

Protótipo de uma Aplicação
LBS Utilizando GPS Conectado
em Celular para Consultar
Dados Georeferenciados

Alex Kuhnen

Prof. Francisco Adell Péricas



Introdução

- Transição dos Sistemas de Informação Geográficas (SIG) para os Serviços Baseados em Localização (LBS)
- Fusão entre dispositivos móveis e a internet
- Java 2 Micro Edition como solução para aplicações voltadas ao mundo sem-fio



Objetivos

- Desenvolver um protótipo de software para telefone celular que permitirá ao usuário fazer consultas via web a uma base de dados georeferenciados
- Objetivos específicos
 - Conectividade da API J2ME
 - Comunicação entre GPS e aplicativo cliente
 - Comunicação com a web
 - Tratar as requisições no servidor WEB

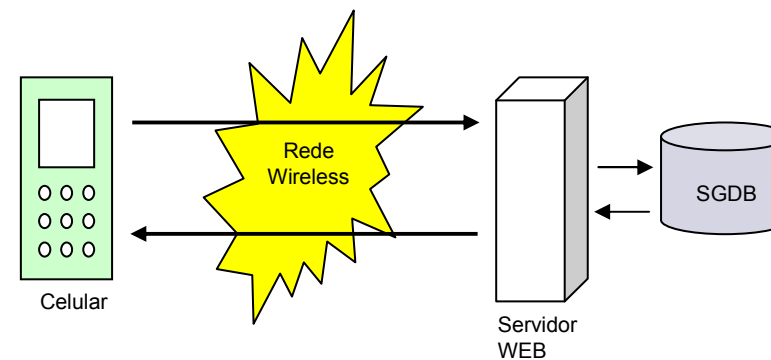


Sistemas de Informações Geográficas

- Sistemas automatizados usados para armazenar, analisar e manipular dados geográficos
- Redução de custos na produção e manutenção de mapas
- LBS

Location Based Services (LBS)

- São serviços que permitem aos usuários móveis utilizarem serviços baseados na sua posição ou localização geográfica
- Convergências de tecnologias
 - SIGs
 - Internet
 - Comunicação sem fio
 - Dispositivos portáteis



Arquitetura de uma aplicação LBS



LBS: Tecnologias de Localização

- Handset-based
 - Global Positioning System (GPS)
- Network-based
 - Cell Identification (Cell ID)
 - Angle of Arrival (AOA)
 - Time Difference of Arrival (TDOA)
- Híbrida
 - Assisted-GPS (A-GPS)



Global Positioning System

- Sinais de pelo menos 4 satélites em órbita são recebidos pelo dispositivo móvel
- Baseado na diferença de tempo entre o sinal recebido do satélite e um sinal gerado localmente
- Vantagem
 - Tecnologia muito precisa
 - Receptores menores
- Desvantagem
 - Precisão ruim em ambientes internos



JAVA

- Páginas web com conteúdo dinâmico
- Plataformas
 - Java 2 Enterprise Edition (J2EE)
 - Aplicações corporativas
 - Java 2 Standard Edition (J2SE)
 - Aplicações para desktop
 - Java 2 Micro Edition (J2ME)
 - Aplicações para pequenos dispositivos



Java 2 Micro Edition

- Voltada a dispositivos de baixa capacidade de processamento e pouca memória
- Camadas
 - Configuração
 - CLDC: Celulares, PDAs, Pagers
 - CDC: PocketPC
 - Perfil
 - MIDP, PDAP, PersonalJava
 - Kilo Virtual Machine
- Generic Connection Framework
 - HTTP/HTTS, TCP, UDP, Serial e Infravermelho



Telefonia Celular

- Deficiências tecnológicas, tamanho e custo marcaram o início da telefonia celular
- Primeira Geração (1G)
- Segunda Geração (2G)
 - 2,5G
- Terceira Geração (3G)
 - Alta velocidade, Vídeo e Música



Desenvolvimento do Protótipo

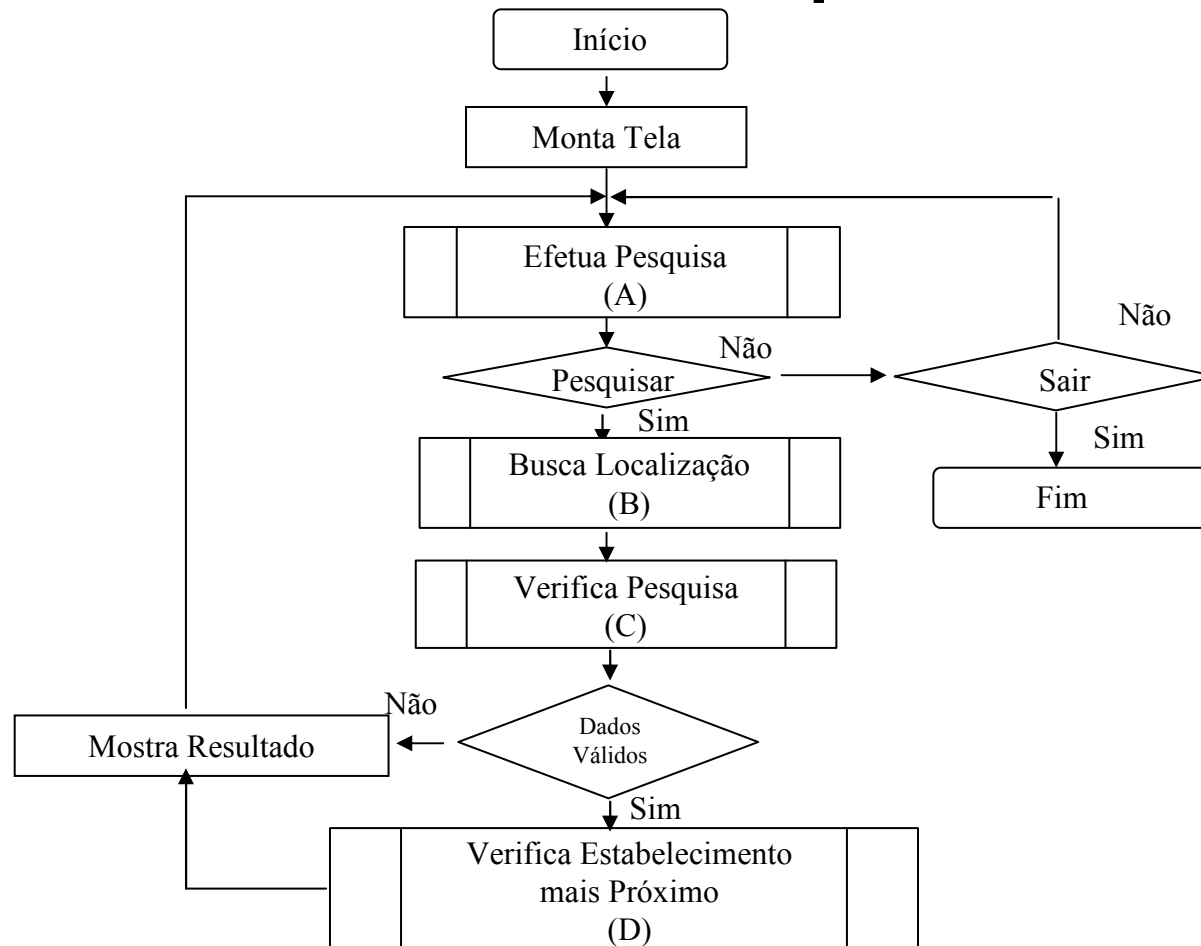
Objetivo:

- Desenvolver um protótipo de uma aplicação LBS onde se pretende identificar qual o estabelecimento mais próximo da localização informada

Requisitos identificados:

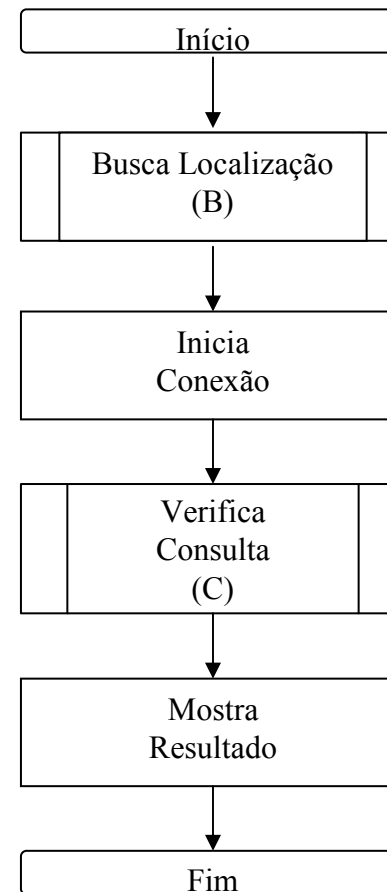
- Buscar a localização através de um GPS
- Permitir a comunicação com Servlets
- Efetuar consulta a dados geoprocessados, determinando qual o estabelecimento mais próximo da posição informada
- Apresentar os valores retornados nas consultas

Especificação: Especificação Genérica do Protótipo



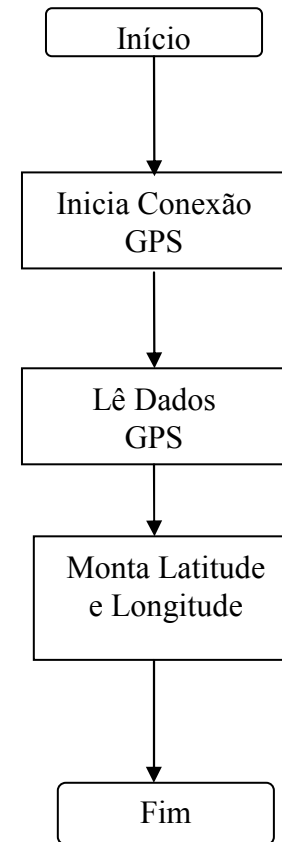
Especificação: Efetua Pesquisa (Processo A)

Busca parâmetros de localização através do processo B e inicia processo de pesquisa



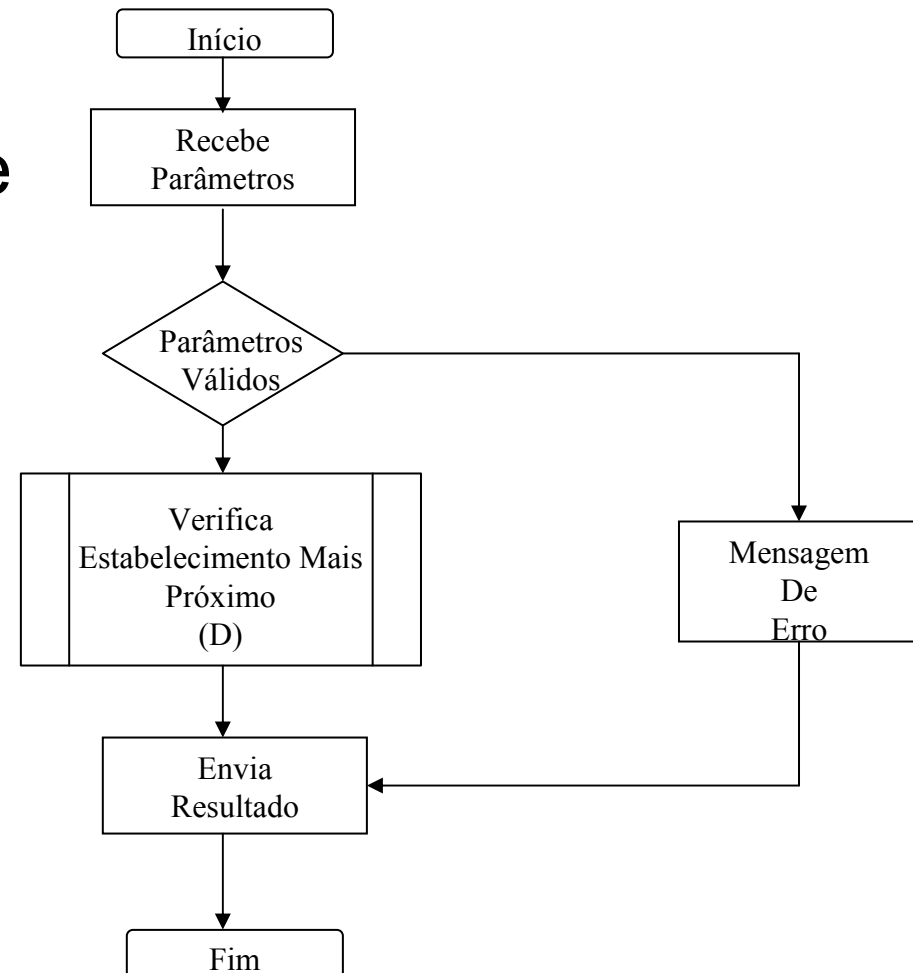
Especificação: Busca Localização (Processo B)

Determina a localização do usuário através da conexão com o GPS



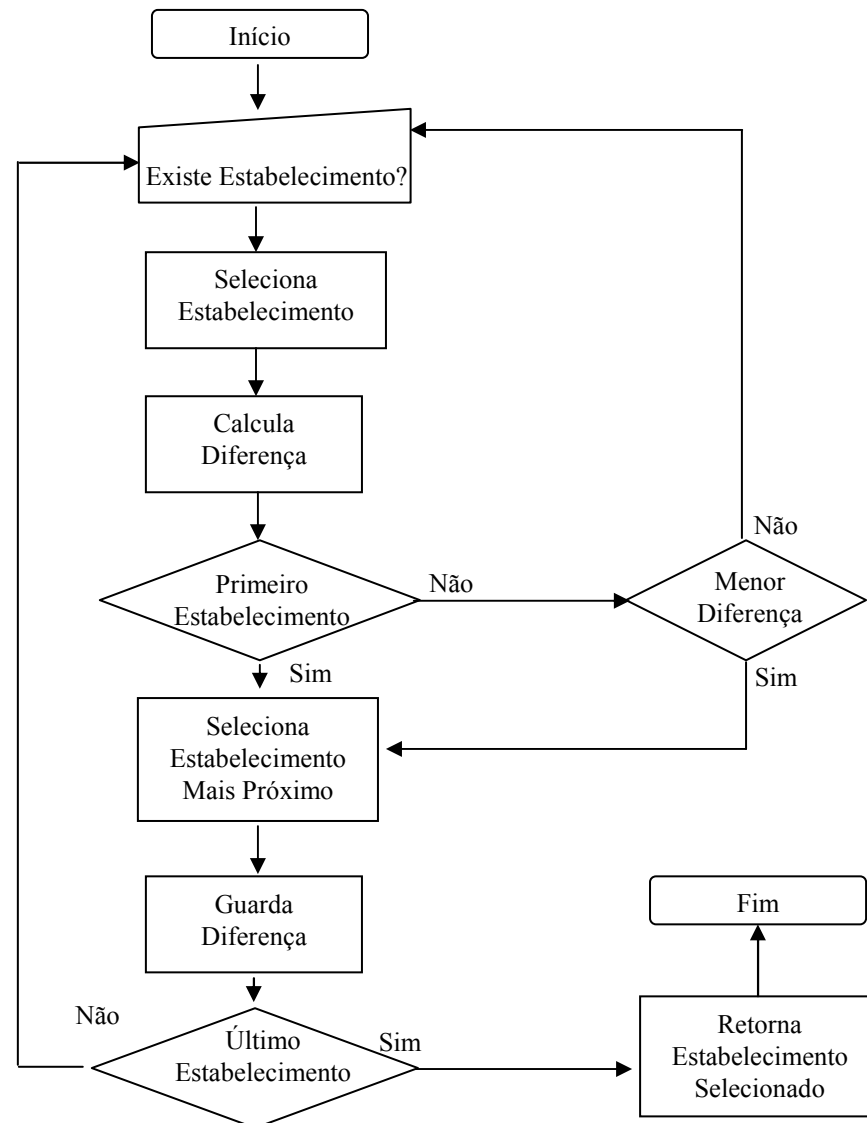
Especificação: Verifica Pesquisa (Processo C)

Valida parâmetros, chama o processo D e envia o resultado



Especificação: Verifica Estabelecimento mais Próximo (Processo D)

Determina o estabelecimento mais próximo da posição pesquisada





Implementação

■ Software Servidor

□ JDeveloper (SDK1.3.0)

■ OC4J

□ Servlet: Consulta

□ Classe: Farmacia

■ Software Cliente

□ J2ME Wireless Toolkit 2.0 (SDK 1.4.0)

■ CLDC 1.1

■ MIDP 2.0

□ MIDlet: LocaFarmaMIDlet

□ GPS

■ COM1



Implementação: Comunicação HTTP

```
private void connect() {
    HttpURLConnection hc = null;
    InputStream in = null;
    String url = getAppProperty("LocaFarmaMIDlet.URL");
    connectSerial();
        url += "posX="+lat+"&posY="+lon;
    mMessageItem.setText(url);
    try {
        hc = (HttpURLConnection)Connector.open(url);
            in = hc.openInputStream();
        int contentLength = (int)hc.getLength();
        byte[] raw = new byte[contentLength];
        int length = in.read(raw);
            // Mostra a resposta ao usuario.
        String s = new String(raw, 0, length);
        in.close();
        hc.close(); // Fecha a conexao http.
        mMessageItem.setText(s);
    }
    catch (IOException ioe) {
        mMessageItem.setText(ioe.toString());
    }
    mDisplay.setCurrent(mMainForm);
}
```



Implementação: Comunicação Serial

```
private void connectSerial() {
    CommConnection cc = null;
    InputStream is = null;
    StringBuffer inputBuffer = new StringBuffer();
    int data = 0;
    try{
        cc = (CommConnection)
Connector.open("comm:com1;baudrate=9600;bitsperchar=8;parity=none");
        is = cc.openInputStream();
        while (data != -1) {
            try {
                data = is.read();
                if (data == -1) { break; }
                if ('\r' == (char)data) { break; }
                else { inputBuffer.append((char)data); }
            } catch (IOException ex) {
                System.err.println(ex);    return;
            }
        }
        is.close();
        cc.close(); // Fecha a conexão serial.
makePoints(inputBuffer.toString());
    } catch (IOException ioe) {
        System.out.println("Erro: " + ioe.toString());
        mMessageItem.setText(ioe.toString());
    }
}
```

Implementação: Emulador

- DefaultColorPhone
(Wireless Toolkit)





Conclusões

- Os objetivos foram alcançados
- Conectividade
 - WEB + Celular + GPS
- Aplicações Wireless
 - Uso controlado dos recursos
- Ferramentas atenderam necessidades



Extensões

- IPA – Consultar cotas de enchentes
- Permitir formar trajetos de menor caminho
- Aviso de radares