Project Assignment (Part II)

Sistemas de Informação e Base de Dados

Grupo nº 59

75661 - João Salvado

75724 - Vera Pedras

76090 - Diogo Ferreira

1. Criar a Base de Dados

```
create table Patient (
        number      integer,
        name        varchar(255),
        address     varchar(255),
        primary key(number));

create table PAN (
        domain      varchar(255),
        phone       integer,
        primary key(domain));

create table Device (
        serialnum       integer,
        manufacturer    varchar(255),
        description     varchar(255),
        primary key(snum,manuf));

create table Sensor (
        snum        integer,
        manuf       varchar(255),
        units       varchar(255),
        primary key(snum,manuf),
        foreign key(snum,manuf) references Device(snum,manuf));

create table Actuator (
        snum        integer,
        manuf       varchar(255),
        units       varchar(255),
        primary key(snum,manuf),
        foreign key(snum,manuf) references Device(snum,manuf));

create table Municipality (
        nut4code    varchar(255),
        name        varchar(255),
        primary key(nut4code));

create table Period (
        start       timestamp,
        end         timestamp,
        primary key(start,end));
```

```
create table Reading (
        snum            integer,
        manuf           varchar(255),
        datetime        timestamp,
        value           numeric(20,2),
        primary key(snum,manuf,datetime),
        foreign key(snum,manuf) references Sensor (snum,manuf));

create table Setting (
        snum            integer,
        manuf           varchar(255),
        datetime        timestamp,
        value           numeric(20,2),
        primary key(snum,manuf,datetime),
        foreign key(snum,manuf) references Actuator (snum,manuf));

create table Wears (
        start           timestamp,
        end             timestamp,
        patient         integer,
        pan             varchar(255),
        primary key(start,end,patient),
        foreign key(start,end) references Period (start,end),
        foreign key(patient) references Patient (number),
        foreign key(pan) references PAN (domain));

create table Lives (
        start           timestamp,
        end             timestamp,
        patient         integer,
        muni            varchar(255),
        primary key(start,end,patient),
        foreign key(start,end) references Period (start,end),
        foreign key(patient) references Patient (number),
        foreign key(muni) references Municipality (nut4code));

create table Connects (
        start           timestamp,
        end             timestamp,
        snum            integer,
        manuf           varchar(255),
```

```
    pan             varchar(255),
    primary key(start,end,snum,manuf),
    foreign key(start,end) references Period (start,end),
    foreign key(snum,manuf) references Device (snum,manuf),
    foreign key(pan) references PAN (domain));
```

2. Triggers para sobreposição de períodos

Para realizar os triggers que evitam a sobreposição de períodos é necessário definir as seguintes funções:

As duas funções que se seguem retornam o tempo de início do período que tem o mínimo tempo de fim maior que o tempo de fim a inserir nas tabelas Connects e Wears, respectivamente.

```
delimiter $$
create function min_start_maior_new_end_connects(s_num integer, new_manuf
varchar(255), new timestamp)
returns timestamp
begin
    declare m_start timestamp;
    set m_start = NULL;
    select start into m_start
    from Connects as c
    where c.snum = s_num
            and c.manuf = new_manuf
            and end > new
            and end <= all(select end
                        from Connects
                        where snum = s_num
                                and manuf = new_manuf
                                and end > new);
        return m_start;
end$$
delimiter ;
```

```
delimiter $$
create function min_start_maior_new_end_wears(pat integer, new timestamp)
returns timestamp
begin
        declare m_start timestamp;
```

```
        set m_start = NULL;
        select start into m_start
        from Wears as w
        where w.patient = pat
                and end > new
                and end <= all(select end
                                from Wears
                                where patient = pat
                                        and end > new);
return m_start;
end$$
delimiter ;
```

As duas funções que se seguem retornam o tempo de fim do período que tem o máximo tempo de início menor que o tempo de início a inserir nas tabelas Connects e Wears, respectivamente.

```
delimiter $$
create function max_end_menor_new_start_connects(s_num integer, new_manuf
varchar(255), new timestamp)
returns timestamp
begin
    declare m_end timestamp;
    set m_end = NULL;
    select end into m_end
    from Connects as c
    where c.snum = s_num
                and c.manuf = new_manuf
                and start < new
                and start >= all(select start
                                from Connects as c2
                                where c2.snum = s_num
                                        and c2.manuf = new_manuf
                                        and start < new);
    return m_end;
end$$
delimiter ;
```

```
delimiter $$
create function max_end_menor_new_start_wears(pat integer, new timestamp)
returns timestamp
begin
        declare m_end timestamp;
        set m_end = NULL;
        select end into m_end
        from Wears as w
        where w.patient = pat
                and start < new
                and start >= all(select start
                                from Wears
                                where patient = pat
                                        and start < new);
        return m_end;
end$$
delimiter ;
```

Para impedir que haja sobreposição de períodos é necessário ter quatro triggers diferentes. Dois para quando se insere algo ou se actualiza a tabela Connects, e outros dois com funções semelhantes mas para a tabela Wears.

1º Trigger: inserção na tabela Connects

```
delimiter $$
create trigger check_overlapping_periods_insert_connects before insert on Connects
for each row
begin

        if(exists(select snum from Connects where end > new.start and end < new.end
                and snum = new.snum and manuf = new.manuf)) then
                call Device_already_connected_in_that_time();
        end if;
        if(exists(select snum from Connects where start > new.start and start <
                new.end and snum = new.snum and manuf = new.manuf)) then
                call Device_already_connected_in_that_time();
        end if;

        if (min_start_maior_new_end_connects(new.snum, new.manuf, new.end) IS
        NOT NULL) then
                if(new.end > min_start_maior_new_end_connects(new.snum,
                new.manuf, new.end)) then
```

```
                call Device_already_connected_in_that_time();
            end if;
        end if;
        if (max_end_menor_new_start_connects(new.snum, new.manuf, new.start) IS
        NOT NULL) then
            if(new.start < max_end_menor_new_start_connects(new.snum,
            new.manuf, new.start)) then
                call Device_already_connected_in_that_time();
            end if;
        end if;
end$$
delimiter;
```

2º Trigger: actualização da tabela Connects

```
delimiter $$
create trigger check_overlapping_periods_update_connects before update on
Connects
for each row
begin
        if(new.end > old.end or new.start < old.start)then
            if(exists(select snum from Connects where end > new.start and end <
            new.end and snum = new.snum and manuf = new.manuf and end <>
            old.end)) then
                call Device_already_connected_in_that_time();
            end if;
            if(exists(select snum from Connects where start > new.start and start <
            new.end and snum = new.snum and manuf = new.manuf and start <>
            old.start)) then
                call Device_already_connected_in_that_time();
            end if;
            if (min_start_maior_new_end_connects(new.snum, new.manuf,
            new.end) IS NOT NULL) then
                if(new.end > min_start_maior_new_end_connects(new.snum,
                new.manuf, new.end) and
                min_start_maior_new_end_connects(new.snum, new.manuf,
                new.end) <> old.start) then
                    call Device_already_connected_in_that_time();
                end if;
            end if;
            if (max_end_menor_new_start_connects(new.snum, new.manuf,
            new.start) IS  NOT NULL) then
```

```
                    if(new.start < max_end_menor_new_start_connects(new.snum,
                    new.manuf, new.start) and
                    max_end_menor_new_start_connects(new.snum, new.manuf,
                    new.start) <> old.end) then
                        call Device_already_connected_in_that_time();
                    end if;
                end if;
        end if;
end$$
delimiter;
```

3º Trigger: inserção na tabela Wears

```
delimiter $$
create trigger check_overlapping_periods_insert_wears before insert on Wears
for each row
begin
        if(exists(select patient from Wears where end > new.start and end < new.end
        and patient = new.patient)) then
            call Patient_already_connected_in_that_time();
        end if;
        if(exists(select patient from Wears where start > new.start and start < new.end
        and patient = new.patient)) then
            call Patient_already_connected_in_that_time();
        end if;
        if (min_start_maior_new_end_wears(new.patient, new.end) IS NOT NULL) then
            if(new.end > min_start_maior_new_end_wears(new.patient, new.end))
            then
                call Patient_already_connected_in_that_time();
            end if;
        end if;
        if (max_end_menor_new_start_wears(new.patient, new.start) IS NOT NULL)
        then
            if(new.start < max_end_menor_new_start_(new.patient, new.start))
            then
                call Patient_already_connected_in_that_time();
            end if;
        end if;
end$$
delimiter;
```

4º Trigger: actualização da tabela Wears

```
delimiter $$
create trigger check_overlapping_periods_update_wears before update on Wears
for each row
begin
	if(new.end > old.end or new.start < old.start)then

		if(exists(select patient from Wears where end > new.start and end < new.end and patient = new.patient and end <> old.end)) then
			call Patient_already_connected_in_that_time();
		end if;
		if(exists(select patient from Wears where start > new.start and start < new.end and patient = new.patient and start <> old.start) IS NOT NULL) then
			call Patient_already_connected_in_that_time();
		end if;
		if (min_start_maior_new_end_wears(new.patient, new.end) IS NOT NULL) then
			if(new.end > min_start_maior_new_end_wears(new.patient, new.end) and min_start_maior_new_end_wears(new.patient, new.end) <> old.start) then
				call Patient_already_connected_in_that_time();
			end if;
		end if;
		if (max_end_menor_new_start_wears(new.patient, new.start) IS NOT NULL) then
			if(new.start < max_end_menor_new_start_(new.patient, new.start) and max_end_menor_new_start_(new.patient, new.start) <> old.end) then
				call Patient_already_connected_in_that_time();
			end if;
		end if;
	end if;
end$$
delimiter;
```

3. Queries

(a)

Esta query será feita para um paciente cujo o número é 1234. Mudando este valor na query é possível obter o resultado para qualquer paciente do centro de saúde.

```
select w.patient, r.datetime, r.value, s.units
from Reading as r, Connects as c, Wears as w, Device as d, Sensor as s
where s.snum = r.snum and s.manuf = r.manuf and r.snum = c.snum
        and r.manuf = c.manuf and c.pan = w.pan
        and d.serialnum = c.snum and d.manufacturer = c.manuf
        and c.end <= w.end and c.start >= w.start and r.datetime>=c.start and
    r.datetime <= c.end
        and timestampdiff(month, r.datetime, current_timestamp) <= 6
        and description like '%blood pressure%'
        and w.patient = 1234;
```

(b)

```
select m.nut4code, m.name
from Municipality as m, Lives as l, Wears as w, Connects as c
where m.nut4code = l.muni and l.patient = w.patient and w.pan = c.pan
                and manuf = 'Philips'
                and c.end >= current_timestamp
                and w.end >= current_timestamp
                and l.end >= current_timestamp
group by nut4code
having count(distinct snum) >= all(select count(distinct snum)
                                   from Municipality as m, Lives as l, Wears as w,
                                        Connects as c
                                   where m.nut4code = l.muni
                                        and l.patient = w.patient
                                        and w.pan = c.pan
                                        and manuf = 'Philips'
                                        and c.end >= current_timestamp
                                        and w.end >= current_timestamp
                                        and l.end >= current_timestamp
                                   group by nut4code);
```

(c)

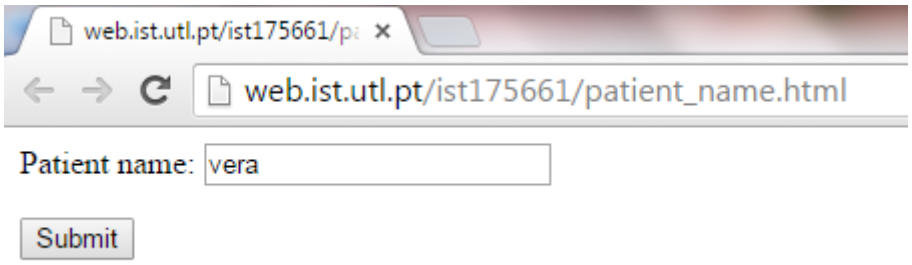
```
select distinct d.manufacturer
from Device as d, Connects as c, Wears w, Lives as l
where d.description like '%scale%' and d.serialnum = c.snum
        and d.manufcturer = c.manuf and c.pan = w.pan
        and l.patient = w.patient and l.end >= w.end and w.end >= c.end
        and l.start <= w.start and w.start <= c.start
        and timestampdiff(year, c.end, current_timestamp) <= 1
        and not exists(
                select nut4code
                from Municipality
                where nut4code not in(
                        select l2.muni
                        from Device as d2, Connects as c2, Wears w2, Lives as l2

                where  d2.description like '%scale%'
                        and d2.serialnum = c2.snum
                        and d2.manufcturer = c2.manuf
                        and c2.pan = w2.pan and l2.patient = w2.patient
                        and l2.end >= w2.end and w2.end >= c2.end
                        and l2.start <= w2.start and w2.start <= c2.start
                        and timestampdiff(year, c2.end, current_timestamp) <= 1
                        and d2.manufacturer = d.manufacturer));
```

4.

(a)

patient_name.html

```
<html>
    <body>
        <form action="readings_settings.php">
            <p>Patient name: <input type="text" name="patient_name"/></p>
            <p><input type="submit" value="Submit"/></p>
        </form>
    </body>
</html>
```

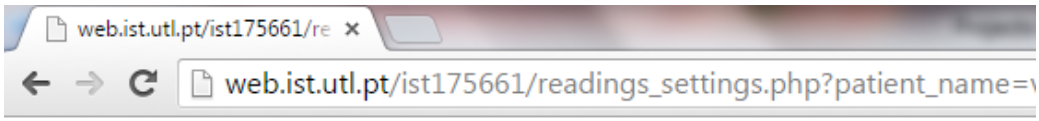


readings_settings.php

```
<html>
	<body>
<?php

/*Ligação à base de dados*/

	$host="db.ist.utl.pt";
	$user="ist175661";
	$pass="gejc9717";
	$dsn="mysql:host=$host; dbname=$user";

	try{
		$connection= new PDO($dsn,$user,$pass);
	}catch(PDOException $exception){
		echo("<p>Error: ");
		echo($exception->getMessage());
		echo("</p>");
		exit();
	}

	$patientname =$_REQUEST['patient_name'];
/*Pacient exists?*/
	$sql5="SELECT number, name
			FROM Patient
			WHERE name like '%$patientname%'
			ORDER BY number";

		$result5 = $connection->query($sql5);

		if ($result5 == FALSE){
			$info = $connection->errorInfo();
			echo("<p>Error: {$info[2]}</p>");
			exit();
		}
		$nrows5=$result5->rowCount();
		if($nrows5 <> 0){

/*Query Readings*/
```

```
	$sql="SELECT p.name,p.number,r.datetime,r.manuf,r.snum,r.value,s.units FROM
Reading as r,Sensor as s,Connects as c,Wears as w,Patient as p Where p.name like
'%$patientname%' and p.number=w.patient and w.pan=c.pan and w.start <= c.start and w.end
>= c.end and c.snum=s.snum and c.manuf = s.manuf and s.snum= r.snum and s.manuf =
r.manuf and r.datetime >= c.start and r.datetime <=c.end order by
p.number,r.datetime,r.manuf";
	$result = $connection->query($sql);

	if ($result == FALSE){
		$info = $connection->errorInfo();
		echo("<p>Error: {$info[2]}</p>");
		exit();
	}

	$nrows=$result->rowCount();
	if($nrows <> 0){

/*Display da tabela_Readings*/
	echo("<h3>Readings</h3>");

	echo("<table border=\"1\">");
	echo("<tr><td>Number<td>Name</td><td>DateTime</td><td>Manufacturer</td><td>
SerialNumber</td><td>Value</td><td>Units</td></tr>");
	foreach($result as $row){
		echo("<tr><td>");
		echo($row['number']);
		echo("</td><td>");
		echo($row['name']);
		echo("</td><td>");
		echo($row['datetime']);
		echo("</td><td>");
		echo($row['manuf']);
		echo("</td><td>");
		echo($row['snum']);
		echo("</td><td>");
		echo($row['value']);
		echo("</td><td>");
		echo($row['units']);
		echo("</td></tr>");
		}
	echo("</table>");
	}else{echo("<p>Patient has 0 Readings</p>");}
/*Query Settings*/

	$sql2="Select p.name,p.number,s.datetime,s.manuf,s.snum,s.value,a.units from Setting
as s,Actuator as a,Connects as c,Wears as w,Patient as p where p.name like '%$patientname%'
and p.number = w.patient and w.pan=c.pan and w.start<= c.start and w.end >=c.end and
c.snum = a.snum and c.manuf=a.manuf and a.snum=s.snum and a.manuf= s.manuf and
s.datetime >= c.start and s.datetime <= c.end order by p.number,s.datetime,s.manuf";
    $result2 = $connection->query($sql2);

    if ($result2 == FALSE){
```

```
            $info = $connection->errorInfo();
            echo("<p>Error: {$info[2]}</p>");
            exit();
        }

          $nrows2=$result2->rowCount();

          if($nrows2 <> 0){
/*Display da Tabela Settings*/

          echo("<h3>Settings</h3>");
          echo("<table border=\"1\">");

echo("<tr><td>Number</td><td>Name</td><td>DateTime</td><td>Manufacturer</td><td>S
erialNumber</td><td>Value</td><td>Units</td></tr>");
        foreach($result2 as $row){
            echo("<tr><td>");
                echo($row['number']);
                echo("</td><td>");
                echo($row['name']);
                echo("</td><td>");
                 echo($row['datetime']);
                 echo("</td><td>");
                 echo($row['manuf']);
                 echo("</td><td>");
                 echo($row['snum']);
                 echo("</td><td>");
                echo($row['value']);
                 echo("</td><td>");
                 echo($row['units']);
                echo("</td></tr>");
        }
        echo("</table>");

}else{echo("<p>Patient has 0 Settings</p>");}
}else{echo("<p>Patient not found!</p>");}
$connection = NULL;
?>
          </body>
</html>
```



Readings

Number	Name	DateTime	Manufacturer	SerialNumber	Value	Units
1	vera	2014-10-23 09:48:23	Philips	33	8.00	mm
1	vera	2015-02-07 00:00:00	Philips	34	13.00	mmHg

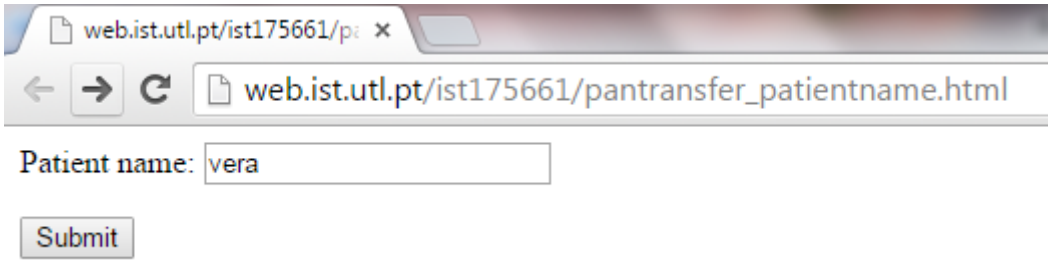
Settings

Number	Name	DateTime	Manufacturer	SerialNumber	Value	Units
1	vera	2014-10-28 00:00:00	Philips	33	33.30	mm
1	vera	2014-10-29 00:07:00	Philips	33	33.40	mm
1	vera	2014-11-01 00:07:00	Philips	33	33.50	mm
1	vera	2015-02-08 03:07:40	Philips	34	10.50	mmHg
1	vera	2015-02-09 03:07:40	Philips	34	10.40	mmHg
1	vera	2015-11-11 02:08:40	SONY	33	12.53	mm
1	vera	2015-11-12 02:08:40	SONY	33	12.54	mm
1	vera	2015-11-23 00:07:40	Philips	33	14.50	mm
1	vera	2015-11-23 03:07:40	Philips	33	13.50	mm

b)

pantranfer_patientname.html

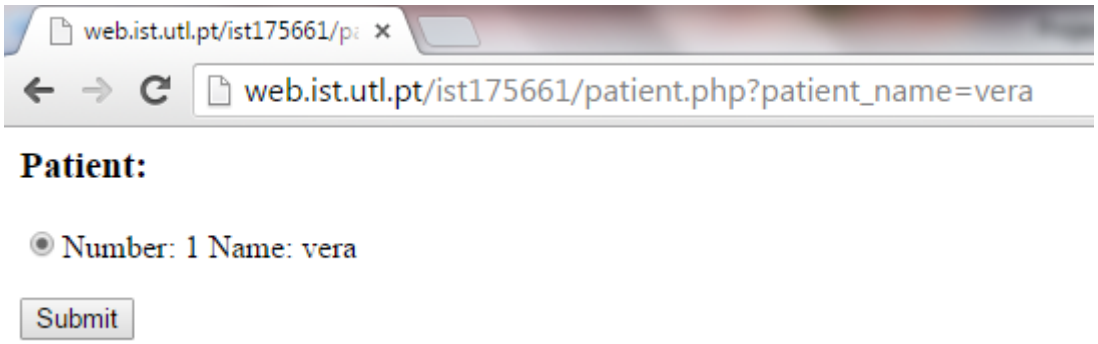
```
<html>
	<body>
		<form action="patient.php">
			<p>Patient name: <input type="text" name="patient_name"/></p>
			<p><input type="submit" value="Submit"/></p>
		</form>
	</body>
</html>
```



patient.php

```
<html>
	<body>
		<?php
			/*Ligação à base de dados*/
			$host = "db.ist.utl.pt";
			$user = "ist175661";
			$pass = "gejc9717";
			$dsn =	"mysql:host=$host;dbname=$user";
			try{
				$connection = new PDO($dsn, $user, $pass);
			}catch(PDOException $exception){
				echo("<p>Error ");
				echo($exception->getMessage());
				echo("</p>");
				exit();
			}

			$patientname = $_REQUEST['patient_name'];

			/*Query*/

			$sql="SELECT number, name
				FROM Patient
				WHERE name like '%$patientname%'
				ORDER BY number";

			$result = $connection->query($sql);

			if ($result == FALSE){
				$info = $connection->errorInfo();
				echo("<p>Error: {$info[2]}</p>");
				exit();
			}
			$nrows=$result->rowCount();
			if($nrows <> 0){

			  /*Escolha do paciente*/
			  echo("<h3>Patient:</h3>");
			  echo("<form action=\"devices_pan.php\">");
			  foreach($result as $row){
				echo("<input type=\"radio\" name=\"patient_name\"
value=\"$row[number]\" />Number: $row[number] Name: $row[name]<br/>");
			  }
			  echo("<p><input type=\"submit\"
```

```
value=\"Submit\"/></p></form>");
                        }else{
                         echo("<p>Patient not found!</p>");
                        }
                ?>

        </body>
</html>
```



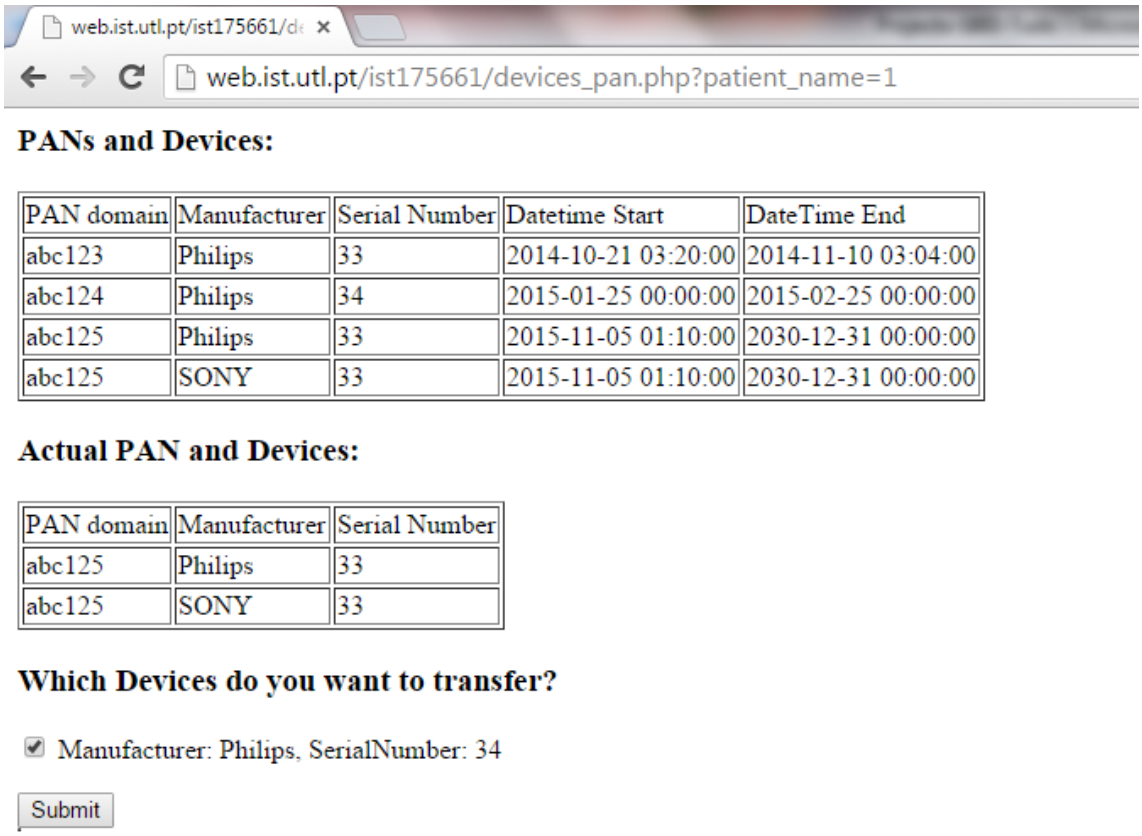
devices_pan.php

```
<html>
        <body>
                <?php
                        /*Ligação à base de dados*/
                        $host = "db.ist.utl.pt";
                        $user = "ist175661";
                        $pass = "gejc9717";
                        $dsn =          "mysql:host=$host;dbname=$user";
                        try{
                                $connection = new PDO($dsn, $user, $pass);
                        }catch(PDOException $exception){
                                echo("<p>Error ");
                                echo($exception->getMessage());
                                echo("</p>");
                                exit();
                        }

                        $patientnumber=$_REQUEST['patient_name'];

                        /*Query*/

                        $sql="SELECT c.pan, c.manuf, c.snum, c.start, c.end
                                        FROM Patient as p, Wears as w, Connects as c
                                        WHERE w.patient = '$patientnumber' AND w.pan = c.pan AND
p.number = w.patient
                                                AND c.start >= w.start AND c.end <= w.end
                                        ORDER BY c.pan,c.manuf,c.snum";
```

```
                $result = $connection->query($sql);

                if ($result == FALSE){
                        $info = $connection->errorInfo();
                        echo("<p>Error: {$info[2]}</p>");
                        exit();
                }

                /*Display dos Devices*/

                echo("<h3>PANs and Devices:</h3>");
                echo("<table border=\"1\">");
                echo("<tr><td>PAN domain<td>Manufacturer</td><td>Serial
Number</td><td>Datetime Start</td><td>DateTime End</td></tr>");
                foreach($result as $row){
                        echo("<tr><td>");
                        echo($row['pan']);
                        echo("</td><td>");
                        echo($row['manuf']);
                echo("</td><td>");
                        echo($row['snum']);
                        echo("</td><td>");
                        echo($row['start']);
                        echo("</td><td>");
                   echo($row['end']);
                        echo("</td></tr>");
                }
                echo("</table>");

                /*Query da penultima pan e da ultima pan*/
                /*Testada*/
                $pan_actual_sql = "select c.pan as PAN, c.manuf as Manufacturer, c.snum
as SerialNumber
                                from Wears as w, Connects as c,Patient as p
                                where p.number = $patientnumber and p.number= w.patient
and w.pan = c.pan
                                        and c.start >= w.start and c.end <= w.end
                                        and c.end > current_timestamp";

                $pan_actual = $connection->query($pan_actual_sql);

                if ($pan_actual == FALSE){
                        $info = $connection->errorInfo();
                        echo("<p>Error: {$info[2]}</p>");
                        exit();
                }

                /*Testada*/
                $pan_anterior_sql = "select c.pan as PAN, c.manuf as Manufacturer, c.snum
as SerialNumber
                                        from Patient as p, Wears as w, Connects as c
```

```
                                        where p.number =$patientnumber and p.number =
w.patient and w.pan = c.pan
                                                and c.start >= w.start and c.end <= w.end
                                                and c.end <= current_timestamp
                                                and c.end >= all(select c.end
                                                        from Patient as p, Wears as w,
Connects as c
                                                        where p.number = $patientnumber
and p.number = w.patient and w.pan = c.pan
                                                                and c.start >= w.start and c.end
<= w.end
                                                                and c.end <=
current_timestamp)
                                                                and (c.manuf,c.snum) not
in(select c.manuf,c.snum
                                                from Patient as p,Wears as w, Connects as c
                                                where p.number = $patientnumber and p.number =
w.patient and w.pan = c.pan
                                                        and c.start >= w.start and c.end <=w.end
                                                        and c.end > current_timestamp)";



                        $pan_anterior = $connection->query($pan_anterior_sql);

                        if ($pan_anterior == FALSE){
                                $info = $connection->errorInfo();
                                echo("<p>Error: {$info[2]}</p>");
                                exit();
                        }
                        /*Display do penultimo e ultimo*/
                        echo("<h3>Actual PAN and Devices:</h3>");
                        echo("<table border=\"1\">");
                        echo("<tr><td>PAN domain</td><td>Manufacturer</td><td>Serial
Number</td></tr>");
                        foreach($pan_actual as $row){
                                echo("<tr><td>");
                                echo($row['PAN']);
                                echo("</td><td>");
                                echo($row['Manufacturer']);
                                echo("</td><td>");
                                echo($row['SerialNumber']);
                                echo("</td></tr>");
                                $pan_actualdomain=$row['PAN'];
                        }
                        echo("</table>");


                        echo("<h3>Which Devices do you want to transfer?</h3>");
                        echo("<table border=\"1\">");
```

```
/*                  echo("<tr><td>PAN domain</td><td>Manufacturer</td><td>Serial
Number</td></tr>");
                    foreach($pan_anterior as $row2){
                        echo("<tr><td>");
                        echo($row2['PAN']);
                        echo("</td><td>");
                        echo($row2['Manufacturer']);
                        echo("</td><td>");
                        echo($row2['SerialNumber']);
                        echo("</td></tr>");
                    }
                    echo("</table>");
                    /*Transferência de Devices*/

                    /*Transferência de Devices*/


                    $nrowsant=$pan_anterior->rowCount();
                    if($nrowsant <> 0){

                    echo("<form action=\"transfer.php\">");
                    foreach($pan_anterior as $row3){
                        $arr=array($row3['Manufacturer'], $row3['SerialNumber']);
                        $device = implode("§", $arr);
                        echo("<input type=\"checkbox\" name=\"device[]\"
value=\"$device\"> Manufacturer: $row3[Manufacturer], SerialNumber:
$row3[SerialNumber]<br/>");

                    }
                    echo("<input type=\"hidden\" name=\"pan\" value=\"$pan_actualdomain\"
/>");
                    echo("<p></p>");
                    echo("<input type=\"submit\" value = \"Submit\">");

                    echo("</form>");
                    }else{
                        echo("<p>No devices to transfer.</p>");
                    }


            ?>
        </body>
</html>
```



transfer.php

```
<html>
	<body>

<?php
		$host = "db.ist.utl.pt";
		$user = "ist175661";
		$pass = "gejc9717";
		$dsn = "mysql:host=$host;dbname=$user";

		try
		{
			$connection = new PDO($dsn, $user, $pass);
		}
		catch(PDOException $exception)
		{
			echo("<p>Error: ");
			echo($exception->getMessage());
			echo("</p>");
			exit();
		}

		$result = $_REQUEST['device'];
		$pan=$_REQUEST['pan'];

		echo("<h3>Devices inseridos:</h3>");
```

```
            foreach($result as $devices){

                $device = explode("§", $devices);
                echo("<p>Manufacturer: $device[0] e Serial Number: $device[1]</p>");
                /*$c_time_sql = "select current_timestamp";
                $c_time = $connection->query($c_time_sql);*/
                $periods_sql = "select * from Period where start = current_timestamp and
end = '2030-12-31 00:00:00'";
                $periods = $connection->query($periods_sql);
                $nrows = $periods->rowCount();
                if($nrows == 0){
                    $insert_period_sql = "insert into Period values(current_timestamp,
'2030-12-31 00:00:00')";
                    $insert_period = $connection->exec($insert_period_sql);
                }
                $insert_sql = "insert into Connects values(current_timestamp, '2030-12-31
00:00:00','$device[1]', '$device[0]', '$pan')";
                $insert = $connection->exec($insert_sql);
                 if ($insert == FALSE){
                $info = $connection->errorInfo();
                echo("<p>Error: {$info[2]}</p>");
                exit();
             }
            }

            $connection = NULL;
?>
    </body>
</html>
```