

Projeto de Graduação



15/12/2015

## **DESENVOLVIMENTO DE CASOS EM LUA PARA A PLATAFORMA SKYWAVE IDP-780**

Rodrigo Trompieri Ribeiro



[www.ele.puc-rio.br](http://www.ele.puc-rio.br)

Projeto de Graduação



## **DESENVOLVIMENTO DE CASOS EM LUA PARA A PLATAFORMA SKYWAVE IDP-780**

**Aluno(s): Rodrigo Trompieri Ribeiro**

**Orientador(es): Marco Antonio Grivet M. Maia**

Trabalho apresentado com requisito parcial à conclusão do curso de Engenharia Elétrica na Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brasil.

## Resumo

O dispositivo Skywave IDP-780 é um terminal integrado de comunicação via satélite e celular, que proporciona conectividade em diversas aplicações tais como gestão de frotas com ativos que se deslocam dentro e para fora da cobertura celular.

O menor custo de transmissão de dados em áreas com cobertura celular e comunicações confiáveis e ininterruptas via satélite quando a rede celular não está disponível, tornam esse aparelho muito atrativo para empresas que fazem o monitoramento de frotas.[6]

O terminal IDP-780 é inteiramente programável em LUA e rico em recursos. É o único terminal em sua classe com uma arquitetura flexível que suporta o desenvolvimento de aplicações personalizadas para soluções mais complexas. Nesse sentido, esse trabalho visa mostrar o sistema embarcado nesse terminal e principalmente o seu funcionamento para três casos em LUA.

**Palavras-chave: Monitoramento; Satélite; Celular; LUA;**

## Case Developing LUA for the Skywave plataform IDP-780

### Abstract

Skywave The IDP-780 device is an integrated communication terminal and satellite phone, which provides connectivity in various applications such as fleet management with active moving in and out of cellular coverage.

The lower cost of data transmission in areas with mobile phone coverage and reliable and seamless communications satellite when the cellular network is not available, make this very attractive device for companies that make monitoring fleets.

The IDP-780 terminal is fully programmable in LUA and rich in resources. It is the only terminal in its class with a flexible architecture that supports the development of custom solutions for complex applications. In this sense, this work aims to show the embedded system that terminal and especially its operation for three cases in LUA.

**Keywords: Monitoring; Satellite; Cellular; LUA;**

## Sumário

<b>1.</b>	Introdução.....	6
1.1	Motivação.....	6
1.2	Sistema.....	6
1.2.1	Traffic Channel.....	7
1.2.2	Global Bulletin Board.....	7
<b>2.</b>	IDP-780.....	8
2.1	Programação.....	8
2.2	Arquitetura de software.....	8
<b>3.</b>	Serviços.....	10
<b>4.</b>	Propriedades.....	11
<b>5.</b>	Estrutura de Mensagens.....	12
<b>6.</b>	Ferramentas de desenvolvimento.....	13
<b>7.</b>	Introdução a linguagem LUA.....	16
<b>8.</b>	Cases em LUA.....	17
8.1.	Envio de mensagem.....	17
8.2.	Monitoramento de sensores.....	18
8.3.	Utilização de cercas eletrônicas.....	19
<b>9.</b>	Resultados.....	23
<b>10.</b>	Conclusão.....	28
<b>11.</b>	Referências.....	29

## 1. Introdução

### 1.1 Motivação

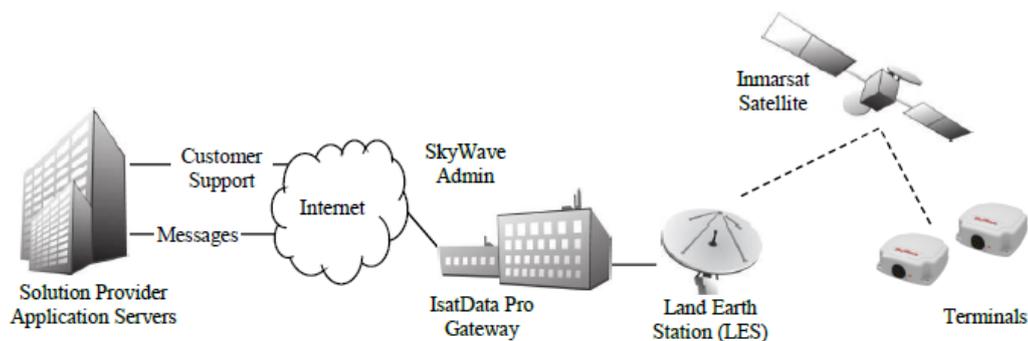
Em lugares remotos e de difícil acesso, como por exemplo minas e jazidas, onde os meios de comunicação não são confiáveis, existe uma grande preocupação de monitorar e localizar os equipamentos utilizados na mineração, pois o seu custo é elevado. Dessa forma, o rastreamento utilizando o equipamento IDP-780 da Skywave é uma excelente solução, já que os operadores de minas podem facilmente coletar e transmitir os dados a respeito da localização e de sensores em instalações remotas.

Além disso, esse dispositivo pode ser usado no rastreamento marítimo e no monitoramento da pesca comercial por meio de cercas eletrônicas, que avisam ao controlador se o barco pesqueiro está ou não em uma área de reserva ambiental.

Outra aplicação do IDP-780 é o gerenciamento de caminhões e monitoramento dos motoristas. Em virtude da grande insegurança que existe nas estradas, o IDP-780 apresenta-se como uma proteção para a integridade do motorista além de determinar que a carga seja transportada de maneira correta e segura.

### 1.2 Sistema

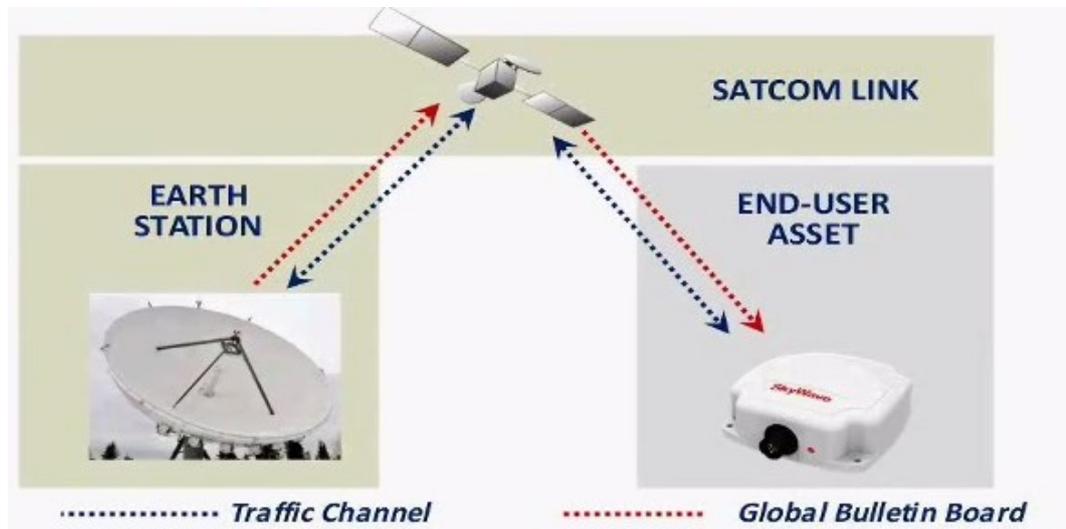
O sistema utiliza-se de um envio versátil de mensagens. Para isso, terminais de comunicação via satélite utilizam o serviço IsatData Pro para fornecer serviços de mensagens bidirecionais entre o terminal e os serviços de satélite gateway via Inmarsat, IsatData Pro ou outros.



**Figura 1.1 – exemplificação do sistema**

A Inmarsat possui e opera 11 satélites em órbita geoestacionária a 35.786 km acima da Terra, controlados a partir da sua sede em Londres através de estações terrenas distribuídas em todo o mundo. O IsatData Pro usa três satélites da constelação de satélite do Inmarsat e três estações terrenas. Essas estações estão localizadas no Canadá, na Nova Zelândia e na Holanda. O IsatData Pro faz uso também de dois gateways localizados em Ottawa na sede da Skywave (gateway principal) e outro em Brunei cujo papel é servir como redundância.

Para a comunicação via satélite o sistema utiliza dois canais, o canal de tráfego (traffic channel) bidirecional e o Global Bulletin Board.



**Figura 1.2 – Canais de transmissão de dados**

### 1.2.1 Canal de Trafego

O envio e recebimento de dados entre o terminal e a estação terra ocorre por meio deste canal. Todos os pacotes que trafegam por esse canal são particionados de forma ordenada e identificados em pacotes menores e assim tornando a comunicação robusta e mais confiável.

### 1.2.2 Global Bulletin Board

Bulletin Board é um servidor de computador rodando software personalizado que permite aos usuários se conectar ao sistema usando um programa de terminal. Uma vez conectado, o usuário pode executar funções tais como upload e download de software e dados, leitura de notícias e boletins e trocar mensagens com outros usuários através de e-mail, fóruns públicos e bate-papo via direta [1]. Esse canal de comunicação é usado como medida de redundância, além de permitir que o terminal reaja dinamicamente no campo, como por exemplo em uma relocação de um dos satélites ou em uma mudança na frequência usada pelo terminal, já que diferentes faixas de frequências são usadas ao redor do mundo.

## 2. IDP-780

O IDP-780 é um terminal de comunicação satélite e celular de alto desempenho e eficiência e por isso torna-se economicamente interessante, pois quando há rede celular disponível, é possível enviar os dados por essa rede que é infinitamente mais barato do que enviar a mesma quantidade de dados pelo satélite.



Figura 2 – IDP-780

Esse aparelho consiste em um único gabinete mecânico, um modem para comunicação via satélite, um subsistema GPS integrado, uma interface I/O para sensores e aparelhos, duas portas RS-232, uma porta RS-485, comunicação wireless para celular, uma antena externa para satélite e outra para celular.

### 2.1 Programação

A Skywave fornece ferramentas que permitem ao usuário programar os terminais de maneira que esses aparelhos atendam as especificações requeridas. Essas ferramentas serão exploradas nas próximas seções. A plataforma IDP é baseada na linguagem de programação conhecida como LUA e toda a funcionalidade desses aparelhos é vinculada a códigos LUA criados pelo provedor de solução.

### 2.2 Arquitetura de software

A arquitetura do software é composta, basicamente, por três componentes:

#### 2.2.1 Boot loader

O boot loader é uma pequena partição do firmware e seu principal objetivo é atualizar em série todos os componentes do firmware do terminal.

## 2.2.2 Base Firmware

O Base firmware é uma aplicação do firmware que fornece o compilador e o interpretador de Lua, extensões em Lua podem ser implementadas de forma mais eficiente no firmware, no kernel em tempo real, um sistema de arquivos flash e drivers para serviço Lua para interfacear com o hardware do terminal.

## 2.2.3 Ambiente em Lua

De forma padrão, os terminais IDPs operam utilizando o ambiente de programação em Lua. O serviço shell encontra-se no ambiente Lua e sua função é prover linhas de comando para que os usuários possam configurar, consultar e programar o terminal móvel.

### Lua Parser

O Lua parser também pode ser encontrado no ambiente Lua. Esse interpretador dá acesso direto ao interpretador Lua. Deste modo, usuários podem escrever e executar códigos em Lua diretamente a partir de linhas de comando.

### Lua Service Framework

O terminal faz uso de um framework baseado em Lua para desenvolvimento de aplicações. Lua é uma linguagem robusta que faz uso eficiente da memória e dos recursos do processador.

O sistema é suportado por dois tipos de serviços:

*Core Services* – Serviços que já vem pré-programados para serem utilizados.

*User Services* – Serviços desenvolvidos pelo programador\provedor de solução.

### 3. Serviços

A maioria das funcionalidades do usuário final no terminal IDP é implementada por software e chamados serviços.

Um serviço contém todas as informações e dados necessários para realizar a funcionalidade de um terminal de IDP. Isto é feito de modo que um serviço pode ser carregado no terminal ou atualizado mesmo quando o terminal se encontrar em operação, sem afetar o resto do sistema. É também por esta razão que é uma boa idéia manter o escopo do serviço bastante estreito; vários serviços menores são geralmente mais fáceis de implantar e manter do que um grande serviço.

Um serviço pode ser identificado por uma sequência única de números, conhecidos por service identification number (SIN). Esse número é usado pelo Lua Service Framework (LSF) para questões de endereçamento e assim todas as mensagens enviadas são relacionadas ao serviço apropriado.

## 4. Propriedades

As Propriedades representam atributos de um serviço. Cada propriedade de serviço é atribuído um número de identificação de propriedade (PIN), que pode ser usado para acessá-lo. Propriedades são muito semelhantes às variáveis exportadas por um módulo, mas com algumas melhorias significativas como:

- Qualquer propriedade pode ser usada dentro de definições de mensagem especificando seu SIN e PIN; em mensagens para o terminal, o valor contido no campo correspondente é automaticamente aplicado à propriedade, enquanto na mensagens de terminais, o campo é automaticamente preenchido com o valor da propriedade.

## 5. Estrutura de Mensagens

Definições de mensagens associadas a um serviço têm como objetivo definir como esse serviço recebe e transmite dados na interface aérea (over-the-air). Eles dividem a mensagem em componentes individuais chamados campos. As mensagens também definem a ordem que os campos aparecem, bem como o seu conteúdo.

Assim como serviços e propriedades têm seus identificadores (SIN e PIN), uma mensagem também é identificada por meio de um conjunto de números chamado de Message Identification Number (MIN). As mensagens têm pelo menos duas características básicas a saber: seu nome e seus diversos tipos de campos, que serão explorados mais a frente.

<p style="text-align: center;"><b>Nome</b></p>	<p>Um nome para a mensagem. Isto é puramente por razões estéticas, como mensagens nunca são referidos pelo nome no código.</p>
<p style="text-align: center;"><b>Campos</b></p>	<p>Este é um vetor de definições de campo. Cada entrada de definição de campo no vetor pode conter os seguintes atributos:</p> <p>nome - O nome do campo, usado para se referir programaticamente para o campo.</p> <p>opcional - Especifica se o campo é opcional dentro de uma mensagem. Uma mensagem com um campo não opcional não preenchido é considerada inválida. Este atributo é por si só opcional; o campo está assumiu a não ser opcional se não especificado.</p>

**Tabela 5.1 – Estrutura de uma mensagem**

## 6. Ferramentas de desenvolvimento

Nessa seção são apresentadas as ferramentas de desenvolvimento que compõem a plataforma IDP da Skywave.

- Console – Interface de linha de comando que serve para acessar o terminal e é por meio dela que firmwares e pacotes de serviço são carregados. Esses pacotes têm extensão “.idppkg” e neles estão contidos o firmware e os códigos em LUA.

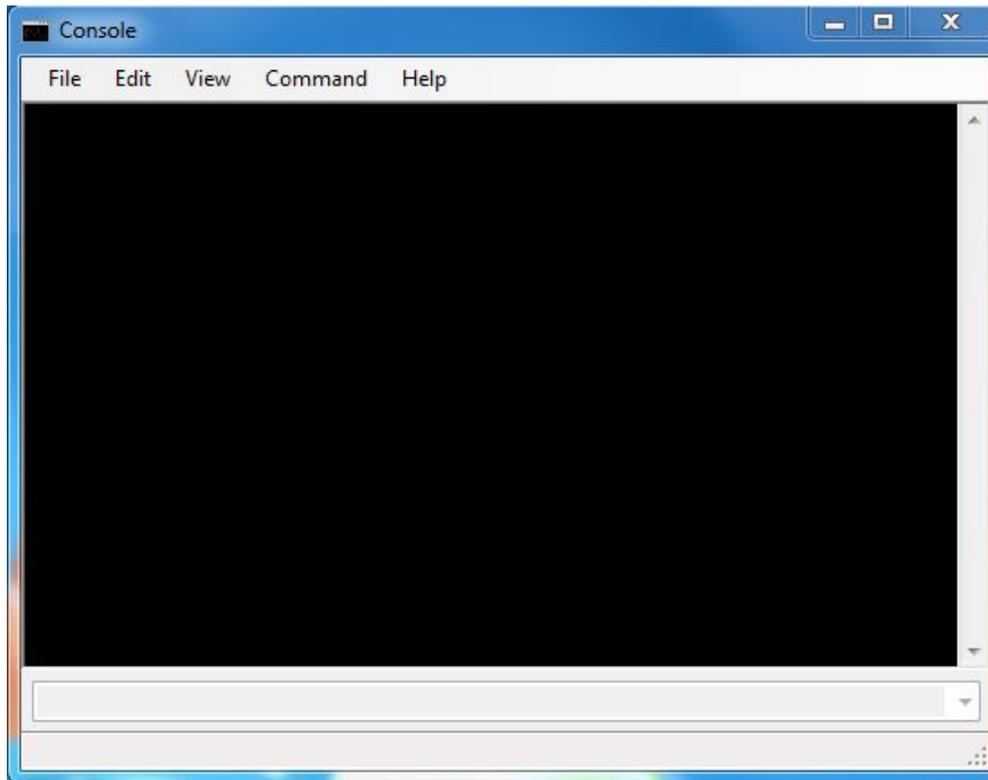


Figura 6.1 - Console

- Messenger - Lugar onde as informações das mensagens tanto from-Mobile quanto to-Mobile são visualizadas. Essa ferramenta utiliza serviço web para receber e enviar mensagens no terminal. Na seção de resultados são mostradas todas as mensagens correspondentes aos códigos LUA desenvolvidos.

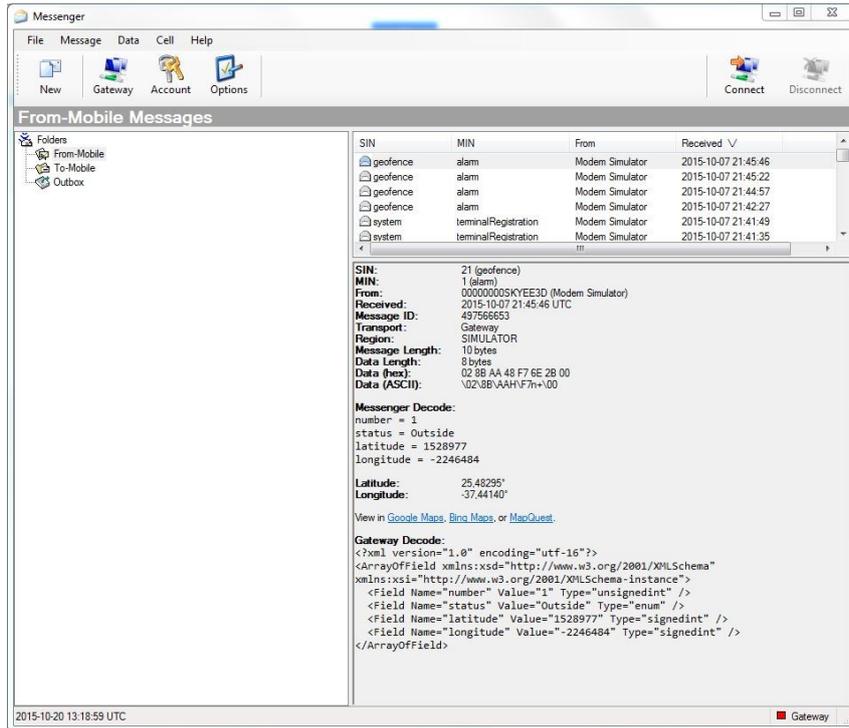


Figura 6.2 - Messenger

- Developer Studio – Plataforma de desenvolvimento do código LUA. Nele é possível criar diversos projetos ou módulos contendo diversos tipos de mensagens tanto to-Mobile quanto from-Mobile e também propriedades em Properties.

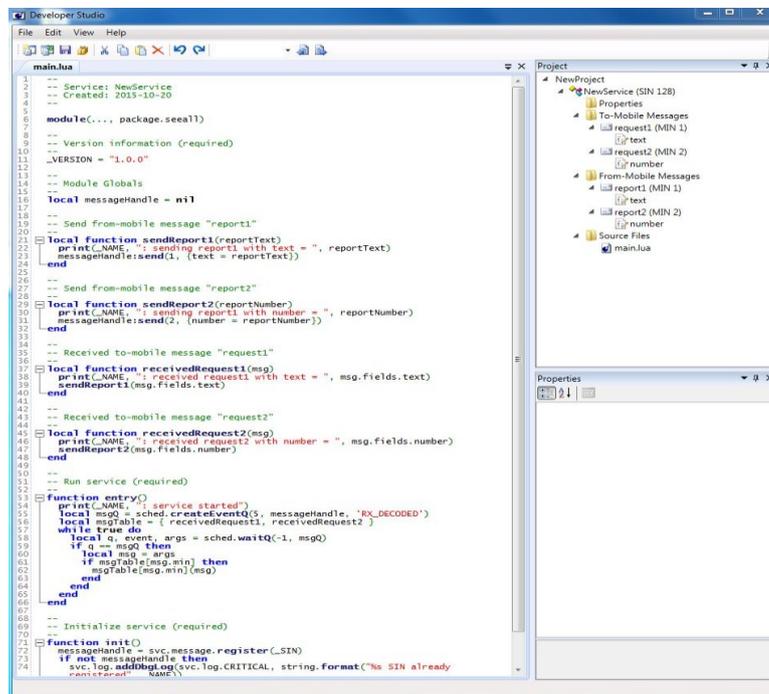
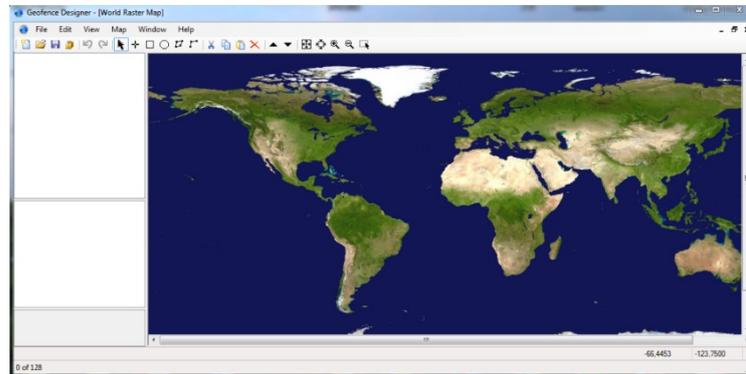


Figura 6.3 – Developer Studio

- Traffic viewer – Aplicação web encontrada no site da Skywave pela qual é possível visualizar todas as mensagens transmitidas via satélite.

- Geofence designer – utilizado para definir rotas e cercas eletrônicas por meio gráfico. Na seção 8.3 essa ferramenta é mostrada em maiores detalhes.



**Figura 6.4 – Geofence Designer**

Lua é uma linguagem de programação bastante robusta, pois combina uma sintaxe simples com construções poderosas para descrição de dados baseadas em tabelas associativas.

Lua é tipada dinamicamente, ou seja, ela é interpretada por uma máquina virtual baseada em registradores e tem um gerenciamento automático de memória, com coleta de lixo incremental. Tudo isso faz com que essa linguagem seja ideal para configuração, automação e prototipagem rápida.

Tipagem dinâmica significa que os tipos das variáveis são verificados pela linguagem somente em tempo de execução do programa. Assim, não é preciso declarar o tipo da variável para que o programa funcione normalmente, o que não acontece em muitas linguagens.

A coleta de lixo incremental torna-se uma vantagem, pois não é mais preciso se preocupar-se com o gerenciamento eficiente de memória.

## 8. Cases em Lua

Todos os códigos em LUA mostrados abaixo foram desenvolvidos com o intuito de ampliar a segurança no controle de ativos de diversas empresas. São três os códigos apresentados:

- 1) Envio de Mensagens
- 2) Monitoramento de baú de caminhão
- 3) Utilização de cercas eletrônicas

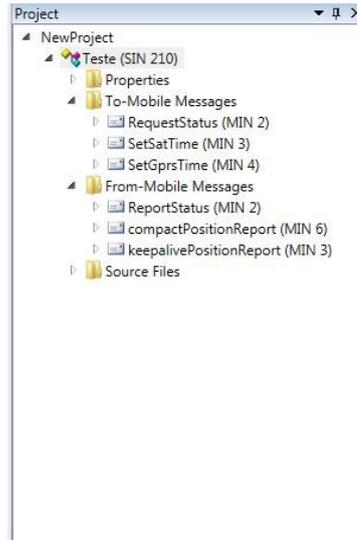
### 8.1 Envio de Mensagens

Esse código foi desenvolvido para uso exclusivo da CarrierWeb. O script Lua tem por objetivo tornar o terminal capaz de receber e enviar mensagens via celular ou satélite. Na figura abaixo é possível encontrar o escopo do que o código abrange.

- Definir qual meio de comunicação deve ser o principal e qual deve ser o secundário (celular e satélite).
- Os dois modos devem ter periodicidades diferentes. As mensagens enviadas por celular devem ser mais frequentes do que as enviadas por satélite. Por exemplo, o terminal deve enviar uma mensagem via GPRS de 3 em 3 minutos enquanto que o satélite deve enviar de 2 em 2 horas, se e somente se o terminal não conseguir por algum motivo, conectar-se à rede celular.
- Se a mensagem foi enviada pelo GPRS, não deve ser enviada pelo satélite, quando o tempo de envio for igual.
- Mudança remota do tempo de periodicidade tanto do celular quanto do satélite.
- Criação de uma mensagem chamada de "Keep Alive", que contém a configuração dos tempos de periodicidade de cada meio de comunicação.
- Possibilidade de desabilitar o envio de mensagens tanto por GPRS quanto por Satélite.

As mensagens enviadas periodicamente ao provedor de solução têm por definição 4 conteúdos em relação ao objeto que deseja ser rastreado (um automóvel ou um navio). Latitude, longitude, velocidade e direção (heading) do objeto.

Para atender a todos os requisitos de funcionalidade foram criadas as mensagens abaixo (to-Mobile e from-Mobile)



**Figura 8.1.1 – Mensagens do escopo do case**

Na seção de resultados é possível encontrar as mensagens acima sendo recebidas e enviadas pelo terminal, que podem ser vistas por meio do Messenger.

## 8.2 Monitoramento do baú de um caminhão

Esse case foi desenvolvido para demonstrar o funcionamento do IDP-780 ao monitorar uma série de sensores. O código simula um baú de um caminhão inicialmente trancado, onde nessa simulação existem 4 componentes.

1. Botão de pânico – Quando pressionado fecha o relé e conseqüentemente ativa o alarme.
2. Solenóide – É ativado a partir do botão de pânico.
3. Buzzer – O alarme soa a partir do momento que o botão de pânico é apertado.
4. Botão de reset – Botão capaz de resetar o sistema e desarmar o alarme.

Para que esse código pudesse ser testado foi utilizada uma placa de teste que continha tanto sensores analógicos quanto digitais, conforme ilustrado na figura abaixo.

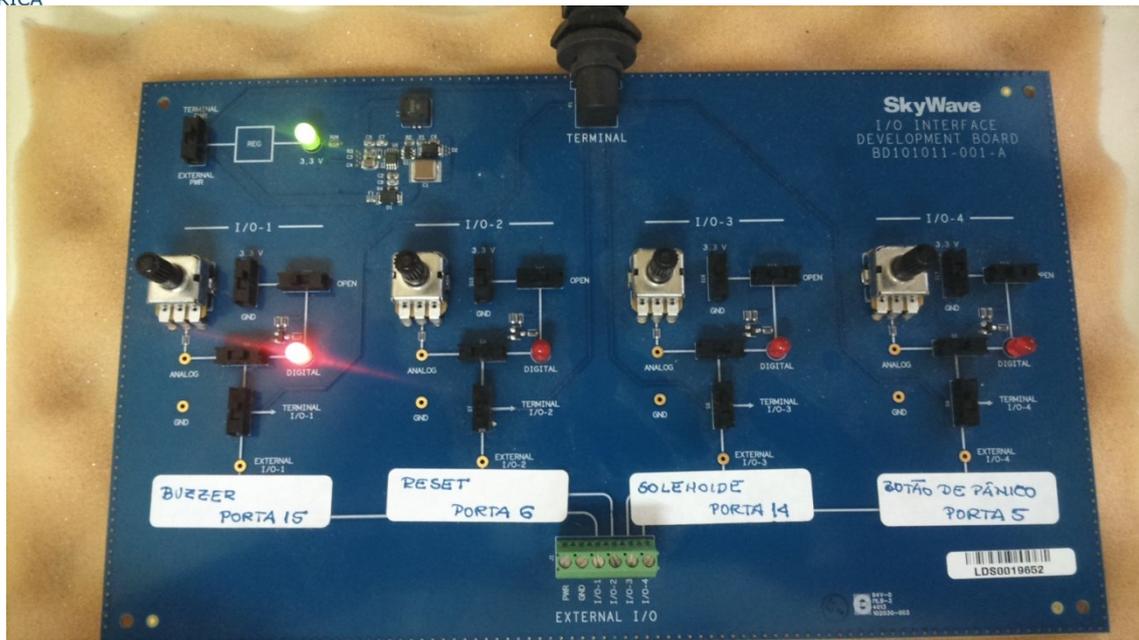


Figura 8.1.2 – Placa de teste para simulação da tranca do baú de um caminhão

### 8.3 Cercas eletrônicas

O IDP-780 é capaz de simular cercas eletrônicas. Essas cercas podem ser definidas de forma visual utilizando-se o software Geofence Designer. A aparência dessa ferramenta é mostrada a seguir.

As cercas podem ser de um dentre três tipos: cercas retangulares, cercas circulares e cercas poligonais. Abaixo, segue a figura de cada uma delas, respectivamente.

Cerca retangular – O botão aonde criar está preenchido de azul, com uma descrição embaixo.

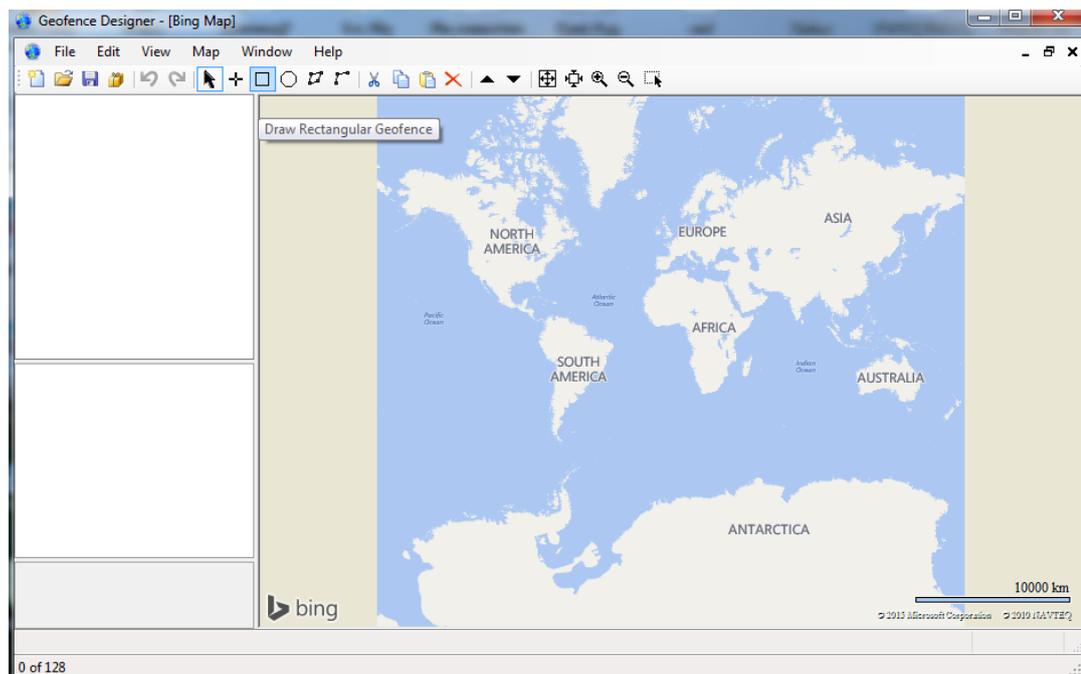
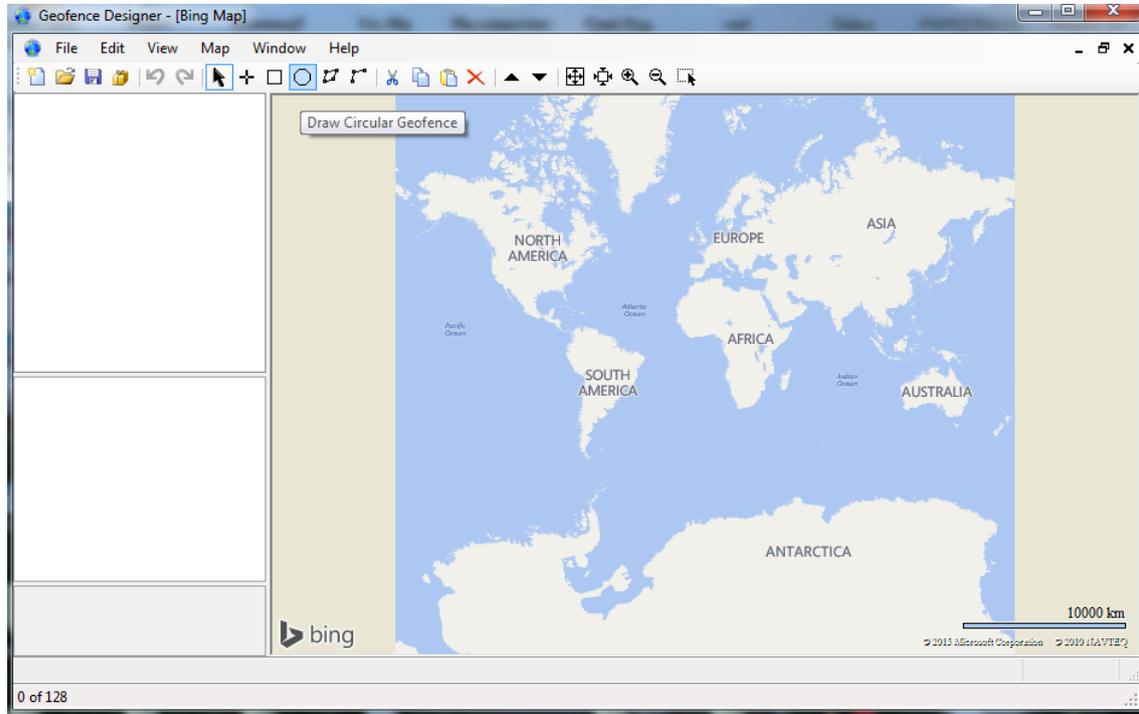


Figura 8.1.3 – Criando cerca retangular

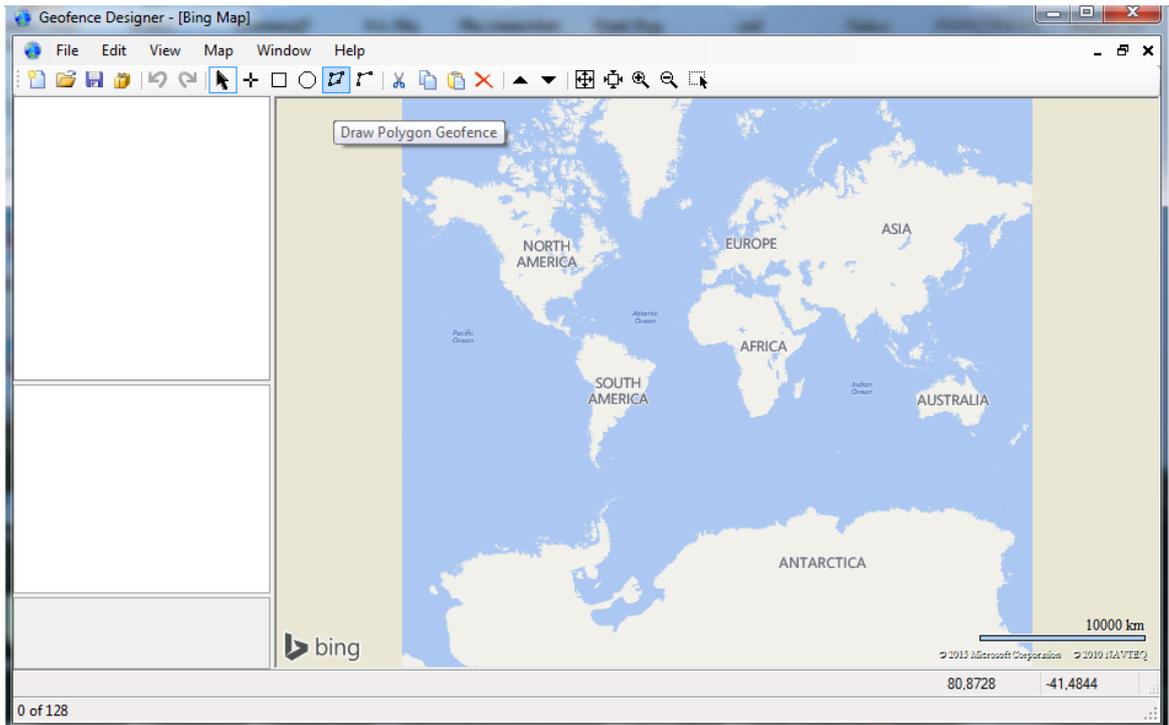
Cerca circular – O botão aonde criar está preenchido de azul, com uma descrição embaixo.



Figura

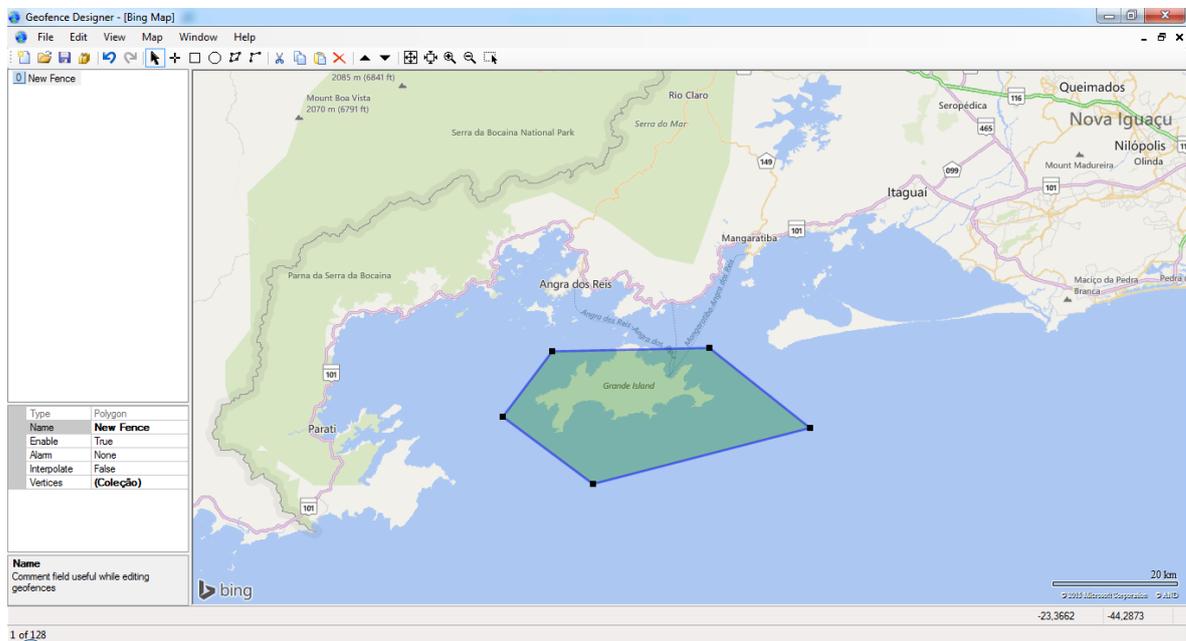
### 8.1.3 – Criando cerca retangular

Cerca poligonal – O botão aonde criar está preenchido de azul, com uma descrição embaixo.



**Figura 8.1.4 – Criando cerca poligonal**

Como exemplo, utilizou-se a cerca poligonal ao redor da ilha grande. Ver próxima figura.



**Figura 8.1.5 – Definindo cerca para desenvolvimento**

De acordo com a imagem acima, é possível ver que uma cerca foi criada e chamada de " [0] New Fence", no topo superior esquerdo. Podemos ver também que além do nome, essa cerca assim como

outras cercas de formatos diferentes apresentam as seguintes características: name, enable e alarm.  
Na cerca poligonal temos duas características particulares são elas: interpolate e vertices.

- Name - trata-se do nome da cerca
- Enable - habilita ou desabilita a cerca tem como valores (true/false)
- Alarm - IDP envia uma mensagem de alarme sempre que uma das opções (Entry, Exit ou Both) for atendida.

## 9. Resultados

Nessa seção são mostrados os resultados de cada caso citado anteriormente. Nela, apresenta-se as respostas do terminal, que definimos como from-Mobile, quando para ele faz uma requisição, definida por to-Mobile.

### 1) Envio de Mensagens

A figura abaixo mostra duas mensagens enviadas pelo gestor ou controlador do terminal, sentido to-Mobile, para saber o status do GPS, (ativo ou não) e também uma mensagem para saber qual o estado do terminal.

No corpo das mensagens enviadas é possível perceber pelo campo (transport), que a comunicação foi via satélite, porque o terminal encontrava-se em algum lugar onde não havia cobertura celular.

#### Solicitação de Status de GPS e Keep-Alive

- Sentido: To-Mobile
- MIN = 2
- Tamanho: 3 bytes (1 byte de payload)
- Composição do Payload:
  - 1 unsigned int (8 bits) que:
    - Se igual a 01, solicita o envio de status do GPS por meio de MIN=2;
    - Se igual a 02, solicita o envio do keep-alive status por meio de MIN=3;

```

SIN:      200 (CW_003_1)
MIN:      2 (RequestStatus)
To:       01043662SKYC643
Transport: Satellite
Submitted: 2014-09-15 17:08:55 UTC
Message ID: 50787652
Message Length: 3 bytes
Data Length: 1 bytes
Data (hex): 01
Data (ASCII): \01

Messenger Decode:
Code = 1
  
```

```

SIN:      200 (CW_003_1)
MIN:      2 (RequestStatus)
To:       01043662SKYC643
Transport: Satellite
Submitted: 2014-09-15 17:11:07 UTC
Message ID: 50787766
Message Length: 3 bytes
Data Length: 1 bytes
Data (hex): 02
Data (ASCII): \02

Messenger Decode:
Code = 2
  
```

Figura 9.1 – GPS Status e Keep-Alive

A figura 9.2 é a resposta do terminal ao requisito feito pelas mensagens to-Mobile. Nela é possível observar se o estado do GPS é ativo ou não.

#### Relatório Decorrente de Solicitação de Status de GPS

- Sentido : From-Mobile
- MIN = 2
- Tamanho: 3 bytes (1 byte de payload)
- Composição do Payload:
  - 1 boolean (8 bits) que:
    - Se igual a 'false' (hex 00), informa que o GPS não está operacional no móvel;
    - Se igual a 'true' (hex 80), informa que o GPS está operacional no móvel;

```

SIN:                200 (CW_003_1)
MIN:                2 (ReportStatus)
From:               01043662SKYC643
Received:           2014-09-17 17:25:24 UTC
Message ID:         325377955
Transport:          Satellite
Region:             AORWSC
Message Length:     3 bytes
Data Length:        1 bytes
Data (hex):         00
Data (ASCII):       \00

Messenger Decode:
ReportGpsStatus = False
  
```

```

SIN:                200 (CW_003_1)
MIN:                2 (ReportStatus)
From:               01043662SKYC643
Received:           2014-09-17 17:18:27 UTC
Message ID:         325371476
Transport:          Satellite
Region:             AORWSC
Message Length:     3 bytes
Data Length:        1 bytes
Data (hex):         80
Data (ASCII):       \80

Messenger Decode:
ReportGpsStatus = True
  
```

Figura 9.2 – GPS Status

As figuras 9.3 e 9.4 abaixo mostram que é possível mudar a frequência com que o terminal envia uma mensagem de latitude e longitude. A figura 9.3 modifica a periodicidade do envio pela rede celular, onde no caso presente essa periodicidade foi alterada para 10 minutos. Já a figura 9.4 apresenta a situação onde o intervalo entre as mensagens enviadas por satélite foi modificado para 15 minutos.

### Solicitação de Status do GPRS

- Sentido: To-Mobile
- MIN = 4
- Tamanho: 4 bytes (2 bytes de payload)
- Composição do Payload:
  - 1 unsigned int (8 bits) que:
    - Se igual a 0, solicita a parada de envio de dados por GPRS;
    - Se maior que 0, define a periodicidade GPRS em minutos.

```
SIN: 200 (CW_003_2)
MIN: 4 (SetGprsTime)
To: 01043662SKYC643
Transport: Satellite
Submitted: 2014-10-16 11:47:38 UTC
Message ID: 53086514
Message Length: 4 bytes
Data Length: 2 bytes
Data (hex): 40 50
Data (ASCII): @P

Messenger Decode:
Set Time = 10
```

Figura 9.3 – Set GPRS Time

### Solicitação de Status do Satélite

- Sentido: To-Mobile
- MIN = 3
- Tamanho: 4 bytes (2 bytes de payload)
- Composição do Payload:
  - 1 unsigned int (8 bits) que:
    - Se igual a 0, solicita a parada de envio de dados por satélite;
    - Se maior que 0, define a periodicidade satelital em minutos.

```

SIN:                200 (CW_003_2)
MIN:                3 (SetSatTime)
To:                 01043662SKYC643
Transport:          Satellite
Submitted:          2014-10-16 11:47:27 UTC
Message ID:         53086506
Message Length:     4 bytes
Data Length:        2 bytes
Data (hex):         40 78
Data (ASCII):       @x

Messenger Decode:
SetTime = 15
  
```

Figura 9.4 – Set Satellite Time

## 2) Monitoramento do baú de um caminhão

Nesse case recebe-se essa mensagem quando o botão de reset foi utilizado (veja ilustração em figura anterior). No caso o sistema informa a porta (porta 6) e o valor para o qual foi alterado (value 0).

O mesmo é válido quando o botão de pânico é acionado. O código foi feito para que houvesse o monitoramento de todas as quatro portas.

```

SIN:                30 (eeio)
MIN:                2 (portAlarm)
From:               00000000SKYEE3D (Modem Simulator)
Received:           2015-10-07 21:26:05 UTC
Message ID:         497566642
Transport:          Gateway
Region:             SIMULATOR
Message Length:     5 bytes
Data Length:        3 bytes
Data (hex):         30 00 00
Data (ASCII):       0\00\00

Messenger Decode:
port = 6
value = 0

Gateway Decode:
<?xml version="1.0" encoding="utf-16"?>
<ArrayOfField xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Field Name="port" Value="6" Type="unsignedint" />
  <Field Name="value" Value="0" Type="signedint" />
</ArrayOfField>
  
```

**Figura 9.5 – Alteração do valor da porta (reset)**

### 3) Cercas eletrônicas

Para as cercas eletrônicas têm-se respostas from-Mobile avisando se a embarcação está dentro ou fora da região demarcada.

```
SIN: 21 (geofence)
MIN: 1 (alam)
From: 00000000SKYEE3D (Modem Simulator)
Received: 2015-10-07 21:44:57 UTC
Message ID: 497566651
Transport: Gateway
Region: SIMULATOR
Message Length: 10 bytes
Data Length: 8 bytes
Data (hex): 04 72 C2 BF 69 82 7A 40
Data (ASCII): \04r\C2\BF\82z@

Messenger Decode:
number = 2
status = Inside
latitude = -1735298
longitude = -5895703

Latitude: -28,92163°
Longitude: -98,26172°

View in Google Maps, Bing Maps, or MapQuest.

Gateway Decode:
<?xml version="1.0" encoding="utf-16"?>
<ArrayOfField xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Field Name="number" Value="2" Type="unsignedint" />
  <Field Name="status" Value="Inside" Type="enum" />
  <Field Name="latitude" Value="-1735298" Type="signedint" />
  <Field Name="longitude" Value="-5895703" Type="signedint" />
</ArrayOfField>
```

**Figura 9.6 – Dentro da cerca**

```
SIN: 21 (geofence)
MIN: 1 (alam)
From: 00000000SKYEE3D (Modem Simulator)
Received: 2015-10-07 21:42:27 UTC
Message ID: 497566650
Transport: Gateway
Region: SIMULATOR
Message Length: 10 bytes
Data Length: 8 bytes
Data (hex): 04 F0 B1 59 75 6E 63 80
Data (ASCII): \04\F0\B1Yunc\80

Messenger Decode:
number = 2
status = Outside
latitude = -2006350
longitude = -2770546

Latitude: -33,43917°
Longitude: -46,17577°

View in Google Maps, Bing Maps, or MapQuest.

Gateway Decode:
<?xml version="1.0" encoding="utf-16"?>
<ArrayOfField xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Field Name="number" Value="2" Type="unsignedint" />
  <Field Name="status" Value="Outside" Type="enum" />
  <Field Name="latitude" Value="-2006350" Type="signedint" />
  <Field Name="longitude" Value="-2770546" Type="signedint" />
</ArrayOfField>
```

**Figura 9.7 – Fora da cerca**

## **10. Conclusão**

De acordo com os casos apresentados nesse trabalho, pode-se concluir que o sistema desenvolvido pela Skywave é uma solução robusta e versátil para aqueles que desejam rastrear e monitorar qualquer coisa ao redor do mundo. Esse sistema torna-se bastante complexo em vários casos de estudo, requerendo que o desenvolvedor do software correspondente tenha amplos conhecimentos na linguagem LUA.

## 11. Referências

- [1] Wikipedia [Online]. [https://en.wikipedia.org/wiki/Bulletin\\_board\\_system](https://en.wikipedia.org/wiki/Bulletin_board_system) . [Acesso em 26/10/2015]
- [2] Skywave [Online]. <http://www.skywave.com/>. [ Acesso 26/10/2015]
- [3] Programing LUA [Online]. <http://www.lua.org/>. [Acesso 10/10/2015]
- [4] T204 IDP Terminal User Guide version 08
- [5] T206 IDP Terminal Hardware Guide