



SPRINT 3: Desarrollo del Frontend

Identificación Proyecto	
Nombre Proyecto:	Deportes
Número Equipo:	02
Integrantes del equipo	
Rol (Líder-Desarrollador – Cliente)	Nombre
Líder	Doris Aroca
Desarrollador	Jorge Moreno
Desarrollador	Juan Álvarez
Desarrollador	Felipe Álvarez
Desarrollador	Brandon Felipe Ortiz
Desarrollador	Christian Felipe mora
Cliente	Álvaro Stid Bolaños Vásquez

Evidencia construcción del Frontend

Como evidencia de la construcción del Frontend, se debe presentar capturas de pantalla donde se visualice el proceso de construcción del Frontend.

Al momento de codificar el frontend se usó código de Bootstrap para agregar las líneas que se observan en el funcionamiento el Frontend, cabe recordar que la base de datos que fue usada en el backend de este proyecto es MONGODB

En las primeras imágenes se evidencia el código de Frontend en la categoría “Componets”



MenuAdmin.js

```

File Edit Selection View Go Run Terminal Help
project015-g2

EXPLORER
  PROJECT015-G2
    node_modules
    public
    src
      components
        MenuAdmin.js
        MenuInicio.js
        PageDeportes.js
        PageEquipos.js
        PageEventos.js
        PageInicio.js
        PageLogin.js
        PageUsuarios.js
        Saludar.js
      css
      imagenes
    App.css
    App.js
    App.test.js
    index.css
    index.js
    logo.svg
    reportWebVitals.js
    setupTests.js
    package-lock.json
    package.json
    README.md

OUTLINE
TIMELINE

src > components > MenuAdmin.js
1 import React from "react"
2 import 'bootstrap/dist/css/bootstrap.min.css'
3 import 'bootstrap/dist/js/collapse'
4 import 'bootstrap/js/dist/offcanvas'
5 import 'bootstrap/js/dist/dropdown'
6 import { Link } from 'react-router-dom'
7
8 function MenuAdmin() {
9
10
11   return (
12     <nav className="navbar navbar-expand-lg navbar-dark bg-dark">
13       <div className="container-fluid">
14         <Link className="navbar-brand" to="/" />
15         
16         </Link>
17         <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedConte
18         <span className="navbar-toggler-icon"></span>
19         </button>
20         <div className="collapse navbar-collapse" id="#navbarSupportedContent">
21           <ul className="navbar-nav mx-auto">
22             <li className="nav-item">
23               <Link className="nav-link" aria-current="page" to="/" />Inicio</Link>
24             </li>
25             <li className="nav-item">
26               <Link className="nav-link" to="/PageUsuarios">Usuarios</Link>
27             </li>
28             <li className="nav-item">
29               <Link className="nav-link" to="/PageEventos">Eventos</Link>
30             </li>
31           </ul>
32         </div>
33       </div>
34     </nav>
35   )
36 }

```

MenuInicial.JS

```

File Edit Selection View Go Run Terminal Help
project015-g2

EXPLORER
  PROJECT015-G2
    node_modules
    public
    src
      components
        MenuAdmin.js
        MenuInicio.js
        PageDeportes.js
        PageEquipos.js
        PageEventos.js
        PageInicio.js
        PageLogin.js
        PageUsuarios.js
        Saludar.js
      css
      imagenes
    App.css
    App.js
    App.test.js
    index.css
    index.js
    logo.svg
    reportWebVitals.js
    setupTests.js
    package-lock.json
    package.json
    README.md

OUTLINE
TIMELINE

src > components > MenuInicial.js
1 import React, { Component } from "react"
2 import 'bootstrap/dist/css/bootstrap.min.css'
3 import 'bootstrap/dist/js/collapse'
4 import 'bootstrap/js/dist/offcanvas'
5 import 'bootstrap/js/dist/dropdown'
6 import { Link } from 'react-router-dom'
7
8
9 import Cookies from 'universal-cookie'
10 const cookies = new Cookies()
11
12
13 class MenuInicial extends Component {
14   state = {
15     estaLoguin: false
16   }
17
18   componentDidMount() {
19     if (cookies.get("usu_nombres")) {
20       this.setState({ estaLoguin: true })
21     } else {
22       this.setState({ estaLoguin: false })
23       // window.location.href="/" // redirigir al inicio
24     }
25   }
26
27   cerrarSesion() {
28     cookies.remove("usu_id", { path: "/" })
29   }
30 }

```



PageDeporte.js

```
File Edit Selection View Go Run Terminal Help
project015-g2

EXPLORER
  PROJECT015...
    > node_modules
    > public
    > src
      > components
        JS MenuAdmin.js
        JS MenuInicial.js
        JS PageDeportes.js
        JS PageEquipos.js
        JS PageEventos.js
        JS PageInicio.js
        JS PageLogin.js
        JS PageUsuarios.js
        JS Saludar.js
      > css
      > imagenes
        # App.css
        JS App.js
        JS App.test.js
        # index.css
        JS index.js
        logo.svg
        JS reportWebVitals.js
        JS setupTests.js
        package-lock.json
        package.json
        README.md
      > OUTLINE
      > TIMELINE

PageDeportes.js X
src > components > JS PageDeportes.js > ...
1 import React, { Component } from "react";
2 import axios from "axios";
3 import "bootstrap/dist/css/bootstrap.min.css";
4 import { FontAwesomeIcon } from "@fortawesome/react-fontawesome";
5 import { faEdit, faTrashAlt } from "@fortawesome/free-solid-svg-icons";
6 import { Modal, ModalBody, ModalFooter, ModalHeader } from "reactstrap";
7
8
9 const url = "http://localhost:5000/deportes"
10
11 class PageDeportes extends Component {
12   state = {
13     data: [],
14     modalInsertar: false,
15     modalEliminar: false,
16     tipoModal: '',
17     form: {
18       _id: '',
19       dep_nombre: '',
20     }
21   };
22
23   peticionGet = () => {
24     axios.get(url).then(response => {
25       this.setState({ data: response.data });
26       //console.log(response.data);
27     }).catch(error => {
28

```

PageEventos.js

```
File Edit Selection View Go Run Terminal Help
project015-g2

EXPLORER
  PROJECT015...
    > node_modules
    > public
    > src
      > components
        JS MenuAdmin.js
        JS MenuInicial.js
        JS PageDeportes.js
        JS PageEquipos.js
        JS PageEventos.js
        JS PageInicio.js
        JS PageLogin.js
        JS PageUsuarios.js
        JS Saludar.js
      > css
      > imagenes
        # App.css
        JS App.js
        JS App.test.js
        # index.css
        JS index.js
        logo.svg
        JS reportWebVitals.js
        JS setupTests.js
        package-lock.json
        package.json
        README.md
      > OUTLINE
      > TIMELINE

PageEquipos.js X
src > components > JS PageEquipos.js > ...
1 import React, { Component } from "react";
2 import axios from "axios";
3 import "bootstrap/dist/css/bootstrap.min.css";
4 import { FontAwesomeIcon } from "@fortawesome/react-fontawesome";
5 import { faEdit, faTrashAlt } from "@fortawesome/free-solid-svg-icons";
6 import { Modal, ModalBody, ModalFooter, ModalHeader } from "reactstrap";
7
8
9 const url = "http://localhost:5000/equipos"
10
11 class PageEquipos extends Component {
12   state = {
13     data: [],
14     modalInsertar: false,
15     modalEliminar: false,
16     tipoModal: '',
17     form: {
18       _id: '',
19       equ_nombre: '',
20     }
21   };
22
23   peticionGet = () => {
24     axios.get(url).then(response => {
25       this.setState({ data: response.data });
26       //console.log(response.data);
27     }).catch(error => {
28

```



PageEventos.js

```

1  import React, { Component } from 'react';
2  import "bootstrap/dist/css/bootstrap.min.css";
3  import { FontAwesomeIcon } from '@fortawesome/react-fontawesome';
4  import { faEdit, faTrashAlt } from '@fortawesome/free-solid-svg-icons';
5  import { Modal, ModalBody, ModalFooter, ModalHeader } from 'reactstrap';
6  import axios from 'axios';
7
8  const url = 'http://localhost:5000/eventos';
9
10 class PageEventos extends Component {
11   state = {
12     data: [],
13     modalInsertar: false,
14     modalEliminar: false,
15     tipoModal: '',
16     form: {
17       _id: '',
18       eve_fecha: '',
19       equ_equipo1: '',
20       equ_equipo2: '',
21       eve_marca1: '',
22       eve_marca2: '',
23       dep_id: '',
24       eve_descrip: '',
25     },
26     equipos: [],
27     deportes: []
28   }

```

PageInicio.js

```

1  import React, { Component } from 'react'
2
3  import axios from 'axios';
4  import "bootstrap/dist/css/bootstrap.min.css";
5
6
7  const urlEventos = 'http://localhost:5000/eventos/home'
8
9  class PageInicio extends Component{
10
11   state={
12     data: []
13   }
14
15   petitionGet = () => {
16     axios.get(urlEventos).then(response => {
17       //console.log(response.data);
18       this.setState({data:response.data})
19     }).catch(error => {
20       console.log(error.message);
21     })
22   }
23
24   componentDidMount(){
25     this.petitionGet()
26   }
27
28

```




PageLogin.js

The screenshot shows the Visual Studio Code editor with the file `PageLogin.js` open. The code is as follows:

```

1 import React, { Component } from 'react'
2 import '../css/Login.css'
3 import '../imagenes/logo.jpg'
4 import 'bootstrap/dist/css/bootstrap.min.css'
5 import axios from 'axios'
6 import Cookies from 'universal-cookie'
7
8 const urlLogin = "http://localhost:5000/usuarios"
9 const cookies = new Cookies();
10
11 class PageLogin extends Component {
12   state = {
13     form: {
14       email: '',
15       password: '',
16     }
17   }
18
19   handleChange = async e => {
20     await this.setState({
21       form: {
22         ...this.state.form,
23         [e.target.name]: e.target.value
24       }
25     })
26     //console.log(this.state.form)
27   }
28 }

```

The Explorer panel on the left shows the project structure, with `PageLogin.js` selected under the `components` folder.

PageUsuarios.js

The screenshot shows the Visual Studio Code editor with the file `PageUsuarios.js` open. The code is as follows:

```

1 import React, { Component } from 'react';
2 import axios from 'axios';
3 import 'bootstrap/dist/css/bootstrap.min.css';
4 import { FontAwesomeIcon } from '@fortawesome/react-fontawesome';
5 import { faEdit, faTrashAlt } from '@fortawesome/free-solid-svg-icons';
6 import { Modal, ModalBody, ModalFooter, ModalHeader } from 'reactstrap';
7 import { config } from '@fortawesome/fontawesome-svg-core';
8
9 const url = "http://localhost:5000/usuarios"
10
11 class PageUsuarios extends Component {
12   state = {
13     data: [],
14     modalResetar: false,
15     modalEliminar: false,
16     tipoModal: '',
17     form: {
18       _id: '',
19       usu_email: '',
20       usu_clave: '',
21       usu_nombres: '',
22       usu_apellidos: '',
23     }
24   }
25
26   petitionGet = () => {
27     axios.get(url).then(response => {
28

```

The Explorer panel on the left shows the project structure, with `PageUsuarios.js` selected under the `components` folder.



Saludar.js

A screenshot of the Visual Studio Code editor showing the 'Saludar.js' file. The Explorer sidebar on the left shows the project structure with 'Saludar.js' selected under the 'components' folder. The main editor area displays the following code:

```
1 import React from "react"
2
3
4 function Saludar(props){
5   return <div>
6     <h1>{props.title}</h1>
7   </div>
8 }
9
10 export default Saludar
```

The bottom status bar shows the time as 06:00 and the file is located at 'Ln 1, Col 1'.

Las siguientes evidencias son frontend al “app css” y el “app.js”:

App.css

A screenshot of the Visual Studio Code editor showing the 'App.css' file. The Explorer sidebar on the left shows the project structure with 'App.css' selected under the 'css' folder. The main editor area displays the following CSS code:

```
1 .App {
2   text-align: center;
3 }
4
5 .App-logo {
6   height: 40vmin;
7   pointer-events: none;
8 }
9
10 @media (prefers-reduced-motion: no-preference) {
11   .App-logo {
12     animation: App-logo-spin infinite 20s linear;
13   }
14 }
15
16 .App-header {
17   background-color: #282c34;
18   min-height: 100vh;
19   display: flex;
20   flex-direction: column;
21   align-items: center;
22   justify-content: center;
23   font-size: calc(10px + 2vmin);
24   color: white;
25 }
26
27 .App-link {
28   color: #61dafb;
```

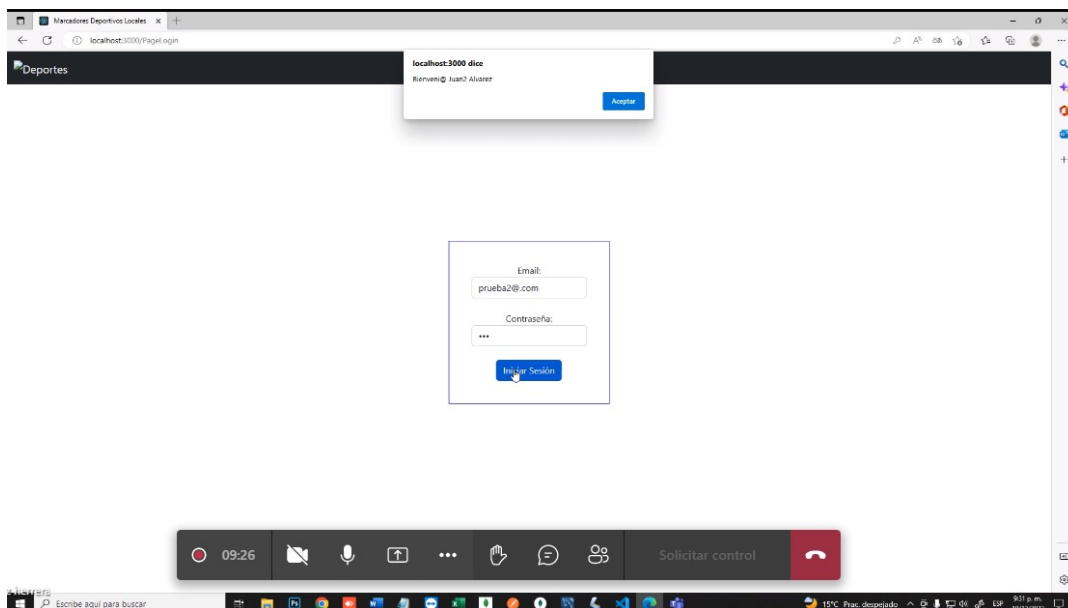
The bottom status bar shows the time as 08:03 and the file is located at 'Ln 1, Col 1'.



Como evidencia del funcionamiento del Frontend, se debe presentar capturas de pantalla donde se visualice el consumo de recursos del API REST, desde el Frontend.

Las imágenes se darán con un título para que se aprecie de manera efectiva cada sección del proyecto

LOGIN: localhost:3000/PageLogin





PAGINA PRINCIPAL: localhost:3000







The screenshot shows a web browser window with the URL 'localhost:3000'. The page title is 'Marcadores Deportivos Locales'. The main content area displays a table with the following data:

Fecha	Equipo1	Marcador	Equipo2	Deporte	Descripción
2021-11-16T13:30:00.000Z	Nacional	0 : 100	Medellin	futbol	copa final
2022-12-20T00:00:00.000Z	Nacional	2 : 1	Medellin	futbol	segunda

At the bottom of the page, there is a navigation bar with a clock showing 09:48, a microphone icon, a speech bubble icon, a 'Solicitar control' button, and a red phone icon.

USO DEL METODO DELETE Y PUT: localhost:3000/PageDeportes

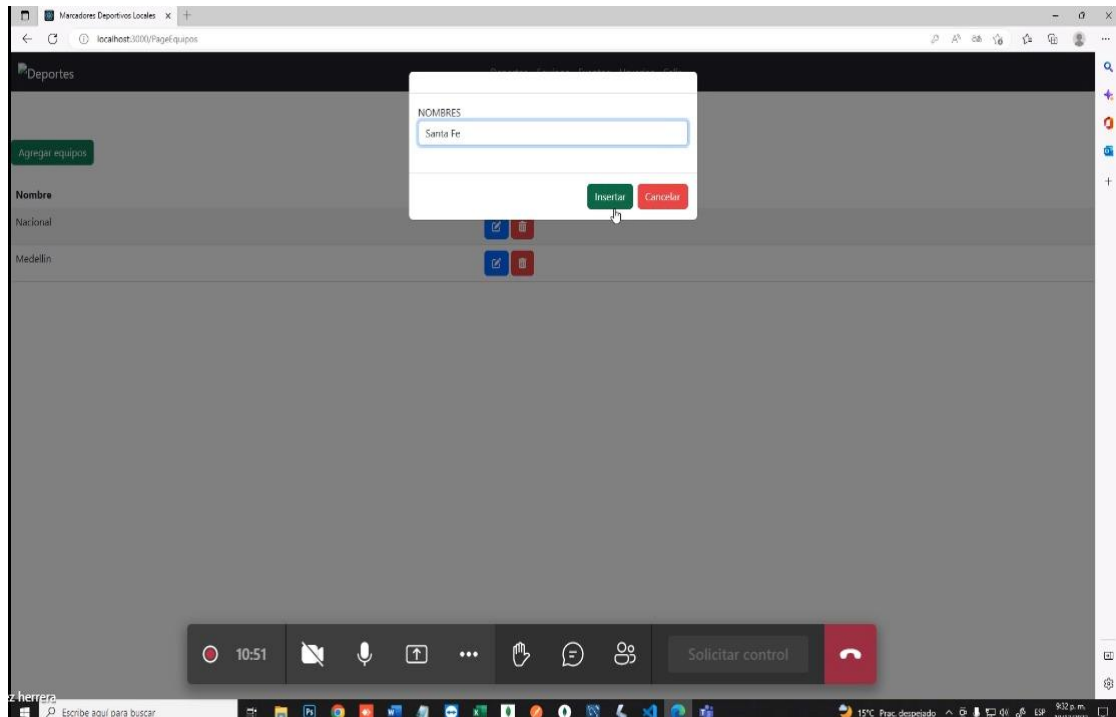
The screenshot shows the 'PageDeportes' page. A modal dialog box is displayed in the center with the text '¿está seguro de eliminar este registro?' and two buttons: 'Si' (Yes) and 'No' (No). The background is dimmed. Below the dialog, there is a table with the following data:

Nombre	
futbol	 
baloncesto	 
tiro	 

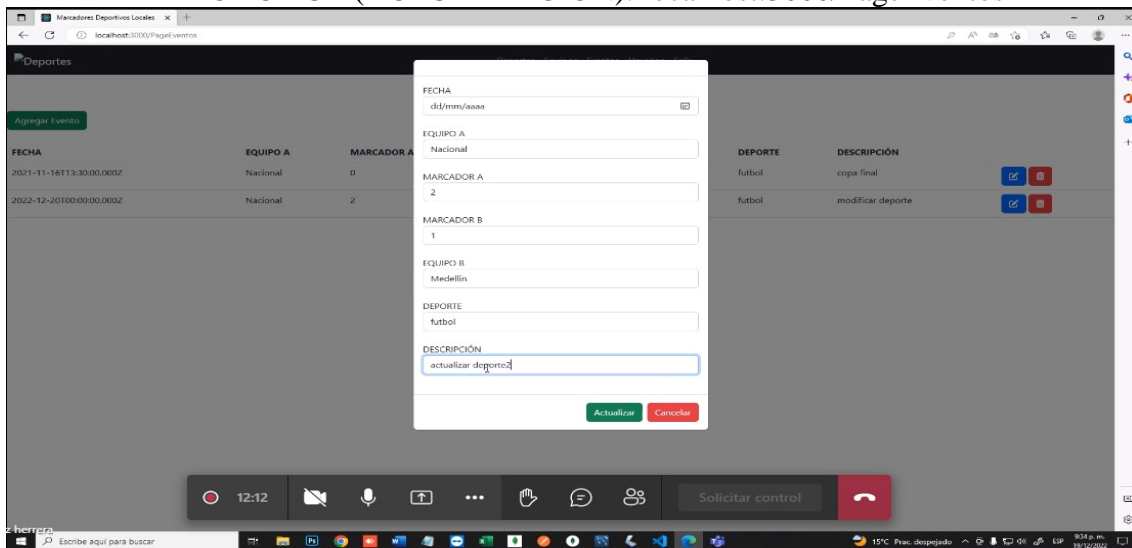
At the bottom of the page, there is a navigation bar with a clock showing 10:06, a microphone icon, a speech bubble icon, a 'Solicitar control' button, and a red phone icon.



SECCION DERPORTE: localhost:3000/PageEquipos



METODO PUT (ACTUALIZACION): localhost:3000/PageEventos





EVENTO ACTUALIZADO: localhost:3000/PageEventos

Deportes

Agregar Evento

FECHA	EQUIPO A	MARCADOR A	MARCADOR B	EQUIPO B	DEPORTE	DESCRIPCIÓN
2021-11-16T13:30:00.000Z	Nacional	2	1	Medellin	futbol	actualizar deporte2
2022-12-20T00:00:00.000Z	Nacional	2	1	Medellin	futbol	modificar deporte

13:02

Solicitar control

USUARIO: localhost:3000/PageUsuarios

Deportes

Agregar Usuario

Email	Clave
prueba2@gmail.com	123
prueba3@gmail.com	123
prueba4@gmail.com	123

EMAIL

estudianteUNAB@gmail.com

CLAVE

999999

NOMBRES

Pedro

APELLIDOS

Programador

Insertar Cancelar

14:51

Solicitar control



USUARIO CREACION: localhost:3000/PageUsuarios

The screenshot shows a web browser displaying a page titled 'Deportes' with a navigation bar containing 'Deportes', 'Equipos', 'Eventos', 'Usuarios', and 'Salir'. A green button labeled 'Agregar Usuario' is visible. Below it is a table with the following data:

Email	Clave	Nombre	Apellido
prueba2@gmail.com	123	Juan2	Alvarez
prueba3@gmail.com	123	Juan3	Alvarez3
prueba4@gmail.com	123	Juan4	Alvarez4
estudiantunab@gmail.com	999999	Pedro	Programador

Each row has two action buttons: a blue checkmark and a red delete icon. The bottom of the browser shows a Windows taskbar with the time 15:09 and a 'Solicitar control' button.

POSTMAN USUARIO:

The screenshot shows the Postman application interface. The left sidebar displays a 'Team Workspace' with a list of collections. The main area shows a REST client request to 'http://localhost:5000/usuarios' with a POST method. The request body is in JSON format:

```

{
  "usu_email": "a@yahoo",
  "usu_clave": "12345",
  "usu_nombre": "Rosa",
  "usu_apellidos": "Salvaje"
}

```

The response status is 200 OK, and the response body is:

```

{
  "message": "nuevo usuario agregado"
}

```

The bottom of the browser shows a Windows taskbar with the time 16:56 and a 'Solicitar control' button.



POST http://localhost:5000/usuarios

```

1 {
2   "usu_email": "doris@unab.com",
3   "usu_cclave": "8888",
4   "usu_nombre": "Doris",
5   "usu_apellido": "Programadora"
6 }
7
8

```

Status: 200 OK, Time: 7 ms, Size: 303 B

Body: "message": "nuevo usuario agregado"

NUEVOS USUARIOS PAGINA: localhost:3000/PageUsuarios

Deportes Deportes Equipos Eventos Usuarios Salir

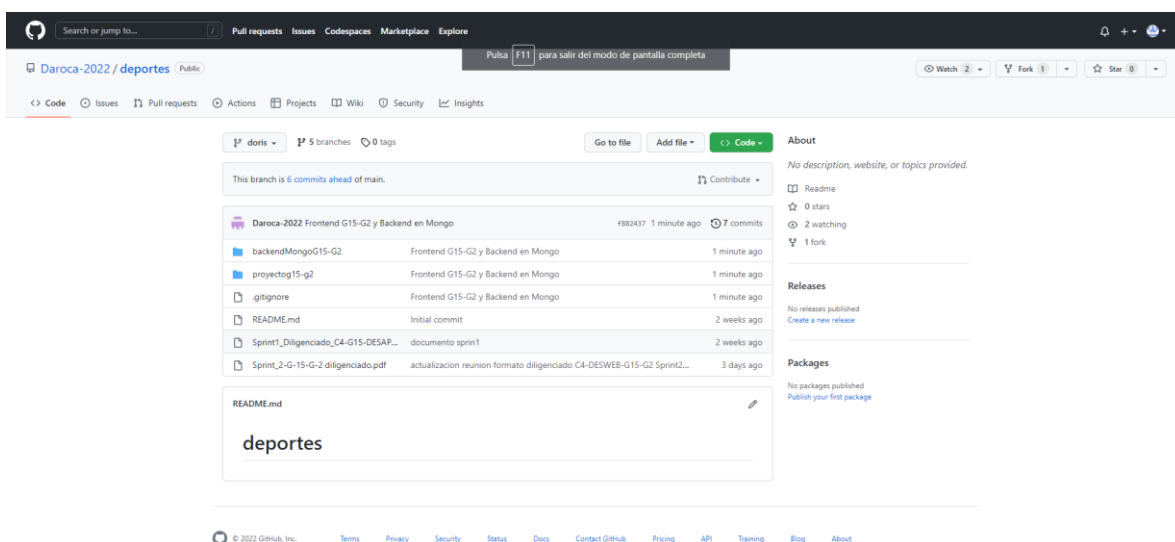
Agregar Usuario

Email	Clave	Nombre	Apellido
prueba2@gmail.com	123	Juan2	Alvarez
prueba3@gmail.com	123	Juan3	Alvarez3
prueba4@gmail.com	123	Juan4	Alvarez4
estudianteunab@gmail.com	999999	Pedro	Programador
felipe@unab.com	5555	Felipe	Programador
jorge@unab.com	7777	Jorge	Programador
doris@unab.com	8888	Doris	Programadora

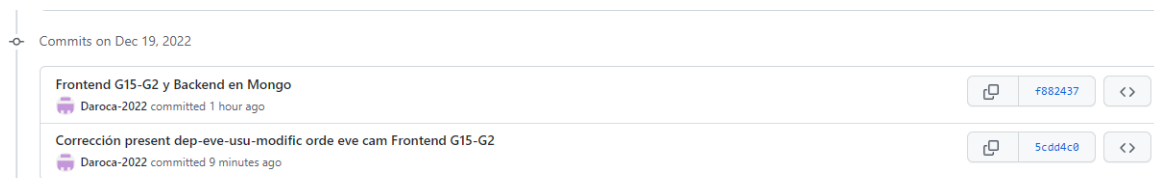


Evidencia GitLab o GitHub

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



Actualizaciones en el GitHub



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 Frontend.



Tu trabajo ▾

Proyectos ▾

Filtros ▾

Paneles ▾

Personas ▾

Aplicaciones ▾

Crear

Q

Buscar

APLICACION MARCAD...

Proyecto de software

PLANIFICACIÓN

Hoja de ruta

Backlog

Tablero

DESARROLLO

Código

Páginas de proyectos

Añadir acceso rápido

Configuración del pr...

Estás en un proyecto gestionado por el equipo

Más información

Proyectos / APLICACION MARCADOR DEPORTES

Tablero Sprint 3

Crear frontend e integrarlo al backend

Q

JC BH CS D +2

AGROUPAR POR Nada ▾

POR HACER 9 INCIDENCIAS

INGRESAR SPRINT 3 EN JIRA

☒

AMD-15

4

BH

Creación del front y actualizaciones iniciales

☒

AMD-16

4

CS

Construir una página de Home donde muestre todos los eventos

☒

AMD-18

3

D

Construir una página que permita registrarse

☒

AMD-19

3

EN CURSO 1 INCIDENCIA

Generar el componente navbar

☒

AMD-17

2

JC

LISTO 1 INCIDENCIA ✓

Construir la página inicial del usuario.

☒

AMD-21

3

JA

Completar sprint

...

Quickstart

X



APLICACION MARCADOR DEPORTES
Proyecto de software

PLANIFICACIÓN

- Hoja de ruta
- Backlog
- Tablero

DESARROLLO

- Código

- Páginas de proyectos
- Añadir acceso rápido
- Configuración del pr...

Estás en un proyecto gestionado por el equipo
Más información

Proyectos / APLICACION MARCADOR DEPORTES

Tablero Sprint 3

Crear frontend e integrarlo al backend

Search bar with icons for filters and team members.

POR HACER

EN CURSO

LISTO 11 INCIDENCIAS ✓

Ingresar Sprint 3 en JIRA

✓ AMD-15 ✓ 4 BH

Creación del front y actualizaciones iniciales

✓ AMD-16 ✓ 4 CS

Generar el componente navbar

✓ AMD-17 ✓ 2 CS

Construir una página de Home donde muestre todos los eventos

✓ AMD-18 ✓ 3 CS

Construir una página que permita registrarse

✓ AMD-19 ✓ 3 CS

Construir una página de login

✓ AMD-20 ✓ 8 CS

Construir la página inicial del usuario.

✓ AMD-21 ✓ 3 JA

Añadir las funcionalidades que le permitan al usuario actualizar,



Jira Software Tu trabajo Proyectos Filtros Paneles Personas Aplicaciones Crear

APLICACION MARCAD... Proyecto de software

PLANIFICACIÓN

- Hoja de ruta
- Backlog
- Tablero

DESARROLLO

- Código
- Páginas de proyectos
- Añadir acceso rápido
- Configuración del pr...

Proyectos / APLICACION MARCADOR DEPORTES

Backlog

Tablero Sprint 3 19 dic - 20 dic (11 incidencias) 0 10 69 Completar sprint

Crear frontend e integrarlo al backend

Checklist	Progress	Status	Assignee
AMD-15 Ingresar Sprint 3 en JIRA	4	FINALIZADA	BH
AMD-16 Creación del front y actualizaciones iniciales	4	FINALIZADA	TS
AMD-19 Construir una página que permita registrarse	3	FINALIZADA	OR
AMD-20 Construir una página de login	8	FINALIZADA	TS
AMD-21 Construir la página inicial del usuario.	3	FINALIZADA	JA
AMD-17 Generar el componente navbar	2	FINALIZADA	TS
AMD-25 Construir una página para eventos donde se podrá actualizar, borrar y, registrar estos.	10	FINALIZADA	TS
AMD-24 Construir una página para equipos donde se podrá actualizar, borrar y, registrar estos.	10	FINALIZADA	TS
AMD-18 Construir una página de Home donde muestre todos los eventos	3	FINALIZADA	TS
AMD-22 Añadir las funcionalidades que le permitan al usuario actualizar, borrar y, registrar eventos.	12	FINALIZADA	BH
AMD-23 Construir una página para deportes donde se podrá actualizar, borrar y, registrar estos.	10	EN CURSO	TS

+ Crear incidencia

Backlog (0 incidencias) 0 0 0 Crear sprint

Tu backlog está vacío.

+ Crear incidencia

Estás en un proyecto gestionado por el equipo Más información

Generar el componente navbar

AMD-17

Finalizada Listo

Descripción

Añadir una descripción...

Detalles

Responsable JORGE ANDRES MORENO CARO Asignarme a mí

Etiquetas Ninguno

Sprint Tablero Sprint 3

Story point estimate 2

Informador JA Juan Alvarez

Creado hace 14 minutos Actualizado hace 3 minutos Resuelto hace 3 minutos

Configurar

Actividad

Mostrar: Comentarios Más recientes primero

BH Añadir un comentario...

Consejo de expertos: pulsa para comentar

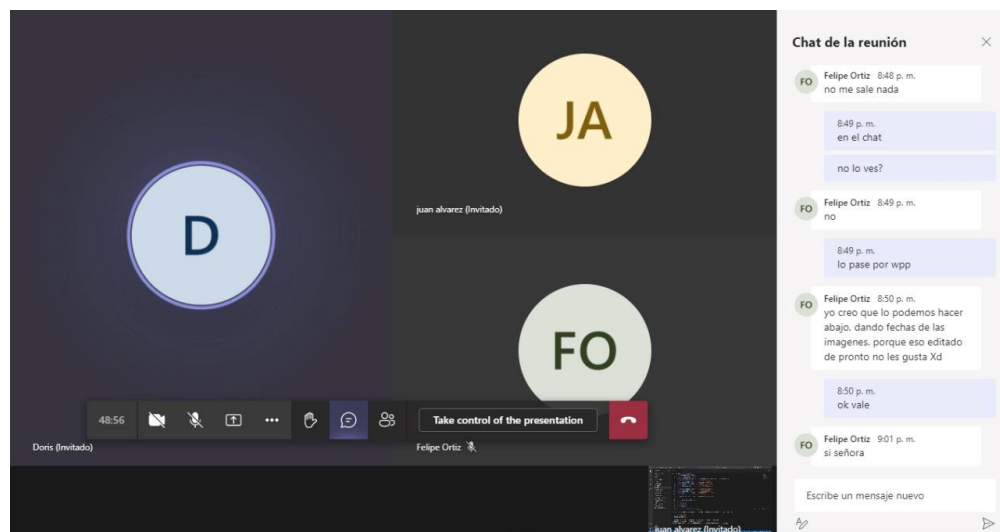
Evidencias realizadas en la plataforma JIRA, en la cual se divide el sprint 3 con todos los integrantes para realizar las tareas del proyecto. En las evidencias se demuestra la creación de las tareas y su finalización. En la primera imagen se da evidencia la división del sprint 3 para poder dividir las tareas en el JIRA y así crear las tareas para los integrantes.



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

1. FECHA: 19/12/2022

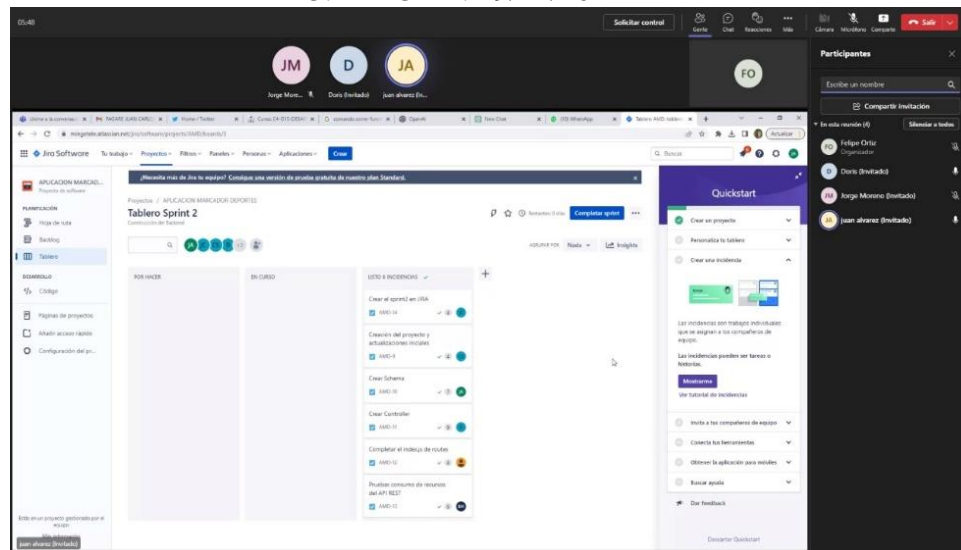




2. FECHA: 19/12/2022

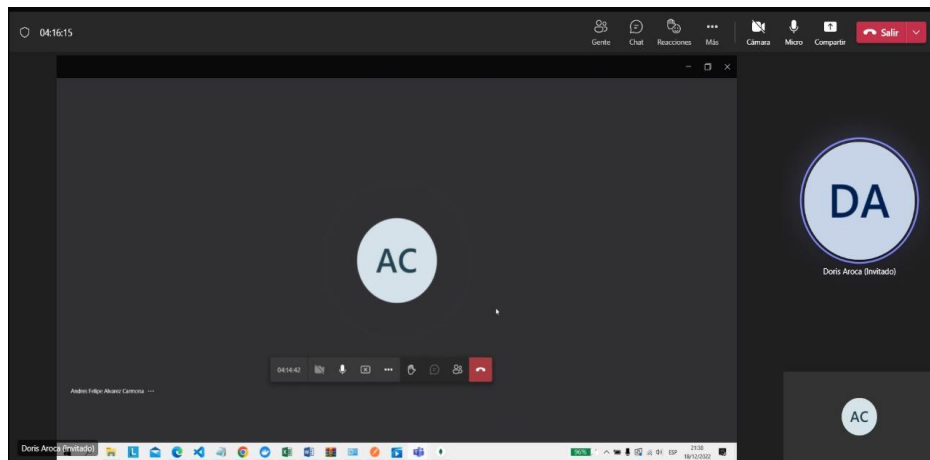


3. FECHA: 19/12/2022

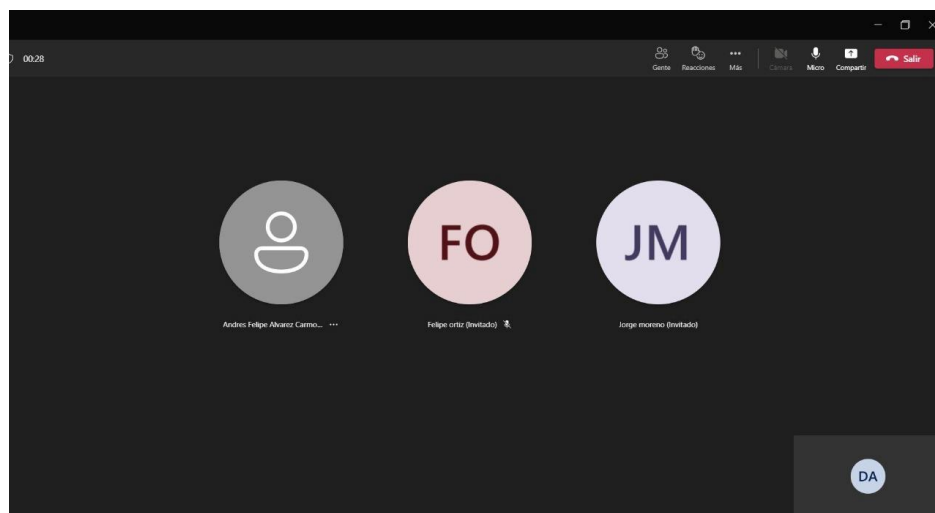




4. FECHA: 19/12/2022



5. FECHA: 19/12/2022





Evidencias de las reuniones en la cuales se trataron las formas de realizar el sprint 3, donde todo el grupo se reunió y se desglosó para la culminación del mismo.