Requirements Specification
for
TracciaSat

July 23, 2011
Version 0.1

Prepared By
Abdul Rasheed Tutakhail

Change History

Date	Version	Description	Updated By
July 23,2011	0,1	Final Draft	Abdul RasheedTutakhail

Document Approvals

Name	Role	Signature

Table of Contents

1 Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of TracciaSat. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system and will be proposed as a compulsory Project for the Software Engineering exam, for its approval.

1.2 Project Scope

This software system will peform the duty of a centralized reciever/controller for GPS based tracker. Being flexible enough to be used in a variety of different applications from Military,Police to Hospitals,Trucking and Taxi companies. SatTraccia gives the user total control on the movements of their vehicles by providing them with their exact location, while they are on the move.The use of TracciaSat will help companies keep their fleet of vehicles under constant serviellence. TracciaSat has a user-friendly interface making it possible for non-expert users to use it without having knowledge regarding behind scene mecchanismw of the system.

More over the system has an automatic update mechanism, whereby according to system settings the vehicle sends information regarding its position(Longitude,Latitude),velocity,date and time which is automatically saved in the system Database.The user through an interface send requests to find out the location and other information of a specific vehicle

1.3 Glossary

Term	Definition
User	Person who introduces the parameters to the software system and recieves results in the form of markers rendered on the map.
Vehicle	The vehicle which holds the GPS tracker, which sends information to the system regarding its position,velocity etc
Longitude	Geographic coordinate that specifies the east-west position of a point on the Earth's surface.
Latitude	Specifies the angular distance south or north of the Equator.
Field	A cell within a form.
Review	A written recommendation about the appropriateness of an article for publication; may include suggestions for improvement.
Software Requirements Specification	A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.
Stakeholder	Any person with an interest in the project who is not a developer.
IMEI	International Mobile Equipment Identity, is a number to identify GSM, WCDMA, and iDEN mobile phones, as well as some satellite phones. In our case it will be unique identifier of the GPS tracker since each Tracker has its own GSM sim-card
Marker	Pointed icons with numbers printed on them,used to pinpoint a location on map.

1.4 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

1.5. Overview of Document

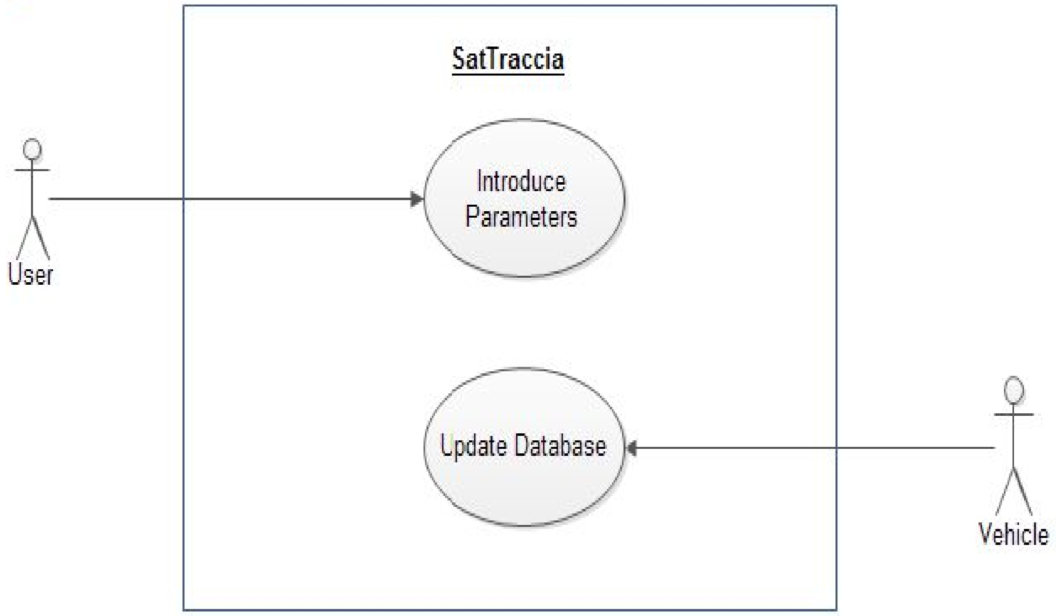
The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2.0. Overall Description

2.1 System Environment



TracciaSat has two actors, User and vehicle.User using the system interface can introduce parameters in form of IMEI,Initial Date and Final Date whereby submitting the information the system renders the associated markers based on the parameters introduced.

Vehicle on the other hand is responsible for sending information regarding its position,velocity etc this data is after validation stored in the Database.

2.2 Functional Requirements Specification

This section outlines the use cases for each of the active readers separately.The User and Vehicle are the main actors in this system.

2.2.1 User Use Case

Use Case: **Introduce Parameters**

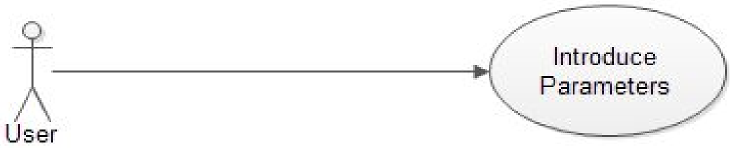


Diagram:

Brief Description
The User enters information about a particular IMEI.

Initial Step-By-Step Description
Before this use case can be initiated, the User has already accessed the main page of the TracciaSat.

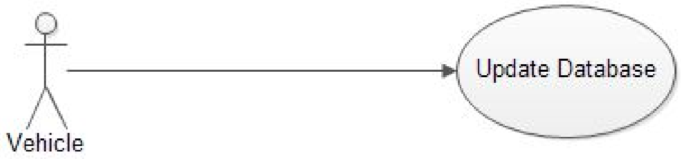
1. The User presents the IMEI.
2. The User specifies the Initial Date.
3. The User specifies the Final Date.

4. The User submits the form once the last three steps have been executed
5. The system processes the information and renders the associated Markers on the map.

2.2.2 Vehicle Use Case

Use case: **Update Database**

Diagram:



Brief Description
The Vehicle sends the System signals regarding its location etc.

Initial Step-By-Step Description
The frequency at which the vehicle tracker device sends the system signale are independent from the software system,which is only responsible of the validation and storage of the information.

1. The Vehicle sends parameters which consists of IMEI,Date,Time,Velocity,Latitude and Longitude.
2. The System checks the parameters.

3. The System checks in the Database if the IMEI is already registered if not registers the IMEI and then stores its corresponding information.

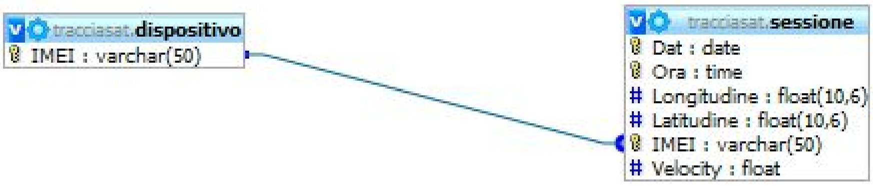
3.0 Database

Struttura della tabella `dispositivo`

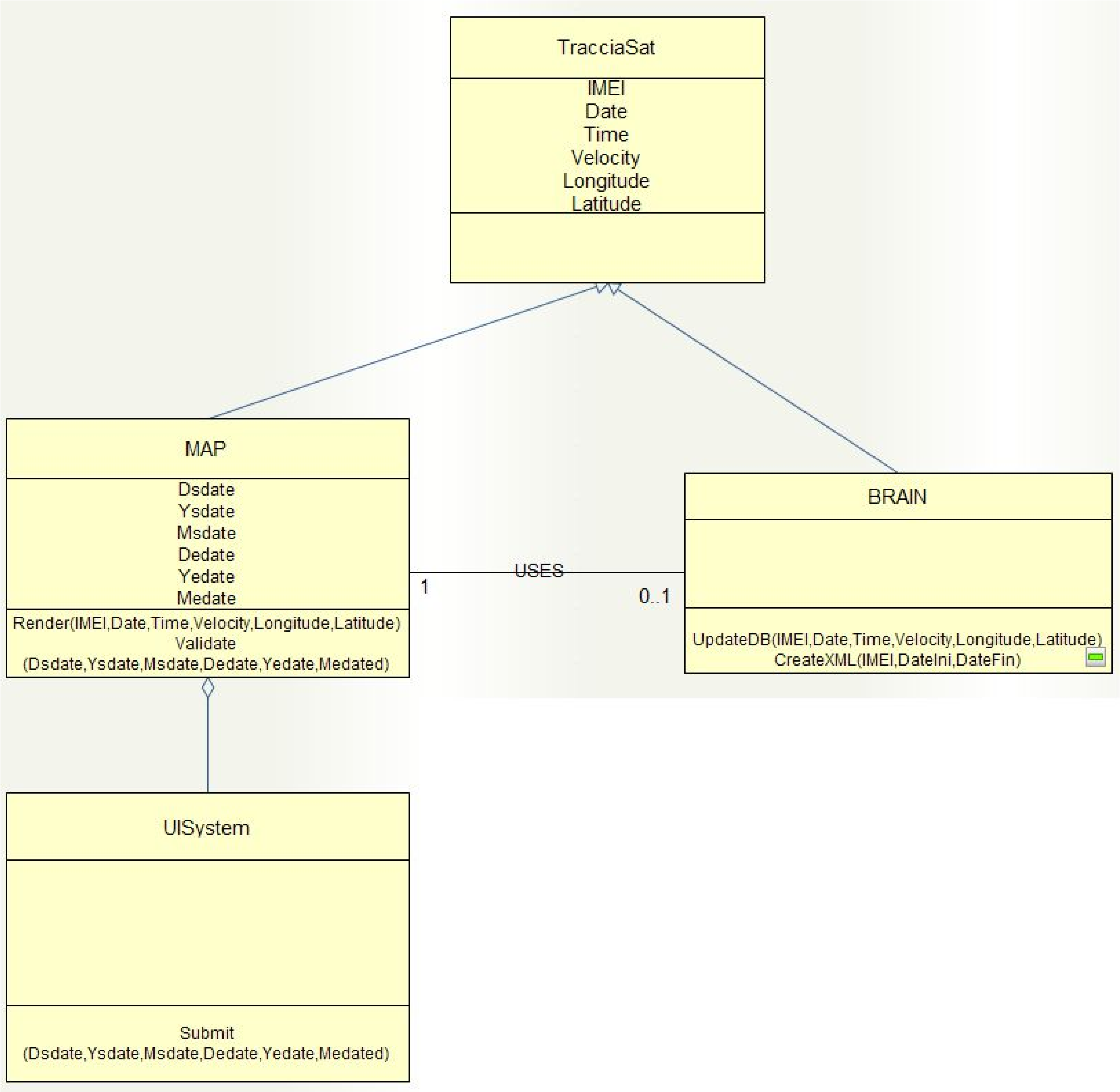
```
CREATE TABLE IF NOT EXISTS `dispositivo` (
  `IMEI` varchar(50) NOT NULL,
   PRIMARY KEY (`IMEI`)
   ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

Struttura della tabella `sessione`

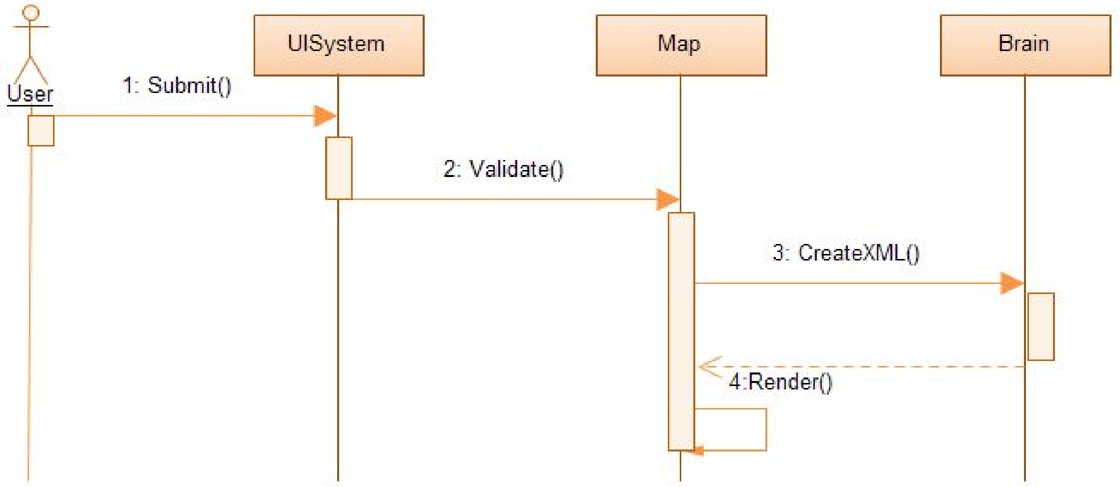
```
CREATE TABLE IF NOT EXISTS `sessione` (
 `Dat` date NOT NULL,
 `Ora` time NOT NULL,
 `Longitudine` float(10,6) NOT NULL,
`Latitudine` float(10,6) NOT NULL,
`IMEI` varchar(50) NOT NULL,
`Velocity` float NOT NULL,
PRIMARY KEY (`Dat`,`Ora`,`IMEI`),
KEY `IMEI` (`IMEI`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

ALTER TABLE `sessione`
ADD CONSTRAINT `sessione_ibfk_1` FOREIGN KEY (`IMEI`) REFERENCES `
dispositivo` (`IMEI`) ON DELETE CASCADE ON UPDATE CASCADE;
```



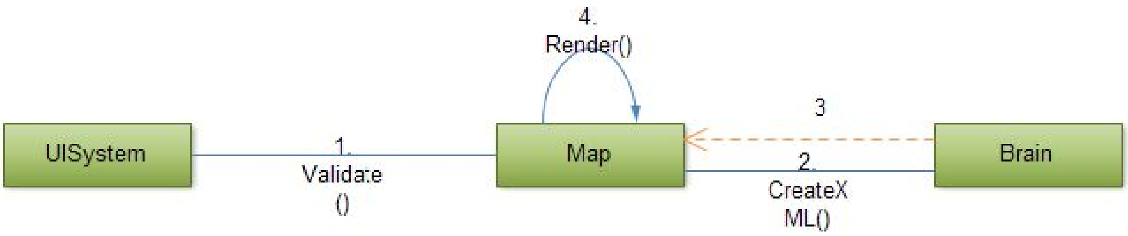
4.0 Class Diagram


TracciaSat
IMEI
Date
Time
Velocity
Longitude
Latitude
MAP
Dsdate
Ysdate
Msdate
Dedate
Yedate
Medate
Render(IMEI,Date,Time,Velocity,Longitude,Latitude)
Validate
(Dsdate,Ysdate,Msdate,Dedate,Yedate,Medated)
USES
1
0..1
BRAIN
UpdateDB(IMEI,Date,Time,Velocity,Longitude,Latitude)
CreateXML(IMEI,DateIni,DateFin)
UISystem
Submit
(Dsdate,Ysdate,Msdate,Dedate,Yedate,Medated)

5.0 Interaction Diagram



6.0 Collaboration Diagram



7.0 Underlying Technologies:

1. Apache Web Server
2. MySQL DBMS
3. JavaScript
4. PHP
5. HTML
6. CSS

and the Google Maps API

8.0 Contents of map.html and second.php

8.1 map.html:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
        <meta name="Author" value="Abdul Rasheed Tutakhail"/>
        <!--Collegamento al file CSS  -->
        <link rel="stylesheet" type="text/css" href="style.css" />

        <!--Chiave del Google Maps  -->

    <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
    <title>TracciaSat</title>
    <script src="http://maps.google.com/maps?
file=api&v=2&key=ABQIAAAAjU0EJWnWPMv7oQ-
jjS7dYxTPZYElJSBeBUeMSX5xXgq6lLjHthSAk20WnZ_iuuzhMt60X_ukms-AUg"
type="text/javascript"></script>
```

```
<!--Collegamento al mapiconmaker.js per creare markers dinamici  -->
<script src="mapiconmaker.js" type="text/javascript"></script      >

<script type="text/javascript">
        //<![CDATA[

        <!--1000 markers insieme con il numero corrispondente vengono create  -->
        var customIcons = [1000];
        for(var j=0;j<1000;j++){
        var iconOptions = {};
        iconOptions.primaryColor = "#0000FF";
        iconOptions.strokeColor = "#000000";
        iconOptions.label = j.toString().replace();
        iconOptions.labelColor = "#000000";
        iconOptions.addStar = false;
        iconOptions.starPrimaryColor = "#FFFF00";
        iconOptions.starStrokeColor = "#0000FF";
        customIcons[j]  = MapIconMaker.createLabeledMarkerIcon(iconOptions);
        }




         function load() {
         var ide=encodeURIComponent(document.getElementById("id").value)
         var ysdate=encodeURIComponent(document.getElementById("ysdate").value)
         var msdate=encodeURIComponent(document.getElementById("msdate").value)
         var dsdate=encodeURIComponent(document.getElementById("dsdate").value)
         var yedate=encodeURIComponent(document.getElementById("yedate").value)
         var medate=encodeURIComponent(document.getElementById("medate").value)
         var dedate=encodeURIComponent(document.getElementById("dedate").value)

        <!--Codice di Validazione del Form -->
        if (ysdate<0 || ysdate>=2100)
{
alert("Il valore nel campo Anno del Data Iniziale non è valido");
return false;
}
       else if (msdate<0 || msdate>12)
{
alert("Il valore nel campo Mese del Data Iniziale non è valido");
return false;
}
        else if (msdate==2 && dsdate>28)
{
alert("Il mese di Febbraio ha 28 giorni, correggi il Campo Giorno del Data Iniziale");
return false;
}
                 else if (dsdate<0 ||dsdate>31)
{
alert("Il valore nel campo Giorno del Data Iniziale non è valido");
return false;
}

                else if (yedate<0 || yedate>=2100)
{
alert("Il valore nel campo Anno del Data Finale non è valido");
return false;
}
       else if (medate<0 || medate>12)
{
alert("Il valore nel campo Mese del Data Finale non è valido");
return false;
}
        else if (medate==2 && dedate>28)
{
```

```
                alert("Il mese di Febbraio ha 28 giorni, correggi il Campo Giorno del Data Finale");
                return false;
                }
                        else if (dedate<0 ||dedate>31)
                {
                alert("Il valore nel campo Giorno del Data Finale non è valido");
                return false;
                }




                    <!--La mappa si concentra sulla posizione con Longitudine e Latitudine
precisata nella funzione GlatLng con l'avvio di funzione Load()  -->
            if (GBrowserIsCompatible()) {
                    var map = new GMap2(document.getElementById("map"));
                    map.addControl(new GSmallMapControl());
                    map.addControl(new GMapTypeControl());
                    map.setCenter(new GLatLng(43.110654076756994, 12.386398315429688),
13, G_HYBRID_MAP);


    <!--GDownloadUrl prende come parametro il file che produce in output XML secondo i dati
contenuti nel Database, Gli vengono passati i dati ottenuti dal Form del utente   -->
    GDownloadUrl("second.php?
strt="+ysdate+"/"+msdate+"/"+dsdate+"&end="+yedate+"/"+medate+"/"+dedate+"&id="+ide,
function(data) {
                    var xml = GXml.parse(data);
                    var markers = xml.documentElement.getElementsByTagName("marker");
                    for (var i = 0; i < markers.length; i++) {
                    var IMEI = markers[i].getAttribute("IMEI");
                    var Velocity = markers[i].getAttribute("Velocity");
                    var Ora = markers[i].getAttribute("Ora");
                    var Data = markers[i].getAttribute("Data");
                    var point = new GLatLng(parseFloat(markers[i].getAttribute("lat")),

        parseFloat(markers[i].getAttribute("lng")));
```

```
                        var marker = createMarker(point, IMEI, Velocity, Ora, Data, i);

                        map.addOverlay(marker);

                     }
                    });
        }
       }

                    function createMarker(point, IMEI, Velocity, Ora, Data, i) {
                        var marker = new GMarker(point, customIcons[i]);
                        var html = "<b>" + "Ora: " + "</b>" + Ora + "<br/>" + "<b>" + "Data: " +
"</b>"+ Data + "<br/>" + "<b>" + "Velocità: " + "</b>" + Velocity + " km/h" ;
                        GEvent.addListener(marker, 'click', function() {
                        marker.openInfoWindowHtml(html); });
                        return marker;
                     }
                    //]]>
            </script>

 </head>

            <!--La mappa di Google  -->
 <body onload="load()" onunload="GUnload()">
   <div id="map" style="width: 1000px; height: 350px; left: 123px; align:center; "></div>

            <!--Form del Utente  -->
            <div id="stylized" class="myform">
            <form id="form" name="form" method="get" action="">

   <h1>TracciaSat</h1>

            <label>IMEI
    <span class="small">(Composto da 15 cifre)</span>
    </label>
    <input type="text" name="id" id="id" /></br></br>

            <label>Data Iniziale
            <span class="small">(YYYY/MM/DD)</span>
            </label>
            <input type="text" id="ysdate" name="ysdate" size="3" value="2000" /> <input type="text" id="msdate" name="ysdate" size="1" /> <input type="text" id="dsdate" name="dsdate" size="1" /> </br></br>

            <label>Data Finale
            <span class="small">(YYYY/MM/DD)</span>
            </label>
            <input type="text" id="yedate" name="yedate" size="3" /> <input type="text" id="medate" name="medate" size="1" /> <input type="text" id="dedate" name="dedate" size="1" /> </br></br>

            <button type="button" onClick="load()">Traccia</button>
            <div class="spacer"></div>
            </form>
            </div>



  </body>
</html>
```

8.2 second.php:

```
<?php


        /*Il file PHP second i parametri che gli vengono passati reagisce in modo diverso, nel primo
caso produce XML in output secondo i parametri dell'utente e nel secondo caso aggiorna il
Database  */
        if (!isset($_GET['dat'],$_GET['tim'],$_GET['spee'],$_GET['LON'],$_GET['LA'],
$_GET['IMEI'])){


        function parseToXML($htmlStr)
                {
                $xmlStr=str_replace('<','&lt;',$htmlStr);
                $xmlStr=str_replace('>','&gt;',$xmlStr);
                $xmlStr=str_replace('"','&quot;',$xmlStr);
                $xmlStr=str_replace("'",'&#39;',$xmlStr);
                $xmlStr=str_replace("&",'&amp;',$xmlStr);
                return $xmlStr;
                }
                // Apri connession verso il server mySQL
                $connection=mysql_connect ("localhost","root","alphabravo");
                if (!$connection) {
                        die('Not connected : ' . mysql_error());
                                                }

                // Rende il Database mySQL attiva
                $db_selected = mysql_select_db("tracciasat", $connection);
                if (!$db_selected) {
                        die ('Can\'t use db : ' . mysql_error());
                                }


                // Seleziona le righe dalla tabella sessione che esistono con il IMEI specificato e
esistono entra le date passate
                $query = "SELECT * FROM sessione WHERE Dat BETWEEN '$_GET[strt]' AND
```

```
'$_GET[end]' AND IMEI = '$_GET[id]'";
                $result = mysql_query($query);
                if (!$result) {
                        die('Invalid query: ' . mysql_error());
                                        }
                header("Content-type: text/xml");

                // Inizio del file XML, echo node padre
                echo '<markers>';

                // Itera tra le righe,creando dei nodi XML per ognuno di loro
                while ($row = @mysql_fetch_assoc($result)){
                echo '<marker ';
                echo 'IMEI="' . parseToXML($row['IMEI']) . '" ';
                echo 'Velocity="' . parseToXML($row['Velocity']) . '" ';
                echo 'Ora="' . parseToXML($row['Ora']) . '" ';
                echo 'Data="' . parseToXML($row['Dat']) . '" ';
                echo 'lat="' . $row['Latitudine'] . '" ';
                echo 'lng="' . $row['Longitudine'] . '" ';
                echo '/>';
                                                                                }


                // Fine del file XML
                echo '</markers>';


                }
 else{

                // Il secondo caso dove aggiorna il Database con i parametri passati automaticamente
dal dispositivo
                $con = mysql_connect("localhost","root","alphabravo");
                if(!$con){
                        die('Could not connect: ' . mysql_error());
                        }
```

```
        else{
            mysql_select_db("tracciasat", $con);
            $resul=mysql_query("select * FROM dispositivo") or die("Error couldn't
execute query: " . mysql_error());
            while($row=mysql_fetch_array($resul)){
            if($row['IMEI']!= $_GET['IMEI']){
                $updat=mysql_query("INSERT INTO dispositivo
VALUES('$_GET[IMEI]')");


                                            }  }


        $sq=mysql_query("INSERT INTO sessione VALUES
('$_GET[dat]','$_GET[tim]','$_GET[LON]','$_GET[LA]','$_GET[IMEI]','$_GET[spee]')") or
die("Error inserting new data: " . mysql_error());


        echo " 1 record aggiunto";
    mysql_close($con);
        }}
?>
```