

Laboratorio 3

Taller de Sistemas Computacionales

Equipo 14:

- Carlos Bugueño Rodríguez
- Oscar Duarte Marín
- Eduardo Ramirez Farías

Antes de empezar...	2
Edición de contribuyente:	2
Revocación de boletas	3
Eliminación de boletas	4
Login persistente	5

Antes de empezar...

Para poder trabajar bajo los datos generados en el laboratorio 2, fue necesario modificar el archivo `js/global.js`, en donde incluimos como variable de autorización el número de nuestro equipo.

```
1 |var global={
2 |    authorization : 14
3 |}
```

Edición de contribuyente:

Para la edición del contribuyente, se editó el archivo `queries/contribuyentes.js`, de tal forma que se agrega la función `patch`. Como base para agregar dicha función, se usó la función equivalente a boletas, presente en `queries/boletas.js`.

```
88     patch: function(contribuyente, data){
89         $.ajax(
90             {
91                 url : "http://67.205.134.51/contribuyentes/" + contribuyente + "/",
92                 data : JSON.stringify(data),
93                 contentType : "application/json; charset=UTF-8",
94                 dataType : "json",
95                 type : "PATCH",
96                 processData : false,
97                 timeout : 30000,
98                 beforeSend: function(xhr) {
99                     xhr.setRequestHeader('Authorization', global.authorization);
100                 },
101                 success : function(json) {
102                     console.log(json);
103                     alert("Contribuyente editado con éxito");
104                     location.reload();
105                 },
106                 error : function(xhr) {
107                     alert("Error en la consulta, ver consola para más detalles");
108                     console.log(xhr);
109                 },
110                 complete : function(xhr, status) {
111
112                 }
113             }
114         );
115     }
116 }
```

En `views/boletas.js`, se modifica el método `edit-contribuyente-button` para aplicar la función de modificación de contribuyente.

```
9      $('#edit-contribuyente-button').click(function(){
10          var data = {};
11          data.nombres = $('#contribuyente-name').val();
12          data.apellidos = $('#contribuyente-lastnames').val();
13          data.direccion = $('#contribuyente-address').val();
14          data.rut = $('#contribuyente-rut').val();
15          contribuyentes.patch(contribuyente,data);
16      });
17  }
```

`views.boletas.js`

Revocación de boletas

El archivo `queries/revocaciones.js` en su inicio estaba vacío. Para que sea funcional, se agregó el método `post` usando como base el respectivo `post` en `queries/contribuyentes.js` y `queries/boletas.js`.

```
1  var revocaciones = {
2      post: function(data){
3          $.ajax(
4              {
5                  url : "http://67.205.134.51/revocaciones/",
6                  data : JSON.stringify(data),
7                  contentType : "application/json; charset=UTF-8",
8                  dataType : "json",
9                  type : "POST",
10                 processData : false,
11                 timeout : 30000,
12                 beforeSend: function(xhr) {
13                     xhr.setRequestHeader('Authorization', global.authorization);
14                 },
15                 success : function(json) {
16                     console.log(json);
17                     alert("Boleta revocada con éxito");
18                     location.reload();
19                 },
20                 error : function(xhr) {
21                     alert("Error en la consulta, ver consola para más detalles");
22                     console.log(xhr);
23                 },
24                 complete : function(xhr, status) {
25
26                 }
27             }
28         );
29     }
30 }
31 }
```

`queries/revocaciones.js`

Luego, en `views/boletas.js` se modifica el método `confirm-revocation-button` con el código necesario para poder ejecutar la revocación de boleta.

```
36      $('#confirm-revocation-button').click(function(){
37          var contribuyente = window.sessionStorage.getItem("contribuyente");
38          var boleta = window.sessionStorage.getItem("selected_boleta");
39          var data = {};
40          data.contribuyente = contribuyente;
41          data.boleta = boleta;
42          data.razon = $('#revocation-reason').val();
43          revocaciones.post(data);
44      });
```

`views/boletas.js`

Eliminación de boletas

Para poder crear el método `delete` en el archivo `queries/boletas.js`, es necesario generar el código relacionado con esta función. Usando el método `post` de `queries/boletas.js` como base para el nuevo método.

```
87      delete: function(contribuyente, boleta){
88          $.ajax(
89              {
90                  url : "http://67.205.134.51/contribuyentes/" + contribuyente + "/boletas/" + boleta + "/",
91                  contentType : "application/json; charset=UTF-8",
92                  dataType : "json",
93                  type : "DELETE",
94                  processData : false,
95                  timeout : 30000,
96                  beforeSend: function(xhr) {
97                      xhr.setRequestHeader('Authorization', global.authorization);
98                  },
99                  success : function(json) {
100                      console.log(json);
101                      alert("Boleta borrada con éxito");
102                      location.reload();
103                  },
104                  error : function(xhr) {
105                      console.log(xhr);
106                      alert("Error en la consulta, ver consola para más detalles");
107                  },
108                  complete : function(xhr, status) {
109                  }
110              }
111          );
112      }
113  }
114  }
115  }
```

`queries/boletas.js`

Luego, en `views/boletas.js` se modifica el método `confirm-revocation-delete` para borrar la boleta respectiva.

```
55
56      $('#confirm-boleta-delete').click(function(){
57          var boleta = window.sessionStorage.getItem("selected_boleta");
58          var contribuyente = window.sessionStorage.getItem("contribuyente");
59          boletas.delete(contribuyente, boleta);
60      });
```

`views/boletas.js`

Login persistente

Al momento de entrar al login de la aplicación, fue necesario modificar `views/index.js` para identificar al usuario y el número de grupo.

```
1 |var indexView = {
2 |  init: function(){
3 |    $('#start-button').click(function(){
4 |      var username = window.localStorage.getItem("username", $('#username').val() );
5 |      var group = window.localStorage.getItem("group", $('#group').val() );
6 |      if(username && group){
7 |        window.location.href = "contribuyentes.html";
8 |      }
9 |      else{
10 |        window.location.href = "login.html";
11 |      }
12 |    });
13 |  }
14 |}
```

views/index.js

En `views/login.js` se agregó el almacenamiento de variable persistente, necesario para mantener abierta la sesión del usuario.

```
1 |var loginView = {
2 |  init: function(){
3 |    $('#button-login').prop('disabled', true);
4 |
5 |    $('#button-login').click(function(){
6 |      window.localStorage.setItem("username", $('#username').val() );
7 |      window.localStorage.setItem("group", $('#group').val() );
8 |
9 |      window.location.href = 'contribuyentes.html';
10 |    });
11 |
12 |    $('#group-number, #username').on('keyup change click', function(){
13 |      var username = $('#username').val();
14 |      var group = $('#group-number').val();
15 |      if(group != '' && username != ''){
16 |        $('#button-login').prop('disabled', false);
17 |      }else{
18 |        $('#button-login').prop('disabled', true);
19 |      }
20 |    });
21 |  }
22 |}
23 |}
```

views/login.js

En `views/navbar.js` se implementa el borrado de variable persistente.

```
1 |var navbarView = {
2 |  init: function(){
3 |    $('#navbar-username').html(window.localStorage.getItem("username"));
4 |
5 |    $('#logout-button').click(function(){
6 |      window.localStorage.removeItem("username");
7 |      window.localStorage.removeItem("group");
8 |      window.location.href = 'index.html';
9 |    });
10 |
11 |    $('#reload-button').click(function(){
12 |      window.location.reload();
13 |    });
14 |  }
15 |}
16 |
```

views/navbar.js