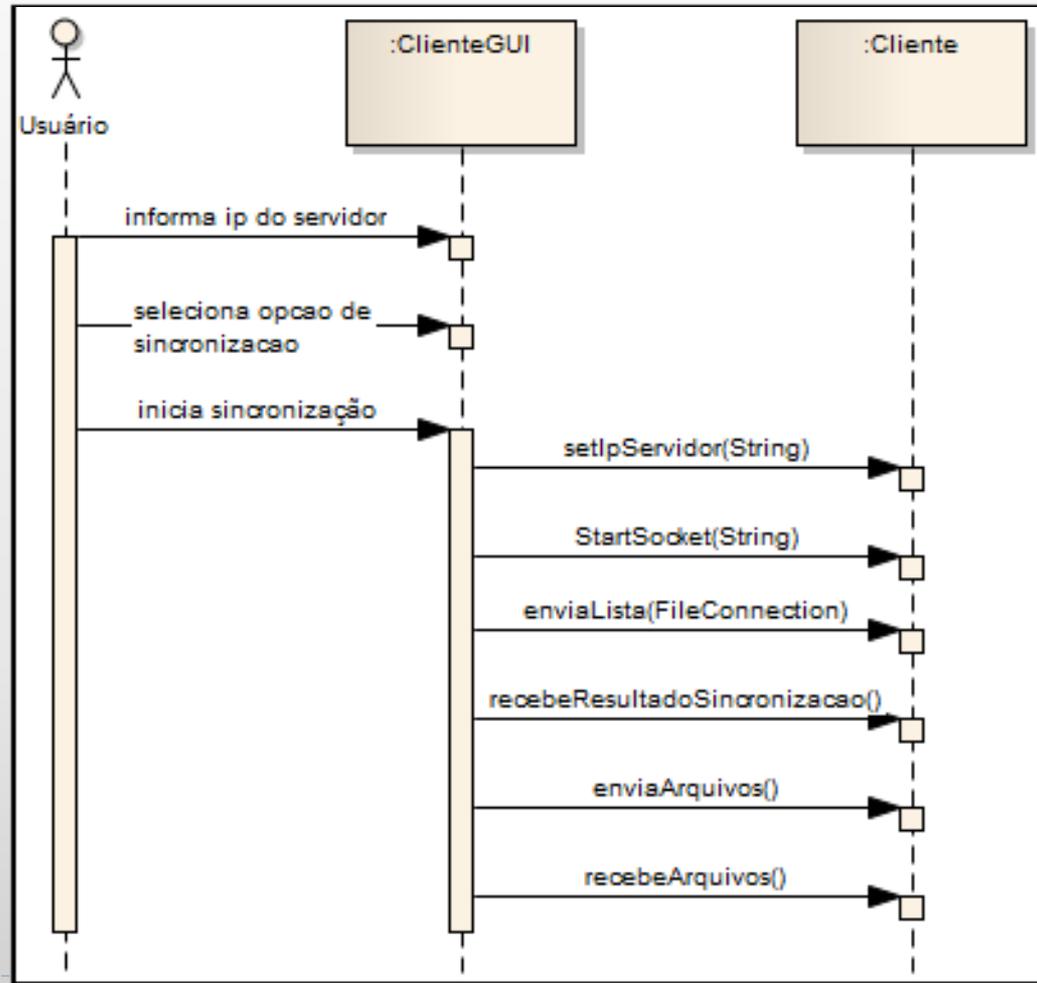


Diagrama de Sequência

▶ Caso de uso UC03 – Efetua Sincronização



Técnicas e ferramentas utilizadas

- ▶ Netbeans 6.9.1 – Programação do cliente e do servidor
- ▶ Componente *File Browser*



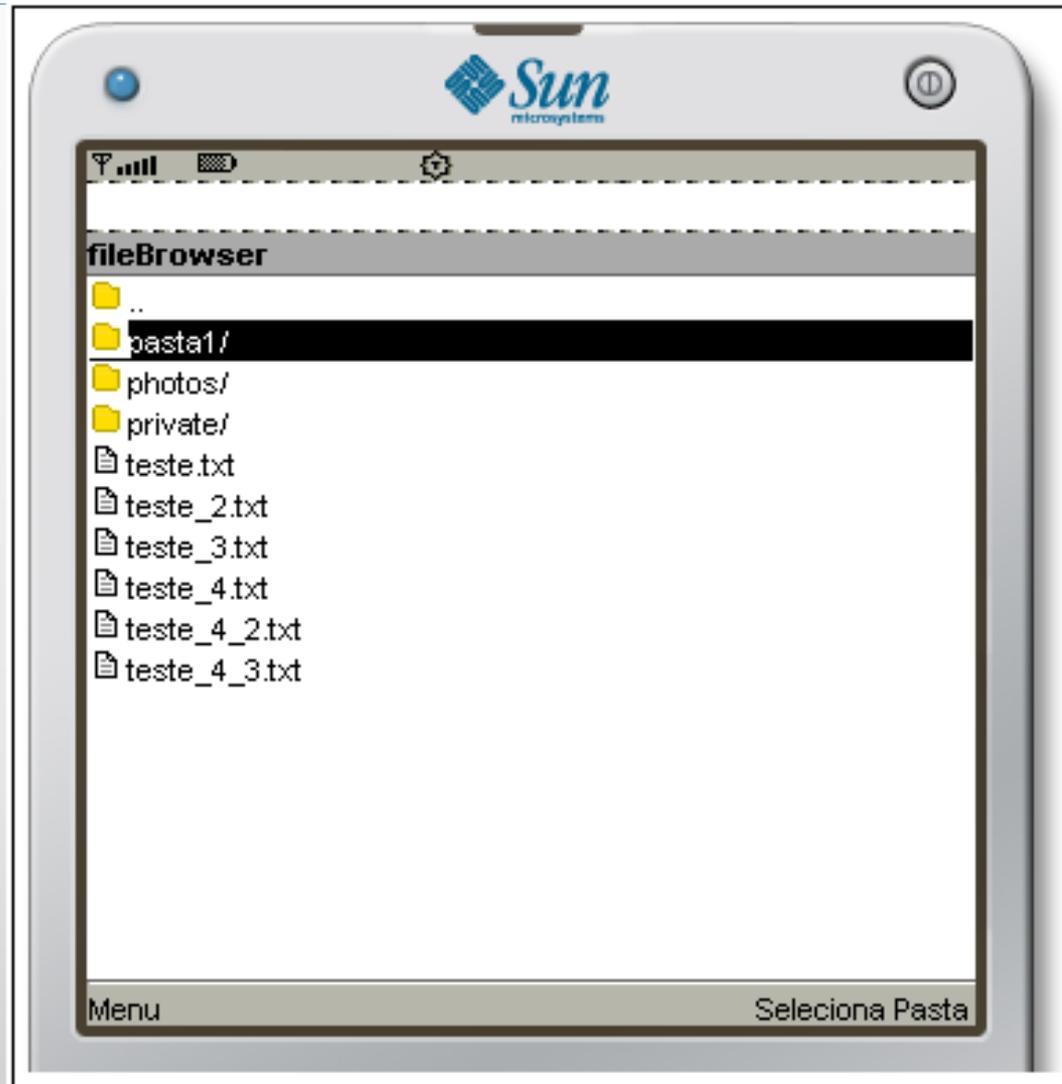
Componente *File Browser*

- ▶ Utilizado para acessar o sistema de arquivos do celular
- ▶ Interface gráfica
- ▶ Facilidade de navegação



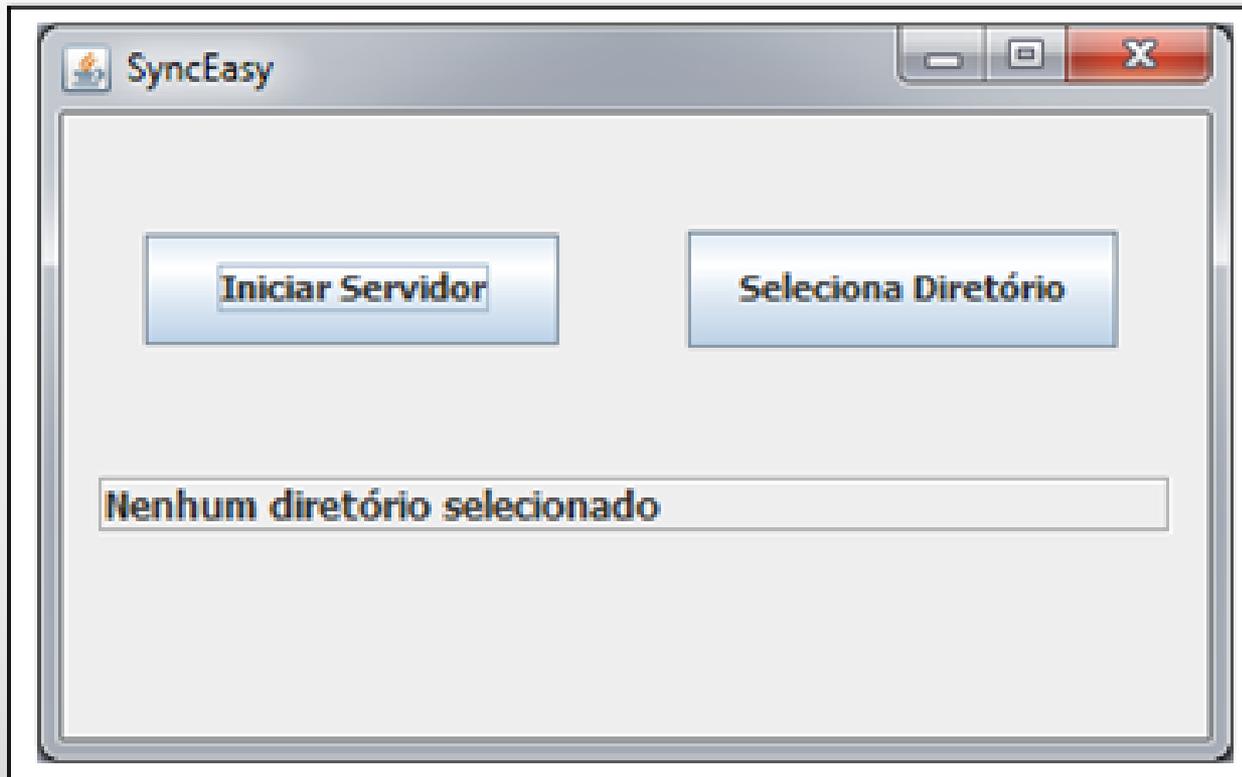
Componente *File Browser*

- ▶ Componente em uso



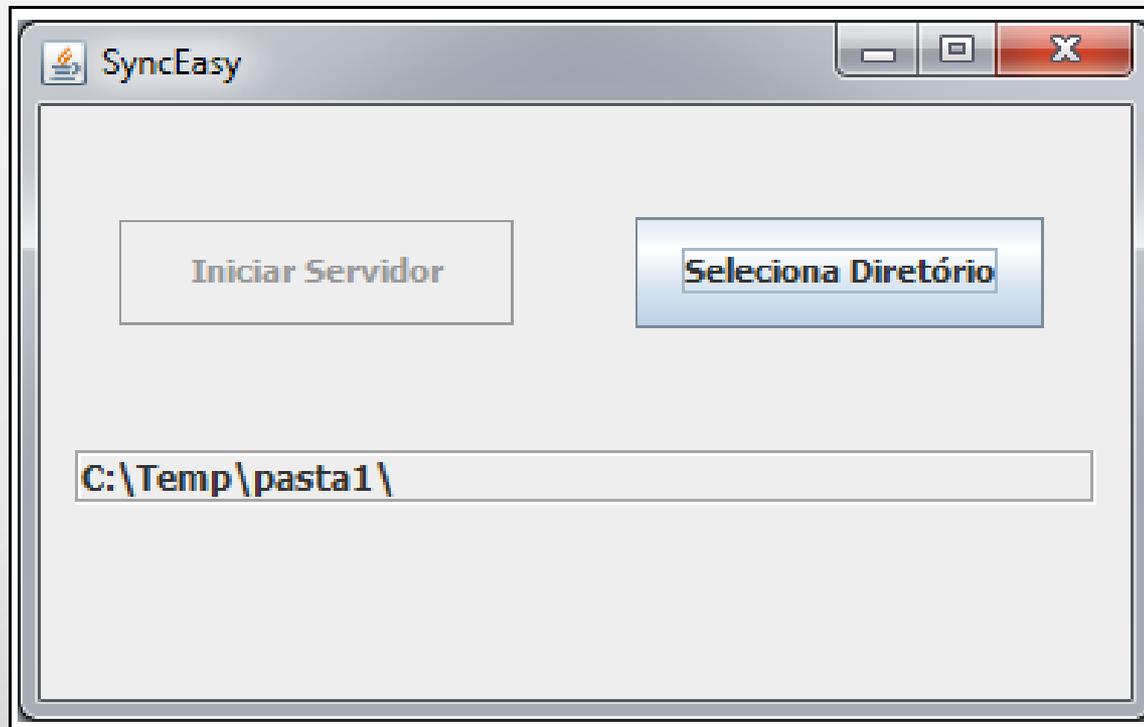
Operacionalidade da implementação

▶ Servidor



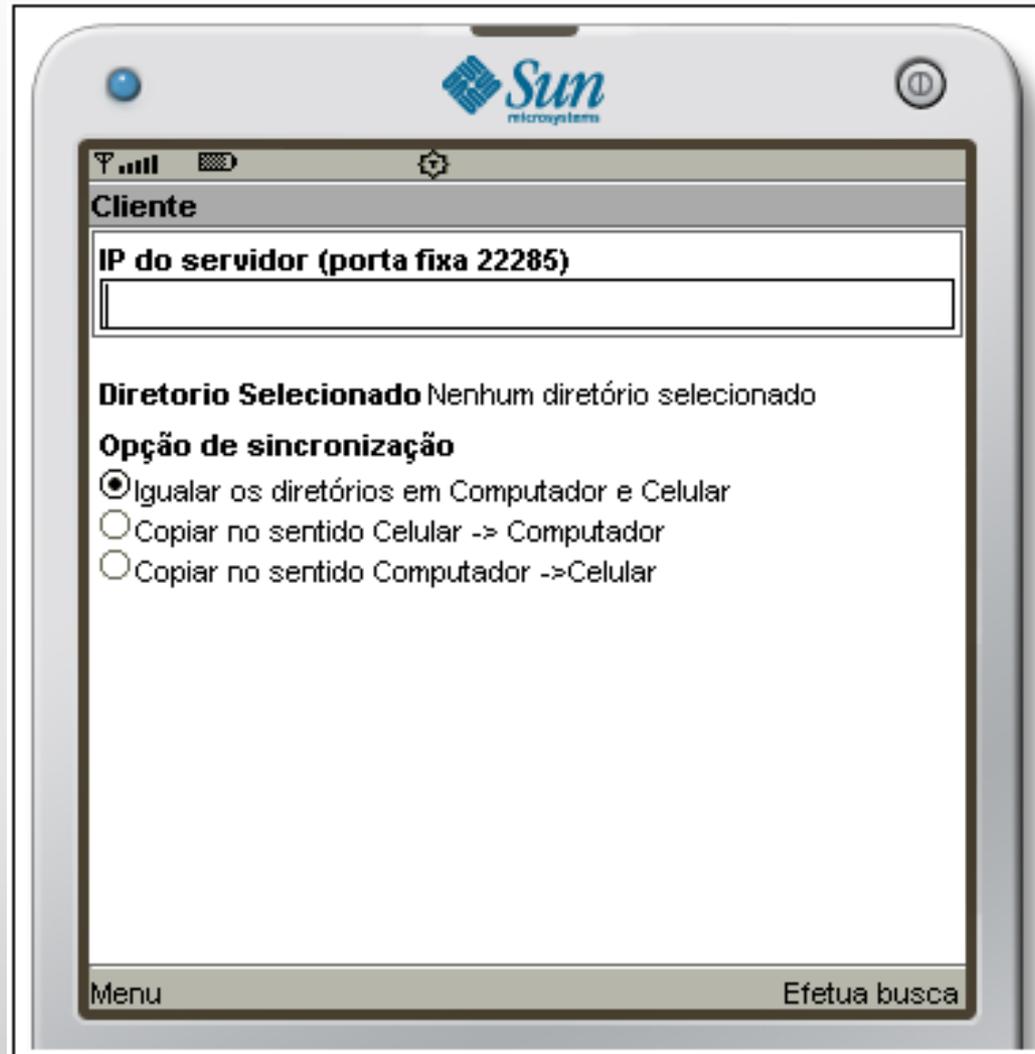
Operacionalidade da implementação

▶ Servidor



Operacionalidade da implementação

▶ Cliente

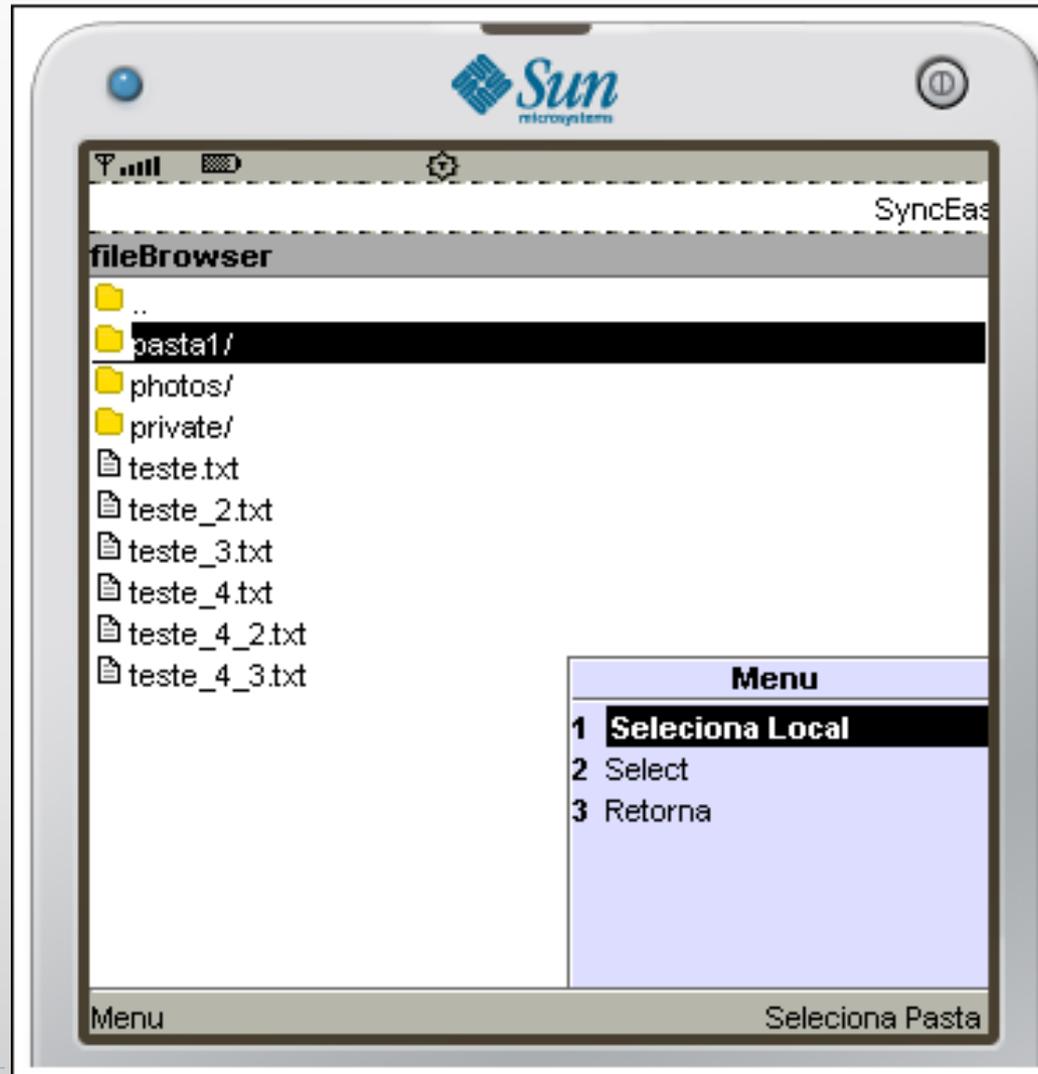


The image shows a screenshot of a mobile application interface on a Sun Microsystems device. The interface is titled "Cliente" and contains the following elements:

- Header:** Sun Microsystems logo and a power button icon.
- Status Bar:** Signal strength, battery level, and a gear icon.
- Title:** "Cliente"
- Form Field:** "IP do servidor (porta fixa 22285)" with an empty input box.
- Text:** "Diretorio Selecionado Nenhum diretório selecionado"
- Section Header:** "Opção de sincronização"
- Radio Buttons:**
 - Igualar os diretórios em Computador e Celular
 - Copiar no sentido Celular -> Computador
 - Copiar no sentido Computador -> Celular
- Footer:** "Menu" on the left and "Efetua busca" on the right.

Operacionalidade da implementação

▶ Cliente



Operacionalidade da implementação

▶ Cliente



Operacionalidade da implementação

- ▶ Exemplo de lista gerada com os metadados

```
pasta \;0; \root1 \pasta1 \subnivel \pasta \;0;dir;pasta \;  
\subnivel \pasta \;
```

```
6513.txt;884; \root1 \outra \6513.txt;1306151530254;  
txt;da0a6248a0a57765b1086c7326283c3a; \6513.txt;
```

```
8+8.txt;874; \root1 \outra \8+8.txt;1306151508199;txt;  
826c0013193aeae5c0625721dcedc3c4; \8+8.txt;
```



Operacionalidade da implementação

- ▶ Exemplo de log gerado após a sincronização

Copiado para CELULAR

\algo_3.txt

\subnive\segundo\

\subnive\segundo\bernardo.txt

\subnive\segundo-2\denovo.txt

Copiado para COMPUTADOR

\subnive\oie.txt

\subnive\outro\mais um\123456.txt



Resultados e discussão

- ▶ Eficiente na sincronização de arquivos
- ▶ Dicionário de metadados é adequado a solução
- ▶ Software de fácil utilização



Conclusão

- ▶ Acuracidade nos resultados
- ▶ Algoritmo de casamento de arquivos se mostrou eficiente
- ▶ Ferramenta pode ser portada para outras plataformas de celulares
- ▶ Objetivos foram atingidos



Conclusão

▶ Dificuldades

- ▶ Implementação de sockets
- ▶ Uso de sockets em plataforma Symbian inviabilizou o perfeito funcionamento nesta plataforma
- ▶ Limitações da linguagem J2ME



Extensões

- ▶ Implementar uma rotina de hash (MD5) de arquivo em J2ME;
- ▶ Adicionar algum nível de segurança, por login e senha ou encriptação dos dados trafegados na rede;
- ▶ Solucionar o problema de envio de dados via socket na plataforma Symbian;
- ▶ Disponibilizar mais opções de configuração para o servidor e o cliente;
- ▶ Implementar um serviço que fique monitorando o diretório selecionado para identificar mudanças nos arquivos;
- ▶ Acrescentar a visualização dos arquivos a serem sincronizados no celular.

