

**UNIVERSIDAD NACIONAL DE SAN AGUSTÍN**

**Facultad de Ingeniería, Producción y Servicios**

**Escuela profesional de Ingeniería de Sistemas**



**Primer año - Semestre B**

**Asignatura de Programación Web 1 - Laboratorio**

**INTEGRANTES**

ARIAS QUISPE, Jhonatan David

JARA MAMANI, Mariel Alisson

MESTAS ZEGARRA, Christian Raul

NOA CAMINO, Yenaro Joel

**TRABAJO:**

Banca

**INGENIERO**

CARLOS JOSE LUIS CORRALES DELGADO

**AREQUIPA 2024 – PERÚ**

# Trabajo Final - Primera Parte

Link del Repositorio: <https://github.com/ynoacamino/bancaFinal.git>

Link de Youtube: <https://youtu.be/7eAqYHjBfvI>

# Código

## 1. Perl:

```
1  #!perl/bin/perl.exe
2
3  # Recibe: nada
4  # Retorna:
5  # <status>
6  #     <movement>
7  #         <amount>cantidad</amount>
8  #         <type>tipo</type>
9  #         <date>fecha</date>
10 #     </movement>
11 #     <movement>
12 #         ...
13 #     </movement>
14 #     <balance>saldo</balance>
15 # </status>
16
17 use strict;
18 use warnings;
19 use CGI;
20 use CGI::Session;
21 use CGI::Cookie;
22 use DBI;
23
24 my $cgi = CGI->new;
25 $cgi->charset("UTF-8");
26
27 my %cookies = CGI::Cookie->fetch();
28 my $session_cookie = $cookies{"session_id_cliente"};
29
30 if ($session_cookie) {
31     my $session_id = $session_cookie->value();
32     my $session = CGI::Session->load($session_id);
33     my $card_id = $session->param("card_id");
34
35     my $u = "query";
36     my $p = "YR4AFJUC3nyRmasY";
37     my $dsn = "dbi:mysql:database=bancafinal;host=127.0.0.1";
38     my $dbh = DBI->connect($dsn, $u, $p);
39
40     my $sth = $dbh->prepare("SELECT 'tarjetas'.moneda, 'movimientos'.monto, 'movimientos'.tipo,
41                             'movimientos'.fecha
42                             FROM tarjetas, movimientos
43                             WHERE tarjetas.id = '$card_id'");
44     $sth->execute($account_id);
45
46     my @row = $sth->fetchrow_array;
47     my $currency = $row[0] eq "s" ? "S/." : "\$";
48     my $total = $row[1] * $row[2];
49     print $cgi->header("text/xml");
50     print "<status>\n";
51     while (@row = $sth->fetchrow_array) {
52         $total = $total + ($row[1] * $row[2]);
53         my $amount = $currency.$row[1];
54         my $type = $row[2] == 1 ? "DEPOSITO" : "RETIRO";
55         my $date = $row[3];
56         print<<BLOCK;
57         <movement>
58             <amount>$amount</amount>
```

```

58         <type>$type</type>
59         <date>$date</date>
60     </movement>
61 BLOCK
62     }
63     my $balance = $currency.$total;
64     print<<BLOCK;
65         <balance>$balance</balance>
66     </status>
67 BLOCK
68 }

```

---

```

1  #!perl/bin/perl.exe
2
3  # Recibe: user, password, type (cliente u operario)
4  # Retorna:
5  # <errors>
6  #     <error>
7  #         <element>elemento</element>
8  #         <message>mensaje</message>
9  #     </error>
10 #     <error>
11 #         ...
12 #     </error>
13 # </errors>
14 # o solo la creacion de una cookie si todo esta bien :>
15
16 use strict;
17 use warnings;
18 use CGI;
19 use CGI::Session;
20 use CGI::Cookie;
21 use DBI;
22
23 my $cgi = CGI->new;
24 $cgi->charset("UTF-8");
25 my $user = $cgi->param("user");
26 my $password = $cgi->param("password");
27 my $type = $cgi->param("type");
28 my $session_time = 600;
29
30 my $user_status = check_user($user);
31 my $password_status = check_password($password);
32 my $type_status = check_type($type);
33 my %errors = (user => $user_status, password => $password_status, type => $type_status);
34
35 login();
36
37 sub login {
38     if (!$user_status && !$password_status && !$type_status) {
39         my $u = "query";
40         my $p = "YR4AFJUC3nyRmasY";
41         my $dsn = "dbi:mysql:database=bancafinal;host=127.0.0.1";
42         my $dbh = DBI->connect($dsn, $u, $p);
43
44         my $query;
45         if ($type eq "cliente") {
46             $query = "SELECT 'cuentas'.'.id', 'clientes'.'.id', CONCAT('clientes'.'.nombres', ' ', 'clientes'.'.paterno',
47                 ' ', 'clientes'.'.materno')
48                 FROM cuentas, clientes
49                 WHERE cuentas.usuario = '$user' AND cuentas.clave = '$password' AND cuentas.cliente_id =
50                     clientes.id";
51         } else {

```

```

50     $query = "SELECT 'nombre' FROM operarios WHERE usuario='$user' AND clave='$password'";
51 }
52 my $sth = $dbh->prepare($query);
53 $sth->execute;
54
55 my @person = $sth->fetchrow_array;
56 if (@person) {
57     my $session = CGI::Session->new();
58     if ($type eq "cliente") {
59         $session->param("account_id", $person[0]);
60         $session->param("client_id", $person[1]);
61         $session->param("name", $person[2]);
62     } else {
63         $session->param("name", $person[0]);
64     }
65     $session->expire(time + $session_time);
66     $session->flush();
67
68     my $cookie = $cgi->cookie(-name => "session_id_$type",
69                             -value => $session->id(),
70                             -expires => time + $session_time,
71                             "-max-age" => time + $session_time);
72
73     $cookie->bake;
74
75     return;
76 } else {
77     $errors{login} = "El usuario y la clave no coinciden.";
78 }
79
80 print_errors();
81 }
82
83 sub check_user {
84     my $card_number = $_[0];
85     my $length = length($card_number);
86     if (!$card_number || $length == 0) {
87         return "Ingrese un usuario.";
88     }
89     if ($length > 30) {
90         return "Usuario muy largo.";
91     }
92 }
93
94 sub check_password {
95     my $password = $_[0];
96     my $length = length($password);
97     if (!$password || $length == 0) {
98         return "Ingrese una clave.";
99     }
100     if ($length > 30) {
101         return "Clave muy larga.";
102     }
103 }
104
105 sub check_type {
106     my $type = $_[0];
107     if ($type ne "cliente" && $type ne "operario") {
108         return "Tipo no valido.";
109     }
110 }
111
112 sub print_errors {
113     print $cgi->header("text/xml");

```

```

114 print "<errors>\n";
115 for my $key (keys %errors) {
116     if ($errors{$key}) {
117         print<<XML;
118         <error>
119             <element>$key</element>
120             <message>$errors{$key}</message>
121         </error>
122 XML
123     }
124 }
125 print "</errors>\n";
126 }

```

---

```

1  #!/perl/bin/perl.exe
2
3  # Recibe: card_id, type (d o w), amount
4  # Retorna:
5  # <errors>
6  #     <error>
7  #         <element>elemento</element>
8  #         <message>mensaje</message>
9  #     </error>
10 #     <error>
11 #         ...
12 #     </error>
13 # </errors>
14 # o nada si todo esta bien :>
15
16 use strict;
17 use warnings;
18 use CGI;
19 use CGI::Session;
20 use CGI::Cookie;
21 use DBI;
22
23 my $cgi = CGI->new;
24 $cgi->charset("UTF-8");
25 my $card_id = $cgi->param("card_id");
26 my $type = $cgi->param("type");
27 my $amount = $cgi->param("amount");
28
29 my %cookies = CGI::Cookie->fetch();
30 my $session_cookie = $cookies{"session_id_cliente"};
31
32 my $card_id_status = check_card_id($card_id);
33 my $type_status = check_type($type);
34 my $amount_status = check_amount($amount);
35
36 if ($session_cookie) {
37     my $session_id = $session_cookie->value();
38     my $session = CGI::Session->load($session_id);
39     $account_id = $session->param("account_id");
40
41     transaction();
42 }
43
44 sub transaction {
45     if (!card_id_status && !$amount_status && !$type_status) {
46         my $u = "query";
47         my $p = "YR4AFJUC3nyRmasY";
48         my $dsn = "dbi:mysql:database=bancafinal;host=127.0.0.1";
49         my $dbh = DBI->connect($dsn, $u, $p);

```

```

50
51     my $type_num = $type eq "d" ? 1 : -1;
52
53     my $sth = $dbh->prepare("INSERT INTO movimientos (tarjeta_id, monto, tipo) VALUES ($card_id, $amount,
54         $type_num)");
55     $sth->execute;
56     print $cgi->header("text/xml");
57
58     return;
59 }
60
61 print_errors();
62 }
63
64 sub check_card_id {
65     my $card_id = $_[0];
66     my $length = length($card_id);
67     if (!$card_id || $length == 0) {
68         return "Tarjeta no valida.";
69     }
70 }
71
72 sub check_amount {
73     my $amount = $_[0];
74     if (!$amount) {
75         return "Ingrese una cantidad.";
76     }
77     if ($amount !~ /\d+$/ || $amount < 0) {
78         return "Cantidad no valida.";
79     }
80     if ($type eq "w") {
81         my $sth = $dbh->prepare("SELECT SUM(monto*tipo) FROM movimientos WHERE tarjeta_id = $card_id");
82         $sth->execute;
83         my @sum = $sth->fetchrow_array;
84         my $balance = $sum[0] || 0;
85         if ($balance - $amount < 0) {
86             return "Balance negativo."
87         }
88     }
89 }
90
91 sub check_type {
92     my $type = $_[0];
93     if (!$type) {
94         return "Marque un tipo";
95     }
96     if ($type !~ /^[dw]$/) {
97         return "Tipo no valido".
98     }
99 }
100
101 sub print_errors {
102     print $cgi->header("text/xml");
103     print "<errors>\n";
104     for my $key (keys %errors) {
105         if ($errors{$key}) {
106             print<<XML;
107             <error>
108                 <element>$key</element>
109                 <message>$errors{$key}</message>
110             </error>
111             XML
112         }
113     }
114 }

```

```

113     print "</errors>\n";
114 }

```

---

```

1  #!perl/bin/perl.exe
2
3  # Recibe: dni, name, plastname, mlastname, bdate
4  # Retorna:
5  # <errors>
6  #     <error>
7  #         <element>elemento</element>
8  #         <message>mensaje</message>
9  #     </error>
10 #     <error>
11 #         ...
12 #     </error>
13 # </errors>
14 # o nada si todo esta bien :>
15
16 use strict;
17 use warnings;
18 use CGI;
19 use CGI::Session;
20 use CGI::Cookie;
21 use DBI;
22
23 my $cgi = CGI->new;
24 $cgi->charset("UTF-8");
25 my $dni = $cgi->param("dni");
26 my $name = $cgi->param("name");
27 my $plastname = $cgi->param("plastname");
28 my $mlastname = $cgi->param("mlastname");
29 my $bdate = $cgi->param("bdate");
30
31 my $dni_status = check_DNI($dni);
32 my $name_status = check_name($name);
33 my $plastname_status = check_plastname($plastname);
34 my $mlastname_status = check_mlastname($mlastname);
35 my $bdate_status = check_bdate($bdate);
36
37 register();
38
39 sub register {
40     if (!$dni_status && !$name_status && !$plastname_status && !$mlastname_status && !$bdate_status) {
41         my $u = "query";
42         my $p = "YR4AFJUC3nyRmasY";
43         my $dsn = "dbi:mysql:database=bancafinal;host=127.0.0.1";
44         my $dbh = DBI->connect($dsn, $u, $p);
45
46         my $sth = $dbh->prepare("INSERT INTO clientes (dni, nombres, paterno, materno, nacimiento) VALUES ($dni,
47             $name, $plastname, $mlastname, $bdate)");
48         $sth->execute();
49         print $cgi->header("text/xml");
50
51         return;
52     }
53
54     print_errors();
55 }
56
57 sub check_DNI {
58     my $dni = $_[0];
59     if (!$dni) {
60         return "Ingresa un DNI.";
61     }
62 }

```



```

60     }
61     if ($dni !~ /\d{8}$/) {
62         return "DNI no valido.";
63     }
64 }
65
66 sub check_name {
67     my $name = $_[0];
68     if (!$name) {
69         return "Ingrese un nombre.";
70     }
71     if (length($name) > 40) {
72         return "Nombre muy largo.";
73     }
74     if ($name !~ /\w{1,40}$/) {
75         return "Nombre no valido.";
76     }
77 }
78
79 sub check_plastname {
80     my $plastname = $_[0];
81     my $length = length($plastname);
82     if (!$plastname) {
83         return "Ingrese un apellido paterno.";
84     }
85     if ($length > 30) {
86         return "Apellido paterno muy largo.";
87     }
88     if ($plastname !~ /\w{1,30}$/) {
89         return "Apellido paterno no valido.";
90     }
91 }
92
93 sub check_mlastname {
94     my $mlastname = $_[0];
95     my $length = length($mlastname);
96     if (!$mlastname) {
97         return "Ingrese un apellido materno.";
98     }
99     if ($length > 30) {
100         return "Apellido materno muy largo.";
101     }
102     if ($mlastname !~ /\w{1,30}$/) {
103         return "Apellido materno no valido.";
104     }
105 }
106
107 sub check_bdate {
108     my $bdate = $_[0];
109     if (!$bdate) {
110         return "Ingrese una fecha de nacimiento.";
111     }
112     if ($bdate !~ /\d{4}\-\d{2}\-\d{2}$/) {
113         return "Fecha de nacimiento no valida.";
114     }
115 }
116
117 sub print_errors {
118     print $cgi->header("text/xml");
119     print "<errors>\n";
120     for my $key (keys %errors) {
121         if ($errors{$key}) {
122             print<<XML;
123             <error>

```

```

124     <element>$key</element>
125     <message>$errors{$key}</message>
126 </error>
127 XML
128     }
129 }
130 print "</errors>\n";
131 }

```

---

```

1  #!perl/bin/perl.exe
2
3  # Recibe: user, password, dni
4  # Retorna:
5  # <errors>
6  #     <error>
7  #         <element>elemento</element>
8  #         <message>mensaje</message>
9  #     </error>
10 #     <error>
11 #         ...
12 #     </error>
13 # </errors>
14 # o nada si todo esta bien :>
15
16 use strict;
17 use warnings;
18 use CGI;
19 use CGI::Session;
20 use CGI::Cookie;
21 use DBI;
22
23 my $cgi = CGI->new;
24 $cgi->charset("UTF-8");
25 my $user = $cgi->param("user");
26 my $password = $cgi->param("password");
27 my $dni = $cgi->param("dni");
28
29 our $client_id;
30 my $user_status = check_user($user);
31 my $password_status = check_password($password);
32 my $dni_status = check_DNI($dni);
33
34 register();
35
36 sub register {
37     if (!$user_status && !$password_status && !$dni_status) {
38         my $u = "query";
39         my $p = "YR4AFJUC3nyRmasY";
40         my $dsn = "dbi:mysql:database=bancafinal;host=127.0.0.1";
41         my $dbh = DBI->connect($dsn, $u, $p);
42
43         my $sth = $dbh->prepare("INSERT INTO cuentas (usuario, clave, cliente_id) VALUES ($user, $password,
44             $client_id)");
45         $sth->execute;
46         print $cgi->header("text/xml");
47
48         return;
49     }
50
51     print_errors();
52 }
53
54 sub check_user {

```

```

54     my $card_number = $_[0];
55     my $length = length($card_number);
56     if (!$card_number || $length == 0) {
57         return "Ingreso un usuario.";
58     }
59     if ($length > 30) {
60         return "Usuario muy largo.";
61     }
62 }
63
64 sub check_password {
65     my $password = $_[0];
66     my $length = length($password);
67     if (!$password || $length == 0) {
68         return "Ingreso una clave.";
69     }
70     if ($length > 30) {
71         return "Clave muy larga.";
72     }
73 }
74
75 sub check_DNI {
76     my $dni = $_[0];
77     if (!$dni) {
78         return "Ingreso un DNI.";
79     }
80     if ($dni !~ /\d{8}/) {
81         return "DNI no valido.";
82     }
83     my $sth = $dbh->prepare("SELECT 'id' FROM clientes WHERE dni = $dni");
84     $sth->execute;
85     my @row = $sth->fetchrow_array;
86     if (!@row) {
87         return "Cliente no existente.";
88     }
89     $client_id = $row[0];
90 }
91
92 sub print_errors {
93     print $cgi->header("text/xml");
94     print "<errors>\n";
95     for my $key (keys %errors) {
96         if ($errors{$key}) {
97             print<<XML;
98             <error>
99                 <element>$key</element>
100                 <message>$errors{$key}</message>
101             </error>
102 XML
103         }
104     }
105     print "</errors>\n";
106 }

```

---

```

1  #!perl/bin/perl.exe
2
3  # Recibe: card_number, password, currency (s o d), dni
4  # Retorna:
5  # <errors>
6  #     <error>
7  #         <element>elemento</element>
8  #         <message>mensaje</message>
9  #     </error>

```

```

10 # <error>
11 # ...
12 # </error>
13 # </errors>
14 # o nada si todo esta bien :>
15
16 use strict;
17 use warnings;
18 use CGI;
19 use CGI::Session;
20 use CGI::Cookie;
21 use DBI;
22 use DateTime;
23
24 my $cgi = CGI->new;
25 $cgi->charset("UTF-8");
26 my $card_number = $cgi->param("card_number");
27 my $password = $cgi->param("password");
28 my $currency = $cgi->param("currency");
29 my $dni = $cgi->param("dni");
30 my $current_date = DateTime->now;
31 my $expire_date = $current_date->add(years => 7)->ymd;
32
33 our $account_id;
34 my $card_number_status = check_card_number($card_number);
35 my $password_status = check_password($password);
36 my $currency_status = check_currency($currency);
37 my $dni_status = check_DNI($dni);
38
39 register();
40
41 sub register {
42     if (!$card_number_status && !$password_status && !$currency_status && !$dni_status) {
43         my $u = "query";
44         my $p = "YR4AFJUC3nyRmasY";
45         my $dsn = "dbi:mysql:database=bancafinal;host=127.0.0.1";
46         my $dbh = DBI->connect($dsn, $u, $p);
47
48         my $sth = $dbh->prepare("INSERT INTO tarjetas (numero, clave, vencimiento, cuenta_id) VALUES (?, ?, ?)");
49         $sth->execute($number, $password, $expire_date);
50         print $cgi->header("text/xml");
51
52         return;
53     }
54
55     print_errors();
56 }
57
58 sub check_card_number {
59     my $card_number = $_[0];
60     if (!$card_number) {
61         return "Ingrese un número de tarjeta.";
62     }
63     if ($card_number !~ /\d{16}/) {
64         return "Número de tarjeta no valido.";
65     }
66 }
67
68 sub check_password {
69     my $password = $_[0];
70     if (!$password) {
71         return "Ingrese una clave.";
72     }
73     if (length($password) > 30) {

```

```

74     return "Clave muy larga.";
75 }
76 if (length($password) < 8) {
77     return "Clave muy corta.";
78 }
79 }
80
81 sub check_currency {
82     my $currency = $_[0];
83     if (!$currency) {
84         return "Marque una moneda.";
85     }
86     if ($currency !~ /^[sd]$/) {
87         return "Moneda no valida.";
88     }
89 }
90
91 sub check_DNI {
92     my $dni = $_[0];
93     if (!$dni) {
94         return "Ingresa un DNI.";
95     }
96     if ($dni !~ /\d{8}$/) {
97         return "DNI no valido.";
98     }
99     my $sth = $dbh->prepare("SELECT 'cuenta'. 'id'
100                             FROM cuentas, clientes
101                             WHERE clientes.dni = '$dni' AND cuentas.cliente_id = 'clientes_id'");
102     $sth->execute;
103     my @row = $sth->fetchrow_array;
104     if (!@row) {
105         return "Cliente no existente."
106     }
107     $client_id = $row[0];
108 }
109
110 sub print_errors {
111     print $cgi->header("text/xml");
112     print "<errors>\n";
113     for my $key (keys %errors) {
114         if ($errors{$key}) {
115             print<<XML;
116             <error>
117                 <element>$key</element>
118                 <message>$errors{$key}</message>
119             </error>
120 XML
121         }
122     }
123     print "</errors>\n";
124 }

```

---

```

1  #!perl/bin/perl.exe
2
3  # Recibe: action (check [revisa si la sesion esta activa], close [cierra la sesion]), type (cliente u operario)
4  # Retorna:
5  # <session>
6  #     <logged_in>0 o 1</logged_in>
7  #     <name>nombre</name>
8  #     <expire_time>tiempo restante</expire_time>
9  # </session>
10
11 use strict;

```

```

12 use warnings;
13 use CGI;
14 use CGI::Session;
15 use CGI::Cookie;
16
17 my $cgi = CGI->new;
18 $cgi->charset("UTF-8");
19 my $action = $cgi->param("action");
20 my $type = $cgi->param("type");
21
22 my %cookies = CGI::Cookie->fetch();
23 my $session_cookie = $cookies{"session_id_$type"};
24
25 print ($cgi->header("text/xml"));
26 print "<session>\n";
27 my $action = $cgi->param("action");
28 if ($action eq "check") {
29     if ($session_cookie) {
30         my $session_id = $session_cookie->value();
31         my $session = CGI::Session->load($session_id);
32         my $name = $session->param("name");
33         my $expire_time = $session->expires() - time;
34         print<<XML;
35         <logged_in>1</logged_in>
36         <name>$name</name>
37         <expire_time>$expire_time</expire_time>
38 XML
39     } else {
40         print "<logged_in>0</logged_in>\n";
41     }
42 } elsif ($action eq "close") {
43     print "<logged_in>0</logged_in>\n";
44     if ($session_cookie) {
45         my $session_id = $session_cookie->value();
46         my $session = CGI::Session->load($session_id);
47         $session_cookie->expires("-1h");
48         $session->delete();
49         $session->flush();
50     }
51 }
52 print "</session>";

```

---

## 2. SQL:

```
1  -- phpMyAdmin SQL Dump
2  -- version 5.2.1
3  -- https://www.phpmyadmin.net/
4  --
5  -- Servidor: 127.0.0.1
6  -- Tiempo de generación: 17-01-2024 a las 04:43:38
7  -- Versión del servidor: 10.4.28-MariaDB
8  -- Versión de PHP: 8.2.4
9
10 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
11 START TRANSACTION;
12 SET time_zone = "+00:00";
13
14
15 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
16 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
17 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
18 /*!40101 SET NAMES utf8mb4 */;
19
20 --
21 -- Base de datos: 'bancafinal'
22 --
23
24 -----
25
26 --
27 -- Estructura de tabla para la tabla 'clientes'
28 --
29
30 CREATE TABLE `clientes` (
31   `id` int(11) NOT NULL,
32   `dni` varchar(8) NOT NULL,
33   `nombres` varchar(40) NOT NULL,
34   `paterno` varchar(30) NOT NULL,
35   `materno` varchar(30) NOT NULL,
36   `nacimiento` date NOT NULL,
37   `creado` timestamp NOT NULL DEFAULT current_timestamp(),
38   `estado` tinyint(1) NOT NULL DEFAULT 1
39 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
40
41 --
42 -- Volcado de datos para la tabla 'clientes'
43 --
44
45 INSERT INTO `clientes` (`id`, `dni`, `nombres`, `paterno`, `materno`, `nacimiento`, `creado`, `estado`) VALUES
46 (1, '54587912', 'Christian Raul', 'Mestas', 'Zegarra', '2006-04-17', '2024-01-17 02:19:50', 1);
47
48 -----
49
50 --
51 -- Estructura de tabla para la tabla 'cuentas'
52 --
53
54 CREATE TABLE `cuentas` (
55   `id` int(11) NOT NULL,
56   `usuario` varchar(30) NOT NULL,
57   `clave` varchar(30) NOT NULL,
58   `creado` timestamp NOT NULL DEFAULT current_timestamp(),
59   `estado` tinyint(1) NOT NULL DEFAULT 1,
60   `cliente_id` int(11) NOT NULL
61 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
```

```

62
63 --
64 -- Volcado de datos para la tabla 'cuentas'
65 --
66
67 INSERT INTO 'cuentas' ('id', 'usuario', 'clave', 'creado', 'estado', 'cliente_id') VALUES
68 (1, 'cmestasz', '123456789', '2024-01-17 02:20:14', 1, 1);
69
70 -----
71
72 --
73 -- Estructura de tabla para la tabla 'movimientos'
74 --
75
76 CREATE TABLE 'movimientos' (
77     'id' int(11) NOT NULL,
78     'tarjeta_id' int(11) NOT NULL,
79     'monto' float NOT NULL,
80     'tipo' tinyint(4) NOT NULL,
81     'fecha' timestamp NOT NULL DEFAULT current_timestamp()
82 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
83
84 --
85 -- Volcado de datos para la tabla 'movimientos'
86 --
87
88 INSERT INTO 'movimientos' ('id', 'tarjeta_id', 'monto', 'tipo', 'fecha') VALUES
89 (3, 1, 50000, 1, '2024-01-17 02:21:50'),
90 (5, 1, 25000, -1, '2024-01-17 02:23:23');
91
92 -----
93
94 --
95 -- Estructura de tabla para la tabla 'operarios'
96 --
97
98 CREATE TABLE 'operarios' (
99     'id' int(11) NOT NULL,
100     'nombre' varchar(30) NOT NULL,
101     'usuario' varchar(30) NOT NULL,
102     'clave' varchar(30) NOT NULL,
103     'creado' timestamp NOT NULL DEFAULT current_timestamp(),
104     'estado' tinyint(1) NOT NULL DEFAULT 1
105 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
106
107 -----
108
109 --
110 -- Estructura de tabla para la tabla 'tarjetas'
111 --
112
113 CREATE TABLE 'tarjetas' (
114     'id' int(11) NOT NULL,
115     'numero' varchar(16) NOT NULL,
116     'clave' varchar(30) NOT NULL,
117     'creado' timestamp NOT NULL DEFAULT current_timestamp(),
118     'vencimiento' datetime NOT NULL,
119     'sellado' tinyint(1) NOT NULL DEFAULT 1,
120     'estado' tinyint(1) NOT NULL DEFAULT 1,
121     'moneda' varchar(1) NOT NULL,
122     'cuenta_id' int(11) NOT NULL
123 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;
124
125 --

```



```

126 -- Volcado de datos para la tabla 'tarjetas'
127 --
128
129 INSERT INTO 'tarjetas' ('id', 'numero', 'clave', 'creado', 'vencimiento', 'sellado', 'estado', 'moneda',
130     'cuenta_id') VALUES
131 (1, '5565894512347787', '123456789', '2024-01-17 02:20:32', '2024-01-17 03:20:19', 1, 1, 's', 1);
132 --
133 -- Índices para tablas volcadas
134 --
135
136 --
137 -- Índices de la tabla 'clientes'
138 --
139 ALTER TABLE 'clientes'
140     ADD PRIMARY KEY ('id'),
141     ADD UNIQUE KEY 'dni' ('dni');
142
143 --
144 -- Índices de la tabla 'cuentas'
145 --
146 ALTER TABLE 'cuentas'
147     ADD PRIMARY KEY ('id'),
148     ADD KEY 'cliente_id' ('cliente_id') USING BTREE;
149
150 --
151 -- Índices de la tabla 'movimientos'
152 --
153 ALTER TABLE 'movimientos'
154     ADD PRIMARY KEY ('id'),
155     ADD KEY 'tarjeta_id' ('tarjeta_id') USING BTREE;
156
157 --
158 -- Índices de la tabla 'operarios'
159 --
160 ALTER TABLE 'operarios'
161     ADD PRIMARY KEY ('id');
162
163 --
164 -- Índices de la tabla 'tarjetas'
165 --
166 ALTER TABLE 'tarjetas'
167     ADD PRIMARY KEY ('id'),
168     ADD UNIQUE KEY 'numero' ('numero'),
169     ADD KEY 'cuenta_id' ('cuenta_id') USING BTREE;
170
171 --
172 -- AUTO_INCREMENT de las tablas volcadas
173 --
174
175 --
176 -- AUTO_INCREMENT de la tabla 'clientes'
177 --
178 ALTER TABLE 'clientes'
179     MODIFY 'id' int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
180
181 --
182 -- AUTO_INCREMENT de la tabla 'cuentas'
183 --
184 ALTER TABLE 'cuentas'
185     MODIFY 'id' int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
186
187 --
188 -- AUTO_INCREMENT de la tabla 'movimientos'

```

```

189 --
190 ALTER TABLE 'movimientos'
191     MODIFY 'id' int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6;
192
193 --
194 -- AUTO_INCREMENT de la tabla 'operarios'
195 --
196 ALTER TABLE 'operarios'
197     MODIFY 'id' int(11) NOT NULL AUTO_INCREMENT;
198
199 --
200 -- AUTO_INCREMENT de la tabla 'tarjetas'
201 --
202 ALTER TABLE 'tarjetas'
203     MODIFY 'id' int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=2;
204
205 --
206 -- Restricciones para tablas volcadas
207 --
208
209 --
210 -- Filtros para la tabla 'cuentas'
211 --
212 ALTER TABLE 'cuentas'
213     ADD CONSTRAINT 'cliente_id' FOREIGN KEY ('cliente_id') REFERENCES 'clientes' ('id') ON DELETE CASCADE ON UPDATE
        CASCADE;
214
215 --
216 -- Filtros para la tabla 'movimientos'
217 --
218 ALTER TABLE 'movimientos'
219     ADD CONSTRAINT 'movimientos_ibfk_1' FOREIGN KEY ('tarjeta_id') REFERENCES 'tarjetas' ('id') ON DELETE CASCADE ON
        UPDATE CASCADE;
220
221 --
222 -- Filtros para la tabla 'tarjetas'
223 --
224 ALTER TABLE 'tarjetas'
225     ADD CONSTRAINT 'tarjetas_ibfk_1' FOREIGN KEY ('cuenta_id') REFERENCES 'cuentas' ('id') ON DELETE CASCADE ON UPDATE
        CASCADE;
226 COMMIT;
227
228 /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
229 /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
230 /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
231
232 -- usuarios
233 CREATE USER 'query'@'%' IDENTIFIED BY 'YR4AFJUC3nyRmasY';
234 GRANT SELECT ON query.* TO 'query'@'%'
235 GRANT INSERT ON query.* TO 'query'@'%'

```

---

### 3. HTML:

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Estado de cuenta</title>
7     <link rel="stylesheet" href="./css/estilos_generales.css" />
8     <link rel="stylesheet" href="./css/menus_clientes.css" />
9     <script src="./scripts/utilidades.js"></script>
10    <script src="./scripts/menus_clientes.js"></script>
11    <script src="./scripts/estado_tarjeta.js"></script>
12 </head>
13
14 <body>
15     <div id="content" class="content">
16         <div>
17             <h4 id="timer">Su sesión concluirá en:</h4>
18             <input type="button" class="logout" onclick="closeSession()" value="Cerrar sesión" />
19         </div>
20         <a href="./index_clientes.html">Volver</a>
21     </div>
22     <div>
23         <h2>Estado de cuenta</h2>
24         <h3>Balance de cuenta:</h3>
25         <div id="balance"></div>
26         <h3>Últimos movimientos:</h3>
27         <div id="movements">
28             <div class="movement">
29                 <h5 style="font-size: 20px;">CANTIDAD</h5>
30                 <h5 style="font-size: 20px;">TIPO</h5>
31                 <h5 style="font-size: 20px;">FECHA</h5>
32             </div>
33         </div>
34     </div>
35 </body>
36
37 </html>
```

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Usuario</title>
7     <link rel="stylesheet" href="./css/estilos_generales.css" />
8     <link rel="stylesheet" href="./css/menus_clientes.css" />
9     <script src="./scripts/menus_clientes.js"></script>
10 </head>
11
12 <body>
13     <div id="content" class="content">
14         <div class="top">
15             <div id="welcome"></div>
16             <h4 id="timer">Su sesión concluirá en:</h4>
17             <input type="button" class="logout" onclick="closeSession()" value="Cerrar sesión" />
18         </div>
19         <div style="height: 5px;"></div>
20         <h2>Acceso rápido</h2>
21         <div style="height: 50px;"></div>
22         <a href="./movimiento_cliente.html">Realizar un movimiento</a>
```

```

23     <div style="height: 50px"></div>
24     <a href="./estado_tarjeta.html">Ver estado</a>
25     <div style="height: 50px"></div>
26 </div>
27 </body>
28
29 </html>

```

---

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Usuario</title>
7     <link rel="stylesheet" href="./css/estilos_generales.css" />
8     <link rel="stylesheet" href="./css/menus_operarios.css" />
9     <script src="./scripts/utilidades.js"></script>
10    <script src="./scripts/menus_operarios.js"></script>
11 </head>
12
13 <body>
14     <div id="content" class="content">
15         <div class="top">
16             <div id="welcome"></div>
17             <h4 id="timer">Su sesión concluirá en:</h4>
18             <input type="button" class="logout" onclick="closeSession()" value="Cerrar sesión" />
19         </div>
20         <div style="height: 5px"></div>
21         <h2>Acceso rápido</h2>
22         <div style="height: 50px"></div>
23         <a href="./registro_clientes.html">Ingresar nuevo cliente</a>
24         <div style="height: 50px"></div>
25         <a href="./registro_cuentas.html">Crear nueva cuenta</a>
26         <div style="height: 50px"></div>
27         <a href="./registro_tarjetas.html">Crear nueva tarjeta</a>
28         <div style="height: 50px"></div>
29     </div>
30 </body>
31
32 </html>

```

---

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Ingreso de Clientes</title>
7     <link rel="stylesheet" href="./css/estilos_generales.css" />
8     <link rel="stylesheet" href="./css/login_clientes.css" />
9     <script src="./scripts/utilidades.js"></script>
10    <script src="./scripts/login_clientes.js"></script>
11 </head>
12
13 <body>
14     <div id="content" class="content">
15         
16     </div>
17     <div class="form">
18         <h1>Ingreso de Clientes</h1>
19         <input type="text" id="number" class="inputText" placeholder="xxxx xxxx xxxx xxxx" required />
20         <input type="password" id="password" placeholder="Contraseña" class="inputText" />
21         <input type="button" class="submitButton" onclick="login()" value="Ingresar"/>

```

```
22     </div>
23 </body>
24
25 </html>
```

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Ingreso de Operarios</title>
7     <link rel="stylesheet" href="../css/estilos_generales.css" />
8     <link rel="stylesheet" href="../css/login_operarios.css" />
9     <script src="../scripts/utilidades.js"></script>
10    <script src="../scripts/login_operarios.js"></script>
11 </head>
12
13 <body>
14     <div id="content" class="content">
15         
16         <div style="height: 35px"></div>
17         <h1>Ingreso de Operarios</h1>
18         <div style="height: 35px"></div>
19         <input type="text" id="user" placeholder="Usuario" />
20         <div style="height: 35px"></div>
21         <input type="password" id="password" placeholder="Contraseña" />
22         <div style="height: 35px"></div>
23         <input type="button" id="login" class="login" onclick="login()" value="Ingresar" />
24         <div style="margin-block: 25px"></div>
25     </div>
26 </body>
27
28 </html>
```

```
1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Realizar un movimiento</title>
7     <link rel="stylesheet" href="../css/estilos_generales.css" />
8     <link rel="stylesheet" href="../css/menus_clientes.css" />
9     <script src="../scripts/utilidades.js"></script>
10    <script src="../scripts/menus_clientes.js"></script>
11    <script src="../scripts/movimiento_cliente.js"></script>
12 </head>
13
14 <body>
15     <div id="content" class="content">
16         <div class="top">
17             <div id="welcome"></div>
18             <h4 id="timer">Su sesión concluirá en:</h4>
19             <input type="button" class="logout" onclick="closeSession()" value="Cerrar sesión" />
20         </div>
21         <div style="height: 15px"></div>
22         <a href="../index_clientes.html">Volver</a>
23         <div style="height: 10px"></div>
24         <h2>Asistente de movimientos</h2>
25         <div style="height: 25px"></div>
26         <h3>Tipo de movimiento:</h3>
27         <div id="type" class="options">
28             <input type="radio" id="deposit" name="type" value="deposit" checked />
```

```

29     <label for="soles">Depósito</label>
30     <div style="width: 50px"></div>
31     <input type="radio" id="withdrawal" name="type" value="withdrawal" />
32     <label for="dollars">Retiro</label>
33 </div>
34 <div style="height: 45px"></div>
35 <input type="text" id="amount" placeholder="Cantidad" />
36 <div style="height: 25px"></div>
37 <input type="button" class="submit" onclick="doMovement()" value="Realizar movimiento" />
38 </div>
39 </body>
40
41 </html>

```

---

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Creación de clientes</title>
7     <link rel="stylesheet" href="./css/estilos_generales.css" />
8     <link rel="stylesheet" href="./css/menus_usuarios.css" />
9     <script src="./scripts/menus_usuarios.js"></script>
10    <script src="./scripts/registro_clientes.js"></script>
11 </head>
12
13 <body>
14     <div id="content" class="content">
15         <div class="top">
16             <div id="welcome"></div>
17             <h4 id="timer">Su sesión concluirá en:</h4>
18             <input type="button" class="logout" onclick="closeSession()" value="Cerrar sesión" />
19         </div>
20         <div style="height: 15px"></div>
21         <a href="./index_operarios.html">Volver</a>
22         <div style="height: 10px"></div>
23         <h2>Asistente de creación de clientes</h2>
24         <div style="height: 25px"></div>
25         <input type="text" id="dni" placeholder="DNI" />
26         <div style="height: 25px"></div>
27         <input type="text" id="names" placeholder="Nombres completos" />
28         <div style="height: 25px"></div>
29         <input type="text" id="plastname" placeholder="Apellido paterno" />
30         <div style="height: 25px"></div>
31         <input type="text" id="mlastname" placeholder="Apellido materno" />
32         <div style="height: 25px"></div>
33         <input type="date" id="bdate" placeholder="Fecha de nacimiento" />
34         <div style="height: 25px"></div>
35         <input type="button" class="submit" onclick="createClient()" value="Crear cliente" />
36     </div>
37 </body>
38
39 </html>

```

---

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Creación de cuentas</title>
7     <link rel="stylesheet" href="./css/estilos_generales.css" />
8     <link rel="stylesheet" href="./css/menus_usuarios.css" />

```

```

9      <script src="./scripts/utilidades.js"></script>
10     <script src="./scripts/menus_usuarios.js"></script>
11     <script src="./scripts/registro_cuentas.js"></script>
12 </head>
13
14 <body>
15     <div id="content" class="content">
16         <div class="top">
17             <div id="welcome"></div>
18             <h4 id="timer">Su sesión concluirá en:</h4>
19             <input type="button" class="logout" onclick="closeSession()" value="Cerrar sesión" />
20         </div>
21         <div style="height: 15px"></div>
22         <a href="./index_operarios.html">Volver</a>
23         <div style="height: 10px"></div>
24         <h2>Asistente de creación de cuentas</h2>
25         <div style="height: 25px"></div>
26         <input type="text" id="number" placeholder="Número de tarjeta" />
27         <div style="height: 25px"></div>
28         <input type="button" class="submit" onclick="redirect('./registro_tarjetas.html', [['fromaccounts', 1]])"
29             value="Creación de tarjetas" />
30         <div style="height: 25px"></div>
31         <h3>Tipo de moneda:</h3>
32         <div id="currency" class="options">
33             <input type="radio" id="soles" name="currency" value="s" checked />
34             <label for="soles">Soles</label>
35             <div style="width: 50px"></div>
36             <input type="radio" id="dollars" name="currency" value="d" />
37             <label for="dollars">Dólares</label>
38         </div>
39         <div style="height: 45px"></div>
40         <input type="text" id="dni" placeholder="DNI del cliente" />
41         <div style="height: 25px"></div>
42         <input type="button" class="submit" onclick="createAccount()" value="Crear cuenta" />
43     </div>
44 </body>
45
46 </html>

```

---

```

1 <!DOCTYPE html>
2 <html>
3
4 <head>
5     <meta charset="UTF-8" />
6     <title>Creación de tarjetas</title>
7     <link rel="stylesheet" href="./css/estilos_generales.css" />
8     <link rel="stylesheet" href="./css/menus_usuarios.css" />
9     <script src="./scripts/menus_usuarios.js"></script>
10    <script src="./scripts/registro_tarjetas.js"></script>
11 </head>
12
13 <body>
14     <div id="content" class="content">
15         <div class="top">
16             <div id="welcome"></div>
17             <h4 id="timer">Su sesión concluirá en:</h4>
18             <input type="button" class="logout" onclick="closeSession()" value="Cerrar sesión" />
19         </div>
20         <div style="height: 15px"></div>
21         <a href="./index_operarios.html">Volver</a>
22         <div style="height: 10px"></div>
23         <h2>Asistente de creación de cuentas</h2>
24         <div style="height: 25px"></div>

```

```
25     <input type="button" class="submit" onclick="generateNumber()" value="Generar número" />
26     <div style="height: 25px"></div>
27     <input type="text" id="number" placeholder="Número de tarjeta" />
28     <div style="height: 25px"></div>
29     <input type="button" class="submit" onclick="generatePassword()" value="Generar clave" />
30     <div style="height: 25px"></div>
31     <input type="text" id="password" placeholder="Clave" />
32     <div style="height: 25px"></div>
33     <input type="button" class="submit" onclick="createCard()" value="Crear tarjeta" />
34 </div>
35 </body>
36
37 </html>
```

---



## 4. CSS:

---

```
1  /* Body Style */
2  body {
3      font-family: sans-serif;
4      background: linear-gradient(to top right, #d9ebfd, #fef9de);
5      min-height: 100vh;
6      margin: 0;
7      padding: 0;
8  }
9
10 /* Content Style */
11
12 .top {
13     float: right;
14     display: flexbox;
15     width: max-content;
16     align-items: center;
17 }
18
19 /* Header Styles */
20
21 /* Titles */
22 h1 {
23     text-align: center;
24     font-size: 40px;
25     color: #002a8d;
26 }
27
28 /* Subtitles */
29 h2 {
30     text-align: left;
31     font-size: 35px;
32     color: #002a8d;
33 }
34
35 /* Messages */
36 h3 {
37     text-align: center;
38     font-size: 20px;
39     color: #002a8d;
40 }
41
42 .error {
43     color: #e30425;
44 }
45
46 /* Text */
47 h4 {
48     text-align: center;
49     font-size: 15px;
50     color: #202e44;
51 }
52
53 /* Text - Inline */
54 h5 {
55     flex: 1;
56     display: inline-block;
57     text-align: center;
58     font-size: 15px;
59     color: #202e44;
60 }
61
```

```

62  /* Links */
63  a {
64      text-align: left;
65      font-size: 20px;
66      text-decoration: none;
67      color: #202e44;
68  }
69
70  a:hover {
71      text-decoration: underline;
72  }

```

---

```

1  * {
2      box-sizing: border-box;
3  }
4
5  body {
6      display: flex;
7      flex-direction: column;
8      justify-content: start;
9      align-items: center;
10 }
11
12 .logo {
13     width: 125px;
14 }
15
16 .logo:hover {
17     cursor: pointer;
18 }
19
20 .content {
21     width: 100%;
22     max-width: 1200px;
23     display: flex;
24     justify-content: start;
25     align-items: center;
26 }
27
28 .inputText::placeholder {
29     color: #99a1ad;
30 }
31
32 .inputText:hover::placeholder {
33     color: #606c7f;
34     border-color: #0a47f0;
35 }
36
37 .inputText {
38     display: block;
39     margin: auto;
40     padding: 15px;
41     border-style: solid;
42     border-radius: 10px;
43     border-width: 1px;
44     border-color: #868f9e;
45     width: 100%;
46     font-size: 20px;
47     color: #202e44;
48     max-width: 400px;
49 }
50
51 .inputText:hover {

```

```

52     border-color: #0a47f0;
53 }
54
55 .inputText:focus {
56     border-color: #0a47f0;
57     outline: none;
58     border-width: 2px;
59 }
60
61 input[type="text"][class=error],
62 input[type="password"][class=error] {
63     border-color: #e30425;
64 }
65
66 /* Button Styles */
67 .submitButton {
68     display: block;
69     border: none;
70     border-radius: 15px;
71     background-color: #ff7800;
72     color: white;
73     margin: auto;
74     max-width: 400px;
75     width: 100%;
76     font-size: 1.2rem;
77     padding: 15px 20px;
78 }
79
80 .submitButton:hover {
81     background-color: #ff961f;
82     box-shadow: 0 4px 4px 0px #c8c8c8;
83     cursor: pointer;
84 }
85
86 .submitButton:active {
87     background-color: #e45e00;
88 }
89
90 .form {
91     display: flex;
92     flex-direction: column;
93     gap: 20px;
94 }

```

---

```

1  /* Logo Styles */
2  .logo {
3      width: 125px;
4  }
5
6  .logo:hover {
7      cursor: pointer;
8  }
9
10 /* Input Styles */
11 input[type="text"]::placeholder,
12 input[type="password"]::placeholder {
13     color: #99a1ad;
14 }
15
16 input[type="text"]:hover::placeholder,
17 input[type="password"]:hover::placeholder {
18     color: #606c7f;
19     border-color: #0a47f0;

```

```

20 }
21
22 input[type="text"],
23 input[type="password"] {
24     display: block;
25     margin: auto;
26     padding: 15px;
27     border-style: solid;
28     border-radius: 10px;
29     border-width: 1px;
30     border-color: #868f9e;
31     width: 900px;
32     font-size: 20px;
33     color: #202e44;
34 }
35
36 input[type="text"]:hover,
37 input[type="password"]:hover {
38     border-color: #0a47f0;
39 }
40
41 input[type="text"]:focus,
42 input[type="password"]:focus {
43     border-color: #0a47f0;
44     outline: none;
45     border-width: 2px;
46 }
47
48 input[type="text"][class=error],
49 input[type="password"][class=error] {
50     border-color: #e30425;
51 }
52
53 /* Button Styles */
54 input[type="button"] {
55     display: block;
56     border: none;
57     border-radius: 25px;
58     background-color: #ff7800;
59     color: white;
60     margin: auto;
61     width: 450px;
62     font-size: 20px;
63     padding: 15px;
64 }
65
66 input[type="button"]:hover {
67     background-color: #ff961f;
68     box-shadow: 0 4px 4px #c8c8c8;
69     cursor: pointer;
70 }
71
72 input[type="button"]:active {
73     background-color: #e45e00;
74 }

```

---

```

1  /* Input Styles */
2  * {
3      box-sizing: border-box;
4  }
5  body {
6      display: flex;
7      flex-direction: column;

```

```

8     justify-content: start;
9     align-items: center;
10 }
11
12 .content {
13     width: 100%;
14     max-width: 1200px;
15     display: flex;
16     justify-content: space-between;
17     align-items: center;
18     flex-direction: row-reverse;
19     padding: 0px 20px;
20 }
21
22 .options {
23     display: flex;
24     margin: auto;
25     justify-content: center;
26 }
27
28 /* Button Styles */
29 input[type="button"] {
30     margin: auto;
31     display: block;
32     border: none;
33     border-radius: 25px;
34     background-color: #ff7800;
35     color: white;
36 }
37
38 input[type="button"]:hover {
39     background-color: #ff961f;
40     box-shadow: 0 4px 4px 0px #c8c8c8;
41     cursor: pointer;
42 }
43
44 input[type="button"]:active {
45     background-color: #e45e00;
46 }
47
48 .logout {
49     width: 150px;
50     font-size: 15px;
51     padding: 5px;
52 }
53
54 .submit {
55     width: 225px;
56     font-size: 20px;
57     padding: 5px;
58 }
59
60 /* Division Styles */
61 .movement {
62     display: flex;
63     justify-content: center;
64 }

```

---

```

1  /* Input Styles */
2  input[type="text"]::placeholder,
3  input[type="password"]::placeholder,
4  input[type="date"]::placeholder {
5      color: #99a1ad;

```

```

6 }
7
8 input[type="text"]:hover::placeholder,
9 input[type="password"]:hover::placeholder,
10 input[type="date"]:hover::placeholder {
11     color: #606c7f;
12     border-color: #0a47f0;
13 }
14
15 input[type="text"],
16 input[type="password"],
17 input[type="date"] {
18     padding: 10px;
19     width: 1225px;
20     border-style: solid;
21     border-radius: 10px;
22     border-width: 1px;
23     border-color: #868f9e;
24     font-size: 15px;
25     color: #202e44;
26 }
27
28 input[type="text"]:hover,
29 input[type="password"]:hover,
30 input[type="date"]:hover {
31     border-color: #0a47f0;
32 }
33
34 input[type="text"]:focus,
35 input[type="password"]:focus,
36 input[type="date"]:focus {
37     border-color: #0a47f0;
38     outline: none;
39     border-width: 2px;
40 }
41
42 input[type="text"][class="error"],
43 input[type="password"][class="error"],
44 input[type="date"][class="error"] {
45     border-color: #e30425;
46 }
47
48 .options {
49     display: flex;
50     justify-content: center;
51 }
52
53 /* Button Styles */
54 input[type="button"] {
55     margin: auto;
56     display: block;
57     border: none;
58     border-radius: 25px;
59     background-color: #ff7800;
60     color: white;
61 }
62
63 input[type="button"]:hover {
64     background-color: #ff961f;
65     box-shadow: 0 4px 0px #c8c8c8;
66     cursor: pointer;
67 }
68
69 input[type="button"]:active {

```

```
70     background-color: #e45e00;
71 }
72
73 .logout {
74     width: 150px;
75     font-size: 15px;
76     padding: 5px;
77 }
78
79 .submit {
80     width: 225px;
81     font-size: 20px;
82     padding: 5px;
83 }
```

---

## 5. JavaScript:

---

```
1 addEventListener("load", getStatus);
2
3 function getStatus() {
4     const xhttp = new XMLHttpRequest();
5
6     xhttp.onreadystatechange = function () {
7         if (xhttp.readyState == 4 && xhttp.status == 200) {
8             const response = xhttp.responseXML;
9             const balance = response.getElementsByTagName("balance")[0].childNodes[0].nodeValue;
10            createBalanceElement(balance);
11
12            const movements = response.getElementsByTagName("movement");
13            for (let i = movements.length - 1; i >= 0; i--) {
14                let amount = movements[i].getElementsByTagName("amount")[0].childNodes[0].nodeValue;
15                let type = movements[i].getElementsByTagName("type")[0].childNodes[0].nodeValue;
16                let date = movements[i].getElementsByTagName("date")[0].childNodes[0].nodeValue;
17                createMovementElement(amount, type, date);
18            }
19        }
20    };
21
22    xhttp.open("POST", "./cgi-bin/estado_tarjeta.pl", true);
23    xhttp.send();
24 }
25
26 function createBalanceElement(balance) {
27     const balanceElement = document.createElement("h4");
28     balanceElement.innerHTML = balance;
29     document.getElementById("balance").appendChild(balanceElement);
30 }
31
32 function createMovementElement(amount, type, date) {
33     const movementElement = document.createElement("div");
34     movementElement.classList.add("movement");
35
36     const amountElement = document.createElement("h5");
37     const typeElement = document.createElement("h5");
38     const dateElement = document.createElement("h5");
39     amountElement.innerHTML = amount;
40     typeElement.innerHTML = type;
41     dateElement.innerHTML = date;
42
43     movementElement.appendChild(amountElement);
44     movementElement.appendChild(typeElement);
45     movementElement.appendChild(dateElement);
46
47     document.getElementById("movements").appendChild(movementElement);
48 }

```

---

```
1 function login() {
2     let xhttp = new XMLHttpRequest();
3     let number = document.getElementById("number");
4     let password = document.getElementById("password");
5     xhttp.onreadystatechange = function () {
6         if (xhttp.readyState == 4 && xhttp.status == 200) {
7             let response = xhttp.responseText;
8             if (response == "wrong" && !document.getElementById("error")) {
9                 let error = document.createElement("h3");
10                error.classList.add("error");
11                error.id = "error";

```



```

12         error.innerHTML = "El número de tarjeta o la contraseña son incorrectos. Verifíquelos y vuelva a
           intentar"
13         document.getElementById("content").appendChild(error);
14         number.classList.add(["error"]);
15         password.classList.add(["error"]);
16     } else if (response == "correct") {
17         window.location = "./index_clientes.html";
18     }
19 }
20 };
21 xhttp.open("POST", "./cgi-bin/login.pl", true);
22 xhttp.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded; charset=ISO-8859-1');
23 xhttp.send("number=" + number.value + "&password=" + password.value + "&type=cliente");
24 }

```

---

```

1 function login() {
2     fetch2(["user", "password"], ["type=operario"], "login.pl", function(response) {
3         const errors = response.getElementsByTagName("error");
4         if (errors.length != 0) {
5             createStatusElements(errors);
6         } else {
7             window.location = "./index_operarios.html";
8         }
9     })
10 }
11
12 function createStatusElements(errors) {
13     for (let i = 0; i < errors.length; i++) {
14         let element = errors[i].getElementsByTagName("element")[0].firstChild.nodeValue;
15         let message = errors[i].getElementsByTagName("message")[0].firstChild.nodeValue;
16         let statusElement = document.getElementById("status_" + element);
17         if (statusElement) {
18             statusElement.remove();
19         }
20         createElement(element, message);
21     }
22 }
23
24 function createElement(element, message) {
25     const statusElement = document.createElement("h3");
26     const placeElement = document.getElementById(element);
27
28     if (message != "Correcto") {
29         placeElement.classList.add("error");
30         statusElement.classList.add("error");
31     } else {
32         placeElement.classList.remove("error");
33     }
34
35     statusElement.id = "status_" + element;
36     statusElement.innerHTML = message;
37     document.getElementById("content").insertBefore(statusElement, placeElement);
38 }

```

---

```

1 onload = function () {
2     firstLog();
3     updateTimer();
4     setInterval(updateTimer, 1 * 1000);
5 };
6
7 function firstLog() {
8     request("check", function (response) {

```

```

9     let logged_in = response.getElementsByTagName("logged_in")[0].childNodes[0].nodeValue;
10    if (logged_in == "1") {
11        let welcome = document.createElement("h4");
12        let name = response.getElementsByTagName("name")[0].childNodes[0].nodeValue;
13        welcome.innerHTML = "Bienvenido, " + name;
14        document.getElementById("welcome").append(welcome);
15    } else {
16        redirectBack();
17    }
18    });
19 }
20
21 function updateTimer() {
22     request("check", function (response) {
23         let logged_in = response.getElementsByTagName("logged_in")[0].childNodes[0].nodeValue;
24         if (logged_in == "1") {
25             let timer = response.getElementsByTagName("expire_time")[0].childNodes[0].nodeValue;
26             document.getElementById("timer").innerHTML = "Su sesión concluirá en: " + timer + "s";
27         } else {
28             redirectBack();
29         }
30     });
31 }
32
33 function closeSession() {
34     request("close", function () {
35         redirectBack();
36     });
37 }
38
39 function request(action, callback) {
40     let xhttp = new XMLHttpRequest();
41     xhttp.onreadystatechange = function () {
42         if (xhttp.readyState == 4 && xhttp.status == 200) {
43             callback(xhttp.responseXML);
44         }
45     };
46     xhttp.open("POST", "./cgi-bin/sesion.pl?type=cliente&action=" + action, true);
47     xhttp.send();
48 }
49
50 function redirectBack() {
51     alert("Su sesión ha expirado");
52     window.location = "./login_clientes.html";
53 }
54
55 function createStatusElements(xhttp) {
56     const xmlresponse = xhttp.responseXML;
57     const errors = xmlresponse.getElementsByTagName("elem_status");
58     for (let i = 0; i < errors.length; i++) {
59         let element = errors[i].getElementsByTagName("element")[0].childNodes[0].nodeValue;
60         let status = errors[i].getElementsByTagName("status")[0].childNodes[0].nodeValue;
61         let statusElement = document.getElementById("status_" + element);
62         if (statusElement) {
63             statusElement.remove();
64         }
65         createElement(element, status);
66     }
67 }
68
69 function createElement(element, status) {
70     const statusElement = document.createElement("h3");
71     const placeElement = document.getElementById(element);
72

```

```

73     if (status !== "Correcto") {
74         placeElement.classList.add("error");
75         statusElement.classList.add("error");
76     } else {
77         placeElement.classList.remove("error");
78     }
79
80     statusElement.id = "status_" + element;
81     statusElement.innerHTML = status;
82     document.getElementById("content").insertBefore(statusElement, placeElement);
83 }

```

---

```

1  onload = function () {
2      firstLog();
3      updateTimer();
4      setInterval(updateTimer, 1 * 1000);
5  };
6
7  function firstLog() {
8      fetch2([], ["action=check", "type=operario", "sesion.pl", function(response) {
9          let logged_in = response.getElementsByTagName("logged_in")[0].childNodes[0].nodeValue;
10         if (logged_in === "1") {
11             let welcome = document.createElement("h4");
12             let name = response.getElementsByTagName("name")[0].childNodes[0].nodeValue;
13             welcome.innerHTML = "Bienvenido, " + name;
14             document.getElementById("welcome").append(welcome);
15         } else {
16             redirectBack();
17         }
18     }]);
19 }
20
21 function updateTimer() {
22     fetch2([], ["action=check", "type=operario", "sesion.pl", function(response) {
23         let logged_in = response.getElementsByTagName("logged_in")[0].childNodes[0].nodeValue;
24         if (logged_in === "1") {
25             let timer = response.getElementsByTagName("expire_time")[0].childNodes[0].nodeValue;
26             document.getElementById("timer").innerHTML = "Su sesión concluirá en: " + timer + "s";
27         } else {
28             redirectBack();
29         }
30     }]);
31 }
32
33 function closeSession() {
34     fetch2([], ["action=close", "type=operario", "sesion.pl", function(response) {
35         redirectBack();
36     }])
37 }
38
39 function redirectBack() {
40     alert("Su sesión ha expirado");
41     window.location = "../login_operarios.html";
42 }

```

---

```

1  function doMovement() {
2      const xhttp = new XMLHttpRequest();
3      const depositInput = document.getElementById("deposit");
4      const withdrawalInput = document.getElementById("withdrawal");
5      const typeInput = depositInput.checked ? depositInput : withdrawalInput;
6      const amountInput = document.getElementById("amount");
7

```

```

8     xhttp.onreadystatechange = function () {
9         if (xhttp.readyState == 4 && xhttp.status == 200) {
10             const response = xhttp.responseText;
11             if (response == "correct") {
12                 alert("Movimiento realizado correctamente");
13                 location = "./index_clientes.html";
14             } else {
15                 createStatusElements(xhttp);
16             }
17         }
18     };
19
20     xhttp.open("POST", "./cgi-bin/movimiento_cliente.pl", true);
21     xhttp.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded; charset=ISO-8859-1');
22     xhttp.send("type=" + typeInput.value + "&amount=" + amountInput.value);
23 }

```

---

```

1 function createClient() {
2     const xhttp = new XMLHttpRequest();
3     const dniInput = document.getElementById("dni");
4     const namesInput = document.getElementById("names");
5     const lastnameInput = document.getElementById("lastname");
6     const mlastnameInput = document.getElementById("mlastname");
7     const birthDateInput = document.getElementById("bdate");
8
9     xhttp.onreadystatechange = function () {
10         if (xhttp.readyState == 4 && xhttp.status == 200) {
11             const response = xhttp.responseText;
12             if (response == "correct") {
13                 alert("Cliente creado correctamente");
14                 location = "./index_usuarios.html";
15             } else {
16                 createStatusElements(xhttp);
17             }
18         }
19     };
20
21     xhttp.open("POST", "./cgi-bin/registro_clientes.pl", true);
22     xhttp.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded; charset=ISO-8859-1');
23     xhttp.send("dni=" + dniInput.value + "&names=" + namesInput.value + "&lastname=" + lastnameInput.value +
24         "&mlastname=" + mlastnameInput.value + "&bdate=" + birthDateInput.value);

```

---

```

1 addEventListener("load", checkFromCards);
2
3 function checkFromCards() {
4     const number = localStorage.getItem("number");
5     if (number) {
6         localStorage.removeItem("number");
7         document.getElementById("number").value = number;
8     }
9 }
10
11 function createAccount() {
12     const xhttp = new XMLHttpRequest();
13     const numberInput = document.getElementById("number");
14     const solesInput = document.getElementById("soles");
15     const dollarsInput = document.getElementById("dollars");
16     const currencyInput = solesInput.checked ? solesInput : dollarsInput;
17     const dniInput = document.getElementById("dni");
18
19     xhttp.onreadystatechange = function () {

```

```

20     if (xhttp.readyState == 4 && xhttp.status == 200) {
21         const response = xhttp.responseText;
22         if (response == "correct") {
23             alert("Cuenta creada correctamente");
24             location = "./index_usuarios.html";
25         } else {
26             createStatusElements(xhttp);
27         }
28     }
29 };
30
31 xhttp.open("POST", "./cgi-bin/registro_cuentas.pl", true);
32 xhttp.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded; charset=ISO-8859-1');
33 xhttp.send("number=" + numberInput.value + "&currency=" + currencyInput.value + "&dni=" + dniInput.value);
34 }

```

---

```

1 function createCard() {
2     const xhttp = new XMLHttpRequest();
3     const numberInput = document.getElementById("number");
4     const passwordInput = document.getElementById("password");
5
6     xhttp.onreadystatechange = function () {
7         if (xhttp.readyState == 4 && xhttp.status == 200) {
8             const response = xhttp.responseText;
9             if (response == "correct") {
10                 alert("Tarjeta creada correctamente");
11                 if (localStorage.getItem("fromaccounts")) {
12                     localStorage.setItem("number", numberInput.value);
13                     localStorage.removeItem("fromaccounts");
14                     location = "./cuentas.html";
15                 } else {
16                     location = "./index_usuarios.html";
17                 }
18             } else {
19                 createStatusElements(xhttp);
20             }
21         }
22     };
23
24     xhttp.open("POST", "./cgi-bin/registro_tarjetas.pl", true);
25     xhttp.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded; charset=ISO-8859-1');
26     xhttp.send("number=" + numberInput.value + "&password=" + passwordInput.value);
27 }

```

```

28
29 function generateNumber() {
30     let number = "";
31     for (let i = 0; i < 16; i++) {
32         number += Math.floor(Math.random() * 9);
33     }
34     document.getElementById("number").value = number;
35 }
36
37 function generatePassword() {
38     const chars = "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";
39     let password = "";
40     for (let i = 0; i < 10; i++) {
41         password += chars.charAt(Math.random() * chars.length);
42     }
43     document.getElementById("password").value = password;
44 }

```

---

```

1 function redirect(url, params) {

```

```

2     if (params) {
3         for (let i = 0; i < params.length; i++) {
4             localStorage.setItem(params[i][0], params[i][1]);
5         }
6     }
7     location = url;
8 }
9
10 // params son los id del formulario
11 // script es la ubicacion del archivo perl, tiene que terminar en .pl
12 // func siempre debe tener el parametro (response)
13 function fetch2(html_params, extra_params, script, func) {
14     let xhttp = new XMLHttpRequest();
15     let query = "";
16     for (let i = 0; i < html_params.length; i++) {
17         try {
18             let param_value = document.getElementById(html_params[i]).value;
19             query += html_params[i] + "=" + param_value + "&";
20         } catch (err) {
21             alert ("Error obteniendo " + html_params[i]);
22         }
23     }
24
25     for (let i = 0; i < extra_params.length; i++) {
26         query += extra_params[i] + "&";
27     }
28
29     xhttp.onreadystatechange = function () {
30         if (xhttp.readyState == 4 && xhttp.status == 200) {
31             let response = xhttp.responseXML || new Document();
32             func(response);
33         }
34     };
35
36     xhttp.open("POST", "./cgi-bin/" + script, true);
37     xhttp.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded; charset=ISO-8859-1');
38     xhttp.send(query);
39 }

```

---