

# Protótipo de Linígrafo Micro-controlado

André Zimmermann  
andrezimmermann@operamail.com

Orientador: Miguel Alexandre Wisintainer



# Roteiro

- Introdução
- Objetivos
- Fundamentação Teórica
- Especificação
- Desenvolvimento
- Operacionalidade da Implementação
- Resultados e Discussão
- Conclusões
- Extensões
- Demonstração



# Introdução

- Utilização da hidrografia em projetos.
- Linígrafos comerciais.
  - Medição com contato.
  - Deficiência em soluções com telemetria.



# Objetivos

- Montagem de um protótipo de linígrafo micro controlado:
  - Utilização de sonar e módulo GSM.
  - Implementação de *web-service* para armazenamento e exportação das medições.
  - Efetuar a comunicação entre o protótipo e o *web-service*.
  - Permitir o cadastramento de curva-chaves para exportação da vazão.



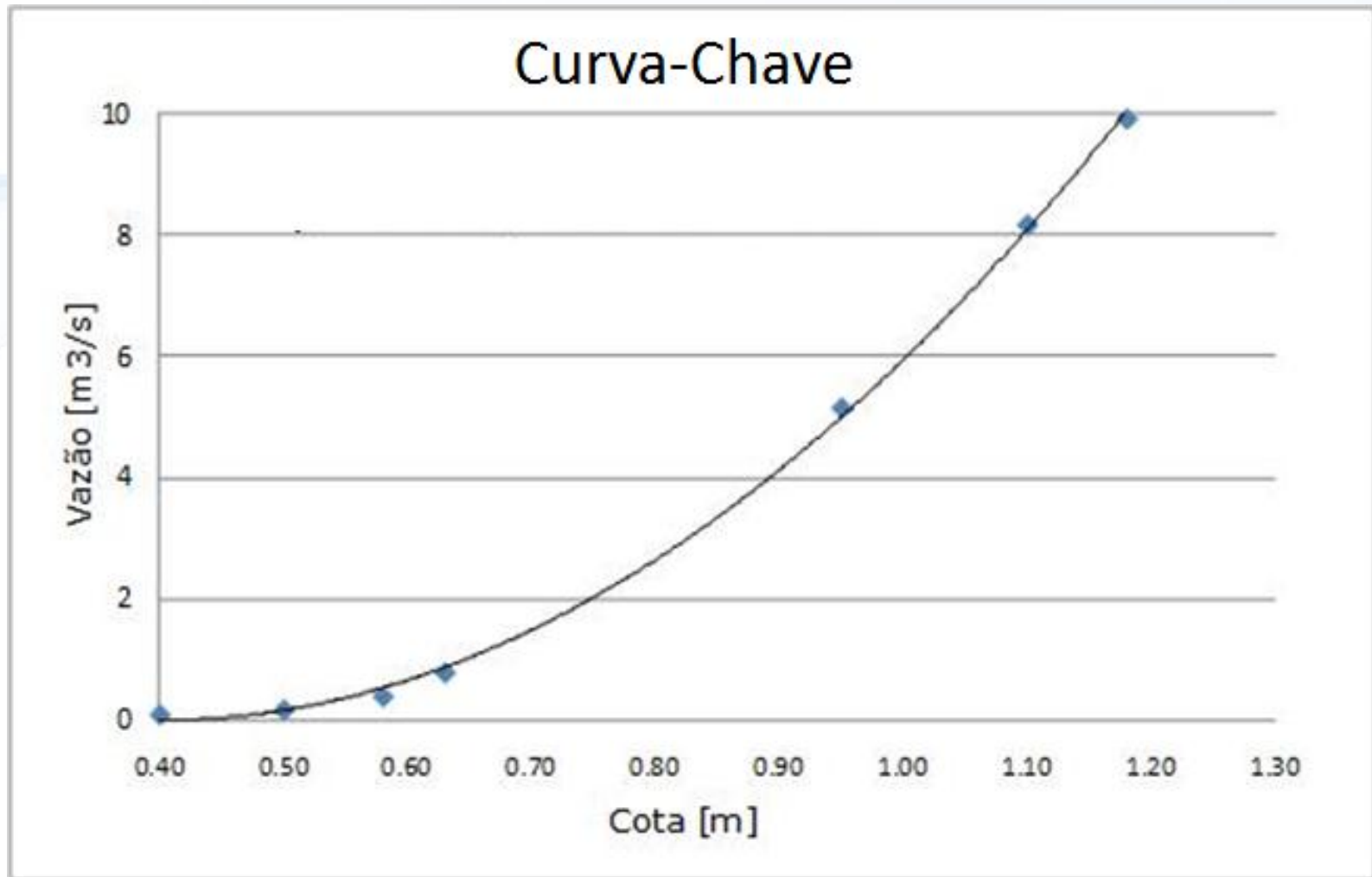
# Fundamentação Teórica

- Principais conceitos:

- Linígrafos.
- Curva-chave.
- GSM, GPRS e TCP/IP.
- Arquitetura *Restful*.
- *XML Schema*.
- JAXB.



# Fundamentação Teórica



# Trabalhos correlatos

<b>Trabalho</b>	<b>Rabello, Cruvinel e Denardin</b>	<b>Thalimedes</b>	<b>RLS</b>	<b>WL-15</b>
<b>Tipo de Medição</b>	Boia - micro controlado	Boia - micro controlado	Sem contato por radar	Sensor de pressão
<b>Faixa útil</b>	0cm até 2,55 cm	0cm até 60m	80cm até 35m	0cm até 6.35m
<b>Quantidade de registros</b>	Não especificado	30.000	0	24.400
<b>Telemetria</b>	-	-	-	-

# Especificação do Protótipo

- Requisitos funcionais:

- efetuar medições da distancia do protótipo até superfície do corpo d'-água
- enviar as medições efetuadas ao web-service e apagar estas medições caso enviou com sucesso.
- permitir a configuração do intervalo de tempo entre as medições e quantidade armazenadas para o início do envio das medições

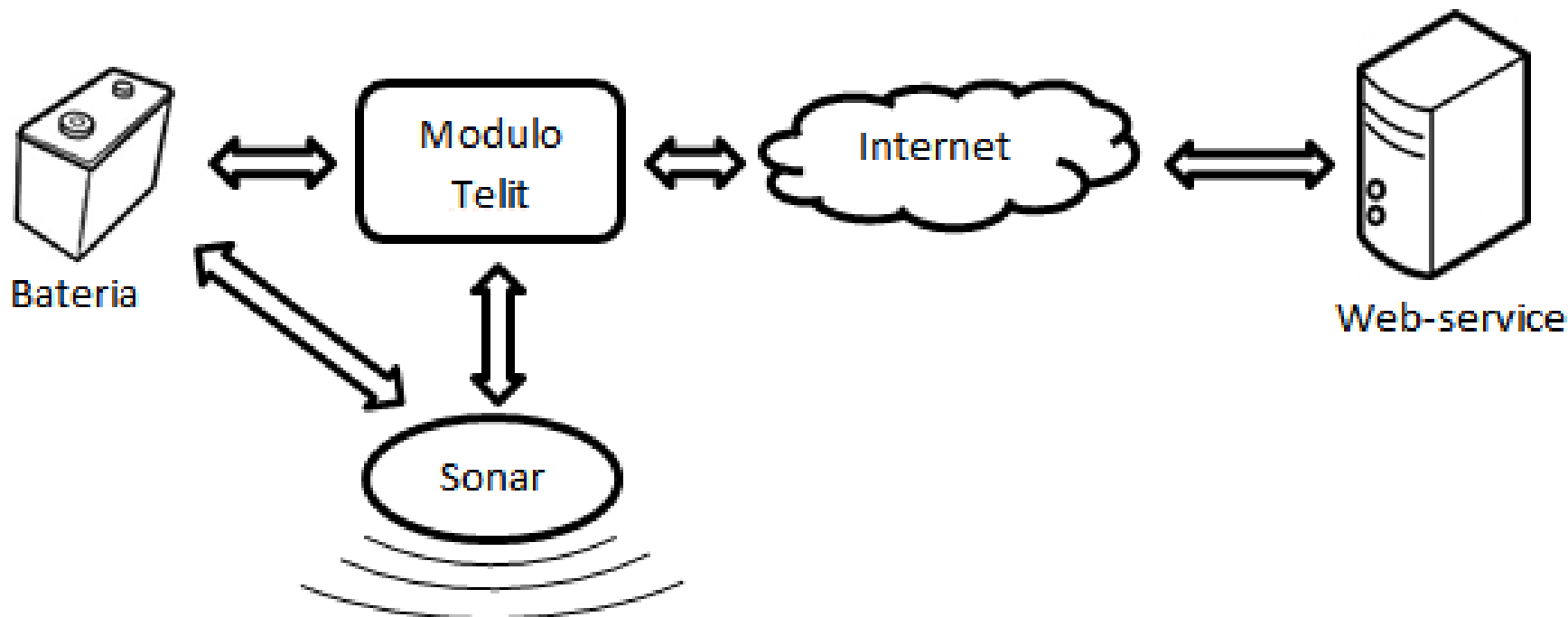
- Requisitos não funcionais:

- deve usar sonar para efetuar a medição da distancia.

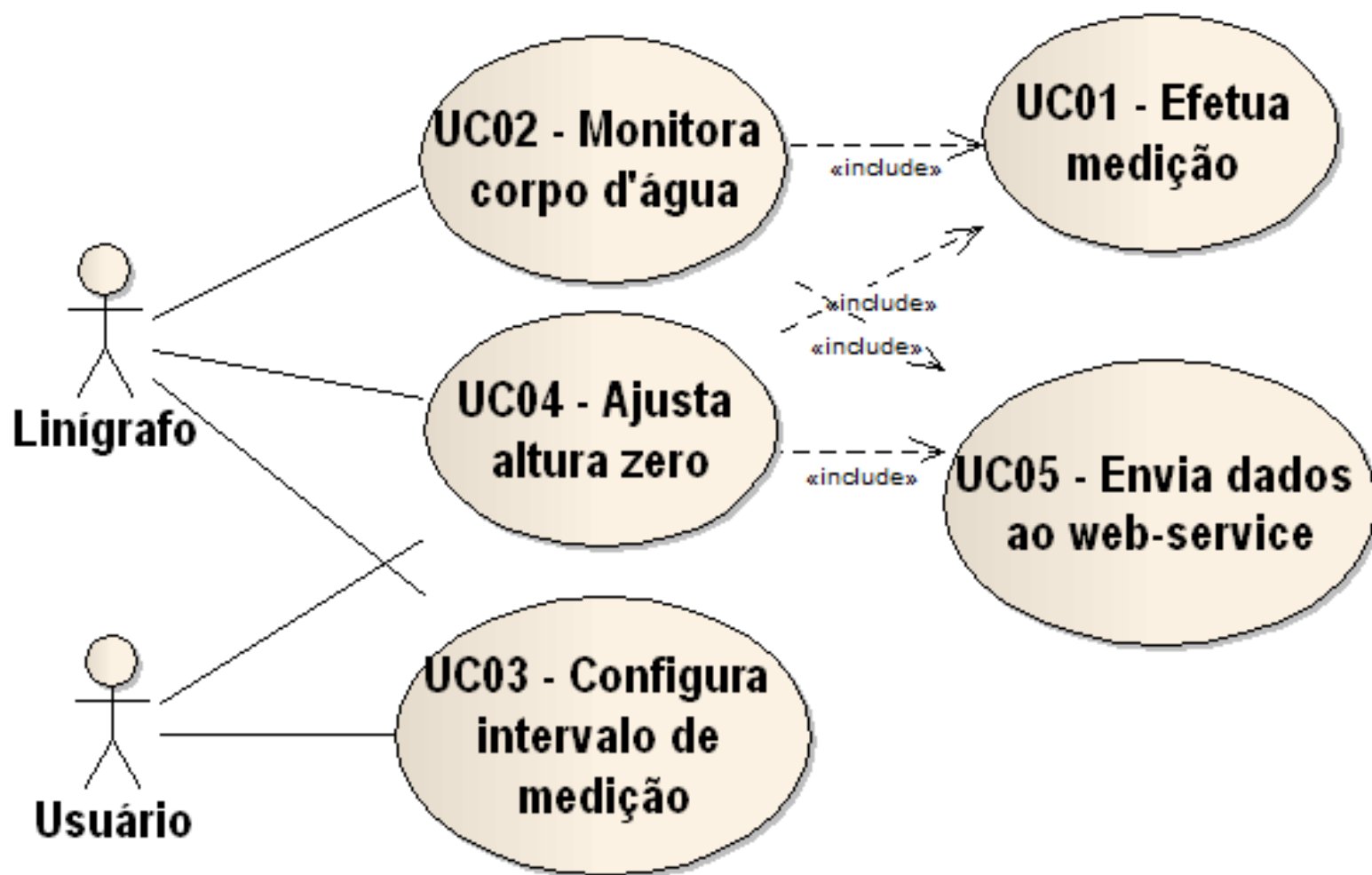




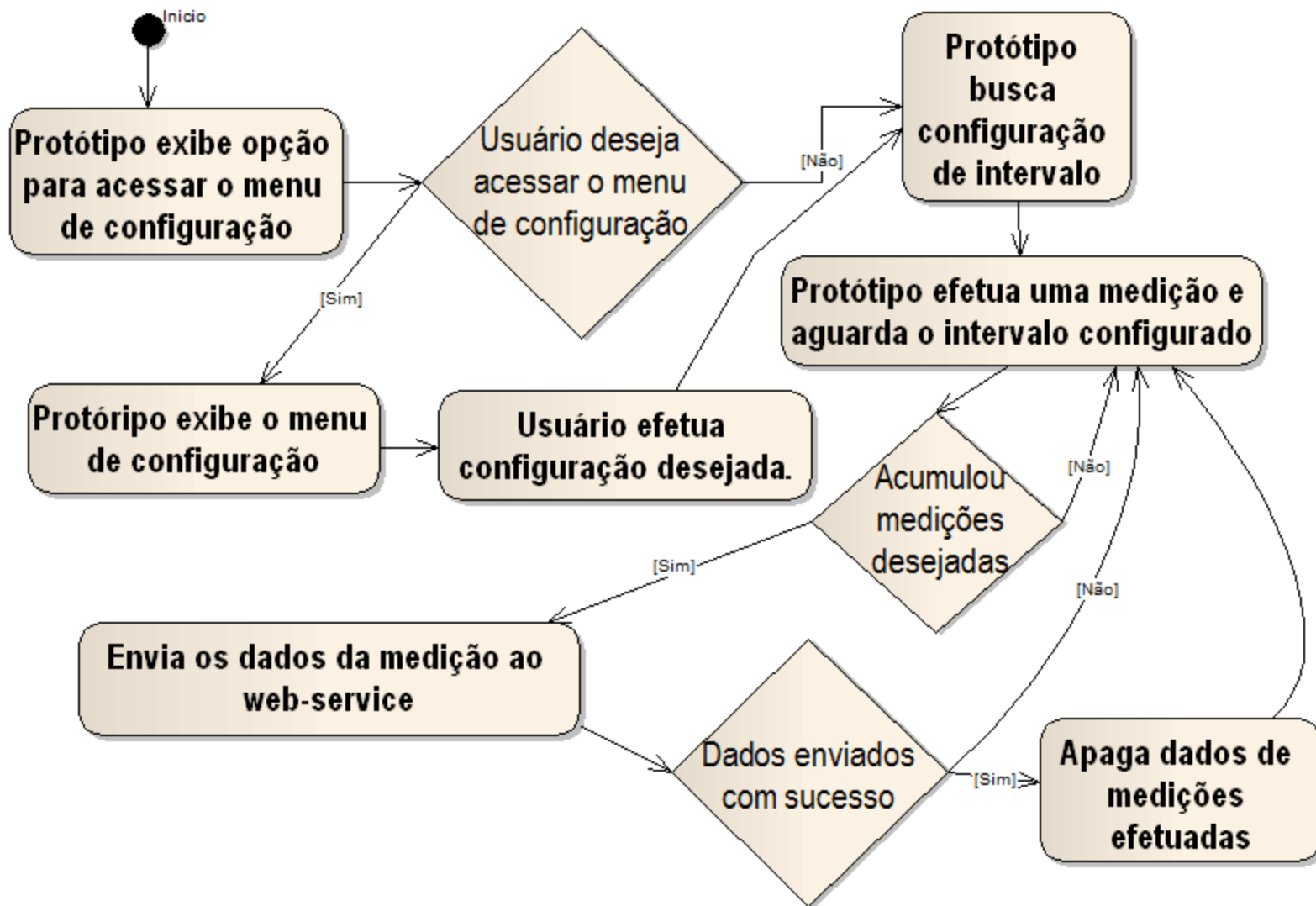
# Especificação do Protótipo



# Especificação do Protótipo



# Especificação do Protótipo



# Especificação do *Web-service*

## •Requisitos Funcionais

- permitir o cadastro de linígrafos, com identificador único, situação e local de instalação.
- receber as informações de medições enviadas pelos linígrafos.
- exportar, em formato XML, as informações de medições enviadas pelos linígrafos, referentes a determinado um período.
- permitir o cadastro de várias curvas chaves para possibilitar o cálculo da vazão de diferentes corpos d'água.



# Especificação do *Web-service*

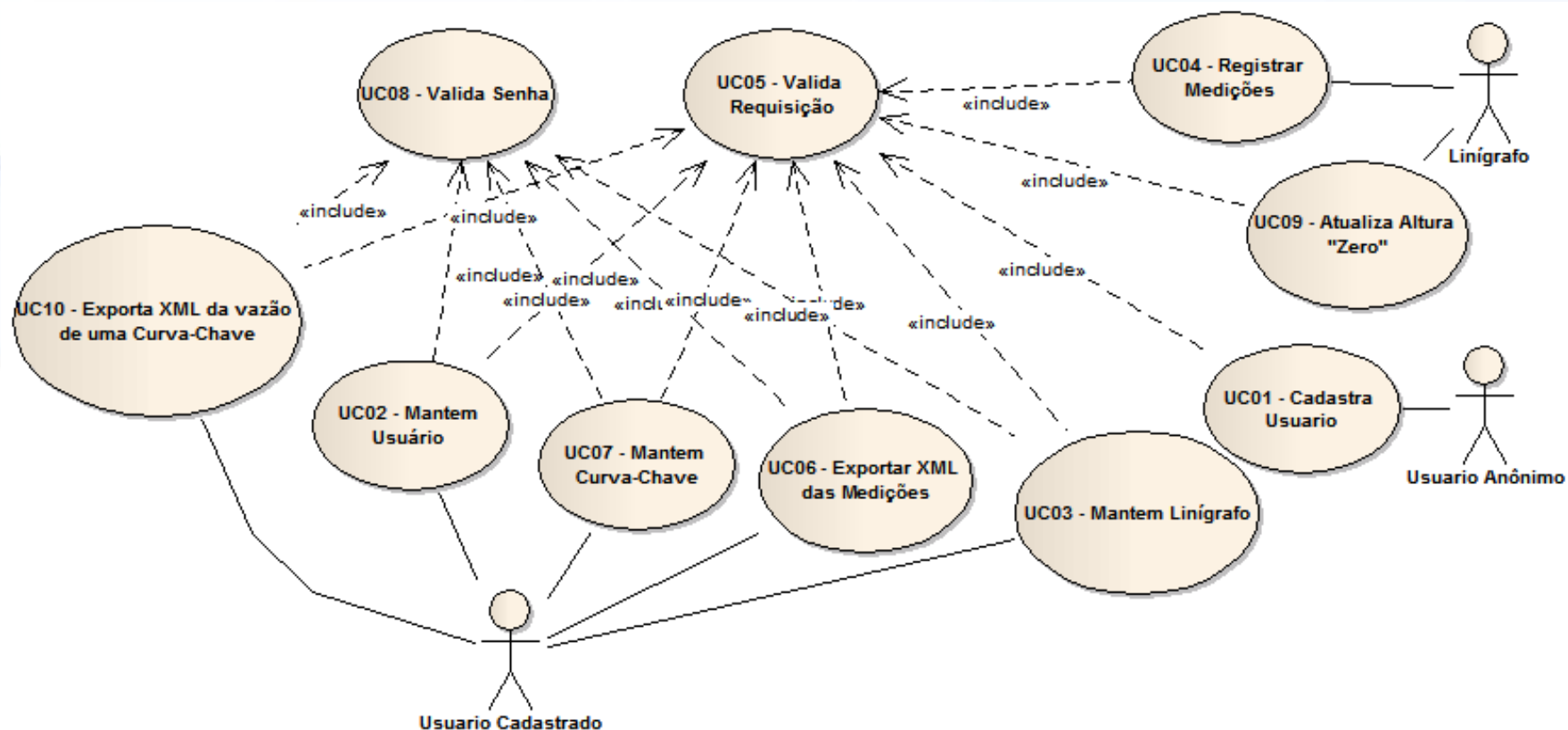


Figura 15, página 35

# Especificação do *Web-service*

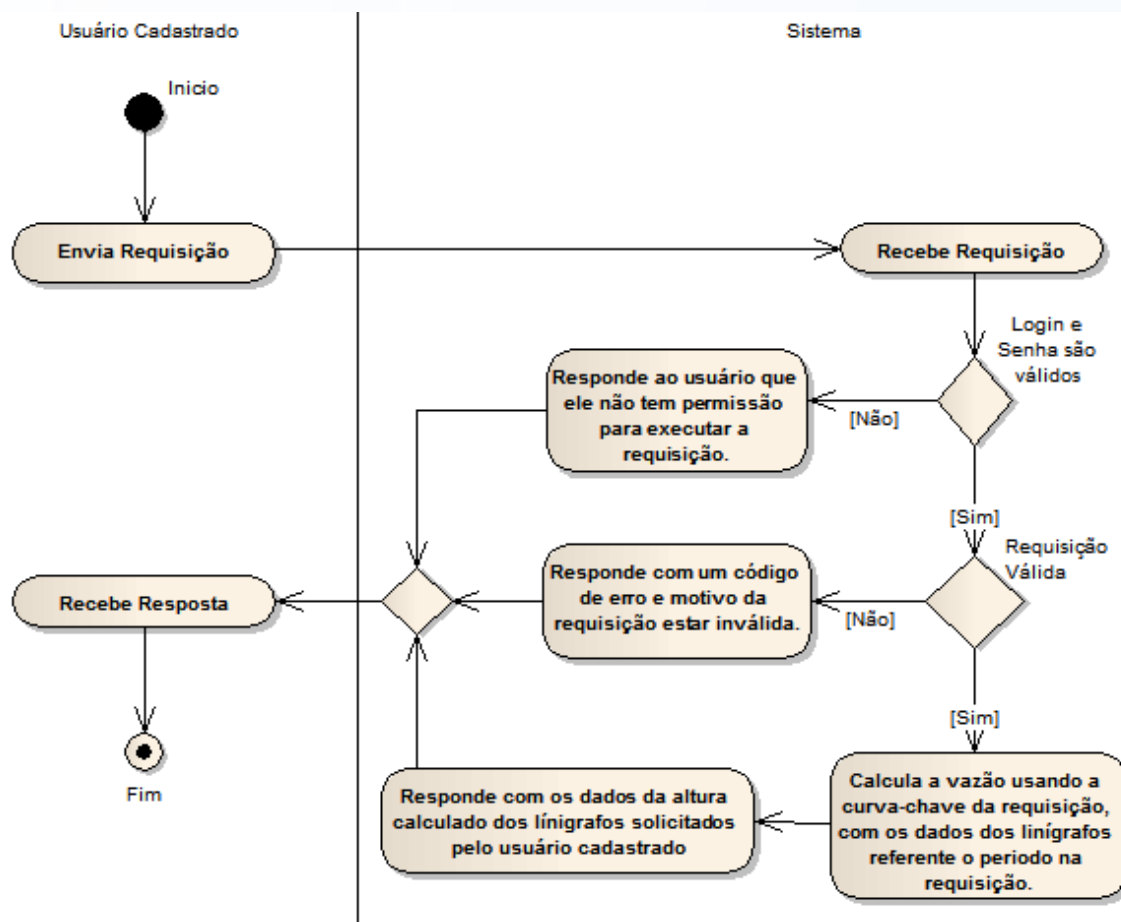


Figura 24, página 45

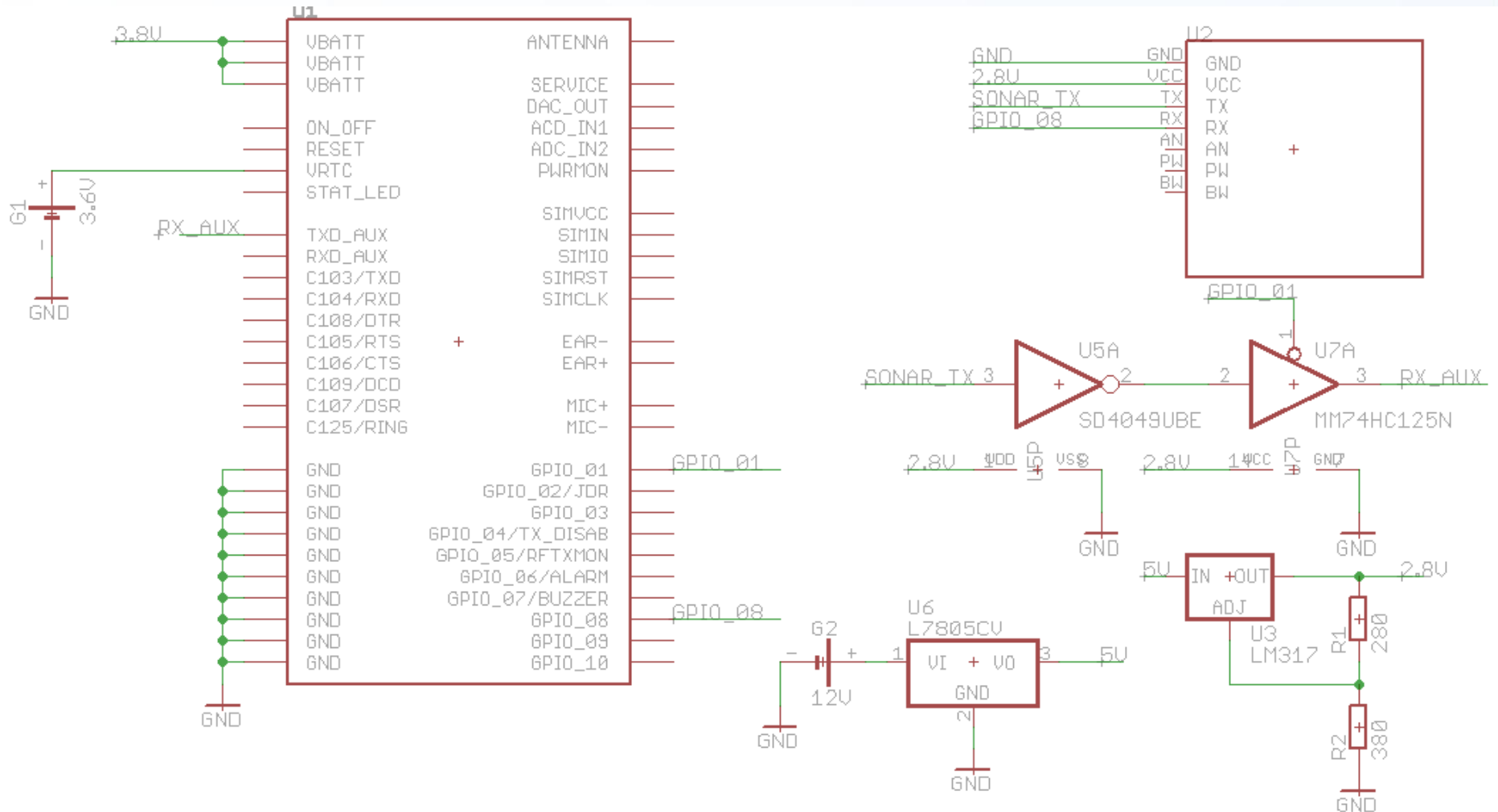
# Desenvolvimento

## •Protótipo

- Módulo Telit GE865-QUAD
- Kit de desenvolvimento da Sparkfun
- Sonar LV-MaxSonar da MaxBotix



# Desenvolvimento





# Desenvolvimento

```
def adquireLeitura(self):
    print "Buscando medição"
    GPIO.setIOvalue(8, 1)#Ao jogar alto, inicia aquisição de dados
    GPIO.setIOvalue(1, 0)#Ao jogar baixo, ativa o tri-state

    MOD.sleep(2)#Tempo necessario para a calibragem do sonar
    #Tudo certo por aqui, posso começar a buscar o valor da medição
    SER2.read()#Tira o lixo da serial
    MOD.sleep(10)#Espera 1s
    GPIO.setIOvalue(8, 0)
    GPIO.setIOvalue(1, 1)
    readedString = SER2.read()#Le os valores
    readedString = readedString.replace(chr(13), '')
    print readedString

    listaMedicoes = readedString.split("R")
    del readedString
    valorMedio = 0
    numeroErros = 0;
    for i in range(len(listaMedicoes)):
        if (len(listaMedicoes[i]) == 3):# Se o número tem 3 dígitos
            while(listaMedicoes[i].startswith("0")):# enquanto tiver zeros na frente
                listaMedicoes[i] = listaMedicoes[i][1:]#Remove o primeiro zero a esquerda
            if(listaMedicoes[i] == ""):
                numeroErros = numeroErros + 1
                continue
            try:
                valorMedio = valorMedio + int(listaMedicoes[i])#Converte aqui para inteiros
            except ValueError:
                numeroErros = numeroErros + 1
        else:
            numeroErros = numeroErros + 1

    ret = valorMedio / (len(listaMedicoes) - numeroErros)

    print "Valor adquirido:" + str(ret) + " inches"
    return ret
```



# Desenvolvimento

- *Web-service*

- Utilização da *framework Restlet*.
- EJB 3.0 e a padrão de projeto DAO
- JAXB para a validação das requisições.



# Desenvolvimento

```
@Get
public String getResource() {
    String xmlValue = RestletUtil.getXmlValue(getQuery()); // Inicia o processamento da requisição
    StringReader stringReader = new StringReader(xmlValue);
    MedicoesSimplesFetchRequest fetchData = validator.unmarshall(// Valida a requisição
        stringReader, MedicoesSimplesFetchRequest.class);
    if (fetchData == null) { // É válida?
        StringWriter xml = new StringWriter(); // Não é válida
        JAXB.marshal(validator.getSchemaError(), xml);
        return xml.toString(); // Responde com a mensagem de erro
    } // else
    Date inicial = fetchData.getDataInicial().toGregorianCalendar().getTime(); // Pega dados enviados ao servidor
    Date dtFinal = fetchData.getDataFinal().toGregorianCalendar().getTime();
    Long[] array = fetchData.getLinigrafos().getId().toArray(new Long[0]);
    // Utiliza o EJB do Linigrafo para pegar as medições
    List<LinigrafoProcessado> buscaMedicaoSimples = getLinigrafoEJB().buscaMedicaoSimples(inicial, dtFinal, array);
    ObjectFactory of = new ObjectFactory(); // Usa JAXB para gerar a resposta
    MedicoesSimplesFetchResponse resp = of.createMedicoesSimplesFetchResponse();
    for (LinigrafoProcessado lp : buscaMedicaoSimples) {
        LinigrafoMedicao lm = of.createLinigrafoMedicao();
        lm.setId(lp.getId());
        for (RegistroProcessado rp : lp.getListaRegistros()) {
            DadosMedicao createDados = of.createDadosMedicao();
            createDados.setData(XMLGregorianCalendarConverter.asXMLGregorianCalendar(rp.getDataRegistro()));
            createDados.setValor(rp.getValor());
            lm.getDadosMedicao().add(createDados);
        }
        resp.getLinigrafoMedicao().add(lm);
    }
    StringWriter xml = new StringWriter();
    JAXB.marshal(resp, xml);
    return xml.toString(); // Responde ao usuário
}
```

# Desenvolvimento

```
@Override
public List<LinigrafoProcessado> buscaMedicaoSimples(Date inicial,
    Date dtFinal, Long... linigrafos) {

    List<LinigrafoProcessado> listaRetorno = new ArrayList<LinigrafoProcessado>();
    // Busca para cada linigrafo
    for (Long integer : linigrafos) {
        LinigrafoProcessado lP = new LinigrafoProcessado();

        lP.setId(integer);
        List<RegistroMedicao> dados = getIDao().getDados(integer, inicial,
            dtFinal);

        for (RegistroMedicao registroMedicao : dados) {
            RegistroProcessado registroProcessado = new RegistroProcessado();
            // Altura do linigrafo em Relação a Água h0
            // Altura da água no momento da Instalação hI
            // Distancia do linigrafo (Medicção) hM
            // Altura atual é a (h0 - hM) + hI

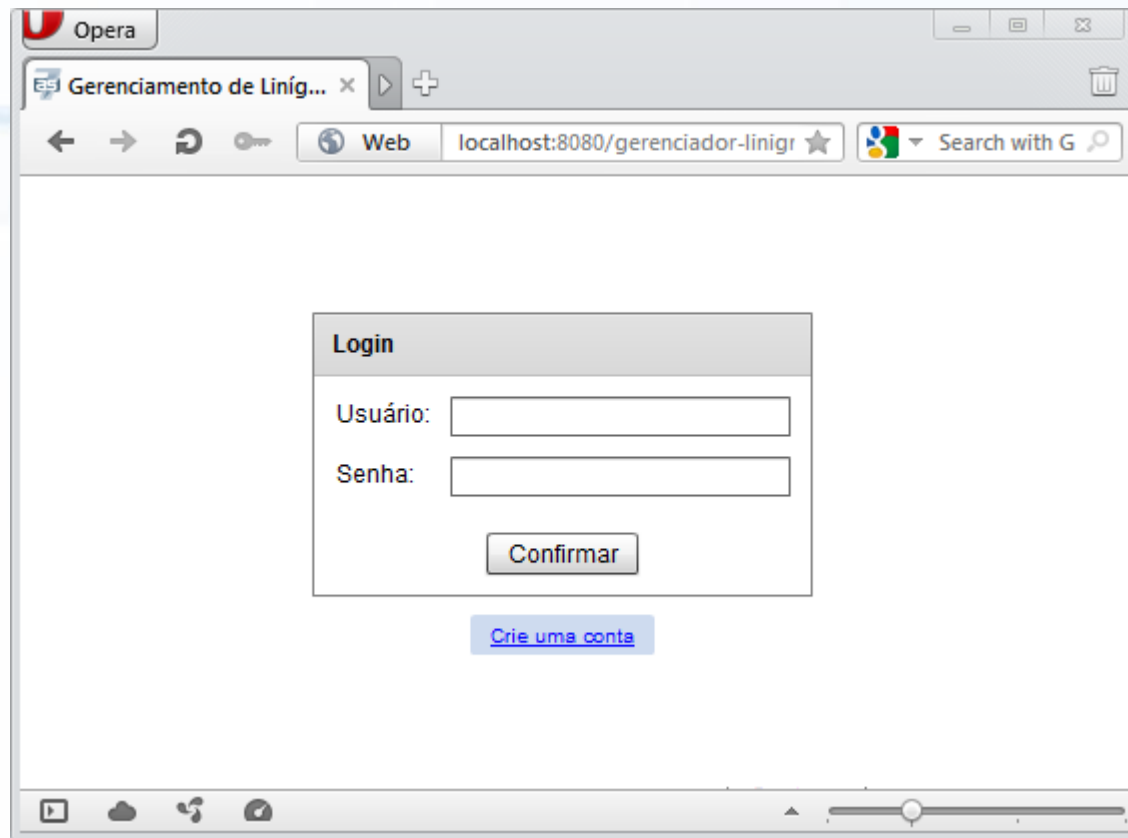
            double altura = (registroMedicao.getAlturaZero() - registroMedicao
                .getAlturaMedicao())
                + registroMedicao.getAlturaInicial();

            altura = round(altura, 2);

            registroProcessado.setValor(altura);
            registroProcessado.setDataRegistro(registroMedicao
                .getDataMedicao());
            lP.getListaRegistros().add(registroProcessado);
        }
        listaRetorno.add(lP);
    }
    return listaRetorno;
}
```



# Operacionalidade da Implementação



# Operacionalidade da Implementação

Opera

Gerenciamento de Linígra... x

Web localhost:8080/gerenciador-li

Principal Novo Linígrafo

ID: 12345

Descrição: Linígrafo de Testes

Endereço: Furb

Altura Zero: 5

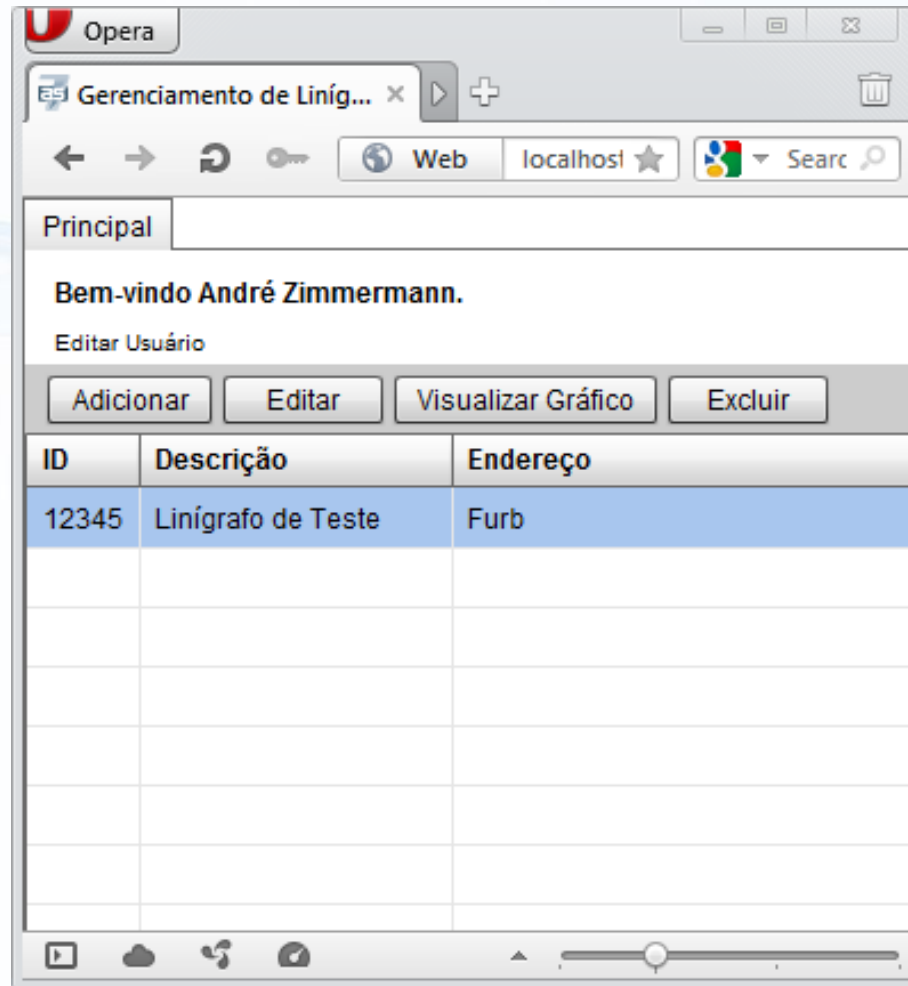
Altura Inicial: 10

DDD: 47 DDI: 55

Telefone: 33335040

Salvar Excluir

# Operacionalidade da Implementação

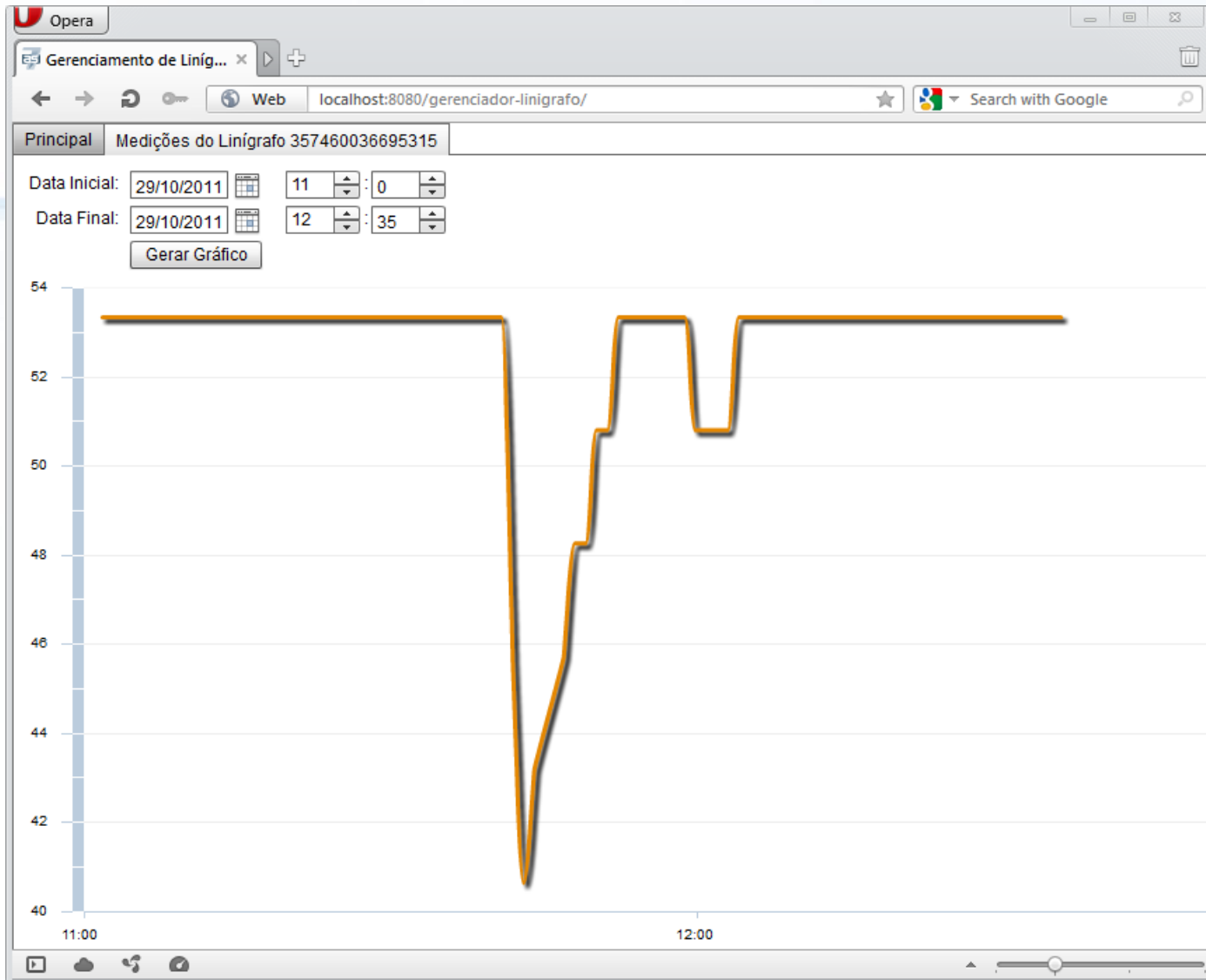


The screenshot shows a web browser window with the following elements:

- Browser title: Opera
- Tab: Gerenciamento de Linígrafo
- Address bar: localhost
- Page content:
  - Principal
  - Bem-vindo André Zimmermann.
  - Editar Usuário
  - Buttons: Adicionar, Editar, Visualizar Gráfico, Excluir
  - Table with columns: ID, Descrição, Endereço

ID	Descrição	Endereço
12345	Linígrafo de Teste	Furb

# Operacionalidade da Implementação





# Resultados e Discussão

- Precisão nas medições.
- Problemas inerentes a utilização da rede GSM.
- Limitações de hardware e pouca documentação.
- *Web-service* é de fácil consumo.
- Problemas em relação à segurança do *web-service*.



# Resultados

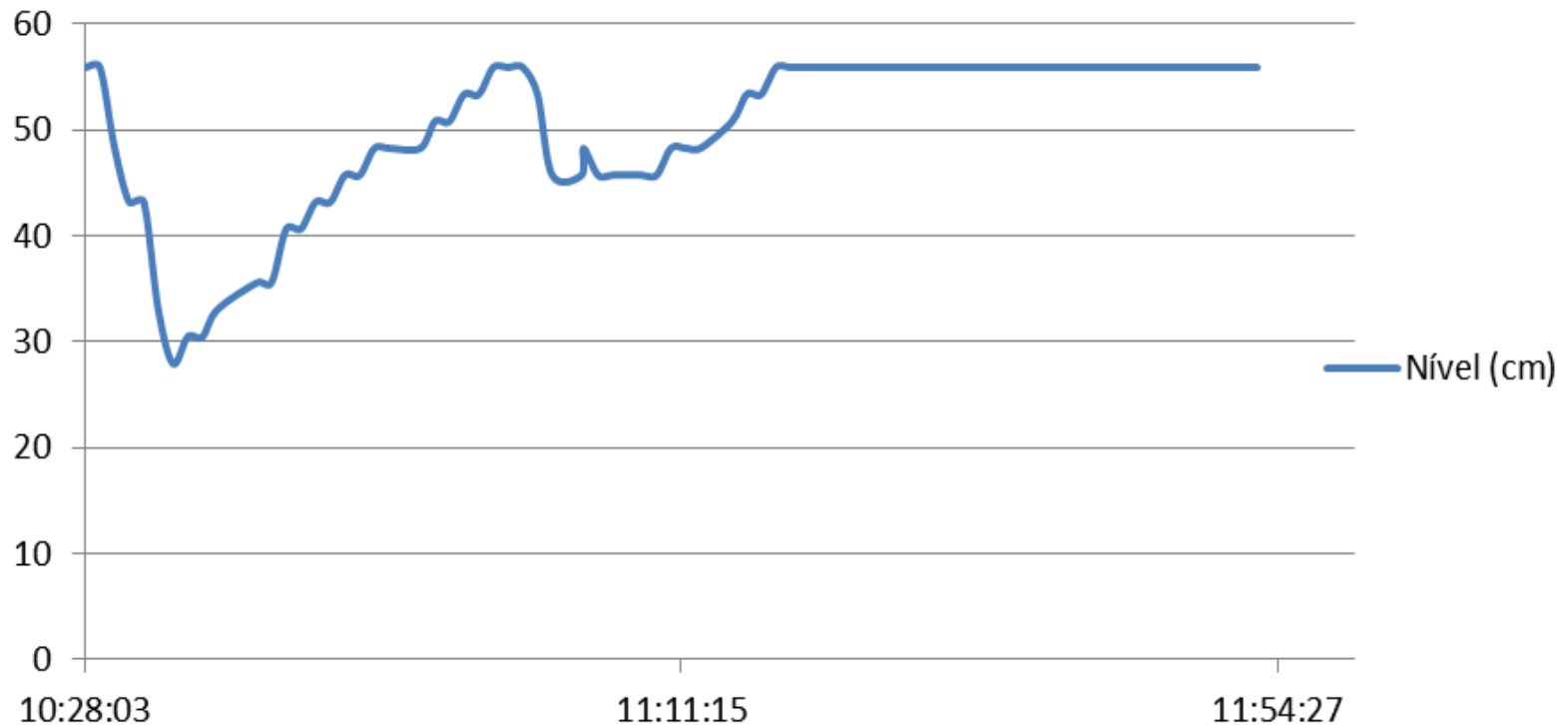


# Resultados



# Resultados

**Gráfico do gerado a partir do documento XML**



# Trabalhos correlatos

<b>Trabalho</b>	<b>Rabello, Cruvinel e Denardin</b>	<b>Thalimedes</b>	<b>RLS</b>	<b>WL-15</b>	<b>Protótipo</b>
<b>Tipo de Medição</b>	Boia - micro controlado	Boia - micro controlado	Sem contato por radar	Sensor de pressão	Sem contato por sonar
<b>Faixa útil</b>	0cm até 2,55 cm	0cm até 60m	80,00 cm até 35 metros	0cm até 6.35m	15,24 cm até 6.45m
<b>Quantidade de registros</b>	Não especificado	30.000	0	24.400	3.000
<b>Telemetria</b>	-	-	-	-	GSM

# Conclusões

- Comercialização.
- Uso em pequenas bacias.



# Extensões

- Melhorar o sinal GSM ou utilizar outro meio para a comunicação.
- Utilizar sonar com maior faixa útil.
- Efetuar a configuração do protótipo através do *web-service*.
- Implementar o cálculo da curva-chave através de métodos iterativos.
- Utilizar um canal seguro para a permuta dos dados.
- Gerar certificados digitais com dupla autenticação para garantir a identidade do protótipo.



# Demonstração

OK

Buscando medição

R026R

Valor adquirido: 26 inches

Enviando AT+CCLK?

Recebeu

+CCLK: "11/12/11,17:34:06"

OK

Buscando medição

R026R

Valor adquirido: 26 inches

Enviando AT+CCLK?

Recebeu

+CCLK: "11/12/11,17:34:20"

OK

Enviando AT+CGATT?

Recebeu

+CGATT: 0

OK

Enviando AT+CGATT=1

Recebeu

OK

Enviando AT#GPRS=1

Recebeu

+IP: 189.116.204.111





Obrigado.

