

*Protótipo de um software para  
dispositivos móveis utilizando  
Java ME para cálculo de  
regularidade em Rally*

Acadêmico Fábio Marcelo Depiné

Orientador Francisco Adell Péricas

# *Objetivos*

- Demonstrar tecnologia J2ME
- Especificar software de cálculo de planilha de rally para dispositivos móveis
- Implementar protótipo
- Entrar com informações da planilha no celular, retornando os cálculos ao usuário

# *Roteiro*

- Java e J2ME
- Dispositivos portáteis
- Rally de regularidade
- Desenvolvimento do protótipo
- Conclusão

# *Java e J2ME*

- Objetivo inicial: produtos eletrônicos
- Sucesso com a Internet
- Plataforma Java 2: J2SE, J2EE e J2ME:
  - J2SE: versão básica para estações de trabalho
  - J2EE: aplicações robustas, soluções corporativas
  - J2ME: versão compacta para dispositivos portáteis

## *Java e J2ME*

- Arquitetura J2ME dividida em 3 camadas:
  - Configuração: variedade de dispositivos com características em comum
    - CLDC
  - Perfil: detalhes específicos do dispositivo
    - MIDP
  - Interpretador: KVM

Observação: CLDC + MIDP = MIDLET

# *Java e J2ME*

MIDLET

Perfil MIDP

Configuração CLDC

KVM

Sistema Operacional

# *Dispositivos Portáteis*

- Crescente expansão devido a evolução tecnológica
- Serviços personalizados e úteis
- Qualquer equipamento com capacidade de processamento e armazenamento pode ser adaptado para J2ME
- Motorola i85s

# *Dispositivos Portáteis*



# *Rally de Regularidade*

- Esporte de aventura e sincronismo
- Referência e tempo ideal
- Cálculos efetuados manualmente pelos competidores:
  - Tempo (min) = (distância (km) / velocidade (km/h)) \* 60
  - Erros de arredondamentos, falta de tempo

## *Rally de Regularidade*

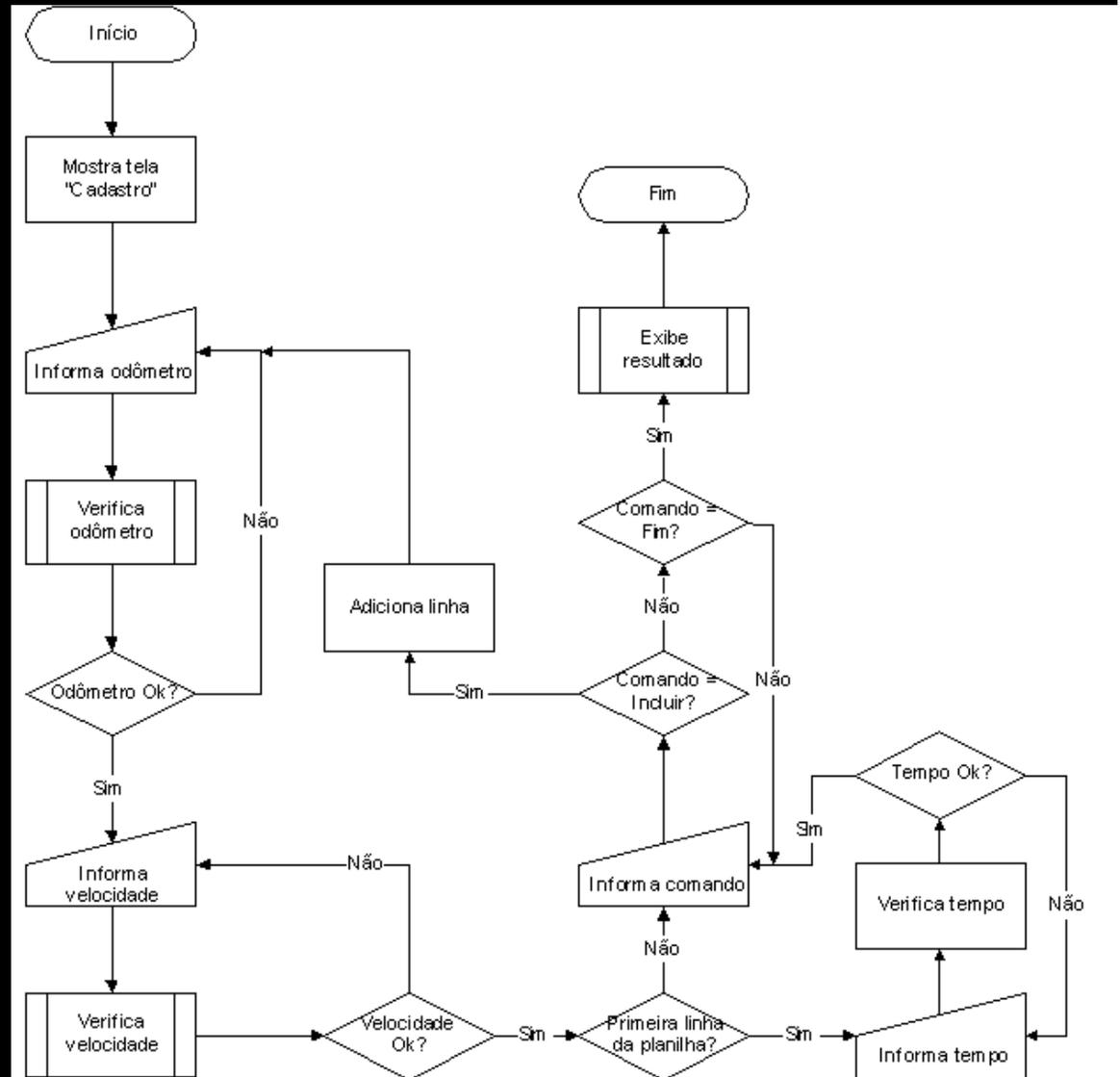
Odômetro	Referência	Velocidade	Tempo
0,00	Placa	40	00:15:00
2,02	Cruzamento		00:18:02
<u>3,52</u> 0,00	Pegue à direita	30	00:20:17
0,21	Ponto de ônibus		00:20:42
<u>0,24</u> 0,00	Deslocamento	D	00:20:46
0,78	Pegue à esquerda		00:00:00
<u>1,48</u> 0,00	Placa à direita	45	00:23:12
1,02	Cuidado: preferencial		00:24:34

# *Desenvolvimento do Protótipo*

- Requisitos:
  - Ambiente J2ME com CLDC + MIDP
  - Armazenará os cálculos em memória
  - Irá obedecer a lógica do livro de bordo
  - Tempo ideal será informado em horas, minutos e segundos
  - O tempo ideal calculado pelo protótipo poderá ser alterado
  - O usuário poderá consultar as informações

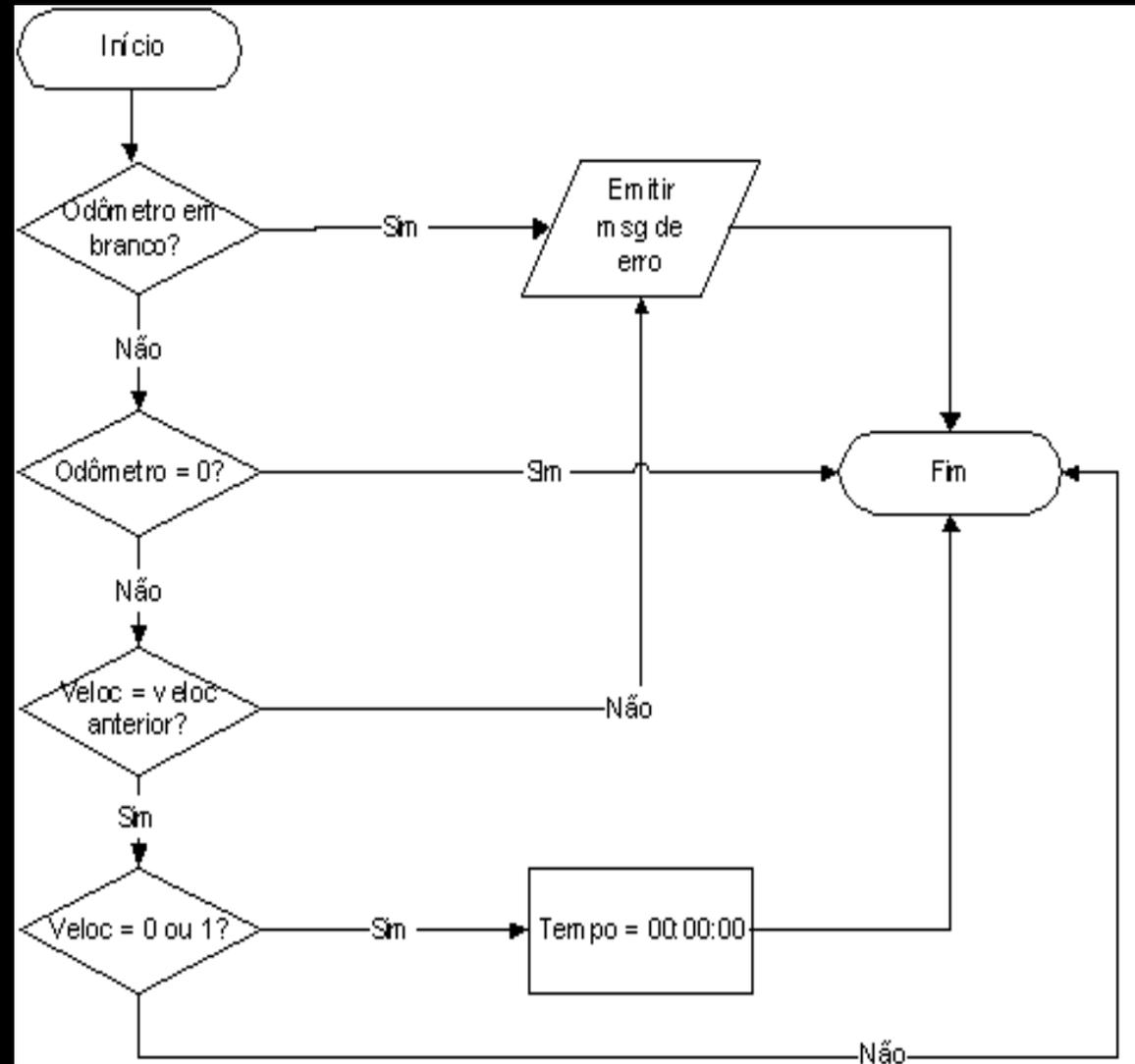
# Desenvolvimento do Protótipo

- Especificação genérica do protótipo:



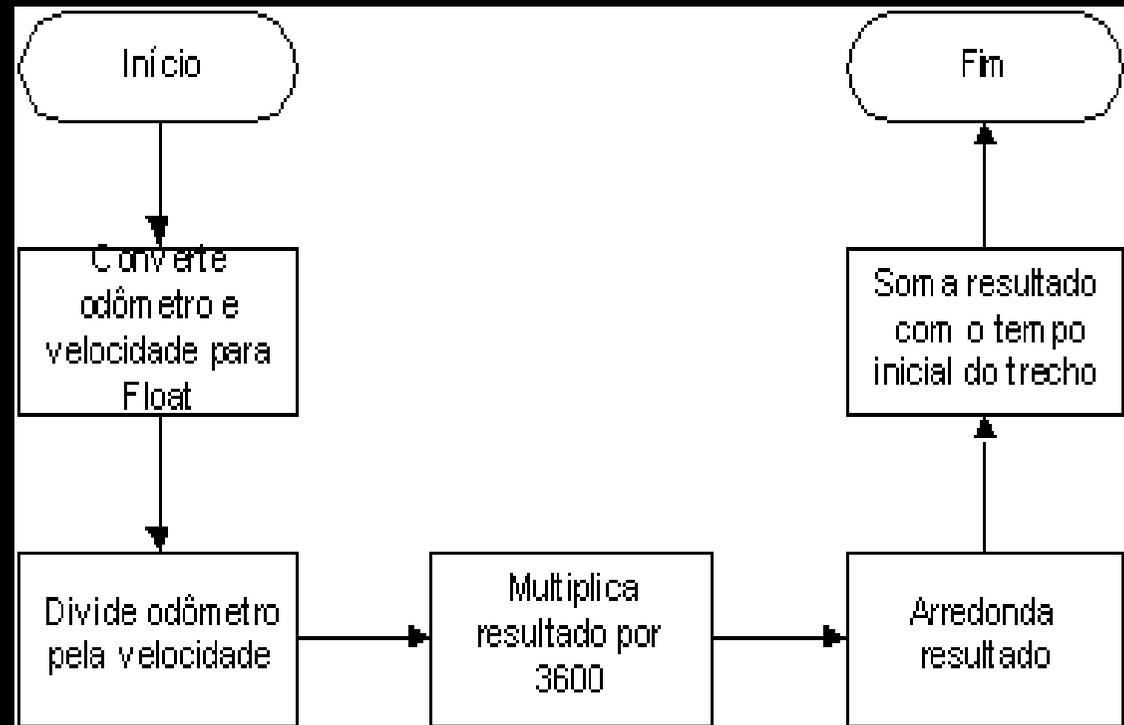
# Desenvolvimento do Protótipo

- Especificação do processo Verifica Velocidade:



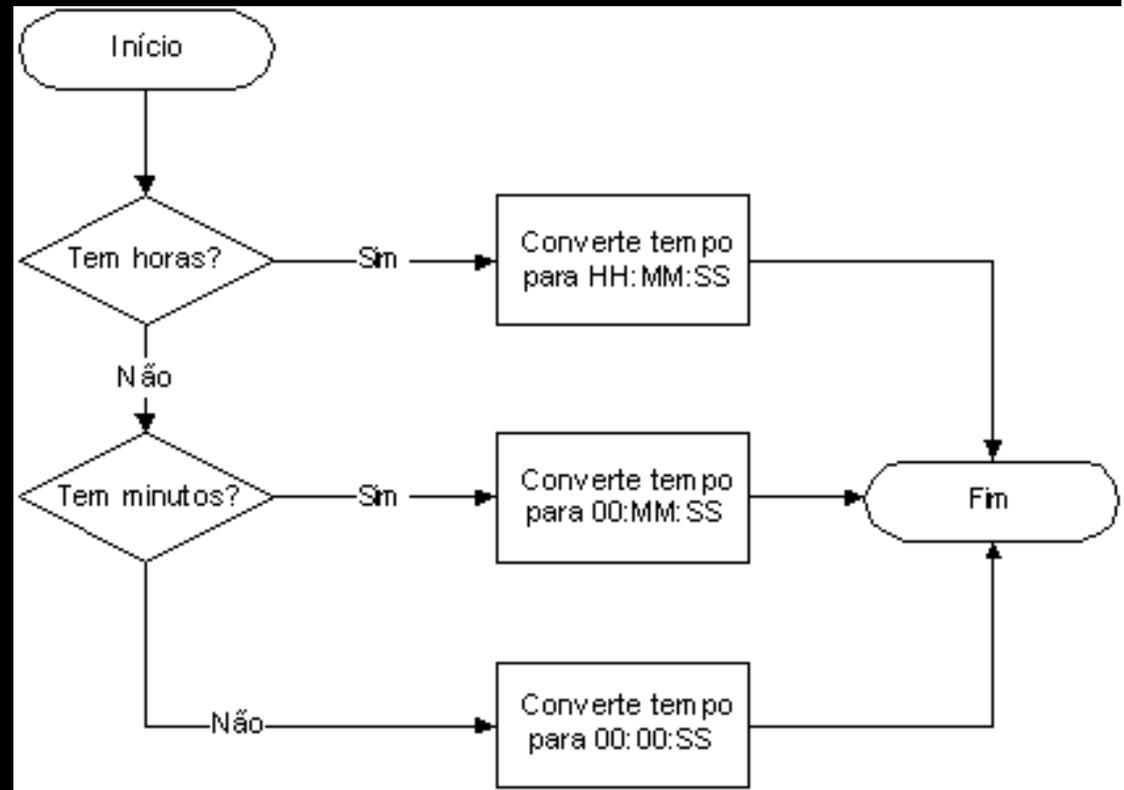
# *Desenvolvimento do Protótipo*

- Especificação do processo  
Calcula Tempo Total:



# *Desenvolvimento do Protótipo*

- Especificação do processo Mostra Tempo Formatado:



# *Desenvolvimento do Protótipo*

- Implementação do processo Calcula Tempo Total:

```
public int CalculaTempoTotal(String aOdometro, String aVelocidade, int
aTempoInicialDoTrecho)
{
    Float xFloat = new Float(Integer.parseInt(aOdometro),-2);
    xFloat = xFloat.Div(new Float(Integer.parseInt(aVelocidade),0));
    xFloat = xFloat.Mul(new Float(3600));

    String vlrArredondado = xFloat.RoundFloatToString(xFloat);
    int xTempoDaLinhaAtual = Integer.parseInt(vlrArredondado);

    return (xTempoDaLinhaAtual + aTempoInicialDoTrecho);
}
```

# *Desenvolvimento do Protótipo*

- Implementação da operação de subtração da classe Float:

```
/** *****  
// Subtração  
/** *****  
  
public Float Sub(Float value)  
{  
    if (value.Equal(ZERO))  
        return new Float(m_Val, m_E);  
    return Add(new Float(-value.m_Val, value.m_E));  
}
```

# *Desenvolvimento do Protótipo*

- Conclusão:
  - Estudo da tecnologia J2ME
  - Desenvolvimento de uma classe para emular cálculos matemáticos com ponto flutuante
  - Protótipo capaz de calcular e armazenar um livro de bordo completo

# *Desenvolvimento do Protótipo*

- Limitações:
  - Imprescindível ambiente J2ME com configuração CLDC e perfil MIDP
  - Não permite armazenar informações fisicamente
  - Após cadastro, informações não podem ser alteradas
  - Visualização limitada

# *Desenvolvimento do Protótipo*

- Extensões:
  - Implementar alteração das informações após cadastro
  - Implementar armazenamento físico
  - Comunicação de dados (cliente-servidor)