

SPRINT 2: Desarrollo del Backend

Identificación Proyecto	
Nombre Proyecto:	Marcadores Deportivos
Número Equipo:	1
Integrantes del equipo	
Rol (Líder-Desarrollador – Cliente)	Nombre
Líder (Scrum Master)	Leyniker Ferley Celis
Desarrolladores (Scrum Team)	Yadir Andrés Ortíz Ruendes
Desarrolladores (Scrum Team)	María Camila Esquivel Álvarez
Desarrolladores (Scrum Team)	Rafael Antonio Cardona Sierra
Cliente	Osmer Eduardo Ruiz Mateus

Repositorio GitHub: <https://github.com/lefcelis/Backend-Marcadores-Deportivos.git>

Evidencia construcción del Backend

Como evidencia de la construcción del Backend, se debe presentar capturas de pantalla donde se visualice el proceso de construcción del Backend, como la creación en Spring Boot, modelo, controlador, etc.



● Controllers:

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda eventscontrollers.js - marcadoresdeportivos - Visual Studio Code [Administrador]

EXPLORADOR controllers > JS eventscontrollers.js > [e] default

```

1 import {evento} from '../models/evento.js';
2
3 //guardar un evento deportivo
4
5 export const regevento = (req, res) => {
6   const cevento = evento(req.body);
7   cevento
8     .save()
9     .then((data) => res.json(data))
10    .catch((error) => res.json({ message: error }));
11
12 //mostrar los eventos deportivos que estan en la bd de mongodb
13
14 export const showevento = (req,res) => {
15
16   const evento
17   .find()
18   .then((data)=> res.json(data))
19   .catch((error)=> res.json({message:error}));
20
21 }

```

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN

[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada ✅ con éxito a la bd:
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada ✅ con éxito a la bd:

Lín. 59, col. 26 Espacios: 2 UTF-8 CRLF () JavaScript ⚡ Go Live ⌂

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda usercontroller.js - marcadoresdeportivos - Visual Studio Code [Administrador]

EXPLORADOR controllers > JS usercontroller.js > [e] showuser

```

1 import { usuario } from '../models/usuario.js';
2
3 // Registrar un usuario en la base de datos
4 export const regusuario = (req, res) => {
5   const user = usuario(req.body);
6   user
7     .save()
8     .then((data) => res.json(data))
9     .catch((error) => res.json({ message: error }));
10
11 // Mostrar información de los usuarios
12
13 export const showuser = (req,res) => {
14
15   const usuario
16   .find()
17   .then((data)=> res.json(data))
18   .catch((error)=> res.json({message:error}));
19
20 }
21

```

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN

[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada ✅ con éxito a la bd:
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada ✅ con éxito a la bd:

Lín. 14, col. 41 Espacios: 2 UTF-8 CRLF () JavaScript ⚡ Go Live ⌂



Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda userscontroller.js - marcadoresdeportivos - Visual Studio Code [Administrador]

EXPLORADOR controllers > JS userscontroller.js > showuser

```

controllers > JS userscontroller.js > showuser
34     export const deluser = (req, res) => {
35         const { id } = req.params;
36         usuario
37             .deleteOne({ _id: id })
38             .then((data) => res.json(data))
39             .catch((error) => res.json({ message: error }));
40     };
41
42     //Actualizar un usuario
43     export const upuser = (req, res) => {
44         const { id } = req.params;
45         const { nomuser, password, correo } = req.body;
46         usuario
47             .updateOne({ _id: id }, { $set: { nomuser, password, correo } })
48             .then((data) => res.json(data))
49             .catch((error) => res.json({ message: error }));
50     };
51
52
53     export default regusuario;

```

PROBLEMAS SALIDA TERMINAL JUPYTER CONSOLA DE DEPURACIÓN

[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada ✅ con éxito a la bd:
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada ✅ con éxito a la bd:

Lín. 14, col. 41 Espacios: 2 UTF-8 CRLF ⓘ JavaScript ⓘ Go Live ⓘ

File Edit Selection View Go Run Terminal Help loginusercontrollers.js - Untitled (Workspace) - Visual Studio Code

EXPLORADOR controllers > JS loginusercontrollers.js > ...

Backend-Marcadores-Deportivos > controllers > JS loginusercontrollers.js > ...

```

You, yesterday | 1 author (You)
1 import bcrypt from 'bcrypt';
2 import usuario from '../models/usuario.js';
3
4
5 module.exports = (req,res)=>{
6     const { nomuser,password } = req.body
7
8
9     User.findOne({nomuser: nomuser},function(error,usuario){
10         if(usuario){
11             bcrypt.compare(password, usuario.password, (error,same)=>{
12                 if(same){
13                     any
14                     req.session.userId = user._id
15                     res.redirect('/')
16                 }
17                 else{
18                     res.redirect('/auth/login')
19                 }
20             })
21         }
22     })
23 }

```

PROBLEMAS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER GITLENS

[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor corriendo en http://localhost:8000/
La conexión fue realizada ✅ con éxito a la Base de Datos: ✨

ⓘ You, yesterday Ln 1, Col 1 Spaces: 4 UTF-8 CRLF ⓘ JavaScript ⓘ Go Live ⓘ Prettier ⓘ



- **Models:**

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left lists project files including 'controllers', 'models', 'routes', and various configuration files like '.env' and '.gitignore'. The 'usuario.js' file in the 'models' folder is selected and open in the main editor area. The code defines a MongoDB schema for a user with fields for name, password, email, and timestamps. The terminal at the bottom shows the output of a nodemon restart command, indicating the application is running on localhost:8000.

```
1 import mongoose from "mongoose";
2 const userSchema = new mongoose.Schema(
3     {
4         nombre: {
5             type: String,
6             require: true
7         },
8         password: {
9             type: String,
10            require: true
11        },
12        correo: {
13            type: String,
14            require: true,
15            unique: true
16        }
17    },
18    {
19        timestamps: true,
20    }
);

```



```
models > JS usuario.js > ...
8   password: {
9     type: String,
10    require: true
11  },
12  correo: {
13    type: String,
14    require: true,
15    unique: true
16  },
17  {
18    timestamps: true,
19    versionKey: false,
20  }
21 );
22
23 export const usuario = mongoose.model('users', userSchema);
24
25
26 export default usuario;
```

PROBLEMAS SALIDA TERMINAL JUPYTER CONSOLA DE DEPURACIÓN

```
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
```

LIN. 1, COL. 1 Espacios: 4 UTF-8 CRLF ⓘ JavaScript ⓘ Go Live ⓘ

● Node Modules:

```
models > JS usuario.js > ...
8   password: {
9     type: String,
10    require: true
11  },
12  correo: {
13    type: String,
14    require: true,
15    unique: true
16  },
17  {
18    timestamps: true,
19    versionKey: false,
20  }
21 );
22
23 export const usuario = mongoose.model('users', userSchema);
24
25
26 export default usuario;
```

PROBLEMAS SALIDA TERMINAL JUPYTER CONSOLA DE DEPURACIÓN

```
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
```



● Routes:

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda routes.js - marcadoresdeportivos - Visual Studio Code [Administrador]

```

EXPLORADOR ... JS eventscontrollers.js U JS userscontroller.js M Configuración JS loginusercontroller.js U JS evento.js JS routes.js M ...
routes > JS routes.js > ...
1 import express from 'express';
2 import { regusuario, showuser, shuserone, deluser, upuser } from '../controllers/userscontroller.js';
3 import { regevento, showevento, showonevento, delonevent, upevento } from '../controllers/eventscontrollers.js';
4 import { db } from '../mongodb.js';
5
6 export const router = express.Router();
7
8 //rutas collection eventos
9
10 router.post('/regevento', regevento);
11 router.get('/shevento', showevento);
12 router.get('/shevento/:id', showonevento);
13 router.delete('/delevento/:id', delonevent);
14 router.put('/upevento/:id', upevento);
15
16 //rutas collection usuarios de la bd de mongodb
17 router.post('/reguser', regusuario);
18 router.get('/shuser', showuser);
19 router.get('/shuser/:id', shuserone);
20 router.delete('/deluser/:id', deluser);
21 router.put('/upuser/:id', upuser);

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
Lin. 14, col. 39 Espacios:4 UTF-8 CRLF ⓘ JavaScript ⓘ Go Live ⓘ

```

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda routes.js - marcadoresdeportivos - Visual Studio Code [Administrador]

```

EXPLORADOR ... JS eventscontrollers.js U JS userscontroller.js M Configuración JS loginusercontroller.js U JS evento.js JS routes.js M ...
routes > JS routes.js > ...
5
6 export const router = express.Router();
7
8 //rutas collection eventos
9
10 router.post('/regevento', regevento);
11 router.get('/shevento', showevento);
12 router.get('/shevento/:id', showonevento);
13 router.delete('/delevento/:id', delonevent);
14 router.put('/upevento/:id', upevento);
15
16 //rutas collection usuarios de la bd de mongodb
17 router.post('/reguser', regusuario);
18 router.get('/shuser', showuser);
19 router.get('/shuser/:id', shuserone);
20 router.delete('/deluser/:id', deluser);
21 router.put('/upuser/:id', upuser);
22
23 export default router;

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
Lin. 14, col. 39 Espacios:4 UTF-8 CRLF ⓘ JavaScript ⓘ Go Live ⓘ

```



● .env:

A screenshot of Visual Studio Code showing the contents of the `.env` file. The file contains a single line of environment variable configuration:

```
MONGO_DBURI = mongodb+srv://mariacamillaea:ciclo4desarrollo@cluster0.3oopdu7.mongodb.net/dbeventos?retryWrites=true&w=majority
```

The code editor interface includes a sidebar with project files like `controllers`, `models`, `routes`, and `models`; a terminal tab showing Node.js logs; and a bottom status bar indicating the file has 1 line, 4 spaces, and is in UTF-8 encoding.

● .gitignore:

A screenshot of Visual Studio Code showing the contents of the `.gitignore` file. The file contains a single line that ignores the `.env` file:

```
.env
```

The code editor interface is similar to the previous screenshot, showing the same project structure, terminal logs, and status bar.



● index.js:

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda index.js - marcadoresdeportivos - Visual Studio Code [Administrador]

EXPLORADOR JS index.js M .env .gitignore package-lock.json package.json tsconfig.json Dockerfile index.js M mongoDB.js M package-lock.json package.json vercel.json

```

1  console.log("Bienvenidos a Node Js");
2
3  import express from "express";
4  import cors from 'cors';
5  import { router } from './routes/routes.js'
6
7  //import blogRoutes from './routes/routes.js'
8
9  const app = express()
10 app.use(cors())
11 app.use(express.json())
12 //app.use('/usuarios',blogRoutes)
13 app.use('/',router);
14
15
16
17 app.listen(8000, ()=> {
18   console.log('Servidor Corriendo en http://localhost:8000/');
19 })

```

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN

```

[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
  La conexión fue realizada 🚀 con éxito a la bd:🚀
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
  La conexión fue realizada 🚀 con éxito a la bd:🚀

```

Lín. 11, col. 24 Espacios:4 UTF-8 CRLF {} JavaScript Go Live

● MongoDB.js:

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda mongodb.js - marcadoresdeportivos - Visual Studio Code [Administrador]

EXPLORADOR JS mongodb.js M New Request index.js M .env .gitignore package-lock.json package.json Dockerfile index.js M mongoDB.js M package-lock.json package.json vercel.json

```

1  import mongoose from "mongoose";
2  import dotenv from "dotenv";
3  dotenv.config();
4
5
6  const bd_url = 'mongodb+srv://mariacamilaea:<ciclo4desarrollo*>@cluster0.36opdu7.mongodb.net/?retryWrites=true';
7  export const db = mongoose.connect(process.env.MONGO_DBURI).then(() => {
8    console.log(`La conexión fue realizada 🚀 con éxito a la bd:🚀`);
9  }).catch((error) => console.error("Problemas al conectar a la Base de datos de MongoDB", error));
10
11
12
13 export default db;
14

```

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN

```

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6
PS C:\Users\Luz Stella Alvarez\Documents\ciclo4\Sprint_2\Backend\marcadoresdeportivos> npm run start
> dockeng51@1.0.0 start
> nodemon index.js
[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
```

Lín. 11, col. 1 Espacios:4 UTF-8 CRLF {} JavaScript Go Live



Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda mongoDB.js - marcadoresdeportivos - Visual Studio Code [Administrador]

EXPLORADOR

```

    < MARCADORESDEPORTIVOS
        < controllers
            JS eventscontrollers.js
            JS loginusercontrollers.js
            JS userscontroller.js
        < models
            JS evento.js
            JS usuario.js
        < node_modules
            JS routes.js
            .dockernignore
        < .env
        .gitignore
        Dockerfile
        JS index.js
        JS mongoDB.js
        {} package-lock.json
        {} package.json
        {} vercel.json
    > ESQUEMA
    > LÍNEA DE TIEMPO
    > DOCKER CONTAINERS
    > DOCKER IMAGES
    > AZURE CONTAINER REGISTRY
    > DOCKER HUB
    > SUGGESTED DOCKER HUB IMAGES
    > MYSQL

```

mongodB.js (M) index.js (M) .env .gitignore package-lock.json (M) package.json (M)

```

1 import mongoose from "mongoose";
2 import dotenv from "dotenv";
3 dotenv.config();
4
5 const bd_url = 'mongodb+srv://mariacamillaea:< ciclo4desarrollo*>@cluster0.36opdu7.mongodb.net/?retryWrites=true&w=majority';
6 export const db = mongoose.connect(process.env.MONGO_DBURI).then(()=> {
7     console.log(` La conexión fue realizada con éxito a la bd: ${bd_url}`);
8 }).catch((error) => console.error("Problemas al conectar a la Base de datos de MongoDB", error));
9
10
11
12
13 export default db;
14

```

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN

```

[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
 La conexión fue realizada con éxito a la bd: mongodb+srv://mariacamillaea:< ciclo4desarrollo*>@cluster0.36opdu7.mongodb.net/?retryWrites=true&w=majority
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
 La conexión fue realizada con éxito a la bd: mongodb+srv://mariacamillaea:< ciclo4desarrollo*>@cluster0.36opdu7.mongodb.net/?retryWrites=true&w=majority

```

Lín. 11, col. 1 Espacios: 4 UTF-8 CRLF {} JavaScript ⌂ Go Live ⌂ ⌂

● package-lock.json / package.json

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda package-lock.json - marcadoresdeportivos - Visual Studio Code [Administrador]

EXPLORADOR

```

    < MARCADORESDEPORTIVOS
        < controllers
            JS eventscontrollers.js
            JS loginusercontrollers.js
            JS userscontroller.js
        < models
            JS evento.js
            JS usuario.js
        < node_modules
            JS routes.js
            .dockernignore
        < .env
        .gitignore
        Dockerfile
        JS index.js
        JS mongoDB.js
        {} package-lock.json (M)
        {} package.json
        {} vercel.json
    > ESQUEMA
    > LÍNEA DE TIEMPO
    > DOCKER CONTAINERS
    > DOCKER IMAGES
    > AZURE CONTAINER REGISTRY
    > DOCKER HUB
    > SUGGESTED DOCKER HUB IMAGES
    > MYSQL

```

package-lock.json (M) index.js (M) .env .gitignore package.json (M) package-lock.json (M) package.json (M)

```

1 package-lock.json > {} dependencies > {} unpipe
2   "name": "dockeng51",
3   "version": "1.0.0",
4   "lockfileVersion": 2,
5   "requires": true,
6   "packages": {
7     "": {
8       "name": "dockeng51",
9       "version": "1.0.0",
10      "license": "ISC",
11      "dependencies": {
12        "dotenv": "^16.0.3"
13      },
14      "devDependencies": {
15        "cors": "^2.8.5",
16        "express": "4.18.2",
17        "mongoose": "6.7.1",
18        "nodemon": "2.0.20"
19      }
20    },
21    "node_modules/@aws-crypto/ie11-detection": {}

```

PROBLEMAS SALIDA TERMINAL JUPITER CONSOLA DE DEPURACIÓN

```

[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
 La conexión fue realizada con éxito a la bd: mongodb+srv://mariacamillaea:< ciclo4desarrollo*>@cluster0.36opdu7.mongodb.net/?retryWrites=true&w=majority
[nodemon] restarting due to changes...
[nodemon] starting `node index.js`
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
 La conexión fue realizada con éxito a la bd: mongodb+srv://mariacamillaea:< ciclo4desarrollo*>@cluster0.36opdu7.mongodb.net/?retryWrites=true&w=majority

```

Lín. 440, col. 1 Espacios: 2 UTF-8 LF {} JSON ⌂ Go Live ⌂ ⌂



The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure with files like `controllers`, `models`, `routes`, and `node_modules`. The `package.json` file is currently selected.
- Code Editor (Center):** Displays the `package.json` file content:

```
9     "test": "echo \"Error: no test specified\" & exit 1"
10 },
11 "keywords": [],
12 "author": "",
13 "license": "ISC",
14 "devDependencies": {
15     "cors": "^2.8.5",
16     "express": "^4.18.2",
17     "mongoose": "^6.7.1",
18     "nodemon": "^2.0.20"
19 },
20 "dependencies": {
21     "dotenv": "^16.0.3"
22 }
23 }
24 }
```
- Terminal (Bottom):** Shows the output of a command, likely `npm start`, indicating the application is running on port 8000.

- **vercel.json:**

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer (Left):** Shows the project structure with files like `controllers`, `models`, `routes`, and `index.js`.
- Code Editor (Top Center):** Displays the `vercel.json` file content.
- Terminal (Bottom):** Shows the output of a node server restart and connection logs.

```
vercel.json > ...
1  {
2    "version": 2,
3    "builds": [
4      {
5        "src": "./index.js",
6        "use": "vercel/node"
7      }
8    ],
9    "routes": [
10      {
11        "src": "/(.*)",
12        "dest": "/"
13      }
14    ]
15 }
```

```
[nodemon] restarting due to changes...
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
[nodemon] restarting due to changes...
[nodemon] starting 'node index.js'
Bienvenidos a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: 
```



● Thunder Client:

The screenshot shows the Thunder Client interface. On the left, there's a sidebar with various icons and a list of recent activities. The main area has a "New Request" button at the top. Below it, a "Query Parameters" section is visible. The central part of the interface shows a "POST" request to "http://localhost:8000/reguser/". The response pane indicates a "Status: 200 OK", "Size: 112 Bytes", and "Time: 562 ms". The response body is a JSON object:

```
1 {  
2   "_id": "637305fdb0ac8d54723b20f3",  
3   "createdAt": "2022-11-15T03:22:37.598Z",  
4   "updatedAt": "2022-11-15T03:22:37.598Z"  
5 }
```

At the bottom, there's a terminal window showing Node.js logs:

```
[nodemon] restarting due to changes...  
[nodemon] restarting due to changes...  
[nodemon] starting `node index.js`  
Bienvenidos a Node Js  
Servidor Corriendo en http://localhost:8000/  
La conexión fue realizada con éxito a la bd:  
[nodemon] restarting due to changes...  
[nodemon] starting `node index.js`  
Bienvenidos a Node Js  
Servidor Corriendo en http://localhost:8000/  
La conexión fue realizada con éxito a la bd:  
[]
```

Evidencias de los “endpoint” con el consumo de recursos del API REST

Como evidencia de los “endpoint” donde se visualice el consumo de recursos del API REST.

Link de conexión a la base de datos “dbeventos” en MongoDB:

mongodb+srv://mariacamilaea:<password>@cluster0.36opdu7.mongodb.net/dbeventos?retryWrites=true&w=majority



cloud.mongodb.com/v2/6366ca89b0ad221f63bbba06#clusters

Maria Camila... Access Manager Billing All Clusters Get Help Maria Camila

Project 0 Atlas App Services Charts

MARIA CAMILA'S ORG - 2022-11-05 > PROJECT 0

Database Deployments

Find a database deployment...

+ Create

Cluster0 Connect View Monitoring Browse Collections ...

FREE **SHARED**

Enhance Your Experience

For production throughput and richer metrics, upgrade to a dedicated cluster now!

Upgrade

0.003 **W 0.003** Connections 8.0 **In 93.0 B/s** **Out 714.3 B/s** Data Size 24.5 KB

Last 6 hours 0.01/s Last 6 hours 17.0 1.4 KB/s Last 10 days 512.0 MB

VERSION REGION CLUSTERTIER TYPE BACKUPS LINKED APP SERVICES ATLAS SEARCH

5.0.13 AWS N. Virginia (us-east-1) MD Sandbox (General) Replica Set - 3 nodes Inactive None Linked Create Index

- Collection “users” de la base de datos “dbeventos”:

cloud.mongodb.com/v2/6366ca89b0ad221f63bbba06#metrics/replicaSet/6366cc26fb74915aab9e671/explorer/dbeventos/users/find

Maria Camila... Access Manager Billing All Clusters Get Help Maria Camila

Project 0 Atlas App Services Charts

MARIA CAMILA'S ORG - 2022-11-05 > PROJECT 0 > DATABASES

Cluster0

VERSION: 5.0.13 REGION: AWS N. Virginia (us-east-1)

Overview Real Time Metrics Collections Search Profiler Performance Advisor Online Archive Cmd Line Tools

DATABASES: 2 COLLECTIONS: 3

+ Create Database

Search Namespaces

dbeventos

- eventos
- users**
- mongodbVSCodePlaygrou...

dbeventos.users

STORAGE SIZE: 20KB LOGICAL DATA SIZE: 60B TOTAL DOCUMENTS: 1 INDEXES TOTAL SIZE: 40KB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

FILTER { field: 'value' } Options Apply Reset

QUERY RESULTS: 1-1 OF 1

```
_id: ObjectId('637305fdbbac8d54723b20f3')
createdAt: 2022-11-15T03:22:37.598+00:00
updatedAt: 2022-11-15T03:22:37.598+00:00
```



- Collection “ceventos” de la base de datos “dbeventos”:

The screenshot shows the MongoDB Atlas interface. On the left, there's a sidebar with sections like DEPLOYMENT, DATA SERVICES, and SECURITY. The main area shows a database named "dbeventos" with a collection named "ceventos". Under "ceventos", there are two documents listed:

```

_id: ObjectId('6373e900b0ac8d54723b20f7')
createdAt: 2022-11-15T03:35:28.740+00:00
updatedAt: 2022-11-15T03:35:28.740+00:00

_id: ObjectId('6373e93cb0ac8d54723b20f9')
createdAt: 2022-11-15T03:36:28.920+00:00
updatedAt: 2022-11-15T03:36:28.920+00:00

```

Validación de Usuarios

- Registrar un usuario en la base de datos: <http://localhost:8000/reguser/>

The screenshot shows a POST request to `http://localhost:8000/reguser/`. The JSON body contains the following data:

```

{
  "nomuser": "Juan",
  "password": "123salodl",
  "correo": "juandv@gmail.com",
  "_id": "637311e403a9b62cc623f220",
  "createdAt": "2022-11-15T04:13:24.256Z",
  "updatedAt": "2022-11-15T04:13:24.256Z"
}

```



Insomnia - New Document – New Request

Application Edit View Window Tools Help

POST http://localhost:8000/register

Send 200 OK 111 ms 103 B Just Now

New Environment Cookies

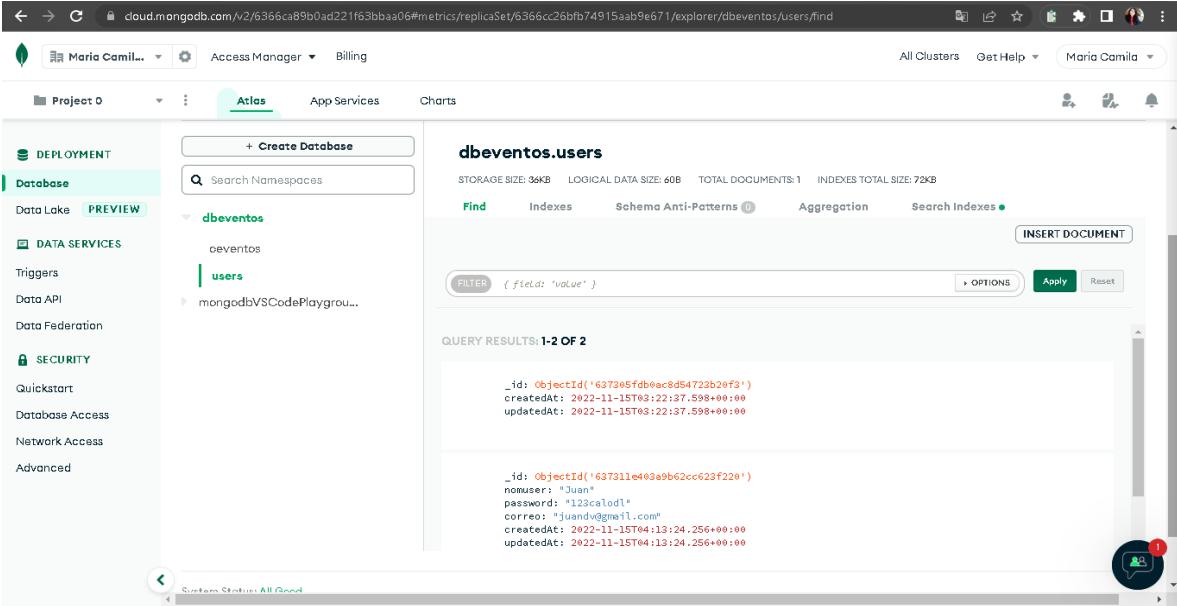
POST New Request

```
1: {
2:   "nomuser": "Juan",
3:   "password": "123salod1",
4:   "correo": "juandv@gmail.com"
5: }
```

Preview Headers Cookies Timeline

```
1: {
2:   "message": {
3:     "index": 0,
4:     "code": 1000,
5:     "KeyPattern": {
6:       "correo": 1
7:     },
8:     "KeyValue": {
9:       "correo": "juandv@gmail.com"
10:    }
11:  }
12: }
```

Beautify JSON \$.store .books [*].author



cloud.mongodb.com/v2/6366ca89b0ad22f1f63bbbaa06#metrics/replicaSet/6366cc26fb74915aab9e671/explorer/dbeventos/users/find

Maria Camila... Access Manager Billing All Clusters Get Help Maria Camila

Project O Atlas App Services Charts

+ Create Database Search Namespaces

DEPLOYMENT Database DATA SERVICES SECURITY

Data Lake PREVIEW Triggers Data API Data Federation

mongodbVSCodePlaygrou...

dbeventos.users

STORAGE SIZE: 86KB LOGICAL DATA SIZE: 60B TOTAL DOCUMENTS: 1 INDEXES TOTAL SIZE: 72KB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

INSERT DOCUMENT

FILTER { field: 'value' } OPTIONS Apply Reset

QUERY RESULTS: 1-2 OF 2

```
_id: ObjectId('637305fdb8ec8d54723b20f3')
createdAt: 2022-11-15T08:22:37.598+00:00
updatedAt: 2022-11-15T08:22:37.598+00:00
```

```
_id: ObjectId('637311e403e9b62cc623f22e')
nomuser: "Juan"
password: "123salod1"
correo: "juandv@gmail.com"
createdAt: 2022-11-15T04:13:24.256+00:00
updatedAt: 2022-11-15T04:13:24.256+00:00
```



Screenshot of a developer environment showing a POST request to 'localhost:8000/reguser' with JSON body:

```
1 {
2   "nomuser": "Gabriela",
3   "password": "gabbcshfhdl",
4   "correo": "gabypdhgg@gmail.com",
5 }
```

The response status is 200 OK, size 191 Bytes, and time 128 ms.

Terminal output:

```
La conexión fue realizada con éxito a la bd: [node:mon] restarting due to changes...
[nodemon] starting 'node index.js'
Buenos días a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: [node:mon] restarting due to changes...
[nodemon] starting 'node index.js'
Buenos días a Node Js
Servidor Corriendo en http://localhost:8000/
La conexión fue realizada con éxito a la bd: [node:mon] restarting due to changes...
[nodemon] starting 'node index.js'
Buenos días a Node Js
Servidor Corriendo en http://localhost:8000/
```

Screenshot of the MongoDB Atlas interface showing the 'users' collection in the 'dbeventos' database.

Document details:

```
_id: ObjectId('637312b403a9b62cc623f226')
nomuser: "Juan"
password: "123calodl"
correo: "juandv@gmail.com"
createdAt: 2022-11-15T04:13:24.256+00:00
updatedAt: 2022-11-15T04:13:24.256+00:00
```

Another document is partially visible:

```
_id: ObjectId('637312b403a9b62cc623f226')
nomuser: "Gabriela"
password: "gabbcshfhdl"
correo: "gabypdhgg@gmail.com"
createdAt: 2022-11-15T04:16:52.138+00:00
updatedAt: 2022-11-15T04:16:52.138+00:00
```



Screenshot of Visual Studio Code showing a POST request to `localhost:8000/reguser`. The response status is 200 OK, size 194 Bytes, and time 110 ms. The JSON response body is:

```
1  {
2    "nomuser": "Cris Mateous",
3    "password": "mateoushi",
4    "correo": "crismateous4@gmail.com"
5 }
```

Screenshot of Insomnia showing a POST request to `http://localhost:8000/reguser`. The response status is 200 Ok, time 106 ms, and size 109 B. The JSON response body is:

```
1 - {
2   "message": {
3     "index": 0,
4     "code": 1000,
5     "keyPattern": {
6       "correct": 1
7     },
8     "keyValue": {
9       "correo": "crismateous4@gmail.com"
10    }
11  }
12 }
```

The screenshot shows the MongoDB Atlas interface. On the left, there's a sidebar with 'Project O' selected, showing 'DEPLOYMENT', 'Database' (with 'dbeventos' selected), 'DATA SERVICES', 'SECURITY', and 'System Status: All Good'. The main area is titled 'dbeventos.users' and shows 'STORAGE SIZE: 34KB', 'LOGICAL DATA SIZE: 478B', 'TOTAL DOCUMENTS: 4', and 'INDEXES TOTAL SIZE: 72KB'. It has tabs for 'Find', 'Indexes', 'Schema Anti-Patterns', 'Aggregation', and 'Search Indexes'. Below these are 'INSERT DOCUMENT' and 'OPTIONS' buttons. A 'FILTER' bar is at the top of the document list. Two documents are visible:

```

{
  "correo": "gadypanggve@gmail.com",
  "createdAt": "2022-11-15T04:16:52.138+00:00",
  "updatedAt": "2022-11-15T04:16:52.138+00:00"
}

{
  "_id": "ObjectID('63731348e3a9b62cc623f22a')",
  "nomuser": "Cris Mateus",
  "password": "mateoush1",
  "correo": "crismateus4@gmail.com",
  "createdAt": "2022-11-15T04:19:28.827+00:00",
  "updatedAt": "2022-11-15T04:19:28.827+00:00"
}

```

- Mostrar usuarios registrados en la base de datos: <http://localhost:8000/showuser/>

The screenshot shows a Visual Studio Code workspace with the 'THUNDER CLIENT' extension open. It lists several API endpoints under 'Activity' and shows a successful 'GET' request to 'localhost:8000/showuser/' with a status of 200 OK, size of 1.35 KB, and time of 478 ms. The response body is a JSON array of user documents:

```

[{"_id": "6379156a9c01b8c578d596c1", "nomuser": "Admin", "password": "Lhdud928", "correo": "admin@gmail.com", "createdAt": "2022-11-19T17:42:03.033Z", "updatedAt": "2022-11-19T17:42:03.033Z"}, {"_id": "63791ae7d9a64a09e7174e46", "nomuser": "Maia", "password": "0kdnksd929", "correo": "maria@gmail.com", "createdAt": "2022-11-19T18:05:33.855Z", "updatedAt": "2022-11-19T18:05:33.855Z"}, {"_id": "63791be87a9a64a09e7174e48", "nomuser": "Carlos", "password": "123456789", "correo": "carlos@gmail.com", "createdAt": "2022-11-19T18:06:00.687Z", "updatedAt": "2022-11-19T18:21:41.051Z"}, {"_id": "63791bf7ca99404de4e46", "nomuser": "Luis", "password": "Hjdjd13e", "correo": "luis@gmail.com", "createdAt": "2022-11-19T18:10:04.844Z", "updatedAt": "2022-11-19T18:10:04.844Z"}, {"_id": "63791c0787292310f655", "nomuser": "Marta", "password": "Khdkdjs982990", "correo": "marta@gmail.com", "createdAt": "2022-11-20T00:38:14.385Z", "updatedAt": "2022-11-20T00:38:14.385Z"}, {"_id": "6379af79f594086517sF962", "nomuser": "Andrés Pérez", "password": "92b$123e19f15wd.PEMRjgNXWm/ex9tLE1HLhL1.07PHAVhN4Gu9eaV8K", "correo": "andres@gmail.com", "createdAt": "2022-11-21T02:16:25.889Z", "updatedAt": "2022-11-21T02:16:25.889Z"}, {"_id": "637afdc6b5b95f767da7dcdf", "nomuser": "Jerson", "password": "$2b$12$IpEh6nsNaJ05mSlD3fgV3.zXZY67n1jRs1l.OMhp0RPKi0tCTCAGy", "correo": "jerson@gmail.com", "createdAt": "2022-11-21T04:25:42.754Z", "updatedAt": "2022-11-21T04:25:42.754Z"}]

```

The screenshot shows a browser window with the URL 'localhost:8000/showuser/'. The page displays the same JSON array of user documents as the VS Code screenshot.



- Mostrar un usuario específico por ID: <http://localhost:8000/showuser/:id>

Screenshot of Visual Studio Code showing the Thunder Client extension. A GET request is made to `http://localhost:8000/showuser/637adf79faf94986517af962`. The response status is 200 OK, size is 1.35 KB, and time is 478 ms. The response body contains two user documents:

```

34  [
35    "_id": "637976f6fb087392310fb365",
36    "nomuser": "Marta",
37    "password": "Khdkdj5982990",
38    "correo": "marta@gmail.com",
39    "createdAt": "2022-11-20T00:38:14.385Z",
40    "updatedAt": "2022-11-20T00:38:14.385Z"
41  },
42  {
43    "_id": "637adf79faf94986517af962",
44    "nomuser": "Andrés Pérez",
45    "password": "$2b$12$0t9ij15wd.PEmRjg0XWm/eX9tLE1LhL1.U7PHAVh0AGu9eaV/8K",
46    "correo": "andres@gmail.com",
47    "createdAt": "2022-11-21T02:16:25.889Z",
48    "updatedAt": "2022-11-21T02:16:25.889Z"
49  },
50  {
51    "_id": "637afdc6b5b95f767da7cdcf",
52    "nomuser": "Jerson",
53    "password": "Khdkdj5982990",
54    "correo": "jerson@gmail.com",
55    "createdAt": "2022-11-21T02:16:25.889Z",
56    "updatedAt": "2022-11-21T02:16:25.889Z"
57  }

```

Screenshot of a web browser showing the response from the previous request. The JSON object returned is:

```
{
  "_id": "637adf79faf94986517af962",
  "nomuser": "Andrés Pérez",
  "password": "$2b$12$0t9ij15wd.PEmRjg0XWm/eX9tLE1LhL1.U7PHAVh0AGu9eaV/8K",
  "correo": "andres@gmail.com",
  "createdAt": "2022-11-21T02:16:25.889Z",
  "updatedAt": "2022-11-21T02:16:25.889Z"
}
```

- Eliminar un usuario registrado en la bd: <http://localhost:8000/deluser/:id>

Screenshot of Visual Studio Code showing the Thunder Client extension. A DELETE request is made to `http://localhost:8000/deluser/63791bf7a964a09e7174e4d`. The response status is 200 OK, size is 1.35 KB, and time is 478 ms. The response body contains two user documents, identical to the ones shown in the previous screenshot.



The screenshot shows a Visual Studio Code interface with a 'THUNDER CLIENT' extension open. The 'Activity' tab is selected, displaying a list of API requests. A successful 'DELETE' request to 'localhost:8000/deluser/63791bcf7a964a09e7174e4d' is highlighted. The 'Headers' tab shows the response status: 'Status: 200 OK', 'Size: 38 Bytes', and 'Time: 188 ms'. The response body contains JSON data: { "acknowledged": true, "deletedCount": 1 }.

```
Status: 200 OK Size: 38 Bytes Time: 188 ms
Response Headers Cookies Results Docs
1 {
2   "acknowledged": true,
3   "deletedCount": 1
4 }
```

- Actualizar un usuario registrado en la bd: <http://localhost:8000/upuser/:id>

The screenshot shows the Thunder Client extension in Visual Studio Code. A new request is being made to the endpoint `/user`. The response status is 200 OK, and the response body contains two user objects:

```
[{"_id": "63791aed7a964a09e7174e46", "nomuser": "Admin", "password": "1hdud928", "correo": "admin@gmail.com", "createdAt": "2022-11-19T17:42:03.033Z", "updatedAt": "2022-11-19T17:42:03.033Z"}, {"_id": "63791b087a964a09e7174e48", "nomuser": "María", "password": "Odkndskd929", "correo": "maria@gmail.com", "createdAt": "2022-11-19T18:05:33.855Z", "updatedAt": "2022-11-22T05:19:14.965Z"}, {"_id": "63791b087a964a09e7174e48", "nomuser": "Carlos", "password": "123456789", "correo": "carlos@gmail.com", "createdAt": "2022-11-19T18:06:00.687Z", "updatedAt": "2022-11-19T18:21:41.051Z"}]
```



The screenshot shows a Visual Studio Code interface with several panes:

- Activity** pane on the left lists recent API requests:
 - PUT localhost:8000/upuser/63791ae7a964a09e7174e46 just now
 - GET localhost:8000/showuser 2 mins ago
 - DEL localhost:8000/deluser/63791bf... 1 sec ago
 - GET localhost:8000/showuser/ 1 sec ago
 - POST localhost:8000/reguser 1 day ago
 - GET localhost:8000/showuser/63791... 1 day ago
 - DEL localhost:8000/delete 1 day ago
 - POST localhost:8000/agregar 1 day ago
 - GET localhost:8000/ 1 day ago
- Terminal** pane at the bottom shows the Node.js application running and connecting to MongoDB.
- Code Editor** pane shows a PUT request to update a user document in MongoDB. The JSON payload is:

```
1 {
2   "nomuser": "María Ángela Suárez Gómez",
3   "password": "HUPWIDHS87shd76JH",
4   "correo": "maria.suarez@hotmail.com"
5 }
```
- Status Bar** at the bottom indicates the connection was successful: "La conexión fue realizada con Éxito a la Base de Datos: 🎉"

The screenshot shows a Visual Studio Code interface with a 'New Request - Untitled (Workspace)' tab open. On the left, the 'THUNDER CLIENT' sidebar lists several recent requests. The main area displays a POST request to 'localhost:8000/showuser'. The 'Body' tab shows the JSON content sent:

```
1 {
2   "nomuser": "María Ángela Suárez Gómez",
3   "password": "HUPWIDHS87shd763n",
4   "correo": "maria.suarez@hotmail.com"
5 }
```

The response on the right shows a successful 200 OK status with a size of 1.22 KB and a time of 115 ms. The response body contains two user objects:

```
5   password: "Loudq9z8",
6   correo: "admin@gmail.com",
7   createdAt: "2022-11-19T17:42:03.033Z",
8   updatedAt: "2022-11-19T17:42:03.033Z"
9 },
10 {
11   _id: "63791a7e964a09e7174e46",
12   nomuser: "María Ángela Suárez Gómez",
13   password: "HUPWIDHS87shd763n",
14   correo: "maria.suarez@hotmail.com",
15   createdAt: "2022-11-19T18:05:33.855Z",
16   updatedAt: "2022-11-22T05:23:44.163Z"
17 },
18 {
19   _id: "63791b87a964a09e7174e48",
20   nomuser: "Carlos",
21   password: "123456789",
22   correo: "carlos@gmail.com",
23   createdAt: "2022-11-19T18:06:00.687Z",
24   updatedAt: "2022-11-19T18:21:41.051Z"
```



Validación de Eventos

- Registrar un evento en la base de datos: <http://localhost:8000/regevento/>

Screenshot of Visual Studio Code showing the Thunder Client extension. A POST request is being sent to <http://localhost:8000/regevento>. The JSON body contains:

```

1 {
2   "fecha": "2022-11-25",
3   "equipo1": "Inglaterra",
4   "equipo2": "Estados Unidos",
5   "marcador1": 4,
6   "marcador2": 2,
7   "tipoevento": "Fútbol"
8 }

```

The terminal shows the Node.js server response:

```

[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`
Bienvenidos a Node JS
Servidor corriendo en http://localhost:8000/
La conexión fue realizada con Éxito a la Base de Datos: 

```

Screenshot of Visual Studio Code showing the Thunder Client extension. A POST request has been successfully sent to <http://localhost:8000/regevento>. The response status is 200 OK, size is 248 Bytes, and time is 758 ms. The JSON response body is:

```

1 {
2   "fecha": "2022-11-25T00:00:00Z",
3   "equipo1": "Inglaterra",
4   "equipo2": "Estados Unidos",
5   "marcador1": 4,
6   "marcador2": 2,
7   "tipoevento": "Fútbol",
8   "_id": "637c60a431b828af353855dd",
9   "createdAt": "2022-11-22T05:39:48.869Z",
10  "updatedAt": "2022-11-22T05:39:48.869Z"
11 }

```

The terminal shows the Node.js server response:

```

[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`
Bienvenidos a Node JS
Servidor corriendo en http://localhost:8000/
La conexión fue realizada con Éxito a la Base de Datos: 

```



- Mostrar los eventos registrados en la bd: <http://localhost:8000/shevento/>

Screenshot of Visual Studio Code showing a Thunder Client extension. The interface includes a sidebar with activity logs, a main panel for sending requests, and a terminal below.

Request: GET http://localhost:8000/shevento

Response:

```

[{"_id": "63793e5905a57256c1ee167b", "fecha": "2022-03-04T00:00:00.000Z", "equipo1": "FC Barcelona", "equipo2": "Real Madrid", "marcador1": 3, "marcador2": 2, "tipoevento": "Fútbol", "v": 0}, {"_id": "63793f2405a57256c1ee167e", "fecha": "2022-07-12T00:00:00.000Z", "equipo1": "Manchester City", "equipo2": "Manchester United", "marcador1": 5, "marcador2": 0, "tipoevento": "Fútbol", "v": 0}, {"_id": "63794074755d081cfcc8609", "fecha": "2022-09-13T00:00:00.000Z", "equipo1": "Arsenal", "equipo2": "PSG", "marcador1": 2, "marcador2": 1, "tipoevento": "Fútbol", "createdAt": "2022-11-22T05:48:15.030Z", "updatedAt": "2022-11-22T05:48:15.030Z"}, {"_id": "637c629f31b828af353855e6", "fecha": "2022-11-23T00:00:00.000Z", "equipo1": "Alemania", "equipo2": "Japón", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol", "createdAt": "2022-11-22T05:48:15.030Z", "updatedAt": "2022-11-22T05:48:15.030Z"}, {"_id": "637c62cc31b828af353855e8", "fecha": "2022-11-26T00:00:00.000Z", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol", "createdAt": "2022-11-22T05:49:00.168Z", "updatedAt": "2022-11-22T05:49:00.168Z"}, {"_id": "637c623b1b828af353855e0", "fecha": "2022-11-19T20:45:40.508Z", "equipo1": "Inglaterra", "equipo2": "Estados Unidos", "marcador1": 4, "marcador2": 2, "tipoevento": "Fútbol", "createdAt": "2022-11-19T20:45:40.508Z", "updatedAt": "2022-11-19T20:45:40.508Z"}, {"_id": "637c624b31b828af353855e2", "fecha": "2022-11-22T05:48:16.869Z", "equipo1": "Argentina", "equipo2": "Arabia Saudita", "marcador1": 1, "marcador2": 0, "tipoevento": "Fútbol", "createdAt": "2022-11-22T05:48:16.869Z", "updatedAt": "2022-11-22T05:48:16.869Z"}, {"_id": "637c6223b1b828af353855e1", "fecha": "2022-11-22T05:48:16.055Z", "equipo1": "Serbia", "equipo2": "Sri Lanka", "marcador1": 1, "marcador2": 0, "tipoevento": "Fútbol", "createdAt": "2022-11-22T05:48:16.055Z", "updatedAt": "2022-11-22T05:48:16.055Z"}, {"_id": "637c6281b1b828af353855e4", "fecha": "2022-11-23T00:00:00.000Z", "equipo1": "Brasil", "equipo2": "Serbia", "marcador1": 4, "marcador2": 0, "tipoevento": "Fútbol", "createdAt": "2022-11-23T00:00:00.000Z", "updatedAt": "2022-11-23T00:00:00.000Z"}, {"_id": "637c6281b1b828af353855e5", "fecha": "2022-11-23T00:00:00.478Z", "equipo1": "Bélgica", "equipo2": "Canadá", "marcador1": 3, "marcador2": 2, "tipoevento": "Fútbol", "createdAt": "2022-11-23T00:00:00.478Z", "updatedAt": "2022-11-23T00:00:00.478Z"}, {"_id": "637c629f31b828af353855e9", "fecha": "2022-11-26T00:00:00.000Z", "equipo1": "Colombia", "equipo2": "Méjico", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol", "createdAt": "2022-11-26T00:00:00.000Z", "updatedAt": "2022-11-26T00:00:00.000Z"}, {"_id": "637c62cc31b828af353855e0", "fecha": "2022-11-26T00:00:00.000Z", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol", "createdAt": "2022-11-26T00:00:00.000Z", "updatedAt": "2022-11-26T00:00:00.000Z"}]

```

Screenshot of a browser window showing the JSON response from the API endpoint.

```

[{"_id": "63793e5905a57256c1ee167b", "fecha": "2022/12/20", "equipo1": "Brasil", "equipo2": "Argentina", "marcador1": 1, "marcador2": 3, "tipoevento": "Fútbol"}, {"_id": "63793f2405a57256c1ee167e", "fecha": "2022/11/15/2022", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "63794074755d081cfcc8609", "fecha": "2022/11/15/2022", "equipo1": "Colombia", "equipo2": "Méjico", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol"}, {"_id": "637c629f31b828af353855e6", "fecha": "2022/11/15/2022", "equipo1": "Alemania", "equipo2": "Japón", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol"}, {"_id": "637c62cc31b828af353855e8", "fecha": "2022/11/15/2022", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "637c623b1b828af353855e0", "fecha": "2022/11/15/2022", "equipo1": "Bélgica", "equipo2": "Canadá", "marcador1": 3, "marcador2": 2, "tipoevento": "Fútbol"}, {"_id": "637c624b31b828af353855e2", "fecha": "2022/11/15/2022", "equipo1": "Serbia", "equipo2": "Sri Lanka", "marcador1": 1, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "637c6281b1b828af353855e5", "fecha": "2022/11/15/2022", "equipo1": "Brasil", "equipo2": "Serbia", "marcador1": 4, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "637c6281b1b828af353855e9", "fecha": "2022/11/15/2022", "equipo1": "Colombia", "equipo2": "Méjico", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol"}, {"_id": "637c62cc31b828af353855e0", "fecha": "2022/11/15/2022", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol"}]

```

Screenshot of the Insomnia REST client showing a successful GET request to the API endpoint.

Request: GET http://localhost:8000/shevento/

Response:

```

[{"_id": "63793e5905a57256c1ee167b", "fecha": "2022/12/20", "equipo1": "Brasil", "equipo2": "Argentina", "marcador1": 1, "marcador2": 3, "tipoevento": "Fútbol"}, {"_id": "63793f2405a57256c1ee167e", "fecha": "2022/11/15/2022", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "63794074755d081cfcc8609", "fecha": "2022/11/15/2022", "equipo1": "Colombia", "equipo2": "Méjico", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol"}, {"_id": "637c629f31b828af353855e6", "fecha": "2022/11/15/2022", "equipo1": "Alemania", "equipo2": "Japón", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol"}, {"_id": "637c62cc31b828af353855e8", "fecha": "2022/11/15/2022", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "637c623b1b828af353855e0", "fecha": "2022/11/15/2022", "equipo1": "Bélgica", "equipo2": "Canadá", "marcador1": 3, "marcador2": 2, "tipoevento": "Fútbol"}, {"_id": "637c624b31b828af353855e2", "fecha": "2022/11/15/2022", "equipo1": "Serbia", "equipo2": "Sri Lanka", "marcador1": 1, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "637c6281b1b828af353855e5", "fecha": "2022/11/15/2022", "equipo1": "Brasil", "equipo2": "Serbia", "marcador1": 4, "marcador2": 0, "tipoevento": "Fútbol"}, {"_id": "637c6281b1b828af353855e9", "fecha": "2022/11/15/2022", "equipo1": "Colombia", "equipo2": "Méjico", "marcador1": 4, "marcador2": 1, "tipoevento": "Fútbol"}, {"_id": "637c62cc31b828af353855e0", "fecha": "2022/11/15/2022", "equipo1": "Francia", "equipo2": "Dinamarca", "marcador1": 3, "marcador2": 0, "tipoevento": "Fútbol"}]

```



- Mostrar un evento específico por ID en la bd: <http://localhost:8000/shevento/id>

Screenshot of Visual Studio Code showing the Thunder Client extension. A GET request is made to `http://localhost:8000/shevento/637c628131b828af353855e4`. The response status is 200 OK, size is 2.01 KB, and time is 314 ms. The response body contains two JSON objects representing events.

```

55  [
56      {
57          "_id": "637c628131b828af353855e2",
58          "fecha": "2022-11-24T00:00:00.000Z",
59          "equipo1": "Brasil",
60          "equipo2": "Serbia",
61          "marcador1": 4,
62          "marcador2": 0,
63          "tipoevento": "Fútbol",
64          "createdAt": "2022-11-22T05:46:51.684Z",
65          "updatedAt": "2022-11-22T05:46:51.684Z"
66      },
67      {
68          "_id": "637c628131b828af353855e4",
69          "fecha": "2022-11-23T00:00:00.000Z",
70          "equipo1": "Bélgica",
71          "equipo2": "Canadá",
72          "marcador1": 3,
73          "marcador2": 2,
74          "tipoevento": "Fútbol",
75          "createdAt": "2022-11-22T05:47:45.478Z",
76          "updatedAt": "2022-11-22T05:47:45.478Z"
77      }
    ]
  
```

Screenshot of Visual Studio Code showing the Thunder Client extension. A GET request is made to `http://localhost:8000/shevento/637c628131b828af353855e4`. The response status is 200 OK, size is 239 Bytes, and time is 116 ms. The response body contains a single JSON object representing an event.

```

1  {
2     "_id": "637c628131b828af353855e4",
3     "fecha": "2022-11-23T00:00:00.000Z",
4     "equipo1": "Bélgica",
5     "equipo2": "Canadá",
6     "marcador1": 3,
7     "marcador2": 2,
8     "tipoevento": "Fútbol",
9     "createdAt": "2022-11-22T05:47:45.478Z",
10    "updatedAt": "2022-11-22T05:47:45.478Z"
11 }
  
```

Screenshot of a web browser showing the result of the GET request. The URL is `localhost:8000/shevento/637c628131b828af353855e4`. The response is a JSON object:

```

{"_id": "637c628131b828af353855e4", "fecha": "2022-11-23T00:00:00.000Z", "equipo1": "Bélgica", "equipo2": "Canadá", "marcador1": 3, "marcador2": 2, "tipoevento": "Fútbol", "createdAt": "2022-11-22T05:47:45.478Z", "updatedAt": "2022-11-22T05:47:45.478Z"}
  
```



- Eliminar un evento registrado en la bd: <http://localhost:8000/delevento/:id>

THUNDER CLIENT ... JS routes.js TC New Request

New Request DELETE http://localhost:8000/delevento/637c629f31b828af353855e6 Send

Activity Collections Env Query Headers² Auth Body¹ Tests Pre Run New

filter activity

Query Parameters

parameter value

Status: 200 OK Size: 2.01 KB Time: 109 ms

Response	Headers ⁷	Cookies	Results	Docs	⋮
<pre> 69 "equipo1": "Bélgica", 70 "equipo2": "Canadá", 71 "marcador1": 3, 72 "marcador2": 2, 73 "tipoevento": "Fútbol", 74 "createdAt": "2022-11-22T05:47:45.478Z", 75 "updatedAt": "2022-11-22T05:47:45.478Z" 76 }, 77 { 78 "_id": "637c629f31b828af353855e6", 79 "fecha": "2022-11-23T00:00:00.000Z", 80 "equipo1": "Alemania", 81 "equipo2": "Japón", 82 "marcador1": 4, 83 "marcador2": 1, 84 "tipoevento": "Fútbol", 85 "createdAt": "2022-11-22T05:48:15.030Z", 86 "updatedAt": "2022-11-22T05:48:15.030Z" 87 }, 88 { 89 "_id": "637c62cc31b828af353855e8", 90 "fecha": "2022-11-26T00:00:00.000Z", 91 "equipo1": "Francia", 92 "equipo2": "Dinamarca", 93 "marcador1": 3, 94 "marcador2": 0, 95 "tipoevento": "Fútbol", 96 "createdAt": "2022-11-22T05:49:00.168Z", 97 "updatedAt": "2022-11-22T05:49:00.168Z" 98 }</pre>					

Response Headers⁷ Cookies Results Docs ⋮

Go Live ⌂ ⌂ ⌂ ⌂

THUNDER CLIENT ... JS routes.js TC New Request

New Request DELETE http://localhost:8000/delevento/637c629f31b828af353855e6 Send

Activity Collections Env Query Headers² Auth Body¹ Tests Pre Run New

filter activity

Query Parameters

parameter value

Status: 200 OK Size: 38 Bytes Time: 101 ms

Response	Headers ⁷	Cookies	Results	Docs	⋮
<pre> 1 { 2 "acknowledged": true, 3 "deletedCount": 1 4 }</pre>					

Response Headers⁷ Cookies Results Docs ⋮

Go Live ⌂ ⌂ ⌂ ⌂



The screenshot shows a Visual Studio Code interface with a REST client extension open. The title bar reads "New Request - Untitled (Workspace) - Visual Studio Code". The main area displays a request to "http://localhost:8000/shevento" via a GET method. The response status is 200 OK, size is 1.77 KB, and time is 142 ms. The response body contains two JSON objects representing events:

```
65 },
66 {
67   "_id": "637c628131b828af353855e4",
68   "fecha": "2022-11-23T00:00:00.000Z",
69   "equipo1": "Bélgica",
70   "equipo2": "Canadá",
71   "marcador1": 3,
72   "marcador2": 2,
73   "tipoevento": "Fútbol",
74   "createdAt": "2022-11-22T05:47:45.478Z",
75   "updatedAt": "2022-11-22T05:47:45.478Z"
76 },
77 {
78   "_id": "637c62cc31b828af353855e8",
79   "fecha": "2022-11-26T00:00:00.000Z",
80   "equipo1": "Francia",
81   "equipo2": "Dinamarca",
82   "marcador1": 3,
83   "marcador2": 0,
84   "tipoevento": "Fútbol",
85   "createdAt": "2022-11-22T05:49:00.168Z",
86   "updatedAt": "2022-11-22T05:49:00.168Z"
87 }
```

- Actualizar un evento registrado en la bd: <http://localhost:8000/upevento/:id>

New Request - Untitled (Workspace) - Visual Studio Code

File Edit Selection View Go Run Terminal Help

THUNDER CLIENT

JS routes.js TC New Request

PUT http://localhost:8000/upevento/637c67f931b828af353855f1 Send

Activity Collections Env

Query Headers² Auth Body¹ Tests Pre Run New

filter activity

GET localhost:8000/shevento just now

POST localhost:8000/regevento just now

DEL localhost:8000/delevento/637c6... just now

GET localhost:8000/shevento/ just now

GET localhost:8000/shevento/637c6... just now

POST localhost:8000/shevento just now

GET localhost:8000/regevento just now

GET localhost:8000/showuser 9 mins ago

PUT localhost:8000/upuser/63791ae... 10 mins ago

Query Parameters

parameter value

Status: 200 OK Size: 2.01 KB Time: 107 ms

Response Headers⁷ Cookies Results Docs

```
76 },
77 {
78   "_id": "637c67f931b828af353855e8",
79   "fecha": "2022-11-26T00:00:00.000Z",
80   "equipo1": "Francia",
81   "equipo2": "Dinamarca",
82   "marcador1": 3,
83   "marcador2": 0,
84   "tipoevento": "Fútbol",
85   "createdAt": "2022-11-22T05:49:00.168Z",
86   "updatedAt": "2022-11-22T05:49:00.168Z"
87 },
88 {
89   "_id": "637c67f931b828af353855f1",
90   "fecha": "2022-11-21T00:00:00.000Z",
91   "equipo1": "Inglaterra",
92   "equipo2": "Irán",
93   "marcador1": 0,
94   "marcador2": 5,
95   "tipoevento": "Fútbol",
96   "createdAt": "2022-11-22T06:11:05.747Z",
97   "updatedAt": "2022-11-22T06:11:05.747Z"
98 }
99 ]
```



New Request - Untitled (Workspace) - Visual Studio Code

THUNDER CLIENT ... JS routes.js New Request X

PUT http://localhost:8000/upevento/637c67931b828af353855f1 Send

Activity Collections Env

filter activity

PUT localhost:8000/upevento/637c6... just now

GET localhost:8000/shevento just now

POST localhost:8000/regevento just now

DEL localhost:8000/delevento/637c6... just now

GET localhost:8000/shevento/637c6... just now

POST localhost:8000/shevento just now

GET localhost:8000/regevento just now

GET localhost:8000/showuser 9 mins ago

Live Share

Status: 200 OK Size: 92 Bytes Time: 117 ms

Response Headers Cookies Results Docs

```
1 { "acknowledged": true,
2 "modifiedCount": 1,
3 "upsertedId": null,
4 "upsertedCount": 0,
5 "matchedCount": 1
6 }
```

New Request - Untitled (Workspace) - Visual Studio Code

THUNDER CLIENT ... JS routes.js New Request X

GET http://localhost:8000/shevento Send

Activity Collections Env

filter activity

GET localhost:8000/shevento just now

PUT localhost:8000/upevento/637c6... just now

POST localhost:8000/regevento just now

DEL localhost:8000/delevento/637c6... just now

GET localhost:8000/shevento/637c6... just now

POST localhost:8000/shevento just now

GET localhost:8000/regevento just now

GET localhost:8000/showuser 9 mins ago

Live Share

Status: 200 OK Size: 2.01 KB Time: 105 ms

Response Headers Cookies Results Docs

```
76 },
77 {
78   "_id": "637c62cc31b828af353855e8",
79   "fecha": "2022-11-26T00:00:00.000Z",
80   "equipo1": "Francia",
81   "equipo2": "Dinamarca",
82   "marcador1": 3,
83   "marcador2": 0,
84   "tipoevento": "Fútbol",
85   "createdAt": "2022-11-22T05:49:00.168Z",
86   "updatedAt": "2022-11-22T05:49:00.168Z"
87 },
88 [
89   {
90     "_id": "637c67f931b828af353855f1",
91     "fecha": "2022-11-21T00:00:00.000Z",
92     "equipo1": "Inglaterra",
93     "equipo2": "Irán",
94     "marcador1": 6,
95     "marcador2": 2,
96     "tipoevento": "Fútbol",
97     "createdAt": "2022-11-22T06:11:05.747Z",
98     "updatedAt": "2022-11-22T06:14:05.560Z"
99   ]
100 ]
```



Conexión en MongoDB Shell a través de Windows Powershell

```
mongosh mongodb+srv://<credentials>@cluster0.36opdu7.mongodb.net/dbeventos
PS C:\Users\Luz Stella Alvarez\Downloads\programas de programación\mongosh-1.6.0-win32-x64\mongosh-1.6.0-win32-x64\bis .\mongosh "mong
odb+srv://<cluster0.36opdu7.mongodb.net/dbeventos" --apiVersion 1 --username mariacamilaea
Enter password: *****
Current Mongosh Log ID: 6373152cb311928c1f8ecc01
Connecting to: mongosh+srv://<credentials>@cluster0.36opdu7.mongodb.net/dbeventos?appName=mongosh+1.6.0
Using MongoDB: 5.0.13 (API Version 1)
Using Mongosh: 1.6.0

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

Atlas atlas-mtqb83-shard-0 [primary] dbeventos> use admin
switched to db admin
Atlas atlas-mtqb83-shard-0 [primary] admin> show collections

Atlas atlas-mtqb83-shard-0 [primary] admin> use dbeventos
switched to db dbeventos
Atlas atlas-mtqb83-shard-0 [primary] dbeventos> show collections
  eventos
  users
Atlas atlas-mtqb83-shard-0 [primary] dbeventos> db.users.find
[Function: find] AsyncFunction {
  returnsPromise: true,
  apiVersions: [ 1, Infinity ],
  returnType: 'cursor',
  serverVersions: [ '0.0.0', '999.999.999' ],
  topologies: [ 'RepSet', 'Sharded', 'LoadBalanced', 'Standalone' ],
  deprecated: false,
  platforms: [ 'Compass', 'Browser', 'CLI' ],
  isDirectShellCommand: false,
  acceptsRawInput: false,
  shellCommandCompleter: undefined,
  help: [Function (anonymous)] Help
}
Atlas atlas-mtqb83-shard-0 [primary] dbeventos> db.users.find()
[
  {
    _id: ObjectId("637305fdb0ac8d54723b20f3"),
    createdAt: ISODate("2022-11-15T03:22:37.598Z"),
    updatedAt: ISODate("2022-11-15T03:22:37.598Z")
  },
  {
    _id: ObjectId("637311e403a9b62cc623f220"),
    nomuser: 'Juan',
    password: '123calod1',
    correo: 'juandy@gmail.com'
  }
]
Atlas atlas-mtqb83-shard-0 [primary] dbeventos>

mongosh mongodb+srv://<credentials>@cluster0.36opdu7.mongodb.net/dbeventos
returnsPromise: true,
apiVersions: [ 1, Infinity ],
returnType: 'cursor',
serverVersions: [ '0.0.0', '999.999.999' ],
topologies: [ 'RepSet', 'Sharded', 'LoadBalanced', 'Standalone' ],
deprecated: false,
platforms: [ 'Compass', 'Browser', 'CLI' ],
isDirectShellCommand: false,
acceptsRawInput: false,
shellCommandCompleter: undefined,
help: [Function (anonymous)] Help
}
Atlas atlas-mtqb83-shard-0 [primary] dbeventos> db.users.find()
[
  {
    _id: ObjectId("637305fdb0ac8d54723b20f3"),
    createdAt: ISODate("2022-11-15T03:22:37.598Z"),
    updatedAt: ISODate("2022-11-15T03:22:37.598Z")
  },
  {
    _id: ObjectId("637311e403a9b62cc623f220"),
    nomuser: 'Juan',
    password: '123calod1',
    correo: 'juandy@gmail.com'
  },
  {
    _id: ObjectId("637312b403a9b62cc623f226"),
    nomuser: 'Gabriela',
    password: 'gabbcdshf1d1',
    correo: 'gabvcdshf1d1@gmail.com',
    createdAt: ISODate("2022-11-15T04:16:52.138Z"),
    updatedAt: ISODate("2022-11-15T04:16:52.138Z")
  },
  {
    _id: ObjectId("6373134803a9b62cc623f22a"),
    nomuser: 'Eric Matheus',
    password: 'miteoushi1',
    correo: 'erimateoushi4@gmail.com',
    createdAt: ISODate("2022-11-15T04:19:20.827Z"),
    updatedAt: ISODate("2022-11-15T04:19:20.827Z")
  }
]
Atlas atlas-mtqb83-shard-0 [primary] dbeventos>
```



Evidencia GitLab o GitHub

Evidencia de la realización de alguna actualización (commit), donde se visualice la actualización y el historial de actualizaciones (Versión).

Link: <https://github.com/lefcelis/Backend-Marcadores-Deportivos.git>

The screenshot shows the GitHub repository page for 'Backend-Marcadores-Deportivos'. The main branch is 'main'. There is one commit by 'mariacamilaea' titled 'Acta. Formato Sprint_2' made 7 hours ago. The commit message indicates the addition of the Backend for the Copa Mundial de la FIFA Qatar 2022. Other files listed include Acta_Sprint2, Sprint1, controllers, models, node_modules, routes, .dockerignore, gitignore, Dockerfile, README.md, index.js, mongodbjs, package-lock.json, package.json, and vercel.json. The repository has 14 commits in total.

Historial de actualizaciones:

The screenshot shows the commit history for the 'main' branch on November 21, 2022. It includes five commits:

- 'Acta. Formato Sprint_2' by 'mariacamilaea' committed 7 hours ago (Verified, 7b1bb30)
- 'Sprint' by 'mariacamilaea' committed 7 hours ago (Verified, 72e5389)
- 'Update README.md' by 'lefcelis' committed 16 hours ago (Verified, 7eb8133)
- 'Excluded .env' by 'lefcelis' committed 17 hours ago (b3dc27f)

The screenshot shows a GitHub repository page for 'Backend-Marcadores-Deportivos' at the 'commits/main' branch. At the top, there are logos for 'Misión TIC 2022', 'UNAB', and 'Universidad de los Andes'. The commit history is organized by date:

- Commits on Nov 20, 2022:**
 - Update README.md (Verified, d284070)
 - Actualización database (28371b5)
 - Se agrega Backend del proyecto UNAB Ciclo 4 (b94e46c)
 - Se agrega Backend del proyecto (4df5c10)
- Commits on Nov 19, 2022:**
 - Se modifica Controllers para Eventos y Usuarios (1f54de8)
 - Controllers (d091f92)
- Commits on Nov 18, 2022:**
 - Creamos Servidor, Add Rutas, Modelos, Conexión a MongoDB (4eca864)

The screenshot shows a GitHub repository page for 'Backend-Marcadores-Deportivos' at the 'commits/main' branch. The commit history is organized by date:

- Commits on Nov 18, 2022:**
 - Creamos Servidor, Add Rutas, Modelos, Conexión a MongoDB (4eca864)
- Commits on Nov 15, 2022:**
 - Actualización archivo Vercel (ac671c3)
 - Se actualiza archivo de Vercel (bc3360b)
- Commits on Nov 14, 2022:**
 - Se agrega el contenido de la carpeta DockerG51 (2586160)

Evidencia JIRA (Seguimiento del proyecto)

Como evidencia del seguimiento del proyecto con la metodología ágil SCRUM, utilizando el software JIRA, se debe presentar capturas de pantalla donde se visualice la ejecución de los Sprint con las historias de usuario relacionadas con el desarrollo del Backend.

Link: <https://g52c3.atlassian.net/jira/software/projects/GRUPO51/boards/3/backlog>

Evidencias de las Reuniones de Equipo

Como evidencia de las reuniones que efectúa el equipo del proyecto, presentar capturas de pantalla de las reuniones efectuadas y si lo consideran pertinente algunas actas de las reuniones.

Reunión Scrum. Ciclo 4 Mintic

Domingo, 13 de noviembre · 5:00 – 7:00pm

Información para unirse a Google Meet

Enlace a la videollamada: <https://meet.google.com/suz-dwxx-qxg>

